



普通高等教育“十一五”国家级规划教材

AVR单片机嵌入式系统 原理与应用实践 (第2版)

AVR



AVR

AVR

马 潮 编著



共享资料下载

http://www.ourdev.cn/bbs/bbs_list.jsp?bbs_id=1003



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS



普通高等教育“十一五”国家级规划教材

AVR 单片机嵌入式系统 原理与应用实践(第2版)

马 潮 编著

北京航空航天大学出版社

内 容 简 介

本书以 ATMEL 公司 AVR 单片机 ATmega16 为蓝本,由浅入深,软硬结合,全面系统地介绍基于单片机的嵌入式系统的原理与结构,开发环境与工具,各种接口与功能单元应用的硬件设计思想和软件编写方法。

本书以夯实基础,面向应用,理论与实践、方法与实现紧密结合为主线展开,根据 AVR 的运行速度快、资源丰富、功能强大、以串行扩展为主等显著特点,采用 C 语言作为系统软件开发平台,以由简到繁、循序渐进、螺旋式上升的方式进行编排。在讲解原理和设计方法的同时,还穿插介绍相关的经验、技巧与注意事项,有很强的实用性和指导性。各章还配有问题思考、实践练习及相关参考文献和资料,供课后复习、实践、开拓知识面及进一步深入研究、提高用。

在作者 AVR 专栏中共享相关的资料,包括:书中所有例程源代码、芯片技术资料、相关技术规范和协议、大量参考文献和应用设计参考。书中还介绍了适合初、中级水平学习人员使用,具有模块独立化、简单、开放、灵活等特点的“AVR-51 多功能实验开发板”,既配合本书的教学实践,又适用于产品的前期开发。

本书可作为高等院校电子、自动化、仪器仪表和计算机等相关专业基于单片机的嵌入式系统课程的教材,也可作为 AVR 单片机的培训教材,供相关技术人员学习和参考。

图书在版编目(CIP)数据

AVR 单片机嵌入式系统原理与应用实践 / 马潮编著
—2 版. —北京:北京航空航天大学出版社,2011.8
ISBN 978-7-5124-0434-2

I. ①A… II. ①马… III. ①单片微型计算机—高等学校—教材 IV. ①TP368.1

中国版本图书馆 CIP 数据核字(2011)第 084447 号

版权所有,侵权必究。

AVR 单片机嵌入式系统原理与应用实践(第 2 版)

马 潮 编著
责任编辑 刘 星

*

北京航空航天大学出版社出版发行
北京市海淀区学院路 37 号(邮编 100191) <http://www.buaapress.com.cn>
发行部电话:(010)82317024 传真:(010)82328026
读者信箱:emsbook@gmail.com 邮购电话:(010)82316936
涿州市新华印刷有限公司印装 各地书店经销

*

开本:787×960 1/16 印张:37.25 字数:834 千字
2011 年 8 月第 2 版 2011 年 8 月第 1 次印刷 印数:5 000 册
ISBN 978-7-5124-0434-2 定价:56.00 元

第 2 版前言

被列入普通高等教育“十一五”国家级规划教材的《AVR 单片机嵌入式系统原理与应用实践》一书,自 2007 年 10 月由北京航空航天大学出版社出版后,分别在 08、09、10 年加印 3 次,截止到 2010 年 9 月总印数达到 14 000 册,在此类专业教材中表现相当突出。2010 年 9 月,美国 ATMEL 公司在上海华东师范大学 AVR 实验室举办了“ATMEL 2010 全国大学教师年会”,国内清华大学、交通大学等 20 多所著名高校的教师参加了会议和培训,预计会有更多的学校将采用 AVR 作为 8 位嵌入式系统教学的硬件平台,未来对此类教材的需求量还会增加。

作者非常感谢购买过本书的读者,尤其是那些能够通过网络,在作者的 AVR 专栏组中(http://www.ourdev.cn/bbs/bbs_list.jsp?xcfrom=302&bbs_id=1003)提出问题讨论并给出建议或指出错误的网友读者。也正是在他们的帮助支持下,使作者在前 3 次的加印过程中做了 100 多处的笔误修正和印刷勘误。由于半导体器件和电子产品的开发应用是发展、变化和更新最快的技术领域,因此作者根据最近几年 ATMEL 公司 AVR 的发展变化、开发平台软件和开发工具的更新等相关资料,以及近几年作者本人在学校、公司从事教学与培训中的经验和积累,决定对本书进行修订推出第 2 版。

第 2 版中主要对 AVR 的开发平台和工具的介绍及使用进行了更新,并将书中全部例程代码在新版 CVAVR 开发平台下进行了测试和整理。另外还根据部分读者的建议,增加了一个基于 AVR 实际应用实例的设计与实现的介绍(第 19 章)。尽管作者本人设计和开发过许多实际的应用项目和产品,但考虑到本书主要面对的读者还是初学者,通常不可能具备和掌握开发各种实际项目或产品所涉及的相关知识和基础,故增加应用实例的选择还是局限于以 AVR 本身的功能使用上,充分发挥其本身的最大效率,以及体现以软件为显著特点的“简易 WAVE 播放器”。尽管这还是一个 8 位系统的应用实例,但其所涉及的 SD/MMC 卡读/写操作、FAT 文件系统的实现等相关协议与技术,已经与使用 32 位系统没有任何的区别。

在本书的再版过程中,对第 1 版所配光盘内容也做了较多的改变,增加了大量与实践操作相关的内容,如编程工具的制作与使用、系统代码的仿真调试、AVR“锁死”的解救等。这部分的编写与修改,主要是由作者的学生,目前在华东师范大学通信工程系就读硕士,同时也是 INTEL 亚太研发中心的优秀实习生周万程同学负责完成的。所有配合本书的电子文档和软件平台资料,读者可以访问中国电子开发网(<http://www.ourdev.cn>),在作者的专栏组中免费下载(http://www.ourdev.cn/bbs/bbs_list.jsp?bbs_id=1003),也可以与作者进行交流。

第2版前言

虽然作者多年从事单片机与嵌入式系统应用的教学和实际产品的研发工作,也力求从适合教学、面向应用、强化实践等方面写好本书,但鉴于技术的不断发展更新和个人水平的局限,书中难免存在不足和错误之处,敬请读者批评指正。

最后,真诚地感谢北京航空航天大学出版社和该社相关的工作人员,以及中国电子开发网(www.ourdev.cn),还有众多不知名的网友读者对本书再版工作的支持和帮助。

作者
2011年5月

第 1 版前言

国内高等院校的单片机教学与研究已经走过了 20 多年的历程：从最早的 Z80 单板机到 MCS-51 的流行；从 8 位 AVR 到 32 位 ARM 的推广演变；从人工编译到用紫外线擦除 EPROM；从 ISP 在线编程到计算机模拟仿真。作者有幸亲身经历并充分体验了 20 多年来单片机技术飞速发展的过程，同时也深感高校的教学内容和教材已跟不上技术的发展和市场的需要。

目前大部分学校开设的“单片机原理及嵌入式应用”一类的课程还是以 20 世纪 80 年代开始流行的 MCS-51 系列单片机为蓝本，以汇编语言（或 C 语言）为编程工具，以并行扩展为核心，讲述单片机的接口技术及单片机的嵌入式应用。尽管 MCS-51 在实际应用中还占据着相当大的市场，但随着微电子技术和信息技术的迅猛发展，以及各种新型数据传输接口技术的出现和新器件的推出，传统的 MCS-51 由于自身结构原因，在数据通信和系统扩展的能力方面开始显得捉襟见肘，与各种新技术的发展和应用产生脱节。

AVR 是最近 10 年间发展起来的新型的、基于增强型 RISC 结构的单片机。AVR 在运行速度，内存容量，内部功能模块的集成化（SOC），以串行接口为主的外围扩展，适合使用高级语言编程，以及在开发技术和仿真调试技术的应用等诸多方面都比 MCS-51 先进，比较充分和全面地代表了当前 8 位单片机和嵌入式应用技术的发展方向。AVR 单片机由 ATMEL 公司于 1997 年强势推出后，很快就得到了市场的认可，并迅速推广开来，成为 MCS-51 强有力的竞争者。与此同时，国外的许多高校，如美国的麻省理工学院、斯坦福电子工程系、加州大学伯克利分校工程学院、普林斯顿大学计算机学院、耶鲁大学工程系、康乃尔大学、卡内基梅隆大学、加拿大的多伦多计算机工程系等，也纷纷跟进市场需求，更新课程内容体系，在开设的相关课程中选择和使用 AVR。

AVR 从最早的 AT90S1200 发展到现在的 megaAVR 和 tinyAVR，构成了一个完整的产品系列，在技术和性能方面已经相当完美和成熟。1999 年，ATMEL 公司与华东师范大学电子系合作，建立了国内第一个 AVR 实验室，并首次在高校中开设了 AVR 课程。作为实验室的技术负责人，作者一直从事 AVR 教学课程的建设，编辑、翻译、出版过 AVR 的器件手册和应用参考书，参与并负责 AVR Studio 开发平台的英文帮助汉化翻译工作，同时也在多年的电子系统和产品的设计研发过程中，积累了丰富的 AVR 应用经验。

2002 年，作者申请获得了华东师范大学教务处教学建设基金的资助，开始对单片机课程的全面改革和建设。在参考大量国外同类课程的基础上，结合国内实际情况，编写了 AVR 的教学讲义，设计制作了配合教学并能有效提高学生的应用动手实践能力和素质及性价比极高的教学实验（实践）板，并在多年的教学实践中，不断改进和完善它们。

本书是作者在多年教学实践与改革及大量实际应用经验积累的基础上，作为“单片机原理及嵌入式应用课程”的教材而编写的。本书已经通过国家教育部评审，被列入“普通高等教育

“十一五”国家级规划教材”。作者希望它能为促进国内单片机教学的发展,为培养更多高质量、应用型的硬件工程师做出微薄的贡献。

本书以 ATMEGA 公司新一代 AVR 系列单片机中的 ATmega16 为蓝本,由浅入深,软硬结合,全面系统地介绍了以 AVR 为核心构成的单片机嵌入式系统的原理与结构,开发环境与工具,各种接口与功能单元应用的硬件设计思想和软件编写方法,以及系统调试与仿真等内容。本书在结构编排和内容选择方面与一般传统单片机教材有所不同。全书以夯实基础,面向应用,理论与实践、方法与实现紧密结合为主线展开,在充分发挥 AVR 的运行速度快,内部资源丰富,功能强大等显著特点的基础上,结合最新嵌入式系统开发和应用技术的发展,遵照单片机嵌入式系统研发的基本步骤和思路,采用从简单到复杂、循序渐进、螺旋式上升的方式进行编排。

全书共分 4 篇。第 1 篇为基础入门篇,着重介绍了与单片机相关的基础知识,AVR 的基本结构、指令系统和基本工作原理,以及开发环境的建立、工具的使用。在本篇的最后,安排一个简单的实例,指导读者进入动手实践,让 AVR 先“动”起来,为下一步的学习和实践做好准备。第 2 篇为基本接口单元的应用设计,将重点放在 AVR 基本接口和功能模块的介绍和应用设计上,主要是 I/O 口的应用、中断、定时器、A/D 等。这些内容也是所有单片机应用的基础。第 3 篇充分体现了外部扩展以串行总线接口为主的技术发展趋势,用了 5 章的篇幅,对串行通信和串行接口进行了详细的介绍,包括 USART、SPI、I²C 等。尽管这些串行接口的硬件连接非常简单,但重要的是通过对串行通信基础的学习,加深对各种协议的了解,建立程序设计思想并在编程能力上得到提高。第 4 篇中对 AVR 本身所具备的特点及使用做了介绍,这些特点实际上也代表了新技术的发展方向,在实际应用中是不可缺少的环节。本篇最后的“迎奥运倒计时时钟设计实例”是对全书的总结和实战演练。说实话,绝大部分本科生并不具备研发“产品”的能力,能够通过教材的指导,独立完成这样的设计,已经是圆满地完成了本课程的学习任务了。

本书在仔细讲解各章主要内容和基本应用设计方法的同时,还穿插介绍了相关的应用经验、技巧与注意事项,有很强的实用性和指导性。书中主要采用 C 语言作为系统软件开发平台,并把外部扩展技术的重点放在了以介绍 SPI、I²C 等串行接口的原理和应用方面,这不仅符合 AVR 本身的特点,也是紧跟技术的发展方向,同时也更加贴近实际应用。本书配有较多的面向系统的实例,例程代码多数取自实际的应用系统。各章都配有一些思考和实践练习题及相关的参考文献,供课后复习、实践、开拓知识面及进一步深入研究和提高用。

本书附带的光盘提供了 150 MB 的内容,收录了书中所有例程的源代码,所使用芯片的技术资料,相关的技术规范和协议,以及大量参考文献和应用设计参考资料。

本着能够使读者在硬件设计和软件编程方面全方位地得到真正的训练和提高的目的,作者还设计了一套适合初中级水平学习人员使用的,具有简单、独特、模块独立化、开放、灵活等特点的“AVR-51 多功能实验开发板”。它不仅适用于配合本书教学实验,而且也适用于设计开发人员作为产品开发前期使用的开发板。该实验板由国内的“我们的 AVR”网站批量生产,读者可以从该网站(<http://www.ourdev.com>)邮购完整产品,也可购买配套散件自行焊接制作。从动手能力的培养和提高角度出发,作者极力推荐后一种方式。本书附带的光盘中提供了制作该板的资料和图纸。

目前,我们(包括作者)都面临一个非常现实的问题:如何才能上好“单片机原理及嵌入式应用”课程?该课程最终应该达到什么样的效果?

作者从多年教学经验和应用实践中体会到,“单片机原理及嵌入式应用”课程应该朝着承前、注重应用能力和启后 3 个方向发展。

承前是本课程内容所决定的。本课程内容涵盖了模拟电路、数字电路、计算机原理、汇编(8086)、C 语言程序设计、计算机应用、专业英语能力等多项知识点和能力,尽管学生在一、二年级已经学习过这些课程,但往往只局限在对基础理论知识的掌握上,基本上不具备实际应用能力,而且各个知识点也是相互独立的。因此,在课程中需要指导学生对前面的基础知识进行复习和回顾,更重要的是通过课程的学习,培养学生能把各方面的基础知识进行综合,并在实际中应用的能力。

注重动手实践能力的培养是由本课程的性质所决定的。单片机嵌入式系统的教学绝不能纸上谈兵,不动手是学不会的。让学生用不动手的办法去学,用纸上谈兵的办法去学,就会越学越觉得枯燥,越学越觉得没有信心。因此,教学中要让学生始终有一种奋发向上的精神,要让学生愿学、爱学。这里的关键就是强化实践。有的人把强化实践理解歪了,看低了,好像实践就是随便动动手,简单操作一下,甚至是给个程序下载就完事了。其实不然,特别是对电子工程师来讲,动手和动脑应该是联系在一起的。当一个电子工程师是很不容易的,既要有理论,又要有实践,而且理论要紧密联系实践。实践是非常重要的,一个学生的能力和水平最后都要通过实践表现出来,让实践来说话。只有学生对基础知识有了更深的认识,同时具备了相应的实际应用能力,才能达到和满足社会的用人需求。

电子技术是一门快速发展的学科,因此“单片机原理及嵌入式应用”课程还应该为向更高层次的嵌入式(32 位 ARM、DSP)系统学习和应用打好基础,这就是该课程的启后功能。目前在一些高校中已经尝试直接对本科学生开设基于 32 位 ARM 的嵌入式系统课程,对于以偏硬件为主的相关专业来讲,作者认为这是一种操之过急和拔苗助长的行为。学习、掌握 32 位嵌入式系统的应用,不能局限于利用现成的 ARM 板,在别人移植的 OS 上编写一些用户应用程序,这与在 PC 机上学习编程没有太大的区别。要真正掌握 32 位嵌入式系统的应用,不仅对硬件和软件的要求非常高,同时也需要掌握大量相关的其他专业方面的知识,如数字信号处理、数字语音图像处理、USB 以及各种网络协议等,这些知识对于本科学生来讲一般是不具备的,更适合研究生阶段的学习。因此在本科阶段,开设基于新型 8 位单片机嵌入式系统的课程更为合适,能够比较好地起到承前、应用能力和启后的作用,符合循序渐进、螺旋式上升的教育规律。

作者非常欣赏和推崇康乃尔大学的《微控制器设计》(Designing with Microcontrollers)课程的设置理念和教学方法(<http://instruct1.cit.cornell.edu/courses/ee476/>),修读该课程的学生每年都会推出几十项课程设计作品(Course Project),很多作品不是简单的芯片应用,而是充满创意,连学生的设计报告都非常生动和可爱。这些课程设计作品并不比现在国内两年一次的全国大学生电子设计竞赛的作品逊色,而且它是真实的学生能力和水平的体现,同时也反映了国内大学生和国外大学生在动手实践能力方面的差距。

对比康乃尔大学的课程,从一个侧面看到了国内在教学上的差距,更重要的是在教学理念、教学方法上的落后,而且这还不仅仅是发生在某一门课程上的问题。作者在 AVR 的教学

实践中,一直以康乃尔大学的《微处理器设计》作为努力的方向,尝试使用与传统教学不同的理念和方法对单片机课程的教学进行改革,但到目前为止,效果并不令人十分满意。因为仅仅通过一门课程,是那么的无能为力,根本不能改变目前国内教学的应试教育现状的。

最后要指出的是,开设“单片机原理及嵌入式应用”一类的课程,对教师的要求比较高,教师不仅要具备书本的理论知识,而且必须有一定的实践应用经验,了解目前嵌入式系统技术的发展。只是仅仅在课堂上介绍 AVR 根本不行,必须要能够随时回答和解决学生提出的问题,例如: C 语言的语法和程序调试的问题,外部上拉电阻的作用与阻值计算问题,英文器件手册上的一段说明解释,甚至计算机环境本身的设置和使用的问题。学生在实践中发生的问题 80% 不在 AVR 的本身,而是硬件连接错误,程序不会编写,环境没有设置等。这对教师是一个挑战,也是上好本课程比较特殊和困难的重要原因之一。

从编写讲义到正式成为教材出版,本书前前后后、断断续续地花费了 3 年多的时间。尽管作者在这期间也编写出版过其他的 AVR 参考书,但对于出版教材还是力求精益求精,不敢马虎。从开始使用 AT90S8515 到最后确定采用 ATmega16 为蓝本;从 AVR 汇编到 BASCOM - AVR,最后选择 CVAVR;从以并行总线为主的扩展转到以串行接口为主的扩展,讲义经过了多次大的改动。就是简单的 AVR - 51 多功能实验板也进行了 3 次修改。要感谢 ATMEL 公司北京代表处总经理施膺先生和 ATMEL 上海联络处贾必有、尹恩龙先生,他们及时地提供了 AVR 的最新技术资料 and 芯片,并给予了 AVR 实验室大力的帮助。感谢北京航空航天大学出版社,在申请“普通高等教育“十一五”国家级规划教材”和本书的出版过程中的大力支持。感谢华东师范大学电子系实验中心主任刘中元和工程师陈慧产先生,他们和作者一起共同参与了华东师范大学电子系 & ATMEL 联合实验室的建设和发展,并在技术、管理及课程建设上给予支持。还要感谢我的夫人和女儿,夫人在繁忙的工作期间,承担了更多的家务;而我的女儿则依靠自己的努力,在 2007 年高考中以优异的成绩被华东师范大学录取。正是由于这些无形的支持,才使我有更充裕的时间和精力投入工作。

最后,尤其要感谢美国 ATMEL 公司副总裁、华东师范大学顾问教授爱新觉罗·任启先生。正是由于任启先生的无私和鼎力赞助,才促成了华东师范大学电子系 & ATMEL 联合实验室的建立。从实验室 1999 年底规划动工到今天 7 年多的时间里,每一个发展环节都得到了任启先生的关心和指导。ATMEL 公司和任启先生不仅为实验室提供了上百万元人民币的实验仪器设备和建设经费,还非常重视大学生的动手能力和实践能力的培养,在电子系设立了“创新研究奖学金”,以鼓励和资助大学生更多的参加实践和 Course Project 的设计制作活动,树立创新精神。

虽然作者多年从事单片机嵌入式系统应用的教学和实际产品的研发工作,也力求从适合教学,面向应用,强化实践等方面写好本书,但鉴于技术的不断发展更新和个人水平有限,书中难免存在不足和错误之处,敬请读者批评指正。

作者

2007 年 8 月

目 录

第 1 篇 基础与入门

第 1 章 单片机嵌入式系统概述	2
1.1 嵌入式系统简介	2
1.1.1 嵌入式计算机系统	2
1.1.2 单片机嵌入式系统	4
1.1.3 单片机的发展历史	5
1.1.4 单片机的发展趋势	6
1.2 单片机嵌入式系统的结构与应用领域	9
1.2.1 单片机嵌入式系统的结构	9
1.2.2 单片机嵌入式系统的应用领域	10
1.3 AVR 单片机简介	11
1.3.1 ATMEL 公司的单片机简介	11
1.3.2 AVR 单片机的主要特点	13
1.3.3 AVR 系列单片机简介	14
1.3.4 AVR 与 51 单片机	18
思考与练习	19
第 2 章 AVR 单片机的基本结构	20
2.1 单片机的基本组成	20
2.1.1 单片机的基本组成结构	20
2.1.2 单片机的基本单元与作用	21
2.2 ATmega16 单片机的组成	24
2.2.1 AVR 单片机的内核结构	24
2.2.2 ATmega16 的特点	26
2.2.3 ATmega16 的外部引脚与封装	27
2.3 ATmega16 单片机的内部结构	29
2.3.1 中央处理器 CPU	29
2.3.2 系统时钟部件	32
2.3.3 CPU 的工作时序	33

2.3.4	存储器	35
2.3.5	I/O 端口	35
2.4	存储器结构和地址空间	35
2.4.1	支持 ISP 的 Flash 程序存储器	35
2.4.2	数据存储器 SRAM 空间	36
2.4.3	内部 EEPROM 存储器	36
2.5	通用寄存器组与 I/O 寄存器	37
2.5.1	通用寄存器组	37
2.5.2	I/O 寄存器	38
2.5.3	状态寄存器和堆栈指针寄存器	41
2.6	ATmega16 单片机的工作状态	43
2.6.1	AVR 单片机最小系统	44
2.6.2	AVR 的复位源和复位方式	45
2.6.3	对 AVR 的编程下载	49
2.6.4	ATmega16 的熔丝位	50
2.6.5	AVR 单片机的工作状态	52
2.6.6	支持 ISP 编程的最小系统设计	53
2.7	AVR 单片机内部资源的扩展和剪裁	55
	思考与练习	55
第 3 章	AVR 的指令与汇编系统	57
3.1	ATmega16 指令综述	57
3.1.1	指令格式及 3 种表示方式	58
3.1.2	AVR 指令系统中使用的符号	59
3.1.3	AVR 指令的寻址方式和寻址空间	60
3.1.4	AVR 指令操作结果对标志位的影响	66
3.2	算术和逻辑指令	66
3.2.1	加法指令	66
3.2.2	减法指令	67
3.2.3	取反码指令	68
3.2.4	取补码指令	68
3.2.5	比较指令	69
3.2.6	逻辑“与”指令	69
3.2.7	逻辑“或”指令	70
3.2.8	逻辑“异或”指令	71
3.2.9	乘法指令	71

3.3	跳转指令	73
3.3.1	无条件跳转指令	73
3.3.2	条件跳转指令	74
3.3.3	子程序调用和返回指令	80
3.4	数据传送指令	81
3.4.1	直接寻址数据传送指令	81
3.4.2	间接寻址数据传送指令	82
3.4.3	从程序存储器中取数装入寄存器指令	84
3.4.4	写程序存储器指令	86
3.4.5	I/O 口数据传送指令	86
3.4.6	堆栈操作指令	87
3.5	位操作和位测试指令	87
3.5.1	带进位逻辑操作指令	87
3.5.2	位变量传送指令	88
3.5.3	位变量修改指令	89
3.6	MCU 控制指令	92
3.7	AVR 汇编语言系统	92
3.7.1	汇编语言语句格式	93
3.7.2	汇编器伪指令	94
3.7.3	表达式	98
3.7.4	器件定义头文件 m16def.inc	100
第4章	AVR 单片机的系统设计与开发工具	103
4.1	单片机嵌入式应用系统设计	103
4.1.1	单片机嵌入式系统开发所需的基础知识和技能	103
4.1.2	单片机嵌入式系统开发过程	105
4.2	单片机嵌入式系统的开发工具与环境	108
4.2.1	单片机嵌入式系统的程序设计语言	108
4.2.2	单片机嵌入式系统的开发软件平台	109
4.2.3	单片机嵌入式系统的硬件开发工具	110
4.2.4	AVR 单片机嵌入式系统的软件开发平台	113
4.2.5	AVR 实验开发板	118
4.2.6	AVR 编程调试工具	124
4.3	自制 ISP 下载电缆	127
4.3.1	STK200/300 并口 ISP 下载电缆	128
4.3.2	USB - ISP 下载电缆	129

4.4 AVR 开发环境的建立	131
4.4.1 AVR 研发型开发环境	132
4.4.2 AVR 学习型实验开发环境	132
思考与练习	133
第5章 实战练习(一)	134
5.1 秒节拍显示器系统的设计	134
5.1.1 秒节拍显示器硬件设计	134
5.1.2 秒节拍显示器软件设计思路	136
5.1.3 秒节拍显示器汇编源程序	136
5.1.4 通用延时子程序分析	137
5.2 AVR Studio 汇编语言集成开发环境的使用	138
5.2.1 AVR Studio 的安装和其他辅助工具的安装	139
5.2.2 系统工程文件与 AVR 汇编源程序文件的建立、编译	139
5.2.3 使用软件模拟仿真调试程序	143
5.2.4 下载执行代码实际运行	147
5.3 CVAVR + AVR Studio——高级语言集成开发环境的使用	147
5.3.1 秒节拍显示器的高级 C 语言源程序代码	148
5.3.2 系统工程文件与源程序文件的建立、编译	149
5.3.3 在 CVAVR 中使用 AVR Studio 进行软件仿真调试程序	152
5.4 AVR 熔丝位的设置和执行代码下载	156
5.4.1 AVR-51 多功能板的硬件连接	156
5.4.2 AVR 熔丝位的配置	157
5.4.3 执行代码文件的下载	161
5.5 一个比较复杂的 AVR 汇编语言实例	162
5.5.1 系统功能与硬件设计	162
5.5.2 AVR 汇编源代码	163
思考与练习	171

第2篇 基本功能单元的应用

第6章 通用 I/O 接口的基本结构与输出应用	173
6.1 通用 I/O 接口的基本结构与特性	173
6.1.1 I/O 接口的基本结构	173
6.1.2 I/O 接口寄存器	176
6.1.3 通用数字 I/O 接口的设置与编程	177
6.2 通用 I/O 接口的输出应用	180

6.2.1	通用 I/O 接口的输出设计要点	180
6.2.2	LED 发光二极管的控制	180
6.2.3	继电器控制	183
6.2.4	步进电机控制	184
6.3	LED 数码显示器的应用	186
6.3.1	单个 LED 数码管控制	186
6.3.2	多位 LED 数码管显示	189
6.3.3	点阵 LED 显示控制	197
6.4	LCD 液晶显示器的应用	199
6.4.1	LCD 的特点与分类	200
6.4.2	通用点阵字符 LCD 显示器的应用	200
	思考与练习	206
第 7 章	中断系统与基本应用	207
7.1	中断的基本概念	207
7.1.1	中断处理过程	207
7.1.2	中断源、中断信号和中断向量	208
7.1.3	中断优先级和中断嵌套	209
7.1.4	中断响应条件与中断控制	210
7.2	ATmega16 的中断系统	212
7.2.1	ATmega16 的中断源和中断向量	212
7.2.2	ATmega16 的中断控制	214
7.2.3	AVR 的中断响应过程	216
7.3	中断服务程序的编写	218
7.3.1	汇编语言 AVR 中断程序的编写	218
7.3.2	CodeVision 中断程序的编写	223
7.4	ATmega16 的外部中断	226
7.4.1	外部中断的触发方式和特点	226
7.4.2	与外部中断相关的寄存器和标志位	227
7.5	外部中断应用实例	230
	思考与练习	241
第 8 章	定时/计数器的结构与应用	242
8.1	定时/计数器的结构	242
8.1.1	8 位定时/计数器 T/C0 的结构	243
8.1.2	8 位 T/C0 的工作模式	251
8.1.3	8 位 T/C0 的计数工作时序	255

8.2	8 位定时/计数器 T/C0 的应用	257
8.2.1	外部事件计数器	257
8.2.2	定时器应用设计	262
8.3	PWM 脉宽调制波的产生和应用	267
8.3.1	PWM 脉宽调制波	267
8.3.2	基于比较匹配输出的脉冲宽度调制 PWM	269
8.4	16 位定时/计数器 T/C1 的应用	272
8.4.1	16 位 T/C1 增强功能介绍	273
8.4.2	16 位 T/C1 应用示例	276
	思考与练习	281
第 9 章	键盘输入接口与状态机设计	282
9.1	通用 I/O 数字输入接口设计	282
9.1.1	I/O 输入接口硬件设计要点	282
9.1.2	I/O 输入接口软件设计要点	284
9.2	基于状态机的按键输入接口设计	285
9.2.1	简单的按键输入硬件接口与分析	285
9.2.2	基于状态机的按键输入软件接口设计	287
9.3	矩阵键盘输入接口设计	300
9.3.1	矩阵键盘的工作原理和扫描确认方式	301
9.3.2	定时扫描方式的键盘接口程序	302
	思考与练习	310
第 10 章	模拟比较器和 ADC 接口	311
10.1	模拟比较器	311
10.1.1	与模拟比较器相关的寄存器和标志位	311
10.1.2	模拟比较器的应用设计	314
10.2	模/数转换器 ADC	315
10.2.1	10 位 ADC 结构	316
10.2.2	与 ADC 相关的 I/O 寄存器	318
10.2.3	ADC 应用设计要点	322
10.2.4	ADC 的应用设计	324
10.2.5	ADC 应用设计的深入讨论	327
	思考与练习	330
第 11 章	实战练习(二)	331
11.1	频率测量和简单频率计的设计与实现	331
11.1.1	频率测量原理	331

11.1.2 测频法测量频率	332
11.1.3 测周法测量频率	335
11.1.4 频率测量小结	339
11.2 基于 T/C1 捕捉功能实现高精度的周期测量	339
11.3 带校时和音乐报时功能时钟的设计与实现	346
思考与练习	353

第3篇 串行接口与通信

第12章 串行数据接口概述	355
12.1 串行接口与串行通信基础知识	356
12.1.1 并行传输	356
12.1.2 串行传输	357
12.1.3 常见的串行传输和通信接口	358
12.2 数字 I/O 口的串行扩展	359
12.2.1 串行扩展并行输出口	360
12.2.2 串行扩展并行输入口	363
12.2.3 数字 I/O 口串行扩展设计要点	365
思考与练习	365
第13章 异步通信与 USART 接口基础	367
13.1 异步传输的基本概念	367
13.1.1 异步传输的字符数据帧格式	367
13.1.2 异步通信	369
13.2 AVR 的异步传输接口 USART	370
13.2.1 概 述	371
13.2.2 串行时钟发生器	372
13.2.3 数据帧格式	374
13.2.4 USART 寄存器	375
13.2.5 串行通信波特率的设置与偏差	380
13.3 USART 的基本操作	383
13.3.1 USART 的初始化	383
13.3.2 数据发送	384
13.3.3 数据接收	387
13.4 基于 USART 接口基本通信的实现与测试	390
13.4.1 USART 的数据发送和接收	390
13.4.2 RS-232C 总线标准介绍	394

13.4.3	AVR 系统的 RS-232C 传输接口的实现与测试	396
13.4.4	异步通信中易产生的问题与 AVR 系统时钟的选择	401
13.5	AVR USART 接口特性的进一步说明	402
13.5.1	使用独立的高精度波特率发生器	402
13.5.2	数据接收采用 3 级接收缓冲器结构	402
13.5.3	硬件自动处理校验位及错误检测	403
13.5.4	USART 数据接收的硬件扫描检测和接收时序	403
	思考与练习	405
第 14 章	USART 实用设计基础	406
14.1	异步通信接口应用设计要点	406
14.1.1	接口的硬件设计	406
14.1.2	上层应用通信协议和规范的制定	408
14.1.3	典型 USART 底层驱动+中间层软件结构示例	409
14.2	一个 USART 应用的完整示例	414
14.2.1	硬件系统构成	415
14.2.2	通信协议的制定	415
14.2.3	下位机系统程序	417
14.2.4	测试和上位机程序	420
14.3	基于异步通信接口实现多机通信	422
14.3.1	多机通信实现原理	422
14.3.2	多机通信实现方式一	423
14.3.3	多机通信的通用实现方式	425
	思考与练习	430
第 15 章	串行 SPI 接口应用	432
15.1	SPI 串行总线介绍	432
15.1.1	SPI 总线的组成	432
15.1.2	SPI 通信的工作模式和时序	433
15.1.3	多机 SPI 通信	435
15.2	AVR 的 SPI 接口原理与使用	436
15.2.1	SPI 接口的结构和功能	436
15.2.2	与 SPI 相关的寄存器	439
15.2.3	SPI 接口的设计应用要点	441
15.3	SPI 接口应用实例	443
15.3.1	SPI 接口基本方式的应用	443
15.3.2	典型 SPI 底层驱动+中间层软件结构示例	451