



Amrita Vishwa Vidyapeetham
Centre for Excellence in Computational Engineering and Networking
Amrita School of Engineering, Coimbatore

ARDUINO RAINWATER ALARM

Prepared By:

Aakash J.V.V
Rakhil M.L
R.Hema Radhika
Thanga Rohan

Supervised By:

Prof. Ambeka P S
Asst. Professor

An End Semester Project submitted to the **Centre for Excellence in Computational Engineering and Networking** in partial fulfillment of the requirements for the degree of B.tech in **Computer Engineering**
"Artificial Engineering"

July, 2022

Acknowledgment

We are deeply thankful to **Center for Excellence in Computational Engineering and Networking (CEN) at Amrita Vishwa Vidyapeetham, Coimbatore** for providing us such a wonderful environment to peruse our research. We would like to express our sincere gratitude to **Prof.Ambika P S , Asst. Professor, Department of Center for Excellence in CEN, Amrita Vishwa Vidyapeetham.** We have completed our research under her guidance. We found the research area, topic, and problem with her suggestions. She guided us with our study, and supplied us many research papers and academic resources in this area. She is patient and responsible. When we had questions and needed her help, she would always find time to meet and discuss with us no matter how busy she was.

We would also like to extend our gratitude to **Dr Vinod, Asst.Professor, Department of Center for Excellence in CEN, Amrita Vishwa Vidyapeetham,** who provided us with the base knowledge of this field of research, without him this research work would not exist in its present form. We would also like to acknowledge our team members for supporting each other and be grateful to our university for providing this opportunity for us. Lastly special thanks to Center for Excellence in CEN for providing this opportunity to research in this field.

Declaration

We do hereby declare that the research works presented in this project entitled, “**ARDUINO RAINWATER ALARM**” are the results of our own works. We further declare that the project has been compiled and written by us under the guidance of our supervisor. The materials that are obtained from other sources are duly acknowledged in this project.

Authors

Aakash J.V.V
ID: CB.EN.U4AIE21019

RAKHIL M.L
ID: CB.EN.U4AIE21048

R.Hema Radhika
ID:CB.EN.U4AIE21050

Thanga Rohan
ID: CB.EN.U4AIE21069

Table of Contents

Acknowledgement	I
Declaration	II
Table of Contents	III
Abstract	V
List of Tables	VI
List of Figures	VII
1 Introduction	1
1.1 Introduction	1
1.2 objective	1
2 Hardware and Software requirement	2
3 Theory	3
3.1 Theory	3
4 Pros and cons	4
5 Design	5
6 Output	6
7 Code	9

8 Conclusion and Future Work	11
Bibliography	12

Abstract

Rain Alarm Project is a simple but very useful project that detects Rain(Rain Water) and automatically triggers an alarm or buzzer. Water is a basic need in every ones life. Saving water and proper usage of water is very important. Here is an easy project which will give the alarm when there is rainfall, so that we can make some actions for rain water harvesting and also save therain water for using it later. Here, a low-cost, large coverage and user-friendly water quality monitoring system with multi-sensor which is based on Wi-Fi Technology is presented. The process of designing the system involves three main parts which are hardware development, software development and also wireless network process. For the hardware development part, different types of the sensor will be employed and the sensors will be integrated with a microcontroller. As for the software development, a few programming software is integrated for data acquisition, processing and displaying. Finally, for the wireless network process, the wireless network module is used for data transmitting and receiving while the website is used to display the obtained data. With the help of saving this rain water through rain water harvesting, we can increase the levels of underground water by using underwater recharge technique.

List of Tables

2.1	Hardware Required	2
-----	-----------------------------	---

List of Figures

1.1	Rain Water Alarm	1
3.1	wifi module	3
5.1	Hardware Setup	5
6.1	Gmail notification	6
6.2	system off interface in arduino software	7
6.3	system off interface in phone	8
7.1	code1	9
7.2	code2	10
7.3	code3	10

Chapter 1

Introduction

Contents

1.1	Introduction	1
1.2	objective	1

1.1 Introduction

Rain is key elements in the Earth's water cycle, which is vital to all life on Earth and Rain will it is vital for the human beings to survive,it can be a double edge sword as it can go from spoiling you laundry to destroying the yield of the farmer.

1.2 objective

Keeping these in mind we have build these project " Rain Water Alarm". These project is to help you detect the rain and inform the user.ideally,when you are away from home,you can get notified if it is raining in your locality or help the farmer to be informed if it is raining.



Figure 1.1: Rain Water Alarm

Chapter 2

Hardware and Software requirement

Hardware: The Components required for these project are:

S.no	Name	Description
1	Rain sensor	Raindrops module consists of two boards, namely Rain Board and Control Board.
2	Buzzer	Buzzers are electric sounding devices that generate sounds. Typically powered by DC voltage
3	Breadboard	Breadboards are temporary work boards for electronic circuits.
4	espn-8266	ESP8266 is a cost-effective and highly integrated Wi-Fi MCU for IoT applications.
5	led	A light-emitting diode (LED) is a semiconductor light source that emits light when current flows through it
6	Connecting wires	jumper wires are used for making connections between items on your breadboard and your Arduino's header pins

Table 2.1: Hardware Required

Software: The software requirement:

1.*Arduino Software:*

<https://www.arduino.cc/en/software>

2.*blynk app:* =

https://play.google.com/store/apps/details?id=cloud.blynk&hl=en_IN&gl=US

Chapter 3

Theory

Contents

3.1 Theory	3
----------------------	---

3.1 Theory

These project is to build a Rain Water Alarm. Our implementation for building the alarm for detecting Rain water which includes the buzzer which start to buzz and led lights up to indicate it is raining and upon that it sends a notification to your phone and we can control (stop) the buzzer using WiFi module.

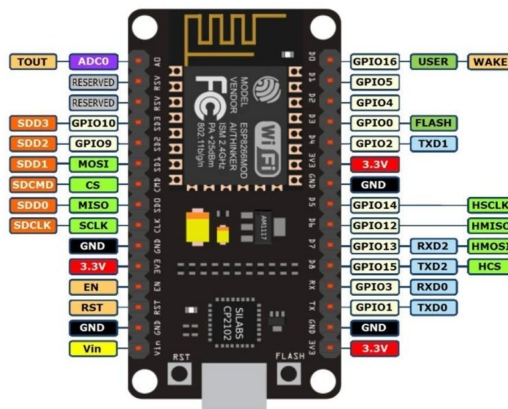


Figure 3.1: wifi module

Chapter 4

Pros and cons

RAIN SENSOR PROS:

Save you water. Rain Sensors will help you water more efficiently and save water. Conserving water is especially useful with city water. Save your time. Rain sensors help busy homeowners that don't have time to think about their irrigation. Save your money. Watering less will usually save money.

RAIN SENSOR CONS:

Rain sensors occasionally need to be serviced. Batteries should be changed once per year and occasionally the rain sensor will stop working and need to be checked. Sometimes rain sensors will cause the homeowner to under water unless set perfectly. At times you will see that the sprinklers are running while it is raining. The rain sensor will not turn on until enough rain has accumulated and then the next sprinkler cycle will be turned off.

Chapter 5

Design

Hardware Setup:

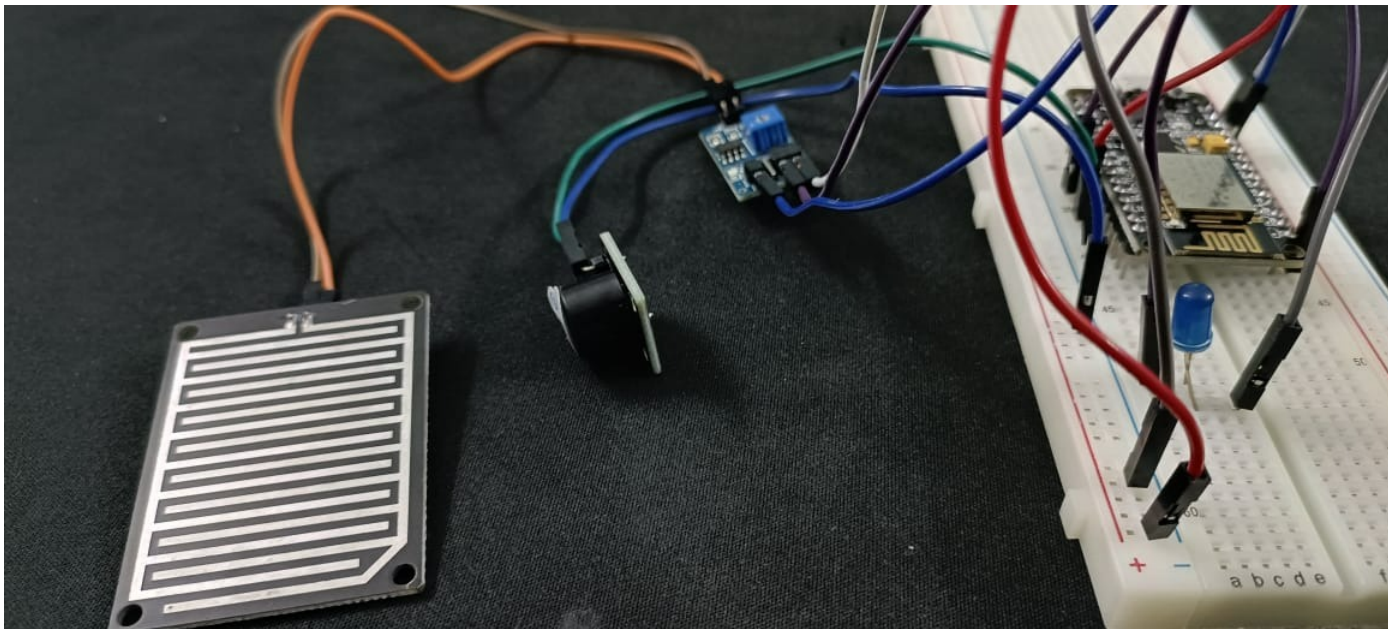


Figure 5.1: Hardware Setup

connection:

Connect BUZZER – D3 – GND

Connect LED – D7 – GND

connect Rain Sensor -- A0 – Vin – GND

Chapter 6

Output

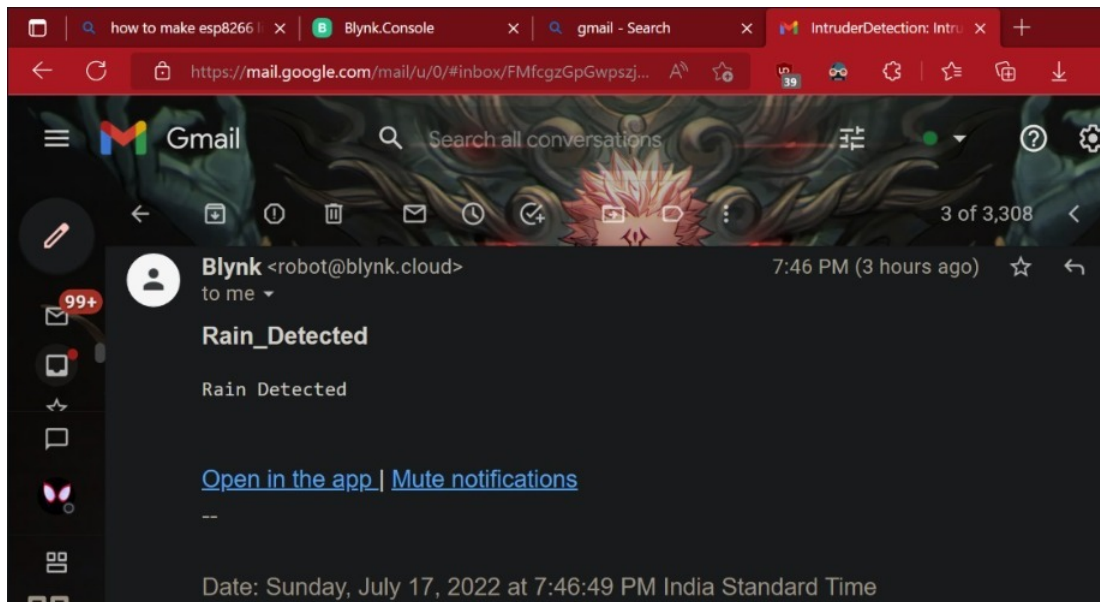


Figure 6.1: Gmail notification

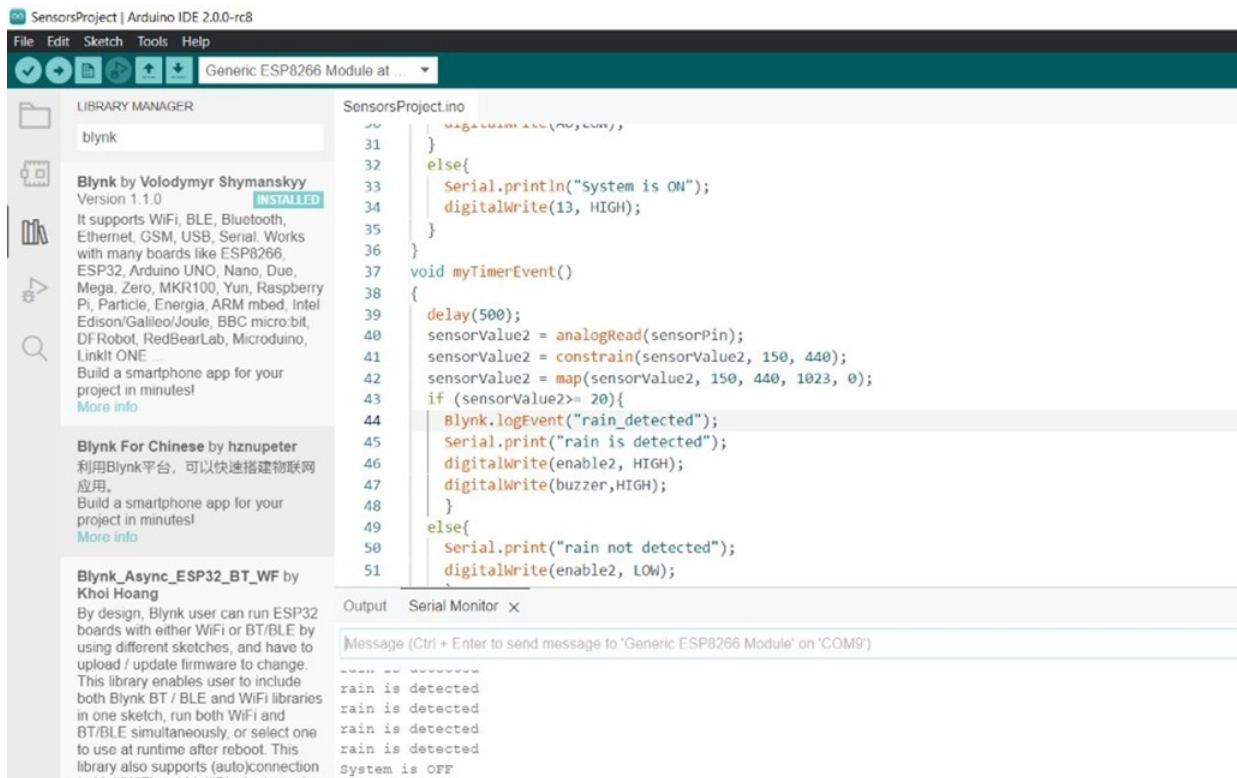
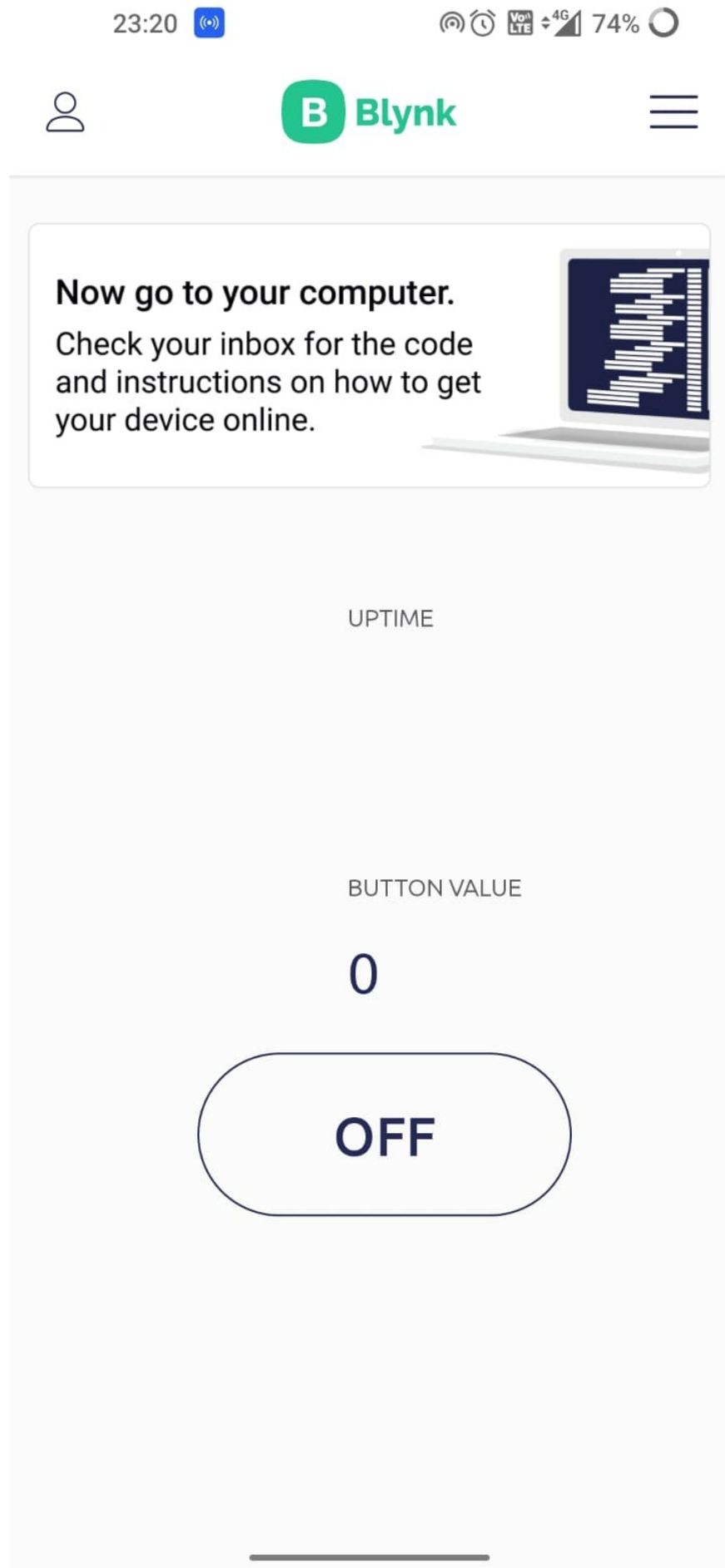


Figure 6.2: system off interface in arduino software



8
Figure 6.3: system off interface in phone

Chapter 7

Code

The Code for the RAIN WATER ALARM:

```
RainSensor

#define BLYNK_TEMPLATE_ID          "TMPLWxqMdEOu"
#define BLYNK_DEVICE_NAME          "RAIN SENSING"
#define BLYNK_AUTH_TOKEN           "IS-HATW00LUfO71MNFnQb5LZ7sLH4Ctj"
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
char auth[] = BLYNK_AUTH_TOKEN;
char ssid[] = "OnePlus8";
char pass[] = "haha@6996";
int pinValue = 0;
int sensorPin = A0;
int enable2 = 13;      // enable reading Rain sensor
int sensorValue2 = 0;  // variable to store the value coming from sensor Rain sensor
int buzzer=0;

BlynkTimer timer;

// This function is called every time the Virtual Pin 0 state changes
BLYNK_WRITE(V0)
{
    // Set incoming value from pin V0 to a variable
    int value = param.asInt();

    // Update state
    Blynk.virtualWrite(V1, value);
    if(value==0){
        Serial.println("System is OFF");
        digitalWrite(0, LOW);
        digitalWrite(13,LOW);
        digitalWrite(A0,LOW);
    }
}
```

Figure 7.1: code1

```

    else{
        Serial.println("System is ON");
        digitalWrite(13, HIGH);
    }
}
void myTimerEvent()
{
    delay(500);
    sensorValue2 = analogRead(A0);
    sensorValue2 = constrain(sensorValue2, 150, 440);
    sensorValue2 = map(sensorValue2, 150, 440, 1023, 0);
    if (sensorValue2 >= 20) {
        Blynk.logEvent("rain_detected");
        Serial.print("rain is detected");
        digitalWrite(enable2, HIGH);
        digitalWrite(buzzer, HIGH);
    }
    else{
        Serial.print("rain not detected");
        digitalWrite(enable2, LOW);
    }
    //Serial.print("Rain value:      ");
    //Serial.println(sensorValue2);
    Serial.println();
    delay(100);
}

```

Figure 7.2: code2

```

void setup()
{
    // Debug console
    Serial.begin(115200);
    pinMode(enable2, OUTPUT);
    Blynk.begin(auth, ssid, pass);
    timer.setInterval(1000L, myTimerEvent);
}

void loop()
{
    Blynk.run();
    timer.run();
}

```

Figure 7.3: code3

Chapter 8

Conclusion and Future Work

While we tried to explain as easy as possible and tried to make the project easy and compact, there are many possible way to improve the current project which we will try to update it before gets outdated . We apologize for any errors you find in this paper. you can freely distribute or use the code as long as it is an open source project.

Finally, We hope this paper is useful for you and we hope you try the following projects. you can pin us with your interpretation of the project.

Bibliography

Blynk:

https://play.google.com/store/apps/details?id=cloud.blynk&hl=en_IN&gl=US

Arduino:

[arduino.cc](https://www.arduino.cc)