

高等学校试用教材

计算机组织与结构

金 兰

高等教育出版社

高等学校试用教材

计算机组织与结构

金 兰

高等教育出版社

内 容 提 要

本书是根据高等学校计算机软件教材编审委员会制定的本课程教学大纲编写的。取材广泛,内容丰富。全书共分八章,重点讨论了硬件对软件的界面,讲述了指令系统的一般构成、数据传送路径和指令控制流程、微程序设计原理、存储器层次结构、中断和输入输出系统等内容。本书既讲述一般原理又讲解具体实例并附有大量习题。

本书可作为高等学校计算机软件及有关专业的教材,也可供计算机科学工作者阅读参考。

高等学校试用教材
计算机组织与结构

金 兰

*

高等教育出版社出版
新华书店北京发行所发行
北京印刷一厂印装

*

开本787×1092 1/16 印张 24.5 字数560,000

1986年5月第1版 1986年5月第1次印刷

印数 00,001—10,300

书号 13010·01173 定价3.75元

前 言

本书是针对高等学校“计算机结构原理”课程编写的一本通用教材。虽然它主要是根据高等学校计算机软件教材编审委员会于1983年6月审定的教学大纲编写且由该编委会推荐出版的,但是其内容的广泛性和可选择性决定了它不仅适用于计算机软件专业,而且适用于以软、硬结合为指导原则的其它计算机科学技术类专业。在当前我国倡导和推广应用计算机的形势下,它也可作为一本有助于广大科学技术人员学习和了解计算机基本工作原理的参考书。

“计算机结构原理”又名“计算机组成原理”或“计算机原理”,这是一门在我国开设已有近30年历史的主要必修课程。早期的课程内容充满了关于计算机标准线路的知识,反映了当时以分立元件为实现手段的技术状况。随着计算机科学技术的飞速发展,计算机早已超出了单纯由元件和设备所代表的硬件概念,成为一个由硬件和软件等庞大“资源”所组成的综合系统。介绍这样一个复杂系统的工作原理也就变得愈来愈不容易。除了需要许多课程配合以外,还需要采取专门的观点和方法。

计算机系统也象所有大型复杂系统一样,具有多方面的属性。这些属性分布在多个不同的层次,有些是表现在外部的,反映使用者对系统的要求,另一些则隐含在内部,是设计者用来实现系统的依据。外层是对内层的抽象,建立起各层虚拟机器,以便更好地适应广大用户的需求;内层是外层的实施,最终实现物理机器,以现时最先进的技术手段提供对系统的硬件支持。这种对计算机系统采取的层次结构观点,能有效地帮助开展计算机系统的研究、设计和应用。

把计算机系统分为多层次结构的结果,使观察者站在不同的层次,他会看到系统具有不同的属性。外层的使用者往往看不到内层系统的细节,而内层的设计者有时又不容易看到系统的全貌。所以为了更透彻地了解计算机系统,应当提倡全面观察、逐层剖析的方法。把软件和硬件相结合,把计算机结构和计算机组织相结合,正是这种方法的体现,因此本书取名“计算机组织与结构”。

计算机系统的每个层次都有它自己的结构。本书所称的计算机结构是定义在机器语言一级,表现为从汇编语言程序设计者所看到的计算机属性。实际上,它是计算机硬件系统的结构,是计算机硬件对软件的界面。计算机组织处在它的内层,是计算机结构的逻辑实现和物理实现。具体说来,研究和建立各功能部件间的相互连接和相互作用,进行各个功能部件的逻辑设计,是逻辑实现的内容;把逻辑设计深化到元、器件级,则是物理实现的内容。围绕提高系统性能价格比的目标,确定和实现计算机系统在机器指令级的功能和特性,乃是计算机结构和计算机组织的共同学科任务,二者是相互联系和相互渗透的,不可能截然分开。

本书基于上述观点和方法,主要是从寄存器传送级以上的层次来阐述计算机的各主要组成部分及其相互联系和相互作用,以此了解计算机的基本工作原理和各主要部件功能的实现方法。由于具体的逻辑电路内容已经下移到“数字电路”、“数字逻辑”等课程(本书只用第二章篇幅提供

浓缩的预备知识),因此本书重点讨论硬件对软件的界面,包括指令系统的一般构成(第四章)、数据传送路径和指令控制流程(第六章)、微程序设计原理(第七章)、存储器层次结构(第五章)、中断和输入输出系统(第八章)等。

本书按横向方式组织课程内容,即不采用从头到尾过细讲解一种小型机的传统做法,而是通过几种典型机器实例来阐明计算机结构原理的各个侧面。选择机型的原则有三:一是尽可能选用实际的机器例子,但允许为了突出原理而简化若干技术细节,甚至也可为了引入某些重要概念(如微程序)而利用假想机作为局部例证。二是主要选用在我国有较大影响而又在结构上有典型意义的机型,例如 IBM 370 系列和 PDP-11 系列就满足这样的条件,也为国外教材所普遍接受。DJS-130 机虽已较少使用,但为我国计算机工作者所熟悉,仍有助于说明计算机原理,所以也适当引用。三是涉及机型不宜过窄,要照顾阐述计算机结构原理各个方面内容的需要。堆栈计算机就属于这一种情况,为此选用了 HP-300 计算机作为实例。

为了克服横向叙述可能带来对某一种机型内容不够连贯的缺点,本书最后在附录中增加了“几种通用计算机纵览表”。目的是把本书各章分散介绍的实例内容按三种机型纵向组织起来,以起到总结和索引的作用。

本书如用作教材,其篇幅与计划学时数70相比可能是偏大一些。希望根据各校具体情况有选择地讲授重点章节,而舍去某些非基本要求或与其它课程重复的内容。另外,还应提倡学生在教师“少而精”讲授的基础上自学钻研教材,并充分发挥习题和实验对加深理解课堂知识的作用。

在本书编写和初稿试用的过程中,清华大学计算机工程与科学系的本课程教学集体给予了很大的帮助,不但积累了一遍又一遍试用的经验,而且参加了部分章节(初稿)和习题的编写工作。他们当中应由作者特别感谢的有谢树煜、汤志忠、张弘志、刘凤云、廖先湜、史翠英等同志。许多兄弟院校参加了本书编写大纲的讨论,不少院校如人民大学、南京工学院、北大二分校等校的有关课程老师试用过本教材油印稿,他们提出了宝贵的意见,有的还帮助审阅了部分书稿,作者在此一并致谢。美国麻省理工学院(M.I.T.)李凡(Francis F. Lee)教授对本书的编写也给予了指导和支持,作者借此机会表示衷心的感谢。

本书承蒙西安交通大学胡正家先生、北京大学许卓群先生审阅在此表示衷心感谢。

本书的编写起步较早,虽几经修改,仍适应不了计算机技术更新的步伐。现在的稿子还是不够满意的,希望读者阅后多加批评指正。

作 者

1984年8月

目 录

第一章 绪论	1	1.2.4.1 微电子学的成就	10
1.1 计算机的特性和组成	1	1.2.4.2 新一代计算机的发展形势	11
1.1.1 计算机的主要特性	1	1.2.4.3 历史的借鉴	11
1.1.1.1 计算机的定义	1	1.3 计算机系统的功能和结构	12
1.1.1.2 计算机的四大特性	1	1.3.1 计算机系统的层次结构	12
1.1.1.3 计算机的外部特性与内部特性 的关系	2	1.3.1.1 硬件与软件相结合的重要形式—— 微程序	12
1.1.2 计算机的硬件组成	2	1.3.1.2 计算机系统的功能模型	13
1.1.2.1 古典冯诺依曼计算机	2	1.3.1.3 计算机系统的功能层次	13
1.1.2.2 计算机组织的学科任务	3	1.3.1.4 计算机硬件结构描述级别	15
1.1.2.3 计算机作业方式	4	1.3.1.5 计算机产品组装级别	15
1.1.2.4 输入输出子系统	4	1.3.2 计算机系统的特性	16
1.1.2.5 软件的配合	4	1.3.2.1 计算机等级	16
1.1.2.6 多层次存储系统	5	1.3.2.2 计算机系列	16
1.1.2.7 计算机的组成	5	1.3.2.3 计算机应用类别	18
1.2 计算机发展简史	6	本章参考文献	19
1.2.1 第一代计算机发展时期 (1945~1958)	6	第二章 计算机逻辑线路	20
1.2.1.1 ENIAC 计算机	6	2.1 引言	20
1.2.1.2 EDVAC 和 EDSAC 存储程序 计算机	7	2.2 数字化信息编码	20
1.2.1.3 计算机工业的出现	7	2.2.1 多重编码概念	20
1.2.1.4 高级程序设计语言的诞生	8	2.2.1.1 重编码的意义	20
1.2.2 第二代计算机发展时期 (1958~1964)	8	2.2.1.2 基 2 码	21
1.2.2.1 晶体管计算机数量上的大发展	8	2.2.1.3 基 2 码的特性	21
1.2.2.2 计算机兼容问题的产生	9	2.2.2 字符编码	22
1.2.3 第三代计算机发展时期 (1964~1975)	9	2.2.2.1 ASCII 码	22
1.2.3.1 IBM 360 系统的创新	9	2.2.2.2 EBCDIC 码和其它字节编码	24
1.2.3.2 对计算机工业的推动	9	2.2.3 十进制数位符号的编码	24
1.2.3.3 向大型/巨型机和小型机的发展	10	2.2.3.1 数制	24
1.2.3.4 计算机向社会化进军	10	2.2.3.2 十进制编码原理	25
1.2.3.5 计算机工业的集中化	10	2.2.3.3 十进制有权码	25
1.2.4 发展新时期及历史的启示	10	2.2.3.4 十进制余 3 码	25
		2.2.3.5 十进制格雷码	26
		2.2.3.6 其它十进制码	27
		2.3 逻辑变量与逻辑函数	27
		2.3.1 逻辑变量与基本逻辑操作	27

2.3.1.1 逻辑变量与逻辑函数	27	2.4.3.5 可编程序逻辑阵列	53
2.3.1.2 布尔代数	28	2.4.4 组合逻辑线路性能分析	54
2.3.1.3 基本逻辑操作	29	2.4.4.1 电平的逻辑赋值	54
2.3.2 逻辑函数的真值表形式	30	2.4.4.2 传送延迟	54
2.3.2.1 真值表的格式	30	2.4.4.3 逻辑线路的空间—时间权衡问 题	55
2.3.2.2 二变量逻辑操作	30	2.4.4.4 组合逻辑线路输出信号的毛刺现 象	55
2.3.3 逻辑函数的布尔表达式	31	2.4.4.5 组合逻辑线路的冒险现象	57
2.3.3.1 逻辑表达式的定义	31	2.5 时序逻辑线路	58
2.3.3.2 逻辑表达式分析树	31	2.5.1 时序逻辑线路的特性和模型	58
2.3.3.3 逻辑表达式的积之和正则形式	32	2.5.1.1 时序逻辑线路的模型	58
2.3.3.4 逻辑表达式的和之积正则形式	32	2.5.1.2 同步和异步时序逻辑线路	58
2.3.3.5 逻辑表达式正则形式的普遍公 式	33	2.5.2 基本的二进位存储元件—— RS 触发器	60
2.3.4 逻辑表达式的化简	34	2.5.2.1 基本功能	60
2.3.4.1 全组合化简规则	34	2.5.2.2 逻辑关系式及其实现	60
2.3.4.2 观察化简法	35	2.5.2.3 RS 触发器状态分析	62
2.3.4.3 卡诺图化简法	35	2.5.3 常用触发器逻辑线路	63
2.3.4.4 带冗余条件的逻辑表达式化简	37	2.5.3.1 钟控 RS 触发器	63
2.4 组合逻辑线路	38	2.5.3.2 从计数看触发器多次翻转现象	63
2.4.1 从逻辑表达式到组合逻辑线 路	38	2.5.3.3 主从 RS 触发器	65
2.4.1.1 组合逻辑线路的模型	38	2.5.3.4 维持和阻塞概念	66
2.4.1.2 “或非”和“与非”逻辑门	38	2.5.3.5 维持阻塞 RS 触发器和 T 触发器	67
2.4.1.3 等价逻辑门概念	39	2.5.3.6 JK 触发器	68
2.4.1.4 等价逻辑门符号替代法	40	2.5.3.7 D 触发器	68
2.4.1.5 电等价门概念	41	2.5.4 计算机常用时序逻辑线路	70
2.4.1.6 电等价门分析法	42	2.5.4.1 各种类型触发器特性小结	70
2.4.2 计算机的典型组合逻辑线路——加 法器	43	2.5.4.2 移位寄存器	70
2.4.2.1 加法器的逻辑表达式	43	2.5.4.3 时钟节拍发生器	72
2.4.2.2 全加器的生成进位和传递进位	44	2.5.4.4 计数器	74
2.4.2.3 行波加法器	45	2.5.4.5 任意计数码顺序的计数器设计	75
2.4.2.4 快速行波加法器	45	本章参考文献	77
2.4.2.5 先行进位加法器	46	习题	77
2.4.2.6 块式先行进位加法器	47	第三章 计算机的算术逻辑运算及其实现	80
2.4.3 计算机常用组合逻辑线路	48	3.1 引言	80
2.4.3.1 译码器	48	3.2 定点数的表示和加减运算	81
2.4.3.2 译码器/信号分离器	49	3.2.1 模运算和补码概念	81
2.4.3.3 多路转接器	49	3.2.1.1 模运算及其性质	81
2.4.3.4 可编程序只读存储器	51		

3.2.1.2 模运算的图形表示和补码概念	82	3.3.4 浮点加减运算方法	101
3.2.1.3 基减1补码或反码	83	3.3.4.1 对阶	101
3.2.2 表示正负数的三种代码制	83	3.3.4.2 结果后处理	101
3.2.2.1 模运算数轴的代码表示和数值解释	83	3.3.4.3 最少警戒位数的分析	102
3.2.2.2 符号-绝对值表示法与补码表示法的区分	83	3.3.5 浮点加减运算的实现	103
3.2.2.3 补码表示法与反码表示法的区分	84	3.3.5.1 用组合逻辑线路实现浮点加减运算	103
3.2.2.4 负数的补码	84	3.3.5.2 流水线运算部件	104
3.2.2.5 小数的模运算情形	85	3.3.5.3 用时序控制方法实现浮点加减运算	106
3.2.2.6 三种代码制的比较	85	3.3.5.4 用程序方法实现浮点加减运算	107
3.2.3 定点数的加减运算	86	3.4 浮点和定点乘法运算	107
3.2.3.1 补码制加减运算规则	86	3.4.1 浮点乘法运算	107
3.2.3.2 补码制加减运算规则的常规证明	87	3.4.2 定点数的常规乘法	108
3.2.3.3 反码制加减运算规则	89	3.4.2.1 手算乘法和机器实现的功能要求	108
3.2.3.4 符号-绝对值码制加减运算规则	90	3.4.2.2 阵列乘法器	109
3.3 浮点数的表示和加减运算	90	3.4.2.3 保留进位加法器树构成的组合逻辑乘法器	109
3.3.1 浮点数的表示	90	3.4.2.4 乘法运算的迭代过程	111
3.3.1.1 浮点数的格式和编码	90	3.4.2.5 用时序控制方法实现乘法	111
3.3.1.2 规格化浮点数	91	3.4.2.6 乘法运算的控制流程	112
3.3.1.3 浮点数机器零和编码制	91	3.4.3 补码制和反码制的乘算法	113
3.3.1.4 浮点数格式举例	92	3.4.3.1 关于补码制的不同解释	114
3.3.2 浮点数的特性	92	3.4.3.2 加终端修正的补码制乘法	114
3.3.2.1 浮点数的表数范围	92	3.4.3.3 加终端修正的反码制乘法	115
3.3.2.2 浮点数和定点数表数范围的比较	94	3.4.3.4 布斯补码乘算法	116
3.3.2.3 浮点数的表数误差	94	3.4.3.5 布斯补码乘算法的原理证明	117
3.3.2.4 浮点数尾数基值 r_m 的选择	95	3.4.4 加速乘法的措施	118
3.3.2.5 浮点数运算引起的问题	96	3.4.4.1 加速乘法的原理	118
3.3.3 舍入方法	96	3.4.4.2 跳过“0”的措施	119
3.3.3.1 舍入的意义	96	3.4.4.3 跳过连续“1”的措施	119
3.3.3.2 恒舍(Chopping or Truncation)	97	3.4.4.4 两位一乘	120
3.3.3.3 恒置 $r/2$ 或恒置1(Jamming)	97	3.4.4.5 流水线乘法器	121
3.3.3.4 下舍上入(Rounding or Proximity)	97	3.5 除法运算	124
3.3.3.5 R^* 舍入(R^* Rounding)	99	3.5.1 手算除法和机器实现	124
3.3.3.6 ROM舍入(ROM Rounding)	99	3.5.1.1 浮点除法运算	124
3.3.3.7 补码制和反码制的舍入情形	100	3.5.1.2 定点除法的溢出情况	124
		3.5.1.3 手算除法步骤的分析	124
		3.5.1.4 机器除法与手算除法的区别	125

3.5.1.5 阵列除法器	125	4.2.1.3 立即数寻址	153
3.5.2 用时序控制原理实现除法	127	4.2.2 指标寻址方式	153
3.5.2.1 硬件组成	127	4.2.2.1 指标寻址方式或变址的意义	153
3.5.2.2 恢复余数除法	127	4.2.2.2 指标寻址的作用	154
3.5.2.3 不恢复余数除法	129	4.2.2.3 指标寻址的实现	154
3.5.2.4 不恢复余数除法的控制流程	130	4.2.2.4 相对寻址	155
3.5.2.5 补码制除法	131	4.2.3 间接寻址方式	156
3.5.3 用乘法实现快速除法	132	4.2.3.1 间接寻址的意义	156
3.5.3.1 用除数求倒作除法	132	4.2.3.2 间接寻址的作用	156
3.5.3.2 用连续乘法作除法	134	4.2.3.3 间址和变址兼用	157
3.6 逻辑运算和其它运算	135	4.3 小型计算机指令系统和CPU组织	157
3.6.1 逻辑运算和算术逻辑单元	135	4.3.1 PDP-11 计算机的 CPU 组织和 寻址方式	157
3.6.1.1 逻辑运算的性质	135	4.3.1.1 PDP-11 机的 CPU 组织	157
3.6.1.2 ALU 通用集成电路	135	4.3.1.2 PDP-11 机的寻址方式	158
3.6.1.3 ALU组件实现加法操作的原理	136	4.3.2 PDP-11 计算机的指令系统	159
3.6.1.4 ALU 组件实现算术运算和逻辑运 算的关系式	137	4.3.2.1 二地址双操作数指令	159
3.6.1.5 ALU 组件实现的全部算术逻辑运 算	137	4.3.2.2 一个半地址双操作数指令	161
3.6.2 双精度算术运算	139	4.3.2.3 单操作数指令	161
3.6.2.1 功能要求	139	4.3.2.4 转移类指令	162
3.6.2.2 双精度数据格式	139	4.3.2.5 跳转和子程序类指令(Jump and Subroutine Instruction)	162
3.6.2.3 双字长加/减和求补操作	139	4.3.2.6 条件码指令(Condition Code Instruction)	164
3.6.2.4 双精度乘法运算	140	4.3.2.7 其它类指令	164
3.6.2.5 双精度除法运算	140	4.3.3 DJS-130 计算机的指令系统和 CPU 组织	166
本章参考文献	141	4.3.3.1 DJS-130 机的 CPU 组织	166
习题	142	4.3.3.2 DJS-130 机的指令系统	166
第四章 指令系统与中央处理器组织	147	4.4 大、中型计算机指令系统和 CPU组 织	168
4.1 指令功能和指令格式	147	4.4.1 IBM 370 计算机的 CPU 组织和 指令格式	168
4.1.1 指令的功能	147	4.4.1.1 IBM 370 的 CPU 组织	168
4.1.1.1 指令的主要类型	147	4.4.1.2 IBM 370 的寻址方式	169
4.1.1.2 指令的要素	147	4.4.1.3 IBM 370 的指令格式	170
4.1.2 指令格式	148	4.4.2 IBM 370 计算机的指令系统	170
4.1.2.1 指令字长	148	4.4.2.1 算术运算指令	170
4.1.2.2 操作码字段	148	4.4.2.2 转移指令	172
4.1.2.3 地址码字段	149		
4.2 寻址方式	153		
4.2.1 直接寻址方式	153		
4.2.1.1 绝对寻址	153		
4.2.1.2 寄存器寻址	153		

4.5 堆栈计算机指令系统和 CPU 组织	173	5.1.4.2 扩大容量问题	197
4.5.1 HP 300 计算机的 CPU 组织	173	5.2 半导体和磁心随机存储器原理	197
4.5.1.1 堆栈型 CPU 组织原理	173	5.2.1 随机存储器一般组织	197
4.5.1.2 堆栈计算机的 CPU 寄存器	174	5.2.1.1 普遍逻辑模型	197
4.5.1.3 堆栈计算机的堆栈标记	175	5.2.1.2 历史的回顾	198
4.5.1.4 堆栈计算机的过程调用和退出	176	5.2.2 选址技术	199
4.5.1.5 堆栈计算机的 CPU 寄存器加载控制	178	5.2.2.1 一维地址译码或线选技术	199
4.5.1.6 堆栈计算机的程序环境和任务环境	178	5.2.2.2 二维地址译码或信号重合技术	199
4.5.1.7 提高堆栈计算机处理速度的栈顶结构	180	5.2.3 半导体存储细胞阵列	200
4.5.2 HP 300 计算机的寻址方式	181	5.2.3.1 线选法半导体存储细胞阵列	200
4.5.2.1 相对寻址	181	5.2.3.2 重合法半导体存储细胞阵列	201
4.5.2.2 指标寻址	182	5.2.4 磁心存储器原理	203
4.5.2.3 间接寻址	182	5.2.4.1 线选法磁心存储器	203
4.5.2.4 寻址方式和数据分配举例	183	5.2.4.2 电流重合法磁心存储器	204
4.5.3 HP 300 计算机的指令系统	185	5.2.4.3 二度半磁心存储器	207
4.5.3.1 存储器访问指令	185	5.2.4.4 各种磁心存储器组织形式的比较	207
4.5.3.2 堆栈和控制指令	186	5.3 半导体随机存储器组织	208
4.5.3.3 其它类型指令	186	5.3.1 半导体存储器件	208
本章参考文献	187	5.3.1.1 半导体存储器件分类	208
习题	187	5.3.1.2 双极型半导体存储器件	209
第五章 主存储器	193	5.3.1.3 MOS 型静态存储器件	210
5.1 主存储器的基本特性	193	5.3.1.4 MOS 型动态存储器件	211
5.1.1 主存储器和辅助存储器	193	5.3.2 半导体存储器集成电路	212
5.1.1.1 主存储器的作用和地位	193	5.3.2.1 小容量一位静态存储器	212
5.1.1.2 随机访问存储器	193	5.3.2.2 中容量一位动态存储器	213
5.1.1.3 串行访问存储器	194	5.3.2.3 中容量多位半导体存储器	213
5.1.2 主存储器容量	194	5.3.3 PDP-11 计算机半导体存储器	214
5.1.2.1 主存储器的二维结构	194	5.3.3.1 存储器组件阵列	214
5.1.2.2 主存储器地址空间	194	5.3.3.2 存储器周期时序控制	216
5.1.2.3 字节寻址计算机的不同方案	194	5.3.3.3 存储器工作周期流程	219
5.1.3 主存储器速度	195	5.4 存储器的系统结构问题	222
5.1.3.1 一次存储器访问过程	195	5.4.1 并行存储器	222
5.1.3.2 主存储器速度指标	196	5.4.1.1 地址空间的划分	222
5.1.4 改进主存储器性能的系统结构措施	196	5.4.1.2 访问周期的控制	224
5.1.4.1 提高速度问题	196	5.4.2 Cache 存储器	225
		5.4.2.1 访问局部化	225
		5.4.2.2 Cache 存储器工作原理——读操作	225
		5.4.2.3 Cache 存储器工作原理——写操作	225

.....	226	时序	248
5.4.2.4 映射函数——直接映射	226	6.2.2 三地址计算机的基本流图	249
5.4.2.5 映射函数——联想映射 (附联想存储器)	227	6.2.2.1 数据流图	249
5.4.2.6 映射函数——分组联想映射	228	6.2.2.2 数据流图与硬件组织的对应	250
5.4.2.7 替换算法——FIFO 算法	229	6.2.2.3 指令前趋图	250
5.4.2.8 替换算法——LRU 算法	230	6.2.3 一地址计算机的基本流图	252
5.4.2.9 Cache 存储器举例	231	6.2.3.1 数据路径和控制时序公共部分	252
5.4.3 虚拟存储器	232	6.2.3.2 指令系统和控制时序操作部分	252
5.4.3.1 主存-辅存层次与 Cache 存储器- 主存层次的比较	232	6.3 计算机基本流图的实现	254
5.4.3.2 主存-辅存层次的性能价格指标	232	6.3.1 数据路径的实现	254
5.4.3.3 主存-辅存层次基本信息传送 单位	234	6.3.1.1 寄存器及其内含功能部件的实现	254
5.4.3.4 页式虚拟存储器结构原理	234	6.3.1.2 多功能函数部件的实现	255
本章参考文献	236	6.3.1.3 从实现合理性对数据路径和前趋 图的调整	257
习题	236	6.3.2 前趋图时间标尺的确定	258
第六章 数据路径与控制流程	239	6.3.2.1 同步控制与异步控制	258
6.1 时序控制过程的流图分析方法	239	6.3.2.2 时钟节拍和控制周期	259
6.1.1 时序控制过程与计算结构	239	6.3.2.3 节拍脉冲与节拍电位的时间配合 关系	260
6.1.1.1 时序控制过程的层次性	239	6.3.2.4 实现时间标尺的例子	261
6.1.1.2 模块化结构概念	240	6.3.3 前趋图的实现	263
6.1.1.3 算法的图形描述方法	240	6.3.3.1 操作时间图	263
6.1.2 原始计算流图	241	6.3.3.2 操作控制器	263
6.1.2.1 计算流图的类别和组成	241	6.3.3.3 硬联逻辑控制器	263
6.1.2.2 原始数据流图	241	6.3.3.4 关于操作时间图例题的讨论	263
6.1.2.3 前趋图	242	6.4 多寄存器计算机的计算流图及其 实现	263
6.1.2.4 数据相关图	243	6.4.1 DJS-130 计算机的计算流图 实例	268
6.1.2.5 数据流图的冲突情形	243	6.4.1.1 数据流图	268
6.1.2.6 数据相关图的确定性条件	243	6.4.1.2 算逻指令和访主存指令的前趋图	271
6.1.2.7 数据流图用组合逻辑线路实现	244	6.4.2 DJS-130 机数据流图的实现	271
6.1.3 基本计算流图	246	6.4.2.1 中央处理器的加法器和通用寄存 器堆	271
6.1.3.1 条件构造和迭代构造	246	6.4.2.2 移位和进位预置线路	273
6.1.3.2 基本数据流图举例	247	6.4.3 DJS-130 机前趋图的实现	274
6.2 计算机基本流图分析	248	6.4.3.1 节拍和周期的产生	275
6.2.1 指令基本时序	248	6.4.3.2 操作时间图	278
6.2.1.1 计算流图与时序控制过程的 对应	248	本章参考文献	278
6.2.1.2 单时序计算机的一般数据路径	248	习题	278
6.2.1.3 单时序计算机的基本指令控制			

第七章 计算机微程序设计281	7.3.2 PDP-11 计算机的垂直微程序设计304
7.1 微程序控制的基本概念281	7.3.2.1 LSI-11 系统构成304
7.1.1 微程序与时序控制281	7.3.2.2 LSI-11 中央处理器305
7.1.1.1 实现机器指令控制时序的三种方法.....281	7.3.2.3 CP 1611 B数据组件305
7.1.1.2 固件的作用.....281	7.3.2.4 CP 1621 B控制组件307
7.1.1.3 微程序仿真.....282	7.3.2.5 CP 163 B微程序只读存储器组件308
7.1.1.4 固件工程的兴起.....282	7.3.2.6 微指令格式和微指令系统308
7.1.2 微程序控制的原理和分类282	7.3.2.7 MPS 系统特点308
7.1.2.1 维尔克斯创造的方案.....282	本章参考文献311
7.1.2.2 基本原理.....283	习题311
7.1.2.3 水平微指令与水平微程序.....283	第八章 输入输出与中断系统318
7.1.2.4 垂直微指令与垂直微程序.....284	8.1 概述318
7.2 微程序控制器设计284	8.1.1 输入输出子系统的特点318
7.2.1 微程序设计的基本问题.....284	8.1.1.1 处理机与外部世界.....318
7.2.1.1 微指令的并行性.....284	8.1.1.2 输入输出的异步性.....318
7.2.1.2 微指令地址的指定.....285	8.1.1.3 输入输出的实时性.....319
7.2.1.3 机器指令操作码分支.....285	8.1.1.4 输入输出的与设备无关性.....319
7.2.1.4 立即数的指定.....286	8.1.2 输入输出组织的基本原则319
7.2.1.5 微操作的时序.....286	8.1.2.1 输入输出特性对输入输出组织的影响.....319
7.2.2 微程序设计举例之一.....287	8.1.2.2 输入输出组织的自治控制.....319
7.2.2.1 数据流程图.....287	8.1.2.3 输入输出的分类组织.....320
7.2.2.2 微指令格式.....288	8.1.2.4 输入输出的层次结构.....320
7.2.2.3 微程序流程图.....290	8.1.3 输入输出方式321
7.2.2.4 微指令周期与主存周期的配合.....290	8.1.3.1 状态驱动输入输出方式.....321
7.2.2.5 条件转移的实现.....290	8.1.3.2 中断驱动输入输出方式.....321
7.2.2.6 微指令地址分配和微程序编写.....293	8.1.3.3 直接存储器访问.....322
7.2.3 微程序设计举例之二.....293	8.1.3.4 通道和输入输出处理机.....322
7.2.3.1 具有内总线结构的中央处理器.....293	8.2 状态驱动输入输出322
7.2.3.2 典型双操作数指令微程序流程.....293	8.2.1 输入输出指令功能322
7.2.3.3 寻址方式分支转移的实现.....296	8.2.1.1 程序控制 I/O 的特点和过程.....322
7.2.3.4 微指令格式和微程序控制器.....296	8.2.1.2 状态驱动 I/O 的程序实现.....323
7.2.3.5 操作码分支转移的 PLA 方法.....297	8.2.2 存储器映射输入输出325
7.3 微程序计算机实例300	8.2.2.1 地址映射原理.....325
7.3.1 PDP-11计算机的水平微程序设计.....300	8.2.2.2 存储器映射 I/O 的实现.....325
7.3.1.1 数据路径样板设计.....300	8.3 中断系统327
7.3.1.2 水平微程序控制器样板设计.....301	8.3.1 中断的基本概念327
7.3.1.3 分支微测试逻辑.....302	8.3.1.1 中断的性质.....327
7.3.1.4 典型指令的微程序.....302	

8.3.1.2 中断源的种类	327	8.4.2.8 通道的启停与检测	355
8.3.1.3 简单中断基本过程	329	8.5 输入输出接口	356
8.3.2 实际中断过程	329	8.5.1 数据传送的控制	356
8.3.2.1 多级中断的产生	329	8.5.1.1 输入输出接口的功能	356
8.3.2.2 判别中断条件的询问法	330	8.5.1.2 I/O 总线的组成和特性	356
8.3.2.3 判别中断条件的串行排队链法和向量 中断	330	8.5.1.3 同步总线的数据传送控制	357
8.3.2.4 判别中断条件的独立请求法	332	8.5.1.4 异步总线的数据传送控制	358
8.3.2.5 中断优先级	333	8.5.2 通用接口举例	359
8.3.2.6 自陷	334	8.5.2.1 并行接口	359
8.3.2.7 中断屏蔽	334	8.5.2.2 串行接口	361
8.3.2.8 中断屏蔽的实现	336	8.5.3 传输中的错码检测与纠正	363
8.3.2.9 中断现场的保留和恢复	336	8.5.3.1 检错纠错码的必要性	363
8.3.2.10 程序状态字	338	8.5.3.2 奇偶校验和代码最小距离的概念	364
8.3.3 中断程序举例	339	8.5.3.3 横向和纵向冗余校验	364
8.3.3.1 DJS-130 机中断程序举例	339	8.5.3.4 海明单纠错双检错码	365
8.3.3.2 PDP-11 机中断程序举例	341	8.5.3.5 循环冗余码	366
8.4 直接存储器访问和通道组织	346	8.6 输入输出设备	366
8.4.1 直接存储器访问(DMA)硬件组 织	346	8.6.1 概述	366
8.4.1.1 DMA 的功能和特性	346	8.6.2 用于输入输出的设备	367
8.4.1.2 DMA 的输入输出基本循环	347	8.6.2.1 电传打字机	367
8.4.1.3 DMA 控制器的组成和连接	348	8.6.2.2 显示终端	368
8.4.2 通道组织	348	8.6.2.3 行打印机	369
8.4.2.1 通道的作用和功能	348	8.6.3 用于辅助存储的设备	370
8.4.2.2 多路通道	349	8.6.3.1 磁盘机	370
8.4.2.3 选择通道	350	8.6.3.2 软磁盘机和温彻斯特磁盘机	372
8.4.2.4 成组多路通道	351	8.6.3.3 磁带机	374
8.4.2.5 通道吞吐率	351	本章参考文献	375
8.4.2.6 通道命令	352	习题	375
8.4.2.7 通道程序举例	353	附录 几种通用计算机系统纵览表	379

第一章 绪 论

1.1 计算机的特性和组成

1.1.1 计算机的主要特性

1.1.1.1 计算机的定义

“什么是计算机?”——这是一个比较难于回答的问题。原因是:计算机这一名词代表了性能极为多样化的一大门类事物。必须弄清楚它们的共性,才能给出一个比较准确的定义。简单说起来,计算机是一种不需人的直接干预能自动完成各种算术和逻辑运算的工具。这个定义把远至算盘、手摇计算机和电动计算机,近至市面上普遍出售的袖珍计算器等都排除在外,就因为这些计算工具一般都不是自动化的,而且不以逻辑运算作为它的主要功能之一。但是,即使这个附加了若干限制条件的定义,仍然是太广泛了,它甚至把慢速的机电式分析计算机也包括进来了。为了把这种早期用于统计工作的计算机和现代电子计算机区分开来,有必要强调计算机的快速性,作为上述定义重要的补充特性之一。此外,上述定义没有对参加运算的数据形式作出明确规定,以至于它还可能包括用连续物理量表示数据并基于数学模拟原理而实现计算过程的模拟计算机(Analog Computer)。因此,还必须规定:本书只讨论数字计算机(Digital Computer),它以数字化编码形式的信息作为加工对象。只要不加特殊说明,通常所称的计算机就都理解为数字计算机。这样,我们便可以归纳出一个范围比较确定而全面的计算机的定义:计算机是一种不需人的直接干预能够对各种数字化信息进行算术和逻辑运算的快速工具。

1.1.1.2 计算机的四大特性

下面分别对这个定义中包含的计算机的四个主要特性作一些说明。

计算机之所以能高速处理信息,除了采用高速的半导体器件以外,还必须依靠从计算机内部结构上采取许多“挖掘潜力”的措施,其中,解决信息处理过程自动化问题便是最起码的条件之一。目前计算机解决这个问题都是用存储程序(Stored Program)的方法,即把计算过程表示为由许多条指令组成的程序,和数据一起预先存入计算机的存储器。算题时,只要按一下启动按钮或从键盘上发出运行命令,程序就会控制计算机按照规定的顺序一条一条执行指令,自动完成预定的信息处理任务。上述高速半导体器件除了能构成快速算术逻辑运算部件以外,还可构成快速存储器,能在运行中高速完成指令和数据的自动存取。这样,高速开关器件与存储程序结构原理相结合,便产生了计算机的重要特性之一——快速性。假如没有存储程序,那么计算机至多只能象普通计算器一样,用手按键盘输入数据,随送随算,再快的内部计算速度也是不能充分发挥作用的。

计算机的另一重要特性是它的通用性。它把任何复杂繁重的信息处理任务分解为大量的基本算术和逻辑操作，反映在计算机的指令操作中，按照执行的先后次序，把它们组织成各种程序。这些程序不但包括由用户自己编写、从外部随时输入的用户程序；更大量的是预先编好，由计算机厂商提供、常驻在计算机内部的系统程序和应用程序包。这种存储好的程序能很快地从存储器中调出来运行，不但是实现计算机自动快速处理信息所必需，也由于十分灵活、易于变更，才使计算机具有极大的通用性。当然，应当强调指出：决定计算机通用性的因素当中，除了这种程序控制方式外，还有程序的内容也起着主要作用。由于计算机具有逻辑判断和处理的能力，所以它能把各种运算有机地组织成为复杂多变的计算和控制流程。计算机程序加工的对象也不只是数值量，而是形式和内容十分丰富多样的各种信息，例如语言、文字、图象、音乐等。表示这种普遍信息的有效方法是数字化信息编码。数字化编码技术不但保证了运算和控制的极高准确性，也是计算机赖以获得其逻辑判断和逻辑运算能力的基础。所以说，数字化信息编码和存储程序

计算机的四大法宝

计算机的四大特性

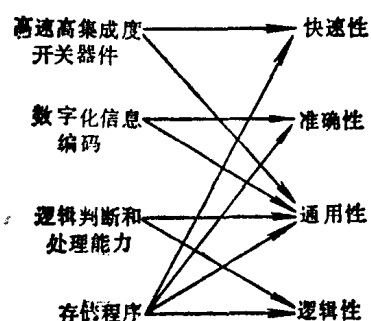


图 1.1 计算机的内部特性和外部特性间的对应关系

序结构原理相结合，便产生了计算机的另外几个重要特性——通用性、准确性和逻辑性。

这样，我们就可以归纳出来计算机内部拥有的四大法宝，对于使它成为高效、快速、通用的信息处理工具作用极大，它们是：

高速高集成度开关器件；
数字化信息编码；
逻辑判断和处理的能力；
存储程序。

1.1.1.3 计算机的外部特性与内部特性的关系

以上四点是从计算机的内部特性来讲的，而从计算机的外部特性来看，计算机的特殊能力蕴藏于它的快速性、准确性、通用性和逻辑性之中。计算机内外特性的对应关系示于图 1.1。带有箭头的线段表示计算机内部特性与其外部特性的因果关系，从上面的简单叙述已能大致理解它们的含义。图中有两处值得提一下：“通用性”被四个箭头所指向，可见它最集中地体现了计算机的结构特点。“存储程序”有四个箭头指向每一项计算机特性，可见它在计算机组成原理中是最关键的一条。因此，弄清楚存储程序的概念，研究它在计算机内部的实现过程，乃是了解现代计算机工作原理的关键，它将是本课学习的重点。

1.1.2 计算机的硬件组成

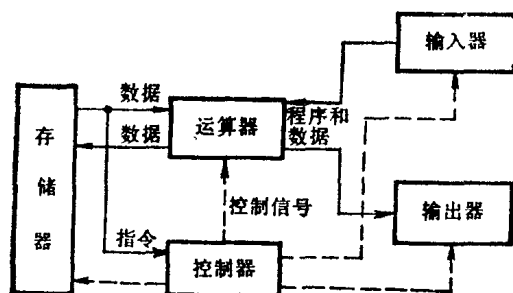
1.1.2.1 古典冯诺依曼计算机

存储程序概念最早是由匈牙利籍数学家 J·冯诺依曼(John Von Neumann)于 1946 年提出来的^[6]，他同时提出了一个完整的现代计算机雏型。30 多年来，虽然计算机结构经历了重大的变化，性能也有了惊人的提高，但就其结构原理来说，至今占有主流地位的仍是以存储程序原理为基础的冯诺依曼型计算机。

古典冯诺依曼计算机从存储程序原理出发得到一个最基本的组成框图如图 1.2 所示。它包

含下列五大功能部件：

1. 存储器——存储单元或存储字的集合体。每个存储单元存放一个字的信息，其二进制位数称为字长。存储单元用二进制码顺序编好地址，根据地址可以随机地对指定单元进行访问。冯诺依曼计算机采用的存储器有 4096 个单元，字长 40 位，用示波管屏面的电容效应存储电荷来表示信息，包括程序和数据。



2. 运算器——包含一个能执行算术逻辑操作的算逻单元，以及提供一个操作数和存放操作结果的累加器（另一个操作数来自存储器），执行乘、除法运算尚需要第二个寄存器。累加器和寄存器都是各由一组二态电子器件组成，存放一个字长的信息。

3. 控制器——用存储程序进行控制。程序由指令组成，指令包含操作码和地址码，分别指出操作种类和操作数在存储器中的单元地址。一般按顺序从存储器中取出指令并执行它，用一个指令计数器陆续提供指令在存储器中的单元地址。也可用转移指令的“地址码部分”指出下一条非顺序执行的指令地址。指令字和数据字都用二进制码表示，在外表上没有什么区别，在存储器中分区存放。这使得程序本身又可以当作数据一样被加工或修改。虽然对于指令目前已不采取修改的办法以利于程序的检查、调试、保护和再入，但程序可作为系统程序自动加工的对象，正是利用了上述重要特性。冯诺依曼计算机每一个字包含两条指令，即每一指令 20 位，其中 6 位是操作码，可给出 64 种操作，12 位是地址码，可为 4096 个存储单元地址编码，还有 2 位未用。控制器实现对指令的控制，解释指令的操作码及地址码，并根据译码将适当的控制信号送到运算器和机器中的其它部分。

4. 输入器和输出器——包括电传打字机、显示器、磁盘或磁带设备等。

1.1.2.2 计算机组织的学科任务

由此可见，古典冯诺依曼计算机是一台面向字的一地址二进制计算机。虽然后来的计算机在操作种类、指令功能（包括地址修改）、存储器层次、控制方式、输入输出能力等许多方面都有新的发展，但上述五大功能部件的划分仍是合适的，只是它们的内部构成和相互关系上已经发生了重大的变化。而“计算机组织”作为一门学科正是为了研究计算机运算器、存储器、控制器、输入器和输出器五大功能部件的相互连接和相互作用。相互连接是指物理上的连接，具有静态的性质；而相互作用是指逻辑上的联系，具有动态的性质。二者相结合，就构成了形态各异的冯诺依曼型计算机。

对计算机五大功能部件的划分还使我们得以引导出一个冯诺依曼型计算机的定义：计算机是一个快速计算工具，在统一的控制器作用下，它能接受数字化输入信息，根据存放在其存储器中的程序，对这些信息进行处理，产生结果信息并输出。

下面进一步从计算机快速性的要求上来阐明计算机组织的重要性。

要提高计算机处理信息的速度，除了采用高速高集成度开关器件完成信息运算和存储功能，并依靠存储程序实现处理过程自动化以外，还要解决一系列其它问题，其中最主要的就是怎样克

服输入输出的速度障碍。这个问题的解决对计算机组织和结构原理的发展有着很大影响。下面是为此采取的几个主要措施。

1.1.2.3 计算机作业方式

第一个措施是让计算机同时为许多用户终端和用户作业处理信息。

反映信息处理内容的程序和数据是从外部输入到存储器的，处理的结果还要从存储器送出来让用户看到。所有由人工产生的数据和程序输入计算机时，不论利用何种中介介质，最终总是离不开键盘的。这和计算机内部信息存储和处理的高速度要求形成了很大的矛盾。最根本的解决办法是设计和制造各种高速的输入输出设备，使人工操作键盘可以脱机进行。即先把数据和程序附在穿孔卡片或穿孔纸带、磁盘或磁带上，等积累了大批数据和程序并经仔细校对以后，才集中一次送入计算机，这样来摆脱计算机输入速度对人的依赖。计算机的批处理作业方式(Batch Mode)提供了这一能力。

但是，对用户最为方便的是交互作业方式(Interactive Mode)，即在联机情况下由用户直接操作键盘，随时能在输出装置的显示屏或打印纸上立刻看到计算机对他操作是否正确作出的回答。带有这种键盘输入和显示或打印输出的装置称为计算机终端(Computer Terminal)。现代的大型计算机系统往往要带几十甚至几百个这样的终端，同时为许多坐在终端前面的用户服务。计算机内部速度之高，完全足以应付全部终端对它的服务请求。不过，它必须做好周密的安排，按照时间先后和轻重缓急的次序轮流地把它的各种“资源”(Resources，包括设备、程序、数据等)提供给所有的用户共享。具有这种功能的系统称为计算机分时系统(Time-Sharing Computer System)。同时处理多个信息过程的工作方式称为多道程序(Multiprogramming)。这自然要求计算机完成极为复杂的组织管理工作，做到忙而不乱。能胜任这一管理工作的仍然是计算机内部被称为操作系统(Operating System)的那一部分存储程序，它是现代计算机高度自动化管理才能的集中体现，是任何一台计算机为管理其全部资源和用户作业所不可缺少的重要组成部分。

1.1.2.4 输入输出子系统

第二个措施是把计算机内部处理信息与输入输出信息在时间上穿插起来。

以上关于计算机终端的例子说明了一个道理，即输入输出设备本身的工作速度固然重要，但同样重要的是如何把它们组织在整个计算机系统之中。这是由于：即使输入输出设备本身速度再高，也总是无法与计算机内部信息处理速度相比拟。重要的是不能让计算机内部的处理器和存储器长时间地处于等待输入输出的状态而无所事事，必须把它们处理信息的过程与输入输出过程在时间上穿插起来，让计算机能够一边处理当前的信息，一边输出过去的结果，一边输入将来需用的信息，使自己一刻也不得空闲。如果这样计算机还“吃不饱”，就是说，还不能取得中央处理速度与输入输出速度之间的匹配，那么还需增加更多的输入输出设备，利用输入输出子系统来进行管理和控制。在输入输出数据太多，而且处理任务又重得忙不过来的时候，甚至于要增加一些小型或微型计算机来专门分担繁忙的输入输出任务，这些就构成了更加完整的输入输出子系统。

1.1.2.5 软件的配合