

章 绪论

1.1

21 世纪是经济竞争的时代。串口通信的应用使得计算机对工程实现实时监控，能够在相关工业实践中帮助企业提高效率，带来更多的效益。以串口通信应用为重点的企业要想在当前国内竞争日益激烈的环境中生存和发展，关键之一就在与注重串口调试系统的开发和管理维护，因而充分利用和发挥生产管理的优势和潜力，取得更大的社会和经济效益。科技的发展及计算机的广泛运用，串口通信的设计与实现将管理信息化和高效化，为提高企业的核心竞争力提供了广阔的平台。

1.2 本课题的发展现状与存在的问题

串口通信的设计与实现，常用的开发环境有 visual basic 和 Delphi；工具有 MSComm 控件、Windows API 编程、SPComm 控件、APRO 组件等。但是无论在开发环境还是工具的选择上，其实现数据的发送、接收功能部分都是大同小异的，也就是先对开发环境和工具组合，然后进行编程。所进行的串口通信设计都是在开发环境和工具的组合下完成的，未来的开发如果能设计出一套系统的、简单易用的串口通信设计工具，能很好的用与后期深层次开发以及其他资源和串口外设进行交互使用，带来的经济效益和社会效益会更大。

1.3 课题来源与主要研究内容

随着当今串口通信技术的发展，串口开发环境和工具功能的改善，以往的串口通信设计已经不能满足当今的需求，所以急需串口通信的设计能突破以前的常规思想，做到新颖、简单、可操作性强等特点。

本次串口通信的设计与实现，主要研究在什么环境下利用什么工具开发，利用什么原理才能比较简单的进行串口通信设计，详细设计其中的每一模块，因而达到设计简单，构思独特，用户界面友好，操作方便，可移植性好等特点。

章 相关技术介绍

2.1

在通常情况下，人们是通过外部设备使用计算机的，由于外设往往不能与 CPU 直接相连，所以他们之间的信息交换一般需要通过接口完成。接口包括主机提供的一组线路和接口电路以及外部设备内部的接口，它们实现规定的接口协议，使得不同外设能够与主机和 CPU 实现互连。

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，前者被称为 I/O 接口，而后者则被称为存储器接口。存储器通常在 CPU 的同步控制下工作，接口电路比较简单；而 I/O 设备品种繁多，其相应的接口电路也各不相同，因此，习惯上说到接口只是指 I/O 接口，称为外部设备接口技术或者输入/输出接口技术。I/O 接口根据接口数据传输形式的不同，可分为并行总线接口和串行总线接口两大类。并行口速度快，实时性好，但是占用的线多，比较复杂，如 IEEE-488。串行口速度一般比较慢，但是简单、灵活、方便，如 RS-232、USB 等。^[10]

2.1.1 并行接口

实现并行通信的接口就是并行接口。主机与接口、接口与外设之间都是以并行方式传送数据。也就是每次传送一个字或者字节的全部代码。并行通信是把一个字符的各数位用几条线同时进行传输，传输速度快，信息率高。但它比串行通信所用的电缆多，故常用在传输距离较短（几米至几十米）、数据传输率较高的场合。

并行接口可设计为只作为输入/输出接口，也可设计为既作为输入又作为输出的接口。它可以用两种方法实现，一种是利用同一个接口中的两个通路，一个作输入通路，一个作输出通路；另一种使用同一个双向通路，既作为输入又作为输出。

连接设备接口有 PS/2，PATA，LPT 等。

其特点有：

- 1) 接口中具有与系统总线直接相连的输入/输出数据缓冲器和一组数据线；
- 2) 接口中具有与外部设备进行联络的应答信号；
- 3) 接口中具有地址译码电路，产生片选择信号和内部寄存器地址选择；
- 4) 接口可使用程序传送方式或程序中断传送方式实现外设与主机之间的信息交换；
- 5) 简单并行接口只有一个 I/O 通路驱动外设，对于复杂的、可编程并行接口可有几个 I/O 通路驱动不同外设的输入/输出。^[14]

2.1.2 串行 RS-232 接口

计算机与计算机或计算机与终端之间的数据传送可以采用串行通讯和并行通讯二种方式。由于串行通讯方式具有使用线路少、成本低，特别是在远程传输时，避免了多条线路特性的不一致而被广泛采用。在串行通讯时，要求通讯双方都采用一个标准接口，使不同的设备可以方便地连接起来进行通讯。

RS-232-C **EIA RS-232-C**) 是目前最常用的一种串行通信接口。串行通信接口标准经过使用和发展, 目前已经有多种, 但都是在 **RS-232** 标准的基础上经过改进而形成的。**RS-232-C** 是在 1970 年由美国电子工业协会 (EIA) 联合贝尔系统、调制解调器厂家及计算机终端生产厂家共同制定的用于串行通信的标准。它的全名是“数据终端设备 (DTE) 和数据通讯设备 (DCE) 之间串行二进制数据交换接口技术标准”。在 PC 机上一般至少有两个串行口 **COM1** 和 **COM2**, 老式的 **DB25** 针连接器引线中有许多是很少使用的, 而在计算机与终端通讯中一般只使用 3-9 条引线, 9 针 **D** 形连接器比较常见。

常用的串行外设有电传打字机、**CRT** 终端设备、传真机以及远程数据采集设备、通信设备等。^[9]

在 **RS-232-C** 中任何一条信号线的电压均为负逻辑关系。即: 逻辑“1”, -5 — -15V; 逻辑“0” +5 — +15V。噪声容限为 2V。即要求接收器能识别低至+3V 的信号作为逻辑“0”, 高到-3V 的信号作为逻辑“1”。一些设备与 PC 机连接的 **RS-232-C** 接口, 因为不使用对方的传送控制信号, 只需三条接口线, 即“发送数据”、“接收数据”和“信号地”。

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, 9 针串口和 25 针串口, 均是 2 与 3 直接相连。^{[10][11]}

DB9 和 **DB25** 的常用信号脚说明如表 2-1 所示:

表 2-1 **DB9** 和 **DB25** 的常用信号脚说明

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
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	数据设备准备好	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

RS-232-C 串口通信接线原理 (三线制) 如表 2-2 所示:

表 2-2 **RS-232-C** 串口通信接线原理

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

● 应该注意的是:

1 RS-232-C 串口通信接线的原则简单的说就是：数据针脚或线相连，彼此交叉，信号地对应相接。

2) 不同编码机制不能混接，如 RS-232-C 不能直接与 RS-422 接口相连，必须通过转换器才能连接；

3) 线路焊接要牢固，不然程序没问题，却因为接线问题而出现错误；

4) 不要带电插拔串口，插拔时至少有一端是断电的，否则串口容易损坏。^[12]

2.1.3 其他相关串行端口

RS-422、RS-485 与 RS-232 不一样，数据信号采用差分传输方式，也称平衡传输，使用的是一对双绞线。

RS-422 标准全称是“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。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。即一个主设备（Master），其余为从设备（Slave），从设备之间不能通信，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，各装置之间任何必须的信号交换均可以按软件方（XON/XOFF 握手）或硬件方式（一对单独的双绞线）实现。

由于 RS-485 是从 RS-422 基础上发展而来的，所以 RS-485 许多电气规定与 RS-422 相仿。如都采用平衡传输方式、都需要在传输线上接终端电阻等。RS-485 可以采用二线与四线方式，二线制可实现真正的多点双向通信。而采用四线连接时，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。^{[13][16][17]}

2.2.1 异步通信和同步通信

在网络通信过程中，通信双方要交换数据，需要高度的协同工作。为了正确的解释信号，接收方必须确切地知道信号应当何时接收和处理，因此定时是至关重要的。在计算机网络中，定时的因素称为位同步。同步是要接收方按照发送方发送的每个位的起止时刻和速率来接收数据，否则会产生误差。

在串行通信中有两种基本的通信方式：异步通信 ASYNC 和同步通信 SYNC。

● 异步通信

CPU 与外设之间有两项约定：

1) 字符格式-字符的编码形式及规定，每个串行字符由以下四个部分组成：

[1] 1 个起始位，低电平；

[2] 5-8 个数据位；

[3] 1 个奇偶校验位；

[4] 1-2 个终止位(停止位)高电平。

2) 波特率

波特率是指单位时间内传送二进制数据的位数，以位/秒位单位，它是衡量串行数据传送速度快慢的重要标志和参数。波特率是一个衡量通信速度的参数，它

bit 的个数。假如数据出送速率是 120 位/秒，则传送的波特率为：120 波特。异步通信的传送速度一般在 50 波特----19200 波特之间。

异步传输将比特分成小组进行传送，小组可以是 8 位的 1 个字符或更长。发送方可以在任何时刻发送这些比特组，而接收方从不知道它们会在什么时候到达。一个常见的例子是计算机键盘与主机的通信。按下一个字母键、数字键或特殊字符键，就发送一个 8 比特位的 ASCII 代码。键盘可以在任何时刻发送代码，这取决于用户的输入速度，内部的硬件必须能够在任何时刻接收一个键入的字符。

异步传输的实现比较容易，由于每个信息都加上了“同步”信息，因此计时的漂移不会产生大的积累，但却产生了较多的开销。在上面的例子，每 8 个比特要多传送两个比特，总的传输负载就增加 25%。对于数据传输量很小的低速设备来说问题不大，但对于那些数据传输量很大的高速设备来说，25% 的负载增值就相当严重了。因此，异步传输常用于低速设备。

● 同步通信

在异步通信中，每个字符要用起始位和终止位作为字符开始和结束的标志，占用了一些时间，为了提高数据块的传送速度，就要设法去掉这些标志，就采用同步通信。同步通信的速度可达 500 千波特。但是它的硬件电路比较复杂。同步传输的比特分组要大得多。它不是独立地发送每个字符，每个字符都有自己的起始位和停止位，而是把它们组合起来一起发送。我们将这些组合称为数据帧，或简称为帧。

数据帧的第一部分包含一组同步字符，它是一个独特的比特组合，类似于前面提到的起始位，用于通知接收方一个帧已经到达，但它同时还能确保接收方的采样速度和比特的到达速度保持一致，使收发双方进入同步。

帧的最后一部分是一个帧结束标记。与同步字符一样，它也是一个独特的比特串，类似于前面提到的停止位，用于表示在下一帧开始之前没有别的即将到达的数据了。

同步传输通常要比异步传输快速得多。接收方不必对每个字符进行开始和停止的操作。一旦检测到帧同步字符，它就在接下来的数据到达时接收它们。另外，同步传输的开销也比较少。^{[15][8]}

2.2.2 单工、半双工和全双工通信

根据信息的传送方向，串行通信可以进一步分为单工、半双工和全双工三种。如果在通信过程的任意时刻，信息只能由一方 A 传到另一方 B，则称为单工。单工通信有明确的传输方向。如果在任意时刻，信息既可由 A 传到 B，又能由 B 传 A，但只能由一个方向上的传输存在，称为半双工传输。半双工通信是双向的，但是必须轮流发送。如果在任意时刻，线路上存在 A 到 B 和 B 到 A 的双向信号传输，则称为全双工。全双工通信能同时双向传输数据。^[14]

单工、半双工和全双工通信原理如表 2-3 所示：

2-3 单工 半双工和全双工通信

	A→B	A→B
	B→A	B→A
单工	半双工	全双工
有明确的传输方向	双向的，但是必须轮流发送	同时双向传输数据

2.3.1 奇偶校验

传输中的差错都是由于噪声引起的。串行数据在传输过程中，由于干扰可能引起信息的出错，由于干扰，可能使位改变，这种情况，我们称为出现了“误码”。我们把如何发现传输中的错误，叫“检错”。发现错误后，如何消除错误，叫“纠错”。

最简单的检错方法是“奇偶校验”，即在传送字符的各位之外，再传送 1 位奇/偶校验位，用一个值确保传输的数据有偶个或者奇个逻辑高位。这样使得接收设备能够知道一个位的状态，有机会判断是否有噪声干扰了通信或者是否传输和接收数据是否不同步。奇偶校验能够检测出信息传输过程中的部分误码（1 位误码能检出，2 位及 2 位以上误码不能检出），但是，它不能纠错。

有些检错方法，具有纠错能力。如定比码、正反码、海明码和循环冗余码检错等。^[11]

2.3.2 数据位

数据位是衡量通信中实际数据位的参数。当计算机发送一个信息包，实际的数据不会是 8 位的，标准的值是 5、7 和 8 位。如何设置取决于传送的信息。比如，标准的 ASCII 码是 0~127（7 位）。扩展的 ASCII 码是 0~255（8 位）。如果数据使用简单的文本（标准 ASCII 码），那么每个数据包使用 7 位数据。每个包是指一个字节，包括开始/停止位，数据位和奇偶校验位。由于实际数据位取决于通信协议的选取，术语“包”指任何通信的情况。

2.3.3 停止位

停止位是用于表示单个包的最后一位。典型的值为 1，1.5 和 2 位。但是在 MSComm 控件中不能指定为 1.5，否则会出现错误。在由于数据是在传输线上定时的，并且每一个设备有其自己的时钟，很可能在通信中两台设备间出现了小小的不同步。因此停止位不仅仅是表示传输的结束，并且提供计算机校正时钟同步的机会。适用于停止位的位数越多，不同时钟同步的容忍程度越大，但是数据传输率同时也越慢。^[10]

2.4 MSComm 控件介绍

● MSComm 控件提供下列两种处理通讯的方式：事件驱动方式和查询方式。

1) 事件驱动方式：

事件驱动通讯是处理串行端口交互作用的一种非常有效的方法。在许多情况下，在事件发生时需要得到通知。

2) 查询方式：

在程序的每个关键功能之后，可以通过检查 **CommEvent** 属性的值来查询事件和错误。

MSComm 控件常用属性

MSComm 控件有很多重要的属性，最重要的有以下几个：

CommPort 设置并返回通讯端口号；

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，在 MSComm 控件中，停止位不能设置为 1.5，否则会出现错误；

PortOpen 设置并返回通讯端口的状态，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。

● MSComm 控件的事件

MSComm 控件只有一个事件，即 **OnComm** 事件。

利用 MSComm 控件编写的应用程序在通信时候如果发生错误或者 **OnComm** 事件，会引起 **OnComm** 事件并且改变其属性值，通过 **GetCommEvent()** 可获得 **OnComm** 产生事件或者错误的代码。在通信程序的设计中可以根据 **GetCommEvent** 属性值来执行不同的操作。^[10]

章 系统分析和开发方案

3.1

在设计串口通信前我对现行情况做了初步调查，调查内容如下：

(1) 现行主流开发技术：如何在 Windows 平台下实现串口通信，越来越多地被得到研究。现行主要用 visual basic 和 Delphi 环境加控件模式开发，适用范围广泛。重要的有利用 MSComm 控件和直接用 Windows API 编程。利用 MSComm 控件编程相对来说更简单一些，而直接使用 Windows API 编程更灵活一些。

(2) 技术可行性情况：能了解串口通信的原理，在开发环境下熟练编程。

(3) 现行设计的概况：串口程序界面美观，程序设计清晰，可移植性好，方便进行二次开发。

串口通信的设计与实现，在硬件方面，要求计算机性能良好，硬盘可用空间在 10G 以上，内存为 256M 或更高的；操作系统是 Windows98/2000/XP，能运行绝大多数应用软件，运行串口通信程序完全没问题。软件方面看，选用开发系统的主流开发环境和开发工具，技术相当成熟；经济方面，基本上无任何经济问题和风险；社会方面，给串口通信相关应用领域所带来的方便性和实用性大大提高。

综合可行性分析的结果，本次串口通信的设计与实现是完全可行的。^{[4][5]}

3.2 需求分析

需求是用户对目标软件系统在功能、行为、性能、设计约束等方面的期望。通过对应用问题及其环境的理解与分析，为问题涉及的信息、功能及系统行为建立模型，将用户需求精确化、完全化，最终形成需求规格说明。

需求分析是介于系统分析和软件设计阶段之间的重要桥梁。是管理信息系统开发工作中重要的、必不可少的环节，是建立管理信息系统的关键，确定的内容是后面系统设计、系统实现的基础。需求分析可分为问题分析、需求描述及需求评审三个阶段。^{[1][2]}

串行端口的本质功能是作为 CPU 和串行设备间的编码转换器。当数据从 CPU 经过串行端口发送出去时，字节数据转换成串行的位。在接收数据时，串行的位被转换为字节数据。串口设计原理的核心主要是对串口属性进行相关的设置，完成数据的发送和接收及其实现方式。一方面要考虑数据发送和接收的完整性与时效性，另一方面有要考虑用户界面的简单易用，程序设计清晰，以及代码的可移植性好，方便进行二次开发。

3.3 功能分析

3.3.1 系统功能介绍

在打开串口的设置上，比较直观和人性化，红灯亮表示串口打开，绿灯亮表示串口关闭，能直观的反映串口的实时情况；串口的相关设置上，简单的实现波特率、数据位和停止位的修改和转换；对发送窗口和接收窗口的现行数据进行清空处理；实现对串口数据的发送和接收功能，并且可以进行定时发送；未打开串口就进行数据发送会有相关的提示；本次串口设计未使用到奇偶校验。

3.3.2 系统数据流图

进行串口属性的设置，包括设置端口号、波特率、校验位、数据位、停止位、自动发送间隔的时间，然后打开串口和输入要发送的数据，设置发送和接收的数据类型，就可以发送和接受数据了。

系统的数据流程图如图 3-1 所示：

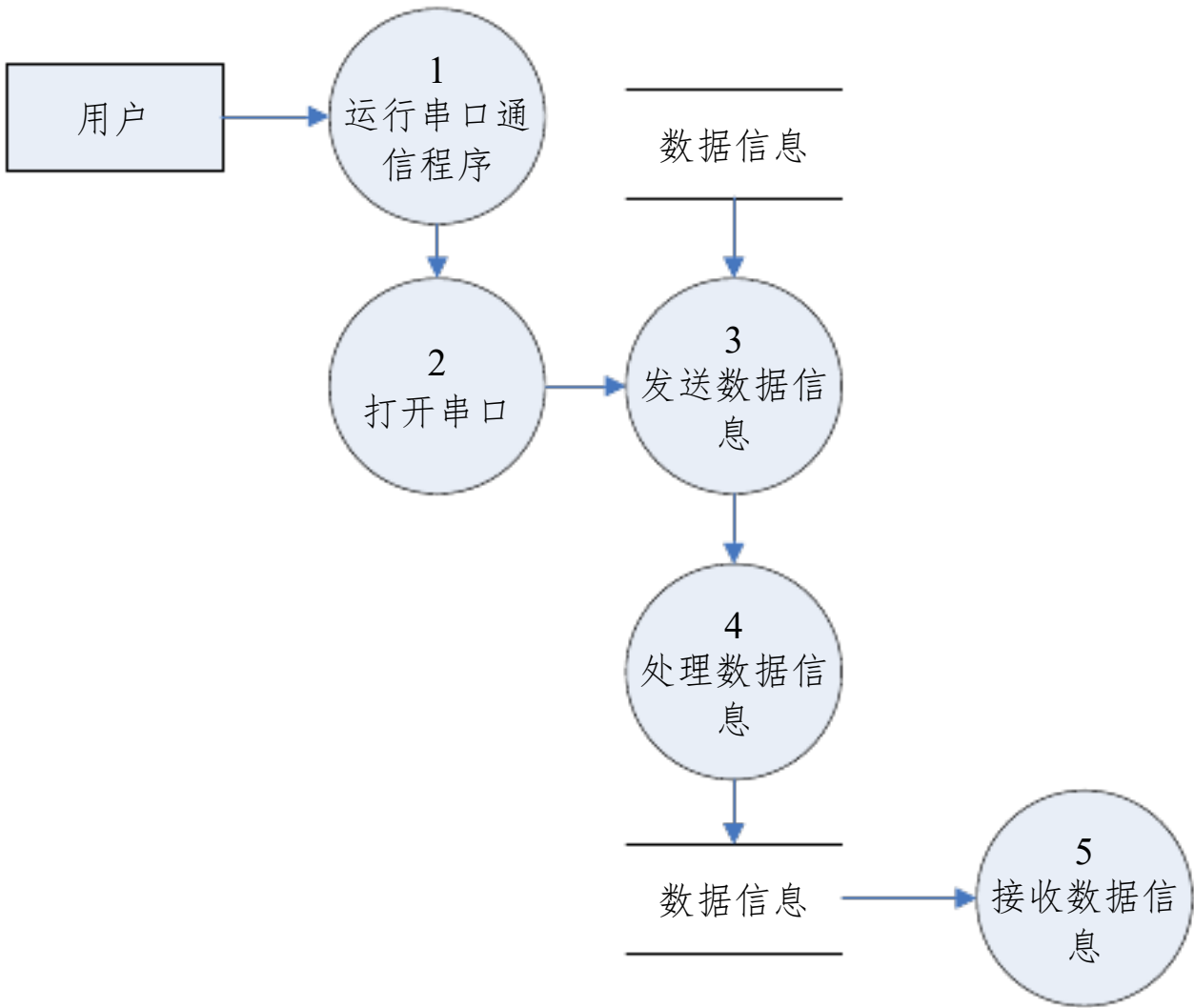


图 3-1 串口通信数据流图

3.4.1 RS-232 连线方案

本次串口通信设计与实现中，RS-232 的连线方案选择了三线制，也就是数据针脚或线相连，彼此交叉，信号地对应相接，其中用了两个 9 针 D 形连接器，接线方案如图 3-2 所示：

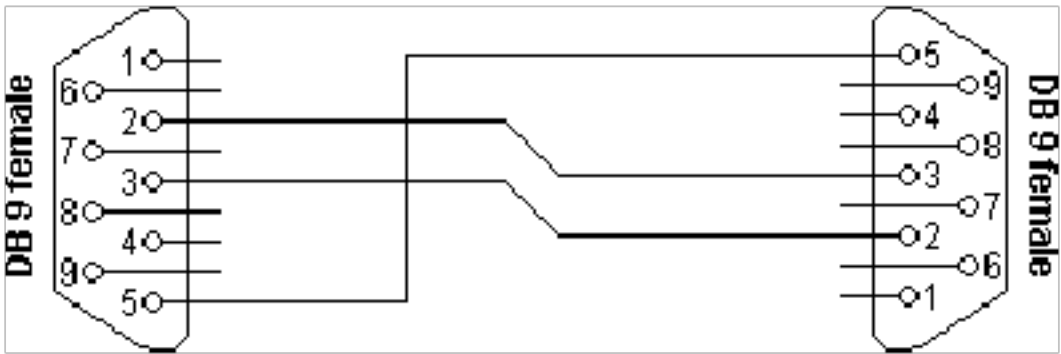


图 3-2 RS232 连线图

3.4.2 环境和控件选择方案

本设计是在 Delphi7 环境的基础上，运用微软 MSComm 控件和 RS-232 串口通信原理，采用异步通信和全双工模式，遵循严格认真的设计开发原则，根据相

Windows 操作系统下实现串口通讯发送、接收功能。本次设计中没有使用到奇偶校验。

本设计有以下的特点：

- 1) 开发环境和工具简单，降低开发成本；
- 2) 严格按照系统工程的方法进行开发，提高开发效率；
- 3) 用户界面简单清晰，方便用户使用；
- 4) 方便的实现串口通讯发送、接收功能；
- 5) 代码的可移植性好。

● 在 Delphi 环境下利用 MSComm 控件开发模式的选择

Microsoft Communications Control（简称 MSComm 控件）是微软的 Visual Basic 中提供的一个串口控件，在 Visual C++、Delphi 等开发环境中均可使用，使用简单、性能良好。它通过串行端口传送和接收数据，为应用程序提供基本的串行通信功能，使程序员不必花时间去了解较为复杂的 API 函数，所以得到广泛的使用。^[8]

Borland 公司推出的 Delphi 是一种功能强大的高级编程语言，其具有的可视化面向对象的特性，特别适于在 Windows 环境下图形界面和用户程序的编制。Delphi 是一种具有功能强大、简便易用和代码执行速度快等优点的可视化快速应用开发工具，利用封装性、继承性、多态性等面向对象的编程技术，开发人员可以有效地利用所创建的对象，使应用程序具有可重用性、可扩展性和其它强有力的功能。它在构架企业信息系统方面发挥着越来越重要的作用，用 Delphi 作为开发工具编制各种应用程序是很多编程人员的选择。

但是，美中不足之处是 Delphi 没有自带的串口通信控件，在它的帮助文档里也没有提及串口通信，这就给编制通信程序的开发人员带来许多不便。目前，利用 Delphi 实现串口通信的常用的方法有 3 种：一是利用控件，如 MSCOMM 控件和 SPCOMM 控件；二是使用 API 函数；三是调用其他串口通信程序。其中利用 API 编写串口通信程序较为复杂，需要掌握大量的通信知识；调用其他串口通信程序也未能掌握；而使用控件则简单容易操作，所以选择在 Delphi 环境下利用 MSComm 控件开发的模式。这种模式设计清晰，由于是使用 Delphi 作为开发环境，代码的可移植性强，后期开发前景一片光明。^{[3][8]}

章 系统设计

4.1

本次串口通信的设计，主要的有三大功能模块：串口设置，串口控制和数据显示。其中串口设置中包括串口属性设置模块和打开串口模块；串口控制中包括数据输入模块、数据发送模块和数据接收模块；数据显示中包括十六进制显示模块和文本方式显示模块。

系统的功能模块图如图 4-1 所示：

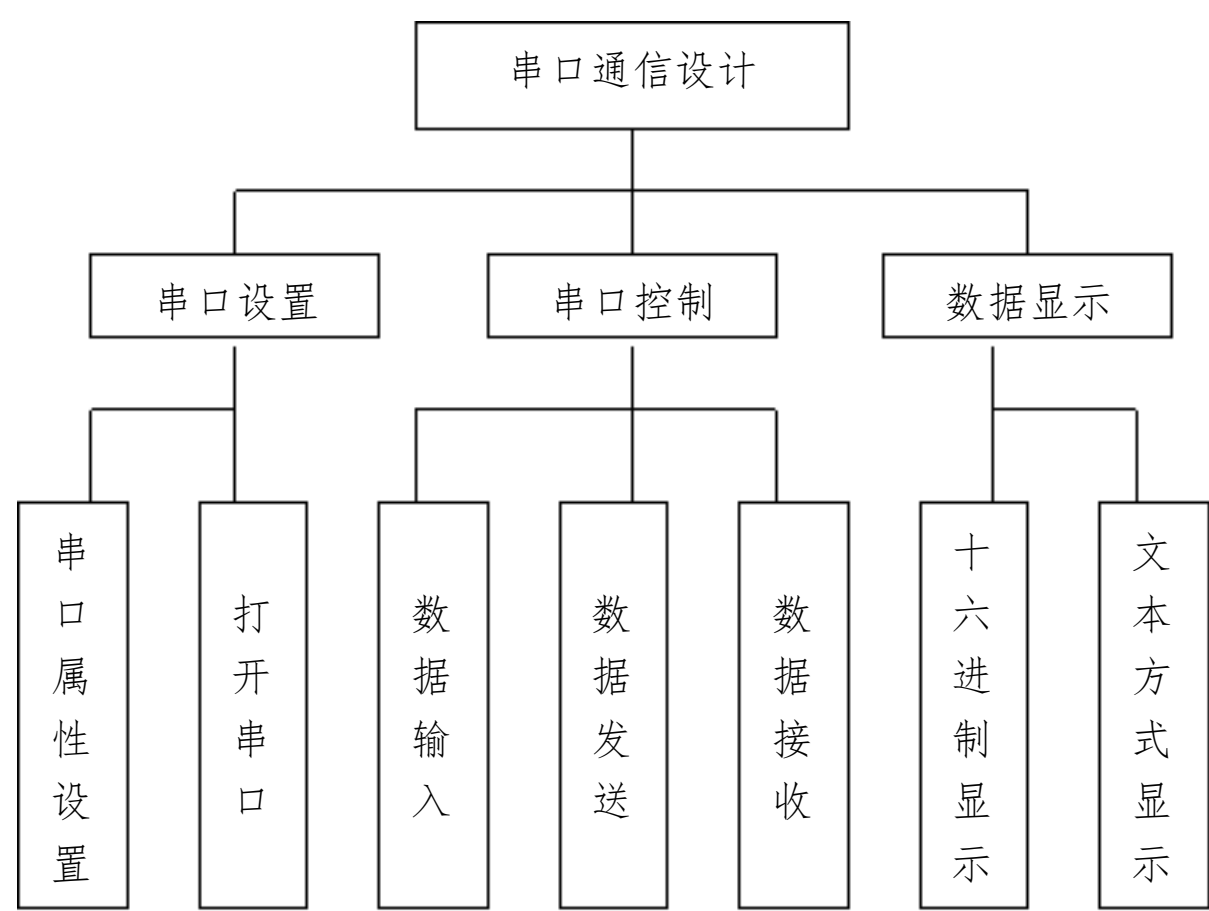


图 4-1 系统功能模块图

4.2 详细设计

4.2.1 系统执行流程

详细设计中应该遵循流程进行，系统的规范进行设计，避免开发中组件使用的不规范，代码的混乱，造成系统功能的不全面，代码的可移植性差等缺点。

系统的执行流程图如图 4-2 所示：

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/337100112001006102>