

单 片 机 高 级 编 程 系 列

AVR单片机 C语言 高级程序设计

周兴华 编著



光盘附赠本书实例所用源代码



中国电力出版社
www.infopower.com.cn

单 片 机 高 级 编 程 系 列

AVR单片机 C语言 高级程序设计

周兴华 编著



中国电力出版社
www.infopower.com.cn

内 容 提 要

本文以实践为主线,以生动短小的程序实例为灵魂,带领读者一步步地学习掌握 AVR 单片机的 C 语言高级程序设计。内容包括 ICC AVR 使用的 C 语言基础知识、ICCAVR6.31A C 语言编译器的使用、AVR 单片机的片上资源使用等,书中还详细介绍了 AVR 单片机开发工具(JTAG 仿真器及并口下载器)的制作方法,最后 3 章则详细介绍了使用 AVR 单片机开发的 3 个产品实例。

本书的学习难度低、实例丰富,非常适合电子技术人员及单片机爱好者作为学习 AVR 单片机的入门用书,也可作为大中专院校电子技术、自动化、计算机科学及其他相关专业的培训教材。

图书在版编目(CIP)数据

AVR 单片机 C 语言高级程序设计 / 周兴华编著. — 北京: 中国电力出版社, 2008
(单片机高级编程系列)

ISBN 978-7-5083-6588-6

I. A… II. 周… III. 单片微型计算机 - C 语言 - 程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2008)第 014709 号

责任编辑: 孙 芳

责任校对: 崔燕菊

责任印制: 李文志

书 名: AVR 单片机 C 语言高级程序设计

编 著: 周兴华

出版发行: 中国电力出版社

地址: 北京市三里河路 6 号 邮政编码: 100044

电话: (010) 68362602 传真: (010) 68316497

印 刷: 航远印刷有限公司

开本尺寸: 185mm × 233mm 印 张: 20 字 数: 436 千字

书 号: ISBN 978-7-5083-6588-6

版 次: 2008 年 3 月北京第 1 版

印 次: 2008 年 3 月第 1 次印刷

印 数: 0001—4000 册

定 价: 35.00 元(含 1CD)

敬 告 读 者

本书封面贴有防伪标签,加热后中心图案消失

本书如有印装质量问题,我社发行部负责退换

版 权 专 有 翻 印 必 究

前言

进入21世纪,电子技术及计算机技术的迅猛发展,促使新型电子产品的更新换代速度越来越快。以单片机为核心构成的智能化产品具有体积小、功能强、应用面广等优点,目前,正以前所未有的速度取代着传统电子线路构成的经典系统,蚕食着传统数字电路与模拟电路固有的领地。

从前,汇编语言是单片机工程师进行软件开发的唯一选择,但汇编语言程序的可读性和可移植性较差,采用汇编语言编写单片机系统应用程序的周期长,而且调试和排错也比较困难。许多读者都发现,采用汇编语言设计一个大型复杂程序时,可读性较差,隔一段时间再看,往往又要从头再来。而且不同类型的单片机,必须采用不同的汇编语言来编写,这是由于汇编语言完全依赖于单片机硬件。这样,对开发者而言,如果由于项目的变化而经常变更单片机类型,其开发的难度是可想而知的。因为学习一种单片机汇编语言开发,总少不了学习、探索、实践、进步这样一个过程,这个过程少则半年,多则1~2年,等你学完后再搞出产品,也许商机早已消失。随着社会竞争的日益激烈,开发效率已成为商战致胜的最重要法宝之一。

为了提高编制单片机系统和应用程序的效率,改善程序的可读性和可移植性,最好的办法是采用高级语言编程。目前,C语言逐渐成为国内外开发单片机的主流语言。

C语言是一种通用的编译型结构化计算机程序设计语言,在国际上十分流行,它兼顾了多种高级语言的特点,并具备汇编语言的功能。它支持当前程序设计中广泛采用的由顶向下的结构化程序设计技术。一般的高级语言难以实现汇编语言对于计算机硬件直接进行操作(如对内存地址的操作、移位操作等)的功能,而C语言既具有一般高级语言的特点,又能直接对计算机的硬件进行操作。C语言的库函数功能丰富、运算速度快、编译效率高,并且采用C语言编写的程序能够很容易地在不同类型的计算机之间进行移植。因此,C语言的应用范围越来越广泛,用C语言进行单片机程序设计是单片机开发与应用的必然趋势。对汇编语言掌握到只要可以读懂程序,在时间要求比较严格的模块中进行程序的优化即可。采用C语言不必对单片机和硬件接口的结构有很深入的了解,编译器可以自动完成变量的存储单元的分配,编程者就可以专注于应用软件部分的设计,大大加快了软件的开发速度。统计资料表明,不同单片机的C语言编译程序80%的代码是公共的,采用C语言可以很容易地进行单片机的程序移植工作,有利于产品中的单片机重新选型。C语言的编译效率也很高,对于同一个问题,用C语言编写的程

序生成代码的效率仅比用汇编语言编写的程序低10%~20%，由于现在片上ROM（或FLASH ROM）空间做到32/64KB（或更大）的单片机比比皆是，因此代码效率所差的10%~20%已经不是重要问题。对开发速度、软件质量、结构严谨、程序坚固等方面进行综合评述，则C语言的完美绝非是汇编语言所能比拟的。

本书以初学者为对象，从零开始，循序渐进地讲解当前最热门的AVR单片机C语言高级程序设计知识，在介绍AVR单片机各单元基本特性的同时，使用入门难度浅、程序长度短且又能立竿见影的初级实例，详细介绍了如何使用ATmega16(L)的片上资源，帮助初学者快速掌握AVR单片机的高效设计。本书最后3章详细介绍了3个以AVR单片机为核心的实际产品的软硬件设计实例。

随书所附的光盘中提供了本书的所有软件设计程序文件，读者朋友可参考使用。

限于作者水平有限，书中必定还存在不少缺点或漏洞，诚挚欢迎广大读者提出意见并不吝赐教。学习AVR单片机需要一定的学习、实验器材，如果读者朋友需要书中介绍的学习、实验器材时，可与作者联系，咨询购买事宜。联系方式如下：

地址：上海市闵行区莲花路2151弄57号201室

邮编：201103

联系人：周兴华

电话（传真）：021-64654216 13044152947 13774280345

技术支持 E-mail: zXH2151@sohu.com

zXH2151@yahoo.com.cn

笔者主页: <http://www.hlelectron.com>

周兴华

2008年1月

目 录

前 言

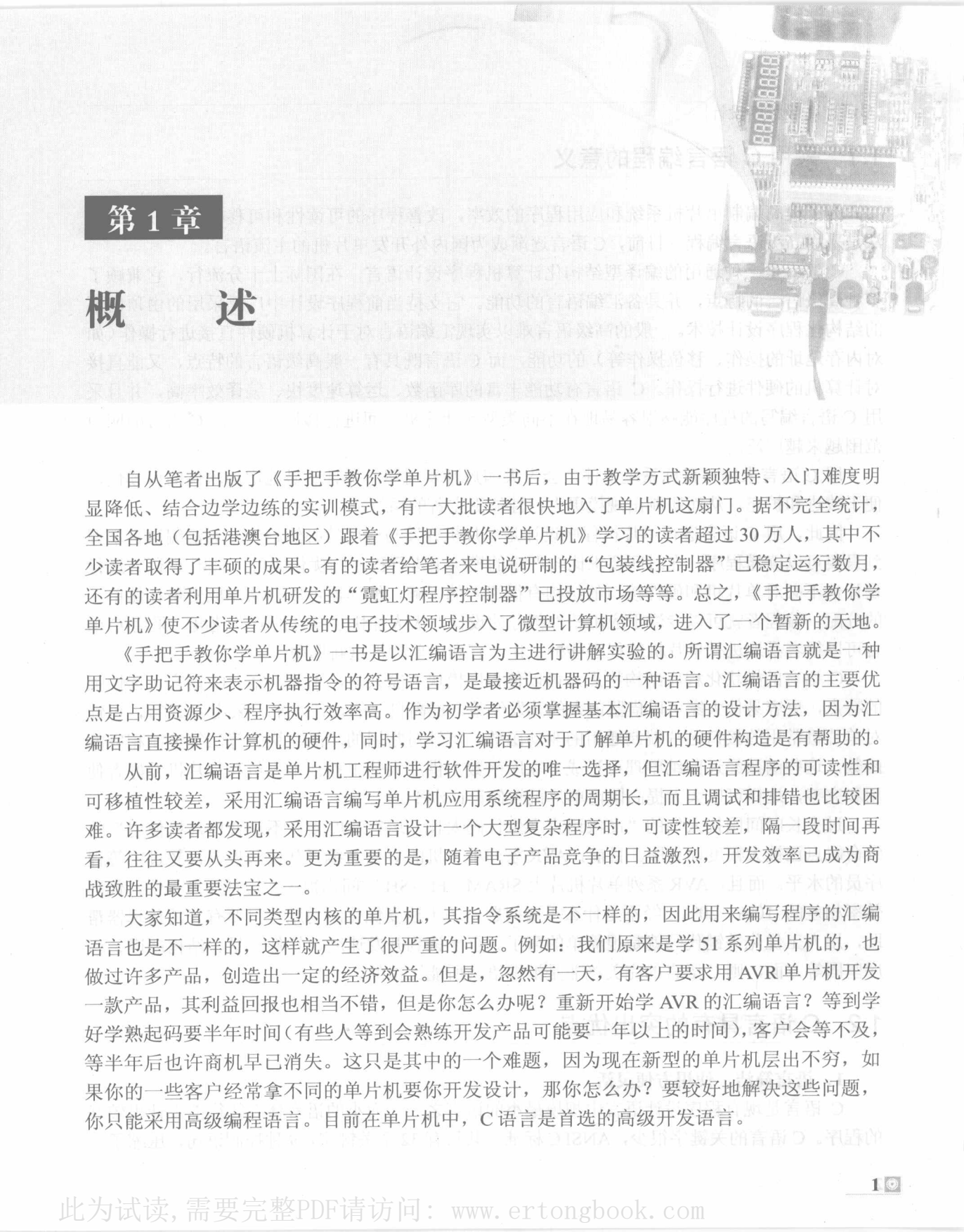
第 1 章 概述	1
1.1 采用 C 语言编程的意义	2
1.2 C 语言具有的突出优点	2
1.3 AVR 单片机的 C 编译器简介	4
第 2 章 AVR 单片机的主要特性及开发工具	5
2.1 AVR 单片机简介及主要特性	5
2.2 学习开发 AVR 单片机用到的实验工具及器材	6
第 3 章 AVR 单片机开发软件的安装及初步使用	12
3.1 ICCAVR6.31A C 语言编译器安装	12
3.2 AVR Studio 集成开发环境安装	16
3.3 PonyProg2000 下载软件安装	17
3.4 创建 AVR 入门程序	17
第 4 章 AVR DEMO 单片机综合试验板原理介绍及使用	33
4.1 AVR DEMO 单片机综合试验板	33
4.2 ATmega16(L) 单片机的主要特点与内部组成	35
4.3 使用 JTAG ICE 仿真器调试 AVR DEMO 单片机综合试验板	38
第 5 章 自制 JTAG 仿真器及并口下载器	41
5.1 JTAG 的由来及简介	41
5.2 传统的在线仿真器与 JTAG 接口的片内调试系统的区别	42
5.3 AVR JTAG ICE 仿真时的一些注意事项	43
5.4 AVR JTAG ICE 的特点	44
5.5 自制 AVR JTAG ICE	44
5.6 自制并口下载器	54

第 6 章 ICC AVR 使用的 C 语言基础知识	56
6.1 C 语言的标识符与关键字	56
6.2 数据类型	58
6.3 AVR 单片机的存储空间	59
6.4 常量、变量及存储方式	59
6.5 数组	59
6.6 C 语言的运算	63
6.7 流程控制	67
6.8 函数	72
6.9 指针	75
6.10 结构体	79
6.11 共用体	84
6.12 中断函数	86
第 7 章 AVR 单片机内部资源的 C 编程——ATmega16(L)的 IO 端口特点及使用	88
7.1 ATmega16(L)的 IO 端口	88
7.2 ATmega16(L)的 I/O 端口使用注意事项	90
7.3 ATmega16(L)的 I/O 端口编程实践	90
第 8 章 AVR 单片机内部资源的 C 编程——ATmega16(L)的中断系统特点及使用	97
8.1 ATmega16(L)的中断系统	97
8.2 ATmega16(L)的外部中断编程实践	98
第 9 章 AVR 单片机内部资源的 C 编程——ATmega16(L)的定时计数器	106
9.1 ATmega16(L)的定时/计数器	106
9.2 ATmega16(L)定时/计数器的时钟选择	106
9.3 计数器的事件	107
9.4 定时/计数器事件的处理	108
9.5 ATmega16(L)的定时/计数器编程实践	108
第 10 章 AVR 单片机内部资源的 C 编程——ATmega16(L)的模数转换器	114
10.1 ATmega16(L)的模/数转换器	114
10.2 模/数转换器相关寄存器	114
10.3 模/数转换器的使用	116
10.4 ATmega16(L)的模/数转换器编程实践	116

第 11 章 AVR 单片机内部资源的 C 编程——ATmega16(L)的内部 E²PROM 读写	123
11.1 ATmega16(L)的内部 E ² PROM	123
11.2 与 E ² PROM 相关的寄存器	123
11.3 ATmega16(L)内部 E ² PROM 编程实践	125
第 12 章 AVR 单片机内部资源的 C 编程——利用 ATmega16(L)的 PWM 功能实现数模输出	134
12.1 ATmega16(L)的定时/计数器的特性及数/模输出实现	134
12.2 ATmega16(L)的 PWM 编程实践	134
第 13 章 AVR 单片机内部资源的 C 编程——ATmega16(L)与 PC 机的串行通信	139
13.1 通用串行接口 USART 的主要特征	139
13.2 USART 的主要寄存器及设置	139
13.3 ATmega16(L)与 PC 机的串行通信编程实践	143
第 14 章 ATmega16(L)驱动 16×2 点阵字符液晶模块	151
14.1 16×2 点阵字符液晶显示器概述	151
14.2 液晶显示器的突出优点	152
14.3 16×2 字符型液晶显示模块 (LCM) 特性	152
14.4 16×2 字符型液晶显示模块 (LCM) 引脚及功能	152
14.5 16×2 字符型液晶显示模块 (LCM) 的内部结构	153
14.6 液晶显示控制驱动集成电路 HD44780 特点	153
14.7 HD44780 工作原理	155
14.8 LCD 控制器的指令	159
14.9 LCM 工作时序	161
14.10 ATmega16(L)驱动 16×2 点阵字符液晶模块的子程序	162
14.11 在 AVR 单片机综合试验板上实现 16×2LCM 演示程序 1	164
14.12 在 AVR 单片机综合试验板上实现 16×2LCM 演示程序 2	168
第 15 章 ATmega16(L)驱动 I²C 串行接口器件 24C01 的编程	172
15.1 E ² PROM AT24CXX 的性能特点	172
15.2 AT24CXX 引脚定义	172
15.3 AT24CXX 系列存储器特点	173
15.4 AT24CXX 系列 E ² PROM 的内部结构	173

15.5	AT24CXX 系列 E ² PROM 芯片的寻址	174
15.6	写操作方式	176
15.7	读操作方式	177
15.8	读写 AT24C01 的相关功能子程序	178
15.9	在 AVR 单片机综合试验板上完成 AT24C01A 的读写实验	181
第 16 章	ATmega16(L)驱动 128×64 点阵图形液晶模块	214
16.1	128×64 点阵图形液晶模块特性	214
16.2	128×64 点阵图形液晶模块引脚及功能	215
16.3	128×64 点阵图形液晶模块的内部结构	216
16.4	HD61203 特点	218
16.5	HD61202 特点	218
16.6	HD61202 工作原理	219
16.7	HD61202 的工作过程	223
16.8	点阵图形液晶模块的控制器指令	224
16.9	HD61202 的操作时序图	225
16.10	ATmega16(L)驱动 128×64 点阵图形液晶模块的子程序	227
16.11	在 AVR 单片机综合试验板上实现 128×64LCM 演示程序	230
第 17 章	ATmega16(L)的系统控制、复位和看门狗定时器使用	239
17.1	ATmega16(L)的系统控制和复位	239
17.2	ATmega16(L)的复位源	240
17.3	看门狗定时器的使用	242
17.4	在 AVR DEMO 单片机综合试验板上, 进行看门狗实验 1 (看门狗启动后在程序中定时清除它): D1~D8 的 8 个 LED (发光管) 依次流水点亮, 形成“流水灯”实验	242
17.5	在 AVR DEMO 单片机综合试验板上, 进行看门狗实验 2 (看门狗启动后在程序中不再清除它, 模拟程序失控的情况): D1~D8 的 8 个 LED (发光管) 依次流水点亮, 形成“流水灯”实验	244
第 18 章	ATmega16(L)应用实例 1——风机测试仪	247
18.1	系统功能描述	247
18.2	系统方案及电路设计	247
18.3	测试类型、参数范围及测试过程	250

18.4	程序设计	251
18.5	调试及应用	266
第 19 章	ATmega16(L)应用实例 2——红外线转速仪	267
19.1	转速仪表的分类	267
19.2	转速传感器	268
19.3	转速测量的方法	268
19.4	红外线转速仪系统功能描述	268
19.5	系统方案及电路设计	269
19.6	程序设计	271
19.7	调试及应用	278
第 20 章	ATmega16(L)应用实例 3——红外线感应自动移门	279
20.1	红外线感应自动移门的组成及基本工作原理	279
20.2	系统方案设计	280
20.3	电路设计	280
20.4	程序设计详解	286
20.5	保养及维护	308
参考文献		309



第 1 章

概 述

自从笔者出版了《手把手教你学单片机》一书后，由于教学方式新颖独特、入门难度明显降低、结合边学边练的实训模式，有一大批读者很快地入了单片机这扇门。据不完全统计，全国各地（包括港澳台地区）跟着《手把手教你学单片机》学习的读者超过 30 万人，其中不少读者取得了丰硕的成果。有的读者给笔者来说研制的“包装线控制器”已稳定运行数月，还有的读者利用单片机研发的“霓虹灯程序控制器”已投放市场等等。总之，《手把手教你学单片机》使不少读者从传统的电子技术领域步入了微型计算机领域，进入了一个崭新的天地。

《手把手教你学单片机》一书是以汇编语言为主进行讲解实验的。所谓汇编语言就是一种用文字助记符来表示机器指令的符号语言，是最接近机器码的一种语言。汇编语言的主要优点是占用资源少、程序执行效率高。作为初学者必须掌握基本汇编语言的设计方法，因为汇编语言直接操作计算机的硬件，同时，学习汇编语言对于了解单片机的硬件构造是有帮助的。

从前，汇编语言是单片机工程师进行软件开发的唯一选择，但汇编语言程序的可读性和可移植性较差，采用汇编语言编写单片机应用系统程序的周期长，而且调试和排错也比较困难。许多读者都发现，采用汇编语言设计一个大型复杂程序时，可读性较差，隔一段时间再看，往往又要从头再来。更为重要的是，随着电子产品竞争的日益激烈，开发效率已成为商战致胜的最重要法宝之一。

大家知道，不同类型内核的单片机，其指令系统是不一样的，因此用来编写程序的汇编语言也是不一样的，这样就产生了很严重的问题。例如：我们原来是学 51 系列单片机的，也做过许多产品，创造出一定的经济效益。但是，忽然有一天，有客户要求用 AVR 单片机开发一款产品，其利益回报也相当不错，但是怎么办呢？重新开始学 AVR 的汇编语言？等到学好学熟起码要半年时间（有些人等到会熟练开发产品可能要一年以上的），客户会等不及，等半年后也许商机早已消失。这只是其中的一个难题，因为现在新型的单片机层出不穷，如果你的一些客户经常拿不同的单片机要你开发设计，那怎么办呢？要较好地解决这些问题，你只能采用高级编程语言。目前在单片机中，C 语言是首选的高级开发语言。

1.1 采用 C 语言编程的意义

为了提高编制单片机系统和应用程序的效率,改善程序的可读性和可移植性,最好的办法是采用高级语言编程。目前,C语言逐渐成为国内外开发单片机的主流语言。

C语言是一种通用的编译型结构化计算机程序设计语言,在国际上十分流行,它兼顾了多种高级语言的特点,并具备汇编语言的功能。它支持当前程序设计中广泛采用的由顶向下的结构化程序设计技术。一般的高级语言难以实现汇编语言对于计算机硬件直接进行操作(如对内存地址的操作、移位操作等)的功能,而C语言既具有一般高级语言的特点,又能直接对计算机的硬件进行操作。C语言有功能丰富的库函数、运算速度快、编译效率高,并且采用C语言编写的程序能够很容易地在不同类型的计算机之间进行移植。因此,C语言的应用范围越来越广泛。

用C语言来编写目标系统软件,会大大缩短开发周期,并且明显地增加软件的可读性,便于改进和扩充,从而研制出规模更大、性能更完备的系统。

因此,用C语言进行单片机程序设计是单片机开发与应用的必然趋势。对汇编语言掌握到只要可以读懂程序,在时间要求比较严格的模块中进行程序的优化即可。采用C语言进行设计也不必对单片机和硬件接口的结构有很深入的了解,编译器可以自动完成变量存储单元的分配,编程者就可以专注于应用软件部分的设计,大大加快了软件的开发速度。采用C语言可以很容易地进行单片机的程序移植工作,有利于单片机重新选型。

C语言的模块化程序结构特点,可以使程序模块大家共享,不断丰富。C语言可读性强的特点,更容易使大家可以借鉴前人的开发经验,提高自己的设计水平。采用C语言,可针对单片机常用的接口芯片编制通用的驱动函数,可针对常用的功能模块、算法等编制相应的函数,这些函数经过归纳整理可形成专家库函数,供广大的工程技术人员和单片机爱好者使用和完善,这样可以大大提高国内单片机软件设计水平。

过去长时间困扰人们的“高级语言产生代码太长,运行速度太慢不适合单片机使用”的致命缺点已被大幅度地克服。目前,AVR系列单片机的C语言编译代码长度,已超过中等程序员的水平。而且,AVR系列单片机片上SRAM、FLASH空间都很大、运行速度很快,代码效率相差的10%~20%已经不是什么重要问题。关于速度优化的问题,只要有好的仿真器帮助,用人工优化关键代码就是很简单的事了。至于谈到开发速度、软件质量、结构严谨、程序坚固等方面,则C语言的完美绝非是汇编语言所能比拟的。

1.2 C语言具有的突出优点

1. 语言简洁,使用方便灵活

C语言是现有程序设计语言中规模最小的语言之一,而小的语言体系往往能设计出较好的程序。C语言的关键字很少,ANSI C标准一共只有32个关键字,9种控制语句,压缩了一

切不必要的成分。C 语言的书写形式比较自由，表达方法简洁，使用一些简单的方法就可以构造出相当复杂的数据类型和程序结构。

2. 可移植性好

用过汇编语言的读者都知道，即使是功能完全相同的一种程序，对于不同的单片机，必须采用不同的汇编语言来编写。这是因为汇编语言完全依赖于单片机硬件。而现代社会中新器件的更新换代速度非常快，也许我们每年都要与新的单片机打交道。如果每接触一种新的单片机就要学习一次新的汇编语言，那么也许我们将一事无成。因为每学一种新的汇编语言，少则几月，多则一年，那么我们真正用于产品开发的时间就会更少。

C 语言是通过编译来得到可执行代码的，统计资料表明不同机器上的 C 语言编译程序 80% 的代码是公共的，C 语言的编译程序便于移植，从而使在一种单片机上使用的 C 语言程序可以不加修改或稍加修改即可方便地移植到另一种结构类型的单片机上去。这大大增强了我们使用各种单片机进行产品开发的能力。

3. 表达能力强

C 语言具有丰富的数据结构类型，可以根据需要采用整型、实型、字符型、数组类型、指针类型、结构类型、联合类型、枚举类型等多种数据类型来实现各种复杂数据结构的运算。C 语言还具有多种运算符，灵活使用各种运算符可以实现其他高级语言难以实现的运算。

4. 表达方式灵活

利用 C 语言提供的多种运算符，可以组成各种表达式，还可采用多种方法来获得表达式的值，从而使用户在程序设计中具有更大的灵活性。C 语言的语法规则不太严格，程序设计的自由度比较大，程序的书写格式自由灵活。程序主要用小写字母来编写，而小写字母是比较容易阅读的，这些充分体现了 C 语言灵活、方便和实用的特点。

5. 可以进行结构化程序设计

C 语言是以函数作为程序设计的基本单位的，C 语言程序中的函数相当于汇编语言中的子程序。C 语言对于输入和输出的处理也是通过函数调用来实现的。各种 C 语言编译器都会提供一个函数库，其中包含许多标准函数，如各种数学函数、标准输入输出函数等。此外，C 语言还具有自定义函数的功能，用户可以根据自己的需要编制满足某种特殊需要的自定义函数。实际上 C 语言程序就是由许多个函数组成的，一个函数即相当于一个程序模块，因此，C 语言可以很容易地进行结构化程序设计。

6. 可以直接操作计算机硬件

C 语言具有直接访问单片机物理地址的能力，可以直接访问片内或片外存储器，还可以进行各种位操作。

7. 生成的目标代码质量高

众所周知，汇编语言程序目标代码的效率是最高的，这就是为什么汇编语言仍是编写计算机系统软件的重要工具的原因。但是统计表明，对于同一个问题，用 C 语言编写的程序生成代码的效率仅比用汇编语言编写的程序低 10%~20%。

尽管 C 语言具有很多的优点,但是和其他任何一种程序设计语言一样也有其自身的缺点,如不能自动检查数组的边界,各种运算符的优先级别太多,某些运算符具有多种用途等。总的来说, C 语言的优点远远超过了它的缺点。经验表明,程序设计人员一旦学会使用 C 语言之后,就会对它爱不释手,尤其是单片机应用系统的程序设计人员更是如此。

1.3 AVR 单片机的 C 编译器简介

目前世界上几乎所有系列的单片机都支持 C 语言开发,开发 AVR 单片机的 C 编译器主要有 IAR Embedded Workbench(简称 IAR)、Codevision AVR(简称 CAVR)、Imagecraft C Compiler(简称 ICC)、GNU C For AVR(简称 GCCAVR)等。

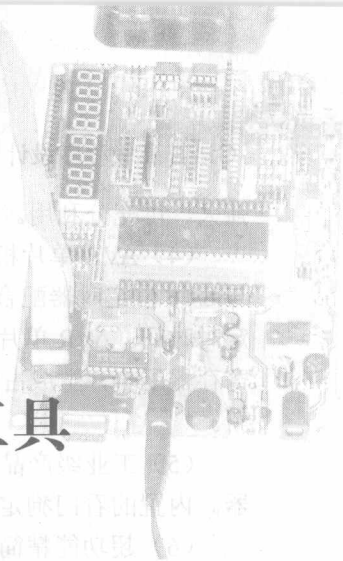
(1) IAR 是瑞典 IAR SYSTEMS 公司开发的 AVR 单片机集成开发环境 (IDE),包含嵌入式编译器、汇编器、连接定位器、库管理器、项目管理及调试器等。其特点是编译效率高、功能齐全,但价格昂贵。

(2) CAVR 也是一个开发 AVR 单片机的集成开发环境,其界面友好,很容易上手。它带有一个叫 Codewizard 的代码生成器,可生成外围器件的相应初始化代码,另外,它还提供了很多常用的器件库代码,如 LCD、UART、SPI、实时时钟、温度传感器等,其价格适中。

(3) ICC 是 Imagecraft 公司开发的使用标准 C 语言的 AVR 单片机集成开发环境,它有一个 Application Wizard 的代码生成器,也可生成外围器件的初始化代码,其价格适中。

(4) GCCAVR 是一个开源代码的自由软件,因此,使用时不必考虑价格因素,其缺点是没有集成开发环境 (IDE),使用时麻烦一些。

在本书中,我们使用 ICC 集成开发环境进行学习开发,ICC 集成开发环境软件能够产生形式简洁、效率较高的程序代码,当程序较大时在代码质量上可以与汇编语言程序相媲美。



第2章

AVR 单片机的主要特性及开发工具

2.1 AVR 单片机简介及主要特性

AVR 单片机是 ATMELE 公司研发的增强型内置 Flash 的 RISC (Reduced Instruction Set CPU) 精简指令集高速 8 位单片机, 设计时吸取了 80C51 及 PIC 单片机的优点, 具备单时钟周期执行一条指令的能力, 运行速度高达 1Mips/MHz。AVR 单片机可以广泛应用于计算机外部设备、工业实时控制、仪器仪表、通信设备、家用电器等各个领域。

AVR 单片机硬件结构采取 8 位机与 16 位机的折中策略, 即采用局部寄存器存堆 (32 个寄存器文件) 和单体高速输入/输出的方案 (即输入捕获寄存器、输出比较匹配寄存器及相应控制逻辑), 提高了指令执行速度, 克服了瓶颈现象, 增强了功能。同时, 又减少了对外设管理的开销, 相对简化了硬件结构, 降低了成本。AVR 单片机在软/硬件开销、速度、性能和成本诸多方面取得了优化平衡, 是一种高性价比的单片机。

AVR 单片机的主要特性如下所述。

(1) 内嵌高质量的 Flash 程序存储器, 可反复擦写, 支持 ISP 和 IAP, 便于产品的调试、开发、生产、更新。内嵌长寿命的 E²PROM 可长期保存关键数据, 避免断电丢失。片内具有大容量的 RAM, 有效支持使用高级语言开发系统程序。

(2) 高速度、低功耗, 具有 SLEEP (省电休眠) 功能。每一指令执行速度可达 50ns (20MHz), 而耗电则在 1~2.5mA 之间 (典型功耗, WDT 关闭时为 100nA), AVR 单片机运用 Harvard 结构概念 (具有预取指令功能), 即对程序存储和数据带有不同的存储器和总线, 当执行某一指令时, 下一指令被预先从程序存储器中取出, 这使得指令可以在每一个时钟周期内被执行。AVR 单片机可宽电压运行 (2.7~5.5V), 抗干扰能力强, 可降低一般 8 位机中的软件抗干扰设计的工作量和硬件的使用量。

(3) AVR 单片机的 I/O 线全部带可设置的上拉电阻, 并行 I/O 口输入输出特性与 PIC 的 HI/LOW 输出及三态高阻抗 HI-Z 输入类同外, 也可设定类同 80C51 系列内部拉高电阻作输入端的功能, 可单独设定为输入/输出、可设定 (初始) 高阻输入。使得 I/O 口资源灵活、功能

强大、可充分利用。AVR 的 I/O 口是真正的 I/O 口，能正确反映 I/O 口的输入/输出真实情况。

(4) AVR 单片机片内具备多种独立的时钟分频器，分别供 UART、IIC、SPI 使用。其中与 8/16 位定时器配合的具有多达 10 位的预分频器，可通过软件设定分频系数提供多种档次的定时时间。AVR 单片机中的定时器/计数器（单）可双向计数形成三角波，再与输出比较匹配寄存器配合，生成占空比可变、频率可变、相位可变方波的脉宽调制输出 PWM，令人耳目一新。

(5) 工业级产品，具有大电流 10~20mA 或 40mA（单一输出），可直接驱动 SSR 或继电器。内置的看门狗定时器（WDT）用于防止程序跑飞，提高产品的抗干扰能力。

(6) 超功能精简指令。具有 32 个通用工作寄存器（相当于 80C51 单片机中的 32 个累加器），克服了单一累加器在数据处理时造成的瓶颈现象。

(7) AVR 单片机内有模拟比较器，I/O 口可作 A/D 转换用，组成廉价的 A/D 转换器。

(8) 像 80C51 一样，AVR 有多个固定中断向量入口地址，因此，可快速响应中断，而不会像 PIC 一样所有中断都在同一向量地址，需要以程序判别后才可响应。

(9) AVR 单片机有自动上电复位电路、独立的看门狗电路、低电压检测电路 BOD，多个复位源（自动上下电复位、外部复位、看门狗复位、BOD 复位），可设置的启动后延时运行程序，增强了系统的可靠性。

(10) 有串行异步通信 UART，不占用定时器和 SPI 传输功能，因其速度高，故可以工作在一般标准整数频率，而波特率可达 576kb/s。

(11) 有多通道 10 位 A/D 转换器及实时时钟 RTC。

AVR 单片机技术表现出单片机集多种器件（包括 FLASH 程序存储器、看门狗、E²PROM、同/异步串行口、TWI、SPI、A/D 模数转换器、定时器/计数器等）和多种功能（增强可靠性的复位系统、降低功耗抗干扰的休眠模式、品种多门类全的中断系统、具输入捕获和比较匹配输出等多样化功能的定时器/计数器、具替换功能的 I/O 端口……）的特点于一身，充分体现了现代单片机技术向“片上系统 SoC”过渡及发展的方向。

2.2 学习开发 AVR 单片机用到的实验工具及器材

学习一种新的单片机技术，不能只是纸上谈兵，实验与实践是必不可少的。这里我们使用以下的器材进行 AVR 单片机的 C 语言设计。

(1) ImageCraft 公司的 ICCAVR6.31A C 语言编译器。

(2) Atmel 公司的 AVR Studio 集成开发环境。

(3) PonyProg2000 下载软件。

(4) AVR DEMO 单片机综合试验板。

(5) AVR 单片机 JTAG 仿真器。

(6) 并口下载器。

- (7) 5V 高稳定专用稳压电源。
- (8) TOP2004 多功能 USB 编程器 (可选购)。
- (9) 一台奔腾级及以上的家用电脑 (PC 机)。

下面简单介绍一下这些实验工具及器材。

2.2.1 ICCAVR6.31A C 语言编译器

ICCAVR6.31A 是 ImageCraft 公司开发的用于 AVR 单片机的 C 语言编译器, 是一个综合了编辑器和工程管理器的纯 32 位集成开发环境 (IDE)。由于 ICCAVR 功能强大, 使用简单方便, 具有良好的技术支持且价格合适, 所以得到了广泛的应用。图 2-1 为 ICCAVR 的工作界面。

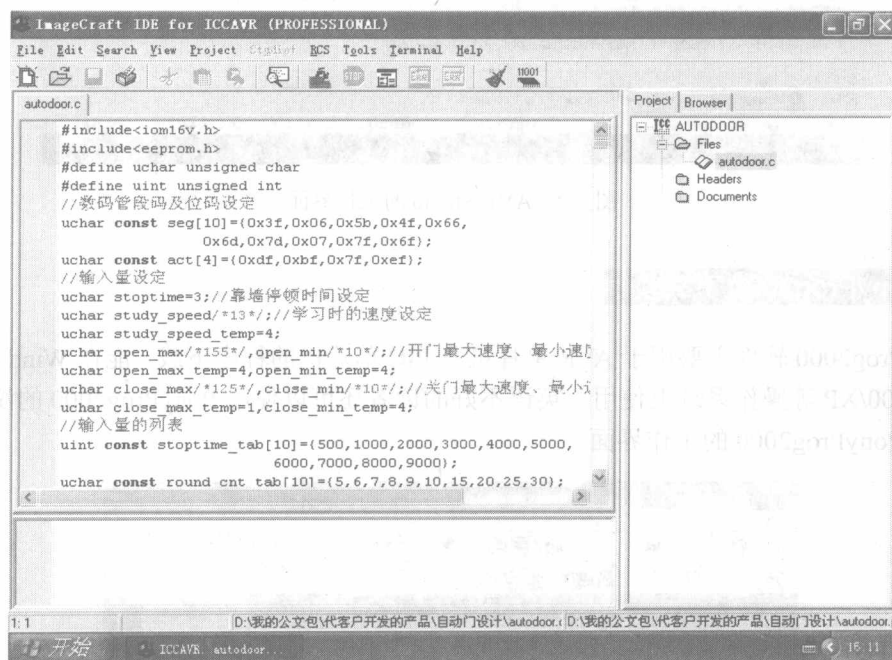


图 2-1 ICCAVR 的工作界面

2.2.2 AVR Studio 集成开发环境

AVR Studio 是一个 ATMEL 公司开发的集项目管理、程序汇编、程序调试、程序下载、JTAG 仿真等功能于一体的集成开发环境。但 AVR Studio 不支持 C 语言编译, 因此当我们用 C 语言开发 AVR 单片机时, 需要先用 ICCAVR 编写 C 语言并进行编译, 然后使用 AVR Studio 打开编译生成的*.cof 文件, 进行程序的仿真调试。图 2-2 为 AVR Studio 的工作界面。