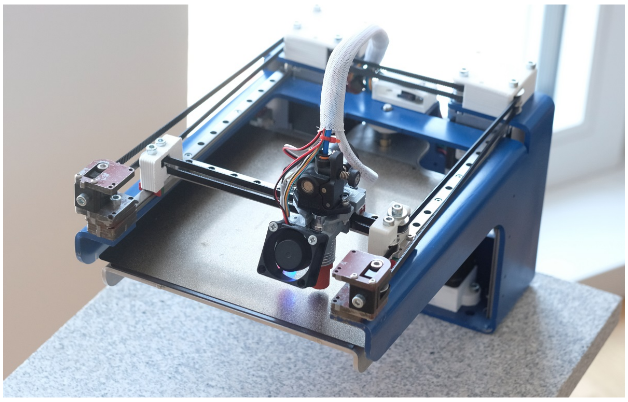
xOneFrame

A cheap superfast 3D Printer

Several design developments are going on. I'd like to combine some parts of them. Build a fast (and not too expensive) 3D Printer for my personal use.

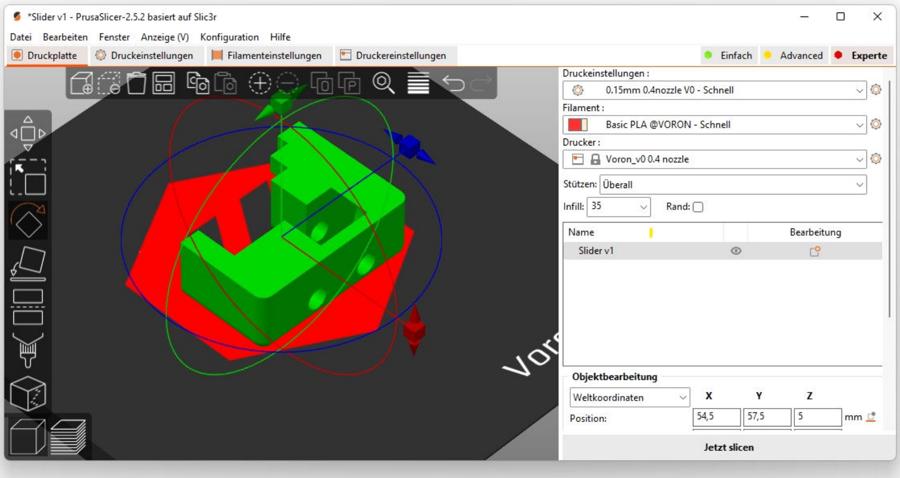


Contents

1 Getting started → first Print

1.1 Files and Preparation

With a given 3D Printer, we need some 3D Files (`.3mf` or `.stl`) to print. If you dont know how to get them, have a look at `www.thingyverse.org`. The files have to be processed for this printer to be printed, here the Prusia-Slicer is helpful. From the Printer is compatible to the *Voron 0.1*.



Put your `.3mf` or `.stl` file to the slicer, position it, so it does not have (too many) overhangs, choose the layer hight (finer is slower) and use the `slice` Button, to get the processed data with each layer for the printer to print. The new `.gcode` File contains each head-movement, each extrusion ammount, each single line of plastic to be added to the part.

1.2 Connect to the Printer

The printer (or it's main controller board, a Raspberry Pi mini computer) has a WebInterface, a WebPage where you can control the printer via the `klipper` Software. This WebInterface Page can be accesed with a browser on your computer. To get to the WebInterface, you have to connect your computer to the same net as the Raspberry Pi. I normally use my own WiFi network, and configure the Raspberry Pi to connect to that network.

WiFi Net: `PrinterNet`
WiFi Password: `geheim`
WebInterface: `www.klipper.local`

1.3 Printer

At startup the Printer is doing its homing–sequence (find the position of the print-head) and the bed–leveling (making shure, the bed–hight is known down to a fraction of a millimeter). The Printer has a WebPage to connect to. A few moments after the startup, the WebPage should be accessable. Go there, and use the *Upload and Start* Button.

2 Concept

As little moving mass as possible, as much stiffness as acchevable. I do like the P P Type (or H Type) Belt drive, with the Motors sitting in one corner. Of course the extruder motor is preferred to be stationary[1]. The H Type Belt drive moves the whole X-Axis. Some printers use carbon materials to make it lighter. My approach is to make it short. Then I dont need carbon material, the MGN9 Rail is lighter, everything is smaller. Only disadvantage: The printer can't print large parts.

With the short X-Axis it's possible to skip on the support as the Rook-Concept[2] does. There I dont like the stress on the corner-pieces. I dont like the idea that it can jam up, if the corner pieces dont hold the force correct. In my opinion it's better if there is a direct connection between both MGN9 – Y – Sliders. And with the short distance to cross, a printed construction should be fine.

3 Forces – Acceleration

It should go up to 1000 $\frac{\mathrm{mm}}{\mathrm{s}}$. With an acceleration–distance of 5 mm I'd get

$$s = \frac{1}{2}at^2 = \frac{1}{2}a\left(\frac{v}{a}\right)^2 = \frac{1}{2}\frac{v^2}{a} \tag{1}$$

$$\Rightarrow a = \frac{v^2}{2\,s} = \frac{1\,\frac{\mathrm{m}^2}{\mathrm{s}^2}}{2\cdot 0.005\,\mathrm{m}} = 100\,\frac{\mathrm{m}}{\mathrm{s}^2} \approx 10\mathrm{G} \tag{2}$$

that's more or less the reason why i dont want the X–MGN9 Rail as single bridge between the Y–Sliders; But some bridge to fix the rail on.

With 100 $\frac{\mathrm{m}}{\mathrm{s}^2}$ and 0.5 kg weight of the X-Axis-assembly I need

$$F = m\cdot a = 0.5\,\mathrm{kg}\cdot 100\,\frac{\mathrm{m}}{\mathrm{s}^2} = 50\,\mathrm{N} \tag{3}$$

and with the $r = 1\,\mathrm{cm}$ traction wheel thats

$$M = F\times r = 50\,\mathrm{N}\cdot 0.01\,\mathrm{m} = 0.5\,\mathrm{Nm}. \tag{4}$$

Here the I of the motor itself is not considered. The regular Nema17 motors (maximal 0.6 Nm) are thus just on the limit.

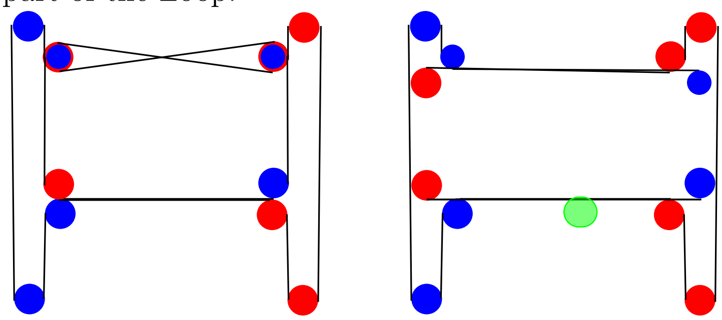
With 3000 RPM and $r = 1\,\mathrm{cm}$ the speed would be 1500 $\frac{\mathrm{mm}}{\mathrm{s}}$. In the upper RPM regime I'll never get the full torque of 0.6 Nm. Thus I need bigger motors. Possibly bigger pullys with $r = 15\,\mathrm{mm}$ on the motor-axis.

[1] With some disadvantage when printing flexible fillament

[2] YouTube → Rook 3D Printer

3.1 CoreXY Variation

There are some different variations. The Voron 0.1 or the Rook work with the Motors in different part of the Loop.

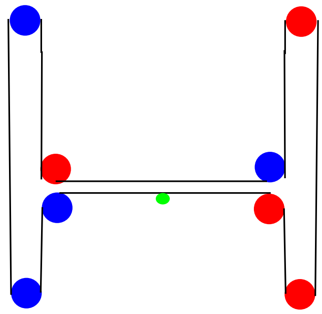


(A) Position of the Crossing (front or back) → I want an accessable front, thus the crossing in the back. (B) Position of the Motor (front or back) → same as A. That implies, that i have to have an extra ideler pulley. (C) Position of the short P Parallel.

I want the resitance of the rollers to be (more or less) equal distance each side between the motor and the nozzle. So I choose the 3:2 roller distance as shown here. A 1:4 positioning of the motor on the other top-part of the P would also be possible, but then the friction in one direction would differ from the friction in the other direction.

3.1.1 Less Rollers

In Youtube 3d printed automatic a different CoreXY Scematic is shown, a simpler one.



3.1.2 Forces on the Rail

With CoreXY we need a stiff frame. The joint between x– and y–axis is relevant, and can take (with mgn09) a maximal load of $M = 7.5\,\mathrm{Nm}$. To get the $\approx$ 50N of acceleration–force, the x–axis can be only 15 cm away from the opppsit roller.

The MGN9 fits though nicely with my 50 N design with about 15 cm X–travel.

3.1.3 Travel and Speed

Nema 23 have plenty of torque. And plenty of inertia. So: A bigger spur gear seems to be the way to go. That helps also with the 3000 RPM Limit of stepper or servo Motors.

I have about 200 Steps per rotation, and want a resolution of 0.4 mm. My radius is thus defined as: $0.4\,\mathrm{mm}\ = x = \frac{U}{200} = \frac{r \cdot 2\pi}{200} \Rightarrow r = 0.4\,\mathrm{mm} \cdot \frac{200}{2\pi} = 12.5\,\mathrm{mm}$. So ... my resolition (in the range of a nozzle width) is defining my spur gear radius. Microstepping does help, but not that much.

3.1.4 Inertia

For travelling at $1000\,\frac{\mathrm{mm}}{\mathrm{s}}$ the motors need about 0.5 Nm as calculated in (). With some r=15 mm pully. Thus some angle–acceleration of $\alpha = \dot{\omega} = \dot{\left(\frac{v}{r}\right)} = \frac{100\,\frac{\mathrm{m}}{\mathrm{s}^2}}{0.015\,\mathrm{m}} \approx 6500\frac{1}{\mathrm{s}^2}$

$$M = I \cdot \alpha \tag{5}$$

Und die Trägheit I ist bei einem Nema17 mit $38\,\mathrm{g} \cdot \mathrm{cm}^2$ angegeben, wohingegen sie bei einem Nema23 mit $400\,\mathrm{g} \cdot \mathrm{cm}^2 = 400 \cdot 10^{-7}\mathrm{kg} \cdot \mathrm{m}^2$ angegeben wird. Das heisst, nur um den Motor zu beschleunigen brauche ich:

$$\begin{aligned} M_{17} &= 6500\frac{1}{\mathrm{s}^2} \cdot 38 \cdot 10^{-7}\mathrm{kg} \cdot \mathrm{m}^2 && (6)\\ &= 0.025\,\mathrm{Nm} && (7)\\ M_{23} &= 6500\frac{1}{\mathrm{s}^2} \cdot 400 \cdot 10^{-7}\mathrm{kg} \cdot \mathrm{m}^2 && (8)\\ &= 0.26\,\mathrm{Nm} && (9) \end{aligned}$$

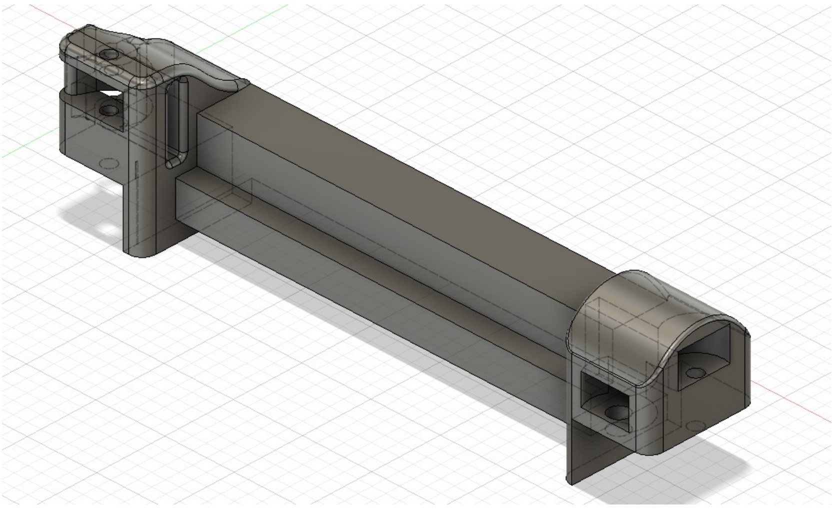
With the torque gain of the Nema23 that should be easily obtainable.

4 Frame & Hardware

The Motors in the back are practical. With the standard tensioneing system of the Creality CR10 and some 2020 Extrusion, i get an easy build with a big piece of framing printed in the back.

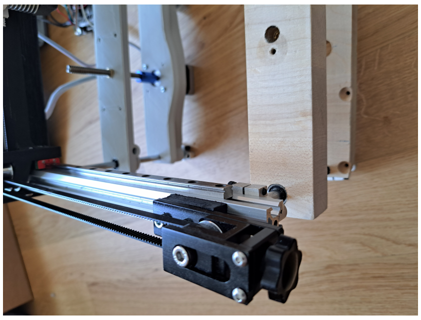
4.1 OnePiece

Both motor holders left and right can be printed in one piece. The X–Axis can be printed in one piece (not each joint separatly as other designs suggest). Thus the name.



4.1.1 Wood

As the 3D Printer to print the Z-Beams had some technical difficulties, I tryed wood. And: No chance the hand-held cuts fit. I could not get rid of the binding problems with the linear rail.



So either my tools or my skills are to blaim. With the 3D printed Z-Beam i did not have any problems.

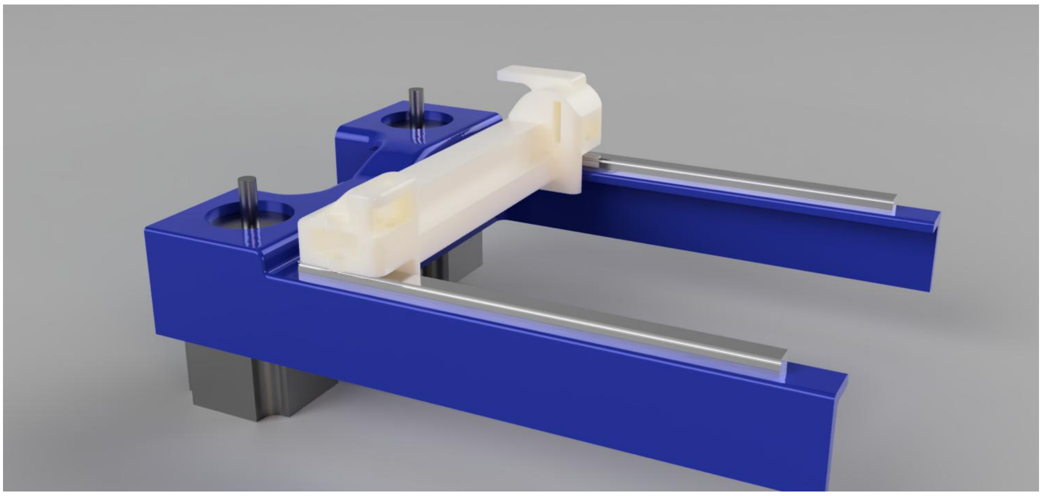
4.2 Housing

Either a Zarges Box ... but well .. it would be nice to present it better.

ELEPURE Vitrinen gibt es 35x25x15, das könnte als Deckel für eine Box dienen.

4.3 Metal Molding

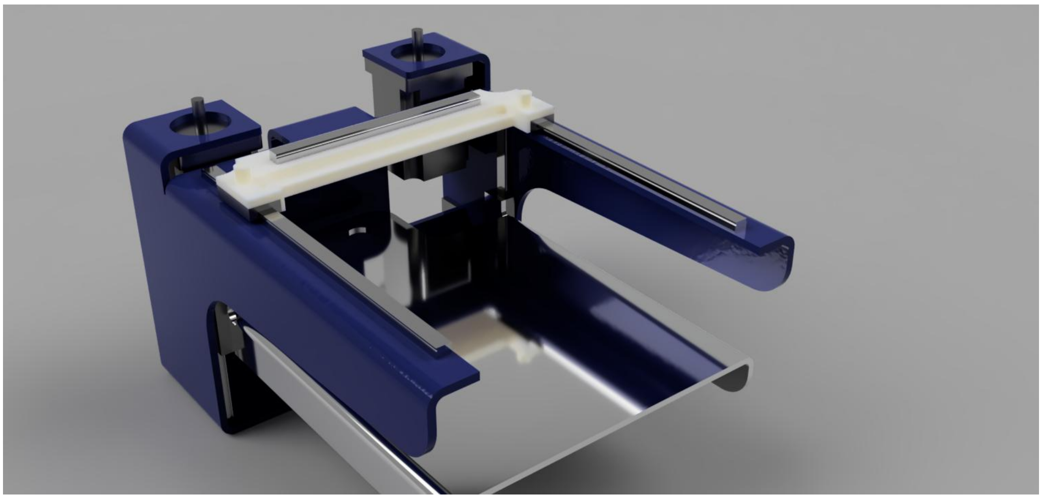
I need weight (counterweight). And the aestetics of an italian ham slicing machine are sort of an ideal. The frame can be cast out of Zamak, an Zink Aluminium alloy, that should be easy to work with.



Oh, and of course: Positioning everything on a granite slab looks good. And prevents vibrations to propagate to the surrounding.

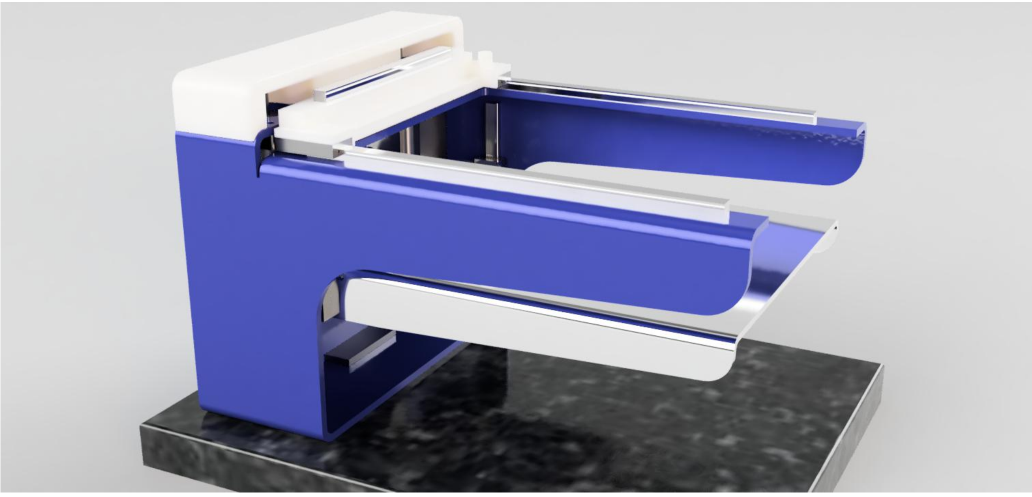
4.3.1 Metal – Bending

The Casting Company did not answer my eMail (about preliminary cost estimates). Well. Blexon does calculate prices. So the bending–option is given.

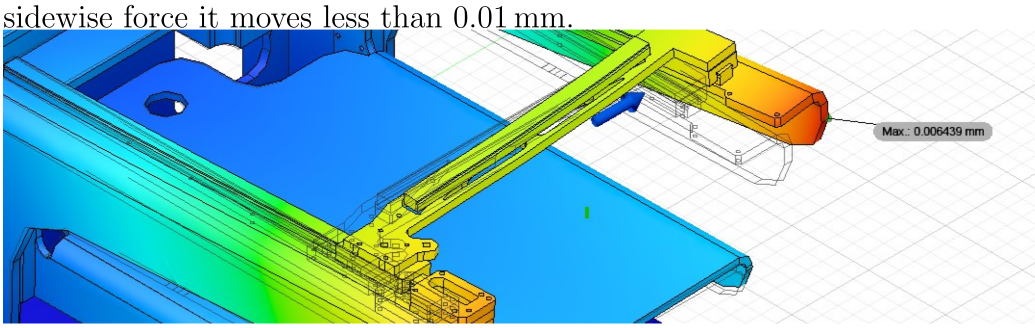


I was planning on some bed swiveling possibilities with auto bedleveling. But with some bed–sensor and the mesh, a more-or-less ok'ish bed works all fine, I dont need auto leveling when I can compensate for uneven beds.

First thought was to position the Motors on different hights (as the driven belts are on different highth). But then the rollers have to be 20mm in front of the motors. I'd loos 20mm of `y` travel. Dont want that. So the motor-bearings have to take a bit of a load, as one motor is connected to the belt with 10mm distance.



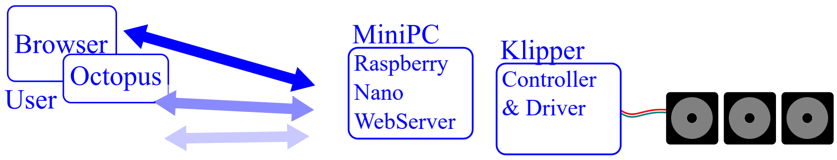
It seems to be stiff, the base-frequencies are above 130 Hz. With 10 N of sidewise force it moves less than 0.01 mm.



5 Electronics

At the moment i'm working with a BigThreeTech SKR Mini E3 V 3.0 just as it's not too expensive and seems to be working. The maximum of 2A is a bit on the weak side for Nema23, still it should be fine for 1.2 Nm.

The main point for the board (as the alternative would be a self-designed ESP32 variant) is the availiablity, the software support. And: I want all those plugs to be connected directly. I dont want any cable mess.



5.0.1 Connection Raspberry → Klipper

Ich habe das SKR mini E3 v 3.0 rumliegen. Dort entsprechend die Klipper Version vom GitHub rüber gespielt. Dann sehe ich entsprechend auch (wenn ich's per USB anschliesse) mit:

```
pi@KlipperPi:~ $ lsusb
Bus 001 Device 002: ID 1d50:614e OpenMoko, Inc. stm32g0b1xx
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
```

Mittelfristig will ich's natürlich auf UART Betrieb umstellen ... aber das hatte komische Probleme verursacht. Das Kabel stört nicht wirklich, da hab' ich genug andere Themen zu lösen.

Activating UART on the RasPi ...

5.0.2 Klipper Config

Klipper has to be configured. The CoreXY has to be set, each motor connection has to be described. The Distance per rotation has to be set. The Homing procedure has to be configured and has to be tested.

I choose sensorless homing. I dont like the extra harware for endstops, and the motor drivers support the current monitoring. Thus the power consumption is utilized to recongnize when the carriage hits frame. This is sufficient to get a known position for the motors after each startup.

5.0.3 BL–Touch

I want a prooven bed-leveling process, even if it's a bit slow. Faster (and more reliable) methods are developed, they can be implemented in a future version. BL–Touch can be connected to `Pin PA1` and `PC14` and needs the `printer.cfg` to be modified with

```
[bltouch]
    sensor_pin: PC14
    control_pin: PA1
    x_offset: 0
    y_offset: 0
    z_offset: 5
```

As it's my only Z–Stop, the Z–Section has to be modified.

```
[stepper_z]
    ....
    endstop_pin: probe:z_virtual_endstop
```

Oh, instead of the Touch-Sensor a LoadCell should be used, so the nozzle can measure directly where the surface is. As shown in: Electronic Basics #33.

5.0.4 Raspberry Pi Pico

For the third Z–Motor it's an Option to connect the Pico as a secondary MCU. Tutorial on youTube to get the Pico as a secondary MCU.

```
[mcu]
    # Obtain definition by "ls -l /dev/serial/by-id/"
    serial: /dev/serial/by-id/usb-Klipper_stm32g0b1xx_2700060009504B573531

[mcu pico]
    serial: /dev/serial/by-id/usb-rp2_23456789-if00
```

This way I have the option to put as many extra motors (secondary extruders, third Z–motor or sensors) on the system, as are needed.

5.0.5 Electrical Connections

The Extruder needs some Connections: • HotEnd Heater (2 wire) • Themperature probe (2 Wire) • Extruder Motor (4 wire) • BL-Touch (5 wire). So I end up with a 13 wire umbilical cord, 6 of them have some relevant voltage and current.

5.1 First testing

The XY Section is running without any unexpected problems. The speed is acceptable, without much tuning I do get up to 500 $\frac{\text{mm}}{\text{s}}$ at 2 A current on the Nema23 Motors.

For printing I use some prusia slicer, try with speeds up to 120mm/s so ... just a bit more than with an regular printer. The results look ok-ish. The lead-screw is not working well, I get horribly bad banding. But over all, it's working.

The dimensions are (within 2%) correct, some fine-tuning can be done later. One problem is the nozzle–heatbreak connection: here i did not tighten the nozzle while hot. So: PLA oozes out of every corner and did spill all over the heater. Well ... cleaning it (while heated) worked, but I'd prefer not to do it often.

X-Y–Direction one axis is at the moment wrong. With the current setting, as I change the X–Motor–Direction, the X- and Y-Axis are changed. If I change both directions, both X- and Y- directions are changed. If i only want to change the X-Direction i am stuck at the current setting. so ... the X-Axis is defined as X+Y-Motor-Movement, the Y-Axis as X-Y-Motor movement. The easiest is to exchange the motors. then the X+Y stays the same, while the the X-Y setting is inverted. Seems like the only solution.

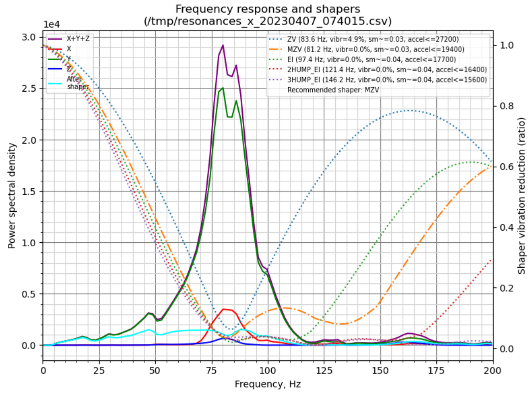
5.1.1 Improvements – InputShaper

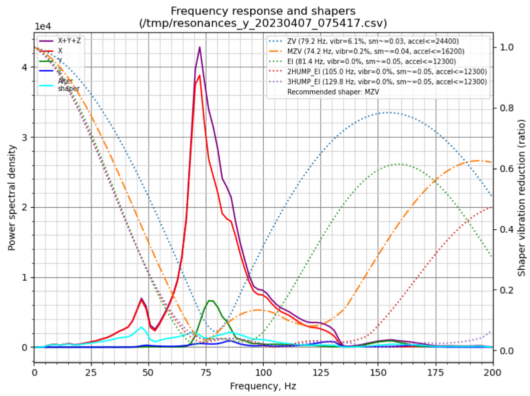
We can measure the resonances. Here the setting as it is today has some 81 Hz Resonance. And ... my Motor gets loud at some frequencies. Not always, but well ... annoying. The measurement can be started with: `TEST_RESONANCES AXIS=Y`. And thus the `printer.cfg` can be modified:

```
[mpu9250]
i2c_mcu: pico
i2c_bus: i2c0a

[resonance_tester]
accel_chip: mpu9250
probe_points:
    50, 50, 20 # an example

[input_shaper]
shaper_freq_x: 81.2
shaper_type_x: mzv
shaper_freq_y: 74.2
shaper_type_y: mzv
```





The Heating PID can be calibrated with `HEATER=heater_bed TARGET=60` and `SAVE_CONFIG`.

5.1.2 Print–Head–Board

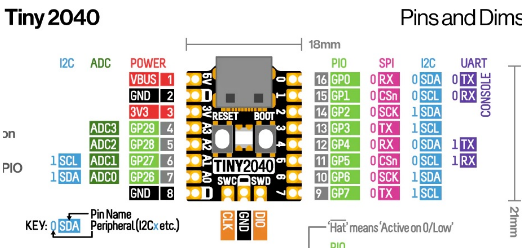
Klipper provides all connectivity. And at the printhead I need a lot of connectivity:

- 24 V Heater (1 Control Pin)
- Thermister of the Nozzle (1)
- Stepper Motor Driver for the Extruder (2)
- Fan Header (1)
- X Limit Switch (1)
- BL-Touch connection (3)
- ADXL345 or MPU-9250 Accelerometer (3)

All these connections can be controled with one RX / TX Uart Wire Pair, to skip all the wiering problems. But the board on the Printhead should be light and small. Well ... needs to be designed, based on the RP2040 or so. As shown on YouTube. Oh, the Design Rules for the RP2040 are in the hardware-design-with-rp2040.pdf.

The i2c0a works with GPIO0 (SDA) and GPIO1 (SCL) with the mpu9250 or ADXL. The other functions should also be able to be moved.

With `/klipper/menukonfig` the MCU can be configured than `make` and copied the klipper.uf2 to the MCU. It can use communication with UART on Pin0 and Pin1. Thus the mpu9250 has to move to another UART.



With the BL-Touch and Accellerometer taking each serveral contacts, the Pinout of the Tiny 2040 has not enough connections. Anyways ... the 24V switching is worth some further electronics. With the Tiny 2040 i can test all connections (and still leave some cables in the Ambilical Chord). With the connections known, I kan think of a board Design that includes a 2040 as well as a Accellerometer, possibly as well as some load cell electronics.

To switch the heating element, a bit of Mosfet and VoltageRegulator logic is nescessary, HeatShield.

5.2 Tuning

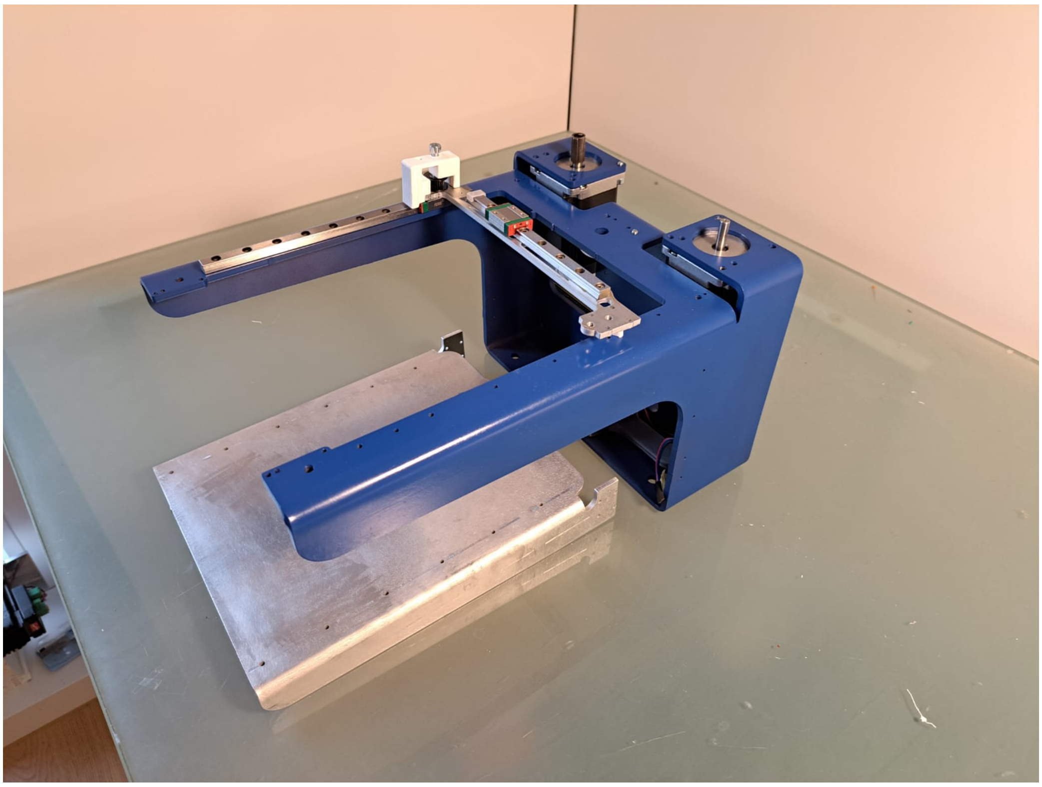
5.2.1 PreassureAdvance

klipper3d.org/Pressure_Advance

With speed change the extruding preassure should be changed. But with a testing tower the indication was, that I dont need preassure advance with this extruder. Well .. that can be easily adjusted later.

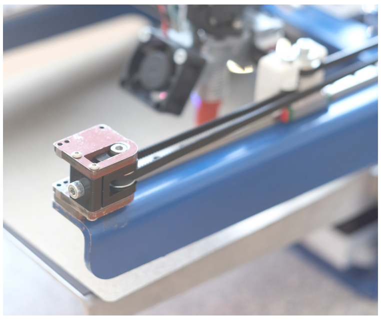
6 Assembly

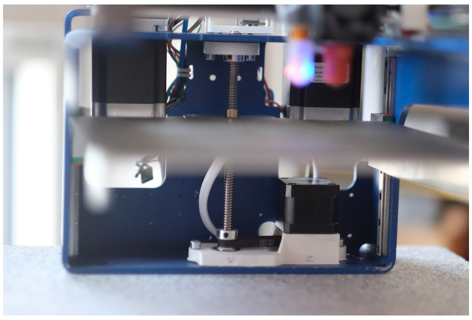
The Frame was produced by Blexon, came to my door. I spray-painted it, etc.



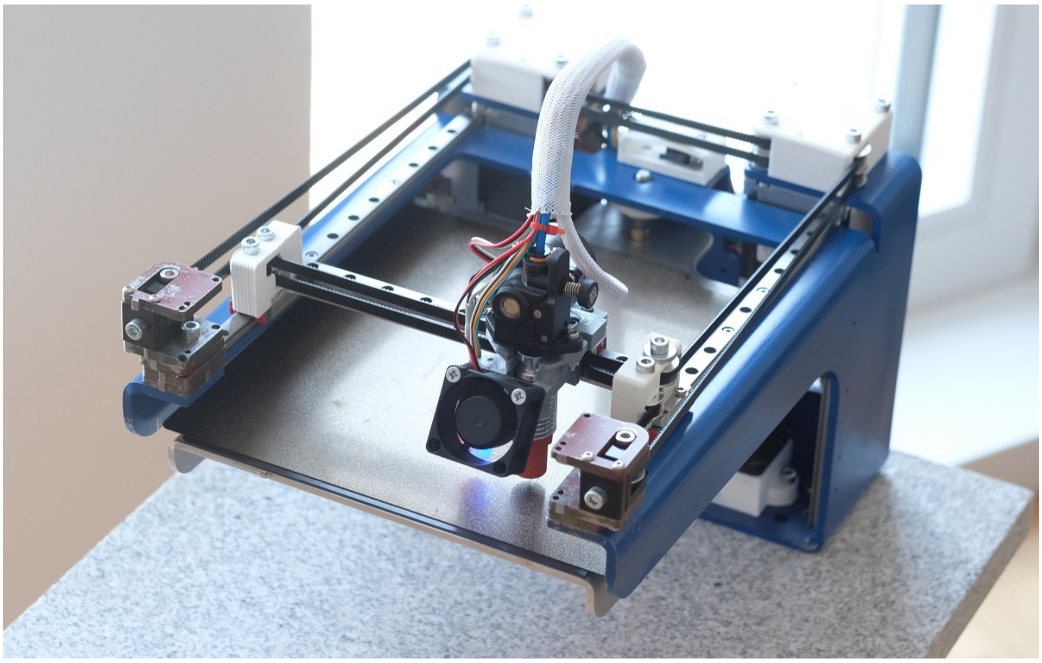
The PLA Parts (including the Printhead Assembly) were printed on the wooden frame.

On the Gantry, the Head and the Core-XY Belts–Rollers are connected, the Front rollers are borrowed from a Ender 3 the Extruder is an off-the-shelve Orbiter. The Z–Assembly uses a T8 LeadScrew, positioned between two 6000RS BallBearings.





The Printer is running. The results are usable. The noise-level is: silent. Each component is working. And each and every component can be improved easily.



I'm looking forward to the improvements you're going to suggest.