

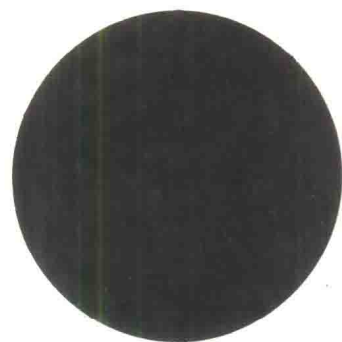
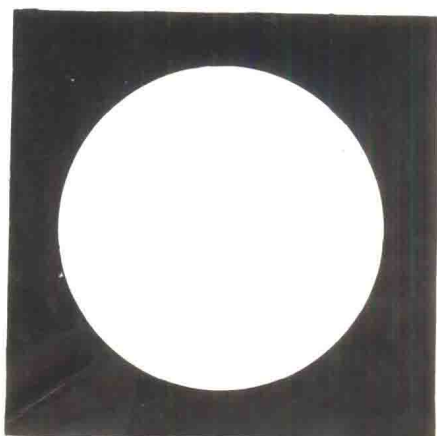
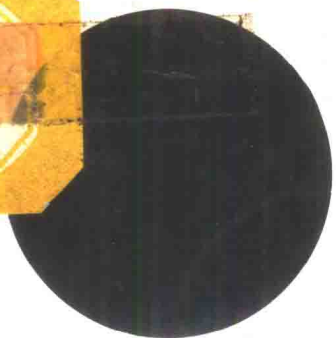
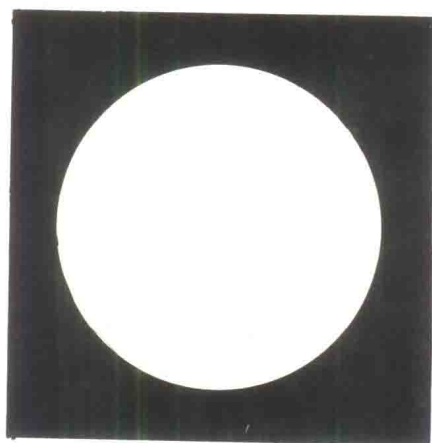
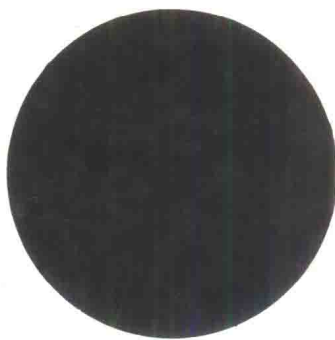
浙江大学出版社

高等学校教学用书

Computer Organization and Design

陶欣 编著

计算机 组成与设计



(浙)新登字 10 号

■ 高等学校教学用书
计算机组成与设计
陶欣 编著
责任编辑 杜希武

浙江大学出版社出版
浙江大学出版社计算机中心电脑排版
浙江印刷发行学校印刷厂印刷
浙江省新华书店发行

开本: 787×1092 毫米 1/16 印张: 16.5 字数: 422 千
1992 年 11 月第 1 版 1992 年 11 月第 1 次印刷
印数: 0001—2000

ISBN 7-308-01023-6/TP·071 定价: 4.65 元

前 言

“计算机组成与设计”是计算机系的专业基础课。在学习本课程之前,学生一般都只学过电子线路和计算机高级语言程序设计等课程,学生对计算机的了解还只是它的外部特性,而对计算机这个“黑匣子”的内部还一无所知,对计算机所具有的各种神奇功能只知其然而不知其所以然。本课程的目的就是让学生了解计算机的内部结构,将“黑匣子”打开,使学生不仅知其然,且亦知其所以然,为学习操作系统、汇编语言程序设计、系统结构等课程奠定必要的基础。

本书共分九章,以70学时左右实施教学。考虑到教学计划的不同,有的专业在讲授本课程时尚未学过数字逻辑电路,因此,本书中安排了“逻辑电路”一章,极其概要的介绍了逻辑电路的设计方法和一些常用逻辑电路的结构和功能,为后继章节的讨论提供必要的准备知识。如学生在学习本课程之前已经学过“逻辑电路”课程,则可以跳过这一章。但为了对本书中采用的一些术语和符号表示方法有所了解,建议粗略地看一遍,熟悉一下有关符号及图例。

各章的学时分配大致如下:

- 第一章(2-3), 第二章(6-8)
- 第三章(10-12), 第四章(14-16)
- 第五章(4-6) 第六章(8-10)
- 第七章(6-8) 第八章(7-8)
- 第九章(4-5)

本着“教学必须改革”的精神,作者在编写此教材时遵循了以下两个原则:①选材力求少而精,对基础知识和基本概念叙述清楚、严密论证,而对那些可以触类旁通的知识或稍加引导学生就能理解的尽量压缩,给学生留有足够的思考余地;②内容的安排上注意逻辑性,使章节之间尽量安排成承上启下的关系。每一章节的内容为解决什么问题都有一个明确的目的,这一章节解决了什么问题,还存在什么问题,为下一章节内容提出任务。一环扣一环,以提高学生学习的主动性。

主观愿望如此,客观效果如何,有待实践检验。欢迎提出宝贵意见。

作者

1992年2月

目 录

第一章 概论

§ 1.1 计算机的特点	1
§ 1.2 存储程序的概念	2
§ 1.3 计算机的基本组成	2
§ 1.3.1 运算器	3
§ 1.3.2 存储器	3
§ 1.3.3 输入设备	4
§ 1.3.4 输出设备	5
§ 1.3.5 控制器	5
§ 1.4 计算机系统的层次结构	5
§ 1.4.1 硬件与软件	5
§ 1.4.2 软件的基本内容	6
§ 1.4.3 计算机系统的层次结构	7
§ 1.4.4 硬、软件的功能分配与逻辑等价	8
§ 1.5 计算机系统的硬件设计	9
§ 1.6 计算机系统的总线结构	10
* 第二章 逻辑电路	
§ 2.1 基本逻辑门电路	13
§ 2.2 数字电路的属系	13
§ 2.3 布尔代数	15
§ 2.4 逻辑式的化简	17
§ 2.4.1 真值表与逻辑式	17
§ 2.4.2 用布尔代数化简逻辑式	17
§ 2.4.3 用图解法化简逻辑式	18
§ 2.4.4 无关条件的考虑	20
§ 2.5 组合逻辑电路	21
§ 2.5.1 定义	21
§ 2.5.2 组合电路设计	22
§ 2.6 时序逻辑电路	27
§ 2.6.1 基本触发器(锁存器)	27
§ 2.6.2 同步触发器	28
§ 2.6.3 触发器的激励特性	30

• 注:带*号的章节,如学时较少,可作为选学。

§ 2.6.4 时序电路分析	31
§ 2.6.5 同步时序电路的设计	32
§ 2.6.6 主-从触发器和边沿触发器	39
§ 2.6.7 常用时序电路介绍	42
§ 2.7 存储器和堆栈	45
习题	48
第三章 数据表示	
§ 3.1 进位计数制	50
§ 3.1.1 进位计数制的基本概念	50
§ 3.1.2 不同进制数之间的转换	51
§ 3.2 带符号数的表示	53
§ 3.2.1 真值与机器数	53
§ 3.2.2 机器数的原码表示法	53
§ 3.2.3 机器数的补码表示法	54
§ 3.2.4 机器数的反码表示法	56
§ 3.3 补码运算的基本规则	57
§ 3.3.1 真值、原码与补码间的转换	57
§ 3.3.2 补码的加法规则	58
§ 3.3.3 补码的减法规则	59
§ 3.3.4 补码加减运算实例	59
§ 3.3.5 机器数的移位规则	60
§ 3.4 定点数和浮点数	61
§ 3.4.1 定点表示法	62
§ 3.4.2 浮点表示法	63
§ 3.5 十进制数的表示	65
§ 3.6 字符的表示	66
§ 3.7 校验码	68
§ 3.7.1 奇偶校验	69
§ 3.7.2 海明码	71
§ 3.7.3 循环冗余码(CRC)校验	73
习题	79
第四章 运算方法与运算器	
§ 4.1 并行加法器的组成与快速进位的实现	82
* § 4.2 原码机器数的加减运算器	87
§ 4.2.1 原码加减法运算的算法	88
§ 4.2.2 原码加减法运算的硬件实现	88
§ 4.3 补码机器数的加减运算器	90
§ 4.3.1 补码加减运算的算法	90
§ 4.3.2 补码加减运算的硬件实现	91
§ 4.4 状态寄存器	91

§ 4.5 十进制加法器	93
§ 4.6 算术逻辑部件(ALU)	94
§ 4.7 定点乘法运算	98
§ 4.7.1 原码一位乘法	98
§ 4.7.2 原码二位乘法	100
§ 4.7.3 补码一位乘法	102
§ 4.7.4 补码二位乘法	106
* § 4.7.5 多位乘法概念	107
§ 4.8 定点除法运算	109
§ 4.8.1 原码比较法与恢复余数法	110
§ 4.8.2 原码不恢复余数法(加减交替法)	112
* § 4.8.3 补码不恢复余数法	113
* § 4.8.4 快速除法简介	116
§ 4.9 浮点运算	119
§ 4.9.1 浮点加减运算	119
§ 4.9.2 浮点乘法运算	120
§ 4.9.3 浮点除法运算	121
§ 4.9.4 浮点运算的舍入问题	121
§ 4.9.5 浮点运算的硬件实现	122
习题	123
第五章 指令系统和汇编语言	
§ 5.1 指令系统的基本概念	125
§ 5.1.1 指令的分类	125
§ 5.1.2 指令的格式	127
§ 5.1.3 寻址方式	129
§ 5.1.4 寻址方式应用举例	134
§ 5.2 机器指令与汇编语言	137
§ 5.3 PDP-11 寻址方式简介	138
习题	146
第六章 中央处理器(CPU)	
§ 6.1 CPU 的设计步骤	147
§ 6.2 控制器的功能及设计方法	147
§ 6.2.1 控制器的功能	148
§ 6.2.2 控制器设计方法	148
§ 6.3 寄存器结构与微操作描述——RPL 语言简介	149
§ 6.3.1 寄存器之间的并行传送	149
§ 6.3.2 数据的总线传送	151
§ 6.4 CPU 指令系统与总体结构	153
§ 6.4.1 指令系统的确定	153
§ 6.4.2 总体结构的确定	153

§ 6.5 控制器的设计	155
§ 6.5.1 几个基本操作的微操作序列	155
§ 6.5.2 指令的执行过程	158
§ 6.5.3 CPU 的定时系统	164
§ 6.5.4 控制器的硬件实现	164
§ 6.5.5 节拍信号的起停控制	168
习题	169
第七章 微程序控制器	
§ 7.1 微程序控制的基本概念	173
§ 7.2 微指令	174
§ 7.2.1 微命令字段的编码方法	174
§ 7.2.2 微地址段的组成	176
7.3 微程序设计实例	179
§ 7.3.1 模型机结构	179
§ 7.3.2 微程序设计	182
§ 7.4 微程序中的定时问题	184
§ 7.5 微程序控制的评价	185
习题	186
第八章 输入输出与中断系统	
§ 8.1 典型 I/O 设备的工作原理	187
§ 8.1.1 键盘输入设备	187
§ 8.1.2 纸带输入机	189
§ 8.1.3 打印机	190
§ 8.2 I/O 指令格式及 I/O 传送的特点	191
§ 8.2.1 I/O 指令格式	191
§ 8.2.2 I/O 数据传送的特点	192
§ 8.3 计算机与外部设备之间的数据传送	193
§ 8.3.1 程序控制方式的 I/O 传送	193
§ 8.3.2 中断方式的 I/O 传送	195
§ 8.3.3 直接存储器存取(DMA)方式	196
§ 8.3.4 I/O 处理器(I/O 通道)	197
§ 8.4 中断系统	198
§ 8.4.1 中断的类型	199
§ 8.4.2 中断请求的监测	200
§ 8.4.3 中断的控制	201
§ 8.4.4 中断源的识别	201
§ 8.4.5 中断优先级	202
§ 8.4.6 中断屏蔽	204
§ 8.4.7 优先级中断系统实例	205
§ 8.4.8 中断服务程序	206

§ 8.5 I/O 接口电路	208
§ 8.5.1 接口的主要功能	208
§ 8.5.2 接口的组成	208
§ 8.5.3 接口的分类	209
§ 8.5.4 并行接口电路举例	209
§ 8.5.5 异步串行接口电路	211
§ 8.6 I/O 处理器的工作原理	213
习题	216
第九章 存储系统	
§ 9.1 存储器的分类	218
§ 9.2 存储器的主要性能指标	219
§ 9.3 主存储器的组成以及与 CPU 的连接	220
§ 9.3.1 存储元件	220
§ 9.3.2 存储单元与它的编址	220
§ 9.3.3 半导体 RAM 的组成	222
§ 9.3.4 半导体只读存储器(ROM)	225
§ 9.3.5 半导体存储器组织举例	226
* § 9.4 辅助存储器	227
§ 9.4.1 磁表面存储原理	227
§ 9.4.2 磁记录方式	228
§ 9.4.3 磁盘存储器	234
§ 9.4.4 磁带存储器	237
§ 9.5 存储系统的层次结构	239
§ 9.6 多模块交叉存储技术	241
§ 9.7 高速缓冲存储器	244
§ 9.7.1 高速缓冲存储器的读/写操作	245
§ 9.7.2 映射函数(Mapping)	245
§ 9.7.3 替换算法	248
§ 9.8 虚拟存储系统	249
习题	252
主要参考文献	253

第一章 概 论

本章将简要地叙述计算机的特点及其应用情况,并从存储程序概念出发,介绍计算机的硬件组成,它所包含的主要模块,各模块的功能及相互联系;同时,引出“计算机系统”的概念,扼要地说明硬件和软件的各自功能及其相互关系;从设计方法学观点考虑,介绍计算机硬件系统的设计方法。通过本章介绍,让读者对计算机的全貌有一个初步的印象,明确本课程的任务,提高学习的主动性。本章中引出了一些专门术语,有些作了粗浅的说明,有些未加说明,建议读者对暂时不理解的术语作为问题记在心里,在以后各章的学习中去寻找答案。

§ 1.1 计算机的特点

通常所说的计算机,确切的名称应该是“电子数字计算机”。“电子”一词意味着它是由电子器件构成的,以区别于其他如机械式、机电式的计算机,它具有独特的高速性;“数字”一词意味着它所处理的信息是“离散”的或称“量化”的,区别于处理连续信号或模拟信号的模拟机,它具有独特的精确性。

计算机之所以会被运用,无非就是因为它比人能更有效、更好地完成某些特定的作业。它能接受更多的信息并能以更快的速度、更高的精度来处理这些信息。人笔算几星期或几个月才能完成的工作,计算机往往只需几分钟甚至更短的时间就能完成。特别是要求较短时间内处理大量数据的条件下,使用计算机往往是解决问题的唯一手段。

早期的计算机只是作为超功能的计算工具,用来解决人工无法解决的复杂数学问题,随着计算机功能的不断提高,软件的不断发展,现代计算机的应用已从数值计算扩展到非数值应用的领域,如控制系统、通讯、人工智能、图形识别,以及信息管理和处理等等,下面介绍几种典型的应用:

设计问题:对于工程设计人员来说,计算机是一种非常有用的计算工具。比如,设计超音速飞机机翼时必须考虑许多不同的因素,设计人员可以通过某种程序以数学方程的形式描述每一个因素,然后用计算机对各种不同的输入数据求得这些方程的数值解。

科学实验:在科研和实验中,可以用计算机来计算、存储和处理来自许多不同传感器的信息,尤其如遥测一类的系统中,计算机更是不可缺少的工具。在这种系统里,必须把接收到的信息迅速记录下来,否则便会丢失。所以,这类应用要求对固定和动态的条件都有快速、精确的处理能力。

过程控制:对于制造业来说,计算机也是一种有用的工具,可以用来自动监督产品的生产过程。例如,计算机经编程后可以控制铣床、车床等,也可以结合特殊的传感器来检查正在加工的零件并根据需要随时调节机床,其反应速度和精度是人所无法比拟的。如果零件有缺损则计算机可将其剔除,然后启动加工下一个零件。

模拟训练:有些技能训练,如驾驶飞机、控制人造卫星、操纵宇航飞船等,往往代价很高,且十分危险,此时就可用计算机来模拟这些条件进行训练。受训人员既可得到长时间的“临

场”训练,又不会危及自身或他人的安全,代价小得多。

以上只是粗略的介绍了一些典型的应用。计算机应用领域已渗透到各个方面,尤其是随着大规模集成电路技术的发展,微型计算机的应用已普及到了人们的日常生活中。

但是,计算机终究是一种机器,也跟其他机器一样,必须受人的操纵和控制。虽然计算机有“电脑”之称,表示它具有记忆、思维、判断的能力,但归根到底这些能力也是人赋予给它的。在人们编制好程序并将它存入存储器之前,计算机完全一无所知,一无所能,甚至连怎样接收数据也不知道。不管怎么高级的计算机,总必须由人直接或间接地“告知”它怎样工作才行。因此,要充分地开展计算机的效用,就必须充分认识计算机的能力和它所受的限制。计算机的能力主要表现在:

(1) 重复操作:计算机可以重复执行完全相同的操作千万次,而绝不会“疲劳”和“厌倦”;

(2) 高速度:计算机处理信息的速度极快;

(3) 灵活性:通用计算机经不同的程序设计后可以用来解决许多不同类型的问题。

(4) 精确性:计算机的计算精度可由程序设计者的要求而确定。

计算机的限制主要表现在:

缺乏“直觉”或“灵感”,人往往能“眉头一皱,计上心来”或“灵机一动”,突然找到某个问题的解答方法,但计算机只能“按部就班”地按预定的方案处理问题。

§ 1.2 存储程序的概念

为了告诉计算机做什么事,按什么步骤做,就需要编制程序,使计算机能够按规定的程序自动的工作。

冯·诺依曼(John Von-Neumann)等人提出了计算机设计的一些基本思想,概括起来有如下一些要点:

(1) 采用二进制形式表示数据和指令。

(2) 将程序(包括数和指令序列)事先存入主存储器中,使计算机在工作时能够自动高速地从存储器中取出指令加以执行。这就是存储程序概念的基本含义。

(3) 由运算器、存储器、控制器、输入装置和输出装置五大基本部件组成计算机硬件系统,并规定了这五部分的基本功能。

这样一些概念奠定了现代计算机的基本结构思想,并开创了程序设计的时代。到目前为止大多数计算机仍沿用这一体制,称为诺依曼机体制,上述结构思想就称为诺依曼思想,它的最主要之点就是存储程序概念。

§ 1.3 计算机的基本组成

原始的诺依曼机在结构上以运算控制器为中心。演变到现在,计算机已转向以存储系统为中心,图 1-1 表示其最基本的组成框图。

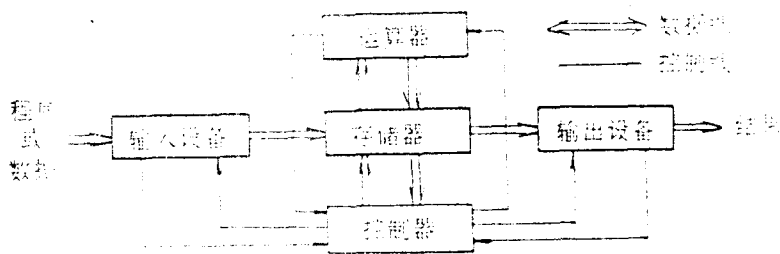


图 1-1 计算机的简单框图

计算机的工作程序,包括指令序列和原始数据,它告诉机器做什么事,按什么步骤做,以及所要处理的原始数据信息。操作人员将程序通过输入设备送入存储器,启动运行后,计算机就从存储器中取出指令送到控制器去识别,分析该指令要求做什么事,控制器根据指令的含义发出相应的命令,例如将某存储单元中存放的操作数取出送往运算器进行运算,再把运算结果送回存储器指定的单元中。当运算任务完成后,就可以根据指令序列将结果通过输出设备输出。操作人员可以通过输入设备(或控制台)启动或停止机器的运行,或对程序的执行进行某种干预。

通常将运算器和控制器合称为中央处理器 CPU(Central Processing Unit)。在采用大规模集成电路的微型计算机中,往往把 CPU 制作在一块芯片上。中央处理器和主存储器一起组成所谓主机部分,而将输入设备和输出设备 I/O(Input / Output)称为外围设备(Peripheral Device)。有时计算机的处理结果需要送入磁带、磁盘一类存储器保存,以便下一次工作时再输入到该机或送入更高一级的计算机进行再处理。从主机的角度看,磁带、磁盘等存储器属于 I/O 设备的范畴。但从整个计算机系统来看,它们又属于存储系统的一部分。为了区别,通常将它们叫做辅助存储器或称为外存储器,而把图 1-1 中的存储器叫做主存储器或内存储器。

下面对五个部件分别作一简要说明。

§ 1.3.1 运算器

运算器是执行算术运算和逻辑运算的部件,简称算术逻辑部件 ALU(Arithmetic and Logic Unit),它是计算机实现高速运算的核心。ALU 的特性主要由它的功能来表征,最简单的 ALU 只能执行加法运算和求反,加 1 等逻辑运算,功能较强的 ALU 可以执行加、减运算,以及乘法、除法运算,功能更强的 ALU 不仅能执行定点加、减、乘、除,而且还能执行浮点运算,甚至向量运算。

运算速度和并行运算的位数也是 ALU 的重要特性标志,尤其是运算器的位数表征了整个计算机中通用寄存器的长度,即计算机的基本字长。

§ 1.3.2 存储器

存储器(Memory)是存放指令和数据的部件。如前所述,把指令和数据存储于同一个存储器中是冯·诺依曼结构的主要特点。

一个存储器是成千上万“信息单元”的集合体。每个信息单元对应有一个位置编号，称为“存储单元地址”，或简称“地址”。

存储器的基本操作是“读”和“写”，统称“访问”。每次访问只能针对一个单元，因此每当要从存储器读取一个单元或向存储器写一个数据到某单元时，必须先给定要访问的“地址”。

每访问一次存储器所交换的信息单元称为“寻址单位”。存储器的寻址单位可以有：

一个二进制位——bit，称一个“码位”；

四个二进制位——nibble，称“半字节”；

八个二进制位——byte，称“字节”；

16个二进制位——称“双字节”；

32个二进制位——称“四字节”。

存储器分为主存(内存)和辅存(外存)两类。主存用来存放当前将要执行的指令和数据，它的存取速度较快，可与处理器的速度基本匹配；辅存用来存放大量暂时不运行的指令和数据，它的速度较慢，但容量很大。

如果访问存储器中任何一个单元所需的时间都是相同的，与该单元所在的地址无关，称这种存储器为“随机访问”存储器，主存储器必须具有这一特征。具有这一特征的存储器件有磁芯存储器、半导体存储器。目前大多使用的是后者；如果访问某个单元所需的时间与该单元所在位置有关的，称这种存储器为“顺序访问”存储器。辅助存储器一般都属此类，如磁带、磁盘即是。有时把磁盘称“半随机访问”存储器。

存储器的主要特征是：

存储容量——包含信息单元的多少。

字长——每个信息单元的二进制位数。

存取时间——访问一个存储单元所需的时间(仅对“随机访问”存储器有意义)。

§ 1.3.3 输入设备

输入设备(Input Device)的任务是将记录在某种介质上的信息转换成计算机能够识别和记录的形式送入计算机。

输入设备按其工作原理分为：机械设备、机电设备、电子设备、光电设备等。可作为计算机输入设备的有：

(1)穿孔设备：如光电输入机、卡片机等，通过光电变换或其他方法将穿孔纸带或穿孔卡片的穿孔信息转换成电信号，送入计算机；

(2)键盘：可以直接通过键盘输入程序或数据；

(3)磁带、磁盘：将磁化形式记录其上的信息通过转换并放大后的电信号送入计算机；

(4)模/数转换装置：在自动检测和自动控制装置中所检测到的原始信号往往是模拟信号，需通过A/D转换(模/数转换)成计算机所能接受的数字信号；

(5)图形信息识别与输入装置：如光笔、图形数字化仪等；

(6)字符信息的识别与输入装置；

(7)语言信息的识别与输入装置。

更为理想的是人们通过书写或讲话直接将手写字或自然语言输入计算机。目前已有这方面的产品，并正在不断完善中。

§ 1.3.4 输出设备

输出设备(Output Device)的任务是将计算机处理的结果转换成人们便于识别的或其他机器能接受的形式输出。

可作为计算输出设备的有：

- (1)打印设备：电传打字机，针式打印机，宽行打印机等；
- (2)绘图仪；
- (3)CRT 显示器，或家用电视机；
- (4)磁带、磁盘；
- (5)D/A 转换器，数/模转换和调制。

输入/输出设备与中央处理器在物理上相距很远，比起处理器来说传送速度比较低，所以解决处理器与外部设备之间的同步问题是计算机设计的重要内容之一。

§ 1.3.5 控制器

控制器(Control Unit)是计算机的管理机构和指挥中心，它通过对存储于存储器中的指令的解释，有条不紊地向 ALU、存储器、输入/输出等部件发出有关的操作命令。例如，在某个时刻，某种条件下需要将某个寄存器清 0，在某个时刻，某种条件下需要向存储器的地址寄存器送地址码等等。这种“寄存器清 0”，“向寄存器送数”等操作是最基本的操作，它是在一个时钟脉冲内完成的操作，称为“微操作”。

一个计算机中包含多少种“微操作”完全取决于机器的指令系统和处理器内各个部件的组织结构形式，如单总线结构、多总线结构等。

每条指令包含的“微操作”种类和“微操作”执行的序列因指令的功能而异。

因此，控制器就是计算机中指令的解释机构和执行机构，更具体的说就是“微操作信号发生器”，有人把控制器比喻为人的大脑，一个“微操作控制”信号相当于一根“神经”，通向计算机某个部分，进行某种操作。

控制器内部包含了指令的译码部分、各种控制条件信号以及定时信号。

根据形成“微操作”控制信号的方法不同，控制器的结构有两种类型：即硬联逻辑或称组合逻辑型控制器和微程序控制器。

§ 1.4 计算机系统的层次结构

§ 1.4.1 硬件与软件

上节概述了一台数字计算机最基本的硬件组成。为了使计算机系统具有尽可能完善的功能，一个完整的计算机系统要包含硬件与软件两大部分。

硬件(Hardware)通常是指构成计算机的设备实体。例如前述五大部件这样一些人们可以触摸到的设备和器件，当然还包括如何把它们组成一个计算机整体的体系结构。一个计算机系统应具备哪些基本功能，包含哪些部件，这些部件按什么结构方式相互连接成有机的整体，各部件应具备何种功能，采用什么样的器件和电路构成，以及在工艺上如何进行组装等，都属于硬件设计的范畴。

软件(Software)通常泛指各类程序和文件。它们实际上是由一些算法(说明如何完成某任务的指令序列)和它们在计算机中的表示所构成,体现为一些触摸不到的信息,所以称为软件。有时把编制程序所依赖的文件也归入软件范畴。按照这种概念,从系统中除去计算机实体(硬件)之后剩下的所有部分都属于软件。它包含:应用程序、系统程序、文件等。

硬件与软件的组合构成了计算机系统。显然二者是相互依存的,硬件是物质基础,没有硬件或者没有良好的硬件支持就谈不上软件的执行或高效率软件的编制,反之,没有软件或没有完善良好的软件,计算机就无法工作或不能高效率地工作。指令系统反映了一台计算机的能力大小,可以看成是硬件与软件的结合点,它既是硬件设计的出发点,也是计算机最低级语言的基本语句集。

§ 1.4.2 软件的基本内容

前面已经指出,计算机系统是一个硬件与软件的综合体,在硬件与软件之间,在系统软件与应用程序之间存在一种层次结构。软件是在硬件支持下工作的,而应用软件又需要系统软件的支持。

基本的软件系统可归纳为下述几方面:

1. 应用程序

在计算机技术中常常用“信息”这样一个名词,它泛指数据、符号、语言、文字、图形等。数字计算机就是进行信息处理的机器。为了将计算机用来解决各种问题,必须编制相应的应用程序。例如各种科学计算程序、数据统计与处理程序、情报检索程序、企业管理程序、生产过程自动控制程序等。由于计算机已应用到几乎所有领域,因而应用程序将是多种多样、极其丰富的。

2. 系统程序

(1)语言处理程序

计算机硬件只能直接识别代码化的机器语言,即用数字代码表示的指令序列。现在已经提出了许多种程序设计语言,相应地要编制编译程序或解释程序来进行转换。两者通称为“翻译程序”,编译程序先将用高级语言编写的程序翻译为机器语言程序,然后执行。解释程序则是边翻译边执行。

(2)管理程序

需要编制一些系统程序来管理调度计算机系统的运行,以便尽可能合理地调用系统的硬件资源与软件资源,这是使系统能高效率运行和易于扩展的关键。

较完善的计算机系统常常可以提供这样的工作方式:操作人员通过带显示器的键盘终端使用主机,开机后屏幕上将显示出机器所处的工作状态和它可供用户选择的各项功能;用户选取所需的功能后,通过键盘或其他手段输入程序;每步操作后显示屏上将作出反应,并提醒你下一步应采取的动作;如果操作有误,显示屏上将显示出错标志,并指出错误性质,甚至提出纠正办法。显然,这样的工作方式对使用者很方便,但需要一套复杂的管理程序支持。

为了提高计算机使用效率,功能稍强的计算机都允许多个用户同时使用,即允许多道程序分时地在一台计算机中运行。为了合理地调用与共享硬软件资源,又需要一个复杂的管理系统。操作系统 Operation System (O.S)就是功能强、规模大的管理程序的典型。

(3)服务程序

一个完善的计算机系统往往配置有许多服务性程序,它们或者被包含在操作系统之内,

或者被操作系统所调用。例如：将记录在某种介质上的用户程序和数据通过输入机送入主存储器时，就需要解决下列一些问题：如何将程序送入主存的指定区间，又如何检测输入是否正确。为此需要编制输入引导程序，先将引导程序送入计算机，机器执行“引导程序”以引导用户程序进入主存的指定区间，并检测输入过程是否正确，如果无误就可启动执行该用户程序。

在程序的输入或调试过程中，常常需要进行修改、整理或进行几段程序间的连接，为此又编制了“编辑程序”。

为了判断机器是否正常工作，如果有故障则希望能判明出错部位以至定位到某一器件上，许多系统配置了“诊断程序”。

以上仅仅是从使用者容易想到的角度引出软件的部分基本内容，事实上系统软件正在极为迅速地发展和丰富，使计算机的功能越来越强。

§ 1.4.3 计算机系统的层次结构

当使用一个较完善的计算机系统来解决问题时，我们可以这样描述它的工作过程：

提出任务→用适合于程序设计的方式描述算法过程(例如用流程图)→用某种语言编制程序→由计算机将它编译为机器语言程序→由硬件实现。

从这一角度出发，可用图 1-2 的形式表示计算机系统的分层结构。

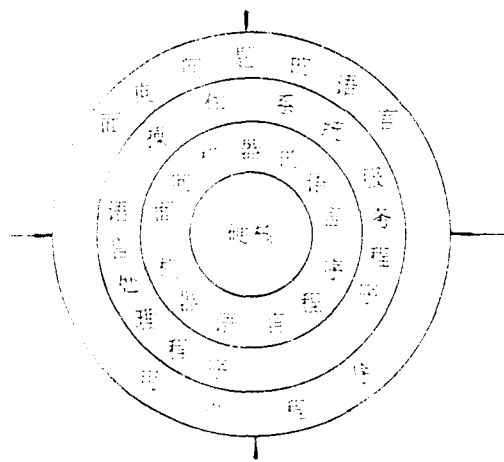


图 1-2 计算机系统层次结构示意图

图 1-2 表示计算机系统硬件与软件的组成及相互间的关系。从外向内看，它表明了从用户提出任务到硬件执行的过程。用户提出任务，然后用面向问题即面向算法的高级语言编写程序，称为用户程序。在操作系统控制下调用系统的硬件与软件资源，例如编译程序，将用户程序翻译为机器语言程序。汇编语言虽然比代码化的机器语言高级些，但它仍是与特定机器密切相关的语言，所以也属于面向机器的语言这一级中。最后，一切程序都需由硬件来实现，在分层结构中就将硬件称为“硬核”，或称为裸机。

在学习控制器的组成方法后，读者将会了解到许多计算机用微程序控制方法来解释并执行指令。这样硬核部分又可以分为两级。从程序员的角度看，某台计算机具有一个指令系

统,表明它能做哪些事,可作为用户编制程序的依据,计算机硬件将能执行它,这就是通常所说的传统机器级。从硬件设计者或维护调试者的角度看,还有基础的一级,即计算机内部有一个微程序控制器,它具体地解释并执行机器指令,这一级可称为微程序控制级。

强调计算机系统的层次概念,有助于我们对整个系统建立一个整体概念,便于根据不同的工作任务要求,从各自的角度并从相应的层次去了解与分析系统的组成和特征,或完成这一级的组成任务。显然,从不同的角度可以将计算机系统表示为各种类型的层次结构,图 1-2 仅是其中的一种。

还需要提及一个常见的概念即“虚拟机”。从软件工作者的角度来看,在编写用户程序时主要关心的是用何种语言编写,机器能否识别并且执行。如果用机器语言 L_1 编写,所看到的是一台能执行语言 L_1 的机器 M_1 ,这是一台实际的机器。如果用高一级的语言 L_2 编写,所看到的是一台能执行语言 L_2 的机器 M_2 。然而机器实际上是依靠编译程序将 L_2 翻译为 L_1 才能执行的,所以称 M_2 为虚拟机。对程序员来说,理想的程序语言应当与日常的语言文字和算法相一致,这和机器所能直接理解的语言 L_1 之间的差距很大,可能需要设计一些中间语言,以便将高级语言逐级翻译。因此从语言的层次上又可提出一种分级的概念,如图 1-3。

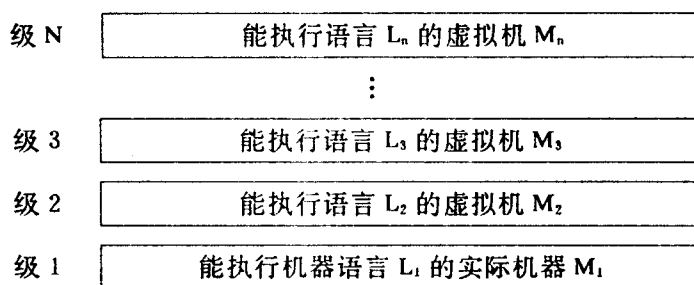


图 1-3 从语言角度区分的多级机层次图

用硬件直接执行高级语言是一个方向,但目前仍较困难,代价太高,所以目前多采用软件编译的方法,例如第三级的用 L_3 编写的程序可以由运行在 M_2 级的翻译程序(用 L_2 编写的)将它翻译成由 L_2 描述的程序,再由运行在 M_1 级的编译程序翻译成由 L_1 描述的程序,当然,如果 L_3 和 L_1 之间的差距不是很大,通常就直接翻译成 L_1 。

§ 1.4.4 硬、软件的功能分配与逻辑上的等价

设计计算机系统时首先需解决一个功能分配问题,即哪些功能应由硬件实现,哪些由软件实现。例如乘法功能,可以设置硬件乘法器,则指令系统中含有乘法指令,也可以只设置加减运算器硬件,用子程序实现乘法,即用软件实现乘法。又如键盘命令识别,可以设置专门的功能键与译码电路来提供命令信息;也可以只设置一些通用键,依靠命令识别程序将约定的通用键组合形成命令信息。这说明某种机器功能的实现往往既可以由硬件直接实现,也可以在较少的硬件支持下通过软件来实现。

硬件与软件的功能实现上如何分配,决定于所选定的设计目标、系统的性能价格比等因素,并与当时的技术水平有关。因而没有一种固定不变的原则和分界线。

早期的计算机依靠硬件实现各种基本功能。小型计算机的出现曾经形成一种简化硬件的趋势,即让硬件只完成较简单的指令系统功能,依靠软件实现更高级一些的功能,这就使

得结构简单而功能较强的小型机得以实现。这种做法称为硬件软化。

随着集成电路技术的发展,可以用比较少的代价由硬件来实现较强的功能。因此又出现了一种软件硬化的趋势,即将一部分原来由软件实现的功能(如乘除运算、浮点运算)改由硬件实现。这样系统将有更高的处理速度,并在软件支持下具有更强的功能。实际上硬件与软件的功能都在向前发展,需要根据现实的技术条件和不同的设计目标来恰当地分配各级的功能。

微程序控制与设计技术的发展使计算机内部结构和硬、软件的功能分配又出现了新的动向。对指令的解释与执行是通过运行在最低级的微程序来实现的,利用程序设计的技巧和扩大微程序的容量,可以使原来属于软件级的一些功能被纳入微程序一级之中。微程序和其它一些固定不变的程序被存放在固定存储器之中,因而称作“固件”(Firmware)。软件、硬件更加相互渗透且紧密结合,出现了所谓软件固化的趋势。

上面是从设计者的角度处理硬、软件之间的功能分配。如果从程序员的角度,则往往主要关心机器能执行哪些基本功能,以作为编制程序的基本依据,至于这项功能是由硬件还是由软件来实现则是等效的。例如乘法运算可以通过执行一条指令由硬件实现,也可以由软件子程序来实现,也可以由一段微程序来实现。在这个意义上我们认为:硬件与软件在逻辑上是等价的。

§ 1.5 计算机系统的硬件设计

上节中介绍了计算机系统的层次概念,目的是让读者对整个系统有一个整体概念。本课程的任务是要帮助读者去掌握物理机器(或称传统机器)级的基本特征。对物理机器级的深入了解是创建高性能的“虚拟”机的物质基础。

考察一台具体的计算机时,常常从三个级别上加以考虑:

(1)计算机的系统结构,所谓“系统结构”通常是指由程序员(确切地说是机器级语言程序员)所能看到的一组“资源”,例如机器的指令系统、通用寄存器、处理器、状态字、地址空间、输入输出设备等,在这一级上主要考虑如何选择指令系统,设置多少通用寄存器,应配置多大容量的存储器,配置多少输入输出设备等。这些问题是整个计算机系统设计的最高层次的决策。

(2)系统结构之下的“硬件组成”级,在这一级主要考虑的是如何利用处理器、存储器、输入输出设备等硬件组成具有特定功能的机器?特别是要考虑机器的“微结构”,例如,机器是否采用“微程序”控制?在同样的系统结构下可以组织起多少不同的计算机(如不同的处理器速度、存储器大小等等)?

(3)“硬件实现”级,在这一级上把计算机系统看成导线、电缆、集成电路芯片,以及各种分立元件的集合。关心的是半导体技术、时钟相位、信号电平及其他电参数的选择。

本课程的重点是在“硬件组成”级,有时也需要适当讨论“硬件实现”的问题,而“系统结构”级的讨论是“系统结构”课程的任务。

从设计方法学观点考虑,计算机的设计也可以分成三个级:

- (1)处理器级;
- (2)寄存器级;
- (3)门级。