



案例解说

Delphi典型控制应用

七种典型控制系统，贴近工程实践

●单片机 ●PLC ●PCI数据采集卡 ●USB数据采集板
●CAN数据采集模块 ●远程I/O模块 ●智能仪器



配套光盘 超值配套光盘，内含：

- ◆ 完整案例程序——提供所有控制系统的完整源程序，并经编译和实际测试。
- ◆ 程序运行录屏——录制程序的运行过程，便于读者了解程序操作使用方法。
- ◆ 系统测试录像——录制、讲解系统的硬件组成，以及软件、硬件测试过程。
- ◆ 丰富学习资源——提供各公司硬件驱动程序、手册，以及图片等丰富资源。

朱东芹

郑瑶

李江全

等编著



案例解说控制应用精品丛书

案例解说 Delphi 典型控制应用

朱东芹 郑 瑶 李江全 等编著

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本书从工程应用的角度出发,通过7种典型的计算机控制系统(包括基于单片机、PLC、PCI数据采集卡、USB数据采集板、CAN数据采集模块、远程I/O模块、智能仪器),使用目前流行的面向对象编程语言Delphi,对工业控制系统中的4类典型应用,即模拟量输入(AI)、模拟量输出(AO)、数字量输入(DI)和数字量输出(DO)的程序设计方法进行了详细的讲解。

本书内容丰富,每个案例均提供了具体的设计任务、详细的操作步骤和完整的程序代码,同时注重解决工程实际问题。本书可供各类自动化、计算机应用、机电一体化等专业的大学生和研究生学习计算机控制技术使用,也可供计算机控制系统研发的工程技术人员参考。

为方便读者学习,本书提供超值配套光盘,内容包括所有案例的源程序、软硬件资源、程序运行录屏和系统测试录像等。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

案例解说 Delphi 典型控制应用 / 朱东芹等编著. —北京: 电子工业出版社, 2011.4
(案例解说控制应用精品丛书)

ISBN 978-7-121-13140-0

I. ①案… II. ①朱… III. ①软件工具—程序设计 IV. ①TP311.56

中国版本图书馆 CIP 数据核字 (2011) 第 046020 号

责任编辑: 陈韦凯

特约编辑: 肖晓强

印 刷: 北京市顺义兴华印刷厂

装 订: 三河市双峰印刷装订有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1 092 1/16 印张: 20.75 字数: 531.2 千字

印 次: 2011 年 4 月第 1 次印刷

册 数: 4 000 册 定价: 49.00 元 (含光盘 1 张)



凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。

前 言

计算机控制技术在通信、遥感、无损检测、智能仪器、工业自动控制等工程领域有着广泛的应用。在开发计算机控制系统时，程序设计是很多技术人员要面临的问题。在高校和科研院所，有众多的研究人员在使用各种计算机控制系统，他们都迫切需要相关的书籍来帮助他们学习相关的编程技术。

Delphi 是美国 Borland 公司推出的基于面向对象程序设计语言的可视化集成开发工具，是源于 Object Pascal 的一种现代程序设计语言。Delphi 为程序开发人员提供了快速应用程序开发的理念，具有编译速度快、IDE 反应速度快、完全开放的 VCL 源代码、领先的数据库技术等优点。使用 Delphi 开发应用程序，将会大大提高编程效率，使您轻松完成工作。

用 Delphi 实现数据采集，一般需要编写 DLL 和 ActiveX 组件，然后通过 Delphi 的 API 功能调用和组件调用，实现对模拟量输入/输出、数字量输入/输出及计数等功能。就 Delphi 应用来说，一般厂商都为他们的数据采集卡提供了丰富的 DLL 函数和 ActiveX 组件，以灵活的方式实现各种数据采集功能。

本书从工程应用的角度出发，通过 7 种典型的计算机控制系统（包括基于 PLC、单片机、PCI 数据采集卡、USB 数据采集板、CAN 数据采集模块、远程 I/O 模块、智能仪器），对工业控制系统中的 4 类典型应用，包括模拟量输入（AI）、模拟量输出（AO）、数字量输入（DI）和数字量输出（DO）的 Delphi 程序设计方法进行了详细的讲解。

淡化理论，建立控制系统整体概念，以工程实践为主，硬件系统设计采用“搭积木”方式，突出程序设计，重在功能实现，有较强的实用性和可操作性，这是本书的特色。

本书内容丰富，提供的实例均有具体的设计任务、详细的操作步骤和完整的程序代码，同时注重解决工程实际问题。本书可供各类自动化、计算机应用、机电一体化等专业的大学生和研究生学习计算机控制技术使用，也可供计算机控制系统研发的工程技术人员参考。

为方便读者学习，本书提供超值配套光盘，内容包括所有案例的源程序、软硬件资源、程序运行录屏和系统测试录像等。

本书由石河子大学朱东芹编写第 1、2 章，郑瑶编写第 3、4 章，刘恩博编写第 5 章，查志华编写第 6、7 章，李江全编写第 8 章及附录。全书由朱东芹担任主编，郑瑶、李江全担任副主编，由李江全教授统稿。参与编写、程序设计等工作的人员还有田敏、李宏伟、郑重、任玲、龚立娇、王洪坤、汤智辉、胡蓉、王平、邓红涛等老师。电子开发网、北京研华科技、西安达泰电子、石河子大学电气工程实验中心等单位或公司为本书的编写提供了宝贵的技术支持和帮助，借此机会向他们致以深深的谢意。

由于编者水平有限，书中难免存在不妥或错误之处，恳请广大读者批评指正。

作 者

2010 年 10 月

目 录

第 1 章 基于单片机开发板的控制应用	1
1.1 单片机概述	1
1.1.1 单片机的组成	1
1.1.2 常用的单片机系列	3
1.1.3 单片机的开发工具	4
1.1.4 单片机的特点及应用	5
1.2 串行通信组件 MSComm	6
1.2.1 MSComm 组件处理通信的方式	6
1.2.2 MSComm 组件的引用	7
1.2.3 MSComm 组件的常用属性	8
1.2.4 MSComm 组件的 OnComm 事件	12
1.2.5 MSComm 组件通信步骤	14
1.3 系统设计说明	14
1.3.1 设计任务	14
1.3.2 硬件系统	15
1.4 数据采集与控制程序设计	17
1.4.1 模拟量输入	17
1.4.2 模拟量输出	34
1.4.3 数字量输入	47
1.4.4 数字量输出	55
第 2 章 基于 PLC 的控制应用	61
2.1 PLC 概述	61
2.1.1 PLC 的构成	62
2.1.2 PLC 的技术特点	64
2.1.3 计算机与 PLC 的连接	64
2.2 串口总线概述	66
2.2.1 RS-232C 串口通信标准	66
2.2.2 串口通信线路连接	68
2.2.3 个人计算机中的串行端口	69
2.2.4 串口通信调试	70
2.3 系统设计说明	73
2.3.1 设计任务	73
2.3.2 硬件系统	74
2.4 数据采集与控制程序设计	81

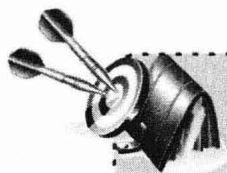
2.4.1	模拟量输入	81
2.4.2	模拟量输出	97
2.4.3	数字量输入	105
2.4.4	数字量输出	126
第 3 章	基于 PCI 数据采集卡的控制应用	147
3.1	数据采集卡概述	147
3.1.1	数据采集系统的含义与功能	147
3.1.2	数据采集系统组成与特点	149
3.1.3	数据采集卡的组成与功能	152
3.1.4	数据采集卡的类型与性能指标	153
3.1.5	数据采集卡的选择	156
3.2	Delphi 数据采集与控制的方式	157
3.2.1	使用 ActiveX 控件	158
3.2.2	使用动态链接库 (DLL)	158
3.3	系统设计说明	160
3.3.1	设计任务	160
3.3.2	硬件系统	161
3.4	数据采集与控制程序设计	167
3.4.1	模拟量输入	167
3.4.2	模拟量输出	177
3.4.3	数字量输入	186
3.4.4	数字量输出	191
第 4 章	基于 USB 数据采集板的控制应用	198
4.1	USB 总线概述	198
4.1.1	USB 的基本概念	198
4.1.2	USB 的总线结构	199
4.1.3	采用 USB 总线的数据采集系统	201
4.2	系统设计说明	203
4.2.1	设计任务	203
4.2.2	硬件系统	204
4.3	数据采集与控制应用程序设计	207
4.3.1	模拟量输入	207
4.3.2	模拟量输出	212
4.3.3	数字量输入	215
4.3.4	数字量输出	218
第 5 章	基于 CAN 总线模块的控制应用	223
5.1	CAN 总线概述	223

5.1.1 现场总线技术概述	223
5.1.2 CAN 总线控制技术	225
5.2 设计说明	228
5.2.1 设计任务	228
5.2.2 硬件系统	229
5.3 数据采集与控制应用程序设计	234
5.3.1 模拟量输入	234
5.3.2 模拟量输出	240
5.3.3 数字量输入	246
5.3.4 数字量输出	251
第 6 章 基于远程 I/O 模块的控制应用	260
6.1 集散控制系统概述	260
6.1.1 集散控制系统的体系结构	260
6.1.2 集散控制系统的特点	262
6.1.3 中小型 DCS 的基本结构	264
6.1.4 RS-485 串口通信标准	264
6.2 系统设计说明	266
6.2.1 设计任务	266
6.2.2 硬件系统	266
6.3 数据采集与控制应用程序设计	277
6.3.1 模拟量输入	277
6.3.2 模拟量输出	279
6.3.3 数字量输入	280
6.3.4 数字量输出	283
第 7 章 基于智能仪器的控制应用	288
7.1 智能仪器	288
7.1.1 智能仪器的组成	288
7.1.2 智能仪器的功能	289
7.1.3 智能仪器的特点	291
7.2 系统设计说明	291
7.2.1 设计任务	291
7.2.2 硬件系统	292
7.3 温度测量与控制程序设计	297
7.3.1 单台智能仪器温度测控	297
7.3.2 多台智能仪器温度测控	303
第 8 章 基于声卡的数据采集	310
8.1 声卡的基本常识	310



8.1.1 声卡的作用与特点	310
8.1.2 声卡的构造与设置	311
8.1.3 声卡的主要技术参数	313
8.1.4 声卡编程	314
8.2 基于声卡的数据采集程序设计	315
附录 A 控制系统的输入与输出	319
参考文献	323

第1章 基于单片机开发板的控制应用



目前,在许多单片机应用系统中,上、下位机分工明确,作为下位机核心器件的单片机往往只负责数据的采集和通信,而上位机(一般为个人计算机,简称PC)通常以基于图形界面的Windows系统为操作平台;为便于查询和保存数据,还需要数据库的支持,这种应用的核心是数据通信,它包括单片机和上位机之间、客户端和服务端之间及客户端和客户端之间的通信,而单片机和上位机之间数据通信则是整个系统的基础。

单片机和PC的通信是通过单片机的串口和PC串口之间的硬件连接实现的。

1.1 单片机概述

目前单片机以其独特的优点,在智能仪表、家用电器、工业控制、数据采集和网络通信等领域得到广泛的应用。各行各业的工程技术人员根据需要对单片机应用系统进行设计和开发,从而改变了传统控制系统的设计思想和设计方法。以前,必须由模拟电路或数字电路才能实现的大部分控制功能,现在由单片机通过软件方法便可实现,这使得控制系统的性能大大提高,应用领域更加广泛。

单片机以嵌入式应用为主,故又被称为嵌入式微控制器。国际上常把单片机称为微控制器(MCU),而国内则比较习惯称为“单片机”。

1.1.1 单片机的组成

单片机又称单片微控制器,它将一个计算机系统集成到一块芯片上,主要包括微处理器(CPU)、存储器(随机访问存储器RAM、只读存储器ROM)及各种输入/输出接口(包括定时器/计数器、并行I/O接口、串行口、A/D转换器及脉冲宽度调制PWM)等,如图1-1所示。

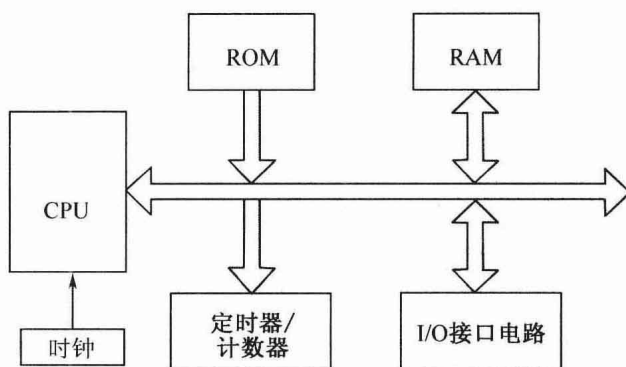


图 1-1 单片机组成框图

1. 程序存储器 (ROM)

ROM 用来存放用户程序，可分类为 EPROM、Mask ROM、OTP ROM 和 Flash ROM 等。

用 EPROM 型存储器编程（把程序代码通过一种算法写入程序存储器的操作）后，其内容可用紫外线擦除，用户可反复使用，因此特别适用于开发过程，但 EPROM 型单片机价格很高。

Mask ROM 型存储器的单片机价格最低，适用于大批量生产。由于 Mask ROM 型单片机的代码只能由生产厂商在制造芯片时写入，用户更改程序代码十分不便，因此在产品未成熟时选用此型单片机风险较高。

OTP ROM 型（一次可编程）单片机价格介于 EPROM 和 MaskROM 型单片机之间，它允许用户自己对其编程，但只能写入一次。

Flash ROM 型单片机可采用电擦除的方法修改其内容，允许用户使用编程工具或在系统中快速修改程序代码，并且可以反复使用。

2. 中央处理器 (CPU)

CPU 是单片机的核心单元，通常由算术逻辑运算部件 ALU 和控制部件构成。CPU 就像人的大脑一样，决定了单片机的运算能力和处理速度。

3. 随机存储器 (RAM)

RAM 用来存放程序运行时的工作变量和数据，由于 RAM 的制作工艺复杂，价格比 ROM 高得多，所以单片机内部的 RAM 非常宝贵，通常仅有几十到几百字节。RAM 的内容是易失性（也有称为易挥发性）的，掉电后会丢失。最近出现的 EEPROM 或 Flash ROM 型的数据存储器，方便用户存放不经常改变的数据及其他重要信息。单片机通常还有特殊寄存器和通用寄存器，也属于 RAM 空间，但它们在单片机中存取数据速度很快；特殊寄存器还有充分发挥单片机各种资源的功效，但这部分存储器占用存储空间更小。

4. 并行输入/输出 (I/O) 接口

并行输入、输出 (I/O) 接口通常为独立的双向 I/O 接口，任意接口既可以用作输入方式，又可以用作输出方式，通过软件编程设定。现代的单片机的 I/O 接口还有不同的功能，

有的内部具有上拉或下拉电阻,有的是漏极开路输出,有的能提供足够的电流直接驱动外部设备。I/O 接口是单片机的重要资源,也是衡量单片机功能的重要指标之一。

5. 串行输入/输出接口

串行输入/输出口用于单片机和串行设备或与其他单片机的通信。串行通信有同步和异步之分,这可以用硬件或通用串行收发器件实现。不同的单片机可能提供不同标准的串行通信接口,如 UART、SPI、I²C 和 MicroWire 等。

6. 定时器/计数器 (T/C)

定时器/计数器 (T/C) 用于单片机内部精确定时或对外部事件 (输入信号如脉冲) 进行计数,通常单片机内部有两个或两个以上的定时器/计数器。

7. 系统时钟

系统时钟通常需要外接石英晶体或其他振荡源提供时钟信号输入,也有的使用内部 RC 振荡器。系统时钟相当于 PC 微机中的主频。

以上只是单片机的基本构成,现代的单片机又加入了许多新的功能部件,如模拟/数字转换器 (A/D)、数字/模拟转换器 (D/A)、温度传感器、液晶 (LCD) 驱动电路、电压监控、看门狗 (WDT) 电路和低压检测 (LVD) 电路等。

1.1.2 常用的单片机系列

自 20 世纪 80 年代以来,单片机产品如雨后春笋般大量涌现。GI、Rochwell、Intel、Zilog、Motorola 和 NEC 等世界上几大计算机公司都纷纷推出自己的单片机系列。

虽然单片机的品种繁多,但在我国使用最多的还是 Intel 公司研发的 MCS-51 系列单片机。MCS-51 系列单片机是于 20 世纪 80 年代初,在 MCS-48 系列的基础上发展起来的,虽然仍然是 8 位的单片机,但它品种齐全、兼容性强,而且性能价格比高;同时,其软、硬件应用设计资料丰富,已为广大工程技术人员所熟悉,因此在我国得到了广泛的应用。

MCS 是 Intel 公司的注册商标。凡 Intel 公司生产的以 8051 为核心单元的其他派生单片机都可以称为 MCS-51 系列,有时简称为 51 系列。MCS-51 系列单片机包括 3 个基本型 8031、8051、8751 和对应的低功耗型 801231、80C51、87C51。

MCS-51 系列及 80C51 单片机有多个品种,它们的引脚及指令系统相互兼容,主要在内部结构上有些区别,最常用的 51 系列单片机是 8051 和 AT89C51 (如图 1-2 所示) 等。

AT89C51 具有片内 EEPROM,是真正的单片机,由于不需要外接 EPROM,所以 AT89C51 的应用非常普遍。8031、8051 片内没有 EPROM,但它的软硬件系统开发成熟,在市场上价格很低,所以应用也非常广泛。目前开发的 51 系列的产品大多是 8031、8051 和 AT89C51 等。

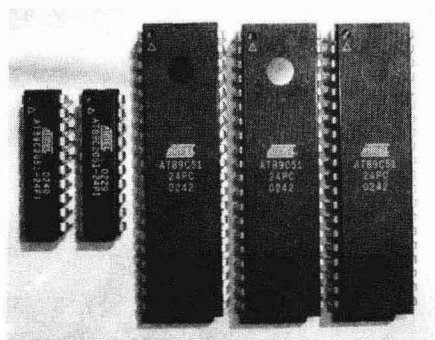


图 1-2 AT89C51 系列单片机产品示意图

宏晶公司生产的 STC89C51RC 单片机为低电压、高性能的 CMOS 8 位单片机，片内含 2KB 的可反复擦写的只读程序存储器（PEROM）和 128B 的随机存取数据存储器（RAM），工作电压为 2.7~6V，且含有两个 16 位的定时器，6 个内部中断源，可编程的串口 UART 和兼容标准 MCS-51 指令系统。该单片机片内置有通用 8 位中央处理器和 Flash 存储单元，封装只有 40 针，体积比较小，工作温度为 -40~85℃。

STC89C51RC 单片机可以利用 STC-ISP 软件方便的实现在线烧写程序。

1.1.3 单片机的开发工具

1. 仿真器

单片机的仿真器本身就是一个单片机系统，具有与所要开发的单片机应用系统相同的单片机芯片。

当一个单片机应用系统电路连接完毕，由于其自身无调试能力，无法检验好坏时，可以将系统中的单片机拔掉，插上在线仿真器提供的仿真头。仿真头是一个 40 脚插头，它是仿真器的单片机信号的延伸，即单片机应用系统与仿真器共用一块单片机芯片。当在开发工具上通过在线仿真器调试单片机应用系统时，就像使用应用系统中真实的单片机一样，这种替代被称为仿真。

在线仿真器是由一系列硬件构成的设备。开发工具中的在线仿真器能够仿真应用系统中的单片机，并能模拟应用系统中的 ROM、RAM 和 I/O 接口的功能，这使得在线仿真的应用系统的运行环境和脱机运行的环境完全一致，以实现单片机应用系统的一次性开发。

2. 编程语言

开发单片机的编程语言主要是汇编语言和 C 语言。

采用汇编语言编程必须对单片机的内部资源和外围电路非常熟悉，尤其是对指令系统的使用必须非常熟练，对程序开发者的要求比较高。用汇编语言开发软件，程序量通常比较大，方方面面需要考虑周全，一切问题都要由程序设计者安排，其实时性和可靠性完全取决于程序设计人员的水平。采用汇编语言编程主要适用于功能比较简单的中小型应用系统。

采用 C 语言编程时，只需对单片机的内部结构基本了解，对外围电路比较熟悉，但对指令系统则不必非常熟悉。用 C 语言开发软件相对轻松，很多细节问题无须考虑，编译软件会替设计者安排好。因此，C 语言在单片机软件开发中的应用越来越广，使用者越来越多。当开发环境为基于操作系统编程时，编程语言通常采用 C 语言。

单纯采用 C 语言编程也有不足之处。在一些对时序要求非常苛刻或对运行效率要求非常高的场合，只有汇编语言能够很好地胜任。因此，在很多情况下，采用 C 语言和汇编语言混合编程是最佳选择。

从编程难度来看，汇编语言比 C 语言要难得多，但作为一个立志从事单片机系统开发的科技人员，必须熟练掌握汇编语言程序设计方法。在熟练掌握汇编语言编程之后，学习 C 语言编程将是一件比较轻松的事情，并且能够将 C 语言和汇编语言非常恰当地融合在一起，同时以最短的时间和最低的代价，开发出高质量的软件。

当系统调试结束，确认软件无故障时，应把用户应用程序固化到 EPROM 中。EPROM 写入器就是完成这种任务的专用设备，也是单片机开发工具中的重要组成部分。



1.1.4 单片机的特点及应用

1. 单片机的特点

单片机具有如下特点:

(1) 集成度高。单片机把 CPU、RAM、ROM、I/O 接口及定时器/计数器都集成在一个芯片上,和常规的计算机系统相比,它具有体积小,集成度高的特点。如 MCS-51 系列单片机,具有 16 位的定时器/计数器和 4 个并行 I/O 接口,此外还提供有串行接口。

(2) 存储量大。采用 16 位地址总线的 8 位单片机可寻址外部 64KB 数据存储器 and 64KB 程序存储器。此外,大部分单片机还有片上 RAM 和内部 ROM,在大多数情况下,使用内部存储器就已足够,从而减少了器件的使用数量、降低了成本。

(3) 性能高、速度快。为了提高速度和执行效率,单片机使用 RISC 体系结构、并行流水线操作和 DSP 等设计技术,使指令运行速度大幅提高。一般单片机的时钟频率可以达到 12MHz。

(4) 抗干扰性高。单片机的各种功能部件都集成在一块芯片上,特别是存储器也集成在芯片内部;单片机布线短,大都在芯片内部传送数据,因此不易受到外部的干扰,增强了抗干扰能力,使系统运行更加可靠。

(5) 指令丰富。单片机一般都有传送指令,逻辑运算指令,转移指令,加、减运算指令,以及位操作指令。

(6) 实时控制能力强。实时控制又称过程控制,是指及时地检测设备、采集数据信息,并按最佳方案对设备进行自动调节和控制。单片机具有很强的逻辑操作、位处理和判断转移功能,运行速度快,特别适合于工业系统实时控制。

(7) 应用开发周期短。单片机结构简单,硬件组合、软件编程都很方便,又容易进行模拟试验,因此付诸实际应用快。

2. 单片机的应用

单片机由于具有可靠性高、集成度高、价格低廉和容易产品化等特点,在智能仪器仪表、工业实时控制、智能终端、通信设备、医疗器械、汽车电器和家用电器等领域都得到了广泛的应用。以下简单介绍一些典型的应用。

1) 家用电器领域

如洗衣机、空调器、汽车电子与保安系统、电视机、录像机、DVD 机、音响设备、电子秤、IC 卡、手机和 BP 机等,在这些设备中使用单片机机芯之后,其控制功能和性能大大提高,并实现了智能化、最优化控制。

2) 终端及外部设备控制

计算机网络终端设备,如银行终端、商业 POS (自动收款机)、复印机等,以及计算机外部设备,如打印机、绘图机、传真机、键盘和通信终端等。在这些设备中使用单片机,使其具有计算、存储、显示和输入等功能,因其有和计算机连接的接口,故而使计算机的能力及应用范围大大提高、更好地发挥计算机的性能。

3) 工业自动化领域

在工业自动化领域,单片机系统主要用来实现信号的检测、数据的采集及应用对象的

控制。这些系统中除一些小型工控机外，许多都是以单片机为核心的单机或多机网络系统。

单片机广泛地应用于各种实时控制系统中。例如，在工业测控、航空航天和尖端武器等实时控制系统中，都可以用单片机作为控制器。单片机的实时数据处理能力和控制功能，能使系统保持在最佳工作状态，提高系统的工作效率和产品质量。

4) 智能仪表

单片机具有体积小、功耗低、控制功能强、扩展灵活、微型化和使用方便等优点，广泛应用于仪器仪表中，结合不同类型的传感器，可实现诸如电压、功率、频率、湿度、温度、流量和压力等物理量的测量。例如，精密的测量设备（功率计、示波器和各种分析仪），采用单片机控制使得仪器仪表能够数字化、智能化和微型化，并且功能比起采用电子或数字电路更加强大，大大提高了其性能价格比。

通过采用单片机软件编程技术，使长期以来测量仪表中的误差修正、线性化处理、存储和数据处理等难题迎刃而解，提高了仪器的精度和可靠性，扩大了仪器的功能。

5) 机电一体化产品

机电一体化是机械工业发展的方向。机电一体化产品是指集机械技术、微电子技术、计算机技术和传感器技术于一体，具有智能化特征的机电产品，例如，微机控制的车床和钻床等。单片机作为产品中的控制器，能充分发挥它体积小、可靠性高、超功能强等优点，大大提高机器的自动化和智能化程度。可编程顺序控制器也是一个典型的机电控制器，核心常常就是由一个单片机构成的。

1.2 串行通信组件 MSComm

MSComm 组件全称为 Microsoft Communications Control，是 Microsoft 公司提供的串行通信编程的 ActiveX 组件，它既可以用来提供简单的串行端口通信功能，也可以用来创建全双工的、事件驱动的和高效实用的通信程序。

MSComm 组件在串口编程时非常方便，程序员不必花时间去了解较为复杂的 API 函数，而且在 Visual Basic、Visual C++ 和 Delphi 等语言中均可使用；使用它可以建立与串行端口的连接、通过串行端口连接到其他通信设备（如调制解调器）、发出命令、交换数据，以及监视和响应串行连接中发生的事件和错误；使用它可以进行诸如拨打电话、监视串行端口的输入数据乃至创建功能完备的终端程序等。

1.2.1 MSComm 组件处理通信的方式

MSComm 组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。

MSComm 组件提供下列两种处理通信的方式。

1. 事件驱动方式

该方式相当于一般程序设计中的中断方式。当串口发生事件或错误时，MSComm 组件会产生 OnComm 事件，用户程序可以捕获该事件进行相应处理，它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，在事件发生时程序会希望得到通知。例如，在

串口接收缓冲区中有一个字符到达或一个变化发生时，程序都可以利用 MSComm 组件的 OnComm 事件捕获并处理这些通信事件；OnComm 事件还可以检查和处理通信错误。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。

在程序设计中，可以在 OnComm 事件处理函数中加入自己的处理代码，一旦事件发生即可自动执行该段程序。这种方法的优点是程序响应及时、可靠性高。

2. 查询方式

在程序的每个关键功能之后，对用户程序中设计定时或不定时的查询，通过检查 CommEvent 属性的值来查询事件和错误，从而作出相应的处理。在进行简单应用程序设计时可采用这种方法。例如，如果写一个简单的电话拨号程序，则没有必要对每接收一个字符都产生事件，因为唯一等待接收的字符是调制解调器的“OK”响应。

1.2.2 MSComm 组件的引用

1. 组件安装

当开始一个项目的设计时，Delphi 的 ActiveX 组件面板中会有许多默认的组件让设计者选用，这些原本就有的组件是内置组件，提供了一些基本的系统设计组件给设计者；不过，功能比较特别的组件就不会出现在其中，而是用来设计通信功能的组件 MSComm 也不在其中。

由于 Delphi 的串行通信组件并不会主动出现在组件面板里，当需要 MSComm 组件时，便首先要把它添加到组件面板中。

在 Delphi 中安装 MSComm 组件，先打开 Delphi 集成开发环境，选择菜单“Component”中的“Import ActiveX Control”命令，在“Import ActiveX”选项卡内选择“Microsoft Comm Control 6.0 (Version 1.1)”项，如图 1-3 所示。

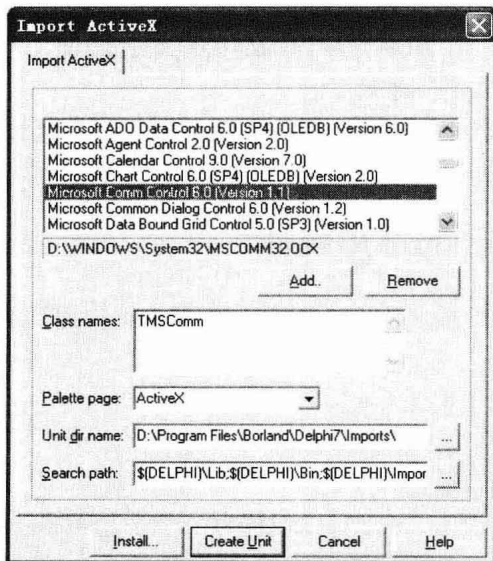


图 1-3 Import ActiveX 对话框

指定相关参数后，单击“Install”按钮，在弹出的“Install”对话框中，单击“OK”

按钮，此时 MSComm 组件就被添加到 ActiveX 组件面板中，其图标为一个电话的形状，如图 1-4 所示。

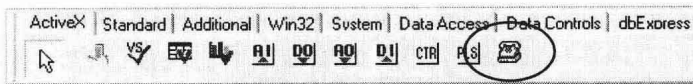


图 1-4 ActiveX 组件面板中的 MSComm 组件

如果图 1-3 列表中没有该组件，则单击“Add”按钮，在出现的对话框中指定 Mscomm32.ocx 文件的路径。如果打开的是以前的项目，项目中含有 MSComm 组件的引用记录，则项目会自动去搜寻 MSComm 组件，并将它载入，不需要以上的步骤。

ActiveX 组件面板中有了 MSComm 组件，就可以选择 MSComm 组件的图标后将其加到程序窗体上。

每个使用的 MSComm 组件对应着一个串行端口，如果应用程序需要访问多个串行端口，必须使用多个 MSComm 组件。

2. 组件操 4F5

在使用 Delphi 所提供的串行通信功能之前，必须对 Delphi 的 MSComm 组件作一个了解，以便可以将串行通信的概念套用上去。

Windows 采用了全新的对象化思想设计：把所有的程序都对象化。对象化之后，在采用 Delphi 设计串行通信的相关项目时，一样是遵循了如下 4 个主要步骤。

- (1) 对象：首先了解所要操作的对象是什么？
- (2) 属性：该对象所具备的特性有哪些？
- (3) 事件：该对象在系统执行的过程中会因其他对象而发生什么样的事情？
- (4) 方法：当该对象被引发了某个事件之后，程序应该采用的步骤是什么？

在通信前还需添加对象。首先我们要使用 MSComm 组件对外作串行通信，因此，在组件面板中选择了 MSComm 组件的图标后将其添加到程序窗体上，便可在窗体上安置了一个 MSComm，形同安装一个和串行端口沟通的管道在画面上。利用该组件 PC 就可以通过 Delphi 实现与串口设备的串口通信了。

接下来就是属性的设置。每一个组件的属性都相当多，通过属性值的设置，可以指定硬件以一定的方式工作。

1.2.3 MSComm 组件的常用属性

MSComm 组件的属性很多，这里介绍串口编程中经常用到的 14 个重要属性。

1. CommPort 属性

语法：MSComm1.CommPort[:=Value];

作用：设置或返回通信端口号。

设计时，CommPort 属性值 value 可以设置为 1~16 之间的任何整数（默认值为 1）来表示串口 COM1, COM2…。如果用 PortOpen 属性打开一个并不存在的端口，MSComm 组件会产生错误 68（设备无效）。

注意：必须在打开端口之前设置 CommPort 属性。



例如, `MSComm1.CommPort := 2; //COM2 上连接有一个调制解调器`

2. Input 属性

语法: `MSComm1.Input`

作用: 返回并删除接收缓冲区中的数据流。

`InputLen` 属性确定被 `Input` 属性读取的字符数。`InputMode` 属性确定用 `Input` 属性读取的数据类型。

例如, 如果希望从接收缓冲区获取数据, 并将其显示在一个文本框中, 可以使用代码如下:

```
TxtDisplay.Text := MSComm1.Input;
```

该属性在设计时无效, 在运行时为只读。

3. InputLen 属性

语法: `MSComm1.InputLen := value;`

作用: 设置并返回 `Input` 属性从接收缓冲区读取的字符数。

`Value` 是整型表达式, 说明 `Input` 属性从接收缓冲区中读取的字符数。

说明: `InputLen` 属性的默认值是 0。设置 `InputLen` 为 0 时, 使用 `Input` 将使 `MSComm` 组件读取接收缓冲区中全部的内容。若接收缓冲区中 `InputLen` 字符无效, `Input` 属性返回一个零长度字符串 ("")。

在使用 `Input` 前, 用户可以选择检查 `InBufferCount` 属性来确定缓冲区中是否已有需要数目的字符。该属性在从输出格式为定长数据的机器读取数据时非常有用。如果读取以定长数据块的形式格式化了的数据时, 则需要将该属性设置为合适的值。

例如, `MSComm1.InputLen:=10; //当程序执行该指令时, 只会读取 10 个字符`

4. InputMode 属性

语法: `MSComm1.InputMode:=Value;`

作用: 设置或返回接收数据的数据类型。

`InputMode` 属性的 `Value` 值可以设置为如下常数:

(1) 0——通过 `Input` 属性以文本方式取回传入的数据。

(2) 1——通过 `Input` 属性以二进制方式取回传入的数据。

例如, `MSComm1.InputMode:=1; //表示以二进制方式读取数据`

5. OutPut 属性

语法: `MSComm1.OutPut:=Value;`

作用: 向传输缓冲区写数据流。

`Output` 属性可以传输文本数据或二进制数据。用 `Output` 属性传输文本数据, 必须定义一个包含一个字符串的 `Variant`; 发送二进制数据, 必须传递一个包含字节数组的 `Variant` 到 `Output` 属性。

正常情况下, 如果发送一个 ANSI 字符串到应用程序, 可以以文本数据的形式发送; 如果发送包含嵌入控制字符或 Null 字符等数据, 要以二进制形式发送。