**Building a 3D printer**

An Automatic Manufacturing Systems I. project

of Szabó Szilárd Márk (E0UQYJ) student of Óbuda University

Kandó Kálmán Faculty of Electrical Engineering

Purpose and introduction:

3D printing is a relatively new technology as the first 3D printer was made in 1984. It developed in close relation with material science and electronics. Immensely high prices of the technology were pushed down by now as it became more widespread. There are 12 established ways of 3D printing. As of 2016 from edible chocolate through spaceship parts functioning and implanted human organs were printed. Some consider DNA printing also a 3D printing process. Despite these successes the technology still lacks several improvements and is in its infancy.

Requirements:

My main goal is to create a 3D printer, the outcome of a 3D printer development can be the combinations of nothing, destruction and printing. Most of all the parts that a 3D printer prints define the printer design. Lacking resources like high precision machining, time, hard to obtain and/or expensive materials DNA printing and spaceship parts are out of question. I set myself an easier achievement as a requirement for this project by intending to print out a plastic matchbox car that:

* Has to be structurally coherent.
* Accurate to its blueprint to a level of +/- 1 mm/cm axial tolerance.
* In two hour.

If these conditions are fulfilled then I have a “decent” printer. The measure of a 3D printer can be its accuracy, its speed, and the physical properties of the workpiece it can produce. These are a compound result of electronic solutions, mechanical accuracy, chemical reactions and external physical actions, so the goals set above are considered ambitious for a solo project.

Features:

My 3D printer will have a cartesian coordinate system axis set. Will be able to use 1,75 mm plastic filament as the printing material. Will print from a computer using a data connection. Will use a common CAM file format, and a common communications interface. Additionally it will have a detachable print head for the functionality of extension/conversion of the device to a CNC router machine in the future.

Functional description:

plastic filament

CAM file

3D

PRINTER

INPUT:

OUTPUT:

MY DEVICE:

The plastic filament is going to be of PLA (polylactic acid), a biodegradable thermoplastic said to have the scent of fresh baked waffles when heated to its melting point of 173 – 178 °C.

Figures of the setup/architecture:

X axis

Z axis

Y axis

plastic filament

filament holder

heated injector

heated bed

The basic conception:

Electrical and electronic block diagram:

CONTROL

CIRCUIT

G code

interpret-er

PC

CAD/

CAM

software

PSU

X axis

power

electronics

Y axis

power

electronics

Z axis

power

electronics

XYZ Axis end stop

sensors

Injector heater

pow. el.

Bed heater power electronics

X axis

stepper

motor

Y axis

stepper

motor

Z axis

stepper

motor

Injector heater

module

Bed heater

module

Injector heater

sensor

Bed heater

sensor

E-STOP

230V (AC)

5 V DC

12 V DC

230 V AC

230 V AC

DCOM1-5

SENS

0-2

DCOM0

PREG1-5

Implementation description:

The Hungarian market is not in favor of automatic manufacturing systems projects as the price driven approach of my fellow-countryman formed it to its present-day form with the background of no natural resources and week inland industry. Coupled with the illusion of the gentrification by the upsurging service sector we are cheerful receivers of import products. The sells usually don’t deal with so specialized needs and the machine shops only serve firms. So it’s obvious that international deals have to be managed to get most of parts required for success. The drawback of it is that there’s always a percentage of chance that the order will not arrive or will not arrive in time. Like it gets struck in a custom bureau or in the bottom of a container for months. The stated shipping time of [www.ebay.co.uk](http://www.ebay.co.uk) from United Kingdom to Hungary is 5-9 days. I expect some more days till the package gets from Budapest to Dunakeszi. Two weeks of shipping time is not negligible in a 3 month project.

As visible on the figure of the basic conception the build-up is cartesian. Other options were cylindrical, spherical and humanoid the problem with them is that they look straightforward and modern, they more challenging so they wouldn’t fit in my three months time interval. The problem is that if the joints are implemented with motors, the force acting on first motor that carries the weight of the others will be too large, also the swinging mass would be too large causing oscillations. Neglecting these oscillations would result in the robotic arm being more of a toy than a real robot. Oscillations could be eliminated by placing the motors by the first motor and transferring the force through drive shafts and pulley systems but this way the flexibility of the complex force transfer system would cause the nonlinearity and complicated custom mechanical parts would be required. Let’s observe that with the cartesian setup there’s supporting elements holding against the axial forces. While the mentioned other classical kinematic chains transform some of the forces into torsion and there is only support from one side in a form of a joint. The structure between the joints bend and/or oscillate as a result of this. These would result in the deflection of the manipulator which in this case would be the 3D printer head, the extruder. The longer the arm, larger the deflection would be. Staying with the simpler cartesian design, the algorithms behind movement is also simpler and more likely to fit in the three months of development and realization, as there are no rotations and redundancies.

The frame will be built mostly based on threaded rods as with nuts the location of the joining elements can be precisely set along the threaded rods this way accurate orthogonal structure can be built if before fixing the joints their position is measured and adjusted. Also posterior corrections are easier. The joints will be made of hardwood preferably treated, premade plastic joints and/or metal. Plastic joints and hardwood might not seem stable enough but they are easier to work with and if large enough bulk of them is used as a support they are solid. Degradation of wood is not also not a concern in three months interval, but treated wood is available too that infused with resins, antioxidants and antifungal hereby lasting for years. The good thing with a 3D printer is that when a printer is up and functioning new and possibly better version of its parts can be printed out with it.

As the widely available RC servo motors lack proper documentation in most of the cases and from second hand sources they lack precision too, suffer from an effect called servo jittering and usually have the maximum rotation angle of 180 degrees meaning they cannot turn around they are out of the question. Despite the fact that they would provide a closed loop control system out of the box which is more bulletproof as there’s feedback from the position of the motor and the also include a driving circuit. On financial grounds magnetic linear motors and piezo motors are out of reach for this project. There are also some concerns with the durability of piezo motors that the crystals might break after a certain amount of work time. Conventional DC motors would require a feedback and the number of poles and gear ration would define its angular accuracy, holding torque would be insufficient. Thankfully inexpensive well documented powerful and accurate stepper motors exist on the international market as everything nowadays they originate from China but they are available from European resellers too. The steppers motor are used from a long time to move the carriages of printers, as they provide large angular accuracy, angular repeatability, high holding torque and require only an open loop control system to operate. So these motors are my choice. A total five of them is required for the materialization of a smart and nowadays very popular build where on the horizontal plane there lies a moving work bed the workplace where printed parts adhere. The heated bed, the workplace moves on one axis because of space-saving and stability purposes. If the vertical support would move it would require more robust parts. The vertical and the other horizontal planar movement remains the same as on the basic conception figure.

The moving heated bed is to be brought to life with one stepper motor, linear bearings and one stepper motor mounted on the frame, and a pulley system between the motor and the carriage. The vertical axis has also a twist in it, or two to be precise two stepper motors are required that move two separate, parallel threaded rods. When they turn they raise or lower two carriages, that support the other horizontal plane axis. This is clever because using one stepper motor it would be difficult to synchronize the two sides, but with two stepper motors connected in parallel they are synchronized and they even synchronize each other electrically. If one is turned a voltage is induced on its connections then a current starts to flow that turns the other motor in sync with the other. Other advantage of it is that a threaded rod is more linear and ideal for holding weights vertically. One sided vertical axis where instead of two pillars only one would require one less motor but would also reduce the accuracy. The second horizontal plane axis is built with also one stepper motor and a pulley system just like the moving heat bed. The carriage of it carries the extruder that also requires a stepper motor to control the amount of filament introduced into the heated nozzle that melts the plastic. The heated nozzle and the heated bed needs temperature sensors.

The motors and the heaters need to be powered and controlled, this and it’s architecture is under research. But an Atmel microcontroller will be used with some power electronics and PC power supply. The communication will be possibly managed on RS232. The PC will run CAD, CAM software that is not made in this project and will send G code and control code, so the Atmel microcontroller has to decode them.

This whole setup is not unlike most of the Reprap machines. Reprap is an open source movement started in 2004 aiming to develop and spread 3D printers that can reproduce themselves.

Verification (test) against requirements:

These 3D printers are basically robots that manipulate objects in space. Their most crucial part is their linearity, if they have uncompensated flexibility in them it is easy to see that they will manipulate the objects inaccurately, in this case resulting in inaccurate 3D prints. Even with the largest care taken during the project there will be some inaccuracy left in the system. The role of the 3D printed part defines the tolerance of it. I estimate my 3D printer will be turn out decent enough and will have of +/- 1 mm/cm axial tolerance. For example when printing out a 1 cm x 1 cm x 1 cm cube the distance between the opposite faces can be minimum 9 mm and maximum 11 mm, when printing out a 1 cm x 1 cm x 1 cm cuboid, the extension of the object in the 3 cm orientation can be minimum 27 mm and maximum 33 mm. I will measure this with a caliper, compare it to the CAD file and document it.

Structural coherence will be tested by taking the model into the hand and visual inspection. If it stays in one piece it passes the test. A photo will be made from it for the documentation. This test looking to be childish at the first glance is very important because fused deposition of materials can result in the material being struck to the workspace, in this case the printer bed. Also improper layer deposition because of level mismatch and temperature errors can result in separation of layers. So there is a very high possibility that the print will not be staying in one piece.

The other factor is the time, which is by the 3D printers inversely proportional to the accuracy of printing. More complex prints can take days even with commercial machines. Time will be measured via a stopwatch and also to be documented. If time from the start of the print to the end of the print is less than two hours it is accounted as a pass.

Time plan:

My time plan is represented on the following Gantt chart where \* meaning milestones.

January:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| process/time: | 1-3. | 4-10. | 11-17. | 18-24. | 25-31. |
| Thinking on possible projects. |  |  |  |  |  |
| \*Sufficient amount of ideas collected |  |  |  |  | \*31. |

February:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| process/time: | 1-7. | 8-14. | 15-21. | 22-28. | 29. |
| Selecting the best project idea. |  |  |  |  |  |
| Best project idea selected and presented. |  |  | \*16. |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Preliminary mechanical and electronic conception, parts list. |  |  |  |  |  |
| \*First set of parts ordered (from internet). |  |  |  |  | \* |

March:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| process/time: | 1-6. | 7-13. | 14-20. | 21-27. | 28-31. |
| \*First set of parts bought (in local shops). | \*2,3 |  |  |  |  |
| Electronic design. |  |  |  |  |  |
| \*First set of online ordered parts acquired. |  |  | \*16. |  |  |
| \*Reordering, shopping non arrived parts. |  |  | \*17,18. |  |  |
| Mechanical building. |  |  |  |  |  |
| \*Completion of mechanical building. |  |  |  |  | \* |
| Stepper motor driver building, programming, testing. |  |  |  |  |  |
| \*Functional stepper motors. |  |  |  |  | \* |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Heater circuits building, programming, testing. |  |  |  |  |  |

April:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| process/time: | 1-3. | 4-10. | 11-17. | 18-24. | 25-30. |
| \*Functional heater modules. | \* |  |  |  |  |
| \*Receiving reordered parts. | \* |  |  |  |  |
| Main electronics assembly. |  |  |  |  |  |
| \*Preparing slide show. | \*2,3 |  |  |  |  |
| \*First demo. |  | \*5 |  |  |  |
| Writing control software. |  |  |  |  |  |
| \*USB communication. |  | \* |  |  |  |
| \*Working G code interpreter. |  |  | \* |  |  |
| \*PC CAD CAM compatibility. |  |  |  | \* |  |
| \*First print. |  |  |  |  | \* |
| Testing. |  |  |  |  |  |

May:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| process/time: | 1. | 2-8. | 9. |  | **∞** |
| Testing verification. |  |  |  |  |  |
| Preparing slideshow. |  |  |  |  |  |
| Presenting slide show. |  |  | \* |  |  |
| Lifecycle support. |  |  |  |  |  |

Budget plan:

|  |  |  |  |
| --- | --- | --- | --- |
| **Part** | **Quantity** | **Price (HUF) (total of n pieces)** | **Inland** |
| 3PCS 45Ncm Nema 17 Stepper Motor 2A 4-wire 1m Cable for DIY 3D Printer CNC Robot | 2 | 24420 | N |
| 3D printer joint pack | 1 | 12210 | N |
| cuboid hardwood 1m | 1 | 2000 | Y |
| hollow iron bars 1m | 3 | 2000 | Y |
| threaded rods 1m | 2 | 9000 | Y |
| Linear rods 1m | 3 | 9000 | Y |
| linear bearings | 15 | 9000 | N |
| extruder nozzle | 1 | 8500 | N |
| extruder mechanics | 1 | 5000 | N |
| screw, bolt, nut set | 1 | 6500 | N |
| screws, bolts, nuts | 1 | 4000 | Y |
| stepper motor bridges | 5 | 10000 | N? |
| voltage regulator/distributor | 1 | 10000 | N? |
| Atmel developer board | 1 | 10000 | N? |
| pc power supply | 1 | 7000 | Y |
| PCB heatbed | 1 | 2500 | N |
| raw PCB | 1 | 2500 | Y |
| temperature sensors | 2 | 6000 | N |
| 1 kg PLA filament | 1 | 8500 | N |
| 1 kg ABS filament | 1 | 8500 | N |
| timing belts and pulleys | 4 | 10000 | N |
| ball bearings | 4 | 3000 | N |
| wires |  | 5000 | Y |
| extra expenses |  | 10000 | N |
|  |  |  |  |
| **Total:** |  | **184630** |  |

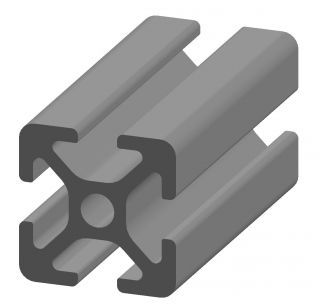
Building the mechanical part:

By the end of February I ordered the parts for the mechanical construction from ebay.co.uk. The reason I chose ebay.co.uk is that most of the parts are hard to access in Hungary or they ask a horror money for it. Additionally I was thinking that as they are the largest island of Europe, they are closer to Hungary than China the origin of products. So the goods will arrive faster. United Kingdom is also an EU member since 1973 and as by the laws of EU the duties for the goods has to be only paid upon the entry of a product to the territory of EU. Had I ordered from China, the prices of goods would be 2/3 of the UK prices, however the shipping time could reach three months if something is messed up. In addition comes the VAT process that can halt the shipping for weeks. Then the authorities randomly pick the packets for VAT process, that is paying 27% of the value of the packet complemented with bureaucratic mailing . Not to mention that it’s a trend for Chinese ebay sellers that they are sell from large warehouses and their advertised shipping time is actually their posting time. So for example they advertise their product with 2 weeks of shipping time, one orders the item and two weeks later he or she waits for the postman, instead of the postman only receives a letter in poor English that someone excavated the items of order after hundreds of rounds with AGVs and posted it with china mail. Then the sea adventures of the pack begins.

So successfully avoiding this the parts have arrived in early March. One order was with hair rising communication. The communication was outsourced to India, and they sent letters they did not even write nor understand nor bothered about, stating that after two months I should receive the motors from China. Most of the email was about begging for positive feedback. Then they automatically sent more begging for positive feedback. I began to cope with a significant amount of money loss then fortunately the day after they posted it I received a text message that my motors are coming from UK with DHL. DHL was super fast, it took only 4 days and they delivered on Saturday as Saturday was a workday then.

So I had there I was having 3x40mm12V 6 watt with 300 mN/m torque stepper motors, 2x30mm 3,75 W 180mN/m stepper motors ,2 20T 2P GT2 pulleys, 2000mm 6mm wide GT2 timing belt. 8x410mm linear guides, 2 8x308 mm linear guides, 2 8x293 linear guides,10 linear bearings, 2 5mm to 8mm flexible coupling, 275x220 mm 12V 120w PCB heatbed, PCB heatbed slider trays, kit of screws, 1 608 bearing, kit of 3D printed 3D printer parts, kit of screws, laser cut acrylic pulley mount, 1 extruder kit, 2 motor mounts, a kit of wires.

For the frame I that adds up to the half volume of the 3D printer I wanted to use parts that can be found in the local home depo, Bauhaus, the initial conception that was mentioned before was that I use threaded rods and despite my rapid visit showed that there are threaded rods can be found there this idea was dumped when during my later visit I discovered that no joining elements can be found there and they would be extremely labour intensive to produce them from the parts found there. I also dumped the idea of using wood thinking despite the fact that wood is sufficiently strong, durable and time resistant it would bring an unprofessional appearance for the build. So I researched everywhere for a good construction, then I found that for such applications Item or Bosch aluminum profiles are used.



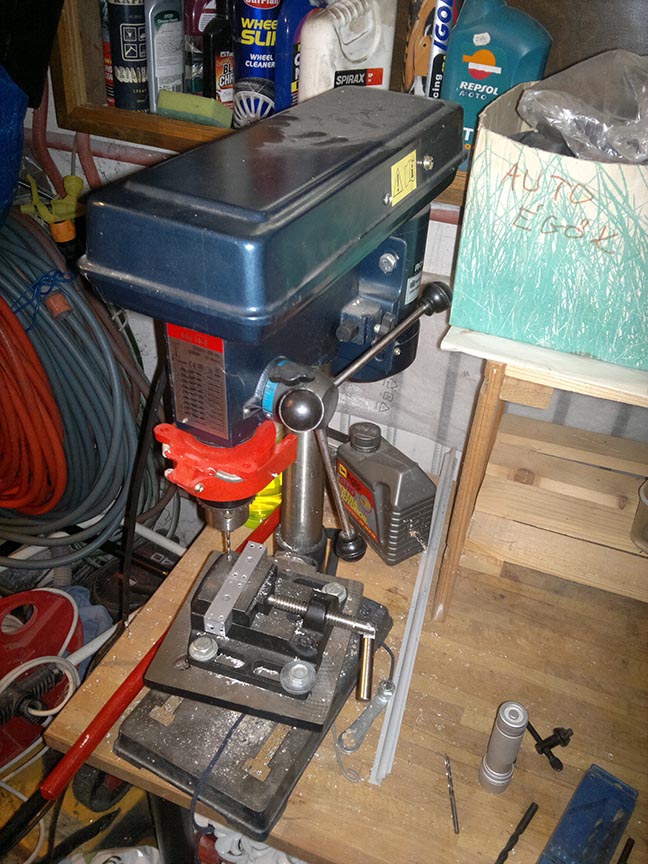
In the grooves T shaped nuts can be inserted so that way the appliances and parts can be bolted on longitudinally with ease and flexibility, however it’s not so cheap and it’s availability is questionable in Hungary. For example if you want some you have be a company and the minimum amount you can order is 15 m and they do not cut the 15 m beam but they transport it with a 15 ton lorry for 45000 HUF to you from the other side of Hungary. Other options were to weld a bracket or to screw together something. Welding would have been straightforward but from my previous welding experiences I know that welding twists the frames due to thermal expansion of metals so bolting together 20x20 aluminum beams remained the option. With the lack of corner elements fastened with steel corner elements and bolts. The material is aluminum because on the surface of it a white contiguous oxide layer forms that prevents it from rusting, iron, and steel rusts. Aluminum is also softer thus it’s easier to shape, and it’s easier on tools: does notch metal tools that easy. Also quite elegant and light.

So construction began in the little family basement workshop. The main rule was to contain and latch the guide rods, that are precut to such lengths so they support on the X axis for the 275 mm length of PCB heatbed, on the Y axis the deflection of the carriage for the 220 mm width of and on the Z axis 190 mm of travel for the Y axis. So for the X axis 400 mm Y axis 308 and on the Z axis 295 mm was plan with the beams joining standing on the Y axis if something too heavy is packed on it here is a blockage preventing it from collapsing. These sizes could have been larger, but this size fits comfortably on a desk and it is still larger than most commercial 3D printers. Also a limiting factor for the size is the shipping fees: an oversize package might be drastically more expensive and also might arrive later. Also a major factor, that if there are angle errors, there are larger diversions at larger distances compared to smaller distances.

Firstly I had to cut the originally 2 m long aluminum bars to the given lengths.

I did it with a metal saw and a vice. The bars were mounted covered in a rag to prevent scratching. Both marking the cuts and cutting was time consuming as high accuracy is expected otherwise there are distortions in the frame. The problem was to mark the lines in right angle and to cut using hands exactly on one line as the saw blade tends to change course as the hands oscillate. A guiding tool might have helped but I haven’t had one. After like 6 hours I had them cut and chiseled with 1 mm accuracy.

Then possibly the most inconvenient and time consuming part started as changing the Item beams to regular beams brings the requirement of drilling 272 bores: 128 in the corners, additional 64 for steel corner elements. 40 for the X slider, 16 for Y slider, 24 for Z slider. Summing it up before doing it might deter anyone doing it. 1 mm accuracy was desired for each bore so most of them was measured and with a pointy tracer and three hammer blow a mark had to be made to guide the drill bit. Also the holes on both sides had to be on one line perpendicular to the surface otherwise the head of the bolt will be misaligned and will not hold as strong as intended. Or fixing something from the other side will have result in misaligned fixing.



My Chinese drill bit could achieve the latter condition, but the stem of it was wobbly and even with the markings it drilled misaligned drills.



After some time I changed to the tactics that I drilled the first holes by hand then used the drill stand to drill the second hole to ensure that it is directly under the first hole.



The X axis holder was hand made like a jewelers work.



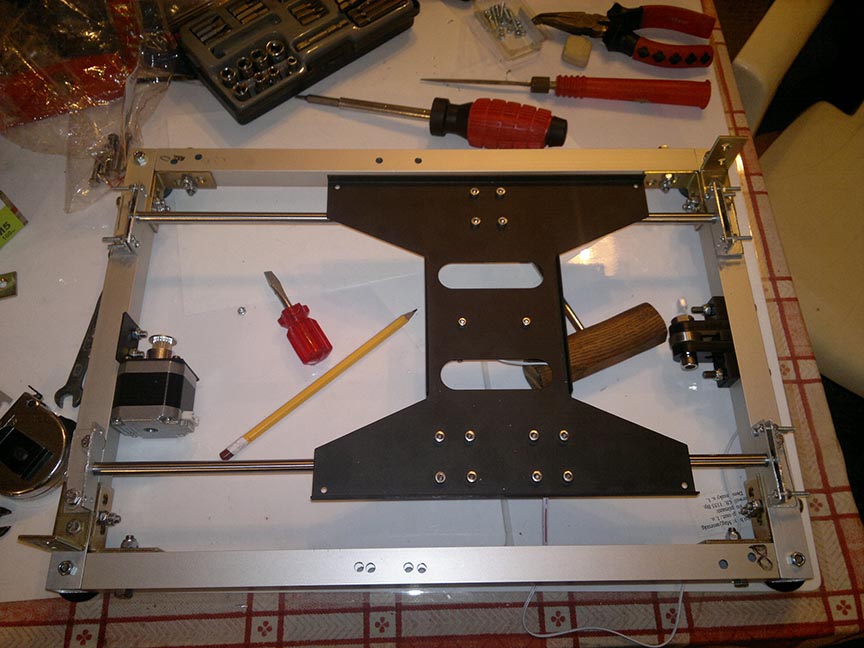
Just like the Y axis mount. But the Y axis mount included some sculpturing with angle grinder.



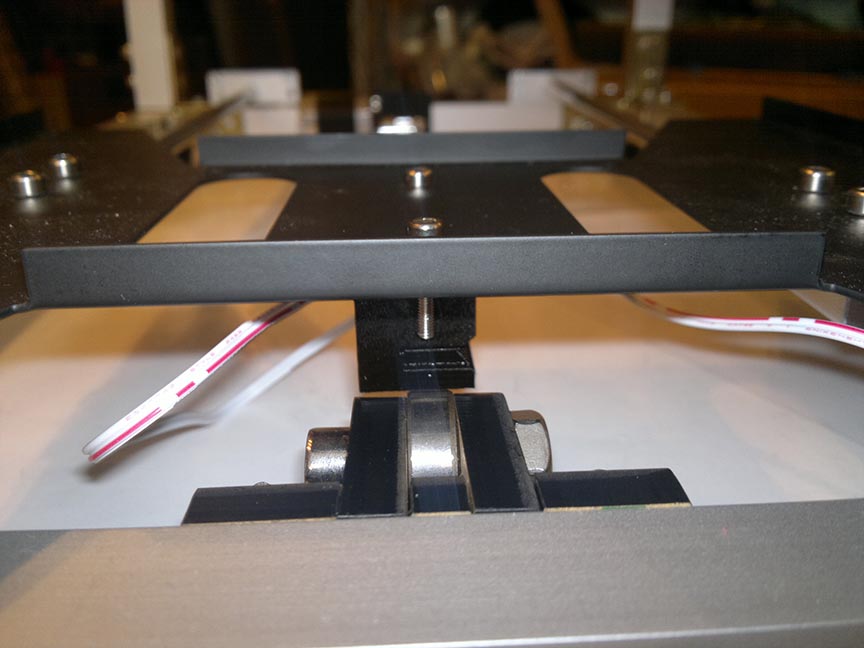
After just 27 hours of drilling and 6 hours of cutting I was ready for 2 days of bolting.



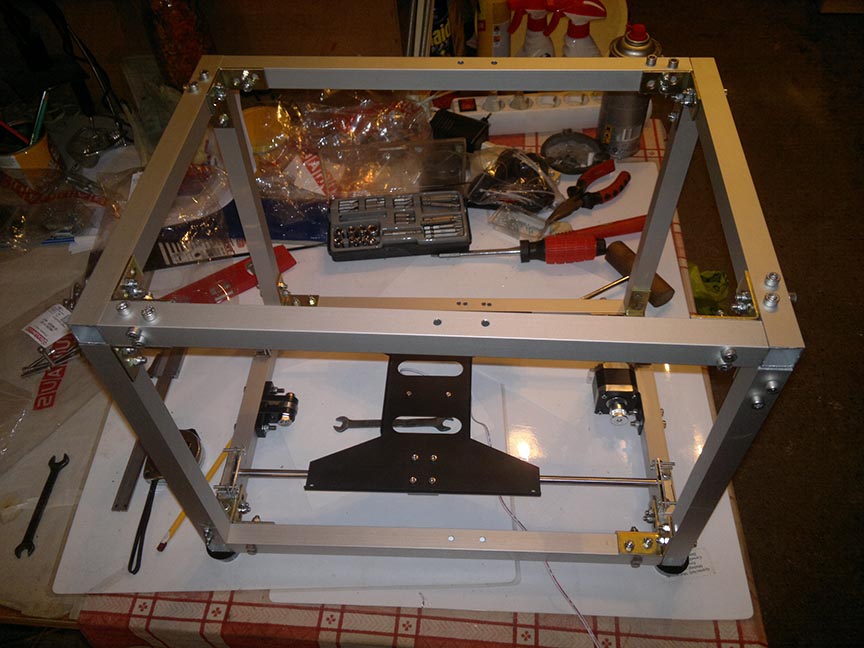
X axis with motor mount, legs and pulley.



X carriage mounted, the align had to be exceptionally precise otherwise the carriage would jam.



The slot is for the timing belt and it aligns well, the measurements were correct.



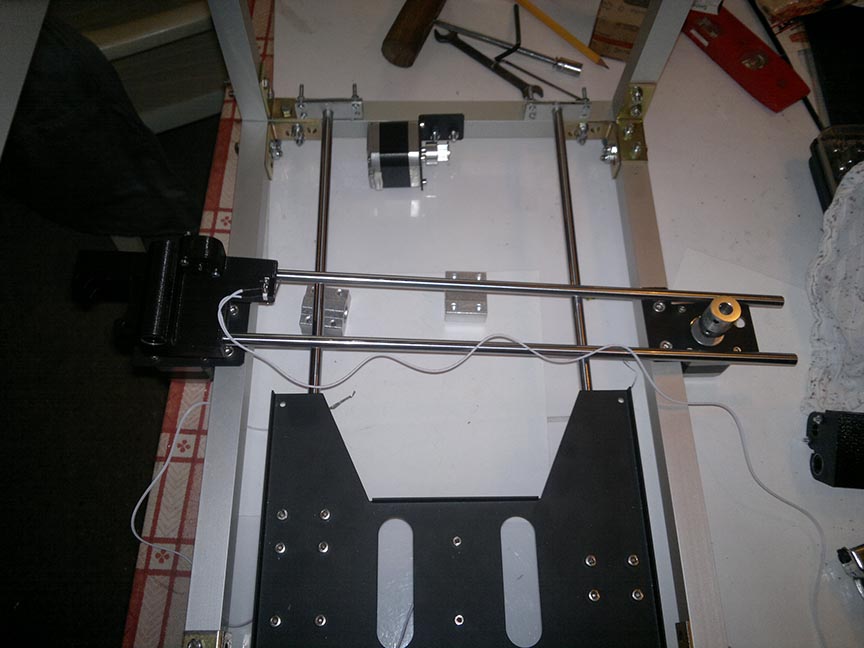
Inserting the top part of the frame.



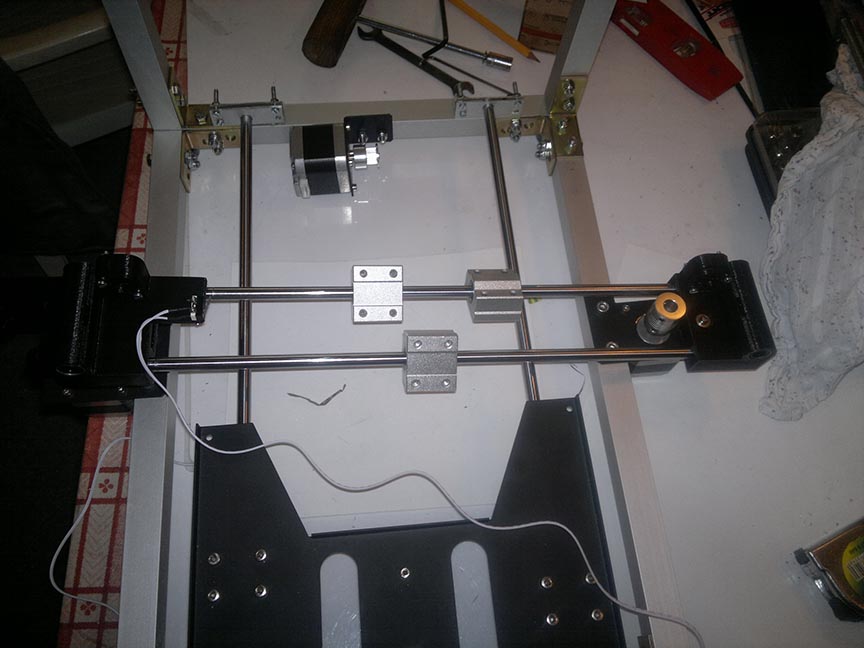
The spacing between the coupling and the motor was set with a screwdriver.



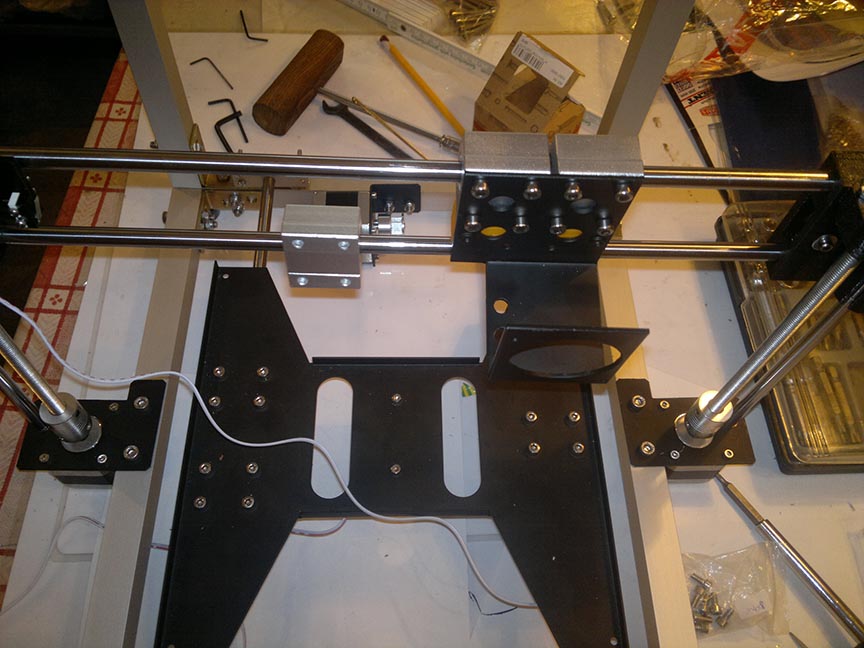
Y axis guide rod holders.



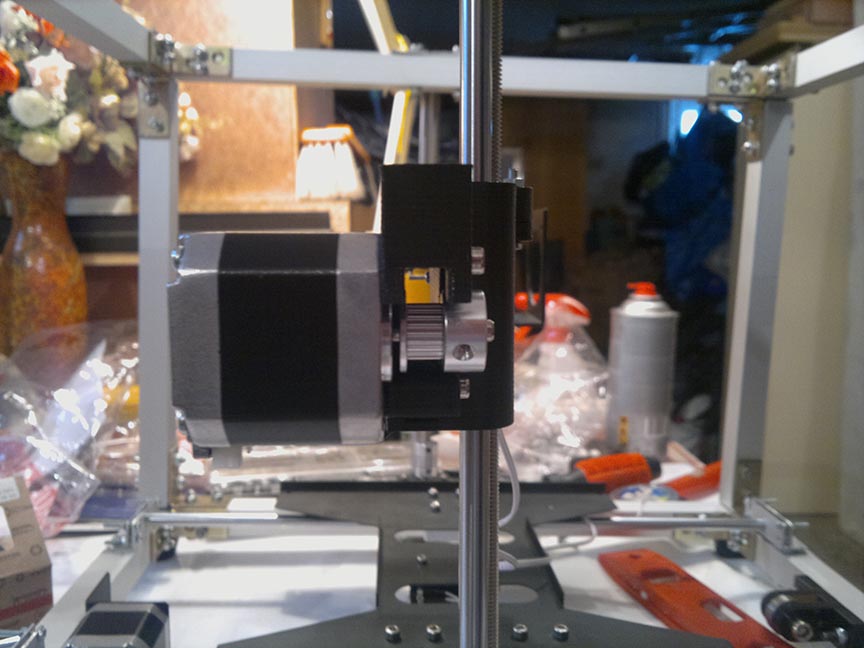
Y axis guide rod holders and one Z axis motor installed.



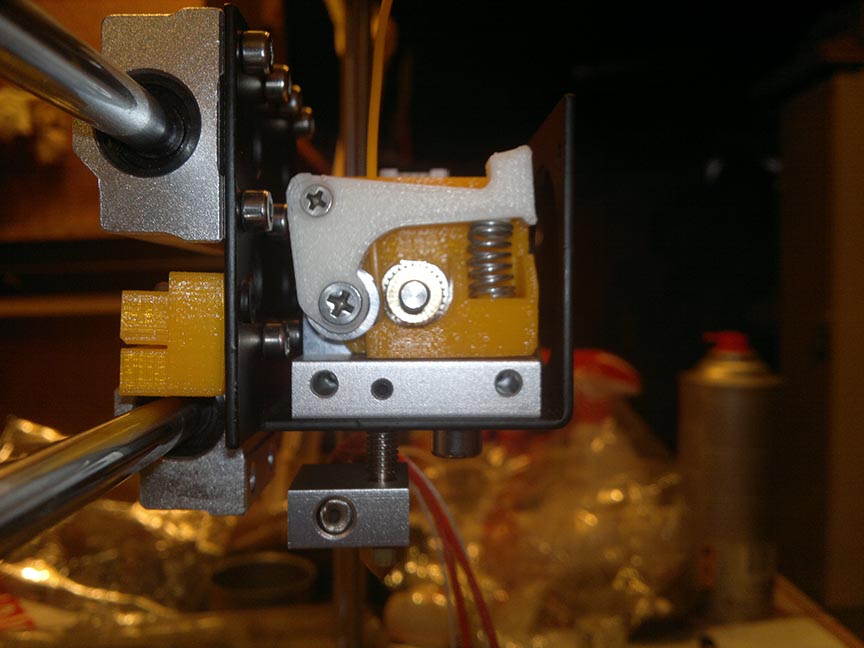
Linear bearings on the Y axis, these hold the extruder mount.



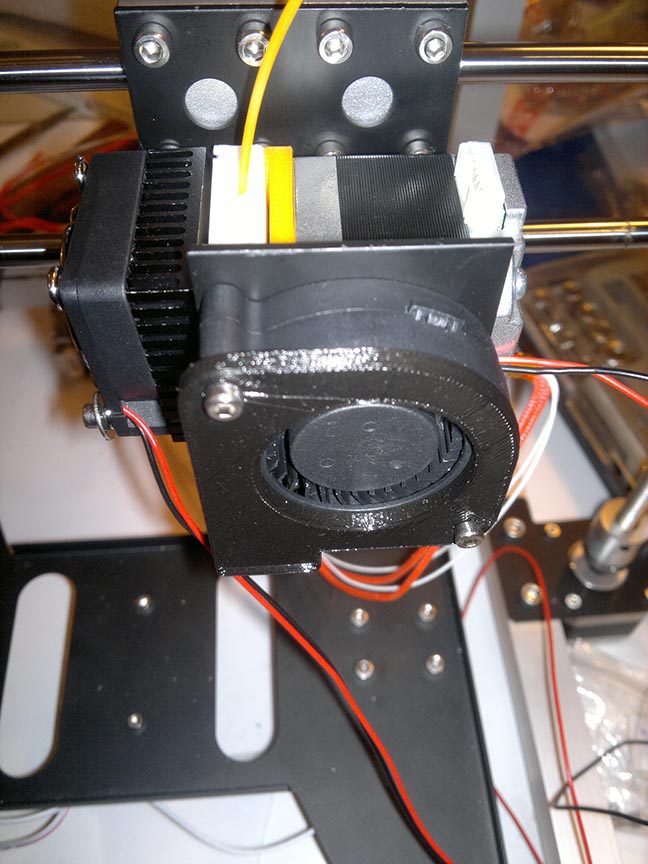
It was tricky to mount the Y axis, I had to turn the threaded rods evenly and watch out for the guiding rods while not letting it fall apart or breaking anything. Then mounting the extruder mount.

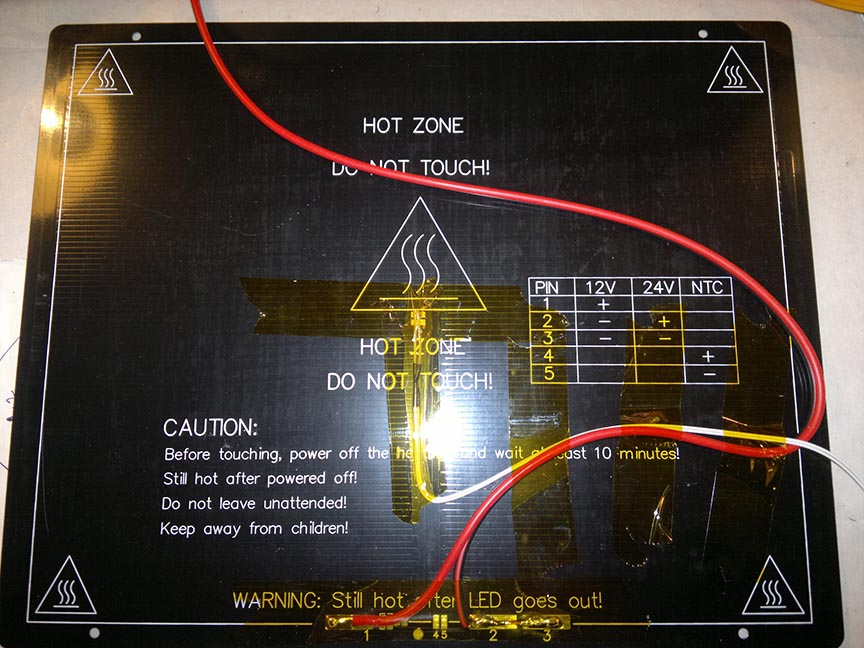


Mounting the Y axis motor.



Assembling the extruder, a copper pinion feeds the filament to the 40 w 0.4mm nozzle. Between the two there is a double heat barrier, one is air isolation one is the gray aluminum by the motor. On the left the place for the Y axis timing belt is visible.

 The heat shield is more effective if cooled (left side). The turbofan is to cool the extruded material, so it hardens faster.



Back side of the 120 W 12 V PCB heated bed. Used to keep the extruded material sticky while extruding.



Without mounting it the X carriage could reach its full deflection



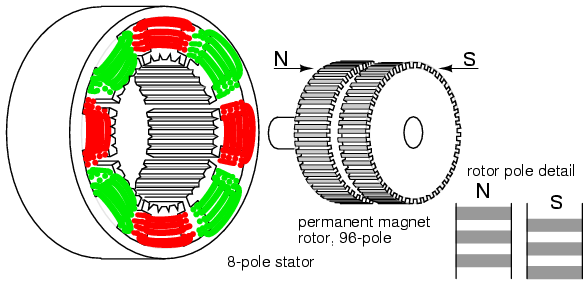
It still qualifies as a 3D printer but some minor future rework is needed. The left side is skew because it was assembled in a hurry.



By 22:00 everything came together that day but the electronics and controller programming was still missing for the final project

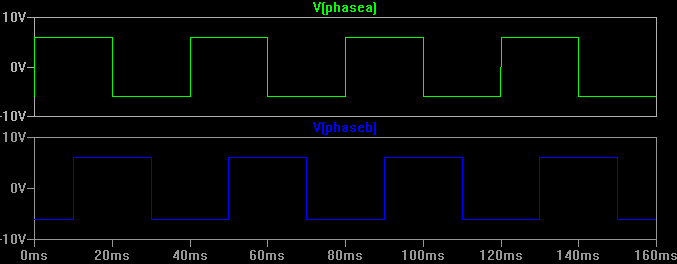
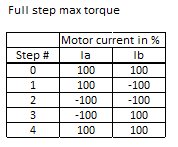
Electronics part:

A simplified diagram of the stepper motor looks like this.

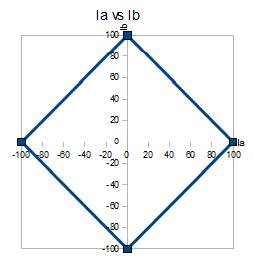
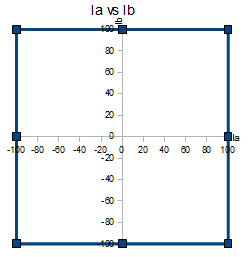
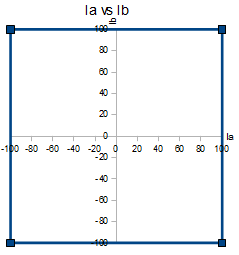
 

The outer part is the stator. On the stator there are poles, the poles can be magnetized with polarity and a polarized stator attracts the opposite pole of rotor. By attracting the poles of the rotor in the correct order the shaft turns. The amount of force excreted is directly proportional to the magnetic field between the two that is directly proportional to the current between the two.

A full step consists of 4 states of excitation:

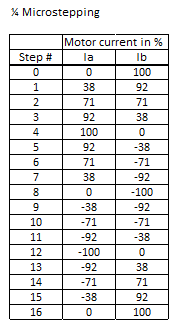
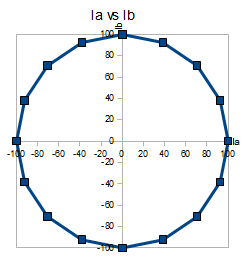


A real stepper motor, like my “goldenmotor.cn 42HD “has 200 steps/rotation resulting in 1.8 degrees/steps. However a much greater accuracy can be achieved if we sketch up the following figures and notice that motor coil windings currents can be vectorized and further quantized, interpolated. (Also in the motor coils there is noting that prevents it to a limit frequency of magnetization and demagnetization hysteresis.)



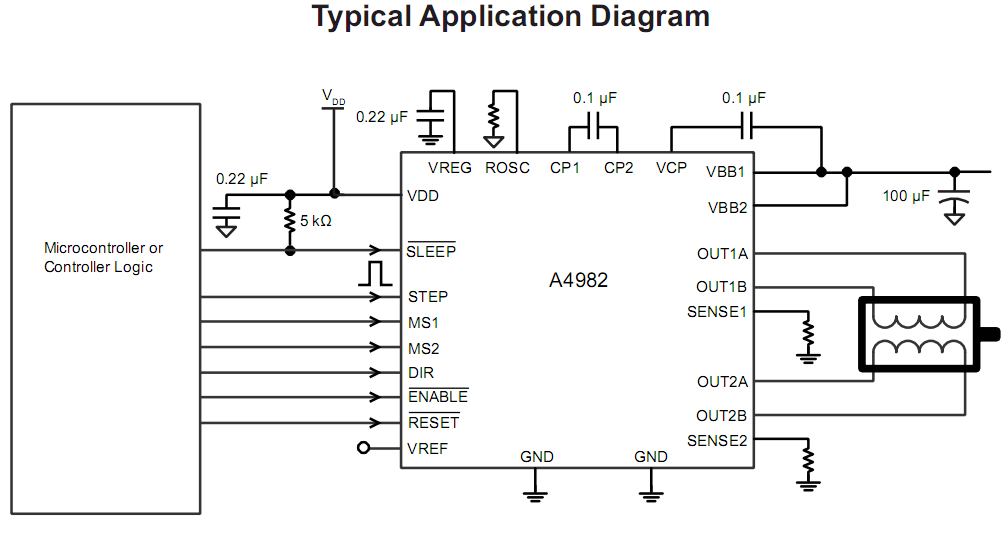
Instead of just pure Ia and Ib current different current combinations and amounts can be feeded to the stator electromagnets. However they cannot be larger than Ia and Ib current. This property results in more sine wave like phase diagram, more elaborate the control. These more elaborate current vectors provide more accurate steps, so called microsteps.

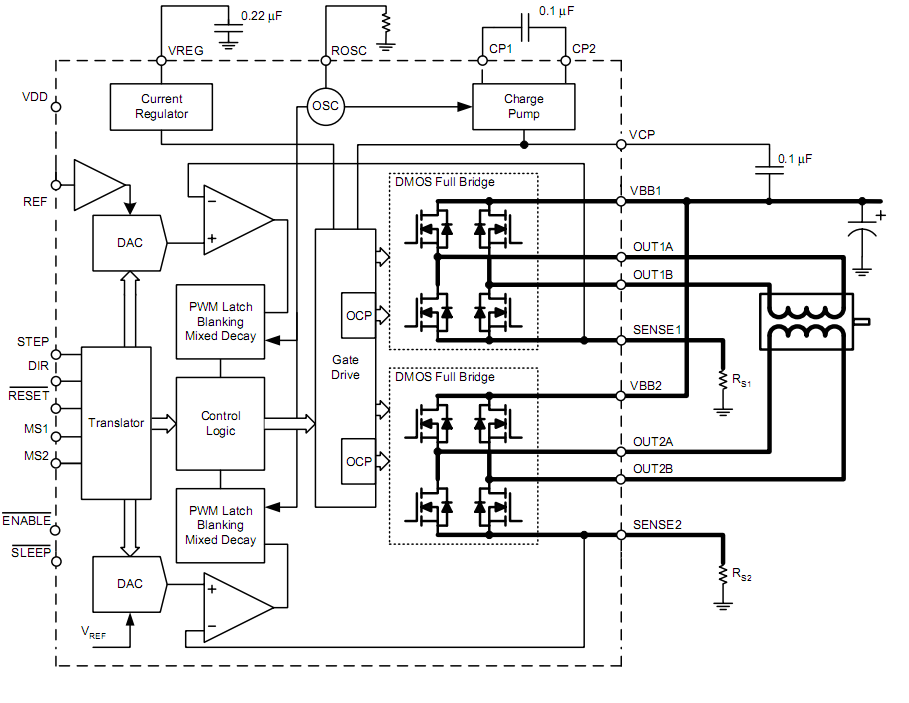
¼ microsteps and its control diagram.



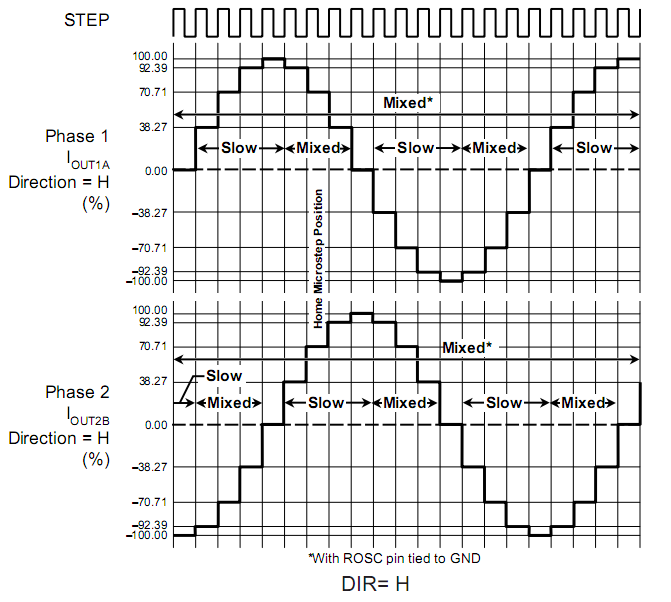
The current vectors could be easily set by PWM and it would have been tempting to build a stepper motor control circuit. But in the age of mass manufacturing a microchip has the same price as an other. A single transistor used for it would have cost as much as one complet motor driver ASIC. Not to mention that I needed four. So I bought 8 A4988 stepper motor driver stamps with heat sinks.







Charge pump supplies the floating voltage. The main control organ is two full bridges. The server circuit sets the voltages.

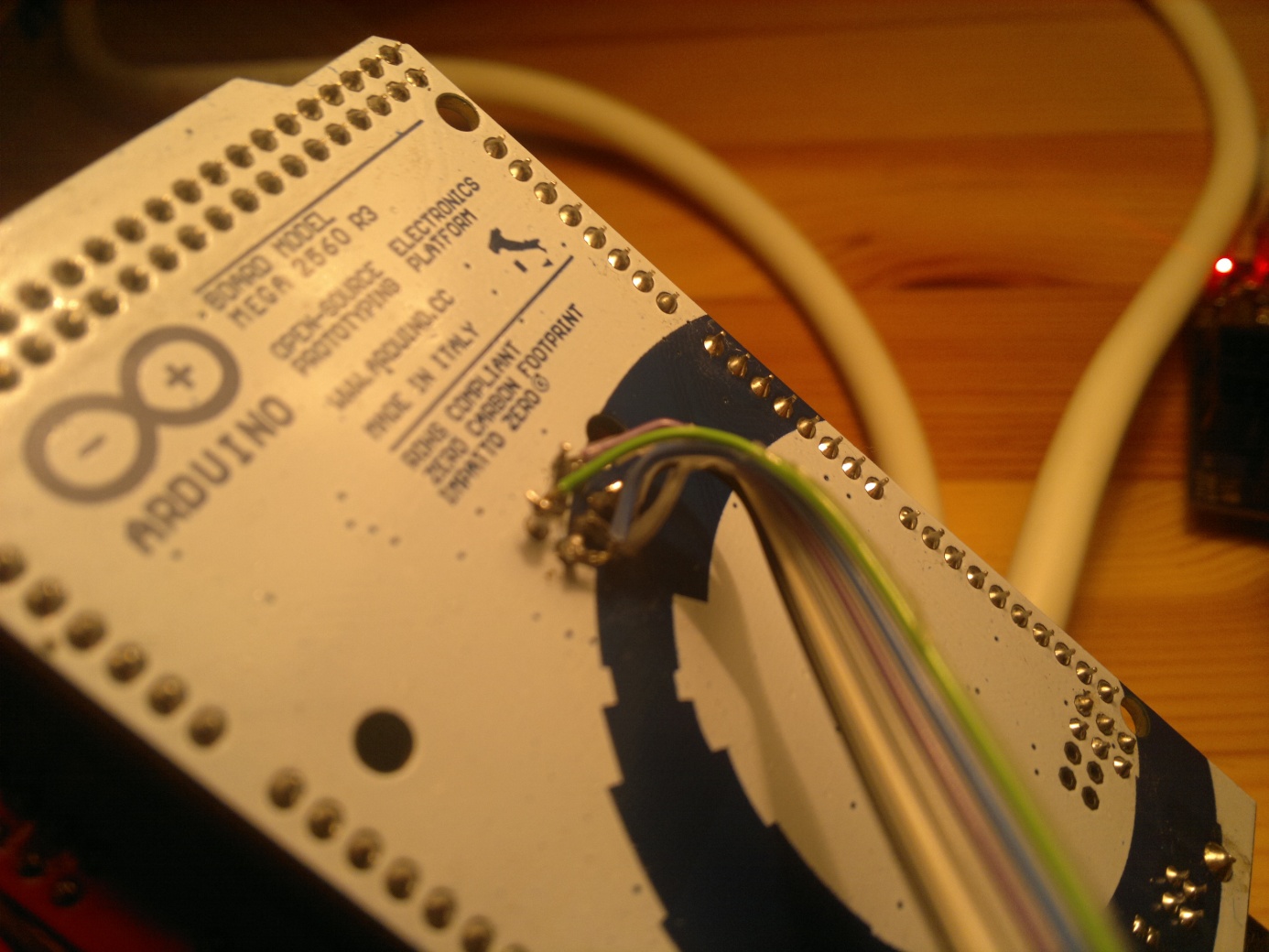


Costing 500 HUF a piece it was a robbery, if building it parts by parts just the server circuit responsible for it’s would have cost me like 20 000 HUF. It’s specs are approx 20 W with cooling, operation range 8 to 12 V.

Digital pins had to be supplied 3 to 5.5 V, MS0, MS1, MS2 has to be set to high to have 1/16 microstep, then enable(negated) had to be connected onto an digital sourcing output set to low capable of sinking. Then pulses on the dir and step pins step 1/16 microsteps.

For logic controller my choice was an Arduino 2560 again it was produced in so high amount that the whole board cost me like the same price of an Atmega 2560. For 3600 HUF I got ready made my 54 digital i/o parts 16 analog input 15 pin PWM output 256kB flash, 8kB SRAM, 4kb EEPROM.

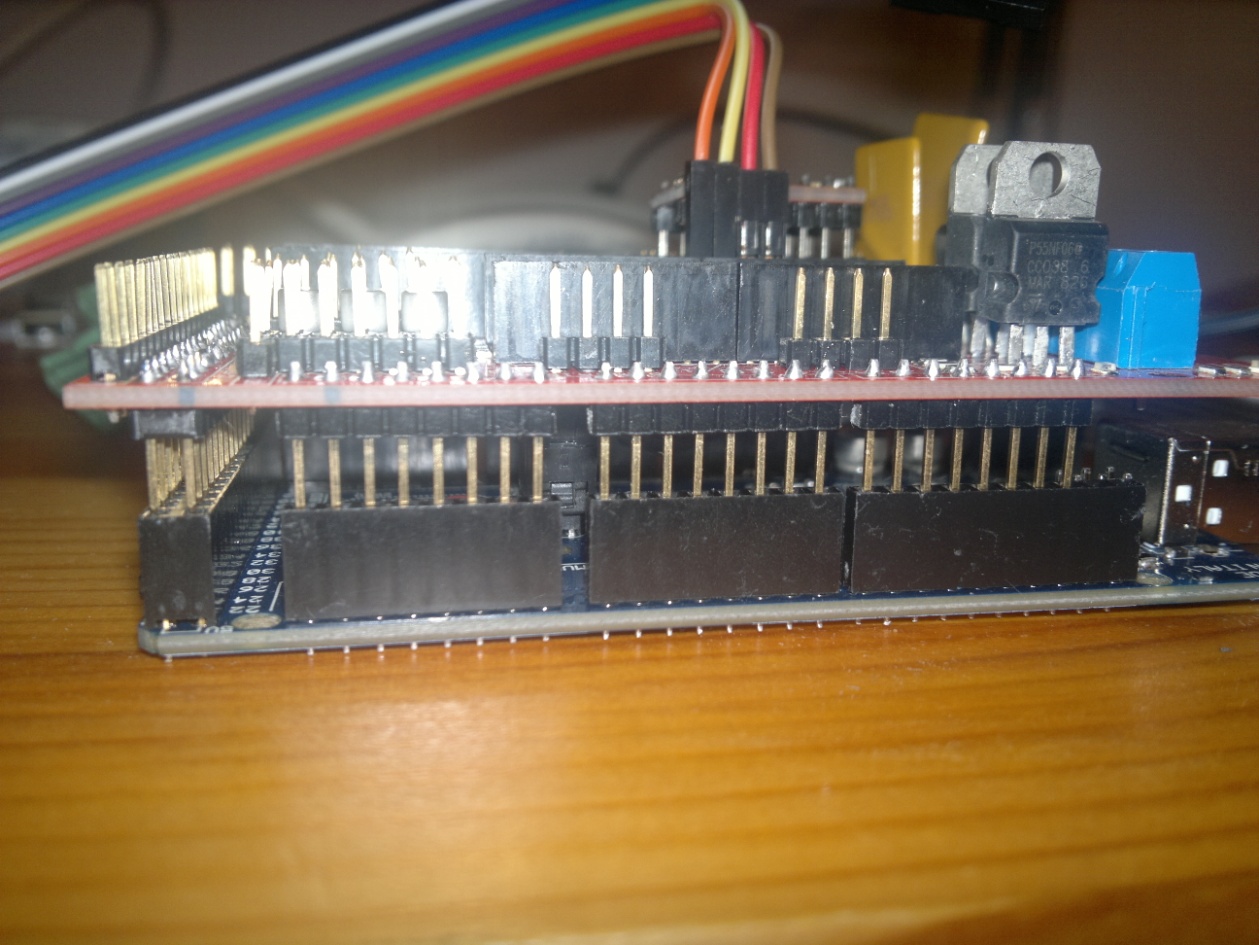




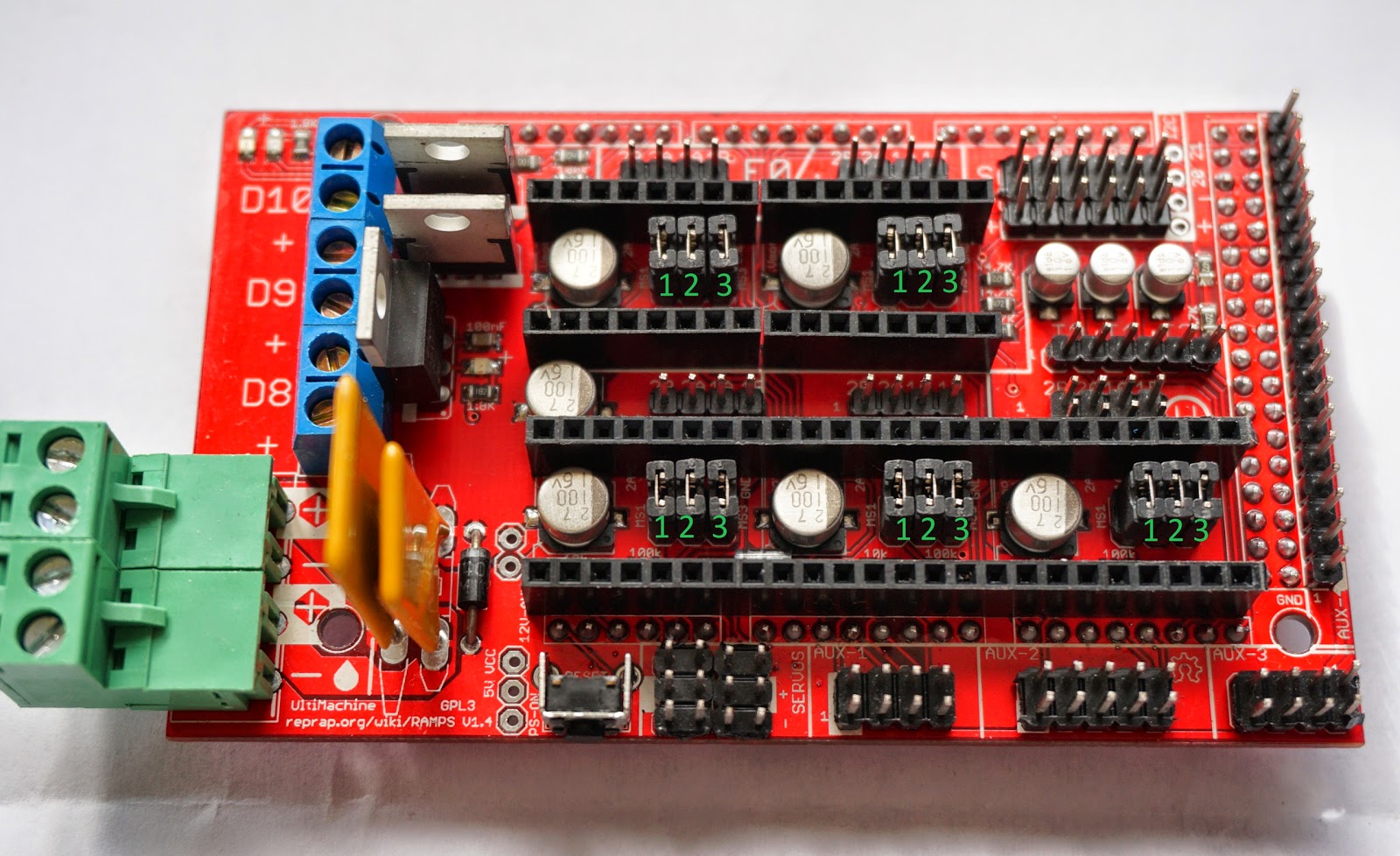
The back states it originates from Italy and was made with zero carbon footprint. But probably it was made in China with coal fire.

The board can operate between 5 (supposedly 3 is enough too) to 20 V from like four different sources. The serial connection is managed with an Atmega 16U2.

With great suffering I made a ribbon cable to program it, then It turned out that the interface between the A4988 drivers and the the Arduino, the RAMPS 1.4 board cannot be inserted into the Arduino if it’s plugged in. So I had to solder a ribbon cable to the back of the Arduino too. ISP programmer was a must otherwise it would have been only programmable in Arduino language.

Arduino and the RAMPS 1.4 

The RAMPS 1.4 board:

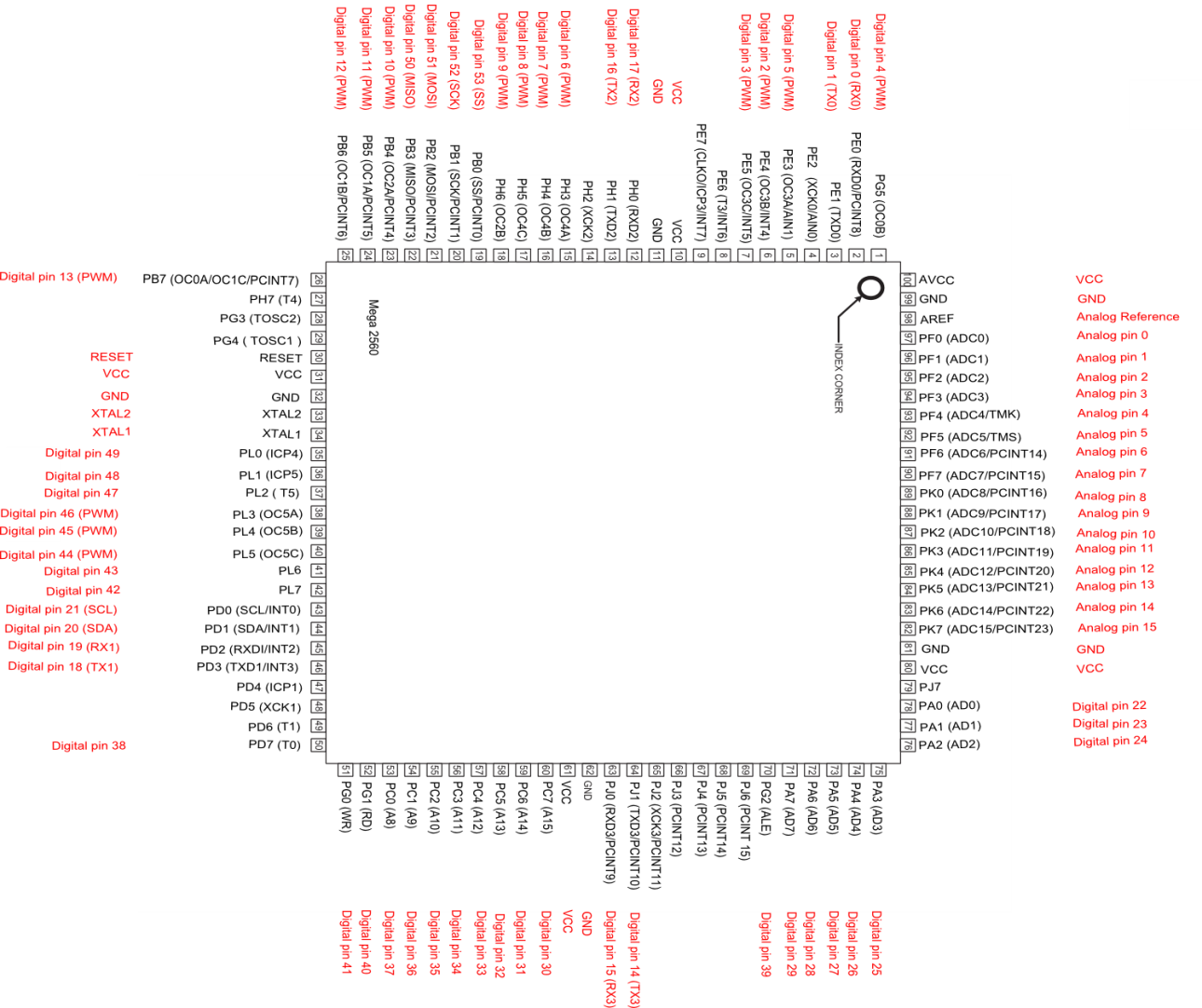


The jumpers are for MSx step mode select, the A4988 stamps has to be plugged into the pins. Green terminal is interestingly 12V 18A (~220W) power supply then two polyfuses that regenerate if they are blown. Diode to prevent backcurrent. Reset switch! And gratis servo drivers. Blue terminals are for the heatbeds and fans switched by 5V, protecting the digital circuit with resistor and a diode. Capacitors are for energizing the coils in the stepper motors and absorbing inductive kickback. There are places to connect and drive the PTN thermistors too.

The open source circuit diagram of RAMPS 1.4



It was frustrating to face that the uses Arduino pin numbering despite the fact that an Atmel Atmega microcontroller is used. So I had to identify the pins relying on this diagram, the datasheet and this map:



It cannot be seen which one is which pin as there is a RAMPS board on the Arduino.

The RAMPS board also had to be slit a bit like a wild grape because some of the pins protruding from it prevented it from being pushed into the Arduino board enough.

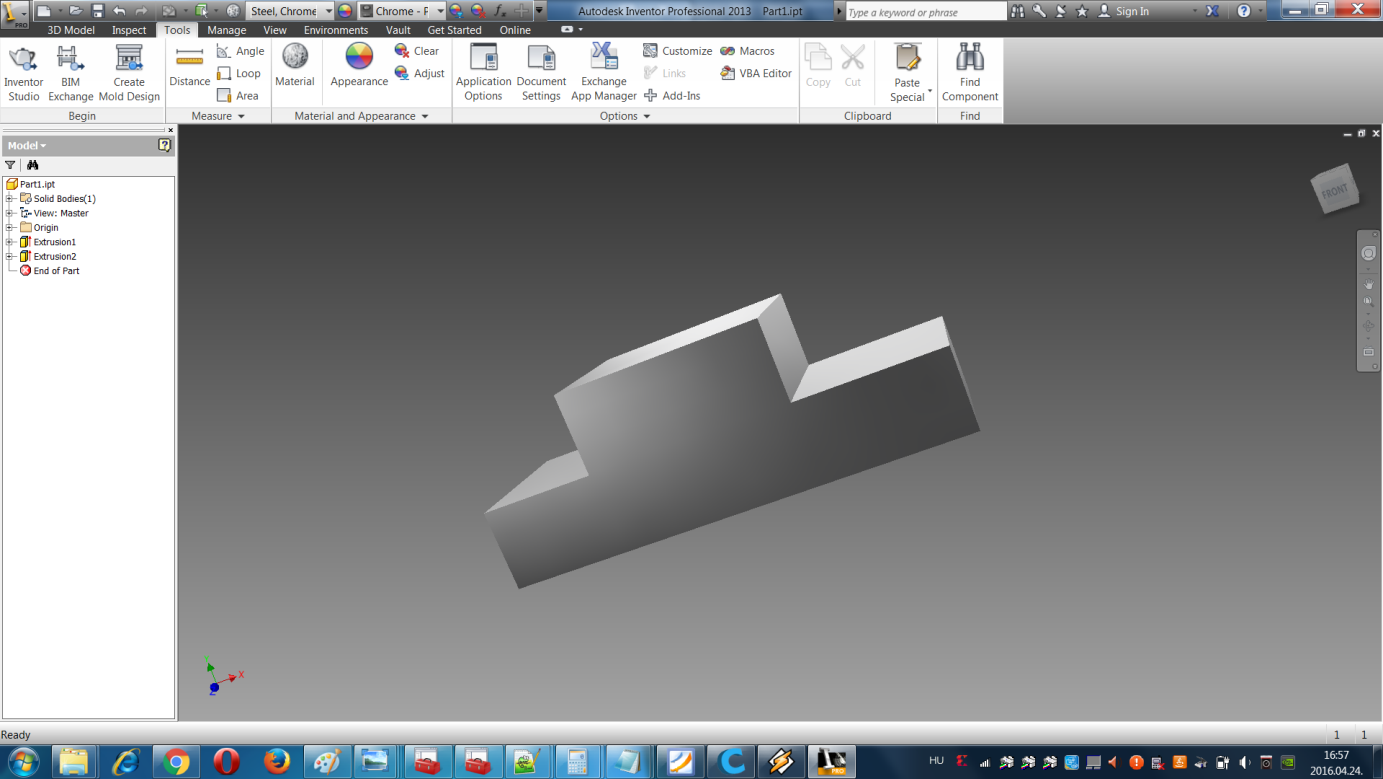


The printer is powered by a 12V 220W Chinese LED PSU, fan cooled. Fortunately with fuses.

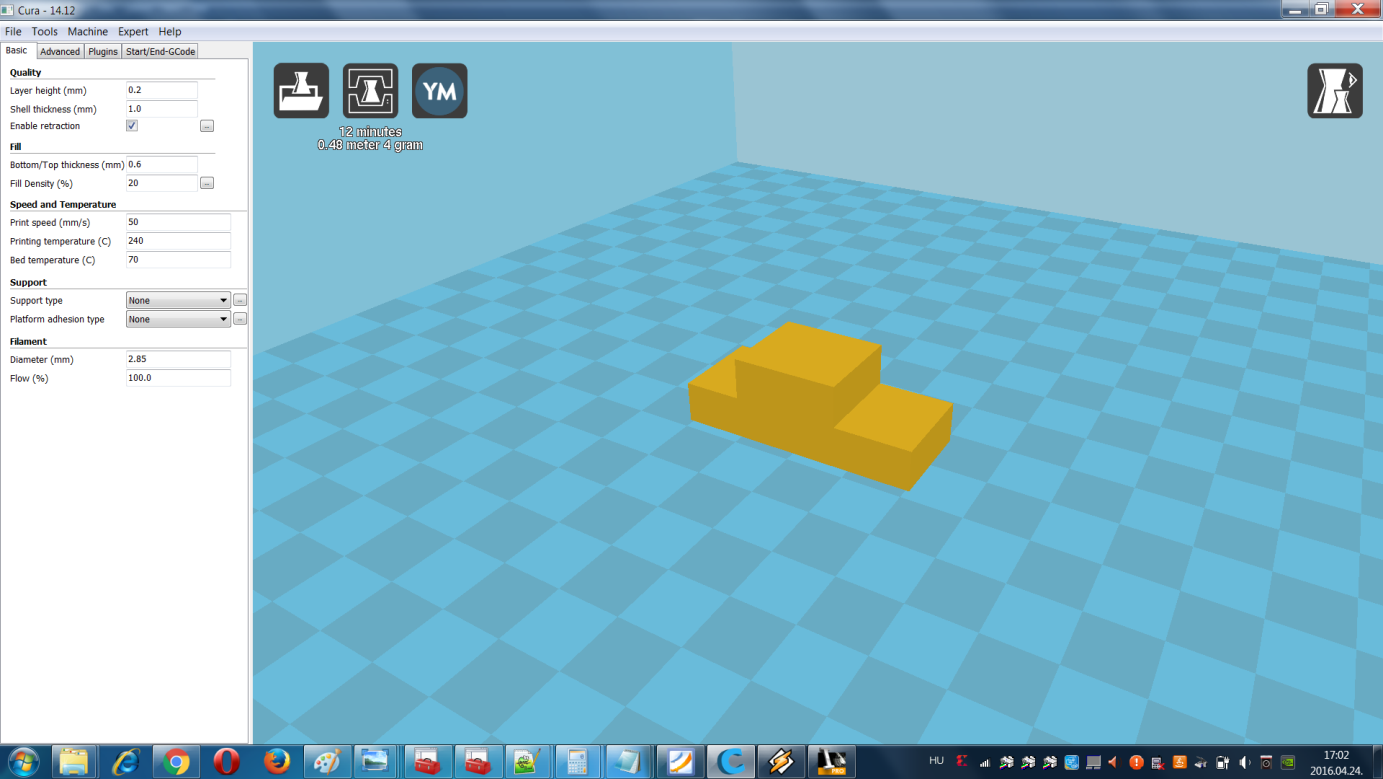
The programming part:

Before programming I had to research what to program. It took a long time. The conception of this project was that I program only the G-code interpreter and controller circuit however this module receives commands from an external program a so called slicer program according to a given protocol.

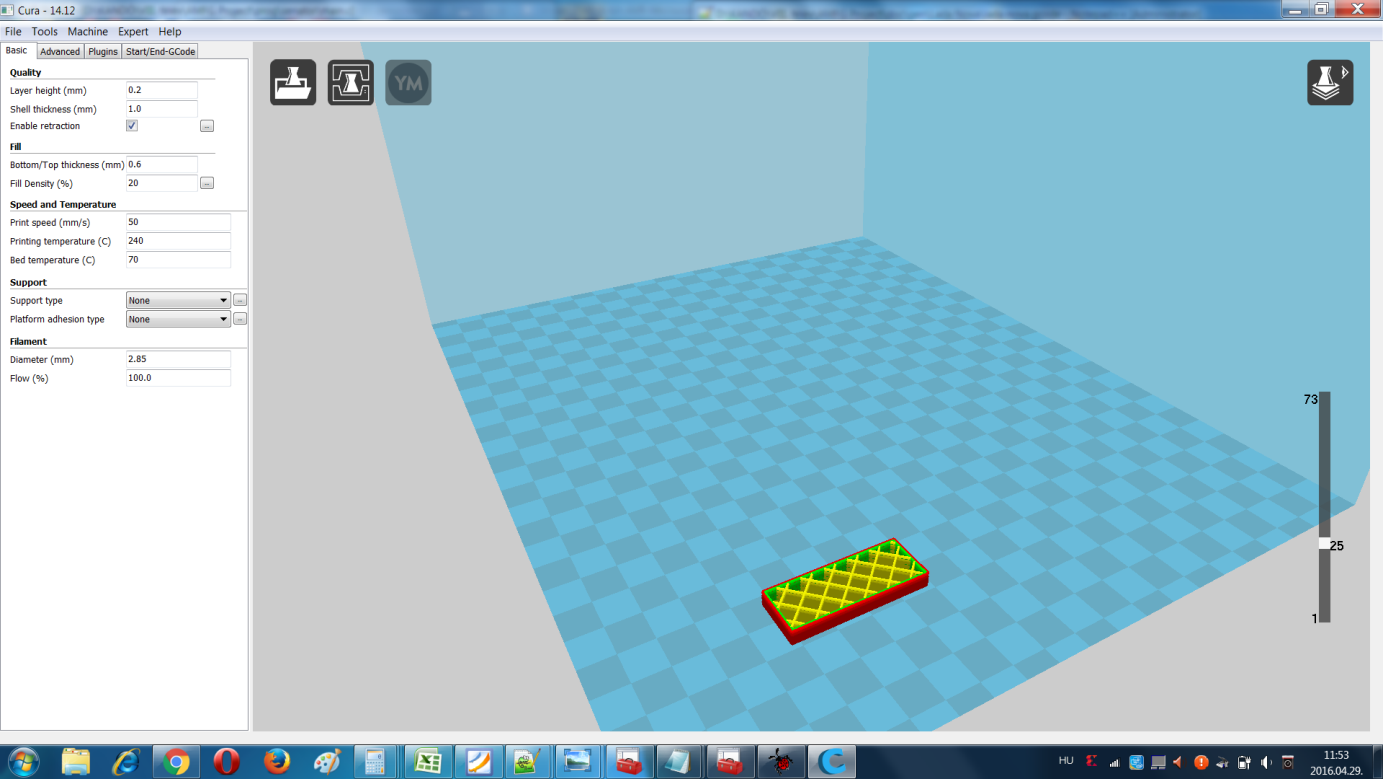
The slicer program I chosen was the “cura “software. The largest factor by choosing it was that it is open source and praised on the internet for the great print quality.



I first prepared a “cube” Lada model using Autodesk inventor.



Then put it in cura.



Transferred it into a G-code print image and reopened it to found this spatial pattern. (The lines would be the extruded plastic)



Then I programmed my TBIRD2 to connect to PC with USART to reverse engineer the protocol. It didn’t work out that well because M10 makes no sense.

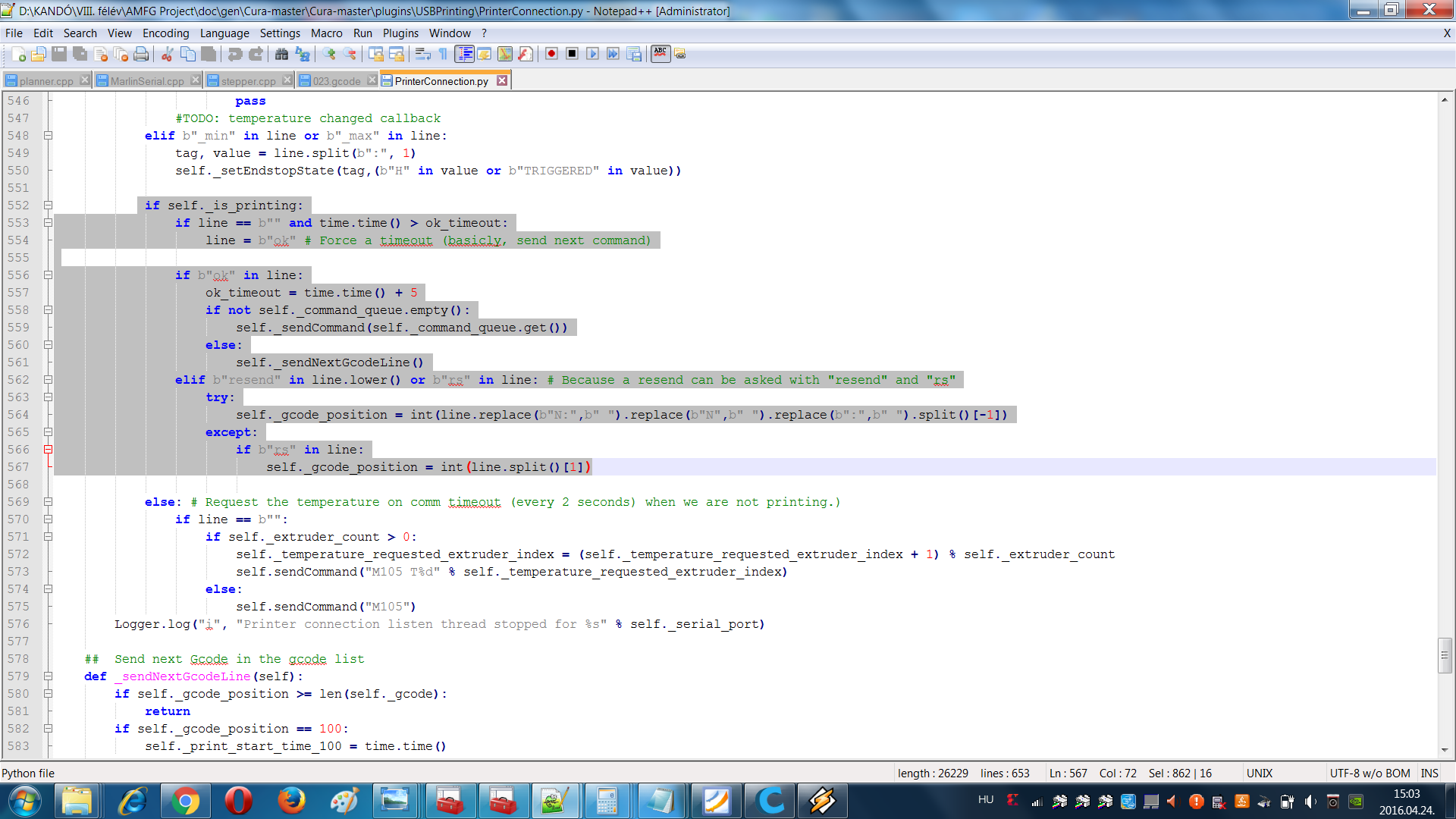


So I changed the LCD display function to display the ASCII codes. But still its M10 and linebreak.

Then I began to read the code of cura written in pyton.



Found out that it sends 3x M105 commands at start requesting the printhead temperature. If it gets it and it’s a correct value then it proceeds.



Also the controller sends ok, if next command can come.

So I started to program the control software.

The final conception of it was that I used the microcontroller as a finite state machine. In mode 0 I processed a line and in mode 1 I processed a command and in mode 2 I parsed the commands in a line and in mode 3 I waited for the commands to be completed with a bit of safety feature that the motors can be halted on a one key command. Finally in mode 4 I reseted the inputs and mode 0 comes again waiting for new command. The commands are stored in a 58 element char array. New line, carriage return, semicolon can terminate a command. New line sets it for parsing. The commands I started to program are diagnostic commands like the M105 and there are configuration commands too like set extruder nozzle temperature and wait. These are incomplete.

The main focus was on getting the G0 G1 commands, the moving commands. Such command looks like G1 Z0.400 X10.500 Y40.250 E41.5719 meaning G1= go and use tool on manipulator arm, elevate (or descend as it is in absolute mode) to Z=0.400 mm, X=10.500 mm and Y=40.250 mm and extrude E=41.5719 mm of plastic filament while doing this.

The largest problem was the number representation as the input values are in rational numbers and the steps required for the stepper motor control circuit are in integers or in long integers. It is also very hard to check their values as there is no debugger on mkII programmer. And printing out a double or a float on USART is also a nightmare. One function could made it a bit easier but due to a IDE bug the library of it cannot be included.

In its current form I convert the coordinates into doubles and use that to determine which direction is to go to for the manipulator. If it’s one coordinate or Z coordinate and one coordinate it’s easy as I made the Z coordinate independent of the others. So the program waits till Z level is set the begins to go to X and/or Y with/without extrusion. The interesting thing starts when X and Y have to move together. For that with basic trigonometric functions: tg(), sin(), cos() and the constant speed rate as the manipulator was set to move with 2 cm/s constant speed. The X and Y axis gain their speed rates respectively to their required speed. In the function of their required displacement. Then this is converted into frequency. The motors are driven by the interrupts as at the beginning no ticker could be used due to the ticker value being optimized by the compiler. For each motor there is an interrupt of 16000000/8 then with a 16 bit timer compare variable the frequency can set and in the interrupt itself goes the motor control. Like for example if(target\_x>current\_x){PINx|=(1<<Motor Step PIN); current\_x++}. The line engine uses the same trigonometric function for the displacements and sets the directions of the A4988 stepper drivers according to the X and Y direction of the required displacement. Then the correct spin speed for the extruder motor is set too. Source code is included in github repository.

Current state:

Heater control programs are missing and the manipulator seamlessly goes to X,Y,Z,E position after a start but fails to go backwards the accurate amount. Goes back less than the required due to some bug with the double conversion.