



全国高等教育自学考试

微型计算机原理与接口技术 自学辅导

全国高等教育自学考试指导委员会 组编
胡旭晓 编著



浙江大学出版社

S
K
X
Z
Y
J
D
G
G
Q

全国高等教育自学考试

微型计算机原理与接口技术

自学辅导

全国高等教育自学考试指导委员会组编

胡旭晓 编著

浙江大学出版社

图书在版编目(CIP)数据

微型计算机原理与接口技术自学辅导/胡旭晓编. —杭州:
浙江大学出版社, 2001. 10
ISBN 7-308-02832-1

I. 微... II. 胡.. III. ①微型计算机—基础理论
—高等学校—自学参考资料②微型计算机—接口—高等
学校—自学参考资料 IV. TP36

中国版本图书馆 CIP 数据核字(2001)第 077416 号

出版发行 浙江大学出版社

(杭州市浙大路 38 号 邮政编码 310027)

(E-mail: zupress@mail. hz. zj. cn)

(网址: <http://www.zjupress.com>)

责任编辑 林昌东

排 版 浙江大学出版社电脑排版中心

印 刷 浙江大学印刷厂

开 本 787mm×1092mm 1/16

印 张 9.25

字 数 237 千字

印 数 0001—3070

版 次 2001 年 10 月第 1 版 2001 年 10 月第 1 次印刷

书 号 ISBN 7-308-02832-1/TP·221

定 价 15.00 元

出版前言

为了完善高等教育自学考试教育形式,促进高等教育自学考试的发展,我们组织编写了全国高等教育自学考试自学辅导书。

自学辅导书以全国考委公布的课程自学考试大纲为依据,以全国统编自考教材为蓝本,旨在帮助自学者达到学习目标,顺利通过国家考试。

自学辅导书是高等教育自学考试教育媒体的重要组成部分,我们将根据专业的开考情况和考生的实际需要,陆续组织编写、出版文字、音像等多种自学媒体,由此构成与大纲、教材相配套的、完整的自学媒体系统。

全国高等教育自学考试指导委员会
1999年10月

目 录

第1章 微型计算机的基础知识	3
1.1 知识要点	3
1.2 典型例题分析	5
第2章 C语言概述与最简单的C程序	7
2.1 知识要点	7
2.2 典型例题分析	11
第3章 位运算	16
3.1 知识要点	16
3.2 典型例题分析	17
第4章 选择结构和循环结构	20
4.1 知识要点	20
4.2 典型例题分析	22
第5章 C语言的数组	29
5.1 知识要点	29
5.2 典型例题分析	30
第6章 C语言函数	34
6.1 知识要点	34
6.2 典型例题分析	35
第7章 C语言指针	39
7.1 知识要点	39
7.2 典型例题分析	40

第 8 章 C 语言结构与联合	45
8.1 知识要点	45
8.2 典型例题分析	46

录 目

第 9 章 C 语言文件	52
9.1 知识要点	52
9.2 典型例题分析	53

第 2 部分 MCS-51 单片机原理与接口技术

第 10 章 MCS-51 单片机的内部结构	61
10.1 知识要点	61
10.2 典型例题分析	66

第 11 章 MCS-51 单片机的指令系统	73
11.1 知识要点	73
11.2 典型例题分析	75

第 12 章 MCS-51 单片机汇编语言程序设计	83
12.1 知识要点	83
12.2 典型例题分析	93

第 13 章 MCS-51 单片机的中断与定时	99
13.1 知识要点	99
13.2 典型例题分析	106

第 14 章 MCS-51 单片机系统扩展与接口技术	114
14.1 知识要点	114
14.2 典型例题分析	134

●第 1 部分●

C 语言程序设计

● 卷 1 附 錄 ●

○ 哥 言 野 氣 新 刊 ○

第1章 微型计算机的基础知识

本章学习的目的是了解微型计算机的发展,掌握计算机中数的表示方法及各种进制数的转换,掌握微型计算机的基本组成原理,了解常用计算机语言及计算机解题过程。

1.1 知识要点

1.1.1 计算机的组成

微型计算机由运算器、控制器、存储器、输入设备和输出设备这五部分组成。

运算器是执行算术运算和逻辑运算的部件。ALU(Arithmetic and Logic Unit)为算术逻辑运算单元,是计算机处理数据的核心。ALU的运算能力、运算速度、运算宽度这三大性能,很大程度上决定了微处理器的特性指标。寄存器用来暂时存放数据,寄存器与算术逻辑运算单元的结合,称为寄存器算术逻辑运算单元(RALU)。

运算器、控制器和寄存器通常合称为中央处理器(Central Processing Unit),简称CPU。CPU制作在一块芯片上就是微处理器。

参与计算机运行的程序和数据都必须存放在存储器中。能够被CPU直接寻址的存储器,称为计算机的主存储器,或叫内部存储器。内部存储器主要分为RAM和ROM两大类。RAM为可读可写存储器,计算机运行时随时可以改变RAM中的内容,缺点是一旦停电,其内容即丢失。它用来存放需要改写的的数据。ROM为只读存储器,由生产厂家事先把数据存放在存储器内,使用过程中,对ROM中的内容只能读,不能写。其优点是停电时,信息不会丢失,因而可以存放系统程序或表格。另外,大批量生产时,ROM较为便宜。

输入设备(Input Device)与输出设备(Output Device)合称为计算机外围设备(Peripheral Device),简称I/O设备或外设。

被主机作为I/O设备看待的磁带、磁盘、光盘等存储器称作计算机的辅助存储器,或叫外部存储器。

主机和外围设备构成了计算机的硬件(Hardware),它是计算机中看得见、摸得着的部分。计算机中还有一部分是看不见、摸不着的东西,如程序、数据、文件等,这些统称计算机软件。计算机软件分为系统软件和应用软件。

计算机的硬件与相应的软件,构成计算机系统。

1.1.2 单片机的组成及分类

在一块芯片上,集成了中央处理器(CPU)、随机读写存储器(RAM)、只读存储器(ROM)、定时器/计数器以及输入/输出接口等部件,以构成基本计算机系统,即称为单片机。

单片机有多种,大多数单片机的内部结构及指令系统都互不相同。单片机的分法有多种,

如按厂家分,有 Intel 公司生产的 MCS-48 系列、MCS-51 系列和 MCS-96 系列单片机, Motorola 公司生产的 M68 系列单片机, Zilog 公司生产的 Z8 系列单片机, Fairchild 公司生产的 F8 单片机等。本书学习的单片机为 MCS-51 系列单片机。MCS-51 系列单片机主要有 8031、8051、8751 三种型号。8031 单片机内部没有程序存储器,使用时必须扩展程序存储器,非常适合于新产品的研制。8051 单片机内有 4K ROM 的程序存储器。8751 单片机内有 4K EPROM 的程序存储器。MCS-51 系列单片机为目前单片机的主流机型,并被作为单片机教学的首选机型之一。Intel 公司把 8051 的内核转让给 Atmel、Philips 等公司后,出现了 Philips 公司生产的 80C552、87C552 单片机,和 Atmel 公司生产的 89 系列和 90 系列单片机,如 89C51、89C52、89C55 等。这些单片机的指令系统与 MCS-51 系列单片机的指令系统是一致的。

如果按照 CPU 的字长分,可以分为 4 位机、8 位机、16 位机和 32 位机等。如 8 位机有 Intel 公司生产生产的 MCS-51 系列单片机,16 位机有 Intel 公司生产的 MCS-96 系列单片机。目前应用最多的是 8 位机和 16 位机。

1.1.3 计算机中的数据

计算机是与数据打交道的,所有需要用计算机处理的信息,都必须被转换成数据。数据信息又可以分为数值数据和非数值数据两大类。

计算机的主机是由数字电路构成的,它只能处理二进制数。计算机中所有信息,包括指挥计算机工作的程序本身,都必须以二进制形式进行编码,才能最终在计算机上操作。我们学习的高级语言(如 C 语言、C++ 语言、Basic 语言等)、汇编语言程序,如要在计算机上运行,必须先转化为机器语言,即二进制指令。这个转化过程,称为编译。编译过程可以手工进行,也可以自动进行。手工编译工作量大,一般都使用计算机自动编译。自动编译程序通常由计算机厂家或软件公司提供。

在微处理机中,常用的数值数据的进制有二进制、十进制和十六进制,其中带符号的机器数,可以用原码、反码、补码等形式表示。而为了适应数的大小,可以用定点数或浮点数两种方法表示一个数。

而在计算机中,非二进制数也必须用二进制代码表示。例如,十进制数常用 BCD 码表示。

在计算机中,文字、字母、逻辑符号、校验码等非数值数据,也只能采用二进制编码形式,才能为机器所接受。例如,10 个数字符号、26 个英文字母、部分专用符号等,通常采用 ASCII 码表示。

1.1.4 计算机的工作过程

计算机处理数字信息的过程是通过 CPU 周而复始地从内部存储器中逐条取出指令、解释指令并执行指令来实现的。

为了让计算机按照我们的意图工作,必须把我们的意图编制成能够被计算机识别和执行的程序,并把它转换成二进制代码装入主存储器中,然后再交计算机去执行。

1.2 典型例题分析

1.2.1 单项选择题

1. 单片机中,把临时数据存放在()。

- A)ROM B)RAM C)EPROM D)PROM

答案:B

分析:ROM 是只读存储器,其内容不能改变。RAM 为可读可写的存储器,因此可以用来存放临时数据;EPROM 为可擦除可编程的只读存储器,可擦除指在紫外线照射下,经过一段时间后内容被擦除。可编程,指用专用编程器编程。因此,一般用它来存放不变的程序或表格。PROM 为可编程的只读存储器,编程后就不能再修改,因此也不适合用来存放临时数据。

2. 目前,生产单片机的主要生产工艺是()。

- A)CMOS B)PMOS C)NMOS D)HCMOS

答案:D

分析:用 PMOS、NMOS 工艺生产的沟道产品,其功耗大、速度低、集成度不高,用于早期产品;用 CMOS 工艺生产的产品,功耗低,但速度不高;而 HCMOS 工艺用来生产高密度和高速度的产品,是目前单片机生产的主要技术。

3. 参考 ASCII 码表,字符 O 的 ASCII 码是()。

- A)0 B)30H C)03H D)30

答案:B

分析:O 作为一个字符,查 ASCII 码表,其 ASCII 码为 30H。H 表示十六进制数,如果转换成十进制数为 48。

4. 8 位无符号二进制数表示的数的范围是()。

- A)0~255 B)0~99 C)0~127 D)0~128

答案:A

分析:8 位无符号二进制数表示的范围为 00H~FFH,转化为十进制数为 0~255。

5. 4 位无符号二进制数表示的数的范围是()。

- A)0~9999 B)0~9 C)0~15 D)0~99

答案:C

分析:4 位无符号数,用二进制表示时最小值为 0000B,最大值为 1111B,如果转换为十进制数,则可表示的数的范围为 0~15。

6. 一个字(Word)占()位。

- A)1 位 B)8 位 C)16 位 D)32 位

答案:C

分析:位(Bit)是二进制数的最小单位,一个字节(Byte)为 8 位;一个字(Word)为两个字节,即为 16 位;双字(Double Word)为两个字,即 4 个字节,为 32 位。

7. 8051 单片机是()位机,80C196 单片机是()位机。

- A)8,8 B)16,16 C)51,196 D)8,16

答案:D

分析:几位机是根据中央处理器(CPU)的字长来分的。CPU的字长长,数据处理能力强,但单片机的价格高。8051单片机CPU的字长为8位,因此是8位机;而80C196单片机CPU的字长为16位,因此是16位机。

8. 十六进制的基数为()。

A)2 B)10 C)16 D)1

答案:C

分析:表示某数制的符号的个数,叫做基数。
二进制逢2进1,用以表示二进制的符号有0、1,二进制数的基数为2。

十进制数逢10进1,用以表示十进制的符号有0、1、2、3、4、5、6、7、8、9,十进制数的基数为10。

十六进制数逢16进1,表示十六进制的符号有0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F,十六进制数的基数为16。

9. 十进制数8917中,9的权是多少()。

A) 10^3 B) 10^2 C)9 D) 10^1

答案:B

分析:某数符的权,表示以基数为底数,以数符在数中所在的位置为幂次。十进制数8917中,数符7的权为 10^0 ,数符1的权为 10^1 ,数符9的权为 10^2 ,数符8的权为 10^3 。又如十进制数23589中,数符9的权为 10^0 ,数符8的权为 10^1 ,数符5的权为 10^2 ,数符3的权为 10^3 ,数符2的权为 10^4 。

10. 二进制数1011011B转化为十六进制数是()。

A)5BH B)B3H C)133H D)551H

答案:A

分析:二进制整数转换为十六进制整数时,可以把二进制数按4位分组进行转换,从最低位开始。如最后余下的不到4位,则在其左边添0,直至4位。因此1011011B可以写成01011011B,转换成十六进制数为5BH。其中,数最后的B表示为二进制数,H表示为十六进制数。

11. 二进制数0.1011011B转化为十六进制数是()。

A)0.5BH B)0.B6H C)0.133H D)0.B3H

答案:B

分析:二进制纯小数转换为十六进制纯小数时,可以将二进制数按4位分组进行转换。分组时,以小数点为基准向右数,不足4位,可以在右边添0直至4位。故0.1011011B,可以写成0.10110110B,转换成十六进制数为0.B6H。

12. 二进制数1011011.1011011B转化为十六进制数是()。

A)5B.B6H B)B3.B6H C)5B.B3H D)B3.B3H

答案:A

分析:二进制小数可看成由二进制整数加二进制纯小数构成。转换成十六进制数时,可参照题10与题11,分别进行整数转换与小数转换,最后将转换后的十六进制整数与小数相加即可。故1011011B.1011011B转换成十六进制数为5B.B6H。

第2章 C语言概述与最简单的C程序

本章学习的目的是使考生了解计算机语言的发展与应用,知道C语言的特点,掌握C语言数据类型的基本概念与定义方法,包括字符型、整型和实型(浮点型)常量及变量的表示方法,掌握利用常量、变量及算术运算符组成的算术表达式。通过实验,掌握赋值语句、数据输入函数 scanf()、数据输出函数 printf()、字符输入函数 getchar()和字符输出函数 putchar()的使用方法,并能正确编写简单的C语言程序。

2.1 知识要点

2.1.1 程序的构成

C语言的程序非常灵活,但其构成基本一致。如下所示为一个完整的C源程序格式:

预编译命令

函数声明

主函数()

{

...

函数 1()

{

...

函数 n()

{

...

}

一个C源程序实际上是若干函数的集合。在组成C源程序的函数中,有一个函数是必不可少的,那就是主函数 main()。C程序的执行从 main()函数开始,正常情况下由 main()函数结束。其他函数直接或间接被主函数调用。因此,其他函数个数可多可少,也可以一个都没有,视编程需要而定。

预编译命令是对程序正式编译以前的预处理。预编译命令不是C语言的一部分,但它与指令有机地结合在一起,从而有效改善程序的组织结构。

函数定义与函数原型声明是有区别的。函数原型声明是说明某一个标识符为函数,可以供其他函数调用。函数定义给出了函数的功能。

C 源程序的函数次序没有强制要求。如果某函数在被其调用的函数之前定义,那么这些被调用的函数必须在此函数之前进行原型声明。否则,编译程序编译时不认识这些被调用的没有声明过的函数。如果主函数在最后,C 语言的程序格式可以表示为:

预编译命令

函数原型声明

函数 1()

函数 n()

主函数()

特别注意:尽管主函数放在后面,C 程序的执行还是从主函数开始。

2.1.2 常量、变量、表达式

数据是对计算机处理对象的总称,可以分为常量与变量两类。常量指程序运行过程中,数值不发生变化的量;而变量指程序运行过程中,数值可能发生变化的量。

计算机存放数据,最后都是把它转化为多位 0、1 存放。一个数据,根据其范围和特性,以及有效位数,可分为字符型、整型和实型等。计算机中数据存储以字节为基本单位,一个字节为 8 位。不同类型的数据存储时占用的字节数不同。数据存储占用的字节数与操作系统及编译程序有关,例如在 Windows 3.1 系统上,常用编译程序的数据类型与存储所占用的字节数之间关系如下:

字符型	字符变量(char)	1 个字节
整型	整型变量(int)	2 个字节
	长整型变量(long int)	4 个字节
	无符号整型变量(unsigned int)	2 个字节
实型	无符号长整型变量(unsigned long int)	4 个字节
	单精度浮点型变量(float)	4 个字节
	双精度浮点型变量(double)	8 个字节

单精度实数提供 6~7 位有效数字,双精度实数提供 15~16 位有效数字,数值范围与机器系统有关。通常,单精度实数的数值范围约为 $-3.4 \times 10^{38} \sim 3.4 \times 10^{38}$,双精度实数的数值范围

为 $-1.7 \times 10^{308} \sim 1.7 \times 10^{308}$ 。

作用域是一个声明在程序正文中有效的那一部分区域。变量声明的作用域决定程序在运行时所建立变量的生存期。静态变量的生存期与程序的运行期相同,只要程序一开始运行,其生存期就存在了,当程序结束时,生存期结束。在函数内部声明的变量有动态生存期。这种变量的生存期开始于声明点,而结束于命名它的标识符所在作用域的结束处。

表达式是由常量、变量、函数通过运算符连接起来的一个算式。一个常量、一个变量或一个函数,都可以看成一个表达式。实际上,表达式用来对数据进行加工。表达式代表一个值,求这种具体的值时,应特别注意各运算符的优先级和结合性。

2.1.3 运算符的优先级及计算方法

在含有多个运算符的表达式中,运算按照运算符的优先级高低顺序进行,如果运算符优先级相同,则根据运算符的结合性来确定运算的次序。

运算符的优先级次序及结合性规定如下:

表 2-1 运算符的优先级

运算符	结合性
()(括号) [] (数组下标) -> (结构或联合成员) . (结构或联合成员)	自左向右
!(逻辑非) ~(位取反) +(正号) -(负号) ++(加1) --(减1) &(取变量地址) *(指针所指内容) type(函数说明) sizeof(长度计算)	自右向左
*(乘) /(除) %(取模)	自左向右
+(加) -(减)	自左向右
<<(位左移) >>(位右移)	自左向右
<(小于) <=(小于等于) >(大于) >=(大于等于)	自左向右
==(等于) !=(不等于)	自左向右
&(位与)	自左向右
^(位异或)	自左向右
(位或)	自左向右
&&(逻辑与)	自左向右
(逻辑或)	自左向右
?: (条件表达式)	自右向左
= (赋值) += -= *= /= %= &= ^= = <<= >>= (联合操作)	自右向左
, (逗号运算符)	自右向左

优先级
↑
高
↓
低

运算符的优先级次序及结合性规定非常重要,必须记住。

1. 自加及自减运算符

自加运算符++的作用是将变量值加1,为单目运算。单目运算是指只有一个操作数的运算。自加运算符++可以放在变量的前面,也可以放在变量的后面,但两种写法的含义不一样。

例如 `int k=3, j;`

`j = k++;`

实际上可以等价为:

`int k=3, j;`

`j = k;`

`k = k + 1;`

即先将k赋值给变量j,然后变量k自加1。又如

int k=3, j;

j = ++k;

可以等价为:

int k = 3, j;

k = k + 1;

j = k;

即变量 k 先自加 1, 然后赋值给变量 j。

自减运算符--的作用是将变量值减 1, 为单目运算。同样, j=k--; 可以等价为 j=k; k=k-1; 而 j=--k; 可以等价为 k=k-1; j=k。自增减运算符在变量前后的区别, 在于先作自增减运算还是先取值。前后位置不同, 可能导致运算结果完全不同。

2. 双目赋值运算符

赋值运算符中, +=、-=、*=、/=、%= 都是双目运算符, 它们的优先级与简单赋值操作符=相同。

假设 OP 表示这些赋值符中符号=左边的操作符, 如+、-、*、/、%, 则表达式

变量 OP= 表达式

可以理解为:

变量 = 变量 OP 表达式

如表达式 x += y 等价为 x = x + y, 而表达式 x *= y + 5 等价为 x = x * (y + 5)。

2.1.4 头文件

头文件可以认为是 C 语言中库函数的集合, 一些同类的函数原型放在一起就形成了头文件。

把正弦 sin(x)、余弦 cos(x)、绝对值 fabs(x)、计算平方根 sqrt(x) 等数学函数原型放在一起, 就形成 math.h 头文件; 把字符转换成大写字母 toupper(ch)、字符转换成小写字母 tolower(ch)、统计字符串中字符的个数 strlen(str)、检查是否大写字母 isupper(ch) 等字符函数原型放在一起, 就形成 string.h 的头文件; 把输出值输出到标准输出设备 printf(format, ...) 等输入、输出函数原型放在一起, 就形成 stdio.h 头文件。各头文件中具体包含了哪些函数, 可以查表。

C 程序中, 当调用某一库函数时, 必须用文件包含预处理命令 #include <包含文件名> (或 #include "包含文件名") 把库函数声明所在的头文件包含进来。比如调用求平方根函数 sqrt(x), 必须有 #include <math.h> 预处理命令。又如调用判断字符变量是否为大写字母函数 isupper(ch), 必须有 #include <string.h> 预处理命令。再如需要调用输出字符串函数 printf(str), 必须有 #include <stdio.h> 预处理命令。

2.1.5 基本语句

C 的基本语句有表达式语句、空语句、复合语句、函数调用语句和流程控制语句等。

表达式语句的形式是表达式及后面跟一分号。分号与表达式合在一起在语法上构成一个语句。最常见的表达式语句是赋值语句。

空语句就是单独一个分号。它的作用只是在某些时候保证语法完整性, 其本身不作任何动作。

复合语句的形式是：{说明语句；可执行语句}。复合语句以“{”开始，以“}”结束。复合语句从形式上讲可以由多个语句组成，但在语法上相当于一个语句。

函数调用实际上也是一个表达式，它后面加一个分号就是一个语句。

前面4种语句是对数据进行操作的，而控制语句可以控制语句的执行顺序。C语言程序有三种执行顺序：顺序结构、分支结构和循环结构。除顺序结构外，另两种结构要用流程控制语句实现。

C语言提供了可以进行格式转换的输入函数scanf和输出函数printf。比如希望将一个整数输出为一个字符，就需格式转换。格式转换由一系列的格式转换符控制完成，具体参见教材。

2.2 典型例题分析

2.2.1 将下列数学表达式写成C语言的表达式

1. $(xy)^2(-2x)^3$

2. $\frac{\sqrt{x+y+z}}{3x}$

3. $\frac{\sqrt{1+|\cos(x)|}}{x+y}$

4. $\frac{1}{\frac{1}{r1} + \frac{1}{r2} + \frac{1}{r3}}$

答案：

1. $(x*y)*(x*y)*(-2*x)*(-2*x)*(-2*x)$

2. $\text{sqrt}(x+y+z)/(3*x)$

3. $\text{sqrt}(1+\text{fabs}(\cos(x)))/(x+y)$

4. $1/((1/r1)+(1/r2)+(1/r3))$

注意：

所有参加运算的数之间，都应该用运算符分开。

数学运算符优先顺序为先乘除，后加减；同级时先左后右，即左结合。有括号时，括号内运算优先。

余弦函数cos(x)、绝对值函数fabs(x)、平方根函数sqrt(x)的原型声明都在math.h的头文件中。写完整程序时，必须包含此头文件，即程序开始处有#include <math.h>。

2.2.2 单项选择题

1. 字符型数据在微机中的存储形式是()。

A) 补码 B) 反码 C) 函数 D) ASCII码

答案：D

分析：将一个字符常量放到一个字符变量中，并不是将该字符本身放到内存单元中，而是将该字符相应的ASCII码放到存储单元中。例如存放字符b，就是存放字符b的ASCII码，即98。

2. 以下不是C语言整数常量的是()。