

## LAUNCH THE PROJECT - 06

# LCD Display – 16x2 *(using I2C Interface)*

**On AYNOP® UNO Launchpad Kit**



*Author: AYNOP Enterprises | Doc Version: 1.0 | Date: 12-9-2025*

## Table of Contents

<b>1. Overview</b>	3
<b>2. Components Required</b>	3
<b>3. Software Required</b>	3
<b>4. Hardware Setup</b>	4
<b>4.1 Wiring Diagram</b>	4
<b>4.2 Circuit Schematic</b>	5
<b>5. Principle – How It Works</b>	6
<b>6. Procedure – Steps to Run</b>	7
<b>7. Expected Output</b>	7
<b>8. Code</b>	8
<b>8.1 Function References</b>	8
<b>9. Troubleshooting Tips</b>	9
<b>10. License</b>	10
<b>11. Support &amp; Feedback</b>	10

## 1. Overview

This project demonstrates how to use a **16x2 LCD display with an I2C interface**.

You will learn to:

- Connect a 16x2 LCD using only **two wires (SDA, SCL)**.
- Use the **LiquidCrystal\_I2C library** to control the display.
- Display text with a **typewriter effect** (characters appear one by one).

## 2. Components Required

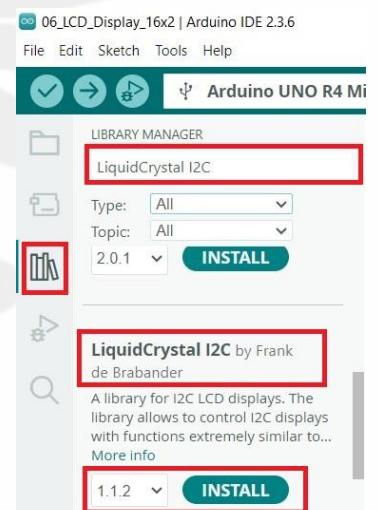
- Arduino® UNO R4 Minima board
- USB Type-C data cable
- 1 × 16x2 LCD with I2C adapter (backpack)
- Jumper wires

## 3. Software Required

- Arduino IDE (v2.3.6 or later recommended)
- Required library: **LiquidCrystal\_I2C**.

### Installing the Library:

- Open Arduino IDE.
- Go to **Sketch → Include Library → Manage Libraries**
- Search for **LiquidCrystal I2C** (by Frank de Brabander)
- Install the library.



### 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 **16x2 LCD Display - I2C Interface** project. It includes a **Wiring Diagram** and a **Circuit Schematic**.

### 4.1 Wiring Diagram

- Connect LCD **VCC** → **5V**.
- Connect LCD **GND** → **GND**.
- Connect LCD **SDA** → **SDA** (pin **D18** on **UNO R4 Minima**)
- Connect LCD **SCL** → **SCL** (pin **D19** on **UNO R4 Minima**)

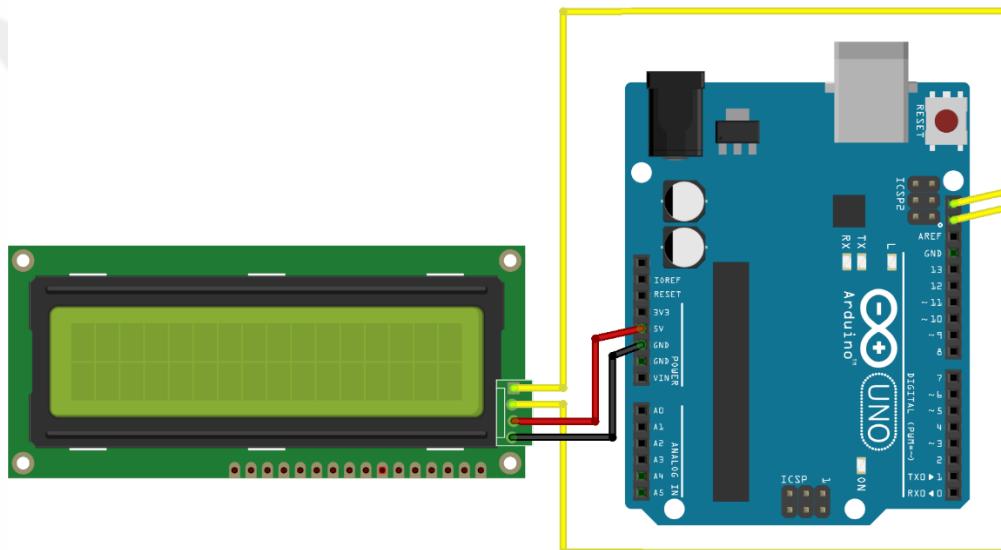


Figure 4.1 – Wiring diagram for 16x2 I2C LCD display

**Tip:** I2C uses the same two wires (SDA, SCL) for multiple devices. Each device has a unique **I2C address**. Most LCDs use **0x27** or **0x3F**.

## 4.2 Circuit Schematic

- **U1 = Arduino UNO R4 Minima**
- **LCD1 = 16x2 LCD with I2C adapter**
  - GND → GND
  - VCC → 5V
  - SDA → D18 (SDA)
  - SCL → D19 (SCL)

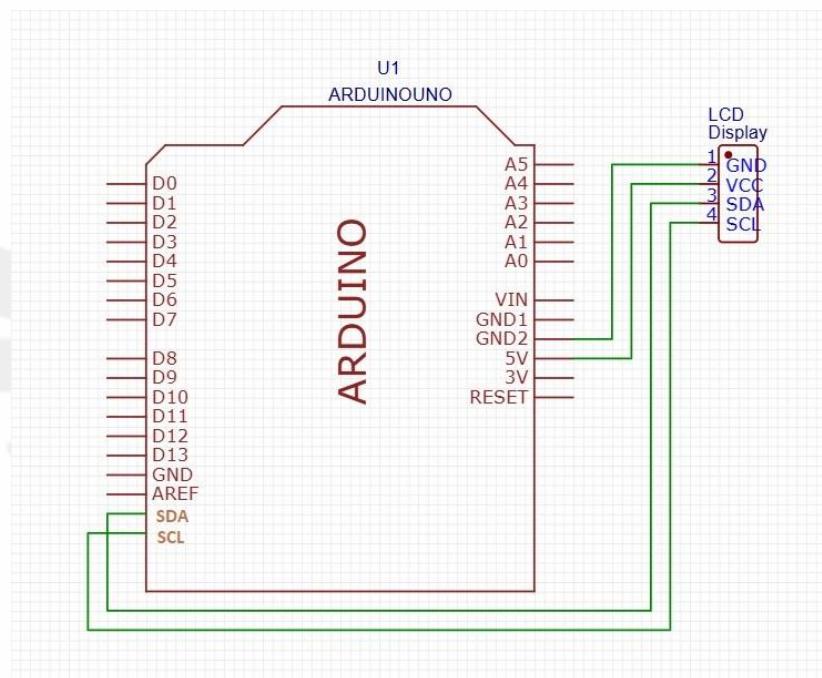


Figure 4.2 – Circuit schematic for LCD I2C project

## 5. Principle – How It Works

A **16x2 LCD** can display up to 32 characters (16 per row).

- The **I2C adapter** reduces the wiring from 16 pins to just 4 (VCC, GND, SDA, SCL).
- Communication happens over the **I2C protocol**, which allows multiple devices to share the same bus.
- The **LiquidCrystal\_I2C library** provides functions to initialize the LCD, set cursor position, and print text.
- In this project, text is displayed character by character to create a **typewriter effect**.

 **Note:** If the LCD shows nothing, check the I2C address (default is 0x27 but some modules use 0x3F)



## 6. Procedure – Steps to Run

### 1. Build the Circuit

- Connect the LCD as shown in the **Wiring Diagram (Figure 4.1)**.

### 2. Connect the Board

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

### 3. Open the Project Code

- Simply **double-click** the file *06\_LCD\_I2C\_Display\_16x2.ino* in the project folder, and it will open directly in the Arduino IDE (if installed).

### 4. Confirm Board Selection

- The IDE usually auto-detects the UNO R4 Minima if the package is installed.
- If not installed, refer to the *00\_Getting Started/00\_GettingStarted\_Arduino\_R4\_Minima* document to install the necessary board package.
- Verify that *Arduino UNO R4 Minima* is selected in the IDE's board selector (top toolbar).

### 5. Upload the Code

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

### 6. Observe the Behaviour

- The LCD backlight turns ON.
- The first line (“Hello Genius”) appears one character at a time.
- The second line (“AYNOP Enterprises”) appears the same way.
- After a short pause, the process repeats.

## 7. Expected Output

- Line 1: *Hello Genius* → appears letter by letter.
- Line 2: *AYNOP Enterprises* → appears letter by letter.
- Both lines remain for 2 seconds, then clear and repeat.

## 8. Code

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

📁 uno-launchpad-kit/01\_Basic\_Projects/06\_LCD\_I2C\_Display\_16x2/06\_LCD\_I2C\_Display\_16x2.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
- **Lcd.init()** – initializes the LCD.
- **Lcd.backlight()** – turns ON the backlight.
- **Lcd.clear()** – clears the display.
- **Lcd.setCursor(col, row)** – sets the cursor position (col = 0–15, row = 0–1).
- **Lcd.print(data)** – prints text or numbers at the cursor position.
- **delay(ms)** – pauses the program for the specified number of milliseconds.

📚 **Library Reference:**

⌚ [Wire Library](#) — used internally by LiquidCrystal\_I2C.

⌚ [LiquidCrystal\\_I2C.h](#) — LiquidCrystal\_I2C functions and their arguments, see the header file.

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

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

## 9. Troubleshooting Tips

- **Error while compiling (LiquidCrystal\_I2C not found)?**
  - This happens if the required library is missing or if the wrong one is installed.
  - Open Arduino IDE → **Library Manager**, search for **LiquidCrystal I2C (by Frank de Brabander)**, and install it
- **LCD backlight ON but no text?**
  - Adjust the small **blue potentiometer** on the I2C adapter to set contrast.
- **LCD backlight not ON?**
  - I2C adapter boards have a small **jumper near the backlight pins**.
  - Ensure this jumper is connected; otherwise, the backlight will stay OFF
- **LCD shows nothing?**
  - Check wiring: VCC → 5V, GND → GND, SDA → D18, SCL → D19.
  - Verify the **I2C address** (0x27 or 0x3F)
- **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](mailto: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.