



微处理机和程序逻辑

[美] K. L. 肖 特 著

江 庚 和 主 译



华中工学院出版社

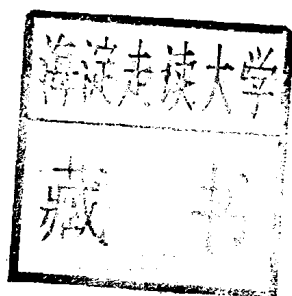


TP368.1
XT/1

微处理机和程序逻辑

[美] K. L. 肖特 著

江庚和 主译



华中工学院出版社

内 容 提 要

《微处理机和程序逻辑》系根据 K. L. 肖特所著《Microprocessors and Programmed Logic》一书的1981年版翻译的。

本书属于论述微处理机及其系统的专著。除讨论微处理机本身外,本书还讨论了与微处理机有关的各集成组件以及组成微处理机系统的硬件和软件设计。本书包括存贮器、微处理机的结构与操作、指令系统、程序汇编与模拟、堆栈与子程序、算术运算、程序控制 I/O、中断和 DMA、数字外围设备、模拟数据的输入和输出、微处理机系统设计等十二章。每章都附有内容丰富的习题。读者从本书获得的有关微处理机及其支援器件的知识可直接应用于工业生产和科学研究的许多实际设计中。

该书适合于从事微处理机系统的设计、生产和应用的广大工程技术人员以及大专院校有关专业的师生阅读。

JS276/14

微 处 理 机 和 程 序 逻 辑

[美] K. L. 肖 特 著

江 庚 和 主 译

责任编辑 殷伯明

*

华中工学院出版社出版
(武昌喻家山)

湖北省新华书店发行 各地新华书店经售

华中工学院印刷厂印刷

*

开本:787×1092 1/16 印张:23 字数:626,000

1983年12月第一版 1983年12月第一次印刷

印数:1—12,000

统一书号: 15255—014 定价: 2.50元

译 者 序

《微处理机和程序逻辑》一书，译自 K. L. 肖特所著的 1981 年版《Micro-processors and Programmed Logic》。

本书从应用的角度系统和全面地论述了微处理机组件以及微机理机系统的硬件和软件设计。无论从硬件观点或从软件观点来看，本书为教学或设计相应的微处理机系统提供了必要的和足够的基础知识，为解决实际问题提供了必需的设计技巧。本书系统完整，逻辑性强，层次分明，概念清晰。另外，为了牢固掌握各章内容、提高相应的设计技巧，每章还提供了相应的习题。

本书由江庚和负责翻译、审校。参加本书翻译的还有方树昌（第七、十一章）、归瑤琮（第四、十章及第八章的一部分）、瞿安连（第一、二章）、邱川弘（第三、六章）、张礼平（第十二章）。龚文浩参加了本书的校核。

本书适合于从事微处理机系统的设计、生产和应用的广大工程技术人员以及大专院校有关专业的师生阅读。

译 者

一九八三年七月

原 序

微处理机——集成于单块硅片上的中央处理机——的问世，使现代数字系统在设计方面取得了最重大的进展。微处理机的处理能力与经济性能，对于数字系统的设计方法及其应用产生了巨大的影响。

本书要讨论的就是微处理机，同时也将讨论与微处理机有关的集成电路和以微处理机为主要组成部分的系统的硬件和软件的设计。本书的目的，首先是使读者能透彻地理解设计以微处理机为主要组成部分的系统所必需的有关硬件和软件的基本概念，使读者深入了解各具体器件及其有关的实际问题，并进而掌握设计一个采用各器件的系统所必需的设计技巧。

本书的独到之处是以 8085A 微处理机为例来阐述基本概念。因为，用某一种微处理机作为教学的范例，使读者可以逐步深入地了解每一个实际器件的操作、特性及其局限性。

此外，用一种微处理机作为实例也便于采用一整套前后一致的符号和名称来阐述组成微处理机系统的许多逻辑部件的接口连接，使读者易于理解硬件接口的概念和了解各种大规模集成电路的工作功能。

8085A 是通用 8 位微处理机。选择这种微处理机作为教学范例的原因是它在工业中应用很广，而且在软件和硬件方面存在着有力的保证。这种保证体现在 8085A 的文件编制、使用须知以及研制软、硬件的辅助设备等各个方面。此外，还有一系列种类繁多的大规模集成电路外围设备与 8085A 兼容。因为 8085A 适用范围广，所以读者将会看到，他获得的有关 8085A 及其有关器件的知识可直接应用于工业生产的许多实际设计之中。

深入研究某一微处理机的另一优点是读者既可熟悉器件的特性，又可了解由这些特性所决定的、在实际应用中必然遇到的各器件的局限性。应当指出，读者适应了以某一特殊器件作为入门的教学方法，就会发现，他一旦熟悉了一种微处理机，并精通了它在数字系统设计中的应用，他就能比较容易地通过阅读厂家提供的用户手册和使用须知去了解其他各种微处理机的工作过程和使用方法。这是近年来在本书作者开设的微处理机课程中接受过这种教学方法的大学生和工厂业余学生的共同感受。

设计微处理机系统需要硬件和软件两方面的知识。我们假定读者已经学到了“门”和“触发器”等有关的数字硬件的基本知识。因为这些内容在任何一本关于数字系统设计的入门书籍中均可找到。虽然本书通过采用 8085A 的汇编语言来说明软件的概念，但并不要求事先懂得汇编语言程序设计，不过倒是希望读者对计算机高级语言程序设计具有一般的了解。

我们打算以基本的、系统的和集中的方式引出必要的硬件和软件的概念，因为这样比较合乎逻辑。显然，通过对任何一本教科书的学习都不可能使读者掌握

一门学科的全部知识，但是可以相信，本书将为你在设计微处理机系统方面的深造打下坚实的基础。

由于电气与电子工程师学会计算机协会典型课程委员会的建议^{*}，本书包括了该会推荐作为计算机科学与工程的全部课程的组成部分，即应在 DL-3 微处理机课程中讲授的各个课题。

最后，我要感谢我的妻子，她为本书的编写提供了许多帮助，并承担了大部分的打字工作。
谢后：

本书在编写过程中许多人做出了贡献。虽然不可能向他们一一致谢，但我想在此向 P. 贝克先生、D. 布乔先生、V. 马索西博士、C. 波维尔先生、R. 雷德先生和 D. 史密斯博士致以衷心的感谢。

我还要感谢许多学生，他们以本书的初稿作为学习材料，并提出了批评和建议，帮助我对本书进行修改。

还要特别感谢 L. 卡什曼，本书的编写有赖于他的编辑、组织和打印技能。

K. L. 肖特

本书是在作者多年从事微处理机系统设计的经验基础上编写而成的。在编写过程中，参考了大量的文献资料，并得到了许多人的帮助。本书力求做到概念清晰、重点突出、由浅入深、循序渐进。本书可作为高等院校计算机专业及相关专业的教材，也可供从事微处理机系统设计的工程技术人员参考。

本书共分八章。第一章介绍微处理机的基本概念；第二章介绍 8086 微处理机的内部结构及指令系统；第三章介绍 8086 微处理机的总线系统；第四章介绍 8086 微处理机的中断系统；第五章介绍 8086 微处理机的 I/O 系统；第六章介绍 8086 微处理机的系统总线；第七章介绍 8086 微处理机的应用；第八章介绍 8086 微处理机的测试与故障排除。

本书在编写过程中，参考了大量的文献资料，并得到了许多人的帮助。本书力求做到概念清晰、重点突出、由浅入深、循序渐进。本书可作为高等院校计算机专业及相关专业的教材，也可供从事微处理机系统设计的工程技术人员参考。

本书共分八章。第一章介绍微处理机的基本概念；第二章介绍 8086 微处理机的内部结构及指令系统；第三章介绍 8086 微处理机的总线系统；第四章介绍 8086 微处理机的中断系统；第五章介绍 8086 微处理机的 I/O 系统；第六章介绍 8086 微处理机的系统总线；第七章介绍 8086 微处理机的应用；第八章介绍 8086 微处理机的测试与故障排除。

^{*}“在计算机科学与工程会议报告中的课程表” IEEE 服务中心，445 Hoeo Lane, Piscataway, NJ 08854, 1977年1月。

目 录

第一章 绪论	(1)
1.1 大规模集成 (LSI) 对逻辑设计的影响.....	(1)
1.2 应用 LSI 的途径.....	(3)
1.2.1 定制的 LSI	
1.2.2 列入商品目录的 LSI	
1.2.3 可编程 LSI	
1.3 微处理机系统	(9)
1.4 应用领域	(11)
1.5 微处理机系统的设计	(13)
参考文献.....	(15)
参考资料.....	(15)
习题.....	(16)
第二章 随机存取半导体存储器	(18)
2.1 存储器基本概念	(18)
2.1.1 触发器/1位寄存器	
2.1.2 m 位寄存器	
2.1.3 存储器	
2.2 存储器的分类	(25)
2.3 半导体 RAM 器件的结构	(26)
2.4 存储器器件的定时和操作	(30)
2.5 存储器系统的构成	(33)
2.6 存储器的接口	(35)
2.7 存储器的制造工艺	(36)
2.8 ROM	(37)
2.8.1 MOS ROM	
2.8.2 双极型 ROM	
2.9 PROM、EPROM 和 EAROM.....	(39)
2.9.1 可熔断熔丝的 PROM	
2.9.2 悬浮栅 EPROM	

2.9.3 PROM 编程器	
2.9.4 EAROM	
2.10 读/写存储器	(43)
2.10.1 双极型 RWM	
2.10.2 静态 MOS RWM	
2.10.3 动态 MOS RWM	
参考文献	(44)
参考资料	(45)
习题	(45)
第三章 微处理机的结构与操作	(47)
3.1 基本微处理机概念	(47)
3.1.1 控制单元	
3.1.2 内部寄存器	
3.1.3 算术和逻辑单元	
3.1.4 微处理机的状态	
3.2 LSI 微处理机的实现	(54)
3.3 8085A 微处理机	(55)
3.3.1 8085A 的结构	
3.3.2 定时和时序	
3.3.3 存储器和 I/O 同步——等待状态	
3.3.4 停机状态	
3.3.5 合电源复位和手动复位	
3.4 最小结构的 8085A 微处理机	(65)
3.5 8080A 微处理机和 CPU 组	(66)
3.6 8085A 与 8080A 的相似之点	(70)
参考文献	(71)
参考资料	(71)
习题	(72)
第四章 数据传送、逻辑操作及转移	(73)
4.1 指令系统	(73)
4.2 数据传送指令	(75)
4.2.1 数据传至和传自 I/O 设备	
4.2.2 附加的数据传至和传自存储器	

4.2.3 附加寻址方式的形成	
4.2.4 在微处理机内的数据传送	
4.3 逻辑操作	(82)
4.4 流程图	(86)
4.5 分支指令	(88)
4.6 循环程序	(91)
4.7 软件延迟	(93)
参考文献.....	(96)
参考资料.....	(96)
习题.....	(96)
第五章 程序汇编与模拟	(98)
5.1 程序设计语言	(98)
5.2 软件研制	(100)
5.3 汇编程序源程序	(103)
5.4 程序的手工汇编	(104)
5.5 汇编程序控制译码指令——伪指令	(106)
5.6 两次扫描汇编程序	(108)
5.7 宏指令	(112)
5.8 模拟程序	(115)
5.9 微处理机研制系统	(117)
参考文献.....	(119)
参考资料.....	(120)
习题.....	(120)
第六章 堆栈与子程序	(122)
6.1 堆栈与堆栈的操作	(122)
6.2 子程序	(125)
6.3 条件调用和返回	(128)
6.4 传递参数	(129)
6.5 子程序文件编制	(131)
6.6 模块程序结构	(132)
6.7 子程序的模拟	(133)
参考资料.....	(134)

习题	(134)
第七章 算术运算	(136)
7.1 无符号的二进制整数	(136)
7.1.1 加法	
7.1.2 减法	
7.1.3 逻辑移位	
7.1.4 乘法	
7.1.5 除法	
7.2 二进制补数	(145)
7.2.1 二进制补数的加法	
7.2.2 二进制补数的减法	
7.2.3 算术移位	
7.2.4 二进制补数的乘法和除法	
7.3 二-十进制数	(153)
7.3.1 无符号的二-十进制数的加法	
7.3.2 十进制补数和二-十进制数的减法	
7.3.3 二-十进制数的乘法和除法	
7.3.4 二-十进制与二进制的转换	
7.4 小数	(159)
7.5 浮点数	(159)
7.5.1 浮点的格式	
7.5.2 浮点运算程序	
参考文献	(165)
参考资料	(166)
习题	(166)
第八章 程序控制 I/O	(168)
8.1 引言	(168)
8.2 专用 I/O	(169)
8.3 存储器对应 I/O	(174)
8.4 MSI I/O 端口	(175)
8.5 程序控制的信息并行传输	(178)
8.6 可编程 LSI 端口	(181)
8.6.1 包含可编程 I/O 端口的多功能装置	
8.6.2 外设接口设备	

8.7 信息的串行传送	(190)
8.7.1 异步串行字符传送	
8.7.2 同步串行字符传送	
8.8 微处理机直接串行 I/O 引线	(195)
8.9 电特性	(196)
8.9.1 总线驱动器和接收器	
8.9.2 端接	
8.9.3 标准	
参考文献	(198)
参考资料	(198)
习题	(199)
第九章 中断和 DMA	(201)
9.1 中断	(201)
9.2 8085A 中断结构	(203)
9.3 优先级中断结构	(208)
9.4 FIFO 缓冲器	(212)
9.5 实时时钟与间隔时间	(214)
9.5.1 8155的定时器	
9.6 使用中断应考虑的因素	(218)
9.6.1 共用子程序	
9.6.2 禁止中断	
9.6.3 优先级分配	
9.7 直接存储器访问方式 (DMA)	(219)
参考文献	(224)
参考资料	(224)
习题	(225)
第十章 数字外围设备	(227)
10.1 人工数据输入	(227)
10.1.1 机械开关	
10.1.2 大键盘与小键盘	
10.2 显示器	(234)
10.2.1 LED 显示器	
10.2.2 CRT 显示	
10.3 打印机	(241)

10.4 大容量存贮系统.....	(243)
10.4.1 纸带	
10.4.2 盒式磁带和盒式存贮器	
10.4.3 软磁盘	
10.5 算术功能的硬件实现.....	(255)
10.5.1 有限功能的高速算术 IC	
10.5.2 计算器 IC	
10.5.3 算术处理单元——APU	
参考文献.....	(260)
参考资料.....	(260)
习题.....	(261)
第十一章 模拟数据的输入和输出.....	(263)
11.1 模拟数据.....	(263)
11.2 运算放大器.....	(265)
11.2.1 理想运算放大器	
11.2.2 通用运算放大器	
11.2.3 差动和测量放大器	
11.2.4 延迟与建立时间	
11.3 数字/模拟转换器——DAC	(272)
11.3.1 输入代码	
11.3.2 权电阻 D/A 转换器	
11.3.3 R-2R 梯形 D/A 转换器	
11.3.4 $2^n R$ D/A 转换器	
11.3.5 参考电压	
11.3.6 乘法式 D/A 转换器	
11.4 D/A转换器的技术特性.....	(279)
11.5 微处理机与D/A转换器的接口.....	(280)
11.6 模拟/数字转换器	(283)
11.6.1 比较器	
11.6.2 计数式 A/D 转换器	
11.6.3 逐次逼近式 A/D 转换器	
11.6.4 电压/频率转换器	
11.6.5 脉冲宽度转换器	
11.6.6 双斜率积分式 A/D 转换器	
11.7 采样和保持电路.....	(291)

11.8 模拟多路转换器和多路信号分离器·····	(292)
11.9 多信道数据采集系统·····	(294)
参考文献·····	(295)
参考资料·····	(296)
习题·····	(296)
第十二章 微处理机系统设计 ·····	(298)
12.1 技术要求阐明·····	(299)
12.1.1 前后关系的分析	
12.1.2 功能的详细说明	
12.1.3 设计约束	
12.2 系统设计·····	(300)
12.2.1 划分	
12.2.2 总实现方式的选择	
12.2.3 微处理机的类型	
12.2.4 微处理机选择	
12.3 系统实现·····	(307)
12.3.1 硬件实现	
12.3.2 软件实现	
12.4 测试与调试·····	(309)
12.4.1 硬件测试和调试	
12.4.2 软件测试和调试	
12.4.3 系统的测试和调试	
12.5 文件编制和维护·····	(312)
12.6 多微处理机系统·····	(313)
12.6.1 主/从分布式系统	
12.6.2 多主机分布式系统	
参考文献·····	(320)
参考资料·····	(320)
习题·····	(321)
附录A 开路集电极与三态输出·····	(322)
附录B 八进制和十六进制数·····	(326)
附录C Intel 8085A指令系统·····	(327)
附录D ASCII 字符表·····	(350)
附录E 虚地分析·····	(351)

第一章 绪 论

在一个到几个大规模集成(LSI)芯片上实现数据流和控制的 LSI 微处理机这项技术进步,正在改变着逻辑设计的习惯做法。现在,逻辑设计人员使用的结构部件是数据流、控制存储器、读/写存储器等集成电路芯片,他的工作就是安装这些芯片和为控制存储器编程。

G. G. 兰登

1.1 大规模集成 (LSI) 对逻辑设计的影响

逻辑设计的基本目的是设计一个能发挥预期功能的,并且使用可靠、维护方便、费用低廉的系统。一般说来,设计是否简单是达到这些目的的关键。过于复杂的设计也许能达到第一个要求,但在其他方面是肯定要落空的。

当然,在实际设计中经济效果是极为重要的因素。一般说,最经济有效的设计往往是那些最简单的设计。正因为简单,所以才更可靠、更容易维护。

就整个系统一级来说,一项设计的总成本包括研制、元件、工具、组装、测试、系统的修理维护以及库存备用零件的费用^[1,2]。总成本是与系统中元件的个数成正比的,而一般元件数是与门和触发器的个数成正比的。为了减少元件个数,使系统成本降至最低,在数字系统设计中使用了传统的开关理论来尽可能地减少门和触发器的个数,如图1.1—1(a)和1.1—1(b)所示。这种最小化技术最初是为了用继电器构成门和触发器的系统而发展起来的。随着技术的进步,先是真空管,然后是分立元件的固态器件代替了继电器。由于元件个数仍然正比于门和触发器的个数,因此,在真空管或分立元件的固态器件所构成的设计中,开关理论的设计技术对于降低系统成本仍然是有效的。

然而,在二十世纪六十年代初期,以**小规模集成** (Small Scale Integration) SSI制成了小规模集成电路,它把多达 12 个门集成在单块硅片上,并且作为单个的多引线元件封装。使用了**集成电路 IC**,系统的元件个数就不再正比于门和触发器的个数了,而是直接等于系统中集成电路的组件数。从而,只要降低所需的集成电路组件的总数,就能降低系统的成本。虽然系统设计人员这时仍然可以运用上述的开关理论技术,通过减少所需的门或触发器来化简电路,但是只有在这种化简能同时减少集成电路组件个数时,才能达到节约的目的。

随着集成电路工艺的进一步发展,以**中规模集成** (Medium Scale Integration) MSI制成了逻辑复杂程度达到每块芯片上有 13 到 99 个等效门的集成电路。尔后,在降低系统成本方面,对传统的开关理论设计技术的效果进行了仔细地研究分析,以**大规模集成** (Large Scale Integration) LSI 制成了逻辑复杂程度达到每块芯片上有 100 个以上等效门的集成电路。用 MSI 和 LSI 电路的制造工艺,把 100 个或 100 个以上的门制造在一块芯片上的

费用同制造10个门一样便宜，这就完全打破了门-触发器/元件个数的比例关系。所以，现在对那些用集成电路构成的系统来说，其总成本是与集成电路组件的个数成正比的*。

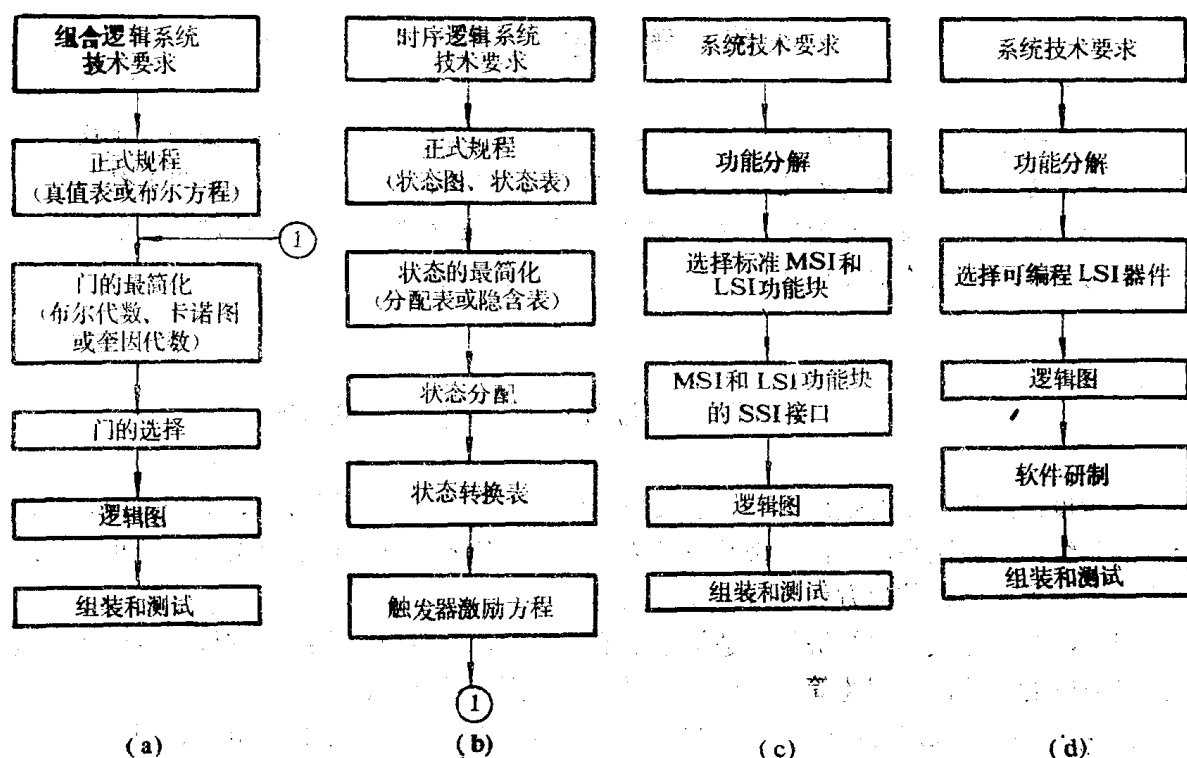


图1.1-1 数字系统的设计步骤

(a)使用门和触发器的组合逻辑设计；(b)使用门和触发器的时序逻辑设计；
(c)使用标准 MSI 和 LSI 电路的设计；(d)使用可编程 LSI 电路的设计

由于 LSI 电路在每块组件上制造了更多的逻辑电路，并且所需的功率消耗更少，所以大量应用 LSI 的结果使系统的各项成本得以降低。此外，对集成电路来说，最普遍的故障发生在通向芯片的引线连接处。而以少量的 LSI 组件来代替大量的 SSI 和 MSI 组件，减少了引线数目，从而能达到更高的可靠性。

从集成电路生产的经济角度来看，要求把需用大量门的完整的功能块制造在单块芯片上^[3]。然而，制造集成电路所要求的芯片面积是限制功能的复杂程度的主要原因，因为该面积一定不能超过生产的高成品率所能容许的范围。**成品率 (yield)** 是生产过程中得到合格集成电路的百分率。为了限制芯片面积，引到芯片上以提供输入、输出数据和控制信号的外部连线的数目也必须受到限制。

当大、中规模集成的功能组件以大批量生产时，生产成本就可以降低，因此它们的价格远远低于 SSI 电路的等效功能组件。所使用的标准 MSI 和 LSI 功能组件中，虽然可能包括一些在某个具体应用中用不上的门，但是通常仍比在“门”的水平上实现最小化逻辑的设计更为经济，见图1.1-1(c)所示。

●在应用微处理机的 LSI 系统中，这种比例关系存在着例外的情况，因为在某些应用中，软件的研制费用可能比硬件在系统的总成本里占有更大的比重。

由于 LSI 器件生产量的多少影响它的生产成本, 因此生产量是提供 LSI 器件时主要考虑的问题。我们把产量和成本之间的关系以经验曲线的形式画了下来^[2] (见图 1.1—2)。经验曲线 (learning curve) 的关系表明, 生产量每增加一倍, 单位生产成本就以一常数百分比下降。这是因为: 随着生产人员经过实际生产的体验, 对器件变得更加熟悉, 生产技能有了提高的缘故。但是, 在降到某一点之后, 即使生产经验再增加, 也只能使生产成本有极微小的下降。对于包括 LSI 在内的集成电路, 这个常数百分比 (即图 1.1—2 中的常数 K) 大致为 30%^[4]。因为标准集成电路的售价遵循着同样一条曲线, 因此这条经验曲线对于系统设计人员是十分重要的。

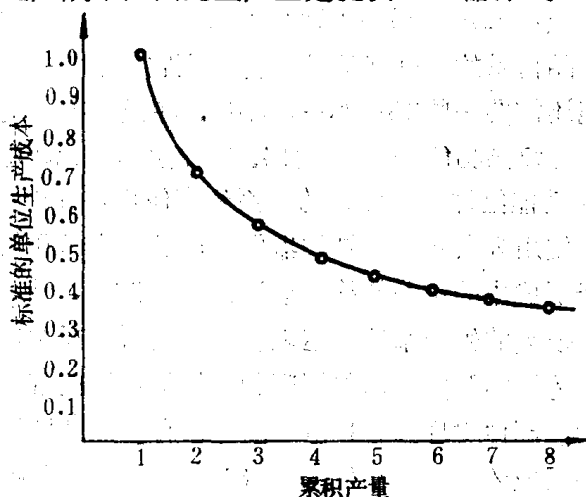


图1.1—2 半导体经验曲线

$$C_F = C_I(1 - K)^{\log \frac{P_T}{P_I}} \quad C_F: \text{最终生产成本} \quad C_I: \text{最初生产成本}$$

K: 常数 P_T : 累积产量 P_I : 最初产量

1.2 应用 LSI 的途径

由于集成电路内部的结构具备复杂的功能, 而且没有一种有效的综合方法适合于使用 LSI 和 MSI 电路的设计, 所以大都以系统设计取代逻辑设计。这里, 之所以缺乏有效的综合方法, 是由于所实现的 LSI 结构一般不能用数学方法来描述^[6]。因此, 当使用 MSI 和 LSI 电路时, 设计的方法实际上大部分是探索性的。需要详细分析的是子系统之间的逻辑连接, 而不是门和触发器之间的逻辑连接。

1.2.1 定制的 LSI

应用 LSI 的第一个途径是制造一种或多种定制的电路, 并让它在特定的系统中提供所要求的功能。使用定制的 LSI 电路, 必然带来研制费用的增加, 这些费用必须分摊在所需的器件数目中。对于那些不是大批量生产的系统来说, 为设计定制的 LSI 电路的研制费用往往过于浩大。但对于那些大批量生产的系统, 这种方法经常是可行的。在使用定制的 LSI 时, 因为缺乏竞争, 经验曲线所预示的那种由大批量生产带来生产成本降低的利益, 一般没有被集成电路制造厂家以降低售价的形式转移给系统设计人员。

定制 LSI 的一个主要问题是修改设计所需的费用。例如, 为了改进一个系统而作的修改, 可能需要把全部 LSI 电路的设计过程重做一遍。存在的另一个问题是, 设计和制造定制的 LSI 电路必须具有专业知识和专用设备。一般说来, 需要把实际的设计和制造工作承包给某一个半导体制造厂家, 这样做就可能导致研制时间的大大加长。最后一个问题是若想得到定制的 LSI 电路的第二来源 (即使有这个可能性) 往往也是十分困难的。这里所谓的第二来源, 就是有另外的生产厂家制造同一种器件。

1.2.2 列入商品目录的LSI

应用 LSI 的另一种途径可以避免定制的 LSI 所存在的问题,那就是选择现成的或已列入商品目录的 LSI 子系统,并且把它们连接在一起(将列入商品目录的 LSI 电路连接起来可能还需要一些SSI和MSI电路)。在设计中,列入商品目录的 LSI 电路的用法类似于可提供中等复杂功能的 MSI 电路。象加法器、比较器、计数器、移位寄存器以及编码变换器等 MSI 功能组件都可以作为子系统来使用,并以探索方法把它们连接起来。

使用列入商品目录的器件,产品的竞争和第二来源迫使集成电路制造厂家把按经验曲线降低成本的利益转移给设计人员。但是,为了提供低成本的器件,集成电路生产厂家必须有很大的生产基数,这就要求他们所选择生产的那些功能组件能被系统设计人员广泛接受。为了需要照顾那些用门和触发器设计的现存系统,如果说集成电路生产厂家对于把哪些功能部件实现为 SSI和MSI 电路有一点儿踌躇不决的话,那么如何选择复杂适度的功能部件集成为列入商品目录的LSI就成了更大的难题。鉴于 LSI 的研制费用远远高于 SSI 和 MSI,因而 LSI 生产厂家所冒的风险也就大得多。可是对具备一定复杂程度的功能组件的需求是毫无疑问的,于是乎象**通用异步收发器**(Universal Asynchronous Receiver Transmitter) UAR/T 那样的 LSI 子系统在事实上就成了工业的标准产品^[8]。

对 UAR/T 的简单分析可以给我们这样一个概念:即使在不甚复杂的 LSI 子系统中,逻辑也是相当复杂的。UAR/T是一个数字通讯子系统,它能接收串行数据并转为并行输出,和/或接收并行数据转为串行输出。更进一步,它能以**全双工法**(full duplex)操作,即接收和发送两者可同时进行。串行发送或接收的**字符**(character)包括一位起始位,五位到八位数据位,一位到两位停止位以及一位或是奇校验或是偶校验的奇偶校验位,而最后一位也可以没有。为了适应多方面的应用,位速率,数据位数,奇偶校验模式以及停止位的位数都可以从外部进行选择。整个 UAR/T 集成在单块芯片上,并封装在一个40根引线的**双列直插式组件**(Dual In-Line Package) DIP 中。UAR/T广泛用于外围设备的接口,例如电传打字机与数字系统的接口。作为列入商品目录的 LSI 之所以有这样的地位,这是同有好几个生产厂家作为产品的第二供货来源密切相关的。

1.2.3 可编程LSI

由于 LSI 子系统具有程度更高的灵活性,所以使用起来是极其便利的。设计人员(而不是半导体生产厂家)能使它们满足各种应用的要求,这种器件必将得到更加广泛的应用。它的灵活性是通过硬件编程来达到的。

集成电路的**硬件编程**(hardware programming)的概念并不是新鲜东西。在很多MSI 电路的常规应用中,我们常对它们进行编程,只不过没有这样来考虑问题罢了。如 74192 那样的可预置计数器就具有预置初始计数状态(编程)的控制输入端,以及控制计数方向(加或减)的控制输入端(见图1.2—1)。这些输入端的数值可以根据具体应用,利用成组开关或锁存器进行编程,这样就能动态地改变计数器的功能。

UAR/T 也是可编程的,编程输入端置入各个参数,这些参数决定每个字的位数,并决定是要奇校验还是要偶校验,或者根本不要奇偶校验,同时也决定停止位的位数。

然而,可编程LSI 这个名词通常专指那些具有非常高级编程能力的 LSI 器件。如只读存储器(ROM),可编程逻辑阵列(PLA),以及微处理机(μ P)等等,这些 LSI 器件的