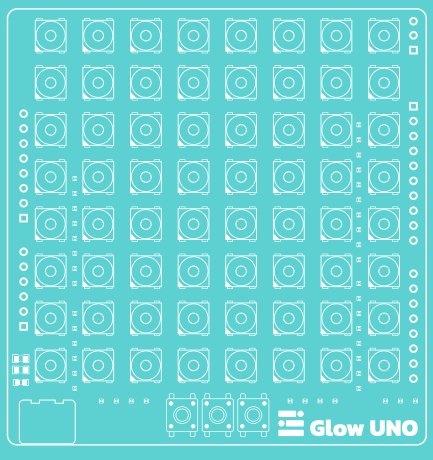



# Glow UNO

# Getting Started Guide Book

Whether a shield for your UNO or wired to your favorite controller, illuminate your projects with this feature-packed addressable RGB matrix.

adafruit

# Contents

# Introduction


## Glow UNO Getting Started Guide

Welcome to the amomii Glow UNO Getting Started Guide! The Glow UNO is a versatile 8x8 addressable WS2812B RGB matrix designed to function as both an Arduino UNO shield and a standalone RGB display. It features 64 individually controllable LEDs, integrated push buttons for user interaction, a USB C power port with Power Delivery support, and an XH connector for additional peripherals. Whether you're a maker, hobbyist, or educator, the Glow UNO offers endless possibilities for creating interactive displays, games, visual effects, and more.

# Overview

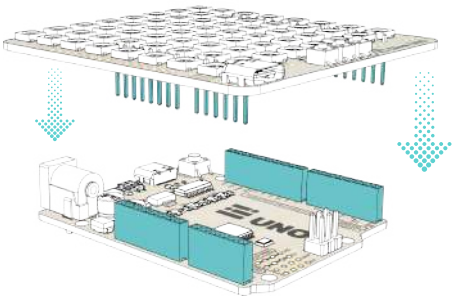
This guide will walk you through setting up your amomii Glow UNO as a shield for the amomii UNO (or any other Arduino UNO type boards). For using the Glow UNO as an RGB matrix panel and for detailed connection methods with other controllers such as ESP boards, Raspberry Pi, and additional Arduino boards, please refer to the **Glow UNO Datasheet**.

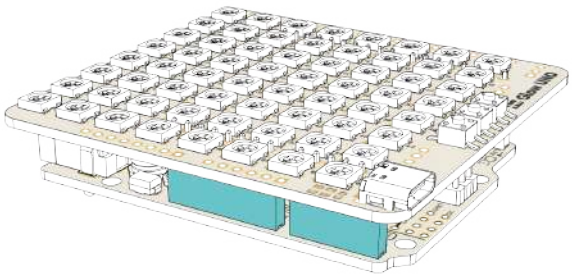
In this guide, we will cover how to connect the Glow UNO as an Arduino UNO shield with step-by-step instructions. You'll also find guidance on downloading and installing the FastLED coding library to program the Glow UNO, along with instructions for uploading a test code to ensure proper functionality. Additionally, we'll demonstrate how to upload an official amomii example project to explore the full capabilities of the Glow UNO.

# Get Connected

To connect the Glow UNO to an Arduino UNO type board:

1. Align the Glow UNO shield's pin headers with the corresponding headers on the UNO.
2. Press the shields together firmly to ensure a secure connection.





Once connected, the Glow UNO can be powered through the UNO or via the onboard USB C power port. For optimal performance, especially when driving a large number of LEDs at high brightness levels, it is recommended to use an external 5V power supply through the PD compatible USB C port on the shield. Using this port will also power the UNO board.

## Downloading the FastLED Library and Uploading a Test Code

### Get the Library

You can write code for your amomii Glow UNO using various IDEs, but we recommend the Arduino IDE with the FastLED library installed on it. We will go over how to install the FastLED library next, but if you don't have the Arduino IDE, you can download it for free from the official Arduino website – **arduino.cc**

You can also find more details about using the IDE in the Getting Started manual for the amomii UNO. This can be found at **amomii.com/pages/downloads**

### Download the FastLED library

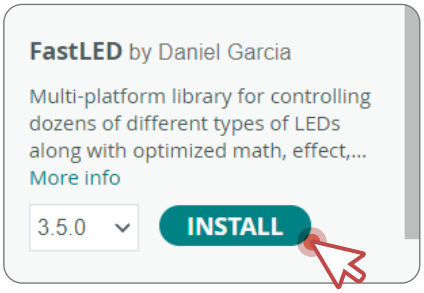
There are various ways to download coding libraries for the Arduino IDE, but the simplest way is to download them directly from within the IDE itself. We will be using Version 2 of the IDE, but the steps for the original are similar.

Click on the **library manager** shortcut on the left hand side of the IDE.

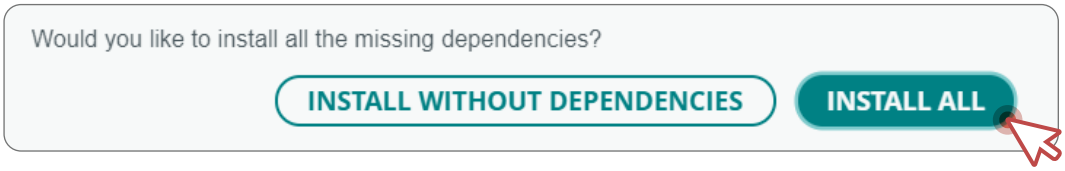
**Note**

*There is no shortcut here on the original Arduino IDE, but the Library Manager can be accessed from* **Tools > Manage Libraries...**


When the Library Manager pops up, search for FastLED and click **INSTALL** on the version written by Daniel Garcia.

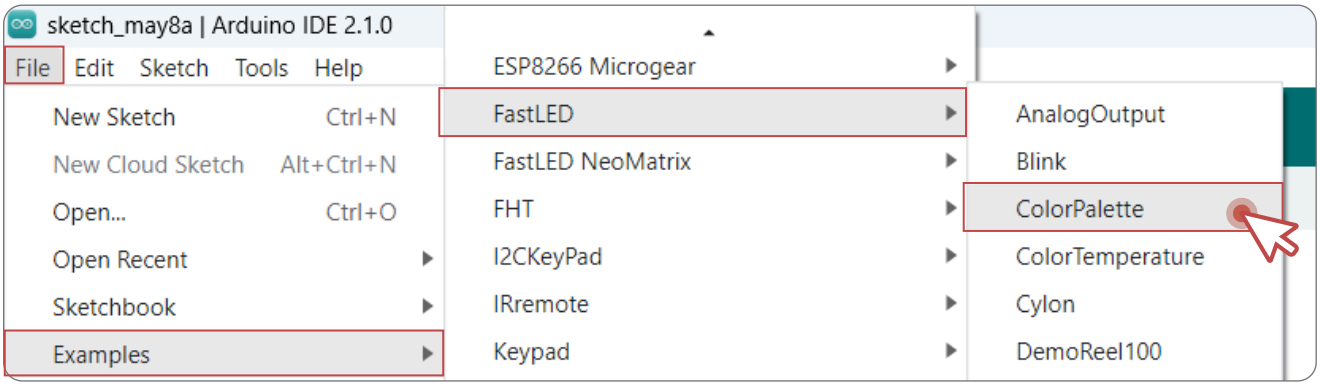



If you are prompted with the question whether you would like to download all the missing dependencies, click **"INSTALL ALL"**.



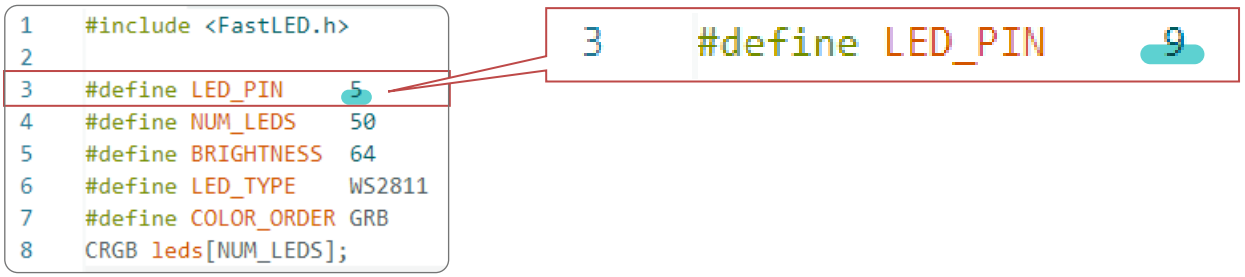
## Editing the Example Code

With your amomii Glow UNO connected to your microcontroller and your microcontroller connected to your computer, open the IDE and navigate to **File› Examples› FastLED› ColorPalette**




Before you upload the code to your board, there are a couple of changes to the sketch you need to make.

On line 3, change the LED_PIN definition to '9'. This is the pin used on the Glow UNO shield to receive digital information from the microcontroller for the LEDs.




Next, we must change the NUM_LEDS value to correspond with the amount of pixels you want to control. Given the Glow UNO panel has a total of 64, we need to adjust this value accordingly.

After that, you have the option of whether you want to change the brightness of the strips. In general, it is a good idea to start low and work your way up to find the right value for brightness. The value you put must be between 0 (completely dark) and 255 (max brightness).

*Before choosing your value, you must consider the fact that the brighter the pixels shine, the more power they demand. You must make sure that you have a power supply capable of delivering enough power to your project. For more details on power consumption, please read the datasheet carefully.*

I will change the value to **50**.

```
5    #define BRIGHTNESS  50
```

Finally, you must change the chipset to match the chips used on the amomii Glow UNO module, **WS2812B**.

```
6    #define LED_TYPE   WS2812B
```

Lines 3 - 6 of your updated code will now look something like this:

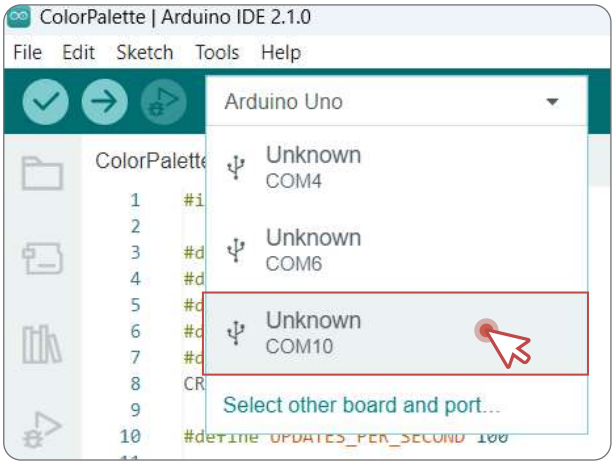
```
3  #define LED_PIN     9
4  #define NUM_LEDS    64
5  #define BRIGHTNESS  50
6  #define LED_TYPE    WS2812B
```

## Uploading the Code

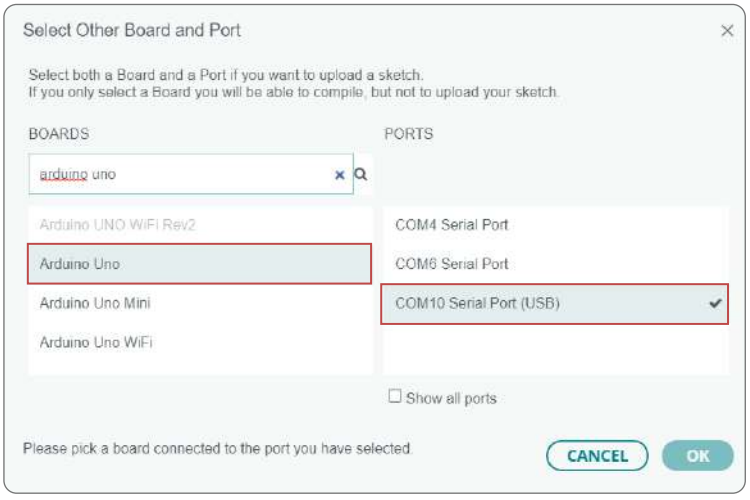
Before hitting the Upload button, you need to make sure your board and comport are connected.

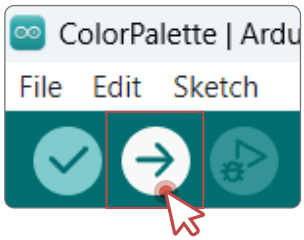
If you are using the amomii UNO, you can find information about this on the Gettin-Started document. You can find this at **amomii.com/products/UNO**.

But in short, to make sure your board is connected, click on the boards dropdown and select the comport.

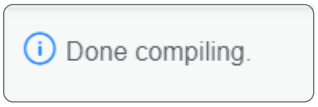



Then select your board. If you are using the amomii UNO, select **"Arduino UNO"** from the boards section as they have the same configuration.




Finally, click the **Upload** shortcut, and after you see the **"Done"** note at the bottom of the IDE, your Glow UNO should start displaying beautiful lighting effects.



# Example Projects

## Uploading Example Code

If the FastLED test code ran successfully, it's time to upload an example code to explore the capabilities of your Glow UNO. We recommend starting with the 'Connect Four' example, but feel free to try out all the examples provided and explore the comments to understand their functions.

## Accessing Example Code

You can download the official amomii Glow UNO example codes from multiple sources:

### ● Welcome Package

You should have received the example code along with documentation, and various other digital assets in your welcome email. If you haven't received your welcome email, scan this QR code or enter the URL into your browser and tell us where to send it.

- Example Code
- Datasheet
- Getting Started Manual
- Schematic
- 3D Printable STLs

**amomii.com/pages/gmx_me3e**

### ● Downloads Section

Visit **amomii.com** and navigate to the Downloads section. If you haven't created an account yet, you'll need to do so to access this area. Once logged in, locate the Glow UNO section and click on the "Code" hyperlink. Download the Glow UNO example code, which will be packaged in a ZIP folder.

## Uploading the Example Code

1. **Unzip the Downloaded Folder:** After downloading, unzip the folder named 'amomii_GLU_code'. Inside, you'll find the 'Glow_UNO_Examples' folder containing various example projects.
2. **Open the Connect Four Example:** Navigate to the 'Glow_UNO_Example' folder, locate the 'Connect_Four' folder, and open the .ino file. This action should automatically launch the Arduino IDE. If not, open the Arduino IDE manually and locate the .ino file by navigating to File > Open within the IDE interface.
3. **Upload to Your Glow UNO:** Connect your Glow UNO to your computer using a USB cable. Ensure you select the correct board and port in the Arduino IDE under Tools > Board and Tools > Port.
4. **Verify and Upload:** Click the Verify button (checkmark icon) to compile the code and ensure there are no errors. Once verified, click the Upload button (arrow icon) to upload the code to your Glow UNO.

**Note:** After uploading the code, before playing the game, it is advised to power the game via the Glow UNO shield's USB C port using a USB C cable and USB power supply capable of delivering enough power for the project. Most phone charger plugs should work fine. You can find more information about power in the **data sheet.**

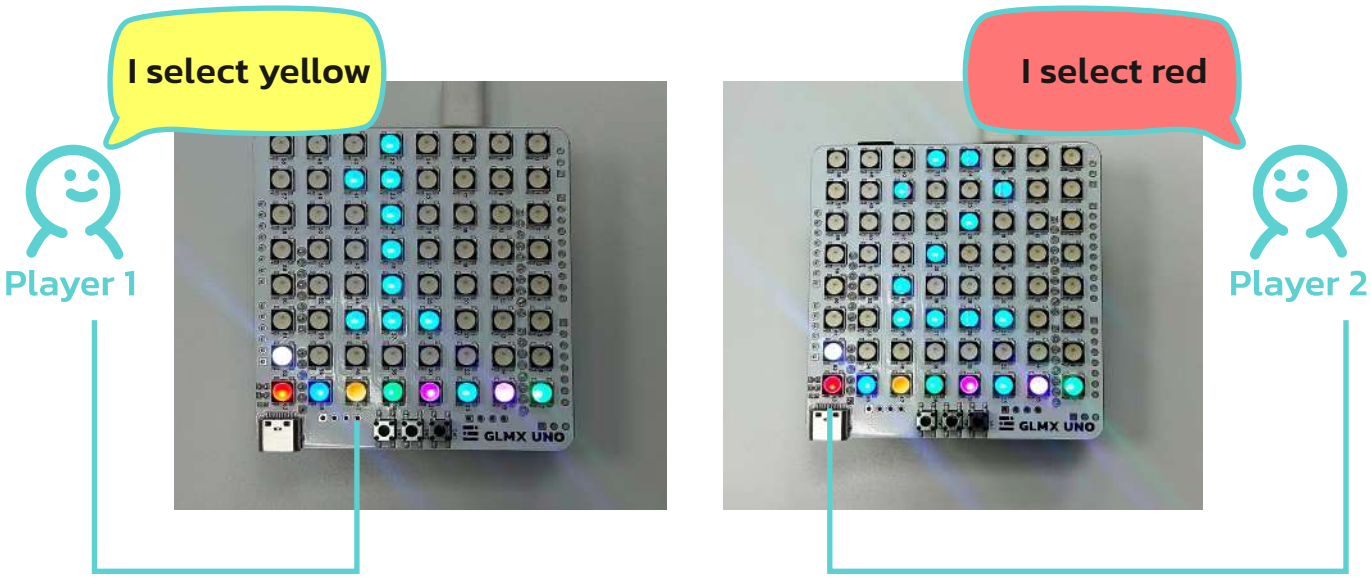
# Play the Game

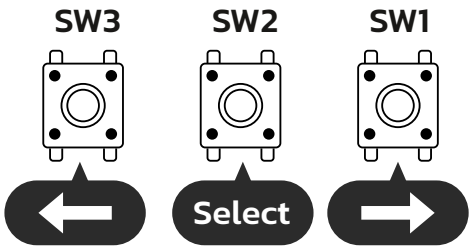
## Objective

Connect Four is a classic two-player game where the goal is to be the first to connect four of your colors in a row, either vertically, horizontally, or diagonally.

## Color Selection

During Player 1's turn, a large '1' will be displayed, and the bottom row of the LED matrix will showcase all available colors. Above these colors, a white dot can be moved left and right using SW1 and SW3. Pressing SW2 selects the color currently positioned under the white dot for Player 1. After Player 1 makes their selection, the process repeats for Player 2, where a '2' is displayed, and the color selection mechanism remains the same.

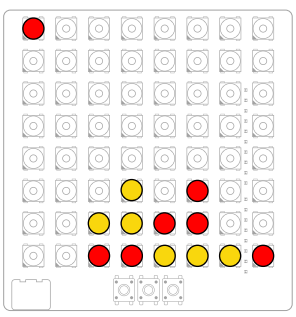


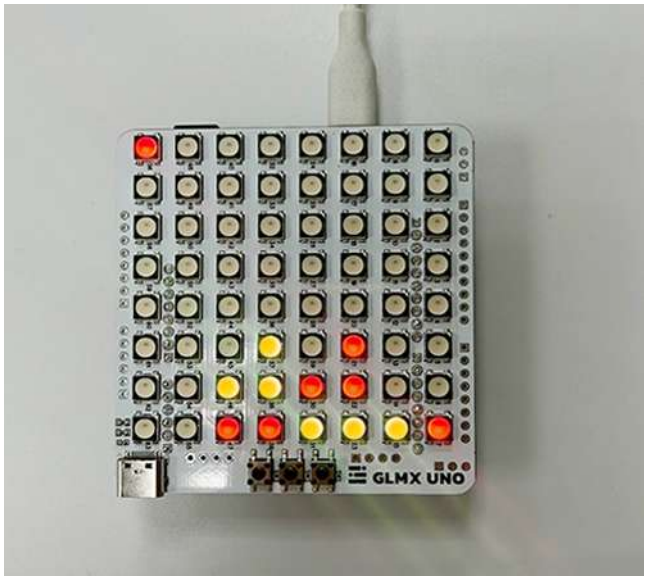




---

## Turns and Moves

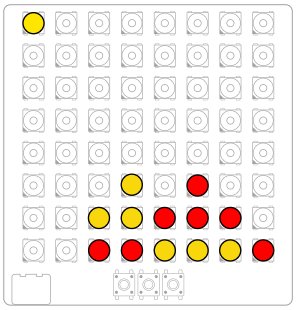
The game starts with Player 1 choosing a column to drop their colored chip using **SW1, SW3, to move and SW2 to select**. Pressing SW2 drops a chip into the lowest available slot in the selected column.

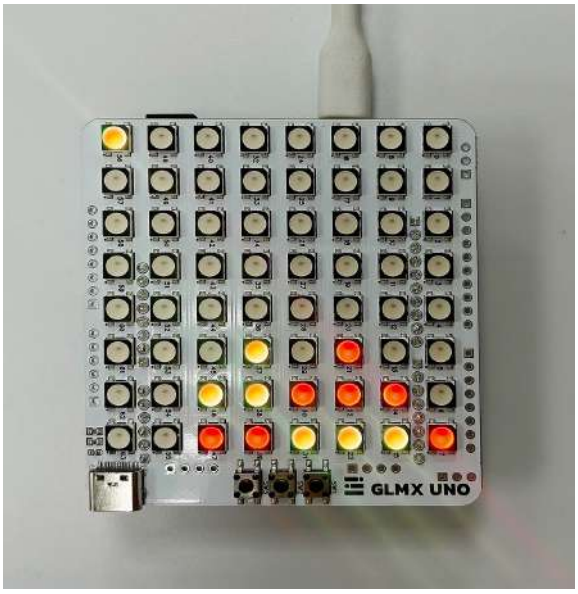




## Next Move

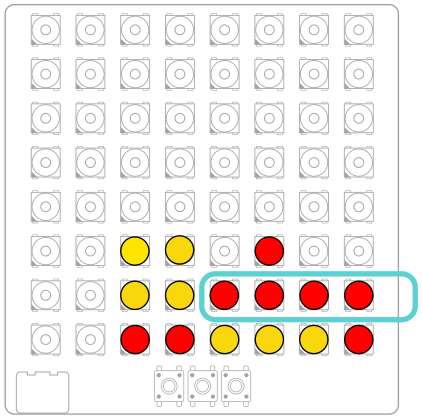
After each player's move, the turn automatically switches to the next player. Players continue taking turns until one player wins or the board fills up without a winner.



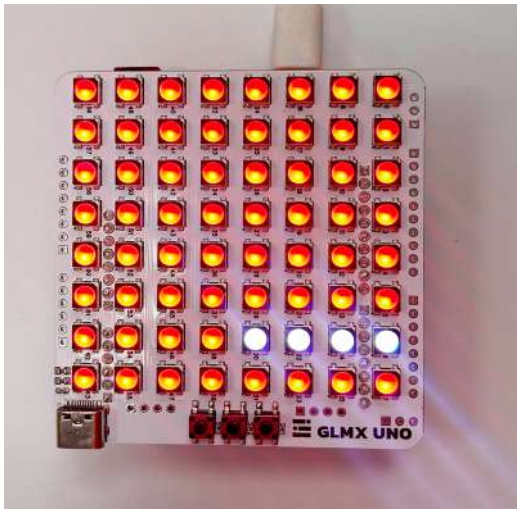



## Win Condition

After each move, the game checks if any player has connected four chips of their color either vertically, horizontally, or diagonally.


If a player achieves this, the LEDs will animate to indicate the winning combination, and the game will pause briefly to celebrate the win.



## Additional Information

**Sound Effects:**
The game features sound effects for various actions, such as winning a round or making a move. To enable these sound effects, connect a speaker or a passive buzzer between pins 11 and GND.

**Screen Rotation:**
By default, the screen orientation is set up for gameplay with the buttons positioned under the display. You can adjust this orientation by modifying the ROTATION definition (0 - 3). The default setting is '1'.

```
#define ROTATION 1
```

**Brightness:**
You can adjust the brightness of the game by modifying the value of the BRIGHTNESS definition, which ranges from 0 to 255. It's crucial not to set the value too high to prevent excessive power consumption and avoid overly bright conditions, which may strain your eyes. If you are not using a diffuser to cover the pixels, a lower value around 20 provides sufficient brightness for safe enjoyment of the game.

```
#define BRIGHTNESS 20
```

# amomii

We hope you had fun setting up and testing your Glow UNO with this guide. Whether you're embarking on your first project or diving into advanced applications, the Glow UNO offers limitless opportunities for creativity and learning. For more inspiration, technical support, or to share your creations, visit amomii.com and join our community. Let your imagination illuminate with the Glow UNO!

amomii.com

linktr.ee/amomii