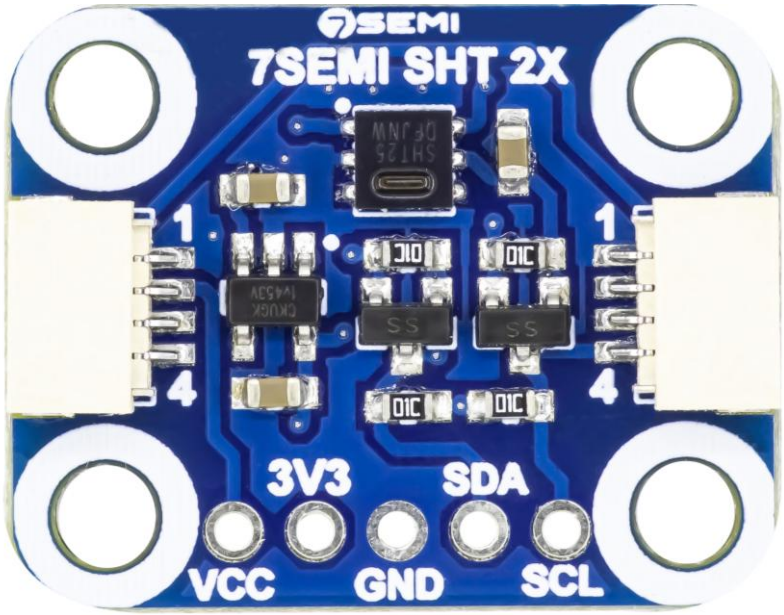
7SEMI

SHT20 Digital Humidity and Temperature Sensor I2C Qwiic Breakout

Version 1.0

Table of Contents

1.Introduction



The **7Semi SHT20 Digital Humidity and Temperature Sensor I2C Qwiic Breakout** integrates Sensirion's CMOSens® technology, delivering fully calibrated digital output signal. This compact and high-accuracy sensor is ideal for precise humidity and temperature measurements. Designed for reliability and efficiency, the SHT20 supports I²C communication and operates across a wide voltage range, making it suitable for various embedded and IoT applications

1.1 Features

- High accuracy: ±3.0 %RH, ±0.3 °C (typical)
- Operating range: 0% to 100% RH, -40 to 125 °C
- Fully calibrated and digital output
- I²C digital interface (0x40 default address)
- Ultra-low power consumption: 0.4 µA (sleep mode), ~300 µA (active)
- Long-term stability: <0.25 %RH/year
- Response time: <8s (humidity), <5-30s (temperature)
- Compatible with Arduino, ESP32, Raspberry Pi, STM32

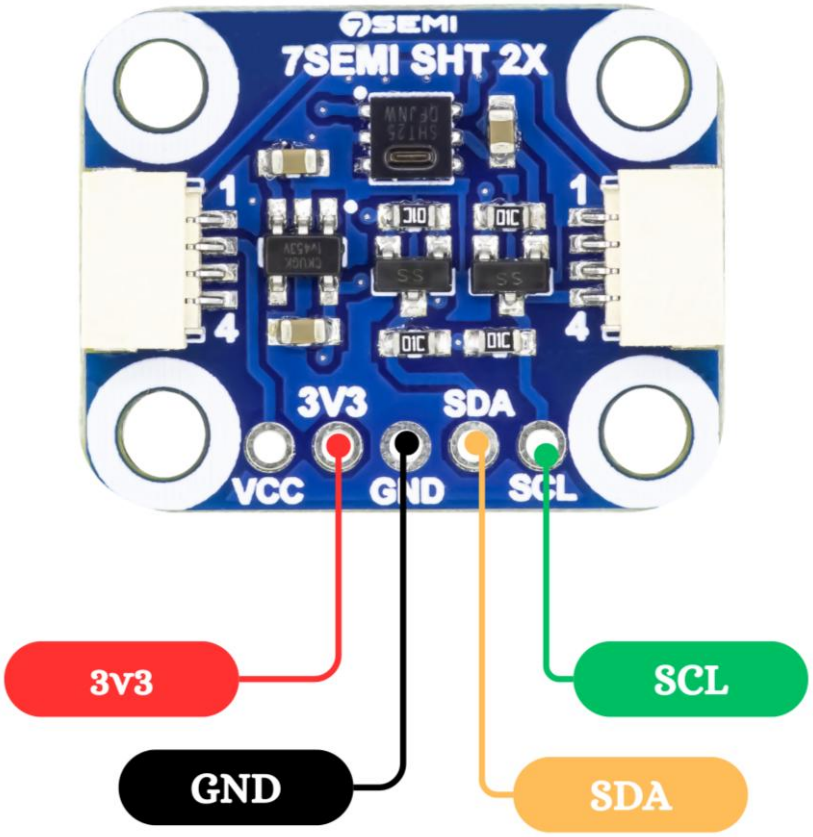
2.Technical Specification

The **Technical Specification** table provides detailed information about **7Semi SHT20 Digital Humidity and Temperature Sensor I2C Qwiic Breakout**, including its operating voltage, current consumption, and electrical characteristics. This data helps users understand the power requirements, communication parameters, and performance capabilities of the sensor. It ensures compatibility with different microcontrollers and embedded systems while providing guidelines for efficient integration into various applications.

SHT20 Specifications

- Relative Humidity Accuracy: ±3.0 %RH (typ.)
- Temperature Accuracy: ±0.3 °C (typ.)
- Supply Voltage: 1.08 V to 3.6 V
- Idle Current: 80 nA
- Current: Sleep Current:0.4 µA
  Active Current : 300 µA (typ.)
- Interface: I²C
- Probe Material: Stainless Steel
- Probe wire length: 1 meters
- Probe Dimensions: 14mm diameter, 93mm length

3.Pinouts

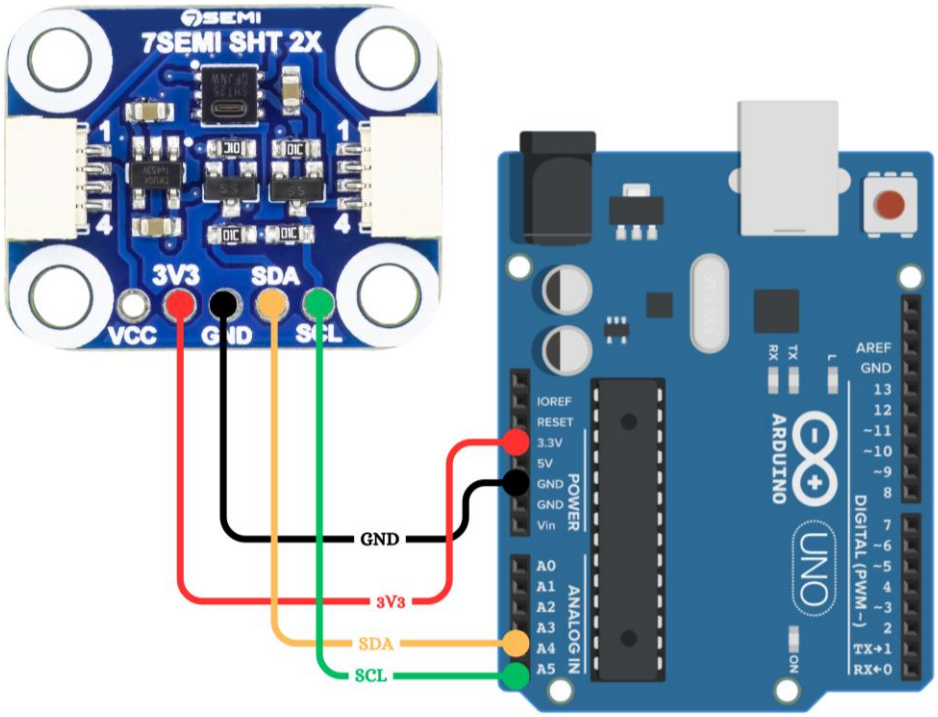


Pin	Name	Description
1	**VCC**	Power Input (3.3V DC)
2	**SCL**	I²C Clock Line
3	**SDA**	I²C Data Line
4	**GND**	Ground Connection

Connection Guidelines

- Ensure a stable 3.3V DC power source.
- Connect SCL to SCL and SDA to SDA for proper communication.

4.Hardware Interface



Connection Explanation :

- The **VCC** pin of the sensor is connected to **3.3V** on the Arduino UNO.
- The **GND** pin of the sensor is connected to the **GND** pin of the Arduino UNO to The **SCL (Clock Line)** of the sensor is connected to **A5 (SCL)** on the Arduino UNO.
- The **SDA (Data Line)** of the sensor is connected to **A4 (SDA)** on the Arduino UNO.
- These connections allow the **Arduino UNO** to communicate with the **SHT20 sensor** using the I²C protocol.

5.Example code link

We provide example codes to help you get started with the **7Semi SHT20 Digital Humidity and Temperature Sensor I2C Qwiic Breakout**. These examples demonstrate how to communicate with the sensor and retrieve temperature and humidity data using the I²C protocol. The code is available for two popular platforms: Arduino and ESP32.

Code Explanation

```
#include <Wire.h>

#include "SHTSensor.h"

SHTSensor sht;
// To use a specific sensor instead of probing the bus use this
command:
// SHTSensor sht(SHTSensor::SHT3X);

void setup() {
  // put your setup code here, to run once:

  Wire.begin();
  Serial.begin(9600);
  delay(1000); // let serial console settle

  if (sht.init()) {
      Serial.print("init(): success\n");
  } else {
      Serial.print("init(): failed\n");
  }
  sht.setAccuracy(SHTSensor::SHT_ACCURACY_MEDIUM); // only supported by
SHT3x

}

void loop() {
  // put your main code here, to run repeatedly:

  if (sht.readSample()) {
      Serial.print("SHT:\n");
      Serial.print("  RH: ");
      Serial.print(sht.getHumidity(), 2);
      Serial.print("\n");
      Serial.print("  T:  ");
      Serial.print(sht.getTemperature(), 2);
      Serial.print("\n");
  } else {
```

```
        Serial.print("Error in readSample()\n");
    }

    delay(1000);
}
```

1. Library Inclusions

- `#include <Wire.h>` → Enables I²C communication protocol.
- `#include "SHTSensor.h"` → Includes the SHTSensor library for SHT2x, SHT3x, SHT85 sensors.

2. Sensor Object and Initialization

- `SHTSensor sht;` → Creates a sensor object to interact with any supported SHT sensor (auto-detect).
- `// SHTSensor sht(SHTSensor::SHT3X);` → (Optional) Manually specifies sensor model if known.

3. Sensor Initialization (`setup()`)

- Begins I²C communication using `Wire.begin()`.
- Starts serial communication at **9600 baud** with `Serial.begin(9600)`.
- `delay(1000);` → Waits to allow the serial console to settle.
- `sht.init()` → Attempts to initialize the sensor.
    - If successful, prints `"init(): success"`.
    - Otherwise, prints `"init(): failed"`.
- `sht.setAccuracy(SHTSensor::SHT_ACCURACY_MEDIUM);`
    - Sets medium accuracy mode (Only for **SHT3x** sensors).

4. Measuring Temperature and Humidity (`loop()`)

- `sht.readSample()` → Reads sensor data.
    - If successful:
        - `sht.getHumidity()` → Returns humidity (%RH) with 2 decimal precision.
        - `sht.getTemperature()` → Returns temperature (°C) with 2 decimal precision.
        - Prints formatted results to Serial Monitor.
    - If failed:
        - Prints `"Error in readSample()"`.
- `delay(1000);` → Waits for 1 second before the next reading (sampling rate: 1Hz).

5. Arduino Example Code

This example is designed for Arduino-compatible boards and demonstrates:

- Initializing the I²C communication with the SHT20 sensor Breakout.
- Reading temperature and Humidity data from the sensor.

- Printing the sensor data to the Serial Monitor.

6. ESP32 Example Code

This example targets ESP32 boards and showcases:

- Configuring the ESP32 I²C interface to communicate with the SHT20 sensor Breakout.
- Reading temperature and Humidity data from the sensor.
- Printing the sensor data to the Serial Monitor.

How to Access the Code

- **Download Link for Arduino and ESP32 Example**: Click Here

5.1 Sample Serial Output Arduino

Sample output Image of Arduino:

```
init(): success
SHT:
  RH: 45.73
  T:  24.61
SHT:
  RH: 45.71
  T:  24.63
SHT:
  RH: 45.69
  T:  24.64
SHT:
  RH: 45.66
  T:  24.62
```

6.Mechanical Specification