

## FICEMETOI B0DM9L458H

# SHT40 Temperature and Humidity Sensor Module User Manual

Model: B0DM9L458H

## 1. INTRODUCTION

This manual provides comprehensive instructions for the SHT40 Temperature and Humidity Sensor Breakout Module. This digital output sensor module utilizes an IIC (I2C) interface and is designed for operation with 3.3V to 5V power supplies. It offers reliable and accurate measurements for various applications requiring environmental monitoring.

## 2. PRODUCT FEATURES

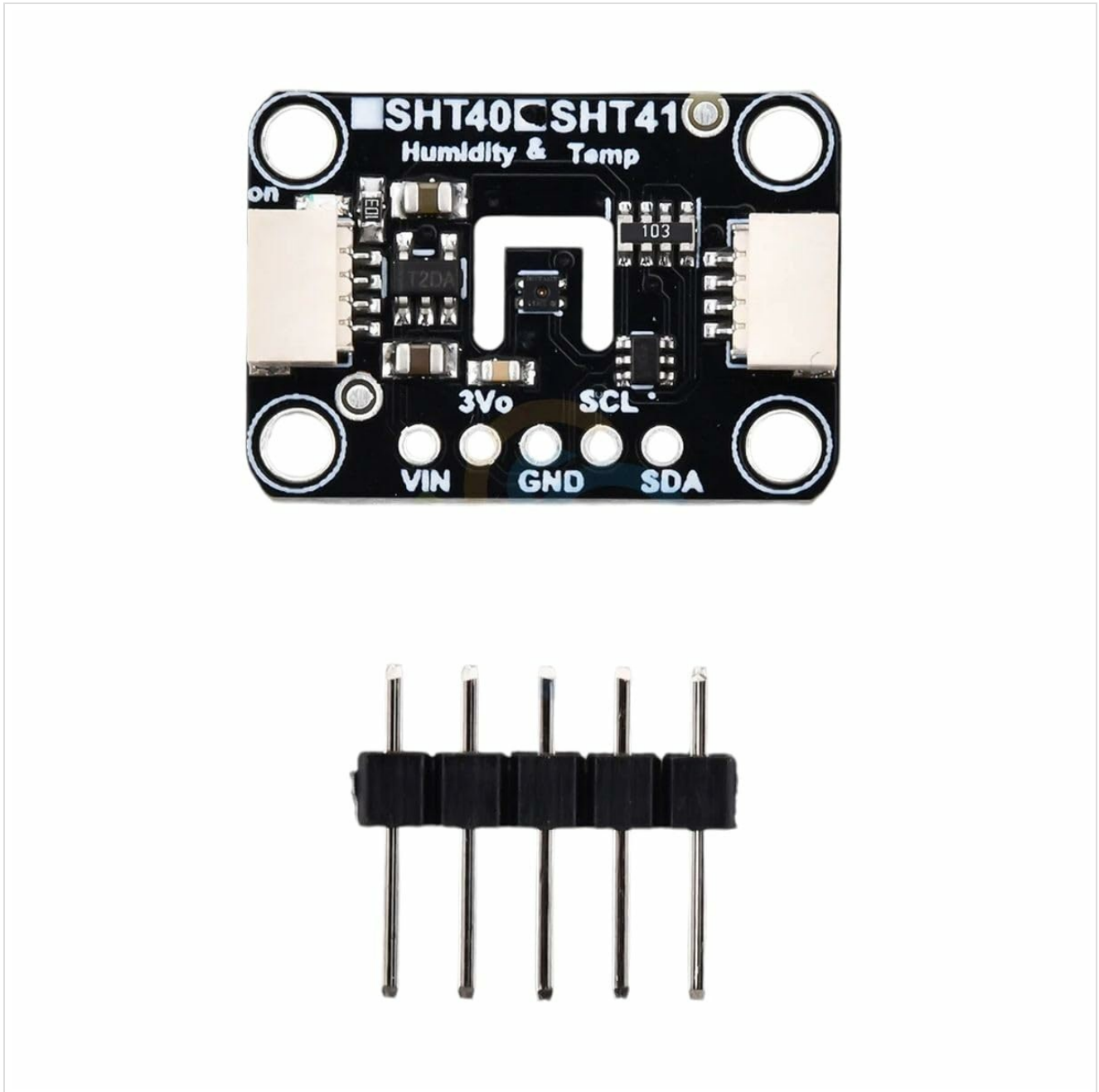
- **High-Quality Electronics Components:** Constructed with top-of-the-line components for reliable and long-lasting performance.
- **Easy to Install and Use:** Designed for user-friendliness with clear instructions and simple installation.
- **Versatile Applications:** Suitable for a variety of applications, including industrial, automotive, and household electronics.
- **Digital Output:** Provides digital temperature and humidity readings via I2C.
- **Wide Voltage Compatibility:** Operates with both 3.3V and 5V power supplies.

## 3. SPECIFICATIONS

Parameter	Value
Model	SHT40 Breakout Module
Interface	IIC (I2C)
Operating Voltage	3.3V - 5V
Sensor Type	Temperature and Humidity

Parameter	Value
Manufacturer	FICEMETOI
ASIN	B0DM9L458H

## 4. PRODUCT OVERVIEW



**Figure 1:** SHT40 Temperature and Humidity Sensor Breakout Module. The image displays the compact black PCB with clearly labeled pins (VIN, GND, SDA, SCL, 3Vo) and the sensor component. Below the module, a set of male header pins is shown, which can be soldered to the module for breadboard compatibility.

## 5. SETUP AND CONNECTION

The SHT40 sensor module uses the I2C communication protocol, requiring four connections: VCC (VIN), GND, SDA, and SCL.

### 5.1 Pinout Description

- **VIN:** Voltage Input (3.3V - 5V DC). Connect this to your microcontroller's 3.3V or 5V power supply.
- **GND:** Ground. Connect this to the ground of your microcontroller.
- **SDA:** Serial Data Line. This is the I2C data line. Connect to your microcontroller's SDA pin.
- **SCL:** Serial Clock Line. This is the I2C clock line. Connect to your microcontroller's SCL pin.
- **3Vo:** 3.3V Output. This pin provides a regulated 3.3V output, useful for powering other low-power components if needed.

## 5.2 Connection Steps

1. Ensure your microcontroller is powered off before making connections.
2. Connect the **VIN** pin of the SHT40 module to the 3.3V or 5V power output of your microcontroller (e.g., Arduino 5V or 3.3V pin, ESP32 3.3V pin).
3. Connect the **GND** pin of the SHT40 module to the Ground (GND) pin of your microcontroller.
4. Connect the **SDA** pin of the SHT40 module to the I2C SDA pin of your microcontroller (e.g., Arduino Uno A4, ESP32 GPIO21).
5. Connect the **SCL** pin of the SHT40 module to the I2C SCL pin of your microcontroller (e.g., Arduino Uno A5, ESP32 GPIO22).
6. (Optional) Solder the included header pins to the module for easier breadboard integration.

## 6. OPERATING INSTRUCTIONS

To operate the SHT40 sensor, you will typically use a microcontroller (like Arduino, ESP32, Raspberry Pi) and a suitable library for I2C communication with the SHT40 sensor.

### 6.1 Software Setup (Example: Arduino IDE)

1. Install the Arduino IDE if you haven't already.
2. Open the Arduino IDE and navigate to *Sketch > Include Library > Manage Libraries...*
3. Search for "SHT40" or "Sensirion SHT4x" and install a compatible library (e.g., "Adafruit SHT4x Library").
4. Once installed, you can find example sketches under *File > Examples > [Installed Library Name]*

### 6.2 Basic Code Example (Conceptual)

The following is a conceptual outline of how to read data. Refer to the specific library's documentation and examples for exact syntax.

```

#include <Wire.h> // Required for I2C communication
#include <Adafruit_SHT4x.h> // Example library

Adafruit_SHT4x sht40 = Adafruit_SHT4x();

void setup() {
  Serial.begin(115200);
  Serial.println("SHT40 Test");
  if (!sht40.begin()) {
    Serial.println("Couldn't find SHT40");
    while (1) delay(10);
  }
  Serial.println("SHT40 Found!");
}

void loop() {
  sensors_event_t humidity, temp;
  sht40.getEvent(&humidity, &temp); // Populate temp and humidity objects
  Serial.print("Temperature: "); Serial.print(temp.temperature); Serial.println(" °C");
  Serial.print("Humidity: "); Serial.print(humidity.relative_humidity); Serial.println(" %");
  delay(1000);
}

```

This example demonstrates initializing the sensor and reading temperature in Celsius and relative humidity percentage. Adjust the delay as needed for your application.

## 7. MAINTENANCE

- **Cleaning:** Keep the sensor clean and free from dust or debris. Use a soft, dry brush or compressed air if necessary. Avoid using liquids or harsh chemicals.
- **Environment:** Operate the sensor within its specified temperature and humidity ranges to ensure optimal performance and longevity. Avoid exposure to corrosive gases or extreme conditions.
- **Storage:** When not in use, store the module in a dry, anti-static environment, preferably in its original packaging.
- **Physical Handling:** Handle the module by its edges to avoid touching the sensor element directly or damaging components.

## 8. TROUBLESHOOTING

Problem	Possible Cause	Solution
Sensor not detected (e.g., "Couldn't find SHT40")	Incorrect wiring (SDA/SCL, VCC/GND), incorrect I2C address, power supply issue, faulty module.	Double-check all connections. Ensure correct voltage (3.3V-5V). Verify I2C address in code (usually default). Try a different I2C port or microcontroller.
Inaccurate readings	Sensor exposed to extreme conditions, dust/debris on sensor, interference, incorrect library usage.	Ensure sensor is in a stable environment. Clean the sensor gently. Check for strong electromagnetic interference. Review library documentation for proper usage and calibration if available.
Module not powering on	No power supply, reversed polarity, short circuit.	Verify power connections (VIN and GND). Check power supply voltage. Inspect for any visible damage or short circuits on the PCB.

## 9. WARRANTY AND SUPPORT

This product comes with a money-back guarantee, reflecting our confidence in its quality. We strive for perfection and customer satisfaction.

For any questions, concerns, or technical assistance, please contact our customer support team. We pride ourselves on providing exceptional customer support with a knowledgeable team ready to assist you.

**Return Policy:** You may return the product within 30 days for a refund or replacement, as per the standard return policy.