MICROPROCESSESOR PROGRAM:

```cpp

#include <Wire.h>

#include <Adafruit\_Sensor.h>

#include <Adafruit\_BMP280.h>

#include <DHT.h>

// Pin configurations for sensors

const int BMP280\_SDA\_PIN = 4; // Connect BMP280 SDA pin to GPIO 4

const int BMP280\_SCL\_PIN = 5; // Connect BMP280 SCL pin to GPIO 5

const int DHT\_PIN = 2; // Connect DHT11 data pin to GPIO 2

const int ULTRASONIC\_TRIGGER\_PIN = 6; // Connect ultrasonic sensor trigger pin to GPIO 6

const int ULTRASONIC\_ECHO\_PIN = 7; // Connect ultrasonic sensor echo pin to GPIO 7

Adafruit\_BMP280 bmp; // BMP280 sensor

DHT dht(DHT\_PIN, DHT11); // DHT11 sensor

long duration; // To store ultrasonic sensor duration

float distance; // To store calculated distance from ultrasonic sensor

void setup() {

Serial.begin(115200);

// Initialize BMP280 sensor

if (!bmp.begin(BMP280\_SDA\_PIN, BMP280\_SCL\_PIN)) {

Serial.println("Could not find a valid BMP280 sensor, check wiring!");

while (1);

}

// Initialize DHT sensor

dht.begin();

// Ultrasonic sensor pin modes

pinMode(ULTRASONIC\_TRIGGER\_PIN, OUTPUT);

pinMode(ULTRASONIC\_ECHO\_PIN, INPUT);

}

void loop() {

// Read temperature and pressure from BMP280 sensor

float temperatureBMP = bmp.readTemperature();

float pressure = bmp.readPressure() / 100.0; // Pressure in hPa

// Read temperature and humidity from DHT11 sensor

float humidity = dht.readHumidity();

float temperatureDHT = dht.readTemperature(); // Read temperature in Celsius

// Read water level from ultrasonic sensor

digitalWrite(ULTRASONIC\_TRIGGER\_PIN, LOW);

delayMicroseconds(2);

digitalWrite(ULTRASONIC\_TRIGGER\_PIN, HIGH);

delayMicroseconds(10);

digitalWrite(ULTRASONIC\_TRIGGER\_PIN, LOW);

duration = pulseIn(ULTRASONIC\_ECHO\_PIN, HIGH);

distance = duration \* 0.034 / 2; // Calculate distance in centimeters

// Print sensor values

Serial.print("Temperature (BMP280): ");

Serial.print(temperatureBMP);

Serial.println(" °C");

Serial.print("Pressure: ");

Serial.print(pressure);

Serial.println(" hPa");

Serial.print("Temperature (DHT11): ");

Serial.print(temperatureDHT);

Serial.println(" °C");

Serial.print("Humidity: ");

Serial.print(humidity);

Serial.println(" %");

Serial.print("Water Level: ");

Serial.print(distance);

Serial.println(" cm");

delay(5000); // Delay for 5 seconds before reading again

}

AQI.html PROGRAM

<!DOCTYPE html>

<html>

<head>

<title>AQI Data from Firebase</title>

<script src="https://www.gstatic.com/firebasejs/8.10.0/firebase-app.js"></script>

<script src="https://www.gstatic.com/firebasejs/8.10.0/firebase-database.js"></script>

</head>

<body>

<h1>Air Quality Index (AQI) Data</h1>

<div id="aqi-data">

<!-- AQI data will be displayed here -->

</div>

<script>

// Initialize Firebase with your project's configuration

var firebaseConfig = {

apiKey: "YOUR\_API\_KEY",

authDomain: "YOUR\_AUTH\_DOMAIN",

databaseURL: "YOUR\_DATABASE\_URL",

projectId: "YOUR\_PROJECT\_ID",

storageBucket: "YOUR\_STORAGE\_BUCKET",

messagingSenderId: "YOUR\_MESSAGING\_SENDER\_ID",

appId: "YOUR\_APP\_ID"

};

firebase.initializeApp(firebaseConfig);

// Reference to your AQI data in Firebase

var aqiRef =

firebase.database().ref("aqi");

// Listen for changes in the AQI data

aqiRef.on("value", function(snapshot) {

var aqiData = snapshot.val();

// Update the HTML to display the AQI data

if (aqiData) {

document.getElementById("aqi-data").innerHTML = "AQI: " + aqiData;

} else {

document.getElementById("aqi-data").innerHTML = "No data available";

}

});

</script>

</body>

</html>

