

<<零点起步>>

图书基本信息

书名：<<零点起步>>

13位ISBN编号：9787111332749

10位ISBN编号：7111332741

出版时间：2011-4

出版时间：机械工业

作者：闫广明//张波//孙小君

页数：309

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## <<零点起步>>

### 内容概要

《零点起步：PIC单片机常用模块与典型实例》共分13章，介绍了PIC单片机的发展与应用、PIC单片机开发流程、PIC16系列单片机系统结构、单片机C语言、输入/输出端口的用法、中断系统、定时/计数器、A/D转换器、USART串行通信、SPI接口、I2C总线、CCP模块和综合实例等内容。

《零点起步：PIC单片机常用模块与典型实例》采用以Proteus ISIS模拟软件为硬件平台，以MPLAB IDE+HI-TECH PICC编译器为软件平台的实例介绍方式，让读者对每个功能模块都能通过实际动手制作、设计的方式来学习和掌握，从而能够把学到的知识运用到实际产品的设计中。

《零点起步：PIC单片机常用模块与典型实例》既可以作为高等院校相关专业的入门教材，也可以作为单片机工程师的参考书。

## 作者简介

《零点起步:PIC单片机常用模块与典型实例》共分13章,介绍了PIC单片机的发展与应用、PIC单片机开发流程、PIC16系列单片机系统结构、单片机C语言、输入/输出端口的用法、中断系统、定时/计数器、A/D转换器、USART串行通信、SPI接口、I2C总线、CCP模块和综合实例等内容。

《零点起步:PIC单片机常用模块与典型实例》采用以ProteusISIS模拟软件为硬件平台,以MPLABIDE+HI-TECHPICC编译器为软件平台的实例介绍方式,让读者对每个功能模块都能通过实际动手制作、设计的方式来学习和掌握,从而能够把学到的知识运用到实际产品的设计中。

《零点起步:PIC单片机常用模块与典型实例》既可以作为高等院校相关专业的入门教材,也可以作为单片机工程师的参考书。

## 书籍目录

前言第1章 PIC单片机概述 1.1 PIC单片机的发展与应用 1.2 PIC单片机型号选择 1.3 PIC16F877单片机的基本参数 1.4 PIC单片机的功能特性 1.5 思考与练习 第2章 PIC单片机开发流程 2.1 软件平台的安装 2.1.1 MPLAB集成开发环境的安装 2.1.2 测试汇编语言开发环境 2.1.3 C语言编译器HI-TECH PICC的安装 2.2 硬件平台的安装 2.2.1 Proteus ISIS软件的安装与测试 2.2.2 ICD2+开发板的硬件开发平台的安装 2.2.3 基于ICD2的硬件开发平台的测试 2.3 单片机项目的建立与运行 2.3.1 C语言项目建立过程 2.3.2 目标代码的生成与运行 2.4 思考与练习 第3章 PIC16系列单片机系统结构 3.1 PIC16F877单片机系统基本结构 3.2 PIC单片机功能概述 3.2.1 基本功能模块 3.2.2 专用功能模块 3.3 思考与练习 第4章 单片机C语言 4.1 实例：第一个单片机C语言程序 4.2 单片机C语言的数据类型和变量 4.2.1 常量和变量 4.2.2 字符型数据 4.2.3 位型数据 4.2.4 整型数据 4.2.5 浮点型数据 4.2.6 变量的命名规则 4.2.7 数据类型修饰符 4.3 C语言的运算符 4.3.1 数学运算符 4.3.2 位运算符 4.3.3 赋值运算符 4.3.4 逻辑运算符 4.4 C语言的重要控制语句 4.4.1 选择结构 4.4.2 分支结构 4.4.3 while循环结构 4.4.4 for循环结构 4.5 数组 4.5.1 数组的定义与初始化 4.5.2 特别的数组：字符串 4.5.3 数组的使用 4.5.4 实例：数码管显示 4.5.5 二维数组与多维数组 4.6 函数 4.6.1 函数的声明和定义 4.6.2 函数的参数 4.6.3 数组作为函数的参数 4.6.4 何时使用函数 4.6.5 实例：延时函数的编写和使用 4.7 宏定义 4.7.1 不带参数的宏定义 4.7.2 带参数的宏定义 4.8 C语言书写规范 4.8.1 变量和函数的命名规范 4.8.2 常量和宏定义的命名规范 4.8.3 程序的缩进和空格的使用 4.8.4 注释的书写方法 4.9 多文件项目管理 4.9.1 .H文件的书写 4.9.2 .C文件的书写 4.9.3 添加到当前项目的方法 4.10 实例：简单的计数器 4.11 思考与练习 第5章 输入/输出端口的用法 5.1 输入/输出端口简介 5.1.1 输入/输出端口的原理 5.1.2 输入/输出端口的相关寄存器 5.2 输出端口的用法 5.2.1 跑马灯的设计 5.2.2 多数码管的显示控制 5.3 输入端口的用法 5.3.1 单个按键状态的读取 5.3.2 个按键状态的读取 5.4 矩阵小键盘的读取方法 5.4.1  $\times 4$ 矩阵式键盘的工作原理 5.4.2 基于扫描法的矩阵式键盘读取 5.5 应用实例：电子计算器 5.6 思考与练习 第6章 中断系统 6.1 PIC16F877中断系统的硬件结构 6.2 PIC16F877的中断过程 6.3 PIC16F877中断系统的相关寄存器 6.4 C语言中断服务程序的编写 6.5 INT中断的用法 6.5.1 INT中断的相关寄存器 6.5.2 INT中断的硬件连接 6.6 实例：用INT中断来控制LED小灯 6.7 实例：B口中断的使用 6.8 思考与练习 第7章 定时/计数器 7.1 PIC16F877单片机的定时/计数器简介 7.2 TIMER0的硬件结构 7.3 TIMER0的相关寄存器 7.4 实例：精确的秒表 7.5 “看门狗”定时器的用法 7.5.1 WDT的系统结构 7.5.2 WDT的硬件设置 7.5.3 WDT的程序设计 7.5.4 WDT的使用注意事项 7.6 实例：WDT的使用 7.7 思考与练习 第8章 A/D转换器 8.1 系统结构及相关寄存器 8.2 外围硬件电路设计 8.3 查询方式下的操作时序 8.4 实例：查询方式下的A/D转换例程 8.5 中断方式下的A/D转换 8.5.1 相关寄存器 8.5.2 实例：中断方式下的A/D转换 8.6 实例：测量电阻阻值 8.7 思考与练习 第9章 USART串行通信 9.1 通信基础知识 9.1.1 通信协议 9.1.2 数据传送方式 9.1.3 波特率 9.1.4 串行通信的检错和纠错 9.2 USART的系统结构 9.2.1 USART发送器的系统结构 9.2.2 USART发送器的相关寄存器 9.2.3 USART接收器的系统结构 9.2.4 USART接收器的相关寄存器 9.3 单片机与RS-232接口电路设计 9.4 实例：异步发送模式下的程序设计 9.5 实例：双机USART异步通信 9.6 思考与练习 第10章 SPI接口 10.1 功能特点和系统结构 10.2 主控发送模式的原理及应用 10.2.1 主控发送模式的工作流程 10.2.2 主控发送模式的相关寄存器 10.2.3 主控发送模式的硬件连接 10.2.4 实例：主控发送模式的程序设计 10.3 主控接收模式的原理及应用 10.3.1 主控接收模式的工作流程 10.3.2 主控接收模式的相关寄存器 10.3.3 主控接收模式的硬件连接 10.3.4 实例：主控接收模式的程序设计 10.4 实例：25XXX系列EEPROM的读/写 10.4.1 C080的系统结构与操作时序 10.4.2 C080的硬件连接 10.4.3 C080的程序设计 10.5 思考与练习 第11章 I2C总线 11.1 I2C接口的功能特点 11.1.1 I2C总线器件到总线线路的电气连接 11.1.2 I2C协议简介 11.1.3 PIC16F877的I2C模块的特点 11.2 I2C主模式数据发送 11.2.1 I2C主模式的系统结构与数据发送流程 11.2.2 I2C主模式数据发送过程中的相关寄存器 11.2.3 I2C模块的硬件连接 11.2.4 实例：I2C主模式数据发送的程序设计 11.3 I2C主模式数据接收 11.3.1 I2C主模式数据接收流程 11.3.2 I2C主模式数据接收过程中的相关寄存器 11.3.3 实例：I2C主模式数据接收的程序设计 11.3.4 I2C主模式的复合数据帧 11.4 实例：I2C接口EEPROM的读/写 11.4.1 C02C功能简介 11.4.2 C02C的系统结构与读/写帧格式 11.4.3 C02C的硬件连接 11.4.4 读/写24C02C的程序设计 11.5 思考与练习 第12章 CCP模块 12.1 CCP1模块的捕捉模式 12.1.1 CCP1模块捕捉模式的系统结构和工作流程 12.1.2 CCP1模块捕捉模式下的相关寄存器 12.1.3 实例：基于捕捉

功能的频率计设计 12.2 CCP1模块的比较模式 12.2.1 CCP1模块比较模式的系统结构和工作流程 12.2.2 CCP1模块比较模式下的相关寄存器 12.2.3 实例：基于比较功能的方波发生器设计 12.3 CCP1模块的脉宽调制模式 12.3.1 CCP1模块脉宽调制模式的系统结构和工作流程 12.3.2 CCP1模块脉宽调制模式下的相关寄存器 12.3.3 脉宽调制模式的周期与占空比 12.3.4 实例：基于脉宽调制功能的方波发生器设计 12.4 CCP2模块与CCP1模块的区别和联系 12.5 思考与练习 第13章 综合实例 13.1 字符式液晶模块原理及编程 13.1.1 LM016L的系统结构与硬件连接 13.1.2 LM016L的指令集 13.1.3 LM016L的初始化 13.1.4 LM016L的字符显示 13.1.5 实例：基于LM016L的秒表显示设计 13.2 LED点阵显示屏原理及应用 13.2.1 典型LED点阵显示屏电路设计与分析 13.2.2 LED点阵显示屏显示汉字的原理 13.2.3 实例：在LED点阵显示屏上显示汉字 13.3 电子密码锁控制器 13.3.1 需求分析 13.3.2 硬件设计 13.3.3 软件设计 13.4 思考与练习 参考文献

## 章节摘录

版权页：插图：volatile类型定义在单片机的C语言编程中如此重要，是因为它可以告诉编译器的优化处理器这些变量是实实在在存在的，在优化过程中不能无故消除。

假定读者的程序定义了一个变量并对其作了一次赋值，但随后就再也没有对其进行任何读 / 写操作，如果是非volatile型变量，优化后的结果是这个变量将有可能被彻底删除以节约存储空间。

另外一种情形是在使用某一个变量进行连续的运算操作时，这个变量的值将在第一次操作时被复制到中间临时变量中，如果它是非volatile型变量，则紧接其后的其他操作将有可能直接从临时变量中取数以提高运行效率，显然这样做后对于那些随机变化的参数就会出问题。

只要将其定义成volatile类型后，编译后的代码就可以保证每次操作时直接从变量地址处取数。

3.const——常数型变量声明如果变量定义前冠以const类型修饰，那么所有这些变量就成为常数，程序运行过程中不能对其修改。

除了位变量，其他所有基本类型的变量或高级组合变量都将被存放在程序空间（ROM区）以节约数据存储空间。

显然，被定义在ROM区的变量是不能再在程序中对其进行赋值修改的，这也是const的本来意义。

实际上这些数据最终都将以retlw的指令形式存放在程序空间，但PICC会自动编译生成相关的附加代码从程序空间读取这些常数，程序员无须太多操心。

<<零点起步>>

编辑推荐

《零点起步:PIC单片机常用模块与典型实例》：循序渐进，由浅入深技术全面，讲解清晰代码规范，实用性强学以致用，注重实践

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>