

 职业院校教学用书（电子类专业）

# 单片机C语言案例教程

白焯贵 余明飞 罗永 编著

SCM C language course of the case



电子工业出版社  
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

职业院校教学用书（电子类专业）

# 单片机 C 语言案例教程

白炽贵 余明飞 罗 永 编著

电子工业出版社

Publishing House of Electronics Industry

北京 • BEIJING

## 内 容 简 介

本书是中职学校电子类专业单片机课程教材。教材在编写上贴近中职学生的学习特点，以“案例+项目”的形式展开，适合初中以上读者自学单片机技术之用。

本书介绍以不使用汇编语言、不使用编程器和不使用仿真器的简约型开发手段，用 Keil C51 对 51 单片机进行系统编程的实用方法，从驱动一位数码管显示“8.”的最简单的程序起步，用十多个案例作为铺垫，翔实地讲解了含时控功能的电子钟温度计和 16×16 点阵 LED 汉字显示屏的功能创意和程序设计，引领学生对这两个具有实际使用价值和市场开发前景的实用项目进行探究式学习和生产式制作，让学生在理论与实践相结合的学习过程中，切实掌握单片机关于两个外中断、两个内中断、单总线器件、双总线器件、三总线器件、串并转换、A/D 转换与 D/A 转换、IAP 编程和用串行通信把汉字字模从 PC 写入单片机片外 Flash 存储器等核心技术。

为了方便教师的“教”和学生的“学”，本书还开发了电子教学包。电子教学包包含本书的电子教案、教学指南、案例源程序和习题解答，并有一篇讲解本书所用单片机学习板印制电路图设计的电子教程。需要时可登录华信教育资源网下载。

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 C 语言案例教程/白炽贵，余明飞，罗永编著. —北京：电子工业出版社，2011.1  
职业院校教学用书. 电子类专业

ISBN 978-7-121-12519-5

I. ①单… II. ①白… ②余… ③罗… III. ①单片微型计算机—C 语言—程序设计—高等学校：技术学校—教材 IV. ①TP368.1②TP312

中国版本图书馆 CIP 数据核字（2010）第 240217 号

策划编辑：杨宏利 yhl@phei.com.cn

责任编辑：毕军志

印 刷：北京市顺义兴华印刷厂

装 订：三河市双峰印刷装订有限公司

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：15.75 字数：400 千字

印 次：2011 年 1 月第 1 次印刷

印 数：4 000 册 定价：27.00 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。

# 前 言



中职学校机电类专业学生学习单片机技术,有利于提高中职学生的职业技能和就业创业能力,有利于中职学生的继续学习和终身学习。结合中等职业学校的教学实际和中职学生的学习特点,我们编写了这本《单片机C语言案例教程》。

学习从兴趣起步!兴趣是最好的老师!激发和发展学生的单片机学习兴趣,是本书的编写宗旨;强化单片机学习的操作和实践,是本书的编写风格。本书以“案例+项目”的形式展开中职学生单片机的学习开发过程。用案例为项目做铺垫,用项目整合案例,用项目向学生展现其学习价值和使用价值。

全书共13章。

第1章简要介绍了MCS-51单片机的内部结构和本书单片机学习板的电原理图及其印制电路板图,详细介绍了用Keil C51编译器和STC-ISP下载软件完成单片机编程的整个过程步骤,并用6个案例展示了C语言对单片机编程的易用性。

第2章集中介绍了单片机C语言编程的基本知识和简明方法,以作为全书各案例源程序分析时的理论依据,可以不按其小节顺序讲授。

第3章以数码管的花样显示为要求,用6个案例说明了单片机C语言编程中“谋篇布局”的思谋策略。

第4章简要介绍了中断的概念和单片机中的中断资源,用5个案例说明了如何设置中断允许寄存器IE和中断优先级寄存器IP,展示了如何使用两个外中断、两个内中断的具体方法,演示了中断嵌套和中断串接的特点。

第5章介绍了对三总线器件的编程要点,用一个案例展示了对三总线器件DS1302编程的软、硬件设计。

第6章介绍了对二总线器件的编程要点,用一个案例展示了对二总线器件AT24C02编程的软、硬件设计。

第7章介绍了对单总线器件的编程要点,用一个案例展示了对单总线器件DS18B20编程的软、硬件设计。

第8章介绍了单片机开发中的编程整合技术,即用“有时控功能和温度显示的电子钟”这一项目,展示了将三总线器件DS1302案例、二总线器件AT24C02案例、单总线器件DS18B20案例整合成一个产品项目的单片机学习开发技术。在这个项目中,还根据项目所需,综合使用了单片机中的四个中断资源。

第9章介绍了16×16点阵LED汉字显示屏的软、硬件设计。用16×16点阵LED汉字显示

屏这一项目，展示了用 C 语言开发小型 LED 汉字显示屏的实用技术方案。

第 10 章介绍了基于宏晶公司技术资料的 STC 单片机片内 E<sup>2</sup>PROM 编程技术。用一个案例演示了 STC 单片机片内 E<sup>2</sup>PROM 的编程效果，用另一项目提升了“有时控功能和温度显示的电子钟”的开发价值。

第 11 章介绍了单片机与 PC 间的串行通信技术。用一个案例展示了单片机与 PC 间串行通信的实用方法及其效果。

第 12 章介绍了单片机扩充片外数据存储器的接口电路，用两个案例示范了 HM62256 和 W29C020 的硬件接口和读写技术，用两个项目展示了使用片外数据存储器的 LED 汉字显示屏的技术升级方案。

第 13 章介绍了模数转换和数模转换知识，用一个案例展示了 ADC0809 的接口电路和模数转换效果，用另一个案例展示了 DAC0832 的接口电路和数模转换效果。

除第 2 章外，本书各章均配有电子教案。每章的电子教案均按节的顺序组织。对书中的每一案例的下载操作和运行效果，电子教案中都制作有一段视频演示利于提高教学效果。

本书还有一篇讲解本书所用单片机学习板印刷电路图设计的电子教程，读者能在该电子教程的辅导下完成本书学习板电路的印制电路设计，设计出来的 PCB 与本书正文中的图 1-4 完全一致。这就能指导读者，全程走通，从单片机的程序设计，到单片机的项目开发，再到单片机的电路制版的整个单片机产品研制过程。

本书全书由重庆市綦江职业教育中心白炽贵统稿。第 1~4 章由罗永编写，第 5~8 章由余明飞编写，第 9~13 章由白炽贵编写。由于编者水平有限，书中难免有错谬之处，敬请广大读者指正。


本书配有电子教案、教学指南、案例源程序、习题解答、PCB 制作电子教程和用于实际制版加工的 PCB 文件，能更多地满足读者在各方面的实用需求。请有此需要的教师登录华信教育网 ([www.hxedu.com.cn](http://www.hxedu.com.cn)) 免费注册后进行下载，有问题时请在网站留言板留言或与电子工业出版社联系 (E-mail: [hxedu@phei.com.cn](mailto:hxedu@phei.com.cn))。有关本书的意见和问题，可以发送电子邮件至 [qjgbai@163.com](mailto:qjgbai@163.com)。

编著者

2010 年 8 月



# 目 录



<b>第 1 章 单片机基础</b> .....	1
1.1 单片机简介 .....	1
1.2 用单片机芯片组成单片机应用系统 .....	2
1.3 对单片机编程以实现单片机系统的功能 .....	5
【案例 1】 在 4 位数码管的最左位上显示“8.” .....	5
1.4 把程序写入单片机芯片的操作平台的操作步骤 .....	6
1.5 从数码管显示学单片机编程 .....	18
【案例 2】 数码管上的“8.8.”显示 .....	18
【案例 3】 被动形式的“8051”显示 .....	20
【案例 4】 主动形式的“8051”显示 .....	22
【案例 5】 模块化的“8051”显示 .....	24
【案例 6】 实用型的“8051”显示 .....	26
1.6 源程序的阅读要点 .....	28
1.7 REG51.h 头文件简介 .....	31
小结 1 .....	33
习题 1 .....	34
<b>第 2 章 C 语言编程基础</b> .....	35
2.1 C 语言程序的组成结构和运行机制 .....	35
2.2 C 语言的字符集、标识符、关键字和语句 .....	36
2.3 C 语言中的数据类型 .....	37
2.4 常量和变量 .....	38
2.5 整型数据 .....	38
2.6 实型数据 .....	39
2.7 字符型数据 .....	39
2.8 C51 扩充数据类型 .....	40
2.9 运算符与表达式 .....	40
2.9.1 算术运算符与算术表达式 .....	40
2.9.2 赋值运算符与赋值表达式 .....	41
2.9.3 关系运算符和关系表达式 .....	42
2.9.4 逻辑运算符与逻辑表达式 .....	43
2.9.5 位运算符与位运算表达式 .....	43

2.9.6	逗号运算符与逗号表达式	43
2.9.7	运算符的优先级和结合性	44
2.10	程序的三种基本结构	45
2.11	选择结构	46
2.12	循环结构	48
2.13	数组	51
2.13.1	一维数组的定义和引用	51
2.13.2	二维数组的定义和引用	52
2.14	变量和数组的存储器类型	54
2.15	函数的定义和调用	54
2.15.1	系统库函数	54
2.15.2	用户自定义函数	55
2.15.3	函数的定义	55
2.15.4	函数调用的方式和规则	56
2.15.5	全局变量和局部变量	57
小结 2		57
习题 2		58
<b>第 3 章</b>	<b>用数码管模拟的循环灯</b>	<b>59</b>
【案例 7】	“并驾齐驱”的跑马灯	59
【案例 8】	“迈步边关”的巡逻灯	64
【案例 9】	“大江东去”的流水灯	69
【案例 10】	“孔雀开屏式”循环灯	73
【案例 11】	奔腾不息的“8051”	77
【案例 12】	9999 倒数数的数码管显示	82
小结 3		85
习题 3		86
<b>第 4 章</b>	<b>中断资源的应用编程</b>	<b>87</b>
4.1	中断的概念	87
4.2	中断允许寄存器 IE 和中断优先级寄存器 IP	87
4.2.1	中断允许寄存器 IE	87
4.2.2	中断优先级寄存器 IP	88
4.3	定时器/计数器简介	89
4.3.1	定时器/计数器体系结构示意图	89
4.3.2	特殊功能寄存器 TCON	89
4.3.3	特殊功能寄存器 TMOD	90
【案例 13】	用外中断 INT1 操作插曲显示	91
【案例 14】	中断插曲的中断插曲显示	95
【案例 15】	基于定时器中断的数字钟	99

【案例 16】 基于计数器中断的中断插曲	104
【案例 17】 定时器 T1 与计数器 T0 的串接使用	107
小结 4	111
习题 4	111
<b>第 5 章 日历时钟 DS1302 的应用编程</b>	113
5.1 日历时钟 DS1302 简介	113
5.2 对 DS1302 的写/读程序设计	114
5.2.1 最底层的按位写/读函数	114
5.2.2 对 DS1302 的字节写/读函数	116
【案例 18】 基于 DS1302 的实用电子钟	117
小结 5	125
习题 5	125
<b>第 6 章 AT24C02 的应用编程</b>	126
6.1 I <sup>2</sup> C 总线简介	126
6.1.1 I <sup>2</sup> C 总线接口	126
6.1.2 I <sup>2</sup> C 器件的地址	126
6.1.3 对 I <sup>2</sup> C 器件进行数据传送的操作时序	127
6.2 AT24C02 简介	127
6.2.1 AT24C02 引脚图	127
6.2.2 读/写 AT24C02 的专用函数	128
6.2.3 9 个专用函数间的调用路线图	131
【案例 19】 AT24C02 应用编程	132
小结 6	138
习题 6	138
<b>第 7 章 DS18B20 的编程应用</b>	139
7.1 DS18B20 使用简介	139
7.1.1 DS18B20 的外形及与单片机的接口	139
7.1.2 DS18B20 的内部结构和操作指令	139
7.2 对 DS18B20 的操作时序和读/写专用函数	140
7.2.1 对 DS18B20 的操作时序	140
7.2.2 DS18B20 的读/写专用函数	141
【案例 20】 DS18B20 的应用编程	143
小结 7	149
习题 7	149
<b>第 8 章 有时控功能和温度显示的电子钟项目设计</b>	150
8.1 项目简介	150
8.2 项目源程序设计要点分析	164



小结 8	168
习题 8	169
<b>第 9 章 16×16 点阵 LED 汉字显示屏项目设计</b>	<b>170</b>
9.1 项目简介	170
9.2 16×16 点阵 LED 汉字显示屏电原理图	170
9.3 74HC138 功能简介	171
9.4 74HC595 功能简介	172
9.5 单片机与 LED 汉字显示屏的接口设计	174
9.6 汉字字模在二维数组中的存储格式	174
9.7 16×64 点阵 LED 汉字显示屏项目源程序及分析	175
9.8 16×16 汉字字模文件的建立	180
小结 9	183
习题 9	183
<b>第 10 章 使用片内 E<sup>2</sup>PROM 存储器</b>	<b>184</b>
10.1 IAP 和 ISP 简介	184
10.2 IAP 技术的接口函数	184
【案例 21】 片内 E <sup>2</sup> PROM 编程案例	187
10.3 用片内 E <sup>2</sup> PROM 取代电子钟项目中的 AT24C02	192
小结 10	194
习题 10	194
<b>第 11 章 单片机与 PC 的串行通信</b>	<b>195</b>
【案例 22】 单片机与 PC 的串行通信案例	198
小结 11	205
习题 11	206
<b>第 12 章 单片机片外数据存储器的扩充</b>	<b>207</b>
12.1 HM62256 存储器的扩充及使用案例	207
12.1.1 HM62256 存储器的扩充	207
12.1.2 HM62256 存储器的使用	209
12.2 用 HM62256 存储汉字字模项目设计	212
12.3 W29C020 存储器的扩充和使用案例	220
12.3.1 W29C020 存储器的扩充	220
12.3.2 W29C020 存储器的使用	221
12.4 用 W29C020 存储字模的 LED 汉字显示屏项目设计	226
小结 12	230
习题 12	231
<b>第 13 章 A/D 转换与 D/A 转换简介</b>	<b>232</b>
13.1 A/D 器件 ADC0809 简介	232

13.2	ADC0809 的使用接口与编程 .....	233
13.2.1	ADC0809 与单片机的接口 .....	233
13.2.2	ADC0809 的应用编程 .....	235
	【案例 23】 基于 ADC0809 的模/数转换示例 .....	236
13.3	D/A 器件 DAC0832 简介 .....	238
13.4	DAC0832 的使用接口与编程 .....	239
13.4.1	DAC0832 与单片机的接口 .....	239
13.4.2	DAC0832 的应用编程 .....	239
	【案例 24】 DAC0832 的应用编程 .....	240
小结 13	.....	242
习题 13	.....	242

# 第1章 单片机基础

## 1.1 单片机简介

单片机本身是一块集成电路芯片，包含几大组成部分，如图 1-1 所示。用它再增添很少的一些电子元件，就可构成一个单片机基本应用系统。大大小小的单片机系统都有最基本、最起码的操作，即用程序设计语言来对其编程，以实现其系统功能。学习单片机技术，最主要的任务就是把单片机组成具体应用系统，并对其编程来实现所需功能。单片机系列众多，本书只介绍 MCS-51 系列，书中各案例均采用 STC89C52RC 型单片机。

下面，先介绍常用的 MCS-51 单片机的各引脚名称，其引脚排列如图 1-2 所示。

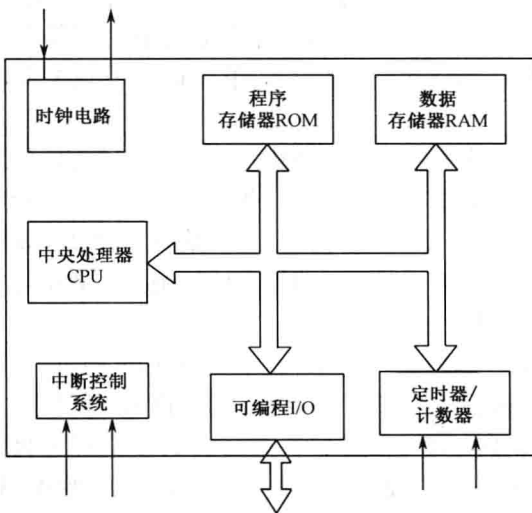


图 1-1 MCS-51 单片机内部结构图

1	P1.0	VCC	40
2	P1.1	P0.0	39
3	P1.2	P0.1	38
4	P1.3	P0.2	37
5	P1.4	P0.3	36
6	P1.5	P0.4	35
7	P1.6	P0.5	34
8	P1.7	P0.6	33
9	RST	P0.7	32
10	P3.0/RXD	$\bar{E}A$	31
11	P3.1/TXD	ALE	30
12	P3.2/INT0	PSEN	29
13	P3.3/INT1	P2.7	28
14	P3.4/T0	P2.6	27
15	P5/T1	P2.5	26
16	P3.6/WR	P2.4	25
17	P3.7/RD	P2.3	24
18	XTAL2	P2.2	23
19	XTAL1	P2.1	22
20	GND	P2.0	21

图 1-2 MCS-51 单片机引脚图

(1) 电源引脚。电源引脚 VCC (40)、地 GND (20)。

(2) 时钟电路引脚。XTAL1 (19)、XTAL2 (18)。

(3) 控制信号引脚。

① 复位信号 RST (9)：其作用是在单片机启动时，使系统从一个确定的初始状态开始运行。



② 锁存信号 ALE (30): 当访问外部存储器时, 其输出用于锁存地址的低位字节。

③ 选通信号 PSEN (29): 外部程序存储器读选通信号。

④ 选择信号 EA: 当 EA 为高电平时, 执行片内程序存储器, 当 EA 为低电平时, 执行片外程序存储器。

(4) 输入/输出引脚。输入/输出引脚共有 32 只, 组成四个口, 依次称为 P0、P1、P2、P3。每口均为 8 位双向 I/O 端口。一般地, 用“P0.0”表示 P0 口的第 0 位引脚, 用“P0.1”表示 P0 口的第 1 位引脚, 用“P3.3”表示 P3 口的第 3 位引脚, 注意, “口”和“位”都从 0 开始编号。

所有对单片机编程的最终目的, 就是让某些口的某些位, 按需要输出高电平或低电平, 以驱动被控电路。这 32 只 I/O 引脚, 既可作为输出引脚, 也可作为输入引脚, 即程序也可从检测某引脚为高电平还是低电平, 据此来判定程序流程该如何应对。

## 1.2 用单片机芯片组成单片机应用系统

用 STC89C52RC 单片机芯片、四位 LED 数码管 (共阳极)、六个 PNP 三极管、实时日历时钟芯片 DS1302、存储器芯片 AT24C02 和串行接口芯片 MAX232, 再加上两个晶振和一些阻容元件, 来组装一个单片机应用系统。组装时所用的电原理图如图 1-3 所示, 根据原理图而设计的印刷电路图如图 1-4 所示。本书中各个案例中的源程序, 都是按如图 1-3 所示的连接线路来考虑的。换在其他电路图版本的单片机应用系统上, 可能没有相应的效果。

在原理图中, 用相同的网络标识符来表示的电路各节点要相互连通。对照电路原理图和印刷电路图, 把集成电路插座等元件对号入座地焊在电路板上, 就能安装好本书实验所需的单片机学习板。在开始编程操作的第一次实训前, 先掌握四位 LED 数码管的显示要领, 从而掌握在单片机系统中的对数码管编程的基本方法。在我们即将开始的绝大多数单片机案例学习中, 其程序运作的实况都要借助数码管的显示来反映。

单片机系统中四位数码管电路的具体线路如图 1-5 所示。这个电路图告诉我们, 每位数码管由 7 个发光二极管组成, 其 7 个发光二极管的正极是连接在一起 (称为共阳极) 而再接在 PNP 管的集电极上的。要使这些二极管正向导通而发光, 一方面该 PNP 管的基极限流电阻左端必须加 0 电平 (低电平), 从而让该 PNP 管导通; 另一方面还要让需发光的二极管的负极加 0 (低电平) 电平, 数码管中的发光二极管才能发光。图 1-5 中的“P0.0”、“P0.1”、“P2.0”、“P2.1”等, 都是指单片机某 I/O 引脚。

从数码管的笔画组成图可知, 要让数码管显示数码“1”, 它的 7 个笔画中的 b、c 两笔画的发光二极管要点亮, 而其余笔画的发光二极管应熄灭。由于 a~g 这 7 个笔画是顺次接在 P0 口的第 0~6 位引脚上的, 这就需要 P0 口对应笔画 b、c 的引脚 P0.1、P0.2 输出 0 电平, 而其余笔画对应的 P0 口其他引脚输出 1 电平。要让 P0 口输出满足这一要求的数字电平, 只需在编程中使用一条“P0=0xf9;”语句即可。在 C 语言中, 0x 是十六进制数的标记。0x 后面是具体的十六进制数码 (0~9, a~f)。

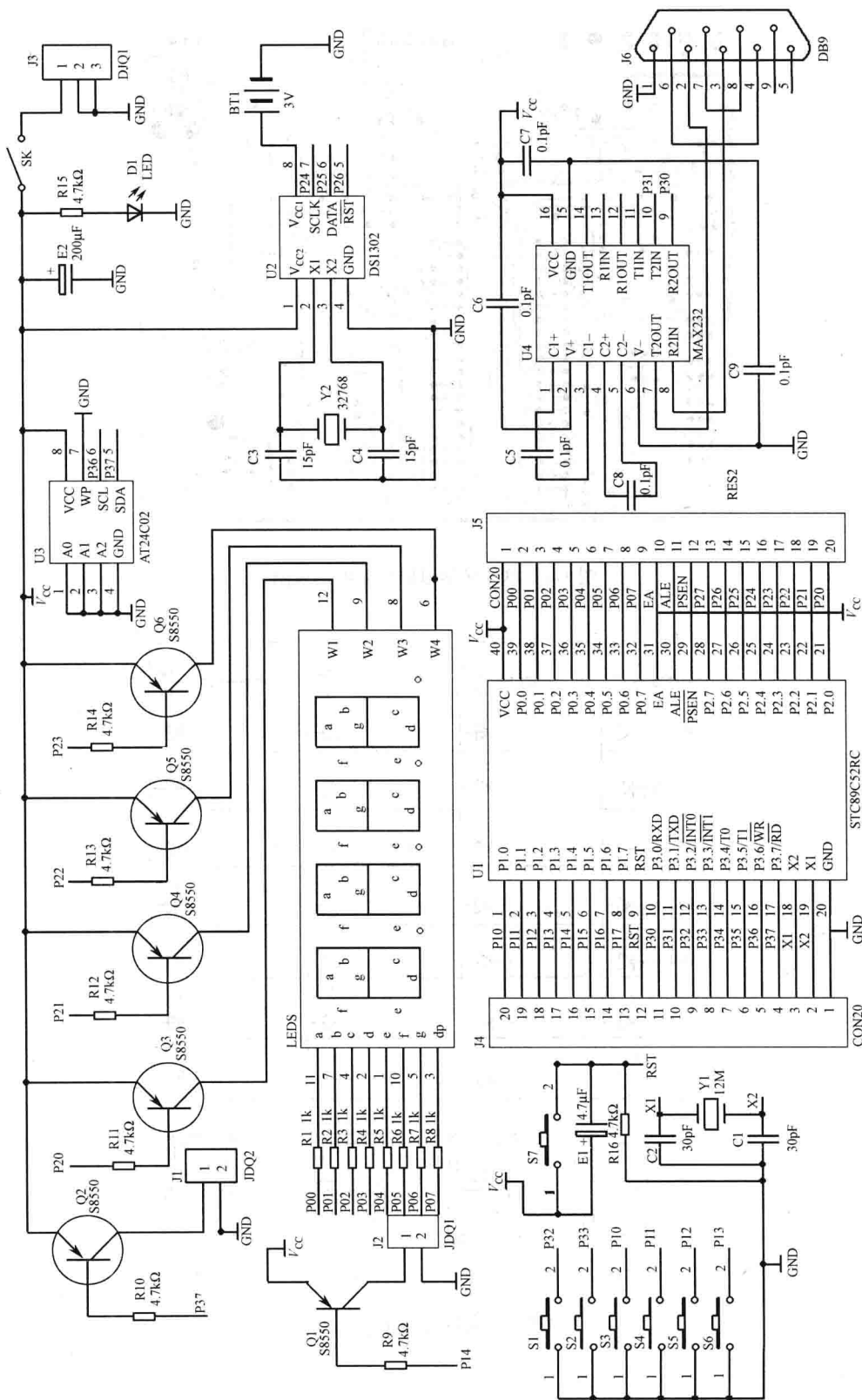


图 1-3 单片机学习板电原理图

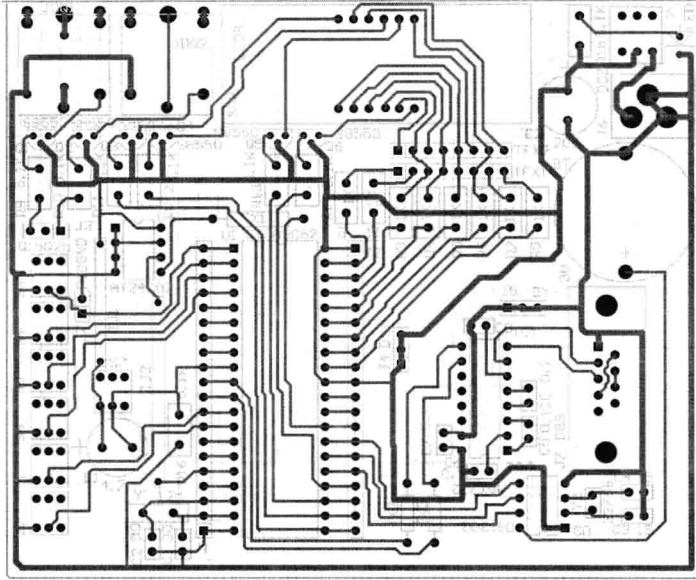


图 1-4 单片机学习板印制电路板图

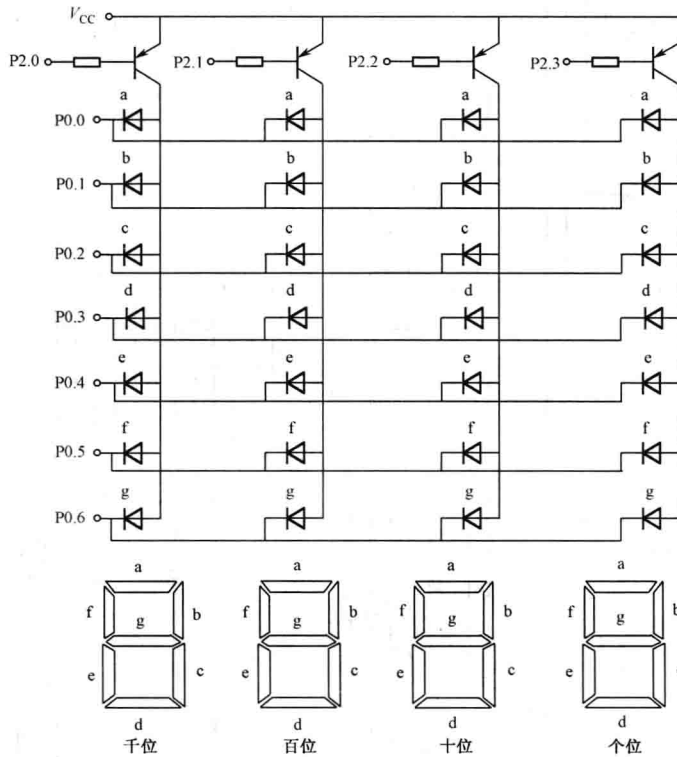


图 1-5 四位数码管连接电路图



在上面式子中的“0xf9”，就是用两位十六进制数来设置 P0 口 8 个电极的输出电平。而这两位十六进制数的来历，是由 7 个笔画的显示要求来确定的。把笔画 a、b、c、d、e、f、g 依次排为由低位到高位 7 位二进制数“g fedcba”。由于 P0 口的 P0.7 位接的是小数点的二极管的阴极，在不考虑小数点显示时，可在那 7 位二进制数的最左边添上一个 1（不让小数点发光），来构成 8 位二进制数“1g fedcba”。组成具体的 8 位二进制数时，把发光的笔画用 0 表示，把不发光的笔画用 1 表示即可。此后，再把所得的 8 位二进制数“11111001”，转换成两位十六进制数，也就是把高 4 位“1111”转换成十六进制的“f”，低 4 位“1001”转换成“9”，再在前面加上表示十六进制数的“0x”标记，就得到十六进制数 0xf9。

经过上面的分析可知，要让数码管显示“2”，就得让笔画 a、b、d、e、g 发光而其余笔画不发光，因此在要表示成二进制的数码串“1g fedcba”中，a、b、d、e、g 对应的位用 0 表示，其余的位用 1 表示，这就得到 8 位二进制数“10100100”，用十六进制数表示，就是“0xa4”，即只需有“P0=0xa4;”这一句，数码管就显示“2”。要显示“3”或其他的数码可以此类推。

从单片机电原理图上看到，四个数码管的同名笔画都是并接在一起的，因此，在同一时刻，只能显示同一数字。要让四个数码管显示出“1234”这种各位不同的数字，则需要四位数码管按位分时轮流重复显示，因此，每个数码管必须要都能被单独通电以受控制。正是这一原因，每个数码管的 7 个发光二极管共阳极引出线要接在一个 PNP 三极管的集电极上，且这个三极管的基极要接在 P2 口低四位中的一个电极上，从而可用对 P2 口低 4 位的位寻址变量赋 0 或赋 1，来开通或关闭某位数码管的显示。

### 1.3 对单片机编程以实现单片机系统的功能

单片机应用系统加电工作时，如果还没对单片机芯片编程，单片机系统还是不能达到使用目的，则必须用单片机编程语言对单片机先编入程序，让单片机按程序运行，单片机系统才能实现我们的意图。

对单片机编写程序可以采用汇编语言或 C 语言，即用键盘上的字符，按相应的语法规则来“打”成一篇“文章”，从而让单片机完全按照这篇“文章”的指挥来工作。对初学者而言，用 C 语言来学习单片机程序的编写，其起步与入门都要容易得多。下面，我们就用 C 语言，来对单片机编写一个简单程序，以实现单片机系统的一个简单功能。

#### 【案例 1】 在 4 位数码管的最左位上显示“8。”

##### 【源程序】

```
#include "reg51.h"
sbit qw = P2^0;
main()
{
    P0=0;
    qw=0;
}
```

### 【源程序要点提示】

程序的第 1 行称为预处理命令。“#”是预处理命令的标志，“include”是包含文件的预处理命令，“reg51.h”是所包含文件的文件名。

第 2 行是定义位寻址变量。“sbit”用来定义位变量，“qw”是被定义的位变量，P2 表示单片机的 I/O 口 2，“^0”指定 P2 口的第 0 位。这个定义就是用“qw”表示 P2 口的第 0 位。

第 3~7 行是程序的主函数。C 程序是由若干个 C 函数组成的，其中必须有一个也只能有一个名为“main”的函数。main 函数就是主函数。C 语言中，函数由函数头和函数体这两部分组成。第 3 行称为函数头，第 4~7 行称为函数体，第 4 行的大括号“{”称为函数体的开始标志，第 7 行的“}”称为函数体的结束标志，开始标志与结束标志之间是若干语句。语句以分号“;”为结束标志。在 C 语言中，程序由函数组成，并由函数来实现程序的系统功能，函数则用其内的语句完成具体操作。

第 5 行是赋值语句。其中的符号“=”称为赋值号，它的作用就是把其右边的数值赋给其左边的变量。在此是把 0 值赋给 P0，P0 是单片机 I/O 口 0 的代名词。MCS-51 单片机中，各 I/O 口都是 8 位的，语句“P0=0;”就是使 I/O 口 0 的 8 个 I/O 引脚全部输出低电平。

第 6 行也是赋值语句。qw 是我们给 I/O 口 2 的最低位引脚取的代名词。语句“qw=0;”就是使口 2 的最低位引脚 P2.0 输出低电平。

这个程序运行时的所有功能，是由两个赋值语句具体实现的。赋值语句“P0=0;”使 I/O 口 0 的 8 个引脚，即单片机芯片的第 39、38、37、36、35、34、33、32 全部输出低电平，从而使四位数码管的所有（32 只）发光二极管的负极均为低电平；赋值语句“qw=0;”使 I/O 口 2 的最低位引脚 P2.0，即单片机芯片的第 21 脚为低电平，这就使单片机学习板上的 PNP 管 Q3 的基极为低电平而导通，从而使最左边位数码管的 8 只发光二极管正极为高电平。

于是，程序用两个赋值语句来实现了仅让最左边位置的七段数码管及其小数点发光，即四位数码管显示为“8.”。

## 1.4 把程序写入单片机芯片的操作平台的操作步骤


随着单片机芯片制造业的发展，特别是我国单片机芯片产业的发展，把单片机程序写入单片机芯片的操作已经非常简单，即可以不必非要借助各种编程器来开发单片机系统了。本书所有案例中的单片机芯片，均采用是深圳宏晶科技有限公司生产的 STC 系列兼容单片机。该公司专门为其 STC 系列单片机配备了 STC-ISP 下载烧片编程软件，此软件可在该公司网站上下载。因此，对于 STC 系列单片机，借助 PC，只须用 Keil C51 单片机软件来编译出相应格式的二进制文件，再用 STC-ISP 下载软件把其下载（烧写）到单片机芯片中，即可开发出单片机应用项目。

下面，首先简要介绍如何在 PC 上建立开发平台，然后再通过把案例 1 中显示“8.”的单片机 C 语言程序，最终写入 STC89C52RC 芯片并实际运行的过程，来说明完成单片机编程的一般步骤。



要在 PC 上建立单片机的开发平台，需要一个编译软件和一个下载（烧片）软件。本书选用 Keil C51 和 STC-ISP。这两个软件都可从网上下载而得，同时，这两个软件的安装也很简单。PC 的操作系统可用一般的 Windows XP。

编程前要在 D 盘上建立一个名为“STC89C52”的文件夹，从而把操作过程中所产生的文件都存入其中，以便于管理。

在桌面上双击 Keil 的图标，即可进入该软件的初始界面，如图 1-6 所示。

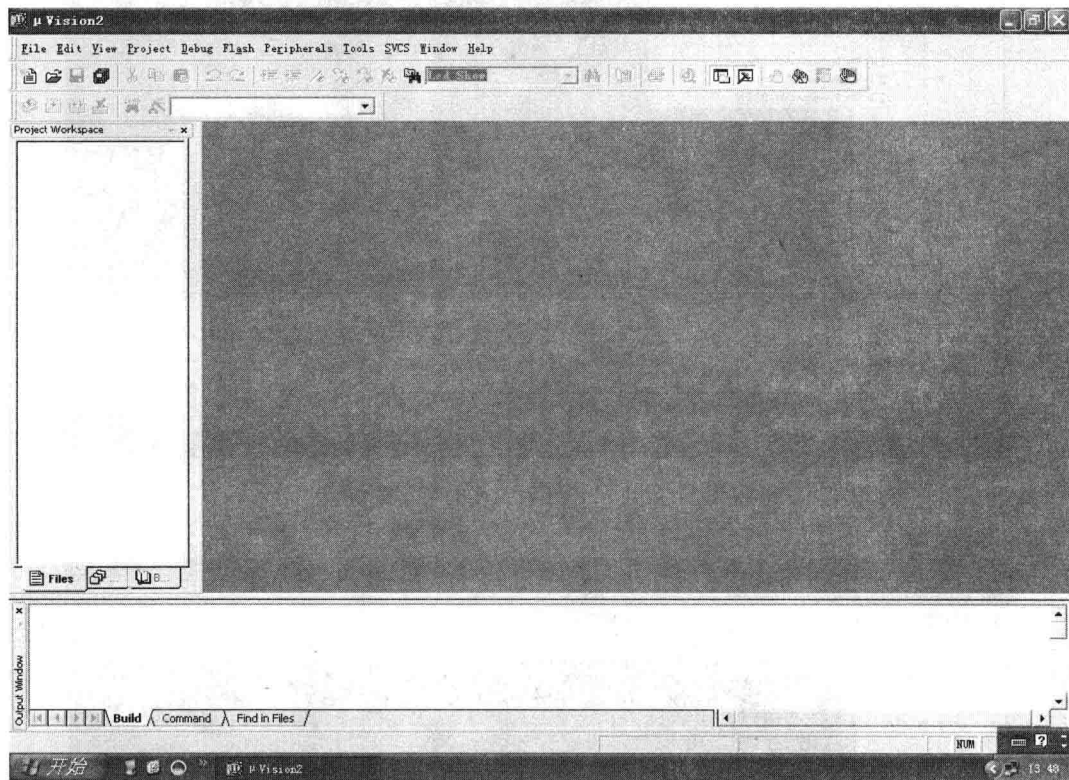


图 1-6 Keil 软件运行的初始界面

### 1. 建立新 nVisionz 工程

(1) 在 Keil 中是以“工程”的形式来组织 C51 程序的。在图 1-6 中，单击菜单栏中的 Project 菜单中的子菜单 New Project，如图 1-7 所示，弹出建立新 nVisionz 工程项目对话框，如图 1-8 所示。

(2) 在“保存在 (I)”下拉列表框中选择 D 盘下的“STC89C52”文件夹，在文件名文本框中，输入工程名“STC89C52”，然后单击“保存”按钮。此时，系统将弹出 CPU 选择窗口，如图 1-9 所示。

(3) 在列表框中选择并展开“Atmel”，如图 1-10 所示。

(4) 在图 1-10 中往下拉滚动条，选中“AT89C52”，如图 1-11 所示。

(5) 选中后单击“确定”按钮。随后，在弹出的询问框中，选择“是 (Y)”按钮即可。