



普通高等教育“十二五”计算机类规划教材

单片机原理与项目实践

——基于 C 语言

◎ 钱宗峰 刘培国 于飞 主编

Danpianji Yuanli Yu Xiangmu Shijian
—— *Jiyu C Yuyan*



机械工业出版社
CHINA MACHINE PRESS

普通高等教育“十二五”计算机类规划教材

单片机原理与项目实践 ——基于 C 语言

主 编 钱宗峰 刘培国 于 飞
参 编 周希伟 王丽慧 王艳姝
主 审 邹志刚



机械工业出版社

本教材共分 10 章, 主要介绍了 51 单片机的基础知识, 常用软件 Proteus、Keil、STC_ISP 的使用, 单片机 C 程序软件架构设计及程序优化, 多功能显示仪项目的开发过程与源程序解读以及 LCD12864、AT24C02、DS1302、DS18B20、GPS 接收器的相关知识。

教材内容难易程度适中, 编排合理, 适合相关专业本科院校、高职高专院校作为教材使用, 也可作为单片机自学者的入门用书。书中所有实例均配有电路图和程序分析, 程序代码的编写符号编写规范并经过实际验证, 涉及的多功能显示仪源程序请到机械工业出版社教育服务网 <http://www.cmpedu.com> 下载。为方便教师教学, 本书配有免费教学课件, 欢迎选用本书作为教材的教师登录 <http://www.cmpedu.com> 下载, 或发邮件到 llm7785@sina.com 信箱索取。

图书在版编目 (CIP) 数据

单片机原理与项目实践: 基于 C 语言/钱宗峰, 刘培国, 于飞主编.
—北京: 机械工业出版社, 2014. 12
普通高等教育“十二五”计算机类规划教材
ISBN 978 - 7 - 111 - 48543 - 8

I. ①单… II. ①钱…②刘…③于… III. ①单片微型计算机
—C 语言—程序设计—高等学校—教材 IV. ①TP368.1②TP312

中国版本图书馆 CIP 数据核字 (2014) 第 266071 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

策划编辑: 刘丽敏 责任编辑: 刘丽敏 王 康

版式设计: 霍永明 责任校对: 陈秀丽

责任印制: 刘 岚

北京京丰印刷厂印刷

2015 年 1 月第 1 版·第 1 次印刷

184mm×260mm·12.75 印张·304 千字

标准书号: ISBN 978 - 7 - 111 - 48543 - 8

定价: 29.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

电话服务

网络服务

社服务中心: (010)88361066

教材网: <http://www.cmpedu.com>

销 售 一 部: (010)68326294

机工官网: <http://www.cmpbook.com>

销 售 二 部: (010)88379649

机工官博: <http://weibo.com/cmp1952>

读者购书热线: (010)88379203

封面无防伪标均为盗版

前 言

目前,单片机作为嵌入式系统的入门课程在各工科院校中广泛开设,长期以来,该课程存在原理难以理解、开发技能难以提升的问题。作者在多年的单片机教学实践中采用项目驱动式的教学方式,取得了良好的教学效果。本教材结合作者多年的教学成果,采用新思路、新方法编写而成,更加适合单片机的学习。

本教材特色如下:

1. 注重综合知识应用,提升学生开发技能

在本教材的最后一章,结合“多功能显示仪”项目,讲解了键盘、液晶显示、I²C 总线、SPI 总线、1-Wire 总线的基础知识与应用,介绍了 12864 液晶、DS1302、DS18B20、GPS 接收模块、AT24C02 等芯片的知识与应用,并系统讲解了项目的研发过程与源程序。

2. 突出了单片机 C 程序软件架构设计及程序优化

在市面上的教材中,很少讲到 C 程序软件架构设计及程序优化,而很多学生在程序编写与项目开发过程中,往往会遇到这方面的困扰,所以本教材特别强化了这部分内容,重点讲解了模块划分、头文件的建立、宏定义的应用、常用程序优化方法及一些重要关键字的使用。

3. 针对技术需求现状,注重学生实践能力的培养

一是 C51 作为主干编程语言,要求熟练掌握,灵活应用;汇编语言作为辅助编程语言,只要求一般了解,能读懂源程序即可。二是突出热点知识,针对学生毕业后的工作需求,把讲授重点放在定时器、计数器、中断、串口、键盘、液晶显示、I²C 总线、SPI 总线、1-Wire 总线等知识及应用上,突出了实践性。三是剔除了一些目前工业界几乎不用的技术,如 8255 芯片的知识。

4. 把实践内容提前,激发学生学习兴趣

在第 1 章中放入了程序实例,使学生提前有了感性认识,带着问题进行学习,更利于对后续理论知识的理解,同时可以激发学生学习的兴趣。

5. 注重程序的规范性

本教材所撰写的程序均严格遵照嵌入式 C 语言编程的常用规范,能引导学生养成良好的编程习惯。

本教材第 1、2、8 章由于飞编写,第 3 章由王丽慧编写,第 4 章由王艳姝编写,第 5、6 章由钱宗峰编写,第 7 章由周希伟编写,第 9、10 章由刘培国编写,钱宗峰、邹志刚负责全书的统编定稿与审阅工作。

本教材第 10 章的多功能显示仪源程序请到机械工业出版社教育服务网 <http://www.cmpedu.com> 下载。

由于作者水平有限,本教材难免有一些不足之处,殷切地期望读者批评指正(请发邮件至 liupg300@163.com)。

编 者

目 录

前言

第1章 单片机基础知识 1

1.1 单片机概述 1

1.2 计算机中的数制与编码 2

1.2.1 数制 2

1.2.2 计算机中数的表示 4

1.2.3 字符编码 5

1.3 单片机的电平特性 6

1.4 MCS-51 单片机的结构 7

1.4.1 MCS-51 单片机的内部结构 8

1.4.2 MCS-51 的外部引脚及功能 10

1.5 MCS-51 的存储器 11

1.5.1 存储器的结构 11

1.5.2 程序存储器 12

1.5.3 数据存储器 12

1.6 MCS-51 单片机的时钟、时序与

复位 14

1.6.1 时钟电路 14

1.6.2 单片机的时序单位 15

1.6.3 复位电路 15

1.7 输入/输出端口结构 16

1.8 初步了解单片机应用 18

习题 21

第2章 单片机常用开发软件 23

2.1 Proteus 23

2.1.1 ISIS 模块应用举例 23

2.1.2 ARES 模块应用举例 26

2.2 Keil C51 软件介绍 31

2.2.1 新建一个工程 31

2.2.2 Keil 的调试功能 35

2.3 STC 单片机烧写程序示例 40

习题 44

第3章 MCS-51 单片机指令系统与汇编语言 45

3.1 MCS-51 指令系统概述 45

3.1.1 指令格式 45

3.1.2 操作数的简记符号 45

3.1.3 寻址方式 46

3.2 MCS-51 指令系统简介 47

3.2.1 数据传送指令 47

3.2.2 算术运算指令 48

3.2.3 逻辑操作及移位指令 49

3.2.4 控制转移指令 50

3.3 汇编语言的伪指令 51

3.4 STARTUP.A51 文件 52

3.5 小结 53

习题 53

第4章 单片机的 C 语言编程 55

4.1 C51 程序结构 55

4.2 C51 的数据类型 56

4.2.1 标识符与关键字 56

4.2.2 C51 的变量 58

4.2.3 C51 的指针 61

4.3 C51 与汇编语言的混合编程 61

4.3.1 C 语言程序和汇编语言程序参数的传递 62

4.3.2 在 C51 中嵌入汇编代码 62

4.4 C51 对存储器和外部 I/O 的绝对地址访问 63

4.5 C51 的运算符 64

4.6 C51 初步应用 65

4.7 小结 68

习题 68

第5章 单片机的中断系统 69

5.1 中断概述 69

5.2 中断控制系统 70

5.2.1 中断源 70

5.2.2 中断控制寄存器 70

5.3 中断处理过程 73

5.4 中断的编程和应用举例 74

5.5 小结 78

习题 79

第6章 定时器/计数器 80

6.1 定时器/计数器的结构和工作原理 80

6.1.1 定时器/计数器的基本结构	80	9.2 程序优化	128
6.1.2 定时器/计数器的工作原理	80	9.2.1 单片机中的 C 语言预处理命令 的应用	128
6.2 定时器/计数器的控制	81	9.2.2 常用程序优化方法	129
6.3 定时器/计数器的工作方式	82	9.3 重要的关键字	133
6.4 定时器/计数器 T2 介绍	85	9.3.1 static	133
6.4.1 定时器/计数器 T2 的寄存器	85	9.3.2 volatile	134
6.4.2 T2 的工作方式	86	9.3.3 extern	136
6.5 定时器/计数器的初始化	88	9.3.4 const	136
6.6 定时器/计数器的应用例题	89	第 10 章 多功能显示仪项目开发	
6.7 小结	96	实践	137
习题	96	10.1 多功能显示仪项目构建	137
第 7 章 串行通信接口	98	10.1.1 项目简介	137
7.1 串行通信的分类	98	10.1.2 硬件设计	137
7.2 串行接口寄存器	99	10.2 矩阵键盘	141
7.2.1 串行口内部结构	99	10.2.1 线反转法与扫描法	141
7.2.2 串行口控制寄存器	100	10.2.2 综合显示仪键盘程序	143
7.3 串行口的工作方式	102	10.3 SG12864-12 带汉字库液晶	147
7.4 RS-232 标准及其应用	102	10.3.1 SG12864-12 带汉字库液晶基础 知识	147
7.5 单片机串行通信电路	104	10.3.2 多功能显示仪液晶显示程序	152
7.6 串行口应用实例	105	10.4 I ² C 总线 AT24C02 设计	158
7.7 小结	113	10.4.1 I ² C 总线	158
习题	114	10.4.2 AT24C02 基础知识	161
第 8 章 A-D 和 D-A 转换器接口	115	10.4.3 多功能显示仪 AT24C02 程序	163
8.1 D-A 转换	115	10.5 SPI 总线 DS1302 实时时钟设计	167
8.1.1 D-A 转换器简介	115	10.5.1 SPI 总线基础知识	167
8.1.2 DAC0832 的工作原理	115	10.5.2 DS1302 实时时钟	170
8.1.3 DAC 的主要性能指标	117	10.5.3 多功能显示仪 DS1302 程序	172
8.1.4 DAC0832 与 51 单片机的接口	117	10.6 1-Wire 单总线 DS18B20 测温 设计	180
8.2 A-D 转换与 ADC0809 应用	119	10.6.1 1-Wire 单总线概述	180
8.2.1 逐次逼近式模数转换器的工作 原理	120	10.6.2 DS18B20 芯片介绍	181
8.2.2 A-D 转换器的主要性能指标	120	10.6.3 多功能显示仪 DS18B20 程序	184
8.2.3 DAC0809 与 51 单片机的 接口	121	10.7 GPS 接收模块设计	188
8.3 小结	123	10.7.1 基础知识	188
习题	123	10.7.2 多功能显示仪 GPS 程序	190
第 9 章 单片机 C 程序软件架构设计与 程序优化	124	10.8 小结	194
9.1 模块化的程序设计	124	参考文献	195
9.1.1 模块化概念	124		
9.1.2 头文件的建立	126		

第1章 单片机基础知识

1.1 单片机概述

1970年微型计算机研制成功后,随后就出现了单片机。美国 Intel 公司在 1971 年推出了 4 位单片机 4004,1972 年推出了雏形 8 位单片机 8008。特别是在 1976 年推出 MCS-48 单片机以后的近 40 年中,单片机的发展和其相关的技术经历了数次的更新换代。其发展速度大约每三四年要更新一代、集成度增加一倍、性能翻一番。尽管单片机出现的历史并不长,但以 8 位单片机的推出为起点,单片机的发展大致可分为 4 个阶段。

第一阶段(1976~1978 年):初级单片机阶段。以 Intel 公司 MCS-48 为代表。这个系列的单片机内集成有 8 位 CPU、I/O 接口、13 位定时器/计数器,寻址范围不大于 4KB,简单的中断功能,无串行接口。

第二阶段(1978~1982 年):单片机完善阶段。在这一阶段推出的单片机其功能有较大的加强,能够应用于更多的场合。这个阶段的单片机普遍带有串行 I/O 口、有多级中断处理系统、16 位定时器/计数器,片内集成的 RAM、ROM 容量加大,寻址范围可达 64KB。一些单片机片内还集成了 A-D 转换接口。这类单片机的典型代表有 Intel 公司的 MCS-51、Motorola 公司的 6801 和 Zilog 公司的 Z8 等。

第三阶段(1982~1992 年):8 位单片机巩固发展及 16 位高级单片机发展阶段。在此阶段,尽管 8 位单片机的应用已广泛普及,但为了更好地满足测控系统嵌入式应用的要求,单片机集成的外围接口电路有了更大的扩充。这个阶段单片机的代表为 8051 系列。许多半导体公司和生产厂以 MCS-51 的 8051 为内核,推出了满足各种嵌入式应用的多种类型和型号的单片机。其主要技术发展有:

(1) 外围功能集成。满足模拟量直接输入的 ADC 接口;满足伺服驱动输出的 PWM;保证程序可靠运行的程序监控定时器(WDT,俗称看门狗电路)。

(2) 出现了为满足串行外围扩展要求的串行扩展总线和接口,如 SPI、I²C Bus、单总线(1-Wire)等。

(3) 出现了为满足分布式系统、突出控制功能的现场总线接口,如 CAN Bus 等。

(4) 在程序存储器方面广泛使用了片内程序存储器技术,出现了片内集成 EPROM、EEPROM、FlashROM 以及 MaskROM、OTPROM 等各种类型的单片机,以满足不同产品的开发和生产的需要,也为最终取消外部程序存储器扩展奠定了良好的基础。与此同时,一些公司面向更高层次的应用,发展推出了 16 位的单片机,典型代表有 Intel 公司的 MCS-96 系列的单片机。

第四阶段(1993 年至今):百花齐放阶段。现阶段单片机发展的显著特点是百花齐放、技术创新,以满足日益增长的广泛需求。其主要方面有:

(1) 单片嵌入式系统的应用是面向最底层的电子技术应用,从简单的玩具、小家电到



复杂的工业控制系统、智能仪表、电器控制以及发展到机器人、个人通信信息终端、机顶盒等。因此，面对不同的应用对象，不断推出适合不同领域要求的，从简易性能到丰富功能的单片机系列。

(2) 大力发展专用型单片机。早期的单片机是以通用型为主的。由于单片机设计、生产技术的提高、周期缩短、成本下降，许多特定类型电子产品，如家电类产品的巨大的市场需求能力，推动了专用单片机的发展。在这类产品中采用专用单片机具有成本低、资源有效利用率高、系统外围电路少、可靠性高的优点。因此专用单片机也是单片机发展的一个主要方向。

(3) 致力于提高单片机的综合品质。采用更先进的技术来提高单片机的综合品质，如提高 I/O 接口的驱动能力、增加抗静电和抗干扰措施、宽（低）电压低功耗等。

随着半导体集成工艺的不断发展，单片机的集成度将更高、体积将更小、功能将更强。在单片机家族中，80C51 系列是其中的佼佼者，加之 Intel 公司将其 MCS-51 系列中的 80C51 内核使用权以专利互换或出售形式转让给全世界许多著名 IC 制造厂商，如 Philips、NEC、Atmel、AMD、华邦等，这些公司都在保持与 80C51 单片机兼容的基础上改善了 80C51 的许多特性。这样，80C51 就变成有众多制造厂商支持的、发展出上百品种的大家族，现统称为 80C51 系列。

1.2 计算机中的数制与编码

1.2.1 数制

1. 进位计数制

数制也称计数制，是指用一组固定的符号和统一的规则来表示数值的方法。按进位的原则进行计数的方法，称为进位计数制。比如，在十进位计数制中，是按照“逢 10 进 1”的原则进行计数的。

常用进位计数制：

十进制（Decimal Notation）；

二进制（Binary Notation）；

八进制（Octal Notation）；

十六进制（Hexadecimal Notation）。

2. 进位计数制的基数与位权

“基数”和“位权”是进位计数制的两个要素。

(1) 基数：所谓基数，就是进位计数制的每位数上可能有的数码的个数。例如，十进制数每位上的数码，有“0”、“1”、“2”、…、“9”这 10 个数码，所以基数为 10。

(2) 位权：所谓位权，是指一个数值的每一位上的数字的权值的大小。例如，十进制数 4567 从低位到高位位权分别为 10^0 、 10^1 、 10^2 、 10^3 。例如 4567 可展开为

$$4567 = 4 \times 10^3 + 5 \times 10^2 + 6 \times 10^1 + 7 \times 10^0$$

(3) 数的位权表示：任何一种数制的数都可以表示成按位权展开的多项式之和。

比如：十进制数 435.05 可表示为

$$435.05 = 4 \times 10^2 + 3 \times 10^1 + 5 \times 10^0 + 0 \times 10^{-1} + 5 \times 10^{-2}$$

位权表示法的特点是：每一项 = 某位上的数字 × 基数的若干幂次，而幂次的大小由该数字所在的位置决定。

3. 二进制数

由于二进制运算简单、电路简单可靠、逻辑性强，所以计算机中常采用二进制。

(1) 定义：按“逢2进1”的原则进行计数，称为二进制数，即每位上计满2时向高位进1。

(2) 特点：每个数的数位上只能是0、1两个数字；二进制数中最大数字是1，最小数字是0；基数为2。比如：10011010与00101011是两个二进制数。

(3) 二进制数的位权表示：

$$(1101.101)_2 = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3}$$

(4) 二进制数的运算规则

加法运算

$$1) 0 + 0 = 0$$

$$2) 0 + 1 = 1 + 0 = 1$$

$$3) 1 + 1 = 10$$

乘法运算

$$1) 0 \times 0 = 0$$

$$2) 0 \times 1 = 1 \times 0 = 0$$

$$3) 1 \times 1 = 1$$

4. 八进制数

(1) 定义：按“逢8进1”的原则进行计数，称为八进制数，即每位上计满8时向高位进1。

(2) 特点：每个数的数位上只能是0、1、2、3、4、5、6、7这8个数字；八进制数中最大数字是7，最小数字是0；基数为8。比如：(1347)₈与(62435)₈是两个八进制数。

(3) 八进制数的位权表示：

$$(107.13)_8 = 1 \times 8^2 + 0 \times 8^1 + 7 \times 8^0 + 1 \times 8^{-1} + 3 \times 8^{-2}$$

5. 十六进制数

(1) 定义：按“逢16进1”的原则进行计数，称为十六进制数，即每位上计满16时向高位进1。

(2) 特点：每个数的数位上只能是0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F这16个数码；十六进制数中最大数字是F，即15，最小数字是0；基数为16。比如：(109)₁₆与(2FDE)₁₆是两个十六进制数。

(3) 十六进制数的位权表示：

$$(109.13)_{16} = 1 \times 16^2 + 0 \times 16^1 + 9 \times 16^0 + 1 \times 16^{-1} + 3 \times 16^{-2}$$

$$(2FDE)_{16} = 2 \times 16^3 + 15 \times 16^2 + 13 \times 16^1 + 14 \times 16^0$$

6. 常用计数制间的对应关系如表1-1所示。

表1-1 常用计数制间的对应关系

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	2	10	2	2
1	1	1	1	4	100	4	4



(续)

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
8	1000	10	8	15	1111	17	F
10	1010	12	A	16	10000	20	10

7. 数制间的转换

(1) 十进制数和非十进制数之间的转换

1) 十进制整数转换成非十进制整数

十进制整数化为非十进制整数采用“余数法”，即除基数取余数。把十进制整数逐次用任意十进制数的基数去除，一直到商是 0 为止，然后将所得到的余数由下而上排列即可。

2) 十进制小数转换成非十进制小数

十进制小数转换成非十进制小数采用“进位法”，即乘基数取整数。

把十进制小数不断的用其他进制的基数去乘，直到小数的当前值等于 0 或满足所要求的精度为止，最后所得到的积的整数部分由上而下排列即为所求。

3) 非十进制数转换成十进制数

非十进制数转换成十进制数采用“位权法”，即把各非十进制数按位权展开，然后求和。

(2) 二进制、八进制、十进制数之间的转换

1) 二进制数与八进制数之间的转换

a. 把二进制数转换为八进制数时，按“3 位并 1 位”的方法进行。

以小数点为界，将整数部分从右向左每 3 位一组，最高位不足 3 位时，添 0 补足 3 位；小数部分从左向右，每 3 位一组，最低有效位不足 3 位时，添 0 补足 3 位。然后，将各组的 3 位二进制数按权展开后相加，得到 1 位八进制数。

b. 将八进制数转换成二进制数时，采用“1 位拆 3 位”的方法进行。即把八进制数每位上的数用相应的 3 位二进制数表示。

2) 二进制数与十六进制数之间的转换

a. 把二进制数转换为十六进制数时，按“4 位并 1 位”的方法进行。

以小数点为界，将整数部分从右向左每 4 位一组，最高位不足 4 位时，添 0 补足 4 位；小数部分从左向右，每 4 位一组，最低有效位不足 4 位时，添 0 补足 4 位。然后，将各组的 4 位二进制数按权展开后相加，得到 1 位十六进制数。

b. 将十六进制数转换成二进制数时，采用“1 位拆 4 位”的方法进行。即把十六进制数每位上的数用相应的 4 位二进制数表示。

1.2.2 计算机中数的表示

1. 机器数与真值

(1) 机器数

数学中正数与负数是用该数的绝对值，加上正、负符号来表示。由于计算机中无论是数值还是数的符号，都只能用 0 和 1 来表示。所以计算机中，为了表示正、负数，把一个数的最高位作为符号位：0 表示正数，1 表示负数。比如，如果用 8 个二进制位表示一个十进制数，则正的 36 和负的 36 可表示为

+36 - - - - > 00100100

-36 - - - - > 10100100

这种连同符号位一起数字化了的数称为机器数。

(2) 真值

由机器数所表示的实际值称为真值。

比如：机器数 00101011 的真值为十进制的 +43 或二进制的 +0101011。

机器数 10101011 的真值为十进制的 -43 或二进制的 -0101011。

2. 机器数的表示方法

(1) 原码

正数的符号位用 0 表示，负数的符号位用 1 表示，数值部分用二进制形式表示，称为该数的原码。比如：

$X = +81$ (X) 原 = 01010001

$Y = -81$ (Y) 原 = 11010001

(2) 反码

正数的反码和原码相同，负数的反码是对该数的原码除符号位外各位取反，即“0”变“1”，“1”变“0”。

例如： $X = +81$ ， $Y = -81$

(X) 原 = 01010001 (X) 反 = 01010001

(Y) 原 = 11010001 (Y) 反 = 10101110

(3) 补码

正数的补码与原码相同，负数的补码是对该数的原码除符号位外各位取反，然后加 1，即反码加 1。

比如： $X = +81$ ， $Y = -81$

(X) 原 = (X) 反 = (X) 补 = 01010001

(Y) 原 = 11010001

(Y) 反 = 10101110

(Y) 补 = 10101111

3. 计算机中，加减法基本上都采用补码进行运算，并且加减法运算都可以用加法来实现。

比如：计算十进制数减法 $36 - 45$ ，可写成： $36 + (-45)$ 。

(36) 补 = 00100100

(-45) 补 = 11010011

$00100100 + 11010011 = 11110111 = (-9)$ 补

结果正确。

1.2.3 字符编码

所谓字符编码就是规定用怎样的二进制编码来表示文字和符号。它主要有以下几种：

1. BCD 码 (二-十进制码)

把十进制数的每一位分别写成二进制数形式的编码，称为二-十进制编码或 BCD 编码。



BCD 编码方法很多,但常用的是 8421 编码,它采用 4 位二进制数表示 1 位十进制数,即每一位十进制数用 4 位二进制数表示。这 4 位二进制数各位权由高到低分别是 2^3 、 2^2 、 2^1 、 2^0 ,即 8、4、2、1。这种编码最自然,最简单,且书写方便、直观、易于识别。

比如:十进制数 1998 的 8421 码为 0001 1001 1001 1000

十进制: 1 9 9 8

8421 码: 0 0 0 1 1 0 0 1 1 0 0 1 1 0 0 0

位权: 8 4 2 1 8 4 2 1 8 4 2 1 8 4 2 1

2. ASCII 码

ASCII 码是计算机系统中使用得最广泛的一种编码。ASCII 码虽然是美国国家标准,但它已被国际标准化组织 (ISO) 认定为国际标准。ASCII 码已为世界公认,并在世界范围内通用。ASCII 码有 7 位版本和 8 位版本两种。国际上通用的是 7 位版本。7 位版本的 ASCII 码有 128 个元素,其中通用控制字符 34 个,阿拉伯数字 10 个,大小写英文字母 52 个,各种标点符号和运算符号 32 个。

比如:“A”的 ASCII 码值为 1000001,即十进制的 65;“a”的 ASCII 码值为 1100001,即十进制的 97;“0”的 ASCII 码值为 0110000,即十进制的 48。

3. 汉字编码

我国用户在使用计算机进行信息处理时,都要用到汉字的输入、输出以及汉字的处理。这就需要对汉字进行编码。通常汉字有两种编码:国标码和机内码。

(1) 国标码

计算机处理汉字所用的编码标准是我国于 1980 年颁布的国家标准 (GB 2312—1980),是国家规定的用于汉字编码的依据,简称国标码。

国标码规定:用两个字节表示一个汉字字符。在国标码中共收录汉字和图形符号 7445 个。

国标码本身也是一种汉字输入码。通常称为区位输入法。

(2) 机内码

机内码是指在计算机中表示一个汉字的编码。机内码是一种机器内部的编码,其主要作用是作为汉字信息交换码使用:将不同系统使用的不同编码统一转换成机内码,使不同的系统之间的汉字信息进行交流。

正是由于机内码的存在,输入汉字时就允许用户根据自己的习惯使用不同的汉字输入法,比如五笔字型、自然码、智能拼音等,进入系统后再统一转换成机内码存储。

1.3 单片机的电平特性

常用的逻辑电平有 TTL、CMOS、LVTTTL、ECL、PECL、GTL、RS-232、RS-422、RS-485、LVDS 等。其中 TTL 和 CMOS 的逻辑电平按典型电压可分为四类:5V 系列 (5V TTL 和 5V CMOS)、3.3V 系列、2.5V 系列和 1.8V 系列。

5V TTL 和 5V CMOS 是通用的逻辑电平。3.3V 及以下的逻辑电平被称为低电压逻辑电平,常用的为 LVTTTL 电平。低电压逻辑电平还有 2.5V 和 1.8V 两种。ECL/PECL 和 LVDS 是差分输入/输出。RS-422/485 和 RS-232 是串口的接口标准,RS-422/485 是差分输入/输

出, RS-232 是单端输入/输出。

TTL 电平信号用的最多, 这是因为数据表示通常采用二进制, $+5V$ 等价于逻辑 1, $0V$ 等价于逻辑 0, 这被称为 TTL (晶体管-晶体管逻辑电平) 信号系统, 这是计算机处理器控制的设备内部各部分之间通信的标准技术。TTL 电平信号对于计算机处理器控制的设备内部的数据传输是很理想的, 首先计算机处理器控制的设备内部的数据传输对于电源的要求不高, 热损耗也较低; 另外 TTL 电平信号直接与集成电路连接而不需要价格昂贵的电路驱动器以及接收器电路; 再者, 计算机处理器控制的设备内部的数据传输是在高速下进行的, 而 TTL 接口的操作恰能满足这一要求。TTL 型通信大多数情况下是采用并行数据传输方式, 而并行数据传输对于超过 10ft 的距离就不适合了。这是由于可靠性和成本两方面的原因。因为在并行接口中存在着偏相和不对称问题, 这些问题对可靠性均有影响; 另外, 并行数据传输、电缆以及连接器的费用比起串行方式来也要高一些。

CMOS 电平 V_{CC} 可达 12V, CMOS 电路输出高电平约为 $0.9V_{CC}$, 而输出低电平约为 $0.1V_{CC}$ 。CMOS 电路中不使用的输入端不能悬空, 否则会造成逻辑混乱。另外, CMOS 集成电路电源电压可以在较大范围内变化, 因而对电源的要求不像 TTL 集成电路那样严格。

CMOS 电平能驱动 TTL 电平, 但 TTL 电平不能驱动 CMOS 电平, 需要加上拉电阻。常用的逻辑芯片的特点如表 1-2 所示。

表 1-2 常用的逻辑芯片的特点

系列名称	芯片类型	输入	输出	系列名称	芯片类型	输入	输出
74LS 系列	TTL	TTL	TTL	74HCT 系列	CMOS	TTL	CMOS
74HC 系列	CMOS	CMOS	CMOS	CD4000 系列	CMOS	CMOS	CMOS

TTL 与 CMOS 差异主要有以下几点:

- (1) CMOS 是由场效应晶体管构成, TTL 为由双极型晶体管构成。
- (2) CMOS 的逻辑电平范围比较大 ($5 \sim 15V$), TTL 只能在 $5V$ 下工作。
- (3) CMOS 的高低电平之间相差比较大、抗干扰性强, TTL 则相差小, 抗干扰能力差。
- (4) CMOS 功耗很小, TTL 功耗较大 ($1 \sim 5mA/\text{门}$)。
- (5) CMOS 的工作频率较 TTL 略低, 但是高速 CMOS 速度与 TTL 差不多相当。

TTL 的电源工作电压是 $5V$, 所以 TTL 的电平是根据电源电压 $5V$ 来定的。CMOS 的电源工作电压是 $3 \sim 18V$, CMOS 的电源工作电压范围宽, 如果 CMOS 的电源工作电压是 $12V$, 那么这个 CMOS 的输入输出电平电压要适合 $12V$ 的输入输出要求。即 CMOS 的电平, 要看用的电源工作电压是多少, $3 \sim 18V$ 都在 CMOS 的电源工作电压范围内, 具体数值要看加在 CMOS 芯片上的电源工作电压是多少。

1.4 MCS-51 单片机的结构

MCS-51 系列单片机分为 51 和 52 两个子系列, 包括 80C51、87C51、80C52、87C52 等典型产品型号。它们的结构基本相同, 主要差别在于片内的资源配置有所不同。其中 52 子系列只是在存储器容量、计数器和中断源数量方面高于 51 子系列, 基本控制原理是一样的, 采用 80C51 更具有代表性。



1.4.1 MCS-51 单片机的内部结构

MCS-51 单片机的内部资源都挂接在单片机内部总线上, 通过内部总线传送数据和指令。其基本结构如图 1-1 所示。

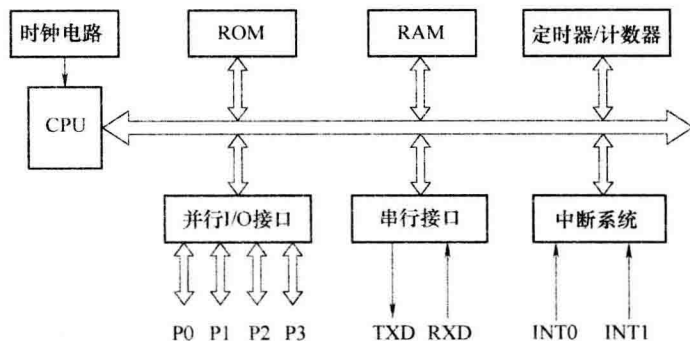


图 1-1 MCS-51 单片机基本结构

80C51 单片机的内部资源主要包括:

- 8 位 CPU;
- 片内振荡器和时钟电路;
- 4KB 片内程序存储器 (ROM);
- 128B 的片内数据存储器 (RAM);
- 2 个 16 位定时器/计数器;
- 4 个 8 位双向 I/O 口;
- 1 个全双工串行口;
- 5 个中断源, 2 个优先级;
- 21 个专用寄存器;
- 对片内外程序存储器和片内外数据存储器寻址的控制电路;
- 逻辑操作位寻址能力。

CPU 是单片机的内部核心部件, CPU 的主要功能是产生各种控制信号, 控制存储器和端口的数据传送、数据运算等。CPU 包括控制器和运算器, 下面分别介绍这两部分的组成及功能。

1. 控制器

控制器包含程序计数器 (PC)、指令寄存器 (IR)、指令译码器 (ID)、和数据指针 (DPTR)。

(1) 程序计数器 (Program Counter, PC)

PC 是一个 16 位的专用寄存器, 存放着下一条要执行指令的首地址。当 CPU 要取指令时, PC 的内容就会出现在地址总线上; 取指令后, PC 内容可自动加 1, 保证程序按顺序执行。此外, PC 内容可以通过指令修改, 以实现程序的跳转。系统复位后, PC 会被自动赋值为 0000H, 这里的地址指的是程序存储器的地址。

利用 Keil 软件在调试状态下, 通过观察 PC 的值可知道目前程序执行的位置。

(2) 指令寄存器 (Instruction Register, IR)

指令寄存器是一个 8 位寄存器, 用于暂存执行的指令, 等待译码。

(3) 指令译码器 (Instruction Decoder, ID)

指令译码器是对指令寄存器中的指令进行译码, 将指令转变为执行此指令所需的电信号。根据译码器输出的信号, 再经过定时控制电路产生执行该指令所需要的各种控制信号。

(4) 数据指针 (Data Pointer, DPTR)

DPTR 是一个 16 位的专用地址指针寄存器, 由两个 8 位寄存器 DPH (高 8 位) 和 DPL (低 8 位) 组成。可作 16 位寄存器, 也可作两个单独 8 位寄存器使用。DPTR 可用来指向全部 ROM 地址空间和片外 RAM 地址空间。

2. 运算器

运算器包含累加器 (ACC)、算术逻辑部件 (ALU) 和程序状态字 (PSW)。

(1) 累加器 (Accumulator, ACC)

ACC 是一个 8 位二进制寄存器, 简称 A, 通过暂存器与 ALU 相连, 用来存放操作数和运算结果。在 CPU 执行某种运算前, 两个操作数中的一个通常放在累加器 A 中, 运算完后便把结果存放在累加器 A 中; 可以说, 累加器 A 是使用最频繁的寄存器。

(2) 算术逻辑部件 (Arithmetic Logic Unit, ALU)

ALU 由一个加法器、两个 8 位暂存器 (TMP1 和 TMP2) 和布尔处理器组成。ALU 是 51 单片机的处理核心, 程序通过累加器 A、寄存器 B 和寄存器组 R0 ~ R7 等控制 ALU 以完成各种算术和逻辑运算, 同时可以用乘法和除法指令来增强运算能力。

(3) 程序状态字寄存器 (Program State Word, PSW)

PSW 是一个 8 位专用寄存器, 用于存放程序运行过程中的各种状态信息。PSW 中的各位信息通常是在指令执行过程中自动形成的, 但也可以由传送指令加以改变。PSW 各位的定义如表 1-3 所示。

表 1-3 PSW 各位的定义

PSW7	PSW6	PSW5	PSW4	PSW3	PSW2	PSW1	PSW0
CY	AC	F0	RS1	RS0	OV	F1	P

1) 进位标志位 (Carry, CY)

用于表示加法运算中的进位和减法中的借位。如果运算时, 操作结果最高位有进位或借位时, CY 由硬件置 “1”, 否则清 “0”。

2) 辅助进位标志 (Auxiliary Carry, AC)

用于加法或减法运算时, 如果操作结果的低 4 位向高 4 位产生进位或借位时, 将由硬件置 “1”, 否则清 “0”。

3) 用户标志位 (F0)

该位是由用户根据程序执行的需要自行设定的标志位, 用户可以通过设置该位来决定程序的流向。

4) 寄存器选择位 (RS1 和 RS0)

用于选择 CPU 当前工作的寄存器组。用户可以设置 RS1 和 RS0 来确定使用哪个寄存器组。如表 1-4 所示。



表 1-4 RS1、RS0 选择工作寄存器组

RS1、RS0 位的值	R0 ~ R7 寄存器组号	在 RAM 中的物理地址
00	0	00H ~ 07H
01	1	08H ~ 0FH
10	2	10H ~ 17H
11	3	18H ~ 1FH

5) 溢出标志位 (Overflow, OV)

可以指示运算过程中是否发生了溢出, 由硬件自动形成。若在执行有符号数加法、减法运算指令过程中, 累加器 A 中的运算结果超出了 8 位数能表示的范围, 即 $-128 \sim +127$, 则 OV 标志自动置“1”, 否则清“0”。因此, 根据 OV 状态可以判断累加器 A 中的结果是否正确。

OV 的状态可以利用异或逻辑表达式算出:

$$OV = C6y \oplus C7y$$

式中, $C6y$ 和 $C7y$ 分别是 D6 位和 D7 位的进位或借位状态, 有进位或借位时为 1, 否则为 0。

6) F1 用户标志位, 同 F0。

7) 奇偶标志位 (Parity, P)

奇偶标志位 (P) 用于指示运算结果中 1 的个数的奇偶属性。若 $P=1$, 则累加器 A 中 1 的个数为奇数; 若 $P=0$, 则累加器 A 中 1 的个数为偶数。

1.4.2 MCS-51 的外部引脚及功能

MCS-51 单片机引脚如图 1-2 所示, 共 40 条引脚, 包括 32 条 I/O 接口、4 条控制引脚、2 条电源引脚和 2 条时钟引脚。

1. I/O (32 条)

(1) P0 口 ($P0.0 \sim P0.7$), $P0.0$ 是最低位, 有两种功能:

1) 通用 I/O 接口: 无片外存储器时, P0 口可作通用 I/O 接口使用。

2) 地址/数据口。

(2) P1 口 ($P1.0 \sim P1.7$): 用作 I/O 口。

(3) P2 口 ($P2.0 \sim P2.7$), 有两种功能:

1) 通用 I/O 接口: 无片外存储器时, P2 口可作通用 I/O 接口使用。

2) 地址口: 在访问外部存储器时, 传送 ROM/RAM 高 8 位地址。

(4) P3 口 ($P3.0 \sim P3.7$), $P3.0$ 是最低位双向 I/O 接口, 有两种功能:

1) 第一功能: 通用 I/O 接口。

2) 第二功能: 用于串行口、中断源输入、计数器、片外 RAM 选通。

2. 控制引脚 (4 条)

(1) $\overline{ALE}/\overline{PROG}$ 它是地址锁存允许/编程脉冲输入信号线, 配合 P0 口引脚的第二功能使用。

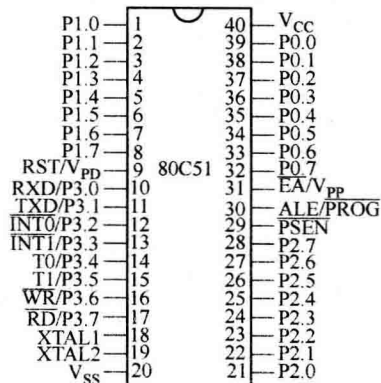


图 1-2 80C51 单片机引脚图

(2) $\overline{\text{PSEN}}$ 是片外程序存储器 ROM 的开放信号引脚, 用作输出访问片外程序存储器时的读选通信号。

(3) $\overline{\text{EA}}/\text{V}_{\text{pp}}$ 允许访问片外程序存储器/编程电源输入引脚, 可以控制 8051 使用片内还是片外 ROM。

(4) $\text{RST}/\text{V}_{\text{PD}}$ 复位/备用电源引脚 (1 条)。

复位端 RST: 单片机上电后, 其内部各寄存器都处于随机状态。若在该引脚上输入满足复位时要求的高电平, 将使单片机复位。

备用电源端 V_{PD} 在主电源掉电期间, 可利用该引脚外接的 +5V 备用电源为单片机片内 RAM 供电, 保证片内 RAM 信息不丢失, 以便电压恢复正常后单片机能正常工作。

3. 电源引脚 (2 条)

V_{CC} : +5V 电源引脚; V_{SS} : 接地引脚。

4. 晶振引脚 (2 条)

MCS-51 单片机芯片内部有一个用于构成振荡器的高增益反相放大器, 引脚 XTAL1 和 XTAL2 分别是此放大器的输入端和输出端, 即:

(1) XTAL1: 反相振荡放大器的输入引脚。

(2) XTAL2: 反相振荡放大器的输出引脚。

1.5 MCS-51 的存储器

1.5.1 存储器的结构

MCS-51 单片机存储器的特点是将程序存储器 (ROM) 和数据存储器 (RAM 和 SFR) 分开编址, 并有各自的寻址方式和寻址单元。对存储器的划分在物理上分为 4 个空间: 片内程序存储器、片外程序存储器、片内数据存储器 and 片外数据存储器, MCS-51 单片机存储空间结构图如图 1-3 所示。

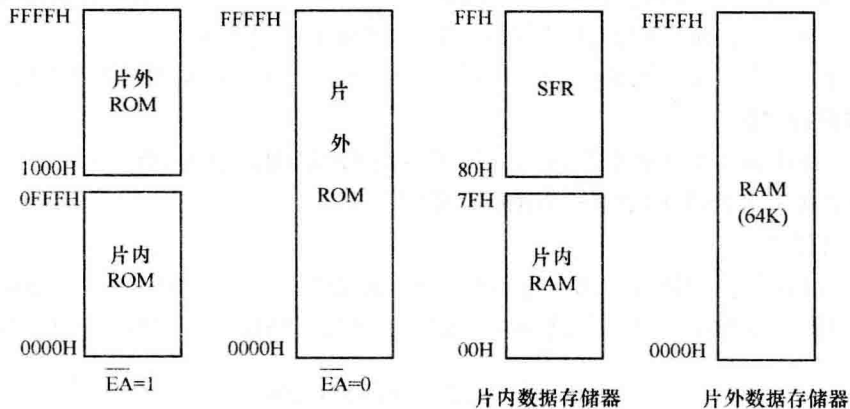


图 1-3 MCS-51 单片机存储空间结构图

当 $\overline{\text{EA}} = 1$ 时, 程序存储器的地址分为两部分, 片内存储器占 0000H ~ 0FFFH 地址范围, 片外程序存储器占 1000H ~ FFFFH 的地址范围; 当 $\overline{\text{EA}} = 0$ 时, 片外程序存储器占 0000H ~ FFFFH 的地址范围, 而片内存储器不被使用。