The University of Waikato
Faculty of Science and Engineering

Development of a Cave Auto-sampler

Author:	*Supervisor:*
Rahul Siddhartha JANGALI	Dr. Shen Hin LIM

February 23, 2018

A thesis
submitted in partial fulfillment
of the requirements for the degree
of
Master of Engineering

THE UNIVERSITY OF
WAIKATO
Te Whare Wānanga o Waikato

Abstract

Researchers at the University of Waikato visit caves on a periodic basis to collect samples of drip water, which over geological time frames form stalagmites and other formations. However, most caves are remotely located, which makes it an arduous task to reach the study site and is economically infeasible for frequent visits. This thesis presents the development of an autonomous cave autosampler deployable inside caves for extended periods to collect drip water and log the atmospheric conditions such as temperature, pressure, and carbon dioxide.

The auto-sampler is developed from two various aspects. Firstly, the collection and storage mechanism. Secondly, control systems, power management, and data logging. The sample drip water is collected from a positioned funnel and stored in an airtight 20ml sample vial through a syringe needle; a new sample vial comes into position on a weekly basis. A carousel styled turntable is designed to store fifty 20 ml sample vials in the minimum possible size. The sample vial position on the turntable is controlled by a stepper motor to maintain precise position relative to the syringe needle. Acrylic is selected as the primary material to reduce weight. Temperature, pressure and carbon dioxide readings are logged during switching operations. The electronics on the auto-sampler are turned off in an idle state, and the Arduino microcontroller is put into sleep mode to minimize power consumption.

The auto-sampler has been tested in lab conditions extensively for position control, power consumption and overall performance. During position control testing, the sample vials made it to the desired position relative to the syringe position on most of the cycles. There are few minor issue which affects the consistency of the auto-samplers operation such as back play in the linear actuator and flexing of the turntable during piercing of the sample. Based on a 50 week deployment, the auto-sampler has an energy requirement of 113.6Ah.

Acknowledgements

The first person I would like to thank is my supervisor, Dr. Shen Hin Lim, for his guidance, enthusiasm, and patience over the last year.

Additionally, I would like to thank Dr. Adam Hartland who provided funding so that the project could be undertaken.

I would like to thank the technical staff, Gordon Neshausen for being very approachable and guiding me with technical queries and purchasing throughout the project. Pete Higgins, for guiding me with the use of lathe machining equipment, And technical manager Shannon McMurray for CNC milling.

Also, thanks to my friends and family for their continued support and patience, Thomas Andrews, Yousef Alshammri, Ali Abdul Ghani, Casey Nordick, Matthew Peebles, Mike min, Ethan Walkinshaw and Ben Mcguiness and Rachel Priyanka.

Finally, I would like to thank my parents, Geeta Rani and Venkatesh Jangali, for their patience and support.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

Auto-samplers are an autonomous analytical instrument often used to collect samples from remote streams, mountainous watersheds and underground caves for periodical analysis in the laboratories [2]. A typical autosampler consists of a rotating arm, turntable and sampler vials as shown in Figure 1.1. This research project was aimed to design and develop a purpose-built deployable autonomous water sampler which should be robust, portable and reliable capable of collecting a continuous series of water samples from cave drip point. A field trip to waipuna cave was carried out to explore the environment of the caves.

Figure 1.1: *A portable auto sampler F9-R from GE Health care [1].*

1.2 Problem Statement

Geochemistry department at the University of Waikato is currently involved in studying climate change, which involves taking drip water samples from speleothems inside the cave. Ideally, researchers would like to have four samples of drip water from each month.

Frequent trips to the sampling site are impractical as cave locations range from Waitomo in the north-island to Mount Arthur in the south island, especially during winter periods most of these caves also remain inaccessible. Waipuna cave is the closest accessible site, which is a few minutes drive from the research site, and then up to 30 minutess of trekking and caving to the drip point.

Current collection method of drip water is entirely manual as shown in Figure 2; the sample containers are placed under the drip point to be filled and replaced on the next trip to the site. This method can be very inefficient as only one container can be filled in one month. Therefore, a solution must be designed to increase the number of collected samples without frequent trips.

Figure 1.2: *Sample vials placed under the drip point .*

1.3 Current method of collecting samples - Waipuna cave site visit

The Waipuna cave is 1.5-hour drive from the university, half an hour trek up the mountains and another half an hour of caving to reach the sampling site, see Figure 1.3.

Figure 1.3: *Passage inside the Waipuna cave.*

The cave opening has a steep drop which required careful planning to get down into the cave. The drip point inside the cave is about 20 minutes of walk from the entrance of the cave. The trail consists of narrow passages and obstacles which would require crawling and slippery elevations to climb up to the drip point.

The research team maximizes the number of samples they can collect during each trip, see Figure 1.4. The research team also measures the CO2 levels, the Ph of the dripping water, electrical conductivity and temperature of the dripping water. All these readings are measured manually by using different probes upon arrival at the sampling site.

Before the cave visit, it was assumed that the auto-sampler would be placed directly under the drip point. From the visit, it was found that the terrain is uneven and doesn't

Figure 1.4: *Science Research team collecting samples at the site.*

comfortably accommodate the setting of the machine. The walkway to the drip point is usually a suitable location within 0.5 m of the drip site where the auto-sampler can be placed.

1.4 Project Specifications

A portable and reliable instrument capable of collecting a continuous series of water samples from either a cave drip point or rain gauge, and maintain them in a sealed state (i.e., air-tight) throughout deployment without human intervention.

The samples should be contained in 20 ml PTFE (Teflon) vials, which remain sealed following sample collection. All components coming into contact with the water should be made of PTFE and be removable to facilitate (acid) cleaning. Contamination of the collected water with metals or other elements should be minimized.

Essential:

- Collect water samples and keep sealed airtight in PTFE 20 mL vials
- Low energy requirements
- Should weigh less than 15 kgs.
- Reliable functioning in a temperature range from +1C to 45C
- Records the air temperature and air pressure through time
- Records the time a vial is in the collection position (from when to when was water collected in each vial)
- Maximises the number of samples that can be collected (at least four samples per month if programmed for 1-year collecting)
- Functionality to select a new vial on a pre-programmed basis
- Programmable in situ (in the cave) via a watertight cursor or similar. Necessary program steps include (non-exhaustively): set starting time, set delay (wait in standby until a certain date).
- Continuous logging and data download with easy-to-use USB interface for wifi PC connection

- Low cost of manufacture
- Elegant and simple design to allow assembly by user – i.e., user can be shipped the autosampler for self-assembly
- Modular design to add additional logging devices (e.g., CO2 meters, etc.).

Ideal:

- Modular design allowing addition of sensors on an as-needed basis (e.g., customer can buy the basic model then upgrade to add sensors as needed, plug and play)
- Unit monitors cave air temperature, pressure and CO_2

Chapter 2

Literature review

As a part of this project, a relevant study on deployable auto-sampler was undertaken. This chapter outlines the information on existing deployable auto-samplers pertinent to this project: mechanical design, control system, and electronics.

2.1 Automatic Water Sampler – Intermountain Forest and Range Experiment Station

A portable water sampler was developed by USDA Forest service to collect water samples from mountainous watersheds [2]. The sampler consisted of a 6-volt 'hot-shot' battery and 1.5-volt telephone battery, which could provide enough to power to operate for six months without any human intervention. Figure 2.1 shows the general arrangement of this sampler [2].

This portable autosampler can be backpacked into remote areas with an approximate weight of 50 pounds (23 kg). This sampler can house 16 samples of 100ml bottles to automatically collect at pre-set intervals ranging from 2 to 24 hours.

Description of Sampler: The sampler consists of a turntable, which can hold up to 16 100 ml sample bottles, an electromechanical control unit, a water supply mechanism, and two batteries, all of these components are enclosed in an aluminum case, see Figure 3.

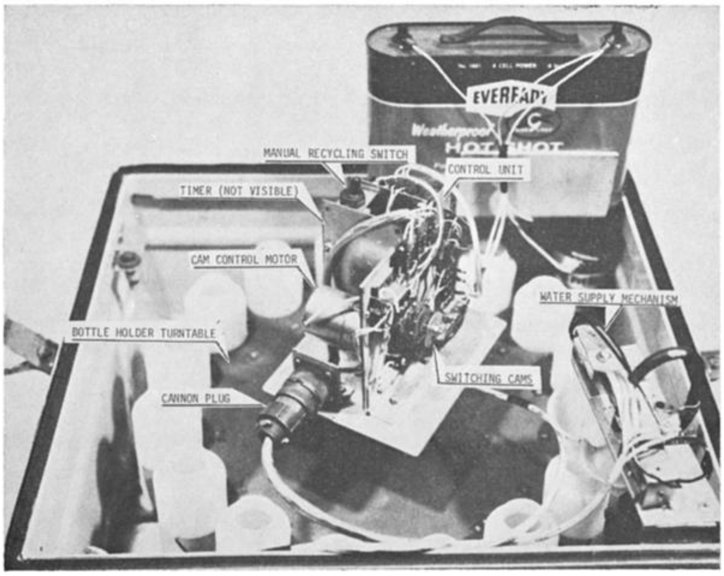


Figure 2.1: *View of sampler showing arrangement of components in bottom half of case. Top of case serves only as a cover [2].*

The primary task of the turntable is to hold and position the sample bottles under the water supply mechanism. The turntable is rotated by a 6-volt dc gear motor assembly [2].

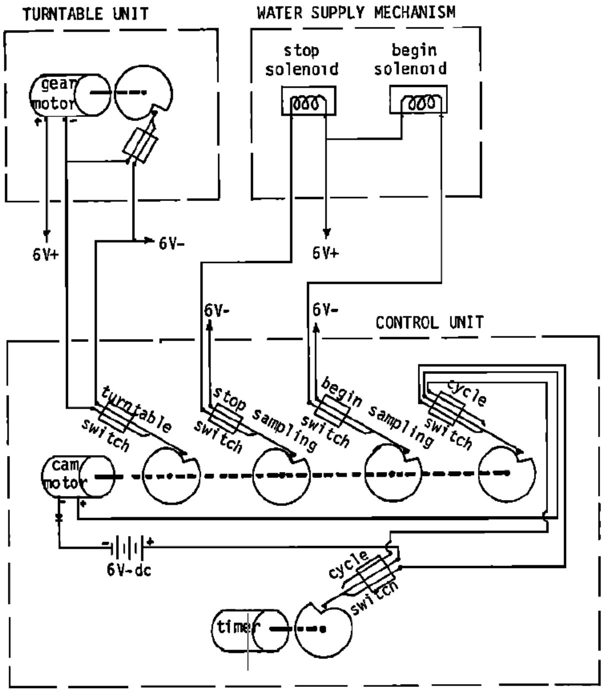


Figure 2.2: *Water sampler wiring diagram [2].*

The control unit operates as the heart of the sampler and it contains a 7.5-volt dc clock timer. At pre-set intervals, the timer actuates the cam motor that turns the three control cams, see Figure 2.1. The wiring arrangement as shown in Figure 2.2 illustrates the relationship of the various components and switches [2].

The water supply mechanism for this sampler is made of a flexible tube that carries water from the stream to the sampler by gravity flow. Two solenoids are used to position the tube, the tube is over the bypass funnel and water returns to the stream, and in the second on the tube is over the fill funnel for the water to enter the sampler bottle. This method of positioning helps in reducing power drain by having the solenoids on only intermittently rather than throughout the cycle [2]. This auto-sampler would not be a suitable option as the sample water stored is open to the atmosphere which would contaminate the collected samples.

2.2 A Passive-Discrete Water Sampler for Monitoring Seepage

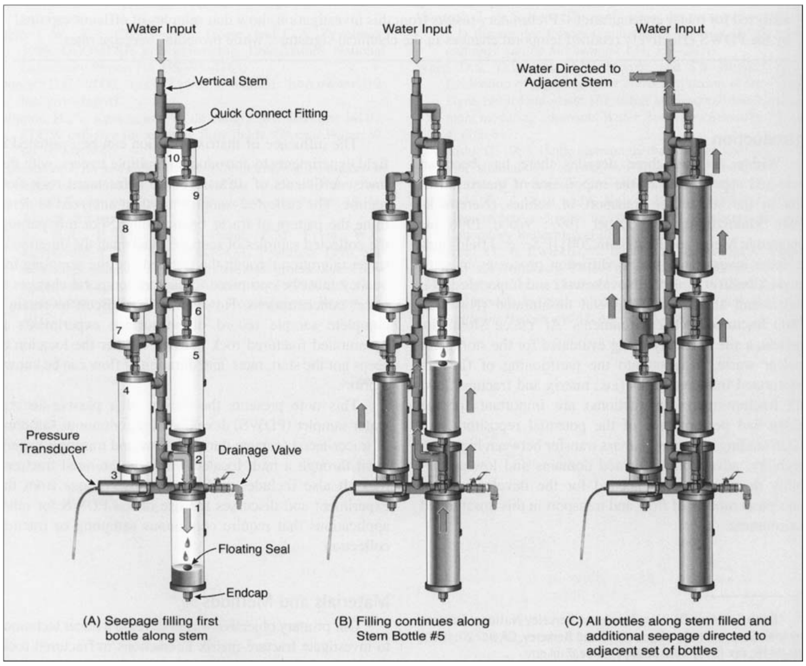


Figure 2.3: *Design and components of the Passive discrete water sampler [3].*

The passive discrete water sampler (PDWS) was built to investigate flow and seepage through a 20m vertical section fault located in Yucca Mountain, Nevada, USA. The sampler has a vertical stem onto which a series of sampling bottles are attached forming a tree-like structure, see Figure 5. Water travels down from the top of the stem and then begins to rise until the water reaches the inlet to the lowest bottle, from where it is diverted into the first sampler bottle (Figure 2.3a). Once the first bottle is filled, the additional water fills the vertical stem until the water reaches the inlet to the second bottle (Figure 2.3b) and so on until the entire series of bottles along the stem are filled (Figure 2.3c). This sampler can be modified to suit the required number of sample bottles if a large number of samples are needed from a single site location, additional number of sample bottles can be attached to the stem [3].

The sample also consists of a pressure transducer which records the height of water at any given moment along the length of the stem; the transducer is connected to a programmable data logger such that the frequency of measurements can be controlled [3].

Drip rates during collection of a particular sample is determined from the sample volume, and also the time to collect the sample which is when the transducer shows constant pressure [3].

The vertical stem in this sampler which is used to convey water to the sample bottles is made from schedule 80 PVC (0.25" ID). The sampling bottles are attached to the stem in a spiral pattern, the number of bottles needed is based on the required number of samples[3].

The sample bottles are cylindrical and are made of transparent PVC, with identical caps attached to the two ends of this cylinder. These sample bottles are connected to the stem using swage-lock fittings for easy removal of the water sampler [3]. This auto-sampler could can be used in cave conditions only where the drip points are high, as the tree like structure can get tall for 50 samples of 20ml. Drip point height varies inside the cave from high to low. Rodents such as bats live in some of the caves, a tall tree like structure could pose a stablity issue for the autosampler.

2.3 A Low-cost composite water sampler for drip and stream flow

This design project aimed to investigate the performance of a proportional flow sampler that collects percolate samples from monolith lysimeters located in North Appalachian, Ohio, USA.

The sampler is designed to direct drip or flow water onto a rotating surface powered by a 1-2 RPM clock motor, see Figure 2.4. A raised dimple slot on the rotating surface made of brass tube collects a small percentage of flow. The brass tube is made of 1.7cmOD and 0.04cm wall thickness formed to resemble a rectangle using a mandrel. The center of the raised slot is 3.2cm from the shaft with a width of 0.4cm, which was able to collect approximately two % of the flow, see the Top view of Figure 2.4. The top edge of the slot was also sharpened to form a knife edge so that any discharge not entering not entering the slot and falling on the rotating surface can be discarded.

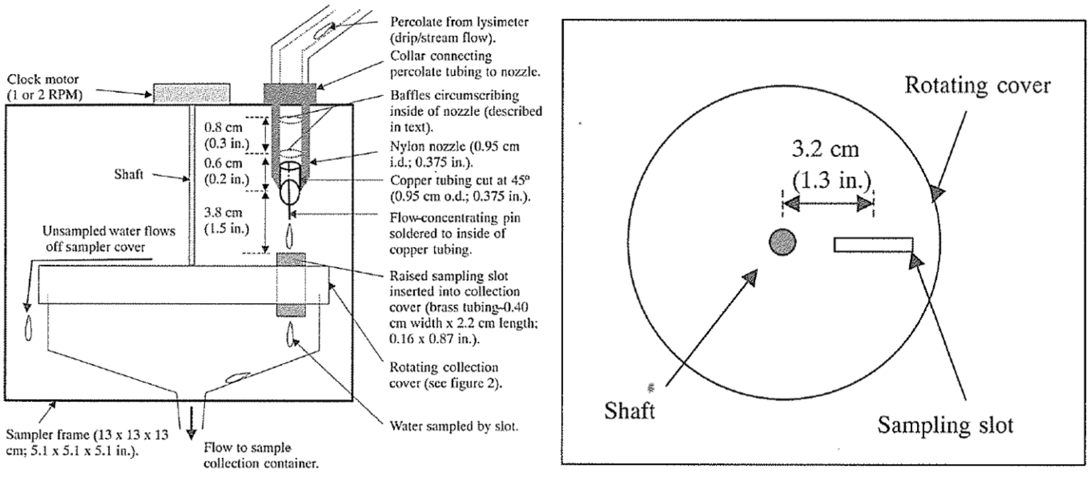


Figure 2.4: *From left to right: Modified rotating-slot smapler design (side view), Top view of rotating collection cover [4].*

A nozzle was made to guide the drip flow to the sampling slot. The nozzle is formed of nylon nipple with 0.95cm ID, internal baffles in the nozzle perpendicular to nozzle discharge, and a pin attached parallel to the nozzle discharge to guide stream flow to the rotating sampler slot. The pin and the baffle plates help in creating a concentrated flow towards a single point which helps in reducing variability between trials. The baffles are made from flexible plastic (Tygon tubing), cut to a size of 0.16cm x 0.16cm with a length of 0.95cm. The purpose of using baffles is to create a random flow to increase sample collection variability between trials [4].

2.4 Automatic sampler for atomic absorption and automatic colorimetric analysis

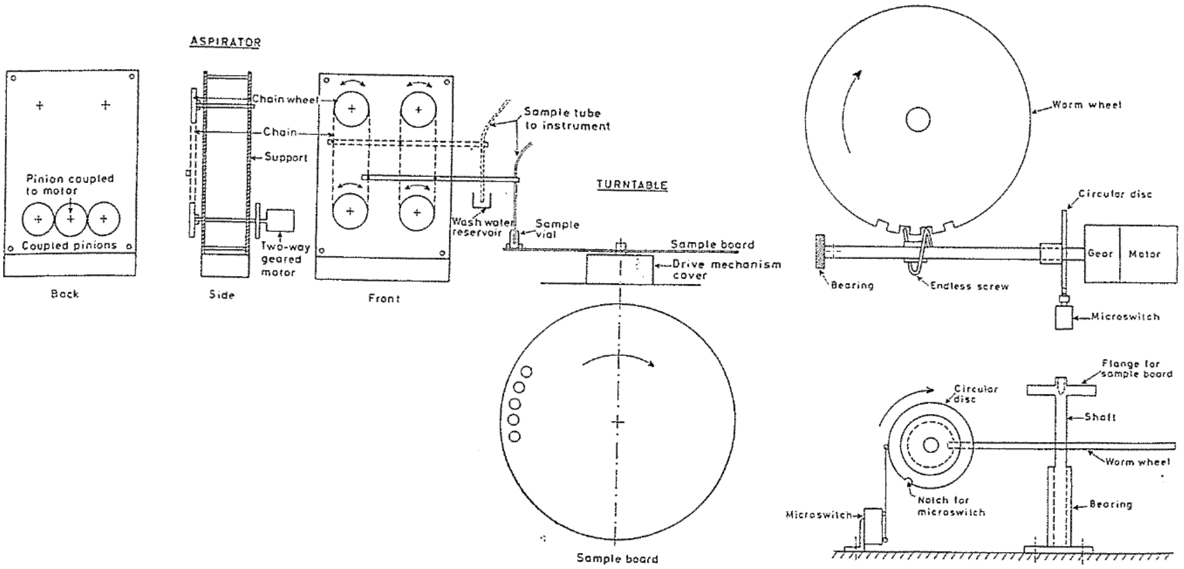


Figure 2.5: From left to right: Schematic diagram of sampler, schematic diagram of turn table [5].

An autosampler was designed and constructed utilizing the benefits of automated instrumentation where a large number of water samples are required; this autosampler is intended to operate in laboratory conditions. Figure 2.5 shows the schematic diagram of the sampler. It consists of an aspirator, turntable and a control unit. The sampling tubes are conveyed from the wash water reservoir to the sample vials positioned on the sample board with the help of an aspirator. The aspirator gives a signal to the turntable between each sample, which then places the next sample vial for aspiration [5].

Two electrical timers and a system of microswitches control the aspirator. Components from the FAC construction kits are used to assemble the aspirator which is driven by a two-way worm geared Heidolph motor. The vertical movement of the sampling hook is varied by the position of two micro switches on the support, and by the distance between the two sets of chain wheels. Time intervals depend on of two electrical timers for washing and sampling. Each timer has its current line to the chain motor, which is coupled in such a way that the chain can be rotated in both directions, on for each timer. The current leading to the motor from each timer passes two micro switches situated at the end position of the sample hook rod. This rod activates the microswitches so that the motor stops when the hook is in its correct position [5].

The turntable as shown in the right of Figure 2.5 consists of a sample board and a drive mechanism. The glass vials are held by the sample board, which uses a worm wheel mechanism driven by a geared motor(Philips Synchronous motor). The gear ratio is chosen so that the sample board is moved 1/40 (9°) of its circumference for each turn of the motor shaft [5].

When the sample board is in the rest position, the microswitch closes the conduit of the motor. The lever of this microswitch is then positioned in the notch of a circular disc mounted directly to the motor shaft. When the microswitch is short-circuited for a fraction of a second, the motor drives the disc a sufficient distance to lift the lever out of the notch. The motor is activated again and moves one revolution and stops still when the bar falls into the groove, giving 1/40 rotation for the next vial to be in position for sampling. A microswitch provides a start with impulse for this movement, mounted so that it is suppressed for a short time when sampling hook rod is moved from sampling to washing position [5].

The body of the aspirator is made from corrosion resistant aluminum of 2.5mm wall thickness, the sample board is made of 6mm PVC sheet, and the drive mechanism of the turntable is covered by an inverted aluminum casserole [5].

2.5 Current Automatic Sampler - Ruhr University Bochum

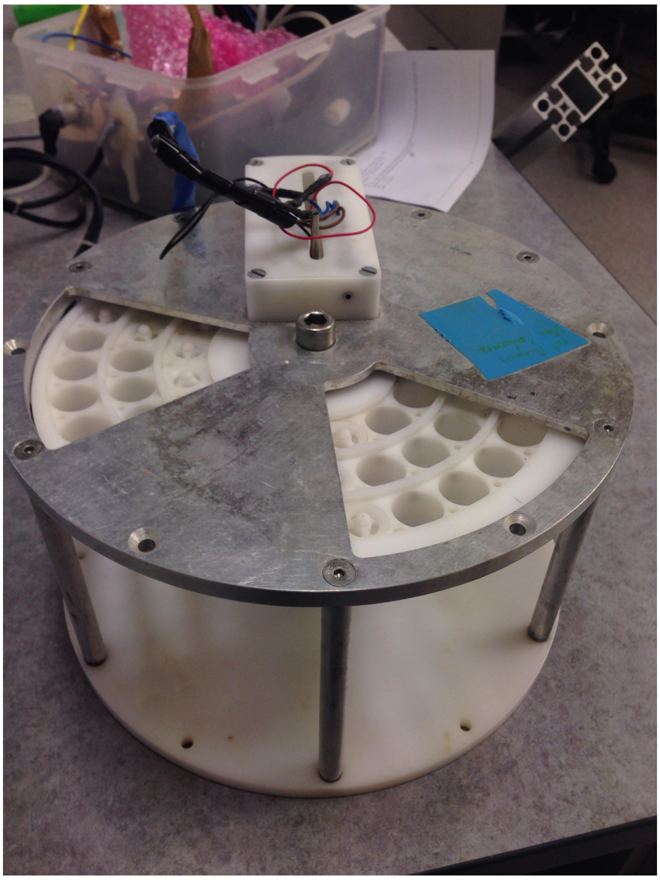


Figure 2.6: *Auto-Sampler imported from Germany for trials in the local Caves*

.

An autosampler from Ruhr University in Germany was requested by Dr. Adam.H in Faculty of Science, the University of Waikato to conduct trials in the caves, see Figure 2.6. The autosampler consists of a turntable, micro-controller, sample test vials, DC motor, a tube for the drip to flow in the vials and a needle made from stainless steel to deliver the water into the vials. This autosampler is portable which can comfortably fit into the backpack and has an efficient design to reduce power consumption. This autosampler uses a passive positioning system to fill the test vials which would require no additional motors to move the needle into the position of the vials. The needle follows a spiral pattern as the turntable rotates and automatically moves in rectangular slot as the turntable rotates.

Though this sample has excellent design features, it was never designed to test drip water from the caves. The primary purpose was to collect samples in any general environment, and it does not have any additional sensors to monitor the required tasks specified in the design brief. The cap on test vial is spring loaded which is made of stainless steel and also the needle which delivers drip into the vials is made of stainless steel which is a contaminant and not ideal for the required tests to be carried out from the dripping water.

Field trials conducted on this sampler have been unsuccessful so far, and the samples obtained were unusable, from close inspection it was found that the springs have corroded (see Figure 2.7) and the autosampler stopped functioning during trials. Quick check with the multimeter revealed faults in the electronic circuit, from further investigation it was found that some of the components in the circuit have corroded.

Figure 2.7: *Corroded spring on the vial cap.*

This autosampler reaffirms the point that the components coming into contact with water should be minimized or eliminated if possible. Also, the electric circuit used in the autosampler should be made waterproof as the environment in the cave is damp and humid.

2.6 Conclusion

There are autosamplers which are designed to collect drip and stream water in other locations such as lakes, rivers, forests, and watersheds. These auto-samplers would be unsuitable to collect drip samples from the cave as the water coming into contact with the sampler components should be chemically inert, and most of the samplers except for the PDWS sampler do not have the capabilities to store sample water in an airtight container. The previously developed samplers use metal in their collection mechanism which is a contaminant, and neither of the already established samplers has the capabilities to monitor cave conditions and log data such as CO2 levels, temperature, and pressure.

Most of the autosamplers developed previously by Robert .D et al. in [2] and L.Berglind et al. in [5] have similar components such as a turntable, DC motor, sample holder for vials and a microcontroller. The collection mechanism in autosampler developed by Salve et al. in [3] is unique and innovative, this type of sampler would need minimum power in comparison with the traditional samplers. This sampler would be a potential concept to consider if it can be placed directly under the drip point after the cave visit the chances of setting the sampler directly under the drip point are slim due to uneven and slippery surfaces and also confined spaces. The rotating autosampler developed by R.W. Malone et al. in [4] will be unsuitable to collect drip water. Its purpose was to collect samples from continuous flow, if placed under the drip the chances of collection of any sample would be quite low as the flow from drip point in the cave is variable.

The waipuna cave site visit reaffirmed the need for the autosampler to be lightweight (less than 15 kgs as per brief) and portable which can fit inside a standard backpack. Anything over the weight and size would be a concern for safety. Cave visits are physically challenging and would require a reasonable level of fitness. Also, the cave has a steep drop, narrow passages, slippery and uneven terrains inside the cave. Having a lightweight and a portable sampler will help the researchers trek to the cave a much safer trip.

From inspecting current auto-sampler imported from Germany showed the need to use chemically inert materials is an absolute necessity, use of metals should be avoided as it is a contaminant for sample testing and also to prevent corrosion issues as the sampler be operating in a damp environment.

Chapter 3

Design and Development of Cave autosampler

This chapter outlines the technical design requirements to be considered, hardware architecture and design methodology for the development of the cave auto-sampler.

3.1 Design requirements and Features

Before laying out potential concepts for the new auto-sampler, the first step was to define the breakdown of essential characteristics and functionalities of the device. Multiple client meetings were to discuss design specification and processes that would involve in achieving them. The next step in the design process was to conduct background research on existing auto-samplers. Once enough information had been acquired, a list of key design features for the auto-sampler was identified and described.

The cave autosampler will be mainly used as science equipment to collect drip water from stalactites and log environmental conditions inside the cave, Geochemistry department at the University of Waikato will be the main users of this equipment. The next step in the process is to know what specific characteristics should the machine have to meet client requirements. Since there aren't many deployable auto-samplers suitable for cave applications, design requirements can only be identified by involving the clients through the design process and exploring the operating environment for the machine.

As a part of the design process, an important step is to look into the some of the problems that were faced using the current autosampler which was developed by RUHR University to which the scientists at the University of Waikato had access. The major issues with this autosampler were the contamination of samples as the materials used weren't chemically inert, the electronics in the machine were damaged due to humidity as this was not intended to operate in cave environments. The below table 3.1 summarises the features the clients would like to have in the new autosampler.

Table 3.1: Summary of essential design requirements obtained from the client

Key design requirements in the new autosampler
Water samples should be kept airtight
Easy to programme on site
Maximise the number of samples that can be collected
Simple design for assembly on site
Data-logging capabilities to monitor cave environmental conditions
Low cost of manufacture

Another critical step in the design process is to investigate and analyze existing autosamplers. Researching existing products gave the opportunity to learn about the existing deployable autosamplers and improve on their existing features to develop a machine suitable for cave deployments. Table 3.2 lists the existing autosamplers along with key features of each device.

Table 3.2: Summary of key features in existing autosamplers

Autosampler	**Key features**
USDA Forest autosampler[2]	Can be carried in a back pack, Can operate upto 6 months, 22kg of weight, Can store 16 samples, carousel style storage.
PDWS water sampler[3]	Tree like structure storage, Capability to maximise no of samples stored, Low power requirements to traditional samplers, can record height of water at any given moment.
Composite water sampler	Not suitable for cave applications, Creates random flow to increase sample collection.
Autosampler for colorimetric analysis	Large no of storage samples, designed to operate in lab, Use of bulky mechanisms to operate.

The main goal of this phase is to utilize the client requirements, the knowledge from existing autosamplers and a list of key design features to generate an initial design concept. One of the most desired features in the design specification is the ability to store samples in an airtight container thereby minimizing evaporation during storage. This functionality is not found in existing auto-samplers. From consultation with the client, samples could be considered airtight if evaporation during and after collection of the sample is minimized or eliminated. Another desired feature for the autosampler is to maximize sample storage without compromising on the portability of the device. Clients would like to have a sample capacity of 45-50 vials, enabling an average of one sample per week based on one-year deployment. The ability to log atmospheric conditions inside the cave such as carbon dioxide, temperature and pressure is also a desired feature. Furthermore, to address the issue of programming on site, another design feature would be to integrate an off the shelf microcontroller which would allow the user to change or upload new programmes without prior programming experience. Lastly, the autosampler should have a simple modular design to allow for easy in-situ assembly. The table 3.3 outlines the agreed design specification.

Table 3.3: Summary of key design requirements

Desired feature	**Description**
Sample storage	Have the ability to collect and store water in an airtight container.
Programme	An easy to use off the shelf microcontroller for insitu programming.
No of samples	Store upto 50 sample vials.
Assembly	Modular design for ease of assembly.
Data logging	Monitor cave atmospheric conditions.

3.2 Concept Modeling

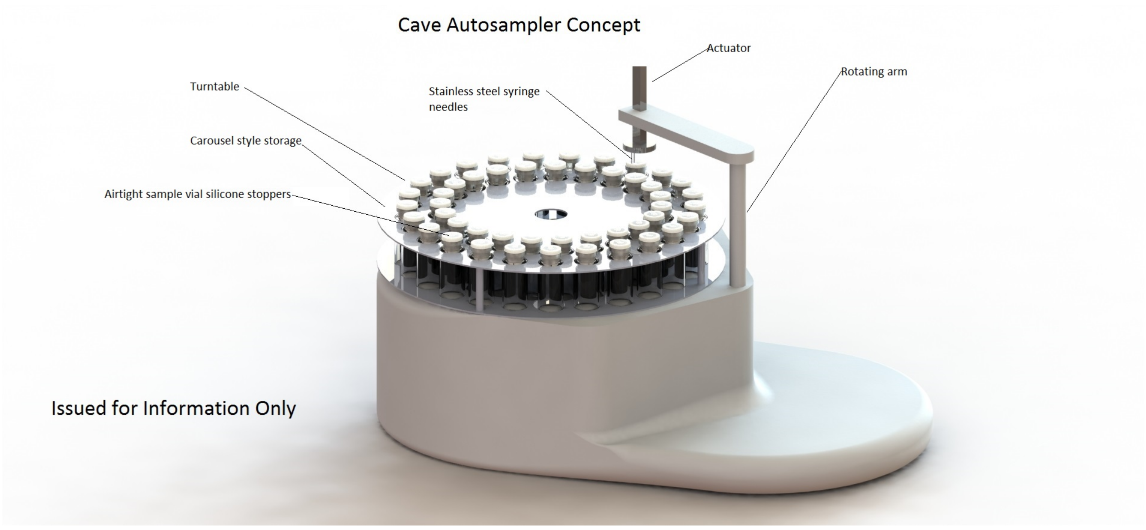


Figure 3.1: *Conceptual model of a deployable cave autosampler.*

Based on the design features, literature review and resources available, several concepts were generated and evaluated until a final design concept was selected. The selected concept design can be seen in Figure 3.1. The proposed concept consists of a traditional carousel styled sample storage to maximize the number of samples, and sample vials made out of polypropylene and silicone caps to ensure airtightness of the sample. The proposed device would have storage space for 50 vials, 25 on the outside row and 25 on the inside row. The transition between the inside and outside sample vial rows is accomplished through the integration of a rotating arm, and the arm would stay on the outside row until all the samples are filled and rotated inwards to fill the remaining sample vials. Turntable and the rotating arm will be motorized to created the desired motion.

The task of storing drip water in the airtight containers is accomplished with two 18 gauge stainless steel needles; During operation, incoming sample water is directed through one of the needles while air is allowed to escape to the atmosphere through the other. All the electronics on the auto-sampler such as motors and sensors will be controlled using an Arduino microcontroller as this is readily available and offers a simple interface.

3.3 Hardware Architecture

This section discusses all the hardware needed for the auto-sampler such as stepper motor, linear actuator, airtight sample vials, cave sensors, data logger and power management. Figure 3.2 summarises all the hardware necessary for this autosampler.

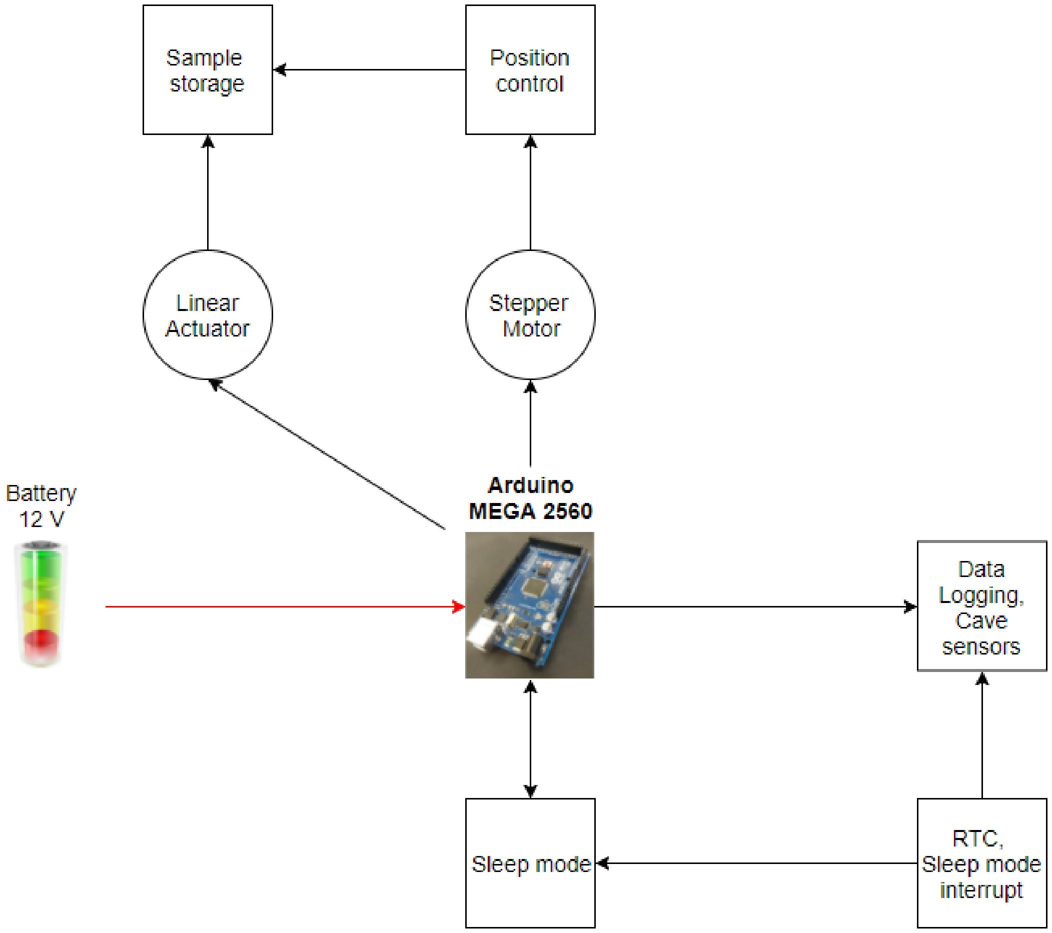


Figure 3.2: *Hardware architecture block diagram.*

3.3.1 Sample vials and Septa

The sample water collected from the caves have to be stored in a chemically inert container to avoid contamination. PTFE sample vials were the first choice but later phased out due to economic reasons, Refer to Appendix D Figure C.1 for PTFE vials pricing. Polypropylene vials were chosen as a replacement due to its low cost and comparable chemical inertness to that of the PTFE vials as per conversation with the client.

A significant requirement for this project is to collect and store water in an airtight container. In analytical chemistry, the two most commonly used septa are made from rubber and silicone. Rubber was phased out as it is a contaminant for the dripping water. Silicone septa were chosen due to its inert chemical nature. Laboratory grade silicone septa were pierced multiple times to test for air tightness and water leaks. A test showed that it had a rubber like behavior and gap closed in everytime and had no water leaks. These silicone seals were readily available from the science store at the University of Waikato and are integrated into the final autosampler.

3.3.2 Stepper motor

Stepper motors act as a central drive for the turntable and the rotating arm. The motors are connected to an Arduino based shield motor driver. The torque requirement for the turntable at full load is 2.472 Nm. The turntable has a higher torque requirement than the rotating arm. Therefore, this was taken as a base figure when selecting an appropriate motor. All the motors were initially tested for position control only with a negligible load. The NEMA 17 geared stepper motor was selected due to its accurate position control, ability to hold position while the power is turned off and a torque rating of 4.7 Nm. Selecting a stepper motor was an iterative process. Few other stepper motor makes such as open loop; hybrid closed-loop stepper motors have been tested for position control which has not met the required specifications. Position control will be further discussed in the following sections.

3.3.3 Linear actuator

The silicone septa need to be pierced with needles to store water in an airtight container. The force required to pierce a silicone septa with needles is 5.89N, and this is achieved through linear motion. An off the shelf linear actuator was chosen to achieve this task. This linear actuator has a stroke length of 100mm and comes with a built-in potentiometer for linear position control. It has force rating up to 1000N which more than required. It has an operating voltage of 12-24V and is waterproof which makes it ideal for cave deployments. Further technical details for the actuator can be found in Appendix A.3.

3.3.4 Micro-controller

Ease of programming, low power consumption, capabilities to upload and change programme on site and capabilities to add additional electronic sensors in the future, are some of the significant client requirements to consider when looking for a suitable micro-controller in this project. These requirements led to the selection of Arduino MEGA 2560.

The Arduino MEGA is based on the ATmega 2560 microcontroller. It has 54 digital I/O pins, 16 analog inputs, 4 UARTs, 16MHz crystal oscillator, USB connection, power jack, ICSP header and a reset button and a recommended input voltage of 7-12V [9].

3.3.5 Cave monitoring sensors

The autosampler should be able to monitor environmental conditions in the cave such barometric pressure, Temperature, and carbon dioxide. The sensors were chosen based on cost and programming libraries available on Arduino IDE software.

BMP280 sensor is used to monitor barometric pressure with a precision of ±1 hPa absolute and temperature within $\pm 1.0°C$ as per the data sheet.

Measurements of carbon dioxide are accomplished by a high precision analog infrared CO2 sensor from DF robot. It has an active measuring range of 0-5000PPM with an accuracy of $\pm(50ppm + 3\%reading)$.

3.3.6 Data Logger

The auto-sampler should be able to log cave environmental conditions on a periodic basis is one of the main requirements. Ada fruit data logging shield was chosen for this project due to compatibility with Arduino mega and for its well supported online documentation and programming libraries. This shield stores the recorded data in a CSV format onto an onboard SD card. The data logging shield also comes with an onboard RTC which will be used to timestamp all the data recorded and the duration of the sample vials position.

The onboard RTC on the shield has an alarm which will be used to trigger an interrupt to wake the Arduino microcontroller from sleep mode.

3.4 Auto-sampler construction

This section of the thesis deals with the fundamental design decisions and development of the auto-sampler as well as power requirements and software implementation. Although the process is written in linear progression, there was an iteration in the design to find the best compromise of portability of mechanical parts and power saving.

Each section states the final design decision and how it meets the project specification mentioned in Chapter 1.

3.4.1 Turntable Assembly

The turntable is one of the major components of the auto-sampler as it dictates the number of samples that can be stored and the size of the remaining components such as the platform and the rotating arm. A traditional carousel styled turntable was chosen for sample storage due to simplicity of the design and minimize any extra movements in the machine.

The turntable must be sufficient in diameter enough to accommodate the required number of samples to be stored. It must also be portable and modular in design for ease of assembly and transportation to the site. The turntable went through many iterations to accommodate for different position control methods, which will be discussed in the below sections. The final turntable which is used in the autosampler as shown in Figure 3.3, is 315 mm in overall diameter as this was found to be the minimum acceptable size to fit 50 sample vials. This turntable was designed to be used with a solenoid locking mechanism hence the oblong slots on the bottom table.

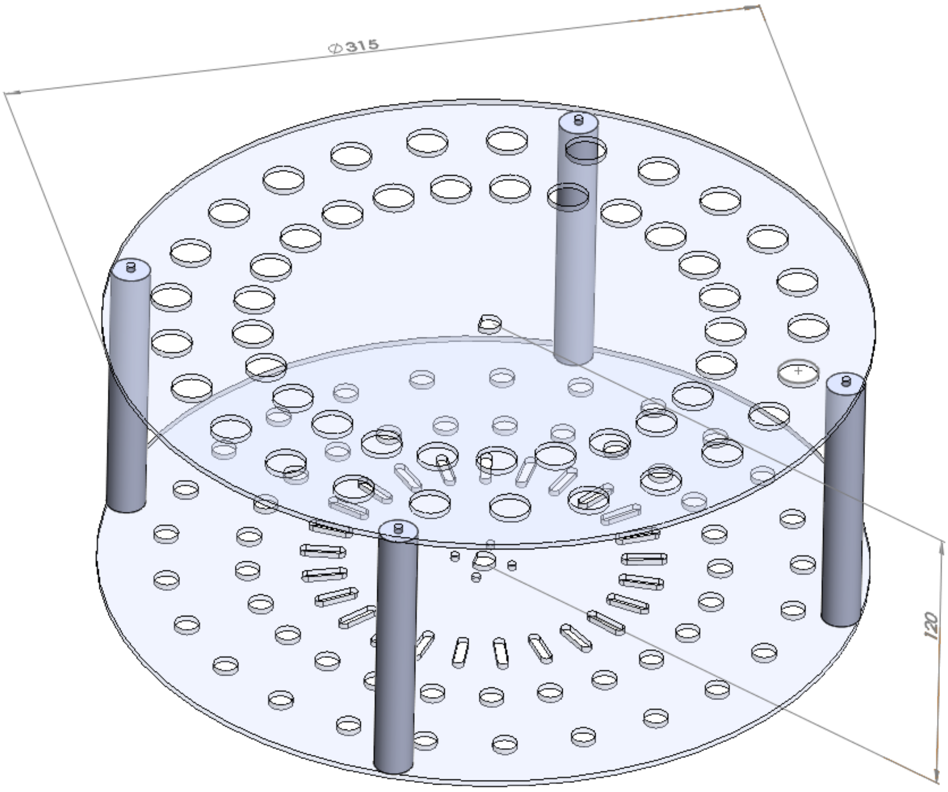


Figure 3.3: *Solidworks model of the final turntable model.*

The top turntable has larger diameter hole of 16mm to hold the sample vials while the bottom table has smaller holes of 10mm for the sample vials to sit, both the turntables are made from 3mm clear acrylic as it was readily available in-house. The table has a total of 50 sample vial slots, 25 slots on the outside and remaining 25 on the inside. The PCD of the inside holes is kept to the smallest optimum number within the manufacturing capabilities of laser cutting; this is a crucial dimension as it dictates the portability of the auto-sampler. The positioning of the outside holes was chosen to be in line with the rotating arm arc to store the dripping water on the outside sample vials as seen in Figure 3.4. The minimum angle of 12.6° (7 steps) between centers of the inside and outside rows was based on the optimisation between the resolution of the NEMA 23 stepper motor and the minimum distance between the oblong slots on the platform for the solenoid locking to ensure position control of the rotating arm.

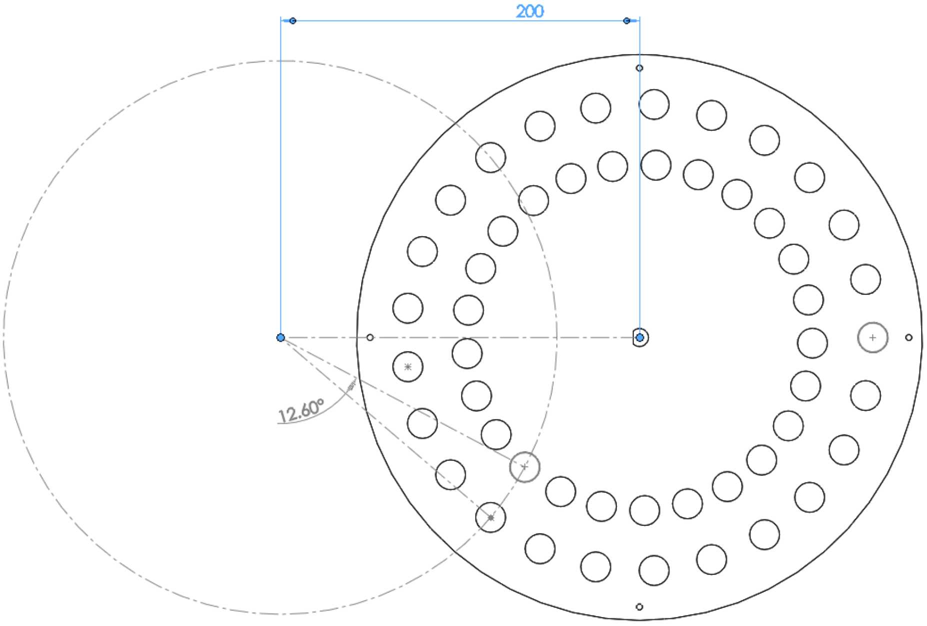


Figure 3.4: *Solidworks geometry of sample slots.*

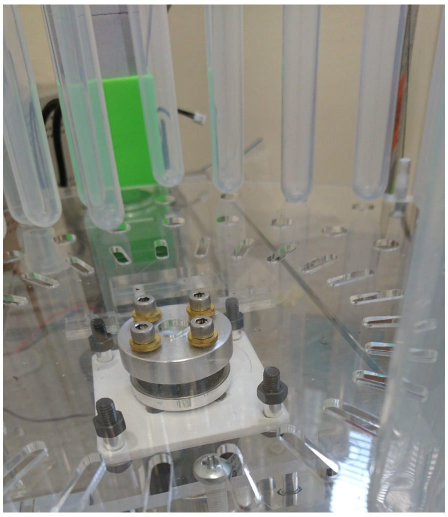


Figure 3.5: *Turntable assembled onto the adaptor with set screws.*

The turntable has a height of 120mm which was sufficient to hold the sample vials in the slots with minimum play, the turntable was tested at lower heights the sample vials were not steady enough for sample delivery. The top and the bottom table are fastened together with four 16mm diameter acrylic rounds. To ensure modularity and ease of assembly, the bottom table is fixed onto an adaptor with set screws which can be easily replaced or removed when necessary as seen in Figure 3.5. This option allows for the user to carry the turntable as a separate unit and assemble it on site. The final render of the Solid-works model can be seen in Figure 3.6.

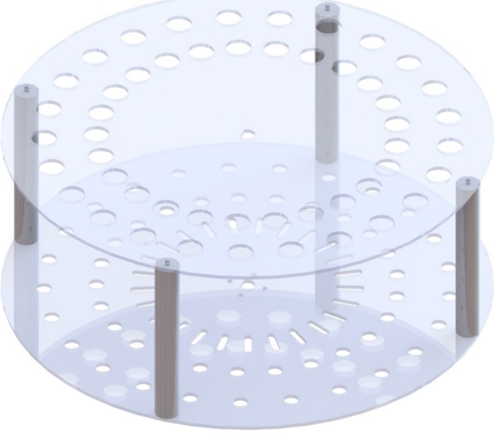
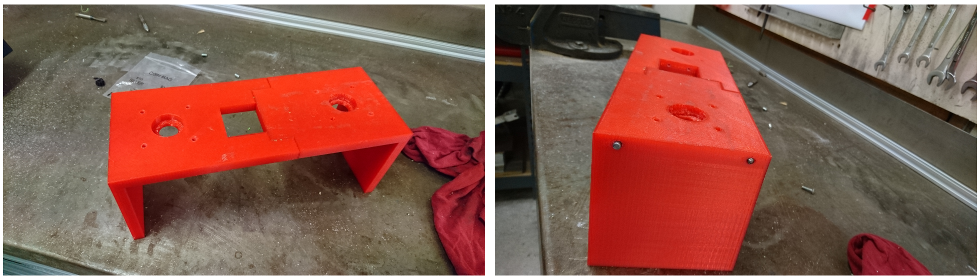


Figure 3.6: *Solidworks render of the final turntable model.*

3.4.2 Platform



(a) 3D Printed platform prototype. *(b) Steel reinforced in the platform.*

Figure 3.7: *3D printed platform.*

After the turntable dimensions were finalized, A platform was designed and built to house all the major components of the auto-sampler. The minimum length of the platform is determined by the center to center distance between the turntable and the rotating arm arc. The width and height of the platform were determined by the size of the initially chosen NEMA23 stepper motor as mentioned in the subsection 3.3.2. The platform went through several iterations using 3D printing as shown in Figure 3.7, this was mainly due to the change in stepper motors through the design process, the final platform is 120mm wide and tall.

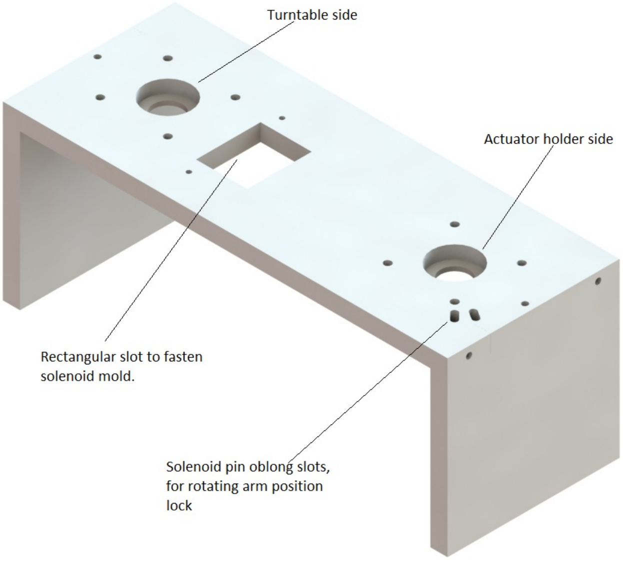


Figure 3.8: *Final platform design solid-works render.*

The final platform is laser cut from 12mm thick acrylic and weighs 826 grams; It is CNC milled for a tight fit to accommodate two 9mm thick deep groove ball bearings to prevent water leak to the stepper motor, Figure 3.8. This material was chosen due to the low cost of manufacture and also due to its relatively high strength to weight ratio. The platform is designed to house two stepper motors, solenoid, micro-switch, sample storage assembly and rotating arm assembly. The dimensions of the platform is a balance between the minimum required measurements to house the significant components and its portability to fit in a backpack. The final platform is 310mm long, 120mm tall and wide.

With all the major dimensions of the platform designed and finalized, FEA analysis was performed in SolidWorks to confirm the platform can withstand the load exerted by components, as can be seen in Figures 3.9 and 3.10. The boundary conditions are based on the real loading scenario. The weight of two stepper motors (5.53N each) is applied at M4 hole slots. The weight of the turntable at full load (15.69N) is applied inside the groove for bearing, silicone septa puncture load (5.89N) and Rotating arm assembly (14.72N) are applied inside the bearing groove to the right-hand side. The results from the FEA analysis showed the maximum induced stress of 493.90 Kpa and a maximum deflection of 0.05mm. Acrylic has a yield strength of 45 Mpa which is higher than the induced stress. Therefore this platform is suitable for use in auto-sampler.

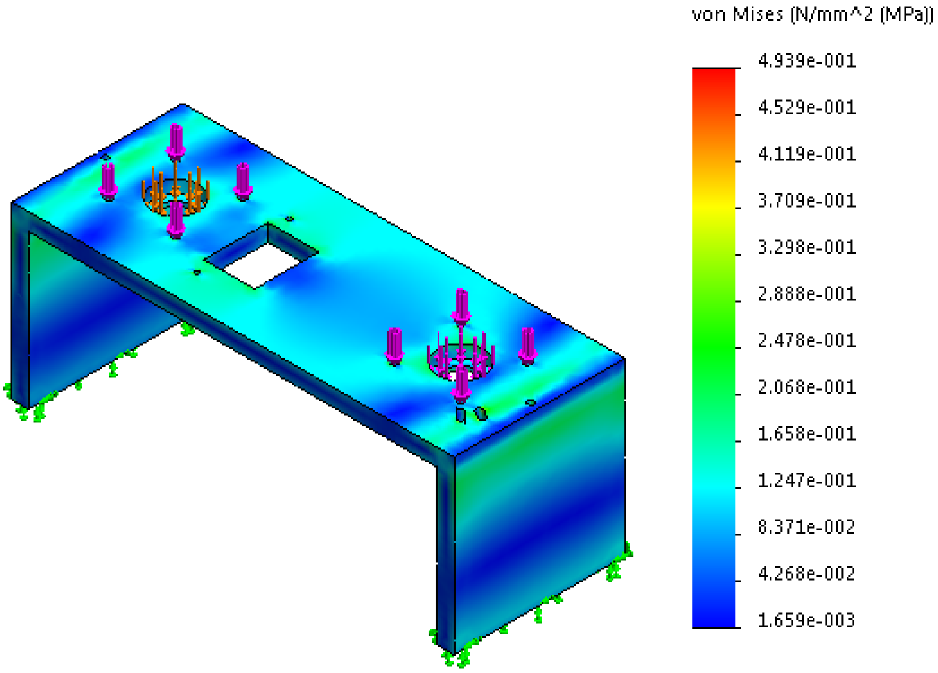


Figure 3.9: FEA stress analysis on the platform.

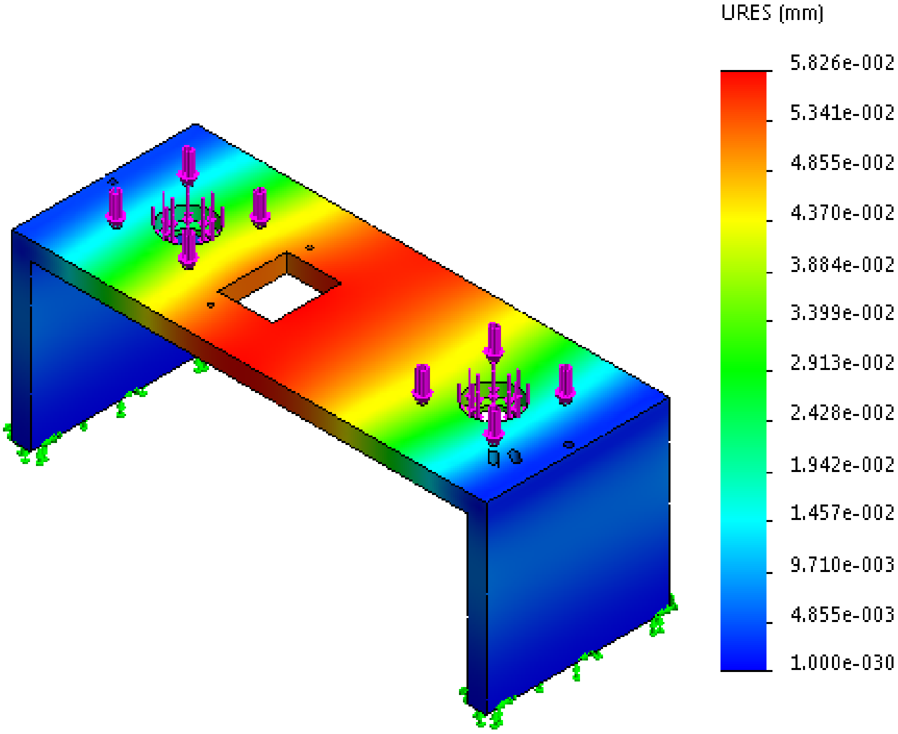


Figure 3.10: FEA deflection analysis on the platform.

3.4.3 Rotating Arm

Rotating arms primary function is to structurally support the delivery of the dripping water to the airtight sample vials and is designed to fit with the linear actuator as discussed in subsection 3.3.3. Load tests were performed on the silicone seals to measure the force required to puncture the seal. The septa needed a weight of 5.89N to puncture with both the needles. The rotating arm is 200 mm long from the point of rotation to the point of piercing point. The first rotating arm Figure 3.12 underwent a few iterations in design and materials. The first arm was designed as a proof of concept to confirm it can pierce the silicone septum on the airtight containers; FEA analysis in SolidWorks showed a maximum induced stress of 0.966 Mpa and a maximum deflection of 0.48mm (Figure 3.11). From the FEA it can be seen that most of the deflection occurs at the point of loading. Physical testing the arm had much higher deflection than obtained from FEA results and eventually bowed after a few test cycles. This variation in FEA and real testing could mainly be attributed to the poor print quality of the 3D printer.

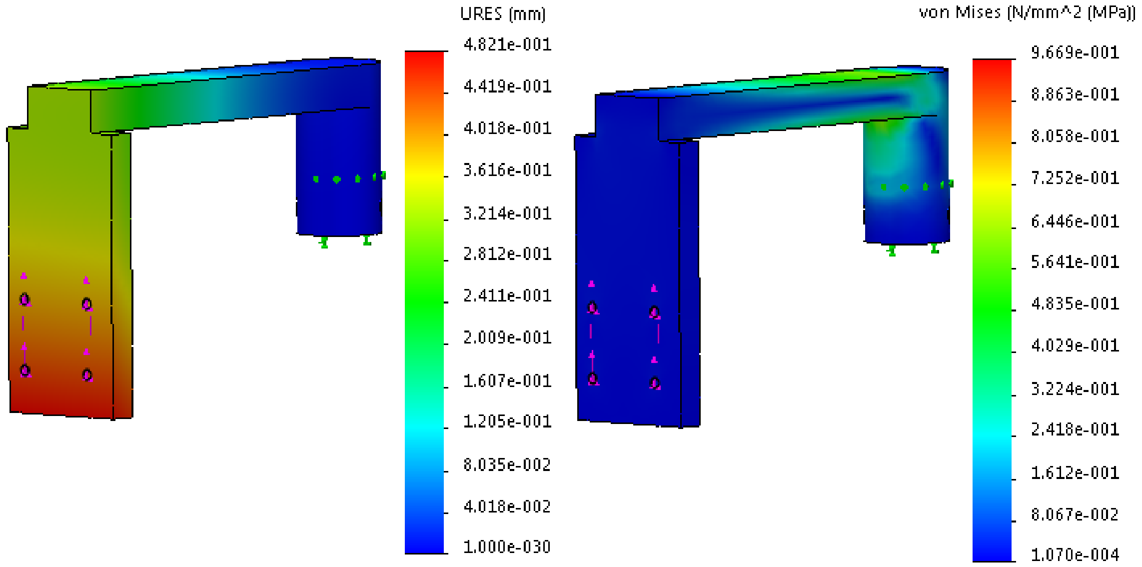


Figure 3.11: *FEA deflection and stress analysis of the first arm.*

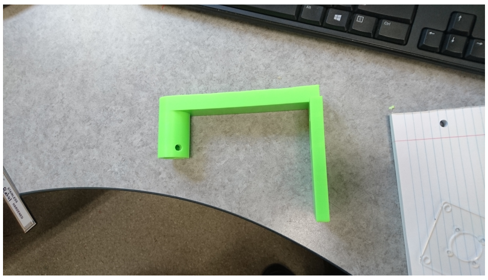


Figure 3.12: *First 3d printed arm ABS Plastic - First iteration.*

The second rotating arm was designed for superior stiffness and strength with extra supports and gussets to stop the deflection at the point of loading. Upon fitting the second arm onto the actuator rod it was found that the 3D printed model's dimensions were smaller with the SolidWorks 3D model, this was due to the parts being printed on a low-resolution 3D printer which had the plastic warp and shrink during print. To resolve this issue, the same design and model was printed in ONYX material on a high-resolution 3D printer which reduced the effect of warping and shrinkage, See Figure 3.14.

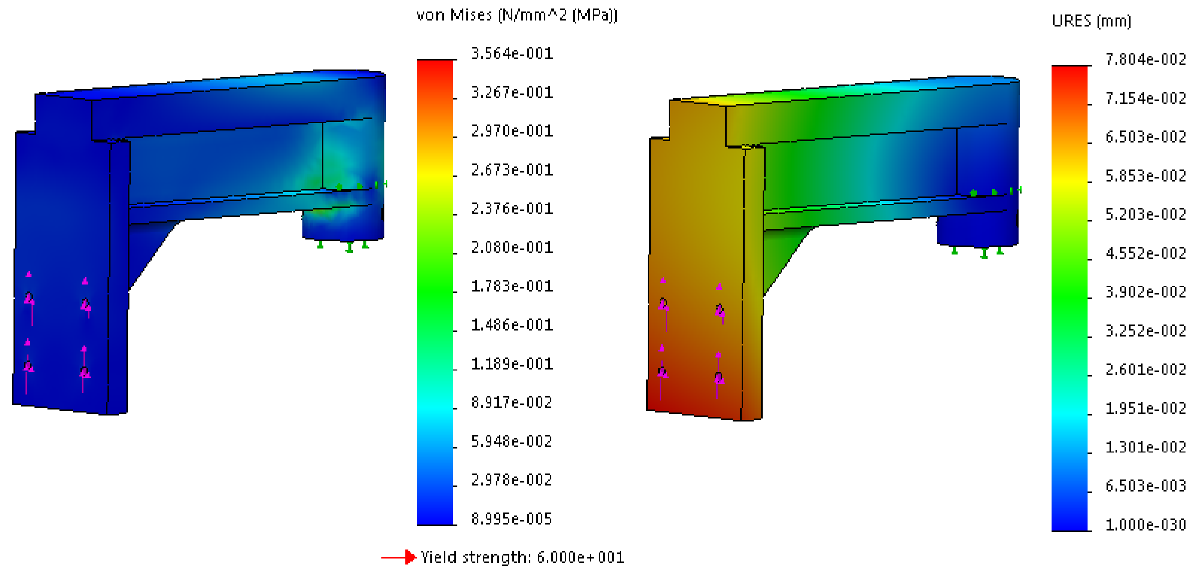


Figure 3.13: *FEA deflection and stress analysis of the second arm.*

After the design of the second arm was finalized, an FEA analysis was carried out on the arm for ONYX material. The boundary condition was similar to the first arm. The results showed that the arm had a maximum induced stress 0.356 Mpa which is much less than the yield strength of 60 Mpa. The arm had a maximum deflection of 0.078mm in FEA. Real testing on the 3D printed arm showed no signs of deflection or flexing, this is part is now press-fitted on the linear actuator rod and fully functional.

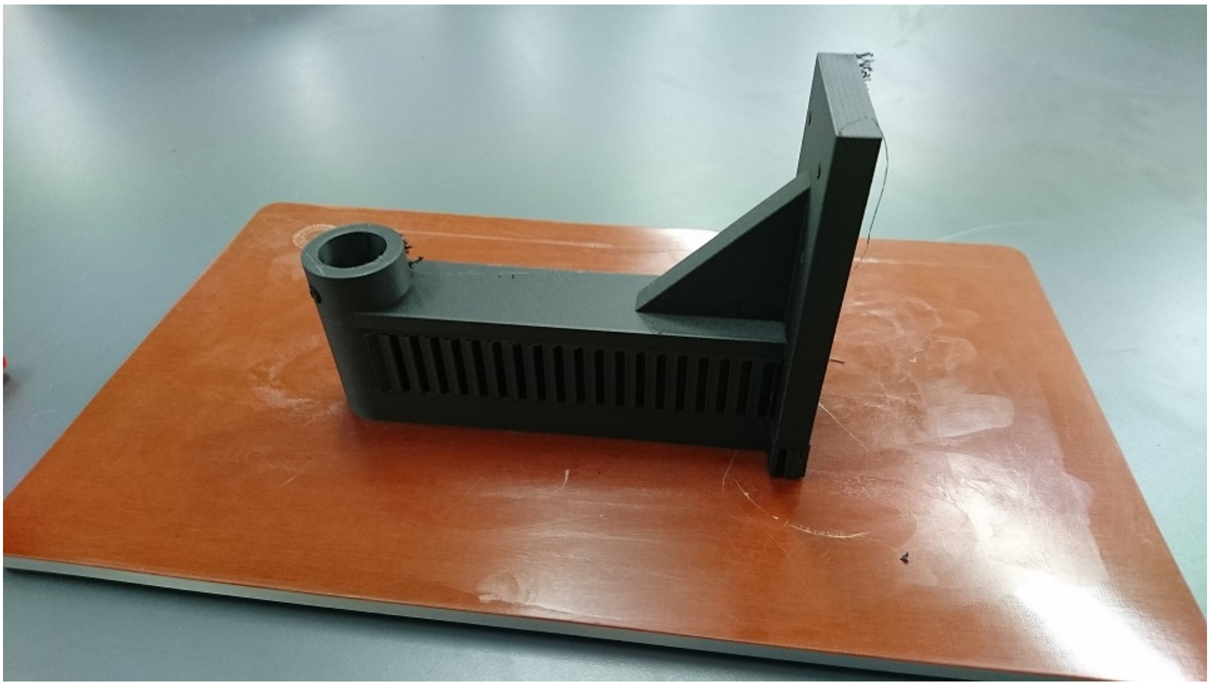


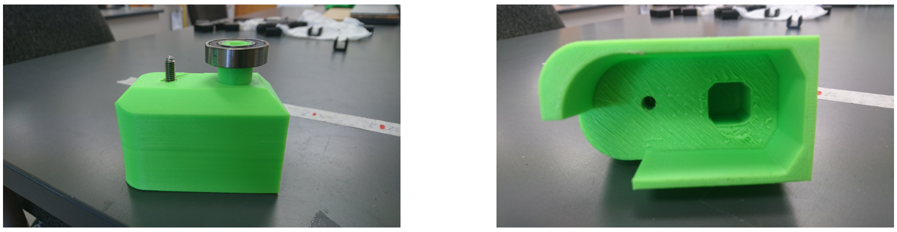
Figure 3.14: *Final 3D printed rotating arm on the print platform.*

3.4.4 Actuator-holder

Figure 3.15: Linear actuator assembled inside the hold.

The main purpose of the actuator-hold is designed for the linear actuator to be placed inside the hold and for positioning of the actuator between the outer and inner rows of sample vials. Actuator-hold was designed to structurally withstand the weight of the actuator and the force exerted from the rotating arm while piercing the septa. The dimensions of the mold are based on the chosen linear actuator as discussed in subsection 3.3.3.

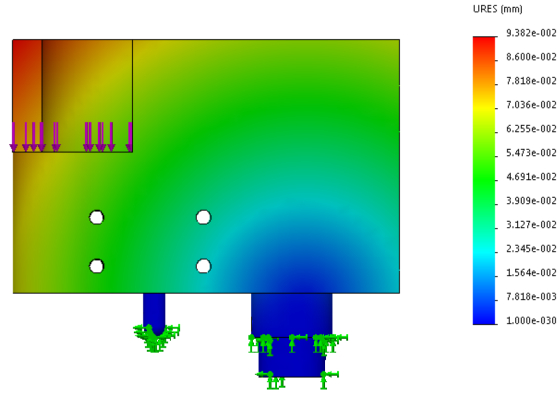
Once the dimensions were obtained for the mold, An FEA analysis was performed in SolidWorks to confirm the load requirements for the mold. The boundary conditions are based on the weight of linear actuator at 14.72N and load required to pierce the silicone septa at 5.89N. The press fit location on the hold is chosen a fixed support, and the ball detent is selected as a roller support. FEA analysis results showed a maximum deflection of 0.09mm and maximum induced stress of 1.241 Mpa which is less than the yield strength of 60 Mpa. From FEA results and real testing, it can be said this hold meets the desired structural requirements and is integrated onto the auto-sampler.



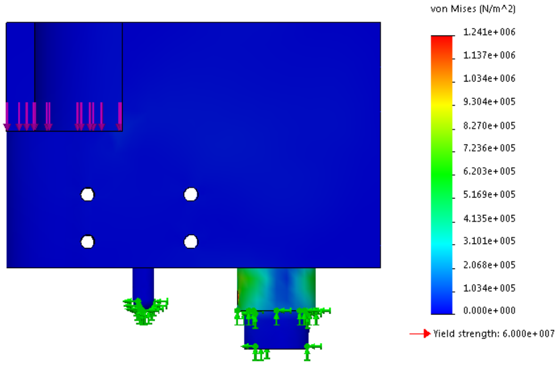
(a) Actuator-hold with bearing and ball detent.

(b) Inside view of the actuator hold.

Figure 3.16: *3D printed model of the Actuator-hold.*



(a) Actuator-hold FEA deflection analysis.

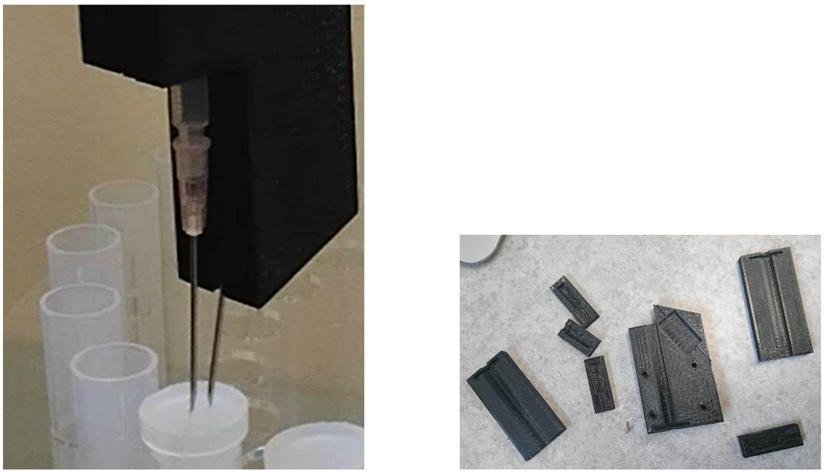


(b) Actuator-hold stress analysis.

Figure 3.17: *3D printed model of the Actuator-hold.*

The final hold is 3D printed out of ONYX plastic and went through a few iterations due to the warping and shrinkage issues with the low-resolution 3D printer, was later made on a high-resolution printer which solved the issue. The hold has been press-fitted into a deep groove ball bearing and is connected to the stepper motor shaft. The ball detent as shown in Figure 3.16a stops the hold from tilting when the arm pierces the airtight sample vials. The hold is designed so that the actuator can be carried as a separate unit and be placed inside the hold upon reaching the sampling site, Figure 3.16b.

3.4.5 Sample water delivery



(a) *Syringe needle buckling during testing.*

(b) *First iterations of syringe moulds.*

Figure 3.18: *3D printed syringe moulds.*

One of the most significant client requirement for this project is to store the dripping water collected from the stalactites into an airtight sample vial. Initial tests were done to check the container for any water leaks and airtightness by piercing the airtight sample vials sealed by silicone septa with 18 gauge stainless steel needles and injecting water through the syringe. The septa were pierced multiple times with various gauge needle size to the find the optimal size. Needle size any less than 18 gauge failed to deliver water into the airtight containers consistently hence this was chosen as the optimum size. The seals were checked for any water leaks after multiple punctures but showed no signs of water leakage and the silicone septa closed the gap every time it was pierced with the chosen needle. This concept was further explored to integrate it into the machine. The test also showed that two needles were necessary for this concept to work since the water has to be stored in an airtight container an additional needle was needed for the air to escape which is trapped inside the airtight container. Air outlet needle was slightly bent inside the mold to minimize the distance between the needles since the septa has a puncture diameter of 6mm and the needles had to be within this diameter, anything over would mean the air would not escape out of the container which would stop the incoming water. The comparison of difference needle configurations is shown in Figure 3.19.

Figure 3.19: *From left to right - Silicone septa pierced with one straight and one bent needle, Silicone septa pierced with straingt needles.*

A two-part styled mold was designed to house a syringe and two 18 gauge needles, one needle for the sample water delivery and the other needle for the air to escape out of the airtight container. This design went through many iterations due to the problems with the buckling of the syringe needle and fitting of the syringe and the needles inside the mold, see Figure 3.18. Increasing the length of the mold reduced the buckling issue, printing the mold on a higher resolution printer helped with the fitting of the syringe and needles inside the mold. This part is now fully functional and has met the client requirements to store samples in an airtight container and keeping the sample sealed after collection, Figure 3.20.

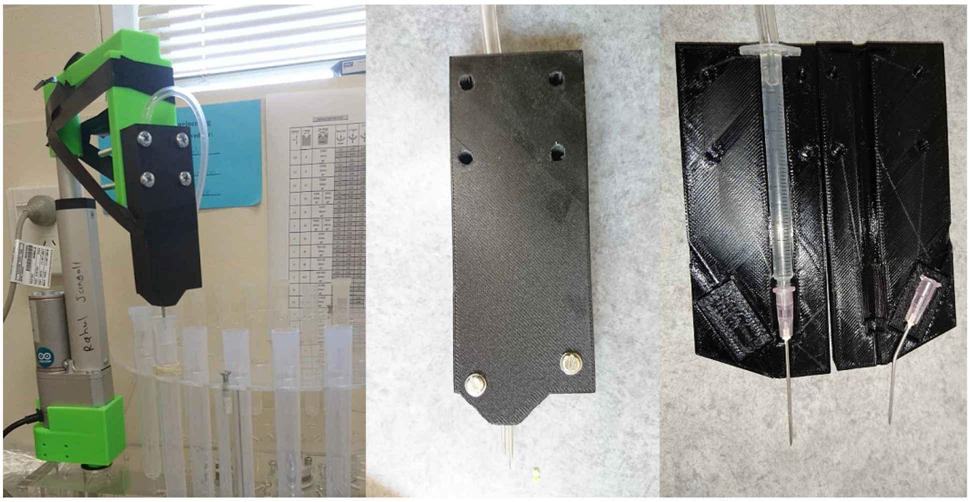


Figure 3.20: *From left to right - Final syringe mould being tested, Final syringe mould 3D printerd on high resolution printer, 3d printed prototype of the final syringe mould.*

3.5 Electronics storage

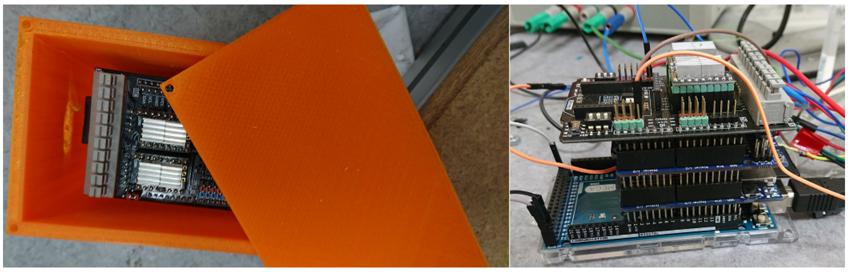


Figure 3.21: *From left to right - 3D printed case to store electronics components, Arduino microncontroller and shields used in the project.*

Caves are mostly moist and have high humidity. It is therefore essential to take preventive measures to waterproof electronic components of the autosamplers to avoid rust and short circuit of the electronic components. A 3D printed case was designed to accommodate the Arduino microcontroller and the shields, as shown in Figure 3.21. The case will be fastened, all the wire openings coming out of the case and gaps will be closed with Room-Temperature-Vulcanizing silicone upon deployment. The size of the case is optimized to be stowed under the platform between two stepper motors.

Chapter 4

Position Control and Performance evaluation

Position control was the most challenging part of the project. The rotation of the turntable has to be intermittent due to the requirement of collecting drip water in new sample vials every week, and another biggest challenge was to ensure the sample vial is directly under the pierce point of the syringe needle within a window of 3mm from the center of the septa. Various methods have been explored to find the best balance between the auto-sampler size and power saving.

4.1 Geneva mechanism

The geneva mechanism is one of the oldest mechanisms to convert continous rotation into intermittent motion and is still commonly used in various applications. The geneva mechanism consists of a rotating drive with a pin that slides into the slot of the driven geneval wheel, Figure 4.1. This mechanism also offers postion locking because of the raised slot on the drive wheel which locks the driven wheel [10], [11].

For this autosampler project, a custom geneva drive with 23 dwells was designed and built as a proof of concept and to demonstrate its position control and locking of the sample vial in place, Figure 4.2, the design calculations of the geneva drive refer to Figure

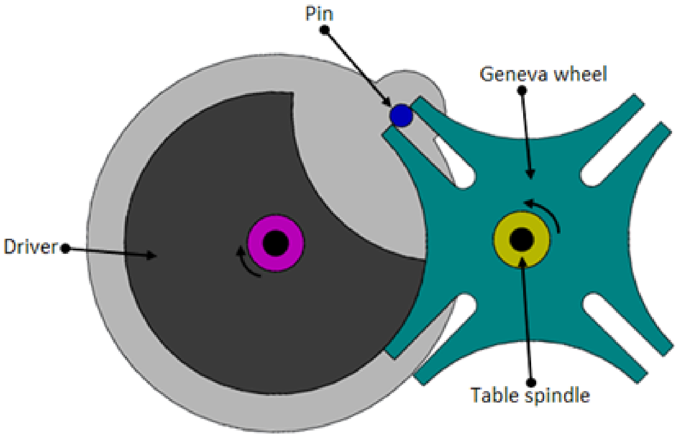


Figure 4.1: *Geneva mechanism description, [6].*

(a) *Geneva mechanism description.* ***(b)*** *Geneva drive prototype.*

Figure 4.2: *Custom geneva drive design and prototype.*

B.1 in Appendix B.1. The Geneva drive proved very reliable regarding position control and its locking abilities, but the size and number of samples was a concern for the client. At 23 dwells the auto-sampler turntable can hold 46 samples, but the diameter of the wheel came to over 300mm. Though this mechanism would solve the problem of position control and locking, this concept was scrapped due to the concerns with the size which would make the auto-sampler bigger than necessary.

4.2 Open-loop position Control with a Stepper Motor

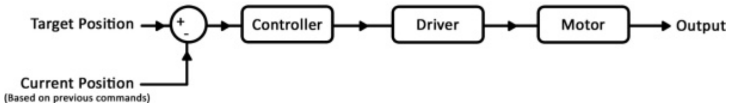
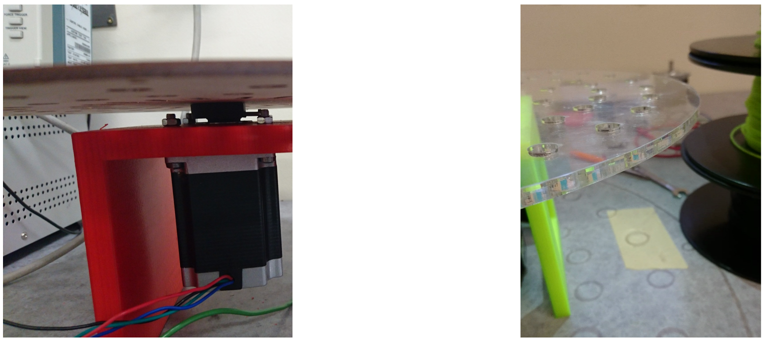


Figure 4.3: *Open loop feedback schematic [7].*

Electromechanical systems can be used in an open loop fashion from a control point of view. In an open loop scenario, Figure 4.3, the devices are driven by applying simple input signals [12]. For this project, a turntable was set-up to test the suitability of a stepper motor to integrate into the auto-sampler. The set-up consists of a stepper motor attached to the 3D printer platform. The turntable was attached to the stepper motor through an adaptor which is fitted into a deep groove ball bearing as shown in Figure 4.4a.

The sequence of the test is based on the block diagram shown in Figure 4.3. The stepper motor used in this experiment is a NEMA 23 JK57HS76-2804, Refer to Chapter A, Figure A.4 in Appendices for technical details. There are 25 sample vials slots on the turntable equally spaced at 14.4°. The target position for the stepper motor is to rotate 14.4° (8 steps) from its current position for 25 times with a few seconds delay to speed up the process. Arduino mega micro-controller and arduino IDE software were used to control the stepper motor at set increments. Many stepper motor drivers have been used with Arduino micro-controller refer to appendix A for further information.

The tests were conducted in two different scenarios, in the first test power was kept on between each rotation so the stepper motor coils were fully energized and would result in no loss of torque. The position was tracked using lighting and target pen mark as seen in Figure 4.4b. From observation, the stepper motor was exact within 3mm from the center of the septa pierce point, though the error between target and current position increased as the number of samples passed the difference was negligible. A multimeter was used

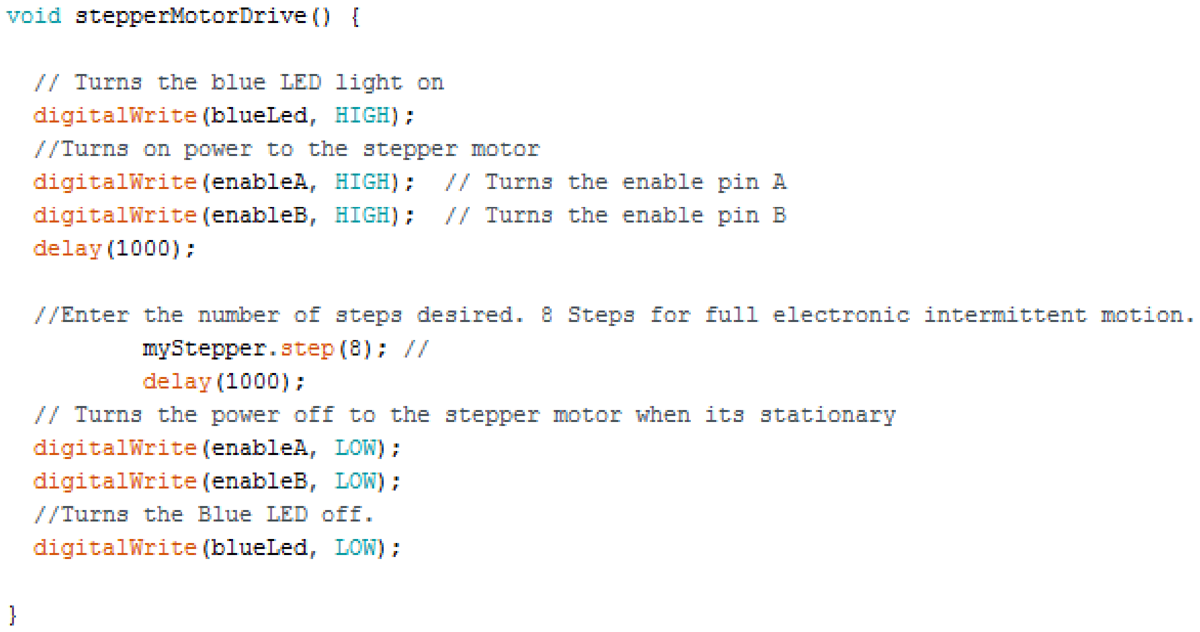


(a) Setup for open loop positioning testing.

(b) Testing position control of an open loop stepper motor.

Figure 4.4: *Open loop stepper motor position control setup.*

to record current draw, during rotation the current was recorded at 0.6A and when the motor was stationary the current draw was recorded at 0.3A which are not suitable for deployments.



```
void stepperMotorDrive() {

  // Turns the blue LED light on
  digitalWrite(blueLed, HIGH);
  //Turns on power to the stepper motor
  digitalWrite(enableA, HIGH);  // Turns the enable pin A
  digitalWrite(enableB, HIGH);  // Turns the enable pin B
  delay(1000);

  //Enter the number of steps desired. 8 Steps for full electronic intermittent motion.
          myStepper.step(8); //
          delay(1000);
  // Turns the power off to the stepper motor when its stationary
  digitalWrite(enableA, LOW);
  digitalWrite(enableB, LOW);
  //Turns the Blue LED off.
  digitalWrite(blueLed, LOW);

}
```

Figure 4.5: *Arduino code to create intermittent motion and power saving with the stepper motor.*

For the second test, power was cut by turning the enable pins off to the stepper motor between rotations to simulate cave deployments as the motor will be in the same position for up to a week, See Figure 4.5. The behavior of the stepper motor was random and unpredictable, few rotations were precise, and then the error kept increasing beyond the septa center. Upon close observation, it was found that when power is enabled back on there is a jitter in the motor, which meant its current position is slightly over or under the desired target position. However, the multimeter recorded the current draw at 0amps between rotation which is the desired outcome and will help in increasing the longevity of the deployment. Due to the position control issues with the open loop feedback this concept was phased out.

4.3 Closed loop proportional control

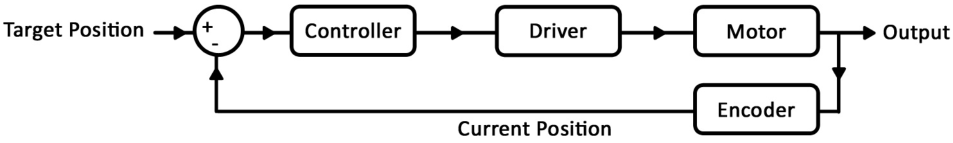


Figure 4.6: *Closed loop feedback schematic [7].*

This experiment aimed to confirm if a closed loop feedback stepper motor with proportional control would be a suitable option for the cave auto-sampler. The test followed the sequence of the block diagram as shown in Figure 4.6 and had the same set-up as discussed in the previous section, Figure 4.7. The stepper motor used in this test is a NEMA 23 57J1880EC-1000 closed-loop hybrid stepper motor with optical encoder feedback. The test was conducted in three different scenarios, the first test with power turned on, the second test with disabling the pins on stepper motor driver and third scenario with proportional control algorithm.

The performance of the motor from the first scenario was as expected, the stepper motor never lost a step, and it was on point at the target position for every cycle. For the second scenario, when the enable pins were turned off, the performance of the motor was similar to that of the motor discussed in the previous section.

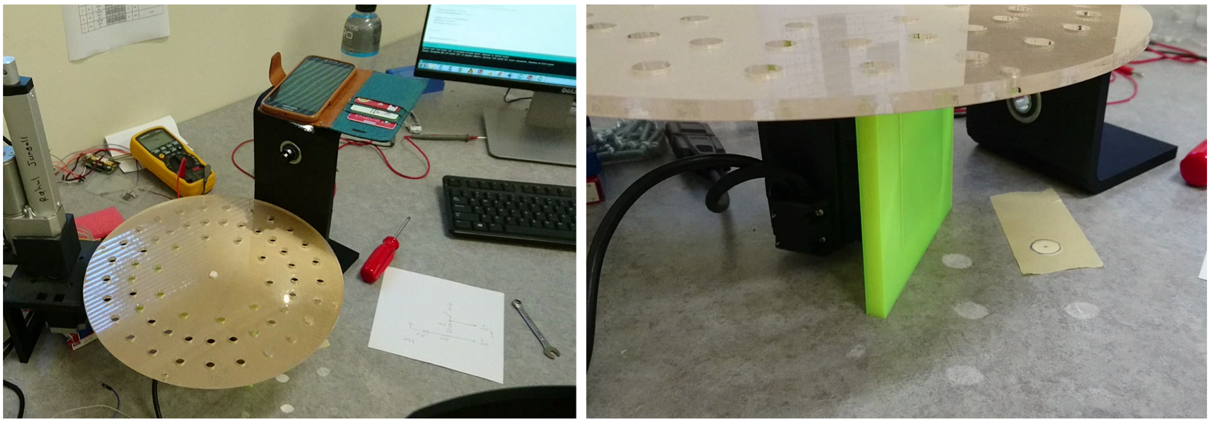
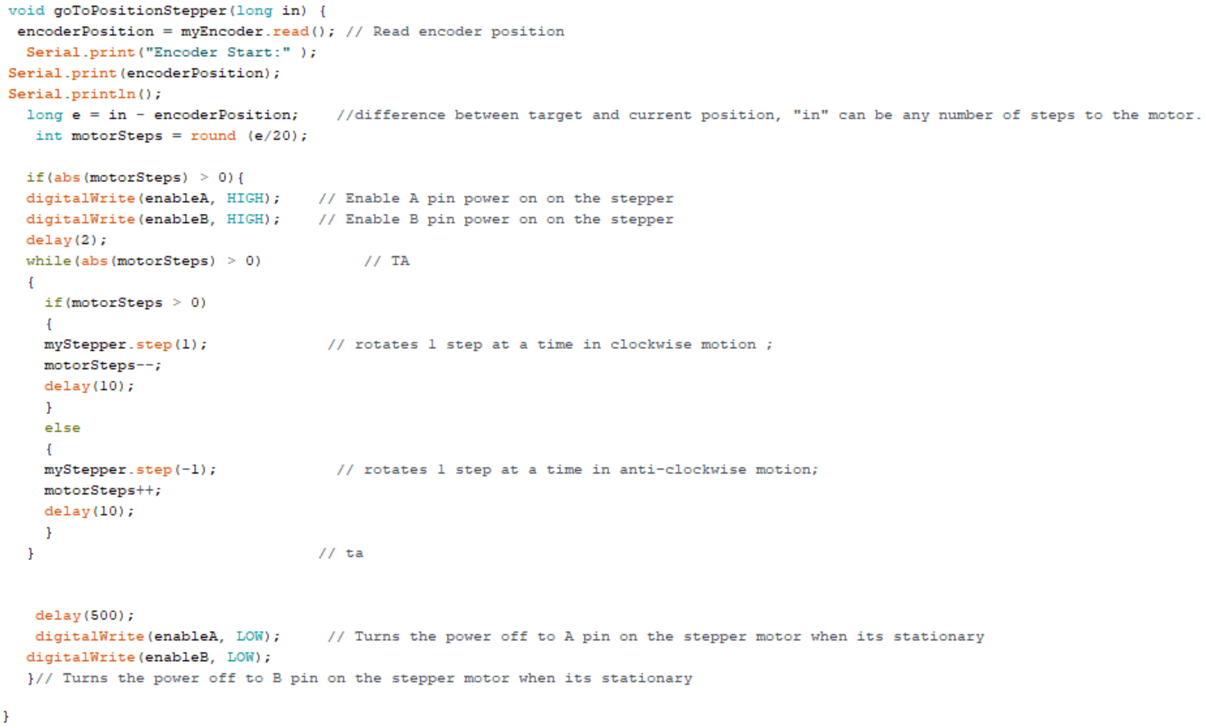


Figure 4.7: *Position control testing with closed loop stepper motor.*

In the third scenario, a proportional control algorithm was integrated into the software (Figure 4.8) to mitigate the error in position as this motor had a built-in encoder with a resolution of 1000 pulses per revolution. This method did show improvement over the open loop. The stepper motor would correct itself towards the specified target position but not on every cycle. Some cycles the encoder would read as though it is on target, but visual tracking would show otherwise.

Various methods such as micro-stepping, changing motor speeds, exploring different stepper motor drivers have been tried, neither of these techniques showed any significant improvement in the position control. From observation, it was found that micro-stepping increased the jitter in the stepper motor when turned back on. The other major drawback with using an optical encoder was that it would lose its position when the power is turned off, and it was also found that the encoder would still read zero with the jitter and wasn't helpful in mitigating the error caused by the jitter. In conclusion, the closed loop proportional method is an effective method if the power to the stepper motor and encoder were kept turned on between rotations, because of the nature of the power requirements during cave deployments closed loop control using an optical encoder for position control is not a suitable option.



```
void goToPositionStepper(long in) {
 encoderPosition = myEncoder.read(); // Read encoder position
  Serial.print("Encoder Start:" );
Serial.print(encoderPosition);
Serial.println();
  long e = in - encoderPosition;    //difference between target and current position, "in" can be any number of steps to the motor.
   int motorSteps = round (e/20);

  if(abs(motorSteps) > 0){
  digitalWrite(enableA, HIGH);    // Enable A pin power on on the stepper
  digitalWrite(enableB, HIGH);    // Enable B pin power on on the stepper
  delay(2);
  while(abs(motorSteps) > 0)           // TA
  {
    if(motorSteps > 0)
    {
    myStepper.step(1);              // rotates 1 step at a time in clockwise motion ;
    motorSteps--;
    delay(10);
    }
    else
    {
    myStepper.step(-1);              // rotates 1 step at a time in anti-clockwise motion;
    motorSteps++;
    delay(10);
    }
  }                               // ta


   delay(500);
   digitalWrite(enableA, LOW);     // Turns the power off to A pin on the stepper motor when its stationary
  digitalWrite(enableB, LOW);
  }// Turns the power off to B pin on the stepper motor when its stationary

}
```

Figure 4.8: *Position control arduino code.*

4.4 Solenoid locking position control

Figure 4.9: *Position control with a pull type solenoid prototype.*

Stepper motor needs to be turned off between rotations to minimize power consumption and ensure the longevity of cave deployments. As a result of a frequent power cut, Jitter in the stepper motor has been a major concern as discussed in both the previous sections. A spring-loaded pull-type solenoid concept was explored to lock the turntable in place when the motor is turned off and to minimize or eliminate jitter when turned on. See Figure A.2 in Appendix A for technical details of the pull-type solenoid used.

The pull-type solenoid sits inside a mold which is fastened to the platform as seen in Figure 4.9. The height of the solenoid is adjusted so the pin slides into an oblong slot to lock the table in position. A quarter step microstepping resolution is used to decrease the speed of the stepper motor and to reduce the rotational impact on the solenoid pin. The oblong slot is 1mm wider than the diameter of the solenoid pin at 4mm to avoid jamming issues as shown in Figure 4.10.

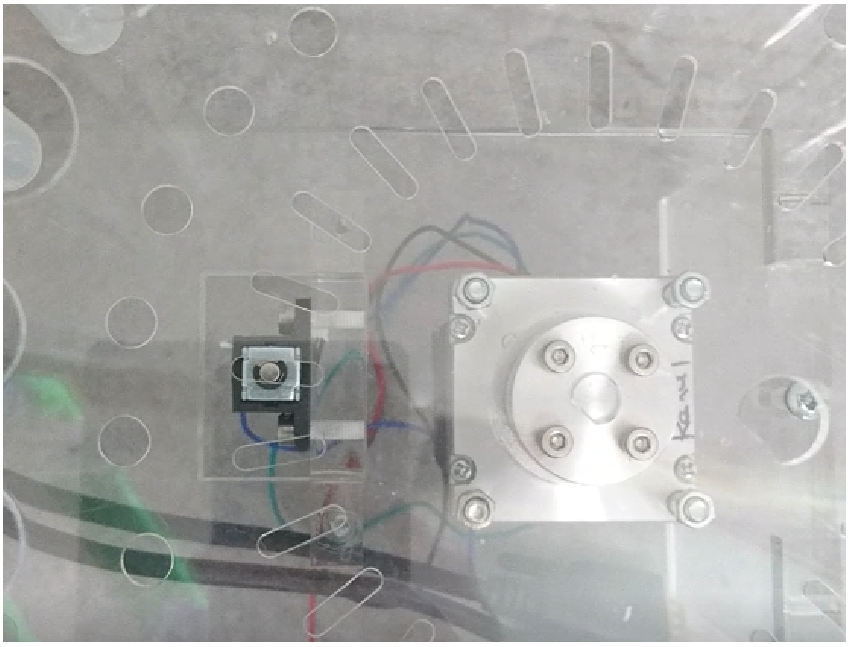
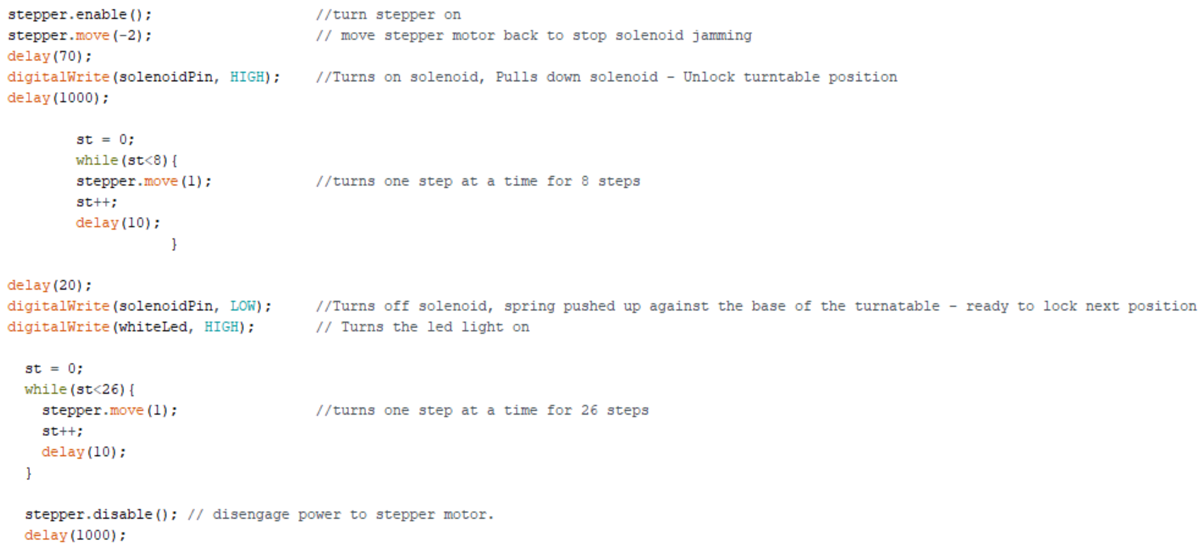


Figure 4.10: *Position control testing with a pull type solenoid.*

This method is very similar to the open loop position control but with solenoid locking the position of the turntable. The operational sequence Arduino code for the position control using a solenoid is shown in Figure 4.11. This method proved to be very efficient to control and lock the position and to minimize the jitter in the stepper motor caused by the frequency of cutting the power. Upon further testing, it was seen that the pin was getting jammed in the oblong slot due to overshooting of stepper motor after a certain number of cycles and also the solenoid pin had a natural play when pushed out which reduced the ability to stay at target position completely. Rotating the stepper anticlockwise two steps and then releasing the solenoid solved the issue of jamming. Though this method was satisfactory and far superior regarding position control than the previous two methods, it was decided not to go ahead due to play issues with the solenoid.



```
stepper.enable();                   //turn stepper on
stepper.move(-2);                   // move stepper motor back to stop solenoid jamming
delay(70);
digitalWrite(solenoidPin, HIGH);    //Turns on solenoid, Pulls down solenoid - Unlock turntable position
delay(1000);

        st = 0;
        while(st<8){
        stepper.move(1);            //turns one step at a time for 8 steps
        st++;
        delay(10);
                   }

delay(20);
digitalWrite(solenoidPin, LOW);     //Turns off solenoid, spring pushed up against the base of the turnatable - ready to lock next position
digitalWrite(whiteLed, HIGH);       // Turns the led light on

  st = 0;
  while(st<26){
    stepper.move(1);                //turns one step at a time for 26 steps
    st++;
    delay(10);
  }

  stepper.disable(); // disengage power to stepper motor.
  delay(1000);
```

Figure 4.11: *Position control with a pull type solenoid.*

4.5 Geared stepper motor - Open loop

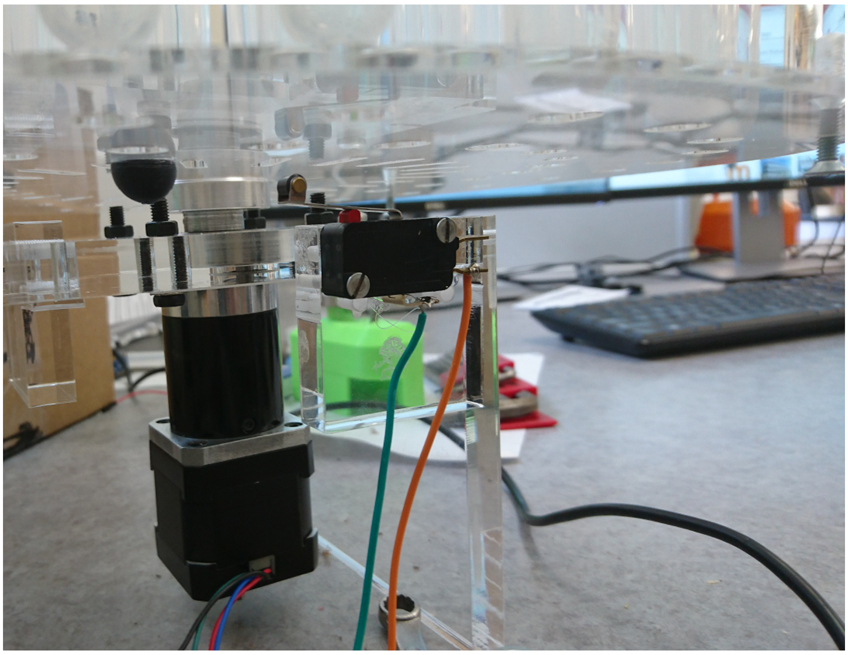


Figure 4.12: *Geared stepper motor position control test.*

A NEMA 17 stepper motor with an integrated planetary gearbox with a gear ratio of 99.5:1 was tested to see if it could be a suitable fit to the auto-sampler and minimize the issues of jitter and position control. The test set-up and sequence are similar to the methods discussed in Subsection 4.2 but with a different motor. The setup is shown in Figure 4.12.

The test was again performed in two scenarios with full power and the second test with enabling and disabling of the pins on the driver. Visual observation showed no difference in the performance of the stepper motor regarding position control. No jitter picked up in the motors due to the frequency of enabling and disabling the pins on the driver; it was also found that there was no loss of torque when the stepper motor enable pins were turned off. This could mainly be attributed to the high gear ratio of the integrated planetary gearbox on the stepper motor. The positioning of the turntable was tracked using manual methods such as lighting and marking as seen in Figure 4.13. Further testing confirmed the

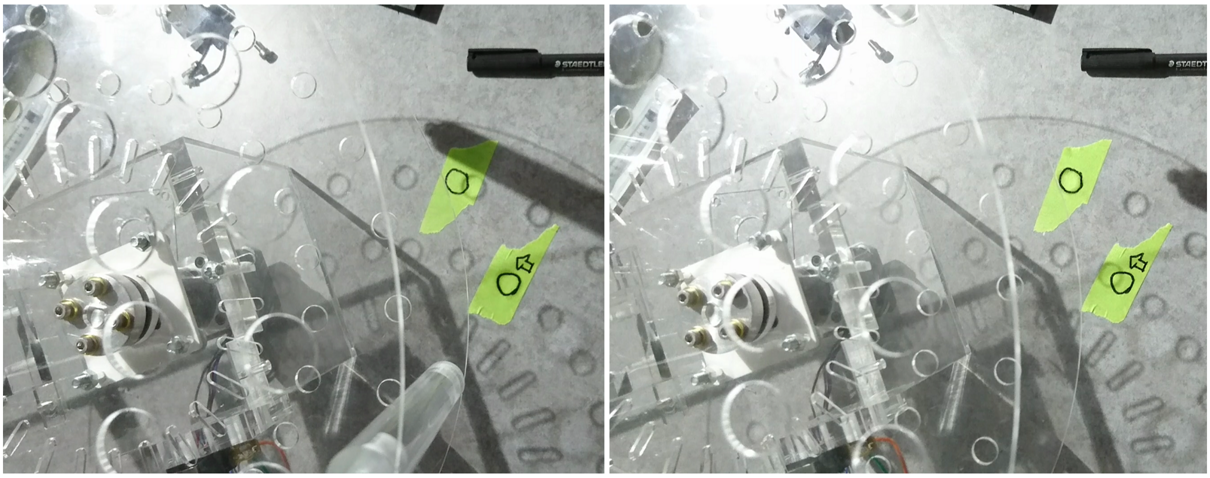
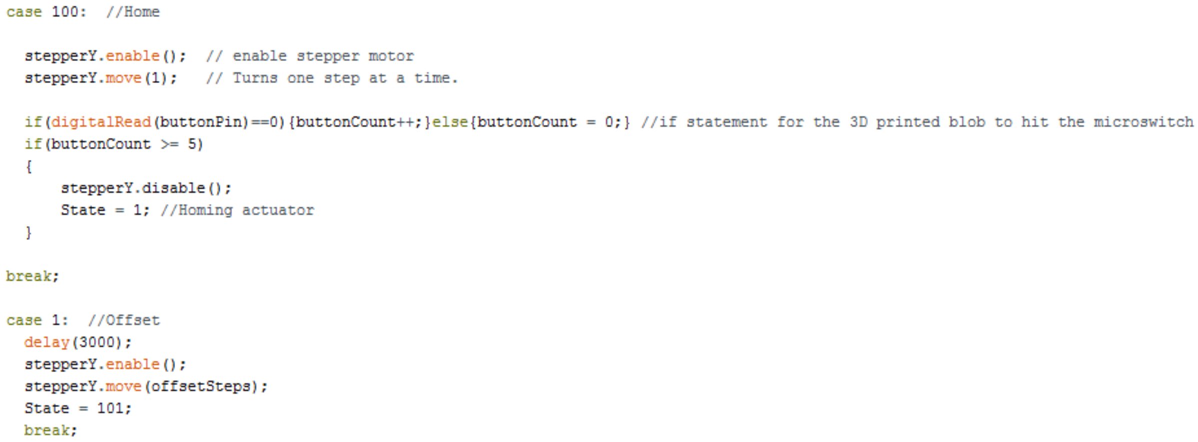


Figure 4.13: *From left to right - Turntable at home position, Turntable at sample vial position.*

Due to the high gear ratio of the stepper motor the turntable cannot be adjusted to the home position manually. A microswitch and a 3D printed blob (Figure 4.12) is used on the auto-sampler to home position the turntable and then offset a few steps from then for the sample vial to be in position for water collection. The Arduino code for this operation can be seen in Figure 4.14. The point of contact between the blob and the microswitch is home position and then offset position is the start of the cycle for sample collection, where the turntable comes under the pierce point from the rotating arm.



```
case 100:  //Home

  stepperY.enable();  // enable stepper motor
  stepperY.move(1);   // Turns one step at a time.

  if(digitalRead(buttonPin)==0){buttonCount++;}else{buttonCount = 0;} //if statement for the 3D printed blob to hit the microswitch
  if(buttonCount >= 5)
  {
      stepperY.disable();
      State = 1; //Homing actuator
  }

break;

case 1:  //Offset
  delay(3000);
  stepperY.enable();
  stepperY.move(offsetSteps);
  State = 101;
  break;
```

Figure 4.14: *Arduino code for home positioning the stepper gear motor.*

During testing a minor backlash from the integrated planetary gearbox was picked up, this issue was mitigated by adding friction to the turntable using a silicone stopper as shown in Figure 4.15. In conclusion, test with this motor proved to be successful and has met the position control requirements. The sample vials were at the desired target position on many cycles performed. This motor is now integrated onto the turntable and Linear actuator mold for position control in the final auto-sampler.

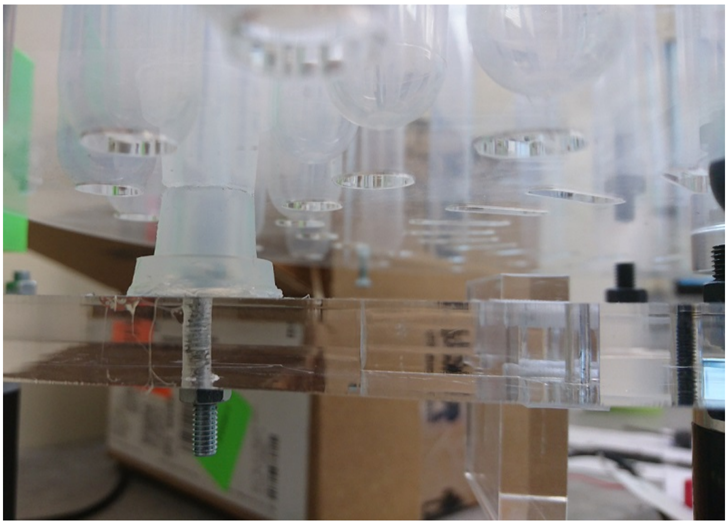


Figure 4.15: *Arduino code for home positioning the stepper gear motor.*

4.6 Sustainability

The scope of this project is based on developing an automated deployable auto-sampler. The need to go to a sample site every month to collect drip water samples is eliminated. Automating the process of sample water collection will reduce travel expenses and environmental impacts due to frequent trips to the cave sampling site.

4.7 Power saving strategies

A new sample has to come into position for collection once every week, which means that the autosampler is idle for most of its time during cave deployment. Maximum power savings are done when the device is not in operation by enabling the microcontroller into sleep mode and using RTC interrupts to wake the microcontroller on a weekly basis. Based on one-year deployment (50 weeks - one sample each week) and seven minutes of operation every week: The autosampler will be in sleep mode for 8730 hrs and in operation for 6 hours in total.

Tests were performed with the autosampler in the lab with a 12V DC power supply; the average current consumption was recorded at 0.6A. The operation includes rotation of the turntable, turning the rotating arm, cave environmental monitoring sensors and data logging. The average energy requirement for operation during cave deployments is 3.6 Ah.

In an idle state, the steppers motors pins are disabled, and no current consumption has been recorded from the motors, the cave sensors are turned off using the digital I/O pins on the Arduino, and the microcontroller is put into sleep mode with most power savings (SLEEP MODE PWR DOWN). The current consumption was recorded at 0.032A in sleep mode. The energy consumption during sleep mode is 279.36 Ah. Further savings were achieved by the closing the 16U2 USB reset mode using a win jumper, during which the current consumption was recorded at 0.0126A, which gives an energy consumption of 109.99 Ah during sleep mode. The total energy consumption for one year comes to 113.598 Ah.

The datalogger used on the autosampler has an onboard PCF8523 RTC and an alarm triggers an interrupt using a square wave pin to wake the Arduino from sleep mode. The energy consumption calculations have been submitted to the client to make an informed decision on the purchase of the battery.

4.8 Modularity

One of the desired features in the autosampler is that is should be modular and easy to assemble on site. This was taken into consideration throughout the design phase.The scientists plan a site visit to the caves usually in a group of at least four people which makes it ideal to carry the autosampler in separate units and assemble it on site. The autosampler can be split into three main modular assembly units to be carried in separate backpacks. It is recommened to pre assemble the main assembly modular units at the research institute.

- Rotating arm assembly, which includes: Linear actuator, actuator holder, rotating arm and syringe mold.
- Platform assembly: Laser cut platform, stepper motors, turntable adaptor and microswitch.
- Turntable assembly: Sample vials and septa, Acrylic rounds, top and bottom tables.

The actuator holder is designed to suit the key hole profile of the stepper motor shaft for ease of assembly as shown in Figure 4.16. This design feature would enable the scientists to assemble the rotating arm assembly onto the platform assembly without requiring addtional steps.

The turntable assembly is fastened onto the adaptor with four set screws. Turntable assembly had a high torque requirement therefor a key hole hole design using 3D print

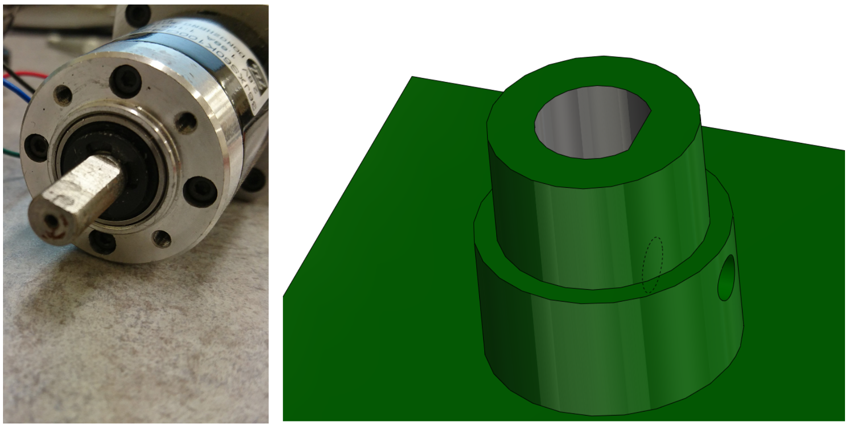


Figure 4.16: *Actuator holder key hole design for ease of assembly.*

models were ineffective and sheared often after a few cycles. The adaptor is machined from an aluminum round which offer the perfect balance between weight, ease of assembly and torque requirements.

4.9 Testing in lab conditions

The auto-sampler was tested in lab conditions extensively for repeatability and consistency in operation. This section will outline the issues that were picked up during testing and present possible solutions.

As the silicone septa is pierced by the actuator, the truntable flexes which is mainly due to the thinner material used to keep the overall weight to minimum. The flex was minimized by adding a 3D printed blob was glued to the platform which minimised flexing, as shown in Figure 4.17. Flexing of the turntable was minimsed as a result but not elimintated. Due to the slight flexing of the turntable the needles pierce the septa on a slight angle and thenrecover which made the needles blunt over time as shown on the right side in Figure 4.17. The flexing can be further minimized by using a thicker material bottom table.

The other issue that was picked up during testing was that the actuator holder does not hold the linear actuator base tight. As a result the actuator bends back. The flex in

Figure 4.17: *From left to right: 3D printed blob to minimise flexing in turntable, Blunt needle tips.*

the needle could also be attributed to this issue. Higher side walls can helo mitigating the issue of backplay of the actuator. These are some of the major issues which affects the repetability of the autosampler and needs to be adressed before deployment.

4.10 Cave testing

After the issues from the above section are addressed. It is recommended to do a few days of cave test since lab environmental conditions do not simulate cave conditions. This test will give an idea as to how the machine will perform in damp and humid conditions; this test will also test the integrity of waterproofing of the electronics in the case. This test will be an opportunity to modify the machine to suit for cave condition during long-term deployments.

4.11 Cost

Table 4.1: Cost breakdown for the autosampler.

Item	***Qty***	***Price (NZD)***
Laser cut platform	1	40
Platform legs	2	40
Turntable	2	25
PP vials	50	9.75
Linear actuator	1	60
Stepper motors	2	130
Arduino Mega	1	59.36
ADA fruit data logger	1	26
DF robot CO2 sensor	1	90
BMP280 climate sensor	1	15
ADA fruit motor shield	1	36
DF robot stepper motor shield	1	31
Bearings	2	14
Microswitch	1	10
Ball catch	2	12.42
Total		**598.53**

Table 4.1 summarises the cost of critical components for the auto-sampler. The cost excludes 3D printing components due to low cost of 3D printing material.

The development of the auto-sampler project did not have a set budget. All the possible cost preventative measures were taken to develop this auto-sampler. This project made frequent use of the machine shop and 3D printers at the university. CNC milling the platform, machining of the adaptor and machining the turntable pillars were manufactured in-house to keep the costs down. Few of the electronic components such as stepper motors, linear actuator were purchased from suppliers in China to keep the costs to a minimum.

4.12 Operational sequence and final design

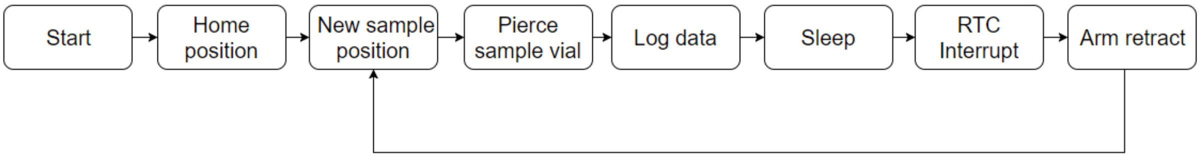


Figure 4.18: *Operational sequence of the auto-sampler during deployment.*

The operational sequence of the autosampler is based on the block diagram as shown in Figure 4.18. When the autosampler is first turned on, the linear actuator pushes away from the turntable to home position using an inbuilt potentiometer, after the actuator has reached its home position the turntable is set to home position using the 3D printed blob and a microswitch.

The stepper motor was calibrated to go to the target position where the sample vial is under rotating arm for sample water delivery. The linear actuator punctures the silicone septa for drip water collection. Cave sensors log data for up to seven minutes, and the microcontroller is set to sleep mode for up to a week. The RTC alarm triggers an interrupt to wake the Arduino up from sleep mode, and the linear actuator pulls out of the sample vial. This cycle repeats for 25 times until the rotating arm changes its position to fill the sample vials in the next 25 rows.

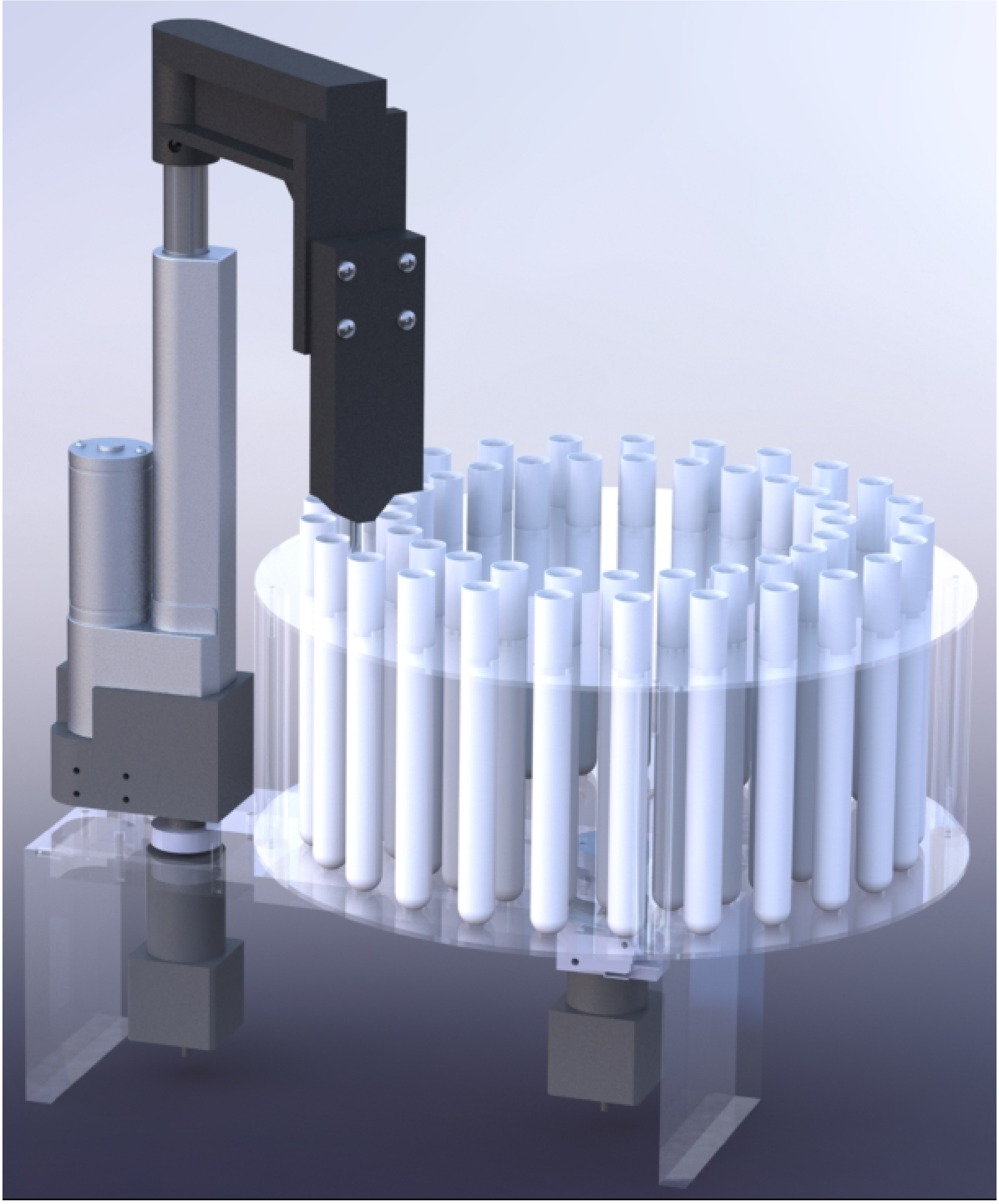


Figure 4.19: *Render of the final auto-sampler.*

Chapter 5

Conclusion

A portable auto-sampler for collecting drip water from the caves was designed and built. This auto-sampler can collect and store sample water in an airtight container. There are no deployable cave auto-samplers currently on the market that can store samples in an airtight container and maintain them in a sealed state, so it has potential to find a place in the market for science equipment. The sample vials are stored on a carousel styles turntable with a capacity for fifty 20 ml sample polypropylene vials. The autosampler has additional features such monitoring and data logging cave environmental conditions such as temperature, CO2, and pressure. Arduino MEGA2560 is used to control all the major electronic components of the autosampler. Based on 50-week deployment, the autosampler has an energy consumption of 113.598 Ah. The autosampler can be carried in three separate backpacks and allows for easy assembly upon reaching the sample site. The auto-sampler has a total weight of 6kgs.

The auto-sampler has been tested in lab conditions. There are a few issues that need to be addressed before deployment to improve the consistency of the machine. The linear actuator does not sit tightly inside the holder. As a result, it has back play when the needles pierce the silicone septa. The turntable has a flex when it's being pierced, placing a 3D printed blob under the table minimized the play but hasn't been eliminated.

Future work

With the development of the auto-sampler showing plausibility, it is vital to consider what further work needs to be done. Some of the aspects to be considered for future work are as follows:

Power savings is a significant aspect in the auto-sampler project which affects the longevity of the deployment. It is recommended to transition from stock market controller such as Arduino to a custom built PCB microcontroller which would have much lower energy requirements during sleep mode.

The geared stepper motor is accurate regarding position control. This system can be made even robust by integrating absolute encoder feedback on the turntable. Unlike optical encoders, absolute encoders do not lose track of position when the power is turned off which is a necessary operation to conserve energy.

For this autosampler to transition into a production unit, other manufacturing methods need to be researched to replace all the 3D printed parts.

Cave drip water discharge rates can vary from high to low depending on the season. Summer usually have low discharge rates compared to winter. The auto-sampler would need a capability to detect when the sample vial is full rather than relying on a pre-programmed basis.

Bibliography

[1] GE Healthcare Life Sciences. Fraction collector f9-r, 2017.

[2] Robert D Doty. A portable, automatic water sampler. *Water Resources Research*, 6(6):1787–1788, 1970.

[3] Rohit Salve. A passive-discrete water sampler for monitoring seepage. *Ground water*, 43(1):133–137, 2005.

[4] R.W.Malone, J.V.Bonta, and D.R.Lightell. A low-cost composite water sampler for drip and stream flow. *American Society of Agricultural Engineers*, 19(1):59–61, 1985.

[5] L. Berglind and A. Henriksen. Automatic sampler for atomic absorption and automatic colorimetric analysis. 19(9):918–920, 1952-1992.

[6] Drives and nptel Mechanisms. Geneva mechanism, 2017.

[7] U.A. Bakshi. *Control Systems*. Technical Publications, 2008.

[8] New Gottland. Make geneva wheels of any size, 2012.

[9] arduino.cc. Getting started with arduino and genuino mega2560, 2017.

[10] J. H Bickford. *Mechanisms for intermittent motion*. Industrial Press, New York, 1972.

[11] G. Meyer. A tested method for precise intermittent motion. *Machine Design*, 60(1):140 – 3, 1988/01/07. Geneva mechanism;intermittent rotary motion;driver;wheel;cam follower;hub;.

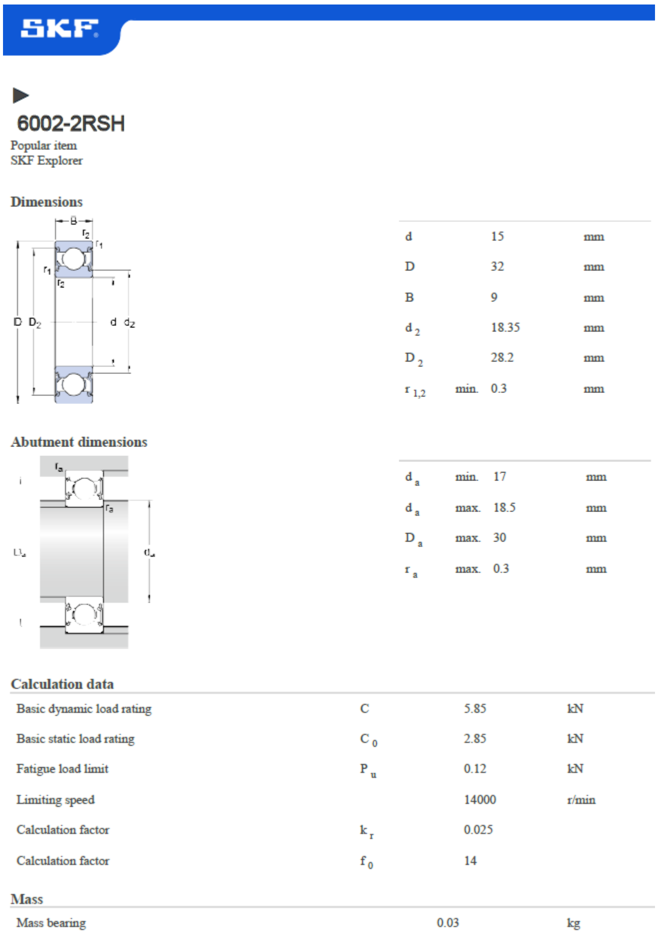
[12] B Borovic, A Q Liu, D Popa, H Cai, and F L Lewis. Open-loop versus closed-loop control of mems devices: choices and issues. *Journal of Micromechanics and Microengineering*, 15(10):1917, 2005.

[13] David Rh. Power saving techniques on a stock mega 2560.

Appendix A

Data Sheets

A.1 Bearings, ball catch and solenoid.



▶

6002-2RSH

Popular item
SKF Explorer

Dimensions

d		15	mm
D		32	mm
B		9	mm
d_2		18.35	mm
D_2		28.2	mm
$r_{1,2}$	min.	0.3	mm

Abutment dimensions

d_a	min.	17	mm
d_a	max.	18.5	mm
D_a	max.	30	mm
r_a	max.	0.3	mm

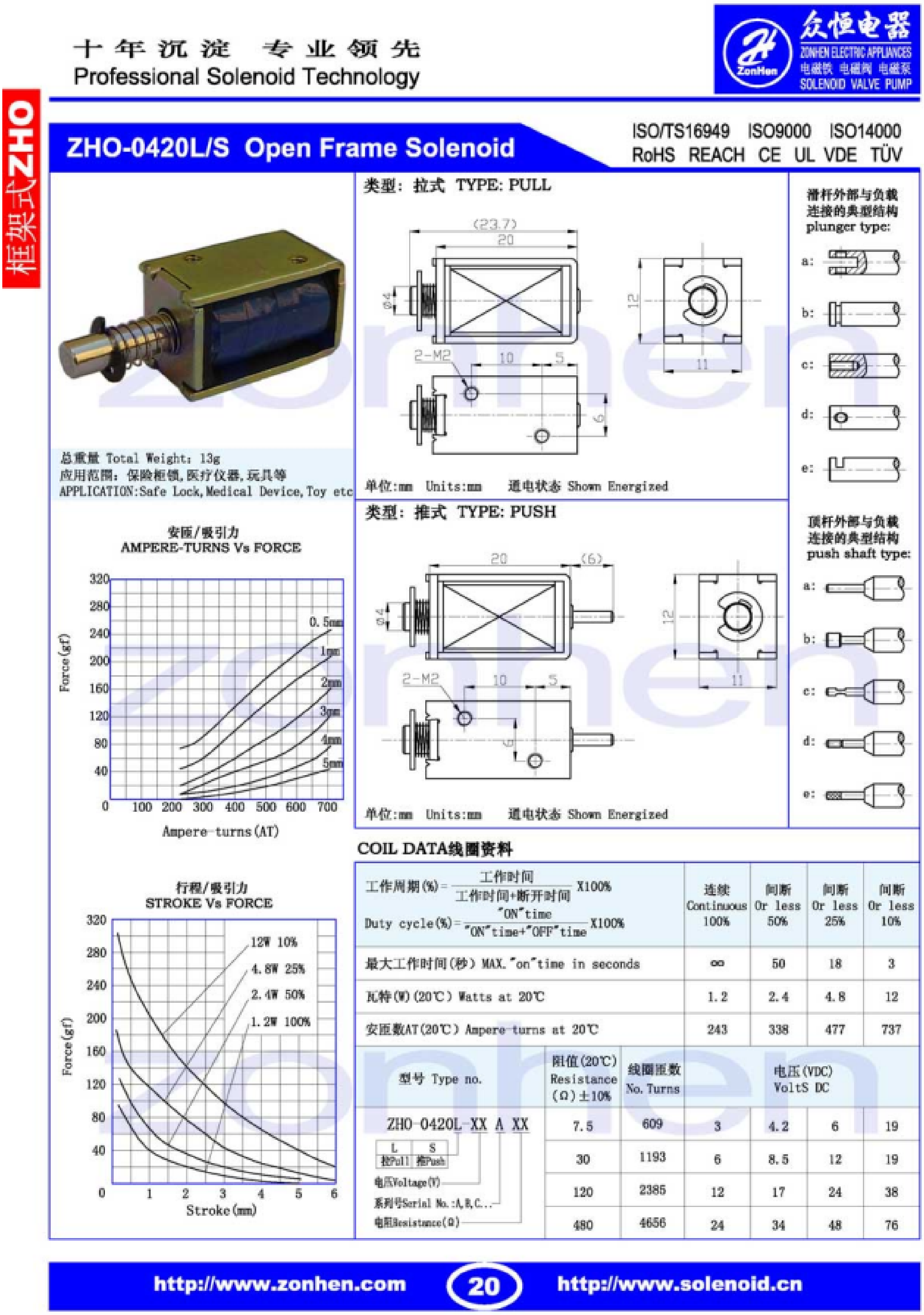
Calculation data

Basic dynamic load rating	C	5.85	kN
Basic static load rating	C_0	2.85	kN
Fatigue load limit	P_u	0.12	kN
Limiting speed		14000	r/min
Calculation factor	k_r	0.025	
Calculation factor	f_0	14	

Mass

Mass bearing	0.03	kg

Figure A.1: *SKF deep groove ball bearing datasheet - 6002-2RSH.*



ZHO-0420L/S Open Frame Solenoid

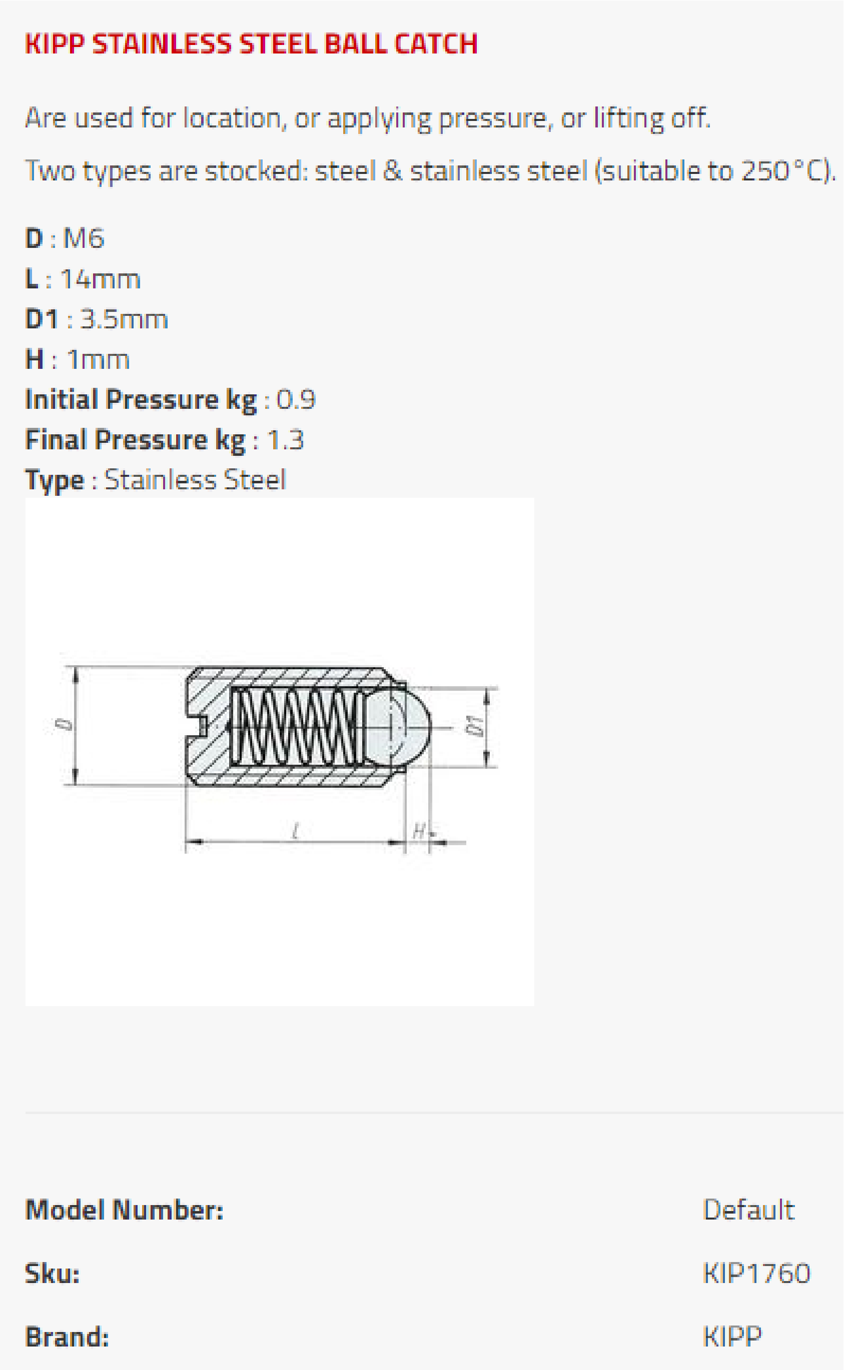
总重量 Total Weight：13g
应用范围：保险柜锁,医疗仪器,玩具等
APPLICATION:Safe Lock,Medical Device,Toy etc

COIL DATA线圈资料

工作周期(%) = 工作时间 / (工作时间+断开时间) X100% Duty cycle(%) = "ON"time / ("ON"time+"OFF"time) X100%	连续 Continuous 100%	间断 Or less 50%	间断 Or less 25%	间断 Or less 10%
最大工作时间(秒) MAX."on"time in seconds	∞	50	18	3
瓦特(W)(20℃) Watts at 20℃	1.2	2.4	4.8	12
安匝数AT(20℃) Ampere-turns at 20℃	243	338	477	737

型号 Type no.	阻值(20℃) Resistance (Ω)±10%	线圈匝数 No. Turns	电压(VDC) VoltS DC			
ZHO-0420L-XX A XX	7.5	609	3	4.2	6	19
	30	1193	6	8.5	12	19
	120	2385	12	17	24	38
	480	4656	24	34	48	76

Figure A.2: *5V solenoid technical details.*



KIPP STAINLESS STEEL BALL CATCH

Are used for location, or applying pressure, or lifting off.

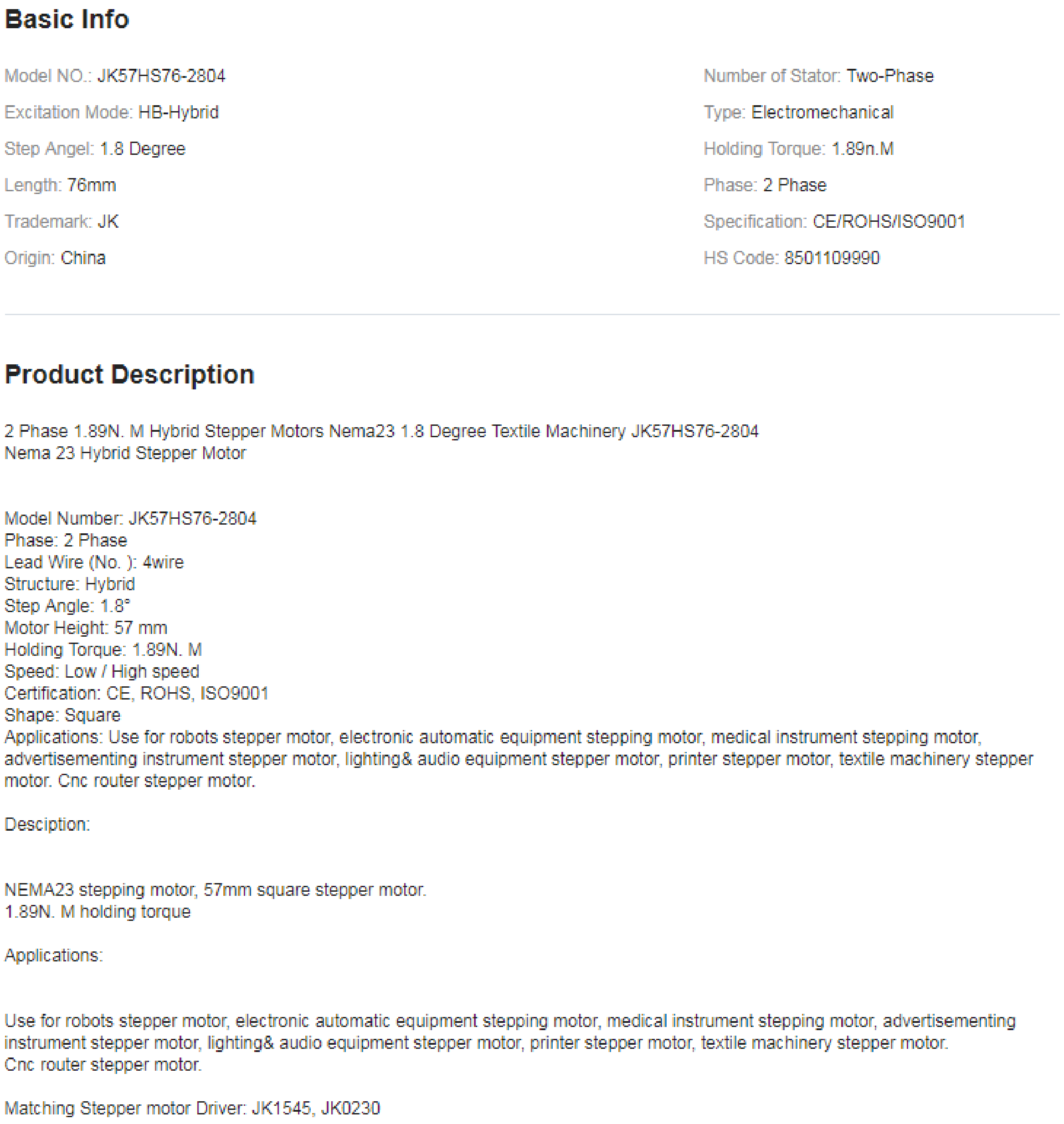
Two types are stocked: steel & stainless steel (suitable to 250°C).

D : M6
L : 14mm
D1 : 3.5mm
H : 1mm
Initial Pressure kg : 0.9
Final Pressure kg : 1.3
Type : Stainless Steel

Model Number:	Default
Sku:	KIP1760
Brand:	KIPP

Figure A.3: *Stainless steel ball catch specifications.*

A.2 Stepper motor



Basic Info

Model NO.: JK57HS76-2804
Excitation Mode: HB-Hybrid
Step Angel: 1.8 Degree
Length: 76mm
Trademark: JK
Origin: China

Number of Stator: Two-Phase
Type: Electromechanical
Holding Torque: 1.89n.M
Phase: 2 Phase
Specification: CE/ROHS/ISO9001
HS Code: 8501109990

Product Description

2 Phase 1.89N. M Hybrid Stepper Motors Nema23 1.8 Degree Textile Machinery JK57HS76-2804
Nema 23 Hybrid Stepper Motor

Model Number: JK57HS76-2804
Phase: 2 Phase
Lead Wire (No.): 4wire
Structure: Hybrid
Step Angle: 1.8°
Motor Height: 57 mm
Holding Torque: 1.89N. M
Speed: Low / High speed
Certification: CE, ROHS, ISO9001
Shape: Square
Applications: Use for robots stepper motor, electronic automatic equipment stepping motor, medical instrument stepping motor, advertisementing instrument stepper motor, lighting& audio equipment stepper motor, printer stepper motor, textile machinery stepper motor. Cnc router stepper motor.

Desciption:

NEMA23 stepping motor, 57mm square stepper motor.
1.89N. M holding torque

Applications:

Use for robots stepper motor, electronic automatic equipment stepping motor, medical instrument stepping motor, advertisementing instrument stepper motor, lighting& audio equipment stepper motor, printer stepper motor, textile machinery stepper motor.
Cnc router stepper motor.

Matching Stepper motor Driver: JK1545, JK0230

Figure A.4: *1.8 Degree NEMA23 JK57HS76-2804 Stepping Motor technical details.*

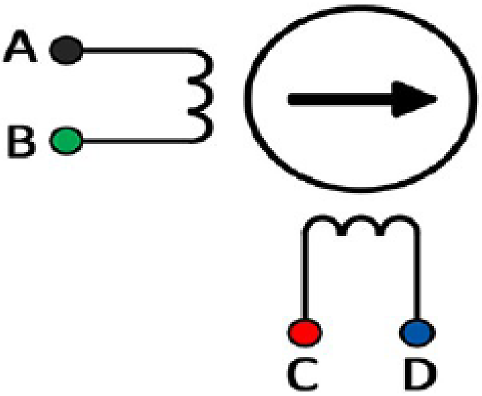
RB-Phi-267
12V, 1.7A, 667oz-in NEMA-23 Bipolar Stepper Motor

This NEMA-17 motor has an integrated Planetary gearbox with a 991044/2057 :1 ratio.It comes with the rear shaft exposed, so you can mount an encoder or shaft coupler. When connected to a Bipolar HC, the 3329 has a maximum speed of 34 RPM. At the output of the gearbox, the step angle is approximately 0.018°. When using the step angle in calculations, you should derive the exact step angle by dividing 1.8° by the gearbox reduction ratio.

At 1.7 Amps, this gearbox stepper can produce a maximum torque of 250 kg-cm. However, the gearbox is only rated for 48 kg-cm of continuous torque, and 100 kg-cm for brief overloads. Loading this gearbox stepper beyond the torque rating of the gearbox will shorten its useful life.

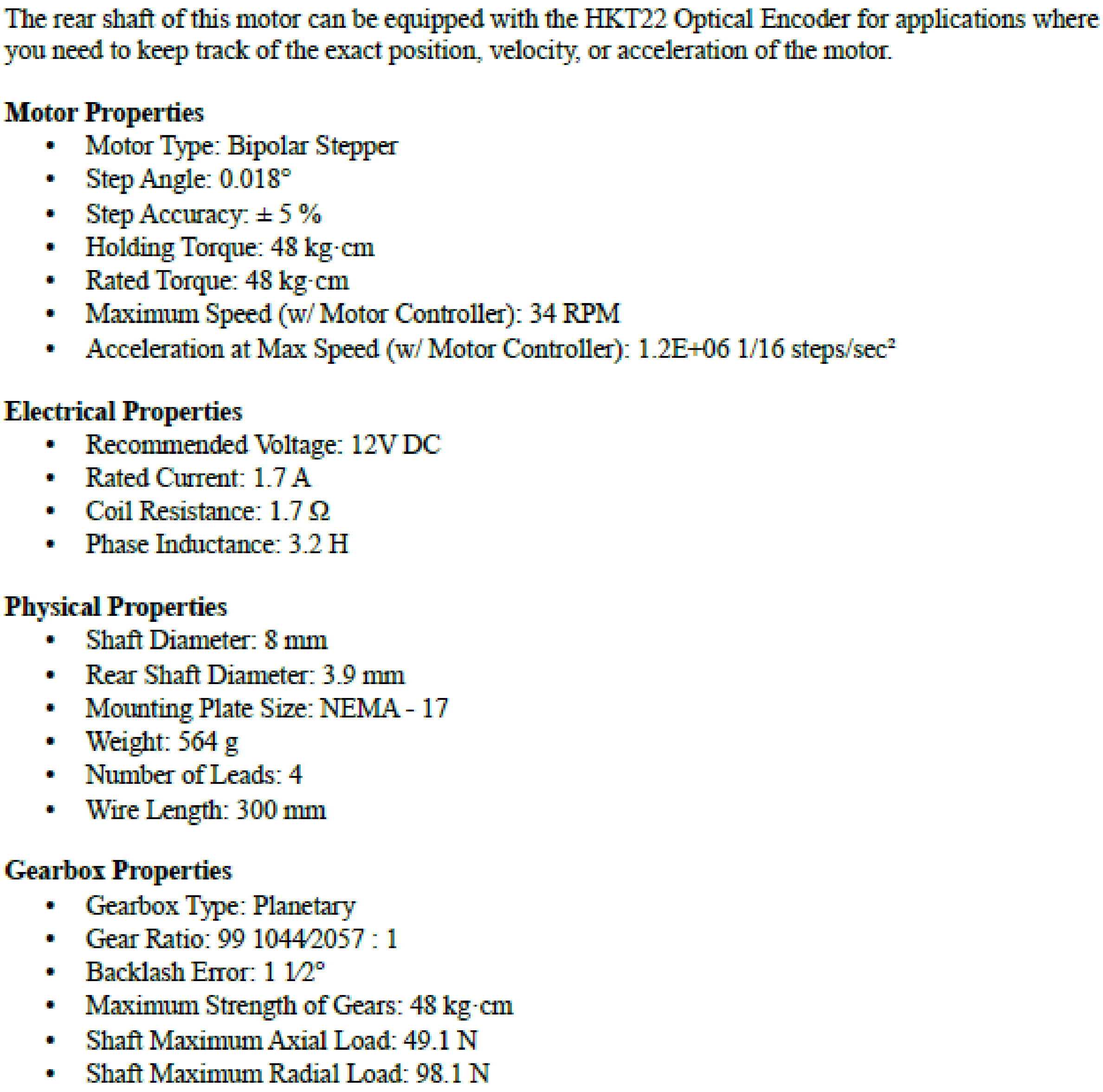
Motor Controller and Connection

The stepper motor connects to a bipolar motor controller such as the Bipolar HC. The following diagram shows how to connect the motor wires to the board connectors to produce a clockwise rotation in the stepper motor when increasing position. To wire for counter-clockwise rotation when increasing position, reverse the red and blue wires.



Note: Make sure to unplug the power cord from the motor controller before switching wires around.

Figure A.5: *12V, 1.7A, 667oz-in NEMA-23 Bipolar Stepper Motor datasheet - 1/2.*



The rear shaft of this motor can be equipped with the HKT22 Optical Encoder for applications where you need to keep track of the exact position, velocity, or acceleration of the motor.

Motor Properties

- Motor Type: Bipolar Stepper
- Step Angle: 0.018°
- Step Accuracy: ± 5 %
- Holding Torque: 48 kg·cm
- Rated Torque: 48 kg·cm
- Maximum Speed (w/ Motor Controller): 34 RPM
- Acceleration at Max Speed (w/ Motor Controller): 1.2E+06 1/16 steps/sec^2

Electrical Properties

- Recommended Voltage: 12V DC
- Rated Current: 1.7 A
- Coil Resistance: 1.7 Ω
- Phase Inductance: 3.2 H

Physical Properties

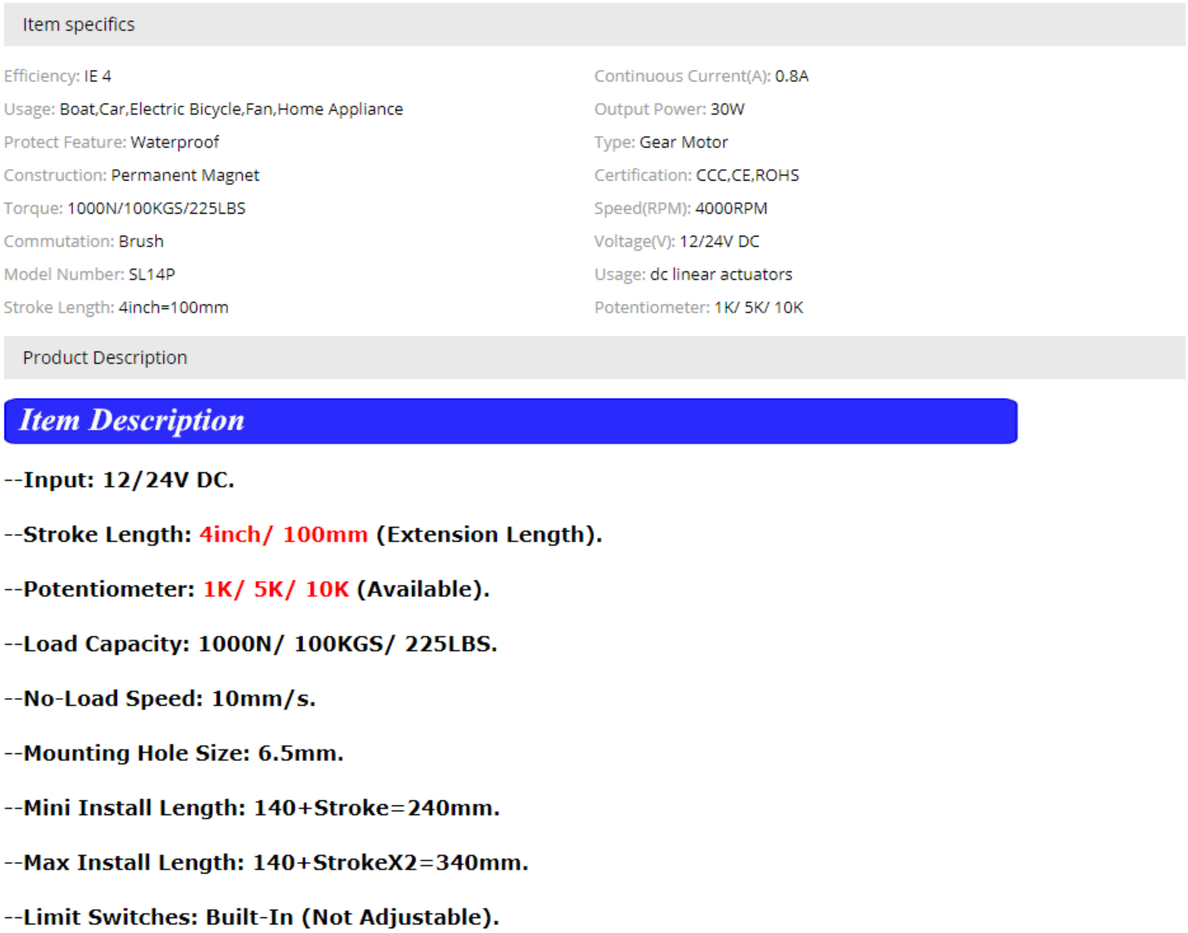
- Shaft Diameter: 8 mm
- Rear Shaft Diameter: 3.9 mm
- Mounting Plate Size: NEMA - 17
- Weight: 564 g
- Number of Leads: 4
- Wire Length: 300 mm

Gearbox Properties

- Gearbox Type: Planetary
- Gear Ratio: 99 1044⁄2057 : 1
- Backlash Error: 1 1⁄2°
- Maximum Strength of Gears: 48 kg·cm
- Shaft Maximum Axial Load: 49.1 N
- Shaft Maximum Radial Load: 98.1 N

Figure A.6: *12V, 1.7A, 667oz-in NEMA-23 Bipolar Stepper Motor datasheet - 2/2.*

A.3 Linear actuator data-sheet



Item specifics

Efficiency: IE 4
Usage: Boat,Car,Electric Bicycle,Fan,Home Appliance
Protect Feature: Waterproof
Construction: Permanent Magnet
Torque: 1000N/100KGS/225LBS
Commutation: Brush
Model Number: SL14P
Stroke Length: 4inch=100mm

Continuous Current(A): 0.8A
Output Power: 30W
Type: Gear Motor
Certification: CCC,CE,ROHS
Speed(RPM): 4000RPM
Voltage(V): 12/24V DC
Usage: dc linear actuators
Potentiometer: 1K/ 5K/ 10K

Product Description

Item Description

--Input: 12/24V DC.

--Stroke Length: 4inch/ 100mm (Extension Length).

--Potentiometer: 1K/ 5K/ 10K (Available).

--Load Capacity: 1000N/ 100KGS/ 225LBS.

--No-Load Speed: 10mm/s.

--Mounting Hole Size: 6.5mm.

--Mini Install Length: 140+Stroke=240mm.

--Max Install Length: 140+StrokeX2=340mm.

--Limit Switches: Built-In (Not Adjustable).

Figure A.7: *Linear actuator data-sheet - 1/2.*

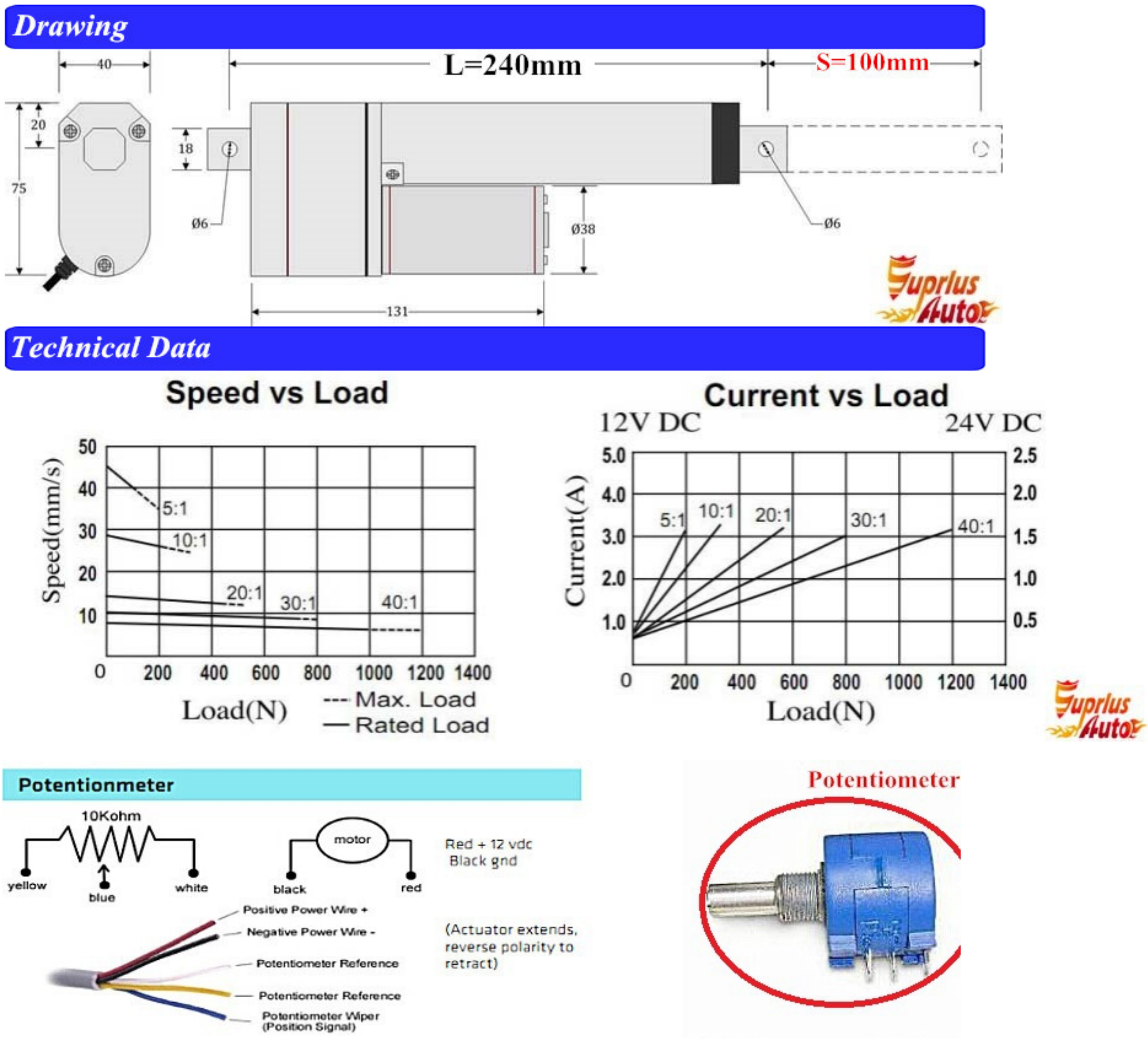
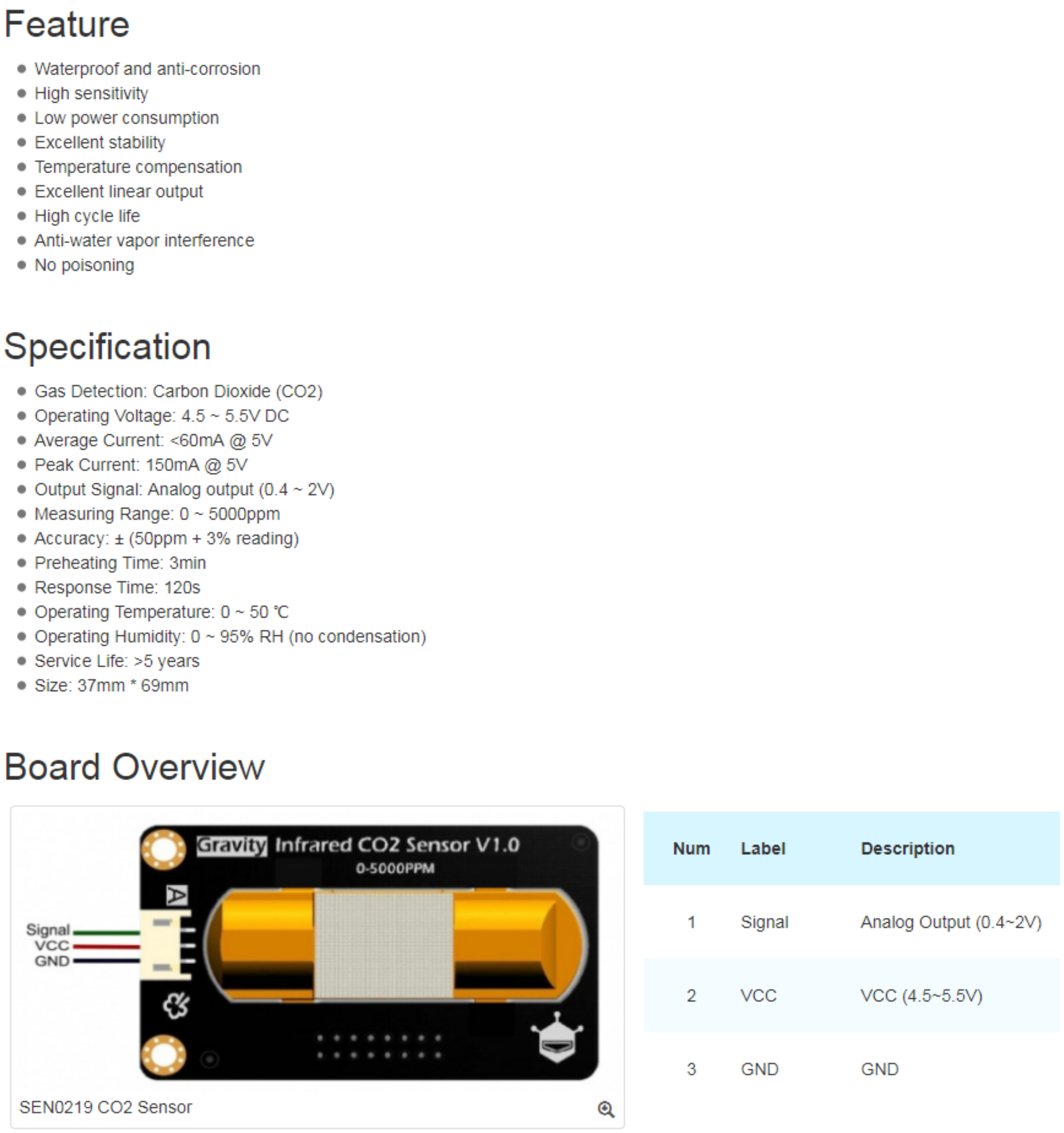


Figure A.8: *Linear actuator data-sheet - 2/2.*

A.4 Sensors



Feature

- Waterproof and anti-corrosion
- High sensitivity
- Low power consumption
- Excellent stability
- Temperature compensation
- Excellent linear output
- High cycle life
- Anti-water vapor interference
- No poisoning

Specification

- Gas Detection: Carbon Dioxide (CO2)
- Operating Voltage: 4.5 ~ 5.5V DC
- Average Current: <60mA @ 5V
- Peak Current: 150mA @ 5V
- Output Signal: Analog output (0.4 ~ 2V)
- Measuring Range: 0 ~ 5000ppm
- Accuracy: ± (50ppm + 3% reading)
- Preheating Time: 3min
- Response Time: 120s
- Operating Temperature: 0 ~ 50 ℃
- Operating Humidity: 0 ~ 95% RH (no condensation)
- Service Life: >5 years
- Size: 37mm * 69mm

Board Overview

SEN0219 CO2 Sensor

Num	Label	Description
1	Signal	Analog Output (0.4~2V)
2	VCC	VCC (4.5~5.5V)
3	GND	GND

Figure A.9: *DF robot carbon dioxide sensor specifications.*

BMP280 Digital Pressure and Temperature Sensor

https://cdn-shop.adafruit.com/datasheets/BST-BMP280-DS001-11.pdf

A.5 Adafruit Data Logger Shield

`https://cdn-learn.adafruit.com/downloads/pdf/adafruit-data-logger-shield.pdf`

A.6 Arduino MEGA 2560

`http://www.mantech.co.za/datasheets/products/A000047.pdf`

A.7 Stepper motor drivers

Stepper Motor Shield For Arduino(DRV8825) SKU:DRI0023

`https://www.dfrobot.com/wiki/index.php/Stepper_Motor_Shield_For_Arduino(DRV8825)_SKU:DRI0023#More`

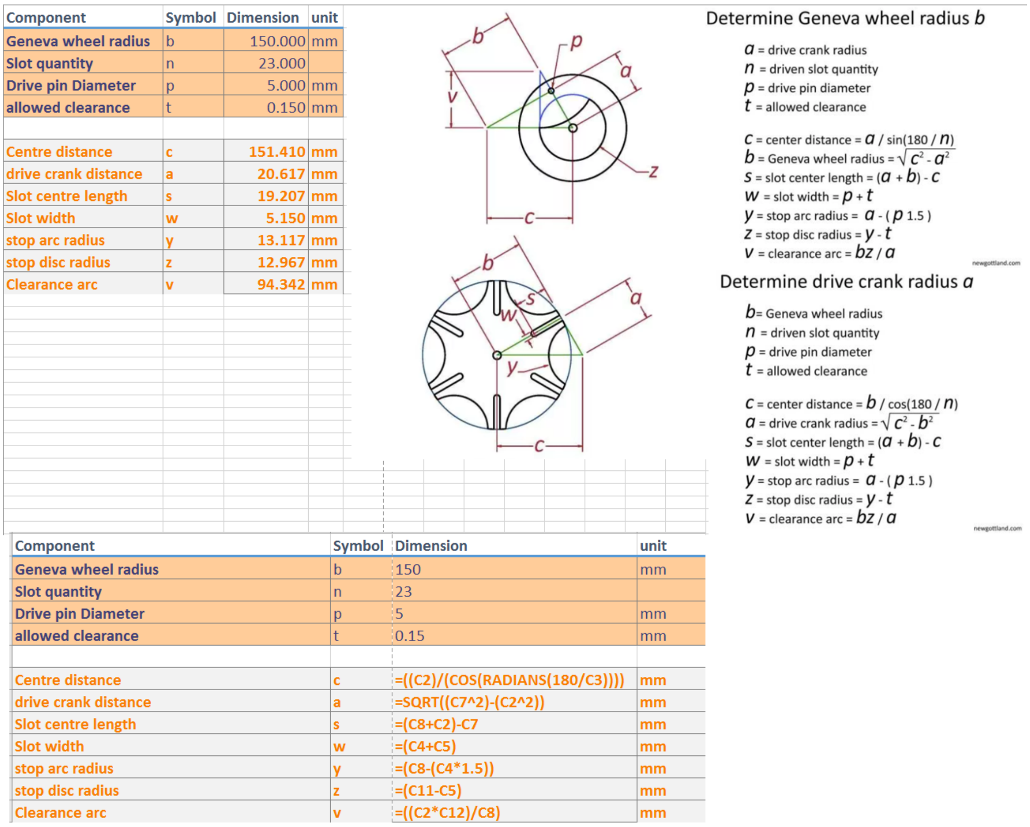
Adafruit Motor/Stepper/Servo Shield for Arduino v2 Kit - v2.3

`https://cdn-shop.adafruit.com/datasheets/TB6612FNG_datasheet_en_20121101.pdf`

Appendix B

B.1 Technical Drawings

Geneva Mechanism



Component	Symbol	Dimension	unit
Geneva wheel radius	b	150.000	mm
Slot quantity	n	23.000	
Drive pin Diameter	p	5.000	mm
allowed clearance	t	0.150	mm
Centre distance	**c**	**151.410**	**mm**
drive crank distance	**a**	**20.617**	**mm**
Slot centre length	**s**	**19.207**	**mm**
Slot width	**w**	**5.150**	**mm**
stop arc radius	**y**	**13.117**	**mm**
stop disc radius	**z**	**12.967**	**mm**
Clearance arc	**v**	**94.342**	**mm**

Determine Geneva wheel radius *b*

a = drive crank radius
n = driven slot quantity
p = drive pin diameter
t = allowed clearance

c = center distance = *a* / sin(180 / *n*)
b = Geneva wheel radius = $\sqrt{c^2 - a^2}$
s = slot center length = (*a* + *b*) - *c*
w = slot width = *p* + *t*
y = stop arc radius = *a* - (*p* 1.5)
z = stop disc radius = *y* - *t*
v = clearance arc = *bz* / *a*

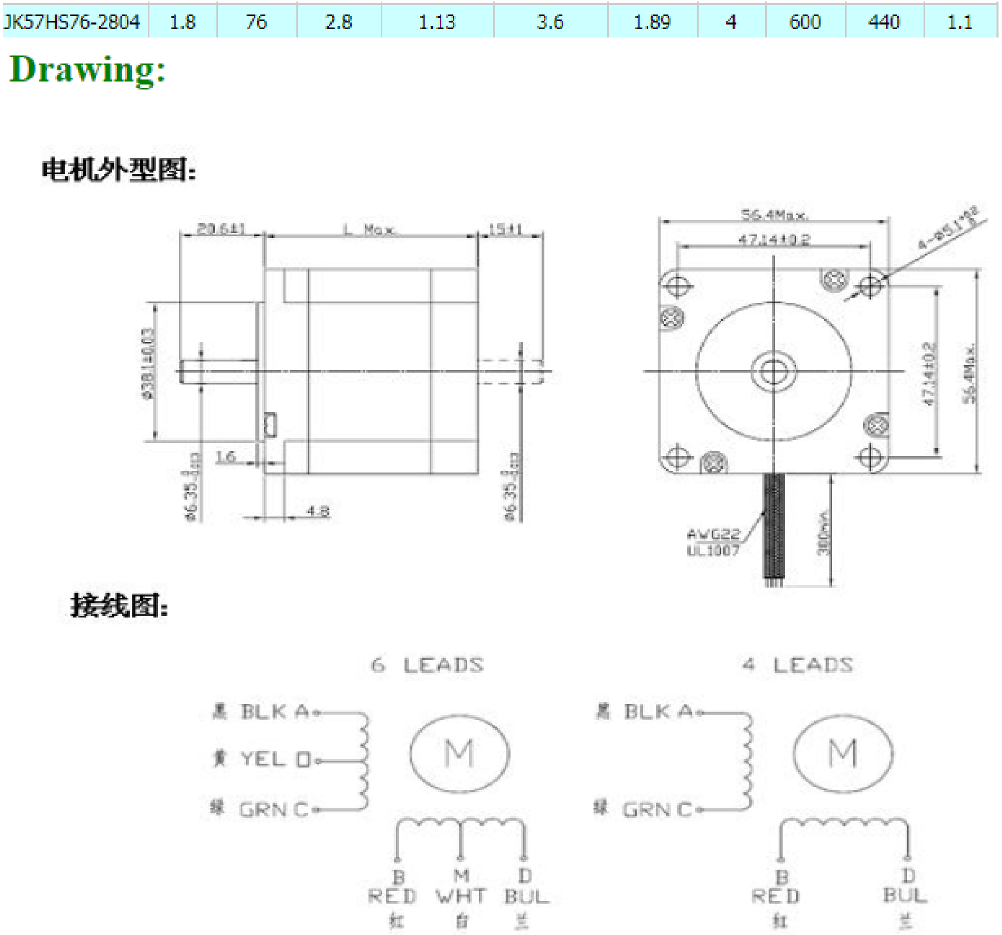
Determine drive crank radius *a*

b= Geneva wheel radius
n = driven slot quantity
p = drive pin diameter
t = allowed clearance

c = center distance = *b* / cos(180 / *n*)
a = drive crank radius = $\sqrt{c^2 - b^2}$
s = slot center length = (*a* + *b*) - *c*
w = slot width = *p* + *t*
y = stop arc radius = *a* - (*p* 1.5)
z = stop disc radius = *y* - *t*
v = clearance arc = *bz* / *a*

Component	Symbol	Dimension	unit
Geneva wheel radius	b	150	mm
Slot quantity	n	23	
Drive pin Diameter	p	5	mm
allowed clearance	t	0.15	mm
Centre distance	**c**	**=((C2)/(COS(RADIANS(180/C3))))**	**mm**
drive crank distance	**a**	**=SQRT((C7^2)-(C2^2))**	**mm**
Slot centre length	**s**	**=(C8+C2)-C7**	**mm**
Slot width	**w**	**=(C4+C5)**	**mm**
stop arc radius	**y**	**=(C8-(C4*1.5))**	**mm**
stop disc radius	**z**	**=(C11-C5)**	**mm**
Clearance arc	**v**	**=((C2*C12)/C8)**	**mm**

Figure B.1: *Geneva drive calculations, [8].*



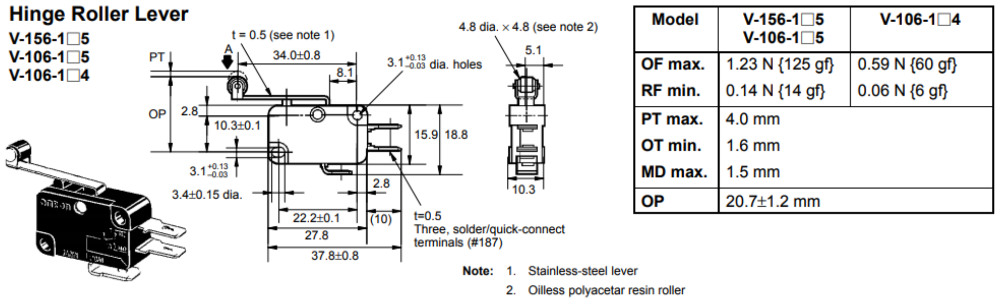
JK57HS76-2804	1.8	76	2.8	1.13	3.6	1.89	4	600	440	1.1

Drawing:

电机外型图:

接线图:

__Figure B.2:__ NEMA23 JK57HS76-2804 Stepping Motor drawing.



Hinge Roller Lever

V-156-1□5
V-106-1□5
V-106-1□4

Model	V-156-1□5 V-106-1□5	V-106-1□4
OF max.	1.23 N {125 gf}	0.59 N {60 gf}
RF min.	0.14 N {14 gf}	0.06 N {6 gf}
PT max.	4.0 mm	
OT min.	1.6 mm	
MD max.	1.5 mm	
OP	20.7±1.2 mm	

Note:
1. Stainless-steel lever
2. Oilless polyacetar resin roller

__Figure B.3:__ SPDT Long Roller Lever Microswitch, 16 A @ 250 V ac technical specifications.

Appendix C

C.1 Quotations

Service <service@purescience.co.nz>
to me

Rahul the approximate prices for the tubes would be:

50x @ about $1000.00/lot

200x @ about $3300.00/lot

I don't know exactly about the freight costs yet – still checking - but these figures will be fairly accurate.

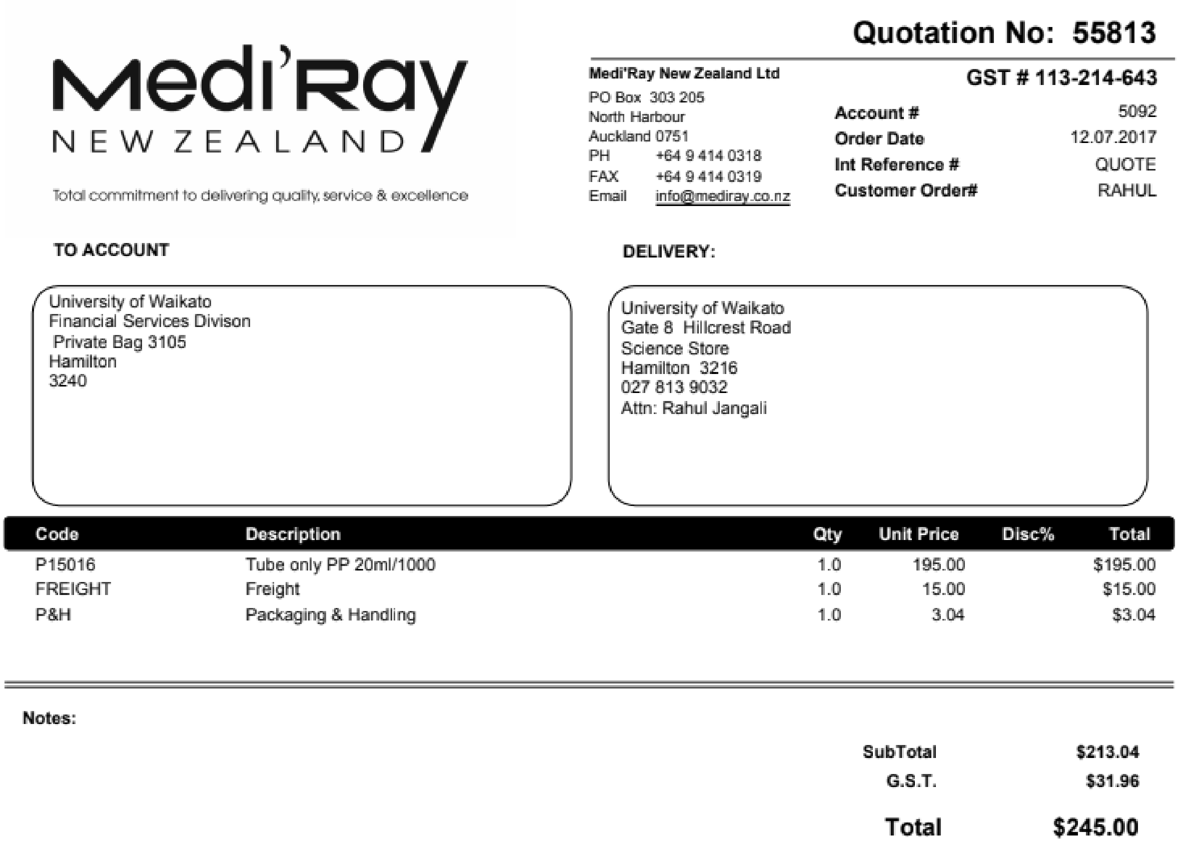
Cheers
Hugh
15.06.17

Hugh Blank Manager

Pure Science Ltd *ISO 9001:2015 Certified by Telarc SAI Limited*
p: +64 4 233 6308 f: +64 4 233 6310 w: www.purescience.co.nz

Unless otherwise specifically stated, all prices we quote are ex store Wellington, in NZ Dollars, exclusive of GST.

Figure C.1: *PTFE sample vial price.*



Medi'Ray
NEW ZEALAND
Total commitment to delivering quality, service & excellence

Quotation No: 55813

Medi'Ray New Zealand Ltd
PO Box 303 205
North Harbour
Auckland 0751
PH +64 9 414 0318
FAX +64 9 414 0319
Email info@mediray.co.nz

GST # 113-214-643

Account #	5092
Order Date	12.07.2017
Int Reference #	QUOTE
Customer Order#	RAHUL

TO ACCOUNT

University of Waikato
Financial Services Divison
Private Bag 3105
Hamilton
3240

DELIVERY:

University of Waikato
Gate 8 Hillcrest Road
Science Store
Hamilton 3216
027 813 9032
Attn: Rahul Jangali

Code	Description	Qty	Unit Price	Disc%	Total
P15016	Tube only PP 20ml/1000	1.0	195.00		$195.00
FREIGHT	Freight	1.0	15.00		$15.00
P&H	Packaging & Handling	1.0	3.04		$3.04

Notes:

SubTotal	**$213.04**
G.S.T.	**$31.96**
Total	**$245.00**

Figure C.2: *Polypropylene sample vial quotation.*

Rahul Jangali <rsj1@students.waikato.ac.nz>
to Laser

Hi There,

Can i please get a quote request for the attached DXF's. All materials in 12mm acrylic.

Flat pattern - LaserCutBase - 2 off
Flat pattern - Legs - 2 off

Cheers

2 Attachments

Laser Profiles <laserprofiles.nz@gmail.com>
to me

Hi Rahul,

$80

Kind regards, Andreas

Figure C.3: *Quotation for manufacture of the platform.*

Appendix D

Arduino Code

```
/*
This is a test sketch for the Adafruit assembled Motor Shield for Arduino v2
It won't work with v1.x motor shields! Only for the v2's with built in PWM control

For use with the Adafruit Motor Shield v2
---->  http://www.adafruit.com/products/1438
*/

#include <Wire.h>
#include <Adafruit_MotorShield.h>
#include "utility/Adafruit_MS_PWMServoDriver.h"

// Create the motor shield object with the default I2C address
//Adafruit_MotorShield AFMS = Adafruit_MotorShield();
// Or, create it with a different I2C address (say for stacking)
Adafruit_MotorShield AFMS = Adafruit_MotorShield(0x60);

// Select which 'port' M1, M2, M3 or M4. In this case, M1
Adafruit_DCMotor *myMotor = AFMS.getMotor(1);
// You can also make another motor on port M2
//Adafruit_DCMotor *myOtherMotor = AFMS.getMotor(2);


const int ActuatorSensorPin = 0; // input pin for the potentiometer.

int ActuatorSensorValue = 0; // Variable to store the value coming from the sensor.


int goalPosition;
int currentPosition = 0;
int extractPosition = 360; //Height of bottom of the stab cycle
int TopPosition = 700; //Max height of actuator. Home position
int ActuatorHomeThreshold = 10;
boolean Extend = false;
boolean Retract = false;










#include <Arduino.h>
#include "BasicStepperDriver.h"
#include "SyncDriver.h"

// Motor steps per revolution. Most steppers are 200 steps or 1.8 degrees/step
#define MOTOR_STEPS 200

// X motor
#define DIR_X 7
#define STEP_X 6
#define ENBL_X 8


// Y motor
#define DIR_Y 4
#define STEP_Y 5
#define ENBL_Y 12

// If microstepping is set externally, make sure this matches the selected mode
// 1=full step, 2=half step etc.
#define MICROSTEPS 1
```

```
// 2-wire basic config, microstepping is hardwired on the driver
// Other drivers can be mixed and matched but must be configured individually
BasicStepperDriver stepperX(MOTOR_STEPS, DIR_X, STEP_X, ENBL_X );
BasicStepperDriver stepperY(MOTOR_STEPS, DIR_Y, STEP_Y,ENBL_Y );

SyncDriver controller(stepperX, stepperY);



// variables will change:
int buttonState = 0;         // variable for reading the pushbutton status

int State = 0; //The Current state of the state machine
int buttonCount = 0;
int offsetSteps = 1400;
const int buttonPin = 25;


void setup() {
  Serial.begin(9600);           // set up Serial library at 9600 bps
  Serial.println("Adafruit Motorshield v2 - DC Motor test!");

  AFMS.begin();  // create with the default frequency 1.6KHz
  //AFMS.begin(1000);  // OR with a different frequency, say 1KHz

  // Set the speed to start, from 0 (off) to 255 (max speed)
  myMotor->setSpeed(110);
  //myMotor->run(FORWARD);
  // turn on motor
  myMotor->run(RELEASE);
  //myMotor->run(BACKWARD);




  /*
     * Set target motors RPM.
     */
    stepperX.begin(150, MICROSTEPS);
    stepperY.begin(200, MICROSTEPS);
    stepperX.disable();
    stepperY.disable();
}







void loop() {



     /*Turn table main drive code below
 *
 *
 *
 */

switch(State)
{

case 0:
//Home for actuator
currentPosition = analogRead(ActuatorSensorPin);
  //home
```

```
 if(currentPosition < TopPosition || currentPosition > (TopPosition +
ActuatorHomeThreshold)){

    if((TopPosition-currentPosition)>0){myMotor->run(FORWARD);}else{
myMotor->run(BACKWARD);} //Runs forward if we are under and back if we are above

    Serial.print(currentPosition);
    Serial.println("");
 }
 else
 {
   myMotor->run(RELEASE);
   State=100; //goes offset
 }


    break;

case 100:  //Home

  stepperY.enable();
  stepperY.move(1);

  if(digitalRead(buttonPin)==0){buttonCount++;}else{buttonCount = 0;}
  if(buttonCount >= 5)
  {
      stepperY.disable();
      State = 1; //Homing actuator
  }

break;

case 1:  //Offset
  delay(3000);
  stepperY.enable();
  stepperY.move(offsetSteps);
  State = 101;
  break;

case 101: //Idle (waiting for next index)
  delay(5000);
  State = 2; //Index the next position
  break;

case 2: //Spin

  /*
  * Linear acturator arm rotation code below
  *
  */
      //   myActuatorUp(); // goes up
      //   Serial.print(currentPosition);
     //    Serial.println("");


  stepperY.enable();
  stepperY.move(797);
  stepperY.disable();
  State = 102;

    //   myActuatorDown();// comes down
   //  Serial.print(currentPosition);
   //   Serial.println("");
  break;

case 102: //Actuator Down
  delay(1000);
  myActuatorDown();
```

```
  State = 104;
  break;

case 104: //Filling
  delay( );
  State = 103;
  break;

case 103: //Actuator Up
  delay(1000);
  myActuatorUp(); //Will get stuck here while the actuator moves down.
  State = 101;
  break;
}

}




//--------------------------------------------------- Actuatorup
-------------------------------------------------------------
void myActuatorUp(){
   //retract
  currentPosition = analogRead(ActuatorSensorPin);
  while(currentPosition < TopPosition){
  myMotor->run(FORWARD);
  currentPosition = analogRead(ActuatorSensorPin);
 }
  myMotor->run(RELEASE);
}




//--------------------------------------------------- ActuatorDown
-------------------------------------------------------------
void myActuatorDown(){
   //retract
  currentPosition = analogRead(ActuatorSensorPin);
  while(currentPosition > extractPosition){
  myMotor->run(BACKWARD);
  currentPosition = analogRead(ActuatorSensorPin);
 }
  myMotor->run(RELEASE);
}
//------------------------------------------------------------------------------
-----------------------------------------------------

void myStepperGeared () {


delay(1000);
stepperY.enable();
stepperY.move(798);


stepperY.disable();

delay(500);



}


//--------------------------------------------------- Sleep
```

```
#include <avr/interrupt.h>
#include <avr/power.h>
#include <avr/sleep.h>
#include <avr/io.h>
#include "RTClib.h"


RTC_PCF8523 rtc;


int interruptPin = 2;             // digital pin 2 used for interrupt
int LedPin = 52;
int interruptChannel;

void setup(void)
{
   Serial.begin(57600);
    interruptChannel = digitalPinToInterrupt(interruptPin);
    Serial.print(interruptChannel);
    Serial.println();
    pinMode(interruptPin,INPUT_PULLUP);      // set pin 2 as input with pull up
enabled
    pinMode(LedPin,OUTPUT);            // set pin 3 as an output so we can use
LED to monitor
    digitalWrite(LedPin,HIGH);         // turn pin 3 LED on

  /* rtc */
 if (! rtc.begin()) {
    Serial.println("Couldn't find RTC");
    while (1);
  }

  if (! rtc.initialized() ){
    Serial.println("RTC is NOT running!");
    // following line sets the RTC to the date & time this sketch was compiled
    rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
    Serial.println("Time Set");
    // This line sets the RTC with an explicit date & time, for example to set
    // January 21, 2014 at 3am you would call:
    // rtc.adjust(DateTime(2014, 1, 21, 3, 0, 0));
  }

  rtc.enableAlarms();

}

void loop(void)
{
    // Stay awake for 1 second, then sleep.
    // LED turns off when sleeping, then back on upon wake.


    delay(10000);

    //rtc.setAlarm(MINUTE, 1); // 1 minute
    rtc.setAlarm(HOUR, 72); // 3 days

    sleepNow();

}

void sleepNow()
{
    // interruptChannel is the channel triggered by digital pin 2
    // pinInterrupt is name of function called when interrupt signal occurs
    // LOW is the polarity of the interrupt signal (RTC interrupt goes LOW when
alarm)
```

```
    // need delay to allow arduino to configure interrupt function
    delay(100);
    // Choose our preferred sleep mode:
    set_sleep_mode(SLEEP_MODE_PWR_DOWN);
    // Set sleep enable (SE) bit:
    sleep_enable();
    attachInterrupt(interruptChannel, pinInterrupt, LOW);  // now digital pin 2
will trigger interrupt
    // Put the device to sleep:
    digitalWrite(LedPin,LOW);   // turn LED off to indicate sleep
    Serial.print("LED oFF");
    /* set alarm here */



    sleep_mode();          // now asleep
    /* clear alarm here */

    rtc.clearAlarm();

    // Only interrupt signal will wake arduino
    // the interrupt will trigger pinInterrupt handler function (below)
    // pinInterrupt function detaches interrupt
    // Upon waking up, sketch continues from this point.
    sleep_disable();
    digitalWrite(LedPin,HIGH);   // turn LED on to indicate awake
    Serial.println();
    Serial.print("LED ON");
    Serial.println();
}

void pinInterrupt(void)
{
    detachInterrupt(interruptChannel);   // now digital pin 2 will not trigger
any more interrupts
}

/***************************************************
 * Infrared CO2 Sensor0-5000ppm
 * ***************************************************
 * This example The sensors detect CO2
 *
 * @author lg.gang(lg.gang@qq.com)
 * @version  V1.0
 * @date  2016-7-6
 *
 * GNU Lesser General Public License.
 * See <http://www.gnu.org/licenses/> for details.
 * All above must be included in any redistribution
 * ***************************************************/
int sensorIn = 3;

void setup(){
  Serial.begin(9600);
  // Set the default voltage of the reference voltage
  analogReference(DEFAULT);
}

void loop(){
  //Read voltage
  int sensorValue = analogRead(sensorIn);

  // The analog signal is converted to a voltage
  float voltage = sensorValue*(5000/1024.0);
  if(voltage == 0)
  {
    Serial.println("Fault");
```

```
  }
  else if(voltage < 400)
  {
    Serial.println("preheating");
  }
  else
  {
    int voltage_diference=voltage-400;
    float concentration=voltage_diference*50.0/16.0;
    // Print Voltage
    Serial.print("voltage:");
    Serial.print(voltage);
    Serial.println("mv");
    //Print CO2 concentration
    Serial.print(concentration);
    Serial.println("ppm");
  }
  delay(500);
}


/***************************************************************************
  This is a library for the BMP280 humidity, temperature & pressure sensor

  Designed specifically to work with the Adafruit BMEP280 Breakout
  ----> http://www.adafruit.com/products/2651

  These sensors use I2C or SPI to communicate, 2 or 4 pins are required
  to interface.

  Adafruit invests time and resources providing this open source code,
  please support Adafruit andopen-source hardware by purchasing products
  from Adafruit!

  Written by Limor Fried & Kevin Townsend for Adafruit Industries.
  BSD license, all text above must be included in any redistribution
 ***************************************************************************/

#include <Wire.h>
#include <SPI.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_BMP280.h>

#define BMP_SCK 13
#define BMP_MISO 12
#define BMP_MOSI 11
#define BMP_CS 10

Adafruit_BMP280 bmp; // I2C
//Adafruit_BMP280 bmp(BMP_CS); // hardware SPI
//Adafruit_BMP280 bmp(BMP_CS, BMP_MOSI, BMP_MISO,  BMP_SCK);

void setup() {
  Serial.begin(9600);
  Serial.println(F("BMP280 test"));

  if (!bmp.begin()) {
    Serial.println(F("Could not find a valid BMP280 sensor, check wiring!"));
    while (1);
  }
}

void loop() {
    Serial.print(F("Temperature = "));
    Serial.print(bmp.readTemperature());
    Serial.println(" *C");

    Serial.print(F("Pressure = "));
```

```
    Serial.print(bmp.readPressure());
    Serial.println(" Pa");

    Serial.print(F("Approx altitude = "));
    Serial.print(bmp.readAltitude(1013.25)); // this should be adjusted to your 
local forcase
    Serial.println(" m");

    Serial.println();
    delay(2000);
}
```