

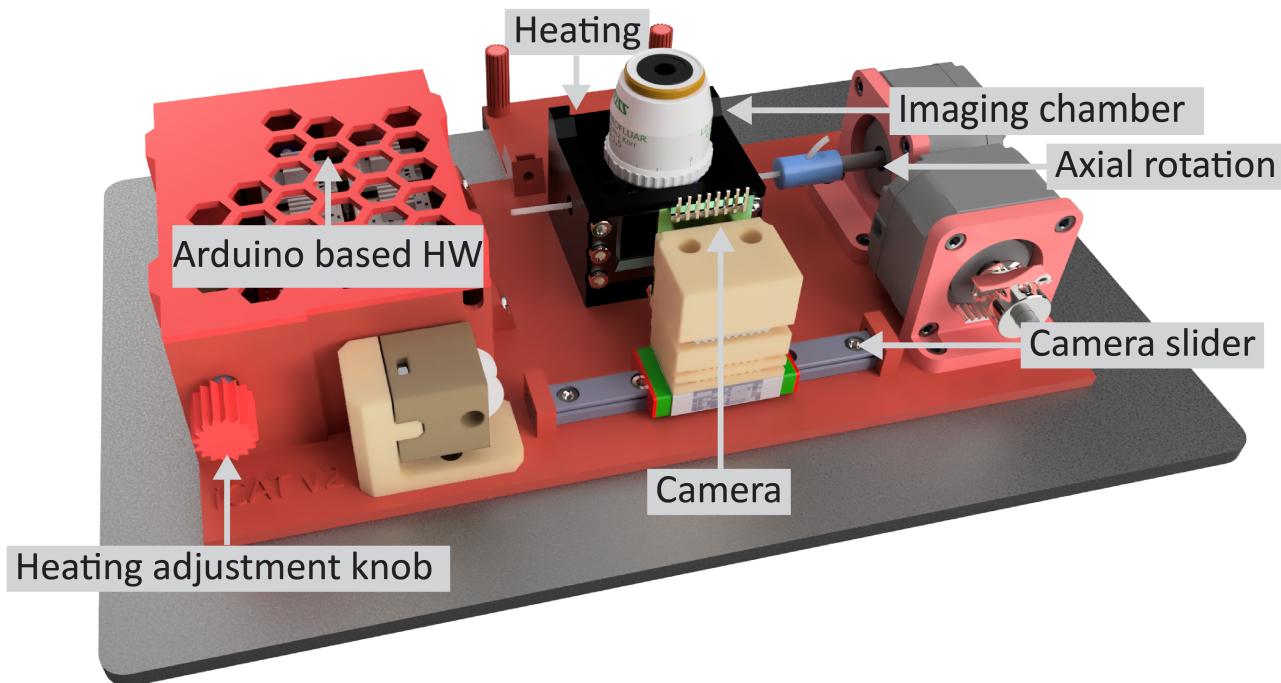
iCAT: A Multifunctional Open-Source Platform for Advanced Light Microscopy

"i" - Intuitive: iCAT represents an innovative approach to light microscopy, incorporating advanced features like intuitive axial rotation and tracking. It introduces a novel solution for imaging and studying biological specimens with enhanced capabilities, pushing the boundaries of traditional microscopy techniques.

"C" - Controlled: iCAT offers precise control over specimen rotation, temperature regulation, and lighting conditions, ensuring optimal imaging conditions and experimental reproducibility.

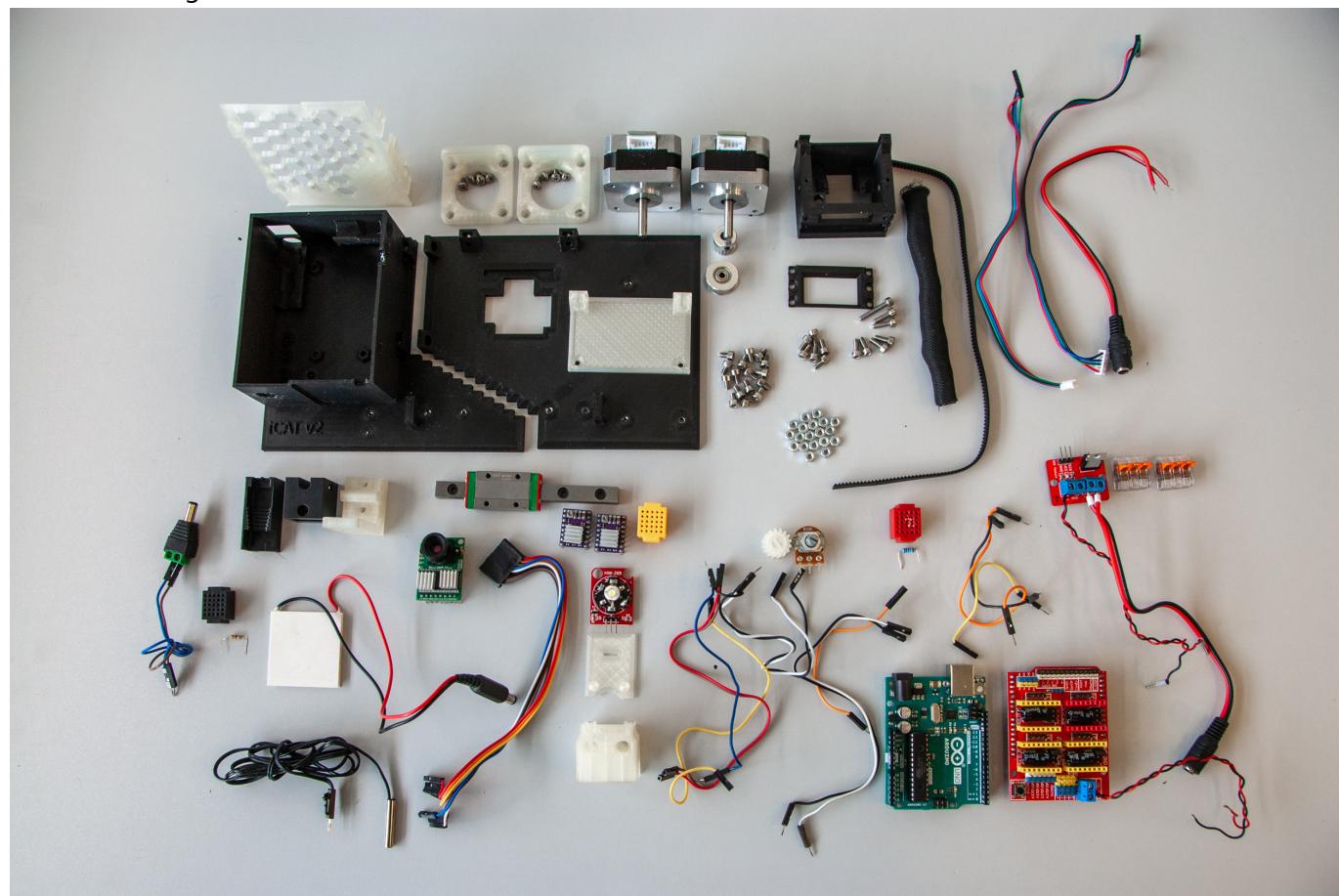
"a" - Axial: iCAT's unique capability to rotate specimens along their axial axis sets it apart from conventional light microscopy techniques, allowing researchers to observe dynamic processes and capture detailed time-lapse sequences.

"t" - Tilt: iCAT offers the unique capability of "Controlled Axial Tilt" providing researchers with a 360-degree perspective. This groundbreaking feature empowers researchers to examine specimens in three dimensions, facilitating the exploration of dynamic cellular processes, developmental biology, and beyond.



Supplies

One of the key advantages of iCAT is its open-source nature, which promotes accessibility, customization, and collaborative development. The device can be fabricated using 3D printing technology, ensuring low-cost production and easy assembly. All the electrical and mechanical components required for iCAT can be readily obtained for less than \$300, making it feasible for researchers with varying resources and expertise levels to replicate and utilize the device in their laboratories. For an overview of the required components, see the illustrative image and list below.



Electronics:

Arduino Uno Rev3, 1x
Arducam 5MP Plus OV5642, 1x
IRF520 Driver Module, 1x
CNC Shield V3, 1x
DRV8825 Stepper Motor Driver, 2x
17HS2408 Stepper Motor, 2x
Peltier plate module TEC1-12706, 1x
NTC Thermistor 10K 1% 3950, 1x
4.7 kOhm resistor, 1x
10 kOhm resistor, 1x
10K Ohm potentiometer, 1x
3W High Power LED Module LED with PCB Chassis for Arduino, 1x
12V 3A Power Supply, 1x

Accessories:

GT2 Pulley 16 Teeth 5mm bore 6 mm width, 1x
GT2 Idler 20 Teeth 5mm bore 3 mm width without teeth, 1x
GT2 Open Timing Belt 2mm Pitch 6mm Width, 1x
MGN9H 100mm linear guide rail with carriage, 1x
Stainless Steel DIN912 Hexagon Hex Socket Head Cap Allen Bolt Screw, M3x4 (22x), M3x5 (4x), M3x6 (8x), M3x14 (1x)
M3 hex nut, 20x
Microscope Cover Glass, 24x40mm, 10x
Lubricant in the syringe, 1x
Sleeving Cord Protector, 6-12 mm, 1x
Neodymium Magnet 5mm x 2mm, 8x
FEP tube clear 0.8 I.D./1.2 O.D., 10-25 m
FEP tube clear - other sizes, 5 m
Silicon rubber tube 1x2 mm, 3 m

Wiring:

Female Plug 12V DC Power Pigtail Cable Jack, 2x
Female and Male DC Connectors 2.1x5.5mm, 1x
Crimp Terminals Set Kit, 1x
Dupont Line M-M + F-M + F-F Jumper Wire 10cm, 20CM, 30x
WAGO Terminals Series Splicing Connector 221-413, 2x
Mini Solderless Prototype Breadboard, 3x
Black Jumper Caps, 2x

3D print:

3D printer, 1x
3D Printer Filament 1.75mm - transparent, black, 2x

Tools:

Pipette Pump 10ml, 1x
Borosilicate Glass Pasteur Pipettes, 1x
Crimping Tools SN-58B + 4 jaws, 1x
Screwdriver Set, 1x

NOTE: The complete shopping list is in the [components sheet](#).

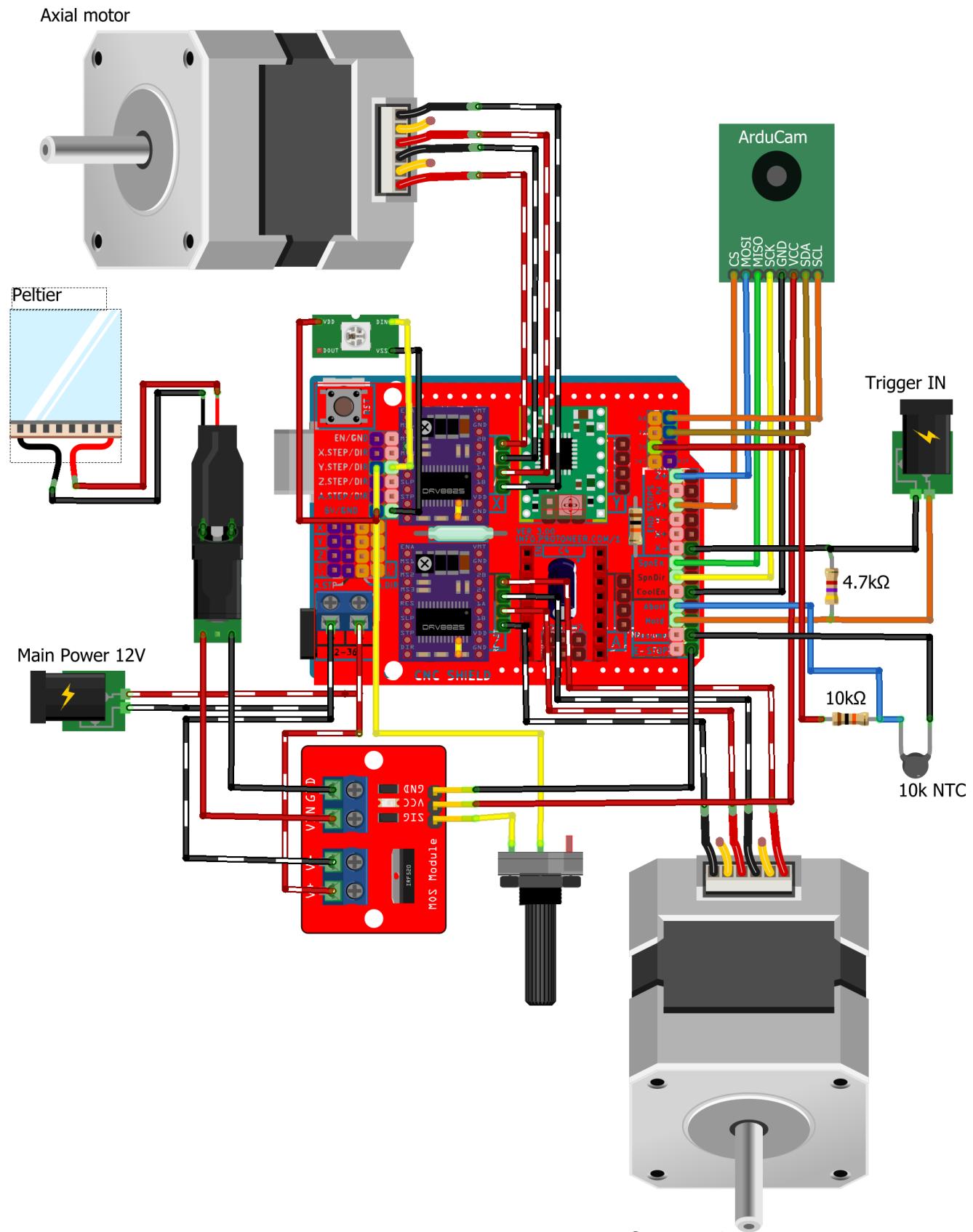
Instructions

Part 1: 3D printing

All 3D-printed parts can be made using PETG or PLA filaments. For the 'chamber', it is recommended to use a transparent material for the first three layers to allow the use of transmitted light from a microscope. The 'chamber' can be printed in approximately 4 hours, requiring 10.5 meters of filament. For the remaining parts, black filament is recommended to minimize light reflections. Printing these parts takes about 25 hours and consumes 75 meters of filament. The individual STL and 3MF files and the complete Fusion360 project are in the [3D folder](#). The complete iCAT model can also be loaded from the [mesh folder](#). The STL data can be visualized using 3D slicer software, while the OBJ file can be opened with any mesh viewer, such as the 3D Viewer application in Windows.

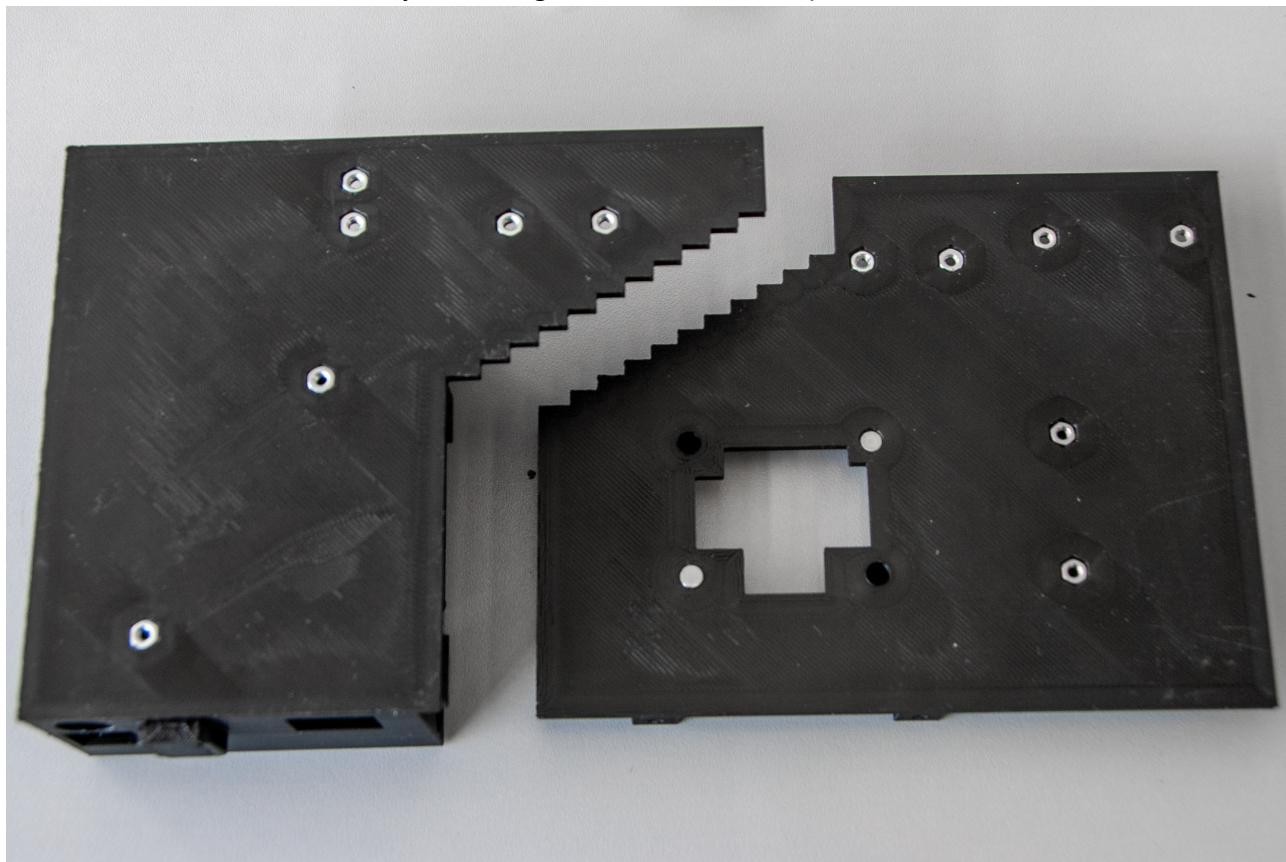
Part 2: iCAT assembly

To assemble iCAT from individual parts, follow the instructions below. If needed, see the [pinout sheet](#), and the [wiring diagram](#).

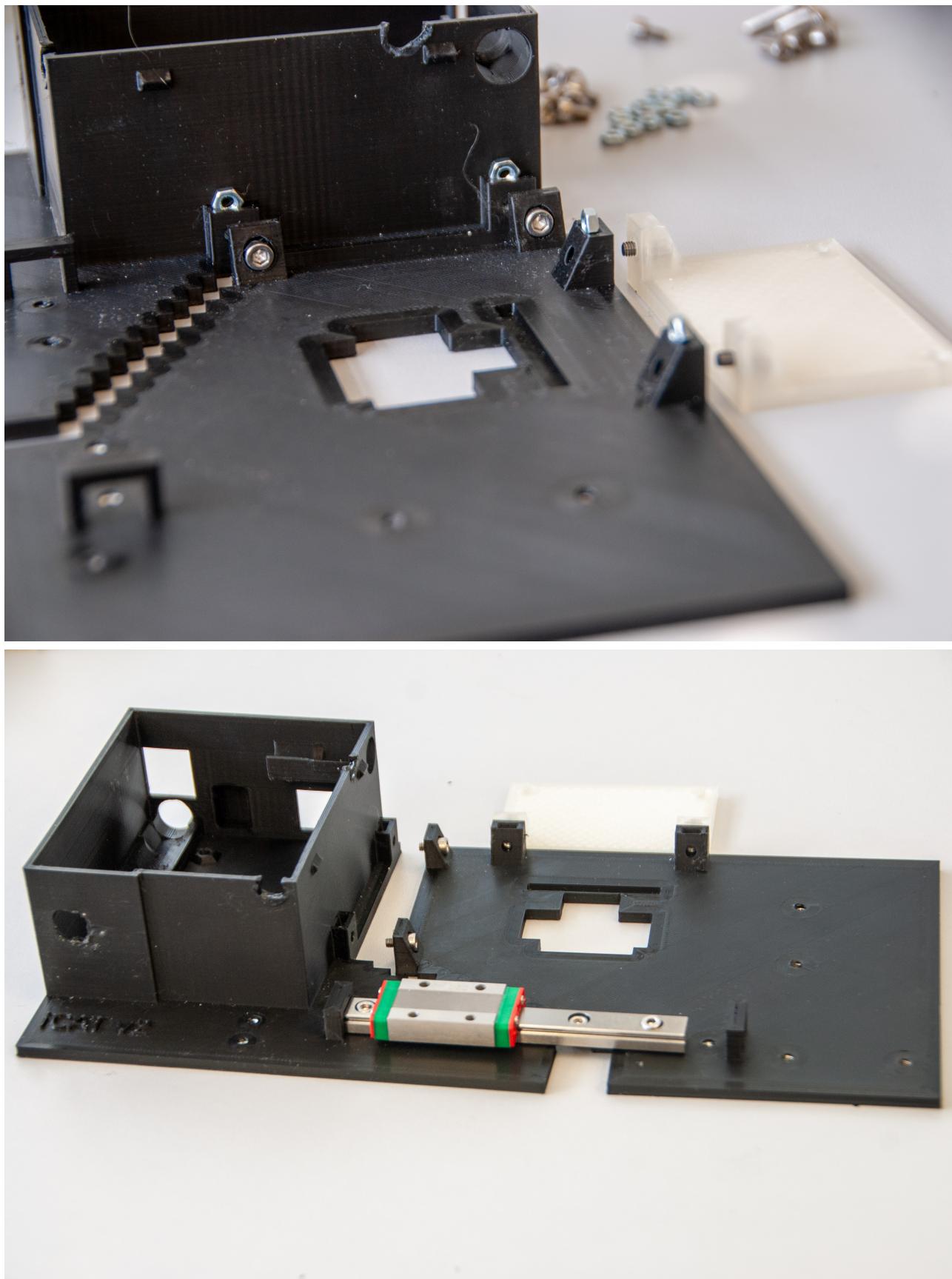


fritzing

1. Insert all M3 nuts and four neodymium magnets into both 'base' parts.

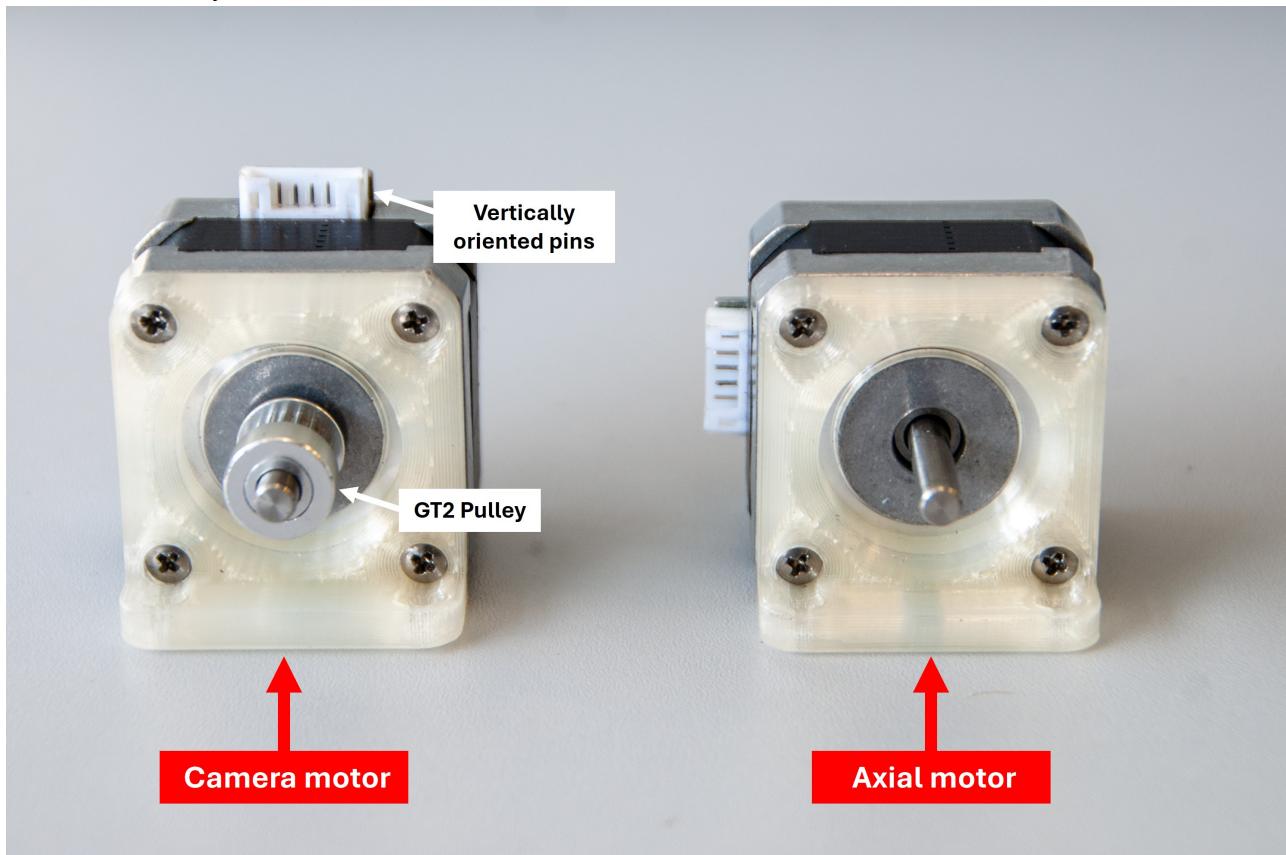


2. Slide the linear guide into the 'base-1', put both 'base' parts together, and secure the connection between them with two M3x6 screws. Attach the 'mounting plate' using two M3x6 screws and secure the 'linear guide' with four M3x5 screws.

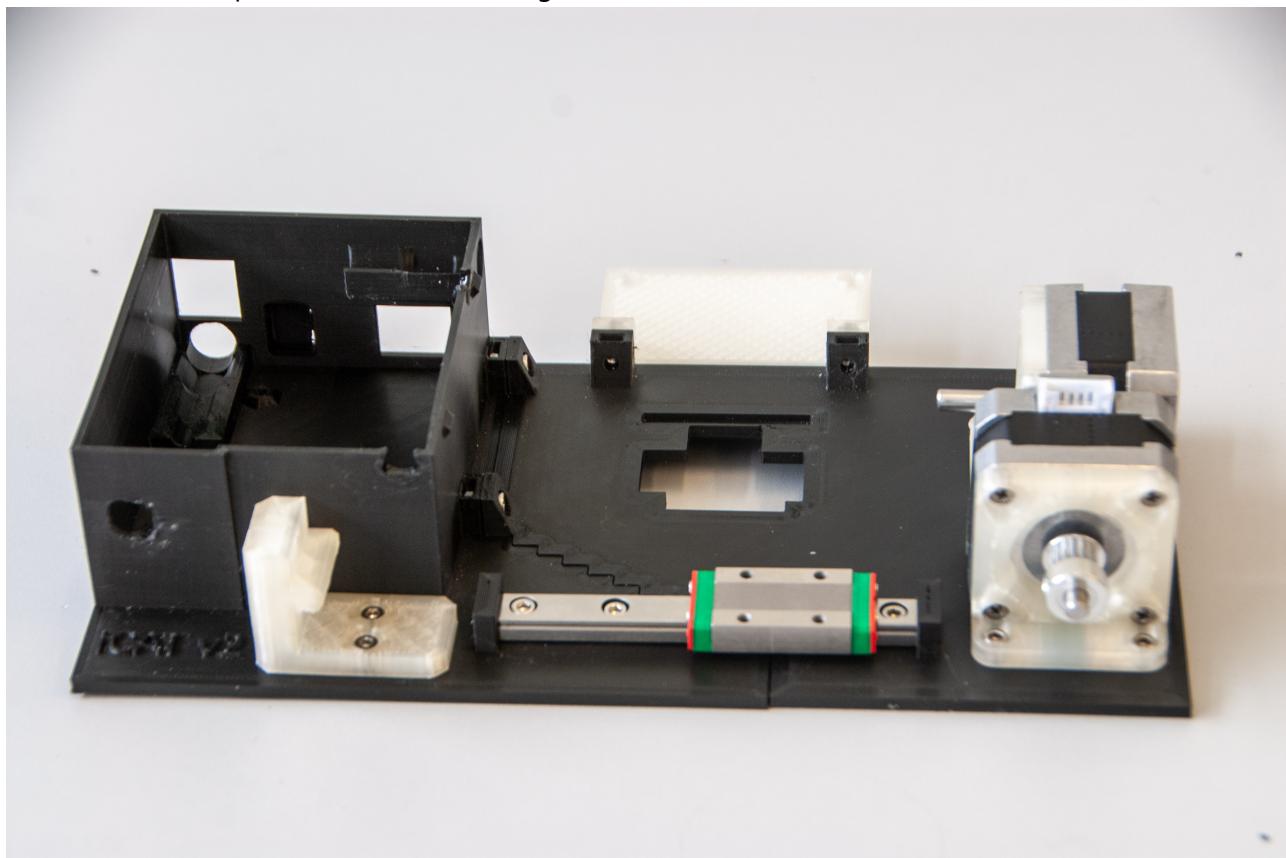


3. Fix the 'motor plates' to 'NEMA motors' using eighth M3x4 screws according to the image below.

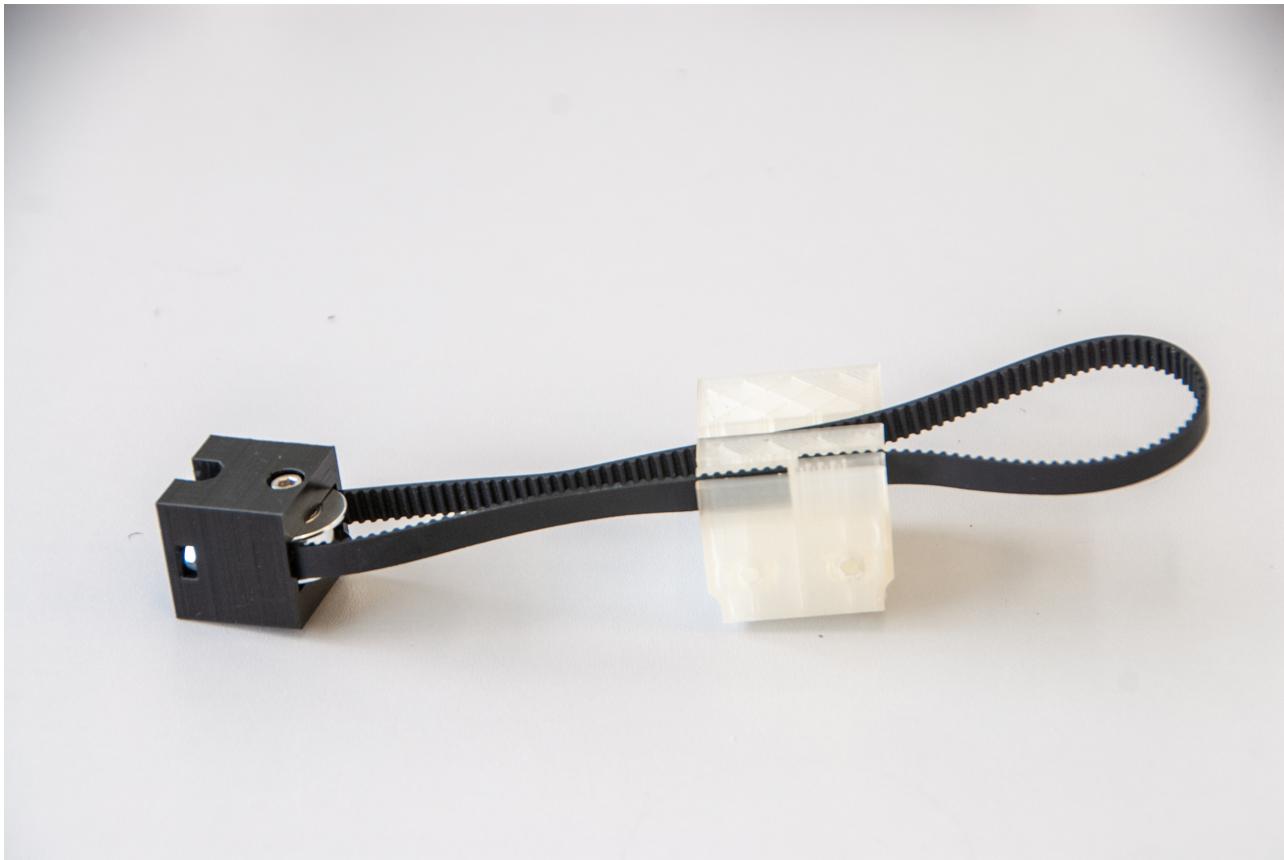
Attach 'GT2 Pulley' to one of the 'NEMA motor'. The pin connections of this 'NEMA motor' need to be oriented vertically.



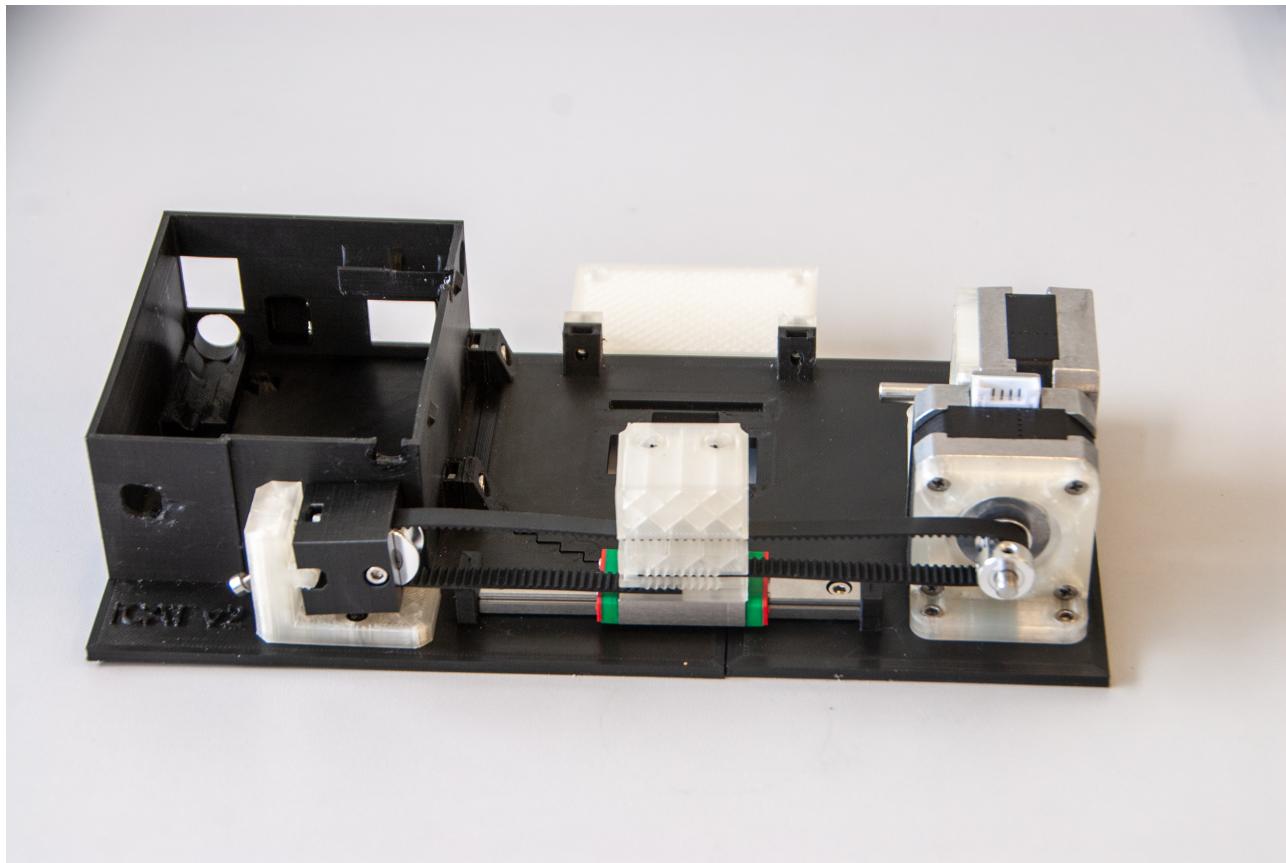
4. Secure the 'motor plates' on the 'base' using M3x4 screws.



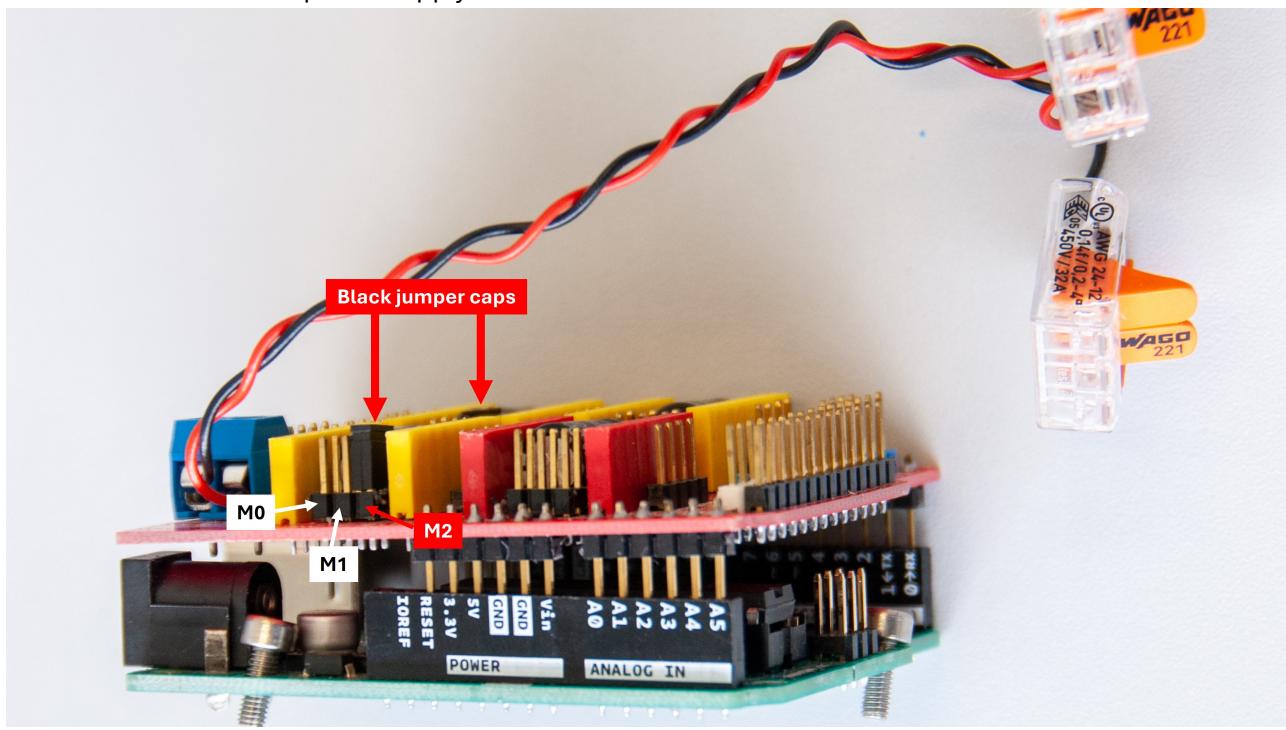
5. Insert three M3 nuts into the 'cargo plate-2' and attach the 'GT2 Idler' to it using a M3x14 screw. Cut 33.5 cm of 'GT2 Timing Belt' and pass it through the 'idler'. Fasten the ends of the belt inside the 'cargo' according to the image below.



6. Secure the 'cargo' to the 'Linear Guide Carriage' using two M3x4 screws and place the belt on the 'GT2 Pulley' that was previously attached to one of the 'NEMA motors'. Attach the 'cargo plate-2' to the 'cargo plate-1' using prepared screws. Do not overtighten these screws, otherwise the 'base' will bend.



7. Insert the M3x6 screws into the mounting holes of the 'Arduino Uno' and plug the 'CNC shield' onto the 'Arduino', ensuring that the pins are properly aligned with the corresponding headers. To adjust micro-stepping (1/16 step), set up the 'M2 pin' of the 'CNC Shield' HIGH by connecting the 'black jumper caps', as shown in the image below. Attach the 'DRV8825 Stepper Motor Drivers' to the 'CNC Shield' and connect the power supply wires to both the shield and the 'WAGO terminals'.



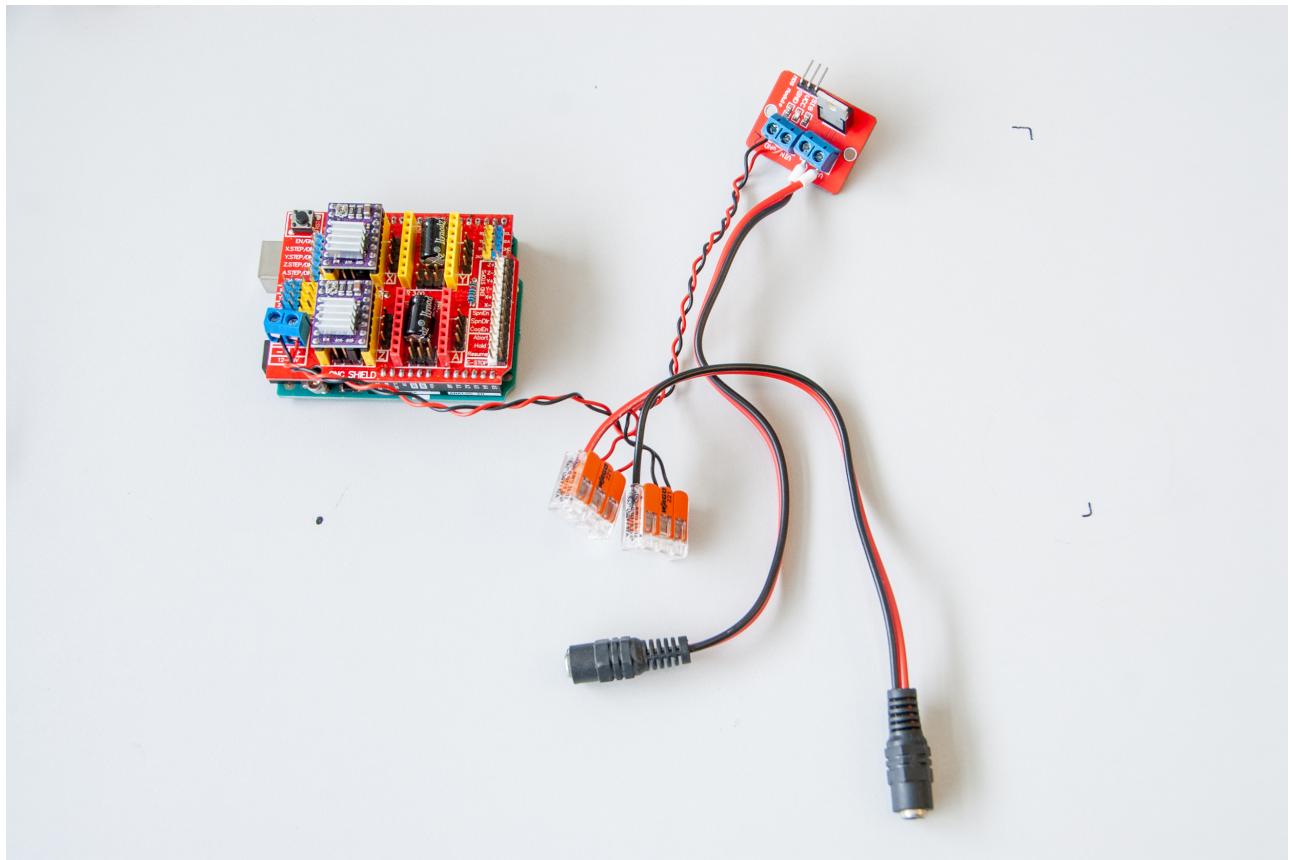
NOTE: Before proceeding any further, the voltage reference (V_{REF}) that corresponds to the maximum current that will flow to the stepper motors needs to be set. The maximum current of used stepper motors is 1.8 A. To calculate the V_{ref} , use this equation:

$$V_{REF} = I_{max}/2$$

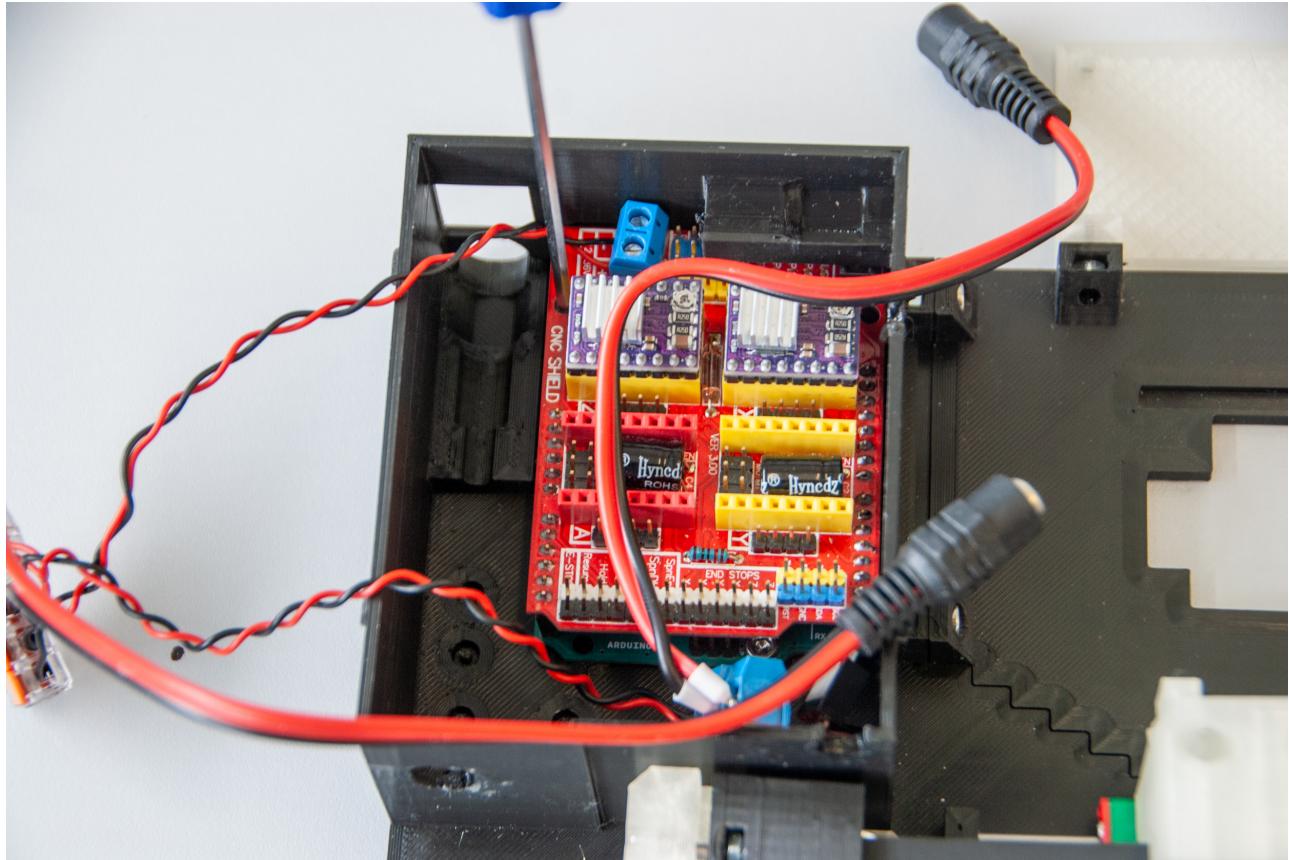
$$V_{REF} = 1.8/2 = 0.9 \text{ V}$$

In order to set up the V_{ref} , plug in the 'Arduino UNO' with 'CNC Shield' and the 'DRV8825 Stepper Motor Drivers' to the USB port, connect the negative probe of the multimeter to the GND, connect the positive probe of the multimeter to the screwdriver tip, set the multimeter to DC Voltage measurement, and use the screwdriver to turn the potentiometer until you get the calculated voltage.

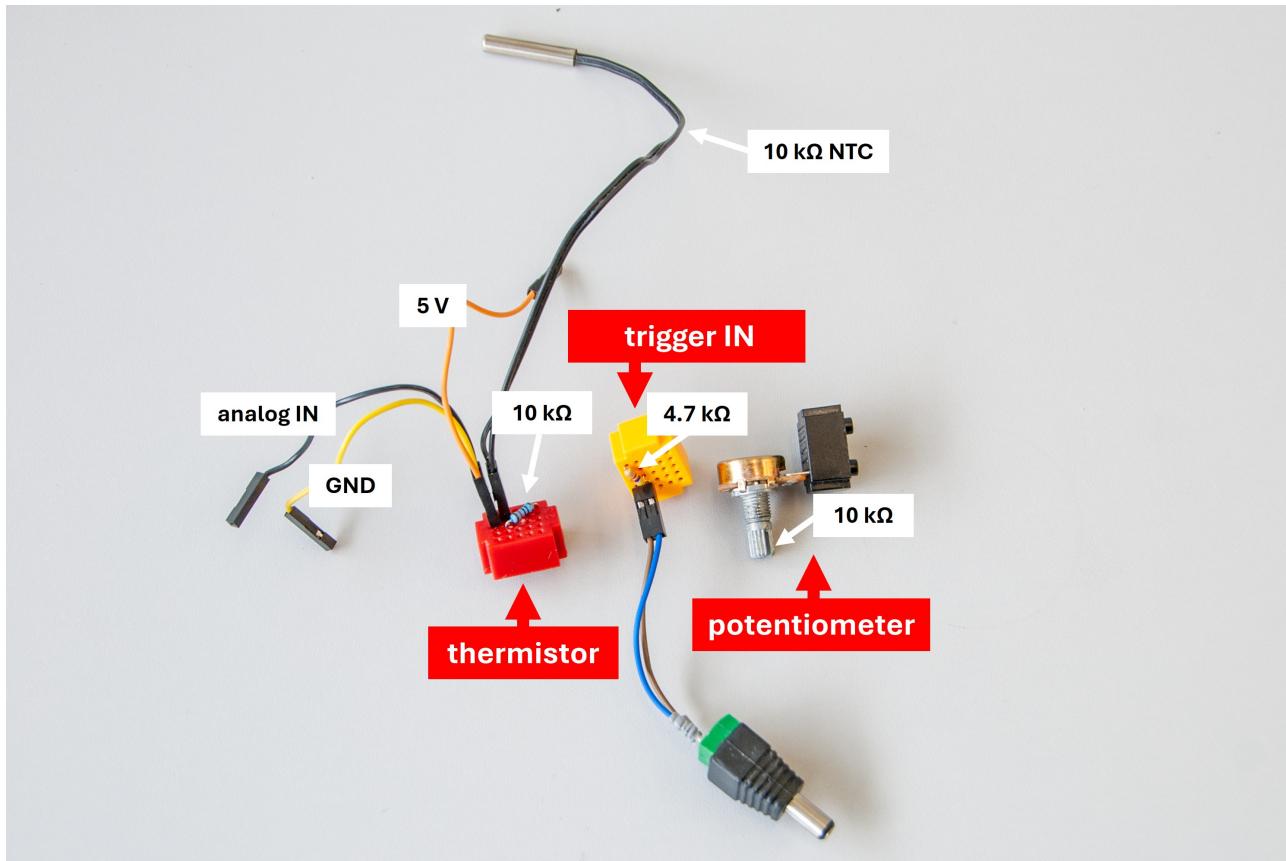
8. Similarly to the previous step, interconnect the 'WAGO terminals' with the 'IRF520 Driver Module' using additional power wires. Next, connect the wire end of the first 'Pigtail Cable Jack' to the 'WAGO terminals'. The wire end of the other 'Pigtail Cable Jack' needs to be connected to the 'IRF520 Driver Module'. If needed, crimp the ends of any wires.



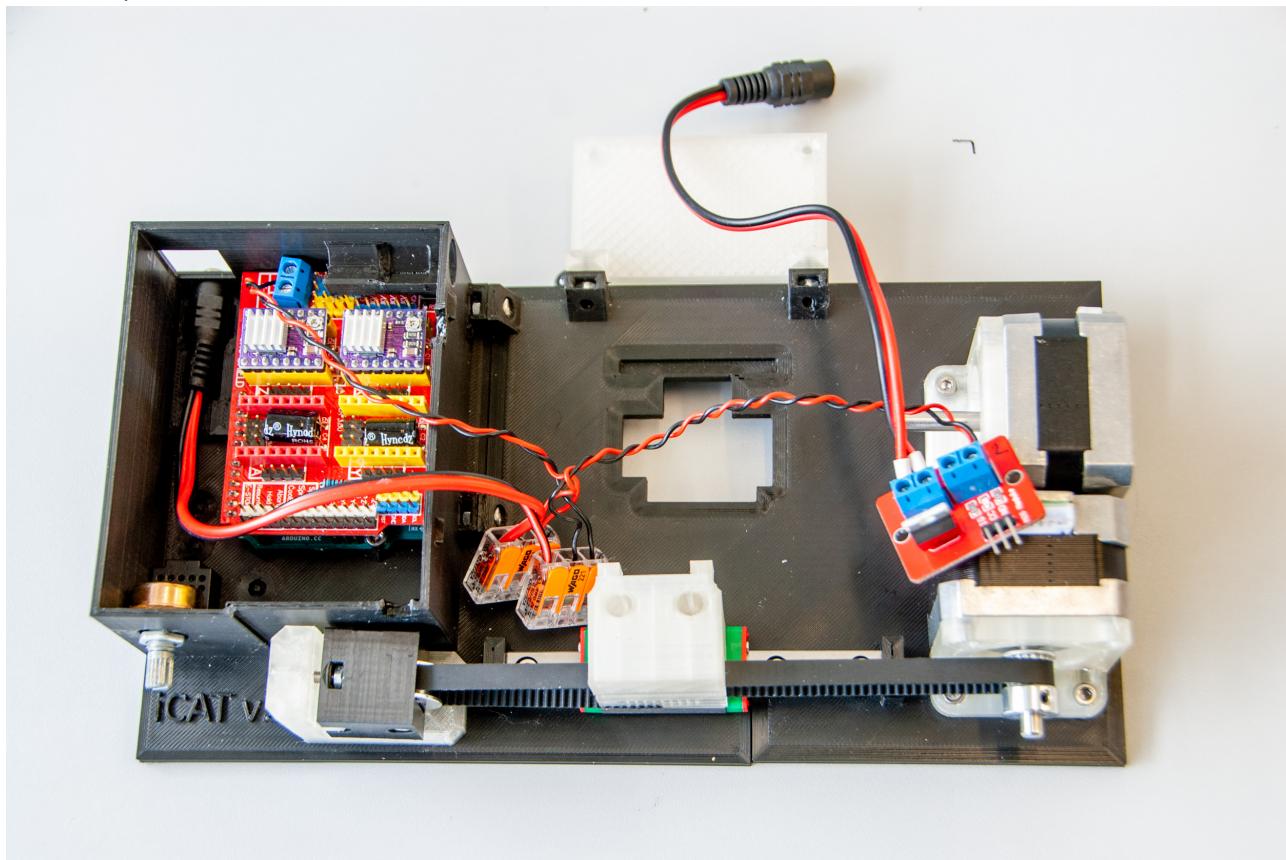
9. Attach the 'Arduinio UNO' coupled with the 'CNC Shield' and all the wirings to the 'base' using previously inserted screws.



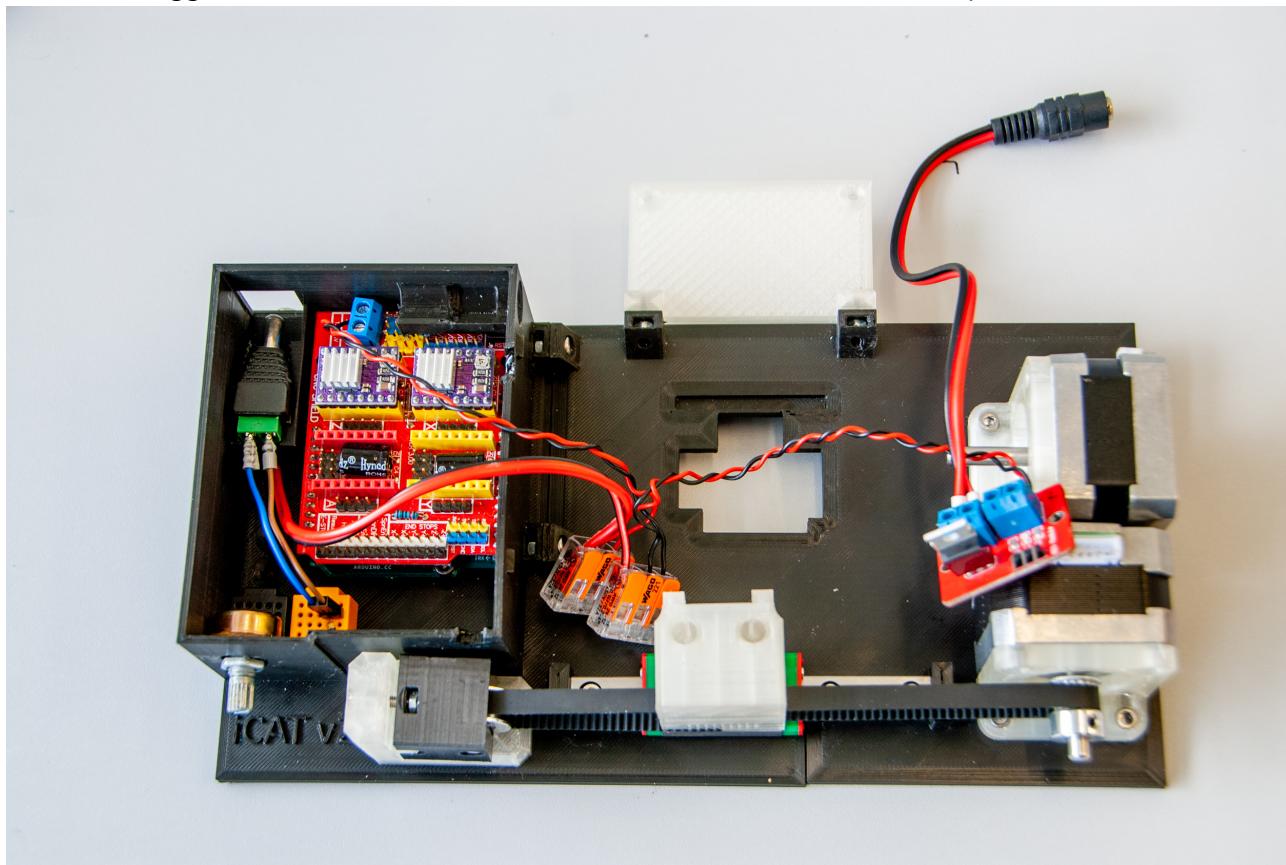
10. Install wiring components on three mini breadboards according to the image below. First, plug '10 kOhm resistor' into the 'Thermistor breadboard'. Shorten the resistor leads, if necessary. Plug '10 kOhm thermistor', while one of its leads connects it with the resistor. Connect the 'Male-Female Dupont wire' to it. This will be connected to the analog input ('Abort pin') on the 'CNC shield' later on. The other 'Dupont wires' need to be connected to the other thermistor lead (ground) and to the second resistor lead (5 V). Second, prepare the 'Trigger IN' breadboard by plugging a '4.7 kOhm resistor' into it and connecting its terminals to the 'Male DC connector'. 'Dupont wires' will be connected to both of these resistor leads later on (step 14) to the ground and to the analog input ('Hold' pin) on the 'CNC Shield'. Third, plug '10 kOhm potentiometer' into the last breadboard. Two 'Dupont wires' will be connected to the side and the middle terminals later on, connecting it to 'PWM digital input' ('Step Y' pin of the 'CNC shield') and to the 'IRF520 Driver Module' signal input. If needed, crimp the ends of any wires.



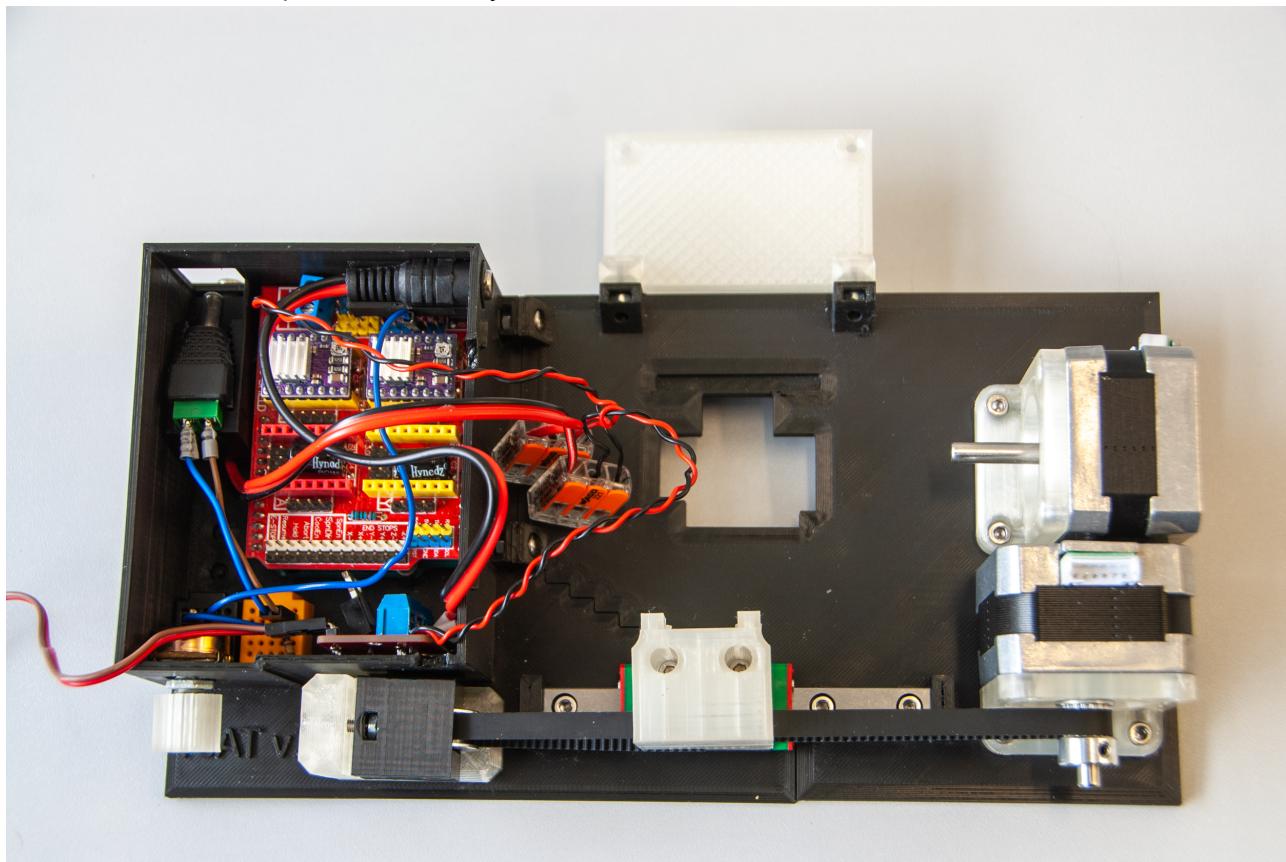
11. Install the 'Potentiometer breadboard' and 'DC Female pigtail' in to the 'base' and cover it with the 'base adapter'.



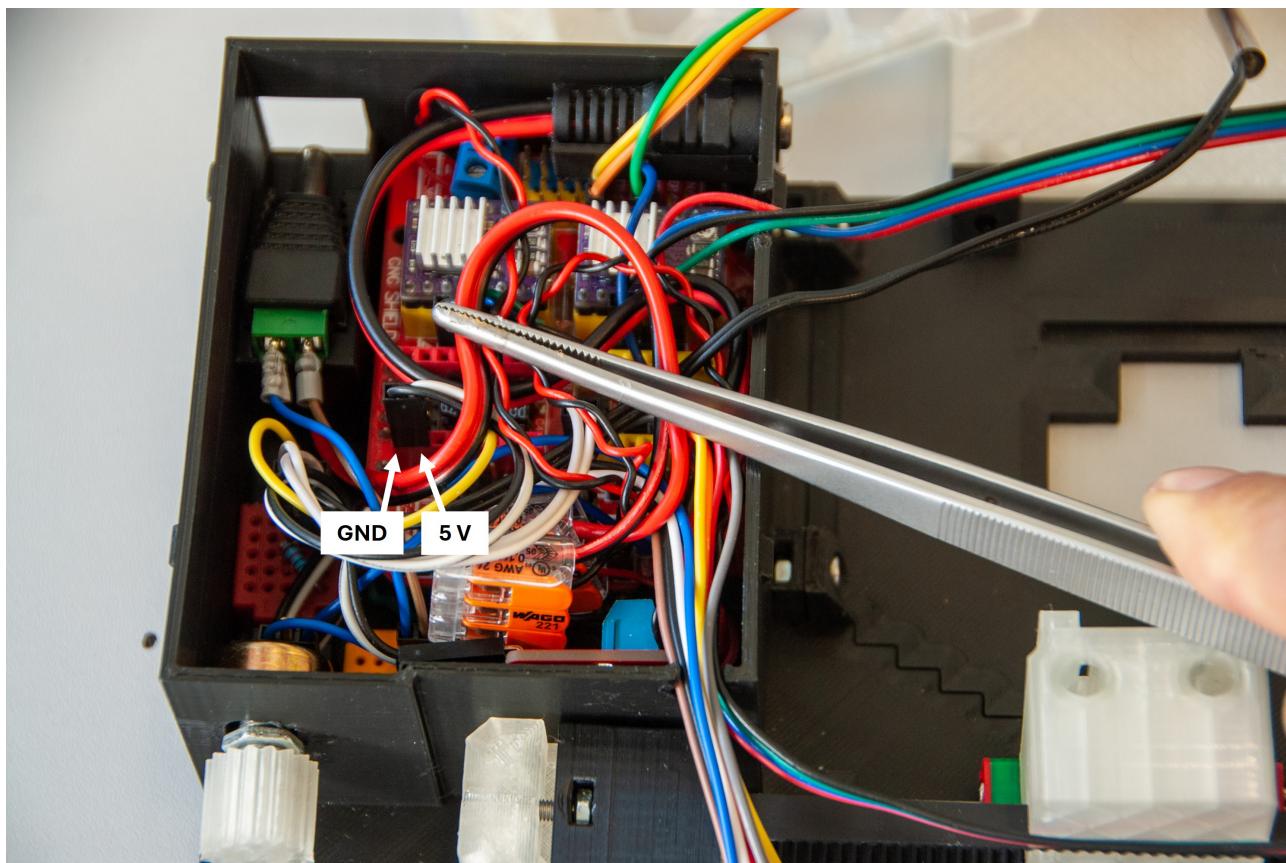
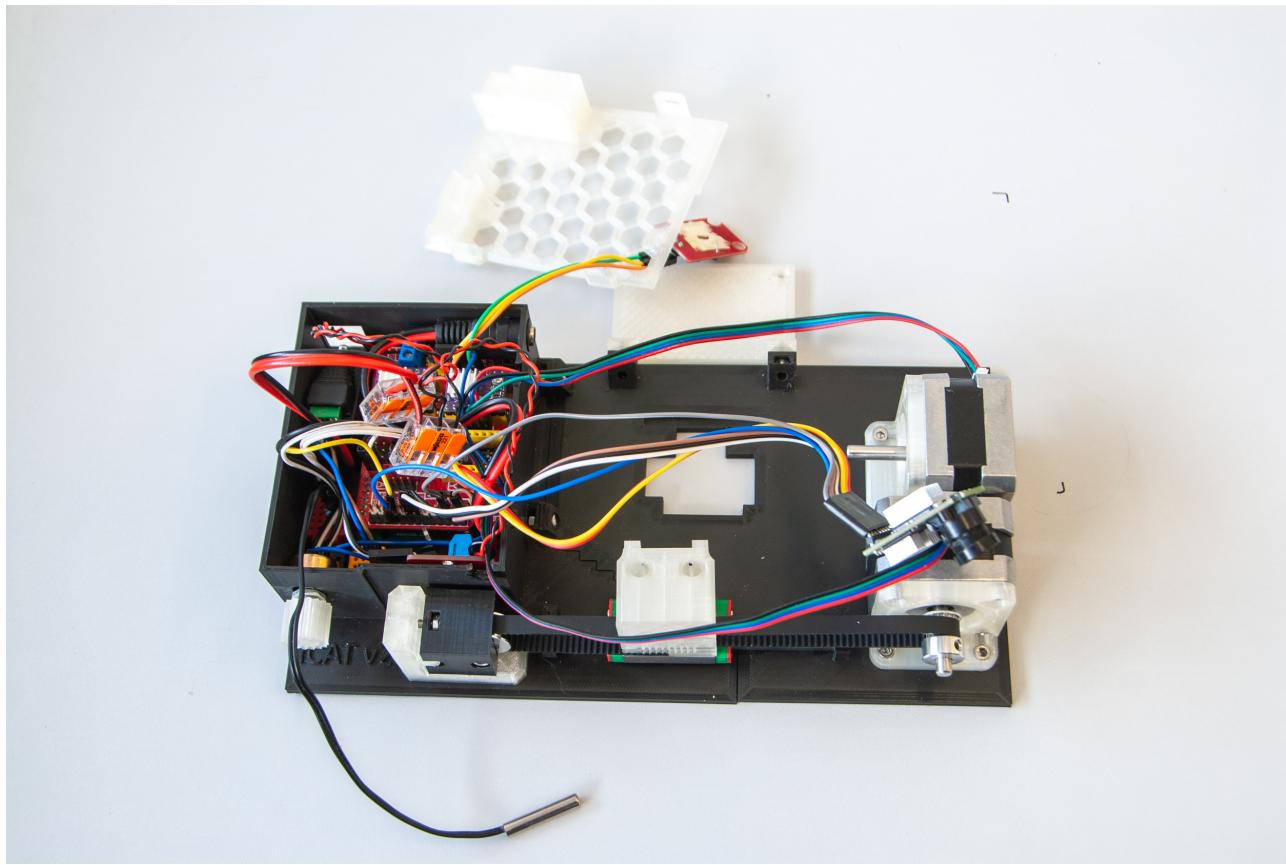
12. Install the 'Trigger IN breadboard' and fit 'Male DC connector' in the 'base adapter'.



13. Install the 'IRF520 Driver Module' and plug its 'Dupont wires' (ground, 5 V). Plug the signal input as described in step 10. Fit 'Male DC connector' connected to 'IRF520 Module' in the upper right corner of the 'base'. Secure the 'potentiometer' by its nut and attach the 'knob' to it.



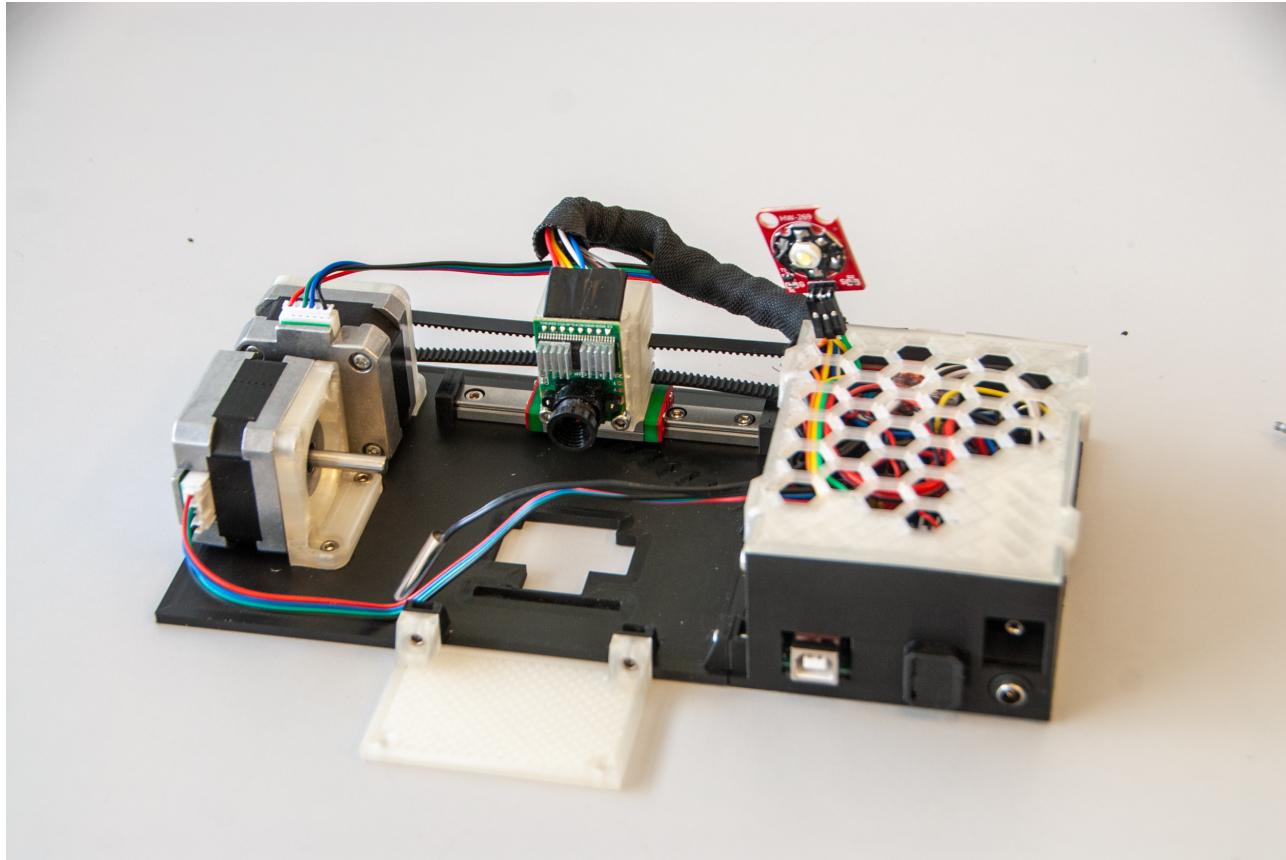
14. Install the 'Thermistor' breadboard in the 'base'. Connect 'Dupont wires' of all the 'breadboards' as described in step 10. Connect 'LED' to 5 V, ground, and digital output ('Dir Y' pin) on the 'CNC Shield'. Plug eight 'Dupont wires' to the 'Arduacam' and connect them to the 'CNC Shield' according to the [pinout sheet](#) and the wiring diagram. Test the functionality of the device at this point before attaching the 'lid'.



NOTE: It is possible to use 5 V and GND pins of unoccupied driver bays on the 'CNC Shield' to connect the

'IRF520 Driver Module' and the 'thermistor'.

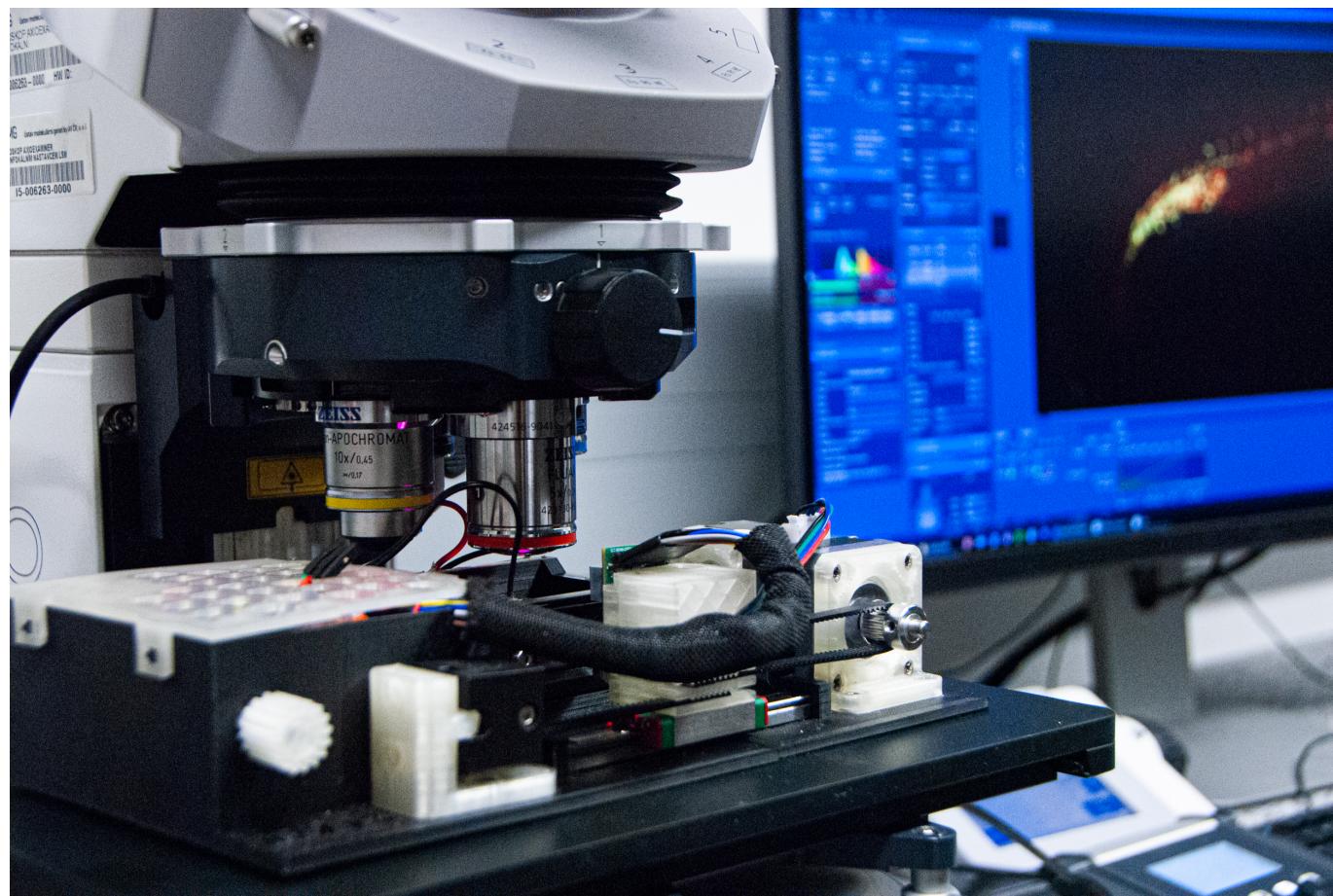
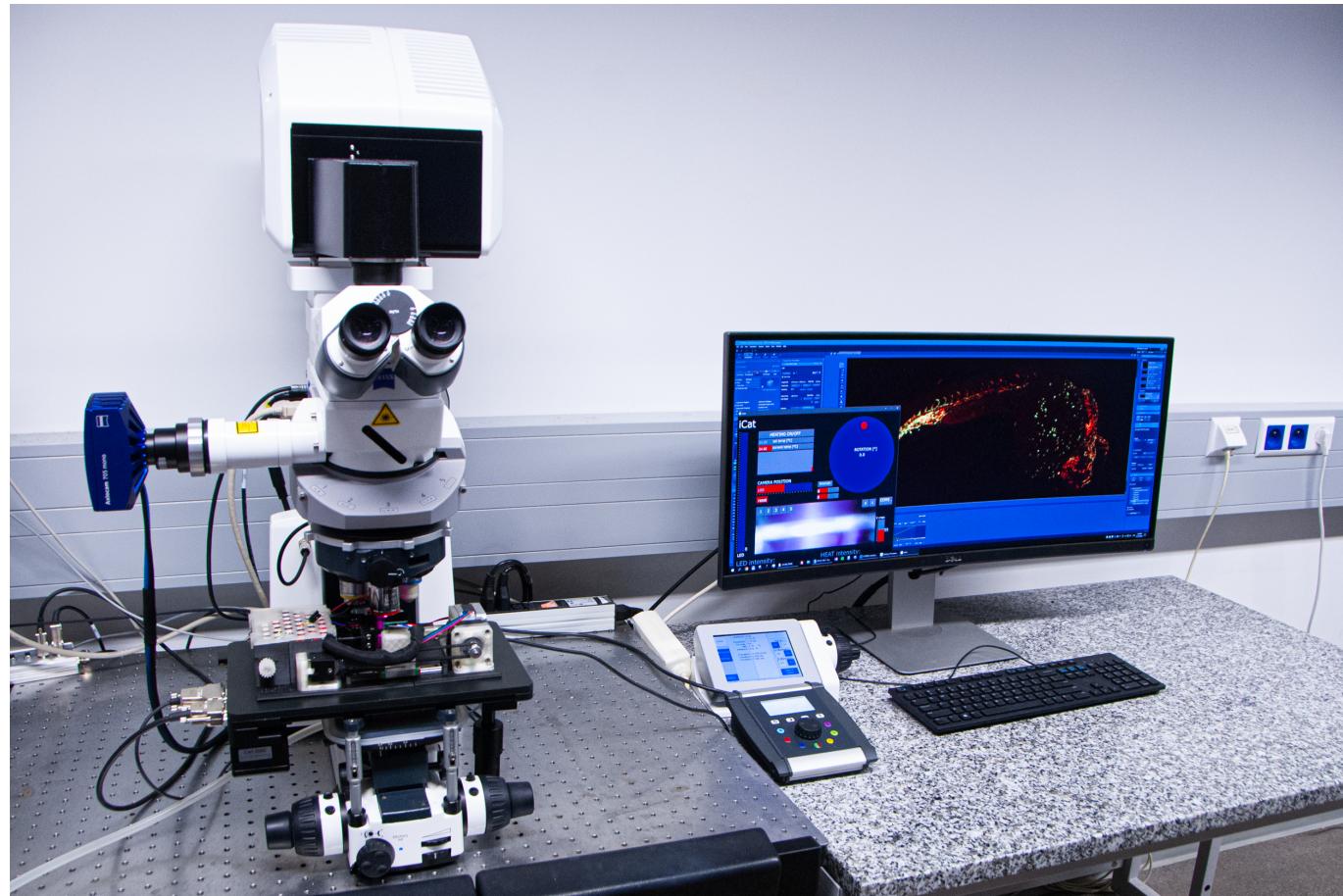
15. Attach the 'lid' to the 'base' and attach 'Arducam' to the 'cargo' using two M3x4 screws. Install a Sleeving Cord Protector to cover the wiring of the camera.

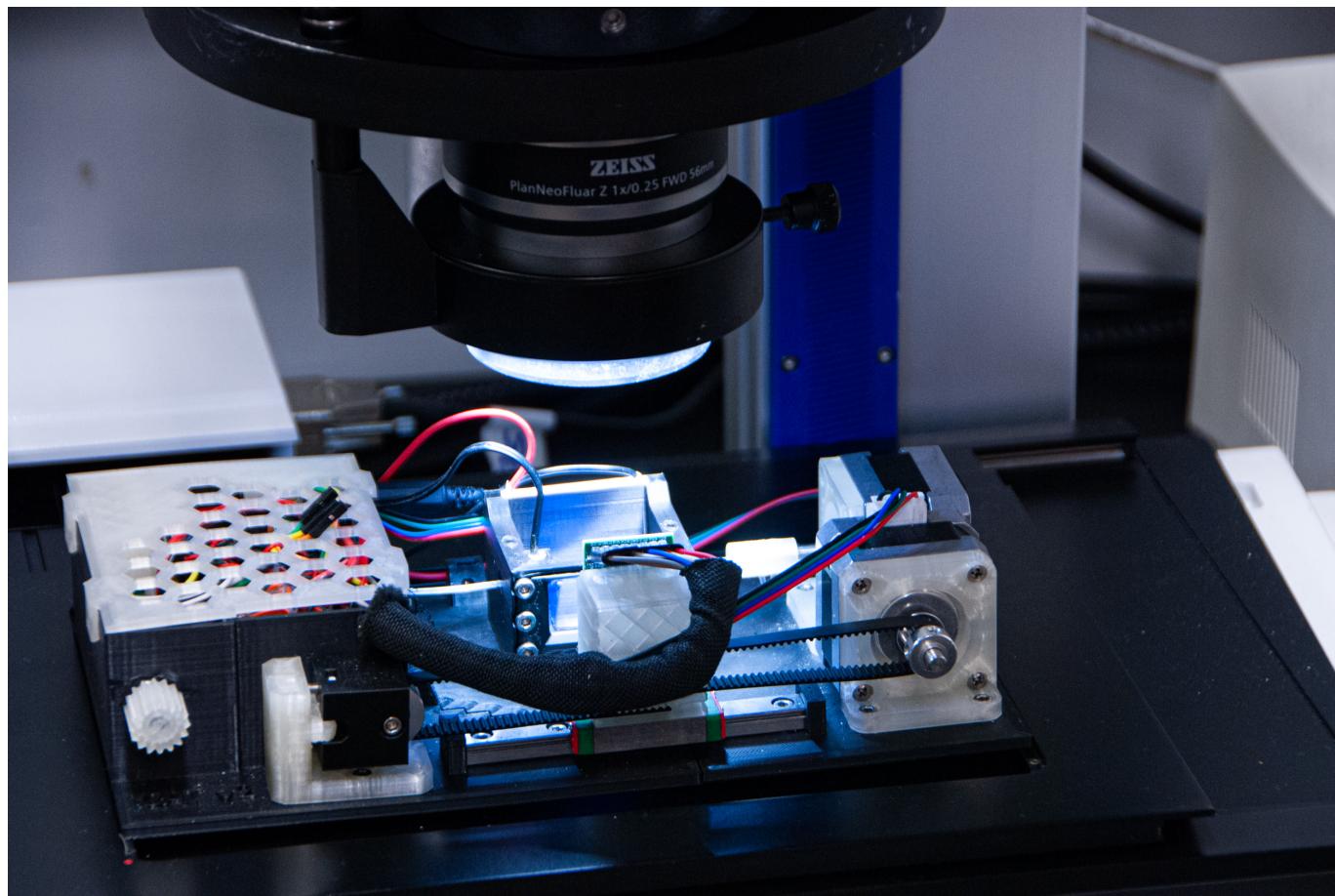


16. Use 'mounting screws' to connect the iCAT's 'mounting plate' to the stage of the microscope.

So far, the iCAT was successfully mounted and used in combination with the following microscopes.

ZEISS Axio Examiner:

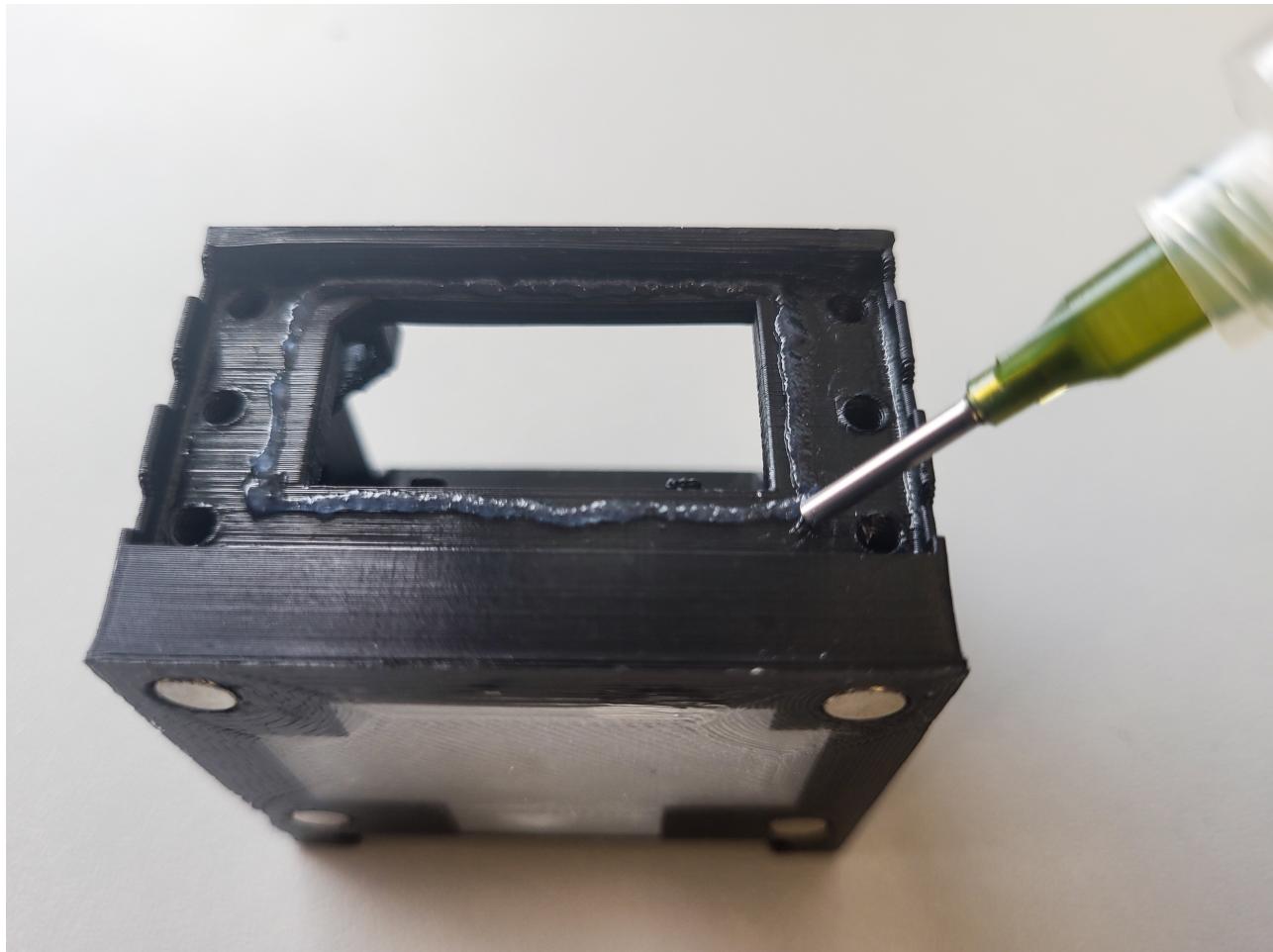


ZEISS Axio Zoom.V16:

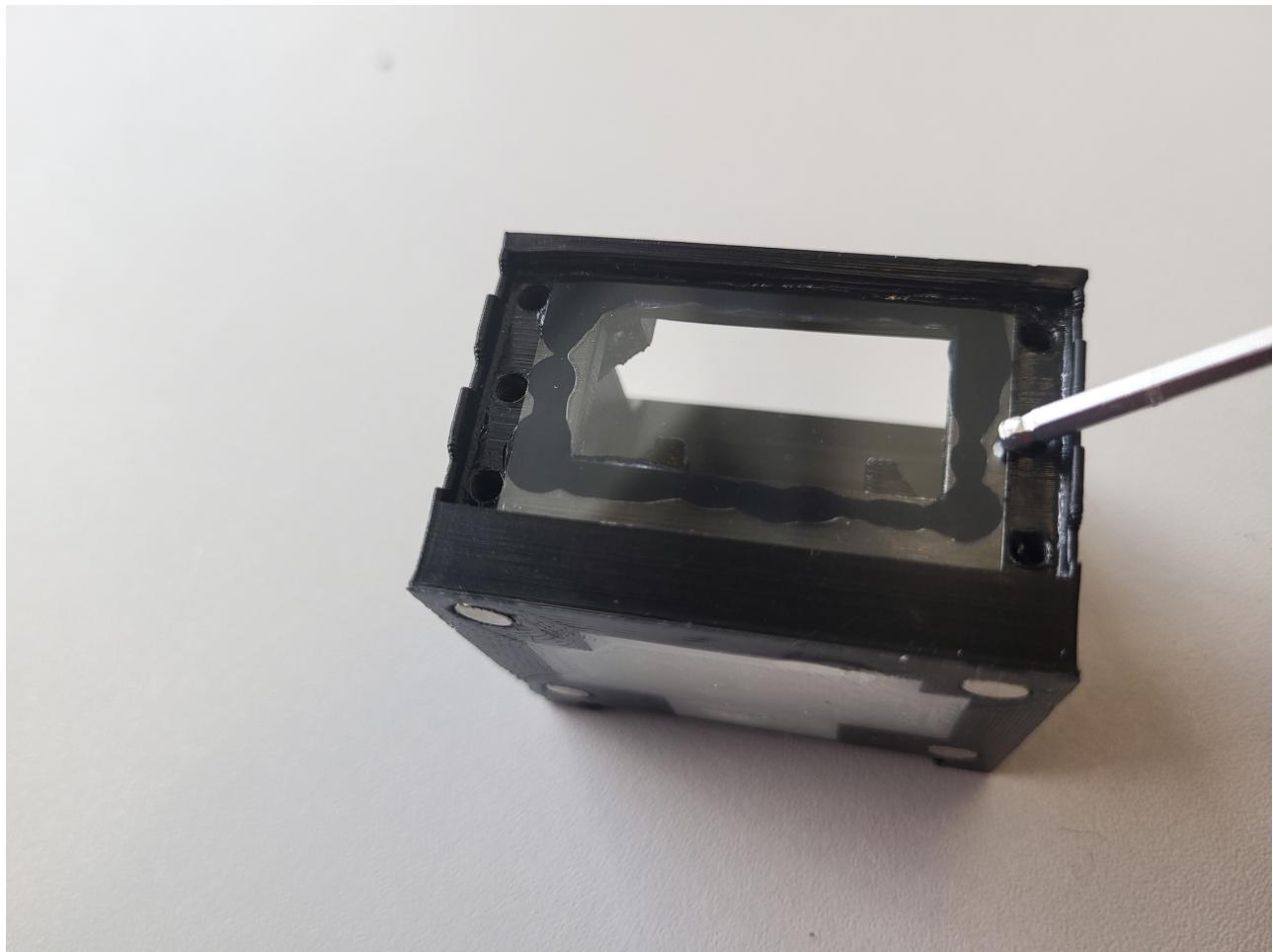
Part 3: Chamber assembly

To install the 'cover' on the sample 'chamber', follow these steps:

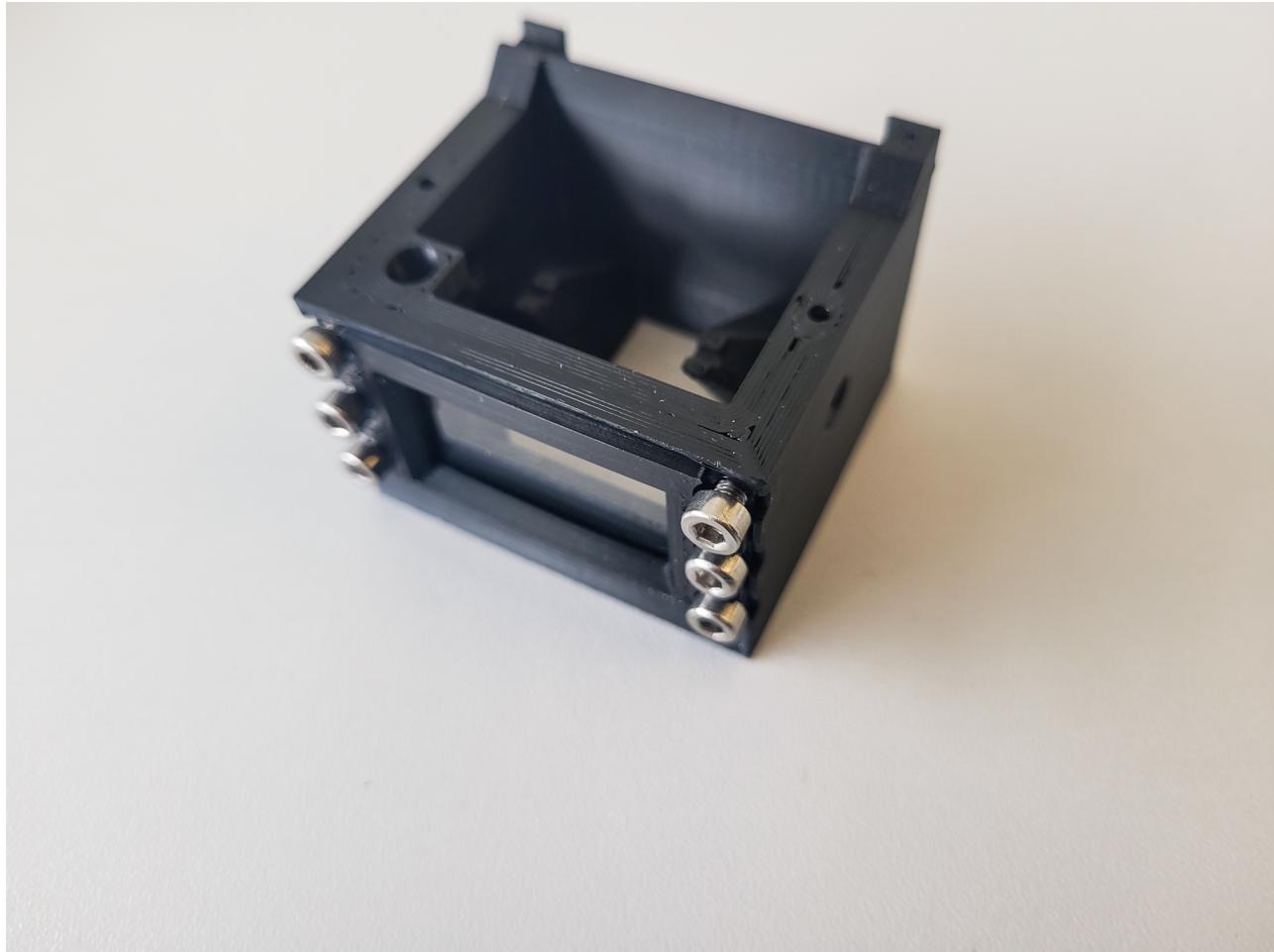
1. Apply grease around the perimeter of the 'chamber' window.



2. Gently place the 'Microscope Cover Glass' (coverslip) onto the applied grease. Ensure that the grease spreads evenly under the coverslip and that it is centered.

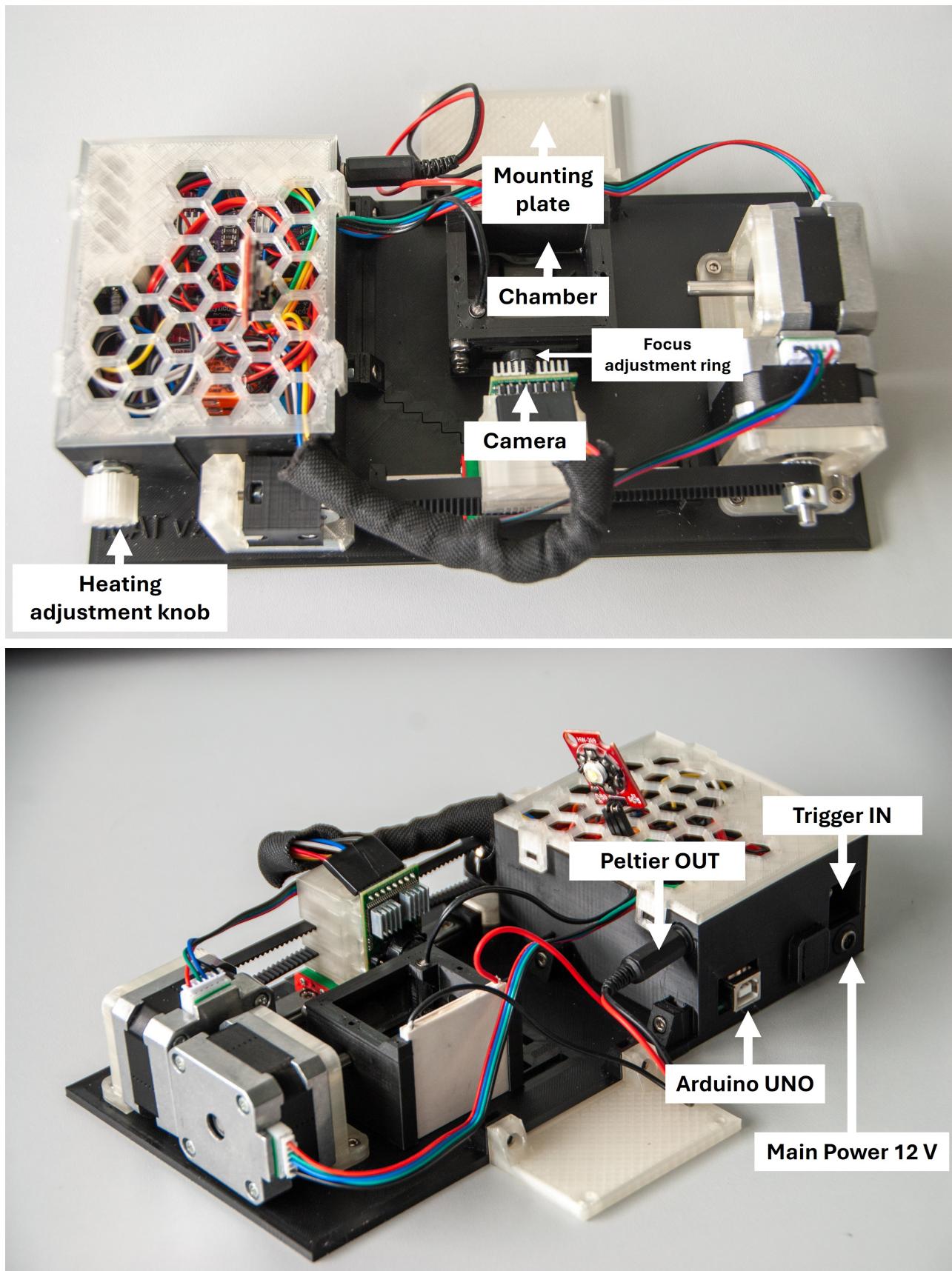


3. Secure the attached coverslip with eight M3x4 screws, and tighten the screws gently.



NOTE: Be careful when tightening the screws, as the coverslip is fragile. It is recommended to tighten the screws gradually in several rounds, testing the 'chamber' after each round by filling it with water. If no leaks are observed, let the 'chamber' sit overnight on a soft tissue surface. No leaks should be present the following day.

4. Slide the Peltier element to the back of the 'chamber'. Place the 'chamber' on the 'base', and connect it to the female DC connector on the side. Focus the camera using the adjustment ring. All iCAT connectors are depicted below.



Part 4: Installation

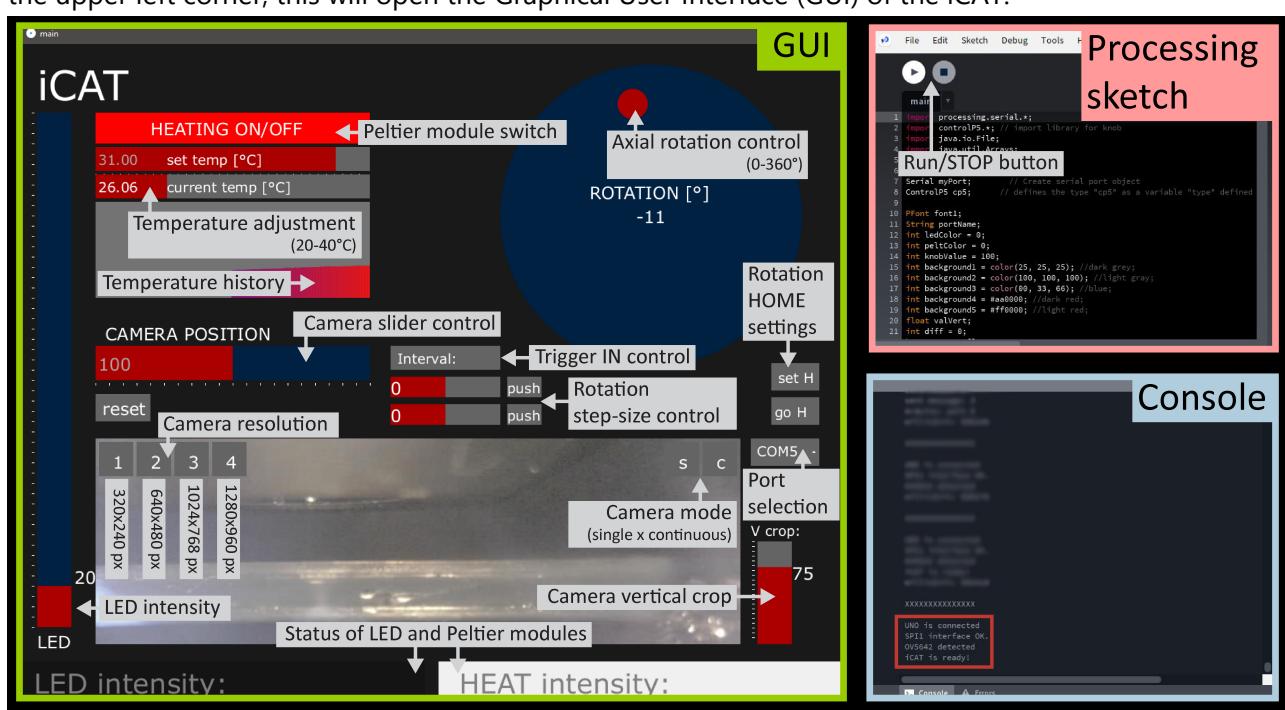
1. Download the entire repository, or simply download the Arduino sketch folder, the `main.pde` Processing sketch, and install the [Processing application](#).

NOTE: If you run the `iCAT.exe` file directly, you will need to install OpenJDK17, as the [Processing sketch folder](#) does not contain Java.

2. Install the [Arduino IDE](#).
3. Open the Arduino IDE by opening the `main.ino` file. Next, go to the Library Manager and install the [AccelStepper](<https://github.com/osvobo/iCAT/tree/dev/>) (<https://www.airspayce.com/mikem/arduino/AccelStepper/>) and [ArduCAM] (<https://github.com/osvobo/iCAT/tree/dev/> <https://github.com/dennis-ard/ArduCAM>) libraries. Once both libraries are installed, press the 'Verify' button to ensure the installation is successful.
4. Check the path of `memoriesaver.h` file by hovering the mouse over its name at row 5 in the line `#include "memoriesaver.h"`. Then navigate to its folder and replace it with the `memoriesaver.h` file provided, which has the camera definition uncommented:

```
#define OV5642_MINI_5MP_PLUS
```

5. Test the iCAT by connecting the Arduino to the PC using USB-B cable, connecting the '12 V main power', the 'trigger IN', and the 'Peltier connector'. Open the `main.pde` and click on the **Run button** in the upper left corner, this will open the Graphical User Interface (GUI) of the iCAT.



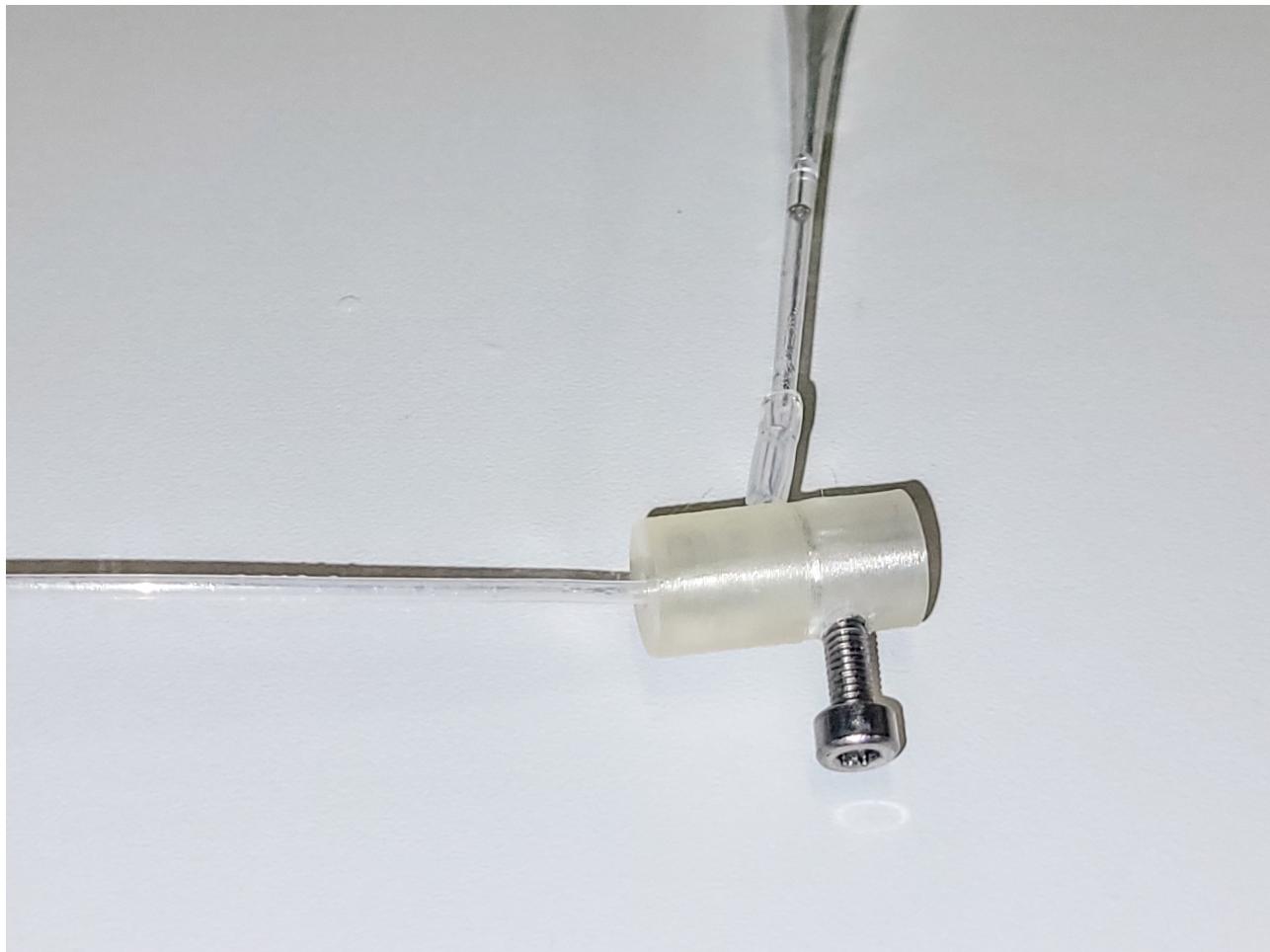
6. Select the port using **Port selection button** to which the iCAT is connected. After successful connection, the message will appear in the console: "*iCAT is ready!*".
7. Test the functionality of the main components: the LED, heating, motors, and camera. Adjust the camera focus using the [Focus adjustment ring](#).

Part 5: General usage

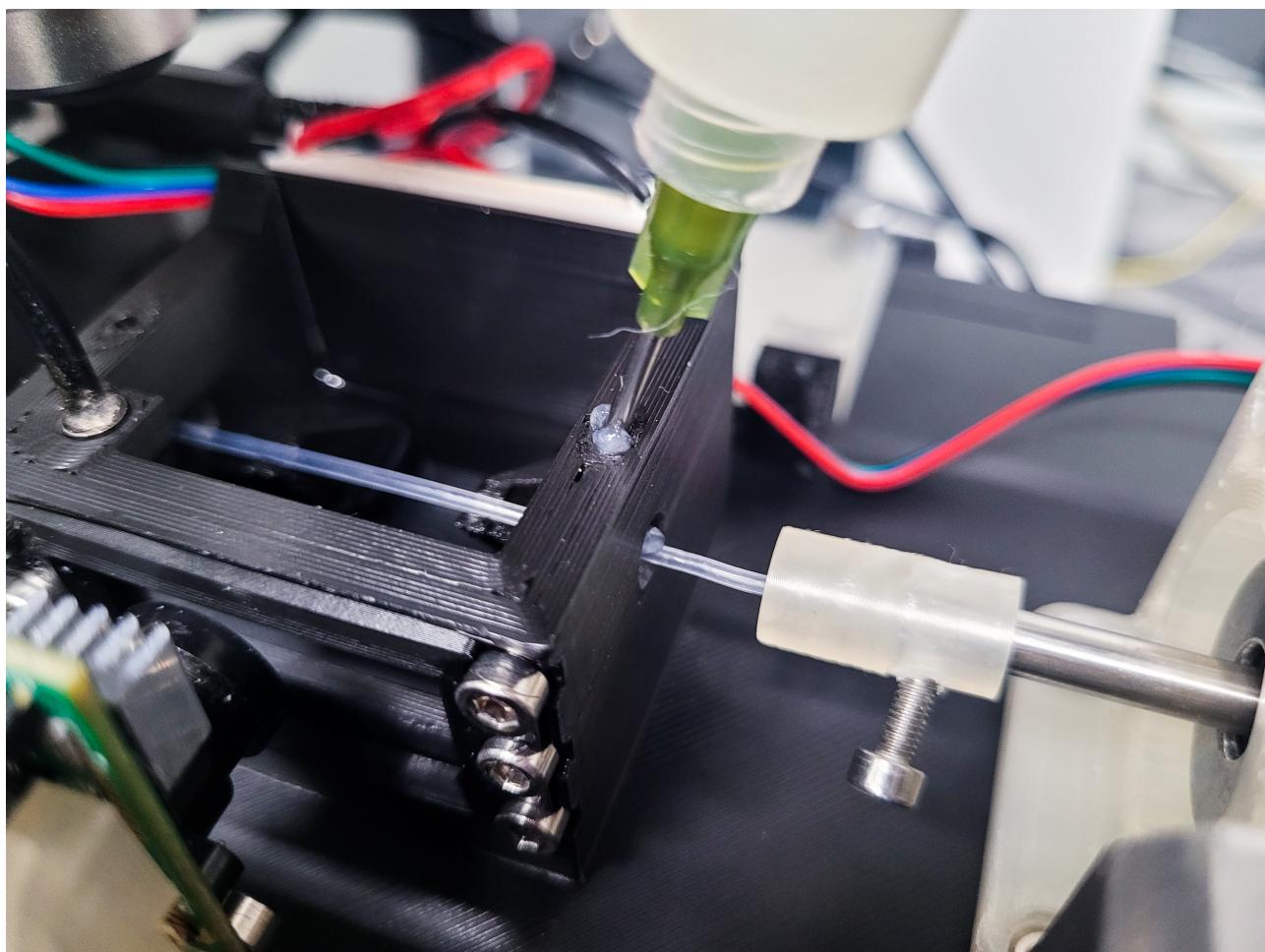
These instructions can be used to mount and image zebrafish embryos between 0 – 4 dpf using iCAT and ZEISS Axio Examiner microscope equipped with LSM900 confocal scan head. Other specimens can be imaged as well. If needed, simply use larger FEP tubes to image larger samples. In such a case, however, the 'chamber' needs to be modified. Using the iCAT in combination with other upright microscopes should be possible however this has not been tested.

1. Cut 9 cm of FEP tube.
2. Insert the FEP tube inside the 'FEP adapter'. Attach 1 cm of the silicone rubber tube to the end of the FEP tube.
3. Cover dechorionated embryos with 0.8 % low melting imaging grade agarose with 0.5x Tricaine mesylate on a 3 cm cell culture dish.
4. Attach a borosilicate glass Pasteur pipette to the pipette pump and aspirate one zebrafish embryo into the pipette.
5. Insert the tip of the pipette into the silicone rubber tube and gently transfer the embryo to the center of the FEP tube. Allow the agarose in the FEP tube to solidify.





6. Gently insert the FEP tube into the 'chamber'. Using a syringe, apply grease inside the openings at the top of the 'chamber'. The grease will fill the gaps and seal the FEP tube in place.



7. Tight the 'FEP adapter' screw to fix it to the axial motor. Clean the grease from the surface of the FEP tube using a soft, lint-free tissue. Rotate the tube as needed to ensure thorough cleaning. Fill the 'chamber' with water. Launch iCAT and set up the desired 'chamber' temperature using the [GUI](#).
NOTE: Depending on the surrounding temperature and the specific model of the Peltier element used, the setup temperature might fluctuate slightly. This can result in inconsistent imaging during long time-lapse experiments due to the expansion and contraction of the 'chamber'. To mitigate this, adjust the heating output using the adjustment knob. Turning the knob counterclockwise decreases its power, while turning it clockwise increases it. Further fine-tuning can be done in the [main.pde](#) Processing sketch by increasing or decreasing [peltPower](#) parameter. Similar to the adjustment knob, decreasing this value reduces the heating power, while increasing it raises the Peltier output.

```
if(valueDiff <0 ) {  
    peltPower = 5;  
    output.println();  
    output.printf("%02d:%02d:%02d    ", hour, min, sec);  
    println("heating OFF");  
    peltColor = color(background4);  
    write(peltPower, 0, 3);  
}
```

8. Use intuitive axial rotation wheel to orient your sample along its x-axis.

Part 6: Special functions

This section provides a detailed explanation of the functions of individual **GUI** buttons and other features of the software.

Reset camera position:

Use the **Camera slider control** to center the camera preview. If the slider reaches its limits, use the **reset button** to recenter it based on the current view.

Camera control:

Use the **Camera resolution buttons** to adjust the resolution. To take a single snapshot, press the **s button**. To activate or deactivate continuous mode, press the **c button**. By default, the camera screen displays the middle portion of the captured image. To scroll up or down, use the Camera vertical crop slider **Camera vertical crop slider**.

Logging and data storage:

The iCAT generates following log files within its root folder:

```
iCAT\logs\log\log_YYYY-MM-DD_HH-mm-ss.txt  
iCAT\logs\rotate\rotate_YYYY-MM-DD_HH-mm-ss.txt
```

The **log_*.txt** file stores all serial communication between the Arduino and the PC to which Processing is connected, while the **rotate_*.txt** log specifically records the rotary movements.

The images captured by the camera are stored here:

```
iCAT\logs\log\capture_no_YYYY-MM-DD_HH-mm-ss.txt
```

Remote axial control:

The iCAT controlling software enables remote axial rotation control, which can be achieved in two ways:

- File-based control:** The iCAT Processing software continuously monitors the **iCAT\logs\temp** folder for the presence of a **temp.txt file**. If such a file is found and contains an integer value, the control software adds this value to the current axial position and uses the result as the target angle for axial rotation. The **temp.txt** file is subsequently renamed to prevent repeated use.
- Trigger-based control:** Remote rotation can also be activated using trigger input (IN) signals. This requires the trigger IN cable to be connected to the **Trigger IN port** and a valid signal to be sent. The signal is a 5 ms, 5 V pulse typically generated by the microscope trigger board via the microscope control software.

To activate the triggering, the **Interval button** must be active, and rotation angles must be configured within the **Rotation step-size control**. The **upper slider** determines the rotation amount.

If two different rotation angles are needed for independent, consecutive trigger inputs, the second

desired angle can be set using the **bottom slider**. This feature is particularly useful for alternating the specimen's rotation back and forth. To achieve this, configure a value in the **upper slider** and set its negative equivalent in the **bottom slider**.

Step-size control:

Additionally, the rotation step-size can also be controlled manually. As with the trigger inputs, set the desired angle(s) using the slider(s). To apply the angle change, press the corresponding **push button**.

Home point:

By default, iCAT always recovers the last known axial angle position based on the stored value within this log:

```
iCAT\logs\rotate\rotate_YYYY-MM-DD_HH-mm-ss.txt
```

It is also possible to save the **HOME position** by adjusting the axial angle and pressing the **set H button**. Once this button is active, toggling the **go H button** will move to the saved angle position. If the **set H button** is inactive, pressing the **go H button** will move to the default position — 0°.

iCAT-ZEN (ZEISS) synchronization:

iCAT can be fully synchronized with the software that controls the microscope. In our showcase example, we developed integration with the **ZEN software**, which is used to control **ZEISS microscopes**. This integration utilizes custom-made **macros** that need to be loaded into the ZEN macro folder: **Documents\Carl Zeiss\ZEN\Documents\Macros** and that need to be run in parallel with the iCAT software.

A demonstration of this integration can be seen in this [video](#).

These macros are used in two steps:

1. During the first step, the **iCAT-mapping_setup.czmac** macro is activated. This guides the user through the calibration of a 3D space. The user is prompted to focus and center the specimen within two distinct angle orientations, thereby generating a 2D model of the specimen's orientation within the FEP tube.
2. Once the mathematical model is created, the user can activate the **iCAT-mapping_sync.czmac** macro to enable automatic synchronization of the focus and the stage with axial rotation adjustments. This feature is particularly useful when the specimen is mounted eccentrically.

Start imaging!

