Linköping University | Department of Management and Engineering
Bachelor’s thesis, 18 credits| Mechanical Engineering
Spring 2024| LIU-IEI-TEK-G--24/02424--SE

Further development of the Green Revolution Energy Converter

Investigating improved control method, sealing and overall performance of LabModel v3

Erik Widström
Richard Zetterman
Simon Eriksson

Supervisor: Johan Hedbrant, Research engineer

Project owner: Johan Renner, Assistant professor

Examiner: Jonas Detterfelt, Assistant professor

Linköping University
SE-581 83 Linköping, Sweden
+46 013 28 10 00, www.liu.se

Abstract

One of the most pressing global issues facing humanity today is global warming. To help fight this problem, an innovative technology called GREC, Green Revolutionary Energy Converter, utilizes the wasted heat in industries to convert it to mechanical work, by rotating a shutter between a heated and cooled zone to create a pressure difference, which in turn can produce mechanical movement. This innovative technology is meant to be scaled and implemented in different applications, such as large industries to make use of excess heat energy.

Multiple previous projects have researched the GREC and developed a working model, the LabModel v3. The purpose of this project is to further develop the previous work that has been done on the GREC LabModel v3. The main objectives this report will bring up is to identify and seal possible leakage to ensure that the model is pressure tight and to choose and implement a method to measure the power output. Another objective was to make LabModel v3 more user-friendly.

The new control method was overseen first, where a Raspberry Pi was implemented to control the stepper motor more easily with the help of a local website on the microcomputer. This dictated how fast the rotating shutter in the LabModel v3 would rotate. A step-by-step instruction manual was also created to make future work easier and to make sure different people can operate the LabModel v3 with ease.

Multiple tests were conducted when the LabModel v3 was declared pressure tight, the purpose of the test is to prove that the LabModel v3 is repeatable and that the pressure inside GREC is the same in all parts of it. Furthermore, the power output was investigated using a reverse speaker and liquid column test.

The results show that the pressure difference created by the temperature difference between the hot- and cold side was sufficient to produce work. The power output that the LabModel v3 produce is not considered big but the results show a proof of concept. To scale the model and find improved ways of extracting the power output is for future work to investigate. This to further improve the chances of GREC being a part of industries to utilize waste heat and help decrease global warming.

Distribution of Work

Table 1. The distribution of work during the project.

Activity	**Erik**	**Simon**	**Richard**	**Comment**
Planning [%]	33	33	33	Planning was done collectively.
Research [%]	33	33	33	Research was done collectively.
Work				
Identifying Leakage [%]	40	30	30	Identifying leakage was done collectively.
Sealing [%]	25	50	25	Simon had the main responsibility.
Code [%]	40	10	50	Richard had the main responsibility.
Wiring [%]	20	60	20	Simon had the main responsibility.
Instruction Manual [%]	35	20	45	Richard had the main responsibility.
Method to calculate power output [%]	70	10	20	Erik had the main responsibility.
Implementation of power method [%]	33	33	33	Implementation was done collectively.
Writing				
Abstract [%]	0	60	40	Simon and Richard wrote abstract together.
Introduction [%]	40	30	30	The introduction was written together.
Theory [%]	40	25	35	Erik had the main responsibility.
Method [%]	20	40	40	Simon and Richard had main responsibility.
Results [%]	25	50	25	Simon had most responsibility regarding data and graphs.
Discussion [%]	60	15	25	Erik had the main responsibility.
Conclusion [%]	60	15	25	Erik had the main responsibility.

Contents

Nomenclature

GREC	–	Green Revolution Energy Converter
RS	–	Rotating Shutter
WGV	–	Work Generating Volume
FRP	–	Fiber Reinforced Plastic

1. Introduction

This section will cover the background and technology of the Green Revolution Energy Converter, GREC. It will also include previous work done regarding GREC.

1.1 Background

Climate change due to carbon emissions is an urgent topic today. A big contributor is the use of fuel for electricity generation. GREC is a new type of heat engine that is supposed to convert waste heat into electricity. There are also other applications for the GREC such as the transportation sector and conversion from solar to motion.

Nilsinside AB is a company with sustainable innovation, development and research as business idea. GREC is a new type of external heat engine and can be described as a revolving Carnot engine controlled by computer logic. An electric motor moves air between a cold and hot place. The pressure difference which occurs when the gas is heated and cooled will result in output work greater than the work required for the electric motor. According to nilsinside AB “This is a highly scalable, low cost, zero carbon engine that makes the GREC an outstanding competitive solution.”. All company operations are currently based on the GREC project. The technology readiness is between 3-4 on a scale between 1 and 9. The next step is to prove that the theory behind GREC also works practically. [1]

1.2 Technology behind GREC

The GREC is a Carnot engine using an electric motor to move its Work Generating Volume, WGV, between a cold and hot reservoir. The GREC moves the WGV by rotating a Revolving Shutter, RS using a stepper motor. As the RS rotates the WGV is either heated or cooled depending on which reservoir it is in contact with. The cold and hot reservoir is on opposite sides of the GREC with insolating nil-fins in-between. The nil-fins are used to insulate the cold and hot fins and prevent the WGV from being cooled and heated simultaneously. When the WGV is heated and cooled the internal pressure of the engine increases and decreases. These pulses of change in pressure can be extracted as mechanical work.

The GREC is made up of several layers with the air between them working as the WGV. This way of building the GREC makes it highly scalable and to increase the WGV another layer of fins and RS can be added, this can be viewed in figure 1. All RS that are fixed on the same axis are not in contact with the fins so the RS moves freely and the WGV uses the friction of the gas flow to enhance its heat transfer coefficient. [2]

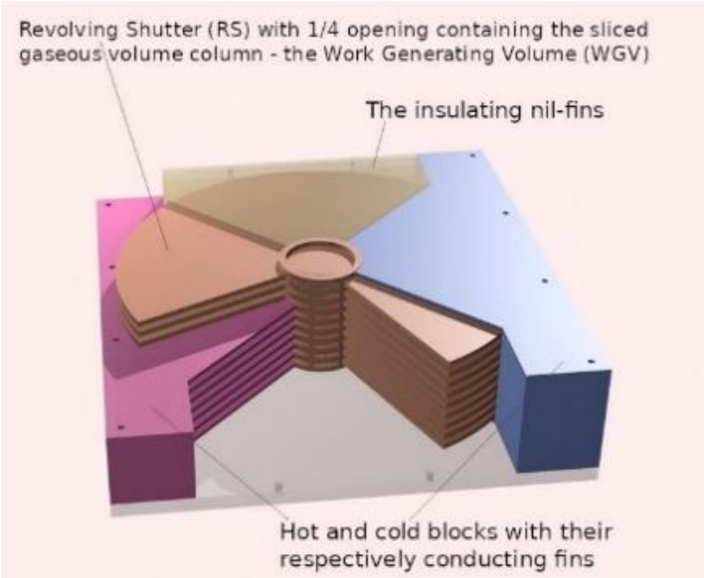


Figure 1. Cut out of GREC, Shows the engines layered design making it scalable. [2]

1.3 Previous work

The GREC engine was studied in six previous projects. These six projects were all conducted by groups of students attending Linköping University and three of them were bachelor theses. The six projects are listed below:

1. *Theoretical Proof of Concept for The Green Revolution Energy Converter: Development of a mathematical model, material analysis and physical model improvements.* [3]
2. *Investigation of the internal heat transfer of GREC.* [4]
3. *Thermal Investigation of the Green Revolution Energy Converter: A study on the heat transfer within the GREC in regards of temperature distribution and heat rate.* [5]
4. *The thermodynamics of the GREC version 3*. [6]
5. *Development of a mechatronic solution for a new type of heat engine: The selection of relevant electronic components and control system for the mechatronics for the lab model of the new heat engine the Green Revolution Energy Converter.* [7]
6. *Design and development of a working prototype for a revolving heat engine*. [8]

The first project was carried out in spring of 2022 by five students from the Department of Management and engineering. In this project, the aim was to find out and present proof of concept regarding the GREC engine and to further develop an understanding of the new technology. They used a mathematical model with the help of thermodynamic assumptions, simplifications, equations and the principle of the Carnot engine, which the GREC is based on. Their results show that the concept is viable and that a larger scale and larger temperature differences improves its performance. They also found out that the materials the engine is built with can improve the performance regarding conductivity.

The second project was done in fall of 2022 in course TMPE09. This project studied how the design of the engine affected its performance regarding the pressure difference, work output and how turbulent versus laminar flow internally affects the performance. The results show that greater pressure differences are achieved by a slower rotating speed of the RS and a thin WGV. They also concluded that turbulent flow inside the WGV is necessary for the work output to be as high as possible. Regarding the design, they also found out that the RS cut-out affected the temperature difference, where the 1/8 cut-out increased the difference instead of the 1/4.

The third project was done by five students from the Department of Management and engineering in fall of 2022. This project investigated how the conducting fins should be externally heated. They found out that heating the fins with pipe heat transfer was better than none pipe heat transfer at larger diameters.

Based on the results of the above-mentioned work, a large group consisting of nine students, divided into three groups of three students were set to make the physical prototype LabModel v3 as their bachelor's theses during spring of 2023. The groups each had a separate aim and results, described below.

The first of these projects (no. 4 in the list above) aimed to investigate how the work output should be taken care of, finding optimal piston dimensions and how to calculate the power and work generated. The results of this project show that three different piston sizes of 5, 10 and 20 CC should be compared in future testing to determine what is the optimal piston size. They also concluded that the LabModel v3 had leakage and that the pressure differences measured during tests were non-conclusive.

The second project of these three (no. 5 In the list above) investigated what mechanical components, consisting of motor and sensors, should be used, how the wiring should be set up and what code should run the complete prototype. They found out that a stepper motor "Nema 23" was the best fit as it can handle rotations up to 1000 RPM and a minimum torque of 0.5 Nm at 500 RPM. The sensors used are: 1 – Hall sensors fast enough to handle 500 RPM and 2 – pressure sensors manage pressure differences of 5 kPa with a safety factor of 10 and have an update frequency of 10 Hz or more. These sensors were also required to be easily incorporated into the LabModel v3 without leakage occurring. They also concluded that temperature sensors would not be able to fit in the LabModel v3 without causing damage. A functioning code was also written.

The third project (no. 6 In the list above) was the project that covered how the LabModel v3 was built. They based their design mainly on previous prototypes and succeeded in constructing a model that met expectations but with some limitations. The model they constructed consisted of one layer of RS with a WGV taking up 1/4 of the RS. The heating fins are made of aluminum and the shell of FRP. There were difficulties that could not be completely ruled out which consisted of reducing dead volume, minimizing friction and eliminating air leakage.

1.4 Given Components

The components given from previous work are listed below:

- 5x SparkFun MicroPressure Sensors
- 2x Hot plates
- 4x Hall sensors
- 2x Arduino UNO
- 1x Arduino MEGA
- 2x small breadboards
- 1x Stepper motor SM2564C60B41 NEMA 23 1.8°
- 1x Stepper motor driver DRV8255
- Multiple jumper wires of different lengths and connections

Furthermore, Nils Karlberg, the founder of nilsinside AB, had been working on the LabModel v3 during summer 2023, prior to this project and left the following components to be used during this project:

- 1x Raspberry Pi 3
- Additional electrical components such as a 12 V power supply for the stepper motor, extension cord, a keyboard, mouse and a monitor as well as various power cables for the different microcomputers and microcontrollers.

1.5 Project goals

At the start of the project the LabModel v3 had flaws. As mentioned earlier, the actual LabModel v3 did not meet the ideal standards set by previous groups. The GREC experienced significant pressure leakage and lacked a clear method for measuring the generated work.

Consequently, the project aims to enable repeated operation of the LabModel v3 and eliminate any potential leakage that may occur. When this is achieved, the new objective is to choose a method for measuring the generated work.

1.6 Long-term goals

The long-term goal of the GREC is to reduce carbon emissions worldwide. Since the GREC has a modular design, it can be used in many different key applications. Its main area of application is usage of waste heat. By increasing the number of layers in the RS of the GREC a large WGV can be obtained. This ensures a large power output despite a small temperature gradient between the cold and hot reservoir. If a small and compact design is desired, the GREC can provide a solution for that too. Where there are high temperatures differences, the GREC can be made compact, making it suitable for the transporting sector.

The long-term goal of this current project is to calculate the actual power output of the GREC more accurately. Long term, this provides concrete answers to the real climate and environmental benefits of the GREC and shows its ability to be scaled and more efficient.

1.7 Budget

An initial budget of 5000 SEK is set. This is believed to be enough for this project since many of the expensive purchases were made by the previous year's project. If an expanded budget is required, it should be brought up with the examiner and/ or the project owner with specifications of what will need to be bought.

1.8 Research questions

The research questions that the project aims to answer are the following:

Q1: How to optimize the electronics in the GREC LabModel v3 and make GREC LabModel v3 user-friendly?

Q2: How do you identify pressure leakage in the GREC LabModel v3? How do you seal the leakage so the LabModel v3 withstands a pressure of 3 kPa and heat of 200 degrees Celsius?

Q3: How does the pressure of GREC LabModel v3 change with temperature and rotational frequency and how is this related to the power output?

Q4: What is a method for measuring power output for GREC LabModel v3?

Q5: Is the pressure inside GREC LabModel v3 homogeneous?

1.9 Delimitations

The project continues the work that was done in the previous projects of the GREC. Therefore, this project will be limited to the decisions that have already been made. The original concept of the engine has a modular design, where one module consists of the shutters and heating/ cooling fins and are layered upon each other, as seen in figure 1. However, the previous project limited themselves to a one-layer engine to simplify calculation and lower the requirements of the motor. The LabModel v3 is designed to show proof of concept, it does not have to have a positive efficiency at this stage.

Another limitation is that the materials used to build Lab Model v3 are not optimal. Since the previous groups had to limit themselves in producing a prototype of cheap and easily obtainable materials.

1.10 Ethical Reflection

Ethical reflections are required for the GREC. The main purpose of GREC is to lower carbon emissions worldwide in different applications. Compared to other engines, GREC can be run by waste heat. While this is positive, the source of the waste heat must be taken into consideration. If the waste heat is created by biomass, coal or oil, the environment could be negatively impacted by deforestation or carbon emissions. If a powerplant uses GREC as a collector of waste heat it will still be more efficient than a powerplant that does not use GREC.

There might also be some concerns regarding material acquisition and disposal. The GREC is only ethically viable if the saved waste heat emissions are greater than the energy and emissions needed to create, maintain and dispose of the GREC. The impact of GREC on society will depend on how it is implemented.

2. Theory

To develop the GREC LabModel v3, certain fields needed to be studied further. These fields will be described in the following chapter and implemented in the development of the GREC.

2.1 Sealants

During this section, the different sealants that were considered will be presented.

O-ring is the most used type of sealant and comes in standard sizes. They are cheap and easy to install. They can be used in high temperature and pressure applications. O-rings are mostly designed out of rubber and can be used in dynamic and static applications. [9]

Polymer based sealants are ideal for moving or expansion joints. It creates a permanent flexible seal that cures in room temperature. These sealants require a long curing time compared to the rest of the sealants.

Epoxy sealants are often used for application where toughness and strength are important. Epoxy sealants consist of a hardener and a resin that must be mixed before application. These sealants are often cured at room temperature, but thermosetting epoxies might need additional heat to harden. [10]

Epoxies have exceptional adhesive properties and provide electrical insulation. Specialty epoxies enhance chemical residence, cure speed, UV resistance and thermal performance. [11]

Silicon based sealants are a strong, versatile and affordable sealant that has a life expectancy of 10 to 20 years. It exists as neutral cure-based sealants and acetoxy-based sealants. [11] Acetoxy silicones give of acetic acid while curing, they have a fast cure time (24 hours). These cure in the presence of atmospheric moisture, making it curable in room temperature. The acid released during the curing process can affect the electrical components negatively. Neutral cure silicones are non-corrosive and are ideal for electronics applications. [12]

Example of silicon-based sealants are:

- NSF Certified Silicone Sealants: which are ideal for foodservice equipment.
- High-Temperature Silicone Sealants: which are great for high temperature applications such as stoves, furnaces or ovens.
- Mildew Resistant Silicone Sealants: They inhibit the mildew(fungus) growth. Used in damp applications. [11]

Acrylic sealants are mostly used in sealing doors and windows. They are odorless, easily paintable and highly resistant to degradation caused by the environment. Acrylic sealants can be cured in many ways, but it is fastest when thermally cured. [10]

Phenolic sealants provide effective bonding and have a high rating against temperatures. This sealant is the only sealant that is available in powder, liquid and film form. [10]

2.2 Stepper motor driver

To run a stepper motor, it is necessary to have a microcontroller, driver circuit and a power supply unit. The driver bridges the gap between the digital signals of the microcontroller and actual movement of the stepper motor. It regulates the voltage and current consumption of the motor to run it as specified in the code uploaded to the microcontroller. [13] Since most stepper motors use two internal coils, see figure 2, at least two H-bridges are used in the driver to control and regulate the current and its direction fed from the power supply. Alternating excitation of the two windings will result in a more precise movement, which is sought after. The stepper motor driver can in many cases perform micro steps, which is the motor taking more steps than specified, to make the movement smoother, but can therefore also need more power. [14]

On Arduino IDE you can download libraries with predefined functions to make the code easier and to be able to perform desired tasks depending on components used in the circuit. To drive a stepper motor and have it accelerate and decelerate, there is a library which has been used in previous projects prior to this, called AccelStepper. With just a few functions, a stepper motor can accelerate which it is not originally designed to do. This is desired because the RS needs to start moving slowly, it cannot abruptly start moving at a rotating frequency of 1 Hz, 60 RPM. [15]

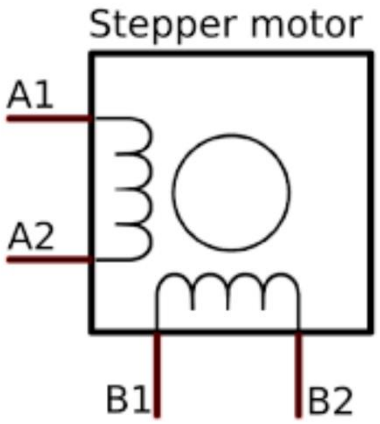


Figure 2. Stepper motor winding. [16]

2.3 Raspberry Pi

Raspberry Pi is a microcomputer operating on a single board (Single Board Computer, SBC) and has all the characteristics of any other fully functioning computer consisting of a processor, storage, memory and video chipset. It was first invented to affordably teach basic computer science in schools and developing countries. It became popular quickly, and many Raspberry Pi’s were sold, not only to schools but to hobbyists making projects at home, to be used as a desktop PC, or in old retro games. The Raspberry Pi runs on the Linux interface, is easy to use and consists of many ports, such as HDMI, USB and Ethernet. This makes it easy to connect monitors, accessories or to communicate in different ways with external microcontrollers using output pins. [17]

2.4 I2C communication

Inter-Intergrated Circuit or more commonly named I2C is a type of communication protocol that allows the connection of up to 128 slaves to a single master, it also allows multiple devices to function as masters. It uses two wires for transmitting data; SDA (serial data) which is used to send data between master and slaves, and SCL (serial clock) which acts as the line that carries the clock signal, so the devices are timed correctly. I2C is a serial communication protocol, meaning each line is transferred one bit at a time over the SDA line. The data is transferred in messages that have the structure as described in figure 3.

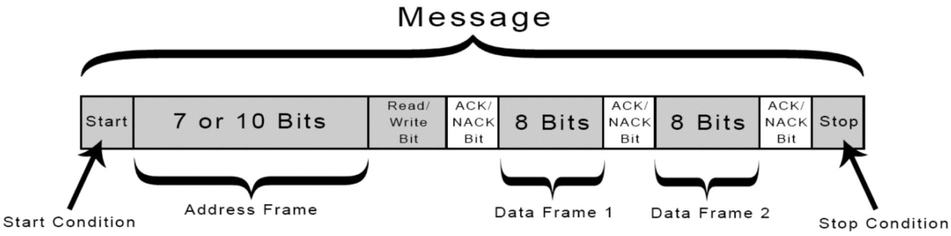


Figure 3. Structure of an I2C message. [18]

Start Condition: SDA switches from HIGH to LOW before SCL switches from HIGH to LOW.

Address frame: Each slave has a unique address. The master sends the address of the slave it wants to communicate with to every slave. All slave compares this address with their own and if it matches, the slave sends a LOW ACK back.

Read/Write Bit: One bit which is HIGH if the master is requesting data and LOW if it is sending data.

ACK/NACK Bit: ACK is sent if the address frame or data frame is sent successfully, NACK if not.

Data frame: one data frame is 8 bits long and is followed by ACK/NACK to verify that the data has been sent properly.

Stop Condition: after the last data frame has been sent the SDA goes from LOW to HIGH after SCL goes from LOW to HIGH to signal that all the data has been sent.

All SDA lines should be connected and all SCL lines should be connected. Each line should also be connected with a 4.7k Ohm resistor to VCC. [18]

2.5 Liquid Column Test

A liquid column can be added to a system to calculate the pressure within the system and to determine if there is a potential pressure leakage. Figure 4 below shows two different states. State a) is when no external pressure is applied, and b) is when external pressure is applied.

When no external pressure is applied, the pressure on both sides of the liquid will be the same, atmospheric pressure, therefore the liquid level will be equal.

However, if external pressure is applied, for example by blowing in air into the system through the tube, the pressure inside the system will become higher than the pressure outside of it. The higher pressure will push the liquid towards the other end of the column, resulting in a difference in liquid level of 2h. The height difference can then be implemented in the formula for hydrostatic pressure:

$$p = \rho * g * 2h \qquad \dots (1)$$

Where p is the external pressure applied, ρ is the density of liquid, g is the gravitational constant and 2h is the height difference between the highest and lowest liquid level.

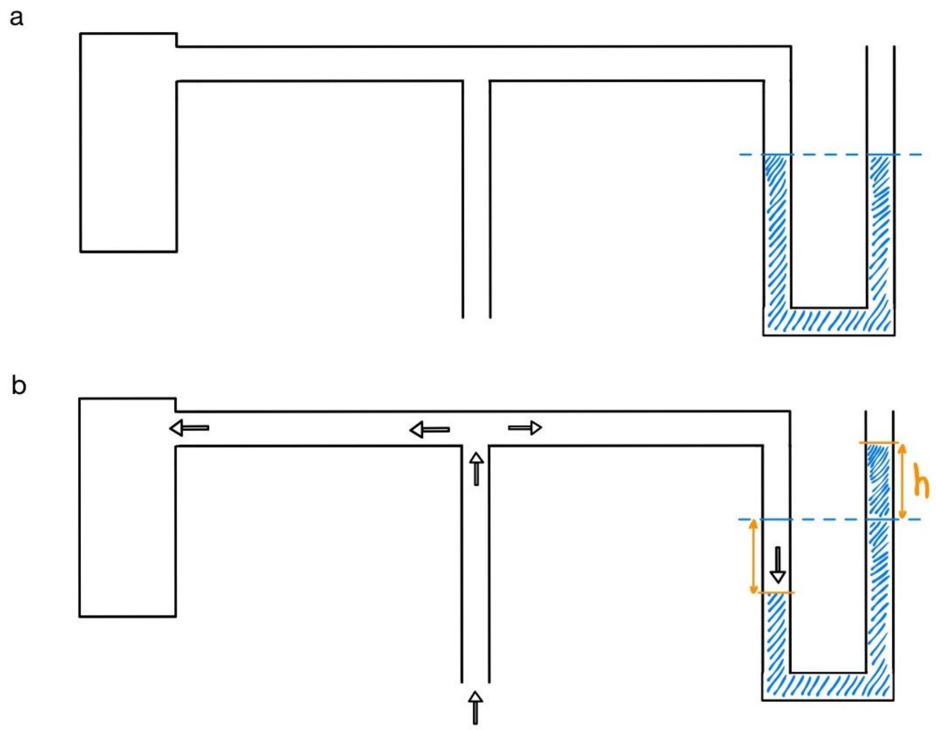


Figure 4. The working principle of a liquid column. a) shows the state when the system is not being pressurized. b) shows the state when the system when external pressure is applied.

To identify the potential pressure leakage, one can simply apply external pressure by blowing in air into the tube and quickly seal the tube when a noticeable liquid level difference is achieved. If there is no leakage, the liquid level difference will remain the same. However, if there is a leakage, the difference will decrease over time until the levels of the liquid are at the same height on both sides, as can be seen in state a) in figure 4. The reason for this is that the added pressure inside the system will continuously leak out to the surrounding until the system reaches the atmospheric pressure. The faster the liquid levels are even out the larger the leakage is.

2.6 Generated Work

2.6.1 Ideal Gas Law

The ideal gas law describes the relationship between volume V, , pressure p, , temperature T and number of moles n together with the gas constant R. The relationship

$$pV = nRT \qquad \dots (2)$$

when dealing with constant volume and the same number of moles, it can be rewritten as:

$$\frac{p}{T} = \frac{nR}{V} = C \qquad \dots (3)$$

where C is a non-zero constant. Whereas it can be written as:

$$\frac{p_1}{T_1} = \frac{p_2}{T_2} \rightarrow p_2 = p_1 \cdot \frac{T_1}{T_2} \qquad \dots (4)$$

to describe the relationship between pressure and temperature when a gas goes from one state to another under constant volume. This relationship is also called *Charles´s law*. [19]

2.6.2 Volume-pressure work

Work is defined as force F multiplied by the path element ds and then integrated over their inner product:

$$w = \int F \cdot ds \qquad \dots (5)$$

In the case of a cylinder with a piston, with a pressurized gas inside within with a pressure of p_{int} and an external pressure of p_{ext} . Both exert a force on each other. This force is defined as $p = \frac{F}{A}$. The piston will move as the gas inside the piston expands or contracts to reach an equilibrium to make the internal and external pressure the same. This result in a motion that is work:

$$w = \int \left(\frac{F}{A}\right) \cdot (A \cdot ds) = \int p \cdot dV \qquad \dots (6)$$

If the external pressure is constant the integral becomes [20]:

$$w = p_{ext} \cdot \int_{V_1}^{V_2} dV = p_{ext} \cdot \Delta V \qquad \dots (7)$$

2.6.3 Regenerative work

A motor uses electrical energy and converts it into mechanical energy in the form of motion. Generators on the other hand converts mechanical energy to electrical energy.

An electrical motor consists of coils of wire which is wrapped around a metal core. A magnetic field is created in the coils when an electric current pass through them. This creates a magnetic field the interacts with another magnetic field created by magnets or electromagnets.

A generator consists of a rotor and a stator. The rotor is driven by a mechanical motion, the stator contains windings and when the rotor rotates around the stator a magnetic field is created. The magnetic field induces a current in the winding which can be used as an electrical output. Electric motors and generators have many similarities, such as magnetic field, windings and electrical wirings. Since they are so similar, an electric motor can be turned into a generator. [21]

A linear generator is a generator that transform a linear motion into electric current. The power output from a linear generator will increase with the number of windings in the coils, strength of permanent magnet, stroke and frequency. [22]

2.6.4 Maximum power theorem

The maximum power theorem states that the maximum power that can be transferred from a source to a load is 50%. This occurs when the source impedance equals the load impedance. At least 50% of the power from source will be lost as heat in the source impedance and at most 50% of the will be delivered to the load.

The maximum power will be achieved at these conditions, but maximum efficiency will be increased when choosing a larger load resistance. The amount of power delivered from the source depends on the load and when the load resistance increases, the delivered power from the source will decrease while the percentage of power delivered to the load increases and therefore the efficiency increases. [23]

2.6.5 Power measurement for AC

AC-circuits contain reactance. Components that create magnetic or electric fields stores power and provides the circuit with reactance. In AC-circuits the voltage and current are sinusoidal and the phase between them are dependent of the reactance in the circuit. The instantaneous power in an AC-circuit is:

$$P = U * I = U_m \sin(\omega t + \theta_U) * I_m \sin(\omega t + \theta_i) \qquad \ldots (8)$$

By applying trigonometric product-to-sum identity and by using $\theta = \theta_U - \theta_I$, the phase angle between voltage and current waveforms into the equation above, the equation will become:

$$P = \frac{I_m U_m}{2} * (\cos(\theta) - \cos(2\omega t + \theta)) \qquad \ldots (9)$$

The mean value from a sinusoidal waveform is called root-mean-squared and can be used to calculate the average values of voltage and current.

$$\frac{I_m\, U_m}{2} = \frac{I_m}{\sqrt{2}} * \frac{U_{RMS}}{\sqrt{2}} = U_{RMS} * I_{RMS} \qquad \ldots (10)$$

By inserting this into the equation (10) in (9) the following equation is achieved:

$$P = U_{RMS} * I_{RMS} * (\cos(\theta) - \cos(2\omega t + \theta) \qquad \dots (11)$$

The first term is constant while the other is a varying sinusoidal wave. When averaging the power over a fixed number of cycles, the equation simply become

$$P = U_{RMS} * I_{RMS} * \cos\theta \qquad \dots (12)$$

When the circuit is purely resistive which means that there is no reactance, $\cos(\theta) = 1$, and the dissipated power can be written as $P = V_{RMS} * I_{RMS}$ it can be rewritten to

$$P = \frac{{U_{RMS}}^2}{R} \qquad \dots (13)$$

In order to calculate the power dissipation over a resistor. [24]

2.6.6 Previous test

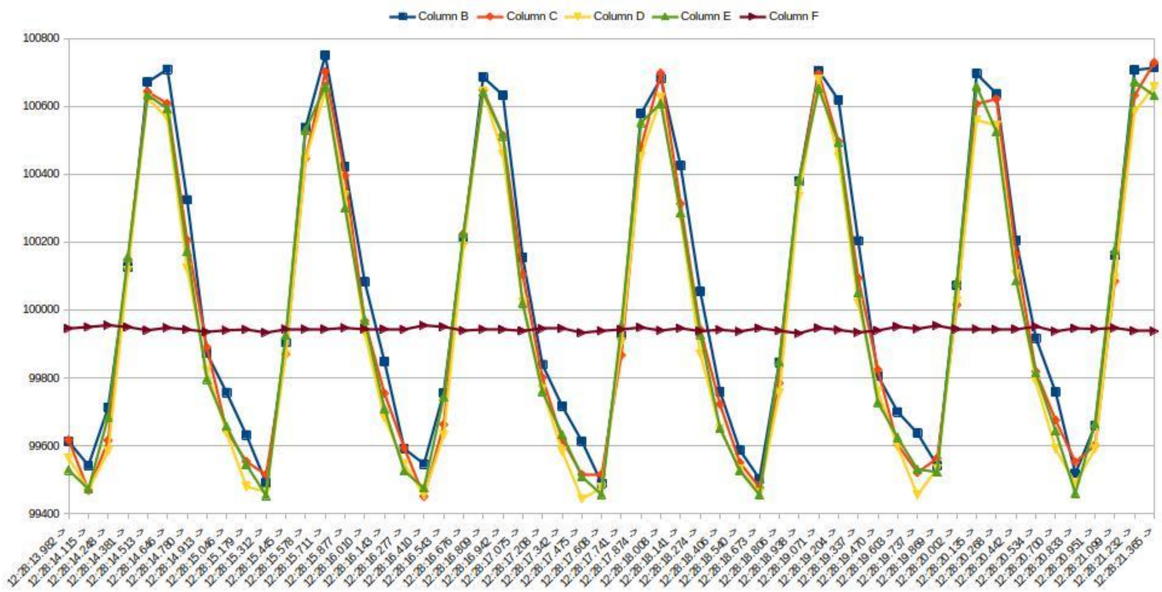


Figure 5. Previous pressure test performed with 4 pressure sensors by Nils Karlberg. [25]

Tests performed by Nils Karlberg during summer of 2023 suggest that the pressure in the GREC is identical in all parts. The test done in figure 5 was performed at 1 Hz, 75°C at the hot fin and 20°C at the cold fin, resulting in a temperature difference of 55°C. The pressure test of the Lab Model v3 was conducted on July 17, 2023, by nilsinside AB in Motala, Sweden and measured the pressure in four places, 4 times per rotation. Each line in figure 5 is the pressure in some part of the GREC. The tests achieved a pressure difference of 1 249 Pa. [25]

2.7 Repeatability and reproducibility

To verify measurements, repeatability is used. If the same results can be produced within a short amount of time, with the same equipment, it can be assumed that the results are not just by chance. Repeatability verifies results from the exact same setup. The following criteria must be the same when calculating repeatability.

- Conditions
- Operator
- Equipment
- Location
- Item or unit under test
- Method

When these criteria are met, test points must be decided. Tests points are what value the parameters that are tested should be. An example is when testing how pressure changes with temperature. The test points would then be the temperatures the tests should be conducted at. If you expect a linear relationship, two test points is sufficient. At least three test points should be used if there is a non-linear relationship.

Reproducibility is a measurement of how equivalent results are when the same study is made by different individuals in different locations and with different instruments. An experiment is reproducible if the result can be replicated by anyone if the correct procedure is followed. Having an experiment that is reproducible minimizes errors and ambiguity when a project moved from the development to the production phase.

Both repeatability and reproducibility are essential for all areas of science. Reproducibility is used for accurate research and to eliminate errors within your own experiments. Repeatability is used to measure the accuracy of the measurements and to confirm the results. [26]

3. Method

During this chapter, different methods needed to answer our research questions will be presented.

3.1 Locating leakage

There were several locations in the GREC that could provide to the leakage in the construction. The most critical points that would be investigated further were decided to be screws, motor connection, sensor connection and the edge between the two halves of the GREC.

Methods to locate the leakage were considered. It was clear that the GREC had to be pressurized to discover the leakage. According to nilsinside AB [27] the relative pressure in the GREC is at most 3 kPa during tests. Therefore, it was decided that the leakage would be acceptable when the GREC could hold a relative pressure of 3 kPa for a satisfactory amount of time, around 2 minutes. Since the pressure in GREC is low, it was enough to blow through a plastic tube to achieve.

To measure the pressure, SparkFun MicroPressure Sensors was used. It was also decided to use a water column to measure the relative pressure in the GREC. This made it easy to see the pressure loss in the GREC as the water column loses its height when the pressure is lost from the GREC.

Methods that were considered for locating the leakage:

- Listening for leaks.
- Using an infrared camera to detect hot air leakage.
- Fill the GREC with smoke to visually see where leakage occurred.
- Patch one part at a time and check leakage after each patching.

Listening for leaks was not deemed a viable strategy since the pressure in the GREC was too low to be heard in any way. Filling the GREC with smoke might have been a way to easily locate the leakage but without any safe way to fill the GREC with smoke this was also ruled out.

The other methods were decided to go forward with. Using an infrared camera was a simple way to locate leaks. Since GREC is a heat engine it already had the necessary heating elements to heat up the air in the GREC. The camera was from the manufacturer FLIR and was provided by Linköping University.

To systematically cover up one potential leak at a time seemed also like a simple and effective way to locate leaks. It was decided to use plastic film and tape to cover each critical point since it was easily accessible, cheap and functional.

3.2 Choosing sealing method and sealant

When choosing what kind of sealant was to be used, a couple of criteria had to be fulfilled:

- The sealant is able to withstand temperature of up to 200°C
- The sealant is easily applicable.
- The sealant is cheap.
- The sealants lifetime exceeding the lifetime of the GREC LabModel v3.

Through tests it had been determined that the hot aluminium plate could reach up to 150 °C while the hot plate could reach temperatures of over 180 °C. Since the GREC could require sealing alongside the edge of the aluminium plate at least 150 °C heat resistant sealant was needed. Depending on where the hot plates were placed on the GREC it might be closes to the sealant. Therefore, it was decided that the sealant had to be resistant to 200 °C. No member of the group had previous experience with sealing and that is why the sealant had to be easily applicable. A requirement was also that there would be no need for sealing any leaks ever again. One goal for this project is to facilitate coming project on the GREC and that is why it is important that the solutions made now will last in the future.

3.3 Choosing new stepper motor driver

In the project's beginning, the setup consisted of a SM2564C60B41 NEMA 23 stepper motor driven by a DRV8825 stepper motor driver. It was discovered early that the driver was too weak for the stepper motor. It would often overheat and could not be operational for longer periods and it was decided that a new driver was needed. SM2564C60B41 NEMA 23 stepper motor is rated for 6 A while the old driver could provide current up to 2.5 A which was enough to run the GREC, however the main problem with the old driver was the driver overheating.

When deciding which new driver circuit to purchase a couple of criteria had to be fulfilled:

- It should be able to support SM2564C60B41 NEMA 23 1/32 microstepping.
- It should have great cooling properties and be able to function over a longer period.
- It should have an output current close to 6 A.
- It should be compatible with Arduino library AccelStepper.
- It should be able to run on 12-24 VCC.
- It should be cheap.

3.4 Selecting new control method

Previous projects did not have sufficient time to develop an effective control method to manage the motor satisfactorily and collect sampled data easily. To start the motor and thereafter collect data from the pressure sensors, a button located on the breadboard had to be pressed. This method works but lacked the ability to make adjustments such as changing the frequency. This improvement was to be implemented in this project.

The founder of nilsinside AB and inventor of the GREC-idea, Nils Karlberg, took over LabModel v3 after it was done at Linköping University during the spring of 2023, and continued to further develop it. The main thing he wanted to improve was the way of controlling the GREC model. To do this he started by implementing a Raspberry Pi where he could store all code, data sampling and start and stop the motor using a html-site. To communicate with the Arduinos and sensors the I2C (Inter Integrated Circuit) communication method was used.

3.5 Instructions manual

It was not easy to get GREC LabModel v3 to operate as intended at the beginning. It was therefore decided to make an instruction manual that will facilitate coming research project on the GREC LabModel v3. The following was focused on:

Sorting of all the electrical components and to make the wiring easier to follow. An instructions manual was also made that goes into detail on how to run the GREC LabModel v3. After reading the manual it should be clear for anyone inexperienced with GREC LabModel v3 on how to get started with testing.

The instruction manual includes:

- Information on what code should be uploaded to what Arduino.
- Pictures and description of the wiring.
- Step by step process of what to do in what order to get GREC to function correctly.

3.6 Measuring output work

There were several different methods to consider when measuring the output work from the GREC. Since a water column had already been constructed, it´s capabilities to measure the output work was investigated. It was concluded that for small frequencies around 1 Hz (60 rpm) it was capable of being a measuring device. For frequencies above 1 Hz, it did not perform well since friction between the tube and the water and gravitational properties did not allow the water in the tube to move fast enough. The inertia of the water made the water column flatten out and no volume change occurred although a clear pressure difference was achieved. It was analyzed that there would be more successful with another liquid with lower viscosity.

One option that was dismissed was a piston-cylinder solution. A standard off-the-shelf piston was required because of the lack of time to build. This would prove too expensive and due to lack of time it was not possible.

The method that would be focused on would be the generation of electricity and measurement of the electricity to determine the output work. As describes earlier, it is possible to produce electricity by reversing an electric motor. For this, a speaker can be used. The speaker membrane will oscillate with the pressure pulses from the GREC. A speaker consists of coils and permanent magnets, just like electric motors. When the membrane oscillates, a current will run through the coils which can be measured by a multimeter. The output from speaker is sound which in turn is just an oscillation of pressure. It was anticipated that a speaker would be able to convert the pressure pulses from the GREC into electricity well.

It was decided that the water column would be used at low frequencies and the speaker at higher frequencies.

When calculating the work produced by the water column the equation

$$P = p \cdot \Delta V \qquad \dots (14)$$

was used where p is the external pressure and ΔV is the volume difference in the tube.

To measure the electricity converted from the vibrating membrane of the speaker, a multimeter was used. To match the impedance of the source (speaker) and the load (short circuit) the circuit used a resistor put in series that matches the load resistance. When the circuit was set up and understood, the voltage drop across the resistor was read by a multimeter. This produced a sinus-looking graph where the maximum- and minimum values was read from the curve and amplitude calculated by following equation:

$$U_{amplitude} = \frac{U_{max} - U_{min}}{2} \qquad \dots (15)$$

where U is voltage. Since it looked like a sinus wave, the root mean square, RMS, was calculated by:

$$U_{RMS} = \frac{U_{amplitude}}{\sqrt{2}} \qquad \dots (16)$$

To further calculate the power, the following equation was used, a rephrasing of Ohms law:

$$P = \frac{{U_{RMS}}^2}{R} \qquad \dots (17)$$

Equation (17) only holds true under the assumption that the impedance is purely resistive. If the impedance consists of just resistance, the power output will be the largest it can be, otherwise it will be needed to be scaled down using a power factor. Equation 4 can still be used to visualize the power output under ideal conditions.

3.7 Different Tests

The GREC was tested when it started to run correctly. Tests was done to prove that the measurements of GREC are repeatable. Pressure tests has previously been done on the GREC LabModel v3 by nilsinside AB and these will be compared to investigate the repeatability of the results.

The tests that have been performed are listed below:

- Repeatability-test to ensure comparable results can be generated during similar conditions.
- Tests to make sure the pressure is the same in all parts inside the LabModel v3.
- Tests to determine the relationship between pressure difference, rotational frequency and temperature
- Speaker connected to the output of the GREC and the produced energy measured.
- Water column test to calculate generated work.

To be able to perform the different test, a heat difference needs to be achieved between the hot- and cold fins. Two hot plates have previously been installed at the hot side with settings to adjust the heat. The hot plate has settings from one to five as well as a “MAX” setting. During the different tests which was be conducted and will be presented further down, the hot plates will use the settings four, five and “MAX”. The table 2 below shows which temperatures these produces across different parts of the hot fin.

Table 2. Results of what the different heat settings of the hot plates generates on the hot- and cold fins of the LabModel v3.

Heat setting	Hot fin			Cold fin
	Top	Side	Middle	
4	80°C	61°C	52°C	23°C
5	100°C	79°C	61°C	24°C
MAX	145°C	101°C	82°C	26°C

These temperatures were measured with the use of an infra-red camera. Figure 6, figure 7 and figure 8 show the thermal pictures that were taken with the heat setting “MAX”.

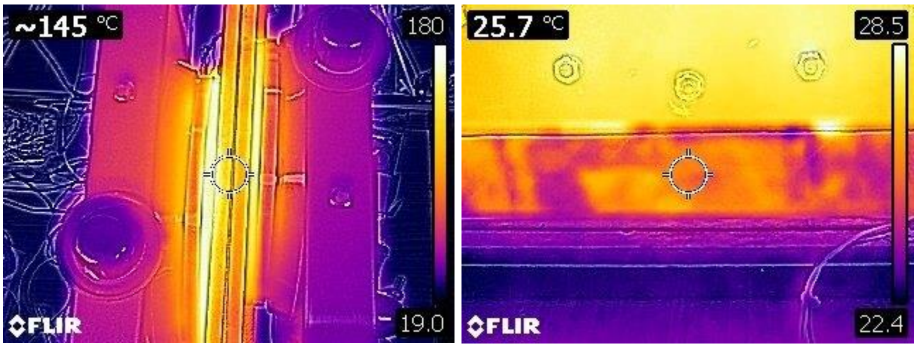


Figure 6. Left picture shows “Top” of the hot fin with measurement directly between the two hot plates and the right picture shows the cold fin. Pictures taken with heat setting “MAX”.

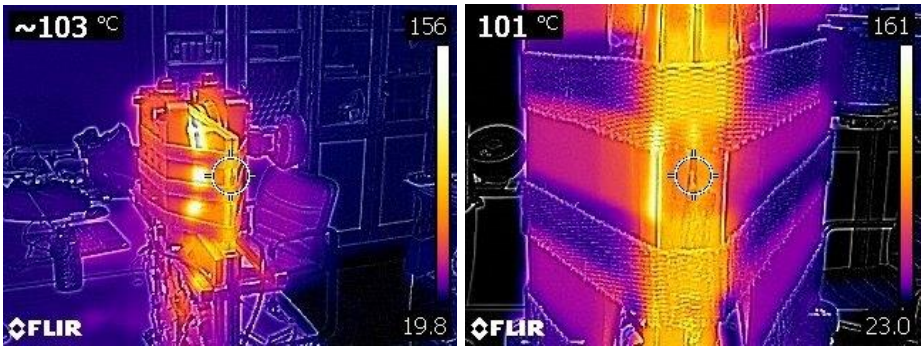


Figure 7. Left picture shows a zoomed-out picture of the "Side" of the hot fin and the right picture shows a zoomed-in picture of the "Side". Picture taken with heat setting "MAX".

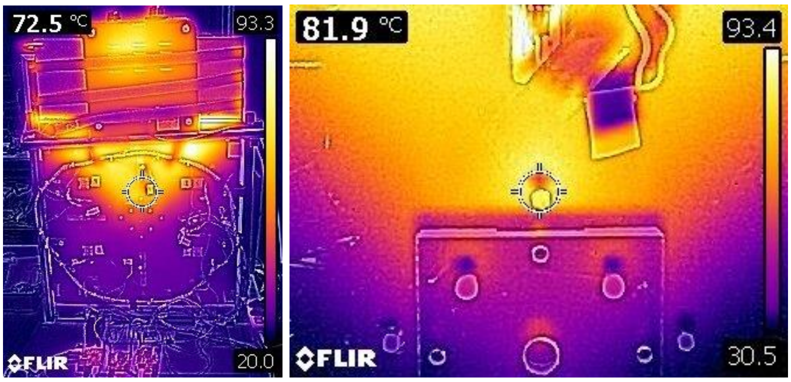


Figure 8. Left picture shows a zoomed-out picture of the "Middle" of the hot fin and the right picture shows a zoomed-in picture of the "Middle". Picture taken with heat setting "MAX".

3.7.1 Repeatability test

To make sure the LabModel v3 provides repeatable results, the same test was to be done at 3 different days and the results analyzed to see if they proved to be the same. Specifications for each test was:

- Rotational frequencies: 1, 2 & 3 Hz.
- Heat setting 4 on the hot plates.
- Ambient pressure must be recorded due to it being different from day to day.
- The same MicroPressure sensor will be used.
- Maximum number of samples per second will be taken.

3.7.2 Homogenous pressure test

The pressure in the GREC LabModel v3 has previously been hypothesized to be the same in all parts. To prove, or disprove this thesis, a test with multiple sensors in the LabModel v3 was conducted. This test was important because if the pressure was proven to be similar in the whole model, in future work, only one pressure sensor will be needed. Specifications for the test:

- Rotational frequencies: 1, 2 & 3 Hz.
- Heat setting "MAX" on the hot plates.

- Ambient pressure must be recorded to calibrate each sensor.
- All four MicroPressure sensors will be used.
- Maximum number of samples per second will be taken.

The sensors will be positioned as shown in figure 9 below:

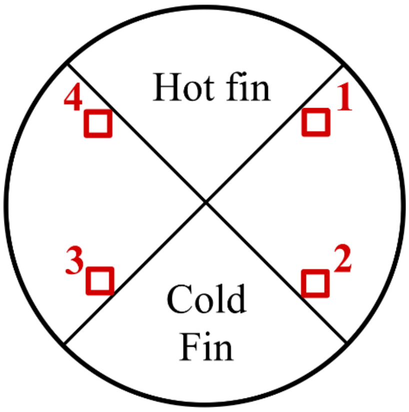


Figure 9. The planned locations of pressure sensors on the GREC LabModel v3 during the homogenous pressure test.

3.7.3 Pressure-temperature-frequency relationship

The pressure in the GREC LabModel v3 has previously been thought to be increase linearly with the temperature and the relationship between pressure and frequency have been unknown. Test was performed to verify and discover the true relationship between these quantities. Specifications for the test:

- Rotational frequencies: 1, 2 & 3 Hz.
- Heat setting: 4, 5 and "MAX"
- 1 sensor will be used.
- 4 samples per second will be taken.

3.7.4 Speaker as output

To measure the power output generated by the LabModel v3 a speaker acted as a linear generator as earlier mentioned. The speaker is a 4.5 cm full spectrum element with an 8 Ω impedance. It was attached to the center output of the LabModel v3 with a fitted tubing and a 3D-printed attachment to make sure no air leakage, see figure 10. Furthermore, the speaker was to be short-circuited with a 10 Ω resistor to satisfy the maximum power theorem and be able to provide the maximum amount of power. Thereafter a multimeter was connected in parallel with the resistor to measure the voltage drop, see wiring diagram in figure 15. The multimeter used was a "Keysight 34465a" which has a max reading rate of up to 50,000 readings/s and can measure in the mV- and µA-range as well as log- and export the data. Specifications for the test:

- Rotational frequencies: 1, 2 & 3 Hz.
- Heat setting "MAX" on the hot plates.

- 100 samples per second will be taken.

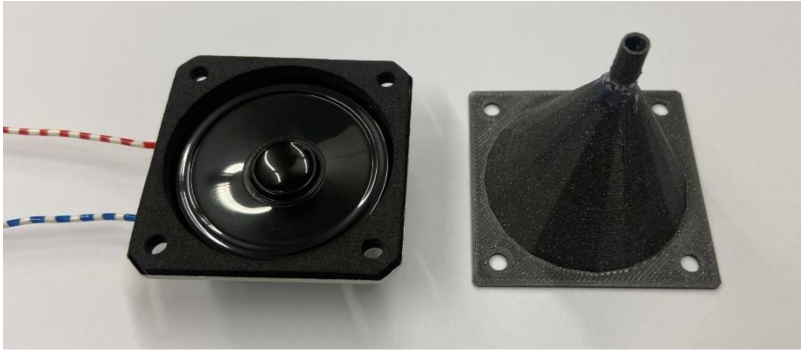


Figure 10. The speaker that will be used in test and the 3D-printed attachment which join by 4 x screws and bolts as well as a plastic tube in the end of the 3D-printed part that connect to the GREC LabModel v3. Front of speaker (left), 3D-printed attachment (right)

3.7.5 Water column

Another method of calculating the power output of the LabModel v3 is through a water column test. As mentioned before, this test performed best at only low rotation frequencies. By doing this test, the boundary work was first calculated followed by the power. Specifications for the test:

- Rotational frequencies: 1 Hz.
- Heat setting “MAX” on the hot plates.
- Water level monitored and noted for 30 seconds.

4. Results

4.1 Choice of Sealant

The chosen sealant was a high temperature acetoxy silicone from Biltema. It is resistant to temperatures over 250 °C and has a curing time of 24 hours and was applied using a caulk gun. It costs 100 SEK and is resistant to aging. [28]

The original setup included an O-ring by the stepper motor connection. It was assumed that the O-ring was functioning as intended since it is a reliable sealing method, and no visible damages could be seen. Since the main leakage appear to come from the stepper motor connection, it might have been smart to investigate the O-ring further. With lack of time, it was decided to keep the original O-ring and to try to cover up any leakage using silicone.

4.2 Identified Leakage

When using the methods to locate the leakage described in the previous chapter it was clear that the main leakage appeared to be where the stepper motor was connected to the GREC. Leakage was also located on the sides of the GREC as well as through the screws around the perimeter.

The infra-red camera showed that heat left the GREC though the screws. Otherwise, there were no clear leakage that occurred through the infra-red camera. When covering each potential leakage point with the high temperature silicone, leakage decreased when covering the central axis, where the stepper motor was attached to the GREC. When the screws keeping the GREC together was loosened, the leakage clearly increased. The leakage then occurred through the sides of the GREC and through the screws. Because of this, the sides as well as the screws and nuts were sealed with the silicone. Pressure tests with the water column were performed consequently to make sure the leakage was minimizing. The total leakage was discovered to be higher when the GREC was at room temperature and less when it was heated up.

4.3 Choice of stepper motor driver

The chosen driver is the TB6600 from DFRobot, which can be seen in figure 11 below [29]. It was chosen because of its great compatibility with Arduino and Nema 23 stepper motor. Nema 23 and TB6600 is a common combination, and it is also compatible with the AccelStepper library. It is an external driver which means that it does not need to be attached directly to the breadboard and thus clearing up space. TB6600 hade the following specifications:

- Supports up to 1/32 micro stepping (in ½^*n* increments, *n*=0,1,2...)
- Input voltage 9~42 VDC
- Max output current of 4A
- 8 kinds of current control (0.5A, 1A, 1.5, 2A, 2.5A, 3A, 3.5A)

- Cost 200 SEK

It also included a large heat sink as well as automatic semi-flow to reduce heat. The driver featured several switches where current and micro steps could be adjusted. This made it easy to control and it could be easily determined what setting the driver used. However, this feature meant that changing the current and micro steps had to be made manually and could not be achieved by using the Arduino. For this project, the goal was to have the motor running smoothly. This was achieved by testing different micro step-settings and implementing different modifications in the code. It was concluded that 1/8 micro stepping was optimal for this setting. This changed through the manual switches on the driver unit and in the code. To change the microstep-setting in the code, 200 must be divided with the current microstep-setting. The Nema 23 requires 200 steps (i.e. 1.8° per step) in full step mode to complete one rotation and in micro stepping 1/8 setting it requires 1600 steps to complete one complete rotation (0.225° per step). [30]

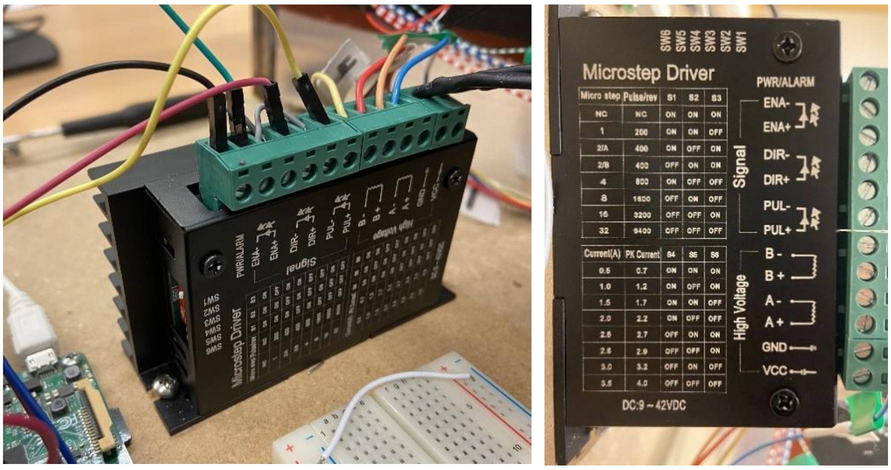


Figure 9. The chosen stepper motor driver TB6600 with the micro step and current adjustments to the right.

4.4 Code and Wiring Diagram

To operate and run the LabModel v3 with its sensors, multiple different codes and wiring schedules was used.

The purpose of each code and when they are used are presented in the table 3 below. Each code can be found in Appendix A.

Table3. Name of all different codes that was used with their purpose. The table also shows during which test the codes were used.

Code	Purpose	Tests
Link02.py	Send data from interface to slave to change motor settings.	All tests
TB6600.ino	Control motor settings depending on received data from master.	All tests
sparkfun_EOC.ino	Take pressure samples from one sensor every 10^{th} millisecond.	Repeatability
sparkfun_EOC_multi.ino	Take a pressure sample from multiple sensors at the same time as often as possible.	Homogenous pressure

The wiring on the LabModel v3 was seen over and organized for easier understanding regarding the different connections. This can be seen in figure 12 below.

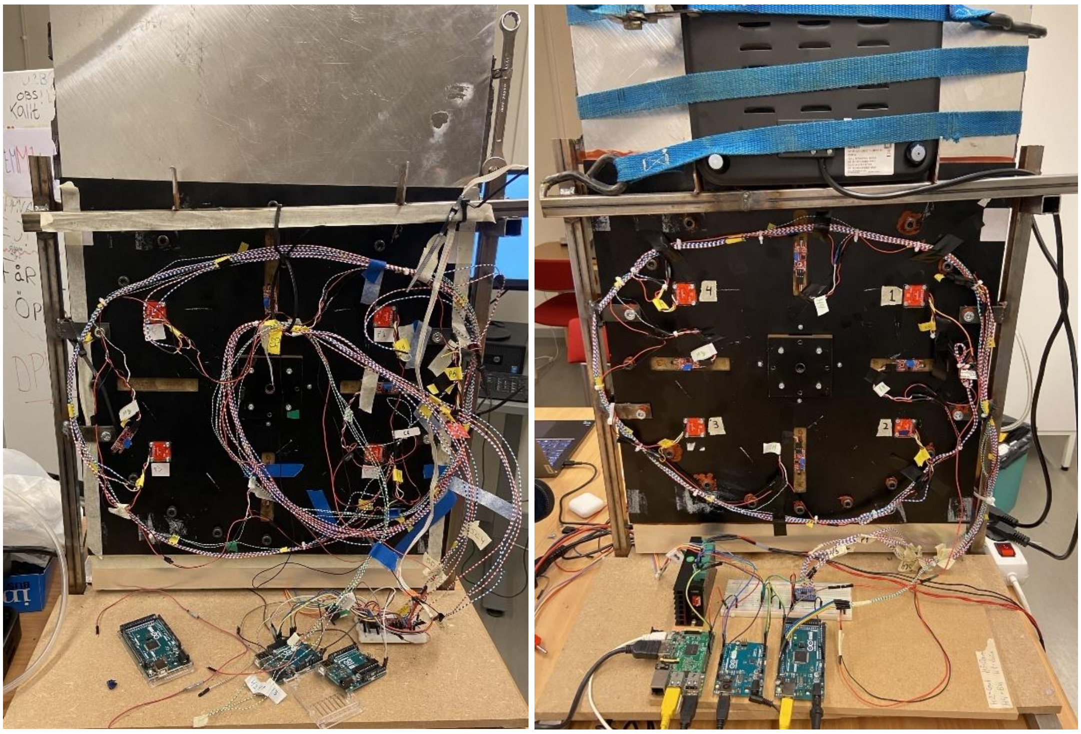


Figure 12. GREC LabModel v3 before (left) and after (right) cleanup and organization.

The wiring schedule for the homogeneous test is shown in figure 13 below:

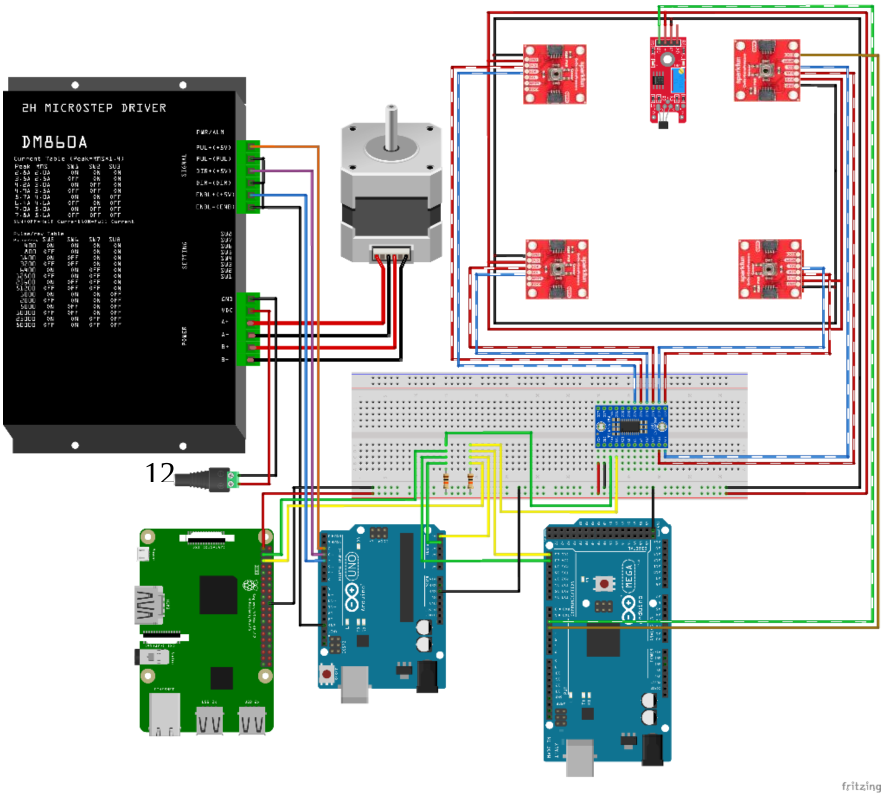


Figure 13. The wiring diagram of the LabModel v3 used for the homogeneous test.

The wiring schedule for the repeatability test is shown in figure 14 below:

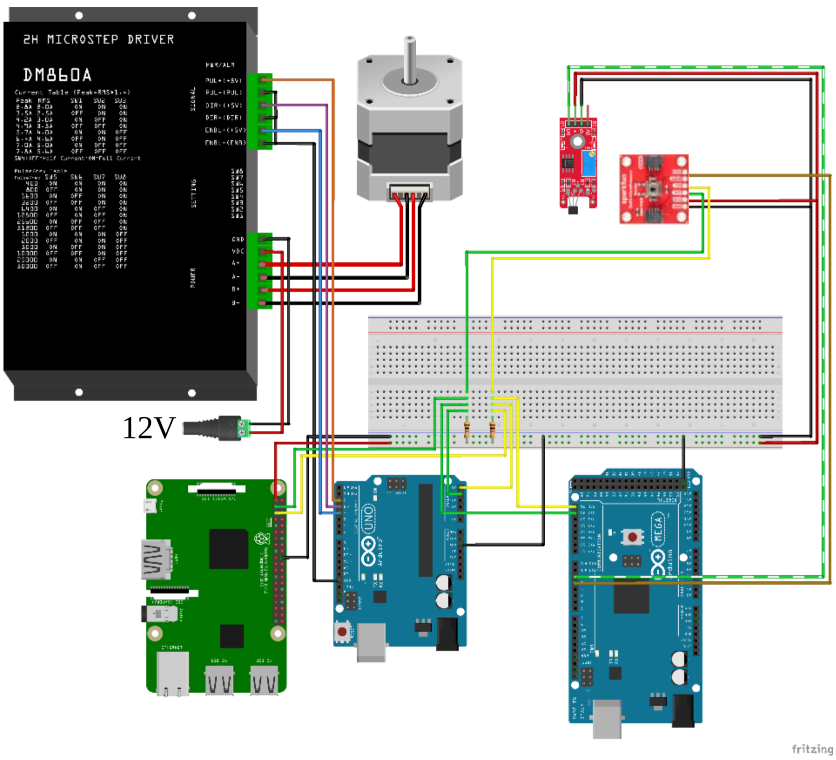


Figure 14. The wiring diagram of the LabModel v3 used for the repeatability test.

To measure the voltage generated by the speaker the wiring diagram seen in figure 15 was used:

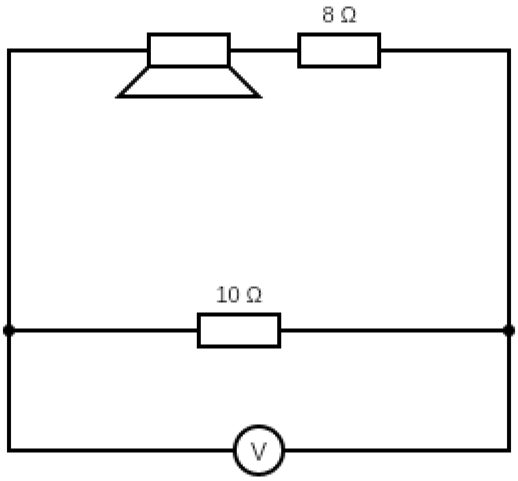


Figure 15. Wiring diagram used to measure the voltage from a speaker with a multimeter.

To get a more in dept instruction on how to operate the LabModel v3 an instruction manual was created (see Appendix B).

4.5 Test results

During this section, the result for every test will be presented. This includes the test results for repeatability, homogeneous pressure, speaker power output, water column power output and the relationship between pressure-temperature-frequency.

4.5.1 Repeatability

Results for repeatability are shown in figure 16, figure 17 and figure 18 below. It was desirable to use 3 measurements but due to failing pressure sensors, only 2 measurements were uses.

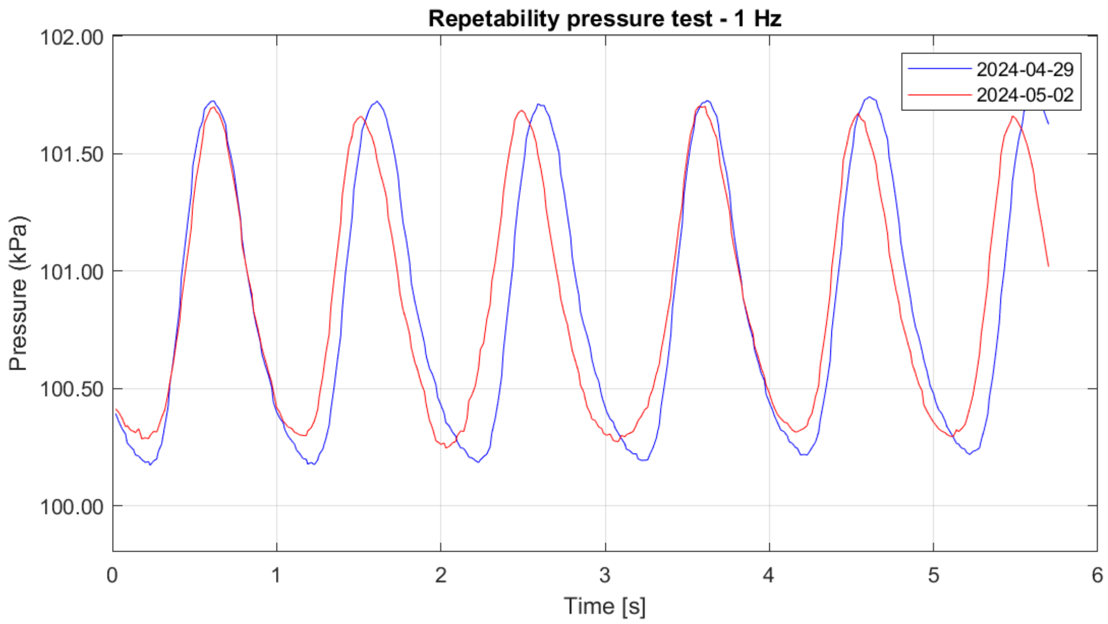


Figure 16. Result for the repeatability test at 1 Hz. Time in seconds on the x-axis and pressure in kPa on the y-axis.

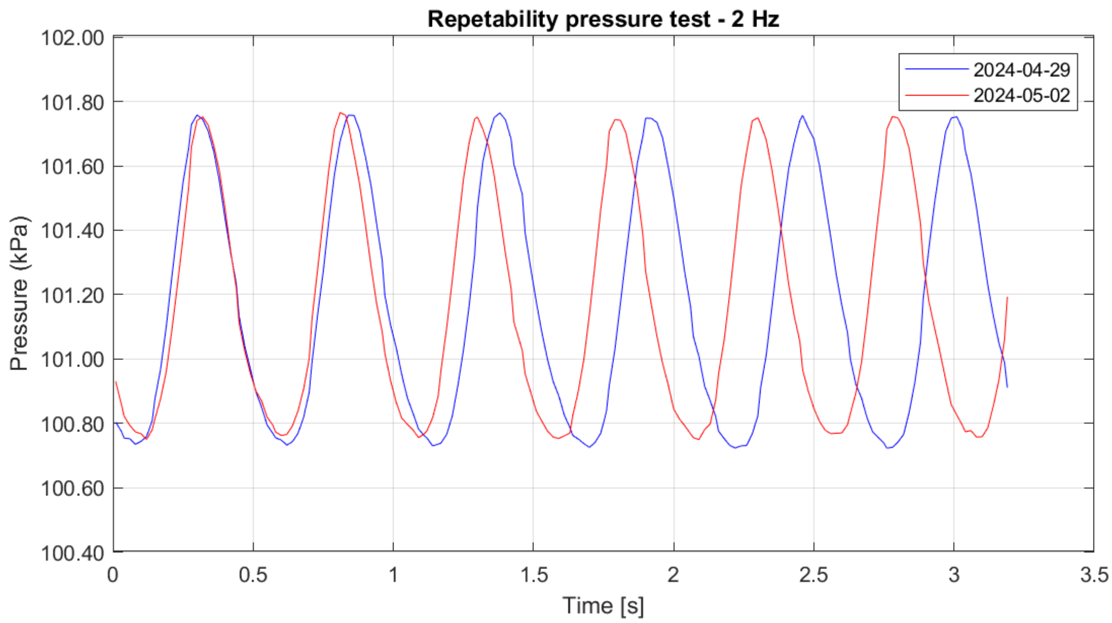


Figure 17. Repeatability test at 2 Hz. Time in seconds on the x-axis and pressure in kPa on the y-axis.

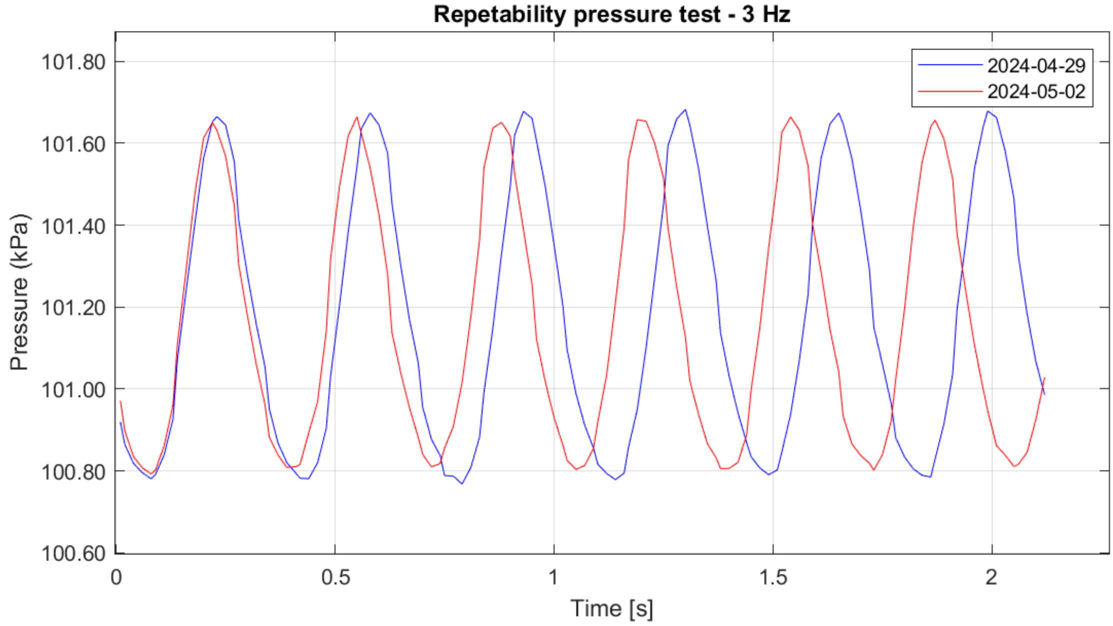


Figure 18. Repeatability test at 3 Hz. Time in seconds on the x-axis and pressure in kPa on the y-axis.

4.5.2 Homogeneous pressure test

During the testing, only three functional Sparkfun MicroPressure sensors were available. They were positioned according to figure 19 below.

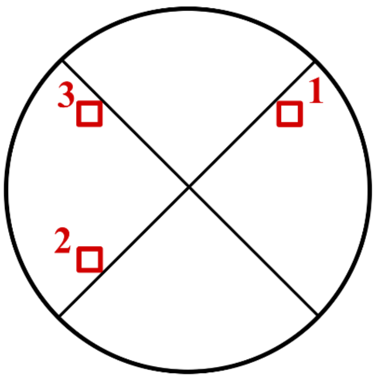


Figure 19. The location of three pressure sensors that was functional and used during the homogenous pressure test.

The result of the measured ambient pressure for each different sensor is presented in table 4 below:

Table 4. Shows the ambient pressure for each sensor.

	P1	P2	P3
Ambient pressure [Pa]	100198.4	99963.4	99965.4

The raw pressure readings, before taking ambient pressure into consideration are presented in figure 20, figure 21 and figure 22 below.

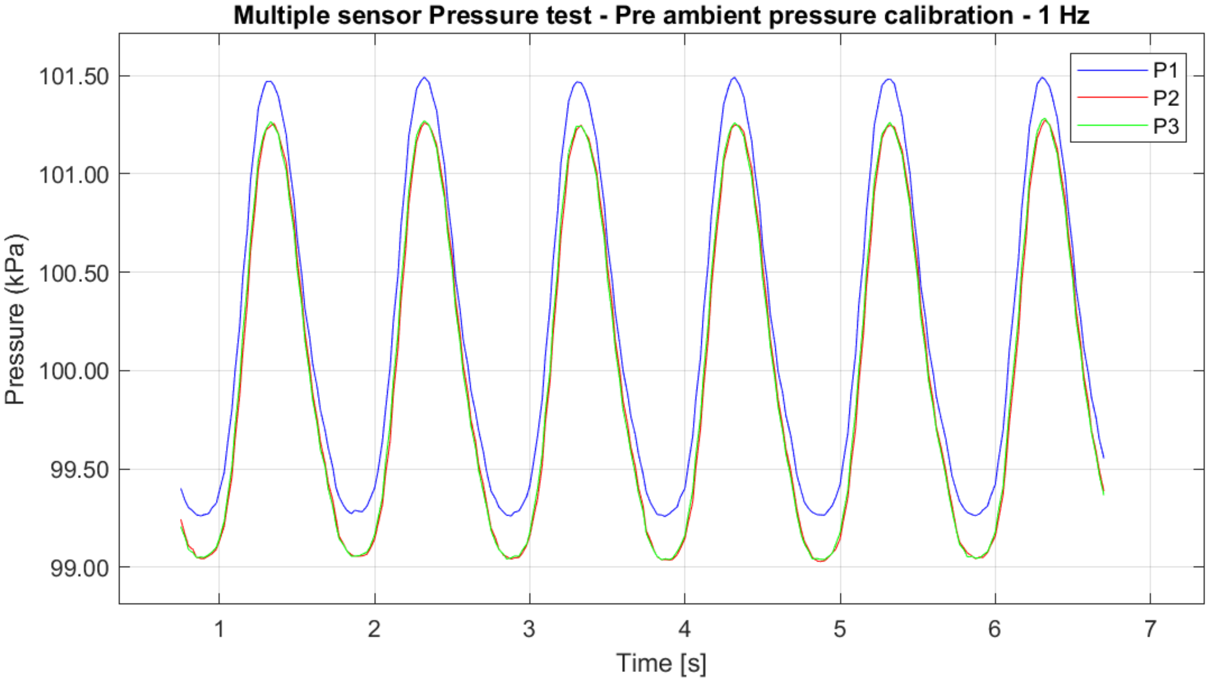


Figure 20. Homogeneous pressure test at 1 Hz, without calibrating the sensors. Time in seconds on the x-axis and pressure in kPa on the y-axis.

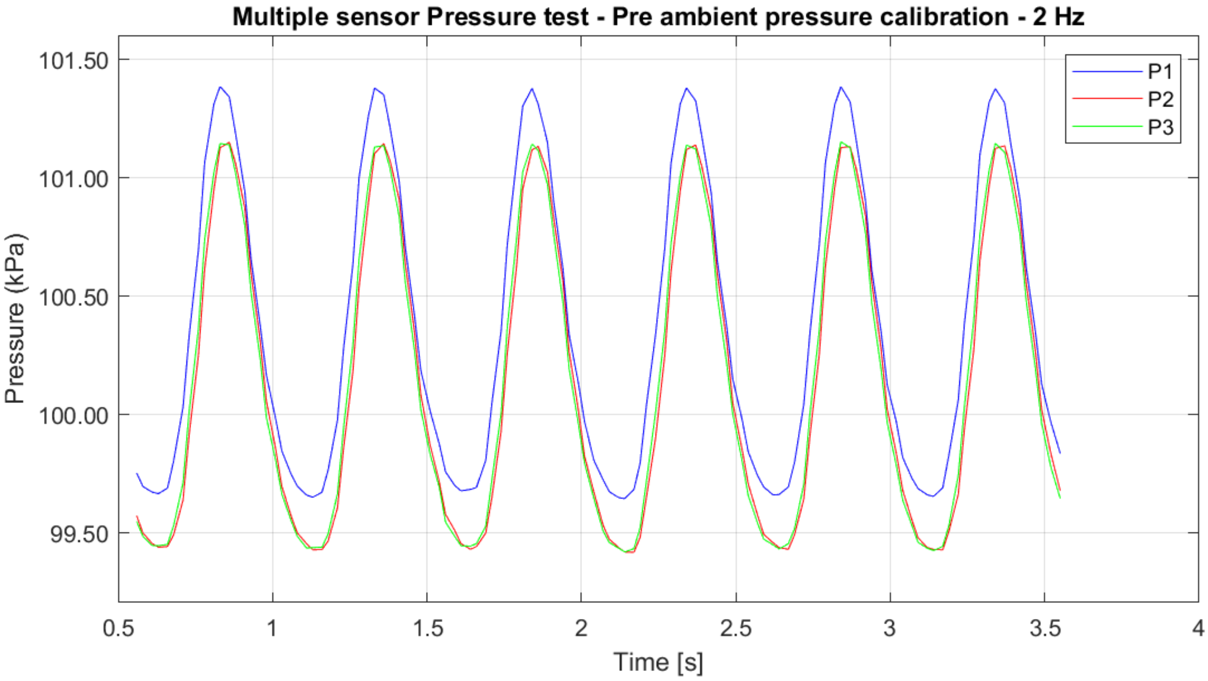


Figure 21. Homogeneous pressure test at 2 Hz, without calibrating the sensors. Time in seconds on the x-axis and pressure in kPa on the y-axis.

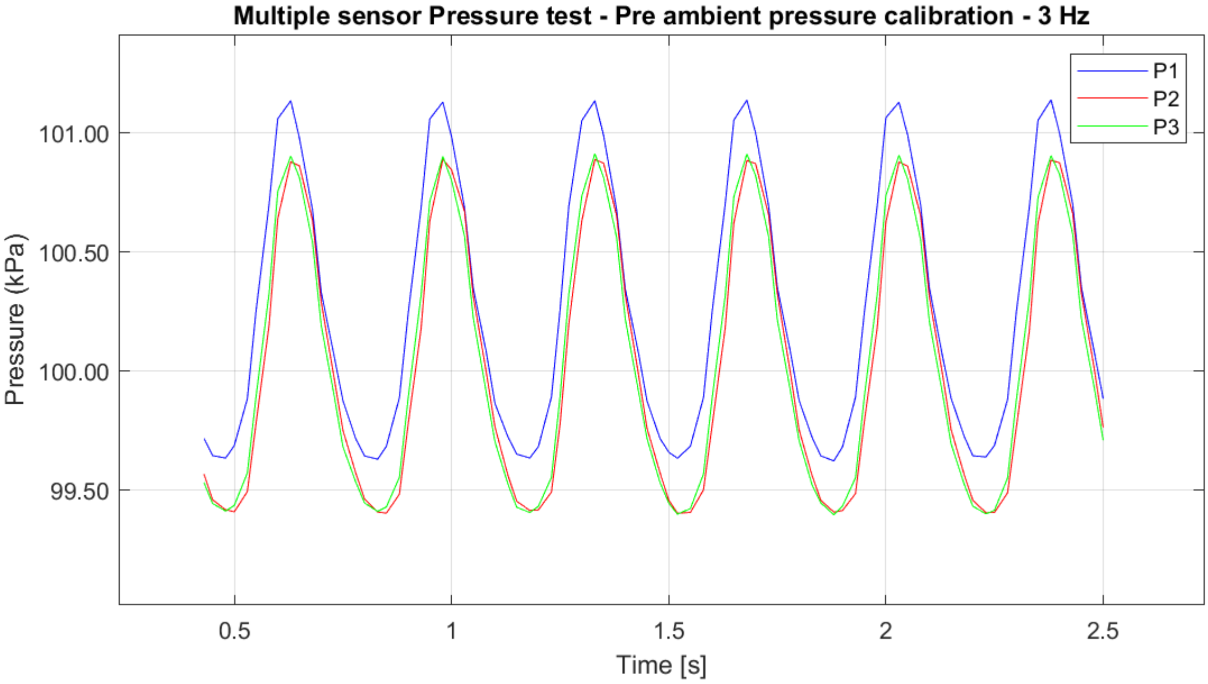


Figure 22. Homogeneous pressure test at 3 Hz, without calibrating the sensors. Time in seconds on the x-axis and pressure in kPa on the y-axis.

After calculations, using P2's ambient pressure as "zero" and adjusting P1 and P3, results are once again drawn up in figure 23, figure 24 and figure 25 below:

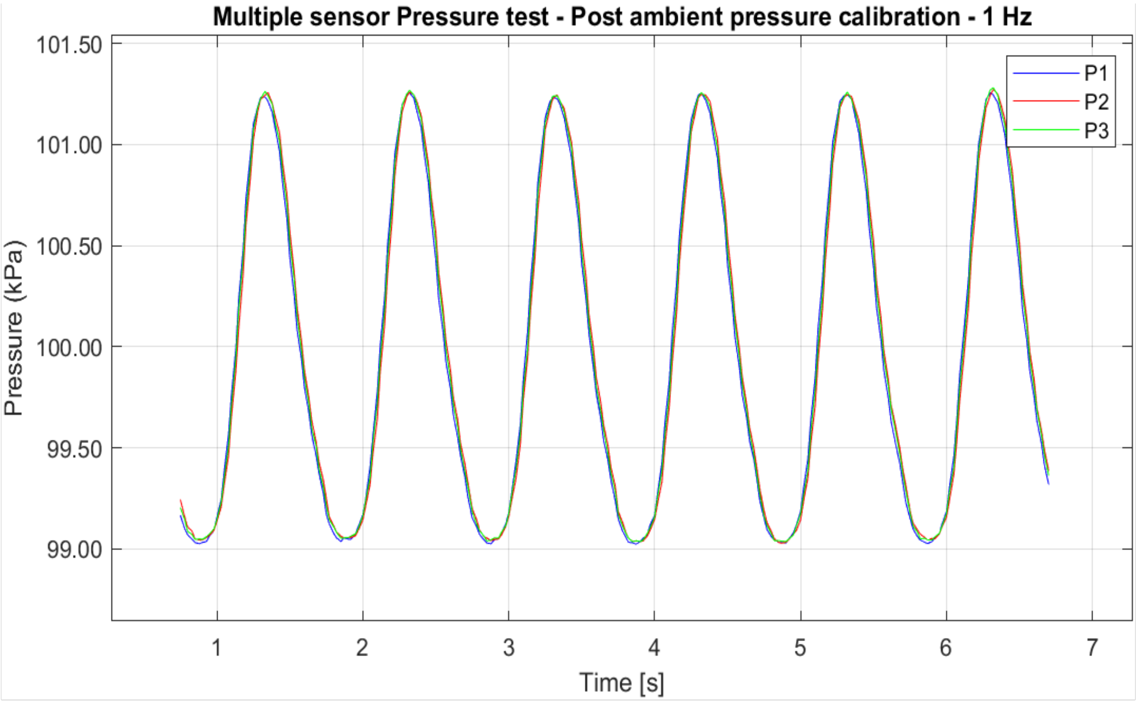


Figure 23. Homogeneous pressure test at 1 Hz, with calibration of the sensors. Time in seconds on the x-axis and pressure in kPa on the y-axis.

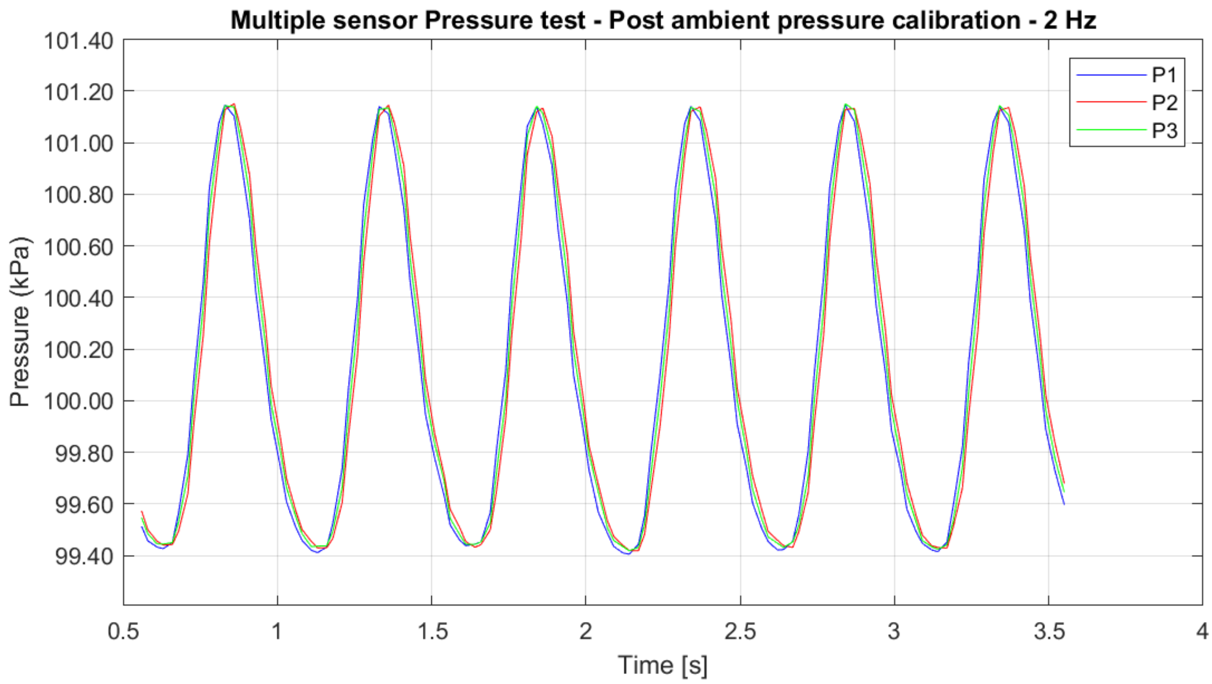


Figure 24. Homogeneous pressure test at 2 Hz, with calibration of the sensors. Time in seconds on the x-axis and pressure in kPa on the y-axis.

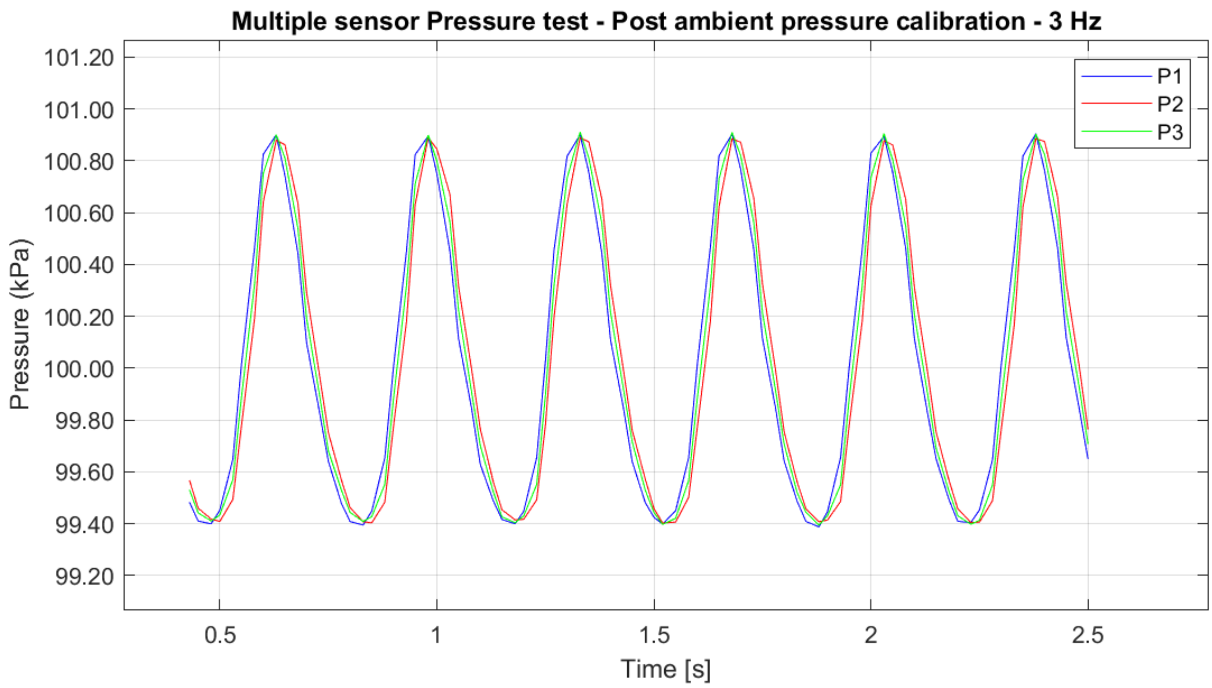


Figure 25. Homogeneous pressure test at 3 Hz, with calibration of the sensors. Time in seconds on the x-axis and pressure in kPa on the y-axis.

4.5.3 Speaker power output

Results of the voltage measurements when using a speaker to convert the pressure pulses into electricity is displayed in figure 26.

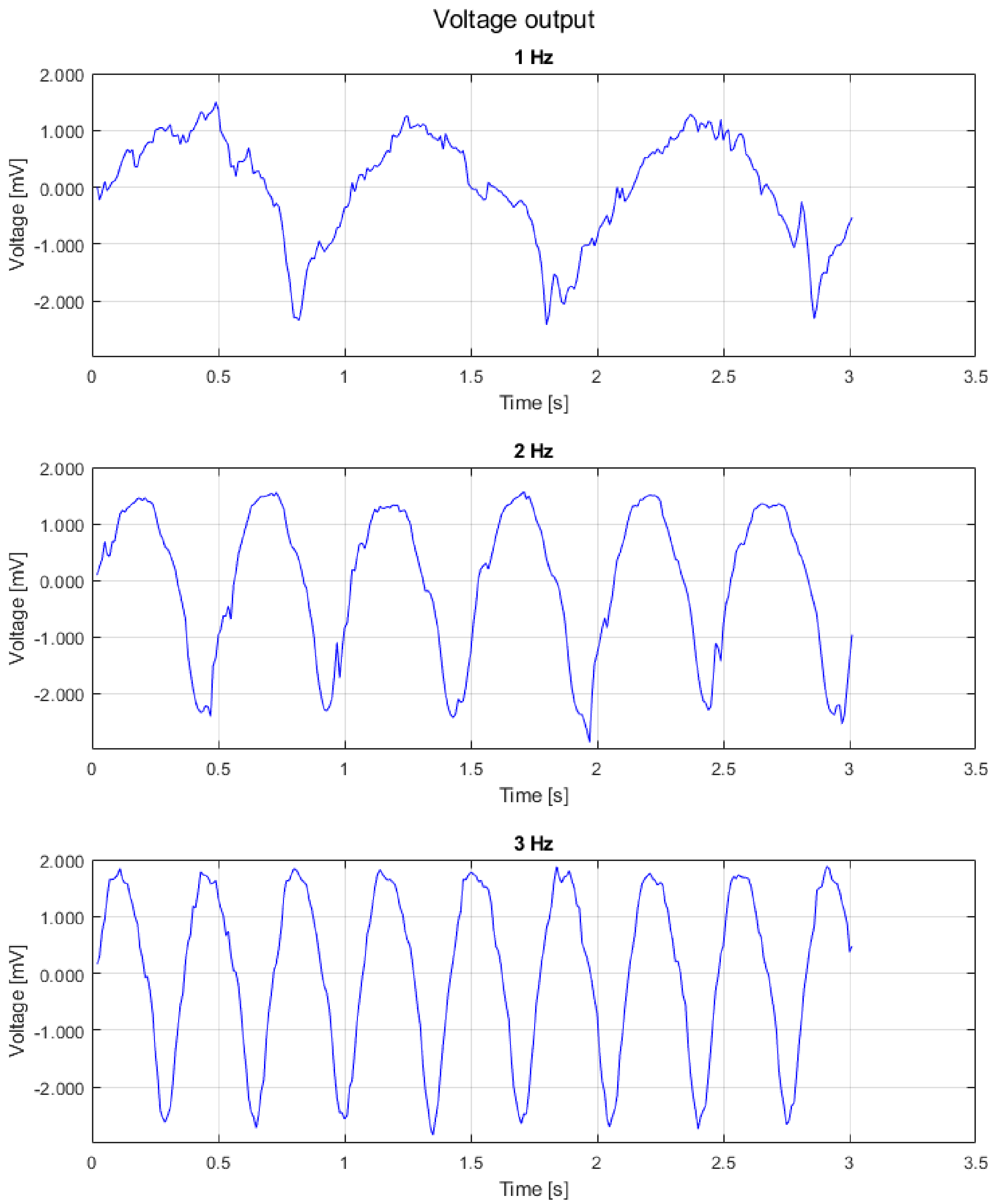


Figure 26. The voltage drop measured over a 10 ohm resistor at 1, 2 and 3 Hz. Time in seconds on the x-axis and the measured voltage in mV on the y-axis.

The maximum- and minimum values from the sinus curves in figure 26 are presented below in table 5.

Table 5. Maximum- and minimum vales of voltage.

	U_{max} [mV]	U_{min} [mV]
1 Hz	1.2	−2.4
2 Hz	1.5	−2.5
3 Hz	1.9	−2.6

Using equations (2) – (4) the power output for each rotating frequency can be calculated and presented in table 6 below.

Table 6. Result of power output.

	$U_{amplitude}$ [mV]	U_{RMS} [mV]	P [μW]
1 Hz	1.8	1.273	0.162
2 Hz	2	1.414	0.200
3 Hz	2,25	1.591	0.253

4.5.4 Water column output

The volume-pressure work of the water column at MAX temperature and 1 Hz is presented in table 7.

Table 7.Power output from the water- column test

ΔL [cm]	r [cm]	ΔV [m^3]	p [kPa]	P [W]
16,5	2,5	0,0000032	100	0,32

4.5.5 Relationship between pressure, frequency and temperature

The test to investigate the correlation between pressure, frequency and temperature is displayed in table 8. This is further visualized in figure 27 and figure 28.

Table 8. Pressure difference depending on combination of frequency and temperature. The temperature differences representing heat setting 4, 5 and MAX on the hot plates.

	1 Hz	2Hz	3Hz
ΔT = 38˚	1160.09	682.92	635.71
ΔT = 55˚	1240.4	794.04	715.32
ΔT = 75˚	1860.17	1226.47	1065.31

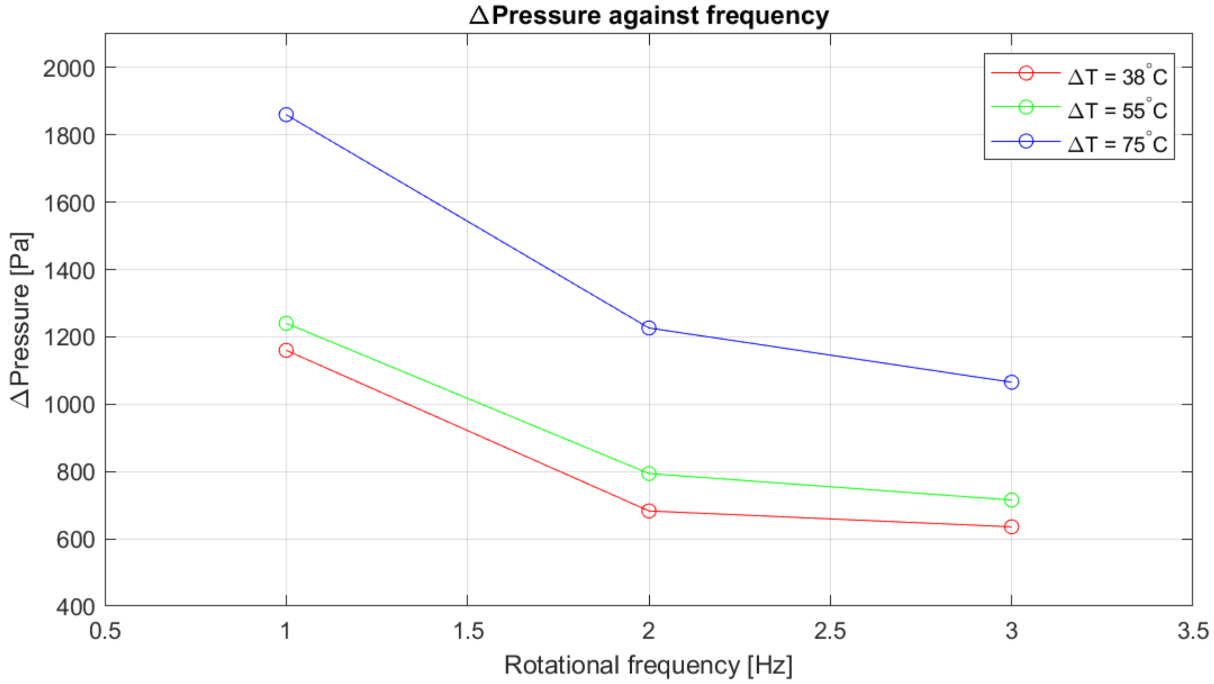


Figure 27. Pressure difference against frequency at different temperature differences representing setting 4,5 and MAX on the hot plates.

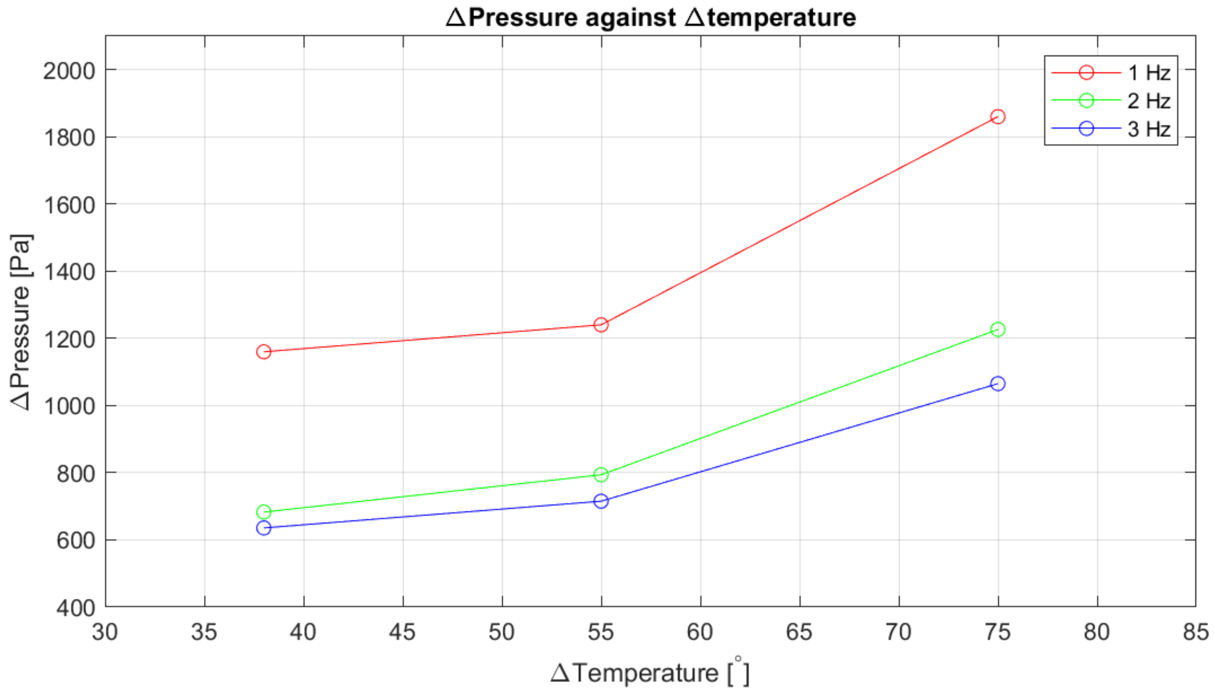


Figure 28. Pressure difference against temperature difference at frequency 1, 2 and 3 Hz.

5. Discussion

In this chapter the results will be discussed. This includes reasoning for certain choices, performance of different components and underlying reasons for the test results.

5.1 Sealant

The chosen sealant was a high temperature acetoxy silicone. It is resistant to temperatures over 250 °C and has a curing time of 24 hours and was applied using a caulk gun. It costed 100 SEK and is resistant to aging. The reason this sealant was chosen over the others was mainly because of its high temperature resistance and wide use. It was a useful choice in the sense that it in fact sealed the air from leaking from the LabModel v3 when pressurized.

Polymer based sealants are used for moving or expanding joints and even though there might be some expansion when the GREC is in use, no polymer-based sealant could support such high temperatures.

Epoxy was after further research discovered to be used for lamination and was not applicable to this problem. Without prior experience of sealing, an easier method was chosen. Only one compound is used when applying silicone but both resin and hardener are required when applying epoxy.

Acrylic sealant was not as easily available as silicone and the acrylic sealants that could be did not support high temperatures.

Phenolic sealants were not available at the local hardware store or online. Therefore, it was not chosen.

5.2 Leakage

Using the infra-red camera to identify the leakage became more of a challenge than anticipated. The camera had a low update frequency which meant that it was difficult to see in real time where leakage occurred. The GREC is made up of different materials, which also has different heat transfer properties. It was therefore difficult to understand if the heat leakage was due to the hot air leaking or heat leaving through conducting. The results from the infra-red camera were that heat left the GREC through the screws. The screws are made of metal and are therefore better at conducting heat than the rest of the GREC. This does not prove that leakage occurred through the screws.

It was discovered that the GREC would become tighter as it was heated. It has been mentioned that the main leakage occurred by the stepper motor mount. When the stepper motor was completely removed and considerable leakage would be anticipated to occur, no major leakage was detected. The expansion of the materials of the GREC that had reduced the leaking.

With all of this said, the LabModel v3 actually got sufficiently air tight, with the air leaking at only around one or two millimeters of the water column test after 10 seconds when pressurized by 3 kPa. This is deemed to be tight enough at this part of the project.

5.3 Failing components: sensors and motor, I2C

Failing equipment has been reoccurring during testing of GREC. Equipment that was been prone to break were the SparkFun MicroPressure sensors, and the stepper motor. Several SparkFun sensors have been needed to be changed. The initial five already attached to GREC worked as intended, but they started to fail after some time. New ones were ordered and worked well in the beginning but also quickly failed. The reason for this was never discovered. It was believed that the sensors had been damaged when connected to 5V when they were rated for 3.3 V. The new set of sensors were only connected to 3.3 V, yet they still failed at the same rate as the first sensors.

Another probable reason that the SparkFun sensors stopped working could be related to the operating temperature of the GREC. From the results we know that the temperature reached 82°C in middle of the GREC when on MAX setting. According to their datasheet, the SparkFun sensors should tolerate temperatures up to 85°C. Some of the SparkFun sensors were placed slightly above the middle, at the edge of the hot fins, meaning it is likely they exceeded 85°C. However, the sensors below the middle, further away from the hot fins, also stopped working.

When further experiments are done on the LabModel v3, it is imperative to find the reason for the failed sensors and to replace them if necessary.

The stepper motor performed well until it was exposed for prolonged periods of heat. The engine has an operating temperature of -10 ～ +50℃ and the outside temperature of the motor was estimated to be up to 70℃. The holding torque of the GREC was reduced after this incident and it could not rotate the shutter as intended. The motor reached such a high temperature when having the heating elements on for a long time. Ways to cool down the motor should be developed in the future. When using the GREC it should not be heated for several hours, 3-4 hours are the maximum until the motor gets too hot, with the current set up. The current motor could also be replaced with a more heat resistant motor that is suited for this application. The driver circuit could only provide 4 A of currents while the stepper motor could operate fully with 6 A. The torque provided by the stepper motor is directly proportional to the input current. It was this in combination with the heating damage that made the motor perform sub-optimal.

The initial idea was to use the Raspberry Pi as a master and control the stepper motor with Arduino 10 and to use Arduino 12 to sample the pressure sensors. The samples would then be sent to the master Raspberry Pi by I2C. During the early stages of the project, when testing if everything operated as intended, problems occurred. When the RS rotated at 1 Hz it took one sample every time it passed a hall sensor, and therefore sampling at 4 Hz (4 times per rotation). However, the I2C communication failed when increasing the rotational speed to 2 Hz and the sampling rate to 8 Hz. I2C supports speeds of 100Kbps, one sensor value is only 64 bits, and the bus should therefore be able to handle the data. Since it was not necessary to use the I2C bus to progress, it was not used in the extent it was initially intended. Instead, the values from the sensors were collected and written in the serial monitor of Arduino 12 to be analyzed.

5.4 Tests

During this section, the different test results will be presented and discussed.

5.4.1 Repeatability

The repeatable tests according to graphs 16, 17 and 18 show that it is difficult to get consistent results. The tests are similar but not exact. This is hypothesized to be because of the difficulties in applying the same conditions to the tests. The difference in ambient pressure was accounted for, however, the temperatures of the GREC were more difficult to determine. The side, middle and top of the hot fin was investigated, and their temperature determined. These temperatures were correct, but other temperatures such as the cold fin were not accounted for. The hot plate used for heating was not precise as they only had heat settings from 0 to 6 and the only way to measure the real temperature was to use an infra-red camera which had its limitations. For instance, it was difficult to precisely measure the correct location due to it only being held by hand. If measurements were made with centimeters apart, the temperature difference would be noticeable and therefore there is also some uncertainty to what the conditions truly are. Graphs 16, 17 and 18 show that the tests produce remarkably similar pressure results. This means that the temperature difference was small enough to produce similar pressure inside the GREC. When studying graph 17 and 18, the tests seem to be going out of phase as the tests continues. This is because the sampling rate of the different tests differed, and this is a direct effect of the rotational frequency differing. Problems with the Stepper motor started to arise as the tests were undergoing. This resulted in a stepper motor that stuttered, making it difficult to correctly control the rotational frequency.

Despite the curves being out of phase and despite only having 2 measurements instead of 3, the fact that the same pressure was achieved proves that the tests of the GREC are repeatable.

The current tests can be compared with old tests done by nilsinside AB and be seen in figure 5. The old tests used a rotational frequency of 1 Hz and a hot fin-temperature of 75°C. When comparing those results to tests that were performed under similar conditions they match well. Tests that measured a temperature of 79°C (352 K) in the middle of the hot fin and used 1 Hz rotational frequency reached a pressure difference of 1240 Pa according to table 8. The pressure difference nilsinside AB achieved was 1249 Pa which is close enough to point towards reproducibility.

5.4.2 Homogeneous pressure test

1 pressure sensor failed, meaning that only 3 could collect data. This was not optimal but was determined to be acceptable. It was acceptable because a pressure difference in the GREC would still be noticed when using 3 sensors instead of 4. In figure 23, figure 24 and figure 25 the pressure was recorded in three different places in the LabModel v3, the results show that the pressure is equal in the different measuring points when rotating at a slower pace, 1 Hz. When the rotational speed increases, reaching 3 Hz, the pressure starts to vary some, it is small but still noticeable. This can be a result of different reasons. For one, there are a lot fewer samples per rotation with a higher rotating frequency. For example, a rotational speed of 3 Hz had 14 samples/rotation compared to the 1 Hz rotational speed where 40 samples/rotation were

taken, giving a more precise graph at lower rotational speeds. Also, the update frequency at a higher rotating frequency can be slower for some sensors and therefore give some differences. This can also be a result of the pressure not filling the entire volume at fast enough pace, which was a thesis discussed prior to the tests. Furthermore, the same pressure is reached across the different pressure sensors within 2 hundredths of a second at rotational frequency 3 Hz, which is considered negligible since the LabModel v3 can only rotate at a maximum 3 Hz with the current motor. This means that taking samples from one sensor is reasonable, in order to take more samples per rotation and therefore make more precise conclusions.

When comparing this result to previous tests done with several pressure sensors, like the one done in figure 5, it can be determined that it also proved that the pressure was homogeneous in the GREC but due to too infrequent sampling frequency it could not be proved. The old test used a sampling frequency of 4 Hz while the new test used 29 Hz for a rotational frequency of 1 Hz. The unclear result from the old test is now verified by the new test to be correct.

5.4.3 Speaker and water column output

The power output from the speaker was small. Less than 1 µW of power dissipated through the 10Ω resistor. This method produces too little power to be usable in any way, but the result achieved in table 6 can still provide vital information about the behavior of the GREC. The speaker is not made for converting mechanical energy into electrical energy and has turned out to be an inefficient way to extract energy. The power output was only measured for the max temperature setting because that ensured that the largest voltage drop across the resistor would occur and that would therefore be easier to measure.

The theory about volume-pressure work $w = \Delta V * P$ says that the output power should increase linearly with increased volume difference. The volume difference should in turn be proportional to the pressure difference inside the GREC. The following calculations will see if this is true.

According to table 6, the power output increased from 0,162 µW to 0,2 µW when increasing frequency from 1 Hz to 2 Hz. It further increased to 0,253 µW when increasing frequency to 3 Hz. This is an increase to 123% and 156% from the power output at 1 Hz. The pressure difference inside the GREC decreased to 66% when increasing frequency from 1 to 2 Hz at MAX- temperature setting. This decreased to 57% when the frequency became 3 Hz. When considering that the pressure pulses will perform work at 2 or 3 times per second at 2 and 3 Hz respectively, the potential power output increase will be $66 \cdot 2 = 112\%$ and $57 \cdot 3 = 171\%$. This can be compared to the 123% and 156% power increase that was achieved. These implies that the power output is proportional to the pressure inside the GREC as the theory about the volume-pressure work also hypothesized.

The power output done by the water column can directly be compared to the power output of the speaker to visualize its inefficiency. The power output achieved by the water column at 1 Hz and max temperature was 0,32 W and the power output for the speaker in the same conditions was 0,162 µW. This clearly show how inefficient speaker as a power output device really are.

5.4.4 Relationship between pressure, frequency and temperature

It has been thought, according to Charle´s law that the pressure inside the GREC is linearly proportional to the heat of the hot fin if the cold fin stays at a constant temperature. Graph 28 shows how much the pressure increases compared to the temperature increase. If temperature setting 4 is the reference, table 8 show that temperature increased with 5% and 12% when increasing setting to 5 and MAX respectively. The following pressure increase were somewhere 7-16% and 60-80% for setting 5 and MAX. The fact that the pressure increased much more than the temperature points towards that the pressure increase in the GREC is not linearly proportional to the temperature.

The pressure difference significantly decreases when the rotational frequency increases as can be seen in figure 27. If the rotational frequency 1 Hz is the reference, the rotational frequency increased by 100% when using 2 Hz and by 200% when using 3 Hz. The pressure difference decreases by 41-34% when using 2 Hz and 45-43% when using 3 Hz. This implies that the pressure stays high despite the rotational frequency increasing. This might occur because the forced convection of air leads to an increased heat transfer as the frequency increases. This means that the air will heat up more quickly at higher frequencies. Since the pressure is directly proportional to power generation of the GREC, this further implies that the GREC can produce more power at higher frequencies. It will also become more efficient at high frequencies since the power output will increase more than the input power to rotate the RS will.

5.5 Method discussion

The general approach to answer the research questions was to approach one problem at a time, as a team. This approach had its benefits and flaws. The benefits included giving more focus on the more important problems, such as running the GREC or getting it sealed, which ensured that these problems got solved in time. It also made it easier to have discussions exploring other possible solutions for the problems. The flaw of the approach was that if the problem did not get solved it could result in a bottleneck, in other words, there would not be time left for the remaining problems. Luckily, this was rarely the case during the project. If the group got stuck on a specific problem, it got resolved rather fast.

Due to time restrictions, the chosen methods to answer the research questions may not have been the most optimal. The reason being that choices had to be made generally fast to ensure the deadline was met. This can be seen in the result of the power output as both methods had their weaknesses. The speaker was inefficient, and the water column worked better for lower frequencies. If given more time, the group could have researched the piston alternative more, which may had resulted in a better solution to measure the power output.

6. Conclusion

6.1 Research questions

Q1: How to optimize the electronics in the GREC LabModel v3 and make the GREC LabModel v3 user-friendly?

The electronic equipment in the GREC LabModel v3 could be optimized by shortening cables as well as fastening them with cable ties. The GREC LabModel v3 was made user friendly by the implementation of a Raspberry Pi to easier control the stepper motor as well as constructing an instruction manual. The manual is a step-by-step instruction of how to get the GREC LabModel v3 running and sampling tests. This will hopefully help the following research groups to quickly get it running and begin research.

Q2: How do you identify pressure leakage in the GREC LabModel v3? How do you seal the leakage so the LabModel v3 withstands a pressure of 3 kPa and temperatures of 200 degrees Celsius?

Locating leakage was most easily done by pressurizing the GREC LabModel v3 by blowing through the mouth into a plastic tube that was connected to the GREC LabModel v3. By constructing a water column, the pressure loss could easily be determined. By covering each part of the GREC LabModel v3 one at a time with plastic film and then testing the pressure loss, it was possible to locate where most leakage occurred.

To cover up these leaks and to be able to withstand temperatures of 200°C as well as a pressure of 3 kPa, a high temperature acetoxy silicone is a great choice due to its cheap cost and quick curing time.

Q3: How does the pressure of GREC LabModel v3 change with temperature and rotational frequency and how is this related to the power output?

The pressure difference was not linearly correlated to the temperature nor the frequency. As the temperature increased, the pressure difference increased exponentially. The same was true for rotational frequency as the pressure remained high even though the frequency was doubled and tripled. The power output increased linearly with the pressure difference GREC LabModel v3.

Q4: What is a method for measuring power output for GREC LabModel v3?

Methods for measuring power output was a speaker and a water column. Water column only worked for low frequencies but did generate a large power output from the GREC LabModel v3. The speaker was reliable and could be used for all frequencies but did not produce any usable amounts of power.

Q5: Is the pressure inside GREC LabModel v3 homogeneous?

The pressure inside GREC LabModel v3 has been proved to be homogeneous for temperature gradient between the hot and cold fin of 80 degrees Celsius and 3 Hz frequency.

6.2 Future work

If research will be conducted on the GREC LabModel v3 in the future, there are some things that should be continued upon:

The new control method to control the whole GREC LabModel v3 through a Raspberry Pi did not work as intended and an alternative solution had to be used to get the project moving forward. Understanding why it did not work is vital and first then can a solution be produced.

Problems with the pressure sensors occurred and no explanation to why was ever given. The following group that works on the GREC LabModel v3 should try to fix this problem to make testing easier as well as minimize costs from needing to order new components too often.

The stepper motor started to fail towards the end. The RS was difficult to rotate, and it has been hypothesized to be because of lubricant in the ball bearing getting too hot and melting resulting in too high friction.

Investigating whether the GREC-idea can be effectively scaled to enhance power output is a crucial area for future research. Exploring the implementation of multiple levels of RS and different sizes of the LabModel represents a promising path in this regard.

7. List of references

[1] nilsinside AB, “Believers in sustainable Innovation,” [Online]. Available: https://www.nilsinside.com/nilsinside/Company-EN.html. [Accessed 26 02 2023].

[2] Nilsinside AB, “The technology behind the GREC,” [Online]. Available: https://www.nilsinside.com/nilsinside/Technology-EN.html. [Accessed 26 02 2024].

[3] M. Eriksson, O. Magnusson, L. Haglund, J. Malmdal and G. Edholm, “1. Theoretical Proof Of Concept For The Green Revolution Energy Converter: Development of a mathematical model, material analysis and physical model improvements,” Linköping University, 2022.

[4] J. Hagströmer, M. Reijm, O. Torsteinstrud and V. Stenholm, “Investigation of the internal heat transfer of GREC.,” Linköping University, 2023.

[5] E. Andersson, E. Gustafsson, M. A. Bolstad, M. Eriksson and W. Fager, “Thermal Investigation of the Green Revolution Energy Converter: A study on the heat transfer within the GREC in regards of temperature distribution and heat rate,” Linköping University, 2023.

[6] J. Åsmo, J. Ross and V. Jeirud, “The thermodynamics of the GREC version 3: Investigation of engine performance and energy conversion,” Linköping, 2023.

[7] O. Brodin, L. Villhelmsson and M. Hollsten, “Development of a mechatronic solution for a new type of heat engine: The selection of relevant electronic components and control system for the mechatronics for the lab model of the new heat engine the Green Revolution Energy Converter,” Linköping University, 2023.

[8] A. Toader, I. Hellström and S. Sandström, “Design and development of a working prototype for a revolving heat engine,” Linköping University, 2023.

[9] Acandia, “VAD ÄR EN O-RING?”.

[10] PHELPS, “Which industrial sealant is right for your application,” [Online]. Available: https://www.phelpsgaskets.com/blog/which-industrial-sealant-is-right-for-your-application. [Accessed 10 05 2024].

[11] Robotics and Automatation News, “5 Sealing Technologies Commonly Used by Manufacturers Today,” [Online]. Available: https://roboticsandautomationnews.com/2020/05/22/5-sealing-technologies-commonly-used-by-manufacturers-today/32395/. [Accessed 23 04 2024].

[12] Chemical Concepts, “What’s the Difference? Acetoxy vs. Neutral Cure Silicones,” 12 12 2018. [Online]. Available: https://www.chemical-concepts.com/whats-the-difference-acetoxy-vs-neutral-cure-silicones/. [Accessed 23 04 2024].

[13] Anusha, “ElectronicsHub,” 4 3 2024. [Online]. Available: https://www.electronicshub.org/stepper-motor-driver-circuit/. [Accessed 12 4 2024].

[14] boris, “PoBlog,” 26 03 2021. [Online]. Available: https://blog.poscope.com/stepper-motor-driver/. [Accessed 10 04 2024].

[15] M. McCauley, “AccelStepper,” [Online]. Available: https://www.arduino.cc/reference/en/libraries/accelstepper/. [Accessed 12 03 2024].

[16] Stan, “42 Bots,” 22 11 2014. [Online]. Available: https://42bots.com/tutorials/stepper-motor-wiring-how-to/. [Accessed 2 12 2024].

[17] J. Buts, “HowToGeek,” 28 9 2021. [Online]. Available: https://www.howtogeek.com/754492/what-is-raspberry-pi/. [Accessed 20 04 2024].

[18] S. Cambell, “BASICS OF THE I2C COMMUNICATION PROTOCOL,” Circuit Basics.

[19] LibreTexts, “The Ideal Gas Law,” [Online]. Available: https://chem.libretexts.org/Courses/Sacramento_City_College/Chem_400%3A_General_Chemistry_I/05%3A_Gases/5.04%3A_The_Ideal_Gas_Law. [Accessed 24 03 2024].

[20] LibreTexts, “Pressure-Volume Work,” [Online]. Available: https://chem.libretexts.org/Bookshelves/Physical_and_Theoretical_Chemistry_Textbook_Maps/Physical_Chemistry_(LibreTexts)/19%3A_The_First_Law_of_Thermodynamics/19.02%3A_Pressure-Volume_Work. [Accessed 25 04 2024].

[21] Storables.com, "https://storables.com/articles/how-to-turn-an-electric-motor-into-a-generator/," 22 02 2024. [Online]. Available: https://storables.com/articles/how-to-turn-an-electric-motor-into-a-generator/. [Accessed 22 04 2024].

[22] R. Murray-Smith, "1392 A Speaker As A Generator," Youtube, 2022.

[23] TRIAD MAGNETICS, "Understanding the Maximum Power Theorem," 12 01 2024. [Online]. Available: https://info.triadmagnetics.com/blog/maximum-power-theorem. [Accessed 21 4 2024].

[24] ElectronicsTutorials, "Power in AC Circuits," [Online]. Available: https://www.electronics-tutorials.ws/accircuits/power-in-ac-circuits.html. [Accessed 12 04 2024].

[25] NIlsinside, "Li.U Lab Model 3 Heat and Pressure Test," 17 6 2023. [Online]. Available: https://nilsinside.com/nilsinside/BREC_V2/LiU_LabModel3heat_and_pressure_test.html. [Accessed 21 04 2024].

[26] ISixSigma, "Repeatability vs. Reproducibility: What's the Difference?," 17 10 2022. [Online]. Available: https://www.isixsigma.com/methodology/repeatability-vs-reproducibility-whats-the-difference/. [Accessed 13 04 2024].

[27] Nilsinside, "Varamon Heat and Pressure Test 2023-08-02 20:20 of the Li.U Lab Model v3," 02 08 2023. [Online]. Available: https://nilsinside.com/nilsinside/BREC_V2/LiU_LabModel3_20230802-2020.html. [Accessed 20 03 2024].

[28] Biltema, "Silikon, högtemperatur," [Online]. Available: https://www.biltema.se/bygg/farg/fog-och-tatningsmassa/silikon/silikon-hogtemperatur-2000017141.

[29] DFRobot, " TB6600 Stepper Motor Driver User Guide," [Online]. Available: https://www.makerguides.com/wp-content/uploads/2019/10/TB6600-Manual.pdf.

[30] DFROBOT, "TB6600 arduino Stepper Motor Driver," [Online]. Available: https://wiki.dfrobot.com/TB6600_Stepper_Motor_Driver_SKU__DRI0043. [Accessed 20 05 2024].

A Appendix A

Codes that were used to run motor and sensors:

TB6600.ino

```
#include <AccelStepper.h>
#include <Wire.h>

#define SLAVE_ADDRESS 0x10
#define dirPin 3       // To DIR pin1 lowest left on 8825 with
red LED
#define stepPin 2     // To STEP pin2 low left on 8825, one
pulse is one step
#define enablePin 4    // yellow LED upper left corner 8825
HIGH no tension to coils

int number = 0;
int ACCELERATION = 200;
int microsteps= 8;
int stepsPerRev=200;
int numberOfHertz=1;
int MAX_SPEED = (microsteps*stepsPerRev*numberOfHertz);

AccelStepper stepper(AccelStepper::DRIVER, stepPin, dirPin);

void setup()
{
  Wire.begin(SLAVE_ADDRESS);     // join I2C bus as slave,
address defined above
  // Attach a function to trigger when something is received
  Wire.onReceive(receiveData);   // define callbacks for i2c
communication
  stepper.setMaxSpeed(MAX_SPEED);
  stepper.setAcceleration(ACCELERATION);
  digitalWrite(enablePin, LOW); // No tension to coils
}
void loop()
{
  stepper.run();
}

// callback for received data
void receiveData(int byteCount){
  while(Wire.available()) {
  number = Wire.read(); //  read one character from the I2C

  if (number == 0){
    MAX_SPEED = (microsteps*stepsPerRev*numberOfHertz);
```

```
    stepper.setMaxSpeed(MAX_SPEED);
    stepper.moveTo(10000000);
    }
  if (number == 1){
    stepper.stop();
    }
  // change speed
  if (number == 9){
    if(numberOfHertz>1){
      stepper.stop();
      numberOfHertz=1;
      }
    else{
      numberOfHertz=1;
      MAX_SPEED = (microsteps*stepsPerRev*numberOfHertz);
      stepper.setMaxSpeed(MAX_SPEED);
      }
    }
  if (number == 8){
    if(numberOfHertz>2){
      stepper.stop();
      numberOfHertz=2;
      }
    else{
      numberOfHertz=2;
      MAX_SPEED = (microsteps*stepsPerRev*numberOfHertz);
      stepper.setMaxSpeed(MAX_SPEED);
      }
    }

  if (number == 7){
    if(numberOfHertz>3){
      stepper.stop();
      numberOfHertz=3;
      }
    else{
      numberOfHertz=3;
      MAX_SPEED = (microsteps*stepsPerRev*numberOfHertz);
      stepper.setMaxSpeed(MAX_SPEED);
      }
    }
  if (number == 6){
    if(numberOfHertz>4){
      stepper.stop();
      numberOfHertz=4;
      }
    else{
      numberOfHertz=4;
      MAX_SPEED = (microsteps*stepsPerRev*numberOfHertz);
      stepper.setMaxSpeed(MAX_SPEED);
      }
```

```
    }
  if (number == 5){
    numberOfHertz=5;
    MAX_SPEED = (microsteps*stepsPerRev*numberOfHertz);
    stepper.setMaxSpeed(MAX_SPEED);
    }
  //Change acceleration
  if (number == 11){
    ACCELERATION = 100;
    stepper.setAcceleration(ACCELERATION);
    }
  if (number == 12){
    ACCELERATION = 200;
    stepper.setAcceleration(ACCELERATION);
    }
  if (number == 13){
    ACCELERATION = 300;
    stepper.setAcceleration(ACCELERATION);
    }
  if (number == 14){
    ACCELERATION = 400;
    stepper.setAcceleration(ACCELERATION);
    }
  if (number == 15){
    ACCELERATION = 500;
    stepper.setAcceleration(ACCELERATION);
    }
  }
}
```

Link02.py

```
#!/usr/bin/python
# -*- coding: utf-8 -*-
import struct
import time                 # sleep function in main loop and rtc
import math                 # log and sqrt functions
import os
import smbus                # i2c communication RPi3 and Arduino
                            #start stop stepper start & stop via
                            #MOTORstate.txt with php HTML
# use the correct SMbus for I2C on RPi3
sda_pin= 2
scl_pin= 3
bus = smbus.SMBus(1)

address_1 = 0x10
old_var = 1
number = -1    # for the first comparision between MOTORstate
               # set time values and save in start (and stop)
start = time.time()
stop = time.time()
last = time.time()
# convert seconds to year date time format
filetime = time.localtime(start)
filenametime = time.strftime("%Y%m%d-%H%M%S", filetime)

# motorfile is connected to the UI where this program gets
# input
motorfile = '/var/www/html/MOTORstate.txt'
logfile = ("MOTORlog"+filenametime+".txt")   # The logfile in
                                             #csv format

# I2C write to SMbus
def writeNumber(value):     # I2C write to Arduino 10
    bus.write_byte(address_1, int(value))
    return -1

# I2C read from SMbus
def readNumber():            # I2C read from Arduino 10
    number = bus.read_byte(address_1)
    return number

def writeFile(filename, contents):  # writes content to a file
    with open(filename, 'a') as f:
        f.write(contents)          # file handle f is oblect,
                                   # write() is method

# main
while True:
```

```
# user interface interaction i.e. textfile to I2C
    txt = open(motorfile, 'r')      # read txtfile to see if
                                    #new user input
    var = (txt.read())
    txt.close()

    if (var != old_var):
        old_var=var
        number = int(readNumber())
        writeNumber(var)         # send user input to Arduino
        print ("RPi3 sent", var, "on 0x10 to Arduino", end = '
' )
        time.sleep(0.01)   # sleep a hundredth of a second
        last = stop
```

sparkfun_EOC.ino

```
#include <Wire.h>
#include <SparkFun_MicroPressure.h>
#define SLAVE_ADDRESS 0x12

SparkFun_MicroPressure pressureSensor;
int pos=0;
float tryck;
volatile bool trigger=true;
volatile bool EOC;
const int triggerPins[]={2,3};

void setup()
{
  Serial.begin(115200);
  Wire.begin(SLAVE_ADDRESS);  // join I2C bus as slave,
                              //address defined above
  // Attach a function to trigger when something is received
  pressureSensor.begin();
  pinMode(triggerPins[0],INPUT_PULLUP);
  pinMode(triggerPins[1],INPUT_PULLUP);

attachInterrupt(digitalPinToInterrupt(triggerPins[0]),triggerI
nterrupt,RISING);

attachInterrupt(digitalPinToInterrupt(triggerPins[1]),triggerI
nterrupt2,RISING);
}

void loop(){
  if (trigger==true){
    trigger=false;
    pos=pos+1;
    tryck = pressureSensor.readPressure(PA);
    Serial.print(pos);
    Serial.print("   ");
    Serial.print(millis()/1000.000);
    Serial.print("   ");
    Serial.println((tryck));
    delay(10);
    }
}
// callback for received data
void triggerInterrupt(){
  trigger=true;
}
void triggerInterrupt2(){
  pos=0;
}
```

sparkfun_EOC_multi.ino

```
#include <Wire.h>
#include <SparkFun_MicroPressure.h>
#define SLAVE_ADDRESS 0x12

SparkFun_MicroPressure mpr1;
SparkFun_MicroPressure mpr2;
SparkFun_MicroPressure mpr3;
SparkFun_MicroPressure mpr4;

float tryck1;
float tryck2;
float tryck3;
float tryck4;

int pos=0;
float tryck;
volatile bool trigger=true;
volatile bool EOC;
const int triggerPins[]={2,3};

#define TCAADDR 0x70
void tcaselect (uint8_t i) {
  if (i > 3) return;
  Wire.beginTransmission(TCAADDR);
  Wire.write(1 << i);
  Wire.endTransmission();
}

void setup(){
  Serial.begin(115200);
  Wire.begin(SLAVE_ADDRESS);        // join I2C bus as slave,
                                    //address defined above
  Wire.begin(0x70); // Send data to the multiplexer

  // check if sensor is connected to multiplexer
  tcaselect(1);
  if (!mpr1.begin()){
     Serial.print(F("Micropressure Nr.1 detected?\t"));
     Serial.println(F("No"));
    }
  else{
     Serial.print(F("Micropressure Nr.1 detected?\t"));
     Serial.println(F("Yes"));
    }
  tcaselect(2);
  if (!mpr2.begin()){
     Serial.print(F("Micropressure Nr.2 detected?\t"));
     Serial.println(F("No"));
    }
```

```
  else{
     Serial.print(F("Micropressure Nr.2 detected?\t"));
     Serial.println(F("Yes"));
    }
  tcaselect(3);
  if (!mpr3.begin()){
     Serial.print(F("Micropressure Nr.3 detected?\t"));
     Serial.println(F("No"));
    }
  else{
     Serial.print(F("Micropressure Nr.3 detected?\t"));
     Serial.println(F("Yes"));
    }
  tcaselect(4);
  if (!mpr4.begin()){
     Serial.print(F("Micropressure Nr.4 detected?\t"));
     Serial.println(F("No"));
    }
  else{
     Serial.print(F("Micropressure Nr.4 detected?\t"));
     Serial.println(F("Yes"));
    }

  pinMode(triggerPins[0],INPUT_PULLUP);
  pinMode(triggerPins[1],INPUT_PULLUP);

attachInterrupt(digitalPinToInterrupt(triggerPins[0]),triggerI
nterrupt,RISING);

attachInterrupt(digitalPinToInterrupt(triggerPins[1]),triggerI
nterrupt2,RISING);
}

void loop(){
  if (trigger==true){
    trigger=false;
    pos=pos+1;

    tcaselect(2);
    tryck2 = mpr2.readPressure(PA);
    tcaselect(3);
    tryck3 = mpr3.readPressure(PA);
    tcaselect(4);
    tryck4 = mpr4.readPressure(PA);
    tcaselect(1);
    tryck1 = mpr1.readPressure(PA);

    Serial.print(pos);
    Serial.print("   ");
    Serial.print(millis()/1000.000);
    Serial.print("     1: ");
```

```
    Serial.print((tryck1));
    Serial.print("     2: ");
    Serial.print(tryck2);
    Serial.print("     3: ");
    Serial.print(tryck3);
    Serial.print("     4: ");
    Serial.println(tryck4);
    }
}
void triggerInterrupt(){
  trigger=true;
}
void triggerInterrupt2(){
  pos=0;
}
```

B Appendix B

Instruction manual

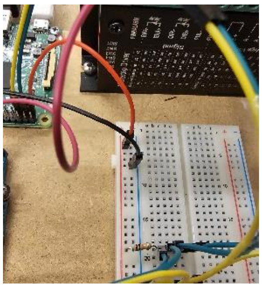
Hardware

The GREC consist of following components:

- Raspberry Pi
- Arduino Uno
- Arduino MEGA
- Stepper motor Nema 23
- Stepper motor driver TB6600
- 4 Hall sensors
- Spark fun micro pressure sensor
- 12 V power supply
- 5 V power adapter to Raspberry Pi
- 9-volt adapters for Arduinos
- Breadboard

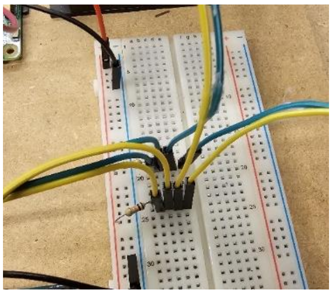
Breadboard

On the breadboard there are connections that may be confusing at first.



Red wires: 3.3V. Black wires: GND.

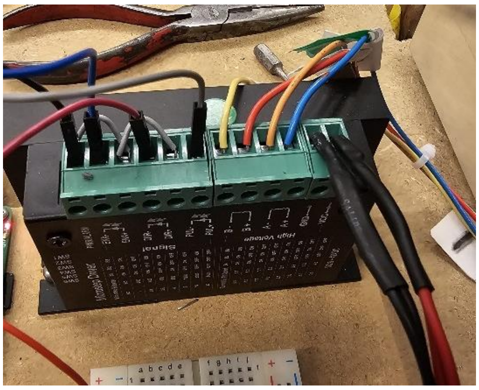
Black wires are ground wires, and all red wires are 3.3 voltage wires. There are 5 V and 3.3V outlets on both Arduino and Raspberry Pi. It is important that 3.3 V is used to make sure no components break down since they only need 3.3 V. Make sure that only one of the Raspberry Pi or Arduinos power the breadboard with 3.3 V. If you are unsure which pin is which on the Raspberry Pi, there are pictures online. All devices should have common ground which is easily connected to ground on the breadboard.



Green wires: SDA. Yellow wires: SCL.

Yellow connections are SCL (serial clock), and green connections are SDA (serial data). These connections are most easily connected like the pictures show. Connected to each line of SDA and SCL there is a 10 kΩ pullup resistor connected to 3.3 V.

Motor driver

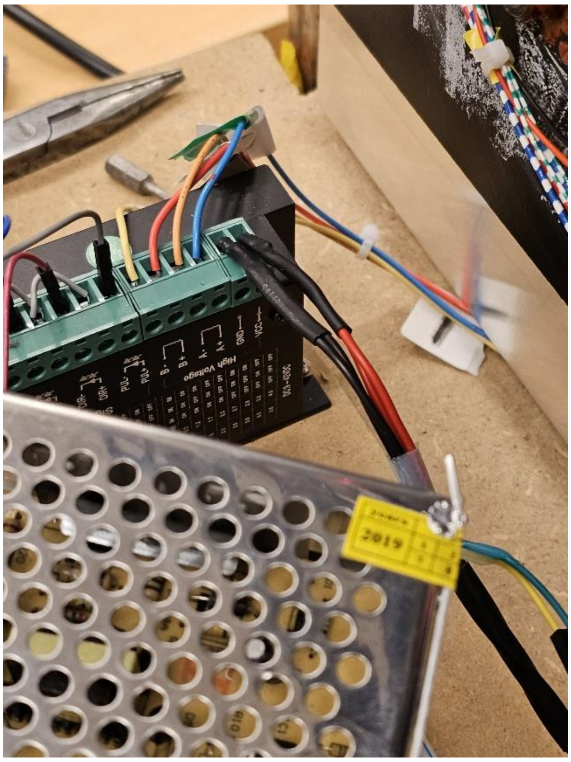


The motor driver is connected to Arduino 10 and the stepper motor. The connections from Arduino 10 are as follows:

- DIR pin- can be used to change directions.
- PUL pin- pin that sends pulses to the driver.
- ENA pin- allows the coils to be turned on and off, enabling the motor shaft able to be turned manually or not.

What pin is connected to what pin on the Arduino is specified in the **“TB6600.ino”** code.

The pins that connect the driver to the stepper motor should be connected as in the picture provided. The stepper motor has colour coded connections where the A- and B coils need to be connected together.



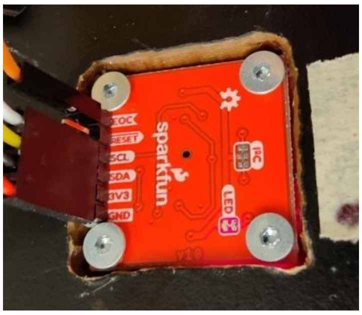
The 12 V power supply should be connected to the driver by the red and black connections provided.

The stepper motor does not require any more connections other than the 4 to the driver. One important thing to be aware of is the fact that the tolerances of the GREC and its components are small. It is easy to have the RS hit the side of the GREC. This also changes as the GREC is heated as the material expands. There might be occasions where you run smoothly at room temperature but get stuck when the GREC is heated. To get it going smoothly again you must adjust the motor screws. Run the motor on 1 Hz and try to unscrew and tighten different sides of the screws to tilt the motor and RS.

Hall-sensor

Above is a hall-sensor, it senses when a magnetic field is close to the black sensor (top of figure) and sends a HIGH when triggered. It is simple as it only needs to be connected to 3.3 V, ground and an output to the Arduino MEGA. The top hall-sensor pin should be connected to the specified trigger pin in the code (should be pin 2). There are marks on the GREC where the upper side of the hall sensors should be placed for correct reading of magnet inside of the rotating shutter.

SparkFun Micropressure sensor



Above is the SparkFun MicroPressure sensor and it has 6 pins; 3.3V, ground, SDA, SCL, reset and EOC. 3.3 V and ground should be connected to the sides of the Arduino. SDA and SCL should be connected to the correct line in the bus according to the color scheme. The EOC pin should be connected to the specified trigger pin in the code (should be pin 3). The reset pin is not used and therefore not required to be connected. In the report it is proven only one pressure sensor is needed in the LabModel v3 since the pressure is always the same in the entire GREC. Therefore, only one sensor is needed, but hardware is set up to handle more sensors if needed, with slight modifications to the software.

Software

Codes

Arduino code "**TB6600.ino**" should be uploaded to Arduino UNO no. 10 (0x10)

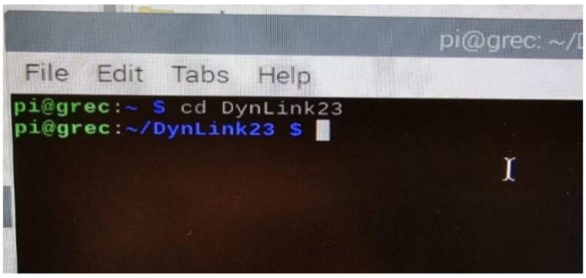
- Code "TB6600.ino" controls the speed and acceleration of the stepper motor.

Arduino code "**sparkfun_EOC.ino"** should be uploaded to Arduino MEGA no. 12 (0x12)

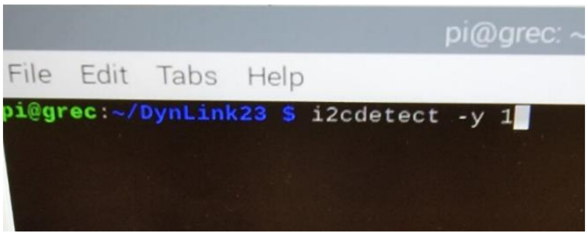
- Code **"sparkfun_EOC.ino"** will sample the pressure every 10th millisecond.

How to run the motor

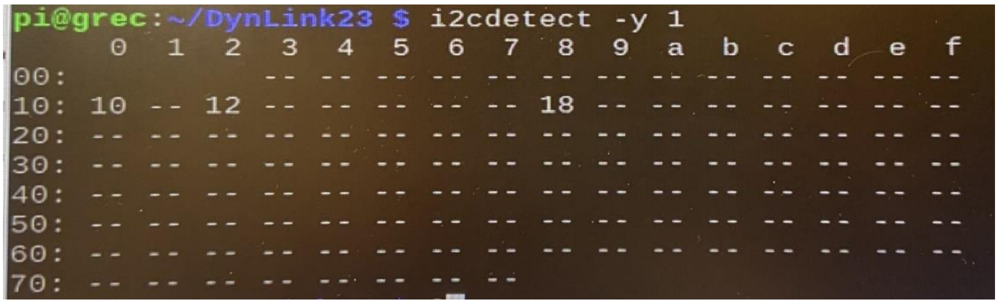
1. Open the *terminal* in the Raspberry Pi 3.
2. Write "**cd DynLink 23**" and press *ENTER* (will enter "DynLink23" folder).



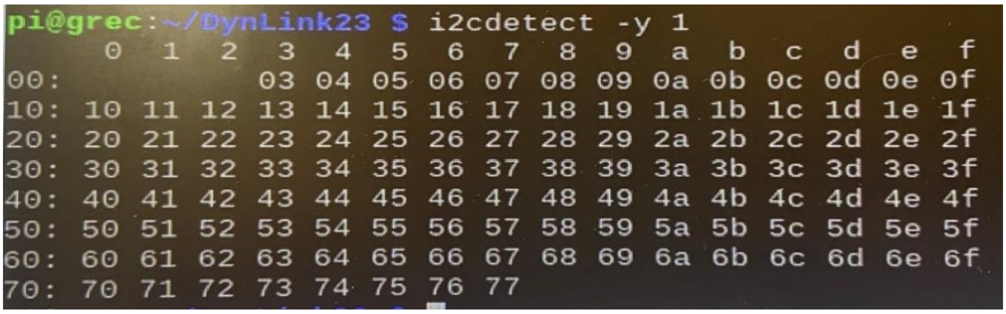
3. Write "**i2cdetect -y 1**" and press *ENTER* (will detect all devices connected to the I2C bus).



You should see addresses written on the screen. If everything is connected correctly the table in the terminal will show the numbers: **10** (Arduino UNO controlling motor), **12** (Arduino Mega controlling the sensors) and **18** (Sparkfun MicroPressure sensor). Like this:



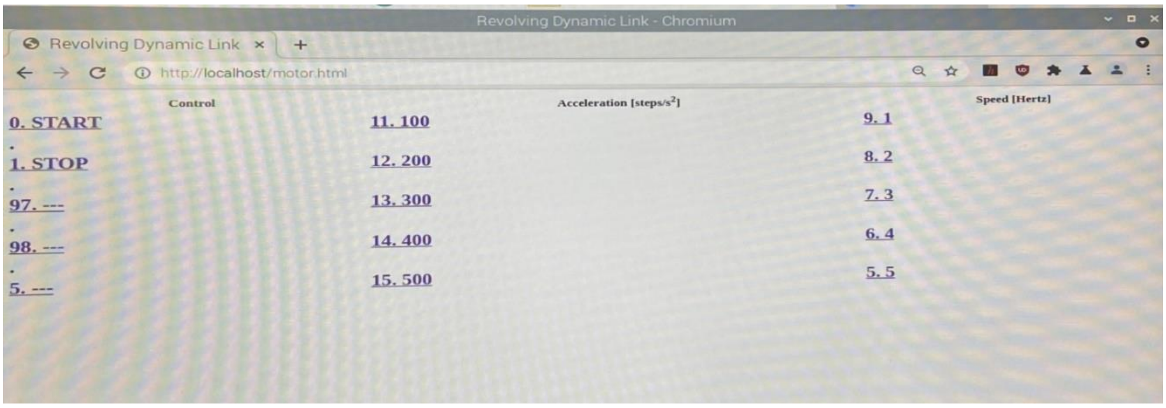
On the other hand, if all addresses show up like this:



Then there might be faulty wiring. If this happens, the SDA line is connected to the ground. SDA will register an address if it sends LOW. If SDA is connected to ground, it will register LOW all the time and show all addresses.
One problem that can be encountered is that the I2C communication would fail if we had floating pins. Floating pins are pins that are in some way not connected to HIGH or LOW, meaning that it is uncertain what it should receive. If there is something wrong with the I2C and you cannot find any explanation, this might be the problem. To avoid this. Try using a pullup resistor.

If everything seems correct and all expected addresses show up, you can continue:

4. Open C*hromium*, the Raspberry Pi browser.
5. Search for "**Localhost/motor.html**".
   This should take you to a local website and should look like this:



This website is a way to control the stepper motor directly without the need to change the “TB6600.ino” code. On the website there are numbers associated with each action. Later when the Raspberry Pi is running a program called “link02.py”, Arduino no. 10 will take a number from a text file called “MOTORstate.txt”. The number written in this text file will be changed depending on the website's input. If you have not read “TB6600.ino” thoroughly you should do that now.

6. In the terminal, write “**python3 link02.py**” and press *ENTER* to run the program “**link02.py**”.

7. Now go to Chromium and click on the desired speed/ frequency (“**1 Hz**” = 60 RPM = 1 rotation/s, “**2 Hz**” = 120 RPM = 2 rotations/s etc.) and desired acceleration, then click “**START**”.

“link02.py” is a python script that communicates with Arduino 10. You should study “link02.py” to get a better understanding of what it does. It is in the folder “DynLink23”.

What “link02.py” does is:

- Reads content of “MOTORstate.txt” and sends this value to Arduino 10 through I2C.
- Arduino 10 will change the setting on stepper motor depending on the value being sent and the stepper motor will act accordingly.

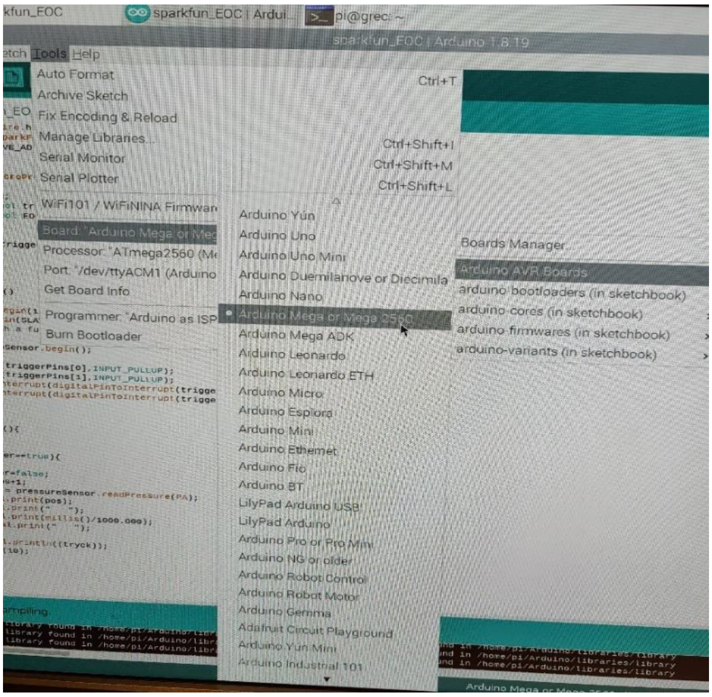
When you are finished and with the tests do this:

8. To stop the test, click “**STOP**” in the Chromium localhost website. The motor should decelerate and finally stop completely.
9. In the terminal, now enter the short command “CTRL” + ”C” which will end the program from running.

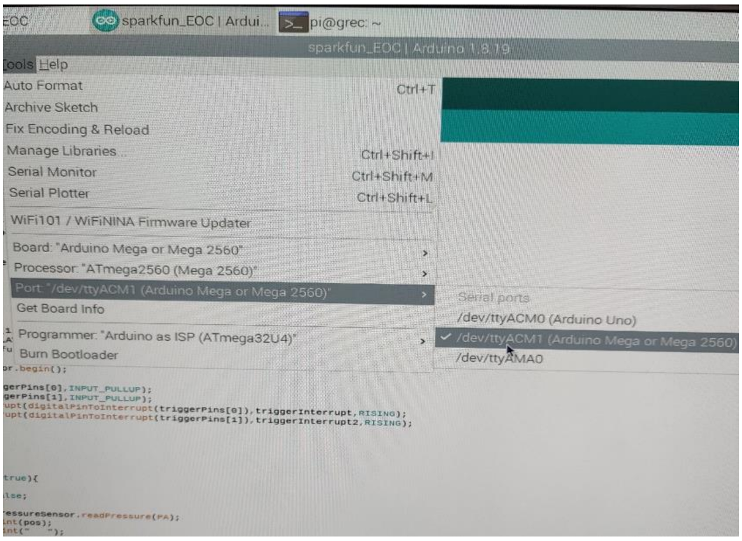
How to collect samples

1. Open the *file manager* on the Rasperberry Pi 3.
2. Open the folder "**Dynlink23**" --> Open the folder "**sparkfun_EOC**" --> Open the file "**sparkfun_EOC.ino**". This will open the code "**sparkfun_EOC.ino**" in the program Arduino IDE.
3. You now want to upload the code to the Arduino MEGA. First complete the checklist:

- Verify the code by clicking on *Verify*.
- Check that the correct board and port is selected. To check the board, click on *Tools*--> *Board*: and select Arduino MEGA:



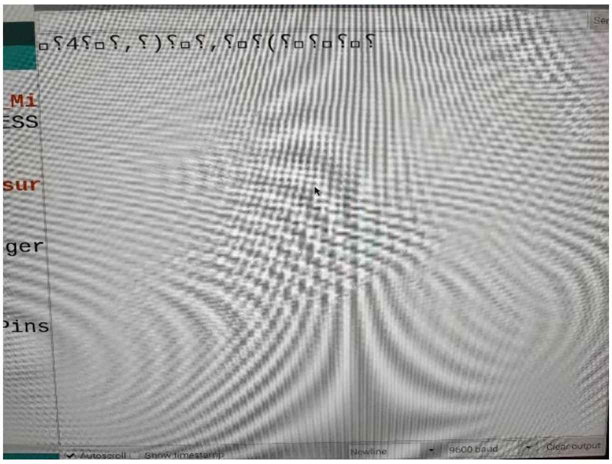
To check the correct port, click on *Tools*--> *Port* and select Arduino MEGA again*:*



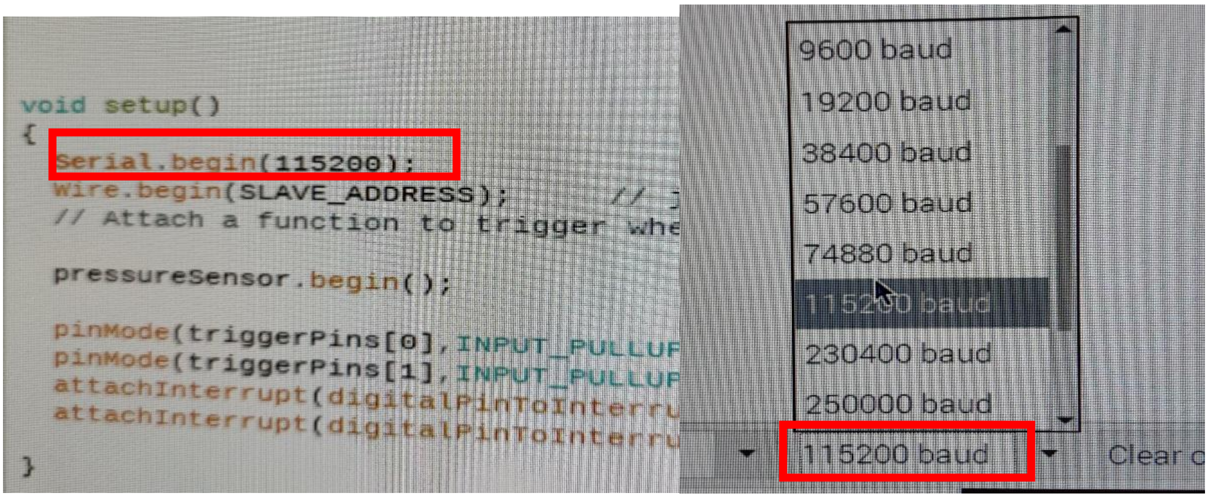
When the checklist is completed, you can now upload the code to the Arduino MEGA.

4. Open the serial monitor by either clicking on *Tools* and then *Serial Monitor* or by pressing “CTRL” + “SHIFT” + “M”.
5. The sampled pressure values should now be printed on the serial monitor, along with the sample number and time. The sample number resets to zero each time it passes the top hall-sensor, to indicate a new revolution.

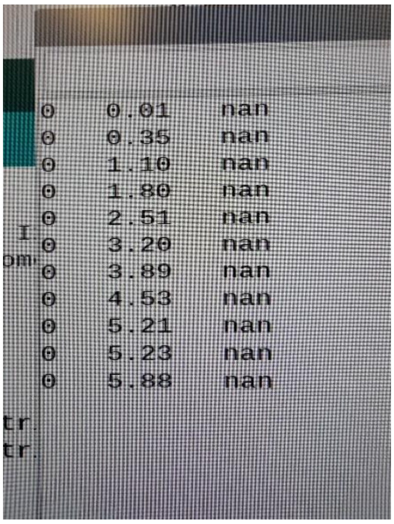
   It is possible two problems occur here:



   If the serial monitor looks like this, the problem is that the serial monitor is not at the same baud rate as the program in printing. To correct this, make sure that you have the same value in “Serial.begin(<baud_rate>)” as the baud rate in the monitor. For example, if the baud rate is 115200, the code and monitor should look like this:



The second problem is if the serial monitor prints “nan”:



“nan” means “not a number” and can occur for two reasons. The first reason is the sensors are not correctly connected, in other words the Arduino MEGA cannot receive any data. To fix this, simply double-check that the wiring is correct and adjust if needed. If the problem remains, it is most likely that the pressure sensor is now broken, and you now need to replace it with a new sensor.

6. If needed, you can reset the sampling time by pressing the Arduino MEGA’s reset button or by uploading the code once again. Tip: When resetting the Arduino MEGA clear the serial monitor by clicking on *CLEAR OUTPUT* to only get the new data.
7. To get the data to a text file simply copy everything printed in the serial monitor into a new text file. When creating a new file on the Raspberry Pi 3, you need to add which file type you want it to be. To create a text file, add “.txt” in the end when naming it. In other words, a name for a new text file could be “example_test_file.txt”.
8. To get the text file to a computer for analyzing simply transfer/ copy the text file to a USB stick.