

微型计算机原理

陈昌志 瞿 中 邹永贵 等 主编



重庆出版集团

天健电子音像出版社

微型计算机原理

陈昌志 瞿中 邹永贵 等 主编



重庆出版集团

天健电子音像出版社

图书在版编目 (CIP) 数据

微型计算机原理 / 陈昌志著. — 重庆: 重庆出版社, 2006.8
ISBN 7-5366-7727-8

I. 微... II. 陈... III. 微型计算机—理论 IV. TP36

中国版本图书馆 CIP 数据核字 (2006) 第 098582 号

微型计算机原理

WEIXINGJISUANJI YUANLI

陈昌志 瞿中 邹永贵 等著

出 版 人: 罗小卫

责任编辑: 陈仕达

封面设计: 汤一丹



重庆出版集团 出版
重 庆 出 版 社

重庆长江二路 205 号 邮政编码: 400016 <http://www.cqph.com>

重庆出版集团艺术设计有限公司制版

重庆青松实业公司印刷厂印刷

重庆出版集团图书发行有限公司发行

E-MAIL: fxchu@cqph.com 电话: 023-68809452

全国新华书店经销

开本: 787mm×1092mm 1/16 印张 25 字数: 578 千

2006 年 8 月第 1 版 2006 年 8 月第 1 版第 1 次印刷

印数: 1~2000 册

定价: 36.00 元 (含光盘)

如有印装质量问题, 请向本集团图书发行有限公司调换: 023-68809955 转 8005

版权所有 侵权必究

目 录

第 1 章 微型计算机系统概述	1
1.1 微型计算机发展概述	1
1.2 计算机中的数制和码制	2
1.3 微型计算机系统组成	7
1.4 当前微型计算机所使用的先进技术	10
第 2 章 80x86 微处理器	12
2.1 x86 微处理器的引脚信号	16
2.2 x86 微处理器的内部结构	23
2.3 x86 微处理器的工作模式	33
2.4 x86 微处理器的总线时序	35
第 3 章 x86 微处理器的寻址方式和指令系统	37
3.1 x86 微处理器的数据类型	37
3.2 x86 微处理器的指令格式	43
3.3 x86 微处理器的寻址方式	47
3.4 x86 微处理器的指令系统	55
第 4 章 汇编语言及其程序设计	98
4.1 伪指令	98
4.2 汇编语言程序结构	112
4.3 DOS 系统功能调用与 BIOS 功能调用	118
4.4 汇编语言程序设计	127
4.5 宏指令与条件汇编	137
第 5 章 高级应用程序设计	145
5.1 EXEC 功能与程序段前缀	145
5.2 保护方式编程	150
5.3 中断程序与中断拦截	160
第 6 章 微机输入输出接口技术	167
6.1 基本概念	167
6.2 接口的组成	168
6.3 I/O 端口地址译码技术	170

第 7 章 微机存储系统	182
7.1 内存的管理	183
7.2 存储器接口技术	188
7.3 高速缓冲存储器的接口	190
7.4 虚拟存储器及其管理技术	194
第 8 章 定时/计数器	200
8.1 概述	200
8.2 8254 可编程定时/计数器	201
8.3 8254 在 PC 系统机中的应用	213
8.4 MC146818 及其应用	223
第 9 章 中断系统	230
9.1 中断系统概述	230
9.2 8259A 中断控制器	246
9.3 8259A 在微机系统中的应用	263
第 10 章 DMA	270
10.1 DMA 概述	270
10.2 8237A DMA 控制器	271
10.3 8237A 在 PC 系列机中的应用	285
10.4 8237A 的编程举例	292
第 11 章 串行通信	297
11.1 8251A 可编程串行接口芯片	297
11.2 16550 可编程串行接口芯片	317
第 12 章 并行接口	333
12.1 CPU 与外设之间的数据传输	333
12.2 并行接口与并行通信	336
12.3 8255A 可编程并行接口芯片	337
12.4 8255A 的编程及应用	352
第 13 章 总线	363
13.1 总线的基本概念	363
13.2 系统总线	365
13.3 局部总线	371
13.4 PCMCIA 扩展总线	377
13.5 AGP	379
13.6 通用串行总线	382

第 1 章 微型计算机系统概述

1.1 微型计算机发展概述

电子计算机的诞生和发展是 20 世纪最重要的科技成果之一。进入 20 世纪 70 年代以来,微型计算机开始登上历史舞台,并以不可阻挡的势头迅猛发展,成为当今计算机发展的一个主流方向。当前,以微型计算机为代表的计算机已日益普及,其应用已深入到社会的各个角落,极大地改变着人们的工作方式、学习方式和生活方式,成为信息时代的主要标志。

1.1.1 微型计算机硬件体系的发展

1944 年夏,著名数学家冯·诺依曼(Von Neumann)偶然获知 ENIAC(Electronic Numerical Integrator And Calculator)的研制,在以后的 10 个月里,他参加了为改进 ENIAC 而举行的一系列专家会议,研究了新型计算机的系统结构,在由他执笔的报告里,提出了采用二进制计算,存储程序并在程序控制下自动执行的思想。按照这一思想,新机器将由五个部件构成,即由运算部件、控制部件、存储部件、输入部件以及输出部件构成。报告还描述了各部件的职能和相互间的联系。此后,这种模式的计算机遂被称为“冯·诺依曼机”。1949 年,这一新思想首先由英国剑桥大学的威尔克斯(M.V.Wikes)等在 EDSAC(Electronic Delay Storage Automatic Calculator)机上实现。

1946 年 2 月,在美国宾夕法尼亚大学的莫尔学院,由物理学博士莫克利(J.W.Mauchly)和电气工程师埃克特(J.P.Eckert)领导的小组研制成了世界上第一台数字式电子计算机 ENIAC。这台计算机用电子管实现,编程通过接插线进行,采用字长 10 位的十进制计数方式,每秒可进行 50 000 次加法运算。该机在 1943 年研制时,其最初目的是用于为军队编制各种武器的弹道表,1946 年后它经过多次改进,成为能进行各种科学计算的通用计算机。

到今天为止,电子计算机的发展已经历了四代,虽然在某些方面有一些突破,但其基本结构没有大的改变。这四个发展阶段以硬件进步为主要标志,但也包括了软件技术的发展。

第一代(1946 ~ 1957)——采用电子管为逻辑部件,以超声波延迟线、阴极射线管、

磁芯和磁鼓等为存储手段;软件上采用机器语言,后期采用汇编语言。

第二代(1957 ~ 1965)——采用晶体管为逻辑部件,用磁芯、磁盘作内存和外存;软件上广泛采用高级语言,并出现了早期的操作系统。

第三代(1965 ~ 1971)——采用中小规模集成电路为主要部件,以磁芯、磁盘作内存和外存;软件上广泛使用操作系统,产生了分时、实时等操作系统和计算机网络。

第四代(1971 ~ 至今)——采用大规模集成电路(LSI)、超大规模集成电路(VLSI)为主要部件,以半导体存储器和磁盘为内、外存储器;在软件方法上产生了结构化程序设计和面向对象程序设计的思想。

近年来,随着集成电路技术的不断提高,包括通用微处理器、面向控制和数字信号处理的专用微处理器、片上系统(System on Chip, SoC)、专用集成电路芯片(Application Specific Integrated Circuit, ASIC)、大规模现场可编程器件(Field-Programmable Gate Array/Complex Programmable Logic Device, FPGA/CPLD)等都获得了飞速发展,进一步促进了网络通信技术、多媒体技术、各种总线技术和因特网接入技术的发展,使计算机在整个社会应用中的广度和深度都得到了前所未有的提高。

1.1.2 微型计算机软件的发展

微型计算机系统软件的发展基本上与硬件的发展是一致的。在电子管时代,主要的微机软件是机器语言和汇编语言写成的完成简单功能的程序;在晶体管时代,主要的微机软件则由高级语言写成的监控程序和简单操作系统;在集成电路发展阶段,微机软件则发展到了功能较强的操作系统和高级会话式语言阶段;到了大规模或超大规模集成电路阶段,微机软件则发展到了更高级的阶段,即软件工程的研究与应用、高级数据库系统和语言编译系统和网络软件的发展应用阶段。

1.2 计算机中的数制和码制

数制是数的表示方法,在日常生活中,最常用的是十进制数。由于用电子器件表示两种状态比较容易实现,也便于存储和运算,因此在计算机中一般采用二进制数。因为二进制数书写格式冗长,不便阅读,所以,在程序设计中又往往使用十六进制数、八进制数、二一十进制数等。

1.2.1 数制

1. 十进制数

在程序设计中,人们广泛使用十进制数。十进制数的特点是:每一位有0 ~ 9这10种数码,故基数为10,高位权是低位权的10倍,加减运算的法则为“逢十进一,借一当十”。

2. 二进制数

在计算机内部,所有信息都以二进制数形式出现。二进制数的特点是:只有两个不同的数字,即0和1,因此基数为2,高位权是低位权的2倍,加减运算的法则为“逢二进一,借一当二”。

3. 十六进制数

十六进制数是二进制数的另一种书写格式。十六进制数的特点是:每一位有0~9和A~F这16种数码,因此基数为16,高位权是低位权的16倍,加减运算的法则为“逢十六进一,借一当十六”。

4. 八进制数

八进制数也是二进制数的另一种书写形式,把3位二进制数作为一组,每一组用等值的八进制数(实际上是十进制数中的0~7)来表示。八进制数的特点是:每一位有0~7这8种数码,因此基数为8,高位权是低位权的8倍,加减运算的法则为“逢八进一,借一当八”。

5. 二—十进制数

二—十进制数是计算机中十进制数的表示方法,二—十进制数用4位二进制数码表示1位十进制数,简称为BCD(binary coded decimal)码。

用4位二进制数编码表示1位十进制数,有多种表示方法,常用的是8421 BCD码,它的表示规则以及与十进制之间的等价关系见表1.1。

表 1.1 BCD 码与十进制之间的等价关系

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

例如: $(2735)_{10} = (0010\ 0111\ 0011\ 0101)_{\text{BCD}}$

BCD码是十六进制数的一个子集,1010~1111是非法BCD码。

上述几种进位计数制有以下共同点:

1. 每种计数制有一个确定的基数R。
2. 按“逢R进一”方式计数。

1.2.2 数值转换

1. 任意进制数→十进制数

二进制、十六进制以及任意进制的数转换为十进制数的方法较为简单,根据按权展开式把每个数位上的代码和该数位的权值相乘,再求累加和即可得到等值的十进制数。如:

$$(1011.101)_2 = 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-3} = (11.625)_{10}$$

$$(731)_8 = 7 \times 8^2 + 3 \times 8^1 + 1 \times 8^0 = (473)_{10}$$

$$(A7)_{16} = 10 \times 16^1 + 7 \times 16^0 = (167)_{10}$$

2. 十进制数→二进制数

当十进制数转换为二进制数,根据该十进制数的类型来决定转换方法。

(1) 十进制整数→二进制数

方法为:“除2取余”,即十进制整数被2除,取其余数,商再被2除,取其余数,……,直到商为0时结束运算,然后把每次的余数按倒序规律排列就得到等值的二进制数。如:

$$N = (47)_{10} = (101111)_2$$

运算过程为: $47 \div 2 = 23$ 余数=1 D_0

$23 \div 2 = 11$ 余数=1 D_1

$11 \div 2 = 5$ 余数=1 D_2

$5 \div 2 = 2$ 余数=1 D_3

$2 \div 2 = 1$ 余数=0 D_4

$1 \div 2 = 0$ 余数=1 D_5

所以 $N = D_5 D_4 D_3 D_2 D_1 D_0 = (101111)_2$

(2) 十进制小数→二进制数

方法为:“乘2取整”,即把十进制纯小数乘以2,取其整数(不参加后继运算),乘积的小数部分再乘以2,取整……,直到乘积的小数部分为0,然后把每次乘积的整数部分按正序规律排列,即为等值的二进制数。如:

$$N = (0.625)_{10} = (0.101)_2$$

运算过程为: $0.625 \times 2 = 1.25$ 乘积的整数部分=1 D_{-1}

$0.25 \times 2 = 0.5$ 乘积的整数部分=0 D_{-2}

$0.5 \times 2 = 1.0$ 乘积的整数部分=1 D_{-3}

所以 $N = D_{-1} D_{-2} D_{-3} = (0.101)_2$

有些纯小数,不断地“乘2取整”也不能使其乘积的小数部分为0,此时只能进行有限次运算,根据需要取其近似值。

(3) 十进制数→二进制数

方法为:整数部分“除2取余”,小数部分“乘2取整”,然后再进行组合。例如:

$$(47.625)_{10} = (101111.101)_2$$

3. 二进制数→十六进制数

以小数点为界,4 为二进制数为一组,不足 4 位用 0 补全,然后每组用等值的十六进制数表述。如:

$$(1101110.11)_2 = (0110\ 1110.1100)_2 = (6E.C)_{16}$$

在汇编语言中,十六进制数用后缀“H”表示。所以, $(1A2B)_{16}$ 应写成 1A2BH。

4. 十六进制数→二进制数

把十六进制数的每一位用等值的 4 位二进制数来替换,如:

$$(1B7.68)_{16} = (0001\ 1011\ 0111.0110\ 1000)_2 = (110110111.01101)_2$$

1.2.3 码制

两个名词:

真值数——符合人们使用习惯的数的原始表示形式。即用“+”、“-”表示的数。

机器数——数据在机器中的实际表示形式。即用符号位上的 0、1 来表示正、负的数。

机器数可以用不同的方法来表示,常用的有原码、反码、补码和移码表示。

1. 机器数的原码

对一个二进制数而言,若是最高位表示数的符号(常以 0 表示正数,1 表示负数),其余各位表示数值本身,则称为原码。如:

原码	真值数
+81=01010001	+1010001
-81=11010001	-1010001
+0=00000000	+0000000
-0=10000000	-0000000

可以看到,在八位机器字长表示数据的计算机中,用原码表示+81 和-81 的原码的低七位是相同的。原码简单,与真值转换方便。

2. 机器数的反码

正数的反码与其原码相同,最高位为 0 表示正数,其余位为数值位。负数的反码是其对应的正数连同符号位取反求得。如:

原码	反码
+81=01010001	01010001
-81=11010001	10101110
+0=00000000	00000000
-0=10000000	11111111

3. 机器数的补码

正数的补码与其原码相同, 负数的补码为其反码加 1, 即在其反码的最低位加 1 得到。

如:

原码	反码	补码
+81=01010001	01010001	01010001
-81=11010001	10101110	10101111
+0=00000000	00000000	00000000
-0=10000000	11111111	无

4. 机器数的移码

与前面的表示方法不同, 其符号位为 0 表示负, 为 1 表示正; 无论正数、负数还是零, 均可在补码表示的基础上通过将符号位取反来获得。如:

原码	反码	补码	移码
+81=01010001	01010001	01010001	11010001
-81=11010001	10101110	10101111	00101111
+0=00000000	00000000	00000000	10000000
-0=10000000	11111111	无	无

补码表示法的特点是: 使符号位参加运算、从而简化了加、减法的规则, 使减法运算转换为加法运算, 从而简化了机器的运算器电路。

补码加减法的规则是: $[X \pm Y]_{\text{补}} = [X]_{\text{补}} + [\pm Y]_{\text{补}}$ 。

例: 以 2^8 为模, 设 $[X]_{\text{补}} = 00000100$, $[Y]_{\text{补}} = 11110010$, 求 $[X+Y]_{\text{补}}$ 和 $[X-Y]_{\text{补}}$ 。

解: $[X+Y]_{\text{补}} = [X]_{\text{补}} + [Y]_{\text{补}} = 00000100 + 11110010 = 11110110$ (-10 的补码)

$[X-Y]_{\text{补}} = [X]_{\text{补}} + [-Y]_{\text{补}} = 00000100 + 00001110 = 00010010$ (18 的补码)

注意: 在两个带符号的二进制数进行补码运算时, 若运算结果的绝对值超过运算装置的容量, 数值部分便会发生溢出, 占据符号位的位置, 从而引起计算出错。这和补码运算过程中的正常溢出(符号位的进位)性质上是不同的。正常溢出是以 2^n (n 为二进制数的位数) 为模的溢出, 它被自然丢失, 不影响结果的正确性。例如以 2^5 为模, 除符号位外有 4 位用来表示数值, 补码运算中的溢出和符号位的进位例如:

$01101\text{B}(13) + 00111\text{B}(7) = 1\ 0100\text{B} = -12$ ——溢出

$11100\text{B}(-4) + 11100\text{B}(-4) = 1\ 11000\text{B} = -8$ ——符号位进位, 自认丢失

任何一种运算都不允许发生溢出, 除非是只利用溢出作为判断而不使用所得的结果, 所以当溢出产生时, 应使计算机停机或进入检查程序找出溢出原因, 然后作相应处理。

微型计算机中常用的溢出判别法是双高位判别法。

C_s : 表示最高位(符号位)的进位情况, 如有进位, $C_s = 1$, 否则, $C_s = 0$

C_r : 表示数值部分最高位的进位情况, 如有进位, $C_r = 1$, 否则, $C_r = 0$

设微型计算机的字长为 8, 例:

$$\begin{array}{r} 01011010\text{B}(+90)_{\text{补}} \\ +01101011\text{B}(+107)_{\text{补}} \\ \hline \end{array}$$

$$11000101\text{B}(-59)_{\text{补}}$$
 $C_s=0, C_p=1$, 正溢出, 结果出错

$$\begin{array}{r} 00101101\text{B}(+45)_{\text{补}} \\ +00101101\text{B}(+45)_{\text{补}} \\ \hline \end{array}$$

$$01011010\text{B}(+90)_{\text{补}}$$
 $C_s=0, C_p=0$, 无溢出, 结果正确

$$\begin{array}{r} 10010010\text{B}(-110)_{\text{补}} \\ +10100100\text{B}(-92)_{\text{补}} \\ \hline \end{array}$$

$$100110110\text{B}(+54)_{\text{补}}$$
 $C_s=1, C_p=0$, 负溢出, 结果出错

$$\begin{array}{r} 11111110\text{B}(-2)_{\text{补}} \\ +11111110\text{B}(-2)_{\text{补}} \\ \hline \end{array}$$

$$111111100\text{B}(-4)_{\text{补}}$$
 $C_s=1, C_p=1$, 无溢出, 结果正确

结论:

在微型计算机中, 只有当溢出 C_s 和 C_p 的状态不同时(为 01 或 10 两种状态), 产生溢出。而当 C_s 和 C_p 的状态相同时(为 00 或 11 两种状态)则不会产生溢出。

1.3 微型计算机系统组成

从系统组成的观点来看, 硬件是构成计算机的设备实体, 软件是指为了运行、管理和维护计算机而编写的各种程序。

1.3.1 微型计算机系统的硬件

微型计算机系统的硬件(如图 1.1 所示)主要包括 CPU(中央处理器包括运算器和控制器)、存储器、I/O(输入/输出)设备及其接口电路。

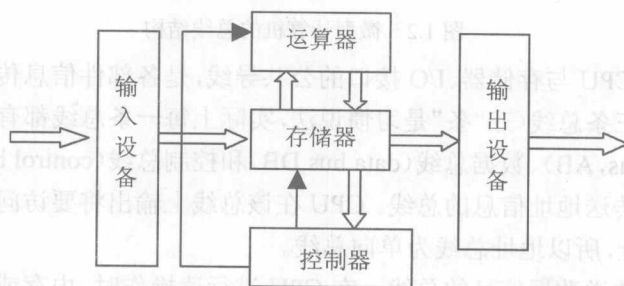


图 1.1 微型计算机系统的硬件组成

1. 微处理器

微处理器简称 CPU, 是用来实现运算和控制功能的部件, 由运算器、控制器和寄存器 3 部分组成。运算器是进行算术运算和逻辑运算的部件, 它是指令的执行部件; 控制器是计算机的指挥中心, 它负责对指令进行译码, 产生出整个指令系统所需要的全部操作的控制信号, 控制运算器、存储器、输入/输出接口等部件完成指令规定的操作。

2. 存储器

图 1.1 中的存储器是指微型计算机的内存储器。它通常由 CPU 之外的半导体存储器芯片组成,用来存放程序、原始操作数、运算的中间结果数据和最终结果数据,简称内存。硬盘、软盘被称为外存储器,即辅助存储器。

3. 输入/输出设备

输入/输出设备统称为外部设备,是微型计算机的重要组成部分。输入设备通过输入接口电路将程序和数据输入内存。最常见的输入设备是键盘;输出设备是 CPU 通过相应的输出接口电路将程序运行的结果及程序、数据送到的设备。最常见的输出设备有显示器和打印机。

4. 微型计算机的总线结构

微型计算机的总线结构如图 1.2 所示。

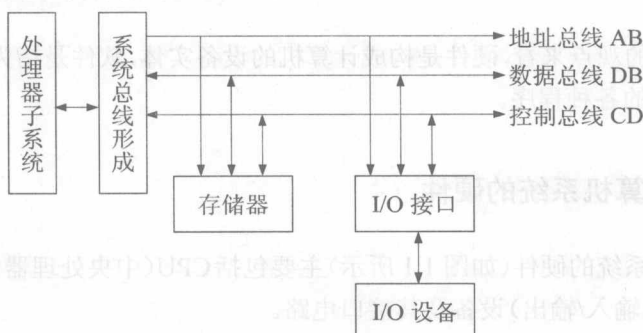


图 1.2 微型计算机的总线结构

总线是连接 CPU 与存储器、I/O 接口的公共导线,是各部件信息传输的公共通道。微型计算机系统有三条总线(“三条”是习惯说法,实际上每一条总线都有若干根),它们是地址总线(address bus, AB)、数据总线(data bus, DB)和控制总线(control bus, CB)。

地址总线是传送地址信息的总线。CPU 在该总线上输出将要访问的内存单元的地址或 I/O 端口的地址,所以地址总线为单向总线。

数据总线是传送数据信息的总线。在 CPU 进行读操作时,内存或外设的数据通过数据总线送往 CPU;在 CPU 进行写操作时,CPU 数据通过数据总线送往内存或外设;所以数据总线为双向总线。

控制总线是传送控制信息的总线。控制信号用于协调系统中各部件的操作。其中,有些信号将 CPU 的控制信号或状态信号送往外界;有些信号线将外界的请求或联络信号送往 CPU;个别的信号线兼有以上两种情况。

1.3.2 微型计算机系统的软件

微型计算机软件系统由系统软件和应用软件组成。系统软件是指管理、控制和维护计算机的各种硬件资源,以及扩大计算机功能和方便用户使用计算机的各种程序的集合,它是构成计算机系统必备的软件。系统软件分为操作系统、语言处理程序、工具软件和数据库管理系统4类。应用软件是为了解决各种实际问题而设计的计算机程序,通常由计算机用户或软件公司开发。

1.3.3 微型计算机系统的性能

衡量一台微型计算机性能的优劣,主要由它的系统结构、硬件结构、系统总线以及软件配置等因素。具体体现在这样几个方面:

1. 主频

微型计算机的时钟频率在一定程度上反映了机器的运算速度,所以很多时候也用指令执行时间来衡量机器的主频。主频越高,微型计算机执行指令的时间越短,机器的速度越快。主频的单位是 MHz(兆赫兹),指令执行的单位是 MIPS(每秒运行多少百万条指令)。

2. 存储容量

存储容量是衡量微型计算机内部存储器能存储二进制信息大小的一个技术指标。通常以字节为单位,存储容量越大,储存程序和数据的能力就越强,相应的处理能力就越强。

3. 字长

微型计算机的字长是指微处理器内部一次能够并行处理二进制代码的位数。它与微处理器内部寄存器以及 CPU 内部总线宽度是一致的,字长越长,所表示的数据精度就越高,一次能够同时处理的信息量就越大。

4. 外部设备的扩展能力

在微型计算机系统中,外部设备占据了重要的地位。计算机信息的输入、输出、存储都必须由外设来完成。微型计算机的外设配置能力,其速度快慢、容量大小、分辨率高低等技术指标都影响着微机系统的整体性能。

5. 软件配置能力

微型计算机系统软件系统是计算机系统不可缺少的组成部分,微机如果缺少了软件系统,就成为了一个无用的裸机。系统软件配置是否齐全,软件功能强弱,是否支持多任务、多用户操作等都是微机硬件系统性能是否得到充分发挥的重要因素。

1.4 当前微型计算机所使用的先进技术

随着微型计算应用领域的进一步扩大,微机的各项技术也在发生巨大的变化,当前微型计算机所使用的先进技术主要有这样几个方面。

1.4.1 CACHE 技术

高速缓存(Cache)是在相对容量较大而速度较慢的主存 DRAM(Dynamic Random Access Memory)与高速处理器之间设置的少量但快速 SRAM(Static Random Access Memory)组成的存储器。Cache 中复制着主存的部分内容(通常是最近使用的信息)。当 CPU 试图读取主存的某个字时,Cache 控制器首先检查 Cache 中是否包含有这个字。若有,则 CPU 直接读取 Cache 而不必访问主存,这种情况称为“命中”;若无,则 CPU 读取主存中包含此字的一个数据块,将此字送入 CPU,同时将此数据块传送到 Cache,这种情况称为“未命中”。

1.4.2 流水线技术

指令流水线的思想类似于现代化工厂的生产流水线。把一个复杂的指令分解成若干个步骤;每个步骤用同样的单位时间,在各自的单位时间内,完成各自步骤的工作。在简单的情况下,可以将指令执行过程分成取指令(Fetch Instruction)和执行指令(Execute Instruction)2 个步骤。在执行指令时,可以利用 CPU 不使用存储器的时间取指令,实现这 2 个步骤的并行操作,这就是所谓的“指令预取”(Instruction Prefetch)。

1.4.3 VM 技术

虚拟存储器(Virtual Memory, VM)是为满足用户对存储器空间不断扩大的要求而提出来的,允许用户将外存看成是主存储器的扩充,即虚拟一个比实际主存储器大得多的存储系统。这是一个概念性的逻辑地址,并非是实际空间地址。虚拟存储系统是在主存和辅存之间,通过存储管理单元 MMU(Memory Manage Unit),进行虚地址和实地址的自动变换而实现的,对应用程序是透明的。

1.4.4 RISC 技术

RISC(Reduced Instruction Set Computer)技术起源于 20 世纪 70 年代初期,1982 年美国加州大学伯克利分校的 Patterson 等人研制成功了第一个 RISC 处理器芯片 RISC-I,随后又完成了 RISC-II 32 位微处理器。在此之后, RISC 技术得以推广,并在高档的工程工

作站得到广泛应用。RISC 是这样一种计算机:指令系统很简单,只有少数简单、常用的指令;指令简单可以使用使处理器的硬件也简单,能够比较方便地实现优化,使每个时钟周期完成一条指令的执行,并将时钟频率提得很高;这样使整个系统的总性能达到很高,有可能超过指令庞大复杂的计算机。

RISC 技术的主要特点有:

- 功能简单、数量有限的指令系统。
- 大量通用的寄存器,通过编译技术优化寄存器的使用。
- 通过优化指令流水线提高性能。

1.4.5 多内核技术

多内核技术就是在单个物理处理器中包含多个处理器的内核逻辑。例如,在一枚英特尔处理器中封装两枚(或更多)英特尔处理器的所有电路和逻辑。多内核技术将多个处理器“内核”放置并封装为单个物理处理器。该技术旨在支持系统同时运行更多项任务,由此实现更为出色的整体系统性能。

习 题

1. 微型计算机主要由哪些基本部件组成?各部件的主要功能是什么?
2. 微型计算机的发展过程是什么?
3. 简述计算机中“数”与“码”的区别,计算机中常用的数制和码制各有哪些?
4. 请完成以下计算:

$$174.66D=(\quad)B=(\quad)H$$

$$10001110101.01011B=(\quad)D=(\quad)H$$

$$3ACH=(\quad)B=(\quad)BCD$$

5. 设字长=8 位, $X=(1A)_{16}$, 当 X 分别为原码、补码、反码和无符号数的时候,其真值=()。

6. 设字长为 8 位,用补码形式完成下列计算,要求有运算结果并讨论是否发生溢出?

$$120+18 \qquad -33-37$$

$$-90-70 \qquad 50+84$$

7. 衡量计算机性能的指标有哪些?
8. 当代计算发展的主要核心技术有哪些?

第2章 80x86 微处理器

随着半导体技术的发展,集成在处理器芯片上的元器件越来越多。1971 年 Intel 公司获得突破性进展,研制出 Intel 4004 芯片。Intel 4004 是第一个将 CPU 的所有元器件包含在单个芯片上的芯片,从而微处理器就这样诞生了。

Intel 4004 能够完成两个 4 比特的数字相加,能够通过重复进行加法实现乘法操作。它的时钟频率在 108kHz,每秒执行 6 万条指令。尺寸规格为 3mm×4mm,计算性能远远超过当年的 ENIAC。

微处理器的发展的第二个主要阶段是 1972 年 Intel 研制出 8008 微处理器。这是第一个 8 位微处理器,复杂度几乎是 4004 的两倍。

Intel 4004 和 8008 都是专用微处理器,是针对专门应用的。它们属于第一代微处理器。1974 年 Intel 研制出第一个通用微处理器 Intel 8080,它也是 8 位微处理器,属于第二代微处理器。它的指令集比 8008 更加丰富,寻址空间更大,处理速度为 0.64MIPS。

当时,Zilog、Motorola 和 Intel 在微处理器领域三足鼎立。Zilog 公司于 1976 年对 8080 进行扩展,开发出 Z80 微处理器,广泛用于微型计算机和工业自动控制设备。直到今天,Z80 仍然是 8 位处理器的巅峰之作,还在各种场合大卖特卖。CP/M 就是面向其开发的操作系统。许多著名的软件如:WORDSTAR 和 DBASE II 都基于此款处理器。

几乎同时,Intel 也开始研制 16 位的微处理器。不过直到上世纪 70 年代末,功能更强的、通用的 16 位微处理器才出现。其中之一是 Intel 8086,其频率为 4.77MHz,标志着第三代微处理器问世。不过当时由于 360 美元的售价过于昂贵,大部分人没有足够的钱购买使用此芯片的电脑,于是 Intel 在 1 年之后推出 4.77MHz 的 8 位微处理器 8088。IBM 公司 1981 年生产的第一台电脑就是使用的这种芯片。这也标志着 x86 架构和 IBM PC 兼容电脑的产生。发布的时候,8086 的时钟频率有 4.77MHz,8MHz 和 10MHz 三个版本,包括了具有 300 个操作的指令集。其中 8MHz 版本包含了大约 28 000 个晶体管,具备 0.8 MIPS 的能力。8088 在性能方面,它在内部以 16 位运行,但支持 8 位数据总线,采用现有的 8 位设备控制芯片,包含 29 000 个 3 微米技术的晶体管,可访问 1MB 内存地址,速度为 0.33MIPS。

1982 年 2 月 1 日,Intel 发布了 80286,它也是 Intel 公司的第三代 16 位微处理器。80286 处理器集成了大约 13.4 万个晶体管,最大主频为 20MHz,采用 16 位数据总线和 24 位地址总线。与 8086 相比,80286 增加了实存(24 位地址)和虚拟存储器管理,可以在两种不同的模式下工作:一种叫实模式,另一种叫保护方式。80286 开始正式采用一种被称为引脚网格阵列(PGA)的正方形包装。