

## <<单片机原理及应用>>

### 图书基本信息

书名：<<单片机原理及应用>>

13位ISBN编号：9787121185861

10位ISBN编号：7121185865

出版时间：2013-1

出版时间：电子工业出版社

作者：林立，张俊亮 编著

页数：265

字数：464000

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>

## &lt;&lt;单片机原理及应用&gt;&gt;

## 前言

“单片机原理及应用”课程是工院校各类专业普遍开设的一门公共技术基础课，它以MCS-51系列单片机为例介绍其硬件系统原理、软件编程方法以及系统接口原理及应用三大板块内容。长期以来，该课程存在原理概念不易理解、软件编程不易掌握、开发技能不易提高的问题，使不少学习者颇感困惑。

近年来随着单片机仿真软件的普及，促进了国内单片机课程教学水平的提升，单片机教材内容也有待进一步跟进。

著名仿真软件Proteus有句响亮的商业口号“从概念到产品”，宣称其功能的多样性。

我们认为，改进单片机教材也需要有“从概念到产品”的思想，即在重视原理教学的同时，强化动手能力的培养，掌握研发单片机应用产品全过程的各项技能。

为此，我们运用新思路、新知识、新方法编写了这部教材。

本教材特色如下：1. 引入仿真工具内容，拓宽课程知识体系 将当前国际上两个最先进的单片机仿真工具软件Proteus和Keil C内容引入到单片机教材中来，形成了硬件、软件、接口、工具四大板块有机结合的新体系。

仿真工具的引入对理解单片机系统原理和软件系统编程都有较大促进作用。

新技术与传统知识的结合不仅拓宽了读者的视野，也使学习方法发生了深刻变化，可收到事半功倍的学习效果。

2. 编程语言重新定位，C51与汇编脚色互换 重新定位了汇编语言与C51语言的教学目标：C51作为主干编程语言，要求熟练掌握，灵活应用；汇编语言作为辅助编程语言，只要求一般了解，能读懂源程序即可。

如此可使读者在仿真工具软件的辅助下，一步到位地掌握实用性强的C51编程语言和适量的汇编语言，为日后解决复杂应用问题打下基础。

3. 设计全新仿真实验，注重实践能力培养 针对单片机实物实验难以提供充足的动手实践机会的问题，特别设计了与重要知识点衔接、具有设计性或综合性特点、基于仿真软件的一套全新教学实验。

为配合课前预习，“实验指导”中还附有内容丰富的阅读材料。

利用Proteus和Keil C近似真实的单片机软、硬件运行环境，为读者打造一个可自由发挥潜力、循序渐进提高应用技能的实践平台。

4. 优化教材内容结构、注重知识自主获取 针对以往教材的应用实例多集中在最末章节的问题，调整了内容编排顺序，将部分I/O口应用内容前移至中部章节，以此缩短理论教学与应用教学的时间差；按照认知规律解析疑难概念、精心组织衔接关系、认真筛选应用实例，从而使教材的可读性与知识自主获取性显著提高。

本书第1版自2009年7月出版后，得到了广大读者的好评和支持，被许多学校选为教材，并于2010年获得“全国电子信息类优秀教材一等奖”。

不少读者和教师对本书提出过许多宝贵意见，虽然在第1版期间的5次重印过程中已陆续作过一些修正，但限于版面不能大改，仍有局限。

本次再版，仍保持了原教材的体系结构，但对原第5~8章及附录A等内容进行了重新编写，以使全书内容和风格更加统一。

应用实例也进行了调整，剔除过长、过难、仿真可视性差的实例，补充了一些与知识点衔接更好的实例，并全部进行了仿真调试。

为解决课堂仿真演示费时费力的问题，在已有PPT课件和仿真实例的基础上，新增了配套仿真视频。

我们愿借这次再版的机会，将自己在教学中积累的心得与广大读者真诚交流。

此次再版工作由林立和张俊亮共同完成，马义来、王中金、赵旭林、高坡对本书的再版做了大量工作。

本书在第1版重印以及再版过程中得到了电子工业出版社的大力支持和帮助，尤其是高等教育分社的凌毅编辑，对本书的重印和再版做了大量细致的工作，在此谨致以诚挚的谢意。

<<单片机原理及应用>>

再版后的教材一定还有许多不足之处，殷切地期望读者给予批评指正。

作者 2012年10月

## <<单片机原理及应用>>

### 内容概要

本书以MCS-51系列单片机80C51为例介绍单片机的工作原理、基本应用与开发技术。主要内容包括单片机基础知识、内外系统结构、汇编与C51语言、中断与定时/计数器、串口通信、系统接口、应用系统设计等。

本书在单片机传统教学体系的基础上进行了较大改进，以C51编程语言作为贯穿全书各章节的主线，并将单片机仿真软件Proteus和C51编译软件Keil的用法与之紧密衔接，书末附有与教学进度呼应的8个实验指导及相关阅读材料。

## <<单片机原理及应用>>

### 书籍目录

#### 第1章 单片机基础知识概述

##### 1.1 单片机概述

###### 1.1.1 单片机及其发展概况

###### 1.1.2 单片机的特点和应用

###### 1.1.3 单片机的发展趋势

###### 1.1.4 MCS-51单片机的学习

##### 1.2 单片机学习的预备知识

###### 1.2.1 数制及其转换

###### 1.2.2 有符号数的表示方法

###### 1.2.3 位、字节和字

###### 1.2.4 BCD码

###### 1.2.5 ASCII码

###### 1.2.6 基本逻辑门电路

##### 1.3 Proteus应用简介

###### 1.3.1 ISIS模块应用举例

###### 1.3.2 ARES模块应用举例

##### 本章小结

##### 习题

#### 第2章 MCS-51单片机的结构及原理

##### 2.1 MCS-51单片机的结构

###### 2.1.1 MCS-51单片机的内部结构

###### 2.1.2 MCS-51外部引脚及功能

##### 2.2 MCS-51的存储器结构

###### 2.2.1 存储器划分方法

###### 2.2.2 程序存储器

###### 2.2.3 数据存储器

##### 2.3 单片机的复位、时钟与时序

###### 2.3.1 复位与复位电路

###### 2.3.2 时钟电路

###### 2.3.3 单片机时序

##### 2.4 并行I/O口

###### 2.4.1 P1口

###### 2.4.2 P3口

###### 2.4.3 P0口

###### 2.4.4 P2口

##### 本章小结

##### 习题

#### 第3章 单片机的汇编语言与程序设计

##### 3.1 汇编语言概述

###### 3.1.1 汇编语言指令格式

###### 3.1.2 描述操作数的简记符号

###### 3.1.3 寻址方法

##### 3.2 MCS-51指令系统简介

###### 3.2.1 数据传送与交换类指令

###### 3.2.2 算术运算类指令

## <<单片机原理及应用>>

3.2.3 逻辑运算及移位类指令

3.2.4 控制转移类指令

3.2.5 伪指令

3.3 汇编语言的编程方法

3.3.1 Proteus仿真开发系统

3.3.2 汇编程序应用举例

本章小结

习题

第4章 单片机的C51语言

4.1 C51的程序结构

4.1.1 C51语言概述

4.1.2 C51的程序结构

4.2 C51的数据结构

4.2.1 C51的变量

4.2.2 C51的指针

4.3 C51与汇编语言的混合编程

4.3.1 在C51中调用汇编程序

4.3.2 在C51中嵌入汇编代码

4.4 C51仿真开发环境

4.4.1 Keil的编译环境  $\mu$  Vision3

4.4.2 基于Proteus和Keil C的程序开发过程

4.5 C51应用编程初步

4.5.1 I/O端口的简单应用

4.5.2 I/O端口的进阶实践

本章小结

习题

第5章 单片机的中断系统

5.1 中断的概念

5.2 中断控制系统

5.2.1 中断系统的结构

5.2.2 中断控制

5.3 中断处理过程

5.4 中断的编程和应用举例

5.4.1 中断程序设计举例

5.4.2 扩充外部中断源

本章小结

习题

第6章 单片机的定时/计数器

6.1 定时/计数器的结构与工作原理

6.1.1 定时/计数器的基本原理

6.1.2 定时/计数器的结构

6.2 定时/计数器的控制

6.2.1 TMOD寄存器

6.2.2 TCON寄存器

6.3 定时/计数器的工作方式

6.3.1 方式1

6.3.2 方式2

## <<单片机原理及应用>>

6.3.3 方式0

6.3.4 方式3

6.4 定时/计数器的编程和应用

本章小结

习题

第7章 单片机的串行口及应用

7.1 串行通信概述

7.2 MCS-51的串行口控制器

7.2.1 串行口内部结构

7.2.2 串行口控制寄存器

7.3 串行工作方式0及其应用

7.4 串行工作方式1及其应用

7.5 串行工作方式2及其应用

7.6 串行工作方式3及其应用

本章小结

习题

第8章 单片机接口技术

8.1 单片机的系统总线

8.1.1 三总线结构

8.1.2 地址锁存原理及实现

8.2 简单并行I/O口扩展

8.2.1 访问扩展端口的软件方法

8.2.2 简单并行输出接口的扩展

8.2.3 简单并行输入接口的扩展

8.3 可编程并行I/O口扩展

8.3.1 8255A的内部结构、引脚及地址

8.3.2 8255A的控制字

8.4 D/A转换与DAC0832应用

8.4.1 DAC0832的工作原理

8.4.2 DAC0832与单片机的接口及编程

8.5 A/D转换与ADC0809应用

8.5.1 逐次逼近式模数转换器的工作原理

8.5.2 ADC0809与单片机的接口及编程

8.6 开关量功率接口技术

8.6.1 开关量功率驱动接口

8.6.2 开关量功率驱动接口应用举例

本章小结

习题

第9章 单片机应用系统的设计与开发

9.1 单片机系统的设计开发过程

9.1.1 单片机典型应用系统

9.1.2 单片机应用系统的开发过程

9.2 单片机系统的可靠性技术

9.2.1 硬件抗干扰技术概述

9.2.2 软件抗干扰技术概述

9.3 单片机系统设计开发应用举例——智能仪器

9.3.1 功能概述

## <<单片机原理及应用>>

9.3.2 硬件电路设计

9.3.3 软件系统设计

9.3.4 仿真开发过程

本章小结

习题

附录A 实验指导

实验1 计数显示器

【阅读材料1】 ISIS模块的电路绘图与仿真运行方法

实验2 指示灯/开关控制器

【阅读材料2】 ISIS模块的汇编程序创建与调试方法

实验3 指示灯循环控制

【阅读材料3】 在  $\mu$  Vision3中创建C51程序的方法

实验4 指示灯/数码管的中断控制

【阅读材料4】 C51程序调试方法

实验5 电子秒表显示器

【阅读材料5】  $\mu$  Vision3与ISIS的联合仿真

实验6 双机通信及PCB设计

【阅读材料6】 基于ARES模块的PCB设计方法

实验7 直流数字电压表设计

【阅读材料7】 ISIS中的虚拟信号发生器

实验8 步进电机控制设计

【阅读材料8】 步进电机控制方法

参考文献



## &lt;&lt;单片机原理及应用&gt;&gt;

## 章节摘录

版权页：插图：去耦电容的选用可以按照 $C=1/f$ 选用，其中 $f$ 为电路频率，即10MHz取0.1 $\mu$ F，100MHz取0.01 $\mu$ F。

去耦电容应直接跨接在芯片的源和地之间，数字电路每一颗芯片原则上应配置一个去耦电容，以便随时充放电。

3.模拟地和数字地的分离 在单片机构成的数据采集系统中，往往既有数字信号又有模拟信号。由于单片机工作频率较高，易于产生开关噪声等高频干扰信号，这些干扰信号经过地线传入模拟量输入电路，会引起模拟量的输入采集信息的误差。

为避免模拟信号与数字信号间的相互窜扰，在模拟、数字混合的单片机系统，将模拟部分和数字部分的地信号相分离为模拟地和数字地，模拟和数字部分各自构成独立回路，与此同时，模拟地和数字地通过电感或磁珠相连接，形成“分区集中并联一点接地”，这样，既可以保证模拟和数字部分具有相同的地电位参考平面，又使得地线电流不会流到其他功能单元的回路中，避免各个单元的相互干扰。另外，在PCB设计时，地线应该尽量加粗，必要时可以采用“铺地”技术，以减少地线的阻抗。

4.“看门狗”技术 应用系统工作在恶劣环境或大噪声的干扰环境下，由于外界干扰对CPU的影响，使得程序不能按照正常的设计要求运行，出现程序跑飞或死循环的现象。

对此，可以通过“看门狗”电路强制CPU复位，使系统重新进入正常运行的轨道。

单片机在正常工作的情况下，通过定时器设置产生脉冲信号，将该脉冲信号送入“看门狗”电路“喂狗”，“看门狗”在定时“吃到”脉冲的情况下，不产生复位的操作。

当单片机工作系统出现异常，不能再定时向“看门狗”提供定时脉冲“喂狗”时，“看门狗”电路自动产生复位信号，通过硬件驱动单片机系统复位，从而，使“跑飞”或陷入死循环的CPU重新运行程序，摆脱由于干扰而造成系统异常的状态。

9.2.2软件抗干扰技术概述 在提高硬件系统抗干扰能力的同时，软件抗干扰以其设计灵活、节省硬件资源、可靠性好也越来越受到重视。

1.指令冗余 CPU取指令过程是先取操作码，再取操作数。

当PC受干扰出现错误时，程序便脱离正常轨道“乱飞”，若乱飞到某双字节指令，且取指令时刻落在操作数上，误将操作数当作操作码，程序将出错。

若“飞”到了三字节指令，出错概率更大。

在关键地方人为插入一些单字节指令，或将有效单字节指令重写称为指令冗余。

通常是在双字节指令和三字节指令后插入两字节以上的NOP。

这样即使程序飞到操作数上，但由于空操作指令NOP的存在，从而避免了后面的指令被当作操作数执行，使程序自动纳入正轨。

此外，在对系统流向起重要作用的指令如RET、RETI、LCALL、LJMP、jc等之前插入两条NOP，也可将乱飞程序纳入正轨，确保这些重要指令的执行。

2.软件陷阱 当乱飞程序进入非程序区，冗余指令便无法起作用。

通过软件陷阱，可以拦截乱飞程序，将其引向指定位置，再进行出错处理。

软件陷阱是指用来将捕获的乱飞程序引向复位入口地址0000H的指令。

通常在EPROM中非程序区填入以下指令作为软件陷阱：其机器码为0000020000。

考虑到程序存储器的容量，软件陷阱一般每1KB空间有2~3个就可以进行有效拦截了。

<<单片机原理及应用>>

编辑推荐

<<单片机原理及应用>>

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>