A PROJECT REPORT ON

**AGRI BOT**

**BACHELOR OF SCIENCE**

**(PHYSICS)**

**UNDER**

**UNIVERSITY OF CALICUT**

FOR THE PARTIAL FULFILLMENT OF DEGREE

PROGRAMME 2017-2020



SUBMITTED BY

**ABDULLA AMAN M P (MEARSPH033)**

UNDER THE GUIDANCE OF

**C.A. SAFEEQUE**

(HEAD OF THE DEPARTMENT, DEPARTMENT OF PHYSICS,

SULLAMUSSALAM SCIENCE COLLEGE)

DEPARTMENT OF PHYSICS, SULLAMUSSALAM SCIENCE COLLEGE, AREEKODE

**CERTIFICATE**



This is to certify that the project report titled “**AGRI BOT**” is an authentic record of the project work carried out by **ABDULLA AMAN M P** with register number **MEAQSPH033** of sixth semester B.Sc Physics, Sullamussalam Science College, Areekode under my supervision and guidance in partial fulfillment of the award of Bachelor of science in physics under the faculty requirements of Science, University of Calicut during the academic year 2017-2020.

PROJECT ADVISOR HEAD OF THE DEPARTMENT

EXAMINER

**DECLARATION**

I, ABDULLA AMAN M P, hereby declare that the project entitled **‘AGRI BOT’** is a bonafide record of work done by me and submitted to University of Calicut in partial fulfillment of the B.Sc Physics programme (2017-2020).

Place : Areekode ABDULLA AMAN M P

Date :

ACKNOWLEDGEMENT

I am extremely indebted to God Almighty for the blessings showered on me through this project.I express my profound gratitude to my project Advisor, C.A. SAFEEQUE , the Head of the Department of Physics, Sullamussalam Science College, Areekode, for his valuable guidance, co-operation and constant encouragement during my work.

I am grateful to all other faculty members in the Department of Physics, Sullamussalam Science College, Areekode, for their inspiration and co-operation for completing this work. I am also grateful to the Innovation and Entrepreneurship Development Cell, Sullamussalam Science College, Areekode for providing me with the necessaries needed for the project. Heartfelt gratitude to my parents for providing all the help I need to complete this work. I express my sincere gratitude to my friends and every person, who gave me valuable support and suggestions for the successful completion of this project.

CONTENTS

PAGE NO

* Abstract 6
* Introduction 7
* Internet of Things (IoT) 8
* Agri Bot 10
* Microcontroller 11
* Arduino 12
  + Arduino Uno
* NodeMCU 13
* Soil Moisture Sensor 14
* DHT11 Sensor 16
* GSM Shield 18
* Cloud Computing 19
* Telegram 20
* Blynk 21
* Arduino IDE 22
* Agri bot Connection
  + DHT11 – Blynk Connection
  + Soil moisture Sensor – Blynk Connection
  + Agri bot Connection
* Conclusion 32
* References 33

ABSTRACT

Here in this project by the help of sensors (Soil Moisture, DHT11) and Microcontrollers (Arduino, NodeMcu, etc.,), we are developing a system which helps farmers to automate the field. This paper summarizes all important steps during AgriBot design, such as how the connection can be done, working principles of sensors, how we can program a Microcontroller using Arduino Ide. This innovative product can, later on, be produced as an industrial product.

The agribot developed as the final product of this project can only be used as a prototype. The industrial development of the product can be done later more easily and at a less cost.

INTRODUCTION

With the growing adoption of **the Internet of Things (IoT)**, connected devices have penetrated every aspect of our life, from health and fitness, home automation, automotive and logistics, to smart cities and industrial IoT. Thus, it is only logical that **IoT, connected devices, and automation** would find its application in agriculture, and as such, tremendously improve nearly every facet of it. How could one still rely on horses and plows when self-driving cars and virtual reality are no longer a sci-fi fantasy but an everyday occurrence?

Farming has seen a number of technological transformations in the last decades, becoming more industrialized and technology-driven. By using various smart agriculture gadgets, farmers have gained better control over the process of raising livestock and growing crops, making it more predictable and improving its efficiency.

Agriculture is the backbone of the Indian economy and around 70% of the population depends on this field to run their livelihood. From time immemorial agriculture has been a part of the human civilization. It has transformed the way humans survive. The economy of a particular area was indirectly dependent on agriculture, and was a major thrust behind the industrial revolution. Advancements in the field of science and technology led to increased yield. Applying electronic monitoring systems is one of the technologies for analysing important conditions required for optimum growth of plants. The conditions can be listed as temperature, humidity and soil moisture. There are valuable data that could decide the plant life cycle. Efficient use of these parameters increases the output per plant and minimizes crop loss

Using Soil moisture sensor, DHT11 sensor and Other Micro controllers like Arduino Uno and Nodemcu in this project we send the current status of the field to the farmer through cloud via internet.

Internet of Things (IoT)

The Internet of Things (IoT) refers to a vast number of “things” that are connected to the internet so they can share data with other things – IoT applications, connected devices, industrial machines and more. Internet-connected devices use built-in sensors to collect data and, in some cases, act on it. IoT connected devices and machines can improve how we work and live. Real-world Internet of Things examples range from a smart home that automatically adjusts heating and lighting to a smart factory that monitors industrial machines to look for problems, then automatically adjusts to avoid failures.

Any device, if it has an on and off switch then chances are it can be a part of the IoT. Very often the connected devices will have an I.P address. With Internet Protocol Version 6 (IPv6), assigning an IP address to billions of devices has become very much feasible.

Examples of ‘things’ which can be connected to internet include:

* Connected Wearables – Smartwatches, Smart glasses, fitness bands etc.
* Connected Homes – connecting household appliances to the network.
* Connected Cars – vehicles that are connected to the internet.
* Connected Cities – smart meters which analyse usage of water, gas, electricity etc connect cities to IoT

These ‘things’ need certain sensors and chips in which the codes are uploaded. The coding can be done in different ways.

**Advantages**

1. Communication: - IoT encourages the communication between devices, also famously known as Machine-to-Machine (M2M) communication.
2. Automation and Control: - Without human intervention, the machines are able to communicate with each other leading to faster and timely output.
3. Monitor: - Monitoring each data is essential for any system which through IoT becomes easy.
4. Efficient and Saves Time: - The machine-to-machine interaction provides better efficiency, hence; accurate results can be obtained fast. This results in saving valuable time.
5. Better Quality of Life: - All the applications of this technology culminate in increased comfort, convenience, and better management, thereby improving the quality of life.

Example*:*

The Agribot in this project is an example for internet of things. Discussion about agribot and its parts are done below.

Agri Bot

Agribot is a complete personal assistant which helps the farmer to maintain his farming field. The main aim of this system is to help the farmer to monitor the temperature, humidity and moisture of the field. The system provides the farmer to see the current state of field and sends notification to the farmer if the field needs any water. The farmer can send another message to the agribot through his/her mobile phone, Laptop, or any other device which provides cloud service.

For example we can use the open source chatting platform called Telegram. In telegram we create our own chatting bot through which all the information are shared. The farmer gives certain commands like “Field temp” then the bot gives the current temperature inside the field “37” and also farmer can command the bot to on the water pump.

TechnologiesUsed

* Arduino Microcontroller
* Nodemcu Wi-Fi Chip (esp8266)
* Sensors (Soil Moisture , DHT11)
* Arduino IDE
* Telegram Bot (Or GSM mod

Microcontroller

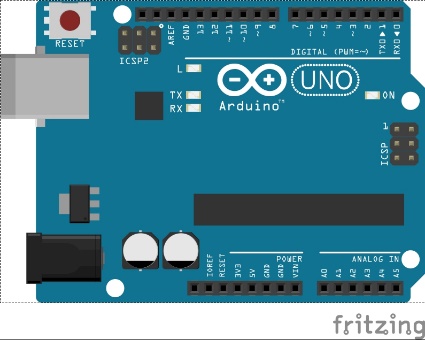
This is the heart of the Agribot. All the arithmetic and non-arithmetic calculations takes place inside this particular board. A **microcontroller** (**MCU** for *microcontroller unit*) is a small computer on a single metal-oxide-semiconductor (MOS) integrated circuit chip. Microcontrollers are designed for embedded applications, in contrast to the microprocessors used in personal computers or other general purpose applications consisting of various discrete chips. Microcontrollers are used in automatically controlled products and devices, such as automobile engine control systems, implantable medical devices, remote controls, office machines, appliances, power tools, toys and other embedded systems. By reducing the size and cost compared to a design that uses a separate microprocessor, memory, and input/output devices, microcontrollers make it economical to digitally control even more devices and processes.

Working

A microcontroller is embedded inside of a system to control a singular function in a device. It does this by interpreting data it receives from its I/O peripherals using its central processor. The temporary information that the microcontroller receives is stored in its data memory, where the processor accesses it and uses instructions stored in its program memory to decipher and apply the incoming data. It then uses its I/O peripherals to communicate and enact the appropriate action.

Arduino

Arduino is an open-source platform used for building electronics projects. Arduino consists of both a physical programmable circuit board (often referred to as a microcontroller) and a piece of software, or IDE (Integrated Development Environment) that runs on our computer, used to write and upload computer code to the physical board.

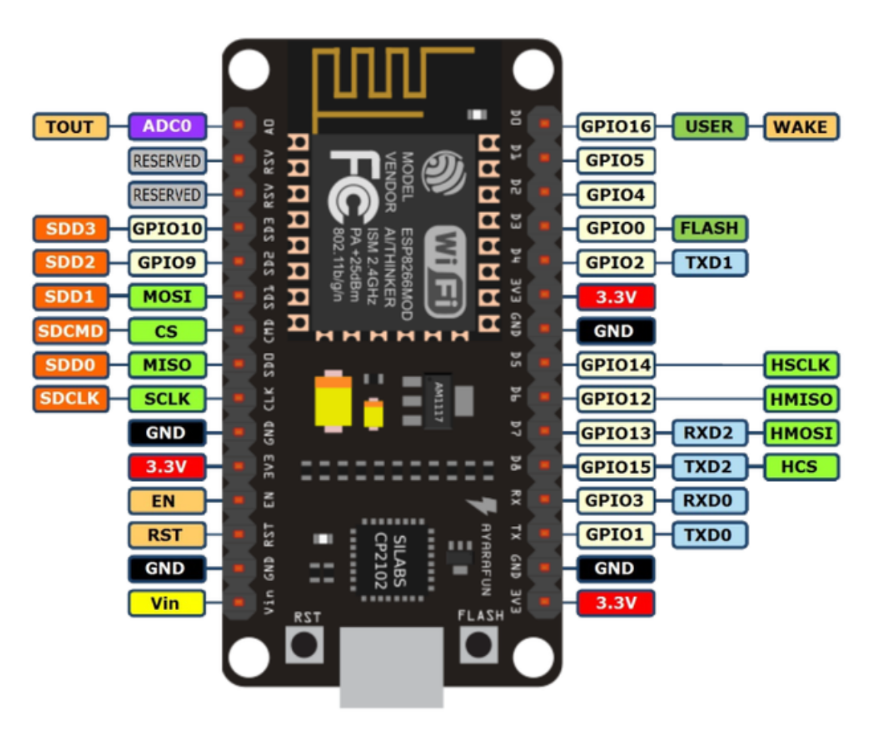
 Arduino board designs use a variety of microprocessors and controllers. The boards are equipped with sets of digital and analogue input/output (I/O) pins that may be interfaced to various expansion boards ('shields') or breadboards (For prototyping) and other circuits. The boards feature serial communications interfaces, including USB on some models, which are also used for loading programs from personal computers. The microcontrollers can be programmed using [C](https://en.wikipedia.org/wiki/C_(programming_language)) and C++ programming languages. In addition to using traditional compiler, the Arduino project provides an integrated development environment (IDE) based on the Processing language project.

Arduino Uno

The **Arduino Uno** is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc. The board has 14 digital I/O pins (six capable of output), 6 analogue I/O pins, and is programmable with the Arduino IDE (Integrated Development Environment), via a type B USB cable. The **ATmega328** is a single-chip microcontroller created by Atmel in the megaAVR family. It has a modified Harvard architecture 8-bit RISC processor core.

In this project we are using Arduino Uno as the microcontroller board. The input pins and signals will be explained later.

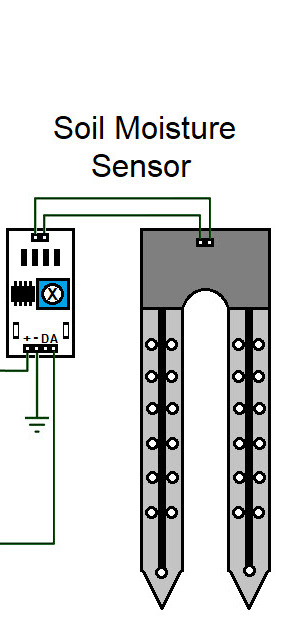
NodeMCU

 NodeMCU is an open-source firmware and development kit that helps you to prototype or build IoT product. It includes firmware which runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware which is based on the ESP-12E module.

NodeMCU is an open source firmware for which open source prototyping board designs are available. The name "NodeMCU" combines "node" and "MCU" (micro-controller unit). The term "NodeMCU" strictly speaking refers to the firmware rather than the associated development kits.

Soil Moisture Sensor

The moisture of the soil plays an essential role in the irrigation field as well as in gardens for plants. As nutrients in the soil provide the food to the plants for their growth. Supplying water to the plants is also essential to change the temperature of the plants. Plant root systems are also developed better when rising within moist soil. Extreme soil moisture levels can guide to anaerobic situations that can encourage the plant’s growth as well as soil pathogens.

 The soil moisture sensor is one kind of sensor used to gauge the volumetric content of water within the soil.

The FC-28 soil moisture sensor includes 4-pins

* VCC pin is used for power
* A0 pin is an analog output
* D0 pin is a digital output
* GND pin is a Ground

Working

The working of this sensor can be done by inserting this sensor into the earth and the status of the water content in the soil can be reported in the form of a percent. The Soil Moisture Sensor uses capacitance to measure dielectric permittivity of the surrounding medium. In soil, dielectric permittivity is a function of the water content. The sensor creates a voltage proportional to the dielectric permittivity, and therefore the water content of the soil. The sensor averages the water content over the entire length of the sensor. **Dielectric permittivity (ε)** is the ability of a substance to hold an electrical charge.

Program for Soil Moisture Sensor

int sensorPin = A0; // select the input pin for the potentiometer

int sensorValue = 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:

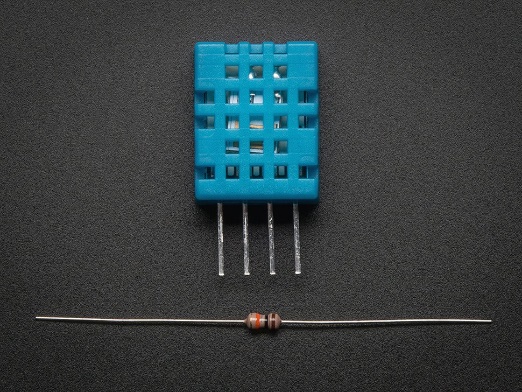
delay(1000);

Serial.print("sensor = " );

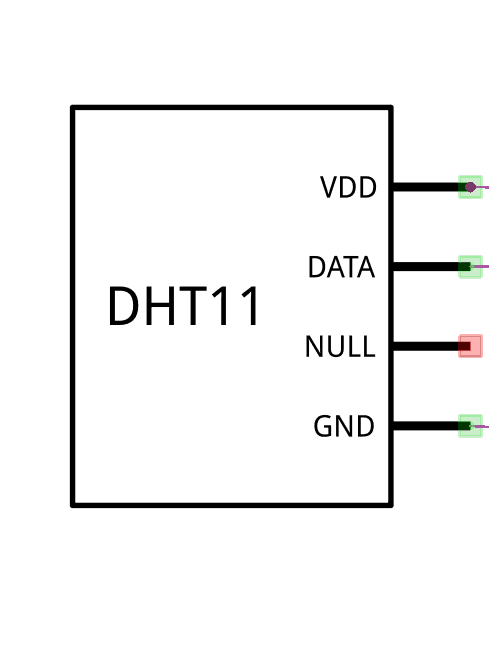
Serial.println(sensorValue);

}

DHT11 Sensor

 Humidity is the measure of water vapour present in the air. The level of humidity in air affects various physical, chemical and biological processes. DHT11 is a low-cost digital sensor for sensing temperature and humidity.  This sensor can be easily interfaced with any micro-controller such as Arduino, Raspberry Pi etc… to measure humidity and temperature instantaneously.  To measure the surrounding air this sensor uses a thermistor and a capacitive humidity sensor.

Working

DHT11 sensor consists of a capacitive humidity sensing element and a thermistor for sensing temperature.  The humidity sensing capacitor has two electrodes with a moisture holding substrate as a dielectric between them. Change in the capacitance value occurs with the change in humidity levels. The IC measure, process this changed resistance values and change them into digital form.

For measuring temperature this sensor uses a Negative Temperature coefficient thermistor, which causes a decrease in its resistance value with increase in temperature. To get larger resistance value even for the smallest change in temperature, this sensor is usually made up of semiconductor ceramics or polymers.

Program for humidity sensor

#include <dht.h>

#define dht\_dpin A0

dht DHT;

void setup()

{

Serial.begin(9600);

}

void loop()

{

DHT.read11(dht\_dpin);

Serial.print("Current humidity = ");

Serial.print(DHT.humidity);

Serial.print("% ");

Serial.print("temperature = ");

Serial.print(DHT.temperature);

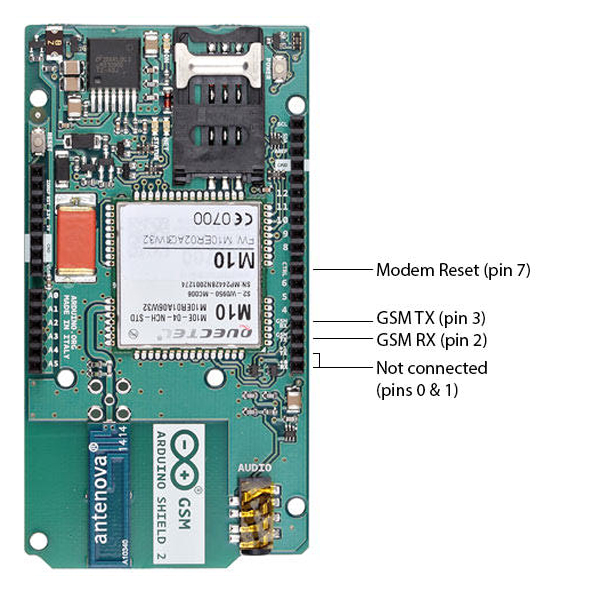
Serial.println("C ");

delay(1000);

}

GSM Shield

The **Arduino GSM Shield 2** allows an Arduino board to connect to the internet, make/receive voice calls and send/receive SMS messages. The shield uses a radio modem M10 by Quectel. It is possible to communicate with the board using AT commands. The GSM library has a large number of methods for communication with the shield.

To interface with the cellular network, the board requires a SIM card provided by a network operator.

 Requires an Arduino board

 Operating voltage 5V (supplied from the Arduino Board)

 Connection with Arduino Uno on pins 2, 3 (Software Serial) and 7 (reset).

 This version includes an integrated antenna

Since Using Arduino along with the GSM Shield is compalsory and more costly comparitively we use Node MCU Breakout board.

Cloud Computing

Cloud computing- an on-demand delivery of computing power, database storage, applications and IT resources. It enables organizations to consume a compute resource, like a virtual machine (VM) instead of building a computing infrastructure on premise. Cloud computing, as well as IoT, work towards increasing the efficiency of everyday tasks and both have a complementary relationship. On one hand, IoT generates lots of data while on the other hand, cloud computing paves way for this data to travel.

Why is cloud computing essential for IoT?

* Provides remote processing power
* Provides security and privacy
* Facilitates inter-device communication

Examples:

* Telegram
* Blynk
* IFTTT
* Adafruit.io

Telegram

**Telegram** is a cloud-based instant messaging and voice over IP service. Telegram client apps are available for Android, iOS, Windows Phone, Windows NT, macOS and Linux. Users can send messages and exchange photos, videos, stickers, audio and files of any type. Telegram's client-side code is open-source software.

Default messages and media in Telegram are encrypted when stored on its servers, but can be accessed by the Telegram service provider, who holds the encryption keys.

Telegram's default messages are cloud-based and can be accessed on any of the user's connected devices. Users can share photos, videos, audio messages and other files (up to 1.5 gigabyte in size per file). Users can send messages to other users individually or to groups of up to 200,000 members. Telegram cloud messages and media remain on the servers at least until deleted by all participants.

Bots

Bots are Telegram accounts operated by programs. They can respond to messages or mentions, can be invited into groups and can be integrated into other programs. Also there are inline bots, which can be used from any chat screen. In order to activate an inline bot, user needs to type in the message field a bot's username and query. The bot then will offer its content. User can choose from that content and send it within a chat.

Blynk

Blynk is a cloud service. Blynk was designed for the Internet of Things. It can control hardware remotely, it can display sensor data, it can store data, visualize it and do many other cool things.

There are three major components in the platform:

* **Blynk App** - allows to you create amazing interfaces for your projects using various widgets we provide.
* **Blynk Server** - responsible for all the communications between the smartphone and hardware. We can use Blynk Cloud or run [private Blynk server](https://docs.blynk.cc/#blynk-server) locally. Its open-source, could easily handle thousands of devices and can even be launched on a Raspberry Pi.
* **Blynk Libraries** - for all the popular hardware platforms - enable communication with the server and process all the incoming and outcoming commands.

Features

* Similar API & UI for all supported hardware & devices
* Connection to the cloud using:
  + WiFi
  + Ethernet
  + USB (Serial)
  + GSM
* Set of easy-to-use Widgets
* Easy to integrate and add new functionality using virtual pins
* Device-to-Device communication using Bridge Widget

**Here in this project I am using Blynk A pp for the machine to machine communication.**

Arduino IDE

The **Arduino Integrated Development Environment (IDE)** is a cross-platform application (for Windows, macOS, Linux) that is written in functions from C and C++. It is used to write and upload programs to Arduino compatible boards, but also, with the help of 3rd party cores, other vendor development boards. The Arduino IDE supports the languages C and C++ using special rules of code structuring.

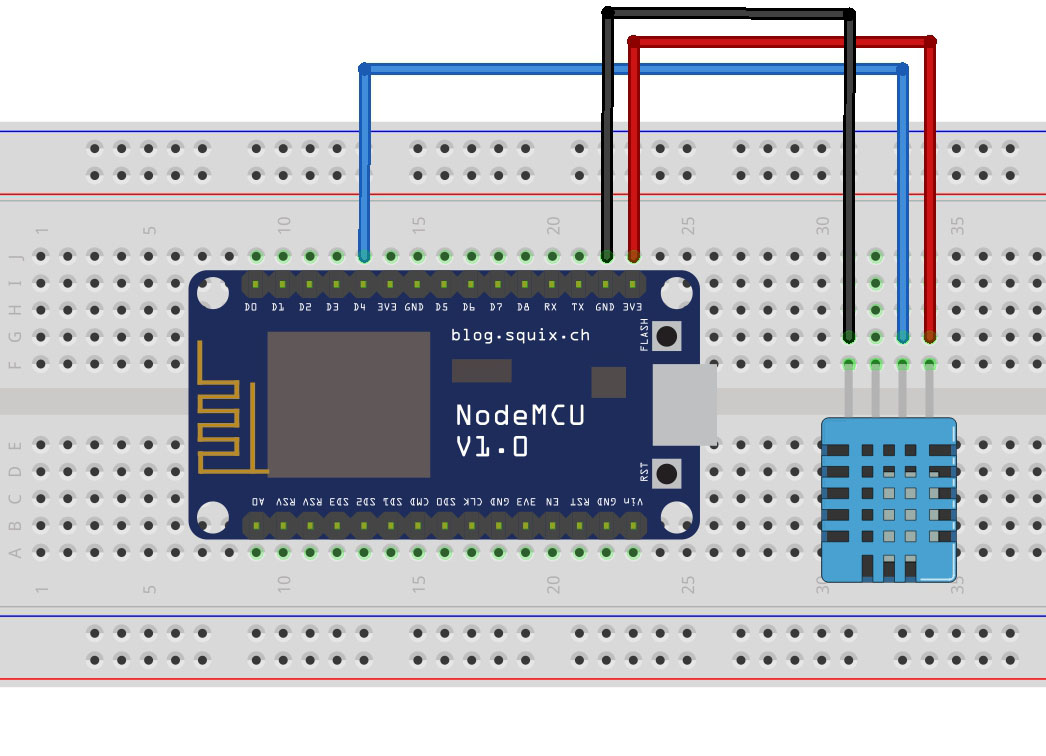


Agri Bot Connection

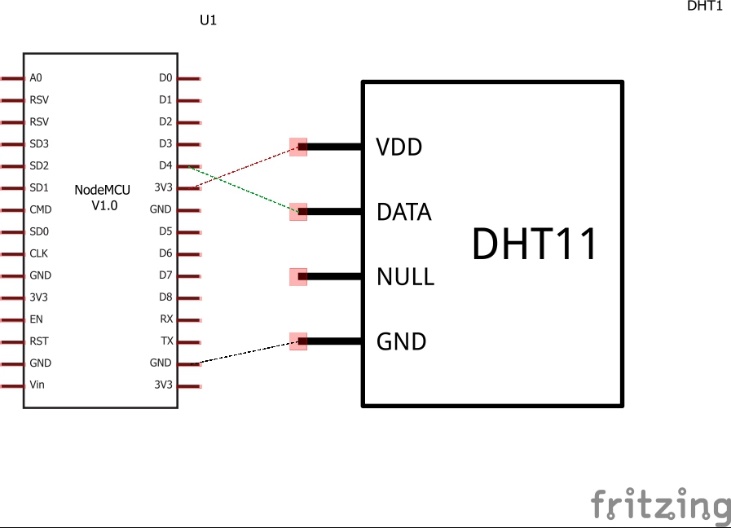
Total connect of the agri bot can be divided into two parts in which first one contains DHT11 sensor and the second one contains Soil Moisture Sensor.

DHT11 - Blynk Connection

To find humidity and temperature of the field



Circuit



Schematic Diagram

Code for DHT sensor :

#define BLYNK\_PRINT Serial // Comment this out to disable prints and save space

#include <SPI.h>

#include <ESP8266WiFi.h>

#include <BlynkSimpleEsp8266.h>

#include <SimpleTimer.h>

#include <DHT.h>

// You should get Auth Token in the Blynk App.

char auth[] = " "; //Enter the Auth code which was send by Blink

// Your WiFi credentials.

// Set password to "" for open networks.

char ssid[] = " "; //Enter your WIFI Name

char pass[] = " "; //Enter your WIFI Password

#define DHTPIN 2 // Digital pin 4

// Uncomment whatever type you're using!

#define DHTTYPE DHT11 // DHT 11

//#define DHTTYPE DHT22 // DHT 22, AM2302, AM2321

//#define DHTTYPE DHT21 // DHT 21, AM2301

DHT dht(DHTPIN, DHTTYPE);

SimpleTimer timer;

// This function sends Arduino's up time every second to Virtual Pin (5).

// In the app, Widget's reading frequency should be set to PUSH. This means

// that you define how often to send data to Blynk App.

void sendSensor()

{

float h = dht.readHumidity();

float t = dht.readTemperature(); // or dht.readTemperature(true) for Fahrenheit

if (isnan(h) || isnan(t)) {

Serial.println("Failed to read from DHT sensor!");

return;

}

// You can send any value at any time.

// Please don't send more that 10 values per second.

Blynk.virtualWrite(V5, h); //V5 is for Humidity

Blynk.virtualWrite(V6, t); //V6 is for Temperature

}

void setup()

{

Serial.begin(9600); // See the connection status in Serial Monitor

Blynk.begin(auth, ssid, pass);

dht.begin();

// Setup a function to be called every second

timer.setInterval(1000L, sendSensor);

}

void loop()

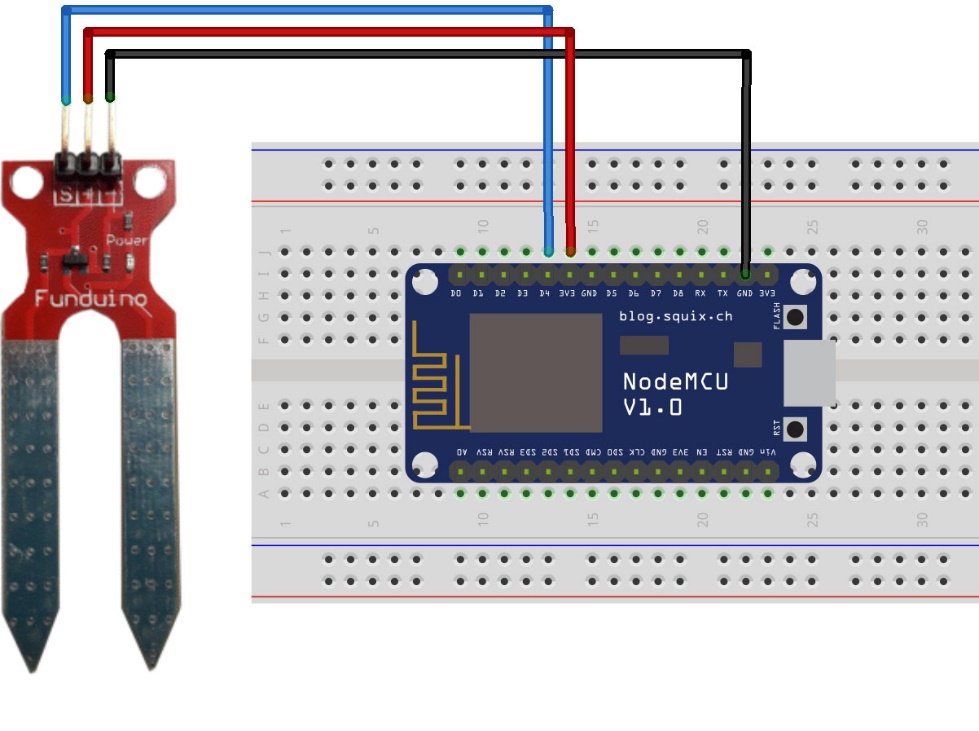
{

Blynk.run(); // Initiates Blynk

timer.run(); // Initiates SimpleTimer

}

Soil Moisture – Blynk Connection



Circuit

Code for Soil moisture Sensor :

#define BLYNK\_PRINT Serial

#include <ESP8266WiFi.h>

#include <BlynkSimpleEsp8266.h>

char auth[] = "90f6fbee96074a519729796ee8673186"; //code sent via email

const int sensorPin = 4;

int sensorState = 0;

int lastState = 0;

void setup()

{

Serial.begin(9600);

Blynk.begin(auth, "Hacked", "0123456789"); //wifi name and password

pinMode(sensorPin, INPUT);

}

void loop()

{

Blynk.run();

sensorState = digitalRead(sensorPin);

Serial.println(sensorState);

if (sensorState == 1 && lastState == 0) {

Serial.println("needs water, send notification");

Blynk.notify("Water your plants");

lastState = 1;

delay(1000);

//send notification

}

else if (sensorState == 1 && lastState == 1) {

//do nothing, has not been watered yet

Serial.println("has not been watered yet");

delay(1000);

}

else {

//st

Serial.println("does not need water");

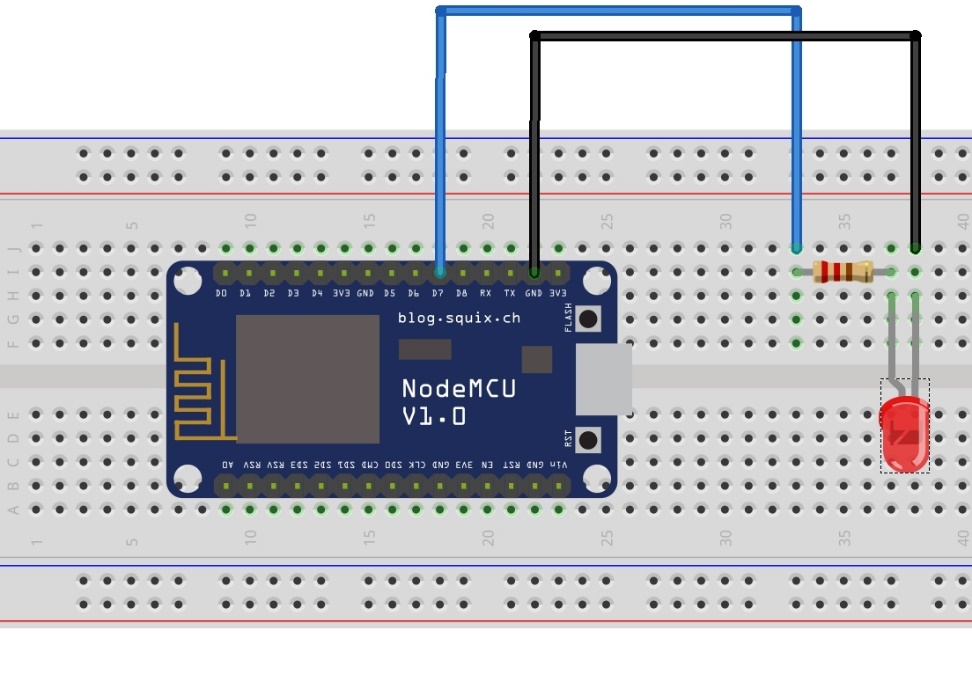
lastState = 0;

delay(1000);

}

delay(100);

}

LED ( Water Pump) – Blynk Connection

**Code For LED Connection :**

#define BLYNK\_PRINT Serial

#include <ESP8266WiFi.h>

#include <BlynkSimpleEsp8266.h>

// You should get Auth Token in the Blynk App.

// Go to the Project Settings (nut icon).

char auth[] = "YourAuthToken";

// Your WiFi credentials.

// Set password to "" for open networks.

char ssid[] = "YourNetworkName";

char pass[] = "YourPassword";

WidgetLED led1(V1);

BlynkTimer timer;

// V1 LED Widget is blinking

void blinkLedWidget()

{

if (led1.getValue()) {

led1.off();

Serial.println("LED on V1: off");

} else {

led1.on();

Serial.println("LED on V1: on");

}

}

void setup()

{

// Debug console

Serial.begin(9600);

Blynk.begin(auth, ssid, pass);

// You can also specify server:

//Blynk.begin(auth, ssid, pass, "blynk-cloud.com", 80);

//Blynk.begin(auth, ssid, pass, IPAddress(192,168,1,100), 8080);

timer.setInterval(1000L, blinkLedWidget);

}

void loop()

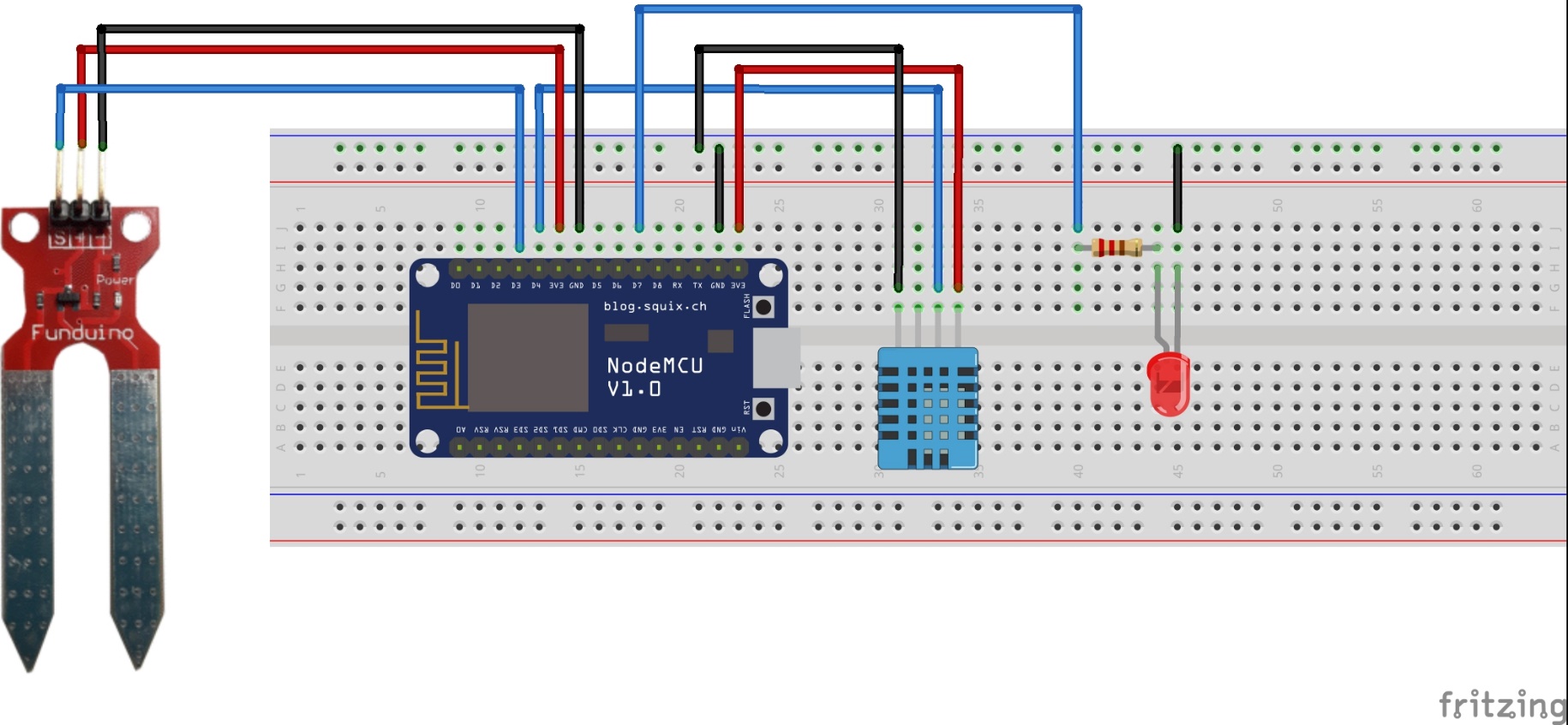
{

Blynk.run();

timer.run();

}

Agribot Connection



**Code For Agri Bot :**

#define BLYNK\_PRINT Serial

#include <ESP8266WiFi.h>

CONCLUSION

Proposed system can improve the efficiency of scientific and agricultural practices. It works as expected and the data obtained was noted and we could enhance system further by adding more sensors and using a motor as a simulation for a pump. We showed that it can be used to automatically irrigate the fields once the soil moisture goes down below the threshold value. Loss and degradation of crops can be minimized.

References

* Kumar, Abhijeet, and Apoorva Sharma. "Internet of Life (IOL)." (2015)
* <https://www.hackster.io>
* <https://github.com/>
* <http://www.circuitstoday.com/>
* <https://www.arduino.cc/>
* <https://www.elprocus.com/>
* <http://ijesc.org/>
* <https://blynk.io/>