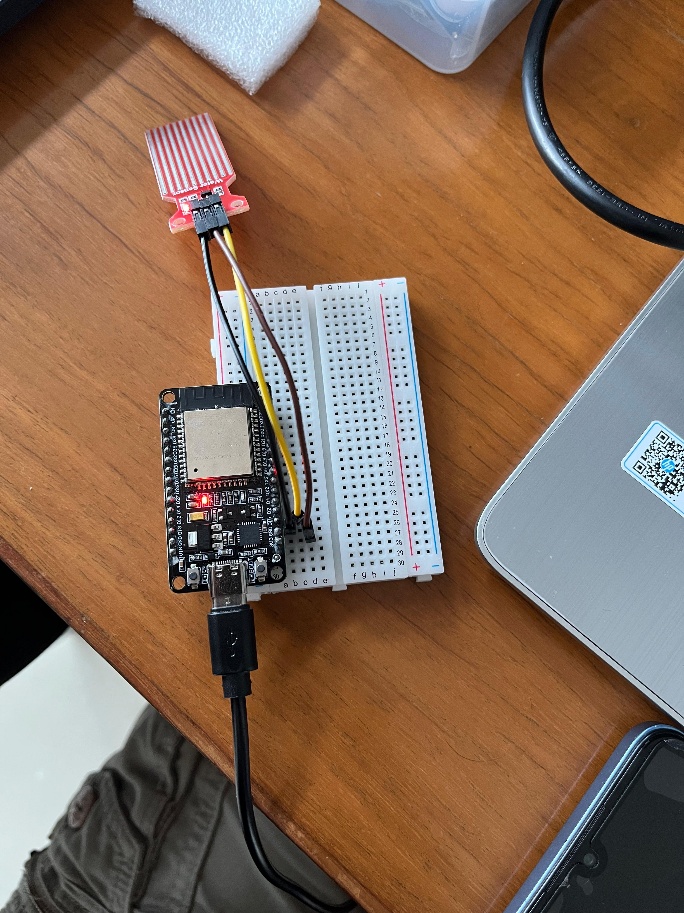
**WATER LEVEL DETECTOR**



**Source Code:**

#include <WiFi.h>

#include <HTTPClient.h>

#include "ThingSpeak.h"

//-- Wi-Fi credentials --//

const char\* ssid = "Woi";        // Your Wi-Fi SSID

const char\* password = "candra123";  // Your Wi-Fi password

//-- ThingSpeak settings --//

unsigned long channelNumber = 2747170; // Replace with your ThingSpeak channel number

const char\* apiKey = "HBT1SVX4RVBCG2PW"; // Replace with your ThingSpeak API key

const char\* server = "https://api.thingspeak.com/update?api\_key=";

//-- Pin configurations --//

#define wlS 15  // Water level sensor on digital pin D15

#define bz 19   // Buzzer on pin 19

WiFiClient client;

void setup() {

  //-- Serial Monitor --//

  Serial.begin(9600);

  delay(1000); // Wait a moment to allow Serial monitor to open

  //-- Pin configurations --//

  pinMode(wlS, INPUT);

  pinMode(bz, OUTPUT);

  //-- Connect to Wi-Fi --//

  WiFi.begin(ssid, password);

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

    delay(1000);

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

  }

  Serial.println("Connected to WiFi");

  //-- Initialize ThingSpeak --//

  ThingSpeak.begin(client);

  Serial.println("Setup complete. Starting loop...");

}

void loop() {

  int val = digitalRead(wlS); // Read the digital value from the water sensor

  Serial.print("Sensor reading (D15): ");

  Serial.println(val);

  // Update the water level status based on sensor reading

  String nilai = val ? "HIGH" : "LOW";

  // Buzzer control and serial output

  if (val == LOW) {

    digitalWrite(bz, LOW); // Turn on buzzer if water level is low

    Serial.println("Buzzer: ON");

  } else {

    digitalWrite(bz, HIGH); // Turn off buzzer if water level is high

    Serial.println("Buzzer: OFF");

  }

  // Sending data to ThingSpeak

  if (WiFi.status() == WL\_CONNECTED) {

    ThingSpeak.setField(1, val); // Sending water level as 0 or 1 to Field 1

    int responseCode = ThingSpeak.writeFields(channelNumber, apiKey);

    if (responseCode == 200) {

      Serial.println("Data sent to ThingSpeak successfully");

    } else {

      Serial.println("Error sending data to ThingSpeak: " + String(responseCode));

    }

  } else {

    Serial.println("WiFi not connected");

  }

  delay(5000); // ThingSpeak allows updates every 15 seconds

}  
------------------------------------------------------------------------------------

**OUTPUT:**

**PEMBAHASAN**

Water Level Detector ini dirancang untuk memantau ketinggian air secara otomatis menggunakan sensor dan mengirim data ke platform ThingSpeak. Berikut adalah poin utama dari sistem ini:

1. Fungsi Sensor Water Level

* Sensor yang digunakan bekerja secara digital dan mengeluarkan nilai HIGH (1) jika ketinggian air cukup, dan LOW (0) jika air rendah.
* Jika air berada di bawah batas tertentu, sensor akan mendeteksi LOW sebagai tanda level air yang rendah.

2. Penggunaan Buzzer

* Saat sensor mendeteksi air rendah (LOW), buzzer menyala sebagai peringatan.
* Ketika air kembali normal (HIGH), buzzer mati secara otomatis, memberikan notifikasi sederhana yang efektif.

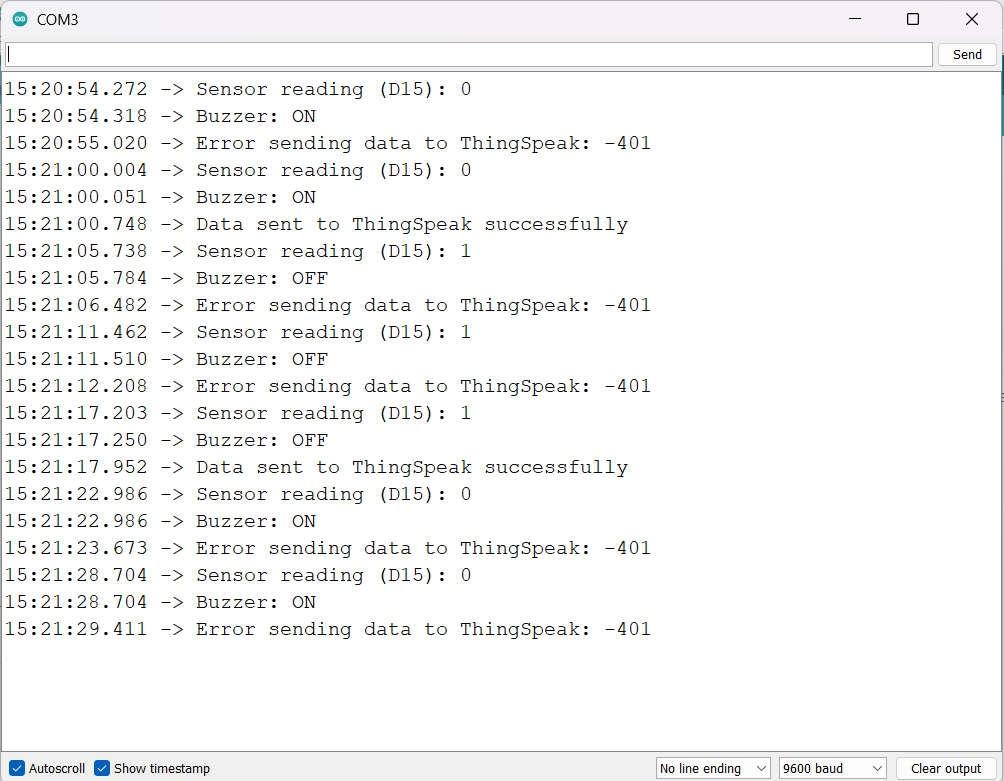
3. Koneksi ke Wi-Fi dan ThingSpeak

* Modul ESP32 menghubungkan perangkat ke Wi-Fi, memungkinkan pengiriman data ketinggian air ke ThingSpeak.
* Data dikirim ke ThingSpeak setiap 15 detik, dan pengguna bisa memantau level air secara jarak jauh melalui dashboard.

4. Kesederhanaan dan Efektivitas

* Sistem ini memadukan monitoring lokal menggunakan buzzer dan monitoring jarak jauh menggunakan ThingSpeak, sehingga pengguna bisa memantau kondisi air secara real-time dari mana saja.
* Jika terjadi kegagalan koneksi, perangkat akan mencoba mengirim ulang data secara otomatis, memastikan keandalan pengiriman.

5. Hasil (Output)

****

Hasil Pada output water level detector ini, sensor membaca ketinggian air dan mengontrol buzzer sesuai dengan hasil pembacaan. Jika air berada di level rendah (nilai sensor 0), buzzer menyala sebagai peringatan, dan mati saat air kembali ke level aman (nilai sensor 1). Data sensor dikirim ke platform ThingSpeak untuk monitoring jarak jauh. Namun, terdapat error -401, yang menunjukkan masalah otentikasi, biasanya karena kunci API yang salah atau tidak valid. Beberapa pengiriman data berhasil, namun perlu diperiksa kembali validitas API key agar data selalu terkirim dengan sukses