

<<单片机原理与应用技术>>

图书基本信息

书名：<<单片机原理与应用技术>>

13位ISBN编号：9787030271143

10位ISBN编号：7030271149

出版时间：2010-4

出版时间：科学

作者：高惠芳 编

页数：332

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>

<<单片机原理与应用技术>>

前言

单片机出现至今已经有30多年的历史，单片机技术也历经了几个发展阶段。目前，单片机已渗透到生活中的各个领域，几乎很难找到哪个领域没有单片机的足迹。导弹的导航装置，飞机的各种控制仪表，计算机的网络通信与数据传输模块，工业自动化过程的实时控制和数据处理设备，广泛使用的各种智能IC卡，民用豪华轿车的安全保障系统，摄像机、全自动洗衣机的控制系统，以及程控玩具、电子宠物等，甚至全自动控制领域的机器人、智能仪表、医疗器械，都离不开单片机。

因此，单片机的学习、开发与应用将造就一批计算机应用与智能化控制的工程师和科学家。科技越发达，智能化的东西就越多，对单片机学习的需求也日益增加。

目前，单片机方面的教材大都采用汇编语言的讲解和设计程序实例，但汇编语言学习困难。在实际应用系统开发调试中，特别是开发比较复杂的应用系统时，为了提高开发效率和使程序便于移植，很多时候采用C语言。

C语言不仅学习方便，而且同汇编语言一样能够对单片机资源进行访问，因此目前大多数院校在开设单片机课程时都引入C语言。

但引入C语言后，就发现在选择教材时存在两方面的问题：有的教材注重于单片机的原理，只使用汇编语言；而另一些教材注重于C语言，一般面向开发，不讲原理，属于高级教程，不适合初学者。能兼顾汇编语言和C语言的教材非常少，而在实际使用中需要一本在学习单片机基本原理的同时能兼顾汇编语言和C语言两个方面的教材。

在整个大学阶段，大多数学生学习的课程中，只有单片机这门课能接触到汇编语言，所以该门课旨在使学生在汇编语言概念的基础上学会单片机的编程。

本书编写的目的是在讲述单片机基本原理的同时能兼顾汇编语言和C语言两个方面。

所以本书在大多数的实例中，相同的功能用汇编语言和C语言分别编程实现，通过用汇编语言和C语言两个方面的编程对比，使学生能够有选择地掌握一种语言，并认识另一种语言。

同时，为了提高学生应用设计的能力，本书还介绍了目前单片机接口常用的接口芯片，列举了几个简单的单片机应用系统开发实例。

<<单片机原理与应用技术>>

内容概要

《单片机原理与应用技术》针对目前最通用的MCS-51单片机，在汇编语言的基础上，增加了目前最流行的C51程序设计语言，内容主要包括：单片机芯片的硬件原理和结构、汇编语言指令系统和程序设计、C51的程序设计、单片机内部资源(包括中断、定时/计数器、串行口)、单片机系统扩展(包括存储器扩展、I/O扩展)及功能扩展(包括键盘、显示器、A/D及D/A转换)、KeilC集成调试软件及Proteus仿真软件的使用介绍等。

《单片机原理与应用技术》的特点是通过汇编语言和C语言穿插进行讲述，实例较多，且很多例子都给出了汇编语言和C语言的对照程序，使读者能同时学习汇编语言和C语言，并使熟悉汇编语言的读者能更快地学好单片机C51程序设计。

《单片机原理与应用技术》可作为高等院校电类、机械类专业本科生的教材，也可作为函授教材或培训班教材。

另外，《单片机原理与应用技术》可供从事单片机应用产品研发的工程技术人员及单片机爱好者参考。

<<单片机原理与应用技术>>

书籍目录

前言第1章 单片机概述1.1 单片机的基本概念1.1.1 什么是单片机1.1.2 微控制器MCU1.1.3 如何使用单片机1.2 单片机的发展概况1.2.1 单片机的发展历史1.2.2 MCS-51单片机的发展1.2.3 现阶段主流单片机系列简介1.3 单片机的应用特点与应用领域1.3.1 单片机应用特点1.3.2 单片机应用领域1.4 单片机的发展趋势习题第2章 MCS-51单片机的结构和原理2.1 MCS-51单片机的基本组成2.1.1 MCS-51单片机的基本组成2.1.2 MCS-51单片机的封装与信号引脚2.2 MCS-51单片机的并行I/O端口结构2.3 MCS-51单片机的存储器结构2.3.1 程序存储器2.3.2 数据存储器2.4 MCS-51单片机的时钟电路与时序2.4.1 时钟电路2.4.2 时序2.5 MCS-51单片机的工作方式2.5.1 复位方式2.5.2 程序执行方式2.5.3 单步执行方式2.5.4 低功耗方式2.5.5 EPROM的编程和校验工作方式习题第3章 MCS-51单片机的汇编语言指令系统3.1 指令格式及其符号说明3.1.1 指令格式3.1.2 常用符号说明3.1.3 指令的字节3.2 寻址方式3.2.1 立即寻址3.2.2 直接寻址3.2.3 寄存器寻址3.2.4 寄存器间接寻址3.2.5 变址寻址3.2.6 相对寻址3.2.7 位寻址3.3 MCS-51单片机指令系统3.3.1 数据传送类指令3.3.2 算术运算类指令3.3.3 逻辑运算及移位类指令3.3.4 控制转移类指令3.3.5 位操作类指令习题第4章 MCS-51单片机汇编语言程序设计4.1 汇编语言的概述4.1.1 汇编语言的特点4.1.2 汇编语言的伪指令4.2 汇编语言源程序的编辑和汇编4.2.1 手工编程和汇编4.2.2 机器编辑和交叉汇编4.3 汇编语言程序设计4.3.1 简单程序设计4.3.2 分支程序设计4.3.3 循环程序设计4.3.4 数制转换程序4.3.5 查表程序设计习题第5章 单片机C51程序设计基础5.1 C51数据与运算5.1.1 C51的数据类型5.1.2 C51数据的存储类型5.1.3 8051特殊功能寄存器的C51定义5.1.4 8051并行接口及位变量的C51定义5.2 C51运算符、表达式及其规则5.2.1 (复合)赋值运算符5.2.2 算术运算符5.2.3 自增和自减运算符5.2.4 关系运算符5.2.5 逻辑运算符5.2.6 位运算符5.2.7 条件运算符5.2.8 指针和地址运算符5.2.9 优先级和结合性5.3 C51流程控制语句5.3.1 C51程序的基本结构及其流程图5.3.2 选择语句5.3.3 循环语句5.4 C51构造数据类型5.4.1 数组5.4.2 指针5.4.3 结构体5.4.4 共用体5.4.5 枚举5.5 函数5.5.1 函数的定义5.5.2 函数的调用5.5.3 函数的嵌套调用与递归调用5.5.4 中断服务函数5.5.5 指向函数的指针变量5.5.6 局部变量和全局变量5.6 C51的库函数5.6.1 一般I/O函数, stdio.h5.6.2 字符函数库string.h5.6.3 标准函数库stdlib.h及其他头文件5.7 C51模块化程序设计5.7.1 基本概念5.7.2 模块化程序开发过程习题第6章 单片机内部资源及编程6.1 中断系统6.1.1 中断概述6.1.2 中断源6.1.3 中断控制6.1.4 中断响应过程6.1.5 中断请求的撤除6.1.6 中断程序设计6.1.7 外部中断源的扩展6.2 定时/计数器6.2.1 定时/计数器的结构及工作原理6.2.2 定时/计数器的控制6.2.3 定时/计数器的工作方式6.2.4 定时/计数器的初始化6.2.5 定时/计数器应用举例6.3 串行通信口6.3.1 数据通信概述6.3.2 单片机的串行通信接口6.3.3 串行通信的工作方式及波特率设置6.3.4 串行口应用举例习题第7章 单片机系统扩展7.1 单片机最小应用系统7.1.1 单片机最小应用系统构成7.1.2 系统扩展的内容与方法7.2 存储器的扩展7.2.1 程序存储器的扩展7.2.2 数据存储器的扩展7.2.3 存储器综合扩展7.2.4 闪速存储器及其扩展7.3 输入与输出口的扩展7.3.1 简单的并行I/O接口扩展7.3.2 用8255扩展并行I/O接口7.4 串行口扩展7.4.1 FC总线及其接口芯片7.4.2 SPI接口及其接口芯片7.4.3 单总线接口及其接口芯片7.4.4 Microwire串行总线及其接口芯片习题第8章 单片机功能扩展8.1 键盘输入及接口8.2 显示器及其接口8.3 D/A转换器的接口与应用8.4 A/D转换器的接口与应用习题第9章 单片机应用系统的开发与设计9.1 单片机开发系统9.2 51实验板9.3 单片机应用系统的设计9.4 单片机应用系统举例习题第10章 Keil C51软件使用介绍10.1 Keil C51软件安装10.2 霏ision3集成开发环境10.3 建立工程项目10.4 程序举例10.5 编译程序10.6 程序调试习题第11章 可视化仿真开发工具Proteus介绍11.1 Proteus ISIS编辑环境介绍11.2 进入Proteus ISIS编辑环境11.3 单片机仿真11.4 Proteus与Keil整合构建单片机虚拟实验室附录 MCS-51单片机指令汇总附表1 数据传送类指令附表2 算术运算类指令附表3 逻辑运算及移位类指令附表4 控制转移类指令附表5 位操作类指令参考文献

<<单片机原理与应用技术>>

章节摘录

如前所述，单片机是一个单片化微型计算机系统，使用单片机完成一个具体应用需要对其编程，一般来说，flash结构的单片机在出厂的时候单片机内ROM（即flash）是空白的，可以认为它的功能未被定义，需要输入针对应用的代码来实现具体的应用功能。

为产生上述代码一般流程是需要先在单片机开发环境当中编写针对程序，经编译获得所需执行代码

。这里的单片机开发环境是指一种在计算机上的软件，通常它具有编辑程序、交叉编译、生成执行代码文件的功能。

不同的单片机具有不同的开发环境，例如：图1.4所示的单片机属于MCSr-51系列，可以使用Keil Vision 2这个集成开发环境，而AVR系列单片机却可以使用AVR Sttdio，除此之外，开发环境也可能支持不同的编程语言，常见的单片机开发语言有汇编语言和C语言。

单片机开发者需要针对具体的应用来写对应的汇编语言或C语言程序，在集成开发环境中进行编译、调试，最后获得所需的执行代码文件，通过编程器或其他特定的编程方式把执行代码文件烧写单片机的flash ROM中，这样该单片机就具有了针对应用的逻辑控制功能。

当然在开发过程当中，开发者的程序可能会存在问题，往往需要反复调试修改代码。

这种调试一般有两种方式：一种是在集成开发环境中，用指令模拟器模拟单片机执行程序的过程；另一种是通过连接计算机与单片机系统的硬件调试工具（一般称为仿真器），在开发环境上进行硬件调试，正常情况下，调试时间远大于初始程序编写时间。

烧写了执行代码的单片机在系统上电后，就会按照开发者的意愿执行相应的逻辑控制功能。

<<单片机原理与应用技术>>

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>