.

Palestine
Polytechnic
University

College Of Information Technology and Computer Engineering

Department of Computer System Engineering

Graduation Project

Project title: Smart System to Measure Rectangular Shape Dimension(SMRD)

Project team: Lara Jbour
Ruaa Manasra
Waad Makhamreh

Supervisor : Eng.Khalid Daghmeen

Palestine

2021/2022

Acknowledgement

Praise be to God, and thank him for his blessings throughout our work on our project. He assisted us with it, inspired us with patience and diligence, and provided us with all the knowledge and invention that enabled us to complete this project successfully.

We would like to thank all the people who helped us, who helped us, directly or indirectly, with contributions to our project. A special thanks to our parents, who always encourage, support, and take care of us and always pray for us in all steps of our lives.

We are thankful to our project supervisor, Eng. Khalid Daghameen, for his insightful direction, support, encouragement, and continuous patience throughout the project's duration. Moreover, it is our duty to thank Eng. Wael Takrouri for helping us and providing assistance when needed. We are also grateful to our friends in the Computer Systems Engineering program, in the College of Information Technology and Computer Systems Engineering at Palestine Polytechnic University.

Finally, thanks to Palestine Polytechnic University and its staff for all their teaching and guidance efforts.

Abstract

There is a need for a handy device to take measurements for rectangular shapes like windows and doors in home construction. The measurement process for such tools could cause some inaccurate results, which could lead to economic loss. In addition, it could cause some injury to workers. In addition, it could save time for workers.

The target of this project is to develop a system that accurately measures the dimensions of rectangular shapes in a short amount of time while reducing errors and adverse effects produced by previous practices. The process is done by directing a laser beam from the sensor in several steps with the help of a servo motor to hit the edge of the rectangle shape in order to take several reading distances from the sensor. The reading from the sensor will be transferred to a microcontroller to be analyzed and processed using several mathematical equations, and then the result will be displayed on an LCD screen. As a result, the system was created and tested to satisfy the project's criteria, and it was discovered that it functions as expected and meets the project's requirements. So, we have a system capable of measuring the dimensions of rectangular shapes with precision.

Table of Contents

List of Figures

List of Tables

Chapter one

Introduction

1.1 Overview

SMRD is a smart system to measure rectangular shape dimensions.It's made to enable individuals, especially career owners, to find the exact dimensions as quickly as possible.

1.2 Motivation and importance

This project is considered to be challenging because the idea itself is considered to be among the latest trends in technology nowadays. The aim of this project is to add a new tool in the world of measurement tools, to find the vertical and horizontal dimensions of rectangular shapes without the need to use traditional measurement tools, such as meters. Therefore, this system will help people, especially career owners, to find the exact dimensions of rectangular shapes, and it can be described as a smart technical solution in the industrial sector because it reduces time and effort and obtains results with higher accuracy than using traditional tools.

1.3 Objectives

1) The tool is able to get the length and width of the rectangular shapes in ordinary units such as meters and centimeters.
2) The results must be shown on the LCD screen with at least one decimal digit.
3) The user will be able to interact with the system through buttons, so it can be considered a good human interface.

1.4 Problem statement

This section is about problem analysis, a list of requirements, and the expected results.

1.4.1 Problem analysis

Many people find it difficult to take exact measurements in some places, especially large spaces in tall buildings, large projects, and spaces with protrusions that could be dangerous to people's lives. As a result, when measurements are taken in the traditional way, there may be resultant errors, resulting in an inaccurate measurement, which causes economic loss and negatively affects people's lives. Also, this procedure is inconvenient and may take a long time to complete. As a result, we need a system that can precisely measure the dimensions of rectangular shapes in a short amount of time with minimal errors.

1.4.2 List of requirements

The system requirements can be summarized as:

1) Get the sensor's correct reading.
2) The system must operate as quickly as possible.
3) Obtain precise dimensions.
4) An easy-to-use and user-friendly measuring device that can be used by anyone.

1.4.3 Expected results

We expect to accomplish the following at the end of the project:

1) Exact dimension of the rectangular shapes.
2) The results are shown in the LCD screen.

1.5 Description of the system

This project is about designing a device that finds the dimensions of rectangular shapes using laser beam sensors that are directed in several steps with the help of a servo motor to hit the edge of the rectangle shape in order to take several reading distances from the sensor. The reading from the sensor will be transferred to a microcontroller to be analyzed and processed using several mathematical equations, and then the result will be displayed on an LCD screen.

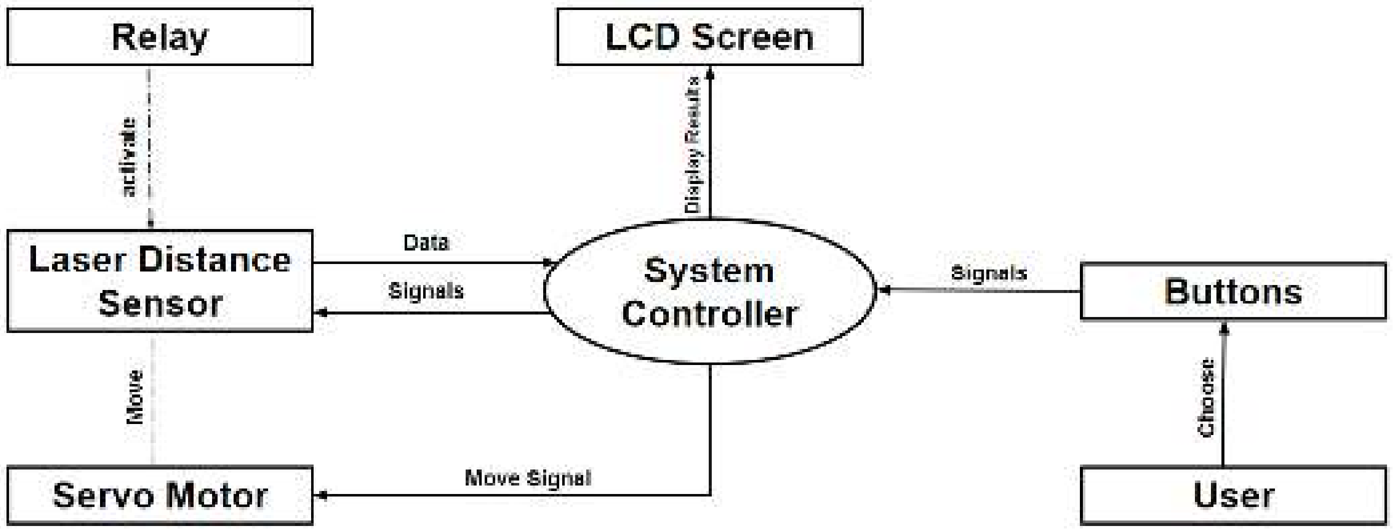

Figure 1.1:SMRD Context diagram

1.6 Overview of the rest of the report

The rest of this document will be as follows: Chapter two will be about theoretical background, a short description of the parts used in the system and why they are chosen, specifications, and design constraints. Chapter three will be about the conceptual description of the system, detailed design, schematic diagram, and block diagram. And chapter four will include description of the implementation, implementation issues, and implementation challenges. It should also include description of the method used to validate the system, validation results, analysis and discussion about the results.Finally,chapter five will be about informative summary,and it should also include future directions and future work.

Chapter two

Background

2.1 Overview

This chapter introduces the theoretical background of the system and some descriptions of the hardware and software components used in the system. Specifications of the design system and constraints are discussed too.

2.2 Theoretical background

The project's main idea is to design and implement a smart system to measure rectangular shape dimensions using a LiDAR distance sensor, an Arduino, and a servo motor. The LiDAR sensor is used to measure the distance between the sensor and the rectangular shape, then send it to the Arduino to be analyzed and processed. The results that have been processed are sent to the LCD screen to be displayed. The servo motor is used to stick the sensor on its movable shaft and make it rotate at several angles.

This project works in two modes: mode 0, which means the system will be approximately in the center of the rectangular shape, and mode 1, where the device will be at the edge of the rectangular shape.

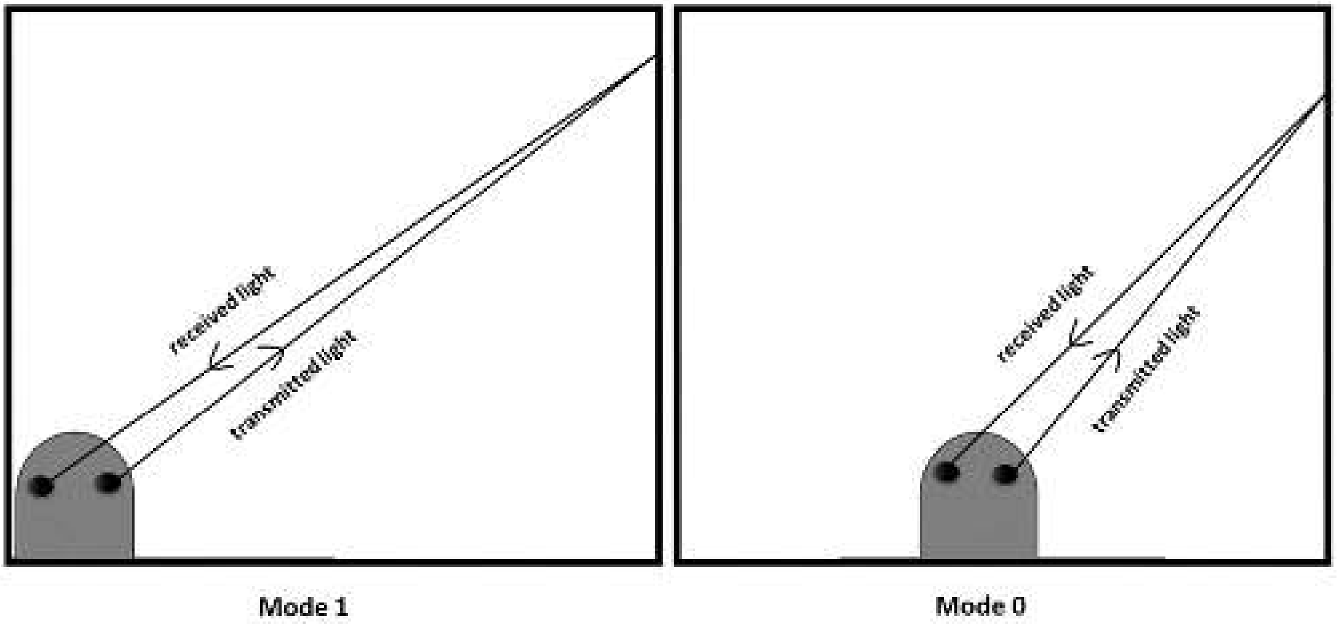

Figure 2.1: SMRD Mode

To achieve the goals of this project, we need several components. To get the distance between the rectangular shape and the device, we need one LiDAR distance sensor. This sensor has an infrared emitting diode (IRED) and a position sensitive detector (PSD), and its workflow is as follows: the IR light leaves the IRED to hit the edge of the rectangular shape and is then reflected off the PSD. The distance between them is calculated using the triangulation of the IR light, as shown in figure 2.2.[1]

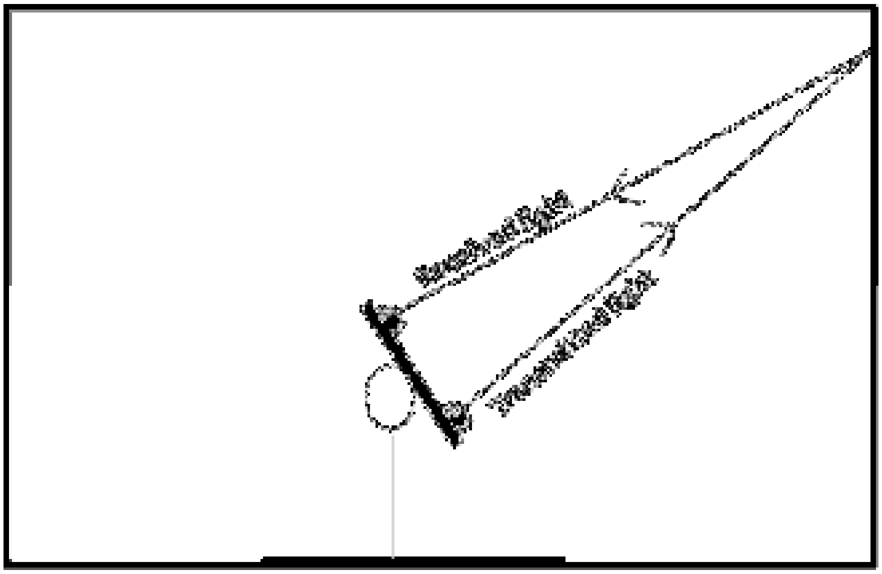

Figure 2.2 :SMRD Work principle

The output that this sensor provides is the time difference between the emission of the light and its return to the sensor after being reflected by the edge of the rectangular shape. In order to determine the distance, we need a function that converts the output time into a range of values. Based on the data from the LiDAR ToF Time of Flight Distance Sensor datasheet, we can calculate the distance like this:[2]

$$Distance\ (cm)\ = \frac{C}{2} \cdot t$$

$$c\text{: } Speed\ of\ light\ on\ air$$

$t\ = \frac{1}{2\pi f}\ = time\ difference$

But, this distance is not the value we aim for. In order to achieve the target, which is to find the dimensions of the rectangular shape, more mathematical calculation must be used in the two system modes.

2.2.1 Mode 0 ($0^{\circ} - 180^{\circ}$)

In this mode, the servo motor shaft will rotate at an angle between 0° and $180^{\circ}$with the aim of taking several different values of X and Y, which can be achieved through the following steps.

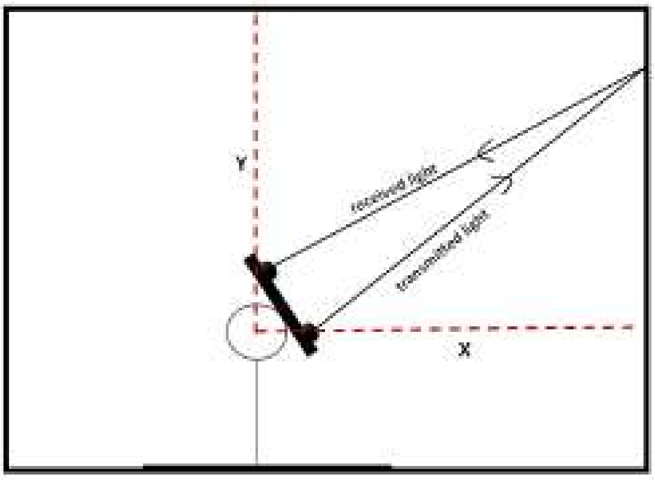

Figure 2.3 :Mode 0

The first step, if you take a line from the intersection of the transmitted light and the received light on the edge of the rectangular shape and extend it to the middle of the distance between the two lenses on the TOF sensor, by using Pythagoras' theorem, as shown in figure 2.4, you can get L by this equation:

$$L\ (cm) = \sqrt{(Distance)^{2} - (\frac{1}{2}\ W)^{2}}$$

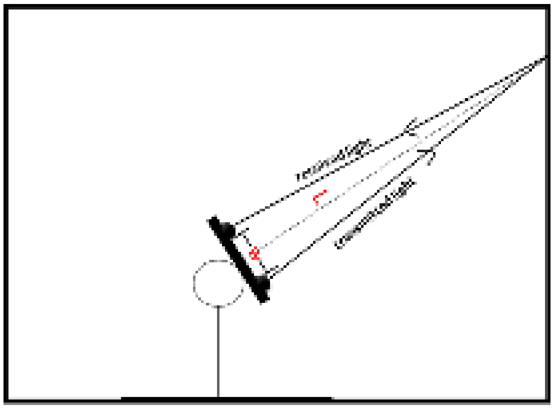

Figure 2.4: L value

The second step is to find the horizontal and vertical components of L by using displacement vector (L) analysis. As shown in figure 2.5, the horizontal component is equal to $Lcos\theta$, and the vertical component is equal to $Lsin\theta$.

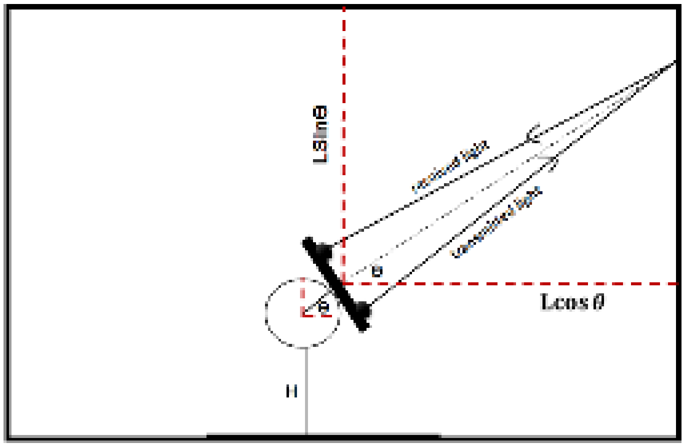

Figure 2.5: L components

The most important point in the project is to take many values of X and Y components in many different directions and angles. To achieve this, we stick the TOF sensor in the servo motor shaft, as shown in figure 2.6.

In a third step, we use the following equations to determine the required (X and Y):

$$\mathbf{X} = \mathbf{L}cos\theta$$

$$\mathbf{Y} = \mathbf{L}sin\theta$$

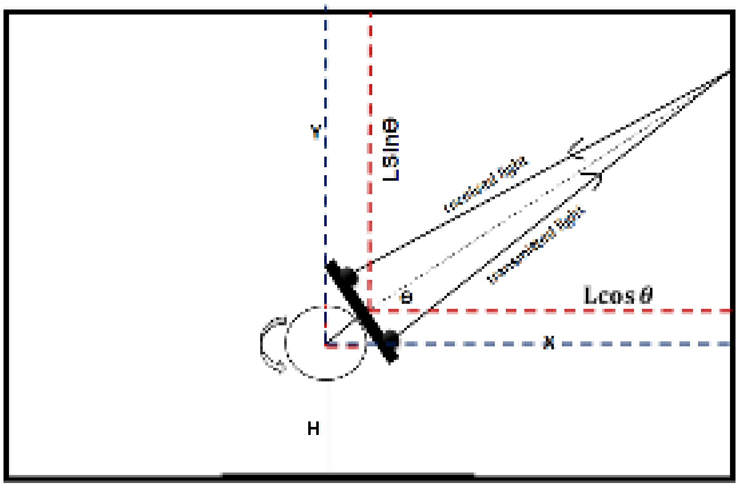

Figure 2.6: XY values

On the fourth step, as mentioned before, the servo motor shaft moves in several steps to rotate the LiDAR sensor at many angles. In this mode, after completing all the previous steps, the average of the first three values of X are added with the average of the last three, in addition to the sensor width, which is 1.1 cm, to find the required horizontal dimension (X). And according to the required vertical dimension (Y), all Y values will be sorted using a sort algorithm to arrange them in ascending order, then the average of the highest five values will be token in order to find Y, then the height of the servo motor (H) and the height of the box that we add all the components inside (System Box) will be added to the Y found after averaging.

So, $\underline{X} = avg(First\ \mathbf{th}ree\ value\ of\ X\ ) + avg(last\ \mathbf{th}ree\ value\ of\ X) + \mathbf{1.1}\ cm$

$$\underline{Y} = avg(highest\ five\ values\ of\ Y) + box\ \mathbf{height} + H$$

H: The height of the servo motor

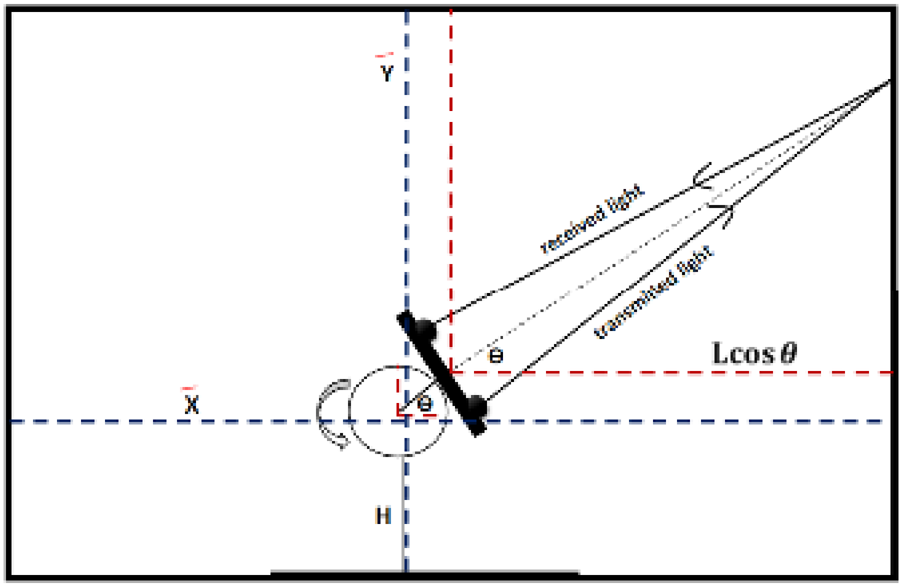

Figure 2.7:SMRD Mode0 dimensions

2.2.2 Mode 1 (0° - 90°)

In this mode, the servo motor shaft will rotate at an angle between 0 and 90 with the aim of taking several different values of X and Y. This mode is appropriate for being on the corner of a rectangular shape or a three-edged shape.

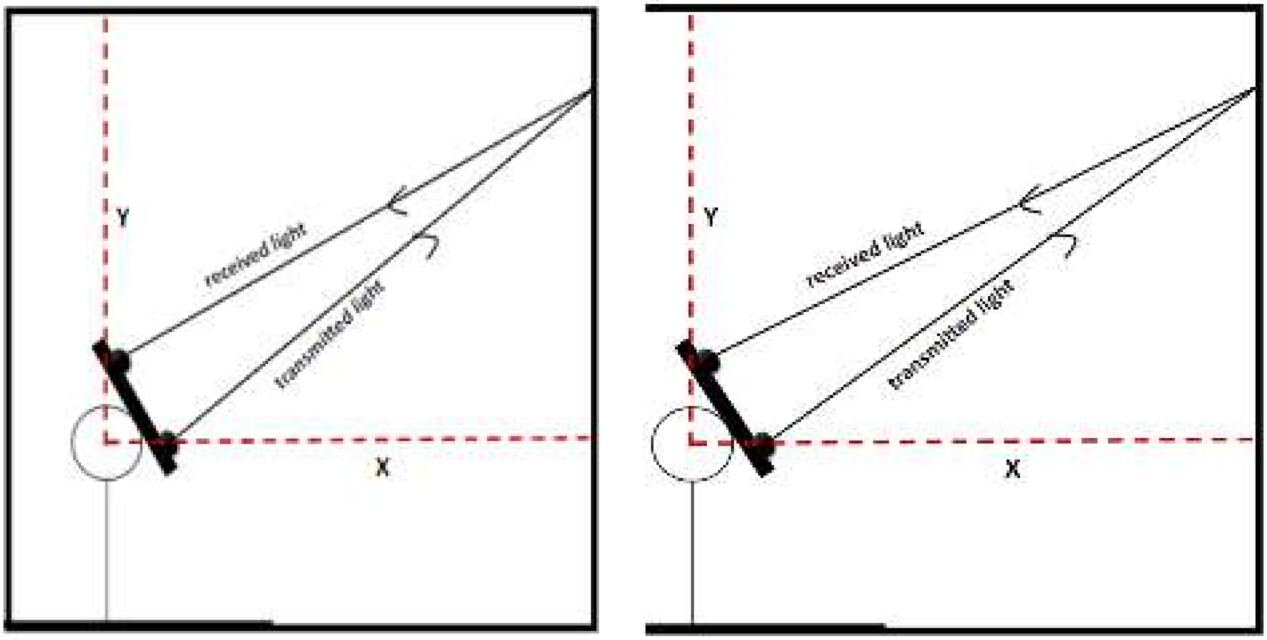

Figure 2.8: Mode 1

In this mode the same mode 0 steps are done, but the step of calculating $\underline{X}$ and $\underline{Y}$ is different.

In this mode:

$\underline{X}= avg(First\ three\ value\ of\ X\ ) +\ sensor\ width + C$

$\underline{Y} = avg(highest\ five\ values\ of\ Y) +\ \boldsymbol{The\ height}\ of\ the\ box's\ bottom\ +\ H$

$C: The\ distance\ between\ the\ sensor\ and\ the\ edge\ of\ the\ system\ box$

H: The height of the servo motor

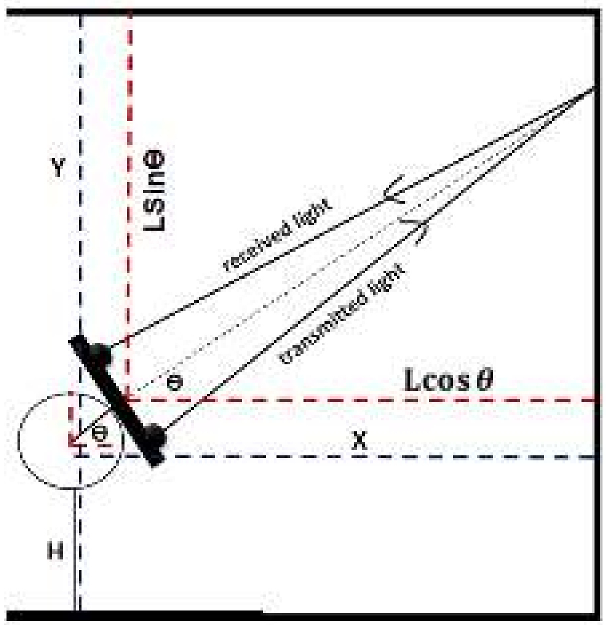

Figure 2.9: SMRD Mode1 dimensions

H: The servo motor's height.

2.3 Techniques to be used in the project

This subsection illustrates the main technologies to be used in this project and what tasks they do.

2.3.1 Arduino programming language

The Arduino Integrated Development Environment, or Arduino Software (IDE), contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions, and a series of menus. It connects to the Arduino hardware to upload programs and communicate with them.It is available on several operating systems, which gives us programming with integrated library support and a way to easily compile and load our Arduino programs onto a board connected to the computer. And it supports the languages C and C++ using special rules of code structuring.[3]

2.4 Hardware components of the system

This part describes the components used in the system.

1) **TFmini-S LIDAR Micro Distance Sensor**

The TFMini-S Micro LiDAR Module is a ToF (Time of Flight) LiDAR sensor capable of measuring the distance to an object as close as 10 centimeters and as far as 12 meters. It has a unique optical, structural, and electronic design. TFmini-S has the advantages of low cost, small volume, and low power consumption. It also has multiple interfaces to meet different requirements.[4] And the following are its main distinguishing characteristics.

Operating Voltage	5 volt
Operating Current	⩽ 10 mA
Operating Range	0.1 - 12 m
Operating Temperature	0 ℃ - 60℃
Measurement Unit	cm
Range Resolution	1 cm
FOV	2°
Frame Rate	1 - 1000 Hz
Accuracy	± 6cm @(0.1-6m), ± 1%@(6m-12m)
Dimensions	42x15x16 mm
Price	50 $

Table 2.1:TFmini-S Distance Sensor Characteristics[2]

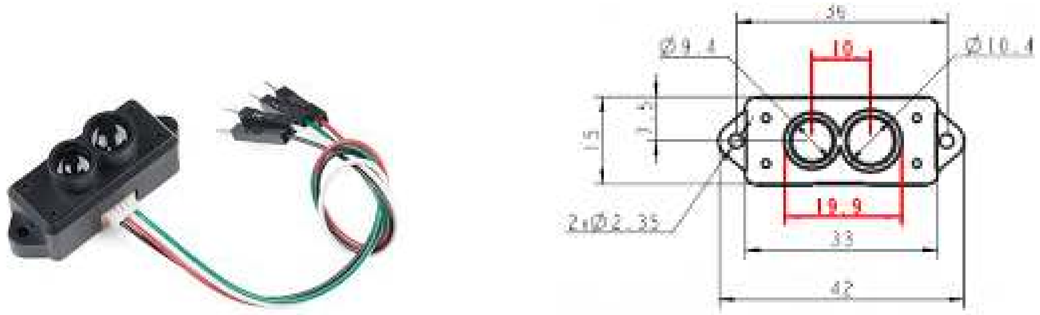

Figure 2.10:TFmini-S Lidar Distance Sensor [2]

2) Arduino Nano

It is a microcontroller board, small, complete, and based on the ATmega328P. It offers the same connectivity and specs as the Arduino Nano board in a smaller form factor. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a battery to get started.[5]

Microcontroller	ATmega328P
Operating Voltage	5V
Digital I/O Pins	14
PWM Digital I/O Pins	6
Analog Input Pins	8
DC Current per I/O Pins	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB
SRAM	2 KB
EEPROM	1KB
Clock Speed	16 MHz
Length	45 mm
Width	18 mm
Price	15 $

Table 2.2 :Arduino Nano Specification [5]

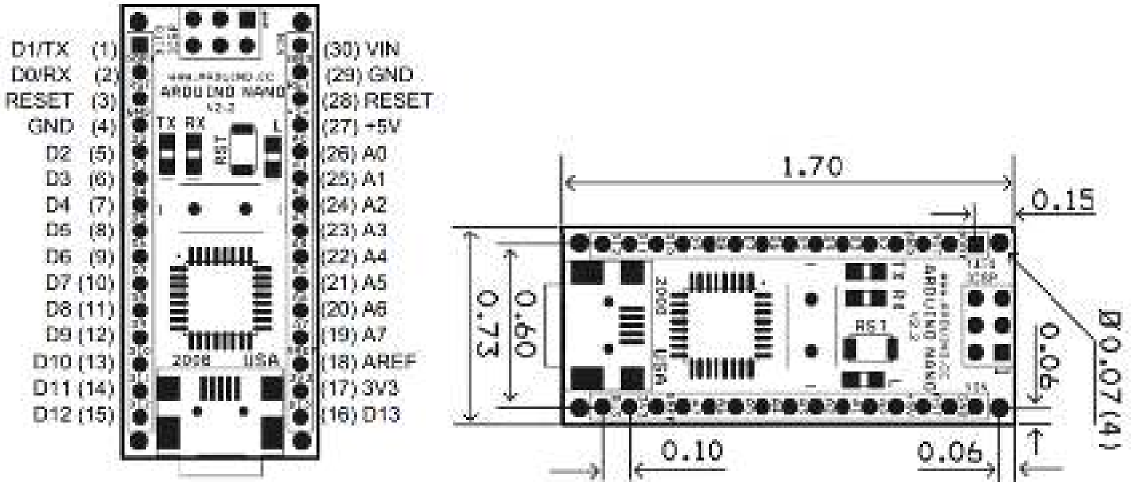

Figure 2.11:Arduino nano pinout & dimension [12]

3) Servo Motor

A servo motor is a DC motor integrated with a gear train, a shaft encoder, and some control logic so that it is easier to use. They have a limited rotation, typically 180°.It has a 3-pin interface with power, ground, and a control input The servo's control logic drives the shaft to a position determined by the duty cycle of the control input. The servo's shaft encoder is typically a rotary potentiometer that produces a voltage dependent on the shaft position.[11]

Voltage Rating	4.8 V - 6.0 V
Weight	9g
Speed	0.12sec/60degrees
Torque	1.5kgf-cm
PWM	50 HZ
Connector type	JR type (Yellow: Signal, Red: VCC, Brown:GND)
Product Dimensions	32 × 11.5 × 24 mm
Price	7 $

Table 2.3 : Servo Motor Specification[11]

4) Relay

The relay module is an electrically operated switch that can be turned on or off [9], deciding to let current flow through or not. They are designed to be controlled with low voltages like 3.3V or 5V.It has three high-voltage terminals that connect to the device that it wants to control, which are: common terminal, normally closed terminal, and normally open terminal. There are also 3 additional voltage pins which connect to the microcontroller, which are: ground, 5V VCC, and a signal pin, and it is considered compatible with any 5V microcontroller such as Arduino. The relay module has one or more channels to perform the communication, and we used a one-channel relay in this system to control the time that the LIDAR sensor is turned on[8].

Operating Voltage	5V
Input Current	100 mA
Rated Load	10A (NO) 5A (NC)
Input interface	Digital
Dimension	43 x 17 x 17mm

Table 2.4 : Relay Specification[8]

5) PowerBank:

Recharge battery-powered electronics,it can be charged up using a USB charger when power is available,and then used to charge the system.

Capacity	10000mAh
Input	5v / 1A
Output1	5v / 2.0A
Output2	5v / 1A

Table 2.5 : PowerBank Specification

6) LCD Screen:

Use an I2C interface to communicate with the Arduino nano to display horizontal and vertical dimensions.

Size	80x36x20mmz(3.1x1.4x0.7in)
Interface	I2C/TWI x1,Gadgeteer interface x2
Supply Voltage	5V
Compatibility with	Arduino/Genuino UNO, Leonardo, Mega, 101 (Intel Curie), Micro, Nano, Mini
I2C Address	0x20-0x27(0x20 default)

Table 2.6 : I2C LCD screen Specification[10]

7) **Connection Series:**

Set of Male to male, male to female, female to female jumper wires.

8) **Buttons:**

We have four buttons (mode0,mode1,Power,Test), used to control the device.

9) 100 *Ω* resistor.

2.5 Constraints:

1) This project cannot handle rectangle shapes without edges.
2) This project does not handle irregular shapes.

Chapter three

System design

3.1 Overview

The following section has a description of the system, detailed design, and necessary information about the design.

3.2 Detailed description of the system

Our system will be designed to measure the rectangular shape dimensions. When the system is activated, and after the user selects the appropriate mode, the relay activates the TF Mini LiDAR sensor to measure the distance between the sensor and the rectangular shape by taking several readings by directing the TF Mini LiDAR sensor at several angles using the servo motor, and then it will send it to the Arduino to be processed inside, get the dimensions, and finally send it to the LCD screen to be displayed.

3.3 System diagrams

In this subsection, three diagrams are represented for the understanding of the project concepts and design.

3.3.1 System block diagram

Figure 3.1 is the general block diagram of the project. As illustrated below, the **TF Mini LiDAR sensor** when the system is activated will collect the necessary data and send it to the **Arduino Nano** .After the arduino gets it, it will examine this data and will do some mathematical operations on it, regarding to find the dimensions of rectangular shape.And then the results will be shown in LCD screen.

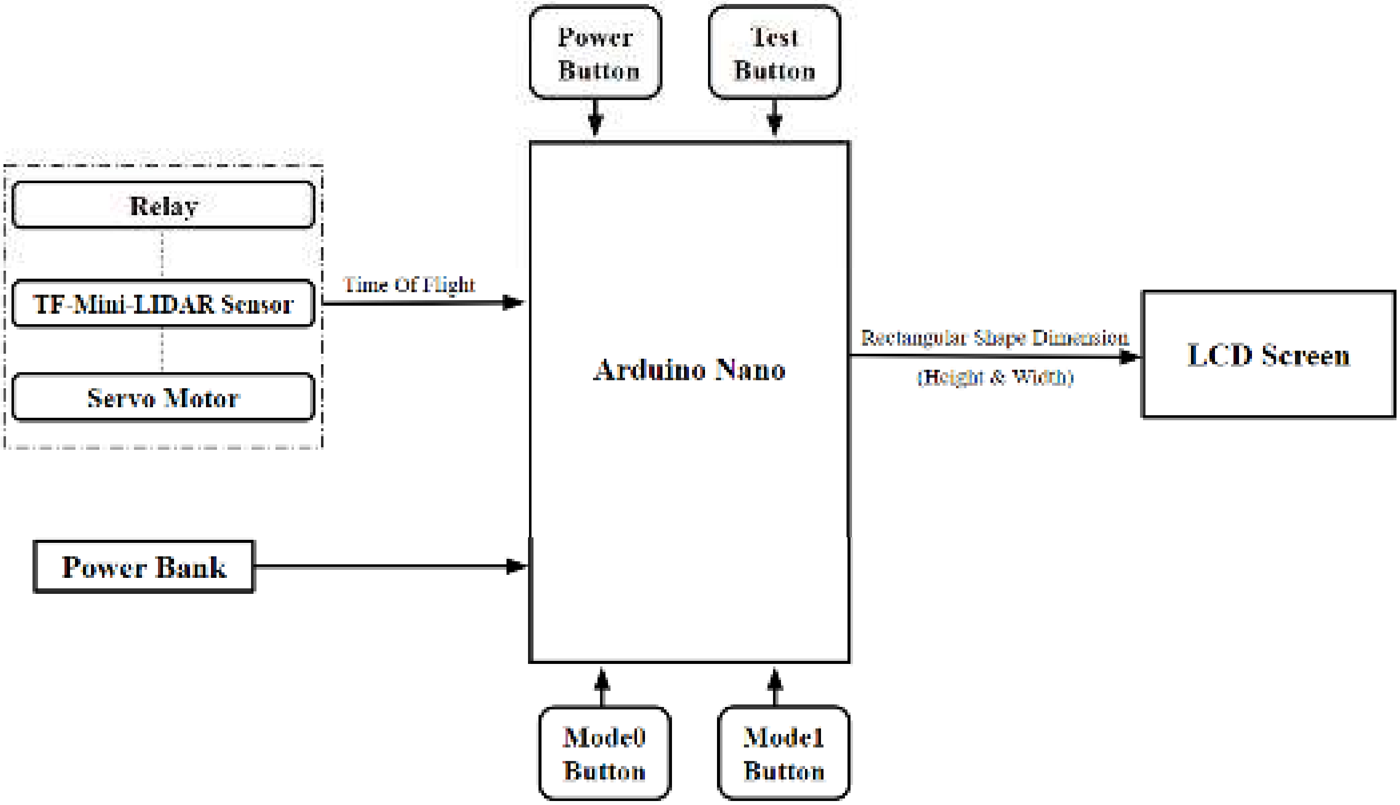

Figure 3.1: SMRD General block diagram

3.3.2 Schematic diagram

Figure 3.2 figure describes the system elements and how they are connected.

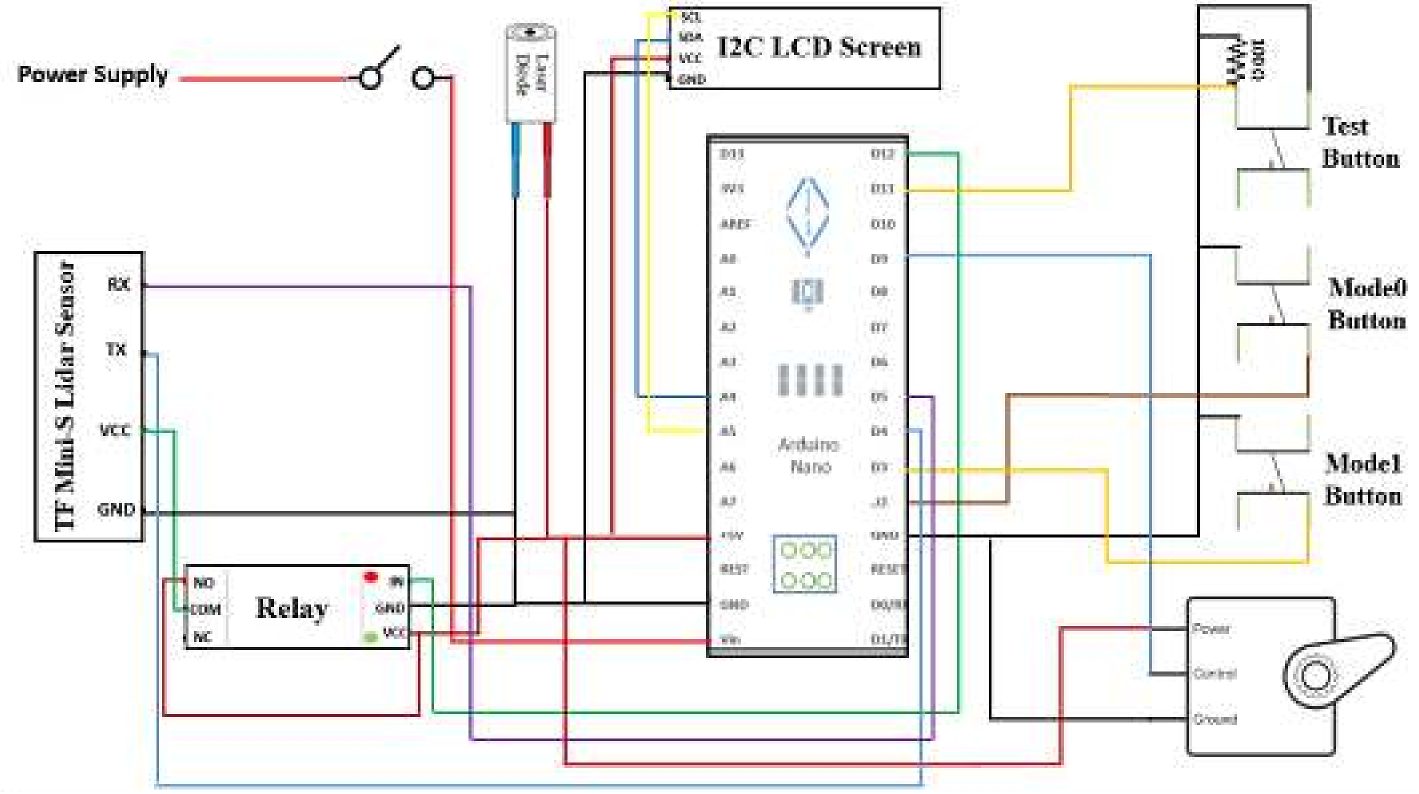

Figure 3.2: SMRD Schematic diagram

1) **Arduino Nano** : The main component in the system, which will be linked to the other components. Represent the inlet and outlet of the signals and information. and it will also send the data to the LCD screen and mobile application to display the results.
2) **TF Mini LiDAR sensor:** It is used to find the dictance between it and the rectangular shape.
3) **Servo Motor:** It used to rotate the TF Mini LiDAR sensor in several directions.
4) **LCD screen:** It used to display the rectangular shape dimenssions.
5) **Relay** : It used to activate and deactivate TF Mini Lidar sensor

3.4 Flow Chart

This diagram shows the flow chart of the system:

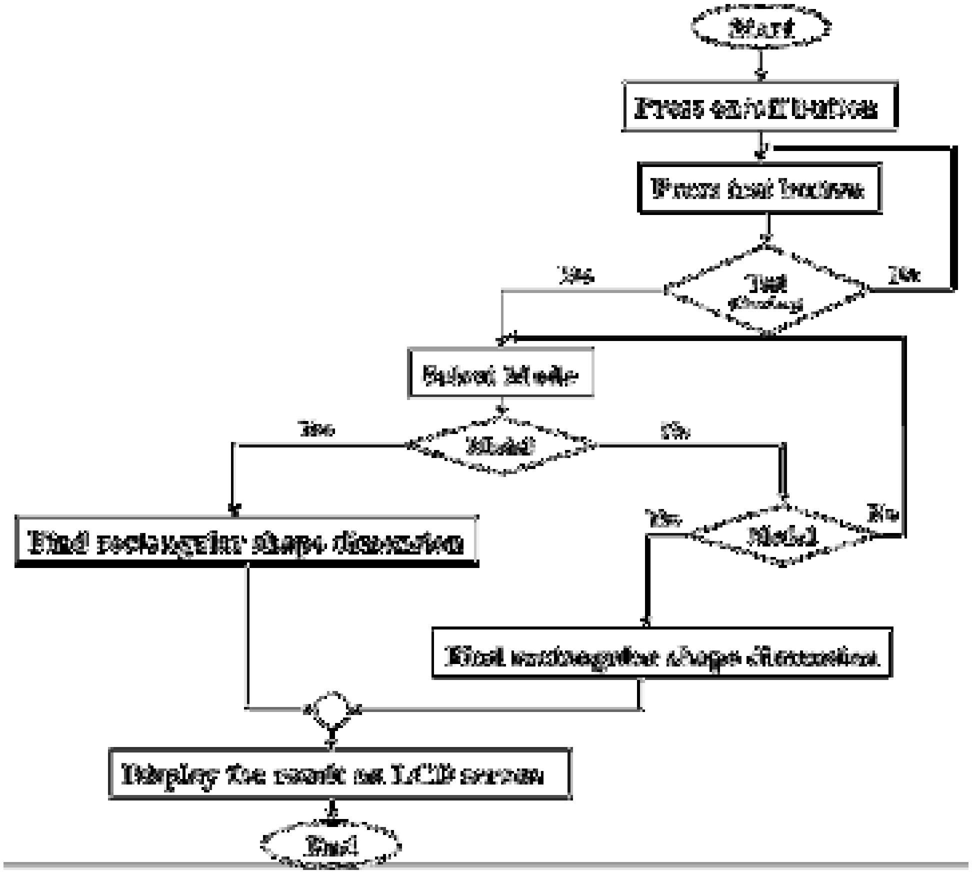

Figure 3.3: SMRD Flowchart

3.5 Detailed design

In this section we will talk about Hardware and Software be used in our project that summarized below:

3.5.1 Hardware

- **Arduino we will choose Nano R3, specifications for this:**

  Arduino Nano is a microcontroller board based on the ATmega328P and 5V oparating voltage . It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 8 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.[5]

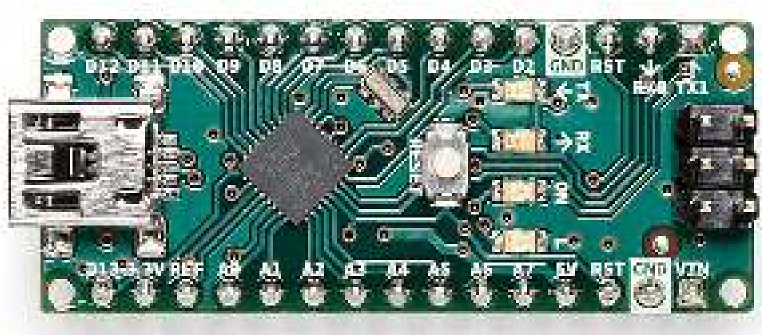

Figure 3.4 : Arduino Nano R3 [5]

3.5.2 Software

- **Arduino IDE**

  The Arduino Integrated Development Environment (IDE), often known as the Arduino Software (IDE), makes writing code and uploading it to the board simple. The Arduino Nano boards can be utilized with this software. A text editor for creating code, a message area, a text console, a toolbar with buttons for common functions, and a series of menus are all included in this IDE. It connects to the Arduino hardware, allowing it to upload and communicate with programs.[3]

Figure 3.5: Arduino IDE logo [7]

Chapter four
System implementation, system testing and discussion

3.1 Overview

This chapter includes a description of the implementation, implementation issues, and implementation challenges. It also includes a description of the method used to validate the system, validation results, analysis, and discussion of the results.

4.2 Description of the implementation

4.2.1 Testing the hardware components

It was important to test all of the hardware components separately as a preliminary step toward the project's success.

1) Testing the TFmini-S lidar sensor through the test software available on the sensor's official website (en.benewake.com)[2]
   - Download the test software.
   - As illustrated in Figure 4.1, connect the TFmini-S to the PC using a TTL-USB board and USB cable.
   - Open the PC software and select (① TFmini) and select automatically recognized occupied serial port (here it is ② COM9), as shown in Figure 4.2. Then click (CONNECT). Upon successful connection, The continuous images of the output data will be displayed in area (④ TIME LINE CHART) on the right. Besides, the real-time data of the Current measure distance (Dist), effective data points per second (Effective Points) and signal strength (Strength) will be displayed in area (⑥ REAL TIME DATA) below.[2]

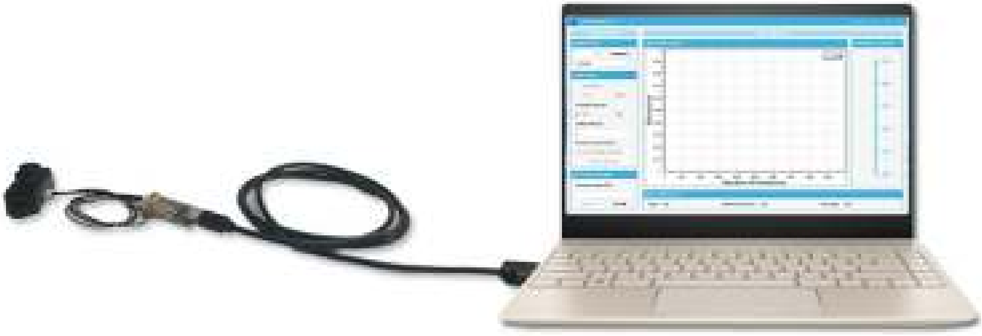

Figure 4.1 Schematic Diagram of correct Connection [2]

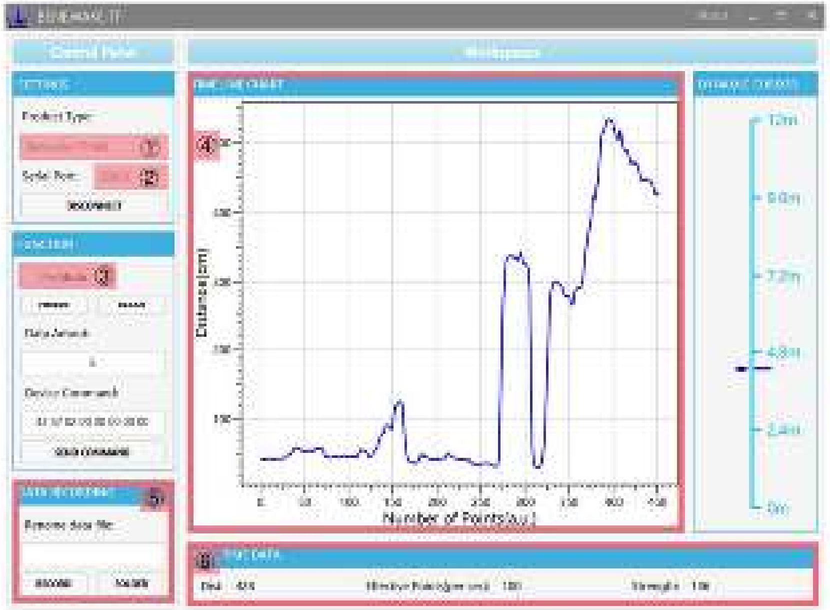

Figure 4.2 Software Interface and Display[2]

2) Testing the Arduino Nano using a USB cable to connect it to a PC and run simple code on the Arduino IDE software.

3) Test the servo motor by running a simple rotation code on the Arduino IDE.

4) Test the buttons by making a simple LED and push button connection to ensure they operate reliably.

5) Test the relay by making a simple LED connection with the help of the Arduino IDE.

6) Test the LCD Screen by making a simple code using Arduino IDE.

4.2.2 Connection of hardware components

1) Connect the TFmini-S lidar sensor with the Arduino Nano.
   - TX from the sensor to Arduino pin 4.
   - RX from the sensor to Arduino pin 5.
   - GND from the sensor to Arduino pin GND.

2) Connect the relay with the Arduino Nano.
   - VCC from the relay to Arduino pin 5v.
   - GND from the relay to Arduino pin GND.
   - Signal pin(IN) from the relay to Arduino pin 12.
   - The relay's normally closed pin is connected to Arduino pin 5v.

3) Connect the TFmini-S lidar sensor with the relay.

- The relay's common pin is connected to the VCC sensor pin.

4) Connect the servo motor with the Arduino Nano.

- The servo motor's PWM pin to Arduino pin 9.
- VCC of the servo motor to Arduino pin 5v.
- GND from the servo motor to Arduino pin GND.

5) Stick the TFmini-S lidar sensor on the servo motor shaft using adhesive.

6) Connect the **test** push button that rotates the servo from 0° to 180° by making a normally open connection with Arduino pin 11.

7) Connect the **mode 0** push button that rotates the servo from 0° to 180° by making a normally open connection with Arduino interrupt pin 2.

8) Connect the **mode 1** push button that rotates the servo from 0° to 90° by making a normally open connection with Arduino interrupt pin 3.

9) Connect the ON/OFF switch and power supply (power bank) with the Arduino Nano.

- Connect the ON/OFF switch's pin 1 to the Vin of the Arduino pin.
- Connect the ON/OFF switch's pin 2 to the positive side of the power supply.
- Connect the negative side of the power supply to the GND pin of the Arduino Nano.

10) Connect an I2C LCD screen to an Arduino Nano.

- A4 from Arduino is connected to the SDA pin of the I2C LCD screen.
- A5 from Arduino is connected to the SCL pin of the I2C LCD screen.
- The 5v from the Arduino is connected to the VCC pin of the I2C LCD screen.
- The GND from the Arduino is connected to the GND pin of the I2C LCD screen.

11) Connect a laser led light to an Arduino Nano.

- The negative side of the led laser light is connected to the Arduino's GND pin.
- The positive side of the led laser light is connected to the Arduino's 5v pin.

4.3 SMRD Work

- The user must press the **test** button on the device to ensure that the system is placed in the closed regular path and that there was no scattering of the sensor beam because, as shown in Figure 2.10, if the light spot reaches two objects at different distances, the output distance value will be a value between the two objects' actual distance values .[2]
  - ➔ This test step is done by putting the device in the rectangular shape and directing the laser light along the closed regular path of the rectangular shape. If there is no scattering, the user can proceed to the next step; if there is any scattering, the user must move the system and perform another test. So for a high accuracy requirement in practice, it's important to check the position of the device.

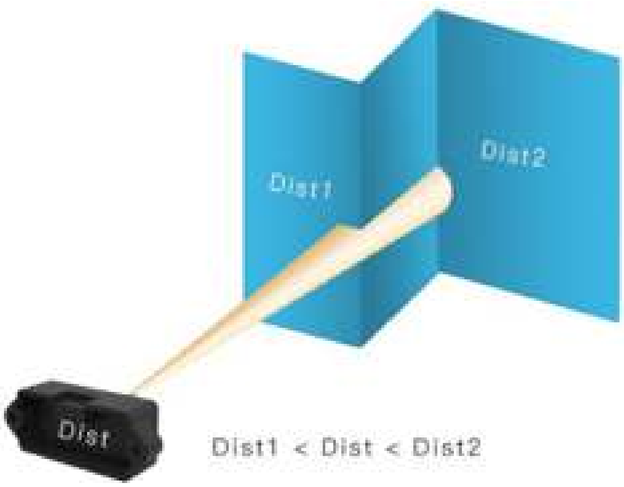

Figure 4.3: TOF Distance Measurements [2]

- After ensuring that the system is in an appropriate position, the user is free to select the mode (mode0 or mode1) he wants.
  - ➔ If he presses mode0, over a period of 60-80 seconds, the system will rotate from $0^\circ \, to \; 180^\circ$ and display the height and width of the rectangular shape on the LCD screen.
  - ➔ If he presses mode1, over a period of 30-50 seconds, the system will rotate from $0^\circ \, to \; 90^\circ$ and display the height and width on the LCD screen.

4.4 Implementation issues and challenges

- Synchronization of servo motor movement with LIDAR sensor reading. We solve this challenge by decreasing the sensor frame rate from 100 Hz to 1Hz and using the **millis()** and **map()** functions to limit the time range based on the current time.
- As soon as the device was powered up, the sensor began reading without selecting a mode, resulting in the repeating of several readings and inaccurate results. We overcome this problem by connecting the sensor to the Arduino with a relay that acts as a switch and only powers the sensor when we press any mode button.
- The push button did not work properly when the large group of instructions was called after pressing the mode0 and mode1 buttons, and this caused the system to stop working. We solve this problem by connecting the mode0 and mode1 buttons to the Arduino nano interrupt routine pins.So, by this connection, the system executes the instructions as a system interrupt without any stopping work.
- The plan was to put the laser light on the same point where the LIDAR sensor beam hit the rectangular shape to make a test step more uniform, but this idea was causing errors in sensor reading because it's affected by any outside laser light. We solved this problem by putting the laser light parallel to the sensor beam.

4.5 Validation result

4.5.1 Hardware testing

The system works as predicted and acquires the correct horizontal and vertical rectangular shape dimensions after connecting all of the components (arduino, sensor, servo motor, relay, buttons, and LCD screen) together and connecting the system to the power supply (power bank).

As a result, we get the following:

➔ The SMRD system can obtain a correct rectangular shape dimension with high precession.

- $\pm 4\ cm\ \ for\ distances\ less\ than\ 6\ m.$
- $\pm 2\ cm\ \ for\ distances\ from\ 6\ m\ to\ 12\ m.$

➔ The result is displayed on an LCD screen.

This image shows the system box details.

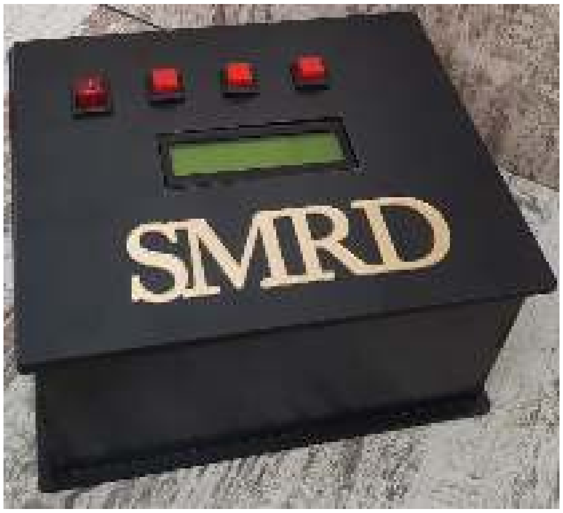

Figure 4.4: SMRD System

4.5.2 Software testing

The system was thoroughly tested to ensure that it functioned properly, and the results of the testing were successful. The following tables summarize some of the tests that we executed.

#	Case	Expected Output	Obtained Output	Pass / Fail
1	Activate the sensor using the relay.	The sensor starts reading after being activated.	The sensor starts reading successfully.	Pass
2	Get data from the sensor	Get correct data from the sensor.	Get data successfully	Pass
3	Servo Motor rotate *90*°	Rotate *90*°	Rotate successfully	Pass
4	Servo Motor rotate *180*°	Rotate *180*°	Rotate successfully	Pass
5	Synchronization of sensor reading and motor rotation	Synchronize reading and rotation.	Synchronization success	Pass
6	Press the on/off button.	When pressed, the system is activated.	The button worked successfully.	Pass
7	Press on Mode 0/1 buttons	When pressing any button, the system works correctly.	The buttons are working successfully.	Pass
8	Show the result on the LCD screen.	Display the final result on the LCD screen.	The results are displayed on the screen successfully.	Pass

Table 4.1:System test

4.6 Discussion

There were various objectives for this project at the beginning. This system, which consists of various components, has been used to achieve this. We can claim that it achieved the expected goals, that it operated as required and gave the correct rectangular shape dimension after developing this system and doing the necessary tests to confirm that it works in the appropriate manner and whether it accomplishes the goals of its development or not.

Chapter five
Conclusion and Future Work

5.1 Overview

This part will talk about the summary of all parts and future work.

5.2 Conclusion

A SMRD system can solve the rectangular shape measurement problems that we get when using traditional measuring tools such as the meter that cause inaccurate measurements, threaten people's lives in tall buildings, and take a lot of time and effort.The SMRD system allows users to find highly accurate rectangular shape dimensions without wasting effort and time, and they can easily see the results on the LCD screen.

This system has been built and verified to work as required to achieve the requirements of this project, and it was found that it works as required and meets the needs of this project.

5.3 Future work

In the future, we look forward to improving this system by adding new features:

- Develop this system to adapt to irregular shapes, not just rectangular shapes.
- Develop this system to deal with distances greater than 12 meters.

References:

[1] how2electronics. Accessed 12 2021.Available from:

https://how2electronics.com/how-to-use-tfmini-s-lidar-distance-sensor-with-arduino/#TFMini-S_Micro_LiDAR_Distance_Sensor

[2] robotshop.Accessed 12 2021.Available from:

https://www.robotshop.com/media/files/content/b/ben/pdf/sj-pm-tfmini-s_a00_product_mannual_en_1_.pdf

[3] arduino.Accessed 12 2021.Available from:

https://docs.arduino.cc/software/ide-v1/tutorials/arduino-ide-v1-basics.

[4] sparkfun.Accessed 12 2021.Available from:

https://www.sparkfun.com/products/16977

[5] Arduino Project Hub.Accessed 12 2021.Available from:

https://create.arduino.cc/projecthub/products/arduino-nano-r3

[6] Amazon.Accessed 12 2021.Available from:

https://www.amazon.com/gp/product/B0713XK923/ref=as_li_tl?ie=UTF8&camp=1789&creative=9325&creativeASIN=B0713XK923&linkCode=as2&tag=shophow2elect-20&linkId=6c191eec3495e493aeac7d5c4d5a4863

[7]freebiesupply.Accessed 12 2021.Available from:

https://freebiesupply.com/logos/arduino-logo/

[8]SeeedStudio,Accessed 23 May 2022.Available from:

https://www.seeedstudio.com/blog/2020/01/03/arduino-tutorial-control-high-voltage-devices-with-relay-modules/

[9]Random Nerd Tutorials,Accessed 23 May 2022.Available from:

https://randomnerdtutorials.com/guide-for-relay-module-with-arduino/

[10]DFRobot, Accessed 23 May 2022.Available from:

https://www.dfrobot.com/product-135.html

[11]Science Direct, Accessed May 2022.Available from:

https://www.sciencedirect.com/topics/computer-science/servo-motor.

[12]Arduino , Accessed May 2022.Available from:

https://www.arduino.cc/en/uploads/Main/ArduinoNanoManual23.pdf