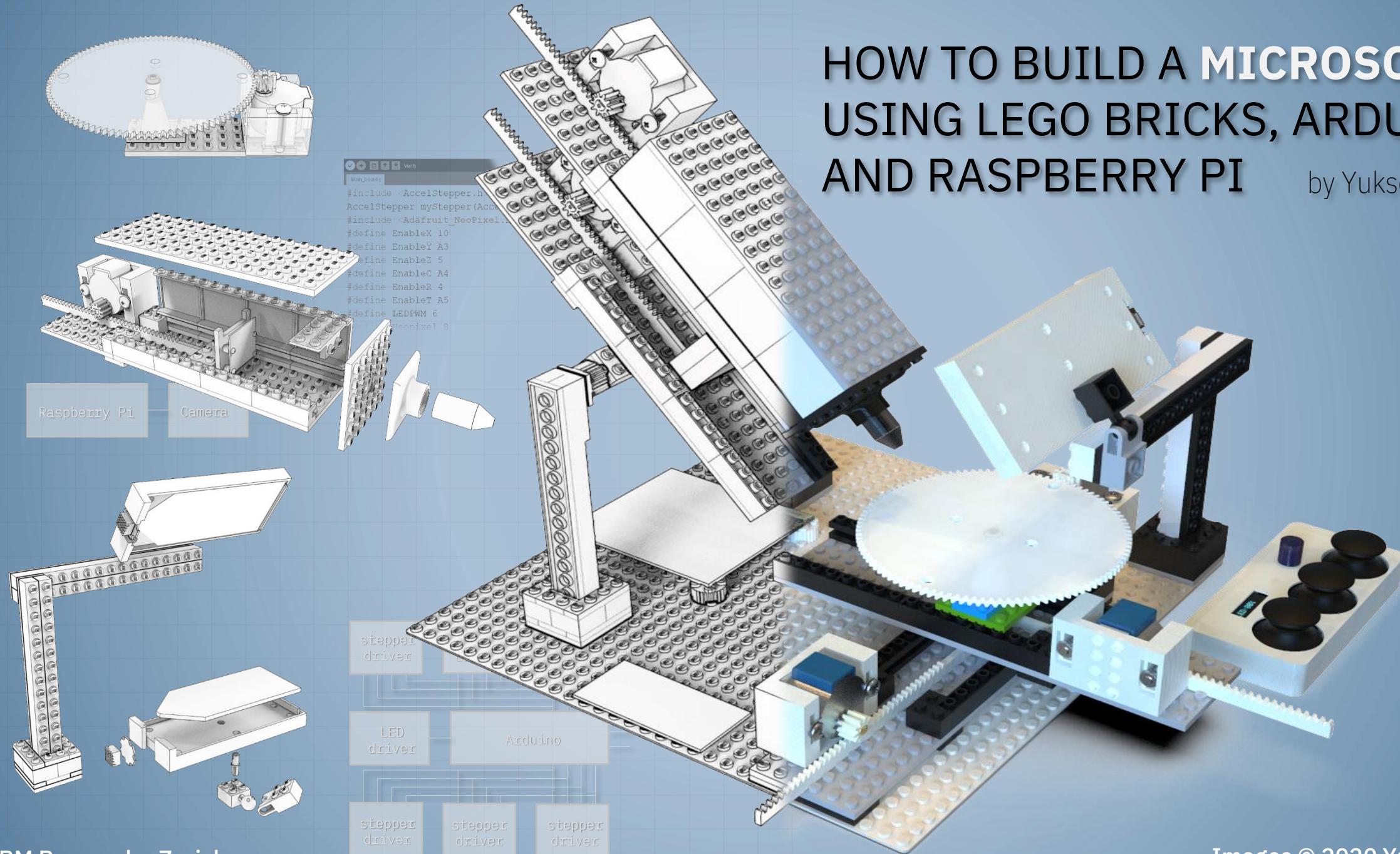
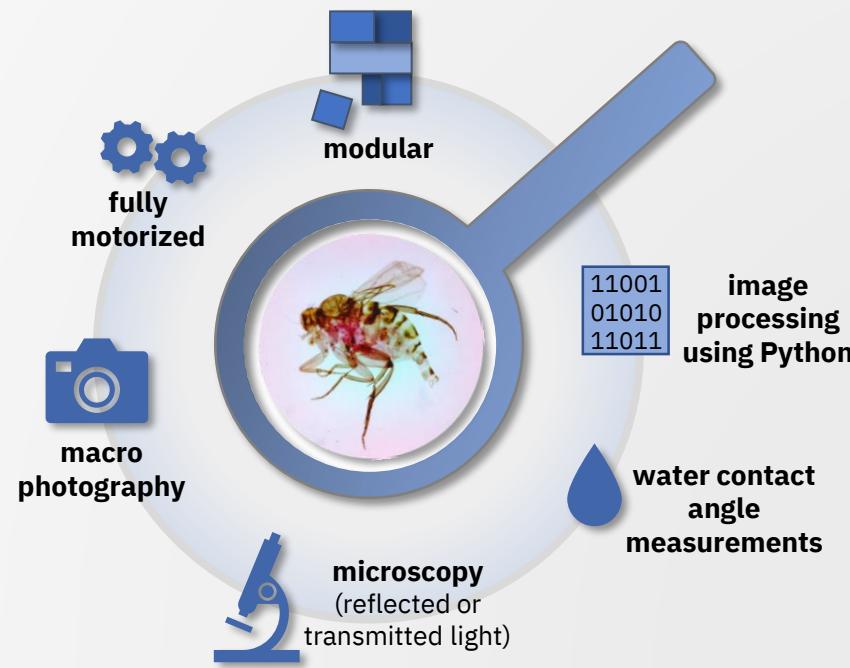


# HOW TO BUILD A **MICROSCOPE** USING LEGO BRICKS, ARDUINO AND RASPBERRY PI

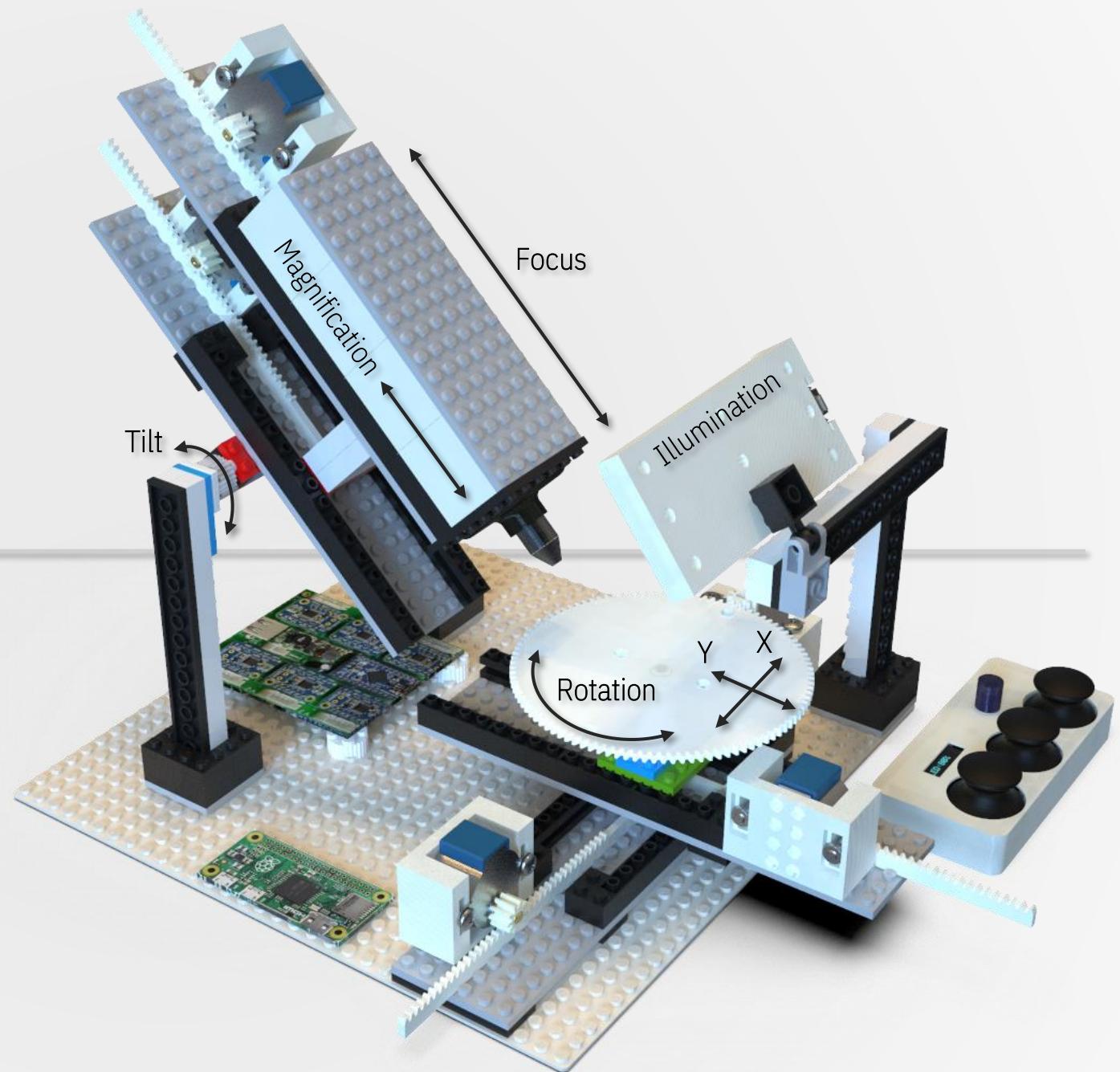
by Yuksel Temiz



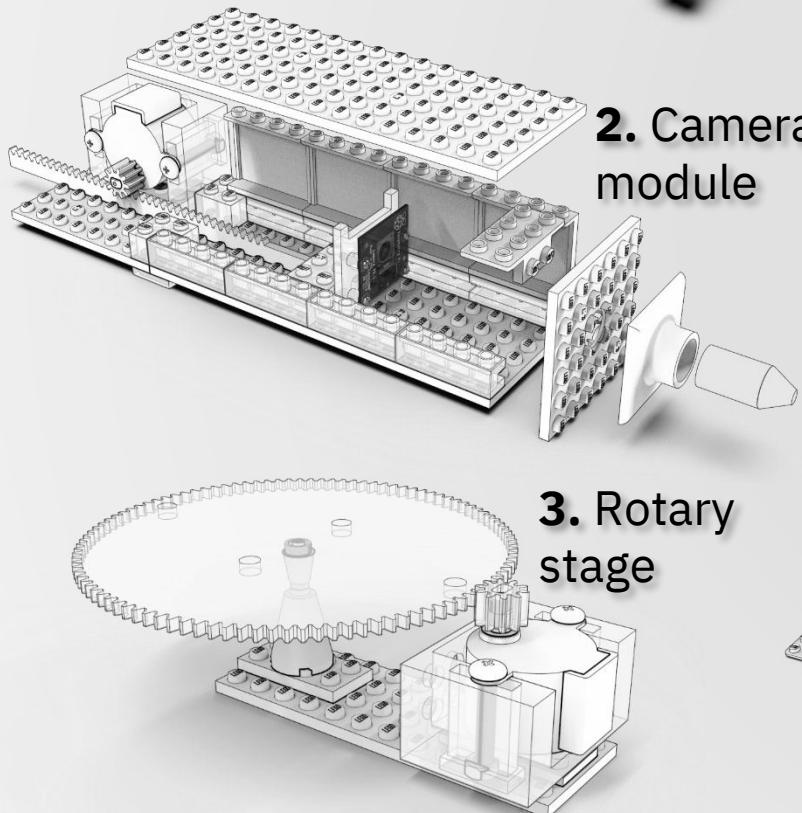


## KEY FEATURES

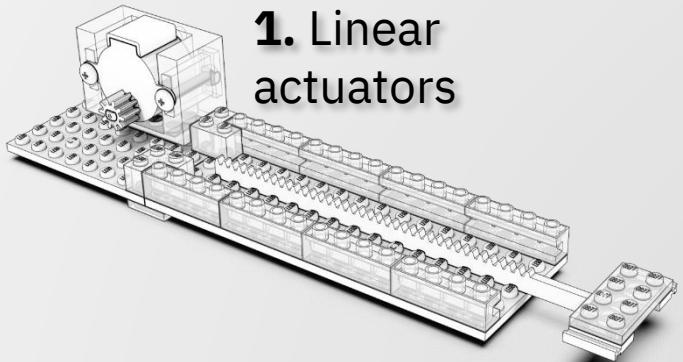
- Raspberry Pi operation system
- 8 MegaPixel CMOS camera (Full HD 30 fps video)
- Imaging features from several centimeters to several micrometers without changing the lens
- 6 stepper motors (X, Y, tilt, rotation, magnification, and focus)
- Variable speed control using a joystick controller or a keyboard
- Uniform illumination for imaging reflective surfaces
- Modular design: stages and modules can be arranged in any configuration depending on the application



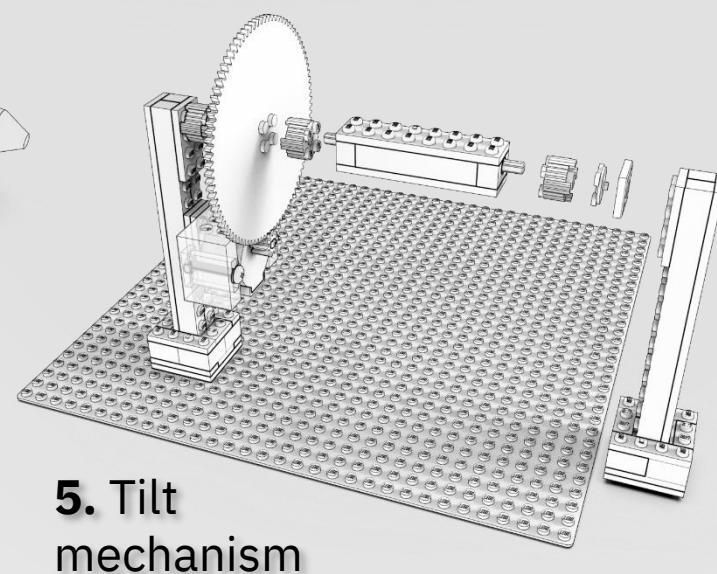
# Contents



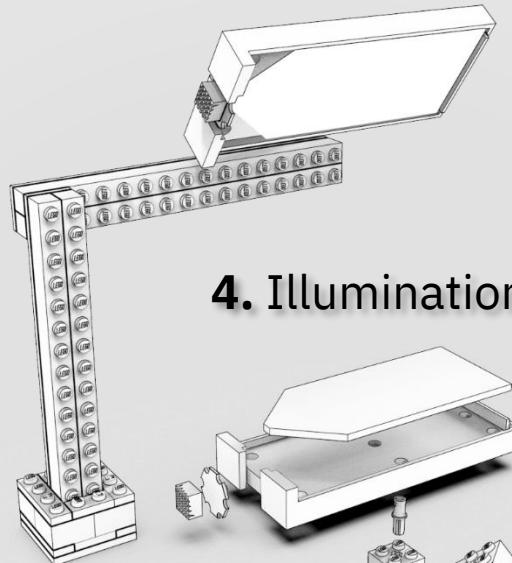
**2. Camera module**



**1. Linear actuators**

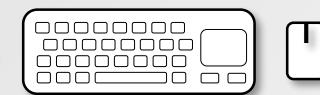


**5. Tilt mechanism**



**4. Illumination**

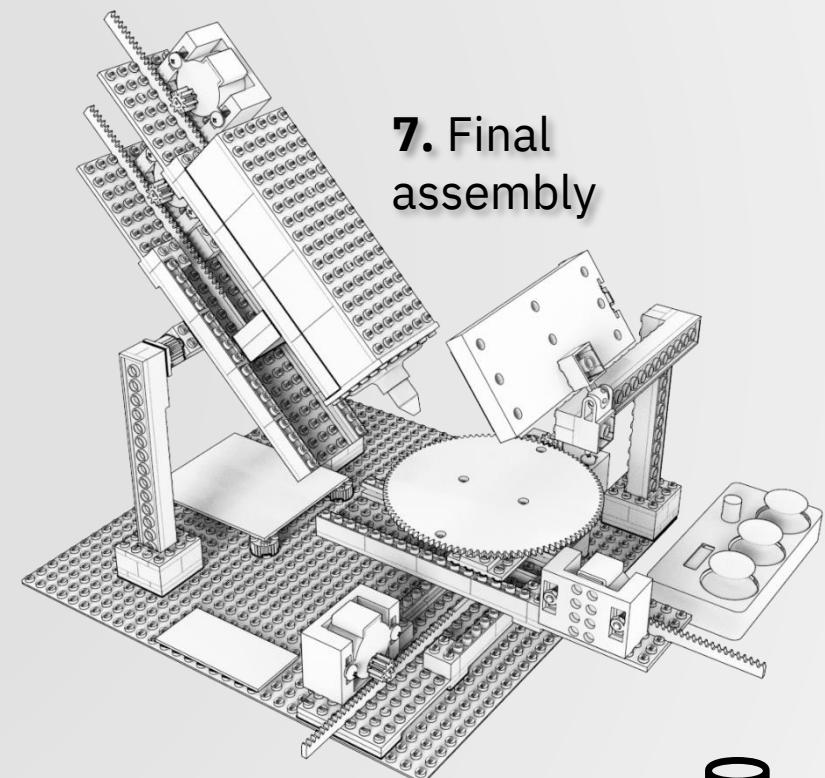
**8. Software**



**6. Electronics**



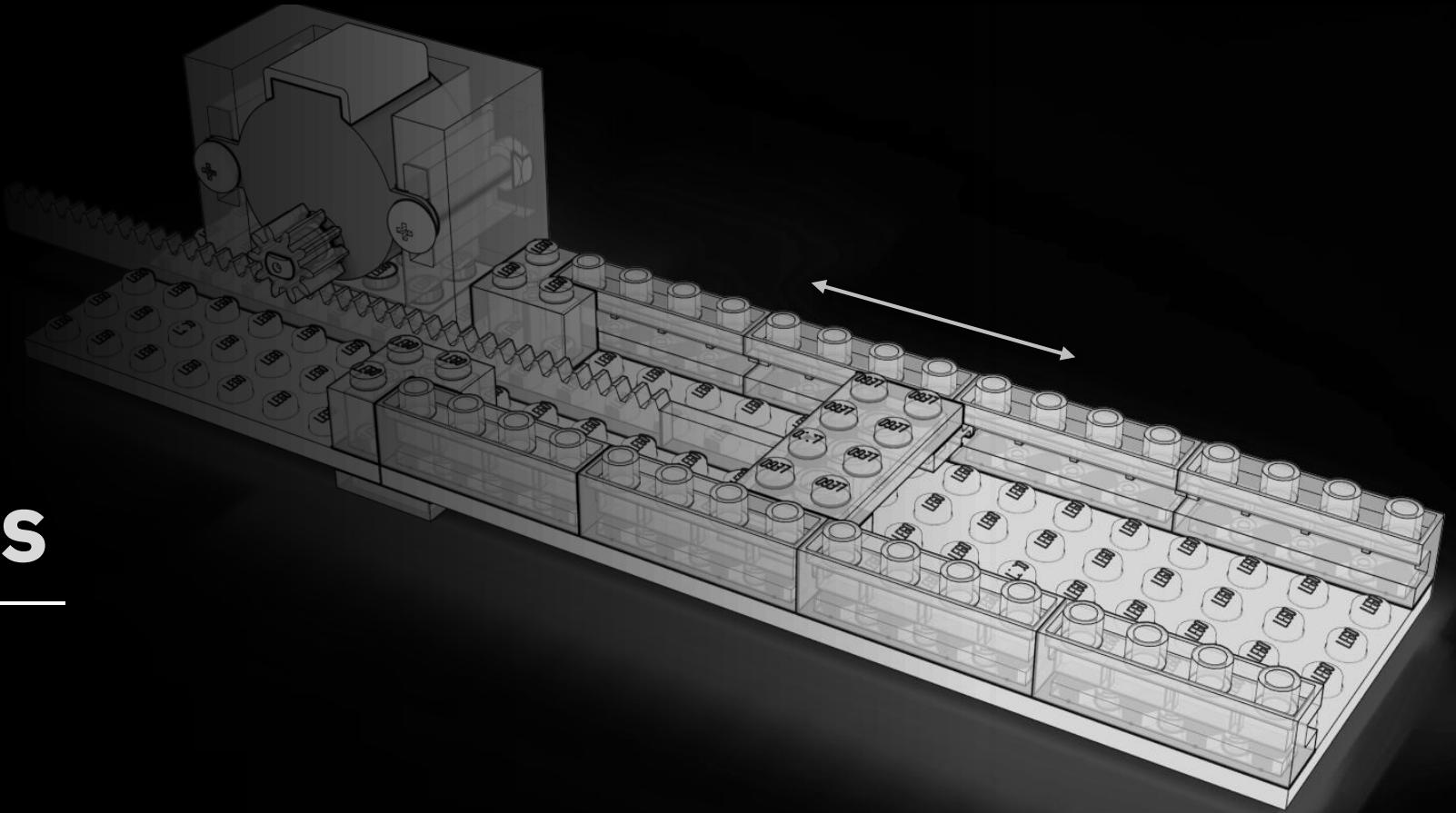
**7. Final assembly**



**BOM**

# Linear actuators

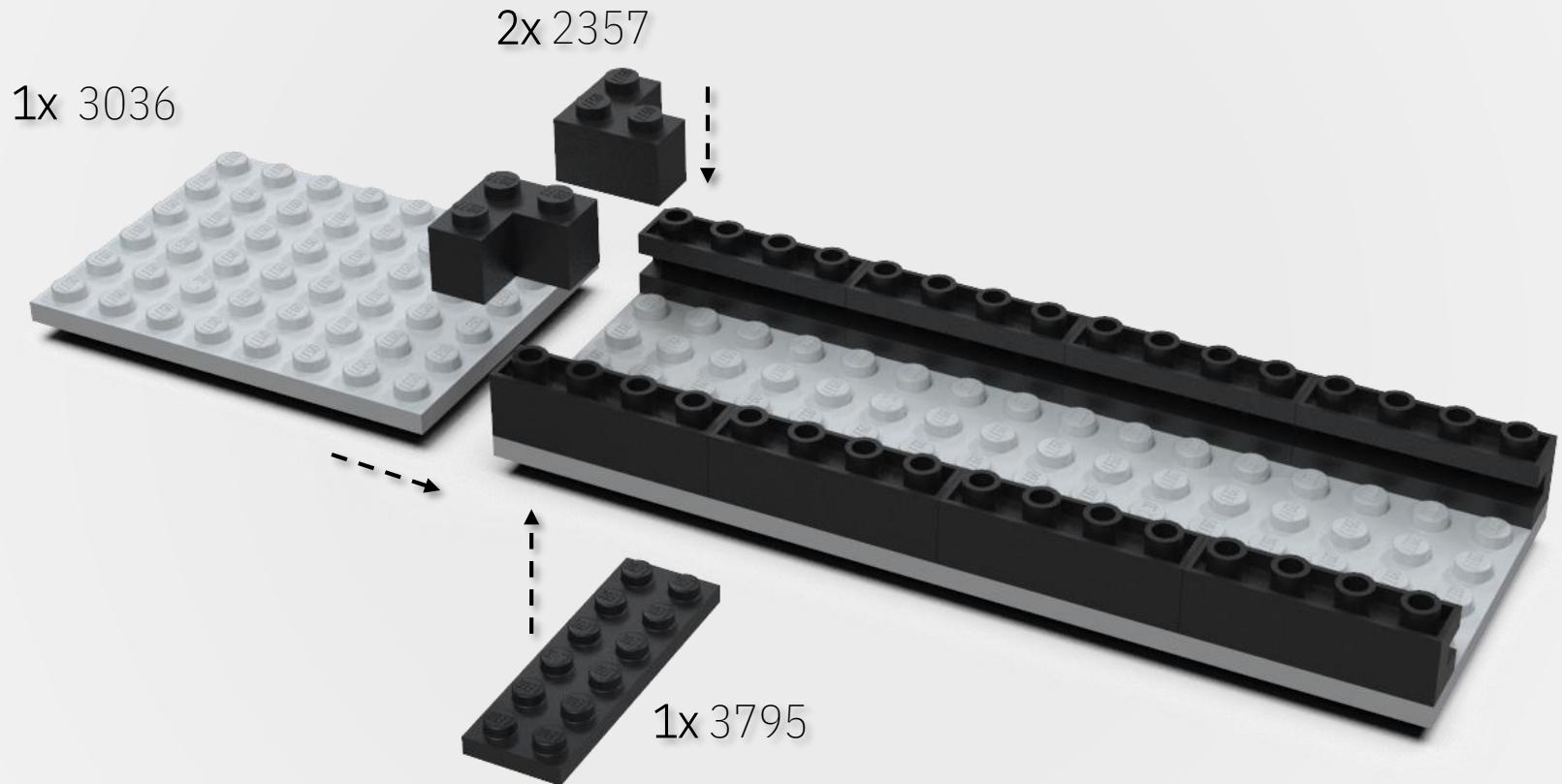
---



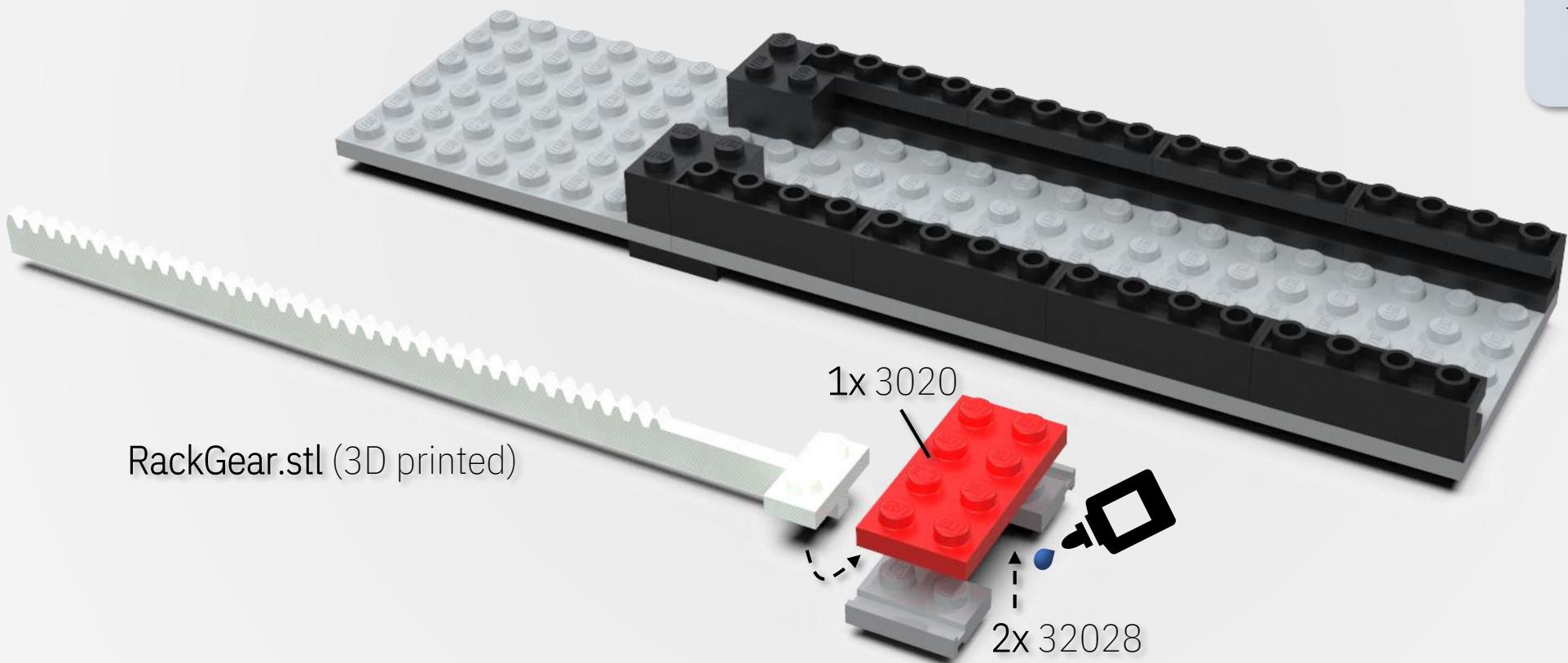
# Linear actuators



# Linear actuators

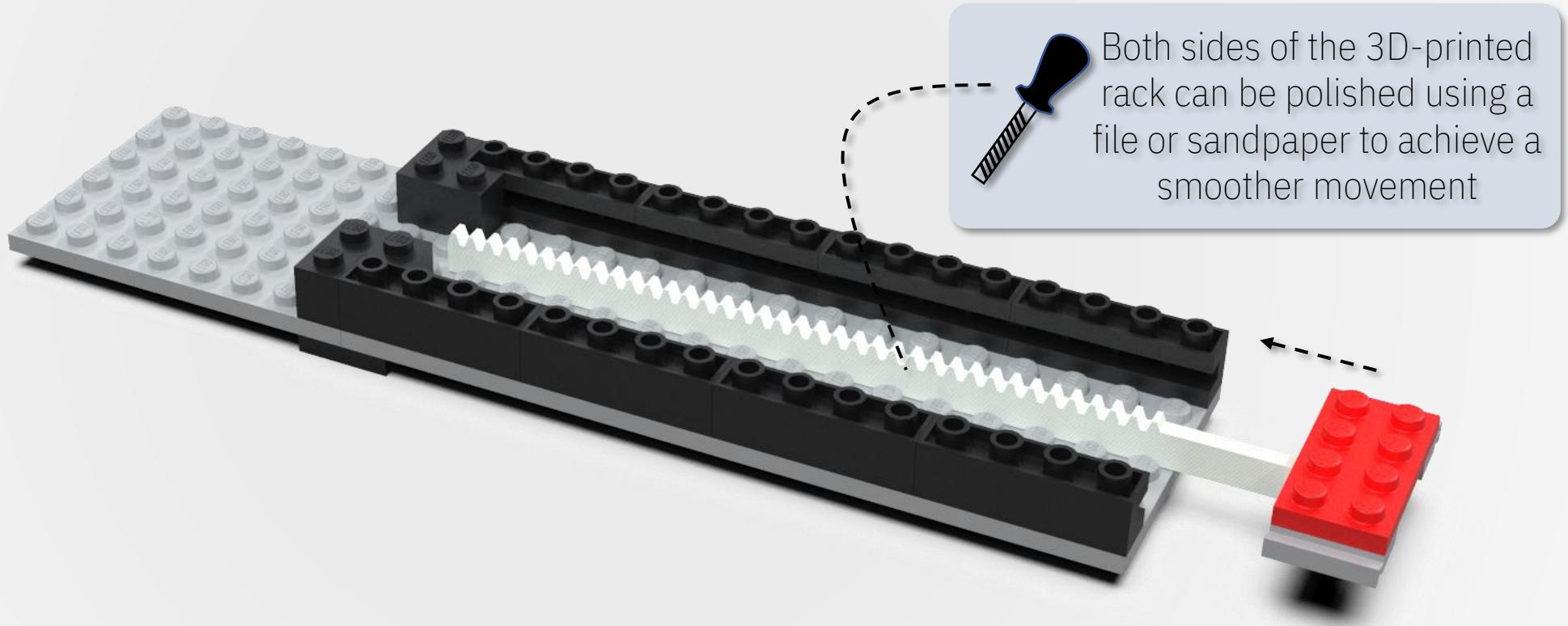


# Linear actuators

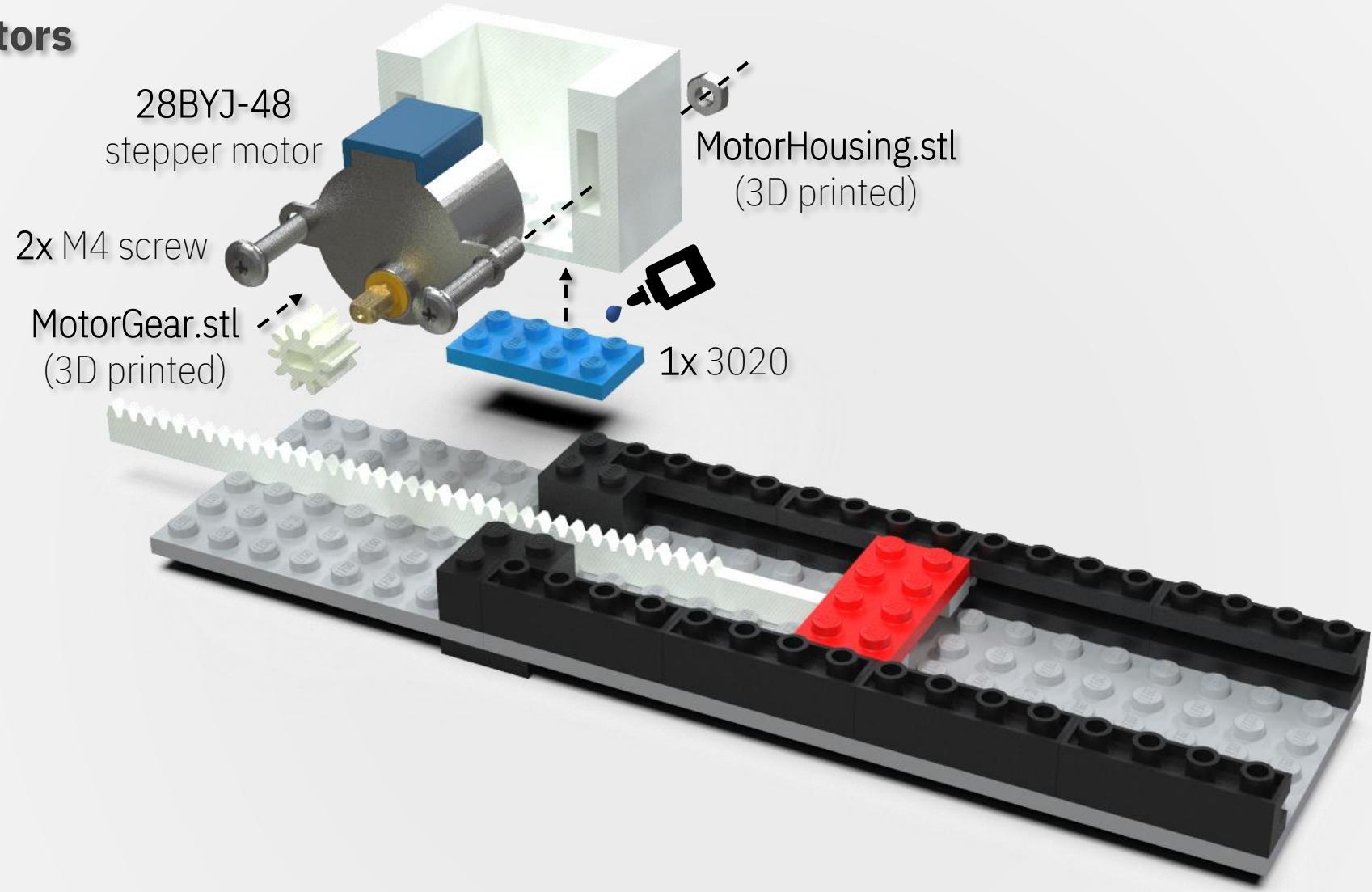


Super glue  
Some parts need  
to be glued for  
better stability

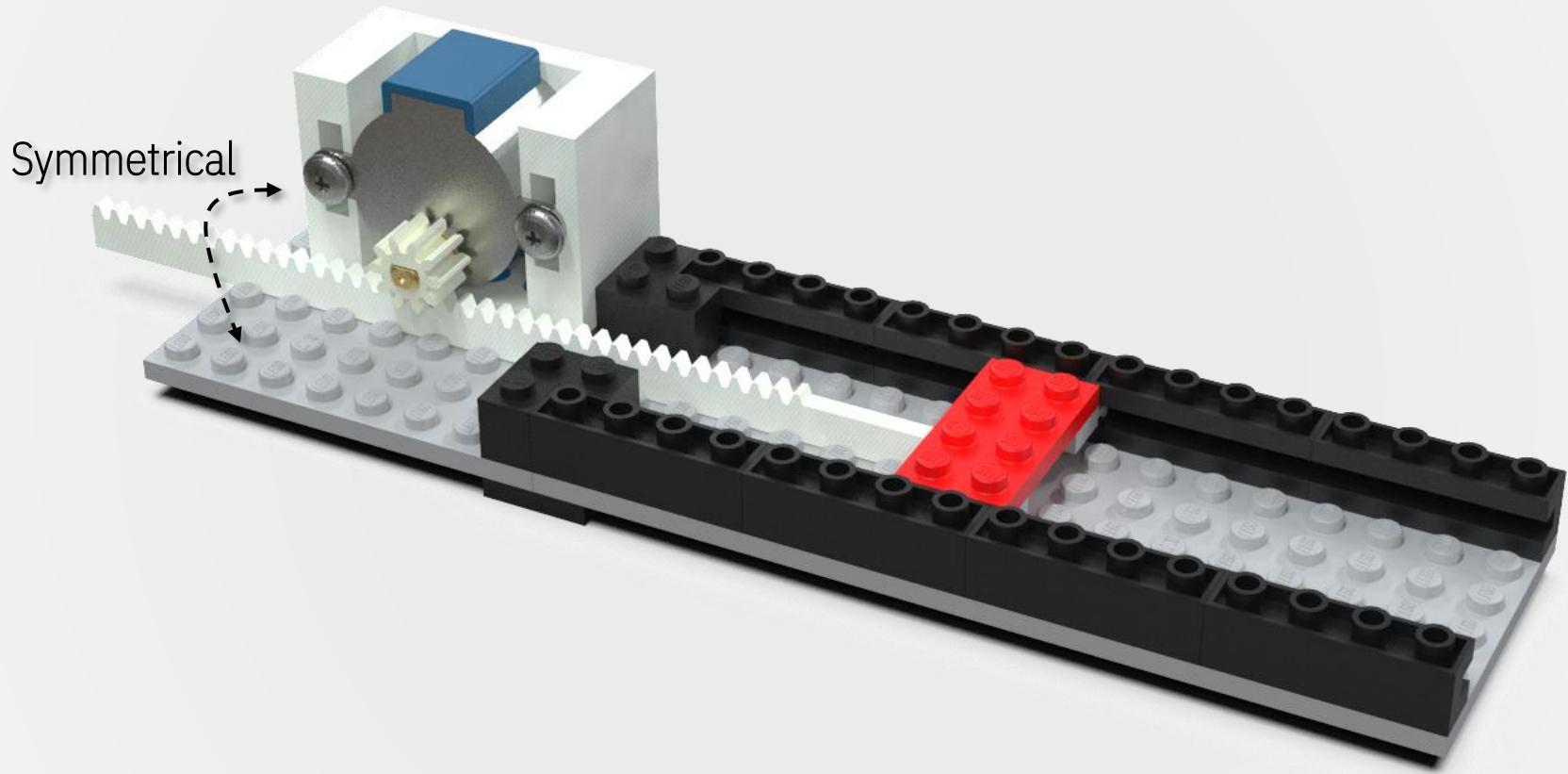
# Linear actuators



# Linear actuators



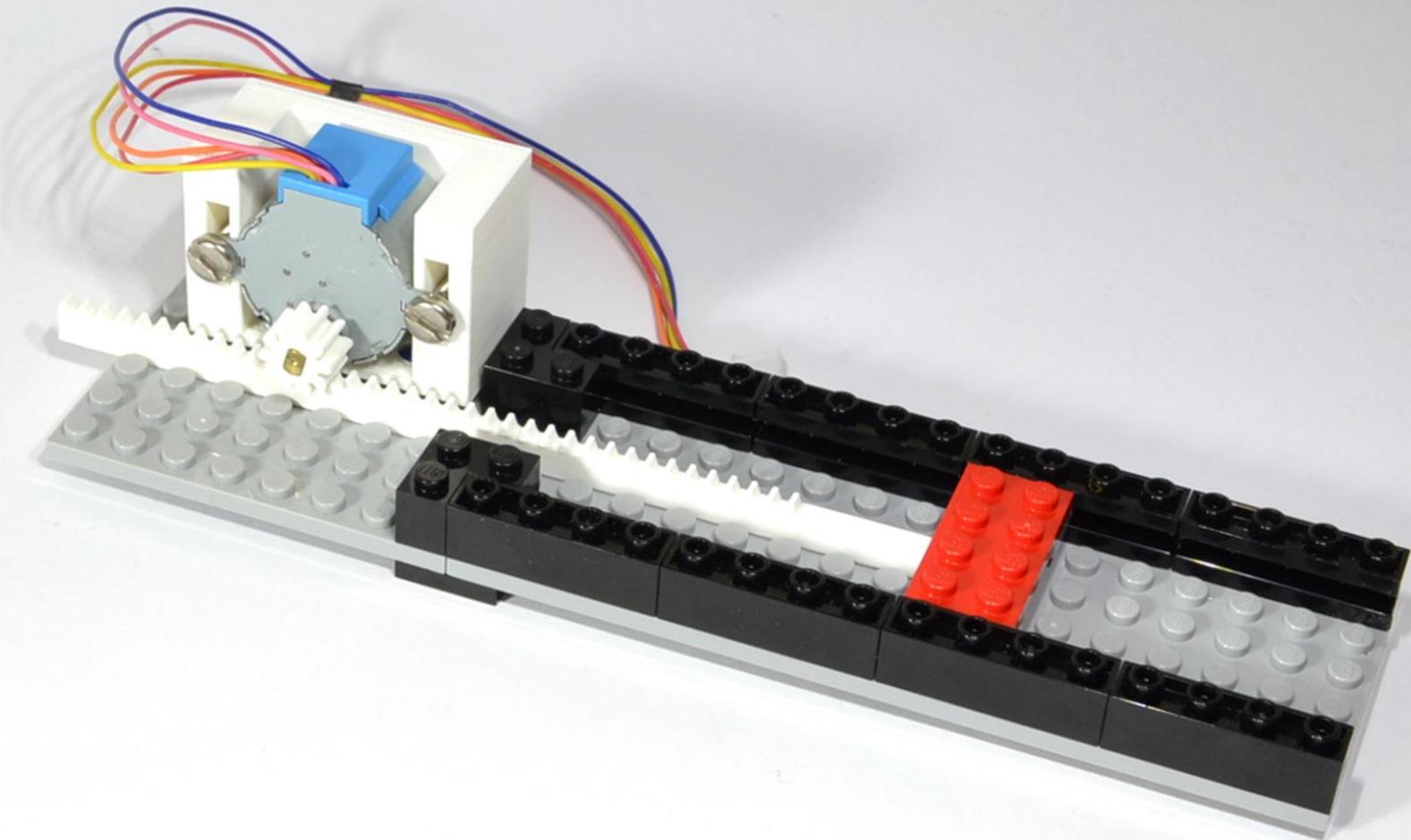
# Linear actuators



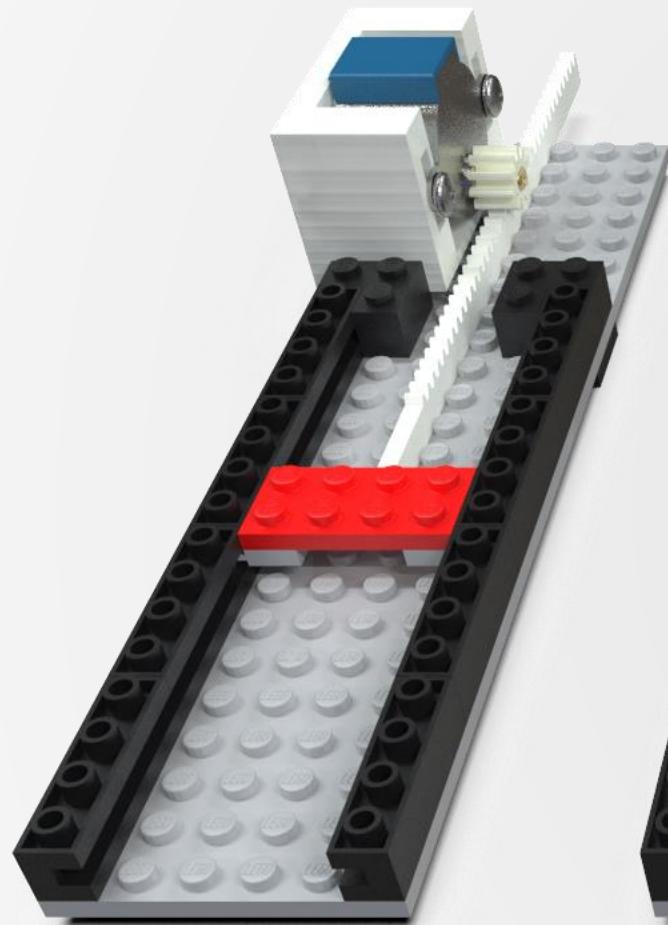
The motor can be positioned on either side of the rack. This would be useful to quickly change the movement direction of the stages with respect to the joystick action (the direction can also be changed from the software).

# Linear actuators

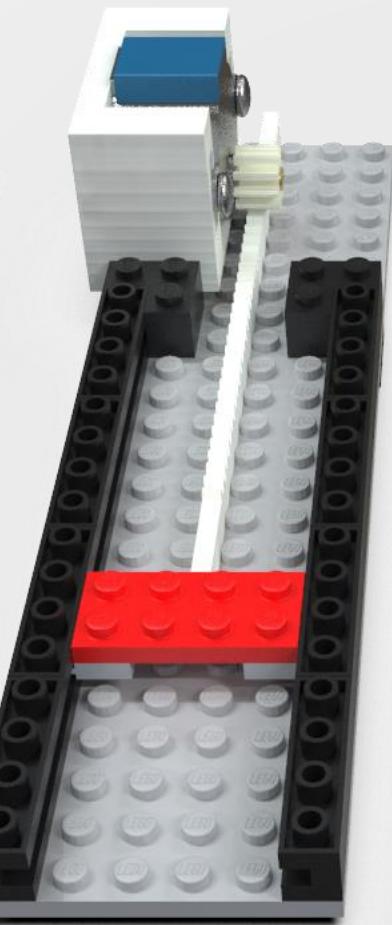
(Photograph)



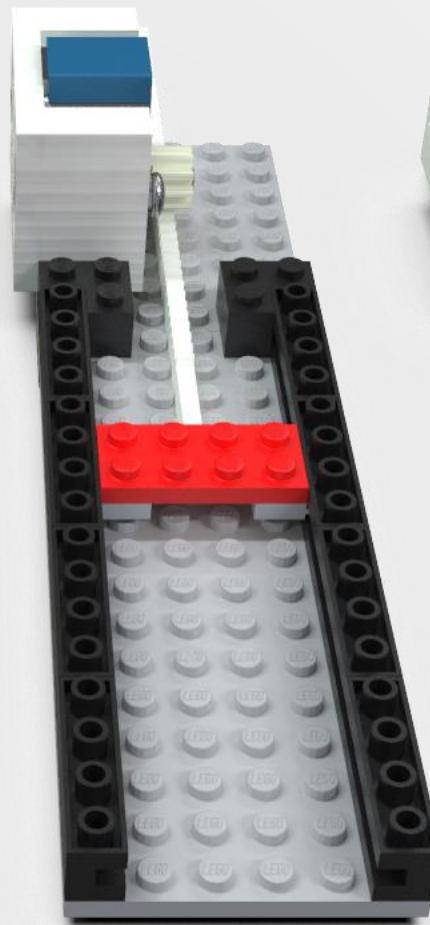
# Linear actuators – 4x



**X**



**Y**

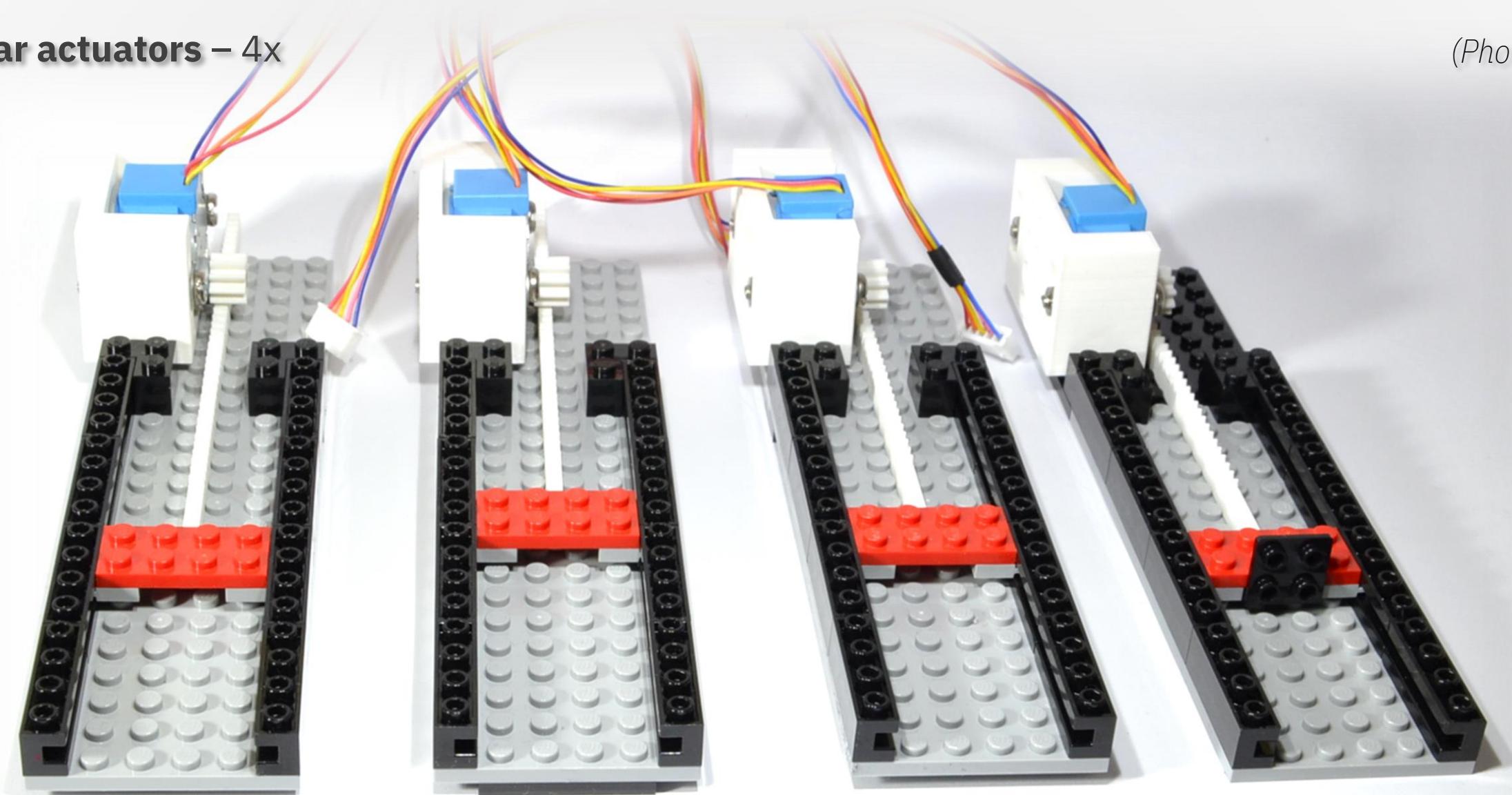


**Z**

**Camera**

# Linear actuators – 4x

(Photograph)



**X**

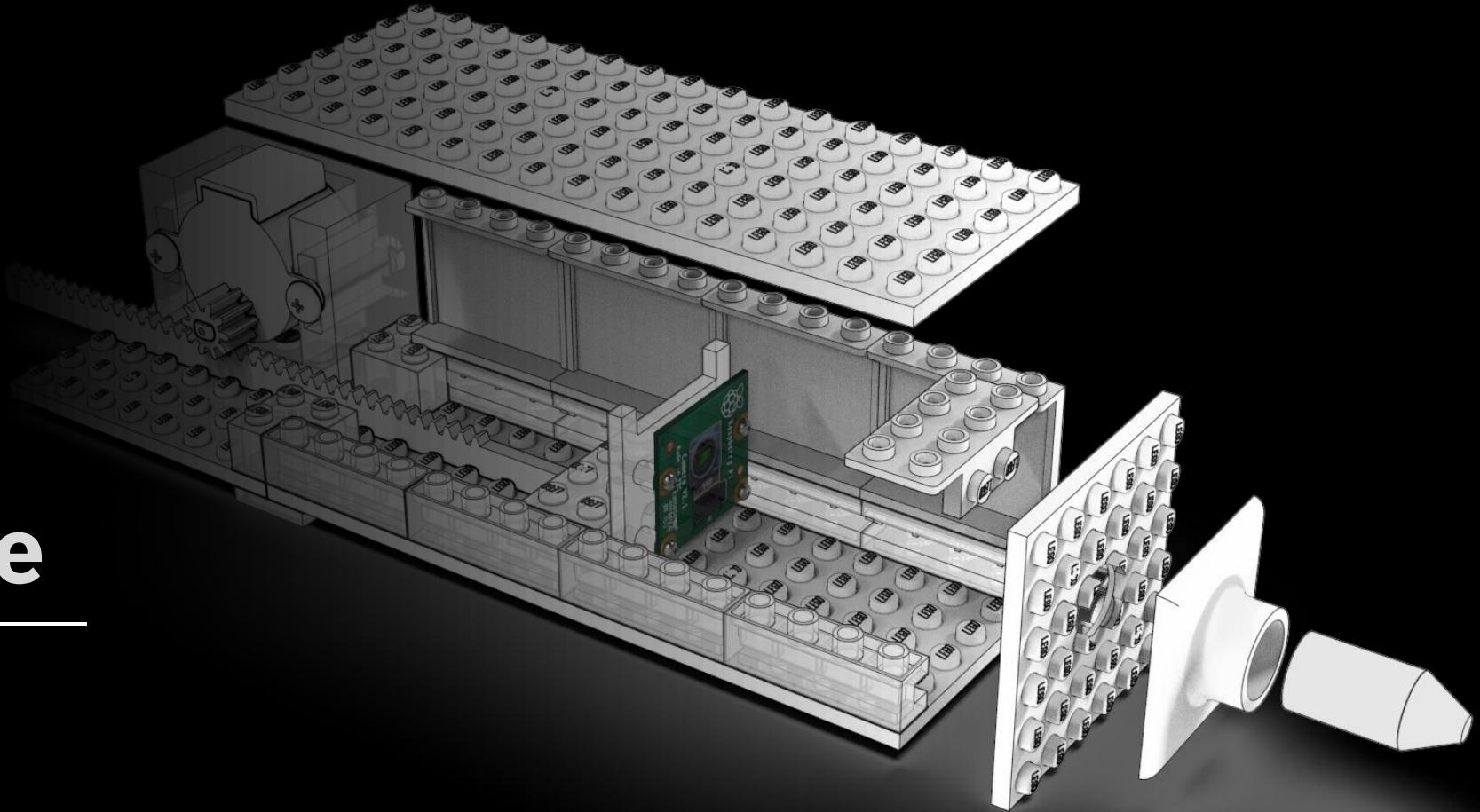
**Y**

**Z**

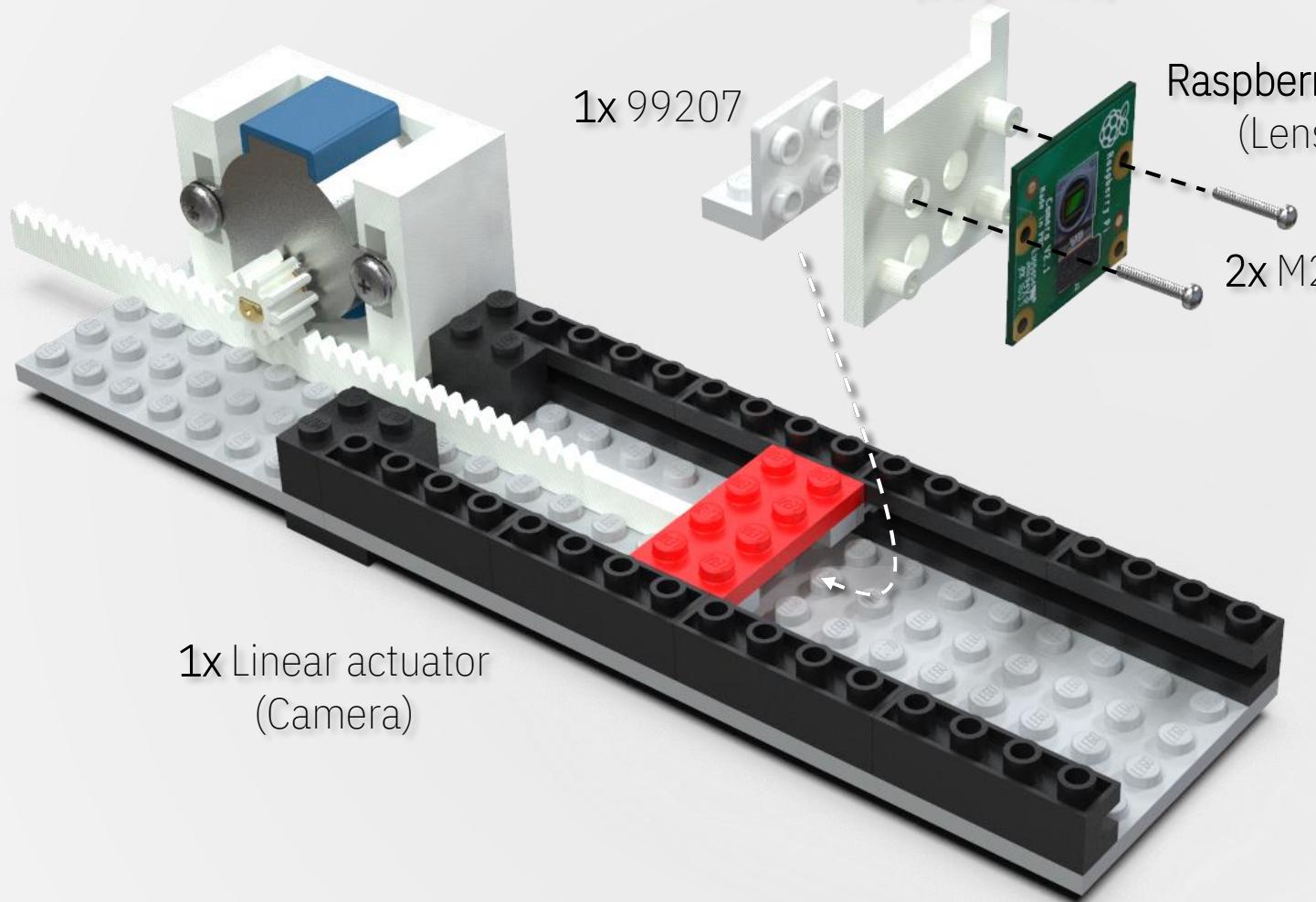
**Camera**

# Camera module

---



# Camera module



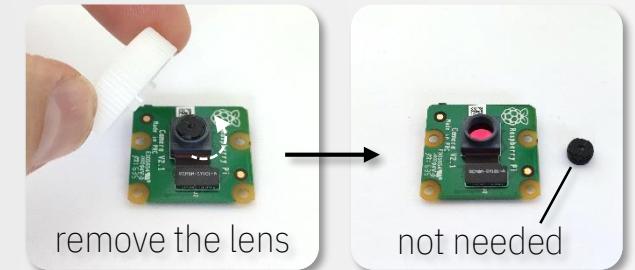
1x Linear actuator  
(Camera)

1x 99207

CameraAdapter.stl  
(3D printed)

Raspberry Pi camera v2  
(Lens removed)

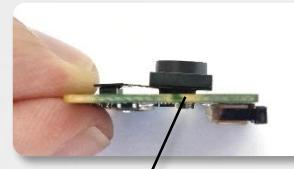
2x M2



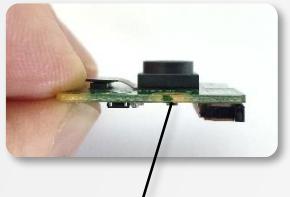
remove the lens

not needed

Recommendation:

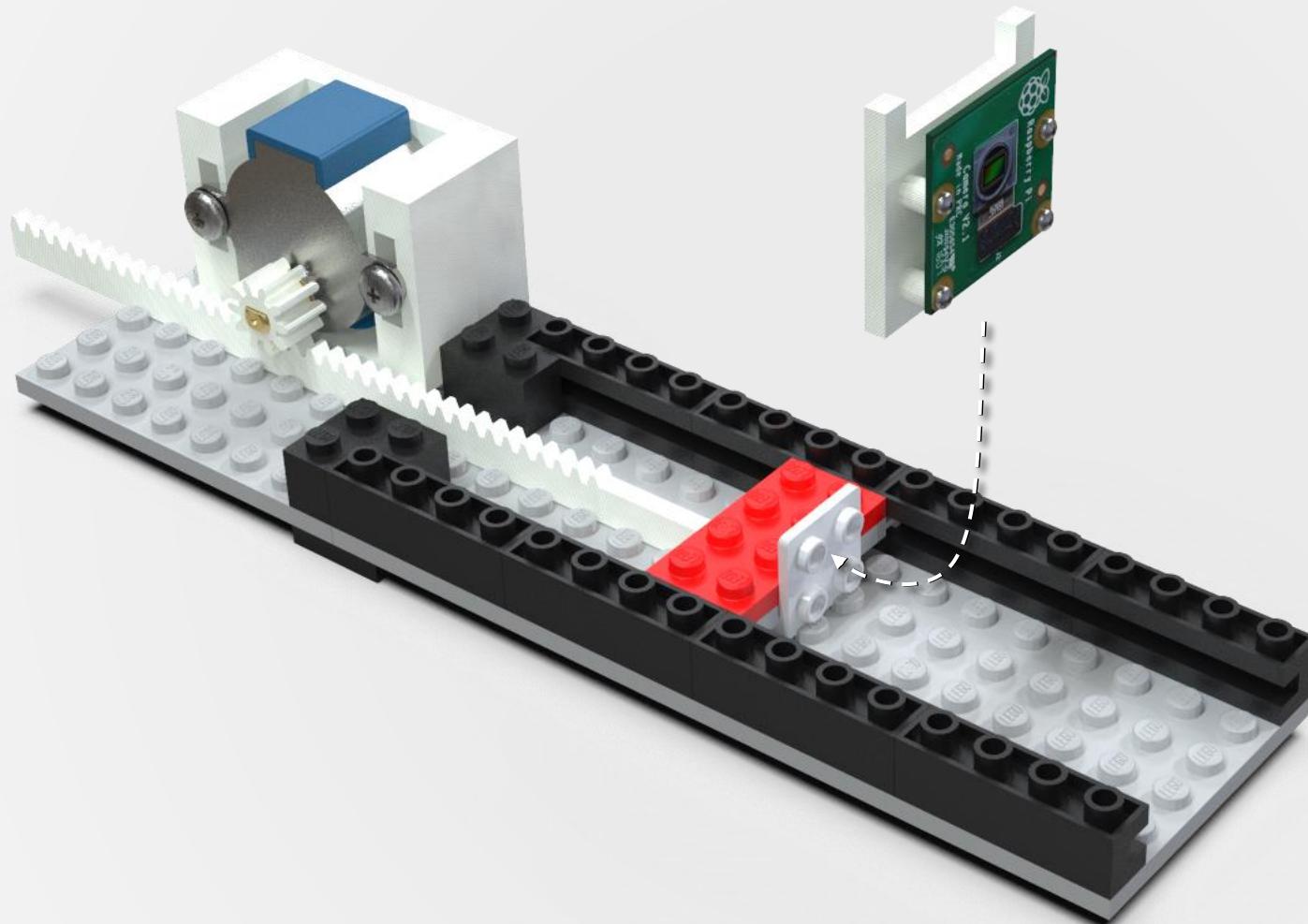


remove the  
rubber spacer

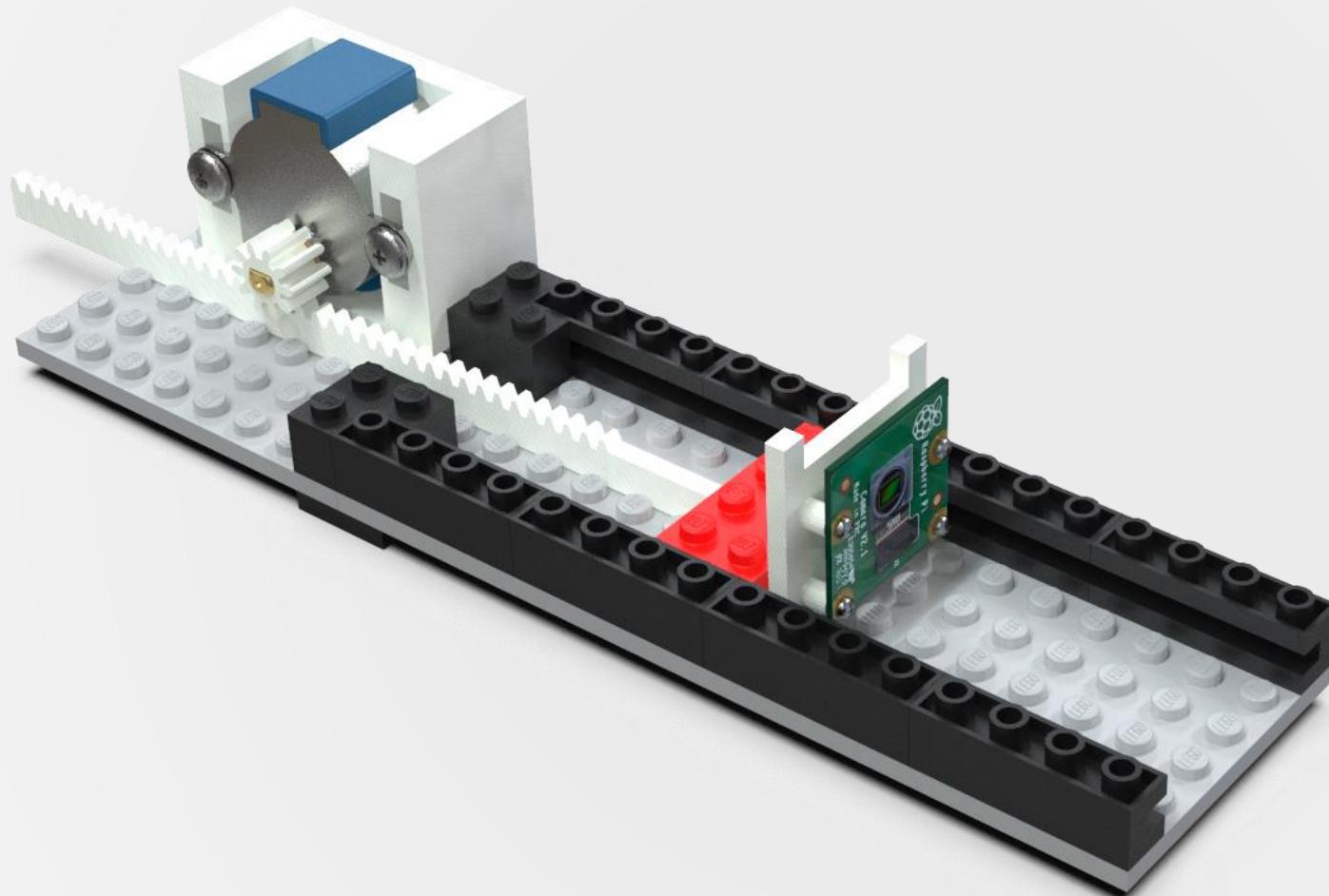


glue the image  
sensor directly  
to the board

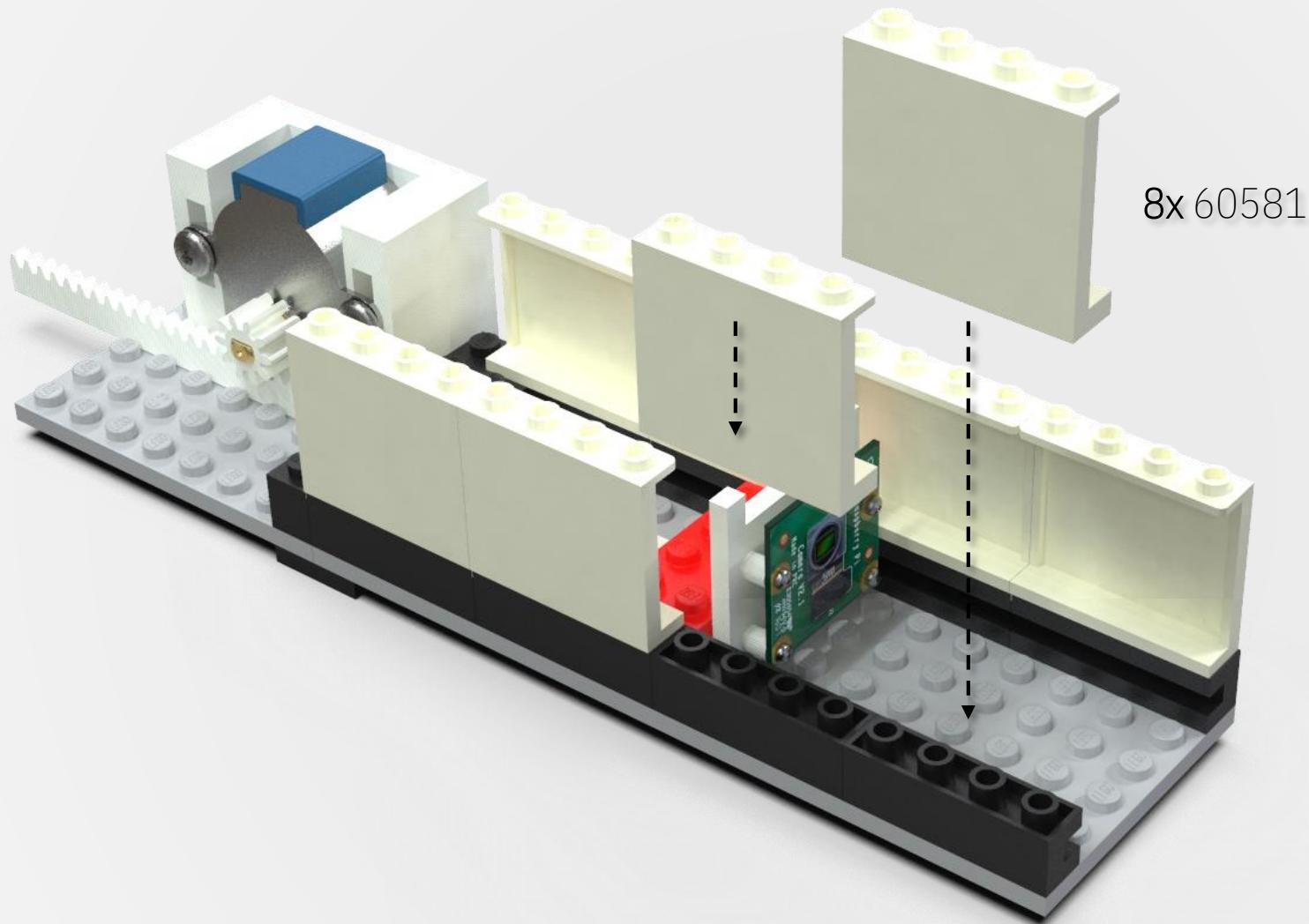
## Camera module



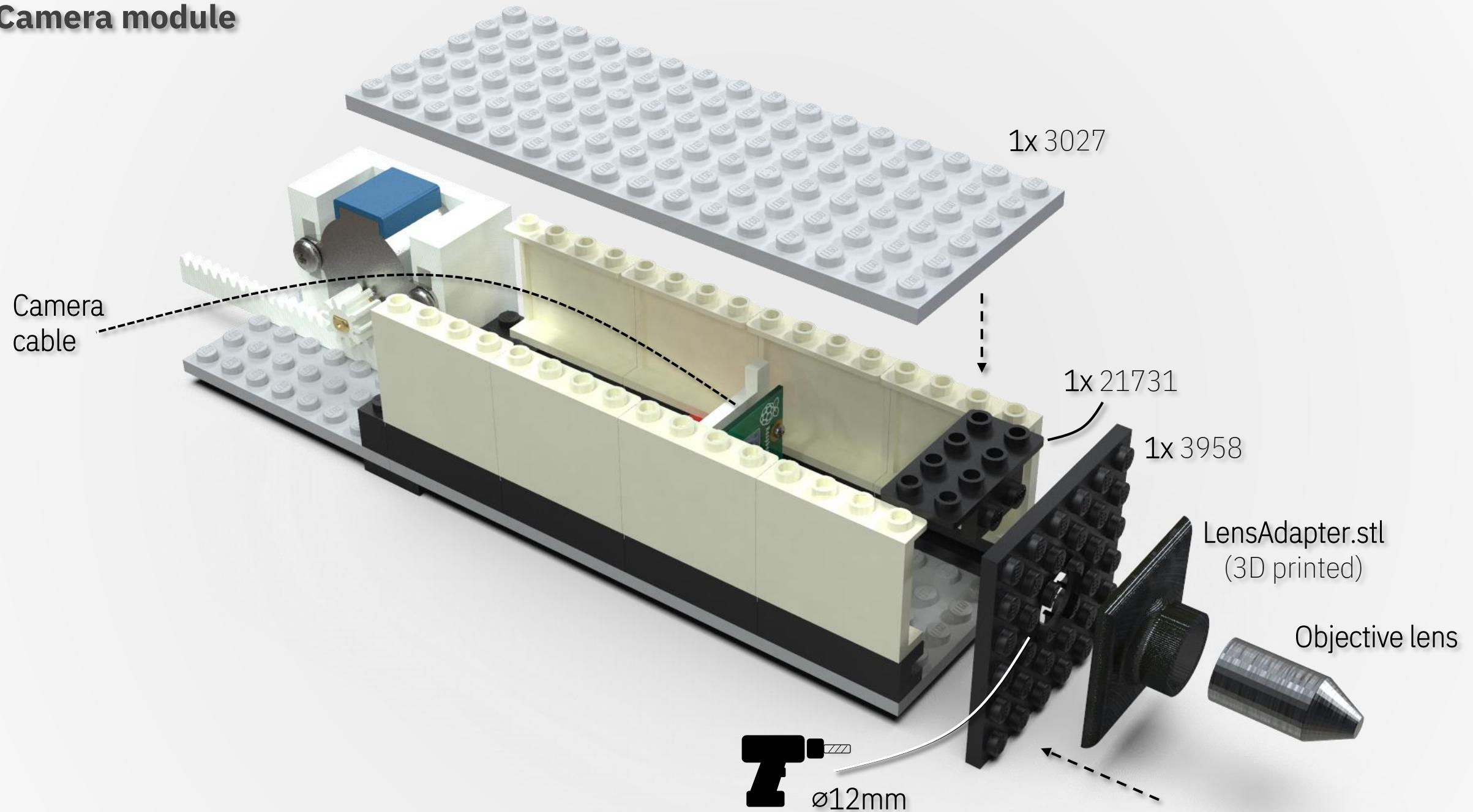
## Camera module



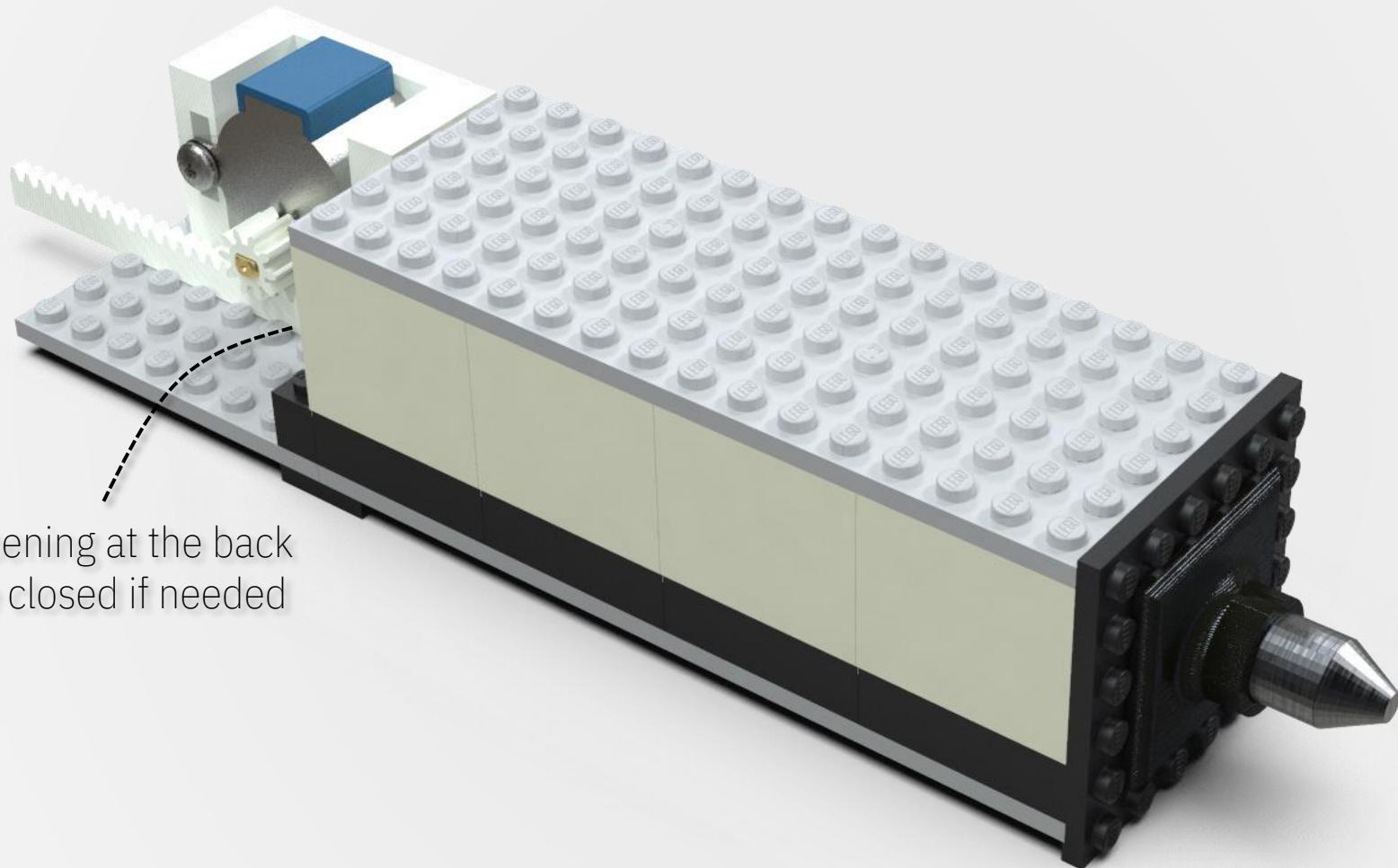
## Camera module



# Camera module

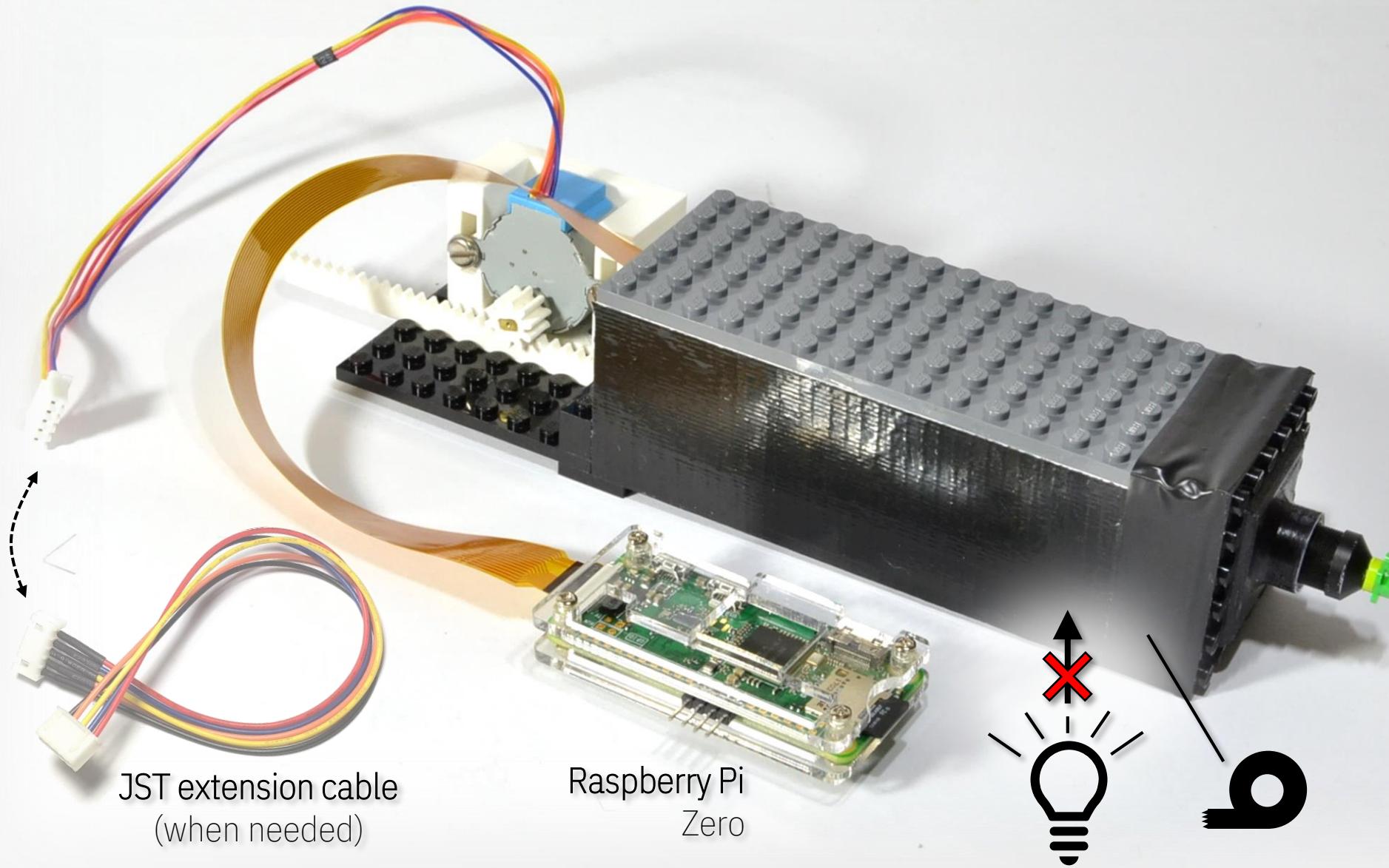


## Camera module



# Camera module

(Photograph)

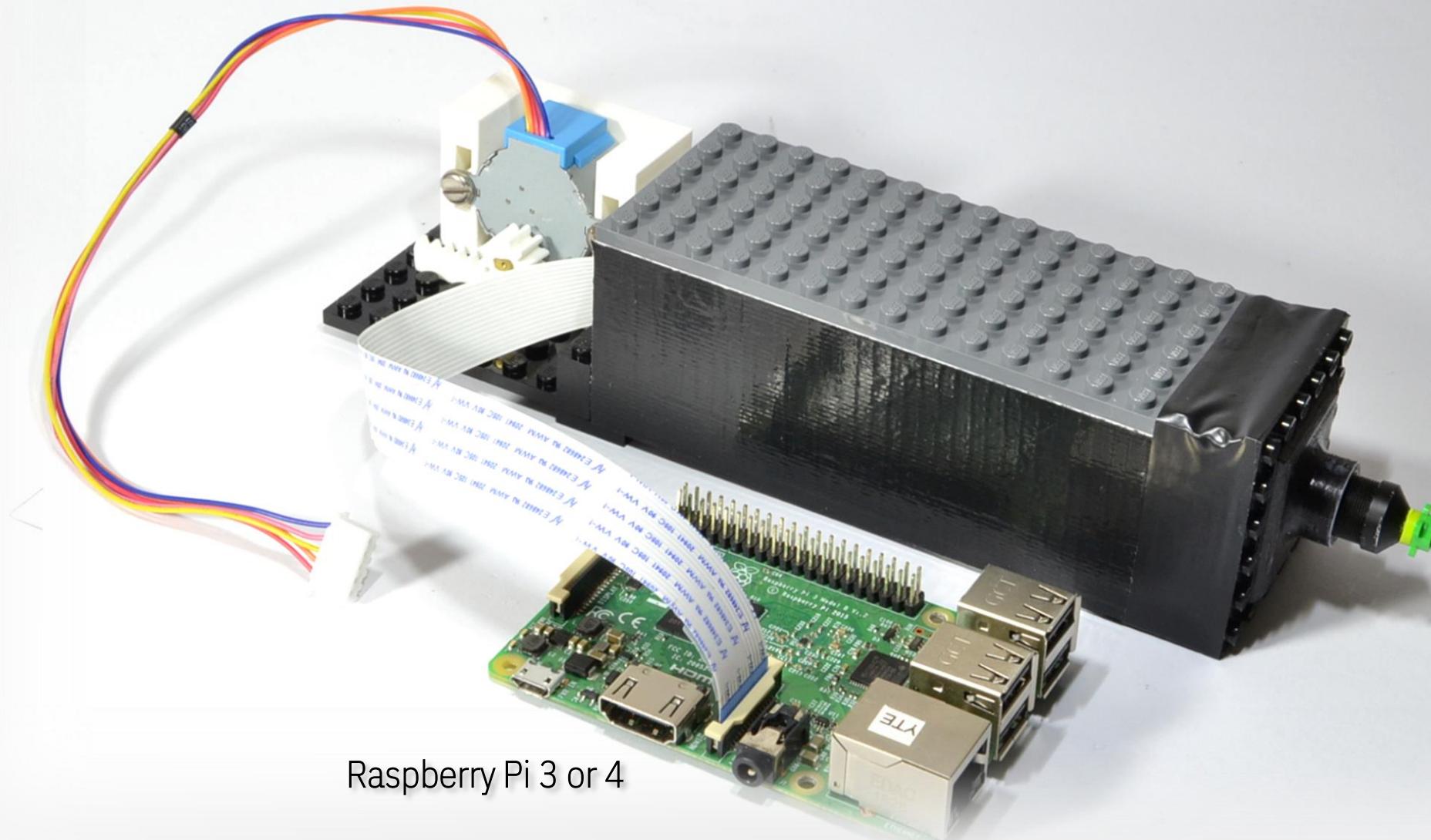


Smaller aperture  
(when needed)



# Camera module

(Photograph)



# Camera module

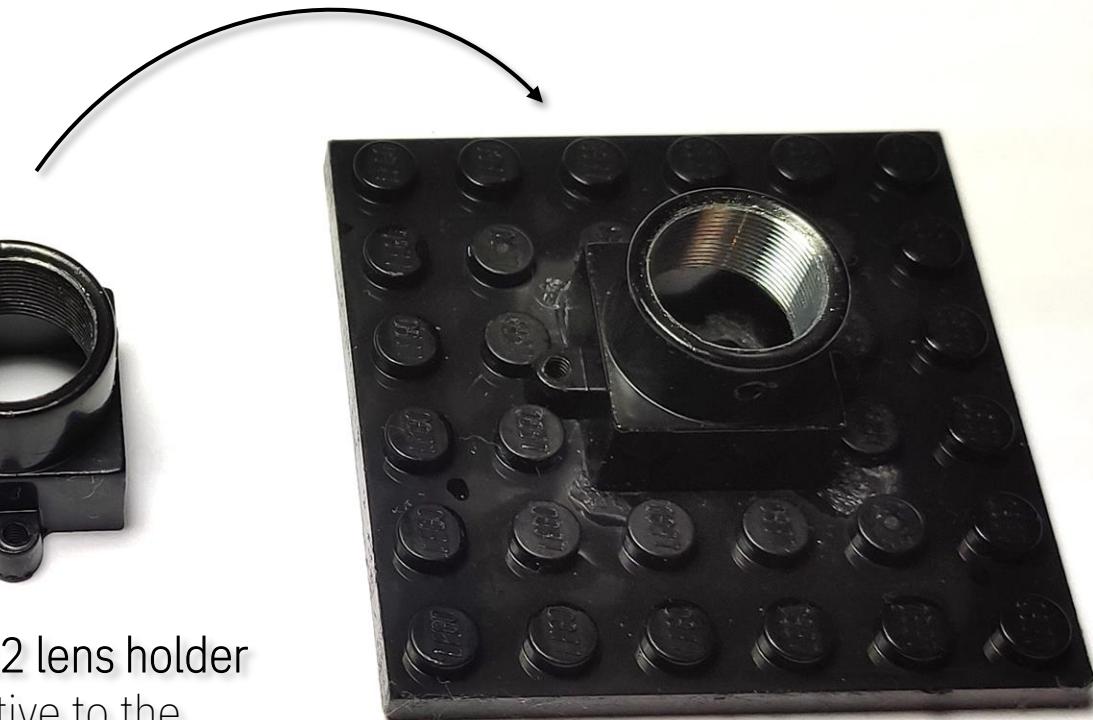
(Photograph)



TREEYE 15.3mm  
magnifying lens

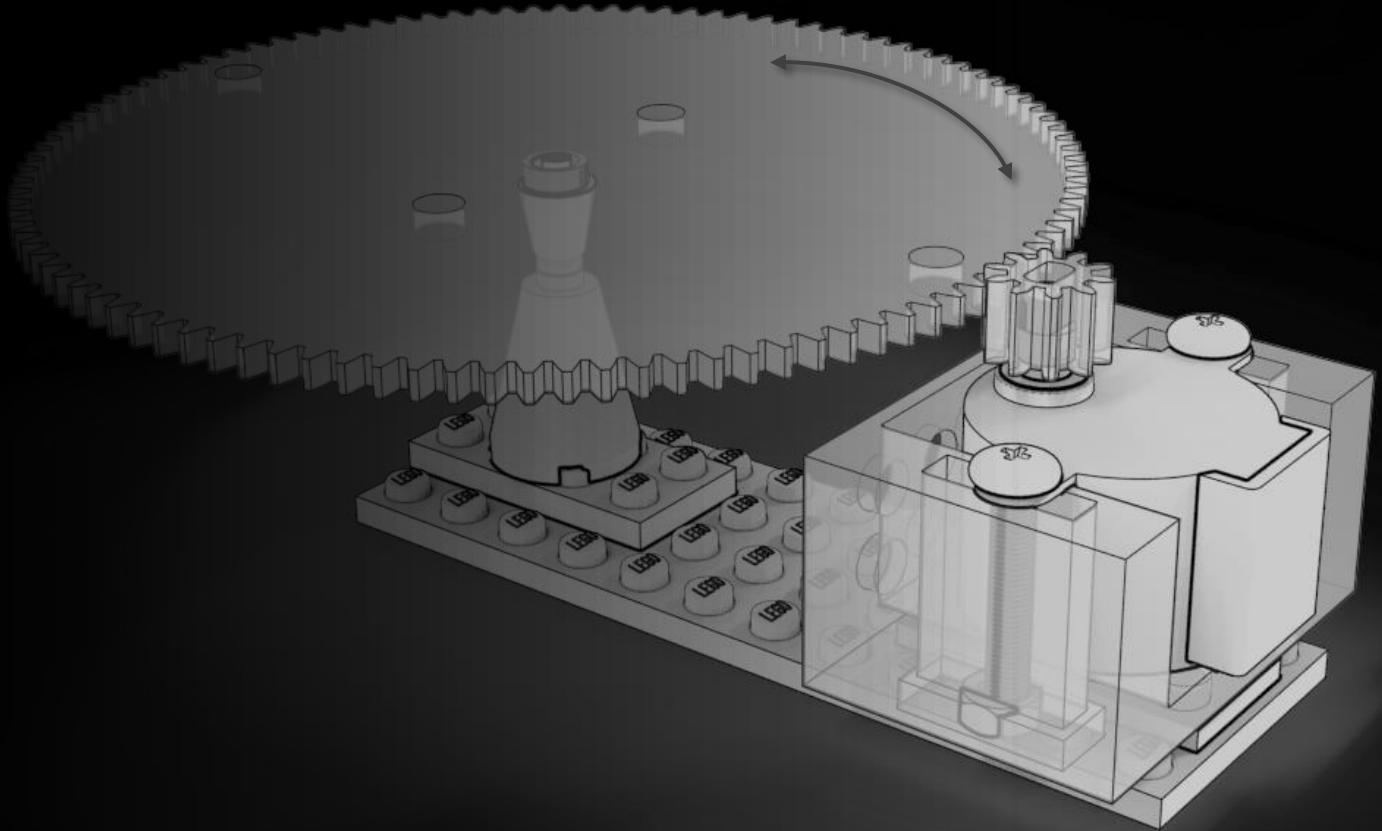


TREEYE M12 lens holder  
(alternative to the  
3D printed adapter)

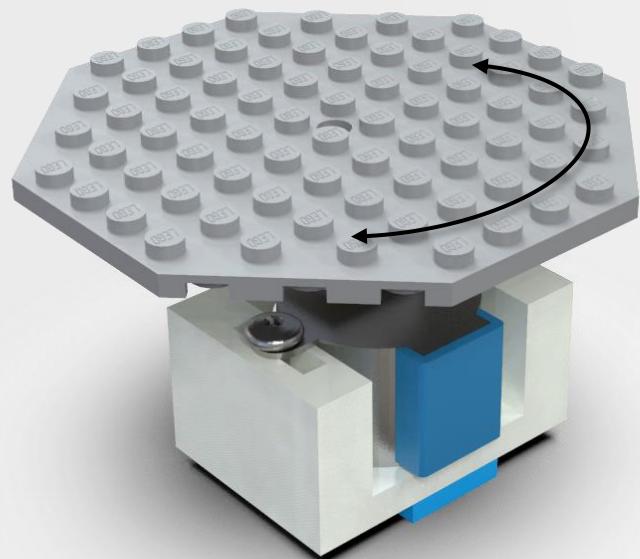
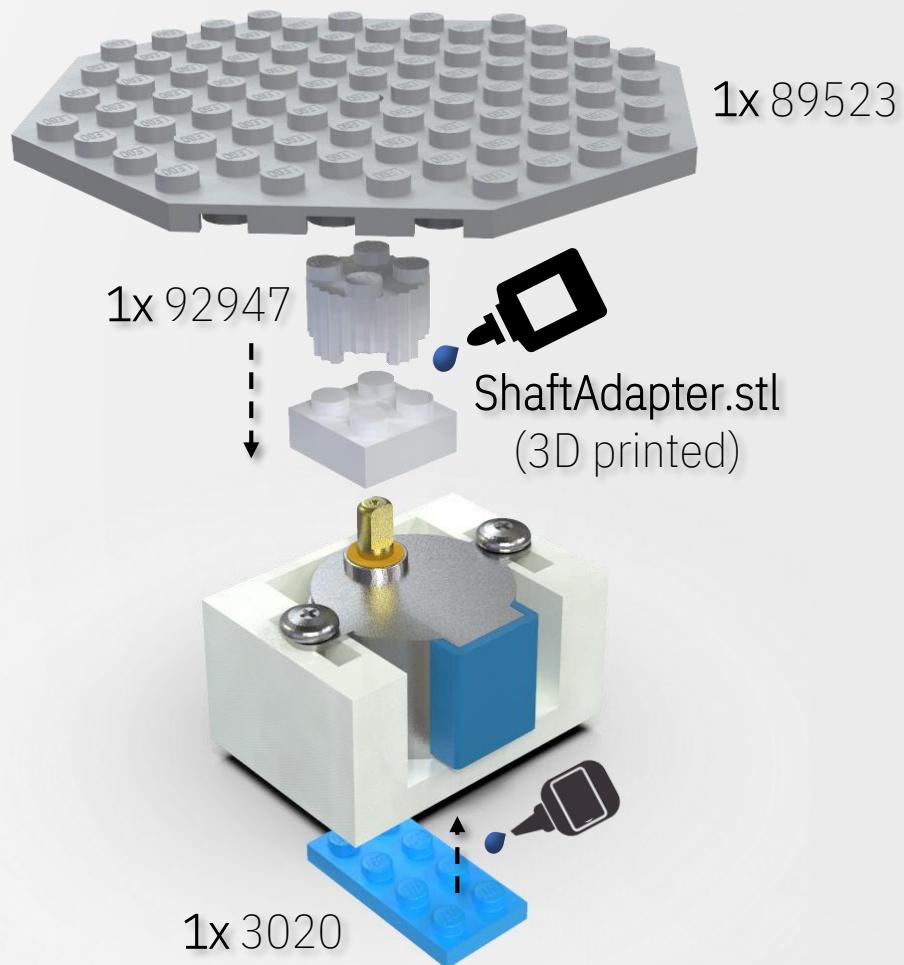


# Rotary stage

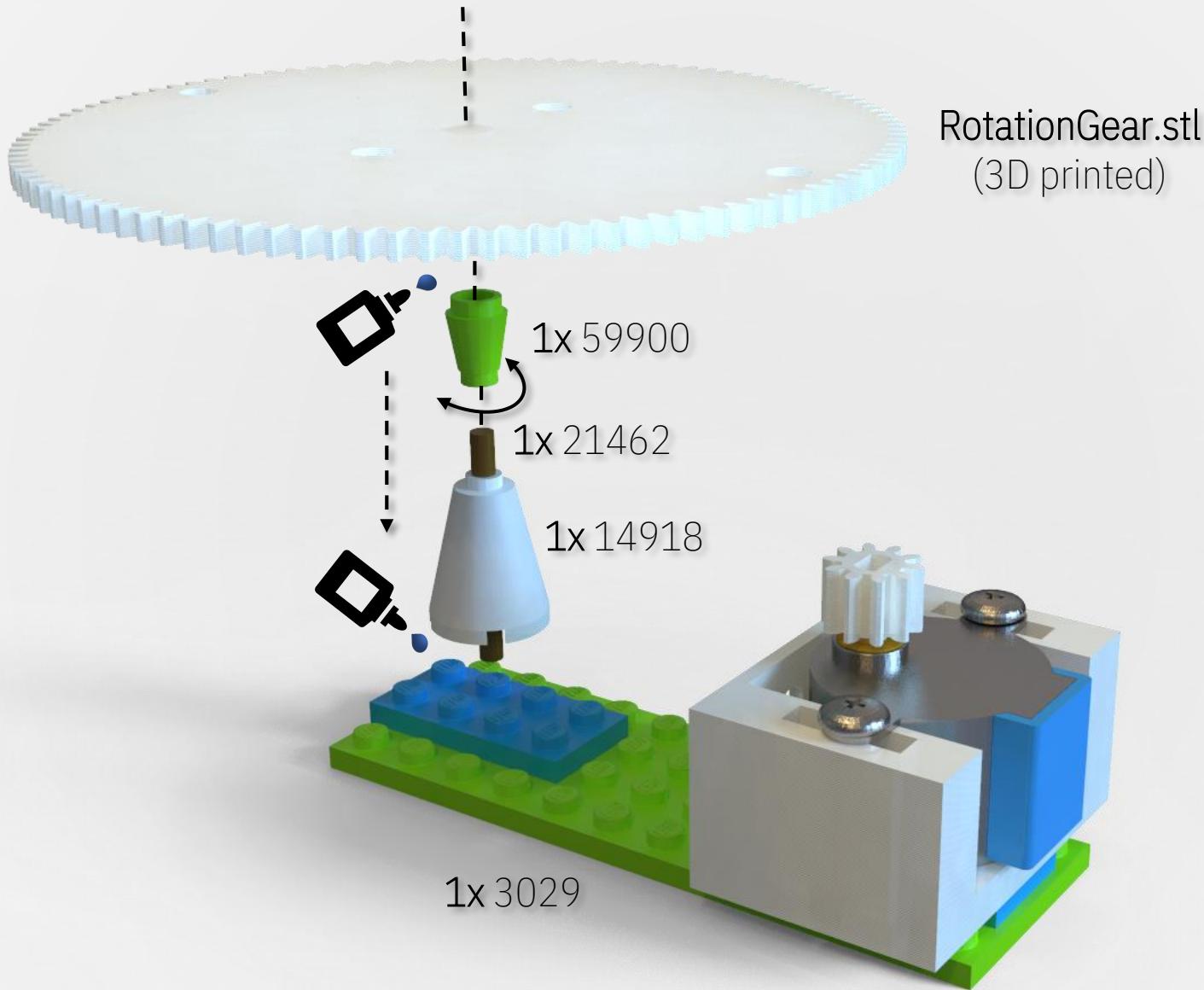
---



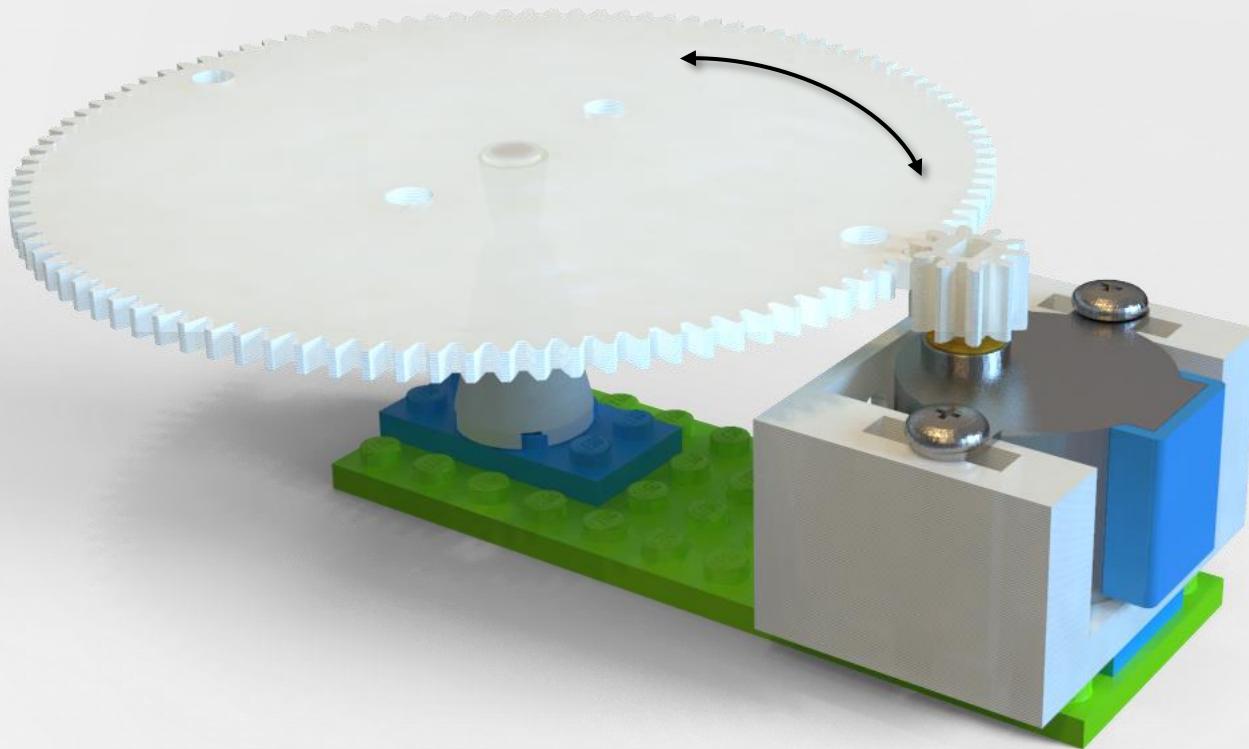
## Rotary stage – Option 1 (simple)



## Rotary stage – Option 2 (precise)

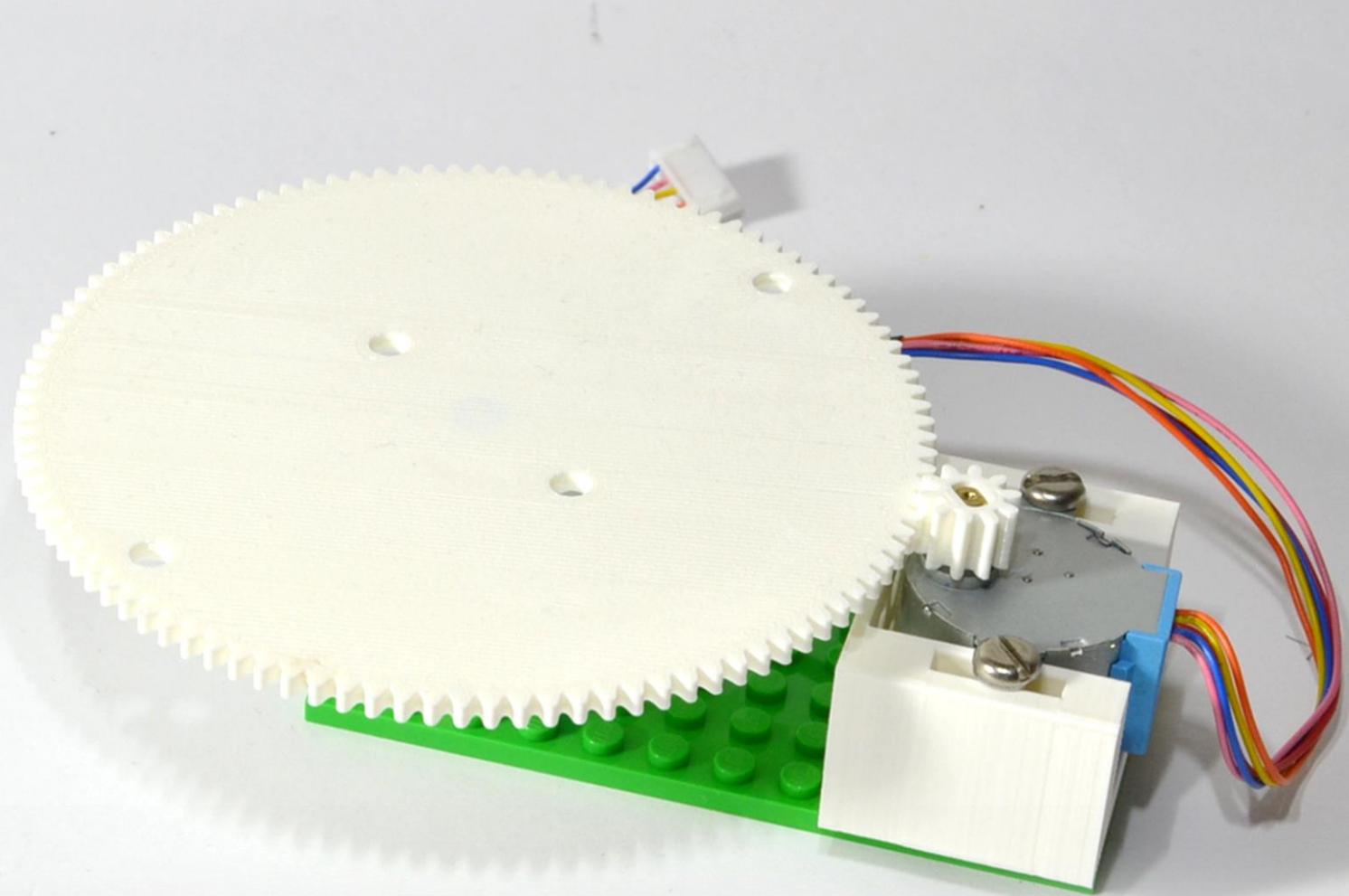


## Rotary stage – Option 2 (precise)



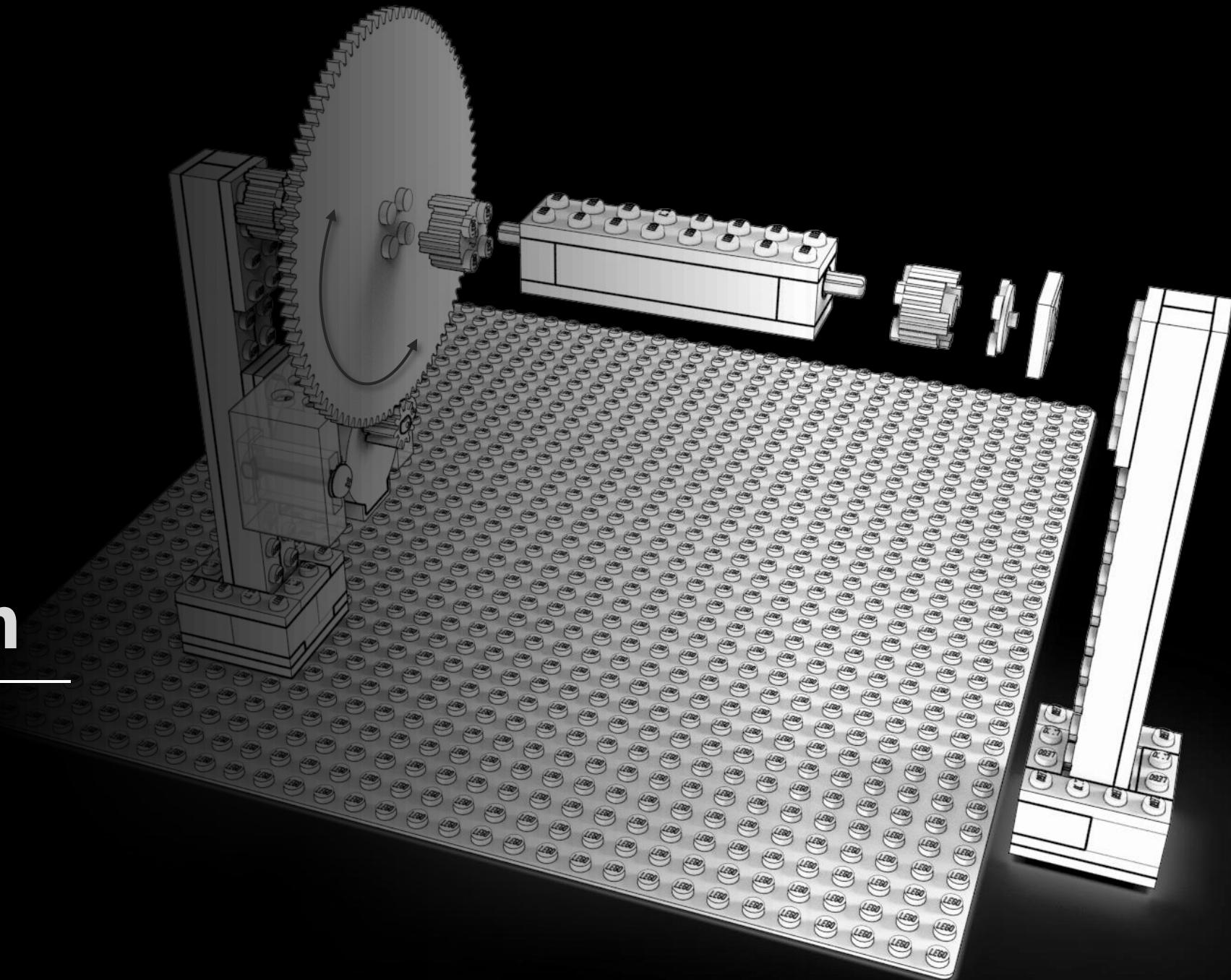
## Rotary stage – Option 2 (precise)

(Photograph)

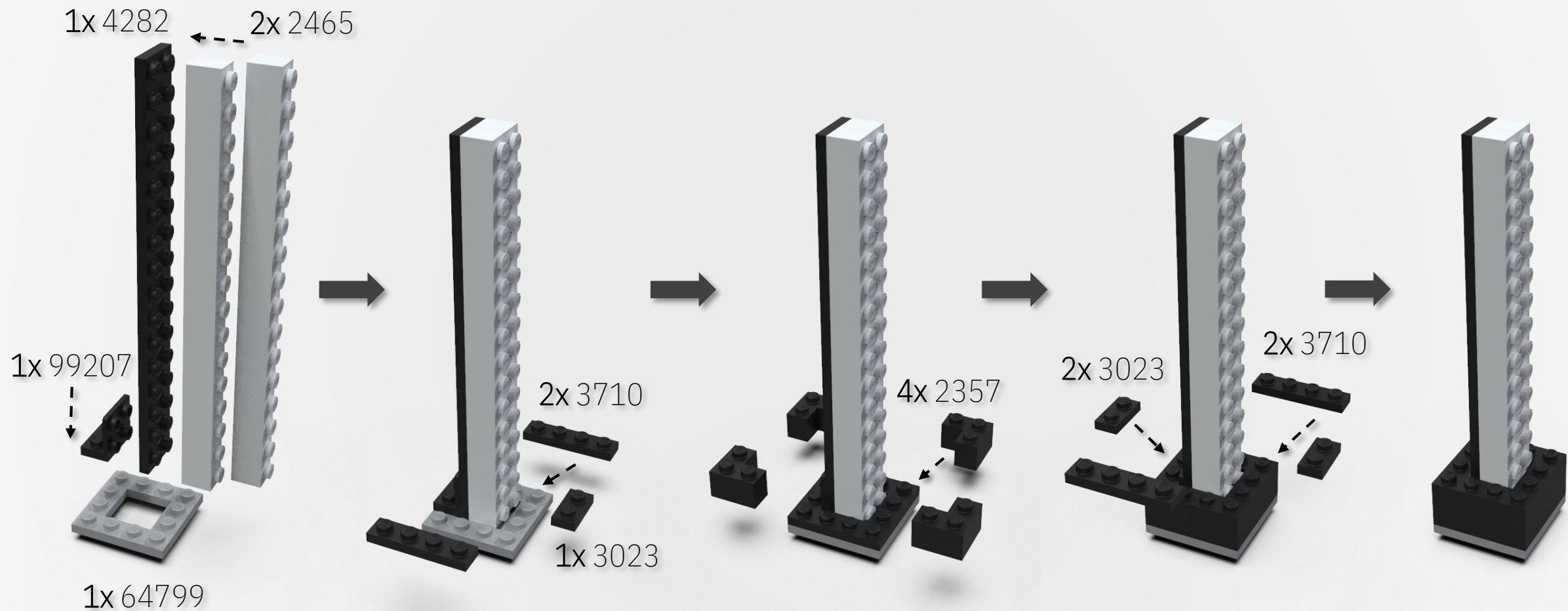


# Tilt mechanism

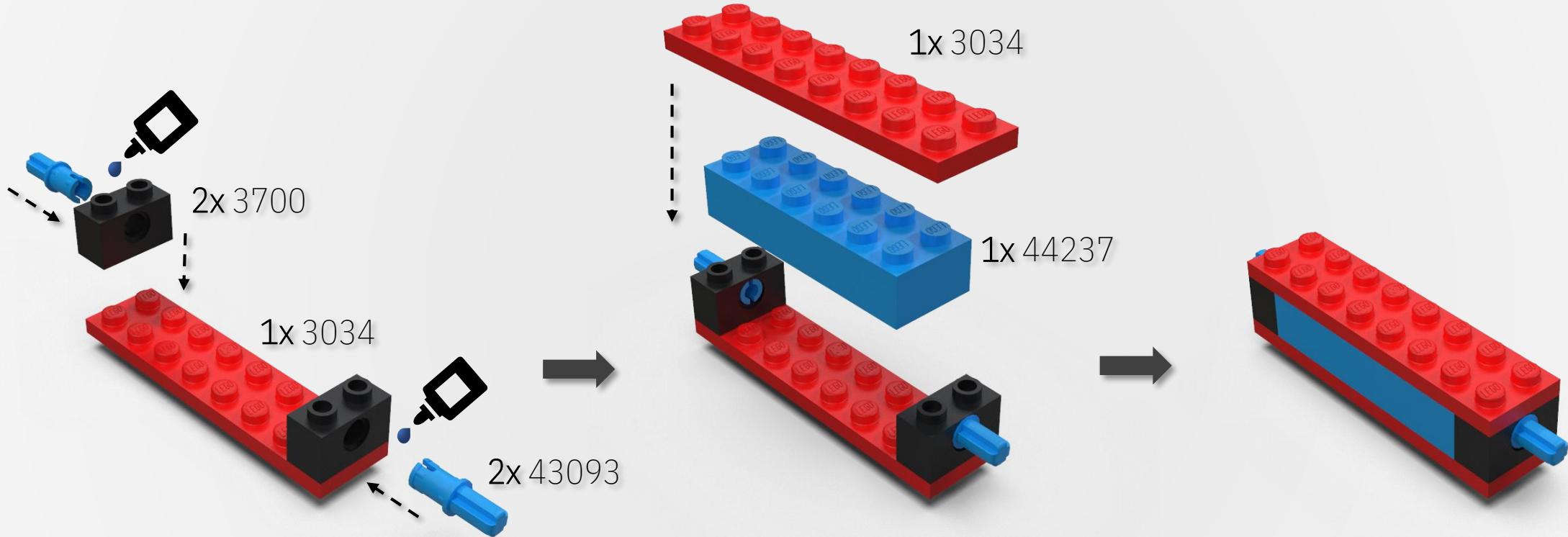
---



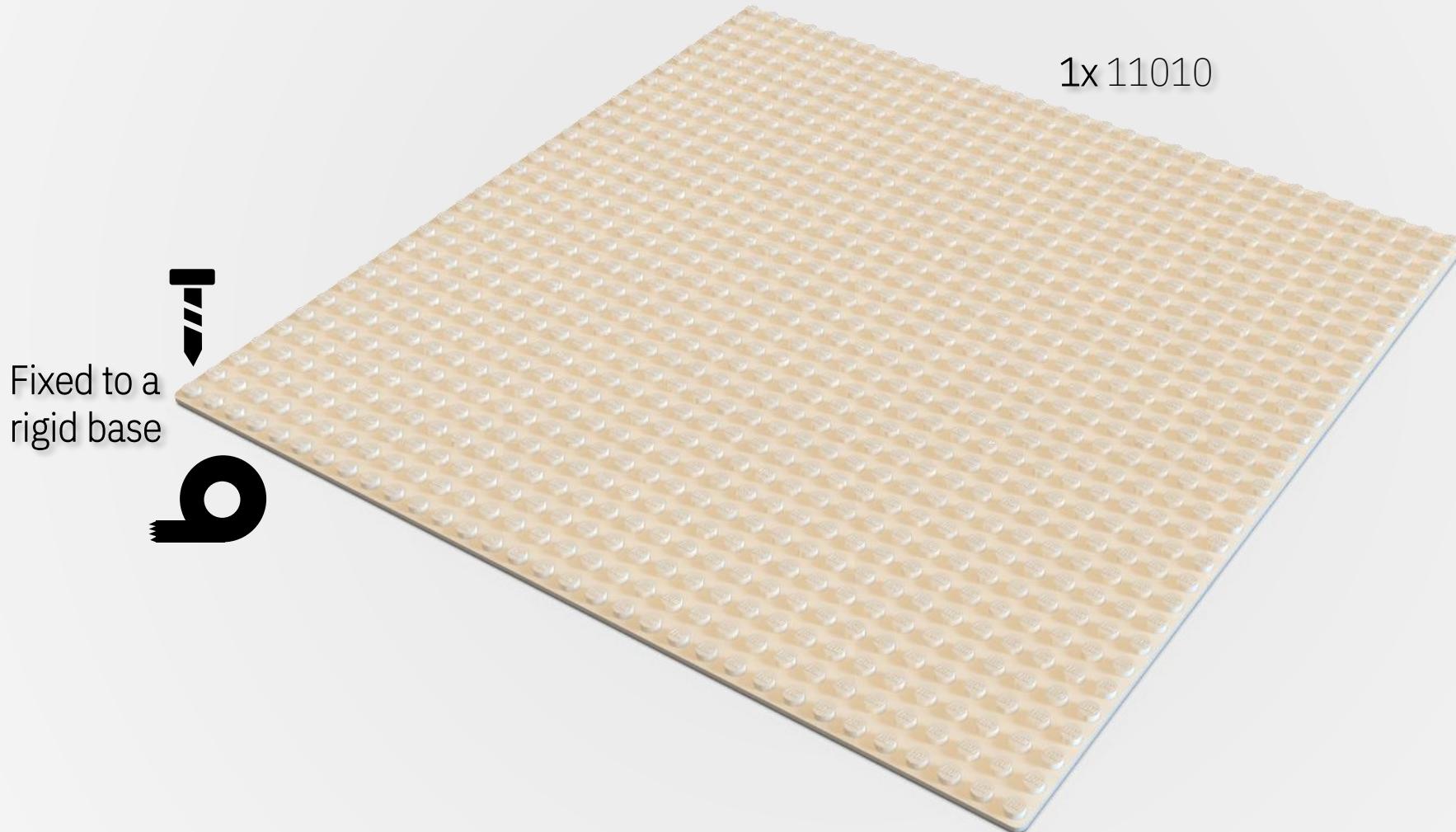
## Tilt mechanism – Vertical supports



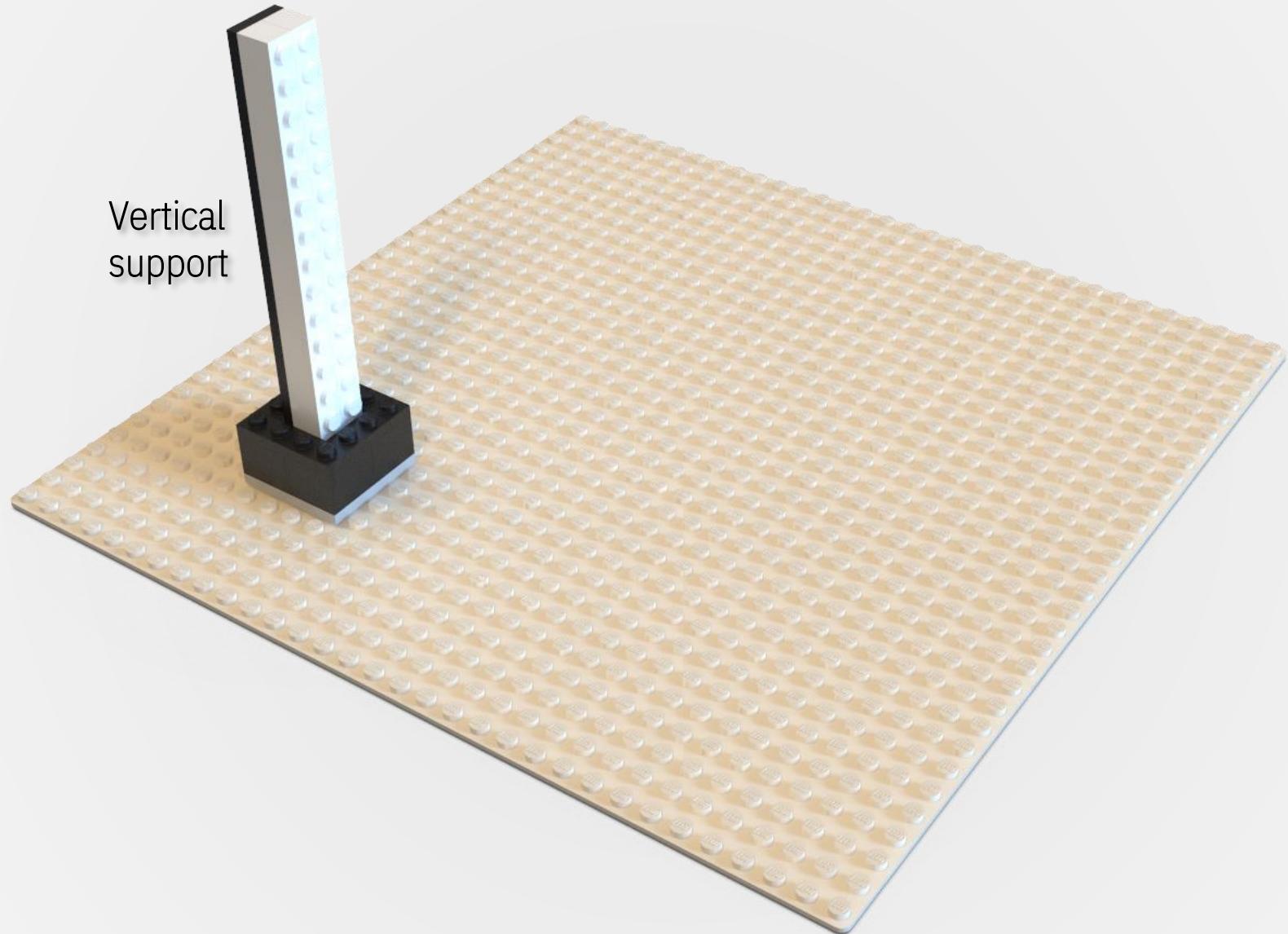
## Tilt mechanism – Camera stage



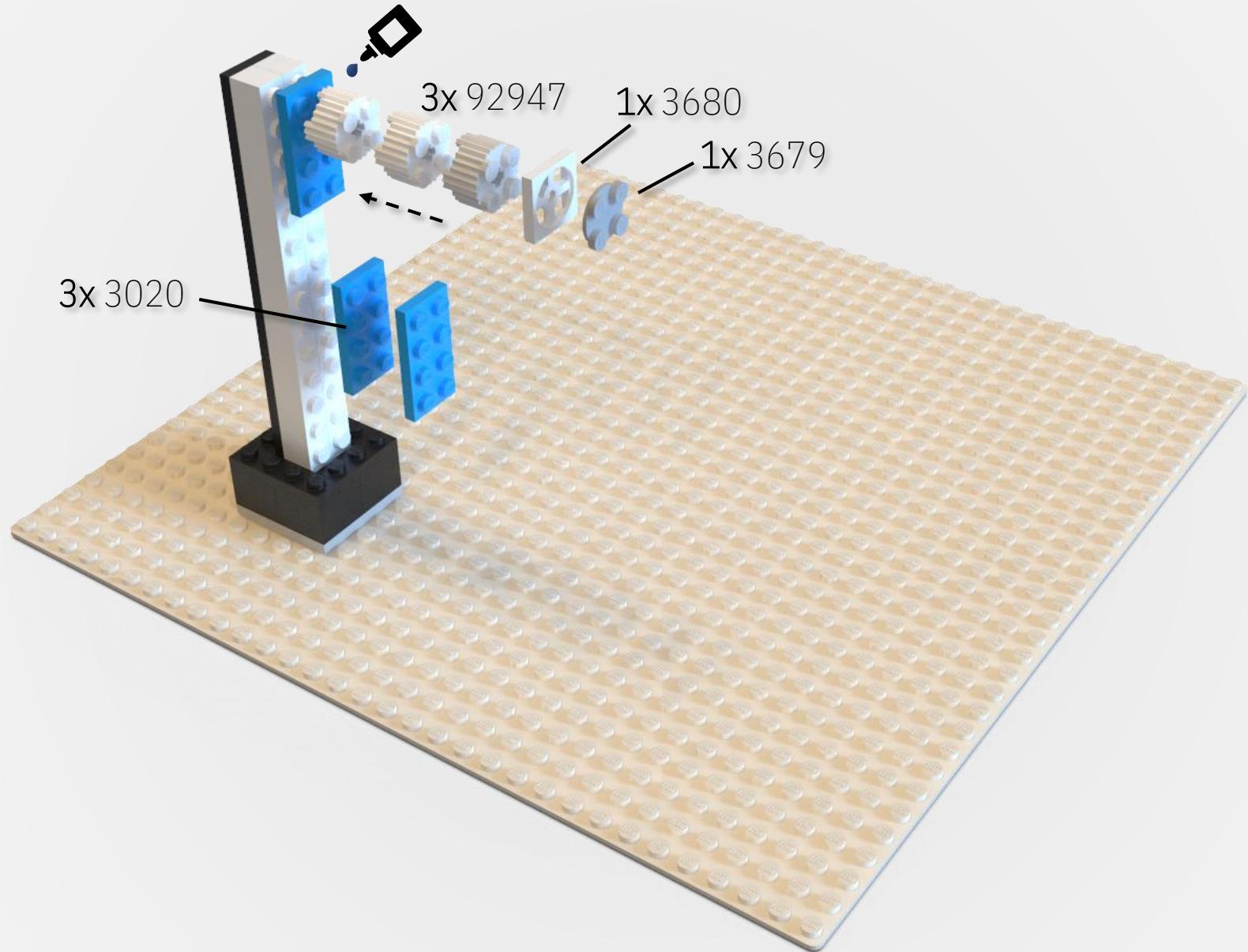
# Tilt mechanism



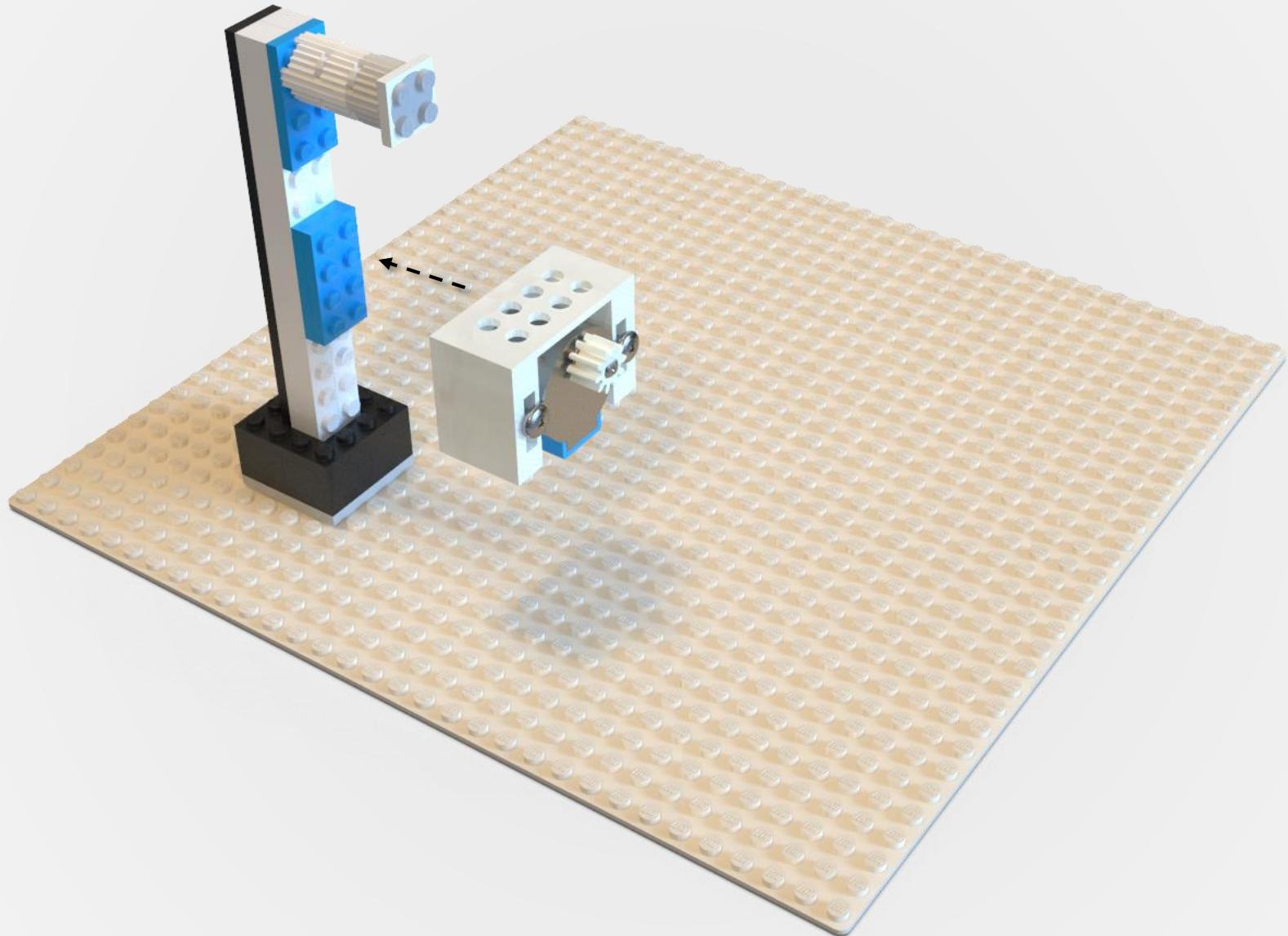
## Tilt mechanism



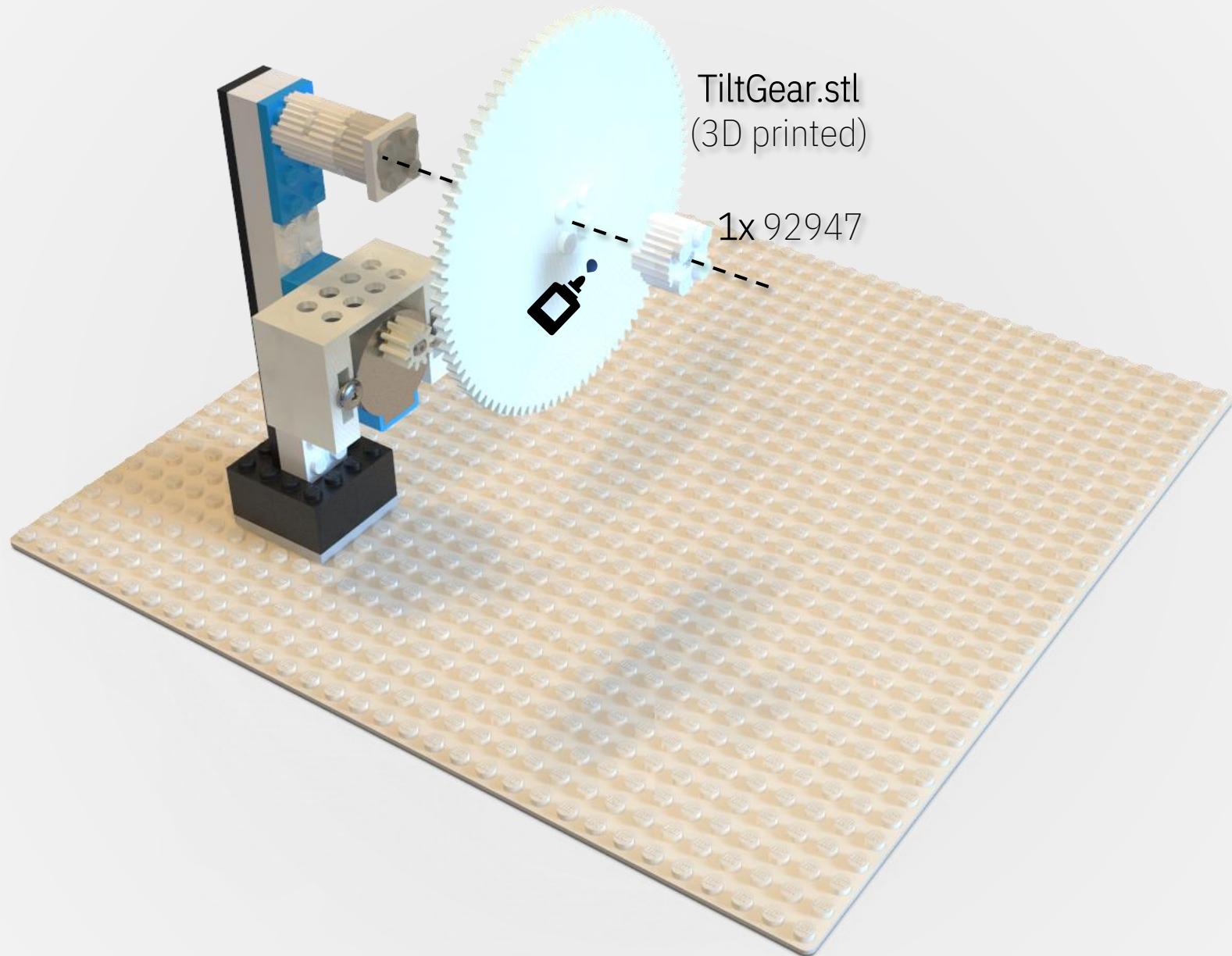
## Tilt mechanism



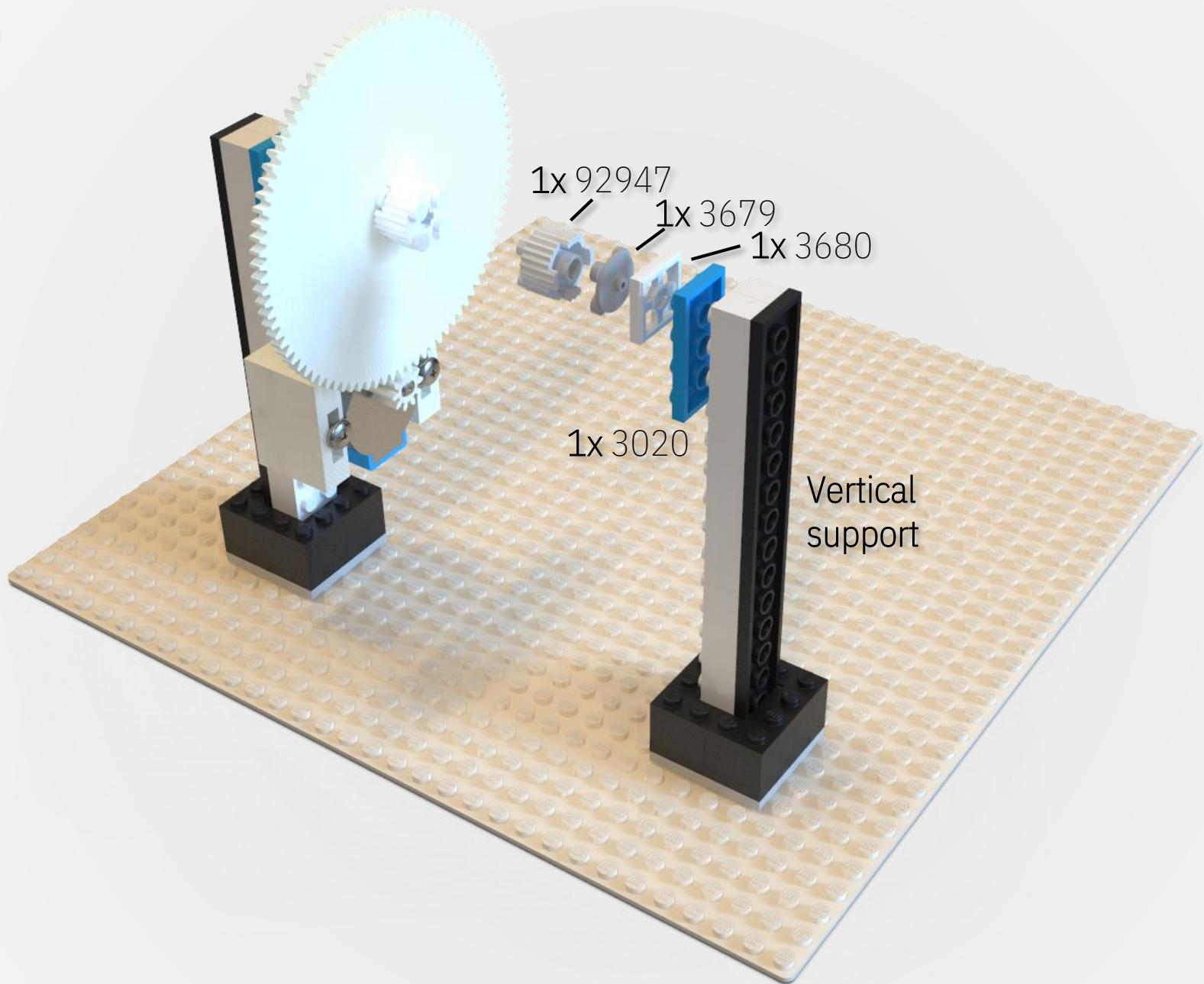
## Tilt mechanism



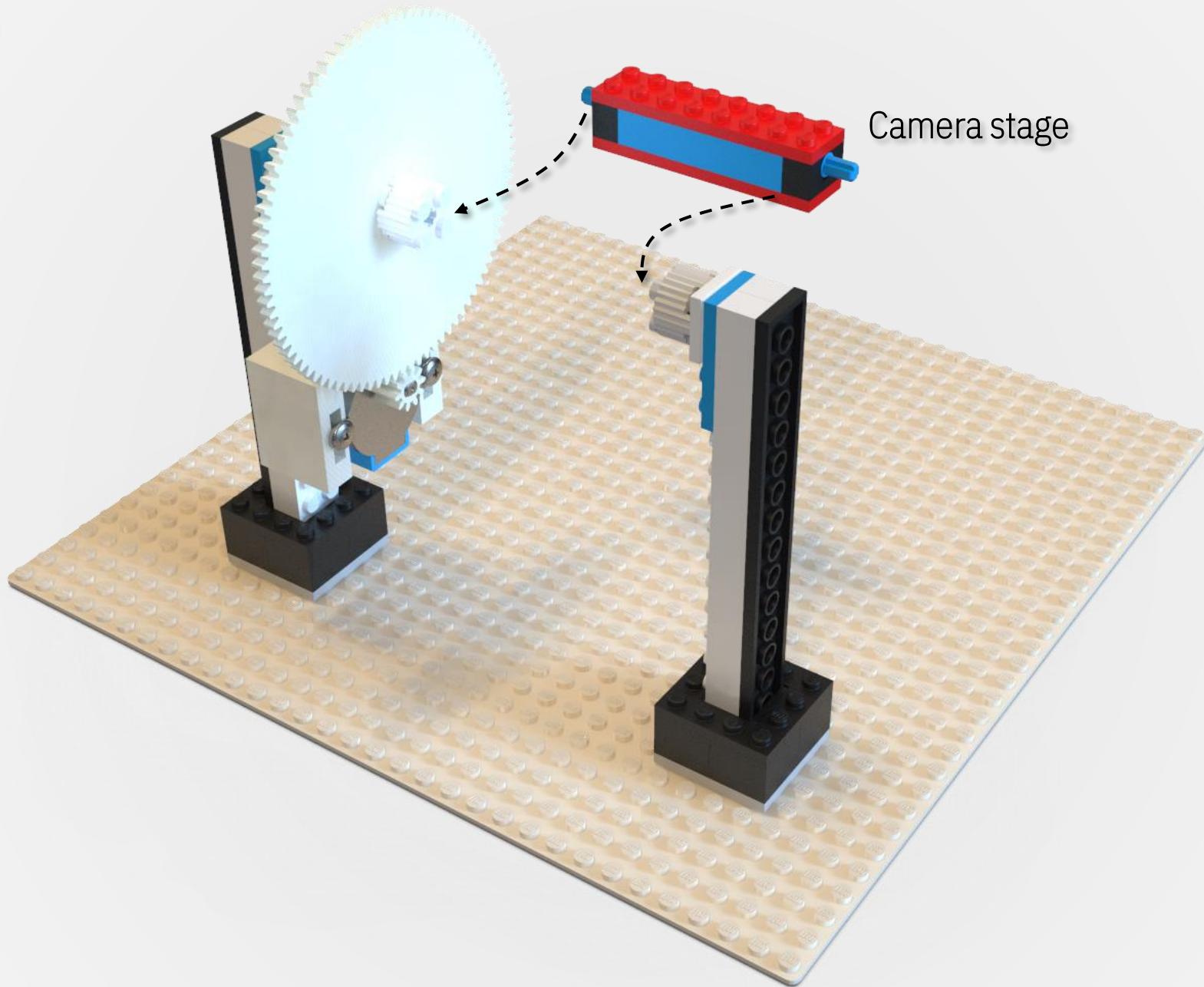
# Tilt mechanism



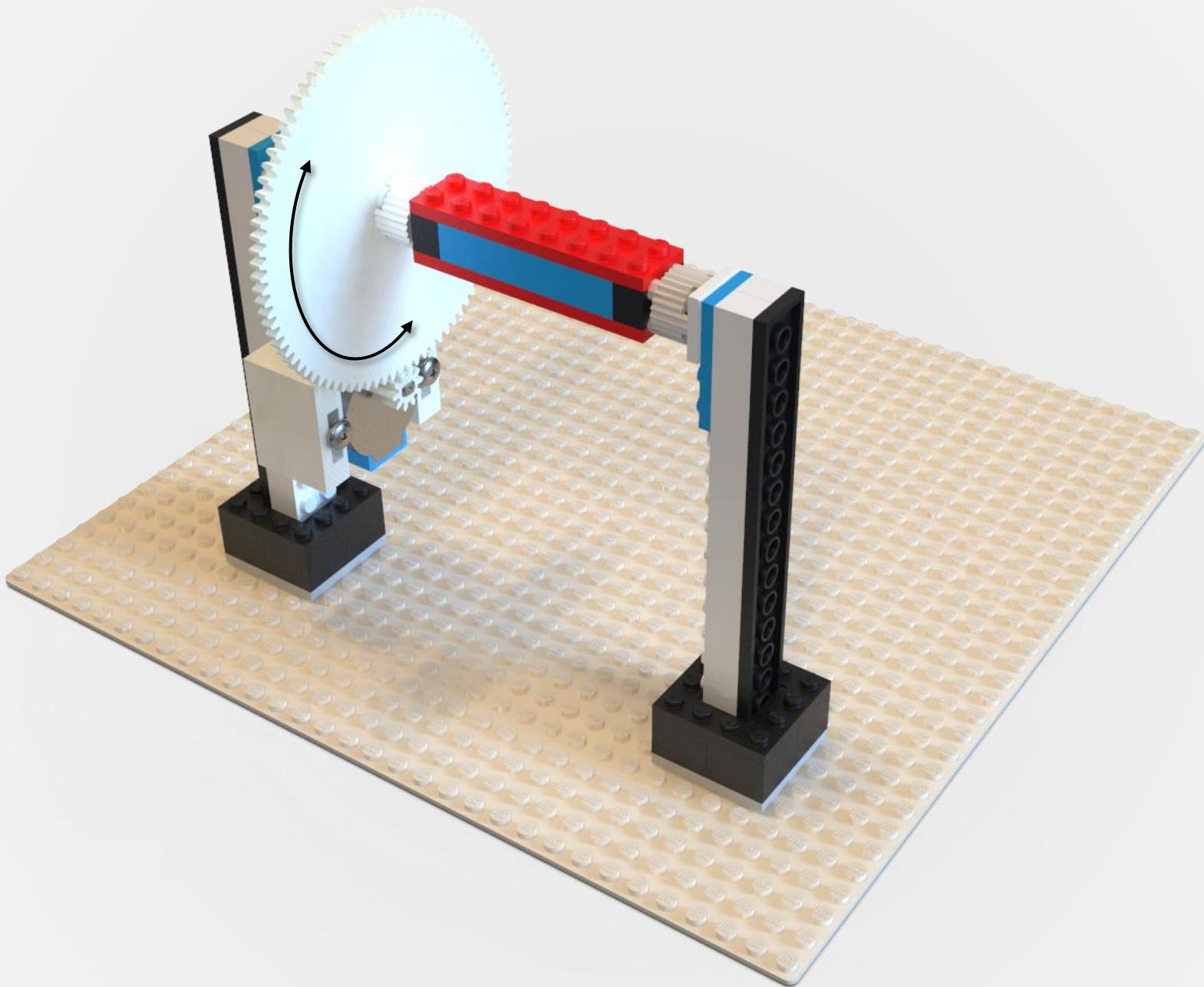
## Tilt mechanism



## Tilt mechanism

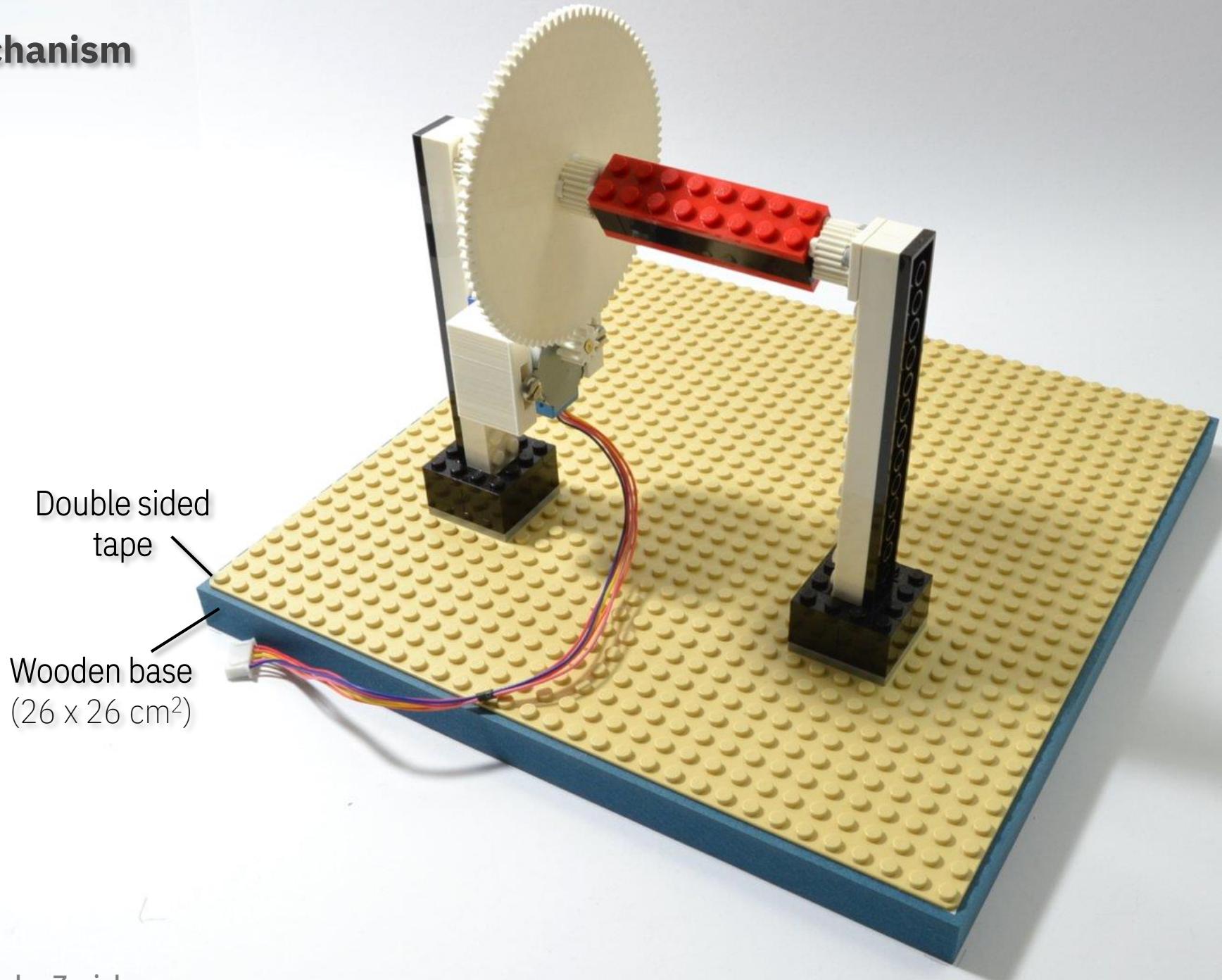


## Tilt mechanism



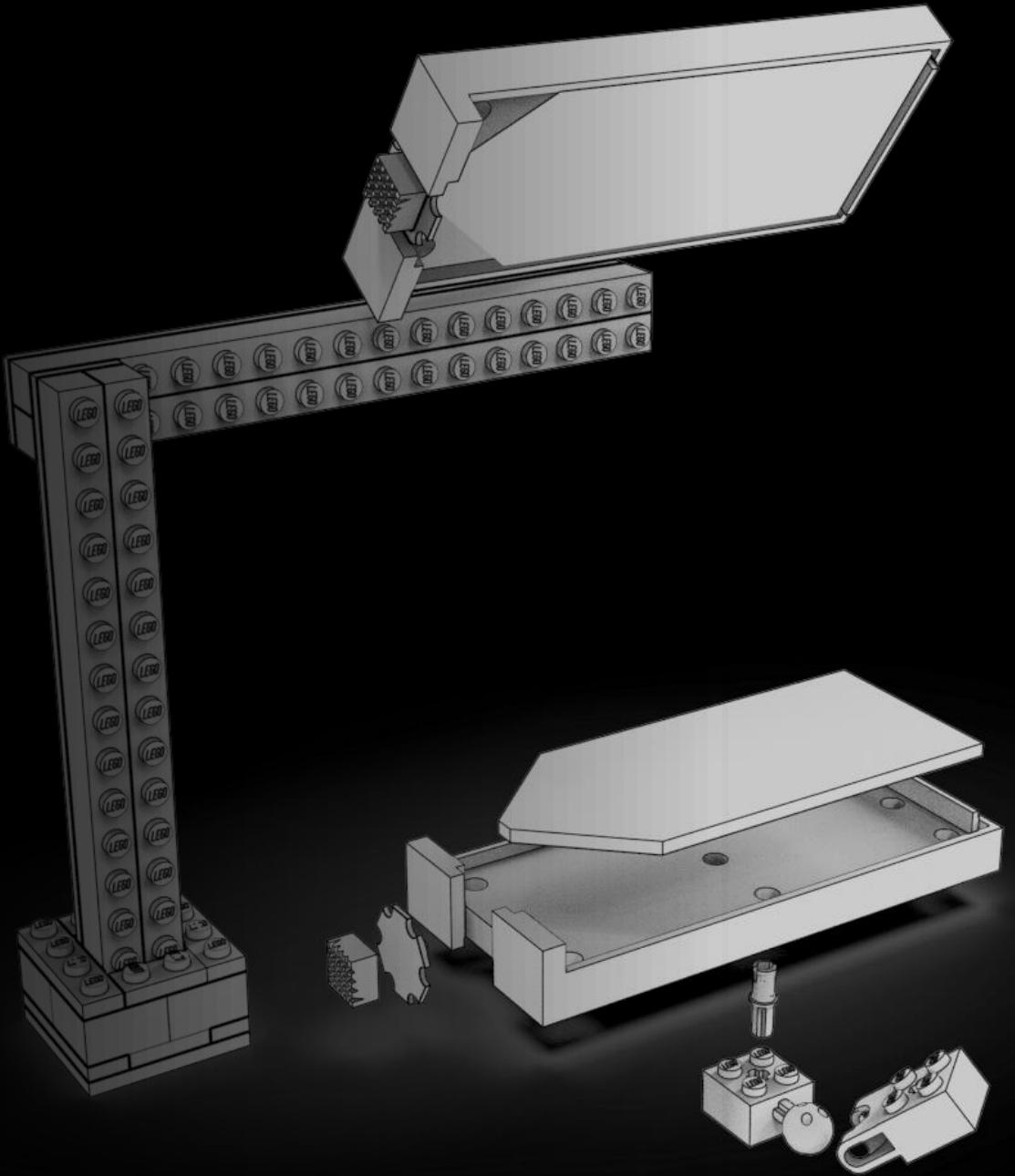
# Tilt mechanism

(Photograph)

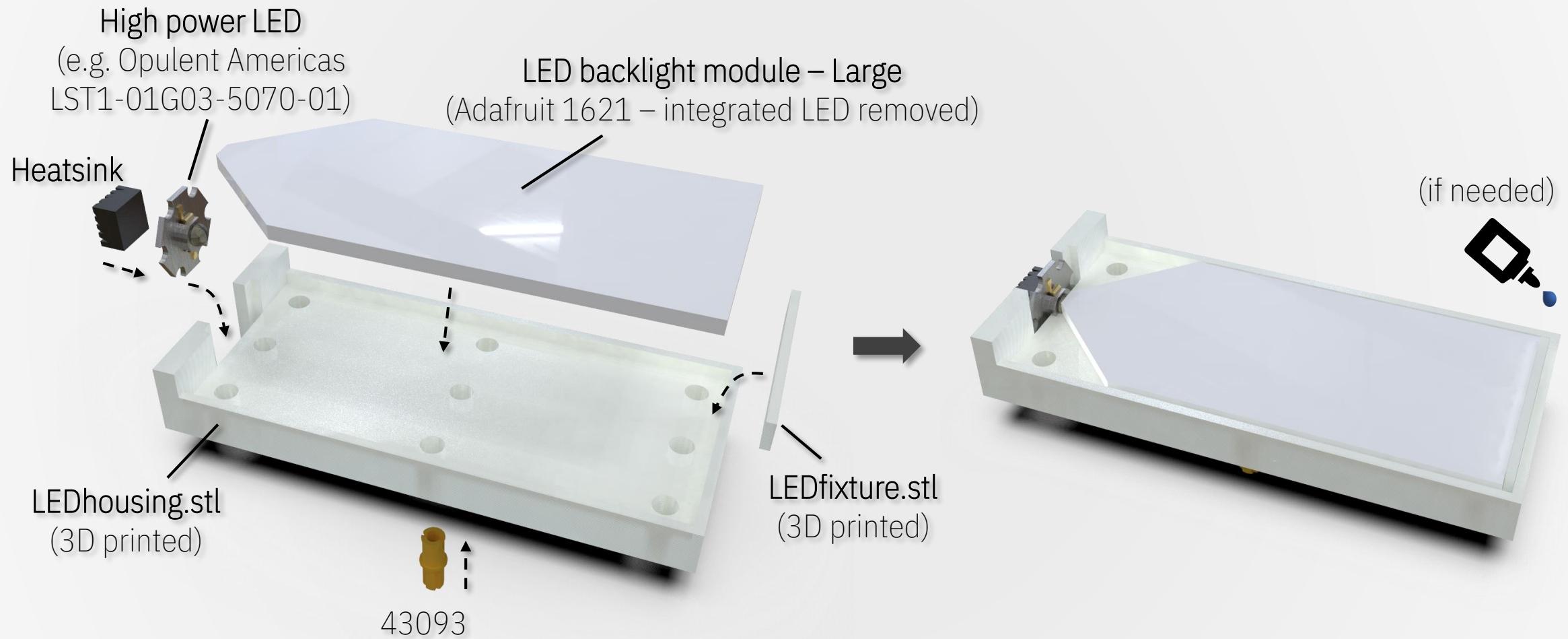


# Illumination

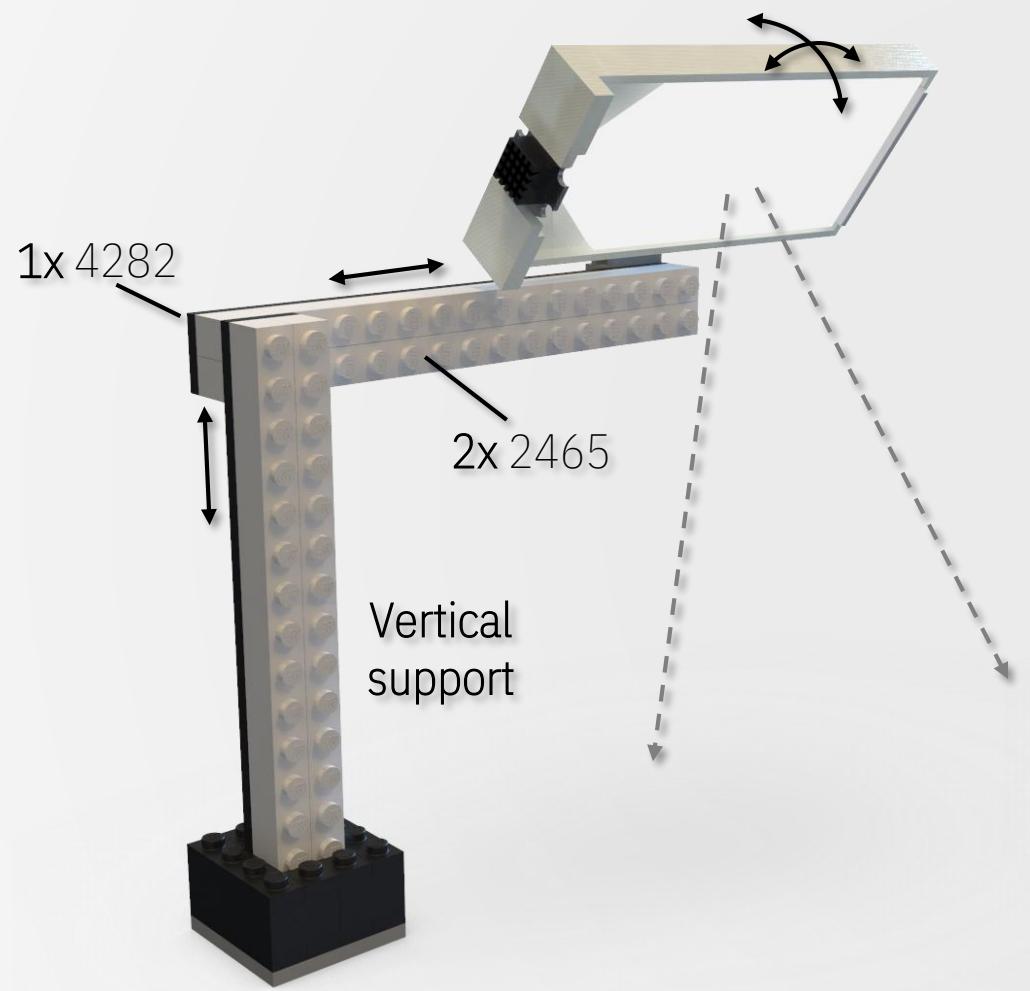
---



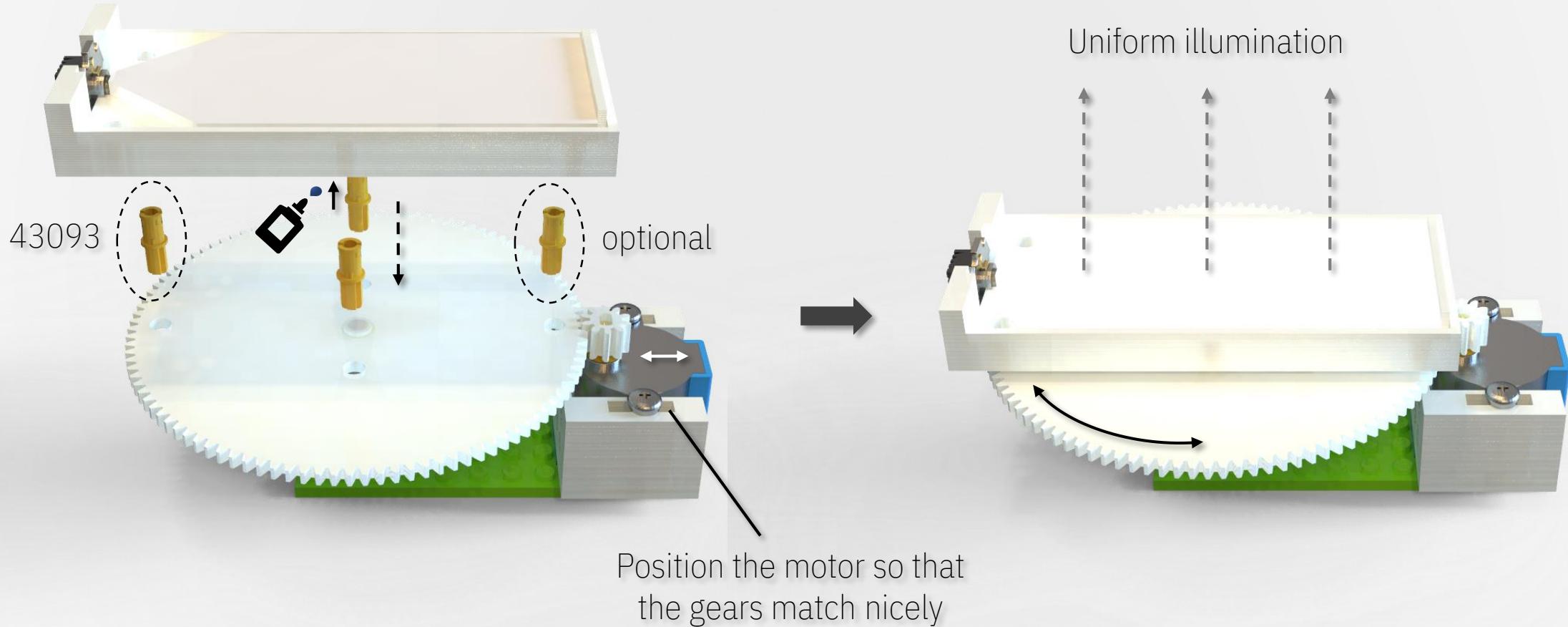
# Illumination – LED diffuser



## Illumination – Reflected light

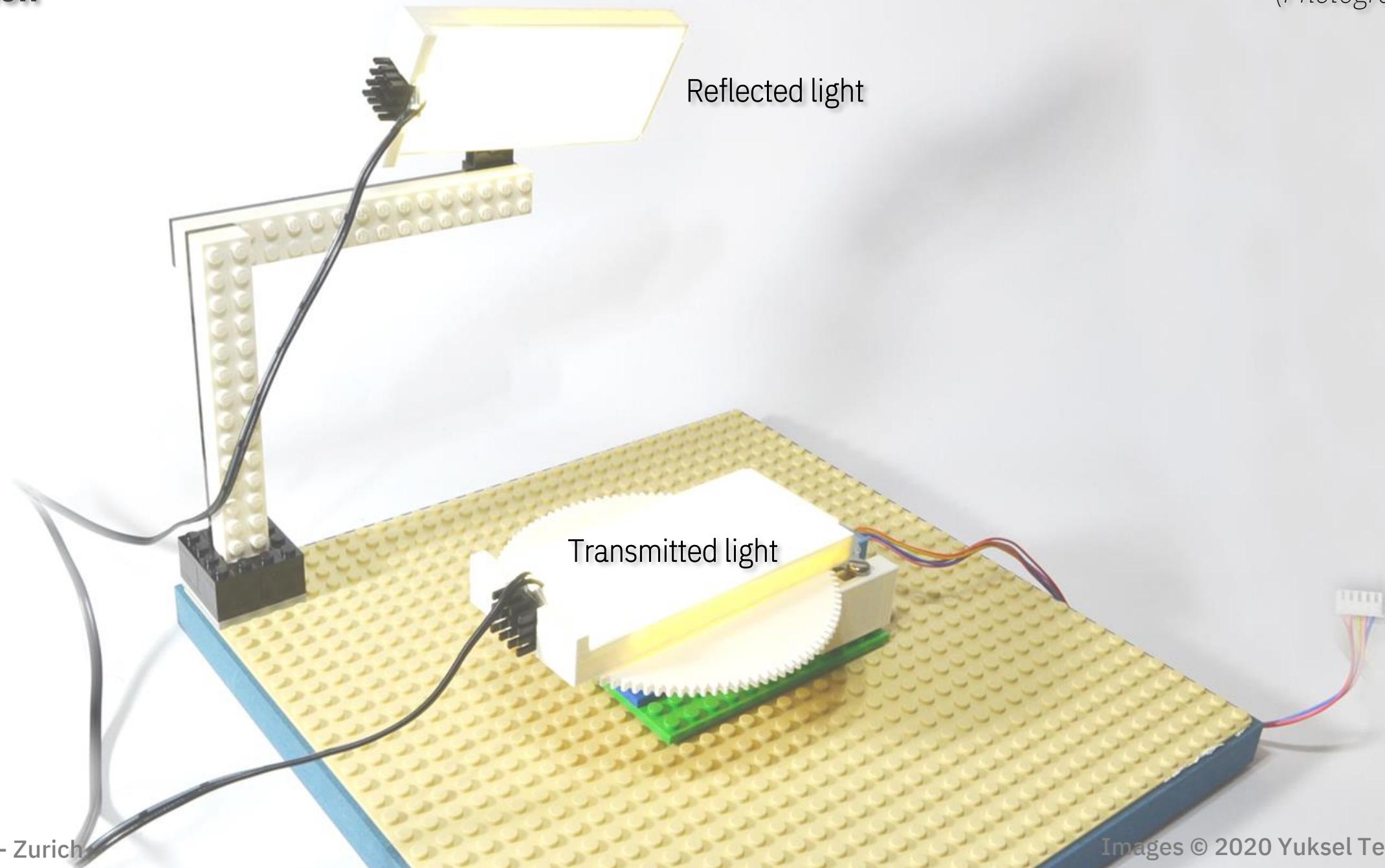


## Illumination – Transmitted light



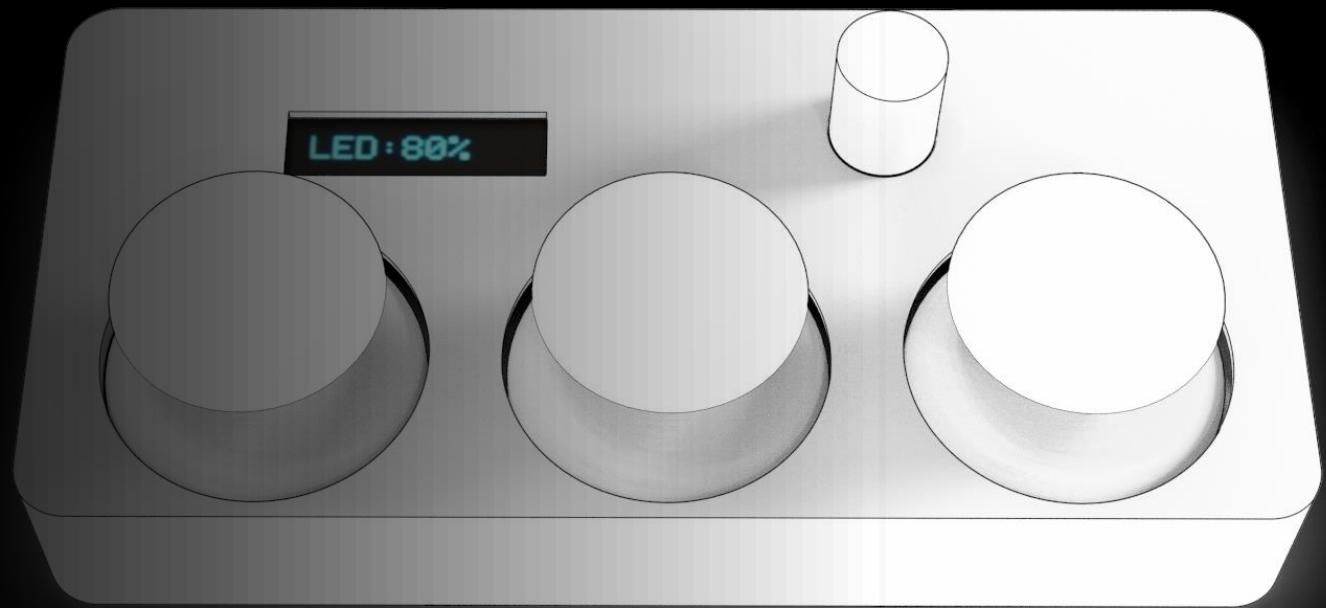
# Illumination

(Photograph)

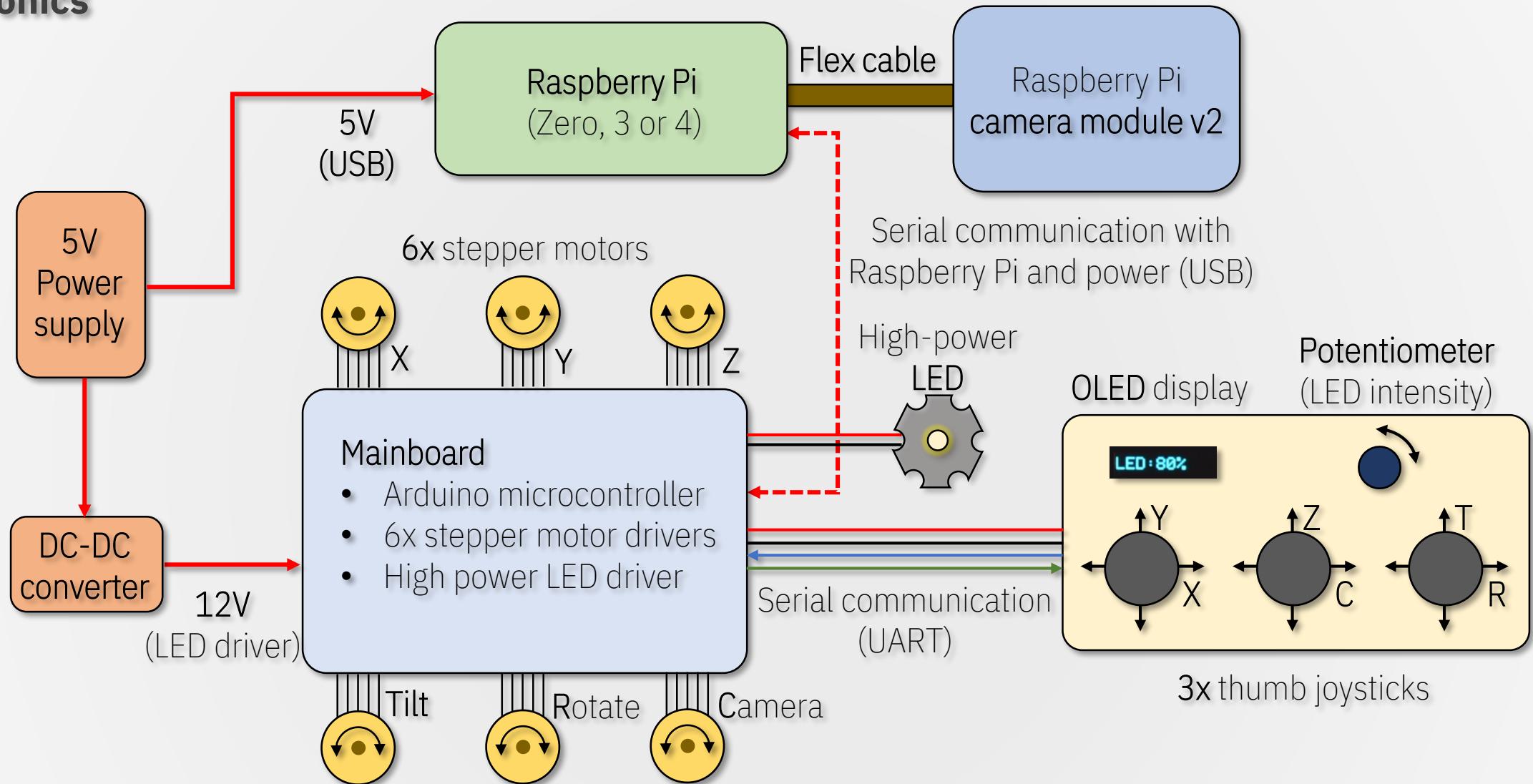


# Electronics

---



# Electronics

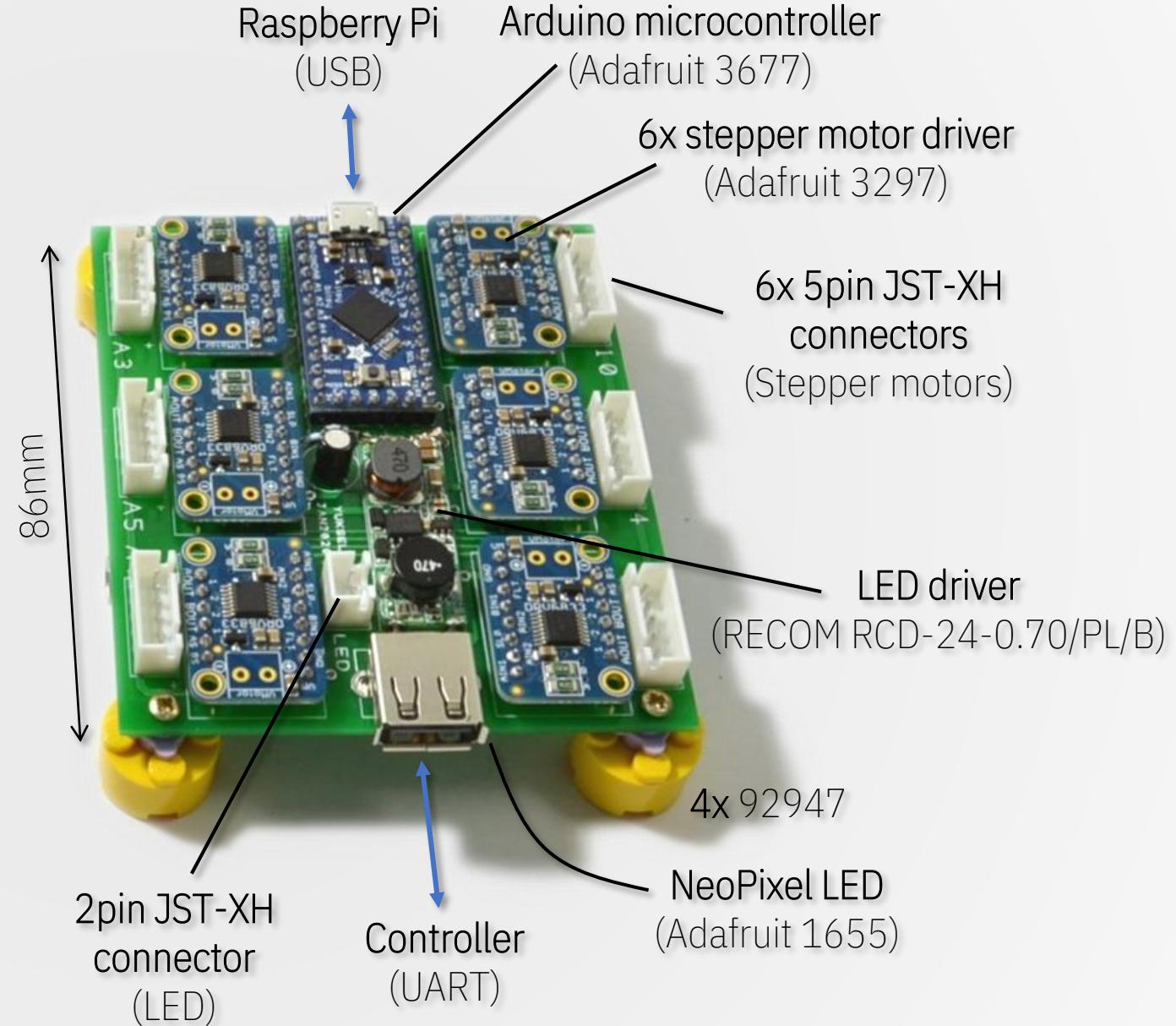
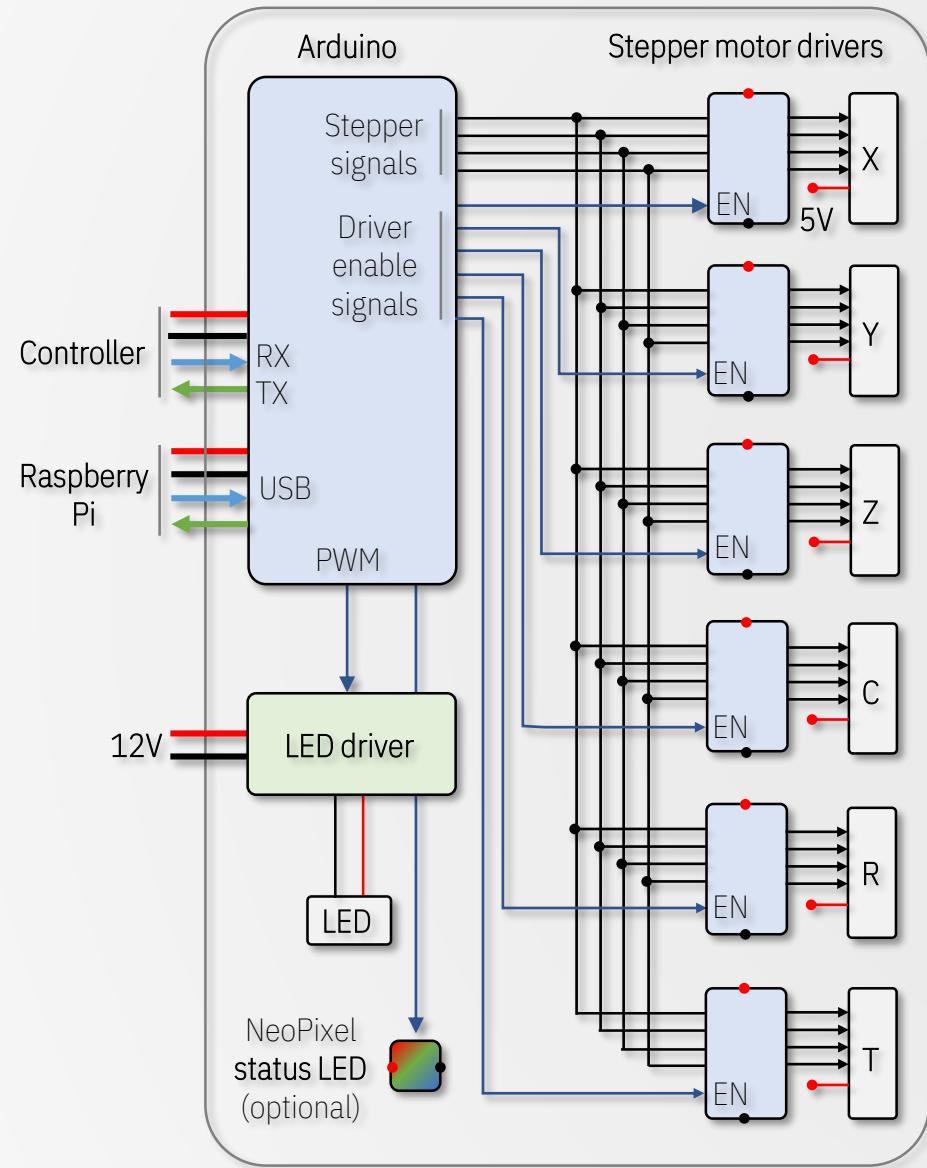


**Note-1:** Stepper motors and the illumination LED can be controlled from the controller (UART) or the Raspberry Pi (USB) or both

**Note-2:** Some high-power LED drivers may require more than 5V, then a DC-DC converter or a separate power supply can be used

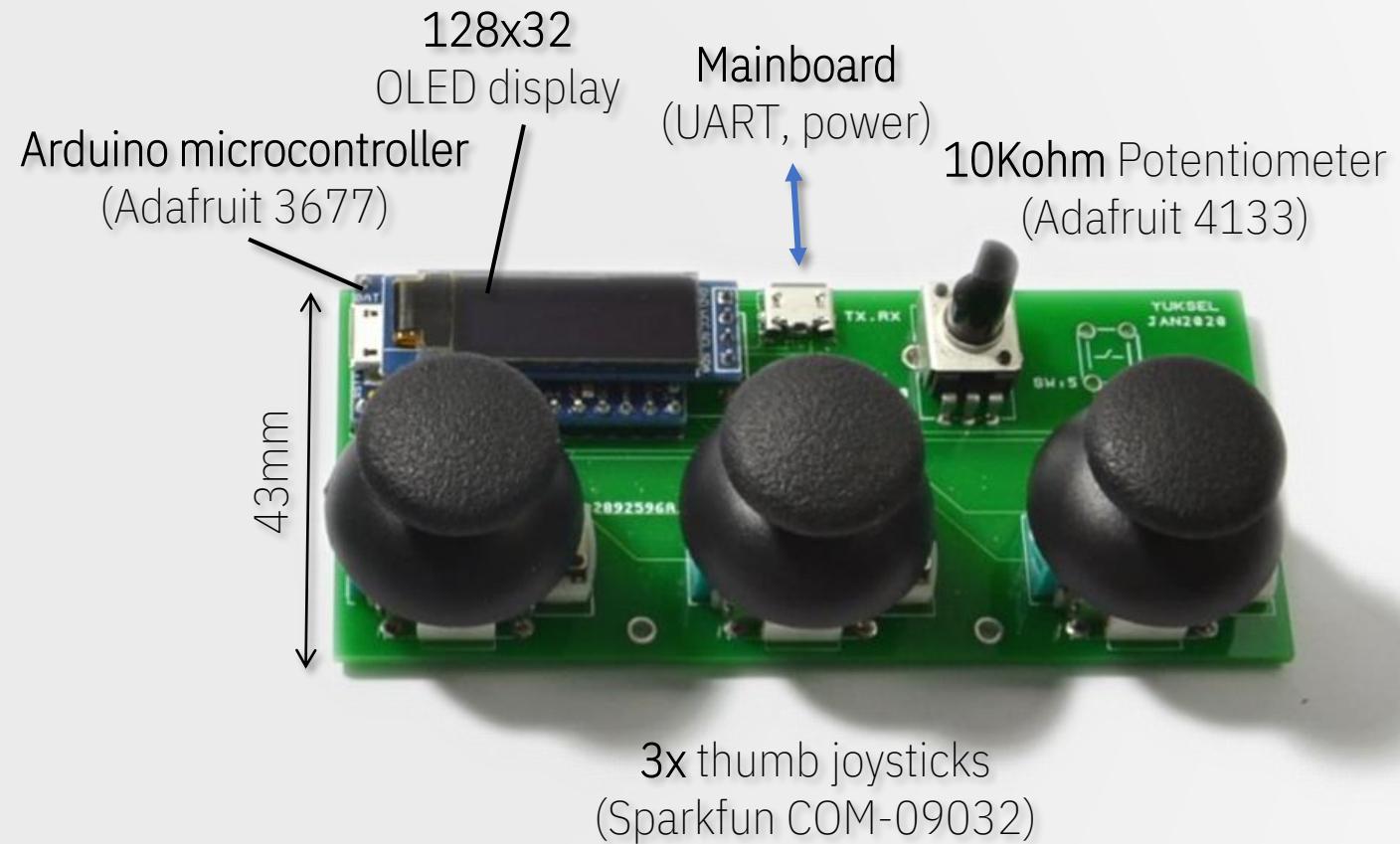
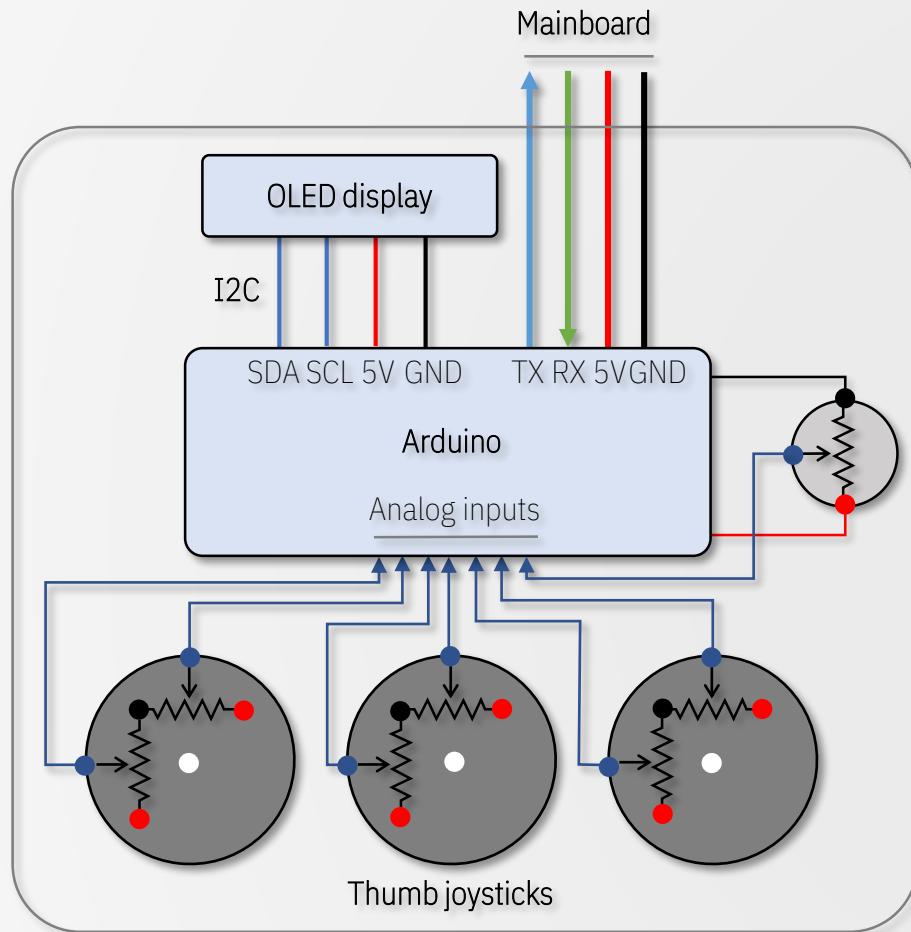
# Electronics – Mainboard

(Photograph)



# Electronics – Controller

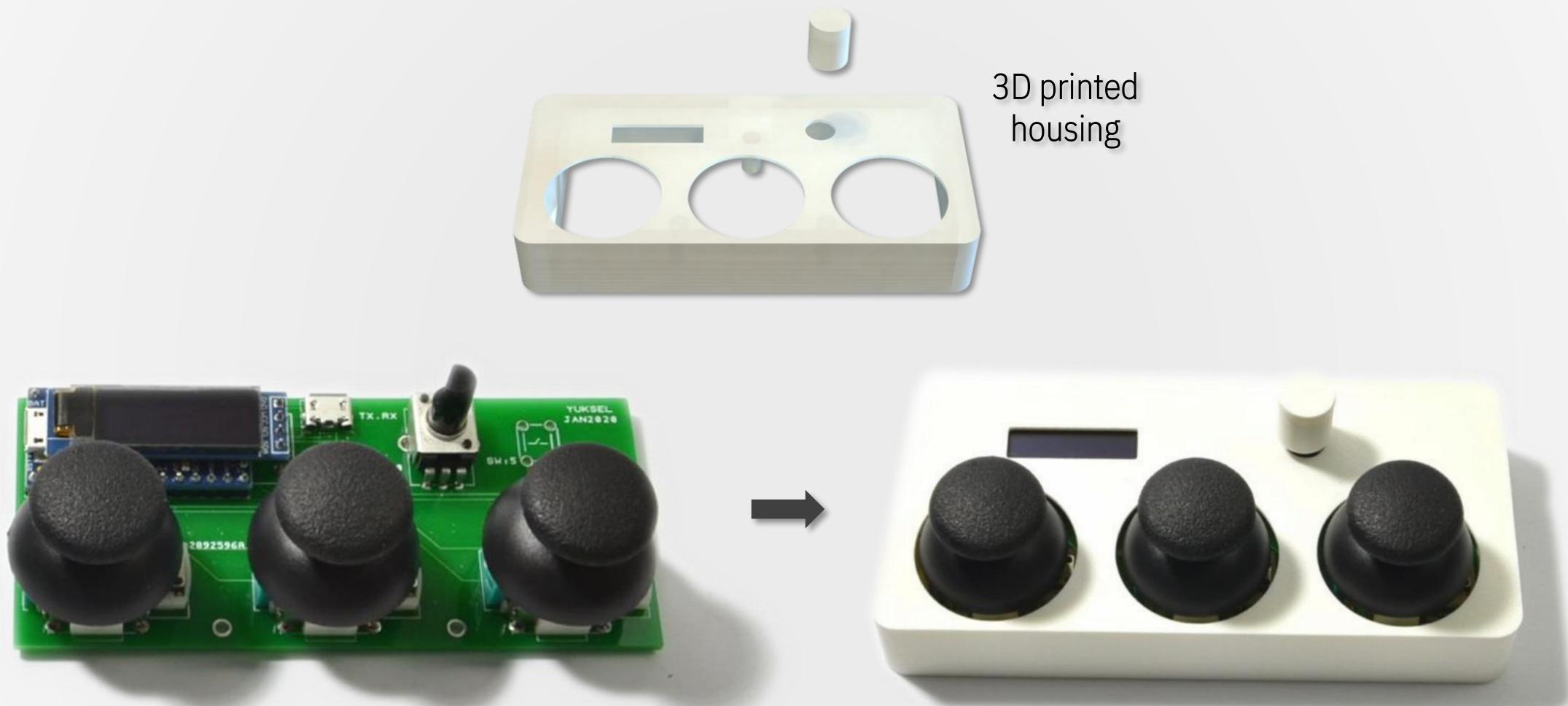
(Photograph)



*This joystick controller is fun to use but it is optional, the microscope can also be controlled directly from a keyboard connected to Raspberry Pi using the Python code.*

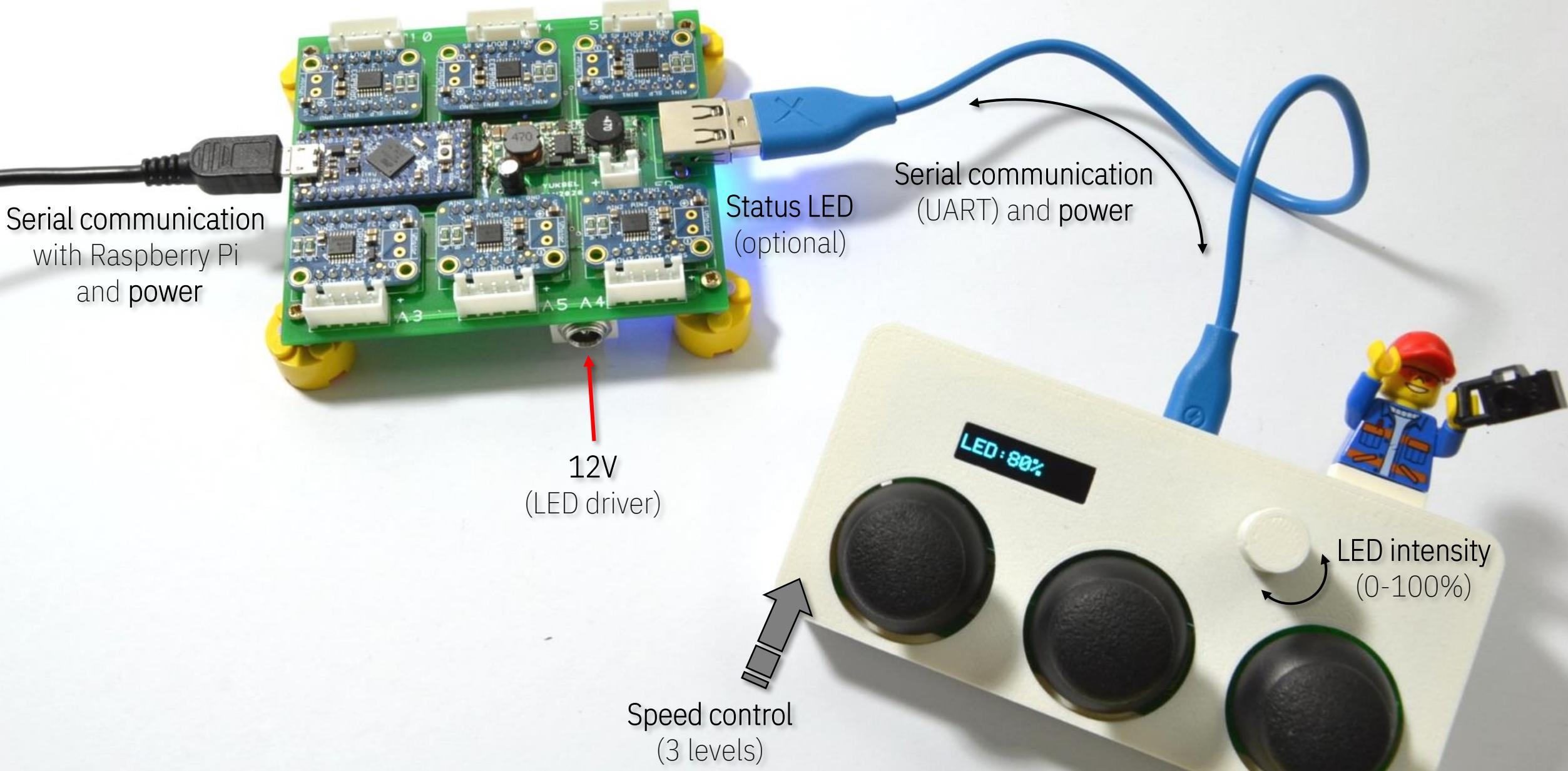
# Electronics – Controller

(Photograph)



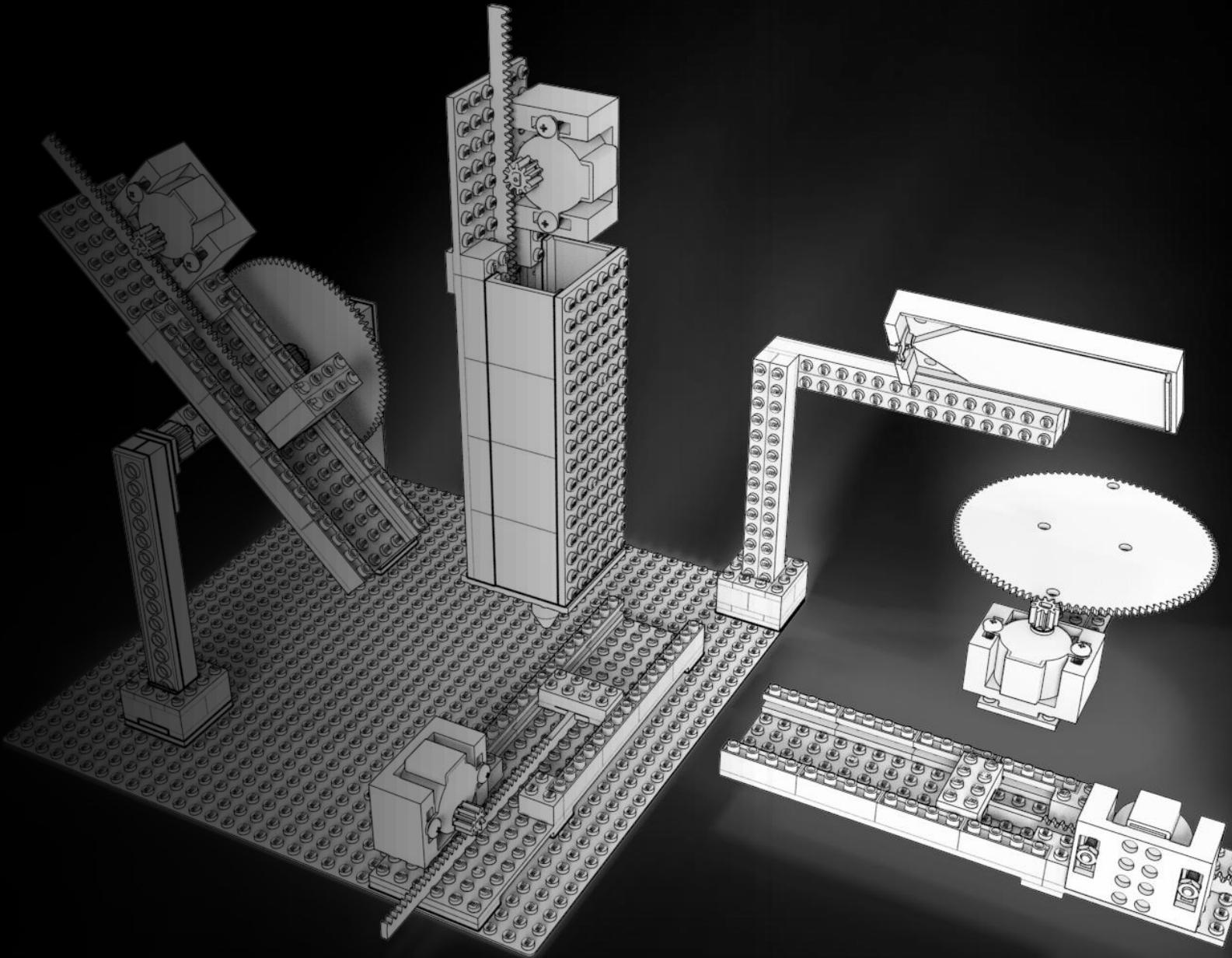
# Electronics – Operation

(Photograph)



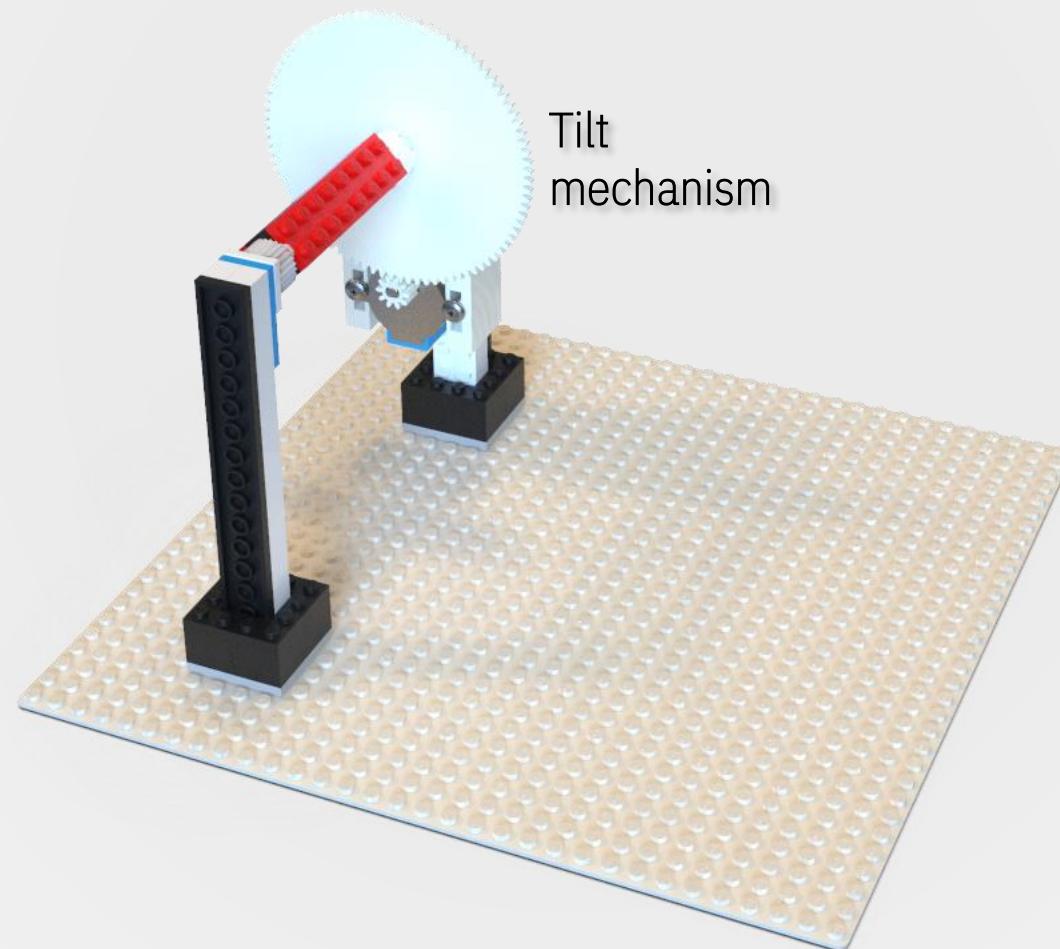
# Final assembly

---

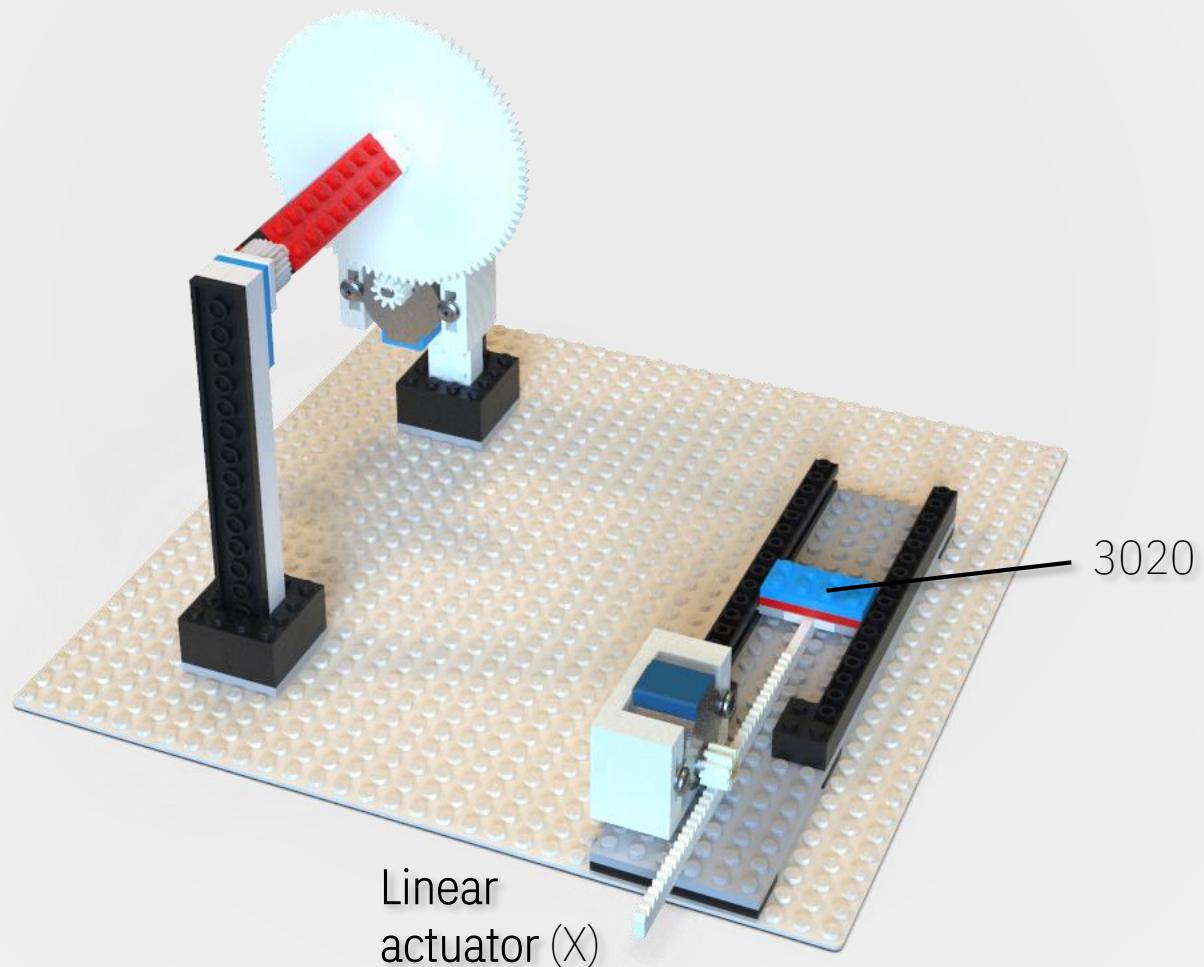


## Final assembly

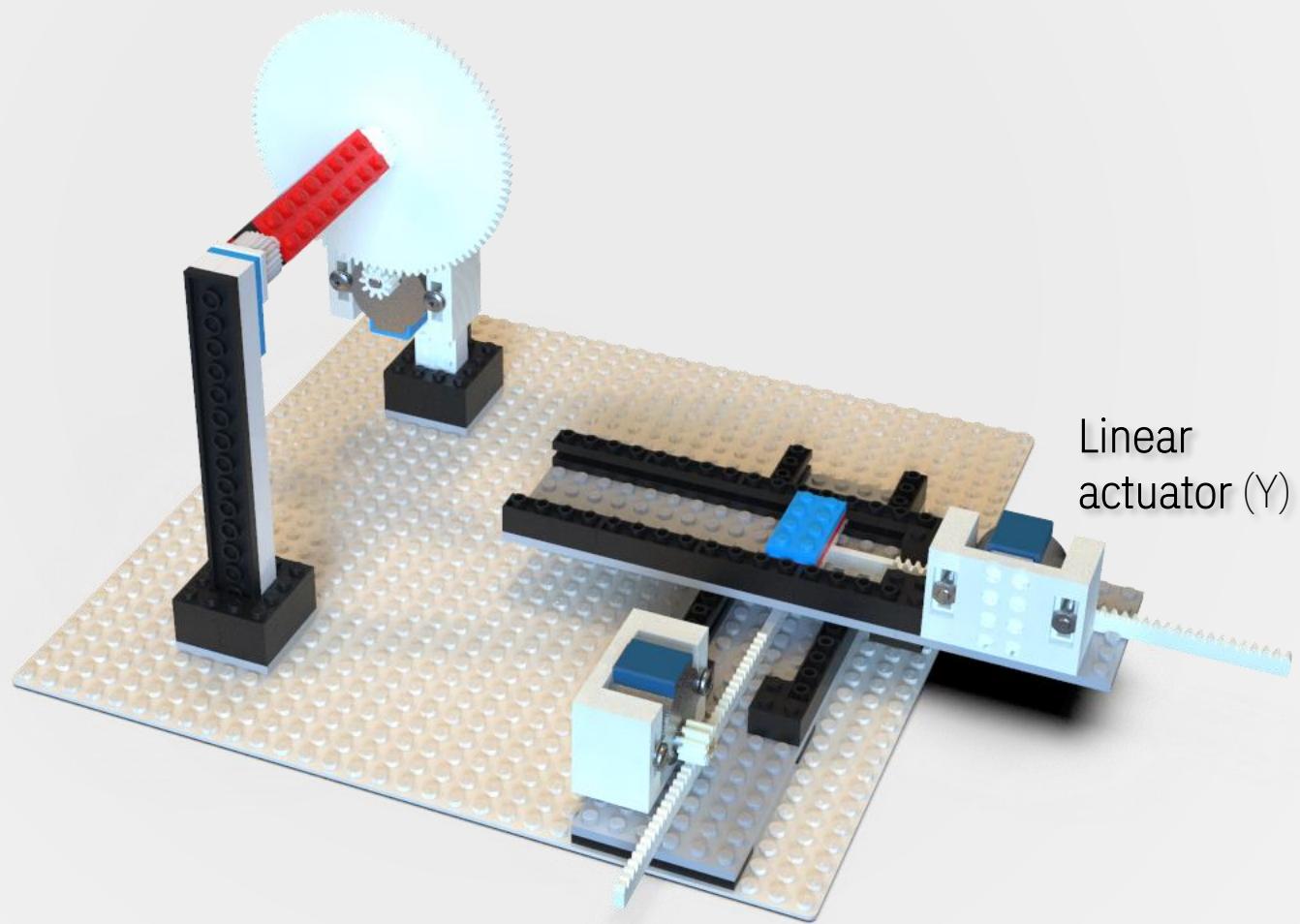
There are several different ways of assembling the microscope. Here is one. Modules can be placed differently depending on the length of the cables, size of the object to be visualized and the microscopy technique.



# Final assembly

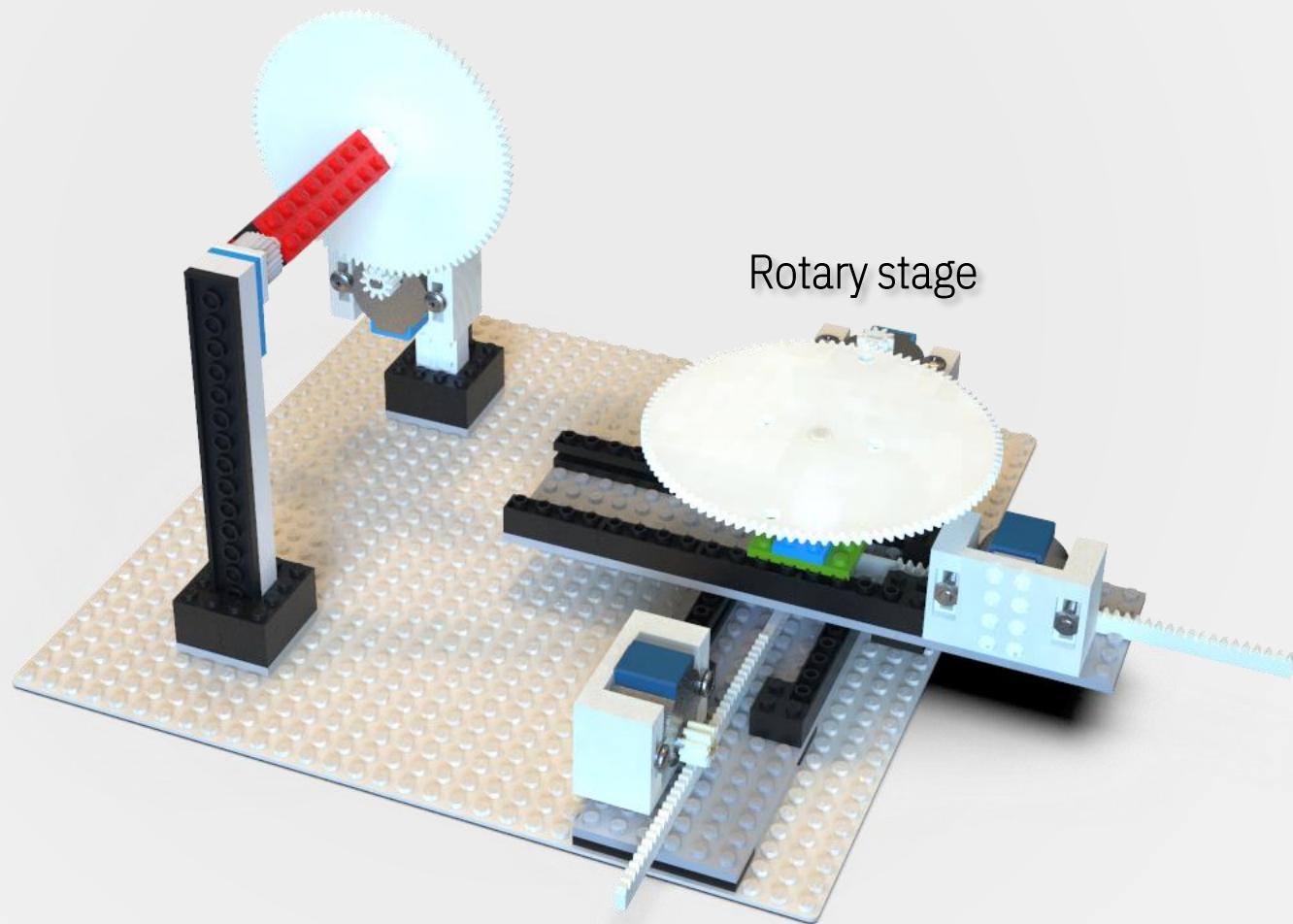


# Final assembly

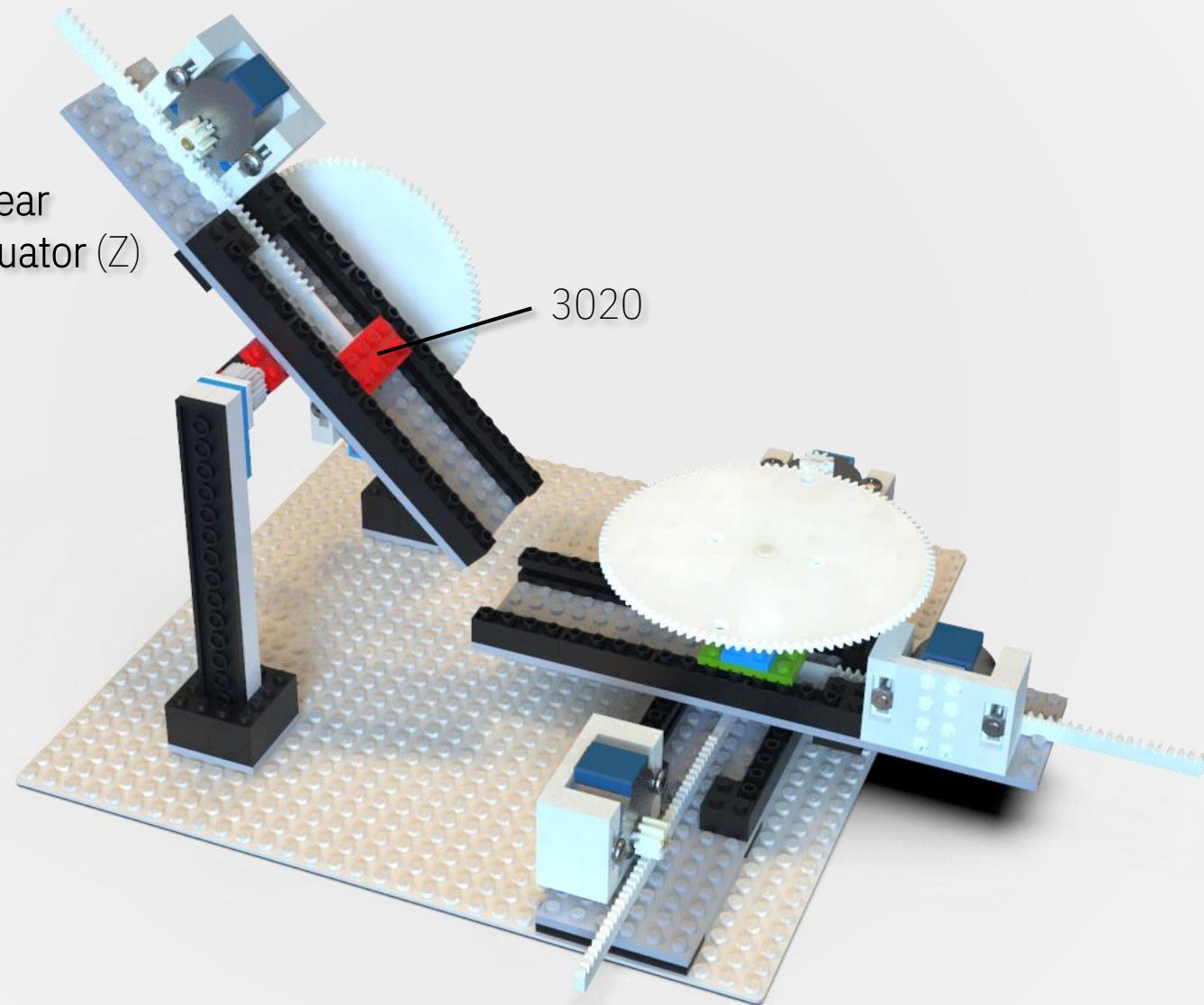


Linear  
actuator (Y)

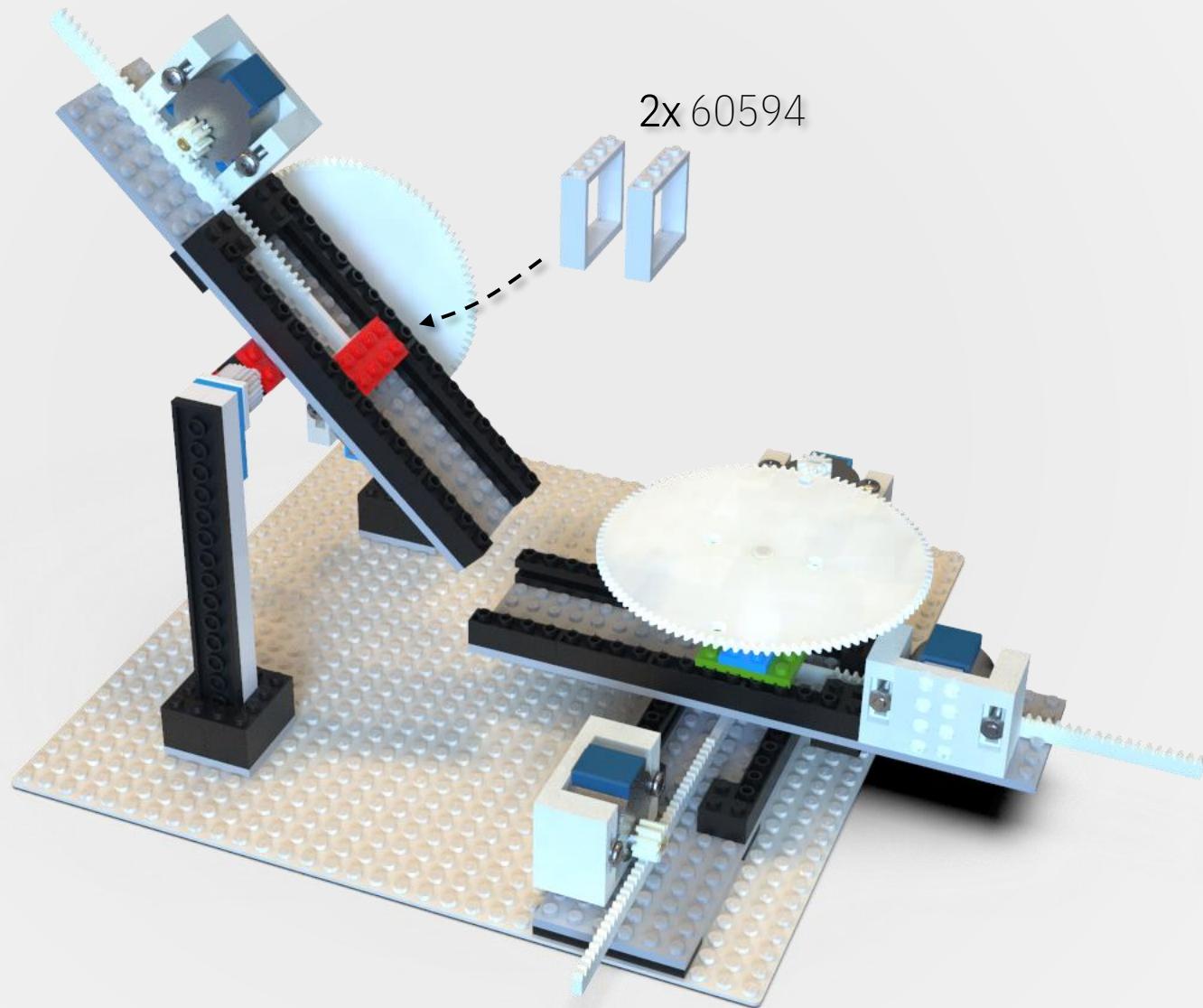
# Final assembly



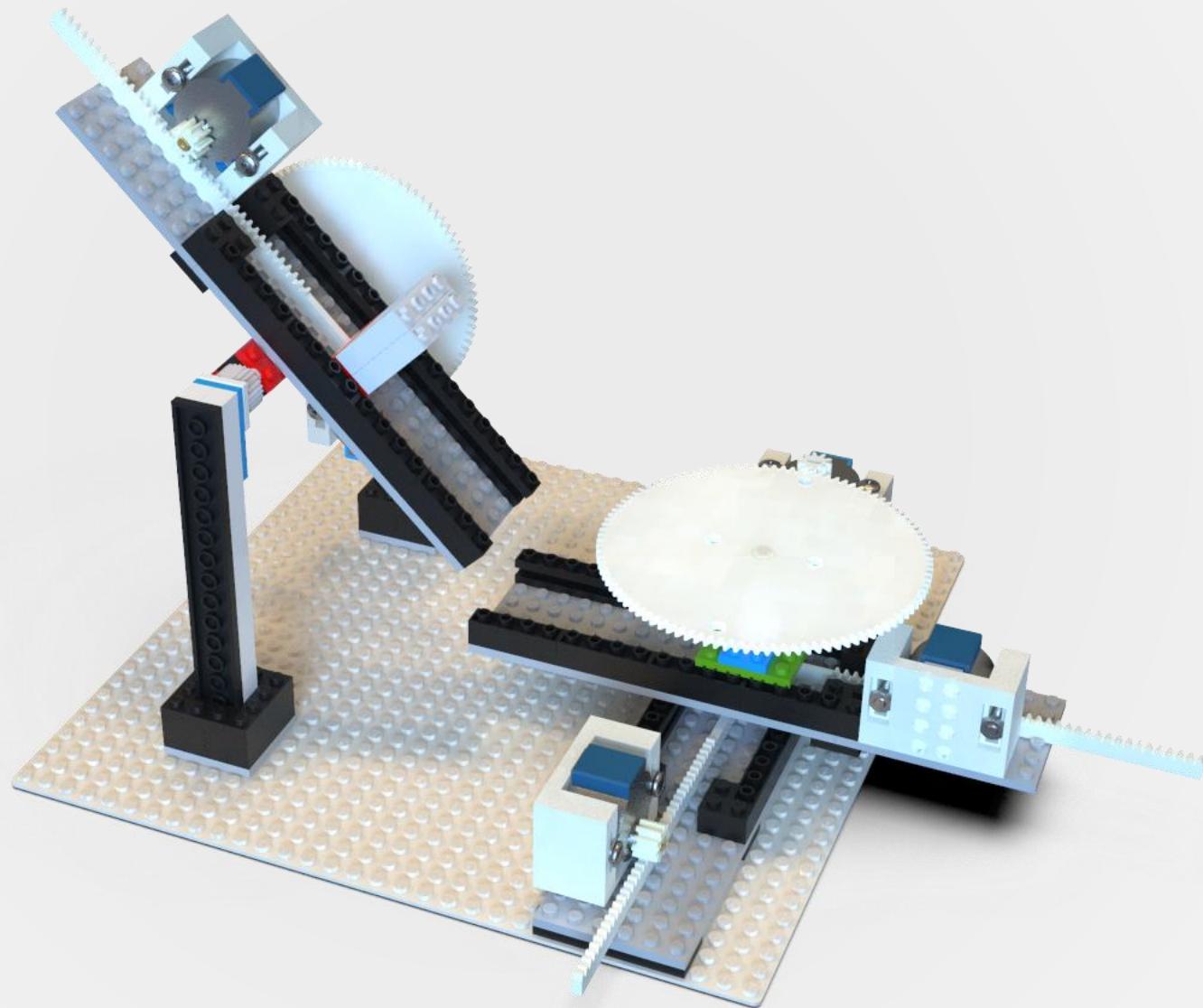
# Final assembly



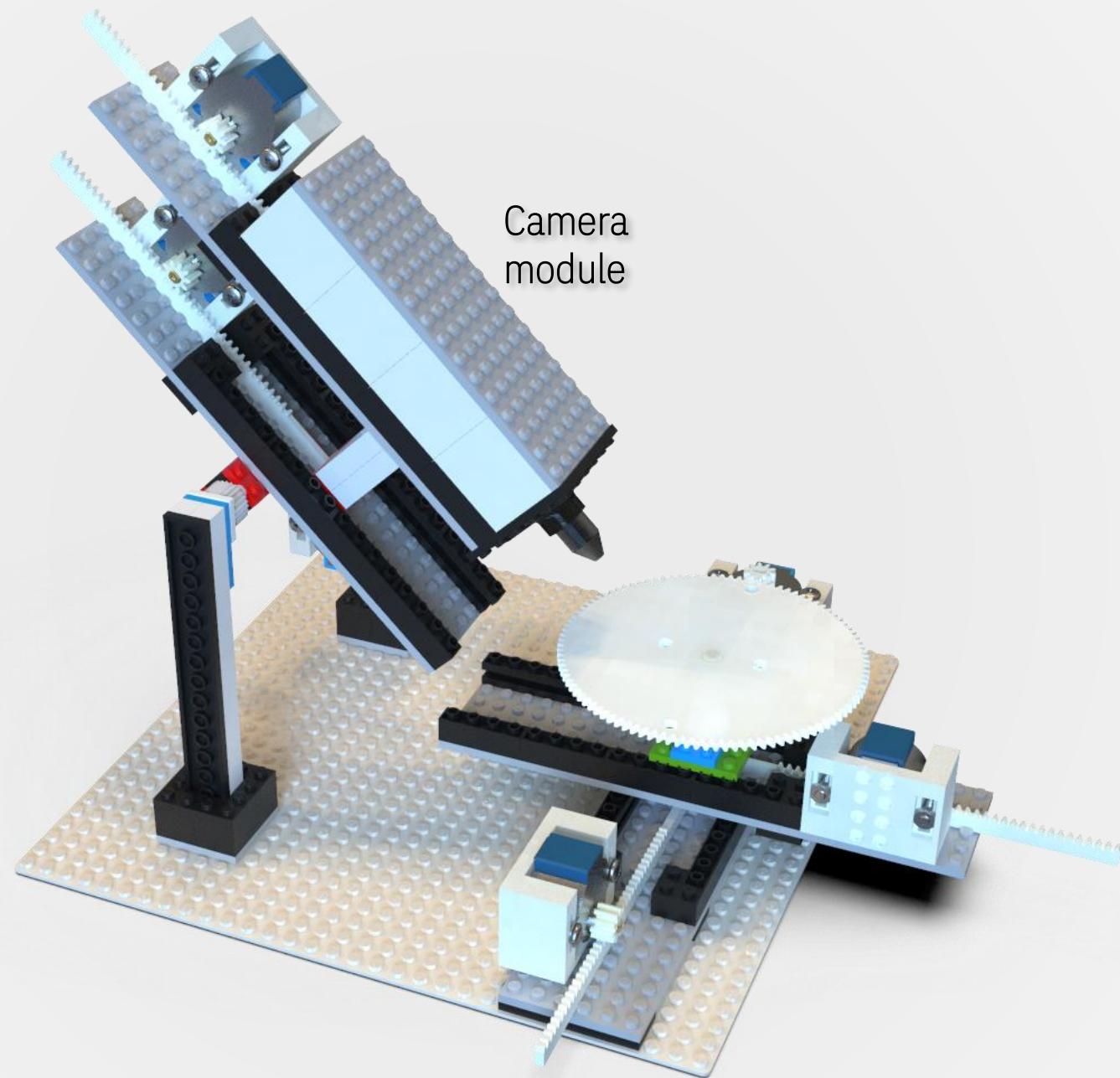
# Final assembly



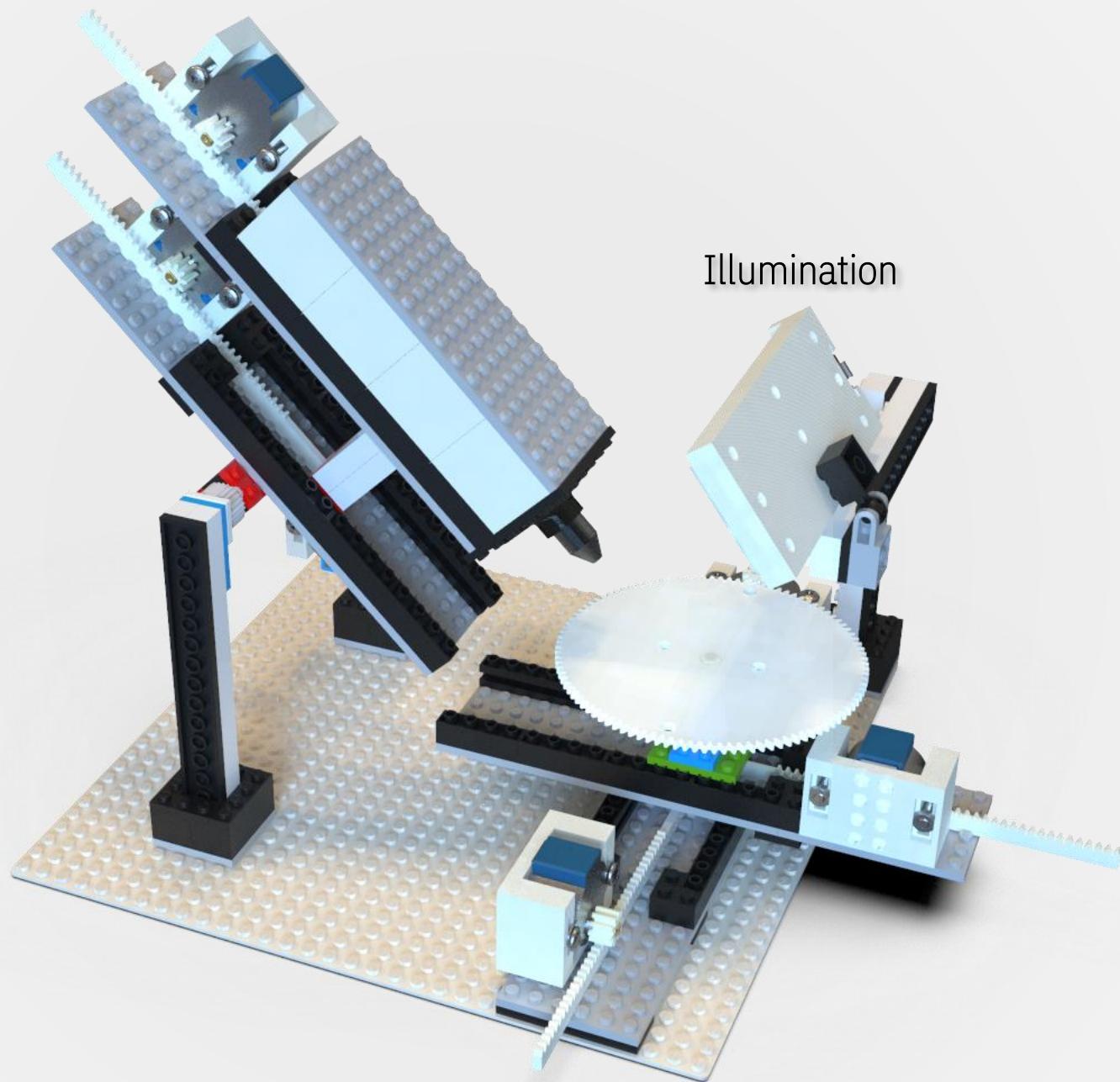
# Final assembly



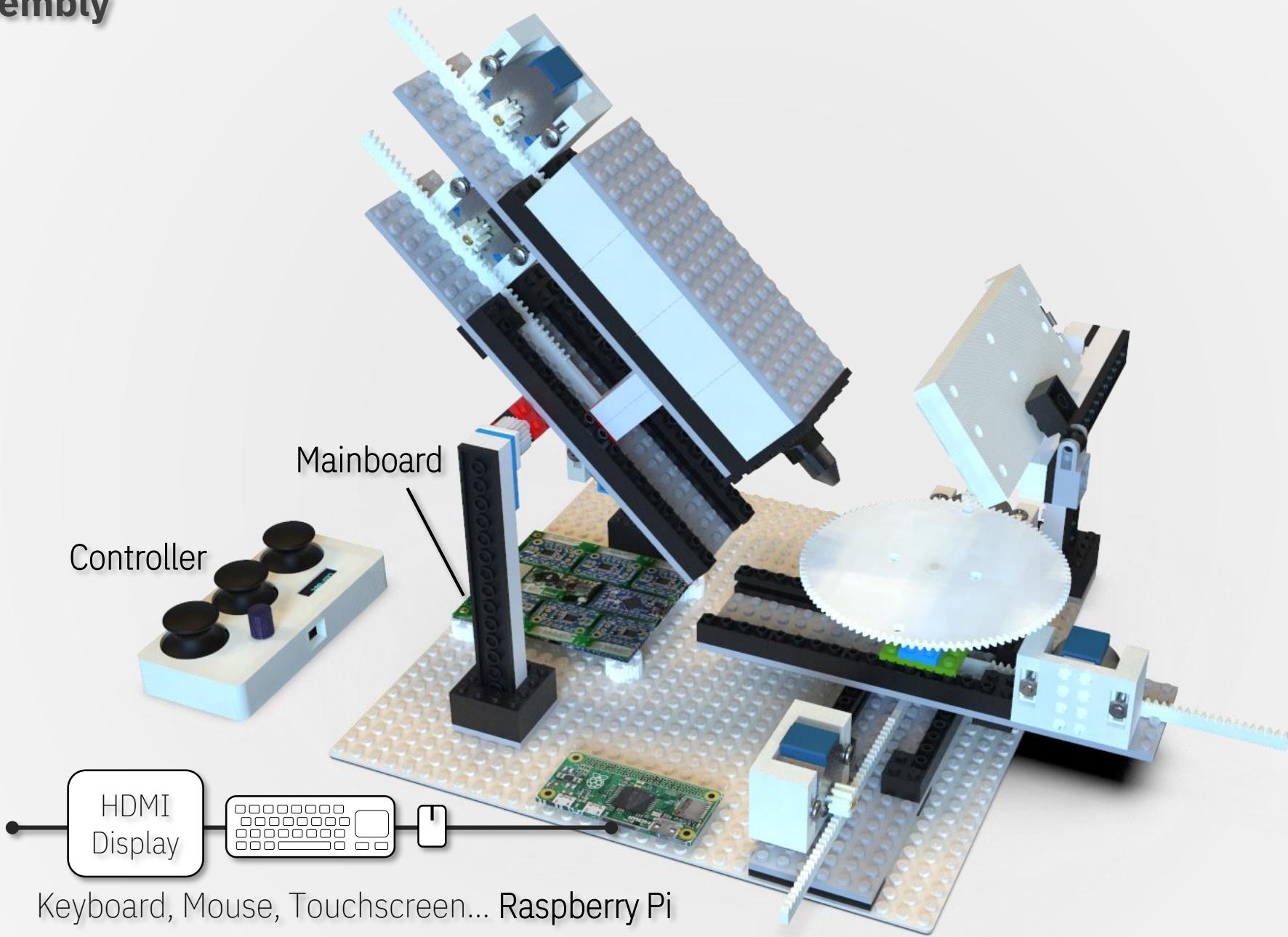
# Final assembly



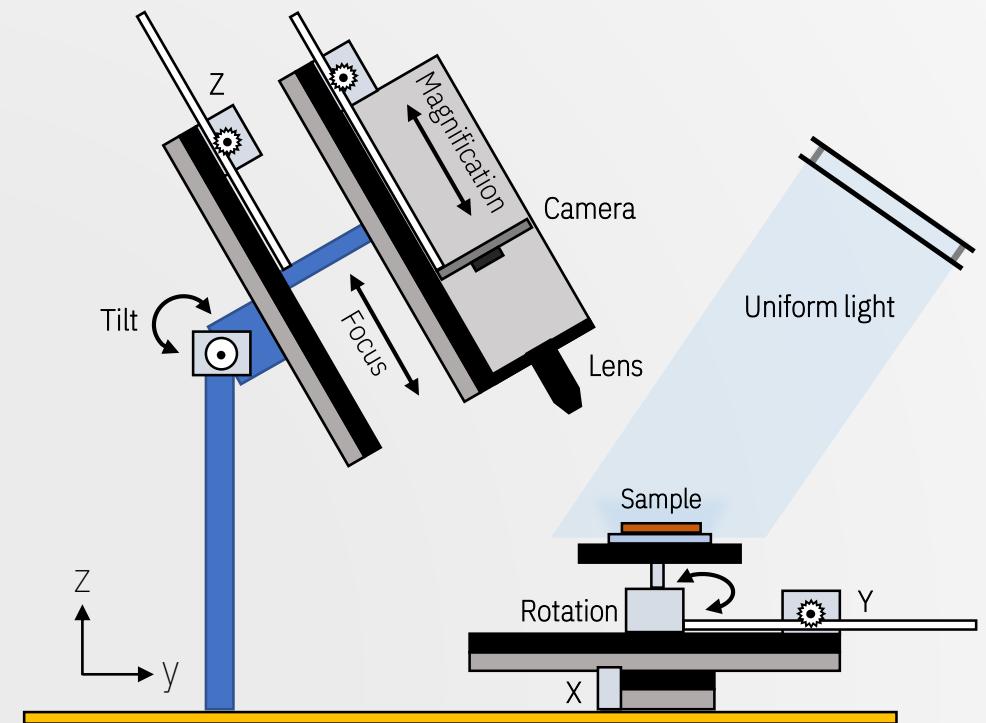
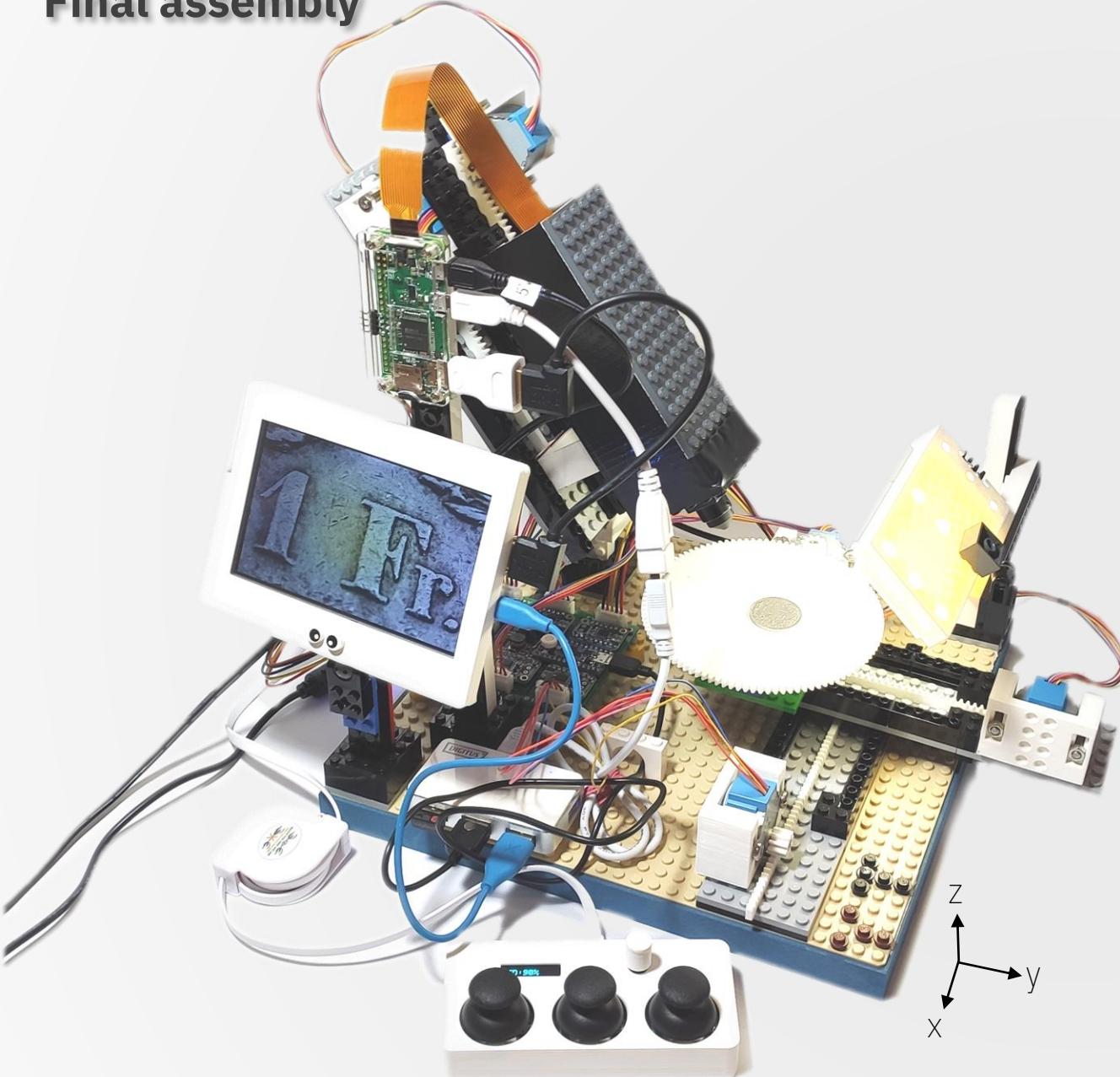
# Final assembly



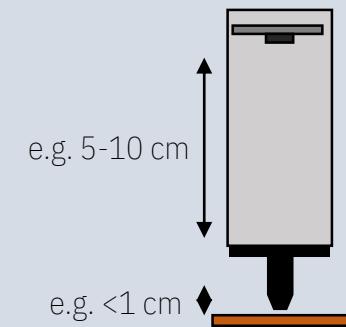
# Final assembly



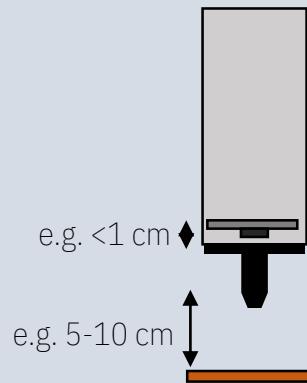
# Final assembly



High magnification  
(Microscopy)

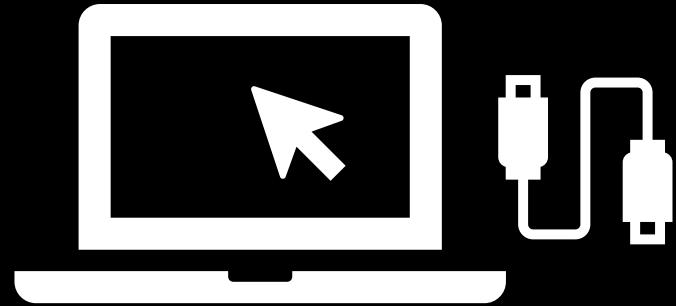


Low magnification  
(Macro photography)



# Software

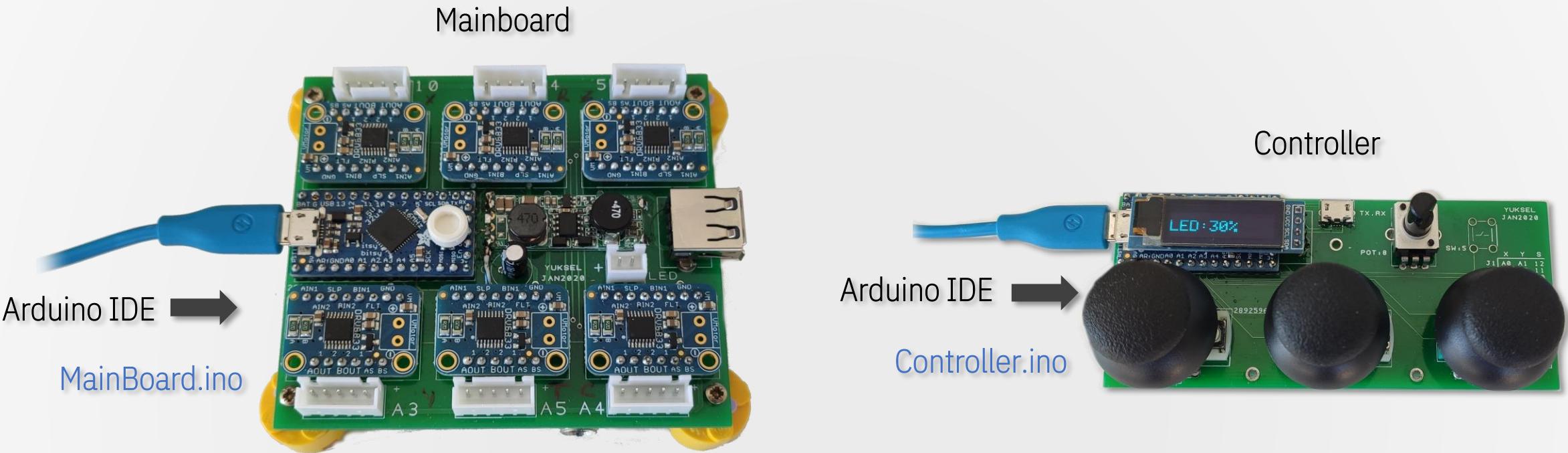
---



1100100111...

# Arduino

(Photograph)



Board: Adafruit ItsyBitsy 32u4 - 5V 16MHz

Required libraries:

- AccelStepper (*for the stepper motors*)
- Adafruit\_NeoPixel (*for the optional status LED*)

Board: Adafruit ItsyBitsy 32u4 - 5V 16MHz

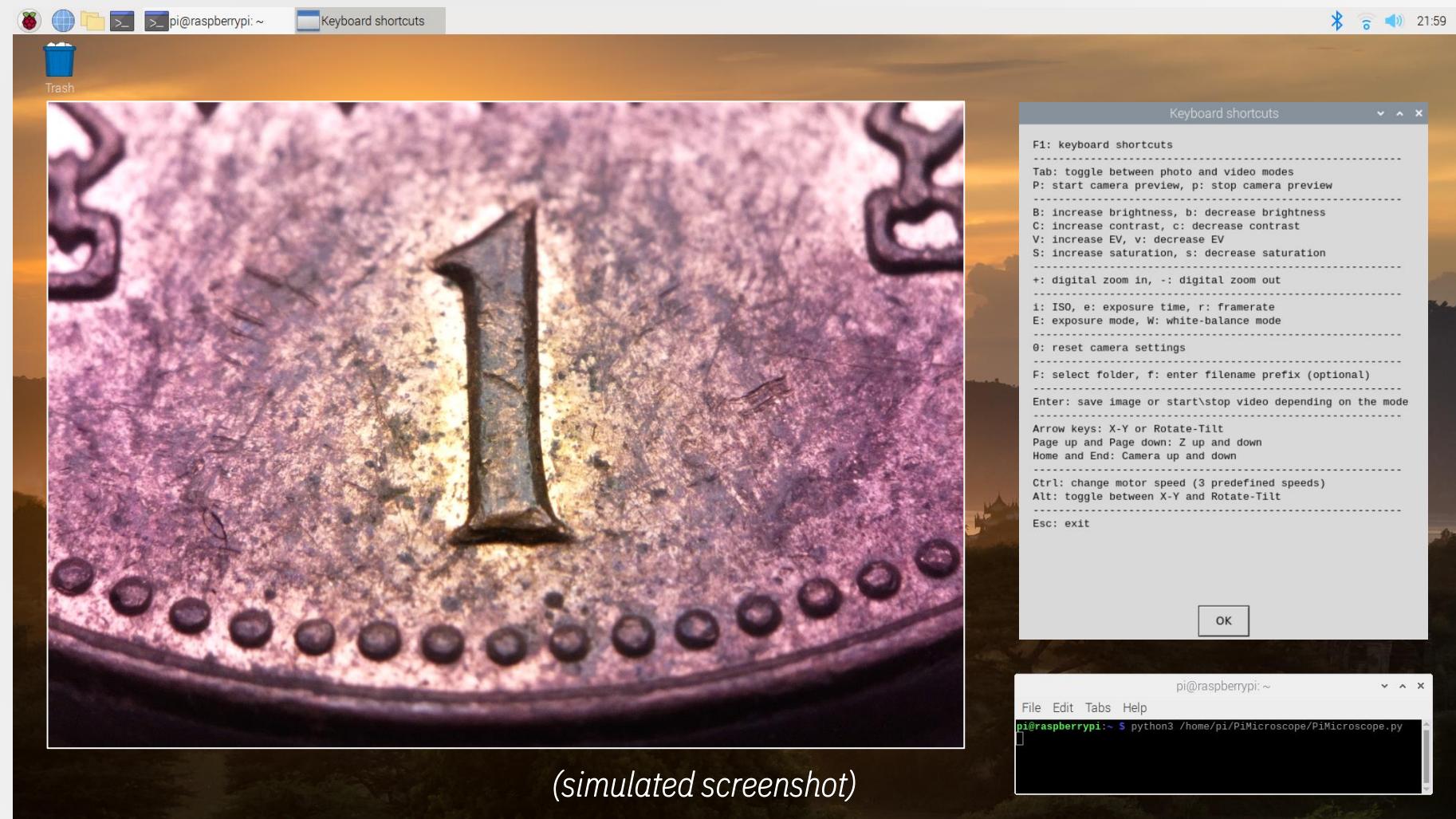
Required libraries:

- Wire (*for the OLED display*)
- Adafruit\_GFX (*for the OLED display*)
- Adafruit\_SSD1306 (*for the OLED display*)

# Raspberry Pi – Python

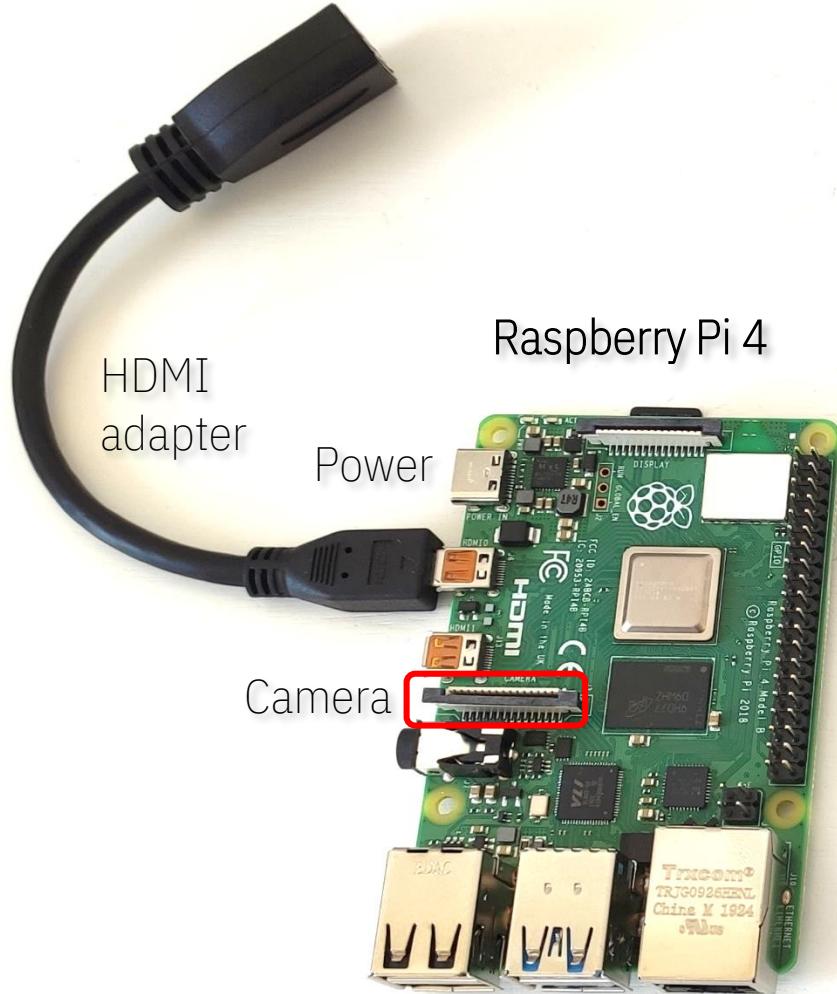
MicroscroPy.py: a program written in Python 3 to control the microscope, modify camera settings and take photos and videos from keyboard

- Sends commands to Arduino mainboard to control the stepper motors and the LED from keyboard (alternative to the joystick controller)
- Modules used:
  - sys
  - easygui
  - pySerial
  - time
  - Pynput
- In Raspberry Pi 4, GPU memory increased to 144MB from 128MB to achieve max. camera resolution



# Raspberry Pi - Python

(Photograph)



HDMI  
adapter

Power

Raspberry Pi 4

Raspberry Pi 3

Power

Camera



USB hub

Raspberry Pi Zero

HDMI adapter

Power

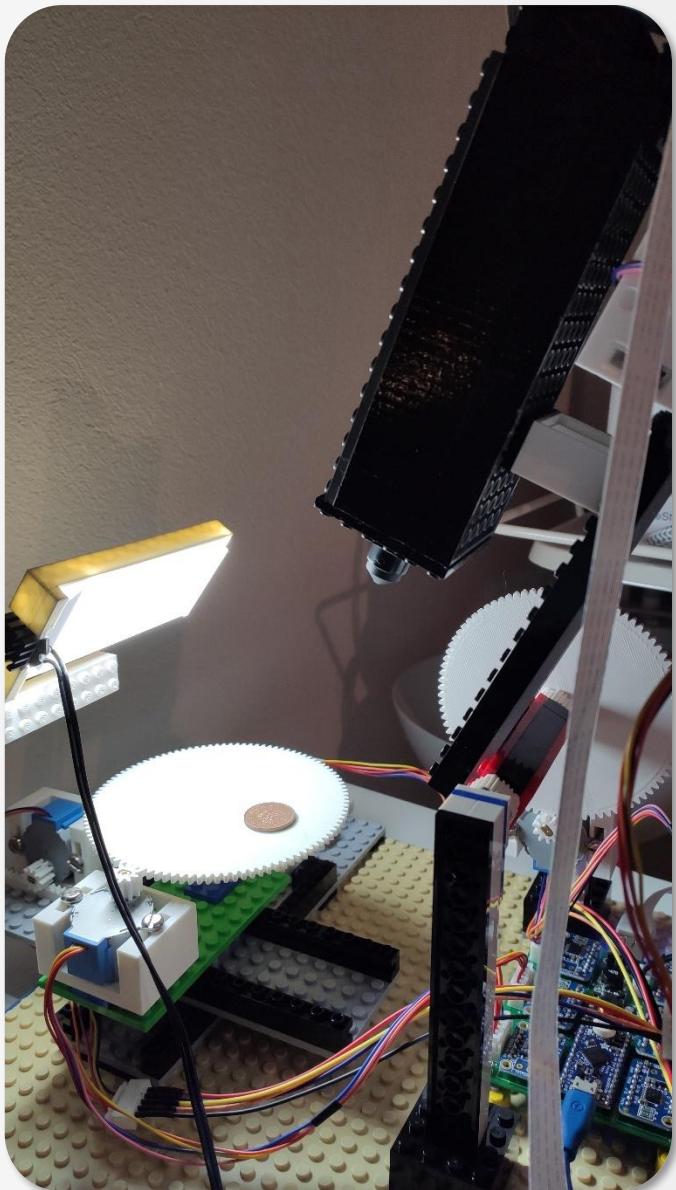
Camera

Tested at **3280×2464** (max.)  
camera resolution

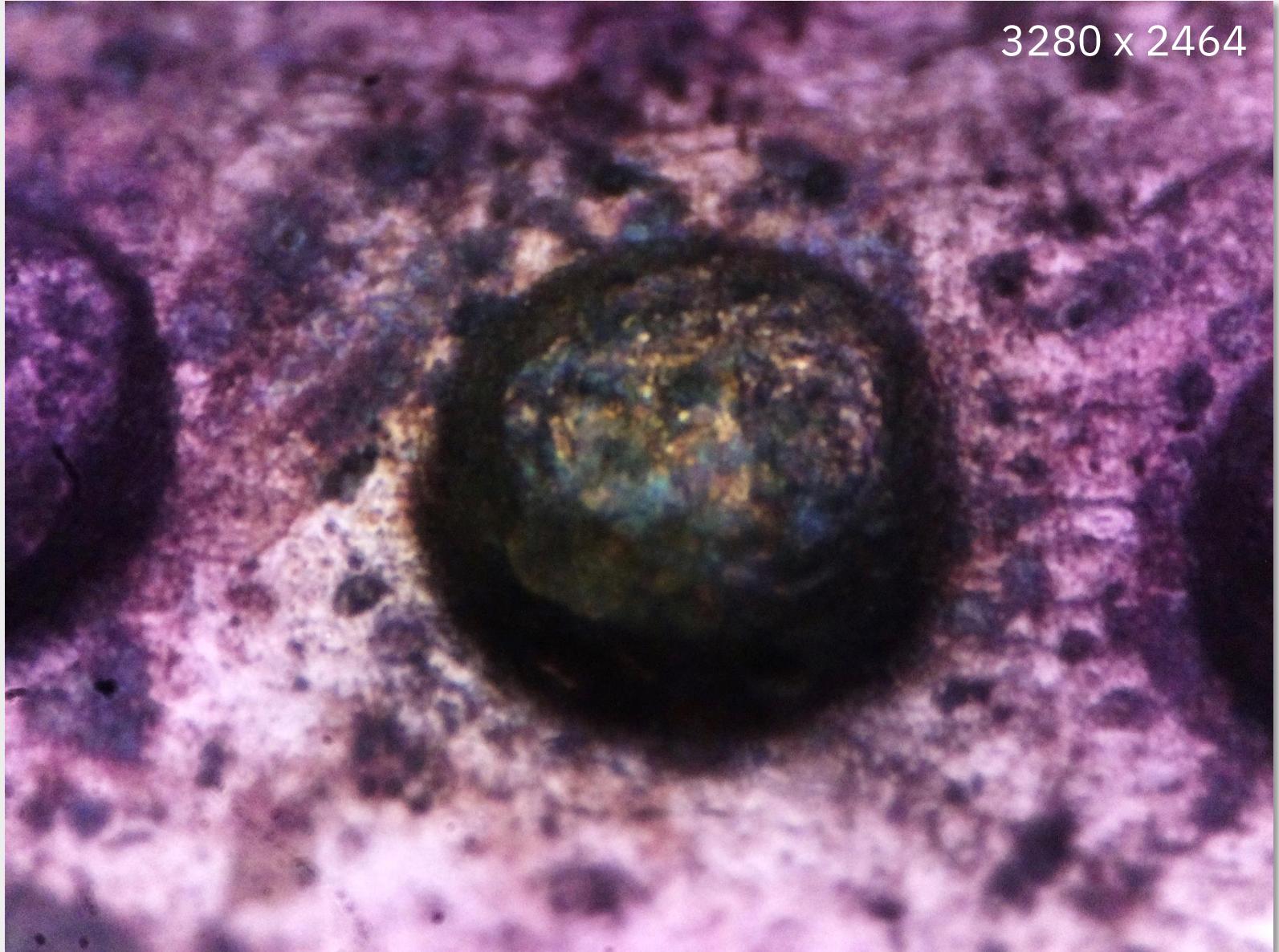
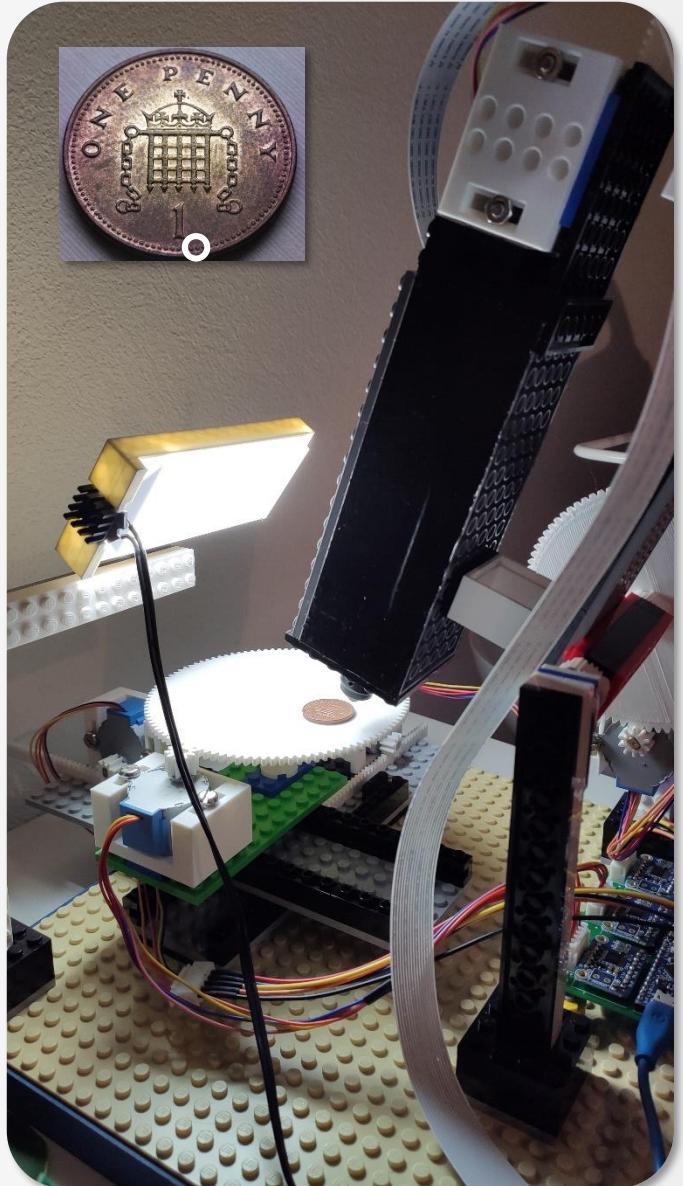
Tested at **1920×1080**  
camera resolution

Tested at **1920×1080**  
camera resolution

## Examples of images - macro



## Examples of images - micro



## Examples of images

Macro photography and video taken at different angles and magnification



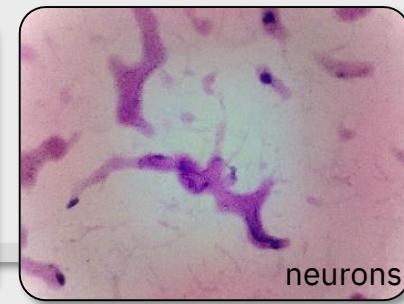
fruit fly



rice

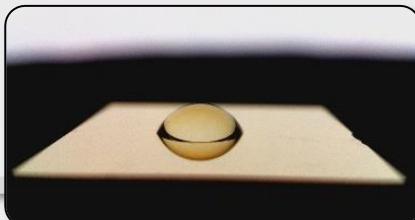
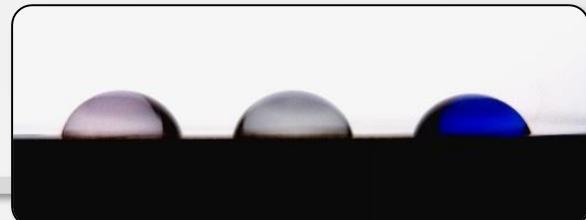
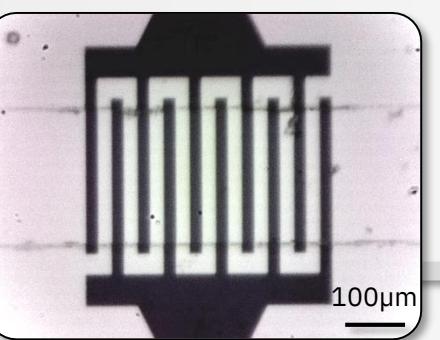
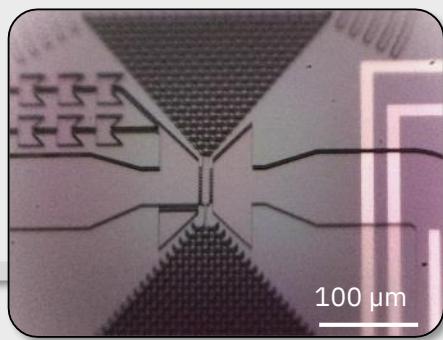
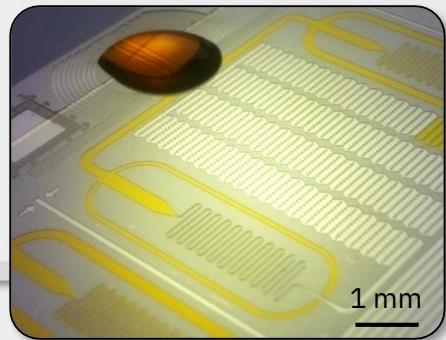


neurons



2D and 3D (tilted) imaging of biological samples

Microscopy (reflected or transmitted light)



Contact angle measurements

# Bill of materials

| Item                                    | Total price (\$) | Remarks   |
|---|------------------|---|
| <a href="#">LEGO bricks</a>             | ~60              | I recommend buying extra bricks in different sizes.   |
| <a href="#">Raspberry Pi</a>            | 10-50            | Any model works.  |
| <a href="#">Raspberry Pi camera</a>     | 25               |   |
| <a href="#">Camera flex cable</a>       | 3                | A long cable (>40cm) is preferred if the Raspberry Pi cannot be placed closer to the camera.                        |
| <a href="#">Magnifying lens</a>         | 10               | M12 thread  |
| <a href="#">Micro SD card</a>           | 10               | Raspberry Pi operating system. Minimum 16GB is recommended.   |
| <a href="#">HDMI display</a>            | 50-150           | Optional if an HDMI display is already available. The price varies depending on the size and the model.             |
| <a href="#">Arduino board</a> x2        | 20               | One for the mainboard, one for the controller. Any Arduino board with enough number of IO pins should work.         |
| <a href="#">Stepper motor driver</a> x6 | 30               | Any other 5V compatible driver should work. Microstepping was not needed in this project.                           |
| <a href="#">Stepper motor</a> x6        | 15               | 28BYJ-48, 5V, 1/64 gear ratio   |
| <a href="#">JST-XH 5P 4S cables</a>     | 7                | A few of these cables are required to extend the default cable of the stepper motor (28BYJ-48)                      |
| <a href="#">High-power LED</a>          | 4                |   |
| <a href="#">High-power LED driver</a>   | 14               |   |
| <a href="#">Light diffuser</a>          | 3                | An LED backlight module is modified by replacing the low-power LED by a high-power one                              |
| <a href="#">Potentiometer</a>           | 1                | LED intensity control using pulse-width modulation.   |
| <a href="#">Thumb joysticks</a> x3      | 12               | Three joysticks for X, Y, Z, Camera, Tilt, and Rotation.  |
| <a href="#">5V power supply</a>         | 20               | At least 3A is preferred. A more powerful supply may be needed if the display is also powered from the same supply. |
| <a href="#">DC-DC converter</a>         | 3                | In case the LED and its driver require more than 5V.  |
| <a href="#">OLED display</a>            | 3                | Displays the LED intensity or any other information on the controller.  |
| <a href="#">USB connector</a> (type A)  | 0.5              | Connection between the mainboard and the controller (mainboard side).   |
| <a href="#">USB connector</a> (micro)   | 0.5              | Connection between the mainboard and the controller (controller side).  |
| <a href="#">JST XH connector set</a>    | 5                | Connectors for the stepper motors and the LED   |
| <a href="#">DC barrel connector</a>     | 1                | In case a separate DC input (e.g. 12V) is needed for the LED  |
| <b>OVERALL TOTAL</b>                    | <b>~300</b>      |   |

- The prices and the links are for exemplary components that were used in this prototype.
- Any other compatible model should work in principle.
- The total price varies significantly depending on the vendor and the model. For example, the type of the Raspberry Pi and the display makes a big difference in the price.
- The cost of 3D-printed parts is not included.
- There are also a few other accessories needed, like USB cables, an HDMI cable, optionally a mouse and a keyboard, an ON/OFF switch, screws and nuts, and printed circuit boards (PCBs).

# List of 3D-printed parts

Click on the design name to go to the corresponding page

| Design name                         | Function/module        | Units needed | Slicer settings – printing speed  | Remarks  |
|-------------------------------------|------------------------|--------------|-----------------------------------|--|
| <a href="#">1-RackGear.stl</a>      | Linear stages          | 4            | Super quality (0.12mm) – 20mm/s   | Requires support structures. Sides of the rack may require minor polishing |
| <a href="#">2-MotorGear.stl</a>     | All stepper motors     | 6            | Super quality (0.12mm) – 20mm/s   |  |
| <a href="#">3-MotorHousing.stl</a>  | All stepper motors     | 6            | Low quality (0.28mm) – 50mm/s     |  |
| <a href="#">4-CameraAdapter.stl</a> | Camera                 | 1            | Standard quality (0.2mm) – 50mm/s |  |
| <a href="#">5-LensAdapter.stl</a>   | Camera                 | 1            | Standard quality (0.2mm) – 50mm/s | Not needed if a commercial lens holder is used                             |
| <a href="#">6-ShaftAdapter.stl</a>  | Rotary stage – option1 | 1            | Super quality (0.12mm) – 20mm/s   | Not needed if the more precise stage (option 2) is used                    |
| <a href="#">7-RotationGear.stl</a>  | Rotary stage – option2 | 1            | Standard quality (0.2mm) – 50mm/s | Not needed if the less precise stage (option 1) is used                    |
| <a href="#">8-TiltGear.stl</a>      | Tilting mechanism      | 1            | Standard quality (0.2mm) – 50mm/s |  |
| <a href="#">9-LEDhousing.stl</a>    | Illumination           | 1            | Low quality (0.28mm) – 50mm/s     |  |
| <a href="#">10-LEDfixture.stl</a>   | Illumination           | 1            | Low quality (0.28mm) – 50mm/s     | A small piece to keep everything in place                                  |

- Parts were designed using **FreeCAD** ([www.freecadweb.org/](http://www.freecadweb.org/)), sliced using **Ultimaker Cura** ([ultimaker.com/software/ultimaker-cura](http://ultimaker.com/software/ultimaker-cura)), and printed using **Creatality Ender 3 Pro** printer with a metal extruder and a BLTouch auto bed-leveling sensor.
- 1.75mm white **PLA filament** (Purefil, Switzerland) was used (extruder Temperature: 200 °C, bed temperature: 60 °C).
- A good quality printing depends on many factors and parameters, especially a good bed leveling.
- Some parts, especially the holes fitting to the studs of LEGO bricks, may require soft grinding using a file or a sandpaper.
- Gears used in the illustrations and the first prototype had a larger pitch. The latest design files have a finer pitch for better precision. All designs have been tested.

# Storage



- LEGO bricks
- 3D printed parts
- Raspberry Pi + camera
- Arduino mainboard
- Arduino controller
- HDMI display
- Cables ...

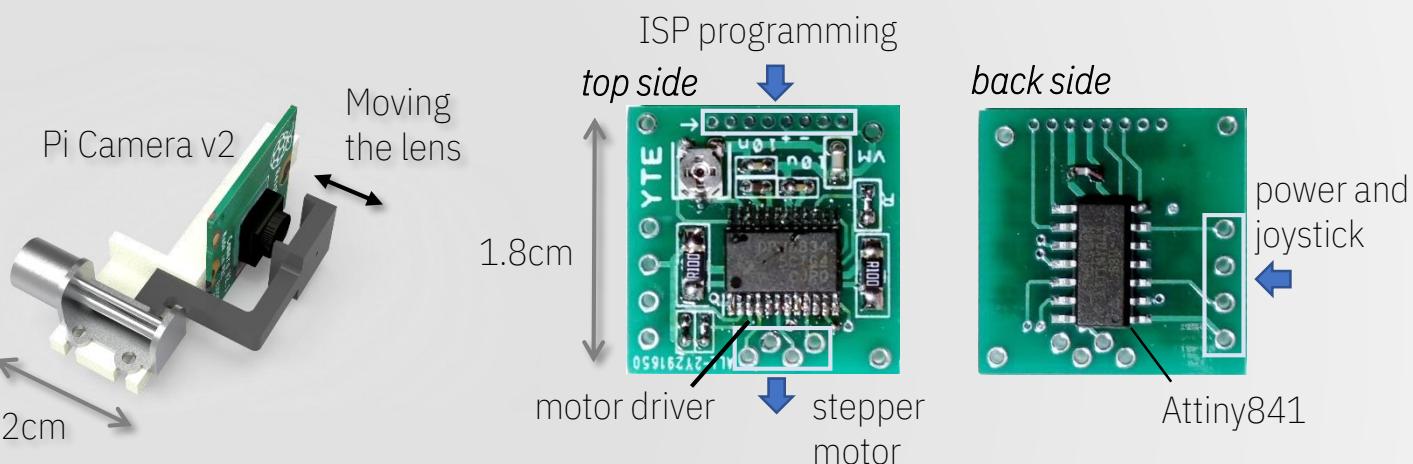
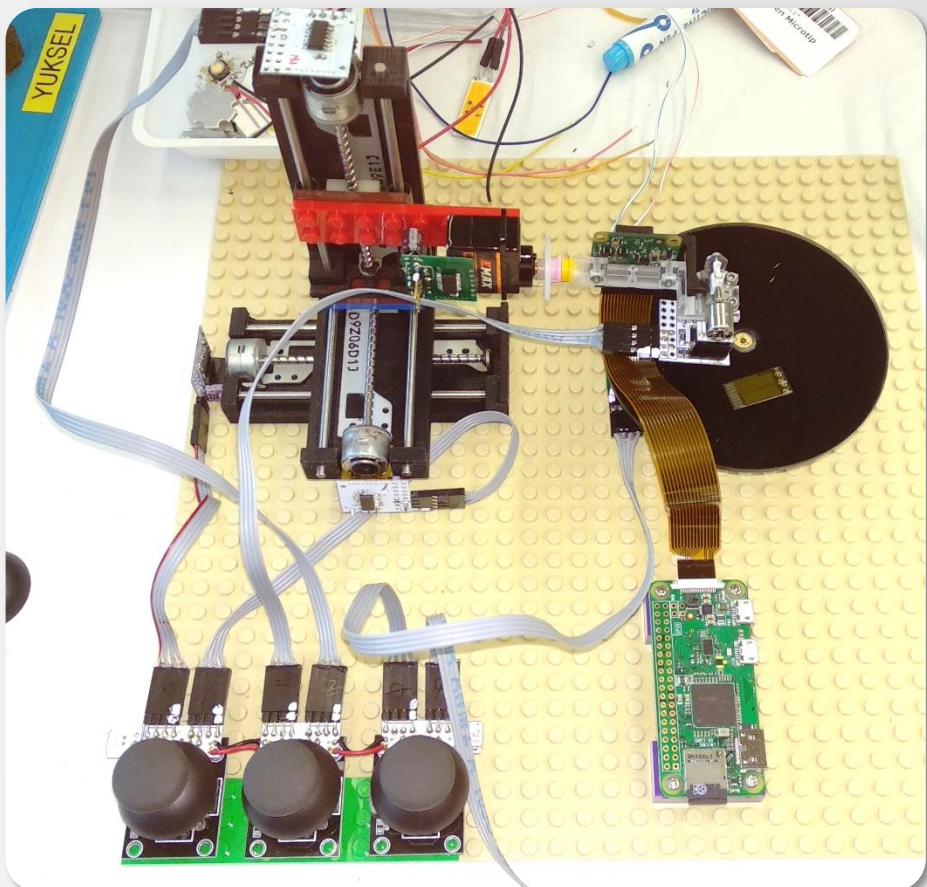
# Learning by failing

“Experience is a truer guide than the words of others”

*Leonardo da Vinci*

# Learning by failing – 1 (stepper motors from CD-ROMs)

- XYZ stage: 3D printed linear actuators using stepper motors from CD-ROMs
- Camera focus/magnification: Moving the camera lens using a tiny stepper motor
- Electronics: Custom Arduino board using Attiny841 microcontroller (one board for each motor)
- Mechanical parts: Mostly 3D printed, only a few LEGO bricks

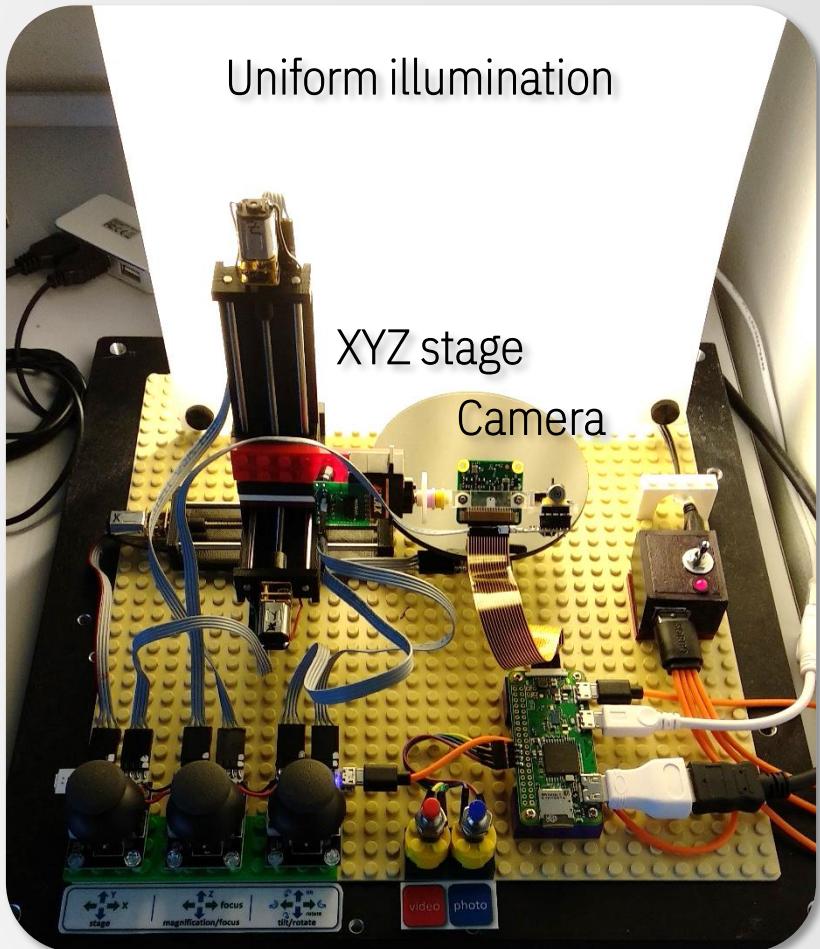


## Issues:

- Custom-design components, not available off-the-shelf
- Limited travel distance and magnification range
- Linear movements not very smooth (18-degree per step)
- Difficult to protect the image sensor from stray light
- Assembly of the custom Arduino/motor driver board requires high precision soldering

## Learning by failing – 2 (DC gear motors)

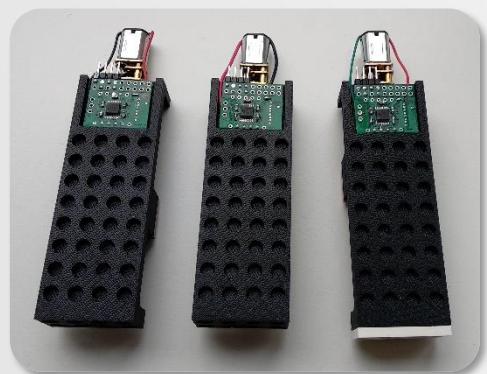
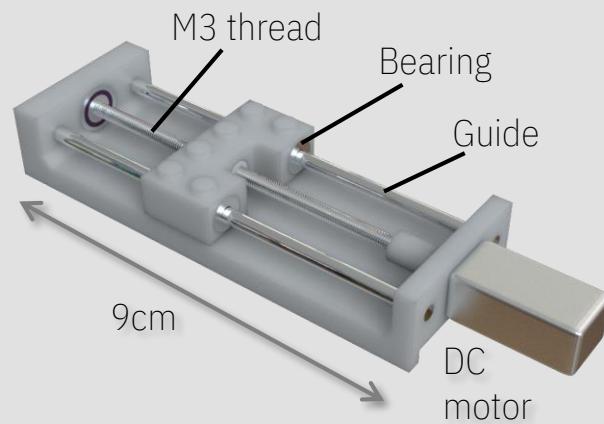
- XYZ stage: 3D printed linear actuators using DC gear motors
- Camera focus/magnification: Moving the camera lens using a tiny stepper motor
- Electronics: Custom Arduino board using Attiny841 microcontroller (one board for each motor)
- Mechanical parts: Mostly 3D printed, only a few LEGO bricks



Uniform illumination

XYZ stage

Camera

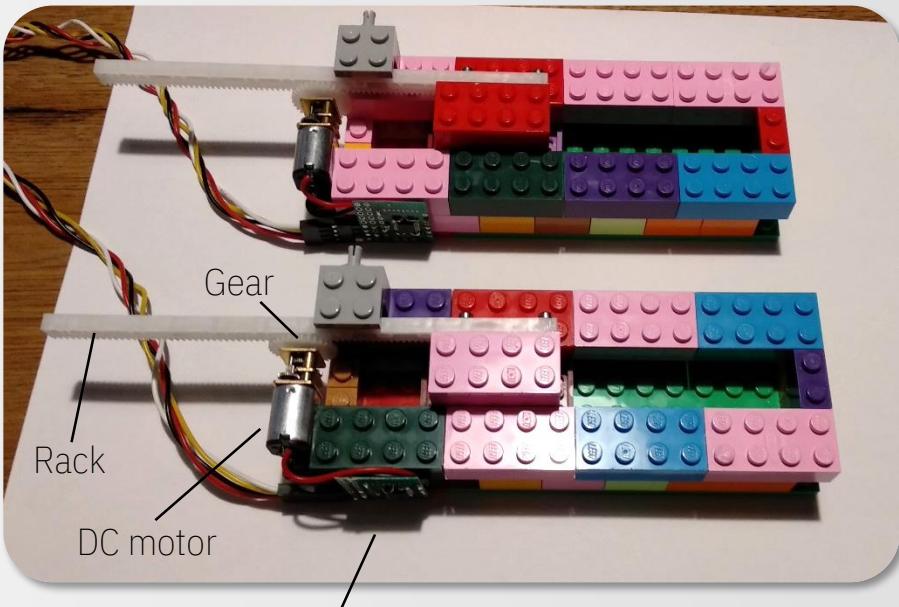


### Issues:

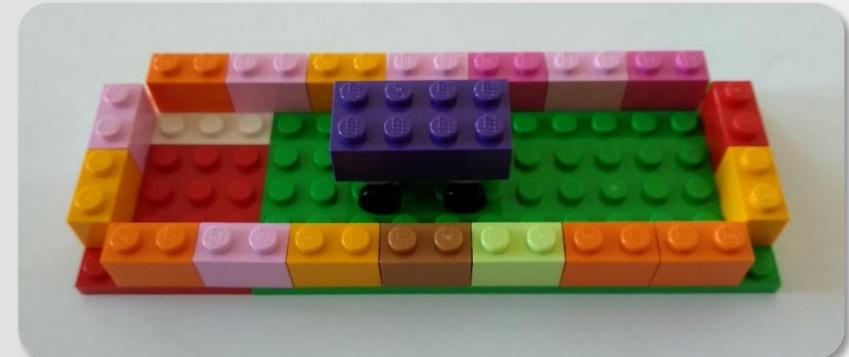
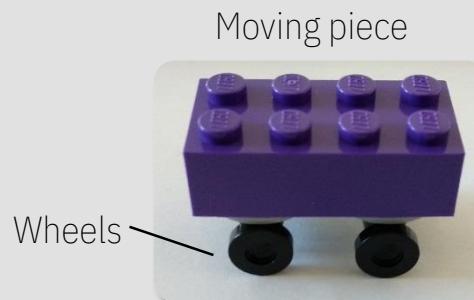
- Custom-design components, not available off-the-shelf
- Smoother movements but cannot take much load at lower speeds
- Thread and guides bend when pushed too much, resulting in wobbles during movement

## Learning by failing – 3 (DC motor + pinion gear + rack)

- XYZ stage: Moving a LEGO piece using a gear + rack combination
- Camera focus/magnification: Moving the camera lens using a tiny stepper motor
- Electronics: Custom Arduino board using Attiny841 microcontroller (one board for each motor)
- Mechanical parts: Mostly LEGO bricks



Arduino + motor driver  
(custom designed)

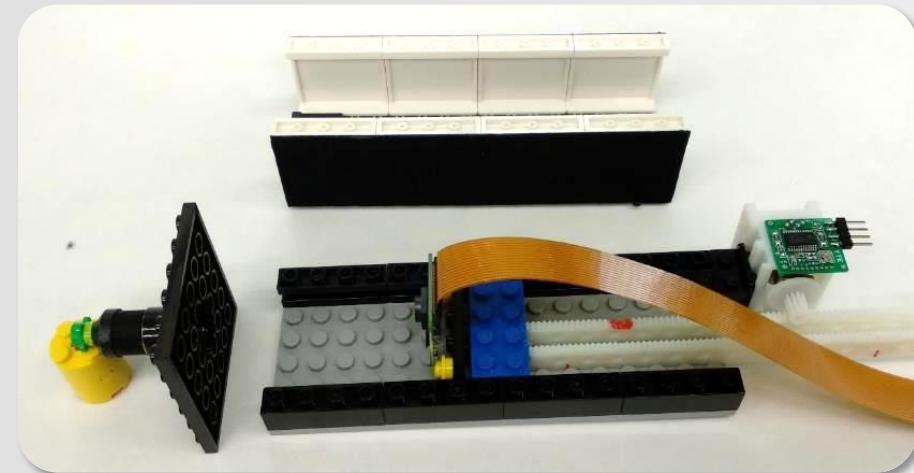
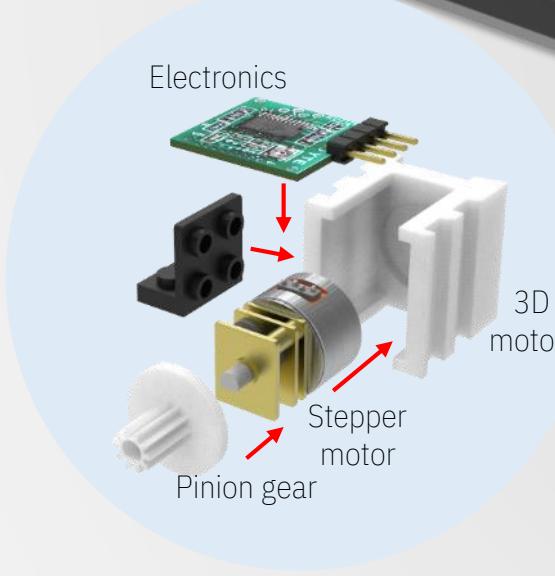
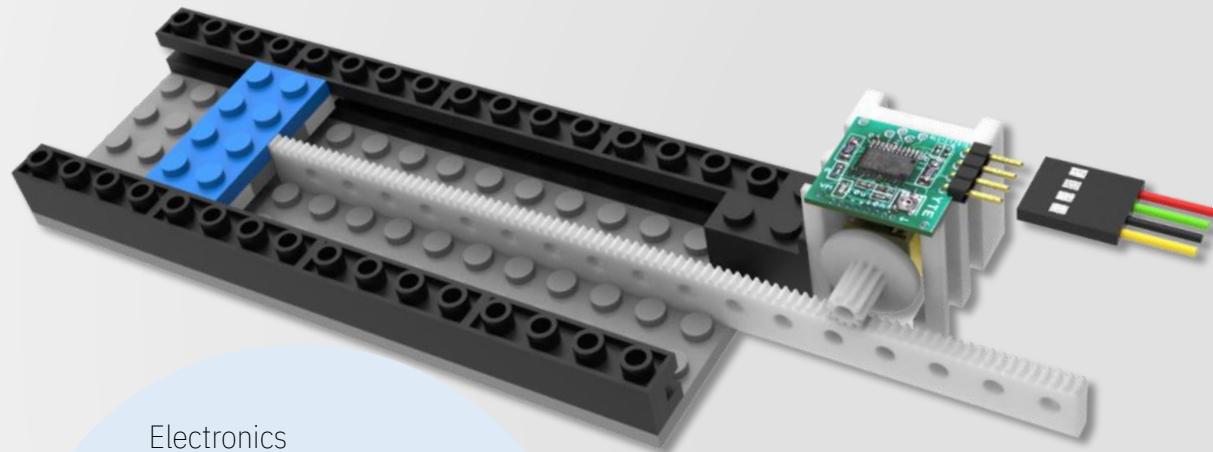


### Issues:

- Difficult to assemble
- Side-to-side and up-and-down movements

## Learning by failing – 4 (Tiny stepper motor + pinion gear + rack)

- XYZ stage: Moving a LEGO piece using a gear + rack combination and LEGO sliding guides
- Camera focus/magnification: Fully enclosed housing with a dedicated lens (same as the latest design)
- Electronics: Custom Arduino board using Attiny841 microcontroller and driver with microstepping (one board for each motor)
- Mechanical parts: LEGO bricks and 3D printed parts



### Issues:

- Requires high precision 3D printing (expensive equipment)
- Stepper motor: expensive and not easily obtainable in popular electronic stores

- This project did not receive any technical or financial support from the companies mentioned in this document.
- LEGO® is a trademark of the LEGO Group of companies.
- Raspberry Pi is a trademark of the Raspberry Pi Foundation.
- Arduino® is a trademark of Arduino AG.

[www.zurich.ibm.com/st/precision\\_diagnostics](http://www.zurich.ibm.com/st/precision_diagnostics)

