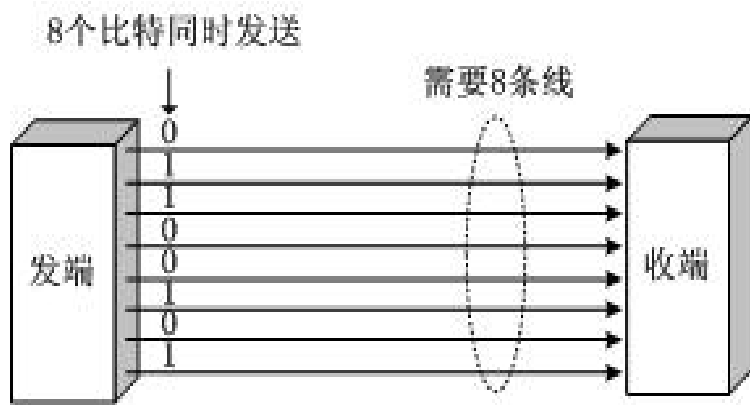


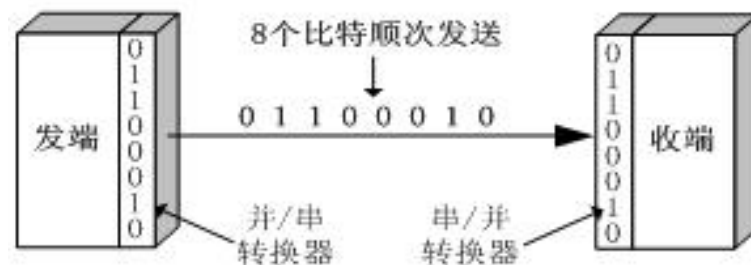
## 串行通信的基本知识

### 基本通信方式

计算机系统与外设的基本通信方式有两种：并行通信和串行通信。例如：**IDE（ATA）**硬盘接口、**PCI**接口等是并行通信接口，而**RS232**接口、**USB**接口、**SATA**硬盘接口等是串行通信接口。



并行通信



串行通信

### 串行通信的基本知识

#### 两种通信方式的优缺点

- 并行通信的优点是传送速度快；缺点是传输线多，通信线路费用较高。并行通信适用于近距离、传送速度高的场合。
- 串行通信的优点是传输线少，传送通道费用低；缺点是传送速度较低。串行通信适合长距离数据传送。

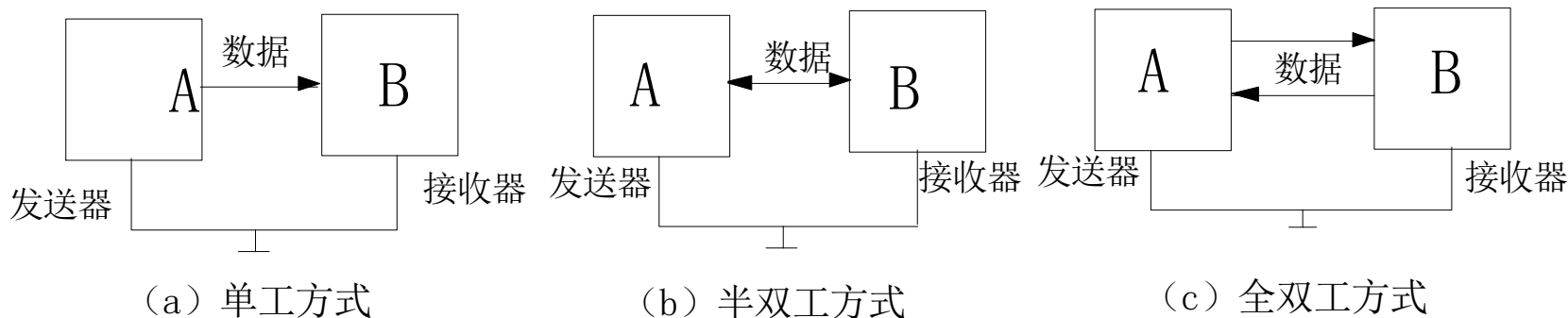
## 串行通信的基本知识

### 串行通信的制式

**单工方式**——通信双方只有一条单向传输线，只允许数据由一方发送，另一方接收。

**半双工方式**——通信双方只有一条双向传输线，允许数据双向传送，但每个时刻只能有一方发送，另一方接收，这是一种能够切换传送方向的单工方式，

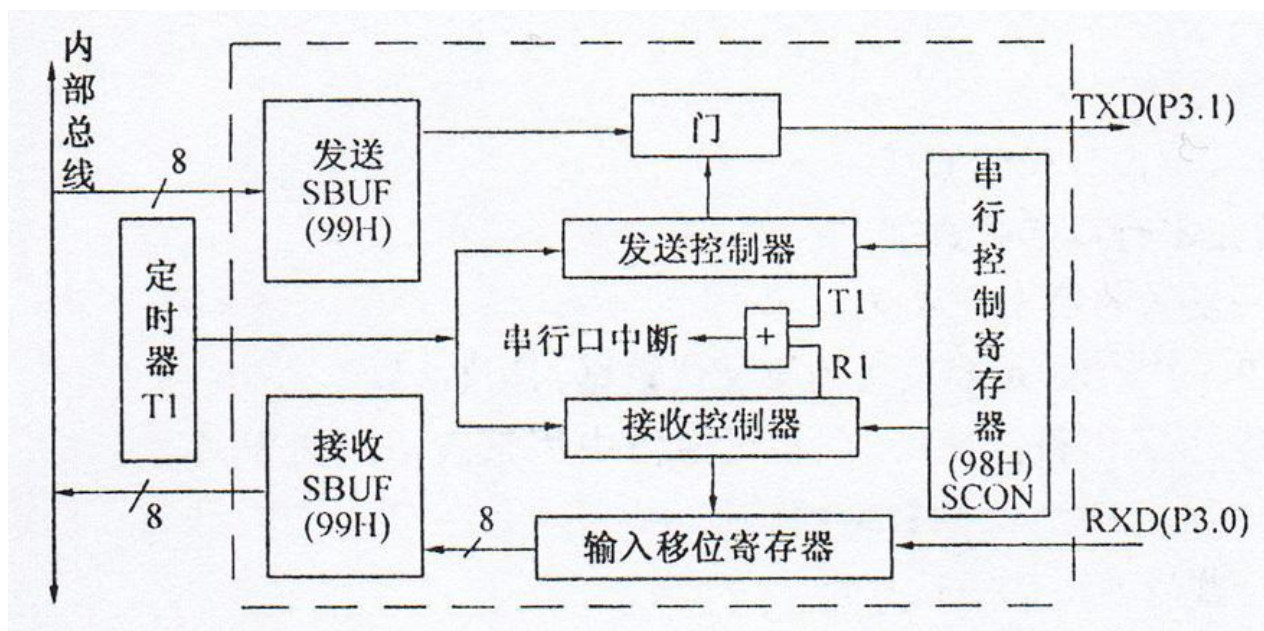
**全双工方式**——通信双方只有两条传输线，允许数据同时双向传送，双方通信设备应具有完全独立的收发功能。



# 第7部分 MCS-51的串行口

## 串行口的结构

- MCS-51单片机内部有一个全双工的串行异步通信口，也称UART（通用异步接收发送器，Universal Asynchronous Receiver/Transmitter）。
- MCS-51的串行口有两个独立的发送、接受缓冲器，发送和接受缓冲器是同一个字节地址（99H），但是两个物理上独立的缓冲器。
- 发送缓冲器只能写入不能读出，接收缓冲器只能读出不能写入。



### 串行口的结构

- 每当向发送缓冲器写一个字节数据时，就自动启动串口，将这个数据发送出去。当在发送完毕时，将TI置1，申请中断。
- 每当串口接收到一个字节数据，就将它放置在接收缓冲器中，准备读取。同时将RI置1，申请中断。
- MCS-51单片机的串行口控制寄存器有两个：SCON（串行口控制寄存器）和 PCON（电源控制寄存器）。

## 串行口的结构

### 串行口控制寄存器SCON

		D7	D6	D5	D4	D3	D2	D1	D0	
SCON		SM0	SM1	SM2	REN	TB8	RB8	TI	RI	98H
	位地址	9FH	9EH	9DH	9CH	9BH	9AH	99H	98H	

SCON的字节地址为98H，可位寻址，位地址为98H~9FH。用来设定串行口的工作方式、控制串行口的接收 / 发送以及状态标志。复位时为00H。

#### ■ SM0、SM1：串行口工作方式选择

SM0	SM1	方式	功 能 说 明
0	0	0	同步移位寄存器方式(用于扩展 I/O 口)
0	1	1	8 位异步收发,波特率可变(由定时器控制)
1	0	2	9 位异步收发,波特率为 $f_{osc}/64$ 或 $f_{osc}/32$
1	1	3	9 位异步收发,波特率可变(由定时器控制)



## 串行口的结构

### 串行口控制寄存器SCON

		D7	D6	D5	D4	D3	D2	D1	D0	98H
SCON		SM0	SM1	SM2	REN	TB8	RB8	TI	RI	
	位地址	9FH	9EH	9DH	9CH	9BH	9AH	99H	98H	

#### ■ SM2：多机通信控制位

在工作方式2和方式3中：

若SM2=1：当接收到第9位数据（RB8）为1，才将接收到的前8位数据装入SBUF，并置位RI；否则将接收到的数据丢弃。

若SM2=0：不论第9位数据（RB8）是否为1，都将接收到的前8位数据装入SBUF，并置位RI。

在方式1中：若SM2=1，则只有接收到有效的停止位时，才置位RI。

在方式0中：必须使SM2=0

## 串行口的结构

### 串行口控制寄存器SCON

	D7	D6	D5	D4	D3	D2	D1	D0		
SCON		SM0	SM1	SM2	REN	TB8	RB8	TI	RI	98H
	位地址	9FH	9EH	9DH	9CH	9BH	9AH	99H	98H	

#### ■ REN：允许串行接收位

REN=1，允许串行口接收数据；REN=0，禁止串行口接收数据。

#### ■ TB8：发送的第9位数据

在工作方式2或方式3时，TB8为发送的第9位数据，可由软件置位或清零。在双机通信中，该位可作为奇偶校验位。在多机通信中，TB8用来表示发送的是地址帧还是数据帧。**TB8=0时为数据帧，TB8=1时为地址帧。**

#### ■ RB8：接收到的第9位数据

在工作方式2或方式3时，RB8存放接收到的第9位数据；在方式1中，RB8是已接收到的停止位；在方式0中，RB8未用。





## 串行口的结构

### 串行口控制寄存器SCON

		D7	D6	D5	D4	D3	D2	D1	D0	
SCON		SM0	SM1	SM2	REN	TB8	RB8	TI	RI	98H
	位地址	9FH	9EH	9DH	9CH	9BH	9AH	99H	98H	

#### ■ TI: 发送中断标志

TI在一帧数据发送结束时由硬件置1，表示一帧数据发送结束，此时可向SBUF写入下一帧要发送的数据。TI可供软件查询，也可申请中断。TI必须由软件清0。

#### ■ RI: 接收中断标志

RI在接收完一帧有效数据时由硬件置1，表示一帧数据接收结束，并申请中断，要求CPU从接收SBUF取走数据。RI可供软件查询。RI必须由软件清0。

## 串行口的结构

电源控制寄存器**PCON**：只能字节寻址（**87H**），不能位寻址。

PCON	D7	D6	D5	D4	D3	D2	D1	D0
	SMOD	—	—	—	GF1	GF0	PD	IDL

### ■ SMOD：串行口波特率选择位。

当SMOD=1时，串行口波特率加倍。复位时，SMOD=0。

工作方式1、方式3时： **波特率** =  $\frac{2^{\text{SMOD}}}{32} \times \text{定时器T1溢出率}$

工作方式2时： **波特率** =  $\text{fosc} \times \frac{2^{\text{SMOD}}}{64}$

- GF1、GF0位通用标志位，PD为掉电方式位，IDL为休眠方式位。PD和IDL被置位后，CPU将分别进入掉电保持方式和休眠运行方式。（P40-P42）

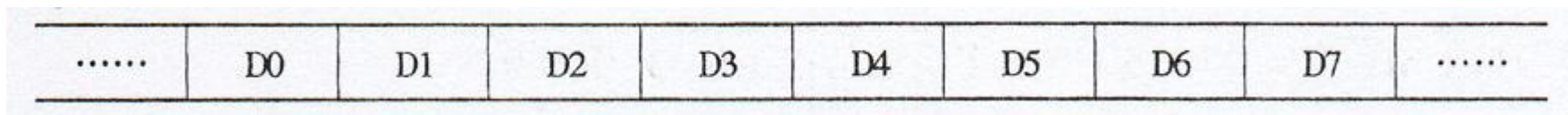


### 串行口的4种工作方式

#### 方式0

串行口的工作方式0为同步移位寄存器输入输出方式，常用于外界移位寄存器，以扩展I/O口。

- 8位数据为一帧，无起始位和停止位。发送/接收时，低位在先。
- RXD：数据输入/输出端。
- TXD：同步脉冲输出端，每个脉冲对应一个数据位。
- 固定波特率  $B = f_{osc}/12$

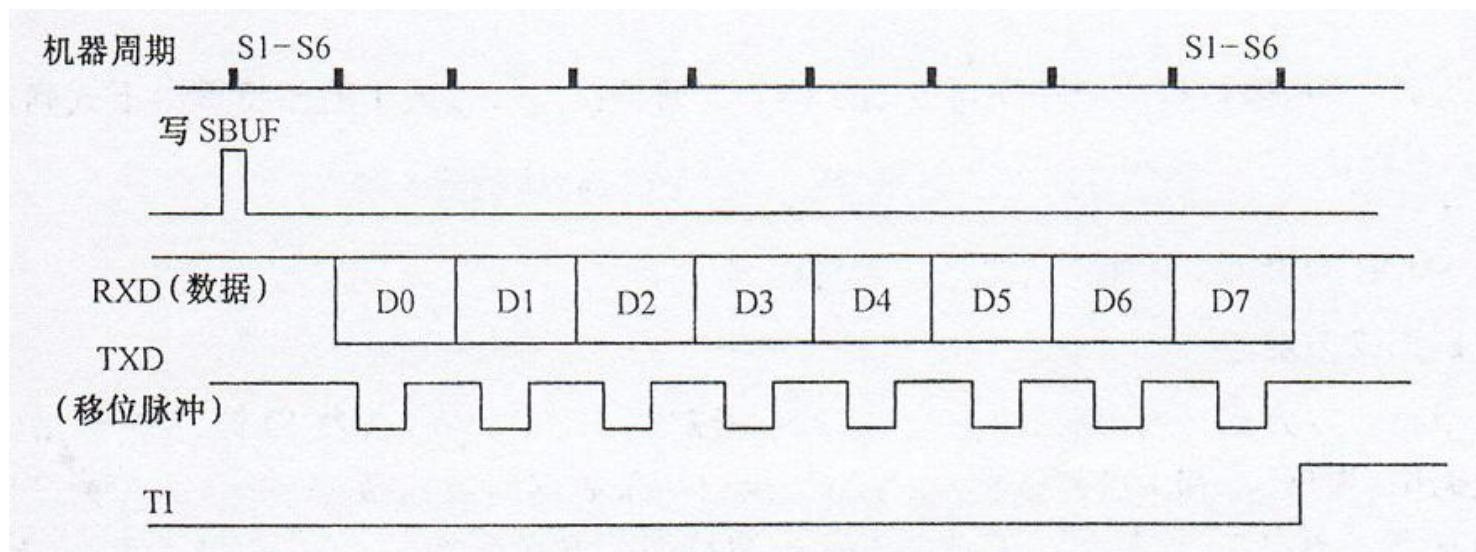


#### 方式0的帧格式

## 串行口的4种工作方式

### 方式0

**发送过程：** CPU执行将数据写入SBUF的指令，启动发送；串行口开始将SBUF中的数据以 $f_{osc}/12$ 的波特率从RXD引脚输出，TXD引脚输出同步移位脉冲。一帧发送结束，TI置1。

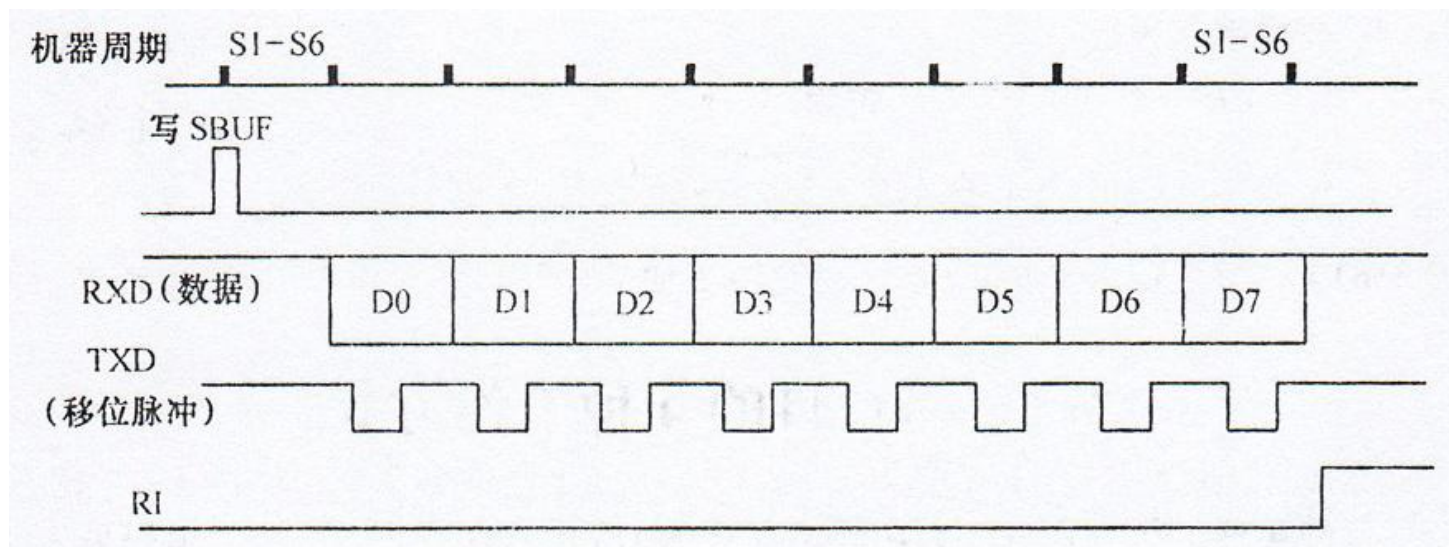


方式0发送时序

## 串行口的4种工作方式

### 方式0

接收过程：写入控制字SCON置方式0、REN=1及RI=0，启动接收；串行口开始将RXD引脚的数据以 $f_{osc}/12$ 的波特率输入SBUF，TXD引脚输出同步移位脉冲。一帧接收完毕，RI置1。



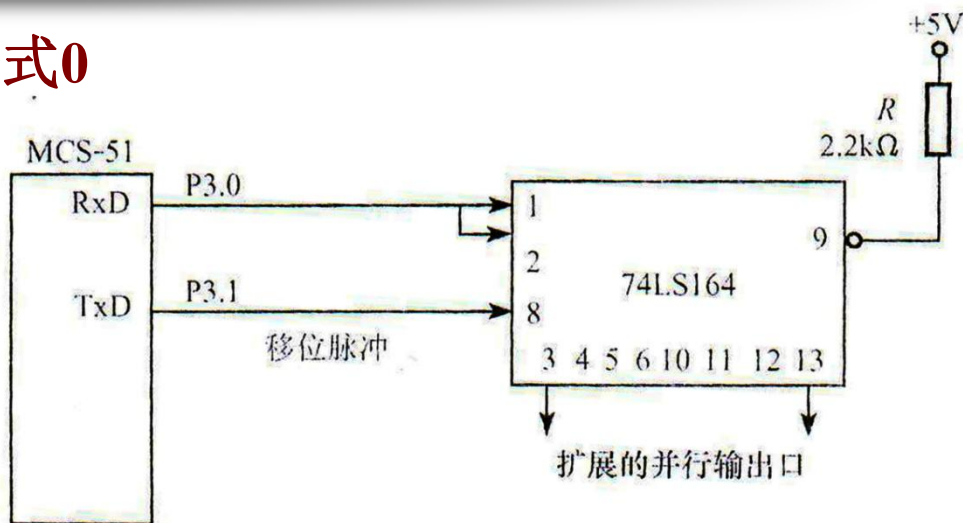
方式0接收时序



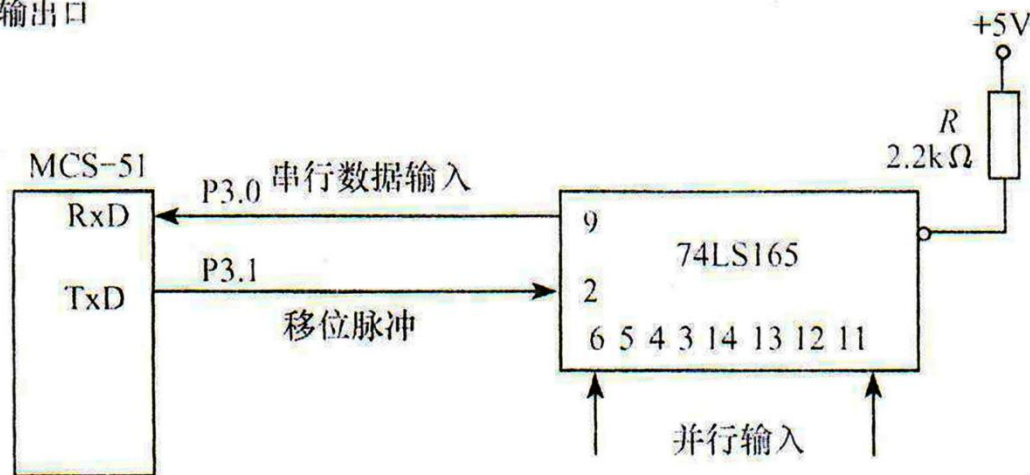
# 第7部分 MCS-51的串行口

## 串行口的4种工作方式

### 方式0



### 方式0发送接口示例



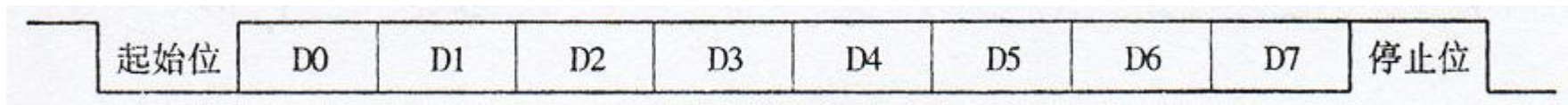
### 方式0接收接口示例

## 串行口的4种工作方式

### 方式1

串行口的工作方式1为异步串行通信方式。

- 10位数据为一帧，1个起始位，8个数据位，1个停止位。发送/接收时，低位在先。
- RXD：数据接收端；TXD：数据发送端。
- 波特率 =  $\frac{2^{\text{SMOD}}}{32} \times \text{定时器T1溢出率}$

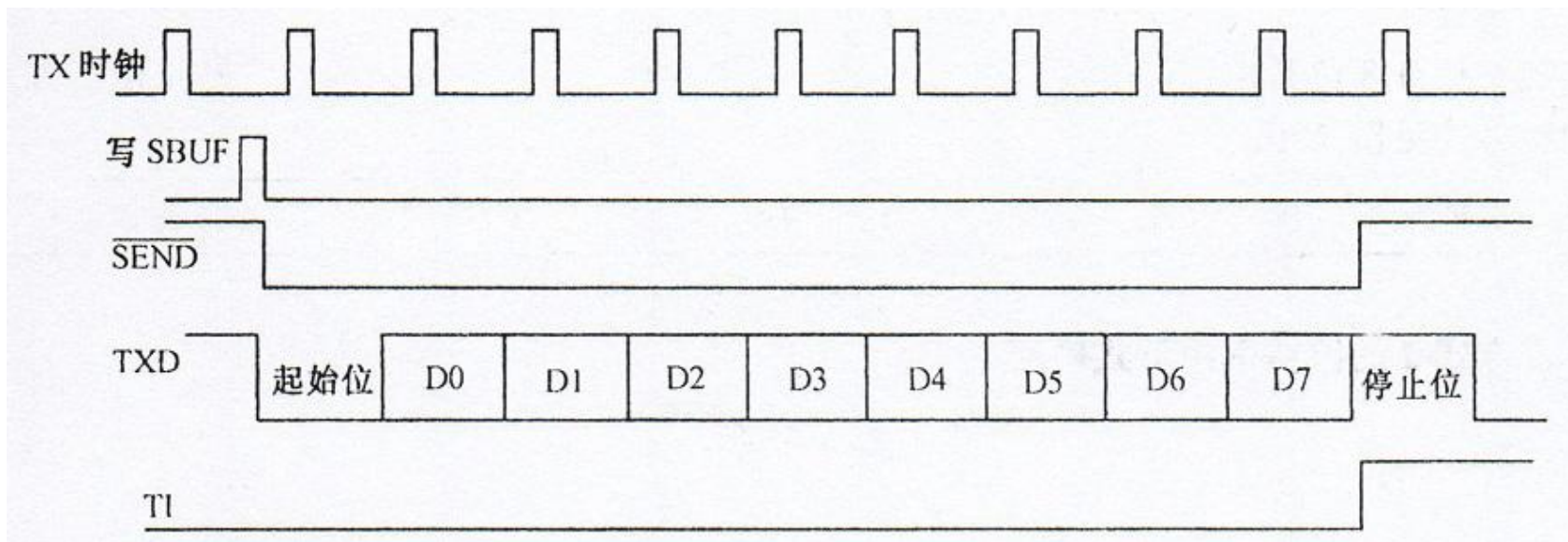


方式1的帧格式

## 串行口的4种工作方式

### 方式1

发送过程：CPU执行将数据写入SBUF的指令，启动发送；串行口开始将SBUF中的数据以波特率从TXD引脚输出。8位数据位发送结束时，TI置1。



方式1发送时序

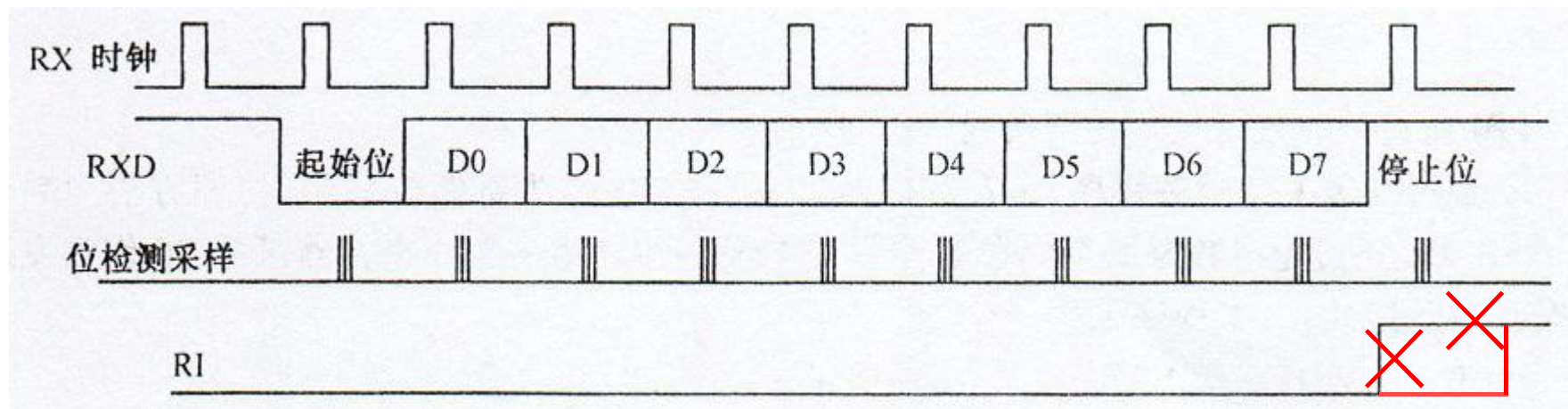
## 串行口的4种工作方式

### 方式1

接收过程：**REN=1**时允许接收。数据从**RXD**引入。当检测到起始位时开始接收。接收完一帧，若同时满足下列两个条件时接收有效：

**1) RI=0; 2) SM2=0或停止位为1。**

接收有效时，将接收数据装入**SBUF**，停止位装入**RB8**，并置**RI**为1；否则丢弃接收数据，不置位**RI**。

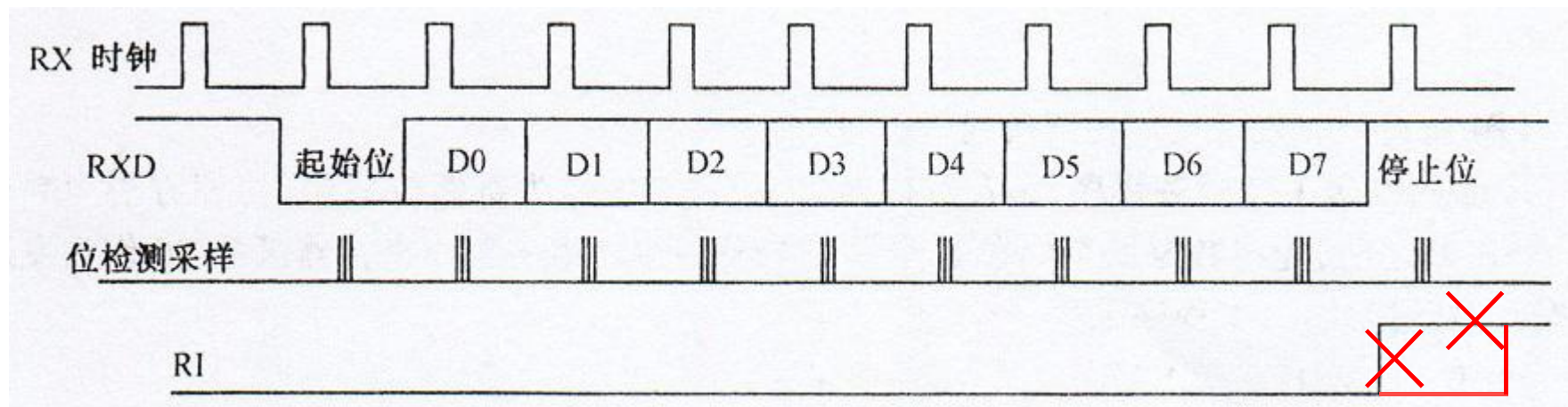


方式1接收时序

## 串行口的4种工作方式

### 方式1

接收过程：位检测器的数据采样速率为波特率16倍频，在数据位中间，用第7、8、9个脉冲采样3次数据位，并3中取2保留采样值。当检测到RXD端从1到0的负跳变时就启动位检测器。



方式1接收时序



### 串行口的4种工作方式

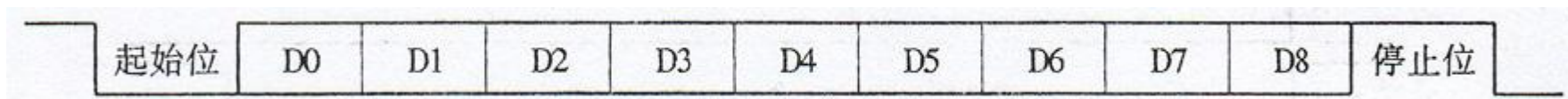
#### 方式2和方式3

串行口的工作方式2和方式3都为异步串行通信方式。两者除了波特率的确定方式不一样之外，其余都是相同的。

- 11位数据为一帧，1个起始位，8个数据位，1个第9位，1个停止位。发送/接收时，低位在先。
- RXD：数据接收端；TXD：数据发送端。

- 方式2的波特率 
$$= f_{osc} \times \frac{2^{SMOD}}{64}$$

- 方式3的波特率 
$$= \frac{2^{SMOD}}{32} \times \text{定时器T1溢出率}$$



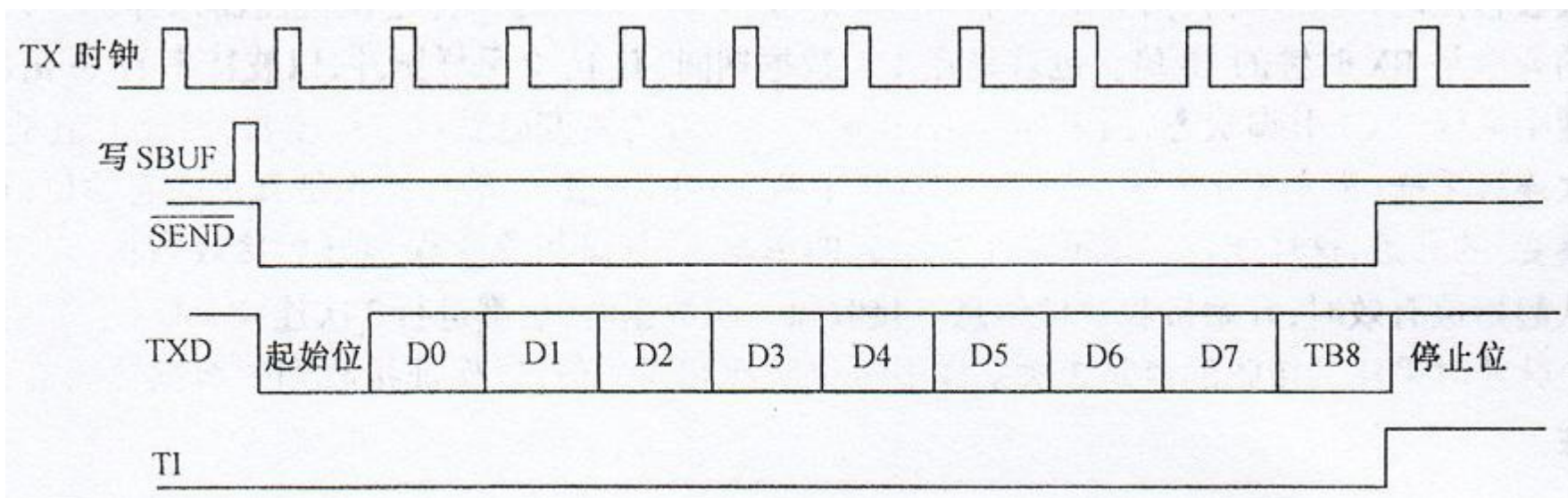
方式2和方式3的帧格式



## 串行口的4种工作方式

### 方式2和方式3

发送过程：先根据通信协议设置**TB8（第9位）**，然后CPU执行将8位数据写入**SBUF**的指令，启动发送；串行口开始将**SBUF**中的8位数据和**TB8**以波特率从**TXD**引脚输出。**TB8发送结束时，TI置1**。



方式2和方式3发送时序

### 串行口的4种工作方式

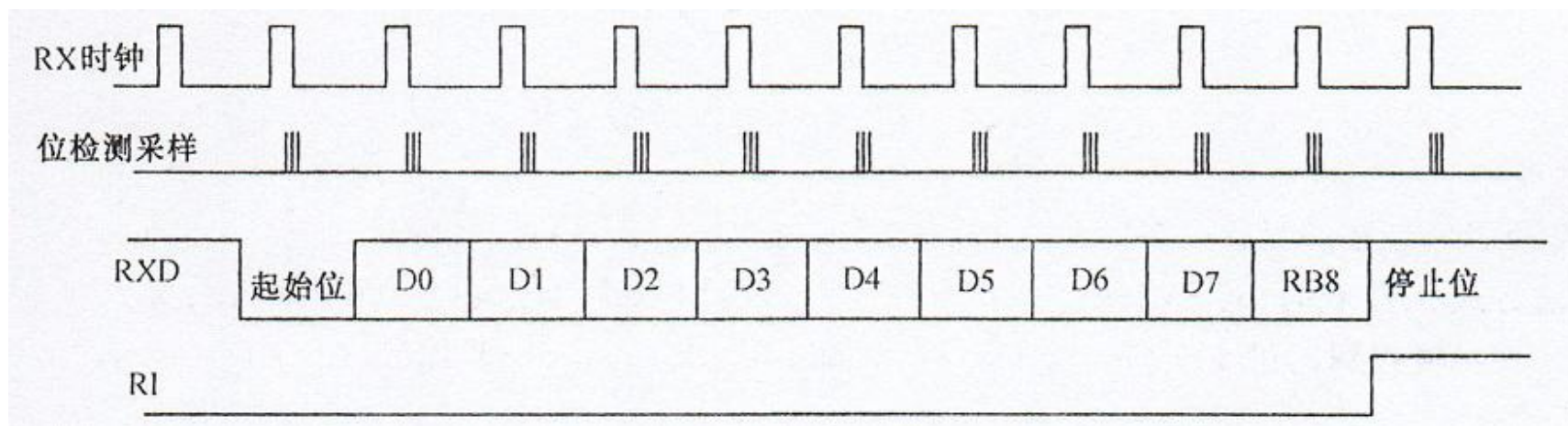
#### 方式2和方式3

接收过程：**REN=1**时允许接收。数据从**RXD**引入。当检测到起始位时开始接收。接收完一帧，若同时满足下列两个条件时接收有效：

**1) RI=0; 2) SM2=0或第9位为1。**

接收有效时，将接收的8位数据装入**SBUF**，第9位装入**RB8**，并置**RI**为1；否则丢弃接收数据，不置位**RI**。

位检测器的工作过程同方式1。



方式2和方式3接收时序

## 串行口的4种工作方式

**例1（P142例6-1）：**方式2和方式3发送在双机通信中的应用，TB8作为奇偶校验位。

<b>PIPTI:</b>	<b>PUSH</b>	<b>PSW</b>	； 现场保护
	<b>PUSH</b>	<b>ACC</b>	
	<b>CLR</b>	<b>RS0</b>	； 选择第2组工作寄存器区
	<b>SETB</b>	<b>RS1</b>	
	<b>CLR</b>	<b>TI</b>	； 发送中断标志清0
	<b>MOV</b>	<b>A, @R0</b>	； 取数据
	<b>MOV</b>	<b>C, P</b>	； 校验位送TB8，采用偶校验
	<b>MOV</b>	<b>TB8, C</b>	； 置第9位
	<b>MOV</b>	<b>SBUF, A</b>	； 数据写入发送缓冲区，启动发送
	<b>INC</b>	<b>R0</b>	； 数据指针加1
	<b>POP</b>	<b>ACC</b>	； 恢复现场
	<b>POP</b>	<b>PSW</b>	
	<b>RETI</b>		； 中断返回

在主程序中需  
要在中断允许的前  
提下，通过设置TI  
为1来启动首次串  
行中断服务程序。

# 第7部分 MCS-51的串行口

## 串行口的4种工作方式

**例2 (P143例6-2)：**方式2和方式3接收在双机通信中的应用

<b>PIR1:</b>	<b>PUSH PSW</b>	； 现场保护
	<b>PUSH ACC</b>	
	<b>SETB RS0</b>	； 选择第1组工作寄存器区
	<b>CLR RS1</b>	
	<b>CLR RI</b>	； 接收中断标志清0
	<b>MOV A, SBUF</b>	； 取接收到的8位数据
	<b>MOV C, P</b>	； 取校验位
	<b>JNC L1</b>	
	<b>JNB RB8, ERP</b>	； ERP为出错处理程序编号
	<b>AJMP L2</b>	
<b>L1:</b>	<b>JB RB8, ERP</b>	
<b>L2:</b>	<b>MOV @R0, A</b>	； 保存数据
	<b>INC R0</b>	； 数据指针加1
	<b>SJMP L0</b>	
<b>ERP:</b>	<b>.....</b>	； 出错处理
	<b>.....</b>	
<b>L0:</b>	<b>POP ACC</b>	； 恢复现场
	<b>POP PSW</b>	
	<b>RETI</b>	； 中断返回



### 波特率的制定方法

#### 波特率的定义

串行口每秒钟发送（或接收）的位数成为波特率。设发送一位所需的时间为T，则波特率为1/T。

方式0：波特率 =  $f_{osc}/12$

方式1：波特率 =  $\frac{2^{SMOD}}{32} \times \text{定时器T1溢出率}$

方式2：波特率 =  $f_{osc} \times \frac{2^{SMOD}}{64}$

方式3：波特率 =  $\frac{2^{SMOD}}{32} \times \text{定时器T1溢出率}$

方式1和方式3的波特率和定时器T1的溢出率有关。

### 波特率的制定方法

#### 定时器T1产生波特率的计算

通常定时器T1在作为波特率发生器时，采用定时器/计数器的工作方式2（自动装初值），即TL1为8位计数器，TH1存放重装初值。溢出率的计算公式为：

$$\text{定时器T1的溢出率} = \frac{f_{osc}/12}{(256 - TH1)}$$

$$\text{所以，串行方式1、方式3的波特率} = \frac{2^{SMOD}}{32} \times \frac{f_{osc}}{12 \times (256 - TH1)}$$

## 串行口的编程和应用

例3（P159-  
例6-7）：串  
行口方式3应  
用编程（双机  
通信）

甲机发送程序

page 1

```
ORG    0000H
LJMP   MAIN
ORG    0023H
LJMP   C_INT
ORG    1000H
MAIN:  MOV    SP, #53H           ; 设置堆栈指针
        MOV    78H, #20H       ; 设置存放数据单元的首地址
        MOV    77H, #00H
        ACALL  TRANS           ; 调用发送子程序
        SJMP   $
TRANS: MOV    TMOD, #20H        , 设置定时器/计数器工作方式
        MOV    TH1, #0FDH       ; 设置波特率为4800
        MOV    TL1, #0FDH
        SETB   TR1              ; 开定时器
        MOV    SCON, #0E0H      ; 设置串行工作方式3
        SETB   TB8              ; 设置第9位数据位
        MOV    IE, #00H         ; 关中断
        MOV    SBUF, 78H        ; 查询方式发送首地址高8位
```

## 串行口的编程和应用

### 例3 串行口方式3应用编程 (双机通信)

#### 甲机发送程序

page 2

```
WAIT:  JNB     TI, WAIT
        CLR     TI
        MOV     SBUF, 77H           ; 发送首地址低8位
WAIT2:  JNB     TI, WAIT2
        CLR     TI
        MOV     IE, #90H           ; 开中断
        CLR     TB8
        MOV     A, #00H
        MOV     SBUF, A             ; 开始发送数据
WAIT1:  CJNE    A, #0FFH, WAIT1     ; 判断数据是否发送完毕
        CLR     ES                 ; 发送完毕则关中断
        RET
C_INT:  CLR     TI                 ; 中断服务子程序
        INC     A                  ; 要发送的数据值加1
        MOV     SBUF, A             ; 发送数据
        RETI                       ; 中断返回
        END
```

## 串行口的编程和应用

### 例4 串行口方式3应用编程 (双机通信)

### 乙机接收程序

page 1

```
ORG 0000H
LJMP MAIN
ORG 0023H
LJMP C_INT
ORG 1000H
MAIN: MOV SP, #53H          ; 设置堆栈指针
      MOV R0, #0FEH        ; 设置地址接收计数器初值
      ACALL RECEI          ; 调用接收子程序
      SJMP $
RECEI: MOV TMOD, #20H      , 设置定时器/计数器工作方式
      MOV TH1, #0FDH      ; 设置波特率为4800
      MOV TL1, #0FDH
      SETB TR1            ; 开定时器
      MOV IE, #90H        ; 开中断
      MOV SCON, #0F0H     ; 设置串行工作方式, 允许接收
                          ; REN=1, SM2=1
      SETB 00H            ; 设置标志位, 位地址00H
WAIT: JB 00H, WAIT        ; 等待接收
      RET
```



## 串行口的编程和应用

### 例7.5 串行口 方式3应用编 程（双机通信）

### 乙机接收程序

page 2

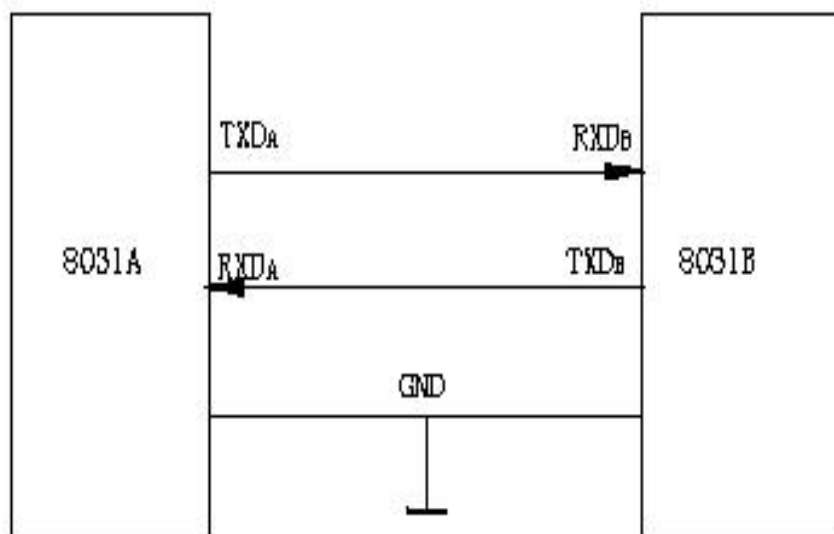
<b>C_INT:</b>	<b>CLR</b>	<b>RI</b>	； 清接收中断标志位
	<b>MOV</b>	<b>C, RB8</b>	； 对第9位进行判断是数据或地址
	<b>JNC</b>	<b>PD2</b>	； 是数据则进入数据存储
	<b>INC</b>	<b>R0</b>	
	<b>MOV</b>	<b>A, R0</b>	
	<b>JZ</b>	<b>PD</b>	
	<b>MOV</b>	<b>DPH, SBUF</b>	
	<b>SJMP</b>	<b>PD1</b>	
<b>PD:</b>	<b>MOV</b>	<b>DPL, SBUF</b>	
	<b>CLR</b>	<b>SM2</b>	； 清地址接收标志位
	<b>SJMP</b>	<b>PD1</b>	
<b>PD2:</b>	<b>MOV</b>	<b>A, SBUF</b>	； 接收数据
	<b>MOVX</b>	<b>@DPTR, A</b>	
	<b>INC</b>	<b>DPTR</b>	
	<b>CJNE</b>	<b>A, #0FFH, PD1</b>	； 判断是否为最后一帧数据
	<b>SETB</b>	<b>SM2</b>	； 置下一次接收地址数据
	<b>CLR</b>	<b>00H</b>	； 清标志位，数据接收结束
	<b>CLR</b>	<b>ES</b>	； 接收完毕关中断
<b>PD1:</b>	<b>RETI</b>		； 中断返回

## 串行口的编程和应用

### 串行通信硬件接口

根据MCS-51单片机的通信距离、抗干扰性的要求，常见的串行通信的硬件接口包括TTL电平传输、RS232C接口传输、RS422A接口传输和RS485总线传输等。

#### 1) TTL电平传输

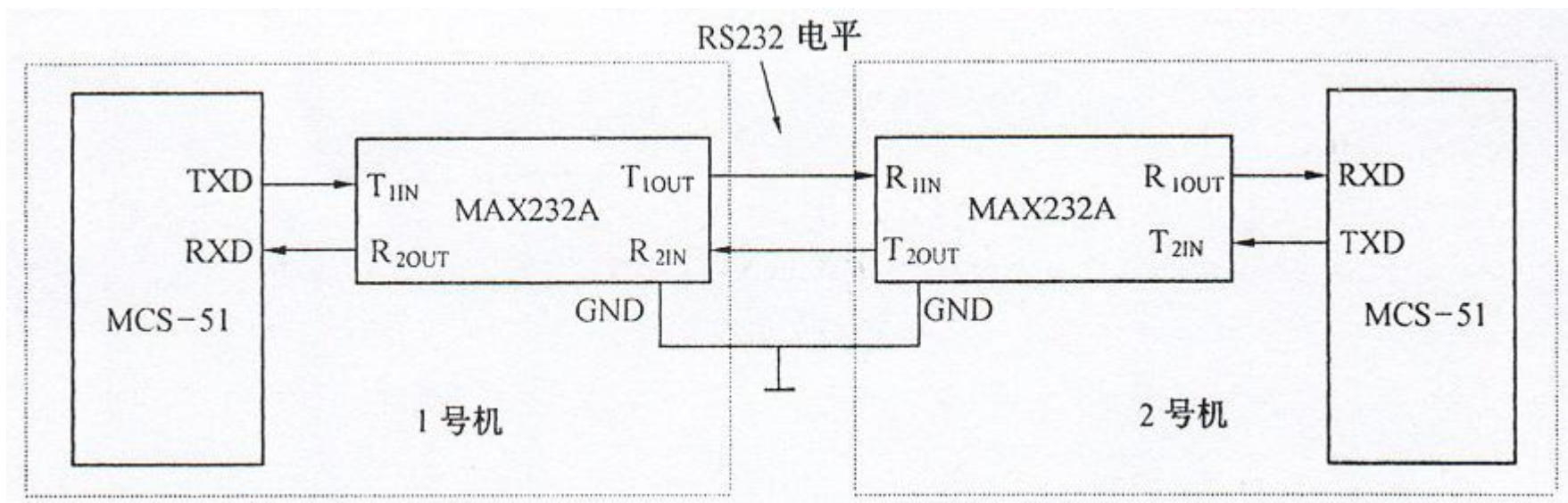


## 串行口的编程和应用

### 串行通信硬件接口

#### 2) RS232C接口传输

逻辑1 = -5V ~ -15V; 逻辑0 = +5 ~ +15V。传输距离: 15m



## 串行口的编程和应用

### 串行通信硬件接口

#### 3) RS422总线传输

RS-422 采用平衡传输模式（差分信号）负逻辑， $+0.2V \sim +6V$  表示“0”， $-6V \sim -0.2V$ 表示“1”。最大传输距离约1200米，最大传输速率为10Mb/s。其平衡双绞线的长度与传输速率成反比。

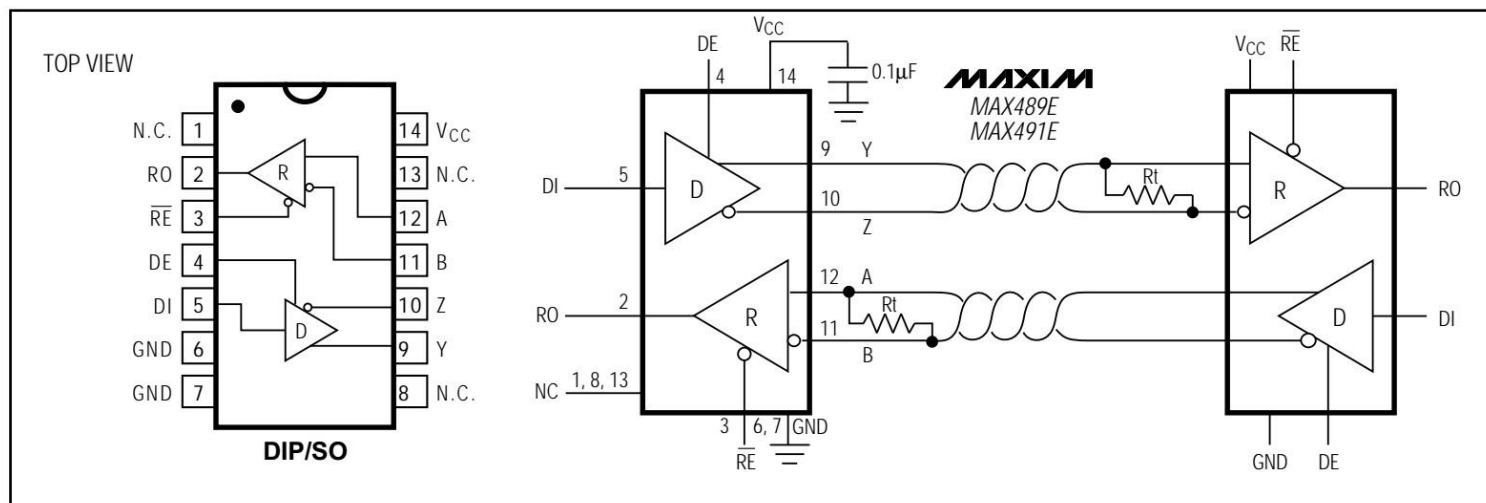


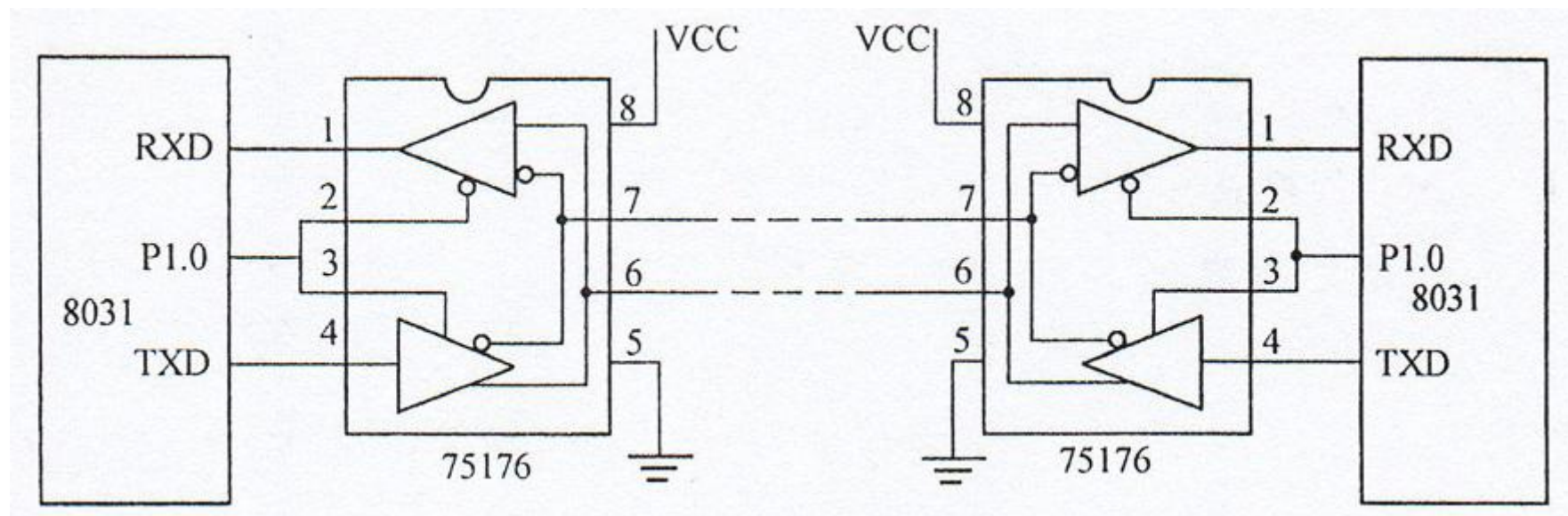
Figure 3. MAX489E/MAX491E Pin Configuration and Typical Operating Circuit

## 串行口的编程和应用

### 串行通信硬件接口

#### 4) RS485总线传输

RS-485和RS422有相同的传输模式和电气参数。区别在于，RS422的接收器和发送器采用不同的总线，可以全双工通信；而RS485的收发采用一条总线，只能半双工通信。总线挂接节点的数量为32~128个（依据收发器的阻抗特性）。



# 第7部分 MCS-51的串行口

## RS485和 RS422收 发器对比

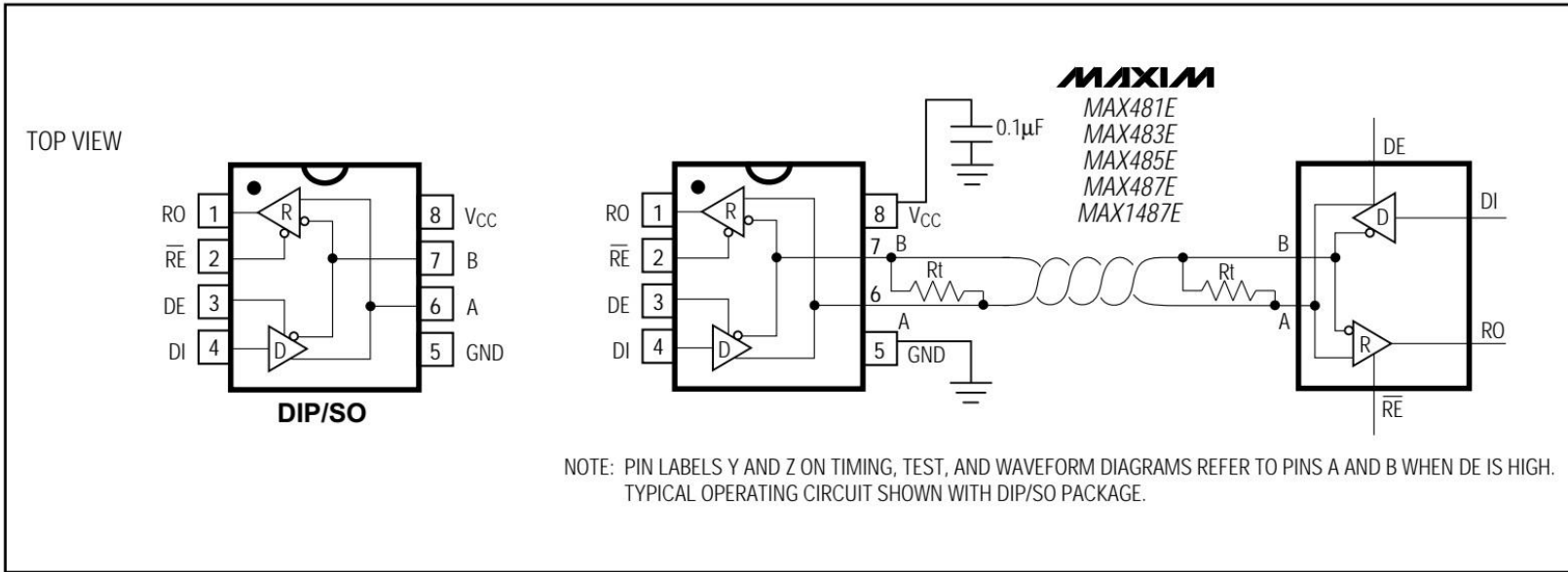


Figure 1. MAX481E/MAX483E/MAX485E/MAX487E/MAX1487E Pin Configuration and Typical Operating Circuit

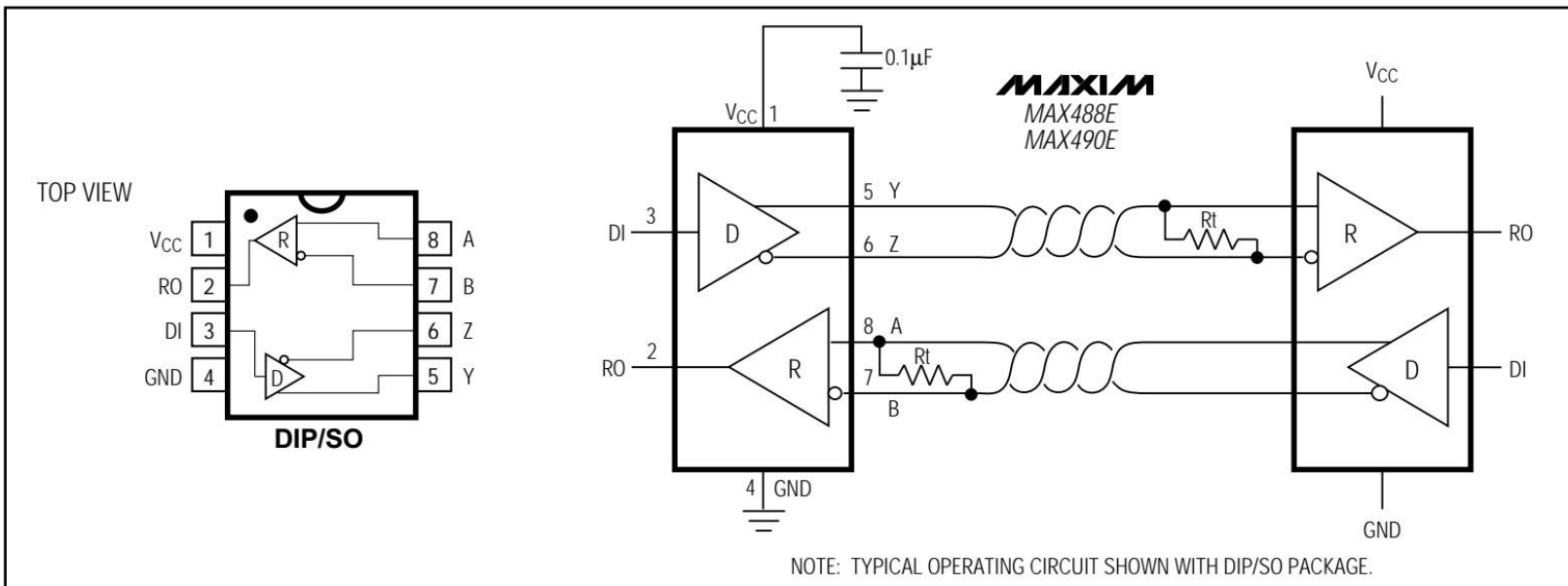


Figure 2. MAX488E/MAX490E Pin Configuration and Typical Operating Circuit



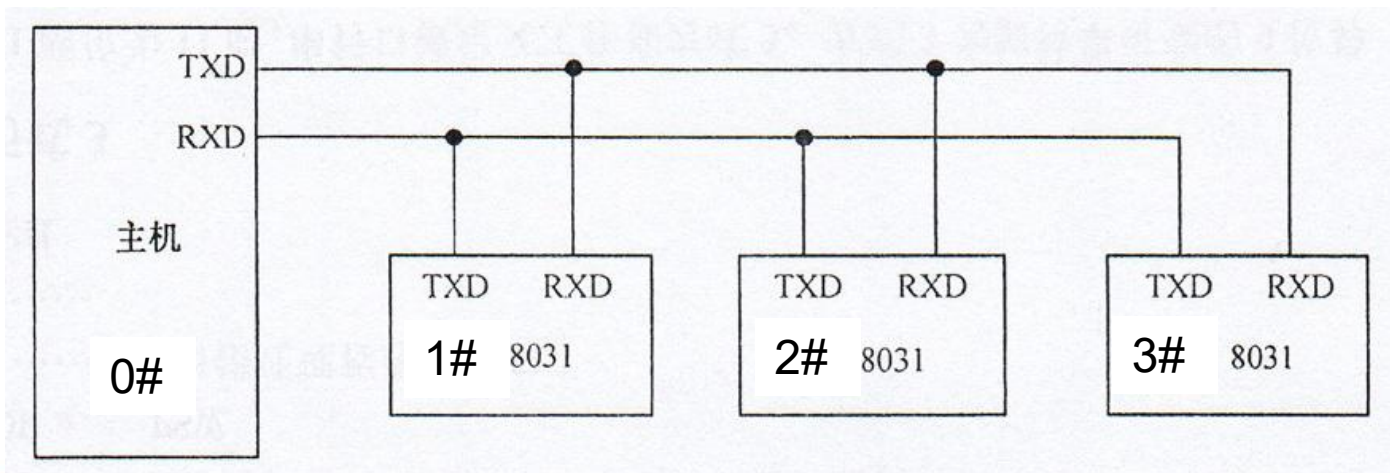
## 多机通信

MCS-51串行通信的方式2和方式3可以实现同一条串行总线上的多机通信过程。基本原理如下：

1) 发送方：发送地址信息时，将多机通信位SM2置1，并将第9位置为1表示是地址帧；

发送数据信息时，将第9位置为0表示是数据帧；

2) 接收方：接收地址帧时，将多机通信位SM2置1；  
接收数据帧时，将多机通信位SM2置0。



# 第7部分 MCS-51的串行口

## 多机通信

发送/接收字节	发送主机	接收从机	其他从机
地址字节=DA	TB8=1, 数据=DA	① SM2=1, 接收到第9位=1, 产生接收中断; ② DA→SBUF; ③ DA符合本机地址, 将SM2置为0, 准备接收数据字节。	① SM2=1, 接收到第9位=1, 产生接收中断; ② DA→SBUF; ③ DA不符合本机地址, 维持SM2=1。
数据字节1=DD1	TB8=0, 数据=DD1	① SM2=0, 产生接收中断; ② DD1→SBUF。	① SM2=1, 接收到第9位=0; ② 放弃DD1, 不产生接收中断。
.....	.....	.....	.....
数据字节n=DDn	TB8=0, 数据=DDn	① SM2=0, 产生接收中断; ② DDn→SBUF。	① SM2=1, 接收到第9位=0; ② 放弃DDn, 不产生接收中断。
校验字节=CRC	TB8=0, 数据=CRC	① SM2=0, 产生接收中断; ② CRC→SBUF; ③ 检查校验是否正确, 本轮接收结束, SM2=1。	① SM2=1, 接收到第9位=0; ② 放弃CRC, 不产生接收中断。

## 作业

### 第七部分 作业

- 一、51 单片机的串行口有几种工作方式？有几种帧格式（画出每种方式的帧格式）？各种工作方式的波特率如何确定？
- 二、若晶体振荡器为 11.0592MHz，串行口工作于方式 3 的数据帧接收状态，波特率为 1200bit/s，T1 采用定时器方式 2，请写出与串行口及 T1 设置相关的初始化程序。  
(T1 计数初值要有计算过程)
- 三、编写串口方式 3 下的接收程序（包括主程序和中断服务程序）。设波特率为 2400bps， $f_{osc}=11.0592\text{ MHz}$ ，接收数据采用偶校验。接收数据存放在片外 RAM，起始地址为 3080H，数据块长度为 40 字节。