# Arduino UNO R4 Minima read and write External AT24C02 EEPROM Module via I2C Protocol

## Description

In this project, we will learn how to interface an Arduino Uno R4 Minima with an external EEPROM (Electrically Erasable Programmable Read-Only Memory) module using the I2C (Inter-Integrated Circuit) protocol. The EEPROM module used here is the AT24C02, which provides non-volatile memory storage for your Arduino projects. We will also incorporate analog sensors such as a raindrop sensor and a soil moisture sensor to demonstrate how to read and write data to the EEPROM module, thereby storing sensor readings persistently.



## How-To Guide

### 1. Gather Components:

* 1 x Arduino Uno R4 Minima
* 1 x Breadboard
* 1 x AT24C02 EEPROM Module
* 1 x Raindrop Sensor
* 1 x Soil Moisture Sensor
* Jumper wires

### 2. Assemble the Circuit

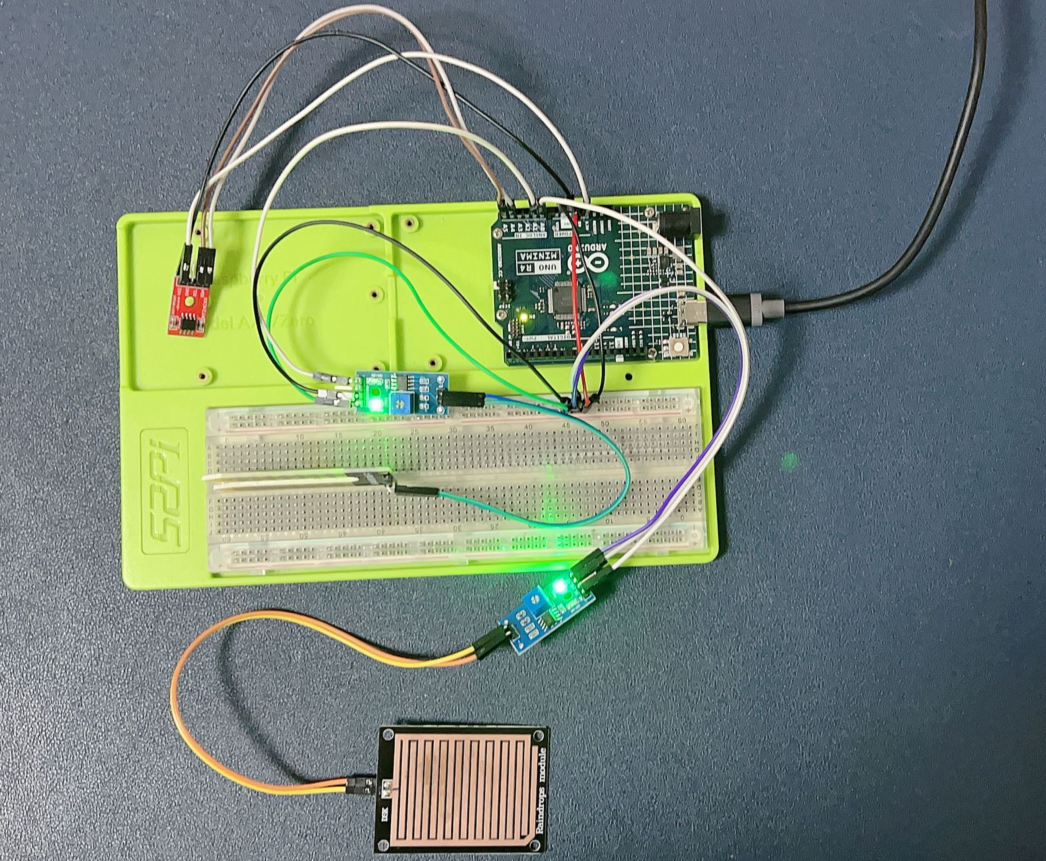
* Connect the VCC and GND pins of the AT24C02 EEPROM module to the 5V and GND rails on the breadboard, respectively.
* Connect the SDA (Serial Data) pin of the EEPROM module to the SDA pin (Analog pin 4) of the Arduino Uno R4.
* Connect the SCL (Serial Clock) pin of the EEPROM module to the SCL pin (Analog pin 5) of the Arduino Uno R4.
* Connect the raindrop sensor and soil moisture sensor to the appropriate analog pins on the Arduino Uno R4 Minima.
* Ensure proper power and ground connections for all components.

#### Wiring Diagram

|  |  |
| --- | --- |
| Arduino UNO R4 Minima | AT24C02 EEPROM Module |
| 3.3V | VCC |
| GND | GND |
| A4 | SDA |
| A5 | SCL |

|  |  |
| --- | --- |
| Arduino UNO R4 Minima | Rain Drop Sensor |
| 5V | VCC |
| GND | GND |
| A0 | AO |
| NC - Not connect | DO |

|  |  |
| --- | --- |
| Arduino UNO R4 Minima | Soil Moisture Sensor |
| 5V | VCC |
| GND | GND |
| A1 | AO |
| NC - Not connect | DO |



### 3. Write the Arduino Code

Open the Arduino IDE on your computer and create a new sketch.

#include <Wire.h>

#include <EEPROM.h>

// Define EEPROM memory address

#define EEPROM\_ADDRESS 0x50  // EEPROM I2C address

// Define analog sensor pins

const int raindropSensorPin = A0;

const int soilMoistureSensorPin = A1;

void setup() {

    Serial.begin(9600);  // Initialize serial communication

    Wire.begin();        // Initialize I2C communication

}

void loop() {

    // Read analog sensor values

    int raindropValue = analogRead(raindropSensorPin);

    int soilMoistureValue = analogRead(soilMoistureSensorPin);

    // Write sensor readings to EEPROM

    writeEEPROM(raindropValue, soilMoistureValue);

    delay(1000);  // Delay for stability

    // Read sensor readings from EEPROM and print to serial monitor

    readEEPROM();

    delay(5000);  // Delay for stability

}

void writeEEPROM(int raindropValue, int soilMoistureValue) {

    // Write sensor readings to EEPROM

    Wire.beginTransmission(EEPROM\_ADDRESS);  // Begin transmission to EEPROM

    Wire.write(0);                            // Set memory address pointer to 0

    Wire.write((byte)(raindropValue >> 8));   // Write high byte of raindropValue

    Wire.write((byte)(raindropValue & 0xFF)); // Write low byte of raindropValue

    Wire.write((byte)(soilMoistureValue >> 8));   // Write high byte of soilMoistureValue

    Wire.write((byte)(soilMoistureValue & 0xFF)); // Write low byte of soilMoistureValue

    Wire.endTransmission();                   // End transmission

}

void readEEPROM() {

    // Read sensor readings from EEPROM and print to serial monitor

    int raindropValue, soilMoistureValue;

    Wire.beginTransmission(EEPROM\_ADDRESS);  // Begin transmission to EEPROM

    Wire.write(0);                            // Set memory address pointer to 0

    Wire.endTransmission();                   // End transmission

    Wire.requestFrom(EEPROM\_ADDRESS, 4);      // Request four bytes from EEPROM

    if (Wire.available() >= 4) {

        raindropValue = (Wire.read() << 8) | Wire.read();  // Combine high and low bytes for raindropValue

        soilMoistureValue = (Wire.read() << 8) | Wire.read();  // Combine high and low bytes for soilMoistureValue

        Serial.print("Raindrop Sensor Value: ");

        Serial.println(raindropValue);

        Serial.print("Soil Moisture Sensor Value: ");

        Serial.println(soilMoistureValue);

    }

}

### 4. Understand the Code

The code initializes the Wire library for I2C communication and the EEPROM library for EEPROM access.

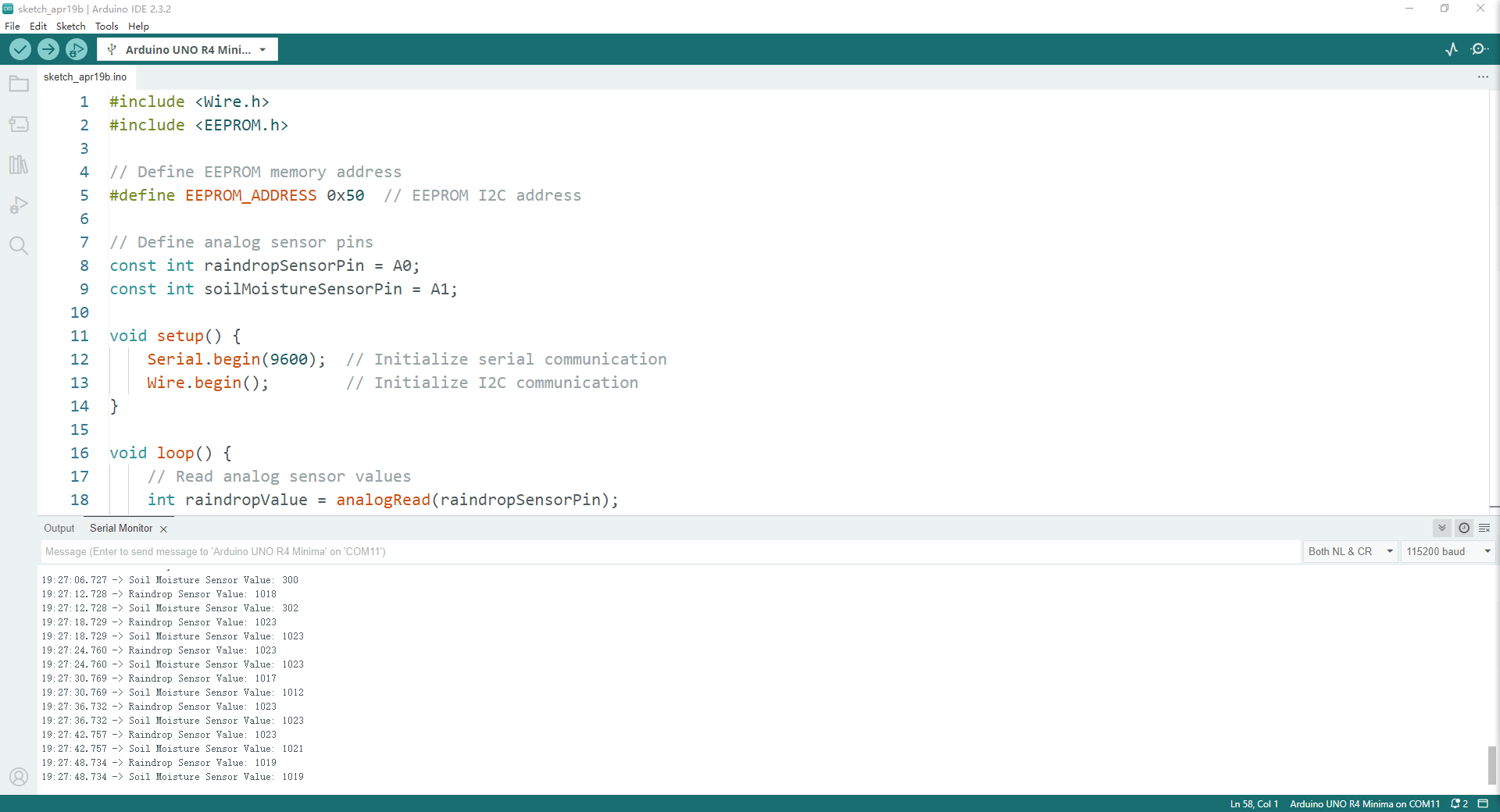
In the **loop()**, it reads sensor values from the raindrop sensor and soil moisture sensor, writes them to the EEPROM, reads them back from the EEPROM, and prints them to the serial monitor.

The **writeEEPROM()** function writes sensor readings to the EEPROM, and the **readEEPROM()** function reads sensor readings from the EEPROM.

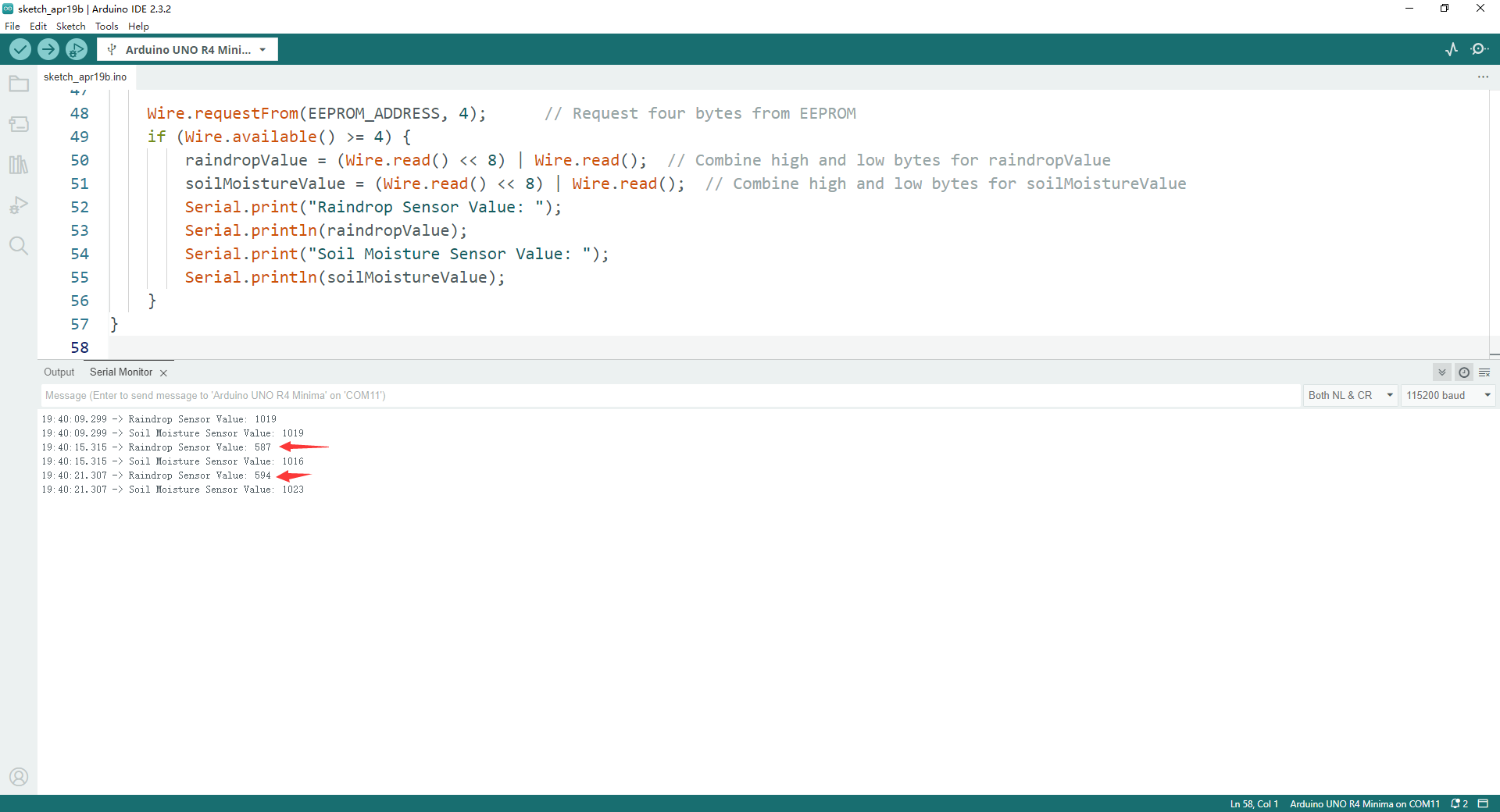
### 5. Test the Project

Upload the code to your Arduino Uno R4 Minima.

Open the serial monitor to observe the sensor readings being stored and retrieved from the EEPROM.



If drop some water on the sensor’s pad.



### 6. Experiment and Learn

Modify the code to store and retrieve additional sensor readings or other data types.

Explore different EEPROM addresses for storing multiple sets of data.

Consider implementing error-checking mechanisms for EEPROM read and write operations for robustness.

By following these steps, you can successfully interface an external EEPROM module with the Arduino Uno R4 Minima via the I2C protocol, allowing for persistent storage of sensor readings and other data in your Arduino projects.