

# 定时器/计数器T0与T1的初始化

## 1. 初始化的步骤:

- 确定工作方式、操作模式、启动控制方式，写入TMOD、TCON。
- 设置定时或计数器的初值，直接将初值写入TH0、TL0或TH1、TL1中。
- 根据需要，开放CPU和定时 / 计数器的中断，即对IE和IP寄存器编程
- 启动定时器 / 计数器工作：若要求用软件启动，编程时对TCON中的TR0或TR1置位即可启动；若由外部中断引脚电平启动，则对TCON中的TR0或TR1置位后，还需给外部中断引脚加启动电平。

# 定时器/计数器T0与T1的初始化

## 2. 计数初值的计算

若设最大计数值为 $2^n$ ， $n$ 为计数器位数，各工作方式下的 $2^n$  值为：

方式0:  $2^n=8192$  ;  $n=13$

方式1:  $2^n=65536$  ;  $n=16$

方式2:  $2^n=256$  ;  $n=8$

方式3:  $2^n=256$  ;  $n=8$ ，定时器T0分成2个独立的8位计数器，所以TH0、TL0的最大计数值均为256。

T0、T1定时器均为加1计数器，当加到最大值（00H或0000H）时产生溢出中断，因此计数器初值 $X$ 的计算式为： $X=2^n - \text{计数值}$

## 定时器/计数器T0与T1的初始化

### ◆ 计数模式

计数模式时，对外部脉冲进行计数，其计数初值X：X=2<sup>n</sup>-计数值。

定时时长  $t = \text{计数值} \times \text{外部时钟周期}$

### ◆ 定时模式

定时模式时，对机器周期进行计数，故计数频率为  $f_{\text{cont}} = f_{\text{osc}} / 12$ ，  
计数周期（机器周期） $T = 1 / f_{\text{cont}} = 12 / f_{\text{osc}}$ ，定时模式的计数初值X等于：

$$X = 2^n - \text{计数值} = 2^n - t / T = 2^n - (t \times f_{\text{osc}}) / 12$$

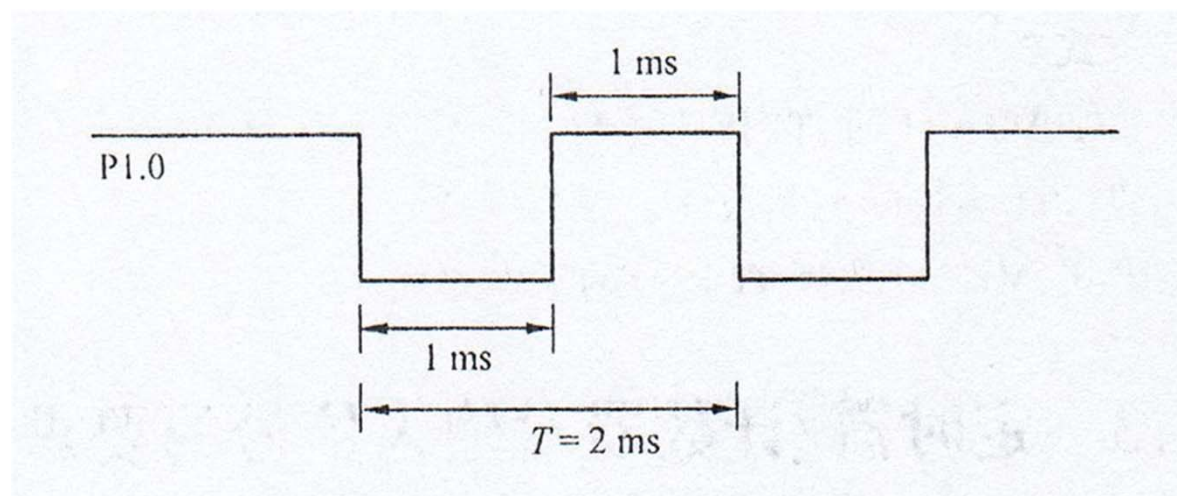
（注：f<sub>osc</sub>单位是MHz，定时时长t的单位是 μs。）

定时时长  $t = \text{计数值} \times \text{机器周期} = (2^n - X) \times 12 / f_{\text{osc}}$

## 定时器/计数器T0与T1的应用—初值计算

### 例（定时器/计数器T0 方式1的应用）

假设系统时钟频率 $f_{osc}$ 采用6MHz，要在P1.0上输出一个周期为2ms的方波，如图所示。



采用定时器T0，每1ms产生一次中断，在中断服务程序程序中对P1.0取反，即P1.0 的值每1ms变化一次。

## 定时器/计数器T0与T1的应用—初值计算

例（定时器/计数器T0， 方式1的应用）

计算初值（  $f_{osc} = 6\text{MHz}$ ）

$$\text{机器周期} = \frac{12}{6 \times 10^6} = 2 \times 10^{-6}(\text{s}) = 2(\mu\text{s})$$

T0的初值X计算：（定时时长  $t = 1\text{ms}$ ）

$$X = 2^n - \text{计数值} = 2^n - t / T = 2^n - (t \times f_{osc}) / 12$$

$$n = 16, \text{计数值} = 1\text{ms} / 2\mu\text{s} = 500$$

$$X = 2^{16} - \text{计数值} = 2^{16} - 500 = 65536 - 500 = 65036 = 0\text{FE0CH}$$

所以，T0的初值为：TH0=0FEH，TL0=0CH。

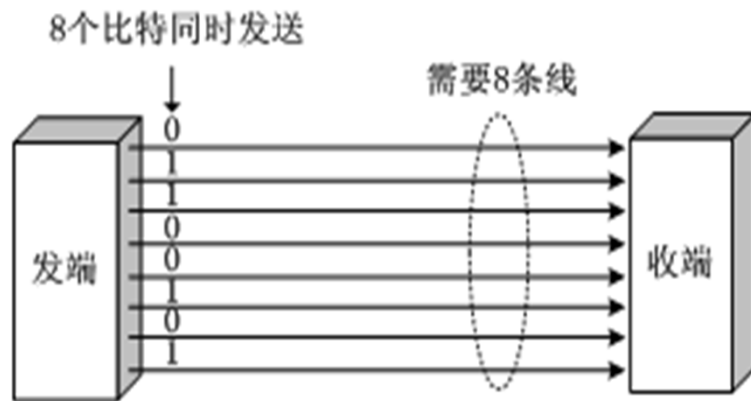
## 1.8 MCS-51的串行口

### 串行通信的基本知识

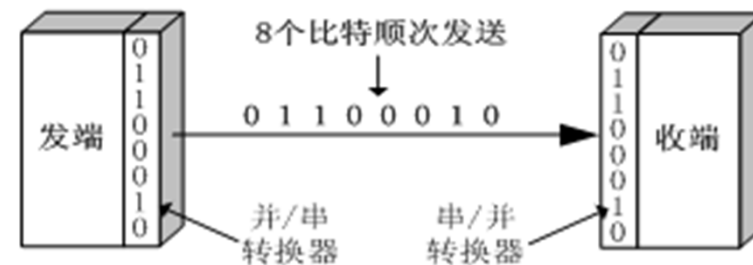
#### 基本通信方式

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。串行通信适合远距离数据传送。

串行通信是一位一位地依次传输（发送或接受）数据的一种通信方式。适合于计算机之间、计算机与外部设备之间的远距离通信。

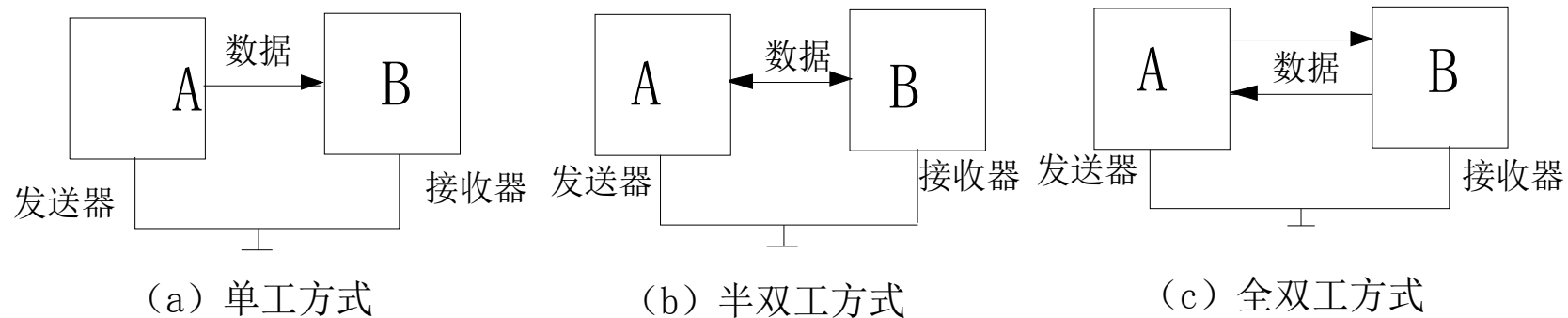
# 串行通信的基本知识

## 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，

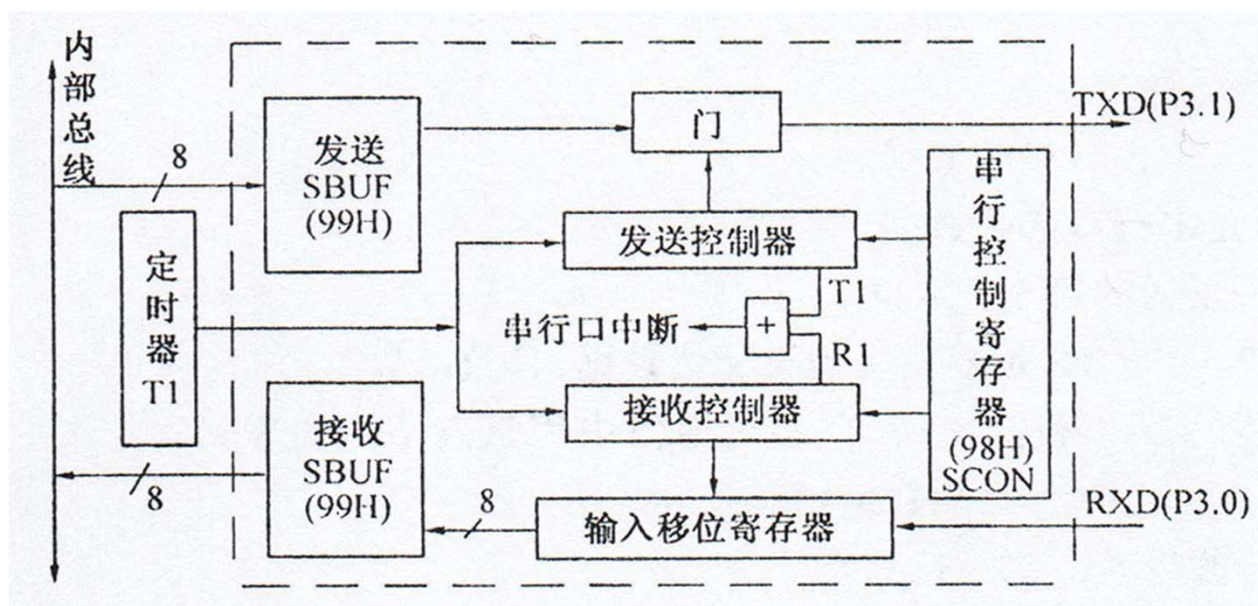
**全双工方式**——通信双方只有**两条**传输线，允许数据同时**双向**传送，双方通信设备应具有完全独立的收发功能。





## 串行口的结构

- MCS-51的串行口是一个全双工的通信接口（同时发送和接收）；
- MCS-51的串行口，可以用作通用异步接收与发送器UART（Universal Asynchronous Receiver/Transmitter），也可用作同步移位寄存器。
- MCS-51的串行口有一个数据缓冲寄存器SBUF，它对应两个独立的发送缓冲器和接收缓冲器，发送和接收缓冲器有同一个字节地址（99H），但它们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（发送中断标志）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（接收中断标志）置1，申请中断。
- MCS-51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用于控制和监视串行口的工作状态，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	模式	功能描述	波特率（每秒传输数据的位数）
00	0	8位同步移位寄存器	$f_{osc}/12$
01	1	8位UART（8位数据位）	可变（T1溢出率）
10	2	9位UART（9位数据位）	$f_{osc}/64$ 或 $f_{osc}/32$
11	3	9位UART（9位数据位）	可变（T1溢出率）

## 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	

### ■ 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：不论接收到第9位数据（RB8）是1还是0，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。**在方式0、1中TB8不用。

### ■ RB8：接收的第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（可能是奇偶位，或者是地址/数据标识位）；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，申请中断，在中断服务子程序中向SBUF写入下一帧要发送的数据。TI状态可供软件查询，在中断服务程序中，TI必须由软件清0，不能硬件清除。

### ■ 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状态可供软件查询。在中断服务程序中，RI必须由软件清0，不能硬件清除。

TI、RI共用一个中断服务入口（0023H），所以要求用户在服务程序中测试中断源性质。

## 电源控制寄存器PCON

只能字节寻址（87H），不能位寻址。

	D7	D6	D5	D4	D3	D2	D1	D0
PCON	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 =  $\text{fosc} \times \frac{2^{\text{SMOD}}}{64}$

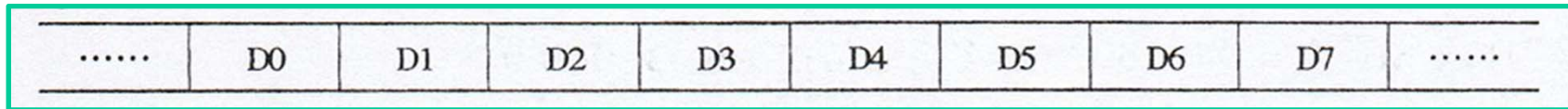
### ■ GF1、GF0位通用标志位，PD为掉电方式位，IDL为休眠方式位。PD和IDL被置位后，CPU将分别进入掉电保持方式和休眠运行方式。

## 串行口相关的寄存器组成

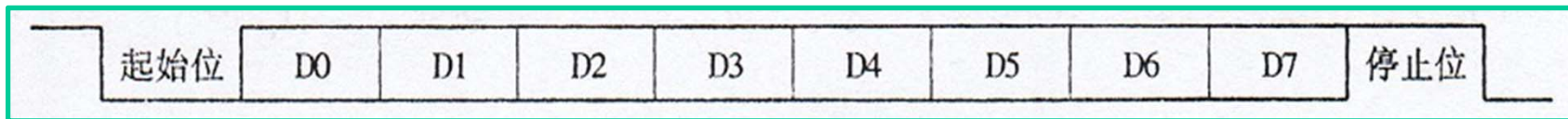
序号	记助符	名 称	地址	位寻址	功 能	备 注
1	SBUF	接收缓冲器	99H	不可	读入串行数据	逻辑地址共用 物理地址独立
2	SBUF	发送缓冲器	99H	不可	输出串行数据	
3	SCON	串行通信控制寄存器	98H	可以	串行口模式控制	
4	PCON	电源和波特率控制寄存器	87H	不可	波特率倍增控制	
5	IE	中断允许寄存器	A8H	可以	ES = 1 允许中断 ES = 0 禁止中断	中断控制
6	IP	中断优先级寄存器	B8H	可以	PS = 1 高优先级 PS = 0 低优先级	
6	TMOD	定时器模式寄存器	89H	不可	选择定时器模式	波特率发生器
7	TCON	定时器控制寄存器	88H	可以	TR1 = 1 启动T1 TR1 = 0 停止T1	
8	TL1 TH1	定时器1 初值寄存器	8BH 8DH	不可	定时器设置初值	



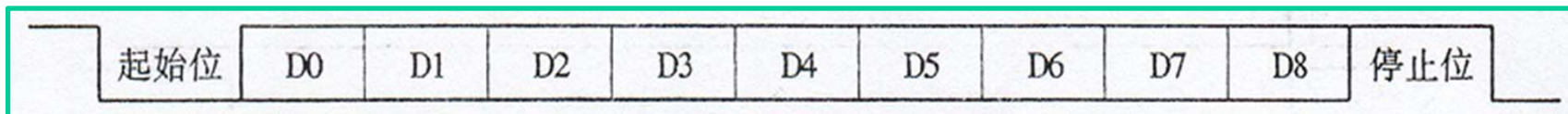
- ❑ 串行口使用P3.0 (RXD)、P3.1 (TXD)，支持全双工串行通信。
- ❑ 支持四种工作方式，每种工作方式传输一帧信息的格式：



方式0的帧格式（8位）



方式1的帧格式（10位）



方式2和方式3的帧格式（11位）

发送：所有方式下，写SBUF启动发送过程，8位数据装入发送SBUF，停止位（1）

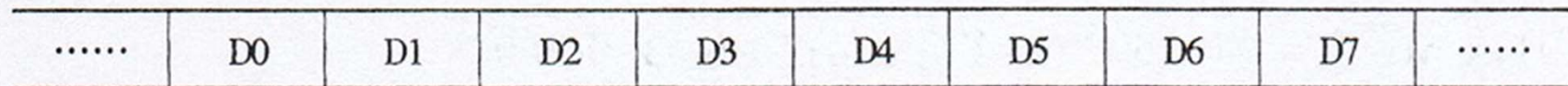
或TB8装入第九位；

接收：数据字节装入读SBUF，第九位装入RB8。

## 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。

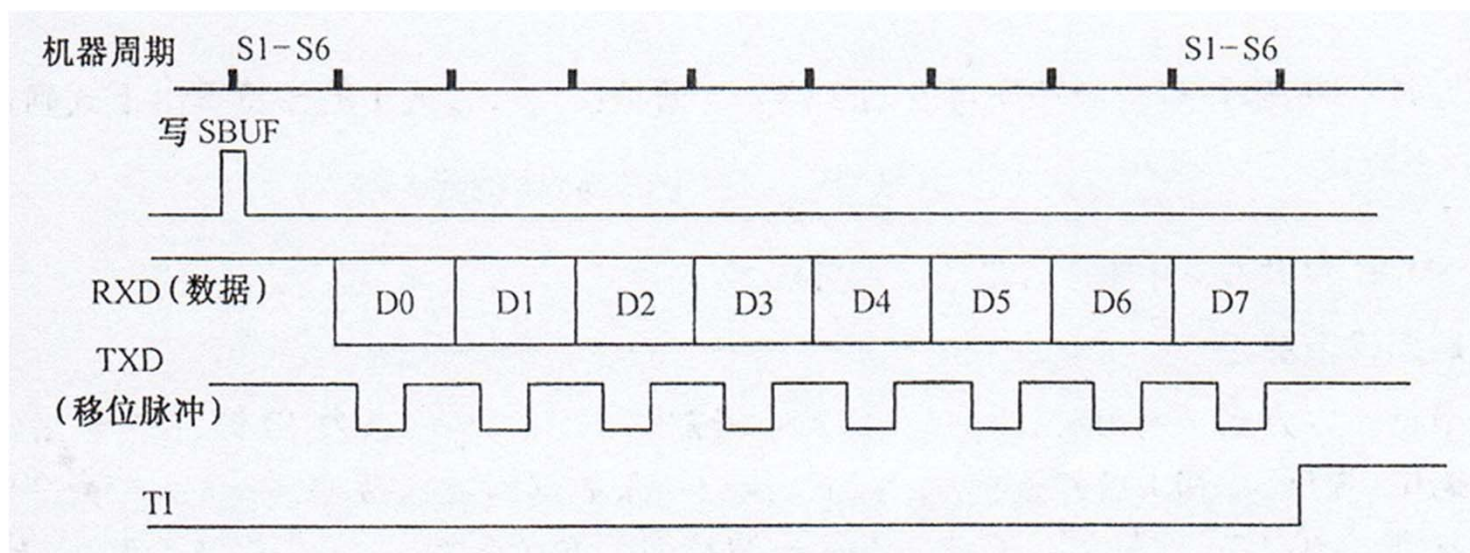
- 8位数据为一帧，无起始位和停止位。发送/接收时，低位在先。
- RXD： 数据输入/输出端。
- TXD： 同步脉冲输出端，每个脉冲对应一个数据位。
- 固定波特率  $B = f_{osc}/12$



方式0的帧格式

## 串行口的4种工作方式——方式0

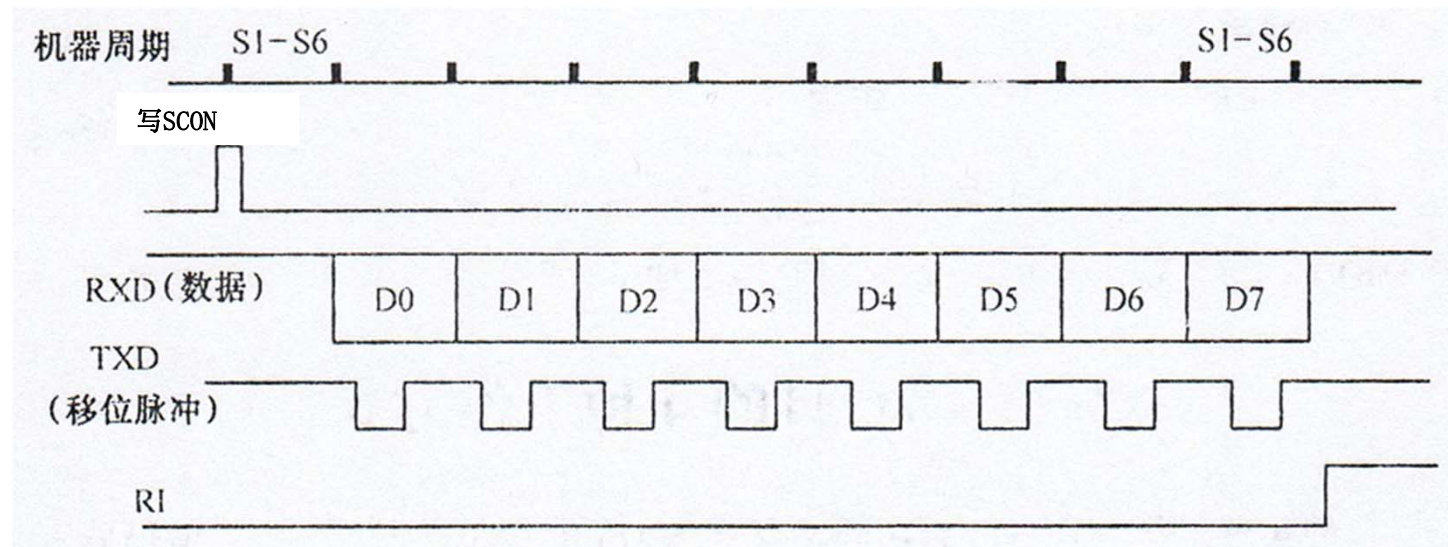
发送过程：CPU执行将数据写入SBUF的指令（如：MOV SBUF, A），启动发送；串行口开始将SBUF中的数据以 $f_{osc}/12$ 的波特率，从RXD引脚输出，TXD引脚输出同步移位脉冲。一帧发送结束，TI置1。



方式0发送时序

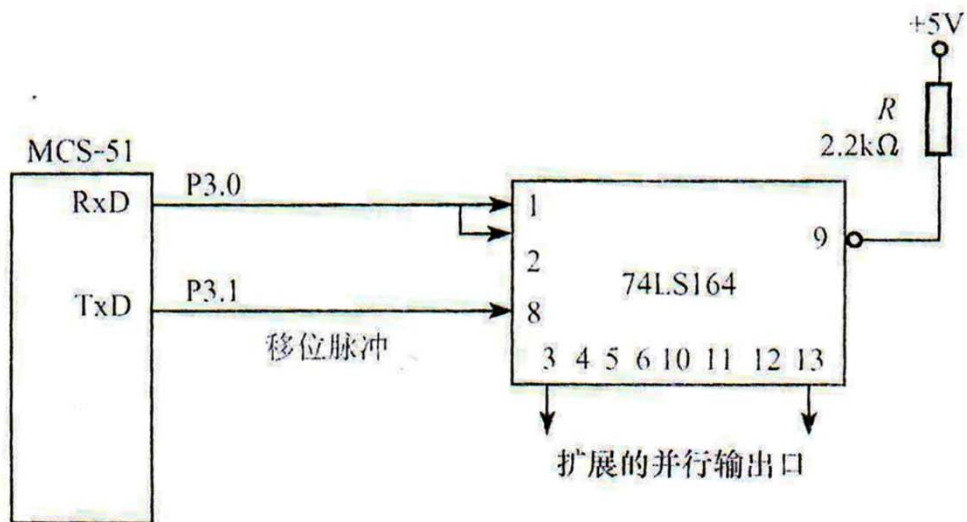
## 串行口的4种工作方式——方式0

**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，接受器以固定 $f_{osc} / 12$ 的波特率采样RXD 。一帧接收完毕， RI置1。

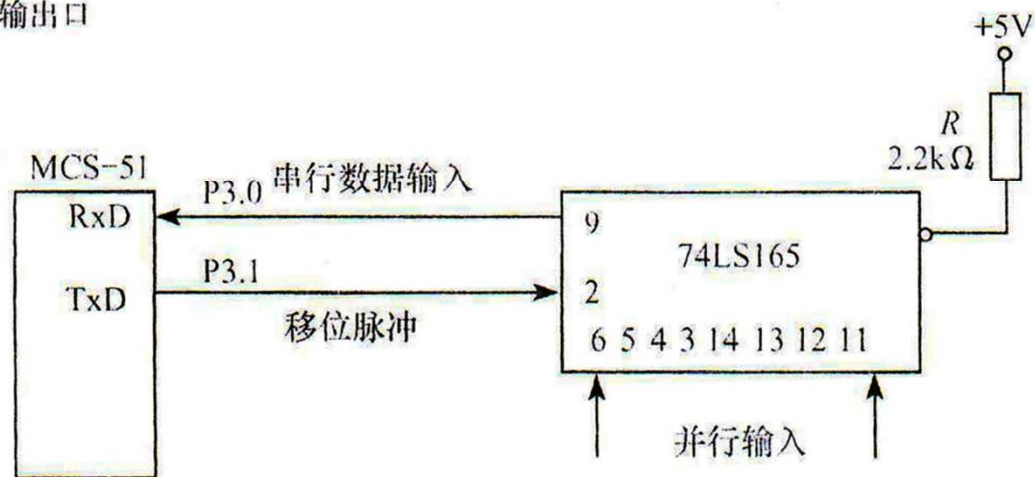


方式0接收时序

## 方式0应用举例



方式0发送接口示例

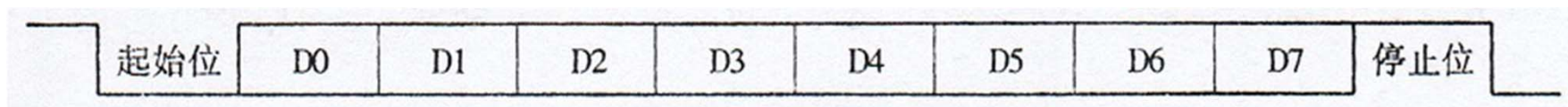


方式0接收接口示例

## 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 =  $\frac{2^{\text{SMOD}}}{32} \times \text{定时器T1溢出率}$

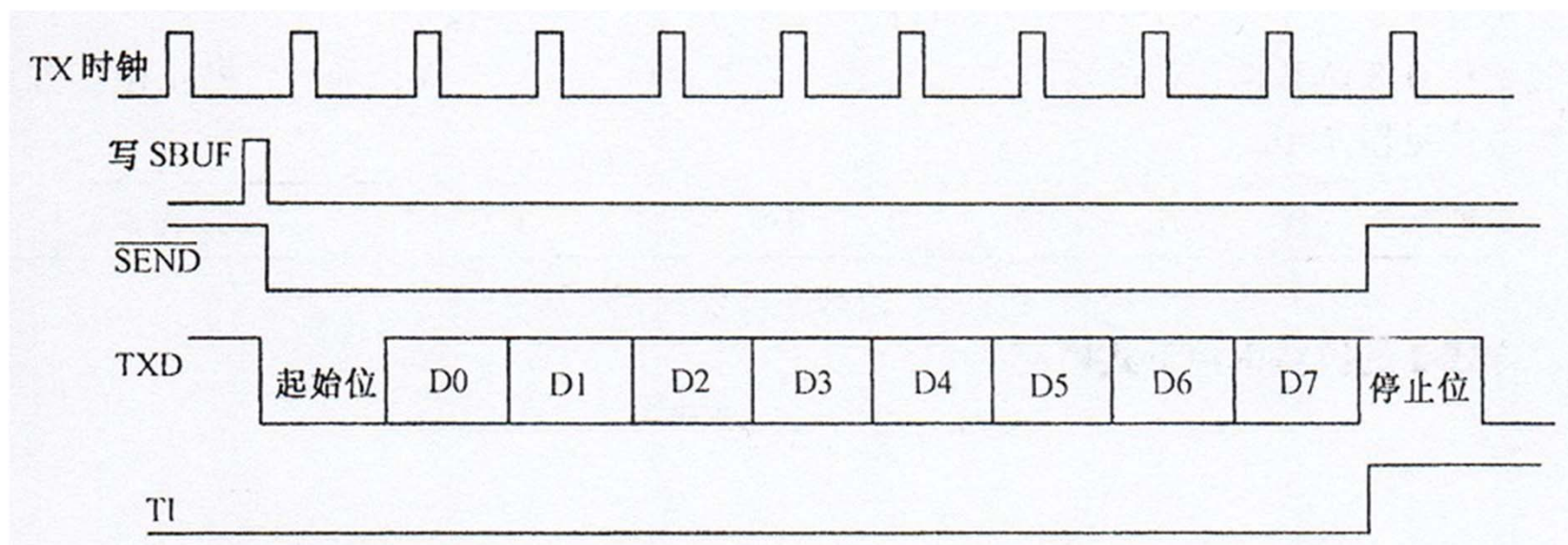


方式1的帧格式



## 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；下图中TX时钟的频率就是发送波特率，串行口开始将SBUF中的数据以波特率从TXD引脚输出（每经过一个TX时钟周期，便产生一个移位脉冲，并由TXD输出一个数据位）。8位数据位发送结束时，TI置1。



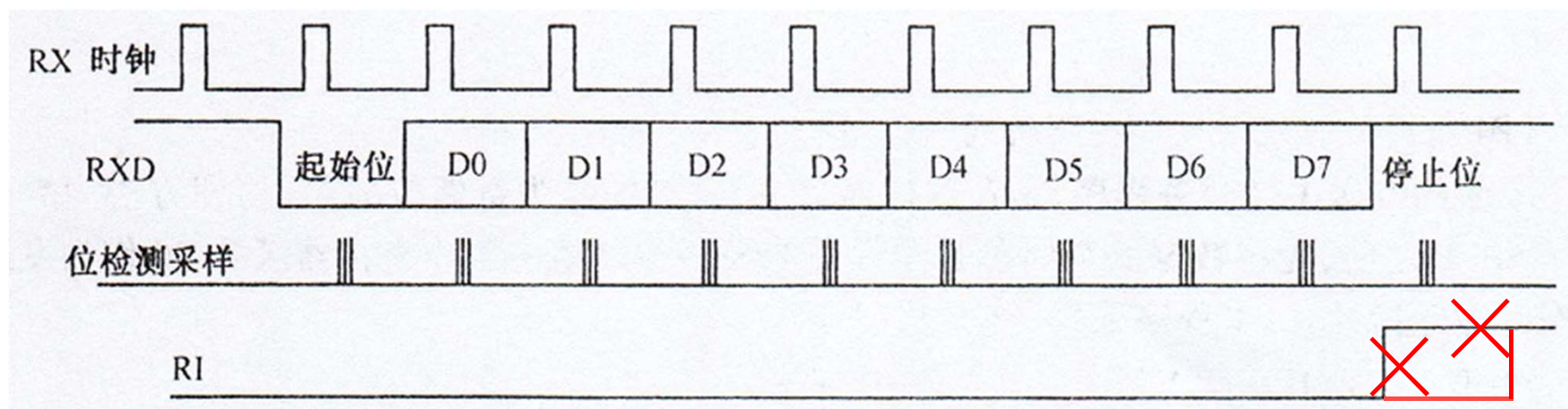
方式1发送时序

## 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

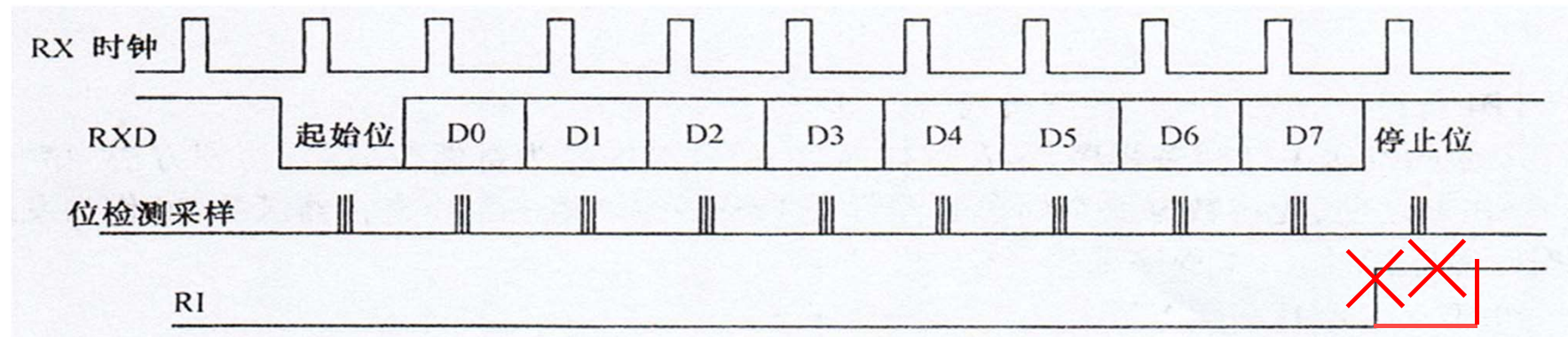


## 串行口的4种工作方式——方式1

### 接收过程：

- 定时控制信号有两种
  - 接收移位时钟（RX时钟）：其频率和波特率相同；
  - 位检测器采样脉冲：以波特率的16倍速率采样RXD引脚电平，在数据位中间，在第7、8、9个脉冲时，连续3次采样进行表决，并3中取2保留采样值。
- 接收过程中，将每个数据位宽度分成16个状态，并在第7、8、9状态时对RXD采样。

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

方式1-3工作原理图

