



RS485 Board (3.3V/5V) 用户手册

1. 产品概述

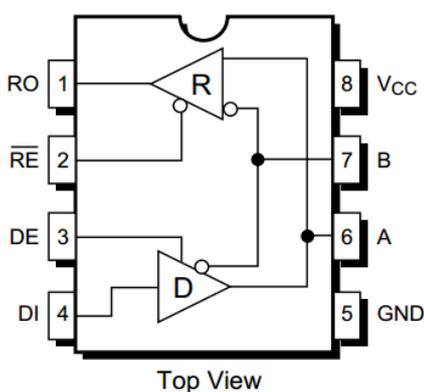
RS485 是一种基于差分信号传送的串行通信链路层协议。它解决了 RS-232 协议传输距离太近(15m)的缺陷，是工业上广泛采用的较长距离数据通信链路层协议。

由于它使用一对双绞线传送差分信号，属半双工通信，所以需要进行接收和发送状态的转换。

- 功能简介： RS485 收发器
- 最大特点： 支持多种通信总线接口
- 典型应用： RS485 通信
- 主要资源： SP485 / MAX485，控制接口，通信总线接口

2. 硬件说明

2.1 芯片引脚功能



SP485/MAX485/SP3485/MAX3485

引脚 (顶视图)

引脚	名称	说明
1	RO	接收器输出 Receiver Output
2	\overline{RE}	接收输出使能 Receiver Output Enable 低电平有效 Active LOW
3	DE	发送输出使能 Driver Output Enable 高电平有效 Active HIGH
4	DI	发送器输入 Driver Input
5	GND	地 Ground Connection
6	A	差分信号正向端 Driver Output/Receiver Input. Non-inverting
7	B	差分信号反向端 Driver Output/Receiver Input. Inverting
8	V_{CC}	$4.75V < V_{CC} < 5.25V$

SP485 / MAX485 是 5V 的 RS485 收发器

SP3485 / MAX3485 是 3.3V 的 RS485 收发器

2.2 模块功能说明

2.2.1 模块引脚说明

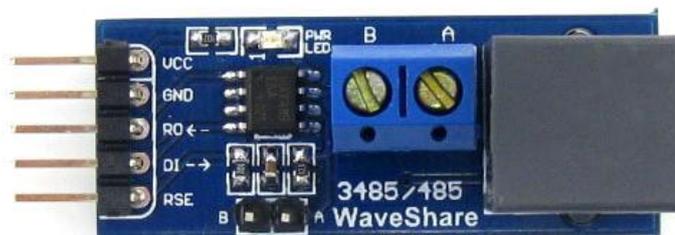


图 1. 模块顶视图

引脚符号	含义
V _{CC}	单片机的电源 V _{CC}
GND	单片机的电源 GND
RO	接收数据端，TTL 电平，可以接单片机的 RX
DI	发送数据端，TTL 电平，可以接单片机的 TX
RSE	高电平有效的发送允许，低电平有效的接收允许；普通 IO 口控制就可以。
A	差分信号正向端
B	差分信号反向端

表 1. 模块引脚含义表

2.2.2 工作原理

RS485 芯片通常处于接收状态。

当要发送数据时：

- 1) 由程序控制 RSE 变为高电平。
- 2) UART 单元发送数据。
- 3) 程序等待发送完毕后，控制 RS485 芯片转换到接收状态。
- 4) 发送完毕的标志一般由 UART 的特定寄存器提供状态指示，程序需要去查询。

2.3 示例程序

2.3.1 准备工作

- RS485 模块两个
- STM32 开发板两个（本手册用的是微雪电子的 Open103R 开发板，主

控芯片是 STM32F103R)

- 线材若干

2.3.2 测试步骤

- 1) 根据下表,将模块的对应管脚分别接入到两块 Open103R 开发板上面。

RS485 模块引脚	Open103R 引脚
V _{CC}	V _{CC}
GND	GND
RO	PA2
DI	PA3
RSE	PA0

表 2. RS485 模块和 Open103R 引脚连接表

- 2) 连接 ULINK2 到其中一个 Open103R 开发板的 JTAG 接口, 下载接收程序 USART_accept。
- 3) 连接 ULINK2 到另一个 Open103R 开发板的 JTAG 接口, 下载发送程序 USART_send。
- 4) 用杜邦线对应连接两个模块的 A、B 两个接口。(A 端连接另一个模块的 A 端, B 连接另一个模块的 B 端)
注意: 除了可以直接用杜邦线连接两个模块的 A 和 B 端之外, 用户也可以用 RJ11 普通电话线把两个模块连接起来, 实现的效果是一样的。
- 5) 上电之后, 如果观察到两个板子的 LED 同时闪烁, 则说明 RS485 通信是成功的。

2.3.3 基本发送程序说明

```
int main(void)
{
    unsigned char i;
    USART_Configuration(); //串口初始化
    //LED (PC9、PC10、PC11、PC12) 和模块 RSE 脚的初始化。
    GPIO_Configuration();
    GPIO_SetBits(GPIOA,GPIO_Pin_0); //PA0 设置为 1, 也就是 RS485 模块
    是发送模式。
    while (1)
    {
        for(i=0;i<4;i++)
        {
            Delay(0x005fffff); //延时
            printf("%c",table[i]); //串口发送数据; 也就是 RS485 发送数
```

```
        据。  
        GPIO_Write(GPIOC,table[i]<<9); //LED 点亮;  
    }  
}  
}
```

2.3.4 基本接收程序说明

```
int main(void)  
{  
    USART_Configuration();//串口初始化  
    //LED（PC9、PC10、PC11、PC12）和模块 RSE 脚的初始化  
    GPIO_Configuration();  
    NVIC_USART_Configuration();//串口中断初始化  
    GPIO_RSEetBits(GPIOA,GPIO_Pin_0); //PA0 设置为 0； 此时 RS485 模  
    块是接收模式。  
    while (1);  
}  
  
void USART2_IRQHandler(void) //串口接收响应中断函数，  
{  
    if(USART_GetITStatus(USART2, USART_IT_RXNE) != RSEET)  
    {  
        USART_ClearITPendingBit(USART2,USART_IT_RXNE); //清除中断  
        USART_SendData(USART2, USART_ReceiveData(USART2)); //串口发  
        送数据  
        GPIO_Write(GPIOC,USART_ReceiveData(USART2)<<9); //把接收到  
        的数据显示到 LED。  
    }  
}
```

2.3.5 实验现象

两个板子的 LED 同时显示相同的效果。