

田 敏 刘恩博 郑 瑶 李江全 编著

工业自动化控制系统 典型应用与编程



6种典型的计算机控制系统

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

3种编程语言

C++Builder、Visual C++、LabWindows/CVI

4类典型应用

模拟量输入(AI)、模拟量输出(AO)、数字量输入(DI)、数字量输出(DO)



中国电力出版社
www.cepp.com.cn



附光盘
CD-ROM

工业自动化控制系统 典型应用与编程

86 例

田 敏 刘恩博 郑 瑶 李江全 编著



中国电力出版社

www.cepp.com.cn

内 容 简 介

本书从工程应用的角度出发,通过6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),使用目前流行的3种不同类型的C/C++编程语言(包括C++Builder、Visual C++和LabWindows/CVI),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)的程序设计方法进行了详细的阐述。为方便读者学习,本书还提供了配套光盘,内容包括所有实例的源程序、软硬件资源等。

本书可作为高等院校自动化、计算机应用、机电一体化等专业的大学生以及相关工程技术人员学习计算机控制系统编程技术的参考用书。

图书在版编目(CIP)数据

工业自动化控制系统典型应用与编程 86 例 / 田敏等编
著. —北京:中国电力出版社, 2010.8

ISBN 978-7-5123-0621-9

I. ①工… II. ①田… III. ①自动控制—工程设计
IV. ①TP273

中国版本图书馆CIP数据核字(2010)第123216号

中国电力出版社出版、发行

(北京三里河路6号 100044 <http://www.cepp.com.cn>)

汇鑫印务有限公司印刷

各地新华书店经售

*

2011年1月第一版 2011年1月北京第一次印刷

787毫米×1092毫米 16开本 19.625印张, 478千字

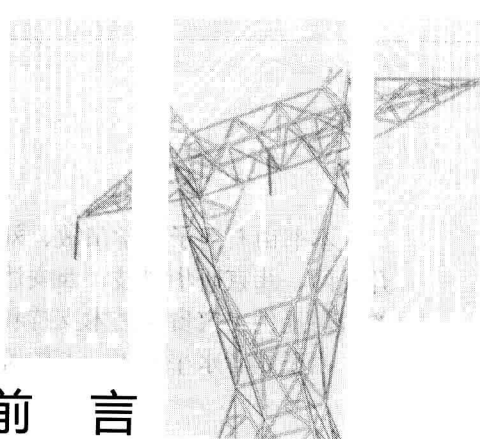
印数0001—3000册 定价39.00元(含1CD)

敬告读者

本书封面贴有防伪标签,加热后中心图案消失

本书如有印装质量问题,我社发行部负责退换

版 权 专 有 翻 印 必 究



在开发工业控制系统时，程序设计是很多技术人员要面临的问题。在高校、科研院所和工矿企业，有众多的技术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。

开发工业控制系统的编程语言很多，但应用最普遍、功能最强大的还是 C 语言，目前市面上流行的 C 语言是 C++ Builder、Visual C++ 和 LabWindows/CVI。

C++ Builder 是 Borland 公司推出的一种基于面向对象程序设计语言的可视化集成编程工具。它具有一个专业 C++ 开发环境所能提供的全部功能，实现了可视化的编程环境和功能强大的编程语言（C++）的完美结合。C++ Builder 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，程序设计者可以利用 C++ Builder 完成许多高级应用开发，扩展空间几乎是无限的。

Visual C++ 是微软公司推出的一种面向对象的可视化集成编程工具，使用 Visual C++ 可以胜任从最简单的用户界面程序到高级、复杂的 Windows 应用程序的编程工作。使用 Visual C++ 编制的程序具有代码尺寸小、运行速度高和移植能力强的特点。Visual C++ 成为目前最受欢迎的开发工具之一。在 Visual C++ 6.0 中既可以使用 MFC 完成大多数的工作，也可以调用 Windows API 函数完成更深层次的开发。

LabWindows/CVI 是 NI 公司开发的 Measurement Studio 软件组中的一员。它是 32 位的面向测控领域的虚拟仪器软件开发平台，可以在多操作系统下运行。LabWindows/CVI 是以 ANSI C 为核心的交互式虚拟仪器开发环境，它将功能强大的 C 语言与测控技术有机结合，具有灵活的交互式编程方法和丰富的库函数，为开发人员建立检测系统、自动测试环境、数据采集系统、过程监控系统等提供了理想的软件开发环境。

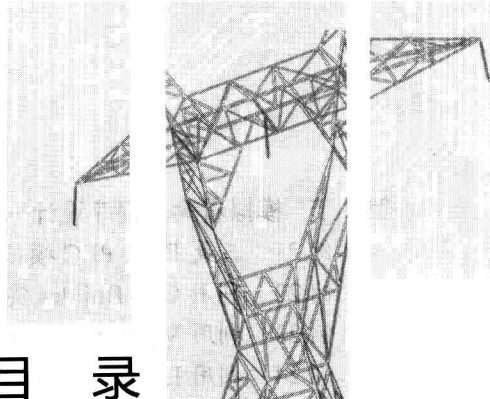
本书同时采用 C++ Builder 6.0、Visual C++ 6.0 和 LabWindows/CVI 9.0 3 种编程语言，通过 6 种典型的计算机控制系统（包括基于单片机、PLC、PCI 数据采集卡、USB 数据采集板、CAN 数据采集模块、远程 I/O 模块），详细介绍了 C/C++ 语言编写工业控制系统的程序设计方法，使读者对 C/C++ 语言开发工业控制系统有一个全面的认识。

为方便读者学习，本书提供配套光盘，内容包括所有实例的源程序、软硬件资源等。

本书由石河子大学田敏、刘恩博、郑瑶、李江全编著，由李江全教授策划并统稿。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，编者借此机会对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。

编 者



前言

第 1 章 利用单片机开发板实现自动化控制

第 1 节 设计说明	1
第 2 节 模拟量输入程序设计	4
例 1 利用 Keil C51 实现单片机模拟电压输入	4
例 2 利用汇编语言实现单片机模拟电压输入	10
例 3 利用 C++Builder 实现 PC 与单片机模拟电压输入	16
例 4 利用 Visual C++ 实现 PC 与单片机模拟电压输入	18
例 5 利用 LabWindows/CVI 实现 PC 与单片机模拟电压输入	21
第 3 节 模拟量输出程序设计	22
例 6 利用 Keil C51 实现单片机模拟电压输出	22
例 7 利用汇编语言实现单片机模拟电压输出	26
例 8 利用 C++ Builder 实现 PC 与单片机模拟电压输出	31
例 9 利用 Visual C++ 实现 PC 与单片机模拟电压输出	32
例 10 利用 LabWindows/CVI 实现 PC 与单片机模拟电压输出	33
第 4 节 数字量输入程序设计	35
例 11 利用 Keil C51 实现单片机数字量输入	35
例 12 利用汇编语言实现单片机数字量输入	37
例 13 利用 C++ Builder 实现 PC 与单片机数字量输入	39
例 14 利用 Visual C++ 实现 PC 与单片机数字量输入	41
例 15 利用 LabWindows/CVI 实现 PC 与单片机数字量输入	43
第 5 节 数字量输出程序设计	45
例 16 利用 Keil C51 实现单片机数字量输出	45
例 17 利用汇编语言实现单片机数字量输出	46
例 18 利用 C++ Builder 实现 PC 与单片机数字量输出	47
例 19 利用 Visual C++ 实现 PC 与单片机数字量输出	49
例 20 利用 LabWindows/CVI 实现 PC 与单片机数字量输出	51

第 2 章 利用 PLC 实现自动化控制

第 1 节 设计说明	54
------------	----

第2节 模拟量输入程序设计	62
例 21 三菱 FX _{2N} PLC 模拟电压输入程序	62
例 22 利用 C++ Builder 实现 PC 与 FX _{2N} PLC 模拟电压输入	63
例 23 利用 Visual C++ 实现 PC 与 FX _{2N} PLC 模拟电压输入	65
例 24 利用 LabWindows/CVI 实现 PC 与 FX _{2N} PLC 模拟电压输入	67
第3节 模拟量输出程序设计	69
例 25 三菱 FX _{2N} PLC 模拟电压输出程序	69
例 26 利用 C++ Builder 实现 PC 与 FX _{2N} PLC 模拟电压输出	70
例 27 利用 Visual C++ 实现 PC 与 FX _{2N} PLC 模拟电压输出	71
例 28 利用 LabWindows/CVI 实现 PC 与 FX _{2N} PLC 模拟电压输出	73
第4节 数字量输入程序设计	75
例 29 利用 C++ Builder 实现 PC 与 FX _{2N} PLC 数字量输入	75
例 30 利用 Visual C++ 实现 PC 与 FX _{2N} PLC 数字量输入	79
例 31 利用 LabWindows/CVI 实现 PC 与 FX _{2N} PLC 数字量输入	84
第5节 数字量输出程序设计	89
例 32 利用 C++ Builder 实现 PC 与 FX _{2N} PLC 数字量输出	89
例 33 利用 Visual C++ 实现 PC 与 FX _{2N} PLC 数字量输出	92
例 34 利用 LabWindows/CVI 实现 PC 与 FX _{2N} PLC 数字量输出	96
第6节 数字量输入与输出程序设计	101
例 35 西门子 S7-200 PLC 数字量输入与输出程序	101
例 36 利用 C++ Builder 实现 PC 与 S7-200 PLC 数字量输入与输出	103
例 37 利用 Visual C++ 实现 PC 与 S7-200 PLC 数字量输入与输出	108
例 38 利用 LabWindows/CVI 实现 PC 与 S7-200 PLC 数字量输入与输出	112

第3章 利用数据采集卡实现自动化控制

第1节 设计说明	116
第2节 模拟量输入程序设计	122
例 39 利用 C++ Builder 实现 PC 与 PCI 数据采集卡模拟电压输入	122
例 40 利用 Visual C++ 实现 PC 与 PCI 数据采集卡模拟电压输入	129
例 41 利用 LabWindows/CVI 实现 PC 与 PCI 数据采集卡模拟电压输入	137
第3节 模拟量输出程序设计	141
例 42 利用 C++ Builder 实现 PC 与 PCI 数据采集卡模拟电压输出	141
例 43 利用 Visual C++ 实现 PC 与 PCI 数据采集卡模拟电压输出	148
例 44 利用 LabWindows/CVI 实现 PC 与 PCI 数据采集卡模拟电压输出	156
第4节 数字量输入程序设计	160
例 45 利用 C++ Builder 实现 PC 与 PCI 数据采集卡数字量输入	160
例 46 利用 Visual C++ 实现 PC 与 PCI 数据采集卡数字量输入	164
例 47 利用 LabWindows/CVI 实现 PC 与 PCI 数据采集卡数字量输入	169

第 5 节 数字量输出程序设计	172
例 48 利用 C++ Builder 实现 PC 与 PCI 数据采集卡数字量输出	172
例 49 利用 Visual C++实现 PC 与 PCI 数据采集卡数字量输出	176
例 50 利用 LabWindows/CVI 实现 PC 与 PCI 数据采集卡数字量输出	181

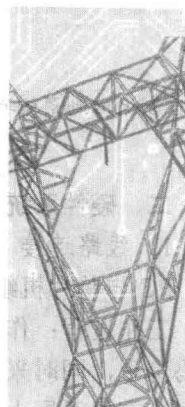
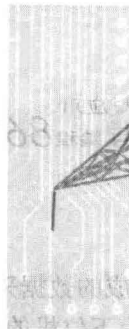
第 4 章 利用 USB 数据采集板实现自动化控制

第 1 节 设计说明	185
第 2 节 模拟量输入程序设计	189
例 51 利用 C++ Builder 实现 PC 与 USB 数据采集板模拟电压输入	189
例 52 利用 Visual C++实现 PC 与 USB 数据采集板模拟电压输入	191
例 53 利用 LabWindows/CVI 实现 PC 与 USB 数据采集板模拟电压输入	194
第 3 节 模拟量输出程序设计	197
例 54 利用 C++ Builder 实现 PC 与 USB 数据采集板模拟电压输出	197
例 55 利用 Visual C++实现 PC 与 USB 数据采集板模拟电压输出	198
例 56 利用 LabWindows/CVI 实现 PC 与 USB 数据采集板模拟电压输出	200
第 4 节 数字量输入程序设计	201
例 57 利用 C++ Builder 实现 PC 与 USB 数据采集板数字量输入	201
例 58 利用 Visual C++实现 PC 与 USB 数据采集板数字量输入	203
例 59 利用 LabWindows/CVI 实现 PC 与 USB 数据采集板数字量输入	207
第 5 节 数字量输出程序设计	209
例 60 利用 C++ Builder 实现 PC 与 USB 数据采集板数字量输出	209
例 61 利用 Visual C++实现 PC 与 USB 数据采集板数字量输出	215
例 62 利用 LabWindows/CVI 实现 PC 与 USB 数据采集板数字量输出	224

第 5 章 利用 CAN 数据采集模块实现数据采集与控制

第 1 节 设计说明	227
第 2 节 模拟量输入程序设计	231
例 63 利用 C++ Builder 实现 PC 与 CAN 数据采集模块模拟电压输入	231
例 64 利用 Visual C++实现 PC 与 CAN 数据采集模块模拟电压输入	235
例 65 利用 LabWindows/CVI 实现 PC 与 CAN 数据采集模块模拟电压输入	238
第 3 节 模拟量输出程序设计	241
例 66 利用 C++ Builder 实现 PC 与 CAN 数据采集模块模拟电压输出	241
例 67 利用 Visual C++实现 PC 与 CAN 数据采集模块模拟电压输出	244
例 68 利用 LabWindows/CVI 实现 PC 与 CAN 数据采集模块模拟电压输出	247
第 4 节 数字量输入程序设计	250
例 69 利用 C++ Builder 实现 PC 与 CAN 数据采集模块数字量输入	250
例 70 利用 Visual C++实现 PC 与 CAN 数据采集模块数字量输入	253
例 71 利用 LabWindows/CVI 实现 PC 与 CAN 数据采集模块数字量输入	256

第 5 节 数字量输出程序设计	259
例 72 利用 C++ Builder 实现 PC 与 CAN 数据采集模块数字量输出	259
例 73 利用 Visual C++实现 PC 与 CAN 数据采集模块数字量输出	263
例 74 利用 LabWindows/CVI 实现 PC 与 CAN 数据采集模块数字量输出	268
第 6 章 利用远程 I/O 模块实现自动化控制	
第 1 节 设计说明	271
第 2 节 模拟电压输入程序设计	282
例 75 利用 C++ Builder 实现 PC 与远程 I/O 模块模拟电压输入	282
例 76 利用 Visual C++实现 PC 与远程 I/O 模块模拟电压输入	284
例 77 利用 LabWindows/CVI 实现 PC 与远程 I/O 模块模拟电压输入	287
第 3 节 模拟电压输出程序设计	289
例 78 利用 C++ Builder 实现 PC 与远程 I/O 模块模拟电压输出	289
例 79 利用 Visual C++实现 PC 与远程 I/O 模块模拟电压输出	290
例 80 利用 LabWindows/CVI 实现 PC 与远程 I/O 模块模拟电压输出	291
第 4 节 数字量输入程序设计	292
例 81 利用 C++ Builder 实现 PC 与远程 I/O 模块数字量输入	292
例 82 利用 Visual C++实现 PC 与远程 I/O 模块数字量输入	295
例 83 利用 LabWindows/CVI 实现 PC 与远程 I/O 模块数字量输入	298
第 5 节 数字量输出程序设计	300
例 84 利用 C++ Builder 实现 PC 与远程 I/O 模块数字量输出	300
例 85 利用 Visual C++实现 PC 与远程 I/O 模块数字量输出	301
例 86 利用 LabWindows/CVI 实现 PC 与远程 I/O 模块数字量输出	303
参考文献	305



第 1 章

利用单片机开发板实现自动化控制

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、客户端和服务器之间以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 机串口之间的硬件连接实现的。

●——第 1 节 设计说明——●

一、设计任务

- 分别利用 Keil C51、汇编语言编写程序实现单片机数据采集与控制;
- 分别利用 C++ Builder、Visual C++和 LabWindows/CVI 编写程序实现 PC 与单片机自动化控制。

任务要求如下。

1. 模拟电压输入

单片机实验开发板接收变化的模拟电压(0~5V)并在数码管上显示(保留 1 位小数);PC 机接收单片机发送的电压值(16 进制,1 个字节),转换成 10 进制形式,以数字、曲线的方式显示。

2. 模拟电压输出

PC 机设置模拟电压值(0~10V),发送到单片机实验开发板,在数码管上显示(保留 1 位小数),并通过模拟电压输出端口输出。

3. 数字量输入

数字信号 0 或 1 送到单片机实验开发板数字量输入端口,并在数码管上显示;将信号发送到 PC 机显示。

4. 数字量输出

PC 机发出开关指令(0 或 1)传送给单片机实验开发板,驱动相应的继电器动作。

单片机与 PC 机通信,在程序设计上涉及两个部分的内容:① 单片机的 C51 数据采集和控制程序;② PC 机的串口通信程序和各种功能程序。

二、硬件系统

1. 线路连接

PC 与单片机组成的数据采集与控制系统结构如图 1-1 所示。

工作过程：作为下位机的单片机实时采集测量到的电压值，并将采集的电压数据显示在数码管上，同时采集的电压值通过串口传送到上位 PC 机。上位机收到下位机传送来的电压数据，在显示屏上显示。上位 PC 机设置电压值，通过串口发送到单片机系统，单片机数码管显示该电压，并通过模拟电压输出端口输出。上位 PC 机发出开关指令传送给单片机系统，驱动继电器动作。电气开关产生开关信号，发送到 PC 机显示。

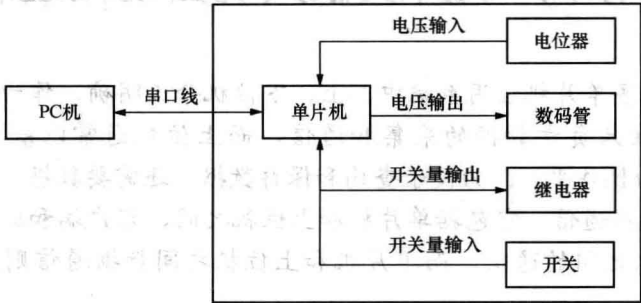


图 1-1 PC 与单片机组成的数据采集与控制系统结构

如图 1-2 所示，单片机实验开发板与 PC 机数据通信采用 3 线制，将单片机实验开发板 B 的串口与 PC 机串口的 3 个引脚（RXD、TXD、GND）分别连在一起，即将 PC 机和单片机的发送数据线 TXD 与接收数据 RXD 交叉连接，两者的地线 GND 直接相连。

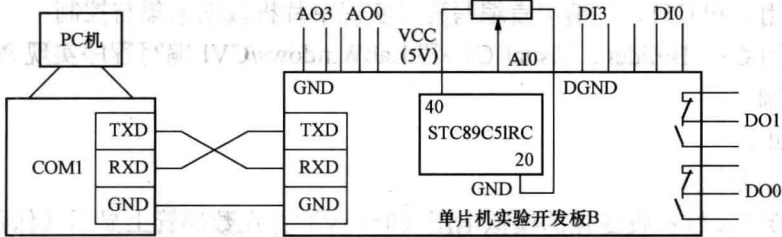


图 1-2 PC 机与单片机实验开发板 B 组成的数据采集与控制系统

由于单片机的 TTL 逻辑电平和 RS-232C 的电气特性完全不同，RS-232C 的逻辑 0 电平规定为 +3~+15V 之间，逻辑 1 电平为 -15~-3V 之间，因此在将 PC 机和单片机的 RXD 和 TXD 交叉连接时必须进行电平转换，单片机实验开发板 B 使用的是 MAX232 电平转换芯片。

模拟电压输入：直接采用单片机的 5V 电压输入（40 和 20 管脚）。将电位器两端与 STC89C51RC 单片机的 40 和 20 管脚相连，电位器的中间端点（输出电压 0~5V）与单片机实验开发板 B 的模拟量输入口 AI0 相连。

模拟电压输出：不需连线，使用万用表直接测量单片机实验开发板 B 的 AO0、AO1、AO2、AO3 端口与 GND 端口之间的输出电压。

数字量输入：使用杜邦线将单片机实验开发板 B 的 DI0、DI1、DI2、DI3 端口与 DGND 端口连接或断开即可。

数字量输出：不需连线，直接使用单片机实验开发板 B 的继电器和指示灯。

2. 单片机实验开发板 B 简介

单片机实验开发板 B 是电子开发网专为单片机初学者设计并开发的一种实验兼开发板，开发这个产品的目的就是为了帮助单片机初学者快速学会单片机技术。在自学单片机的过程中，通过做一系列的实验，从而比较容易地领会了单片机那些枯燥、难懂的专业术语，而且这款实验开发板弥补了市场上常见的单片机实验板的一些不足，有针对性地面向最终的实用控制功能，包括模拟量输入与输出接口、数字量输入与输出接口，增加了实用的继电器接口，可以使实验板能够直接用于控制各种负载，成为一个实用化的嵌入式控制系统。

图 1-3 所示为单片机实验开发板 B 的实物图。

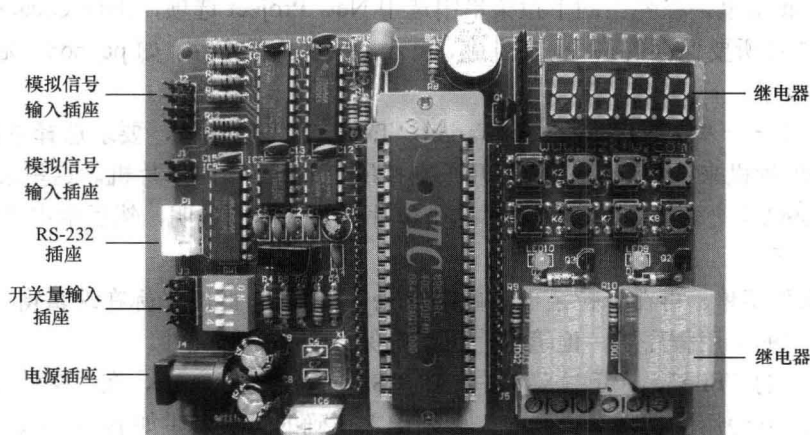


图 1-3 单片机实验开发板 B 实物图

单片机实验开发板 B 主要元件清单如下。

(1) 电源部分元件。9V 左右直流插头式小电源，带插头（空载 12V），电源插座一个，7805 稳压芯片一个，470 μ F/16V 电源滤波电容两个，0.1 μ F 独石电容两个，电源指示绿色 LED 一个，LED 限流电阻（560 Ω ）一个。

(2) 单片机最小系统部分的元件。STC89C51 单片机芯片一片，40 脚零拔插力 ZIF 插座一个，复位用电容 22 μ F/16V 电容一个，复位用电阻 1k Ω 一个，30pF 小电容 2 个，12MHz 晶振一个。

(3) 实验部分元件。8 个小红色长方形 LED，8 位 1k Ω 数码管限流排阻，共阴两位一体化的数码管一个，1 个 3V 电磁型蜂鸣器，1 个 8550 驱动三极管，3 个 1k Ω 三极管基极驱动电阻，微型轻触开关 4 个，4 位的红色拨码开关一个，2 个 12V JQC-3F 继电器（一组动合转动断），2 个 1N4007 防反峰二极管，2 个 8050 驱动三极管（e/b/c），2 个继电器状态指示红色发光二极管，MAX232 芯片一片，5.1k Ω 上拉电阻两个，10 μ F 电容 4 个，塑封一体化红外线接收头一个，AT24C02 存储器芯片一片，220 μ F 滤波电容一个，0.1 μ F 电容一个，32 个按键的红外遥控手柄一个，串口通信电缆一根，4.7k Ω 上拉电阻一个。

单片机实验开发板 B 可以做很多实验，如：模拟电压输入与输出，开关量输入与输出，红外线遥控器编码分析仪、通用频率计、温度测控等。

有关单片机实验开发板 B 的详细信息请查询电子开发网 <http://www.dzkfw.com/>。

第2节 模拟量输入程序设计

例1 利用Keil C51实现单片机模拟电压输入

Keil C51 软件是众多单片机应用开发的优秀软件之一, 它集编辑、编译、仿真于一体, 支持汇编、PLM 语言和 C 语言的程序设计, 界面友好, 易学易用。

启动 Keil C51, 几秒钟后出现编辑界面。

1. 建立一个新工程

单击 Project 菜单, 在弹出的下拉菜单中选中 New Project 选项, 出现 Create New Project 对话框。然后选择所要保存的路径、文件夹, 输入工程文件的名字, 如 pc_com (后缀名默认), 单击“保存”按钮。

这时会弹出一个“Select Device for Target ‘Target 1’”对话框, 要求选择单片机的型号, 根据所使用的单片机来选择, Keil C51 几乎支持所有的 51 核的单片机。这里选择 Atmel 的 89C51。选择 89C51 之后, 右边一栏是对这个单片机的基本的说明, 然后单击“确定”按钮。

2. 编写程序

单击“File”菜单, 再在下拉菜单中单击“New”选项。此时光标在编辑窗口里闪烁, 这时可以键入用户的应用程序了, 但笔者建议首先保存该空白的文件。

单击菜单上的“File”, 在下拉菜单中选中“Save As”选项单击, 在“文件名”栏右侧的编辑框中, 键入欲使用的文件名, 同时, 必须键入正确的扩展名, 如 pc_com.c, 然后, 单击“保存”按钮。

注意

如果用 C 语言编写程序, 则扩展名为 (.c); 如果用汇编语言编写程序, 则扩展名必须为 (.asm)。

回到编辑界面后, 单击“Target 1”前面的“+”号, 然后在“Source Group 1”上单击右键, 弹出菜单, 然后单击“Add File to Group ‘Source Group 1’”。

选中 pc_com.c, 然后单击“Add”按钮, 再单击“Close”按钮。此时注意到“Source Group 1”文件夹中多了一个子项“pc_com.c”。子项的多少与所增加的源程序的多少相同。

现在, 请输入 C 语言源程序。

在输入程序时, 读者已经看到了事先保存待编辑的文件的好处了吧, 即 Keil C51 会自动识别关键字, 并以不同的颜色提示用户加以注意, 这样会使用户少犯错误, 有利于提高编程效率。

3. 编译程序

单击“Project”菜单, 在下拉菜单中单击“Options for Target ‘Target 1’”选项, 出现对话框, 选择 Output 选项卡, 选中“Create HEX Files”项, 单击“确定”按钮。

再单击“Project”菜单, 在下拉菜单中单击“Built Target”选项 (或者使用快捷键 F7), 进行编译。若有错误会在 output 窗口提示, 可根据此提示, 找出错误并修改, 直至编译通过。Keil C51 编译界面如图 1-4 所示。



图 1-4 Keil C51 编译界面

至此，用 Keil C51 上做了一个完整工程，其中，生成一个编程器烧写文件：pc_com.hex。

4. 烧写程序

程序经过调试运行之后就可以将其烧写进单片机了。STC 系列单片机在线下载程序只需要用串口连接到单片机上就可以。用串口线连接 PC 与单片机实验板，将编写好的汇编程序用 Keilμ Vision3 编译生成 HEX 文件就可以实现程序的简便烧写。

到网站 <http://www.mcu-memory.com/> 下载 STC 单片机 ISP 下载编程软件。按照提示在计算机上运行该程序，其界面如图 1-5 所示。

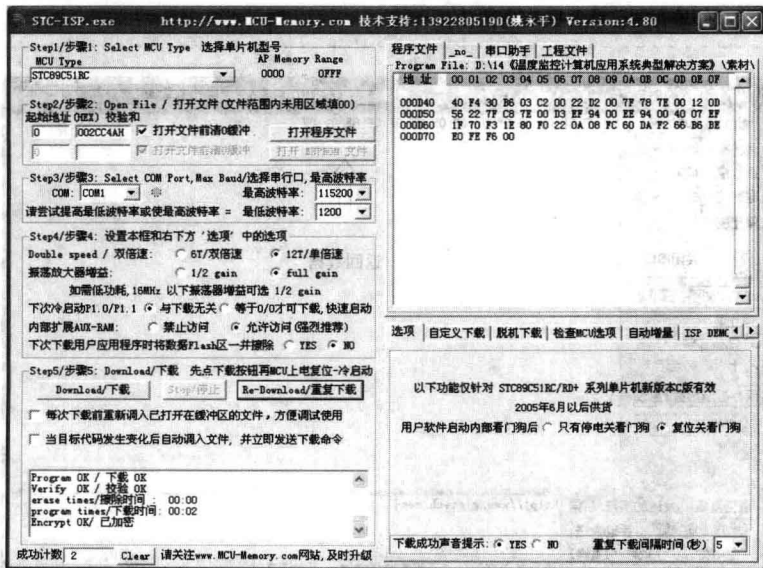


图 1-5 单片机烧写程序界面

烧写程序步骤如下：

1

连接单片机实验开发板与 PC 后，实验板电源先不要接通。

按下面步骤完成单片机程序的在线下载。

第一步：选择要下载程序的单片机型号。

第二步：打开编译完成要下载到单片机中扩展名为 HEX 的文件。

第三步：选择与实验板连接的串口。

第四步：选择合适的通信波特率（这步可以省略）。

第五步：将该选项选中（选中的目的是便于程序的调试，若下载的是调试通过的程序，此步可省略）。

第六步：单击该按钮。

第七步：接通实验板电源。

几秒钟后就可以将程序下载到单片机中，并运行了。

程序烧写进单片机之后，就可以给单片机实验开发板通电了，这时数码管上将会显示检测的电压值。

5. 串口通信调试

在进行串口开发之前，一般要进行串口调试，经常使用的工具是“串口调试助手”程序。它是一个适用于 Windows 平台的串口监视、串口调试程序。它可以在线设置各种通信速率、通信端口等参数，既可以发送字符串命令，也可以发送文件，可以设置自动发送/手动发送方式，可以十六进制显示接收到的数据等，从而提高串口开发效率。

打开“串口调试助手”程序（ScomAssistant.exe），其界面如图 1-6 所示。首先设置串口号 COM1、波特率 9600、校验位 NONE、数据位 8、停止位 1 等参数选择“十六进制显示”，打开串口。

注意

设置的参数必须与单片机设置的一致。

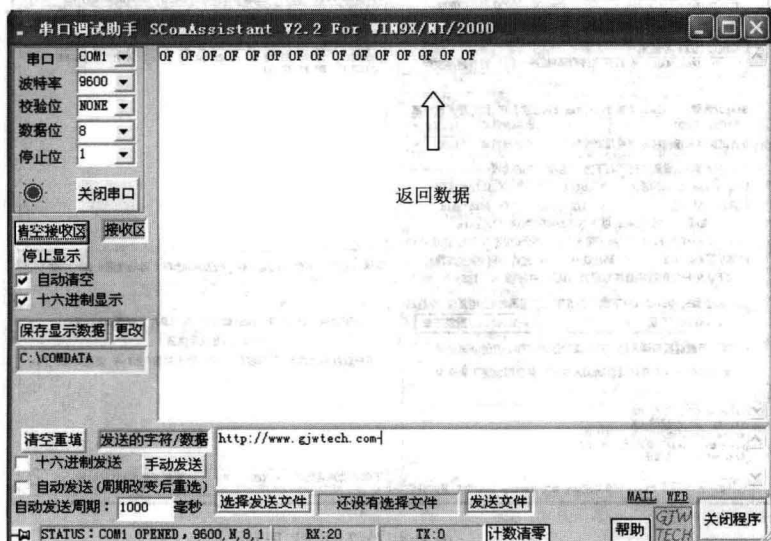


图 1-6 串口调试助手界面

如果 PC 机与单片机实验开发板串口连接正确, 则单片机连续向 PC 发送检测的电压值, 用 1 个字节的 16 进制数据表示, 如: 0F, 该数据串在返回信息框内显示。

将单片机返回数据转换为 10 进制, 并除以 10, 即可知当前电压测量值为 1.5V。

以下是完成单片机模拟电压输入的 C51 参考程序:

```

/*****
** 模拟电压输入, 显示屏显示 (保留 1 位小数), 并发送给 PC 机
** 晶振频率: 11.0592M
** 线路 -> 单片机实验开发板 B
*****/
#include <REG51.H>
#include <intrins.h>
/*****TLC0832 端口定义*****/
sbit ADC_CLK=P1^2;
sbit ADC_DO=P1^3;
sbit ADC_DI=P1^4;
sbit ADC_CS=P1^7;
/*****数码显示 键盘接口定义*****/
sbit PS0=P2^4; //数码管小数点后第一位
sbit PS1=P2^5; //数码管个位
sbit PS2=P2^6; //数码管十位
sbit PS3=P2^7; //数码管百位
sfr P_data=0x80; //P0 口为显示数据输出口
sbit P_K_L=P2^2; //键盘列
sbit JDQ1=P2^0; //继电器 1 控制
sbit JDQ2=P2^1; //继电器 2 控制
//字段转换表

unsigned char tab[]={0xfc, 0x60, 0xda, 0xf2, 0x66, 0xb6, 0xbe, 0xe0, 0xfe, 0xf6, 0xee,
0x3e, 0x9c, 0x7a, 0x9e, 0x8e};

unsigned char adc_change(unsigned char a); //操作 TLC0832
unsigned int htd(unsigned int a); //进制转换函数
void display(unsigned int a); //显示函数
void delay(unsigned int); //延时函数
void main(void)
{
    unsigned int a, temp;
    TMOD=0x20; //定时器 1--方式 2
    TL1=0xfd;
    TH1=0xfd; //11.0592MHZ 晶振, 波特率为 9600
    SCON=0x50; //方式 1
    TR1=1; //启动定时
    while(1)
    {
        temp=adc_change('0')*10*5/255;
        for(a=0; a<200; a++) //显示, 兼有延时的作用
            display(htd(temp));
        //SBUF=(unsigned char)(temp>>8); //将测量结果发送给 PC
        //while(TI!=1);
        //TI=0;
        SBUF=(unsigned char)temp;
        while(TI!=1);
    }
}

```

```

TI=0;
if(temp>45)
    JDQ1=0;          //继电器 1 动作
else
    JDQ1=1;          //继电器 1 复位
if(temp<5)
    JDQ2=0;          //继电器 2 动作
else
    JDQ2=1;          //继电器 1 复位
}
}
/*****数码管显示函数*****/
/*函数原型:void display(void)
/*函数功能:数码管显示
/*调用模块:delay()
/*****/
void display(unsigned int a)
{
    bit b=P_K_L;
    P_K_L=1;          //防止按键干扰显示
    P_data=tab[a&0x0f]; //显示小数点后第 1 位
    PS0=0;
    PS1=1;
    PS2=1;
    PS3=1;
    delay(200);
    P_data=tab[(a>>4)&0x0f]|0x01; //显示个位
    PS0=1;
    PS1=0;
    delay(200);
    //P_data=tab[(a>>8)&0x0f]; //显示十位
    PS1=1;
    //PS2=0;
    //delay(200);
    //P_data=tab[(a>>12)&0x0f]; //显示百位
    //PS2=1;
    //PS3=0;
    //delay(200);
    //PS3=1;
    P_K_L=b;          //恢复按键
    P_data=0xff;       //恢复数据口
}
/*****
; 函数名称: adc_change
; 功能描述: TI 公司 8 位 2 通 adc 芯片 TLC0832 的控制时序
; 形式参数: config(无符号整型变量)
; 返回参数: a_data
; 局部变量: m、n
*****/
unsigned char adc_change(unsigned char config) //操作 TLC0832
{

```