

角色系列

AVR 单片机工程师 是怎样炼成的

——基于C语言+Proteus仿真

老 杨 李鹏举 ○ 编著

Proteus
ICC AVR
AVR Studio
VC++ 6.0
Office Visio

等诸多工具领你轻松入门



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

角色系列

AVR 单片机工程师 是怎样炼成的

——基于 C 语言+Proteus 仿真

老 杨 李鹏举 编著

電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

这是一本教你如何使用 AVR 单片机的经典之作，是一位工程师的学习笔记，同时也包含了一个嵌入式工程师近 10 年的学习、工作总结。

本书以 ATmega16 为核心，结合 ICC AVR、Proteus、AVR Studio、VC++ 6.0 等诸多开发工具，软硬结合、由浅入深、全面系统地介绍了 AVR 单片机的原理和应用。本书内容丰富、使用性强，以按键为例：书中既介绍了按键的基本应用（扫描按键），同时也讲解了外部中断应用按键、按键的复用、按键消抖处理、矩阵键盘的扫描、借助译码器扩展键盘等在科研工作中常用的技术、技能。

本书适用于 AVR 单片机的初学者和使用 AVR 单片机从事项目开发的技术人员，可供从事自动控制、智能仪器仪表、电力电子、机电一体化等相关专业的技术人员参考，也可作为大学本、专科院校单片机课程教材。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有，侵权必究。

图书在版编目（CIP）数据

AVR 单片机工程师是怎样炼成的：基于 C 语言+Proteus 仿真 / 老杨，李鹏举编著. —北京：电子工业出版社，2012.11

（角色系列）

ISBN 978-7-121-18750-6

I. ①A… II. ①老… ②李… III. ①单片微型计算机—C 语言—程序设计 IV. ①TP368.1②TP312

中国版本图书馆 CIP 数据核字（2012）第 246897 号

策划编辑：王敬栋（wangjd@phei.com.cn）

责任编辑：徐 萍

印 刷：北京丰源印刷厂

装 订：三河市皇庄路通装订厂

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本：787×1 092 1/16 印张：34.25 字数：877 千字

印 次：2012 年 11 月第 1 次印刷

印 数：4 000 册 定价：79.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线：（010）88258888。

此书献给在单片机的路上探索的年轻人

——怀念我那年少无知的日子

前言

提起本书的写作，还要从我以前写的一份学习心得《Proteus-ATmega8 仿真》说起。上大学时我学的是应用数学专业，因而对硬件了解比较少，在学习单片机时走了很多的弯路。自己学习单片机入门后，就想写点东西，用来帮助和我一样想学单片机但又不知道如何入门的人，于是就有了《Proteus-ATmega8 仿真》。也许是当时国内介绍 Proteus 的书还不多的原因吧，不长的时间里，该文档就被各技术论坛转载，也有不少读者来信索取后续文档，但由于我后来开始学习 ARM-Linux 系统，就再也没有时间把后面的章节补充完整。

2011 年年初，应电子工业出版社王敬栋编辑的邀请，我写了第一本书《51 单片机工程师是怎样炼成的——基于 C 语言+Proteus 仿真》。写完第一本书，有了书的框架，就有把 AVR 系列补充完整的冲动，因而在去年的下半年开始着手 AVR 部分的写作。不过由于我的懒惰，迟迟没有完稿。今年开春，王编辑传来喜讯：第一本书在 3 个多月的时间里发货 3 000 多本，作为一本技术类书已经很不错，因而希望我尽快完成后面的部分。

本书虽然是一本 AVR 单片机的自学教材，但是如果读者没有一点儿单片机基础，是从头学起的，我建议最好先读一下《51 单片机工程师是怎样炼成的——基于 C 语言+Proteus 仿真》或郭天祥老师的《新概念 51 单片机 C 语言教程——入门、提高、开发、拓展全攻略》，因为 51 单片机学起来比较简单、容易上手，不需要花费很多时间，并且有利于培养学习单片机的兴趣。其实从 51 单片机入门是很轻松的一件事情，如果在假期学习也许一周就能入门；边上课（或工作）边学习的话，我想 1 个月的时间就足够了。对 51 单片机有了基本的认识，再来学习 AVR，我觉得会轻松很多。

本书就是写给已经有 C 语言基础，但还没有接触过单片机理论的初学者，尤其适合于大二、大三的理工科学生（电信、计算机、自动化等专业）。如果读者没有 C 语言程序设计的基础，我还是建议读者先认真地学习一下 C 语言程序设计。在此推荐两本有关 C 语言的书（有了 C 语言的基础，您才有希望在嵌入式开发中走得更远）：

- 谭浩强老师的《C 语言程序设计》
- 林锐老师的《高质量程序设计指南——C++/C 语言》

重点推荐林锐老师的《高质量程序设计指南——C++/C 语言》，该书可以说是 C 语言程序设计的规范。即使您学过了 C 语言程序设计，我也建议您读读该书，因为它能指导您编写更规范的代码，会在以后的工作中给您更多的帮助！

孔子曰：知之者不如好之者，好之者不如乐之者。写本书的目的是想用另一种方式来讲述单片机，培养读者学习单片机的兴趣。在 2011 年“中国达人秀”的舞台上，有一位科学达人叫卢驭龙，16 岁就能用特斯拉线圈来做表演，我想这就是他在科学研究中找到了自己的乐趣。由此，我们可以观察一下这届达人秀，凡是能走到最后的，大多都是对自己的表演有浓厚兴趣的人，如街舞达人卓君对街舞的热爱、菜花甜妈蔡洪平对歌唱的热爱、力量达人安东对体操的热爱等，这些人都是为了获奖而表演的，而是这些表演活动已经成了他们生活的



一部分,在这些表演中他们找到了自己的快乐。因而,我并不期望读者读过我的教材就能成为 AVR 方面的“专家”,但我希望读者能够通过学习找到单片机程序设计的乐趣,找到调试程序的那种成就感和刺激感。

本书共分为 4 篇,可以说 4 篇内容是各自独立的。第 1 篇基础篇是对单片机概念的引入、工具的介绍。工欲善其事,必先利其器。要想学好 AVR,还是要先了解一下我们要使用的开发工具;当然,如果读者对 AVR 的开发工具有所了解,也可以跳过本篇的学习。第 2 篇入门篇是单片机基础知识的讲解。学习完本篇内容你会对单片机有一个整体的认识,也会知道单片机编程是怎么一回事,如果要深入学习,可以看后面的章节,也可以直接选择别的教程做进一步研究。第 3 篇提高篇是在第 2 篇内容的基础上通过扩展外围器件进一步讲解单片机的应用,可以作为扩展内容学习。第 4 篇应用篇通过实例的形式来应用前两篇的学习内容,如果你有一定的单片机基础,也可以直接用来作参考样例。

另外,本书除第 20 章外,所有的实例都没有提供电子版代码,因为本人一直认为学习程序设计,最重要的就是要自己动手。只有自己把代码一个字一个字地敲进去,自己去建立工程、编译、仿真,这样才能学到其中的精髓,如果仅仅是运行、仿真一下别人写好的程序,那是没有什么意义的。甚至我认为你不但要看懂、学会、调通书上的例程,而且还要学会将自己的想法融入到实例之中,学会修改、优化代码。正所谓传授者应授人以渔而非授人以鱼,而读者若能做到收人以渔而非收人以鱼,那将是完美的教与学的过程。书中的所有实例都是我一个一个输入计算机做过仿真之后才复制到文中的,如果因为我的描述不清楚或其他原因使您不能通过调试,希望您能给我发邮件,或者直接登录我的博客(young45.cublog.cn)提问,我会尽快给您答复并给出解决办法。

我使用的开放环境是 Windows XP 系统,使用的工具 Proteus 是 7.8 SP2 版、ICC AVR 是 7.22 版、AVR Studio 是 4.19 版,学习时最好能选择相近的版本。

本书由老杨、李鹏举编著。参与本书编写的还有蔡振华、陈锦泉、段富军、樊战胜、黄得建、聂运中、王盛、徐春雨、于维涛、张琦、张延杰、周乐平、周宏云等;另外,王路、康磊、李楠、刘远峰、刘增、赵景丽、赵铮等也为本书的编写提供了不少的帮助。

“学然后知不足,教然后知困。”以前随手写点东西倒也没感觉有什么,现在真正要写成书了,才发现自己知识的匮乏。由于本人能力有限,写作过程中难免会出现一些错误,如果您发现错误,希望您能给我发邮件:young45@126.com。

最后感谢我的家人,特别是我的父母,是他们用辛勤的汗水浇灌了我今天的成长;感谢我的读者,是他们热心的来信鼓励了我的写作,他们的建议更是完善了本书的编写,特别是读者郭凯、张锋等,他们那种认真的读书态度深深地激励着我,而且他们也为本书的编写提供了很好的建议;另外,也感谢我的朋友郝云鹏,在近几年的工作和生活中给我提供了很多的帮助。同时也感谢电子工业出版社的策划编辑王敬栋和本书的责任编辑徐萍,感谢这些年来在学习和工作中给我提供帮助的老师 and 朋友!

老 杨

2012 年 8 月

古都·洛阳



老杨学习单片机之路

老杨者杨百军也，豫东一农民之子。千禧之年，也是一个龙年，我第一次走出农村走出小县城，走进古都洛阳城。

这一年，我考进了一个二本院校——洛阳工学院。我之所以能走进洛工，还要特别感谢两位老师：杨明祥老师和高克权老师。杨老师是我的高三班主任，是他在填报志愿之时，拿起笔为我修改并填写了第一报考院校——洛阳工学院，我也是在那时才知道该院校，没想到之后它就成了我的大学象牙塔；高老师是我大学时的系党委书记，进入大学之后，听班长说是高老师在省招生办招生时将我们几个总成绩稍差但数学成绩较好的学生调剂到数学系的。可以说没有这两位老师的帮助，也就没有我今天的学习成果了。

【按语】由此可见一位教师对一个学生可以产生多么大的影响，也许他一个小小的动作就可以改变一个学生的一生。前年暑假回老家，同样听说一位老师建议学生修改志愿的事，不过与我的相反，该学生平时成绩一般，高考发挥超常，但是老师建议其按平时成绩填报志愿，就这样该生失去了进入重点院校的机会。

就这样，我走进了洛阳工学院，走进了数学与应用数学专业，在这里结识了我的 58 位兄弟姐妹。

大一上学期（2000 年下半年），我知道了什么是计算机，学习了 Windows 98 系统和 DOS 系统（那时流行的是 Windows Me 系统）。当时计算机还不是很普及，我对计算机的认识几乎是空白。学计算机基础（Windows 98）时，我到图书馆办了一张计算机学习卡，每天过去练习打字（练习键盘盲打）。每次过去，空荡荡的一个机房只有两三个人在那里“啪嗒、啪嗒”地敲击键盘，就这样练习了一个月左右，老杨也可以“挥指如飞”了。

大一下学期，我们数学专业还没开 C 语言课（工科专业都是此时开的），我就跟着我的师兄师姐们（99 级）学习。不知道什么时候，我开始迷上了计算机，并且开始了我大学里的第一次蹭课经历。

后来，为了更好地学习计算机，我在涛哥的帮助下买了学姐毕业离校时卖的二手电脑，那是我的第一台计算机，是一台老古董：386 的 CPU，2GB 硬盘，64MB 内存。就这还是品牌机呢！据那位学姐说，买时价值 15 000 多元呢！这台计算机旧是旧了点，不过我在它身上还是学到了不少的东西，开始是安装操作系统，Windows 98、Windows Me，我自己都不知道折腾了多少遍；后来打开机箱研究内部的零件，计算机硬件的那点知识都是在它身上学到的；再后来就是安装 VC++ 6.0，调试 C 语言程序。

大二上学期跟着同学们开始报考计算机二级 C 语言考试。同时，也开始了我的大学蹭课之路，从电信、自动化、计算机到夜大，几乎我能在大学里找到的喜欢的课程都去蹭过，一直到大四，我还去蹭计算机专业的编译原理。

大二下学期报考计算机三级数据库考试，我们专业开了数据库课程。同时为了不影响学

习，我把那台古董计算机转出去了，虽然搭进去了几百块“大洋”，不过感觉还是蛮值的（如果保存到现在，不知道会不会像 Apple I 那样升值呢……）。

大三报考计算机三级 PC 技术、网络技术考试，10 月份参加高级程序员考试。通过一系列的考试，最后我才发现自己比别人多的只是几个证书，从能力上讲和别的学生没有太大区别。

大四上学期，我开始专心学习 VC++ 6.0。学习 VC 的过程是痛苦的，买了几本入门级的教材，错误百出，又无处请教，那种痛苦只有编程的人在调试 BUG 时才能体会到。就这样磕磕绊绊学了一个学期，基本还是处于“Hello World!”的阶段。

大四下学期开始忙于毕业设计、帮同学做毕业设计，能者多劳嘛！兄弟们都知道老杨学了点计算机，只好帮大家做点事了。我们自己的专业主要是用 Matlab、Mathematica 编程解数学题。

总结一下我的大学生活，感觉还是很充实的。自己的本专业除了学一些数学基础课，还学了 C 语言、数据结构、数据库等计算机课程，同时又旁听了电路、模电、数电、软件工程、操作系统、微机原理、计算机组成原理、编译原理等计算机课程。虽然不能说都学得很好，但有些课程的理论基础还是有的，这也为我以后走上工作岗位自学单片机奠定了基础。

【按语】大学里，老杨几乎没利用过的资源就是实验室——也是学校最为宝贵的资源之一。当年的老杨盲目于考证书，现在的大学生倒是钟情于参加电子竞技，参加竞赛可以快速提高自己的学习水平，这本是一件好事，郭天祥老师就是这么走过来的，但如果过于追求竞赛——为竞赛而竞赛，我觉得也会影响到大学理论课程的学习。大学里，我们重点要修炼的是自学的能力和基础理论的积累，以嵌入式程序设计为例：在大学里，如果学好了电路、模电、数电、微机原理、C 语言、数据结构、软件工程、操作系统等几门基础课，工作之后不管是 AVR、ARM、DSP 等，还是当下比较流行的其他片子，都能很快上手，而且还可以在这条路上走得很远。因而，我并不建议在大学里过于追求参加竞赛、过于追求当下流行单片机的学习，这样会影响基础理论的学习（因为那些课程的学习过程本身就是比较枯燥的），有一天离开了大学，就很难再拾起这些理论知识了。

2004 年我就业了，进了洛阳本地的一家科研院所。工作之后，我才认识到什么叫 VC++ 编程，在这里，我开始在同事的带领下啃侯捷老师和潘爱民老师的著作。可是这段学习是短暂的，我后来被派到深圳一家分公司学习，在那里开始认识单片机程序设计，也了解到一些深圳小企业的发展模式：抄板、尽快出新产品。我在那里跟的产品是电话交换机——智能话务员，也就是现在很多公司电话总机用的那种自动提示的东西，如 10086。通过这个产品，我有两点比较深刻的认识：第一，要想设计附加值比较高的产品，就要将软件和硬件结合起来，做单片机程序设计；第二，通信协议到底是怎么一回事，以前学计算机网络，七层协议考试时几乎可以倒背，可是真正地理解通信协议还是在那时。

2005 年，我被公司召回洛阳，安排进一个新的产品线——安防产品线，开始指纹产品的设计。在这个新产品线上，我开始零距离接触单片机 AVR、DSP。也是从这一年起，我开始自学 51 单片机，真正地开始了老杨的学习单片机之路。

学习 51 单片机大概用了 4 个月左右的时间，白天上班，晚上和周末用来看相关的视频、看书模仿实例，最后编写、调试代码。之后开始着手学习 AVR——ATmega8。刚好公司新来了一批北航的大学生，后来也是这批学生中的小同事李兴国向我介绍了 Proteus 这款软件，当

时的版本是 6.7，我一接触就喜欢上了它。

2006 年，我利用晚上的时间把学习过的内容整理了一遍，于是就有了《Proteus-ATmega8 仿真》这份资料。我上传的第一个网站好像是阿莫的 ouravr（后来改名为 ourdev），当时阿莫还在禁传 Proteus，但好东西总是会流传开的，也许是当时写 Proteus 仿真的教程还不多吧，很快该教程就在各个 AVR 网站流传开了。2010 年年底，电子工业出版社的王敬栋编辑找我写书，我想应该也是看到该篇教程的缘故吧。

【按语】当年随手之作，没想到为我后来著书做了铺垫。这也提醒各位读者，要学会无私地奉献自己的学习心得，学习的经验只有互相交流才能有更好的进步，如果我学习单片机时也像我在大学里学习 VC++ 一样，闭门造车，可能就不会有今天的成果了。

2007 年至 2009 年，是我漫长而痛苦的学习 ARM+Linux 的过程。当时学习困难有三个原因：一是当时的 ARM 还是比较新的事物，在南方的公司已经成为产品，而地处中原的洛阳，很难找到能够学习的地方（当时网上的相关资料还不是那么丰富，好的教材也没有现在多）；二是要学习的内容太多，Linux 系统、GCC 开发环境、ARM 系统、BootLoad 移植、系统移植、驱动开发等，每个环节对我来说都是新的；三是自身的硬件基础比较薄弱，学习 ARM 内核和移植 uboot、移植 Linux 的过程艰难而漫长。

2010 年，我终于开始尝试着带领一个小团队将“夹生”的 ARM+Linux 系统移植到我们的产品中。

2011 年，受王敬栋编辑相邀，完成《51 单片机工程师是怎样炼成的——基于 C 语言+Proteus 仿真》一书。

2011 年下半年到 2012 年 6 月，完成《AVR 单片机工程师是怎样炼成的——基于 C 语言+Proteus 仿真》一书。

单片机的路上，老杨愿与你同行！

老 杨

目 录

第 1 篇 基 础 篇

第 1 章 走近单片机	2
1.1 我们身边的单片机	2
1.2 认识单片机	5
1.3 了解 AVR 单片机	8
1.3.1 单片机的历史	8
1.3.2 AVR 单片机的特点	9
1.3.3 AVR 单片机的代表 ATmega16	13
第 2 章 修炼倚天剑: Proteus	17
2.1 Proteus 介绍	17
2.2 安装 Proteus	17
2.3 运行 Proteus	18
2.4 学用 Proteus	21
2.5 补充内容	26
2.5.1 汉化 Proteus	26
2.5.2 Proteus 元件库	27
第 3 章 宝刀屠龙: ICC AVR、AVR Studio	29
3.1 编译工具介绍	29
3.1.1 安装 ICC AVR	30
3.1.2 运行 ICC AVR	31
3.1.3 学用 ICC AVR	32
3.2 开发调试软件 AVR Studio	38
3.2.1 下载 AVR Studio	39
3.2.2 安装 AVR Studio	41
3.2.3 使用 AVR Studio	45

第 2 篇 入 门 篇

第 4 章 让你的单片机“眨眨眼睛”	54
4.1 我们的第一个单片机程序	54
4.1.1 用 ICC AVR 新建一个工程	54
4.1.2 画出我们要用的电路	59
4.1.3 用 Proteus 仿真、调试	65



4.1.4	用 AVR Studio 仿真调试	68
4.1.5	AVR Studio 与 Proteus 联合调试	70
4.1.6	分析代码	72
4.1.7	补充一点发光二极管的知识	73
4.2	不仅仅是让它亮起来	75
4.2.1	如何让发亮的灯闪烁呢	75
4.2.2	代码分析	77
4.2.3	调试分析	78
4.3	做些程序的改动	80
4.3.1	改动延时时间	80
4.3.2	做个众人皆知的跑马灯	81
4.3.3	复习 LED 示例工程	84
4.4	能不能玩点花样呢	85
4.4.1	想想你能画什么	85
4.4.2	用代码显示数字	86
第 5 章	让数字显示出来	90
5.1	引入数码管的概念	90
5.1.1	数码管介绍	90
5.1.2	写段程序让它亮起来	91
5.1.3	数码管的驱动方式	93
5.2	多显示几个数字看看	94
5.2.1	电路实现	94
5.2.2	程序实现	96
5.2.3	你的眼睛欺骗了你	97
5.3	仿真万年历	98
5.3.1	电路实现	99
5.3.2	程序实现	100
5.3.3	优化程序	101
5.4	补充内容	104
5.4.1	排阻	104
5.4.2	数码管的应用	104
5.4.3	比较 PC 口和 PA 口	105
第 6 章	用按键告诉单片机你想做什么	107
6.1	能不能让闪烁的灯停下来	107
6.1.1	让跑马灯停下来	107
6.1.2	认识我们用的按键	110
6.2	如何实现软件消抖	111
6.3	如何实现按键的复用	115
6.4	补充内容	119
6.4.1	有关程序流程图	119



6.4.2	认识一下 PB 口和 PD 口	120
第 7 章	如何实时控制单片机	123
7.1	中断的引入	123
7.1.1	写段程序实现按键的中断	123
7.1.2	用代码生成器生成程序框架	124
7.1.3	实现中断计数功能	126
7.1.4	中断函数的写法	130
7.1.5	汇编指令	131
7.2	外部中断寄存器	132
7.2.1	状态寄存器——SREG	132
7.2.2	MCU 控制寄存器——MCUCR	132
7.2.3	MCU 控制与状态寄存器——MCUCSR	133
7.2.4	通用中断控制寄存器——GICR	134
7.2.5	通用中断标志寄存器——GIFR	134
7.2.6	T/C 中断屏蔽寄存器——TIMSK	135
7.2.7	外部中断操作要点	135
7.2.8	外部中断操作流程	135
7.3	不同的中断触发方式	136
7.4	补充内容	138
7.4.1	了解更多中断的概念	138
7.4.2	比较一下 4 个 I/O 口	139
第 8 章	让定时更精确	142
8.1	定时器/计数器概念	142
8.1.1	用代码生成器生成定时器函数	142
8.1.2	了解 ATmega16 的 T/C0	146
8.1.3	与 T/C0 相关的寄存器	148
8.2	解读代码生成器生成的代码	151
8.2.1	电路设计	151
8.2.2	代码解读	152
8.2.3	完整的代码实例	155
8.3	T/C0 的其他三种工作模式	158
8.3.1	CTC 模式	158
8.3.2	快速 PWM 模式	163
8.3.3	相位修正 PWM 模式	167
8.3.4	重新认识 T/C0	168
8.4	定时器/计数器 2	168
8.4.1	了解定时器/计数器 2	168
8.4.2	与 T/C2 相关的寄存器	169
8.4.3	用 T/C2 的异步模式做电子钟	172

8.5	定时器/计数器 1	178
8.5.1	了解 T/C1	178
8.5.2	T/C1 的工作模式	179
8.5.3	与 T/C1 相关的寄存器	181
8.5.4	应用 T/C1	184
8.6	本章使用的寄存器与 I/O 端口	199
8.6.1	与定时器/计数器相关的寄存器	199
8.6.2	与定时器/计数器相关的引脚	200
第 9 章	单片机与外界的联系——串口通信	202
9.1	串口通信的概念	202
9.2	和 USART 相关的寄存器	203
9.2.1	USART I/O 数据寄存器——UDR	203
9.2.2	USART 控制和状态寄存器 A——UCSRA	203
9.2.3	USART 控制和状态寄存器 B——UCSRB	204
9.2.4	USART 控制和状态寄存器 C——UCSRC	205
9.2.5	USART 波特率寄存器——UBRRL 和 UBRRH	206
9.2.6	访问 UBRRH/UCSRC 寄存器	207
9.3	先试试如何发送	208
9.3.1	使用代码生成器	208
9.3.2	系统设计	210
9.3.3	编译和仿真	214
9.3.4	补充一点 AVR 的串口知识	216
9.4	实现收发通信	218
9.4.1	将 Proteus 中的串口连接到硬件上	218
9.4.2	实现串口收发通信	223
9.4.3	用中断的方式实现串口收发	227
第 10 章	让单片机“响”起来	230
10.1	单片机还会说话呢	230
10.1.1	了解一下蜂鸣器	230
10.1.2	写段程序让它响起来	230
10.2	用定时器模拟门铃	233
10.3	用蜂鸣器演奏曲子	236
10.4	重新认识 ATmega16	239
10.4.1	ATmega16 的总体认识	239
10.4.2	I/O 口的第二功能	242
10.4.3	ATmega16 的中断向量	244
10.4.4	有关 ATmega16 的寄存器	245

10.4.5 有关 AVR 程序设计的学习	249
-----------------------------	-----

第 3 篇 提 高 篇

第 11 章 再说数码管	252
11.1 使用扩展芯片驱动数码管	252
11.1.1 认识扩展芯片 74LS138	253
11.1.2 74LS138 的应用	254
11.1.3 74HC154 的应用	257
11.1.4 有关译码器	258
11.2 显示译码器	259
11.2.1 显示译码器 74LS48	259
11.2.2 显示译码器的应用	261
第 12 章 按键多了怎么办	265
12.1 看一个密码键盘	265
12.1.1 矩阵键盘的实现	265
12.1.2 矩阵按键的扫描	266
12.2 是否可以扩展呢	270
12.2.1 了解编码器	270
12.2.2 应用编码器 74LS148	272
第 13 章 深入串口通信	276
13.1 通信协议的引入	276
13.1.1 协议制定	276
13.1.2 样例实现	277
13.2 让单片机和 PC 真正通信	287
13.2.1 新建一个 VC 工程	287
13.2.2 添加 VC 控件	288
13.2.3 添加控件到界面	290
13.2.4 添加变量、完成代码编写	293
13.2.5 运行、调试	301
第 14 章 IIC (TWI) 总线及其应用	303
14.1 IIC 总线介绍	303
14.1.1 初步了解 IIC 总线	303
14.1.2 IIC 总线上的信号	304
14.2 认识 E ² PROM AT24C02	308
14.2.1 认识 AT24C02	308
14.2.2 AT24C02 的操作地址	309
14.2.3 AT24C02 的读/写操作	310
14.3 写一个 IIC 通信的例子	312
14.4 AVR 的 TWI 总线	322



14.4.1	TWI 模块综述	323
14.4.2	TWI 寄存器说明	324
14.4.3	使用 TWI	326
14.5	应用 TWI	328
第 15 章	SPI 总线及其应用	338
15.1	SPI 总线介绍	338
15.2	一款 SPI 接口的 E ² PROM	339
15.2.1	认识 25AA040	339
15.2.2	25AA040 的工作原理	341
15.3	应用 25AA040	350
15.4	AVR 的 SPI 接口	358
15.4.1	了解 AVR 的 SPI 接口	358
15.4.2	与 SPI 相关的寄存器	360
15.4.3	SPI 示例程序	362
15.4.4	应用 SPI 接口	363
第 16 章	A/D 转换器和 D/A 转换器	369
16.1	了解 A/D 转换器	369
16.1.1	A/D 转换器的分类	369
16.1.2	A/D 转换器的主要技术指标	370
16.2	先看一个 A/D 转换器的例子	370
16.2.1	了解 ADC0809	370
16.2.2	应用 ADC0808	373
16.3	AVR 自带的 A/D 转换器	379
16.3.1	ATmega16 内部 ADC	379
16.3.2	应用 ATmega16 内部 ADC	386
16.4	看看 D/A 转换器	390
16.4.1	了解 D/A 转换器	390
16.4.2	D/A 转换器的主要技术指标	392
16.4.3	了解 DAC0832	392
16.4.4	应用 DAC0832	396
16.5	模拟比较器	399
16.5.1	ATmega16 的模拟比较器	399
16.5.2	与模拟比较器相关的寄存器	400
16.5.3	模拟比较器的应用	401
第 17 章	字符液晶	405
17.1	认识 LED 点阵屏	405
17.1.1	关于 LED 点阵屏	405
17.1.2	LED 点阵的应用	406
17.2	使用字符 LCD	410
17.2.1	了解 LCD	410

17.2.2	认识 LCD1602	410
17.2.3	LCD1602 指令说明	412
17.2.4	使用 LCD1602	416
第 18 章	图形液晶	421
18.1	了解 LCD12864	421
18.2	认识 HDG12864L-6	422
18.2.1	HDG12864L-6 的主要技术参数	422
18.2.2	HDG12864L-6 引脚说明	422
18.2.3	HDG12864L-6 指令说明	423
18.2.4	HDG12864L-6 时序图	432
18.3	LCD12864 的应用	434
18.3.1	LCD12864 的并行应用	434
18.3.2	LCD12864 的串行应用	442

第 4 篇 应 用 篇

第 19 章	简易万年历	454
19.1	电路设计	454
19.2	认识时钟芯片 DS1302	456
19.3	系统设计	462
第 20 章	智能温度控制系统	479
20.1	系统模型	479
20.2	分析电路	481
20.2.1	电路分解	481
20.2.2	认识 DS18B20	483
20.2.3	了解液晶 AMPIRE128X64	494
20.3	系统设计	499
20.3.1	嵌入式系统设计	499
20.3.2	PC 端软件设计	510
思考		515
附录 A	ATmega16 I/O 口寄存器速查表	516
附录 B	片内 E ² PROM 操作	518
附录 C	看门狗定时器	524
附录 D	书籍、网站推荐	529
参考文献		530
结束语		531

工欲善其事 必先利其器

——孔子

第1篇

基础篇

在看本篇之前，您可以没有单片机基础，也可以没有使用过仿真工具 Proteus，但希望您有一定的 C 语言基础。不要认为学会一款单片机有多么重要，其实它远没有学会一门程序设计语言重要。学会一门程序设计语言，可以上手很多种单片机，可以说学会 C 语言是掌握了一门技术，学习单片机就是使这门技术有了用武之地。因此，如果你还没有学过 C 语言，希望在看本篇之前你能先学习一下 C 语言程序设计。

本篇主要引入单片机的概念，讲述 AVR 单片机的基本知识；介绍本书要使用的开发工具——Proteus、ICC AVR 和 AVR Studio 等。

本篇共 3 章内容：第 1 章通过我们身边的电器引入单片机的概念、讲解 AVR 单片机的基本知识；第 2 章介绍本书要使用的仿真工具 Proteus；第 3 章介绍本书要使用的编程工具 ICC AVR 和调试工具 AVR Studio。