

## CODE :

```
#include <OneWire.h>
#include <DallasTemperature.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

// === Pin Definitions ===

#define PIN_TURB_ADC  A0 // Turbidity sensor analog pin
#define PIN_DS18B20  D4 // DS18B20 temperature sensor data pin
#define LED_RED      D7 // Red LED for dirty water
#define LED_GREEN    D8 // Green LED for clean water
#define LED_YELLOW   D0 // Yellow LED for abnormal temperature

// === OLED Pins for ESP8266 ===

#define OLED_SDA D6
#define OLED_SCL D5
#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64

Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, -1);

// === DS18B20 Setup ===

OneWire oneWire(PIN_DS18B20);
DallasTemperature tempSensor(&oneWire);
```

```

// === Turbidity Conversion Constants ===
#define NTU_SLOPE    -1000.0
#define NTU_INTERCEPT 3000.0

// === Thresholds ===
#define TURBIDITY_HIGH 500
#define TURBIDITY_MED 200
#define TEMP_HIGH    35
#define TEMP_LOW     15

// --- Function to read turbidity ---
float readTurbidityNTU() {
    int raw = analogRead(PIN_TURB_ADC);
    float voltage = (raw / 1023.0) * 3.3; // ESP8266 ADC reference voltage
    float ntu = NTU_SLOPE * voltage + NTU_INTERCEPT;
    return (ntu < 0) ? 0 : ntu;
}

void setup() {
    Serial.begin(9600);

    // LEDs
    pinMode(LED_RED, OUTPUT);
    pinMode(LED_GREEN, OUTPUT);
    pinMode(LED_YELLOW, OUTPUT);
    digitalWrite(LED_RED, LOW);
    digitalWrite(LED_GREEN, LOW);

```

```
digitalWrite(LED_YELLOW, LOW);
```

```
// DS18B20
```

```
tempSensor.begin();
```

```
// Initialize I2C with custom pins
```

```
Wire.begin(OLED_SDA, OLED_SCL);
```

```
// Initialize OLED
```

```
if(!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) {
```

```
    Serial.println("SSD1306 allocation failed");
```

```
    for(;;);
```

```
}
```

```
display.clearDisplay();
```

```
display.setTextSize(1);
```

```
display.setTextColor(SSD1306_WHITE);
```

```
display.setCursor(0,0);
```

```
display.println("Water Quality Monitor");
```

```
display.display();
```

```
Serial.println("System Started...");
```

```
}
```

```
void loop() {
```

```
    // --- Read Sensors ---
```

```
    tempSensor.requestTemperatures();
```

```
float tempC = tempSensor.getTempCByIndex(0);
float ntu = readTurbidityNTU();

// --- Serial Output ---
Serial.print("Temp: "); Serial.print(tempC); Serial.print(" C | ");
Serial.print("Turbidity: "); Serial.print(ntu); Serial.println(" NTU");

// --- OLED Display ---
display.clearDisplay();
display.setCursor(0, 0);
display.setTextSize(1);
display.setTextColor(SSD1306_WHITE);
display.println("Water Quality Monitor");
display.print("Temp: "); display.print(tempC); display.println(" C");
display.print("Turbidity: "); display.print(ntu); display.println(" NTU");

// --- Turbidity Check ---
if (ntu > TURBIDITY_HIGH) {
    digitalWrite(LED_RED, HIGH);
    digitalWrite(LED_GREEN, LOW);
    display.println("Water: VERY DIRTY");
}
else if (ntu > TURBIDITY_MED) {
    digitalWrite(LED_RED, HIGH);
    digitalWrite(LED_GREEN, LOW);
    display.println("Water: MOD DIRTY");
}
```

```
else {  
    digitalWrite(LED_RED, LOW);  
    digitalWrite(LED_GREEN, HIGH);  
    display.println("Water: CLEAR");  
}  
  
// --- Temperature Check ---  
if (tempC > TEMP_HIGH || tempC < TEMP_LOW) {  
    digitalWrite(LED_YELLOW, HIGH);  
    display.println("Temp: OUT OF RANGE");  
} else {  
    digitalWrite(LED_YELLOW, LOW);  
    display.println("Temp: NORMAL");  
}  
  
display.display();  
delay(3000); // wait 3 seconds  
}
```