

OCLab assembly instruction

Dimitri Fichou and Gertrud Morlock

Useful resources

Some knowledge is mandatory to build this device. The following links should give an overview. Over this document, it is assumed a significant amount of time had been spend on those websites and that their content is known.

3D printing:

- [Reprap project](#)
- [Gcode listing](#)
- [Course on 3D printing](#)

Information on each electronic board used:

- [Ramps 1.4 board](#) (electronic of our machines)
- [Inkshield board for inkjet](#), more info from people who used the board [here](#), [here](#), [here](#). [Place to buy the cartridge](#)
- [Raspberry Pi project](#)
- [Arduino project](#) (see [those courses](#) for an introduction on the Internet of things)

Some software to install:

- [Openscad software](#) (for drawing, see [this tutorial](#))
- [Slic3r software](#) (to create Gcode file for 3d printing, [here is the main page](#) but it's better to install the Prusa edition)
- [Arduino IDE](#) (to modify and upload the [marlin firmware](#))

Others:

- [Instructable for LED control from the Ramps 1.4](#)

Print the parts

SCAD folder

SCAD file: `full_view("printed")`.

SCAD files can be found in the SCAD folder, normally, only the file `OCLab.scad` is of interest. Each different module can be observed separately by changing the argument of the `full_view` module.

STL folder

The STL files in this folder contain several files, only the file `OCLab.stl` should be considered up to date. Parts are designed to be printed on a Prusa i3 MK2 with the *0.35mm fast* setting. Normally, no supports are needed but a brim could be usefull. Be sure to rotate the part so they print correctly.

Bill of material

Table 1: Bill of material - Estimated total price: 810 Euro (VAT)

Name	Amount/comment	Seller	Euro (VAT)
Plastic PLA	1kg	Colorfabb	13
Zip ties	pack of 100	Local workshop	2
m8uu bearing	8	Motedis	10
lm8uu housing	4	Motedis	12
623zz bearing	4	Amazon	2
gt2 belt	2 meters	Motedis	3
gt2 pulley 20 teeth	2	Motedis	3
Nema 14	2	Emotiontech	30
8mm stainless steel rod	4x255 mm	Motedis	15
20x20 profile 5 I-type	5x215 mm (X) + 4x255 mm (Y) + 6x 210 mm (Z)	Motedis	10
Nut for 20x20 profile 5 I_type	pack of 100	Motedis	20
Cube connector for 20x20 profile	pack of 10 (6 needed)	Motedis	25
Angle connector for 20x20 profile	pack of 10 (10 needed)	Motedis	10
Aluminium frame for box	2 mm thick	Local workshop	30
Screw M5 x 10	pack of 100 hex type	Local workshop	5
Screw M5 x 16	pack of 100 hex type	Local workshop	5
Screw M3 x 30	pack of 100 phillips or slotted type	Local workshop	5
Screw M3 x 10	pack of 100 hex type	Local workshop	5
Screw M3 x 16	pack of 100 phillips or slotted type	Local workshop	5
Screw M4 x 20	pack of 100 phillips or slotted type	Local workshop	5
Nut M3	pack of 100	Local workshop	3
Nut M4	pack of 100	Local workshop	3
Nut M5	pack of 100	Local workshop	3
Washer M3	pack of 100	Local workshop	3
Washer M4	pack of 100	Local workshop	3
Washer M5	pack of 100	Local workshop	3
Magnet	8: cylinder OD: 8 mm, height: 3mm	Amazon	2
Glass plates	1 mm thick, recycled from commercial plates	Recycled	0
Raspberry pi	Model 3 is better but model 2 is compatible.	Conrad	35
5v 2.5 A power supply	For raspberry pi	Conrad	10
16 Giga bit SD card	1	Conrad	10
Ethernet cable	1	Local workshop	5
Raspebrry pi camera	1	Conrad	25
Raspberry pi camera longer cable	200 mm is enough	Conrad	5
Arduino megga 2560	1	Conrad	30
Ramps 1.4	1	Amazon	30
A9688 motor driver	2	Amazon	20
Endstop	2	Amazon	10
Inkshield board	1	Nerdcreationlab	100
12v 10A power supply	5A could be enough	Amazon	20
NPN transistor 2n3904h33	3	Conrad	5
LED strip	1 meter	Conrad	10
Prototype soldering plate	60*40 mm	Amazon	5
Breadboard Jumper Wires Ribbon Cable	NA	Amazon	5
255 nm LED	Optan 225P SMD 4-6 mW	Crystal IS	200
Power supply for UV LED	No information	Local workshop	50

Frame

SCAD file: `full_view("frame")`.

Needed parts:

- 76x M5 sliding nuts
- 20x20 profiles type I
- 6x 20x20 cube connectors
- 10x 20x20 angle connectors
- M5 screw for cube connector (sold with it normally)
- M5*10 screw for angle connectors
- printed parts: `feets`; `green_top_front`
- closing plates

Before assembling the frame, set the nuts to have one nut for each M5 screw shown in Figure 1. There is 76 in total. The best is to use the openscad file and turn the model around.

The verticale profile in the middle as well as the horizontale one in the top-middle are situated 120 mm away from the ones in the back.

The outside frames can be found in the `outside frames` folder in pdf format. The plates to close the box can be added now or later but can make the assembly difficult, worth case scenario, they are put to early and need to be dismounted to grant access.

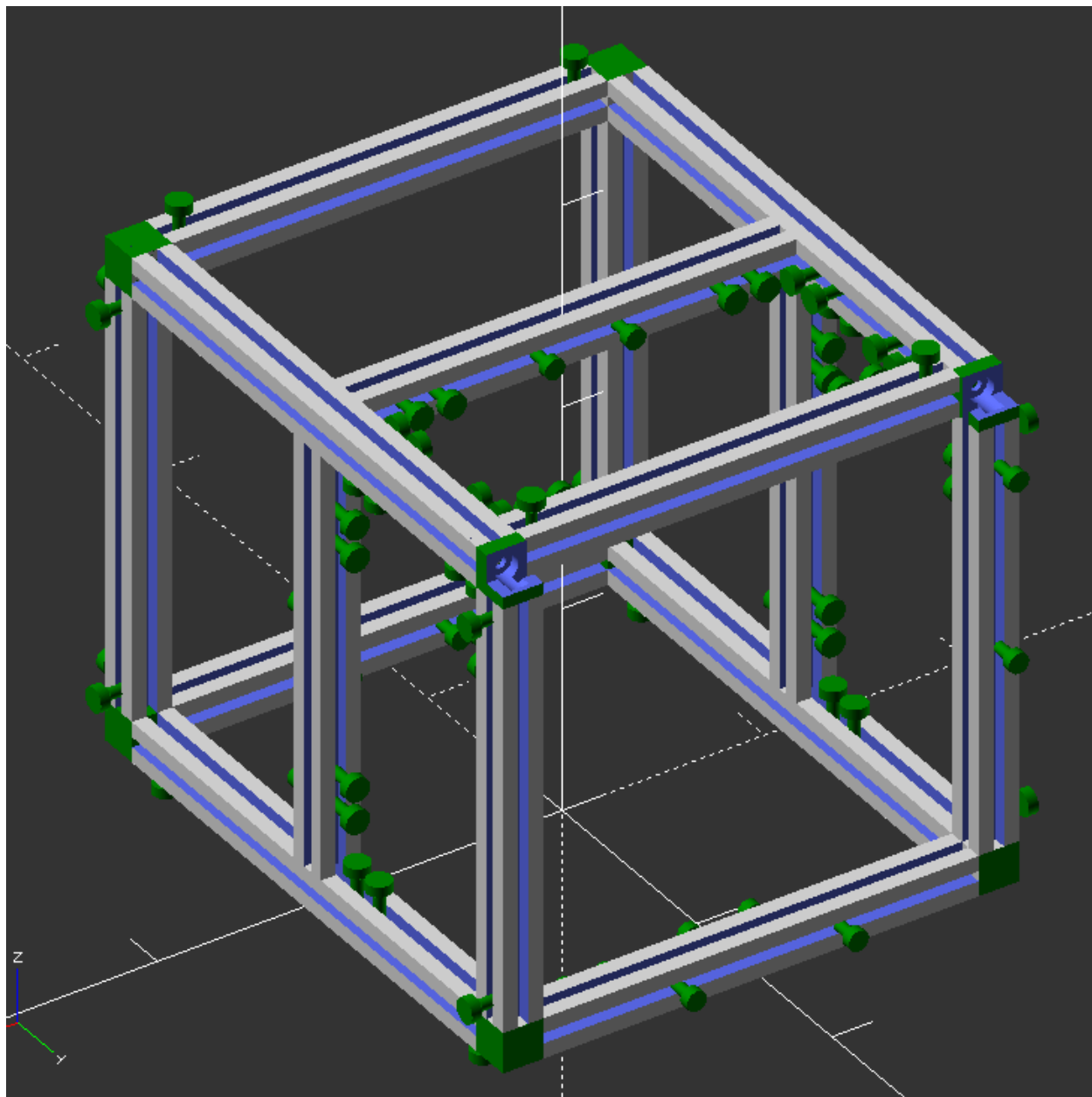


Figure 1: Frame assembly.

Y axis

SCAD file: `full_view("Y axis")`.

Needed parts:

- 2x 8mm rods
- 4x lm8uu bearing
- 4x lm8uu housing
- 2x 623zz bearing
- 1x endstop
- 1x nema14
- 1x gt2 pulley
- printed parts: `Y_endstop_holder`, `Y_moving`, `plate_holder`, `Y_belt_holder`, `Y_end` and `Y_motor`
- 8x magnets: cylinder OD = 8mm, height = 3mm
- 16x M4x10 (lm8uu housing)
- 3x M3x30 (belt holder + 623zz bearing)
- 4x M3x12 (motor)
- 2x M3x10 (endstop)
- 9x M3 washers
- 5x M3 nuts

Assemble the Y axis as in Figure 2. Start by the `Y_moving` part, screw the lm8uu housing (with the lm8uu in). For the belt holder, be careful with the direction, it can be changed when the belt will be tightened during the assembly to the frame if needed.

Pull the magnets in the `Y_moving` part and the `plate_holder` part, be careful with the direction.

Insert the rods in the lm8uu housing. The rods must then fit in the `Y_end` and `Y_motor` parts, if not, drill a 8mm hole.

The endstop must be screw to the `Y_endstop_holder` and can be put apart for the moment and will be set during the assembly.

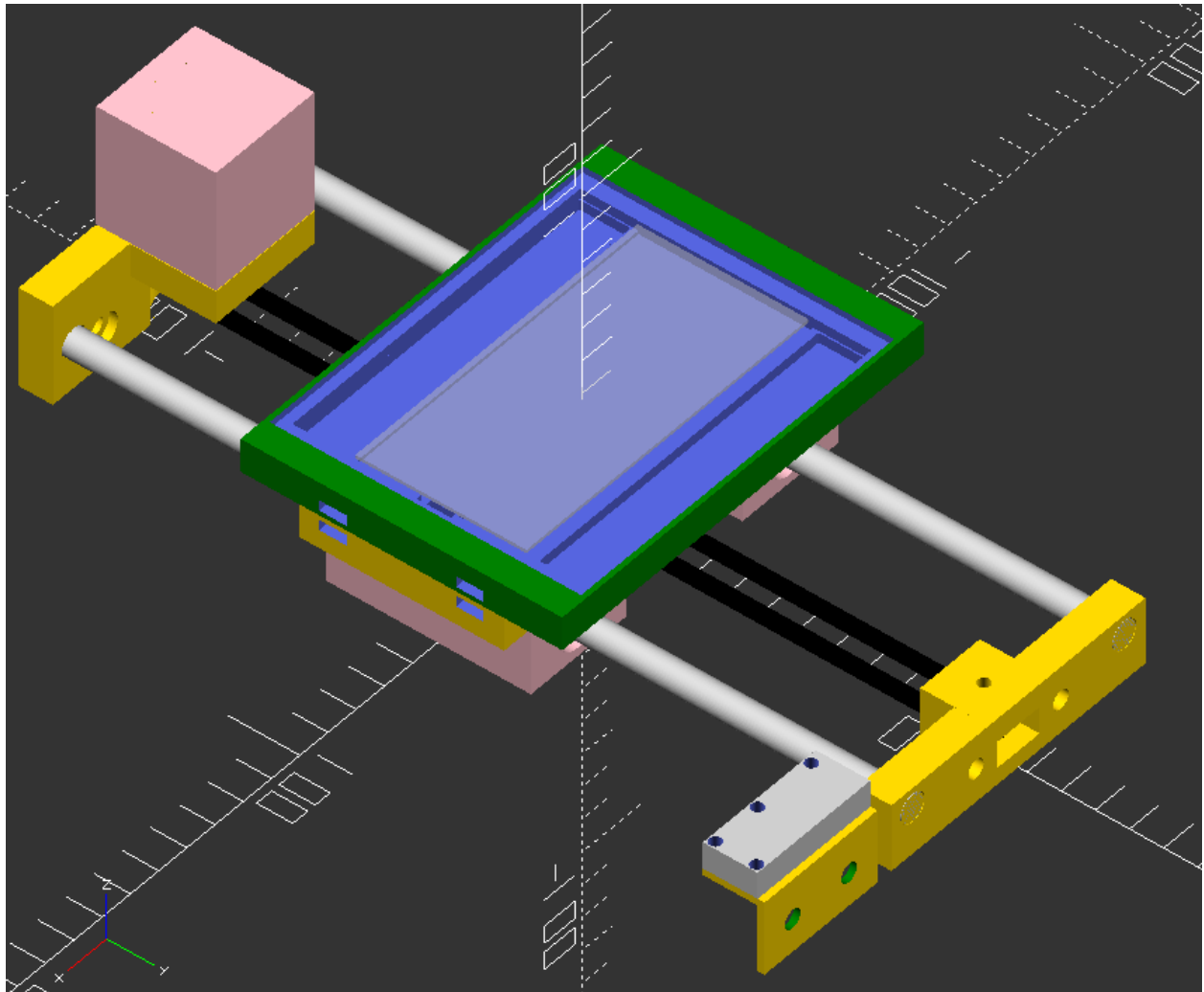


Figure 2: Y axis.

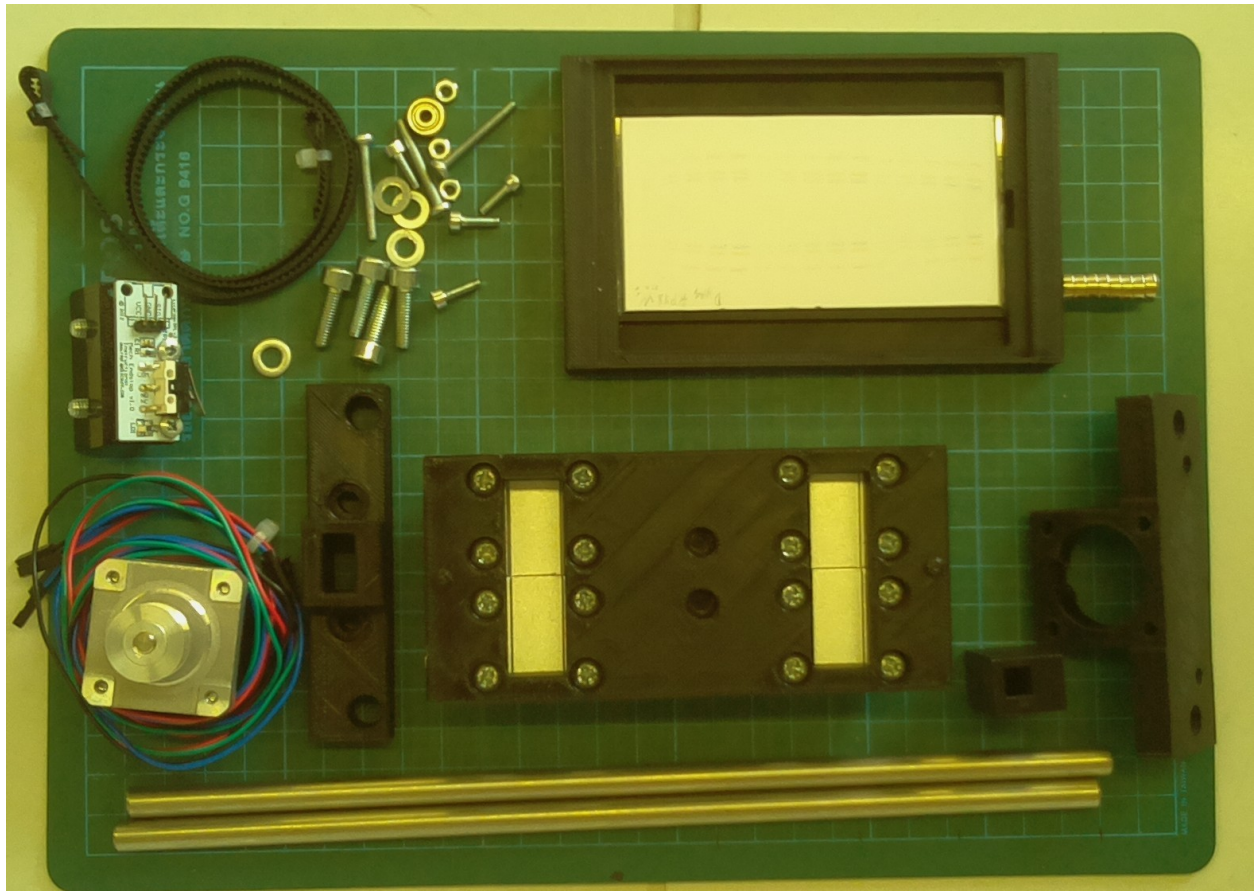


Figure 3: Y axis before.

X axis

SCAD file: `full_view("X axis")`.

Needed parts:

- 2x 8mm rods
- 4x lm8uu bearing
- 2x 623zz bearing
- 1x endstop
- 1x nema14
- 1x gt2 pulley
- printed parts: X_motor, X_end, X_moving, HP_C6602_holder_holder, X_end_stop_holder
- HP C6602 holder
- 2x M4x20 (HP_C6602_holder_holder)
- 2x M4 nut (HP_C6602_holder_holder)
- 1x M3x30 (623zz bearing)
- 4x M3x10 (motor)
- 2x M2*10 (HP_C6602_holder_holder)
- 1x M3 nut (623zz bearing)
- 2x M2 nut (HP_C6602_holder_holder)
- 4x M3 washer (motor)

Assemble the X_motor, X_end and X_moving parts as for the Y axis. Note that the X_moving and

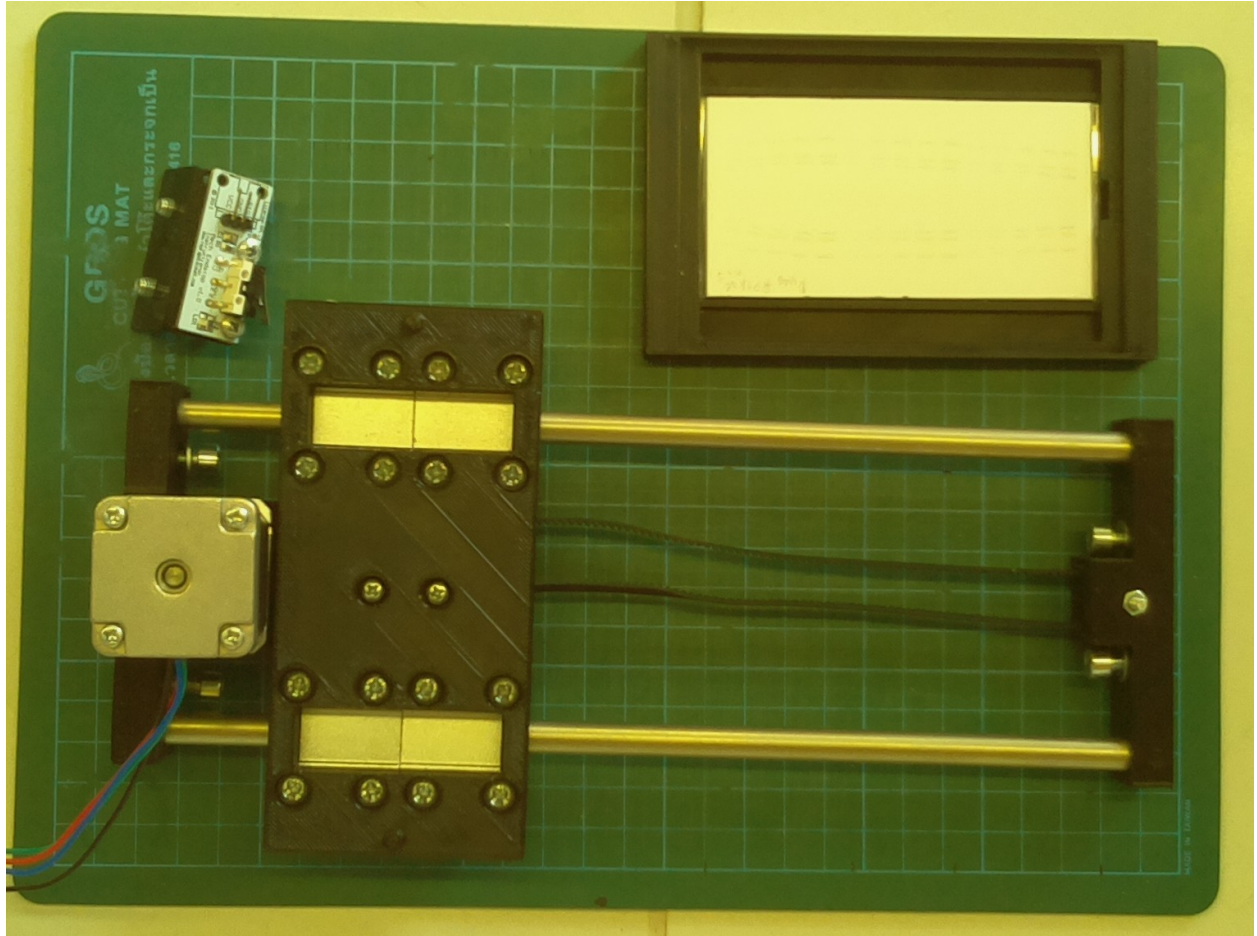


Figure 4: Y axis after.

`X_end_stop_holder` are taken from the other designs, the STL files are in the STL folder separately but also in the main STL file, taken from [prusa i3 rework](#).

The link between the `HP_C6602_holder` and `X_moving` is made by the `HP_C6602_holder_holder` part, use M4 and M2 screws and nuts for this. M4 link to the `X_moving` and M2 to the `HP_C6602_holder`.

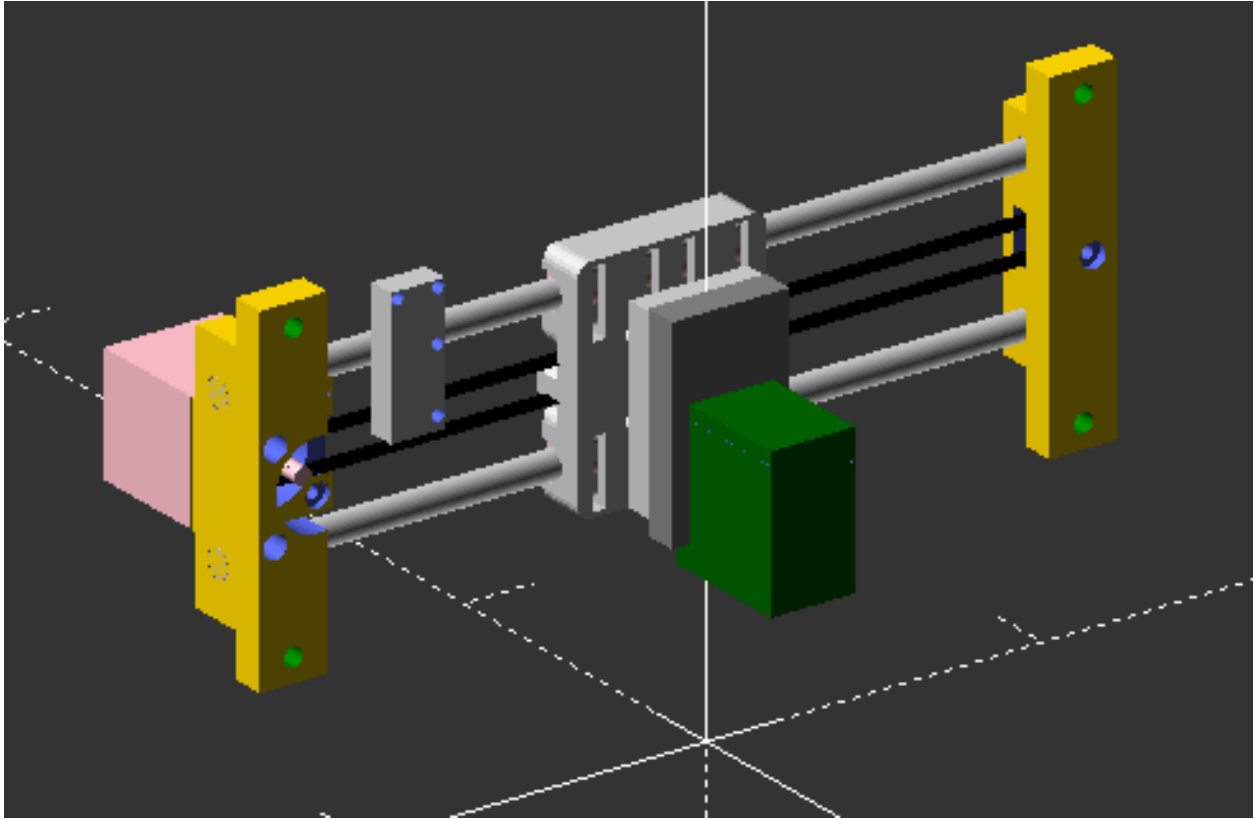


Figure 5: X axis SCAD.

Axis assembly

SCAD file: `full_view("assembled")`.

Needed parts:

- X axis
- Y axis
- frame
- 10x M5
- belts

Position the X axis and Y axis as in Figure 8. The belt must be tightened and blocked with zip-ties.

The X axis must be position as low as possible while avoiding collision with the plate holder. The Y axis will be positioned later when the cartridge will be able to fire.

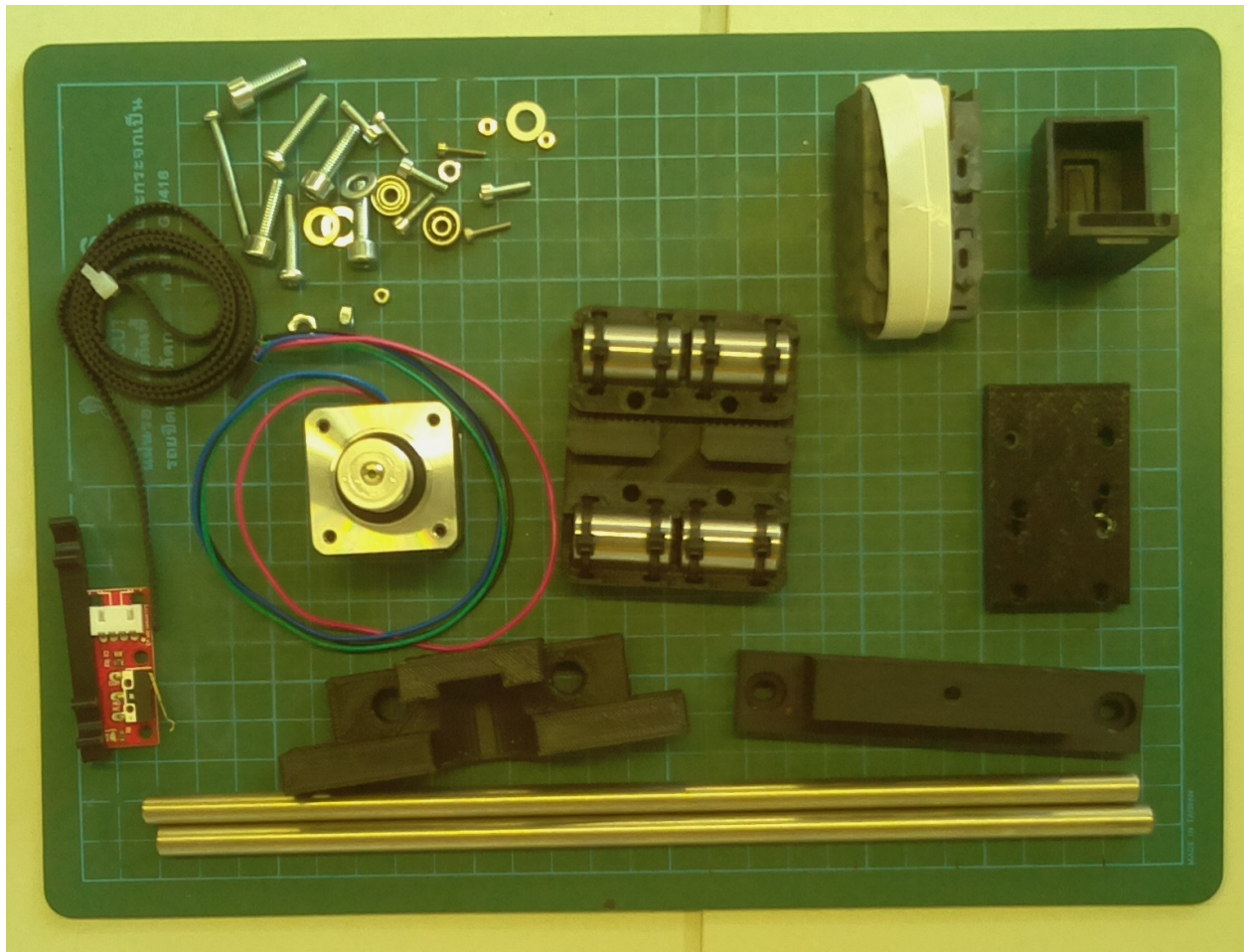


Figure 6: X axis before.

Electronics

SCAD file: `full_view("elec")`.

Needed parts:

- printed parts: `elec_holder`, `camera_holder`
- Raspberry Pi
- Arduino mega
- Ramsp 1.4
- 2x A9688 motor drivers
- LED strip
- Prototype soldering plate
- 3x NPN transistor 2n3904h33
- Breadboard Jumper Wires Ribbon Cable
- 6x M3x20 screws
- 8x M2x10 screws
- 8x M2 nuts
- 255 nm LEDs

Solder the inkshield board as shown on the [website](#).

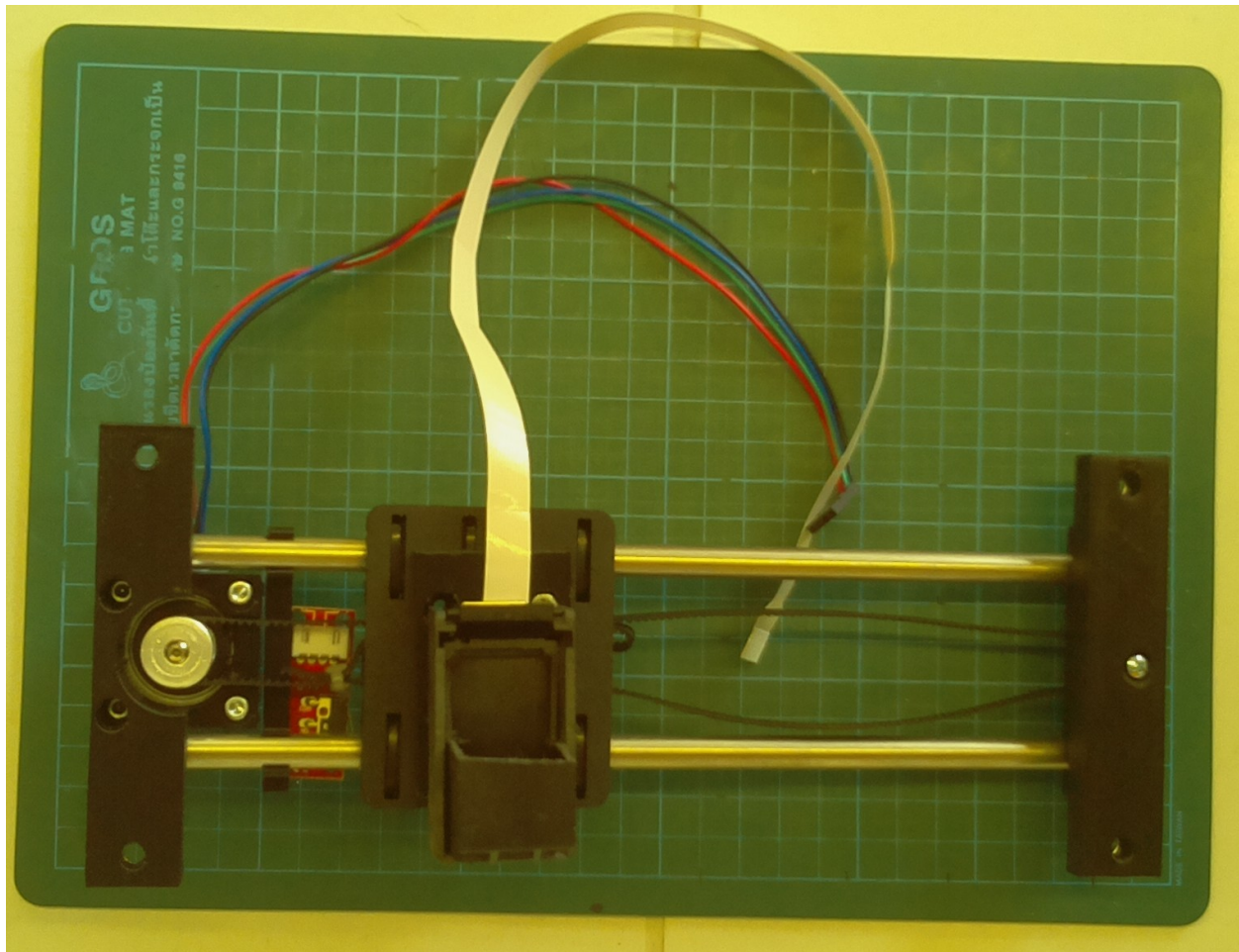


Figure 7: X axis after.

Mount the UV LED to an appropriate board (screw distance 35.5mm, hole for camera lens centered in the middle) and connect it to the `camera_holder`.

Follow [this instructable](#) to make the electronics for the RGB LED strips. The LED strips must be attached with double sided tape to the `camera_holder` part.

Screw the Arduino/Ramps and inkshield boards to the `elec_holder` part and mount it in the frame as in Figure 9. Plug the motor driver in the X and Y axis positions.

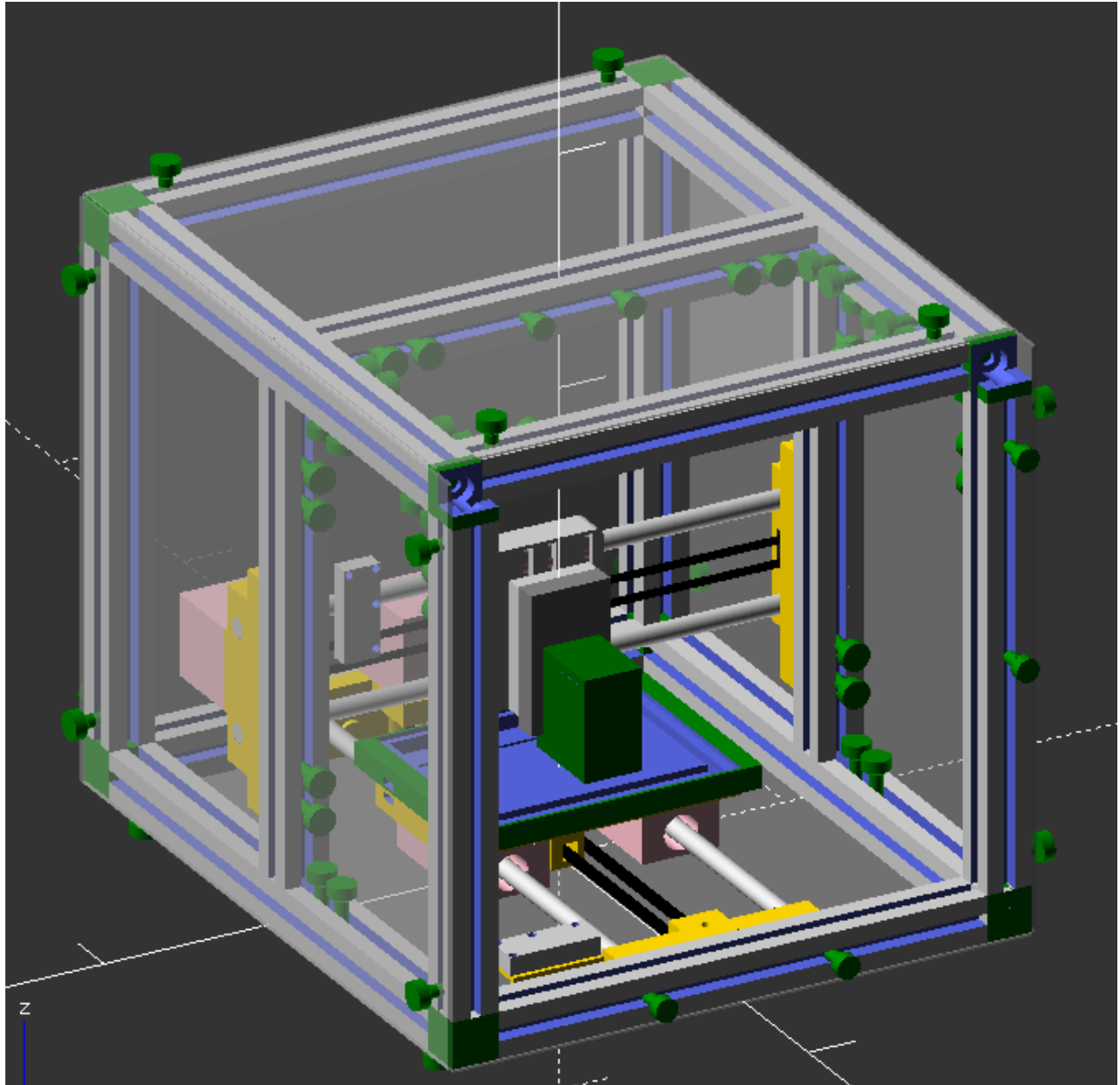


Figure 8: Assembled.

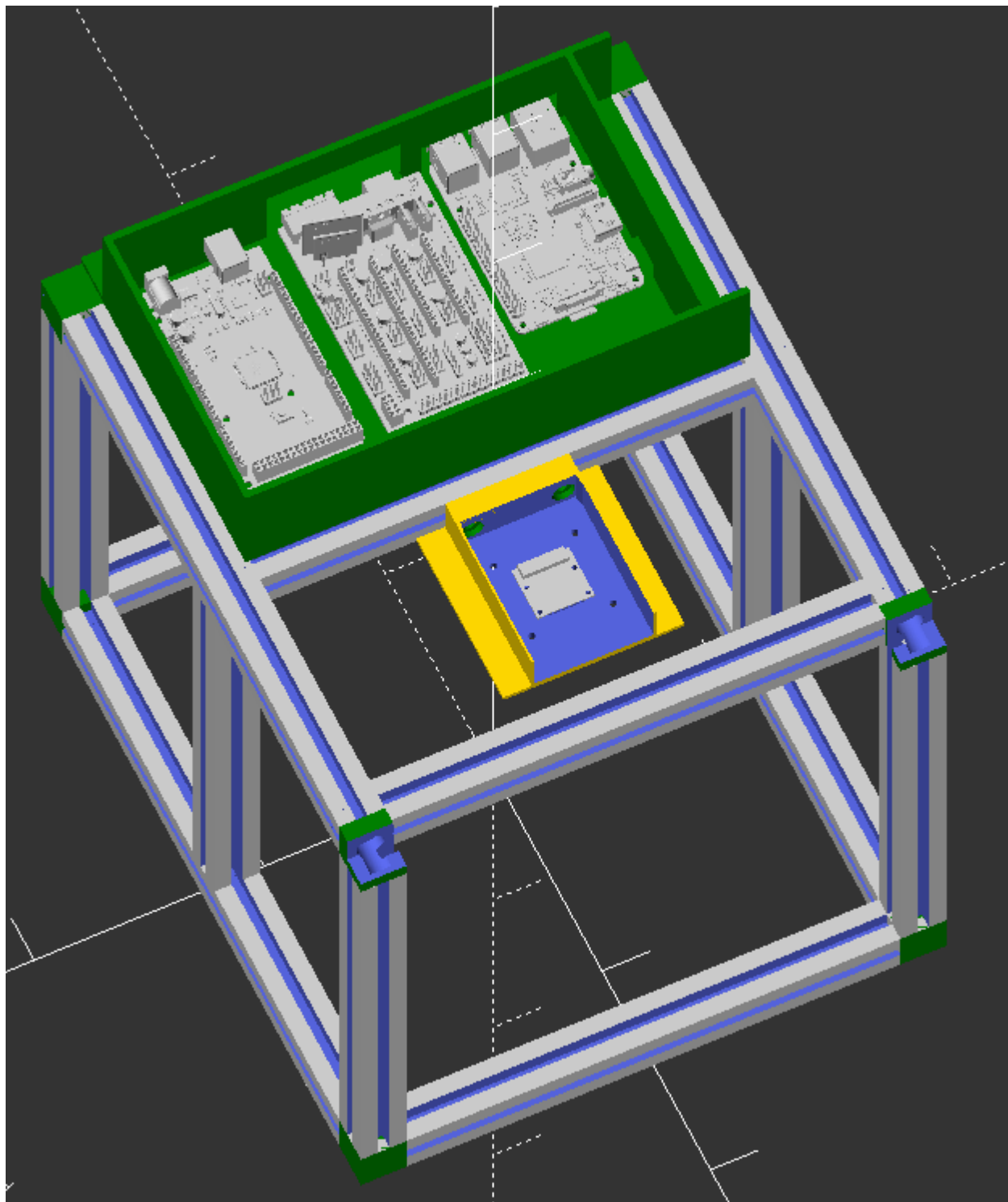


Figure 9: Electronic SCAD.

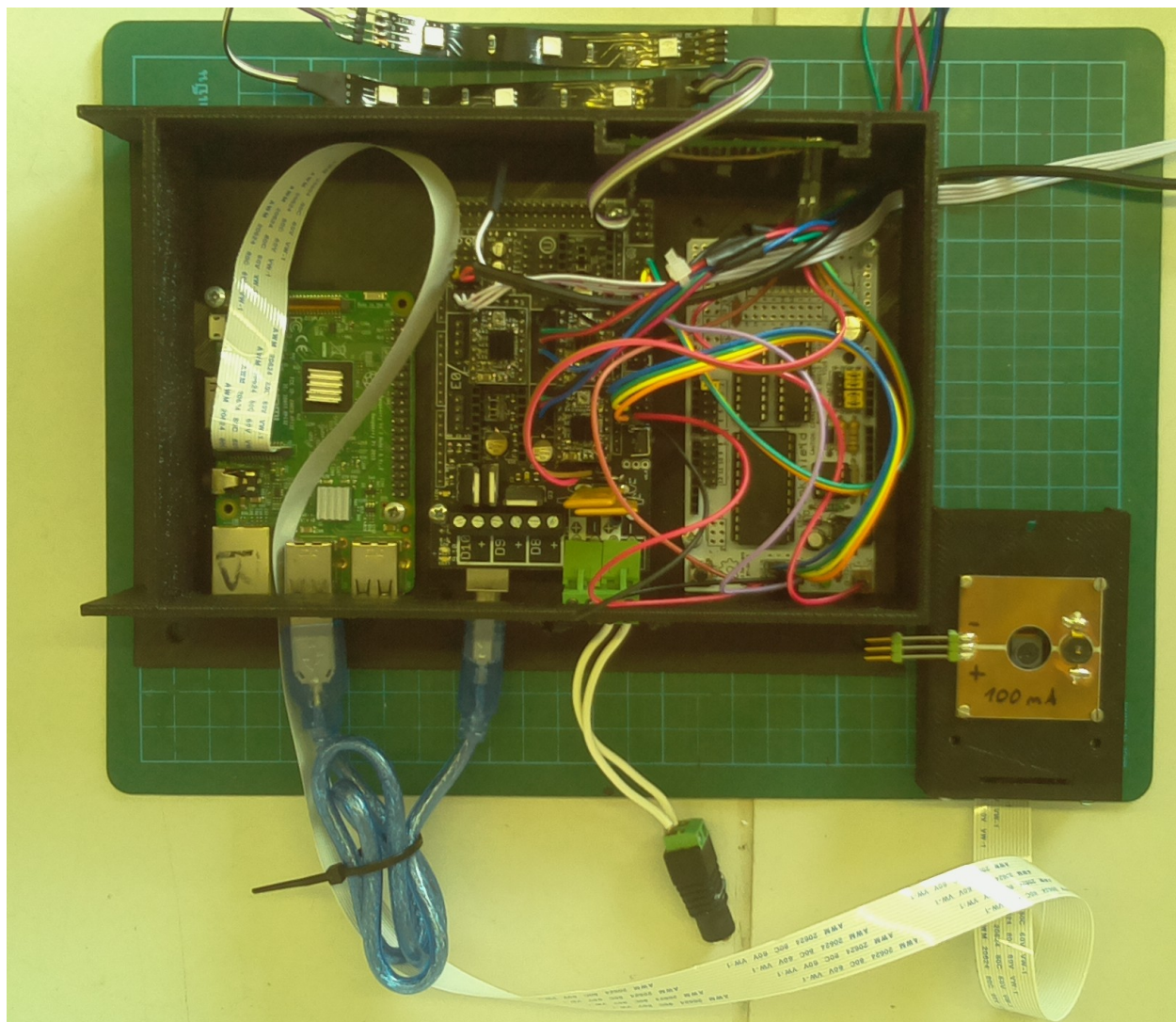


Figure 10: Electronic.

Reversing input power, and inserting stepper drivers incorrectly will destroy electronics.

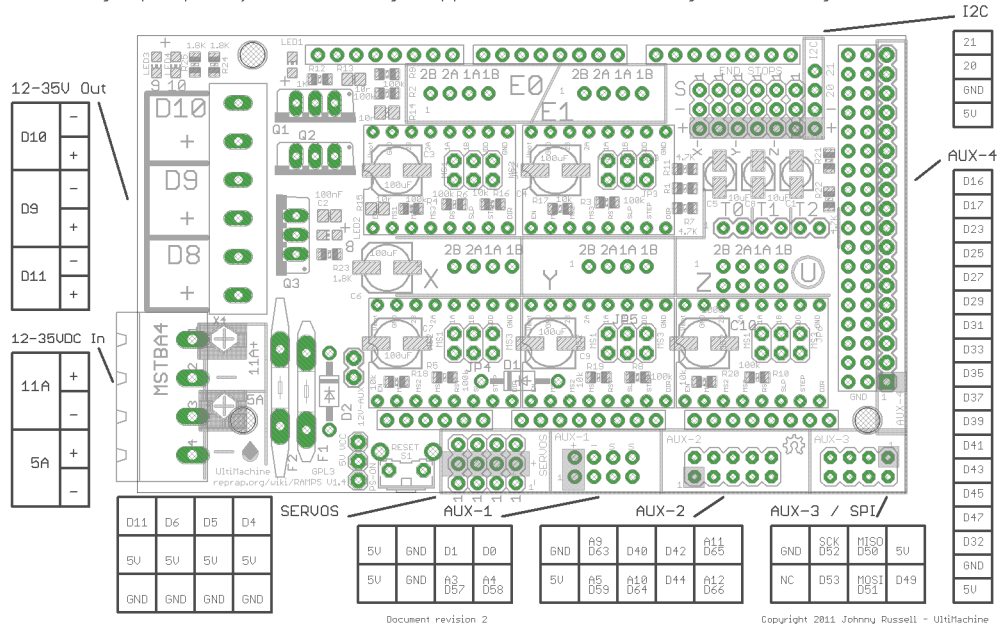


Figure 11: Ramps 1.4 schematic.

Wiring

The schematic of the Ramps 1.4 is presented in Figure 11.

Motors and endstop

Wire the Ramps 1.4 board as in the Figure 12. Only the X and Y motor are used, including their respective endstops. Note that this figure is extracted from a 3D printer wiring.

LEDs

Wire the RGB LED as in Figure 13 with the difference that “Red” = 44, “Green”=66, “Blue”=64. Again, go back to [the instructable](#) if needed and refer to Figure 11.

For the UV LEDs, a separated power supply is necessary and it cannot be controlled by the software. If a solution is found, the pin 59 is reserved for the 255 nm LED.

Inkshield board

Similarly to the RGB LEDs, connect the inkshield board with ribbon female-female cable. The auxiliary input must be used and the connections are: “A”= 11, “B”= 6, “C”= 5, “D”= 4, “pulse”=63. The 12V, 5V and ground can be taken from the Ramps and RGB LED board.

RepRap Arduino Mega Pololu Shield 1.4

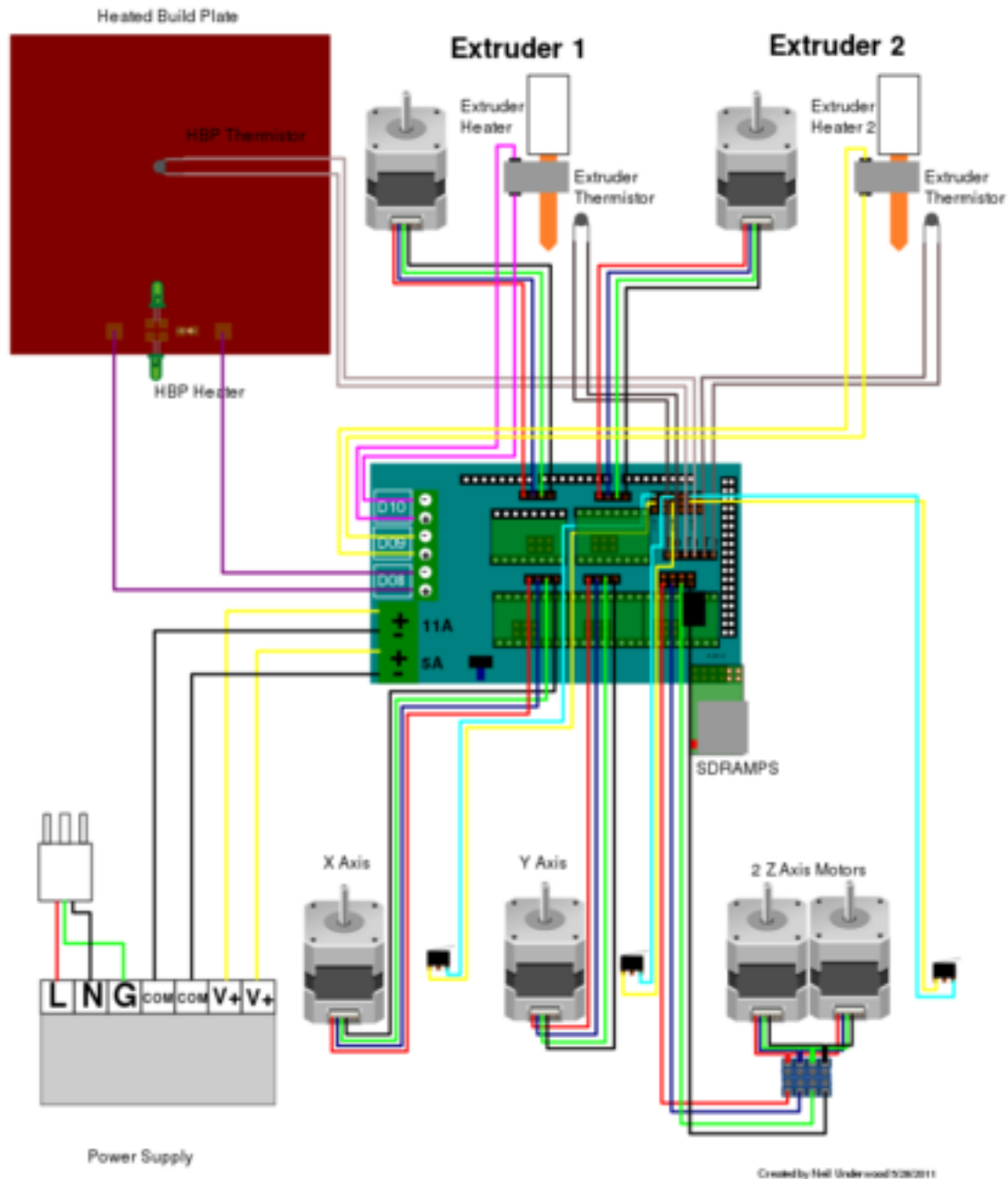


Figure 12: Motor wiring.

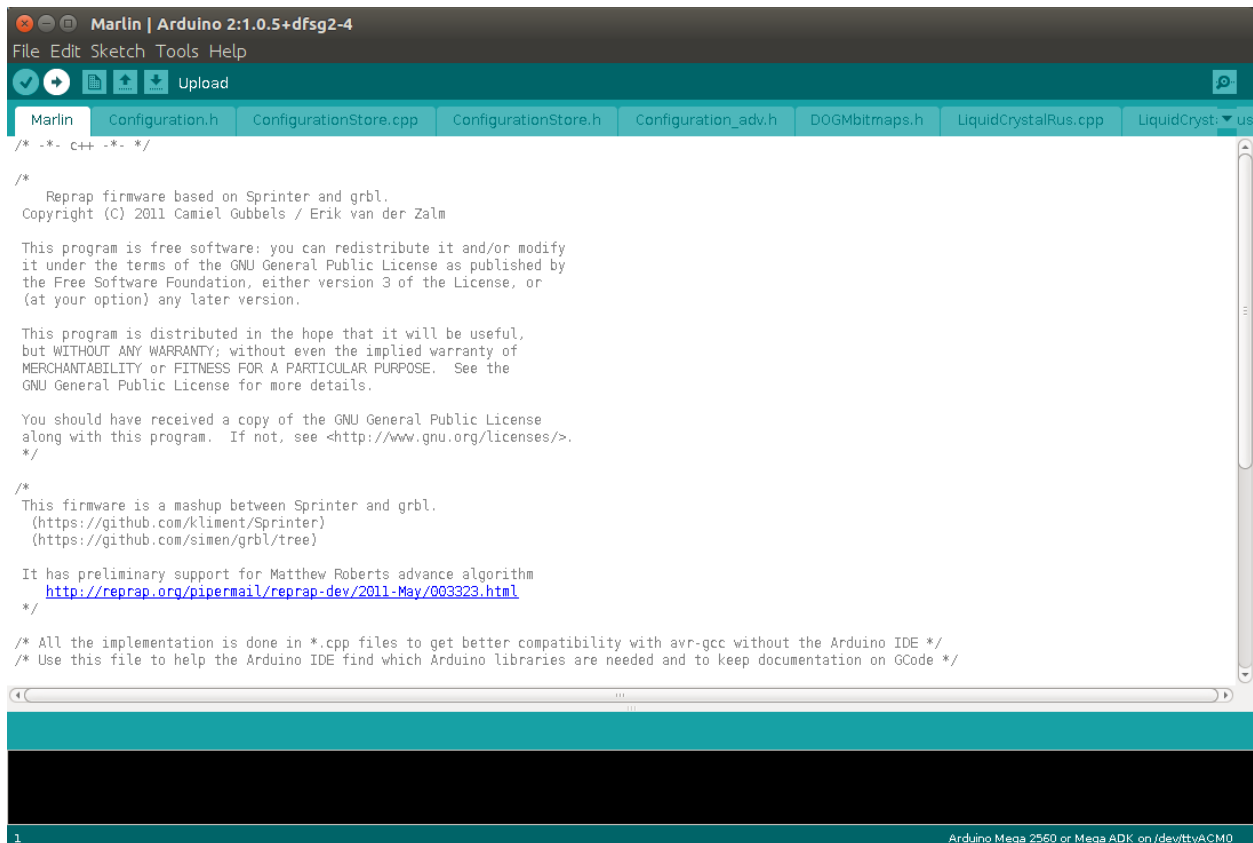


Figure 14: Marlin upload.

Calibration

The only calibration necessary is to position correctly the Y axis. To do so:

- Open OC_manager by going in your browser on the IP address of the Raspberry Pi,
- Connect the board
- In the Fine control tab (Figure 15): home the X axis and the Y axis, (if the motor are not in the good direction, turn off the full system and simply reverse the cable on the ramp)
- With an ink cartridge in the cartridge holder and a plate in the 10x10 cm plate holder, click on **fire selected nozzles**. The 12 ink droplets must be on the plate in the top left corner. Move the Y axis if necessary.

This calibration procedure is not enough and is a known problem, ideally, a complementary procedure should be inserted in the code to add a bias in X and Y to every movement. The 3D printing environment was done to have reproducible movement **during the print** and not between print.

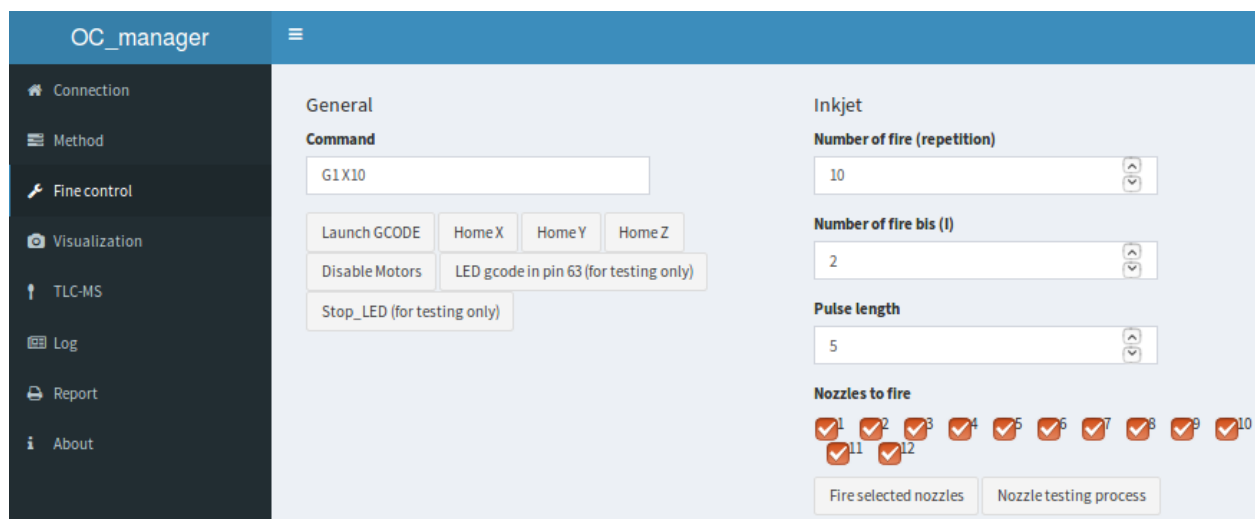


Figure 15: Fine tuning OC manager.