

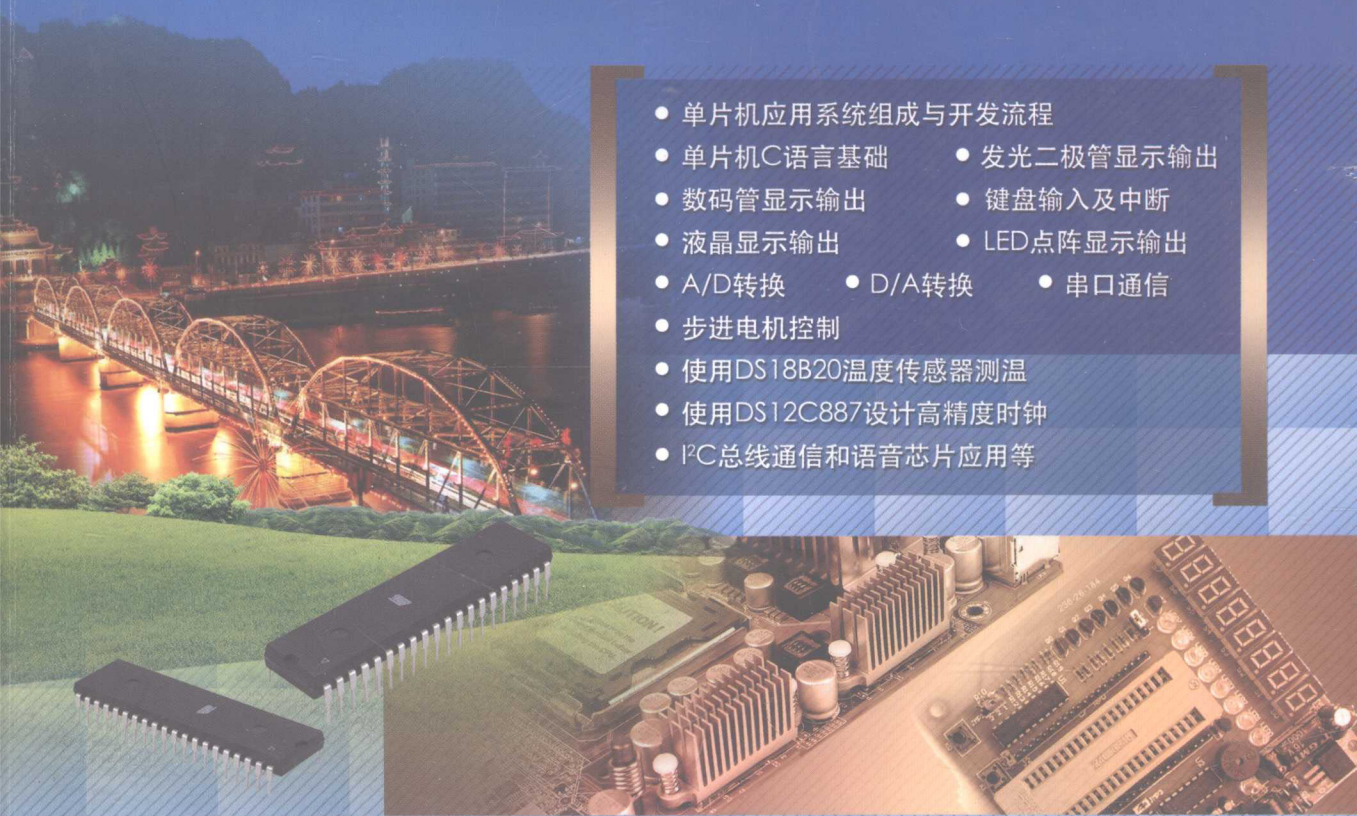


全国高等职业教育“十二五”规划教材
中国电子教育学会推荐教材
全国高职高专院校规划教材·精品与示范系列

学练一本通： 51单片机应用技术

◎ 陈宏希 主编 ◎ 贾 达 主审

- 单片机应用系统组成与开发流程
- 单片机C语言基础
- 数码管显示输出
- 液晶显示输出
- A/D转换
- D/A转换
- 串口通信
- 步进电机控制
- 使用DS18B20温度传感器测温
- 使用DS12C887设计高精度时钟
- I²C总线通信和语音芯片应用等
- 发光二极管显示输出
- 键盘输入及中断
- LED点阵显示输出



- ◆ 通过46个典型的单片机应用开发项目实例，使读者由浅入深地学习和掌握单片机系统开发方法和技巧
 - ◆ 所有实例都通过实际电路的教学实践验证，技术难度由浅入深、循序渐进，方便教学和读者自学
- 以解决工程实际问题为目标，将必要的单片机和C语言知识融入到具体项目中，使本课程教学变得简单配有免费的电子教学课件、思考题参考答案以及硬件电路图和源程序代码，详见前言



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

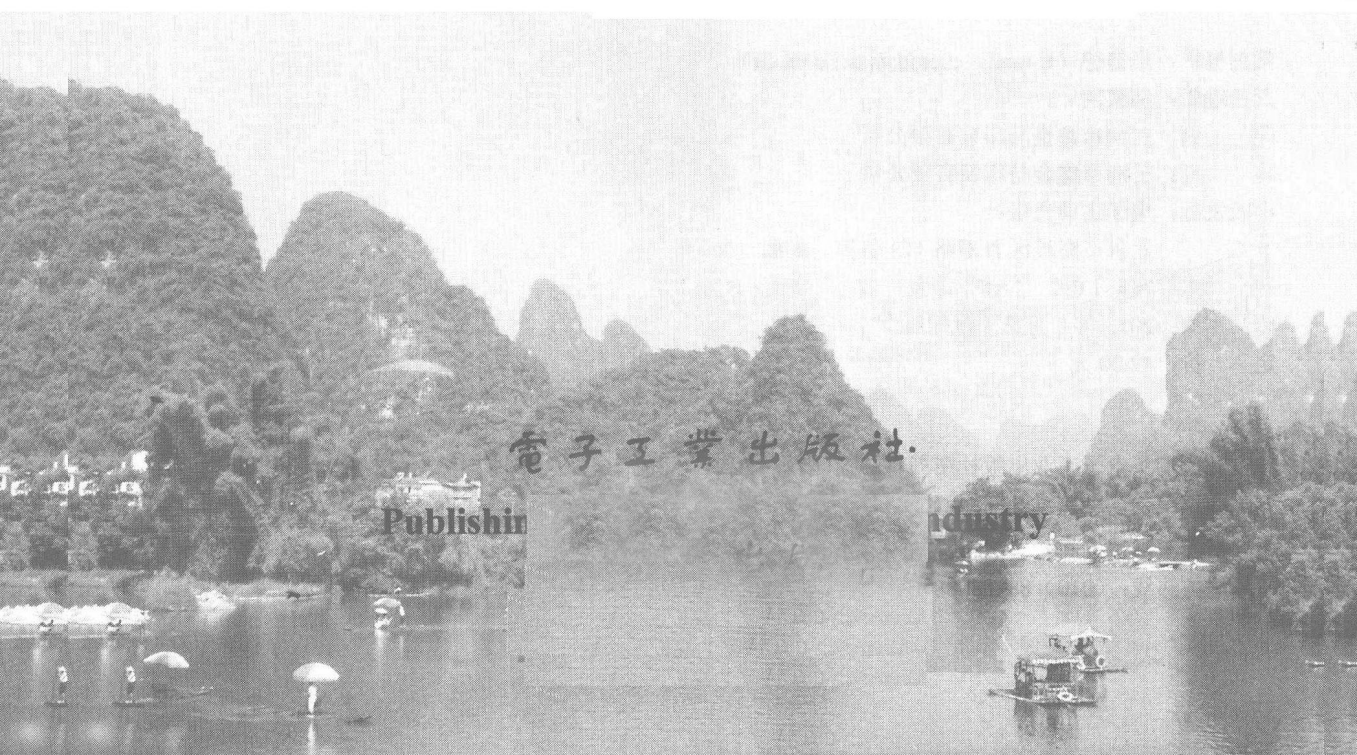
<http://www.phei.com.cn>

全国高等职业教育“十二五”规划教材
中国电子教育学会推荐教材
全国高职高专院校规划教材·精品与示范系列

学练一本通： 51 单片机应用技术

陈宏希 主编

贾 达 主审



電子工業出版社

Publishing

Industry

内 容 简 介

本书是一本特色鲜明、易学易练的 51 单片机入门教材,使用 C 语言编程,通过 46 个真实案例,由浅入深、循序渐进,介绍 51 单片机的基本知识、基本操作方法和应用开发技术。主要内容包括:发光二极管显示输出,数码管显示输出,键盘输入及中断,液晶显示输出,LED 点阵显示输出,A/D 转换,D/A 转换,串口通信,步进电机控制,使用 DS18B20 温度传感器测温,使用 DS12C887 设计高精度时钟,I²C 总线和语音芯片。附录还给出常用字符 ASCII 代码对照表以及单片机程序下载烧片的具体方法。

单片机应用开发是一门综合学科。为了给学习者提供最大方便,本书所有实例都给出完整的电路图和源程序清单,并就实例涉及的 C 语言知识和单片机知识,也给予适度及时的介绍、解释和说明,便于读者掌握与单片机相关的知识,并在实践中逐步提高综合应用与开发能力。另外,本书配有免费的电子教学课件和思考题参考答案。

本书图文并茂,语言严谨精练,操作步骤清晰易懂,为高职高专院校单片机技术课程的教材,也可作为应用型本科、成人教育、自学考试、电视大学、中职学校、培训班的教材,以及技术开发人员的参考用书。

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

学练一本通:51 单片机应用技术/陈宏希主编. —北京:电子工业出版社,2013.8

全国高职高专院校规划教材·精品与示范系列

ISBN 978-7-121-20749-5

I. ①学… II. ①陈… 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(2013)第 135010 号

策划编辑:陈健德(E-mail: chenjd@phei.com.cn)

责任编辑:郝黎明

印 刷:三河市鑫金马印装有限公司

装 订:三河市鑫金马印装有限公司

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:787×1 092 1/16 印张:22.5 字数:576 千字

印 次:2013 年 8 月第 1 次印刷

定 价:42.00 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



单片机是单片微型计算机的简称。目前, 51 系列、STC 系列、PIC 系列、AVR 系列和 430 等多个系列的单片机共存于市场和应用开发领域, 51 单片机以其简单实用、性价比高、应用开发技术成熟等优势, 占有单片机市场的大部分份额, 因此要很好地学习和掌握 51 单片机的开发技能。

单片机的应用开发是一个“软硬兼施”的过程, 硬件和软件缺一不可, 且需要密切配合和相互弥补。单片机的软件编程语言有汇编语言和 C 语言, 与汇编语言相比较, C 语言具有可读性、可移植性、可维护性好等优点, 使用 C 语言编写单片机的软件程序已是必然的选择。本书根据教育部最新的职业教育教学改革要求, 紧紧围绕电子行业技术发展与职业岗位技能, 结合高职教育人才培养目标与特点进行编写。

在编写过程中, 打破了以往传统的单片机学习模式, 设计一系列从简单到复杂的单片机应用开发项目实例, 使读者在由浅入深地学习和掌握这些实例的过程中, 边练边学, 步步深入, 逐步学习和掌握 C 语言单片机应用开发的方法和技巧。为保证实例的正确性, 本书所有实例都通过实际电路的实践验证, 同时, 为了方便读者实践学习, 同时给出所有实例完整的硬件电路图、源程序。对于实例涉及的 C 语言知识点、单片机知识点, 以实用、够用为原则, 以解决实际问题为最终目的, 将其融入具体项目的编写中, 进行现场及时的介绍、解释或说明; 实例不涉及的知识点暂且不提, 使单片机应用开发这门综合性学科, 变得简单易学和易用。这样, 无论你以前学过还是没有学过 C 语言、了解还是不了解 51 单片机, 都没有关系, 只要紧跟本书的章节和每个具体实例, 认真操作, 积极思考, 通过不断的研究和学习, 你一定会掌握单片机 C 语言应用开发的精髓, 成为单片机应用开发的高手。本书除正常的章节内容目录外, 作者还专门将本书涉及的 C 语言知识点和单片机知识点在目录中悉数列出, 方便读者查找使用。

本书由黄双成担任主编, 负责全书的统稿工作, 并编写了项目三, 吕恩胜编写了项目一, 张秋红编写了项目二的任务一到任务五, 刘庆花编写了项目二的任务六到任务九, 孙彩云编写了项目二的任务十到任务十二, 王雷和朱运晓编写了项目四及附录。另外, 本书在编写过程中, 参考了有关书籍和资料, 同时得到了郑州金特莱电子有限公司技术部工程师何威风的大力支持, 在此一并表示感谢。

在本书的编写过程中, 借鉴了许多现行教材的宝贵经验, 在此仅向这些作者表示诚挚的感谢!

由于作者水平有限, 加之时间仓促, 书中错误之处在所难免, 恳请广大读者朋友批评指正。

为了方便教师教学, 本书配有免费的电子教学课件、思考题参考答案以及硬件电路图和源程序代码, 请有需要的教师登录华信教育资源网 (<http://www.hxedu.com.cn>) 免费注册后进行下载, 有问题时请在网站留言或与电子工业出版社联系 (E-mail: hxedu@phei.com.cn)。

编者





第 1 章 基础知识	(1)
1.1 单片机的概念和应用领域	(1)
1.2 单片机应用系统的组成	(2)
1.3 单片机应用系统的硬件与软件开发特点	(3)
1.4 单片机应用系统的软件开发步骤	(4)
1.5 MCS-51 单片机的主要引脚	(15)
1.6 晶振电路和复位电路	(17)
1.7 电平	(18)
1.8 数制及其转换	(19)
1.9 单片机 C 语言基础	(20)
1.9.1 数据类型	(20)
1.9.2 常量和变量	(21)
1.9.3 C 语言的运算符	(23)
1.9.4 C 语言程序基本结构	(24)
思考题 1	(27)
第 2 章 发光二极管的显示输出	(28)
2.1 发光二极管	(28)
2.2 点亮一只发光二极管	(29)
2.2.1 硬件电路	(29)
2.2.2 源程序及其结构分析	(30)
C 语言知识 sfr 和 sbit	(33)
C 语言知识 赋值语句	(35)
实例 1 使用 P1 口	(38)
实例 2 使用 P0 口	(39)
2.3 一只闪烁的发光二极管	(39)
C 语言知识 循环语句	(40)
2.3.1 源程序及其结构分析	(42)
C 语言知识 宏	(43)
C 语言知识 注释	(43)
2.3.2 for 循环延时时间的测量	(44)
2.3.3 延时子函数及其调用	(46)

2.4	流水灯	(49)
2.4.1	硬件电路	(49)
2.4.2	源程序	(50)
2.4.3	使用数组查表方法实现流水灯	(51)
	C 语言知识 数组	(51)
2.4.4	使用位运算中的左/右移位方法	(53)
2.5	蜂鸣器控制和继电器控制	(55)
	思考题 2	(57)
第 3 章	数码管显示输出	(59)
3.1	数码管的结构与分类	(59)
3.1.1	数字和字符的数码管显示图样	(60)
3.1.2	共阳和共阴数码管	(60)
3.2	数码管的显示输出原理	(61)
3.2.1	共阳数码管的显示输出原理	(61)
3.2.2	共阴数码管的显示输出原理	(62)
	实例 2 用数码管静态显示	(64)
	实例 3 用数码管动态显示字符 (1)	(66)
	实例 4 用数码管动态显示字符 (2)	(67)
	实例 5 用数码管动态显示时间	(70)
	思考题 3	(72)
第 4 章	键盘输入及中断	(73)
4.1	独立按键	(74)
	实例 5 按键计数	(75)
	C 语言知识 if 语句	(76)
	实例 6 多个按键的识别	(80)
	C 语言知识 switch 语句	(82)
	实例 7 用一键实现多功能按键	(83)
4.2	矩阵键盘	(85)
	实例 8 4×4 矩阵键盘序号显示	(86)
4.3	中断	(92)
4.3.1	中断的概念	(92)
4.3.2	单片机中使用中断的意义	(93)
4.3.3	单片机的中断源	(93)
	实例 9 使用外部中断控制数字显示	(93)
4.3.4	单片机的外部中断	(96)
	C 语言知识 中断服务子函数	(98)
	实例 10 有优先级的外部中断控制数字显示	(99)
4.4	定时器/计数器	(102)
4.4.1	定时器/计数器的基本概念	(102)

实例 11 定时器工作在方式 1 下的电子钟设计	(103)
4.4.2 机器周期与外接晶振频率的关系	(106)
4.4.3 定时器的工作原理	(107)
4.4.4 与定时器有关的寄存器	(108)
实例 12 定时器工作在方式 2 下的电子钟设计	(110)
4.4.5 定时器/计数器初值的计算与装载	(112)
实例 13 定时器工作在查询方式下的电子钟设计	(113)
思考题 4	(115)
第 5 章 液晶显示输出	(116)
5.1 1602/0802 字符液晶显示输出	(116)
5.1.1 1602/0802 字符型液晶的引脚定义	(117)
5.1.2 1602/0802 液晶的特点与使用	(117)
实例 14 1602 液晶的字符显示	(119)
5.2 不带字库 12864 液晶显示输出	(121)
5.2.1 12864 点阵液晶的引脚功能	(121)
5.2.2 12864 点阵液晶的特点与使用	(122)
实例 15 无字库 12864 液晶的显示输出	(124)
5.2.3 51 单片机存储器类型和数据的存储类型	(136)
5.2.4 存储器映像寻址	(137)
5.2.5 对片外存储器的访问	(137)
5.3 带字库 12864 液晶显示输出	(138)
5.3.1 带字库 12864 液晶的引脚功能	(138)
5.3.2 带字库 12864 液晶的特点与使用	(139)
实例 16 并行工作方式下带字库 12864 液晶显示输出	(142)
实例 17 串行工作方式下带字库 12864 液晶显示输出	(147)
思考题 5	(150)
第 6 章 LED 点阵显示输出	(151)
6.1 8×8 LED 点阵显示输出	(151)
6.1.1 初识 8×8 LED 点阵	(151)
6.1.2 8×8 LED 点阵的显示原理	(153)
实例 18 8×8 LED 点阵显示输出	(154)
实例 19 8×8 LED 点阵显示运动的箭头	(158)
6.2 16×16 LED 点阵显示输出	(160)
6.2.1 用 8×8 LED 点阵模块搭建 16×16 LED 点阵	(160)
6.2.2 16×16 LED 点阵的驱动	(160)
实例 20 16×16 LED 点阵屏显示汉字	(162)
6.3 32×64 LED 点阵显示输出	(166)
实例 21 使用 32×64 LED 点阵显示汉字	(168)

思考题 6	(171)
第 7 章 A/D 转换	(172)
7.1 A/D 转换器的转换分辨率和时间	(172)
7.2 ADC0809 的功能与使用	(173)
实例 22 模拟口线方式下 ADC0809 模数转换	(175)
实例 23 总线控制方式下 ADC0809 模数转换	(178)
C 语言知识 指针	(181)
7.3 AD574 的功能与使用	(186)
7.3.1 AD574 的引脚功能	(186)
7.3.2 AD574 控制逻辑及特点	(188)
实例 24 总线控制方式下 AD574 单极性模数转换	(190)
实例 25 模拟口线方式下 AD574 单极性模数转换	(193)
7.4 ADC0832 的功能特点与使用	(196)
7.4.1 ADC0832 的引脚功能	(196)
7.4.2 ADC0832 的特点	(197)
实例 26 用 ADC0832 实现 A/D 转换	(198)
7.5 TLC2543 的功能特点与使用	(201)
7.5.1 TLC2543 的引脚功能	(201)
7.5.2 TLC2543 的特点	(202)
实例 27 用 TLC2543 实现 A/D 转换	(204)
思考题 7	(207)
第 8 章 D/A 转换	(208)
8.1 D/A 转换器的分辨率和建立时间	(208)
8.2 DAC0832 的功能特点与使用	(209)
实例 28 多种工作模式下的 DAC0832 数模转换	(211)
实例 29 用两片 DAC0832 实现多模式数模转换	(214)
8.3 AD7237 的结构功能及特点	(217)
实例 30 AD7237 数模转换	(221)
8.4 TLV5625 的功能特点与使用	(224)
实例 31 TLV5625 数模转换	(227)
8.5 AD7543 的引脚功能与使用	(229)
实例 32 AD7543 数模转换	(231)
思考题 8	(232)
第 9 章 串口通信	(233)
9.1 串行通信的分类	(233)
9.2 串行通信的制式	(234)
9.3 单片机的串口缓冲器和工作寄存器	(235)
9.3.1 串口缓冲器 SBUF	(235)

9.3.2	串行口的工作寄存器	(235)
9.3.3	串行口工作方式	(237)
9.3.4	波特率	(238)
实例 33	单片机间的串行通信	(239)
9.4	单片机多机通信	(242)
实例 34	三个单片机间的通信与显示控制	(243)
9.5	单片机与 PC 间通信	(250)
实例 35	单片机向 PC 发送和显示数据	(253)
实例 36	PC 向单片机发送数据	(255)
思考题 9		(257)
第 10 章	步进电机控制	(258)
10.1	步进电机的工作原理与控制	(258)
10.1.1	步进电机的分类	(258)
10.1.2	步进电机的工作原理	(259)
10.1.3	步进角和励磁线圈通电方式	(259)
10.1.4	步进电机的驱动电路	(261)
10.2	步进电机的线路连接	(270)
10.2.1	二相四线步进电机	(270)
10.2.2	4 相 6 线步进电机	(270)
10.2.3	4 相 8 线步进电机	(270)
实例 37	用独立按键控制步进电机的转速	(271)
思考题 10		(275)
第 11 章	使用 DS18B20 温度传感器测温	(276)
11.1	DS18B20 温度传感器	(276)
11.2	DS18B20 温度传感器的测温工作原理	(277)
11.2.1	DS18B20 内部的存储器	(277)
11.2.2	DS18B20 的指令	(279)
11.2.3	DS18B20 的通信规则	(280)
11.2.4	DS18B20 的初始化、数据读写操作时序	(280)
实例 38	用一片 DS/8B20 实现温度测量	(283)
实例 39	用四片 DS18B20 实现温度测量	(287)
思考题 11		(292)
第 12 章	使用 DS12C887 设计高精度时钟	(293)
12.1	时钟芯片 DS12C887 的特性与引脚功能	(293)
12.2	DS12C887 实时时钟芯片工作原理	(295)
12.2.1	DS12C887 内部的存储器	(296)
12.2.2	DS12C887 工作时序分析	(299)
实例 40	可调高精度时钟设计	(300)

实例 41 具有闹铃功能的高精度时钟设计.....	(310)
思考题 12.....	(317)
第 13 章 I ² C 总线和语音芯片.....	(318)
13.1 单片机与 I ² C 总线通信.....	(318)
13.1.1 I ² C 总线与单片机的连接和工作方式.....	(318)
13.1.2 I ² C 总线的通信协议.....	(319)
13.2 串行 I ² C 总线 E ² PROM 芯片 AT24C02.....	(322)
实例 42 使用 I ² C 总线通信对 AT24C02 进行数据读/写操作.....	(325)
13.3 XF-S4240A 语音合成模块及应用.....	(329)
实例 43 采用 UART 通信方式通过 XF-S4240 播放合成语音.....	(332)
实例 44 采用 SPI 通信方式通过 XF-S4240 播放合成语音.....	(334)
实例 45 采用 I ² C 通信方式通过 XF-S4240 播放合成语音.....	(336)
实例 46 采用 I ² C 通信方式在 AT24C02 中存/取数据并使用 XF-S4240 播放合成语音....	(338)
思考题 13.....	(342)
附录 A 常用字符与 ASCII 码对照表.....	(343)
附录 B 单片机程序的下载烧片.....	(344)
参考文献.....	(350)

第1章

基础知识

1.1 单片机的概念和应用领域

在深入、全面学习单片机开发应用技术之初，初学者一定会提出许多与单片机有关的问题（如下所列），正确回答并理解掌握这些问题，无疑为学好、用好单片机开了个好头。

1. 什么是单片机

单片机是单片微型计算机的简称。图 1-1 是常见单片机实物图，其中包括 51 系列、STC 系列、PIC 系列和 AVR 系列单片机。说它们是一款计算机，读者可能会产生质疑：这也是计算机？从外观来看，它与人们日常使用的台式计算机、笔记本电脑大相径庭，但它们的确是计算机。俗话说：“麻雀虽小，五脏俱全。”一般计算机所拥有的基本结构，如处理器、存储器、输入/输出设备等，单片机同样拥有。之所以称为单片计算机，是因为它们将处理器、存储器、输入/输出等组件全部集成在一块芯片上的原因。

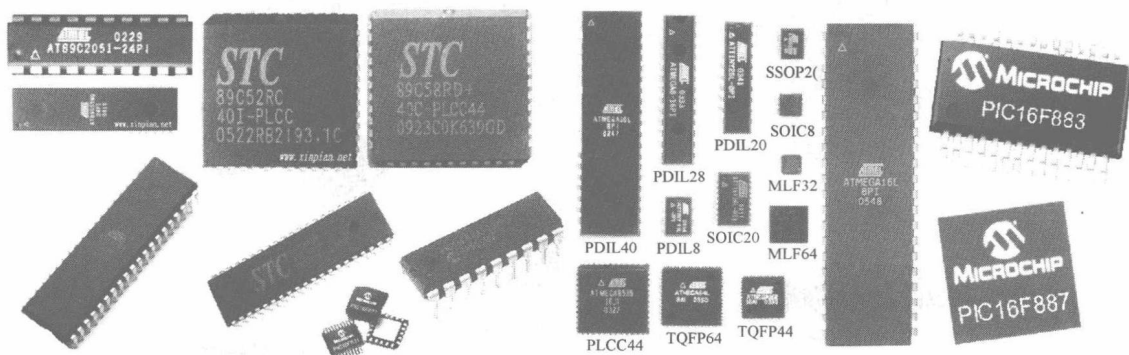


图 1-1 常见单片机实物图



2. 单片机能干什么

目前，单片机在工业控制、智能仪器仪表、消费类电子、军事、医用、网络通信等领域都有十分广泛的应用，且其应用领域还在进一步地拓展。

(1) 工业控制领域：工业现场实时测控、数据采集等。

(2) 智能仪器仪表领域：数字万用表、数字示波器、数字信号源、数字频率计等。

(3) 消费类电子领域：洗衣机、电冰箱、空调、电视机、微波炉、IC 卡、电子玩具、数码相机、数码摄像机等。

(4) 军事领域：飞机、坦克、导弹、鱼雷、制导、智能武器等。

(5) 医用领域：呼吸机、监护仪、超声诊断、病床呼叫等。

(6) 网络通信领域：电话机、手机、程控交换机、楼宇自动呼叫、无线通信等。

总之，单片机的应用已经渗透到人类工作和生活的多个领域。

1.2 单片机应用系统的组成

单片机应用系统是软件、硬件相结合的综合应用系统，软件和硬件二者缺一不可，如图 1-2 所示。

对于硬件，指的就是单片机。单片机种类较多(51、AVR、PIC、STC 等系列)，我们选用 51 系列单片机，图 1-1 中就有 40 个引脚的双列直插式(PDIP40)封装的 51 系列单片机。

一个单片机应用系统的硬件部分，只有单片机是远远不够的。单片机是必需的，但还需其他外部硬件设备或元件，这些外部硬件设备或元件一般被称为外部设备，简称外设，如图 1-3 所示。这些外设器件与单片机一起工作，才能完成或者实现具体功能。

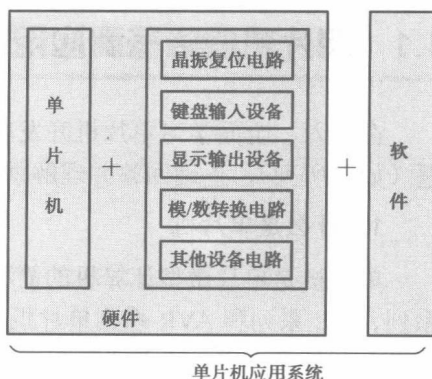


图 1-2 单片机应用系统组成

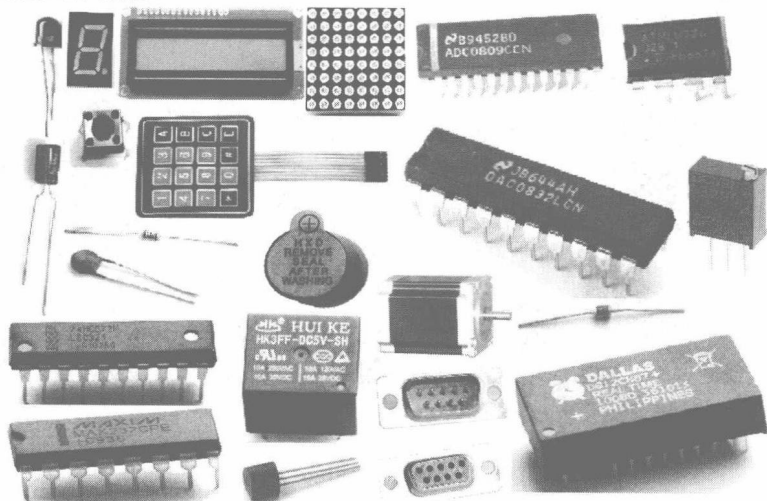


图 1-3 单片机硬件系统中使用的部分外设



对于软件,软件开发简言之就是编写程序。选用什么语言、用什么软件、怎样编写单片机 C 语言程序等问题都将接踵而来,以下先简单介绍一下这些问题。

选用什么语言? C 语言!其实,在单片机软件系统开发中,有两种编程语言: C 语言和汇编语言。之所以选用 C 语言而未选用汇编语言,是由于在编写单片机程序方面, C 语言相比汇编语言有许多优势,在此,不再介绍那些优势具体是什么,总之,就用 C 语言了。

用什么软件? Keil!这是目前用得最多、最广泛的单片机 C 语言软件开发环境。Keil 有 μ Vision2、 μ Vision3、 μ Vision4 等版本,这里选用 μ Vision3。

怎样编写单片机 C 语言程序?这个问题不是一两句话能解释清楚的,但有一个总体的目标,这就是以硬件为基础,准确运用 C 语言,编写出结构完整、具有一定功能、能在单片机上实际运行、能实现具体功能的 C 语言程序。C 语言的语法比较少,单片机中用到的 C 语言语法更少,所以在使用 C 语言进行单片机编程前,没有必要将 C 语言的全部知识系统地学习或复习一遍。以往,要学习单片机,并准备用 C 语言编写程序,一般是先系统地学习 C 语言,再系统地学习单片机知识,最后将二者结合起来,再学习 C 语言的单片机开发应用。现在,我们的思路与以前大不相同,具体做法是,打破传统的知识体系框架,设计一系列从简单到复杂的单片机应用开发项目,将 C 语言知识、单片机知识融入具体项目中,在具体应用开发项目的实际工作过程中,学习并掌握基于 C 语言的单片机应用开发技术。教学中,针对具体项目,仅对本项目涉及的 C 语言知识点、单片机知识点做细化讲解,本项目不涉及的知识点暂且不提,以够用、实用为原则,以解决实际问题为最终目的。在一个个应用项目开发的实践和实现过程中,逐步学习和掌握 C 语言单片机开发的方法和技巧。在本书后续章节中,将举出大量单片机 C 语言应用开发的实例,结合这些具体实例,无论读者以前学过还是没有学过 C 语言、了解不了解单片机知识,都没有关系,只要紧跟本书的章节和每一个具体实例,深入研究和学习,相信读者一定会掌握单片机 C 语言开发的技术,成为单片机应用开发的高手。

1.3 单片机应用系统的硬件与软件开发特点

如图 1-2 所示,单片机应用系统包括硬件系统和软件系统,对应地,单片机应用系统的开发,也主要包括硬件部分开发和软件部分开发。实际工作中,当软件和硬件开发工作完成之后,还要将软件开发生成的 .hex 文件下载(俗称“烧片”)到单片机的程序存储器 ROM 中。由于下载烧片工作一般使用专门的设备(如编程器)来完成,因此操作比较简单,使用者只要正确操作该设备及相关软件,一般都能成功下载烧片,所以在单片机应用系统的开发流程中,下载烧片这一开发环节往往被淡化或者忽略不提。本书沿袭惯例,对下载烧片环节不做过多说明,有兴趣的读者请参看本书附录 B 或其他资料。此处专门提出这一环节,只是希望能引起读者的注意,不要忘记最后这一环节。

再说单片机开发中的软件和硬件,可以说,硬件是整个应用系统的基础,而软件则依赖于系统的硬件。在硬件不再改变的条件下,软件程序的改变,可以部分改变系统的功



能。但从整体而言，单片机应用系统的开发，本是一个“软硬兼施”的过程，软件和硬件需要互相弥补，密切配合。部分用硬件不能实现或者不便实现的功能，可以考虑用软件去补充或实现；同样道理，选择适当的硬件，也可以弥补软件功能上的缺憾或不足，二者相得益彰，互相配合，才能使任务最终实现。

就单片机应用系统的开发流程而言，硬件和软件在开发次序方面，原则上无先后之分，先开发哪一个都可以，有时二者还可以同步进行设计和开发。但考虑到硬件是基础，所以通常情况下是先开发硬件，再开发软件。

对于硬件部分的开发，简言之就是设计和加工电子线路板，或者手动焊接电子线路板。具体而言，硬件部分的开发绝非易事。首先是硬件电路所用元器件种类繁多，特性各异，全部掌握或了解实属不大可能；再次，硬件电路的设计开发需要很高的设计技巧，绝非一日之功可以成就。本书作为单片机初学者的入门教材，对后续各章节中列举的各个实例，针对不同的任务要求，直接给出了硬件电路图，并对部分主要和重要元器件的特性和功能，也做了详细介绍，方便初学者参考学习并逐渐积累硬件设计的经验，以便日后能自行设计较复杂的硬件电路。

相对于硬件部分的开发，软件部分的开发则较为灵活。正如前面所述，开发语言可以选择汇编语言，也可以选择 C 语言，还可以选择汇编语言和 C 语言混合编程。本书主要选用的是 C 语言，开发环境则选用 Keil μ Vision3。需要特别说明的是，在软件开发部分，开发流程几乎是固定不变的，如图 1-4 所示，主要包括工程建立、源程序编写和编译调试三个步骤；软件开发的成果是生成扩展名为 .hex 的文件，该文件被用于下载或者烧片到单片机的程序存储器 ROM 中，供单片机上电后再读出来执行，从而驱动或者控制外部设备按照预定的要求正常工作。从软件开发的流程可见，针对不同的项目任务，编写的源程序不大相同，其余步骤则基本固定不变。与硬件开发的处理思路一样，本书对于后续各章节中列举的各个实例，针对不同的任务要求，直接给出源程序清单，对主要和重要的功能模块，给予解释和说明，供初学者参考学习，而软件开发过程中的其他步骤则一概简化或略去，重点放在功能实现和源程序的编写上，其他与 Keil 相关的软件操作则被淡化，只因这些操作和步骤基本上是固定不变的。

鉴于以上概述，以下仅就软件开发的具体流程做详细说明。图 1-5 是在图 1-4 的基础上，细化其中的三个主要步骤所得单片机软件开发流程图。

1.4 单片机应用系统的软件开发步骤

1. 工程建立

1) 启动 Keil μ Vision3 软件

假设 Keil μ Vision3 软件已正确安装，启动 Keil μ Vision3 软件与启动其他软件的方法完全相同：选择【开始】→【程序】→【Keil μ Vision3】选项，或者直接双击桌面上 Keil μ Vision3 的快捷方式图标，均可启动 Keil μ Vision3。启动过程中，屏幕出现如图 1-6 所示的启动界面。启动界面消失后，Keil μ Vision3 就进入了如图 1-7 所示的编辑界面，此时 Keil μ Vision3 软件已成功启动。

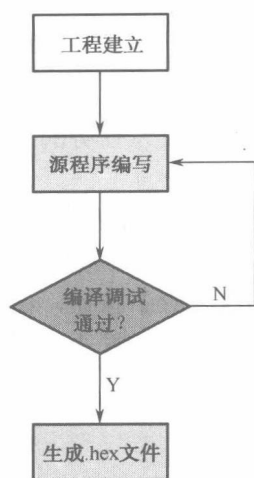


图 1-4 软件开发流程



图 1-6 Keil μVision3 启动界面

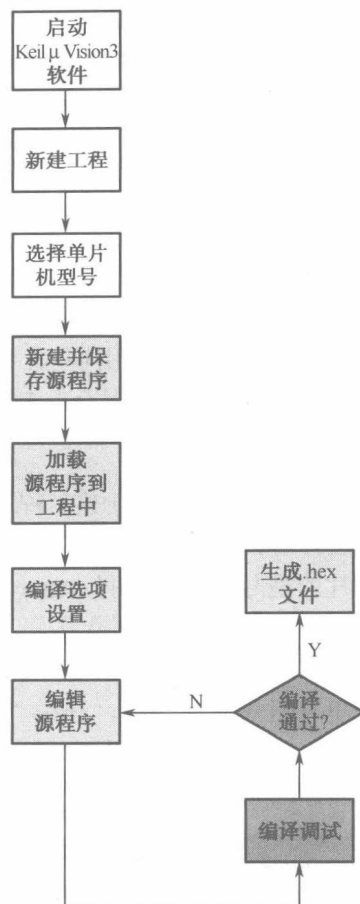


图 1-5 细化的软件开发流程

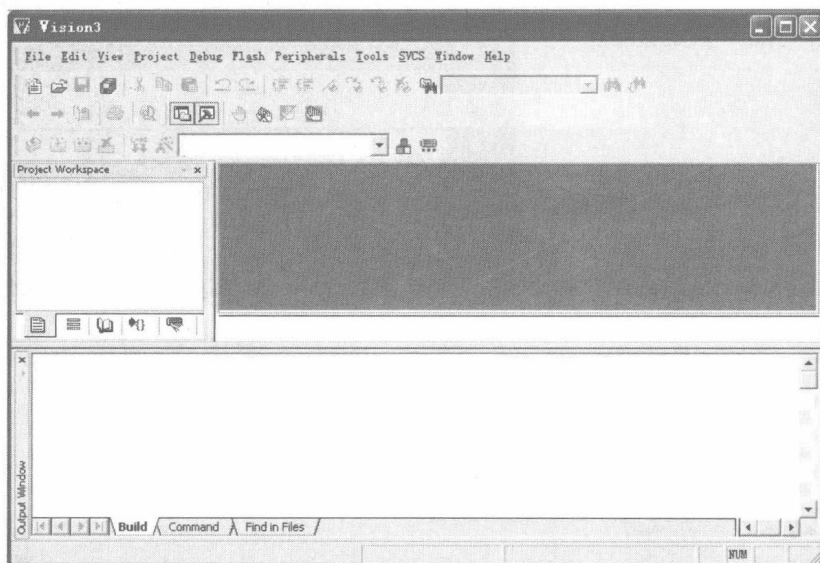


图 1-7 Keil μVision3 启动成功后的编辑界面



2) 新建工程

Keil 与现在众多的软件开发环境一样，均采用工程或者项目的管理方式来管理文件。一个应用开发任务就是一个工程或项目。一般来说，以开发任务的名称作为新建的文件夹名，所有与该开发任务有关的文件都存放在该文件夹中，不同开发任务的文件夹名一般不相同。

下面首先新建工程。选择 Keil μ Vision3 编辑界面中的【Project】菜单，在展开的下拉菜单中选择【New Project】选项，如图 1-8 所示。在随后打开的保存新建工程对话框中，选择新建工程的保存路径，并为新建的工程命名，最后单击【保存】按钮，保存该新建的工程。注意，工程的扩展名不用输入，系统会默认选择为“.uv2”，如图 1-9 所示。此处将工程暂且命名为“LED1”，并保存在 F 盘的文件夹 LED 下。

3) 选择单片机型号

当单击图 1-9 所示保存新建工程对话框中的【保存】按钮之后，Keil μ Vision3 会弹出单片机型号选择对话框，如图 1-10 所示。在该对话框中，要求用户选择该工程准备使用的单片机芯片的型号。此处选择 Atmel 公司生产、使用最普及的 AT89C51 芯片。具体操作是：在图 1-10 左侧栏所列众多厂家中找见“Atmel”，单击“Atmel”名称前面的加号“+”，展开 Atmel 公司生产的单片机系列产品，如图 1-11 所示；接着，在展开的 Atmel 公司生产的单片机系列产品中，选中“AT89C51”，如图 1-12 所示；最后，单击【确定】按钮，芯片型号的选择完成。此时，Keil μ Vision3 会弹出如图 1-13 所示的对话框，询问是否复制并添加标准 8051 启动代码到当前工程，一般单击【是】按钮即可。

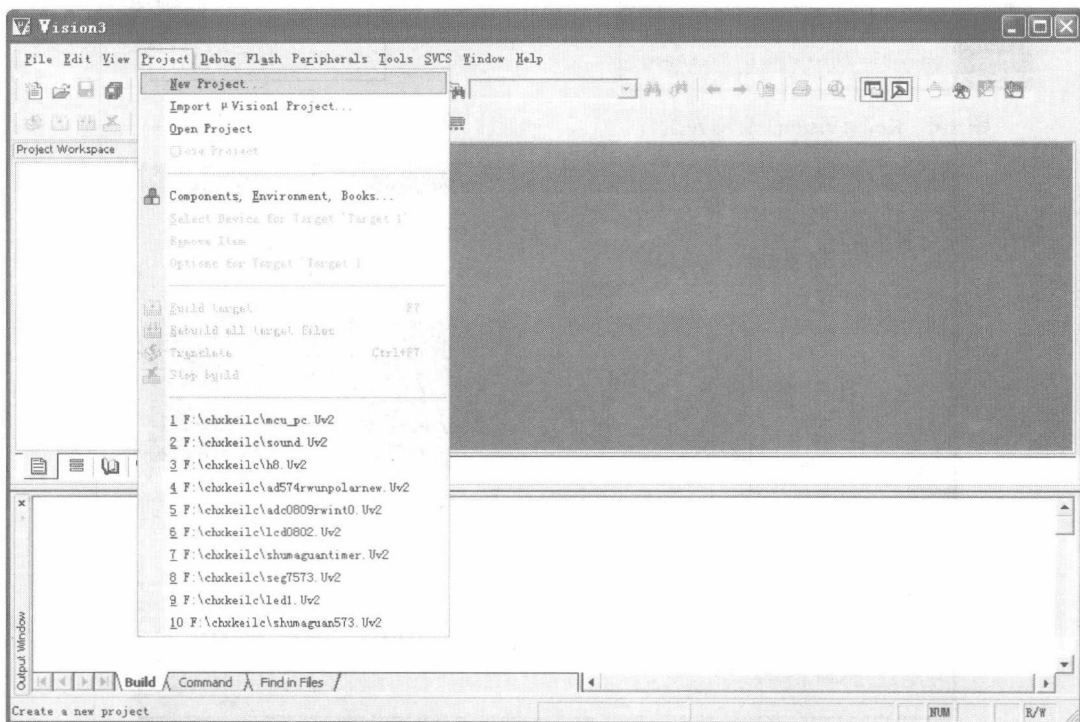


图 1-8 新建工程

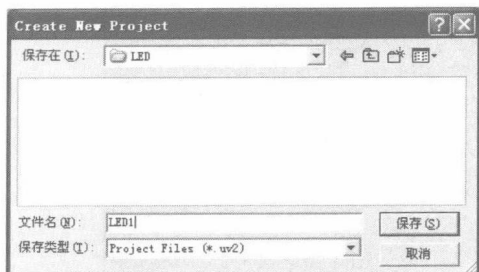


图 1-9 保存工程

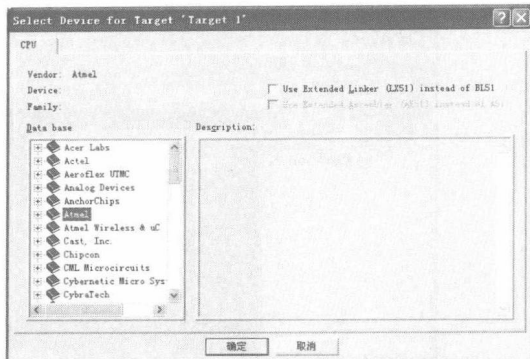


图 1-10 选择 51 单片机的生产厂家

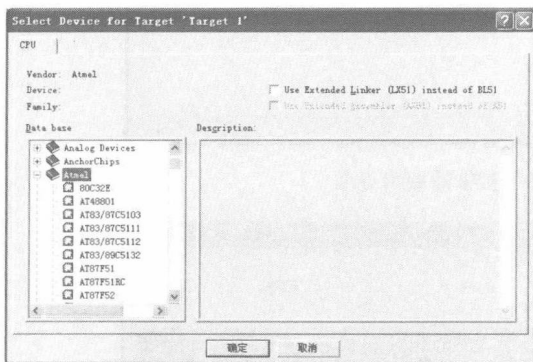


图 1-11 展开 Atmel 公司生产的单片机产品

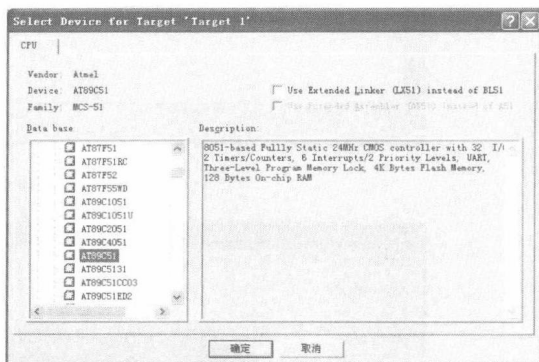


图 1-12 选中 Atmel 公司生产的单片机 AT89C51

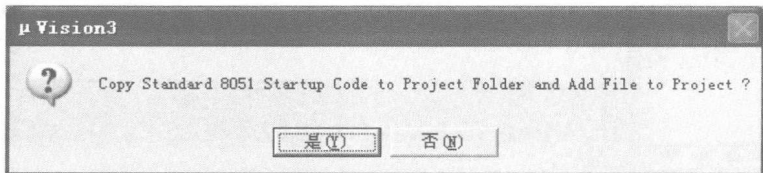


图 1-13 是否添加 8051 启动代码到当前工程对话框

完成以上步骤之后，屏幕重新回到如图 1-14 所示的编辑界面，图 1-14 与图 1-7 基本相同，不同之处是左侧“工程管理工作台”中的内容。图 1-7 中，“工程管理工作台”中什么也没有；而图 1-14 中，“工程管理工作台”中有名为“Target 1”的对象，单击其名称前面的加号，还可以看到该对象内所包含的文件。

2. 源程序编写

新建工程完成之后，下面该进入源程序的编写环节了。

1) 新建并保存源程序文件

选择【File】→【New】选项，或者直接单击工具栏中“创建一个新文件”快捷图标，都可以新建一个默认名字为“Text 1”的文件。新建之后，“1”形光标就在“Text 1”文件的编辑窗口中闪烁，如图 1-15 所示。需要说明的是，默认的文件名也许由于多次新建文件而出现“Text x”的字样，其中 x 是一整数，因为马上要将该文件进行保存或另存为其他名称的文件，所以原来的文件名是什么都无关紧要。