Adriann Liceralde

Dr. Brian McPherson

Carbon Science Research at the University of Utah

CSR Arduino Collection Project

Last Updated February 20, 2022

**Information for the SCD30 CO2 Sensor**

**Introduction**

* The SCD30 is a low-cost NDIR-based sensor that measures CO2, temperature, and humidity.
* Any microcontroller can control this sensor. However, this guide and the associative codes use an Arduino Uno to operate the device.
* It is compatible with three different communication methods: I2C, UART, and PWM. This guide only covers how to use I2C.
* A separate device, the Bi-Directional Logic Level Converter, is needed to operate the sensor more reliably.

**Important Notes**

* This device is EXTREMELY sensitive to electrostatic discharge.
* NEVER touch the sensor UNLESS wearing Anti-Static Gloves.
* Even if the sensor is not connected to a power source, exercise extreme caution.
* DO NOT place the SCD30 in direct contact with sunlight.

**Specifications**

|  |  |
| --- | --- |
|  | **Via I2C** |
| **Range** | 0 – 40,000 ppm |
| **Accuracy** | ± (30 ppm + 3%MV) for 400-10000 ppm |
| **Repeatability** | ± 10 ppm |
| **Frequency** | 2.1 sec |
| **Current** | 19 mA (avg)  75 mA (max) |
| **Voltage** | 3.3 V - 5.5 V |
| **Response Time** | 20 sec (τ63%) |
| **Lifespan** | 15 years |

**Links**

* Product Info for Sensor:

<https://sensirion.com/products/catalog/SCD30/>

* Product Info for Level Converter:

<https://www.sparkfun.com/products/12009>

* Arduino Library:

<https://github.com/sparkfun/SparkFun_SCD30_Arduino_Library>

* Arduino Code

<https://github.com/RiceAllDay22/EGI_Arduino_Collection/tree/main/SCD30>

**Wiring**

* To operate the sensor, a minimum of 4 pins are required to be connected to an Arduino:
  + VIN (Voltage Input)
  + GND (Ground)
  + SCL (Clock Line for I2C communication)
  + SDA (Data Line for I2C communication)
* The sensor can accept a voltage input range of 3.3V to 5.5 V.
* The I2C pins can be at 5V but are highly recommended to be at 3.3 V.
* Therefore, the sensor can be wired up in two ways.

**Wiring – Method #1 (Recommended for reliability)**

* This method powers the sensor at 5V with the SCL and SDA lines operating at 3.3 V using a Bi-Directional Logic Converter.
* Below is a wiring diagram and a schematic diagram of the circuit.

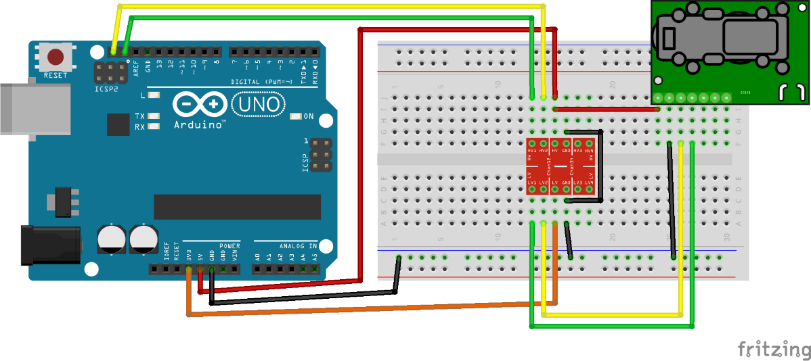


Figure 1. Wiring Diagram for I2C lines at 3.3 V

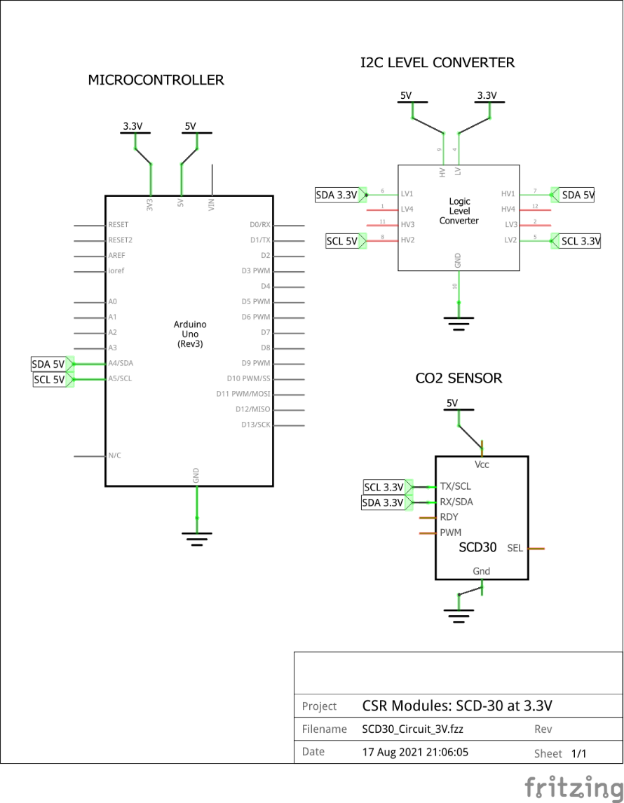


Figure 2. Schematic Diagram for I2C lines at 3.3 V

**Wiring – Method #2 (Not recommended, but is simpler)**

* This method powers the sensor at 5V with the SCL and SDA lines operating at 5 V.

Below is a wiring diagram and a schematic diagram of the circuit.

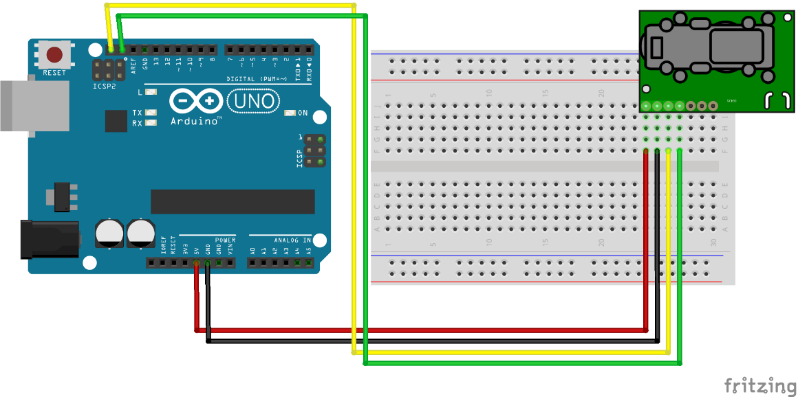


Figure 3. Wiring Diagram for I2C lines at 5 V

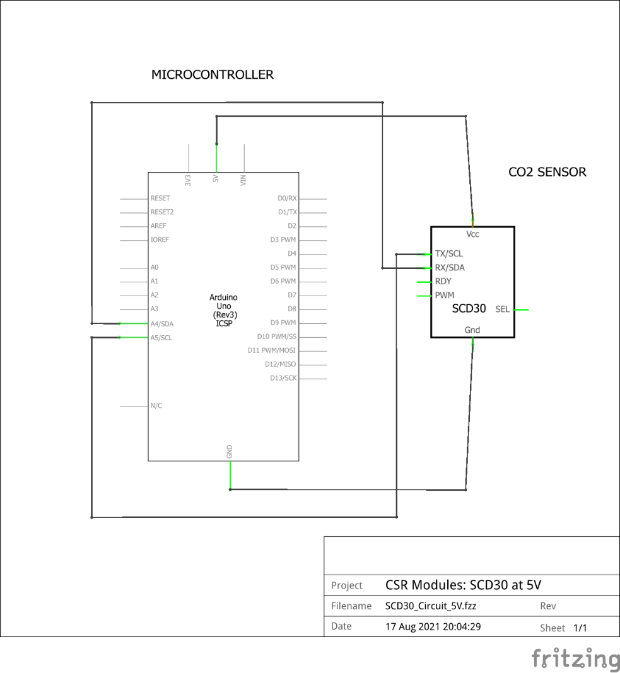


Figure 4. Wiring Diagram for I2C lines at 5 V

**Calibration**

* Manual calibrations occur via software.
* The following line of Arduino code performs a calibration to 1000 ppm:
  + airSensor.setForcedRecalibrationFactor(1000)
  + The number inside the parentheses can is changeable to any value

**Contact**

For any questions or assistance, email Adriann Liceralde at adriann8399@gmail.com.