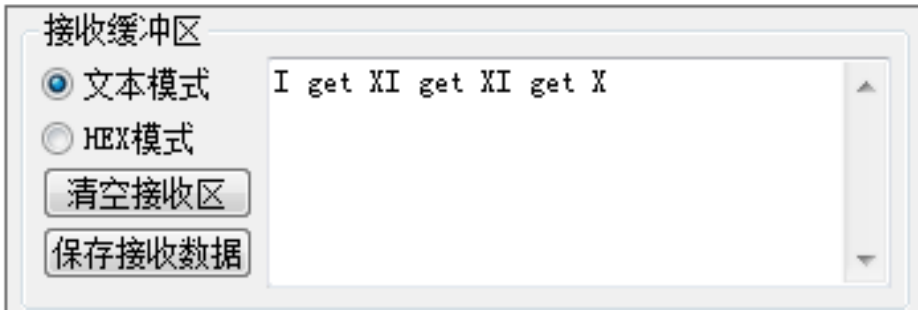
单片机原理与应用

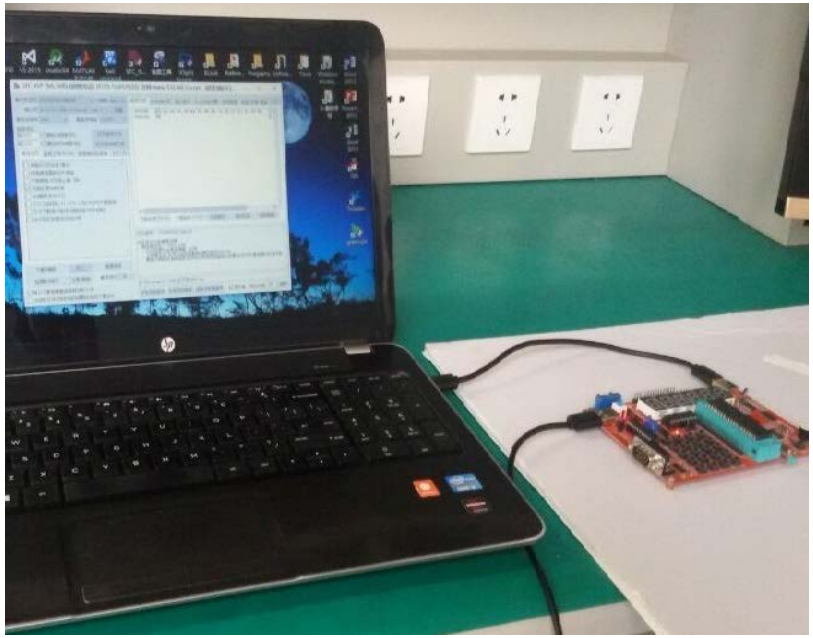
北京航空航天大学
电子信息工程学院
张玉玺
zhangyuxi@buaa.edu.cn

第八章 串口通信

- 本章实现功能

  电脑通过串口助手向单片机发送字符“X”，每发送一次，单片机收到之后返回“I get X”



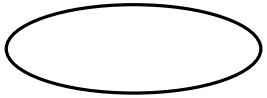


第八章 串口通信

- 如何实现功能
  - 串口工作原理
  - 串口寄存器控制
  - 工作方式与波特率
  - 流程图
  - 程序代码
  - 烧写

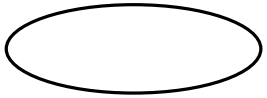
第八章 串口通信

- 串口的工作原理
  - 串行通信的概念：将数据字节分成一位一位的形式在一条传输线上逐个地传送
  - 波特率的概念
  - 串口工作流程（中断实现）
  - 接口电路



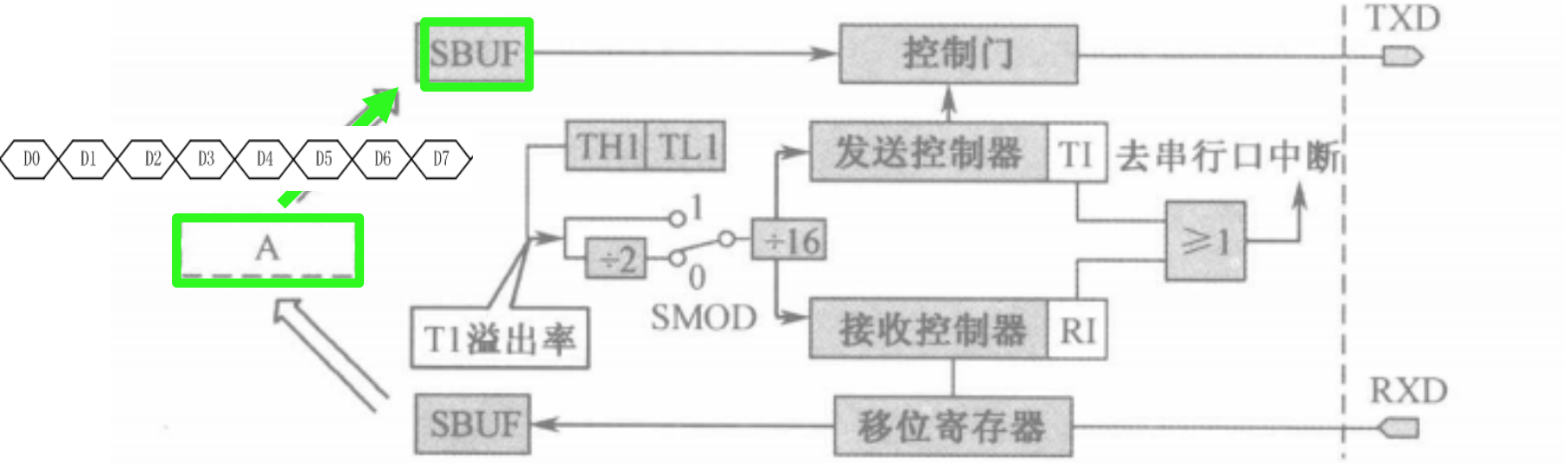
第八章 串口通信

- 波特率的概念
  - 理解为一个设备在一秒钟内发送（或接收）了多少码元的数据。代表单片机或计算机在串口通信的速率
  - 为保证串行通信顺利进行，发数据的一方发送数据的速率与接受数据方的接受数据的速率要一致



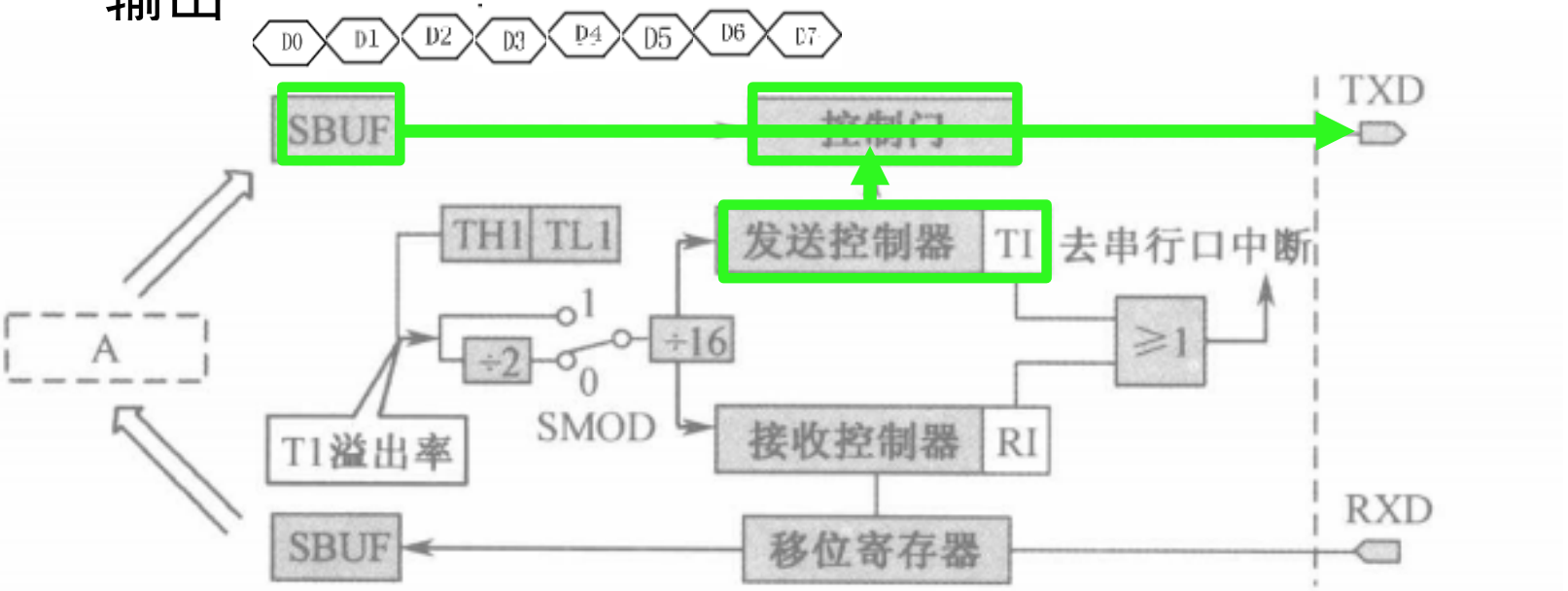
第八章 串口通信

- 串口工作流程(发送)
  - 1. 累加器A将数据输入发送缓冲器SBUF



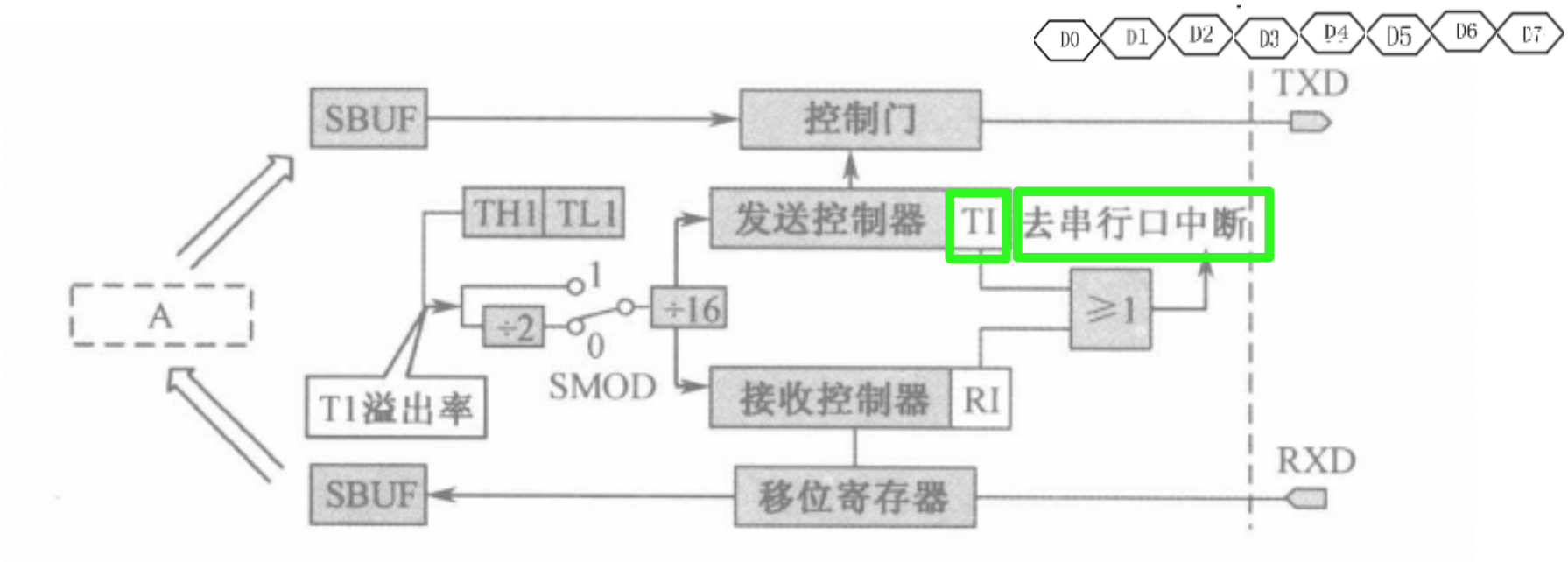
第八章 串口通信

- 串口工作流程(发送)
  - 2. 由发送控制器TI打开控制门
  - 3. 在发送时钟的控制下，将该并行数据逐位移位输出



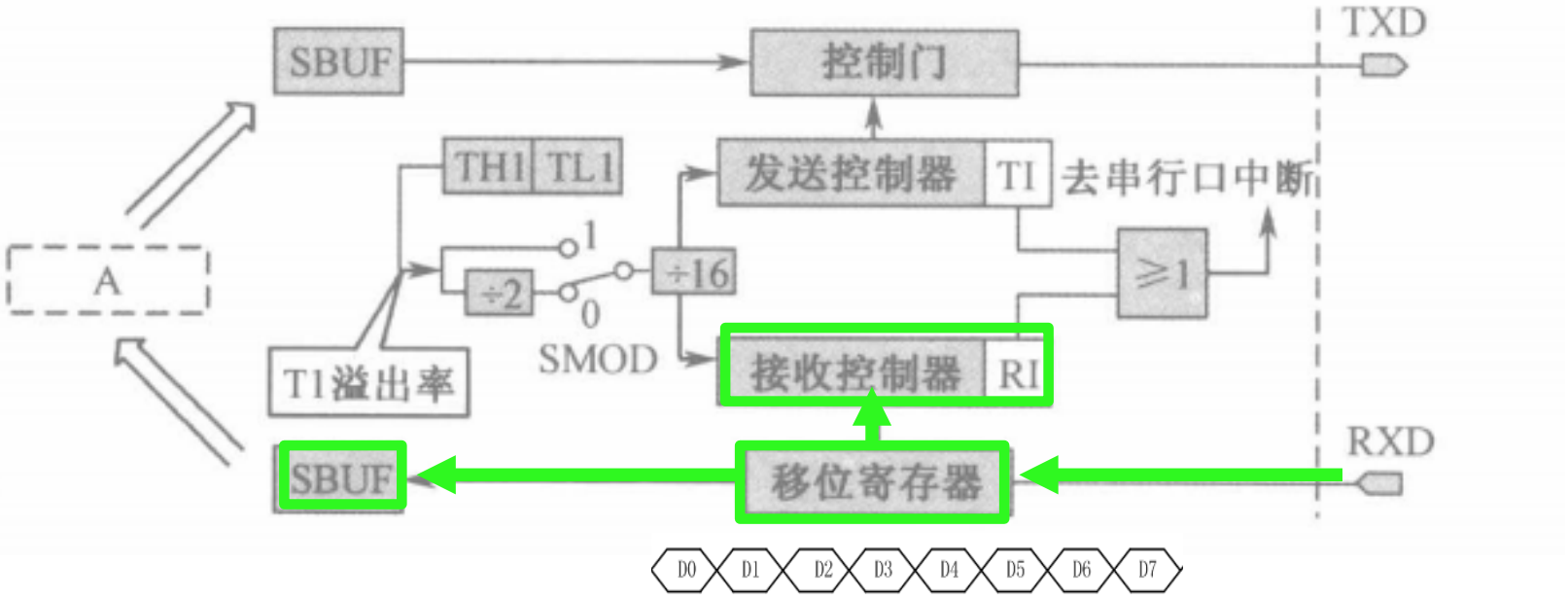
第八章 串口通信

- 串口工作流程(发送)
  - 4. 当单片机发送完一帧数据后，会置位TI，并触发串行口发送中断



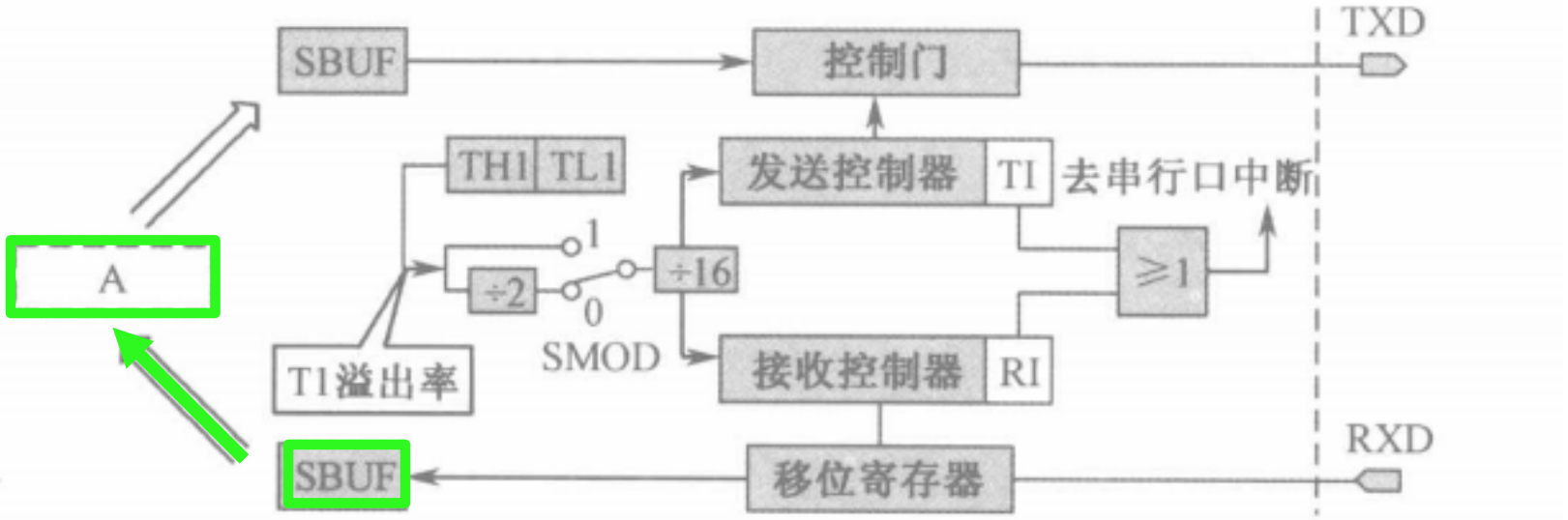
第八章 串口通信

- 串口工作流程(接收)
  - 1. 移位寄存器一位位接收数据并转换为并行格式并传给缓冲器SBUF



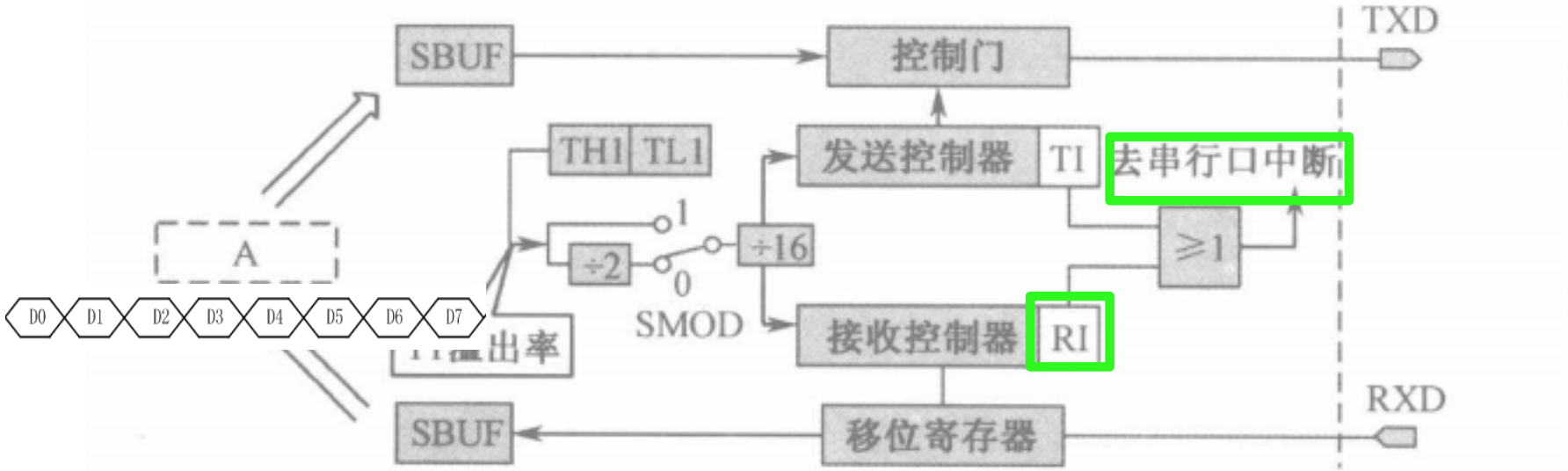
第八章 串口通信

- 串口工作流程(接收)
  - 2. SBUF接收到数据，将数据并行传输给累加器A



第八章 串口通信

- 串口工作流程(接收)
  - 3. 当单片机接收完一帧数据后，会置位RI，触发单片机接收中断
  - 4. 接收中断中通常进行数据处理与储存工作



第八章 串口通信

接口电路

RS-232接口

- 串行数据通信的接口标准，广泛用于计算机串行接口外设连接
- RS232接口定义（以常用的DB9接口为例）。

1 DCD 载波检测

2 RXD 接收数据

3 TXD 发送数据

4 DTR 数据终端准备好

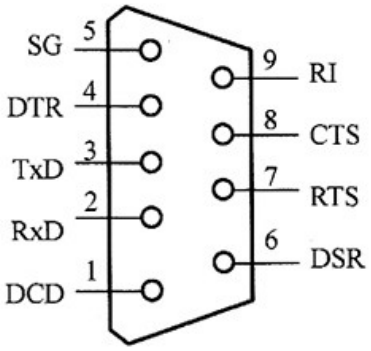
5 SG 信号地

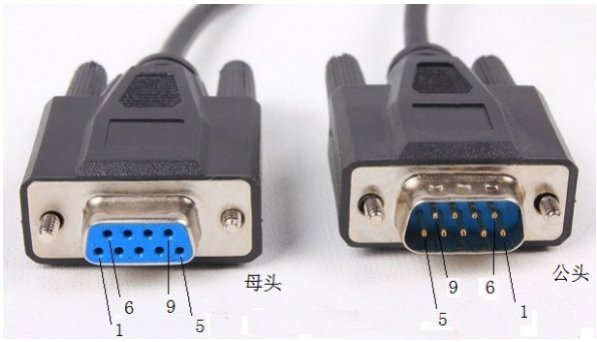
6 DSR 数据准备好

7 RTS 请求发送

8 CTS 清除发送

9 RI 振铃提示



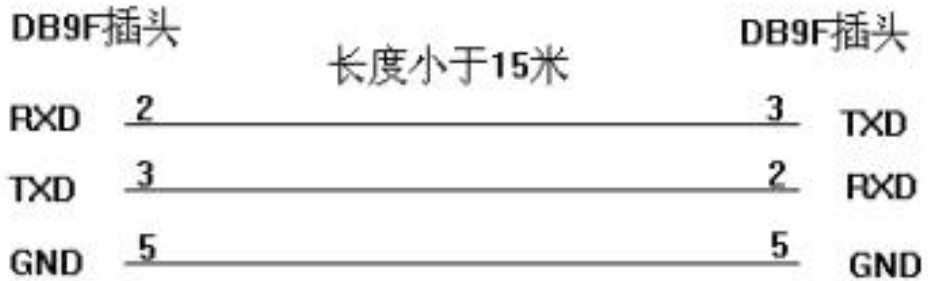


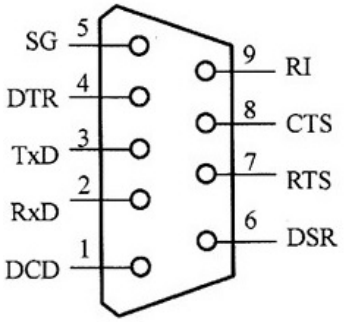
第八章 串口通信

接口电路

RS-232接口

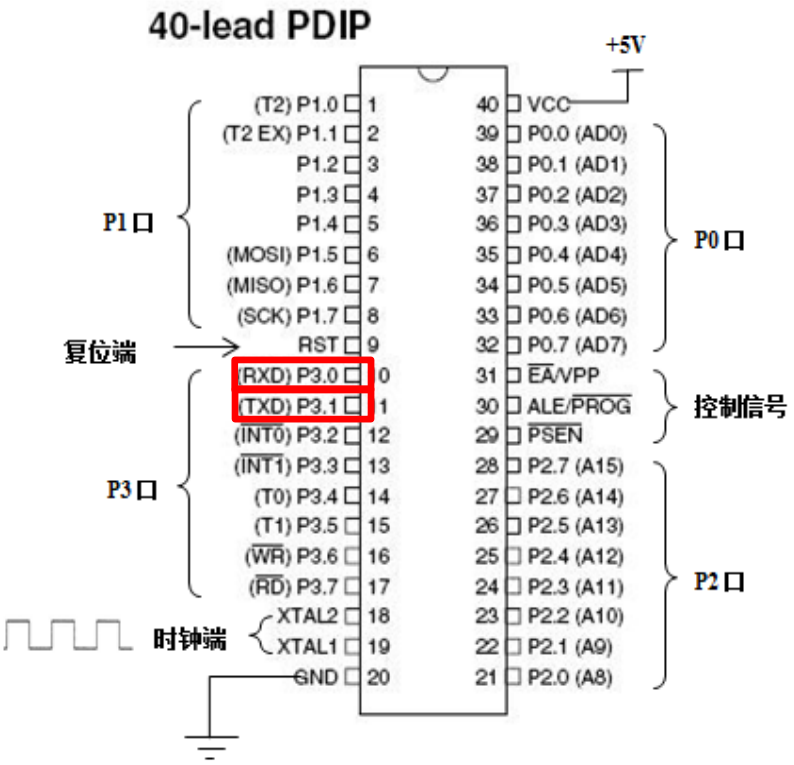
- RS232（DB9）接口有9个引脚，但对于51单片机串口通信而言，常用的引脚只有3个（ 2 RXD 接收数据 / 3 TXD 发送数据/5 SG 信号地 ）
- 两个设备想通过RS-232接口进行通信，只需要将设备1接口RXD/TXD/SG和设备2接口TXD/RXD/SG对应相连，即可进行通信。





第八章 串口通信

- 接口电路
  - 与单片机连接
    - 51单片机有两个与串口通信相关的引脚（P3.0/P3.1）
      - P3.0（RXD）：单片机串口数据接收引脚
      - P3.1（TXD）：单片机串口数据发送引脚
    - 单片机TXD与串口RXD相连，单片机RXD与串口TXD相连，并共地

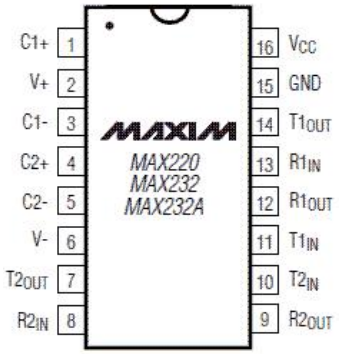


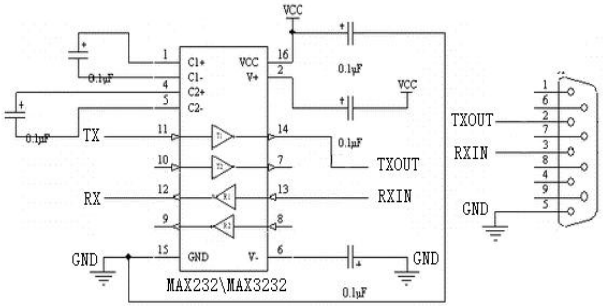
第八章 串口通信

接口电路

与单片机连接

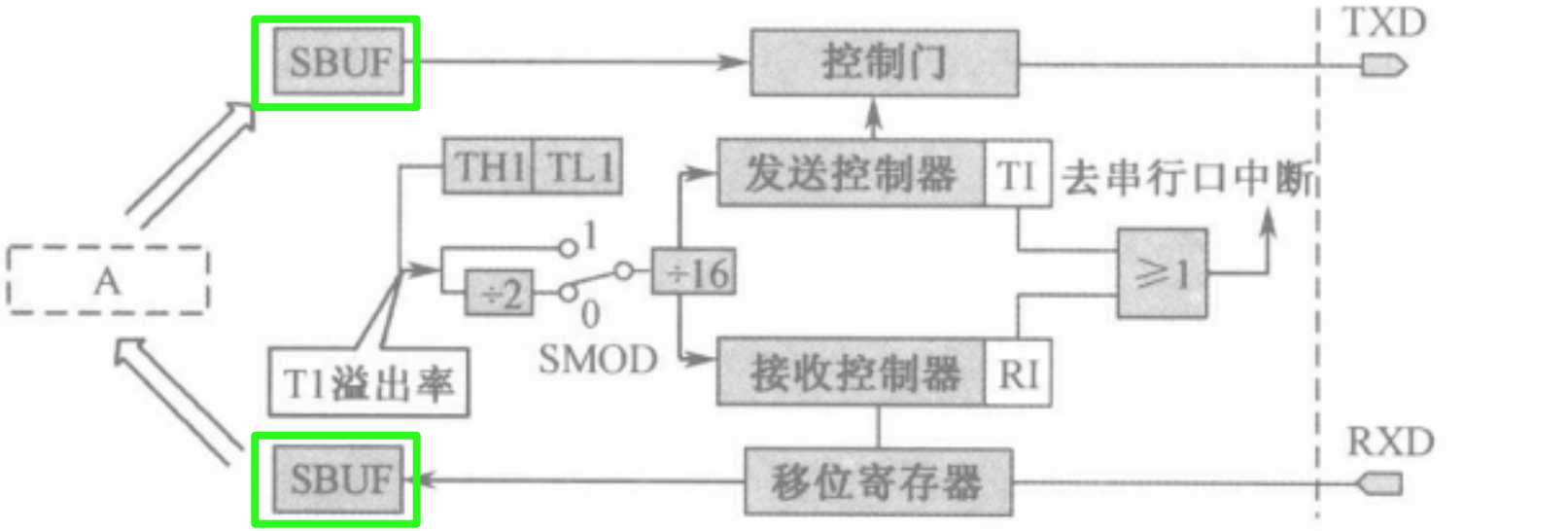
- 51单片机引脚的输出电平为TTL电平（输出低电平<0.8V、高电平>2.4V，输入低电平<1.2V、高电平>2.0V）
- RS232标准电平（高电平电压为-3~-15V，低电平电压为+3~+15V）
- 如果51单片机需要采用RS232标准进行通信，则需要使用相应的电平转换电路，例如MAX232转换电路
  - MAX232芯片是美信（MAXIM）公司专为RS232标准串口设计的单电源电平转换芯片





第八章 串口通信

- 串口寄存器控制
  - SBUF：串行口数据缓冲器
    - 两个物理上独立的接收、发送缓冲器
    - 占用同一地址99H



第八章 串口通信

串口寄存器控制

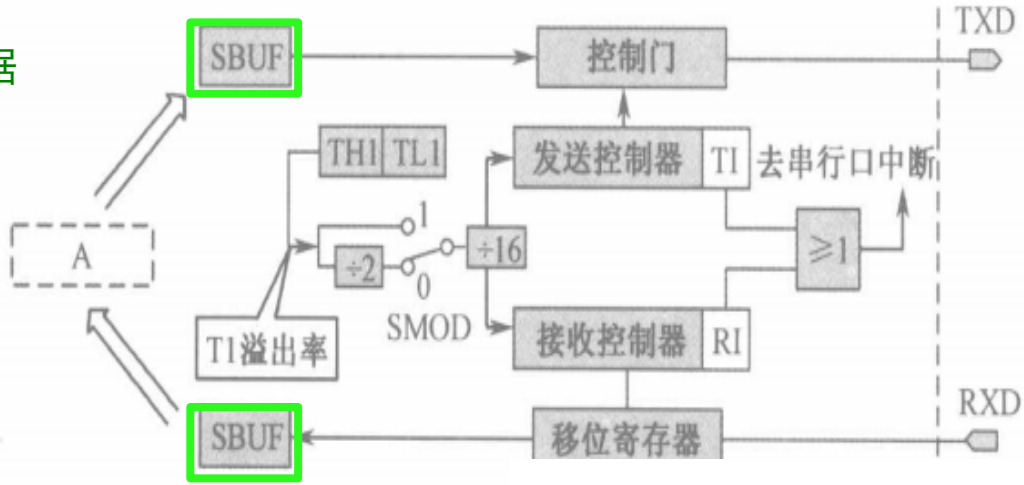
SBUF：串行口数据缓冲器

- 发送时，向SBUF写入数据：

```
SBUF=0x80;  //向SBUF写入发送数据
```

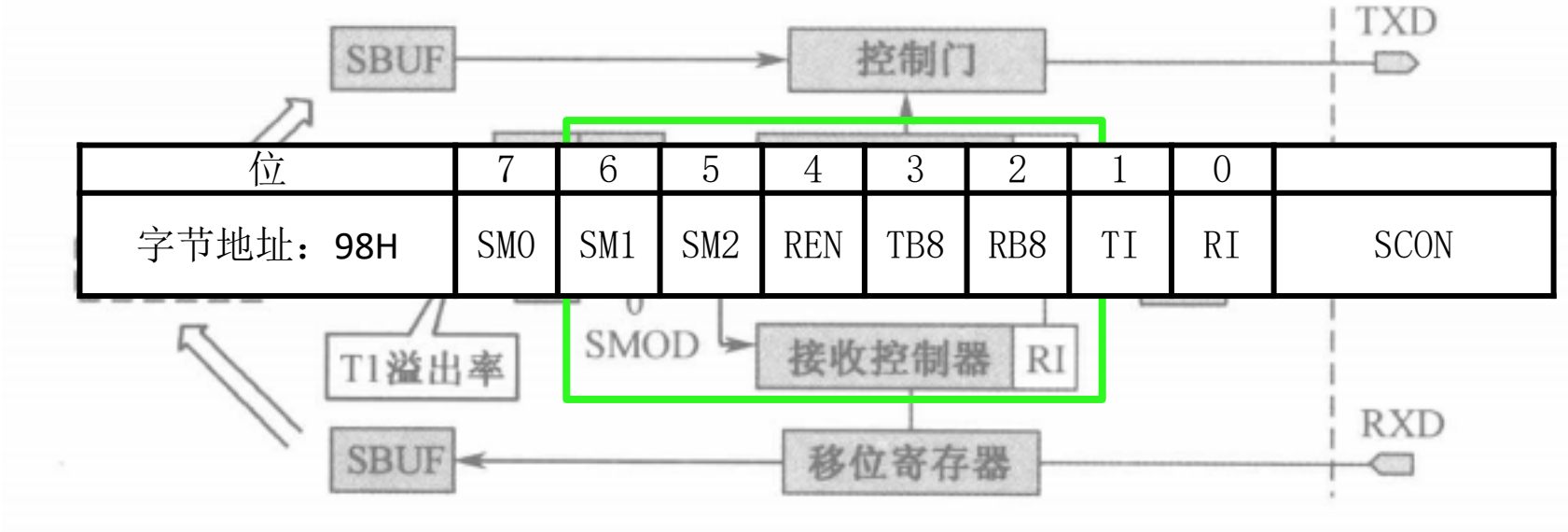
- 接收时，从SBUF中读出数据

```
unsigned char c;
while(!RI);  //等待接收数据
RI=0;  //清标志位
c=SBUF;  //读取数据
```



第八章 串口通信

- 串口寄存器控制
  - SCON：串行口控制寄存器
    - 设定串行口工作方式、接收/发送控制及设置状态标志



第八章 串口通信

■ 串口寄存器控制

➢ SCON

- SM0, SM1：工作方式设置位，可选择四种工作方式：

SM0	SM1	方式	说明	波特率
0	0	0	移位寄存器	fcos/12
0	1	1	10位异步收发器（8位数据）	可变
1	0	2	11位异步收发器（9位数据）	fcos/64 or fcos/32
1	1	3	11位异步收发器（9位数据）	可变

位	7	6	5	4	3	2	1	0	
字节地址：98H	SM0	SM1	SM2	REN	TB8	RB8	TI	RI	SCON

第八章 串口通信

串口控制寄存器

SCON

- SM2：多机通信控制位，主要用于方式2和方式3。当接收机的SM2=1时可以利用收到的RB8来控制是否激活RI。当SM2=0时，不论收到的RB8为0和1，均可以使收到的数据进入SBUF，并激活RI
- REN：允许串行接收位。由软件置REN=1，则启动串行口接收数据；若软件置REN=0，则禁止接收

位	7	6	5	4	3	2	1	0	
字节地址：98H	SM0	SM1	SM2	REN	TB8	RB8	TI	RI	SCON

第八章 串口通信

串口控制寄存器

SCON

- TB8：在方式2或方式3中，是发送数据的第九位，可以用软件规定其作用。可以用作数据的奇偶校验位，或在多机通信中，作为地址帧/数据帧的标志位
- RB8：在方式2或方式3中，是接收到数据的第九位，作为奇偶校验位或地址帧/数据帧的标志位。在方式1时，若SM2=0，则RB8是接收到的停止位

位	7	6	5	4	3	2	1	0	
字节地址：98H	SM0	SM1	SM2	REN	TB8	RB8	TI	RI	SCON

第八章 串口通信

■ 串口控制寄存器

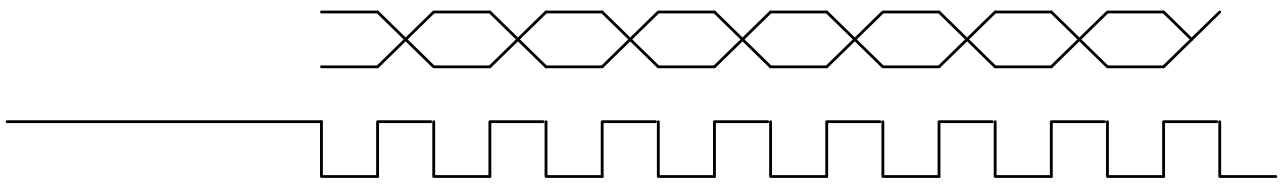
➢ SCON

- T1：发送中断标志位。在方式0时，当串行发送第8位数据结束时，或在其它方式，串行发送停止位的开始时，由内部硬件使TI置1，向CPU发中断申请。在中断服务程序中，必须用软件将其清0，取消此中断申请
- R1：接收中断标志位。在方式0时，当串行接收第8位数据结束时，或在其它方式，串行接收停止位的中间时，由内部硬件使RI置1，向CPU发中断申请。也必须在中断服务程序中，用软件将其清0，取消此中断申请

位	7	6	5	4	3	2	1	0	
字节地址：98H	SM0	SM1	SM2	REN	TB8	RB8	TI	RI	SCON

第八章 串口通信

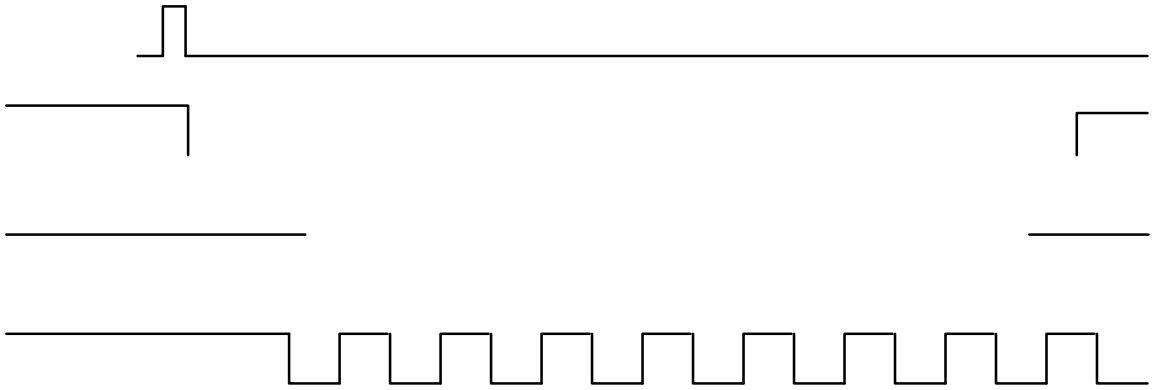
- 串口工作方式0(SM0=0，SM1=0)
  - 方式0：串行口为同步移位寄存器的输入输出方式。主要用于扩展并行输入或输出口。数据由RXD（P3.0）引脚输入或输出，同步移位脉冲由TXD（P3.1）引脚输出。发送和接收均为8位数据，低位在先，高位在后。波特率固定为fosc/12
  - 方式0发送时序：



SM0	SM1	方式	说明	波特率
0	0	0	移位寄存器	fcos/12
0	1	1	10位异步收发器（8位数据）	可变
1	0	2	11位异步收发器（9位数据）	fcos/64 or fcos/32
1	1	3	11位异步收发器（9位数据）	可变

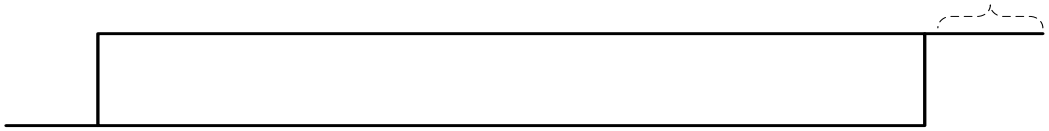
第八章 串口通信

- 串口工作方式0(SM0=0，SM1=0)
  - 方式0接收时序：



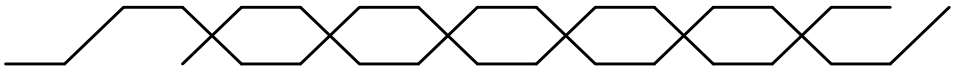
第八章 串口通信

- 串口工作方式1(SM0=0，SM1=1)
  - 方式1：方式1是10位数据的异步通信口。TXD为数据发送引脚，RXD为数据接收引脚。一帧数据中有1位起始位，8位数据位，1位停止位
  - 方式1数据格式：



第八章 串口通信

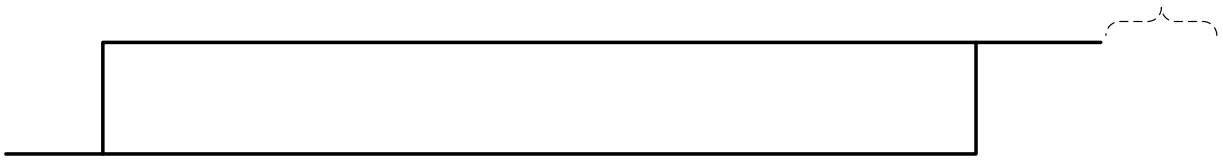
- 串口工作方式1(SM0=0，SM1=1)
  - 方式1发送时序：



  - 方式1接收时序：

第八章 串口通信

- 串口工作方式2(SM0=1，SM1=0)
  - 方式2：方式2为11位数据的异步通信口。TXD为数据发送引脚，RXD为数据接收引脚 。方式2一帧数据中有数据9位（含1位附加的第9位，发送时为SCON中的TB8，接收时为RB8），停止位1位。方式2的波特率固定为晶振频率的1/64或1/32
  - 方式2数据格式：



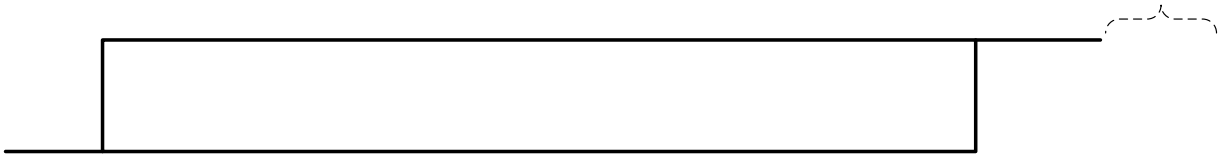
第八章 串口通信

- 串口工作方式2(SM0=1，SM1=0)
  - 方式2发送时序：

  - 方式2接收时序：

第八章 串口通信

- 串口工作方式3(SM0=1，SM1=1)
  - 方式3：方式3为11位数据的异步通信口。TXD为数据发送引脚，RXD为数据接收引脚 。方式2—帧数据中有数据9位（含1位附加的第9位，发送时为SCON中的TB8，接收时为RB8），停止位1位。方式3的波特率由定时器T1的溢出率决定。
  - 方式3数据格式：



第八章 串口通信

- 串口工作方式3(SM0=1，SM1=1)
  - 方式3发送时序：

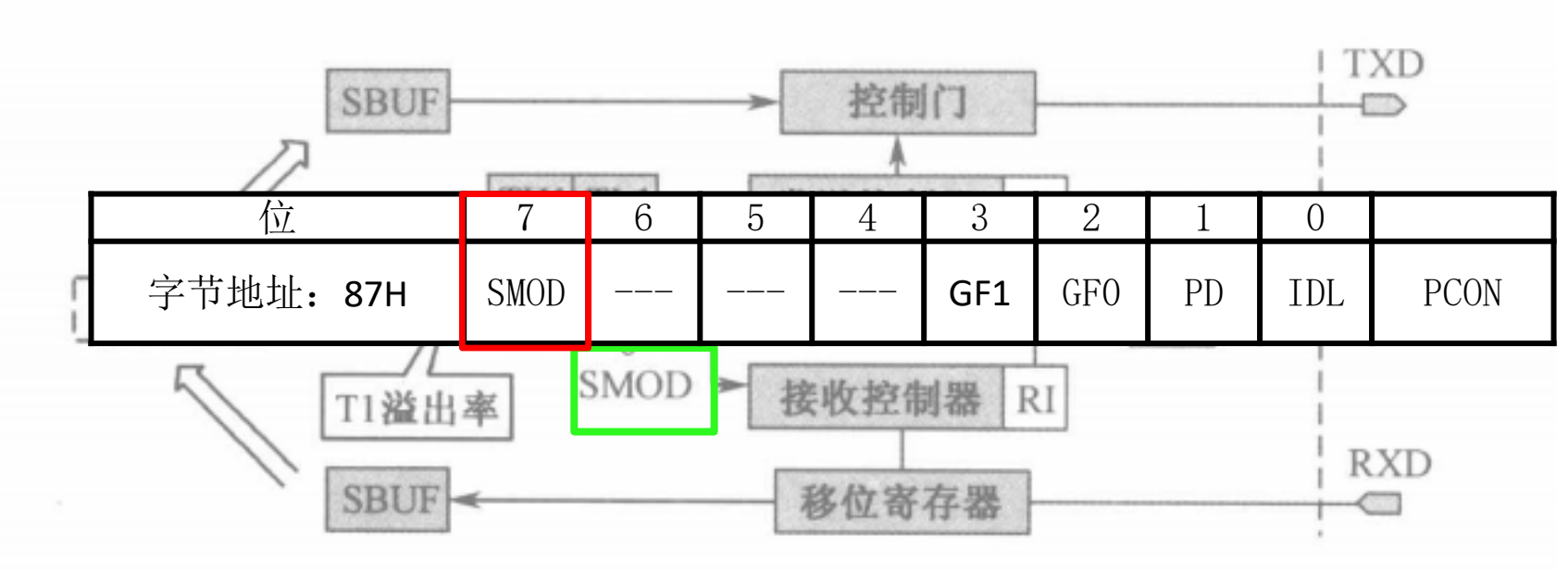
  - 方式3接收时序：

第八章 串口通信

■ 串口寄存器控制

➢ PCON：电源控制寄存器

- PCON中只有SMOD位与串口通信相关。SMOD=1时，波特率加倍。SMOD=0时，波特率不加倍。

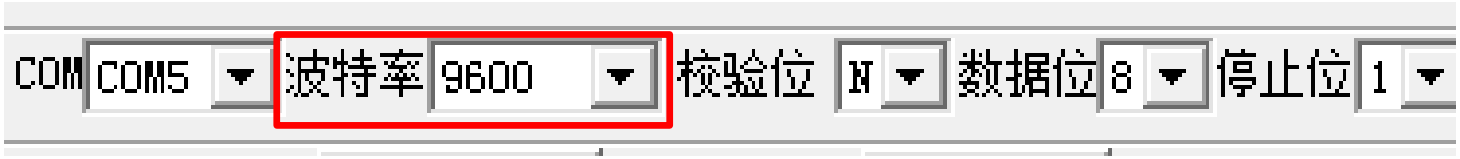


第八章 串口通信

SM0	SM1	方式	说明	波特率
0	0	0	移位寄存器	fcos/12
0	1	1	10位异步收发器（8位数据）	可变
1	0	2	11位异步收发器（9位数据）	fcos/64 or fcos/32
1	1	3	11位异步收发器（9位数据）	可变

串口波特率计算

- 回顾波特率的概念：
  - 代表单片机或计算机在串口通信的速率
  - 收发两方波特率须一致
- 计算：四种工作方式对应三种波特率。
  - 方式0波特率 = fosc/12
  - 方式2的波特率 =（$2^{SMOD}/64$）· fosc
  - 方式1的波特率 =（$2^{SMOD}/32$）·（T1溢出率）
  - 方式3的波特率 =（$2^{SMOD}/32$）·（T1溢出率）



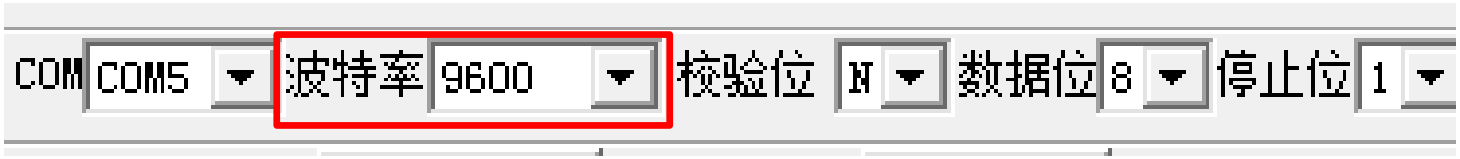
第八章 串口通信

SM0	SM1	方式	说明	波特率
0	0	0	移位寄存器	fcos/12
0	1	1	10位异步收发器（8位数据）	可变
1	0	2	11位异步收发器（9位数据）	fcos/64 or fcos/32
1	1	3	11位异步收发器（9位数据）	可变

串口波特率计算

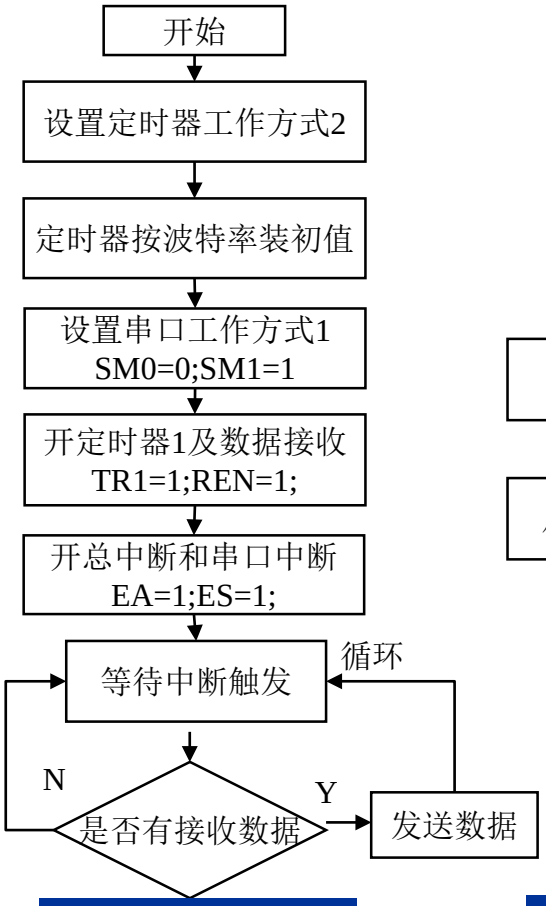
- 例：单片机和PC上位机都设置波特率为9600，采用工作方式1，SM0/SM1设置为0/1（取SMOD = 0）
- 方式1的波特率 =（$2^{SMOD}/32$）·（T1溢出率）
- 代入得到T1溢出率= 307200
- 由于T1溢出率 = 11.0592MHz/([256-X]*12)

  （11.0592MHz是系统晶振频率，其的倒数是时钟周期。计数器的计数周期是机器周期。机器周期 = 12*时钟周期）
- 从而X=253，TH1=0XFD，TL1=0XFD。

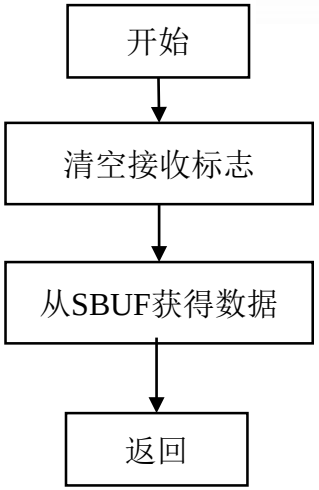


第八章 串口通信

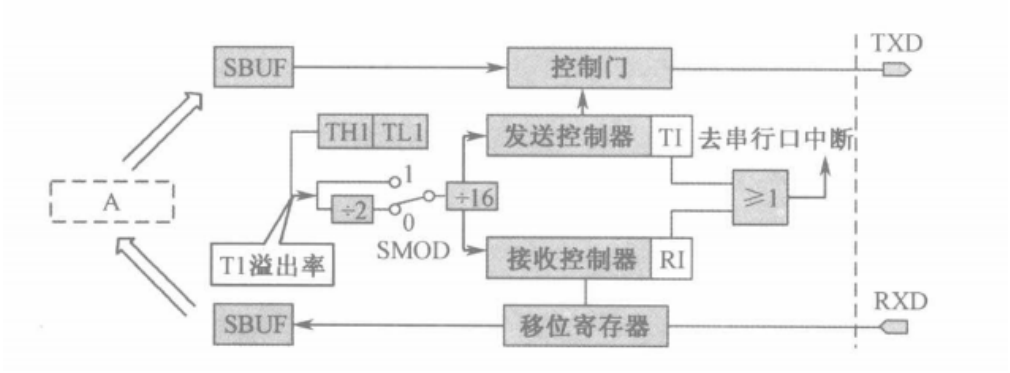
流程图

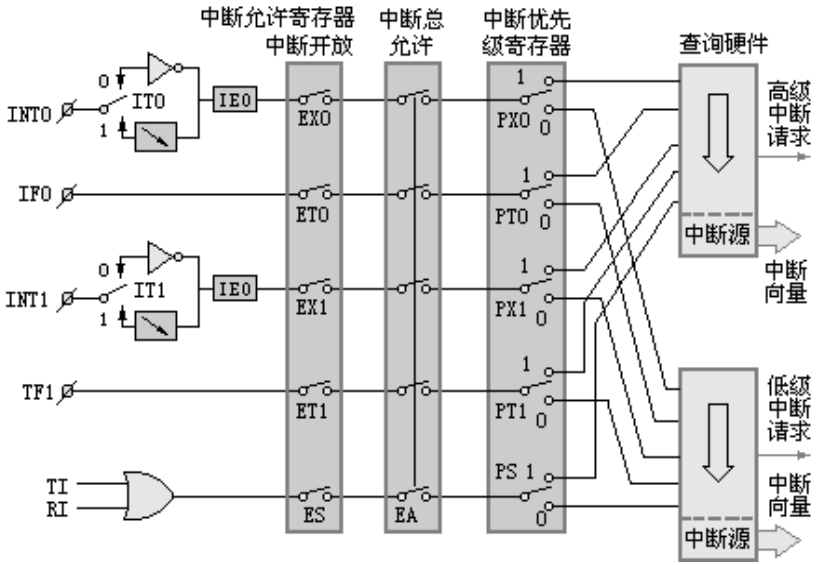


单片机主程序



中断子程序

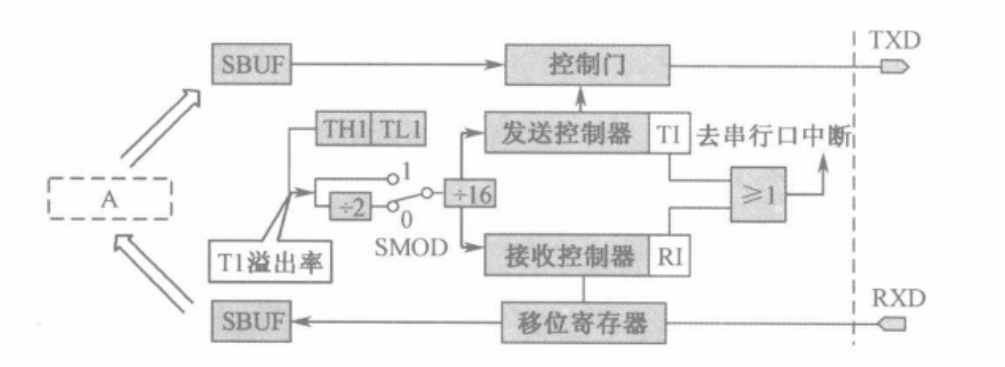


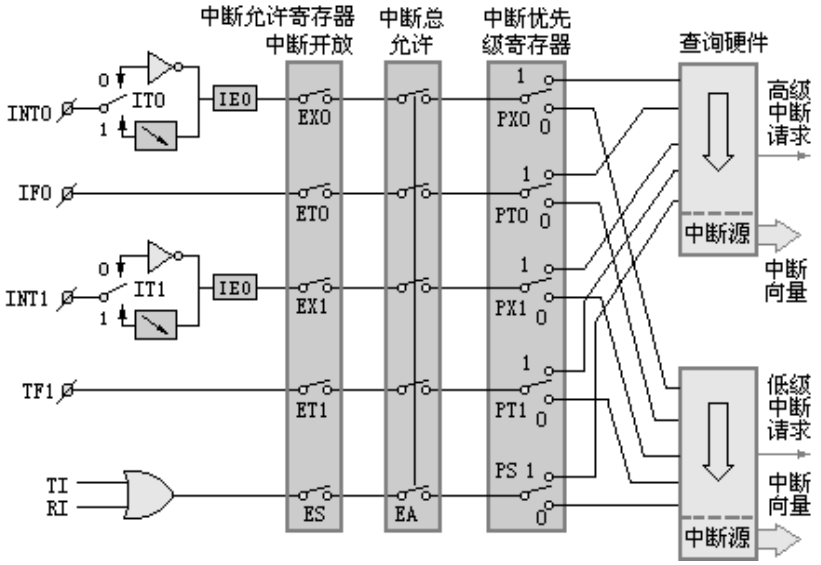


第八章 串口通信

程序代码

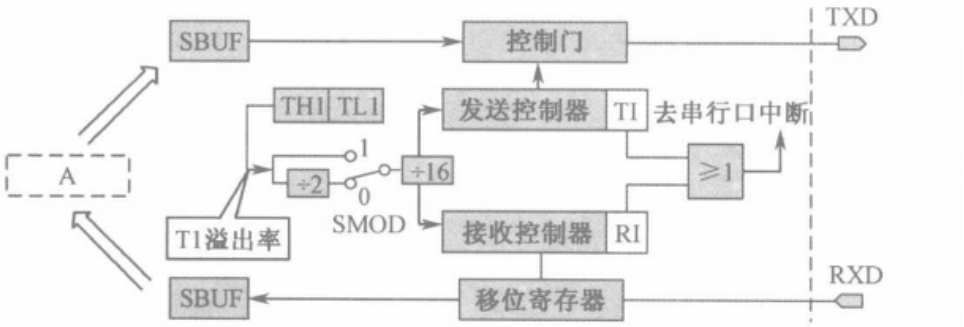
```
#include <reg52.h>
#define uchar unsigned char
#define uint unsigned int
uchar flag,a,i;
uchar code table[]="i get ";
void init()  //定时器初始化函数
{
    TMOD=0x20;//设置定时器工作方式
    TH1=0xfd; //预装初值
    TL1=0Xfd;
    TR1=1;   //开定时器1
    REN=1;   //开启数据接收
    SM0=0;
    SM1=1;   //设置串口工作方式
    EA=1;    //开总中断
    ES=1;    //开串口中断
}
```





第八章 串口通信

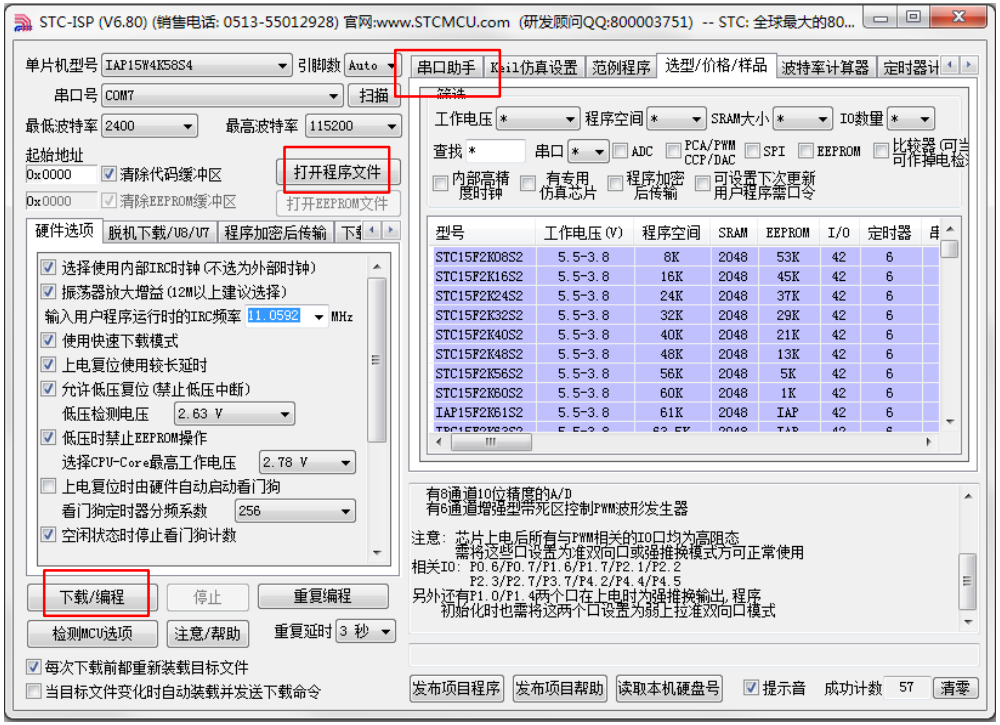
```
void ser()interrupt 4 //中断服务函数
{
  RI=0;
  a=SBUF;  //将数据赋值给a
  flag=1;  //置位标志位表示收到
}
void main() //主程序
{
  init();  //初始化
  while(1)
  {
   if(flag==1)//判断是否收到数据
   {
    ES=0;
    for(i=0;i<6;i++)//循环输出字符串
    {
     SBUF=table[i];  //将字符写入缓冲区
     while(!TI);   //等待发送结束
     TI=0;  //清空发送结束标志位
     ES=1;
    }
```




```
    SBUF=a;  //将收到的数据写入缓冲区
    while(!TI);  //等待发送结束
    TI=0; //清空发送结束标志位
    ES=1;
    flag=0;  //清空标志位
   }
  }
}
```

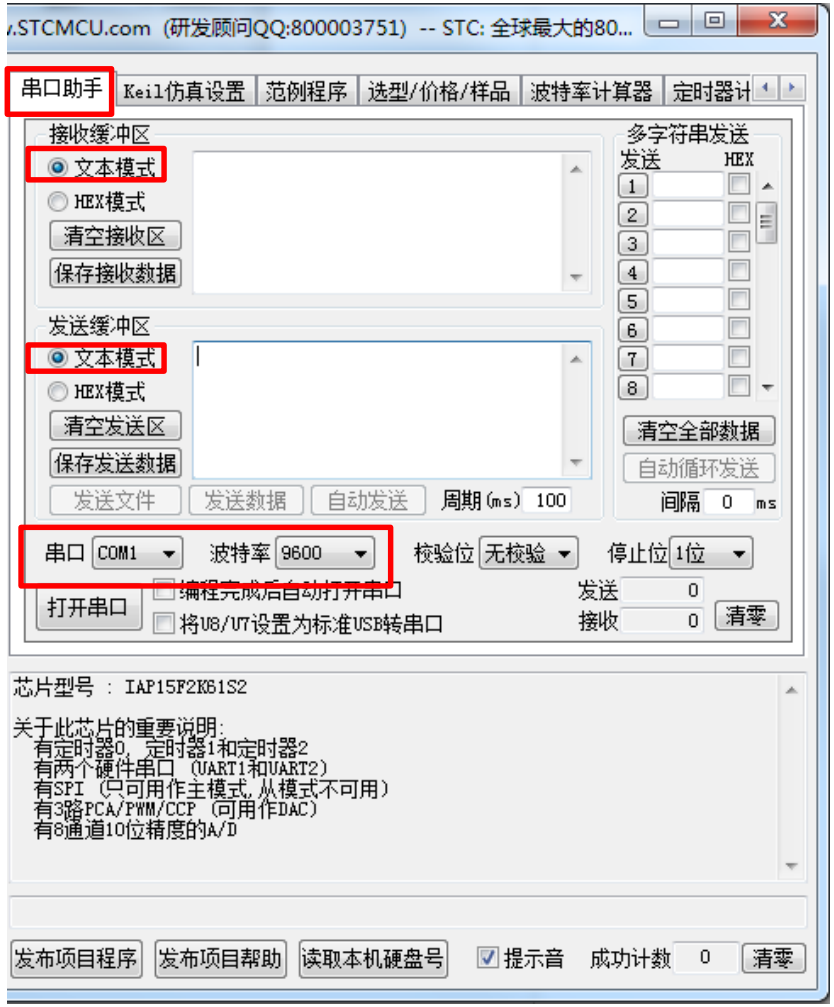
第八章 串口通信

- 烧写
  - 打开STC-ISP软件，选择单片机型号以及与电脑相连的COM口，选择hex文件，点击下载，同时重新启动单片机



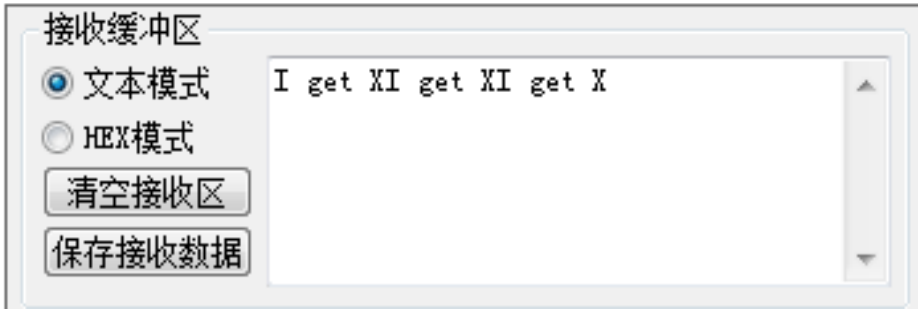
第八章 串口通信

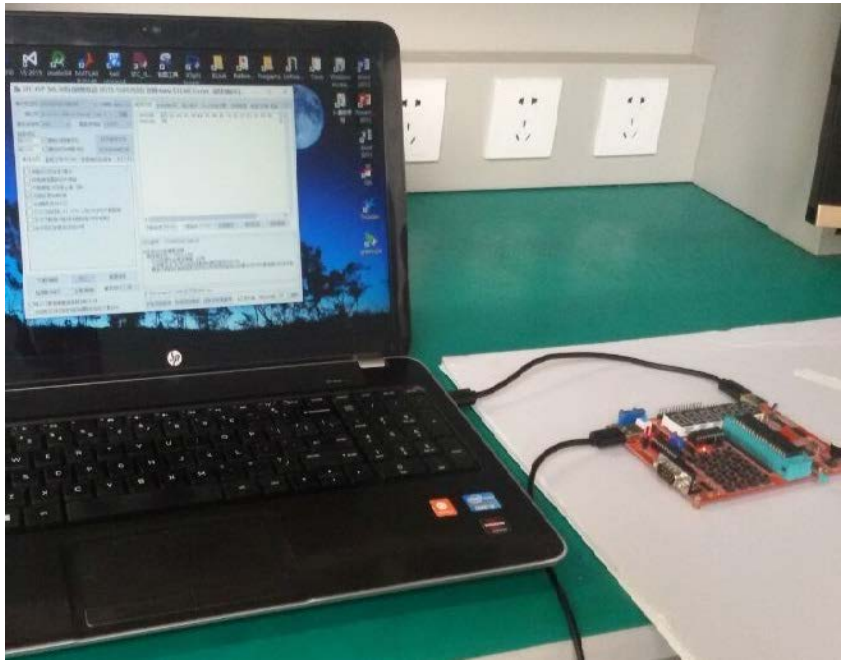
- 调试
  - 打开软件右上角串口助手
  - 选择正确的串口号、波特率以及数据格式
  - 在接收区和发送区都选择按文本/字符模式显示
  - 在发送区输入‘X’
  - 点击发送数据/字符并观察现象



第八章 串口通信

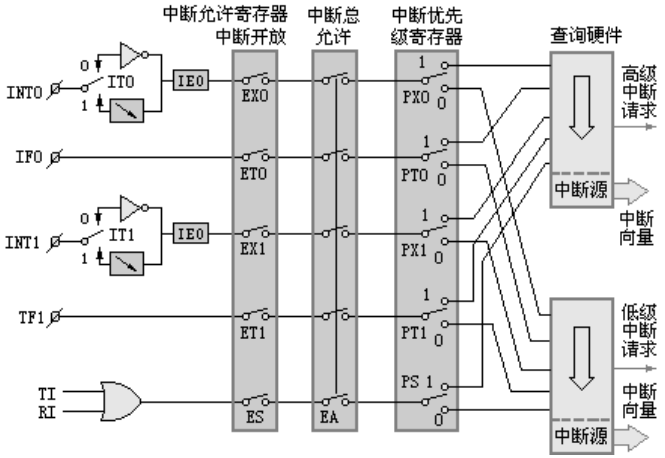
- 烧写
  - 观察现象：

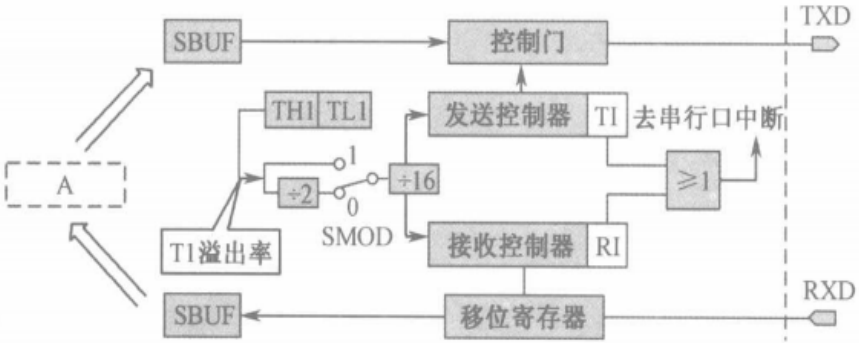




小结

- 串口通信
  - 概念及工作流程
  - 接口电路
- 串口控制寄存器
  - SBUF：数据缓冲器
  - SCON：工作方式选择
    - 波特率计算
- 调试
  - 串口助手的使用





再见