

高等学校
电子信息类 规划教材

计算机系统结构

(第三版)

李学干 编著

西安电子科技大学出版社

内 容 简 介

本书讲述计算机系统结构的基本概念、基本原理、基本结构、基本分析方法以及近年来的重要进展。

全书共分 8 章。第 1 章讲述计算机系统层次结构,计算机系统结构、组成、实现的定义和相互关系,软硬取舍原则及设计方法,软件移植手段,应用与器件对系统结构的影响,并行性发展与计算机系统分类。第 2 章讲述数据表示,寻址方式,指令系统的设计与改进,RISC 技术。第 3 章讲述总线,中断系统,通道处理机和外围处理机。第 4 章讲述存储体系,虚拟存储器,Cache 存储器,主存保护。第 5 章讲述重叠,流水,向量处理机,超标量、超长指令字、超流水线处理机。第 6 章讲述并行处理机和相联处理机。第 7 章讲述多处理机的硬件结构、程序并行性、性能分析和操作系统。第 8 章讲述脉动阵列机,数据流机,归约机,智能机,大规模并行处理机(MPP)和机群系统。

本书内容丰富,每章均有大量习题,可作为计算机专业本科生和有关专业研究生的教材,也可作为科技人员的参考书。

图书在版编目(CIP)数据

计算机系统结构/李学干编著. —3 版. —西安:西安电子科技大学出版社,2000.6

国家重点图书·高等学校电子信息类规划教材

ISBN 7-5606-0139-1

I. 计… II. 李… III. 电子计算机-系统结构-高等学校-教材 IV. TP303

中国版本图书馆 CIP 数据核字(1999)第 73418 号

责任编辑 仝德源

出版发行 西安电子科技大学出版社(西安市太白南路 2 号)

电 话 (029) 8227828 邮 编 710071

http://www.xduph.com E mail: xdupfxb@pub.xaonline.com

经 销 新华书店

印 刷 陕西画报印刷厂

版 次 1984 年 11 月第 1 版 1991 年 6 月第 2 版

2000 年 6 月第 3 版 2002 年 1 月第 21 次印刷

开 本 787 毫米×1092 毫米 1/16 印张 21 75

字 数 520 千字

印 数 226 001~238 000 册

定 价 22.00 元

ISBN 7-5606-0139-1/TP·0046

XDUP 0578073-21

*** 如有印装问题可调换 ***

本书封面贴有西安电子科技大学出版社的激光防伪标志,无标志者不得销售。

出版说明

为做好全国电子信息类专业“九五”教材的规划和出版工作，根据国家教委《关于“九五”期间普通高等教育教材建设与改革的意见》和《普通高等教育“九五”国家级重点教材立项、管理办法》，我们组织各有关高等学校、中等专业学校、出版社，各专业教学指导委员会，在总结前四轮规划教材编审、出版工作的基础上，根据当代电子信息科学技术的发展和面向 21 世纪教学内容和课程体系改革的要求，编制了《1996—2000 年全国电子信息类专业教材编审出版规划》。

本轮规划教材是由个人申报，经各学校、出版社推荐，由各专业教学指导委员会评选，并由我们与各专指委、出版社协商后审核确定的。本轮规划教材的编制，注意了将教学改革力度较大、有创新精神、有特色风格的教材和质量较高、教学适用性较好、需要修订的教材以及教学急需、尚无正式教材的选题优先列于规划。在重点规划本科、专科和中专教材的同时，选择了一批对学科发展具有重要意义，反映学科前沿的选修课、研究生课教材列入规划，以适应高层次专门人才培养的需要。

限于我们的水平和经验，这批教材的编审、出版工作还可能存在不少缺点和不足，希望使用教材的学校、教师、学生和其他广大读者积极提出批评和建议，以不断提高教材的编写、出版质量，共同为电子信息类专业教材建设服务。

电子工业部教材办公室

前 言

本教材系按原电子工业部的《1996~2000年全国电子信息类专业教材编审出版规划》，由全国高等院校计算机专业教学指导委员会编审并推荐出版。

本教材由西安交通大学郑守其教授主审，钱德沛教授为责任编委。

本教材的前身作为计算机系统结构的全国统编教材是1981年由国防工业出版社出版的。它是国内最早较完整、系统讲述计算机系统结构的教材。1984年经重新编写后，改由西北电讯工程学院出版社(西安电子科技大学出版社前身)出版。1991年再由作者重新修改作为第二版发行。《计算机系统结构(第二版)》先后曾获全国第三届工科电子类专业优秀教材一等奖和国家级优秀教学成果二等奖。这次在修编时，对各章内容又做了较大修改、增删，但作为教材的继承性，不宜在体系、内容和风格上作太大变动。

本教材仍着眼于有关计算机系统结构和组成的基本概念、基本原理、基本结构和基本分析方法的叙述。全书共分8章。第1章讲述计算机系统的层次结构，计算机系统结构、组成、实现的定义和相互关系，软硬取舍原则与系统设计思路，软件移植手段，应用、器件对系统结构的影响，并行性发展与计算机系统分类。第2章讲述数据表示，寻址方式，指令系统的设计与改进，RISC技术。第3章讲述总线设计，中断系统，通道处理机和外围处理机。第4章讲述存贮体系，虚拟存贮器，Cache存贮器和主存保护。第5章讲述重叠，流水，向量处理机，超标量、超长指令字、超流水线处理机。第6章讲述并行处理机和相联处理机。第7章讲述多处理机的硬件结构、程序并行性、性能分析及操作系统。第8章讲述脉动阵列机，数据流机，归约机，智能机，大规模并行处理机(MPP)和机群系统。本教材力求反映出近十几年来在系统结构上的重要进展和今后可能的发展。

本课程应在“数字逻辑”、“计算机(组成)原理”、“程序设计语言”等课程之后开设。学生最好有“数据结构”方面的知识。本课程可在“操作系统”、“编译原理”课程之后，或与它们同时开设。本教材也可作为其它相关专业的研究生或本科生的选修教材。本课程的参考教学时间为60~80学时，可根据情况作适当剪裁。

本书在编写过程中多次得到有关领导部门及不少兄弟院校、研究所的专家、教授和同行的热情鼓励和帮助。西安电子科技大学教材科及出版社为本书的出版也做了大量工作，在此表示衷心的感谢。

由于作者水平有限，书中难免存在错误和缺点，恳切希望广大读者批评指正。

李学干
1999年6月

目 录

第 1 章 计算机系统结构的基本概念

1.1 计算机系统的多级层次结构	1
1.2 计算机系统结构、组成与实现	3
1.2.1 结构、组成与实现	3
1.2.2 计算机系统结构、组成和实现三者的相互影响	7
1.3 软硬取舍与计算机系统的设计思路	9
1.3.1 软硬取舍的基本原则	9
1.3.2 计算机系统的设计思路	11
1.4 软件、应用、器件对系统结构的影响	12
1.4.1 系统结构设计中必须解决好软件的可移植性	13
1. 统一高级语言	13
2. 采用系列机思想	14
3. 模拟与仿真	16
1.4.2 应用对系统结构的影响	18
1.4.3 器件发展对系统结构的影响	22
1.5 系统结构中的并行性发展及计算机系统的分类	24
1.5.1 并行性概念	24
1. 并行性的含义与并行性级别	24
2. 并行性开发的途径	26
1.5.2 计算机系统的并行性发展	27
1. 1960 年以前	27
2. 1960 年至 1970 年	28
3. 1970 年至 1980 年	28
4. 1980 年至 1990 年	30
5. 1990 年以来	31
1.5.3 并行处理系统的结构与多机系统的耦合度	32
1. 并行处理计算机的结构	32
2. 多机系统的耦合度	33
1.5.4 计算机系统的分类	34
习题	37

第 2 章 数据表示与指令系统

2.1 数据表示	40
2.1.1 数据表示与数据结构	40
2.1.2 高级数据表示	42

1. 自定义数据表示	42
2. 向量数组数据表示	46
3. 堆栈数据表示	47
2.1.3 引入数据表示的原则	48
2.1.4 浮点数尾数基值大小和下溢处理方法的选择	50
1. 浮点数尾数基值的选择	50
2. 浮点数尾数的下溢处理方法	55
2.2 寻址方式	57
2.2.1 寻址方式分析	57
2.2.2 逻辑地址与主存物理地址	59
2.3 指令系统的设计和改进	62
2.3.1 指令格式的优化	62
1. 操作码的优化表示	62
2. 指令字格式的优化	65
2.3.2 按增强指令功能的方向发展与改进指令系统	68
1. 面向目标程序的优化实现来改进	69
2. 面向高级语言的优化实现来改进	72
3. 面向操作系统的优化实现来改进	76
2.3.3 按简化指令功能的方向发展与改进指令系统	80
1. 精简指令系统思想的提出	80
2. RISC 结构采用的基本技术	82
3. RISC 技术的发展	86
习题	91

第 3 章 总线、中断与输入输出系统

3.1 输入输出系统概述	94
3.2 总线设计	96
3.2.1 总线的类型	96
3.2.2 总线的控制方式	97
3.2.3 总线的通信技术	99
1. 同步通信	99
2. 异步通信	99
3.2.4 数据宽度与总线线数	101
1. 数据宽度	101
2. 总线的线数	102
3.3 中断系统	103
3.3.1 中断的分类和分级	103
3.3.2 中断系统的软硬件功能分配	108
3.4 通道处理机	109
3.4.1 工作原理	109
3.4.2 通道流量的分析	112
3.5 外围处理机	114
习题	117

第4章 存贮体系

4.1 存贮体系的形成与性能	119
4.1.1 发展存贮体系的必要性	119
4.1.2 并行主存系统频宽的分析	120
4.1.3 存贮体系的形成与分支	124
4.1.4 存贮体系的性能参数	126
4.2 虚拟存贮器	127
4.2.1 不同的虚拟存贮管理方式	127
1. 段式管理	128
2. 页式管理	131
3. 段页式管理	132
4.2.2 页式虚拟存贮器构成	134
1. 地址的映象和变换	134
2. 替换算法	137
3. 虚拟存贮器工作的全过程	143
4.2.3 页式虚拟存贮器实现中的问题	144
1. 页面失效的处理	144
2. 提高虚拟存贮器等效访问速度的措施	145
3. 影响主存命中率和 CPU 效率的某些因素	150
4.3 高速缓冲存贮器(Cache)	151
4.3.1 基本结构	152
4.3.2 地址的映象与变换	154
1. 全相联映象和变换	154
2. 直接映象及其变换	155
3. 组相联映象及其变换	156
4. 段相联映象	160
4.3.3 替换算法的实现	160
1. 堆栈法	161
2. 比较对法	162
4.3.4 Cache 的透明性及性能分析	164
1. Cache 的透明性分析	164
2. Cache 的取算法	166
3. 任务切换对失效率的影响	167
4. 影响 Cache 存贮器性能的因素	168
4.3.5 “Cache—主存—辅存”存贮层次	170
4.4 主存保护	171
习题	174

第5章 重叠、流水和向量处理机

5.1 重叠解释方式	178
5.1.1 基本思想和一次重叠	178
5.1.2 相关处理	181

1. 指令相关的处理	181
2. 主存空间数相关的处理	182
3. 通用寄存器组相关的处理	183
5.2 流水方式	187
5.2.1 基本概念	187
1. 流水是重叠的引申	187
2. 流水线的分类	188
5.2.2 流水线处理机的主要性能	192
1. 吞吐率	192
2. 效率	195
3. 流水线工作举例	196
5.2.3 流水机器的相关处理和控制机构	197
1. 局部性相关的处理	198
2. 全局性相关的处理	202
3. 流水器的中断处理	204
4. 流水线调度	204
5.3 向量的流水处理与向量流水处理机	208
5.3.1 向量的流水处理	208
5.3.2 向量流水处理机	210
1. 向量处理机的指令系统	210
2. 向量流水处理机的结构	210
3. 超级向量流水处理机举例	216
5.4 指令级高度并行的超级处理机	217
5.4.1 超标量处理机	217
5.4.2 超长指令字处理机	219
5.4.3 超流水线处理机	220
习题	221

第 6 章 并行处理机和相联处理机

6.1 并行处理机原理	226
6.1.1 并行处理机的构形与特点	226
1. 并行处理机的基本构形	226
2. 并行处理机的特点	228
6.1.2 并行处理机的算法	228
1. ILLIAC IV 的处理单元阵列结构	229
2. 阵列处理机的算法举例	229
6.1.3 SIMD 计算机的互连网络	234
1. 互连网络的设计目标及互连函数	234
2. 基本的单级互连网络	235
3. 多级互连网络	238
4. 全排列网络	243
6.1.4 并行存贮器的无冲突访问	243
6.2 并行处理机举例	246

6.2.1	ILLIAC N阵列处理机	246
6.2.2	BSP 科学处理机	248
6.2.3	MPP 位平面阵列处理机	250
6.2.4	CM 连接机	252
6.3	相联处理机	253
6.3.1	相联处理机和相联存贮器的组成	253
1.	相联处理机的特点和组成	253
2.	相联存贮器的组成及相联处理机的结构类型	254
6.3.2	相联检索算法	256
6.3.3	相联处理机结构举例	258
1.	PEPE 系统	258
2.	STARAN 系统	260
	习题	262

第 7 章 多 处 理 机

7.1	多处理机的特点及主要技术问题	265
7.2	多处理机的硬件结构	267
7.2.1	紧耦合和松耦合	267
1.	紧耦合多处理机	267
2.	松耦合多处理机	269
7.2.2	机间互连形式	270
1.	总线形式	270
2.	环形互连形式	271
3.	交叉开关形式	272
4.	多端口存贮器形式	273
5.	开关枢纽结构形式	274
7.2.3	存贮器组织	275
1.	并行主存贮器的构成	275
2.	多 Cache 的一致性问题的	276
7.3	程序并行性	278
7.3.1	并行算法	279
1.	算术表达式的并行运算	279
2.	递归程序的并行性	281
7.3.2	程序并行性的分析	283
7.3.3	并行程序设计语言	285
7.4	多处理机的性能	289
7.4.1	任务粒度与系统性能	289
7.4.2	性能模型与分析	290
1.	$N=2$ 且计算与通信不能重叠	290
2.	$N>2$ 且计算与通信不能重叠	290
3.	额外开销与计算工作可以重叠	293
4.	机间通信可以多路同时进行	294
7.5	多处理机的操作系统	295

7.5.1 多处理机操作系统的难度和特点	295
1. 多处理机操作系统的难度	295
2. 多处理机操作系统的特点	296
7.5.2 多处理机操作系统的类型	297
1. 主从型	297
2. 各自独立型	297
3. 浮动型	298
7.5.3 多处理机操作系统的发展	298
习题	299

第 8 章 具有现代特色的计算机

8.1 脉动阵列机	302
8.1.1 脉动阵列结构的原理	302
8.1.2 面向特定算法的脉动阵列结构	304
8.1.3 通用的脉动阵列结构	307
8.2 数据流机	309
8.2.1 数据驱动的概念	309
8.2.2 数据流程序图和语言	311
1. 数据流程序图	311
2. 数据流语言	315
8.2.3 数据流计算机的结构	316
1. 静态数据流机	316
2. 动态数据流机	318
8.2.4 数据流机器存在的问题	321
8.3 归约机	322
8.3.1 函数式程序设计语言	322
8.3.2 面向函数程序设计的归约机	324
1. 归约机的基本结构特点	324
2. 串归约机	326
3. 图归约机	327
8.4 智能机	328
8.4.1 智能信息处理与智能机	328
8.4.2 智能机的结构及所用的机器语言	330
1. 智能机的结构	330
2. 逻辑程序设计语言	331
3. 智能计算机的进展	332
8.5 大规模并行处理机 MPP 与机群系统	333
习题	336
参考文献	338

第 1 章 计算机系统结构的基本概念

本章先从多级层次结构观点定义什么是计算机系统结构，并说明结构、组成、实现三者的含义和关系。然后，讲述计算机系统软、硬件功能分配的基本原则和设计思路，软件、应用、器件对系统结构的影响。最后介绍计算机系统中并行性的概念和发展，以及计算机系统的分类。

1.1 计算机系统的多级层次结构

现代通用的计算机系统是由紧密相关的硬件和软件组成的。从使用语言的角度，可以将系统看成是按功能划分的多个机器级组成的层次结构，如图 1.1 所示。层次结构由高到低分别为应用语言机器级、高级语言机器级、汇编语言机器级、操作系统机器级、传统机器语言机器级和微程序机器级。对于一个具体的计算机系统，层次的多少会有所不同。

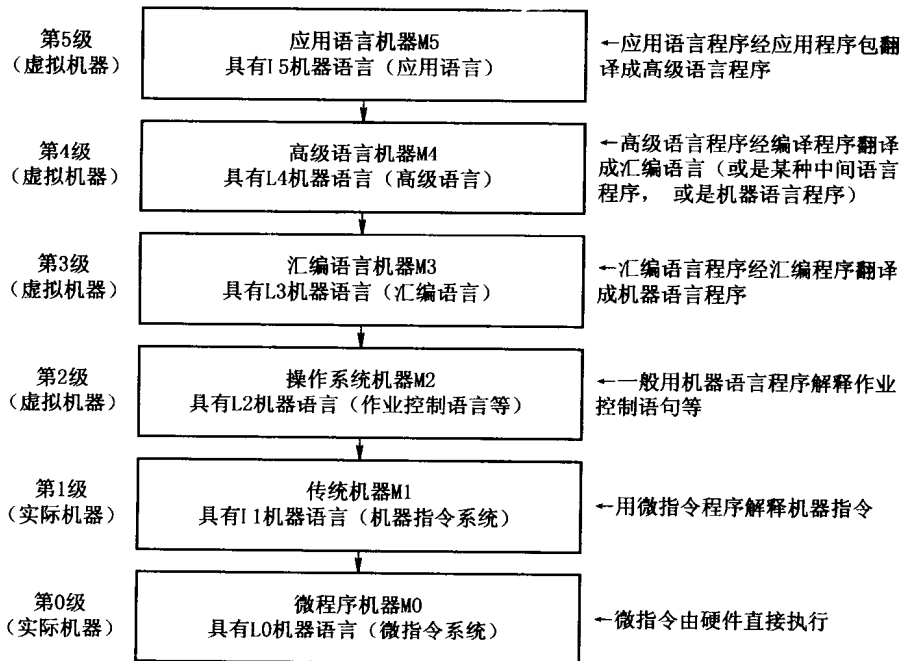


图 1.1 计算机系统的多级层次结构

对使用某一级语言编程的程序员来讲，只要熟悉和遵守使用该级语言的规定，所编程序总能在该机器级上运行，得到结果，而不用管这个机器级是如何实现的。就好像该程序员有了一台可以直接使用这种语言作为机器语言的机器一样。这里，“机器”被定义为能存储和执行相应语言程序的算法和数据结构的集合体。实际上，只有二进制机器指令即传统所讲的机器语言与机器硬件直接对应，方可直接被硬件识别和执行。

各机器级的实现主要靠翻译或解释，或者是这两者的结合。翻译(Translation)是先用转换程序将高一级的机器级上的程序整个地变换成低一级机器级上可运行的等效程序，然后再在低一级机器级上去实现的技术。解释(Interpretation)则是在低一级机器级上用它的一串语句或指令来仿真高一级的机器级上的一条语句或指令的功能，通过对高一级的机器语言程序中的每条语句或指令逐条解释来实现的技术。

应用语言虚拟机器级 M5 是为了满足管理、人工智能、图像处理、辅助设计等专门的应用来设计的。使用面向某种应用环境的应用语言(L5)编写的程序一般是经应用程序包翻译成高级语言(L4)程序后，再逐级向下实现的。高级语言机器级 M4 上的程序可以先用编译程序整个地翻译成汇编语言(L3)程序或机器语言(L1)程序，再逐级或越级向下实现，也可以用汇编语言(L3)程序、机器语言(L1)程序，甚至微指令语言(L0)程序解释实现。对汇编语言(L3)源程序则先用汇编程序整个将其变换成等效的二进制机器语言(L1)目标程序，再在传统机器级 M1 上实现。操作系统程序虽然已发展成用高级语言(如面向编写操作系统软件的 C 语言)编写，但最终还是要用机器语言程序或微指令程序来解释的。它提供了传统机器级 M1 所没有，但为汇编语言和高级语言使用和实现所用的基本操作、命令和数据结构，例如，文件管理、存储管理、进程管理、多道程序共行、多重处理、作业控制等所用到的操作命令、语句和数据结构等。因此，操作系统机器级 M2 放在传统机器级 M1 和汇编语言机器级 M3 之间是适宜的。传统机器级采用组合逻辑电路控制，其指令可直接用硬件来实现，也可以采用微程序控制，用微指令(L0)程序来解释实现。微指令直接控制硬件电路的动作。

就目前的状况来看，M0 用硬件实现，M1 用微程序(固件)实现，M2 到 M5 大多用软件实现。我们称以软件为主实现的机器为虚拟机器，以区别于由硬件或固件实现的实际机器。虚拟机器不一定全都由软件实现，有些操作可以用固件或硬件实现。如操作系统中的某些命令可由比它低两级的微程序解释，或全部用硬件来实现。高级语言机器则直接用微程序解释或用硬件实现，没有编译软件。

采用何种实现方式，要从整个计算机系统的效率、速度、造价、资源状况等方面全面考虑，对软件、硬件、固件取舍进行综合平衡。软件和硬件在逻辑功能上是等效的。原理上，软件实现的功能完全可以用硬件或固件完成，硬件实现的功能也可以由软件的模拟来完成，只是其性能、价格、实现的难易程度等有所不同。具有相同功能的计算机系统，其软、硬件功能分配比例可以在很宽的范围内变化，如图 1.2 所示。这种分配比例随不同时期及同一时期的不同机器

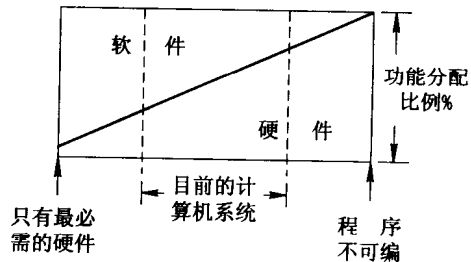


图 1.2 计算机系统的软、硬件功能分配

在动态地改变。由于软、硬件紧密相关，软硬界面常常是模糊不清的。例如，很难分清中断处理、存贮管理等功能中哪些是硬件完成的，哪些是软件完成的。软、硬分配的比例主要看满足应用的前提下，能否充分利用硬件、器件技术的现状和进展，使系统达到较高的性能价格比。此外，对计算机系统使用者来说，还要考虑他所直接接触到的应用语言机器级或高级语言机器级的状况。

从概念和功能上把一个复杂的计算机系统看成是由多个机器级构成的层次结构，有利于理解软件、硬件、固件的地位和作用。可以看出，微程序机器级、传统机器级、操作系统机器级主要是为运行支持更高层机器级程序所需的解释程序和翻译程序设计的，以便能设计和实现新的虚拟机器级，并不是直接为应用程序员解题而设计的。

把计算机系统看成是由多个机器级构成的层次结构推动了计算机系统结构的发展。例如，可以为应用语言级、高级语言级、操作系统级提供更多更好的硬件支持，重新合理地分配好软、硬件的比例，改变硬件、器件迅速发展而软件却日益复杂、开销过大的状况；或者直接就用硬件或固件实现，发展所谓高级语言机器或操作系统计算机结构。既然层次结构中的每一级都有其自己的用户、实现方法和指令系统，因此就可以设想让各级虚拟机器用真正存在的处理机来代替，摆脱以往各级功能都在同一台实际机器上实现的状况，发展出多处理机、分布处理、计算机网等系统结构。可以在一台真正的宿主机上通过模拟或仿真另一台不同的机器，来推动自虚拟机、多种操作系统共同执行等技术的采用，促进软件移植、计算机系统性能评价、计算机设计自动化等的发展。

1.2 计算机系统结构、组成与实现

1.2.1 结构、组成与实现

从计算机系统的层次结构上定义，系统结构(System Architecture)是对计算机系统中各机器级之间界面的划分和定义，以及对各级界面上、下的功能进行分配。各级都有它自己的系统结构。IBM PC 和 VAX-11 是两种不同系列的计算机。它们的汇编指令和二进制机器指令的操作类型、指令格式、寻址方式、寄存器组织、I/O 设备连接方式等都不一样。就是说，从传统机器语言程序员或汇编语言程序员看，这两种机器的概念性结构和功能特性差别很大，为使他们所编的程序能在机器上正确运行而应当了解的计算机属性是很不相同的。然而，高级语言程序员对这些差别却看不到。本来存在的事物或属性，从某个角度上看不到，称对他是透明的(Transparent)。不同机器级的程序员所看到的计算机属性是不同的，这个属性就是计算机系统不同层次的界面。系统结构设计就是要研究对某级，哪些应当透明，哪些不应当透明。透明的好处是不用管了，可以简化该级的设计。但由于看不到，就会无法控制，带来不利。因此，正确合理地进行透明性分析和取舍是非常重要的。

我们这里所称的计算机系统结构或计算机体系结构(Computer Architecture)只是系统结构中的一部分。它指的是层次结构中传统机器级的系统结构，其界面之上的功能包括操作系统级、汇编语言级、高级语言级和应用语言级中所有软件的功能。界面之下的功能包括所有硬件和固件的功能。因此，这个界面实际上是软件与硬件/固件的交界面，是机器语

言、汇编语言程序设计者，或编译程序设计者看到的机器物理系统的抽象。所以，计算机系统结构研究的是软、硬件之间的功能分配以及对传统机器级界面的确定，提供机器语言、汇编语言程序设计者或编译程序生成系统为使其所设计或生成的程序能在机器上正确运行，应看到和遵循的计算机属性。

就目前的通用机来说，计算机系统结构的属性应包括：

硬件能直接识别和处理的数据类型和格式等的数据表示；

最小可寻址单位、寻址种类、地址计算等的寻址方式；

通用/专用寄存器的设置、数量、字长、使用约定等的寄存器组织；

二进制或汇编级指令的操作类型、格式、排序方式、控制机构等的指令系统；

内存的最小编址单位、编址方式、容量、最大可编址空间等的存贮系统组织；

中断的分类与分级、中断处理程序功能及入口地址等的中断机构；

系统机器级的管态和用户态的定义和切换；

输入输出设备的连接、使用方式、流量、操作结束、出错指示等的机器级 I/O 结构；

系统各部分的信息保护方式和保护机构；

等等。

可以看出，机器级内部的数据流和控制流的组成、逻辑设计和器件设计等都不包含在计算机系统结构中，也就是说，它们都对系统结构设计是透明的。

计算机组成(Computer Organization)指的是计算机系统结构的逻辑实现，包括机器级内的数据流和控制流的组成以及逻辑设计等。它着眼于机器级内各事件的排序方式与控制机构、各部件的功能及各部件间的联系。计算机组成设计要解决的问题是在所希望达到的性能和价格下，怎样最佳、最合理地把各种设备和部件组织成计算机，以实现所确定的系统结构。近 40 年里，计算机组成设计主要是围绕提高速度，着重从提高操作的并行度、重叠度，以及分散功能和设置专用功能部件来进行的。

计算机组成设计要确定的方面一般应包括：

数据通路宽度(在数据总线上一次并行传送的信息位数多少)；

专用部件的设置(设置哪些专用部件，如乘除法专用部件、浮点运算部件、字符处理部件、地址运算部件等，每种专用部件设置的数量，这些都与机器所需达到的速度、专用部件的使用频度高低及允许的价格等有关)；

各种操作对部件的共享程度(共享程度高，即使操作在逻辑上不相关也只能分时使用，限制了速度，但价格便宜。可以设置多个部件降低共享程度，提高操作并行度来提高速度，但价格也将提高)；

功能部件的并行度(功能部件的控制和处理方式是采用顺序串行，还是采用重叠、流水或分布处理)；

控制机构的组成方式(事件、操作的排序机构是采用硬联控制还是用微程序控制，是采用单机处理还是用多机处理或功能分布处理)；

缓冲和排队技术(在不同部件之间怎样设置及设置多大容量的缓冲器来弥补它们的速度差异；是采用随机方式，还是先进先出、先进后出、优先级或循环方式来安排等待处理事件的先后顺序)；

预估、预判技术(为优化性能和优化处理，采用什么原则来预测未来的行为)；

可靠性技术(采用什么样的冗余技术和容错技术来提高可靠性);

.....

计算机实现(Computer Implementation)指的是计算机组成的物理实现,包括处理机、主存等部件的物理结构,器件的集成度和速度,器件、模块、插件、底板的划分与连接,专用器件的设计,微组装技术,信号传输,电源、冷却及整机装配技术等。它着眼于器件技术和微组装技术。其中,器件技术在实现技术中起着主导作用。

下面我们通过举例来说明什么是计算机系统结构、计算机组成和计算机实现。

指令系统的确定属计算机系统结构。指令的实现,如取指令、指令操作码译码、计算操作数地址、取操作数、运算、送结果等的操作如何安排和排序属计算机组成。实现这些指令功能的具体电路、器件的设计及装配技术属计算机实现。

确定指令系统中是否要设乘法指令属计算机系统结构。乘法指令是用专门的高速乘法器实现,还是靠用加法器和移位器经一连串时序信号控制其相加和右移来实现,属计算机组成。乘法器、加法—移位器的物理实现,如器件的类型、集成度、数量、价格及微组装技术的确定和选择属计算机实现。

对主存系统来说,主存容量与编址方式(按位、按字节还是按字访问等)的确定属计算机系统结构。为达到所定的性能价格比,主存速度应多快,逻辑结构是否采用多体交叉属计算机组成。主存系统的物理实现,如存储器器件的选定、逻辑电路的设计、微组装技术使用,均属计算机实现。

又如,IBM 370 系列有 115、125、135、145、158、168 等由低档到高档的多种型号机器。汇编语言、机器语言程序设计者看到的概念性结构都如图 1.3 所示,均由中央处理机/主存—通道—设备控制器—外围设备 4 级构成,以主存为中心,采用通道方式输入输出。

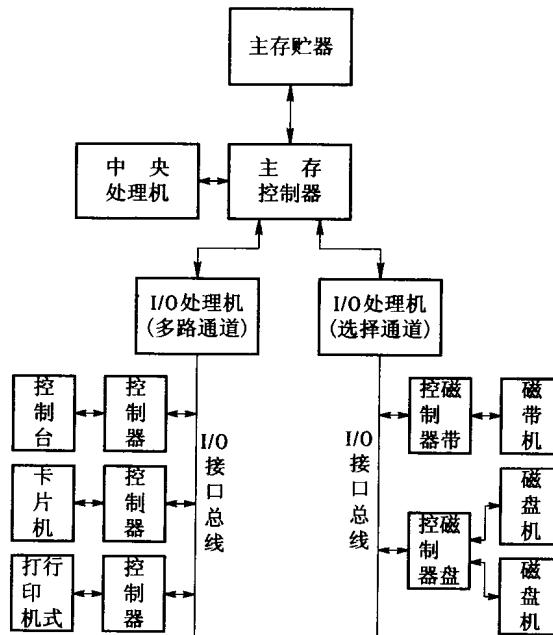
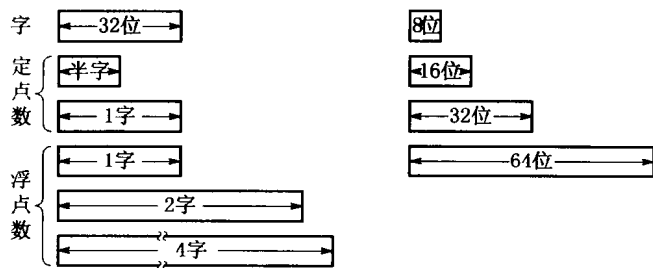


图 1.3 IBM 370 系列机的概念性结构

从层次结构看，IBM 370 系列中不同型号的机器从高级语言机器级、汇编语言机器级到传统机器语言机器级都是相同的。只是由于性能价格的要求不同，它们分别采用各自不同的组成技术和实现技术，具有不同的微程序机器级。因此，高级语言编译软件和应用软件都可以通用于不同型号的机器，只是运行的速度不同，机器的价格不同而已。换句话说，设计什么样的系列机属于计算机系统结构，系列内不同型号计算机的组织则属于计算机组成。

IBM 370 系列机的中央处理机都具有相同的机器指令和汇编指令系统，只是指令的分析、执行，在低档机上采用顺序处理方式，在高档机上则采用重叠、流水或其他并行处理方式。程序设计者编程序时所看到的数据形式(即数据表示)都是相同的 32 位字长，定点数都是半字长 16 位或全字长 32 位，浮点数都是单字长 32 位、双字长 64 位或四字长 128 位，如图 1.4(a)所示。由于速度、价格的要求不同，在组成和实现时，数据通路宽度(数据总线线数)可以分别采用 8 位、16 位、32 位或 64 位，如图 1.4(b)所示。一个 64 位的字，在 8 位数据通路宽度的机器上需分 8 次传送完，而在 64 位数据通路宽度的机器上一次即可传送完。这样，速度快了，但硬件多了，价格高了。因此，数据总线宽度对程序员是透明的，是他不需要知道的。



(a) 统一的字长与定、浮点数表示 (b) 不同的数据通路宽度

图 1.4 IBM 370 系列机字长、数的表示和数据通路宽度

IBM 370 系列机的各档机器都采用通道方式进行输入输出，但在计算机组成上，低档机器可以采用结合型通道，让通道的功能借用中央处理机的某些部件来完成。同一套硬件分时执行中央处理机和通道的功能，虽然系统速度性能低，但可以降低成本。而高档机器上却采用独立型通道，让通道单独设置硬件，与中央处理机并行工作，成本虽高，但系统的速度性能也提高了。

由此可见，机器/汇编指令系统、数据表示、是否采用通道方式输入/输出等的确定都属于系统结构的问题；而指令的分析执行是采用顺序、重叠、流水还是其他并行处理方式，数据通路宽度的确定，通道是采用结合型还是独立型等，都属于计算机组成的问题。

再如，DEC 公司早先推出的 PDP-11 系列以单总线结构著称，实际上它不属于计算机系统结构。为了满足不同的速度和价格要求，不同型的机器采用了多种不同的总线形式。后来推出的 VAX-11 系列(与 PDP-11 兼容)，先后投产了 725、730、750、780、MICRO-VAX、782、785、6330、8550 等多种型号，也采用多种不同的总线。但是它们都具有相同的 I/O 连接和使用方式，即在逻辑上将 I/O 设备端口寄存器看作是主存的一部分，与主存统一编址。通过访问主存的这些指定单元来与相应的 I/O 设备进行通信，完成

对该设备的读/写等。可见，对 PDP—11 或 VAX—11 来说，单总线结构属于计算机组成，其机器级的 I/O 连接和使用方式才属于系统结构，是程序设计者编写 I/O 程序时应当看到的。具体采用哪种总线取决于速度和价格的要求。如 VAX—11 系列中，730 只用 1 个单总线适配器与辅存及外围设备连接；750 通过一个单总线适配器和多达 3 个多总线适配器，或者再加一个单总线适配器和两个多总线适配器与辅存及外围设备进行连接；而 780 以及紧耦合非对称多机系统的 782，则通过多达 4 个单总线适配器和 4 个多总线适配器进行这种连接。从 1986 年开始又逐渐形成 VAXB2 分布式概念的智能总线。

1.2.2 计算机系统结构、组成和实现三者的相互影响

计算机系统结构、计算机组成和计算机实现是 3 个互不相同的概念，但它们是相互联系而又相互影响的。

从上一小节所举的例子可以看出，具有相同系统结构(如指令系统相同)的计算机可以因为速度要求等因素的不同而采用不同的组成。例如，指令的取出、译码、取数、运算、存结果可以顺序进行，也可以让它们的时间上重叠进行以提高速度。又如乘法指令可以利用专用乘法器来实现，也可以通过加法器重复相加、右移来实现，这取决于要求的速度、程序中乘法指令出现的频度及所采用的乘法运算方法。对速度要求高和乘法出现频度高的可用专用乘法器，反之宜采用后一种方法，使速度下降不明显，但却可以显著降低价格。

同样，一种计算机组成可以采用多种不同的计算机实现。例如，主存器件既可以选用双极型的，也可以选用 MOS 型的；可以用单片大规模集成电路，也可以用多片小规模集成电路进行组搭。显然，这取决于要求达到的性能价格比和器件技术的现状。

采用不同的系统结构会使可以采用的组成技术产生差异。例如，为实现

$A := B + C$

$D := E * F$

如果采用面向寄存器的系统结构，其程序可以是

```
LOAD    R1, B
ADD     R1, C
STORE  R1, A
LOAD    R2, E
MPY    R2, F
STORE  R2, D
```

而对面向主存的具有三地址寻址方式的系统结构，其程序可以是

```
ADD     B, C, A
MPY    E, F, D
```

要提高机器的运算速度，可以让相加与相乘并行。为此，这两种系统结构在组成上都要求独立设置加法器和乘法器。然而，对于面向寄存器的系统结构还要求能同时访问 R1 和 R2。对于面向主存的具有三地址寻址方式的系统结构却无此要求，只要求能同时形成多个操作数的地址和并行访存。

反过来，计算机组成也会影响系统结构，微程序控制就是一个典型的例子。微程序控制最初是为了使控制器规整。将存贮技术引入到控制器内，把硬联控制器网络的复杂逻辑