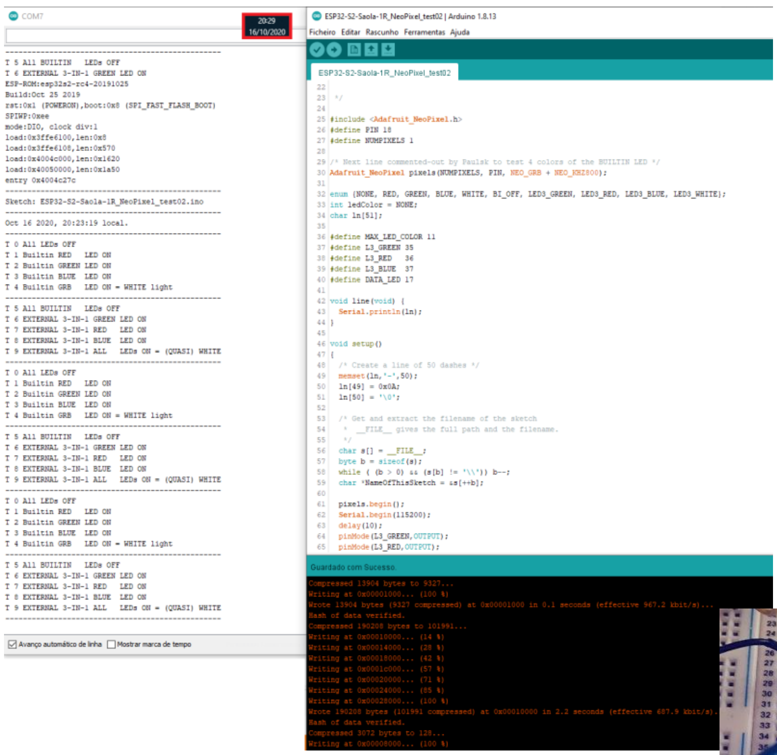
I like to share with you some knowledge I gained during recent experiments with microcontroller boards of the Espressif ESP32-S2 family. Reason: on internet there is a lot of material published about microcontrollers of the Espressif ESP32 family.The ESP32-S2 devices came on the market about a year ago. Outside of YouTube videos announcing the ESP32-S2 family of products, yet I found few publications of specific experiments or experiences. This report gives an answer to a simple aspect of the ESP32-S2: how do I get the builtin LED working?

When I received my first ESP32-S2 series microcontroller development board on August 2 2020, an ESP32-S2-Saola-1R, there was a factory-installed binary running on the device. This factory-installed binary caused the builtin LED to blink in the colors: Green, Red, Blue and White. As soon as I uploaded my first project into this mcu, the builtin LED ceased to lit. Espressif documentation and software is massive but a replacement for the factory installed binary I did not find. I created various test projects while trying to get the builtin LED working. It lasted until Friday, October 16 2020, that I managed to get the builtin LED of an ESP32-S2-Saola-1R blinking: Green, Red, Blue, White and Off, on my desktop PC, using the Arduino IDE version 1.8.13 also using the AdaFruit's NeoPixel library. The sketch I downloaded from: https://tutorial.cytron.io/2020/06/25/program-esp32-s2-using-arduino-ide-unofficial/. I named the sketch: 'ESP32-S2-Saola-1R_NeoPixel_test02.ino'. Beside the builtin LED I had connected to the Saola-1R board: an external 3-in-1 LED and another one small external red LED.

Below an image of the sketch running in the Arduino IDE:

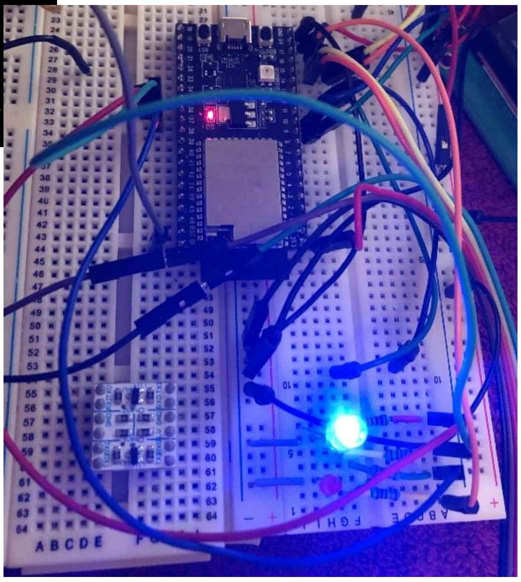
Remarks on the sideline: first I "lost" many days fighting 'starters' problems like installing the ESP-IDF development environment. I installed it in MS Windows. Then I installed in on Ubuntu 20.04 Linux. Then I tried the Arduino IDE v 1.8.13 in MS Windows, but every simple sketch gave an 'wrong head of packet 0x50' error. Only by re-installing the Arduino IDE two days ago and installing the Arduino-ESP library version ESP32S2, these problems became history. Pfff....


COM7
20:29
16/10/2020
ESP32-S2-Saola-1R_NeoPixel_test02 | Arduino 1.8.13
Ficheiro Editar Rascunho Ferramentas Ajuda
ESP32-S2-Saola-1R_NeoPixel_test02
Avanço automático de linha
Mostrar marca de tempo
Guardado com Sucesso.

```
--------------------------------------------------
T 5 All BUILTIN   LEDs OFF
T 6 EXTERNAL 3-IN-1 GREEN LED ON
ESP-ROM:esp32s2-rc4-20191025
Build:Oct 25 2019
rst:0x1 (POWERON),boot:0x8 (SPI_FAST_FLASH_BOOT)
SPIWP:0xee
mode:DIO, clock div:1
load:0x3ffe6100,len:0x8
load:0x3ffe6108,len:0x570
load:0x4004c000,len:0x1620
load:0x40050000,len:0x1a50
entry 0x4004c27c
--------------------------------------------------
Sketch: ESP32-S2-Saola-1R_NeoPixel_test02.ino
--------------------------------------------------
Oct 16 2020, 20:23:19 local.
--------------------------------------------------
T 0 All LEDs OFF
T 1 Builtin RED   LED ON
T 2 Builtin GREEN LED ON
T 3 Builtin BLUE  LED ON
T 4 Builtin GRB   LED ON = WHITE light
--------------------------------------------------
T 5 All BUILTIN   LEDs OFF
T 6 EXTERNAL 3-IN-1 GREEN LED ON
T 7 EXTERNAL 3-IN-1 RED   LED ON
T 8 EXTERNAL 3-IN-1 BLUE  LED ON
T 9 EXTERNAL 3-IN-1 ALL   LEDs ON = (QUASI) WHITE
--------------------------------------------------
T 0 All LEDs OFF
T 1 Builtin RED   LED ON
T 2 Builtin GREEN LED ON
T 3 Builtin BLUE  LED ON
T 4 Builtin GRB   LED ON = WHITE light
--------------------------------------------------
T 5 All BUILTIN   LEDs OFF
T 6 EXTERNAL 3-IN-1 GREEN LED ON
T 7 EXTERNAL 3-IN-1 RED   LED ON
T 8 EXTERNAL 3-IN-1 BLUE  LED ON
T 9 EXTERNAL 3-IN-1 ALL   LEDs ON = (QUASI) WHITE
--------------------------------------------------
T 0 All LEDs OFF
T 1 Builtin RED   LED ON
T 2 Builtin GREEN LED ON
T 3 Builtin BLUE  LED ON
T 4 Builtin GRB   LED ON = WHITE light
--------------------------------------------------
T 5 All BUILTIN   LEDs OFF
T 6 EXTERNAL 3-IN-1 GREEN LED ON
T 7 EXTERNAL 3-IN-1 RED   LED ON
T 8 EXTERNAL 3-IN-1 BLUE  LED ON
T 9 EXTERNAL 3-IN-1 ALL   LEDs ON = (QUASI) WHITE
--------------------------------------------------
```

```cpp

 */

#include <Adafruit_NeoPixel.h>
#define PIN 18
#define NUMPIXELS 1

/* Next line commented-out by Paulsk to test 4 colors of the BUILTIN LED */
Adafruit_NeoPixel pixels(NUMPIXELS, PIN, NEO_GRB + NEO_KHZ800);

enum {NONE, RED, GREEN, BLUE, WHITE, BI_OFF, LED3_GREEN, LED3_RED, LED3_BLUE, LED3_WHITE};
int ledColor = NONE;
char ln[51];

#define MAX_LED_COLOR 11
#define L3_GREEN 35
#define L3_RED   36
#define L3_BLUE  37
#define DATA_LED 17

void line(void) {
  Serial.println(ln);
}

void setup()
{
  /* Create a line of 50 dashes */
  memset(ln,'-',50);
  ln[49] = 0x0A;
  ln[50] = '\0';

  /* Get and extract the filename of the sketch
   *  __FILE__ gives the full path and the filename.
   */
  char s[] = __FILE__;
  byte b = sizeof(s);
  while ( (b > 0) && (s[b] != '\\')) b--;
  char *NameOfThisSketch = &s[++b];

  pixels.begin();
  Serial.begin(115200);
  delay(10);
  pinMode(L3_GREEN,OUTPUT);
  pinMode(L3_RED,OUTPUT);
```

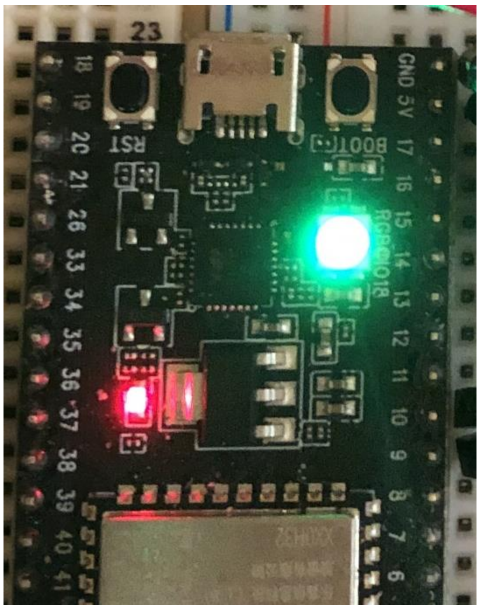
```
Compressed 13904 bytes to 9327...
Writing at 0x00001000... (100 %)
Wrote 13904 bytes (9327 compressed) at 0x00001000 in 0.1 seconds (effective 967.2 kbit/s)...
Hash of data verified.
Compressed 190208 bytes to 101991...
Writing at 0x00010000... (14 %)
Writing at 0x00014000... (28 %)
Writing at 0x00018000... (42 %)
Writing at 0x0001c000... (57 %)
Writing at 0x00020000... (71 %)
Writing at 0x00024000... (85 %)
Writing at 0x00028000... (100 %)
Wrote 190208 bytes (101991 compressed) at 0x00010000 in 2.2 seconds (effective 687.9 kbit/s).
Hash of data verified.
Compressed 3072 bytes to 128...
Writing at 0x00008000... (100 %)
```

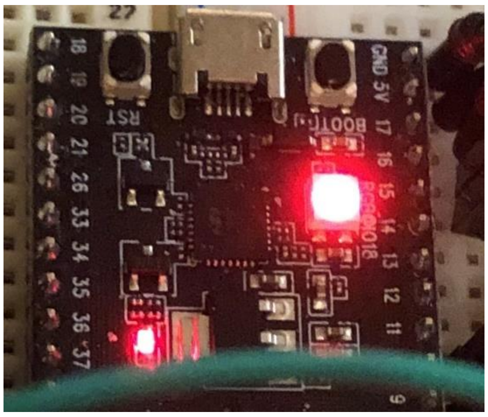


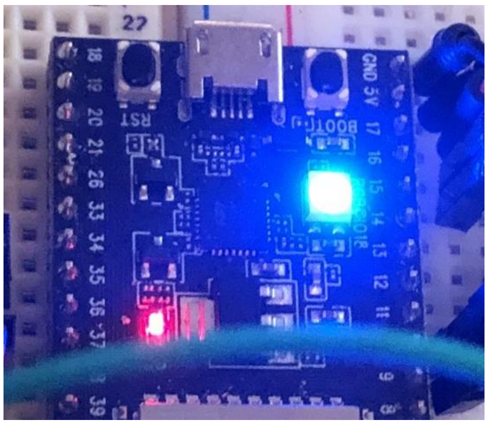
Within the Arduino IDE I selected as type of board: 'ESP32S2 Dev Module'.

Image to the right: the external 3-in-1 LED Displaying the blue led part.

The builtin LED of the Saola-1R board.



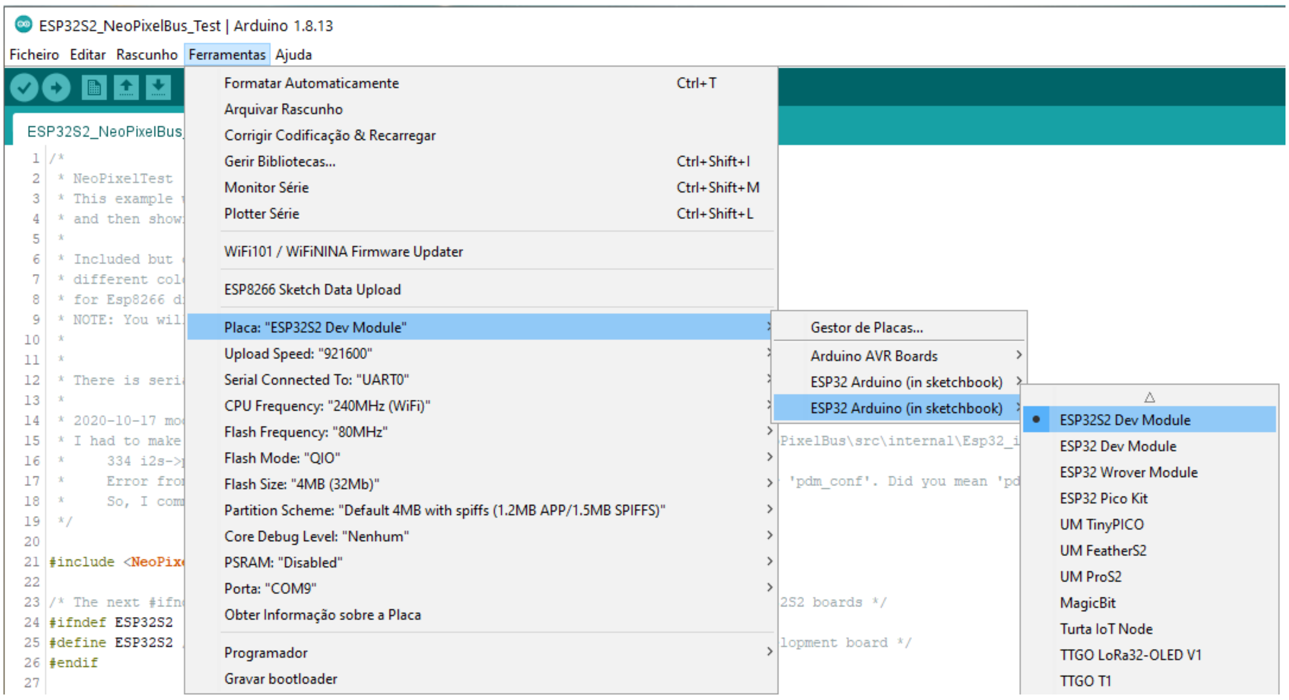




The Saola-1 has GPIO pin 18 connected to the builtin LED (as labeled on the PCB near the RGB LED: 'RGB@GPIO18').

This was the first test that I was able to get the builtin LED of the ESP32-S2-Saola-1R working, in a way that I had in mind.

Below a screenshot showing the type of board used (labels in Portuguese language).



ESP32S2_NeoPixelBus_Test | Arduino 1.8.13

Ficheiro Editar Rascunho Ferramentas Ajuda

ESP32S2_NeoPixelBus

```
/*
 * NeoPixelTest
 * This example
 * and then show
 *
 * Included but
 * different col
 * for Esp8266 d
 * NOTE: You wil
 *
 *
 * There is seri
 *
 * 2020-10-17 mo
 * I had to make
 *     334 i2s->
 *     Error fro
 *     So, I com
 */

#include <NeoPix

/* The next #ifn
#ifndef ESP32S2
#define ESP32S2
#endif
```

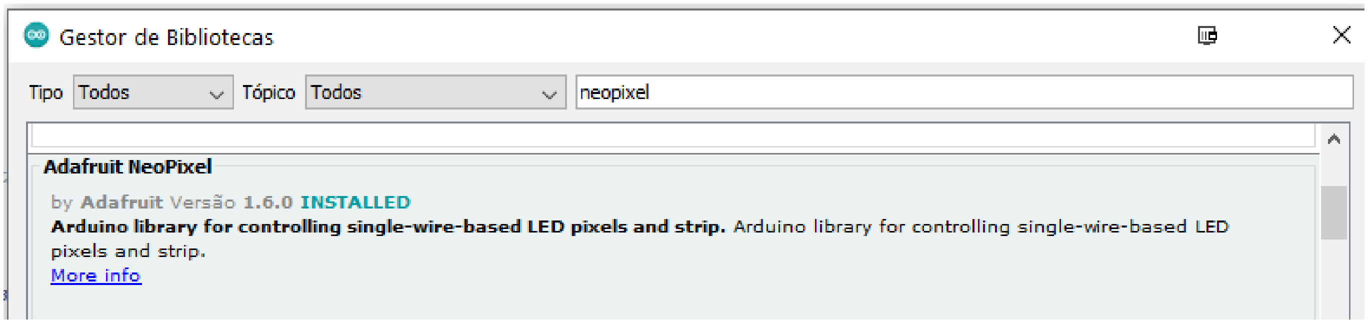
Formatar Automaticamente	Ctrl+T
Arquivar Rascunho	
Corrigir Codificação & Recarregar	
Gerir Bibliotecas...	Ctrl+Shift+I
Monitor Série	Ctrl+Shift+M
Plotter Série	Ctrl+Shift+L
WiFi101 / WiFiNINA Firmware Updater	
ESP8266 Sketch Data Upload	
Placa: "ESP32S2 Dev Module"	
Upload Speed: "921600"	
Serial Connected To: "UART0"	
CPU Frequency: "240MHz (WiFi)"	
Flash Frequency: "80MHz"	
Flash Mode: "QIO"	
Flash Size: "4MB (32Mb)"	
Partition Scheme: "Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS)"	
Core Debug Level: "Nenhum"	
PSRAM: "Disabled"	
Porta: "COM9"	
Obter Informação sobre a Placa	
Programador	
Gravar bootloader	

Gestor de Placas...
Arduino AVR Boards
ESP32 Arduino (in sketchbook)
ESP32 Arduino (in sketchbook)

● ESP32S2 Dev Module
ESP32 Dev Module
ESP32 Wrover Module
ESP32 Pico Kit
UM TinyPICO
UM FeatherS2
UM ProS2
MagicBit
Turta IoT Node
TTGO LoRa32-OLED V1
TTGO T1

```
PixelBus\src\internal\Esp32_i
'pdm_conf'. Did you mean 'pd
2S2 boards */
lopment board */
```

Image: the Adafruit NeoPixel library installed in the Arduino IDE.



Adafruit NeoPixel

by Adafruit Versão 1.6.0 INSTALLED
Arduino library for controlling single-wire-based LED pixels and strip. Arduino library for controlling single-wire-based LED pixels and strip.
More info

Next the listing of the sketch of the test with the ESP32-S2-Saola-1R board:

```
/*
  Tutorial: Program ESP32-S2 Using Arduino IDE (Unofficial)
  Board: ESP32S2 (Cucumber from Gravitech)

  External libraries:
  - Adafruit NeoPixel V1.5.0 (Manager)

  Notes Paulsk 2020-10-16:

  I downloaded this sketch today from: https://tutorial.cytron.io/2020/06/25/program-esp32-s2-using-arduino-ide-unofficial/

  It worked immediately on my ESP32-S2-Saola-1R. Only the colors Green, Red, Blue and nothing were shown. Not the white color.
  I added 'white' in the enumeration definition in line 28 and I added 'case WHITE' in the switch structure from line 59 to get the white color
  be displayed also. Then I added to the swtich structure parts for the external 3-in-1 LED. I also added some other parts like the definition of
  ln, the function line(), the definition, extracting and printing of the name of the sketch file and printing of date and time.
  As we know from physics lessons displaying Green, Red and Blue colors together, results in seeing White color.
  The value for brightness as originally programmed in this sketch is nice, not too strong as default brightness I have seen in my previous
  Experiments.Anyway, I decided to dim Green a bit to half of the original brightness value. I also set the intensity of the three colors
  to make white to a quarter of  the original value for each of the colors: 5.

  Until this example sketch I never was able to control good the internal LED of neither the Saola-1R or the Kaluga-1 development board.

  I experienced, by chance, while uploading this sketch that a LED connected to hardware pin 19 (= GPIO16) lit at the moment the Arduino
  IDE was uploading data to the ESP32-S2-Saola-1R.

 */
#include <Adafruit_NeoPixel.h>
#define PIN 18
#define NUMPIXELS 1

Adafruit_NeoPixel pixels(NUMPIXELS, PIN, NEO_GRB + NEO_KHZ800);

enum {NONE, GREEN, RED, BLUE, WHITE, BI_OFF, LED3_GREEN, LED3_RED, LED3_BLUE, LED3_WHITE};
int ledColor = NONE;
char ln[51];

#define MAX_LED_COLOR 11
#define L3_GREEN 35
#define L3_RED   36
#define L3_BLUE  37
#define DATA_LED 17

void line(void) {
  Serial.println(ln);
}
```

```
void setup()
{
 /* Create a line of 50 dashes */
 memset(ln,'-',50);
 ln[49] = 0x0A; /* followed by a new line character */
 ln[50] = '\0';     /* end of string marker */

 /* Get and extract the filename of the sketch
  *  __FILE__ gives the full path and the filename.
  */
 char s[] = __FILE__;  /* __FILE__ contains a full path with the filename of the sketch */
 byte b = sizeof(s);
 while ( (b > 0) && (s[b] != '\\')) b--;    /* This code snippet extracts the filename of the sketch from the full path name */
 char *NameOfThisSketch = &s[++b];

 pixels.begin();
 Serial.begin(115200);
 delay(10);
 pinMode(L3_GREEN,OUTPUT);
 pinMode(L3_RED,OUTPUT);
 pinMode(L3_BLUE, OUTPUT);
 pinMode(DATA_LED, OUTPUT);
 /* Print the filename of this sketch */
 Serial.print(ln);
 Serial.print("Sketch: ");
 Serial.println(NameOfThisSketch);
 Serial.print(ln);
 /* Print also the current date and time */
 Serial.println((__DATE__ ", " __TIME__ " local."));
 Serial.print(ln);
}

void loop()
{
 switch (ledColor) {
   case NONE:
     Serial.println("T 0 All LEDs OFF");
     digitalWrite(L3_GREEN,LOW);
     digitalWrite(L3_RED,  LOW); /* Switch off the 3 conventional LEDs (3-in-1) */
     digitalWrite(L3_BLUE, LOW);
     pixels.setPixelColor(0, pixels.Color(0, 0, 0));
     pixels.show();
     break;

   case GREEN:
     Serial.println("T 2 Builtin GREEN LED ON");
     pixels.setPixelColor(0, pixels.Color(0, 10, 0));
     pixels.show();
     break;
   case RED:
     Serial.println("T 1 Builtin RED   LED ON");
     pixels.setPixelColor(0, pixels.Color(20, 0, 0));
     pixels.show();
     break;

   case BLUE:
   Serial.println("T 3 Builtin BLUE  LED ON");
     pixels.setPixelColor(0, pixels.Color(0, 0, 20));
     pixels.show();
     break;

   case WHITE:
     Serial.println("T 4 Builtin GRB   LED ON = WHITE light");
     Serial.print(ln);
     pixels.setPixelColor(0, pixels.Color(5, 5, 5));
     pixels.show();
```

```
      break;
    case BI_OFF:
      Serial.println("T 5 All BUILTIN   LEDs OFF");
      pixels.setPixelColor(0, pixels.Color(0, 0, 0));
      pixels.show();
      break;

    case LED3_GREEN:
      Serial.println("T 6 EXTERNAL 3-IN-1 GREEN LED ON");
      digitalWrite(L3_GREEN, HIGH);
      digitalWrite(L3_RED,    LOW);
      digitalWrite(L3_BLUE,   LOW);
      break;

    case LED3_RED:
      Serial.println("T 7 EXTERNAL 3-IN-1 RED   LED ON");
      digitalWrite(L3_GREEN, LOW);
      digitalWrite(L3_RED,  HIGH);
      digitalWrite(L3_BLUE,  LOW);
      break;

    case LED3_BLUE:
     Serial.println("T 8 EXTERNAL 3-IN-1 BLUE  LED ON");
      digitalWrite(L3_GREEN, LOW);
      digitalWrite(L3_RED,   LOW);
      digitalWrite(L3_BLUE, HIGH);
      break;

    case LED3_WHITE:
      Serial.println("T 9 EXTERNAL 3-IN-1 ALL   LEDs ON = (QUASI) WHITE");
      Serial.print(ln);
      digitalWrite(L3_GREEN, HIGH);
      digitalWrite(L3_RED,   HIGH);
      digitalWrite(L3_BLUE,  HIGH);
      break;

    default:
      break;
  }

  ledColor++;
  if (ledColor == MAX_LED_COLOR) {
    ledColor = NONE;
  }

  delay(1000);
}
```

This completes the test of the builtin LED of an ESP32-S2-Saola-1R.

On September 28 2020, I received another ESP32-S2 series microcontoller board: an ESP32-S2-Kaluga-1 development board, complete with audio-board, lcd-board, touchpanel and camera. After the experience gained as described above, I continued my experiments, first focussed on the builtin LED of the Kaluga-1 board. This board has its builtin LED connected to PIN 45. Having learned a lesson from my mistake with the Saola-1R board: not making a backup of the factory-installed binary (also at that moment not yet customed with new tools like the ESPTOOL) I now first created a backup copy of the factory-installed binary of the Kaluga-1 board before uploading any binary to the Kaluga-1 board.

On Saturday October 17 2020 I modified the sketch used for the Saola-1R to get it running on an ESP32-S2-Kaluga-1 development board. I managed to get the sketch running but the builtin LED did not lit.
Then I visited the webpage: https://learn.adafruit.com/adafruit-neopixel-uberguide/arduino-library-use, but I was not able to get the sketch working on the Kaluga-1.
I continued my research. I google'd. I found the NeoPixelBus library from the Github webpage: https://github.com/Makuna/NeoPixelBus. By studying the material, I read that the NeoPixelBus software initially was aimed for the ESP8266.
The author, Michael Miller, alias Makuna, wrote a WiKi-page on: https://github.com/Makuna/NeoPixelBus/wiki/Library-Comparisons.

It appeared to me that this library possibly would be suitable for the Kaluga-1 board. That is why I gave it a try. I cloned the NeoPixelBus library into the folder: <Drive>:\Users\<User>\<Documents>\Arduino\libraries (on my desktop PC running MS Windows 10 Pro ver 2004).
From the NeoPixelBus library folder, I selected the example: NeoPixelBus/Examples/NeoPixelTest/NeoPixelTest.ino. I studied this sketch. Then I modified this sketch to make it ready to use with the Kaluga-1 board.

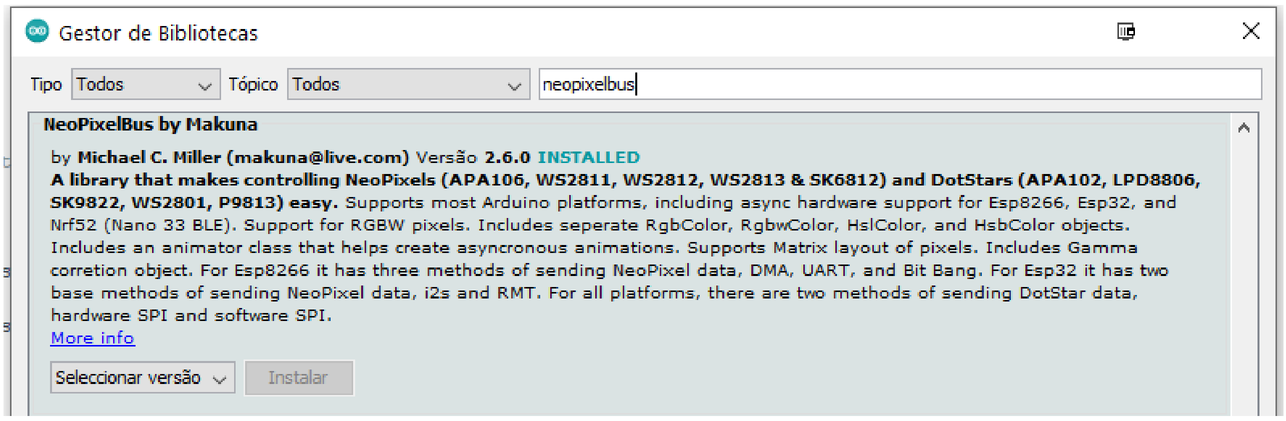
When I ran the modified sketch for the first time, the compiler reported several errors related to a non-existant member of an object, in the file:
<Drive>:\Users\<User>\<Documents>\Arduino\libraries\NeoPixelBus\src\internal\Esp32_i2c.c, lines: 334, …
The error text was: 'i2s_dev_t' has no member 'pdm_conf'. This is why I commented-out the lines: 334, 336, 364 and 365:

```
334   // i2s->pdm_conf.pcm2pdm_conv_en = 0; Note Paulsk: error from Arduino IDE compiler
      // for ESP32S2 Dev Module: 'i2s_dev_t' has no member 'pdm_conf'
      // Did you mean 'pd_conf'? So, I commented-out line 334, 336, 364, 365
336   //i2s->pdm_conf.pdm2pcm_conv_en = 0;
364   //i2s->pdm_conf.rx_pdm_en = 0;
365   //i2s->pdm_conf.tx_pdm_en = 0;
```

Next I was able to run the sketch. I had to make some more alterations. Then the sketch was running in a way that I intended. The builtin LED of the Kaluga-1 board was altering color, starting at Green, then Red, Blue, White and Off. I took several images. I also created a short video. The Kaluga-1 has GPIO pin 45 connected to the builtin LED. The maximum brilliance of the builtin LED is very strong. Looking at it, my eyes were blinded a short time. That is why I used much lowered value of colorSaturation from the initial defined value of 128 to 20.

Also for this test the 'ESP32S2 Dev Board' was selected in the Arduino IDE.

Below a screenshot of the NeoPixelBus library installed in the Arduino IDE:

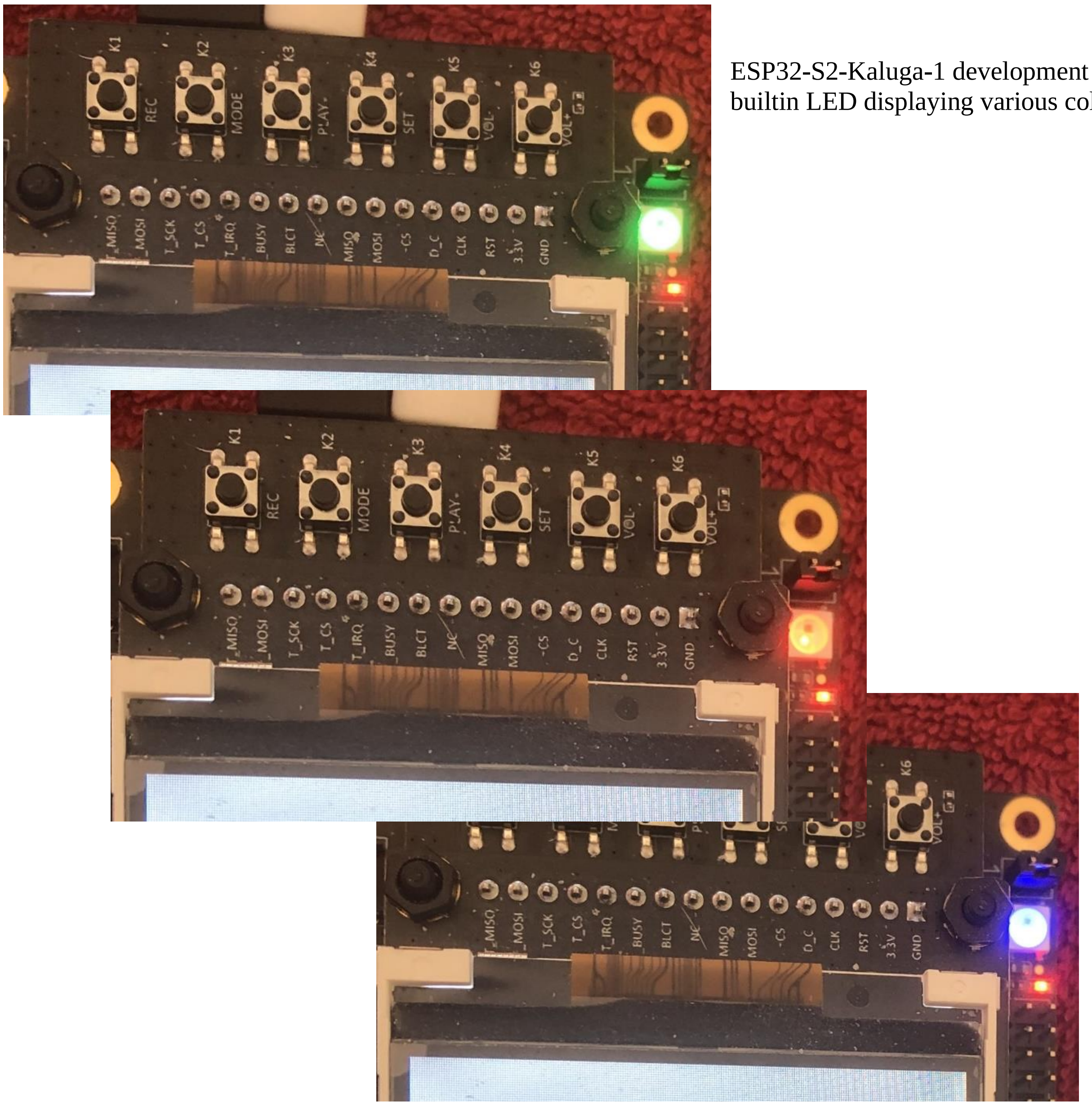

Gestor de Bibliotecas

Tipo Todos Tópico Todos neopixelbus

NeoPixelBus by Makuna

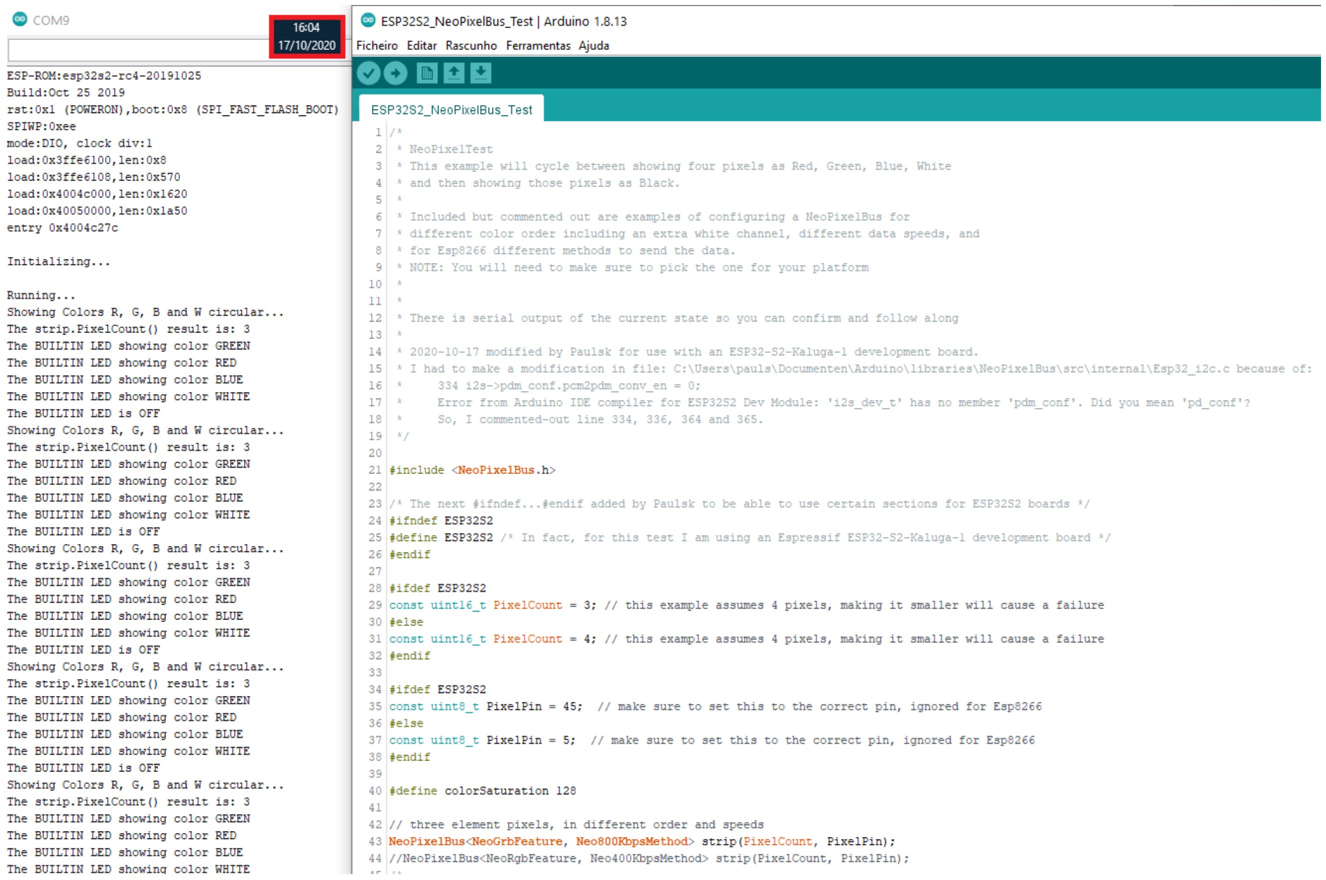
by **Michael C. Miller (makuna@live.com)** Versão **2.6.0** **INSTALLED**
A library that makes controlling NeoPixels (APA106, WS2811, WS2812, WS2813 & SK6812) and DotStars (APA102, LPD8806, SK9822, WS2801, P9813) easy. Supports most Arduino platforms, including async hardware support for Esp8266, Esp32, and Nrf52 (Nano 33 BLE). Support for RGBW pixels. Includes seperate RgbColor, RgbwColor, HslColor, and HsbColor objects. Includes an animator class that helps create asyncronous animations. Supports Matrix layout of pixels. Includes Gamma coretion object. For Esp8266 it has three methods of sending NeoPixel data, DMA, UART, and Bit Bang. For Esp32 it has two base methods of sending NeoPixel data, i2s and RMT. For all platforms, there are two methods of sending DotStar data, hardware SPI and software SPI.
More info

Seleccionar versão Instalar



ESP32-S2-Kaluga-1 development board's builtin LED displaying various colors.

Below an image of the sketch running in the Arduino IDE.


COM9
16:04
17/10/2020
ESP32S2_NeoPixelBus_Test | Arduino 1.8.13
Ficheiro Editar Rascunho Ferramentas Ajuda
ESP32S2_NeoPixelBus_Test

```
ESP-ROM:esp32s2-rc4-20191025
Build:Oct 25 2019
rst:0x1 (POWERON),boot:0x8 (SPI_FAST_FLASH_BOOT)
SPIWP:0xee
mode:DIO, clock div:1
load:0x3ffe6100,len:0x8
load:0x3ffe6108,len:0x570
load:0x4004c000,len:0x1620
load:0x40050000,len:0x1a50
entry 0x4004c27c

Initializing...

Running...
Showing Colors R, G, B and W circular...
The strip.PixelCount() result is: 3
The BUILTIN LED showing color GREEN
The BUILTIN LED showing color RED
The BUILTIN LED showing color BLUE
The BUILTIN LED showing color WHITE
The BUILTIN LED is OFF
Showing Colors R, G, B and W circular...
The strip.PixelCount() result is: 3
The BUILTIN LED showing color GREEN
The BUILTIN LED showing color RED
The BUILTIN LED showing color BLUE
The BUILTIN LED showing color WHITE
The BUILTIN LED is OFF
Showing Colors R, G, B and W circular...
The strip.PixelCount() result is: 3
The BUILTIN LED showing color GREEN
The BUILTIN LED showing color RED
The BUILTIN LED showing color BLUE
The BUILTIN LED showing color WHITE
The BUILTIN LED is OFF
Showing Colors R, G, B and W circular...
The strip.PixelCount() result is: 3
The BUILTIN LED showing color GREEN
The BUILTIN LED showing color RED
The BUILTIN LED showing color BLUE
The BUILTIN LED showing color WHITE
The BUILTIN LED is OFF
Showing Colors R, G, B and W circular...
The strip.PixelCount() result is: 3
The BUILTIN LED showing color GREEN
The BUILTIN LED showing color RED
The BUILTIN LED showing color BLUE
The BUILTIN LED showing color WHITE
```

```
/*
 * NeoPixelTest
 * This example will cycle between showing four pixels as Red, Green, Blue, White
 * and then showing those pixels as Black.
 *
 * Included but commented out are examples of configuring a NeoPixelBus for
 * different color order including an extra white channel, different data speeds, and
 * for Esp8266 different methods to send the data.
 * NOTE: You will need to make sure to pick the one for your platform
 *
 *
 * There is serial output of the current state so you can confirm and follow along
 *
 * 2020-10-17 modified by Paulsk for use with an ESP32-S2-Kaluga-1 development board.
 * I had to make a modification in file: C:\Users\pauls\Documenten\Arduino\libraries\NeoPixelBus\src\internal\Esp32_i2c.c because of:
 *     334 i2s->pdm_conf.pcm2pdm_conv_en = 0;
 *     Error from Arduino IDE compiler for ESP32S2 Dev Module: 'i2s_dev_t' has no member 'pdm_conf'. Did you mean 'pd_conf'?
 *     So, I commented-out line 334, 336, 364 and 365.
 */

#include <NeoPixelBus.h>

/* The next #ifndef...#endif added by Paulsk to be able to use certain sections for ESP32S2 boards */
#ifndef ESP32S2
#define ESP32S2 /* In fact, for this test I am using an Espressif ESP32-S2-Kaluga-1 development board */
#endif

#ifdef ESP32S2
const uint16_t PixelCount = 3; // this example assumes 4 pixels, making it smaller will cause a failure
#else
const uint16_t PixelCount = 4; // this example assumes 4 pixels, making it smaller will cause a failure
#endif

#ifdef ESP32S2
const uint8_t PixelPin = 45;  // make sure to set this to the correct pin, ignored for Esp8266
#else
const uint8_t PixelPin = 5;  // make sure to set this to the correct pin, ignored for Esp8266
#endif

#define colorSaturation 128

// three element pixels, in different order and speeds
NeoPixelBus<NeoGrbFeature, Neo800KbpsMethod> strip(PixelCount, PixelPin);
//NeoPixelBus<NeoRgbFeature, Neo400KbpsMethod> strip(PixelCount, PixelPin);
```

Listing of the sketch for testing the builtin LED on an ESP32-S2-Kaluga-1 development board:

```
/*
 * NeoPixelTest
 * This example will cycle between showing four pixels as Red, Green, Blue, White
 * and then showing those pixels as Black.
 *
 * Included but commented out are examples of configuring a NeoPixelBus for
 * different color order including an extra white channel, different data speeds, and
 * for Esp8266 different methods to send the data.
 * NOTE: You will need to make sure to pick the one for your platform
 *
 *
 * There is serial output of the current state so you can confirm and follow along
 *
 * 2020-10-17 modified by Paulsk for use with an ESP32-S2-Kaluga-1 development board.
 * I had to make a modification in file: <Drive:>:\Users\<User>\<Documents>\Arduino\libraries\NeoPixelBus\src\internal\Esp32_i2c.c
because of:
 *     334 i2s->pdm_conf.pcm2pdm_conv_en = 0;
 *     Error from Arduino IDE compiler for ESP32S2 Dev Module: 'i2s_dev_t' has no member 'pdm_conf'. Did you mean 'pd_conf'?
 *     So, I commented-out line 334, 336, 364 and 365.
 */

#include <NeoPixelBus.h>

/* The next #ifndef...#endif added by Paulsk to be able to use certain sections for ESP32S2 boards */
#ifndef ESP32S2
#define ESP32S2 /* In fact, for this test I am using an Espressif ESP32-S2-Kaluga-1 development board */
#endif
```

```
#ifdef ESP32S2
const uint16_t PixelCount = 3; // this example assumes 4 pixels, making it smaller will cause a failure
#else
const uint16_t PixelCount = 4; // this example assumes 4 pixels, making it smaller will cause a failure
#endif

#ifdef ESP32S2
const uint8_t PixelPin = 45;  // make sure to set this to the correct pin, ignored for Esp8266
#else
const uint8_t PixelPin = 5;  // make sure to set this to the correct pin, ignored for Esp8266
#endif

#define colorSaturation 128

// three element pixels, in different order and speeds
NeoPixelBus<NeoGrbFeature, Neo800KbpsMethod> strip(PixelCount, PixelPin);
//NeoPixelBus<NeoRgbFeature, Neo400KbpsMethod> strip(PixelCount, PixelPin);
/*
 * For Esp8266, the Pin is omitted and it uses GPIO3 due to DMA hardware use.
 * There are other Esp8266 alternative methods that provide more pin options, but also have
 * other side effects.
 * for details see wiki linked here https://github.com/Makuna/NeoPixelBus/wiki/ESP8266-NeoMethods

 * You can also use one of these for Esp8266,
 * each having their own restrictions
 *
 * These two are the same as above as the DMA method is the default
 * NOTE: These will ignore the PIN and use GPI03 pin
 * NeoPixelBus<NeoGrbFeature, NeoEsp8266Dma800KbpsMethod> strip(PixelCount, PixelPin);
 * NeoPixelBus<NeoRgbFeature, NeoEsp8266Dma400KbpsMethod> strip(PixelCount, PixelPin);

 * Uart method is good for the Esp-01 or other pin restricted modules
 * for details see wiki linked here https://github.com/Makuna/NeoPixelBus/wiki/ESP8266-NeoMethods
 * NOTE: These will ignore the PIN and use GPI02 pin
 * NeoPixelBus<NeoGrbFeature, NeoEsp8266Uart1800KbpsMethod> strip(PixelCount, PixelPin);
 * NeoPixelBus<NeoRgbFeature, NeoEsp8266Uart1400KbpsMethod> strip(PixelCount, PixelPin);

 * The bitbang method is really only good if you are not using WiFi features of the ESP
 * It works with all but pin 16
 * NeoPixelBus<NeoGrbFeature, NeoEsp8266BitBang800KbpsMethod> strip(PixelCount, PixelPin);
 * NeoPixelBus<NeoRgbFeature, NeoEsp8266BitBang400KbpsMethod> strip(PixelCount, PixelPin);

 * four element pixels, RGBW
 * NeoPixelBus<NeoRgbwFeature, Neo800KbpsMethod> strip(PixelCount, PixelPin);
*/


#ifdef ESP32S2
RgbColor red(20, 0, 0); /* 20 is the value I used instead of the defined 'colorSaturation of 128' */
RgbColor green(0, 20, 0);
RgbColor blue(0, 0, 20);
RgbColor white(20, 20, 20);
RgbColor black(0, 0, 0);
#else
RgbColor red(colorSaturation, 0, 0);
RgbColor green(0, colorSaturation, 0);
RgbColor blue(0, 0, colorSaturation);
RgbColor white(colorSaturation);
RgbColor black(0);

HslColor hslRed(red);
HslColor hslGreen(green);
HslColor hslBlue(blue);
HslColor hslWhite(white);
HslColor hslBlack(black);
#endif
```

```
void setup()
{
   Serial.begin(115200);
   while (!Serial); // wait for serial attach

   Serial.println();
   Serial.println("Initializing...");
   Serial.flush();

   // this resets all the neopixels to an off state
   strip.Begin();
   strip.Show();


   Serial.println();
   Serial.println("Running...");
}

void loop()
{
   //delay(5000);

   Serial.println("Showing Colors R, G, B and W circular...");
   /* The next two lines added by Paulsk */
   Serial.print("The strip.PixelCount() result is: ");
   Serial.println(strip.PixelCount());

#ifdef ESP32S2
   // set the colors,
   // if they don't match in order, you need to use NeoGrbFeature feature
   Serial.println("The BUILTIN LED showing color GREEN");
   strip.SetPixelColor(0, green); /* Modification by Paulsk: 1, changed into 0, */
   strip.Show();
   delay(1000);
   Serial.println("The BUILTIN LED showing color RED");
   strip.SetPixelColor(0, red);
   strip.Show();
   delay(1000);
   Serial.println("The BUILTIN LED showing color BLUE");
   strip.SetPixelColor(0, blue); /* Modification by Paulsk: 2, changed into 0, */
   strip.Show();
   delay(1000);
   Serial.println("The BUILTIN LED showing color WHITE");
   strip.SetPixelColor(0, white); /* Modification by Paulsk: 3, changed into 0, */
   strip.Show();
   delay(1000);
   Serial.println("The BUILTIN LED is OFF");
   //Serial.println("Off ...");
   // turn off the pixels
   strip.SetPixelColor(0, black);
   strip.Show();
   delay(1000);
#else
   // set the colors,
   // if they don't match in order, you need to use NeoGrbFeature feature
   strip.SetPixelColor(0, red);
   delay(1000);
   strip.SetPixelColor(1, green);
   delay(1000);
   strip.SetPixelColor(2, blue);
   delay(1000);
   strip.SetPixelColor(3, white);
   delay(1000);
   // the following line demonstrates rgbw color support
   // if the NeoPixels are rgbw types the following line will compile
   // if the NeoPixels are anything else, the following line will give an error
```

```
  //strip.SetPixelColor(3, RgbwColor(colorSaturation));
  strip.Show();
  delay(5000);

  Serial.println("Off ...");

  // turn off the pixels
  strip.SetPixelColor(0, black);
  strip.SetPixelColor(1, black);
  strip.SetPixelColor(2, black);
  strip.SetPixelColor(3, black);
  strip.Show();

  delay(5000);

  Serial.println("HSL Colors R, G, B, W...");
  // set the colors,
  // if they don't match in order, you may need to use NeoGrbFeature feature
  strip.SetPixelColor(0, hslRed);
  strip.SetPixelColor(1, hslGreen);
  strip.SetPixelColor(2, hslBlue);
  strip.SetPixelColor(3, hslWhite);
  strip.Show();

  delay(5000);

  Serial.println("Off again...");

  // turn off the pixels
  strip.SetPixelColor(0, hslBlack);
  strip.SetPixelColor(1, hslBlack);
  strip.SetPixelColor(2, hslBlack);
  strip.SetPixelColor(3, hslBlack);
  strip.Show();
#endif
}
```

This ends my notes on the experiments to get the builtin LEDs of an ESP32-S2-Saola-1R and an ESP32-S2-Kaluga-1 displaying various colors, using the Arduino IDE v1.8.13 in a Microsoft Windows 10 Pro (v2004) operating system environment.

I am awaiting an ordered panel with 64 color LEDs of the WS8212 family.

Finally I want to express my gratitude to all the workers at Espressif, Adafruit and many other great people out there in the world that, paid or volontarily, spent a lot of time to make great software, documentation and examples which they publish for the world community of electronics hobbyists. There are too many people to thank but one I want to name: Neil Kolban from Texas. Neil, if you read this: thank you very much for your contributions, especially your: ‘Kolban’s book on ESP32’. I learned a lot from it. I will continue to read parts of it.
Thanks guys! Lets continue to share our knowledge and experience.

A few facts about me: Dutchman, retired, 74. Former ‘sparks’ in the Royal Dutch Navy (and Naval LRMP’s) 14yrs. Then almost 30yrs for the Dutch FCC. Living in Lisbon, Portugal since 2012. 1st ‘computer’: an Apple][in 1978. Since then hooked on ‘computing’, hard- and software. Last years: with Raspberry Pi and Arduino. As you read above: recently focussed on ESP32-S2.

Paulus Schulinck
Lisbon, October 17, 2020

Reactions, feedback welcome via twitter: @ct7agr