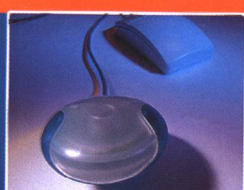
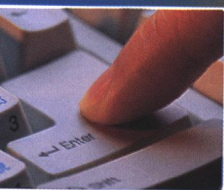


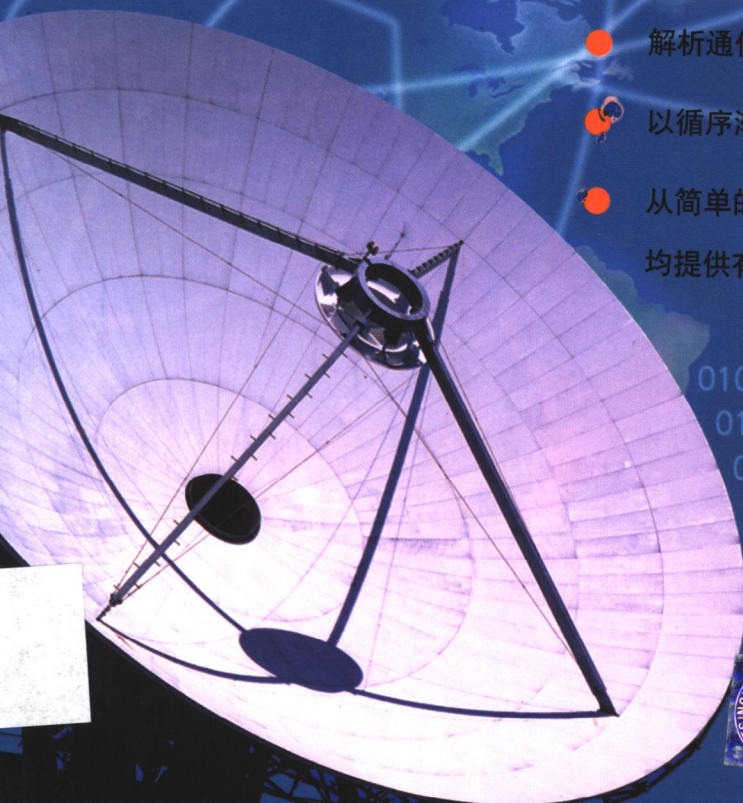
Delphi

与RS-232串行通信控制

范逸之 陈立元 编著



- 使用Delphi详细说明自动化的基础——RS-232串行通信
- 利用 Win32 API 建立通信程序
- 解析通信用 VCL 组件的开发步骤
- 以循序渐进的方式介绍串行通信理论
- 从简单的串行通信概念、实验到复杂的仪器控制，均提供有完整的 Delphi 范例



169

TH919
T-25a

Delphi 与 RS-232

串行通信控制

范逸之 陈立元 编著

本书附盘可从本馆主页 <http://lib.szu.edu.cn/>
上由“馆藏检索”该书详细信息后下载，
也可到视听部复制

清华大学出版社

(京)新登字 158 号

内 容 简 介

本书使用 Delphi 详细说明自动化的基础——RS-232 串行通信, 涉及以下内容: 将 Delphi 的应用扩展到硬件控制, 以 Delphi 开发串行通信的组件, 控制调制解调器, 进行文件传输, 通过串行端口来控制仪器, 使串行通信扩展至分布式监控系统等。本书以循序渐进的方式介绍串行通信的理论, 从简单的串行通信概念和实验到复杂的仪器控制, 并提供了完整的 Delphi 范例。

本书面向操作, 重点突出, 偏重应用。书中提供了完整的设计步骤和程序代码, 并给予详细的注释, 既适合有程序设计基础的读者使用, 也适合没有程序设计基础的读者学习。本书可供自动化控制和通信领域的开发人员和其他技术人员使用或参考。

本书繁体字版书名为《Delphi 与 RS-232 串列通讯控制》, 由文魁资讯股份有限公司出版, 版权属范逸之、陈立元所有。本书简体字中文版由文魁资讯股份有限公司授权清华大学出版社独家出版。未经本书原版出版者和本书出版者书面许可, 任何单位和个人均不得以任何形式或任何手段复制或传播本书的部分或全部内容。

北京市版权局著作权合同登记号: 图字 01-2002-1734 号

版权所有, 翻印必究。

本书封面贴有清华大学出版社激光防伪标签, 无标签者不得销售。

书 名: Delphi 与 RS-232 串行通信控制

作 者: 范逸之 陈立元 编著

责任编辑: 桑任松

出 版 者: 清华大学出版社(北京清华大学学研大厦, 邮编 100084)

<http://www.tup.tsinghua.edu.cn>

印 刷 者: 北京通州区大中印刷厂

发 行 者: 新华书店总店北京发行所

开 本: 787×1092 1/16 印张: 26.25 字数: 628 千字

版 次: 2002 年 6 月第 1 版 2002 年 6 月第 1 次印刷

书 号: ISBN 7-900641-88-2

印 数: 0001~5000

定 价: 49.00 元(含 1 张光盘)

前 言

当今世界是一个高度自动化的社会，各式各样的设备都可以和计算机进行联机，而最简单的自动化联机方式就是使用串行通信。串行通信存在的时间已经不短了，随着时代的进步，它并没有被取代，反倒是应用越来越广泛。这是因为硬件的实现比较简单，而且用户在使用上也很简便。

Delphi 给程序设计人员提供了相当好的开发环境，开发 VCL 组件也相当简单。作者将串行通信组件的开发过程写下来，让开发者学习使用 Delphi 开发串行通信的详细过程。作者根据经验，用简单易懂的编排方式，将串行通信的课题分若干个章节。有关程序设计的一部分，尽量在书中交代清楚，希望读者可以了解设计流程中的细节，同时避免按书本操作与实际操作不一致的情况发生。本书使用 Delphi 6 作为开发环境，基本也适合 Delphi 5 的用户。

本书不仅说明了串行通信的概念，也说明了在窗口环境中设计通信程序上的一些技巧与注意事项。如果读者已经有程序设计的基础，看本书应该会比较轻松；如果不清楚程序设计的细节，书中也会交代。不管如何，本书应是一本颇具实用价值的参考书。

书中的各章节内容简述如下：

第 1 章：基本概念，主要介绍串行通信的相关硬件概念。

第 2 章：Delphi 简介，介绍 Delphi 的常用组件。如果读者已了解 Delphi，可以跳过此章。

第 3 章：说明如何通过 Win32 API 建立串行通信的程序，并有相关的基础实验程序。

第 4 章：将第 3 章说明的方法封装起来，以 VCL 的方式制成组件，并详述其开发的流程及方法，方便后面的程序开发。

第 5 章：串行通信的传输有字符及字节两种方式，本章详细说明它们之间的差别并以实验进行验证。

第 6 章：讨论串行数据的处理方式，重点是文字、图形及字节的展现和处理。

第 7 章：介绍调制解调器的控制，包括拨号调制解调器及专线调制解调器的控制方式，并以 Delphi 程序做最详细的说明。

第 8 章：Delphi 可以完成大部分的串行通信事务，还有其他通信组件可以辅助 Delphi 进行更好的串行通信，将 Delphi 的串行通信功能进一步提升。本章将说明如何运用协作厂的组件提升 Delphi 的串行通信功能。

第 9 章：使用串行通信的方式可以达到仪器控制的目的，本章就数种市面上可见的仪器，经过串行接口实现仪器控制的自动化。

第 10 章：分布式的串行通信可以使串行接口功能向监控系统扩展，本章简单说明如何使用 Delphi 控制分布式系统。

由于作者才疏学浅，所书所言均是工作经验的累积整理所成，若有未臻理想之处，还请各位读者不吝指正。

范逸之(tom@hwada.com.tw)

陈立元(0930912770@fetnet.net)

2001 年 11 月于桃园

目 录

第 1 章 基本概念	1	2.2.1 Label 组件	39
1.1 通信	1	2.2.2 Button 组件	40
1.1.1 数据传送	1	2.2.3 Timer 组件	41
1.1.2 通信的种类	2	2.2.4 PaintBox 组件	42
1.2 串行通信	3	2.2.5 RadioButton 组件	43
1.2.1 RS-232 串行通信	3	2.2.6 GroupBox 组件	44
1.2.2 RS-485 串行通信	4	2.2.7 Shape 组件	44
1.2.3 USB 接口	6	2.2.8 Edit 组件	45
1.2.4 IEEE-1394	8	2.2.9 Memo 组件	46
1.3 串行通信端口	9	2.3 程序简述	46
1.3.1 信号定义	9	2.3.1 单元的内容	46
1.3.2 ASCII 码对照表	10	2.3.2 运算符	47
1.3.3 引脚意义及方向	11	2.3.3 语法	48
1.3.4 通信参数	15	2.3.4 基础的信息对话框	50
1.4 模式及流量	18	2.4 常见问题解答及本章习题	51
1.4.1 工作模式	18	2.4.1 常见问题解答	51
1.4.2 硬件握手	19	2.4.2 本章习题	51
1.4.3 软件握手	21	第 3 章 串行通信程序及组件建立	53
1.5 接线和错误预防	23	3.1 串行通信的 Windows API 简述	53
1.5.1 接线方法	23	3.1.1 串行通信相关函数	53
1.5.2 错误预防	24	3.1.2 使用流程	61
1.5.3 CRC 过程解析	25	3.1.3 检查资源设置	62
1.6 常见问题解答及本章习题	26	3.2 通信测试	64
1.6.1 常见问题解答	26	3.2.1 通信步骤	64
1.6.2 本章习题	27	3.2.2 回路测试	64
第 2 章 Delphi 及常用组件简介	28	3.2.3 串行端口的数字输出控制	71
2.1 窗口程序概念简述	28	3.2.4 串行端口的数字输入检测	79
2.1.1 对象的概念	28	3.3 自动与事件	84
2.1.2 接口成员	29	3.3.1 自动读取传入的字符串	84
2.1.3 以 Delphi 开发系统	31	3.3.2 通信事件	90
2.1.4 Delphi 的环境模式	36	3.3.3 建立多线程	99
2.1.5 开发工程步骤	37	3.4 常见问题解答及本章习题	109
2.1.6 Delphi 和操作系统的关系	38	3.4.1 常见问题解答	109
2.2 常用组件介绍	39	3.4.2 本章习题	109

第 4 章 串行通信组件的建立	110	第 6 章 串行数据的处理	178
4.1 组件的基础	110	6.1 命令字符串	178
4.1.1 组件分类	110	6.1.1 通信方式	178
4.1.2 组件组成	111	6.1.2 CheckSum 的使用	181
4.1.3 组件建立步骤	112	6.2 PSIS 仿真器简介	187
4.2 通信组件的建立	113	6.2.1 PSIS 功能简介	188
4.2.1 组件分析	113	6.2.2 PSIS 上的接头定义	189
4.2.2 属性建立	115	6.2.3 串行仪器控制的实习	190
4.2.3 事件的建立	122	6.2.4 通信参数的设置	191
4.2.4 方法的考虑	125	6.3 客户端的建立	191
4.2.5 集成组件	127	6.3.1 文本框与数据显示	192
4.3 安装及使用串行通信组件	141	6.3.2 状态信号灯与数据显示	198
4.3.1 TComm 的安装	141	6.3.3 PaintBox 与数据显示	205
4.3.2 组件测试——数据的传送 与接收	143	6.3.4 PaintBox 与数据显示—— 字节数据	214
4.3.3 组件测试——数字输入/ 输出	147	6.4 常见问题解答及本章习题	222
4.3.4 组件事件测试——自动 读取数据	152	6.4.1 常见问题解答	222
4.4 常见问题解答及本章习题	156	6.4.2 本章习题	222
4.4.1 常见问题解答	156	第 7 章 调制解调器的控制	223
4.4.2 本章习题	156	7.1 调制解调器	223
第 5 章 串行通信中的字符与字节	157	7.1.1 何谓调制解调器及其种类	223
5.1 字符与字节	157	7.1.2 指令集	226
5.1.1 字符和字节的差别	157	7.1.3 文件传输协议介绍	231
5.1.2 Delphi 中的字符串种类及 处理函数	158	7.2 Delphi 与调制解调器	232
5.1.3 中英文字符串长度计算	160	7.2.1 Delphi 与调制解调器的 通信	232
5.1.4 字符编码	164	7.2.2 调制解调器的相连	238
5.2 字节数据的传送与接收	168	7.2.3 调制解调器与专线	244
5.2.1 字节类型、声明与传送/ 接收	168	7.2.4 专线模式设置与控制	245
5.2.2 动态数组	173	7.3 常见问题解答及本章习题	252
5.3 常见问题解答及本章习题	177	7.3.1 常见问题解答	252
5.3.1 常见问题解答	177	7.3.2 本章习题	252
5.3.2 本章习题	177	第 8 章 其他串行通信组件	253
		8.1 Windows 的超级终端程序	253
		8.1.1 选择与使用超级终端程序	253
		8.1.2 与设备的连接测试	256
		8.2 PComm Pro 软件	257

8.2.1 PComm Pro 的超级终端.....	258	9.4.4 控制程序的建立.....	346
8.2.2 PComm Pro 的串行端口 性能测试	260	9.5 温度记录仪.....	353
8.2.3 PComm Pro 的数据监视器.....	261	9.5.1 温度记录仪简介.....	353
8.3 PComm Pro 通信链接库	264	9.5.2 接口及命令格式.....	354
8.3.1 函数库的使用及结构	264	9.5.3 通信程序的建立.....	358
8.3.2 数字输入/输出控制	266	9.6 转速计.....	365
8.3.3 使用 DLL 进行串行数据的 传送	279	9.6.1 转速计介绍	365
8.3.4 事件的触发	284	9.6.2 接口及命令格式.....	366
8.3.5 文件传输	293	9.6.3 工程程序的建立.....	368
8.4 常见问题解答及本章习题.....	304	9.7 条形码扫描器.....	373
8.4.1 常见问题解答	304	9.7.1 条形码扫描设备介绍.....	373
8.4.2 本章习题	304	9.7.2 接口及格式说明.....	375
第 9 章 仪器设备上的串行通信.....	305	9.7.3 工程程序的建立.....	375
9.1 噪音计	305	9.8 测量电表.....	378
9.1.1 仪器联机	305	9.8.1 Source Meter 设备介绍	378
9.1.2 噪音计简介	306	9.8.2 接口及命令格式.....	379
9.1.3 命令格式	307	9.8.3 工程程序的建立.....	381
9.1.4 通信程序的设计	309	9.9 常见问题解答及本章习题	387
9.1.5 噪音计读数的抽取	314	9.9.1 常见问题解答.....	387
9.2 电功率计	318	9.9.2 本章习题	388
9.2.1 电功率计简介	319	第 10 章 串行通信的分布式监控.....	389
9.2.2 接口及命令格式讨论	320	10.1 分布式.....	389
9.2.3 测试程序的建立	323	10.1.1 何谓分布式监控.....	389
9.3 测量用放大器	332	10.1.2 多模块的网络系统.....	390
9.3.1 BK-2525 振动计简介.....	333	10.1.3 RS-232 与 RS-485 的转换 ...	392
9.3.2 RS-232 接口及命令格式 说明	334	10.2 指令与格式.....	393
9.3.3 RS-232 命令说明	335	10.2.1 格式讨论	393
9.3.4 程序的建立	337	10.2.2 发送和接收流程.....	396
9.4 电源设备	342	10.2.3 取得模块的配置.....	396
9.4.1 电源设备简介	342	10.3 常见问题解答及本章习题	400
9.4.2 接口及命令格式	344	10.3.1 常见问题解答.....	400
9.4.3 RS-232 命令说明	345	10.3.2 本章习题	401
		附录 有关补充资料.....	402

第 1 章 基本概念

万丈高楼平地起，学习任何一项技术总要对该技术的基础知识有一些了解，这样才能够知其然并知其所以然，也才能够真正掌握此项技术。本章将说明串行通信的一些基本概念。

1.1 通 信

不同的独立系统通过线路互相交换数据，便是通信；而构成整个通信的线路则称为网络。交换信息的系统若为计算机系统，则称为计算机网络通信。通信的目的是进行数据交换，因为数据必须经过交换才能由一端传送到另一端。发送端所使用的方法就是将数据通过一定的程序与线路发送出去，接收端则按照协商好的方式将数据收集起来并保存或显示在画面上。

1.1.1 数据传送

通信的主要目的是将数据从一端传送到另一端，实现数据的交换。人与人之间的对话，计算机与设备间的数据交换，计算机与计算机间的数据传送，乃至广播或卫星信号的传送，都属于通信的范畴。

一个完整的通信系统包括发送端、接收端、转换数据的接口及传送数据的实际信道或媒体。一般说来，发送与接收的节点称为数据终端设备(Data Terminal Equipment, DTE)，如个人计算机、工作站或计算机系统。数据在到达正确的目的地之前，可能需要经过一系列中间节点，这些中间节点负责数据的转送工作，以送达目的地。这些中间节点称为数据交换设备(Data Switching Equipment, DSE)，如电信局的交换机等。终端设备发送数据时，必须先将数据转换为电气信号，以便在线路上传递，而负责数据与电气信号转换的设备称为数据通信设备(Data Communication Equipment, DCE)，如调制解调器。因此，计算机等 DTE 设备要通信时，须先经过调制解调器将其转换为电气信号，才能送上电话线路；同样，信号由外界进入计算机时，也是先经过调制解调器将外界的电气信号进行转换，数据才会进入计算机。DTE 与 DCE 间的数据传输线路通常使用 RS-232，而 DCE 与 DSE 间的媒体则包括了双绞线、同轴电缆、光纤或无线电等。

通过上述的机制，设备与设备间、计算机与计算机间或是计算机与设备间便可以互相传输数据。又如图 1.1 所示，不仅可以通过线路的联系进行数据的传输，也可将两部设备进行直接连接，不需经过调制解调器及交换设备的服务，就可进行数据的交换。这种方式在传输距离较短时比较适用。

随着个人计算机的迅速发展，个人计算机成为信息传输的良好工具。数台计算机可以连接成一个速度很快的网络系统，互相交换数据。传输设备与媒体的快速发展，成本的大幅下降，也使距离越来越不是问题；因此以计算机为基础发展出相当多的通信方法。现在，

以一台计算机遨游世界是多么简单。这都是因为科技的进步与通信协议的改进，提供了一个通道，让计算机可以通过调制解调器或其他媒体与其他设备通信，从而造就了如今高速信息流的时代。

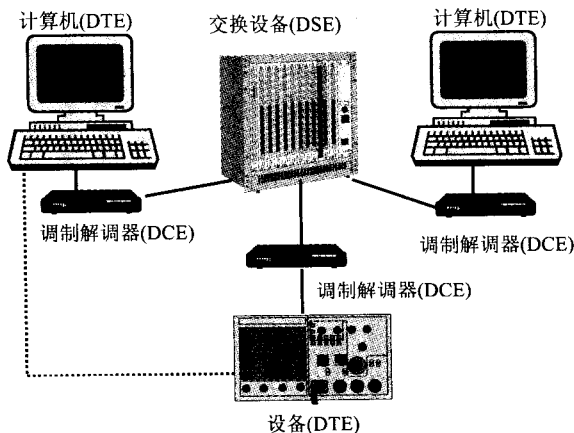


图 1.1 设备的连接

现在是一个自动化的时代，所见所闻的设备均可与计算机相连接，而借以传输数据；通过计算机的作用，自动化在人们的生产生活中将会得到更加广泛的应用。

1.1.2 通信的种类

通信的方式通常可以分为两种：并行通信(Parallel Communication)和串行通信(Serial Communication)。这两种不同的通信模式如图 1.2 所示。

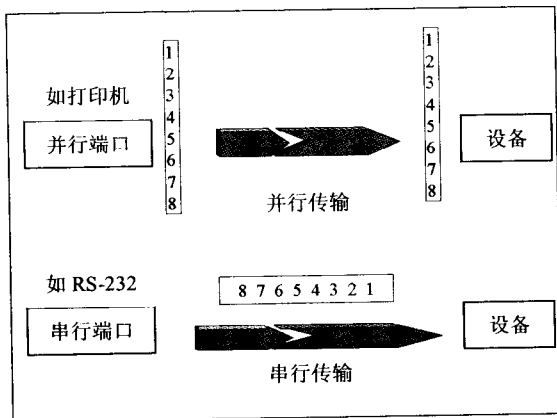


图 1.2 两种不同的传输方式

由图 1.2 中可以了解，所谓的并行通信，即一次的传输量为 8 个位(1 个字节)；而串行通信则是一次只传输 1 个位(也就是一个标准电位的状态)。两者之间的数据传输速度相差了 7 倍之多，但并不是说串行通信就不好，它之所以存在这么久的时间，自然有它的长处。并行通信虽然可以在一次的数据传输中就传输 8 个位，但是因为数据电压传输的过程中，

容易因线路的因素而使标准电位发生变化(最常见的是电压衰减问题,以及信号间互相干扰问题(CrossTalk)),因而使传输的数据发生错误。如果传输线比较长的话,电压衰减效应及互相干扰问题会更加明显,数据的错误也就会比较容易发生。相比之下,串行通信一次只传1个位,相对来说,处理的数据电压只有一个标准电位,因此不太容易把数据漏失,再加上一些防范措施,要漏失就更不容易了。

由最早的串行通信发展到现在,由于不同情况的需求,串行通信的样式也越来越多;而并行通信的发展其实也没有间断过。在传输距离较短的应用场合中,使用并行通信的高传输率特性可以让数据的传输更快;在部分科学仪器、医疗诊断仪器由于需要传输的数据量都相当大,因此使用并行传输接口(如 GPIB、LPT 等)就经常见到。

1.2 串行通信

常用的串行通信有两种,本节将分别对其进行说明,并概括介绍近年来相当盛行的 USB 及 IEEE-1394,这两种也属于串行通信的扩展。

1.2.1 RS-232 串行通信

串行通信端口(Serial Communication Port)在系统控制的范畴中一直占有极重要的地位,不仅没有因为时代的进步而被淘汰,反而是在规格上越来越向其极限挑战。现在,计算机上的串行通信端口(RS-232)是标准配置,用途上则以连接调制解调器进行通信传输(尤其是因特网成为时尚后,接上 WWW 取得数据是相当重要的一个收集信息方式)最为常见,当然它的重要性还不只这些!在本书中还会对通信端口的应用进行深入介绍。

RS-232 的通信端口是每台计算机上的必要配置,通常含有 COM1 与 COM2 两个端口。一般的计算机将 COM1 以 9 引脚的接头接出,而以 25 引脚的接头将 COM2 接出。新一代的计算机均以 9 引脚的接头接出所有的 RS-232 通信端口。在计算机上的 RS-232 均是公头,即使是 25 引脚也是公头,千万不要与其他的设备弄混淆了(打印机连接端口也是 25 引脚,不过它是母头,请仔细分别)。通常与计算机连接的设备,最简单的通信接口就是 RS-232,不仅实现简单,而且价格上也便宜很多;在市面可见的数码相机、调制解调器等很多都提供 RS-232 作为与计算机通信的接口。

仔细检查计算机后面的接线部分,大大小小的接头一堆,有两个 9 引脚的接头区(这是以较新的计算机来说,而以前的计算机可能有 25 引脚的型号),这就是串行通信端口。在计算机背后的各接头中可以看到如图 1.3 所示的接头,这就是 RS-232 的标准接头。

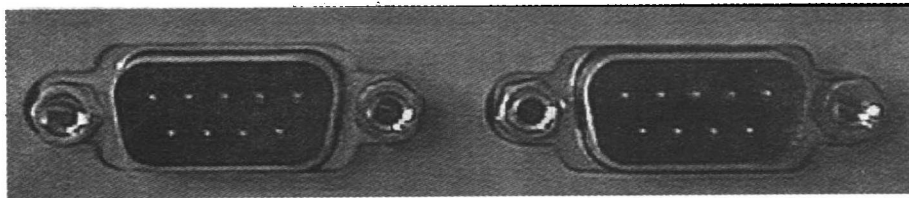


图 1.3 个人计算机上 RS-232 的外观

工业应用的场合中, 工业计算机的使用比一般计算机来得普遍, 在工业计算机上的接头位置很可能如图 1.4 所示。

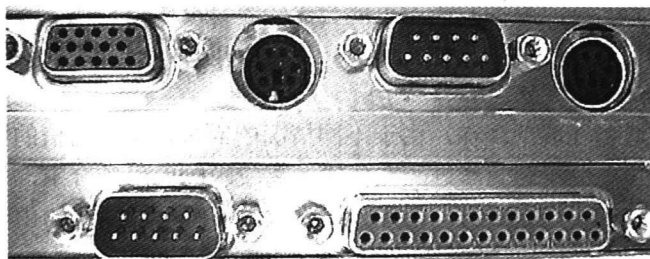


图 1.4 工业计算机中的 COM 端口及其他接头

其中 9 引脚的端口有两个, 分别位于上一排的从右边数第二个及下排的左边。不管工业级或是一般的台式计算机, 只要是 RS-232 就是公头 9 引脚, 这一点是相同的。

计算机上的串行通信端口一定是公头, 这点是不会改变的。如果你所使用的计算机比较旧, 上面还有 25 引脚的接头, 肯定也是公头; 这与打印机的母头接口是不一样的。如果从操作系统中进行查看, 则可以选择【开始】|【设置】|【控制面板】命令, 在【控制面板】中双击【系统】图标, 在打开的【系统属性】对话框中找到计算机的连接端口, 其过程如图 1.5 所示。

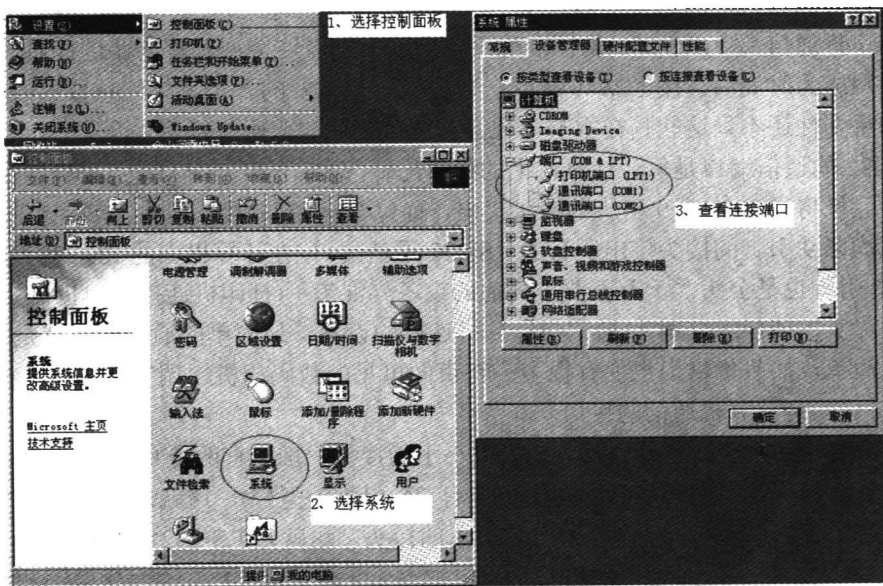


图 1.5 查看计算机的连接端口

只要显示通信连接端口的名称, 且其左边没有黄色感叹号的话, 就表示这个 COM 端口是可以使用的。

1.2.2 RS-485 串行通信

由于串行通信的简单易用, 使工业领域大量使用串行通信来进行数据传输; 可是工业

环境通常会有噪声干扰传输线路,在以 RS-232 进行传输时就经常会受到外界的电气干扰而使得信号发生错误。为了解决这个问题,RS-485 的通信方式就应运而生了。

为了了解信号的干扰情况,首先来看看如图 1.6 所示的 RS-232 的信号传输模式。

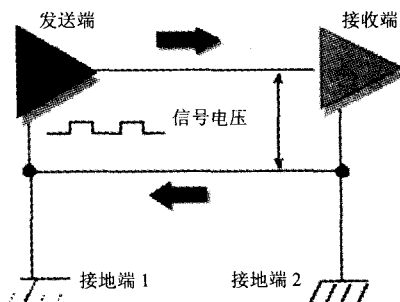


图 1.6 RS-232 的信号传输方式

由图 1.6 可知,RS-232 的信号标准电位乃是参考地线而来的。发送端参考接地端 1 来发送数据;接收端则参考接地端 2 来还原发送端信号的标准电位;在两个接地端同电压的前提下,发送端与接收端的信号标准电位是相同的。但若传输数据的过程中有噪声进到传输线路中,可能会如图 1.7 所示。

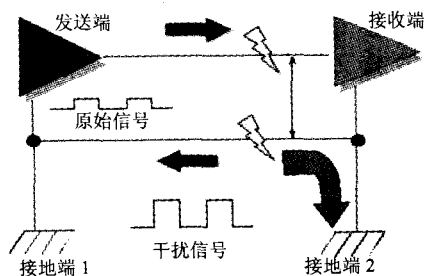


图 1.7 RS-232 信号受干扰时的情况

干扰的噪声信号在地线及信号上均会产生影响,原始信号在加上噪声信号后依然是传输到接收端;而地线部分的信号则被地电压给平均掉了,因此,信号便发生了扭曲,当然信号整个都不对了。

而 RS-485 的信号传输方式如图 1.8 所示。

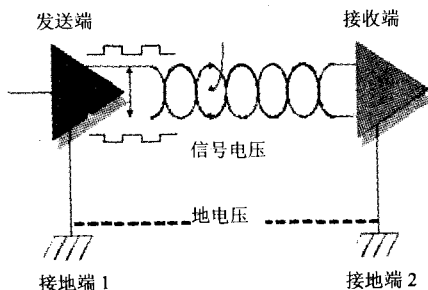


图 1.8 RS-485 的信号传输方式

RS-485 的信号将被发送出去时会先分成正负的两条线路，当到达接收端后，再将信号相减还原成原来的信号；如果将原始的信号记为 (D_T) ，而被分成后的信号分别记为 (D_+) 及 (D_-) ，则原始的信号与离散的信号在由发送端发送出去时的运算关系就如下：

$$(D_T) = (D_+) - (D_-)$$

同样，接收端在接收到信号后，也按上式的关系将信号还原成原来的样子。而如果此线路受到干扰，其情况可能如图 1.9 所示。

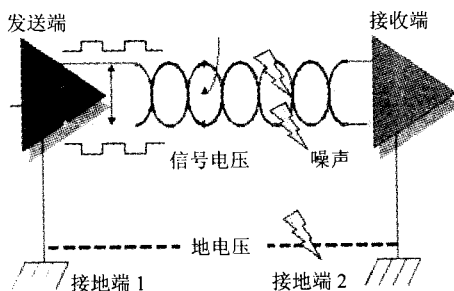


图 1.9 RS-485 传输受干扰时的情况

这时候在两条传输线上的信号会分别成为 $(D_+) + \text{Noise}$ 及 $(D_-) + \text{Noise}$ 。如果接收端接收此信号，它必须按照一定的方式将其合成，因此合成的运算式如下：

$$\begin{aligned} (D_T) &= [(D_+) + \text{Noise}] - [(D_-) + \text{Noise}] \\ &= (D_+) - (D_-) \end{aligned}$$

此式与前一式的结果是一样的。

所以，使用 RS-485 网络可以有效地防止噪声信号的干扰。也因为此种特性，工业上的应用比较适合此种串行传输的方式。

注意： RS-485 虽可防止噪声，但也并非无懈可击。若噪声信号的电压很大(如闪电)，一样会造成错误。

1.2.3 USB 接口

由于时代的进步，串行通信也朝向高速化发展。近年来个人计算机的快速发展下，使用串行通信技术发展出了通用串行总线(Universal Serial Bus, USB)接口，此种规格的目的在于集成一般计算机所使用的外围设备的连接方式，而且其所采用的信号传输方式也是串行通信(一次只传送一个位)。

一般的计算机，其与外围设备的联机方式如图 1.10 所示。

每一种设备都使用一条专用的连接线与计算机后面的接头相连接，而计算机的背后则都为不同的设备事先安装了可以接到主板上的接头。由这种连接方法可以发现，计算机的背后一定是一大堆难以分辨的线路。每一次更换一个设备都得将计算机的背后给翻过来，辛苦地整理线路，并在安装好新设备后，再将计算机位置还原。这绝对不是一件经常可以

做的事情,因为光是整理线路就会浪费掉不少的时间。

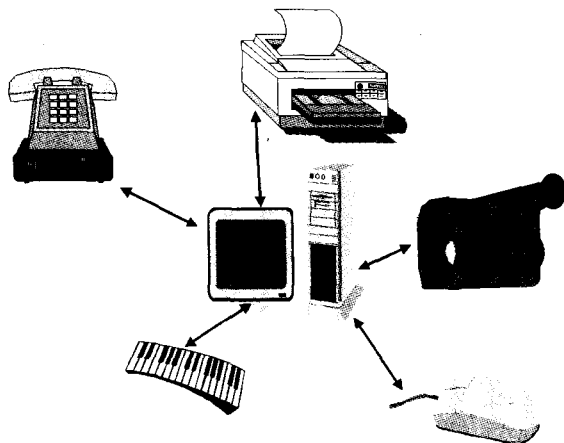


图 1.10 传统设备与计算机间的连接

而如果可以采用 USB 接口的连接方式,则这些设备的接线方式将改为如图 1.11 所示的连接方式。

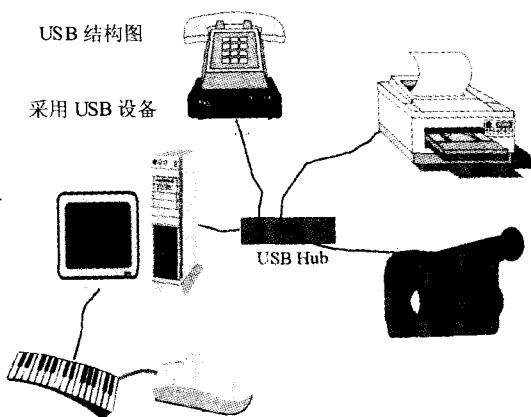


图 1.11 USB 设备和计算机间的连接

相同的外围设备若采用 USB 接口,则连接时只需用串接的方式将各个设备连接在一起即可;而如果设备比较多,也可以采用 USB 集线器进行集中管理(这与网络的集线器管理方式很相似,其原理是一致的)。设备也可以有 Windows 操作系统支持的“即插即用”功能,各个设备与集线器间的最大距离为 5 米,最多可以连接 127 个外围设备。

另一方面,使用串行通信时,速度也是一个考虑的重点。USB 集线器可以将使用 USB 接口的各个计算机外围设备连接起来,设备一多,如果众多的设备均在同时传输数据,速度必定会成为 USB 线路的一个负担。不过 USB 采用的速度高达 12 Mbps(每秒钟传输 1 200 万个位),每个单一设备最高传输带宽高达 6 Mbps,比起一般 RS-232 的最快 115 Kbps,USB 在传输上是比较快的。其主要的特性如表 1.1 所示。

USB 一经发布,在最初两年,没有引起很大的反响,各种支持 USB 接口的设备也没有流行起来。其原因之一是当时的 Windows 操作系统不支持 USB 接口。Windows 98 以后

的操作系统支持 USB 接口, 使用 USB 产品在安装上更容易, 诸如鼠标、键盘、电话、调制解调器、打印机、麦克风等, 只要支持 USB 均可得到操作系统的支持, 使得安装与使用都非常方便。现在, 已经有非常多的设备都使用 USB 作为和计算机间的连接接口了。

表 1.1 USB 特性表

特 性	描 述
低价位	USB 提供低价位连接的外围设备方案
热连接	USB 自动检测设备连接, 软件会自动对其配置, 以供立刻使用。此过程中, 不需要用户参与
单一的连接头方式	使用单种类的连接头连接所有的 USB 装置。可通过 USB 集线器增加额外的连接头
连接数多	最高可连接 127 个低速或高速的装置
线上供电	连接的缆线可提供 5 V 的直流电源, 而不同的集线器可提供的电流范围在 100 mA 至 500 mA 之间
不占系统资源	USB 装置不占系统的地址空间及 IRQ 线
错误检测与复原	USB 数据交换时包含了错误的检测; 当错误发生时, 该笔数据将被重传
节省能源	若是经过了 3 毫秒总线仍无动作, USB 装置将会自动进入省电状态; 此时装置所消耗的电流不超过 500 μ A
支持 4 种传输	USB 定义了巨量、实时、中断及控制等 4 种传输模式, 可依不同的情况启用不同的传输模式

在 1999 年所发表的 USB 2.0 已经将其速度提升到 480 Mbps, 这将使得更多的设备可以使用 USB 作为数据传输的接口。

1.2.4 IEEE-1394

IEEE-1394(也称为 FireWire, 火线)与 USB 一样拥有即插即用功能, 也是用于解决计算机与外围设备复杂的连接问题, 同样具有容易使用与便利等优点, 并且也是使用串行通信的传输方式。但不一样的是, IEEE-1394 在高速传输上占尽了优势, 现在已达到的传输速度是 400 Mbps(是 USB 1.0 版的 33 倍以上), 而且速度还向 1 000 Mbps 迈进。另外 IEEE-1394 最多可串接 63 个外围设备, 还可提供异步或同步两种传输模式。因此在不久的将来, 将会发现: 硬盘、打印机、扫描仪、VCD、CD-ROM、数码相机、调制解调器、音响系统、家庭电影院、电缆调制解调器(Cable Modem)、电视等, 都可以使用 IEEE-1394 连接起来, 而除了可进行信息技术、消费性电子产品的图像及声音传输之外, 未来更可以应用在卫星传输、通信数据、工厂自动化、太空探测等用途上。IEEE-1394 应用领域之广泛, 将使得家庭数字化及网络化的理想更易实现。

既然 IEEE-1394 这么强势, USB 是否就注定了没有存在的价值了呢? 其实不然。两者应该是合作的关系, USB 主要应用于中低速的产品; 而 IEEE-1394 则主要应用于高速传输的场合, 两者的结合才能使串行传输的功能得到最佳的支持。表 1.2 是 USB 与 IEEE-1394 这两种高速传输接口的比较。

表 1.2 USB 和 1394 的比较

比较项目	USB	IEEE-1394
应用	低速设备	高速设备
带宽(Mbps)	1.5, 12	100, 200, 400
电缆长度	5 米	4.5 米
电缆	4 线	6 线
即插即用	支持	支持

1.3 串行通信端口

本节概括介绍串行通信端口的相关术语，包括引脚定义、信号、参数等。

1.3.1 信号定义

计算机并无法分辨所谓的数字或文字，在其内部的记录中，所有的数据都是 0 与 1，任何的数据都是这两个数字的组合。想想看，现在的科技如此发达，竟然都是 0 与 1 的组合!?

所有电子仪器的基础均来自于“开”、“关”两个状态的改变，可以将它们表示为 0、1，或是“高电位”、“低电位”；不管表示方法如何，目的在于造成“状态的改变”。将很多的 0 与 1 组合成一大串的数字序列后，就可以定义它们所代表的意义了。

既然是高低电位可以形成 0、1，因此由 0、1 的无数组合可以有无穷的意义。但是何谓高电位？何谓低电位呢？任何的电子零件或芯片，其高低电位的定义均不相同，可是相同性质的电子零件会有一些的规定，大家遵守此规定，所定义出来的芯片高低电位表现就会一致了。当然，RS-232 也会有相关的电气标准电位定义，是大家共同遵循的。

ITU(国际电信联盟)的咨询机构 CCITT(国际电报电话咨询委员会)公布了规格 RS-232C，之后由 EIA(美国电子工业协会)制定；很多国家都依此为模板，而日本亦依此制定了 JIS X5101 “数据回路终端装置和数据终端装置的接口”。在此标准里，其 0、1 的规定如图 1.12 所示。

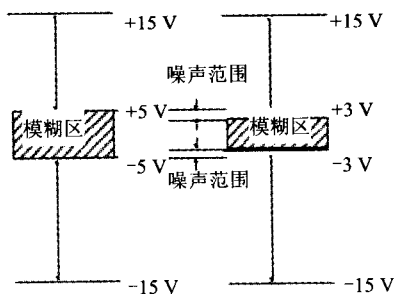


图 1.12 RS-232 标准电位图