

高等学校电子信息类专业“十二五”规划实践教材

微处理器类新技术 实验教程

周佳社 石光明 等编著



西安电子科技大学出版社
<http://www.xduph.com>

高等学校电子信息类专业“十二五”规划实践教材

微处理器类新技术实验教程

周佳社 石光明 许 辉 张剑贤 编著
何 迪 孙江敏 楼顺天

西安电子科技大学出版社

内 容 简 介

本教程包括微处理器导论及微机原理与接口技术实验、PIC 单片机原理与应用实验、英飞凌单片机应用开发实验、TI-DSP 技术基础实验、嵌入式系统设计实验等五门实验课内容,每门实验课为一章。每章开始均对该课程的知识要点及教学要求进行了总结,以便于学生对理论知识进行复习巩固。在实验内容安排上,既有基础实验,又有综合设计实验,以夯实学生的基础,培养学生综合设计与创新设计的能力。

本教程着眼于实验技能与工程应用,通过学习与实验,为微处理器应用及系统设计打下坚实的基础。各实验内容简明扼要、深入浅出,融入了作者的经验与体会,便于学习和掌握。本教程除可作为高校工科相关专业的实验教材外,对于一般工程技术人员也有很好的参考价值。

图书在版编目(CIP)数据

微处理器类新技术实验教程/周佳社等编著. —西安:西安电子科技大学出版社,2012.12

高等学校电子信息类专业“十二五”规划实践教材

ISBN 978-7-5606-2949-0

I. ①微… II. ①周… III. ①微处理器—实验—高等学校—教材 IV. ①TP332-33

中国版本图书馆 CIP 数据核字(2012)第 254651 号

策划编辑 张 媛

责任编辑 李惠萍 曹媛媛 张 绚

出版发行 西安电子科技大学出版社(西安市太白南路2号)

电 话 (029)88242885 88201467 邮 编 710071

网 址 www.xduph.com 电子邮箱 xdupfxb001@163.com

经 销 新华书店

印刷单位 陕西天意印务有限责任公司

版 次 2012年12第1版 2012年12第1次印刷

开 本 787毫米×1092毫米 1/16 印张 19

字 数 450千字

印 数 1~1000册

定 价 33.00元

ISBN 978-7-5606-2949-0/TP

XDUP 3241001-1

如有印装问题可调换

本社图书封面为激光防伪覆膜,谨防盗版。

前 言

电工电子技术实践教学在电子信息类人才培养过程中具有举足轻重的作用,特别是在现代电子技术飞速发展的今天,如何保持和提升电工电子技术实践教学水平,是高校电子信息类专业建设的永恒主题。西安电子科技大学以教育部“质量工程”为引领,针对电子信息类本科教学滞后该领域发展以及人才培养不能满足社会发展需求的现状,与时俱进,借鉴国际先进教材,结合多年的教学经验,面向现代电子技术发展,突破传统实验教材验证理论知识的模式,以综合性、设计性、研究性的视角呈现现代电子技术的特点,系统化整合了原来零散的单一主题实验课程,形成了“微处理器类新技术实验”国家级精品课程。该精品课程是面向所有电子信息类专业本科学生开设的由基础层向提高层逐渐深化的一系列实验课程,是由“微机原理与接口技术实验”、“PIC 单片机原理与应用实验”、“英飞凌单片机应用开发实验”、“TI-DSP 技术基础实验”、“嵌入式系统设计实验”等五门实验课程组成。该课程以微处理器和微控制器为核心,通过研究型实验教学手段和方法,以基础性、设计性、综合性和研究性的实验项目为载体,使学生学习掌握科学的研究问题的方法,建立创新意识,增强解决问题的能力,提高自主学习的能力。

本书作者均有十几年承担“微机原理与系统设计”、“单片机原理与应用”、“DSP 技术与应用”、“嵌入式系统设计”等课程的理论与实践教学经验,完成了十多项相关的科研项目,自主开发了“8086 单板型微型计算机实验教学系统”、“基于 EISA/PCI 系统总线的微机原理及接口技术实验教学系统”、“可重构微处理器应用系统”、“模型机编译系统”等实践教学平台与设备,并有出版多本教材和专著的写作经验,2006 年出版的相关教材《微机原理与接口技术》(科学出版社)入选“十一五”国家级规划教材。正是基于作者的这些经验使本书具有下列特点:

(1) 每门实验课程都围绕理论知识要点、重点及难点进行了总结,使学生在做实验的同时能对理论知识进行复习与巩固,从而增强学生积极参与实验与科技实践的自觉性及积极性。

(2) 面向学生,面向教师。根据多年的教学经验,从学生学习、教师讲授的角度出发,内容由浅入深,循序渐进,每门实验课程既有基础型实验内容,也有综合性、设计型实验内容。在培养学生扎实的理论基础的同时,注重培养学生工程设计及创新设计的能力。

(3) 结合作者多年的教学、工程设计经验,对工程设计中常用的实例及常见问题的解决方法等进行了较全面的总结,凝练成实验题目,使教材具有一定的实用价值。

(4) 语言流畅,内容选择合理,结构编排适当,保证了教材的质量。

本书由周佳社主编,石光明负责统稿与审核,编写分工为:第一章由石光明、楼顺天共同编写,前言和第二章由周佳社、石光明共同编写,第三章由许辉编写,第四章由张剑贤编写,第五章由何迪编写,第六章由孙江敏编写。

感谢西安电子科技大学的相关老师给予本书的支持,更要感谢西安电子科技大学出版社的同志对本书进行的认真编校以及为本书的推广应用所作的辛勤劳动。

由于作者水平和知识面的限制,书中难免会有疏漏和欠妥之处,敬请读者批评指正,以便及时更正,使本书有益于更广大的读者。

编著者

2012年8月

目 录

第一章 微处理器导论	(1)
1.1 微处理器的发展历程	(1)
1.2 单片机	(2)
1.3 嵌入式系统	(2)
1.4 数字信号处理器(DSP)	(3)
1.5 基于 FPGA 的模型机	(3)
1.6 微处理器的共性及新技术实验	(4)
本章参考文献	(5)
第二章 微机原理与接口技术实验	(6)
2.1 微机原理与接口技术课程主要知识点总结	(6)
2.2 汇编语言程序设计上机方法	(8)
2.2.1 汇编语言程序设计上机实验步骤	(8)
2.2.2 源程序的汇编、连接与调试方法	(9)
2.3 汇编语言程序设计实验	(14)
实验 1 8086/8088 寻址方式练习实验	(14)
实验 2 分支程序设计实验	(18)
实验 3 循环程序设计实验	(21)
实验 4 子程序设计实验	(23)
实验 5 汇编语言程序设计综合实验	(26)
2.4 实验平台简介	(55)
2.4.1 系统安装与使用	(56)
2.4.2 汇编语言程序设计实验开发集成环境	(59)
2.4.3 PCI 9054 配置寄存器	(60)
2.5 接口技术基础实验	(63)
实验 1 I/O 地址译码实验	(63)
实验 2 基本 I/O 端口设计实验	(64)
实验 3 可编程并行接口 8255 实验	(67)
实验 4 可编程定时器/计数器 8253 实验	(69)
实验 5 8 位存储器读写实验	(71)
实验 6 中断应用程序实验	(72)
2.6 接口技术综合设计实验	(76)
实验 1 ADC 0809 A/D 模数转换实验	(76)
实验 2 DAC 0832 D/A 转换实验	(78)
实验 3 竞赛抢答器综合设计实验	(80)
实验 4 数字式温度测量综合设计实验	(81)

实验 5 键盘显示控制综合设计实验	(84)
实验 6 16×16 LED 中文字幕移动实验	(89)
本章附录 A PCI BIOS 函数	(90)
本章附录 B PCI9054D 集成操作软件使用说明	(92)
本章附录 C Turbo Debugger 调试器的使用	(94)
本章参考文献	(100)
第三章 PIC 单片机原理与应用实验	(101)
3.1 PIC 单片机主要知识点总结	(101)
3.2 PIC 单片机资源介绍	(101)
3.2.1 8/16/32 位 PIC 单片机介绍	(101)
3.2.2 PIC 单片机开发工具资源介绍	(104)
3.3 MPLAB 集成开发环境软件的使用介绍	(106)
3.3.1 安装	(106)
3.3.2 使用介绍	(106)
3.4 ICD 2 在线调试器的使用介绍	(115)
3.4.1 MPLAB ICD 2 系统组件	(115)
3.4.2 模块接口连接	(115)
3.4.3 调试模式	(116)
3.4.4 ICD 2 的安装、配置和使用	(117)
3.5 APP009 实验板介绍	(124)
3.6 基础实验	(130)
实验 1 I/O 端口实验	(130)
实验 2 外部中断实验	(139)
实验 3 定时器实验	(141)
3.7 综合实验	(144)
实验 1 输入捕捉实验	(144)
实验 2 输出比较实验	(148)
实验 3 UART/SPI 串行通信接口实验	(150)
实验 4 10 位 A/D 转换器实验	(154)
本章参考文献	(158)
第四章 英飞凌单片机应用开发实验	(159)
4.1 英飞凌单片机课程的知识点总结	(159)
4.2 英飞凌单片机应用开发方法	(161)
4.3 英飞凌单片机基础实验	(161)
实验 1 通用 I/O 接口实验	(161)
实验 2 通用定时器实验	(163)
实验 3 异步串行接口通信实验	(167)
实验 4 脉宽调制(PWM)实验	(172)
实验 5 ADC 转换实验	(177)
实验 6 CAN 协议通信实验	(179)

4.4 英飞凌单片机综合应用实验	(189)
实验 1 串口定时器中断综合实验	(189)
实验 2 电子时钟设计实验	(190)
实验 3 可调亮度 LED 实验	(191)
实验 4 CAN 协议组网通信实验	(192)
附录 A 英飞凌单片机集成开发环境使用	(194)
附录 B 英飞凌单片机基本实验 DAVe 软件配置	(200)
本章参考文献	(216)
第五章 TI-DSP 技术基础实验	(217)
5.1 实验系统概述	(217)
5.1.1 实验系统构成	(217)
5.1.2 TMS320C5402 DSK 开发板	(218)
5.1.3 DSK 开发板信号处理框架	(219)
5.2 实验部分	(223)
实验 1 CCS 的使用及 I/O 口和中断仿真实验	(223)
实验 2 C54x 外中断编程实验	(224)
实验 3 C54x 定时器实验	(225)
实验 4 C54x 存储器及 FLASH 应用编程实验	(227)
实验 5 C54x 多通道缓冲串口应用实验	(231)
实验 6 C54x 异步通信接口 UART 实验	(234)
本章参考文献	(236)
第六章 嵌入式系统设计实验	(237)
6.1 嵌入式系统设计知识要点及难点	(237)
6.1.1 嵌入式系统设计硬件系统相关知识	(237)
6.1.2 嵌入式系统设计系统软件相关知识	(239)
6.1.3 嵌入式系统设计应用软件相关知识	(240)
6.2 嵌入式系统体系结构综述	(241)
6.2.1 常用嵌入式处理器体系结构	(241)
6.2.2 ARM 体系结构	(244)
6.2.3 Linux 操作系统体系结构	(247)
6.3 嵌入式开发平台及开发方式	(255)
6.3.1 嵌入式开发平台概述	(255)
6.3.2 嵌入式系统开发方式	(259)
6.4 嵌入式系统实验	(260)
实验 1 嵌入式开发平台的搭建实验	(260)
实验 2 嵌入式系统内核及文件系统镜像的定制实验	(271)
实验 3 综合实验(1)	(277)
实验 4 综合实验(2)	(283)
本章参考文献	(296)

第一章 微处理器导论

1.1 微处理器的发展历程

微处理器的发展可以追溯到 19 世纪初,受大不列颠皇家天文协会委托,Charles Babbage 于 1823 年开始研制一种可以编程的计算机——分析机,它是一种由蒸汽驱动的机械式计算机,可以存储 1000 个 20 位的十进制数及其程序,开创了可编程计算的新河。

二十世纪的三四十年代,算法理论方面的研究取得了突破性进展,涌现出了几个重要的计算模型。其中,英国科学家艾兰·图灵于 1937 年发表了《论可计算数及其在判定问题中的应用》论文,提出了图灵机模型。美国科学家冯·诺依曼根据图灵的设想,于 1946 年提出了计算机的基本原理——存储程序原理,并构造出计算机 EDSAC 和 EDVAC。这两台计算机分别于 1949 年和 1952 年在英国剑桥大学和美国宾州大学投入运行,为微处理器的发展奠定了基础。被称为冯·诺依曼计算机的 EDSAC 是第一台存储程序的计算机,采用了 18 000 多个电子管,导线长度超过 500 英里(1 英里=1.6093 千米),是一个不折不扣的庞然大物。1955 年开始出现的晶体管计算机大大地缩减了体积,计算速度也得到了提高。1959 年开始了集成电路的时代。1971 年 Intel 公司研制出第一种微处理器 Intel 4004,这也是世界上第一个微处理器,它只是一个 4 位的微处理器,可以寻址 4096 个 4 位宽存储单元位(半字节)。随后,Intel 公司于 1972 年推出了第一个 8 位微处理器 Intel 8008,在运行速度和存储数据方面有了较大的提高。稍后,Motorola 公司推出了 MC6800,从此开启了微处理器时代。这一阶段中,比较有代表性的产品还有 Intel 公司的 8080, MOS Technology 公司的 6502, Zilog 公司的 Z-80。

1978 年,Intel 推出了 8086 和 8088 微处理器,成为广泛应用的第一代个人计算机的 CPU。继后的更新换代产品有 80286、80386、80486,到 1993 年推出了 Pentium(80586)微处理器。“Pentium”在拉丁文中就是“五”的意思,Intel 公司还替它起了一个相当好听的中文名字——奔腾。最早期推出的 Pentium 时钟速度为 66 MHz 和 60 MHz。Intel 公司在 1996 年推出了第六代 x86 系列 CPU——PentiumPro,其时钟频率达到了 200 MHz。1998 年推出的 Pentium II,其时钟频率达到了 540MHz。1999 年推出了 Pentium III,其时钟频率达到了 733 MHz。2000 年推出了 Pentium 4,它属于第七代的 x86 微处理器,其时钟频率达到了 1.3 GHz。2003 年推出了 Pentium M,其时钟频率达到了 1.6 GHz。2007 年推出了 Pentium E2000,它采用了双核处理器结构,其时钟频率达到了 2.2 GHz。2011 年推出了 Pentium G620,其时钟频率达到了 2.6 GHz。

与此同时,生产微处理器的其它公司也相继推出同类产品,这种竞争局面推动了微处理器行业的发展。创办于 1969 年美国硅谷的 AMD 总公司,是集成电路供应商,专为电脑、通信及电子消费类市场供应各种芯片产品,它是唯一能与 Intel 竞争的 CPU 生产厂

家。AMD 公司的产品现在已经形成了以 Athlon XP 及 Duron 为核心的一系列产品。由于在 CPU 核心架构方面的优势,与 Intel 公司的 CPU 相比,同主频的 AMD 处理器具有更好的整体性能。创办于 1976 年的苹果公司(Apple Inc.)是美国的一家高科技公司,该公司的核心业务为研发生产电子产品,知名的产品有 Apple II、Macintosh 电脑、Macbook 笔记本电脑、iPod 音乐播放器、iTunes 商店、iMac 一体机、iPhone 手机和 iPad 平板电脑等。

1.2 单片机

微处理器具有运算和控制的功能,它通过外围电路构成数据总线、地址总线和控制总线,与外部设备进行连接,并形成计算机系统。例如,通过地址译码和数据驱动,与 RAM、ROM 连接,形成内存;通过 DMA 机构与磁盘驱动器连接,形成外存;通过端口与定时器连接,形成时钟系统;通过端口与 UART 连接,形成串行通信接口等。因此,微处理器需要设计复杂的外围电路,才能应用到实际的系统中。

将微处理器及其必要的外围电路(计算机系统)集成在同一片 IC 上,称为单片机,它是典型的嵌入式微控制器(Microcontroller Unit),简称 MCU。它相当于一个微型的计算机,并不只是完成某一个逻辑功能的芯片。与普通计算机相比,单片机只缺少了 I/O 设备,然而它的体积小、重量轻、价格便宜,为学习、应用和开发提供了便利条件。

单片机的硬件主要包括 CPU、ROM、RAM、定时/计数器、并行口、全双工串行口、ADC/DAC、SPI、I²C、ISP、IAP 等。单片机的主要特点如下:

- (1) 系统结构简单,使用方便,实现模块化;
- (2) 可靠性高,平均无故障时间达到 10^6 小时以上;
- (3) 处理功能强,速度快;
- (4) 低电压,低功耗,便于生产便携式产品;
- (5) 控制功能强;
- (6) 环境适应能力强。

1.3 嵌入式系统

嵌入式系统(Embedded System)是一种“完全嵌入受控器件内部,为特定应用而设计的专用计算机系统”。根据英国电气工程师协会的定义,嵌入式系统为控制、监视或辅助设备、机器正常运转或保证工厂正常运作的设备。与通用的个人计算机不同,嵌入式系统通常执行的是预先定义的具有特定要求的任务,故设计人员能够对它进行优化,减小尺寸,降低成本。

嵌入式系统的核心由一个或几个预先编程的微处理器或者单片机组成。与个人计算机能够运行各种通用软件不同,嵌入式系统上的软件通常是暂时不变的,所以经常称为“固件”。嵌入式系统是面向用户、面向产品、面向应用的,它必须与具体应用紧密结合,必须结合实际系统需求进行合理的设计。

嵌入式系统的几个重要特征如下:

- (1) 系统内核小。由于嵌入式系统一般是应用于小型电子装置的,系统资源相对有限,

所以内核较传统的操作系统要小得多。

(2) 专用性强。嵌入式系统的个性化很强,其中的软件系统和硬件的结合非常紧密,一般要针对硬件进行系统的移植,即使在同一品牌、同一系列的产品中也需要根据系统硬件的变化和增减不断进行修改。

(3) 系统精简。嵌入式系统一般没有系统软件和应用软件的明显区分,不要求其功能设计及实现上过于复杂,这样一方面利于控制系统成本,另一方面也利于实现系统安全。

(4) 实时性强。这是嵌入式系统软件的基本要求,而且软件要求固态存储,以提高速度;软件代码要求高质量和高可靠性。

1.4 数字信号处理器(DSP)

数字信号处理器(DSP)与微处理器(CPU)一起被公认为芯片工业两大核心技术。DSP是一种具有特殊结构的微处理器。DSP芯片的内部采用程序和数据分开的哈佛结构,具有专门的硬件乘法器,广泛采用流水线操作,提供特殊的DSP指令,可以用来快速地实现各种数字信号处理算法。与通用微处理器相比,DSP芯片的其他通用功能相对较弱些。

根据数字信号处理的要求,DSP芯片主要特点如下:

- (1) 在同一个指令周期内可以完成一次乘法和一次加法;
- (2) 程序和数据空间分开存放,这样访问指令和数据就可以同时进行;
- (3) 片内包含快速RAM,通常可通过独立的数据总线同时访问RAM和ROM;
- (4) 具有低开销或无开销循环及跳转的硬件支持;
- (5) 快速的中断处理和硬件I/O支持;
- (6) 具有多个硬件地址产生器;
- (7) 可以并行执行多个操作;
- (8) 支持流水线操作,使取指、译码和执行等操作可以重叠执行。

1.5 基于FPGA的模型机

自2000年以来,FPGA(现场可编程门阵列)技术获得了长足的发展,集成度得到了快速提升,其内部可编程资源丰富,外部连线很少、电路简单、便于控制。FPGA具有下列特点:

- (1) 采用FPGA设计专用电路时,用户不需要投片生产,就能得到ASIC芯片;
- (2) FPGA可用作全定制或半定制ASIC电路的中试样片;
- (3) FPGA内部有丰富的触发器和I/O引脚;
- (4) FPGA是ASIC电路中设计周期最短、开发费用最低、风险最小的器件之一;
- (5) FPGA采用高速CMOS工艺,功耗低,可以与CMOS、TTL电平兼容。

在实际应用中,经常采用FPGA为硬件平台,内嵌CPU模块,根据应用需要用户自行设计各种接口模块,以此降低硬件成本,提高设计灵活性和保密性。因此,我们可以通过设计一种基于FPGA的模型机(包含一种微处理器,也称为内核),并构建相应的指令系统,配合以适当的各种外围电路,如定时器、并行接口、串行接口等,形成一个面向应用的包含微处理器的系统,整个系统由一片FPGA和很少的外围器件构成。

1.6 微处理器的共性及新技术实验

30 多年前, 37 岁的美国科学家戈登·摩尔在一本杂志上发表了一篇文章, 大胆提出了后来为世人称之为“摩尔定律”的论述。摩尔在文中提出了这样的观点: 处理器(CPU)的功能和复杂性每年(其后期减慢为 18 个月)会增加一倍, 而成本却成比例地递减。在当时看来, 这种预测简直就是不可思议。但今天, 摩尔的预言非常接近现实。现在处理器的处理能力每隔一年半就会增长一倍, 比最初的 Intel 4004 已经有了百万倍的提高, 而价格却日趋下降。

摩尔在文章中还预见到: 电脑不仅可用来保存数据和作为强大的计算机器, 一旦处理器芯片的功能像大型机一样强劲, 价格又非常之低时, 电脑将会应用于社会的每个领域, 如家庭电脑、手持电话等都会问世。信息的集中处理将为分散处理所取代, 未来人们将生活在电脑无所不在的社会里。

摩尔预见了计算机时代的美好未来。科学需要预见, 产业需要眼光。这两者摩尔都具备了, 所以他参与创办了 Intel 公司, 身体力行地去验证了自己的预言, 并使 Intel 公司在数年间就发展成为领导电脑工业潮流的全球性企业。

然而, 不管是最初的 Intel 4004 微处理器, 还是今天的 Pentium 处理器, 乃至各类型号的单片机、数字信号处理器(DSP)、嵌入式系统等, 处理器内部结构及工作原理是基本相同的。在学习时, 只需掌握每种微处理器的共性, 掌握以微处理器为核心的智能电子信息系统的基本组成; 在实际应用中, 根据实际选型, 再进行拓展学习。这是提高学习效率的最佳方法。为此, 我们开设了“微机原理与接口技术实验”、“PIC 单片机原理与应用实验”、“英飞凌单片机应用开发实验”、“TI-DSP 技术基础实验”、“嵌入式系统设计实验”等系列微处理器类实验课程, 供同学们学习和选用。

微处理器是组成计算机系统的核心部件, 不管哪种型号的微处理器, 都应具有下列运算和控制功能: ① 进行算术和逻辑运算; ② 具有接收存储器和 I/O 接口来的数据, 发送数据给存储器和 I/O 接口的能力; ③ 可以暂存少量数据; ④ 能对指令进行寄存、译码并执行指令所规定的操作; ⑤ 能提供整个系统所需的定时和控制信号; ⑥ 可响应 I/O 设备发出的中断请求。

典型的微处理器内部结构如图 1.1 所示。

从 CPU 的内部结构可以看出, CPU 由四部分构成: 算术逻辑运算单元(ALU)、工作寄存器、控制器和 I/O 控制逻辑。

(1) 算术逻辑运算单元 ALU (Arithmetic/Logic Unit): 这是运算器的核心, 完成所有的运算操作。它是一个组合电路, 无记忆功能, 通常有两个输入端和一个输出端, 在控制信号的控制下可以完成不同的操作。

(2) 工作寄存器: 可以暂存寻址信息和计算过程中的中间结果。其中, 数据寄存器用于暂存操作数及中间结果, 地址寄存器用于暂存操作数的寻址信息。

(3) 控制器: 这是 CPU 的“指挥中心”, 完成指令的读入、寄存和译码, 并产生控制信号序列, 使 ALU 完成指定的操作。

(4) I/O 控制逻辑: 处理 I/O 操作。

在以微处理器为核心的应用系统中，通常除了包含一个 CPU 之外，还可能包含并行、串行、网络、键盘、显示、扬声器、USB 等接口电路，如图 1.2 所示。ROM 用于保存微处理器的系统程序；RAM 提供了程序所需的临时存储空间；硬盘系统可供存储不同的应用软件；软盘、光驱和 USB 接口用于提供软件更新、添加功能；键盘接口用于提供直接控制台输入；显示接口提供了直接控制台输出；扬声器接口可以使微处理器系统产生告警信号或提示声音；并行接口提供了多路数据的输入输出；串行接口提供了与其它设备的通信手段；网络接口提供了远程控制的可能性；PCI 槽为进一步扩充系统提供了便利。

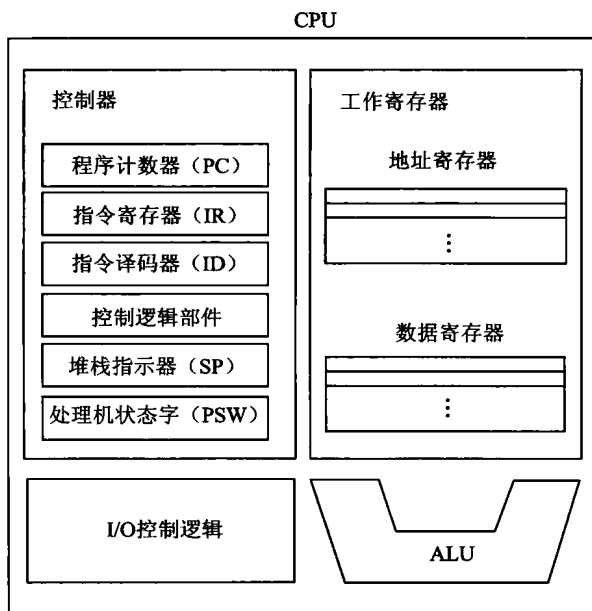


图 1.1 微处理器的内部结构

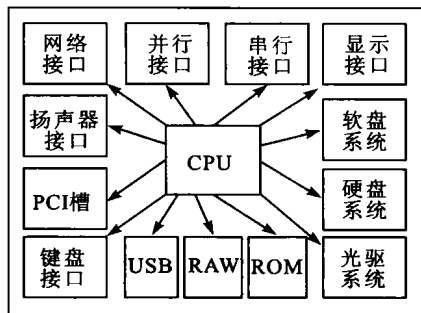


图 1.2 微处理器系统组成

本章参考文献

- [1] Barry B. Brey. Intel 微处理器[M]. 金惠华, 艾明晶, 等, 译. 北京: 机械工业出版社, 2010.
- [2] 李广军, 阎波, 等. 微处理器系统结构与嵌入式系统设计[M], 北京: 电子工业出版社, 2011.
- [3] 楼顺天, 周佳社. 微机原理与接口技术[M]. 北京: 科学出版社, 2006.
- [4] 周佳社, 石光明. 微机原理与应用实验指导书. 西安电子科技大学讲义, 2009.

第二章 微机原理与接口技术实验

2.1 微机原理与接口技术课程主要知识点总结

“微机原理与接口技术”课程采用楼顺天、周佳社编著的由科学出版社出版的《微机原理与接口技术》教材(普通高等教育国家级“十一五”规划教材),全书共 11 章,理论教学课时为 60 学时,重点学习第 1 章至第 10 章,第 11 章自学,不作为考试范围。本节首先对教材中每章的重要知识点及教学要求进行概括总结,以便于同学们学习、掌握。

第 1 章 数制与码制

- (1) 掌握各种数制之间的相互转换,理解字节、字等概念。
- (2) 掌握二进制数的加减运算、逻辑运算,并结合第三章加减运算指令学习,掌握 PSW 中状态位的设置原则。
- (3) 掌握有符号数的原码、补码表示方法;掌握用补码表示的数加、减运算后溢出(OV)的设置原则。
- (4) 掌握 BCD 数的两种表示方法,并结合第三章 BCD 数调整指令的学习,掌握 BCD 数运算后结果的调整方法。
- (5) 掌握 0~9、A~F、a~f、回车、换行、空格、Esc 等字符的 ASCII 码,并结合第三章逻辑运算指令、比较指令、条件转移指令等指令的学习,掌握大、小写字符之间的相互转换以及 0~9、A~F 数字与 0~9、A~F 字符之间的相互转换。

第 2 章 8086 CPU 结构与功能

- (1) 掌握总线、微机中的系统总线(三总线)以及每类总线所担负的功能;掌握 I/O 接口、I/O 端口以及微机系统对 I/O 端口的编址方式等概念;掌握 8086 CPU 的地址线条数以及寻址能力;掌握 8086 CPU 的数据线条数以及通过数据线能传送的数的范围。
- (2) 掌握微处理器的一般内部结构及其工作原理。
- (3) 掌握 8086 CPU 的内部结构以及 EU、BIU 的主要功能。
- (4) 掌握 8086 的 14 个寄存器及其功能,为后续 8086 汇编语言程序设计打下坚实的基础。
- (5) 掌握存储器为什么要分段、怎么分段,以及逻辑地址与物理地址之间的关系;掌握当前段、段超越前缀等概念,为后续 8086 汇编语言程序设计打下基础。
- (6) 掌握 8086 CPU 的存储器地址空间与 I/O 地址空间,以及 IBM-PC/XT 机中的 I/O 地址空间的概念,为后续存储器接口、I/O 接口设计打下基础。

第3章 8086 CPU 指令系统

- (1) 掌握机器语言、机器语言程序、汇编语言、汇编语言程序、汇编等概念。
- (2) 掌握 8086 汇编语言程序中语句的种类及其组成。
- (3) 掌握 8086 汇编语言程序的语句中常数(立即数)及算术表达式、属性表达式、PTR 属性临时修改表达式的应用。
- (4) 掌握标号、变量的定义以及定义后具有哪些属性,重点掌握变量定义后的存储器数据分配示意图的画法。
- (5) 掌握 8086 的寻址方式,重点掌握关于数据的寻址方式,并结合第三章图 3.3 掌握指令书写时的语法规则,会判断语句书写语法的正确性。
- (6) 原则上 3.4 节至 3.16 节介绍的每类指令的每一条都要掌握,但根据程序设计自己应理顺哪些指令是常用到的,应重点复习掌握这些常用指令。结合指令学习,逐步掌握 8086 汇编语言程序设计的方法。

第4章 汇编语言程序设计

- (1) 掌握 8086 汇编语言程序设计的步骤,掌握段定义、ASSUME 段寻址、END、EQU 等伪指令及其使用,熟练掌握 8086 汇编语言程序的完整结构。
- (2) 掌握 8086 汇编语言程序的汇编、连接以及上机方法与步骤。
- (3) 重点掌握中等难度的分支、循环程序设计,以及中等难度程序的分析。
- (4) 掌握子程序的设计。

第5章 总线及其形成

- (1) 掌握总线的定义、分类及其采用标准总线的优点。
- (2) 掌握 74LS373/74LS374、74LS244/74LS245 等常用接口芯片的使用。
- (3) 掌握 8086 CPU 的常用引脚功能,在此基础上掌握系统数据总线、地址总线、形成电路的设计。
- (4) 掌握基本总线周期,了解存储器或 I/O 的读/写总线时序。
- (5) 能够区别 8088 最小方式系统总线、8088 最大方式系统总线、8086 最小方式系统总线、8086 最大方式系统总线、PC/XT 系统总线,为后续接口电路设计打下基础。

第6章 存储器设计

- (1) 掌握存储器的分类和主要技术指标。
- (2) 根据系统需求的总容量和已知的存储器芯片,会计算所需片数。
- (3) 以 SRAM 6264、EPROM 2764 为例,掌握存储器的扩展设计,重点放在 8086 最小方式系统总线以及 8088 最大方式系统总线存储器的扩展设计上。

第7章 常用芯片的接口技术

- (1) 掌握 I/O 接口、I/O 端口以及 I/O 接口技术等概念,了解 I/O 接口的功能。

- (2) 掌握 I/O 接口地址的译码方法及其优缺点。
- (3) 掌握 I/O 端口的编址方式及其优缺点。
- (4) 掌握输入/输出的基本方法和基本模型, 重点掌握利用常用芯片设计无条件输入/输出以及查询法输入/输出的硬软件设计。

第 8 章 中断系统与可编程中断控制器 8259A

- (1) 掌握中断、中断源、中断识别(类型号)、中断向量、中断向量表等概念;掌握中断类型号与中断向量在向量表中存储单元地址的关系。
- (2) 掌握外部可屏蔽中断的响应过程。
- (3) 了解 8259A 的内部结构及其工作原理。掌握一片 8259A 能管理多少级外部可屏蔽中断源, 多片 8259A 级联能管理多少级外部可屏蔽中断源。
- (4) 了解 8259A 的初始化编程及简单的中断方式的应用。

第 9 章 定时/计数器 8253 应用设计

- (1) 掌握 8253 外部结构及其方式控制寄存器每位的含义, 进而掌握 8253 的初始化编程。
- (2) 掌握 8253 的工作方式, 重点掌握 8253 的方式 2、3 的应用。
- (3) 掌握 8253 与系统总线的连接方法及应用。

第 10 章 并行接口芯片 8255A 应用设计

- (1) 掌握 8255 外部结构及其方式控制寄存器每位的含义, 进而掌握 8255A 的初始化编程。
- (2) 掌握 8255A 的工作方式, 重点掌握 8255A 的方式 0 的应用。
- (3) 掌握 8255A 与系统总线的连接方法及应用。

2.2 汇编语言程序设计上机方法

2.2.1 汇编语言程序设计上机实验步骤

- (1) 用全屏幕编辑软件(如 MS - DOS 携带的 EDIT. EXE)编辑源程序, 扩展名必须为 .asm。
- (2) 用汇编程序(MASM. EXE)对源程序进行汇编。如果汇编正确, 则会在磁盘上形成 .obj 目标程序。如果源程序中有语法错误, 则不会在磁盘上形成 .obj 目标程序, 而是退出汇编, 在屏幕上显示出错信息及错误在源程序中的行号。这时, 用户应在全屏幕编辑软件下对源程序中的错误逐个进行检查修改。
- (3) 用连接程序(LINK. EXE)对目标程序进行连接。如果连接正确, 则会在磁盘上形成 .exe 可执行程序。
- (4) 用动态调试程序(DEBUG. EXE)对 .exe 可执行程序进行调试。

2.2.2 源程序的汇编、连接与调试方法

根据汇编语言源程序结构，可以利用任何一种编辑器编写源程序，可用的编辑器有 Windows 操作系统下的“记事本”、“UltraEdit”等，以及 DOS 操作系统下的“EDLIN”、“EDIT”等。不论采用何种编辑器，在保存文件时，文件扩展名必须是 asm。

要将以汇编语言编写的源程序转换成可以直接执行的二进制文件，还必须经过两个步骤：汇编源程序、连接目标程序。

这样，汇编语言的基本编程过程可以总结成表 2.1 所述步骤。

表 2.1 汇编语言的基本编程步骤

步 骤	输 入	涉及的程序	输 出
1. 编辑源程序	键盘	EDIT 等	myfile.asm
2. 汇编源程序	myfile.asm	MASM 或 TASM	myfile.obj
3. 连接程序	myfile.obj	LINK 或 TLINK	myfile.exe
4. 程序调试		DEBUG.EXE	

1. 编辑源程序

利用编辑程序“记事本”等编写源程序，其规则应该遵循 8086 CPU 指令系统的要求，源程序名的扩展名必须为 asm。例如 myfile.asm，下面以此为例加以说明。

2. 汇编源程序

汇编过程是利用汇编程序 MASM 对源程序文件进行汇编的，MASM 通过对源程序的扫描，找出其指令格式的错误、标号变量定义的错误(存在没有定义或重复定义标号变量)等，以便用户及时修改源程序。如果源程序汇编没有错误，则至多可以得到三个文件：目标文件(myfile.obj，必须产生)、列表文件(myfile.lst)和交叉索引文件(myfile.crf)。目标文件为指令、伪指令编译后的目标代码文件；列表文件中列出了程序代码、偏移地址以及出错信息，可以方便地分页打印装订；交叉索引文件中列出了程序所定义的所有标识符和标号及其引用情况。

汇编程序一般采用 MASM，其使用格式如下：

MASM source, object, list, crossref

其中，source 用于指定源程序文件名(可以不带扩展名)；object 用于指定所产生的目标文件名(也不带扩展名)；list 和 crossref 分别用于指定要产生的列表和交叉索引的文件名。一般情况下，可以只产生目标文件。

在实际使用时，可以采用下列简略方式：

- MASM myfile; ;表示对 myfile.asm 进行汇编，并只生成 myfile.obj
- MASM myfile ;没有命令末的分号，这时可按屏幕提示进行操作
- MASM myfile, list; ;表示要生成 myfile.obj 和 myfile.lst

3. 连接程序

连接程序 LINK 将汇编后产生的目标程序连接成可执行文件。其输入有两个文件：目