/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* 实验名称：键控串口信息打印实验

\* 可配书籍：上册-《深入浅出STC8增强型51单片机进阶攻略》已经出版

下册-《深入浅出STC8增强型51单片机实战攻略》还在写作

\* 书籍备注：龙顺宇编著 清华大学出版社出版

\* 淘宝店铺：https://520mcu.taobao.com/

\* 实验平台：思修电子工作室SX-RFID-B低频ID识别与应用开发板 Long

\* 芯片型号：STC8G1K08/17（微调后可移植至STC8A/F/C/G/H系列单片机）

\* 时钟说明：芯片内部11.0592MHz，在使用STC-ISP软件时需要先配置然后

\* 下载到单片机使得时钟配置生效。

\* 串口速率：9600bps

\* 实验说明：用USB线插入到Uart-USB接口给开发板供电，做好STC-ISP配置

并下载程序，在电脑上打开串口调试助手软件，选择对应的串口号（具体串

口号可在电脑设备管理器中查询），然后配置波特率为9600bps，选择文本/

字符模式接收数据，此时按下K1按键后串口将会接收到“【USER】:K1 is pr

essed!”提示，按下K2按键后串口将会接收到“【USER】:K2 is pressed!”提

示，以此学习串口资源编程及串口调试助手的使用。

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include "STC8G.h" //主控芯片的头文件

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*常用数据类型定义\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#define u8 uint8\_t

#define u16 uint16\_t

#define u32 uint32\_t

typedef unsigned char uint8\_t;

typedef unsigned int uint16\_t;

typedef unsigned long uint32\_t;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*端口/引脚定义区域\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

sbit KEY1=P3^4; //按键1

sbit KEY2=P3^3; //按键2

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*用户自定义数据区域\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#define FOSC 11059200UL //系统外部时钟频率（无符号长整型）

#define BAUD 9600 //欲配置的串口通信波特率值

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*函数声明区域\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void delay(u16 Count); //延时函数

void IO\_init(void); //IO初始化函数

void UART\_init(void); //串口初始化函数

void SendData(u8 dat); //串口发送单字节数据函数

void SendString(u8 \*s); //串口发送字符串函数

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*主函数区域\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void main(void)

{

IO\_init(); //IO初始化

UART\_init(); //串口初始化

delay(100); //等待配置稳定

while(1)

{

if(KEY1==0) //若K1按键按下

{

delay(5); //软件延时去除按键抖动

if(KEY1==0) //若K1按键确实被按下

{

SendString("【USER】:K1 is pressed!\r\n");

while(!KEY1); //等待用户按键松手

}

}

if(KEY2==0) //若K1按键按下

{

delay(5); //软件延时去除按键抖动

if(KEY2==0) //若K1按键确实被按下

{

SendString("【USER】:K2 is pressed!\r\n");

while(!KEY2); //等待用户按键松手

}

}

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//延时函数delay()，有形参Count用于控制延时函数执行次数，无返回值

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void delay(u16 Count)

{

u8 i,j;

while (Count--)

{

for(i=0;i<50;i++)

for(j=0;j<20;j++);

}

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//IO初始化函数IO\_Init()，无形参，无返回值

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void IO\_init(void)

{

//配置P3.3-4为准双向/弱上拉模式

P3M0&=0xE7; //P3M0.3-4=0

P3M1&=0xE7; //P3M1.3-4=0

delay(10); //等待I/O模式配置稳定

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//串口初始化函数UART\_init()，无形参，无返回值

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void UART\_init(void)

{

SCON=0x50; //8位数据,可变波特率

AUXR|=0x40; //定时器1时钟为Fosc,即1T

AUXR&=0xFE; //串口1选择定时器1为波特率发生器

TMOD&=0x0F; //设定定时器1为16位自动重装方式

TL1=(65536-(FOSC/4/BAUD)); //设置波特率重装值

TH1=(65536-(FOSC/4/BAUD))>>8;//设置波特率重装值

ET1=0; //禁止定时器1中断

TR1=1; //启动定时器1

ES=1; //使能UART中断开关ES

EA=1; //使能单片机总中断开关EA

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//串口发送单字节数据SendData()，有形参dat用于接收欲发送的单字节

//数据，无返回值

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void SendData(u8 dat)

{

SBUF=dat; //发送数据到发送缓冲区内

while(TI==0); //等待串口数据发送完毕

TI=0; //清除发送完成标志位TI

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

//串口发送字符串函数SendString()，有形参\*s属于指针变量，用于指向

//要发送的字符串首字节，通过指针的自增，逐一取出字符依次发送直至

//字符串结束标志'\0',无返回值

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

void SendString(u8 \*s)

{

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

{

SendData(\*s); //调用SendData()函数依次发送单字节数据

s++; //指针自增，便于发送下一个数据

}

}