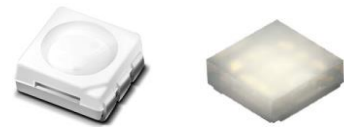


One-wire RGB+IC application note



1. Introduction:

Traditional RGB LED products, the design circuit needs to use multi-pin control to achieve the effect of mixing light, the circuit is complex and takes up a lot of space. One-wire RGB+IC products can connect multiple products in series with only one signal. Use DIN and DOUT two pins to connect multiple RGB+IC products in series. Change the DIN signal, each color has 8bit = 256-order PWM Duty control. This product is suitable for applications where space and wiring are limited.

2. Application Circuit:

Control circuit and RGB chip are integrated in a package of one-wire components.

Recommendation : Power pin should have Bypass capacitor, signal transmission path between components need to add RC Filter (reserved design), as shown in Figure 1.

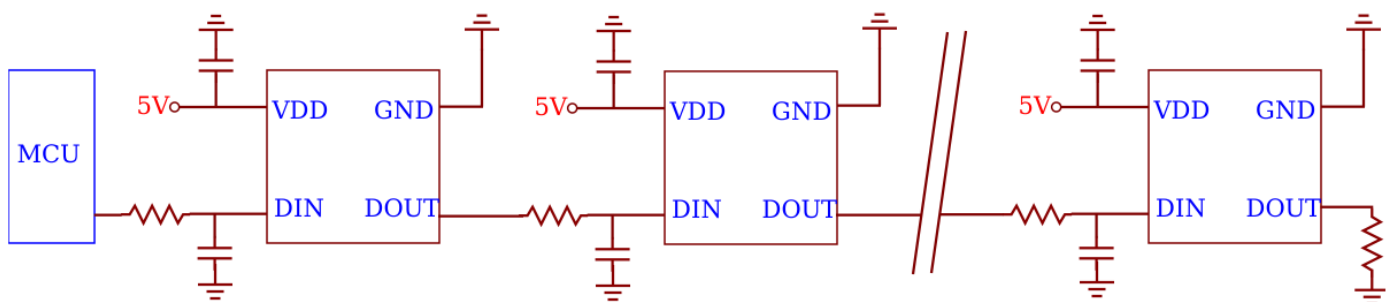


Figure 1. One-wire RGB+IC Application circuit

3. Control Signal

The RGB+IC product control signal is divided into two parts : Data transmission (24bit) and data update (50us reset time).The data transmission 24-bit is R=8-bit G=8-bit B=8-bit. The signal specifications of Logic 0, Logic 1 and Reset Time are shown in Figure 2. When transmitting data, it is sent by the MSB. The single transmission sequence is sent in the order of R (8bit)→G (8bit)→B (8bit)→Reset Time ,as shown in Figure 3. The RGB+IC product will display the new color setting after receiving the Reset Time.

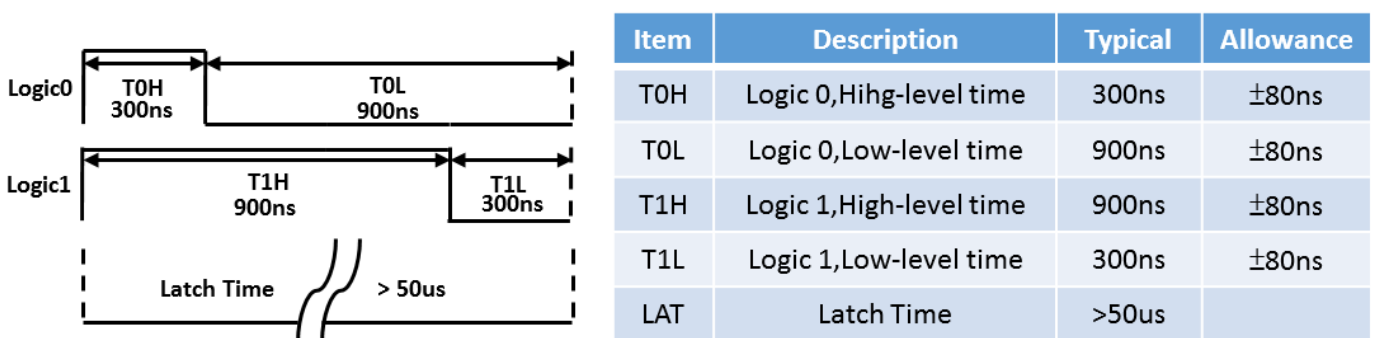


Figure 2. Control signal definition



Figure 3. Send sequence diagram

In the case of a series RGB + IC product, the signal is sent in the order of serial connection, and the data update (50us reset time) will be sent at the end. For example, three RGB+IC products are connected in series, and the transmission order is: LED1(24bit)→LED2 (24bit)→LED3(24bit)→Reset Time ,as shown in Figure 4.

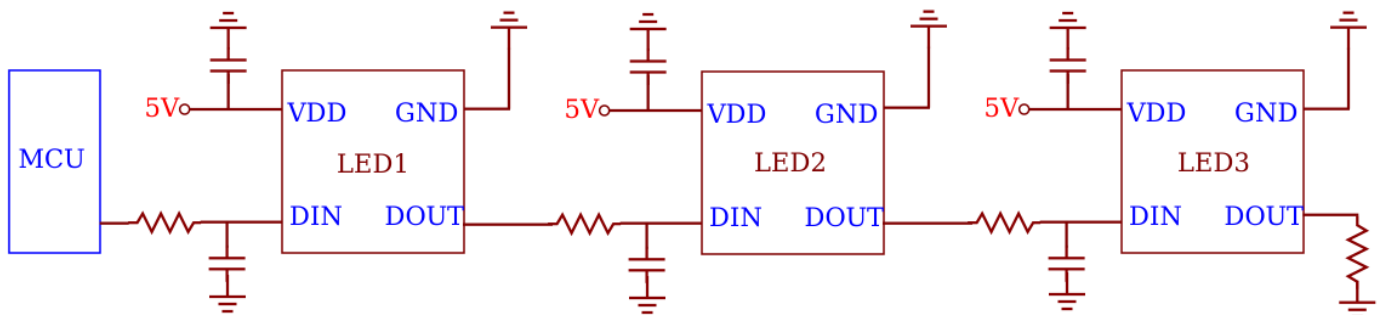


Figure 4. Serial sequence diagram

4. Pseudocode

The pseudocode is connected in series with three RGB+IC products. The LEDs will received the data in order and will be lit after the rest time. The LEDs colors are RED, GREEN, and BLUE in order. Using the SPI function of the MCU, the SPI CLK frequency is set at 3.3MHz, one SPI Bit is about 300ns, and the Logic 0 and Logic 1 of the RGB+IC correspond to 4 SPI Bits respectively, thereby completing the control waveform, and the pseudocode is as follows.

SPI Configuration :

- SPI Clock \cong 3.3MHz // One bit \cong 300ns
- Master Mode
- SPI Mode 0 (Clock Phase = 0 , Clock Polarity= 0)
- Data is latched on positive edge of bus clock
- When SPI idle , SPI Data and Clock is normal low

Data Structure :

```
Struct LED { R_reg , G_reg , B_reg} // One color register = 8bit  
Array LED Display {LED1 , LED2 , LED3} // Three LED Structure Array
```

Step 1. Assign Display Data

```
LED1 : R_reg = 0xFF G_reg = 0x00 B_reg = 0x00  
LED2 : R_reg = 0x00 G_reg = 0xFF B_reg = 0x00  
LED3 : R_reg = 0x00 G_reg = 0x00 B_reg = 0xFF
```

Step 2. Data Send Order

```
LED1 (R_reg→G_reg→B_reg)→LED2 (R_reg→G_reg→B_reg)→LED3 (R_reg→G_reg→B_reg)  
Repeat Step3 and Step4 to convert and send Data.
```

Step 3. LED Color_reg 8bit Convert to 32bit

```
// LED one bit data : (Logic 0)→Four bits data 1000 (300ns High + 900ns Low)  
// LED one bit data : (Logic 1)→Four bits data 1110 (900ns High + 300ns Low)  
// According to the above rules ,One color(8bit) will be converted to 32bit  
// Declaration 32bit variable (Prepared_Data)  
// Prepared_Data is used to save One Color_reg conversion results
```

```
// Prepared_Data and Color_reg Leftmost bit is MSB
Input : Color_reg (8bit Data)
OutPut : Prepared_Data(32bit Data)
for i ←7 to 0 do
  // i=7(MSB) , i=6(MSB-1)..... i=0(LSB)
  If Color_reg(i) = 1
    Prepared_Data (i*4+3)~(i*4) = 1110 (binary) //0xE(hex)
  else //Color_reg(i) = 0
    Prepared_Data (i*4+3)~(i*4) = 1000 (binary) //0x8(hex)
end for
//e.g. 0xA5 conversion results = 0xE8E88E8E
```

Step 4. Send Procedure

If SPI FIFO not full, Send Prepared_Data to SPI FIFO

Step 5. Send Latch Time

If all data send procedure is finished, Send Latch time (>50us low).

5. Actual Waveform

Logic 0 and Logic 1 waveform refer to the SPI setting of the pseudocode in the previous section, as shown in the following figure 5. 950ns High + 315ns Low is Logic 1, and 315ns High + 950ns Low is Logic 0.







Figure 5. Logic 0 and Logic 1 waveform

The waveform of a single LED is as shown in the following figure 6. When the data is not transmitted, Data remains low. After sequentially transmitting the R, G, and B data, and then maintaining > 50us Low level, the product will display updated colors.



Figure 6. Single LED waveform

6. Suggestion Part List

	Part No.	Package	Size(L*W*Hmm)	Luminous Intensity R/G/B (mcd)	IF (mA)	Viewing Angle
	19-C47/RSGHBHC-5V01/2T	SMD(Top)	1.8*1.8*0.65	70/180/40	5	120°
	61-236-ICRQHGRBYC-A05-ET-CS	SMD(Top)	5.42*5.0*1.6	185/590/148	5	120°
	12-23C/RSGHBHW-5V01/2C	SMD(Side)	3.0*1.0*1.5	47/93/38	5	130°
	C4516SDWN3S1-RGBC0120-2H	SMD(Side)	4.5*1.6*1.7	785/1960/495	20	120°

* In application, for 19-C47, 61-236, 12-23C the series LED (RGB+IC) quantities should be less than 100 to ensure the display performance.

This application manual provides customer design reference. If there are design changes, system performance may be degraded. If there are any problems in the design of the system, please contact Everlight for further technical support.