

实验 4：串口实验

一、实验器材

硬件：ARM-STM32 开发板，J-Link/St-Link。

软件：Win7/Win8/Win10, Keil uVision5

二、实验要求

1. 功能 1 实现：从串口调试助手输入自己的姓名，返回“Hello, xxx”。
2. 功能 2 实现：从串口调试助手输入命令：led0 on/led0 off/led1 on/led1 off 四种命令，可控制 led 灯的亮灭。
3. 功能 3 实现：按键输入时，串口调试助手输出提示“XX pressed.”。

三、实验过程

1. 根据 lab 课内容，新建项目，添加文件，添加依赖。
2. 实现三个功能：
 - (1) 判断输入的内容是不是 led 的调试命令，如果是，控制 led 灯的亮灭。
 - (2) 如果不是 led 的调试命令，串口调试助手输出“Hello, xxx”。
 - (3) 判断按钮是否被触发，如果是，串口调试助手输出“xxx pressed”。
3. 烧录，修改 Target 的 Options 中 Debug 选项，选用“Use ST-Link Debugger”，load 上板。
4. 测试，测试所有情况，观察实验板是否和预期显示一致。

四、实验结果

软件代码（只需用户实现功能的主要代码部分）：

实际验证附图（含开发板状态拍照、仿真截图等）：

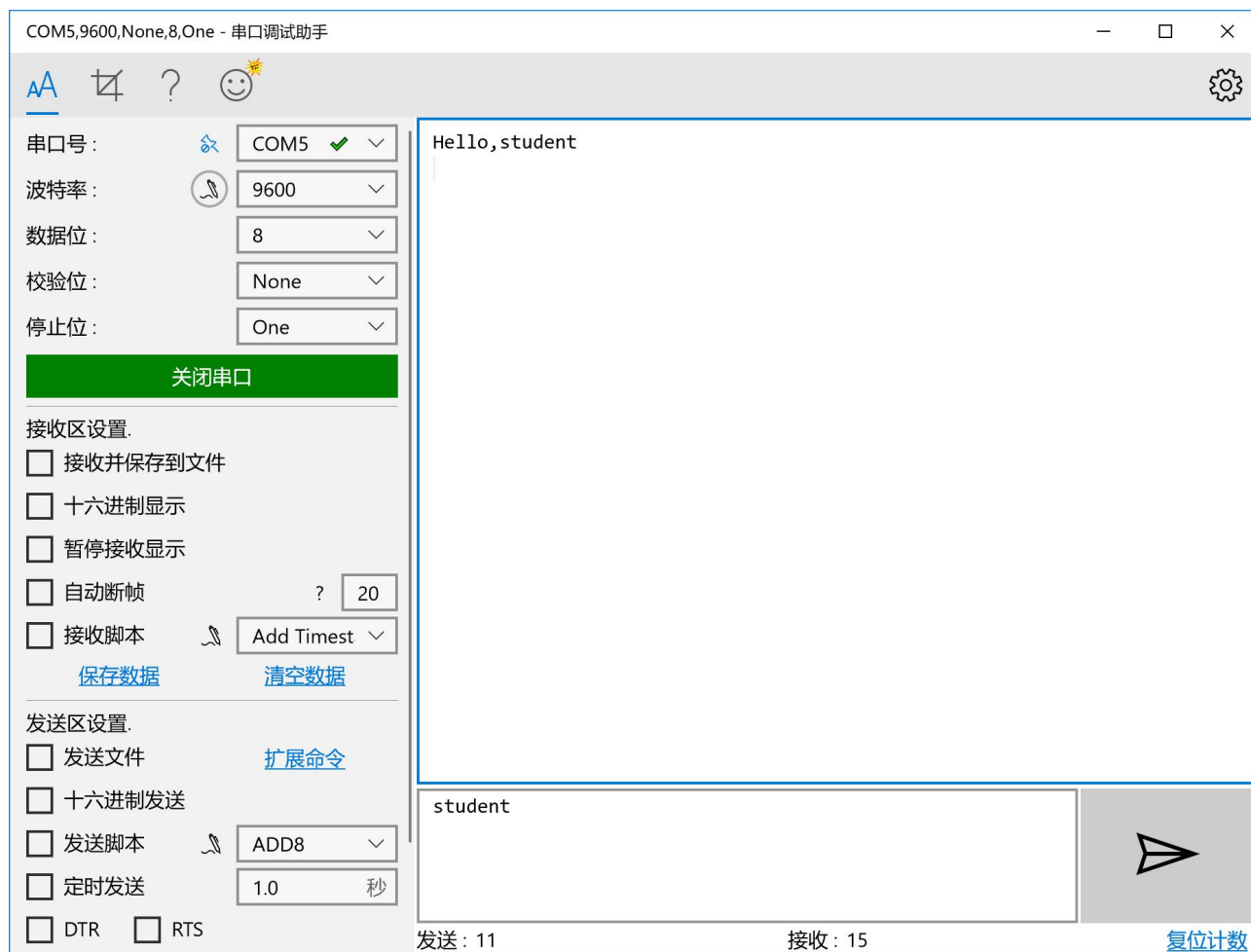
功能 1：

```

8  char str[1024];
9  void function()
10 {
11     u8 t;
12     u8 len;
13     u8 key;
14     while(1) {
15         if(USART_RX_STA&0x8000)
16         {
17             len = USART_RX_STA & 0x3fff;    // 获取数据长度
18             for (t = 0; t < len; t++) {
19                 str[t] = USART_RX_BUF[t];
20             }
21             if(strcmp(str, "led0 on") == 0)
22             {
23                 GPIO_ResetBits(GPIOA,GPIO_Pin_8);    // led0亮
24                 delay_ms(100);
25             }
26             else if(strcmp(str, "led0 off") == 0)
27             {
28                 GPIO_SetBits(GPIOA,GPIO_Pin_8);    // led0灭
29                 delay_ms(100);
30             }
31             else if(strcmp(str, "led1 on") == 0)
32             {
33                 GPIO_ResetBits(GPIOD,GPIO_Pin_2);    // led1亮
34                 delay_ms(100);
35             }
36             else if(strcmp(str, "led1 off") == 0)
37             {
38                 GPIO_SetBits(GPIOD,GPIO_Pin_2);    // led1灭
39                 delay_ms(100);
40             }
41             else if(len != 0)    // 如果不是led调试命令, 则返回"Hello, xxx"
42             {
43                 printf("Hello, ");
44
45                 for(t=0;t<len;t++)
46                 {
47                     USART1->DR=USART_RX_BUF[t];
48                     while( (USART1->SR&0X40)==0);
49                 }
50                 printf("\r\n");
51             }
52             memset(USART_RX_BUF,0, sizeof(USART_RX_BUF));
53             memset(str,0, sizeof(str));
54             USART_RX_STA = 0;
55         }

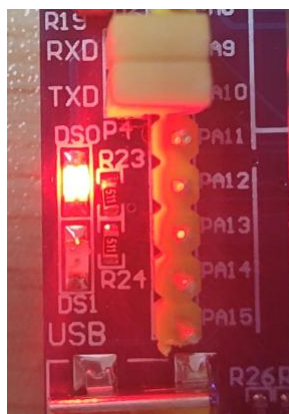
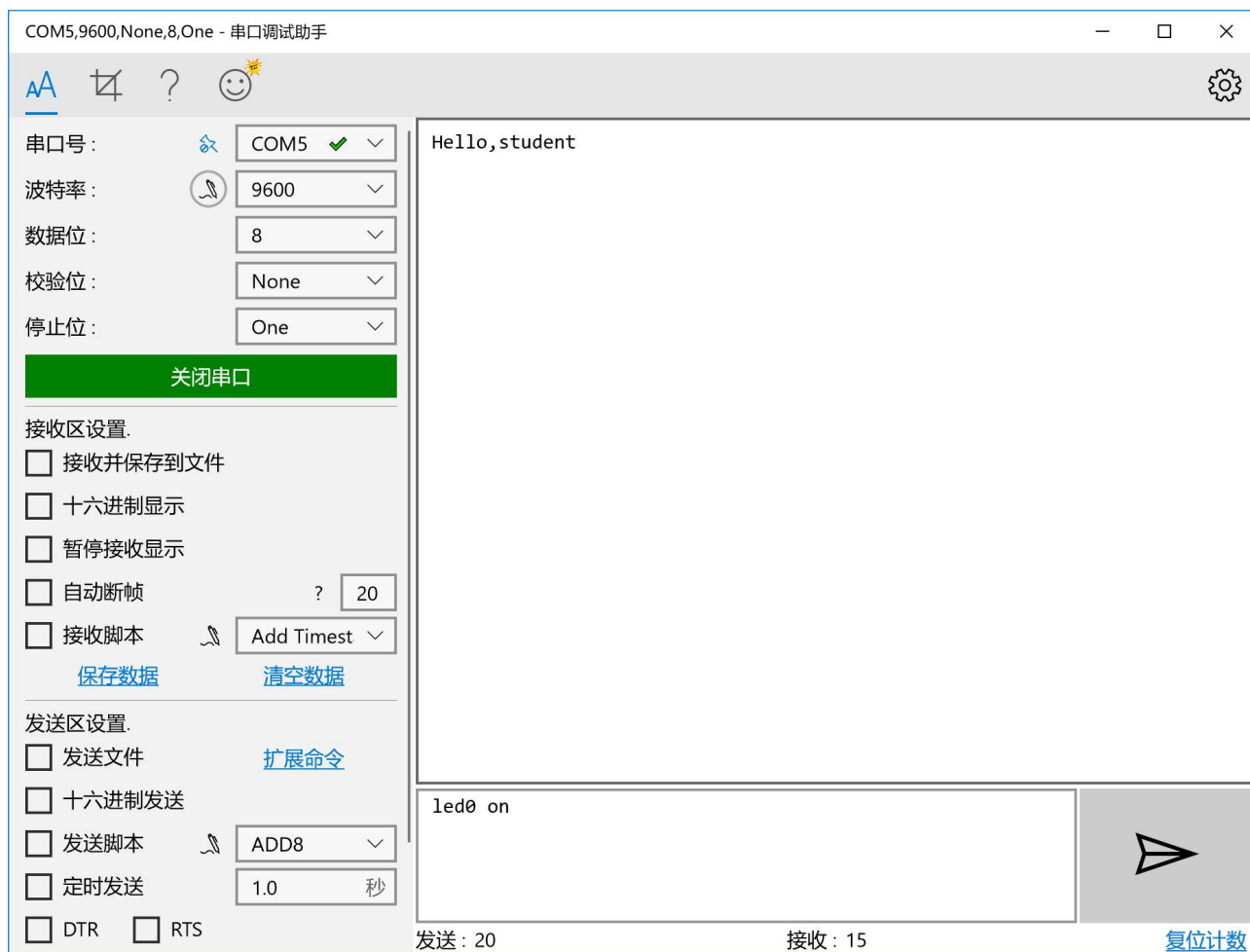
```

```
56     else
57     {
58         key = KEY_Scan(0); // 读取按键状态
59         if(key == KEY0_PRES)
60             printf("key0 pressed\r\n");
61         else if (key == KEY1_PRES)
62             printf("key1 pressed\r\n");
63
64     }
65 }
66 }
67
68 int main(void)
69 {
70     delay_init(); // 延时初始化
71     NVIC_PriorityGroupConfig(NVIC_PriorityGroup_2);
72     uart_init(9600); // 串口波特率为9600
73     LED_Init(); // LED灯初始化
74     KEY_Init(); // 按键初始化
75
76     function();
77 }
78
```

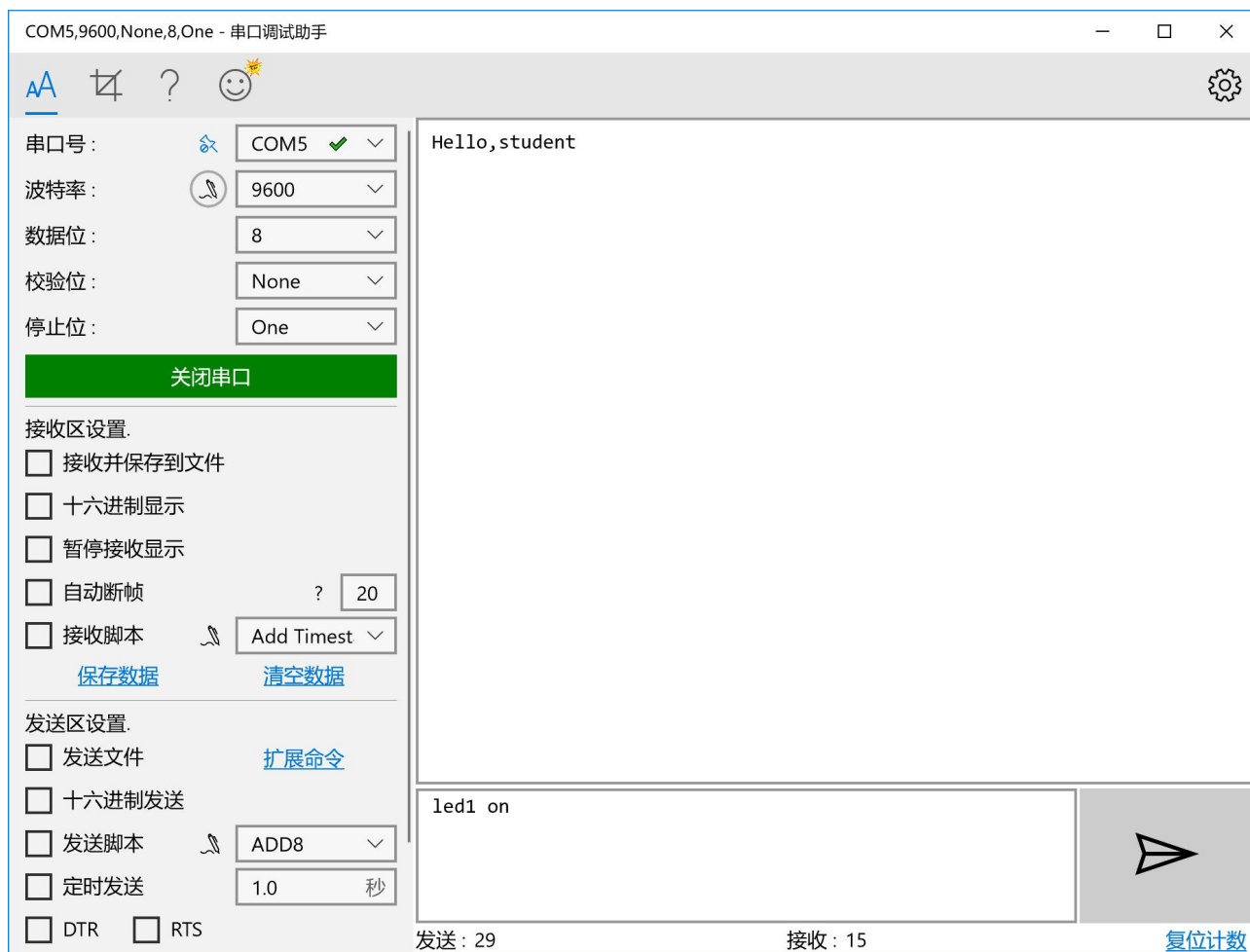


功能 2:

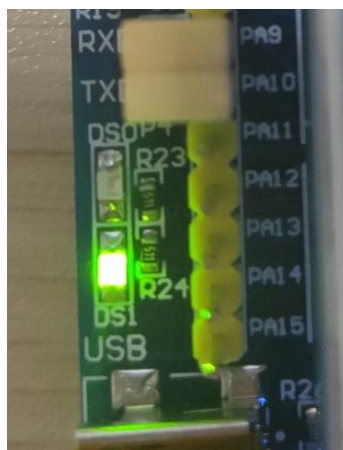
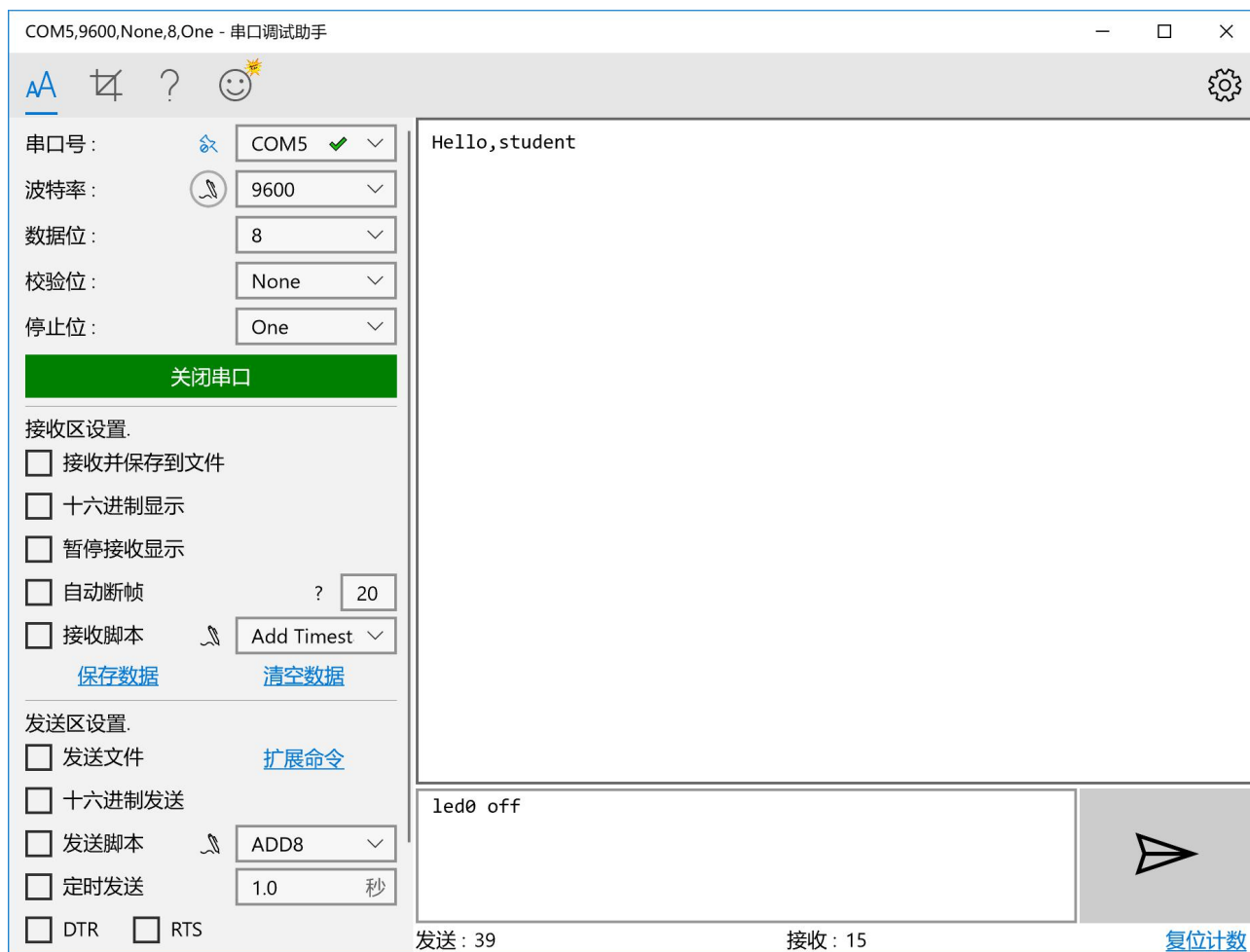
led0 on:



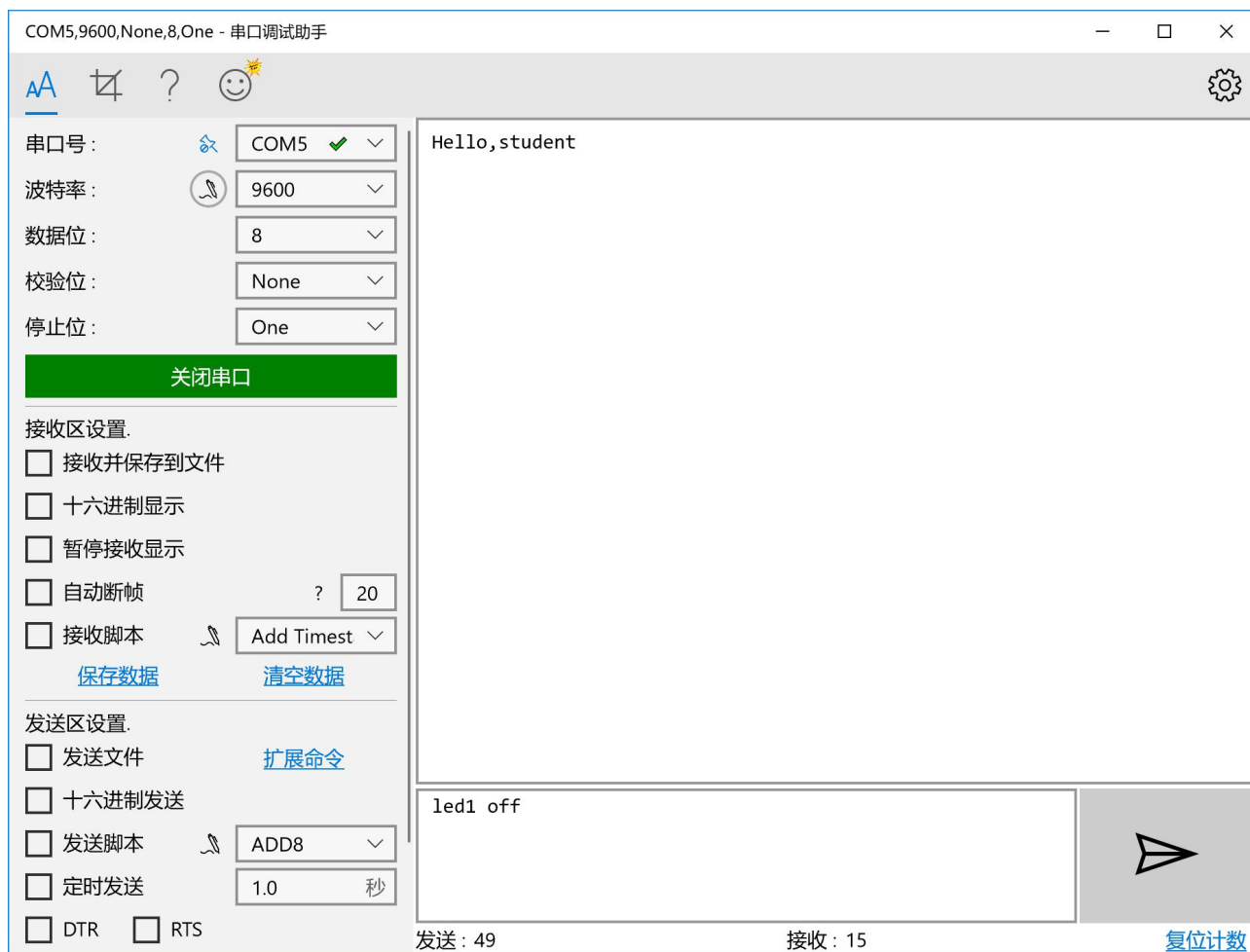
led1 on:



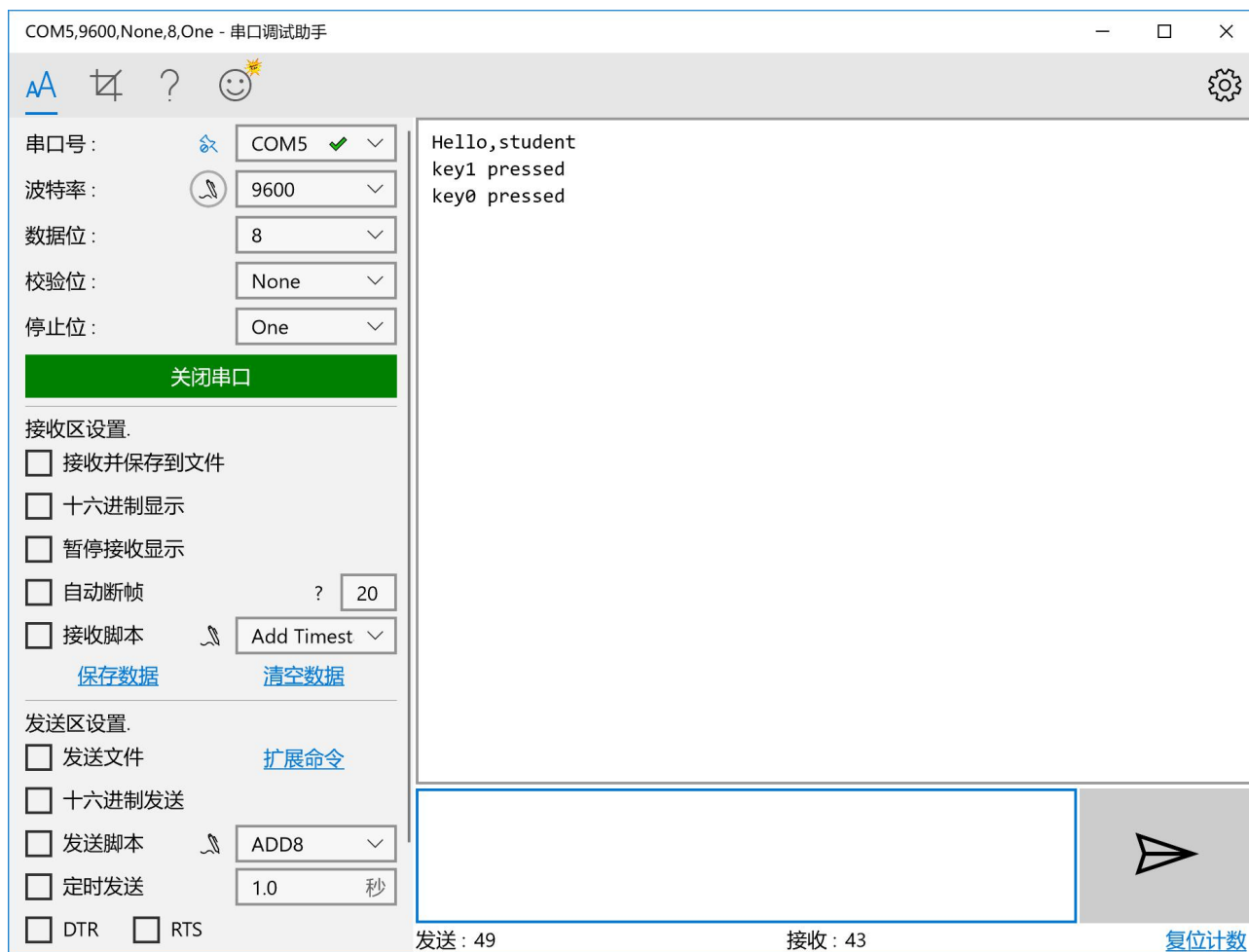
led0 off:



led1 off:



功能 3:



五、实验总结

简述在实验过程中出现的问题，解决的过程和结果，及其他需要说明的情况。

1. 比较字符串的时候，需要 include <string.h>，使用里面的 strcmp 方法，如果返回值为 0 说明两个字符串相同。
2. 每次串口接受完信息后，记得清楚 USART_RX_BUF，并将 USART_RX_STA 置为 0。
3. 串口波特率需要设置为默认的 9600。
4. 串口调试工具内，如果代码里面返回值有中文，需要将编码格式改为 utf-8。
5. 串口调试工具内，发送的内容后需要加一个回车。
6. 确认好自己的板子是 J-Link 还是 ST-Link，在 Options->Debug 选择，如果是错误的连接方式，会检测不到板子。
7. 板子的杜邦线需要按照不同的连接方式对应连好，插紧，如果差不紧也无法检测到板子。
8. 课件中是 J-Link 的示范，如果使用 ST-Link 需要下载并添加对应的驱动。
9. 上板测试，需要在 Options->Debug 选择 Use ST-Link Debugger。

10. 每次上板前记得重新 build 一次。
11. 写代码的时候，注意提前把需要调用的库文件先添加进来。
12. c 语言不可以在 for 循环内定义参数，所以提前定义循环次数变量。