



面向“十二五”高等学校精品规划教材·机电类

AVR单片机原理与应用

李晓锋 主编



本书含配套光盘

 北京理工大学出版社
BEIJING INSTITUTE OF TECHNOLOGY PRESS

面向“十二五”高等学校精品规划教材·机电类

AVR 单片机原理与应用

李晓锋 主编

 **北京理工大学出版社**
BEIJING INSTITUTE OF TECHNOLOGY PRESS

内 容 简 介

本书以 Atmel 公司 AVR 单片机 Atmega48 为载体, 由浅入深, 全面介绍了初学单片机所必须掌握的相关专业知识, 同时配以大量的图示和实例程序, 力求深入浅出, 使读者通过实际动手操作在愉悦中完成专业知识的学习和应用。

本书首先介绍了计算机的基础知识, 然后介绍以 GCC + AVRStudio 作为 AVR 的开发环境, 接着循序渐进地介绍了 AVR 单片机的常用片内模块的结构原理和应用设计方法。

本书附带光盘收录了书中 AVR 开发中所需用到的软件工具、所有实例的源代码、芯片技术资料、相关技术规范 and 协议, 以及 Atmel 公司针对 AVR 的应用笔记资料。

本书可作为高等学校电子、自动化、机电一体化等相关专业的单片机原理课程的教材, 也可作为 AVR 单片机的培训教材, 供相关技术人员学习参考。

版权专有 侵权必究

图书在版编目 (CIP) 数据

AVR 单片机原理与应用/李晓锋主编. —北京: 北京理工大学出版社, 2010. 2

ISBN 978 - 7 - 5640 - 2971 - 5

I. ①A… II. ①李… III. ①单片微型计算机 IV. ①TP368. 1

中国版本图书馆 CIP 数据核字 (2010) 第 001475 号

出版发行 / 北京理工大学出版社

社 址 / 北京市海淀区中关村南大街 5 号

邮 编 / 100081

电 话 / (010)68914775(办公室) 68944990(批销中心) 68911084(读者服务部)

网 址 / <http://www.bitpress.com.cn>

经 销 / 全国各地新华书店

印 刷 / 北京地质印刷厂

开 本 / 787 毫米 × 1092 毫米 1/16

印 张 / 18.5

字 数 / 430 千字

版 次 / 2010 年 2 月第 1 版 2010 年 2 月第 1 次印刷

印 数 / 1 ~ 2000 册

定 价 / 39.00 元

责任校对 / 陈玉梅

责任印制 / 边心超

图书出现印装质量问题, 本社负责调换

出版说明

近年来，我国高等教育的改革和发展实现了历史性的跨越，培养了大量人才，为我国经济的发展作出了巨大的贡献，但从 IMD 国际竞争力指标体系中的分析数据来看，我国企业需要的工程技术人员特别是工程应用型技术人才严重不足，这也热切地呼唤着高等院校培养出更多具备全面的知识、能力和综合素质，面向生产、建设、管理、服务第一线的高级应用型专门人才。教育部在 2003 年启动了本科教学评估工作，并在 2007 年提出了本科教育、教学“质量工程”，鼓励和支持高等学校在教学理念等方面进行创新，形成有利于多样化人才成长的培养体系，满足国家对社会紧缺的创新型人才和应用型人才的需要。

北京理工大学出版社组织知名专家、学者，以培养应用型人才为主题进行深入的研讨，规划出版了这套“面向‘十二五’高等学校精品规划教材·机电类”。着力于培养能直接从事实际工作、解决具体问题、维持工作有效运行的高等应用型人才。

本套教材在规划过程中体现了如下基本原则和特点：

- 学科体系完整，课程间相互衔接紧密。

本套教材根据工程实践需要，按教学体系要求进行整合编排。包括了机电类专业的基础课、专业基础课和部分专业课。除了考虑单门课程自身体系的完整，兼顾不同课程间的衔接。

- 强调实用性和工程概念。

工程的概念体现在整套教材中，以工程实践要求为核心编写教材。

- 减少了部分理论推导方面的内容。

强调概念和应用，减少了部分理论推导。在实验环节强调创新型的实验，减少验证型的实验。

- 结合新技术和新工艺。

充分吸收新技术和新工艺的内容，反映国内外机械学科最新发展。

- 注重培养学生职业能力。加强学生对 Autocad、UG、Pro/E、Mastercam 等

软件进行设计和仿真的能力。

- 提供教学包，可在北京理工大学出版社网站 [www. bitpress. com. cn](http://www.bitpress.com.cn) 下载。

本套教材既严格遵照学科体系的知识构成和教材编写的一般规律，又针对本科人才培养目标及与之相适应的教学特点，精心设计写作体例，科学安排知识内容，表达了一批教育工作者和出版人“精心打造精品，教材服务教育”的理念。

本套教材可作为高等教育应用型本科院校机电类相关专业的课程教学用书，也可以作为机电类技能培训用书。

北京理工大学出版社

前 言

作者连续多年担任“单片机原理与应用”课程的教学工作，教学中深深体会到单片机是当今电子技术的一个极为重要的组成部分，一个电子类专业的学生，如果毕业后还没能掌握单片机，根本算不上一个合格的毕业生。另一方面，也深感目前高校的教学内容和教材已经跟不上技术的发展和市场的需要。

目前，大部分学校仍以 20 世纪 80 年代开始流行的 MCS-51 系列单片机为载体，以汇编语言为编程工具，以并行扩展为核心讲述单片机的原理及应用。尽管 MCS-51 在实际应用中还占有相当大的市场，但随着电子技术的迅猛发展，各种新型数据传输技术的出现和新器件的推出，传统 51 单片机由于自身结构的原因，在数据处理能力、系统扩展方面等多个方面显得不能适应新技术的发展，而且通过汇编语言来介绍单片机技术，对于电子类的学生，显得难度太大，并且容易偏离介绍单片机技术这个主题，这些原因使得选用 51 单片机作为载体讲授单片机技术已不再是最好的选择。

AVR 作为增强型基于 RISC 结构的单片机，在运行速度、内存容量，内部集成功能模块、基于串行接口的外围扩展、高级语言编程支持、仿真调试技术等诸多方面都比传统 51 单片机先进，比较充分、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。因此本书选择了 AVR 单片机系列的 Atmega48 为载体来讲述单片机软件开发中的基础知识。

目前，AVR 的高级语言开发中，有两种业界广泛使用的 C 语言编译器：IAR 和 GCC 其中使用 IAR 需要高昂的价格购买许可，而 GCC 作为自由软件很自然作为教学中的首选。但比较遗憾的是尚未发现一本以 GCC 作为编译器的适合高职类学生的 AVR 单片机教材本书力图填补这一空白。

本书是作者在多年的教学实践和大量实际应用经验积累的基础上，作为“单片机原理与应用”的教材而编写的，作者希望它能促进国内单片机教学的发展，为培养更多高质量、应用型人才做出微薄的贡献。

全书共分为 11 章，第 1 章介绍单片机入门所需要的一些基础知识，第 2 章从整体上对 AVR 单片机做一个概要性的介绍，第 3 章介绍 AVR 单片机开发所需要的软硬件环境，是学习后续章节的重要基础；第 4 章~第 7 章介绍 AVR 单片机的基本接口和常用功能模块，包括 I/O 口的应用、中断、定时器、A/D 等，这些内容是所有单片机应用的基础，是学习单片机技术的重中之重。第 8 章讲述单片机通用程序设计，主要目的在于介绍单片机程序中一种通用而规范的设计方法；第 9 章~第 11 章对串行通信和串行接口进行了详细的介绍，包括了 USART、SPI、I²C 等；这些串行接口在目前的单片机开发中有着广泛的应用。

本书在每一章节的编写中，设计了大量实例，希望读者通过对这些实例的实践，将枯燥乏味的学习过程变得轻松、有趣，充分享受学习单片机的乐趣和成功的快乐；并且使读者在单片机学习过程中，花费尽可能少的时间和精力，掌握尽可能多的单片机理论知识和实际的

开发技术。

本书附带光盘提供了书中 AVR 开发中所需用到的软件工具、所有实例的源代码、芯片技术资料、相关技术规范 and 协议，以及 Atmel 公司针对 AVR 的应用笔记资料。

在此感谢北京理工大学出版社全体工作人员对出版本书的关心和支持；感谢学生和同事对本书的建议和支持；感谢父母在我编写这本书时对我的支持和关心；尤其要感谢我的爱人，为了写书，在她身怀六甲期间少了很多陪她的时间，谢谢她的理解、支持和关爱。

虽然作者已有多年单片机教学和开发经验，也力求编写一本好书，但因时间仓促和个人水平有限，错误和不足之处在所难免，敬请读者批评指正。

作 者

目 录

引论 单片机技术引论	1
第一节 认识单片机	1
第二节 计算机中信息的表示	3
第三节 计算机如何工作	11
第一章 AVR 单片机概述	13
第一节 AVR 单片机简介	13
第二节 单片机的基本组成	14
第三节 ATmega48 单片机的组成	18
第四节 ATmega48 的存储器结构和地址空间	23
第二章 AVR 单片机的开发环境	25
第一节 AVR 开发环境的组成	25
第二节 AVR 高级语言开发平台	25
第三节 AVR Studio 集成开发环境	28
第四节 AVR 单片机的最小系统	32
第五节 构建自己的 AVR 实验板	35
第六节 AVR 可执行代码的下载	39
第三章 通用 I/O 端口的结构与应用	46
第一节 通用 I/O 端口的结构	46
第二节 AVR 单片机 I/O 端口的编程	47
第三节 实例 1 跑马灯	49
第四节 端口的位操作	54
第五节 数码管显示	59
第六节 实例 2 一位数码管的静态显示	60
第七节 多位 LED 数码管显示	64
第八节 实例 3 动态扫描的多位数码管显示	64
第九节 实例 4 用延时实现的时钟系统	68
第十节 键盘输入接口	72
第十一节 实例 5 矩阵键盘	75
第四章 中断系统及其基本应用	79
第一节 什么是中断	79
第二节 中断的特性	80
第三节 ATmega48 单片机中断源和中断向量	84

第四节	如何编写中断服务程序代码	85
第五节	AVR 单片机中断的开关控制	86
第六节	AVR 单片机中断标志位	87
第七节	AVR 的中断优先级	88
第八节	ATmega48 的外部中断	88
第九节	实例 6 按键控制的 1 位 LED 数码管显示	91
第五章	定时/计数器的结构与应用	95
第一节	单片机定时的原理	95
第二节	ATmega 48 的定时器 0 (Timer/Counter0)	98
第三节	编写第一个定时器程序	107
第四节	重加载的方法控制定时中断周期	108
第五节	CTC 模式控制定时中断周期	109
第六节	实例 7 采用 T/CO 硬件定时器的数字时钟	109
第七节	实例 8 用单片机演奏音乐	111
第八节	PWM 波及其应用简介	115
第九节	实例 9 会呼吸的灯	117
第六章	ADC 接口	121
第一节	逐次比较式 A/D 转换器的工作原理	121
第二节	ATmega 48 的 A/D 转换器	123
第三节	ATmega 48 单片机中与 A/D 相关的引脚	124
第四节	ATmega 48 单片机中与 A/D 相关的寄存器	125
第五节	ADC 应用要点	129
第六节	编写 ADC 转换程序	132
第七节	实例 10 简易数字电压表	133
第八节	ADC 噪声抑制	137
第七章	单片机通用程序设计	140
第一节	实例 11 电子时钟设计	140
第二节	有限状态机的概念及其实现方法	183
第八章	串行 SPI 接口应用	190
第一节	串行接口与串行通信基础知识	190
第二节	SPI 串行总线介绍	192
第三节	AVR 的 SPI 接口原理	198
第四节	与 SPI 相关的寄存器	200
第五节	SPI 接口设计的应用要点	202
第六节	编写 SPI 基本操作函数	203
第七节	实例 12 端口扩展	204
第九章	异步通信与 USART 接口基础	212
第一节	异步通信的基本概念	212

目 录

第二节	AVR 的异步传输接口 USART	215
第三节	USART 的基本操作	222
第四节	基于 USART 接口基本通信的实现与测试	226
第五节	实例 13 USART 自发自收测试	226
第六节	AVR 与 PC 机的串口通信	229
第七节	采用通信缓冲区的 USART 底层驱动编写	234
第八节	利用标准 I/O 流调试程序	236
第十章	串行 TWI (I²C) 接口应用	240
第一节	I ² C 串行总线介绍	240
第二节	AVR 的 TWI 接口与使用	245
第三节	使用 TWI 总线	250
第四节	基于状态机的 TWI 通信模块	253
第五节	实例 14 存储器 24C02 的读写	274
附录 I	ATmega48 常用熔丝的作用及配置方法	281
附录 II	Intel HEX 文件格式	284
参考文献	286

引论

单片机技术引论

终于要开始学单片机了，也许大家心中迷惑重重，什么是单片机？为什么学单片机？单片机有什么用？本章中我们首先就来回答这些问题，以帮助大家认识单片机。然后介绍计算机中信息的表示和处理有关的基础知识，概略地讨论计算机的体系结构，计算机是如何工作的。以帮助大家掌握学习单片机所需要的一些软、硬件基础知识。

第一节 认识单片机

一、什么是单片机

我们都知道计算机是什么样子——它就是在你的桌面上单调地咕嘟咕嘟叫（如果风扇坏掉了则咔嗒嗒地响）的箱子。这个箱子里有运行着的软件、你存储的信息并把你与外部世界连接起来的电子设备。如果你走在大街上随便问一个人家里总共有多少台计算机，他也许会回答家里有那么一两台。可是，如果你告诉他，他家里可能有 30 台计算机，甚至更多，这时这个人就会露出迷惑的眼光，或者轻微发怒道：别到这里忽悠我了。然而实际上，这些计算机确实在那里，它们可能藏在电视机、录像机、DVD 播放器、遥控器、手机、洗衣机、玩具以及其他一些设备里。这些计算机就是我们这本书要讨论的对象——单片机。

那么什么是单片机呢？简单地说，单片机就是一个芯片上的微型计算机系统，它将计算机系统的主要功能部件——中央处理器 CPU（Central Processing Unit）、存储器（Memory）、I/O（Input/Output）接口电路等集成到了一块集成电路芯片上。中文“单片机”的称呼就是由英文名称“Single Chip Microcomputer”直接翻译而来的。

单片机主要应用于工业控制领域，面向测控对象，突出控制功能。它从功能和形态上来说都是应控制领域应用的要求而诞生的。随着单片机技术的发展，在芯片内集成了许多面向测控对象的接口电路，如 ADC、DAC、高速 I/O 口、PWM、WDT、SPI、I²C、CAN 等。所以，更正式的单片机被称为微控制器 MCU（Micro Controller Unit）。另外由于单片机芯片的微小体积、极低成本和面向控制的设计，使得它通常作为控制和/或监控的目的而嵌入到另外的专用系统中，因此又被称为嵌入式微控制器（Embedded Microcontroller）。但因为在国内多年来习惯了单片机的叫法，所以我们在本书中也使用“单片机”一词。

二、为什么学单片机

在知道什么是单片机后，再来说说为什么学单片机。也许不同的人对这个问题有不同的答案，如果要写下这些理由，可能可以洋洋洒洒写出几千言。不管答案如何，这些理由最终都可以归结为几个字：有趣、神奇、有用。

先来说说有趣，因为单片机本身的特点，决定了它是新时代电子爱好者的“玩物”。在过去由于条件所限，今天已五六十岁的电子爱好者用简陋的元件制造出只能用耳机听音的矿石收音机，这曾令他们兴奋不已；三四十岁的电子爱好者用半导体分立元件制作再生来复式或超外差晶体管收音机，也曾让他们享受一把成功的喜悦；而当今新生代电子爱好者却拥有一个前所未用的大好时机和廉价丰富的物质条件，不仅可以用芯片制作集成电路收音机，还可以用单片机制作许多带智能的小电器，可以更容易地圆自己一个创新发明和创造专利的成功之梦。一只固化有专用软件的单片机芯片，配上一只液晶显示屏和几只小按钮，再装入一只小塑料壳，就可构成一只妙趣无穷的电子宠物（是一位日本女工程师发明的），其成本只不过几元，但市场售价竟一度高达一二百元。理由在于它是具备高科技背景的产品，技术含量高，其中的软件凝聚着开发者的聪明和智慧。不仅如此，我们可以用单片机完成各种各样的创意电子产品，这可以给我们的生活带来很多的趣味性。

对于一名单片机初学者或电子爱好者，一旦掌握了单片机的理论知识和开发应用技术，就进入了一个崭新而又广阔的创作天地，任由自己去发挥想象力和创造灵感，使得我们不仅能够充分享受到成就感，而且可以提高自己的业务素质，增强自己的创新能力，增加自己的就业机会，而对于单片机，定会有相见恨晚之感。

当然，要学好单片机，与电子制作中常用的 TTL 或 CMOS 通用数字集成电路以及其他专用集成电路（ASIC）相比，它掌握起来不太容易，问题在于单片机具有智能化功能，不光需要学习其硬件电路，还需要学习其特有的指令系统、配置的语言工具和开发环境软件、配套的硬件仿真器和程序烧写器等工具，而且软件设计需要有一定的创造性。这虽然给学习单片机的人带来一定的难度，但这也正是它的迷人之处。创作者可以把单片机作为一种载体，将自己的知识和智慧嵌入和固化其中，不仅可以创造自己知识产权和专利技术，还能够使其数倍甚至数十倍地升值，来创造社会效益和经济效益。

再来说说单片机的神奇和有用性。今天，我们的生活和工作环境中有着越来越多称之为单片机的小电脑为我们服务，可我们并没有意识到这些“小精灵”的存在。比如：当我们每天用遥控器操纵电视机或 VCD 享受其丰富功能的时候，正是单片机在接收我们的遥控命令；单片机在寻呼机和移动电话手机中也发挥着不可替代的重要作用；就连曾经一度令许多青少年朋友痴迷的电子宠物，也是单片机在大显神威。时下，家用电器和办公设备的网络化、智能化、遥控化、模糊控制化已成为发展趋势，而这些高性能几乎无一不是靠单片机来实现的。单片机能够在如此广阔的领域发挥其神奇的功能，不能不令人感叹——太神奇、太有用了。同时也以事实向我们表明，如果我们不具备单片机方面的知识、不掌握单片机的应用技术，对这些电器设备的日常保养和故障维修都会形成很大的障碍，就更不用说设计和开发以单片机为控制核心的各种电子电路和电器产品了。

三、单片机有什么用

再来了解一下单片机的应用领域。提到单片机的应用，有人这样说，“凡是能想到的地方，单片机都可以用得上”，这并不夸张。由于全世界单片机的年产量数以十亿计（根据权威调查机构公布，2001年世界单片机产量高达70亿片，仅仅中国大陆的年需求量就高达6亿片左右），应用范围之广，花样之多，一时难以详述，这里仅列举一些典型的应用领域或场合供大家参考。

(1) 电信产品：程控电话交换机、台式电话机、无绳电话机、投币电话机、磁卡电话机、光卡电话机、数字寻呼机、汉字寻呼机、数字蜂窝移动通信手持机（简称手机）、彩屏手机、无线对讲机、业余无线电台、传真机、调制解调器、通话计费器、电话密码锁、来电显示器（Caller ID）等。

(2) 家用电器：遥控电视机、录像机、VCD、CD、LD、语言复读机、MP3播放器、卫星电视接收器、音响空调控制器、卡拉OK点唱机、胶卷照相机、数码照相机、全自动洗衣机、电冰箱、空调机、洗碗机、微波炉、电饭煲、面包机、热水器、万年历、智能充电器、各种安防报警器、电卡电度表等。

(3) 计算机外围设备：键盘、鼠标、软盘驱动器、硬盘驱动器、光盘驱动器、移动硬盘、U盘（USB接口存储器）、显示器、打印机、绘图机、扫描仪、智能终端、智能扩充卡、调制解调器MODEM等。

(4) 办公自动化：复印机、传真机、考勤设备、智能打字机、电子词典、电子计算器、个人数字助理PDA等。

(5) 工业控制领域：自动生产线、数控机床、智能机器人、可编程顺序控制、电机控制、变频控制、过程控制、温度控制、智能传感器等机电一体化系统中。

(6) 商用电子领域：自动售货机、自动柜员机、智能广告牌、电子收款机、智能卡、IC卡读写器等。

(7) 电子玩具：袖珍游戏机、电子宠物、智能玩具、遥控玩具、学习玩具等。

(8) 仪器仪表：用于医疗、化工、电子、计量、实验等领域的各种智能仪器仪表。

(9) 汽车电子：点火控制、变速控制、防滑控制、防撞控制、排气控制、最佳燃烧控制、计程计费、防盗报警、电子地图、车载通信装置等。

(10) 军用电子：各种导弹、鱼雷的精确制导控制、智能武器、雷达系统、电子战飞机等。

第二节 计算机中信息的表示

通过前面的介绍，我们知道了单片机在本质上就是一个计算机，因此我们的学习从计算机开始。今天，计算机几乎已经无所不能了，计算机的使用遍及科学计算、商务处理、通信、工业控制、娱乐等各个领域，可以说我们手头上的绝大部分工作都可用计算机来代劳。在惊叹其无比强大功能的同时，我们自然禁不住要问：它为什么能这么神奇，发挥如此巨大的神力应用在如此广泛的行业？它如何工作？

简单来说,计算机是一台设计用来处理、存储和重新获取数据的机器。这些数据可以是电子表格的数字,一个文档中文本的字符,图像中颜色的像素,声音的波形或者网络游戏的人物角色。所有的这些数据,系统中的所有信息都是作为数字来存储的。

计算机通过对这些数字执行操作来实现不同的功能。比如,计算机通过把一个数字阵列传送到显存中来完成一个图像在屏幕上的显示,每个数字代表图像的一个像素;要播放 MP3 音频文件,计算机就从磁盘读一串数字到存储器,然后对这些数字进行巧妙地处理,把压缩了的硬盘数据转换成原始音频数据,并将这些新的数字传送到音频芯片中,动人的音乐就在音箱中播放出来。

总之,计算机所做的任何事情,从网页浏览到视频播放,都仅仅是对数字的转移和处理,计算机的电子设备只不过是设计用来保存、转移和改变这些数字的一个系统而已。这就是计算机的神奇之源,所以关键在于对这些数字的理解,以及如何处理这些数字。这涉及两个方面,信息的表示和信息的处理。

我们先来讨论第一个方面——计算机中信息的表示。可以说这也是了解计算机的第一个很重要、也是很基础的问题,对这个问题的理解将帮助我们对计算机系统的认识。

在计算机的世界里,很容易通过穿孔卡片上的有洞和无洞、电平的高或低、或者引起磁场的顺时针或逆时针来表示、存储和传输二值信号。所以构造存储和处理信息的计算机时,计算机中的一切信息都是以二值信号来表示,这些二值信号常以某种形式的“二值位序列”出现。

信息有诸多类型,下面我们就逐一讨论一些常见而基本的信息是如何以“二值位序列”的形式在计算机中表示的。

一、数制及其转换

计算机中最基本的信息就是“数”。所以,我们从最基本的“数”的表示讲起。在计算机中数字的机器表示方式与常用的整数和实数表示方式是不同的,它使用了不同的进制。

为了计数,人们发明了 0, 1, 2, 3, …, 9 这样的符号,但当需要记更大的数时,就遇到了严重的困难,因为要表示的数太多了,这时要么发明更多的符号(这显然不可能),要么发明更聪明的办法。随着文明的进步,人们逐渐掌握并发明了一个特殊的方法来表示大的数——位置计数法。这种方法以一个特殊的位置来代表更大的一组对象,比如数“111”,这个记法中,只出现了符号“1”,但却表示了更大的数,其中符号“1”因为位置的不同代表了不同的大小。这个发明,避免了发明无穷多的数字符号,解决了人类因计数的需要而面临的问题。

基数 b (或者 b 进制) 的位置计数法由规则

$$\begin{aligned} & (\cdots a_3 a_2 a_1 a_0 \cdot a_{-1} a_{-2} \cdots)_b \\ & = \cdots + a_3 b^3 + a_2 b^2 + a_1 b^1 + a_0 + a_{-1} b^{-1} + a_{-2} b^{-2} + \cdots \end{aligned}$$

来定义,例如, $(520.3)_6 = 5 \times 6^2 + 2 \times 6^1 + 0 + 3 \times 6^{-1} = 192.5$ 。当然,我们通常使用的十进制计数法,基数 b 等于 10, a 取值范围是“十进数” 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 这种情况下,下标 b 可以省略。

1. 二进制数制

由于在计算机中一切信息都是由二值位序列表示的,所以在计算机中表示“数”最自然的数制就是二进制。当基数 b 等于 2 时,我们就得到了二进制数,这时诸 a 只需取两个数字: 0 和 1。这两个数字 0 和 1,通常称为比特 (bit),即二进制的“位”。

二进制数 $(1011.11)_2$ 所表示的值为

$$(1011.11)_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2}$$

在表示二进制数时,为了书写的方便和不致引起混乱,二进制数用后缀字母 B 作标记,如二进制数 $(1110)_2$ 记为 1110B。

2. 十六进制数制

由于在书写大的二进制数时,需要写一长串的 0, 1, 为了避免这个麻烦,人们发明了十六进制。当基数 b 等于 16 时,我们就得到了十六进制数。这时诸 a 需要 16 个数码,通常用 0, 1, 2, ..., 9 和 A, B, C, D, E, F 表示,如 $(24A4)_{16}$ 。其中字母 A 到 F 既可使用大写字母也可使用小写字母。

例如,十六进制数 $(5EC.D4)_{16}$ 所表示的值为

$$(5EC.D4)_{16} = 5 \times 16^2 + 14 \times 16^1 + 12 \times 16^0 + 13 \times 16^{-1} + 4 \times 16^{-2}$$

十六进制相对二进制的优点在于,用十六进制书写相同的数时,所用数位比二进制数大大缩短,便于书写和记忆;另一方面,由于后面马上要提及的二进制数和十六进制数之间方便的转换关系,为了缩短书写二进制数的长度,我们没有采用习惯的十进制数。

为了不致引起误解和表示的方便,十六进制数通常加后缀 H,如 $(3F)_{16}$ 记为 3FH,或者加前缀 0x,记为 0x3F。若采用后缀的表示法,对于以字母开头的十六进制数,必须带有前缀 0,以与一般的字符串区别,如 $(FE)_{16}$ 记为 0FEH。通常在 C 语言中我们以前缀的形式表示十六进制数。

3. 二进制与十进制数之间的转换

将二进制数转换成十进制数,非常容易,只需按照下式将二进制数按权展开即可。

$$\begin{aligned} & (\dots a_3 a_2 a_1 a_0 \cdot a_{-1} a_{-2} \dots)_2 \\ & = \dots + a_3 2^3 + a_2 2^2 + a_1 2^1 + a_0 + a_{-1} b^{-1} + a_{-2} b^{-2} + \dots \end{aligned}$$

例如, $1101B = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 13$ 。

而将十进制数转换成二进制数,一种方法^①是将十进制数以及所得到的商数不断除以 2,直至商数为零,依次记下每次的余数。这样所得到的余数从低位到高位排列,便得到二进制数。

例如,将 $(25)_{10}$ 转换成二进制数,方法如下所示。

$$\begin{array}{r} 2 \overline{) 25} \quad 1 \quad \text{最低位 (LSB)} \\ \underline{20} \\ 5 \\ 2 \overline{) 5} \quad 0 \\ \underline{4} \\ 1 \\ 2 \overline{) 1} \quad 1 \\ \underline{0} \\ 0 \end{array}$$

① 因为后面讨论中只需用到整数,这里只介绍整数的转换方法。

即 $25 = 11001\text{B}$ 。

4. 二进制与十六进制之间的转换

很容易看出，在 b 进制和 b^k 进制之间的一个简单关系

$$(\cdots a_3 a_2 a_1 a_0 \cdot a_{-1} a_{-2} \cdots)_b = (\cdots A_3 A_2 A_1 A_0 A_{-1} A_{-2} \cdots) b^k$$

其中

$$A_j = (a_{kj+k-1} \cdots a_{kj+1} a_{kj})_b$$

于是，我们一眼就能看出 2 进制与 16 进制之间的简单转换技术。在 b 等于 2， b^k 等于 16 时， k 等于 4，那么上面 A_j 即可写成

$$A_j = (a_{4j+3} a_{4j+2} a_{4j+1} a_{4j})_2$$

这就是说，将二进制数整数部分从个位开始，每 4 位一组，划分整数部分（如果最后一组不足 4 位，可在前面补 1~3 个零）；对小数部分，从 a_{-1} 位开始，每 4 位一组，划分小数部分（同样，但最后一组不足 4 位时，可在后面补 1~3 个零）。然后把每组中的 4 位二进制数用对应的十六进制数表示，即可获得十六进制数。

例如：

$$\begin{aligned} 1110010101.10101\text{B} &= 0011 \quad 1001 \quad 0101. \quad 1010 \quad 1000 \\ &= 3 \quad 9 \quad 5. \quad \text{A} \quad 8 \end{aligned}$$

即 $1110010101.10101\text{B} = 395.\text{A}8\text{H}$ 。

十六进制转换为二进制时，方法与上类似，将十六进制数的整数部分和小数部分的每一位十六进制数码用对应的 4 为二进制数表示，然后删除整数部分前面和小数部分后面多余的零，即可获得对应的二进制数，如：

$$\begin{aligned} 3\text{E}.\text{CH} &= 00111110.1100\text{B} \\ &= 111110.11\text{B} \quad (\text{删除多余的零}) \end{aligned}$$

可见，二进制数与十六进制数之间转换非常方便，只要记住 4 位二进制数 0000~1111 与十六进制数 0~F 之间的对应关系即可。表 0-1 列出了 0 到 15 的二进制、十进制、十六进制表示。

表 0-1 0 到 15 的二、十、十六进制表示

十进制	二进制	十六进制	十进制	二进制	十六进制
0	0000	0	8	1000	8
1	0001	1	9	1001	9
2	0010	2	10	1010	A
3	0011	3	11	1011	B
4	0100	4	12	1100	C
5	0101	5	13	1101	D
6	0110	6	14	1110	E
7	0111	7	15	1111	F

对于上表的一个记忆窍门是，记住十六进制数字 A、C 和 F 对应的十进制值，而把十六进制数字 B、D 和 E 翻译成十进制值，利用它们与前三个值的相对关系很容易就可记下这个表格中数字的对应关系了。

5. 整数的表示

现在让我回到数字的机器表示方式这个问题上来，讨论一下编码整数的两种不同的方式：一种只能表示非负数，而另一种能够表示负数、零和整数。在 C 语言里，我们把前者称为无符号数，后者称为有符号数。

1) 整型数据类型

在后面单片机程序的设计中，我们要用到 C 语言，因此这里先讨论一下 C 所支持的整型数据类型。C 支持多种整型数据类型，不同的类型所能表示的整数范围不同，这些类型如表 0-2 所示。每种类型都有一个大小指示符：char、short、int 和 long，同时还有一个数字为非负数的指示符 unsigned，默认情况下（不加 unsigned 修饰）数字都是有符号的。表 0-2 给出了对这些不同整型数据类型的 C 标准和 AVR 中的大小分配，C 标准定义了每种类型必须能够表示的最小数值范围。

表 0-2 C 的整型数据类型

C 声明	保证的		AVR 中整型数据类型	
	最小值	最大值	最小值	最大值
char	-127	127	-128	127
unsigned char	0	255	0	255
short [int]	-32 767	32 767	-32 768	32 767
unsigned short [int]	0	65 535	0	65 535
int	-32 767	32 767	-32 768	32 767
unsigned [int]	0	65 535	0	65 535
long [int]	-2 147 483 647	2 147 483 647	-2 147 483 648	2 147 483 647
unsigned long [int]	0	4 294 967 295	0	4 294 967 295

2) 无符号整数的表示

对于一个 w 位的“二进制位串”，我们可以将其记为 $[x_{w-1}, x_{w-2}, \dots, x_0]$ ，表示“位串”中的每一位，或者简记为向量 \mathbf{x} 。把 \mathbf{x} 看成一个二进制表示的数，我们就得到了 \mathbf{x} 的无符号数表示。由前面对二进制的讨论可知， \mathbf{x} 表示的无符号数的大小，用函数 $B2U_w(\mathbf{x})$ （代表“二进制数到无符号数的转换”，长度为 w ）表示，可由下式决定：

$$B2U_w(\mathbf{x}) \stackrel{\text{def}}{=} \sum_{i=0}^{w-1} x_i 2^i$$

在这个等式中，函数 $B2U_w(\mathbf{x})$ 将一个长度为 w 位的“二进制位串”映射到了非负整数。很容易可以看出，它的最小值用位串 $[00\dots 0]$ 表示，也就是整数 0，它的最大值由位串 $[11\dots 1]$ 表示，也就是整数值 $BMax_w \stackrel{\text{def}}{=} \sum_{i=0}^{w-1} 2^i = 2^w - 1$ 。