

小型/微型计算机硬件设计

G. D. 克拉夫特 W. N. 托伊 著

陈森锦 王希仁 译

科学出版社

1984

内 容 简 介

本书有三大特色。首先，它是论述小型计算机与微型计算机硬件设计的绝无仅有的专著。其次，本书是从系统的高度去透视硬件设计的有关问题。第三，本书力求反映计算机设计的最新思想。

全书共分九章。主要介绍计算机原理发展史、数字逻辑电路、整机结构、存储器组织结构、存储器访问方式、指令系统、常规控制逻辑与PLA控制逻辑、直接程控输入/输出操作以及通道控制的输入/输出操作。

本书可作为计算机专业人员或接触小型/微型计算机的有关电气工程技术人员的参考书，亦可作为大专院校有关专业的教科书。

George D. Kraft Wing N. Toy
MINI/MICROCOMPUTER HARDWARE DESIGN
Prentice-Hall, 1979

小型/微型计算机硬件设计

G. D. 克拉夫特 W. N. 托伊 著

陈森锦 王希仁 译

责任编辑 李淑兰 孙月湘

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

*

1984年12月第 一 版 开本: 787×1092 1/16
1984年12月第一次印刷 印张: 25 3/4
印数: 0001—21,500 字数: 585,000

统一书号: 15031·612

本社书号: 3764·15—8

定价: 4.00 元

译者序

随着集成电路工艺技术的突飞猛进，小型计算机和微型计算机在计算机领域中的地位与日俱增。新的超级小型计算机不断地向大型计算机提出挑战；另一方面，小型计算机又面临着微型计算机的激烈竞争。32位微处理机芯片的出现正在加剧这场角逐。现在，用微处理机芯片做成的8位和16位微型计算机正蜂拥般进入每个家庭和社会的各个角落。这些进展使得“计算机社会”成了指日可以实现的人类崇高理想。社会科学家现在热切地讨论起计算机将会给人类社会带来的种种变革。他们指出，第一次工业革命——蒸汽机时代使人类从笨重的体力劳动中解放出来；而第二次工业革命——计算机时代将使人类从繁重的脑力劳动中解放出来。小型/微型计算机在这场社会现代化进程中扮演着极其重要的角色。因此，研究小型/微型计算机的设计方法显然是当务之急。

贝尔实验室研究人员 G. D. 克拉夫特和 W. N. 托伊花了五年的时间，撰写了《小型/微型计算机硬件设计》一书。内容丰富，分析精辟，说理透彻。其主要特点是从系统的角度来论述硬件的设计。因此，指令系统的设计、存储器管理方法、控制逻辑、总线结构以及输入/输出方法是讲述的重点。此外，还讨论了多处理机结构和网络化的有关问题。这本书反映了小型/微型计算机硬件设计的当代水平。译者希望，它将会有助于我国计算机工作者了解这一领域的最新研究成果。

由于水平所限，错误疏忽之处在所难免，敬请读者指教。

陈森锦

王希仁

原 序

本书起初拟以小型计算机的使用为主题。在撰写过程中逐步修改而转向介绍小规模处理机的硬件设计（处理机与计算机这两个术语在此处为同义词）。本书的主旨是论述小计算机的设计与使用方法。在撰写原稿之初必须决定在软件方面要侧重介绍哪些内容。由于我们两人天天都在研究各种硬件设计方法和软件设计思想，因而在开始时我们自然倾向于详细论述软件发展的各个方面。但后来我们确信，至今也仍然确信，小型/微型计算机的软件技术需要另写一本书才行。所以，最后采取的决策就是在本书中只论述小规模计算机的硬件设计和系统设计的有关问题。

具体来说，我们尽了一切努力把这类计算机作为计算系统的构件来处理，它们以经济实用的方式分布于这样或那样的应用环境中，以致既简化了日常的维修工作，又改善了整个系统的可靠性。根据这个方针，在课文和每章的习题中，我们都从基本设计原理出发讲解了小计算机被置于各种应用系统中所遇到的有关控制方面的各种问题。重点是讲解流程图，而不是讲解为特定的小型计算机或微型计算机专门设计的个别程序的编程方法。详细的软件定义与基本概念归并在附录里。我们相信，这种叙述方法可以更加鲜明地突出各个构件与一个应用系统之间的硬件连接方式；而且书中详细介绍的控制原理也就可以普遍适用于各种小机器，而不是只专门针对某种小机器。

由于LSI工艺技术日新月异地蓬勃发展，因而写一本关于小型/微型计算机的书就好比是想指出一个活动目标的准确位置。所以，我们作了这样的尝试：在每一章介绍基本的设计思想、定义以及早已成熟了的设计方法；同时利用附于每章末尾的习题向读者介绍最新的设计思想。这些习题故意安排得很难。在许多情况下并没有一个绝对正确的答案，相反地，读者必须首先从列出的许多答案中找出适当的答案，再设法加以优化。本书的对象是大学的高、低年级学生和新生。需要使用小型/微型计算机的电气工程师和（或）计算机科学家也可以把它作为参考书。本书讲述的小型计算机和微型计算机所使用的硬件控制方法既不是定论，也不是包罗万象。它只介绍一些我们认为与新的发展方向有关的设计思想和原理；新的LSI工艺技术正在形成一个全新的计算机技术——分布处理，这就是我们所说的新发展方向。

第一章叙述小计算机的发展历史，从ENIAC和Mark I一直讲到用微处理机作为中央处理单元设计而成的最新小型计算机与微型计算机。本章末尾的习题特意提出了组织一个小型/微型计算机系统所面临的各种实际问题，并要求说出分布处理的严格含义。

第二章讲述用来组成一台小计算机的各种数字逻辑电路。讲述的内容包括各种存储单元、集成电路工艺技术概况以及诸如ROM, PROM, EPROM, PLA和RAM等LSI器件的技术细节。本章末尾的一些习题提出如何把这些器件应用于特殊的场合。

第三章论述通用计算机结构的基本特点。数据传输电路和总线系统被视为设计任何一种数字计算机所必须考虑的基本设计内容。本章还介绍了计算机系列化这一设计思想，并讲述了小型计算机与微型计算机生产厂家如何根据这一设计思想采取行动。本章

末尾的习题进一步发展了这些设计思想。

第四章研究内存储器的组织结构,并讨论操作寄存器组、读/写寄存器阵列以及后进先出(LIFO)缓冲器或下推栈的用法等设计思想。本章的习题提出了各种存储器结构的硬件实现方法。

第五章讨论如何决定一个特定处理机的最合适字长以及可以使用的各种指令格式。还详细地研究了中央处理机为访问主存储器中的一个操作数所采取的各种方法。这些寻址方法是从实际的硬件实现办法这一角度来讨论的。本章的习题把这些寻址方式具体应用于各种中央处理机结构。

第六章规定数字计算机指令系统的确切定义,并介绍机器命令的一种划分方法。此外还简要讨论了如何根据特定用途的需要去优化指令系统的设计。

第七章讲解小计算机控制部分的常规设计方法以及另一种新方法——用可编程逻辑阵列(PLA)来实现控制逻辑。还举了一个实际的设计例子,在本章末尾的有关习题中又进一步加以引伸。

第八章详细讲解小型计算机和微型计算机的程控I/O操作。本章叙述的内容包括查询方法、插入机构和中断机构。

第九章讨论通道的一般概念,以及直接内存存取操作的具体内容。此外还从实际结构和可靠性的角度探讨了串行通信通道的用途。

五个附录为读者提供了下列详细参考资料:(a) MOS工艺,(b) 双极工艺,(c) 软件特性,(d) Bell System公司的MAC-8微处理机,(e) Tandem16——一种高可靠性的商品化小型计算机。

本书原稿是作为贝尔实验室(Bell Laboratories)内部进修教育计划的部分教材,花了五年时间编写而成的。其中的一部分文稿是著者在伯克利的加利福尼亚大学和芝加哥的伊利诺斯工学院讲课时写就的。

致谢

本书是在许多人的帮助下完稿的。在我们撰写本书的过程中,承蒙四位先生多方指教。他们是:J. R. 希汉(John R. Sheehan),原在贝尔电话实验室,现在Western Electric公司工作;E. P. 费尔奇(E. Pierce Felch),W. A. 德普(W. A. Depp)和E. A. 艾兰德(E. A. Irland),这三位先生一直在贝尔实验室工作。我们之所以能动笔写这本书,全凭这四位先生的热情支持与鼓励。

我们谨向贝尔实验室教育中心和N. F. 福斯特(Norman F. Foster)深表谢意。我们根据贝尔实验室内部进修教育计划从事教材的编写工作时得到了福斯特先生的大力支持。在本书从动笔到定稿的几年里,这种支持一直是始终如一的。我们还要特别感谢伯克利的加利福尼亚大学的E. S. 库(E. S. Kuh)教授和T. E. 埃夫哈特(T. E. Everhart)教授。在他们的帮助下,W. N. 托伊(Wing N. Toy)才能够在伯克利度过1973—1974这一学年。本书的部分文稿就是在此期间写就的。

贝尔实验室印第安山技术资料部的K. K. 哈戈皮恩(Kareen K. Hagopian)女士极其出色地协助整理本书的文稿,在此谨表谢意。

贝尔实验室印第安山技术资料部的P. 洛普雷特(Patrick Loprete)先生在本书文

稿与封面的最后审定中出力最多。在两年多的时间里，洛普雷特先生为我们两人的文稿进行了修改润色。

衷心感谢 R. A. 米线 (Ralph A. Meacham) 先生领导下的印第安山技术资料部与我们的合作。

最后还要强调，如果没有贝尔实验室的全力支持，如果没有它的各种设施所提供的方便，本书决不可能以目前的形式奉献于读者面前。

G. D. 克拉夫特

W. N. 托伊

目 录

译者序

原序

第一章 计算机原理发展史	1
1.1 引言	1
1.2 早期计算机	4
1.2.1 首批开创者	4
1.2.2 阿塔纳索夫-莫克利诉讼案	7
1.2.3 同一体系: 从 ENIAC 到 UNIVAC	7
1.2.4 IAS 计算机及更高级的机器	9
1.3 计算机市场的发展	10
1.4 小计算机的出现	12
1.4.1 MIT 旋风号计算机	12
1.4.2 MTC, TX-0, TX-1 和 TX-2	13
1.4.3 E-101, G-15 和 LGP-30	13
1.4.4 PDP-1	14
1.4.5 CDC-160	16
1.4.6 PDP-5	19
1.4.7 PDP-8	19
1.4.8 IBM1800 与 1130	20
1.4.9 Interdata 4	21
1.4.10 PDP-11 系列	24
1.4.11 Data General 公司的 Nova 计算机	25
1.4.12 PDP-15	25
1.4.13 Interdata 7/32	27
1.4.14 Interdata 8/32	28
1.5 小型计算机: 在计算机规模的不断演变中出现	31
1.5.1 “小型计算机”名称的第一次出现	31
1.5.2 根据主存容量确定机器类别	32
1.5.3 小型计算机的正式定义	33
1.5.4 中型计算机的正式定义	33
1.5.5 大型计算机的正式定义	34
1.5.6 小型化的小型计算机: 过渡性的机种	34
1.6 微处理器: LSI 工艺发展的必然结果	35
1.6.1 MOS/LSI 工艺的出现	35
1.6.2 Intel 4004 微处理器	36
1.6.3 Intel 8008 微处理器	39
1.6.4 微处理器基本结构	41
1.6.5 微处理器市场的早期状况——继 Intel 之后	42
1.7 “微”与“小”的比较	43
1.8 应用类别的合理划分	45
1.8.1 按计算机用途分类	45
1.8.2 按可靠性要求分类	45

1.9 向分布式处理系统的方向发展	47
1.10 小结	48
参考文献	48
习题	50
第二章 数字逻辑电路	53
2.1 DTL, TTL 与 ECL 逻辑电路	53
2.1.1 二极管-晶体管逻辑	53
2.1.2 晶体管-晶体管逻辑	53
2.1.3 发射极耦合逻辑	55
2.1.4 速度-功率的比较	56
2.2 逻辑功能	57
2.2.1 基本逻辑门	57
2.2.2 异或门	59
2.2.3 线逻辑	60
2.2.4 线逻辑的应用	60
2.3 存储单元	61
2.3.1 触发器特性	61
2.3.2 锁存器工作原理	62
2.3.3 触发器的一般形式	62
2.3.4 主从工作方式	64
2.3.5 边沿触发工作方式	66
2.3.6 JK 触发器的通用性	67
2.4 集成电路	68
2.4.1 芯片的集成度	68
2.4.2 小规模集成电路	68
2.4.3 中规模集成电路	68
2.4.4 大规模集成电路	69
2.5 LSI 控制存储器: ROM	69
2.5.1 ROM 的特性	69
2.5.2 ROM	70
2.5.3 PROM	72
2.5.4 EPROM	73
2.6 PLA	73
2.7 LSI 数据存储器: RAM	74
2.7.1 RAM 器件的发展情况	74
2.7.2 静态 MOS RAM 存储单元	75
2.7.3 动态 MOS RAM 存储单元	76
2.7.4 双极静态 RAM 存储单元	78
2.8 小结	79
参考文献	79
习题	81
第三章 整机结构	84
3.1 计算机基本结构	84
3.1.1 机器的组成	84
3.1.2 哈佛机与普林斯顿机	86
3.1.3 存储器访问的局部集中性	92
3.2 内部结构	93

3.2.1 执行周期	93
3.2.2 中央处理单元	94
3.2.3 控制器	95
3.2.4 微处理机芯片的典型结构	97
3.3 数据传输选通逻辑	98
3.3.1 寄存器互连逻辑	98
3.3.2 专用数据通路与分时公共数据总线	101
3.3.3 单路选通与双路选通	103
3.3.4 数据总线结构	106
3.4 总线系统	109
3.4.1 基本考虑	109
3.4.2 单总线结构	109
3.4.3 双总线结构	110
3.4.4 三总线结构	112
3.4.5 一元化 I/O 与存储器总线的设计原理	112
3.4.6 微处理机总线结构	114
3.5 数据传输电路的划分方法	117
3.6 系列机原理与功能模件集装原理	121
3.6.1 兼容性原理	121
3.6.2 系列机原理	122
3.6.3 主机选购件	123
3.7 小结	124
参考文献	124
习题	125
第四章 存储器组织结构	130
4.1 内存储器的管理	130
4.1.1 存储器系统的分层结构	130
4.1.2 交错存储器操作	133
4.1.3 操作寄存器组	134
4.1.4 专用寄存器	136
4.2 寄存器群的组织结构	137
4.2.1 读写并行式寄存器阵列	137
4.2.2 单存取寄存器阵列	139
4.2.3 双接口并读式RAM	140
4.2.4 寄存器栈	142
4.3 下推栈	145
4.3.1 栈的特性	145
4.3.2 TOS 的管理方法	146
4.3.3 栈的物理结构	149
4.3.4 栈控制逻辑电路	150
4.4 小结	154
参考文献	154
习题	155
第五章 存储器访问方式	159
5.1 字长考虑	159
5.1.1 出发点	159
5.1.2 单字指令与多字指令的对比	160
5.1.3 决定字长时的折衷考虑	162

5.2 指令格式	162
5.2.1 操作码的考虑	162
5.2.2 四地址指令格式	163
5.2.3 三地址指令格式	164
5.2.4 两地址指令格式	165
5.2.5 单地址指令格式	167
5.2.6 零地址指令格式	168
5.2.7 单地址机器	170
5.3 寻址系统	170
5.3.1 寻址系统设计概念	170
5.3.2 直接寻址方式	172
5.3.3 间接寻址方式	174
5.3.4 多级间接寻址方式	175
5.3.5 自动变址寻址方式	177
5.3.6 变址寻址方式	178
5.3.7 基本地址寄存器寻址方式	180
5.3.8 广义存储体转接寻址方式	183
5.3.9 固定页寻址方式	185
5.3.10 程序相关分页寻址方式	187
5.3.11 通用寄存器寻址方式	187
5.4 小结	192
参考文献	192
习题	193
第六章 指令系统剖析	197
6.1 指令系统	197
6.1.1 宏指令定义	197
6.1.2 指令长度类别	197
6.1.3 操作码类别	199
6.2 机器命令类别	201
6.2.1 数据迁移命令类	201
6.2.2 计算命令类	202
6.2.3 输入/输出命令类	208
6.2.4 控制命令类	210
6.2.5 特殊功能命令类	217
6.2.6 扩充功能命令类	220
6.3 高效指令系统的设计	221
6.4 指令系统的其它优化方法	222
6.5 小结	223
参考文献	223
习题	223
第七章 常规控制逻辑与 PLA 控制逻辑	230
7.1 引言	230
7.2 控制功能	230
7.3 指令时序	232
7.3.1 按序寻址	233
7.3.2 操作数读取	234
7.3.3 无条件转移	235
7.3.4 条件转移	235

7.3.5	中断寻址	237
7.3.6	指令预取	239
7.3.7	指令译码与执行	240
7.3.8	周期时序发生器	242
7.3.9	移位寄存器方法	243
7.3.10	指令周期的编排	244
7.3.11	公共的指令读取周期	244
7.3.12	利用逻辑门设计译码器	245
7.3.13	指令执行	247
7.3.14	控制逻辑设计例解	248
7.4	总体控制逻辑的设计步骤	257
7.5	两级控制逻辑	258
7.6	可编程逻辑阵列	261
7.6.1	PLA 输出选择	263
7.6.2	PLA 在周期时序逻辑中的用法	264
7.7	用标准 PLA 作为基本构件	265
7.8	用标准 PLA 实现加法指令	267
7.9	现场可编程逻辑阵列	270
7.10	异步存储器读取	271
7.11	小结	273
	参考文献	274
	习题	274
第八章	程控输入/输出操作	276
8.1	引言	276
8.2	插入	278
8.2.1	程控 I/O 与插入原理	278
8.2.2	软件控制的插入方法	283
8.2.3	固件控制的插入方法	285
8.3	中断方法	290
8.3.1	程序中中断的概念	290
8.3.2	单级中断机构	292
8.3.3	硬件查询法核查中断	296
8.3.4	多级优先中断机构	308
8.3.5	二维多级优先中断机构	302
8.3.6	优先权分配的动态调整	306
8.3.7	利用下推栈进行中断管理	307
8.3.8	处理机状态字	309
8.3.9	广义的上下文转折	314
8.3.10	中断机构的微程序控制	318
8.4	小结	319
	参考文献	320
	习题	320
第九章	通道控制的输入/输出操作	327
9.1	通道概念	327
9.2	直接内存存取操作	331
9.2.1	DMA 的基本特点	331
9.2.2	专用 DMA	335

9.2.3 选择式 DMA	336
9.2.4 复用式 DMA	338
9.2.5 使用 DMA 调度程序	340
9.2.6 微处理机的 DMA 操作	342
9.3 串行通信方法	346
9.3.1 串行通道概述	346
9.3.2 专用局部串行通道	348
9.3.3 专用外部串行通道	350
9.3.4 公用串行通道	351
9.4 小结	354
参考文献	354
习题	355
附录A MOS 工艺	362
A.1 PMOS	362
A.2 NMOS	362
A.3 CMOS	363
附录B 双极工艺	365
B.1 双极晶体管	365
B.2 肖特基二极管箝位的晶体管	366
B.3 集成注入逻辑	367
B.4 肖特基 I ² L	369
附录C 软件特性	371
C.1 定义	371
C.2 软件的一般类别	372
C.3 语言翻译程序	373
附录D MAC-8: 一种通信用的微处理机	377
附录E Tandem16不间断计算机系统	388
汉英对照索引	391

第一章 计算机原理发展史

1.1 引言

计算机工业形成于四十年代末期，成熟于五十年代。那一时期的计算机所表现出来的特点是：计算能力集中于庞大笨重的中央处理机，而输入/输出设备很有限；此外，作业是以按序成批处理的方式进行的。这些早期计算机所使用的真空管耗电量大，又是造成机器体积庞大的主要原因。机器的可靠性很差，而且主存储器的容量只有数百个字至数千个字。此外，主存储器的工作周期一般为几百微秒至几十毫秒。这一时期是真空管数字计算机的时代，人们认为这是**计算机技术的第一代**。第一代从1945年一直绵延到1958年左右。在这一时期中，计算机的价格高得惊人，只有最大的公司和机关团体才买得起。

1948年，贝尔实验室(Bell Laboratories)发明了晶体管，到1957年—1958年间发展成为可投入市场的电路元件。那时晶体管开始应用于数字计算机，这就诞生了**计算机技术的第二代**，或称之为**晶体管数字计算机时代**。这一时代从1958年算起，约至1964年为止。有了晶体管之后，设计人员就可以建造可靠性更高、处理速度更快、成本更低的数字计算机。晶体管数字计算机的功耗比真空管数字计算机小得多。再加上磁心存储器工艺技术也取得了相应的进展，因而指令周期从数百微秒减少到数十微秒。由于这种计算机系统的成本仍然很高，因而集中处理仍是基本的设计思想。在第二代时期内，磁心主存储器的容量扩大到几十万个字。

苏联人造卫星的发射促使美国计算机技术加速向前发展。从各种运载工具发射出去的美国空间飞行器，需要有极端可靠而又小巧灵活的先进计算机来进行导航和控制；这就引起了对集中式计算原理进行一场变革。于是逐渐形成了一种新的设计思想，也就是将各种级别的计算能力（也就是“智能”）分配到远离中央处理机的地方。中央处理机仍然可以同这些分布式智能设备进行有意义的对话，但决策将是分级进行的；最初由当地的控制设备开始处理，尔后要经过若干级预处理才能与远地的中央处理机进行通信。就地控制这一设计思想可能是在五十年代末期各种军事发展项目的推动下形成的。这些项目要求被导航的导弹和其它的抛射器必须在飞行中自己作出决策。这样的控制设备并不需要又大又复杂的数字计算机；相反，较小的专用计算机显得更为经济实惠，也更容易组装在导弹等飞行器里。这样的小型计算机还为用户提供了垂手可得的处理能力，这有助于揭开五十年代末期笼罩在大型数字计算机上的神秘面纱。这种处理能力也促使应用工程师去设法利用这些小型计算机作就地控制之用，这与大量的数据处理大不相同。

随着集成电路工艺技术的加速发展，一个芯片甚至可以实现好几个逻辑功能，这就导致了**计算机技术第三代**的到来。这一时代从1964年延续到七十年代中期。在这一时期中，一个芯片上可以做12个或更多的门，这一工艺技术通常称为小规模集成(SSI)。

到了1970年,这一工艺技术已发展到可以在一个芯片上做100个或更多的门,而且这样的芯片可以用作整机系统的一个功能逻辑部件。这样的工艺技术称为中规模集成(MSI)。许多工艺学家所称的计算机技术的第三代半也随之问世了〔1〕。所带来的后果是,半导体生产厂家受到了日益巨大的压力,要求它们更进一步提高一个集成电路(IC)芯片上所集成的门数。由于电子产品市场有着如此强烈的要求,半导体生产厂家不得不去努力提高产量并研制更复杂的芯片。它们的最终奋斗目标是要把一个中央处理机做在一个芯片上,同时要达到更高的可靠性、更低的功耗和更小的尺寸。正是在第三代半计算机时代,一个芯片上真的可以实现多达1,000至10,000个门的元件密度。前一种工艺技术(每片集成1,000个门)称为大规模集成(LSI),这种产品在1975年开始涌入市场。后一种工艺技术(每片多达10,000个门)称为超大规模集成(VLSI)或巨规模集成(GSI)。这表示着第三代半已经发展到它的尽头。

正是在六十年代末和七十年代初,小规模计算机开始正式称为“小型计算机”,它的用途也开始定形。MSI使得小型计算机可以兼备大型处理机的许多结构特点,而其成本只有后者的几分之一。在此期间,小型计算机生产厂家所提供的软件也变得相当高级,使得用户不必自己去开发应用程序和系统软件。到了1972年底,小型计算机已经可以作为一个系统构件放在任何需要的应用环境中去工作,只要这样使用在经济上划得来即可。由于使用小型计算机在经济上很合算,因而它们的应用很快扩及很多不同的领域,其中包括过程控制、数据通信以及事务数据处理。由许多台小型计算机配置而成的系统则用来分配计算能力、提高整个系统的可靠性并使维修工作方便到只是简单地更换整个中央处理机。

六十年代中期除了发展起先进的计算机工艺技术以外,更重要的是发展了容错和超可靠的计算机技术。在美国空间计划的推动下,设计了许多容错数字计算机;这些机器即使在它们的控制电路出现一个或多个故障时仍能继续工作。采用这种技术的既有宇航用的计算机,也有民用计算机,如IBM System/360。这种容错技术可以检查出各种数字故障。为了区分所查出的故障并提供必要的信息以便更快地修复故障设备,发展了诊断程序检查方法。为了对付可能出现的机器故障,所采取的一种方法是系统的各个组成部分甚至整个处理机都配有备份。这种冗余措施广泛用来为宇航计算机创造超可靠的系统环境。

Bell System公司在它的1型与2型电子交换系统(ESS)中使用了双处理机冗余方法。这两个系统都由两台每步互相进行校对的处理机所组成。两台处理机的某些特定输出信号彼此进行对比,以便确定它们的机器状态和运算结果是否一致。这个过程实际并不多费什么时间。如果两者不一致,就用故障检测程序去找出出错的处理机。在出故障的机器尚未修复以前,系统可以一直以“单工”方式进行工作(也就是只有一台处理机处于联机工作状态)。故障识别逻辑将使失效机器自动脱离“校对”工作状态。此外,Bell System公司还首创了一种高分辨力的自检诊断程序,它可以查出故障是发生在哪一种可更换的电路部件里。Bell System公司在设计上述两种系统时,规定的可靠性指标是四十年内停机时间只有两小时。

民用小型计算机生产厂家并不特别注重机器的可靠性,它们只是给主存储器里的每一个字配备一个奇偶校验位,并对磁带上所存储的信息进行循环冗余字符校验。在1974

年秋以前，没有哪一家小型计算机厂商所采取的措施超出上述这些基本的校验方法。这是因为从成本考虑不宜于使用诸如自检、误差检测与校正、以及高分辨力的诊断程序等措施。这些措施意味着要增加硬件和软件包，而小型计算机的一般用户或者不需要如此高的可靠性，或者不愿意为此多花费这么多金钱。

到了1974年秋，Data General公司推出了它的Eclipse小型计算机系列。根据用户的选购，Eclipse系列机的交错式磁心存储模块与半导体存储模块可以附设误差检测与校正电路。这个附加电路使得主存储器出现故障条件时关键的应用程序仍能继续工作。这种措施是Texas Instruments公司首创的，但Data General公司继承并发展了这一方法。这一行动表明一部分小型计算机厂商认识到机器可靠性已是民用用户的一个主要着眼点，并为此而付出了极大的努力。Hewlett Packard公司的HP21MX系统和HP3000系统的半导体存储模块也使用了误差校正码电路。在这以前，超可靠计算机技术主要是属于海军、空间计划、Bell System公司以及生产大型计算机的大公司（如IBM公司）所专有。

在1972年底，小型计算机的可能用途是如此广泛，工业部门解决不了供不应求的状况。正是在这个时候，利用新出现的LSI工艺做成的第一批微处理机问世了。微处理机显然是Intel公司于六十年代末期致力于存储器研究工作所附带取得的研究成果〔2, 3, 4〕。它一出现就被广泛采纳并蓬勃发展起来，影响之广遍及计算机工艺技术的全部领域。它的应用首先见之于手持式计算器以及专用控制机。用专门的小型计算机来做专用控制机是不经济的，而一旦使用微处理机来做，就显示出强大的生命力。

在微处理机向前发展的同时，存储器工艺技术也取得了重大的进展。磁心存储器的工作周期几乎减少了一半，一些主要厂家也开始利用半导体存储器来作为计算机的主存储器。微处理机结构与这些半导体存储器相结合，提供了廉价的分布式计算能力。

微处理机的问世使计算机进入了革命性的时代。数字系统的基本构件已不再是电子元件，而是各种功能单元，如存储器、对话式终端设备和中央处理单元（CPU）。微处理机工艺的前身是先进的MSI工艺。MSI将各种逻辑功能、存储器和接口做在几种半导体芯片里，因而所有这些功能不必用一个个电路板来实现。微处理机生产厂家在原有的MSI CPU上又增加了一些新功能，采取的解决办法是提高半导体芯片的功能复杂性。这样的复杂性促使人们去发展可用作系统构件的真正LSI电路。复杂性的增加带来了成本的明显下降。售价远低于一千美元的先进计算机终于成了现实。这些功能很强的廉价计算机使计算机技术很快地进入它的第四代。Diebold学会〔1〕称这一代是“受用户欢迎的一代”。人们当时就认识到，在这样的工艺技术条件下，计算机将以经济有效的方式分布于任何需要的地方〔5, 6〕。LSI和GSI将得到广泛的使用，群式处理机或处理机群将用来完成从前一台大机器所做的工作。这些群式处理机通常将是一些非常考究的廉价微处理机。一般认为第四代是成熟于1978年。表1-1汇集了计算机工艺技术的四个时代以及它们的主要特点。

生产厂家的保守估计是，到八十年代初期，单片随机存取存储器（RAM）的存储容量可达256K位；而单片双极逻辑电路将包含1万个以上的门。它们还预计，这种超级芯片工艺技术的突破将引起小型计算机和微型计算机的性能出现另一次飞跃。

非常高级的廉价计算机可供个人（或就地）使用，这一事实所包含的全部意义至今

表 1-1 四代计算机技术的演变

第 一 代	第 二 代	第 三 代	第 三 代 半	第 四 代
1945年——ENIAC, 这是第一台商品化的计算机, 于1951年正式使用	1958年	1964年	1970年	1978—1980年
专用	通用	兼容式系列机, 集中控制的多处理机	与第三代相同; 出现了小型计算机网络	分布式控制的多处理机
事务处理与科学计算	数据处理	信息处理	信息处理	联机解题
真空管	晶体管	集成电路	中规模集成电路	大规模集成电路
电子元件	逻辑门	多门电路	寄存器、译码器、加法器、I/O设备	处理机、存储器、终端设备
简单, 少	复杂, 少	复杂, 多	可编程	可动态修改
数万指令/秒	数十万指令/秒	数兆指令/秒	数兆指令/秒	数十兆指令/秒
机器语言子程序, 公用程序库, 符号汇编程序	高级语言 (COBOL, FORTRAN), 监督程序, 宏汇编程序, 执行程序	操作系统, 多种语言, 多程序, 程序包, 模拟语言	可扩充语言, 元语言编译程序, 硬件分程序, 对话系统	面向用户与面向用途的更高级语言, 用微码或硬件来实现的系统软件

尚未被现代计算机应用专家所充分理解。阐明它的应用范围的一种叙述方法就是从历史发展的角度来说明, 也就是研究计算机原理的发展如何导致小型计算机和微处理机的出现。第一章的其余部分将讨论计算机的发展史, 叙述的角度是用户的需要以及为了满足那些需要而发展起来的性能特点。我们并不冀求面面俱到, 重点将放在小型计算机和微型计算机原理的发展史上。

1.2 早期计算机

1.2.1 首批开创者

数字计算机的设计原理并不是从某个人的头脑里一下子蹦出来的。相反地, 它是由一小批实用科学家从 1937 年开始整整经历了 13 个年头的不懈努力才逐渐形成的。在这批人当中, 最有影响力的有三个人, 他们的开创性工作奠定了存储程序数字计算机的发展基础。在 1947 年 1 月于哈佛大学召开的大型数字计算机第一届会议前夕, J. W. 莫克利 (John. W. Mauchly) 在一篇论文中〔7〕概括介绍了当时宾夕法尼亚大学正在研制中的 EDVAC 机与那时已有的四种其它计算机所存在的根本区别; 这四种机器是: Bell Laboratories 的继电器计算机, Mark I, Mark II, 以及 ENIAC。在评价这些机器的时候, 莫克利也为数字计算机的三名开拓者树碑立传。这三名科学家的名字与计算机的发展紧紧联系在一起, 他们每个人的贡献都值得好好研究。

1937 年, G. R. 斯蒂比茨 (George R. Stibitz) 为了取乐, 在家里用几个继电器、几个手电筒电池和一些灯泡组装了一个简单的二进制加法器。H. 特罗普 (Henry Tro-

pp) 在谈及计算机发明以前的那段光辉历史时指出, 这可能是人类第一次在机电式计算设备中成功地使用了二进制的数制〔8〕。由于同事的鼓励, 斯蒂比茨不久之后便利用这些原理去研制一种能够计算复数的机电式计算机。该机于1939年投入使用。它使用斯蒂比茨发明的余3代码, 这种代码可以查出失效的继电器。这台复数计算机在设计时结合了电传打字机的功能, 因而可以从远处进行操作。在1940年夏季美国数学学会举行的一次会议上, 斯蒂比茨利用这些功能成功地做了远地操纵计算机的第一次表演。这次会议是在大特莫瑟大学的大礼堂里举行的。键盘和打印机放在大礼堂里, 而那台继电器计算机则放在 Bell Laboratories 的纽约实验室里, 它们之间由普通的电话线进行联系。与会者当中有一位当时还不出名的大学教授, 他就是 J. W. 莫克利〔9〕¹⁾。这次表演是成功的, 给人留下了深刻的印象。

第二次世界大战促使斯蒂比茨去研制一种继电器式的转发器和弹道计算机。阿伯丁试验基地利用这种继电器计算机去解决防空火力控制的弹道计算问题。后来, Bell Laboratories 考虑到电话部门对于高负荷使用条件下的可靠性有很高的要求, 因而生产了五台这样的继电器计算机。

H. 艾肯 (Howard Aiken) 1935 年在哈佛大学准备他的博士论文时, 生平第一次遇到了繁杂乏味的人工计算问题。艾肯敏锐地认识到, 高速自动计算机在满足当时物理学家日益增长的计算需要方面该会带来多大的益处。在获得了博士学位之后, 艾肯勾划了一台继电器计算机的详细设计蓝图。这台拟议中的计算机的控制信息来自 24 列的纸带, 数据来自 IBM 卡片; 六十个 10 刀拨盘开关用来确定常数。输出手段是穿孔卡片或电传打字机。1937 年艾肯首先向 Monroe 计算机公司提议研制这一机器, 但遭到拒绝。在几个同事的鼓动下, 艾肯去 IBM 公司访问 T. 沃森 (Thomas Watson) 及其副手, 使他们相信这种继电器计算机是切实可行的, 按照当时的工艺技术条件是做得成的。1939 年, IBM 公司和美国海军部联合向艾肯的研制工作提供财政支持。IBM 公司还派出四个工程师参加。

1944 年 4 月, IBM 公司的自动程控计算机在哈佛大学全面投入运转 (每天工作 24 小时, 每周工作 7 天)。该机又名 Mark I, 它一直工作到 1947 年秋天才停机, 以便迁往他处。这台机器与艾肯于 1937 年所提出的设计方案几乎完全相同, 它完全证明了艾肯的技术眼光和自信力。Mark I 的数据或者由穿孔卡片输入, 或者由手动开关设定。为了取得控制信息, 从 24 列的纸带上按照顺序将指令逐条读入机器。这种安排使指令和数据存放在完全分开的计算机存储设备里, 我们称之为哈佛机结构。艾肯利用 Mark I 发展起程序设计的某些最早期的设计思想。G. M. 霍珀 (Grace Murray Hopper) 后来将这些设计思想发展成第一个正式的编程语言结构, 并提出编译程序的设计思想。Mark I 旗开得胜, 艾肯又继续研制三种新机; 每种新机都借鉴前一种机器的有益经验。Mark II 也是继电器计算机, 但它比 Mark I 快得多, 也大得多。它在处理内部数据时是采用 10 位的十进制数。Mark III 使用了电子元件, 而且利用磁鼓来作为存储介质。图 1-1 是 1946 年拍的 Mark I 实物照片。

把 IBM 公司领进计算机这个新领域是艾肯个人的功劳, 但特罗普认为, 艾肯的最

1) Rosenberg, Jerry M., The Computer Prophets (Chap. 8), New York: Macmillan Publishing Co., Inc., 1969.