PROJECT TITLE: DESIGN OF AN AUTOMATED IRRIGATION SYSTEM USING AN ARDUNIO MICROCONTROLLER TO DEVELOP A MORE EFFECTIVE AND CONVENIENT AUTOMATED SYSTEM WHICH IS CAPABLE OF MONIOTRING, TESTING AND INCREASING THE PRODUCTIVITY OF CROPS WITH LITTLE OVERHEAD FOR FARMERS OR GARDENERS IN TRINIDAD. THIS WILL BE SHOWN ON AN LCD DISPLAY.

STUDENT:
KYLE MOHAMMED 89171

SUPERVISOR:
SATISHWAR BALKISSOON

Declaration

I, the undersigned student, declare that the work contained in this report is the result of my own study and is free from plagiarism.

The contributions to the project by each student are as follows:

STUDENT NAME	STUDENT ID	CONTRIBUTION	SINGNATURE
Kyle Mohammed	89171	100% of the project	*Kyle Mohammed*

Abstract

In Trinidad the push towards wider economic diversification has highlighted agriculture as a potential sector for transformation. It was stated that T&T will not suddenly become capable of being self-sufficiency in food cultivation as manual labour will continue to limit any efforts to create that. The ongoing global pandemic have caused farmers to face a range of problems, such as the loss of jobs and more farms being operated manually with fewer hands. Traditionally, manual labour is being replaced with semi-automated and automated techniques. Therefore, this project is trying to develop a more effective and convenient automated system which is capable of monitoring, testing and increasing the productivity of farms or gardens in Trinidad. This further seeks to implement the use of the Arduino microcontroller with its code and an LCD display. This will be achieved by code, using the Arduino IDE software.

Acknowledgments

I would like to thank God and the internet for their assistance in the completion of this project.

Table of Contents

List of Tables

List of Figures

Introduction

This project implements the microcontroller to design an effective automated irrigation system which can be used by farmers or gardeners in Trinidad.

Background

In recent years the President and Minister of Agriculture stated the agriculture sectors needs an improvement, such that funds were invested to allow for more productivity for farmers. As the population increases and food supplies imported raise, there will be a serious food problem around the world. Most of the suitable and accessible agricultural land in Trinidad has already been occupied, some being cultivated in a semi-automated farm while others are not. With the use of technology such as computers, microprocessors, sensors, integrated circuits and microcontrollers, it is possible to effectively and efficiently utilize these lands. Effective irrigation and maintenance is one of the key requirements in agriculture. In Trinidad, there are mixtures of systems which are both manually operated and automated. This technique is time consuming and labour intensive, moreover it is expensive and inefficient in the long term. Rajpaul and Jain (2011) first suggested the automated concept of irrigation which has been found to be more effective. As such, the traditional technique is increasingly being replaced by semiautomatic and automated techniques (Meir et al, 1990; Clemens, 1990).

Rationale

Considering the food requirements of a rapidly growing population and the cost of imported foods, automated/ automation, efficient methods of fertilization, irrigation and pest control are required. This gives rise to the need for and use of microprocessors and computerized control systems to maximize the output of farms with lower overhead.

Literature Review

The economic recession in recent years has forced the government of Trinidad to closely examine its consumption of foreign goods, prioritizing production and self-sustainability as the national food import bill has become a concern (*Trinidad And Tobago Prioritises Agriculture, Crop Diversification And Food Security, 2020)*. The push towards wider economic diversification has highlighted agriculture as a potential sector for transformation. It was stated that T&T will not suddenly become capable of being self-sufficiency in food cultivation as manual labour will continue to limit any efforts to create that (Willaims, L et al., Farming Problems, 2020). The ongoing global pandemic have also caused farmers to face a range of problems, such as the loss of jobs and more farms being operated manually with fewer hands. Traditionally, manual labour is being replaced with semi-automated and automated techniques which considers the needs of a rapidly expanding population of a country. To increase the food production modern tools are used. In order to do so, supervisory automatic control systems are used (S.R kumbhar et al., Multi-Terminal distributed control system, Nov 1997, pp5), since in many processes, factors like soil, salinity, irrigation, temperature, light intensity etc needs repeated tasks and have to work in abnormal, environmental, conditions. This lead to develop the microprocessors as well as the computerized control systems (Proceeding of the international multiconference of engineers and computer scientist 2013 volume II, March 13-15,2013). Due to the T&T dollar exchange rate on international markets using these types of tools are becoming a greater option to go towards.

According to the central Statistical Office, import values between January to November 2017 were higher than usually on stapled food products. (*Trinidad And Tobago Prioritises Agriculture, Crop Diversification And Food Security, 2020*)

It was indicated that microcontrollers and assembly language were used in the creation of similar automated irrigation systems, to measure the humidity or the temperature. Depending upon the condition it was given, it will provide the needed water for the crops respectively etc (Providing Water to the plants automatically using microcontroller, 2018).

This can be used in different times of the year such as the dry season being the most effective if implemented in Trinidad's agricultural farms. When dealing with plants the humidity sensors, sensors which senses, measures and reports both moisture and air temperature (Humidity sensor From Nikita Gupta, 9 June 2017), is placed inside the soil near each tree/ crop. In previously built system, such as this project, components such as Sensors, comparators, signal conditioning circuits, relay logic systems, interfacing cards and system software are used. (Proceeding of the international multiconference of engineers and computer scientist 2013 volume II, March 13-15,2013)

In this project, I intend to use a microcontroller and an LCD display to display the information to the farmer or operator from a comfortable area other than that of manually doing the task. Although a study was conducted which resulted in three individuals discussing on low-cost information monitoring systems, it was stated that, "the existing information monitoring solutions are inefficient and limited in communication range".

They further corrected this by proposing a tree-based communication mechanism, which is used to extend the communication range by adding intermediate nodes. (Information Monitoring System for Smart Farming Applications, 2020-04-22)

Furthermore, to measure the humidity of soil, temperature, light intensity and depending upon the condition it of the microcontroller it will provide water when it is needed so on. This can further be shown on an LCD display which allows the farmer to monitor the process of his farm from an area. This would be extremely effective as it saves the farmer time and energy, allowing the farmer to monitor multiple farms or areas at the same time. Compared to the research found, an alarm signal will be implemented to notify the farmer of any issues that would arise, example from tanks or an excess amount of water within plants etc, therefore can also result in the microcontroller being used to drain and recycle the water. In conclusion to accomplish this, I will use the A technique called the "SMART" outcome goal. The general objective with this project is to develop an effective and convenient automated system which is capable of monitoring and increasing the productivity of crops with little overhead.

If plants get water at the proper time, then it helps to increase the production from 25 to 30% (Proceeding of the international multiconference of engineers and computer scientist 2013 volume II, March 13-15,2013). The Specific objectives are to minimize human labour used in irrigation and to provide an easier way of accessing the system from specific areas for famers.

The objectives of this project are outlined in the Objectives page.

Objectives

1. **To design and build an automated monitoring irrigation/ plant system which can be used by farmers and gardeners.**

2. **To make a model of where this can be used and implemented. This will demonstrate the functionality of the system.**

3. **To write a program using an HMI software (citect / Blynk- IOT for Arduino app) to link this with the Arduino microcontroller. This will provide a visual display of the process/ processes.**

4. **To Write a Code using Arduino IDE software and link this to the Arduino microcontroller.**

Scope

The focus of the project was to design, construct and test an Automated irrigation system using the Arduino mega 2560 microcontroller, moisture sensor, water point, relay and water level sensor. The programming of the Arduino microcontroller and visual display would be done via the program, Arduino IDE which would be directly linked to it by computer connection. In the future, there can be much improvement on the project depending on the needs of the individual.

Methodology

Project Components

Arduino 2650 Mega microcontroller

The Arduino microcontroller is an open source programable microcontroller board that is based on a microchip and is developed by Ardunio.cc. These boards are equipped with sets of digital and analog input/output (I/O) pins that can read inputs and turn them into outputs which can result in the activation of motors, turning on LEDs and controlling other various components. (Arduino - Introduction, 2021)

I choose the Arduino microcontroller to act as the main household of this project for several reasons which benefited me. Those reasons are due to its open software and hardware, having a huge community, which allows for easy learning of the software and having low-cost boards. Below shows a picture of the Arduino 2650 Mega microcontroller that is used within this project.

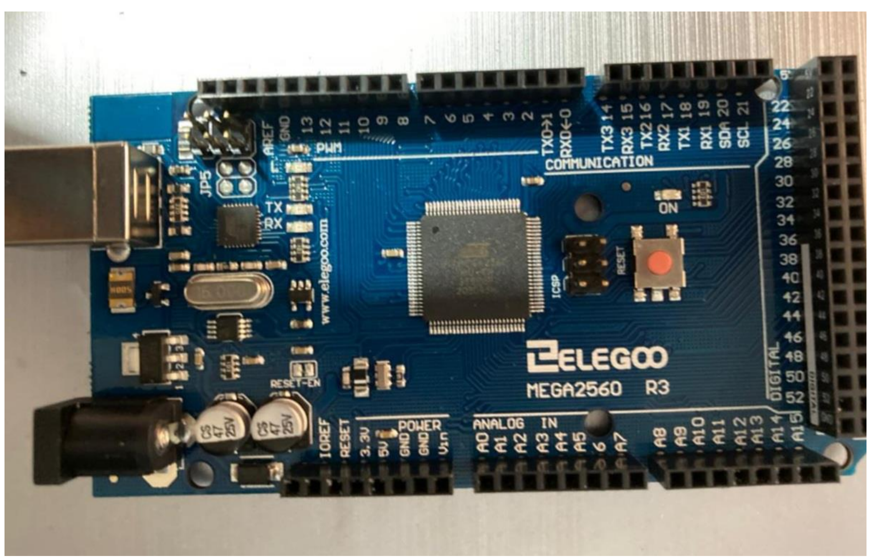


Figure 1 Picture of Arduino 2650 Mega microcontroller used within this project.

In this project the Arduino 2650 Mega microcontroller is used as the main unit or household as mentioned above. This meaning, the required components will be connected to the input / output of the analog / digital pins, together with the code constructed from the program IDE software produce the desired output. This being the automated irrigation system. The Arduino bord will also be accompanied by the breadboard.

IDE SOFTWARE –

My reason for choosing this software is because the community-driven framework is the best thing about using Arduino IDE. It is easier to find inventions from seasoned programmers because there's a large user base. You will also get troubleshooting advice and suggestions on your coding abilities.

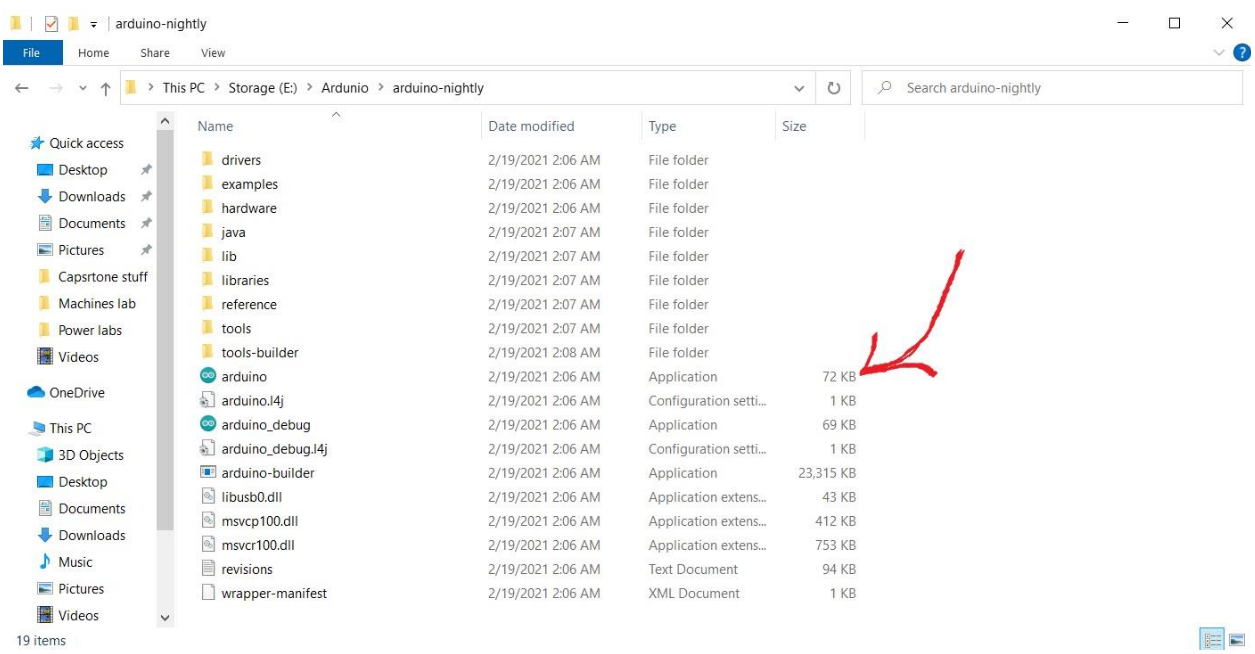


Figure 2 Showing the IDE Software

Breadboard

A Breadboard is A thin rectangular plastic board that is used to hold electronic components such as transistors, resistors, chips that are wired together. It is commonly used to develop prototypes of electronic circuits and can be reused after being used within the span of the future.

The breadboard is made of three section: which are two sets of very long power rails and a large middle section that is full of 5 hole long terminal strips. It can be shown that there is + and – shown where the power rails are and a – j letters shown within the 5-hole terminal strips. Below shows a Picture of the breadboard which will be used within this project. (*Breadboards for Beginners*, 2016)

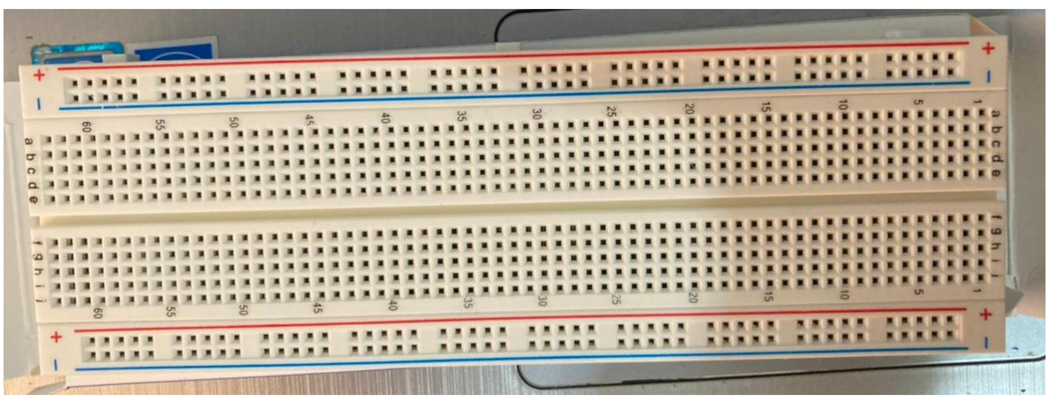


Figure 3 Picture of Breadboard used within this project.

In this project the breadboard is used with accordance of the Arduino microcontroller to create a home irrigation system. The breadboard acts as a component holder that will be used to connect multiple live, ground and items that are used within this project. One such component is the Soil moisture Sensor.

Soil Moisture Sensor

The soil moisture sensor that is used in this project is the FC- 28 soil moisture sensor which is connected to the Arduino microcontroller. This sensor measures the volumetric content of water in soil and gives the moisture level. The moisture level can range from 0 – 1023 and can give both analog and digital output signals. Below shows a figure of a labelled picture of soil moisture sensor. (*How to Test Soil With Arduino and an FC-28 Moisture Sensor*, 2017)

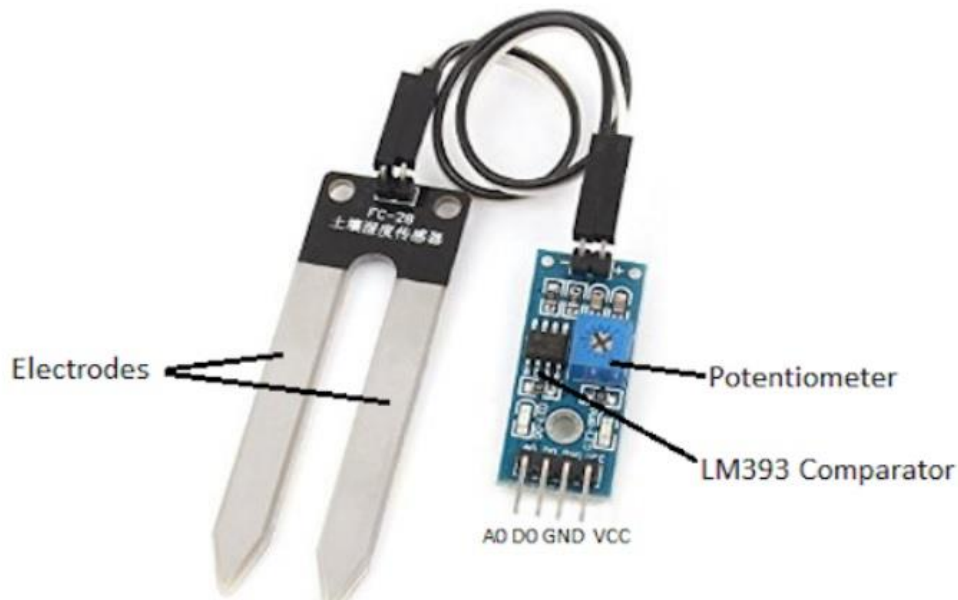


Figure 4 labelled Picture of Soil Moisture Sensor. (*How to Test Soil With Arduino and an FC-28 Moisture Sensor*, 2017)

The soil moisture sensor consists of two probes that are used to measure the volumetric content of water. The two probes allow the current to pass through the soil, which gives the resistance value to measure the moisture value. When there is water, the soil will conduct more electricity, which means that there will be less resistance whereas dry soil conducts electricity poorly. This results in the soil conducting less electricity, which means there will be more resistance. Below shows the Specifications and Pin-Out details of the soil moisture sensor.

Specification:

Input Voltage	3.3–5V
Output Voltage	0–4.2V
Input Current	35mA
Output Signal	both analog and digital

Table 1 showing Specification of Soil Moisture Sensor

Pin- Out Details:

VCC	Power
A0	Analog Output
D0	Digital Output
GND	Ground

Table 2 showing Pin- Outs of Soil Moisture Sensor

Water Level Sensor

The water level sensor which is being used within this project is a sensor that can be used to detect leakage, measure water level, track a sump pit, and detect rainfall. There are ten exposed copper trades on the sensor, five of which are power traces and five of which are sense traces.

These traces are interlaced such for every two power traces there is one sense trace. When submerged, these traces are usually not connected but are bridged by water. Below shows a diagram of the water level sensor being used.

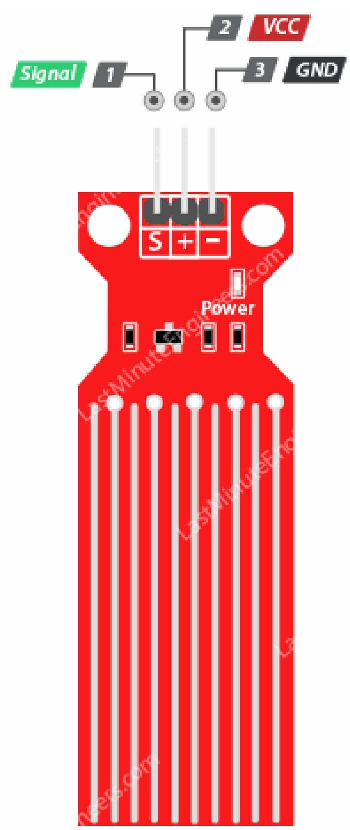


Figure 5 labelled Picture of Water level Sensor. (Engineers, 2020)

The water level sensor works as follows:

The series of exposed parallel conductors' functions as a variable resistor, with resistance that varies with water level. The change in resistance is proportional to the distance between the sensor's top and the water's surface. The resistance is proportional to the water's height in reciprocal proportion. The more water the sensor is submerged in, the higher the conductivity and the lower the resistance. The less water the sensor is submerged in, the worse its conductivity will be and the higher its resistance will be.

The sensor generates an output voltage based on the resistance, which can be used to assess the water level by measuring it. (Engineers, 2020)

16 x 2 LCD display

The LCD (Liquid Crystal Display) is a type of flat panel display that operates primarily with liquid crystals. An LCD (Liquid Crystal Display) screen is a type of electronic display that can be used in a variety of ways. (Contributor, 2019) A 16x2 LCD display is a very basic module that can be found in a variety of devices and circuits. This component can display 16 characters per line on each of its two lines. Each character is displayed in a 5x7 pixel matrix. The 224 different characters and symbols can be displayed on the 16 x 2 intelligent alphanumeric dot matrix display. Command and Data are the two registers on this LCD.

Various commands given to the display are stored in the command register while the data register holds the information that will be displayed. Below shows A diagram of the LCD display being used within this project. (Thingbits Electronics, 2021)

Figure 6 Picture of the 16 x 2 LCD Display.

In this project the 16 x 2 LCD display is used to showcase the operation of the irrigation system. Such that the water level status, moisture sensor level, pump activation etc will be shown onto this screen using the Command and Data registers. I choose this component to display my processes because you can easily interface this with an Arduino and that it is commonly used to display data in devices. Below shows the details of the LCD terminals.

Terminal 1	GND
Terminal 2	+5V
Terminal 3	Mid terminal of potentiometer (for brightness control)
Terminal 4	Register Select (RS)
Terminal 5	Read/Write (RW)
Terminal 6	Enable (EN)
Terminal 7	DB0
Terminal 8	DB1
Terminal 9	DB2
Terminal 10	DB3
Terminal 11	DB4
Terminal 12	DB5
Terminal 13	DB6
Terminal 14	DB7
Terminal 15	+4.2-5V
Terminal 16	GND

Table 3 showing LCD terminals (*Interface an LCD with an Arduino - Projects*, 2018)

Potentiometer (10kohm)

A potentiometer is a simple knob that provides variable resistance, which can be read as an analog value on the Arduino board. (*Arduino - Potentiometer,* 2021) In this project the Potentiometer is used as a variable resistance with accordance to the LCD display. This component allows the user to ability to control the contrast of light shown, making it dimmer or brighter.

Led

A light-emitting diode (LED) is a semiconductor device that emits light when an electric current is passed through it in its most basic form. When the current-carrying particles (known as electrons and holes) collide inside the semiconductor material, light is emitted. (*What Is a Led?*, 2004) Below shows A diagram of a led.

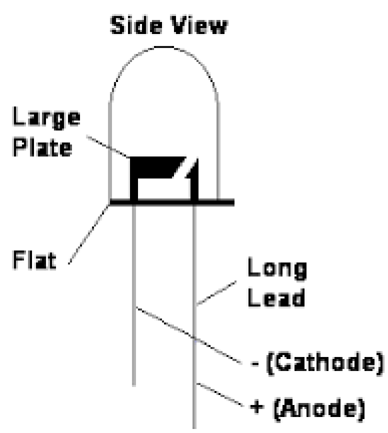


Figure 7 Labelled diagram of A led. (*Electronics 2000 | Pin-Outs | LEDs,* 2018)

Within the practical operation of a led the positive pin of the LED is connected to the battery's positive terminal, and the negative pin is connected to a resistor that is connected to the battery's negative terminal. The current that passes through the LED causes it to light up is supplied by the battery.

In this project I intent to use the led to indicate when the water is low, and the pump is on. I choose the led because as stated above it is used to produce light, which depends on which led is used different outcomes will occur. This being an effective way of showing what is being intended.

Buzzer

An active buzzer is a device or component that can generate a tone using an internal oscillator by using a small dc voltage where a passive buzzer requires an AC signal to make a sound. Below shows A diagram of Both an active and Passive Buzzer. (*Tutorial: Active Buzzer: ARTC 4330/5330*, 2017)

Figure 8 Picture of Passive and Active Buzzer (*Tutorial: Active Buzzer: ARTC 4330/5330*, 2017)

In this project I choose the buzzer because of its ability to make sounds or tone as indicated above and to illustrate when the water level is low within the tank.

5v Relay

A 5V relay is an electronic switch that uses a low-current signal to power a high-current circuit. The input voltage range for 5V relay signals is 0-5V. The device is operated by VCC. In the power supply, there is a JD-VCC relay. Below shows A labelled diagram of A 5V relay module. (Others, 2021)

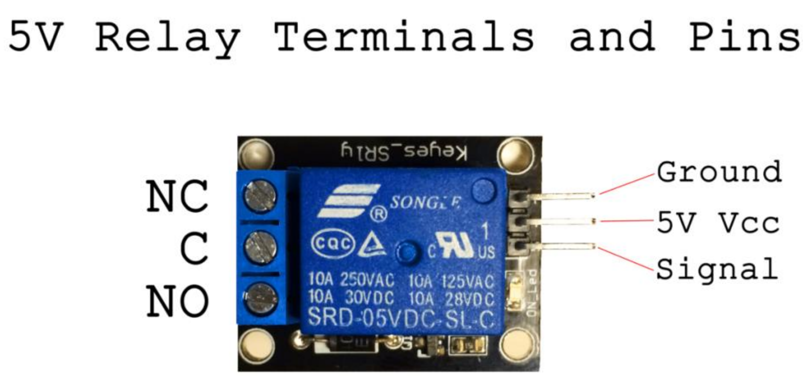


Figure 9 Labelled diagram of A 5V Relay module (*How the 5v Relay Works ??*, 2018)

Below shows A table of the Pins and Pin meaning

Pin	Pin Definitions
NC	Normally closed 120-240V terminal
NO	Normally open 120-240V terminal
C	Common terminal
Ground	Connects to the ground pin on the Arduino
5V Vcc	Connects the Arduino's 5V pin
Signal	Carries the trigger signal from the Arduino that activates the relay

Table 4 showing Pin and Pin Definitions of a 5V relay module

The 5v relay works as explained: A 120-240V switch is connected to an electromagnet within the relay. The electromagnet becomes charged and pushes the contacts of the switch open or closed when the relay receives a HIGH signal at the signal pin.

Inside the relay are two types of electrical contacts: normally open (NO) and normally closed (NC) (NC). If you want the 5V signal to turn the switch on or off will determine which one you use. In both configurations, the 120-240V supply current enters the relay via the common (C) terminal. (*How the 5v Relay Works ??*, 2018)

Normally Open

When the relay receives a HIGH signal, the 120-240V switch closes, allowing current to flow from the C terminal to the NO terminal in the usually open configuration. The relay is deactivated, and the current is stopped when the signal is LOW.

Figure 10 Labelled diagram showing Normally Open. (*How the 5v Relay Works ??*, 2018)

Normally Closed

A HIGH signal opens the switch and interrupts the 120-240V current in the usually closed configuration. The switch is closed when the signal is LOW, allowing current to flow from the C terminal to the NC terminal.

Figure 11 Labelled diagram showing Normally Closed. (*How the 5v Relay Works ??*, 2018)

In this project a 4 channel 5-volt relay module will be used to step down the voltage of a water pump hence allowing it to be used as Normally Open or Normally Closed as stated above to acquire its output task.

Water Pump

Figure 12 diagram showing Water Pump.

Water pump specifications:

PULACO
Pl-128
Submersible pump
AC 110 – 120 V/ 60Hz 5W
Flow: 95GPH Head:3.0FT

Table 5 showing Water Pump specifications

Water pumps are devices that transfer water from one location to another. They are important in agriculture because they transport water from the source to the fields and crops. Water pumps can be used for a variety of irrigation systems, including drip irrigation, sprinklers, and hose irrigation.

In this project the water pump is used to transfer water from the source to the plant/s chosen. I choose this component because of its applications that ranged from various topics contributing to irrigation systems.

Effect of Arduino Micro-controller in irrigation

Implementing the Arduino micro-controller in irrigation has many benefits such that it can affect the entire growth process positively. This was stated within the literature review.

Modelling of the Automatic Irrigation System

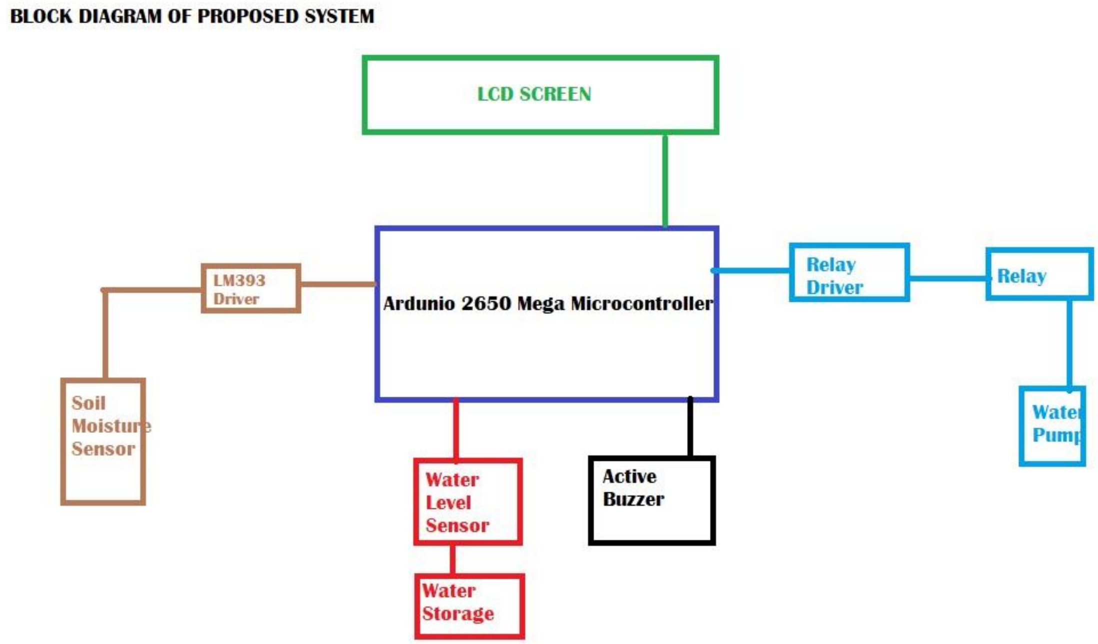


Figure 13 Block Diagram of proposed system.

Expected Operation of Project

Using the Arduino as the central core, the proposed device is designed to automatically water the plants when the soil moisture sensor detects that the soil is deficient in water when there is sufficient water. A soil moisture sensor, an LCD monitor to show the moisture percentage and pump status, a relay module to control the on and off switch of the water pump, a water pump, a water level sensor and active buzzer make up the automatic irrigation system. When the soil moisture sensor detects dry soil, the given moisture value is displayed on the LCD monitor. The relay module will automatically turn on the water pump to begin the watering operation, or vice versa. You will be alerted with a sound if the water is unavailable. When the soil moisture sensor detects soggy or damp conditions, the water pump is automatically turned off. Hardware testing is carried out to ensure that the proposed device is fully functional within the parameters of the Results and discussion below.

Results and Discussion

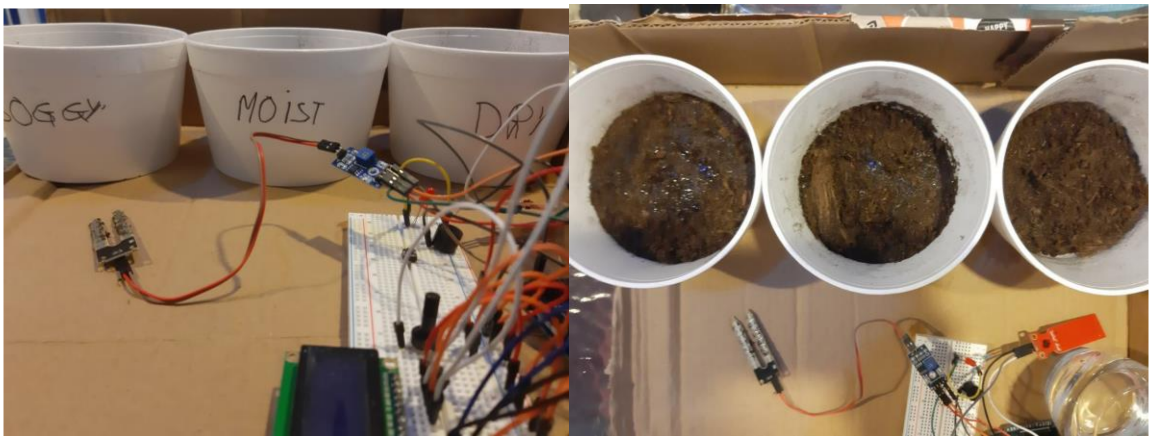


Figure 14 model of where it can be implemented.

Figure 14 shows the model of where the project test had been recorded. The test conducted and recorded where that of Soil and Water level test. As shown above there are 3 cups labelled soggy, moist and dry which all containing soil with different soil properties. Below shows the test conducted by both the soil moisture sensor and water level sensor.

The following steps can be followed to test the project.

- **Connect the Arduino to a 5V power source via USB or an external power supply.**
- **The moisture sensor should then be buried in the dirt near the roots.**
- **In the water container/tank, position the water level sensor.**
- **Switch on the mains by connecting the water pump to the Relay (N/O and Common terminals).**

Soil Moisture Sensor

Dry Soil Test

Figure 15 Showing the practical operation of Dry Soil Test.

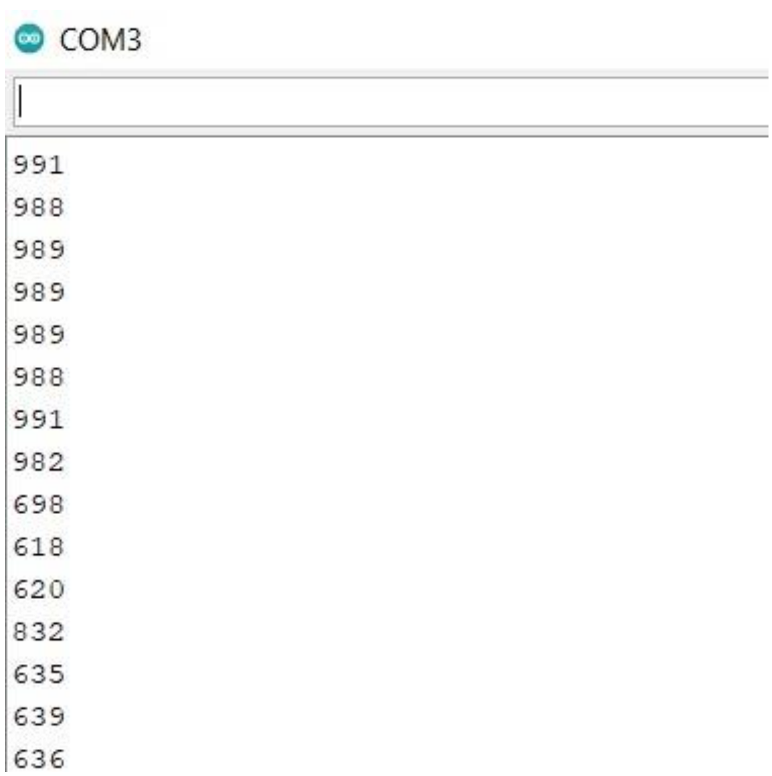


Figure 16 Showing the serial display for Dry Soil Test.

During the Dry soil test, it can be observed that it worked as expected as shown within Figures 15-16. It can be noted that the test was done when the water was low. Figure 16 shows the serial display for the dry soil test. It can be observed that when the soil moisture sensor raw value is above or equal to 600 it will send a signal to the LCD display stating that it is dry soil. Whereas, if the project had been connected to water will be "HIGH" and the pump will be "ON". For further clarification refer to APENDIX I where the code is located.

Moist Soil Test

Figure 17 Showing the practical operation of Moist Soil Test.

Figure 18 Showing the serial display for Moist Soil Test.

During the Moist soil test, it can be observed that it worked as expected as shown within Figures 17-18. It can be noted that the test was done when the water was low. Figure 18 shows the serial display for the Moist soil test. It can be observed that when the soil moisture sensor raw value is above or equal to 500 but not over 599 it will send a signal to the LCD display stating that it is Moist soil. Whereas, if the project had been connected to water, the water will be "LOW" and the pump will be "OFF". This meaning whenever the soil reached a certain serial value it will automatically turn of the water being poured within the plant. For further clarification refer to APENDIX I where the code is located.

Soggy Soil Test

Figure 19 Showing the practical operation of Soggy Soil Test.

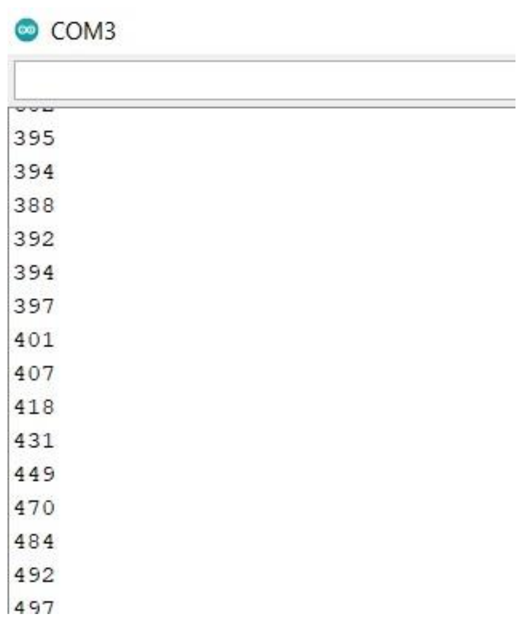


Figure 20 Showing the serial display for Soggy Soil Test.

During the Soggy soil test, it can be observed that it worked as expected as shown within Figures 19-20. It can be noted that the test was done when the water was low. Figure 20 shows the serial display for the Soggy soil test. It can be observed that when the soil moisture sensor raw value is below or equal to 500, A signal will be sent to the LCD display stating that it is Soggy soil. Alike both tests stated above where the project is connected to water or has an availability of water a "LOW" Signal will be sent to the pump. The LCD display will display "PUMP OFF" as shown within figure 19. This also further means, whenever the soil reaches a certain serial value, it will automatically turn of the water being poured within the plant. For further clarification refer to APENDIX I where the code is located.

Water Level Sensor

Water level sensor Test

```
COM3

PIN Pin Error level is 026% (raw value is 0091)
945
PIN Pin Error level is 044% (raw value is 0154)
952
PIN Pin Error level is 070% (raw value is 0247)
954
PIN Pin Error level is 080% (raw value is 0281)
955
PIN Pin Error level is 077% (raw value is 0270)
945
PIN Pin Error level is 033% (raw value is 0116)
944
935
PIN Pin Error level is 029% (raw value is 0103)
930
PIN Pin Error level is 004% (raw value is 0017)
926
940
PIN Pin Error level is 000% (raw value is 0000)
942
PIN Pin Error level is 004% (raw value is 0014)
```

Figure 21 Shows Water level status on serial display.

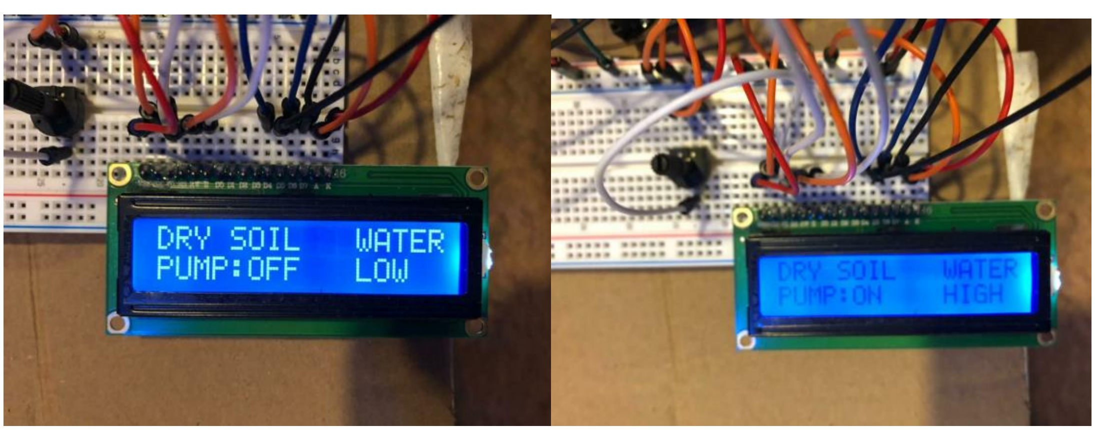
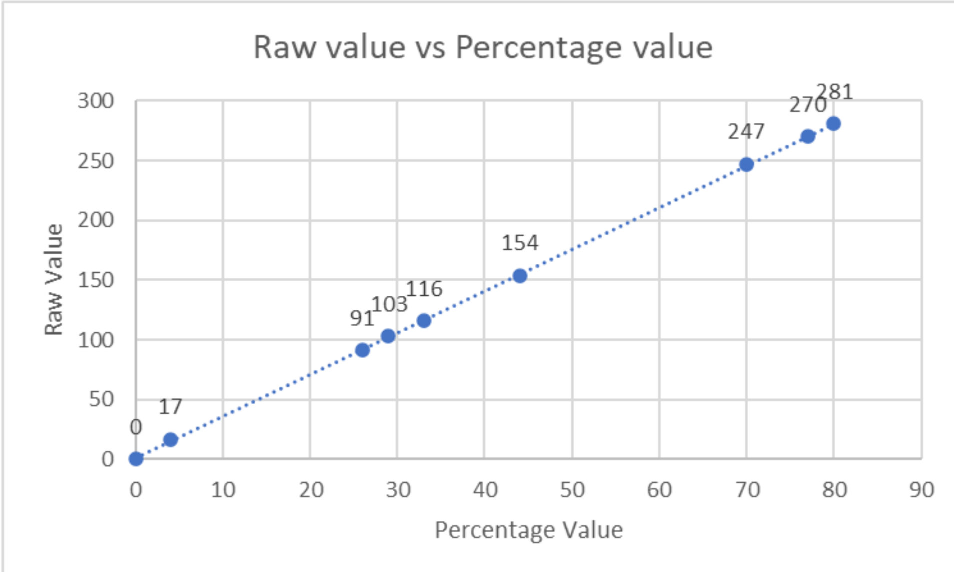


Figure 22 Shows Water level status on LCD display.

Figure 21 shows the serial display for the water level sensor test. It can be observed that the value was recorded both in its raw value and the tank's corresponding percentage value. Ongoing this test, there were a few problems that arise but was delt with without hindering its operation. This will be discussed within the next subheading. The aftermath result resulted in the expected outcome I desired as shown within figure 22. This figure shows the water level status shown on the LCD display stating, "WATER HIGH" and "WATER LOW" together with showing the water pumps active operation "ON" or "OFF".



Percentage Value	Raw value
26	91
44	154
70	247
80	281
77	270
33	116
29	103
4	17
0	0

Figure 23 Shows Graph of Raw value vs Percentage Value.

Figure 23 shows A graph of the Raw value vs the Percentage value obtained from the water level sensor test results. As observed, it can be noted that the water level within the tank or container used was 80% full at a raw value of 281. The graph can further show the corresponding values and the percentage value in an ascending order. When the water level hits 0 Raw value a signal will be sent to an active buzzer which indicates the person that there is no water in the tank.

Challenges Occurred Containing the Water Level Test.

<u>Water level sensor Acting as a potentiometer.</u>

On building the circuit and connecting the water level sensor unto the breadboard. It was observed that the LCD display begun to dim as the function of the potentiometer referring to the mythology of this project. After multiple research done, the issue was fixed by changing the input and output pins of the water level sensor onto a separate input and output of the Arduino mega 2560 microcontroller. The cause of problem was that the water level sensor had been taking up most of the voltage allocated from the 5 Volts.

<u>Water level sensor only reading one value at a time with a delay.</u>

On testing the water level sensor, it was observed that the readings obtained showed the sensor only reading one vale at a time with a long delay if the sensor is not moved. This caused a problem as each time the sensor did not receive a value the buzzer will go off stating that it needs more water. I fixed this issue by changing the code of the project, which allows a delay of 10 seconds whenever the value is read. However, if the moisture value exceeds a specific rang the pump will automatically shot off. Refer to Appendix I for code.

Water Pump

Water Pump Test

Figure 24 Showing Water pump test.

Above shows Figure 24 showing the water pump test. It can be observed that the water pump works as expected. As shown within the moisture and water level test the pump will be turned on when a "HIGH" signal is sent and shown onto the LCD display.

Cost analysis

The project which was attempted was An Automatic irrigation system for farmers or gardeners in Trinidad.

Project Component	Cost Price
The Most Complete Starter Kit Mega 2560 Project	$60 USD
4 channel 5V relay module	$8 USD
Pulaco Submersible Water Pump	$14 USD
Moisture level sensor	$8 USD
Total	$90 USD

Table 6 showing Cost of each component and total cost.

Conclusion

The Design of an automated irrigation system using an Arduino microcontroller that is capable of monitoring, testing and increasing the productivity of crops will little overhead was a success. Referring to the results and discussion of this report it can be shown that the moisture sensor worked as expected, allowing the LCD display to display when it is Dry, soggy and moisture soil. However, the water level sensor had its draw backs but with improvisation both worked well together as explained within 'Challenges Occurred Containing the Water Level Test' subheading. It can be observed that the program used was not that of citect / app but was done on the Ide software where an LCD display was used to provide a visual display of the process/ processes as stated in the objectives of this report.

Constraints

The device worked as expected with a few compromissions, however the limitation of the project model is that it may not be able to monitor large areas of plants with the Soil moisture sensor being used. This can mean it is suitable for persons that has small gardens inside or outside of their homes.

The water level sensor being used only records one value and sends to the serial display. This can sometimes cause problems when it is left unattended.

Budget Constraints

Although due to the COVID-19 pandemic which resulted in the closure of the University of Trinidad and Tobago. The challenge that occurred was getting the components and budgeting the amount to be spent. As shown within the Cost analysis the total amount came up to 90 USD. I was lucky enough to had gotten free shipping which greatly played within the budget of this project, indicating that the project is not costly.

Suggestions for improvement and further work

Looking back onto the project report there can be many ways of improvement that can undertaken to improve this project. Such improvements can be that of, but not limited to, adding temperature sensors and a server motors. Implementing global trends in agriculture can greatly increase productivity. These such global trends are that artificial intelligence, blockchain, RFID and internet of things sensors. Internet of things stands out more to me as you can monitor the plants from anywhere in the world with the use of an app and internet connection. Some other improvements can be, Analysis of water mineral content and fertilizer concentration and controlling humidity and temperature in greenhouses.

Bibliography

A Low-Cost Information Monitoring System for Smart Farming Applications.,Sensors - X-MOL. (2020, April 22). A Low-Cost Information Monitoring System for Smart Farming Applications. https://www.x-mol.com/paper/1253455240163647488?recommendPaper=1254238557230227456

ACHPA: A sensor based system for automatic environmental control in hydroponics. (2020, September). Food Science and Technology. https://www.scielo.br/pdf/cta/v40n3/0101-2061-cta-fst13319.pdf

Agriculture. (2020, March 9). Oxford Business Group. https://oxfordbusinessgroup.com/trinidad-tobago-2020/agriculture

Arduino - Introduction. (2021, January). Arduino. https://www.arduino.cc/en/guide/introduction

Arduino - Potentiometer. (2021, January). Ardunio. https://www.arduino.cc/en/tutorial/potentiometer

Arduino Tutorial - Lesson 3 - Breadboards and LEDs. (2012, April). Ladyada. https://www.ladyada.net/learn/arduino/lesson3.html#:%7E:text=The%20positive%20pin%20of%20the,LED%2C%20making%20it%20light%20up.

Breadboards for Beginners. (2016, September 7). Adafruit Learning System. https://learn.adafruit.com/breadboards-for-beginners?view=all

Contributor, T. (2019, September 13). ***LCD (Liquid Crystal Display)*****. WhatIs.Com. https://whatis.techtarget.com/definition/LCD-liquid-crystal-display**

Electronics 2000 | Pin-outs | LEDs**. (2018, January). Electronics 2000. https://www.electronics2000.co.uk/pin-out/led.php**

Engineers, L. M. (2020, December 18). ***How Water Level Sensor Works and Interface it with Arduino*****. Last Minute Engineers. https://lastminuteengineers.com/water-level-sensor-arduino-tutorial/**

How the 5v relay works ?? **(2018, July 25). IOT - Building Reliability. https://mounishkokkula.wordpress.com/how-the-5v-relay-works/**

How to Test Soil With Arduino and an FC-28 Moisture Sensor**. (2017, March 23). Maker Pro. https://maker.pro/arduino/projects/arduino-soil-moisture-sensor#:%7E:text=How%20Does%20the%20Arduino%20Soil,to%20measure%20the%20moisture%20value.**

Interface an LCD with an Arduino - Projects**. (2018, January 4). All About Circuits. https://www.allaboutcircuits.com/projects/interface-an-lcd-with-an-arduino/#:%7E:text=a%20user%20interface.-,You%20can%20easily%20interface%20a%20liquid%20crystal%20display%20(LCD)%20with,and%20many%20other%20electronic%20devices**

Newsday. (2020, April 13). ***Farming problems*****. Trinidad and Tobago Newsday. https://newsday.co.tt/2020/04/13/farming-problems/**

Others, R. T. A. (2021, January). *2-Channel 5V Relay Module(Black)*. RhydoLABZ.Com. https://www.rhydolabz.com/hardwares-relays-c-170_182/2channel-5v-relay-moduleblack-p-2580.html#:%7E:text=It%20is%20frequently%20used%20in,relay%20in%20the%20power%20supply.

***Providing Water to the Plants Automatically Using Microcontroller*. (2018, January). Edelweiss Applied Science and Technology. http://edelweisspublications.com/articles/44/157/Providing-Water-Plants-Automatically-Using-Microcontroller**

S. (2020, December 15). *Humidity Sensor: Basics, Usage, Parameters and Applications*. Electronics For You. https://www.electronicsforu.com/electronics-projects/humidity-sensor-basic-usage-parameter#:%7E:text=A%20humidity%20sensor%20(or%20hygrometer,factor%20when%20looking%20for%20comfort.

Thingbits Electronics. (2021, January). *Standard LCD 16x2 Display*. https://www.thingbits.in/products/standard-lcd-16x2-display

***Trinidad and Tobago prioritises agriculture, crop diversification and food security*. (2019, November 13). Oxford Business Group. https://oxfordbusinessgroup.com/overview/sustaining-progress-food-security-and-diversification-are-priorities-long-underfunded-sector#:%7E:text=This%20looks%20to%20be%20a,and%20consumption%20of%20local%20produce.**

***Tutorial: Active Buzzer: ARTC 4330/5330*. (2017, June). Bgsu Instructure. https://bgsu.instructure.com/courses/1157282/pages/tutorial-active-buzzer#:%7E:text=Active%20Buzzer%20is%20the%20slightly,will%20make%20sounds%20when%20electrified**

***What is a Led?* (2004, September). Ledsmagazine. https://www.ledsmagazine.com/leds-ssl-design/materials/article/16701292/what-is-an-led**

S.R kumbhar, M. S gaikwad and M. D up-lane, Multi-terminal distributed control system, raman memorial conf., Nov 1997, Pune.(india) pp 5.

Proceeding of the international multiconference of engineers and computer scientist 2013 volume II, March 13-15,2013,Hong Kong

Internation journal of scientific research and management studies Volume 2 issue 1, pg 21-30

Appendix I – Code for Project

Steps to take when uploading code.

- **Connect your machine to the Arduino.**
- **Open the code that is attached to this message.**
- **Choose your COM Port and Arduino Board from the Tools menu.**
- **Select the Upload Button.**

```
#include <LiquidCrystal.h> //LCD Library

#define NOTE_C4 262
#define NOTE_D4 294
#define NOTE_E4 330
#define NOTE_F4 349
#define NOTE_G4 392
#define NOTE_A4 440
#define NOTE_B4 494
#define NOTE_C5 523

int temp;
//int T_Sensor = A4;
int M_Sensor = A0;
int W_led = 7;
const int W_Sensor = A2;
bool tapWater = true;
int HistoryValue = 0;
char printBuffer[45];
int P_led = 13;
int Speaker = 9;
int val;
int cel;
String analogPin;

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
```

Figure 25 Code for Project

```
void setup()
 {
    lcd.begin(16, 2);
    lcd.clear();
    pinMode(13,OUTPUT);
    pinMode(A2,INPUT);
    pinMode(9,OUTPUT);
    pinMode(A0,INPUT);
    Serial.begin(9600);
    analogPin = analogPinConverter(W_Sensor);
    /*
    val = analogRead(T_Sensor); //Read Temperature sensor value

    int mv = ( val/1024.0)*5000;
    cel = mv/10;
    */
    lcd.setCursor(0,1);
    lcd.print("KYLE'S");
    delay(2000);
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("AUTOMATIC");
    lcd.setCursor(0,1);
    lcd.print("IRIGATION SYSTEM");
    delay(2500);
    lcd.clear();


    delay(1000);
 }
```

Figure 26 Code for Project

```
}

int levelConverter(int sensorValue, bool isCleanWater = true){
  int percentage;
  static int maxValue = 350;

  if (isCleanWater){
    percentage = map(sensorValue, 0, maxValue, 0, 100);
    if (percentage > 100) {
      maxValue = sensorValue;
      percentage = 100;
    }
  }else{
    percentage = map(sensorValue, 0, 1023, 0, 100);
  }
  return percentage;
}

void loop()
{

  lcd.clear();
    /* val = analogRead(T_Sensor); //Read Temperature sensor value
    Serial.println(val);
    int mv = ( val/1024.0)*5000;
     cel = mv/10;
  */
  int Moisture = analogRead(M_Sensor); //Read Moisture Sensor Value
  Serial.println(Moisture);
/*
```

Figure 27 Code for Project

```
if (Moisture> 600)   // for dry soil
 {
       lcd.setCursor(0,0);
       lcd.print("DRY SOIL");
       //lcd.setCursor(11,1);
       //lcd.print("SOIL");
 int currentValue = analogRead(W_Sensor);    // get sensorPin analog value
 int diffValue = HistoryValue - currentValue;
 if((abs(diffValue) >= 10)||((currentValue == 1)&&(HistoryValue != 1)))

      {
        int percentValue = levelConverter(currentValue, tapWater);
        sprintf(printBuffer,"PIN %s level is %03d%% (raw value is %04d)\n", analogPin.c_str(), percentValue, currentValue);
        Serial.print(printBuffer);
        HistoryValue = currentValue;
        digitalWrite(13, HIGH);
        digitalWrite(W_Sensor, HIGH);
        digitalWrite(7, LOW);
        lcd.setCursor(0,1);
        lcd.print("PUMP:ON");
        lcd.setCursor(11,0);
       lcd.print("WATER");
       lcd.setCursor(11,1);
       lcd.print("HIGH");
       delay(10000);

      }
```

Figure 28 Code for Project

```
   else
       {
         digitalWrite(13, LOW);
         digitalWrite(W_Sensor, LOW);
         digitalWrite(7, HIGH);
         lcd.setCursor(0,1);
         lcd.print("PUMP:OFF");
         lcd.setCursor(11,0);
        lcd.print("WATER");
        lcd.setCursor(11,1);
        lcd.print("LOW");


           tone(Speaker, NOTE_C4, 500);
           delay(500);
           tone(Speaker, NOTE_D4, 500);
           delay(500);
           tone(Speaker, NOTE_E4, 500);
           delay(500);
           tone(Speaker, NOTE_F4, 500);
           delay(500);
           tone(Speaker, NOTE_G4, 500);
           delay(500);
       }
    }
```

Figure 29 Code for Project

```
    }

     if (Moisture>= 70 && Moisture<=600) //for Moist Soil
    {
      lcd.setCursor(0,0);
     lcd.print("MOIST SOIL");
     //lcd.setCursor(11,1);
     //lcd.print("SOIL");
     digitalWrite(13,LOW);
     lcd.setCursor(0,1);
     lcd.print("PUMP:OFF");
  }

  if (Moisture < 500)  // For Soggy soil
  {
    lcd.setCursor(0,0);
     lcd.print("SOGGY SOIL");
     //lcd.setCursor(11,1);
     //lcd.print("SOIL");
     digitalWrite(13,LOW);
     lcd.setCursor(0,1);
     lcd.print("PUMP:OFF");
  }
delay(1000);
```

Figure 30 Code for Project

```
  }
 delay(1000);
}
String analogPinConverter(int value){
  String analogPinValue;

  switch(value){
    case 14:                                  // D14 = A0
      analogPinValue = "A0";
      break;
    case 15:                                  // D15 = A1
      analogPinValue = "A1";
      break;
    case 16:                                  // D16 = A2
      analogPinValue = "A2";
      break;
    case 17:                                  // D17 = A3
      analogPinValue = "A3";
      break;
    case 18:                                  // D18 = A4
      analogPinValue = "A4";
      break;
    case 19:                                  // D19 = A5
      analogPinValue = "A5";
      break;
    default:
      analogPinValue = "Pin Error";
      break;
  }
  return analogPinValue;

  }
```

Figure 31 Code for Project

Appendix II – Project Connection Details

ARDUINO PINS

0	**N/C**
1	**N/C**
2	**LCD-14**
3	**LCD-13**
4	**LCD-12**
5	**LCD-11**
6	**N/C**
7	**LM35_(TEMPERATURE_SENSOR)**
8	**N/C**
9	**Active Buzzer**
10	**N/C**
11	**LCD-6**
12	**LCD-4**
13	**PUMP_STATUS_AND_TO_RELAY**
A0	**SOIL_MOISTURE_SENSOR**
A2	**WATER_LEVEL_SENSOR**
LCD-1	**GND**
LCD-5	**GND**
LCD-2	**Vcc**

Table 7 shows Arduino pins and their respected connections

- **Note: Pin 7 is a red led which indicates the Temperature Sensor, it was not placed within the Block diagram as it was a last-minute additional feature.**

Circuit Diagram of Proposed Project

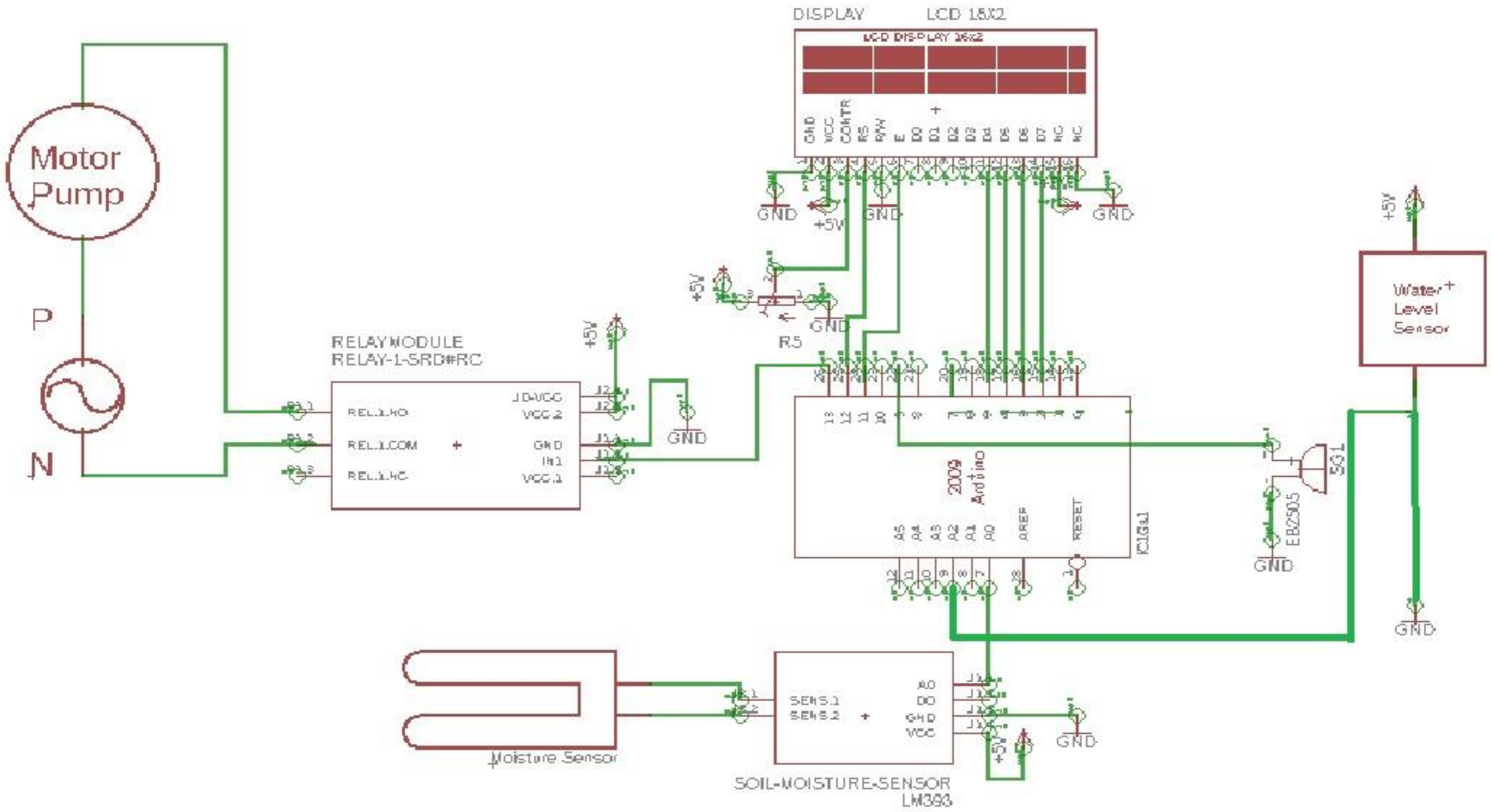


Figure 32 Circuit Diagram of Proposed Project

Appendix III – Gantt Chart

Task to be done	WEEK 1	WEEK 2	WEEK 3	WEEK 4	WEEK 5	WEEK 6	WEEK 7	WEEK 8	WEEK 9	WEEK 10	WEEK 11	WEEK 12	WEEK 13	WEEK 14
RESEARCH														
TITLE SELECTION / PROPOSAL														
LITERATURE REVIEW														
CONCEPTUAL STUDY														
IDEATION AND SKETCHING														
MODELING / DESINGING														
PROPOSAL DEFENSE														

Figure 33 Gantt Chart