LAUNCH THE PROJECT - 05

Controlled Active Buzzer

(using Button)

On AYNOP® UNO Launchpad Kit

Author: AYNOP Enterprises | Doc Version: 1.0 | Date: 11-9-2025

Table of Contents

1. Overview

This project demonstrates how to control an **active buzzer** with a **pushbutton** on the Arduino® UNO R4 Minima.

You will learn to:

- Connect and use an **active buzzer**.
- Control sound output with a button press**.**
- Understand the difference between **active** and **passive** buzzers.

2. Components Required

- Arduino® UNO R4 Minima board
- USB Type-C data cable
- 1 × Active Buzzer
- 1 × Push button (4-pin tactile switch)
- 1 × 10 kΩ resistor (pull-down resistor for button)
- 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 **Button Controlled Active Buzzer** project. It includes a **Wiring Diagram** and a **Circuit Schematic**.

4.1 Wiring Diagram

- Connect the **+ pin** of the active buzzer to Arduino **D8**.
- Connect the **– pin** of the buzzer to **GND**
- Connect the **push button** as follows:
  - Leg 1 → Arduino **D2**
  - Leg 2 → **10 kΩ pull-down resistor → GND**
  - Leg 3 → **5V**
  - Leg 4 → **Not connected**

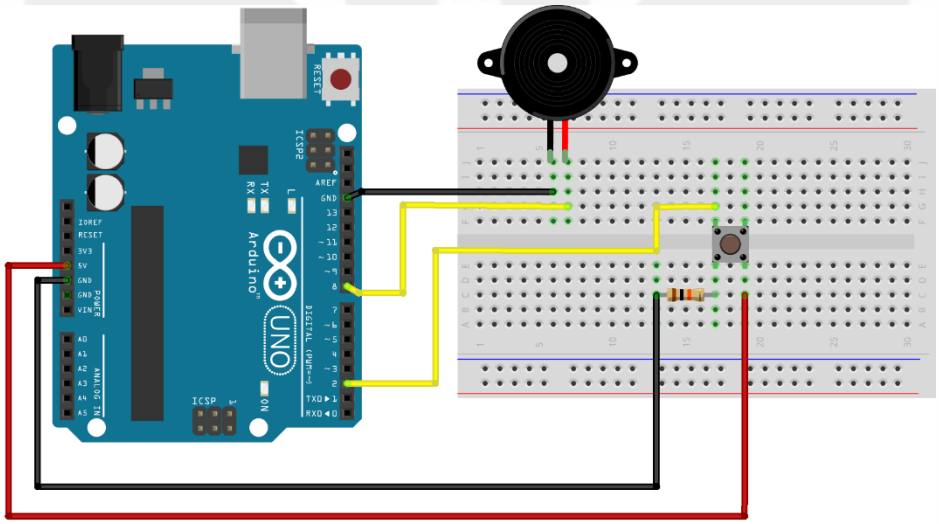

Figure 4.1 – Wiring diagram for Button Controlled Active Buzzer project

Tip: Active buzzers are **polarity-sensitive**. Ensure the **+ pin** goes to D8 and the **– pin** goes to GND.

4.2 Circuit Schematic

- **BZ1 = Active buzzer**
  - Pin 1 → Arduino D8
  - Pin 2 → GND
- **SW1 = Push button**
  - Pin 1 → Arduino D2
  - Pin 2 → 10 kΩ pull-down → GND
  - Pin 3 → 5V
  - Pin 4 → NC (not connected)

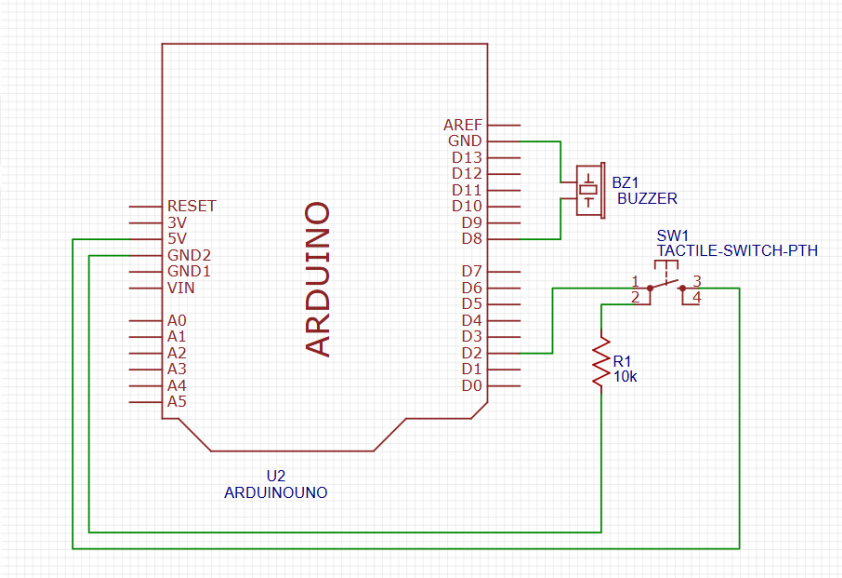

Figure 4.2 – Circuit schematic for Button Controlled Active Buzzer project

5. Principle – How It Works

An **active buzzer** has a built-in oscillator circuit. When supplied with power (HIGH signal), it automatically produces a fixed tone. Unlike a passive buzzer, which requires a frequency input using the tone() function, the active buzzer only needs an ON/OFF digital signal.

In this project:

- The **push button** provides a HIGH signal to pin D2 when pressed.
- Arduino reads this input and switches the buzzer ON or OFF accordingly.

👉 **Key Difference:**

- **Passive buzzer** – no internal circuit, not polarity sensitive, produces multiple tones with tone().
- **Active buzzer** – has internal circuit, polarity sensitive, produces a single fixed tone when powered.

6. Procedure – Steps to Run

1. **Build the Circuit**
   a. Assemble the button and buzzer 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 *05_Button_Controlled_Active_Buzzer.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. Press the button → the buzzer turns ON (produces sound).
   b. Release the button → the buzzer turns OFF (silent)**.**

7. Expected Output

- When the **button is pressed**, the buzzer emits a continuous tone.
- When the **button is released**, the buzzer becomes silent.
- The behaviour repeats reliably for each button press.

8. Code

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

📂 uno-launchpad-kit/01_Basic_Projects/05_Button_Controlled_Active_Buzzer/05_Button_Controlled_Active_Buzzer.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 res.et.
- loop() – runs continuously after setup() finishes.
- pinMode(pin, mode) – configures a pin as INPUT or OUTPUT.
- digitalRead(pin) – reads the state of a digital pin (HIGH/LOW).
- digitalWrite(pin, value) – writes a digital value (HIGH/LOW) to a pin.

📚 **For more details and advanced usage, visit:**
🔗 Arduino Language Reference — The official guide for all Arduino functions.

9. Troubleshooting Tips

- **Buzzer not making sound?**
  - Ensure the **+ pin** of the buzzer is connected to D8 and the **– pin** to GND.
  - Confirm that you are using an **active buzzer** (a passive buzzer will not work with this code).

- **Button not working?**
  - Verify button wiring:
    - Leg 1 → Arduino D2
    - Leg 2 → 10 kΩ pull-down → GND
    - Leg 3 → 5V
    - Leg 4 → not connected

- **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

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.