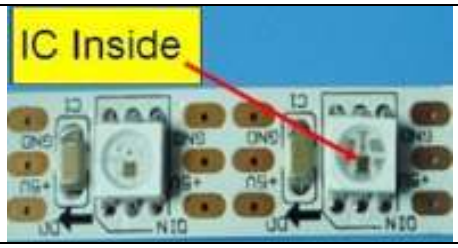
RGB strips

One can buy strips of colored leds. Several models exist. One color or many colors, programmable or not.
The best functionnality is when one control circuit is associated with every tri-color LED. WS2801, LPD8806 and LPD6806 are the frequently used circuits.
Let us talk about the most easy to implement, available from Boxtec (part 3xxxx), using the W2801 circuit,

Quick compare.
For a customer, space between Leds is an important parameter. Check also the number of leds controlled by the same integrated circuit, they will get the same color and a higher voltage will be required (12V if 3 leds per circuit).
WS2801 This 14-pin circuit includes three 8-bit PWM . Easy to program, as shown later. Strips using this circuit are sold by Boxtec and many other Arduino distributors.
LPD8806/LPD6806 are 16-pin Strips using this circuit are sold by Adafruit , AliExpress, etc.
The circuit includes three 5-bit PWM. One more bit is a control bit. No timing constraint for clock durations, but more pulses are required before and after the steam of 16-bit transfers for the leds.

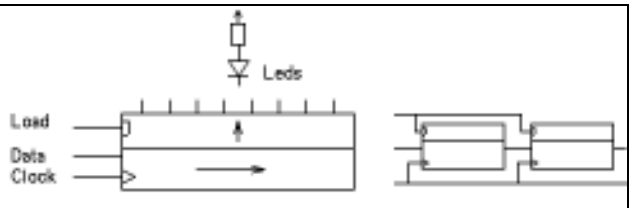
WS2811 is a 8-pin circuit. Data transfera are done on a single with different timings around 1 microsecond, impossible to program. Special hardware is required. The commercial advantage is high density strips are possible, with the WS2811 bonded inside the 5050 RGB led. But it is necessary to buy the controller and use its pre-programmed sequences.



Existing strips
Several strips have one RGB led every 32mm and one associated control circuit. Other strips have 3 leds for one circuit, and need 12V to power these groups of 3 leds of the same color.
We prefer one circuit per led, and this circuit is rather complex since it has three PWM channels top control the intensity of the three channels, that is three control registers of 8 bits, 256 levels for every color. This mean a 24 bit register, and all these registers are serially connected.
You should have heard about SPI, you may have played with a 595 shift register to control many meds using only three wires. The 2801 do something very similar, but with only 2 wires, one data line and one clock that "push" the data bits through this very long serial register.
But indeed, it is not like a single shift register, with the first bit that cross all the strip to reach the end. In every circuit, there are counters and switches. The first 24 bit word you send will stay in the first circuit of the chain, the second word will bypass the first circuit and stay in the second circuit, etc. You do not need to know the length of the strip when you fill it. The software for a strip of 20 can be used for a shorter strip. If the strip is longer, the additional leds will stay off. A shift register would behave differently. But how to know that all the data has been transferred? There is a timing constraint. Between the bits of a sequence, there is a maximum delay of 100 microseconds. At the end of the shift sequence, the delay must be greater than 1ms. All circuits have a timeout circuit, and if there is no clock for 1ms, the data in the 24 bit receive register is transferred to an additional parallel 24 bit register that controls the pwm circuitry. If you have a strip of 32 RGB leds, this means 768 bits to be shifted at more than 10 kHz (duration < 77 ms) and then you can pause as long as you want, but more than 1ms. In practice, shift clock is much faster, but there is no need to change the intensity of the Leds more frequently than 20 times per second. Adding meters of strip imply of course faster clock rate.

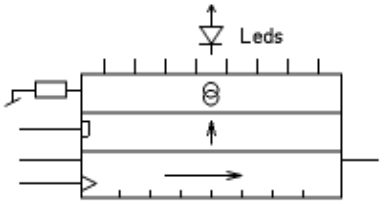
Refresh about serial transfers

Shift registers are used to convert a stream of bits synchronized by a clock into parallel data. A parallel register keeps the previous data during the shift and is updated at the end of the transfer..
This is the principle of the SPI transfer.
Registers can be cascaded.



The 74HC595

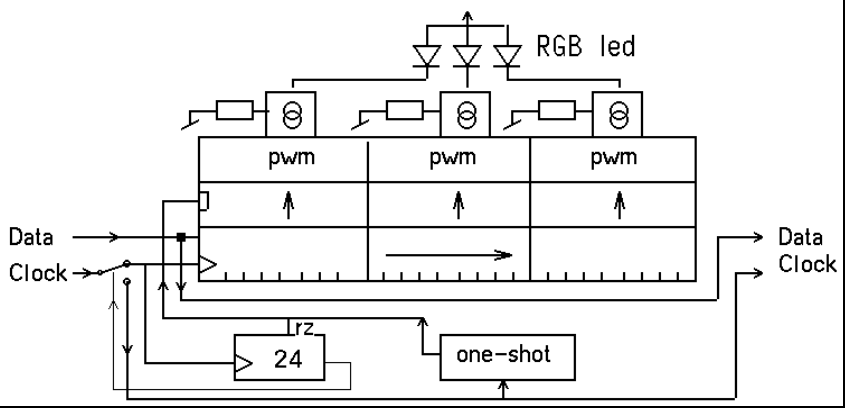
The frequent application is to control Leds. Several manufacturers propose shift registers with constant current sources on the output, A single resistor set the current, the same for all Leds. If RGB leds are connected, each color will get the same current, but the apparent intensity of the color may not be the same.
PWM can be done on every Leds, see .www.didel.com/diduino/CommandeLeds.pdf but it is limited to ~50 Leds and imply a continuous shift at maximum speed.



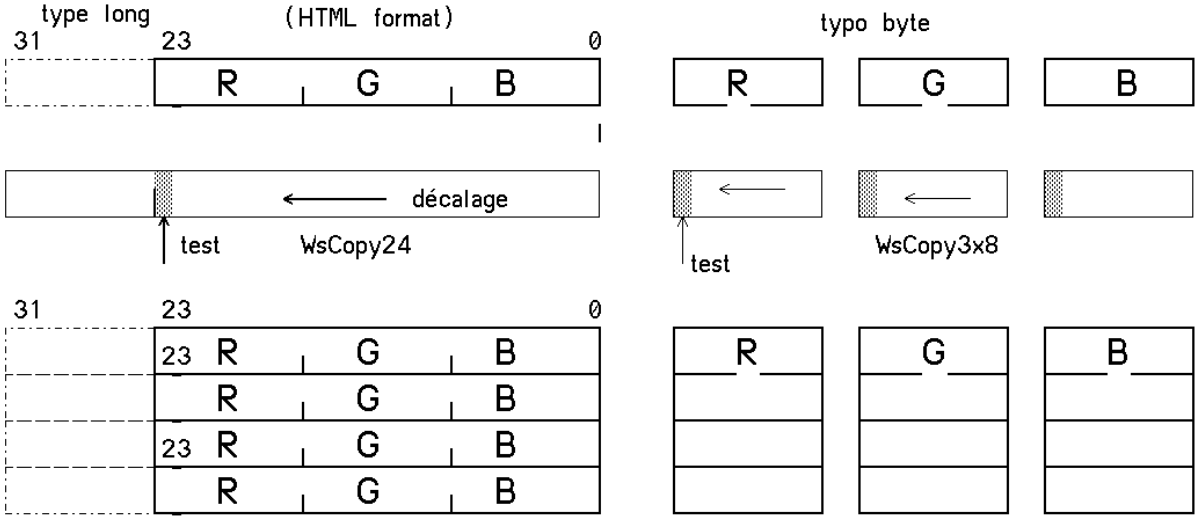
WS2801 - the circuit

In order to control the intensity of a Led, the simplest is to generate a PWM signal with several counters and registers. The WS2801 is a 24 bit shift register with three 8-bit pwm controllers. Three Leds are controlled by the circuit, usually a RGB SMD Led in a 505 package.

The block diagram show something new. Data is not shifted through the complete serially connected circuits. The first 24 bits stay in the first circuit. When 24 clocks are counted, a switch transfer the clock to the output. Since there is no Load line to transfer the shifted data to the PWM circuit, a one-shot circuit is activated when there is no clock for 10 ms.



Color coding



Color	Color HEX	Color RGB
	#000000	rgb(0,0,0)
	#FF0000	rgb(255,0,0)
	#00FF00	rgb(0,255,0)
	#0000FF	rgb(0,0,255)
	#FFFF00	rgb(255,255,0)
	#00FFFF	rgb(0,255,255)
	#FF00FF	rgb(255,0,255)
	#C0C0C0	rgb(192,192,192)
	#FFFFFF	rgb(255,255,255)