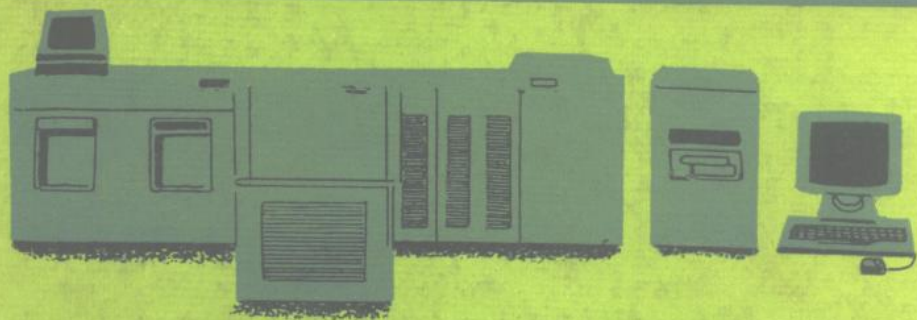


# 计算机体系结构技术

董占球 刘邦君 编著



科学出版社

# 计算机体系结构技术

董占球 刘邦君 编著

科学出版社

1987

## 内 容 简 介

本书将迄今为止有代表性的计算机在体系结构上的特点抽取出来，归纳为寄存器、微程序、存储、堆栈、寻址、总线、通道、接口、中断、容错十大关键技术，分别予以较为深入的讨论，对现代计算机发展有重要影响的体系结构技术，讨论尤为详尽。本书可供计算机专业的本科生、研究生及计算机研制、维护人员参考。

## 计算机体系结构技术

董占球 刘邦君 编著

责任编辑 黄岁新

科学出版社出版

北京朝阳门内大街137号

中国科学院印刷厂印刷

新华书店北京发行所发行 各地新华书店经售

1987年10月 第一版 开本：850×1168 1/32

1987年10月第一次印刷 印张：11 3/4

印数：精 1—2,000 插页：精 2

平 1—5,300 字数：302,000

统一书号：15031·846

本社书号：5122·15-8

定价：布脊精装 4.45 元  
平 装 3.35 元

## 序

人类制造计数、计算工具的历史可以追溯到几千年以前,只有到了十七世纪,随着数学、物理学、天文学、机械制造等科学技术的发展,才产生了能进行机械演算的帕斯卡尔机。又经历了三百年,新兴的电子学和深入发展的数学才将第一台电子数字计算机推上了历史舞台。

冯·诺伊曼总结了前人的经验,提出了存储程序这个极为重要的概念,开辟了计算技术迅速发展的新时期。近四十年来,计算技术的发展完全可以和原子能、半导体、航天技术、仿生学等科技领域的佼佼者相媲美,它们相互促进、彼此推动,为人类认识自然、改造自然提供了强大的动力。今天,电子计算机已经深入到社会的各个领域,成为信息时代的重要标志之一,成为研究自然科学和社会科学必不可少的工具。

计算机越是促进别的科学技术的蓬勃发展,就越会遇到更新、更难、更大量的问题,而实际应用的需要又成了计算机发展的动力。今天,冯·诺伊曼结构的计算机,已经发展得相当成熟,不仅仅表现在存储容量、计算速度上,更主要地表现在设计、组织、运行、使用冯·诺伊曼结构计算机上,已经形成了一整套方法。

尽管冯·诺伊曼结构计算机随着器件和部件的改进、新技术的采用、使用范围的扩大,经历了四代的发展,出现了纷繁复杂、性能各异的机种,但其基本结构并没有多大变化。这种结构已经越来越明显地表现出它的局限性,这种局限性又成了计算机进一步发展的桎梏。只有冲破这种束缚,计算机才有可能出现新的飞跃。正因为如此,一场大的变革将在预料之中。

在这样一种形势下,计算机工作者应该给自己提出什么样的要求呢?加倍努力固然是重要的,但更应该清醒地思考,认真总结

前人的成果,深入分析已有的教训,升华实践经验,探索客观规律,跳出迷津,从而有可能获得新的发现、开始新的突破。本书的作者正是希望在这方面作一次尝试。

新的科学技术的诞生绝不是突然从天而降,它必须依赖于已有的科学技术的哺育,电子计算机的产生和发展也正是如此。最初,人们把电子计算机可能看作电子学的一个副产品,也可能看作数字演算的一种新机器,就象一个襁褓中的婴儿。但是,它今天已经强壮起来,不仅被看作一种成功的技术,而且越来越多的人认为它可以被列入科学的领域,计算机正跃跃欲试地要跻身于世界科学技术之林,成为一门独立的、有强大影响的新型科学技术。

根据生产实践经验和自然科学原理而发展成的各种工艺操作方法与技能称为“技术”,研究客观世界发展过程的某一阶段或某一种运动形式,揭示事物发展的客观规律;关于自然、社会和思维的知识体系被称为“科学”,二者是紧密联系但又有严格区别的概念。计算机技术早就被人们所承认,但今天,是否对人脑计算的运动形式已经研究得很清楚了呢?是否有更符合高速计算、推理、判断客观规律的非冯·诺伊曼计算机结构呢?把计算机知识体系和数学、物理学一样称之为“科学”是否有十分充足的理由呢?这些问题不能简单地回答是与否,而应该努力去探索尚未发现的证据。本书正是想透过具体机种这些庞杂的现象,抽出本质的、能反映某些客观规律的内容,进行比较深入的、有条理的阐述,这或许是有助于上述的探索。

计算机体系结构所包含的内容应该不同于计算机原理,也应该不同于计算机发展史,它应该阐明电路层、部件层、系统层、应用层的内部以及它们之间,是如何配合、组织,如何互相联系、互相制约,而构成一个计算机整体的。而且,人机界面也是体系结构必须考虑的重要因素。研究计算机体系结构的不是为了模仿,而是为了在描述、定义、设计、组织一个实现某种算法的计算机系统时,既能遵循某些必要的规律,又能运用自如。因此,本书作者没有采用断代的方法或者机型分类的方法来介绍体系结构,而是采

用抽取、提炼的方法,将迄今为止出现过的代表机型在体系结构上的特点和共性,归纳为十大核心技术。这种阐述方法也许是比较清晰、比较新颖的。

不管是设计一个逻辑电路,还是设计一个庞大的计算机系统,都可能因人而异。尽管他们所达到的目标可能完全相同,但其具体实施方案在不同程度上是有主观随意性的。但是,这种随意性是要受到某些基本规律制约的,而绝不是可以异想天开的。本书所阐述的计算机体系结构十大技术,就是希望在寻求基本规律上做一次大胆的试验。

本书作者之一董占球同志,是我国自己培养的第一代计算机科技工作者,从五十年代起,在电路、部件上做过许多研究工作,参加了我国第一台电子计算机的研制,以后还参加过一些计算机系统的研制及应用、开发工作。两位作者都从事过比较长期的部件研究和实际系统运行、维护、应用工作,对不同的计算机作过一定的分析、比较、评价。这本书就是结合他们多年来的实践经验和积累的大量资料编著而成的。

正是由于以上的原因,这是一部有一定水平与实用价值的书。对学习、研究计算机体系结构的同志,定会有所启发;对设计计算机系统、部件的同志,也可提供参考。希望该书能够对正处于徘徊中的我国计算机事业起到促进其发展的作用。

范新弼

1985.9.28.

# 目 录

<b>第一章 引言</b> .....	1
1.1 概念 .....	1
1.2 内容 .....	2
1.3 回顾 .....	4
1.4 评论 .....	20
1.4.1 应熟悉计算机的发展史 .....	20
1.4.2 应了解计算机的局限性 .....	22
1.4.3 应了解并行处理的局限性 .....	22
1.4.4 附带说明 .....	23
<b>第二章 计算机体系结构的一般问题</b> .....	25
2.1 信息表示 .....	25
2.1.1 位及位串 .....	25
2.1.2 字节 .....	26
2.1.3 字 .....	27
2.2 数据格式 .....	29
2.2.1 关于补码 .....	29
2.2.2 关于十进制 .....	30
2.2.3 关于浮点数 .....	32
2.3 指令系统 .....	33
2.3.1 指令类型 .....	34
2.3.2 指令中操作数的确定 .....	41
2.3.3 有关指令的其它问题 .....	44
2.4 关于补码乘法和浮点运算 .....	46
2.4.1 补码乘法 .....	46
2.4.2 浮点运算 .....	47
2.5 并行处理 .....	50

2.5.1	流水线工作方式	51
2.5.2	多部件并行处理	58
2.5.3	流水线式向量计算机	59
2.6	评论	61
<b>第三章</b>	<b>寄存器技术</b>	64
3.1	一般特征	64
3.2	CDL 的简单介绍	66
3.3	寄存器的应用	70
3.3.1	寄存器级在控制部分的应用	70
3.3.2	寄存器级在处理部分的应用	75
3.4	寄存器级设计技术	78
3.4.1	寄存器级部件的设计	79
3.4.2	寄存器级线路的设计	87
3.5	评论	88
<b>第四章</b>	<b>微程序技术</b>	92
4.1	概述	92
4.2	关于微程序的若干问题	95
4.2.1	微指令格式	95
4.2.2	微指令的地址	98
4.2.3	微命令的定时	102
4.3	微程序的设计	104
4.4	微指令长度最小化的问题	108
4.4.1	决定MCC	108
4.4.2	决定最小MCC覆盖	110
4.4.3	决定最小化微指令字长W的覆盖	112
4.5	评论	114
<b>第五章</b>	<b>存储技术</b>	115
5.1	存储技术中的速度问题	116
5.1.1	多寄存器技术	116
5.1.2	增加带宽	118
5.1.3	交叉存取技术	118



5.1.4	高速缓冲存储器	121
5.1.5	联想式存储器(相联存储器)	125
5.1.6	慢速大容量存储器的存放	128
5.1.7	小结	131
5.2	存储技术中的容量问题	132
5.2.1	虚拟存储器	134
5.2.2	信息转换中的调度策略	143
5.2.3	信息转换中的分配策略	144
5.3	评论	145
<b>第六章</b>	<b>堆栈技术</b>	147
6.1	堆栈的概念和工作原理	147
6.2	堆栈用于子程序链接	150
6.3	堆栈用于中断链接	152
6.4	堆栈用于再入性程序	153
6.5	堆栈式计算机	159
6.6	面向高级语言的计算机	162
6.7	评论	164
<b>第七章</b>	<b>寻址技术</b>	166
7.1	立即操作数	167
7.2	操作数在寄存器中	168
7.3	操作数在存储器中	169
7.3.1	直接相对寻址	171
7.3.2	自动递增方式	172
7.3.3	自动递减方式	173
7.3.4	立即变址	173
7.3.5	隐含操作数地址寻址	174
7.4	间接寻址	174
7.5	动态地址变换	177
7.5.1	逻辑地址及其控制	177
7.5.2	变换表及变换过程	178
7.5.3	地址变换缓冲器	181

7.6	非连续编址的地址变换 .....	183
7.6.1	地址转换表法 .....	185
7.6.2	找零取整法 .....	185
7.7	评论 .....	187
<b>第八章</b>	<b>总线技术 .....</b>	<b>188</b>
8.1	概述 .....	188
8.2	单总线的构成 .....	192
8.2.1	单总线的连线 .....	192
8.2.2	单/双向线 .....	193
8.3	单总线的工作方式 .....	195
8.3.1	主-从关系 .....	195
8.3.2	互锁通讯 .....	196
8.4	单总线的定时 .....	196
8.5	总线互连设备间优先级的控制 .....	198
8.5.1	优先级控制 .....	198
8.5.2	总线优先次序链 .....	201
8.6	按优先权转移 .....	202
8.7	以总线组织系统的体系结构 .....	204
8.8	评论 .....	207
<b>第九章</b>	<b>接口技术 .....</b>	<b>208</b>
9.1	概述 .....	208
9.2	数据终端与数据通信设备的接口标准 .....	210
9.3	RS-232-C 串行接口——EIA 标准 .....	214
9.4	基本接口 .....	217
9.4.1	接口操作 .....	218
9.4.2	接口程序设计 .....	219
9.5	程序控制接口 .....	219
9.5.1	接口操作 .....	219
9.5.2	接口程序设计 .....	221
9.6	中断服务接口 .....	222
9.6.1	接口操作 .....	222

9.6.2	接口程序设计	223
<b>9.7</b>	<b>直接内存访问(DMA)接口</b>	<b>224</b>
9.7.1	接口操作	225
9.7.2	接口程序设计	227
<b>9.8</b>	<b>接口的实现</b>	<b>227</b>
9.8.1	三态推挽驱动器	227
9.8.2	线或型发送器	228
9.8.3	地址选择	229
9.8.4	一个接口实例	232
<b>9.9</b>	<b>评论</b>	<b>233</b>
<b>第十章</b>	<b>中断技术</b>	<b>234</b>
10.1	概述	234
10.2	中断类型	236
10.2.1	机器校验中断	236
10.2.2	访管中断	237
10.2.3	程序中断	238
10.2.4	外部中断	239
10.2.5	输入/输出 (I/O)中断	239
10.2.6	重新启动中断	239
10.3	中断识别	240
10.3.1	隐指令方式	240
10.3.2	中断矢量方式	242
10.3.3	程序状态字方式	242
10.4	中断点的选择	243
10.5	中断机构	244
10.5.1	中断屏蔽、开放和关断	245
10.5.2	中断的优先级及其控制	249
10.6	中断的进入和退出	258
10.7	评论	259
<b>第十一章</b>	<b>通道技术</b>	<b>260</b>
11.1	概述	260

11.2	通道的功能及组成	262
11.3	通道的类型及操作方式	264
11.4	I/O设备的控制	269
11.4.1	I/O 指令	269
11.4.2	I/O 设备的编址	273
11.4.3	I/O 的系统状态	273
11.4.4	I/O 系统的清除	275
11.5	I/O操作的执行	276
11.5.1	通道地址字	276
11.5.2	通道命令字	277
11.5.3	通道状态字	279
11.5.4	I/O 操作的启动	282
11.6	通道与 ISP 的调度策略	282
11.7	评论	285
<b>第十二章</b>	<b>容错技术</b>	<b>287</b>
12.1	概述	287
12.2	分析故障逻辑网络的“结构与类似观测输出函数(SPOOF)”	289
12.3	信息传送、保存过程中的自动检错、纠错	295
12.3.1	海明码	296
12.3.2	多项式码	302
12.4	错误诊断	304
12.5	动态系统重构和功能复执	308
12.6	容错系统中的软件问题	316
12.7	容错计算系统的得失估计问题	318
12.8	评论	319
<b>第十三章</b>	<b>回顾与展望</b>	<b>321</b>
13.1	概述	321
13.2	关于流水线技术	324
13.3	关于阵列技术	331
13.4	关于数据流技术	339

13.5 关于硅基 VLSI 的发展及其对体系结构可能产生的影响.....	348
13.5.1 VLSI 技术在处理部件中的进展.....	348
13.5.2 VLSI 技术在微处理机中的进展.....	350
13.5.3 VLSI 技术在系统级的进展.....	350
13.6 评论.....	351
<b>参考文献</b> .....	353
<b>英文缩写、全称对照表</b> .....	355
<b>中文、外文人名对照表</b> .....	358
<b>后记</b> .....	360

# 第一章 引言

## 1.1 概念

是否可以这样认为：计算机技术是人类智力的延伸。

迄今为止，人类对于自身智力的规律性尚未充分认识，因此，对于作为其延伸的计算机技术的内在规律性自然地也就尚处于探索的时期，所以要象在传统科学中那样地使用公理、定理、定律，用解析的方法讨论问题，似乎尚有一些困难。甚至对问题本身作一 定义，有时也是不容易的。关于计算机体系结构这一概念，至今有各种各样的理解，很难有一个通用的定义。

我们倾向于这样的理解：

计算机体系结构是为实现某种或某些算法，而描述、定义、设计、组织一个完整的计算机系统。而所有这一切应当实现图 1.1 所示的目标。

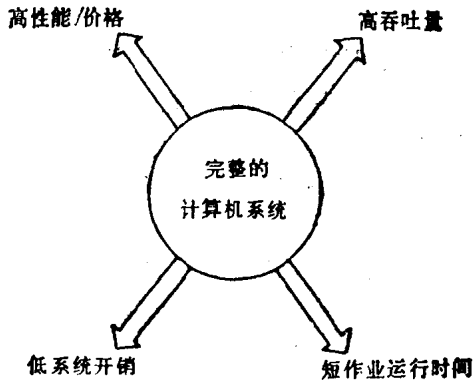


图 1.1 计算机体系结构追求的目标

## 1.2 内 容

读者从图 1.1 会直观地认识到：计算机体系结构所涉及的内容是十分广泛的。概略地说，计算机系统的所有问题，无一不是计算机体系结构关心的对象。

试图在一本书中对所有的内容进行有系统、有深度的讨论是不可能的，至少是困难的，因而本书的讨论将偏重于计算机系统的硬件方面，然而这仍然是一个非常广泛的领域。

作者的意图是：摆脱众所周知的分类学研究，不局限于某些所谓的典型对象机种，而将迄今为止在计算机体系结构中出现过的硬件技术抽取出来，予以较深入的讨论。目的是使读者在阅读本书后，能自行得出一些如何定义、描述、设计、组织计算机系统的基本技术。读者可能会发现：本书中所讨论的这些技术或多或少地出现在迄今为止任何一台实际的计算机中。作者希望能达到：无论是研制一台计算机，配置一个合理的计算机系统，还是有效地运行一个完整的计算机系统，本书的内容都将会起到某些参考作用。

因此，本书将要讨论的内容是：

●计算机体系结构的一般性问题。这是指计算机共同的本质性问题。

●寄存器技术。这是出现在计算机指令处理基本功能级的基本技术，当然，也出现在诸如接口之类的非指令处理结构中。

●微程序技术。这是涉及到功能部件的控制和组织的技术，显然，它是一种时序逻辑；另一方面，部件的控制和组织还有硬接线逻辑。

●存储技术。这是计算机中一个十分重要的组成部分。众所周知，任何算法的实现都将需要信息存储，进而言之，存储问题将会使某些有空间复杂性的算法难以实现。

●堆栈技术。它可以看成存储技术的一个子集。采用堆栈技术，对实现调用子程序和处理中断，以及程序嵌套，是十分有效的。

●寻址技术。寻址技术是与存储技术直接相关的。是否可以说：任何一台实际运行的计算机系统，其大部分的活动是寻址。这是显而易见的，因为实现任何算法总是要不断地取出指令、存取数据，或者有时要更改进程，所以寻址技术将在相当大的程度上影响一个计算机系统的功能或效能。

●总线技术。这是在一个计算机系统中，为各个子系统提供信息通路的技术。换言之，是把各个独立的子系统组织为一个计算机系统的技术。不同的总线结构首先直接地影响到计算机系统的吞吐量，逻辑地推论它，亦将影响到系统的其它功能或性能。

●通道技术。这是在稍大的计算机系统中实现指令系统处理机和I/O（输入输出）设备连接的技术。这里往往是一个计算机系统中可能会出现少数几个瓶颈口之一。

●接口技术。这是把各种各样的I/O设备，即使它们彼此之间有极大的差异性，也可方便地连接到计算机系统中的有用技术。这一技术对于那些从事各种各样实时控制领域工作的读者是特别有用的。

●中断技术。我们是否可以稍微夸张地说：如果没有中断技术，将不会有近代意义的计算机系统。读者在阅读中断技术的内容时会发现：I/O设备的工作；由于算法本身的需要，而要改变程序的当前进程；对计算机内部或外部故障的处理；等等，如果没有中断技术会出现何种局面？

●容错技术。顾名思义，读者会发现这是提高计算机系统的可靠性、可用性的十分重要的技术。众所周知，错误校正编码这样的容错技术在近代计算机系统中已是十分普遍的。

所有前所述及的内容绝不是彼此无关的，而且主要取材于国内外的一些重要著作及文献，因而在讨论中前后交叉的论述方式是不可避免的。读者正应该以此为契机，对本书中讨论的内容做前后连贯，发现它们之间的内在联系。

综观前述，在阅读本书时并不要求读者具有相当的数学修养，然而在对某些方面作出性能评价时或许会用到有限的数学知识，



特别是概率统计方面的知识。同时,由于本书内容的牵涉,我们希望读者在诸如汇编语言、程序设计、逻辑设计及其工程化等方面有相当熟悉的知识。当然,有关操作系统及编译技术方面的知识也是需要的。更为理想的是:阅读本书的读者最好能有完整地运行过一个计算机系统的经历。必须说明,这里列举的仅是充分条件,而不是必要条件,必要条件是读者应该有:清楚、牢固的概念,清晰、逻辑的思维,熟练的基本技巧。就作者的经验而言,这是一个合格的计算机体系结构设计者最为重要的素质。

### 1.3 回 顾

我们追溯计算机的过去,目的是为了引出后续各章将要深入讨论的问题,同时也为读者提供一个了解本学科发展过程的某些情况。

远在1642年,法国科学家布莱斯·帕斯卡尔(Blaise Pascal)就制造了加减法的机械计算器。在这台机器中有一点是值得注意的,即负数采用补码表示。众所周知,补码表示法的优点是,0000的补码还是0000,这在执行条件转移指令时,或当要形成某种判据而要实现零测试指令时,就比原码(“+0”为0000,“-0”为1000)以及反码(“+0”为0000,“-0”为1111)要方便得多。其次,补码加法可以将符号位与数值位同样处理,因为补码的形式使得模2加法只要舍去符号位产生的进位,则包括符号位在内的所有数位均可统一处理,近代计算机中负数都是采用补码形式表示的。

被有的作者誉为计算机之父的英国人查尔斯·巴贝(Charles Babbage),在1823年基于任何连续函数都可用多项式逼近这一事实,设计了一个仅有加法,而通过有限差分法技术来计算这类函数的差分机。这也许是所有近代计算机的算术运算能力只有四则运算的发端,然而查尔斯·巴贝最重大的贡献,是通过他在1834年设计的分析机而体现的,在这台机器中有两点是十分重要的:

第一,它包含了后来的计算机的全部基本特点,这些特点在实