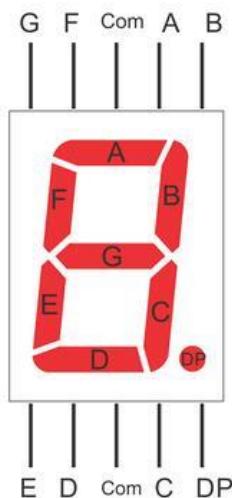


LAUNCH THE PROJECT - 07

1-Digit 7-Segment Display

(Common Cathode)

On AYNOP® UNO Launchpad Kit



Author: AYNOP Enterprises | Doc Version: 1.0 | Date: 13-9-2025

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
7. Expected Output	7
8. Code	8
8.1 Function References	8
9. Troubleshooting Tips	9
10. License	10
11. Support & Feedback	10

1. Overview

This project demonstrates how to interface a **single-digit 7-segment display** with the Arduino® UNO R4 Minima.

You will learn to:

- Connect and drive a 7-segment display (common cathode).
- Understand how numbers 0–9 are represented by segment patterns.
- Use an array to store digit patterns and display them sequentially.

2. Components Required

- Arduino® UNO R4 Minima board
- USB Type-C data cable
- 1 × Single-digit 7-segment display (common cathode)
- 1 × Current-limiting resistor (220 Ω) on the common cathode pin
- 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 **single-digit 7-segment display – Common Cathode** project. It includes a **Wiring Diagram** and a **Circuit Schematic**.

4.1 Wiring Diagram

- Connect the 7-segment display segments as follows:
 - Segment C → D2
 - Segment D → D3
 - Segment E → D4
 - Segment G → D5
 - Segment F → D6
 - Segment A → D7
 - Segment B → D8
- Connect the **common cathode pin** of the display to **GND** through a **220 Ω resistor**.

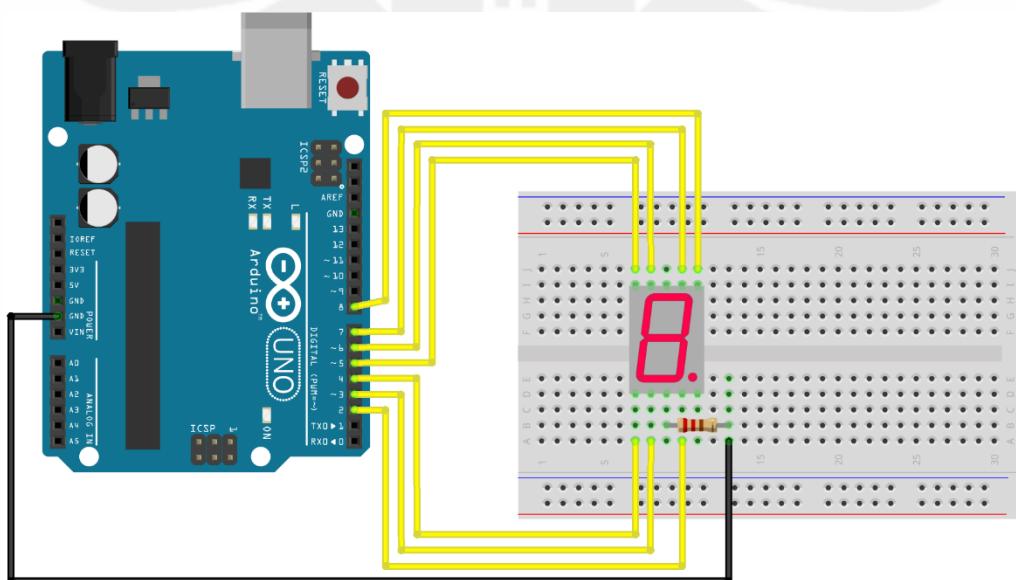


Figure 4.1 – Wiring diagram for Single-Digit 7-Segment Display project

Tip: Always use a resistor on the common cathode to limit current through the segments.

4.2 Circuit Schematic

- **7-Segment Display (Common Cathode):**
 - GND → GND
 - A → D7
 - B → D8
 - C → D2
 - D → D3
 - E → D4
 - F → D6
 - G → D5
- **Common Cathode → GND via $220\ \Omega$ resistor**

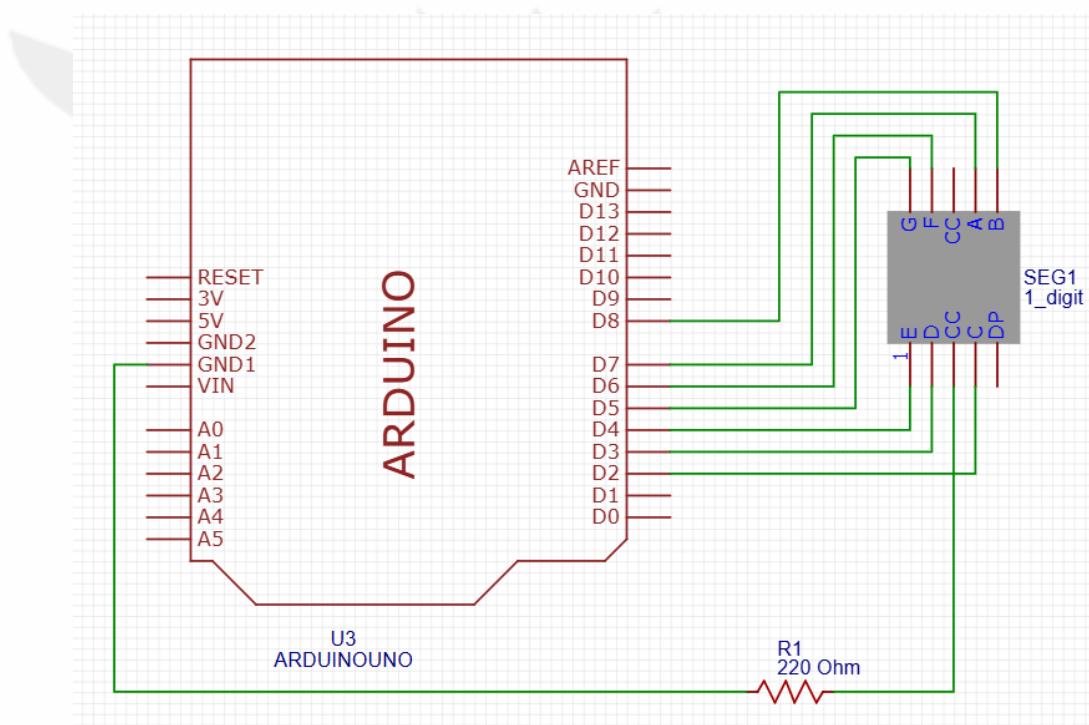


Figure 4.2 – Circuit schematic for Single-Digit 7-Segment Display project

5. Principle – How It Works

A **7-segment display** consists of 7 LEDs (segments) labelled **A–G**, arranged to form digits 0–9.

- **Common Cathode (CC):** All cathodes (–) are tied together and connected to GND. Segments light up when the corresponding pin receives a HIGH signal.
- **Common Anode (CA):** All anodes (+) are tied together and connected to VCC. Segments light up when the corresponding pin receives a LOW signal.

 In this project, we use a **Common Cathode (CC)** display. The Arduino sends HIGH signals to light up segments according to digit patterns.

The code stores digit patterns (0–9) in a **2D array**, where each row represents a digit and each column represents whether a segment is ON (1) or OFF (0)

Example:

- Digit **0** = Segments A, B, C, D, E, F ON → G OFF
- Digit **1** = Segments B, C ON → others OFF

The Arduino cycles through numbers **0–9**, displaying each for 1 second.

6. Procedure – Steps to Run

1. Build the Circuit

- a. Connect the 7-segment display pins to Arduino as shown in the **Wiring Diagram (Figure 4.1)**.
- b. Connect the common cathode to GND through a $220\ \Omega$ resistor.

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 *07_Single_Digit_7Segment_Display.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. The 7-segment display will show digits **0 → 9** one by one.
- b. Each digit remains ON for 1 second.
- c. After 9, the sequence restarts at 0.

7. Expected Output

- The 7-segment display sequentially shows numbers **0 to 9**, each for 1 second.
- The cycle repeats indefinitely.

8. Code

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

📁 *uno-launchpad-kit/01_Basic_Projects/07_Single_Digit_7Segment_Display/07_Single_Digit_7Segment_Display.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 at startup.
- **loop()** – runs continuously after **setup()**.
- **pinMode(pin, mode)** – configures a pin as INPUT or OUTPUT.
- **digitalWrite(pin, value)** – writes HIGH/LOW to a pin.
- **delay(ms)** – pauses execution for a specified time.

📚 **For more details and advanced usage, visit:**

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

9. Troubleshooting Tips

- **No digit displayed?**
 - Ensure common cathode pin is connected to GND through a resistor.
 - Check that Arduino pins are correctly connected to the right segments.
- **Digits appear incorrect or scrambled?**
 - Verify segment-to-pin mapping (A–G) matches the code.
 - Recheck breadboard connections.
- **Too dim?**
 - Use a lower resistor value (e.g., 220 Ω). Avoid going below 150 Ω to protect LEDs.
- **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: 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.