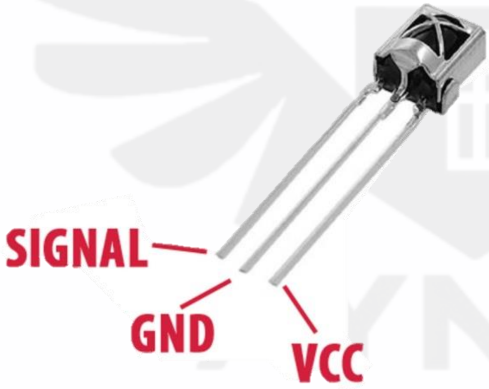
LAUNCH THE PROJECT - 14

IR Remote Key Reading

(via VS1838B)

On AYNOP® UNO Launchpad Kit



⚠ Important: **CR2032 3V IR Remote battery** is **not included** in product package

Author: AYNOP Enterprises | Doc Version: 1.0 | Date: 20-9-2025

Table of Contents

1. Overview

This project demonstrates how to read key presses from an **IR remote control** using the **VS1838B infrared receiver module** with the Arduino® UNO R4 Minima. This experiment introduces you to the basics of **infrared communication**, which is widely used in TV remotes, air conditioners, and other consumer electronics

You will learn to:

- Connect the VS1838B IR receiver to Arduino.
- Use the **IRremote library** to decode IR signals.
- View decoded key values in the Serial Monitor.
- Ignore repeat codes for cleaner output.

2. Components Required

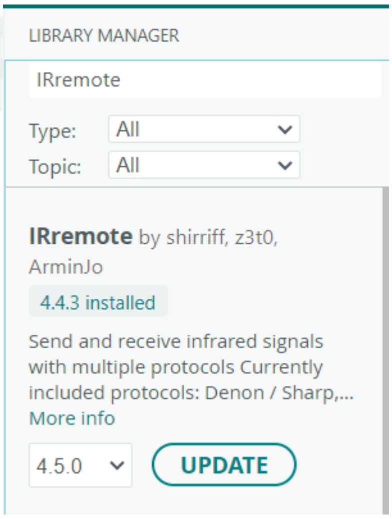
- Arduino® UNO R4 Minima board
- USB Type-C data cable
- VS1838B IR receiver module
- IR Remote
- IR Remote Battery - **CR2032 3V Lithium Coin Battery** (⚠ **not included** in this kit – must be purchased separately to power the remote)
- Breadboard
- Jumper wires

3. Software Required

- Arduino IDE (v2.3.6 or later recommended)
- Library required: **IRremote by shirriff.**

Installing the Library:

- Open Arduino IDE.
- Go to **Sketch → Include Library → Manage Libraries**
- Search for **IRremote (by shirriff).**
- Install the library.



LIBRARY MANAGER

IRremote

Type: All

Topic: All

IRremote by shirriff, z3t0, ArminJo

4.4.3 installed

Send and receive infrared signals with multiple protocols Currently included protocols: Denon / Sharp,...

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 VS1838B IR receiver.

4.1 Wiring Diagram

- VS1838B **VCC** → **5V**.
- VS1838B **GND** → **GND.**
- VS1838B **OUT** (signal pin) **→ Arduino D2**.

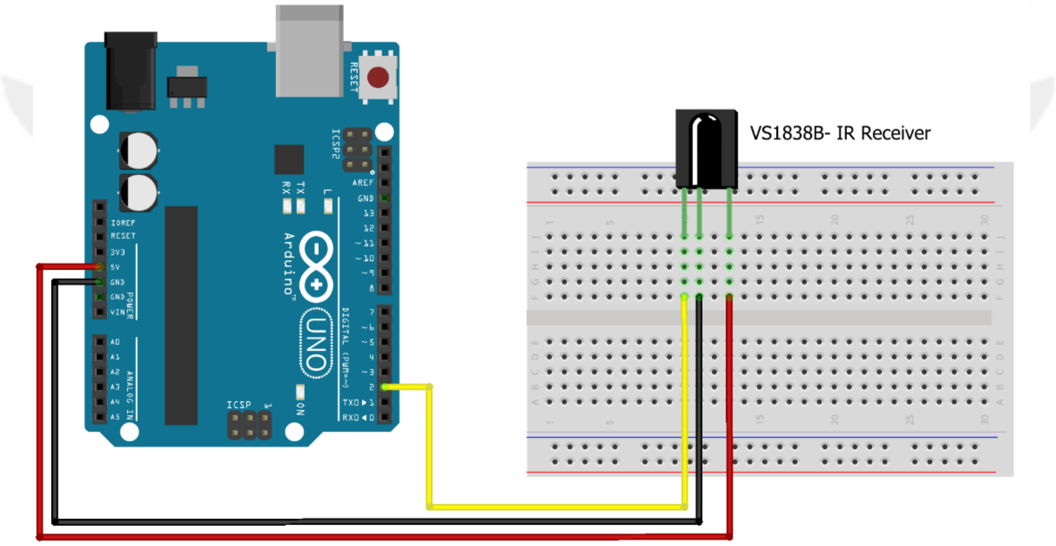


Figure 4.1 – Wiring diagram for VS1838B IR receiver module with Arduino UNO R4 Minima

💡 **Tip:** The IR receiver is sensitive to ambient light. For best results, avoid direct sunlight or bright lamps when testing

⚠ **Important: Insert a CR2032 3V battery into the IR remote before testing. The battery is not included in this product package.**

5. Principle – How It Works

An infrared (IR) remote works by sending invisible light signals whenever you press a button. Each button has a unique signal pattern, which allows the receiver to tell them apart.

- The **VS1838B IR receiver** captures these light signals and converts them into a digital format that the Arduino can read. With the help of the **IRremote library**, these signals are translated into hexadecimal codes.
- Each button on the remote produces a **unique code** (for example: 0xFF30CF).
- These codes are shown as hexadecimal values in the Serial Monitor.
- If you **hold a button down**, the remote sends a **repeat code (0x0)**. In this project, repeat codes are ignored to keep the output clear.

6. Procedure – Steps to Run

1. **Build the Circuit**
   a. Connect the VS1838B IR receiver as shown in the **Wiring Diagram (Figure 4.1)**.

2. **Insert the Battery**
   a. Place a **CR2032 3V coin battery** into the remote. Ensure correct polarity.

3. **Connect the Board**
   a. Use a USB Type-C data cable to connect your UNO R4 Minima to your computer.

4. **Open the Project Code**
   a. Simply **double-click** the file *14_IR_Remote_Read.ino* in the project folder, and it will open directly in the Arduino IDE (if installed).

5. **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).

6. **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**."

7. **Observe the Behaviour**
   a. Point the remote towards the VS1838B receiver.
   b. Press any button on the remote.
   c. The Arduino will decode the signal and display a **hexadecimal code** on serial monitor**.**
   d. Try pressing different buttons and observe that each one produces a unique code.
   e. Holding a button will not produce repeated codes, as these are filtered out in the code.

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.

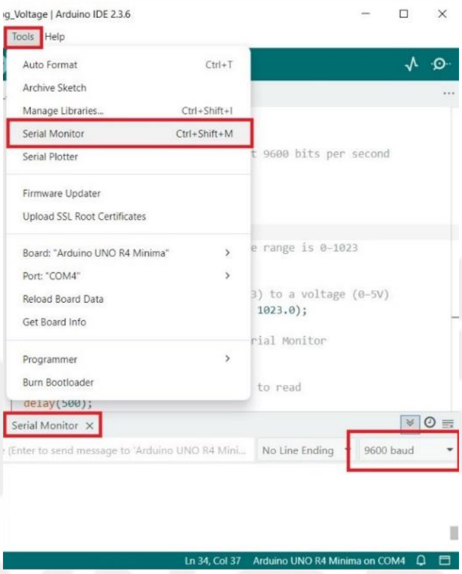


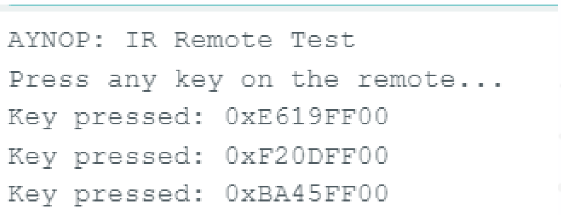
Figure 6.1 – Arduino IDE Serial Monitor set to 9600 baud

💡 **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

- The Serial Monitor will show a unique hexadecimal code for each button pressed, as shown below:



```
AYNOP: IR Remote Test
Press any key on the remote...
Key pressed: 0xE619FF00
Key pressed: 0xF20DFF00
Key pressed: 0xBA45FF00
```

8. Code

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

📁 *uno-launchpad-kit/01_Basic_Projects/ 14_IR_Remote_Read/14_IR_Remote_Read.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.
- IrReceiver.begin(pin, feedback) – initializes the IR receiver on a pin.
- IrReceiver.decode() – checks if a signal has been received.
- IrReceiver.decodedIRData.decodedRawData – provides the received IR code value.
- IrReceiver.resume() – prepares the receiver for the next signal.
- Serial.begin(baud) – starts serial communication.
- Serial.print() / Serial.println() – prints decoded values to Serial Monitor.

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

🔗 IRremote library — The GitHub repo of IRremote library.

🔗 Arduino Language Reference — The official guide for all Arduino functions.

9. Troubleshooting Tips

- **Remote not working?**
  - Check if a **CR2032 3V battery** is installed in the remote.
  - Ensure correct polarity of the battery.

- **No output in Serial Monitor?**
  - Check that the baud rate is set to 9600.
  - Ensure the VS1838B OUT pin is connected to Arduino D2.
  - Confirm that the correct IRremote library is installed.

- **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 vis USB?**
  - 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@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.