ANNALS OF THE UNIVERSITY OF ORADEA
Fascicle of Management and Technological Engineering
ISSUE #2, AUGUST 2018, http://www.imtuoradea.ro/auo.fmte/

Designing a delta tripod robot based fused deposition modelling 3 dimensional printer using an open-source Arduino development platform

Tamás Templom[1], *Timotei István* Erdei[1], *Zsolt* Molnár[1,*], *Edwin* Shaw[2] and *Géza* Husi[1]

[1]University of Debrecen, Mechatronics Department, 4028 Ótemető street 2., Debrecen, Hungary
[2]London South Bank University, School of Law and Social Sciences, London, United Kingdom

Abstract. The pinnacle of 3D printing is its effect on the field of rapid prototyping. The major advantage comes from the fact that designers can quickly materialize a part or object, which then could be tested in practice, and can be effortlessly modified if needed. This obviously cuts the development expenses and time by a significant percent. Moreover, it's possible to create complex and precise shapes with the technology, which would be much more time and resource intensive if done by older methods, for example manual or automatic machining.

1 Introduction

The University of Debrecen's Building Automation Research Faculty houses numerous projects of the Faculty of Engineering [1]. These projects always inspire us for further research, like our IoT (Internet of Things) enabled FDM (Fused Deposition Modeling) 3D printer [2]. During my studies, smaller projects always required a modest amount of mechanical parts, which were often needed to be special sized, or hard to manufacture, and would cost an unproportional amount or money. Enter in: rapid prototyping. The explosion of CAD (Computer-Aided Design) systems in 1980 drastically changed the field of engineering, and enabled us to quickly create and manufacture parts at a negligible cost and time. This whole principle made us consider the project. In comparison to the "classical" Cartesian 3D printers, I've decided to use a less common, Delta mechanism, as an essential challenge. The problem comes from the much more complex kinematical and calibration issues of said parallel delta systems. The project started in 2016/2017/2 semester, and with the aid of University of Debrecen's Program for Gifted Students (DETEP).

2 Design points

One of the most crucial design point was cost efficiency, that's why I've chosen an Open-Source Arduino platform. [3]. For the Control Software, I've picked the Arduino firmware Marlin, which is specially designed for Delta 3D printers [4] The other control and slicing software communicates via USB (Universal Serial Bus) on the serial bus. Delta robots can have two types of movement, linear and radial, and I've decided in favour of the linear. Furthermore, the aim was to create a workspace capable of manufacturing parts of standard sizes, and general uses, not only proof of concept smaller ones. Delta tripods consist of three vertical axes, each rotated from 60 [°] from one another. By comparing different present designs, We've chosen toothed belt as a method of moving the system, as it's fast, and precise enough for our demands [5].

The drive consists of a toothed belt of 2 [mm] pitched, GT2 belt and a gear with 20 teeth driven by hybrid stepper motors. We've picked a Nema 17 type bipolar stepper motor, with the dimensions of 42,3 x 42,3 x 34 [mm] [6]. The sled structure consists of 2-2 pieces of Ø8 [mm] diameter circular steel pieces connected with linear bearings.

To reach the entire working area with the hotend, 5 parameters needed adjustments: height of axis(H), radius of delta (DR), consisting of the circle included by the 3 axis, length of pushrod (L), that connects the sled and the effector, offset of the effector (EO), and the offset of the sled (SZO).

The sled structure is downloaded from an Open-Source STL (stereolithography) file sharing site, printed from ABS (Acrylonitrile Butadiene Styrene), and designed for LM8UU type bearings and GT2 toothed belt [7]. The pushrod is made out of carbon fibre, and has ball joints on each end. The heating solution of the hotend is a E3D V6 type element. To connect the pushrods to the hotend, we need an adapter, a so-called effector, this connects the 6 pushrods to the hotend. As we choose the MK2Y type, d = 220 [mm] diameter heated bed, we defined our printing work area as well. On the Z axis, we aimed for 110 [mm] maximum height.

* Corresponding author: zsolt.molnar94@gmail.com

Fig 1. shows our 3 variables (H, DR, L) are heavily dependent on each other. The larger the delta, the longer pushrods we need to reach the correct area on the X-Y plane. but this brings the need to lengthen the axis as well, to keep the movement range on the Z axis as well. The pushrods' angle with the bed at the robot's HOME position is: α. However, during movements pushrod's angle with the bed cannot go under 20°, so the angle in the endpoint is: γ.

- $SZO_z = 20$ [mm]
- $HEO_z = 60$ [mm]

$$[L * \cos(\gamma) + SZO + EO] - [L * \cos(\alpha) + SZO + EO] = 110 \quad (1)$$

$$[L * \cos(\gamma)] - [L * \cos(\alpha)] = 110 \quad (2)$$

$$Z\_max = H - SZO\_Z - HEO\_Z - (L * \sin(\alpha)) \quad (3)$$

Table 1. Specified physical parameters

Component	Specified size
Pushrod length [mm]	336
Radius of Delta [mm]	260,75
Height of Axis [mm]	460

After we laid down our design points, we used the free Autodesk Inventor to create the 3D model of our system [8], which was necessary for further planning. The 3D model enables us to check the fitting and usability of any part that will be added to the system later. It's also beneficial that we can check the possible collisions in a simulation, preventing real accidents.

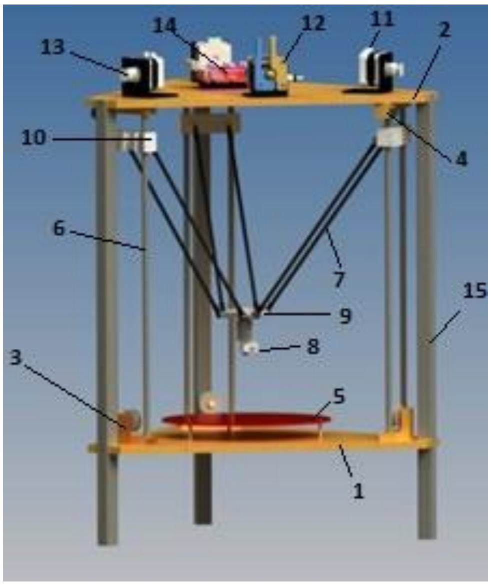

Fig. 1. The model 3D rendered in Autodesk Inventor

Components on Fig 1.:

1. Bottom Tray
2. Top Tray
3. Bottom Axis Support
4. Top Axis Support
5. Heated Bed
6. Round Bar
7. Pushrod
8. Hotend
9. Effector
10. Sled
11. Stepper Motor
12. Extruder motor
13. Pulley
14. Control Circuit
15. Hollow Section Legs

We've used a Bowden solution for the feeder unit, which means that the extruder motor is not directly above the hotend, but connected to the top tray, and connected via a Teflon tube, which guides the thermoplastic material [9]. On the plus side, the total moved weight is less, However, it requires a motor with higher torque, some leftover filament will remain in the pipe, and It's not suitable for feeding flexible materials.

During the assembly, we used tensioners to fasten the toothed belts, and It is also advantageous to use some kind of elastic connecting element in between the pushrod pairs.

3 The completed circuit

Our next job was to install the electronic components. The servo drivers, and the other necessary components were connected to our Arduino Mega with a Ramps 1.4 as a shield, which is a specialized instrument for 3D printers. The shield run on 12 [V] DC (Direct Current), provided by a 280 [W] ATX PSU (Power SUpply). It possesses two 5 [A] outputs, one for the hotend, which is in charge of heating the thermoplastic material, and one for the fan, to cool down the already extruded layers. Additionally, it includes an 11 [A] output, which provides the required power for the heated bed. We can connect up to 3 thermistors to it, in our case, we used 100 [kΩ] resistance ones. Moreover, we added 6 limit switches, which of 3 used as the top endpoints, as basic mechanical NO (Normally Open) switches. Ramps 1.4 can support up to five stepper drivers, currently we use 3 for movement, and 1 for the extruder, so in the future we could add an extra extruder for a second hotend.

Our chosen stepper motors' technical data: 1,3 [A], 0,22 [Nm] torque, roll per each step is 1,8°. We used DRV825 (Pololu) drivers to control the motors [10]. This IC (Integrated Circuit) also houses the necessary resistors and potentiometers to operate the motors.

Fig. 2. Pololu DRV8825 module [10]

One of the greatest benefit of the IC is the so-called microstep function, which enables us to split the steps of the motor to even more steps. In microstep, we distinguish 1/2, 1/4, 1/8, 1/16, 1/32 step angles, depending on the setting of the circuit. We can set the correct microstep on the jumper on the driver. Because DRV8825 is capable of 1/32 microsteping, we choose that. The maximum amperage and voltage is 2,2 [A], and 8-45 [V] over the motors in the case of Pololu.

We used the following formulas to calculate the movement resolution: [11]:

$$\frac{Steps}{mm} = \left(\frac{\left(\frac{360°}{step\ angle°}\right) * \left(\frac{1}{microstep}\right)}{belt\ pitch * dial's\ number\ of\ teeth} \right) \quad (4)$$

$$\frac{Steps}{mm} = \left(\frac{(360°/1{,}8° * (1/\frac{1}{32}))}{2 * 20} \right) = 160 \quad (5)$$

4 Calibration

During calibration, we differentiate between Software and Hardware calibration. During the Software calibration, we set our Arduino program, and the slicing software to the correct configuration. In the firmware Marlin, we have to submit the physical parameters of the printer, the data for the driver, limit switches, and temperature data, and also the pinouts. The free slicing software Ultimaker Cura even possesses an official manual, to set the correct work area, correct communication, and the diameter of the nozzle [12]. We also set the parameters for the current print here as well, for example: fill factor, print speed, and the filament's specified heat values.

Before the physical calibration of the printer, we have to calibrate some individual parts as well. Such parts are like the limit switches, that needs an individual check-up for each component. Moreover, it's important to set the current limit for the stepper motors, thus to achieve the highest torque, we have to use a driver with higher current flow than the maximum current on the stepper. But we don't want to overpower our motors, the maximum set current should be the highest possible power draw of the motor. The correct calibrations result in cooler motors, and no loss of steps. After we correctly set the parameters, we can connect the units to the system. For the correct settings, we need to measure the reference voltage of the instrument. The DRV8825 this method follows the equation:

$$I_{limit} = 2 * U_{ref} \quad (6)$$

Because our chosen motors' maximum current per phase is 1,3 [A], the formula no. 6. let's us calculate the reference voltage: $U_{ref} = 650$ [mV].

The first step of calibrating a Delta-Tripod type printer is to set the P(0,0,0) mechanical null point in advance, manually. The robot set in the HOME position will be moved along the Z axis in the negative direction (down) until the hotend reaches our glass plane covered with a sheet of ordinary paper, but it shouldn't press it onto the glass. Because a sheet of paper usually has about ≈0,1 [mm] thickness, we can enter the correct parameters into the Arduino program from the total distance corrected with the girth of the sheet. This is the so-called paper test [13].

These corrections are followed by refining the limit switches. This is done by printing a one layer, but thick test prop, to measure the difference between the thickness of the layer approaching to the axes from the centre. If the measured values do not correspond to the pre-set thickness of the layer, we can use the difference for correction purposes again.

Our latest task is to set the dimensions. This can be done with two methods. In the first one we use an equilateral triangle based prism. By measuring the height of the object on different edges and sides, I can come to the conclusion of some errors, such as loose belts.

On the other hand, we can use a cube as our test print as well, but all in all, the main goal is to measure the dimensions, correct the erroneous calibrations, then print once more, and measure once more as well, repeatedly, until we get the same values on the physical manifestation that we gave to the program.

5 Reference prints

With the calibration prints, we came to the conclusion of:

- X and Z axes had marginal differences, but on the Y axis at 20 [mm] had -0,1 [mm] difference.
- At 80 [mm/s] print speed, we can still able to use tear-free fill factors.

After accomplishing the correct print conditions, we implemented a nozzle to cool down the extruded material, to further increase the print quality. We also fabricated tensioners for the belts, and a filament drum holder with bearings, to ease the feeding process.

I've had the opportunity to print out a three finger gripping tool designed by Viktor Varga, one of my undergraduate colleagues, for the KUKA KR5, located at the Mechatronical Faculty of the Engineering Department, University of Debrecen. I've used 1,75 [mm] diameter PLA (Polylactic Acid) filament, with its correct heat values visible in the table bellow

The print consisted of the three fingers, and the pushrods and rotary elements as well. Additionally, We've printed a holder for the tool, to kept secure when not in use. All parts have been made with 0,2 [mm] layer thickness, 30 [%] fill factor and 50 [mm/s] print speed.

Fig. 3. Pushrods and rotary element

The Pushrods' print time was: 3 minutes per piece, and the rotary element took 10 minutes total.

The total print time per finger was: 85 minutes, and required supporting material, the additional holder took another 163 minutes. The total print time did not go over 8 hours and their total weight is only 52,2 [g].

Fig. 4. The finished griping tool on the holder

6 Conclusion

After carefully planning out the mechanical frame, I've proceeded to install the electronic components as well. After I've set up the chose programs, I've started to overcome my greatest challenge of configuring, calibrating, and correcting the robot in every aspect. In addition to my theoretical knowledge of the field, I've acquired practical experience in the field of FDM 3D technology. My final task of printing the aforementioned 3-finger gripping tool, I've been able to present my ability to quickly and efficiently use the printer for rapid prototyping.

For future development I want to remove myself as a constant watchman for the printing process, I'd like to implement and Open-Source Linux-based remote surveillance system accessible via Internet, enabling me to print or get real-time footage of the printing process ongoing from anywhere.

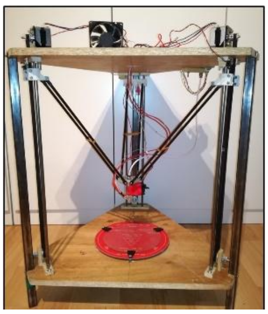

Fig. 5. The finished 3D Printer

Acknowledgment

The publication was supported by the project EFOP-3.6.1-16-2016-00022. This project was co-funded by the European Union, from the European Social Fund.

References

1. Kalmár F, Kalmár T, Csáki I, Husi G „Interrelation between ACH and air temperature distribution in a room," 2009: Proceedings of the 11th International ROOMVENT Conference. Busan, Dél-Korea, 2009.05.24-2009.05.29. Busan: [s. n.], 2009. pp. 911-917.
2. T. I. Erdei, ZS. Molnár, G. Husi, „Selecting Equipment and Supplies for Self–Replicating 3D Printer," Acta Technica Corviniensis - Bulletin of Engineering, Hunedoara9.1, Jan-Mar 2016, pp. 59-62.
3. S. Glaves, „Rocholl (Rostock) Style Delta Robot Kinematics".
4. „Marlin," [Online]. Available: http://marlinfw.org/. [Access Date: 27 11 2017].
5. R. Santos, J. James, T. Chris, S. Marshall, P. Maalouf, „Delta 3D Printer," 2015
6. „Reprap," [Online]. Available: http://reprap.org/wiki/Main_Page. [Access Date: 27 11 2017].
7. „Thingiverse," [Online]. Available: https://www.thingiverse.com/thing:1495442. [Access Date: 27 11 2017].
8. „Autodesk Inventor," [Online]. Available: https://www.autodesk.com/products/inventor/overview. [Access Date: 27 11 2017].
9. T. I. Erdei, Zs. Molnár, N. C. Obinna, G. Husi, „Cyber physical systems in mechatronic research centre," MATEC Web Conf. Volume 126, 2017.
10. „Pololu," [Online]. Available: https://www.pololu.com/product/2133. [Access Date: 27 11 2017]
11. J. Prusa és N. Zadoks, „RepRap Calculator V3," [Online]. Available: https://nathan7.eu/stuff/RepRapCalculator/RepRapCalculator.html. [Access Date: 27 11 2017].
12. „Ultimaker Cura," [Online]. Available: https://ultimaker.com/en/products/ultimaker-cura-software. [Access Date: 27 11 2017].
13. C. T. Brittain, „Ladvien," 2014. [Online]. Available: https://ladvien.com/robots/kossel-mini-calibration/. [Access Date: 27 11 2017].