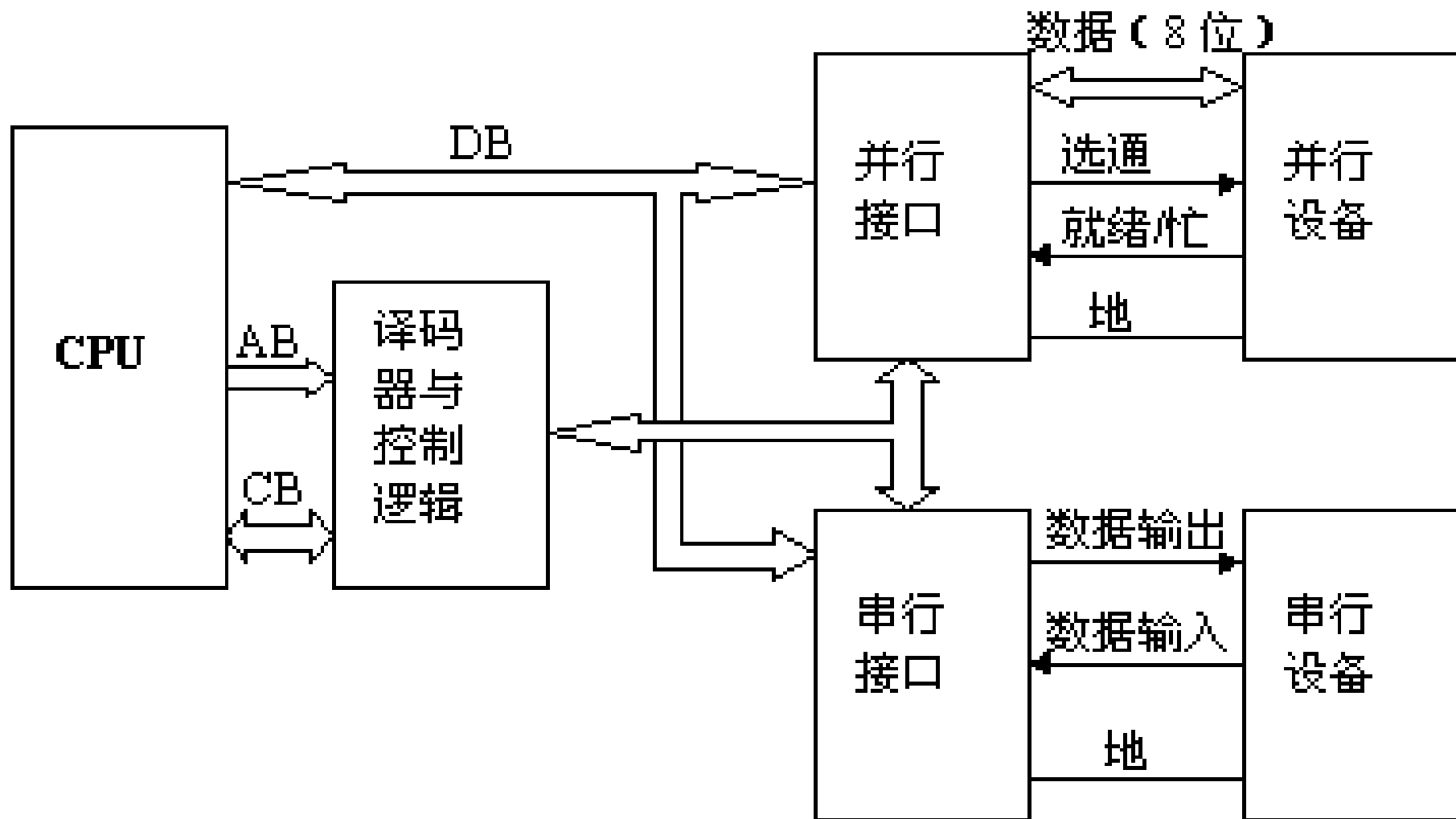


并行接口和串行接口的结构示意图



§ 5.6 串行输入输出接口

主要内容:

- 串行通信的基本概念及异步转送方式
- 串行通信接口标准RS-232C
- 可编程串行接口Ins 8250

要点:

- ◆ 实现异步串行转送的要素有哪些?
- ◆ 可编程串行接口Ins 8250 有哪些功能块及如何编程控制?

一、概述

1. 串行通信的概念

串行通信是指将数据按照一位一位地顺序进行传送，它只占用一条传输线。可以采用两种方式来实现：一种是将8位数据通道中的一位通过软件来实现串行数据传送；另一种是通过专用的通信接口，将并行数据转换为串行数据进行传送。

2. 串行通信的基本方式

通常情况下，串行通信可分为异步传送和同步传送两种方式：

5.6.1 串行通信的基本特点:

1. 数据在一根传输线上一位一位的传送，既传送数据信息，又传送联络控制信息。
2. 串行传送分异步和同步两种固定的数据格式。
3. 串行通信对信号的逻辑定义与TTL不兼容,需进行逻辑关系和逻辑电平的转换。
4. 串行通信既可用于近距离，又可以用于远距离。而后者需要外加MODEM。
5. 串行通信要求双方的速率必须一致，故需进行传输速率控制。
6. 串行通信易受干扰，出错难以避免，故需要进行差错控制。

5.6.2 串行通信接口电路需要解决的问题:

- ① 怎样才能知道数据传输的开始和结束?
- ② 怎样判断所接收数据的正确性?
- ③ 如何进行传输速度控制?
- ④ 如何进行通信双方的外部连接?
- ⑤ 如何进行信号的电平转换与逻辑转换?
- ⑥ 如何进行数据的串/并转换?

实际上, 串行接口设计正是围绕这些问题展开的。并且, 从硬件和软件两方面来解决这些问题。

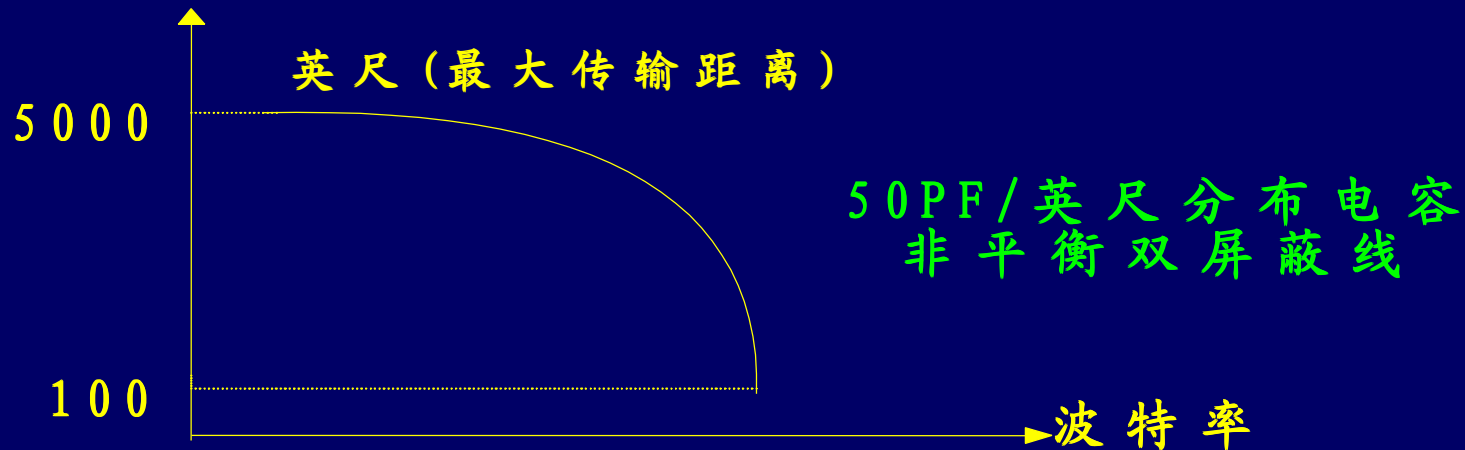
5.6.3 串行通信传输方向(制式):

串行通信中, 数据通常是在二个站(点对点)之间进行传送, 按照数据流的**方向**可分成三种传送模式:

基本传送方式 (按数据流的方向)	{	全双工	<u>图14-1</u>
		半双工	<u>图14-2</u>
		单工	<u>图</u>

5.6.4 串行通信中的调制和解调

计算机的通信要求传送数字信号，而在进行远程数据通信时，线路往往是借用现有的公用电话网，电话网是为音频模拟信号的设计的。一般为300~3400Hz，不适合于数字信号。



对于同一种传输线，传输的距离随传输率的增加而减少。

因此需要对二进制信号进行**调制**，以适合在电话网上传输相应的音频信号，

在接收时，需要进行**解调**，还原成数字信号。

调制器 MODULATOR (数据通信设备DCE)

在发送时将二进制信号调制成相应的音频信号

解调器 DE-MODULATOR (数据通信设备DCE)

把从通信链路上收到的模拟音频信号转换成数字信号

系统结构如下图所示

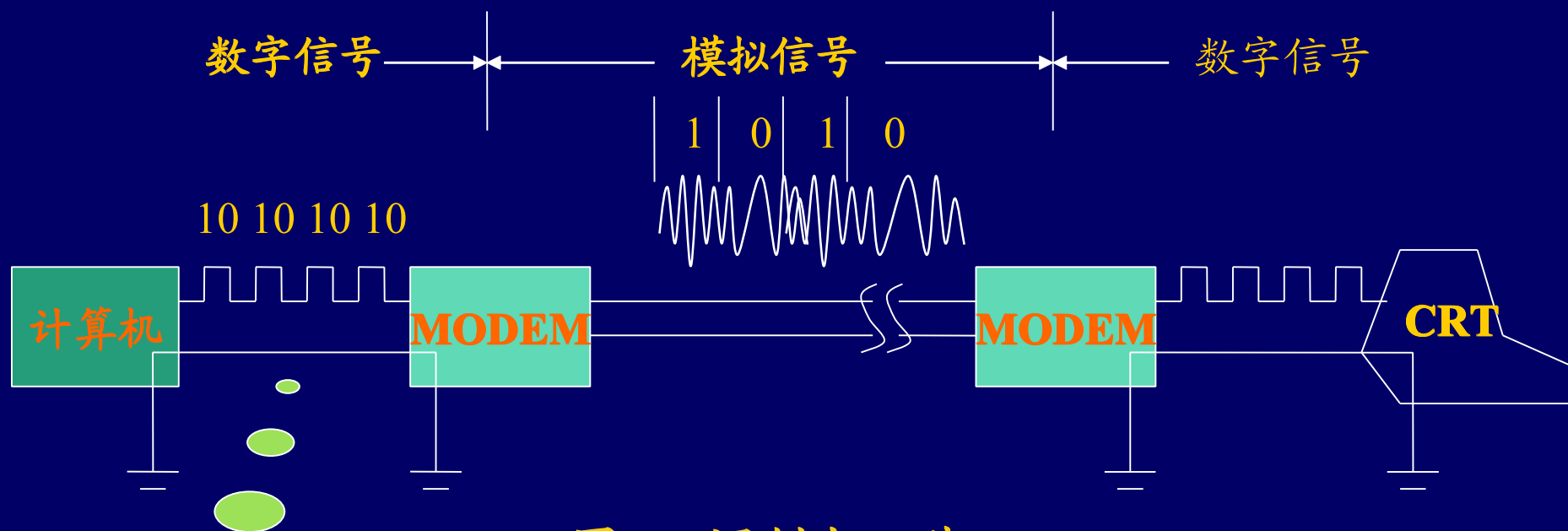


图 调制电话线

RS-232接
口标准

调制器和解调器常做在一个装置中
=>调制解调器 **MODEM**

- MODEM的类型

调制解调器

振幅键控 (ASK): 用调幅 (AM) 法

频移键控 (FSK): 用调频 (FM) 法

相移键控 (PSK): 用调相 (PM) 法

[返回](#)

调制方法:

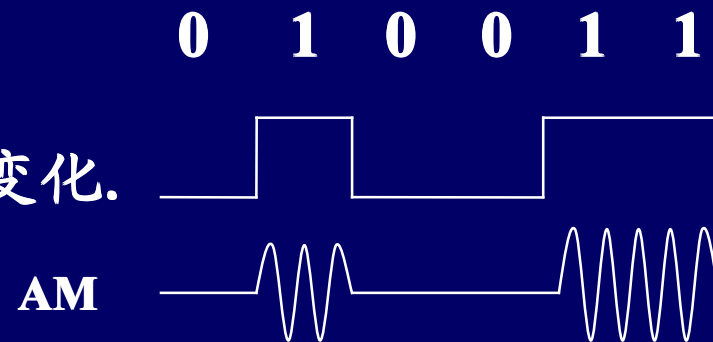
最基本的调制方法有以下几种:

(1) 调幅 (AM)

即载波的振幅随基带数字信号而变化.

“1”对应有载波

“0”对应无载波

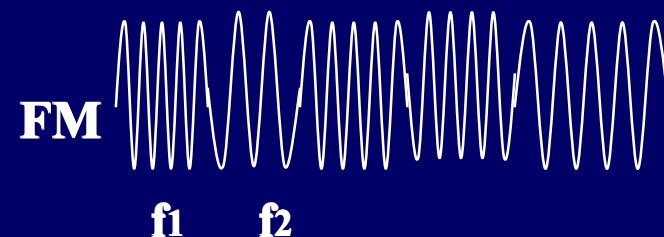


(2) 调频 (FM)

即载波频率随数字信号而变化

“0”对应“f1”

“1”对应“f2”

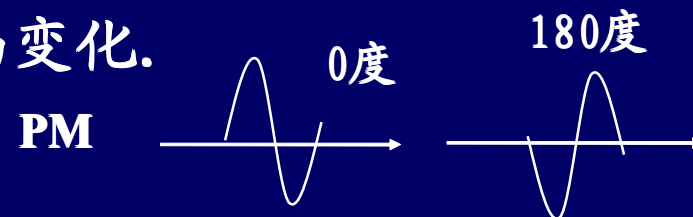


(3) 调相 (PM)

即载波初始相位随基带数字信号而变化.

“0”对应相位0度

“1”对应相位180度



5.6.5 串行通信中的差错控制

1. 误码率的控制

误码率: 是指数据经传输后发生错误的位数与总传输位数之比。

- 从硬件和软件着手对通信系统进行可靠性设计, 以达到尽量少出差错的目的;
- 对所传输的信息采用检纠错编码技术, 以便及时发现和纠正传输过程出现的差错。

3. 错误状态的分析与处理

异步串行通信过程中常见的错误有奇偶检验错、溢出错、帧格式错。这些错误状态一般都存放在接口电路的状态寄存器中, 以供CPU进行分析和处理。

2. 检纠错编码方法的使用

检错: 如何发现错误

检错 { 奇偶校验 (用于基本通信规程中)
方阵校验 (用于基本通信规程中)
循环冗余码 (CRC) (用于高级通信规程中)

纠错: 如何纠正错误

反馈重发方式-----基本

自动纠错方法-----高级

信息传输的可靠性是以牺牲传输效率为代价的。

3. 错误状态的分析与处理

异步串行通信过程中常见的错误有奇偶检验错、溢出错、帧格式错。这些错误状态一般都存放在接口电路的状态寄存器中，以供CPU进行分析和处理。

- ① **奇偶校验错**：在接收方接收到的数据中，1的个数与奇偶校验位不符。这通常是由噪声干扰而引起的，发生这种错误时接收方可要求发送方重发。

奇偶校验

检查每个字符（包括检验位）中“1”的总个数是奇数或偶数。

校验位

如：偶校验	11000011	0	保证为偶数个1
	00001101	1	保证为偶数个1

② **溢出错**: 接收方没来得及处理收到的数据, 发送方已经发来下一个数据, 造成数据丢失。这通常是由收发双方的速率不匹配而引起的, 可以采用降低发送方的发送速率或者在接收方设置FIFO缓冲区的方法来减少这种错误。

③ **帧格式错**: 接收方收到的数据与预先约定的格式不符。这种错误大多是由于双方数据格式约定不一致或干扰造成的, 可通过核对双方的数据格式减少错误。

④ **超时错**: 在查询方式的通信程序中出现。

一般由接口硬件电路速度跟不上而产生。

4. 错误校验只在接收方进行

一般是在接收程序中，采用软件编程方法，从接口电路的状态寄存器中，读出错误状态位，判断有无错误，进行检测，或者通过调用BIOS软中断INT14H的状态查询子程序来检测。

5.6.6 串行通信中的传输速率控制

1. 数据传输速率控制的实现方法

串行通信时，要求**双方的传输速率严格一致**。并在传输开始之前，要预先设定，否则，会发生错误。因此，对传输速率要进行控制。

在数字通信中，传输速率也常称作为波特率，单位是波特。

2. 波特率与发送/接收时钟

(1) 什么是波特率

波特率：每秒钟传输的二进制位数 (bit/s)

位周期：传送1位所需的时间

波特率 = $1 / \text{位周期}$

常用的标准波特率是：

110, 300, 600, 1200, 2400

4800, 9600, 19200, 115200, 38400b/s

- 例如, 串行通信的数据传输率为1200b/s, 则每一个数据位的传输时间 T_d 为波特率的倒数:

$$T_d = 1 \text{ 位/Baud} = 1 \text{ b} / 1200 \text{ b/s} = 0.833 \text{ ms}$$

- 例: 在异步串行通信中, 传送一个字符有12位 (一个起始位, 8个数据位, 一个奇偶位, 2个停止位), 若波特率是1200b/s:

则每秒钟能传送的字符数是?

$$1200 / 12 = 100 \text{ 个, 即字符速率 } 100 \text{ 个/秒}$$

(2) 发送/接收时钟

在串行传输过程中，二进制数据序列是以数字信号波形的形式出现，要把这个数字波形定时发送出去，或接收进来，发方和收方之间的数据传输怎样同步，由此引出发送/接收时钟。

(3) 波特率因子

发送/接收时钟频率 (Tx_c/Rx_c)

=波特率 (Baud) × 波特因子 (Factor)

|

常取1,16,64

例子

(4) 波特率时钟发生器

由专门的波特率时钟发生器来产生串行通信所需的各种波特率的时钟脉冲。

有些可编程串行接口芯片内部内嵌了波特率时钟发生器（如**8250**、**16450**、**16550**），有些则没有（如**8251**）。

(5) 波特率时钟的使用

① 在波特因子选定的情况下，可利用改变发送/接收时钟频率来控制串行通信的波特率。

② 在串行通信的收发过程中，为了保证通信的正确性，收发双方应该使用相同的波特率。但是，双方所使用的发送时钟和接收时钟的频率可以不同。

通过 $T_{xc}/R_{xc} = \text{Baud} \times \text{Factor}$ 式，调整波特率因子，来确保双方的波特率保持一致。

例14.2 甲乙两机进行串行通信：

甲机的发送时钟频率 $T_{xC}=38400\text{Hz}$ ，波特率因子 $\text{factor1}=16\text{b}^{-1}$ ；

乙机选用的波特率因子 $\text{factor2}=64\text{b}^{-1}$ 。

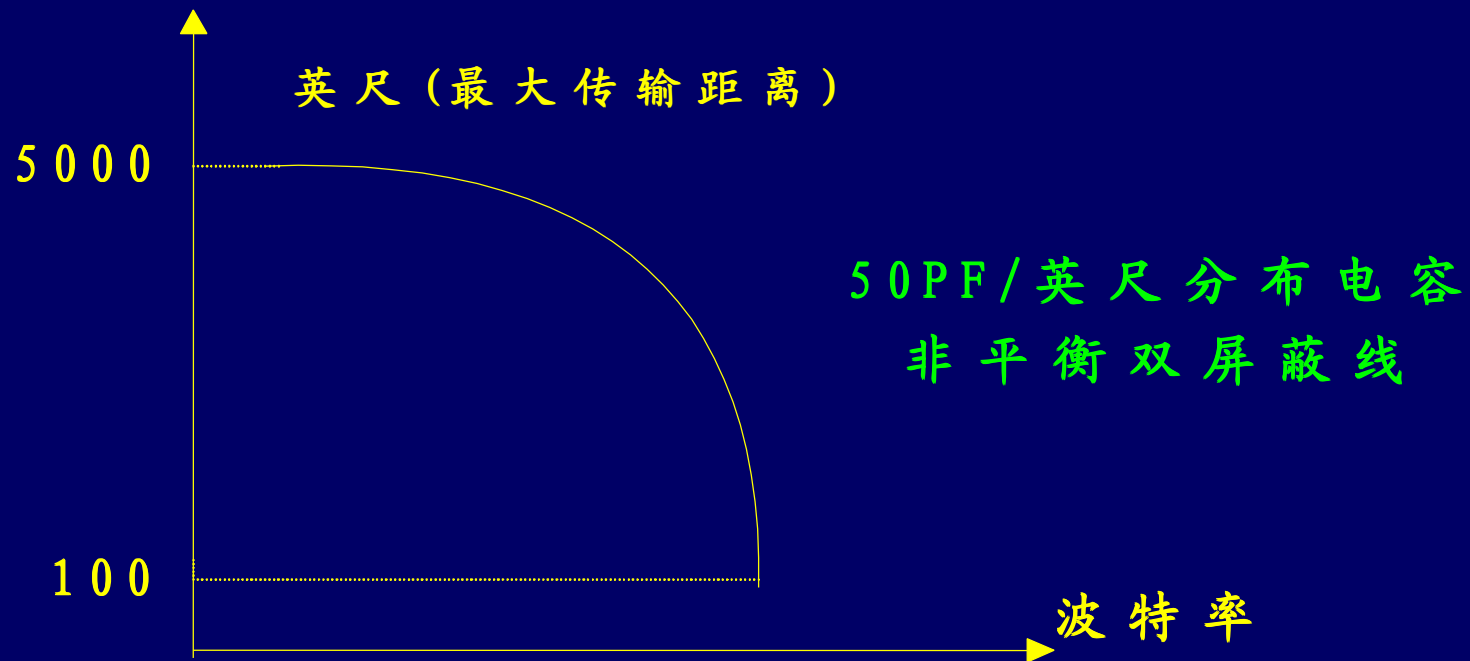
若要使双方的波特率保持一致，则乙机的接收时钟 R_{xC} 应为多少？

解：甲机的波特率： $\text{Baud}=T_{xC1}/\text{factor}=38400\text{Hz}/16 \text{ b}^{-1}=2400\text{b/s}$

乙机的接收时钟： $R_{xC}=\text{factor2} \times \text{Baud}$
 $=64\text{b}^{-1} \times 2400\text{b/s}=153600\text{Hz}$

可见，甲乙两机的发送/接收时钟脉冲的频率虽然不同，但是通过波特率因子的改变，仍然可以使两者的波特率保持一致。不过这适应于异步通信，对同步通信其双方的发收时钟要严格一致。

3. 传输距离与传输速率的关系



对于同一种传输线，传输的距离随传输率的增加而减少。

5.6.7 串行通信中的同步问题

串行传输的一个重要问题就是接收端如何判断数据何时开始。

1. 字符同步的方案

① 同步通信（块同步）对双同步通信（BISYNC），接收器通过搜索1~2个特定的同步字符来判断1个数据块的开始。对高级数据链路控制同步通信（HDLC），接收器通过搜索特定字符（01111110）来判断一个数据块的开始。

② 异步通信（字符同步）对起止式异步通信，接收器通过检测起始位来判断一个数据字符的开始。

2. 位同步方案

接收器通过时钟信号来接收每一位数据。规定若干个（如16个）时钟脉冲就接收一位数据。

5.6.8 串行通信的基本方式

(1) 异步串行通信方式:

基本特点: 以字符为信息单位传送。每个字符作为一个独立的信息单位（1帧数据），可以随机出现在数据流中。字符与字符之间是异步的，字符内部位与位之间的传输是同步的。

(2) 同步串行通信方式:

基本特点: 以数据块（字符块）为信息单位传送，每帧信息包括成百上千个字符，**收/发两端必须使用同一时钟**来控制数据块传输中字符与字符、字符内部位与位之间的定时。

因此:

异步 串行通信一般用在数据传送时间不能确知, 发送数据不连续, 数据量较少和数据传送速率较低の場合;

同步 串行通信则用在要求快速、连续传输大批量数据の場合。



5.6.9 串行通信中的数据格式

通信协议:

是指通信双方的一种约定, 包括对数据格式、同步方式, 传送速度、传送步骤、检纠错方式等问题作出统一规定。也称通信控制规程。

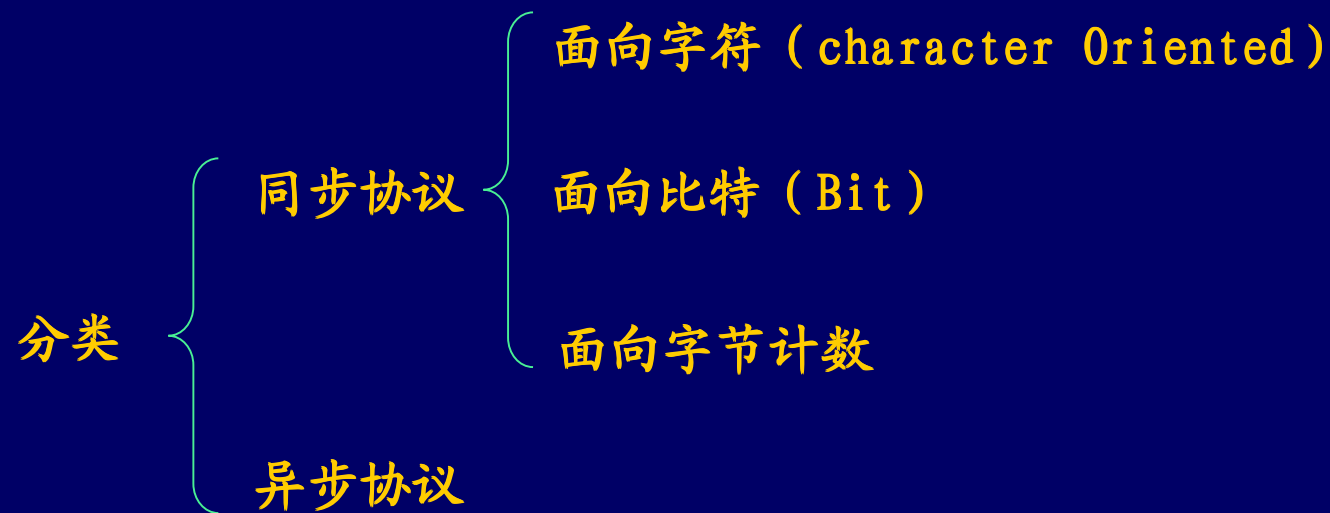
ISO (Inter national Standard Organization)

国际标准化组织。

OSI (Open System Interconnection)

开放系统互连参考模型。

通信控制规程属于**ISO'S OSI**七层参考模型中的
数据链路层



一、起止式异步通信数据格式

二、面向字符的同步通信数据格式

5.6.9.1 起止式异步通信数据格式

1. 起止式数据帧格式



异步通信是以字符为单位进行传输的。每帧信息（即每个字符）由4个部分组成。

- ① 1位起始位（低电平，逻辑值0）。
- ② 5~8位数据位紧跟在起始位后，是要传输的有效信息。规定从低位至高位依次传输。
- ③ 1位校验位（也可以没有校验位）。
- ④ 最后是1位，或1位半，或2位停止位，停止位后面是不定长度的空闲位。停止位和空闲位都规定为高电平（逻辑值1），这样就保证起始位开始处一定有一个下跳沿。

2. 起止位的作用

作为联络信号而附加进来的，为通信双方提供了何时开始收发、何时结束的标志。

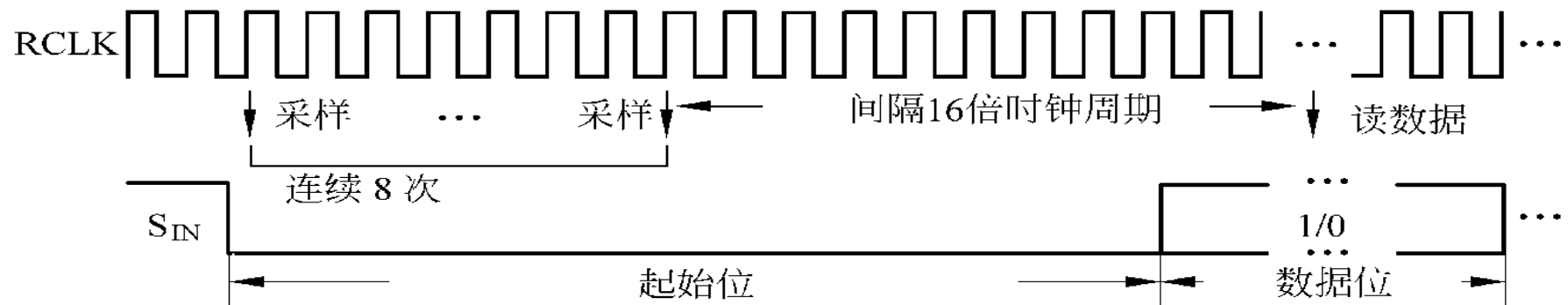
优点：

异步串行通信的可靠性高，而且，也比较易于实现。

缺点：

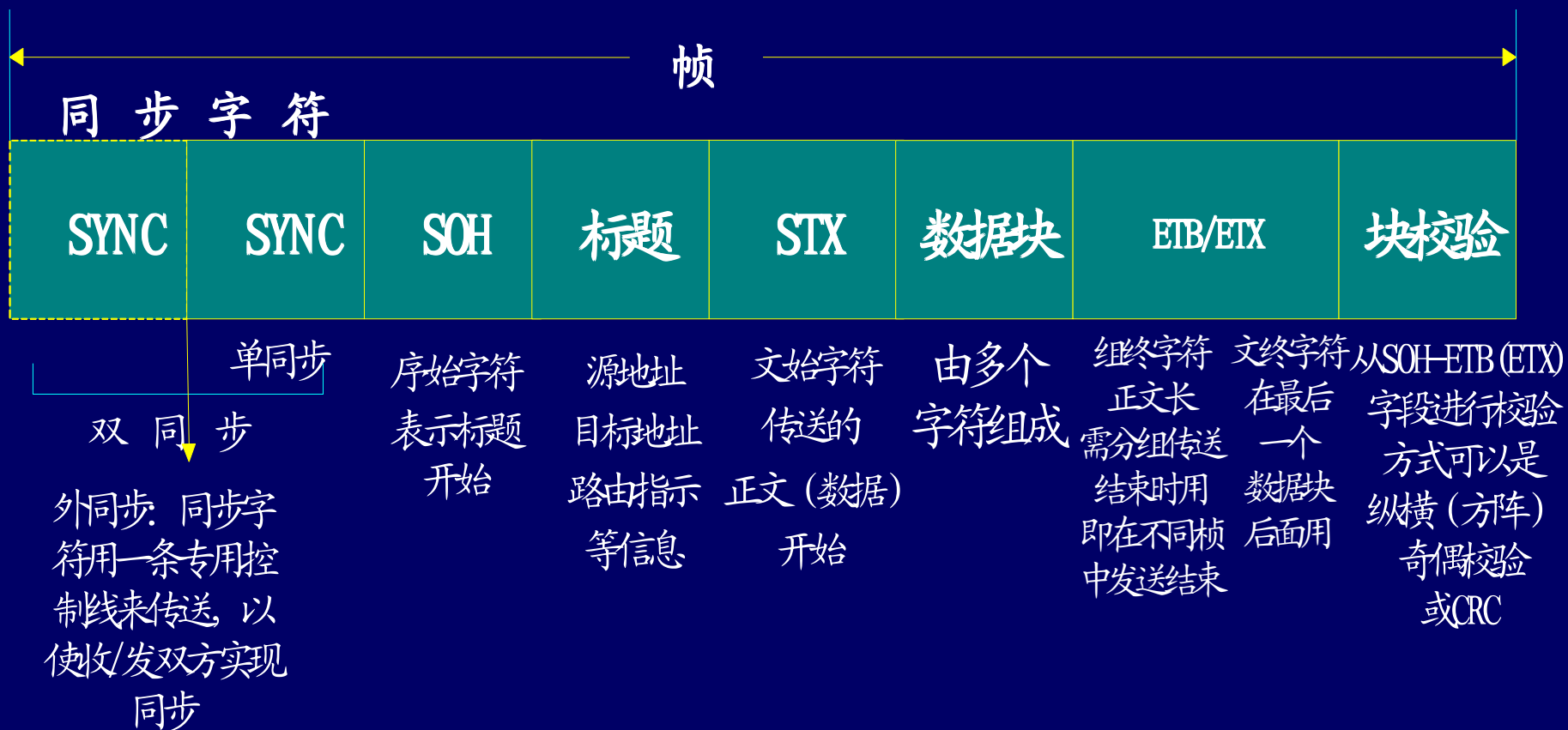
★ 附加位使得传输有用（效）的数据位减少，即传输效率变低了（只有约80%）。

★ 数据帧之间有空闲位，故数据传输速率慢。



P290 图5.54 数据接收过程

5.6.9.2 面向字符的同步通信数据格式



5.6.10 串行通信接口标准

1. RS-232C接口标准
2. RS-485接口标准（多发送器标准）

1. EIA-RS-232接口标准

RS-232标准是美国EIA（电子工业联合会）与BELL等公司一起开发的，1969年公布的通信协议，适合的数传率：0~20Kbps。

- ❖ 最初是为远程通信连接数据终端设备DTE 和数据通信设备DCE而制定的
- ❖ 通过一个连接器来体现
- ❖ “发送”和“接收”都是站在DTE（终端）的立场上

DTE——数据终端设备，例如计算机

DCE——数据通信设备（数传机），例如调制解调器



1. EIA-RS-232C标准的信号线的功能定义

- TxD : 发送数据线
 - RxD : 接收数据线
 - RTS : 请求发送线
 - CTS : 允许发送线
 - DSR : 数据装置准备好
 - DTR : 数据终端装置准备好
 - DCD : 数据载波输出
 - RI : 振铃指示
 - SG : 信号地
- 数据线
- 联络线
- 状态线

RS-232C的引脚（1）

TxD: 发送数据（终端→数传机）

- 串行数据的发送端

RxD: 接收数据（终端←数传机）

- 串行数据的接收端

RS-232C的引脚（2）

RTS: 请求发送（终端→数传机）

- 当数据终端设备准备好送出数据时，就发出有效的RTS信号，用于通知数据通信设备准备接收数据

CTS: 清除发送（允许发送）（终端←数传机）

- 当数据通信设备已准备好接收数据终端设备的传送数据时，发出CTS有效信号来响应RTS信号

RTS和CTS是数据终端设备与数据通信设备间一对用于数据发送的联络信号

RS-232C的引脚（3）

DTR: 数据终端准备好（终端→数传机）

- 通常当数据终端设备一加电，该信号就有效，表明数据终端设备准备就绪

DSR: 数据装置准备好（终端←数传机）

- 通常表示数据通信设备（即数据装置）已接通电源连接到通信线路上，并处在数据传输方式

DTR和DSR也可用做数据终端设备与数据通信设备间的联络信号，例如应答数据接收

RS-232C的引脚（4）

GND: 信号地

- 为所有的信号提供一个公共的参考电平

CD: 载波检测（DCD）（终端←数传机）

- 当本地调制解调器接收到来自对方的载波信号时，该引脚向数据终端设备提供有效信号

RI: 振铃指示（终端←数传机）

- 当调制解调器接收到对方的拨号信号期间，该引脚信号作为电话铃响的指示、保持有效

RS-232C的引脚（5）

保护地（机壳地）

- 起屏蔽保护作用的接地端，一般应参照设备的使用规定，连接到设备的外壳或大地

TxC：发送器时钟

- 控制数据终端发送串行数据的时钟信号

RxC：接收器时钟

- 控制数据终端接收串行数据的时钟信号

2. 信号线的使用

1) 远距离：传输距离大于15m的通信

a. 需用MODEM和专用电话线

b. 需用2~9条信号线（在接口与MODEM之间）

图 (a) 图 (b)

2) 近距离：不用MODEM 图 图14-8

3. 电气特性

1) 在TxD和RxD数据线上

逻辑“1”= $-3V \sim -15V$

“0”= $+3V \sim +15V$

2) 在RTS、CTS、DSR、DTR、CD等控制线上

信号有效（接通、ON状态、正电压）= $+3V \sim +15V$

信号无效（断开、OFF状态、负电压）= $-3V \sim -15V$

EIA与TTL之间的区别:

差异	EIA	TTL
逻辑关系	负逻辑	正逻辑
逻辑电平	高 ($\pm 15V$)	低 ($+5V$)
电平摆幅	大 ($-15V \sim +15V$)	小 ($0 \sim 5V$)

因此采用TTL/EIA电平转换器进行逻辑电平和逻辑关系转换:

如: MC1488、SN75150: TTL \rightarrow EIA (需正负15V)

MC1489、SN75154: EIA \rightarrow TTL (需正负15V)

MAX232 EIA \longleftrightarrow TTL (+5V供电)

RS-232C的电气特性

232C接口采用EIA电平（负逻辑）

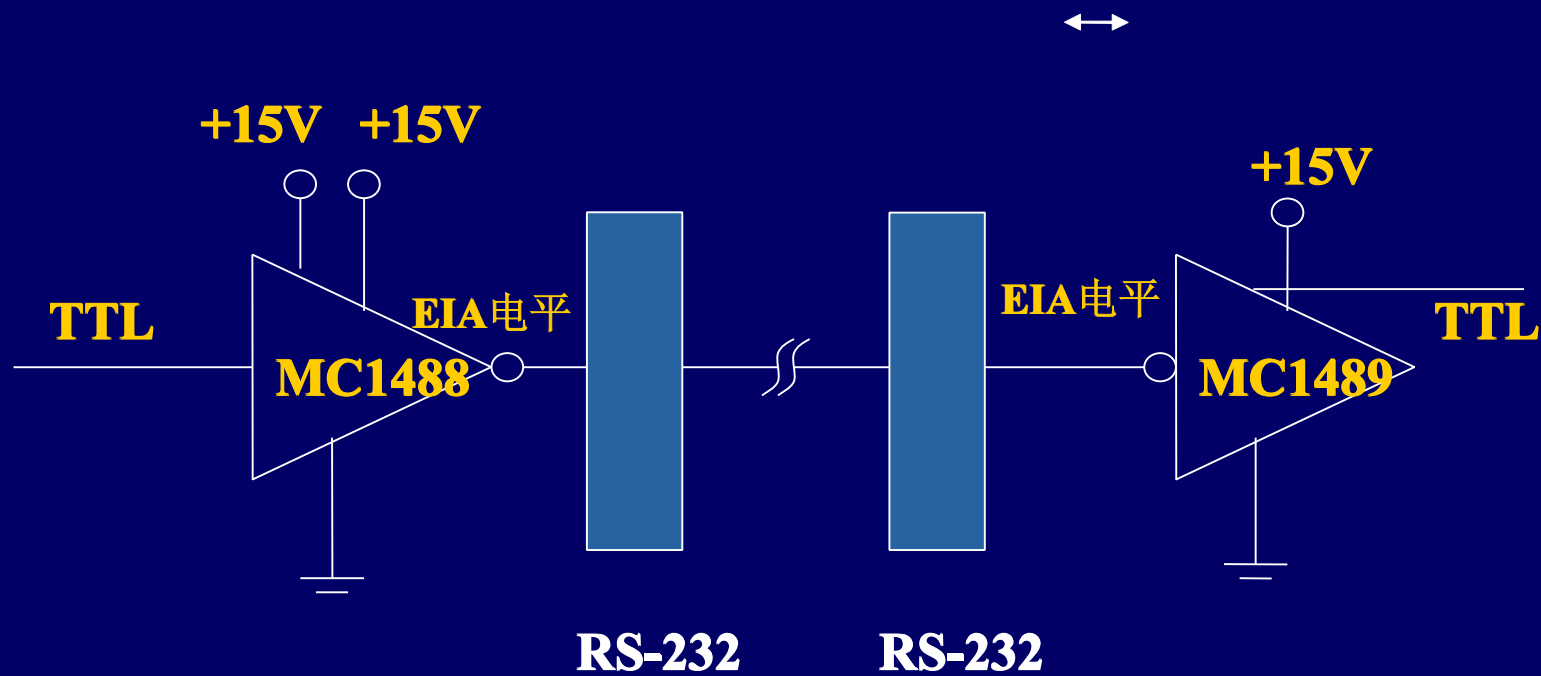
- “0”电平为 $+3V \sim +15V$
- “1”电平为 $-3V \sim -15V$
- 实际常用 $\pm 12V$ 或 $\pm 15V$

■ 标准TTL电平（正逻辑）

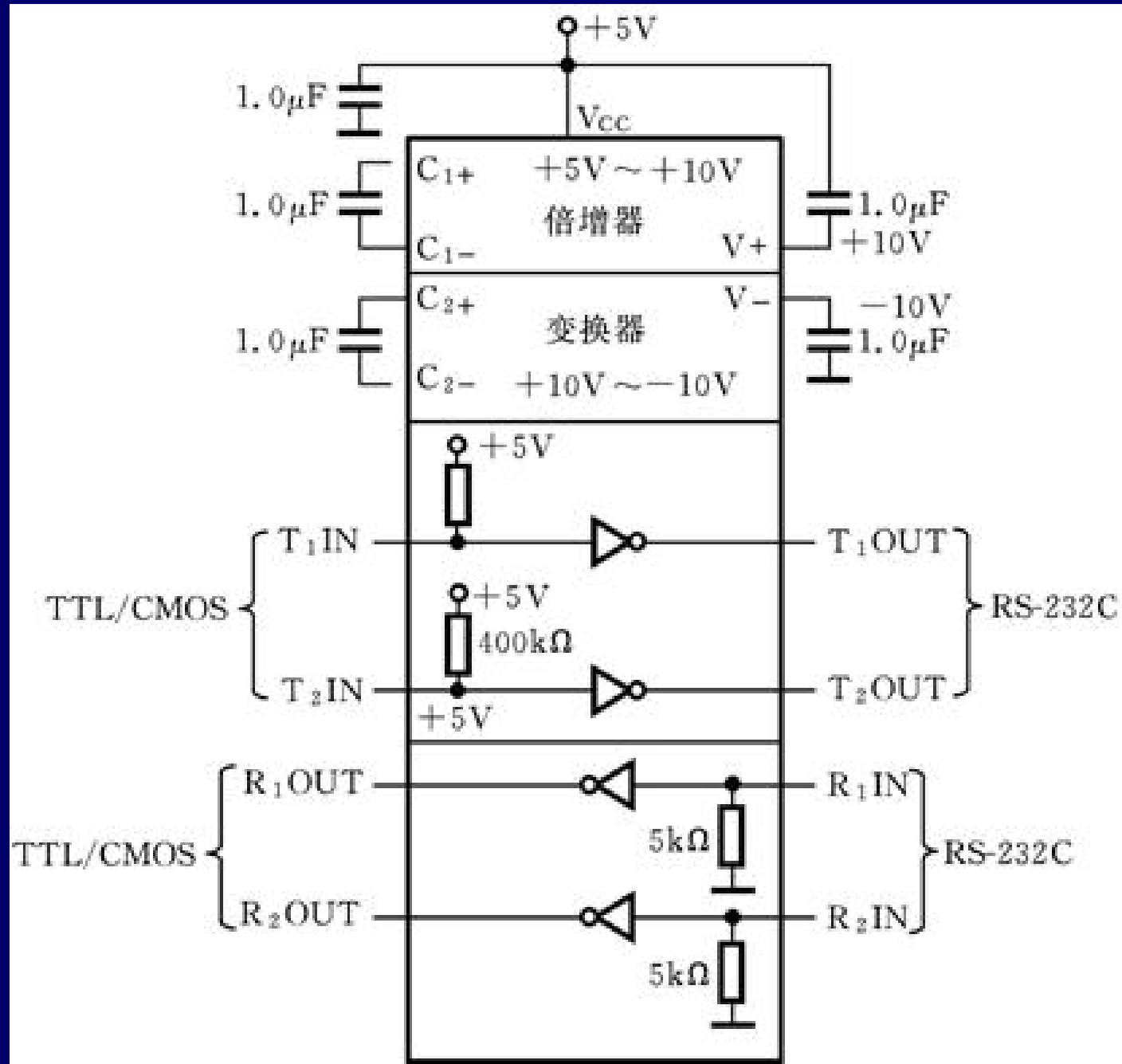
- “1”电平： $+2.4V \sim +5V$
- “0”电平： $0V \sim 0.8V$

相互转换





采用EIA电平比TTL电平具有更强的抗干扰性能。
MAX232省电可连接二对收/发线，只用单电源。

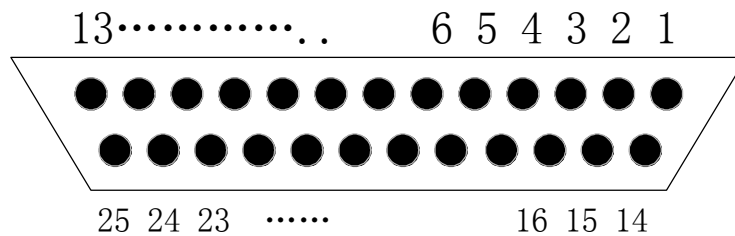


MAX232内部
逻辑框图

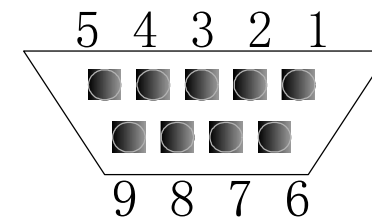
4. 机械特性

1) 连接器 (Connector)

串行通信**RS-232**是一种总线标准，这个标准仅保证硬件兼容而没有软件兼容。通常**RS-232C**接口有**9针**、**25针**等规格，**RS-232C**标准接口的引脚排列。



(a) 25脚排列图

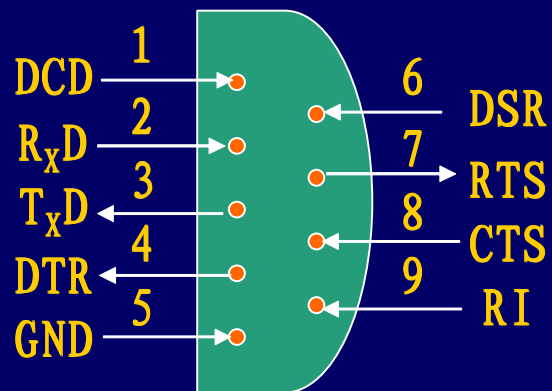


(b) 9脚排列图

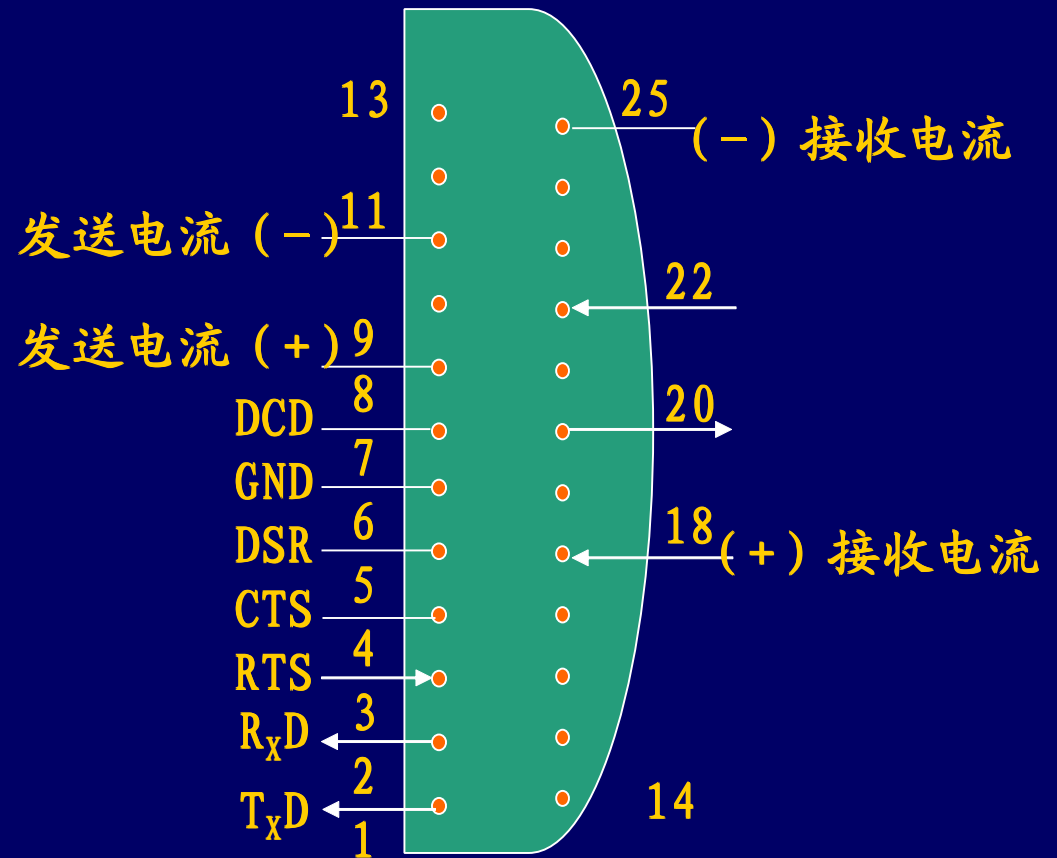
RS-232C引脚排列

①DB-25型，25脚，用9个信号（2个数据线，6个控制线，1个地址线）

②DB-9型 9针（全用）



DB-9型连接器



DB-25型连接器

(2) 通信电缆长度

指在通信传输速率低于20Kb/s时，RS-232C的电缆所能直接连接（不采用MODEM）两台计算机或终端的最大物理距离（15m）

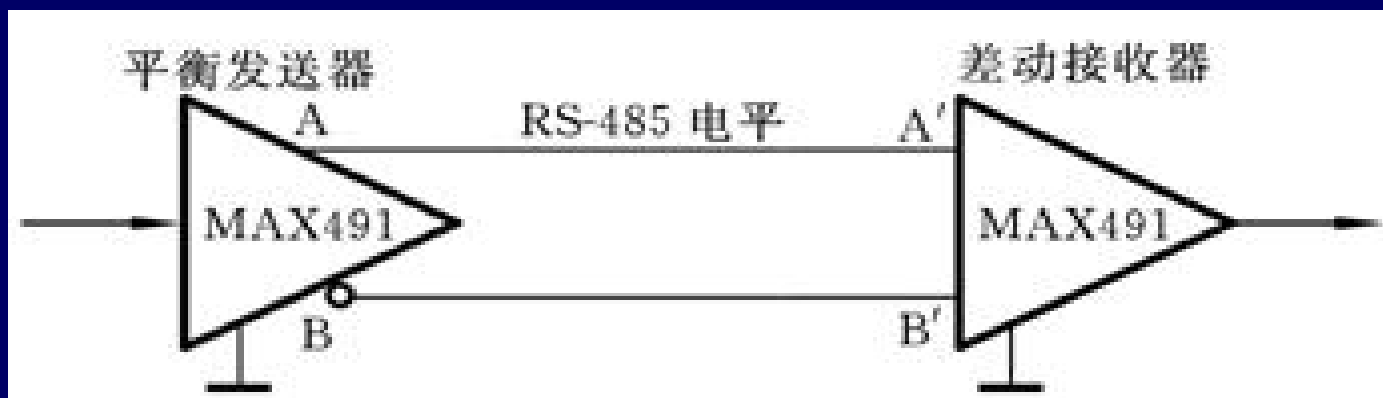
这是根据RS-232C规定最大负载电容为2500pF的要求计算出来的。

5.6.11 RS-485接口标准（多发送器标准）

1. RS-485接口标准的新技术与新概念

RS-485是RS-232C的改进型标准

（1）采用双线平衡方式传输



（2）采用电位差值定义信号逻辑

采用两条传输线之间的电位差值来定义逻辑**1**和逻辑**0**。
当**AA'**线的电平比**BB'**线的电平高**200mV**时表示逻辑**1**；
当**AA'**线的电平比**BB'**的电平低**200mV**时表示逻辑**0**

(3) 采用4芯水晶头连接器

4芯水晶头连接器类似于电话线的接头，比RS-232C标准的DB-9型或DB-25型连接器使用方便且价格低廉。

(4) 单线—双线—单线转换

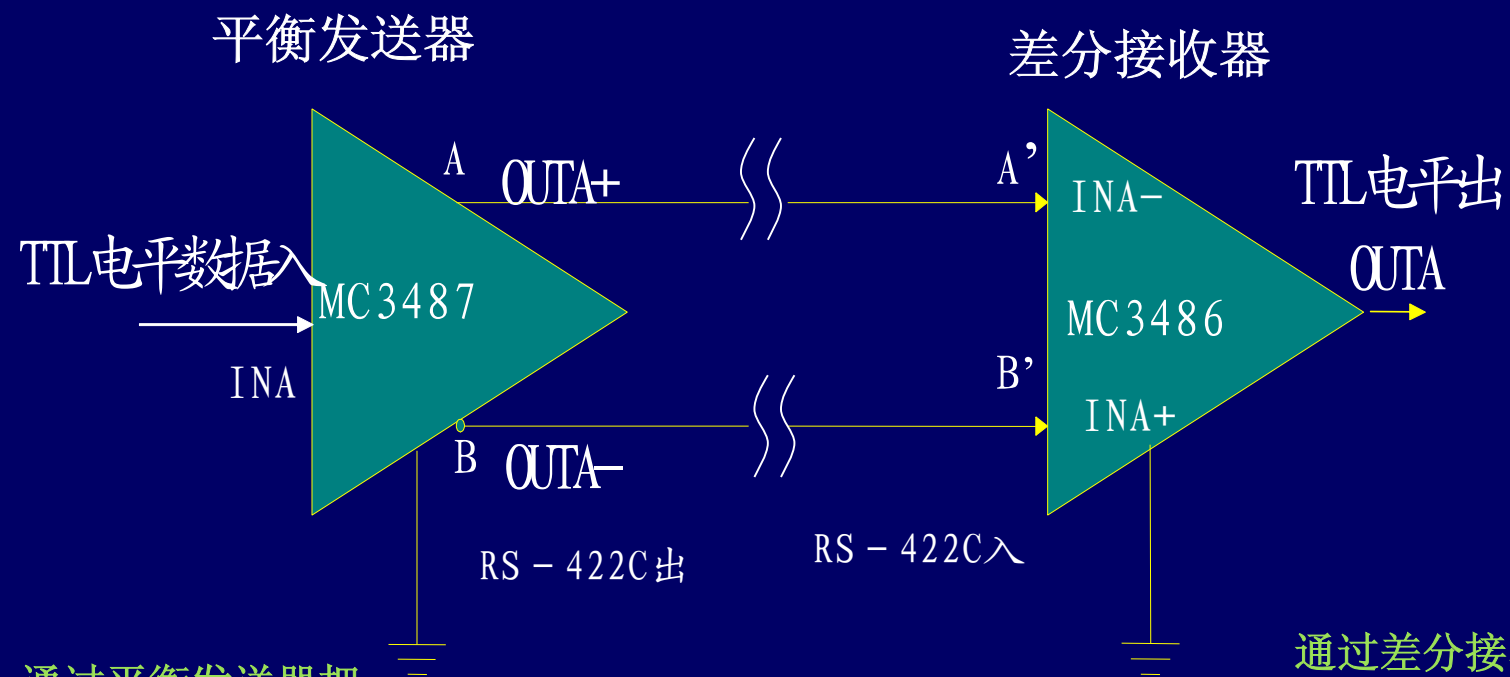
通过平衡发送器（例如，MAX491）把信号的逻辑电平变换成信号的电位差，完成始端的信息发送；

通过差动接收器（例如，MAX491），把电位差值变换成信号的逻辑电平，实现终端的信息接收。

收发器芯片：

MAX485：用于半双工

MAX491：用于全双工



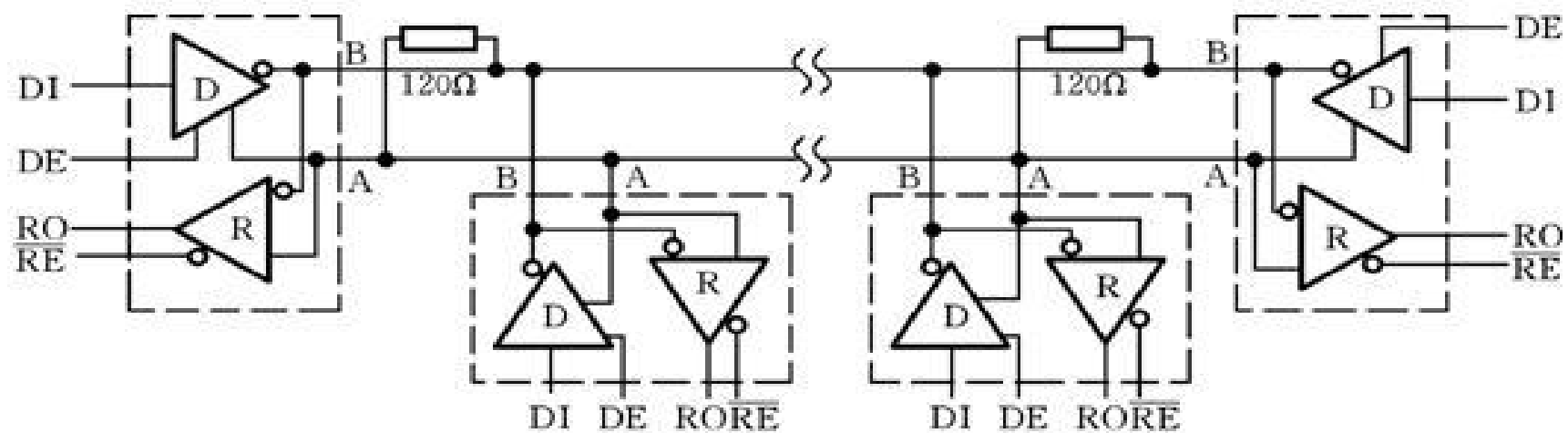
通过平衡发送器把逻辑电平变换成电位差，完成始端的信息传送

通过差分接收器把电位差变成逻辑电平，实现终端信息接收

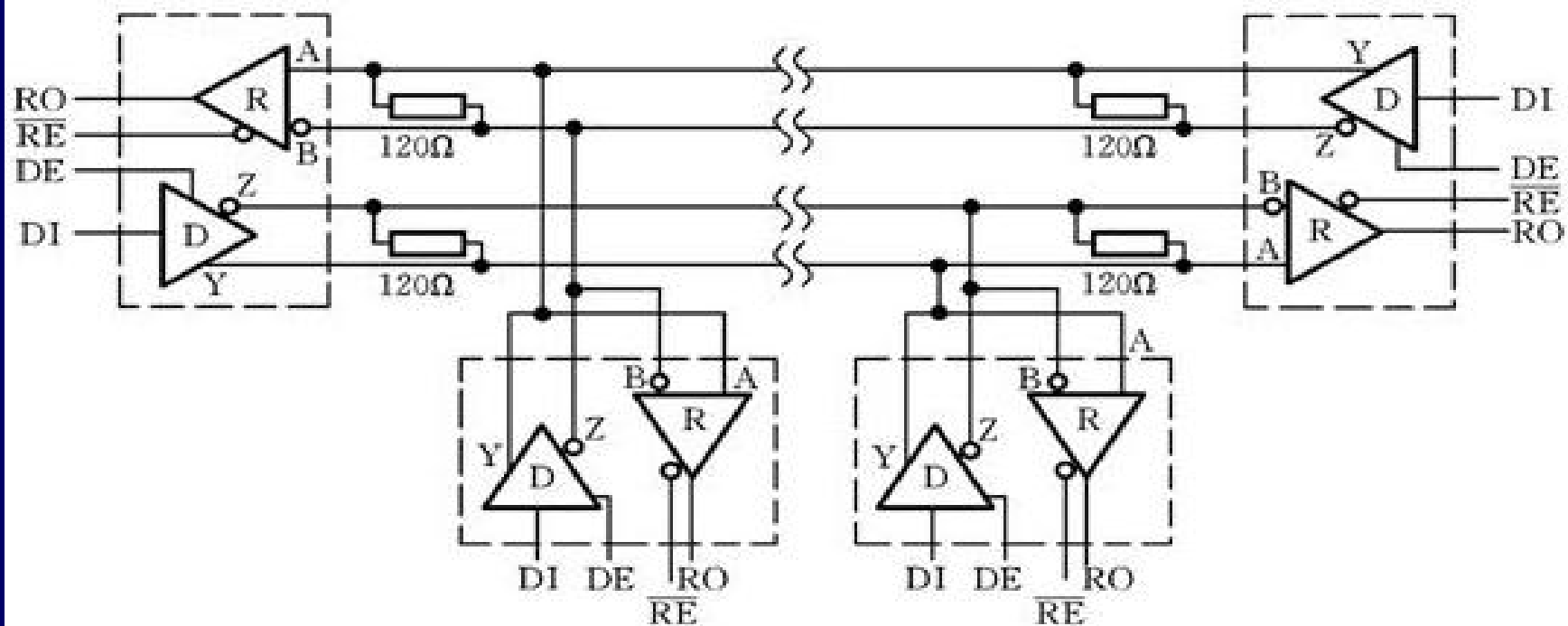
图

2. RS-485接口标准的特点

- ① 由于采用差动发送/接收和双绞线平衡传输，所以共模抑制比高、抗干扰能力强。因此，特别适合在干扰比较严重的环境下工作，如大型商场和车间使用。
- ② 传输速率高，可达10 Mb/s（传输15m），传输信号摆幅小（200mV）。
- ③ 传输距离长，不使用MODEM，采用双绞线，传输距离为1.2 km（100Kb/s）。
- ④ 能实现多点对多点通信。



(a)



(b)

串行通信接口电路的任务

1、实现数据格式化

发送器：将来自CPU的并行数据转变成异步起止式帧数据格式或同步数据块(前后加同步字符)

接收器：将接收到的数据恢复成CPU所要求的并行数据

2、进行串 \longleftrightarrow 并转换 (同上)

3、进行错误检测

在发送时接口电路对传送的字符数据自动生成奇偶校验位或其它校验码

在接收时, 接口电路检查字符的奇偶校验位或其它校验码, 以确定是否发生传送错误。



4、提供符合RS-232C接口标准所要求的信号线

远距离通信采用MODEM时需9根信号线

近距离零MODEM方式通信只需3根信号线


这些信号线由接口电路提供，以便与MODEM或终端进行联络与控制。

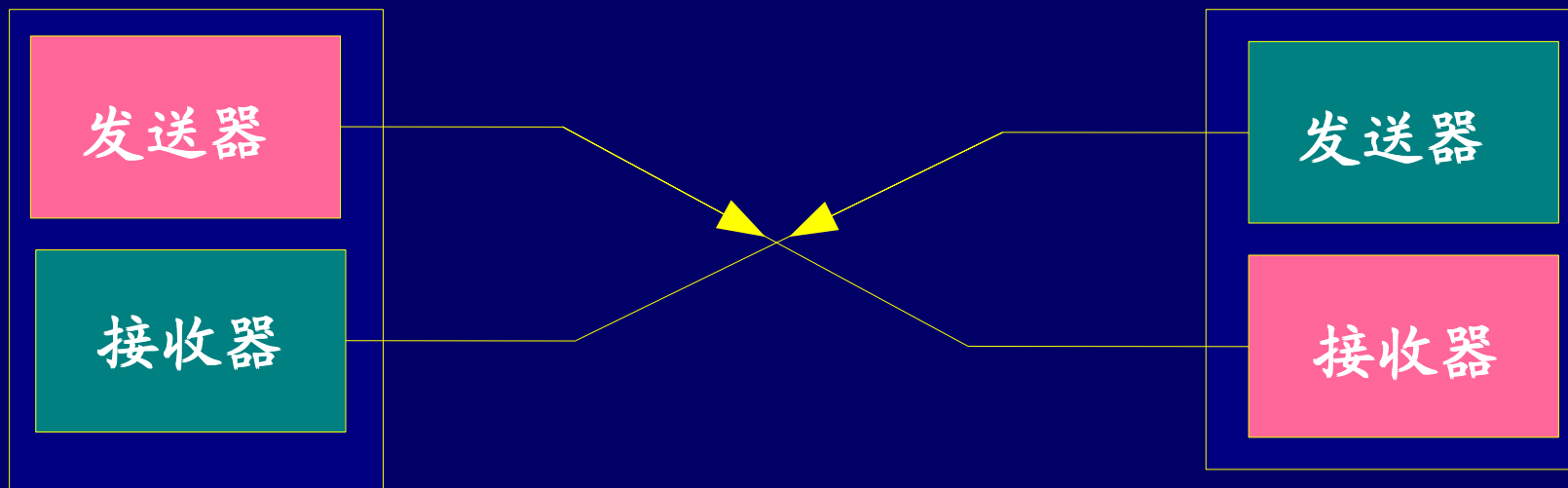
5、进行TTL与EIA电平转换

6、控制数据传输速率：

对波特率能进行选择和控制

串行接口的组成

- ① 可编程的串行接口芯片  同步USRT
异步UART
- ② 波特率发生器
- ③ EIA与TTL电平转换器
- ④ 地址译码器



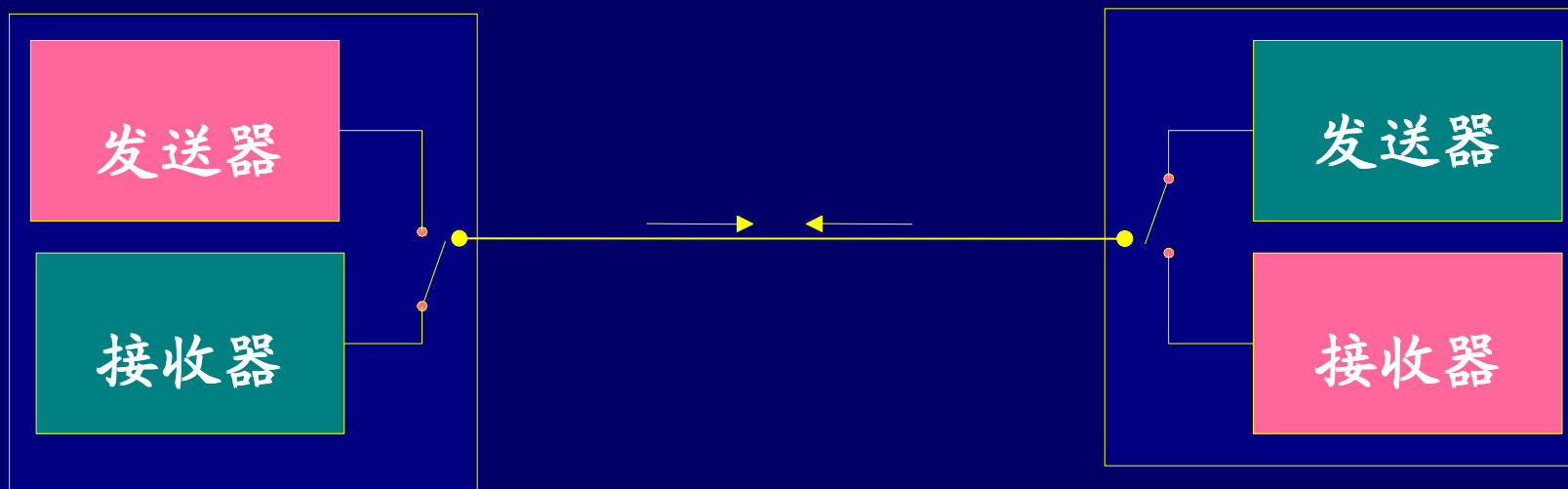
全双工

特点: ①每一端都有发送器和接收器

②有二条传送线

应用: 交互式应用, 远程监测控制

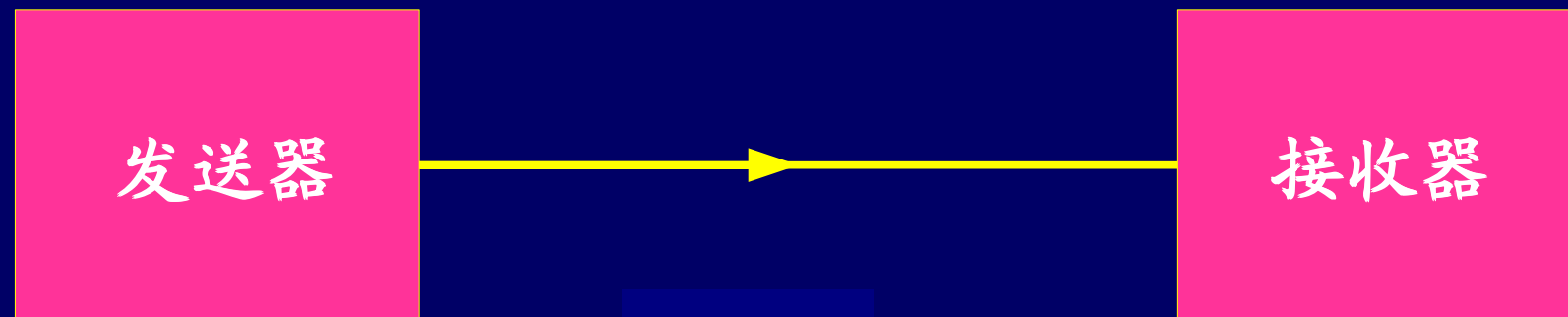
[返回](#)



半双工

特点： ①每端需有一个收/发切换电子开关
②因有切换，会产生时间延迟

应用： 打印机串口，单向传送设备，发送器→接收器



单工

[返回](#)

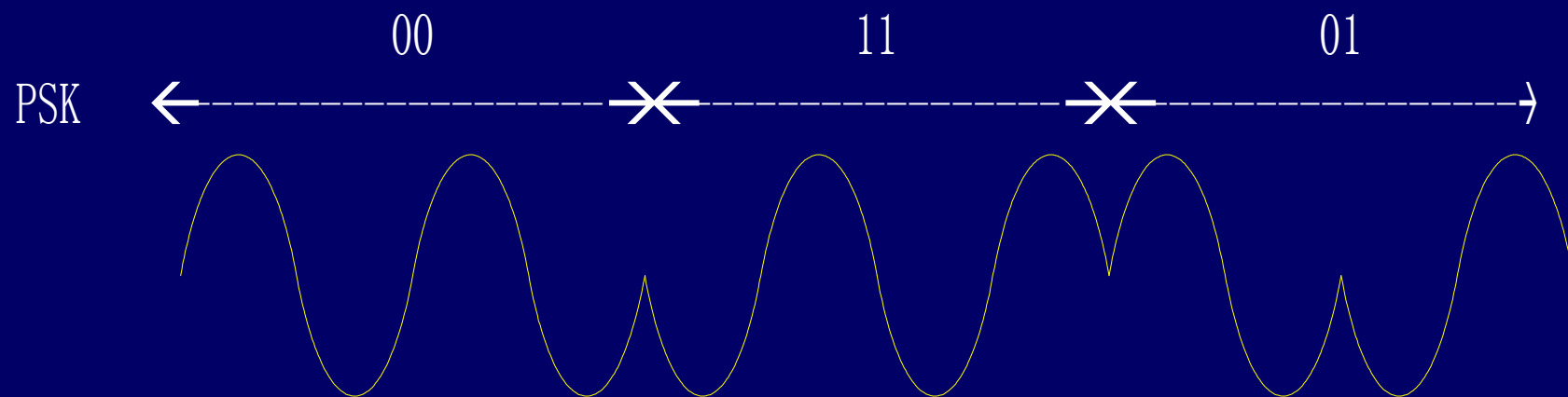
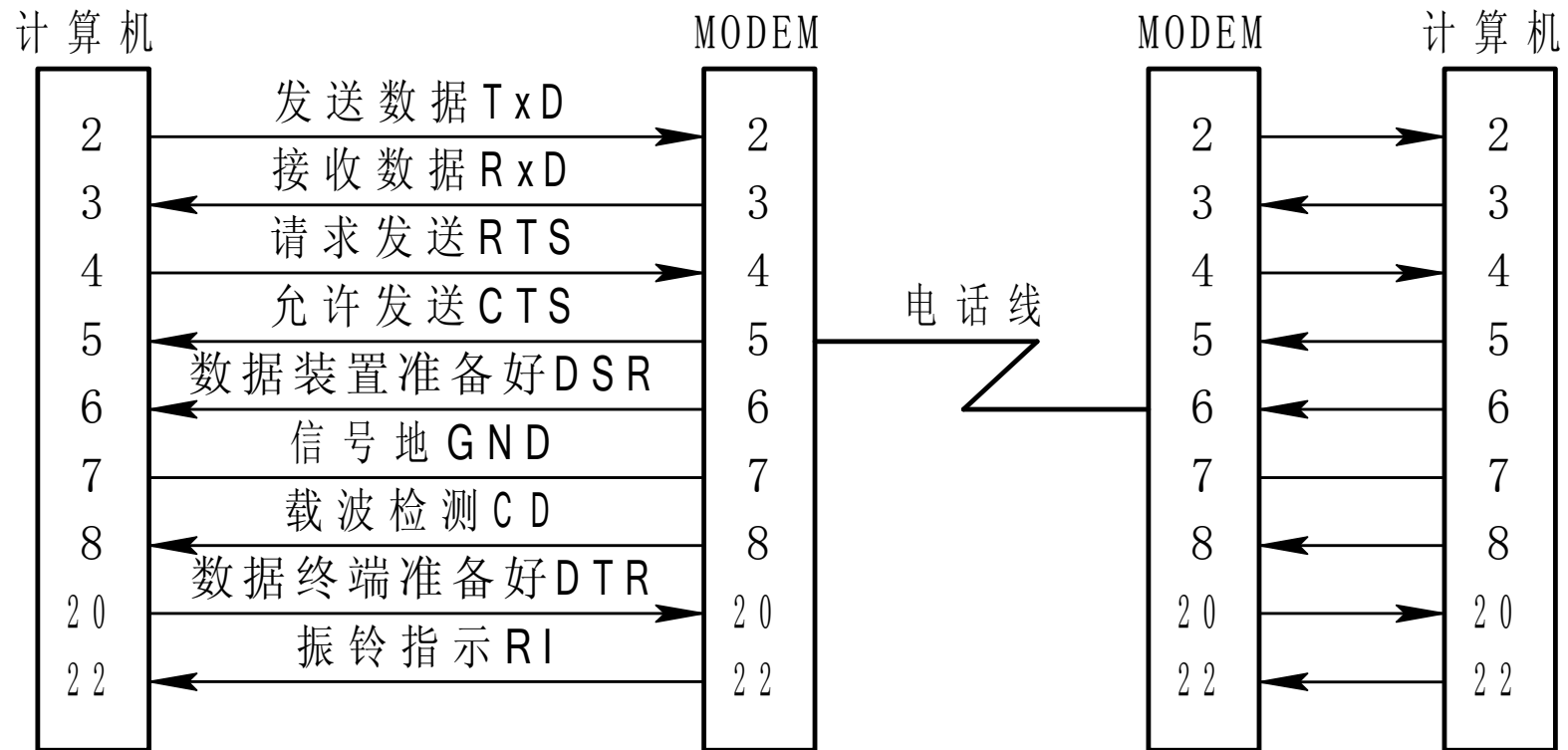


图 9-5

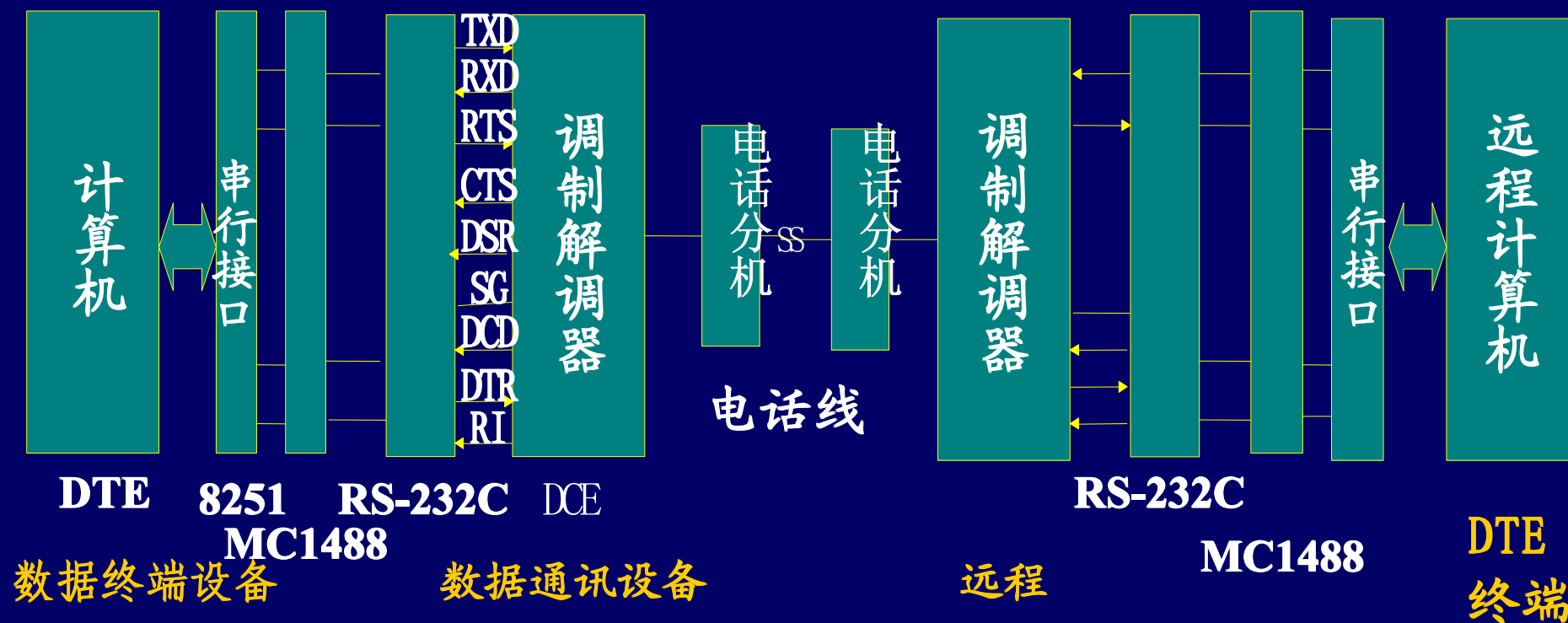


采用MODEM时RS-232C信号线的使用

图 (a)

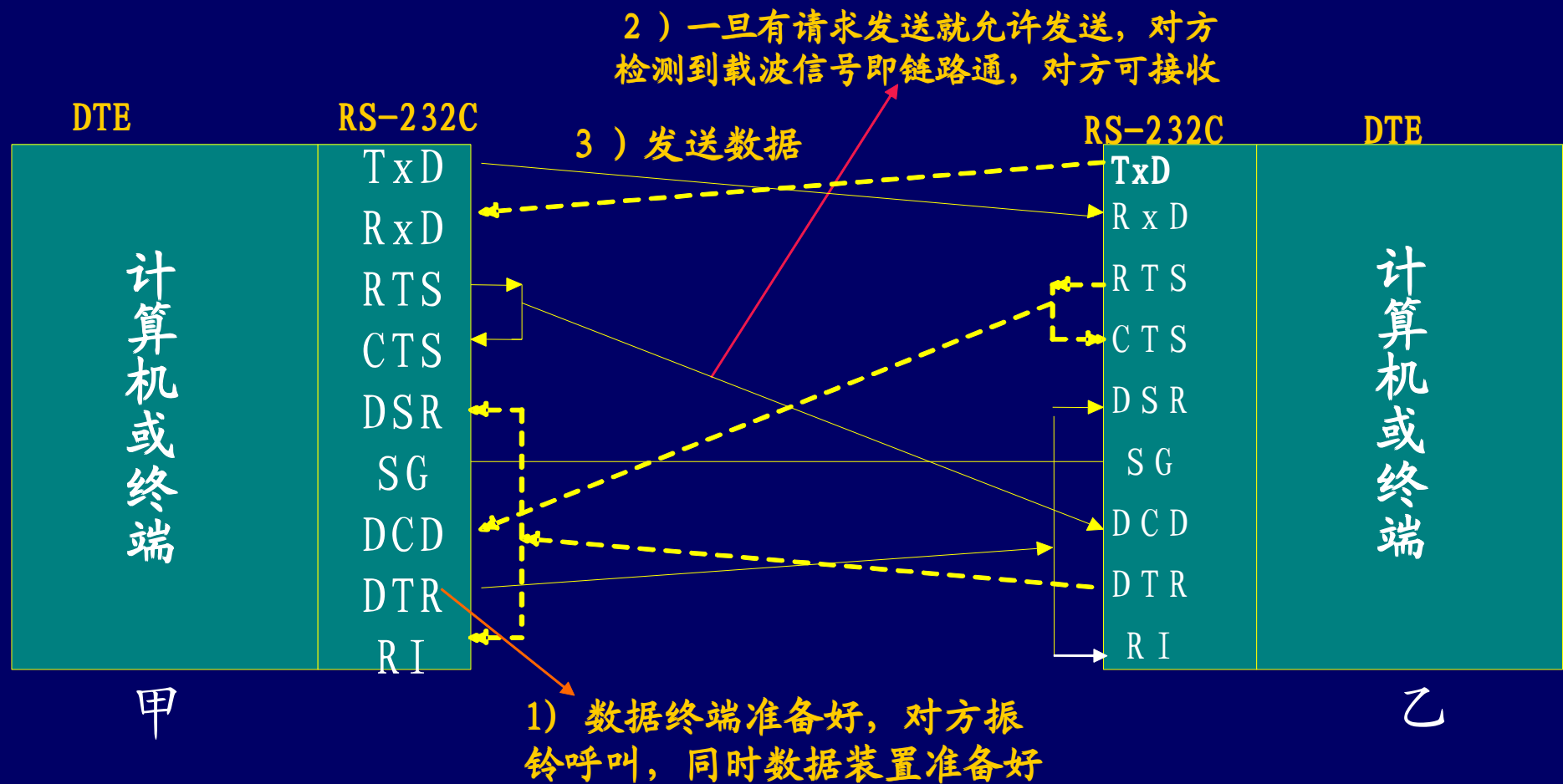


使用Modem的RS-232C接口



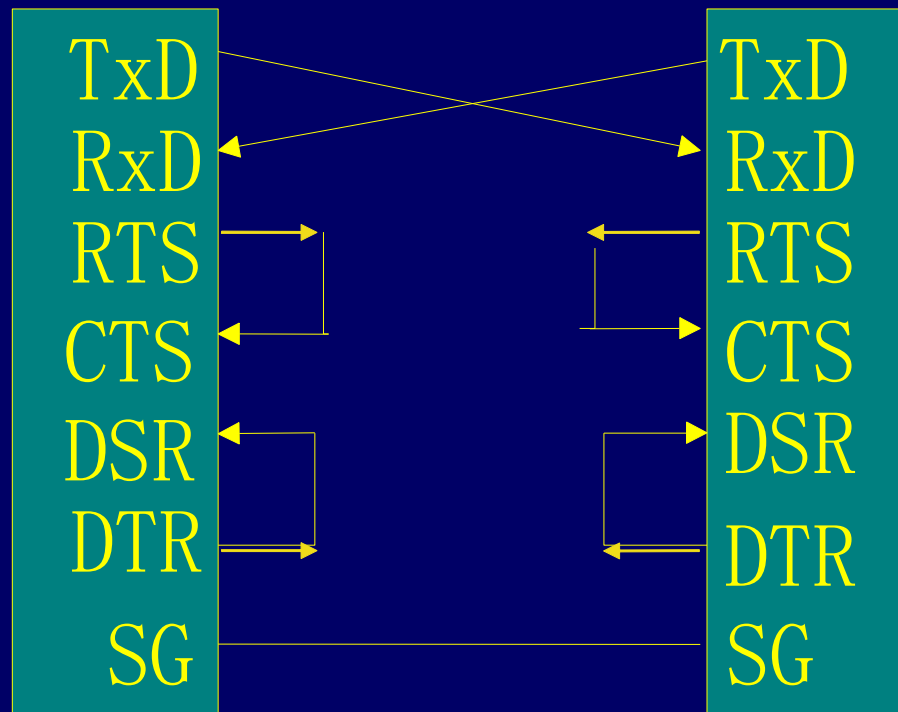
采用MODEM时RS-232C信号线的使用

图 (b)



零MODEM方式的标准信号连接

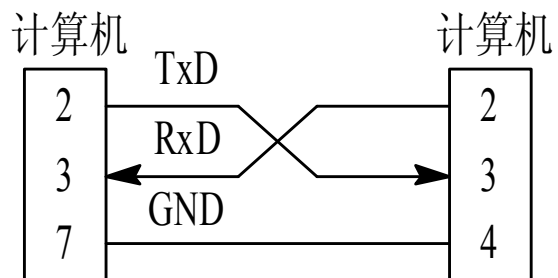
实线: 甲为DTE, 乙为DCE, 虚线: 乙为DTE, 甲为DCE



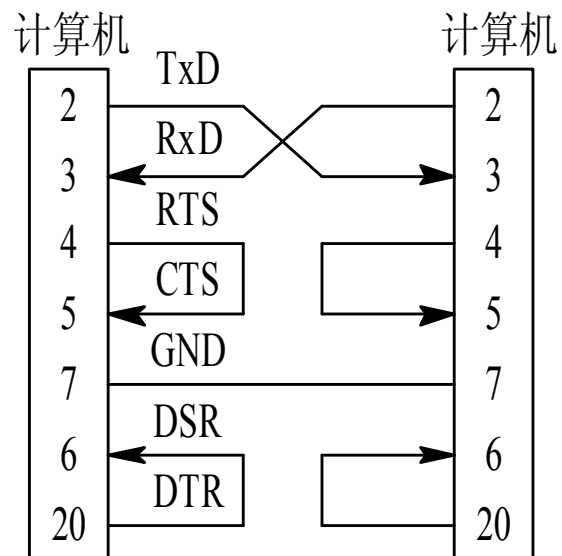
通信双方的任何一方，只要请求发送RTS有效和数据终端准备好DTR有效，就能开始发送和接收

零MODEM简单连接

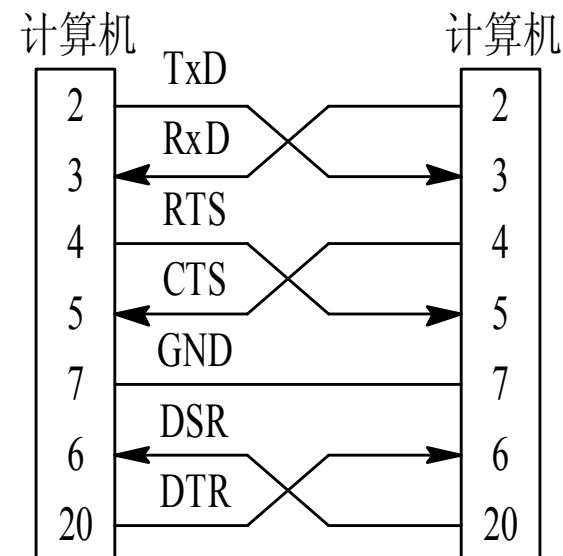
图14-8



(a)



(b)



(c)

不用Modem的RS-232C接口

在异步传输过程中，利用16倍频与波特率的接收时钟对串行数据流进行检测和定位采样，接收器能在一个位周期内采样16次

