

实验八 UART 以及与 PC 的 RS232 通信技术

1. 实验要求

完成平台自带上述实验例程验证，读懂代码并理解 UART 的硬件接口，以及软件配置使用，在此基础上，完成下面作业，掌握串口通信的软硬件原理；

1) 特别提醒在完成 3.9.1 实验时，在短接 TX 和 RX 的时，务必在断电状态下插拔跳线，违反操作规则的同学，直接扣分。

2) 基本功能：实现 MSP430-PC 之间的双向通信。

通过 USB 虚拟仿真串口，实现 PC 与实验箱之间的 UART-RS232 通讯，PC 端的软件可以用“超级串口测试终端”或者“自编软件”，需要实现双向通行。例如

a) PC 传一些字符/一组数据/一段波形数据/一幅图片到 MSP430,并在 430 的液晶屏上显示,; b) MSP430 实验箱上的数字或者字符（可通过键盘输入），通过串口输出到 PC 并显示,并将数据在 PC 本地保存为数据文件。

注意事项：中断服务程序只能完成紧急的任务，不要在中断服务程序中执行很耗时的任务，否则会影响整个系统的实时性，比如发生 UART 收发会丢数据的现象等。

2. 设计思路

本实验通过 RS232 进行双向通信，所使用的配置为：波特率 9600，8 位数据位，1 位停止位，无奇偶校验。在实验一例程（短接 TX 和 RX 实现通讯）的基础上，使用 J6 接口与 PC 机进行互联，此功能可通过串口通信助手进行展示。在原程序的基础上，选择合适的端口和波特率后即可实现发送字符串的最

后一位字符的互相通信，为了实现其他形式信息的通信（字符串/长数据/图片等），在接收中断里，我们使用数组 `recv_data` 进行实时存储并通过定时器进行刷新显示，并根据传递信息的形式设计定时器进行清除。

3. 实验代码

以下为相对例程的更改部分：

```
31
32 while(1)
33 {
34     if(flag0)          //flag0在定时器计时到达1s时被赋值为1
35     {
36         flag0=0;
37         UCA1TXBUF=1;    //写一个字符到发送缓存发送数据
38         etft_DisplayString(send_data,170,100,65535,0);    //TFT屏上显示发送的字符
39         send_data[0]++;    //字符加1
40         if(send_data[0]>'9')    //字符超过'9'则重新置'0'
41             send_data[0]='0';
42         recv_data[0] = '\0';
43         l=0;
44         etft_AreaSet(0,0,319,239,0);
45     }
46
47     if(flag1)          //flag1在接收中断中被赋值为1
48     {
49         flag1=0;
50         etft_AreaSet(0,0,319,239,0);
51         etft_DisplayString(recv_data,80,140,65535,0);    //TFT屏幕上显示接收到的字符
52         etft_DisplayImage(recv_data,60,60, 60, 60);    //这是显示图片
53     }
54 }
55
56 }
57
58
59 #pragma vector=USCI_A1_VECTOR    //USCI中断服务函数
60 __interrupt void USCI_A1_ISR(void)
61 {
62     switch(__even_in_range(UCA1IV,4))
63     {
64         case 0:break;    //无中断
65         case 2:          //接收中断处理
66             while(!(UCA1IFG&UCTXIFG));    //等待完成接收
67             recv_data[l]=UCA1RXBUF;    //数据读出
68             recv_data[l+1]='\0';
69             l++;
70             //flag1=1;    //接收中断标志位
```

4. 实验结果

3.9.1 实验短接 TX 和 RX 实验结果可见附件 lab.mp4

与 PC 双向通信视频可见附件 exp8.MP4

5. 分析与思考

在一开始使用其他串口通信助手时出现一直显示端口不可用的情况，转换成例程文件夹中通信助手未出现此问题。在实验中，需要根据不同的数据形式在接收双方选择不同的处理和存储方式。同时，对于较长较多的数据，可以通过合适的时间间隔比较或实现约定好的结束符进行传输结束的判断。