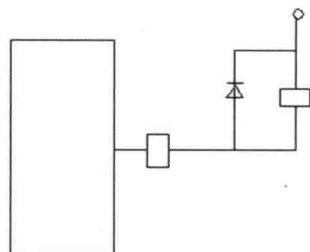


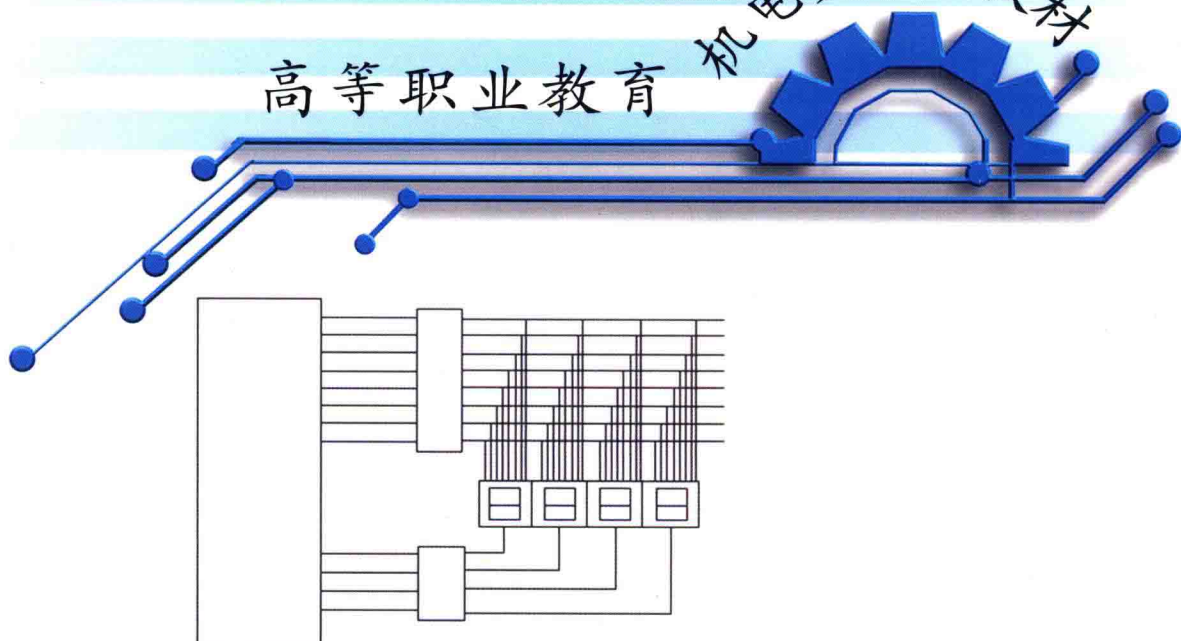


李 珍 付植桐 编 著



# 单片机原理与 应用技术

机电类系列教材  
高等职业教育



清华大学出版社

内容简介

高等职业教育机电类系列教材

# 单片机原理与应用技术

李 珍 付植桐 编著

12368  
88

本书是为适应高等院校、中等职业学校机电类专业学生必修的一门专业基础课程。通过本书的学习，使学生可以掌握必需的单片机基本理论和实际应用技能，为后续专业课程的学习以及从事单片机应用开发打下良好的基础。

本书在编写过程中，力求做到概念清晰、重点突出、由浅入深、循序渐进。全书共分8章，主要内容包括：单片机的组成与工作原理、单片机的指令系统、单片机的并行接口、单片机的串行接口、单片机的中断系统、单片机的定时器/计数器、单片机的I/O接口、单片机的应用。本书可作为高等院校、中等职业学校机电类专业教材，也可供从事单片机应用开发的工程技术人员参考。

本书在编写过程中，力求做到概念清晰、重点突出、由浅入深、循序渐进。全书共分8章，主要内容包括：单片机的组成与工作原理、单片机的指令系统、单片机的并行接口、单片机的串行接口、单片机的中断系统、单片机的定时器/计数器、单片机的I/O接口、单片机的应用。本书可作为高等院校、中等职业学校机电类专业教材，也可供从事单片机应用开发的工程技术人员参考。

本书在编写过程中，力求做到概念清晰、重点突出、由浅入深、循序渐进。全书共分8章，主要内容包括：单片机的组成与工作原理、单片机的指令系统、单片机的并行接口、单片机的串行接口、单片机的中断系统、单片机的定时器/计数器、单片机的I/O接口、单片机的应用。本书可作为高等院校、中等职业学校机电类专业教材，也可供从事单片机应用开发的工程技术人员参考。

本书在编写过程中，力求做到概念清晰、重点突出、由浅入深、循序渐进。全书共分8章，主要内容包括：单片机的组成与工作原理、单片机的指令系统、单片机的并行接口、单片机的串行接口、单片机的中断系统、单片机的定时器/计数器、单片机的I/O接口、单片机的应用。本书可作为高等院校、中等职业学校机电类专业教材，也可供从事单片机应用开发的工程技术人员参考。

本书在编写过程中，力求做到概念清晰、重点突出、由浅入深、循序渐进。全书共分8章，主要内容包括：单片机的组成与工作原理、单片机的指令系统、单片机的并行接口、单片机的串行接口、单片机的中断系统、单片机的定时器/计数器、单片机的I/O接口、单片机的应用。本书可作为高等院校、中等职业学校机电类专业教材，也可供从事单片机应用开发的工程技术人员参考。

本书在编写过程中，力求做到概念清晰、重点突出、由浅入深、循序渐进。全书共分8章，主要内容包括：单片机的组成与工作原理、单片机的指令系统、单片机的并行接口、单片机的串行接口、单片机的中断系统、单片机的定时器/计数器、单片机的I/O接口、单片机的应用。本书可作为高等院校、中等职业学校机电类专业教材，也可供从事单片机应用开发的工程技术人员参考。

清华大学出版社

北京

本书在编写过程中，力求做到概念清晰、重点突出、由浅入深、循序渐进。全书共分8章，主要内容包括：单片机的组成与工作原理、单片机的指令系统、单片机的并行接口、单片机的串行接口、单片机的中断系统、单片机的定时器/计数器、单片机的I/O接口、单片机的应用。本书可作为高等院校、中等职业学校机电类专业教材，也可供从事单片机应用开发的工程技术人员参考。

清华大学出版社 北京

此外，三星还介绍了CC51系列单片机的中

本书包含了许多浅显易懂、典型实用的例题。各章节后都有简短的小结,并附有多种类型的习题,第12章专门介绍应用程序,这些程序均经过上机调试,读者可直接调用。附录中给出了单片机常用外围器件,供读者参考。

ISBN 7-302-07128-4

中国版本图书馆 CIP 数据核字(2003)第 073192 号

<http://www.tup.com.cn> 邮 编: 100084

社总机: 010-62770175 客户服务: 010-62776969

文稿编辑：马幸兆

装 订 者：北京市密云县京文制本装订厂

开 本: 185 × 260 印张: 14 字数: 320 千字

书 号: ISBN 7-302-07128-4/TP · 5213

定 价: 21.00 元

本书如存在文字不清、漏印以及缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770175-3103 或(010)62795704

# 目 录

|                              |    |
|------------------------------|----|
| <b>第 1 章 单片微型计算机知识概述</b>     | 1  |
| 1.1 单片机知识概述                  | 1  |
| 1.2 单片机的基础知识                 | 3  |
| 本章小结                         | 5  |
| 思考题与习题                       | 5  |
| <b>第 2 章 单片机开发系统与模拟仿真技术</b>  | 6  |
| 2.1 单片机开发系统                  | 6  |
| 2.2 万利仿真器的安装                 | 6  |
| 2.3 万利仿真器的使用                 | 6  |
| <b>第 3 章 80C51 单片机的结构和原理</b> | 13 |
| 3.1 80C51 单片机的结构及引脚          | 13 |
| 3.2 80C51 存储器配置              | 16 |
| 3.3 数据存储器的配置                 | 18 |
| 3.4 80C51 CPU 的时序            | 24 |
| 3.5 复位及复位电路                  | 26 |
| 本章小结                         | 28 |
| 思考题与习题                       | 28 |
| <b>第 4 章 80C51 指令系统</b>      | 30 |
| 4.1 指令格式和寻址方式                | 30 |
| 4.2 指令系统                     | 34 |
| 4.3 简单指令的模拟仿真                | 49 |
| 本章小结                         | 51 |
| 思考题与习题                       | 51 |
| <b>第 5 章 汇编语言程序设计</b>        | 54 |
| 5.1 程序设计语言                   | 54 |
| 5.2 汇编语言程序设计                 | 54 |
| 5.3 汇编语言程序设计举例               | 58 |
| 5.4 实用子程序设计                  | 67 |

|                             |            |
|-----------------------------|------------|
| 5.5 简单程序的模拟仿真               | 72         |
| 本章小结                        | 73         |
| 思考题与习题                      | 73         |
| <b>第6章 并行输入输出接口</b>         | <b>75</b>  |
| 6.1 80C51 单片机中的并行输入输出       | 75         |
| 6.2 并行口的输出                  | 79         |
| 6.3 利用仿真软件模拟并行口的输出          | 80         |
| 6.4 显示器的输出                  | 81         |
| 6.5 并行输入                    | 85         |
| 本章小结                        | 90         |
| 思考题与习题                      | 90         |
| <b>第7章 中断系统</b>             | <b>92</b>  |
| 7.1 微机的输入输出方式               | 92         |
| 7.2 中断的概述                   | 93         |
| 7.3 80C51 中断系统结构及中断控制       | 94         |
| 7.4 中断处理过程                  | 100        |
| 7.5 外部中断应用和模拟仿真             | 103        |
| 本章小结                        | 108        |
| 思考题与习题                      | 108        |
| <b>第8章 单片机的定时计数功能及其应用</b>   | <b>110</b> |
| 8.1 定时器的结构与工作原理             | 110        |
| 8.2 定时器/计数器的控制              | 111        |
| 8.3 定时器/计数器的工作模式及应用         | 113        |
| 8.4 定时器/计数器模拟仿真             | 119        |
| 8.5 定时器/计数器的综合应用            | 121        |
| 本章小结                        | 128        |
| 思考题与习题                      | 129        |
| <b>第9章 80C51 串行口及串行通信技术</b> | <b>131</b> |
| 9.1 串行通信基础知识                | 131        |
| 9.2 串行口的特点                  | 132        |
| 9.3 串行通信工作方式                | 136        |
| 9.4 80C51 串行口的应用            | 138        |
| 9.5 RS-232C 标准接口总线          | 146        |
| 9.6 串行口模拟仿真                 | 148        |
| 本章小结                        | 149        |
| 思考题与习题                      | 150        |

|                                  |     |
|----------------------------------|-----|
| 第 10 章 存储器及 I/O 口的扩展 .....       | 152 |
| 10.1 扩展系统的组成 .....               | 152 |
| 10.2 存储器概述 .....                 | 153 |
| 10.3 程序存储器的扩展 .....              | 155 |
| 10.4 数据存储器的扩展 .....              | 158 |
| 10.5 片选方式和地址分配 .....             | 160 |
| 10.6 80C51 单片机并行 I/O 接口的扩展 ..... | 164 |
| 本章小结 .....                       | 172 |
| 思考题与习题 .....                     | 172 |
| 第 11 章 接口技术 .....                | 174 |
| 11.1 前向通道的配置及接口技术 .....          | 174 |
| 11.2 系统后向通道配置及接口技术 .....         | 178 |
| 本章小结 .....                       | 185 |
| 思考题与习题 .....                     | 185 |
| 第 12 章 实用程序设计 .....              | 187 |
| 12.1 数字滤波程序设计 .....              | 187 |
| 12.2 算术运算类程序设计 .....             | 189 |
| 12.3 交通灯控制 .....                 | 196 |
| 12.4 倒计时器设计 .....                | 198 |
| 附录 A 80C51 指令表 .....             | 205 |
| 附录 B 常用集成电路引脚图 .....             | 211 |
| 参考文献 .....                       | 215 |

### 1.1.2 什么是单片机

单片机也被称为“单片微型计算机”、“微控制器”、“嵌入式微控制器”等。单片机一词源于“single-chip microcomputer”，简称 SCM。国际上逐渐采用“MCU(micro controller unit)”来代替。单片微型计算机就是将 CPU、RAM、ROM、定时器/计数器和多种接口电路都集成到一块集成电路芯片上的微型计算机。一块芯片就构成了一台小型计算机。它已成为工业控制领域、智能仪器仪表、尖端武器和日常生活中最广泛使用的计算机。

### 1.1.3 单片机的发展

自单片机诞生至今,已发展成为上百种系列、数千个机种。单片机的发展历史大致分为以下几个阶段:

(1) 第一阶段(1976~1978 年),低性能单片机的探索阶段。以 Intel 公司的 MCS-48 为代表,采用了单片结构,即在一块芯片内含有 8 位 CPU、定时器/计数器、并行 I/O 口、RAM 和 ROM 等。主要用于工业领域。“单片机”一词由此而来。

## 第1章 单片微型计算机知识概述

### 1.1 单片机知识概述

#### 1.1.1 计算机的发展

计算机的发展经历了从电子管、晶体管、集成电路到大(超大)规模集成电路共4个阶段,即通常所说的第一代、第二代、第三代、第四代计算机。现在广泛使用的微型计算机是大规模集成电路技术发展的产物,属于第四代计算机。由于实际应用的需要,微型计算机向着两个不同的方向发展,即高速度、大容量、高性能的高档微机发展方向与稳定可靠、体积小、价格廉的单片机方向,形成了通用计算机系统和嵌入式计算机系统两大分支。

通用计算机系统主要用于大型科学研究和试验以及超高速数学计算,它的研究水平标志着一个国家的科学技术和工业发展的程度,象征着一个国家的实力。通用计算机系统的数据总线宽度从8位、16位、32位发展到了64位,操作系统不断完善,突出发展了高速计算能力,并在数据处理、模拟仿真、人工智能、图像处理、多媒体及网络通信中得到了广泛的应用。

通用计算机的巨大体积和高成本,无法嵌入到大多数仪器仪表、家用电器、汽车、机器人等智能化仪器中。20世纪70年代嵌入式微型计算机诞生之后,把计算机的应用领域推向了全社会。嵌入式微型计算机不断增强控制能力,降低成本,减小体积,改善开发环境,可广泛地嵌入到现代电子系统中,对社会生产力的发展和人类生活的改变起到了极大的促进作用。

#### 1.1.2 什么是单片机

单片机也被称为“单片微型计算机”、“微控制器”、“嵌入式微控制器”等。单片机一词源于“single chip microcomputer”,简称SCM。国际上逐渐采用“MCU(micro controller unit)”来代替。单片微型计算机就是将CPU、RAM、ROM、定时器/计数器和多种接口电路都集成到一块集成电路芯片上的微型计算机,一块芯片就构成了一台小型计算机。它已成为工业控制领域、智能仪器仪表、尖端武器和日常生活中最广泛使用的计算机。

#### 1.1.3 单片机的发展

自单片机诞生至今,已发展成为上百种系列、近千个机种。单片机的发展历史大致分为以下几个阶段:

(1) 第一阶段(1976—1978年):低性能单片机的探索阶段。以Intel公司的MCS-48为代表,采用了单片结构,即在一块芯片内含有8位CPU、定时器/计数器、并行I/O口、RAM和ROM等。主要用于工业领域。“单片机”一词由此而来。

(2) 第二阶段(1978—1982年):高性能单片机发展阶段。这一类单片机带有串行 I/O口,8 位数据线、16 位地址线、控制总线和较丰富的指令系统等。典型的单片机系列为 MCS-51,这类单片机的应用范围较广,并在不断地改进和发展。

(3) 第三阶段(1982—1990年):16 位单片机发展阶段。16 位单片机除 CPU 为 16 位外,片内 RAM 和 ROM 容量进一步增大,实时处理能力更强,体现了微控制器的特征。例如 Intel 公司生产的 MCS-96 系列单片机,其振荡频率为 12 MHz,片内 RAM 为 232 B,ROM 为 8 KB,中断处理能力为 8 级,片内带有 10 位 A/D 转换器和高速输入/输出部件等。

(4) 第四阶段(1990 年至今):微控制器的全面发展阶段。各公司的产品在尽量兼容的同时,出现了高速、强运算能力、寻址范围大的 8 位、16 位、32 位通用型单片机,以及小型廉价的专用型单片机。

#### 1.1.4 单片机的组成特点

单片机是微型计算机的一个重要分支,结构上的最大特点是把 CPU、存储器、定时器和多种输入输出接口电路集成在一块超大规模电路芯片上,就其组成和功能而言,一块单片机芯片就是一台计算机。单片机具有许多显著的特点:

① 集成度高、体积小、有很高的可靠性。单片机把各功能部件集成在一块芯片上,内部采用总线结构,减少了各芯片间的连线,大大提高了单片机的可靠性与抗干扰能力。另外,单片机体积小,对于强磁场环境易于采取屏蔽措施,适合在恶劣环境下工作。

② 控制功能强。一般单片机指令系统中均有丰富的转移指令、逻辑操作以及位处理功能。

③ 低功耗、低电压、便于生产便携式产品。

④ 外部总线增加了 I<sup>2</sup>C 及 SPI 等串行总线方式,进一步简化了结构。

⑤ 专用型单片机可针对某一类产品设计,小封装,低价格,外围器件和外部设备接口电路集成度更高。

#### 1.1.5 单片机与单片机系统

单片机是由不同制造商生产的,集成了微型计算机基本组成部分的一块芯片,包括运算放大电路、控制电路、存储器、中断系统、定时器/计数器、输入输出等电路。但是一块单片机芯片不可能把组成微型计算机的全部电路,如谐振电路、复位电路等都集成在一起,组成这些电路的石英晶振、电阻、电容等元器件只能以散件的形式出现。因此一块单片机芯片并不能组成具有一定控制功能的系统。只有在单片机芯片基础上,扩展具有其他功能的控制系统外围电路和外围芯片,才能构成一定控制功能的单片机控制系统。

#### 1.1.6 单片机的应用

##### 1. 单片机在智能仪器中的应用

单片机广泛使用于各种仪器仪表中,因此可以提高仪器仪表的智能化程度,提高测量

精度,并简化仪器仪表的硬件结构。

## 2. 单片机在机电一体化中的应用

机电一体化是机械工业的发展方向。单片机作为机电一体产品中的控制器件,能发挥其体积小、功能强的优点,可以大大提高机械产品的自动化、智能化程度。

## 3. 单片机在实时控制中的应用

单片机广泛用于各种实时控制系统中,单片机的实时处理能力和控制功能,可使系统保持在最佳工作状态,提高系统的工作效率和产品质量。

## 4. 单片机在人类生活中的应用

自从单片机诞生以后,它就步入了人们的生活。如洗衣机、电冰箱、电子玩具等家用电器配上单片机后,提高了智能化程度,增加了功能,倍受人们喜爱。单片机将使人类生活更加方便、舒适、丰富多彩。

# 1.2 单片机的基础知识

## 1.2.1 数和数制

单片机中经常使用的数有十进制数、二进制数、十六进制数。

十进制的基为“十”,它所使用的数码为0~9,共10个数字。十进制数的后缀为D,通常对十进制数不加后缀。

二进制的基为“二”,它所使用的数码为0,1,共2个数字。二进制数的后缀为B,如1001B,代表十进制数9;但二进制数位太长,不易记忆和书写。

十六进制的基为“十六”,它所使用的数码为0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F,共16个数字。其中A~F相当于十进制数的10~15,十六进制数的后缀为H。十六进制数如果是字母打头,则在使用汇编指令时前面需加一个0。

## 1.2.2 计算机的编码

### 1. 二-十进制编码

二-十进制编码称为二进制码的十进制数,它即不是二进制数也不是十进制数,简称BCD(binary coded decimal)码。在BCD码中是用4位二进制数给0~9这10个数字编码,为了书写简便,在程序中BCD码也常简写成十六进制形式。

使用BCD码既考虑了计算机的特点,又照顾了人们使用十进制数的习惯,广泛地使用于单片机的输入、输出操作中。

按照BCD码与十进制的关系,可以很容易地实现BCD码和十进制数之间的转换。

**例 1.1** BCD码0010 1001 0111=297D。但是,BCD码和二进制数、十六进制数都

是有区别的。例如：十进制数 23，它的 BCD 码，用二进制表示为 0010 0011B，简写成十六进制形式为 23H。

若把它转换成二进制数为：00010111B，而十六进制数为：17H。

BCD 码、二进制数、十六进制数和十进制数之间的转换关系如表 1.1 所列。

表 1.1 BCD 码、二进制数、十六进制数和十进制数之间的转换关系

| 十进制数 | 十六进制数 | 二进制数  | BCD 码 | 十进制数 | 十六进制数 | 二进制数  | BCD 码     |
|------|-------|-------|-------|------|-------|-------|-----------|
| 0    | 0H    | 0000B | 0000B | 8    | 8H    | 1000B | 1000B     |
| 1    | 1H    | 0001B | 0001B | 9    | 9H    | 1001B | 1001B     |
| 2    | 2H    | 0010B | 0010B | 10   | 0AH   | 1010B | 00010000B |
| 3    | 3H    | 0011B | 0011B | 11   | 0BH   | 1011B | 00010001B |
| 4    | 4H    | 0100B | 0100B | 12   | 0CH   | 1100B | 00010010B |
| 5    | 5H    | 0101B | 0101B | 13   | 0DH   | 1101B | 00010011B |
| 6    | 6H    | 0110B | 0110B | 14   | 0EH   | 1110B | 00010100B |
| 7    | 7H    | 0111B | 0111B | 15   | 0FH   | 1111B | 00010101B |

注意：在程序中，人们习惯使用十六进制数表示的 BCD 码，要特别注意 BCD 码与十六进制数的区别。

2. 字母与字符的编码

由于计算机中使用的是二进制数，所以计算机中使用的字母、字符也要用特定的二进制码表示。字母与字符的二进制码表示方法很多，普遍采用的是 ASCII 码 (american standard code for information interchange)。它采用 7 位二进制编码表示 128 个字符，其中包括数码 0~9 以及英文字母等可打印的字符，如表 1.2 所列。

表 1.2 ASCII 码表

| <div>b6b5b4<br/>b3b2b1b0</div> | 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   |
| 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  | US  | /   | ?   | O   | ←   | o   | DEL |

### 1.2.3 位、字节和字

1 位二进制代码称为 1 位,用“bit”表示,可简写为“b”;8 位二进制代码称为 1 个字节,用“byte”表示,可简写为“B”;两个字节称为一个字,用“word”表示。位、字节和字都是表示二进制数的常用单位。另外,以后学习中还会遇到“半字节”,半字节即 4 位二进制数。

### 1.2.4 电平

逻辑电路中,电位的高低常用高电平、低电平来描述。由于温度变化、电源波动、电磁干扰及元件特性变化等原因的影响,实际的高低电平都不是一个固定值。因此,通常考虑一个电平的变化范围,如果在此范围内,就判断为“1”或“0”状态。对于各种集成电路,规定了一个高电平的下限值和低电平的上限值,称为标准高电平和标准低电平。标准高电平为 2.4 V,标准低电平为 0.4 V。电压大于 2.4 V 为高电平,用“1”表示,电压小于 0.4 V 为低电平,用“0”表示。

## 本章小结

本章介绍了单片机以及它的发展概况、特点及其应用领域。单片微型计算机简称单片机,就是将 CPU、RAM、ROM、定时器/计数器和多种接口都集成到一块集成电路芯片上的微型计算机,一块芯片就构成了一台计算机。它广泛应用于工业控制领域、智能仪器仪表、尖端武器和日常生活中。

在单片机芯片基础上扩展其他电路或芯片构成的具有一定功能的系统称为单片机系统。

计算机中常使用二进制数,它的后缀为 B,为了书写方便,常将二进制数写成十六进制数,它的后缀为 H,在输入、输出操作中,计算机经常使用的编码为 BCD 码和 ASCII 码。

一个字节为 8 位二进制数,简写为“B”;两个字节称为一个字,用“word”表示;半字节为 4 位二进制数。

逻辑电路中一般称电压小于 0.4 V 为低电平,用“0”表示;电压大于 2.4 V 为高电平,用“1”表示。

## 思考题与习题

1. 什么是单片机,它与一般计算机有何区别?
2. 单片机运行时采用什么数制?为什么?在编程时为何经常使用十六进制?
3. 单片机主要应用于哪些领域?

## 第2章 单片机开发系统与模拟仿真技术

### 2.1 单片机开发系统

单片机系统是与控制对象紧密结合的一个完整的应用控制系统。由于单片机本身软硬件资源所限,单片机系统不能实现自我软件开发和硬件调试。要进行系统开发设计,必须使用专门的单片机开发系统,该系统称为在线仿真器。

在线仿真器是单片机系统开发调试的工具,包括仿真器、电源、仿真头和数据线4部分。其中,数据线用来和计算机进行通信;仿真头可模拟应用系统中单片机的功能,其大小、引脚排列位置和单片机一样。利用在线仿真器进行软件开发并与应用系统硬件电路一起调试的过程,称为在线仿真调试。达到控制目的后,把经过在线仿真调试的软件程序烧录到单片机芯片中,这块装有程序的单片机芯片与调试过的硬件电路一起才能组成单片机应用系统。

为了方便单片机产品开发人员、工程技术人员、大中专学生学习使用和尽快掌握单片机技术,许多公司开发了能在 Windows 环境下运行的单片机仿真开发系统。MedWin 是万利电子有限公司的高性能集成开发环境,它集编辑、编译/汇编、在线仿真调试及模拟仿真调试为一体,采用 VC 风格的用户窗口,配合该公司的系列仿真器,可以方便地学习和使用 80C51 单片机。如果不使用该公司的仿真器,也可利用 MedWin 仿真软件提供的模拟仿真功能,通过计算机模拟调试用汇编语言编写的软件程序,还可以模拟单片机中的定时器、中断、串口传送数据等情况,是初学单片机知识和工程技术人员的有力助手。

### 2.2 万利仿真器的安装

在计算机上,利用万利仿真器及其模拟仿真功能,可以与所设计的单片机硬件系统一起实现在线仿真调试,还可以在模拟仿真状态下模拟调试用汇编语言编写的软件程序。

MedWin 集成开发环境安装程序可向万利公司索取或从 [www.Manley.com.cn](http://www.Manley.com.cn) 下载。

万利仿真软件安装完毕后进入 Windows 环境,用鼠标单击“开始→程序→Manley → MedWin”选项,即可进入万利调试仿真器,也可在计算机屏幕上创建快捷方式。

### 2.3 万利仿真器的使用

#### 2.3.1 项目的建立与源程序输入

进入万利仿真器调试程序 MedWin,屏幕上会出现如图 2.1 所示的开始画面提示框。提示框中几个按钮的功能如下:

(1) 如果在计算机的串口已经连接了万利仿真器,要进行单片机在线仿真调试,单击“仿真器”按钮,可进入如图 2.2 所示的仿真调试画面。

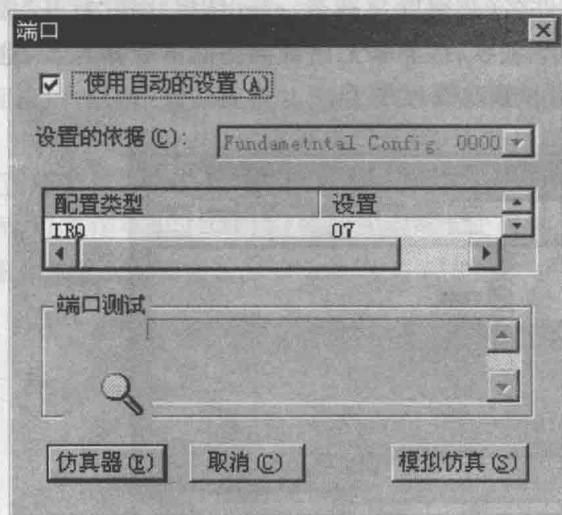


图 2.1 开始画面

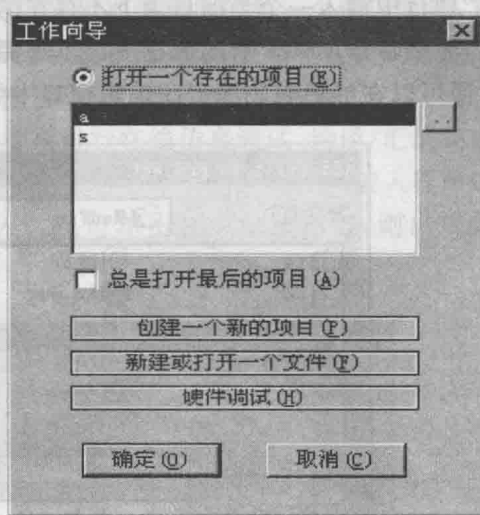


图 2.2 仿真调试画面

(2) 如果没连接仿真器,而只利用计算机进行模拟仿真调试,则单击“模拟仿真”按钮,同样可进入如图 2.2 所示的仿真调试画面,屏幕上出现的提示框功能分别是:

- ① 打开一个存在的项目;
- ② 创建一个新的项目;
- ③ 创建或打开一个新的文件;
- ④ 硬件调试。

项目相当于一个管理文件,每个项目可以管理多个子文件。单击“取消”按钮,则不打开或建立任何项目。单击“确定”按钮,仿真软件进入创建项目或打开项目过程。项目建立还可以在用户菜单中完成,单击菜单“项目管理→新建项目文件”,屏幕上出现如图 2.3 所示的建立项目提示画面。

在“项目名”栏目为您的项目文件取一个英文名字,如“project1”,单击“高级”按钮可以选择项目的存放路径,仿真软件一般要求将源程序与项目文件存放于同一目录。然后单击“确定”按钮,屏幕上出现如图 2.4 所示的添加源程序画面,为项目文件添加源程序。若在图 2.1 复选框“使用自动的设置”前已画“√”,

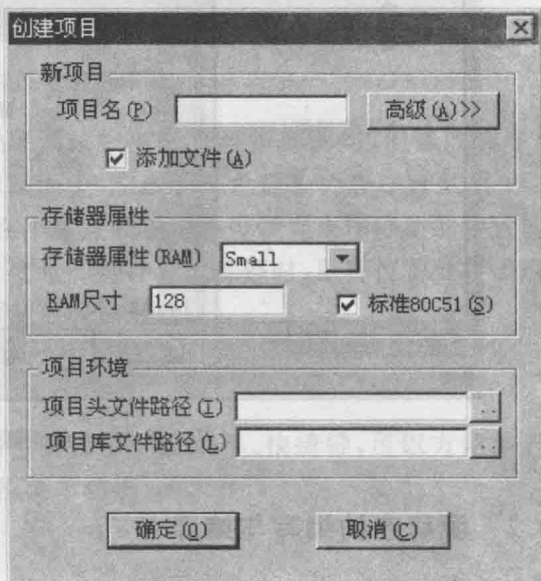


图 2.3 建立项目画面

图 2.3 中的“项目环境”栏可以不进行选择。

在图 2.4 中可以选择一个或几个存在的源程序并将其添加至本项目,也可在“文件名”栏目中键入一个当前目录下不存在的文件名(必须以.c 或者.asm 结尾),按“打开”按钮,为项目添加一个新的源程序。添加源程序名称后,屏幕上出现如图 2.5 所示画面,这时便可以在屏幕右侧源程序编辑窗口中编辑和修改源程序了。

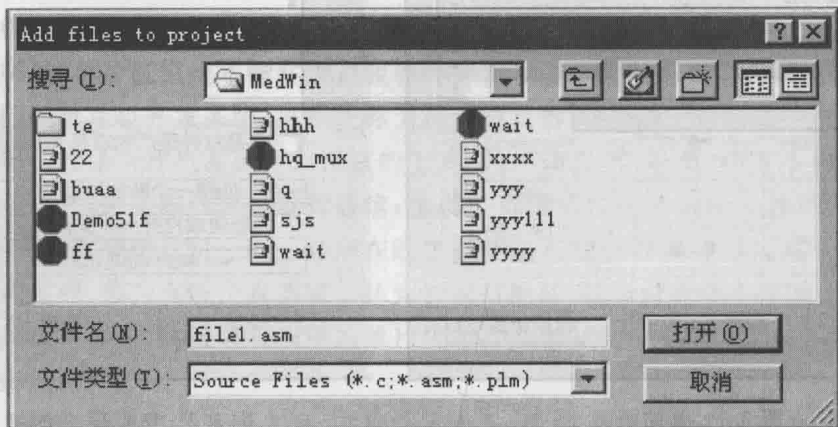


图 2.4 添加源程序画面

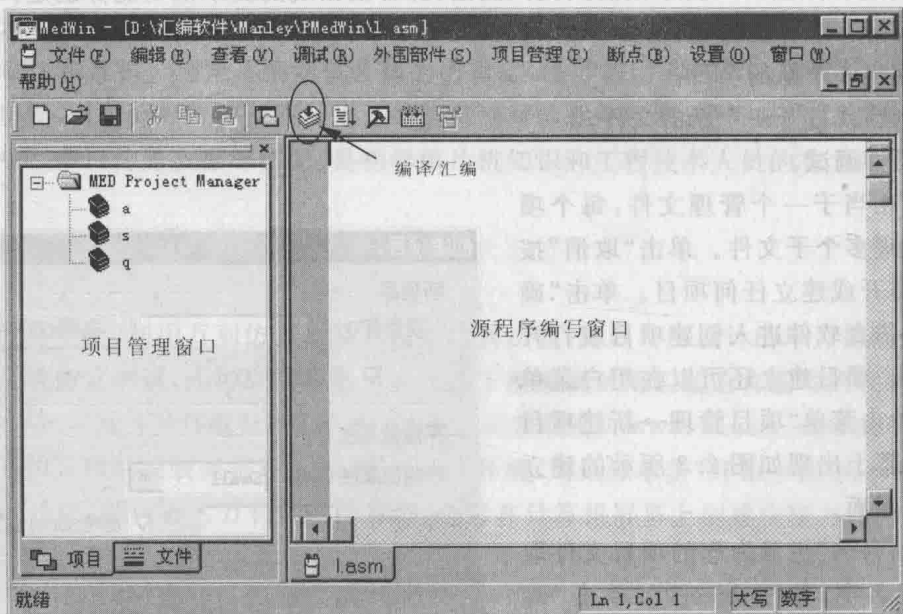


图 2.5 编辑源文件画面

### 2.3.2 源程序的编写与编译

用汇编语言或 C 语言助记符编写的程序,称为源程序。把源程序转换成计算机能识别的机器码的过程称为编译/汇编。

源程序编写完毕,按“Ctrl+F7”键,或单击图 2.5 中的“编译/汇编”快捷键,还可单击菜单“项目管理→编译/汇编”就可完成程序的编译过程。如果程序中有语法错误,则会在源程序编辑窗口中出现一个高亮度的红色条,它将固定位在源程序的第一个错误之处,其他的错误将在下面的提示窗口中出现;用鼠标双击错误提示语句,红色条将定位到源程序相应的错误位置。此项功能可以提示将所有的语法错误一次全部修改完毕。

程序修改完毕,再重复执行编译/汇编过程。如果进行在线仿真调试,编译/汇编完成后,还要把程序装入仿真器中,称为“产生代码并装入”。若想一次完成汇编和装入程序过程,可直接单击菜单“项目管理→产生代码并装入”,也可单击相应的快捷键,则出现如图 2.6 所示的画面。

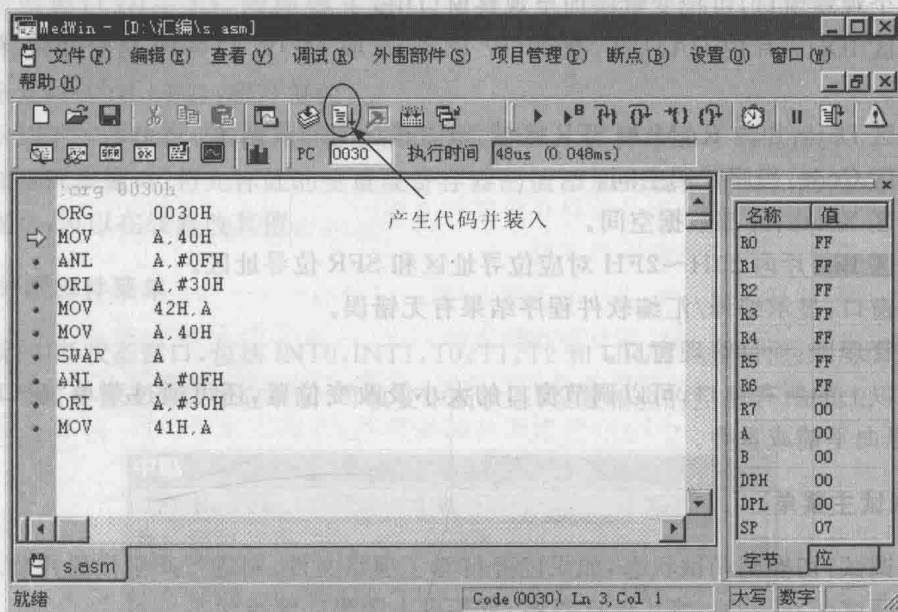


图 2.6 程序正常运行画面

万利仿真器支持源程序级调试,可在程序调试中随时修改指令语句,但修改程序后,需重新执行编译/汇编过程。在图 2.6 所示画面中,左边的黄色箭头为程序计数器 PC 指针,小蓝点表示程序可单步执行的每一步,在小蓝点处单击便可设置程序停留的位置即断点,再次单击小蓝点可取消断点。当鼠标在变量和寄存器上悬停时,鼠标位置处将出现该变量的值,且分别以十进制数(前)和十六进制数(后)显示。

### 2.3.3 菜单命令简介

在 Windows 环境下的万利仿真器窗口中有多项下拉菜单、快捷键,可以方便地进行编程、汇编、运行等操作。下面分别介绍下拉菜单功能。

#### 1. 查看菜单

寄存器(R):寄存器窗口,显示 80C51 中的寄存器 R0~R7, A, B, DPH, DPL, SP 和

PSW 中的各位。以 16 进制形式显示字节单元内容,以位显示 PSW 的内容。

特殊功能寄存器(S):CPU 所包含的特殊功能寄存器窗口,以 16 进制显示。

选择这两项子菜单,寄存器窗口和特殊功能寄存器窗口将出现在图 2.6 中的源程序编辑窗口右侧。

反汇编窗口:将程序代码以机器码方式及源程序方式显示。

观察窗口第一观察组(1),

观察窗口第一观察组(2),

观察窗口第一观察组(3),

观察窗口第一观察组(4)等:为了方便用户,避免多次添加和删除需要观察的变量,而设置的几个观察窗口,可把变量添加至观察窗口中。

数据区 IData:片内 RAM 区,被 MOV @Ri,A 或 MOV A,@Ri 指令间接寻址访问的数据区。

数据区 Data:片内 RAM 和 SFR 区域,被直接访问的数据区。

数据区 Code:程序代码空间。

数据区 Xdata:外部数据空间。

数据区 Bit:片内 20H~2FH 对应位寻址区和 SFR 位寻址区。

消息窗口:显示编译/汇编软件程序结果有无错误。

项目管理器:项目管理窗口。

打开以上的所有窗口,可以调节窗口的大小及改变位置,还可通过菜单“窗口”,使它们横向、纵向平铺或重叠。

## 2. 调试主菜单

开始调试:切换到调试状态,如果已经打开了项目文件,则进行产生代码并装入操作。

终止调试(Ctrl+D):终止调试程序,重新进入源程序编辑状态。

全速运行(F9):可使程序全速运行,遇断点停止。运行时程序显示如图 2.7 所示。

图 2.7 中程序计数器显示的“001F”表示程序运行的当前位置即 PC 值。程序运行在“S1”为符号地址定位,表示程序已执行到哪一个符号地址之下。单击“停止”按钮可返回监控状态。

禁止断点并全速运行(Alt+F9):  
可以屏蔽所有的断点运行。

跟踪(F7):单步执行,程序跟踪进入子程序。

单步(F8):单步执行,子程序只作为一步运行。

运行到光标处(F4):程序运行到光标所在行停下来。

运行到 RETURN 处(Alt+F8):运行到子程序或中断返回处停下来。

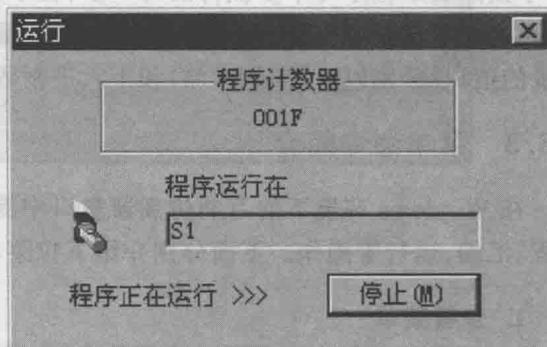


图 2.7 程序运行画面

执行到:程序运行到设置的 PC 地址或符号地址处停下来。

设置新的程序计数器(Ctrl+N):可以改变程序运行的当前位置。

自动单步:单击此菜单,将弹出如图 2.8 所示的窗口。

可以键入一个时间间隔,单击“确定”按钮,程序自动单步连续执行。

设置重复计数值:可以设定断点的计数值。

返回监控(Ctrl+T):程序停止运行,返回监控状态。

程序复位(Ctrl+F2):程序复位。

显示到下一步执行:将下一步将要执行的指令显示在屏幕的可见位置。

添加至观察窗口:将光标处的变量或寄存器的值添加到观察窗口。通过双击观察窗口里的变量,可以在线修改其值。

### 3. 外围部件菜单

中断:中断状态窗口,包括 INT0,INT1,T0,T1,T2 和 URAT 中断状态以及优先级和允许设置。可以设置或清除相应中断,可改变中断状态,设置相应的控制字,如图 2.9 所示。

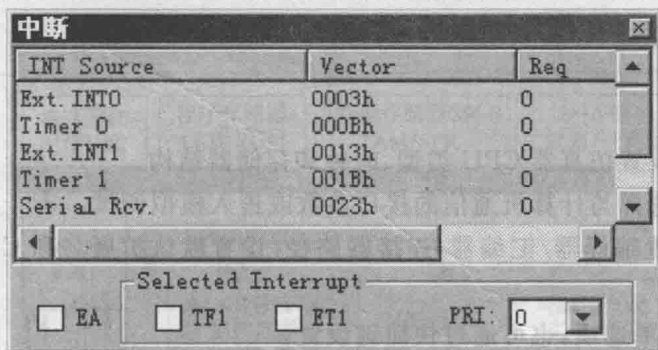


图 2.9 中断状态

端口:端口设置窗口,单击窗口中的小亮点,可改变端口显示状态,如图 2.10 所示, P1~P3 口初态为点亮状态。

定时器/计数器 T0:设置工作模式和控制窗口,TMOD 和 TCON 的值都可设置,如图 2.11 所示。可单击各设置值下的小亮点,改变或恢复设置值。

串行口:设置工作模式和控制窗口,SMOD 和 SCON 的值都可设置,如图 2.12 所示。

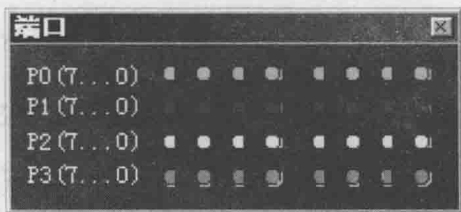


图 2.10 端口状态窗口