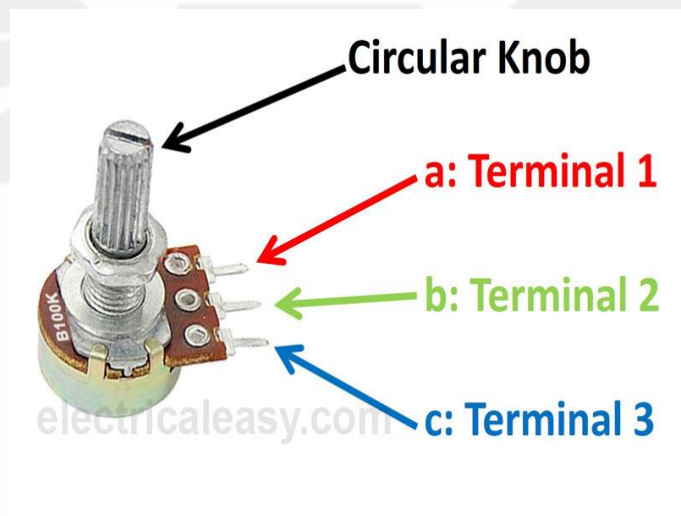


LAUNCH THE PROJECT - 03

Read Analog Voltage

(via 10K Potentiometer)

On AYNOP® UNO Launchpad Kit



Author: AYNOP Enterprises | Doc Version: 1.0 | Date: 03-9-2025

Table of Contents

1. Overview.....	3
2. Components Required.....	3
3. Software Required.....	3
4. Hardware Setup	4
4.1 Wiring Diagram.....	4
4.2 Circuit Schematic	5
5. Principle – How It Works	6
6. Procedure – Steps to Run	7
6.1 How to Use the Serial Monitor:.....	8
7. Expected Output.....	8
8. Code.....	9
8.1 Function References.....	9
9. Troubleshooting Tips.....	10
10. License.....	11
11. Support & Feedback.....	11

1. Overview

This project demonstrates how to measure and display **analog voltage values** using a **10 kΩ potentiometer** and the Arduino® UNO R4 Minima.

You will learn to:

- Connect and use a **potentiometer** as an analog input device.
- Understand how the **ADC (Analog-to-Digital Converter)** converts signals into digital values.
- Display real-time voltage values using the **Serial Monitor**.

2. Components Required

- Arduino® UNO R4 Minima board
- USB Type-C data cable
- 1 × 10 kΩ potentiometer
- Breadboard
- Jumper wires

3. Software Required

- Arduino IDE (v2.3.6 or later recommended)
- No additional libraries required.

Note:

We assume the **Arduino UNO R4 Minima board package** is already installed on your machine, as explained in the 00_Getting_Started/00_GettingStarted_Arduino_R4_Minima guide. If it is not installed, please refer to that document and complete the installation before proceeding.

4. Hardware Setup

This section explains how to connect the components for the **READ ANALOG VOLTAGE** project. It includes a **Wiring Diagram** and a **Circuit Schematic**.

4.1 Wiring Diagram

- Connect the **middle pin** of the potentiometer to Arduino **A0** (analog input).
- Connect one **Terminal 1(Outer) pin** of the potentiometer to **5V**.
- Connect one **Terminal 2(Outer) pin** of the potentiometer to **GND**.

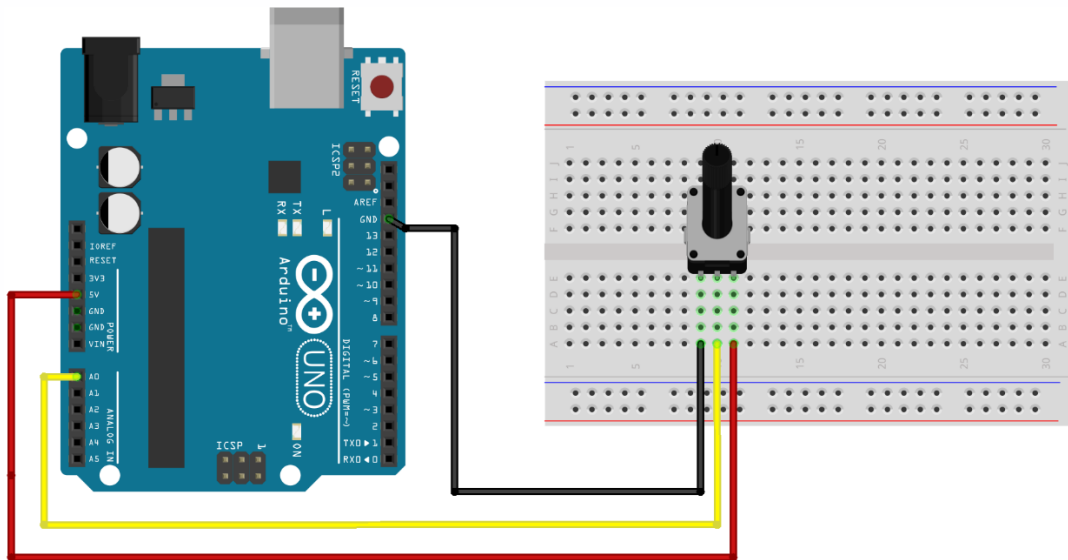



Figure 4.1 – Wiring diagram for Read Analog Voltage project

 **Tip:** Always disconnect the USB cable before making or changing hardware connections. This project is safe, but it's a good habit for advanced projects.

4.2 Circuit Schematic

- **POT1 = 10 kΩ potentiometer**
 - Wiper (middle pin) → A0
 - One side → 5V
 - Other side → GND

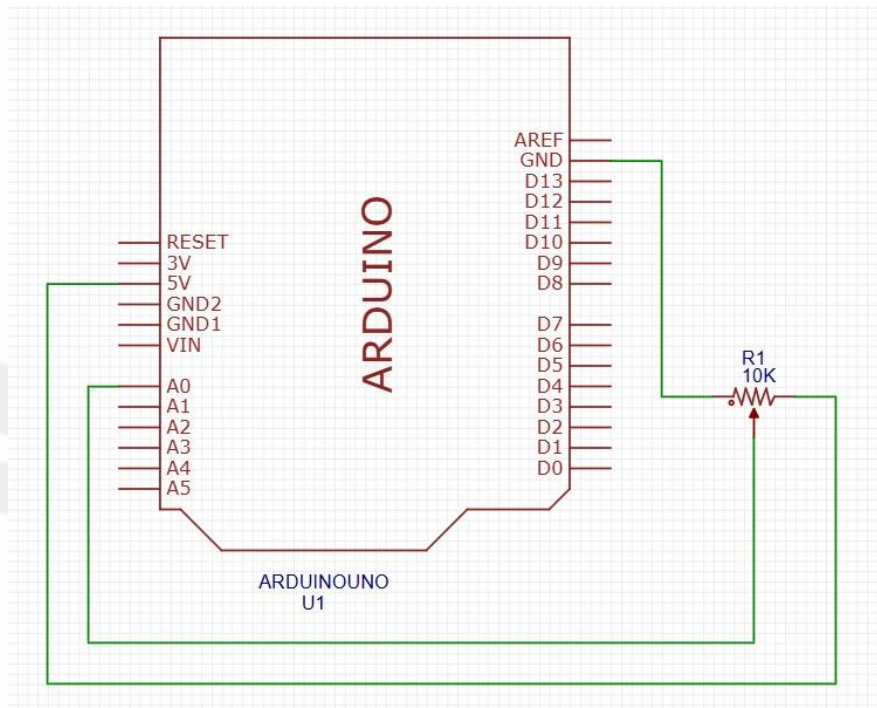


Figure 4.2 – Circuit schematic for Read Analog Voltage project

5. Principle – How It Works

A **potentiometer** is a variable resistor that divides voltage.

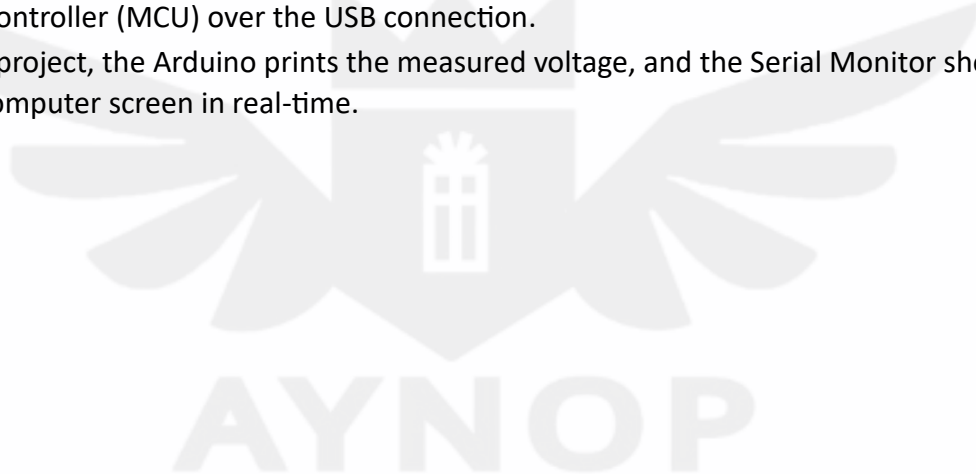
- The two outer pins connect across **5V and GND**.
- The middle pin (wiper) provides a voltage between **0V** and **5V**, depending on knob position.
- The Arduino UNO R4 Minima uses a **10-bit ADC**, so:
 - **analogRead(A0)** returns values between **0** and **1023**.
 - These values correspond to voltages between **0V** and **5V**.

Formula for conversion:

$$\text{Voltage} = (\text{analogRead}(A0) * 5.0) / 1023.0$$

To display these values, we use the **Serial Monitor**:

- The Serial Monitor is a **tool in the Arduino IDE** that lets you see text/data sent by the microcontroller (MCU) over the USB connection.
- In this project, the Arduino prints the measured voltage, and the Serial Monitor shows it on your computer screen in real-time.



6. Procedure – Steps to Run

1. Build the Circuit

- a. Assemble the potentiometer connections as shown in the **Wiring Diagram (Figure 4.1)**.

2. Connect the Board

- a. Use a USB Type-C data cable to connect your UNO R4 Minima to your computer.

3. Open the Project Code

- a. Simply **double-click** the file *03_Read_Analog_Voltage.ino* in the project folder, and it will open directly in the Arduino IDE (if installed).

4. Confirm Board Selection

- a. The IDE usually auto-detects the UNO R4 Minima if the package is installed.
- b. If not installed, refer to the *00_Getting_Started/00_GettingStarted_Arduino_R4_Minima* document to install the necessary board package.
- c. Verify that *Arduino UNO R4 Minima* is selected in the IDE's board selector (top toolbar).

5. Upload the Code

- a. Click the **Upload** button (arrow icon) in the top-left corner of the IDE.
- b. Wait until the console displays **"Done uploading."**

6. Observe the Behaviour

- a. Rotate the potentiometer knob and observe real time voltage value **on** Serial Monitor.

6.1 How to Use the Serial Monitor:

The Serial Monitor is a **tool in the Arduino IDE** that lets you see text/data sent by the microcontroller (MCU) over the USB connection.

- In Arduino IDE, go to **Tools → Serial Monitor**, or press **Ctrl + Shift + M**.
- A new window will open showing real-time data from your UNO R4 Minima.
- At the bottom-right of the Serial Monitor window, set the baud rate to **9600**. This must match the *Serial.begin(9600)* in the code.

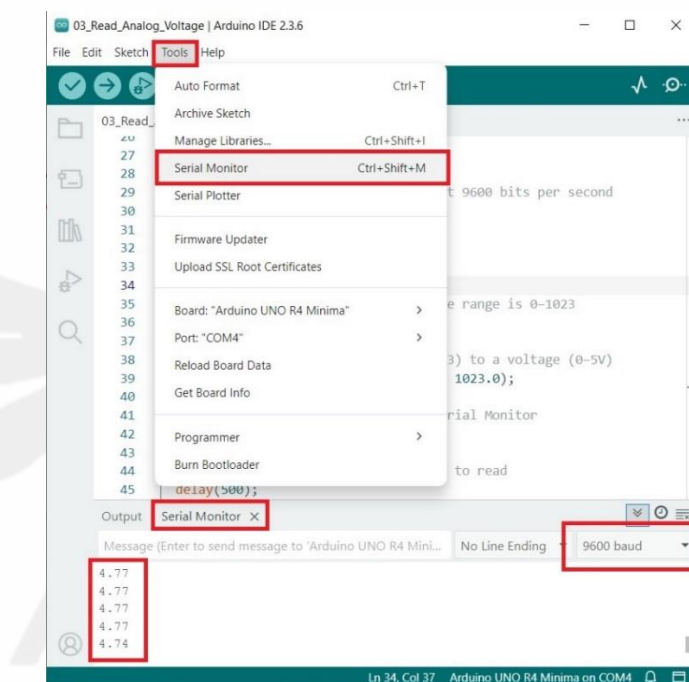


Figure 6.1 – Arduino IDE Serial Monitor set to 9600 baud

Tip: If the Serial Monitor shows nothing, check that:

- The correct **COM port** is selected.
- The baud rate is set to **9600**

7. Expected Output


- As you turn the potentiometer knob:
 - One extreme → near **0.00V**
 - Middle → around **2.50V**
 - Other extreme → near **5.00V**
- The Serial Monitor continuously prints the measured voltage values.

8. Code

The source code for this project is included in the downloaded folder:

 `uno-launchpad-kit/01_Basic_Projects/03_Read_Analog_Voltage/03_Read_Analog_Voltage.ino`

Tip:

- To open the project, simply **double-click the .ino file**. If the Arduino IDE is installed, it will launch automatically and load the code.
- If you **haven't installed the Arduino IDE yet**, please refer to:
 `uno-launchpad-kit/00_Getting_Started/00_GettingStarted_Arduino_R4_Minima` to **download and install it**.


8.1 Function References

- `setup()` – runs once when the board is powered on or reset.
- `loop()` – runs continuously after `setup()` finishes.
- `analogRead(pin)` – reads the value from the specified analog pin (0–1023 on UNO R4).
- `Serial.begin(baudrate)` – sets the data rate for serial communication.
- `Serial.println(data)` – prints data followed by a new line to the Serial Monitor.
- `delay(ms)` – pauses the program for the specified number of milliseconds


Arduino Reference:


- [`setup\(\)`](#)
- [`loop\(\)`](#)
- [`analogRead\(\)`](#)
- [`Serial.begin\(\)`](#)
- [`Serial.println\(\)`](#)
- [`delay\(\)`](#)

For more details and advanced usage, visit:

 [Arduino Language Reference](#) — The official guide for all Arduino functions.

9. Troubleshooting Tips

- **No output on Serial Monitor?**
 - Open Serial Monitor (Ctrl + Shift + M).
 - Ensure baud rate is set to **9600**.
- **Voltage not changing when knob is rotated?**
 - Verify potentiometer connections:
 - Middle pin → A0
 - One side → 5V
 - Other side → GND
- **Upload error in Arduino IDE?**
 - Verify that the correct board (**Arduino UNO R4 Minima**) is selected in the IDE.
 - Check that the correct **COM port** is chosen.
- **Board not detected?**
 - Ensure you are using a **data-capable USB Type-C cable** (some cables only provide charging).
 - Try reconnecting the cable or using a different USB port.
- **Board not listed in Arduino IDE?**
 - If you don't see **Arduino UNO R4 Minima** in the board selector, the **board package is not installed**.
 - To fix this, follow the installation steps in:
 [uno-launchpad-kit/00_Getting_Started/00_GettingStarted_Arduino_R4_Minima](#)

 **Tip:** If nothing works, press the **RESET** button on the UNO R4 Minima and try uploading the code again.

10. License

This content (source code and documentation) is licensed under **Creative Commons BY-NC-SA 4.0**.
© 2025 AYNOP. You may use, modify, and share for personal and educational purposes only.
Commercial use or redistribution without prior written permission is strictly prohibited.
Refer to the **LICENSE** file for complete details.

11. Support & Feedback

We value your feedback and are happy to assist with any questions, troubleshooting, or suggestions you may have.

✉ Email: support@aynop.com

When sending an email, please include your kit name (AYNOP® UNO Launchpad Kit) and, if applicable, the project name in the subject line. This will help our team respond faster and more accurately. We aim to respond to all queries within 2–3 business days. Your feedback helps us improve our products and create even better learning experiences for all Arduino beginners.

