

单片机C语言程序 设计实训100例

—— 基于PIC + Proteus仿真

彭伟 编著

单片机 C 语言程序设计 实训 100 例

——基于 PIC+Proteus 仿真

彭 伟 编著

電子工業出版社

Publishing House of Electronics Industry

北京 · BEIJING

内 容 简 介

本书基于 MPLAB IDE+HI-TECH PICC/PICC18/MCC18 组合环境和 Proteus 硬件仿真平台,精心编写了 100 个 PIC 单片机 C 语言程序设计案例。全书提供了所有案例完整的 C 语言源程序,各案例设计了难易适中的实训目标。

全书基础设计类案例涵盖 PIC 单片机最基本的端口编程、定时/计数器应用、中断程序设计、A/D 转换、CCP 程序设计、EEPROM、Flash、USART 及看门狗程序设计等;硬件应用类案例涉及单片机存储器扩展、接口扩展、译码、编码、驱动、光电、机电、传感器、I²C 及 SPI 接口器件、MMC、ATA、遥控等器件;综合设计类案例涉及消费类电子产品、仪器仪表及智能控制设备等相关技术,部分案例涉及 IrDA/RS-485/Modbus/CAN/Ethernet 等技术应用。

本书可作为大专院校学生学习实践 PIC 单片机 C 语言程序设计技术的教材或参考书籍,也可作为电子工程技术人员或单片机技术爱好者的学习参考书。

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

单片机 C 语言程序设计实训 100 例:基于 PIC+Proteus 仿真 / 彭伟编著. —北京:电子工业出版社, 2011.11
ISBN 978-7-121-14833-0

I. ①单… II. ①彭… III. ①单片微型计算机—C 语言—程序设计 IV. ①TP368.1 ②TP312

中国版本图书馆 CIP 数据核字(2011)第 211771 号

责任编辑:周宏敏 文字编辑:韩 蕾

印 刷:北京天宇星印刷厂

装 订:三河市皇庄路通装订厂

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:39 字数:992 千字

印 次:2011 年 11 月第 1 次印刷

印 数:4 000 册 定价:88.00 元

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

目前,各高校电类专业都将 C 语言作为专业基础课程纳入教学计划,由于 C 语言功能强大,便于模块化开发,所带库函数非常丰富,编写的程序易于移植,此诸多优点使之成为单片机应用系统开发最快速高效的程序设计语言,仅具有 C 语言基础知识但不熟悉单片机指令系统的读者也能很快掌握单片机 C 语言程序设计技术,C 语言在单片机应用系统设计上的效率优势已经远远高于汇编、BASIC 等开发语言。

单片机 C 语言程序设计不同于通用的计算机应用程序设计,它必须针对具体的微控制器及外围电路来完成。为便于学习单片机应用程序设计和系统开发,很多公司推出了单片机实验箱、仿真器、编程器、下载器、开发板等,这些硬件设备可用于验证单片机程序,开发和调试单片机应用系统,但由于这些设备价格不菲,它们阻碍了普通读者对单片机技术的学习和研究。幸运的是,英国 Labcenter 公司推出了具有单片机系统仿真功能的 Proteus 软件,单片机系统开发通常是基于上位机加目标系统进行的,Proteus 的出现使得仅用一台 PC 在纯软件环境中完成系统设计与调试成为可能。目前 Proteus 支持 8051、AVR、PIC 等多种单片机,系统库中包含大量的模拟、数字、光电和机电类元器件,系统还提供了多种虚拟仪器,用 MPLAB IDE + HI-TECH PICC/PICC18/MCC18 开发的程序可以在用 Proteus 设计的仿真电路中调试和交互运行,这无疑为读者学习和提高 PIC 单片机 C 语言程序设计技术,为单片机应用系统高水平工程师的成长提供了理想平台。

为帮助读者快速提高 PIC 单片机 C 语言程序设计水平,本书基于 MPLAB+HI-TECH PICC/PICC18/MCC18 组合开发环境和 Labcenter 公司的 Proteus 仿真平台,精心编写了 100 个 PIC 单片机 C 语言程序设计案例,各案例同时给出了难易适中的实训目标。

本书前两章分别对 PIC 单片机 C 语言程序设计和 Proteus 操作基础做了概述,第 3 章基础程序部分给出的案例涵盖 PIC 单片机端口编程、中断程序设计、定时/计数器应用、模拟比较器程序设计、A/D 转换、EEPROM、USART 及看门狗程序设计等,各案例分别对相关知识和技术要点做了阐述与分析,源程序中也给出了丰富的注释信息。第 4 章硬件应用部分则针对 PIC 单片机的存储器扩展、接口扩展、译码、编码、驱动、光电、机电、传感器、I²C 及 SPI 接口器件、MMC、ATA、遥控等器件给出了数十个案例,对案例中涉及的硬件技术资料亦进行了有针对性的分析,以便于读者快速理解相关代码的编写原理。第 5 章的案例综合应用了单片机内部硬件资源和外部扩展资源,通过对这些案例的分析研究与调试运行及对设计实践要求的独立完成,读者用 C 语言开发 PIC 单片机应用系统的能力会得到大幅提升。

本书在非常繁忙的工作之中编写,是单片机 C 语言程序设计仿真系列 8051 版与 AVR 版之后的第三册,为使本书能早日与读者见面,笔者坚持挤出时间不懈耕耘。本书从选题、撰稿到出版的全过程中,学院领导、学院教务处、科研处及高教研究所对本选题始终给予大力支持,并提供项目资助,在此一并对学院和部门领导的关心与支持表示由衷感谢!

由于编者水平有限,加之时间仓促,书中错漏之处在所难免,在此真诚欢迎读者对本书多多提出宝贵意见,笔者邮箱是:pw95aaa@foxmail.com。

至此,本套单片机 C 语言程序设计仿真系列的 8051 版、AVR 版、PIC 版已经编写完成。三册图书的相关案例包可到出版社网站或者 <http://www.ieee1000.com> 免费下载。

应电子工业出版社要求，全新改版的 8051 分册第二版已经开始撰写，笔者将继续努力争取使之早日出炉，以飨读者。另外，已纳入笔者写作计划的还有《计算机高级接口实用开发技术》及《实用数据结构与算法设计（C#版）》，希望这两册图书也能得到读者的关注！

在这里，我还要郑重地感谢恩师陈文成先生，他的数学思想一直在深刻影响着我从事专业技术教学与研究工作的。我还要深深感谢父母，他们虽早已过花甲之年，但仍坚持劳作不息。我要特别感谢我的妻子魏莉萍女士，是她一直以来的鼓励和期望，使我能坚持本系列书籍的写作。她默默无闻地承担了大量家庭事务，使我能够安心研究、静心思索，她所做的许多稿件审校工作也为我节省了大量时间。我还要谢谢聪明可爱的乖女儿，她那天真爽朗的笑声总能驱走我的倦怠，愿她的脑海里总能充满幻想，并能实现所有的美丽幻想。

最后要感谢美国微芯公司的 Shaw Lam 与 Jacqueline Eichman 对本书撰写提供的重要支持与帮助，感谢电子工业出版社的曲昕编辑对本书撰写与出版所做的大量细致而周密的工作，谢谢！

谨以此书献给我所有的同行与朋友们，所有忠实的读者们，是你们的期待使我坚持完成了单片机 C 语言程序设计仿真实训系列书籍的写作，谢谢你们！

彭 伟

2011 年 7 月于武昌

目 录

第 1 章 PIC 单片机 C 语言程序设计概述	1
1.1 PIC 单片机简介	2
1.2 MPLAB+C 语言程序开发环境安装及应用	6
1.3 PICC/PICC18/MCC18 程序设计基础	8
1.4 PIC 单片机内存结构	17
1.5 PIC 单片机配置位	19
1.6 基本的 I/O 端口编程	21
1.7 中断服务程序设计	23
1.8 PIC 单片机外设相关寄存器	24
1.9 C 语言程序设计在 PIC 单片机应用系统开发中的优势	35
第 2 章 PROTEUS 操作基础	36
2.1 PROTEUS 操作界面简介	36
2.2 仿真电路原理图设计	37
2.3 元件选择	40
2.4 仿真运行	44
2.5 MPLAB IDE 与 PROTEUS 的联合调试	44
2.6 PROTEUS 在 PIC 单片机应用系统开发中的优势	47
第 3 章 基础程序设计	48
3.1 闪烁的 LED	48
3.2 用双重循环控制 LED 左右来回滚动显示	50
3.3 多花样流水灯	51
3.4 LED 模拟交通灯	53
3.5 单只数码管循环显示 0~9	55
3.6 4 只数码管滚动显示单个数字	57
3.7 8 只数码管扫描显示多个不同字符	59
3.8 K1~K5 控制两位数码管的开关、加减与清零操作	61
3.9 数码管显示 4×4 键盘矩阵按键	63
3.10 数码管显示拨码开关编码	66
3.11 继电器及双向可控硅控制照明设备	68
3.12 INT 中断计数	70
3.13 RB 端口电平变化中断控制两位数码管开关与加减显示	73
3.14 TIMER0 控制单只 LED 闪烁	76
3.15 TIMER0 控制流水灯	80
3.16 TIMER0 控制数码管扫描显示	81
3.17 TIMER1 控制交通指示灯	84
3.18 TIMER1 与 TIMER2 控制十字路口秒计时显示屏	88
3.19 用工作于同步计数方式的 TMR1 实现按键或脉冲计数	93

3.20	用定时器设计的门铃	95
3.21	报警器与旋转灯	98
3.22	用工作于捕获方式的 CCP1 设计的频率计	101
3.23	用工作于比较模式的 CCP1 控制音阶播放	105
3.24	CCP1 PWM 模式应用	108
3.25	模拟比较器测试	113
3.26	数码管显示两路 A/D 转换结果	117
3.27	EEPROM 读写与数码管显示	121
3.28	睡眠模式及看门狗应用测试	125
3.29	单片机与 PC 双向串口通信仿真	130
3.30	PIC 单片机并行从动端口 PSP 读写测试	139
第 4 章	硬件应用	146
4.1	74HC138 与 74HC154 译码器应用	146
4.2	74HC595 串入并出芯片应用	149
4.3	用 74HC164 驱动多只数码管显示	152
4.4	数码管 BCD 解码驱动器 7447 与 4511 应用	155
4.5	8×8LED 点阵屏显示数字	157
4.6	8 位数码管段位复用串行驱动芯片 MAX6951 应用	160
4.7	串行共阴显示驱动器 MAX7219 与 7221 应用	167
4.8	14 段与 16 段数码管串行驱动显示	171
4.9	16 键解码芯片 74C922 应用	175
4.10	1602LCD 字符液晶测试程序	178
4.11	1602 液晶显示 DS1302 实时时钟	183
4.12	1602 液晶工作于 4 位模式实时显示当前时间	189
4.13	带 RAM 内存的实时时钟与日历芯片 PCF8583 应用	194
4.14	2×20 串行字符液晶演示	201
4.15	LGM12864 液晶显示程序	204
4.16	PG160128A 液晶图文演示	211
4.17	TG126410 液晶串行模式显示	227
4.18	HDG12864 系列液晶演示	236
4.19	Nokia7110 液晶菜单控制程序	243
4.20	8 通道模拟开关 74HC4051 应用测试	252
4.21	用带 I ² C 接口的 MCP23016 扩展 16 位通用 I/O 端口	254
4.22	用带 SPI 接口的 MCP23S17 扩展 16 位通用 I/O 端口	261
4.23	用 I ² C 接口控制 MAX6953 驱动 4 片 5×7 点阵显示器	266
4.24	用 I ² C 接口控制 MAX6955 驱动 16 段数码管显示	272
4.25	用带 SPI 接口的数/模转换器 MCP4921 生成正弦波形	277
4.26	用带 SPI 接口的数/模转换器 MAX515 控制 LED 亮度循环变化	283
4.27	正反转可控的直流电机	286
4.28	PWM 控制 MOSFET 搭建的 H 桥电路驱动直流电机运行	288
4.29	正反转可控的步进电机	293
4.30	用 L297+L298 控制与驱动步进电机	296
4.31	PC 通过 RS-485 器件 MAX487 远程控制单片机	304

4.32	I ² C 接口 DS1621 温度传感器测试	308
4.33	SPI 接口温度传感器 TC72 应用测试	314
4.34	温度传感器 LM35 全量程应用测试	320
4.35	K 型热电偶温度计	324
4.36	用铂电阻温度传感器 PT100 设计的测温系统	329
4.37	DS18B20 温度传感器测试	337
4.38	SHT75 温湿度传感器测试	347
4.39	1-Wire 式可寻址开关 DS2405 应用测试	353
4.40	光敏电阻应用测试	359
4.41	MPX4250 压力传感器测试	361
4.42	用 I ² C 接口读写存储器 AT24C04	364
4.43	用 SPI 接口读写 AT25F1024	372
4.44	PIC18 I ² C 接口存储器及 USART 接口测试程序	380
4.45	PIC18 SPI 接口存储器测试程序	384
4.46	PIC18 定时器及 A/D 转换测试	389
4.47	用 PIC18 控制 Microwire 接口继电器驱动器 MAX4820	392
4.48	MMC 存储卡测试	395
4.49	ATA 硬盘数据访问	404
4.50	微芯 VLS5573 液晶显示屏驱动器演示	412
第 5 章	综合设计	425
5.1	用 DS1302/DS18B20+MAX6951 设计的多功能电子日历牌	425
5.2	用 PCF8583 设计高仿真数码管电子钟	433
5.3	用 4×20LCD 与 DS18B20 设计的单总线多点温度监测系统	439
5.4	用内置 EEPROM 与 1602 液晶设计的加密电子密码锁	453
5.5	用 PIC 单片机与 1601LCD 设计的计算器	459
5.6	电子秤仿真设计	468
5.7	数码管显示的 GP2D12 仿真测距报警器	473
5.8	GPS 全球定位系统仿真	479
5.9	能接收串口信息的带中英文硬字库的 80×16 点阵显示屏	485
5.10	用 M145026 与 M145027 设计的无线收发系统	494
5.11	红外遥控收发仿真	499
5.12	交流电压检测与数字显示仿真	506
5.13	带位置感应器的直流无刷电机 PMW 控制仿真	511
5.14	3 端可调正稳压器 LM317 应用测试	516
5.15	模拟射击训练游戏	520
5.16	带触摸屏的国际象棋游戏仿真	531
5.17	温室监控系统仿真	543
5.18	PIC 单片机 MODBUS 总线通信仿真	549
5.19	PIC 单片机内置 CAN 总线通信仿真	563
5.20	基于 PIC18+Microchip TCP/IP 协议栈的 HTTP 服务器应用	585
参考文献		613

第 1 章 PIC 单片机 C 语言程序设计概述

美国微芯科技公司 (Microchip Technology Inc) 是从事单片机及半导体器件研发生产的全球知名企业, 所开发的 CMOS 工艺 PIC (Peripheral Interface Controller, 外设接口控制器) 系列 8 位单片机, 采用精简指令结构 (RISC), 指令数量少, 其 8 位机指令集只有 35 条, 系统结构简洁, 非常容易学习与应用开发。PIC 单片机硬件采用哈佛 (Harvard) 总线结构, 即程序存储器与数据存储器地址完全分开, 指令总线与数据总线完全分开, 其高速度、低电压、低功耗、大电流 LCD 驱动能力和低价位的 OTP 技术都体现出现代单片机产业的新趋势。

PIC 系列单片机广泛应用于计算机外设、工业控制、通信控制、智能仪器、汽车电子、金融电子及消费类家电控制、玩具设计等各个领域, PIC 单片机已经成为世界上最有影响力的嵌入式微控制器之一。2003~2010 年, 其 8 位机的出货量一直居世界第一位, PIC 单片机的可靠性也在业界有口皆碑。另外, PIC 单片机众多的型号可以很好的适应不同的应用需求。虽然 PIC 单片机型号众多, 但彼此之间的系统移植却非常容易, 产品升级换代变得非常方便。

PIC 的 8 位机有基本级、中档级和高档级 3 个系列, 其内部指令宽度分别为 12、14、16 位, 下面对这三个系列的单片机作简要介绍。

(1) 基本级系列

该系列的低档单片机价位低, 适应于控制任务相对简单且对价格敏感的低端消费类产品。例如: PIC12C5xx、PIC16C5x 等。

(2) 中档级系列

该系列产品型号是最丰富的, 它在基本级产品上进行了改进, 并保持了很高的兼容性。其外部结构也是多样的, 具有从 8 引脚到 68 引脚的各种封装, 适用于各种低、中、高档电子产品的设计。例如: PIC16F7x、PIC16F87xA。

(3) 高档级系列

这个系列的特点是速度快, 适用于高速数字运算场合, 可取代某些 DSP 产品。此外, 这个系列单片机还具备丰富的 I/O 控制功能, 可外接扩展 EPROM 和 RAM, 使其成为 8 位单片机性能最高的供选机型。这个系列型号名称以 PIC18 开头。

PIC 的 8 位单片机支持众多外设及外设接口, 包括 TIMER、CCP、ADC、I²C、SPI、PSP、USB、CAN、LIN 等。

本书案例涉及 PIC12/16/18 系列单片机, 其中以中档的 PIC16F877A 单片机和高档的 PIC18F452 为主, 在开发 PIC 单片机 C 语言程序时, 使用的是微芯公司免费推出 MPLAB 集成开发环境, 综合了编辑器、项目管理器和设计平台, 适用于选择 PICmicro 系列单片机进行嵌入式系统开发。

全书在使用 C 语言程序开发设计各类案例时, 所使用的编译器有第三方的 HI-TECH PICC/PICC18 及微芯公司推出的 MCC18。使用微芯公司提供的集成开发环境 MPLAB IDE 及 HI-TECH PICC/PICC18 或 MCC18 编译器组合搭建的 PIC 单片机 C 语言程序开发平台, 可大大降低开发成本, 缩短开发周期, 大幅提高开发效率, 程序可读性好且易于移植。

本书的编写, 旨在进一步提高读者的 PIC 单片机 C 语言程序开发能力。全书的 100 个案例全部用 PICC、PICC18 或 MCC18 编译并调试通过, 除与 CAN BUS 有关的案例需要实物电路支持以外, 其余各案例都给出了完整的经过测试的 Proteus 仿真电路。

阅读使用本书之前要求已经学习了基本的 PIC 单片机 C 语言程序设计技术, 本章仅介绍使

用 C 语言设计 PIC 单片机应用系统必须参考和重点掌握的技术内容, 这些内容会对阅读、调试、研究本书案例及进行设计实践提供重要参考。

1.1 PIC 单片机简介

图 1-1 和图 1-2 分别给出了全书出现最多的 PIC16F877A 单片机和 PIC18F452 单片机的不同封装形式及引脚分布。

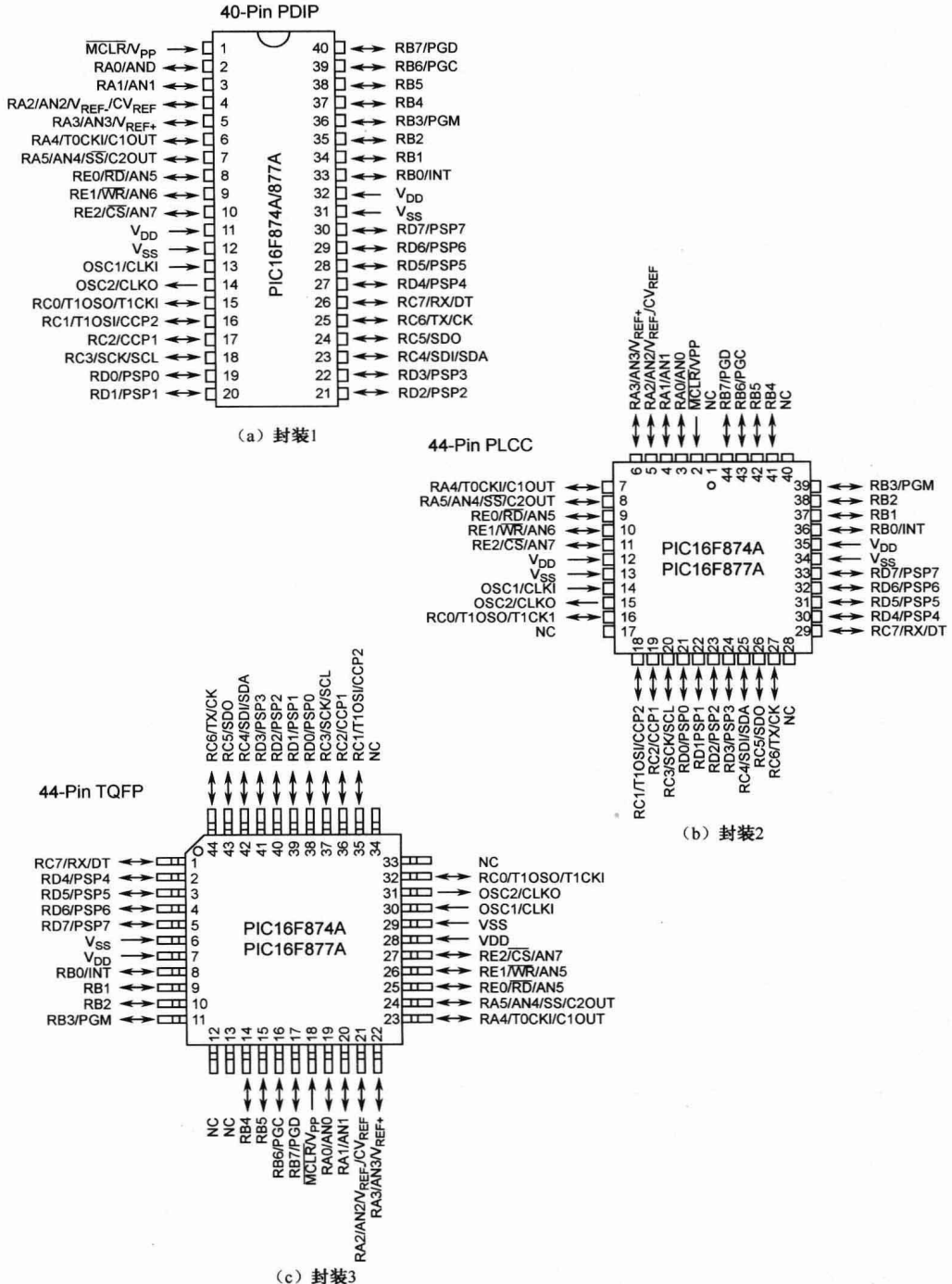


图 1-1 PIC16F877A 单片机三种不同封装 (QFN 封装未列出)

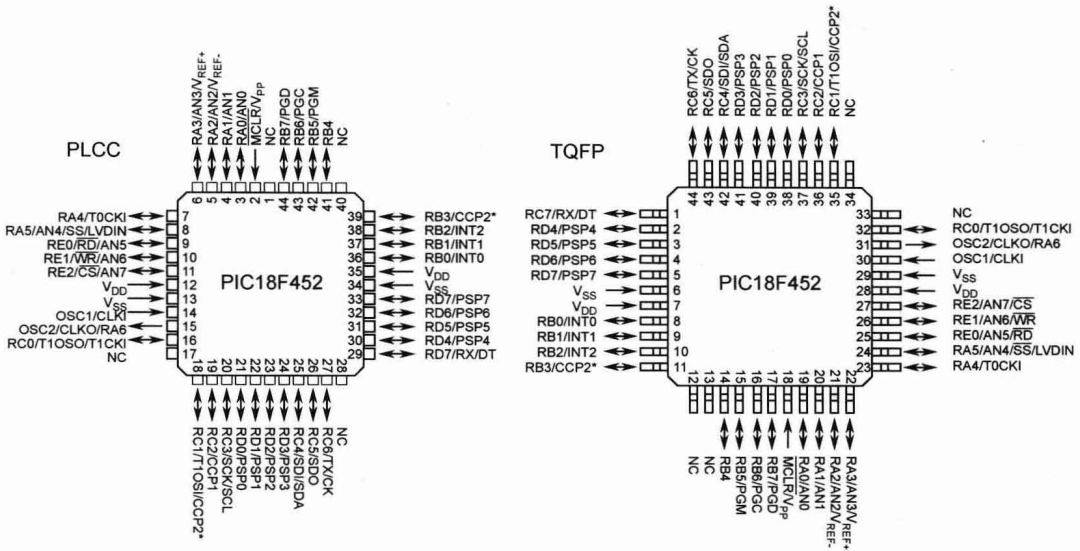


图 1-2 PIC18F452 单片机两种不同封装

下面介绍这两种单片机的主要性能与外设特征。

一、PIC16F877A 单片机主要性能与外设特征简介

(1) 主要性能

“哈佛结构”，功能强大的精简指令（RISC）单片机内核

- 35 条单字节指令
- DC-20 MHz 时钟输入，DC-200 ns 指令周期
- 可选择的振荡器
- 8 级深度硬件中断堆栈，用于保护和恢复程序计数器
- 共有 14 个中断源
- 直接、间接和相对寻址方式
- 8K Flash 程序字，368 字节数据 RAM 存储器，256 字节 EEPROM 数据存储器
- 处理器可读/写程序存储器（Flash）
- 全静态设计
- 具有上电复位功能 POR（Power-on Reset）
- 具有电源上升定时器 PWRT（Power-up Timer）和振荡器起振定时器 OST（Oscillator Start-up Timer）
- 具有片内看门狗定时器 WDT（Watchdog Timer）
- 具有片内可靠运行的 RC 振荡器
- 两线式串行编程能力（ICSP, In-Circuit Serial Programming）
- 具有单电源 5 V 在线串行编程能力
- 可经两个引脚进行在线调试
- 可编程的代码保护，防止程序泄密
- 运行电压范围 2.0~5.5 V
- 高吸入/拉出电流 25 mA
- 低功耗睡眠方式

- 低功耗，具有商用、工业用温度范围

(2) 外设特征

- **TIMER0**: 带有 8 位预分频器的 8 位定时器/计数器
- **TIMER1**: 带有预分频器的 16 位定时器/计数器，使用外部晶体振荡时钟时，在 SLEEP 期间仍能工作
- **TIMER2**: 带有 8 位周期寄存器、预分频器和后分频器的 8 位定时/计数器
- 2 个捕捉器（16 位，最大分辨率为 12.5 ns），比较器（16 位，最大分辨率为 200 ns），PWM 模块（最大分辨率是 10 位）
- 10 位 8 通道模/数转换器
- 模拟电压比较器
- 带有 SPI（主模式）和 I²C（主/从）模式的 MSSP
- 带有 9 位地址探测的通用同步/异步接收/发送器（USART，也称 SCI）
- 带有 RD/WR/CS（仅 40/44 引脚）控制 8 位字宽的并行从动端口（PSP）
- 带有欠压复位（BOR）电路

二、PIC18F452 单片机的主要性能与外设特征简介

(1) 主要性能

- “哈佛结构”，功能强大的精简指令（RISC）单片机内核
- 75 条单字节指令
- DC-40 MHz 时钟输入，最高执行速度达 10 MIPS
- 可选择的振荡器
- 32K Flash 程序存储器，1 536 字节数据 RAM 存储器，256 字节 EEPROM 数据存储器
- 多达 20 种中断源，具有两级中断优先级和中断嵌套
- 31 级硬件堆栈
- 立即寻址、直接寻址、间接寻址、寄存器间接寻址
- 内置 8×8 单周期硬件乘法器
- 扩展的看门狗定时器，可编程周期从 4 ms~131 s
- 多达 9 种复位方式：POR、BOR、RESET 指令、堆栈满、堆栈下溢、PWRT、OST、MCLR、WDT
- 多达 10 种振荡器方式选择：LP、XT、HS、HSPLL、RC、RCIO、INTIO1、INTIO2、EC、RCIO
- 在线串行编程（ICSP）与在线串行调试（ICD）
- 宽工作电压：2.0~5.5 V
- I/O 端口驱动能力强，驱动电流达 25 mA
- 具有商用、工业用温度范围
- 超低功耗设计：在空闲模式下，电流降至 5.8 μA，在休眠模式下，电流降至 0.1 μA。

(2) 外设特征

- **TIMER0**: 带有 8 位预分频器的 8/16 位定时器/计数器
- **TIMER1**: 带有预分频器的 16 位定时器/计数器，可在休眠模式下通过外部时钟计数
- **TIMER2**: 带有 8 位周期寄存器的 8 位定时/计数器（作为 PWM 的时基）
- 2 个 CCP 模块（捕捉/比较/脉宽调制）
- 8 路 10 位精度的 A/D 转换器

- MSSP 接口有 SPI、I²C 两种
- 带有 SPI（主模式）和 I²C（主/从）模式的 SSP
- 比较器参考电压模块
- 可寻址的 USART
- 并行从动端口（PSP）
- 可编程的低电压检测模块（LVD）。

图 1-3、图 1-4 分别给出了 PIC16F877A 和 PIC18F452 两种单片机的内部结构。

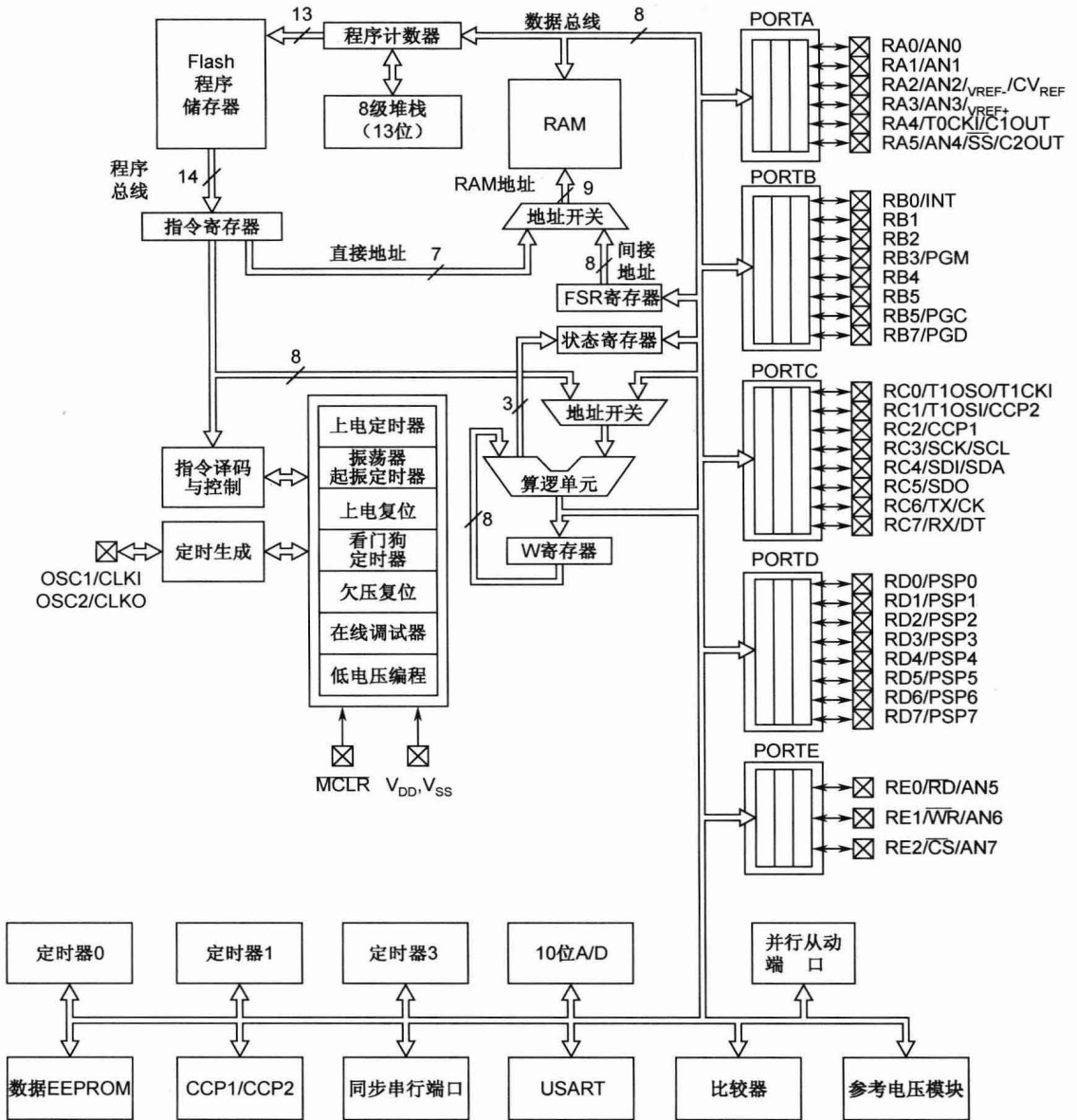


图 1-3 PIC16F877A 单片机内部结构图

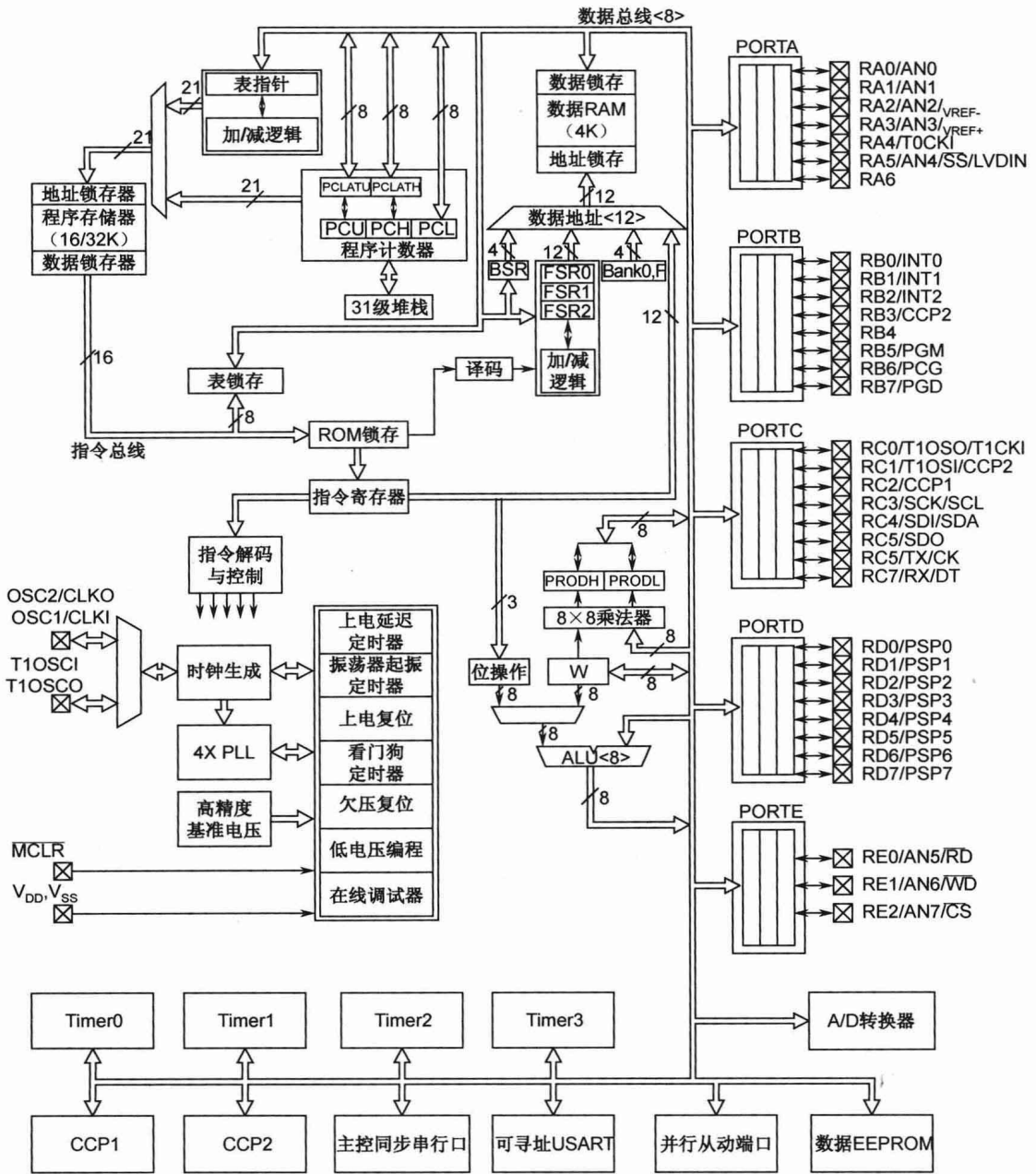


图 1-4 PIC18F452 单片机内部结构图

1.2 MPLAB + C 语言程序开发环境安装及应用

PIC 单片机集成开发平台 MPLAB IDE 可从微芯公网站 (www.microchip.com) 免费下载, 当前版本为 v8.53, 其运行界面如图 1-5 所示。

MPLAB IDE 默认支持编译汇编程序, 编译本书的所有 C 语言源程序时, 还需要单独安装 C 语言程序编译器, 包括 HI-TECH PICC、PICC18 及 Microchip C18 (MCC18) 编译器。

HI-TECH PICC、PICC18 是优化的 PIC 8 位系列单片机 C 编译器, 由第三方公司 HI-TECH 提供。PICC、PICC18 符合 ANSI 标准, 但不支持函数递归调用, PIC 单片机特殊的硬件堆栈结构决定了它无法实现需要大量堆栈操作的递归算法。PICC 与 PICC18 编译器采用“静态覆盖”

技术，给 C 语言程序中的局部变量分配固定地址空间，这样可最大限度地利用 PIC 单片机有限的 RAM 空间。另外，PICC、PICC18 编译器可以与 MPLAB IDE 无缝集成，允许汇编语言和 C 语言混合编程。支持硬件仿真（ICE）、软件仿真（SIM）、在线调试器（ICD）。PICC、PICC18 还提供了丰富的库函数供使用，特别是 PICC18 提供了强大的硬件外设（Hardware Peripheral）与软件外设（Software Peripheral）库函数。



图 1-5 MPLAB IDE 集成开发环境

MPLAB C18(或称 MCC18)是功能齐全的、符合 ANSI 标准的 C 编译器，适用于 Microchip Technology 的 PIC18CXXX 系列 PICmicro® MCU。MPLAB C18 与 Microchip 的 MPLAB IDE 完全兼容，允许用 MPLAB ICE 和 MPLABSIM 模拟器进行源代码调试。MPLAB C18 提供了一个便利的、面向项目的开发环境，缩短了开发时间。

MPLAB C18 提供了强大的 PICmicro MCU 库，用 C 语言为 PIC18CXXX 系列单片机编写程序，使开发人员能有更多的时间投入在应用开发上，减少了耗费在处理器细节的时间。

MPLAB C18 的外设库函数功能强大，HI-TECH PICC18 的大量外设库函数也是直接由 MCC18 提供的外设库函数移植而来。

本书的 PICC/PICC18/MCC18 编译器分别安装在：

- C:\Program Files\HI-TECH Software\PICC\9.70
- C:\Program Files\HI-TECH Software\PICC-18\PRO\9.63
- C:\MCC18

以使用 PICC 编译 PIC16 系列单片机 C 程序项目为例，在创建一个新的开发项目时，首先选择 PIC 单片机型号，然后在第二步“Select a language toolsuite”时选择“C:\Program Files\HI-TECH Software\PICC\9.70\bin\picc.exe”作为编译器，选择对话框如图 1-6 所示。创建项目之后如果要重新选择编译器，可单击菜单“Project/Select Language Toolsuite”。

创建项目后如果要重新选择微处理器，可单击菜单“select device”，在图 1-7 所示的选择设备窗口中重新选择当前项目所选用的微处理器。

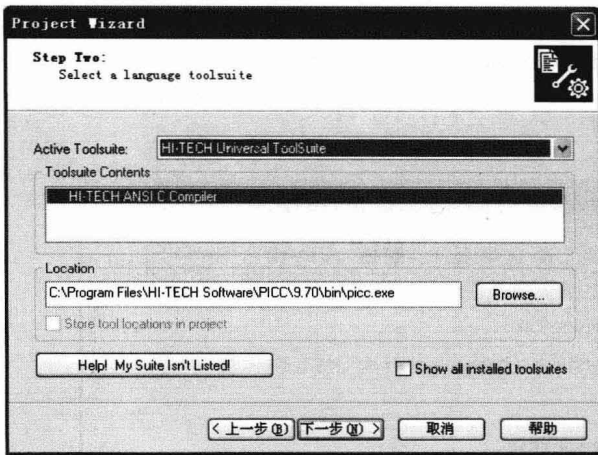


图 1-6 语言工具套件选择对话框

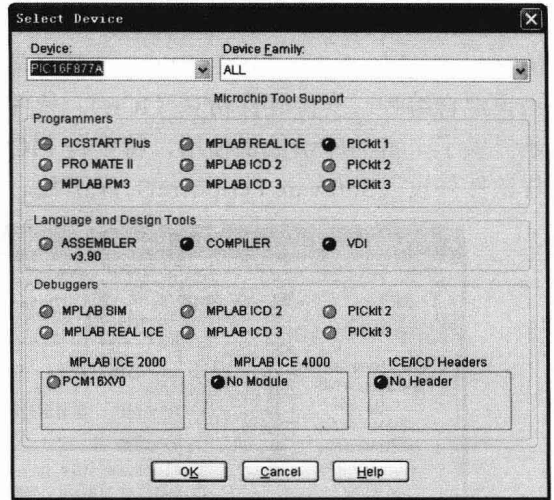


图 1-7 选择设备窗口

1.3 PICC/PICC18/MCC18 程序设计基础

全书涉及的 PIC 单片机 C 语言程序编译器包括：HI-TECH PICC/PICC18 和 Microchip C18 (MCC18)，本节并不准备全面讲述这些编译器所支持的 PIC 单片机 C 语言程序设计基础知识，所列出的内容仅仅是在编写调试本书案例程序时需要参考的内容或需要引起注意的内容。

一、基本数据类型、有符号与无符号数应用、数位分解、位操作

本书所有 PIC 单片机 C 语言程序设计的基本数据类型大部分与 ANSIC 相同，表 1-1 列出了 PICC/PICC18/MCC18 常用的基本数据类型。

表 1-1 PICC/PICC18/MCC18 常用的基本数据类型

数据类型	本书定义	长度(位)	取值范围
bit		1	取值为 0 或 1
signed char (char)	INT8	8	-128~127
unsigned char	INT8U	8	0~255
char		8	默认为无符号类型，通过设置编译选项可改为有符号类型
int (signed int)	INT16	16	-32 768~32 767
unsigned int	INT16U	16	0~65 535
long (signed long)	INT32	32	-2 147 483 648~2 147 483 647
unsigned long	INT32U	32	0~4 294 967 295
float		24/32	默认为 24 位，通过设置编译选项可改为 32 位
double		24/32	同上

设计单片机应用系统时，使用的多数都是无符号数，全书约定将 0~255 以内的无符号整数定义为 INT8U，它相当于字节类型 BYTE；0~65 535 以内的无符号整数定义为 INT16U，它相当于字类型 WORD。

如果涉及正负数的处理，在定义时要注意使用 signed 类型。例如，温度控制程序中有正负温度，DS18B20 温度传感器可处理的实际温度范围为-55℃~125℃，为使程序能对温度值进行正确比较，应使用 INT8 类型（即 signed char），其取值范围为-128~127。

PICC/PICC18/MCC18 遵循 Little-endian 标准, 多字节变量的低字节放在存储空间的低地址, 高字节放在高地址。

对于大量要将整数或浮点数显示在数码管上的案例, 需要对待显示数据进行数位分解, 例如:

```
INT8U d=227;
INT8U c[3];
c[0]=d/100;c[1]=d/10%10;c[2]=d%10;
又如:
float x=123.45;
```

如果要得到 x 的各个数位, 可以先将 x 乘以 100, 然后再分解各数位:

```
INT16U y=x*100;
INT8U c[5],i;
for (i=4;i!=0xFF;i--)
{
    c[i]=y%10; y/=10;
}
```

上面 for 循环中的循环条件一般都会写成 $i \geq 0$ 的, 但由于当 $i=0$ 时, 如果将 i 再减 1, i 将变为 $0xFF$, 这个无符号数仍被认为大于等于 0, 这样就不能保证 5 次循环了, 因此要改写成 $i \neq 0xFF$ (或者 $i \neq -1$)。如果将 i 定义成 INT8 类型 (signed char) 而不是 INT8U 类型 (unsigned char), 使用 $i \geq 0$ 时则可以得到正确结果。

使用 `<stdlib.h>` 提供的函数 `itoa` 或 `utoa` 可将有符号或无符号整数转换为字符串, 将字符串中各字符编码减去 $0x30$ 或 '0' 也能分解出其各个数位。

十进数的数位分解方法常用于大量的数码管显示程序, 显示时需要根据各数位到数码管段码表中提取对应的段码。

位操作在本书案例中也会大量出现, 例如在 LED 流水灯、数码管位码控制、串行收/发、键盘扫描等大量案例中, 各种位操作符都会频繁使用, 例如位左移 (\ll)、右移 (\gg)、与 ($\&$)、或 (\mid)、取反 (\sim)、异或 (\wedge) 等, 这些位操作符都要熟练掌握和运用。

下面是有关位操作符的几个应用示例。

(1) 例如, 要 PORTB 端口 RB7~RB0 逐位循环轮流置 1, 可先定义变量 i , 并使之在 7~0 范围内循环取值, 然后使 $PORTB=1 \ll i$ 。在循环过程中, PORTB 将分别输出 10000000、01000000、00100000、...、00000001, 如此往复。

(2) 如果要 RB7~RB0 逐位循环轮流置 0, 可有 $PORTB=\sim(1 \ll i)$ 。这类位操作在 LED 流水灯设计或集成式数码管位码扫描程序中都会用到。

(3) 又如, 已知 RD7 外接 LED 或蜂鸣器, 要 LED 闪烁或蜂鸣器发声, 可先定义:

```
#define LED_BLINK() RD7=~RD7 或
#define BEEP()      RD7=~RD7
```

然后在循环中反复调用 LED_BLINK() 或 BEEP() 即可实现所要求的输出, RD7 引脚输出序列为...01010101...

(4) 本书多个案例使用了字符液晶, 向连接在 PORTC 端口的液晶屏发送显示数据时, 需要先判断液晶是否忙, 其忙标志在读取字节的最高位, 因此又有类似语句:

```
if ((PORTC&0x80)==0x80) { //液晶忙 }
```

由于“&”操作符的优先级低于“==”, 因而要注意给“PORTC & 0x80”加上括号提升其运算优先级。

(5) 本节前面讨论了十进制数的数位分解及应用, 如果有 BCD 码 $k=0x98$, 要分解出 9 和 8, 还可以有: