

新编微机原理及接口技术

New Microcomputer Principle and Interface Technology

耿仁义 主编

新编微机原理及接口技术

主 编:耿仁义

副主编:张 隆

编 者:王 炜 陈红艳
包乃兰 袁臣虎



天津大学出版社
TIANJIN UNIVERSITY PRESS

内容简介

本书根据我国高等院校普遍开设的“微机原理及接口技术”课程的基本教学要求而编写。

本书主要分为两大部分:原理部分和应用技术部分。原理部分主要包括微型计算机的基本知识、8086/8088 微处理器的基本结构、中断系统、指令系统及基本的汇编语言程序设计方法和编程技巧。应用技术部分主要包括 MCS-51 单片机的内部结构、指令系统及程序设计、内部定时器/计数器、存储器和接口的扩展技术、串行通信接口的工作原理及应用技术、D/A 与 A/D 的工作原理和应用技术等。

本书内容丰富、侧重应用、通俗易懂、便于自学,可作为高等院校本科生、专科生的教材,也可作为硕士研究生入学考试、计算机等级考试的辅导书,还可供科技工作者和工程技术人员参考。

图书在版编目(CIP)数据

新编微机原理及接口技术/耿仁义主编. —天津:天津大学出版社,2006.9

ISBN 7-5618-2338-X

I.新... II.耿... III.①微型计算机—理论—高等学校—教材 ②微型计算机—接口—高等学校—教材
IV.TP36

中国版本图书馆 CIP 数据核字(2006)第 096932 号

出版发行 天津大学出版社
出 版 人 杨欢
地 址 天津市卫津路 92 号天津大学内(邮编:300072)
电 话 发行部:022-27403647 邮购部:022-27402742
网 址 www.tjup.com
短信网址 发送“天大”至 916088
印 刷 廊坊市长虹印刷有限责任公司
经 销 全国各地新华书店
开 本 185mm × 260mm
印 张 26.25
字 数 656 千
版 次 2006 年 9 月第 1 版
印 次 2006 年 9 月第 1 次
印 数 1-5 000
定 价 38.00 元

前 言

酝酿已久的《新编微机原理及接口技术》一书终于出版了。本书凝聚着所有参编人员多年辛勤工作的心血。

随着计算机技术的发展及应用的日趋广泛,许多新技术不断涌现,微型计算机不论结构还是性能都发生了翻天覆地的变化。从早期的 8 位机发展到现今的 Pentium 机和 64 位机,其工作原理越来越复杂,功能越来越强大,但提供给用户的界面却越来越简单。以 MCS-51 单片机为主流的单片机、微型机如雨后春笋般出现,并以体积小、功能全、性价比高等诸多优点,而在工业控制、智能化仪器仪表、通信设备、家用电器等领域中被广泛应用。

随着嵌入式系统、片上系统等概念的提出,信息时代、数码时代的来临,各种数字、数码设备如潮水般不断涌现。单片机的发展及应用又进入了一个新的历史阶段。为了适应大型程序的设计需要,结构化的 C、C++ 等程序设计语言被移植到单片机上,使单片机的应用如虎添翼。所以,仅以 8086 微处理器或单纯以 MCS-51 单片机为背景的微机原理课程的教学内容已不能适应科技发展的需要。尤其随着微机应用的操作平台已由 DOS 向 Windows 转变,对微机的管理技术也进入了一个新的阶段。描述符、虚拟存储器、多任务等新的管理技术等已成为计算机发展历程中新的里程碑,掌握新操作系统平台的接口软件的编写也是微机接口设计者应具备的能力。微机的应用正在从接口技术的应用逐渐向总线技术的应用过渡,所以,各种总线技术相继出现,例如当前正在广泛应用的 PCI 总线、 Γ C 总线以及应用最广泛的串行通信标准总线 RS-232-C、RS-422/RS-485 总线、USB 总线和 IEEE1394 总线等。总线的应用将是计算机应用的主要内容之一。

目前,“微机原理及接口技术”课程的教材已经很多,如百花齐放,各有所长。但是,随着微机技术的发展,许多教材已不能满足人才培养的需要。该类教材的编写应逐渐淡化繁冗的原理论述,强化提高学生的实际应用技能,从而使学生能够学得懂、用得上。本书在内容选取上进行了必要的改革,主要包括两大部分,即原理部分和应用技术部分,尤其侧重于应用技术部分。

本书原理部分以 8086 微处理器为背景机型,并要求读者掌握一些微机的基础知识和基本工作原理,从而为学习 80286~Pentium 机奠定基础。应用技术部分以 MCS-51 单片机为背景机型。MCS-51 单片机是 20 世纪 80 年代最早引入我国的单片机之一,应用技术较为成熟,教学资料也较为丰富,至今仍是单片机的主流机型。除了介绍 MCS-51 单片机的内部结构及基本的工作原理之外,本部分还重点介绍了在实际应用中经常遇到的一些应用技术和技巧,例如 MCS-51 单片机的复位方式、掉电保护、解决程序“飞跑”的指令冗余技术和软件陷阱技术、存储器和接口的扩展技术、“看门狗”和 Γ C 总线应用技术等。

参加本教材编写的人员都是多年工作在教学、科研第一线,并有丰富的教学和实践经验的老师。本书的出版是大家集体智慧的结晶,也是学校领导和相关朋友大力支持的结果。在编写过程中,编者得到了天津工业大学信息与通信工程学院苗长云院长、王金海副院长的精心指导,并得到学校教务处教材科的大力支持,在此表示衷心感谢。

全书由张隆组织编写,耿仁义负责全书的框架设计、内容修改和最终定稿。具体分工是:

耿仁义编写第 1、2、3、4、5、6、12、13 章;包乃兰编写第 7、8 章;陈红艳编写第 9、10、11 章;王伟编写第 14、15 章;张隆编写第 16、17 章;袁臣虎编写 MCS-51 单片机的应用实例(附录 D),并且做了大量的编辑工作。冯志红老师为教材的编写提供了许多帮助,并做了大量的编辑工作,在此表示由衷的感谢。天津大学出版社的李叩存老师、韩振平老师和本书责任编辑周小明老师对本书的出版给予了大力支持,在此表示由衷的谢意。

由于编者水平有限,时间仓促,难免有疏漏之处,殷切希望广大同仁和读者提出批评指正。

编者

2006 年 7 月于天津工业大学

目 录

第1章 基础知识	(1)
1.1 微型计算机系统的组成	(1)
1.2 微型计算机主机结构	(1)
1.2.1 CPU	(2)
1.2.2 内存储器	(2)
1.2.3 总线	(2)
1.2.4 地址译码电路	(3)
1.2.5 接口	(3)
1.3 计算机中数的表示方法	(3)
1.3.1 定点表示	(3)
1.3.2 浮点表示	(4)
1.4 计算机中常见的编码	(4)
1.4.1 ASCII 码	(4)
1.4.2 8421BCD 编码	(4)
1.4.3 BCD 码的运算	(5)
1.5 计算机中的运算电路及补码运算	(6)
1.5.1 计算机中的运算电路	(6)
1.5.2 补码运算	(7)
练习与思考题	(8)
第2章 存储器	(9)
2.1 半导体存储器的分类	(9)
2.1.1 静态存储器(SRAM)	(9)
2.1.2 动态随机存储器(DRAM)	(11)
2.1.3 只读存储器(ROM)	(12)
2.1.4 闪速存储器(Flash Memory)	(17)
2.2 存储器的扩展技术	(20)
2.2.1 位扩展	(20)
2.2.2 字扩展	(20)
2.2.3 字位全扩展	(21)
练习与思考题	(21)
第3章 8086/8088 微处理器及系统	(22)
3.1 8086 微处理器结构	(22)
3.1.1 8086/8088 寄存器	(22)
3.1.2 8086 微处理器引脚功能	(25)
3.2 8086 CPU 的时序	(27)

3.3	8086 CPU 的工作模式	(28)
3.3.1	最小工作模式	(28)
3.3.2	最大工作模式	(29)
3.4	8086 存储器系统	(29)
3.4.1	8086 存储体结构	(29)
3.4.2	8086/8088 存储器系统的地址译码	(30)
3.5	8086/8088 的 I/O 端口地址译码	(32)
3.6	8086/8088 指令系统	(33)
3.6.1	8086/8088 汇编语言指令语句格式	(33)
3.6.2	8086/8088 的寻址方式	(34)
3.6.3	通用数据传送指令 MOV	(36)
3.6.4	目标地址传送指令	(38)
3.6.5	标志位传送指令	(39)
3.6.6	输入/输出数据传送指令 IN、OUT	(39)
3.6.7	算术运算类指令	(40)
3.6.8	逻辑运算和移位指令	(44)
3.6.9	控制类指令	(47)
3.6.10	串操作指令	(50)
3.7	程序设计	(54)
3.7.1	简单程序设计	(54)
3.7.2	分支程序设计	(55)
3.7.3	循环程序设计	(56)
3.7.4	子程序设计	(58)
3.8	汇编语言程序设计	(58)
	练习与思考题	(61)
第 4 章	中断系统	(63)
4.1	输入/输出方式	(63)
4.1.1	无条件传送方式	(63)
4.1.2	查询式传送方式(又称有条件传送)	(64)
4.1.3	中断传送方式	(64)
4.1.4	DMA 传送方式	(64)
4.2	中断概述	(65)
4.2.1	中断的定义和作用	(65)
4.2.2	中断响应条件	(65)
4.2.3	中断嵌套	(65)
4.3	8086/8088 中断系统	(66)
4.3.1	中断矢量和中断矢量表	(66)
4.3.2	内部中断	(68)
4.4	外部中断	(69)

4.4.1	非屏蔽中断 NMI	(70)
4.4.2	可屏蔽中断 INTR	(70)
4.5	中断矢量表的初始化	(70)
4.5.1	常用的 DOS 系统功能调用	(71)
4.5.2	常用的 BIOS 中断调用	(74)
4.6	可编程序中断控制器 8259A	(78)
4.6.1	8259A 的引脚与功能结构	(78)
4.6.2	8259A 内部结构框图和中断工作过程	(79)
4.6.3	8259A 中断嵌套控制方式	(81)
4.6.4	控制字格式	(82)
4.6.5	8259A 应用举例	(87)
	练习与思考题	(87)
第 5 章	输入/输出及接口应用技术	(89)
5.1	可编程并行接口芯片 Intel 8255A	(89)
5.1.1	8255A 的内部结构和引脚功能	(89)
5.1.2	8255A 的工作方式及工作方式控制字	(90)
5.1.3	8255A 的 PC 口的位控方式	(91)
5.1.4	8255A 的联络信号功能	(92)
5.1.5	8255A 的状态字	(93)
5.2	可编程定时器/计数器接口芯片 8253A	(96)
5.2.1	8253A 芯片内部结构	(96)
5.2.2	8253A 芯片的工作方式	(99)
5.2.3	8253A 应用举例	(102)
5.3	串行通信	(104)
5.3.1	异步通信	(104)
5.3.2	同步通信	(105)
5.3.3	波特率和接收/发送时钟	(106)
5.3.4	通用异步接收/发送器(UART)硬件结构	(106)
5.4	串行异步通信接口芯片 8250	(108)
5.4.1	8250 芯片引脚功能	(108)
5.4.2	8250 芯片的内部结构和寻址方式	(109)
5.4.3	8250 通信编程	(113)
5.4.4	8250 应用举例	(116)
	练习与思考题	(117)
第 6 章	从 80286 到 Pentium	(119)
6.1	80286 微处理器	(119)
6.2	描述符与描述符表	(119)
6.2.1	80286 寄存器组	(119)
6.2.2	描述符	(120)

6.2.3	段选择字与虚拟地址指针	(121)
6.2.4	描述符表	(122)
6.3	虚拟存储器管理	(124)
6.3.1	80386/80486 微处理器	(124)
6.3.2	虚拟地址与物理地址及地址重定位	(124)
6.3.3	虚拟存储器	(126)
6.3.4	段页式管理机制	(126)
6.4	80386/80486 多任务及任务转换	(127)
6.4.1	任务结构	(128)
6.4.2	任务寄存器 TR(TSSR)	(130)
6.4.3	任务门	(130)
6.4.4	任务的转换	(130)
6.5	Pentium 机的结构及特点	(131)
6.5.1	Pentium 机的内部结构	(132)
6.5.2	Pentium 机的寄存器结构	(134)
6.5.3	Pentium 处理器的中断系统	(138)
	练习与思考题	(140)
第 7 章	MCS-51 单片机内部结构	(141)
7.1	MCS-51 单片机的内部结构	(141)
7.1.1	MCS-51 单片机 CPU 的结构	(142)
7.1.2	MCS-51 单片机内部存储器结构	(144)
7.1.3	MCS-51 单片机并行 I/O 接口	(147)
7.1.4	MCS-51 单片机串行接口和定时器/计数器	(147)
7.1.5	中断系统	(147)
7.2	MCS-51 单片机时序	(148)
7.2.1	几个基本概念	(148)
7.2.2	MCS-51 单片机指令的取指/执行时序	(148)
	练习与思考题	(149)
第 8 章	MCS-51 单片机指令系统	(150)
8.1	指令系统概述	(150)
8.1.1	指令格式	(150)
8.1.2	指令系统分类	(151)
8.1.3	指令中的常用符号	(151)
8.2	指令的寻址方式	(151)
8.2.1	立即寻址	(151)
8.2.2	直接寻址	(151)
8.2.3	寄存器寻址	(152)
8.2.4	寄存器间接寻址方式	(152)
8.2.5	位寻址方式	(152)

8.3 数据传送类指令	(153)
8.3.1 内部数据传送指令(MOV)	(153)
8.3.2 外部数据存储器 RAM(或 I/O 接口)的字节传送指令(MOVX)	(155)
8.3.3 ROM 的字节传送指令(MOVC)	(156)
8.3.4 数据交换指令	(157)
8.3.5 堆栈操作指令(PUSH 和 POP)	(158)
8.4 算术运算和逻辑运算类指令	(158)
8.4.1 算术运算指令	(158)
8.4.2 逻辑运算类指令	(163)
8.5 移位和位操作类指令	(165)
8.5.1 移位指令	(165)
8.5.2 位操作类指令(也称布尔操作指令)	(166)
8.6 跳转及控制类指令	(167)
8.6.1 无条件转移指令	(168)
8.6.2 条件转移指令	(169)
8.6.3 子程序调用指令和返回指令	(172)
8.6.4 空操作指令	(173)
练习与思考题	(173)
第 9 章 汇编语言及程序设计	(176)
9.1 汇编语言的基本概念	(176)
9.1.1 程序设计语言	(176)
9.1.2 汇编语言格式	(177)
9.1.3 伪指令	(178)
9.2 汇编语言的程序设计	(181)
9.2.1 顺序结构程序设计	(182)
9.2.2 分支结构程序设计	(184)
9.2.3 循环结构程序设计	(188)
9.2.4 查表结构程序设计	(191)
9.2.5 子程序结构程序设计	(193)
练习与思考题	(195)
第 10 章 MCS-51 的中断系统	(197)
10.1 MCS-51 的中断源和中断标志	(197)
10.1.1 中断源	(197)
10.1.2 中断标志	(198)
10.2 MCS-51 对中断请求的控制	(199)
10.2.1 中断允许的控制	(199)
10.2.2 对中断优先级的控制	(199)
10.3 MCS-51 对中断的响应	(200)
10.3.1 中断响应的条件	(200)

10.3.2	中断响应过程	(201)
10.3.3	中断响应时间	(201)
10.3.4	中断请求的撤除	(201)
10.3.5	MCS-51 中断系统的初始化	(203)
10.4	中断程序应用举例	(203)
10.4.1	主程序	(203)
10.4.2	中断服务程序	(203)
	练习与思考题	(205)
第 11 章	MCS-51 内部定时器/计数器	(206)
11.1	定时器/计数器的内部结构及工作原理	(206)
11.2	定时器/计数器的控制	(207)
11.2.1	工作方式寄存器 TMOD(89H)	(207)
11.2.2	控制寄存器 TCON(88H)	(207)
11.3	定时器/计数器的工作方式	(208)
11.3.1	工作方式 0	(208)
11.3.2	工作方式 1	(208)
11.3.3	工作方式 2	(208)
11.3.4	工作方式 3	(209)
11.4	定时器/计数器的应用举例	(209)
	练习与思考题	(211)
第 12 章	MCS-51 单片机外部存储器的扩展技术	(213)
12.1	MCS-51 单片机程序存储器的扩展	(213)
12.1.1	构造总线	(213)
12.1.2	外部程序存储器的操作时序	(214)
12.1.3	程序存储器与 MCS-51 单片机的连接	(215)
12.1.4	地址译码电路与地址译码方式	(215)
12.1.5	程序存储器 EPROM 与 MCS-51 单片机连接举例	(216)
12.2	MCS-51 单片机数据存储器的扩展	(217)
12.2.1	MOVX @Ri,A 及 MOVX A,@Ri 指令	(218)
12.2.2	MOVX @DPTR,A 及 MOVX A,@Ri 指令	(218)
12.3	串行 EEPROM 与 MCS-51 单片机的连接	(218)
	练习与思考题	(219)
第 13 章	MCS-51 单片机并行接口及扩展技术	(221)
13.1	MCS-51 内部并行 I/O 端口	(221)
13.1.1	MCS-51 内部 I/O 端口结构	(221)
13.1.2	I/O 端口的读—修改—写操作	(222)
13.1.3	各端口的特点	(222)
13.1.4	I/O 端口的应用举例	(223)
13.2	MCS-51 单片机系统并行接口的扩展技术	(224)

13.3 可编程并行接口芯片 Intel 8155A/8156A	(226)
13.3.1 8155A 内部结构及引脚功能	(227)
13.3.2 8155A 的工作方式及工作方式命令字	(228)
13.3.3 8155A 内部定时器/计数器	(229)
13.3.4 8155A 的应用举例	(231)
13.4 键盘与显示技术	(231)
13.4.1 BCD 拨盘接口技术	(231)
13.4.2 七段数码显示器(LED)接口技术	(233)
13.4.3 键盘接口技术	(236)
13.5 液晶显示(LCD)接口技术	(241)
13.5.1 LCD 的基本工作原理	(242)
13.5.2 LCD 的驱动方式	(242)
练习与思考题	(247)
第 14 章 MCS-51 单片机串行通信接口	(249)
14.1 MCS-51 单片机串行口结构	(249)
14.2 串行口的工作方式	(250)
14.3 串行口工作方式设置	(251)
14.4 串行口各种工作方式的工作过程及应用举例	(253)
14.4.1 方式 0	(253)
14.4.2 方式 1	(254)
14.4.3 方式 2 和方式 3	(257)
14.4.4 主从式多机通信	(260)
练习与思考题	(266)
第 15 章 D/A 和 A/D 转换技术	(268)
15.1 D/A 转换与 D/A 转换接口	(268)
15.1.1 D/A 转换器(DAC)的原理	(268)
15.1.2 D/A 转换器的性能指标	(269)
15.1.3 DAC0832	(270)
15.1.4 DAC0832 与 MCS-51 单片机连接应用举例	(271)
15.2 A/D 转换与 A/D 转换接口	(274)
15.2.1 A/D 转换器(ADC)的原理	(274)
15.2.2 A/D 转换器的性能指标	(276)
15.2.3 ADC0809	(276)
15.2.4 双积分式 A/D 转换器 5G14433	(280)
15.2.5 LM331 系列 V/F 转换器	(284)
练习与思考题	(287)
第 16 章 MCS-51 单片机的其他应用技术	(289)
16.1 MCS-51 单片机的复位方式	(289)
16.2 MCS-51 单片机的节电工作方式	(290)

16.3	MCS-51 单片机的编程和校验方式	(293)
16.3.1	编程	(293)
16.3.2	校验	(294)
16.3.3	保密	(295)
16.4	微处理器监控电路	(295)
16.4.1	芯片主要性能及引脚功能	(296)
16.4.2	典型应用电路	(297)
16.4.3	基于 MAX813L 芯片的 WDT 应用举例	(299)
16.5	指令冗余技术与软件陷阱技术	(300)
16.5.1	指令冗余技术	(300)
16.5.2	软件陷阱技术	(300)
16.6	电气隔离技术与继电器控制	(301)
16.6.1	电气隔离技术	(301)
16.6.2	继电器输出接口技术	(302)
16.7	步进电机的控制技术	(302)
16.7.1	步进电机的工作原理	(302)
16.7.2	步进电机的控制技术	(303)
16.8	I ² C 总线技术	(306)
16.8.1	I ² C 总线的电气结构	(306)
16.8.2	I ² C 总线的数据传输	(307)
16.8.3	I ² C 总线的模拟实现方法	(310)
16.8.4	I ² C 总线模拟应用举例——带 I ² C 总线接口的 E ² PROM 应用	(314)
	练习与思考题	(317)
第 17 章	总线技术	(319)
17.1	总线的基本概念	(319)
17.2	PCI 总线	(321)
17.2.1	PCI 的特点	(321)
17.2.2	PCI 总线系统结构	(322)
17.2.3	PCI 总线信号	(323)
17.2.4	PCI 总线周期和地址空间	(324)
17.2.5	PCI 配置空间	(326)
17.3	串行通信标准总线 RS-232-C	(331)
17.3.1	RS-232-C 的电器特性	(331)
17.3.2	RS-232-C 机械特性及引脚功能	(333)
17.3.3	RS-232-C 的连接	(334)
17.3.4	PC 系列 RS-232-C 接口	(336)
17.3.5	IBM-PC 机与 8031 双机通信技术	(337)
17.4	RS-423/RS-422/RS-485 总线	(342)
17.4.1	RS-423A 总线	(342)

17.4.2 RS-422A 总线	(343)
17.4.3 RS-485 总线	(344)
17.5 通用串行总线 USB	(345)
17.5.1 USB 总线的构成	(347)
17.5.2 USB 系统的接口信号和电器特性	(348)
17.5.3 USB 数据传输	(349)
17.5.4 USB 外围芯片及应用	(352)
17.6 高性能串行总线标准 IEEE1394	(353)
17.6.1 IEEE1394 的主要性能特点	(353)
17.6.2 IEEE1394 拓扑结构	(354)
17.6.3 IEEE1394 地址分配	(355)
17.6.4 IEEE1394 的主要技术规范	(355)
17.6.5 IEEE1394 与 USB 的比较	(356)
练习与思考题	(357)
附录 A ASCII 码字符表	(359)
附录 B MCS-51 系列单片机指令表	(360)
附录 C 8086/8088 指令系统表	(364)
附录 D 常用集成电路引脚图	(384)
附录 E 单片机在工业控制中的应用举例	(388)
参考文献	(403)

第 1 章 基础知识

1.1 微型计算机系统的组成

一个微型计算机系统主要由软件和硬件两大部分组成。软件主要有系统软件(如操作系统、监控程序等)和应用软件;硬件主要有计算机、外围设备及电源。外围设备主要包括输入/输出设备和用于过程控制的 I/O 通道等。

微型计算机的结构是指按照总体布局的设计要求将计算机硬件系统内的各部件构成某个系统的连接方式。至今为止,微型计算机硬件系统的组成无一不是采用冯·诺依曼(出生于匈牙利的美国数学家)结构。冯·诺依曼结构的基本内容是:

- ①用二进制形式表示数据和指令;
- ②将程序和数据预先存放在主存储器中,使计算机在工作时能够自动高速地从主存储器中取出指令和数据,并加以执行(这是“存储程序控制”的基本特点);
- ③确立计算机系统的 5 大基本部件,即运算器、控制器、存储器、输入设备、输出设备。

1.2 微型计算机主机结构

冯·诺依曼系统除输入/输出设备外,其余部件的组成称为主机。主机包括微处理器(MPU)、内存(主存储器)、输入/输出接口(简称 I/O 接口)、总线及地址译码电路等。MPU 嵌入到系统之后,便称为 CPU(中央处理器)。主机结构最突出的特点是模块化,其系统结构如图 1.1 所示。

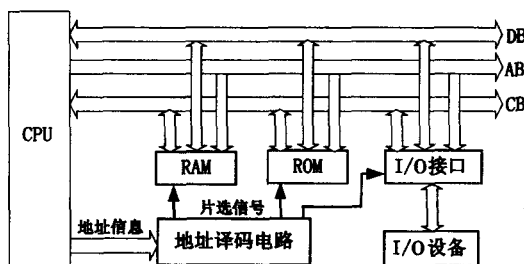


图 1.1 主机系统结构图

图 1.1 中的系统是一种总线结构,系统中各部件都“挂”在总线上,总线是用来传送信息的公共导线。根据所传送信息的内容与作用的不同,可将总线分为三类:数据总线 DB(Data Bus)、地址总线 AB(Address Bus)、控制总线 CB(Control Bus)。由于各总线传送信号的类型是单一的,该结构又称为面向系统的单总线结构。系统各部件均以同一形式“挂”在总线上,结构简单,易于扩充,体现了灵活方便的结构特点。

1.2.1 CPU

CPU 是计算机的核心部件,主要由运算器和控制器构成。运算器(ALU)主要负责算术运算和逻辑运算。控制器由指令寄存器、指令译码器和控制电路组成,并按指令的要求对计算机各部件发出相应的控制信息。

1.2.2 内存存储器

内存存储器(也称主存)是用于“记忆”信息的存储元件,均采用集成度高、容量大、体积小、功耗低的半导体存储器芯片。存储器以字节(Byte)为数据的基本存储单位。计算机中规定,8 位二进制数称为一个字节,例如:“01001101”。能存储一个字节数据的存储空间称为一个存储单元。也就是说,一个存储单元能存储 8 位二进制数,每一位也称为 1“bit”。

计算机的存储器是由许多存储单元集合而成的。每一个存储单元唯一地对应一个单元地址(也称为存储器的物理地址),计算机中的地址一般都用十六进制数表示。CPU 执行访问内存的指令时,按指定的单元地址对相应存储单元进行“读/写”操作,从某存储单元取出一个数,称为“读”操作,往某存储单元存入一个数,称为“写”操作。存储容量单位换算如下:

$$1\ 024\ \text{B} = 1\ \text{kB},\ 1\ 024\ \text{kB} = 1\ \text{MB},\ 1\ 024\ \text{MB} = 1\ \text{GB},\ 1\ 024\ \text{GB} = 1\ \text{TB}$$

$$\text{存储器容量}(s) = \text{存储单元数}(p) \times \text{数据位数}(i)$$

例如,一个存储芯片容量为 $2\ 048 \times 8$,说明它有 2 048 个存储单元。

计算机一条指令所能处理的一个基本信息单位,称为一个字。例如,一个数据(25、35.67、-0.003 8……)、一个字符(A、a、#……)等均称为一个字。

处理一个基本信息单位所占用的最大二进制位数,称为字长。例如,MCS-51 单片机内部寄存器长度为 8 位,执行一条指令,最多能处理 8 位二进制数,所以,MCS-51 单片机字长为 8 位,称为 8 位机。而 8086/8088 CPU 内部寄存器长度为 16 位,所以字长为 16 位,称为 16 位机。依此类推,80386、80486、80586 CPU(Pentium)字长均为 32 位,故均称为 32 位机。近年来(2001 年)英特尔公司新推出的“安腾”(Itanium)处理器为 64 位机。

字长是计算机的主要性能指标之一。字长越长的计算机,其运算速度越快,数的表示范围越大,数据的运算精度越高,机器的整体功能越强。

1.2.3 总线

1. 数据总线 DB

DB 双向传输数据信息,其宽度(根数)与 MPU 提供的数据线的引脚数有关。数据总线越宽,传输数据的能力越强。

2. 控制总线 CB

CB 用于传送各种控制信号和状态信号,对于每一根控制总线而言信息传送是单向的。

3. 地址总线 AB

CPU 执行指令时,AB 用于单向传送地址信息。地址信息包括两种:指令代码在程序存储器中的地址信息和操作数在数据存储器中的地址信息。CPU 执行一条指令时,首先从程序存储器中将欲执行指令的代码取入 CPU 中的指令寄存器(IR),经指令译码器(ID)译码后,产生相应的操作时序,再根据指令提供的操作数的地址信息,对操作数进行读/写操作。

AB 的宽度决定了计算机系统的最大寻址能力(寻址空间)。系统最大寻址空间为 2^N 字节, N 为 AB 的宽度。

例如,对于 MCS-51 单片机, $N = 16$, 则最大寻址空间 = 2^{16} 字节 = 65 536 字节 = 64 kB。

又如,8086/8088 CPU 的 $N = 20$, 则最大寻址空间 = 2^{20} 字节 = 1 MB。

1.2.4 地址译码电路

凡是“挂”在总线上的部件都被系统分配一个地址域, CPU 访问某部件时, 由指令提供被访问部件的地址信息, 该地址信息经地址译码电路译码后唯一地产生一个选通信号(也称片选信号), 将被选中的部件“门”打开, 使得数据得以传输。

1.2.5 接口

接口是主机与外设连接的必然通路, 是必经的“桥梁”。复杂的设备有复杂的接口, 简单的设备有简单的接口。即使一个灯、一个开关或按钮与计算机连接也必须通过接口。每个接口可包含若干个端口, 每个端口对应一个端口地址, 可由指令按地址访问端口。

接口的主要功能有:

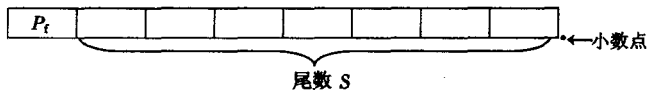
- ①隔离主机与外设之间的数据;
- ②向外设传输控制信号和接收外设的状态信号;
- ③数据类型的转换。

1.3 计算机中数的表示方法

用二进制表示一个数有两种形式: 定点表示法和浮点表示法。所谓定点表示就是小数点在数中的位置是固定的; 所谓浮点表示就是小数点的位置是浮动的。用浮点表示法的计算机称为浮点机(例如 PC 机), 用定点表示法的计算机称为定点机(例如 MCS-51 单片机)。通常用定点法表示整数, 用浮点法表示实数。

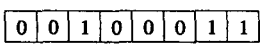
1.3.1 定点表示

定点表示有两种形式: 纯整数形式和纯小数形式, 定点机中通常采用纯整数形式。以 8 位机为例, 用 8 位二进制数表示一个纯整数, 格式如下:

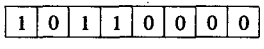


其中, P_i 表示数的正负: 若为“0”表示正数, 若为“1”表示负数。

例如, 数 35 的定点表示为



又如, 数 -48 的定点表示为



由上述可知, 在计算机中对于带正、负号的数, 其符号也被数码化了。这种已经数码化了