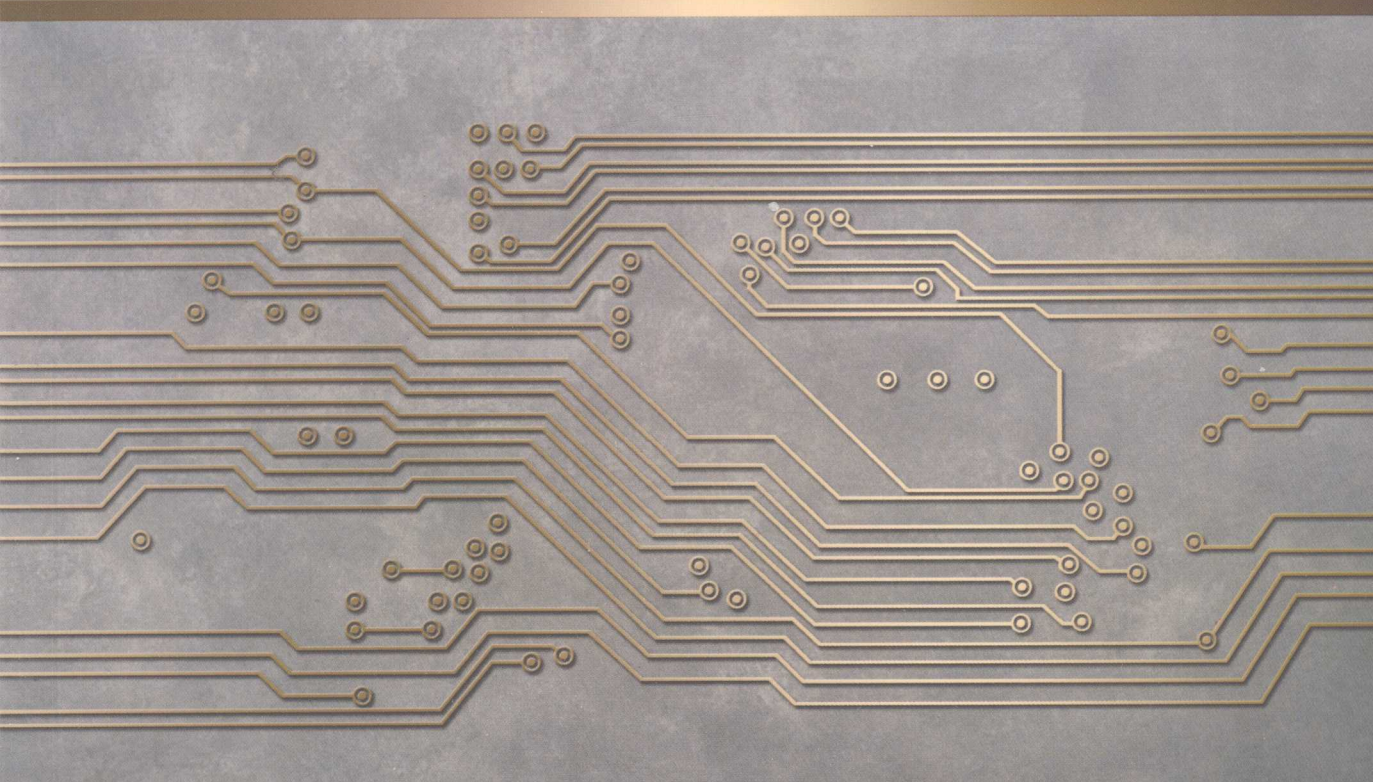


新编电气与电子信息类本科规划教材

单片机原理及应用

(第2版)

姜志海 黄玉清 刘连鑫 编著



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

新编电气与电子信息类本科规划教材

单片机原理及应用

(第2版)

姜志海 黄玉清 刘连鑫 编著

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书从教学的角度出发,系统、全面地介绍了 MCS-51 系列单片机的基本知识,是一本重在原理与应用、兼顾理论的实用教程。主要内容包括概述、MCS-51 系列单片机硬件结构、MCS-51 系列单片机指令系统、汇编语言程序设计、MCS-51 系列单片机硬件资源的应用、MCS-51 系列单片机系统扩展及接口技术、应用系统的设计、单片机的 C 语言设计。全书通过大量的例子阐述了单片机的基本问题,通过这些实例的学习,读者可以很容易地掌握单片机的基础知识。本书配有教学课件和配套的辅助文件。

本书既可以作为高等工科院校自动化、电气工程及其自动化、计算机应用、电子信息工程以及机电一体化等电气类专业教学用书,也可供从事单片机应用与产品开发工作的工程技术人员参考。

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

单片机原理及应用 / 姜志海, 黄玉清, 刘连鑫编著. —2 版. —北京: 电子工业出版社, 2009.8

(新编电气与电子信息类本科规划教材)

ISBN 978-7-121-09274-9

I. 单… II. ①姜… ②黄… ③刘… III. 单片微型计算机—高等学校—教材 IV. TP368.1

中国版本图书馆 CIP 数据核字 (2009) 第 120896 号

策划编辑: 王羽佳

责任编辑: 秦淑灵

印 刷: 涿州市京南印刷厂

装 订: 涿州市桃园装订有限公司

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: 16.75 字数: 440 千字

印 次: 2009 年 8 月第 1 次印刷

印 数: 4 000 册 定价: 28.80 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 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。

再版前言

单片机作为嵌入式微控制器，在工业测控系统、智能仪器和家用电器中得到广泛的应用。虽然单片机的品种很多，但 MCS-51 系列单片机仍不失为单片机中的主流机型。本书以 MCS-51 系列单片机为主介绍单片机的原理与应用，其特点是内容系统全面，论述深入浅出，循序渐进，注重接口技术和应用。

在本书的编写内容中，融入了编者多年教学、科研的经验与应用实例，从应用的角度出发，对单片机的硬件结构、工作原理、指令系统进行了简明扼要的介绍，对程序设计方法、系统扩展、接口电路的设计、应用系统等进行了详细的介绍，最后介绍了单片机的 C 语言程序设计的基本问题。

第 2 版仍然保持原来的写作风格，在内容上对原书进行了仔细的修订，增加了大量的例题，使一些叙述更加合理和通畅，更便于阅读和理解。

全书共分 9 章，主要内容包括概述、MCS-51 系列单片机的硬件结构、MCS-51 系列单片机的指令系统、汇编语言程序设计、MCS-51 系列单片机硬件资源的应用、MCS-51 系列单片机系统扩展技术、MCS-51 系列单片机接口技术、单片机应用系统的设计、单片机的 C 语言应用程序设计。

为了方便教学，本书配有**教学课件和配套辅助文件**，需要者可以到华信教育资源网 <http://www.hxedu.com.cn> 免费注册下载。

本书既可以作为高等工科院校自动化、电气工程及其自动化、计算机应用、电子信息工程以及机电一体化等电气类专业教学用书，也可供有关院校师生和有关从事单片机应用与产品开发等工作的工程技术人员参考。作为本科教材时，可在微机原理及应用的课程之后开设单片机原理及应用课程，参考学时为 48 学时。

本书由山东理工大学姜志海、西南科技大学黄玉清、山东理工大学刘连鑫编著。第 1, 2, 3, 5, 6 章由山东理工大学姜志海编写；第 4, 9 章由西南科技大学黄玉清编写；第 7, 8 章由山东理工大学刘连鑫编写。全书由姜志海负责整理和统稿。

本书在编写过程中得到了许多专家和同行的大力支持和热情帮助，他们对本书提出了许多建设性的建议和意见，在此一并表示衷心的感谢。

鉴于编者的水平有限，加之新的单片机芯片不断涌现，其应用技术也在不断发展，书中难免有不完善、不足之处，恳请广大读者批评指正。

编著者

2009 年 4 月

目 录

| | |
|-------------------------------------|----|
| 第 1 章 概述 | 1 |
| 1.1 微型计算机的基础知识概述 | 2 |
| 1.1.1 微型计算机的基本结构 | 2 |
| 1.1.2 计算机中的数制 | 5 |
| 1.1.3 计算机中常用的编码 | 7 |
| 1.2 单片机概述 | 8 |
| 1.2.1 单片机特点及应用 | 8 |
| 1.2.2 单片机技术现状及发展趋势 | 10 |
| 1.2.3 常用的单片机产品 | 12 |
| 1.2.4 单片机开发系统概述 | 14 |
| 本章小结 | 15 |
| 习题 | 16 |
| 第 2 章 MCS-51 系列单片机硬件结构 | 17 |
| 2.1 MCS-51 单片机的总体结构 | 18 |
| 2.1.1 主要组成 | 18 |
| 2.1.2 外部引脚说明 | 19 |
| 2.2 MCS-51 系列单片机的微处理器 | 21 |
| 2.2.1 运算器 | 22 |
| 2.2.2 控制器 | 23 |
| 2.3 MCS-51 单片机的存储器 | 24 |
| 2.3.1 程序存储器 | 24 |
| 2.3.2 数据存储器 | 25 |
| 2.4 MCS-51 单片机的辅助电路及时序 | 29 |
| 2.4.1 时钟电路 | 29 |
| 2.4.2 复位及复位电路 | 29 |
| 2.4.3 CPU 的时序 | 31 |
| 本章小结 | 33 |
| 习题 | 33 |
| 第 3 章 MCS-51 系列单片机指令系统 | 34 |
| 3.1 MCS-51 单片机指令系统概述 | 35 |
| 3.1.1 概述 | 35 |
| 3.1.2 指令格式 | 35 |
| 3.1.3 操作数的类型 | 35 |
| 3.1.4 指令描述约定 | 36 |
| 3.2 MCS-51 单片机的寻址方式 | 37 |
| 3.2.1 概述 | 37 |

| | |
|--|-----------|
| 3.2.2 寻址方式说明 | 37 |
| 3.3 MCS-51 单片机的指令系统 | 39 |
| 3.3.1 数据传送类指令 | 40 |
| 3.3.2 算术运算类指令 | 45 |
| 3.3.3 逻辑运算类指令 | 51 |
| 3.3.4 控制转移类指令 | 54 |
| 3.3.5 位操作类指令 | 59 |
| 3.4 MCS-51 单片机的伪指令 | 61 |
| 3.4.1 概述 | 61 |
| 3.4.2 伪指令说明 | 61 |
| 本章小结 | 64 |
| 习题 | 65 |
| 第 4 章 汇编语言程序设计 | 67 |
| 4.1 程序设计概述 | 68 |
| 4.1.1 程序设计步骤 | 68 |
| 4.1.2 程序设计技术 | 68 |
| 4.2 汇编语言基本程序设计 | 69 |
| 4.2.1 顺序结构程序 | 69 |
| 4.2.2 分支程序 | 71 |
| 4.2.3 循环程序 | 73 |
| 4.2.4 查表程序 | 77 |
| 4.2.5 子程序设计 | 78 |
| 4.3 算术逻辑处理程序 | 81 |
| 4.4 数制转化程序 | 87 |
| 本章小结 | 93 |
| 习题 | 94 |
| 第 5 章 MCS-51 系列单片机硬件资源的应用 | 95 |
| 5.1 MCS-51 单片机的并行口 | 96 |
| 5.1.1 概述 | 96 |
| 5.1.2 内部并行 I/O 口应用举例 | 99 |
| 5.2 MCS-51 单片机的中断系统 | 100 |
| 5.2.1 中断系统结构与控制 | 100 |
| 5.2.2 中断优先级结构 | 103 |
| 5.2.3 中断响应 | 103 |
| 5.2.4 中断请求的撤除 | 104 |
| 5.2.5 中断系统应用举例 | 106 |
| 5.3 MCS-51 单片机的定时器/计数器 | 109 |
| 5.3.1 定时器/计数器的结构 | 109 |
| 5.3.2 定时器/计数器的控制寄存器 | 110 |
| 5.3.3 定时器/计数器的工作方式 | 111 |

| | |
|--|------------|
| 5.3.4 定时器/计数器应用举例 | 115 |
| 5.4 MCS-51 单片机的串行接口 | 118 |
| 5.4.1 串行口寄存器的结构 | 118 |
| 5.4.2 串行口控制寄存器 | 119 |
| 5.4.3 串行口的工作方式 | 120 |
| 5.4.4 波特率的设定 | 122 |
| 5.4.5 串行口应用举例 | 123 |
| 本章小结 | 128 |
| 习题 | 128 |
| 第 6 章 MCS-51 系列单片机系统扩展技术 | 129 |
| 6.1 MCS-51 单片机系统扩展基础 | 130 |
| 6.1.1 系统扩展总线结构图 | 130 |
| 6.1.2 典型的存储器芯片 | 131 |
| 6.1.3 典型的 I/O 接口芯片 | 134 |
| 6.1.4 系统扩展的寻址方法 | 141 |
| 6.2 MCS-51 单片机存储器扩展技术 | 144 |
| 6.2.1 存储器扩展时有关信号的连接 | 144 |
| 6.2.2 存储器扩展举例 | 145 |
| 6.3 MCS-51 单片机并行 I/O 口的扩展技术 | 146 |
| 6.3.1 单片机的口线直接做 I/O 口 | 146 |
| 6.3.2 简单 I/O 接口的扩展 | 147 |
| 6.3.3 利用 8255A 可编程芯片扩展并行 I/O 口 | 149 |
| 6.4 串行存储器 E ² PROM 的扩展 | 150 |
| 6.4.1 概述 | 150 |
| 6.4.2 AT24CXX 系列串行 E ² PROM | 151 |
| 6.4.3 AT24C02 与单片机的接口实例 | 154 |
| 本章小结 | 158 |
| 习题 | 158 |
| 第 7 章 MCS-51 系列单片机接口技术 | 159 |
| 7.1 MCS-51 单片机键盘接口技术 | 160 |
| 7.1.1 概述 | 160 |
| 7.1.2 独立式按键接口技术 | 160 |
| 7.1.3 行列式键盘 | 161 |
| 7.2 MCS-51 单片机显示器接口技术 | 164 |
| 7.2.1 概述 | 164 |
| 7.2.2 LED 静态显示与动态显示 | 165 |
| 7.2.3 串行口控制的 LED 显示器与单片机的接口 | 168 |
| 7.2.4 键盘/显示接口综合设计举例 | 169 |
| 7.3 MCS-51 单片机 D/A 转换器接口技术 | 178 |
| 7.3.1 概述 | 178 |

| | | |
|--------------|------------------------------|------------|
| 7.3.2 | 8 位并行 D/A 转换器 DAC0832 | 179 |
| 7.3.3 | 12 位串行 D/A 转换器 DAC7512 | 182 |
| 7.3.4 | 串行电压输出型 D/A 转换器 | 185 |
| 7.4 | MCS-51 单片机 A/D 转换器接口技术 | 187 |
| 7.4.1 | 概述 | 187 |
| 7.4.2 | 并行 A/D 转换器 ADC0809 | 189 |
| 7.4.3 | 并行 A/D 转换器 AD574 | 192 |
| 7.4.4 | 串行 A/D 转换器 TLC2543 | 194 |
| 7.4.5 | V/F 式 A/D 转换电路的应用 | 197 |
| | 本章小结 | 198 |
| | 习题 | 199 |
| 第 8 章 | 单片机应用系统的设计 | 200 |
| 8.1 | 单片机应用系统结构以及设计内容 | 201 |
| 8.1.1 | 单片机应用系统的一般硬件组成 | 201 |
| 8.1.2 | 单片机应用系统的设计内容 | 203 |
| 8.2 | 单片机应用系统的一般设计方法 | 204 |
| 8.2.1 | 确定系统的功能与性能 | 204 |
| 8.2.2 | 确定系统基本结构 | 204 |
| 8.2.3 | 单片机应用系统硬件与软件设计 | 205 |
| 8.2.4 | 资源分配 | 207 |
| 8.3 | 单片机应用系统的调试 | 208 |
| 8.3.1 | 单片机应用系统调试工具 | 209 |
| 8.3.2 | 单片机应用系统的一般调试方法 | 209 |
| 8.4 | 单片机应用系统的设计实例 | 213 |
| | 本章小结 | 224 |
| | 习题 | 224 |
| 第 9 章 | 单片机的 C 语言应用程序设计 | 225 |
| 9.1 | 概述 | 226 |
| 9.2 | C51 的扩展 | 226 |
| 9.2.1 | C51 扩展数据类型 | 227 |
| 9.2.2 | 存储类型 | 228 |
| 9.2.3 | 存储器模式 | 230 |
| 9.2.4 | 指针 | 230 |
| 9.2.5 | 函数声明 | 232 |
| 9.3 | Keil 8051 开发工具简介 | 234 |
| 9.3.1 | Keil 8051 简介 | 234 |
| 9.3.2 | 一个工程的创建 | 235 |
| 9.4 | C51 应用程序设计 | 236 |
| 9.4.1 | 并行口应用 | 236 |
| 9.4.2 | 定时器应用 | 237 |

| | |
|----------------------------------|------------|
| 9.4.3 外中断应用 | 239 |
| 9.4.4 串行口应用程序 | 240 |
| 9.4.5 A/D 转换 | 248 |
| 9.4.6 智能小车寻线控制系统 | 249 |
| 本章小结 | 255 |
| 习题 | 255 |
| 附录 A ASCII 码字符表 | 256 |
| 附录 B 单片机应用资料的网上查询方法 | 257 |
| 参考文献 | 258 |

第 1 章 概 述

内容提要

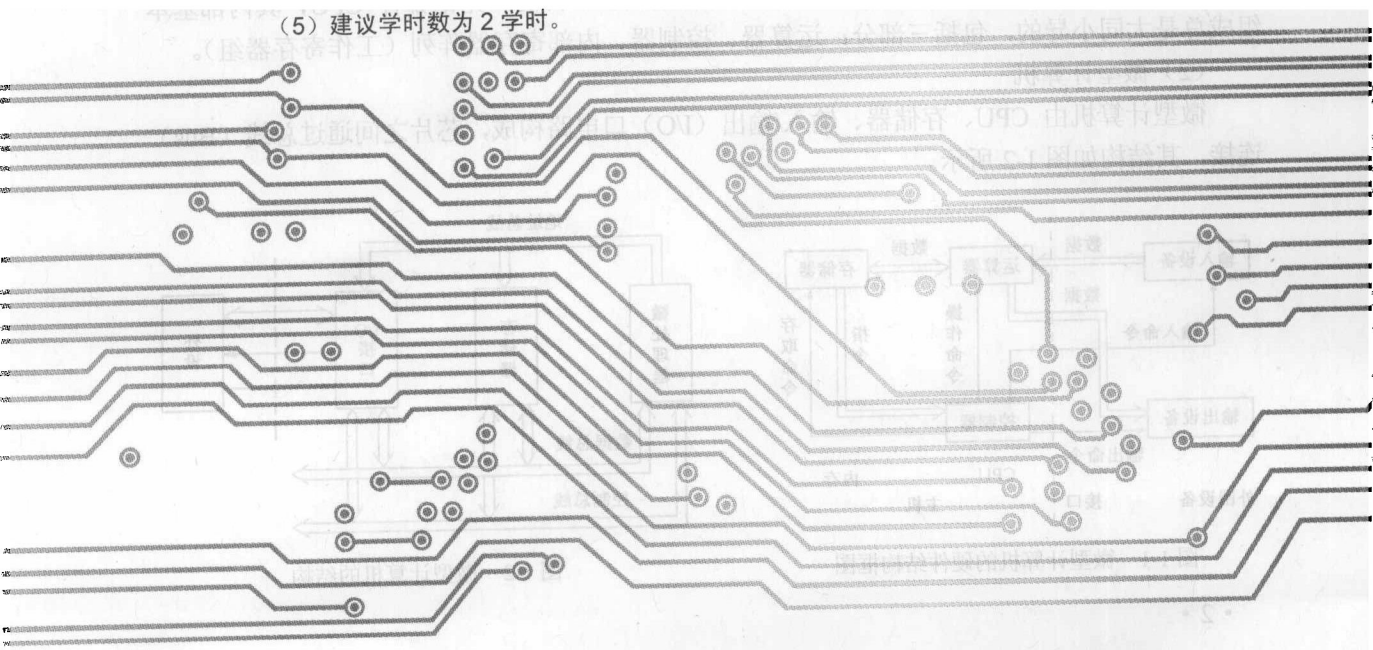
本章主要讲述了单片机的基本问题及学习单片机用到的基础知识。首先复习了微型计算机原理的基本问题，包括微型计算机硬件组成及软件、计算机中的数制等问题。然后介绍了单片机的基本问题，包括特点、应用、技术现状、发展趋势等内容。学完本章，使学生对单片机的基本问题有一个全面的了解，并能够和以前学过的微型计算机原理的基本问题进行结合，本章主要起到承上启下的作用。

知识要点

- (1) 单片机的特点、应用、技术现状、发展趋势；
- (2) 单片机应用系统的开发问题；
- (3) 微处理器、微型计算机、微型计算机系统；
- (4) 计算机中的数制。

教学建议

- (1) 了解单片机的特点、应用、技术现状、发展趋势；
- (2) 掌握微型计算机的结构及计算机中常用的数制；
- (3) 本章的基础知识，属于复习内容，主要起到承上启下的作用；
- (4) 可根据具体计划进行有选择的讲授；
- (5) 建议学时数为 2 学时。



1.1 微型计算机的基础知识概述

电子计算机是 20 世纪最重要的科学技术成就之一。目前计算机已渗透到国民经济和社会生活的各个领域，极大地改变着人们的工作方式和生活方式，并成为推动社会发展的巨大生产力。本节属于复习内容，主要对微型计算机的基础知识进行简单的论述。

1.1.1 微型计算机的基本结构

微型计算机由硬件和软件两大部分组成。硬件是指那些为组成计算机而有机联系的电子、电磁、机械、光学的元件、部件或装置的总和，它是有形的物理实体。软件是相对于硬件而言的，从狭义的角度看，软件包括计算机运行所需要的各种程序；而从广义的角度看，软件还包括手册、说明书和有关资料。

1. 微型计算机的硬件结构

微型计算机在硬件上由运算器、控制器、存储器、输入设备和输出设备五大部分组成，如图 1-1 所示。

运算器是计算机处理信息的主要部分；控制器控制计算机各部件自动、协调一致地工作；存储器是存放数据与程序的部件；输入设备用来输入数据与程序，常用的输入设备有键盘、光电输入机等；输出设备将计算机的处理结果用数字、图形等形式表示出来，常用的输出设备有显示终端、数码管、打印机、绘图仪等。

通常把运算器、控制器、存储器这三部分称为计算机的主机，而输入设备、输出设备则称为计算机的外围设备（简称“外设”）。由于运算器、控制器是计算机处理信息的关键部件，所以常将它们合称为中央处理单元 CPU（Central Process Unit）。

(1) 微处理器 CPU

微处理器是利用微电子技术将计算机的核心部件（运算器和控制器）集中做在一块集成电路上而形成的一个独立芯片。它具有解释指令、执行指令及与外界交换数据的能力。该芯片称为微处理器或微处理机（Microprocessor），也称 CPU。目前，无论哪种 CPU，其内部基本组成总是大同小异的，包括三部分：运算器、控制器、内部寄存器阵列（工作寄存器组）。

(2) 微型计算机

微型计算机由 CPU、存储器、输入/输出（I/O）口电路构成，芯片之间通过总线（Bus）连接，其结构如图 1-2 所示。

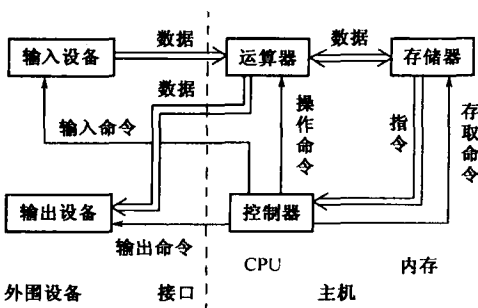


图 1-1 微型计算机的硬件结构框图

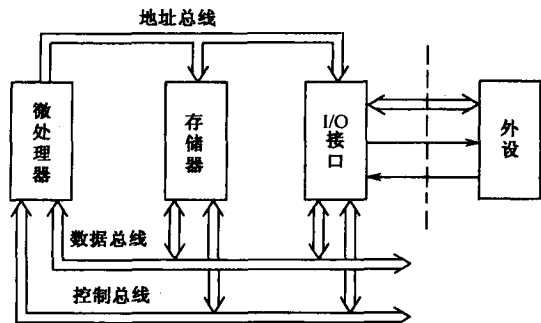


图 1-2 微型计算机的结构

① CPU: CPU 是微型计算机的核心, 它的性能决定了整个微型计算机的各项关键指标。微处理器本身不能构成独立工作的系统, 也不能独立执行程序, 必须配上存储器、外部输入/输出接口构成一台微型计算机才能工作。CPU 的基本知识前面已经说明了。

② 存储器: 存储器是微型计算机的重要组成部分, 用来存放程序和数据, 计算机有了存储器才具备记忆的能力。

从应用的角度讲, 计算机工作时, CPU 对存储器的操作只有“读”和“写”操作。CPU 将数据存入存储器的过程称为“写”操作, CPU 从存储器中取数据的过程称为“读”操作。写入存储单元的数据取代了原数据, 而且在下一个新的数据写入之前一直保留着, 即存储器具有记忆的功能。在执行“读”操作后, 存储单元中原有的内容不变, 即存储器的读出是非破坏性的。对存储器的操作是对存储器地址的操作。

③ 输入/输出接口电路: 输入/输出接口电路是微型计算机的重要组成部分, 是微型计算机连接外部输入、输出设备及各种控制对象并与外界进行信息交换的逻辑控制电路。从应用的角度讲, 计算机工作时, CPU 对 I/O 口的操作只有“读”和“写”。CPU 对输入口的操作称为“读”操作, 对输出口的操作称为“写”操作, 操作是对所选择的 I/O 口的口地址进行的。

④ 总线: 所谓总线, 就是在微型计算机各芯片之间或芯片内部各部件之间传输信息的一组公共通信线。微型计算机采用总线结构后, 芯片之间不需要单独走线, 这就大大减少了连接线的数量。采用总线结构后, 系统中各功能部件间的相互关系转变为各部件面向总线的单一关系, 符合总线标准的设备都可以连接到系统中, 使系统功能得到扩展。微型计算机总线的种类非常多, 从使用的角度可分为内部总线、元件级总线、系统总线、外部总线四大类, 在微型计算机中使用比较多的是元件级总线。计算机元件级总线包括地址总线 AB (Address Bus)、数据总线 DB (Data Bus)、控制总线 CB (Control Bus) 三种。

地址总线: 地址总线是 CPU 用来向存储器或 I/O 口传送地址信息的, 是三态单向总线。地址总线的宽度决定了 CPU 可直接寻址的内存容量。8 条地址线用 A7~A0 表示, A7 为最高位地址线, A0 为最低位地址线, 最大寻址范围为 $2^8 = 256$; 16 条地址线用 A15~A0 表示, A15 为最高位地址线, A0 为最低位地址线, 最大寻址范围为 $2^{16} = 65536 = 64K$ 。通过地址总线确定要操作的存储单元或 I/O 口的地址。

数据总线: 数据总线是 CPU 与存储器及外设交换数据的通路, 是三态双向总线。数据总线的位数与微处理器的位数相同, 一般有 8 位、16 位、32 位等。8 位数据线用 D7~D0 表示, D7 为最高有效位, D0 为最低有效位; 16 位数据线用 D15~D0 表示, D15 为最高有效位, D0 为最低有效位。最高有效位用 MSB 表示, 最低有效位用 LSB 表示。

控制总线: 控制总线是用来传输控制信号的, 传送方向依具体控制信号而定, 例如, CPU 向存储器或 I/O 接口电路输出读信号、写信号、地址有效信号, 而 I/O 接口部件向 CPU 输入复位信号、中断请求信号等。控制总线的宽度根据系统需要而定。

将微处理器、存储器、I/O 接口电路以及简单的输入、输出设备组装在一块印制电路板上, 称为单板微型计算机, 简称单板机。将微处理器、存储器、I/O 接口电路集成在一块芯片上, 称为单片微型计算机, 简称单片机。

(3) 微型计算机系统

以微型计算机为主体, 配上外部输入/输出设备、电源、系统软件一起构成应用系统, 称为微型计算机系统。

图 1-3 概括了微处理器、微型计算机、微型计算机系统三者的关系。

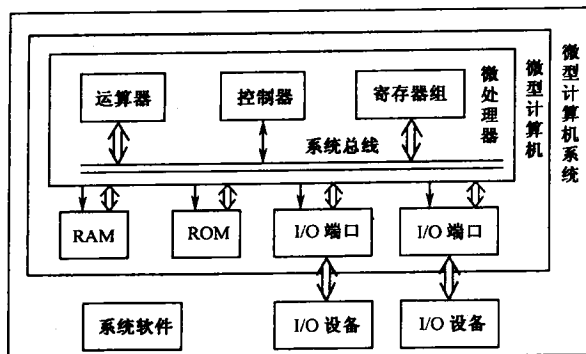


图 1-3 微型计算机系统结构

2. 微型计算机的软件

上述的微型计算机设备称为硬件。计算机要能够自动地操作与运算，还必须要有软件。软件是指使用和管理计算机的各种程序（Program），而程序是完成任务所需要的一系列指令（Instruction）序列。程序的集合构成了计算机中的软件系统。

(1) 指令与程序

控制计算机完成各种操作的命令称为指令。

为了计算一个数学式，或者控制一个生产过程，需要事先制订计算机的计算步骤或操作步骤。计算步骤是由一条条指令来实现的。这种一系列指令的有序集合称为程序。编制程序的过程称为程序设计。

例如，计算 $63+56+36+14=?$

编制的程序如下：

```

MOV    A, #63      ; 数 63 送入寄存器 A
ADD    A, #56      ; A 的内容 63 与数 56 相加，其和 119 送回 A
ADD    A, #36      ; A 的内容 119 与数 36 相加，其和 155 送回 A
ADD    A, #14      ; A 的内容 155 与数 14 相加，其和 169 保存在 A 中
    
```

为了使机器能自动进行计算，要预先用输入设备将上述程序输入到计算机存放。计算机启动后，在控制器的控制下，CPU 按照顺序依次取出程序的一条条指令，加以译码和执行。程序中的加法操作是在运算器中进行的。运算结果可以保存在 A 中，也可以通过输出设备从计算机中输出。

如上所述，计算机的工作是由硬件、软件紧密结合、共同完成的，这与一般的数字电路系统不同。

表 1-1 汇编语言和机器语言对照表

| 汇编语言 | 机器语言 |
|------------|------------------------|
| MOV A, #63 | 0111 0100 0011 1111 |
| ADD A, #56 | 0010 0100 0011 1000 |
| ADD A, #36 | 0010 0100 0010 0100 |
| ADD A, #14 | 0010 0100 0000 1110 |

(2) 机器语言、汇编语言和高级语言

编制程序可使用汇编语言或高级语言。

计算机中只能存放和处理二进制信息，所以无论高级语言程序还是低级语言程序，都必须转换成二进制代码形式后才能送入计算机，这种二进制代码形式的程序就是机器语言程序。二进制代码形式的指令又称为机器指令或机器码。汇编指令与机器指令具有一一对应的关系。表 1-1 是用汇编语言和机器语言编写的同一段程序。

(3) 汇编、编译与解释程序

汇编语言程序与高级语言程序统称为源程序，而机器语言程序称为目标程序。

机器语言只有 0, 1 两个符号，用它来直接编写程序十分困难。因此，往往先用汇编语言或高级语言编写程序，然后再转换成目标程序。将汇编语言程序翻译成目标程序的过程称为汇编，汇编时用到的软件称为汇编程序。高级语言转换成机器语言的工作只能由计算机完成，转换时所用的软件称为编译程序或解释程序。

例如，把表 1-1 所示的目标程序存入容量为 256 个单元的存储器，且从地址为 0000 0000 的单元开始存放，如图 1-4 所示。

指令机器码第 1 个字节所在单元的地址 (0000 0000, 0000 0010, 0000 0100, 0000 0110) 称为指令地址。第 1 条指令的地址 (0000 0000) 称为该程序的首地址，又称程序的入口地址，带有二进制地址和机器码的程序示例如表 1-2 所示。

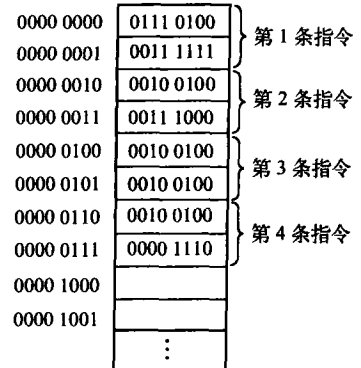


图 1-4 存储器中的程序

表 1-2 二进制地址和机器码的程序示例

| 地 址 | 机 器 码 | 汇 编 指 令 |
|-----------|---------------------|------------|
| 0000 0000 | 0111 0100 0011 1111 | MOV A, #63 |
| 0000 0010 | 0010 0100 0011 1000 | ADD A, #56 |
| 0000 0100 | 0010 0100 0010 0100 | ADD A, #36 |
| 0000 0110 | 0010 0100 0000 1110 | ADD A, #14 |

二进制位数多，读写不便，所以地址和机器码实际上多以十六进制数表示。

3. 硬件和软件的关系

微机系统是硬件和软件有机结合的整体。计算机的硬件和软件是密不可分又相互独立的。硬件是计算机工作的基础，没有硬件的支持，软件将无法正常工作；软件是计算机的灵魂，没有软件，硬件就是一个空壳。没有软件的计算机称为裸机，裸机如同一架没有思想的躯壳，不能做任何工作。

1.1.2 计算机中的数制

计算机的工作过程就是对数据的处理。计算机是一个典型的数字化设备，它只能识别 0 和 1，所有的计算机都是以二进制数的形式进行算术运算和逻辑操作的。

1. 计算机中的数制

在使用微型计算机时常用的进位计数制有二进制、十六进制、十进制三种。十进制数、二进制数及十六进制数之间的关系如表 1-3 所示。

表 1-3 十进制数、二进制数及十六进制数对照表

| | | | | | | | | | | | | | | | | |
|-------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| 十进制数 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| 二进制数 | 0000 | 0001 | 0010 | 0011 | 0100 | 0101 | 0110 | 0111 | 1000 | 1001 | 1010 | 1011 | 1100 | 1101 | 1110 | 1111 |
| 十六进制数 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |

为了区别十进制、二进制及十六进制三种数制，可以在数的后面加一字母。规定 B (Binary) 表示二进制数，D (Decimal) 表示十进制数，H (Hexadecimal) 表示十六进制数，其中十进制数后面的字母 D 可以省略。

2. 带符号数的表示

一个数据是带符号的数还是不带符号的数，事先是已知的。在微型计算机中所有带符号的数都是用补码表示的，采用补码的目的在于，可用加法运算代替减法运算，从而简化硬件结构，降低成本。补码运算时符号位不需要单独处理，符号位与数值部分一起参加运算，在不发生溢出的情况下，运算结果（包括符号位）是正确的。

8 位二进制数用来表示无符号数、有符号数（原码、反码和补码），如表 1-4 所示。

从表 1-4 可以看出，8 位二进制数、无符号数表示范围是 0~255，有符号数原码表示范围是 -127~+127，反码表示范围是 -127~+127，补码表示范围是 -128~+127。

表 1-4 原码、反码和补码表

| 二进制数 | 无符号数 | 有符号数 | | |
|-----------|------|------|------|------|
| | | 原 码 | 反 码 | 补 码 |
| 0000 0000 | 0 | +0 | +0 | +0 |
| 0000 0001 | 1 | +1 | +1 | +1 |
| 0000 0010 | 2 | +2 | +2 | +2 |
| ... | | | | |
| ... | | | | |
| 0111 1110 | 126 | +126 | +126 | +126 |
| 0111 1111 | 127 | +127 | +127 | +127 |
| 1000 0000 | 128 | -0 | -127 | -128 |
| 1000 0001 | 129 | -1 | -126 | -127 |
| ... | | | | |
| ... | | | | |
| 1111 1101 | 253 | -125 | -2 | -3 |
| 1111 1110 | 254 | -126 | -1 | -2 |
| 1111 1111 | 255 | -127 | -0 | -1 |

3. 带符号数溢出及其判断方法

(1) 什么是溢出

在微型计算机中，所有带符号的数都是用补码表示的。所谓溢出，是指带符号数的补码加、减运算的结果超出了补码表示的范围。若发生了溢出，则带符号数的运算结果必然是错误的。

(2) 判断溢出的方法

溢出只能出现在两个同符号数相加或两个异符号数相减的情况下。利用双进位方法判断有无溢出是一种常用的方法，它的规则如下。

在两个同符号数相加或两个异符号数相减时，如果次高位向最高位有进位（或借位），而最高位向前无进位（或借位），则结果发生溢出；反过来，如果次高位向最高位无进位（或借位），而最高位向前有进位（或借位），则结果也发生溢出。

这种方法是利用最高位和次高位的进位/借位状态进行“异或”来判断的。对 8 位二进制数来说，如果溢出记为 OF，最高位和次高位的进位/借位状态分别记为 C7 和 C6，则 $OF = C7 \oplus C6$ ，OF = 1 表示有溢出，OF = 0 表示无溢出。

在微型计算机中，可用多字节表示更大的数，避免产生溢出错误。

1.1.3 计算机中常用的编码

计算机除了用于数值计算外，还要进行大量的文字信息处理，也就是要对表达各种文字信息的符号进行加工。例如，计算机和外设的键盘、(字符)显示器、打印机之间的通信都是采用字符方式输入/输出的。目前，计算机中最常用的两种编码是美国信息交换标准代码(ASCII码)和二-十进制编码(BCD码)。

1. 美国信息交换标准代码(ASCII码)

ASCII(American Standard Code for Information-Interchange)码是美国信息交换标准代码的简称，主要用来给西文字符进行编码。它采用7位二进制数表示一个字符，包括32个标点符号，10个阿拉伯数字，52个英文大、小写字母和34个控制符号，共128个。编码与字符之间的对应关系如附录A所示。

在计算机系统中，存储单元的长度通常为8位二进制数(即一个字节)，为了存取方便，规定一个存储单元存放一个ASCII码，其中低7位表示字母本身的编码，第8位(即bit7)用做奇偶校验位或规定为零(通常如此)。因此，也可以认为ASCII码的长度为8位。例如：

“8”的奇校验ASCII码为00111000B，偶校验ASCII码为10111000B；

“B”的奇校验ASCII码为11000010B，偶校验ASCII码为01000010B。

2. BCD码(二进制编码的十进制数)

十进制毕竟是人们最习惯的计数方式，在向计算机输入数据时，常用十进制数输入，但计算机只识别二进制数，因此每1位十进制数必须用二进制数表示。1位十进制数包含0~9个数码，必须用4位二进制数表示，这样就需要确定0~9与4位二进制数0000B~1111B之间的对应关系，其中较常用的8421BCD码规定了十进制数0~9与4位二进制数编码之间的对应关系，见表1-5。

表 1-5 十进制数 0~9 与 4 位二进制数编码之间的对应关系

| 十进制数 | 8421BCD码 | 十进制数 | 8421BCD码 |
|------|----------|------|----------|
| 0 | 0000 | 5 | 0101 |
| 1 | 0001 | 6 | 0110 |
| 2 | 0010 | 7 | 0111 |
| 3 | 0011 | 8 | 1000 |
| 4 | 0100 | 9 | 1001 |

注：在BCD码中，不使用1010B(0AH)~1111B(0FH)。

计算机中存储BCD码有两种形式：压缩BCD码和非压缩BCD码。

(1) 压缩BCD码

压缩BCD码用4位二进制数表示1位十进制数，一个字节可以表示2位十进制数。例如，10010111B表示十进制数97。

(2) 非压缩BCD码

非压缩BCD码用8位二进制数表示1位十进制数，高4位总为0000，低4位的0000~1001表示0~9。例如，00001001B表示十进制数9。

前面介绍了在使用计算机时二进制数、十进制数、十六进制数、ASCII 码、BCD 码以及带符号数的表示等问题，这里要注意微型计算机能处理的数据只有二进制数，计算机并不认识什么正数、负数、BCD 码、ASCII 码等，计算机中数的表现形式只有二进制数，其他的数制和性质需要人们来进行分析与说明。例如，在某存储器中存放一个二进制数 11111111B (0FFH)，这个数有多大？这要看人们如何看了，如果是一个无符号数，就是 255；如果是一个有符号数，就是-1；如果是个 BCD 码，就是一个无效的数；如果是个 ASCII 码，就代表“DEL”键的 ASCII 码值。

1.2 单片机概述

单片机又称为微控制器，它采用一定的工艺手段将 CPU、存储器和 I/O 口集成在一个芯片上，其发展十分迅速。自 1975 年美国得克萨斯仪器公司 (Texas Instruments) 第一块微型计算机芯片 TMS-1000 问世以来，在短短的 20 年间，单片机技术已发展成为计算机领域一个非常有前途的分支，它有自己的技术特征、规范、发展道路和应用领域。单片机是为了满足工业控制需要而诞生的，是自动控制系统的核心部件，因而也主要用于工业控制、智能化仪器仪表、家用电器中。它具有体积小、个性突出（某些方面的性能指标大大优于通用微机中央处理器）、价格低廉等特点，应用领域不断扩大，除了工业控制、智能化仪表、通信、家用电器外，在智能化高档电子玩具产品中也大量采用单片机芯片作为核心控制部件。

1.2.1 单片机特点及应用

在通用微机中央处理器基础上，将输入/输出 (I/O) 接口电路、时钟电路以及一定容量的存储器等部件集成在同一芯片上，再加上必要的外围器件，如晶体振荡器，就构成了一个较为完整的计算机硬件系统。由于这类计算机系统的基本部件均集成在同一芯片内，因此被称为单片微控制器 (Single-Chip-Micro Controller, 简称单片机) 或微控制单元 (Micro Controller Unit, 简称 MCU)。

对于通用微处理器来说，其主要任务是数值计算和信息处理，对运算速度和存储容量方面的要求是速度越快越好，容量越大越好，因此它朝着高速、大容量方向发展：字长由 8 位（如 8085 处理器）、16 位（如 8086 处理器、80286 处理器），迅速向 32 位（如 80486 处理器）、64 位（如 Pentium 系列 CPU，Pentium 系列 CPU 内部数据总线为 32 位，对外数据总线为 64 位，因而 Pentium 还不是真正意义上的 64 位微处理器）过渡，时钟信号的频率由最初的 4.77 MHz 向 33 MHz，66 MHz，100 MHz，200 MHz，400 MHz，600 MHz，1 GHz，2 GHz，甚至更高频率过渡。而单片机主要面向工业控制，8 位字长已足够（在工业控制中，一般仅需要控制线路的通、断，触点的吸合与释放，有时 4 位单片机也能胜任），尽管也有 16 位、32 位的单片机芯片，但这些高档单片机芯片主要用于语音、图像处理系统。

1. 单片机的特点

单片机芯片作为控制系统的核心部件，除了具备通用微机 CPU 的数值计算功能外，还必须具有灵活、强大的控制功能，以便实时监测系统的输入量、控制系统的输出量，实现自动控制。由于单片机主要面向工业控制，工作环境比较恶劣，如高温、强电磁干扰，甚至含有腐蚀性气体，在太空中工作的单片机控制系统，还必须具有抗辐射能力，这决定了单片机 CPU 与通用微机 CPU 具有不同的特点。