# 51单片机中串口通信总结

## 一、初始化设置

TMOD=0x20; //设定T1定时器工作方式2

TH1=0xfd; //T1定时器装初值，高位

TL1=0xfd; // T1定时器装初值，低位，这样设置后，波特率被设置成9600

TR1=1; //启动定时器T1

SM0=0; //设置串口工作方式1

SM1=1; //同上

REN=1; //允许串口接收

EA=1; //开总中断

ES=1; //开串口中断

## 二、接收数据

串口中断函数：

void serial() interrupt 4

{

//接收数据

flag=0;

RI=0;

a=SBUF; //将缓存中的字符（1个）保存到变量a中

flag=1;

}

其中a和flag为全局变量，a用于保存接收到的数据，而flag用于标志正在接收数据。

由于SBUF中只有一个字节，如果想要接收一个字符串，则可以像下面这样处理。

假如数据一共有5位，协议以0xAA为数据的开始标志

void serial() interrupt 4

{

//接收数据

flag=0;

RI=0;

a=SBUF;

if(a==0xAA)

{

table[0]=0xAA;

bContinue=1;

i=1;

bFinish=0;

return;

}

else if(bContinue)

{

table[i++]=a;

}

if(i>=5)

{

table[5]='\0';

bContinue=0;

i=0;

bFinish=1;

//对字符串进行解码

……

}

flag=1;

}

这里为了便于理解，定义了两个全局变量，bContinue和bFinish，前者用于标识是否还要继续接受数据，并放入到table中，后者用于标识是否已经读取完数据。显然，bContinue和bFinish为两个相反的值。接收到0xAA时，表示数据已经开始，这时将bContinue置为1，bFinish置为0，当读取到5个数据时，数据接收完成将bContinue置为0，bFinish置为1。这时，一般要对数据进行解码，以分析上位机发送过来的指令信息。

## 三、发送数据

在发送数据时，应保证当前不正在接收数据，这可以通过上述的flag标志得知，另外需要关闭串口中断，在发送完数据以后再打开。当向SBUF中写入一个数据后，使用while(!TI)来等待数据发送完毕，因为当发送完毕后TI会由硬件置1，该标志需要手动置0。

void SendData(uchar \*pBuffer，uint n)

{

if(flag==1)

{

ES=0;//关闭串口中断

for(k=0;k<n;k++)

{

SBUF=pBuffer[k];

while(!TI);

TI=0;

}

ES=1;//打开串口中断

flag=0;

}

}