

Bonezegei WS2812: Arduino Library for RGB LED

Author : Jofel Batutay

Abstract

Bonezegei WS2812 is an Arduino library for controlling WS2812 or WS2812B RGB LED strips, single LEDs, or matrices. WS2812 LEDs are addressable LEDs that have an integrated circuit inside each LED, allowing them to communicate via a one-wire interface (Bonezegei, 2023). This library supports the ESP32 architecture and provides functions to set the color, brightness, and effects of the LEDs. To use this library, one needs to install it from the Arduino Library Manager or download it from GitHub. The installation process is described in detail by Batutay (2023). This library is useful for creating colorful and dynamic lighting projects with Arduino and WS2812 LEDs.

Code Examples

The library follows the idea of using flowchart-based algorithms to design and implement embedded systems, as described in the paper "Designing and Implementing Embedded Systems Using Flowchart-Based Algorithms: Microcontroller-based Project Examples" by Batutay (2023). The paper provides multiple examples of how to use the library to control the LED strip, such as changing the color, brightness, speed, and pattern of the lights. The library is easy to use and compatible with ESP32 Boards and WS2812 LED strips.

1. Single LED

```
/*
Single WS2812 LED
Author: Bonezegei (Jofel Batutay)
Date: October 2023
Uses 24 bit Hex color format
*/
#include <Bonezegei_WS2812.h>

//param WS2812 pin
Bonezegei_WS2812 rgb(14);

void setup() {
  Serial.begin(115200);
  rgb.begin();
}

void loop() {
  rgb.setPixel(0x0f0000);
  delay(1000);
  rgb.setPixel(0x0f00);
  delay(1000);
  rgb.setPixel(0x0f);
  delay(1000);
  rgb.setPixel(0x0);
  delay(1000);
}
```

2. LED Strip

```
/*
 Strip RGB LED Demo WS2812 LED
 Author: Bonezegei (Jofel Batutay)
 Date: October 2023
 Uses 24 bit Hex color format
 Note : The Colors used on this demo was not set to maximum since it will be too
 bright during testing
 */

#include <Bonezegei_WS2812.h>

#define LED_COUNT 64
Bonezegei_WS2812 rgb(14, LED_COUNT);

void setup() {
    Serial.begin(115200);
    rgb.begin();
}

void loop() {
    rgb.fill(0x0f0000); //RED
    delay(1000);
    rgb.fill(0x0f00); //GREEN
    delay(1000);
    rgb.fill(0x0f); //BLUE
    delay(1000);
    rgb.fill(0x0);
    delay(1000);

    for (int a = 0; a < LED_COUNT; a++) {
        rgb.setPixel(a, 0x0f0000);
        delay(10);
    }
    for (int a = 0; a < LED_COUNT; a++) {
        rgb.setPixel(a, 0x0f00);
        delay(10);
    }

    for (int a = 0; a < LED_COUNT; a++) {
        rgb.setPixel(a, 0x0f);
        delay(10);
    }

    for (int a = 0; a < LED_COUNT; a++) {
        rgb.setPixel(a, a << 16);
        delay(20);
    }

    for (int a = 0; a < LED_COUNT; a++) {
        rgb.setPixel(a, a << 8);
        delay(20);
    }

    for (int a = 0; a < LED_COUNT; a++) {
        rgb.setPixel(a, a);
        delay(20);
    }

    //fade
    for (int a = 0; a < 96; a++) {
        rgb.fill(a); //BLUE
    }
}
```

```
    delay(10);
}
for (int a = 96; a > 0; a--) {
    rgb.fill(a); //BLUE
    delay(10);
}

//fade
for (int a = 0; a < 96; a++) {
    rgb.fill(a << 8);
    delay(10);
}
for (int a = 96; a > 0; a--) {
    rgb.fill(a << 8);
    delay(10);
}

//fade
for (int a = 0; a < 96; a++) {
    rgb.fill(a);
    delay(10);
}
for (int a = 96; a > 0; a--) {
    rgb.fill(a);
    delay(10);
}

//fade
for (int a = 0; a < 96; a++) {
    rgb.fill(a << 16);
    delay(10);
}
for (int a = 96; a > 0; a--) {
    rgb.fill(a << 16);
    delay(10);
}

//fade
for (int a = 0; a < 96; a++) {
    rgb.fill((a << 16) | (a << 8));
    delay(10);
}
for (int a = 96; a > 0; a--) {
    rgb.fill((a << 16) | (a << 8));
    delay(10);
}

//fade
for (int a = 0; a < 96; a++) {
    rgb.fill((a << 8) | a);
    delay(10);
}
for (int a = 96; a > 0; a--) {
    rgb.fill((a << 8) | a);
    delay(10);
}

//fade
for (int a = 0; a < 96; a++) {
    rgb.fill((a << 16) | a);
    delay(10);
}
for (int a = 96; a > 0; a--) {
    rgb.fill((a << 16) | a);
    delay(10);
}
```

```

//fade
for (int a = 0; a < 96; a++) {
    rgb.fill((a << 8) | (a << 16) | a);
    delay(10);
}
for (int a = 96; a > 0; a--) {
    rgb.fill((a << 8) | (a << 16) | a);
    delay(10);
}
}

```

3. RGB Fill

```

/*
Author: Jofel Batutay
Date: October 2023
//Fill the LED with the Primary colors Red Green and Blue
*/
#include <Bonezegei_WS2812.h>

//param1 pin
//param2 number of LED
Bonezegei_WS2812 rgb(14, 64);

void setup() {
    Serial.begin(115200);
    rgb.begin();
}

void loop() {
    rgb.fill(0x0f0000); //RED
    delay(1000);
    rgb.fill(0x0f00);   // GREEN
    delay(1000);
    rgb.fill(0x0f);     // BLUE
    delay(1000);
    rgb.fill(0x0);
    delay(1000);
}

```

4. Matrix 8x8 RGBW

```

/*
Matrix 8x8 RGBW LED Demo WS2812 LED
Author: Bonezegei (Jofel Batutay)
Date: November 2023
Uses 24 bit Hex color format
Note : The Colors used on this demo was not set to maximum since it will be too
bright during testing
*/

#include <Bonezegei_WS2812.h>

//Declare matrix
//param1 pin
//param2 Row
//param3 Column
Bonezegei_WS2812 rgb(14, 8, 8);

void setup() {
    Serial.begin(115200);
    rgb.begin();
}

```

```

rgb.fill(0x0);
delay(5);

for(int a=0; a<8; a++){
    rgb.setPixel(0, a, 0x0F0000); //param1 = X      param2 = Y      param3 = Color
    rgb.setPixel(1, a, 0x0F0000);
}
rgb.display();

for(int a=0; a<8; a++){
    rgb.setPixel(2, a, 0x0F00);
    rgb.setPixel(3, a, 0x0F00);
}
rgb.display();

for(int a=0; a<8; a++){
    rgb.setPixel(4, a, 0x0F);
    rgb.setPixel(5, a, 0x0F);
}
rgb.display();

for(int a=0; a<8; a++){
    rgb.setPixel(6, a, 0x0F0F0F);
    rgb.setPixel(7, a, 0x0F0F0F);
}
rgb.display();
}

void loop() {
}

```

5. Matrix 8x8 Rainbow

```

/*
Matrix 8x8 Rainbow Demo WS2812 LED
Author: Bonezegei (Jofel Batutay)
Date: November 2023
Uses 24 bit Hex color format
Note : The Colors used on this demo was not set to maximum since it will be too
bright during testing
*/

#include <Bonezegei_WS2812.h>

Bonezegei_WS2812 rgb(14, 8, 8);
int c = 0;

void setup() {
    Serial.begin(115200);
    rgb.begin();
}

rgb.fill(0x0);
delay(5);

for (int a = 0; a < 8; a++) {
    rgb.setPixel(0, a, 0x0F0000);
}
rgb.display();

for (int a = 0; a < 8; a++) {
    rgb.setPixel(1, a, 0x0F0400);
}
rgb.display();

```

```

for (int a = 0; a < 8; a++) {
    rgb.setPixel(2, a, 0x040F00);
}
rgb.display();

for (int a = 0; a < 8; a++) {
    rgb.setPixel(3, a, 0x000F00);
}
rgb.display();

for (int a = 0; a < 8; a++) {
    rgb.setPixel(4, a, 0x000F04);
}
rgb.display();

for (int a = 0; a < 8; a++) {
    rgb.setPixel(5, a, 0x00040F);
}
rgb.display();

for (int a = 0; a < 8; a++) {
    rgb.setPixel(6, a, 0x0F000F);
}
rgb.display();

for (int a = 0; a < 8; a++) {
    rgb.setPixel(7, a, 0x0F0F0F);
}
rgb.display();
}

void loop() {
}

```

References

Bonezegei (2023). Bonezegei_WS2812 - Arduino Reference. Retrieved from https://reference.arduino.cc/reference/en/libraries/bonezegei_ws2812/

Batutay, J. (2023). Arduino Library Installation on Arduino 2 IDE: Bonezegei Library. Retrieved from https://www.researchgate.net/publication/376862516_Arduino_Library_Installation_on_Arduino_2_IDE_Bonezegei_Library

Batutay, Jofel. (2023). Arduino Library Installation on Arduino 2 IDE: Bonezegei Library. 10.13140/RG.2.2.34183.57762.

Batutay, Jofel. (2023). Designing and Implementing Embedded Systems Using Flowchart-Based Algorithms: Microcontroller-based Project Examples. 10.13140/RG.2.2.33042.02246.



bonezegei

Website : <https://bonezegei.com>

Github URL: https://github.com/bonezegei/Bonezegei_WS2812