Design and Implementation of Paste Extrusion 3D printing technology on Articulated Robot Arm

By

Fan Zhu

A thesis submitted in partial fulfilment of the requirements for the award of the degree of

BACHELOR OF ENGINEERING
(Mechatronics)

From

UNIVERSITY OF WOLLONGONG

FACULTY OF ENGINEERING AND INFORMATION SCIENCES

June 2018

ACKNOWLEDGEMENTS

I wish to express my gratitude to the Faculty of Engineering and Information Sciences at University of Wollongong for providing all the necessary facilities and equipment for my research.

A special thank you to my thesis supervisor, Dr Zengxi (Stephan) Pan, for assisting me with my thesis through the duration of my research and providing suggestion towards improvement and completion of my thesis project.

A quick thank you to Mr Daniel Gailer for setting up and maintenance/fixing the ABB IRB 120 robot during the thesis experiment period.

I would also like to take this opportunity to thank my parents for their encouragement and support during my study.

CERTIFICATION

I, Fan Zhu, declare that this thesis submitted in fulfilment of the requirements for the conferral of the award of the degree Bachelor of Engineering Mechatronics, is in its entirety my own work unless otherwise referenced or acknowledged. This document has not been submitted for qualifications at any other academic institution.

Fan Zhu

28/05/2018

ABSTRACT

This thesis looks at incorporating 3D printing technologies/methods with Industrial Robotics. By adding 3D printing technology as a tool to an Industrial Robot, allowing the Industrial Robot to perform 3D printing techniques.

The thesis project also looks at unique materials used in 3D printing and their applicability when used with the Industrial Robot 3D printing process.

The research is aimed to discover new techniques to conduct 3D printing methods as well as the accuracy and feasibility of having two of the technologies, 3D printing and Industrial Robotics combined, in hope of leading to further knowledge/research in both areas of study.

To apply the concept in the thesis experiment, a tool containing 3D printing technology in the form of paste extrusion is attached to an Industrial Robot, the Robot performs a range of movements and various tests are conducted to find results in a range of areas such as settings, accuracy, repeatability, material properties, etc.

The results are documented, calculated, tabulated and graphed to allow the reader to have an easier understanding of these thesis research results.

In conclusion, this thesis research has found that combining 3D printing technology with Industrial Robotics is feasible, the 3D printing results produced by an Industrial Robot 3D printer meet the standards of a 3D printing process. The material used during the project is also an appropriate choice to be used in the 3D printing procedure.

Table of Contents

List of Figures

List of Tables

List of Equations

Chapter 1: Introduction

3D printing is an advancing field of design and research in Engineering and Technology, it is an Additive Manufacturing (AM) method, where the produced results, come from computer-aided design (CAD). [1]

3D printing has only recently gained popularity in the past decade. From only being accessible by researchers and companies to the present day where a 3D printer can be owned personally just like a printer. This relatively new concept was controversial yet still can be advanced further.

Figure 1. "Replicator+ Desktop 3D Printer | MakerBot": A currently available 3D printer [2]

Printing technology existed for centuries, where the most common form of application would be printing with Ink, this form of printing was innovated and advanced for only one purpose, to speed up the process of documenting and copying of images and text.

3D printing differs from previous methods as the technology can produce a range of objects from simple to complex in design, with the patent of each design easily distributable since the initial design comes from CAD. The process of operating the 3D printers are also relatively simple, where in-depth knowledge is not required. [3].

3D printing technology have an endless amount of applications such as producing components for various machines, furniture and other objects, extending to even the medical field, where a recent research paper discusses the use of 3D printing to produce a prosthetic joint of any geometry, matching the patient, by scanning the desired joint though X-ray, MRI or CT scanning. [4].

Charles Hull invent a method called stereolithography (SLA). This technology is the forefront of 3D printing, Charles Hull patented this process in 1984, the method involves polymer/plastic materials in liquid form, converted to solid state in a cross-section format, applied layer by layer using an ultraviolet laser. [5]

In the present day, 3D printing materials not only include synthetic polymers, such as nylon, plastics, resins, ABS, etc. But metals such as steel, gold, silver, titanium, Ceramics and Clay can all be printed. [6] However, the most intriguing and researchable area of 3D printed material/application would be food.

3D printing with food incorporates Solid Free-Form (SFF) method, this method branches into three other technologies, however the focus is the Fused Deposition Modelling (FDM). FDM applies to 3D printing food of materials in liquid state, where the material is deposited without the addition of binding agents and other forms of structuring hydrocolloids. [7]

Recently multiple 3D printers have been developed with specifically food as a material, the design allows basic food types such as chocolate and candy to be printed and constructed, the 3D printing of these various food creations will require a specific deposition fitting. The most common type of 3D printing deposition technology would be the extrusion and direct deposition method.

So, for food 3D printing, the extrusion part will need to allow various food materials to be deposited, without altering the texture, taste and edibility of the food material within the extruder. The part that can match the criteria would be a paste extruder.

A team of researches associated with Fab lab have worked on producing a paste extruder by modifying an existing 3D printer. First, filament extruder, direct drive stepper motor, hot end, inductor probe, and fan were removed from original 3D printer.

These components are then replaced with an aluminium heat block designed for a 60-ml syringe, plunger using the previously removed stepper motor, 1/4 "Acme push rod and nut, two 3D printed gears, temperature sensor placed within heater block and two end stops for mina and max limits. [8]

Figure 2. "Paste & Food Extruder | Printrbot": modified paste extruder from previous 3D printer. [9]

The modified 3D printer showed competency in successfully producing 3D results after testing with clay and chocolate materials.

Another problem would be the method of delivery, once the food object is 3D printed, depending on the type of food material used, cooling and heating will be required respectively per food type. Therefore, what if the food object is printed using an articulated robot arm instead of within the 3D printer? This allows for the location of the creation of the 3D object to be changed to match the required action that needs to apply to the food. This also allows for some special methods of application, e.g. applying on pre-existing food as a platform instead of 3D printing a food object entirely from scratch.

AM introduces possibilities with multiple possibilities in combining with articulated industrial robotics, unlike the traditional 3D printing methods and technologies, with the addition of 6 degrees of freedom (D.O.F) robotic arm, the fabrication path can be designed to follow curved surfaces or other specific physical property requirements. The fabrication path program generated in G-code can be converted to a robot program, the building process can be simulated for the given robot and optimized for the robot's specification. The robot dispensing software and hardware interface will guide the tool tip to follow required coordinates with specific deposition rates. [7]

Another example of this would be the 'Robot Kitchen' developed by Moley Robotics. The concept is to use a pair of robot arms to cook and prepare food,

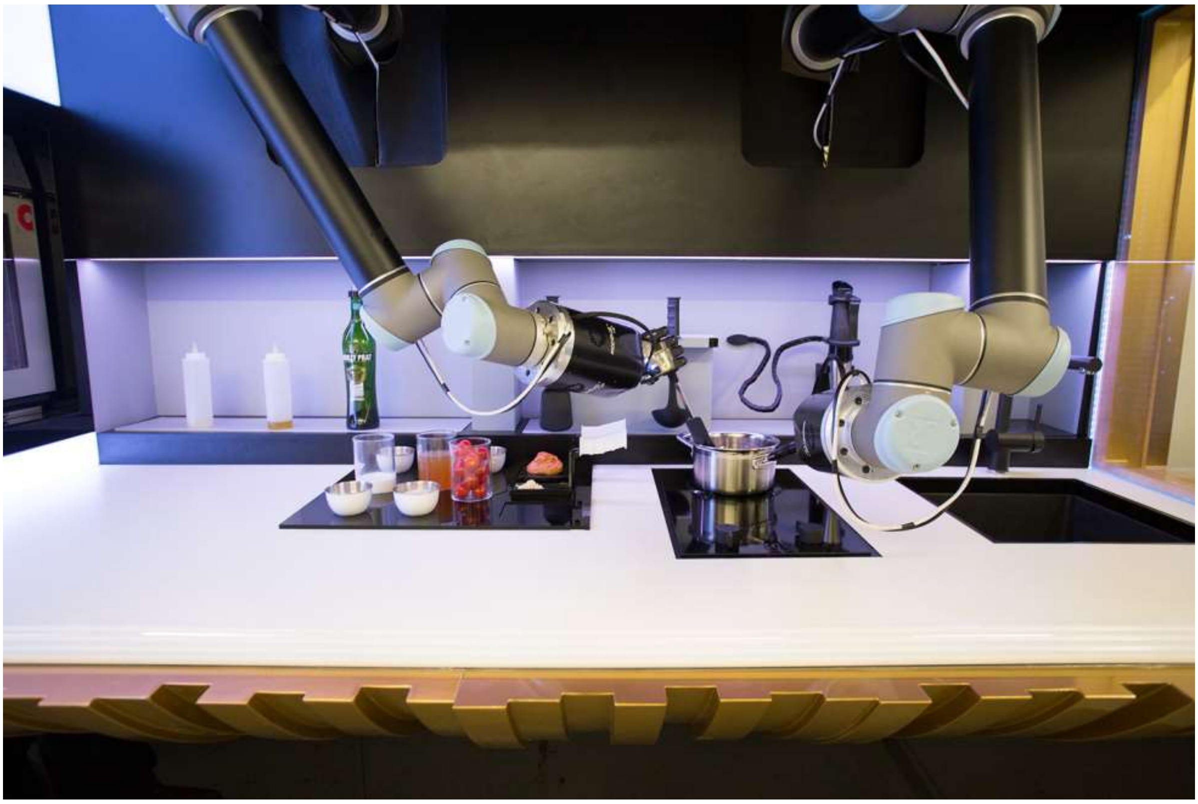


Figure 3."Robotic Kitchen | Moley Robotics": Moley Robotic Kitchen, early in the crab bisque cooking process. [10]

If 3D printing technology can be combined with the precision of articulated robot arms, the result could lead to accurate presentation and preparation of food in the most hygienic and precise environment.

Aim

The aim of this research topic is to examine the applicability of food as a 3D printing material in 3D printing technology and the design and implementation a new method of 3D printing by integrating two existing technologies within the areas of 3D printing and Industrial Robotics. The design will incorporate the use of an articulated industrial robot arm with 6 (D.O.F) fitted with a food paste extruder as the deposition component for 3D printing. The resulting object will be analysed and tested to insure the food upholds its integrity as a material, as well as the functionality of the combined hardware setup, a 3D Food Robot Printer.

Objectives

- Study current existing 3D printing technologies
- Research 3D printing processes involving food as a material.
- Study the technology behind the paste extruder and its functionality during its use as the deposition technology.
- Analyse the material properties and feasibility of food as a material for 3D printing.
- Hardware integration of:
  - Paste extruder, aka. deposition component with control module (e.g. Arduino).
  - Paste extruder with industrial articulated robot arm.
- Software development for:
  - 3D print planning.
  - CAD modelling.
  - Path modelling.
  - Code generation.
  - Code conversion.
  - Industrial robot arm motion and control.
  - Control module connecting to paste extruder, in control of deposition rates.
- Compare food 3D print result with any existing method, study its performance and quality of the end design.
- Investigate the various possibilities the material may offer in the future from 3D print result and outline its potential based on the quality of the product.

Chapter 2: Literature Review

2.1 Introduction

Additive Manufacturing, also commonly known as 3D printing, is a type of industrial fabrication, 3D printing is a relatively new engineering manufacturing method, where the earlier patents for the method were introduced in 1980s. Instead of the common fabrication methods of removing material, through additive fabrication, the material is added layer by layer to produce a result. [11]

In this thesis project, the experiment is designed to allow an industrial robot arm to act as the 3D printer with an attached extruder head with a unique material. This chapter will look at various existing technologies available on the market for 3D printers and components. Combining the technologies with industrial robotics to achieve the desired result.

2.2 Concept

In industry, there are a wide variety of manufacturing and fabrication methods that are being applied and researched in. In relation of 3D printing and manufacturing in industry, we narrow it down to two fundamental fabrication concepts, 'Subtractive and Additive Fabrication'.

Subtractive fabrication is the most common method of industrial fabrication of solid objects, where in its simplicity, material is removed from a given solid block until the desired shape is obtained. [11]

For example, the earlier concept of subtractive fabrications would be the using a chisel and removing material from various stones or minerals such as marble, to produce shapes of complex dimensions and designs, some are the cases of statues shaped to be the same as an entire human body.

Examples of this method can be seen in museums around the world, dating back to the earliest of human civilizations.

The modern methods of subtractive fabrication are operations such as milling, lathing, drilling, planning, boring, cutting and various other techniques that have being developed overtime and are now incorporated in various machines.

With the assist of computers, currently a CNC machine can perform almost all the above-mentioned operations plus others in a highly accurate and speedy manner just within the single machine.

Additive fabrication would be considered the vice versa of subtractive fabrication, the method is like construction, where projects are built layer by layer.

Additive fabrication in general is the addition of material layer by layer to build up the three-dimensional object that is desired. [11]

Specifically, in this thesis, we are concentrating on 3D printing, which follows a principle like the ink-jet printer that spits out droplets of ink on a piece of paper to create images, a 3D printer deposits a variety of material and gradually build the model up in three dimensions.

Without any additional procedures the resulting 3D object is completed, this allows 3D printing to quickly and accurately produce complicated parts and realise real “freedom fabrication”. [12]

2.3 3D printing in Detail

In this section, the general method process of the 3D printing method is explained.

2.3.1 General 3D Printing Method

There are several steps required for producing a resulting 3D object through additive manufacturing aka. 3D printing.

2.3.1.1 CAD

Computer Aided Design (CAD) and Drafting (CADD) is a type of computer software that allows the user to create design and design documentation on the computer, replacing the manual drafting process with a digital and automated process. Currently one of the biggest CAD software developers is a company called AUTODESK. [13]

3D printing relies on the capabilities of the CAD software, either by producing a 3D design using the CAD software itself or by scanning a 3D project, converting and generating the design in the CAD software, both cases are the first stages of a 3D printing production.

2.3.1.2 STL files

STL Files (*.stl) relates to stereolithography, one of the 3D printing processes, the file format derives from this and is a file format accepted within CAD software. Another name is rapid prototyping, the parts are created using a faceted mesh representation in STL files. [13]

After a design is produced in the CAD software, the model is converted to the STL file format. STL is the standard format, accepted in nearly every additive manufacturing machine, the file describes the external closed surfaces of the original CAD model and forms basis for calculation of the slices. [14]

2.3.1.3 Transfer to 3D printer and STL file editing

Once the STL file is produced, the file is transferred to the 3D printer, after the file may require additional editing onboard the 3D printer so the correct sizes, position and orientation prior to building have being met. [14]

2.3.1.4 Machine setup

The 3D printer must be set up before the build process commences, various settings will need to be edited to suit the transferred design from the STL file, such as material constraints, energy sources, layer thickness, extrusion parameters, etc. [14]

2.3.1.5 Build

The 3D printer is designed to build the three-dimensional object automatically, [15,16,17] therefore supervision is not required, however, some monitoring of the machine is needed during the process to confirm the condition of the machine, making sure the machine is functioning smoothly without errors.

2.3.1.6 Removal

Once the model is produced by the 3D printer, during the stage of removing the object, check that the components of the 3D printer have finished moving and if temperature change were involved, make sure the object and surrounds are properly reset including the temperature. Once everything is cleared, proceed with the result checking and removal. [15,16,17]

2.3.1.7 Post process

After the resulting 3D model is cleared from the work space of the 3D printer, carry out necessary repairs and maintenance of various components of the 3D printer ready for further production. [17]

2.3.2 common 3D printing techniques

Stereolithography (SLA)

Stereolithography, patented in 1986 by Charles Hull of 3D systems, is one of the earliest rapid prototyping method. [18] The method is designed to produce solid models by printing layers of curable materials on top of each other, the material is UV curable, so by using a UV light beam programmed to spot at the surface of the material, the method forms a solid cross-section of the model at the surface of the material. The process is repeated until layer by layer the material forms a three-dimensional shape. [19]

There are two primary process to building the three-dimensional object using SLA, the bottom-up approach features resin on a movable build platform, during printing the platform begins with a thin layer of the resin at the surface, as the laser scans the surface to cure a layer, the platform then moves down so a new thin surface of resin is atop of the newly cured layer of material. The laser movement used in this process is defined as projection-based stereolithography (PSL), the laser prints each layer with a spot beam like the method in engraving. [20]

The top-down method is similar except the platform begins at the bottom of the resin reservoir, as the platform moves up layer by layer of exposing resin, a laser scanner, situated at the base of the system, scans each surface as the platform moves. This process uses scanning-based stereolithography (SSL) laser movement, where the laser scans the pattern onto the surface of each layer. [20]

Fused Deposition Modelling (FDM)

Fused Deposition Modelling was first patented by S. Scott Crump in 1992, the apparatus involves a movable dispensing head provided with material that solidifies at a certain temperature, pre-set for the work. The material is provided in a fluid state layer by layer, the dispensing nozzle moves in the ‘Z’ axis, while the base member where the model will form on, moves in the ‘X’ and ‘Y’ axes, as the material is dispensed, the base member and dispensing head moves relative to each other, with movements driven by mechanical motors. The motors are driven by the computer system with respect to the CAD software, as each layer of material is produced, the pattern corresponds to the CAD model. [21]

FDM can be seen used in majority of current 3D printing systems, however, FDM is limited to extrusion of thermos plastics at designed temperatures, these plastics include polyamide, ABS, PLA, polypropylene, polyether esters and others. Furthermore, researchers are still addressing the effects of various components of the FDM system such as filament feed, nozzle design, melt viscosity, shear thinning, crystallization rate and others to improve the surface finish, dimensional accuracy and mechanical properties of the finished product. [22]

3D Extrusion

Extrusion types of Additive Manufacturing involves layer by layer deposition of materials in various degrees of liquid states. While FDM mentioned above is limited to various polymer materials, 3D extrusion caters for an additional variety of materials including thermosets, rubbers, silicones, organic and inorganic pastes, latex, and more, the solidification process is achieved differently depending on the material, but mostly left to its natural chemical states and processes. [22]

Since this thesis focus around 3D printing with food materials, 3D extrusion is the best process to apply to the project. With food in paste form, the method can be achieved using a type of paste extruder.

2.4 Hardware Theory

2.4.1 Available Hardware

2.4.1.1 3D Printers

Currently the field of Additive Machining or 3D printing is an expanding field, with the 3D printers that are now commercially available, there are almost as many variety as inkjet printers. [12]

The sizes and shapes of 3D printers vary; however, they are designed in a similar concept, similar to an industrial gantry robot. Some printers have enclosed casing for additional functions, while others have an open set up to allow for various tool tips to be fitted. [11]

Majority of the 3D printers on the market are designed to be desktop sized, same as the conventional printers, however the height of 3D printers is much greater than the normal inkjet printers. Within the 3D printer range, there are generally 2 or 3 size variants per company and designed to various capability. [16,17]

Figure 4. "Ultimaker 3 Desktop 3D Printer" | Ultimaker [15] (right), "da Vinci 1.0 Pro Desktop 3D Printer" | XYZprinting [16] (left)

2.4.1.2 3D printing component Tool Heads

Out of all the components required for 3D printing, the most relevant component for this thesis project is the tool head, as this is the component that is directly related the material to be used, for this project design, food types are the materials selected, the following chapters and topics will discuss materials and functionality in further details.

In consideration of the topic on tool heads, one must relate the specific tool head design with the additive manufacturing technology it is designed from. Out of the 3 most common kinds of 3D printing techniques, 3D extrusion tool heads would prove most relevant for this application. With food material in any liquid form, the 3D extrusion tool head will be able to deposit the material in various desired CAD designs.

Coming across various tool heads from different companies, there are only a few relevant tool heads that match the needs for this project design. Specifically, the tool head need to be detachable and independently controlled without a company specified software, since the entire 3D system will incorporate an industrial robot arm for the movement in all 3 XYZ axes. Currently a small variety of 3D printers designed with paste materials exist, however even between these printers, they are catered for a singular entity for 3D printing the result, with no clear design to show capabilities of detachability of their tool head, the 3D printers are also advertised as one printer, so the purchase of singular tool head attachment is difficult. [15,16,17]

2.5 3D Printing Material: Food

There are plenty of materials to select from when choosing to conduct 3D printing technology, [6] among the current commercialised 3D printers, (examples discussed in section 2.4.1.1), majority of the 3D printers use various polymers as a material, specifically, they use ABS and PLA the most. [15,16,17] Amongst many material choices, food is chosen for this project.

2.5.1 Food Types and Methods

3D printing food has been a developing research topic in the Additive Manufacturing (AM) scene, a term Food Layered Manufacture (FLM) is now used to describe the process, stemming from AM. FLM is different to the normal manufacturing methods in the food industry, the process is of AM technology, allowing creation of various food types with different materials. [23]

Currently various FLM technologies, designs, concepts and prototypes have being produced by various research companies and groups, these research topics involve a range of food materials.

In 2001, (Yang et al. 2001) developed an invention capable of producing complex-shaped three-dimensional food objects, such as a custom-designed birthday cake. The method uses AM technology, where food object is printed layer by layer, dispensed from a nozzle and deposited onto a support member. The materials are dispensed in a fluent state where the food composition is composed of liquid ingredients. The material itself has the properties capable of sustaining itself under its own weight as well as subsequent layers. [24]

In 2006, Fab@Home, an opensource 3D printer capable of operating with various materials was introduced, the DIY printer was capable of normal 3D printing functions, as well as printing shapes with frosting, Nutella, and chocolate. The methods used to produce these shapes is still the same as any 3D printer. Various models of the Fab@Home 3D printer have been used in research topics to produce satisfactory results. [25, 27, 47]

2006 - 2009, Windell Oskay and Lenore Edman of Evil Mad Scientist Laboratories designed and constructed 3D printers with Sugar as the material under The CandyFab Project. These machines are capable of printing large objects out of pure sugar, by melting sugar grains together with hot air. A modified hot air rework module (modified soldering iron) was used to produce a low-velocity beam of hot air, the heat fuses selective print media grains forming a two-dimensional image out of the medium. Following that, another layer of the material and the new layer is then applied to the heat again, fusing shapes in the new layer as well as layers below. The method is repeated layer by layer until a three-dimensional object is produced. [26]

In 2010, A research team (Hao et al, 2010), based in University of Exeter, UK, put forward a concept of applying AM methods specifically to chocolate, labelling the process as chocolate additive layer manufacture (ChocALM). The experiment evolved the use of specific chocolate type in the form of milk chocolate buttons, from Cadbury Ltd. The chocolate is processed and tested to a set of temperature ranges and applied into the material chamber, the material is then extruded through a deposition head, combined with X-Y-Z positioning system to allow accurate deposition of chocolate on a platform with precise geometry. Various shapes of specific dimensions were designed for testing and measuring of the three-dimensional chocolate printout. The results of the experiment found chocolate at a range between 32° and 40° provided the most consistency with a range of 3.5 – 7 Pa.s for ChocALM. [28]

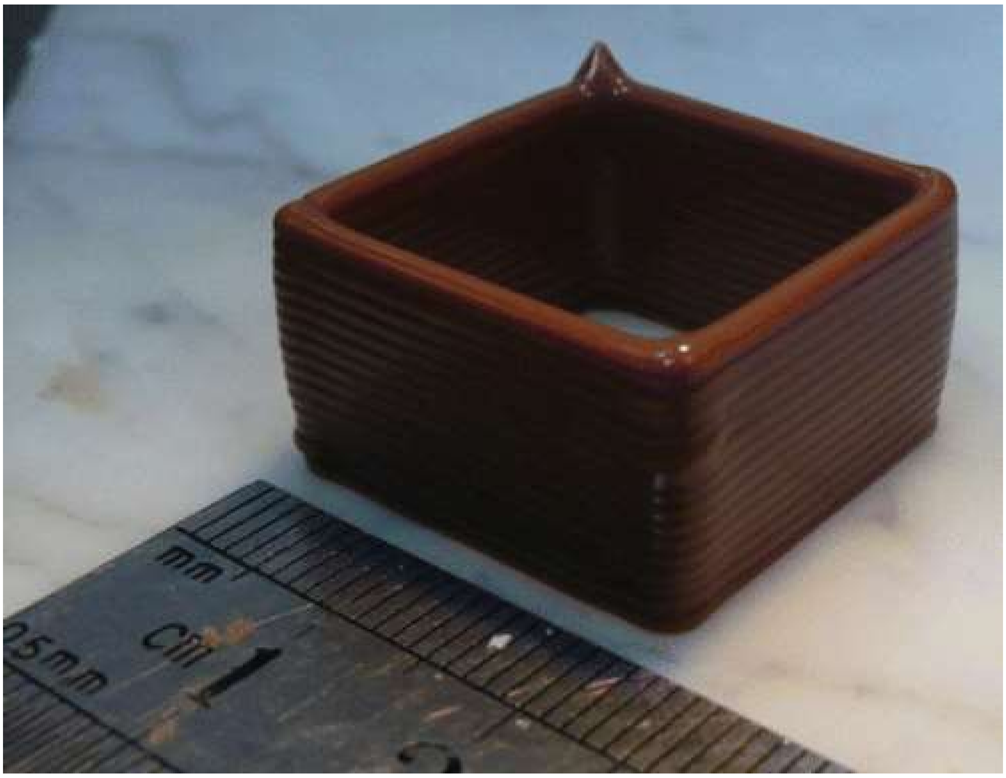


Figure 5. 3D print result of ChocALM in the shape of a square, measured for geometry accuracy | (Hao et al. 2010) [28]

The experiment by (Hao et al. 2010) were primarily designed from and for AM machines, while finding important research data based on applying the ChocALM technique, the accuracy cannot be carried over to this thesis topic as the machine to be used for the experiment will be an industrial robot.

However, with the deposition method of both (Periard et al. 2007) Fab@Home experiment and (Hao et al. 2010) experiment having been an extrusion head with a nozzle aperture, the findings relate specifically to that method of deposition. [25, 28] Some of the results of the findings can be applied to this thesis project, such as testing methods to acquire sufficient temperature range

[28] for the potential material and later incorporate the findings to discover a suitable range of deposition rate for the paste extrusion tool head used for this thesis project.

Recently in 2017, (Yang et al. 2018) carried out an experiment with lemon juice gel as a material for 3D printing. The experiment first looked at the proposed material and its properties, with Lemon juice stored at 4°C, the juice is mixed with starches, afterwards the mixture is steam cooked for 20 min, cooled to room temperature before being stored at 4°C. The resulting material is analysed for its material properties, and once data is collected, the material is put through a 3D printer fitted with a feed hopper with auger mixer and conveyor, extrusion system and a X-Y-Z positioning system featuring stepper motors. The three-dimensional models to be printed are designed with various geometries from cylinders, cubes, lines and even complex shapes such as an Anchor, Gecko and Snowflake. (Yang et al. 2018) mainly looked at the material properties of the lemon juice gel pre-printing and post-printing, checking for hardness, cohesiveness, springiness, gumminess and overall structural integrity. The results showed the lemon juice gel to be suitable for 3D printing with the experimental nozzle diameter of 1mm and extrusion rate of 24 mm^3/s. [29]

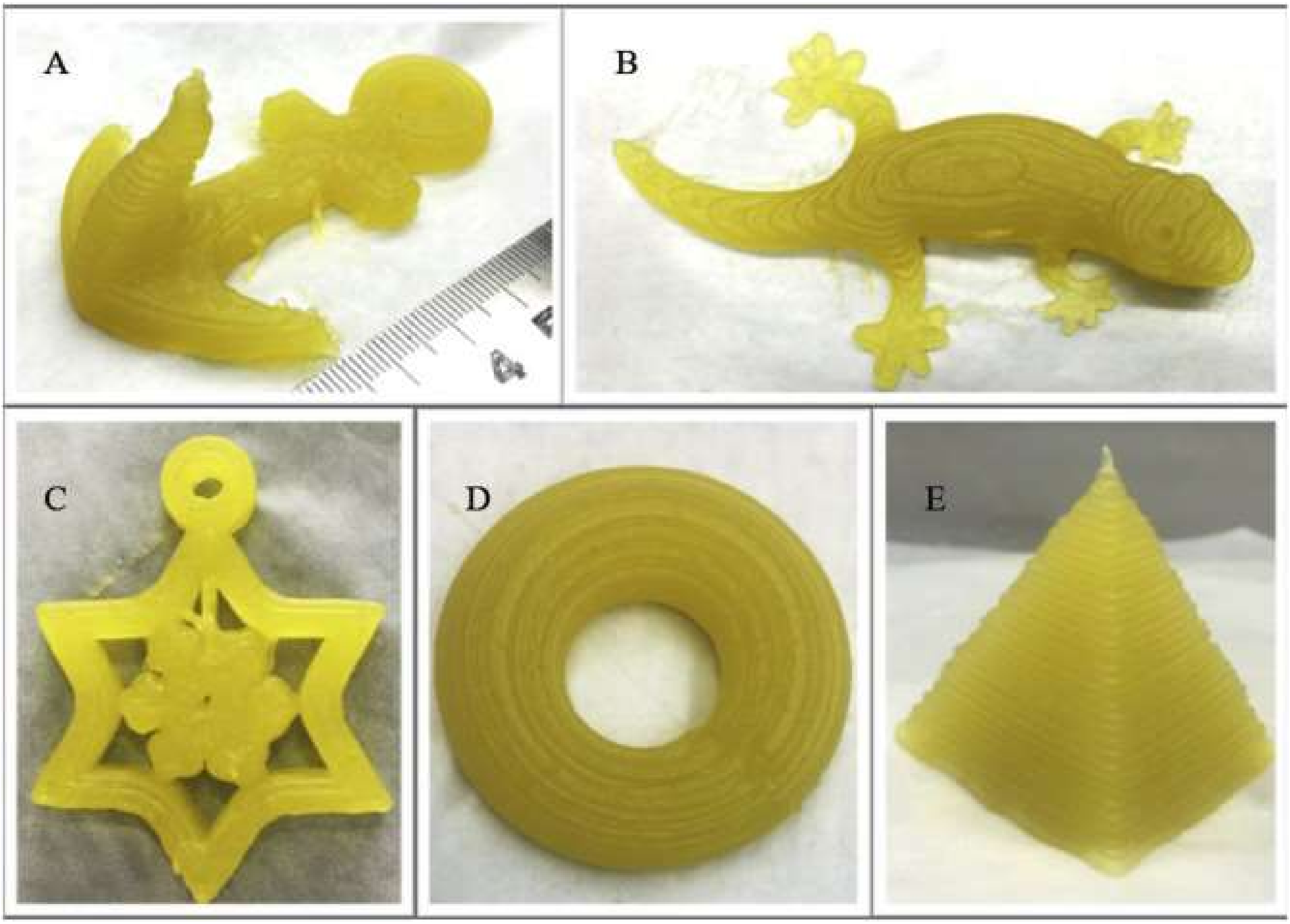


Figure 6. 3D Printed lemon juice gel samples A. Anchor, B. Gecko, C. Snowflake, D. Ring, E. Tetrahedron | (Yang et al. 2018) [29]

2.5.2 Applications

The applications of the having food as 3D printing material extend to a wide range of field of studies. Areas such as military and space food, to allow readily prepared meals for soldiers on demand on the battlefield, for NASA astronauts during long space missions. Food for the elderly is another research area, with countries such as Japan, Sweden and Canada facing up to 60% of nursing home residents and 15-25% of elderly people age 50 and over, suffering from chewing and swallowing difficulties. [30,31] Thus, porridge like food is provided instead, leading to poor appetite and nutritional deficiency. 3D printing can resolve the issue by personalising manufactured food with desired textures and tastes. The biggest application is the confectionery market, with multiple companies already dabbing into the scene with their 3D printers [26, 28, 32, 33, 34] making specially designed sugar models, chocolate constructs and other candies. [31]

One of the largest application of 3D printed food materials would no doubt be the food industry, not including the manufacturing side of the food industry, since 3D printing is Additive Manufacturing, [3, 11, 12, 14, 23, 25] but the focus of using industrial robot as the machine for 3D printing leans towards the use of 3D printing as a tool to replicate tasks of the professions in the restaurants and catering sector. Tasks such as decorations and their designs conducted by pastry chefs, plating designs for service and specially created food menu items from 3D printing can be replicated by the AM machine designed in this thesis subject.

An example of 3D printing technology in the restaurant and catering business is from Michelin star chef Paco Pérez from Catalan, Spain, recently worked with 3D food printing technology from a company called Natural Machines, and their product 'Foodini'. The machine is identified as a 3D printer under Kitchen Appliance, [35] the chef uses 3D printing technology to produce various food designs for plating ready for service, Lynette Kucsma, the cofounder of Natural Machines and 'Foodini' states in the article (Tufvesson. 2017) "Imagine you're a Michelin-starred chef and you have a dish that requires breadsticks in the shape of tree branches and you have to serve 50 people that evening – you can automate that process with Foodini". Kucsma, in the article (Tufvesson. 2017) also claims for the printer to produce pasta, burgers, pizza, brownies and chocolate sculptures. [36]

The article shows labour intensive tasks with delicacy as well as speed required can be reproduced by 3D printing technology.

A highly unique application was developed by a team (Zhao et al. 2017) based in china, the team developed a method to extract 3D print pathing from a photograph portrait, to allow the 3D print result to resemble the portrait printed in a jam material. [37] The idea originates from sugar painting (糖画), an ancient Chinese folk-art dating back to the Ming Dynasty, it involves the artist creating images of animals on a bamboo stick. [38] By scanning a portrait graphically and processing the image through image processing, image abstraction and path optimisation, the result allows the 3D printer to accept the pathing design and deposit jam accurately. The method not only works with photographs but also extend to drawings and sketches. The results provide an effective framework that automatically create food images from input pictures after steps of various image processing techniques. [37]

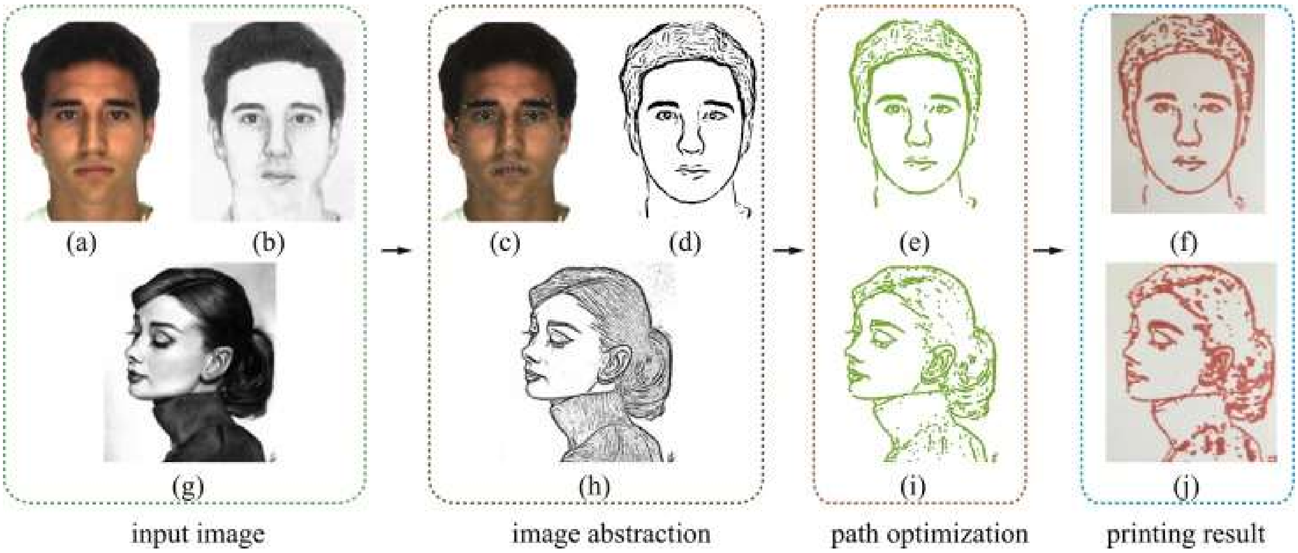


Figure 7. Overview of framework and the steps used to fabricate 3D printed image | (Zhao et al. 2017) [37]

By creating this image processing framework to allow images to translate to pathing data for the 3D printer, the method can be optimised for other shapes and designs, by simply scanning an object and replicating the result with 3D printer's tool path.

It is potentially possible to incorporate this concept into the Industrial Robot 3D Printer Design of this thesis project, allowing the robotic arm tool tip to follow a generated path from scanning an object or image.

2.6 Thesis Project Experimental Hardware Research

This section contains research for hardware components to be used in the thesis experiment's hardware setup.

The research involves each component's functionality within the hardware design, the reason each component is chosen and an estimation of their performance once the experimental hardware setup is complete.

2.6.1 Industrial Robot Arm

Industrial robots are programmable machines designed to resemble human movement, the robot is programmed to complete sophisticated tasks that can replace a human worker.

An industrial robot consists of a mechanical manipulator and a controller to move it and perform other functions, consisting of joints and links that can position the end manipulator relative to its base. [39]

For this project, the industrial robot arm to be closely worked with is the ABB IRB 120 robot. [40] The robot will be programmed to move in a path that the 3D printing extruder will follow, when attached to the robot.

The concept of incorporating Additive Manufacturing (AM) technologies with Industrial Robotics is not new, however there have being only a few research topics based on this area in the past decade.

In 2015, (Zhang et al. 2015) developed a Robotic Additive Manufacturing System. The system composed of a standard industrial robot, commercially available extruder, heating pad and extrusion flowrate control hardware and software. [41]

The system uses paths generated from a software that is designed from the structural geometry and surface quality requirements, the path is converted from G-code for AM use to a robot path program. The extrusion rate control works separately to the industrial robot control, first a robotic dispensing software is used with its analog I/O interface, the software provides analog signal to match the speed of the robot tool tip. [41]

Analog signal inputs to a pulse generation card that creates specific frequencies based on input voltage, then the signal is input to a stepper motor drive to control the extrusion rate of the extrusion head. [41]

The results of the research show the possibilities to combine AM extrusion technology with Industrial Robots and the software for both technologies also work together though conversion.

The research also demonstrates the benefits of using a 6 Degrees of Freedom (DOF) articulated robot arm to apply material to paths normally difficult for the AM technology. [41]

(Brooks et al. 2016) Published the design, development and testing of a Robot-assisted 3D printing of biopolymer thin shells technology. The system uses Fused Deposition Modelling (FDM), an AM technology to extrude pellets of biomaterial on organically shaped 3D curve surfaces. [42]

Due to the work surface to be curved, traditional flat platforms used for FDM cannot be applied, thus a 6-DOF industrial robot arm is used for the X-Y-Z movement. [42]

As the printing process being different to normal, conventional data formats are not applied directly, however the model will now be required to be converted to data formats that the industrial robot can work with. There are two main stages, model conversion stage and printing stage. [42]

The model conversion stage converts the design model file to robot target file. The printing stage use the target file to move the robotic arm along the required trajectory, as well as controlling extrusion for the FDM. [42]

2.6.2 3D Extrusion Tool

The best tool head to extrude food material for 3D printing is the ZMorph Thick Paste Extruder, the extruder tool head has printing speeds ranging from 10 mm/s up to 100 mm/s, working with a range of materials with medium to dense masses such as silicone, porcelain, ceramics and pastes of chocolate, cheese, dough and other food pastes. [43]

The specifications are given as:

Table 1. "Thick Paste Extruder Properties | ZMorph" [43]

Construction:	**High-quality 3 mm aluminium frame**
Default work speed:	10 mm/s (depends on the material)
Nozzle:	2 mm / 4 mm
Motor:	Nema 17 stepper
Extrusion:	Direct Drive
Work Area L × W × H:	250 × 235 × 165 mm (without covers)
Recommended Printing Resolution:	2 mm layer height
Material capacity:	100 ml

Figure 8. "Thick Paste Extruder | ZMorph" [43]

This tool head is an ideal choice the thesis project, the extruder is designed for the ZMorph 2.0 VX 3D printer, [17] however, the extruder head is capable of being independently configured at the motor connection, since the motor is give as Nema 17 stepper motor, for direct control of the stepper motor is possible with a motor controller such as a stepper motor shield.

Unfortunately, the exact Nema 17 stepper motor specification is not given by ZMorph's website.

2.6.3 Microcontroller Development Board (Dev Board)

The purpose of the Microcontroller Development Board (Dev Board) in the 3D Robot Printer setup is to act as the medium for programming the motor controller that will drive the motor on the extruder tool.

In the case of the Thick Paste Extruder from ZMorph, the motor is a stepper motor, so a stepper motor shield is used as the motor controller, this motor controller must be compatible with the Dev Board chosen for the 3D Robot Printer's Hardware Setup.

The potential microcontroller dev board for this application is the Arduino Uno Rev3 (R3), by the Arduino company. The Arduino Uno is a microcontroller board based on the ATmega328P. The microcontroller is a programmable module supported with the Arduino Software (IDE), to allow new code to be uploaded and programmed, communicating through original STK500 protocol. [44]

Figure 9. ARDUINO UNO REV3 | ARDUINO [44]

2.6.4 Stepper Motor shield

The easiest way to drive a stepper motor is with a stepper motor shield set up, a potential motor shield that can be successfully used is the Adafruit Motor shield comes with the following specs related to the project:

- 4 H-Bridges: TB6612 chipset, providing 1.2A per bridge. Running motors on 4.5VDC to 13.5 VDC.
- Up to 2 stepper motors with single coil, double coil, interleaved or micro-stepping

- Motors automatically disabled on power-up
- Tested compatible with Arduino UNO

The motor shield is tested and assembled beforehand, allowing easy assembly with the Arduino Uno Rev3 Dev Board, the motor shield itself will provide all the capabilities to drive the Nema 17 stepper motor on the ZMorph Thick Paste Extruder.

The motor shield is compatible with various NEMA stepper motors listed by the manufacturer, Adafruit, including a range of NEMA 17 stepper motors. [45]

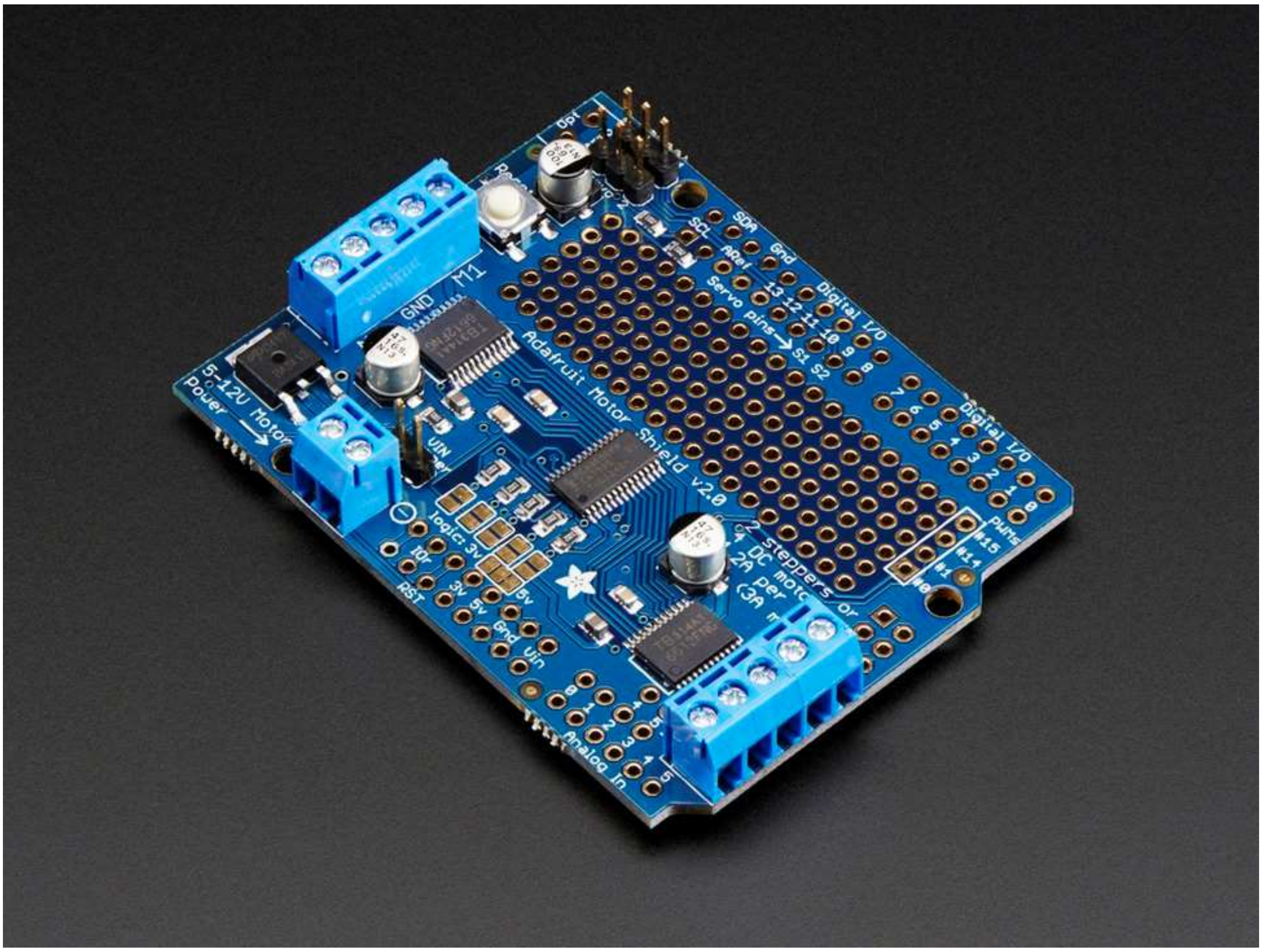


Figure 10. Adafruit Motor/Stepper/Servo Shield for Arduino v2 Kit - v2.3 | Adafruit [45]

Chapter 3: Experimental Equipment

3.1 3D Robot Printer Hardware

3.1.1 Industrial Robot Arm

3.1.1.1 ABB Industrial Robot: IRB 120

The IRB 120 is one of the smaller industrial robots designed and manufactured by ABB Robotics. The robot is equipped with the IRC5 controller and robot control software. [40]

IRB 120 features 6 manipulator axes allowing 6 Degrees of Freedom (D.O.F):

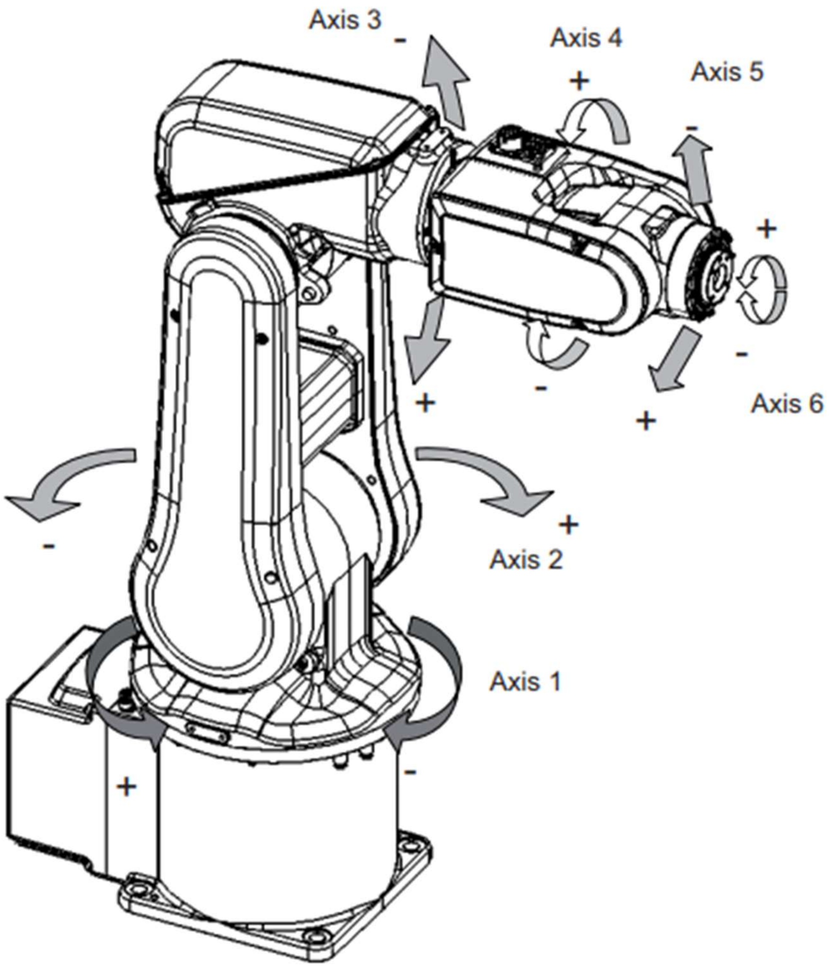


Figure 11. Manipulator axes | ABB [40]

The IRB 120 is used for majority of the movement component of the setup, following a path designed for the robot through programming. The extruder is fitted to the robot as a tool, the robot will move along the designed path to match the designed shape while the extruder is in operation.

The final operation speed of the robot is determined via design after several tests conducted during experimentation stages of the thesis in relation to varying speeds of the extruder.

3.1.1.2 IRC5 with FlexPendant

As mentioned in section 3.1.1, the industrial robot IRB 120 comes with an IRC5 controller with FlexPendant. The IRC5 is connected to the PC station via ethernet, communication with the PC is implemented in the online option within the program RobotStudio® and all programming aspect will involve both the RobotStudio software aswell as the FlexPendant. More information on RobotStudio will be explained in the software section 3.2.

The robot can be controlled manually or automatically via the FlexPendant that is connected to the ICR5. Many tasks include running the programs, jogging the manipulator and program editing. [46]

All calibration will involve exclusive use of the flex pendant, ranging from calibration of the extruder tool and troubleshooting events that may occur, since the tool is not a standard industrial robot tool.

3.1.2 Food Extruder Tool

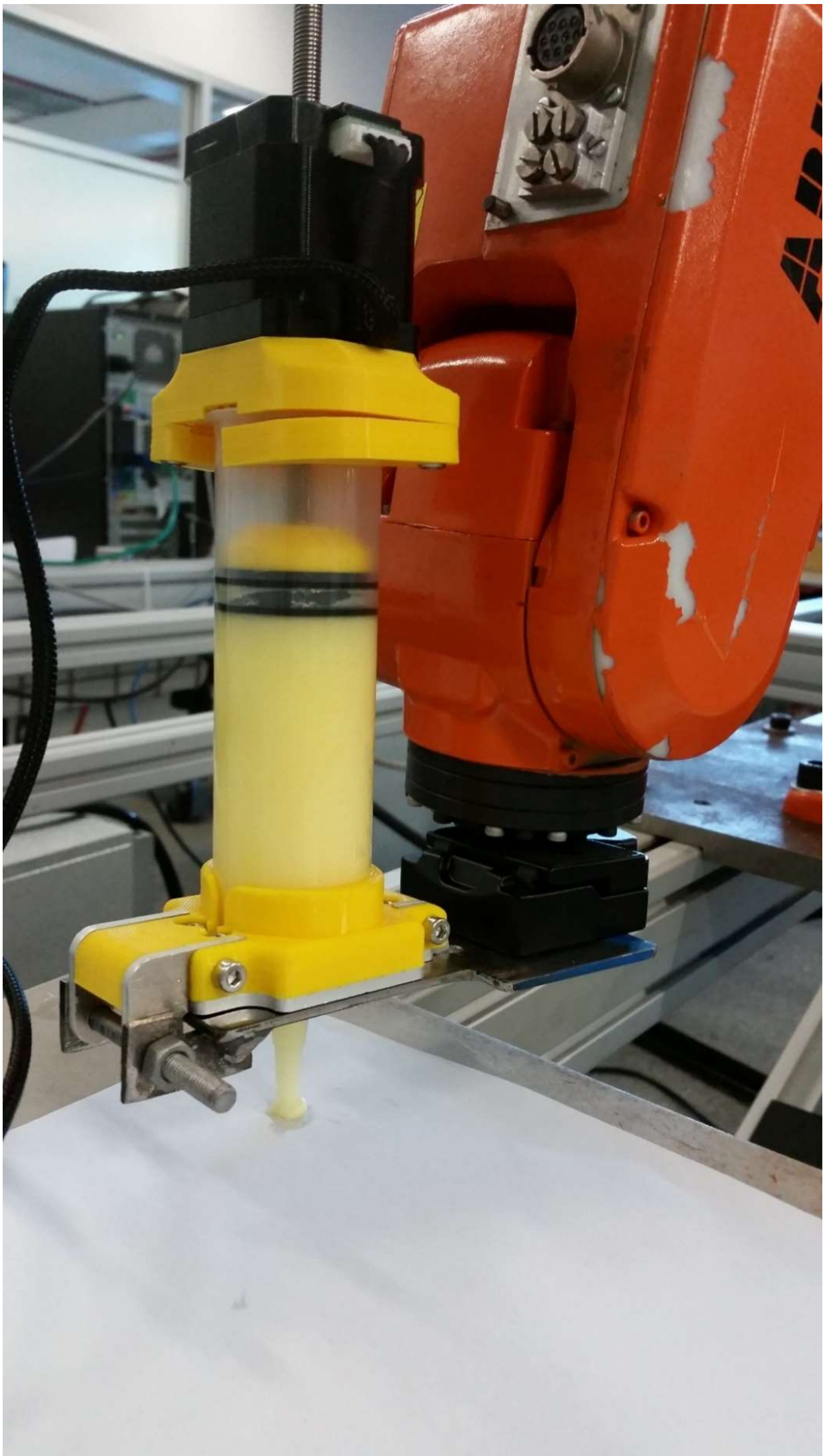


Figure 12. Extruder Tool Full Configuration (Attached to IRB 120)

The extruder tool consists of three major components, designed to be connected onto the industrial robot IRB 120 as a moving tool, similar in design to a welding gun.

3.1.2.1 Thick Paste Extruder

The main component of the extruder tool to be fitted onto IRB 120 is the same Thick Paste Extruder from ZMorph, first mentioned in section 2.5.2. The extruder's motor is controlled by a Microcontroller + Motor shield combination, fitted to a mount design that is connected to the IRB 120 with a quick release rectangular plate adapter.

The motor comes with the part number 17LS19-1684N-200D, which indicates the motor is indeed a NEMA17 stepper motor, matching the data given in the datasheet by ZMorph, however, the part number gives the specific type of NEMA17 stepper motor, which is the non-captive linear actuator.

The linear actuator is a bipolar motor, with the following specifications:

Table 2. NEMA17 non-captive linear actuator (17LS19-1684N-200D) specifications [47]

Electrical Specifications

Bipolar/Unipolar	Bipolar
Holding Torque (Ncm)	44
Inductance (mH)	2.8
Motor Type	Non-Captive
Phase Resistance (ohm)	1.65
Rated Current (A)	1.68
Step Angle (°)	1.8

Physical Specification

Body Length (mm)	48
Frame Size (mm)	42x42
Lead Screw Diameter (mm)	6.35
Lead Screw Length (mm)	200
Lead Travel/Revolution (mm)	2
Lead Travel/Step (mm)	0.01
No. of Lead	4
Lead Length (mm)	400

The wire connector of the motor has the following pinout on the plug, designed by ZMorph.

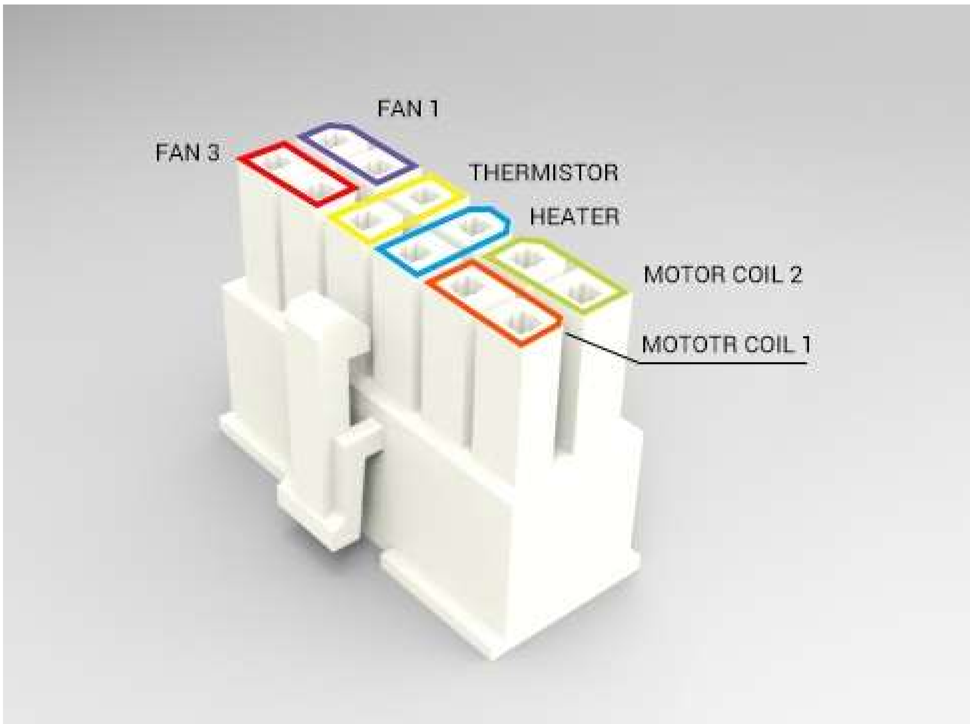


Figure 13. pinout diagram for stepper motor plug | ZMorph [48]

3.1.2.2 Quick Release Rectangular Plate Adapter

A camera quick change rectangular plate adapter is used for easy setup and removal of the tool from the industrial robot. The rectangle plate on the top (figure 26) requires a M5 screw, this is the component connected to IRB 120. The base of the adapter is fitted onto the Mounting plate along with the Thick Paste Extruder.

Figure 14. Quick Change Rectangular Plate Adapter, universal threads | Manfrotto [49]

3.1.2.3 Mounting plate

The mounting plate is designed to allow the Thick Paste Extruder to be fitted with the nozzle and clip ends of the extruder firmly latched onto the plate. The extruder is also secured using an M6 bolt and nut. The base of the quick release rectangular plate adapter is to be fitted at an adequate distance apart from the extruder, connected via a M8 Bolt.

The plate dimensions also consider for when the tool is mounted onto IRB 120, the tool has room for the robot’s axis 6 rotation.

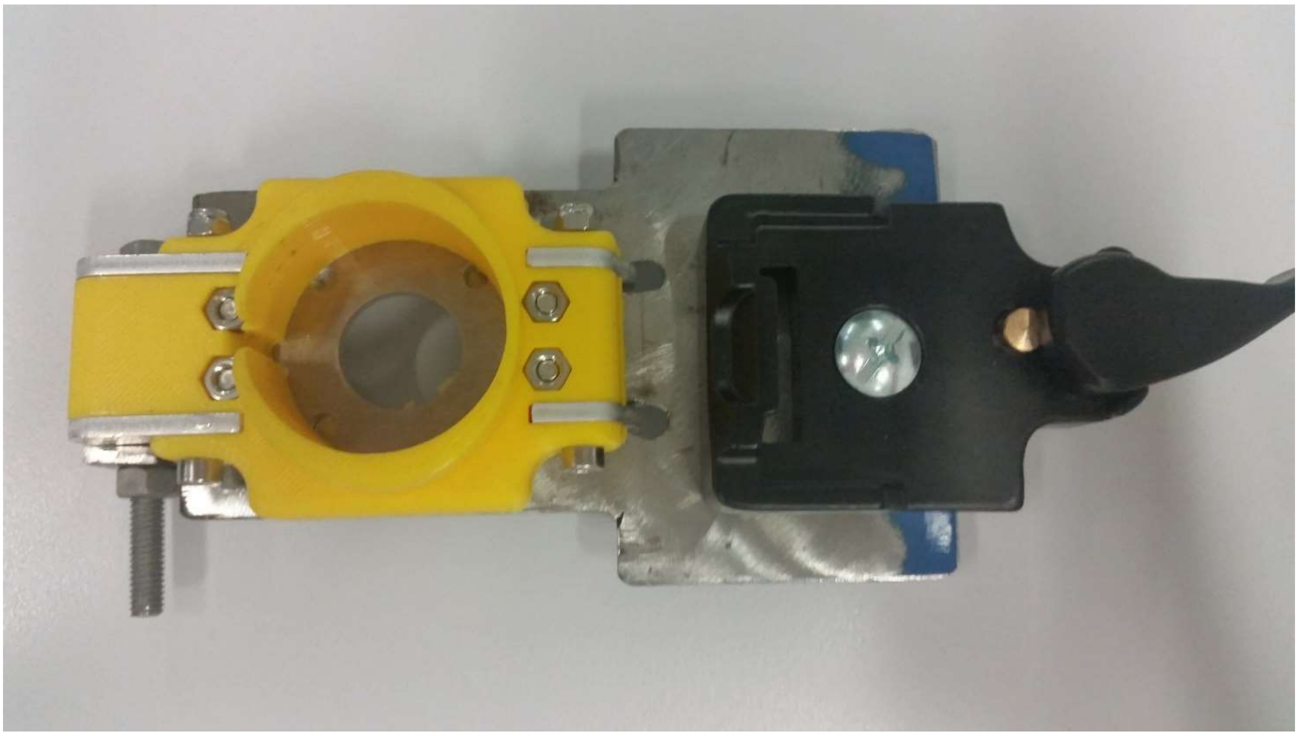


Figure 15. Mounting plate top view, Thick Paste Extruder component + Rectangular Plate Adapter attached.

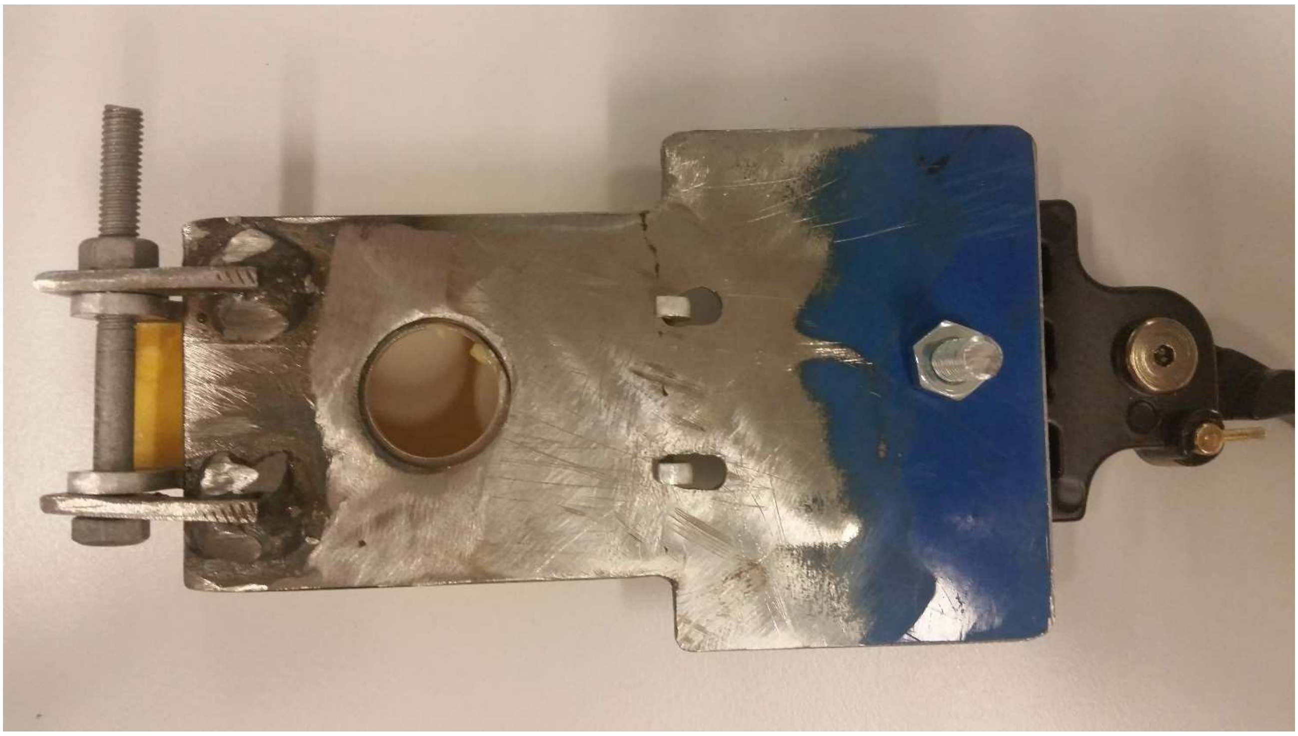


Figure 16. Mounting plate bottom view.

3.1.3 Extruder Motor Controller

The Thick Paste Extruder identified in section 3.1.2.1 is explained to have a stepper motor.

To drive the stepper motor, a motor shield and microcontroller set up is used.

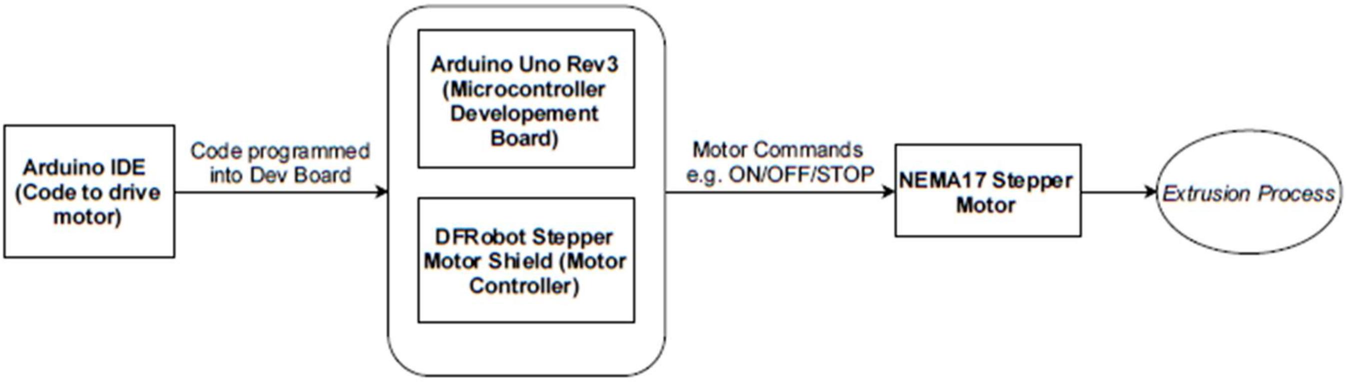


Figure 17. Extruder Motor Controller Setup

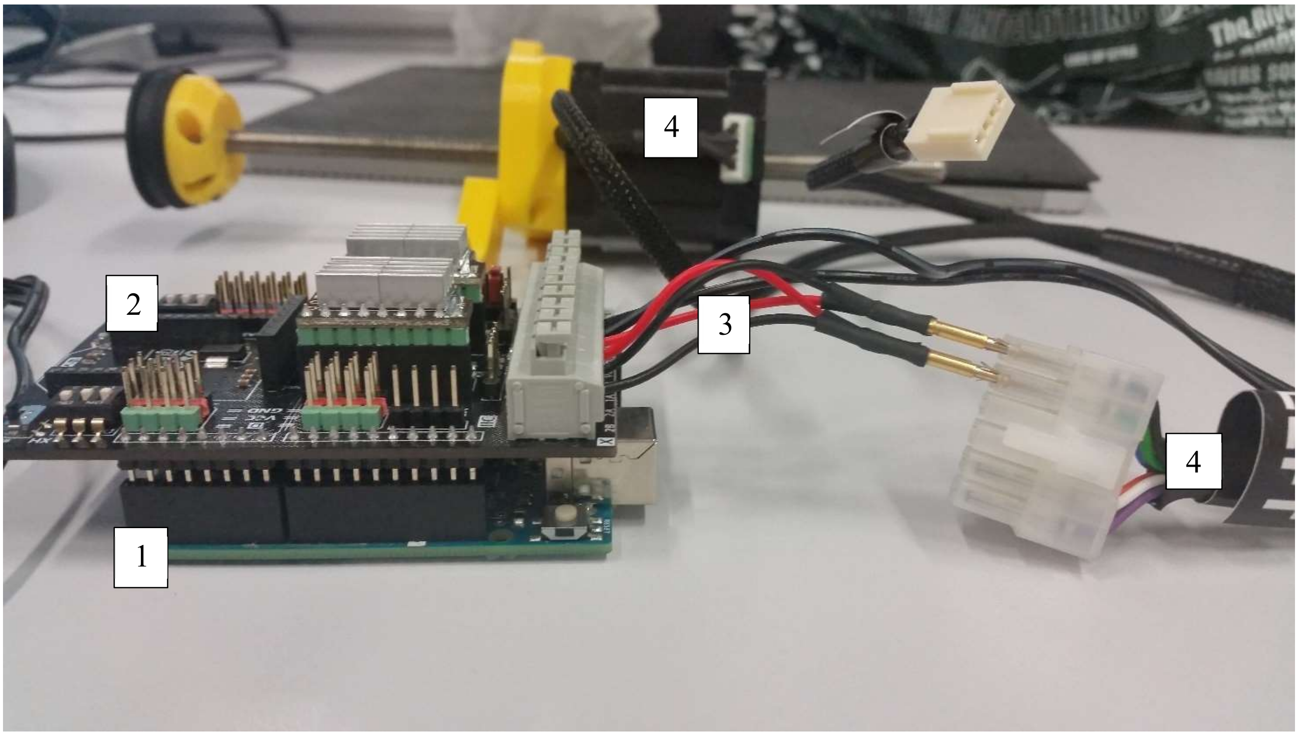


Figure 16. Arduino UNO Rev3 (1), DFRobot Stepper Shield (2), Wiring to Motor Pinout (3), Thick paste Extruder Motor (4)

3.1.3.1 Microcontroller

Arduino Uno Rev3 (R3) is the microcontroller dev board used for the experiment, same as the microcontroller initially researched and proposed in section 2.4.2.2 of chapter 2.

The Uno R3 is a standard microcontroller board containing an ATmega328P microcontroller microchip along with a USB interface. [44]

3.1.3.2 Motor Shield

Adafruit Motor/Stepper/Servo Shield for Arduino v2 Kit – v2.3
Initially Adafruit motor shield proposed and researched in section 2.4.2.3 of chapter 2 is used, however once the Thick Paste Extruder arrived, after looking up the motor part number, it is understood that the Adafruit motor shield could not drive the motor properly as the chipset used (TB6612) only provides 1.2A per bridge (3A for brief 20ms peaks) [45].

When setting up the motor shield for initial motor testing, the motor shield showed signs of overheating and stalling. Since the current is ~0.4A off from the required 1.68A of the motor specifications [47] a new motor shield is chosen, and the Adafruit motor shield is kept as a backup shield.

DFRobot Stepper Motor Shield for Arduino (DRV8825) SKU: DRI0023
The DFRobot Stepper shield is specially designed for stepper motors with the DRV8825 chipset, compatible with the Arduino UNO R3. [50]

The shield is specified to have 1.6A output current per coil [50] which matches with the 1.68A current requirements in the motor specification. [47] This solves the issue of the previous Adafruit motor shield.

3.1.3.3 Power source

The Arduino Uno R3 can be powered by the USB connection operating at 5V and an external power supply is supported, [44] however the external power supply is only applied when the motor shield is Adafruit motor shield. When using the DFRobot Stepper shield, an external power supply is not required for the Arduino Uno R3, but the external power supply is applied to the DFRobot Stepper shield.

DFRobot Stepper shield is designed to have 8/2 – 45V DC Input Voltage, used to power the stepper motor driver only. [50]

The power source used to power the DFRobot Stepper shield is a DC power supply rated at 22V DC, which fits within the specifications of the DFRobot Stepper shield.

3.2 Software

Full Software Interface Flow Diagram

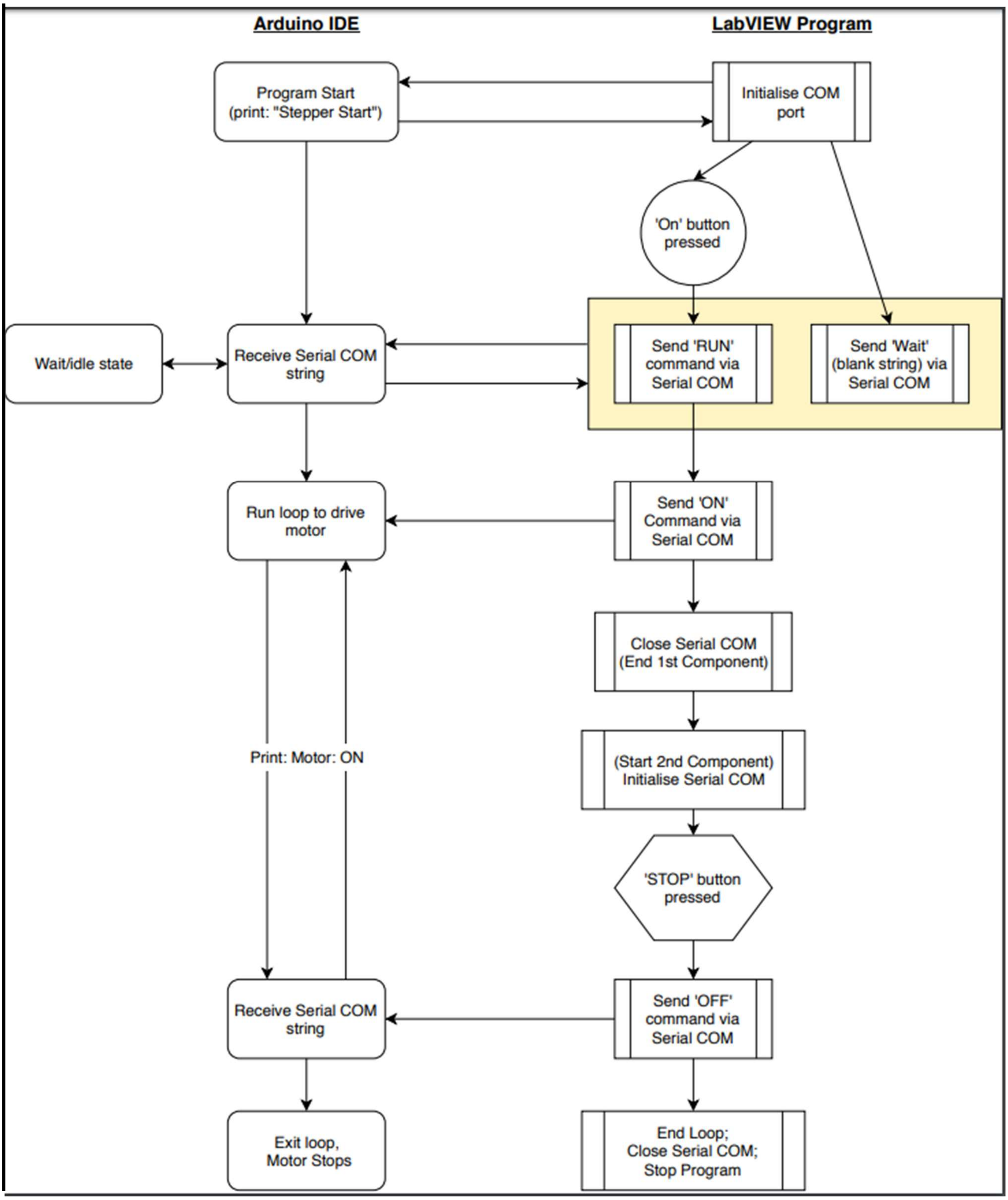


Figure 17. Arduino IDE and LabVIEW interface

3.2.1 Arduino Software (IDE)

To drive the motor of the extruder tool, the code is written and designed for Arduino Uno R3 in the Arduino Software (IDE). [44]

The motor drive code initially is designed with Adafruit motor shield in the design, but it is changed when the DFRobot Stepper shield was introduced.

The final code used to drive the extruder tool's motor throughout the experiments is designed for the DFRobot Stepper Motor Shield following the sample code given on the DFRobot Stepper Motor Shield product manual / Wiki page. [50]

The method to implement control of the motor within the code is through serial communication via USB. The code is designed so when the three different commands, 'ON' 'OFF' and 'RUN' are read through the serial bus, the motor is at different states.

The serial communication method allows the implementation of LabVIEW to be used with the Arduino IDE, the string commands are sent by LabVIEW instead, to control the motor states.

The LabVIEW coding will be explained further in the next section, section 3.2.2.

The following passage explains the steps the code takes for the motor control operation:

(Note: LabVIEW section will be explained in the same steps)

1. The beginning of the code will be indicated by receiving the message 'Stepper start', indicating the code has begun running.
2. Until the code receives the 'RUN' command, it considers all other serial string inputs as rubbish inputs, keeping the motor in a 'waiting' state. The moment the 'RUN' command is received, the string 'Motor: RUN' is sent from the code to the serial bus, preparing the motor to be turned on.
3. The next string command the code looks for is 'ON' command. Until 'ON' is received via serial, the motor remains in 'waiting' state. Once 'ON' is received, the motor goes through a 'do-while' loop containing code to run the motor.
4. Within the loop, the code continues to receive serial data, this is to look for the final command 'OFF' to stop the motor, Once the 'OFF' command is received, the 'do-while' loop breaks causing the motor to stop running and returns the string 'STOP' to the serial bus, indicating the motor has stopped.
5. The code is started from the beginning again when the reset button is pressed on the Microcontroller/DFRobot Stepper Shield

The full code used to drive the motor is provided in Appendix D.

3.2.2 LabVIEW

LabVIEW is a software made by National Instruments for easier visualisation for the design of distributed test, measurement and control systems. [51]

For the thesis experiment, LabVIEW is used to send commands to Arduino UNO R3 code via serial communications, allowing the computer to communicate with Arduino Uno R3.

The Front Panel of LabVIEW program contains buttons and visual panels to allow the user to turn on/off components and read various data.

The Block Diagram is the window that contains the LabVIEW coding, where three loops are implemented with serial communication blocks to perform the required control system design.

Both Front Panel and Block Diagram are provided in Appendix E.

The following steps are numbered corresponding to the steps the Arduino IDE code is implemented in. The reason is due to LabVIEW is required to run exclusively along with the Arduino IDE code for full control of driving the extruder tool motors.

1. The LabVIEW code begins with the initialisation of the serial communication (COM) port, the user selects the COM port the USB is connected to on the PC under 'VISA resource name'. The serial COM is setup using the 'VISA serial block' with the same baud rate, data bits, parity bit and stop bits as the COM port on the PC to allow communication between the serial COM. The code extends into the 1st while loop.
2. Within the while loop, there is the 'ON/OFF' button:
    - In the 'OFF' condition (Red), LabVIEW sends 'Wait' string down the serial bus, on the receiving end at the Arduino UNO R3 board, the code will simply ignore the string as it is not a command. This keeps the motor in the 'waiting' state.
    - In the 'ON' condition (Green), LabVIEW sends 'RUN' string down the serial bus, on the receiving end at the Arduino UNO R3 board, the code will recognise 'RUN' command and prepare motor for running and returns the string 'Motor: RUN'.
    - LabVIEW has the condition of stopping the 1st while loop if 'Motor: RUN' string is received. This means once the button is in ON state, the loop terminates and the 2nd while loop commences.
3. In the 2nd while loop, LabVIEW sends 'ON' string, on the receiving end at the Arduino UNO R3, the command 'ON' is received and the motor begins running. The loop will

continuously send ‘ON’ command to the Arduino UNO R3 keeping the motor in the running state until the ‘STOP’ button is pressed on the LabVIEW Front Panel. If the ‘STOP’ button is pressed, the loop terminates, and the buffer is refreshed before continuing onto the final while loop.

4. The final while loop contains 1st the closing of the serial COM and a reinitialization of the serial COM, this allows all string buffers on the serial bus to be clean. LabVIEW then sends the ‘OFF’ string, on the receiving end at the Arduino, the motor is stopped and returns ‘STOP’ message. LabVIEW does not necessarily need to stop only when ‘STOP’ is received but rather anything received indicates the motor is stopped. So simply comparing against any letter will allow the loop to terminate.
5. LabVIEW itself resets the program once the final loop is terminated, so the LabVIEW program can start up again once the Microcontroller/DFRobot Stepper Shield is reset.

3.2.3 RAPID (RobotStudio programming)

RAPID is the programming software within the RobotStudio software to program the industrial robot. RAPID is commonly used for the motion programming aspect, where move instructions are written to move the robot to a specific point. The RAPID code can also be edited directly within the FlexPendant of the IRC5. [52]

The motion programming instructions are the main instructions used in this thesis experiment, the code is written with the following movement instructions as well as some additional miscellaneous functions: [53]

- MoveJ – Robot moves to point using joint movement
- MoveL – Robot moves to point linearly
- MoveC – Robot moves in a circular motion
- Stop – Stopping the program execution, the next program instruction is suspended until 'play' is pressed on the FlexPendant.

In RobotStudio, the points used by the motion programming instructions are called 'Targets' and are created by entering the x, y and z coordinates of a point in relation to the world object coordinates. [52] The motion instructions move to these points to create paths that the robot IRB 120 will follow.

The layout of the RAPID code follows the following design:

1. A home position is set by moving to a point using MoveJ at v250.
2. The robot moves to starting position with MoveJ at v60. (starting position varies depending on the test)
3. 'Stop' instruction added.
4. A range of movement instructions added to complete a movement path, so the robot moves corresponding to the desired shape. The speed at which the instructions are written with changes per each pathing design.
5. 'Stop' instruction added.
6. Move to appropriate end position for the movement path designed in 4, using MoveJ at v60.
7. Return to home position with MoveJ at v250.

All RAPID programming code are provided in Appendix F with their corresponding path patterns.

3.3 Material

Material designed to be used in this thesis project must come under the Food category, materials that can be easily moulded into a 3D shape often relate to baked goods / deserts.

3.3.1 Chocolate

Chocolate is the most common food material that can be moulded, while heated, chocolate remains in a liquid state and once cooled, chocolate solidifies and takes up a form.

3.3.2 Cream Frosting

Cream or various cream frostings are commonly found on cakes as decoration, the common ingredients in a cream frosting contains Butter, Milk and Icing sugar, the consistency of the cream frosting remains in a fluid state at room temperature while slightly hardening in cooler temperature due to the butter within the ingredients.

Chapter 4: Experimental Method

4.1 Calibration Tests

4.1.1 Calibrating Food Extruder Tool

The Food Extruder Tool is connected to the end of the IRB 120 robot. Following the 'creating a tool' instructions in the 'Operating manual' for IRC5 with FlexPendant, the tool is calibrated with 4 points as reference and the new tool named 'extruder' is calibrated for its TCP. [46]

Within the tool data, the weight of the tool (0.5 kg) is added and an estimate for the tool's centre of gravity (x,y,z) data are measured and added before the tool can be used. [54]

4.1.2 Calibrating Optimal Speed Relationship

Between Robot and Extruder's motor speed (aka. Rate of extrusion), the speed is set after running a series of tests to determine the best possible speed to program the movement instructions within RAPID for the robot pathing, corresponding with the rate of extrusion.

The rate of extrusion is determined by the DFRobot Stepper Motor Shield, the shield has dip switches designed to implement 6 different Microstep Resolution: [50]

Table 3. Dip switch settings for Microstep resolution | DFRobot [66]

MS1	MS2	MS3	Microstep Resolution
Low	Low	Low	Full step
High	Low	Low	Half step
Low	High	Low	1/4 step
High	High	Low	1/8 step
Low	Low	High	1/16 step
High	Low	High	1/32 step
Low	High	High	1/32 step
High	High	High	1/32 step

The calibration tests will involve testing of RAPID motion instructions with changing speeds (mm/s) against the Microstep Resolution.

IRB 120 model is only capable of a maximum speed of v250 (250 mm/s) in manual mode. [46]

5 RAPID speed settings for IRB 120 will be tested for each Microstep Resolution.

To measure each speed setting, a simple robot path is designed to extrude a line 150 mm in length 5 times for each RAPID speed setting.

Each line is measured for its height, starting, middle and end thickness. An average thickness measurement is calculated using starting, middle and end thickness, as height can be changed to suit different purposes by lowing the extruder tool's tip in the z plane.

4.1.2.1 Calculation

4.1.2.1.1 Extruder Motor Maximum Speed

To find the speed equivalent of the each Microstep resolution of the DFRobot motor shield, we need to calculate the maximum speed the stepper motor can run at. This will also be the speed at 'full step' Microstep resolution.

Equation 1. Max Speed of Stepper Motor [55]

$$v_{max} = \frac{V}{2LI_{max} \cdot spr}$$

Equation 2. Steps Per Revolution

$$spr = \frac{360°}{Step\ Angle}$$

v_{max} = Maximum Speed (revolutions per sec.)

V = Voltage (V)

I_{max} = Maximum Current (A)

L = Inductance (H)

spr = steps per revolution

Using Equations 1 and 2 and NEMA17 stepper motor specification data from Table 2, section 3.1.2.1, we can calculate the extrusion rate of the motor at different Microstep resolutions.

$$spr = \frac{360°}{1.8°} = 200; \quad v_{max} = \frac{2.8V}{2(2.8\times10^{-3}H)(1.68A)(200)} = 1.49\ rev/s$$

$$rpm\ (revolution\ per\ minute) = \frac{1.49rev}{s} \times \frac{60s}{1min} = 89.30\ rpm$$

Using the same calculations, the results for all Microstep resolutions are:

Table 4. Maximum Speed of Microstep Resolutions

Microstep Resolution	Steps per revolution	Maximum Speed (rpm)
Full Step	200	89.30
Half Step (1/2)	400	44.64
1/4	800	22.32
1/8	1600	11.16
1/16	3200	5.58
1/32	6400	2.79

4.1.2.1.2 Flow Rate Calculation

Flow Rate Equation:

Equation 3. Volume Flow Rate Equation [56]

$$\dot{V} = V_{avg} A_c \ (m^3/s)$$

Know Data:

$Lead\ travel/revolution\ = 2\ mm$ (Table 2)

Syringe barrel measurement:

$D\ (diameter) \cong 34.57\ mm$ (measured)

$A_c = \pi r^2 = \pi(\frac{34.57}{2})^2 \cong 938.62\ mm^2$

Syringe tip / Food Extruder Tool Tip measurement:

$d\ (diameter) = 4\ mm$ (Table 1)

$A_c = \pi r^2 = \pi(\frac{4}{2})^2 \cong 12.57\ mm^2$

Velocity calculation for Microstep Resolution (Example using Full Step):

$$V_{avg} = \frac{2\ mm}{rev} \cdot RPM = \frac{2\ mm}{rev} \cdot \frac{89.30\ rev}{min} \cdot \frac{min}{60s} \cong 2.98\ mm/s\ @\ Full\ Step$$

Volume Flow Rate (Example at Full Step):

$\dot{V} = 2.98\ mm/s\ \cdot 938.62\ mm^2\ \cong 2793.96\ mm^3/s$

For incompressible flow due to the material being Cream Frosting, we can use the conservation of mass relations for steady, incompressible flow. [56]

Equation 4. Steady, incompressible flow; (single stream) [56]

$$\sum_{in} \dot{V} = \sum_{out} \dot{V}\ \ (m^3/s)$$

$$\therefore V_{in}A_{in} = V_{out}A_{out}$$

The V_{out} in this case is the average speed the material leaves the tool tip of the extruder, though the 4mm diameter.

To make sure the material keeps its dimensions after leaving the extruder tool tip, the speed estimated at which the robot moves must be equal to this V_{out};

$\therefore\ V_{out} = Estimated\ RAPID\ motion\ instruction\ speed.$

Table 5. Calculation Results for all Microstep Resolutions

Microstep Resolution	Volume Flow Rate (mm³/s)	V_{out} (mm/s)
Full Step	2793.96	222.27
Half Step (1/2)	1396.67	111.11
1/4	698.33	55.56
1/8	349.17	27.78
1/16	174.58	13.89
1/32	87.29	6.94

4.1.3 Testing Optimal Material

While conducting the speed test to determine the Optimal speed at which the robot and the extruder tool will be running at, two different materials are added to see which material is best suited with the extruder tool. A range of parameters to be looked at are:

- Relevant material properties for extrusion.
    - Feasibility for extrusion
    - Material state during extrusion process
    - Material state after extrusion process
- Preparation of material for extrusion.
- Preservation of material for extrusion.
- Material consistency.
- Material costs.

4.2 2D Printing

2D printing experiment is essentially experimentation on a single layer from that of a 3D model.

4.2.1 Rectangle Layer

RAPID programming will contain a set of motion instructions to move the robot in the path to fill in a 2D rectangle following the path designed in CAD (Creo Parametric).

In the CAD model shown below, the shaded and risen edge shows the height of the layer as well as the area to be filled.

The path IRB 120 robot needs to follow is highlighted in red.

Rectangle dimensions: 100 mm x 50 mm

Using the CAD dimensions, points are worked out for Motion Instructions in RAPID program code.

The points are entered into the RAPID code as ‘targets’ the RAPID code is then synchronised with RobotStudio’s virtual station to generate a virtual path.

The virtual display is used as a guidance to ensure the 2D layer designed in CAD diagrams are produced.

A range of tests with different methods to fill in the rectangle are implemented leading up to tests producing final required rectangle layers:

1. A single rectangle layer is filled following horizontal deposition path of the rectangle CAD design.
2. A single rectangle layer is filled following the vertical deposition path of the rectangle CAD design.

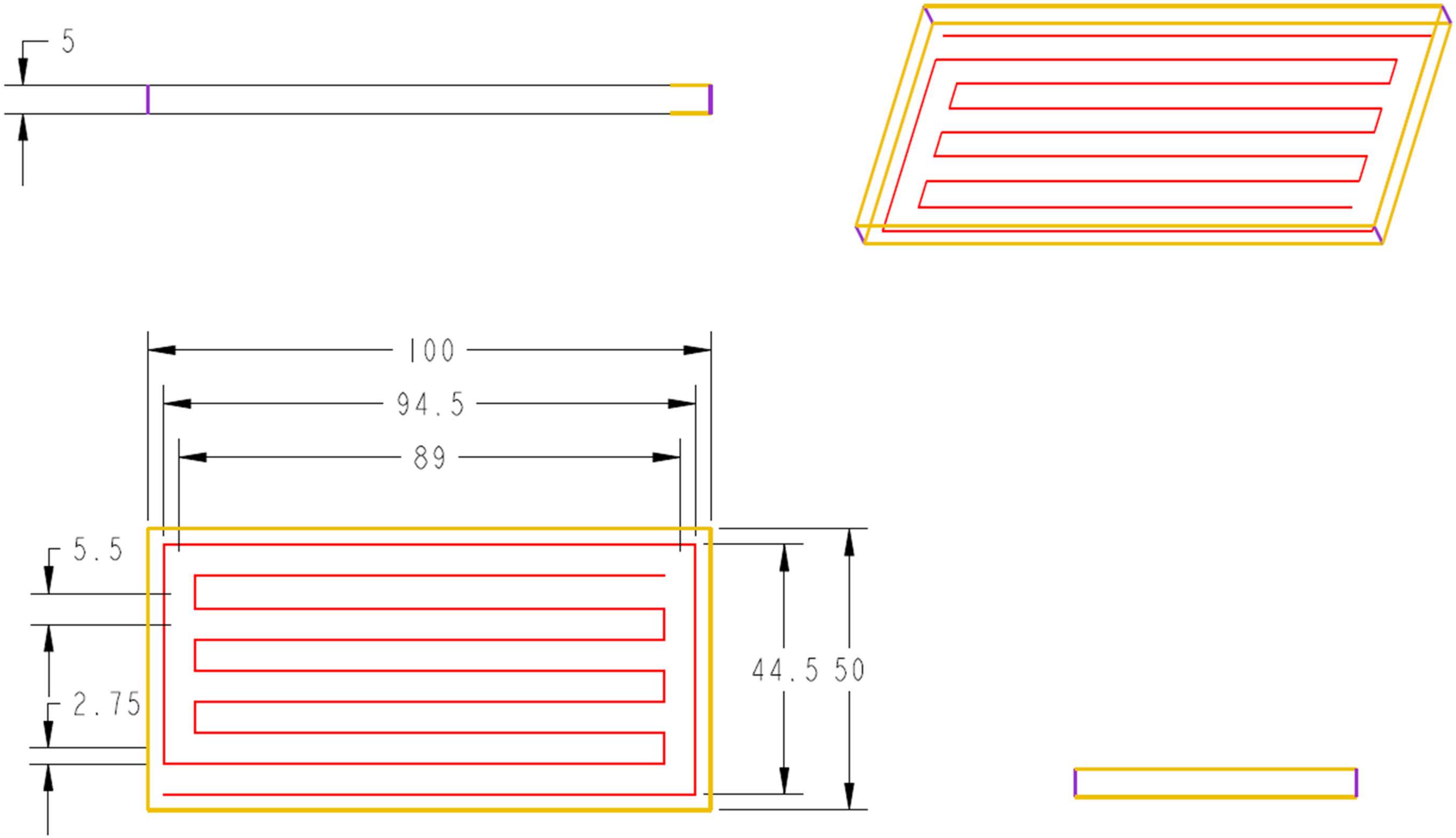


Figure 18. Rectangle 2D Layer Horizontal Fill CAD drawing.

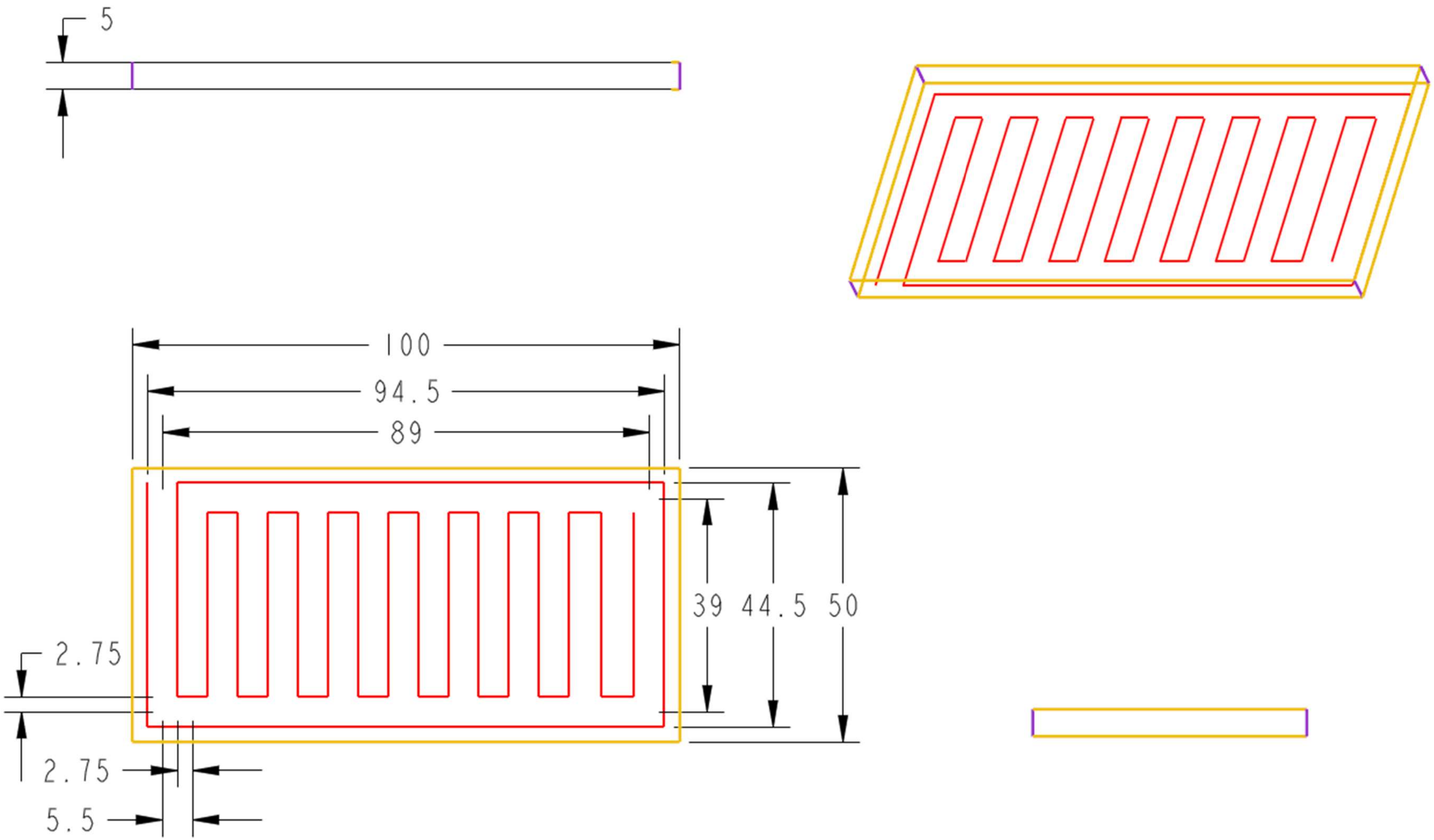


Figure 19. Rectangle 2D Layer Vertical Fill CAD Drawing.

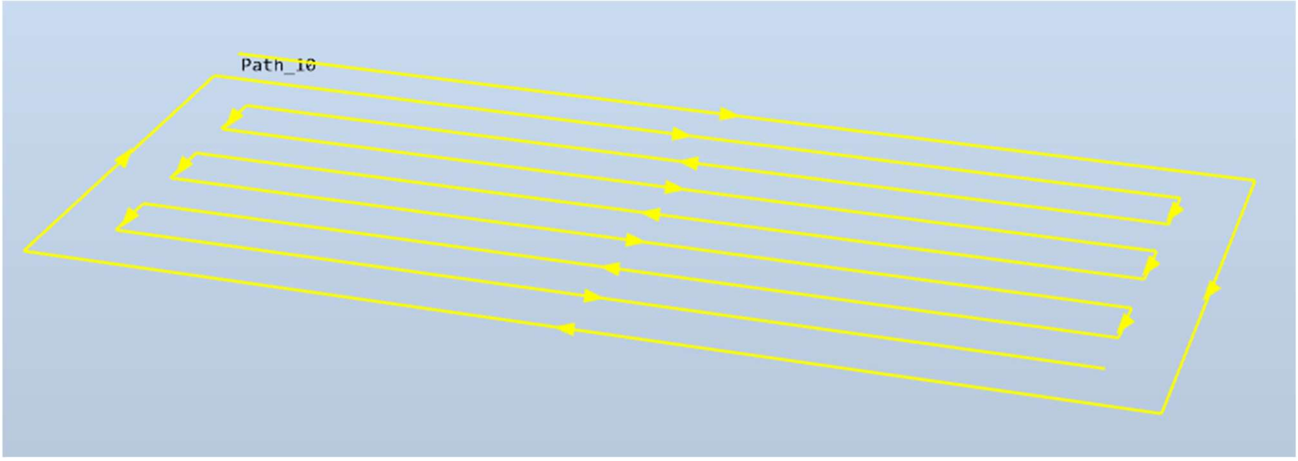


Figure 20. RobotStudio Virtual Path: Rectangle Layer Horizontal Fill

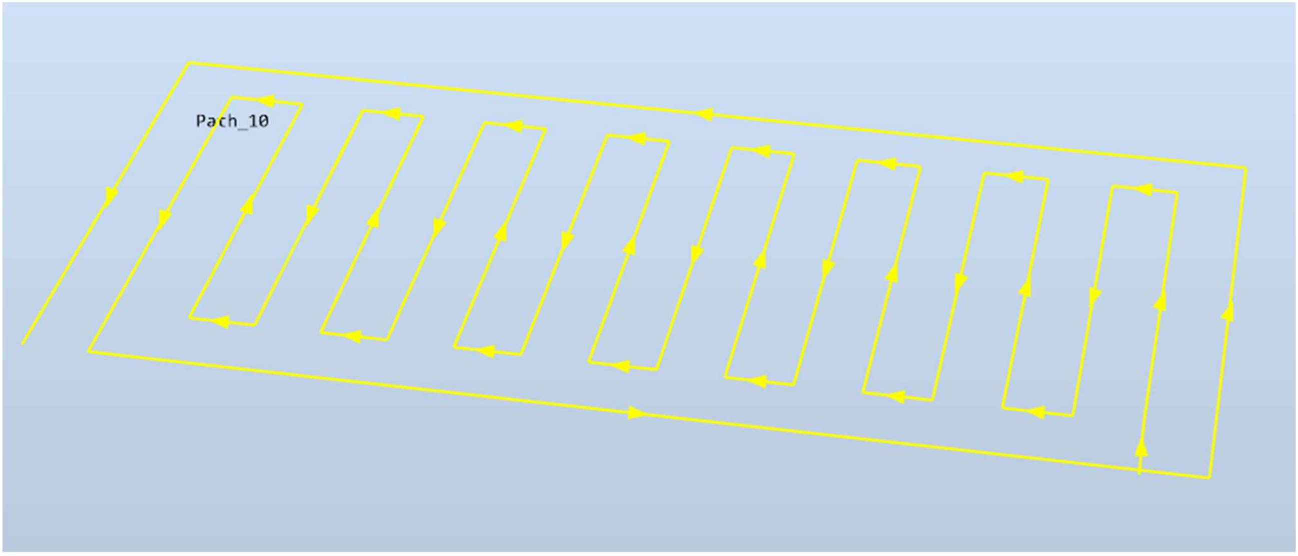


Figure 21. RobotStudio Virtual Path: Rectangle Layer Vertical Fill

4.3.1 Rectangular Block/Prism

Incorporating the rectangle fill methods from section 4.2.1, of the 2D printing tests, the rectangle block is created in four layers.

Layer 1: Horizontal deposition method

Layer 2: Vertical deposition method

Layer 3: Horizontal deposition method

Layer 4: Vertical deposition method

This style of layering results in a crisscross pattern to ensure the block is properly filled.

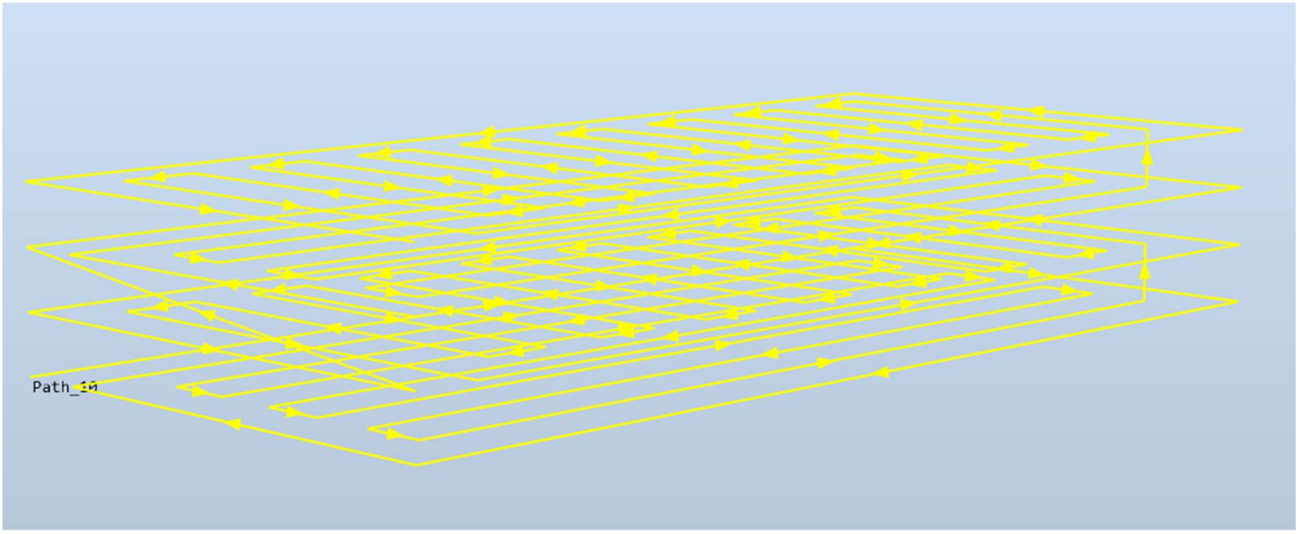


Figure 22. Rectangle Block composed of four layers

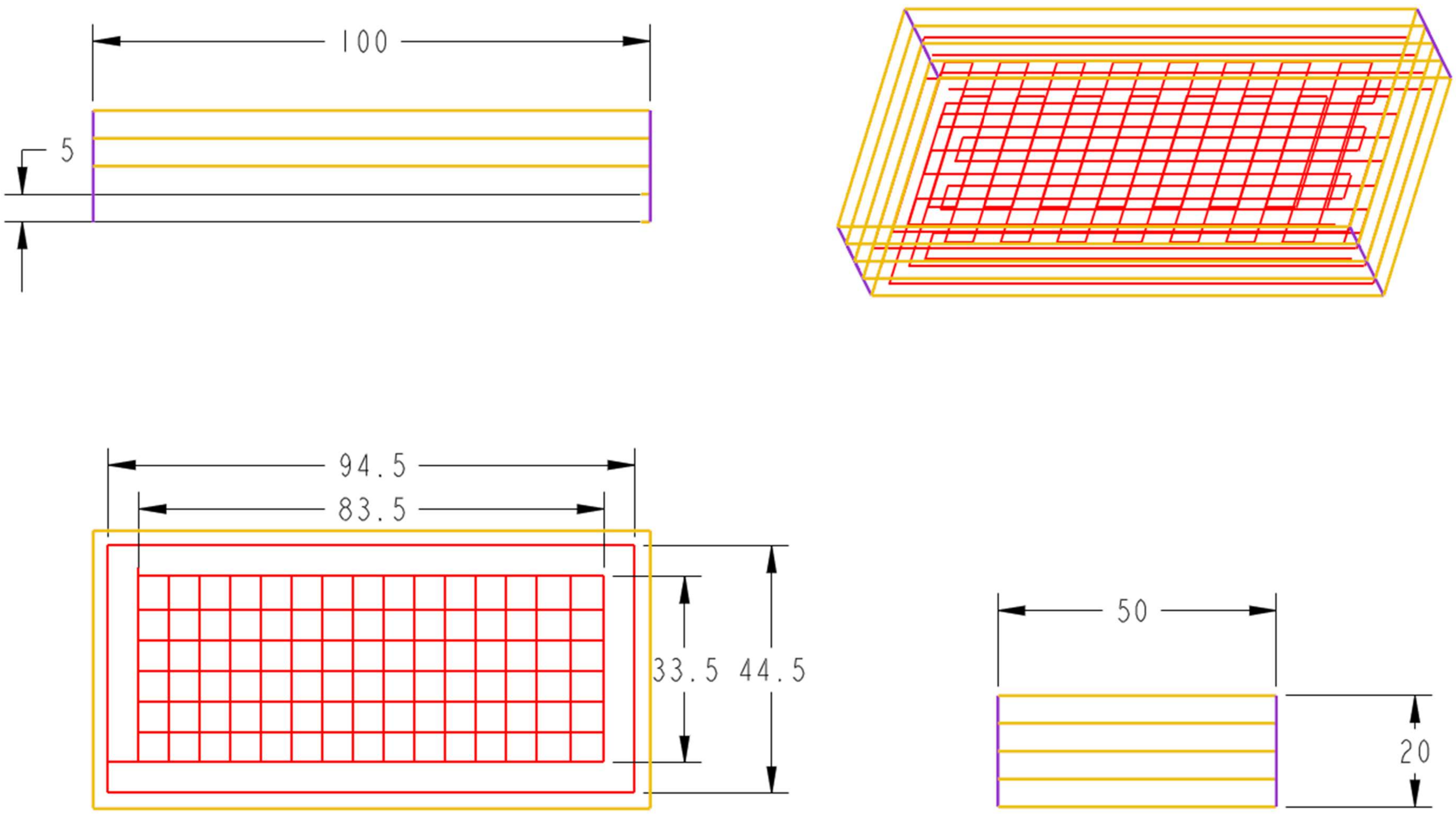


Figure 23. 3D Rectangle Prism CAD Drawing

Chapter 5: Experimental Results

5.1 Calibration results

5.1.1 Food Extruder Tool Calibration

The extruder tool worked properly throughout the experimental process.

A problem that has occurred but does not directly apply to / is not covered within the extent of the scope of this thesis research is the issue with communication from the IRB 120 robot and LabView.

Currently the motor of the extruder tool is operated with LabView separately to the motion movement of IRB 120. This means when initiating the movement instruction of the IRB 120 robot, the motor must be started up and stopped at the same time manually, relative to the beginning and end of the movement path the IRB 120 robot follows.

5.1.2 Material Test

The two materials this thesis experiment is designed for are chocolate and cream frosting.

Figure 24. Chocolate Extrusion Sample

Chocolate is the material first tested, although the chocolate extruded as expected, the process to preserve the chocolate for extrusion is too meticulous.

To preserve the chocolate at an appropriate extrusion condition, the chocolate needed to be kept in a specific temperature range, so the chocolate remains in the melted liquid form.

Due to the speed at which the IRB 120 robot reset to the beginning of its motion paths and the initialisation of the beginning of the next motion, starting and stopping the motor, all these steps cause time to which the chocolate starts to thicken/harden in room temperature.

A major problem with chocolate has being observed and during the experiment and accident has occurred with the extruder tool, the details will be discussed in the chapter 6, however a decision was made to continue the remaining experiments with the cream frosting material only, removing chocolate as a material for testing in the remaining experiments.

The Cream Frosting material contains the ingredients of Butter, Icing Sugar Mixture and Milk, the material remains in extrudable state in room temperature and takes on the properties of butter, solidifying in cooler temperatures and melting in higher temperatures.

The material required longer time in preparation compared to chocolate, since the chocolate used is pre-bought and simply melted when heat is applied. (aka. In a microwave oven) Compared to the Cream Frosting, where the material is made from the combination of the ingredients through a ‘cooking process’.

The first time the material was extruded was done as a quick experiment at RAPID motion instruction speed of v10 and ¼ Microstep stetting on the motor of the extruder tool.

5.1.3 Speed Calibration

Before the speed calibration tests, the motor speed of the extruder tool was initially set at ¼ Microstep and the RAPID code had motion instructions set at 10v which is the 2nd lowest setting, this showed a fast extrusion but a slow path movement resulting in an extremely poor result.

Although the actual speed calibration tests have not begun at this point, this first test showed ¼ Microstep to be too fast of a setting and the RAPID motion instruction must be set at a faster rate.

A quick test was conducted at v50, v60, v80 and v100.

The results show as the speed of the RAPID motion instruction is increased, the extrusion becomes closer to a proper material deposition however to get the width and height to an appropriate dimension, the speed needs to exceed v80.

At v100 the speed becomes too fast to properly start and stop the motor of the extruder tool, causing a failed result.

In the end ¼ Microstep is too fast to produce proper extrusion results.

The decision was made to continue testing with 1/8, 1/16 and 1/32 Microstep settings on the extruder tool and 5 varying speed settings for the robot in its RAPID motion instructions.

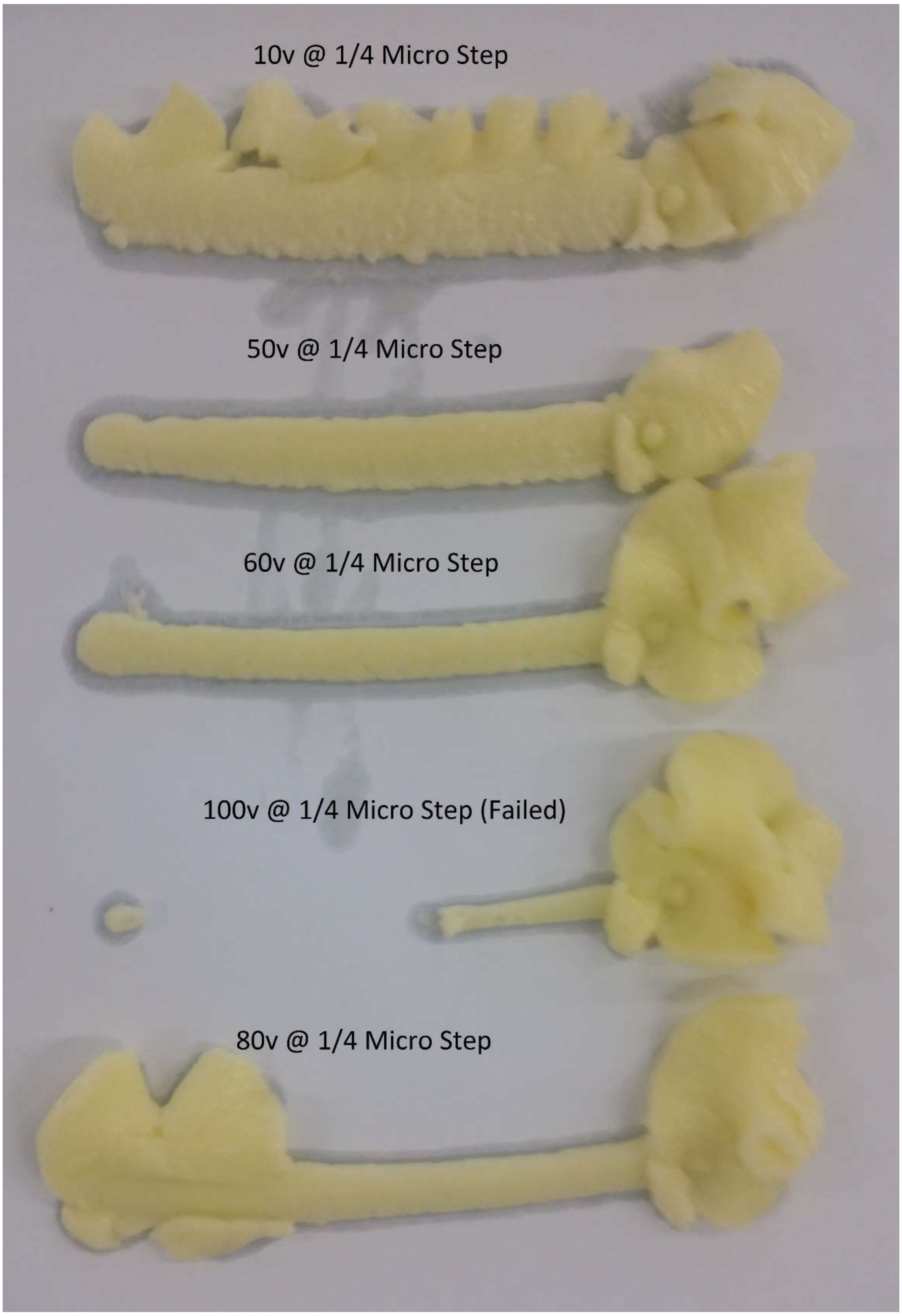


Figure 25. 1/4 Microstep Speed Calibration Tests

5.1.3.1 1/8 Microstep setting

The 1/8 Microstep setting is now the fastest setting the extruder tool is set at for the experiment phase of this thesis project.

Judging from the results from the ¼ Microstep speed setting, the 5 RAPID motion instruction speeds are chosen giving the following results.

Table 6. 1/8 Microstep Setting Thickness Measurement Results

1/8 Microstep Setting					
	Height (mm)	**Start (mm)**	**Middle (mm)**	**End (mm)**	***Average (mm)**
v60	4.28	6.19	5.49	5.52	5.73
v80	3.86	5.18	4.89	4.57	4.88
v100	3.45	4.59	4.42	4.70	4.57
v150	3.04	4.45	4.06	3.82	4.11
v200	2.84	4.10	3.96	3.98	4.01

*Average: Taken from Start, Middle and End Thickness measurements.

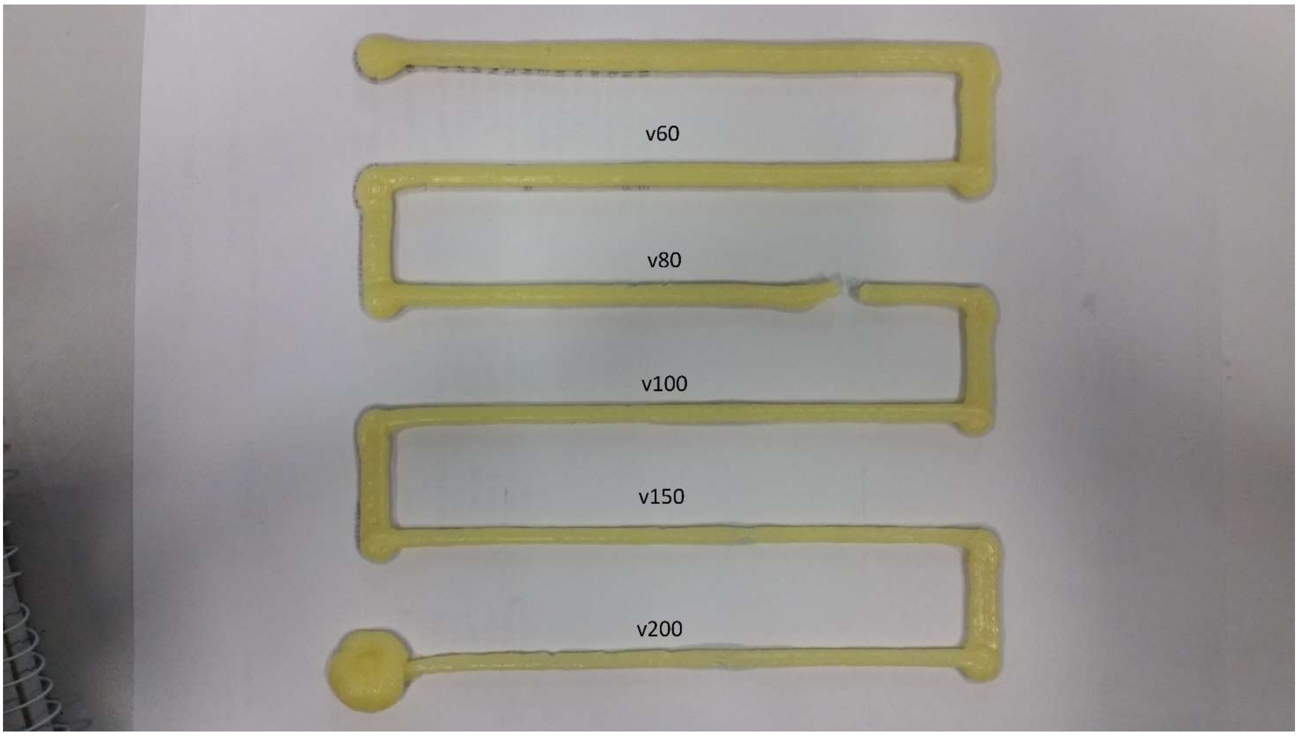


Figure 26. 1/8 Microstep Test

The results show that all speeds set in RAPID motion instructions produced results that kept consistency and form beside v80, where a single break is seen, however this break was caused by an air bubble trapped within a section of material during the extrusion process.

The break did not occur due to the RAPID speed setting being too fast.

In the case where the speed of the RAPID motion instruction becoming too fast is at v150 and v200. As we zoom in the results at the two highest RAPID speeds, in Figure 27 the circled locations show deformation in the extrusion results begun occurring.

The deforming sections can be seen to be under tensile stress with the surface showing cavities, a clear sign of the extruded result being under stress.

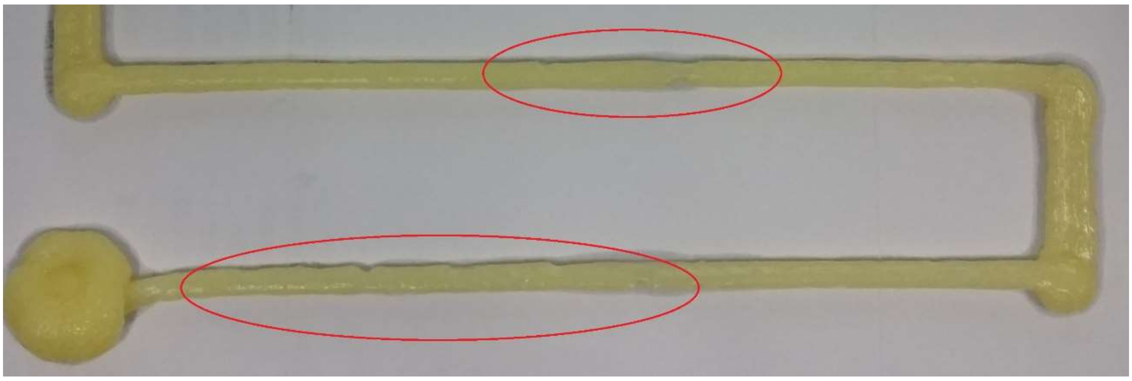


Figure 27. deformation at speeds v150 and v200

The measurements for 1/8 Microstep show the dimensions are at a decrease in size when comparing the averages and height for each RAPID motion instruction speeds.

Dimension differences at v150 and v200, the two speeds under stress, are much smaller compared to differences of the prior 3 RAPID speeds.

Although the speed change from v150 to v200 is a 50 mm/s increase while v60 to v80 to v100 is a 20 mm/s speed increase every time, comparatively much slower.

The best speed to set RAPID Motion Instructions to are v60, v80 and v100, for 1/8 Microstep Resolution. The v60 having a more well-rounded measurement and v100 is an option with thinner dimensions.

In this specific experiment, the v80 unfortunately had that air cavity build up at the one location but the continuous section shows consistency as good as the other two RAPID speed selections, making it also an appropriate selection.

5.1.3.2 1/16 Microstep setting

Table 7. 1/16 Microstep Setting Thickness Measurement Result

1/16 Microstep Setting					
	Height (mm)	**Start (mm)**	**Middle (mm)**	**End (mm)**	***Average (mm)**
v30	4.34	5.17	5.40	4.98	5.18
v40	4.11	5.33	5.02	4.95	5.10
v50	3.61	4.33	4.72	4.88	4.64
v60	3.20	4.74	4.43	4.71	4.63
v80	3.05	3.98	4.18	4.29	4.15

*Average: Taken from Start, Middle and End Thickness measurements.

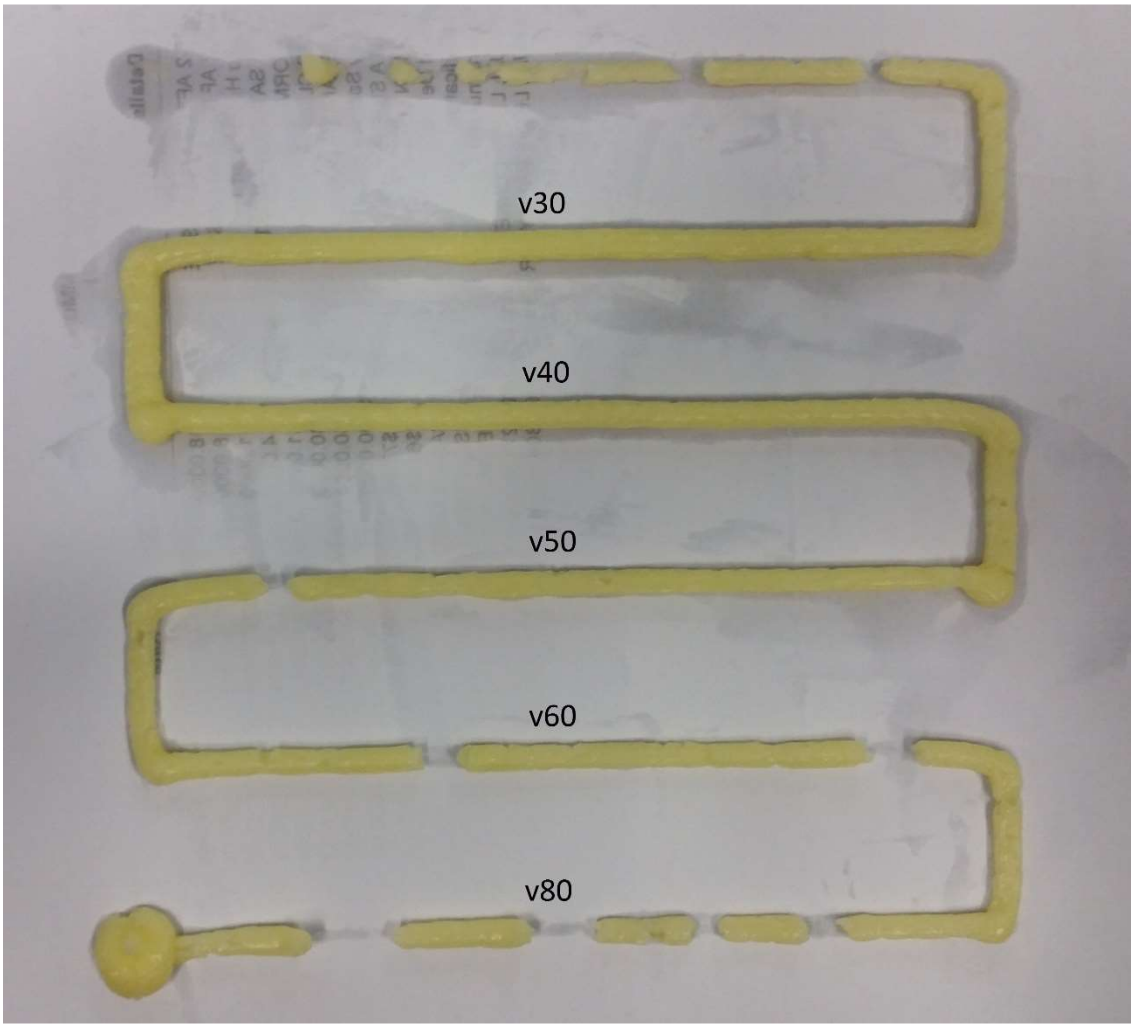


Figure 28. 1/16 Microstep Setting

The results for 1/16 Microstep setting show v50, v60 and v80 RAPID Motion Instructions' speed selections having breakage in the extrusion results. Zooming in on the results in Figure 29 we see the results in more detail, the number of breakage points clearly increase as the speed increases.

In the extrusion result of v50, we see a single fully broken strand circled, however the result is already under tensile stress at this RAPID speed.

Highlighted within the red rectangle, we see as the extrusion is thinning, cavities and concaves begin appearing, these sections are proof of the extruded material under stress and soon after, highlighted in the red circle, a breakage occurs.

For v60 and v80, the breakage locations are prominent and numerous, these two RAPID speed selections are not usable with the 1/16 Microstep resolution.

The size of the broken section increases directly relative to the RAPID speed, in this test, the number of breakage also increase as RAPID speeds have increased.

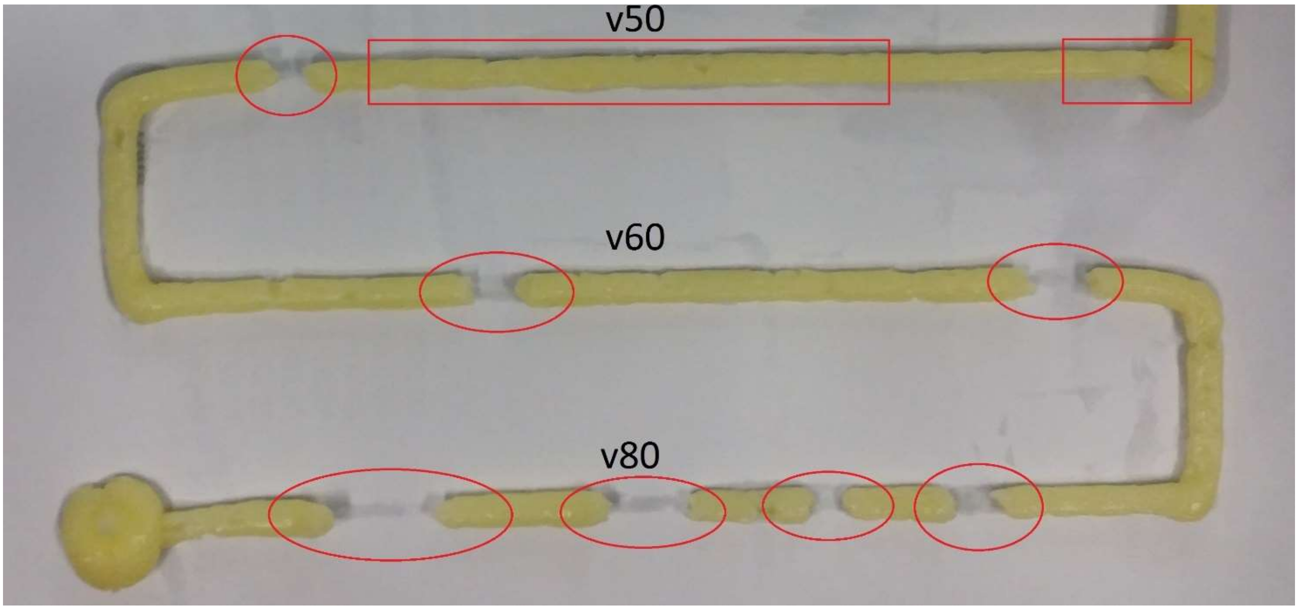


Figure 29. Deformation occurring at speeds v50, v60 and v80

A noticeable observation can be made from comparing the Average measurement results, for 1/16 Microstep setting, the thickness dimensions are very similar between speeds with a difference of 10 mm/s (± v10) only, e.g. v30/v40 and v50/v60, while a large thickness discrepancy is seen at every speed increase of 20 mm/s. e.g. v30/v50, v40/v60 and v60/v80.

Unfortunately, RAPID Motion Instructions cannot be set to v70, therefore no additional data can be added and used to support this observation.

The results ended with only two appropriate RAPID Motion Instruction speed selections for 1/16 Microstep resolution setting, the v30 and v40. Both speeds' measurement results are similar in dimension size, so choosing either speeds for RAPID Motion Instruction will not create a large result difference.

5.1.3.3 1/32 Microstep Setting

Table 8. 1/32 Microstep Setting Thickness Measurement Result

1/32 Micro Step Setting					
	Height (mm)	**Start (mm)**	**Middle (mm)**	**End (mm)**	***Average (mm)**
v10	5.10	8.18	7.38	7.28	7.61
v20	3.91	5.43	5.11	4.83	5.12
v30	3.39	4.24	4.10	4.43	4.26
v40	2.84	4.31	4.27	4.01	4.20
v50	1.82	3.53	3.66	4.30	3.83

*Average: Taken from Start, Middle and End Thickness measurements.

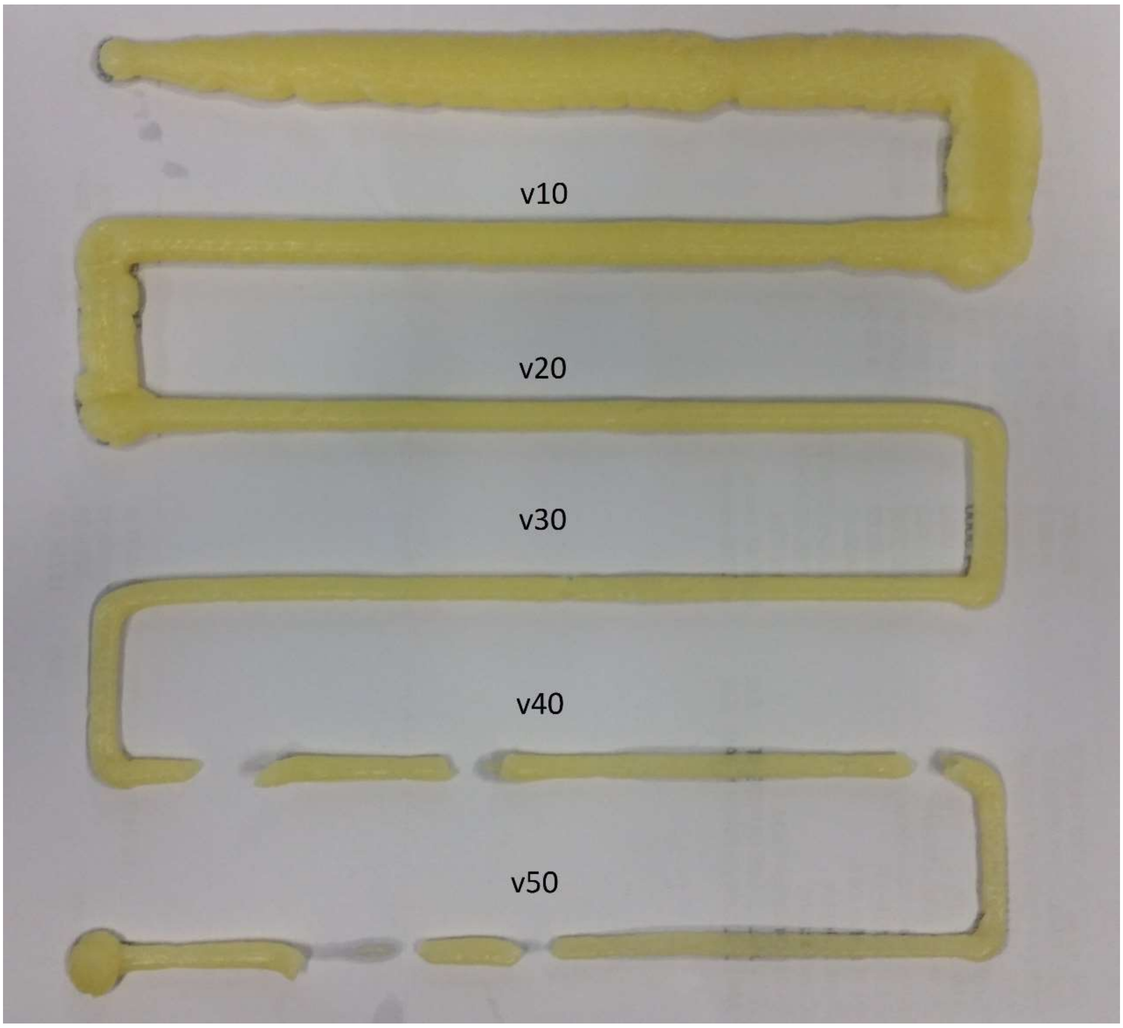


Figure 30. 1/32 Microstep Setting

The results of 1/32 Microstep resolution are similar to 1/16 Microstep resolution results, where the two highest speed settings of the test shows breakage, and the first two slower speeds show more consistency.

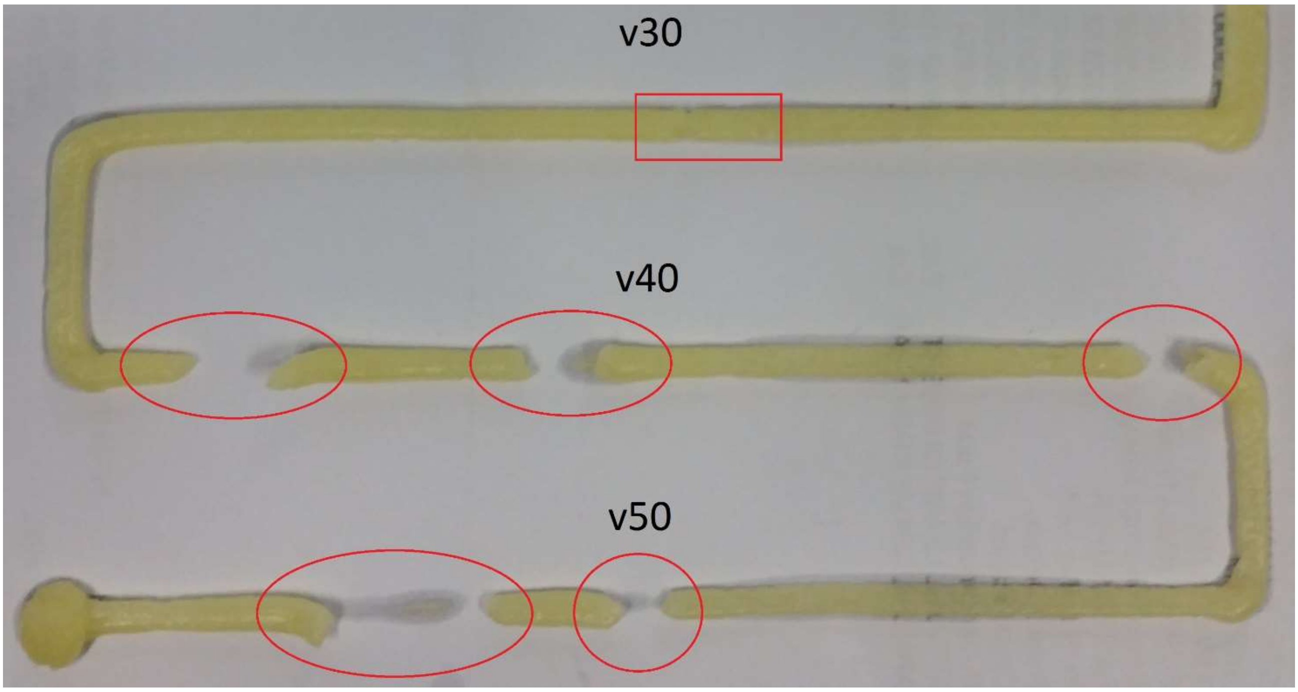


Figure 31. Breakage at speeds v60 and v80

Zooming in and looking at RAPID speed choices that show deformation in Figure 31, v40 and v50 show prominent broken sections, while v30 does not show any obvious deformations.

In v30, the single section highlighted in the red rectangle is a small deformation location, the tensile stress is visible at this location, just as with the 1/16 Microstep result, this section has the thinning characteristics along with cavity and concaves appearing.

The results could be viewed as a speed before tensile stress reaches the point of breakage exists after v30 RAPID speed selection, but the v30 is the final speed the RAPID motion instruction can be set at without breaking the extruded material, as the next speed is v40.

In v40, the breakage is shorter in length between the break point and the new starting point compared to v50 but unlike 1/16 Microstep results, the number of broken sections does not increase as speed increased.

It is noticeable that in the average thickness measurement results for 1/32 Microstep, the dimension differences between v30 and v40 speeds almost don't exist, exactly same as 1/16 Microstep results.

The height however keeps consistency in its decrease at every RAPID speed selection.

The speeds v10, v20 and v30 have complete material extrusions with no signs of breakage, the v30 RAPID speed choice does have slight deformation probability so it could be usable as long as the deformation does not occur too often, same as this test.

RAPID speed v10 is the largest dimension, although the extruded result is consistent, the dimensions are too large compare to the extrusion tool's tool tip cross section area, so v40 visually shows the best results.

5.2 2D Printing results

From the results of section 5.1 Speed Calibration, the RAPID Motion Instruction speed is selected to be v40 and the DFRobot Stepper Motor Shield is set to 1/16 Microstep resolution for the 2D print tests.

Taken from 'Table 5. 1/16 Microstep Setting Thickness Measurement Result', the thickness of the extrusion is on average: 5.1 mm.

The CAD drawings in Chapter 4, section 4.2 2D Printing are designed for the extrusion to have a maximum width of 5.5mm.

5.2.1 Rectangle Horizontal Deposition Layer

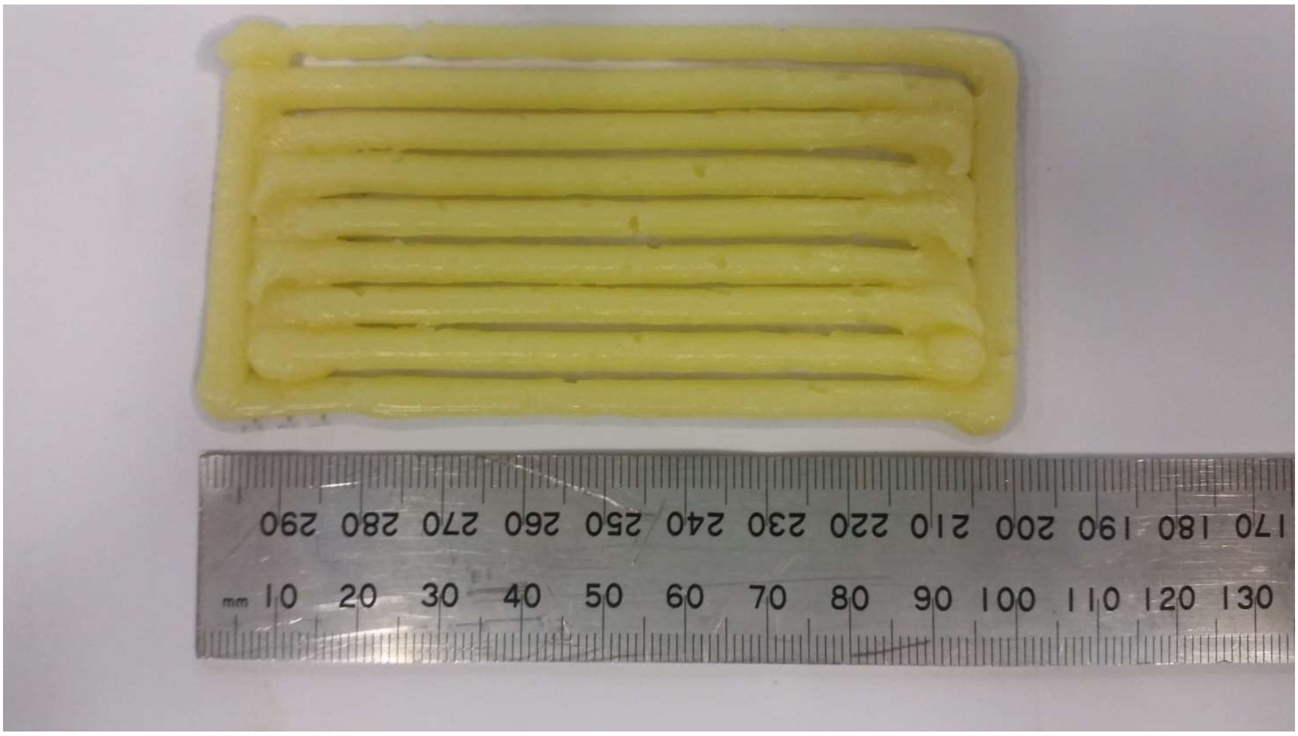


Figure 32. Rectangle Horizontal Deposition Layer Length measurement

The length of the rectangle is measured to be 100 mm ±2 mm, matching the initial design dimensions in the CAD drawings.

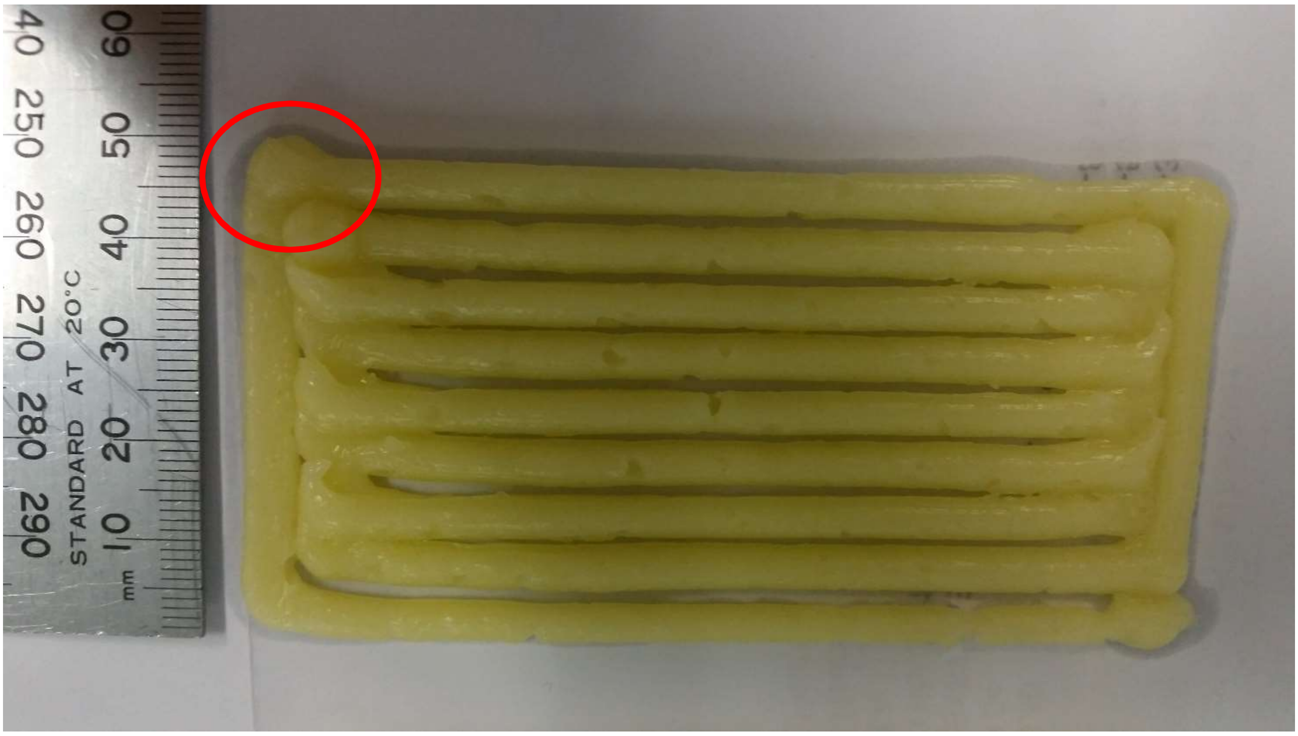


Figure 33. Rectangle Horizontal Deposition Layer Width measurement

The measurement is approximately 50 mm including the protruded section highlighted in the red circle or approximately 47 mm not including the protrusion. Both measurements satisfy the initial designed dimensions in the CAD drawing.

5.2.2 Rectangle Vertical Deposition Layer

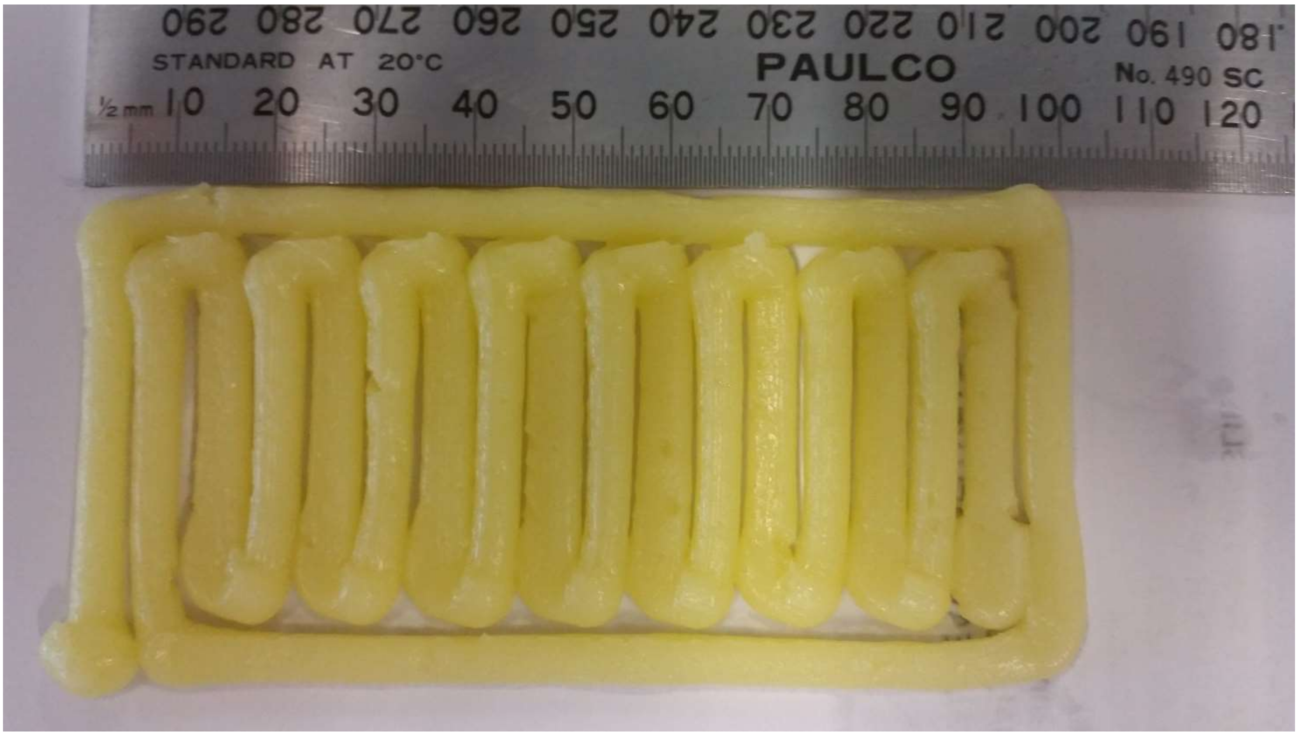


Figure 34. Rectangle Vertical Deposition Layer Length measurement

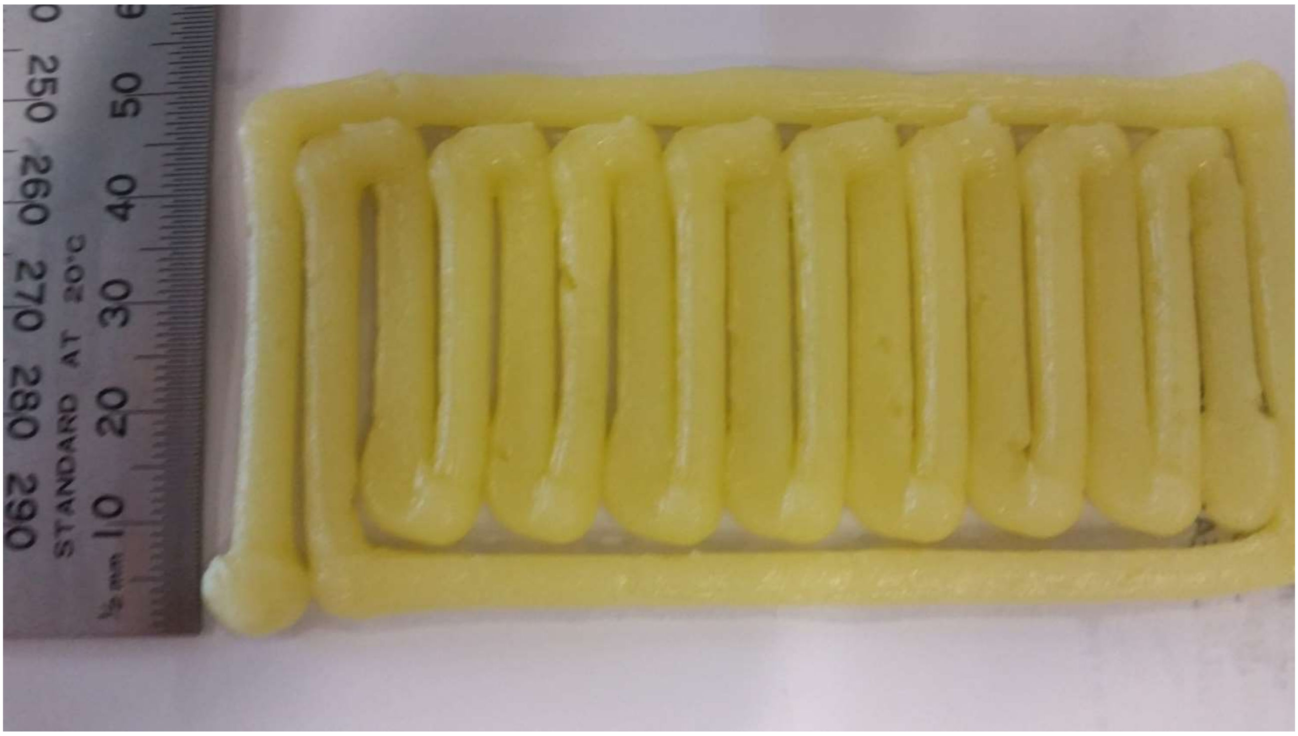


Figure 35. Rectangle Vertical Deposition Layer Width measurement

The dimensions of the layer are measured to be 100 mm x 50 mm, matching initial CAD designs.

5.3 3D Printing results

Using the exact same setting as 2D layer printing tests, the 3D print results are produced using RAPID Motion Instruction speed set at v40 and DFRobot Stepper Motor Shield set to 1/16 Microstep resolution.

5.3.1 Rectangle Block/Prism

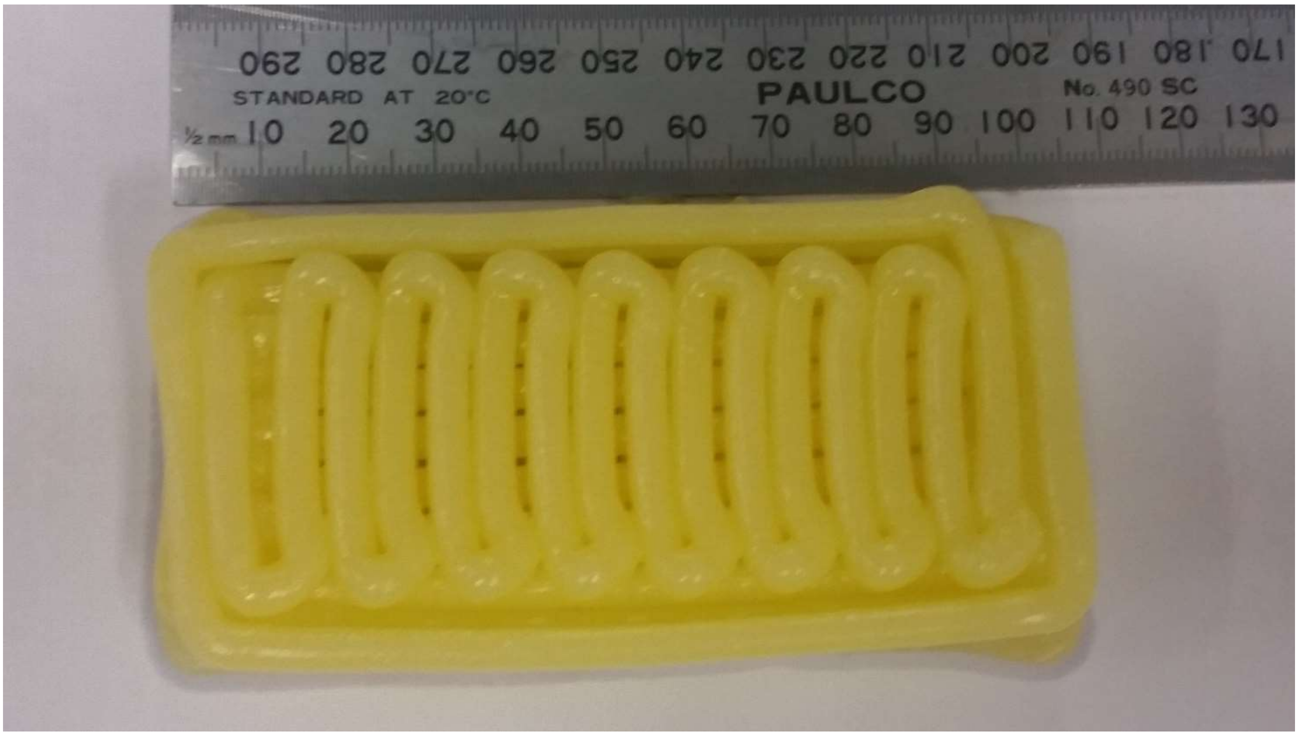


Figure 36. 3D Rectangle Prism Horizontal measurement

The measurement view from the top shows to be approx. 105 mm ±2 mm, the dimension is larger than initial CAD design.

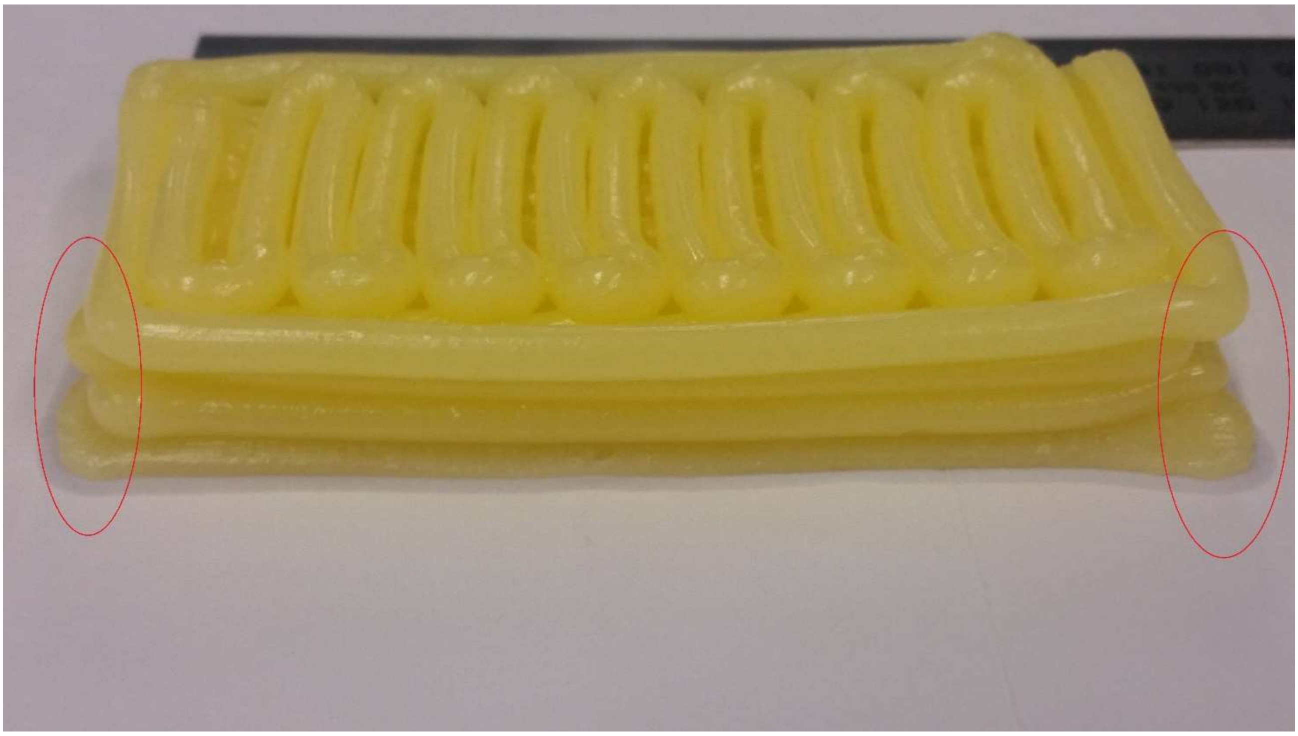


Figure 37. 3D Rectangle Prism Deformation

Looking at the edges of the 3D result, we see the layers are collapsing. This protrusion causes the dimension to stretch out the dimension, overextending the initial dimensions of the CAD design.

The material observed at this stage of testing appear to be more fluid compared to the beginning of the test session, most likely due to the long-time exposure to room temperature.

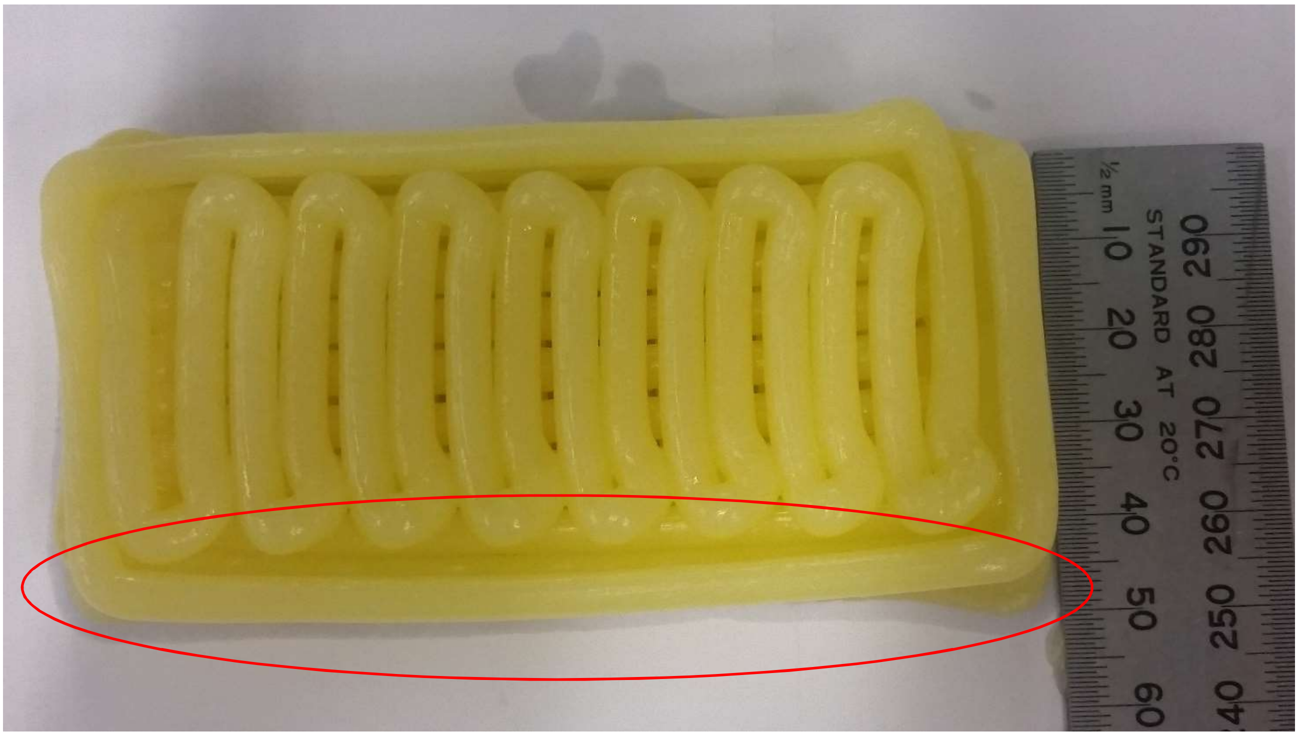


Figure 38. 3D Rectangle Prism Width measurement

The dimension measures approx. to 50 mm however the measurement cannot be made easily as the edges are collapsing slightly, no longer keeping a straight edge.

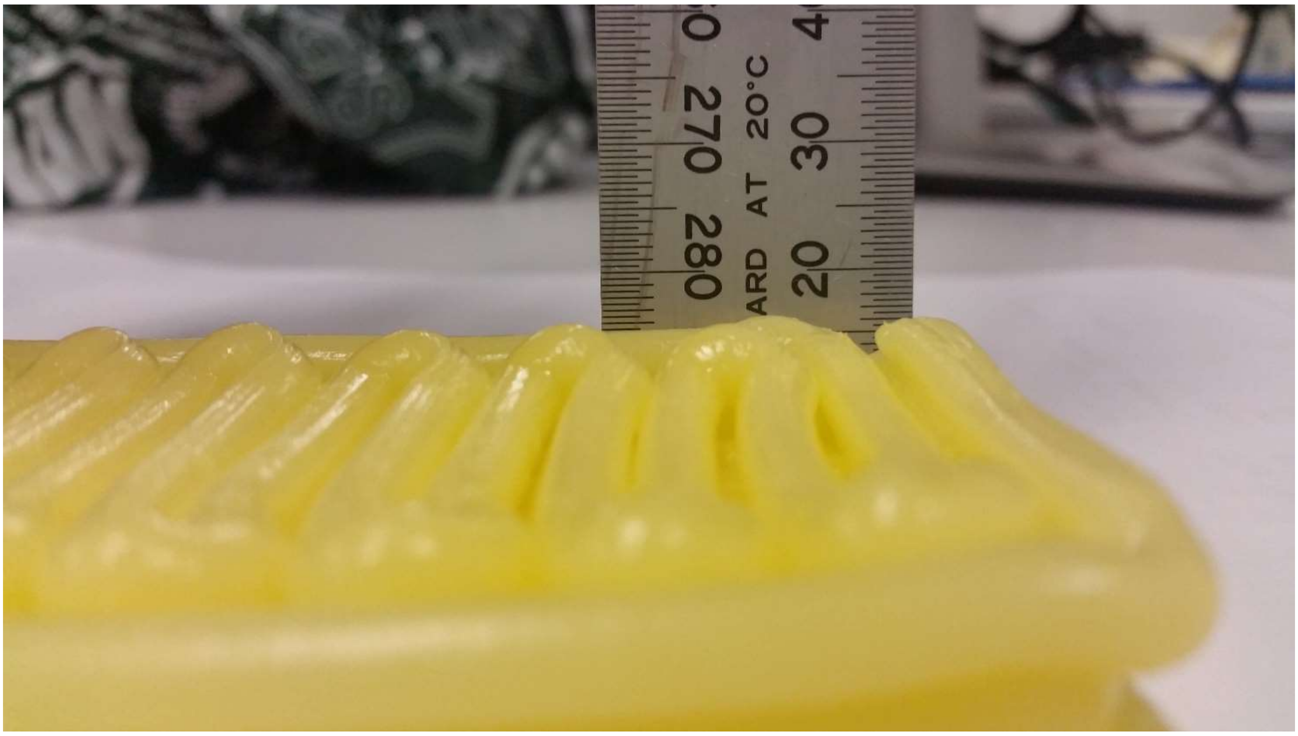


Figure 39. 3D Rectangle Prism Height measurement

The 3D Rectangle Prism is designed to have 4 layers, the original CAD drawings are designed with 5 mm layers so 4 layers lead to 20 mm.

However, the height measurement at the v40 and 1/16 Microstep settings is 4.11 mm (Table 5), this leads to a calculated estimate of 16.44mm height for this experiment result.

The measurement shows the result to be approx. 16 mm, closely matching the estimated height. This is an acceptable result since the original CAD design does not use the experimental settings.

Chapter 6: Discussion

6.1 Food Extruder Tool

The Extruder Tool is designed to fit the IRB 120 and perform 3D food printing processes, an issue that occurred during the process of calibrating the new tool for IRB 120 is the configuring the tool's centre of gravity (cog).

The actual data of tool's cog used in this thesis project is an estimated cog calculated using the tool's dimensions, but the dimensions are measured by eye with a ruler/calliper, this did not pose too great of a problem as the Food Extruder Tool itself was not heavy enough to potentially affect the accuracy of the ABB IRB120 Industrial Robot.

An easier and more accurate method is for the robot to automatically generate the weight, cog, and moments of inertia, this method is using the 'LoadIdentify', load identification service routine. This routine moves in calibration angles of 30° for three axes, axes 3, 5 and 6. [57]

The problem that occurred is the inability to implement the load identification service routine on the Food Extruder Tool design since the tool is permanently parallel to the joint that controls axis 5 movement, during testing the tool would be required to move beyond the 0° margin into negative angles of axis 5, the tool would run into the robot as the angles are in the negative range.

To improve the extruder tool design, the mounting plate that is made to connect the Thick Paste Extruder to the robot can be changed to be like a welding tool/equipment, where the connection is not close to the tool tip but rather the end of the tool at the motor end. [57]

Some minor details of the tool need to be considered when using the Food Extruder Tool, first when the motor is left on for a while with power applied to the DFRobot stepper motor shield, the motor begins heating up, there is no issue with the motor's performance in this condition but does feel warmer on touch.

Another detail is when attaching the wires to connect to the pinout of the motor, for its very 1st start, when everything on the DFRobot stepper motor shield is connected, and power is connected to the stepper shield, connect only 3 out of the 4 wires to the motor's pinout and start the motor before connecting the last wire. Afterwards the motor should run smoothly, the process only needs to be applied to the 1st time after power is applied/connected to the DFRobot stepper shield.

6.2 Material

Two materials were used in preparation for the experiment process, chocolate and cream frosting. Before moving onto discussing the major incident that occurred with chocolate as material, first mentioned in chapter 5, section 5.1.2 Material Test, the cream frosting performed as expected.

Cream frosting is the material used for the entire experimental testing process after the incident with chocolate occurred.

Cream frosting remained fluid in room temperature and extruded easily during the extrusion process, the preparation stage is longer compared to chocolate, however keeping the cream frosting in extrudable condition did not require much, if any, preparation.

Figure 40. Material Storage Component of the Extruder Tool (Syringe barrel)

The result of the incident involving chocolate is shown in Figure 40, possibly due to the design of the Thick Paste Extruder from ZMorph, used for the extruder tool, the flap of the syringe cylinder snapped off under the force during extrusion.

The design of the Thick Paste Extruder has been changed from my model used in this thesis experiment verses the current model displayed in their demonstration video of their 3D printer. [58]

Once the chocolate has being deposited within the syringe barrel, due to the properties of the liquid state of the chocolate, there is no airflow from either end of the syringe. Once the extrusion process begins, the plunger that is attached to the motor of the Thick Paste Extruder is capable of extrusion, however with no airflow, it requires a tremendous force to retract the plunger, which is essentially impossible for the motor or by hand.

Another major issue mention earlier is the perseverance of the chocolate in its liquid state, before trying to remove the plunger component of the syringe, the chocolate was already thickening/hardening within the syringe barrel.

Once removing the plunger was given up, the idea was to proceed to extrude the remaining chocolate from the barrel. Due to the chocolate hardening, the thicker liquid required too much pressure to extrude for the motor, this resulted in the breakage of the syringe barrel used in the chocolate extrusion test.

A potential method to avoid this issue is to reheat the barrel containing the chocolate before continuing the extrusion.

And to solve the retraction of the plunger as airflow are seal anyways once the plunger reaches the end of the extrusion, a gap can be left before plunger reaches the end of the barrel, so the material can be first removed, and airflow can be re-established within the barrel.

Even with the methods to make the process easier, the process of maintaining the chocolate in an extrudable form requires the chocolate to be kept at a specific temperature range, this is difficult to do within this thesis's scope, the Food Extruder Tool will require a device covering the syringe barrel to provide the required temperature.

A temperature sensing device would also be needed to maintain the temperature, as well as applying cooling devices such as a fan to cool down the material after the extrusion is completed so the material solidifies layer by layer without melting.

The process to create melted chocolate is easy but the maintenance factor during extrusion makes it a much harder material to implement in 3D printing processes with industrial robots, compared to the cream frosting material.

6.3 Optimal Operating Speed of IRB 120 robot and Motor of Extruder Tool

6.3.1 Experimental Results

From the results of the experiments, three micro step resolutions were tested against the range of RAPID motion instruction speeds that the IRB 120 robot can operate at.

The optimal speed data can be read from the graph in Figure 41.

Using the results, we can calculate the extrusion rate of the extruder tool, compare the result with the estimated speed and confirm the accuracy of the experimental results.

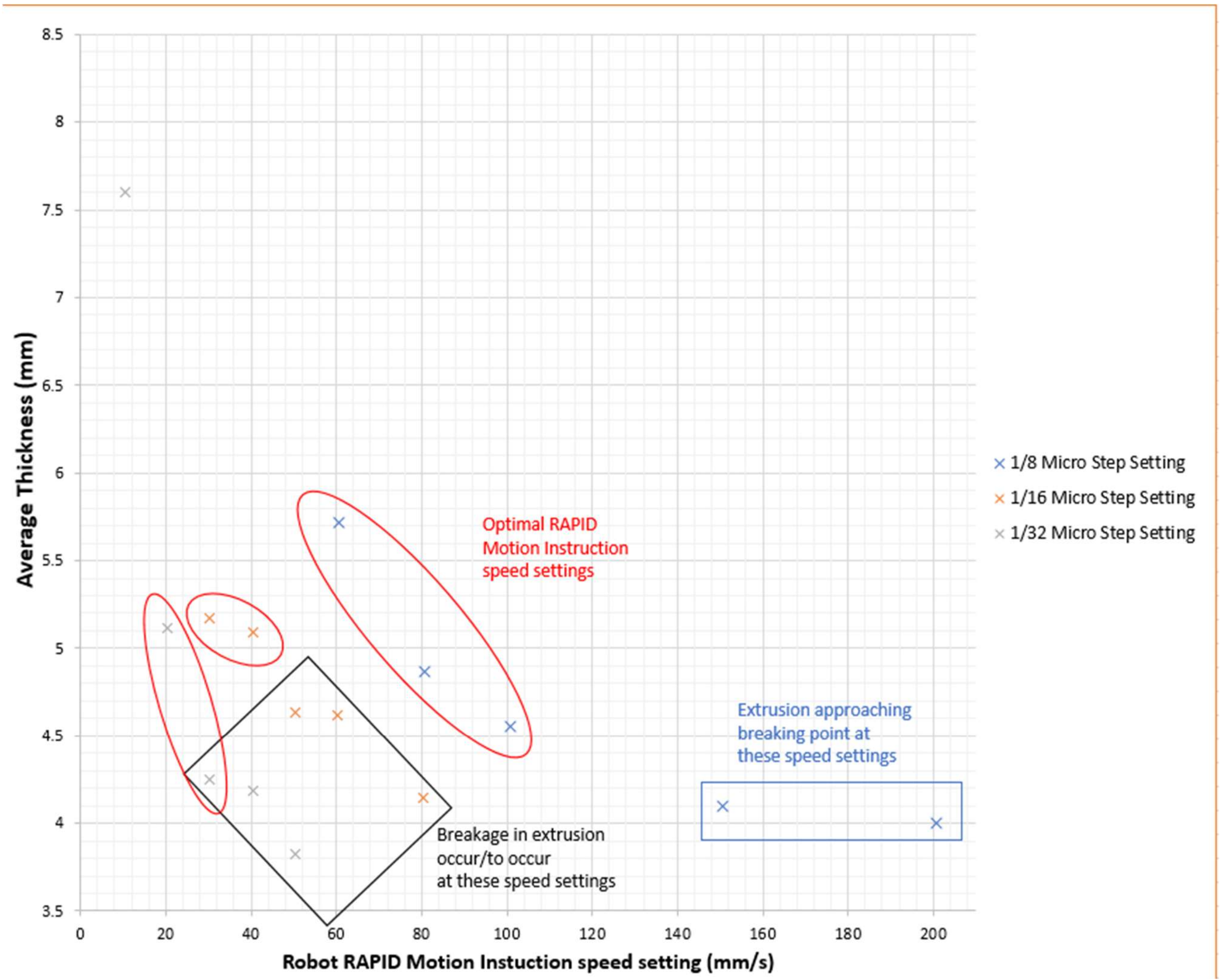


Figure 41. Graph of Speed Calibration Tests

6.3.2 Estimate vs Result

Table 9. Optimal Speed Relation

Microstep Resolution Setting	Estimated RAPID Motion Instruction Speed (mm/s)	RAPID Motion Instruction Speed Results (mm/s)
1/8	27.78	60, 80, 100
1/16	13.89	30, 40
1/32	6.94	20

Comparing the RAPID Motion Instruction Speed Results (Experimental Results) with the Estimated speeds, we can see the data of the Experimental Results are approximately three times larger than the Estimated data.

Microstep resolution:

@1/8: Average Experimental Result Speed = 80 mm/s;

$$Ratio = \frac{Average\ Experimental\ Speed}{Estimated\ Speed} = \frac{80}{27.78} \cong 2.88$$

@ 1/16: Average Experimental Result Speed = 35 mm/s;

$$Ratio = \frac{Average\ Experimental\ Speed}{Estimated\ Speed} = \frac{35}{13.89} \cong 2.52$$

@ 1/32: Average Experimental Result Speed = 20 mm/s;

$$Ratio = \frac{Average\ Experimental\ Speed}{Estimated\ Speed} = \frac{20}{6.94} \cong 2.88$$

We notice after some calculations, the speed ratio shows an increase of approximately 2.88, beside micro step 1/16, however if the average speed of 1/16 is 40 mm/s instead, the ratio would have been 2.88 as well.

This result shows, on average, for the Food Extruder Tool, by multiplying the estimated speed data (V_{out} from section 4.1.2.1.2 calculations) by a factor of 2.88, the industrial robot's RAPID instruction motion speed setting can be set, to produce a desirable result for 3D food printing procedure.

6.4 Printing Results

6.4.1 2D Single Layer Print

The horizontal filled single layer and the vertical filled single layer are printed using v40 RAPID motion instruction speed setting, along with 1/16 Microstep resolution.

The results show the dimensions meet the initial CAD designs however the shorter edges are not perfectly at 90° with the longer side, this is possibly due to slight movement in the tool mounting point since the initial calibration, where the rectangle adapter plate is screwed into the IRB 120 robot, at axis 6.

Due to the mounting plate design, the connection with the IRB 120 robot is only via the rectangle adapter plate, with a single screw holding it on the location of tool0 of the robot.

The design can be improved by instead of using the rectangle adapter plate method, extend the mounting plate and fit 4 screw holes that match the 4 screw hole locations on the axis 6 joint of the robot. This will secure the extruder tool better compared to a single screw, but it is much harder to equip and remove the extruder tool.

From the CAD designs before the test, each path the extrusion to take was left with 5.5 mm distance between each other, the centre of the too tip of the Food Extruder Tool will follow the path and the width of the extrusion must not be larger than 5.5 mm.

The results show that the selected Optimal Speed Result could operate within the dimension, an approx. 1 – 2mm gap is still visible between each line of filling, however the corner locations of the extrusion where some extrusion material was piled on, the 5.5 mm accommodated for the extra filling.

6.4.2 3D Print Result

The 3D printed Rectangular Prism is built up as desired using the two different single layer designs.

A noticeable problem however is the 3D model ended up distorting, the cream frosting material begun to soften causing top layers to collapse while the bottom layers losing strength to support the upper layers.

The problem is solved when the material is swapped with another cream frosting sample batch that were not exposed to room temperature for as long.

The chilled patch meant the material properties of cream frosting is thicker, more solidified than the previous, although this meant the extruder will require more force to extrude the material, the dimensions could also be slightly different to the measured results from the Optimal Speed Tests, however the difference should be still within the 5.5 mm width, left between each line of the extruder tool tip path.

Chapter 7: Conclusion & Further Research

7.1 Food Extruder Tool

The tool performed as a 3D printing extruder too should, performing extrusion of food material at a consistent rate, and maintained its position during the entire 3D printing process.

The tool was easy to attach to IRB 120 robot, simply following the 'new tool' procedure within the manual. [40]

Improvements in the future can be made on the extrusion component of the tool itself, such as improvement of the device holding the material, e.g. a more solidified syringe made from metal/glass.

The motor of the tool can be improved by using a higher torque motor for more force when applying during extrusion process.

The Thick Paste Extruder from ZMorph is adequate as an extrusion device but limited to a range of material only.

Another improvement can be the connection to robot, if both the extrusion device, motor, syringe etc can be designed into a single housing, and that housing is designed with a robot tool fitting. Connecting to ABB industrial robots can be made simpler and produce more accurate results.

7.2 Food Material

The two materials used within the project displayed their pros and cons, in the end only cream frosting is used for majority of the experimental component of the research.

The cream frosting material is an excellent choice for material for 3D printing, the material properties suit every aspect of the extrusion process and after the extrusion is complete, the material maintains its shape.

Chocolate is a common choice as a material for unique 3D printing, however for 3D printing with an industrial robot, since the environment is not enclosed like a common 3D printer, the procedure to maintain its properties for extrusion is more complicated.

A specially designed extrusion tool could be used for chocolate extrusion with 3D printing, however the Food Extruder Tool used in this research is not adequate to be used with chocolate for long periods of time/printing processes.

Other material not tested in this thesis can be chosen to be tested with the Food Extruder Tool in further research, however the material properties must be considered carefully, since as mentioned above, some materials such as chocolate did not work well with the Thick Paste Extruder component of the Food Extruder Tool.

7.3 Optimal Speed

The optimal speed for RAPID motion instruction, relative to rpm and extrusion rate is provided in the table as a summary, since rpm is applicable with any motor/motor controller while Microstep resolution frequently used in the thesis only apply to the DFRobot Stepper Motor Shield used in this research.

From the findings in '5.1.3 Speed Calibration' results and '6.3 Optimal Speed' discussion results, we concluded that by multiplying the estimated speed (which equals to the output velocity at the extruder tool tip) by a factor of 2.88, we produce the average RAPID motion instruction speed the robot can run at to produce a desirable 3D food printing result with the Cream Frosting material.

Table 10. Final Speed Data including all estimations.

Mircrostep Resolution	RPM	V_{out} (mm/s)	RAPID motion instruction speed (mm/s)
Full Step	89.30	222.27	*600* (640*)
Half Step	44.64	111.11	*300* (320*)
1/4	22.32	55.56	*150* (160*)
1/8	11.16	27.78	60, 80, 100
1/16	5.58	13.89	30, 40
1/32	2.79	6.94	20

*The values within brackets are $(2.88)V_{out}$, while the *estimated value* is what the RAPID motion instruction speed can be actually set to. For 1/8, 1/16 and 1/32, the RAPID speeds remain the same from experimental results.

7.4 Printing Result

Both 2D and 3D printing result perfectly demonstrate the capabilities of attaching an extrusion tool used in 3D printing, as a robot tool to the industrial robot for 3D printing to be performed. While using food as a material.

The process in early stages are simple, but this research is limited to implementing an extrusion tool to be used with an industrial robot to perform 3D printing.

The Food Extruder Tool is controlled via PC to be switch ON and OFF, with Stopping implemented but is still controlled manually via LabView.

Now that it is confirmed it is possible to have an extrusion tool added as a robot tool to perform 3D printing, other additional process such as allowing the robot to command the extrusion tool via LabView can be implemented or testing of complex shapes with non-standard pathing.

Appendix A

ABB IRB 120 Robot Datasheet [59]

Appendix B

ARDUINO UNO REV3 SPECIFICATIONS [44]

Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g

APPENDIX C

DFROBOT STEPPER MOTOR SHIELD FOR ARDUINO(DRV8825)

SPECIFICATION [50]

- Support 3.3V and 5V operating voltage
- Suitable for two-phase and four-wire stepper motor
- Board with two DRV8825 driver chip and a heat sink has been mounted.
- Input Voltage：8.2-45V DC (Just power the stepper motor driver), 1.6A output current per coil
- Driving Pins: D4,D5,D6,D7,D8,D12
- Squeeze connector, quite easy and convenient.
- 8 channel digital I/O pins & 6 channel Analog input pins
- DRV8825 Microstepping bipolar stepper motor driver
- Six different microstep resolutions (full-step, 1/2-step, 1/4-step, 1/8-step, 1/16-step, 1/32-step)
- Compatible with Arduino UNO R3, Leonardo, Mega and other controllers, with full port extensions. Support XBee, XBee Wi-Fi, Bluetooth and RF modules, ensure that the needs of your wireless communications.
- There is a switch on the board Xbee wireless interfaces corner. PROG stop wireless module, available USB programming. RUN time can use the wireless communication module.

The new version expand two enable pins, you can enable/disable the motor driver to save the power consumption, Motor X enable pin for the D8, Motor Y enable pin for the D12, low voltage enable, the following truth table:

D8	D12	M1	M2
Low	Low	ENABLE	ENABLE
High	Low	DISENABLE	ENABLE
Low	High	ENABLE	DISENABLE
High	High	DISENABLE	DISENABLE

Dip switch settings for Microstep resolution:

MS1	MS2	MS3	Microstep Resolution
Low	Low	Low	Full step
High	Low	Low	Half step
Low	High	Low	1/4 step
High	High	Low	1/8 step
Low	Low	High	1/16 step
High	Low	High	1/32 step
Low	High	High	1/32 step
High	High	High	1/32 step

APPENDIX D

ARDUINO IDE

Motor Driving Code

```
#define BUFFER_SIZE 32  //Array size
#include <Wire.h>


int M1dirpin = 7;
int M1steppin = 6;
int M1en = 8;
int M2en = 12;



void Moto_RUN()
{
  Serial.println("Motor: RUN\n"); // send string to serial bus
}

//Motor Initialisation and Setup
void setup() {
  Serial.begin(9600);           // set up Serial library at 9600 bps
  Serial.println("Stepper start");  // send string to indicate the code has begun

  pinMode(M1dirpin, OUTPUT);
  pinMode(M1steppin,OUTPUT);
  pinMode(M1en,OUTPUT);

  digitalWrite(M1en,LOW); //Enable (HIGH: Disable, LOW: Enable)
  digitalWrite(M2en,HIGH);  // Disable

}
void loop()
```

```
{
}
void serialEvent()
{
 static char Buffer[BUFFER_SIZE] = "";  // Buffer array
 static char temp[BUFFER_SIZE] = "";    // Temporary array
 char c;
 while(Serial.available())
 {
   c = processCharInput(Buffer, Serial.read());  // Read input from serial buffer
   if (c == '\n')
   {
     // Buffer contents fully recieved once terminal byte is read
     if (strcmp("ON", Buffer) == 0)
     {
          // Proceed for 'ON' command to turn motor on
          do
          {
          Serial.println("Motor: ");
          // Code for DFRobot Stepper shield to drive motor
          int j;
          delayMicroseconds(2);
          digitalWrite(M1dirpin, HIGH);  // Direction setting
          for(j=0; j<=5000;j++)
          {
            // Do not need to be changed
            digitalWrite(M1steppin, LOW);
            delayMicroseconds(2);
            digitalWrite(M1steppin,HIGH);
            delay(1);
```

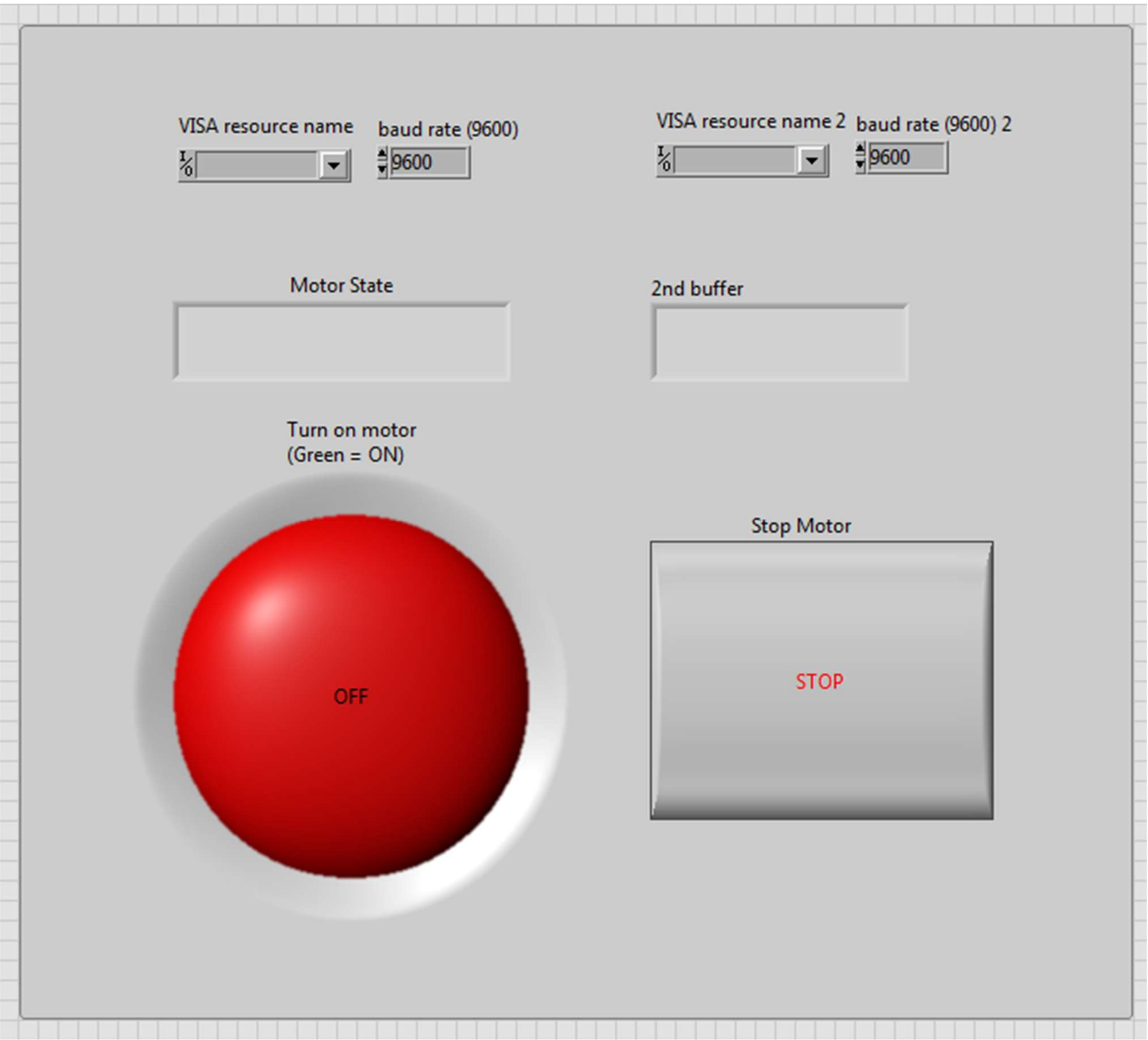
```
        processCharInput(temp, Serial.read());  // Read serial buffer each loop
        }
        Serial.println("ON.\n");  // Motor status is ON
        }while (strcmp("OFF", temp) != 0); // Stop motor when OFF is read from serial buffer
      Serial.println("STOP\n");  // If 'OFF' chosen, STOP is printed to indicate motor has stopped
    }
    if (strcmp("RUN", Buffer) == 0)
    {
      // Motor is recieving RUN string in buffer, labview will proceed to send 'ON' command.
      Moto_RUN();
    }
    Buffer[0] = 0;
  }
 }
 delay(10);
}

char processCharInput(char* Buffer, const char c)
{
 //Store the character in the input buffer
 if (c >= 32 && c <= 126) //Ignore control characters and special ascii characters
 {
  if (strlen(Buffer) < BUFFER_SIZE)
  {
    strncat(Buffer, &c, 1);   //Add it to the buffer
  }
  else
  {
    return '\n';
  }
 }
```

```
 else if ((c == 8 || c == 127) && Buffer[0] != 0) //Backspace
 {

  Buffer[strlen(Buffer)-1] = 0;
 }

 return c;
}
```

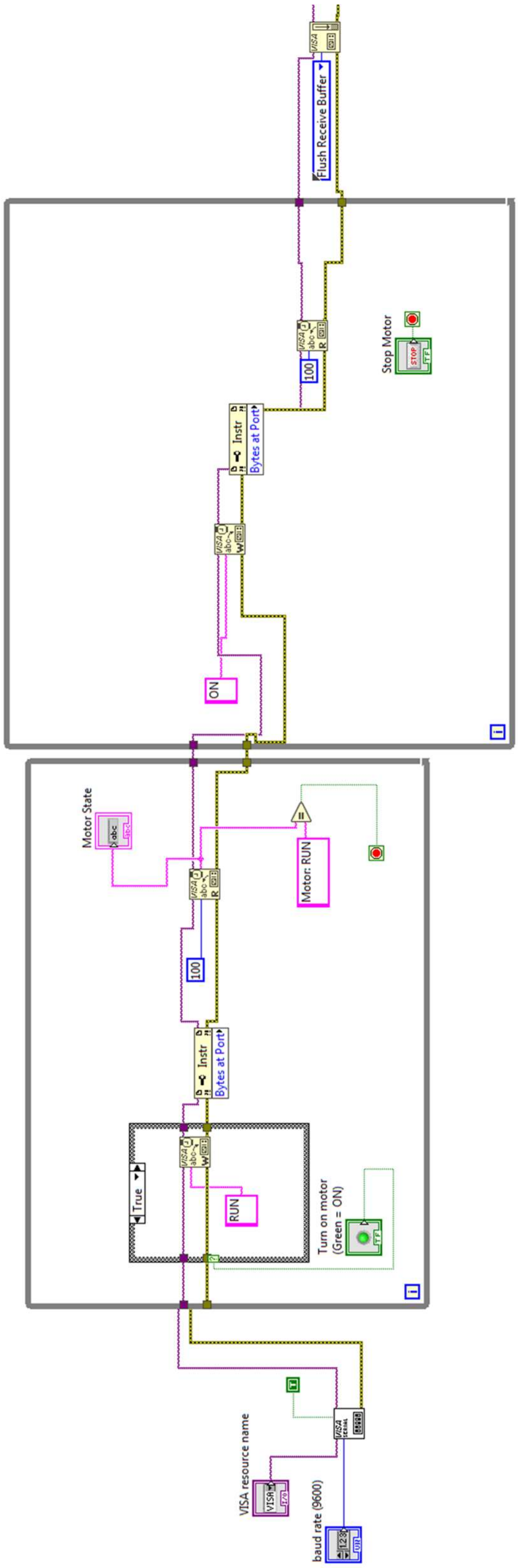
APPENDIX E

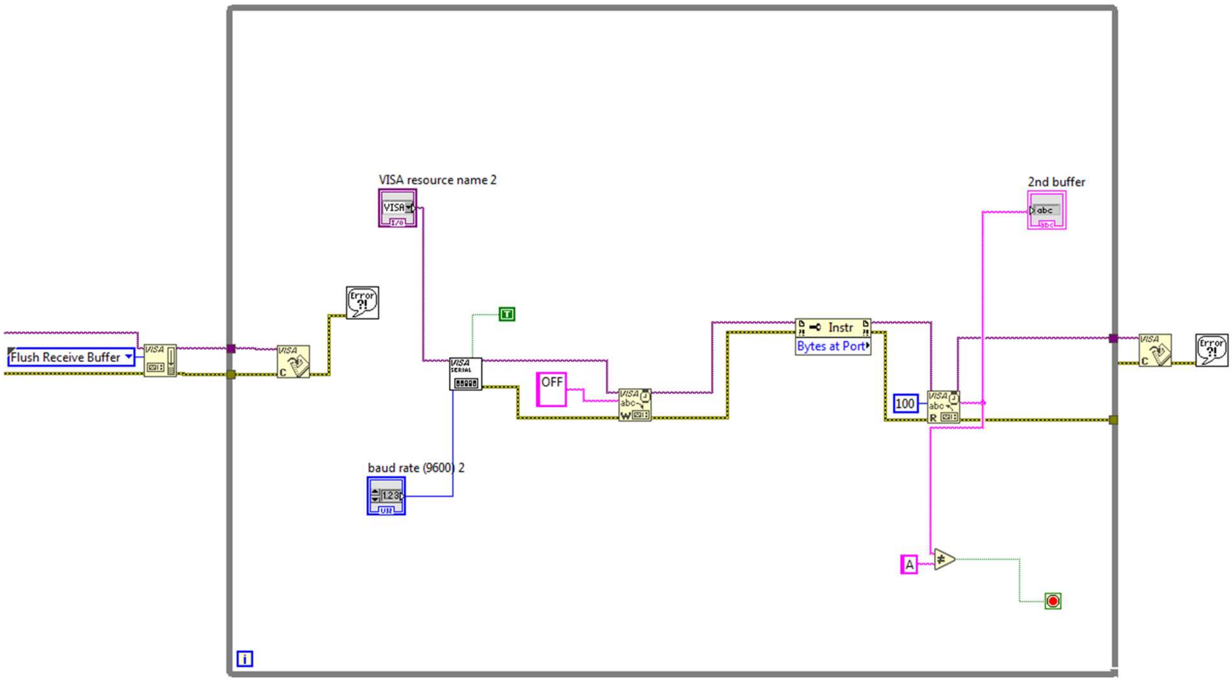
LABVIEW

FRONT PANEL



BLOCK DIAGRAM





VISA resource name 2
2nd buffer
Flush Receive Buffer
Instr
Bytes at Port
OFF
100
baud rate (9600) 2
A

APPENDIX F

ROBOTSTUDIO CODE

Includes all code used in RobotStudio RAPID programming for all tests involving the use of the ABB IRB 120 Robot.

1/8 Microstep Resolution Test Code

```
MODULE FanMod

    CONST robtarget Target_10:=[[400,-75,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_20:=[[400,75,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_30:=[[430,75,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_40:=[[430,-75,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_50:=[[460,-75,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_60:=[[460,75,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_70:=[[490,75,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_80:=[[490,-75,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_90:=[[520,-75,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_100:=[[520,75,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_110:=[[550,75,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_120:=[[550,-75,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];


    PERS tooldata tExtruder:=[TRUE,[[-92.3257,2.14224,443.917],[1,0,0,0]],[0.5,[0,80,5],[1,0,0,0],0,0,0]];
```

```
PROC Fan_Extruder() !Change this to suit.

        MoveJ [[350,0,-300],[0,0,-1,0],[-1,-1,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tExtruder;

        MoveL [[400,-50,-300],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v60, z50, tExtruder;

    MoveL Target_10,v60,fine,tExtruder\WObj:=wobj0;

    Stop;

        MoveL Target_20,v50,fine,tExtruder\WObj:=wobj0;

        MoveL Target_30,v50,fine,tExtruder\WObj:=wobj0;

        MoveL Target_40,v60,fine,tExtruder\WObj:=wobj0;

        MoveL Target_50,v60,fine,tExtruder\WObj:=wobj0;

        MoveL Target_60,v80,fine,tExtruder\WObj:=wobj0;

        MoveL Target_70,v80,fine,tExtruder\WObj:=wobj0;

        MoveL Target_80,v100,fine,tExtruder\WObj:=wobj0;

        MoveL Target_90,v100,fine,tExtruder\WObj:=wobj0;

        MoveL Target_100,v150,fine,tExtruder\WObj:=wobj0;

        MoveL Target_110,v150,fine,tExtruder\WObj:=wobj0;

        MoveL Target_120,v200,fine,tExtruder\WObj:=wobj0;

    Stop;
    MoveJ [[450,0,-100],[0,0,-1,0],[-1,-1,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tExtruder;

ENDPROC
ENDMODULE
```

1/16 Microstep Resolution Test Code

```
MODULE FanMod

    CONST robtarget Target_10:=[[400,-75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_20:=[[400,75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_30:=[[430,75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_40:=[[430,-75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_50:=[[460,-75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_60:=[[460,75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_70:=[[490,75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_80:=[[490,-75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_90:=[[520,-75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_100:=[[520,75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_110:=[[550,75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_120:=[[550,-75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];


    PERS tooldata tExtruder:=[TRUE,[[-92.3257,2.14224,443.917],[1,0,0,0]],[0.5,[0,80,5],
[1,0,0,0],0,0,0]];
```

```
PROC Fan_Extruder() !Change this to suit.
        MoveJ [[350,0,-300],[0,0,-1,0],[-1,-1,-
1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tExtruder;
        MoveL [[400,-50,-300],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v60, z50, tExtruder;
    MoveL Target_10,v60,fine,tExtruder\WObj:=wobj0;
    Stop;
        MoveL Target_20,v50,fine,tExtruder\WObj:=wobj0;
        MoveL Target_30,v50,fine,tExtruder\WObj:=wobj0;
        MoveL Target_40,v30,fine,tExtruder\WObj:=wobj0;
        MoveL Target_50,v30,fine,tExtruder\WObj:=wobj0;
        MoveL Target_60,v40,fine,tExtruder\WObj:=wobj0;
        MoveL Target_70,v40,fine,tExtruder\WObj:=wobj0;
        MoveL Target_80,v50,fine,tExtruder\WObj:=wobj0;
        MoveL Target_90,v50,fine,tExtruder\WObj:=wobj0;
        MoveL Target_100,v60,fine,tExtruder\WObj:=wobj0;
        MoveL Target_110,v60,fine,tExtruder\WObj:=wobj0;
        MoveL Target_120,v80,fine,tExtruder\WObj:=wobj0;
    Stop;
    MoveJ [[450,0,-100],[0,0,-1,0],[-1,-1,-
1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tExtruder;
ENDPROC
ENDMODULE
```

1/32 Microstep Resolution Test Code

```
MODULE FanMod

    CONST robtarget Target_10:=[[400,-75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_20:=[[400,75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_30:=[[430,75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_40:=[[430,-75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_50:=[[460,-75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_60:=[[460,75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_70:=[[490,75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_80:=[[490,-75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_90:=[[520,-75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_100:=[[520,75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_110:=[[550,75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

    CONST robtarget Target_120:=[[550,-75,-387],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];


    PERS tooldata tExtruder:=[TRUE,[[-92.3257,2.14224,443.917],[1,0,0,0]],[0.5,[0,80,5],
[1,0,0,0],0,0,0]];
```

```
PROC Fan_Extruder() !Change this to suit.

        MoveJ [[350,0,-300],[0,0,-1,0],[-1,-1,-
1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tExtruder;

        MoveL [[400,-50,-300],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v60, z50, tExtruder;

    MoveL Target_10,v60,fine,tExtruder\WObj:=wobj0;

    Stop;

        MoveL Target_20,v5,fine,tExtruder\WObj:=wobj0;

        MoveL Target_30,v5,fine,tExtruder\WObj:=wobj0;

        MoveL Target_40,v10,fine,tExtruder\WObj:=wobj0;

        MoveL
Target_50,v10,fine,tExtruder\WObj:=wobj0;

        MoveL Target_60,v20,fine,tExtruder\WObj:=wobj0;

        MoveL Target_70,v20,fine,tExtruder\WObj:=wobj0;

        MoveL Target_80,v30,fine,tExtruder\WObj:=wobj0;

        MoveL Target_90,v30,fine,tExtruder\WObj:=wobj0;

        MoveL Target_100,v40,fine,tExtruder\WObj:=wobj0;

        MoveL Target_110,v40,fine,tExtruder\WObj:=wobj0;

        MoveL Target_120,v50,fine,tExtruder\WObj:=wobj0;

    Stop;
    MoveJ [[450,0,-100],[0,0,-1,0],[-1,-1,-
1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tExtruder;

ENDPROC
ENDMODULE
```

Horizontal Layer Robot Print Path Code

```
MODULE  FanMod

        CONST  robtarget  Target_10:=[[402.75,-50,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

      CONST  robtarget  Target_20:=[[402.75,44.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

      CONST  robtarget  Target_30:=[[447.25,44.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E

+09,9E+09,9E+09,9E+09]];

      CONST  robtarget  Target_40:=[[447.25,-50,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

      CONST  robtarget  Target_50:=[[408.25,-50,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

      CONST  robtarget  Target_60:=[[408.25,39,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

      CONST  robtarget  Target_70:=[[413.75,39,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

      CONST  robtarget  Target_80:=[[413.75,-44.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

      CONST  robtarget  Target_90:=[[419.25,-44.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

      CONST  robtarget  Target_100:=[[419.25,39,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

      CONST  robtarget  Target_110:=[[424.75,39,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

      CONST  robtarget  Target_120:=[[424.75,-44.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

      CONST  robtarget  Target_130:=[[430.25,-44.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];
```

```
CONST robtarget Target_140:=[[430.25,39,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget Target_150:=[[435.75,39,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget Target_160:=[[435.75,-44.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_170:=[[441.25,-44.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_180:=[[441.25,39,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

PERS tooldata tExtruder:=[TRUE,[[-92.3257,2.14224,443.917],[1,0,0,0]],[0.5,[0,80,5],[1,0,0,0],0,0,0]];

PROC Fan_Extruder() !Change this to suit.

        MoveJ [[350,0,-300],[0,0,-1,0],[-1,-1,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tExtruder;
        MoveL Target_10,v50,fine,tExtruder\WObj:=wobj0;
        Stop;
        MoveL Target_20,v40,fine,tExtruder\WObj:=wobj0;
        MoveL Target_30,v40,fine,tExtruder\WObj:=wobj0;
        MoveL Target_40,v40,fine,tExtruder\WObj:=wobj0;
        MoveL Target_50,v40,fine,tExtruder\WObj:=wobj0;
        MoveL Target_60,v40,fine,tExtruder\WObj:=wobj0;
        MoveL Target_70,v40,fine,tExtruder\WObj:=wobj0;
        MoveL Target_80,v40,fine,tExtruder\WObj:=wobj0;
        MoveL Target_90,v40,fine,tExtruder\WObj:=wobj0;
```

```
        MoveL  Target_100,v40,fine,tExtruder\WObj:=wobj0;
        MoveL  Target_110,v40,fine,tExtruder\WObj:=wobj0;
        MoveL  Target_120,v40,fine,tExtruder\WObj:=wobj0;
        MoveL  Target_130,v40,fine,tExtruder\WObj:=wobj0;
        MoveL  Target_140,v40,fine,tExtruder\WObj:=wobj0;
        MoveL  Target_150,v40,fine,tExtruder\WObj:=wobj0;
        MoveL  Target_160,v40,fine,tExtruder\WObj:=wobj0;
        MoveL  Target_170,v40,fine,tExtruder\WObj:=wobj0;
        MoveL  Target_180,v40,fine,tExtruder\WObj:=wobj0;
    Stop;
    MoveL  [[450,110,-300],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]],  v100,  fine,  tExtruder;
    MoveJ  [[450,0,-100],[0,0,-1,0],[-1,-1,-
1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]],  v200,  z50,  tExtruder;
  ENDPROC
  ENDMODULE
```

Vertical Layer Robot Print Path Code

```
MODULE FanMod

CONST robtarget Target_10:=[[441.25,39,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E
+09,9E+09,9E+09]];

CONST robtarget Target_20:=[[407.75,39,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget Target_30:=[[407.75,33.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_40:=[[441.25,33.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_50:=[[441.25,28,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget Target_60:=[[407.75,28,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget Target_70:=[[407.75,22.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_80:=[[441.25,22.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_90:=[[441.25,17,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget Target_100:=[[407.75,17,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget Target_110:=[[407.75,11.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_120:=[[441.25,11.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_130:=[[441.25,6,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];
```

```
CONST robtarget Target_140:=[[407.75,6,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget Target_150:=[[407.75,0.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_160:=[[441.25,0.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E

+09,9E+09,9E+09,9E+09]];

CONST robtarget Target_170:=[[441.25,-5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget Target_180:=[[407.75,-5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget Target_200:=[[441.25,-10.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_190:=[[407.75,-10.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_210:=[[441.25,-16,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_220:=[[407.75,-16,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_230:=[[407.75,-21.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_240:=[[441.25,-21.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_250:=[[441.25,-27,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_260:=[[407.75,-27,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_270:=[[407.75,-32.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];
```

```
CONST robtarget Target_280:=[[441.25,-32.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_290:=[[441.25,-38,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_300:=[[407.75,-38,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_310:=[[407.75,-44.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E
+09,9E+09,9E+09,9E+09]];

CONST robtarget Target_320:=[[447.25,-44.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_330:=[[447.25,44.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E
+09,9E+09,9E+09,9E+09]];

CONST robtarget Target_340:=[[402.75,44.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_350:=[[402.75,-50,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E
+09,9E+09,9E+09,9E+09]];

CONST robtarget Target_360:=[[447.25,-50,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];


PERS tooldata tExtruder:=[TRUE,[[-92.3257,2.14224,443.917],[1,0,0,0]],[0.5,[0,80,5],[1,0,0,0],0,0,0]];


PROC Fan_Extruder() !Change this to suit.

        MoveJ [[350,0,-300],[0,0,-1,0],[-1,-1,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tExtruder;

    MoveL Target_10,v50,fine,tExtruder\WObj:=wobj0;
```

```
Stop;
    MoveL  Target_20,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_30,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_40,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_50,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_60,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_70,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_80,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_90,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_100,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_110,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_120,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_130,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_140,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_150,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_160,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_170,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_180,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_190,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_200,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_210,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_220,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_230,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_240,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_250,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  Target_260,v40,fine,tExtruder\WObj:=wobj0;
```

```
MoveL  Target_270,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target_280,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target_290,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target_300,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target_310,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target_320,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target_330,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target_340,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target_350,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target_360,v40,fine,tExtruder\WObj:=wobj0;
Stop;
MoveJ  [[450,0,-100],[0,0,-1,0],[-1,-1,-
1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]],  v200,  z50,  tExtruder;
ENDPROC
ENDMODULE
```

3D Rectangle Prism Robot Print Path Code

```
MODULE FanMod

        CONST robtarget Target_10:=[[402.75,-50,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

        CONST robtarget Target_20:=[[402.75,44.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

        CONST robtarget Target_30:=[[447.25,44.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E

+09,9E+09,9E+09,9E+09]];

        CONST robtarget Target_40:=[[447.25,-50,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

        CONST robtarget Target_50:=[[408.25,-50,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

        CONST robtarget Target_60:=[[408.25,39,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

        CONST robtarget Target_70:=[[413.75,39,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

        CONST robtarget Target_80:=[[413.75,-44.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

        CONST robtarget Target_90:=[[419.25,-44.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

        CONST robtarget Target_100:=[[419.25,39,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

        CONST robtarget Target_110:=[[424.75,39,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

        CONST robtarget Target_120:=[[424.75,-44.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

        CONST robtarget Target_130:=[[430.25,-44.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];
```

```
CONST robtarget Target_140:=[[430.25,39,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget Target_150:=[[435.75,39,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget Target_160:=[[435.75,-44.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_170:=[[441.25,-44.5,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target_180:=[[441.25,39,-387],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert_10:=[[441.25,39,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E

+09,9E+09,9E+09]];

CONST robtarget vert_20:=[[407.75,39,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert_30:=[[407.75,33.5,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E

+09,9E+09,9E+09]];

CONST robtarget vert_40:=[[441.25,33.5,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert_50:=[[441.25,28,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E

+09,9E+09,9E+09]];

CONST robtarget vert_60:=[[407.75,28,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert_70:=[[407.75,22.5,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E

+09,9E+09,9E+09]];
```

CONST robtarget vert_80:=[[441.25,22.5,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert_90:=[[441.25,17,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert_100:=[[407.75,17,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert_110:=[[407.75,11.5,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert_120:=[[441.25,11.5,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert_130:=[[441.25,6,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert_140:=[[407.75,6,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert_150:=[[407.75,0.5,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert_160:=[[441.25,0.5,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert_170:=[[441.25,-5,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert_180:=[[407.75,-5,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert_200:=[[441.25,-10.5,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget vert_190:=[[407.75,-10.5,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget vert_210:=[[441.25,-16,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

```
CONST robtarget vert_220:=[[407.75,-16,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];
CONST robtarget vert_230:=[[407.75,-21.5,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];
CONST robtarget vert_240:=[[441.25,-21.5,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];
CONST robtarget vert_250:=[[441.25,-27,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];
CONST robtarget vert_260:=[[407.75,-27,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];
CONST robtarget vert_270:=[[407.75,-32.5,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];
CONST robtarget vert_280:=[[441.25,-32.5,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];
CONST robtarget vert_290:=[[441.25,-38,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];
CONST robtarget vert_300:=[[407.75,-38,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];
CONST robtarget vert_310:=[[407.75,-44.5,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];
CONST robtarget vert_320:=[[447.25,-44.5,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E

+09,9E+09,9E+09,9E+09]];
CONST robtarget vert_330:=[[447.25,44.5,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];
CONST robtarget vert_340:=[[402.75,44.5,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];
CONST robtarget vert_350:=[[402.75,-50,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];
```

```
CONST robtarget vert_360:=[[447.25,-50,-382],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget Target2_10:=[[402.75,-50,-377],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target2_20:=[[402.75,44.5,-377],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target2_30:=[[447.25,44.5,-377],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target2_40:=[[447.25,-50,-377],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target2_50:=[[408.25,-50,-377],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E

+09,9E+09,9E+09,9E+09]];

CONST robtarget Target2_60:=[[408.25,39,-377],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget Target2_70:=[[413.75,39,-377],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget Target2_80:=[[413.75,-44.5,-377],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target2_90:=[[419.25,-44.5,-377],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target2_100:=[[419.25,39,-377],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target2_110:=[[424.75,39,-377],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target2_120:=[[424.75,-44.5,-377],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target2_130:=[[430.25,-44.5,-377],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];
```

```
CONST robtarget Target2_140:=[[430.25,39,-377],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target2_150:=[[435.75,39,-377],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target2_160:=[[435.75,-44.5,-377],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target2_170:=[[441.25,-44.5,-377],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget Target2_180:=[[441.25,39,-377],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget vert2_10:=[[441.25,39,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert2_20:=[[407.75,39,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert2_30:=[[407.75,33.5,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert2_40:=[[441.25,33.5,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert2_50:=[[441.25,28,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert2_60:=[[407.75,28,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert2_70:=[[407.75,22.5,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert2_80:=[[441.25,22.5,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert2_90:=[[441.25,17,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];
```

CONST robtarget vert2_100:=[[407.75,17,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert2_110:=[[407.75,11.5,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget vert2_120:=[[441.25,11.5,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget vert2_130:=[[441.25,6,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert2_140:=[[407.75,6,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert2_150:=[[407.75,0.5,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E

+09,9E+09,9E+09]];

CONST robtarget vert2_160:=[[441.25,0.5,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert2_170:=[[441.25,-5,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E

+09,9E+09,9E+09]];

CONST robtarget vert2_180:=[[407.75,-5,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert2_200:=[[441.25,-10.5,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E

+09,9E+09,9E+09,9E+09]];

CONST robtarget vert2_190:=[[407.75,-10.5,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget vert2_210:=[[441.25,-16,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E

+09,9E+09,9E+09]];

CONST robtarget vert2_220:=[[407.75,-16,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert2_230:=[[407.75,-21.5,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E

+09,9E+09,9E+09,9E+09]];

CONST robtarget vert2_240:=[[441.25,-21.5,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget vert2_250:=[[441.25,-27,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E

+09,9E+09,9E+09]];

CONST robtarget vert2_260:=[[407.75,-27,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert2_270:=[[407.75,-32.5,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E

+09,9E+09,9E+09,9E+09]];

CONST robtarget vert2_280:=[[441.25,-32.5,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

CONST robtarget vert2_290:=[[441.25,-38,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E

+09,9E+09,9E+09]];

CONST robtarget vert2_300:=[[407.75,-38,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

CONST robtarget vert2_310:=[[407.75,-44.5,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E

+09,9E+09,9E+09,9E+09]];

CONST robtarget vert2_320:=[[447.25,-44.5,-372],[0,0,-1,0],[-1,0,-1,0],[9E+09,9E+09,9E +09,9E+09,9E+09,9E+09]];

```
CONST robtarget vert2_330:=[[447.25,44.5,-372],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E

+09,9E+09,9E+09,9E+09]];

CONST robtarget vert2_340:=[[402.75,44.5,-372],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E

+09,9E+09,9E+09,9E+09]];

CONST robtarget vert2_350:=[[402.75,-50,-372],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E

+09,9E+09,9E+09]];

CONST robtarget vert2_360:=[[447.25,-50,-372],[0,0,-1,0],[-1,0,-
1,0],[9E+09,9E+09,9E+09,9E +09,9E+09,9E+09]];

PERS tooldata tExtruder:=[TRUE,[[-92.3257,2.14224,443.917],[1,0,0,0]],[0.5,[0,80,5],
[1,0,0,0],0,0,0]];

PROC Fan_Extruder() !Change this to suit.

        MoveJ [[350,0,-300],[0,0,-1,0],[-1,-1,-
1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tExtruder;

         MoveLTarget_10,v50,fine,tExtruder\WObj: wobj0;

         Stop;

         MoveL Target_20,v40,fine,tExtruder\WObj:=wobj0;

         MoveL Target_30,v40,fine,tExtruder\WObj:=wobj0;

         MoveL Target_40,v40,fine,tExtruder\WObj:=wobj0;

         MoveL Target_50,v40,fine,tExtruder\WObj:=wobj0;

         MoveL Target_60,v40,fine,tExtruder\WObj:=wobj0;

         MoveL Target_70,v40,fine,tExtruder\WObj:=wobj0;

         MoveL Target_80,v40,fine,tExtruder\WObj:=wobj0;

         MoveL Target_90,v40,fine,tExtruder\WObj:=wobj0;
```

```
MoveL  Target_100,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target_110,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target_120,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target_130,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target_140,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target_150,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target_160,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target_170,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target_180,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_10,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_20,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_30,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_40,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_50,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_60,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_70,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_80,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_90,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_100,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_110,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_120,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_130,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_140,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_150,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_160,v40,fine,tExtruder\WObj:=wobj0;
```

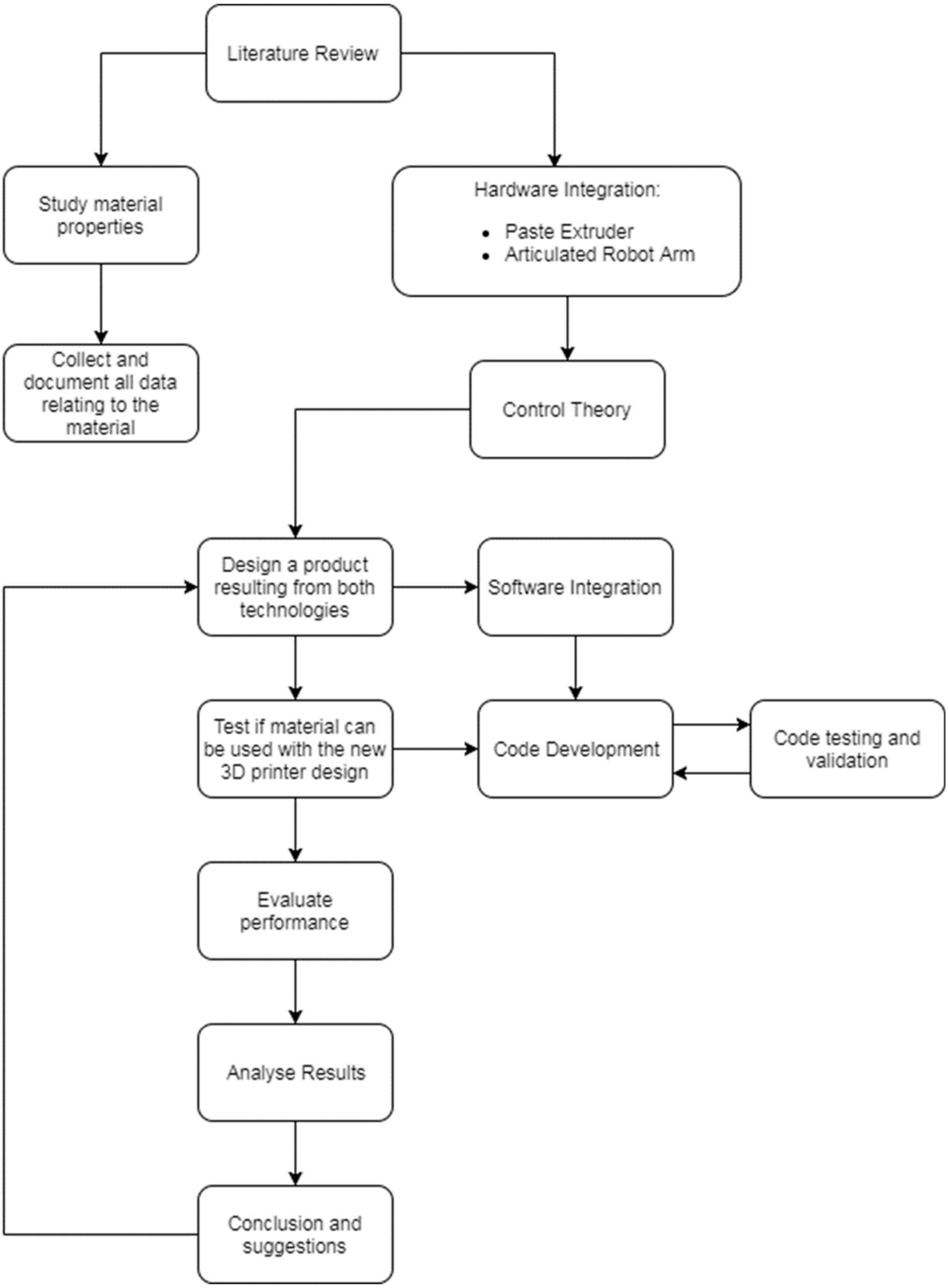
```
MoveL  vert_170,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_180,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_190,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_200,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_210,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_220,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_230,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_240,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_250,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_260,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_270,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_280,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_290,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_300,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_310,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_320,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_330,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_340,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_350,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert_360,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target2_10,v100,fine,tExtruder\WObj:=wobj0;
MoveL  Target2_20,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target2_30,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target2_40,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target2_50,v40,fine,tExtruder\WObj:=wobj0;
```

```
MoveL  Target2_60,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target2_70,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target2_80,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target2_90,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target2_100,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target2_110,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target2_120,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target2_130,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target2_140,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target2_150,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target2_160,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target2_170,v40,fine,tExtruder\WObj:=wobj0;
MoveL  Target2_180,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert2_10,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert2_20,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert2_30,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert2_40,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert2_50,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert2_60,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert2_70,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert2_80,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert2_90,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert2_100,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert2_110,v40,fine,tExtruder\WObj:=wobj0;
MoveL  vert2_120,v40,fine,tExtruder\WObj:=wobj0;
```

```
    MoveL  vert2_130,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  vert2_140,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  vert2_150,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  vert2_160,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  vert2_170,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  vert2_180,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  vert2_190,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  vert2_200,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  vert2_210,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  vert2_220,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  vert2_230,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  vert2_240,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  vert2_250,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  vert2_260,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  vert2_270,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  vert2_280,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  vert2_290,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  vert2_300,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  vert2_310,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  vert2_320,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  vert2_330,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  vert2_340,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  vert2_350,v40,fine,tExtruder\WObj:=wobj0;
    MoveL  vert2_360,v40,fine,tExtruder\WObj:=wobj0;
Stop;
```

```
MoveJ [[450,0,-100],[0,0,-1,0],[-1,-1,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]], v200, z50, tExtruder;
ENDPROC
ENDMODULE
```

APPENDIX G: Work Plan



APPENDIX H: Gantt Chart

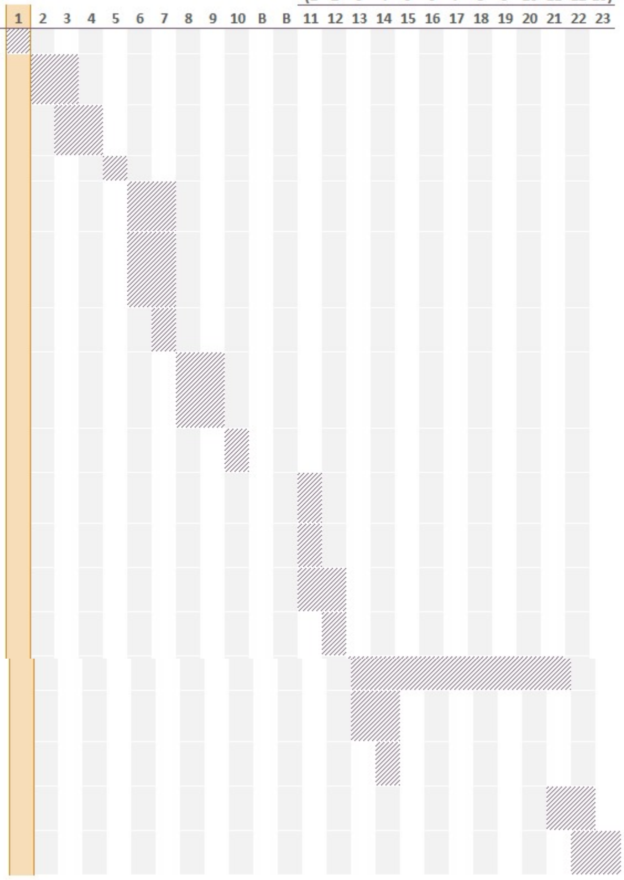
Project Planner

Select a period to highlight at right. A legend describing the charting follows.

Period Highlight: 1 | Plan Duration | Actual Start | % Complete | Actual (beyond plan) | % Complete (beyond

ACTIVITY	PLAN START	PLAN DURATION	ACTUAL START	ACTUAL DURATION	PERCENT COMPLETE
Design 3D printer	1	1	0	0	0%
Construct Extrusion System	2	2	0	0	0%
Develop Software for Extrusion System	3	2	0	0	0%
Test Extrusion System	5	1	0	0	0%
Complete Extrusion System	6	2	0	0	0%
Assemble Extrusion System with Industrial Robot	6	2	0	0	0%
Develop Software for Industrial Robot	7	1	0	0	0%
Develop Software for Entire 3D Printing System	8	2	0	0	0%
Test Entire System	10	1	0	0	0%
Complete Prototype of 3D Printer System	11	1	0	0	0%
Test Materials	11	1	0	0	0%
Analyse Performance	11	2	0	0	0%
Make Final Adjustments	12	1	0	0	0%
Draft Thesis	13	9	0	0	0%
Finish all testing and results	13	2	0	0	0%
Study Results	14	1	0	0	0%
Draft Oral Presentation	21	2	0	0	0%
Final Thesis	22	2	0	0	0%

WEEKS (Summer Session)(10 weeks) WEEKS(Autumn Session)
(1 2 3 4 5 6 7 8 9 10 11 12 13)
1 2 3 4 5 6 7 8 9 10 B B 11 12 13 14 15 16 17 18 19 20 21 22 23



References

[1] Pucci, J.U, Christophe, B.R, Sisti, J.A, Connolly Jr, E.S, "Three-dimensional printing: technologies, applications, and limitations in neurosurgery", *Biotechnology Advances*, vol. 35, Iss. 5, pp. 521-529, 2017, <https://doi.org/10.1016/j.biotechadv.2017.05.007>

[2] MakerBot Industries, "THE ALL-NEW MAKERBOT REPLICATOR+", 2009-2017, Available: <https://www.makerbot.com/replicator/>

[3] Comotti, C, Regazzoni, D, Rizzi, C, Vitali, A, "Additive Manufacturing to Advance Functional Design: An Application in the Medical Field", *Journal of Computing and Information Science in Engineering*, Vol. 17, Iss. 3, 9 pages. 2017, <http://computingengineering.asmedigitalcollection.asme.org/>

[4] Ackland, D.C, Robinson, D, Redhead, M, Lee, P.VS, Moskaljuk, A, "A personalized 3D-printed prosthetic join replacement for the human temporomandibular joint: From implant design to implantation", J*ournal of the Mechanical Behavior of Biomedical Materials*, Vol. 69, pp. 404-411. 2017,<https://doi.org/10.1016/j.jmbbm.2017.01.048>

[5] 3D Systems, Inc. What is Stereolithography (SLA)?, Resources, 2017, Available: <https://au.3dsystems.com/resources/information-guides/stereolithography/sla>

[6] 3D Hubs, 3D Printing Materials, 2017, Available: <https://www.3dhubs.com/materials>

[7] Yang, F, Zhang, M & Bhandari, B, "Recent development in 3D food printing", *Critical Reviews in Food Science and Nutrition*, Vol. 57, Iss. 14, pp. 3145-3153. 2015, <http://dx.doi.org/10.1080/10408398.2015.1094732>

[8] Jovanović, B., Gadjanski, I., Burazer, J., Nikolić, L., Babić, N., Lečić, M., "R&D in a Fab Lab: Examples of Paste Extrusion Method", *In: Majstorovic V., Jakovljevic Z. (eds) Proceedings of 5th International Conference on Advanced Manufacturing Engineering and Technologies. NEWTECH 2017. Lecture Notes in Mechanical Engineering*. Springer, Cham. 2017, <https://doi-org.ezproxy.uow.edu.au/10.1007/978-3-319-56430-2_35>

[9] Printrbot, 2017, Paste & Food Extruder, last viewed on 08/16/2017, Available: <https://printrbot.com/shop/paste-food-extruder/>

[10] Anthony, S, The world's first robotic kitchen prepares crab bisque for Ars Technica, arsTECHNICA, weblog post, 14 April 2015, Available: <https://arstechnica.com/gadgets/2015/04/the-worlds-first-robotic-kitchen-prepares-crab-bisque-for-ars-technica/>

[11] Savini, A & Savini, G.G, ‘A short History of 3D Printing, a Technological Revolution Just Started’, ICOHTEC/IEEE International History of High-Technologies and their Socio-Cultural Contexts Conference (HISTELCON), Tel-Aviv, pp. 1-8. 2015, doi: 10.1109/HISTELCON.2015.7307314

[12] Hu, C & Yin, G, ‘3D printing, a new digital manufacturing mode’, pp. 333-342. 2014, doi: 10.3233/978-1-61499-440-4-333.

[13] Dassault Systèmes, SOLIDWORKS Help: STL Files (*.stl), Dassault Systèmes, 2017, Available: <http://help.solidworks.com/2014/English/solidworks/sldworks/c_stl_files.htm>

[14] Gibson, I, Rosen, D & Stucker, B, Additive Manufacturing Technologies: 3D Printing, Rapid Prototyping, and Direct Digital Manufacturing, 2nd Edn, Springer, New York. 2015,

[15] Ultimaker, “Ultimaker 3”, 2017, Available: <https://ultimaker.com/en/products/ultimaker-3>

[16] XYZprinting, Inc, “da Vinci 1.0 Pro”, 2017, Available: <https://www.xyzprinting.com/en-US/product/da-vinci-pro >

[17] ZMorph, “ZMORPH 2.0 VX FULL SET”, 2017, Available: <https://store.zmorph3d.com/products/zmorph-2-0-vx-full-set>

[18] Gross, B.C, Erkal, J.L, Lockwood, S.Y, Chen, C, Spence, D.M, ‘Evaluation of 3D Printing and Its Potential Impact on Biotechnology and the Chemical Sciences’, Analytical chemistry 86 (7), pp. 3240-3253, 2014. doi:10.1021/ac403397r.

[19] Hull, C. W, 1986, Apparatus for production of three-dimensional objects by stereolithography. U.S. Patent 4,575,330.

[20] Manapat, J.Z, Chen, Q, Ye, P, Advincula, R.C, ‘3D Printing of Polymer Nanocomposites via Stereolithography’, Macromolecular Materials and Engineering, Macromol. Mater. Eng. 2017, 302, 1600553, 2017, <https://doi.org/10.1002/mame.201600553>

[21] Crump, S. S, 1992, Apparatus and method for creating three-dimensional objects, U.S. Patent 5121329 A.

[22] Ligon, S.C, Liska, R, Stampfl, J, Gurr, M, Mülhaupt, R, ‘Polymers for 3D Printing and Customized Additive Manufacturing’, Chemical Reviews 117(15), pp.10212-10290, 2017, DOI: 10.1021/acs.chemrev.7b00074

[23] Wegrzyn, T.F, Golding, M & Archer, R.H, ‘Food Layered Manufacture: A new process for constructing solid foods’, Trends in Food Science & Technology, Volume 27, Issue 2, Pages 66-72, ISSN 0924-2244, 2012, https://doi.org/10.1016/j.tifs.2012.04.006.

[24] Yang, J, Wu, L, & Liu, J, 2001, Rapid prototyping and fabrication method for 3-D food objects, U.S. Patent No. 6280785.

[25] Periard, D, Schaal, N, Schaal, M, Malone, E, Lipson, H, 'Printing food', Proceedings of the 18th Solid Freeform Fabrication Symposium, pp. 564-574, 2007,

[26] CandyFab, "The CandyFab Project", 2007-2014, Available: <http://candyfab.org/>

[27] Lipton, J, Arnold, D, Nigl, F, Lopez, N, Cohen, D, Norén, N, Lipson, H, 'Multi-material food printing with complex internal structure suitable for conventional post-processing', 2010.

[28] Hao, L, Mellor, S, Seaman, O, Henderson, J, Sewell, N, Sloan, M, 'Material characterisation and process development for chocolate additive layer manufacturing', Virtual and Physical Prototyping Vol. 5, Iss. 2, 2010.

[29] Yang, F, Zhang, M, Bhandari, B, Liu, Y, 'Investigation on lemon juice gel as food material for 3D printing and optimization of printing parameters', LWT - Food Science and Technology, Volume 87, 2018, Pages 67-76, 2017, ISSN 0023-6438, https://doi.org/10.1016/j.lwt.2017.08.054.

[30] Sun, J, Peng, Z, Yan, L, Fuh, J, & Hong, G, '3D food printing an innovative way of mass customization in food fabrication', International Journal of Bioprinting, vol. 1(1), pp. 27-38, 2015, doi: http://dx.doi.org/10.18063/IJB.2015.01.006.

[31] Liu, Z, Zhang, M, Bhandari, B, Wang, Y, '3D printing: Printing precision and application in food sector', Trends in Food Science & Technology, Volume 69, Part A, Pages 83-94, 2017, ISSN 0924-2244, <https://doi.org/10.1016/j.tifs.2017.08.018>

[32] Park, R, BITTER SWEET – 3D SYSTEMS ACQUIRES THE SUGAR LAB, 3D Printing Industry, weblog post, 12 September 2013, Available: <https://3dprintingindustry.com/news/bitter-sweet-3d-systems-acquires-the-sugar-lab-16923/>

[33] News Desk, Barilla and TNO collaborate on 3D pasta printing technology, FoodBev Media Ltd, weblog post, 19 May 2015, Available: <https://www.foodbev.com/news/barilla-and-tno-collaborate-on-3d-pasta-printing-technology/>

[34] 3D Systems, Inc, CocoJetTM chocolate 3D printer developed with The Hershy Company Advances 3DS' culinary 3D printing reach into the hospitality space, 3D Systems, Inc, weblog post, 6 Jan 2015, Available: <https://au.3dsystems.com/press-releases/3d-systems-previews-new-chocolate-3d-printer-cocojettm-2015-international-ces>

[35] Natural Machines, "PRESS KIT", 2017, Available: <https://www.naturalmachines.com/press-kit/>

[36] Tufvesson, A, “Will cooks one day be replaced by robots? Automated workers, 3D printing and machines that cook entire meals could well change the way restaurants feed customers”, Restaurant & Catering magazine, weblog post, 12 July 2017, Available: <http://rca.asn.au/magazine/rise-of-the-machines-2/>

[37] Zhao, H, Wang, J, Ren, X, Li, J, Yang, Y.L, Jin, X, ‘Personalized food printing for portrait images’, Computers & Graphics, 2017, , ISSN 0097-8493, <https://doi.org/10.1016/j.cag.2017.07.012>

[38] Brueckner, R, Sugar painting(糖画), InternChina, weblog post, 29 May 2014, Available: <https://internchina.com/sugar-painting%E7%B3%96%E7%94%BB/>

[39] Groover, M.P, 2013, Modern Manufacturing, SI Version, 5th edn, 2011 John Wiley & Sons Singapore Pte. Ltd.

[40] ABB, “Product specification IRB 120”, ABB Group, 2010-2017, Available: https://search-ext.abb.com/library/Download.aspx?DocumentID=3HAC035960-001&LanguageCode=en&DocumentPartId=&Action=Launch

[41] Zhang, G.Q, Mondesir, W, Martinez, C, Li, X, Fuhlbrigge, T.A, Bheda, H, “Robotic Additive Manufacturing along Curved Surface – a Step towards Free-form Fabrication”, *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Zhuhai, 2015, pp. 721-726. <10.1109/ROBIO.2015.7418854>

[42] Brooks, B.J, Arif, K.M, Dirven, S, Potgieter, J, ‘Robot-assisted 3D printing of biopolymer thin shells’, Int J Adv Manuf Technol (2017) 89: 957. 2016, <https://doi-org.ezproxy.uow.edu.au/10.1007/s00170-016-9134-y>

[43] ZMorph, “Thick Paste Extruder”, 2017, Available: <https://store.zmorph3d.com/products/thick-paste-extruder?taxon_id=17>

[44] Arduino, “ARDUINO UNO REV3”, 2017, Available: <https://store.arduino.cc/usa/arduino-uno-rev3>

[45] Adafruit, “DESCRIPTION”, 2017, Available: <https://www.adafruit.com/product/1438>

[46] ABB Robotics, “Operating manual IRC5 with FlexPendant”, ABB Group, 2004-2013, http://isa.uniovi.es/~jalvarez/abb/en/3HAC16590-en.pdf

[47] STEPPERONLINE, “Nema 17 Non-captive 48mm Stack 1.68A Lead 2mm/0.07874" Length 200mm”, omc-stepperonline.com, Available: <https://www.omc-stepperonline.com/linear-motor/nema-17-non-captive-48mm-stack-168a-lead-2mm007874-length-200mm-17ls19-1684n-200d.html>

[48] ZMorph Sales Team, Motor plug pinout diagram, 18 august 2017, Email from sales@zmorph3d.com,

[49] Manfrotto, "Quick Change Rectangular Plate Adapter, universal threads", 1996-2018, Available: <https://www.manfrotto.us/quick-change-rectangular-plate-adapter-universal-threads>

[50] DFRobot, "Stepper Motor Shield For Arduino(DRV8825) SKU:DRI0023", DFRobot, 29 June 2017, Available: <https://www.dfrobot.com/wiki/index.php/Stepper_Motor_Shield_For_Arduino(DRV8825)_SKU:DRI0023>

[51] NATIONAL INSTRUMENTS, "What Is LabVIEW?", 2018, Available: <http://www.ni.com/en-au/shop/labview.html>

[52] ABB Group, "Operating manual RobotStudio", ABB, 2008-2018, Available: <http://search-ext.abb.com/library/Download.aspx?DocumentID=3HAC032104-001&LanguageCode=en&DocumentPartId=&Action=Launch>

[53] ABB Robotics, "Technical reference manual RAPID Instructions, Functions and Data types", ABB Group, 2004-2010, Available: <https://library.e.abb.com/public/688894b98123f87bc1257cc50044e809/Technical%20reference%20manual_RAPID_3HAC16581-1_revJ_en.pdf>

[54] ABB, "Editing the tool data", ABB Group, 2018, Available: <http://developercenter.robotstudio.com/BlobProxy/manuals/IRC5FlexPendantOpManual/doc100.html>

[55] ALL ABOUT CIRCUITS, "Stepper Motor Calculator", allaboutcircuits.com, 2018, Available: <https://www.allaboutcircuits.com/tools/stepper-motor-calculator/ >

[56] Yunus A. Çengel and John M. Cimbala, "BERNOULLI AND ENERGY EQUATIONS", in *FLUID MECHANICS: FUNDAMENTALS AND APPLICATIONS*, 3rd edition in SI units, Singapore, McGraw-Hill Education, 2014, ch. 5, pp. 187-193.

[57] ABB, "LoadIdentify, load identification service routine", ABB Group, 2018, Available <http://developercenter.robotstudio.com/BlobProxy/manuals/IRC5FlexPendantOpManual/doc138.html>

[58] ZMorph, "Chocolate printing with ZMorph Personal Fabricator 2.0 S", YouTube, Apr. 3, 2015, [video file], Available: <https://youtu.be/1H7oQrSBEbc>

[59] ABB, "Technical data for the IRB 120 industrial robot", ABB Group, pp. 2, 2018, Available: <https://new.abb.com/products/robotics/industrial-robots/irb-120/irb-120-data>