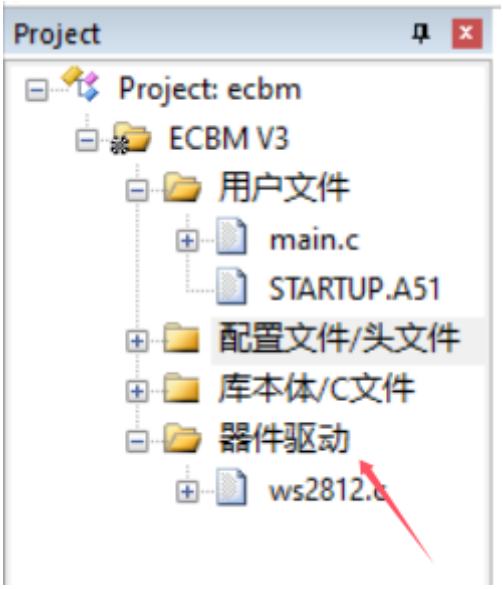
WS2812库

本库是以驱动WS2812B芯片为主的函数库，同时也兼容那些类似WS2812通信协议的芯片。

本库隶属于device，不是单片机本身自带的外设，因此需要手动将ws2812.c添加到工程文件中：



推荐如图所示操作，在keil的左侧栏双击【器件驱动】，然后添加ws2812.c进来。然后在main.c里添加头文件#include "ws2812.h"即可。

另外，本库是在内置RC的条件下测试通过的，而每一个单片机的RC都多多少少有制造误差，同时也存在温飘。因此如果颜色不对，可以在ws2812.c里自行微调一下WS2812_DELAY_T1H、WS2812_DELAY_T1L、WS2812_DELAY_T0H和WS2812_DELAY_T0L这四个宏定义。如果不能通信，则先去ecbm_core.h检查运行频率是不是实际工作的频率，再回来调整这四个宏定义。

有示波器或者逻辑分析仪最好，按下图调整时序：

T0H	0 码， 高电平时间	220ns~380ns
T1H	1 码， 高电平时间	580ns~1μs
T0L	0 码， 低电平时间	580ns~1μs
T1L	1 码， 低电平时间	580ns~1μs

API

ws2812_init

函数原型：void ws2812_init(void);

描述

WS2812的初始化函数。

输入

无

输出

无

返回值

无

调用例程

```
void main(void){  //main函数，必须的。
    system_init();//系统初始化函数，也是必须的。
    ws2812_init();//初始化ws2812。
    while(1){

    }
}
```

注意事项

1. ws2812对时序十分敏感，请确保在ecbm_core.h中选择了正确的频率。
2. 11.0592M以下无法使用ws2812库。

ws2812_send

函数原型：void ws2812_send(u8 dat[],u16 len);

描述

WS2812的发送函数。

输入

- dat：需要发送的数据，以GRB三色数据为单元，每3个数据为一组。
- len：dat数据的长度，肯定是3的倍数。

输出

无

返回值

无

调用例程

```
#include "ecbm_core.h"  //加载库函数的头文件。
#include "ws2812.h"
u8 test[]={
0xFF,0x00,0x00,//第一个灯，绿色。
0x00,0xFF,0x00,//第二个灯，红色。
0x00,0x00,0xFF,//第三个灯，蓝色。
};
void main(void){  //main函数，必须的。
```

```
    system_init();//系统初始化函数，也是必须的。
    ws2812_init();//初始化ws2812。
    ws2812_send(test,9);//发送数据到ws2812。
    while(1){

    }
}
```

注意事项

1. dat缓存每3个数据构成一个灯的颜色，按绿色、红色、蓝色的顺序排。如上所示，test数组里一共9个数据，前三个构成第一个灯，由于只有绿色部分是0xff，其他是0x00，所以第一个灯是纯绿色的。以此类推。
2. len参数直接填发送的数量，由于一个灯有3个数据，所以这个参数也是3的倍数。如果填的不是3的倍数或者超过了dat缓存的大小，那么会引起不可预知的错误。
3. ws2812的时序不可被打断，但考虑到实际项目的复杂性，没有在本函数里进行中断的屏蔽，因此请自行选择“关闭中断”或“在中断里执行”。
    - 关闭中断是指在执行ws2812_send之前执行“EA=0”关闭全部中断，在执行完ws2812_send之后再执行“EA=1”打开全部中断。
    - 在中断里执行是指将ws2812_send直接放在中断回调里面执行，比如定时器0中断、串口中断等。同时确保这个中断的优先级是最高的（在NVIC库里可以调节），这样就能保证ws2812的时序不会被打断。

优化建议

无