Ambiental Lights for PC

Introduction

The project I have chosen is a device that captures the dominant colors displayed on a PC screen and reproduces them on an LED strip mounted behind it.

1/5

Its purpose is to provide a pleasant experience for users who spend a lot of time in front of the computer, especially gamers.

The idea comes from TVs that have this LED system on the back to provide a unique experience while watching movies.

The utility of the project consists in enhancing the user's visual experience, creating a more captivating atmosphere, and reducing eye strain in low-light conditions.

General Description

×

Hardware Design

Component	Description
Arduino UNO	Microcontroller
LED Strip WS2812B	LEDs for color display
Button	Switches between operation modes
Wires	Connect components on the breadboard
Breadboard	Circuit assembly
Power Source	Powers the circuit
USB Cable	PC Connection

Circuit

×

Schematic

×

Software Design

Used Arduino UNO and an external library (FastLED) for controlling WS2812B LED strip.

• Abstracts the timing-sensitive signal required by addressable LEDs.

Operating Modes

The project features multiple operating modes that can be switched by pressing a button. The mode switching is handled via an external interrupt, allowing fast and responsive detection of the button press event.

```
ISR(INT0_vect) {
    if ((long)(micros() - lastDebounceTime) >= debounceDelay) {
        mode = (mode + 1) % 4;
        lastDebounceTime = micros();
        updateNeeded = true;
    }
}
```

- Mode 0 & 1 All LEDs are set to a specific color (Red and Green)
- Mode 2 Automatic fade effect cycling through Magenta, Cyan, and Blue. Used Timer1 to generate a tick every 1 ms:

```
ISR(TIMER1 COMPA vect) {
   if (mode == 2) { // just in mode 2
       millis counter++;
          if (millis_counter >= 30) { // every 30 ms
              millis counter = 0;
              fadeBrightness += fadeDirection * 5; // change intensity
                 if (fadeBrightness >= 255) {
                     fadeBrightness = 255;
                     fadeDirection = -1; // starts to decrease
                 } else if (fadeBrightness <= 0) {</pre>
                     fadeBrightness = 0;
                     fadeDirection = 1;
                                          // starts to increase
                     fadeColorIndex = (fadeColorIndex + 1) % 3; // change
color after a complete cycle
                 }
                 updateNeeded = true;
          }
```

}

• Mode 3 - Real-time color updates via serial connection using the Adalight protocol (Prismatik)

3/5

Results

Physical Implementation







Multicolor TEST

Avengers

Anyma Visuals

×

Conclusions

In conclusion, this project proved to be a great way to test and apply what I've learned. I was able to build the proposed circuit, connect all the necessary components, and get the expected results. Building and testing the circuit gave me a much clearer, hands-on understanding of how the circuit actually works in practice.

This project was both interesting and rewarding, giving me the chance to put into practice what I learned during the semester. The lab work helped me better understand the subject and made it easier to tackle the final task. In the future, the project could be improved by making it more compact, adding new features, or integrating it into a larger system.

Download

ambientallights.zip

Resources

- 1. https://www.youtube.com/watch?v=nzDnr76fZCc&t=1s
- 2. https://github.com/psieg/Lightpack/releases
- 3. https://github.com/Wifsimster/adalight_ws2812
- 4. https://github.com/FastLED/FastLED
- 5. https://www.tinkercad.com
- 6. https://ocw.cs.pub.ro/courses/pm

From: http://ocw.cs.pub.ro/courses/ - **CS Open CourseWare**

Permanent link: http://ocw.cs.pub.ro/courses/pm/prj2025/cmoarcas/gabriel.tintu

Last update: 2025/05/30 04:57

Export to PDF