



# **第10章 STC单片机串行异步收发器**

## **原理及实现**

**何宾**

**2015.02**

# 串口1通信实例1

在该设计中，将在STC提供的学习板上，使用串口1，定时器1的模式0实现STC学习板和主机电脑的串口通信。

## ■ STC通过串口1向主机发送菜单界面。

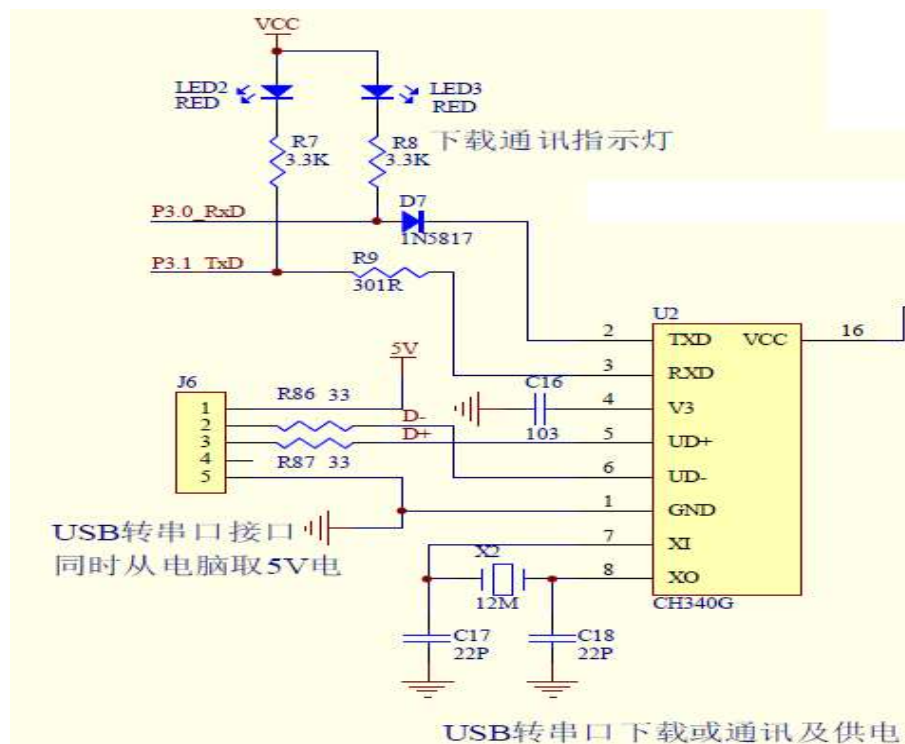
- 在主机上，按1键，用于控制STC学习板上的标记为LED10的LED灯；
- 按2键，用于控制STC学习板上标记为LED9的LED灯；
- 按其他键显示退出程序的信息。

```
-----main menu-----  
      input 1:  Control LED10  
      input 2:  Control LED9  
      other   :  exit program  
-----end menu-----
```

# 串口1通信实例1

在该设计中，使用STC学习板上的串口1。

- CH340G芯片用于将IAP15W4K58S4单片机的串口信号TxD和RxD转换成USB信号，方便与电脑USB接口的连接。



# 串口1通信实例1

- 串口发送信号TxD信号连接到STC单片机的P3.1引脚，该引脚将从STC单片机发送数据给主机；
- 串口接收信号RxD信号连接到STC单片机的P3.0引脚，该引脚将接收来自主机的数据。
- 在该电路设计中，LED2和LED3上拉，并且连接到RxD和TxD信号线
  - 用于指示STC单片机串口和主机之间发送和接收数据的情况。
- 在P3.0引脚上加入IN5817二极管，以及在P3.1引脚串入电阻的作用是为了防止USB器件给芯片供电。

# 串口1通信实例1

**【例】 主机通过串口控制STC板上LED灯C语言描述的例子。**

```
#include "reg51.h"

#define FOSC 1843200L           //声明当前单片机主时钟频率
#define BAUD 115200            //声明波特率常数115200
sfr AUXR =0x8E;                //声明AUXR寄存器的地址0x8E
sfr TH2  =0xD6;                //声明TH2寄存器的地址0xD6
sfr TL2  =0xD7;                //声明TL2寄存器的地址0xD7
bit busy=0;                    //声明比特位busy
xdata char menu[]={"\r\n-----main menu-----" //声明字符型数组menu
                    "\r\n  input 1: Control LED10 "
```

# 串口1通信实例1

```
"\r\n  input 2: Control LED9 "
```

```
"\r\n  other : Exit Program"
```

```
"\r\n-----end menu-----"};
```

```
void SendData(unsigned char dat)
```

```
{  while(busy);
```

```
    SBUF=dat;
```

```
    busy=1; }
```

```
void SendString(char *s)
```

```
{
```

```
    while(*s!= '\0' )
```

```
        SendData(*s++);}
```

```
//声明SendData子函数, 参数dat
```

```
//判断是否忙, 忙等待
```

```
//将dat写入SBUF发送缓冲器
```

```
//将busy标志置1
```

```
//声明SendString子函数, 参数s
```

```
//判断字符是否结束, 如果没结束
```

```
//调用SendData子函数发送数据
```

# 串口1通信实例1

```
void uart1() interrupt 4
```

```
//声明串口1中断服务程序uart1
```

```
{
```

```
    if(RI)
```

```
//如果接收标志RI为1,有接收数据
```

```
        RI=0;
```

```
//将RI标志清零
```

```
    if(TI)
```

```
//如果发送标志TI为1,已发送数据
```

```
        TI=0;
```

```
//将TI标志清零
```

```
    busy=0;
```

```
//busy清零，表示已经发送完数据
```

```
}
```

```
void main()
```

```
{
```

```
    unsigned char c;
```

```
//定义无符号字符型变量c
```

# 串口1通信实例1

```
P46=0;           //P4.6端口置0, 灯亮
P47=0;           //P4.7端口置1, 灯亮
SCON=0x50;       //串口1方式1, 允许接收
AUXR=0x14;       //允许定时器2, 不分频
AUXR|=0x01;      //选择定时器2作为波特率发生器
TL2=(65536-((FOSC/4)/BAUD)); //初值低8位赋值给TL2寄存器
TH2=(65536-((FOSC/4)/BAUD))>>8; //初值高8位赋值给TH2寄存器
ES=1;           //允许串口中断
EA=1;           //CPU允许响应中断请求
SendString(&menu); //在串口中断上打印menu的内容
```



# 串口1通信实例1

```
P46=0;
```

//P4.6端口置0, 灯亮

```
while(1){
```

//无限循环

```
    if(RI==1)
```

//如果接收到上位机发送的数据

```
    {
```

```
        c=SBUF;
```

//从SBUF缓冲区读数据到变量c

```
        if(c==0x31)
```

//判断如果接收的数据是字符 '1'

```
            P46=!P46;
```

//P4.6取反

```
        else if(c==0x32)
```

//判断如果接收的数据是字符 '2'

```
            P47=!P47;
```

//P4.7取反

```
        else
```

//其它任何输入

```
        {SendString( "\r\n Exit Program" ); //串口上打印Exit Program信息
```

```
    }  
}
```

# 串口1通信实例1

下面说明该代码的设计原理和验证方法，步骤包括：

- 使用T2定时器，根据前面给出的IRC的时钟频率为18.432MHz，波特率为115200，由于，T2的溢出率和波特率存在下面的关系，即：

$$\text{串口1的波特率} = \text{SYSclk} / (65535 - [\text{RL\_TH2}, \text{RL\_TL2}] / 4)$$

$$\text{因此，} [\text{RL\_TH2}, \text{RL\_TL2}] = 65536 - \text{SYSclk} / (\text{串口1波特率} \times 4)$$

注：RL\_TH2是T2H的自动重加载寄存器，RL\_TL2是T2L的自动重加载寄存器。

- 打开STC-ISP软件，在该界面内，选择硬件选项。将“输入用户程序运行时的IRC频率”设置为18.432MHz。
- 单击下载/编程按钮，按前面的方法下载设计到STC单片机。

# 串口1通信实例1

■ 在STC-ISP软件右侧串口中，选择串口助手标签。在该标签串口界面下，按下面设置参数：

- 串口：COM3（读者根据自己电脑识别出来的COM端口号进行设置）
- 波特率：115200。
- 校验位：无校验。
- 停止位：1位。



# 串口1通信实例1

- 单击打开串口按钮。
- 在STC学习板上，找到并按一下SW19按键，重新运行程序。可以看到在上面的接收窗口中，显示出菜单信息。
- 在发送窗口中输入1。
- 单击发送数据按钮。观察LED10的变化。
- 在发送串口中输入2。
- 单击发送数据按钮。观察LED9的变化。
- 在发送串口中输入其它字符。
- 单击发送数据按钮。看到在接收窗口中，显示 “Exit Program” 提示信息。