

通宵敲代码



论坛 ▾

苏子麦田 |   QQ帐号绑定 | 我的 ▾ | 设置 |  提醒  | 退出

打卡签到 | 积分: 0 | 用户组: 一粒金砂 (初级)



帖子 ▼

高级搜索

发帖 ▶

论坛 > 创意与实践 > DIY/开源硬件专区 > WS2812灯珠的STM32驱动方式(三)——DMA+SPI

TA的帖子

TA的资源

版主



 发消息 加好友

发帖 ▾

[回复](#)

◀ 返回列表

查看: 1612 | 回复: 18

通宵敲代码



1171

TA的帖子

2

TA的资源

版主



 发消息 加好友

[原创] WS2812灯珠的STM32驱动方式(三)——DMA+SPI [\[复制链接\]](#)



发表于 2018-5-7 00:06:10 | 只看该作者 | 只看大图 | 倒序浏览 | 阅读模式

楼主 电梯直达



接上节我们介绍了一种STM32控制WS2812最常用，也是最用一想到的方法，就是用PWM来模式控制信号，并通过DMA传输数组数据，来保证信号传输的连续性。这篇我们来介绍一个不常用的方法——用SPI单总线控制

估计好多不熟悉STM32的小伙伴都没通过这种通信方式吧，其实我也不熟，只是知道这种模式从未用过，今天也算学习了。下面是STM32F1的Reference Manual里给的一段介绍，了解一下。

23.3.8 关闭SPI

当通讯结束，可以通过关闭SPI模块来终止通讯。清除SPE位即可关闭SPI。

在某些配置下，如果再传输还未完成时，就关闭SPI模块并进入停机模式，则可能导致当前的传输被破坏，而且BSY标志也变得不可信。

为了避免发生这种情况，关闭SPI模块时，建议按照下述步骤操作：

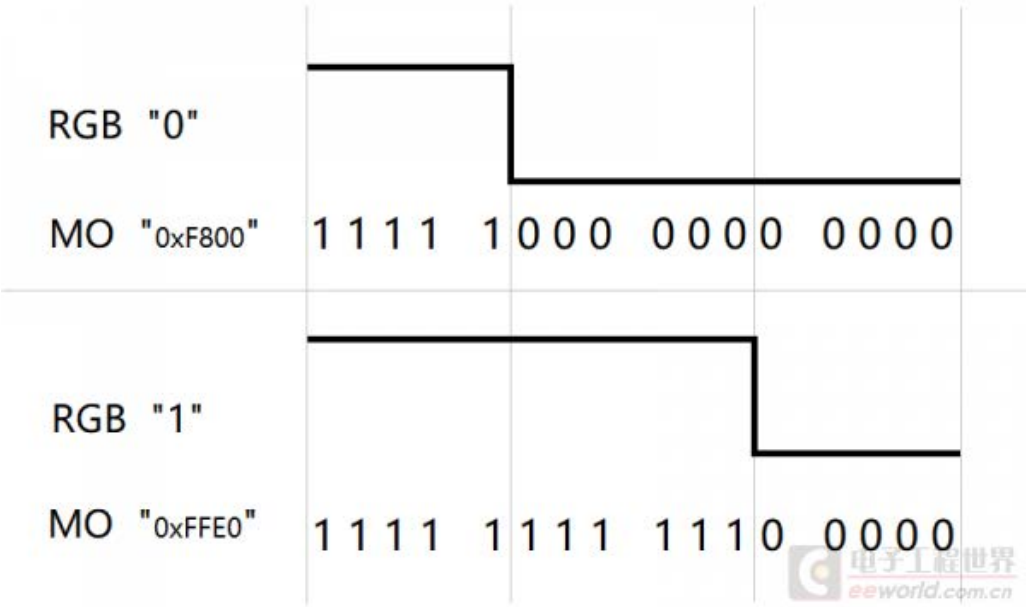
在主或从模式下的单向只发送模式(BIDIMODE=0, RXONLY=0)或双向的发送模式(BIDIMODE=1, BIDIOE=1)

在SPI_DR寄存器中写入最后一个数据后：

1. 等待TXE=1；
2. 等待BSY=0；
3. 关闭SPI(SPE=0)，最后进入停机模式(或关闭该模块的时钟)。



其实说白了，SPI不是有一个MOSI引脚吗，
我们就是用MO功能，通过发送一个字节或半字的数据，
来模拟我们的LED控制信号0或1，看下图。



是不是有恍然大悟的感觉，本来就很简单嘛。
可仔细看来这是不是有点得不偿失啊，
用16位数据去模拟1位数据，是不是感觉太浪费资源了！！
还在我们的SPI通信速率足够高，完全可以应付的来。



此帖出自[DIY/开源硬件专区论坛](#)

快速回复

评分



查看全部评分 lb8820265

最新活动

- 【泰有聊】第二期：泰克CEO为你解读低...
- TE Connectivity 小有乾坤蕴藏无限可能
- ADI 系统方案精选【用于多路抽头输出隔...
- 我是大侦探之：寻找遗落的TI电源拼图
- EEWORLD 示波器问卷有奖大调查

推荐资源

- 汽车电子嵌入式系统简介
- 运算放大器应用技术手册
- CAN总线技术在设备故障诊断系统中的...
- 开关电源变换：设计和分析
- 电动伺服电动机方程和时间常数

相关帖子

- WS2812灯珠的STM32驱动方式(二)——DM...
- WS2812灯珠的STM32驱动方式(一)——了...
- 求购AM335X工控核心板
- 串口通信最远传输距离能达到多少
- 可视对讲系统~~小区改造简单安装~~~



★ 收藏 2

♥ 干货

👍 评分

❤ 分享

DIY论坛 开源硬件论坛

回复

使用道具 ▼ 举报

通宵敲代码

楼主 | 发表于 2018-5-7 00:06:25 | 只看该作者

沙发

本帖最后由 通宵敲代码 于 2018-5-7 00:28 编辑



1171 TA的帖子 | 2 TA的资源

版主



发消息 | 加好友

原理介绍完了，我们就直接来看程序了

时钟就不用说了，还是能跑多快就跑多块咯，
重点来看看我们SPI的单总线模式是如何配置的，
当然第一个要配置的自然是我们的信号输出引脚，
SIM32F1系列SPI1的MOSI引脚是PA7，

```
29
30     GPIO_InitStructure.GPIO_Pin = GPIO_Pin_7;
31     GPIO_InitStructure.GPIO_Mode = GPIO_Mode_AF_PP;
32     GPIO_InitStructure.GPIO_Speed = GPIO_Speed_50MHz;
33     GPIO_Init(GPIOA, &GPIO_InitStructure);
34
```

注意要配置成复用输出模式，最大驱动频率50MHz，
接下来就是我们的主角SPI1的配置了。

```
40     SPI_I2S_DeInit(SPI1);
41
42     SPI_InitStructure.SPI_Direction = SPI_Direction_1Line_Tx;
43     SPI_InitStructure.SPI_Mode = SPI_Mode_Master;
44     SPI_InitStructure.SPI_DataSize = SPI_DataSize_16b;
45     SPI_InitStructure.SPI_CPOL = SPI_CPOL_Low;
46     SPI_InitStructure.SPI_CPHA = SPI_CPHA_1Edge;
47     SPI_InitStructure.SPI_NSS = SPI_NSS_Soft;
48     SPI_InitStructure.SPI_BaudRatePrescaler = SPI_BaudRatePrescaler_8;
49     SPI_InitStructure.SPI_FirstBit = SPI_FirstBit_MSB;
50     SPI_InitStructure.SPI_CRCPolynomial = 7;
51     SPI_Init(SPI1, &SPI_InitStructure);
52
53     SPI_I2S_DMACmd(SPI1, SPI_I2S_DMAREQ_Tx, ENABLE);
54
55     SPI_Cmd(SPI1, ENABLE);
56
57     for (i = 0; i < 1024; i++)
58     {PixelBuffer[i] = 0xAAAA;}
59     PixelPointer = 0;
60 }
```

前两句就可以看到，我们将SPI配置成了主模式下的单线发送模式，
采用了16微数据帧格式，时钟引脚悬空未用，片选引脚未用。
然后就是我们的DMA数据传输的配置，
通过我们的Reference Manual可以查到SPI的TX使用的DMA1_CH3通道。

电子工程世界
= MEM -> DR */
eeeworld.com.cn

eeeworld.com.cn

 电子工程世界
eeworld.com.cn

转化成我们即将使用SPI发送出去的16位帧信号。

注意，此处帧的格式是根据LED所需的信号0和1定义好的，

```
78 void LED_SPI_SendBits(uint8_t bits)
79 {
80     int zero = 0xF800; //1111 1000 0000 0000
81     int one = 0xFFE0;  //1111 1111 1110 0000
82     int i = 0x00;
83
84     for (i = 0x80; i >= 0x01; i >>= 1)
85     {
86         LED_SPI_WriteByte((bits & i) ? one : zero);
87     }
88 }
```



然后我们要将转化出来的16位帧数据一次存放打我们给SPI准备的数组中，也就是我们通过DMA功能给SPI的TX引脚传送数据的数组，注意对应哦。

```
62 void LED_SPI_WriteByte(uint16_t Data)
63 {
64     /* Wait until the transmit buffer is empty */
65     /*
66     while(SPI_I2S_GetFlagStatus(SPI1, SPI_I2S_FLAG_TXE) == RESET)
67     {
68     }
69     */
70
71     PixelBuffer[PixelPointer] = Data;
72     PixelPointer++;
73
74     /* Send the byte */
75     /* SPI_I2S_SendData16(SPI1, Data); */
76 }
```



看看是不是跟DMA初始化的时候配置的一样呢。


```

3  uint16_t PixelBuffer[1024] = {0};
4  uint16_t PixelPointer = 0;
5
6  void LED_SPI_LowLevel_Init(void)
7  {
8      uint16_t i = 0;
9
10     GPIO_InitTypeDef  GPIO_InitStructure;
11     SPI_InitTypeDef    SPI_InitStructure;
12     DMA_InitTypeDef    DMA_InitStructure;
13
14     RCC_AHBPeriphClockCmd(RCC_AHBPeriph_DMA1, ENABLE);
15     RCC_APB2PeriphClockCmd(RCC_APB2Periph_GPIOA | RCC_APB2Periph_SPI1, ENABLE);
16
17     DMA_DeInit(DMA1_Channel3);
18     DMA_InitStructure.DMA_BufferSize = 0;
19     DMA_InitStructure.DMA_PeripheralBaseAddr = (uint32_t) &(SPI1->DR);
20     DMA_InitStructure.DMA_MemoryBaseAddr = (uint32_t)PixelBuffer;
21     DMA_InitStructure.DMA_DIR = DMA_DIR_PeripheralDST;
22     DMA_InitStructure.DMA_Priority = DMA_Priority_Low;
23     DMA_InitStructure.DMA_PeripheralInc = DMA_PeripheralInc_Disable;
24     DMA_InitStructure.DMA_MemoryInc = DMA_MemoryInc_Enable;
25     DMA_InitStructure.DMA_PeripheralDataSize = DMA_PeripheralDataSize_HalfWord;

```

接下来就是如何把已经转化好的16位帧数据发送出去了。

具体函数如下，记得要使用DMA功能传送数据哦。

```

115 void LED_SPI_Update(unsigned long buffer[], uint32_t length)
116 {
117     uint8_t i = 0;
118     uint8_t m = 0;
119     if(DMA_GetCurrDataCounter(DMA1_Channel3) == 0)
120     {
121         for (i = 0; i < length; i++)
122             {LED_SPI_SendPixel(buffer[i]);}
123         for (i = 0; i < 20; i++)
124             {LED_SPI_WriteByte(0x00);}
125         PixelPointer = 0;
126
127         // DMA_Cmd(DMA1_Channel3, DISABLE);
128         // DMA_ClearFlag(DMA1_FLAG_TC3);
129         DMA_SetCurrDataCounter(DMA1_Channel3, 24*length+20);
130         DMA_Cmd(DMA1_Channel3, ENABLE);
131         while(!DMA_GetFlagStatus(DMA1_FLAG_TC3)) ; // wait until transfer c
132         DMA_Cmd(DMA1_Channel3, DISABLE); // disable DMA channel 3
133         DMA_ClearFlag(DMA1_FLAG_TC3); // clear DMA1 Channel 3 transf
134     }
135 }
136

```

0 点评



naga568

謝謝分享。 详情 回复 发表于 2018-5-7 07:52

开源代码共享 开源设计共享 开源硬件

回复

评分 举报

楼主 | 发表于 2018-5-7 00:06:37 | 只看该作者

板凳

本帖最后由 通宵敲代码 于 2018-5-7 00:33 编辑

我们在主函数中，通过数据发送函数，将RGB信号传送到LED，同时，我们简单通过一个循环移位函数，实现了LED的三色流水灯。

```
38 while(1)
39 {
40     WS_SetAll();
41     LED_SPI_Update(WsDat,18);
```

```
96 {
97     temp=WsDat[0];
98     for(j=0;j<18;j++)
99     {
100         WsDat[j]=WsDat[j+1];
101     }
102     WsDat[17]=temp;
103 }
104 }
105 }
```


附上程序



[STM32F1_DMA_SPI_16B_WS2813E_2018_05_06-Bingo.zip](#) (6.94 MB, 下载次数: 6492)

回复

评分 举报

naga568

发表于 2018-5-7 07:52:32 | 只看该作者

4楼



368
TA的帖子

0
TA的资源

纯净的硅（初级）



发消息 加好友



[通宵敲代码](#) 发表于 2018-5-7 00:06
原理介绍完了，我们就直接来看程序了

时钟就不用说了，还是能跑多快就跑多块咯，
重点来看看我们SPI ...



謝謝分享。

回复

评分 举报

bioger

发表于 2018-5-7 08:53:27 | 只看该作者

5楼



294
TA的帖子

0
TA的资源

点评

一粒金砂（高级）

发消息

加好友

通宵敲代码



1171

TA的帖子

2

TA的资源

版主

发消息

加好友

bioger



294

TA的帖子


0

TA的资源

一粒金砂（高级）

发消息

加好友

 通宵敲代码

附件里有，可以下载 [详情](#) [回复](#) 发表于 2018-5-7 08:55

回复

评分 举报

 楼主 | 发表于 2018-5-7 08:55:32 | 只看该作者

6楼

“

bioger 发表于 2018-5-7 08:53
能共享一下这个程序的代码吗 谢谢啊!

”

附件里有，可以下载

点评

 bioger

谢谢 看到了 [详情](#) [回复](#) 发表于 2018-5-7 08:56

回复

评分 举报

 发表于 2018-5-7 08:56:35 | 只看该作者

7楼







“

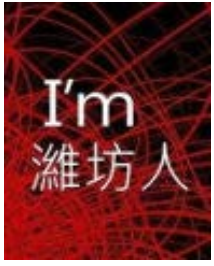
通宵敲代码 发表于 2018-5-7 08:55
附件里有，可以下载

”

谢谢 看到了

http://bbs.eeworld.com.cn/thread-643895-1-1.html[2018/9/25 15:36:20]

	<div>回复</div> <div>评分 举报</div>
<div><div>xsc</div><div></div><div><div>253</div><div>TA的帖子</div></div><div><div>0</div><div>TA的资源</div></div><div><div>一粒金砂（中级）</div><div><div></div></div><div><div></div></div><div></div></div><div><div>发消息</div><div>加好友</div></div></div>	<div><div> 发表于 2018-5-7 09:00:30 只看该作者</div><div>8楼</div></div> <div><div>详细，很好的教程!!!</div></div> <div><div>回复</div><div>评分 举报</div></div>
<div><div>bioger</div><div></div><div><div>294</div><div>TA的帖子</div></div><div><div>0</div><div>TA的资源</div></div><div><div>一粒金砂（高级）</div><div><div></div></div><div><div></div></div><div></div></div><div><div>发消息</div><div>加好友</div></div></div>	<div><div> 发表于 2018-5-8 10:30:59 只看该作者</div><div>9楼</div></div> <div><div>for (i = 0; i < 20; i++) { LED_SPI_WriteByte(0x00); }</div><div>请问这句话是什么用意呢?</div><div><div>点评</div></div><div><div> 通宵敲代码 用来填充数组的空闲数据，防止发出去一堆乱数据。 详情 回复 发表于 2018-5-9 09:56</div></div><div><div>回复</div><div>评分 举报</div></div></div>
<div>通宵敲代码</div>	<div><div> 楼主 发表于 2018-5-9 09:56:07 只看该作者</div><div>10楼</div></div>



1171
TA的帖子

2
TA的资源

版主



bioger



294
TA的帖子

0
TA的资源

一粒金砂 (高级)



bioger 发表于 2018-5-8 10:30

```
for (i = 0; i < 20; i++)  
{  
    LED_SPI_WriteByte(0x00);  
}
```

请问这句 ...



用来填充数组的空闲数据，
防止发出去一堆乱数据。



点评



bioger

那从数量上计算也是884，不是数组定义时候的1024了，填充20是什么道理呢？ [详情](#) [回复](#) 发表于 2018-5-9 10:17

回复

评分 举报



发表于 2018-5-9 10:17:07 | 只看该作者

11楼



通宵敲代码 发表于 2018-5-9 09:56

用来填充数组的空闲数据，
防止发出去一堆乱数据。



那从数量上计算也是884，不是数组定义时候的1024了，填充20是什么道理呢？

点评

发消息 加好友

通宵敲代码



1171
TA的帖子

2
TA的资源

版主



发消息 加好友

bioger



294
TA的帖子

0
TA的资源

一粒金砂 (高级)



发消息 加好友




通宵敲代码

不必纠结细节，程序能用就行 (之前调程序加的冗余代码，完全可以删掉) [详情](#) [回复](#) 发表于 2018-5-9 10:26

回复

评分 举报

 楼主 | 发表于 2018-5-9 10:26:05 | 只看该作者

12楼



[bioger](#) 发表于 2018-5-9 10:17

那从数量上计算也是884，不是数组定义时候的1024了，填充20是什么道理呢？



不必纠结细节，程序能用就行
(之前调程序加的冗余代码，完全可以删掉)



点评




bioger

好吧 [详情](#) [回复](#) 发表于 2018-5-9 13:09

回复

评分 举报

 发表于 2018-5-9 13:09:25 | 只看该作者

13楼



[通宵敲代码](#) 发表于 2018-5-9 10:26

不必纠结细节，程序能用就行
(之前调程序加的冗余代码，完全可以删掉)
:L ...



好吧

jishuaihu



3552

TA的帖子

6

TA的资源

裸片初长成（初级）



发消息

加好友

通宵敲代码



回复

评分 举报

 发表于 2018-5-10 10:48:45

只看该作者

14楼

本帖最后由 jishuaihu 于 2018-5-10 11:00 编辑

我也这么用过，也用过DMA+PWM的方式驱动。缺点你上面也说了。用8个字节的数据去模拟一个字节，太浪费空间了。我打算改成另外一种方式去做，直接用PWM控制，在每个周期内根据需要发送的数据的bit去调节PWM的占空。这种每个周期调节占空比的方法在其他应用上也挺常用的，比如数字电源。这样只需要处理好两个索引，即LED的编号和每个字节中的bit位。比起上述方法应该能省不少内存。

不过最近没时间弄，也不着急用，就一直搁下了。

有兴趣的可以试一下

刚才看到楼主在以前的帖子里介绍过我上面说的方法了。楼主提到要不断地改变定时器CCR的数值，觉得这个很麻烦，可能影响信号的稳定性。

这一点楼主可以放心，稳定性没有问题，数字电源就是这么用的，我以前的项目中用PWM控制采样频率，也是根据捕获的频率实时改变采样频率的。在定时器的溢出中断里去改变下一个周期的占空比，消耗的CPU资源也不是很多，当然可能比DMA方式多一些，但是省了不少内存，可以去控制更多是LED。

点评



通宵敲代码

嗯，用外设实现信号输出， 相比直接用I/O口能节省很多资源， 再加上DMA直接做好数据映射， 基本可以解放MCU了。 [详情](#) [回复](#) 发表于 2018-5-11 09:03

回复

评分 举报

楼主 | 发表于 2018-5-11 09:03:50

只看该作者

15楼

“

jishuaihu 发表于 2018-5-10 10:48

我也这么用过，也用过DMA+PWM的方式驱动。缺点你上面也说了。用8个字节的数据去模拟一个字节，太浪费空间了 ...

”

嗯，用外设实现信号输出，

相比直接用I/O口能节省很多资源，

再加上DMA直接做好数据映射，

http://bbs.eeworld.com.cn/thread-643895-1-1.html[2018/9/25 15:36:20]

1171

TA的帖子

2

TA的资源

版主







 发消息

 加好友

基本可以解放MCU了。



 回复

[评分](#) [举报](#)

formooting0807



1

TA的帖子

0

TA的资源

一粒金砂（初级）







 发消息

 加好友

 发表于 2018-5-11 21:07:53 | 只看该作者

16楼

感谢分享 

 回复

[评分](#) [举报](#)

DSCX05



32

TA的帖子


0

TA的资源

一粒金砂（中级）








 发表于 2018-8-23 15:20:59 | 只看该作者

17楼

好，尝试尝试

<div><div>发消息</div><div>加好友</div></div>		<div>回复</div> <div>评分 举报</div>	
<div>openpv</div>		<div>发表于 5 天前 只看该作者</div> <div>18楼</div>	
<div><div></div><div><div>1</div><div>TA的帖子</div></div><div><div>0</div><div>TA的资源</div></div><div><div>一粒金砂 (初级)</div><div><div></div></div><div>☆</div></div><div><div>发消息</div><div>加好友</div></div></div>		<div> 谢谢分享!!!</div>	
<div>发消息</div> <div>加好友</div>		<div>回复</div> <div>评分 举报</div>	
<div>zhuyebb</div>		<div>发表于 5 天前 只看该作者</div> <div>19楼</div>	
<div><div></div><div><div>43</div><div>TA的帖子</div></div><div><div>0</div><div>TA的资源</div></div><div><div>一粒金砂 (中级)</div><div><div></div></div><div>🌙</div></div><div><div>发消息</div><div>加好友</div></div></div>		<div>好，尝试尝试</div>	
<div>发消息</div> <div>加好友</div>		<div>回复</div> <div>评分 举报</div>	

发帖 ▾

[回复](#)[◀ 返回列表](#)

发表回复

回帖后跳转到最后一页

本版积分规则



扫码关注

EEWORLD 官方微信



扫码关注

EE福利 唾手可得

Powered by **EEWORLD** 电子工程世界

© 2018 <http://bbs.eeworld.com.cn/>

[举报](#) | [小黑屋](#) | [手机版](#) | [Archiver](#) | 电子工程世界 (京ICP证 060456) 

GMT+8, 2018-9-25 15:34 , Processed in 0.498149 second(s), 25 queries , Gzip On, Redis On.