



微机原理和接口技术

第十七讲 串行通信2



提 纲

1. 总线与通信

6. UART的应用-1

2. 通信协议与校验方式

7. UART的应用-2

3. UART的组成结构

8. RS232/RS485通信技术与应用

4. UART的工作方式

5. UART的波特率

提 纲

4. UART的工作方式

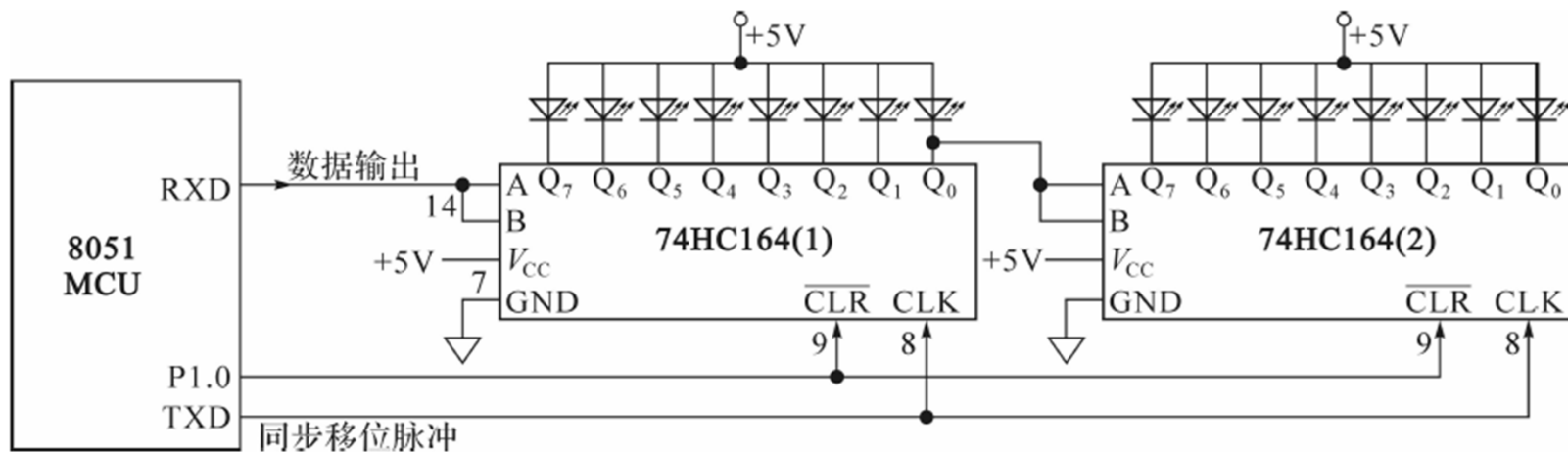
UART的工作方式

8051 MCU的UART 有4种工作方式，通过SCON中的SM0、SM1选择。

1. 方式0：同步移位寄存器输入/输出方式

- 8位数据为一帧，发送次序为从低到高（D0到D7）；
- 引脚RXD是数据输入/输出端；引脚TXD是同步移位脉冲输出端，为外围芯片（如“串入并出”或“并入串出”移位寄存器）提供同步移位信号。
- 移位脉冲的频率是固定的为 $f_{osc}/12$ ，即每个机器周期发送/接收1位数据。

方式0实际为UART的同步串行通信方式，常用于I/O接口的扩展。





UART的工作方式

1. 方式0：同步移位寄存器输入/输出方式

(1) 发送过程

- 在TI=0情况下，将要发送的数据写入**发送SBUF**，即启动了一次发送。
- 数据从RXD端串行输出（先低位后高位），TXD端输出移位同步信号。
- 8位发送完毕后，由硬件将TI 置为 “1” （表示一个字节数据发送完毕）。在软件清除TI后，可发送下一个数据。

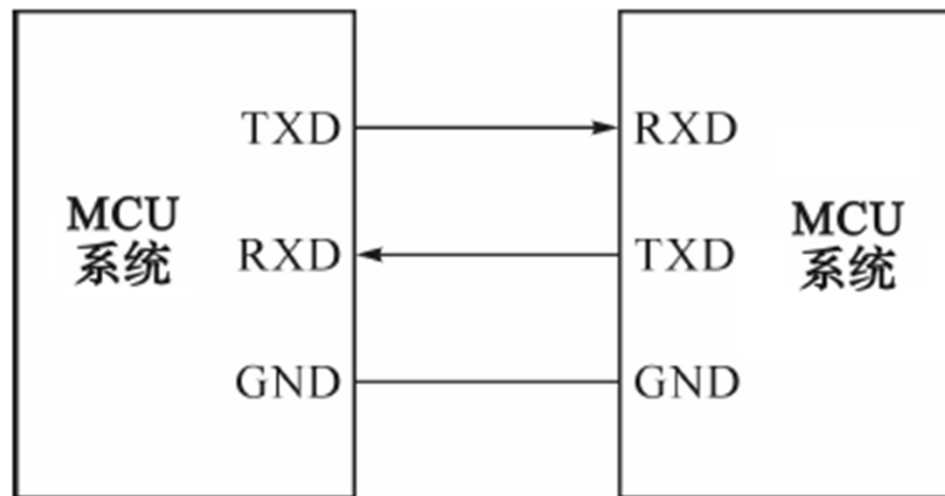
(2) 接收过程

- 在RI = 0条件下，将REN置 “1” 便启动了串行口的接收；
- RXD为数据输入端，TXD为同步移位信号输出端。
- 在接收到8位数据后，将被移入**接收SBUF**，并置RI 为 “1” （表示接收到一个字节数据）。在CPU读取该数据、清除RI 后，可接收下一个数据。

UART的工作方式

2. 方式1-3：异步通信方式

工作方式	数据帧	波特率
方式1：10位UART	起始位、数据位、停止位	波特率 = $2^{\text{SMOD}} / 32 \times (\text{T1的溢出率})$
方式2：11位UART	起始位、数据位、校验位 (第8位)、停止位	$2^{\text{SMOD}} \times f_{\text{osc}} / 64$
方式3：11位UART		波特率 = $2^{\text{SMOD}} / 32 \times (\text{T1的溢出率})$



TXD：发送引脚，RXD：接收引脚；
空闲（即不通信）时，收发引脚
均为高电平。

异步通信连接方式



UART的工作方式

2. 方式1-3：异步通信方式

(1) 发送过程

以11位数据帧发送为例：

- 将要发送的第8位数据写入SCON的TB8；
- 然后将要发送的数据写入**发送SBUF**，即启动了一次发送；
- 依次发送：起始位“0”（硬件自动插入）、SBUF中的8位数据位（先低后高）、TB8的内容、停止位“1”（硬件自动插入）；
- 发送速率：由设置的波特率决定，即数据帧信息在内部波特率发生器输出的移位脉冲控制下，逐位从TXD引脚输出；
- 当停止位发送完毕后，置TI=1，通知CPU可以发送下一帧数据；同时维持TXD引脚为高电平状态。



UART的工作方式

2. 方式1-3: 异步通信方式

(2) 接收过程:

令REN=1, 接收器就开始接收数据。接收步骤:

- 检测起始位: 在内部接收脉冲 (16倍的波特率) 的控制下, 检测RDX引脚的状态; 检测到有效的“0”, 即认为检测到了起始位。
- 于是开始连续接收数据位、校验位和停止位。
- 若接收的停止位 = 1, 表示接收的数据有效, 就将移位寄存器中的数据移入**接收SBUF**, 并将 RI置为1。不然, 所有接收信息将丢失。

数据的采样与确定方法:

- 以16倍波特率的速率检测RXD端的电平, 即1位数据检测16次;
- 取中间7、8、9三个点的电平, 并把2次及以上相同的电平确定为有效的数据电平 (即按少数服从多数的规则确定出该bit的电平)。

UART的工作方式

2. 方式1-3：异步通信方式

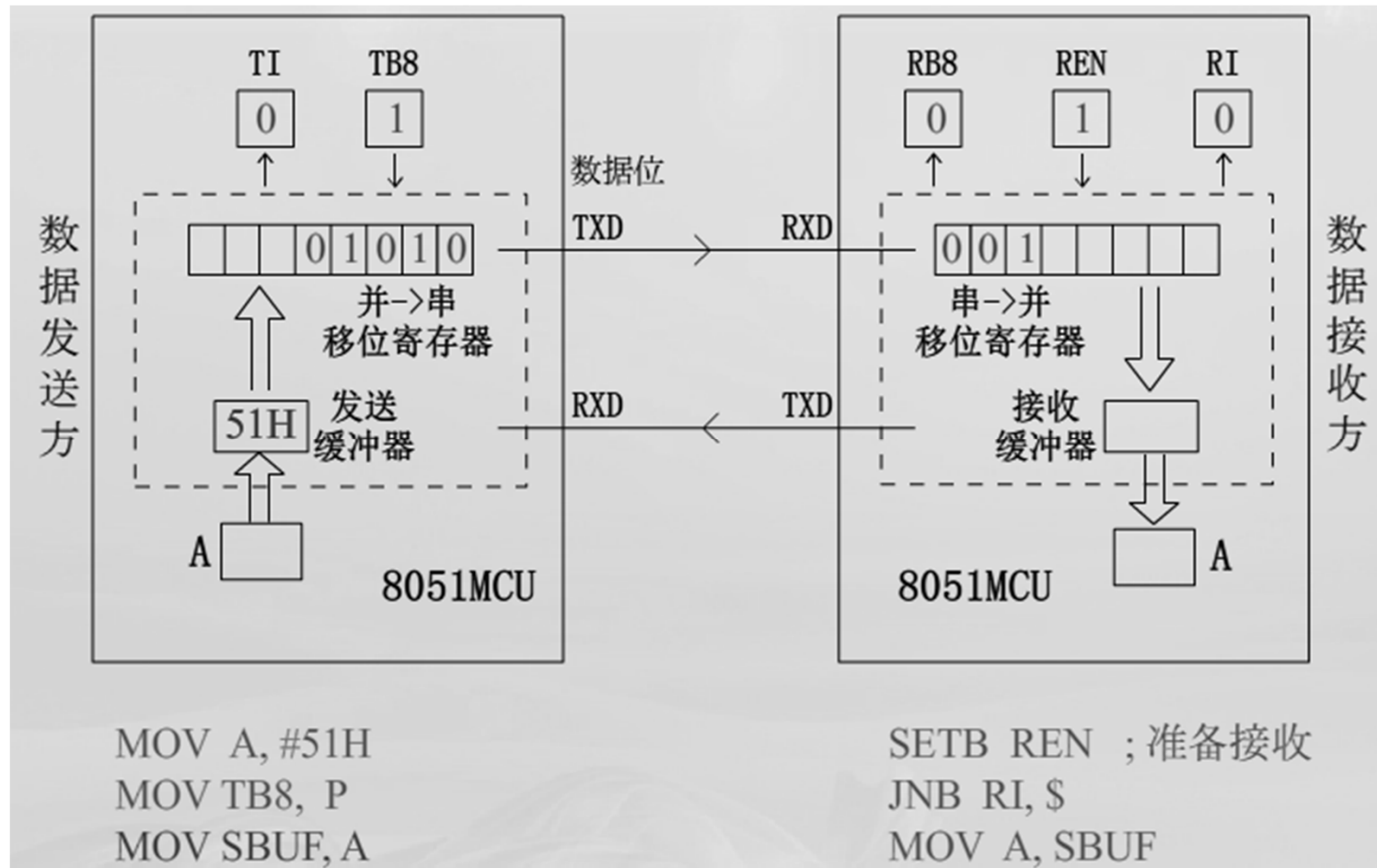
发送第8位（TB8）的确定：

发送数据	校验方式		数据的P 标志位	TB8	
	偶校验位	奇校验位		偶校验时	奇校验时
33H	0	1	0	0	1
43H	1	0	1	1	0
9EH	1	0	1	1	0
AAH	0	1	0	0	1

偶校验时：TB8=P;

奇校验时：TB8= \overline{P} 。

UART的工作方式



右边要先准备好

提 纲

5. UART的波特率



UART的波特率

1. 四种工作方式的波特率

➤ **方式0的波特率**是固定的，波特率 = $f_{osc}/12$ ，其周期即为机器周期。

➤ **方式2的波特率**有两种选择：

$$\text{波特率} = 2^{SMOD}/64 \times f_{osc}.$$

- 置PCON中的SMOD = 0，则波特率为 $f_{osc}/64$ ；
- 置PCON中的SMOD = 1，则波特率为 $f_{osc}/32$ 。

➤ **方式1和方式3的波特率**是可编程的，由T1或T2的溢出率控制。（对于经典8051 MCU默认T1作为波特率发生器）

$$\text{波特率} = 2^{SMOD}/32 \times (\text{T1溢出率})$$

- 当SMOD = 0，波特率为 $1/32 \times (\text{T1溢出率})$ ；
- 当SMOD = 1，波特率= $1/16 \times (\text{T1溢出率})$ 。



UART的波特率

2. 定时器T1作为波特率发生器

T1作波特率发生器时，应工作在定时方式2，即8位自动重装载方式，禁止中断。设初值为X，则T1的定时时间（溢出周期）T为：

$$T = \frac{12}{f_{osc}} \times (2^8 - X)$$

溢出率为溢出周期的倒数，所以

$$\text{波特率} = \frac{2^{SMOD}}{32} \times T1\text{溢出率} = \frac{2^{SMOD}}{32} \times \frac{f_{osc}}{12 \times (2^8 - X)}$$

则T1的定时初值X为：

$$X = 2^8 - \frac{f_{osc} \times (SMOD + 1)}{384 \times \text{波特率}}$$



UART的波特率

2. 定时器T1作为波特率发生器

常用的波特率有：1200、2400、4800、9600、38400、115200bps等。

波特率与定时器T1初值的对应表

常用波特率	Fosc (MHz)	SMOD	TH1初值
115200	11.0592	1	FFH
38400	11.0592	1	FEH
19200	11.0592	1	FDH
9600	11.0592	0	FDH
4800	11.0592	0	FAH
2400	11.0592	0	F4H
1200	11.0592	0	E8H

提 纲

6. UART的应用-1

UART的应用-1

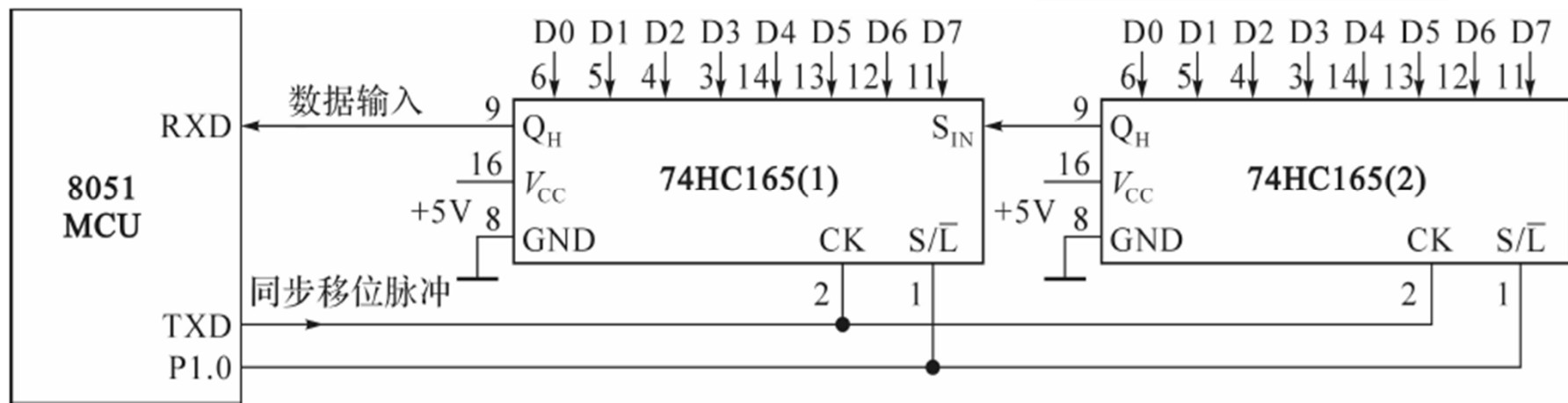
1. 利用串行口扩展I/O接口

例7-2：用2个8位并入串出移位寄存器74HC165扩展2个并行输入接口。编程实现从2个输入口读入20H字节数据，并保存到内部RAM的50H-6FH中。

分析：74HC165是并入串出移位寄存器，D0-D7为并行输入端， Q_H 为串行输出端，CK为移位脉冲输入端， S/\bar{L} 是串行移位/并行数据置入选择端。前级的输出端 Q_H 与后级的输入端 S_{IN} 相连，实现多片74HC165的串接。

$S/\bar{L}=0$ ：并行数据置入

$S/\bar{L}=1$ ：串行移位





UART的应用-1

汇编程序:

```
MAIN: MOV R7, #10H           ; 设置读入字节数
      MOV R0, #50H           ; 设片内RAM指针
      SETB F0                ; 设置奇数次/偶数次读入标志
RCV0: CLR P1.0                ; 允许并行置入数据, 此时165引脚D7-D0上状态存入寄存器
      SETB P1.0              ; 允许串行移位
RCV1: MOV SCON,#10H           ; 设串行口方式0, 启动接收
      JNB RI,$                ; 等待接收到1帧数据
      CLR RI                  ; 清接收中断标志位
      MOV A,SBUF              ; 读取SBUF中数据
      MOV @R0, A              ; 保存到内部RAM
      INC R0
      CPL F0                  ; F0求反。F0=1 (奇数次) 读入165 (1) 的内容,
                                ; F0=0 (偶数次) 读入165 (2) 的内容
      JNB F0, RCV1            ; 再接收1个寄存器的数据
      DJNZ R7, RCV0           ; 判断是否已读入预定的字节数
      .....                  ; 对读入数据进行处理
```

UART的应用-1

2. 利用串行口进行异步通信

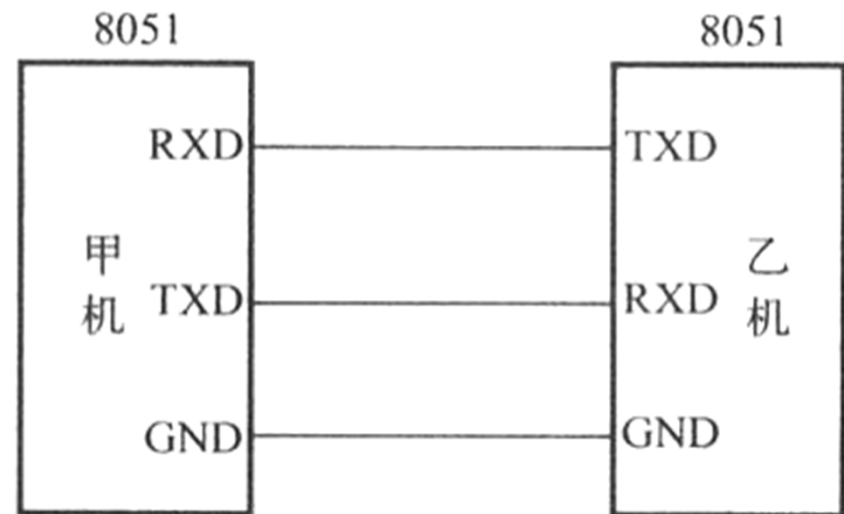
例7-4：点对点双机通信。甲机将50-5FH的16个数据串行发送给乙机。

分析：

- 采用方式3，偶校验方式。波特率为1200bps，波特率发生器T1的重装载常数为E8H。
- 在数据写入SBUF前，先将校验位写入TB8，即准备好要发送的第8位。

➤ 线路连接方式：

- 甲机的TXD与乙机的RXD相连；
- 甲机的RXD与乙机的TXD相连；
- 地线与地线相连。



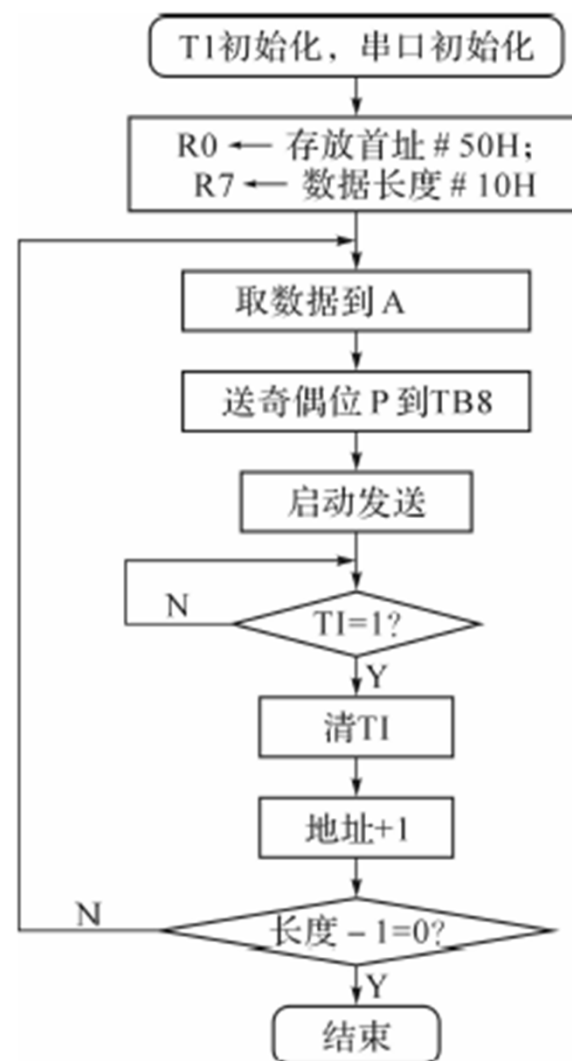
UART的应用-1

2. 利用串行口进行异步通信

(1) 发送程序 (汇编)

```

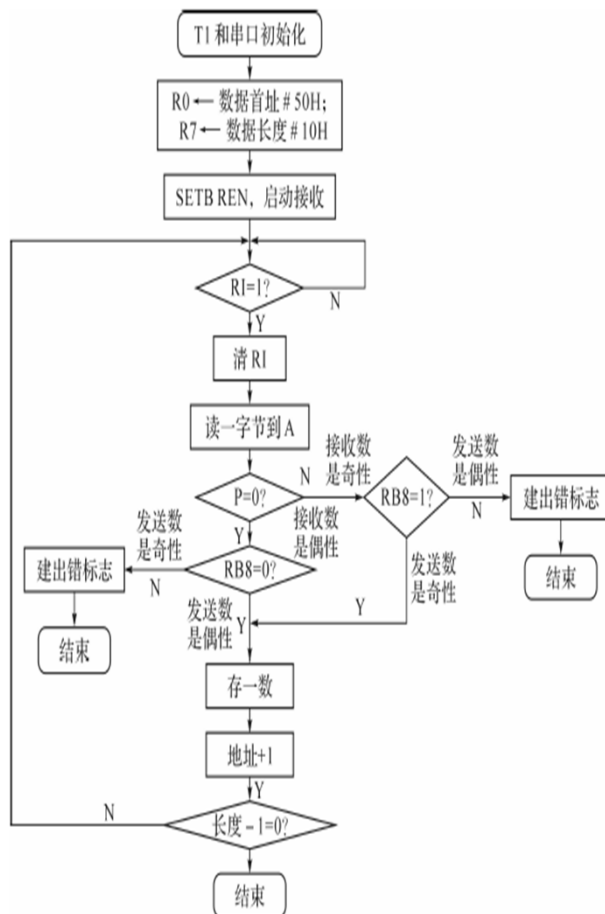
TR:    MOV    SCON,#0C0H    ; 串口初始化, 方式3
        MOV    TMOD,#20H    ; T1初始化, 方式2
        CLR    TR1
        MOV    TH1,#0E8H    ; 装载初值
        MOV    TL1,#0E8H
        SETB   TR1          ; 启动波特率发生器
        MOV    R0,#50H      ; 首地址赋给R0
        MOV    R7,#10H      ; 设置数据长度
LOOP1:  MOV    A,@R0         ; 将发送内容存于A中
        MOV    C,P          ; 校验位传送到TB8
        MOV    TB8,C
        MOV    SBUF,A       ; 启动串行发送
WAIT:   JNB    TI,WAIT       ; 等待发送完毕
        CLR    TI           ; TI 清零
        INC    R0           ; 修改指针
        ; 这里可以加一点延时, 更保险
        DJNZ   R7,LOOP1      ; 未完继续
        RET
    
```



UART的应用-1

2. 利用串行口进行异步通信

(2) 接收程序 (汇编)



RVE:	MOV	TMON,#20H ;T1为方式2
	CLR	TR1
	MOV	TH1,#0E8H
	MOV	TL1,#0E8H
	SETB	TR1 ;启动波特率发生器工作
	MOV	R0,#50H ;接收数据存储首地址
	MOV	R7,#10H ;设置接收数据长度
	MOV	SCON,#0D0H ;方式3,REN=1 (启动接收)
W1:	JNB	RI,W1 ;等待接收完毕
	CLR	RI ;R1软件清零
	MOV	A,SBUF ;读数据到A, P 反映该数据的奇偶性
	JNB	PSW.0,NEXT ;接收数据的 P=0, 跳转
	JNB	RB8,ERROR ;接收的第8位=0, 校验出错, 跳转
	SJMP	RIGHT
NEXT:	JB	RB8,ERROR ;校验出错, 转移
RIGHT:	MOV	@R0,A ;数据传送正确
	INC	R0
	DJNZ	R7,W1
	CLR	F0 ;置正确标志
	RET	
ERROR:	SETB	F0 ;置出错标志
	RET	

Thank you!

