

简简单单学通 51单片机开发

(21小时高清多媒体教学视频)

王晋凯 等编著

39个完整实例、42个实例扩展、50幅整机电路、91幅流程图、210个习题

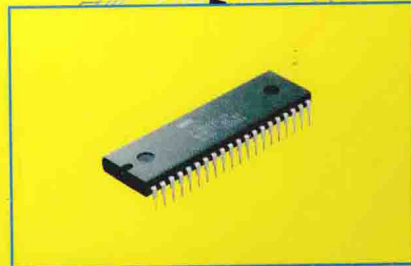
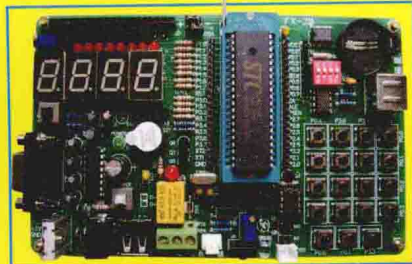
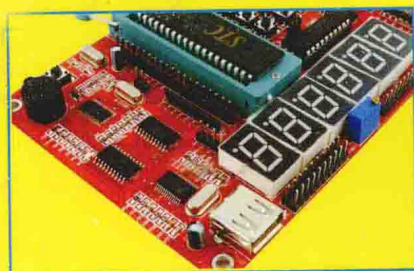
一个电子技术的狂热爱好者、研发者的51单片机学习与开发感悟
学以致用，将51单片机的理论知识和动手实践很好地结合了起来
集理论知识、开发实例、练习题、实例扩展、教学视频和电子教案于一体
利用Keil和Proteus等工具软件将每个实例的开发过程清晰而直观地呈现了出来
使用汇编和C两种编程语言进行实例开发，可以加深读者对单片机的理解

超值、大容量DVD光盘



DVD-ROM

- ◆ 本书实例源文件
- ◆ 21小时高清配套教学视频
- ◆ keil和Proteus软件的获取方式



清华大学出版社

简简单单学通 51单片机开发

王晋凯 等编著



清华大学出版社

内 容 简 介

学习单片机开发离不开实践。将理论和实例结合起来是最好的方式。本书结合实践,系统地介绍了 51 单片机开发的方方面面。书中的每章内容都围绕实例展开,这些例子大多采用汇编和 C 两种语言开发,并利用知名 EDA 仿真软件 Proteus 清晰地演示了每个实例的最终运行效果。这无疑可以大大提高读者的学习兴趣,并加深读者对单片机的理解。当然,每个实例的制作都需要有一定的理论去支撑,所以在实例设计之前会先讲解一些理论基础,并在实例完成之后对这些理论和开发经验进行总结。这种教学方式可以让读者将 51 单片机的理论知识和动手实践很好地结合起来,从而达到更好的学习效果。另外,配书 DVD 光盘中提供了作者专门录制的 21 小时高清配套教学视频和本书源文件,以方便读者高效、直观地学习。

本书共包含 12 章。其主要内容有:简单了解 51 单片机;从 I/O 端口开始学 51 单片机;51 单片机对中断的控制;51 单片机对时间的控制;数码管显示技术;通信利器——串口;数字电子时钟的设计;更先进的电子时钟;51 单片机外设扩展;点阵液晶 LCD 和矩阵键盘;计算器程序分析与设计;ZLG/GUI 在 51 单片机中的移植及运用。本书每章最后都提供了多个习题和实例扩展题,以帮助读者巩固和提高。

本书从 51 单片机基础知识开始讲解,逐步深入到较复杂的实例,特别适合 51 单片机初学者阅读。书中还介绍了很多同类图书中较少涉及的知识点和作者的实际项目经验,也适合有基础的读者和单片机爱好者进一步研读。本书注重实践教学,还非常适合大中专院校的相关专业作为教材使用。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

简简单单学通 51 单片机开发/王晋凯等编著. —北京:清华大学出版社, 2014

ISBN 978-7-302-36479-5

I. ①简… II. ①王… III. ①单片微型计算机—程序设计 IV. ①TP368.1

中国版本图书馆 CIP 数据核字(2014)第 099294 号

责任编辑:夏兆彦

封面设计:欧振旭

责任校对:胡伟民

责任印制:沈 露

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座

邮 编:100084

社总机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印装者:北京密云胶印厂

经 销:全国新华书店

开 本:185mm×260mm

印 张:31.5

字 数:790 千字

(附光盘 1 张)

版 次:2014 年 9 月第 1 版

印 次:2014 年 9 月第 1 次印刷

印 数:1~3000

定 价:69.80 元

产品编号:056599-01



前 言

单片机的出现可以追溯到 20 世纪 70 年代。在 1970~1974 年期间，诞生了第一代 4 位单片机。在 1978~1983 年期间，Intel 公司推出了 MCS-51 系列单片机（即 51 单片机），标志着单片机进入了 8 位时代。几十年来，以 8051 系列为代表的 8 位单片机在世界范围内长盛不衰，得到了不断发展和加强。51 系列单片机也以其强大的功能在工业控制领域得到了广泛的应用。近年来，以 ARM 为代表的 32 位控制器开始流行起来。所以有些人可能会觉得 51 单片机即将退出历史的舞台。然而，事实上 51 系列单片机在工控领域依然有着非常广泛的应用，而且也会在相当长的一段时间内继续发挥重要作用。

国内在较早的时期就引入了 51 单片机，而且相关院校也普遍开设了 51 单片机的课程，这使得单片机在国内的院校中广为流行。相应地，图书市场上也出版了大量的单片机教程。然而这些教程大多都是以生涩的理论知识讲解为主，缺乏实践教学。这给读者的学习造成了很多困惑：所学知识不能很好地应用于实际开发。时至今日，这种现象依然还存在。

笔者从大学期间就是一个电子技术的狂热爱好者，毕业后又从事了多年的电子产品的研发工作。作为一个过来人和从业者，想尽自己的绵薄之力，给广大的 51 单片机爱好者提供一些学习上的帮助，于是便有了这本《简简单单学通 51 单片机开发》的图书。学习单片机开发离不开实践，所以本书强调实践性教学，每个知识点都结合了相应的实例来讲解。这些实例都是通过仿真软件 Proteus 进行演示，效果很直观。而且大部分实例采用了汇编和 C 两种语言开发，一方面可以弥补读者在语言方面的不足，更为关键的是可以加深读者对单片机的理解。另外，为了帮助读者更加高效、直观地学习，笔者专门为本书详细录制了长达 21 小时的高清配套教学视频，以辅助读者学习。

相信在笔者的带领下，读者不但可以很好地掌握 51 单片机开发的知识，而且还可以感受到 51 单片机学习过程中的极大乐趣和做出实实在在产品的兴奋！

本书特色

1. 实例运用 Proteus 进行仿真

很多读者花费了大量的时间和金钱，去制作一块资源有限的调试开发板。而 Proteus 完全可以取代这个过程。Proteus 中包含了非常多的资源模块，供开发人员快捷、方便地搭建一个学习系统。本书便很好地利用了 Proteus，清晰地演示了每个实例的最终运行效果。

2. 重点介绍软件仿真

大型软件项目的调试和排错并非常规分析就能实现，而需要借助仿真调试。51 单片机开发环境 Keil 具有软件仿真的功能。本书将重点介绍利用软件仿真调试程序的方法。

3. 同步使用汇编语言和 C 语言

很多单片机书籍鼓励读者放弃汇编语言。这是非常不合理的。汇编语言在单片机的开发中是无法完全被 C 语言所取代的。一些要求实时性非常高的控制场合还得依赖于汇编代码而实现。此外,学习汇编语言有助于读者对单片机的内部资源有一个全面的认识,从而提高代码的编写效率。在本书中,大部分实例采用了汇编和 C 两种编程语言进行开发。

4. 使用大量的程序流程图

程序流程图可以将繁杂的语言叙述简化,让人看后一目了然,非常直观。这对于读者理解相关知识至关重要,是编程图书中不可或缺的讲解方式。本书讲解过程中使用了大量的程序流程图,几乎每个程序模块,甚至一些程序语句都使用了程序流程图。

5. 使用外部资源

51 单片机的内部资源并不是非常丰富,但可以通过使用外部扩展资源来对它的功能进行增强。本书中使用的外部扩展主要有两方面:一是硬件系统,包括功能芯片的使用和程序资源的扩充;二是软件系统,包括使用 C51 库函数和移植外部软件包。

6. 提供大量的习题和实例扩展

本书每章最后都提供了笔者精心编写的多个填空题、选择题、解答题和实例扩展题等。读者通过完成这些题目不但以检查自己对相关内容的掌握情况,而且可以巩固和提高所学的知识。这些题目的参考答案和解题思路需要读者自行按照本书封底的下载说明下载。

7. 提供 21 小时高清配套教学视频

笔者为本书录制了长达 21 小时的高清教学视频。这些视频不是泛泛而谈,而是非常详细和直观地再现了书中的大量操作。这对于实践性要求较高的单片机学习有很大的帮助。这些视频和书中涉及的所有实例源文件及电路图一并收录于配书 DVD 光盘中。另外,光盘中还收录了本书所用工具软件(Keil 编译器与 Proteus 演示版)的获取方式。

本书内容

第 1 章 简单了解 51 单片机

本章涵盖的内容有:51 单片机的内部资源,包括引脚结构、存储器空间分配、一个最小系统的开发等;开发环境 Keil 和仿真软件 Proteus 介绍;一个实例电路和程序的演示。

第 2 章 从 I/O 端口开始学习 51 单片机

本章结合一个流水灯实例,讲解了 51 单片机 I/O 端口的控制方法。其中需要重点掌握的内容有:汇编语言的数据传送指令和 C 语言赋值语句;几个特殊功能的寄存器;使用 Proteus 绘制电路图。

第 3 章 51 单片机对中断的控制

本章主要介绍了单片机的中断控制技术。需要掌握的内容有:汇编语言及 C 语言的循环语句和移位指令;通过一个外中断的实例改进流水灯控制方法;理解汇编语言和 C 语言

处理中断的区别。

第4章 51 单片机对时间的控制

本章涵盖两大知识点：一是单片机软件定时编程方法；二是单片机定时器，主要介绍单片机内部的两个定时器的 4 种工作方式，对于每种工作方式都提供了对应的程序实例。

第5章 数码管显示技术

本章主要介绍了数码管显示技术。需要掌握的内容有：数码管静态显示的方法；内部数据存储器的读取方法和内部程序存储器的使用方法；数码管动态显示方法。

第6章 通信利器——串口

本章主要介绍了串口通信技术。需要掌握的内容有：51 单片机串口通信的 4 种模式；串口双机通信的实现方法；串并互转的编程思想。

第7章 数字电子时钟的设计

本章介绍了利用单片机的内部资源完成一个电子时钟的设计，使用了包括定时器、串口、外中断等单片机的固有资源。这是一个综合型的实例，需要重点掌握分模块编程的思路。

第8章 更先进的电子时钟

本章主要介绍了运用外扩资源，配合单片机的控制而实现一个功能更加强大的电子时钟的实例。本章使用的外部设备有字符液晶 1602 和时钟芯片 DS1302 等。读者需要重点掌握 51 单片机外部资源和单片机接口的方法。

第9章 51 单片机外设扩展

51 单片机的内部资源并不丰富，所以进行外部资源的扩展就变得非常重要。本章为 51 单片机扩展了一片外部数据存储器和两组 8 位键盘。通过本对章内容的学习，应该掌握汇编语言和 C 语言进行外部扩展程序的实现方法。

第10章 点阵液晶 LCD 和矩阵键盘

本章主要介绍了单片机系统的两个常用电路模块：点阵液晶 LCD 和矩阵键盘。需要掌握的内容有：点阵液晶接口电路和驱动程序；矩阵键盘按键电路原理及扫描的方法。

第11章 计算器程序分析与设计

本章主要有两个任务：一是完成对 Proteus 自带计算器（汇编语言编写）的分析；二是对另外一个 Proteus 自带计算器（C 语言编写）进行改造。需要读者重点掌握的内容是 C51 库函数的调用。

第12章 ZLG/GUI 在 51 单片机中的移植及运用

ZLG/GUI 是广州周立功公司研发的轻型图形用户界面，具有较为强大的功能。本章重点内容是将该软件包移植到 51 单片机之中，并掌握综合运用该软件包提供的功能函数。通过本章内容的学习，读者不仅可以学习一个外部程序资源移植的范例，而且还可以更进一步理解使用外部函数简化单片机开发过程的思想。

附录

本书最后提供了 3 个附录，以方便读者学习时查阅相关资料。附录 A 给出了 51 单片

机的汇编指令集；附录 B 提供了 ASCII 码表；附录 C 给出了 C 语言运算符及其优先级。

本书读者对象

- ☐ 51 单片机初学者；
- ☐ 想全面学习 51 单片机开发技术的人员；
- ☐ 单片机产品开发人员；
- ☐ 单片机开发爱好者；
- ☐ 电子产品设计爱好者；
- ☐ 大中专院校的学生；
- ☐ 社会培训班学员。

本书作者

本书由王晋凯主笔编写。参与本书编写的人员还有吴万军、项延铁、谢邦铁、许黎民、薛在军、杨佩璐、杨习伟、于洪亮、张宝梅、张功勤、张建华、张建志、张敬东、张倩、张庆利、赵剑川、赵薇、郑强、周静、朱盛鹏、祝明慧、张晶晶。

读者阅读本书的过程中，有任何疑问都可以发邮件到 bookservice2008@163.com，我们会及时解决您的问题。

编者

目 录

第 1 章 简单了解 51 单片机 ( 教学视频: 77 分钟)	1
1.1 51 单片机硬件简述	1
1.1.1 51 单片机的逻辑结构及信号引脚	2
1.1.2 51 单片机的存储器	4
1.1.3 51 单片机的时钟电路和时序	10
1.1.4 51 单片机的复位电路	11
1.1.5 增强型 51 单片机 AT89C52 的特点介绍	12
1.2 51 单片机的程序设计语言和软件编译环境	13
1.2.1 来看一段汇编语言吧	13
1.2.2 看看更容易理解的 C 语言吧	13
1.2.3 两种程序编程语言的关系及相互的优缺点	14
1.2.4 51 单片机的编译开发环境介绍	15
1.3 51 单片机开发环境 Keil 的介绍	15
1.3.1 Keil 开发环境的介绍	15
1.3.2 项目建立的方法	16
1.4 51 单片机学习的利器——仿真软件 Proteus	19
1.4.1 Proteus 简介	20
1.4.2 用 Proteus 建立一个 51 单片机最小系统的电路图	25
1.5 通过一个小实例了解 51 单片机开发	27
1.5.1 在 Keil 中编译一段程序	27
1.5.2 在 Proteus 中演示程序	30
1.6 习题及操作练习	32
第 2 章 从 I/O 端口开始学 51 单片机 ( 教学视频: 53 分钟)	35
2.1 从一个漂亮的流水灯看 I/O 端口	35
2.1.1 演示流水灯实例	35
2.1.2 流水灯电路图的绘制	37
2.1.3 P1 端口介绍	43
2.1.4 流水灯程序设计流程图	45
2.2 用汇编语言实现流水灯	47
2.2.1 用汇编语言小试牛刀	47
2.2.2 测试流水灯汇编程序	48
2.2.3 51 单片机汇编语言指令介绍——数据传送指令	52
2.3 用 C 语言实现流水灯	53

2.3.1	根据流程图来编写程序	53
2.3.2	当代最优秀的程序设计语言——C 语言介绍	55
2.4	I/O 端口的输入控制	62
2.4.1	用开关控制流水灯	63
2.4.2	程序设计流程	64
2.4.3	汇编语言实现开关控制流水	64
2.4.4	汇编语言知识扩充——专用寄存器介绍	66
2.4.5	用 C 语言实现开关控制流水灯	69
2.4.6	C 语言知识扩展	71
2.5	习题和扩展练习	73
第 3 章	51 单片机对中断的控制 (教学视频: 89 分钟)	76
3.1	流水灯汇编程序的改进	76
3.1.1	流水灯程序改进的思路	76
3.1.2	改进后的汇编语言流水灯	78
3.1.3	软件仿真介绍	82
3.2	流水灯 C 语言的改进	89
3.2.1	新流水灯 C 语言设计思路	90
3.2.2	软件仿真 C 语言项目	93
3.3	外中断控制概述	98
3.3.1	51 单片机的中断源	98
3.3.2	外部中断的执行方式	99
3.3.3	外中断控制电路	100
3.4	汇编语言控制外中断	101
3.4.1	汇编知识介绍	101
3.4.2	编写中断控制汇编语言	104
3.4.3	在 Proteus 中仿真效果	106
3.5	用 C 语言完成中断控制	107
3.5.1	C 语言相关知识介绍	107
3.5.2	编写 C 语言外中断程序	110
3.5.3	软件仿真 C 语言外中断程序	112
3.6	习题和实例演练	117
第 4 章	51 单片机对时间的控制 (教学视频: 177 分钟)	119
4.1	51 单片机的软件定时	119
4.1.1	汇编语言实现精确定时	119
4.1.2	用 C 语言实现延时的方法	125
4.2	51 单片机定时器概述	131
4.2.1	51 单片机定时器的功能	131
4.2.2	定时器/计数器相关寄存器	131
4.3	定时工作方式 0	134
4.3.1	定时工作方式 0 概述	134

4.3.2	用定时器方式 0 做方波信号发生器	135
4.3.3	在工作方式 0 下, 利用查询方法实现方波信号发生器	137
4.3.4	汇编语言实现定时器在工作方式 0 的中断控制方法	142
4.3.5	C 语言实现定时器在工作方式 0 的中断控制	147
4.3.6	在 Proteus 中仿真中断实例	148
4.4	定时工作方式 1	149
4.4.1	定时器工作方式 1 概述	149
4.4.2	实例设计方案	150
4.4.3	汇编语言实现可编程定时	152
4.4.4	C 语言实现可编程定时	155
4.4.5	Proteus 仿真可编程定时	156
4.4.6	软件仿真可编程定时	157
4.5	定时工作方式 2	160
4.5.1	定时工作方式 2 概述	160
4.5.2	实例规划	161
4.5.3	用汇编语言实现目标	163
4.5.4	用 C 语言实现编程	164
4.5.5	在 Proteus 中进行仿真	166
4.6	定时工作方式 3	168
4.6.1	定时工作方式 3 概述	168
4.6.2	实例规划	169
4.6.3	用汇编语言实现编程	171
4.6.4	用 C 语言实现编程	173
4.6.5	在 Proteus 中进行仿真	174
4.7	习题和实例扩展	175
第 5 章	数码管显示技术 (教学视频: 92 分钟)	177
5.1	数码管静态显示	177
5.1.1	数码管介绍	177
5.1.2	汇编语言对数据存储器 (RAM) 的空间的攫取	180
5.1.3	利用 C 语言实现对数据存储器 (RAM) 的存取	189
5.1.4	汇编语言对程序存储器 (ROM) 的开拓	193
5.1.5	C 语言对程序存储器的读取	198
5.2	数码管动态显示	199
5.2.1	动态显示思路	200
5.2.2	用 Proteus 绘制数码管动态显示电路图	200
5.2.3	汇编语言实现动态显示	203
5.2.4	C 语言实现动态显示	206
5.3	汇编语言实现数码显示计数器	209
5.3.1	总体规划	210
5.3.2	汇编基础知识扩展——算数运算指令	212

5.3.3	分模块编写程序	213
5.3.4	在 Proteus 中仿真实例	219
5.3.5	在 Keil 中仿真	220
5.4	习题和实例扩展	223
第 6 章	通信利器——串口 (教学视频: 127 分钟)	225
6.1	一个实例来说明串口通讯	225
6.1.1	新式流水灯	225
6.1.2	串行通信基本概述	227
6.1.3	RS-232 总线标准	228
6.1.4	51 单片机串行口以及控制寄存器	229
6.1.5	51 单片机串行通信工作方式	230
6.1.6	串行工作方式 0 实例详述	232
6.2	双机通信	237
6.2.1	准备电路图	237
6.2.2	程序设计方案	239
6.2.3	汇编语言实现双机通信	241
6.2.4	C 语言实现双机通信	244
6.2.5	仿真观察双机通信	248
6.3	单片机和电脑通信	248
6.3.1	硬件电路的实现	248
6.3.2	程序设计方案	251
6.3.3	汇编语言编写实例程序	252
6.3.4	C 语言实现编程	255
6.3.5	实例在 Proteus 中的仿真	256
6.4	习题和实例扩展	260
第 7 章	数字电子时钟的设计 (教学视频: 157 分钟)	262
7.1	电子时钟总体规划	262
7.1.1	电子时钟整机电路图	262
7.1.2	数码管显示	263
7.1.3	串口修改数值	265
7.1.4	时钟工作单元	266
7.1.5	键盘控制	266
7.2	键盘分析	268
7.2.1	键值获取	269
7.2.2	根据键值执行相应程序	273
7.2.3	各按键子程序	274
7.3	电子时钟其他模块设计	277
7.3.1	程序预定义变量	277
7.3.2	主程序程序设计	279
7.3.3	显示子程序	281

7.3.4	定时器控制单元	283
7.3.5	串行修改单元	285
7.4	C 语言知识扩展	286
7.4.1	函数定义的形式	286
7.4.2	指针变量	288
7.4.3	加 1 子程序的分析	292
7.5	习题和实例扩展	293
第 8 章	更先进的电子时钟 (教学视频: 191 分钟)	295
8.1	字符和 ASCII 码	295
8.1.1	ASCII 码的概念	295
8.1.2	汇编语言中运用字符	296
8.1.3	C 语言中表达字符	296
8.1.4	串行口输出字符	298
8.2	字符型液晶 1602	302
8.2.1	字符液晶 1602 简介	302
8.2.2	在编程中学习液晶驱动	305
8.2.3	液晶显示其他接口函数	309
8.3	时钟模块研究	312
8.3.1	DS1302 介绍	312
8.3.2	DS1302 控制方式	313
8.3.3	时钟程序编写	315
8.4	文件的分割管理	325
8.4.1	C 项目的分割方法	325
8.4.2	汇编语言分割方式	328
8.5	习题和实例扩展	335
第 9 章	51 单片机外设扩展 (教学视频: 72 分钟)	337
9.1	外部数据存储器的扩展	337
9.1.1	数据存储芯片介绍	337
9.1.2	外扩 RAM 分析	340
9.1.3	外部设备数据传送指令	342
9.1.4	用实例验证对外部 RAM 的存取过程	345
9.2	LCD 的扩展	347
9.2.1	实现总线控制 LCD 的电路分析	347
9.2.2	汇编语言实现编程方式	349
9.2.3	C 语言实现编程方法	351
9.3	更多外设的扩展方法	352
9.3.1	实现多个外设扩展的电路实现方法	352
9.3.2	综合运用外设实例	356
9.3.3	汇编语言实现实例要求	358
9.3.4	C 语言实现实例目标	361

9.4 习题和实例扩展	364
第 10 章 点阵液晶 LCD 和矩阵键盘 (教学视频: 84 分钟)	366
10.1 点阵液晶 12864	366
10.1.1 12864 液晶概述	366
10.1.2 12864 直接控制方式底层驱动程序	370
10.1.3 12864 点阵液晶接口函数	373
10.1.4 实例显示成果	386
10.2 矩阵键盘	392
10.2.1 矩阵键盘的电路图及实现原理	392
10.2.2 矩阵键盘判键程序	394
10.2.3 矩阵键盘在 12864 液晶上的演示	397
10.3 习题和实例扩展	401
第 11 章 计算器程序分析与设计 (教学视频: 82 分钟)	403
11.1 汇编语言知识补充	403
11.1.1 汇编语言执行跳转指令	403
11.1.2 汇编语言伪指令扩充	404
11.2 C 语言库函数介绍	407
11.2.1 C51 库函数简介	407
11.2.2 典型库函数介绍	409
11.3 Proteus 自带计算器分析 (汇编语言编写)	413
11.3.1 实例电路分析	414
11.3.2 外设模块程序分析	416
11.3.3 程序分配布局	419
11.3.4 编译信息观察及程序总结	425
11.4 C 语言计算器分析和改进	425
11.4.1 Proteus 提供的计算器分析	426
11.4.2 改造 C51 计算器	431
11.4.3 项目设置和仿真效果	437
11.5 习题和实例扩展	438
第 12 章 ZLG/GUI 在 51 单片机中的移植及运用 (教学视频: 44 分钟)	440
12.1 C 语言基础知识补充	440
12.1.1 C51 结构体	440
12.1.2 C51 枚举类型	445
12.1.3 C51 共用体	446
12.1.4 类型定义符 typedef	448
12.2 ZLG/GUI 介绍	449
12.2.1 ZLG/GUI 的文件浏览	449
12.2.2 CONFIG.H 文件	454
12.2.3 ZLG/GUI 的配置	455
12.3 移植 ZLG/GUI	457

12.3.1	LCD 底层驱动	457
12.3.2	ZLG/GUI 的移植文件	459
12.3.3	底层程序编写	462
12.3.4	Keil 配置	465
12.4	ZLG/GUI 的应用举例	466
12.4.1	实现基本画图函数	466
12.4.2	字符绘制	468
12.4.3	汉字显示和图形显示	469
12.4.4	窗口显示	475
12.4.5	菜单功能和按钮	477
12.5	习题和实例扩展	480
附录 A	51 单片机汇编指令集	482
附录 B	ASCII 码表	486
附录 C	C 语言运算符及其优先级	488

第 1 章 简单了解 51 单片机

现在开始学习 51 单片机的征程。本章将要介绍 51 单片机的基础硬件知识，学习 51 单片机的开发环境，对 51 单片机的程序语言做一些简单的介绍。重点学习仿真软件 Proteus 的安装及使用方法。最后做一个小例子来了解 51 单片机项目的开发。本章的学习目的是让大家对 51 单片机感兴趣，尽可能简化学习的过程。

1.1 51 单片机硬件简述

51 单片机是对目前所有兼容 Intel 8031 指令系统的单片机的俗称。该系列单片机的始祖是 Intel 公司。80C51 单片机目前已成为 8 位单片机的经典代表，原因是 Intel 公司将 51 单片机技术公开，授权给其他公司生产。生产单片机的公司很多，常见的公司及他们生产的 51 系列单片机主要产品如下：

- ❑ Intel 公司的 80C31、80C51、87C51，80C32、80C52、87C52 系列等。
- ❑ ATMEL 公司的 89C51、89C52、89S52 系列等。
- ❑ Philips、华邦、Dallas、Siemens (Infineon) 等公司生产的产品也占有很大的市场。

随着 Flash ROM 技术的发展，80C51 单片机取得了更长足的进展，尤其是 ATMEL 公司的 AT89 系列，它广泛应用于工业测控系统之中。在我国，高校学生和电子发烧友等学习所用的单片机很多都是 ATMEL 公司的 AT89 系列，因为它除了性能稳定之外，程序烧录过程特别简单。所以在本书的学习中就以 ATMEL 公司的 AT89C52 单片机为例来介绍 51 单片机。

如图 1-1 所示为常见的 51 单片机的两种封装。图中左侧所示的封装较为常见，使用较为方便；图中右侧所示的封装因为体积较小，便于集成，所以在工业运用中较为常见。

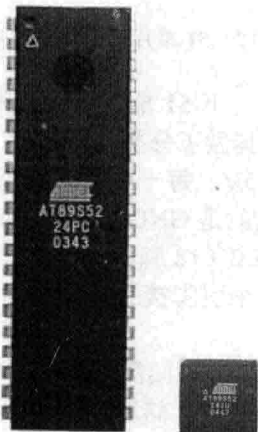


图 1-1 51 单片机的外形封装

1.1.1 51 单片机的逻辑结构及信号引脚

如图 1-2 所示为 51 单片机的内部逻辑结构图，够复杂吧，不过我们不必害怕。因为不需要重点来研究这个图，只需要对 51 单片机的内部逻辑结构有一个大概的理解即可。在这个图中，可以了解 51 单片机的内部结构和外设资源等信息。

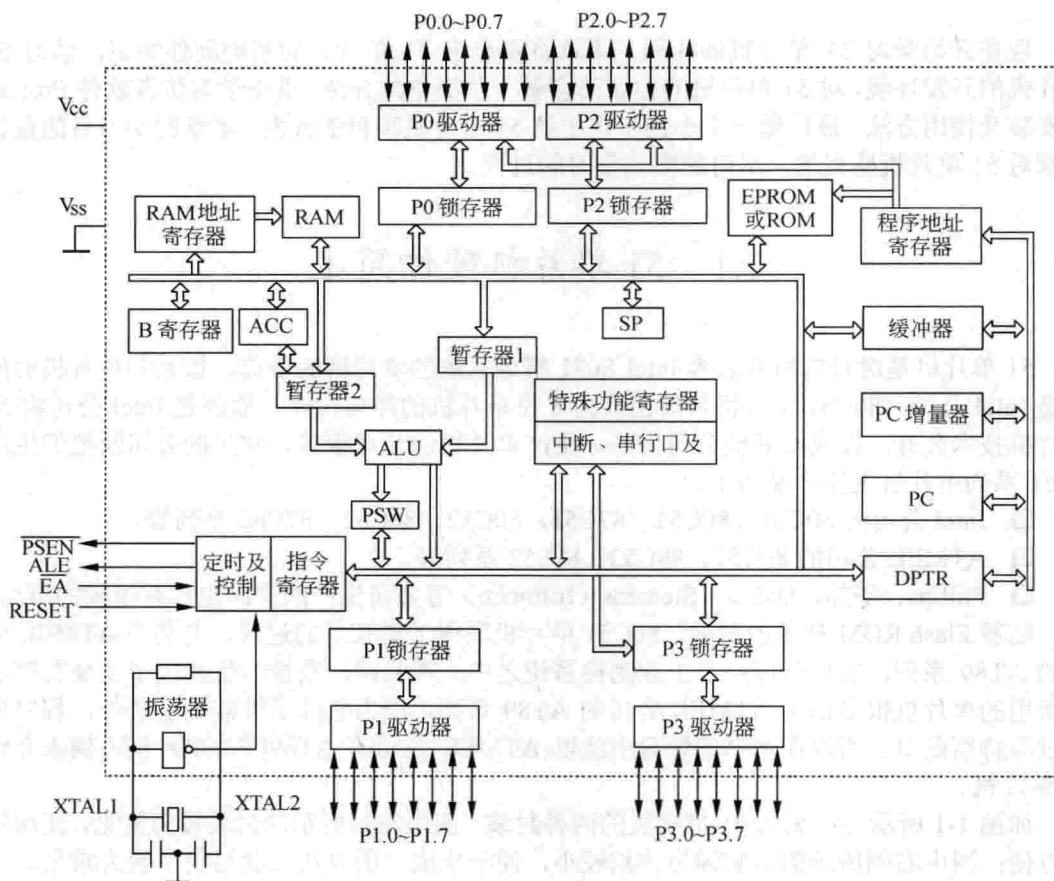


图 1-2 51 单片机的逻辑结构图

看完了内部结构，我们来认识一下 51 单片机各引脚的名称吧，如图 1-3 所示。

在图 1-3 中，将单片机的各引脚做了分类，以帮助大家理解。

- (1) VDD (40 脚)：接电源+5V，每一个芯片都需要电源来提供能量。
- (2) VSS (20 脚)：接地，也就是 GND，同理，每个芯片都需要接地来保证回路。
- (3) XTAL1 (19 脚) 和 XTAL2 (18 脚)：单片机需要时钟的支持，就得有时钟信号的产生或者输入。一般来说，这两个引脚接的是石英晶体，但也可以直接接入外部的时钟信号。
- (4) PSEN (29 脚)：51 单片机有时候需要扩展程序存储器（ROM），为了方便目标的选取，特意安排了此引脚。当对外部存储器进行读取数据时，此引脚为低电平。当涉及程序存储器的时候，我们再来深入地讨论。

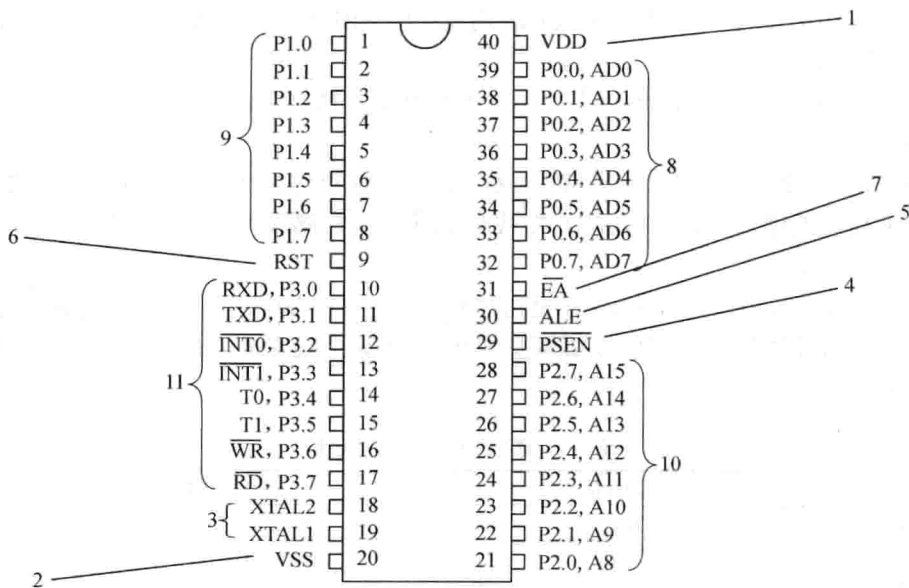


图 1-3 51 单片机的引脚分类

(5) ALE (30 脚): 这个引脚的作用是地址锁存信号的产生。现在提起这个概念可能比较抽象, 当学习到存储器扩展时, 会为大家来解答。在这里, 仅需了解 ALE 引脚输出的是六分之一晶体频率的正脉冲波。

(6) RST (9 脚): 复位信号输入端。就像我们的电脑需要开启或者重启一样, 51 单片机需要在次引脚输入两个机器周期以上的高电平来确保单片机进入正常的工作状态。有一套成熟的复位电路将在后面的章节中介绍。

(7) $\overline{\text{EA}}$ (7脚): 访问程序存储器控制信号。当 $\overline{\text{EA}}$ 为低电平时, 对程序存储器(ROM)的访问限定在外部程序存储器。当 $\overline{\text{EA}}$ 为高电平时, 对程序存储器的访问从内部程序存储器开始, 并延续至外部存储器。

(8) P0 (32~39 脚): 8 个双向 I/O 端口, 每一个端口都是完全相同的, 但是相互独立的。在实际应用过程中, P0 口在很多情况下都是用做单片机的地址和数据总线的。用做外部设备的扩展, 如存储器、I/O 设备等。P0 口和 ALE 引脚配合来完成外设数据的传输。不过随着串行传输技术的不断发展, P0 口将会被解放出来单独作为一个普通的 I/O 端口。在图中我们看到, P0 口有第二功能 (AD0~AD7), 为数据和信号共用端口。

P0 口在做普通 I/O 端口的时候,必须外接上拉电阻才会有高电平输出。所谓上拉电阻,就是此端口通过一个电阻来连接电源。如图 1-4 所示为上拉电阻和下拉电阻的连接方法。

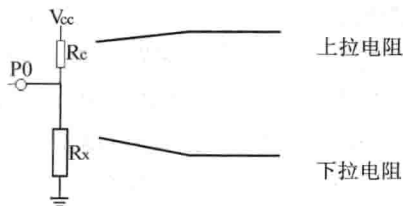


图 1-4 上拉电阻和下拉电阻