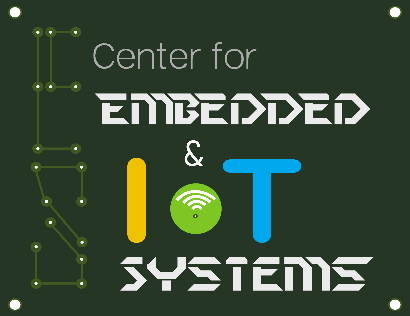
# GODOWN PHYSICAL PARAMETERS CONTROLLING SYSTEM



A project report submitted in partial fulfilment of requirement for the course

On

## Internet of Things

By

**G.ASHMITHA (2103A51512)**

**M.DEEKSHITH REDDY (2103A51185)**

**B.BHANUSRI (2103A51355)**

**P.SAI PAVANI (2103A51406)**

Undertheguidanceof

## Dr.Sumit Gupta

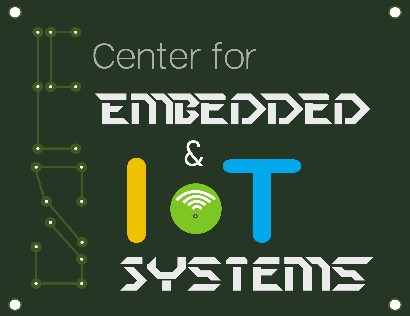
Asst.Prof. ,Centre for Embedded Systems and IoT

Department of Computer science and Engineering



**Center for Embedded Systems and Internet of things**

**SR UNIVERSITY**



**CERTIFICATE**

This is to certify that the course project entitled **“GODOWN PHYSICAL PARAMETERS CONTROLLING SYSTEM”**is the bonafied work carried out by **G.ASHMITHA** (2103A51512), **M.DEEKSHITH REDDYP** (2103A51328) , **B.BHANUSRI** (2103A51355) and **P.SAI PAVANI** (2103A51406)in the partial fulfilment of the requirement for the award of course IoT during the academic year 2023-2024 under our guidance and Supervision.

**Dr.Sumit Gupta**

Asst.Prof , CentreforEmbedded SystemsandIoT,

DepartmentofECE

# ABSTRACT

As food is the basic component for the survival we need to store it cleanly and according to the grain requirements. As we store some grains like rice, wheat, etc …. in the godown to use it for the future purpose. We need to maintain it correctly. In order to use them it should be good without getting bacteria due to humidity and dried due to temperature ,it should be in the perfect needs, like the grain suitable temperature, humidity, and the moisture .If the grains contain some moisture it should be dried using some techniques. We use some techniques to make all these requirements satisfied by doing this project.

**CONTENTS**

**Chapter No. Title Page No.**

**1 Introduction 1**

**2 System Description 2**

**2.1 Block Diagram 3**

**3 Hardware & Software Tools 5-8**

**4 Implementation 9-12**

**5 Results 13**

**6 Conclusion 14**

**7 References 15**

# INTRODUCTION

A godown is a holding area that is used to store items. Warehouses are generally used in ecommerce to hold goods in stock so that the fast shipping periods needed can be reached. Agriculture, the cornerstone of the Indian economy, contributes to the country's overall economic development and influences the standard of living for more than half of the population. With 179.9 million hectares under cultivation, India is the world's second largest agricultural region. In the global scenario, the country has emerged as a major player in agriculture.

The food grains should be stored in the food requirements as the wheat - temperature(10C-20C), humidity(65%-75%), moisture(12%-14%) and the rice - temperature(10C-15C), humidity(60%-70%), moisture(12%-13%) , maize - temperature(10C-21C), humidity(60%-70%),moisture(13%-14%),corn - temperature(15C-20C),humidity(65%-70%),moisture(13%-15%) as this grains need certain parameters it should be stored in the above conditions to use in the future .If we store it in this conditions the grains will not be spoiled after the months also it will be fresh and good.



# SYSTEM DESCRIPTION

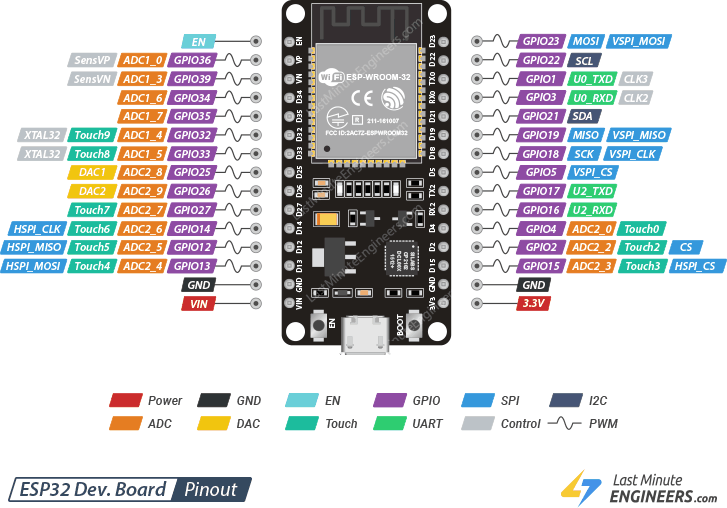
The below block diagram is about Godown physical parameters controlling system

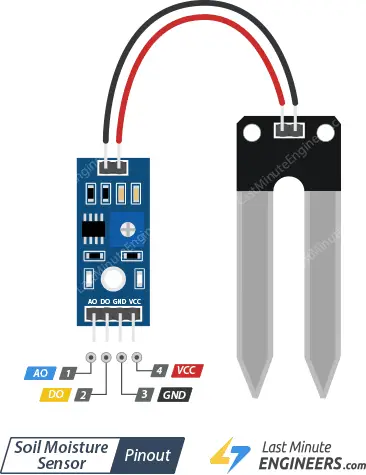
using IOT. In this system we are monitoring physical conditions of a grains. All

the sensors are interfaced to ESP32. The sensors used in this project are DHT sensor,

Moisture sensor and software tools are Arduino IDE, ThingSpeak.

The ESP32 sensor network is a wifi/Bluetooth based wireless sensor network board that can connect with a variety of sensors DHT sensor is used for measuring the temperature and humidity of the grains in the godown. Moisture sensor moniters the moisture in the grains which are stored in the godown. Data from sensors are uploaded to the cloud.





**2.1-BLOCK DIAGRAM:**

MOISTURE SENSOR

THINKSPEAK

CLOUD

ESP 32

DHT SENSOR

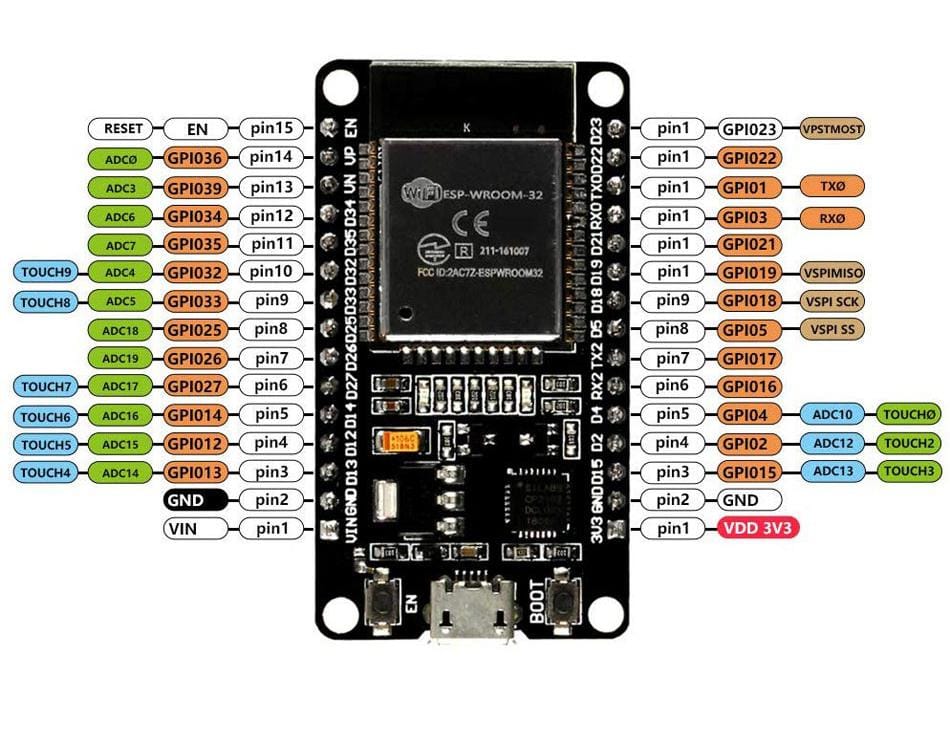
DC MOTORS

# Hardware & Software Tools

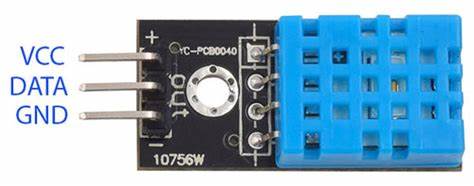
**Hardware Tools:**

**1.ESP32**

ESP32 is the name of the chip that was developed by Espressif Systems. This provides Wi-Fi dual-mode Bluetooth connectivity to embedded devices. While ESP32is technically just the chip, modules and development boards that contain this chip are often also referred to as “ESP32” by the manufacturer.Totally it has 36 GPIO (General purpose input output) pins. It also has 18 channels of 12-bit ADC (Analog to digital converters), Two channels for 8-bit DAC (Digital to analog converters). With an output voltage of 3.3v and 5.0v. ESP32 can interface with other systems to provide Wi-Fi and Bluetooth functionality through its SPI / SDIO or I2C / UART interfaces.



**2.DHT sensor**



DHT11–Temperature and Humidity Sensor

DHT11 Sensor Pinout

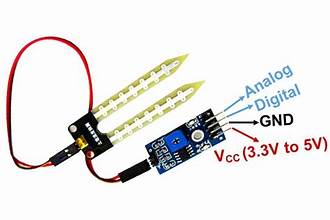
The DHT11 is a commonly used Temperature and humidity sensor that comes with a dedicated NTC to measure temperature and an 8-bit microcontroller to output the values of temperature and humidity as serial data. There are three pins in this sensor . They are VCC,GND,DATA in the sensor VCC is mentioned as “+”,GND is mentioned as “-” , and the data pin is mentioned as “OUT”.

**3.Soil Moisture Sensor:**

Soil moisture sensors measure the volumetric [water content](https://en.wikipedia.org/wiki/Water_content) in [soil](https://en.wikipedia.org/wiki/Soil).[[1]](https://en.wikipedia.org/wiki/Soil_moisture_sensor#cite_note-1) Since the direct [gravimetric measurement](https://en.wikipedia.org/wiki/Gravimetric_analysis) of free soil moisture requires removing, drying, and weighing of a sample, soil moisture sensors measure the volumetric water content indirectly by using some other property of the soil, such as electrical resistance, dielectric constant, or interaction with [neutrons](https://en.wikipedia.org/wiki/Neutron), as a proxy for the moisture content.

The relation between the measured property and soil moisture must be calibrated and may vary depending on environmental factors such as [soil type](https://en.wikipedia.org/wiki/Soil_type), [temperature](https://en.wikipedia.org/wiki/Temperature), or [electric conductivity](https://en.wikipedia.org/wiki/Electric_conductivity). Reflected [microwave](https://en.wikipedia.org/wiki/Microwave) radiation is affected by the soil moisture and is used for [remote sensing](https://en.wikipedia.org/wiki/Remote_sensing) in [hydrology](https://en.wikipedia.org/wiki/Hydrology) and agriculture. Portable probe instruments can be used by farmers or gardeners.

Soil moisture sensors typically refer to sensors that estimate volumetric water content. Another class of sensors measure another property of moisture in soils called [water potential](https://en.wikipedia.org/wiki/Water_potential); these sensors are usually referred to as soil water potential sensors and include [tensiometers](https://en.wikipedia.org/wiki/Tensiometer_(soil_science)) and gypsum blocks.

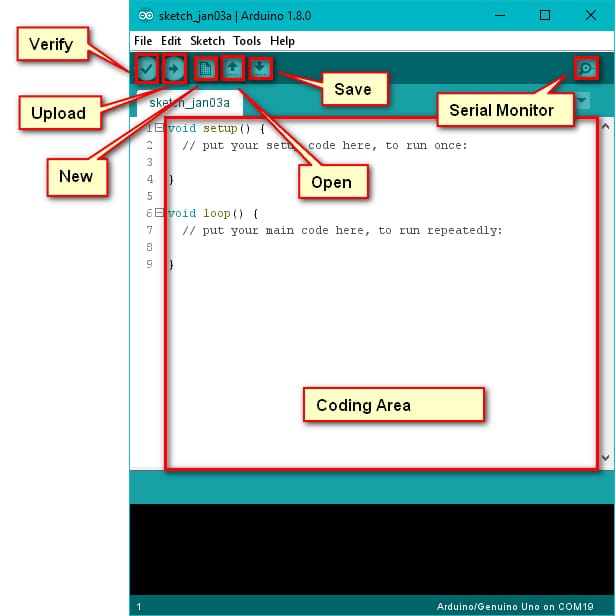


**Software Tools:**

**1.Arduino Ide**

The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino hardware to upload programs and communicate with them.

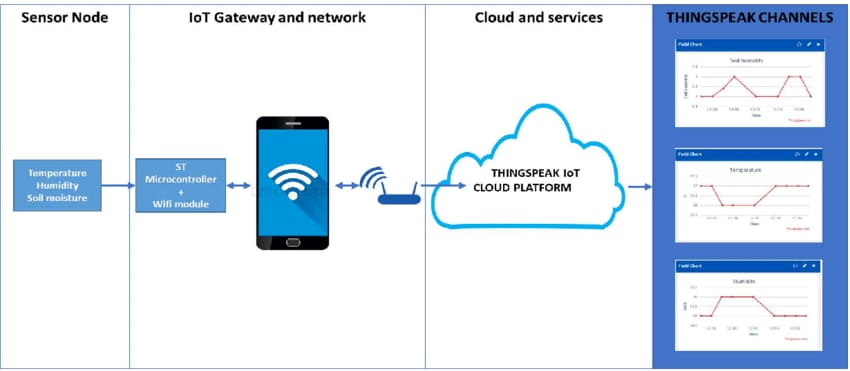
****



**2.ThingSpeak**

ThingSpeak™ is an IoT analytics platform service that allows you to aggregate, visualize and analyze live data streams in the cloud. ThingSpeak provides instant visualizations of data posted by your devices to ThingSpeak.Additional commands are found within the five menus: File, Edit, Sketch, Tools, Help. The menus are context sensitive, which means only those items relevant to the work currently being carried out are available.





# 4.IMPLEMENTATION

# // Include necessary libraries

# #include <DHT.h>

# #include <DHT\_U.h>

# #include <WiFi.h>

# #include <HTTPClient.h>

# #define DHTPIN 2

# #define DHTTYPE DHT11

# DHT dht(DHTPIN, DHTTYPE);

# // Set our wifi name and password

# const char\* ssid = "IOT";

# const char\* password = "Apple@123";

# // Your thingspeak channel url with api\_key query

# String serverName = "https://api.thingspeak.com/update?api\_key=PWQUV91Z4CX1G6XP";

# // Moisture sensor pin

# int moisturePin = 5;

# // Define thresholds for rice and wheat storage

# float wheatTempMin = 10;

# float wheatTempMax = 20;

# float wheatHumidityMin = 65;

# float wheatHumidityMax = 75;

# int wheatMoistureMin = 12;

# int wheatMoistureMax = 14;

# float riceTempMin = 10;

# float riceTempMax = 15;

# float riceHumidityMin = 60;

# float riceHumidityMax = 70;

# int riceMoistureMin = 12;

# int riceMoistureMax = 13;

# void setup() {

# Serial.begin(9600);

# dht.begin();

# pinMode(moisturePin, INPUT);

# WiFi.begin(ssid, password);

# Serial.println("Connecting");

# while(WiFi.status() != WL\_CONNECTED) {

# delay(500);

# Serial.print(".");

# }

# Serial.println("");

# Serial.print("Connected to WiFi network with IP Address: ");

# Serial.println(WiFi.localIP());

# }

# void loop() {

# float h = dht.readHumidity();

# float t = dht.readTemperature();

# int moistureValue = digitalRead(moisturePin);

# sendData(t, h, moistureValue);

# Serial.print(F("Humidity "));

# Serial.print(h);

# Serial.print(F("% Temperature "));

# Serial.print(t);

# Serial.print(" Moisture Value: ");

# Serial.println(moistureValue);

# // Check conditions for rice storage

# if (t >= riceTempMin && t <= riceTempMax && h >= riceHumidityMin && h <= riceHumidityMax && moistureValue >= riceMoistureMin && moistureValue <= riceMoistureMax) {

# // Control one DC motor for rice storage

# // Code to turn on one DC motor

# digitalWrite(4,HIGH);

# Serial.println("Rice storage conditions met. Turning on one DC motor.");

# }

# else

# {

# digitalWrite(4,LOW);

# Serial.println("Rice storage conditions not met. Turning off one DC motor.");

# }

# // Check conditions for wheat storage

# if (t >= wheatTempMin && t <= wheatTempMax && h >= wheatHumidityMin && h <= wheatHumidityMax && moistureValue >= wheatMoistureMin && moistureValue <= wheatMoistureMax) {

# // Control two DC motors for wheat storage

# // Code to turn on two DC motors

# digitalWrite(4,HIGH);

# digitalWrite(5,HIGH);

# Serial.println("Wheat storage conditions met. Turning on two DC motors.");

# }

# else

# {

# digitalWrite(4,LOW);

# digitalWrite(5,LOW);

# Serial.println("Wheat storage conditions not met. Turning off two DC motor.");

# }

# delay(60000); // Wait for a minute before taking the next reading

# }

# void sendData(float t, float h, int moistureValue) {

# HTTPClient http;

# String url = serverName + "&field1=" + h + "&field2=" + t + "&field3=" + moistureValue;

# http.begin(url.c\_str());

# int httpResponseCode = http.GET();

# if (httpResponseCode > 0) {

# Serial.print("HTTP Response code: ");

# Serial.println(httpResponseCode);

# } else {

# Serial.print("Error code: ");

# Serial.println(httpResponseCode);

# }

# http.end();

# }

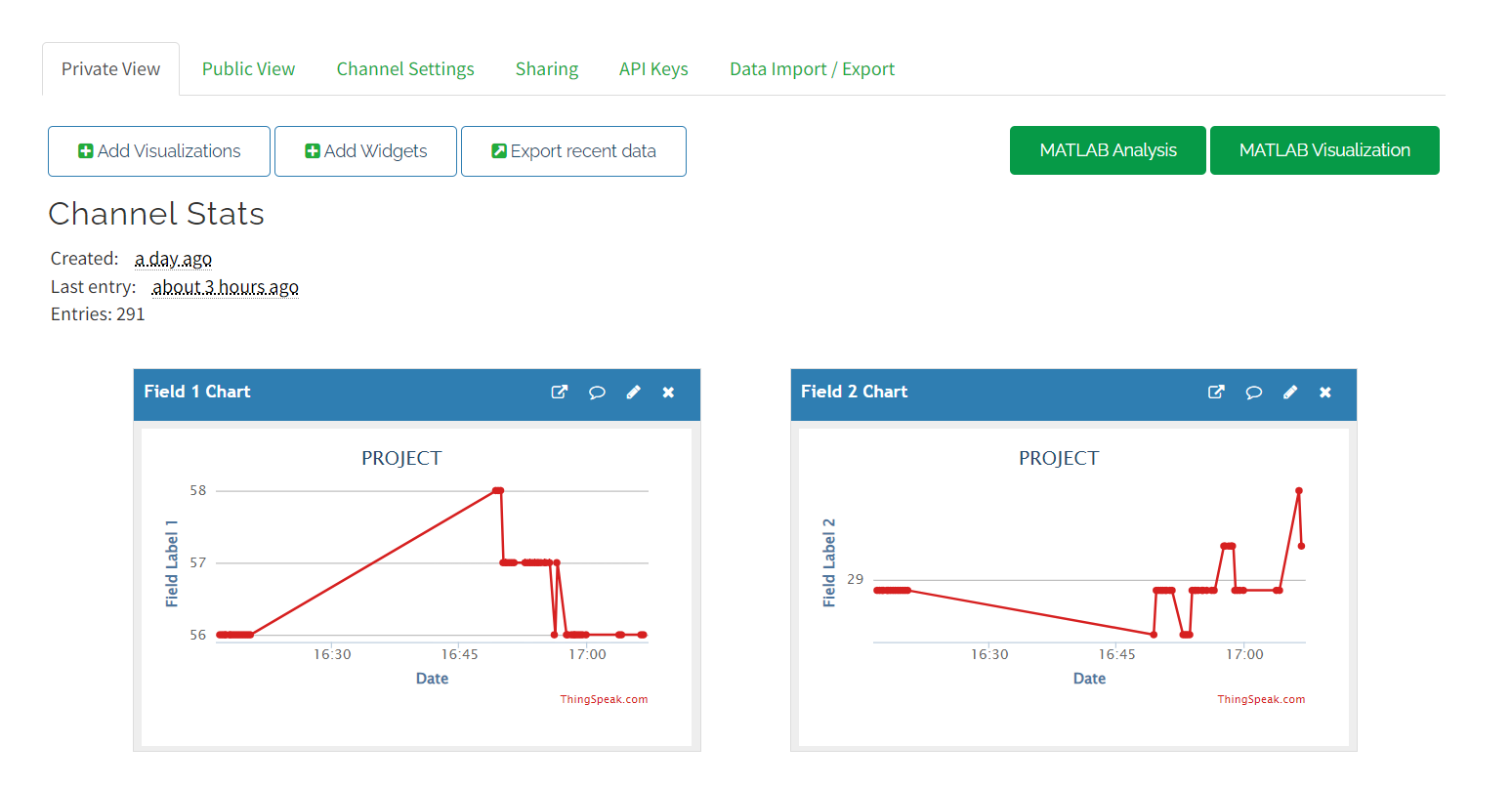
# 

# CIRCUIT DIAGRAM (CONNECTION)

# 

# 5.RESULT

In the below figure the graph shows the result that is the first graph shows the humidity in the graph as the values changes continuously the graph results as the figure given below. And the second graph shows the temperature of the project as it changes continuously gradually the below graph is the result of the project.



# 6.CONCLUSION

As the result of the project we conclude that by reading the values of the grains from the godown using the sensors such as using the DHT sensor to read the humidity and the temperature. Soil Moisture Sensor to moniter the moisture of the grains to make the perfect needs of the grains, we worked on the project and solved this query .Using the dc motors we can on the fans and make the grains as perfect grains that are used for future use. Therefore the grains which are stored in the godown are not spoiled by using the project and make the grains stored in the perfect physical parameters storage space.

# 7.REFERENCES

1. [**https://randomnerdtutorials.com/esp32-dht11-dht22-temperature-humidity-sensor-arduino-ide/**](https://randomnerdtutorials.com/esp32-dht11-dht22-temperature-humidity-sensor-arduino-ide/)
2. [**https://microcontrollerslab.com/esp32-iot-soil-moisture-monitoring-system-adafruit-io/#:~:text=The%20VCC%20pin%20of%20the%20soil%20moisture%20sensor,board.%20All%20three%20devices%20will%20be%20commonly%20grounded**](https://microcontrollerslab.com/esp32-iot-soil-moisture-monitoring-system-adafruit-io/#:~:text=The%20VCC%20pin%20of%20the%20soil%20moisture%20sensor,board.%20All%20three%20devices%20will%20be%20commonly%20grounded)**.**
3. [**https://esp32io.com/tutorials/esp32-dc-motor**](https://esp32io.com/tutorials/esp32-dc-motor)