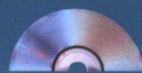




程序员书库

李长林 高洁 编著

Visual C++ 串口通信技术与典型实例



清华大学出版社

程序员书库

Visual C++串口通信技术 与典型实例

李长林 高洁 编著

清华大学出版社

北 京

内 容 简 介

本书详细介绍了利用 Visual C++进行串口通信编程的方法,并通过一系列应用实例,详细阐明了串口通信技术与 Visual C++其他技术相结合解决实际问题的基本思路和方法,同时也介绍了 Visual C++与网络开发的基本知识,以使读者真正掌握这一技术。

本书强调实用性,对于利用 Visual C++进行通信项目开发的技术人员具有较高的参考价值,同时也适合于中高级程序员、软件开发人员使用。

版权所有,翻印必究。举报电话:010-62782989 13501256678 13801310933

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

本书防伪标签采用特殊防伪技术,用户可通过在图案表面涂抹清水,图案消失,水干后图案复现;或将表面膜揭下,放在白纸上用彩笔涂抹,图案在白纸上再现的方法识别真伪。

图书在版编目(CIP)数据

Visual C++串口通信技术与典型实例/李长林 高洁 编著. —北京:清华大学出版社, 2006.5
(程序员书库)

ISBN 7-302-12903-7

I. V… II. ①李… ②高… III. C 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2006)第 037460 号

出 版 者:清华大学出版社 地 址:北京清华大学学研大厦
<http://www.tup.com.cn> 邮 编:100084
社 总 机:010-62770175 客 户 服 务:010-62776969

组稿编辑:胡伟卷

文稿编辑:刘金喜

封面设计:王 永

版式设计:康 博

印 装 者:北京鑫海金澳胶印有限公司

发 行 者:新华书店总店北京发行所

开 本:185×260 印张:23.5 字数:543 千字

版 次:2006 年 5 月第 1 版 2006 年 5 月第 1 次印刷

书 号:ISBN 7-302-12903-7/TP·8198

印 数:1~5000

定 价:39.00 元(含光盘)

前 言

串口通信作为一种非常基础而又灵活的通信方式，被广泛地应用于 PC 间的通信以及 PC 和单片机之间的通信中。提到串口通信编程，人们往往立刻想到 C、汇编等对系统底层操作支持较好的编程语言以及大串繁琐的代码。实际上，借助相关 ActiveX 控件的帮助，Visual C++ 一样能够实现串口通信，甚至其实现方法和 C、汇编相比，更加快捷方便。本书介绍了在 Visual C++ 中实现串口通信的几种典型的方法。

在 Visual C++ 中有一个 Microsoft Communication Control(简称 MSComm)的通信控件。只要通过对此控件的属性和事件进行相应编程操作，就可以轻松地实现串口通信。

本书以串口通信技术的实际应用为基础，向读者介绍了将串口技术合理地实施到实际应用中的各种方法和实例。通过这些实例，读者可以掌握串口通信技术的具体应用方法。

本书主要内容如下：第 1 章 串行口通信概述；第 2 章 利用 MSComm 控件编程；第 3 章 Windows API 编程；第 4 章 云台控制系统；第 5 章 对 Modem 的控制；第 6 章 TAPI 编程；第 7 章 串口编程调试精灵；第 8 章 VC 单片机通信；第 9 章 综合实例——自动报警系统。

本书的代码全部编译通过，并经过实际的测试，其中很多代码具有非常高的实用价值。

本书由李长林、高洁编写，参加编写的人员还有王波波、姜艳波、顾正大、艾丽香、赵辉、辛征、李志、王晶、兰婵丽、张玉平、赵光、王烁、刘群、赵木清、李刚、刘娜等，在此表示感谢！

由于时间仓促以及作者本身水平有限，书中难免会出现一些错误，恳请广大读者提出宝贵的意见和建议。

作 者
2006 年 2 月

目 录

第 1 章 串行口通信概述	1
1.1 接口的方式	1
1.1.1 并行接口	1
1.1.2 串行接口	2
1.2 串行接口的标准	2
1.2.1 RS-232、RS-422 与 RS-485 标准概述	2
1.2.2 RS-232 串行接口标准	3
1.2.3 三线制 RS-232C 串口通信接线方法	4
1.3 RS-422 与 RS-485 串行接口标准	5
1.3.1 RS-422 与 RS-485 的传输方式	5
1.3.2 RS-422 电气规定	5
1.3.3 RS-485 电气规定	6
1.3.4 接地	8
第 2 章 利用 MSComm 控件编程	10
2.1 MSComm 控件	10
2.1.1 MSComm 控件描述	10
2.1.2 MSComm 控件串口编程步骤	14
2.2 MSComm 控件串口数据通信——字符传输	18
2.2.1 串口通信实例	18
2.2.2 常见问题解决	25
第 3 章 Windows API 编程	29
3.1 Windows API 串口通信编程概述	29
3.1.1 16 位的 Windows API 通信函数	30
3.1.2 Windows 串口通信相关 API 函数	30
3.1.3 串口的参数设置	34
3.1.4 DCB 结构	34
3.1.5 其他设置	36
3.2 Windows API 串口编程	42
3.2.1 功能目标	43
3.2.2 主要技术/算法	43
3.2.3 具体实现	45
3.3 网络搜索程序	64
3.3.1 IE 的体系结构	64

3.3.2	WebBrowser Control 与 IE 的比较	65
3.3.3	搭建程序框架结构	67
3.3.4	代码实现	69
第 4 章	云台控制系统	90
4.1	云台控制系统简介	90
4.1.1	设备布局与线路连接	90
4.1.2	功能分析与设计	91
4.2	程序设计基础	93
4.2.1	数据编码	93
4.2.2	新建程序	94
4.3	代码分析与实现	96
4.3.1	资源代码的编写	96
4.3.2	程序中心代码	102
第 5 章	对 Modem 的控制	135
5.1	利用 MSComm 控件实现 Modem 的控制	135
5.1.1	功能目标	135
5.1.2	Modem 控件	135
5.1.3	关键技术与算法	137
5.1.4	对 Modem 控制的基本流程	139
5.1.5	代码分析与实现	146
5.2	用 AT 命令与 Modem 建立连接	167
5.2.1	功能目标	167
5.2.2	关键技术与算法	167
5.2.3	基本流程	168
5.2.4	代码分析与实现	170
第 6 章	TAPI 编程	211
6.1	TAPI 概述	211
6.1.1	TAPI 功能简介	211
6.1.2	TAPI 协议	213
6.1.3	TAPI 的体系	219
6.1.4	Windows 完整 TAPI	223
6.2	利用 TAPI 发送和接收传真实例	223
6.2.1	搭建程序框架结构	224
6.2.2	代码分析与说明	229
第 7 章	串口编程调试精灵	256
7.1	串口编程调试精灵简介	256

7.1.1	COM 接口技术	256
7.1.2	缓冲存储器	261
7.1.3	奇偶校验	261
7.2	串口编程调试精灵的基本流程	262
7.3	代码分析与实现	265
7.3.1	SSDlg 类的源代码	266
7.3.2	在 OnInitDlg()中添加对话框的初始化代码	267
7.3.3	CCWDlg 类	269
7.3.4	使用 CSSDlg.h 类设置发送字符串时间间隔	271
7.3.5	MSC.cpp 源代码	278
7.3.6	使用 MSCComm 类	287
7.3.7	CW.cpp 文件的实现	289
7.3.8	CW 类的源代码	291
7.4	设置与测试	292
7.4.1	串口编程调试精灵的安装项目的设置	292
7.4.2	对串口编程调试精灵的测试	293
第 8 章	VC 单片机通信	294
8.1	设计系统	294
8.1.1	系统设计	294
8.1.2	通信协议	295
8.2	VC 单片机通信的实现	295
8.2.1	界面设计	295
8.2.2	接口电路	296
8.2.3	单片机通信设计	297
8.2.4	程序设计核心代码	299
第 9 章	综合实例——自动报警系统	315
9.1	系统功能简介	315
9.1.1	系统功能设计	315
9.1.2	系统的执行步骤	316
9.2	数据库的实现	316
9.3	界面和代码设计	320
9.3.1	搭建程序变量框架结构	320
9.3.2	窗体功能	322
9.3.3	代码分析与说明	328

第 1 章 串行口通信概述

一般来说，计算机都有一个或多个串行端口，它们依次为 Com1、Com2、Com3、Com4 等。这些串口提供了外部设备与计算机进行数据传输和通信的通道。串行口通信是指外设和计算机间使用 3 根信号线进行数据的传送。这 3 根信号线分别是数据信号线、地线及控制线。

数据在数据信号线上是一位一位地进行传输的，每一位数据都占据一个固定的时间长度。这种通信方式使用的数据线少，在远距离通信中可以节省成本，当然，其传输速度比并行传输慢。如图 1-1 所示为串口示意图。

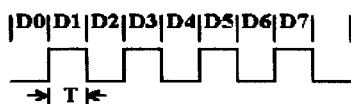


图 1-1 串口示意图

1.1 接口的方式

CPU 与外部设备、存储器的连接和数据交换都需要通过接口设备来实现，前者称为 I/O 接口，后者称为存储器接口。习惯上接口只是指 I/O 接口。I/O 接口通常分为并行接口、串行接口、磁盘接口、SCSI 接口及 USB 接口等几种。下面介绍并行接口和串行接口。

1.1.1 并行接口

并行接口是指 8 位数据同时通过并行线进行传送，这样数据传送速度能得到极大的提高，但并行传送的线路长度因为长度增加，干扰就会增加，容易出错，因此会受到限制。

如今，计算机中的并行接口主要作为打印机端口，接口使用的有 36 针接头和 25 针 D 形接头，目前以 25 针 D 形接头为主。现在有 5 种常见的并口：4 位、8 位、半 8 位、增强并行口 EPP 和扩展并行口 ECP。

4 位口一次只能输入 4 位数据，但可以输出 8 位数据；8 位口和半 8 位口可以一次输入和输出 8 位数据；EPP 口允许 8 位双向数据传送，可以连接各种非打印机设备，如扫描仪、LAN 适配器、磁盘驱动器和 CD-ROM 驱动器等；ECP 口能支持命令周期、数据周期和多个逻辑设备寻址，在多任务环境下可以使用直接存储器 DMA 访问。几

乎所有的 PC 机的主板都集成了并行口插座, 标注为 Parallel 或 LPT1, 是一个 25 针的双排针插座。

1.1.2 串行接口

串行口也是计算机的一种标准接口, PC 机一般至少有两个串行口 COM1 和 COM2。串行口不同于并行口之处在于, 它的数据和控制信息是一位接一位串行地传送下去。这样, 虽然速度会慢一些, 但传送距离较并行口更长, 因此长距离的通信应使用串行口。通常 COM1 使用的是 9 针 D 形连接器, 而 COM2 有些使用的是老式的 DB25 针连接器。

由于 CPU 与接口之间按并行方式传输, 接口与外设之间按串行方式传输, 因此, 在串行接口中, 要由接收移位寄存器把串行方式转换成并行方式, 由发送移位寄存器把并行方式转换成串行方式。完成这种转换功能的电路, 叫做通用异步收发器 UART。

1.2 串行接口的标准

串行通信接口标准经过使用和发展, 目前已经有 RS-232、RS-422 与 RS-485 等几种不同的标准。

1.2.1 RS-232、RS-422 与 RS-485 标准概述

RS-232、RS-422 与 RS-485 都是串行数据接口标准, 最初是由电子工业协会(EIA)制订并发布的。RS-232 在 1962 年发布, 命名为 EIA-232-E, 作为工业标准, 以保证不同厂家产品之间的兼容。

RS-422 由 RS-232 发展而来, 它是为弥补 RS-232 之不足而提出的。为改进 RS-232 通信距离短、速率低的缺点, RS-422 定义了一种平衡通信接口, 将传输速率提高到 10Mb/s。速率低于 100Kb/s 时, 传输距离延长到 4 000ft(英尺), 并允许在一条平衡总线上连接最多 10 个接收器。RS-422 是一种单机发送、多机接收的单向、平衡传输规范, 被命名为 TIA/EIA-422-A 标准。

为扩展应用范围, EIA 又于 1983 年在 RS-422 的基础上制定了 RS-485 标准, 增加了多点、双向通信能力, 即允许多个发送器连接到同一条总线上, 同时增加了发送器的驱动能力和冲突保护特性, 扩展了总线共模范围, 后命名为 TIA/EIA-485-A 标准。由于 EIA 提出的建议标准都以“RS”作为前缀, 所以在通信工业领域, 仍然习惯将上述标准以 RS 作前缀称谓。

RS-232、RS-422 与 RS-485 标准只对接口的电气特性作出规定, 而不涉及接插件、

电缆或协议，在此基础上用户可以建立自己的高层通信协议。因此在视频界的应用，许多厂家都建立了一套高层通信协议，或公开，或厂家独家使用。RS-232、RS-422 与 RS-485 的电气规定如表 1-1 所示。

表 1-1 RS-232、RS-422 与 RS-485 的电气规定表

规 定		RS-232	RS-422	RS-485
工作方式		单端	差分	差分
节点数		1 收、1 发	1 发 10 收	1 发 32 收
最大传输电缆长度		50ft	400ft	400ft
最大传输速率		20Kb/s	10Mb/s	10Mb/s
最大驱动输出电压		$\pm 25\text{V}$	$-0.25\sim+6\text{V}$	$-7\sim+12\text{V}$
驱动器输出信号电平 (负载最小值)	负载	$\pm 5\sim\pm 15\text{V}$	$\pm 2.0\text{V}$	$\pm 1.5\text{V}$
驱动器输出信号电平 (空载最大值)	空载	$\pm 25\text{V}$	$\pm 6\text{V}$	$\pm 6\text{V}$
驱动器负载阻抗(欧姆)		$3\text{k}\Omega\sim 7\text{k}\Omega$	100Ω	54Ω
摆率(最大值)		$30\text{V}/\mu\text{s}$	N/A	N/A
接收器输入电压范围		$\pm 15\text{V}$	$-10\sim+10\text{V}$	$-7\sim+12\text{V}$
接收器输入门限		$\pm 3\text{V}$	$\pm 200\text{mV}$	$\pm 200\text{mV}$
接收器输入电阻(欧姆)		$3\text{k}\Omega\sim 7\text{k}\Omega$	$4\text{K}\Omega$ (最小)	$\geq 12\text{k}\Omega$
驱动器共模电压			$-3\sim+3\text{V}$	$-1\sim+3\text{V}$
接收器共模电压			$-7\sim+7\text{V}$	$-7\sim+12\text{V}$

1.2.2 RS-232 串行接口标准

目前 RS-232 是 PC 机与通信工业中应用最广泛的一种串行接口。RS-232 被定义为一种在低速率串行通信中增加通信距离的单端标准。

RS-232 采取不平衡传输方式，即所谓单端通信。收、发端的数据信号相对于信号地。9 针串口引脚和 25 针串口引脚定义如表 1-2 所示。

表 1-2 9 针串口引脚和 25 针串口引脚定义

9 针串口(DB9)			25 针串口(DB25)		
针 号	功 能 说 明	缩 写	针 号	功 能 说 明	缩 写
1	数据载波检测	DCD	8	数据载波检测	DCD
2	接收数据	RxD	3	接收数据	RxD
3	发送数据	TxD	2	发送数据	TxD

(续表)

9 针串口(DB9)			25 针串口(DB25)		
针 号	功 能 说 明	缩 写	针 号	功 能 说 明	缩 写
4	数据终端准备	DTR	20	数据终端准备	DTR
5	信号地	GND	7	信号地	GND
6	数据设备准备好	DSR	6	数据准备好	DSR
7	请求发送	RTS	4	请求发送	RTS
8	清除发送	CTS	5	清除发送	CTS
9	振铃指示	DELL	22	振铃指示	DELL

典型的 RS-232 信号在正负电平之间摆动, 在发送数据时, 发送端驱动器输出正电平在+5~+15V, 负电平在 - 5~- 15V。当无数据传输时, 线上为 TTL, 从开始传送数据到结束, 线上电平从 TTL 电平到 RS-232 电平再返回 TTL 电平。接收器典型的工作电平是+3~+12V 和 - 3~- 12V。由于发送电平与接收电平的差仅为 2~3V 左右, 所以其共模抑制能力差, 再加上双绞线上的分布电容, 其传输距离最大约 15m, 最高速率为 20Kb/s。

RS-232 是为只用一对收、发设备通信而设计的, 其驱动器负载为 3~7kΩ。所以 RS-232 适合本地设备之间的通信。

RS-232C 是用正负电压来表示逻辑状态, 与 TTL 以高低电平表示逻辑状态的规定不同。因此, 为了能够同计算机接口或终端的 TTL 器件连接, 必须在 RS-232C 与 TTL 电路之间进行电平和逻辑关系的变换。实现这种变换的方法可用分立元件, 也可用集成电路芯片。目前较为广泛地使用集成电路转换器件, 如 MC1488、SN75150 芯片可完成 TTL 电平到 EIA 电平的转换, 而 MC1489、SN75154 可实现 EIA 电平到 TTL 电平的转换。MAX232 芯片可完成 TTL 到 EIA 双向电平的转换。

1.2.3 三线制 RS-232C 串口通信接线方法

首先, 串口传输数据只要有接收数据针脚和发送针脚就能实现。同一个串口的接收脚和发送脚直接用线相连, 两个串口相连或一个串口和多个串口相连。

同一个串口的接收脚和发送脚直接用线相连, 对 9 针串口和 25 针串口, 均是 2 与 3 直接相连。其连接方式如表 1-3 所示。

表 1-3 串口连接方式

9 针—9 针		25 针—25 针		9 针—25 针	
2	3	3	2	2	2
3	2	2	3	3	3
5	5	7	7	5	7

表 1-3 是对微机标准串行口而言的, 还有许多非标准设备则需要看具体情况而定。

1.3 RS-422 与 RS-485 串行接口标准

RS-422A 串行总线是一种常用的接口总线，支持一点对多点的通信。它在传输速率、传送距离及抗干扰性能等方面均优于 RS-232C，采用差分收发的工作方式，利用双端线来传送信号，最高数据传输速率为 10Mb/s，此时的传输距离为 120m，可连接 32 个收发器。如适当降低传输速率，可增加其通信距离。例如在 10kb/s 时距离可达 1200m。

RS-485 是一种典型的串行总线，支持一点对多点的通信，采用双绞线连接，可连接 32 个收发器，其他特性与 RS-422A 总线接近，在测控系统中得到较为普遍的应用，但不能满足高速测试系统的应用要求。

1.3.1 RS-422 与 RS-485 的传输方式

RS-422、RS-485 的数据信号采用差分传输方式，它使用一对双绞线，将其中一线定义为 A，另一线定义为 B，如图 1-2 所示。

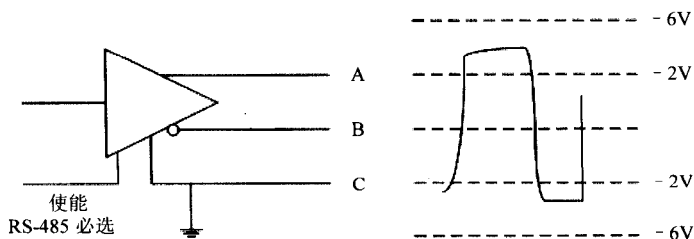


图 1-2 RS-422、RS-485 的传输方式

通常情况下，发送驱动器 A、B 之间的正电平在 +2~+6V 之间，是一个逻辑状态，负电平在 -2~-6V 之间，是另一个逻辑状态。另有一个信号地 C，在 RS-485 中还有一个“使能”端，而在 RS-422 中这是可用可不用的。“使能”端用于控制发送驱动器与传输线的切断与连接。当“使能”端起作用时，发送驱动器处于高阻状态，称作“第三态”，即它是有别于逻辑“1”与“0”的第三态。

接收器也作与发送端相对的规定，收、发端通过平衡双绞线将 AA 与 BB 对应相连，当在收端 AB 之间有大于 +200mV 的电平时，输出正逻辑电平，小于 -200mV 时，输出负逻辑电平。接收器接收平衡线上的电平范围通常在 200mV~6V 之间。

1.3.2 RS-422 电气规定

RS-422 标准的全称是“平衡电压数字接口电路的电气特性”，它定义了接口电路的特性。

典型的 RS-422 四线接口如图 1-3 所示。要注意的是，还有一根信号地线。

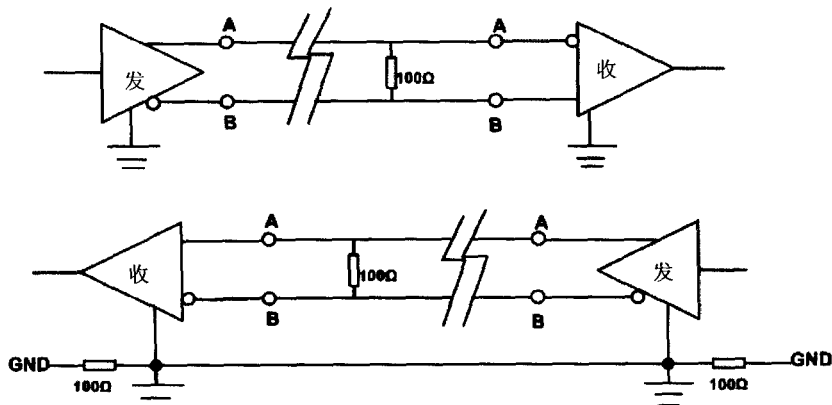


图 1-3 RS-422 四线接口

DB9 接口的引脚定义如图 1-4 所示。

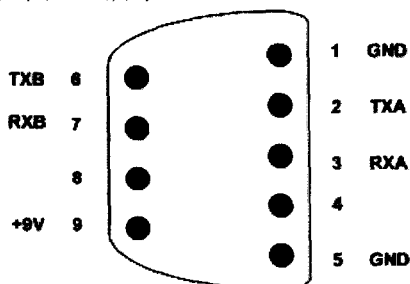


图 1-4 DB9 接口的引脚定义

由于接收器采用高输入阻抗和发送驱动器比 RS-232 更强的驱动能力，故允许在相同传输线上连接多个接收节点，最多可接 10 个节点，所以 RS-422 支持点对多的双向通信。

RS-422 四线接口由于采用单独的发送和接收通道，因此不必控制数据方向，各装置之间任何必需的信号交换均可以按软件方式或硬件方式。

RS-422 的最大传输距离为 1219m，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，在 100Kb/s 速率以下，才可能达到最大传输距离。只有在很短的距离下才能获得最高传输速率。一般 100m 长的双绞线上所能获得的最大传输速率仅为 1Mb/s。

RS-422 需要一终接电阻，要求其阻值约等于传输电缆的特性阻抗。在短距离传输时可不需终接电阻，即一般在 300m 以下不需终接电阻。终接电阻接在传输电缆的最远端。

1.3.3 RS-485 电气规定

RS-485 是从 RS-422 的基础上发展而来的，因此，RS-485 许多电气规定与 RS-422 相同。如都采用平衡传输方式、都需要在传输线上接终接电阻等。RS-485 可以采用二

线，也可以采用四线方式，如图 1-5 和 1-6 所示。

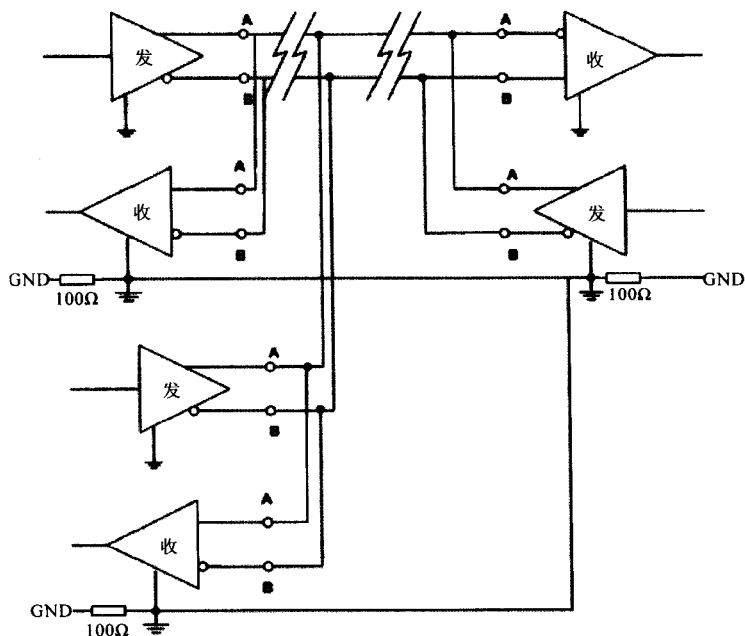


图 1-5 RS-485 采用二线

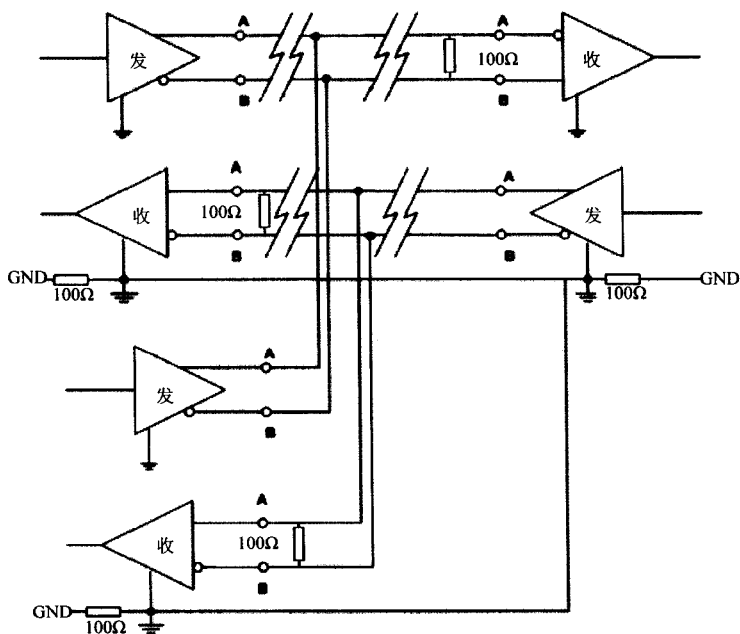


图 1-6 RS-485 采用四线

采用四线连接时，与 RS-422 一样只能实现点对多的通信，即只能有一个主(Master)设备，其余为从设备，但它比 RS-422 有改进，无论四线还是二线连接方式，总线上

可多接到 32 个设备。

RS-485 与 RS-422 的不同还在于其共模输出电压是不同的, RS-485 在 $-7\sim+12\text{V}$ 之间, 而 RS-422 在 $-7\sim+7\text{V}$ 之间, RS-485 接收器最小输入阻抗为 $12\text{k}\Omega$ 。RS-485 满足所有 RS-422 的规范, 所以 RS-485 的驱动器可以在 RS-422 网络中应用。

RS-485 与 RS-422 一样, 其最大传输距离约为 1219m , 最大传输速率为 10Mb/s 。平衡双绞线的长度与传输速率成反比, 在 100Kb/s 速率以下, 才可能使用规定最长的电缆长度。只有在很短的距离下才能获得最高的传输速率。一般 100m 长双绞线最大传输速率仅为 1Mb/s 。

RS-485 需要 2 个终接电阻, 其阻值要求等于传输电缆的特性阻抗。在短距离传输时可不需终接电阻。

理论上, 在每个接收数据信号的中点进行采样时, 只要反射信号在开始采样时衰减到足够低就可以不考虑匹配。

当信号的转换时间超过电信号沿总线单向传输所需时间的 3 倍以上时就可以不加匹配。例如具有有限斜率特性的 RS-485 接口 MAX483 输出信号的上升或下降时间最小为 250ns , 典型双绞线上的信号传输速率约为 0.2m/ns (24AWG PVC 电缆), 那么只要数据速率在 250Kb/s 以内、电缆长度不超过 16m , 采用 MAX483 作为 RS-485 接口时就可以不加终端匹配。

RS-422 在总线电缆的远端并接电阻, RS-485 则在总线电缆的开始和末端都需并接终接电阻。终接电阻一般在 RS-422 网络中取 100Ω , 在 RS-485 网络中取 120Ω , 相当于电缆特性阻抗的电阻, 因为大多数双绞线电缆特性阻抗大约在 $100\sim120\Omega$ 。这种匹配方法简单有效, 但有一个缺点, 匹配电阻要消耗较大功率。

1.3.4 接地

RS-422 与 RS-485 接地系统不合理会影响整个网络的稳定性, 尤其是在工作环境比较恶劣和传输距离较远的情况下, 对于接地的要求更为严格。否则接口损坏率较高。

通常, 连接 RS-422、RS-485 通信链路时只是简单地用一对双绞线将各个接口的 A、B 端连接起来。而忽略了信号地的连接, 这种连接方法在许多场合是能正常工作的, 但却产生两个问题, 共模干扰和 EMI。

1. 共模干扰

RS-422 与 RS-485 接口均采用差分方式传输信号, 并不需要相对于某个参照点来检测信号, 系统只需检测两线之间的电位差就可以了。但收发器有一定的共模电压范围, 如 RS-422 共模电压范围为 $-7\sim+7\text{V}$, 而 RS-485 收发器共模电压范围为 $-7\sim+12\text{V}$, 只有满足上述条件, 整个网络才能正常工作。当网络线路中共模电压超出此范围时就会影响通信的稳定可靠, 甚至损坏接口。

RS-422 与 RS-485 标准均规定共模电压 $\leq 3\text{V}$, 但地电位差可能会有十几伏甚至数

十伏，并可能伴有强干扰信号，致使接收器共模输入共模电压超出正常范围，并在传输线路上产生干扰电流，轻则影响正常通信，重则损坏通信接口电路。

2. EMI

发送驱动器输出信号中的共模部分需要一个返回通路，如没有一个低阻的信号地，就会以辐射的形式返回源端，整个总线就会像一个巨大的天线向外辐射电磁波。

由于上述原因，RS-422、RS-485 尽管采用差分平衡传输方式，但对整个 RS-422 或 RS-485 网络，必须有一条低阻的信号地。一条低阻的信号地将两个接口的工作地连接起来，使共模干扰电压被短路。这条信号地可以是额外的一条线，或者是屏蔽双绞线的屏蔽层。这是最通常的接地方法。

值得注意的是，这种做法仅对高阻型共模干扰有效，由于干扰源内阻大，短接后不会形成很大的接地环路电流，因此对于通信不会有很大影响。当共模干扰源内阻较低时，会在接地线上形成较大的环路电流，影响正常通信。

第2章 利用 MSComm 控件编程

MSComm 控件在串口编程时非常方便，程序员不必花费时间去了解较为复杂的 API 函数，而且在 Visual C++、Visual Basic 及 Delphi 等语言中均可使用。Microsoft Communications Control(以下简称 MSComm)是 Microsoft 公司提供的简化 Windows 下串行通信编程的 ActiveX 控件，它为应用程序提供了通过串行接口收发数据的简便方法。

2.1 MSComm 控件

MSComm 控件通过串行端口传输和接收数据，为应用程序提供串行通信功能。

首先我们来了解一下 MSComm 控件的常用属性，MSComm 控件有很多重要的属性，但首先必须熟悉如下几个属性：

- **CommPort** 设置并返回通信端口号。
- **Settings** 以字符串的形式设置并返回波特率、奇偶校验、数据位、停止位。
- **PortOpen** 设置并返回通信端口的状态。也可以打开和关闭端口。
- **Input** 从接收缓冲区返回和删除字符。
- **Output** 向传输缓冲区写一个字符串。

2.1.1 MSComm 控件描述

1. CommPort 属性

作用：设置并返回通信端口号。

语法：object.CommPort[value] (value —— 整型值，说明端口号)。

说明：在设计时，value 可以设置成从 1 到 16 的任何数(默认值为 1)。但是如果用 PortOpen 属性打开一个并不存在的端口，MSComm 控件会产生错误 68(设备无效)。

注意：必须在打开端口之前设置 CommPort 属性。

2. RThreshold 属性

作用：在 MSComm 控件设置 CommEvent 属性为 comEvReceive 并产生 OnComm 之前，设置并返回要接收的字符数。

语法：object.Rthreshold [= value] (value —— 整型表达式，说明在产生 OnComm