



机械与能源工程系

SUSTech

Department of  
Mechanical and Energy  
Engineering

# 实 验 报 告

课程名称： 机电一体化

课程编号： ME333

实验题目： 串口通信实验

学 号： 12313215

姓 名： 梁家源

专 业： 机器人工程

指导教师： 曾千里 柯文德

实验成绩：

实验日期： 2025 年 4 月 5 日

# 串口通信实验

## 一、实验目的

1. 上位机向下位机发送数据，下位机接收到后，将所接收到的数据完整发回上位机。

## 二、实验仪器和用具

主要仪器设备：

1. 计算机
2. Keil uVision 软件
3. XCOM/SSCOM 串口助手

## 三、实验方法与步骤

### 1、代码执行流程

主循环持续调用按键检测函数，获取按键值 key。按下 Key0 后，开始统计 STM32 接收到的信息，按下 Key\_UP 后，STM32 停止统计，在电脑端显示出 STM32 已经接收了多少条消息，并用 LED 灯的亮灯次数表示消息条数。未检测到按键时延时 10ms 降低 CPU 占用。

### 2、串口助手初始化

调整波特率为 115200 并连接上串口。



### 3、程序初始化

在 main 函数中先声明 count 变量作为接收消息的计数器。在原代码基础上增加 KEY\_Init() 便于后续按键操作。

```
16 int main(void)
17 {
18     u8 key;
19     int count=0;
20     u8 led0sta=1,led1sta=1;
21     u16 times=0;
22     Cache_Enable();           //打开L1-Cache
23     HAL_Init();               //初始化HAL库
24     Stm32_Clock_Init(432,25,2,9); //设置时钟,216Mhz
25     delay_init(216);          //延时初始化
26     uart_init(115200);        //串口初始化
27     LED_Init();               //初始化LED
28     KEY_Init();               //初始化按键
```

4、按下 Key0 后，开始统计 STM32 接收到的信息，按下 Key\_UP 后，STM32 停止统计。

如程序所示，在循环中检测 Key0 是否按下，当按下后，count 归零，在 Key\_UP 按下前，进入下一级 while 循环并实时检测 Key\_UP，同时若检测到上位机有数据发送，则使得 count 加一。

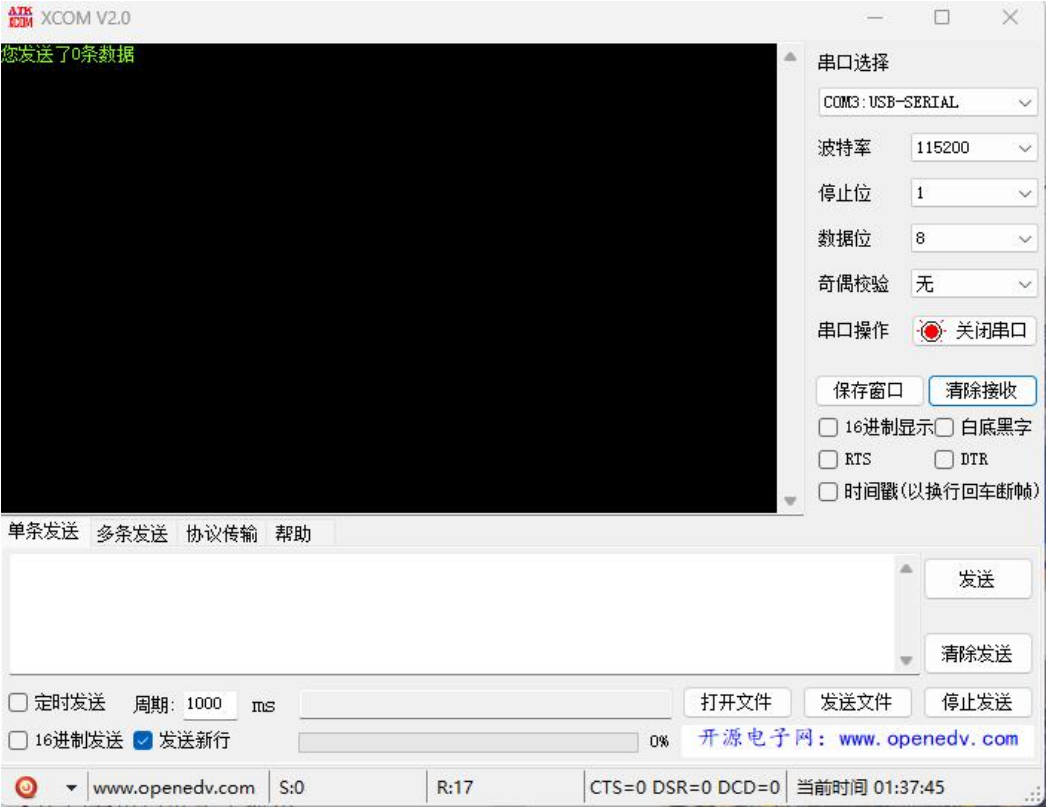
在检测到 Key\_UP 按下后，跳出循环，并通过 printf 语句在串口助手输出消息条数。然后通过 while 循环控制 LED 闪烁，通过控制循环条件和每次循环 count 一来使得闪烁次数为消息条数。

```
29 while(1)
30 {
31     key= KEY_Scan(0);
32     if(key == KEY0_PRES){           //当Key0按下时
33         count = 0;                  //重置count
34         USART_RX_STA=0;             //重置接收到的消息
35         while(key!=WKUP_PRES){      //在Key_UP按下之前循环
36             key= KEY_Scan(0);        //更新按键状态
37             if(USART_RX_STA&0x8000){ //判断上位机是否发送数据
38                 while(!_HAL_UART_GET_FLAG(&UART1_Handler,UART_FLAG_TC)!=SET); //等待发送结束
39                 USART_RX_STA=0;
40                 count++;              //计数器加一
41             }
42         }
43         printf("您发送了%d条数据\r\n",count);
44         while(count>0){              //通过count数值来循环亮灯次数
45             led1sta=0;
46             led0sta=0;
47             LED0(led0sta);
48             LED1(led1sta);           //灯亮
49             delay_ms(100);
50             led0sta=1;
51             led1sta=1;
52             LED0(led0sta);
53             LED1(led1sta);           //灯灭
54             delay_ms(200);
55             count--;                 //每次循环count减一
56         }
57     }
58     delay_ms(10);
59 }
```

# 四、实验分析及结论

## 一、实验结果分析

按下 Key0 后向下位机发送数据，按下 Key\_UP 后停止统计并在客户端显示数量，同时 LED 灯闪烁。



按下 Key0 后立刻按下 Key\_UP，显示发送 0 条数据，LED 灯不亮。



按下 Key0 后通过“多条发送”发送十条数据，按下 Key\_UP，显示发送 10 条数据,LED 灯闪烁十次。

## 二、实验结论

实验成功实现了通过按键控制 STM32 消息统计的功能，验证了外部中断、串口通信和 GPIO 控制的协同工作能力。串口接收中断的高效性确保了消息统计的准确性，LED 灯和串口输出的双重反馈增强了系统的可观测性。