IOT Based Agricultural System

Department of Electrical & Electronic Engineering (EEE)

Sonargaon University (SU)

Supervised By

Nurul Ambia Alaul

Lecturer

Department of EEE

Sonargaon University (SU)

Submitted By

STUDENT ID	STUDENT NAME	SECTION
EEE1803015004	Mohammad Zohurul Islam	15C
EEE1803015019	Mohammad Shahidul Islam	
EEE1803015058	Muhammad Bayezid Hasan	
EEE1803015060	Md. Shafiqul Islam Khan	
EEE1803015061	Md. Kauser Hossain	

Date of Submission:

Declaration

It is declared hereby that this thesis paper or any part of it has not been submitted to anywhere else for the award of any degree.

Md. Kauser Hossain

Muhammad Bayezid Hasan

Md. Shafiqul Islam Khan

Mohammad Shahidul Islam

Mohammad Zohurul Islam

Under Supervision of

Nurul Ambia Alaul
Lecturer
Department of EEE
Sonargaon University (SU)

<u>Certification</u>

This is to certify that this project entitled "**IOT Based Agricultural System**" is done by the following students under my direct supervision. This project work has been carried out by them in the laboratories of the Department of Electrical and Electronic Engineering under the Faculty of Engineering, Sonargaon University (SU) in partial fulfillment of the requirements for the degree of Bachelor of Science in Electrical and Electronic Engineering.

<u>Supervisor</u>

Nurul Ambia Alaul

Lecturer

Department of Electrical and Electronic Engineer (EEE)

Sonargaon University (SU)

ACKNOWLEDGEMENT

The report titled as on’’ **IOT Based Agricultural System**” has been prepared to fulfill the requirement of our practicum program. In the process of doing and preparing our practicum report, we would like to pay our gratitude to some persons for their enormous help and vast co-operation.

At first, we would like to show our gratitude to the University authority to permit us to do our practicum. Specially, we would like to thank to our honorable teacher **Nurul Ambia Alaul,** Lecturer, Department of Electrical & Electronics Engineering, Sonargaon University (SU), Dhaka, for his valuable and patient advice, sympathetic assistance, co-operation, contribution of new idea. Deep theoretical and hardware knowledge & keen interest of our supervisor in this field influenced us to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stage have made it possible to complete this project.

Finally, we would like to thanks again to the respected **Vice- Chancellor of SU, Professor Dr. Md. Abul Bashar** also thanks to **Head of Department of SU, Electrical & Electronics Engineering, Professor Dr. M. Bashir Uddin** because they are designated such an environment for learning through which we got the opportunity to acquire knowledge under B.Sc. in EEE program, and that will be very helpful for our prospective career.

We are, indeed, grateful to all those from whom we got sincere cooperation and help for the preparation of this report

ABSTRACT

Smart agriculture is an emerging concept, as IoT sensors are able to provide information about agriculture and then works based on user input.In this project, it is proposed to develop a smart agriculture system that is compatible with Node MCU, IoT and Arduino technologies, such as sensor networks.The goal of this project is to develop automation using automation, ie IoT The use of smart agriculture is one of the main reasons for monitoring the environmental conditions to improve the yield of efficient crops.Features of this project include the development of a system that uses temperature, humidity and soil moisture and turn on/off of the pump motor through sensors using Node MCU boards and in case of any error, application notification has been made for the farmer's smartphone using internet. The system also has a dual communication link based on a cellular Internet interface that allows data inspection via an Android application and Allows irrigation scheduling to be programmed.Due to the autonomy of energy and the low cost, the system can be effective in limited areas of water in geographically isolated regions.

CHAPTER 1: INTRODUCTION

CHAPTER 2: THEORY OF THE PROJECT

CHAPTER 3: DESIGN AND FABRICATION

CHAPTER 4: RESULT & DISCUSSION

CHAPTER 5: CONCLUSION

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Topics	ABBREVIATIONS
DC	Direct Current
AC	Alternating Current.
LCD	Liquid Crystal Display
ADC	Analog-to-Digital Converter
IR	Infrared Receiver
IC	Integrated Circuit
LED	Light Emitting Diode
PCB	Printed Circuit Board
NodeMCU	Node MicroController Unit
IOT	Internet of things
DHT	Dehydrogenation Heat Treatment
RELAY	Regulating Emotions and Looking After Yourself
ESP	Enhanced Synchronization Protocol
IDE	Integrated Development Environment
LM	linear monolithic
VSS	Voltage Source Source
VDD	Voltage Drain Drain
VO	Voice Over
RS	Register selects
R/W	Read/Write

CHAPTER : 1

INTRODUCTION

1.1 Introduction

Despite the assumption that people may have an idea about the agricultural process, the reality is that today's agriculture is more data-centric, precise and smarter than ever. The rapid rise of Internet-of-Things (IoT) based technology has redesigned almost every industry, including " IOT Based Agricultural System," which has moved the industry from statistics to quantitative methods. These national revolutionary changes are shaking up existing agriculture and creating new opportunities along with various challenges. This article addresses the challenges that wireless technology and IoT have in agriculture as well as the challenges faced when integrating this technology with technology-driven farming. IoT devices and communication technologies related to wireless sensors that were connected to farming fields are analyzed in detail.We have two systems here to control the motor .In this project we will monitor the weather of agriculture, such as the temperature and humidity of the environment and soil moisture. Even turn on/off of the pump motor, through sensors using Node MCU boards and in case of any error, application notification has been made for the farmer's smartphone using internet. In addition, the use of unmanned aerial vehicles for other favorable applications such as crop surveillance and crop yield optimization has been considered in this article. Advanced IoT-based architectures and platforms used in agriculture have been highlighted where appropriate. Finally, based on this thorough review, we identify current and future trends in IoT in agriculture and highlight potential research challenges. We can see all the project data in an LCD display. We also monitor all data via the Internet and we can control irrigation water from any place. We used Node MCU as the controller and we created the C language as a programming language and uploaded it to the controller.

1.2 IoT System

The Internet of Things (IoT) is an interconnected computing device, mechanical and digital machine, object, animal or human being that is provided with a unique identifier and gives people the ability to transfer data from network to network without need or communication from computer to human. The definition of the Internet of Things has evolved due to the integration of multiple technologies, real-time analytics, machine learning, product sensors and embedded systems.

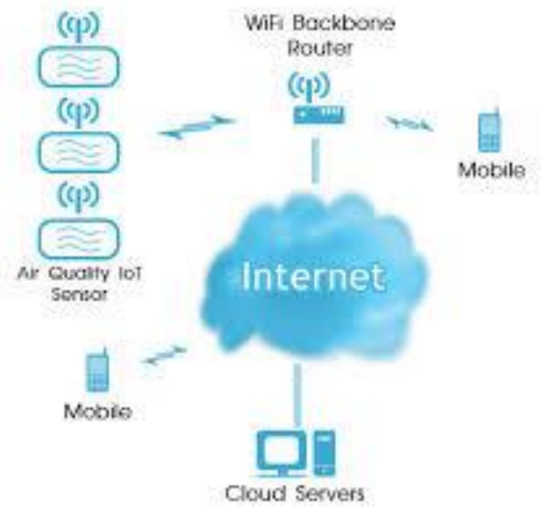

Fig-1.1: IoT System

The embedded systems, wireless sensor networks, control systems, automation (including home and building automation) and others all contribute to enabling the Internet of Things. In the consumer market, IoT technology is most synonymous with products related to the "agricultural system" concept, covering devices and applications (such as lighting, thermostats, home security systems and cameras and other home appliances) that support one or more common ecosystems, and smarter. Device ecosystems like phones and smart speakers Ira can be controlled through.

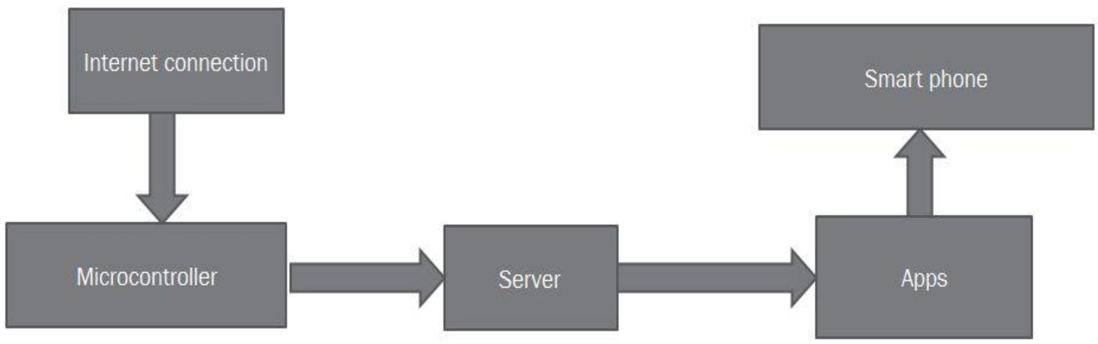

Fig- 1.2: Block Diagram of IoT

There are several serious concerns about the dangers of IoT growth, especially in terms of privacy and security; And as a result, industry and government initiatives begin to address them.

1.3 Proposed Methodology

The proposed system will be very advanced for centralized surveillance and control of agricultural land. It can be managed and worked wirelessly from any location using a mobile device. Application users can automatically control the basic activities of environmental, soil and rain irrigation data collection, and automatically take actions taken by the system. The approach that uses the app is to look at the effectiveness of the crop for a particular farm, calculate the crop forecast and personalize the crop recommendations. The main block of the proposed system is shown in fig. 1 is the microcontroller. It is battery powered, secure and low power port for fast connection Variability in environmental conditions will affect the overall yield of the crop. Plants require very specific conditions for optimal development and health. It is very important to monitor the condition of the crop so sensors are used. Temperature infrared thermopile sensors are used; It is built on digital control and math engines. It senses temperature values in real time, and the humidity sensor tracks the relative humidity of the air within the agriculture sector.

First, the microcontroller will test three conditions:

I. Availability of uninterrupted power supply.
II. The moisture level in the soil.
III. The temperature level in the Weather.

1.4 Future Scope of This Study

The proposed framework will revolutionize the traditional agriculture system used by majority of the farmers. This will allow the younger generation to do farming without previous experience and with ease. The system if implemented in maximum fields, will generate vast amount of data that can be used for data mining and further research in this field. This framework will conserve all the major resources by utilizing them more efficiently. This system requires minimum human interaction at present, which can be further reduced, and can even completely be eliminated by further improving the framework. There are many machine learning algorithms for image classification. This framework had only used SVM based classifier, hence other machine learning algorithms can be implemented and tested to further improve the system.

1.5 Objective of this Work

We have created a modern project called **IOT Based Agricultural System**. In this Project we combine transformer, rectifier capacitor, buck module. We have created a dc power supply to provide power to the project controller. In the project we use temperature & humility sensor and Soil Sensor, through programming we have shown the dirt, temperature, humidity, soil of the land. Besides, we have used finger print sensor to control the motor in the absence of internet.Only authorized persons can control the motor through finger print sensor.We have two systems here to control the motor.And see all the data in a display, And That data can be monitored from anywhere in the world through the Internet, using only one software. The above project circuit diagram is a mature circuit diagram that combines all instruments Are given and details of all the instruments are given in the project book.We have discussed result and discussion, cost estimation, future modification, conclusion.

CHAPTER : 2

THEORY OF THE PROJECT

2.1 Introduction

Traditional Agricultural weather stations lack self-sustainability, autonomous logging capabilities and the ability to transmit data wirelessly. Furthermore, professional temperature, humidity and soil data system monitoring systems and controls are very expensive for the average consumer and they have received limited transmission limits. Although some places are not easily accessible, it is not possible to monitor the agricultural weather parameters of all places, but advances in science and technology can provide the means of obtaining information about such places with wireless devices. Using IoT (Internet of Things) remote server system temperature, humidity, soil information monitoring system. We can monitor the agricultural weather conditions of the desired remote locations. Remote location data is made available to the user by sending standardized weather conditions applications using the IoT system. At any time, the user is able to transmit data to remote temp, gas, moisture monitoring system. The proposed system will use wireless communication technology to provide real-time access to weather conditions in remote locations. The main objective of this study is to design and apply a wireless remote temperature, humidity, and soil information system monitoring system and Control system. A central micro-controller will be used to collect all data from the sensor and transmit it to a remote user using the IoT network.

2.2 Android Apps

For visualization and authentication, in this research work, the team has an Android App. The Android Application does not restrict itself to Thing Speak but also has a Firebase login system. Firebase is a Google-provided API to create a database and fetch from it in real time. It also provides enhanced security for the developed App. Also, it is used for backend support and other functionalities like data storage, user authentication and hosting. Real-time data variations are recorded automatically and updates are sent to clients. The HTTP protocol works on a simple request and response system but firebase is different as its real-time database uses data synchronization. It can be used for user authentication purposes as the user credentials are securely stored using crypt. Firebase authentication to the android studio is added using Firebase Android Studio tool. Whenever someone authenticates during the sign-in process either using an Email or Google, it handovers a firebase user object that represents that the authentication is successfully done by the user. This object contains the basic information about the user. Once the firebase API is included in Android or IOS App, firebase features like Analytics, Authentication, Storage, Messaging, Hosting, Crash reporting, Real-time Database etc.

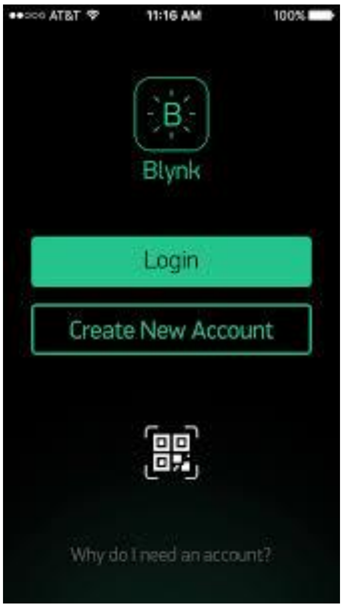

Figure 2.1: Android Apps

2.3 System

Our project operates through the Internet, In the project we can monitor the data of various sensors used which can be monitored from anywhere on the Internet.

CHAPTER : 3

HARDWARE DEVELOPMENT

3.1 Introduction

In this proposed system we have used the Node MCU Development board, sensor and Wi-Fi module. In this project the system is been totally operated by power supply. The main aim of IOT Based Agricultural System. In above block diagram there are Node MCU, Soil Sensor, Dht22 sensor, Pump motor and Relay module which we have used for sensing when is detected the agriculture weather system. Microcontroller continuously check the output of sensor and gives signal to the IoT system circuit which drives the apps.

3.2 Block Diagram

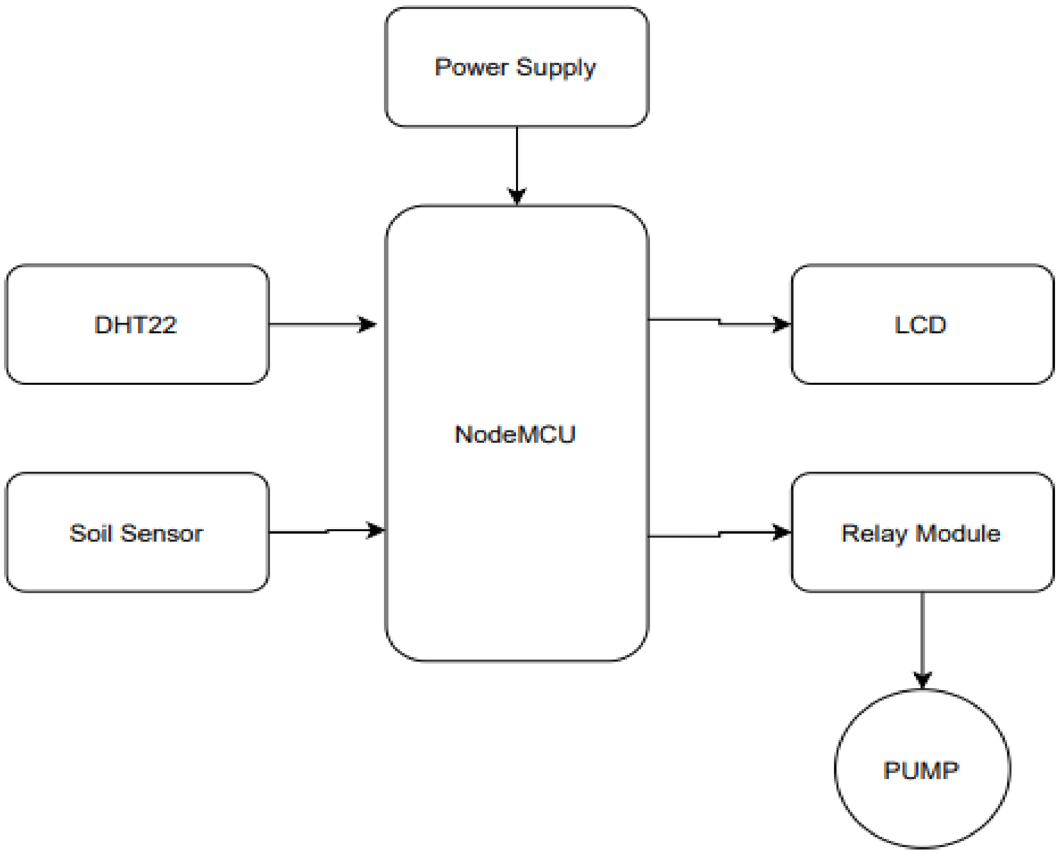

Fig- 3.1: Block Diagram of the Overall System

3.3 Circuit Diagram

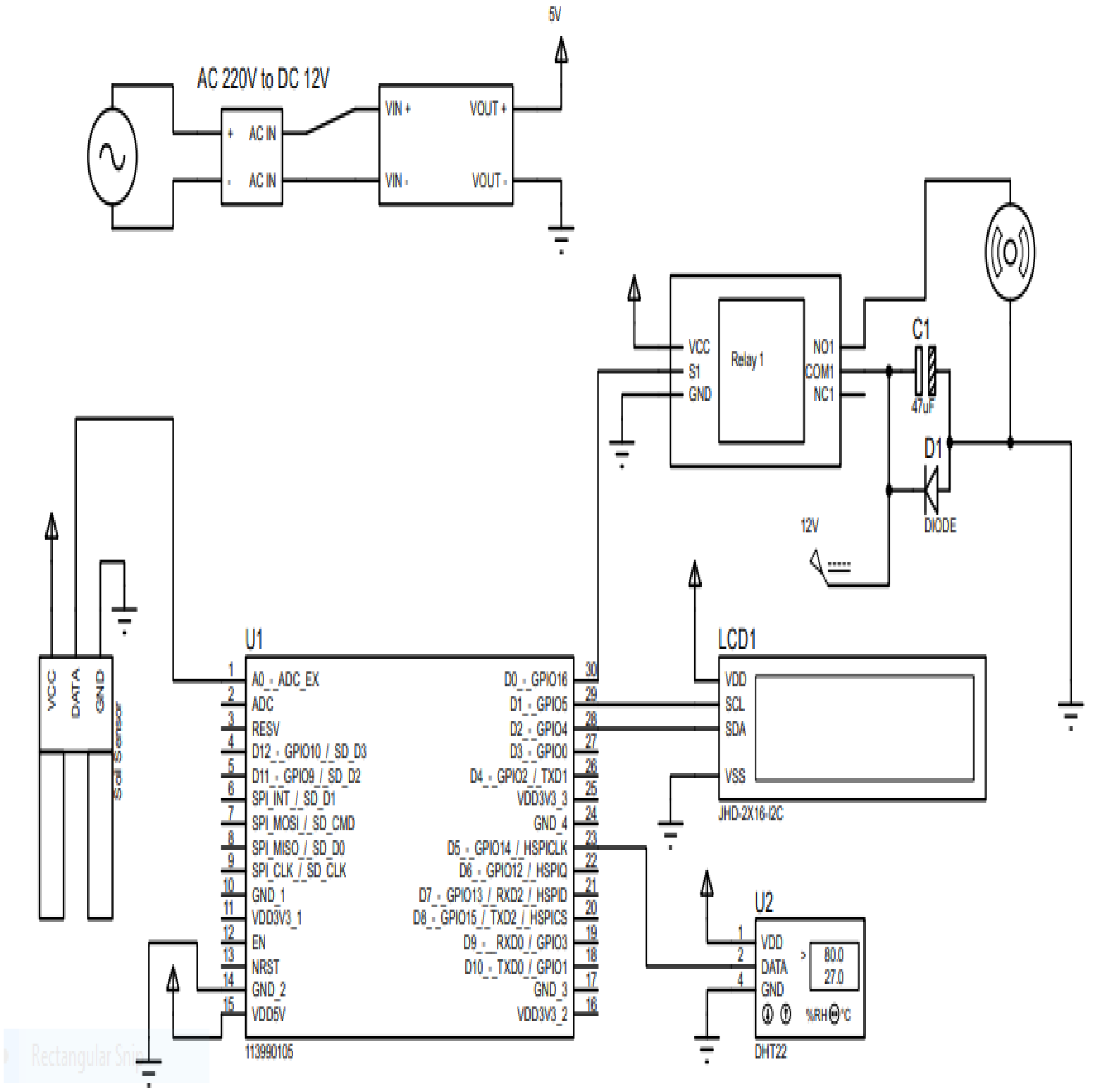

Fig- 3.2: Circuit Diagram

The systems used in our project are shown above, our project we have created a IOT Based Agricultural System that can be controlled with anywhere. We'll get to use all of these systems as we need to, while we use whatever we need, just getting instructions from a internet will start working.

3.4 The list of devices used in the project is given below:

- Node MCU
- Adapter.
- Buck Converter.
- LCD Display.
- DHT22 Sensor.
- Soil Sensor.
- Pump Motor.
- Relay Module.

3.5 Node MCU

NodeMCU is an open source IoT platform.It includes firmware which runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware which is based on the ESP-12 module. The term "NodeMCU" by default refers to the firmware rather than the development kits. The firmware uses the Lua scripting language. It is based on the eLua project, and built on the Espressif Non-OS SDK for ESP8266. It uses many open source projects, such as lua-cjson and SPIFFS.

Fig-3.3: Node MCU

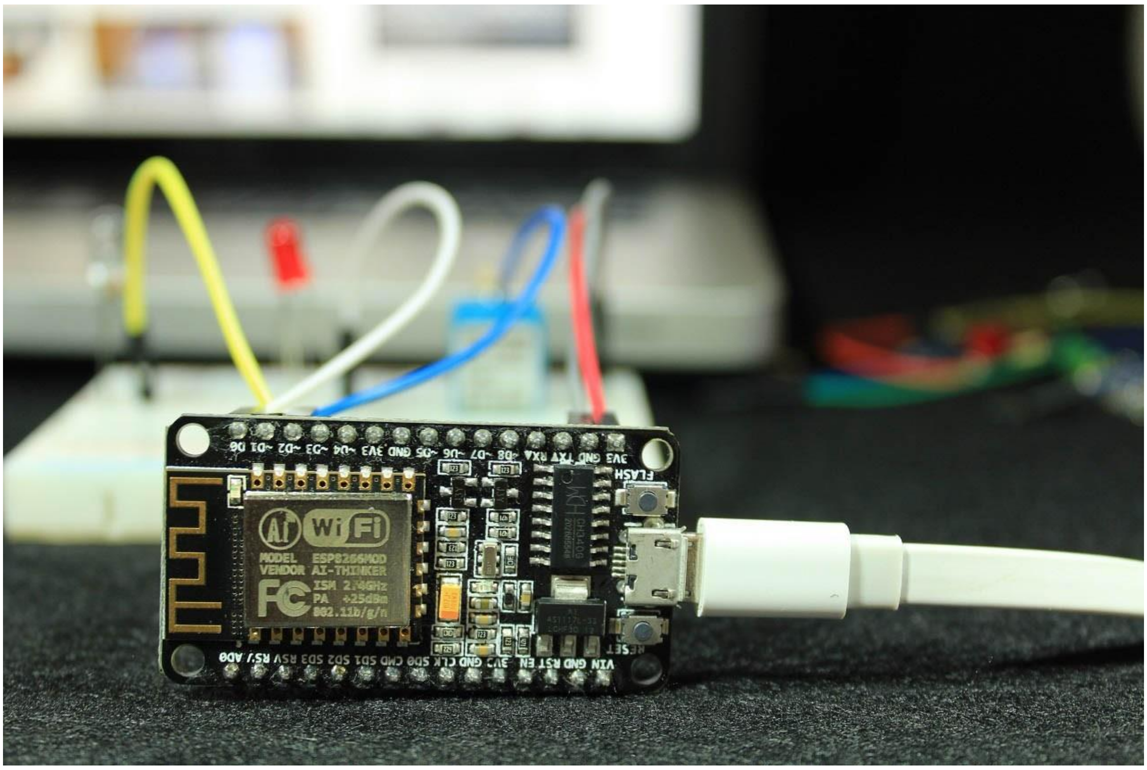

Fig-3.4: Node MCU

ESP8266 Arduino Core:

As Arduino.cc began developing new MCU boards based on non-AVR processors like the ARM/SAM MCU and used in the Arduino Due, they needed to modify the Arduino IDE so that it would be relatively easy to change the IDE to support alternate toolchains to allow Arduino C/C++ to be compiled for these new processors. They did this with the introduction of the Board Manager and the SAM Core. A "core" is the collection of software components required by the Board Manager and the Arduino IDE to compile an Arduino C/C++ source file for the target MCU's machine language. Some ESP8266 enthusiasts developed an Arduino core for the ESP8266 Wi-Fi SoC, popularly called the "ESP8266 Core for the Arduino IDE”. This has become a leading software development platform for the various ESP8266-based modules and development boards, including NodeMCUs..

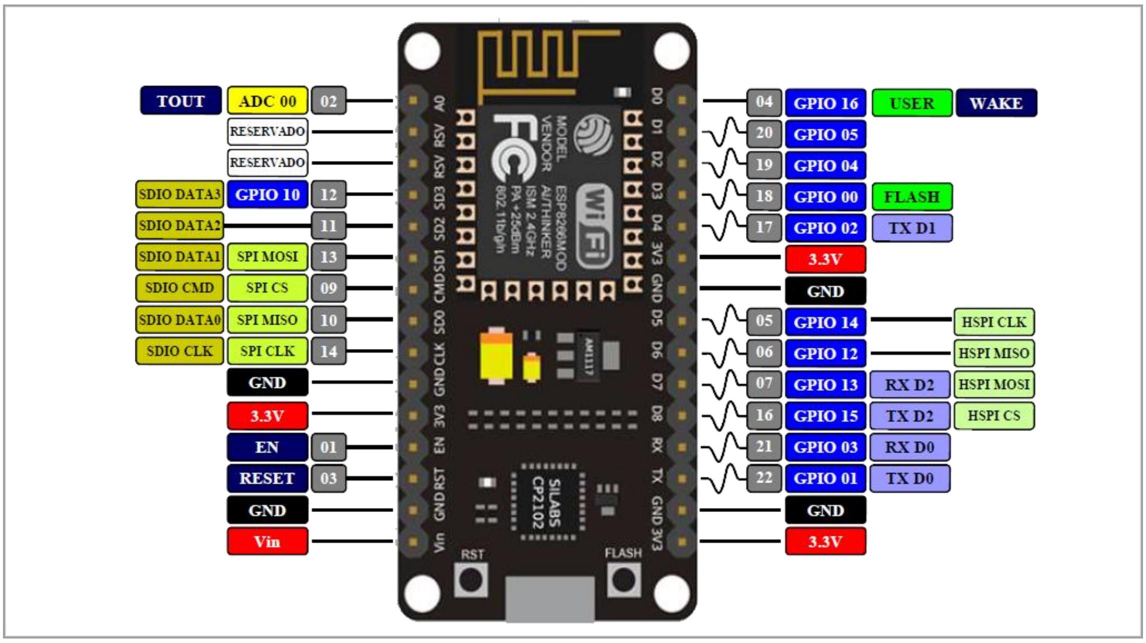

Fig-3.5: Pinout of Node MCU

Summary:

- Easy to access wireless router
- Based on Lua 5.1.4 (without debug, os module.)
- Event-Drive programming preferred.
- Build-in json, file, timer, pwm, i2c, spi, 1-wire, net, mqtt, coap, gpio, wifi, adc, uart and system api.
- GPIO pin re-mapped, use the index to access gpio, i2c, pwm.
- Both Floating Point and Integer versions of the firmware can be built.

Features:

Wi-Fi Module – ESP-12E module similar to ESP-12 module but with 6 extra GPIOs.

USB – micro USB port for power, programming and debugging

Headers – 2x 2.54mm 15-pin header with access to GPIOs, SPI, UART, ADC, and power pins

Misc – Reset and Flash buttons

Power – 5V via micro USB port

Dimensions – 49 x 24.5 x 13mm

Documentation:

You can find the firmware source code and documentation on GitHub, as well as nodemcu-flasher, a Windows only tools to flash the firmware to a module. There's also a separate tool called esptool that will let you flash nodemcu from Linux. In case you find the documentation is all over the place, you might want to checkout NodeMCU video tutorial below.Nodemcu.com is the official website for the project, but you'll find more information on GitHub.

Specification:

- Wi-Fi Module – ESP-12E module similar to ESP-12 module but with 6 extra GPIOs.
- USB – micro-USB port for power, programming and debugging
- Headers – 2x 2.54mm 15-pin header with access to GPIOs, SPI, UART, ADC, and power pins
- Misc. – Reset and Flash buttons
- Power – 5V via micro-USB port
- Dimensions – 49 x 24.5 x 13mm

Software:

A program for Node MCU hardware may be written in any programming language with compilers that produce binary machine code for the target processor. Atmel provides a development environment for their 8-bit AVR and 32-bit ARM Cortex-M based microcontrollers: AVR Studio (older) and Atmel Studio (newer).

IDE:

The Node MCU integrated development environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in the programming language Java. It originated from the IDE for the languages Processing and Wiring. It includes a code editor with features such as text cutting and pasting, searching and replacing text, automatic indenting, brace matching, and syntax highlighting, and provides simple one-click mechanisms to compile and upload programs to an Arduino board. It also contains a message area, a text console, a toolbar with buttons for common functions and a hierarchy of operation menus. The source code for the IDE is released under the GNU General Public License, version 2.

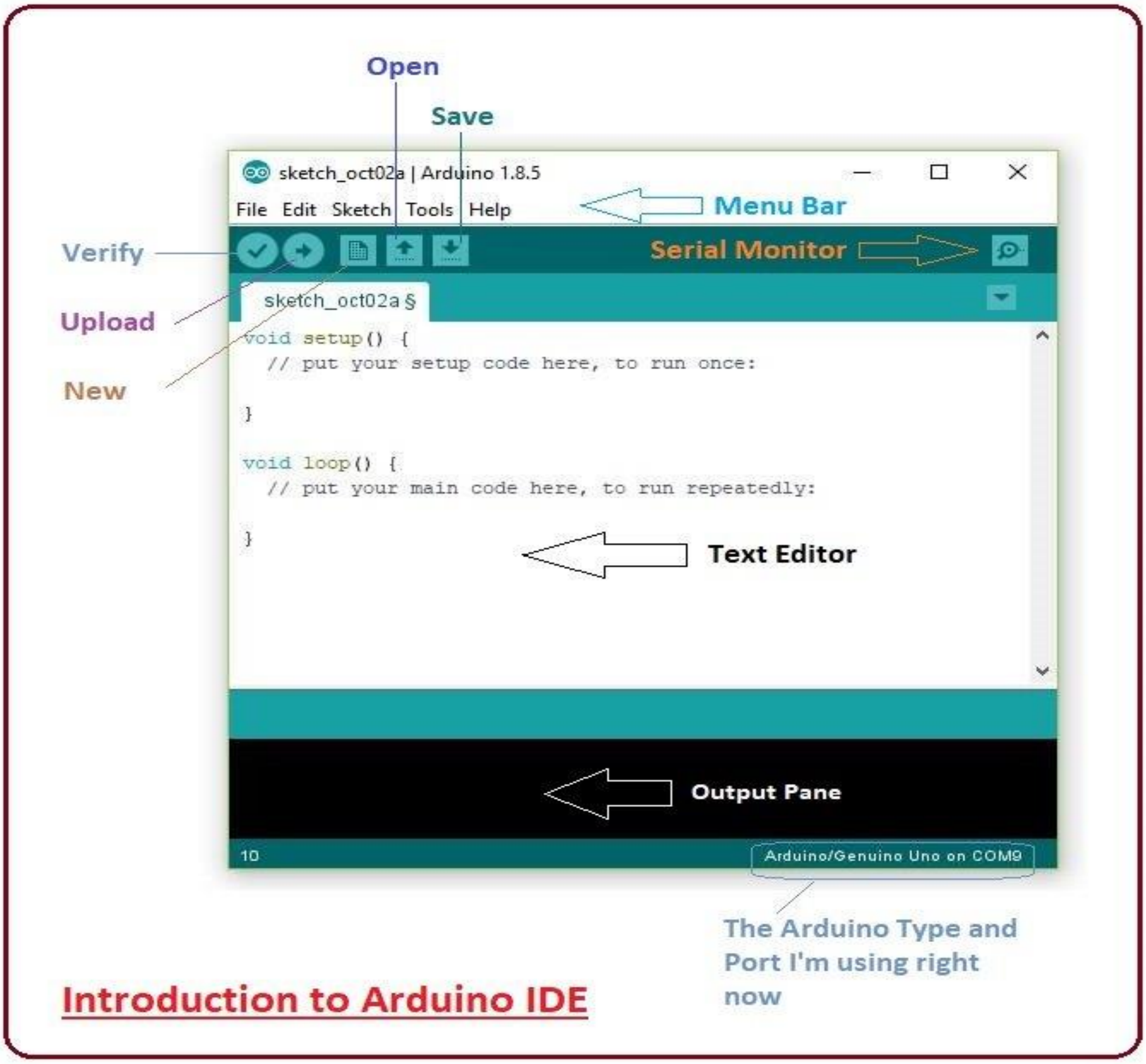

Fig-3.6: IDE software

The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub main() into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution. The Arduino IDE employs the program avrdude to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.

Sketch:

A sketch is a program written with the Arduino IDE. Sketches are saved on the development computer as text files with the file extension. ino. Arduino Software (IDE) pre-1.0 saved sketches with the extension. pde.

A minimal Arduino C/C++ program consists of only two functions:

setup(): This function is called once when a sketch starts after power-up or reset. It is used to initialize variables, input and output pin modes, and other libraries needed in the sketch. It is analogous to the function main().

loop(): After setup() function exits (ends), the loop() function is executed repeatedly in the main program. It controls the board until the board is powered off or is reset. It is analogous to the function while(1).

Blink example:

Most Arduino boards contain a light-emitting diode (LED) and a current limiting resistor connected between pin 13 and ground, which is a convenient feature for many tests and program functions. A typical program used by beginners, akin to Hello, World!, is "blink", which repeatedly blinks the on-board LED integrated into the Arduino board. This program uses the functions pinMode(), digitalWrite(), and delay(), which are provided by the internal libraries included in the IDE environment. This program is usually loaded into a new Arduino board by the manufacturer.

3.6 Adaptor

12V 5A Micro Power Adapter. For using with Arduino /Raspberry Pi 3 Model A+/B/B+/Zero and running any other high current devices.

Specifications:

Input: 100V - 240V AC, 50Hz/60Hz

Output: 12V 5A

Fig- 3.7: Adapter

3.7 LM2596 Voltage Converter

LM2596 Buck DC-DC Adjustable Step Down Power Supply Module Converter.

* Module Property: Non-detached money
* Correction mode: Non-synchronous correction
* Size: About 48 * 23 * 14mm (Length * Width * Height)
* Type: LM2596 adjustable power supply module
* Input voltage: DC 4V-35V
* Output voltage: DC 1.23V-30V
* Output Current: 3A (maximum)
* Conversion efficiency: 92% (maximum)
* Output ripple: 30 mV (maximum)
* Switching frequency: 150KHz
* Load control: 0.5%
* Voltage control: 2.5%
* Working temperature: -40? - +85?

Fig-3.8 LM2596 Voltage Converter

3.8 LCD

Pin	Function
VSS	connected to ground
VDD	connected to a +5V power supply
VO	to adjust the contrast
RS	A register select pin that controls where in the LCD's memory you are writing data to. You can select either the data register, which holds what goes on the screen, or an instruction register, which is where the LCD's controller looks for instructions on what to do next.
R/W	A Read/Write pin to select between reading and writing mode
E	An enabling pin that reads the information when High level (1) is received. The instructions are run when the signal changes from High level to Low level.
D0-D7	to read and write data
A	Pins that control the LCD backlight. Connect A to 3.3v.
K	Pins that control the LCD backlight. Connect K to GND.

It is a 16-character X2-line blue background super-twisted nematic (STN) LCD with built-in HD44780 equivalent controller (also known as alphabetical display).Interfacing is facilitated by 4 bits or 8 bits, and the programming code is widely available for many different controllers and systems.

Tabel-01: Lcd display outline

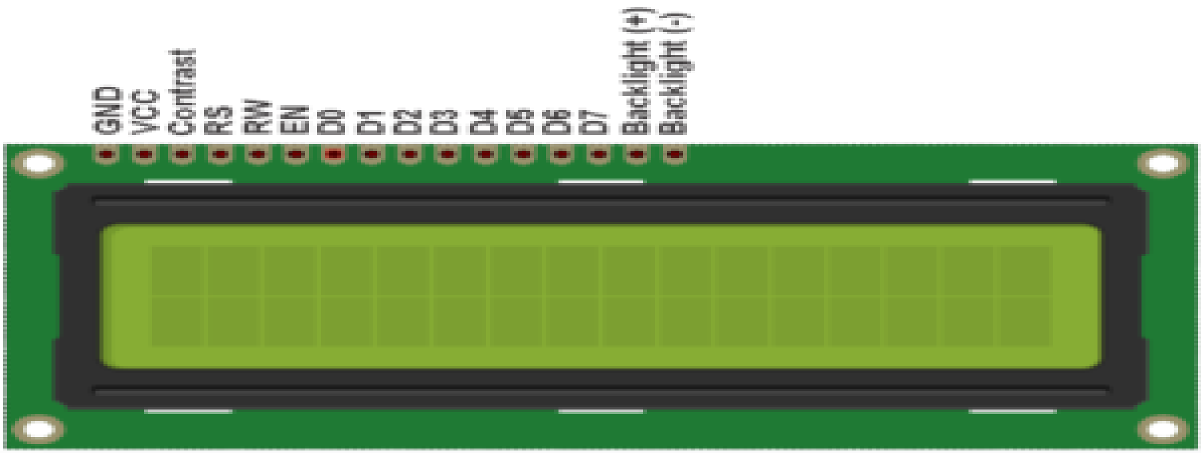

Fig- 3.9: LCD Display

3.9 DHT22

DHT22 is a composite digital temperature and humidity sensor that outputs calibrated digital signals. Thanks to the dedicated digital module acquisition technology and the application of temperature and humidity sensing technology to the module, DHT22 brings very high reliability and excellent long-term durability. The DHT22 has a capacitive moisture sensor and an NTC temperature measuring element that is connected to a high-performance 8-bit microcontroller, resulting in a great quality, super-fast response time, powerful anti-interference capability and highly cost effective.

Fig- 3.10: DHT-22 Sensor

Speciation:

I. Supply voltage: 5V
II. Temperature Range: -40-80 resolution / resolution 0.1 ℃ / error <± 0.5 ℃
III. Humidity Range: 0-100% RH / Resolution 0.1% RH / Error ± 2% RH
IV. Temperature response time: Condition: 1 / E (63%) Minimum 6s maximum 20s
V. Humidity Response Rhyme: Condition: 1 / e (63%) 25 ℃, 1 m / s in air <5 s
VI. Interface Sequence: VCC, GND, SS
VII. Size: 38 x 20 mm

3.10 Soil Sensor

The moisture sensor is used to measure the amount of soil water (moisture).When there is a water shortage on the ground, the module output is at a high level, otherwise the output is at a low level.This sensor reminds users to water their plants and also monitors soil moisture.It has been widely used in agriculture, land irrigation and plant gardens.

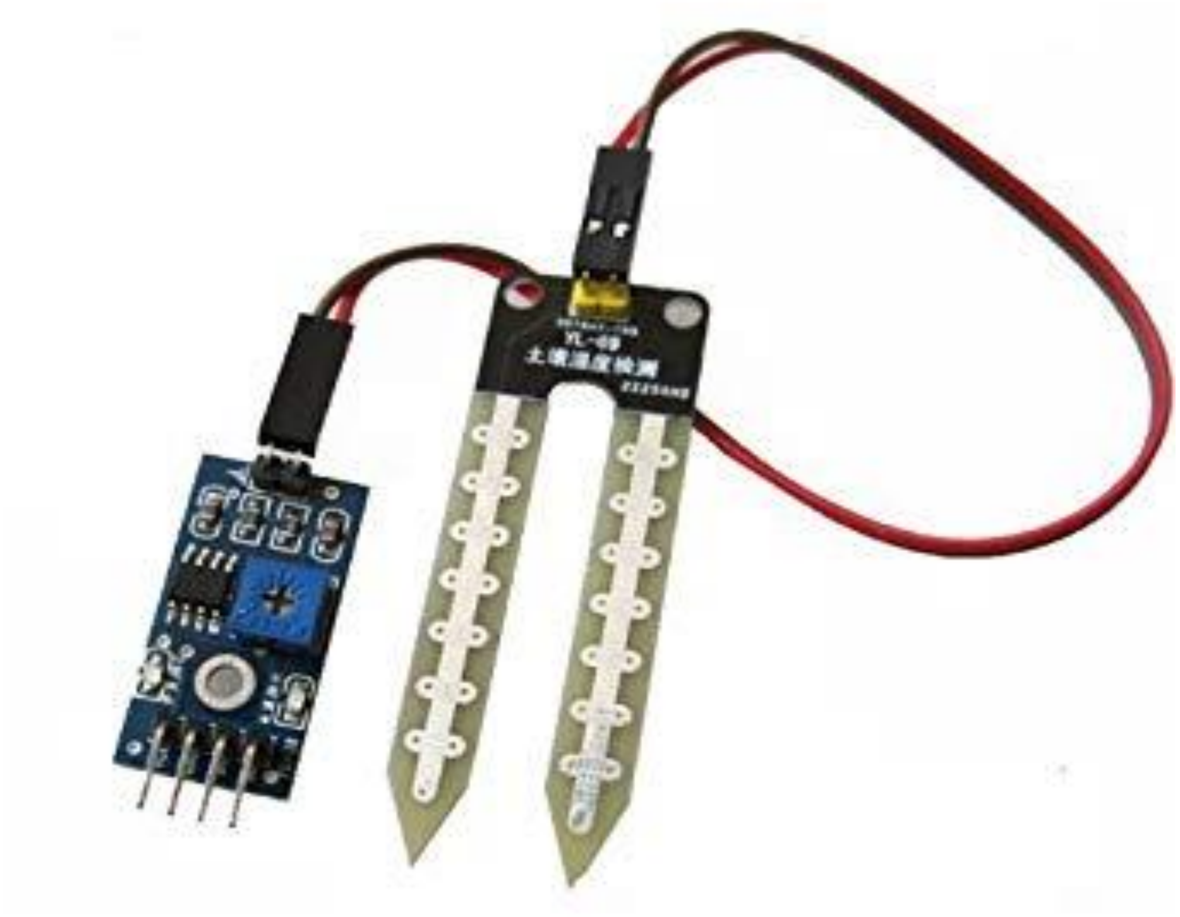

Fig-3.11: Soil Sensor

Specifications:

I. Working voltage: 5V
II. Running running: <20mA
III. Interface Type: Analog
IV. Working temperature: 10 ° C ~ 30 ° C

Sample Program:

```
intsensorPin = A0;    // select the input pin for the potentiometer
intsensorValue = 0;  // variable to store the value coming from the sensor
void setup()
{
Serial.begin(9600);
}
void loop()
{
sensorValue = analogRead(sensorPin);  // read the value from the sensor:
delay(1000);
Serial.print("sensor = " );
Serial.println(sensorValue);
}
```

3.11 Pump Motor

DC powered pumps use direct current from motor, battery, or solar power to move fluid in a variety of ways. Motorized pumps typically operate on 6, 12, 24, or 32 volts of DC power. Solar-powered DC pumps use photovoltaic (PV) panels with solar cells that produce direct current when exposed to sunlight.

Fig-3.12: Pump Motor

DC 5V Pump Mini water pump For Fountain Garden Mini water circulation System.

- DC Voltage:2.5-6V
- Maximum lift:40-110cm / 15.75"-43.4"
- Flow rate:80-120L/H
- Outside diameter of water outlet: 7.45mm / 0.3" (our 7mm inner diameter tube is ok for this pump)
- Inside diameter of water outlet: 4.7mm / 0.18"
- Diameter:Approx. 24mm / 0.95"
- Length:Approx. 45mm / 1.8"
- Height:Approx. 33mm / 1.30"
- Material:engineering plastic
- Driving mode: brushless dc design, magnetic driving

Working principle of Pump Motor:

DC motor is any rotary electrical machine converting direct current electrical energy into mechanical energy. Once the motor is on, it creates a magnetic field around it making the rotor spin. When the rotor is in action, it causes the impeller to spin, hence, powers the pump.

3.12 Relay Module

The relay module is an electrically operated switch that can be turned on or off deciding to let current flow through or not. They are designed to be controlled with low voltages like 3.3V like the ESP32, ESP8266, etc, or 5V like your Arduino.

Features:

The features of 1-Channel Relay module are as follow:

1) Good in safety. In power system and high voltage system, the lower current can control the higher one.

2) 1-channel high voltage system output, meeting the needs of single channel control

3) Wide range of controllable voltage.

4) Being able to control high load current, which can reach 240V, 10A

5) With a normally-open (NO) contact and a normally-closed (NC) contacts
Interface specifications:

The output contacts of a relay (including NO, NC, and the common port) works as a SPDT – Single Pole Double Throw switch. Its operating principle is as follow: VCC----5V,

GND----for ground

IN1 connects to the control valve which output 3V-5V

Output contacts:

Connect to applications Interface Connecting and Setting:

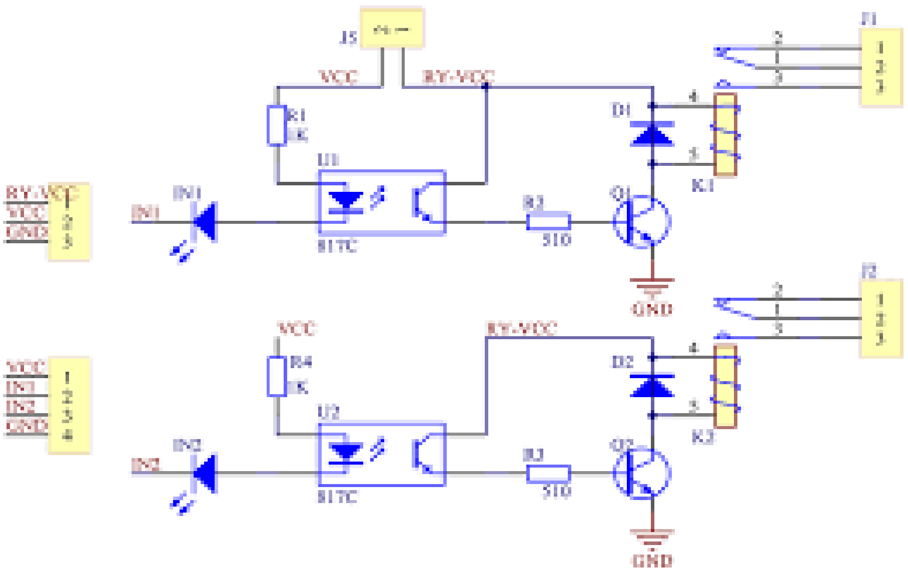

Fig- 3.13: Stematic Of Relay Module

Get Started:

Firmware resources: Arduino board (any versions), wires, LED, 5v power supply. Software resource: Arduino IDE The one-channel relay can be programmed to realize the open and close automatically. NB: customers can use any software or firmware to control the module as long as the IN1 of which can input a voltage of 3V-5V.

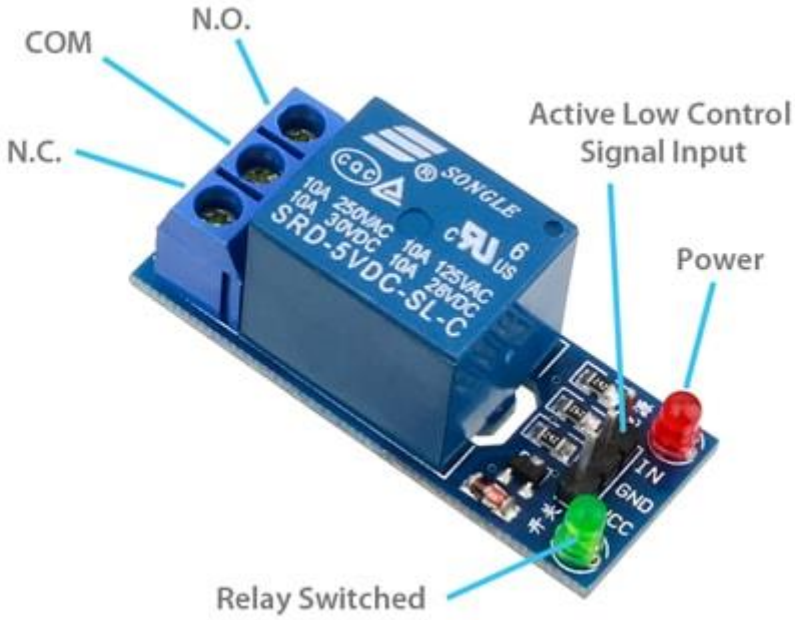

Fig-3.14: Pinout Of Relay Module

you can do further development with the development tool you like as well as test it in the way of testing firmware. Firmware test: after the connection as in picture1-4, pay attention to the blink of LED, listen to the flicker of relay when it is working. Software test code:

```
 void setup(){
  pinMode(7,OUTPUT);
  }
  void loop(){
 pinMode(7,OUTPUT);
 delay(2000);
 digitalWrite(7,LOW);
 delay(2000);
 }
```

Feedback of test:

after the upload of code, connect the IN1 pins to the 7th pin of Arduino board separately, you can hear the ticktack of the relay (it keeps opening and closing), in addition, the LED-G is blinking every 2s.

3.13 System physical appearance

The below image is showing the total system is in OFF mode. There is no power is supplied & device is not connected with the android phone.

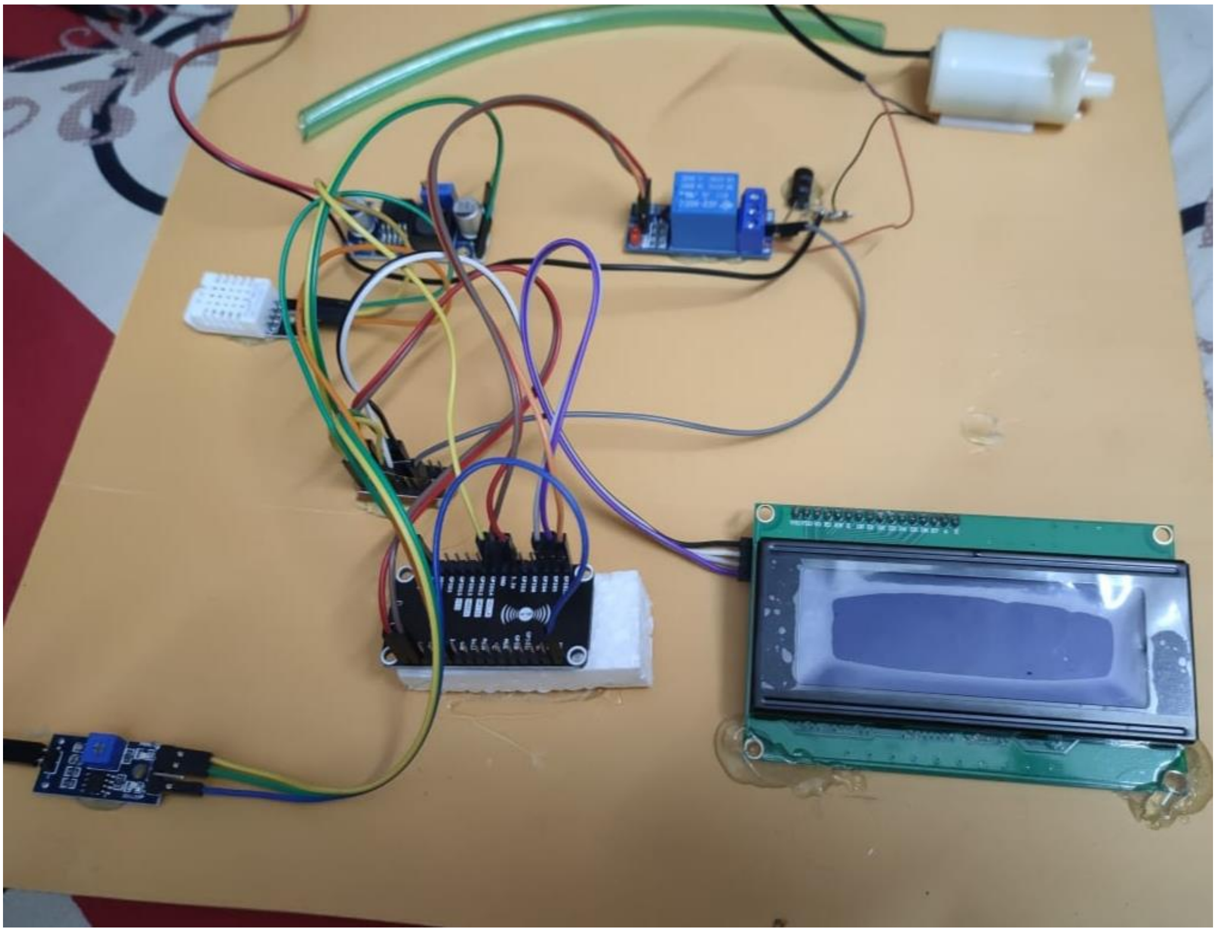

Fig-3.15: Project Picture

3.14 Result

In this project we will monitor agricultural weather, Such as the environment Temperature & humidity and soil moisture. We can see all the project data in an lcd display. Our project operates through the Internet, In the project we can monitor the data of various sensors used which can be monitored from anywhere on the Internet.

We also monitor all data via the Internet and we can control irrigation water from any place. We used Node MCU as the controller and we created the C language as a programming language and uploaded it to the controller.

Fig-3.16: LCD Display Monitoring System

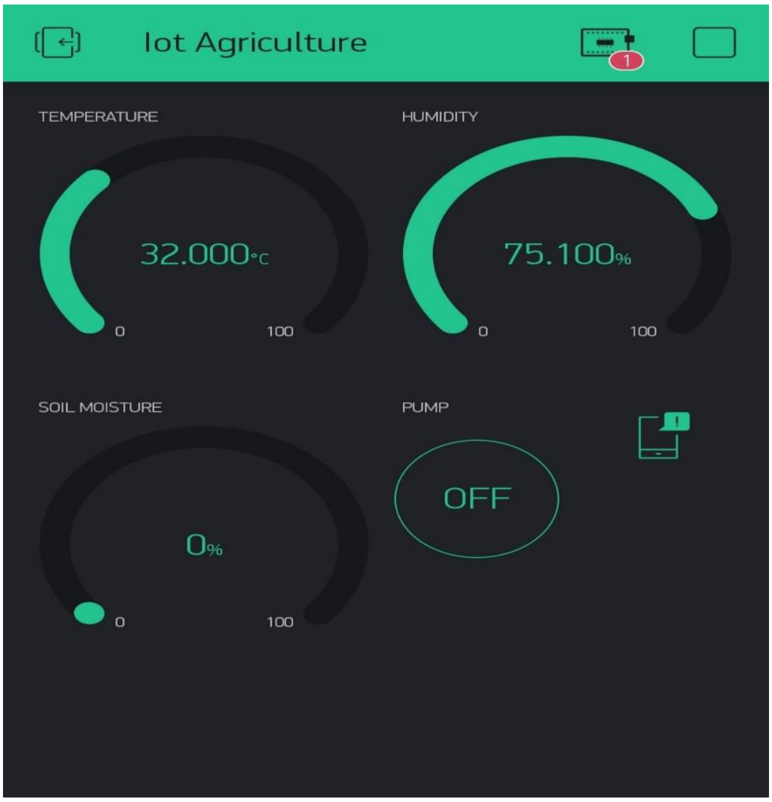

Fig-3.17: Apps Monitoring and Controlling System

3.15 List of Components

SL NO	Components Name	Quantity
1	Node MCU	1
2	Buck Module	1
3	Adaptor	1
4	Pump Motor	1
5	Relay Module	1
6	DHT 22 Sensor	1
7	Soil Sensor	1
8	LCD Display	1
9	Others	1

Table02 -List of Components

CHAPTER : 4

SOFTWARE DEVELOPMENT

4.1 Introduction

Software part is one of the main part of the project, controlling & protection system. The Algorithm is based on different conditions & measuring parameters. Total function of the system is controlled by the software. The code is written on C.

4.2 Project Flow Chart

In any programming related project there is a part named “flowchart” is must. As per rules we made our project flowchart then wrote our proposed project program. There are several parts of our project thus many flowcharts we made.

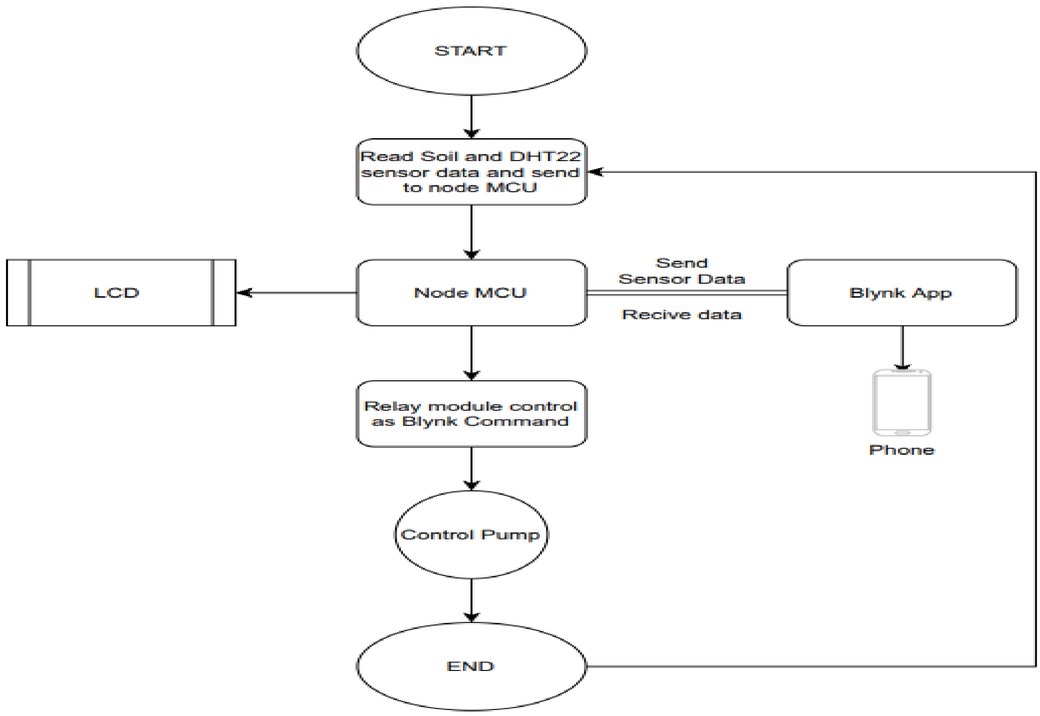

Fig-4.1: Project Flow Chart

4.3 Software Tools

The software that is used to program the microcontroller is open-source-software and can be downloaded for free on www.arduino.cc. With this “Arduino software” we can write little programs with the microcontroller. These programs are called “Sketch”.

In the end the sketches are transferred to the microcontroller by USB cable. More on that later on the subject “programming”.

Installation

Now one after another the Arduino software and the USB driver for the board have to be installed.

Installation and setup of the Arduino software

1. We have downloaded the Arduino software from www.arduino.cc and installed it on the computer (This was NOT connected to the PC). After that we opened the software file and installed the program named arduino.exe.

Two set ups on the program are important and should be considered.

a) The board that we want to connect has to be selected on the Arduino software. The “Arduino Uno” is here known as “arduino/genuino uno”.

b) We have to choose the right “Serial-Port”, to let the computer know to which port the board has been connected. That is only possible if the USB driver has been installed correctly. It can be checked this way:

At the moment the Arduino wasn’t connected to the PC. If we now choose “Port”, under the field “Tool”, we will already see one or more ports here (COM1/ COM2/ COM3…). The quantity of the shown ports doesn't depend on the quantity of the USB ports on the computer. When the board gets connected to the computer, we will find one more port.

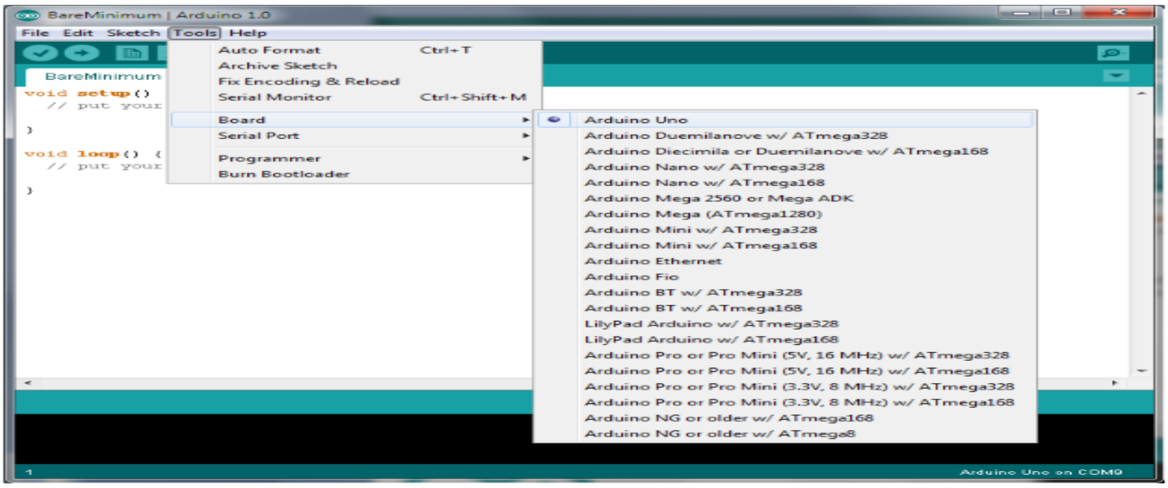

Fig- 4.2: Program installation process

4.4 Programming

The development cycle is divided into 4 phases:

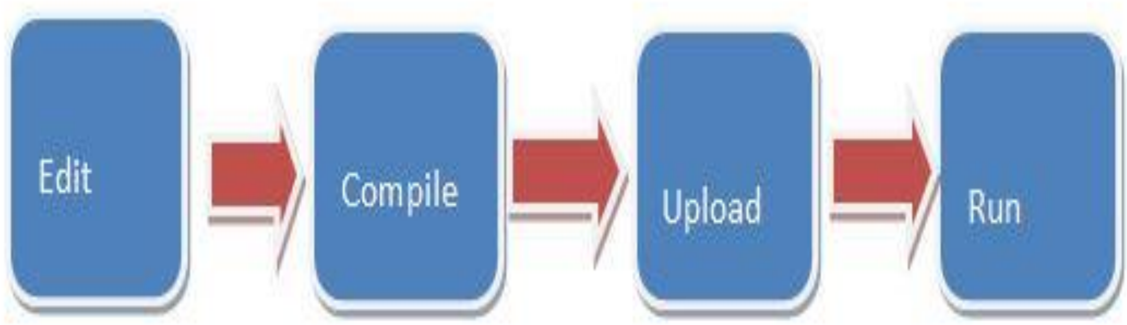

Fig-4.3: Flowchart of the compiling process

Compile: Compile means to translate the sketch into machine language, also known as object.

Code Run: Arduino sketch is executed as soon as terminates the step of uploading on the board.

4.5 Arduino Program Development

- Based on C++ without 80% of the instructions.
- A handful of new commands.
- Programs are called 'sketches'.
- Sketches need two functions:
- void setup ( )
- Void loop ( )
- Setup ( ) runs first and once.
- loop ( ) runs over and over, until power is lost or a new sketch is loaded.

4.6 Proteus 8.1

Proteus 8 is best simulation software for various designs with microcontroller. It is mainly popular because of availability of almost all microcontrollers in it. So, it is a handy tool to test programs and embedded designs for electronics hobbyist. You can simulate your programming of microcontroller in Proteus 8 Simulation Software. After simulating your circuit in Proteus 8 Software you can directly make PCB design with it so it could be an all-in-one package for students and hobbyists. So, I think now you have a little bit idea about what is proteus software. Proteus 7.0 is a Virtual System Modeling (VSM) that combines circuit simulation, animated components and microprocessor models to co-simulate the complete microcontroller-based designs. This is the perfect tool for engineers to test their microcontroller designs before constructing a physical prototype in real time. This program allows users to interact with the design using on-screen indicators and/or LED and LCD displays and, if attached to the PC, switches and buttons. One of the main components of Proteus 7.0 is the Circuit Simulation -- a product that uses a SPICE3f5 analogue simulator kernel combined with an event-driven digital simulator that allow users to utilize any SPICE model by any manufacturer. Proteus VSM comes with extensive debugging features, including breakpoints, single stepping and variable display for a neat design prior to hardware prototyping. In summary, Proteus 7.0 is the program to use when you want to simulate the interaction between software running on a microcontroller and any analog or digital electronic device connected to it.

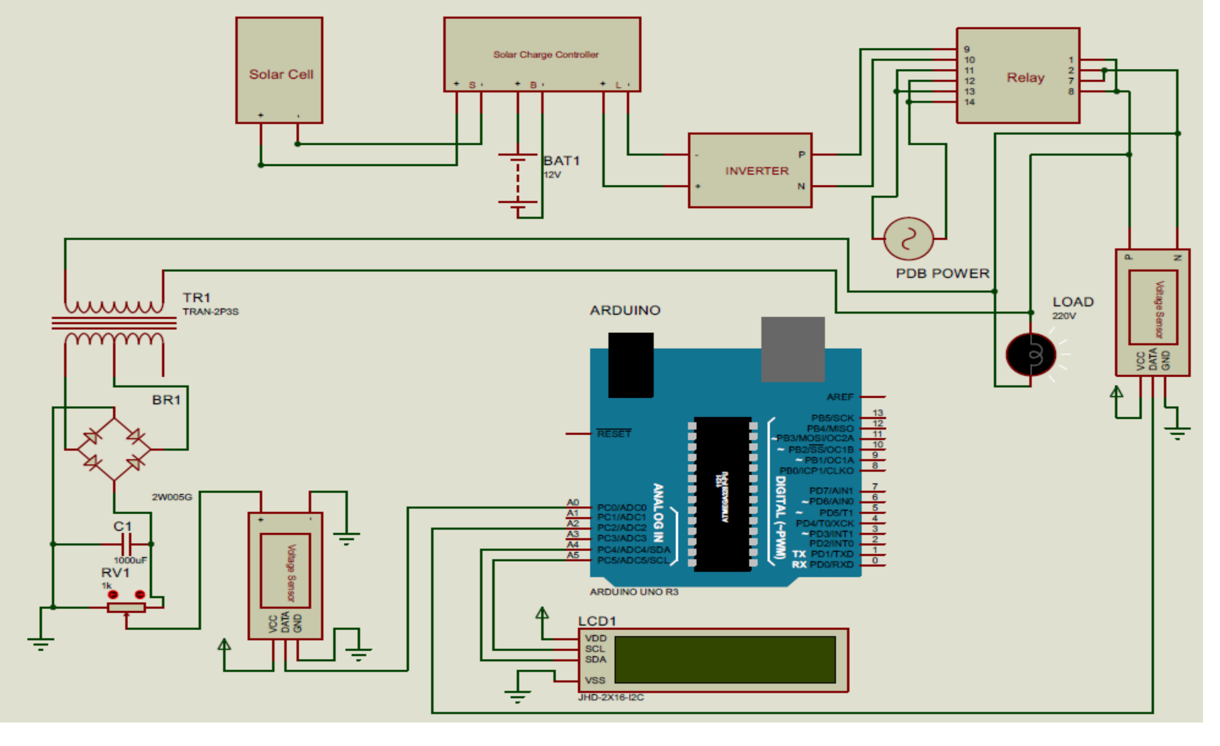

Fig-4.4: User Interface of Proteus 8.1

Chapter : 5

Discussions and Conclusions

5.1 Suggestion for future Work

We have successfully completed our project with available sources. But the results and changes do not depend on expectations. This can be further improved by incorporating the following changes to get better results. The process that we used in the Scotch yoke mechanism does not work efficiently. This skill can be enhanced by using some other mechanisms.

5.2 Conclusions & Discussions

The proposed framework will revolutionize the traditional agriculture system used by majority of the farmers. This will allow the younger generation to do farming without previous experience and with ease. The system if implemented in maximum fields, will generate vast amount of data that can be used for data mining and further research in this field. This framework will conserve all the major resources by utilizing them more efficiently. This system requires minimum human interaction at present, which can be further reduced, and can even completely be eliminated by further improving the framework. There are many machines learn in g algorithms for image classification. This framework had only used SVM based classifier, hence other machine learning algorithms can be implemented and tested to further improve the system.

5.3 Advantages

I. Easy to Install
II. Updates On mobile phone directly
III. Agriculture weather monitoring System
IV. Remote monitoring and controlling.

5.4 Appendix

Programme:

```
#define BLYNK_PRINT Serial



#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include "DHT.h"

#define DHTPIN D5
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);

// Set the LCD address to 0x27 for a 16 chars and 2 line display
LiquidCrystal_I2C lcd(0x27, 20, 4);

// You should get Auth Token in the Blynk App.
// Go to the Project Settings (nut icon).
char auth[] = "-LzzWlstQNyfwhsUUbbZh8sXkYDI56s9";

// Your WiFi credentials.
// Set password to "" for open networks.
char ssid[] = "student";
char pass[] = "iotstudent";

int pinValue;
int pinValue1;
int pinValue2;

int relay = D0;
```

```
int soil = A0;
int soildata;

int relay = D6;

int flag = 0;

void setup()
{
  // Debug console
  Serial.begin(9600);
  pinMode(relay, OUTPUT);
  pinMode(soil, INPUT);

  digitalWrite(relay, HIGH);

  dht.begin();
  lcd.init();                      // initialize the lcd
  lcd.backlight();

  // You can also specify server:
  Blynk.begin(auth, ssid, pass, IPAddress(192,168,1,100), 8080);
}



void loop()
{
  soildata = analogRead(soil);
  soildata = map(soildata, 1024, 650, 0, 100);
  //  Serial.println(soildata);
  if (soildata <= 0)
  {
    soildata = 0;
```

```
}

if (soildata >= 100)
{
  soildata = 100;
}

float h = dht.readHumidity();
// Read temperature as Celsius (the default)
float t = dht.readTemperature();



Blynk.virtualWrite(V0, t); //virtual pin V0
Blynk.virtualWrite(V1, h); //virtual pin V1
Blynk.virtualWrite(V2, soildata); //virtual pin V2
delay(1000);
lcd.clear();

lcd.setCursor(0, 0);
lcd.print("Temperature:");
lcd.print(t);
lcd.print("C");

lcd.setCursor(0, 1);
lcd.print("Humidity:");
lcd.print(h);
lcd.print("%");

lcd.setCursor(0, 2);
lcd.print("Soil Moisture:");
lcd.print(soildata);
lcd.print("%");
```

```
  if (soildata <= 20 && flag == 0)
  {
    Blynk.notify("Water lavel Low!");
    digitalWrite(relay high, HIGH);
    flag = 1;
  }

  if (soildata >= 80 && flag == 1)
  {
    digitalWrite(relay, LOW);
    flag = 0;
  }

}
```