DISCLAIMER

I hereby certify that the attached report is my own work except where otherwise indicated. I have identified my sources of information; in particular I have put in quotation marks any passages that have been quoted word-for-word, and identified their origins.

Signed ……………………Vasileios Fermelis……………………………………

Date ………………………26/05/2021…..………………………………………

Abstract

Many plants are extremely sensitive to extreme temperatures and need just the right amount of warmth to thrive. This project entails the powering and controlling of an appropriate electrical system that will be able to alter a movable window in order to maintain a greenhouse's internal temperature at the desired level. To ensure constant temperature monitoring and management, the controlling electronics, as well as the transducer and any other circuitry, must be planned, simulated, installed, and checked.

Table of Contents

List of Figures

List of Tables

1. Introduction

The project was created to design, develop and test a control apparatus that would control the internal temperature of a greenhouse. The device would open and close a window automatically with the help of a microprocessor and an electric motor. The project focuses on designing a mechanism that will hold the window open for a reasonable amount of time to let the hot air of the greenhouse escape. Thus, decreasing the internal temperature. When temperature drops, the window will close according to the threshold of the system, allowing the internal temperature to increase to a desired temperature range.

Greenhouse temperatures can vary during the day due to weather changes or other natural phenomena like the greenhouse effect. These temperature changes can cause problems for the plants, vegetables or fruits that need a consistent temperature to grow efficiently. As a result, the system's temperature thresholds should be manually adjusted depending on the plant being cultivated in the greenhouse.

A variety of techniques, components, and control methods will be investigated in order to select practical options for the system functionality. These choices would include, for instance:

1. Selection of mechanical and electrical components after comparisons and evaluations of each.
2. Temperature sensor choice.
3. Temperature control methods.
4. Microprocessor options.
5. Electric motor options.
6. Electric motor selection and their features.
7. Motor driving techniques/selection.
8. Linear actuator design/construction.
9. Window types and choices.
10. Wiring connection techniques.
11. System design/construction.

Aim

To demonstrate a professional and technical system which can control the temperature in a greenhouse by opening and closing the windows using a microprocessor and an electric motor. To demonstrate the effectiveness of this system and its engineering application based on relating literature and in-depth technical analysis.

Objectives

1. Research and analyse previous applications of greenhouse ventilation and how they improved and became more common in today's technology.

    - Reflect on the advantages, disadvantages, limitations and any other technical aspects of any model or design developed as part of this project
    - In-depth research of developed constructions

2. Research temperature transducers, controllers, DC motors and actuators along with software control in order to control the temperature in the greenhouse.

    - Use reliable resources
    - Analyse further the function of microcontrollers, motors, temperature sensors and linear actuators
    - Research different software packages to control successfully the internal temperature of the greenhouse
    - Decide on which components will be added for the project needs

3. Research a wide range of windows for the greenhouse and their functionality

4. Research control methods, i.e. Hysteresis and PID

    - Analyse various temperature and motor control methods
    - Investigate appropriate control techniques for the project needs

5. Attempt to build a demonstration model (pending due to covid)

    - Investigate various methods for designing and constructing mechanical components such as linear actuators
    - Make a circuit board that will include all the components together and connected with the window

6. Produce diagrams, simulations and code which illustrate the system construction process

    - Having a logical and obvious structure by using graphs, tables and appropriate simulation packages

Online resources	• Online University Library
Hardware	• Stepper Motors, protoboard, Arduino board + wire, stainless steel nuts & bolts, temperature sensors, Arduino screen display, protoboard, resistors, wires. • Microcontroller, microprocessors, screwdrivers, drill bits, holesaws, nut drivers, electric drill, depth stop collars, Hex key set, soldering iron, wire cutters, pilers set, Combination Spanner Set, power supply, multimeter, heat shrinks.

Software	Appropriate simulation packages: • Arduino, Solidworks, Excel, EasyEDA, MS Word.
Other resources:	Supervisor, my laptop, laboratory technicians, university laboratories (Covid dependant).

Table 1: List of resources

3. Background Research

Greenhouses are glass structures which protect plants from the cold weather and help them to grow. Automation in greenhouses is an important factor since it ensures an optimal environment for the plants to grow. This project aims to investigate the ventilation of greenhouses, which is crucial in that it keeps balance between indoor and outdoor temperature. During the summer, the ventilation system plays a key role to prevent the indoor temperature from exceeding the outdoor temperature. There are two different types of greenhouse ventilation:natural ventilation and fan ventilation. [1]

- **Natural ventilation** – Natural ventilation depends on the heat rising and air moving. Windows will open when the temperature is above the desired levels and close when the temperature falls.

- **Fan ventilation** – Fan ventilation relies on electric fans which moves the hot air outside.

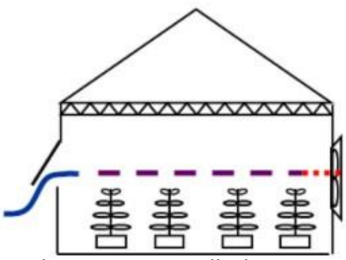

Fig. 1: Fan ventilation

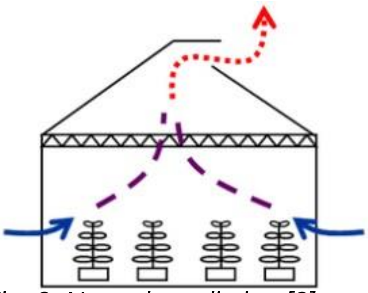

Fig. 2: Natural ventilation [2]

Ventilation Systems	Advantages	Disadvantages
Natural Ventilation	-Suitable for warm and temperature climates -Lower maintenance costs	-Easily affected by outdoor climate -Difficult to design and predict performance
Fan Ventilation	-Suitable for all climates and weather -More controlled and comfortable environment	-Expensive maintenance and installation costs -Equipment generates noise

Table 3: Ventilation systems pros and cons [3]

This project is focused on natural ventilation methods.

Temperature sensors

Temperature sensors are devices which use electrical signals to calculate temperature readings. Temperature detectors are usually built up from two metals that, when exposed to any change in temperature, produce electrical voltage or resistance. Specifically, temperature sensors are electrical apparatus which measures the environment temperature and convert the analogue input data into digital data to control or store information about temperature changes. This type of transducer is essential for maintaining an accurate temperature value inside any manufacturing equipment. There are various types of temperature sensors such as thermocouples, resistance temperature detectors (RTDs), IC sensors and thermistors. Some of these sensors measure indirectly the temperature of a physical object and called non-contact temperature sensors. While others require direct contact with the object and called contact temperature sensors. Usually, non-contact temperature sensors are infrared sensors (IR), which detects infrared radiation from an object and send a signal to an electronic circuit that calculates the temperature of the object. Temperature sensors are often used to control temperature in buildings, refrigerators and to regulate water temperatures. Also, temperature sensors are used in medical and industrial electronics. [4] [5] [6]

Thermocouples

Thermocouples are temperature sensors which measure the difference in temperature between a measuring hot junction and a reference cold junction. Thermocouples are made up of two dissimilar metal wires to generate a voltage proportional to the temperature at their junction (hot junction). A thermal gradient between the hot and cold junctions is generated as the temperature at the hot junction increases, resulting in an electromotive force (emf) measured in millivolts. The temperature difference along the conductor length affects the voltage measured at the conductor ends. The cold junction is typically mentioned at 0 ºC and the millivolt signal is then translated to degrees Celsius or degrees Fahrenheit. Also, thermocouples is a type of contact temperature sensors and come in a variety of shapes and sizes– they can include different combinations of metals to calculate a variety of characteristics and temperature ranges. [6] [7]

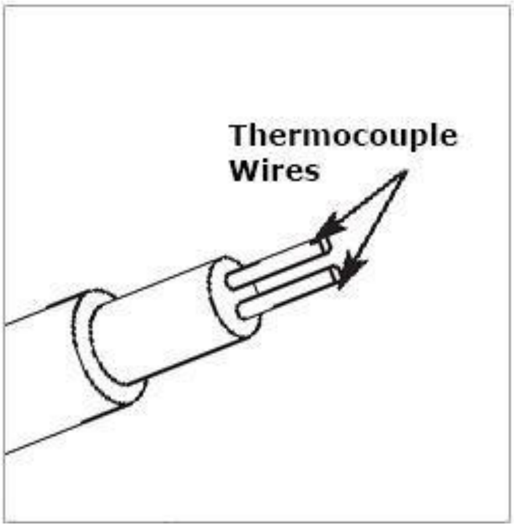

Fig. 3: Thermocouple assembly [6]

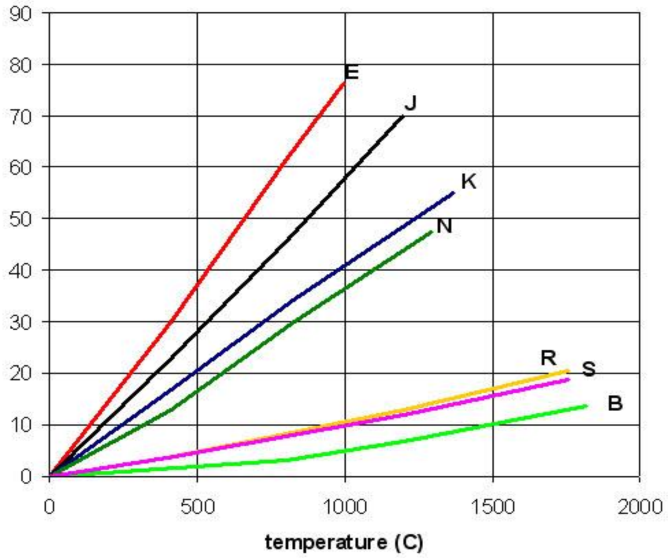

Fig. 4: Voltage vs Temperature relationship for different types of thermocouples [8]

International Type Designation	Conductor material (+Lead/-Lead)	Temperature range (ºC)
R	Platinum 13% Rhodium (+)/ Platinum (-)	0 to +1600
S	Platinum 10% Rhodium (+)/ Platinum (-)	0 to +1500
B	Platinum 30% Rhodium (+)/ Platinum (-)	+100 to +1600
K	Nickel-Chromium (Ni-Cr) (+)/Nickel-Aluminum (Ni-Al) (-)	0 to +1100
J	Iron (Fe) (+)/Constantan (Cu-Ni) (-)	+20 to +700
E	Chromel (Ni-Cr) (+)/ Constantan (Cu-Ni) (-)	0 to +800
N	Nicrosil (+)/ Nisil (-)	0 to +1250

Table 4: Thermocouple types [7]

Resistance temperature detectors (RTDs)

An RTD (Resistance Temperature Detector) is a contact temperature sensor that contains a metal resistor and measures temperature based on the resistance changes on this resistor. The resistance of an RTD varies as the temperature changes. When the temperature of the sensor rises, the resistance starts to increase. Platinum (Pt) is the most widely used element in detectors because it has a very linear resistance-temperature relationship and also this relationship is repeatable over time. The resistance of the sensor is calculated by external electronic apparatus. The resistance of an RTD can be measured by applying a small amount of current (1mA to 5mA) through the RTD and measuring the resulting voltage drop across the resistor. The RTDs are also called platinum resistance thermometers (PRT). The most popular RTDs are PT100 sensors, which have a resistance of 100 ohms at 0°C and they are made using platinum. [9] [10] [11]

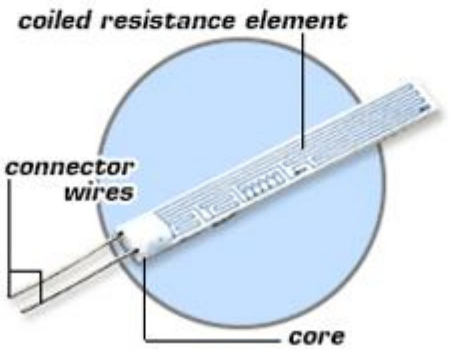

Fig. 5: RTD design [9]

Element Type	Base Resistance in Ohms	Temperature coefficient (Ohm/Ohm/°C)
Platinum	100 Ohms at 0°C	.00385
Nickel	120 Ohms at 0°C	.00672
Copper	10 Ohms at 25°C	.00427

Table 5: RTD Element Types [11]

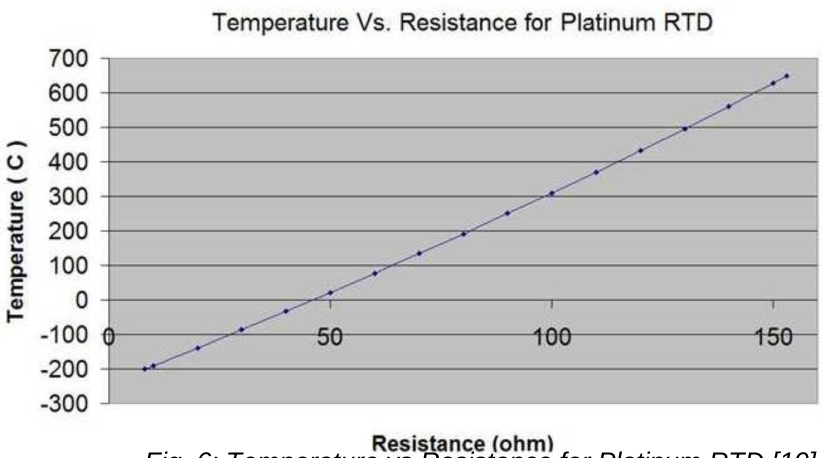

Fig. 6: Temperature vs Resistance for Platinum RTD [10]

Thermistors

A thermistor is a contact temperature sensor which usually includes a ceramic or polymer resistor. A thermistor operates as an RTD, so the resistance is dependent on temperature. Thermistors are non-linear devices, which means that the relationship between resistance and temperature will not be a straight line on a graph. The thermistor's construction defines the location of the non-linear line and how much it changes. Thermistors are classified into two categories: Negative temperature coefficient (NTC) and Positive temperature coefficient (PTC). These two categories differ in the changes of the resistance which are proportional to the temperature. An NTC thermistor decreases its resistance as the temperature increases. A PTC thermistor increases its resistance as the temperature increases. Typically, thermistors are constructed of metallic oxides, that have been pressed into a bead, disk or cylindrical shape before being encased in an impermeable substance like epoxy or glass. The type of the elements used in a thermistor determines how much the resistance changes. [12] [13] [14]

Fig. 7: Thermistor [12]

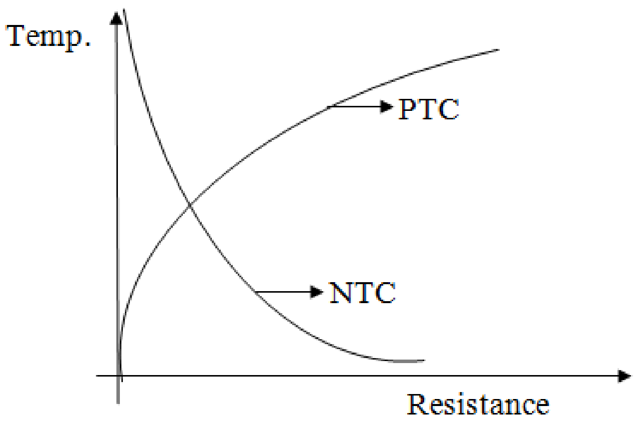

Fig. 8: Temperature vs Resistance for PTC, NTC thermistors [15]

IC Temperature sensors

IC Temperature sensors are commonly a two or three terminal integrated circuits that generate an output (current or voltage) proportional to absolute temperature. Others provide alarm functions by combining analog-sensing circuits with voltage comparators. Other sensor ICs use a combination of digital inputs/outputs with analog circuitry and control registers to be an ideal choice for microprocessor-based systems. These transducers are used to measure a limited temperature range (-55° to 150° C). The IC Temperature sensors are preferred over thermocouples, RTDs and thermistors because they can provide an output without external circuitry. Also, IC sensors are extremely linear and in some cases inexpensive. There are three types of IC sensors: [16] [17]

- Voltage output IC sensors – These sensors have a 0°C offset, allowing them to be used and read at temperatures below 0°C without the use of a negative power supply. Also, across their temperature spectrum, non-linearity is usually less than 1°C.

- Current output IC sensors – They have a nominal output of 298 µA at 25 °C and they have 1 µA output per °C.

- Digital output IC sensors – They have built in A-D converters and the number of digits in these converters displays the resolution.

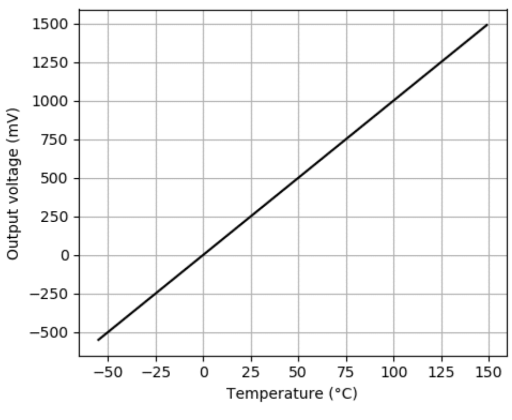

Fig. 9: IC sensor (LM35) output voltage in mV versus temperature [18]

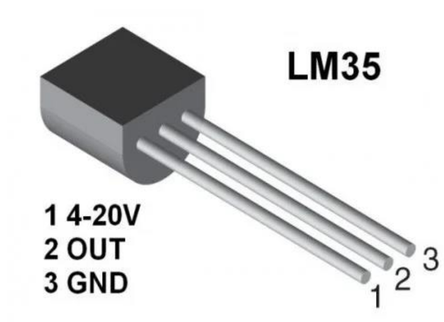

Fig. 10: LM35 – Voltage output IC sensor [19]

Temperature sensors	Temperature range(typical)	Accuracy	Linearity	Power required	Response time	Cost
Thermocouple	-200 to 1750°C	0.5°C to 5°C	Non-linear	Self-powered	0.10 to 10 seconds	Low
RTD	-200 to 600°C	0.1°C to 1°C	Fairly linear	Constant voltage or current	1 to 7 seconds	High
Thermistor	-90 to 325°C	0.1°C to 1.5°C	Exponential	Constant voltage or current	0.12 to 10 seconds	Low to moderate
IC Sensor	-55 to 150°C	0.5°C	Linear	Constant voltage or current	1-2 minutes	Low

Table 6: Comparison of sensor types

Sensor	Advantages	Disadvantages
Thermocouple	Wide Temperature range Self-powered Best for high temperatures	Accuracy Stability
RTD	Accuracy Stability Linearity	Response time Sensitivity Cost
Thermistor	Accuracy Cost Fast	Non-linearity Limited temperature range
IC Sensor	Linearity Cost Highest Output	Slow Temperature range Power supply required

Table 7: Advantages and disadvantages of each sensor [16]

Criteria	Thermocouple	RTD	Thermistor	IC Sensors
	Weight (out of 5)			
Ease of use	2	2	4	4
Temperature range	5	3	1	3
Cost(Min)	2	1	3	4
Accuracy	2	4	5	4
Sensitivity	2	3	5	4
Previous knowledge	2	1	3	4
Long-term stability	2	4	3	3
Total	17	18	24	**26**

Table 8: Temperature sensors

The decision matrix shows that an IC temperature sensor should be used for the project. This is because it provides the highest output among the other sensors and it is the most inexpensive.

Part number	Temperature range	Accuracy (Typical)	Input voltage range	Linear temperature slope
LM35	55 to +150°C	(+/-)0.5°C	4 - 30V	10 mV / °C
AD590	-55 to +150°C	(+/-)0.5°C	4 - 30V	1 µA / °K
DS18B20	-55 to +125°C	(+/-)0.5°C	3 - 5.5V	10 mV / °C
MCP9509	-40 to +125°C	(+/-)1°C	2.7 - 5.5V	10 mV / °C

The LM35 temperature sensor should be used for the project, according to the decision matrix. The LM35 provides a wide temperature, input voltage range with great accuracy and linearity for the project needs.

Temperature Control

Temperature control is a mechanism that detects or measures any change of temperature in the environment and then controls this change of energy of the environment until a stable temperature is achieved.

PID Temperature Controller

PID controller is also known as proportional, integral and derivative temperature controller. This temperature controller is used in applications where stability is important. The PID controller's task is to take an error signal and run it through three different mathematical operations. Specifically, these mathematical operations detect current errors, accumulate past errors, and forecast potential errors based on the current rate of change. Then, the PID controller sums up the results of these three operations and creates an output that will drive the device to the desired setpoint (SV). Nowadays, a huge number of temperature controllers use an auto-tuning process. This process adjusts the controller's output while monitoring the process response. After which, the controller measures an optimal set of PID values to accurately manage the process. [21] [22]

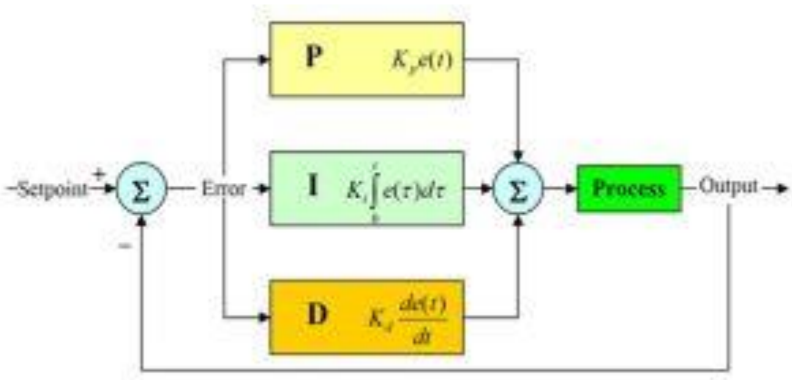

Fig. 11: PID Temperature controller diagram [21]

Hysteresis control

The Hysteresis control is used in applications where stability is not important. It uses a range of temperature values to control its process. For instance, the hysteresis control will respond to maintain the temperature in a room when the present temperature value (PV) is +/- 20°C from SV. Also, the Hysteresis control is often added to a circuit to prevent any unwanted rapid switching. This method is used to equalize for contact bounces in switches or noise in an electrical signal. In this project, a set value of 22°C will be used and a range band of +/- 2°C for hysteresis response. [21] [23]

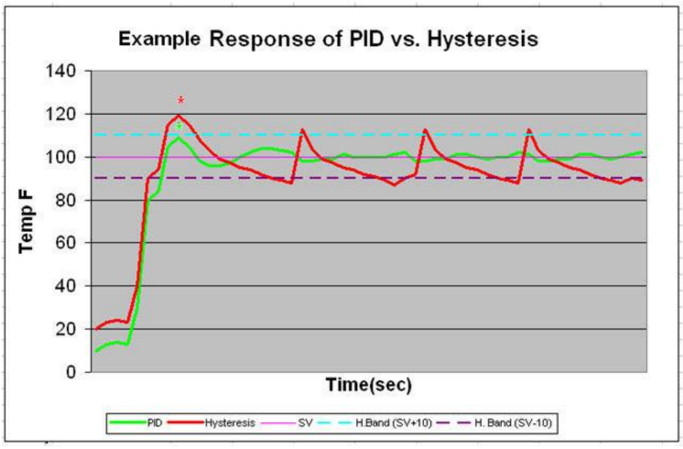

Fig. 12: Example Response of PID vs. Hysteresis [21]

Temperature control methods	Advantages	Disadvantages
PID Control	Good at controlling the temperature to a specific value, SV Overshoot is better than hysteresis when the process started Better for sensitive processes	Oscillating output bad for relay type output PID constants should be tuned accurately PID algorithm have to be accurate
Hysteresis Control	Good at controlling the temperature to a temperature band Better for relay type outputs Neither PID functions nor auto-tuning are needed	Cannot control the temperature to a specific value, SV Overshoot can be larger than PID depending on process response

Table 10: Temperature control methods pros and cons [21]

Microcontrollers

A microcontroller or a microcontroller unit (MCU) is a small computer that performs a single function repeatedly. In physical terms, a microcontroller is a semiconductor IC (integrated circuit) chip that contains several components such as a CPU (Central Processing Unit), ROM (Read-Only Memory), RAM (Random Access Memory), I/O (Input/Output) Ports and other peripherals. [24] [25]

Nowadays, there are several well-known manufacturing brands and programming architectures for microcontrollers. The three main types currently used are: 8-bit microcontrollers, 16-bit microcontrollers and 32-bit microcontrollers. The key difference between these types comes in terms of their bus widths. The bus term refers to the parallel lines that link the microcontroller components, which transfer instructions and data between the CPU, memory and I/O ports. There are three types of buses in a microcontroller: data bus, address bus and control bus. The number of parallel lines, wires or links in a bus relates to the bus width. The overall performance and the precision of the microcontroller improve as the bus width increases. For example, to perform 16-bit or 32-bit calculations on an 8-bit microcontroller, more bus accesses and instructions are needed. As a result, it will provide the answer to a process slower than a 16 or 32-bit MCU. The 8-bit MCUs are the most cost-effective choices but they have a limited range of capabilities in some applications. 16-bit and 32-bit MCUs are normally more costly than 8-bit microcontrollers, but they provide better performance. [24] [25]

(a) *(b)* *(c)*

Fig. 13: Three types of MCUs,

(a) An 8-bit microcontroller, (b) an 16-bit microcontroller and (c) an 32-bit microcontroller [25]

Microcontroller Applications:

- Consumer Electronics Products - Toys, Cameras, Robots, Washing Machine, Microwave Ovens.
- Instrumentation and Process Control - Oscilloscopes, Multi-meter
- Communication - Cell Phones
- Office Equipment - Fax, Printers
- Multimedia Application - Mp3 Player [26]

Despite the fact that there are only three basic types of microcontrollers, there are several MCU manufacturer brands and architectures to choose from. The following are some of the most popular types of generally used MCUs:

- Microchip Technology Atmel AVR microcontrollers
- Microchip Technology PIC microcontrollers
- Intel 8051 microcontrollers
- ARM core processors [25]

Microcontrollers	Cost	Speed	Bus Width	Power Consumption
8051	Very Low	12 Clock/instruction	8-bit	Average
PIC	Moderate	4 Clock/instruction	8/16/32 - bit	Low
AVR	Moderate	1 Clock/instruction	8/32 - bit	Low
ARM	Low	1 Clock/instruction	32/64 - bit	Low

Table 11: Main differences between microcontrollers

Microprocessors

A microprocessor is an electronic component that a microcontroller or a computer uses to perform its functions. These functions are arithmetic and logic operations. A microprocessor, also known as a CPU (Central Processing Unit), is made up of millions of tiny components such as transistors, diodes and resistors that operate together on a single IC (integrated circuit) chip. Microprocessors use three stages called Fetch, Decode and Execute. In the fetch step, commands are transferred from the MCU memory to the microprocessor. Then, in the decode phase, the microprocessor determines what arithmetic or logic function the data will operate. In this phase, the information is interpreted by the ALU (arithmetical and logic unit). After that, this procedure is carried out in the execute phase. Various MCUs may have different series of commands. Microprocessors can be classified as: [27] [28]

- Based on word length – These CPUs are classified according to their word length and they are categorised as 8-bit, 16-bit and 32-bit.

- Reduced Instruction Set Computer (RISC) - Microprocessors with a RISC instruction set are more widely used than those with a more complex set of instructions. Since RISC processors have fewer instructions, their designs are simpler than the other CPUs, which means that they can execute data faster. Also, they have a fixed clock speed.

- Complex Instruction Set Computer (CISC) - CISC processors are the opposite of RISC CPUs. The target of CISC CPUs is to cut down on the number of instructions required for each program. The number of IPC (Instructions per Cycle) is not taken into account. Also, these processors have several clock cycles for each instruction.

- Digital Signal Multiprocessors – These CPUs are used to convert signals from analogue to digital and the opposite. Also, they are used to encode and decode videos. These CPUs have to be capable of performing complex mathematical calculations.

The following terms are commonly used in microprocessors.

- IPC - The IPC refers to Instructions per Cycle. It is a calculation that determines how many instructions a microprocessor can process in a single clock cycle.

- Clock Speed - Clock speed or clock rate stands for the number of operations that a CPU can perform per second. The Clock rate can be expressed in MHz (megahertz) or GHz (gigahertz).

- Bandwidth – Bandwidth (BW) is the number of bits that can be executed in a single instruction.

- Word Length – Word length is the number of bits that a CPU can handle at a time.

- Data Types – Binary, ASCII, signed and unsigned numbers are data types that usually a microprocessor can deal with.

- Bus – Buses are used to send control signals and data between the processor and other components.

- Instruction set – The instruction set is the set of commands that the microprocessor can understand. [29]

Fig. 14: Different types of microcontroller [29]

Criteria	Arduino Uno	8051	PIC16F8X	Cortex-M0
	Weight (out of 5)			
Ease of use	5	2	4	2
Performance	3	3	3	3
Cost(Min)	3	4	5	4
Power Consumption	4	3	4	4
Speed	3	4	3	5
Previous knowledge	5	1	4	1
Pins	4	4	3	3
Total	**27**	21	26	22

Table 12: Decision Matrix for Microcontrollers

The Arduino uno microcontroller was chosen for this project, according to the decision matrix. Especially, the arduino uno Rev3 board is going to be used for the project needs. The Arduino uno is an AVR microcontroller board that uses the ATmega328P, an 8-bit microcontroller with 2KB of RAM and 32KB of flash memory. Also, the arduino uno Rev3 board has 14 input/output pins, 6 analogue pins, a 16 MHz ceramic resonator, a reset button, a power jack and a USB connection.

Electric Motors

Electric motors are devices which transform electrical energy into mechanical energy. This transformation can be achieved by putting electricity into one end of a motor which causes the metal rod to spin around at the other end. Specifically, the current that is inserted into a motor through the wires creates a magnetic field around them. This magnetic field interacts with the motor's magnetic field resulting in the generation of a force in the form of torque. This force is produced by the fundamental law of magnetism: opposite poles attract and like poles repel. Then, this force is applied on the motor's shaft. As a result, the metal rod provides power to drive a system. Electric motors are commonly used in motion control systems and they can be classified as: [30] [31] [32]

- DC motors – These motors are powered by direct current sources such as batteries. DC electric motors are usually categorised into brushless or brushed motors. Brushless motors need less maintenance than brushed motors because the brushes do not need to be replaced. Furthermore, brushless motors normally spin faster than brushed motors.

- AC motors – These motors are powered by alternating current sources such as electrical generators. AC electric motors are categorised into synchronous or asynchronous motors. Regardless of the load, a synchronous motor operates at a constant speed at a fixed frequency. On the other hand, the speed of an asynchronous motor decreases as the load increases.

- Stepper Motors – These motors are brushless DC motors that divide a complete rotation into a series of equal steps. Each of these equal steps causes the shaft of the motor to rotate at a certain angle.

- Servo Motors – These motors are electronic apparatus and linear actuators that precisely rotate and drive parts of a system. Servos are primarily used to monitor angular or linear orientation, velocity and acceleration. [30] [31]

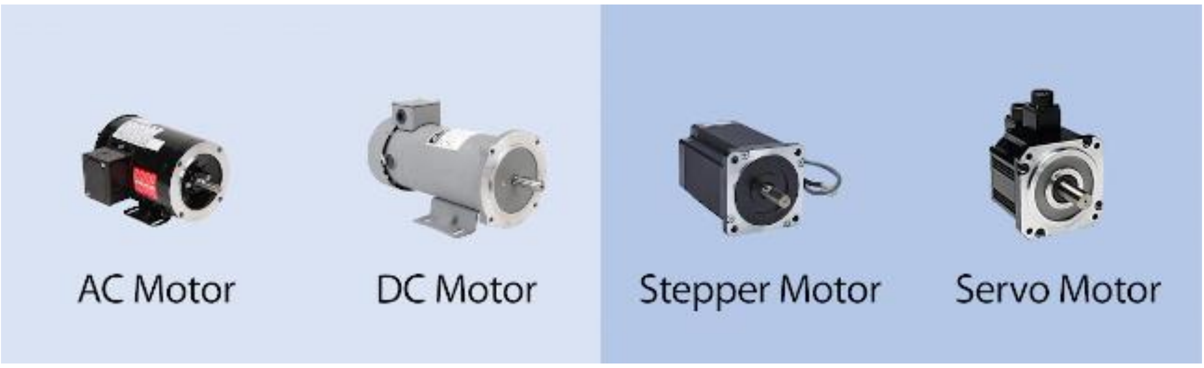

Fig. 15: Various types of electric motors [33]

Parameters	DC Motors	AC Motors	Servo Motors	Stepper Motors
	Weight (out of 5)			
Cost(Min.)	3	3	2	4
Efficiency	4	3	4	3
Speed	5	3	4	3
Accuracy	2	2	5	4
Reliability	3	3	2	5
Previous knowledge	4	2	4	5
Ease of use	4	2	3	5
Total	25	18	24	**29**

Table 13: Decision Matrix for motors

The desicion matrix's results indicate that a stepper motor should be used for the project. The stepper motor can control better the opening angle of a window due to his high accuracy. Also, stepper motors are widely used with arduino microcontrollers, so this type of motors it will preferred other the others.

Stepper Motors

Stepper motors are brushless DC motors that are used in precision applications. Stepper motors are a kind of a digital input-output system and they are designed to complete a full revolution in smaller steps. Specifically, these motors are designed in such a way that the current flowing through them reaches a series of phased coils that can be turned on and off in rapid succession. This process enables the motor to move in fixed steps in a full rotation. As a result, a stepper motor may be used to pass minutely precise motions to mechanical parts that require extreme precision. This attribute makes them differ from a usual DC motor which rotate continuously for a random number of spins until the voltage that is passing through them stops. Furthermore, this type of electric motors are essential components of open loop control systems and they are usually digitally controlled. [34]

A rotor and a stator are the two main components of a stepper motor. The rotor is the component of the motor that spins and generates torque. The stator is the motor's stationary component that houses the rotor. The rotor of a stepper motor is a permanent magnet. When an electrical current is passed through several coils in the stator, they act as electromagnets. As the electromagnetic coil is charged, the rotor will interact with it. The rotor starts to accelerate when the current starts running through the coils. [35]

Two coils are used each time to control the stepper motor. These coils have a centre connection: when the centre connections of those two are joined together, the stepper motor is called a unipolar stepper. If the centre connections are not connected, the motor will behave similarly to a bipolar stepper, with each coil running independently.

A bipolar stepper motor contains a stepper motor driver board that reverses the current flow through the phases. The driver completes this process with the help of an H bridge circuit. All of the coils can be put to work turning the motor by energising the phases while reversing the polarity. A unipolar stepper motor does not need the current flow to be reversed in order to conduct stepping functions. As a result, a unipolar stepper is easier and cheaper to manufacture. However, the bipolar stepper motors due to their complex design are more powerful and efficient to run than the unipolar ones. [34]

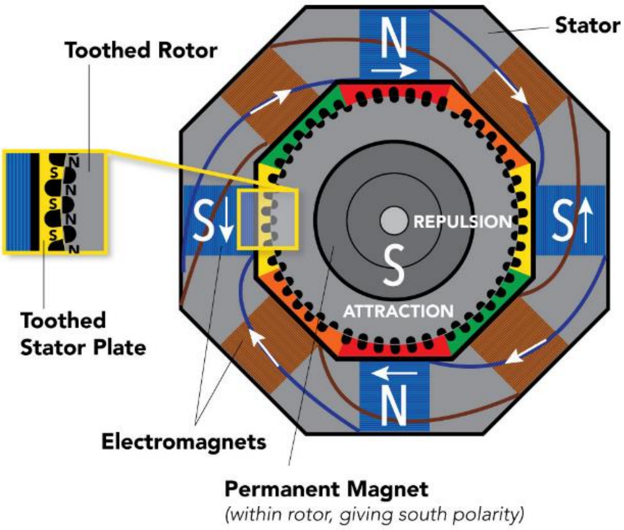

Fig. 16: Diagram of a Stepper Motor [36]

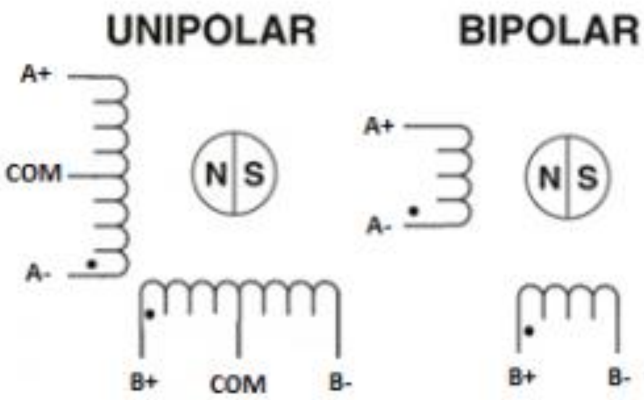

Fig. 17: Unipolar vs Bipolar Configuration [37]

Stepper motor drivers

H-Bridge

An H-Bridge is a four-switch circuit capable of safely driving a DC or stepper motor. These four switches could be transistors or relays. The transistor is a switch that can be turned off by applying a small current signal to it. The H-bridge can control both motor's speed and the direction of rotation. It accomplishes this by opening different transistors to allow the current to move in different directions. As a result, this process changes the polarity of the motor. The switches S1 and S2 or S3 and S4 should never be closed together. This will result in a short circuit and potential microcontroller damage. H-Bridges will prevent the microcontroller from being too hot by the motors it is controlling. The motors store electrical energy in magnetic fields. When the current stops flowing to the motor, magnetic energy reverts to electrical energy, which can cause damage to the components. The H-Bridge improves the isolation of the microcontroller, so the MCUs should never be connected directly to the motors. [35]

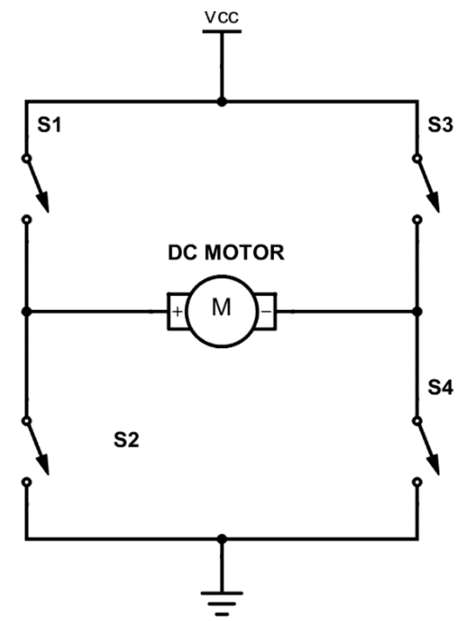

Fig. 18: H-Bridge concept [38]

Darlington arrays

Darlington apparatus are switch arrays that contain multiple open-collector Darlington pairs and can drive a stepper motor. A Darlington transistor (also known as a Darlington pair) is made up of two transistors that can switch much more current than a typical transistor's collector-emitter circuit. The Darlington transistors can switch high current from a parallel-port output, enough to power a mechanical relay or an electrical motor. Instead of using one pair of Darlington transistors, an integrated circuit designed specifically for driving high-current loads from TTL-level inputs can be used. The ULN2003 is the most common IC of this kind. [39]

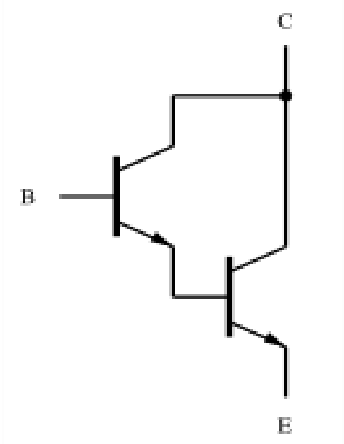

Fig. 19: Circuit diagram of a Darlington pair using NPN transistors [40]

For this project a 28BYJ-48 stepper motor and a ULN2003 motor driver board were used due to limited university lab supplies during the COVID-19 pandemic.

The 28BYJ-48 stepper motor is a 5-wire unipolar stepper motor that operates at 5 volts. These motors have two step angle modes: the half-step mode and the full-step mode. The half-step mode contains an eight step control signal sequence and a stride angle of 5.625°/64, which corresponds to a rotation of 5.625° for each step. Also, it means that the motor has to do 64 steps for one revolution (360°/5.625° = 64). The full-step mode includes a four step control signal sequence and a stride angle of 11.25°/ 32. The full-step mode powers two out of four coils at each step which means that the full step mode gives a higher maximum speed than the half-step mode. The stepper library of the arduino (<Stepper.h>) was used to operate the full step mode. [41] [42]

Technical specifications:

- Rated Voltage: 5V DC
- Speed variation ratio: 1/63.68395≈1/64
- Frequency: 100 Hz
- Friction torque: 600-1200 gf.cm
- Pull-in torque: 300 gf.cm
- Insulated electricity power: 600VAC/1mA/1s
- Weight: 30 g

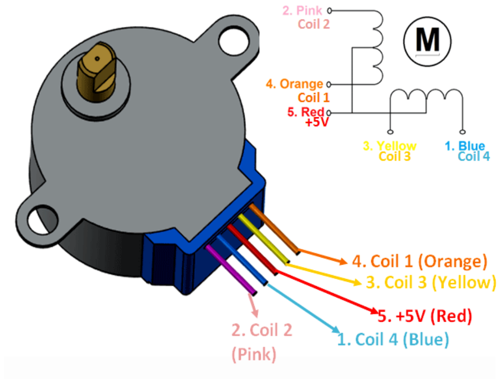

Fig. 20: 28BYJ-48 Stepper Motor Pinout Wiring Diagram [41]

The table below shows the order in which each coil should be triggered in the full-step mode.

Step	A (Coil 1)	B (Coil 2)	C (Coil 3)	D (Coil 4)
1	1	0	0	1
2	1	1	0	0
3	0	1	1	0
4	0	0	1	1

Clockwise (CW)

Counter Clockwise (CCW)

These stepper motors consume a lot of current, so a motor driver IC such as the ULN2003 is required.

The ULN2003 apparatus is a high-voltage, high-current darlington transistor array, that consists of an array of seven NPN Darlington transistor pairs. Each pair can drive loads of up to 50V and 500mA. For higher current capacity, the Darlington pairs can be connected in parallel. To improve their effectiveness, these devices include common-cathode clamp diodes for switching inductive loads. The ULN2003 system has a series base resistor for each darlington pair, allowing direct operation with TTL or CMOS at 5 V or 3.3 V supply voltages. These IC motor drivers allow to drive high current loads such as relays and electric motors that need more power than a microcontroller can provide or sink. [43] [44]

Technical specifications:

- Package: 16 pin DIP
- No. of Darlington Pairs: 7
- Peak Output Current: 500mA
- Maximum Output Voltage: 50V
- Input voltage: 5V
- Maximum Switching voltage: 50V
- Delay time: 250ns
- Input compatibility: CMOS, TTL
- Temperature range: –40°C to 105°C

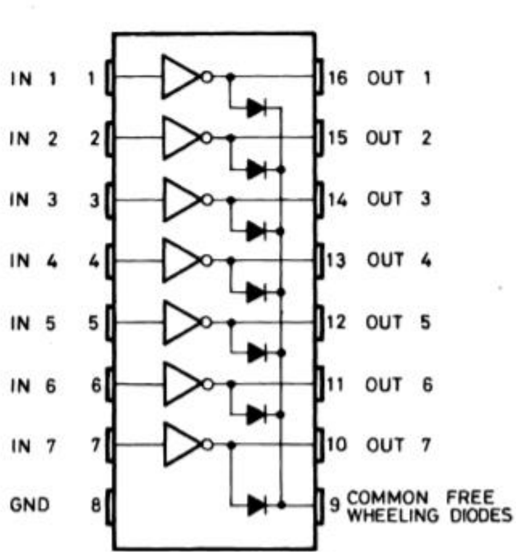

Fig. 21: Pin connections (Top view) [44]

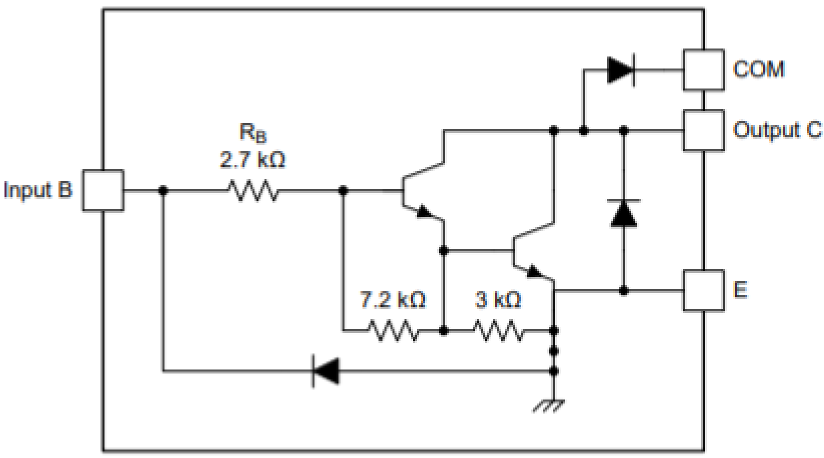

Fig. 22: ULN2003 block diagram [43]

Linear drive method

Linear drive devices or linear actuators are mechanical or electrical apparatus that transform rotational motion in motors into linear or straight push/pull motions. These devices are used in a huge amount of applications where tilting, moving, dragging, or pushing with pounds of force are needed.

Rack and pinion

Rack and pinion is a linear actuator that converts rotational motion into linear motion. It consists of a pinion (a round gear) and a rack (a straight gear) which are connected together. The rotation motion of the pinion is converted to the linear motion of the rack. Typically, the rotation of the pinion is driven by a hand or a motor. [45]

The following formula calculates how far the pinion can travel across the rack:

$C = \pi * D$,

C = Linear distance

D = The pitch diameter of the pinion

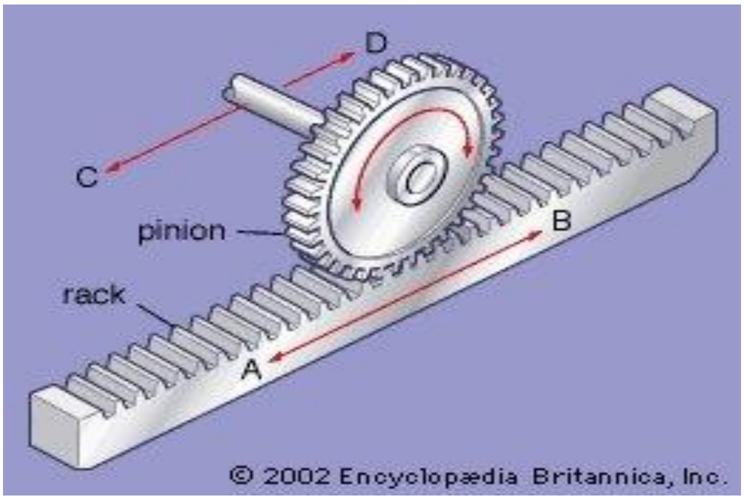

Fig. 23: Rack and Pinion [46]

Worm drive

Worm drive is a linear drive method, consisting of a worm screw and a worm wheel. The worm screw is a screw-shaped gear and the worm wheel is in the form of a pinion. Worm drive converts the rotation motion of the worm screw into torque in the pinion. At high reduction ratios, the worm wheels generate a lot of heat due to friction when sliding. As a result, their efficiency starts to decrease. To reduce friction and efficiency loss, the worm screw and worm gear are constructed of dissimilar metals. [47]

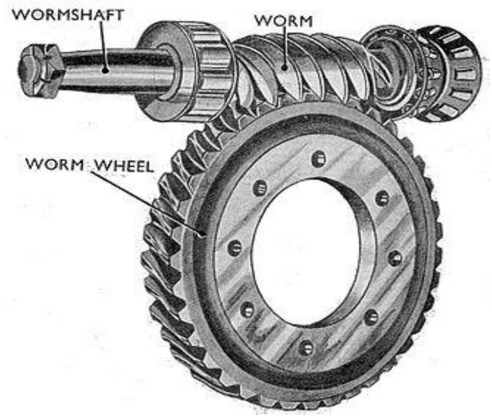

Fig. 24: Worm drive [47]

The rack and pinion actuator will be used to open and close the window. The rack and pinion actuator is more efficient than the worm drive in window applications because the friction that is generated when sliding in the worm drive is higher. Also, the rack and pinion can handle larger forces than the worm drive. Moreover, the worm drives produce a lot of noise when operating at high speeds and are difficult to design due to the complex shape of the worm screw.

Greenhouse Window

There are a lot of types of windows that can be used in a greenhouse such as single hung, double hung, arched, awning or casement windows. The awning windows are one of the best choices to filter light and air inside of a greenhouse. They provide adequate ventilation during the warm seasons and waterproof security during rainy days. They are also energy-efficient and simple to install.

Nowadays, awning windows vary in size, depending on the use. On average, the width of awning windows varies from 5 centimeters to 110 centimeters. The height ranges from 50 centimeters to 230 centimeters. A greenhouse is required to use multiple awning windows side by side to control the temperature inside of it. [48]

The task of measuring the size (width, height, depth) of an awning window is simple and needs only a tape measure.

For awning windows, the flow can enter the greenhouse via two paths as shown in the green areas A1 and A2 in Fig. 25.

The ventilation rate due to area A1 can be calculated as:

$$Q_1 = C_{d,rec} w_1 \sqrt{C_p \frac{\int_{z_{0,1}}^{h_1+h_0} \sqrt{z^{2/7} - z_0{}^{2/7}}\, dz}{z_{ref}{}^{1/7}}}\, U_{ref}$$

Where $h_1 = h(1 - \cos a)$.

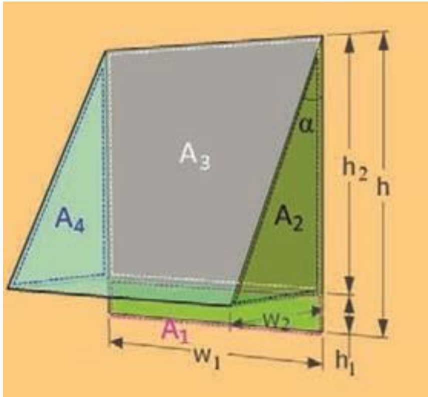

Fig. 25: Awning window dimensions and opening areas

For the ventilation rate via A2, it can be calculated as:

$$Q_2 = \frac{(1 - 0.5|\cos\theta_w|)}{2} \frac{w_1}{w_1 + \frac{w_2}{2}} C_{d,rec} w_2 \sqrt{C_p \frac{\int_{z_{0,2}}^{h_2+h_0} \sqrt{z^{2/7} - z_0{}^{2/7}}\, dz}{z_{ref}{}^{1/7}}}\, U_{ref}$$

Where:

Cd = Discharge coefficient

Cp = Pressure Coefficient

h = Opening height

U = Wind speed

w = Opening width

Z = position (Vertical direction)

z0 = Z position of the neutral plane

α = Window opening angle

θw = Wind incident angle

A = Area

P = Pressure

Q = Ventilation rate

Fig. 26: Awning window [50]

In the first part of the equation, the “2” in the denominator represents the triangular region. When the wind is normal to A2, parallel to the opening, the nominator shows that the ventilation rate across area A2 is the highest. When the wind is parallel to A2, this term is not zero since the eddy penetration effect still exists. The second term shows that only a portion of the outside air would reach the greenhouse via Area A2, while the rest would exit via Area A4. The assuming ventilation ratio that could enter the room is proportional to the area ratio of A3 and A3+A4, which is w1/(w1+w2/2). The sum of these two equations gives the total ventilation rate through the awning window. [49]

Because of the pandemic situation, an awning window was difficult to find and tested, so a hypothetical awning window with an aluminum alloy frame and size (80cm x 50 cm) is going to be “used”. A recommended opening awning window angle for all the seasons is 45°, but most of them can open till 90° for cleaning purposes.

Heat losses

The heat loss of a greenhouse can be calculated by multiplying the surface area, the difference in air temperature inside - outside and the heat transfer coefficient. This equation can used to measure the size of the heating system needed during the day.

$Q = A \times U \times \Delta T,$

Where:

Q = Heat loss (Btu/hr or British thermal unit per hour)

A = Surface area of the greenhouse (sq. ft.)

U = U-valve - Overall heat transfer coefficient of the greenhouse (Btu/hr – sq ft-°F)

ΔT = Temperature difference between the inside and outside air (°F)

The surface area is equal to the distance between the top of the greenhouse multiplied by the length of the greenhouse plus the area of the endwalls.

The combined roof vent area should equal the combined sidewall vent area, each of which should be at least 15 - 20% of the floor area, according to the American Society of Agricultural and Biological Engineers guidelines. In northern climates, 15% may be sufficient, but in warmer climates higher amounts are needed. Larger roof vents are needed in large greenhouses, where the sidewall vent area per floor area decreases as the width increases. Where more than 50% of the roof is open, sidewall/endwall vents may be unnecessary. [51] [52]

Fig. 27: Greenhouse Dimensions [53]

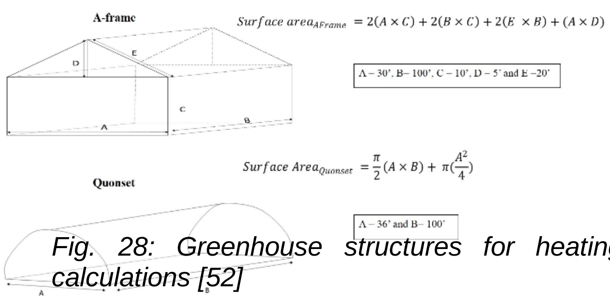

Fig. 28: Greenhouse structures for heating calculations [52]

3D Printing

3D printing, also known as additive manufacturing, is the process of creating a three-dimensional object from a CAD model or a digital 3D model. "3D printing" may refer to several processes in which materials such as plastics, metals, cement, and liquids are deposited, joined, or solidified under computer control to create a three-dimensional object.

CAD models used for 3d printing usually are saved as .STL files (stereolithography file format) and then sent for 3d printing. This type of file format stores data based on triangulations of the surface of CAD models.

Laser Cutting

Laser cutting is a method of cutting various materials with a laser for both industrial and artistic purposes. Laser cutting uses a high-powered laser that is guided by optics and CNC to direct the beam or the material. A motion control device is typically used to obey a CNC or G-code of the design to be cut onto the material. The focused laser beam burns, melts, vaporises or is blown away by a jet of gas to leave a high-quality surface finished edge. [54]

Both of these techniques were used to create a demonstration model.

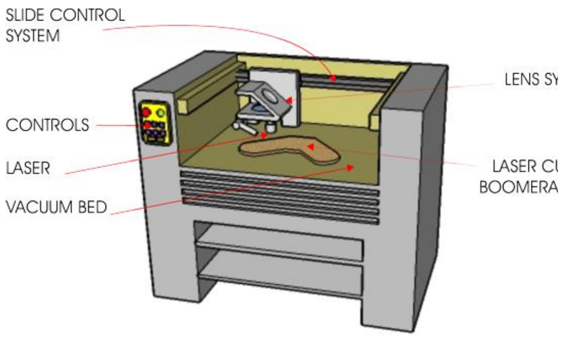

Fig. 29: Laser Cutter [55]

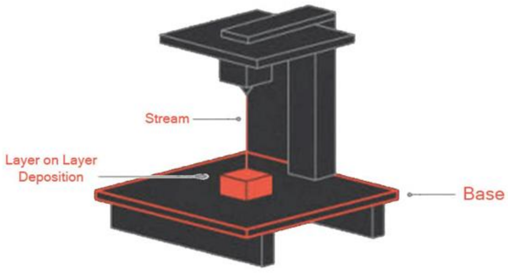

Fig. 30: Working of 3D Printer [56]

4. Methods

<u>Hardware Setup</u>

Temperature sensor circuit

The LM35 temperature sensor circuit was constructed in a breadboard. The LM35 was connected as the basic centigrade option, which means that the negative temperature values will not be calculated. The LM35 package that was used for this project is the 3-pin TO-92 (4.30 mm × 4.30 mm).

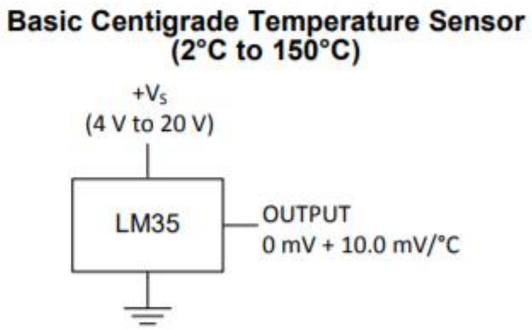

Fig. 31: Basic Centigrade Temperature Sensor [57]

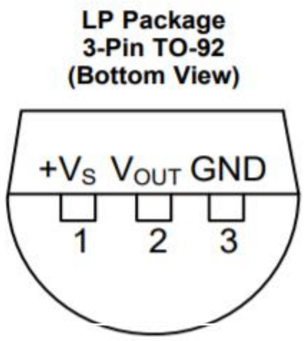

Fig. 32: LM35 Package 3-Pin TO-92 [57]

The LM35 output (Vout) was connected to A0 pin of the arduino uno Rev3 (orange wire). The ground (GND) was connected to the GND pin of the arduino (black wire) and the voltage (+Vs) was supplied from the 5V pin of the arduino (blue wire). An arduino code was used to test the LM35 functionality. This code is included in the appendices.

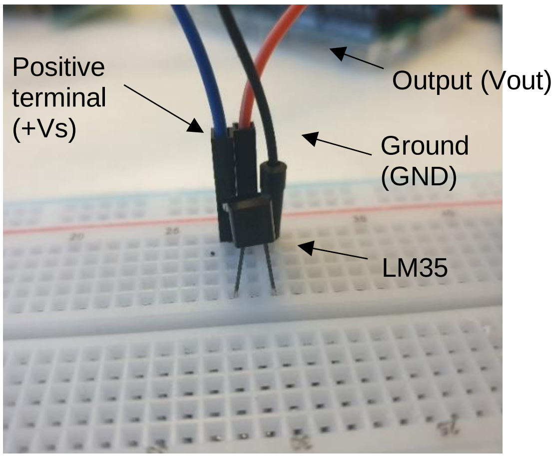

Fig. 33: Temperature sensor circuit

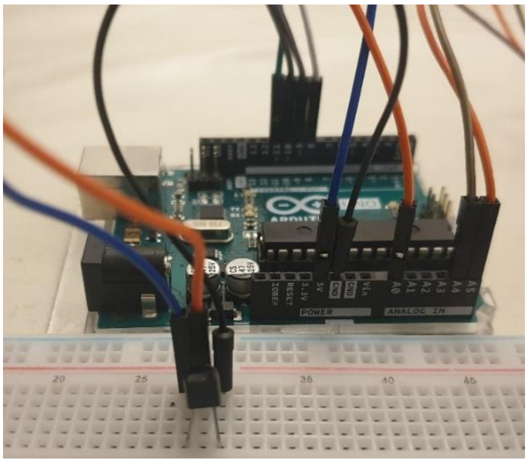

Fig. 34: Temperature sensor connections

Stepper motor circuit

The stepper motor circuit was made up of a stepper motor driver board (ULN2003) and a stepper motor (28BYJ-48).

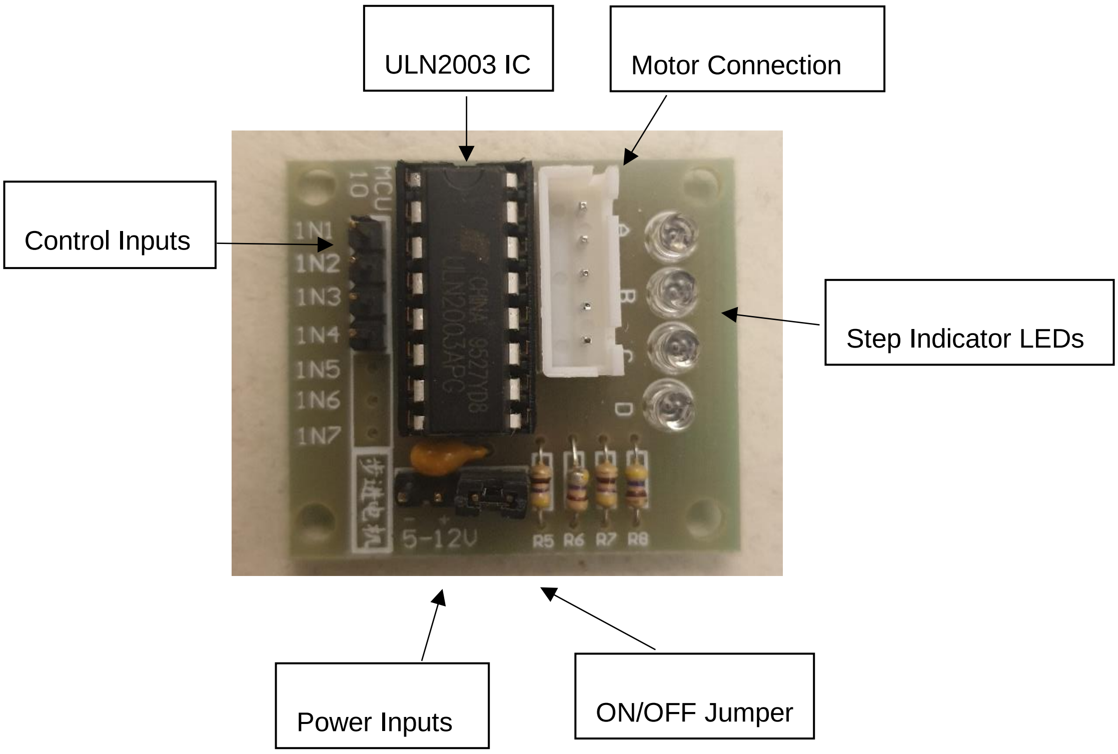

Fig.35: ULN2003 Circuit board

The stepper motor cable was hooked onto the driver board

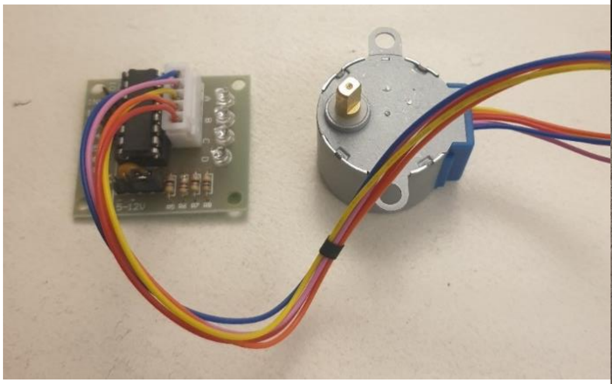

Fig.36: Motor connection with the motor driver

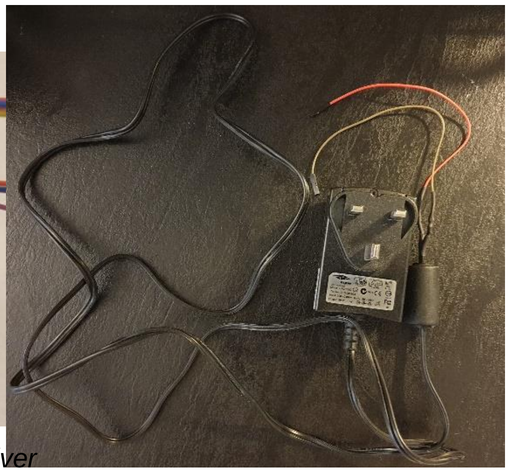

Fig.37: 5V Power supply

An external power supply (5V) was used to power the stepper motor circuit. Also, the IN1, IN2, IN3, IN4 pins of the driver board (ULN2003) were connected to the Arduino digital pins 8, 9, 10, and 11 respectively.

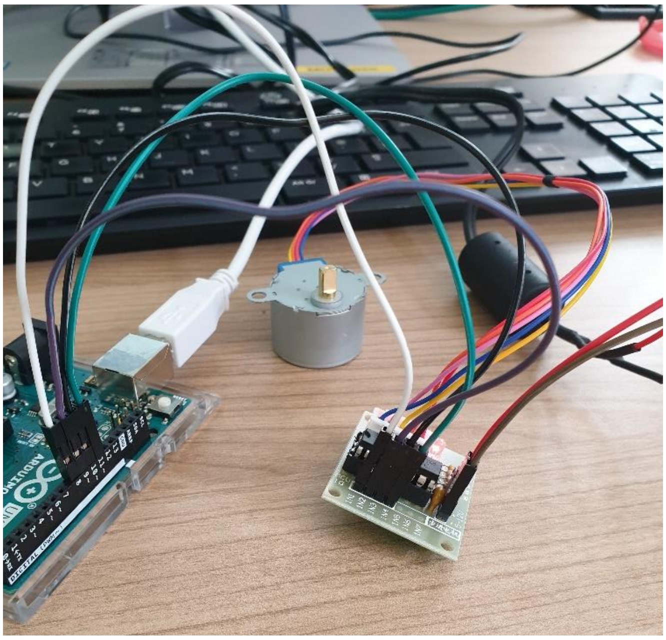

Fig.38: Stepper motor circuit

Control Inputs to Arduino

i) White (IN1) to 8

ii) Purple (IN2) to 9

iii) Black (IN3) to 10

iv) Green (IN4) to 11

LCD Display Circuit

The LCD display circuit was used to display the temperature of the greenhouse digitally. The LCD display was connected with the arduino by using cables supplied by the university's lab.

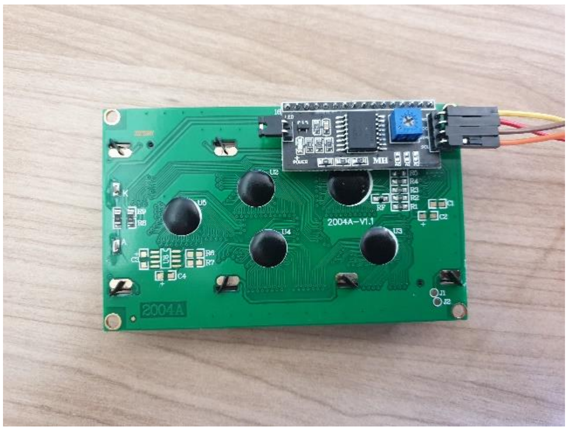

Fig.39: I2C Liquid Crystal Display

Display(Wires)	to	Arduino
i) Yellow(Gnd)	to	Gnd
ii) Red (+5V)	to	Vcc
iii) Brown (SDA)	to	A4
iv) Orange(SCL)	to	A5

All the components were supplied by the university's laboratory. The hardware setup of the project was designed on a web-based EDA tool suite (EasyEDA). The circuit drawing and a part of the physical construction of the circuit are included in the appendices.

Software Setup

Arduino code

The code of the system was written in the C++ programming language. Also, the code was separated into six parts:

- Block comment – This part of the code gives some information about the author and the resources used. The arduino does not execute this part.

- Libraries inclusion – This part contains outside libraries which provide extra functionality in the system. The libraries that were selected for the system are: <Wire.h>, <LiquidCrystal_I2C.h> and <Stepper.h>.

- Constant definition – This section defines different constants in the system such as the number of steps in one revolution of the motor.

- Global variables declaration – This piece of the code declares variables with global scope.

- Setup code – The setup code operates only once, after each start or reset of the arduino board. Typically, it is used to determine variables and pin modes.

- Loop code – The loop code is the main body of the code and runs continuously.

Code explanation

The code starts with the block comment part.

```
XE636_code
// Author: Vasileios Fermelis
// Get the LCD I2C Library here:
// https://bitbucket.org/fmalpartida/new-liquidcrystal/downloads
```

Then, the libraries inclusion sector is added.

```
#include <Wire.h>  // Comes with Arduino IDE
#include <LiquidCrystal_I2C.h>
#include <Stepper.h>
```

The <Wire.h> library was used to allow the communication between the arduino board and other devices. The <LiquidCrystal_I2C.h> and <Stepper.h> libraries were added to control the LCD display and the stepper motor respectively.

After that, the constant definition and the global variables declaration parts are designed.

```
#define STEPS 2038 // the number of steps in one revolution of your motor (28BYJ-48)

Stepper stepper(STEPS, 8, 10, 9, 11);
float Vout;
float Temp;
// set the LCD address to 0x27 for a 20 chars 4 line display
// Set the pins on the I2C chip used for LCD connections:
//                    addr, en,rw,rs,d4,d5,d6,d7,bl,blpol
LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE);  // Set the LCD I2C address
```

Firstly, a constant 'STEPS 2038' was defined, which represents the number of 'steps' required for the motor to complete one revolution. In this case 2038 steps (32 x 63.68395 = 2037.8864 ≈ 2038).

The 28BYJ-48 unipolar stepper motor has four control inputs (IN1-IN3-IN2-IN4), so a new stepper library called 'Stepper' was made to connect these control pins to the arduino with the pin sequence of 8, 10, 9, 11 (Arduino pins).

Then, two variables called ‘Vout’ and ‘Temp’ were made for the LM35 output and the greenhouse temperature respectively.

Also, the LCD I2C address was set to display words or numbers in the LCD display of the project. I2C is a serial communication protocol.

The setup code:

```
void setup()
{ Serial.begin(9600);
  lcd.begin (20,4);
 }
```

A function called ‘Serial.begin(9600)’ was used to set the data rate in bits per second (baud) for serial data transmission. For this case, 9600 bits per second.

The ‘lcd.begin(20.4)’ function was added to set up the LCD’s number of columns and rows.

The loop code:

```
void loop()
 {
  Vout=analogRead(A0) ;
  Temp= (Vout*500)/1023 ;
  lcd.clear();
  lcd.setCursor (0,0); //Start writing on 0.0 on lcd screen
  lcd.print("Temperature"); // In Degree Celsius
  lcd.setCursor (0,1);
  lcd.print(Temp); //Temperature value
  lcd.print ((" C")); //Celsius of course :D
  delay(1000); //Refresh every 1s

 if (Temp > 24) {
  stepper.setSpeed(16); // 16 rpm
  stepper.step(2038); // do 2038 steps -- corresponds to one revolution in one minute
  delay(1000);
}
else if (Temp < 20) {
  stepper.setSpeed(16); // 16 rpm
  stepper.step(-2038); // do 2038 steps in the other direction with faster speed -- corresponds to one revolution in 10 seconds
   delay(1000);
}
else { // 20 < Temp < 24
  // Safe! Continue usual tasks.
}
 }
```

The first function of the loop code was added to read the values of the analogue input of the arduino (A0 pin). This input is connected to the output of the LM35 temperature sensor.

Then, the ‘temp’ variable was used to convert the output voltage of the sensor into temperature values in degree Celsius. The LM35 has a scale factor of 0.01V per degree Celcius, so 1V will give 100 °C. The arduino uno board contains a 10-bit analog to digital converter (ADC), which converts input voltages between 0 and the operating voltage 5V to integer values between 0 and 1023. The ‘temp’ variable will calculate the temperature by multiplying the reading of the analog pin with the operating voltage (5V) and the scale factor of the lm35. Then, it will divide these values by 1023, because the values that will enter the arduino will be in integer form between 1 and 1023 and not 0 to 1023.

After that, different lcd functions were used to print the temperature values of the system on the lcd display.

Next, the if_else function was used to apply hysteresis control. This function sets two thresholds on the system and the system is triggered when the environment’s temperature exceeds these thresholds. For instance, if the greenhouse temperature exceeds the maximum threshold of the system (24°C), the Arduino will move the motor forwards to open the window. Furthermore, two functions called ‘stepper.setSpeed()’ and ‘stepper.step()’ were inserted into the if_else function to determine the speed of the motor in rpm and the direction of the shaft of the motor (clockwise or anti-clockwise) respectively.

Also, delay functions were used in the whole process to pause the system for one second before it starts to operate the next command.

Solidworks

The linear actuator of the system was designed in Solidworks software from scratch. Two models of rack and pinion were built, one was a prototype, the other the original.

Prototype model

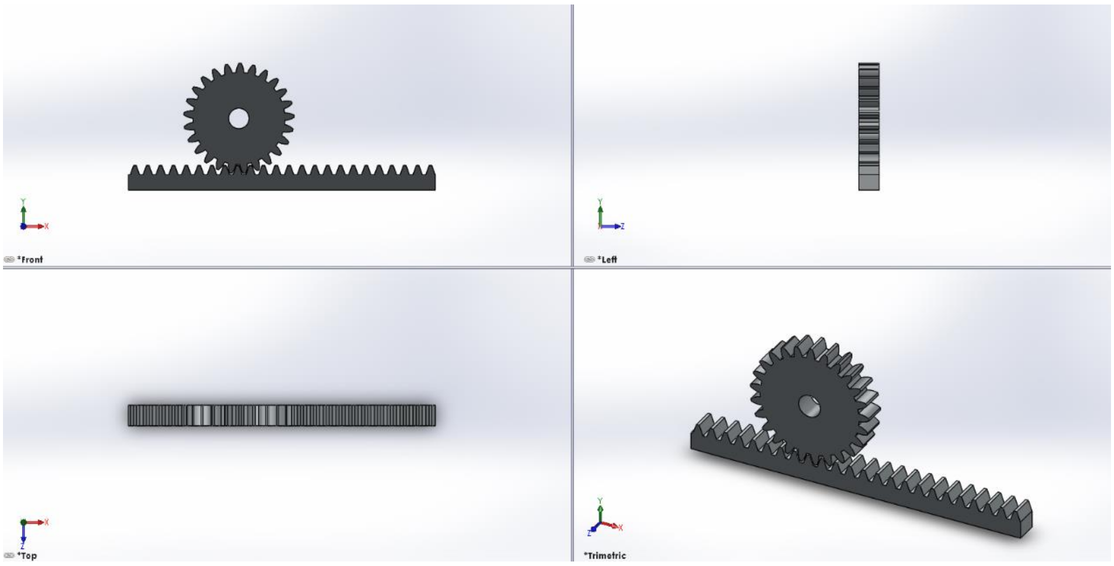

Fig. 40: 3D design of the prototype model

Pinion dimensions:	Rack dimensions:
Number of teeth = 25	Number of teeth = 25
Face width = 10mm	Face width = 10mm
Pitch diameter = 50mm	Pitch height = 10mm
Nominal shaft diameter = 10mm	Length = 150mm

Table 14: Design specifications of the prototype model

The prototype model was constructed to test the functionality of the rack and pinion. The mathematical formulae that was used to calculate how far the pinion will travel across the rack are also referred in the background research.

C = π * D,

$C = \pi \times 50mm = 157.0796\text{mm}$, The result shows that the pinion will move 157.0796mm for one revolution across the rack. Typically, a suitable opening angle for an awning window is 45° which is around 40 to 60cm opening width, so a larger rack is required to open the window. Furthermore, the stepper motor that was used in the system has a nominal shaft diameter of 4mm, so a new pinion design is also required to be suitable for the stepper motor.

Original model

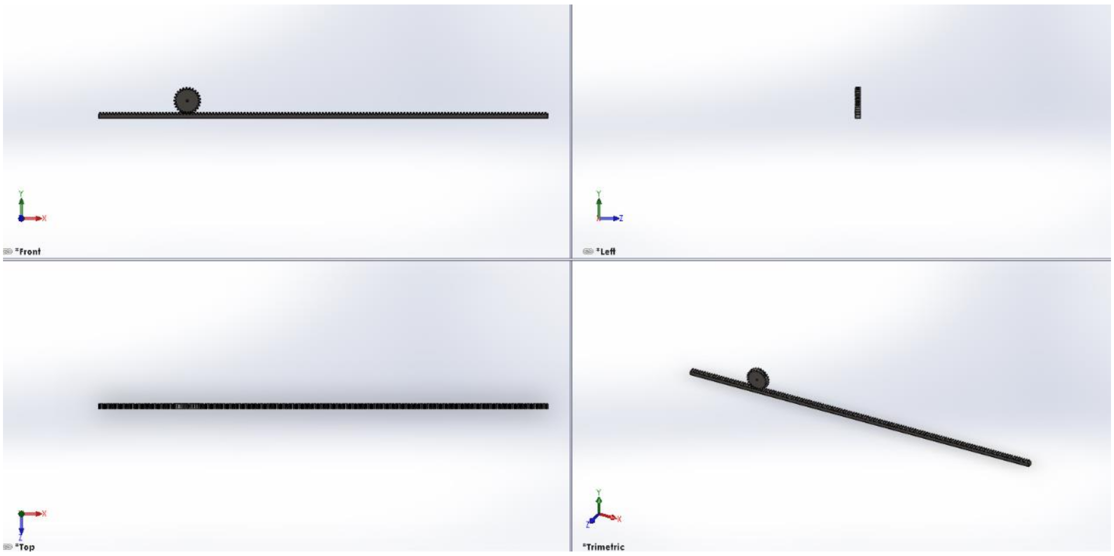

Fig. 41: 3D design of the original model

Pinion dimensions:	Rack dimensions:
Number of teeth = 25	Number of teeth = 143
Face width = 10mm	Face width = 10mm
Pitch diameter = 50mm	Pitch height = 10mm
Nominal shaft diameter = 4mm	Length = 900mm

Table 15: Design specifications of the original model

$C = \pi \times 50mm = 157.0796\text{mm}$

40cm → 400mm

$\mathbf{400mm \div 157.0796mm = 2.54\, revolutions}$

$\mathbf{900mm \div 157.0796mm = 5.73\, revolutions}$

This model was built to fit with the shaft diameter of the 28BYJ-48 (4mm) and the height requirements of the window. The results demonstrate the revolutions that the pinion would need to rotate across the rack.

3D print & Laser cutting

The Solidworks files of the rack and pinion models were converted into .STL files and then were sent to the CEM three dimensional print service for 3d printing. Due to the limitations of the dimensions of the 3D printers the rack part of the actual design was not printed.

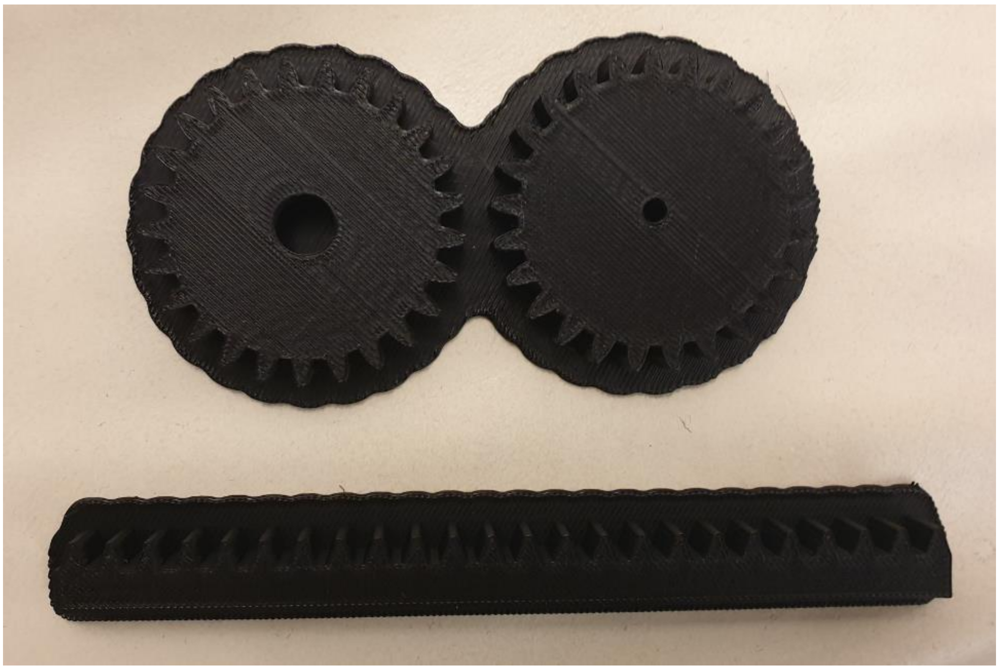

Fig. 42: 3D printed parts of rack and pinion

After that, the same solidworks files were converted into .dxf files and were sent to the CEM laser-cut office for laser cutting. The rack and pinion were made from MDF and they have a width of 9mm.

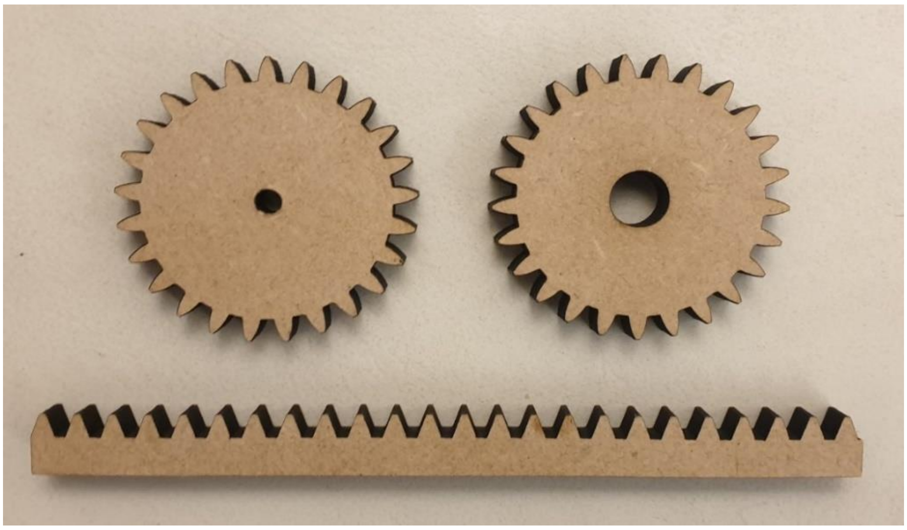

Fig. 43: Laser cut parts of rack and pinion

Awning Window

To calculate the height that the awning window will open, the arc length formulae was used:

$$Opening\ width = 2\pi r \times (\frac{\theta}{360})$$

Where:

r = The height of the awning window (50cm)

θ = Window opening angle (45°)

$$Opening\ width = 2\pi \times 50 \times (\frac{45}{360}) = 39.27 \approx 39cm$$

The opening width should be 39cm to reach the desired opening angle which is 45°. As a result, the pinion would need to rotate for 2.5 revolutions across the rack to open the window.

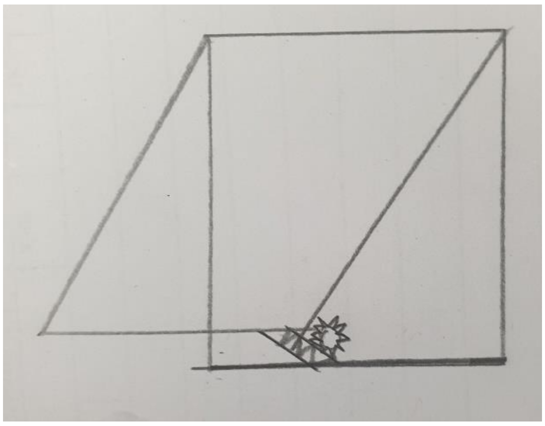

Fig. 44: Awning window drawing

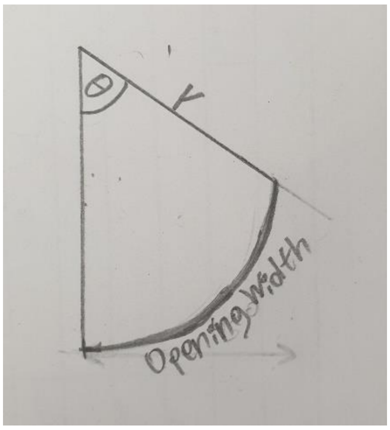

Fig. 45: Opening width drawing

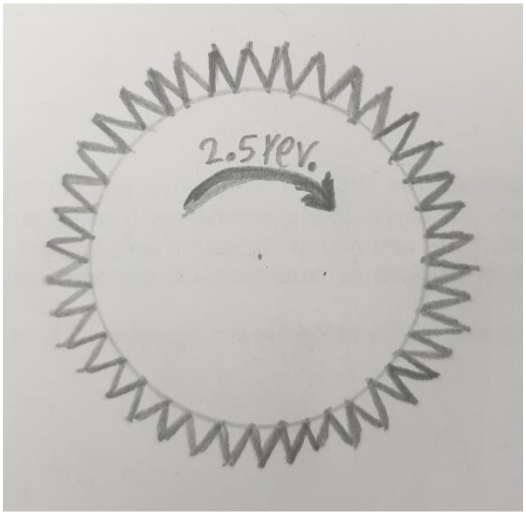

Fig. 46: Pinion drawing

5. Discussion

Overall, the project performed effectively: the desired linear motion to open and close the window was performed and an appropriate temperature detection circuit was built.

The elements used for the linear actuator of the system were selected for their long lifespan and ability to be recycled:

- Medium-density fibreboard (MDF) was the product that was selected for laser cutting. MDF is a sustainably wood-based material that is manufactured from real wood and recyclable materials. Thus, Mdf is referred to, most of the time, as an eco-friendly product. New types of MDF that will be made of fibres will be fully recycled in the next few years.

Within this project, there were both advantages and disadvantages. The main advantage was that the constructed circuit board of the system operated well, with a reasonable efficiency even under different temperature conditions. A disadvantage was the inability of the system to calculate losses that may occur when the window starts to open, which means the use of a mechanism that will stop the motor at the desired window opening angle is necessary.

Previous applications suggest that it is better to use smaller scale greenhouses, instead of large scale ones, because they do not require adjustments to factors restricting airflow and would, instead, focus solely on the window requirements.

If this project were to be adjusted into a larger scale greenhouse, the following factors would have to be considered:

- Which option would be more cost-effective and efficient: larger windows with stronger motors to open them, or smaller windows with weaker motors, like the ones currently in use, which will require more energy and materials.
- Which type of linear actuator would be more efficient to open large sized windows.

- What type of ventilation system would be more practical for those greenhouses (natural or fan ventilation).
- Other temperature or motor control techniques.

Furthermore, previous studies suggest using limit switches in the system in order to stop, start, accelerate or slow down the motor. A limit switch is an electromechanical device that is activated by a physical force applied to it by an object.

Fundamental research on the opening angle of an awning window has been done. In the future, further studies should seek to determine the best options for the opening angle of the window.

It can be said that the objectives set out at the start of the project have been successfully achieved, for the most part, despite the university's closure due to the pandemic.

6. Conclusion

The results suggest that this system can effectively control the temperature in a greenhouse with the use of a microprocessor and an electric motor to open and close the window. All parts of the system contribute to the maintenance of a steady temperature in the greenhouse. The steps taken in this project demonstrate the detailed construction method of a mechanism that can be used to assist the daily functioning and ventilation of greenhouses. This can be adjusted or updated in future studies according to each greenhouse's needs. The system can be further improved by adding adjusting mechanisms and additional features.

7. Future work

There are several suggestions for further development of this system. One of them is about using additional arduino libraries to further support the acceleration of the motor and the heat generation when it is moving. Another recommendation is about the use of the microprocessor (Arduino Uno Rev3), which was selected for the project and had a range of pins available that can be used to extend the capabilities of the system. For instance:

- Temperature and motor control shields could be plugged on top of the Arduino to provide better acceleration or control to the motor and to detect and regulate the greenhouse's temperature more accurately.
- Heat shields should be also used to protect the Arduino board from the extreme heat that will probably generate inside the greenhouse.
- Bluetooth or Wi-Fi shields could be inserted in the Arduino to regulate system's temperature or the motor's speed from the mobile phone.

Further investigation into various types of greenhouses windows and their opening angle adjustments is also suggested. Finally, the system can be tested into a physical awning window and any possible losses from the feedback forces of the window to the system can be calculated. That could be mainly applied in the linear actuator (rack and pinion) of the system, which receives most of the forces.

8. References

1. S.r.l. TP. (2017). The importance of ventilation in greenhouse: the importance of air circulation in hot climate. [online] Available from: https://www.pericoli.com/EN/news/92/The-importance-of-ventilation-in-greenhouse-the-importance-of-air-circulation-in-hot-climate [Accessed 3 May 2021].
2. Hortidaily.com. (2020). Natural or mechanical ventilation. [online] Available from: https://www.hortidaily.com/article/9243323/natural-or-mechanical-ventilation/ [Accessed 3 May 2021].
3. VALAN A S. (2020). Ministry of Health and Family Welfare Government of India, National Guidelines for Infection Prevention and Control in Healthcare Facilities - Research gate. [online] Available from: https://www.researchgate.net/figure/Advantages-and-disadvantages-of-different-ventilation-systems_tbl1_338825913/actions#reference [Accessed 3 May 2021].
4. Jost D. (2019). What is a temperature sensor. [online] Available from: https://www.fierceelectronics.com/sensors/what-a-temperature sensor#:~:text=A%20temperature%20sensor%20is%20an,monitor%2C%20or%20signal%20temperature%20changes.&text=Non%2Dcontact%20temperature%20sensors%20are%20usually%20infrared%20(IR)%20sensors [Accessed 3 May 2021].
5. Pyrosales. (2020). How do temperature sensors work. [online] Available from: https://www.pyrosales.com.au/blog/news/how-do-temperature-sensors-work [Accessed 3 May 2021].
6. Engineering O. (2020). What is a thermocouple and how does it work. [online] Available from: https://www.omega.co.uk/prodinfo/thermocouples.html [Accessed 3 May 2021].
7. TC Ltd. (2020). Thermocouple sensors - Reference information. [online] Available from: https://www.tc.co.uk/thermocouple-information/thermocouple-data.html#thermocouple-theory. [Accessed 3 May 2021].
8. Sterling sensors Ltd. (2019). Thermocouple Technical Reference Information. [online] Available from: https://www.sterlingsensors.co.uk/thermocouples. [Accessed 3 May 2021].
9. Engineering O. (2020). Pt100 temperature sensor. [online] Available from: https://www.omega.co.uk/prodinfo/rtd.html [Accessed 3 May 2021].

10. Sterling sensors Ltd. (2017). RTD Sensors including Pt100 and Pt1000 Elements (PRT). [online] Available from: https://www.sterlingsensors.co.uk/rtds [Accessed 3 May 2021].

11. TE Connectivity. (2020). Understanding RTDs. [online] Available from: https://www.te.com/global-en/industries/sensor-solutions/insights/understanding-rtds.html. [Accessed 3 May 2021].
12. Engineering O. (2020). What is a thermistor. [online] Available from: https://www.omega.co.uk/prodinfo/thermistor.html. [Accessed 3 May 2021].
13. En.wikipedia.org. (2021). Thermistor. [online] Available from: https://www.omega.co.uk/prodinfo/thermistor.html. [Accessed 3 May 2021].
14. Wavelength Electronics. (2020). What is a thermistor. [online] Available from: https://www.teamwavelength.com/thermistor-basics/#:~:text=A%20thermistor%20is%20a%20resistance,such%20as%20epoxy%20or%20glass [Accessed 3 May 2021].
15. Kufre Jack, Israel Etu et al. (2016). A Simple Thermistor Design for Industrial Temperature Measurement - Research gate [online] Available from: https://www.researchgate.net/publication/309267235_A_Simple_Thermistor_Design_for Industrial_Temperature_Measurement [Accessed 3 May 2021].
16. Engineering O. (2020). IC Sensors. [online] Available from: https://www.omega.co.uk/prodinfo/Integrated-Circuit-Sensors.html#:~:text=An%20IC%20Temperature%20Sensor%20is,58%20to%20302%C2%B0F) [Accessed 3 May 2021].
17. Elprocus. (2020) Temperature Sensors – Types, Working & Operation. [online] Available from: https://www.elprocus.com/temperature-sensors-types-working-operation/ [Accessed 3 May 2021].
18. Bakker Bd. (2021). LM35 analog temperature sensor with Arduino tutorial - Makerguides.com. [online] Available from: https://www.makerguides.com/lm35-arduino-tutorial/ [Accessed 3 May 2021].
19. Salab. (2018). Arduino Temperature Sensor Using LM35 - Instructables circuits. [online] Available from: https://www.instructables.com/Arduino-Temperature-Sensor-Using-LM35/ [Accessed 3 May 2021].
20. Wikipedia. (2021). List of temperature sensors. [online] Available from: https://en.wikipedia.org/wiki/List_of_temperature_sensors [Accessed 3 May 2021].

21. Panasonic. (2011). Temperature controllers using hysteresis. [online] Available from: https://na.industrial.panasonic.com/blog/temperature-controllers-using-hysteresis [Accessed 3 May 2021].
22. Microcontrollerslab.com. (2019). PID controller implementation using Arduino. [online] Available from: https://microcontrollerslab.com/pid-controller-implementation-using-arduino/ [Accessed 3 May 2021].
23. Wikipedia. (2021). Hysteresis. [online] Available from: https://en.wikipedia.org/wiki/Hysteresis [Accessed 3 May 2021].
24. TECHNOLOGY E. (2021). Microcontroller – Types of Microcontrollers & their Applications. [online] Available from: https://www.electricaltechnology.org/2020/05/types-of-microcontrollers.html [Accessed 3 May 2021].
25. RS Components Ltd. (2021). A Complete Guide to Microcontrollers. [online] Available from:https://uk.rs-online.com/web/generalDisplay.html?id=ideas-and-advice/microcontrollers-guide [Accessed 3 May 2021].
26. Sidhartha. (2015). Different Applications of Microcontroller – VLSIFacts. [online] Available from: http://www.vlsifacts.com/different-applications-microcontroller/ [Accessed 3 May 2021].
27. Wikipedia. (2021). Microprocessor. [online] Available from: https://en.wikipedia.org/wiki/Microprocessor [Accessed 3 May 2021].
28. Wardynski D. (2019). What is a Microprocessor and How Does it Work - Brainspire. [online] Available from: https://www.brainspire.com/blog/what-is-a-microprocessor-and-how-does-it-work [Accessed 3 May 2021].
29. Elprocus. (2021). Evolution of Microprocessor – Types of Microprocessors. [online] Available from: https://www.elprocus.com/evolution-of-microprocessor-with-applications/ [Accessed 3 May 2021].
30. Wikipedia. (2021). Electric motor. [online] Available from: https://en.wikipedia.org/wiki/Electric_motor [Accessed 3 May 2021].
31. RS Components Ltd. (2021). Electric Motors. [online] Available from: https://uk.rs-online.com/web/c/automation-control-gear/electric-motors/ [Accessed 3 May 2021].
32. Apprentice E. (2017). The basic principles of electromagnetism. [online] Available from: https://electricalapprentice.co.uk/the-basic-principles-of-electromagnetism/#:~:text=All%20magnets%20have%20a%20north,south%20pole%20of%20the%20magnet [Accessed 3 May 2021].

33. Kimbrell J. (2021). Improved stepper motor systems support more applications – Control Engineering. [online] Available from: https://www.controleng.com/articles/improved-stepper-motor-systems-support-more-applications/ [Accessed 3 May 2021].
34. RS Components Ltd. (2021). The Complete Guide to Stepper Motors [online] Available from: https://uk.rs-online.com/web/generalDisplay.html?id=ideas-and-advice/stepper-motors-guide [Accessed 3 May 2021].

35. Maximous. (2017). Controlling a Stepper Motor With an Arduino - Instructables circuits. [online] Available from: https://www.instructables.com/Controlling-a-Stepper-Motor-with-an-Arduino/#:~:text=For%20a%20stepper%20motor%2C%20the,connected%20to%20power%20the%20motors [Accessed 3 May 2021].
36. Clippard. (2021). How Stepper Motors Provide Precision Control. [online] Available from: https://clippard.com/cms/wiki/how-stepper-motors-provide-precision-control [Accessed 3 May 2021].
37. KELLINGHUSEN M. (2021) Choosing and integrating a medical device stepper motor. [online] Available from: https://starfishmedical.com/blog/medical-device-stepper-motor/ [Accessed 3 May 2021].
38. Dahl ØN. (2018). What Is an H-Bridge - buildelectroniccircuits. [online] Available from: https://www.build-electronic-circuits.com/h-bridge/ [Accessed 3 May 2021].
39. Lowe D. (2020). Digital Electronics: Darlington Arrays to Drive High-Current Outputs [online] Available from: https://www.dummies.com/programming/electronics/diy-projects/digital-electronics-darlington-arrays-to-drive-high-current-outputs/ [Accessed 3 May 2021].
40. Wikipedia. (2021). Darlington transistor [online] Available from: https://en.wikipedia.org/wiki/Darlington_transistor [Accessed 3 May 2021].
41. Components 101. (2018). 28BYJ-48 - 5V Stepper Motor. [online] Available from: https://components101.com/motors/28byj-48-stepper-motor [Accessed 3 May 2021].
42. Stan. (2014). 28BYJ-48 Stepper Motor with ULN2003 driver and Arduino Uno - 42 Bots. [online] Available from: https://42bots.com/tutorials/28byj-48-stepper-motor-with-uln2003-driver-and-arduino-uno/ [Accessed 3 May 2021].

43. Texas Instruments. (2019). ULN2003A. [online] Available from: https://www.ti.com/product/ULN2003A [Accessed 3 May 2021].
44. HobbyTronics Ltd. (2016). ULN2003A Darlington Array 7 NPN 500mA. [online] Available from: https://www.hobbytronics.co.uk/uln2003a-darlington-array [Accessed 3 May 2021].
45. Wikipedia. (2021). Rack and pinion. [online] Available from: https://en.wikipedia.org/wiki/Rack_and_pinion [Accessed 3 May 2021].
46. Britannica. (2007). Rack and pinion. [online] Available from: https://www.britannica.com/technology/rack-and-pinion [Accessed 5 May 2021].
47. Wikipedia. (2021). Worm drive. [online] Available from: https://en.wikipedia.org/wiki/Worm_drive [Accessed 5 May 2021].
48. Thompson Creek. (2020). Awning Window Sizes. [online] Available from: https://www.thompsoncreek.com/blog/standard-size-awning-windows/ [Accessed 5 May 2021].
49. Haojie Wanga, Qingyan (Yan) Chena. (2015). Energy and Buildings - Modelling of the impact of different window types on single-sided natural ventilation. [online] Available from: https://core.ac.uk/download/pdf/82655971.pdf [Accessed 5 May 2021].
50. Thompson Creek. (2020). Awning Windows and Hopper Windows. [online] Available from: https://www.thompsoncreek.com/windows/awning-and-hopper-windows/ [Accessed 10 May 2021].
51. Jr. JWB. (2015). Basic greenhouse engineering calculations - Greenhouse Management. [online] Available from: https://www.greenhousemag.com/article/basic-greenhouse--engineering-calculations/ [Accessed 5 May 2021].
52. NEMALI K. (2019). Calculating Greenhouse Heating Requirements. [online] Available from: https://www.purdue.edu/hla/sites/cea/article/calculating-greenhouse-heating-requirements/ [Accessed 5 May 2021].
53. Botanic H. (2021). Greenhouse power calculator. [online] Available from: https://hartley-botanic.co.uk/guides/greenhouse-power-calculator/ [Accessed 5 May 2021].
54. TWI Ltd. (2021). WHAT IS LASER CUTTING? - A DEFINITIVE GUIDE TO THE PROCESS. [online] Available from: https://www.twi-global.com/technical-knowledge/faqs/what-is-laser-cutting [Accessed 5 May 2021].

55. Ryan V. (2019). Continuous Improvement (CI). [online] Available from: https://technologystudent.com/prddes1/contin1.html [Accessed 5 May 2021].
56. Nikhil A. (2020). What is 3D Printing. [online] Available from: https://www.engineersgarage.com/tech-articles/what-is-3d-printing-part-1-8/ [Accessed 5 May 2021].
57. Texas Instruments. (2017). LM35 Precision Centigrade Temperature Sensors datasheet. [online] Available from: https://www.ti.com/lit/ds/symlink/lm35.pdf [Accessed 5 May 2021].

Appendix 1: Arduino code to test the LM35

```
#include <Wire.h>  // Comes with Arduino IDE

#include <LiquidCrystal_I2C.h>

float temp; //Variable where we will stock the temperature value

int tempPin = 0; //Pin used with the sensor output here it's A0


// set the LCD address to 0x27 for a 20 chars 4 line display

// Set the pins on the I2C chip used for LCD connections:

//                    addr, en,rw,rs,d4,d5,d6,d7,bl,blpol

LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE);  // Set the LCD I2C address

void setup() {

Serial.begin(9600);

lcd.begin (20,4);

}

void loop() {

temp = analogRead(tempPin); //Reading the value from the analog input

temp = temp * 500/1023; //Sensor calibration to get the real value

lcd.clear();

lcd.setCursor (0,0); //Start writing on 0.0 on lcd screen

lcd.print("Temperature");

lcd.setCursor (0,1);

lcd.print(temp); //Temperature value

lcd.print ((" C")); //Celsius of course :D

delay(1000); //Refresh every 1s

}
```

Appendix 2: Greenhouse ventilation system code

```
// Author: Vasileios Fermelis

// Get the LCD I2C Library here:

// https://bitbucket.org/fmalpartida/new-liquidcrystal/downloads


#include <Wire.h>  // Comes with Arduino IDE

#include <LiquidCrystal_I2C.h>

#include <Stepper.h>


#define STEPS 2038 // the number of steps in one revolution of your motor (28BYJ-48)


Stepper stepper(STEPS, 8, 10, 9, 11);

float Vout;

float Temp;

// set the LCD address to 0x27 for a 20 chars 4 line display

// Set the pins on the I2C chip used for LCD connections:

//                    addr, en,rw,rs,d4,d5,d6,d7,bl,blpol

LiquidCrystal_I2C lcd(0x27, 2, 1, 0, 4, 5, 6, 7, 3, POSITIVE);  // Set the LCD I2C address


void setup()

{ Serial.begin(9600);

  lcd.begin (20,4);

 }
```

```
void loop()
 {
  Vout=analogRead(A0) ;
  Temp= (Vout*500)/1023 ;
  lcd.clear();
  lcd.setCursor (0,0); //Start writing on 0.0 on lcd screen
  lcd.print("Temperature"); // In Degree Celsius
  lcd.setCursor (0,1);
  lcd.print(Temp); //Temperature value
  lcd.print ((" C")); //Celsius of course :D
  delay(1000); //Refresh every 1s

 if (Temp > 24) {
  stepper.setSpeed(16)
  stepper.step(2038); // do 2038 steps -- corresponds to one revolution in one minute
  delay(1000);
}
else if (Temp < 20) {
  stepper.setSpeed(16);
  stepper.step(-2038); // do 2038 steps in the other direction with faster speed -- corresponds to one revolution in 10 seconds
   delay(1000);
}
else { // 20 < Temp < 24
  // Safe! Continue usual tasks.
}
 }
```