Chromalchemy

Arduino-Based Color Mixer:

Tinkering with Tints and Tones

by Julius Šlepetis

Abstract:

Chromalchemy is a project aimed at creating an Arduino-based color mixer prototype that demonstrates how RGB colors are created and mixed. The color mixer includes individual red, green, and blue LEDs, along with an RGB LED to display the resultant color. The project has undergone several iterations and improvements to enhance the user experience and improve the understanding of color mixing in the RGB color space for both me as a developer and users of the product.

Introduction and Overview:

The primary objective of this project was to delve into Arduino programming while creating a prototype that demonstrates the fundamentals of RGB color mixing in a visual manner. This project served as a stepping stone to a more ambitious personal project: creating an Arduino-based, VST-controlled LED grid. The color mixer was built using a breadboard, consisting of individual red, green, and blue LEDs, as well as an RGB LED to showcase the resulting color. As the project evolved, several versions were developed, each with different control mechanisms and hardware configurations.

Hardware Configuration:

I initially started my project with an Arduino Uno, a breadboard, and a random assortment of LEDs, pots, and resistors borrowed from the ISSD loan desk. However, I soon realized that the Uno's limited PWM pins might not be sufficient for my project. To overcome this limitation, I switched to an Arduino Mega, which has more PWM pins, but later found out that the limitation can be easily overcome with better hardware design.

For this project, I used an Arduino Uno, two red LEDs, two green LEDs, two blue LEDs, two RGB LEDs, four 10k trim pots to control the colors and brightness, and 220ohm resistors to provide the correct power. The specific choice of components was mainly dictated by their availability at the start, although the LEDs were purchased by myself, to ensure the same manufacturer between the pairs of LEDs

Challenges and Solutions:

One major challenge I faced was adjusting my initial project idea due to time constraints and the limited resources available. I started with the intention of building a device to display audio loudness in a DAW within a set range of frequencies as a VST plug-in. However, I quickly realized that this would be too complex for the given timeframe. As a result, I decided to create a visually appealing color mixer that demonstrates how RGB colors are made and mixed together.

Another significant challenge was dealing with the inverted behavior of the single-colored LEDs compared to the RGB LED. This issue was due to the RGB LED being common cathode, while the single-colored LEDs were common anode. Troubleshooting this problem and finding a solution required multiple attempts at rearranging pins and rewriting the code. Eventually, I managed to resolve the issue by applying mathematical logic to a simplified function on paper to invert the behavior in the code. This allowed the single-colored LEDs to represent the correct color channels and work in conjunction with the RGB LED.

Although during a later redesign I realized that I could just simply flip the single colored LEDs to behave in a desired manner.

Code Explanation:

Version 1: This version uses two potentiometers to control hue and brightness. The hue and brightness values are read from the potentiometers and then converted to RGB values. The RGB values are applied to the RGB LED, while the inverted values are applied to the single-colored LEDs. This is necessary due to the difference in behavior between the RGB and single-colored LEDs.

Version 2: In this version, the color mixer uses one potentiometer to control brightness and a jumper wire to imitate button presses for color selection. The brightness value is read from the potentiometer and applied to the corresponding color channel. A button press toggles the color index, allowing users to switch between channels. The RGB values are applied to the RGB LED, while the inverted values are applied to the single-colored LEDs.

Version 3: In this version, the color mixer uses four potentiometers to control the brightness of each individual RGB channel and overall brightness. The RGB values are read from their respective potentiometers, and the overall brightness is read from a separate potentiometer. The RGB values are then multiplied by the overall brightness value to calculate the upper limit for each channel. Finally, the upper limit values are applied to the RGB LED, and the single-colored LEDs

Version 3 offers several improvements over Version 1:

Individual color channel control: Version 3 allows users to control the brightness of each color channel (R, G, B) individually, providing more precise control over the resulting color mix, just like in Version 2.

Overall brightness control: In addition to individual color channel control, Version 3 introduces a fourth potentiometer for overall brightness control, offering users even greater flexibility in creating their desired color combinations.

Simplified hardware layout: Version 3 adopts a more efficient hardware layout that simplifies connections and improves the visual appearance of the project. The updated hardware layout also fixes the inverted LED behavior by connecting the other terminal of the single-colored LEDs.

If Version 1 were implemented better with three potentiometers to control hue, saturation, and value (HSV) separately, it could provide some benefits:

More intuitive color mixing: Users familiar with the HSV color model might find it more intuitive to mix colors using separate controls for hue, saturation, and value, as it closely resembles how colors are adjusted in digital imaging software.

Greater color range: The ability to control saturation independently in Version 1 could potentially result in a broader range of color combinations compared to Version 2, which only allows users to control the brightness of each color channel.

However, even with an improved Version 1, Version 3 still offers the advantage of individual color channel control, allowing users to fine-tune the resulting color mix. Moreover, the simplified hardware and easier color selection in Version 3 make it a more practical and user-friendly solution for an Arduino-based color mixer project.

Further improvements:

As part of the ongoing refinement process for the Chromalchemy project, I have considered several enhancements that could potentially improve the design, functionality, and overall user experience. Below is a list of potential improvements:

Hardware optimization: As seen in the updated version of Chromalchemy, connecting the RGB channels in parallel or using multiplexers to reduce the number of PWM pins required has improved the project. However, it is still essential to consider the electrical specifications and limitations of the components used in the project to avoid exceeding the Arduino's current limits.

Sliders: To provide a more intuitive user experience, consider replacing the potentiometers with sliders, allowing for more precise control over individual color channels.

Addressable LED strips: For larger projects with higher power requirements or more complex color mixing, consider using addressable LED strips. These strips allow for individual control of each LED's color and brightness, providing greater flexibility and creative possibilities while still working within the limitations of the Arduino board.

External power supply and LED driver ICs: To handle higher power requirements or more extensive LED arrays, consider using external power supplies and LED driver ICs. These components can provide better control and performance while ensuring the project operates safely within the Arduino's current limits.

Enclosure design: Design and build an enclosure for the Chromalchemy color mixer to protect the components and provide a more polished and professional appearance.

Software enhancements: Implement additional software features, such as color presets, color transitions, or animations, to create a more engaging and interactive user experience.

Sharing the project: While the primary goal of this project is personal learning and exploration, consider sharing the project with others as an open-source educational resource. This could help inspire others who are learning Arduino programming, hardware development, and color theory to build upon the work and create their own unique projects.

By incorporating these suggestions and continually refining the project, the Arduino-based color mixer can serve as a valuable learning experience and a starting point for more advanced LED-based projects and applications. It also encourages users to experiment and play with Arduino freely, fostering creativity and a deeper understanding of the technology.

Conclusion:

Chromalchemy successfully demonstrates the principles of RGB color mixing and provides a foundation for future projects involving LEDs and Arduino-controlled systems. Despite some limitations and challenges, such as the use of a jumper wire instead of a proper button and the inverted behavior of single-color LEDs, the project serves as a valuable learning experience in Arduino programming, hardware development, and color theory.