

微机系统 原理与接口

朱庆保 张颖超 孙 燕 编著



南京大学出版社

微机系统原理与接口

朱庆保 张颖超 孙 燕 编著

南京大学出版社

内 容 简 介

本书以 8086 至奔腾 4 微处理器为核心的微型计算机作为主讲机型,以较完整的体系反映了当代微机系统的原理和接口技术,并体现了现代技术在微机中的应用。为拓宽读者视野和提高读者实际工作能力,本书注意了理论联系实际,注意了原理和应用相结合,突出了实用技术。作为教科书,注意了合理的知识结构,力争内容全面,基本概念清楚,使学生掌握较宽厚的基础知识和较系统的整机概念。

全书共分 14 章,内容包括微机系统的基本构成及基本工作原理、微处理器、指令系统、汇编语言程序设计、存储系统、输入输出、中断系统、DMA 系统、并行接口芯片及应用、可编程定时器计数器、串行通信、现代总线技术、奔腾系列微处理器、A/D、D/A 转换及其接口。

本书可作为大、专院校计算机专业和非计算机专业的微机原理或接口技术等课程的教科书,也可作为各类微机学习者的参考用书。

图书在版编目(CIP)数据

微机系统原理与接口/朱庆保编著. —南京:南京大学出版社,2003.8

ISBN 7-305-04102-5

I. 微... II. 朱... III. ①微型计算机-理论-高等学校-教材②微型计算机-接口设备-高等学校-教材 IV. TP36

中国版本图书馆 CIP 数据核字(2003)第 051071 号

书 名 微机系统原理与接口
编 著 者 朱庆保 张颖超 孙 燕
出版发行 南京大学出版社
社 址 南京市汉口路 22 号 邮编 210093
电 话 025-3596923 025-3592317 传真 025-3303347
网 址 <http://press.nju.edu.cn>
电子邮件 nupress1@public1.ptt.js.cn
经 销 全国各地新华书店
印 刷 常熟市华顺印刷有限公司
开 本 787×1092 1/16 印张 27.75 字数 724 千
版 次 2003 年 8 月第 1 版 2003 年 8 月第 1 次印刷
印 数 1-5000
ISBN 7-305-04102-5/TP·269
定 价 40.00 元

* 版权所有,侵权必究

* 凡购买南大版图书,如有印装质量问题,请与所购
图书销售部门联系调换

前 言

随着集成电路和超大规模集成电路的迅速发展,微处理器和有关半导体器件的集成度成倍翻番,价格迅速下降,导致了不断把过去用于大中型计算机的现代技术下移到微型机中来。相应地,微机总线宽度不断翻番,总线体系结构不断改进和发展。这种日新月异的飞速发展,使微机应用几乎扩展到人类社会的所有领域,促进了各个专业技术的进步和发展,并正在急剧地改变着社会生产和社会生活面貌。学好微机,成为各行各业和大中专院校学生的共识。而从微机系统原理与接口方面学好微机,是从高层次应用好微机和充分利用、开发微机软硬件资源的关键。

我们在微机原理与接口技术的教学中深深感到,要使学生走上社会即能服务于社会,很快就能熟练地应用微机并能进行项目开发,就必须理论联系实际,学以致用,所学知识能较系统地反映当代微机的技术特征及应用技术,为此我们编写了《微机系统原理与接口》一书。本书以 8086 至奔腾 4 微处理器为核心的微型计算机作为主讲体系,力图使学生较全面系统地建立起整机概念,并能学习或掌握现代微机中采用现代技术的思想方法,了解最新动态和最新技术。为了达到理论联系实际,本书尽可能地突出原理与应用相结合,突出学生基本技术技能的培养,包括硬件分析、设计和调试能力。从另一个方面,作为教科书,必须有合理的知识结构,力争内容全面,系统完整,基本概念清楚,使学生掌握较扎实的基础知识和较系统的整机概念。从便于教和学及适应各种层次读者的需要出发,在表达上力求深入浅出,通俗易懂,体系安排则尽可能由浅入深。

Intel 公司推出 8088/8086 微处理器以来,经历了 20 多年的时间,已从上世纪 80 年代的 8088/8086 微处理器发展到今天的奔腾 4,已有许多过去用于大中小型机的现代技术下移到奔腾系列微处理器中来,使其性能发生了巨大的变化。然而,Intel 公司为了兼顾老用户的利益,新的微处理器型号总是兼容了老型号的指令系统,其硬件体系结构也是在老型号基础上的逐步改进和发展。因此,从 8086 到奔腾 4 微处理器有共同的基本原理,都兼容了 8086 的指令功能,8086 仍然是奔腾系列微处理器的基础。因此,本书从体系安排上,充分考虑了便于教学或自学,力求适合不同层次的教学要求,从基础入手,由浅入深,并加强了现代技术的比重。第 1 至 10 章为系统原理及接口技术,这十章以基本原理和基本接口技术为基础,延伸或扩展到现代技术、现代微机中的原理及其应用。其中第 1 章建立起微型计算机的多种基本概念和整机概念及基本工作原理以作为基础。第 2 至 10 章以 PC 系列微机为主线,讲述了有关原理和相关的接口技术。这一体系反映了微机系统的微处理器、指令系统、汇编语言程序设计、存储器技术、输入输出、中断系统、DMA 系统、并行接口芯片、定时器/计数器。第 11 章至第 14 章为扩展与提高部分。第 11 章为串行通信的有关内容及 8251 的编程方法,第 12 章重点描述了现代微机中的总线技术,第 13 章以较大篇幅介绍了奔腾系列微处理器及采用的现代技术,第 14 章重点介绍了 A/D 与 D/A 的工作原理及有关接口技术。

由于不同的学校开设的前期课程差异较大,对本课程安排的课时数差异也较大,因此,为

了兼顾不同学校和专业不同的教学要求,以及兼顾到系统的完整性,本书内容较为系统全面。因此,有些章节可根据不同的教学要求选学。

本书由朱庆保主编,张颖超、孙燕副主编。其中第1、3、4、5、13章由朱庆保写作;第6、7、8、9、14章由张颖超写作;第2、10、11、12章由孙燕写作。

由于编者水平有限,书中难免有错误和不妥之处,恳请同行专家不吝指教及广大读者批评指正。

编 者

2003年4月于南京

目 录

第 1 章 微型计算机基础	1
1.1 概述	1
1.1.1 微型机发展概述	1
1.1.2 微型机的分类与应用	2
1.2 微型计算机基础	3
1.2.1 微型计算机最基本的硬件结构	3
1.2.2 指令系统与程序	4
1.2.3 微型计算机的组成与基本工作过程	5
1.2.4 计算机系统硬件与软件	14
1.3 多媒体计算机的组成	15
1.3.1 多媒体计算机概述	15
1.3.2 多媒体计算机系统构成	15
练习与思考	25
第 2 章 微处理器 8086/8088	27
2.1 8086/8088 微处理器的特点	27
2.2 8086/8088 微处理器结构	27
2.2.1 8086/8088 功能结构	28
2.2.2 8088 寄存器结构	30
2.2.3 8086/8088 存储器组织	32
2.2.4 8086/8088 外部结构与引脚	33
2.3 8086 外总线和总线时序	35
2.3.1 8086 最小工作组态	36
2.3.2 最小工作组态典型时序	37
2.3.3 8086 最大工作组态	41
2.3.4 最大工作组态典型时序	46
2.4 微型计算机系统框图	48
练习与思考	50
第 3 章 80x86 系列微处理器的指令系统	52
3.1 概述	52
3.2 寻址方式	53
3.3 8088/8086 指令系统	55
3.3.1 数据传送指令	55
3.3.2 算术运算指令	62

3.3.3 逻辑运算和移位指令	68
3.3.4 控制转移指令	72
3.3.5 循环指令	75
3.3.6 子程序调用与返回指令(SUBROUTINE)	75
3.3.7 中断指令	77
3.3.8 处理机控制指令	78
3.3.9 串操作指令	79
3.4 32位微处理器指令系统	81
3.4.1 32位微处理器的寄存器结构	81
3.4.2 32位微处理器指令系统特点	82
练习与思考	83
第4章 汇编语言程序设计	87
4.1 汇编语言概述	87
4.2 汇编语言程序语句行的构成	88
4.2.1 标记	88
4.2.2 表达式	89
4.2.3 语句	92
4.3 常用伪指令	92
4.3.1 表达式赋值伪指令 EQU	93
4.3.2 数据定义及存储器分配伪指令	93
4.3.3 分析运算符与综合运算符	95
4.3.4 段定义伪指令	98
4.3.5 定义子程序的伪指令 PROC、ENDP、NEAR 和 FAR	101
4.4 DOS 功能调用	103
4.5 程序设计及其设计举例	104
4.5.1 分支程序设计	105
4.5.2 循环程序设计	106
4.5.3 子程序结构的程序设计	108
4.6 键盘 I/O 与显示 I/O	114
4.6.1 键盘 I/O	114
4.6.2 DOS 显示功能调用	119
练习与思考	120
第5章 存储器技术	125
5.1 半导体存储器分类与主要技术指标	125
5.1.1 只读存储器 ROM 分类与特点	125
5.1.2 随机读写存取器 RAM 的分类	126
5.1.3 半导体存储器的主要技术指标	127
5.2 随机读写存储器 RAM 的结构	128
5.2.1 基本存储电路	128
5.2.2 RAM 的结构	129

5.3 RAM 与总线的基本连接方法	134
5.3.1 位扩展法	134
5.3.2 字扩展法	134
5.3.3 字位同时扩展法	135
5.4 存储器的工作时序	136
5.4.1 存储器对读周期的时序要求	136
5.4.2 存储器对写周期的时序要求	137
5.5 现代微机中的新型 RAM 及并行主存系统	137
5.5.1 现代 RAM 存储器采用的两种基本技术	137
5.5.2 并行主存系统	138
5.5.3 现代 RAM	141
5.6 现代微机存储体系	144
5.6.1 Cache-主存存储层次	144
5.6.2 主-辅存存储层次	148
练习与思考	150
第6章 输入输出的接口技术	152
6.1 I/O 接口	153
6.1.1 接口概说	153
6.1.2 接口的功能及其作用	153
6.1.3 接口的分类	155
6.2 I/O 端口及其编址与译码方法	156
6.2.1 I/O 端口	156
6.2.2 I/O 端口的编址方法	157
6.3 输入输出方式和接口原理	159
6.3.1 程序控制传输方式	159
6.3.2 中断传输方式	164
6.3.3 DMA 方式	166
6.3.4 四种 I/O 方式的比较	167
6.4 I/O 接口卡的设计	168
6.4.1 I/O 接口卡设计应注意的问题	168
6.4.2 ISA 槽的引线信号安排和 ISA 接口卡的几何尺寸	169
6.4.3 I/O 端口的译码方法	169
6.4.4 总线接口	173
6.4.5 离散信号输出接口技术	173
6.4.6 接口驱动程序分析与设计	175
练习与思考	176
第7章 DMA 控制器 Intel8237A	177
7.1 DMA 传输技术概述	177
7.1.1 DMA 过程	177
7.1.2 DMA 的数据传送方式	178

7. 1. 3 DMA 的特点	179
7. 1. 4 DMA 接口的基本模型	180
7. 1. 5 DMA 应用	180
7. 2 DMA 控制器 Intel 8237A	181
7. 2. 1 8237A 的基本功能	181
7. 2. 2 8237A 的传输类型	181
7. 2. 3 8237A 的传输对象	182
7. 2. 4 8237A 的操作时序	182
7. 2. 5 Intel 8237A 的基本结构及引脚功能	185
7. 2. 6 8237A 的工作方式	191
7. 2. 7 8237A 的控制字和编程	192
7. 2. 8 8237A 的编程举例	197
练习与思考	199
第 8 章 中断技术	200
8. 1 中断的基本概念及其应用	200
8. 1. 1 中断的定义和有关名词解释	200
8. 1. 2 中断的典型应用	200
8. 2 中断系统的功能和中断优先级	202
8. 2. 1 中断系统应具有的基本功能	202
8. 2. 2 中断优先级	202
8. 3 8086/8088 的中断操作	207
8. 3. 1 8086/8088 的中断分类	207
8. 3. 2 中断类型码、中断向量与中断向量表	208
8. 4 中断响应过程与时序	211
8. 4. 1 硬件中断的响应过程和中断服务	211
8. 4. 2 8086/8088 中断响应的流程	213
8. 4. 3 硬件中断的时序	214
8. 4. 4 软件中断的响应过程	214
8. 5 中断控制器 Intel 8259A	216
8. 5. 1 8259A 的主要功能及结构	216
8. 5. 2 8259A 的工作过程	220
8. 5. 3 8259A 的中断管理	222
8. 5. 4 8259A 的编程	227
8. 5. 5 8259A 的应用举例	235
练习与思考	243
第 9 章 可编程计数器/定时器	244
9. 1 8253 的主要功能	245
9. 2 8253 的引脚功能和编程结构	245
9. 2. 1 8253 的引脚功能	245
9. 2. 2 8253 的编程结构	246

9. 2. 3 8253 内部寄存器的选择	247
9. 3 8253 的编程	248
9. 3. 1 8253 的控制字	248
9. 3. 2 8253 的计数初值	249
9. 3. 3 8253 的初始化编程	249
9. 3. 4 8253 的计数器读操作	251
9. 4 8253 的工作方式	252
9. 4. 1 方式 0(计数结束产生中断)	252
9. 4. 2 方式 1(可重复触发的单稳态触发器)	254
9. 4. 3 方式 2(速率发生器)	256
9. 4. 4 方式 3(方波发生器)	257
9. 4. 5 方式 4(软件触发选通信号发生器)	260
9. 4. 6 方式 5(硬件触发选通信号发生器)	261
9. 5 六种工作方式的比较	263
9. 6 8253 的综合应用举例	264
9. 6. 1 用计数器测量频率的原理	264
9. 6. 2 用 8253 实现频率测量	264
9. 7 8254 芯片	267
9. 7. 1 8254 的状态寄存器和状态锁存器	267
9. 7. 2 8254 的锁存命令和读回命令	268
练习与思考	270
第 10 章 并行接口及应用	271
10. 1 并行接口概述	271
10. 2 可编程并行接口 8255	272
10. 2. 1 8255 基本功能	272
10. 2. 2 8255 内部结构与引脚	272
10. 2. 3 8255 控制字和工作方式	275
10. 3 并行接口应用举例	281
10. 3. 1 一般并行接口	281
10. 3. 2 可编程 8255 并行接口应用	282
10. 3. 3 计算机的打印机接口	284
练习与思考	287
第 11 章 串行通信及接口电路	289
11. 1 串行通信基础	289
11. 1. 1 并行通信与串行通信概述	289
11. 1. 2 串行通信的数据传送方式	290
11. 1. 3 串行传输基本模型	290
11. 1. 4 调制与解调	291
11. 1. 5 串/并和并/串转换的基本原理	293
11. 1. 6 异步通信与同步通信	293

11. 1. 7 串行通信接口	297
11. 2 可编程串行接口 8251A	300
11. 2. 1 8251A 基本功能	300
11. 2. 2 8251A 内部结构与引脚	300
11. 2. 3 8251A 内部寄存器	305
11. 3 串行接口应用举例	309
11. 3. 1 一般并串转换电路	309
11. 3. 2 可编程 8251A 同步串行接口应用	310
11. 3. 3 PC 系统双机通信	312
练习与思考	315
第 12 章 总线技术	317
12. 1 概述	317
12. 1. 1 总线分类	317
12. 1. 2 总线组成	318
12. 1. 3 总线基本功能	319
12. 1. 4 总线标准	320
12. 1. 5 总线配置结构	321
12. 1. 6 总线数据传输	322
12. 1. 7 总线裁决方式	323
12. 1. 8 局部总线技术	324
12. 2 PC/XT 的 ISA 总线	325
12. 2. 1 ISA 特点	325
12. 2. 2 总线结构与引脚定义	325
12. 3 PCI 总线	327
12. 3. 1 PCI 总线特点	327
12. 3. 2 PCI 总线结构与引脚定义	329
12. 3. 3 PCI 总线周期	331
12. 4 USB 串行总线	335
12. 4. 1 特点	335
12. 4. 2 USB 总线结构	335
12. 4. 3 USB 总线体系结构	337
12. 4. 4 USB 协议帧	338
12. 4. 5 USB 协议工作流程	341
练习与思考	344
第 13 章 奔腾系列微处理器	345
13. 1 从 8086 到奔腾系列微处理器发展概述	345
13. 2 流水线与超标量流水线技术	346
13. 2. 1 指令重叠操作方式	347
13. 2. 2 标量流水工作原理	348
13. 2. 3 流水线中的相关问题	348

13. 2. 4	程序转移对流水线的影响	349
13. 2. 5	超级标量流水方法	349
13. 2. 6	超长指令字(VLIW)技术	350
13. 2. 7	80386 和 80486 微处理器中的指令流水线	350
13. 3	精简指令系统计算机 RISC	351
13. 3. 1	CISC 的含义	351
13. 3. 2	RISC 的含义	351
13. 3. 3	RISC 的起源及其发展历史	352
13. 3. 4	RISC 发展与应用	353
13. 4	奔腾系列微处理器的主要工作方式	353
13. 4. 1	实方式	353
13. 4. 2	保护方式	354
13. 4. 3	保护方式下的虚拟 8086 模式	368
13. 5	奔腾微处理器	369
13. 5. 1	奔腾微处理器的基本结构	369
13. 5. 2	奔腾微处理器采用的超级标量流水线	372
13. 5. 3	奔腾微处理器片内高速缓存	375
13. 6	高能奔腾至奔腾 4 采用的几种现代技术	379
13. 6. 1	双重独立总线技术	379
13. 6. 2	单指令多数据处理	380
13. 6. 3	SSE 技术简介	381
13. 6. 4	动态执行技术	381
13. 7	从高能奔腾至奔腾 4 处理器的主要特色	385
13. 7. 1	高能奔腾(Pentium Pro)处理器的特色	385
13. 7. 2	多能奔腾(MMX 奔腾)处理器的主要特色	386
13. 7. 3	奔腾 2 处理器的主要特色	387
13. 7. 4	奔腾 3 处理器的主要特色	388
13. 7. 5	赛扬处理器	389
13. 7. 6	便携式处理器	389
13. 7. 7	奔腾 4 处理器的主要特色	390
练习与思考		391
第 14 章 A/D、D/A 转换技术及其接口设计		392
14. 1	数/模(D/A)转换器及其与主机的接口	393
14. 1. 1	D/A 转换器的基本组成	393
14. 1. 2	D/A 转换原理	395
14. 1. 3	D/A 转换器的主要技术指标及选用要点	397
14. 1. 4	D/A 转换器 DAC0832	399
14. 1. 5	使用 D/A 转换器时应注意的问题	402
14. 1. 6	串行接口 D/A 转换器	404
14. 1. 7	D/A 转换器应用举例	405

14. 2 模/数(A/D)转换及其与主机的接口	407
14. 2. 1 A/D 转换接口	407
14. 2. 2 A/D 转换原理	413
14. 2. 3 借助计算机的软硬件资源实现 A/D 转换	416
14. 2. 4 A/D 转换器的主要技术指标	417
14. 2. 5 A/D 转换器的选择原则	418
14. 2. 6 标度变换	419
14. 2. 7 A/D 转换器 ADC0809	419
14. 2. 8 12 位 A/D 转换器 AD574	424
练习与思考	427
主要参考文献	429

第 1 章 微型计算机基础

一个实际的微型计算机的电路结构是相当复杂的,对于初学者来说,从一个实际的高档微机了解它的构成和程序的执行过程会感到不知所措而事倍功半。因此,本章先从简单的微机入手,介绍微型计算机的组成与工作过程,建立指令系统、时序、指令执行过程、I/O 接口、总线结构、软件和硬件及整机等方面的基本概念,为后续各章奠定基础。此外,本章还介绍了多媒体个人计算机的构成、各部件的用途及其主要的性能评价方法。

1.1 概 述

电子计算机通常按体积、性能(诸如运算速度、字长、存储容量及输入输出能力等)和价格分为巨型机、大型机、中型机、小型机和微型机。这种划分并没有一个一成不变的严格标准,由于计算机性能在日新月异地变化,划分标准也在不断改变。例如,上世纪 70 年代巨型机的标准是速度超过每秒运算 1000 万次(10MIPS, Million Instructions Per Second),存储容量超过 1000 兆(外存),价格 1000 万美元以上,但到了 80 年代,巨型机的速度超过每秒运算 1 亿次(100MIPS)以上。尤其是随着超大规模集成电路(VLSI)的出现和发展,芯片集成度不断提高,器件价格不断下降,使得过去在大、中、小型机中才采用的技术(诸如流水线技术、高速缓冲存储器技术、虚拟存储器管理技术、多处理机并行处理技术及精简指令集系统 RISC 等),下移到微型机系统中来,因而使大、中、小、微型计算机的分界面不断发生变化,界限随时代的发展而趋向消失。因此,从系统结构和基本工作原理上说,微型机与大、中、小型计算机并没有本质上的区别。其主要不同点是:从结构上看,微型计算机的中央处理器(CPU)是集成在一小块硅片上的,而大中型计算机的 CPU 可以由相当多的电路或集成电路组成。为了区别于大、中、小型计算机的 CPU,称微型计算机的 CPU 芯片为微处理器 MPU(Micro-Processing Unit 或 Microprocessor)。通常在微型计算机中直接用 CPU 表示微处理器。从用途上看,大、中、小型计算机主要是面向多用户多任务系统设计并能从事复杂的科学计算。而微型机主要是面向控制对象或主要是面向单人使用的个人计算机,其功能和性能虽然没有大、中、小型机强大,但具有结构简单、灵活、可靠性高(由于元件少,器件数目少)、体积小、重量轻、价格低及应用面广等特点。显见,虽然本书以微型机为范例和主线,但介绍的一些基本原理也适于大、中、小型机。

1.1.1 微型机发展概述

1971 年美国 Intel 公司生产出第一个 4 位的微处理器 4004 并宣布微型计算机 MCS-4 研制成功。接着 8 位处理器 8008 又被开发出来。这些芯片的集成度约在 2000 器件/片,时钟频率为 1MHz,平均指令执行时间约 20 μ s。这就是第一代微处理器。1973~1977 年,许多处理器厂推出了第二代微处理器,其代表产品是 Intel 公司 8080/8085, Zilog 公司的 Z80, Motorola 公司

的 6800/6802, Rockwell 公司的 6502。这些处理器字长为 8 位,集成度 5000 器件/片 ~ 10000 器件/片,时钟频率为 2M ~ 4MHz,指令周期 1 ~ 2 μ s。1978 ~ 1979 年,一些厂家推出了性能可与过去中档小型计算机相比的 16 位处理器,具代表性的芯片是 Intel 公司的 8086/8088, Zilog 公司的 Z8000 以及 Motorola 公司的 M68000。这些处理器的集成度 20000 器件/片 ~ 60000 器件/片,时钟频率为 4M ~ 8MHz,指令周期 0.5 μ s。人们将这些微处理器称为第一代超大规模集成电路的微处理器或第三代微处理器。1983 年,Intel 公司又推出了高性能的 16 位微处理器 80186 和 80286。1985 年 32 位的微处理器 80386 及 M68020 推出市场,集成度达 45 万个晶体管/片,时钟频率达 40MHz,1989 年推出了 80486,1993 年 Pentium 问世,其集成度高达 310 万个晶体管/片,性能之高,足以同以前的高档小型机的 CPU 相匹敌。近年,Intel 公司又推出了性能更高的 Pentium 4 处理器,它几乎使 32 位机的技术发展到了顶峰,相信用于个人计算机的 64 位的处理器在不远的将来就会问世。

1981 年以来,IBM 公司先后推出了采用以 8088 CPU 为核心的个人计算机(Personal Computer),即 IBM PC 和 IBM PC/XT,以后相继推出了以 80286 和 80386CPU 为核心的 PC/AT 机。目前以 Pentium 3 处理器为核心的微型机也显得落后了,而大量应市的是基于 Pentium 4 处理器的超级微型计算机。

1.1.2 微型机的分类与应用

微型计算机由微处理器、存储器和配套的输入输出接口以及系统总线等组成。用户可以按不同的应用要求,配置相应的外部设备和系统软件以构成一个完整的计算机系统。

如果按所用微处理器的字长划分,可分为 4、8、16、32 和 64 位机(关于字长见 1.2 节)。从微型计算机的结构形式划分,可分为单片机、单板机和多板机(或个人计算机)。到了上世纪 90 年代,局域网、广域网、城际网以及 Internet 迅速普及与发展,微型计算机从功能上又分为网络工作站(客户端—Client)和网络服务器(Server)两大类型。网络客户端一般采用个人计算机(Personal Computer)。

单片微型计算机简称单片机。是把微型计算机的主要部件——微处理器、一定容量的存储器及输入输出接口电路都集成在一块芯片上的单芯片式微型计算机。

单片机的应用可分为单机应用和多机应用。单机应用主要广泛应用在智能产品、智能仪表、简单的测控系统、经济型数控机床和智能接口等等。多机应用是单片机在高科技领域中的主要应用模式,这种多机系统,单片机作为面向对象的控制核心,若干个单片机和系统主机组成集散型或分布式控制系统。例如,在机器人的计算机多机控制系统中,机器人的感觉系统、姿态控制系统、遥控系统和行走系统都可以分别由一片单片机组成的应用系统来承担。再用一片单片机来完成它们之间的协调管理。这样五片单片机就可以构成一个机器人的计算机简易控制系统。

单板微型计算机简称单板机。是将微处理器、一定容量的存储器(包括 ROM 和 RAM)、输入输出接口、辅助电路和简单的输入输出设备(如键盘和数码显示器等)通过总线全部安装在一块印刷电路板上的单板式微型计算机。单板机由于结构简单,价格便宜,目前主要用于实验室,其它用途类似于单片机。

多板机是将 CPU 模块、存储器模块和输入输出接口模块等插到带电源的机箱底板上,通过底板上的系统总线互相连接而组成的微型计算机。其典型代表是 STD 总线的工业控制机。应用最为广泛的 PC 系列微机也属于多板结构。

PC(Personal Computer) 机指那些小型化较为便宜并且是为单人使用或每次只能有一个人使用而设计的“个人”计算机。PC 这一称呼最早用来专指 IBM 公司推出的第一批以 8088 CPU 为核心的“个人”计算机,现人们常把基于 86 系列的微机称为 PC 系列微机。

目前,微型计算机的应用范围几乎涉及人类社会的所有领域:从国民经济各部门到个人家庭生活,从军事部门到民用部门,从科学教育到文化艺术,从生产领域到消费娱乐,无一不是微机应用的天下,简要的归纳为如下几个方面:

- (1) 科学计算:随着微机性能提高,现在在各个学科中,也常用微机进行科学计算。
- (2) 自动控制:目前在各个领域,自动控制进入了以微机为控制核心的新阶段。
- (3) 测量和测试:在这一领域,用微机对测试设备进行控制并进行数据采集和处理。
- (4) 信息处理:包括图像、声音、文字等各类信息处理。如微机网络通讯和管理等。
- (5) 教育和卫生:如多媒体微机辅助教学、各种先进的微机医疗设备和专家系统等。
- (6) 人工智能:人工智能又称智能模拟,就是要使计算机能模仿人的高级思维活动。
- (7) 计算机辅助设计与计算机辅助制造及用于家用电器等等。

1.2 微型计算机基础

一个实际的微型计算机的电路结构是相当复杂的,对于初学者来说,从一个实际的高档微机了解它的构成和程序的执行过程会感到不知所措而事倍功半。因此,本节从一个最简单的微机构成和工作过程入手,建立指令系统、时序、I/O 接口、软件和硬件等基本概念。

1.2.1 微型计算机最基本的硬件结构

电子数字计算机一开始是作为一个计算工具出现的。不难想像,计算机如能脱离人的干预而自动地完成计算,它必须具备以下几个基本功能:① 要有能进行运算的部件——运算器;② 要有能记忆原始数据、运行程序及运算结果的部件——存储器;③ 要有能发出各种控制信息,以便使计算机各部件协调工作的部件——控制器;④ 要有能将原始数据及运算程序输入计算机,并将结果及其它信息输出的部件——输入和输出设备。计算机的硬件就是由这五大部分构成,结构示意图如图 1-1 所示(图中 I/O 接口未画出)。这种模型最早是由数学家冯·诺依曼提出的,故称为冯·诺依曼结构。

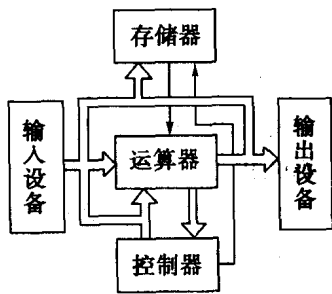


图 1-1 计算机的基本组成

运算器称为 ALU(Arithmetic Logical Unit),它由逻辑部件及逻辑电路组成,其功能是进行算术和逻辑运算。控制器发出各种控制信号,使整个运行过程自动进行。控制器和运算器合称为中央处理单元(Central Processing Unit),简称 CPU。CPU 是一片大规模集成电路,是微机的核心。在 CPU 内一般有多个寄存器,用以在数据运算和数据传输过程中临时存储数据等。其中包含一个累加器(Accumulator 简称 A,16 位机简称 AX),累加器是一个具有特种功能的寄存器,它虽叫累加器,但并不能在其中进行加法运算,其主要作用是用来传输、临时存储 ALU 运算过程的结果和其它数据,并能把存在其中的数据左移或右移。存储器由记忆单元组成,用于存放数据、中间结果及一系列指令。输入输出设备称外部设备,简称外设,用于输入原始数据、控制命令及输出运行结果等。

值得指出的是,大中型计算机的 CPU 可以由相当多的电路或集成电路组成。而微型计算机的中央处理器(CPU)是集成在一小块硅片上的,因而称微型计算机的 CPU 芯片为微处理器 MPU,但通常也直接用 CPU 表示微型机中的微处理器。

在计算机中,有两股信息在流动:一股是数据流。各种原始数据由输入设备输入至运算器,再存入存储器中。在运算过程中,数据从存储器读入运算器进行运算,运算的中间结果存入存储器中或由外设输出。另一股信息流为控制信息流,人们给计算机的各种命令(即程序),也以数据的形式由存储器送入控制器,由控制器译码后发出控制信号,控制计算机各部件的工作。

1.2.2 指令系统与程序

1. 指令系统

计算机之所以能脱离人的干预自动完成各种运算和输入输出等,是靠执行人们预先编制的各种命令,这些命令通常称为指令。CPU 研制单位研制一种新型 CPU 时,设计出若干条指令,并根据这些指令设计出 CPU 的硬件电路,使得 CPU 能惟一地识别并执行这些指令。某类处理器所有指令的集合就是该类处理器的指令系统(Instruction System)。显见,指令系统的性能如何,决定了处理器的基本功能。因此,指令系统的设计是处理器设计的一个核心问题。从计算机组成的角度看,指令是软件和硬件的接口。计算机设计者根据指令系统的要求设计硬件,软件工作者根据指令系统提供的功能编制程序。

在目前的微机中,通常一条指令实现一种基本功能(超长指令微机例外),每一种功能包含一个或多个基本操作。例如:

ADD A, # 13H; 把立即数 13H 与累加器 A 中的内容相加(H 表示 16 进制),结果放在 A 中这是 MCS-51 型单片微机中的一条加法指令。这种指令和对应的功能是由 MCS-51 单片机的研制商提供的。因为计算机只认识二进制数码,所以计算机指令系统中的指令,都必须以二进制编码的形式存放到计算机的主存储器中,才能由微处理器一条一条读出来加以执行。这种指令的二进制编码通常称为指令的机器码(Machine Code)。有的指令较短,用一个字节即可表示,有的指令较长,用两个字节或多个字节表示。ADD A, # 13H 的机器码为 00100100 00010011B(B 表示二进制),是两字节指令。执行这条指令时,微机从主存中读出第一个字节 24H,经译码电路译码就知道这是一条把紧跟在该字节后的数 13H(称立即数)与 A 中的内容相加,结果放 A 中,故紧接着执行读第二字节 13H 并与 A 中内容相加的操作。可见,该指令有两个基本操作,取操作码译码和取操作数并相加。24H 表示计算机完成什么操作,通常称为操作码,13H 是参加操作的数的本身,称操作数(本例中第二字节如为操作数所在的地址也称操作数)。

计算机通过执行指令来处理各种数据,为了指出各种数据的来源、操作结果的去向及所执行的操作,一条指令必须包括下列信息:

(1) 操作码,它具体说明了操作的性质及功能。一台计算机可能有几十条至几百条指令,每一条指令都有一个惟一的操作码,计算机通过识别该操作码来完成不同操作。

(2) 操作数的地址。CPU 通过该地址就可以取得所需的操作数。

(3) 操作结果的存储地址。把对操作数的处理结果保存在该地址中,以便再次使用。

(4) 下一条指令的地址。通常情况下,当程序顺序执行时,下条指令的地址用一个程序计