DEPARTMENT OF INFORMATION TECHNOLOGY
FACULTY OF ENGINEERING & TECHNOLOGY

IOT PROJECT REPORT

SUBJECT TITLE : INTERNET OF THINGS

SUBJECT CODE: 15IT422E

SUBMITTED TO: Prof Kayalvizhi Jayavel

TEMPERATURE AND HUMDITIY DATA LOGGING

R SASIDHARAN

RA1611008010127

Department of Information Technology

SRM University, SRM Nagar, Kattankulathur-603203
Kanchipuram District, Tamil Nadu

LINKS TO GITHUB AND YOUTUBE:

YouTube: https://www.youtube.com/watch?v=yUDJIgMzDX4&t=3s

Github: https://github.com/sasidharan2998/IOT2019_SASIDHARAN_TEMP-HUMIDIDTY-DATA-LOGGING

ACKNOWLEDGEMENT

I would like to express my special thanks of gratitude to my teacher mrs Kayalvizhi jayavel who gave me the golden opportunity to do this wonderful project on the topic temperature and humidity data logging which also helped me in doing a lot of Research and i came to know about so many new things I am really thankful to them.

Secondly i would also like to thank my parents and friends who helped me a lot in finalizing this project within the limited time frame.

ABSTRACT

As Engineers we always rely upon the data collected to design or improve a system. Recording data and analyzing them is a common practice in most of the industries, here we are building **Data Logger Project** where we will learn how we can log data at a specific interval of time. We will use an esp8266 wifi module board to read some data (here temperature, humidity, date and time) and save them on a SD card and the cloud simultaneously.

HARDWARE REQUIRED:

- Nodemcu ESP8266 module -- 380rs [Amazon]
- Micro SD card module --150rs [Amazon]
- Jumper Wires
- DHT11 Temperature and Humidity sensor --215rs[Amazon]
- DS3231 Real time clock module --222rs[Amazon]
- Micro USB cable

SOFTWARE REQUIRED :

- Arduino IDE.
- ESP8266 library.
- RTC MODULE Library.
- DHT11 Library.

TOTAL COST OF COMPONENTS :- RS.1200

SYSTEM OVERVIEW

The main component of the setup is the Nodemcu ESP8266 module. All the other hardware components are connected to the Nodemcu. The board is programmed in Arduino IDE and uses the ESP8266,RTC ,DHT11 sensor libraries. These libraries have been added to the Arduino IDE.

DS3231 is a RTC (Real Time Clock) module. It is used to maintain the date and time for most of the Electronics projects. This module has its own coin cell power supply using which it maintains the date and time even when the main power is removed or the MCU has gone though a hard reset. So once we set the date and time in this module it will keep track of it always

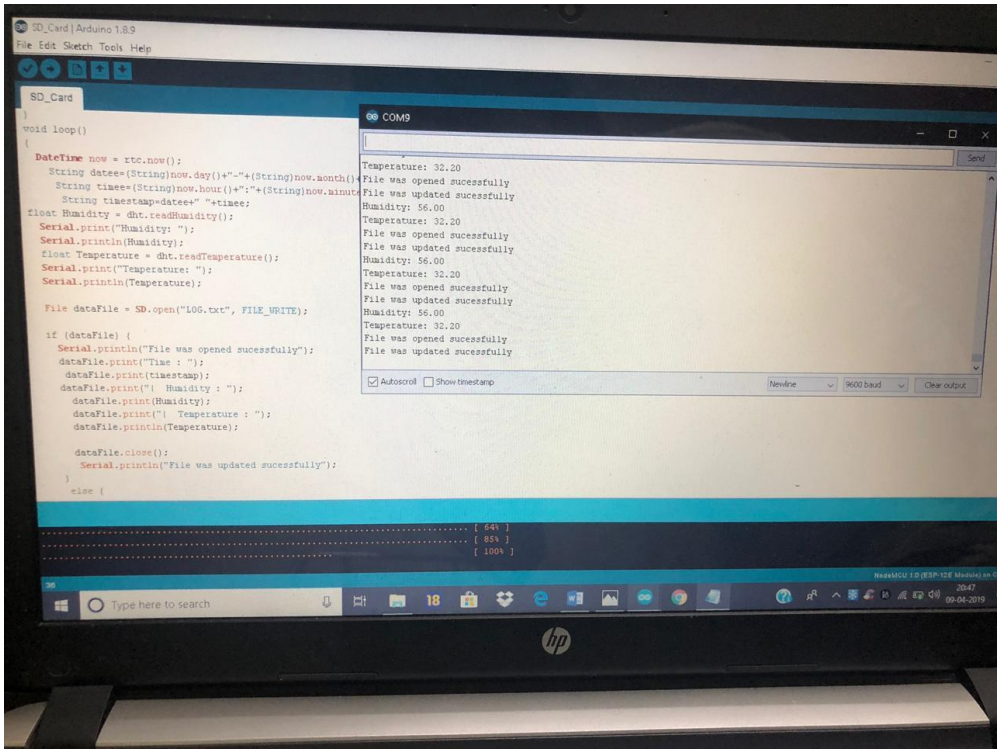
DHT11 is a Temperature come Humidity sensor. It sends the values of temperature and humidity as an 8-bit data serially through the output pin of the module. The library reads this data by using the software serial function of the Arduino.

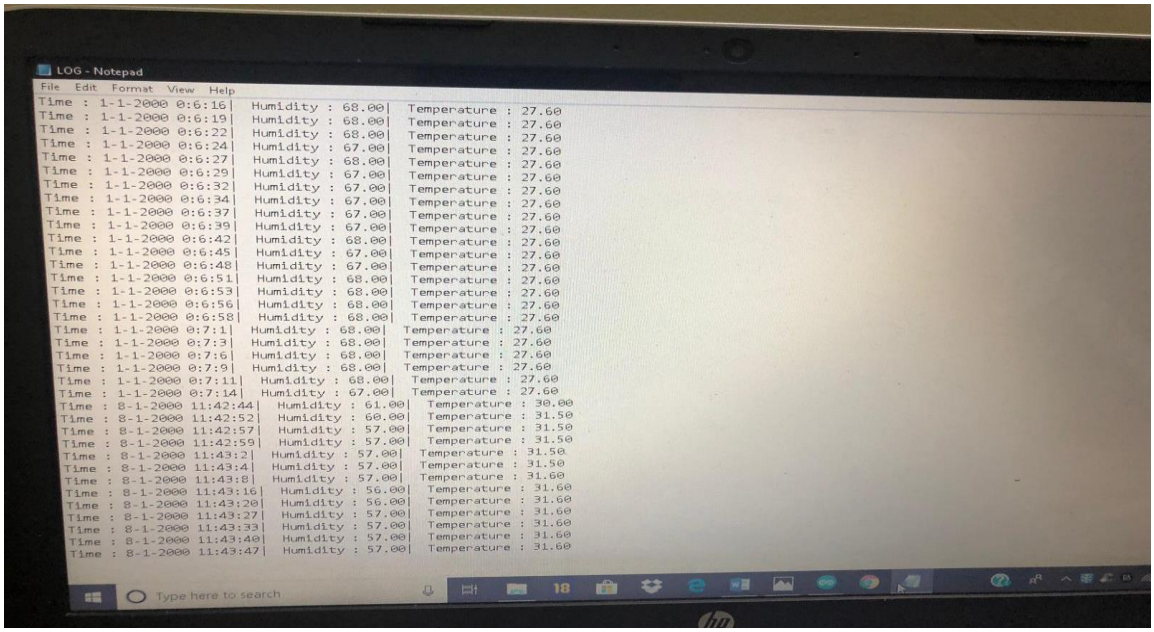
CODE

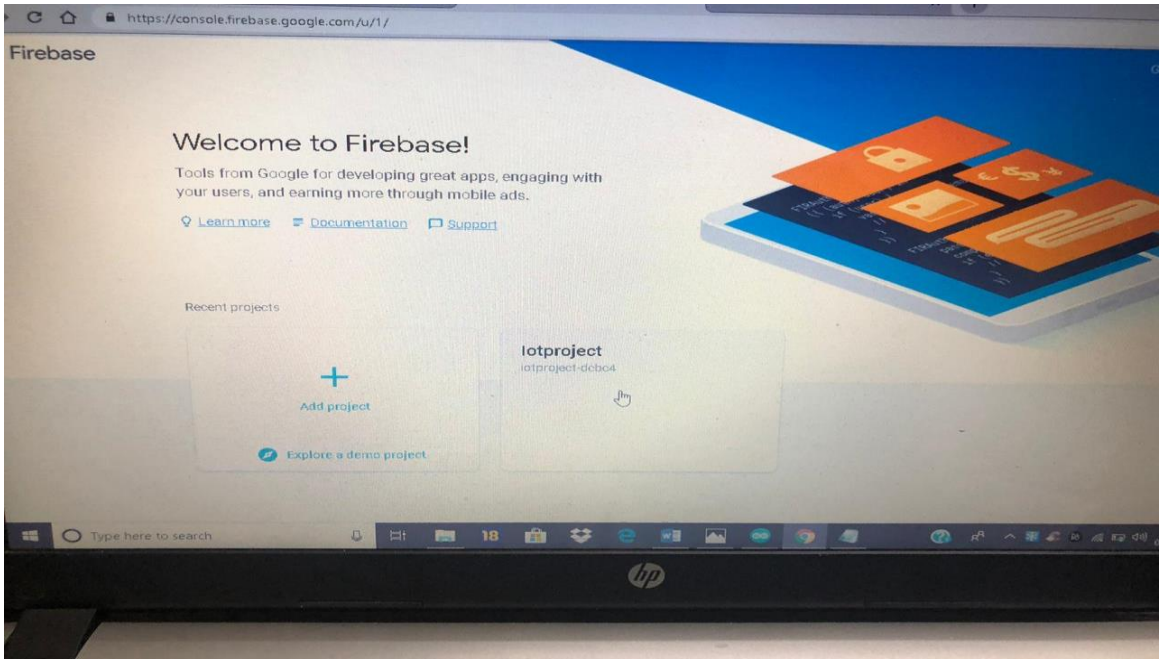
```
#include <ESP8266WiFi.h>
#include <SD.h>
#include "RTClib.h"
#include <ESP8266WiFi.h>
#include <FirebaseArduino.h>
// including all necessary header files
RTC_DS3231 rtc;
#include <DHT.h>
//fire key which is available on the website needs to be copied
 #define FIREBASE_HOST "iotproject-dcbc4.firebaseio.com"
#define FIREBASE_AUTH "735eMELLw1lId782Pfo2Fw7LGAGJxF200rX9ULJA"
#define WIFI_SSID "sasi"
#define WIFI_PASSWORD "012345678"
#define DHTPIN D3
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);
#define CS_PIN  D4
void setup(){
  Serial.begin(9600);
  Serial.print("Connecting with SD Card");
   dht.begin();
  //pinMode(D4, OUTPUT);
  if (!SD.begin(CS_PIN)) {
    Serial.println("Failed to connect to SD Card");
    return;
  }
```

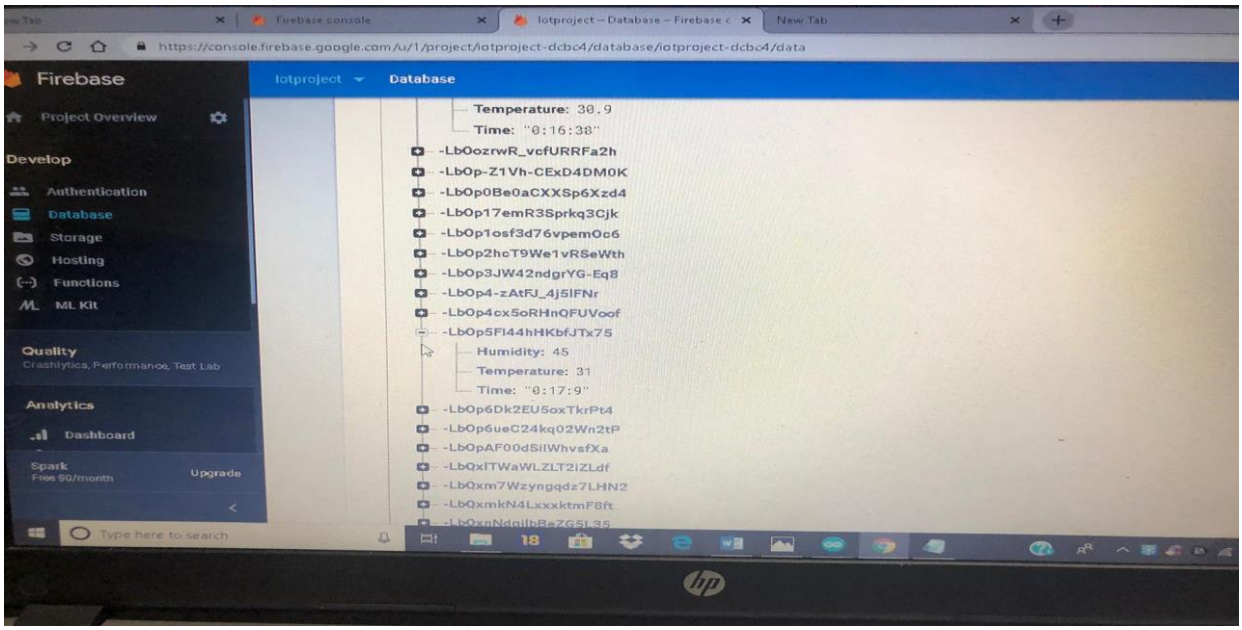
```
//now sd is connected and rtc starts to function
Serial.println("SD Card Connected");
  if (! rtc.begin()) {
    Serial.println("Couldn't find RTC");
    while (1);
  }//connecting wifi
   WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("connecting");
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");
 delay(500);
  }
  Serial.println();
  Serial.print("connected: ");
  Serial.println(WiFi.localIP());
  Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
 }
void loop()
{//data logging in sd card
  DateTime now = rtc.now();
    String datee=(String)now.day()+"-"+(String)now.month()+"-"+(String)now.year();
     String timee=(String)now.hour()+":"+(String)now.minute()+":"+(String)now.second();
      String timestamp=datee+" "+timee;
float Humidity = dht.readHumidity();
  Serial.print("Humidity: ");
  Serial.println(Humidity);
  float Temperature = dht.readTemperature();
```

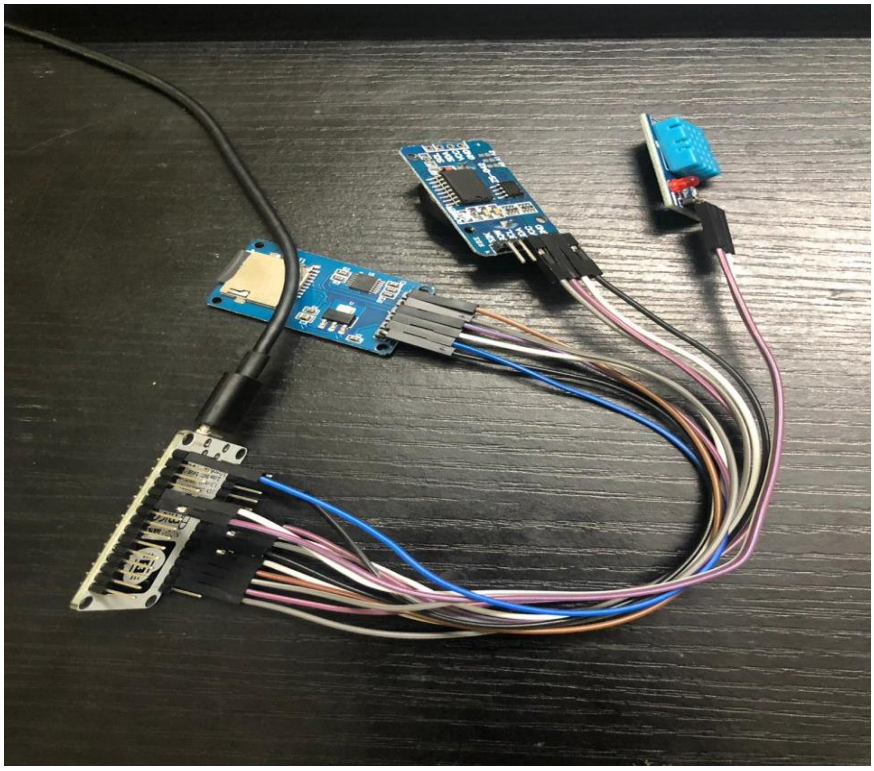
```
 Serial.print("Temperature: ");
  Serial.println(Temperature);
  File dataFile = SD.open("LOG.txt", FILE_WRITE);
  if (dataFile) {
    Serial.println("File was opened sucessfully");
    dataFile.print("Time : ");
     dataFile.print(timestamp);
    dataFile.print("|  Humidity : ");
      dataFile.print(Humidity);
      dataFile.print("|  Temperature : ");
      dataFile.println(Temperature);
       dataFile.close();
       Serial.println("File was updated sucessfully");
    }
     else {
    Serial.println(" Failed to open file LOG.txt");
  }//data logging in cloud
StaticJsonBuffer<200> jsonBuffer;
    JsonObject & root=jsonBuffer.createObject();
    root["Time"]=timee;
    root["Humidity"]=Humidity;
root["Temperature"]=Temperature;
    Firebase.push(datee,root);
  delay(2000);
```

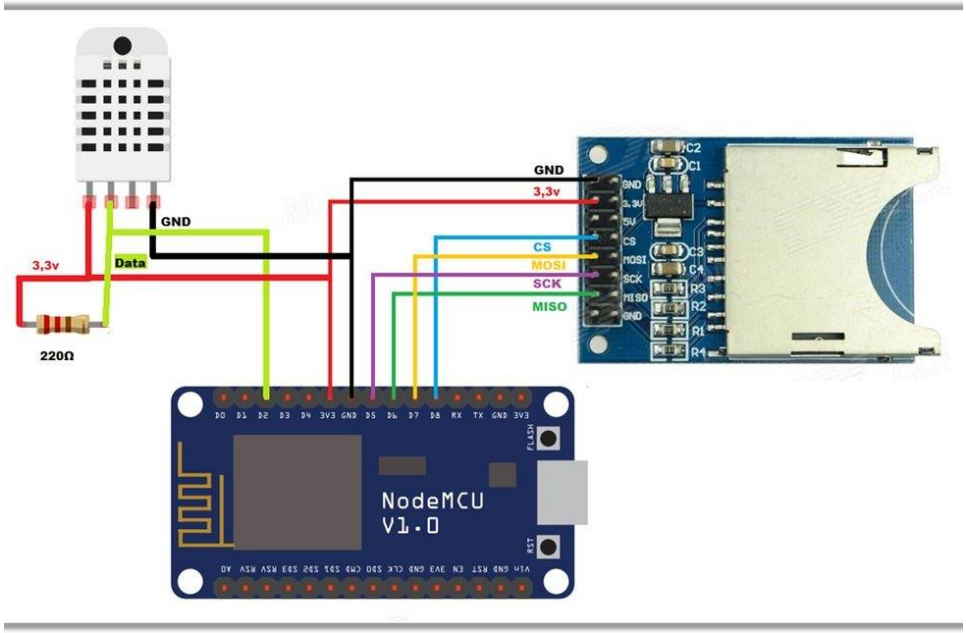

SD_Card | Arduino 1.8.9
File Edit Sketch Tools Help
SD_Card
void loop()
{
DateTime now = rtc.now();
float Humidity = dht.readHumidity();
Serial.print("Humidity: ");
Serial.println(Humidity);
float Temperature = dht.readTemperature();
Serial.print("Temperature: ");
Serial.println(Temperature);
File dataFile = SD.open("LOG.txt", FILE_WRITE);
if (dataFile) {
Serial.println("File was opened sucessfully");
dataFile.print("Time : ");
dataFile.print(timestamp);
dataFile.print("| Humidity : ");
dataFile.print(Humidity);
dataFile.print("| Temperature : ");
dataFile.println(Temperature);
dataFile.close();
Serial.println("File was updated sucessfully");
}
else {
COM9
Send
Temperature: 32.20
File was opened sucessfully
File was updated sucessfully
Humidity: 56.00
Temperature: 32.20
File was opened sucessfully
File was updated sucessfully
Humidity: 56.00
Temperature: 32.20
File was opened sucessfully
File was updated sucessfully
Humidity: 56.00
Temperature: 32.20
File was opened sucessfully
File was updated sucessfully
Autoscroll
Show timestamp
Newline
9600 baud
Clear output
[64%]
[85%]
[100%]
Type here to search
20:47
09-04-2019


LOG - Notepad
File Edit Format View Help
Time : 1-1-2000 0:6:16| Humidity : 68.00| Temperature : 27.60
Time : 1-1-2000 0:6:19| Humidity : 68.00| Temperature : 27.60
Time : 1-1-2000 0:6:22| Humidity : 68.00| Temperature : 27.60
Time : 1-1-2000 0:6:24| Humidity : 67.00| Temperature : 27.60
Time : 1-1-2000 0:6:27| Humidity : 68.00| Temperature : 27.60
Time : 1-1-2000 0:6:29| Humidity : 67.00| Temperature : 27.60
Time : 1-1-2000 0:6:32| Humidity : 67.00| Temperature : 27.60
Time : 1-1-2000 0:6:34| Humidity : 67.00| Temperature : 27.60
Time : 1-1-2000 0:6:37| Humidity : 67.00| Temperature : 27.60
Time : 1-1-2000 0:6:39| Humidity : 67.00| Temperature : 27.60
Time : 1-1-2000 0:6:42| Humidity : 68.00| Temperature : 27.60
Time : 1-1-2000 0:6:45| Humidity : 67.00| Temperature : 27.60
Time : 1-1-2000 0:6:48| Humidity : 67.00| Temperature : 27.60
Time : 1-1-2000 0:6:51| Humidity : 68.00| Temperature : 27.60
Time : 1-1-2000 0:6:53| Humidity : 68.00| Temperature : 27.60
Time : 1-1-2000 0:6:56| Humidity : 68.00| Temperature : 27.60
Time : 1-1-2000 0:6:58| Humidity : 68.00| Temperature : 27.60
Time : 1-1-2000 0:7:1| Humidity : 68.00| Temperature : 27.60
Time : 1-1-2000 0:7:3| Humidity : 68.00| Temperature : 27.60
Time : 1-1-2000 0:7:6| Humidity : 68.00| Temperature : 27.60
Time : 1-1-2000 0:7:9| Humidity : 68.00| Temperature : 27.60
Time : 1-1-2000 0:7:11| Humidity : 68.00| Temperature : 27.60
Time : 1-1-2000 0:7:14| Humidity : 67.00| Temperature : 27.60
Time : 8-1-2000 11:42:44| Humidity : 61.00| Temperature : 30.00
Time : 8-1-2000 11:42:52| Humidity : 60.00| Temperature : 31.50
Time : 8-1-2000 11:42:57| Humidity : 57.00| Temperature : 31.50
Time : 8-1-2000 11:42:59| Humidity : 57.00| Temperature : 31.50
Time : 8-1-2000 11:43:2| Humidity : 57.00| Temperature : 31.50
Time : 8-1-2000 11:43:4| Humidity : 57.00| Temperature : 31.50
Time : 8-1-2000 11:43:8| Humidity : 57.00| Temperature : 31.60
Time : 8-1-2000 11:43:16| Humidity : 56.00| Temperature : 31.60
Time : 8-1-2000 11:43:20| Humidity : 56.00| Temperature : 31.60
Time : 8-1-2000 11:43:27| Humidity : 57.00| Temperature : 31.60
Time : 8-1-2000 11:43:33| Humidity : 57.00| Temperature : 31.60
Time : 8-1-2000 11:43:40| Humidity : 57.00| Temperature : 31.60
Time : 8-1-2000 11:43:47| Humidity : 57.00| Temperature : 31.60
Type here to search


https://console.firebase.google.com/u/1/
Firebase
Welcome to Firebase!
Tools from Google for developing great apps, engaging with your users, and earning more through mobile ads.
Learn more
Documentation
Support
Recent projects
Add project
Explore a demo project
Iotproject
iotproject-dcbc4
Type here to search


Firebase console
Iotproject – Database – Firebase c
New Tab
https://console.firebase.google.com/u/1/project/iotproject-dcbc4/database/iotproject-dcbc4/data
Firebase
Iotproject
Database
Project Overview
Develop
Authentication
Database
Storage
Hosting
Functions
ML Kit
Quality
Crashlytics, Performance, Test Lab
Analytics
Dashboard
Spark
Free $0/month
Upgrade
Temperature: 30.9
Time: "0:16:38"
-LbOozrwR_vcfURRFa2h
-LbOp-Z1Vh-CExD4DM0K
-LbOp0Be0aCXXSp6Xzd4
-LbOp17emR3Sprkq3Cjk
-LbOp1osf3d76vpemOc6
-LbOp2hcT9We1vRSeWth
-LbOp3JW42ndgrYG-Eq8
-LbOp4-zAtFJ_4j5lFNr
-LbOp4cx5oRHnQFUVoof
-LbOp5Fl44hHKbfJTx75
Humidity: 45
Temperature: 31
Time: "0:17:9"
-LbOp6Dk2EU5oxTkrPt4
-LbOp6ueC24kq02Wn2tP
-LbOpAF00dSilWhvsfXa
-LbOxlTWaWLZLT2iZLdf
-LbOxm7Wzyngqdz7LHN2
-LbOxmkN4LxxxktmF8ft
Type here to search





Result :

Temperature and humidity data logging in sd card and cloud has been successfully studied and implemented with an esp8266 .