

How to use DHT11 sensor with ESP32

**Introduction**

The DHT11 sensor is well known and a cost-effective device used for measuring temperature and humidity for multiple projects and applications across the domain of electronics. When it is combined with the ESP32 microcontroller, which is known to be a powerful and versatile development board, it opens up a plethora of possibilities for real-time environmental monitoring and control. In this blog, we will explore how to effectively use the DHT11 sensor with the ESP32 to acquire temperature and humidity data accurately.

The sensor employs a digital sensor module that integrates a calibrated temperature and humidity sensing element along with a signal conditioning circuit. A simple and convenient solution is offered by this integration for measuring environmental parameters, making it suitable for home automation, weather stations, agricultural monitoring, and more.

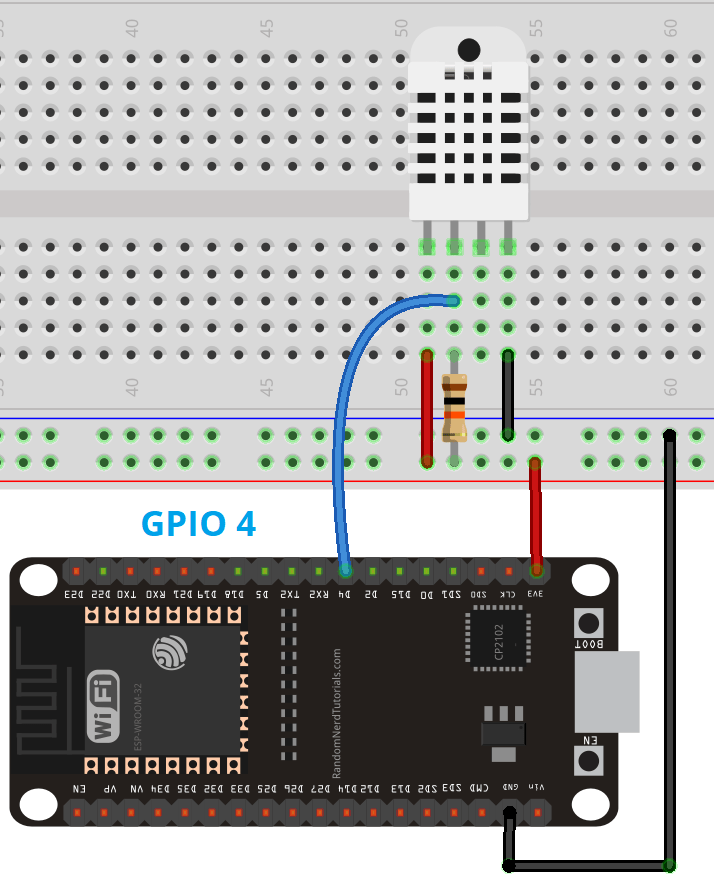
The ESP32, equipped with Wi-Fi and Bluetooth capabilities, provides a platform for connecting and interfacing with the DHT11 sensor. With its high processing power, memory, and a wide range of available GPIO pins, the ESP32 offers the flexibility to read data from the DHT11 sensor and transmit it wirelessly or process it locally for various applications.

By the end of this tutorial, you will have a solid understanding of how to effectively integrate the DHT11 sensor with the ESP32 and use the set up to effectively acquire temperature and humidity data. So, let's deep dive into the exciting world of sensor integration and unleash the potential of temperature and humidity measurement with the ESP32 and DHT11.

To make this project, you will need the following components:

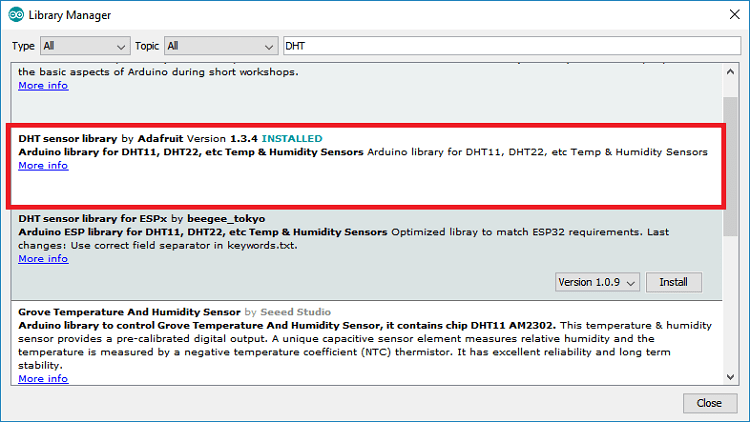
1. ESP32 Development Board: This is the main microcontroller board used to interface with the DHT11 sensor. It offers built-in Wi-Fi and Bluetooth capabilities, GPIO pins, and sufficient processing power for data acquisition and transmission.
2. DHT11 Sensor: As already discussed, the DHT11 is a digital temperature and humidity sensor module. It consists of a sensing element, signal conditioning circuitry, and a digital interface.
3. Jumper Wires: Male-to-female jumper wires will be required to make the necessary electrical connections between the ESP32 and the DHT11 sensor.
4. Breadboard : It’s used to create a temporary connection platform for prototyping purposes. It allows for easy and flexible arrangement of components and simplifies the wiring process.

**Circuit Diagram**



Here, we have connected the DHT11 data pin to GPIO 4. You are free to use any other suitable digital pin.

In order to read from the DHT11 sensor, we’ll use the DHT library from Adafruit. To use this library you need to install the Adafruit Unified Sensor library in the Arduino IDE.



Code :

#include "DHT.h"

#define DHTPIN 4

#define DHTTYPE DHT11

// Initialize DHT sensor.

.

DHT dht(DHTPIN, DHTTYPE);

void setup() {

Serial.begin(9600);

Serial.println(F("DHTxx test!"));

dht.begin();

}

void loop() {

// Wait a few seconds between measurements.

delay(2000);

// Reading temperature or humidity takes about 250 milliseconds!

float h = dht.readHumidity();

// Read temperature as Celsius (the default)

float t = dht.readTemperature();

// Read temperature as Fahrenheit (isFahrenheit = true)

float f = dht.readTemperature(true);

// Check if any reads failed and exit early (to try again).

if (isnan(h) || isnan(t) || isnan(f)) {

Serial.println(F("Failed!"));

return;

}

// Compute heat index in Fahrenheit (the default)

float hif = dht.computeHeatIndex(f, h);

// Compute heat index in Celsius (isFahreheit = false)

float hic = dht.computeHeatIndex(t, h, false);

Serial.print(F("Humidity: "));

Serial.print(h);

Serial.print(F("% Temperature: "));

Serial.print(t);

Serial.print(F("°C "));

Serial.print(f);

Serial.print(F("°F Heat index: "));

Serial.print(hic);

Serial.print(F("°C "));

Serial.print(hif);

Serial.println(F("°F"));

}

Once the code is uploaded, open the Serial Monitor at a baud rate of 9600. You will see the latest temperature and humidity readings in the Serial Monitor every two seconds.

**Working**

The microcontroller sends a start signal to the sensor, initiating the data transmission. The sensor responds by sending the temperature and humidity data in a serial format. The ESP32 reads this data and decodes it to obtain the temperature and humidity values.

The DHT11 sensor employs a single-wire communication protocol with time-based data transmission. To represent distinct pieces of information, the sensor generates a digital signal that alternates between high and low levels.

Using a GPIO pin set as an input, the ESP32 reads the digital signal from the DHT11 sensor. To decode the data bits and extract the temperature and humidity values, it measures the duration of the high and low signals. Once received, the data can be processed, presented, or transferred to other devices or platforms.

Also, it's important to note that the DHT11 sensor has limitations in terms of accuracy and speed compared to more advanced sensors. However, it still provides reasonably accurate temperature and humidity measurements for most applications.

If the application is not working properly, do take note of the points below:

1. Wiring: check for all kinds of loose connections and fix them to check the working conditions.
2. Power: the DHT11 sensor has an operating range of 3V to 5.5V (DHT11) or 3V to 6V (DHT22). If the sensor is powered from the ESP32 3.3V pin, in some cases powering the DHT11 with 5V will mitigate the problem.
3. Bad USB port or USB cable: Often powering the ESP32 directly from a PC USB port is not enough. Hence, we should try to plug it to a USB hub powered by an external power source.
4. Power source: As said above, the ESP might not be supplying enough power to properly read from the DHT sensor. In some cases, you might need to power the ESP with a power source that provides more current.
5. Sensor type: double-check that you’ve mentioned right sensor name for your code. In this project, we were using the DHT11. Also install the required libraries.

Summary

The combination of the DHT11 sensor with the ESP32 microprocessor provides a simple, effective and low-cost solution for temperature and humidity monitoring. The DHT11 sensor gives precise and accurate readings because of its digital sensing module, while the ESP32 provides the computing capacity and connection necessary for data processing and transfer. This combination permits the creation of a wide range of applications, including home automation, weather monitoring, and environmental management. By using these components' features, hobbyists and developers may design systems that give significant data regarding temperature and humidity conditions. The combination of the DHT11 sensor with the ESP32, whether for personal or business applications, offers up a world of possibilities for real-time environmental monitoring, improving efficiency, comfort, and informed decision-making.