**1.IOT PROTOCOLS**

**(THEORY)**

**2. TEMPERATURE SENSOR (cloud storage)**

**CODING:**

**Arduino software**

**#include <ESP8266WiFi.h>**

**#include <ESP8266HTTPClient.h>**

**#include "DHT.h"**

**#define DHTPIN D1**

**#define DHTTYPE DHT11**

**DHT dht(DHTPIN, DHTTYPE);**

**const char\* ssid = "aruna"; //phn hotspot name**

**const char\* password = "arunashankar"; //phn hotspot password**

**const char\* apiKey = "1IAJKX4DE5D6A99T"; //it is taken from thingspeak app**

**const char\* serverAddress = "http://api.thingspeak.com/update";**

**//https://api.thingspeak.com/update?api\_key=H02UFYAJGTK5D0E9&field1=0**

**//https://api.thingspeak.com/channels/700867/feeds.json?results=2**

**void setup() {**

**dht.begin();**

**Serial.begin(115200);**

**WiFi.begin(ssid, password);**

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

**delay(1000);**

**Serial.println("Connecting to WiFi...");**

**}**

**}void loop() {**

**float humidity = 0;**

**float temperature = 0;**

**{**

**Serial.print("Humidity: ");**

**Serial.print(humidity);**

**Serial.print("%\tTemperature: ");**

**Serial.print(temperature);**

**Serial.println("°C");**

**}**

**WiFiClient client;**

**HTTPClient http;**

**http.begin(client, serverAddress);**

**http.addHeader("Content-Type", "application/x-www-form-urlencoded");**

**String postData = "api\_key=" + String(apiKey) + "&field1=" + String(temperature) + "&field2=" + String(humidity);**

**int httpCode = http.POST(postData);**

**if (httpCode > 0) {**

**Serial.print("HTTP Response Code: ");**

**Serial.println(httpCode);**

**} else {**

**Serial.println("HTTP POST failed.");**

**}**

**http.end();**

**delay(3000);**

**temperature++;**

**humidity++;**

**}**

**3. LED BLINKING APPLICATION**

**Ardiuno software:**

**Coding:**

**int led=13;**

**void setup(){**

**pinMode(led,OUTPUT);**

**}**

**void loop(){**

**for(int led=0;led<=5;led++){**

**digitalWrite(led,HIGH);**

**delay(1000);**

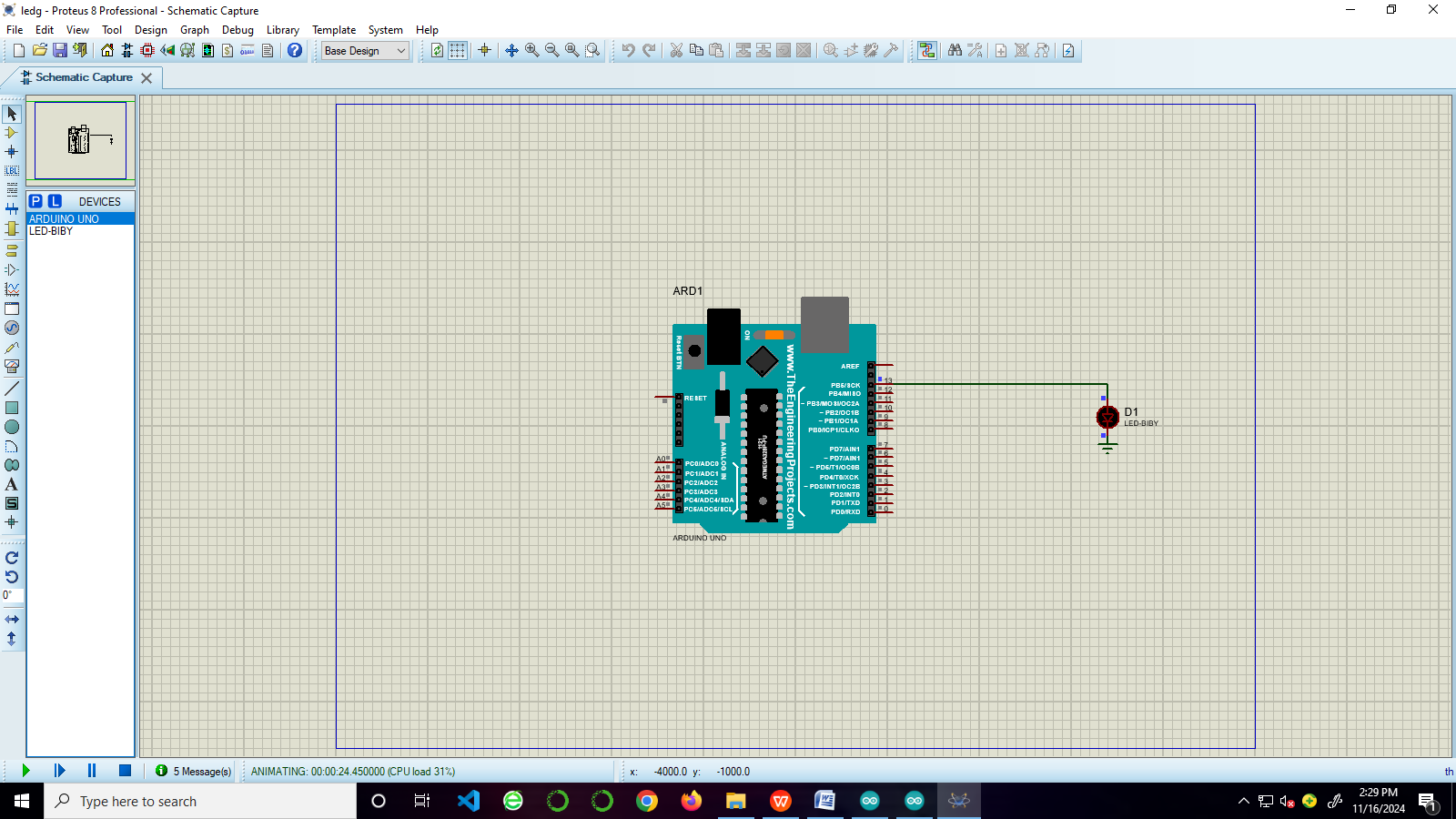
**digitalWrite(led,LOW);**

**delay(1000);**

**}**

**}**

**Proteus stimulation:**



5.Room temperature refer 3 experiment

6. **MEASURING SOIL MOISTURE SENSOR**

#include "DHT.h"

#define DHTPIN D1

#define DHTTYPE DHT11

#include <Arduino.h>

#if defined(ESP32)

#include <WiFi.h>

#elif defined(ESP8266)

#include <ESP8266WiFi.h>

#endif

#include <Firebase\_ESP\_Client.h>

DHT dht(DHTPIN, DHTTYPE);

//Provide the token generation process info.

#include "addons/TokenHelper.h"

//Provide the RTDB payload printing info and other helper functions.

#include "addons/RTDBHelper.h"

// Insert your network credentials

#define WIFI\_SSID "aruna"

#define WIFI\_PASSWORD "arunashankar"

// Insert Firebase project API Key

#define API\_KEY "AIzaSyAIW\_LxE8GFiihvn6tQ5ChOuYucSb5SqPI"

// Insert RTDB URLefine the RTDB URL \*/

#define DATABASE\_URL "soilmoisture-a5e62-default-rtdb.firebaseio.com/"

//Define Firebase Data object

FirebaseData fbdo;

FirebaseAuth auth;

FirebaseConfig config;

//unsigned long sendDataPrevMillis = 0;

//int count = 0;

bool signupOK = false;

void setup(){

pinMode(DHTPIN, INPUT);

dht.begin();

Serial.begin(115200);

WiFi.begin(WIFI\_SSID, WIFI\_PASSWORD);

Serial.print("Connecting to Wi-Fi");

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

Serial.print(".");

delay(300);

}

Serial.println();

Serial.print("Connected with IP: ");

Serial.println(WiFi.localIP());

Serial.println();

/\* Assign the api key (required) \*/

config.api\_key = API\_KEY;

/\* Assign the RTDB URL (required) \*/

config.database\_url = DATABASE\_URL;

/\* Sign up \*/

if (Firebase.signUp(&config, &auth, "", "")){

Serial.println("ok");

signupOK = true;

}

else{

Serial.printf("%s\n", config.signer.signupError.message.c\_str());

}

/\* Assign the callback function for the long running token generation task \*/

config.token\_status\_callback = tokenStatusCallback; //see addons/TokenHelper.h

Firebase.begin(&config, &auth);

Firebase.reconnectWiFi(true);

}

void loop(){

int data=analogRead(A0);

data=1024-data;

delay(1000);

if (Firebase.ready() && signupOK ) {

if (Firebase.RTDB.setFloat(&fbdo, "soilmoisture",data)){

// Serial.println("PASSED");

Serial.print("soilmoisture");

Serial.println(data);

}

else {

Serial.println("FAILED");

Serial.println("REASON: " + fbdo.errorReason());

}

// Write an Float number on the database path test/float

}

Serial.println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

}

7.ultrasonic sensor

int tig=7,echo=5;

int dist,Time;

void setup(){

pinMode(tig,OUTPUT);

pinMode(echo,INPUT);

Serial.begin(9600);

}

void loop(){

digitalWrite(tig,LOW);

delayMicroseconds(2);

digitalWrite(tig,HIGH);

delayMicroseconds(10);

digitalWrite(tig,LOW);

Time=pulseIn(echo,HIGH);

dist=Time/29/2;

Serial.println(dist);

delay(1000);

}

**8.DEVELOP A SIMPLE APPLICATION BASED ON SENSORS**

**Refer a any prg about sensor eg: ultrasonic sensor prg ,soil moisture sensor**

**9.DEVELOP A COMMERCIAL IOT APPLICATION (AUTOMATED HAND SANITIZING)**

#include<Servo.h>

#define echoPin 4

#define trigPin 5Servo

Myservo:

int long duration;

int distance: void setup()

{

Myservo.attach(3):

pinMode(echoPin,INPUT);

pinMode(trigPin, OUTPUT);

void loop()

digitalWrite(trigPin,LOW);

delayMicroseconds(2);

digitalWrite(trigPin,HIGH);

delayMicroseconds(10):

duration=pulseIn(echoPin,HIGH);

distance (duration 0.034/2):

if(distance<=5)

{

Myservo.write(180):

else { Myservo.write(0);

}

delay(500);

}

10.refer 9th prg