User Manual: Optimized ESP8266 Environmental Sensor (SHT31)

Welcome to the official guide for your upgraded Temperature, Humidity, and VPD (Vapor Pressure Deficit) monitor. This version features a high-precision Sensirion SHT31 sensor and completely overhauled firmware for enhanced stability, responsiveness, and functionality.

1. Introduction

This device is designed to provide accurate environmental readings and integrate seamlessly into smart home or data monitoring systems. The core of this upgrade is the replacement of the previous sensor with the industry-grade **Sensirion SHT31**, offering superior accuracy and reliability for temperature and humidity measurements.

The firmware has been re-engineered from the ground up to be fully non-blocking, ensuring the device is always responsive. It serves a mobile-friendly web interface for viewing data and configuration, and it publishes sensor readings to an MQTT broker for easy integration with platforms like Home Assistant, Node-RED, or other IoT systems.

2. Key Features

- **High-Precision Sensing:** Utilizes the Sensirion SHT31 sensor for accurate temperature and humidity readings via I2C.
- **Vapor Pressure Deficit (VPD):** Automatically calculates and displays VPD in kilopascals (kPa), a critical metric for optimizing plant growth.
- **Robust Web Interface:**
    - A clean, auto-refreshing dashboard to view live data.
    - An easy-to-use configuration portal for WiFi and MQTT setup.
    - A web-based firmware updater.
- **MQTT Protocol Support:** Publishes sensor data to configurable MQTT topics, allowing for robust home automation integration.
- **Resilient Connectivity:** Features non-blocking WiFi and MQTT connection management. If the MQTT broker is unavailable, the device will attempt to reconnect with an intelligent exponential backoff strategy to avoid network spam.
- **Over-The-Air (OTA) Updates:** Supports firmware updates both through the web interface and the Arduino IDE.
- **Highly Optimized Firmware:**
    - **Memory Efficient:** Stores static web content in flash memory to preserve RAM for core operations.

- **Stable:** Designed for long-term operation without the need for periodic reboots.

3. First-Time Setup (AP Mode)

If the device has not been configured or cannot connect to a known WiFi network, it will automatically enter Access Point (AP) mode.

1. Using your phone or computer, scan for WiFi networks.
2. Connect to the network with the following credentials:
   - **SSID:** `TempHumSensor-Config`
   - **Password:** `password`
3. Once connected, your device may automatically open a "captive portal" page. If not, open a web browser and navigate to `http://192.168.4.1`.
4. You will be presented with the **"WiFi & MQTT Config"** page.
5. Enter your home WiFi network's **SSID** and **Password**.
6. Enter the IP address or hostname of your **MQTT Broker**.
7. Click **"Save & Reboot"**. The device will save your settings and restart.

4. Normal Operation

After configuration, the device will reboot and connect to the WiFi network you provided.

4.1. Accessing the Web Interface

You can access the sensor's dashboard from any device on the same network. Open a web browser and navigate to its IP address. You can find the IP address from your router's client list or by listening to the serial monitor on boot.

Alternatively, the device uses mDNS, so you might be able to access it at:
`http://TempHumSensor.local`

4.2. Main Dashboard

The main page displays the latest sensor readings and is designed to be clear and concise.

- **Readings:** Shows Temperature (in ◦C), Relative Humidity (in %), and Vapor Pressure Deficit (in kPa).
- **Auto-Refresh:** The page automatically refreshes every 20 seconds to load the latest data.
- **Navigation:**
  - **Configure:** Takes you to the configuration page to update WiFi or MQTT settings.
  - **Update FW:** Takes you to the firmware update page.

5. MQTT Integration

The sensor publishes its data to an MQTT broker, making it a powerful tool for data logging and automation.

- **Broker:** The MQTT broker address is set on the configuration page. The port is fixed at `1883`.
- **Client ID:** The device identifies itself to the broker as `TempHumClient`.
- **Topics:** The sensor publishes data to the following topics:
    - `H-T-Sensor2/temperature`: The temperature reading (e.g., `24.50`).
    - `H-T-Sensor2/humidity`: The relative humidity reading (e.g., `55.25`).
    - `H-T-Sensor2/vpd`: The calculated Vapor Pressure Deficit (e.g., `1.25`).

6. Firmware Updates (OTA)

You can update the device's firmware without needing a physical connection.

6.1. Web-Based Update

This is the easiest method for most users.

1. Compile your new firmware in the Arduino IDE or PlatformIO.
2. Export the compiled binary file (e.g., `firmware.bin`).
3. Navigate to the device's web interface and click the **"Update FW"** button (or go directly to `/update`).
4. Click **"Choose File"**, select your `.bin` file, and click **"Update Firmware"**.
5. The device will upload the file, flash the new firmware, and reboot automatically.

6.2. Arduino OTA

For developers, updates can be sent directly from the Arduino IDE.

1. Make sure your computer is on the same network as the sensor.
2. In the Arduino IDE, go to **Tools > Port**.
3. The device should appear as `TempHumSensor at <IP_Address>`. Select it.
4. Click the "Upload" button as you normally would. The IDE will send the new firmware over the network.

7. Troubleshooting

- **Cannot find SHT31 Sensor:** If the serial monitor shows "Couldn't find SHT31 sensor!", please check the I2C wiring between the ESP8266 and the sensor. The device will continue to function but will not be able to provide readings.
- **Device Reverts to AP Mode:** If the device cannot connect to the configured WiFi (e.g., due to a wrong password or weak signal), it will revert to AP Mode after 15 seconds of trying. Reconnect to the `TempHumSensor-Config` network to re-enter your credentials.
- **MQTT Disconnected:** The device will continuously try to reconnect to the MQTT broker if the connection is lost. It uses an exponential backoff algorithm, waiting longer between each failed attempt (from 5 seconds up to 2 minutes) to avoid flooding your network. It will automatically resume publishing data once the connection is restored.