

به نام خدا

گزارش آزمایشگاه ریزپردازنده

محمد مهدی آقاجانی

علی اکبر بدری

عنوان : RGB LED WS2812

## ویژگی های الکتریکی

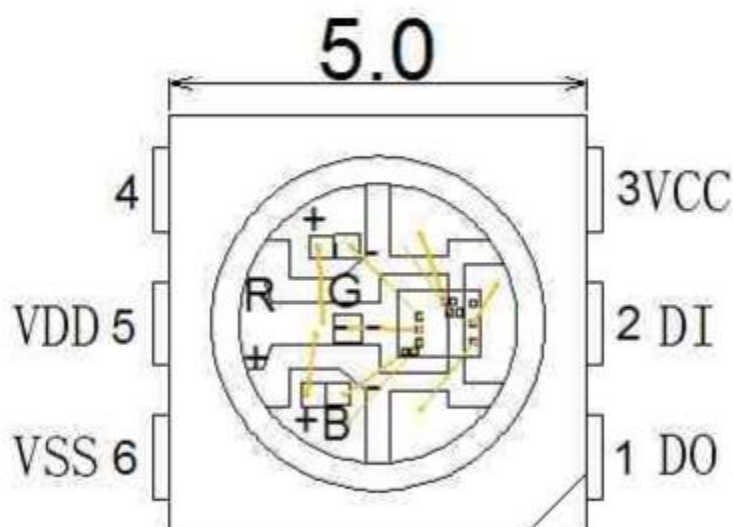
**Electrical Characteristics** ( $T_A = -20 \sim +70^\circ\text{C}$ ,  $V_{DD} = 4.5 \sim 5.5\text{V}$ ,  $V_{SS} = 0\text{V}$ , unless otherwise specified)

<b>Prameter</b>	<b>Smybol</b>	<b>conditions</b>	<b>Min</b>	<b>Tpy</b>	<b>Max</b>	<b>Unit</b>
Low voltage output current	$I_{OL}$	ROUT	—	18.5	—	mA
	$I_{dout}$	$V_O = 0.4\text{V}$ , $D_{OUT}$	10	—	—	mA
Input current	$I_I$	$V_I = V_{DD}/V_{SS}$	—	—	$\pm 1$	$\mu\text{A}$
Input voltage level	$V_{IH}$	$D_{IN}$ , SET	$0.7V_{DD}$	—	—	V
	$V_{IL}$	$D_{IN}$ , SET	—	—	$0.3 V_{DD}$	V
Hysteresis voltage	$V_H$	$D_{IN}$ , SET	—	0.35	—	V

<b>Prameter</b>	<b>Symbol</b>	<b>Ratings</b>	<b>Unit</b>
Power supply voltage	$V_{CC}$	$+6.0 \sim +7.0$	V
Power supply voltage	$V_{DD}$	$+6.0 \sim +7.0$	V
Input voltage	$V_I$	$-0.5 \sim V_{DD} + 0.5$	V
Operation junction temperature	$T_{opt}$	$-25 \sim +80$	$^\circ\text{C}$
Storage temperature range	$T_{stg}$	$-55 \sim +150$	$^\circ\text{C}$

## عملکرد های پایه

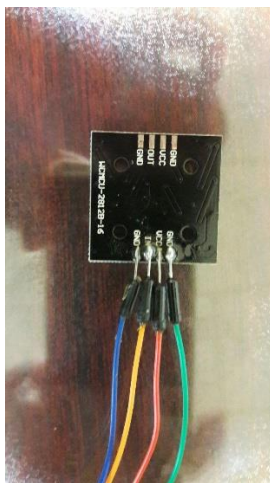
در ابتدا تصویر زیر را در نظر میگیریم :



عملکرد پایه های آن به صورت زیر است :

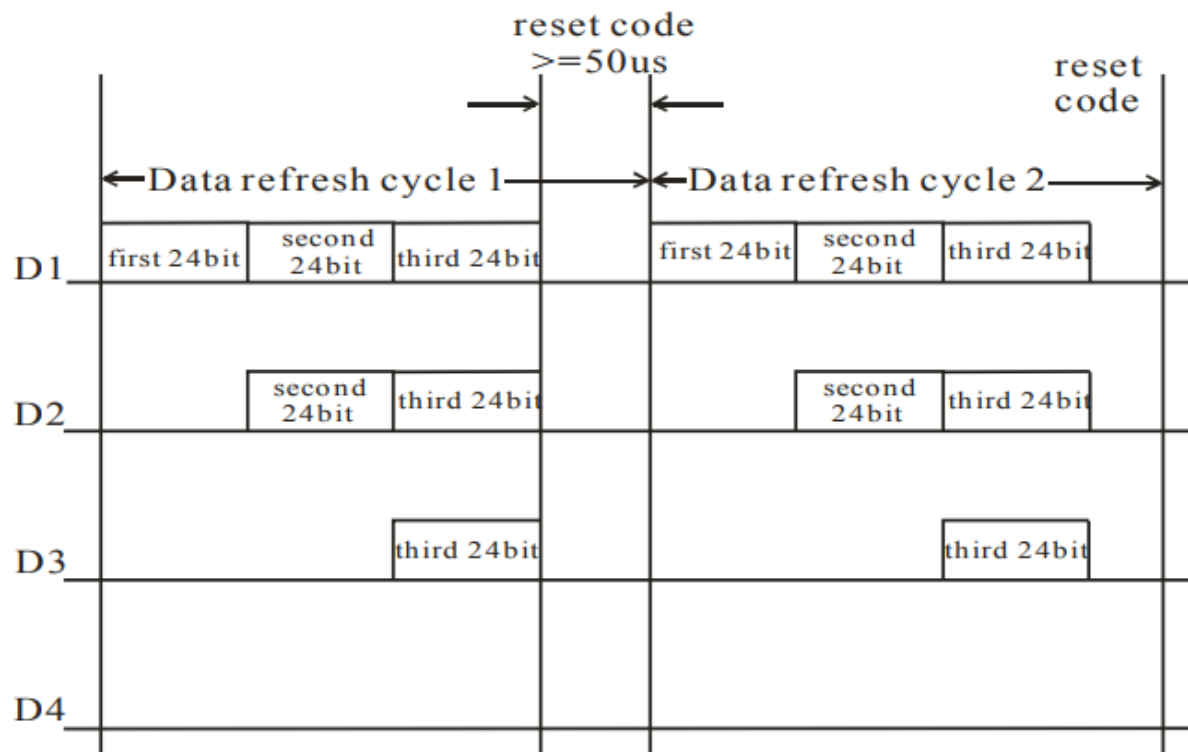
NO.	Symbol	Function description
1	DOUT	Control data signal output
2	DIN	Control data signal input
3	VCC	Power supply control circuit
4	NC	
5	VDD	Power supply LED
6	VSS	Ground

در قطعه مورد نظر پورت DIN برای ارسال داده ها برای ست کردن رنگ مورد نظر بر روی قطعه می باشد همچنین پورت DOUT برای سریال بستن قطعات می باشد. البته در قطعه ای که بر روی آن آزمایش انجام شد دو پورت GND وجود داشت که یکی از آنها بلا استفاده ماند.



در تصویر بالا سیم سبز رنگ بدون استفاده باقی ماند . همچنین سیم آبی برای GND و سیم قرمز برای ولتاژ تغذیه و سیم نارنجی به عنوان ورودی قطعه استفاده شدند.

## نحوه ارسال بیت ها



سیگنال D1 توسط میکروکنترلر ارسال میشود ولی سیگنال های D2 , D3, D4 از طریق تقویت پیکسل های درونی فرستاده میشود.

الگوی ارسال به این صورت است که ابتدا بیت های مربوط به رنگ سبز ارسال میشود و بعد قرمز و بعد آبی :

G7	G6	G5	G4	G3	G2	G1	G0	R7	R6	R5	R4	R3	R2	R1	R0	B7	B6	B5	B4	B3	B2	B1	B0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

## زمان انتقال داده ها

طبق دیتاشیت باید زمان های ولتاژ بالا و پایین code 1 , code 0 از رابطه زیر پیروی نمایند :

$$T_H + T_L = 1.25\mu s \pm 600ns$$

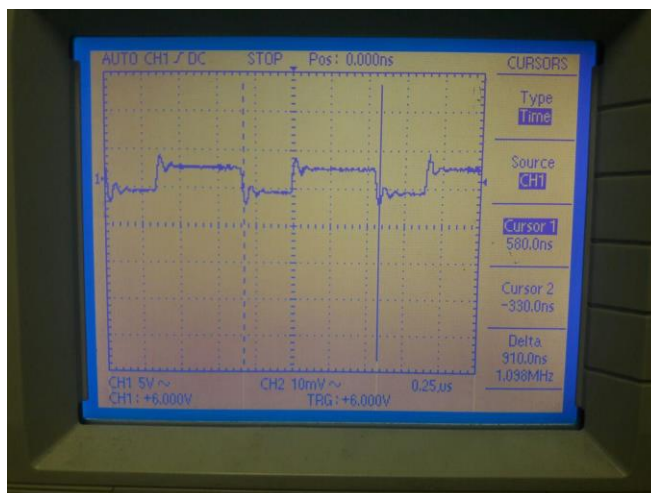
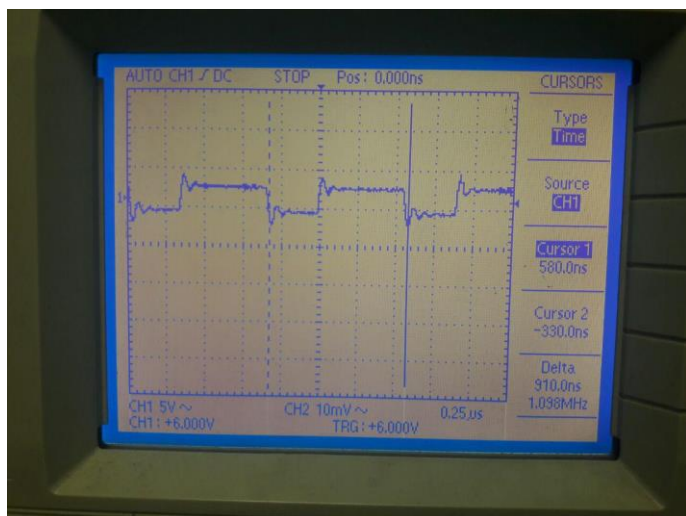
طبق آزمایش انجام شده و تصاویر زیر مقادیر به دست آمده به صورت زیر می باشد :

$$T_{0H} + T_{0L} = 0.8 + 0.35 = 1.15\mu s$$

$$T_{1H} + T_{1L} = 0.8 + 0.7 = 1.5\mu s$$

همانطور که ملاحظه میشود اعداد به دست آمده در بازه معرفی شده در دیتاشیت می باشند . در زیر هم تصاویر مربوطه که

بر روی اوسیلوسکوپ نمایش داده شده را آورده ایم :



## کد پروژه

```
#include "mbed.h"
#include "PololuLedStrip.h"

PololuLedStrip ledStrip(PTD7);
#define LED_COUNT 16
rgb_color colors[LED_COUNT];

InterruptIn sw1(PTD5);
InterruptIn sw2(PTD2);

void pattern1(){
    for(int i = 0; i < LED_COUNT; i++)    {
        if( i%2 == 0 ){
            colors[i] = (rgb_color){255, 0, 0}; //red
        }else{
            colors[i] = (rgb_color){43, 0, 255}; //blue
        }
    }
    ledStrip.write(colors, LED_COUNT);
}

void pattern2(){
    for(int i = 0; i < LED_COUNT; i++){
        if( i%2 == 0 ){
            colors[i] = (rgb_color){0, 255, 85}; //green
        }else{

```

```

        colors[i] = (rgb_color){255, 247, 0}; //yellow
    }
}
ledStrip.write(colors, LED_COUNT);
}

int main () {
    sw1.rise(&pattern1);
    sw2.rise(&pattern2);
    for(int i = 0; i < LED_COUNT; i++)
    {
        colors[i] = (rgb_color){255, 0, 247}; //purple
    }
    ledStrip.write(colors, LED_COUNT);
}

```

در این کد از کتابخانه PoluluLedStrip استفاده شده است. این کتابخانه را میتوانید از آدرس زیر دانلود نمایید.

<https://os.mbed.com/users/DavidEGrayson/code/PoluluLedStrip/>

این کد برای برد ARM KL25Z تست شده است و این برد برای سیستم های نهفته به کار میرود.

تابع write در واقع بر اساس الگوی داده شده (رنگ های RGB تولید شده) قطعه مورد نظر را برنامه ریزی میکند تا آن قطعه رنگ ها را به درستی نشان دهد.

قطعه مورد نظر به پورت D و پین ۷ آن وصل شده است. همچنین سوییچ های SW1, SW2 به ترتیب به پورت D و پین شماره ۲ و پورت D و پین شماره ۵ وصل شده است. این پورت ها به طور وقفه برنامه ریزی شده اند و در حالت rising edge قرار دارند و با فشردن هر یک از آن ها روتین وقفه pattern مربوطه اجرا میگردد که آن روتین الگو مورد نظر را بر روی آرایه color ست میکند و بعد هم توسط تابع write کتابخانه بالا بر روی قطعه نوشته میشود