



普通高等教育“十三五”规划教材

单片微机原理与 接口技术

Principle and Interface Technology of
Single Chip Microcomputer

◎ 主编 郑洪庆 安玲玲 程蔚





Principle and Interface Technology of
Single Chip Microcomputer



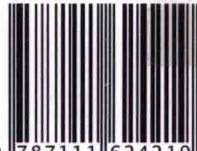
机工教育微信服务号

ISBN 978-7-111-62421-9

策划编辑◎王雅新 / 封面设计◎张静



ISBN 978-7-111-62421-9



9 787111 624219 >

定价：39.80元

普通高等教育“十三五”规划教材
福建省本科高校重大教育教学改革研究项目（FBJG20170332）
泉州市高等学校中青年学科（专业）带头人培养计划

单片微机原理与接口技术

主 编 郑洪庆 安玲玲 程 蔚
参 编 章 玲 黄毓芯 林木泉
陈双燕 汤巧治 蔡 凡



机械工业出版社

本书围绕着 51 内核单片机的硬件结构、C 语言编程技巧和应用系统开发,采用任务驱动与项目实训的方式,以理论够用、注重应用为原则,激发学生的学习兴趣,逐步提高学生的单片机外围接口电路设计和编写程序的逻辑思维能力。本书的实训项目由简单的如何点亮 LED 灯项目开始,引导读者入门,逐步扩展知识面,逐步增加项目难度,使读者积累项目开发的经验,提高实践动手能力、分析与解决问题的能力,掌握单片机系统开发技能。

本书层次结构合理,叙述简明易懂,参编人员有着多年的实际项目开发经验。本书项目来自科研、竞赛、教学实践等,实用性强,提供实训项目代码,还可以提供配套的单片机开发板。

本书可作为高等院校电子信息、计算机、电气工程、自动化、机电一体化、仪器仪表等专业的单片机课程教材,也适合于单片机初学者和从事单片机应用的工程技术人员参考。本书还可以作为电子设计竞赛、蓝桥杯电子类比赛的培训教材。

图书在版编目 (CIP) 数据

单片微机原理与接口技术/郑洪庆,安玲玲,程蔚主编. —北京:机械工业出版社,2019.6

普通高等教育“十三五”规划教材

ISBN 978-7-111-62421-9

I. ①单… II. ①郑… ②安… ③程… III. ①单片微型计算机-基础理论-高等学校-教材②单片微型计算机-接口-高等学校-教材
IV. ①TP368.1

中国版本图书馆 CIP 数据核字 (2019) 第 061584 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:王雅新 责任编辑:王雅新 王小东

责任校对:肖琳 封面设计:张静

责任印制:郜敏

河北宝昌佳彩印刷有限公司印刷

2019 年 8 月第 1 版第 1 次印刷

184mm × 260mm · 15.25 印张 · 371 千字

标准书号: ISBN 978-7-111-62421-9

定价: 39.80 元

电话服务

客服电话: 010-88361066

010-88379833

010-68326294

封底无防伪标均为盗版

网络服务

机工官网: www.cmpbook.com

机工官博: weibo.com/cmp1952

金书网: www.golden-book.com

机工教育服务网: www.cmpedu.com

前 言

单片机是工科类本科生教学的主要学科基础课之一，是电子信息工程、通信工程、电气工程、自动化、计算机等专业的一门重要的专业基础课。

目前，在市场上的众多产品中均能看到单片机的身影。单片机以其高性价比、高速度、体积小、可重复编程和方便功能扩展等优点，在实践中得到了广泛应用。目前，51 单片机 C 语言（简称 C51 语言）的使用越来越广泛，大有取代汇编语言的趋势。学习并熟练掌握 C51 语言，对于单片机的系统设计和程序开发都非常有用。

本书为“福建省本科高校重大教育教学改革研究项目（FBJG20170332）、泉州市高等学校中青年学科（专业）带头人培养计划、福建省本科高校重大教育教学改革研究项目（JZ160227）、闽南理工学院校级应用型课程改革项目”的研究成果。本书特色如下：

1. 内容全面，由浅入深

本书涵盖了 C51 语言程序设计所需掌握的各方面知识点。首先详细介绍了 51 系列单片机的基础知识，包括集成开发环境和开发流程。然后对 C51 语言程序设计基础知识结合实例进行全面详细的介绍，包括数据类型与结构、函数、存储结构等内容。接着对 C51 语言的程序设计方式进行了详细的讲解，包括定时计数器、中断设计、串行接口设计等内容。最后介绍了电子设计各个领域具有代表性的案例，包括键盘设计、总线接口、单片机通信，以及 A/D 转换等方面的内容。最后设计了一个完整的综合应用实例。

2. 结合实例，强理解

本书在介绍每个相关知识点的同时，均给出了其在程序设计中的编程示例，每个例子都可以执行，读者可以在学习独立知识点的同时，根据应用示例举一反三，快速掌握相应知识点在整个程序设计系统中的实际应用。

3. 联系硬件，切合需求

本书不仅介绍 51 单片机的 C 语言，还对单片机的硬件资源，以及如何使用 C51 语言来编程控制单片机的各种片上资源进行了详细介绍，主要包括单片机定时器/计数器、中断、USART 串行通信接口、EEPROM、SPI 串行通信接口、PWM 脉宽调制和 A/D 转换。

4. 案例丰富，分析全面

本书案例丰富，基本上涵盖了电子设计的各个领域，如键盘接口、RS-232 通信、SPI 总线、掉电参数保护、电动机控制、电压检测等。本书对每一个案例都详细介绍了相关的背景知识、硬件知识、电路设计、程序设计，以及扩展思考等内容，并对整体程序代码按功能分块进行详细的注释，更加易于读者理解。

本书由郑洪庆、安玲玲、程蔚担任主编，郑洪庆负责统稿。蔡凡编写第 1 章，

黄毓芯编写第5、8章，章玲编写第6、7章，陈双燕编写第9章，林木泉编写第10、16章，程蔚编写第12、13章，汤巧治编写第11章，安玲玲编写第14、15章，其余内容由郑洪庆编写。

由于编者水平有限，书中肯定存在错误和不足之处，敬请各位同仁不吝批评指正。

编 者

目 录

前言	
第 1 章 单片机入门——基础必备	
知识	1
1.1 单片机概述	1
1.1.1 什么是单片机?	1
1.1.2 单片机应用领域	2
1.1.3 如何学好单片机	3
1.2 数制与编码	3
1.2.1 数制	3
1.2.2 数制之间的转换	4
1.2.3 常用的编码	5
1.3 微型计算机的基本组成和工作过程	6
1.3.1 基本组成	6
1.3.2 指令、程序与编程语言	7
1.3.3 工作过程	8
1.4 开发软件环境搭建	9
1.5 开发板功能简介	13
本章小结	14
实训项目	14
第 2 章 点亮 LED 灯——Keil 软件与	
单片机 I/O	15
2.1 Keil μ Vision4 使用方法	15
2.1.1 Keil μ Vision4 工作界面	15
2.1.2 Keil 工程的建立	15
2.2 单片机最小系统	20
2.2.1 电源	20
2.2.2 复位电路	21
2.2.3 晶振	21
2.3 时钟周期、机器周期和指令周期	22
2.4 点亮 LED 灯	23
2.4.1 LED (发光二极管)	23
2.4.2 特殊功能寄存器和位定义	23
2.4.3 编写程序	25
2.5 程序下载	26
2.6 74HC573 锁存器	29
2.7 74HC02 或非门	30
2.8 74HC138 三八译码器	31
2.9 单片机资源扩展方式	34
本章小结	36
实训项目	36
第 3 章 经典再现——C 语言基础	37
3.1 C51 程序开发	37
3.1.1 采用 C51 的优点	37
3.1.2 C51 程序开发过程	37
3.1.3 C51 程序结构	37
3.2 C51 语言的数据类型	38
3.3 C51 语言的运算符	39
3.4 C51 语言的控制语句	41
3.4.1 if 语句	41
3.4.2 switch 语句	44
3.4.3 while 语句	46
3.4.4 for 语句	48
3.4.5 中断语句 break/continue	51
3.5 C51 函数	53
3.5.1 函数定义	54
3.5.2 函数的调用	54
3.5.3 中断服务函数	55
3.6 程序划分为多个文件	57
3.6.1 共享宏定义和类型定义	57
3.6.2 共享函数原型	57
3.6.3 共享变量声明	58
3.6.4 保护头文件	58
本章小结	58
实训项目	59
第 4 章 流水灯实现——C51 编程	60
4.1 设计思路	60
4.2 软件延时	61
4.3 程序实现	62
本章小结	64
实训项目	64
第 5 章 计数器——数码管显示与	
独立按键	65
5.1 数码管的显示原理	65

5.2 数码管的真值表与静态显示	66	9.1.1 设计要求	116
5.3 独立按键检测	68	9.1.2 硬件电路分析	116
5.4 计数器实现	70	9.1.3 程序设计	119
5.5 数码管动态显示	72	9.2 PWM 基础知识	123
5.5.1 C 语言数组回顾	72	9.3 直流电动机 PWM 调速	123
5.5.2 动态显示原理	73	9.3.1 直流电动机简介	123
5.5.3 程序实现	74	9.3.2 直流电动机恒速运行	124
5.5.4 数码管显示消隐	75	9.3.3 直流电动机调速	127
本章小结	76	9.4 PWM 调光	129
实训项目	76	本章小结	132
第 6 章 数字秒表——中断系统及 定时器	77	实训项目	132
6.1 中断系统	77	第 10 章 数据传输——串口通信	133
6.2 中断系统处理过程	78	10.1 串口通信基础	133
6.2.1 51 内核单片机的中断结构	78	10.1.1 基本通信方式及特点	133
6.2.2 单片机的中断源	78	10.1.2 串行通信数据传送方式	133
6.2.3 中断响应及处理过程	81	10.1.3 串行通信的传输方向	134
6.3 定时器的结构和工作原理	83	10.1.4 串行通信的传输速率	134
6.4 定时器的寄存器	84	10.2 单片机与 PC 常见通信接口	135
6.4.1 T0、T1 的方式寄存器 TMOD	84	10.3 串口结构与工作原理	137
6.4.2 T0、T1 的控制寄存器 TCON	87	10.4 波特率与定时器	138
6.5 定时器的应用	87	10.5 编程与实现	139
本章小结	91	10.6 ASCII 码	142
实训项目	92	本章小结	143
第 7 章 频率计——定时器进阶	93	实训项目	143
7.1 NE555	93	第 11 章 温度计——DS18B20 温度 传感器应用	144
7.2 方波频率的测量	94	11.1 DS18B20 温度传感器介绍	144
7.3 定时器计数模式	95	11.1.1 DS18B20 主要特性	144
7.4 频率计实现	95	11.1.2 DS18B20 应用电路原理图	145
本章小结	99	11.2 DS18B20 工作原理	147
实训项目	99	11.2.1 DS18B20 的通信协议	147
第 8 章 简易加法计算器——矩阵按键与 函数进阶	100	11.2.2 DS18B20 单总线通信过程	151
8.1 函数的调用	100	11.3 DS18B20 应用实例	152
8.2 形参与实参	103	11.3.1 DS18B20 的测温与显示——整数 显示	152
8.3 矩阵按键扫描	104	11.3.2 DS18B20 的测温与显示——带 1 位 小数显示	156
8.3.1 按键消抖	104	本章小结	158
8.3.2 矩阵按键的识别与编码	105	实训项目	158
8.4 简易加法计算器的实现	108	第 12 章 记录开机次数——IIC 总线与 EEPROM (AT24C02) 应用	159
本章小结	114	12.1 IIC 总线基础	159
实训项目	115		
第 9 章 知识沉淀——交通灯设计和 PWM 控制	116		
9.1 交通灯实现	116		

12.2 IIC 寻址模式	161	15.1 LCD1602 模块的外形及引脚	203
12.3 IIC 总线时序模拟	162	15.2 LCD1602 模块的组成	204
12.4 AT24C02 操作 (写和读操作)	165	15.3 LCD1602 模块的命令	205
12.5 记录开机次数实现	167	15.4 电子万年历	206
本章小结	170	15.4.1 设计任务	206
实训项目	171	15.4.2 硬件电路分析	207
第 13 章 光照强度检测——A/D 与 D/A		15.4.3 程序设计	207
(PCF8591 应用)	172	本章小结	222
13.1 A/D 和 D/A 的基本概念	172	实训项目	222
13.2 ADC 的主要指标	172	第 16 章 综合应用设计	223
13.3 PCF8591 与单片机的接口	173	16.1 模拟风扇控制系统设计	223
13.4 PCF8591 程序实现	174	16.2 智能物料传送系统设计	224
13.5 光照强度检测	178	本章小结	226
13.6 D/A 输出	181	实训项目	226
本章小结	185	第 17 章 单片机应用系统设计方法	227
实训项目	185	17.1 单片机应用系统设计过程	227
第 14 章 电子时钟——DS1302 应用 ..	186	17.1.1 系统设计的基本要求	227
14.1 DS1302 的基础知识	186	17.1.2 系统设计的步骤	228
14.2 DS1302 芯片简介	186	17.2 提高系统可靠性的一般方法	229
14.3 DS1302 显示时钟的实例	188	17.2.1 电源干扰及其抑制	229
14.3.1 电子时钟基础	188	17.2.2 地线干扰及其抑制	230
14.3.2 电子时钟进阶——带时间调整、		17.2.3 其他提高系统可靠性的方法	230
闹铃功能的电子时钟设计	194	本章小结	231
本章小结	202	实训项目	231
实训项目	202	附录 常用 ASC II 码表	232
第 15 章 电子万年历——LCD1602 液晶		参考文献	233
显示器的应用	203		

第1章 单片机入门——基础必备知识

教学目标

1. 通过本章的学习，使学生了解单片机的组成。
2. 掌握单片机开发过程中的数制转换。
3. 掌握单片机开发环境搭建、单片机的学习方法。

重点内容

1. 数制转换。
2. 单片机开发环境搭建。

1.1 单片机概述

1.1.1 什么是单片机？

介绍单片机之前，先了解一下计算机（computer）。由硬件系统和软件系统所组成，没有安装任何软件的计算机称为裸机。硬件系统包括电源、主板、CPU、内存、硬盘、声卡、网卡、显卡、光驱、键盘、鼠标、显示器等。

单片机（micro-controller）属于一种特殊的计算机，它是把一台计算机的多种功能集成到了一块芯片里。简单说，单片机就是一种集成电路芯片，把中央处理器（CPU）、随机存储器（RAM）、只读存储器（ROM）、多种输入/输出（I/O）口等功能集成到一块硅片上构成的一个微型计算机系统。表1-1为单片机与计算机的对比。

表1-1 单片机与计算机对比

序号	单片机		计算机
1	CPU	相当于	CPU
2	RAM		内存
3	ROM		硬盘
4	多种 I/O 口		声卡、网卡、显卡、光驱、键盘、鼠标

单片机在工业领域有着广泛的应用，通过学习掌握单片机的编程语言，编写不同的程序，让它能按照工程师的想法完成从各个引脚上发出不同的高、低电平信息，完成不同的输入、输出控制要求，代替人完成各种智能控制任务。可以采用C语言和汇编语言编程，建议采用C语言编程。

本书主要讲解比较基础的51内核的单片机。世界上不同芯片厂家生产的单片机种类很多，如51单片机、STM32单片机、AVR单片机、PIC单片机等。部分厂家生产的51内核单片机如表1-2所示。

表 1-2 部分厂家生产的 51 内核单片机

公司	产品
AT (Atmel)	AT89C51、AT89C52、AT89C53、AT89S51、AT89S52、AT89LS53 等
STC	STC89C51RC、STC89C52RC、STC89C53RC 等
Winbond (华邦)	W78C54、W78C58、W78E54、W78E58 等

由于厂家及芯片型号较多，这里不一一列出。51 内核单片机是最基础的单片机，学会该系列单片机，再学其他高级的单片机就比较轻松，基本思路和方法是一样的。

本书重点讲解 STC89C52RC 单片机，如图 1-1 所示。

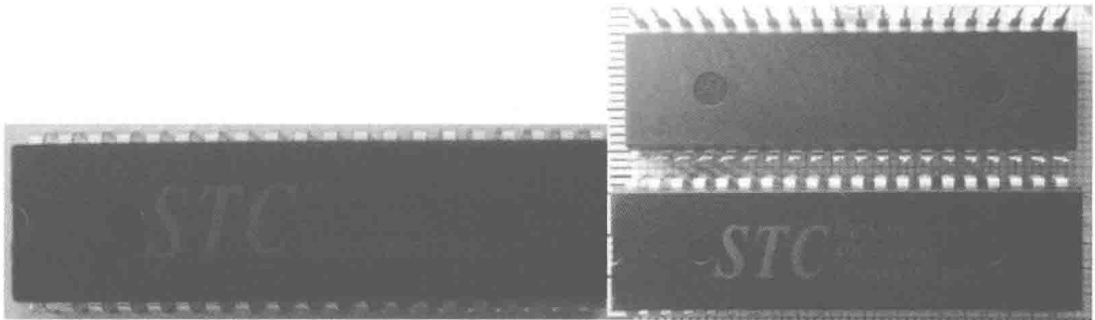


图 1-1 单片机实物图

其标识含义如下：

STC——前缀，表示芯片为 STC 公司生产的产品。其他前缀还有如：AT、I、Winbond、SST 等。

8——表示该芯片为 8051 内核芯片。

9——表示内部含 Flash、E²PROM 寄存器。还有如 80C51 中 0 表示内部含有 Mask Rom (掩模 ROM) 存储器；又如 87C51 中 7 表示内部含有 EPROM 存储器（紫外线可擦除 ROM）。

C——表示该器件为 CMOS 产品。还有如 89LV52 和 89LE58 中的 LV 和 LE 都表示该芯片为低电压产品（通常为 3.3V 供电）；而 89S52 中 S 表示该芯片含有可串行下载功能的 Flash 存储器，即具有 ISP 可在线编程功能。

5——固定不变。

2——表示该芯片内部程序存储空间大小。1 为基本型 4KB，2 为加强型 8KB……。

RC——STC 单片机内部 RAM（随机读写存储器）为 512B。还有如 RD + 表示内部 RAM 为 1280B。

40——表示芯片外部晶振最高可接入 40MHz。对 AT 单片机数值一般为 24，表示外部可接入晶振最高 24MHz。

I——产品级别，表示芯片使用温度范围。I 表示工业级，温度范围：-40 ~ 85℃。

PDIP——表示封装型号。PDIP 表示双列直插式。

1.1.2 单片机应用领域

单片机属于控制类芯片，近年来在各种领域都有着广泛的应用，具体如下：

(1) 实时工业控制 单片机用于各种物理量采集与控制,如电流、电压、温度、液位、流量等物理参数的采集和控制。可以根据被控对象的不同特征采用不同的算法,实现期望的控制指标,从而提高生产效率。如:电动机转速控制、温度控制、自动生产线控制等。

(2) 智能仪器仪表 单片机用于各种仪器仪表,提高了仪器仪表的使用功能和精度,使仪器仪表智能化。如:各种智能电气测量仪表、智能传感器等。

(3) 消费类电子产品 如洗衣机、电冰箱、空调机、电视机、微波炉、IC卡、汽车电子设备、高档玩具、医疗设备、打印机、点钞机等。

(4) 通信 如调制解调器、程控交换机、手机等。

(5) 交通设备 在交通领域中,单片机在汽车、火车、飞机、航天器等方面均有着广泛应用,如汽车自动驾驶系统、航天测控系统等。

1.1.3 如何学好单片机

对于单片机的初学者来说,学习的方法和途径非常重要。学习单片机不是为了应付考试,而是学会如何利用单片机进行系统开发。

单片机的学习方法:1块开发板+实践、2个老师、3个重点、4个步骤。

1块开发板+实践:目前,大部分大学生处于这种状态。

课堂上听了,回去就忘了,考前复习下,考后又忘记了。

学习单片机一定要动手实践,并在实践中成长,切记犯“眼高手低”的毛病。

2个老师:在实践过程中,遇到问题,要先学会从网络、书本中找答案,不要碰到问题就马上想到问老师。老师直接告诉的答案,没有自己找的答案印象深刻。只有自己经历过,琢磨过,印象才更加深刻。

3个重点:①单片机内部结构及工作原理;②寄存器配置;③编程思路-逻辑思维。

4个步骤:

1) <照葫芦画瓢>。对于单片机的初学者来说,可能对单片机的内部结构、C语言编程方法、单片机外部器件等概念全无或不理解。建议刚开始,先记住,照着做。

2) <他山之石,可以攻玉>。通过修改前人的程序来实现自己产品的功能,在工程师实际产品研发的时候,比如一个产品,如果从零起步的话,可能会走很多弯路,遇到前人已经遇到的问题。所以通常的做法是寻找或购买同类的产品,根据自己的功能,在同类产品的的基础上设计出自己的产品,缩短开发周期。

3) <温故知新>。实践—理论—实践—理论—实践,一直循环下去。有的知识点在学的时候不明白,过了段时间,回头再学习的时候就明白了(恍然大悟)。

4) <举一反三>。

1.2 数制与编码

1.2.1 数制

单片机常用的数制有二进制、十进制、十六进制。

1. 二进制

数字电路中只有两种电平特性，即高电平和低电平，只需要用“0”和“1”两个数字区分就可以了。所以数字电路中使用二进制，单片机、计算机也是采用二进制。

在二进制数的末尾加字母 B 表示它是一个二进制数，如：1010B。二进制的基数为 2，二进制遵循“逢二进一，借一当二”的原则。

2. 十进制

十进制是大家熟悉的进制，有 0~9 十个数。在十进制数的末尾加字母 D 或省略表示它是一个十进制数，如 6789D。十进制的基数为 10，十进制遵循“逢十进一，借一当十”的原则。

3. 十六进制

十六进制有 0~9、A、B、C、D、E、F 共十六个数。在十六进制数的末尾加字母 H 表示它是一个十六进制数，如 6A8CH。十六进制的基数为 16，十六进制遵循“逢十六进一，借一当十六”的原则。在 C 语言编程中，把十六进制数写成带前缀 0x。采用十六进制可以大大减轻阅读和书写二进制数时的负担。

1.2.2 数制之间的转换

三种常用进制之间的转换如表 1-3 所示。

表 1-3 常用进制之间的转换

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

以上三种进制之间的转换，记住“8421”码。

如： 8 4 2 1

二进制 0 1 0 1 转化成十进制 $4 + 1 = 5$ ，十六进制 0x5 (5H)

8 4 2 1

二进制 1 1 0 1 转化成十进制 $8 + 4 + 1 = 13$ ，十六进制 0xD (DH)

8 4 2 1

十进制 $9 = 8 + 1$ ，二进制为 1 0 0 1。

十六进制与二进制之间的转换采用四位合一或一分四位的方法：

如二进制数 110100111010B 转十六进制数为 D3AH

1101 0011 1010

D 3 A

十六进制数 D08AH 转二进制数为 1101 0000 1000 1010B

D 0 8 A

1101 0000 1000 1010

思考：0~255 共 256 个数，用十六进制怎么表示？

首先十进制 0~7 共 8 个数，用二进制 000~111 表示，需要 3 个位表示，即 $8 = 2^3$ 表示。得到幂的值刚好就是二进制的位数。

$256 = 2^8$ ，也就需要 8 个位才够表示，00000000~11111111 转换成十六进制为，00H~FFH（C 语言中为 0x00~0xff）。

同理：8K 字节 ROM，用十六进制表示其地址。

$8K = 2^3 * 2^{10} = 2^{13}$ ，需要 13 位才够表示，0 0000 0000 0000~1 1111 1111 1111 转换为十六进制为 0000H~1FFFH。

1.2.3 常用的编码

单片机只能识别二进制数，因此涉及的数字、字母和符号需要事先进行二进制编码，便于单片机识别、存储、处理、传送。单片机常用的编码主要有两种。

1. ASCII 码（字符编码）

单片机有时要处理大量数字、字母和符号，需要对这些数字、字母和符号进行二进制编码。这些被编码的信息统称为字符。这些数字、字母和符号的二进制编码又称为字符编码。

通常 ASCII 码由 7 位二进制数码构成，共 128 个字符编码。这些字符分为两类：一类是图形字符，共 96 个；另一类是控制字符，共 32 个。图形字符又包括十进制字符 10 个，大小写英文字母 52 个，其他字符 34 个。数字 0~9 的 ASCII 码为 30H~39H，大写字母的 ASCII 码为 41H~51H。

详见附录常用 ASCII 码表。

2. BCD 码

BCD 码是一种十进制的二进制编码。BCD 码种类比较多，以 8421 码为例，因组成它的四位二进制数码的权为 8、4、2、1 而得名。

压缩 BCD 码采用四位二进制来代表一位十进制数码，BCD 码与十进制代码的对应关系如表 1-4 所示。

表 1-4 8421BCD 码编码表

十进制数	8421 码	十进制数	8421 码
0	0000	8	1000
1	0001	9	1001
2	0010	10	0001 0000
3	0011	11	0001 0001
4	0100	12	0001 0010
5	0101	13	0001 0011
6	0110	14	0001 0100
7	0111	15	0001 0101

由表可见，用 BCD 码表示十进制数时，10 以上的十进制数至少需要 8 位二进制数字（两位 BCD 码字）来表示。BCD 码以二进制形式出现，是逢十进位的，但它不是一个真正的二进制数，因为二进制数是逢二进位的。

1.3 微型计算机的基本组成和工作过程

1.3.1 基本组成

1942年2月15日，第一台电子数字计算机 ENIAC (Electronic Numerical Integrator and Computer) 问世，标志着计算机时代的到来。

1946年6月，匈牙利籍数学家冯·诺依曼提出了“程序存储”和“二进制运算”的思想，进一步构建了由运算器、控制器、存储器、输入设备和输出设备组成的这一经典的计算机结构，如图 1-2 所示。

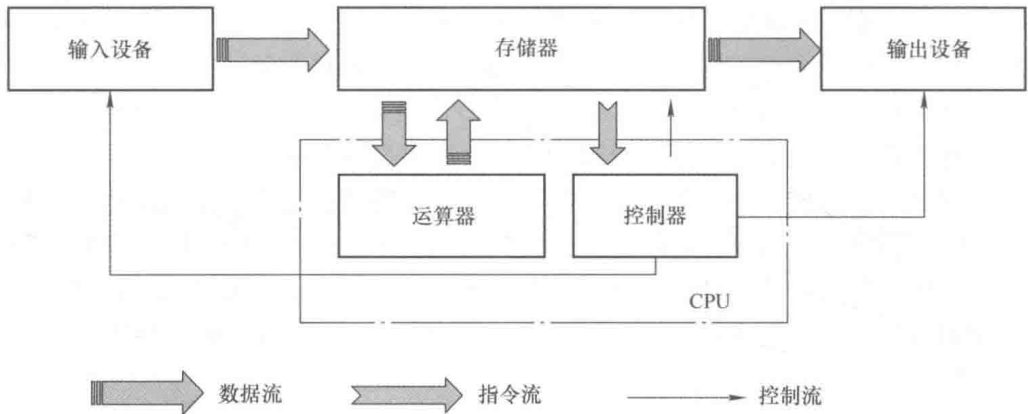


图 1-2 冯·诺依曼计算机结构

电子计算机技术的发展，相继经历了电子管计算机、晶体管计算机、集成电路计算机、大规模集成电路计算机和超大规模计算机五个时代，但是计算机的结构始终没有突破冯·诺依曼的经典结构。

随着集成电路技术的飞速发展，1971年1月，Intel公司的德·霍夫将运算器、控制器以及一些寄存器集成在一块芯片上，即称为微处理器或中央处理单元（简称 CPU）。图 1-3 为微型计算机的组成框架，由微处理器、存储器（ROM、RAM）和输入输出接口（I/O 接口）和连接它们的总线组成。

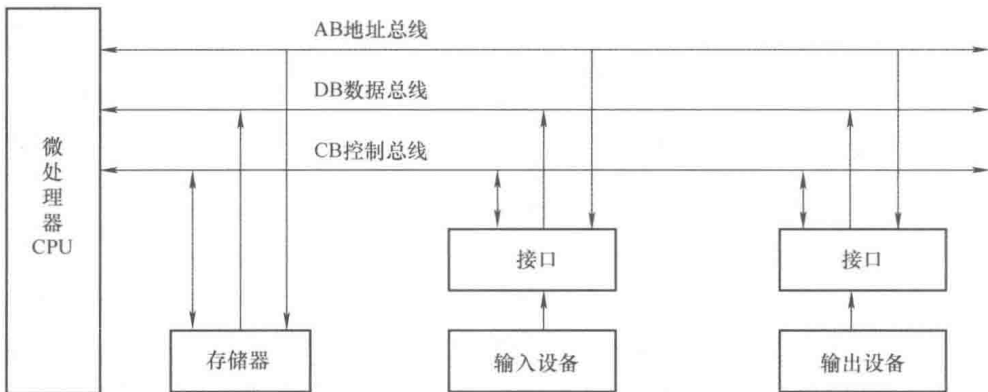


图 1-3 微型计算机组成框图

1. 运算器

运算器又称算术逻辑单元 (Arithmetic Logic Unit, ALU)。它是计算机对数据进行加工处理的部件, 包括算术运算 (加、减、乘、除等) 和逻辑运算 (与、或、非、异或、比较等)。

2. 控制器

控制器负责从存储器中取出指令, 并对指令进行译码; 根据指令的要求, 按时间的先后顺序, 负责向其他各部件发出控制信号, 保证各部件协调一致地工作, 一步一步地完成各种操作。控制器主要由指令寄存器、译码器、程序计数器、操作控制器等组成。

硬件系统的核心是中央处理器 (Central Processing Unit, CPU)。它主要由控制器、运算器等组成, 并采用大规模集成电路工艺制成的芯片, 又称微处理器芯片。

3. 存储器

存储器是计算机记忆或暂存数据的部件。计算机中的全部信息, 包括原始的输入数据。经过初步加工的中间数据以及最后处理完成的有用信息都存放在存储器中。而且, 指挥计算机运行的各种程序, 即规定对输入数据如何进行加工处理的一系列指令也都存放在存储器中。存储器分为内存储器 (内存) 和外存储器 (外存) 两种。

4. 输入设备

输入设备是给计算机输入信息的设备。它是重要的人机接口, 负责将输入的信息 (包括数据和指令) 转换成计算机能识别的二进制代码, 送入存储器保存。

5. 输出设备

输出设备是输出计算机处理结果的设备。在大多数情况下, 它将这些结果转换成便于人们识别的形式。

1.3.2 指令、程序与编程语言

一个完整的计算机是由硬件和软件两部分组成的, 缺一不可。计算机的硬件部分是看得到、摸得着的实体部分, 但硬件部分只有在软件的指挥下, 才能发挥其功能。计算机事先把程序加载到存储器中, 启动运行后, 自动按照程序进行工作。

指令是规定计算机完成特定任务的命令, 微处理器就是根据指令指挥与控制计算机各部分协调地工作。

程序是指令的集合, 是解决某个具体任务的一组指令。在计算机完成某个任务之前, 须事先把程序以二进制代码 (机器代码) 的形式存放在程序存储器中。

编程语言分为机器代码、汇编语言和高级语言。

机器代码是用二进制代码表示的, 是机器能直接识别的语言 (计算机是以二进制计数)。

汇编语言是用英文助记符来描述指令的, 但不能独立于机器。

高级语言是采用独立于机器, 人们习惯使用的语言形式, 比如 C 语言。

举例:

把累加器 A 的内容加 1, 就是一条单片机指令, 用二进制代码 “0000 0100B” 表示, 单片机可以识别并执行。但如果指令较多时, 记忆起来比较困难。用英文助记符 “INC A” 表示, 容易记忆。

汇编语言实质上是机器语言的助记符。CPU 只能运行它所支持的指令集，而这些指令集当中的每条指令都是一些二进制数的序列，也就是“0”和“1”的有序组合；而“0”和“1”的组合不便于程序员的记忆，因此有了“MOV A 0x40”等这样的助记符。所以汇编语言编译成 CPU 可执行的机器语言，其实只要做一个翻译的动作就好了。而用 C 语言编写完程序后，需要通过编译器将 C 语言编译成与相应 CPU 指令集对应的机器语言。汇编语言与机器语言是一一对应的。但是 C 语言呢？C 语言的语法是固定的，C 语言编写的程序要编译成 CPU 能读懂的机器语言，因此需要有编译规则，所以运行效率低一些。也可以说，C 语言是面向程序员的语言，而汇编语言是直接面向 CPU 的语言。

1.3.3 工作过程

微型计算机的工作过程就是执行程序的过程，计算机执行程序是一条指令、一条指令执行的。执行一条指令的过程分为三个阶段：取指、指令译码与执行指令，每执行完一条指令，自动转向下一条指令的执行。

1. 取指

根据程序计数器 PC 中的地址，到程序存储器中取出指令代码，并送到指令寄存器中，然后 PC 自动加 1，指向下一指令（或指令字节）地址。

2. 指令译码

指令译码器对指令寄存器中的指令代码进行译码，判断出当前指令代码的工作任务。

3. 执行指令

判断出当前指令代码任务后，控制器自动发出一系列微指令，指挥计算机协调地动作，完成当前指令指定的工作任务。图 1-4 所示为微型计算机工作过程的示意图，程序存储器从 0000H 开始存放了如下所示的指令代码。

汇编源程序	对应的机器代码
ORG 0000H	; 伪指令，指定下列程序代码从 0000H 地址开始存放
MOV A, #0FH	; 740FH
ADD A, 22H	; 2522H
MOV P1, A	; F590H
SJMP \$; 80FEH

下面分析微型计算机的工作过程：

- 1) 将 PC 内容 0000H 送地址寄存器 MAR。
- 2) PC 值自动加 1，为取下一条指令字节的机器代码做准备。
- 3) MAR 中的地址经地址译码器找到程序存储器 0000H 单元。
- 4) CPU 发读命令。
- 5) 将 0000H 单元内容 74H 读出，送至数据寄存器 MDR 中。
- 6) 接着，将 74H 送指令寄存器 IR 中。
- 7) 经指令译码器 ID 译码，判断出指令代码所代表的功能，由操作控制器 OC 发出相应的微操作控制信号，完成指令操作。
- 8) 根据指令功能要求，PC 内容 0001H 送地址寄存器 MAR。
- 9) PC 值自动加 1，为取下一个指令字节的机器代码做准备。