

高等职业教育教材

单片机应用技术 项目化教程

■ 丛书主编 吴建宁 ■ 主编 姚晓平



本书配有电子教学参考资料包



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

高等职业教育教材

单片机应用技术项目化教程

姚晓平 主 编
张 平 副主编

電子工業出版社

Publishing House of Electronics Industry

北京 · BEIJING

内 容 简 介

本书以 51 系列单片机为主体,共设置了 5 个项目 14 个任务,通过对霓虹灯的设计与制作,电子钟的设计与制作,测量仪表的设计与制作,通信口应用与控制的设计与制作,微波炉控制系统的设计与制作任务的讲解,实现从产品概念、设计、制作的全过程训练。结合大量的软、硬件实例详细介绍单片机应用的基础知识和条件。

本书以 C 语言作为编程语言,全部实例都经 Keil C51 和 Proteus 软件仿真,项目实例都经实际制作验证,另有与本书配套的《单片机技术技能操作和学习指导》一书,特别适合自学,使读者在实践中掌握学习的技巧,了解单片机应用的方法。更多详细内容参见精品课程网站 <http://jpkc.njvc.cn/dpj/index.asp>。

本书在编写时力求通俗、易懂,知识以有用、够用为原则,内容以实践为特色。所以,本书适合零起点的初学者,可作为高职高专及中等职业院校的单片机教材,也可作为电子爱好者及各类工程技术人员的参考书。

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

单片机应用技术项目化教程 / 姚晓平主编. —北京:电子工业出版社, 2012.9

高等职业教育教材

ISBN 978-7-121-17472-8

I. ①单… II. ①姚… III. ①单片微型计算机—高等职业教育—教材 IV. ①TP368.1

中国版本图书馆 CIP 数据核字(2012)第 140011 号

策划编辑:施玉新

责任编辑:郝黎明 文字编辑:裴杰

印刷:涿州市京南印刷厂

装订:

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:787×1092 1/16 印张:11.75 字数:300.8 千字

印次:2012 年 9 月第 1 次印刷

定价:24.00 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 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线:(010) 88258888。

前 言

过去对于单片机的教学，均是采用以单片机的结构为主线，先介绍单片机的硬件结构，然后是指令，接着是软件编程，再介绍单片机系统的扩展和各种外围器件的应用，最后再介绍一些实例。按照这种教学结构，现在的学生普遍感到难学。尤其是在不知道一个单片机开发的完整过程的时候，很多人就长叹：单片机太难学了！放弃吧。

基于以上情况，我们在编写本书时，尝试以项目为教学单元，打破原有界限，不管硬件结构、指令、编程的先后顺序，将各部分知识分解成一个个知识点，为了完成一个任务，抽取每个部分的不同知识点，加以组合，完成第一个项目就能清楚单片机的开发过程，完成第二个项目就能自己模仿性地制作，使得学习过程是一个不断增强自信、充满成功喜悦地完成的过程。当所有任务全部完成，知识点就全学完了。即便只学习、完成了部分任务，也可以去做一些程序——事实上，并没有必要学完全部知识才可以去做开发的。

本书分两大部分，一是项目化的教材，二是技能操作和学习指导。项目化的教材部分分5个项目，项目一是霓虹灯的设计与制作，主要介绍最小系统，项目制作流程，虚拟实验室。项目二是电子钟的设计与制作，主要介绍数码管显示技术，键盘识别技术，计数、中断技术的应用。项目三是测量仪表的设计与制作，主要介绍A/D、D/A的转换技术和应用。项目四是通信口应用与控制的设计与制作，主要介绍串口和并口通信技术，通信控制程序设计。项目五是微波炉控制系统的设计与制作，主要介绍液晶显示技术和应用，多任务系统的单片机控制。附录主要介绍C语言编程指令，编程规范和常用的C语言编程实例。技能操作和学习指导部分，主要有学生学习指导，以及单片机技能训练方案。

我们期望本书能达到以下效果：

- (1) 以学生的认知规律为主线，突显快乐学习，玩中增长才干。
- (2) 建立以Keil C51和Proteus软件建立的仿真系统，检验自己的工作成果，最终以PCB实际制作完成产品。通过每个任务的知识点学习、技能训练，构建单片机应用能力。
- (3) 完成第一个项目即可进行单片机的初步应用尝试，不必学完单片机的全部知识体系。随着项目的逐渐进行，知识逐渐完善，能力逐渐提高，所有任务完成时，已具有初步模仿开发能力。
- (4) 希望能培养以工程实践为导向的项目化课程结构。

本书由姚晓平担任主编，张平担任副主编，其中本书项目一由张敏菊编写，项目二由姚晓平编写，项目三由魏小林编写，项目四由张平编写，项目五由陈章余编写，附录由朱来安编写。部分实例经郭星辰仿真和制作验证。我们虽然查阅了大量资料，但是在知识点的整合上，项目、任务的选择和设计上等，还有许多需要商榷的地方。由于水平有限，时间仓促，不足之处再所难免，恳请读者批评指正。

编 者
2012年8月

目 录

项目一 霓虹灯的设计与制作	(1)
任务一 点亮一个 LED 灯	(1)
任务二 闪烁灯	(24)
任务三 流水灯	(37)
任务四 霓虹灯	(41)
项目二 电子钟的设计与制作	(44)
任务一 秒表的设计与制作	(44)
任务二 按键变数的设计与制作	(65)
任务三 电子钟的设计与制作	(79)
项目三 测量仪表的设计与制作	(83)
任务一 数字电压表的设计与制作	(83)
任务二 信号发生器的设计与制作	(95)
项目四 通信口应用与控制的设计与制作	(109)
任务一 单片机双向通信控制系统的设计与制作	(109)
任务二 无线抄表系统的设计与制作	(120)
项目五 微波炉控制系统的设计与制作	(134)
任务一 12864 液晶显示	(134)
任务二 电动机控制	(144)
任务三 微波炉控制系统的实现	(151)
附录 A 虚拟实验室的构建	(164)
附录 B Proteus 常用元器件中英文对照	(173)
附录 C 单片机 C 语言编程规则	(176)

单片机本身不能完成特定的任务，只有与某些器件和设备有机的结合在一起并配以特定的程序才能构成一个真正的单片机应用系统，完成特定的任务。一个单片机应用系统从提出任务到实现任务的全过程分为以下几个步骤：

(1) 任务分析，拟订设计任务书。分析任务情况，拟订可实现的任务实现方案。

(2) 根据任务要求，设计硬件电路。根据设计任务书，先要确定系统的总体设计方案。然后根据总体设计方案的要求，选择单片机的型号，确定硬件所需的元器件，画出电路图。

(3) 设计程序，制作硬件电路。在确定了硬件的设计方案后，即可以进行软件程序设计。一般按照系统要实现的功能，画出程序设计流程图，编写程序代码。

(4) 软/硬件综合调试、仿真。在硬件图和程序代码都完成的情况下，将程序直接下载到实际的硬件进行调试前，可以采用仿真软件或者仿真机先进行调试。这样可以保护单片机芯片，节省使用成本。

(5) 完成设计报告书。根据任务要求，完成设计报告书。设计报告书应尽量详细地阐述系统设计的详细过程。

项目一 霓虹灯的设计与制作

霓虹灯在生活中非常常见。例如，许多城市的商场、饭店、宾馆等地方，用霓虹灯来吸引来来往往人群的眼球。这些不同颜色，不同形状的霓虹灯给我们的生活增添了许多色彩。那么，制作简单的霓虹灯的原理是什么呢？用 LED 如何来制作霓虹灯？

项目任务简介：

LED 的发光形式有很多种，常见的霓虹灯发光现象有长亮、闪烁和流动。将这几种现象组合穿插，就能得到各种各样的霓虹灯。

项目任务设计：

霓虹灯设计制作主要完成以下两项：

- (1) 单片机最小系统的设计制作；
- (2) LED 灯形状阵列的设计。

基于 AT89S51 单片机最小系统的硬件设计，LED 发光驱动的实现。

LED 灯形状阵列的设计。LED 的发光颜色有很多种，发光形状也不尽相同，在具体的设计过程中，可以依据自己的思路来设计，避免千篇一律。

通过以上的任务分析，将项目一划分为四个任务，循序渐进的学习如何用单片机来实现制作霓虹灯。

任务一 点亮一个 LED 灯

一、任务分析

任务一是要用单片机来实现点亮一只发光二极管 LED，学习板上的 P1 口连接了 8 个 LED

发光二极管。这次的任务就是要让其中的一个 LED 发光。这里要解决的问题主要有以下几个：

- (1) 发光二极管如何才能发光？
- (2) 单片机怎么来控制 LED 发光？
- (3) 单片机控制 LED 发光的指令怎么写？

这 3 个问题解决了，任务一就完成了。要让发光二极管发光，根据二极管的单向导电性，只要能让二极管的正、负极得到相应的高低电平就能实现。那这个高低电平的得来，要分析电路原理图和对单片机写指令来实现。

二、任务准备

(一) 单片机介绍

单片机的全称是单片微型计算机 (Single Chip Microcomputer)。它是把组成计算机的主要功能部件：微处理器 (CPU)、数据存储器 (RAM)、程序存储器 (ROM、EPROM、EEPROM 或 Flash ROM)、定时/计数器和各种输入/输出 (I/O) 接口电路等都集成在一块半导体芯片上，构成了一个完整的计算机系统，如图 1-1-1 所示。与通用的计算机不同，单片机的指令功能是按照工业控制的要求设计，因此，它又被称为微控制器 (Microcontroller)。

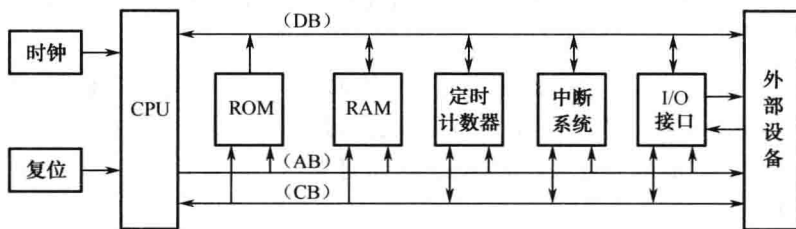


图 1-1-1 单片机内部结构框图

MCS-51 (Micro Controller System) 系列单片机是美国 Intel 公司于 1980 年推出的一种 8 位单片机系列。该系列的基本型产品是 8051、8031 和 8751。这 3 种产品之间的区别只是在片内程序存储器方面。8051 的片内程序存储器 (ROM) 是掩膜型的，即在制造芯片时已将应用程序固化进去；8031 片内没有程序存储器；8751 内部包含有用做程序存储器的 4KB 的 EPROM。MCS-51 系列单片机优异的性能/价格比使得它从面世以来就获得用户的认可。Intel 公司把这种单片机的内核，即 8051 内核，以出售或互换专利的方式授权给一些公司，如 Atmel、Philips、ADI 公司等。这些公司的这类产品也被称为 8051 兼容芯片，这些 8051 兼容芯片在原来的基础上增加了许多特性。例如，STC 公司生产的 STC89C52RC 单片机，它是一款性价比非常高的单片机，普通用户可完全将其当做一般的 51 单片机使用，高级用户可使用其扩展功能。STC 公司的单片机内部资源比起 ATMEL 公司的单片机来要丰富得多，它内部有 8KB 的基于 Flash 技术的程序存储器，512B 的 RAM 数据存储器，8 个中断源，3 个定时器，片内自带 EEPROM，片机自带看门狗、双数据指针等。目前，STC 公司的单片机在国内市场上的占有率与日俱增。STC 系列单片机具有多种封装形式，包括 PDIP40、PLCC44 和 TQFP44。最适合学校实验室使用的是 PDIP40 封装形式。PDIP40 封装形式的单片机芯片可以很方便地使用面包板来组成应用电路。

MCS-51 系列单片机一般采用 HMOS（高密度 NMOS）和 CHMOS（高密度 CMOS）两种工艺制造，前者如 8051，后者如 80C51，89S52 的 S 指的是支持串行下载。STC 公司的 C 系列也支持串行下载。

综上所述，单片机由两大部分组成。

- (1) 硬件系统：组成单片机系统的具体实体。
- (2) 软件系统：对单片机硬件系统进行管理使用。

1. 硬件

1) 微处理器（CPU）

CPU 由以下几个部分组成。

- (1) 寄存器阵列：通用寄存器，专用寄存器。
- (2) 运算器：累加器，暂存寄存器，标志寄存器，算术逻辑单元。
- (3) 控制器：程序计数器 PC，指令寄存器，指令译码器，定时和控制逻辑电路。

2) 总线

总线是用于传送信息的公共途径，分为以下几种：

- (1) 数据总线（DB）。
- (2) 地址总线（AB）。
- (3) 控制总线（CB）。

3) 存储器

存储器是存放程序和数据的，其分成以下几个部分。

(1) RAM：随机存储器。

特点：读写速度快，可随机写入或读出，读写方便；电源断电后，存储信息丢失。

作用：存放各种数据。

(2) ROM：只读存储器。

特点：信息写入后，能长期保存，不会因断电而丢失。

作用：存放固定程序和数据。

ROM 中常见的有 EPROM、EEPROM、Flash ROM 等

4) 输入/输出设备及其接口电路

- (1) 输入设备：典型的输入设备有键盘、鼠标等。
- (2) 输出设备：常见的输出设备有打印机等。
- (3) I/O 接口电路。

输入/输出设备一般不能与 CPU 直接相连，而是通过某种电路完成寻址、数据缓冲、输入/输出控制、功率驱动、A/D、D/A 等功能，这种电路称为 I/O 接口电路。

2. 软件系统

单片机程序设计语言可分为三类：

- (1) 机器语言。
- (2) 汇编语言。
- (3) 高级语言。

本教材使用 C 语言来实现单片机系统的开发。

(二) AT89S51 单片机

1. AT89S51 简介

AT89S51 是一个低功耗、高性能的 CMOS 8 位单片机，片内含 4KB ISP (In-System Programmable) 的可反复擦写 1000 次的 Flash 只读程序存储器，器件采用 ATMEL 公司的高密度、非易失性存储技术制造，兼容标准 MCS-51 指令系统及 80C51 引脚结构，芯片内集成了通用 8 位中央处理器和 ISP Flash 存储单元，功能强大的微型计算机的 AT89S51 可为许多嵌入式控制应用系统提供高性价比的解决方案。

AT89S51 具有如下特点：40 个引脚，如图 1-1-2 所示，4KB 的 Flash 片内程序存储器，128B 的随机存取数据存储器 (RAM)，32 个外部双向输入/输出 (I/O) 口，5 个中断优先级，2 层中断嵌套中断，2 个 16 位可编程定时计数器，2 个全双工串行通信口，看门狗 (WDT) 电路，片内时钟振荡器。

此外，AT89S51 设计和配置了振荡频率，并可通过软件设置省电模式。在空闲模式下，CPU 暂停工作，而 RAM 定时计数器、串行口、外部中断系统可继续工作，掉电模式冻结振荡器而保存 RAM 的数据，停止芯片其他功能，直至外部中断激活或硬件复位。同时，该芯片还具有 PDIP、TQFP 和 PLCC 三种封装形式，以适应不同产品的需求。

AT89S51 的主要功能特性：

- (1) 兼容 MCS-51 指令系统。
- (2) 4KB 可反复擦写 (>1000 次) ISP Flash ROM。

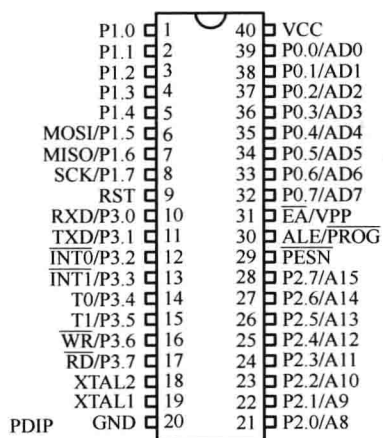


图 1-1-2 AT89S51 引脚图

- (3) 32 个双向 I/O 口。
- (4) 4.5~5.5V 工作电压。
- (5) 2 个 16 位可编程定时/计数器。
- (6) 时钟频率为 0~33MHz。
- (7) 全双工 UART 串行中断口线。
- (8) 128×8bit 内部 RAM。
- (9) 2 个外部中断源。
- (10) 功耗空闲和省电模式。
- (11) 中断唤醒省电模式。
- (12) 3 级加密位。
- (13) 看门狗 (WDT) 电路。
- (14) 软件设置空闲和省电功能。
- (15) 灵活的 ISP 字节和分页编程。
- (16) 双数据寄存器指针。

2. 引脚说明

- (1) VCC: 供电电压。
- (2) GND: 接地。
- (3) P0 口: P0 口为一个 8 位漏级开路双向 I/O 口，每脚可吸收 8TTL 门电流。当 P0 口的

引脚第一次写“1”时，被定义为高阻输入。P0 能够用于外部程序数据存储器，它可以被定义为数据/地址的第 8 位。在 Flash 编程时，P0 口作为原码输入口，当 Flash 进行校验时，P0 输出原码，此时 P0 外部必须被拉高。

(4) P1 口是一个内部提供上拉电阻的 8 位双向 I/O 口，P1 口缓冲器能接收输出 4TTL 门电流。P1 口引脚写入 1 后，被内部上拉为高，可用做输入；P1 口被外部下拉为低电平时，将输出电流，这是由于内部上拉的缘故。在 Flash 编程和校验时，P1 口作为第 8 位地址接收。

(5) P2 口：P2 口为一个内部上拉电阻的 8 位双向 I/O 口，P2 口缓冲器可接收，输出 4 个 TTL 门电流，当 P2 口被写“1”时，其引脚被内部上拉电阻拉高，且作为输入。并因此作为输入时，P2 口的引脚被外部拉低，将输出电流，这是由于内部上拉的缘故。当 P2 口用于外部程序存储器或 16 位地址外部数据存储器进行存取时，P2 口输出地址的高 8 位。在给出地址“1”时，它利用内部上拉优势，当对外部 8 位地址数据存储器进行读写时，P2 口输出其特殊功能寄存器的内容。P2 口在 Flash 编程和校验时接收高 8 位地址信号和控制信号。

(6) P3 口：P3 口引脚是 8 个带内部上拉电阻的双向 I/O 口，可接收输出 4 个 TTL 门电流。当 P3 口写入“1”后，它们被内部上拉为高电平，并用做输入。作为输入，由于外部下拉为低电平，P3 口将输出电流 (ILL)，这是由于上拉的缘故。

P3 口也可作为 AT89S51 的一些特殊功能口，见表 1-1-1。

表 1-1-1 P3 口的备选功能表

引 脚	备 选 功 能
P3.0	RXD (串行输入口)
P3.1	TXD (串行输出口)
P3.2	INT0 (外部中断 0)
P3.3	INT1 (外部中断 1)
P3.4	T0 (记时器 0 外部输入)
P3.5	T1 (记时器 1 外部输入)
P3.6	WR (外部数据存储器写选通)
P3.7	RD (外部数据存储器读选通)

P3 口同时为闪烁编程和编程校验接收一些控制信号。

(7) RST: 复位输入。当振荡器复位器件时，要保持 RST 脚两个机器周期的高电平时间。

(8) ALE/PROG: 当访问外部存储器时，地址锁存允许的输电平用于锁存地址的低字节。当 CPU 访问片外存储器时，ALE 为访问片外存储器的地址锁存允许信号输出，在不使用外部存储器时此引脚悬空；PROG 是对内部 EPROM 编程时的脉冲输入。

(9) PSEN: 外部程序存储器的选通信号。当访问片外程序存储器时，通过产生负脉冲作为片外程序存储器选通信号，如果不使用片外程序存储器，此引脚可悬空。

(10) \overline{EA} /VPP: 内/外部 ROM 选择端。

单片机应用电路中引脚 \overline{EA} (引脚 31) 可以总是接高电平。VPP 是编程电源输入。

(11) XTAL1: 反向振荡放大器的输入及内部时钟工作电路的输入。

(12) XTAL2: 来自反向振荡器的输出。

3. 振荡器特性

XTAL1 和 XTAL2 分别为反向放大器的输入和输出。该反向放大器可以配置为片内振荡器。石晶振荡和陶瓷振荡均可采用。如采用外部时钟源驱动器件，XTAL2 应不接。有时输入至内部时钟信号要通过一个二分频触发器，因此，对外部时钟信号的脉宽无任何要求，但必须保证脉冲的高低电平要求的宽度。

(三) LED 灯基本知识

LED (Light Emitting Diode), 即发光二极管, 是一种半导体固体发光器件, 它利用固体半导体芯片作为发光材料。当两端加上正向电压时, 半导体中的少数载流子和多数载流子发生复合, 放出过剩的能量而引起光子发射, 直接发出红、橙、黄、绿、青、蓝、紫、白色的光。常用的 LED 灯导通压降为 1.7V 左右, 导通电流在 3~10mA 发光比较合适。如果提供的电源电压为 5V, 选用 1kΩ 的排阻, 得 $\frac{5V-1.7V}{1k\Omega}=3.3mA$, 满足要求。

(四) 计算机中数的表示方法

1. 常用数制

在日常生活中, 人们最熟悉的是十进制数, 但在计算机中, 采用二进制数“0”和“1”可以很方便地表示机内的数据与信息。

1) 十进制数

十进制数有两个主要特点。

(1) 有十个不同的数字符号: 0、1、2、…、9。

(2) 低位向高位进、借位的规律是按“逢十进一”、“借一当十”的计数原则进行计数。

例如, $1234.45=1\times 10^3+2\times 10^2+3\times 10^1+4\times 10^0+4\times 10^{-1}+5\times 10^{-2}$, 式中的 10 称为十进制数的基数, 10³、10²、10¹、10⁰、10⁻¹ 称为各数位的权。十进制数用 D 结尾表示。

2) 二进制数

在二进制中, 只有两个不同数码: 0 和 1, 进位规律是按“逢二进一”、“借一当二”的计数原则进行计数。二进制数用 B 结尾表示。

例如, 二进制数 11011011.01 可表示为 $(11011011.01)_2=1\times 2^7+1\times 2^6+0\times 2^5+1\times 2^4+1\times 2^3+0\times 2^2+1\times 2^1+1\times 2^0+0\times 2^{-1}+1\times 2^{-2}$ 。

3) 八进制数

在八进制中有 0、1、2…、7 八个不同数码, 采用“逢八进一”“借一当八”的计数原则进行计数。八进制数用 O 结尾表示。

例如, 八进制数 (503.04) O 可表示为 $(503.04)_O=5\times 8^2+0\times 8^1+3\times 8^0+0\times 8^{-1}+4\times 8^{-2}$ 。

4) 十六进制数

在十六进制中有 0、1、2…、9、A、B、C、D、E、F 共十六个不同的数码, 采用“逢十六进一”“借一当十六”的计数原则进行计数。十六进制数用 H 结尾表示。

例如, 十六进制数 (4E9.27) H 可表示为 $(4E9.27)_H=4\times 16^2+14\times 16^1+9\times 16^0+2\times 16^{-1}+7\times 16^{-2}$ 。

2. 不同进制数之间的相互转换

表 1-1-2 列出了二、八、十、十六进制数之间的对应关系, 熟记这些对应关系对后续内容的学习会有较大的帮助。

表 1-1-2 各种进位制的对应关系

十进制	二进制	八进制	十六进制	十进制	二进制	八进制	十六进制
0	0	0	0	9	1001	11	9
1	1	1	1	10	1010	12	A
2	10	2	2	11	1011	13	B
3	11	3	3	12	1100	14	C
4	100	4	4	13	1101	15	D
5	101	5	5	14	1110	16	E
6	110	6	6	15	1111	17	F
7	111	7	7	16	10000	20	10
8	1000	10	8	17	10001	21	11

1) 二、八、十六进制数转换为十进制数

根据各进制的定义表示方式,按权展开相加,即可转换为十进制数。

【例 1-1-1】 将 (10101)_B、(72)_O、(49)_H 转换为十进制数。

$$(10101)_B = 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 37$$

$$(72)_O = 7 \times 8^1 + 2 \times 8^0 = 58$$

$$(49)_H = 4 \times 16^1 + 9 \times 16^0 = 73$$

2) 十进制数转换为二进制数

十进制数转换二进制数,需要将整数部分和小数部分分开,采用不同的方法进行转换,然后用小数点将这两部分连接起来。

(1) 整数部分:除 2 取余法。

具体方法是,将要转换的十进制数除以 2,取余数;再用商除以 2,再取余数,直到商等于 0 为止,将每次得到的余数按倒序的方法排列起来作为结果。

【例 1-1-2】 将十进制数 100 转换成二进制数。

2	100	余数
	2 50	0 (最低位)
	2 25	0
	2 12	1
	2 6	0
	2 3	0
	2 1	1
	0	1 (最高位)

答案: 100D=1100100B

(2) 小数部分:乘 2 取整法。

具体方法是,将十进制小数不断地乘以 2,直到积的小数部分为零(或直到所要求的位数)为止,每次乘得的整数依次排列即为相应进制的数码。最初得到的为最高有效数字,最后得到的为最低有效数字。

【例 1-1-3】 将 0.625D 转换成二进制数。

乘 2 取整:	整数部分
0.625	
× 2	
1.250	1
0.25	
× 2	
0.50	0
× 2	
1.0	1

答案: $0.625D=0.101B$

整合: $100.625D=1100100.101B$

(3) 二进制与八进制之间的相互转换。

由于 $2^3=8$, 故可采用“合三为一”的原则, 即从小数点开始向左、右两边各以 3 位为一组进行二一八转换: 若不足 3 位的以 0 补足, 便可以将二进制数转换为八进制数。反之, 每位八进制数用三位二进制数表示, 就可将八进制数转换为二进制数。

【例 1-1-4】 将 $(10100101.01011101)_2$ 转换为八进制数。

即 $(10100101.01011101)_2 = (245.272)_8$

【例 1-1-5】 将 $(756.34)_8$ 转换为二进制数。

7 5 6 . 3 4

111 101 110 . 011 100

即 $(756.34)_8 = (111101110.0111)_2$

(4) 二进制与十六进制之间的相互转换。

由于 $2^4=16$, 故可采用“合四为一”的原则, 即从小数点开始向左、右两边各以 4 位为一组进行二一十六转换, 若不足 4 位的以 0 补足, 便可以将二进制数转换为十六进制数。反之, 每位十六进制数用四位二进制数表示, 就可将十六进制数转换为二进制数。

【例 1-1-6】 将 $(1111111000111.100101011)_2$ 转换为十六进制数。

0001 1111 1100 0111 . 1001 0101 1000

1 F C 7 . 9 5 8

即 $(1111111000111.100101011)_2 = (1FC7.958)_{16}$

【例 1-1-7】 将 $(79BD.6C)_{16}$ 转换为二进制数。

7 9 B D . 6 C

0111 1001 1011 1101 . 0110 1100

即 $(79BD.6C)_{16} = (111100110111101.011011)_2$

(五) 常用的信息编码

1. 二一十进制 BCD 码 (Binary-Coded Decimal)

二一十进制 BCD 码是指每位十进制数用 4 位二进制数编码表示。由于 4 位二进制数可以表示 16 种状态, 可丢弃最后 6 种状态, 而选用 0000~1001 来表示 0~9 十个数符。这种编码又称 8421 码, 见表 1-1-3。

表 1-1-3 十进制数与 BCD 码的对应关系

十进制数	BCD 码	十进制数	BCD 码
0	0000	10	00010000
1	0001	11	00010001
2	0010	12	00010010
3	0011	13	00010011
4	0100	14	00010100
5	0101	15	00010101
6	0110	16	00010110
7	0111	17	00010111
8	1000	18	00011000
9	1001	19	00011001

【例 1-1-8】 将 69.25 转换成 BCD 码。

6 9 . 2 5

0110 1001 . 0010 0101

结果为 69.25=(01101001.00100101)BCD

【例 1-1-9】 将 BCD 码 100101111000.01010110 转换成十进制数。

1001 0111 1000 . 0101 0110

9 7 8 . 5 6

结果为 (100101111000.01010110) BCD=978.56

2. 字符编码 (ASCII 码)

计算机使用最多、最普遍的是 ASCII (American Standard Code For Information Interchange) 字符编码, 即美国信息交换标准代码, 见表 1-1-4。

ASCII 码的每个字符用 7 位二进制数表示, 其排列次序为 d6d5d4d3d2d1d0, d6 为高位, d0 为低位。而一个字符在计算机内实际是用 8 位表示。正常情况下, 最高一位 d7 为“0”。7 位二进制数共有 128 种编码组合, 可表示 128 个字符, 其中, 数字 10 个、大小写英文字母 52 个、其他字符 32 个和控制字符 34 个。

表 1-1-4 七位 ASCII 代码表

d3 d2 d1 d0 位	0 d6 d5 d4 位							
	000	001	010	011	100	101	110	111
0000	NUL	DEL	SP	0	@	P	'	p
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	"	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s
0100	EOT	DC4	\$	4	D	T	d	t
0101	ENQ	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	'	7	G	W	g	w

续表

d3 d2 d1d0 位	0 d6 d5d4 位							
	000	001	010	011	100	101	110	111
1000	BS	CAN	(8	H	X	h	x
1001	HT	EM)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[k	{
1100	FF	FS	,	<	L	\	l	
1101	CR	GS	-	=	M]	m	}
1110	SO	RS	.	>	N	↑	n	~
1111	SI	HS	/	?	O	←	o	DEL

要确定某个字符的 ASCII 码，在表中可先查到它的位置，然后确定它所在位置的相应列和行，最后根据列确定高位码（d6d5d4），根据行确定低位码（d3d2d1d0），把高位码与低位码合在一起就是该字符的 ASCII 码。例如，数字 9 的 ASCII 码为 00111001B，即十六进制为 39H；字符 A 的 ASCII 码为 01000001，即十六进制为 41H 等。

数字 0~9 的 ASCII 码为 30H~39H。

大写英文字母 A~Z 的 ASCII 码为 41H~5AH。

小写英文字母 a~z 的 ASCII 码为 61H~7AH。

对于 ASCII 码表中的 0、A、a 的 ASCII 码 30H、41H、61H 应尽量记住，其余的数字和字母的 ASCII 码可按数字和字母的顺序以十六进制的规律写出。

ASCII 码主要用于微机与外设的通信。当微机接收键盘信息，微机输出到打印机、显示器等信息都是以 ASCII 码形式进行数据传输。

3. 带符号数的表示

在计算机中，带符号数可以用不同的方法表示，常用的有原码、反码和补码。

1) 原码

【例 1-1-10】 当机器字长 $n=8$ 时：

$[+1]_{\text{原}}=0\ 0000001$ ， $[-1]_{\text{原}}=1\ 0000001$

$[+127]_{\text{原}}=0\ 1111111$ ， $[-127]_{\text{原}}=1\ 1111111$

由此可以看出，在原码表示法中：

最高位为符号位，正数为 0，负数为 1，其余 $n-1$ 位表示数的绝对值。

在原码表示中，零有两种表示形式，即 $[+0]=00000000$ ， $[-0]=10000000$ 。

2) 反码

【例 1-1-11】 当机器字长 $n=8$ 时，

$[+1]_{\text{反}}=00000001$ ， $[-1]_{\text{反}}=11111110$

$[+127]_{\text{反}}=01111111$ ， $[-127]_{\text{反}}=10000000$

由此看出，在反码表示中：

正数的反码与原码相同，负数的反码只需将其对应的正数按位求反即可得到。机器数最高位为符号位，0 代表正号，1 代表负号。

在反码表示方式中，零有两种表示方法： $[+0]_{\text{反}}=00000000$ ， $[-0]_{\text{反}}=11111111$ 。

3) 补码

【例 1-1-12】 当机器字长 $n=8$ 时,

$$[+1]_{\text{补}}=00000001, \quad [-1]_{\text{补}}=11111111$$

$$[+127]_{\text{补}}=01111111, \quad [-127]_{\text{补}}=10000001$$

由此看出, 在补码表示中:

正数的补码与原码、反码相同, 负数的补码等于它的反码加 1。

机器数的最高位是符号位, 0 代表正号, 1 代表负号。

在补码表示中, 0 有唯一的编码: $[+0]_{\text{补}}=[-0]_{\text{补}}=00000000$ 。

补码的运算方便, 二进制的减法可用补码的加法实现, 使用较广泛。在计算机中的运算都是用补码进行运算。

【例 1-1-13】 假设计算机字长为 8 位, 试写出 122 的原码、反码和补码。

$$[122]_{\text{原}}=[122]_{\text{反}}=[122]_{\text{补}}=01111010\text{B}$$

【例 1-1-14】 假设计算机字长为 8 位, 试写出 -45 的原码、反码和补码。

$$[-45]_{\text{原}}=10101101\text{B}$$

$$[-45]_{\text{反}}=11010010\text{B}$$

$$[-45]_{\text{补}}=11010011\text{B}$$

对于用补码表示的负数, 首先认定它是负数, 然后用求它的补码的方法可得到它的绝对值, 即可求得该负数的值。例如, 补码数 $(11110011)\text{B}$ 是一个负数, 求该数的补码为 $(00001101)\text{B}$, 该数相应的十进制数为 13, 故求出 $(11110011)\text{B}$ 为 $(-13)\text{D}$ 。

【例 1-1-15】 试写出原码 11011001 的真值。

$$(\text{原码})_{\text{补}}=(\text{原码})_{\text{反}}+1=10100111\text{B}=-39$$

(六) 逻辑运算

1) “与”运算

“与”运算的运算规则是“有 0 为 0, 全 1 为 1”。

$$0&0=0 \quad 0&1=0$$

$$1&0=0 \quad 1&1=1$$

【例 1-1-16】 二进制数 01011101B 和 11010101B 相与。

$$01011101&11010101=01010101\text{B}$$

2) “或”运算

“或”运算的运算规则是“有 1 为 1, 全 0 为 0”。

$$0|0=0 \quad 0|1=1$$

$$1|0=1 \quad 1|1=1$$

【例 1-1-17】 二进制数 10101101 和 01010000 相或。

$$10101101\text{B}|01010000\text{B}=11111101\text{B}$$

3) “异或”运算

“异或”运算的运算规则是“相同为 0, 相异为 1”。

$$0\oplus 0=0 \quad 0\oplus 1=1$$

$$1\oplus 0=1 \quad 1\oplus 1=0$$

【例 1-1-18】 二进制数 10101101 和 01101110 相异或。

10101101B⊕01101110=11000011

(七) Keil 软件使用

Keil μ Vision2 是众多单片机应用开发软件中优秀的软件之一，它支持众多不同公司的 MCS-51 架构的芯片，它集编辑，编译，仿真等于一体，同时还支持 PLM，汇编和 C 语言的程序设计，它的界面和常用的微软 VC++ 的界面相似，界面友好，易学易用，在调试程序，软件仿真方面也有很强大的功能。

Keil 单片机模拟调试软件安装完成以后，计算机桌面上将产生一个标注有“Keil μ Vision2”的图标，双击这个图标就可以进入 Keil 单片机模拟调试软件的集成开发环境，出现如图 1-1-3 所示的屏幕，进入图 1-1-4 所示的编辑界面。



图 1-1-3 Keil 图标

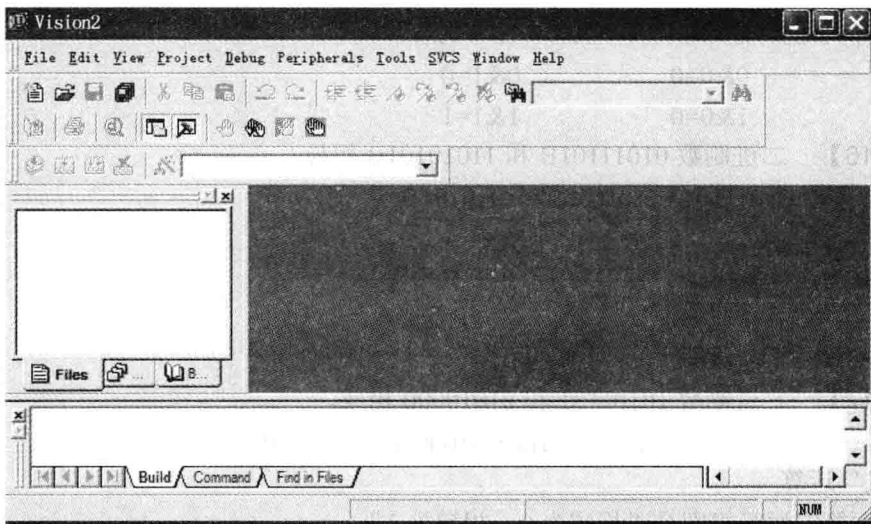


图 1-1-4 Keil 编辑界面

接着按下面的步骤建立第一个项目。