

接口与通信

高传善 郭健民 陈章龙 编著

JIE KOU YU TONG XIN

复

版社

接口与通信

高传善 郭健民 陈章龙 编著

复旦大学出版社

内 容 简 介

《接口与通信》是按全国计算机教育学会的本科推荐大纲编写。在简述微型计算机原理、系统总线和各种数据传送方式(查询、中断、DMA 和专用 I/O 处理器)的基础上,着重介绍了存储器接口、并行 I/O 接口、串行 I/O 接口、模拟接口和人机接口的各种芯片的接口方法和应用,以及网络通信中的接口和数据传输。本书收编了大量的各种接口应用实例和 IBM PC 微型机系统中的系统接口实例。

该书可作大专院校的教科书,也可作为从事微型机应用的工程技术人员的技术参考书。

责任编辑 陆盛强

责任校对 马金宝

接 口 与 通 信

高传善 郭健民 陈章龙 编著

复旦大学出版社出版

(200433 上海国权路 579 号)

新华书店上海发行所发行 复旦大学印刷厂印刷

开本 787×1092 1/16 印张 27.75 字数 711千字

1989 年 12 月第 1 版 1996 年 4 月第 7 次印刷

印数 32,001—35,000

ISBN 7 309-01328-X/T·104

定价: 28.00 元

TP360.4

1

前言

计算机科学与技术发展迅速、日新月异,相应的知识范围和教学要求也要随之不断更新。IEEE颁布了新的83教程。全国计算机教育与培训学会在几十所高等院校参与下,经多年研究与讨论也于1987年制定了向全国各高等院校计算机科学与工程类专业推荐的本科教学计划与教学大纲。在这两个大纲中,都将“接口与通信”列为主干课程,以取代原有的“微型计算机原理”。本书正是按照后一推荐大纲,并参照了IEEE 83教程编写的一本教材。本课程的先行课程为“汇编语言程序设计”和“计算机组成原理”。本书中,凡上述两课程中已有的有关内容,除反映了微型计算机特点的部分作简要介绍外都不再重复,而着重介绍计算机作为信息或控制系统的核心与外界联系的“接口与通信”的基本原理与方法。这也是计算机硬件与软件衔接而组构系统的一个重要领域。由于有较大的应用性,本本也适合于工程技术人员阅读和培训使用。

全书共分十章。第一章介绍了微型计算机的基本结构、接口的原理和网络互连的七层参考模型;第二章介绍了总线的结构、传输应答方式和共享资源的方法;第三章是存储器接口,介绍了存储器模块、接口方法和磁盘的接口;第四章为系统模块间控制与同步,介绍了查询、中断、DMA和专用I/O处理器的数据传送方式,并介绍了有关中断控制器、DMA控制器8237 (Z80-DMA)、专用I/O处理器8089的原理与应用,还介绍了Z80-CTC和8253定时/计数器的接口方法和应用;第五章是并行接口,介绍了并行接口的原理和常用的8255、6820和Z80-PIO接口芯片的接口方法和相应的应用,另外还介绍了IEEE-488通用并行接口总线;第六章是数据通信基础,介绍了数据通信的基本概念、各种差错检测与处理技术以及通信协议;第七章串行接口中,介绍了RS-232和RS-449等物理接口,常用的MC 6850、8251和Z80-SIO等串行接口芯片的功能和编程方法;第八章为模拟接口,介绍了A/D、D/A接口和相应的应用;第九章是人机接口,介绍了人机接口原理、CRT接口和新型的人机接口(包括图像、声音和汉字的输入/输出的原理);第十章中介绍了输入/输出软件模块的设计和微型机开发系统的原理。附录中列有Z80微处理器的结构、指令系统和程序设计举例,8086/8088微处理器结构、指令系统和相应程序设计,以及STD、MULTIBUS和VME几种典型总线,可供对照与查阅。

第二章和第十章根据实际情况要求不同可以少讲或不讲;第二章也可以放在第九章以后再讲述。本书为了说明接口原理和方法,介绍了各种系列的接口芯片,收编了相应的接口应用实例和IBM-PC/XT微机系统接口。可以根据不同的教学要求和进度选讲其中部分接口芯片及其应用。附录中的Z80和8086原理、指令系统,也可根据需要选讲。

本书由高传善同志组织编写。其中,第一、六、七章由高传善同志编写,第二、四章和第五章中第三节由郭健民同志编写,第三、八、九、十章和第五章中第一、二节由陈章龙同志编写。最后,由陈章龙同志统编成书。王小勇、孙慰迟等同志对本书提出了许多宝贵的意见。在编写过程中,我们力求尽可能符合推荐大纲的要求,但由于水平有限,错误和不妥之处敬

请批评指正。

在本书的编写和出版中得到了陆盛强同志的帮助,在此谨向所有为本书编写、编辑、出版和发行中作出贡献的同志一并致谢。

编 者

1989.4

目 录

前 言

第一章 概论	1
§ 1.1 微型计算机概述	1
1.1.1 微型计算机的发展	1
1.1.2 微型计算机的基本结构	2
§ 1.2 接口的基本原理	8
1.2.1 接口与通信在微机系统结构中的作用	8
1.2.2 接口的操作原理	9
§ 1.3 开放系统互连参考模型	15
1.3.1 计算机网络的发展	16
1.3.2 OSI 七层模型	17
第二章* 总线	21
§ 2.1 总线连接	21
§ 2.2 总线的数据传输	23
2.2.1 同步式传输	23
2.2.2 异步式传输	25
2.2.3 半同步式传输	27
2.2.4 分离式传输	29
§ 2.3 常用的总线标准	30
第三章 存储器接口	33
§ 3.1 半导体存储器	33
3.1.1 概述	33
3.1.2 读/写存储器 RAM	33
3.1.3 只读存储器 ROM	39
§ 3.2 存储器的接口技术	43
3.2.1 存储器模块的组成	43
3.2.2 存储器接口技术	46
3.2.3 存储器的寻址	50
§ 3.3 磁盘存储器及其接口	54
3.3.1 软磁盘	55
3.3.2 软磁盘接口	58
3.3.3 硬磁盘及其接口	64
第四章 系统模块间控制与同步	67
§ 4.1 模块间输入输出传送方式	67
4.1.1 模块间的接口信号	67
4.1.2 传送的方式	67

§ 4.2 程序查询的传送方式	69
4.2.1 查询式输入	69
4.2.2 查询式输出	71
§ 4.3 中断的传送方式	72
4.3.1 概述	72
4.3.2 中断的处理过程	74
4.3.3 中断的优先级	76
4.3.4 微处理机的中断机构	81
§ 4.4 直接存储器存取(DMA 传送)方式	90
4.4.1 概述	90
4.4.2 DMA 控制器	92
4.4.3 Z80-DMA	95
§ 4.5* 输入输出专用处理器 IOP	109
4.5.1 CPU 与 IOP 间的信息交换	110
4.5.2 通道的运行和 DMA 传输	113
4.5.3 多处理机信号灯与双口存储器	115
§ 4.6 系统的时间基准	116
4.6.1 可编程的定时器/计数器电路	116
4.6.2 Z80-CTC的组成和功能	118
第五第 并行接口	132
§ 5.1 并行接口的特性	132
5.1.1 输入/输出数据缓冲器	132
5.1.2 并行接口的应答	135
5.1.3 并行接口的结构	136
5.1.4 输入/输出的寻址方法	137
§ 5.2 常用并行接口芯片及接口方法	138
5.2.1 8255 及其接口方法	139
5.2.2 MC 6820 及其接口方法	154
5.2.3 Z80-PIO 及其接口方法	161
§ 5.3* IEEE-488 通用接口总线	179
5.3.1 IEEE-488 总线的接口信息与设备信息	180
5.3.2 IEEE-488 总线的信号交换	181
5.3.3 IEEE-488 总线的接口功能	182
5.3.4 IEEE-488 总线的接口电路	184
第六章 数据通信基础	190
§ 6.1 基本概念	190
6.1.1 通信系统的模型	190
6.1.2 信道的带宽与数据速率	191
6.1.3 数字调制技术	193
6.1.4 多路复用	195
§ 6.2* 差错控制	196
6.2.1 传输差错的特性	196

6.2.2	常用的简单差错控制编码	197
6.2.3	海明码	201
6.2.4	循环冗余码	203
§ 6.3	通信协议	207
6.3.1	起止式异步协议	208
6.3.2	面向字符的同步协议	209
6.3.3	面向比特的同步协议	211
6.3.4	面向字节计数的同步协议	214
第七章	串行接口	216
§ 7.1	物理接口标准	216
7.1.1	EIA-RS-232 C	216
7.1.2	RS-422, RS-423 和 RS-449	221
§ 7.2	简单的串行接口芯片	222
7.2.1	一般结构及 UART	222
7.2.2	MC6850-ACIA	224
7.2.3	MC 6852-SSDA	225
§ 7.3	较复杂的串行接口芯片	227
7.3.1	8251-USART	227
7.3.2	8273 和 MC 6854	231
7.3.3	Z80-SIO	233
第八章	模拟接口	241
§ 8.1	概述	241
8.1.1	模拟信号的采样、量化与编码	242
8.1.2	采样-保持器与多路开关	244
§ 8.2	数/模转换接口	247
8.2.1	数/模转换的方法	247
8.2.2	数/模转换器与微处理器的接口	253
§ 8.3	模/数转换接口	255
8.3.1	模/数转换的方法	255
8.3.2	模/数转换器与微处理器的接口	262
8.3.3	数据采集系统	268
第九章	人-机接口	273
§ 9.1	概述	273
§ 9.2	CRT 接口	274
9.2.1	CRT 显示器概述	274
9.2.2	CRT 显示器控制电路及接口	278
9.2.3	计算机的图形接口	282
§ 9.3	新型的人-机接口	300
9.3.1	语音识别与语音合成	300
9.3.2	图像的输入和输出	306
9.3.3	交互式人-机接口	313
9.3.4	汉字输入与汉字输出	318

第十章* 软件设计和系统开发	325
§ 10.1 软件设计.....	325
10.1.1 存储区域的分配.....	325
10.1.2 建立输入/输出的规范.....	327
10.1.3 实时编程.....	328
§ 10.2 微型机开发系统.....	331
10.2.1 微型机系统的开发.....	331
10.2.2 微型机开发系统.....	334
10.2.3 在线仿真器的开发过程.....	342
附录一 Z 80 微处理器	343
一、Z 80 微处理器的结构.....	343
二、Z 80 指令系统.....	348
三、Z 80 程序设计举例.....	362
四、Z 80 微处理器时序.....	369
• Z 80 指令系统表.....	376
附录二 8086/8088 微处理器	386
一、8086/8088 微处理器结构.....	386
二、8086/8088 指令系统.....	395
• 8086/8088 指令系统表.....	414
附录三 几种典型总线	418
一、STD 总线.....	418
二、MULTIBUS 总线.....	421
三、VME 总线.....	430
参考文献	435

第一章 概 论

§ 1.1 微型计算机概述

电子数字计算机是人类生产和科学技术发展的必然产物。微型计算机则是计算机中当今发展最迅速、应用最广泛的一种,是目前电子计算机的四个主要发展趋势(微型计算机、巨型计算机、计算机网络和计算机智能模拟)之一。

电子计算机系统由硬件和软件组成。计算机的硬件又分为运算器、控制器、存贮器和输入/输出设备等四部分。如果把电子计算机比喻为一个信息加工厂,那么这四部分在其中的作用就分别相当于信息加工厂的生产车间、指挥部门、仓库和供销部门。在计算机中运算器和控制器是核心,通常又将它们合称为中央处理器(CPU——Central Processing Unit)。微型计算机作为计算机的一种,在组成结构上与一般计算机有许多共性。其一个显著的特点是微型计算机的中央处理器是由一片或几片大规模集成电路构成的,称作为微处理器(μP 或MP——Microprocessor)。有时为了与大、中、小型机的中央处理器(CPU)区分,也称为MPU(Microprocessing Unit)。若用一个等式来示意,即

$$\mu P(MPU) = CPU$$

以微处理器为核心,配以大规模集成电路制作的存贮器、接口电路以及连接它们的系统总线,就构成了微型计算机(μC 或MC——Microcomputer),即

$$\mu C(MC) = \mu P + \text{存贮器} + \text{接口} + \text{系统总线}$$

微型计算机再配以相应的外部设备以及软件才构成一个完整的微型计算机系统(μCS 或MCS——Microcomputer System),即

$$\mu CS(MCS) = \mu C + \text{外部设备} + \text{软件}$$

1.1.1 微型计算机的发展

廿世纪七十年代,一方面军事工业、空间技术和工业自动化技术的发展,不断提出了生产体积小、可靠性高和功耗低的计算机的需要。另一方面大规模集成电路技术和计算机技术的发展提供了可能,两者相辅相成促成了微型计算机的产生和发展。

自1971年Intel公司推出其Intel 4004微处理器和由它构成的MCS-4微型计算机以来,微处理器和微型计算机得到了迅猛的发展,已经经历了三代的演变,现已进入第四代。

人们通常按微处理器中字长的位数及其功能来划分微处理器和微型计算机的换代。

第一代(1971~1973年)是4位或低档8位微机。继前述Intel首先制成的4004和MCS-4后,Intel公司后来又改进制成4040,并于1972年研制了8位的8008微处理器和由其组成的MCS-8微型计算机。这一代微机的基本特点是元件采用PMOS工艺,集成度较低(1,200~2,000晶体管/芯片),系统结构与指令都较简单,且速度慢(其基本指令执行时间为10~20 μs)。这一代微机的软件主要采用简单的机器语言或简单的汇编语言。虽然它的功能较差,但在如家用电器、计算器和简单控制器等一些简单的应用场合,已能满足要

求,其生产成品率极高,价格也十分低廉。所以,目前仍有生产与应用。

第二代(1974~1977年)是中高档的8位微处理器和微型计算机。其典型代表有Intel公司的8080(后来又有8085)、Motorola公司的MC 6800(后来又有6809)、Zilog公司的Z 80以及MOS Technology公司的6500系列。例如,Apple公司生产的著名微型计算机系统——苹果机,就是采用了6500系列中的6502微处理器芯片。这一代微机的特点是采用了NMOS工艺、集成度提高了约4倍(5,000~9,000晶体管/芯片),速度提高了10~15倍(其基本指令执行时间为1~2 μ s)。系统结构与指令都已比较完善,已具有典型计算机的体系结构,如中断、DMA等功能。软件除汇编语言外,还配有BASIC、FORTRAN等高级语言,并在后期配上了操作系统,如CP/M(Control Program/Monitor)。以8位微处理器组成的各种学习机、单板机和微型计算机系统像雨后春笋般涌现,在各个领域里得到广泛的应用。8位的单片微型计算机(Single-Chip Microcomputer),或简称为单片机,有时也称为微控制器(Microcontroller)也在这一阶段产生。它把CPU、存贮器和输入/输出接口电路都集成在一个芯片上,主要用于工业控制和智能仪器仪表,由于其性能/价格比高,销路很广。例如Intel公司就有8048和8051系列的单片机。

第三代(1978~1981年)是16位的微处理器和微型计算机,包括有准16位的微机(CPU内部字长为16位,而对外的系统总线仍为8位数据)和完全16位的微机(对外的数据总线也是16位的)。其典型代表有Intel 8086、Z8000和MC68000。目前,广泛流行的IBM PC及各种兼容机就采用了准16位的Intel 8088芯片为CPU。这一代微机的特点是采用了HMOS工艺,其集成度(达20,000~70,000晶体管/芯片)和速度(基本指令执行时间约0.5 μ s)都比第二代微型机提高了一个数量级。其系统结构与指令都更为完善与丰富,采用了多级中断、多重寻址、段式寄存器结构、乘除运算硬件等技术,并普遍配备有强有力的软件系统。各项系统性能都已经达到或超过七十年代中、低档小型计算机的水平。其中有些就是已经流行的小型计算机的微型化,例如DEC公司开发LSI-11系列微型机就是以其小型机PDP-11为背景的。16位微型机由于弥补了8位微机在字长和速度限制方面的缺陷,有更广泛的应用前途,得到了迅速的发展。

第四代(1982年以后)是32位的微处理器和微型计算机。开始也是将CPU内部的字长扩充为32位,成为准32位的;而后对外的数据总线也改为32位的,成为完全的32位机。典型的代表有Intel公司的80386、Motorola公司的68020、Zilog公司的Z80000以及AT&T公司的WE32100等。第四代微处理器大多采用HMOS或CMOS的超大规模集成电路,它的集成度已达每芯片上数十万个晶体管,而基本指令执行时间却降为0.125 μ s以下,系统速度大为提高。可靠性增加、成本降低、功能已足以和过去高档的小型计算机匹敌,大有取代中、小型计算机之势。

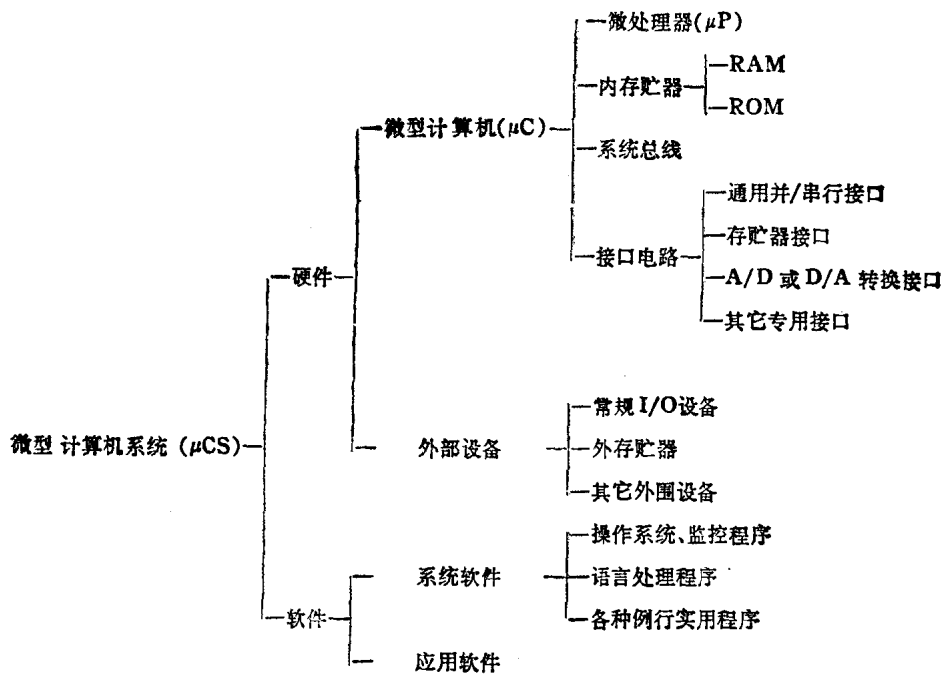
综上所述,微型机约每隔3年就更新换代一次,可见其发展之迅速,目前已进入第四代。关于各代间的比较可归纳为表1.1。

1.1.2 微型计算机的基本结构

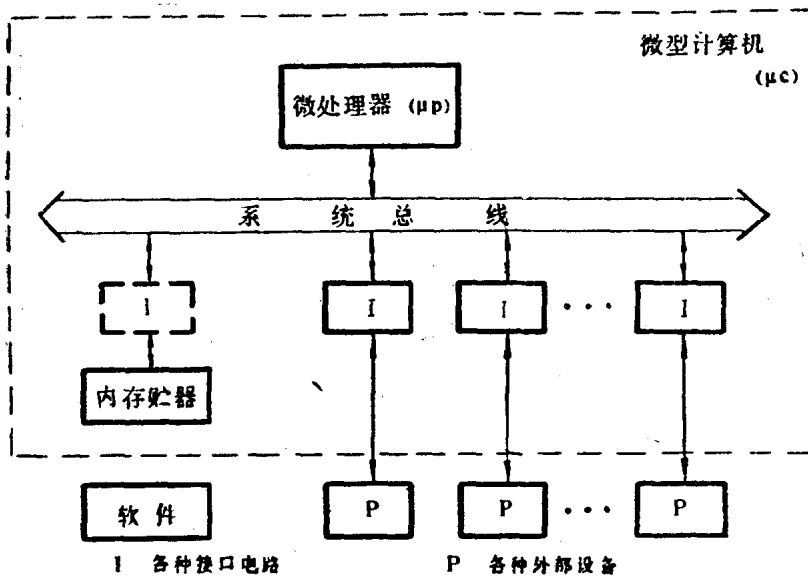
如前所述,一个完整的微型计算机系统由微处理器、存贮器(为了与磁盘、磁带或软盘这类相对来说速度慢而容量大的外存贮器区分开来,又称为内存贮器或主存贮器)、接口电路、系统总线、外部设备和软件等部分组成。其相互关系及进一步的细分见p4表所示:

表 1.1 各代微型机的比较

代 项 目	第 一 代 (1971~1973)	第二代 (1974~1978)	第三代 (1979~1981)	第 四 代 (1982年以后)
字长(位)	4/8	8	16	32
集成电路技术	PMOS LSI	NMOS LSI	HMOS LSI/VLSI	HMOS/CMOS VLSI
基本指令执行时间 (μ s)	10~20	1~2	0.5	≤ 0.125
典 型 产 品	Intel 4004 4040 8008	Intel 8080 Motorola 6800 Zilog z 80 MT 6500	Intel 8086 Motorola 68000 Zilog Z 8000	Intel 80386 Motorola 68020 Zilog Z80000 AT&T WE 32100
功 能 及 其 它	系统结构与指令较简单; 采用机器语言和简单的汇编语言; 价格十分低廉	系统结构与指令都比较完善; 具有中断、DMA等功能; 除汇编语言外, 还配有高级语言; 后期配上了操作系统; 开始出现单片机	系统结构与指令更完善与丰富; 采用了多级中断、多重寻址、段式寄存器结构、乘除运算硬件等技术; 配有完善的软件系统; 性能已达到或超过70年代中档小型计算机的水平	各方面性能进一步提高, 功能已足以和过去高档的小型计算机匹敌, 有取代中、小型计算机之势



或者我们可画出如图 1-1 所示的示意结构框图。



注：除了较复杂的微处理器与大容量存储器接口时，需要相对地址变为绝对地址等功能，因而要用专门的存储管理接口芯片外，对于绝大多数8位微机来说，常常只需要简单的地址译码电路，故这里接口电路用虚框画出。

图 1-1 典型的微机结构框图

其中，微处理器是核心，各种微处理器芯片的内部结构是不完全一样的。图 1-2、图 1-3 和图 1-4 分别给出了典型的 8 位微处理器 Intel 8080、Z80 和 MC 6800 的内部结构。

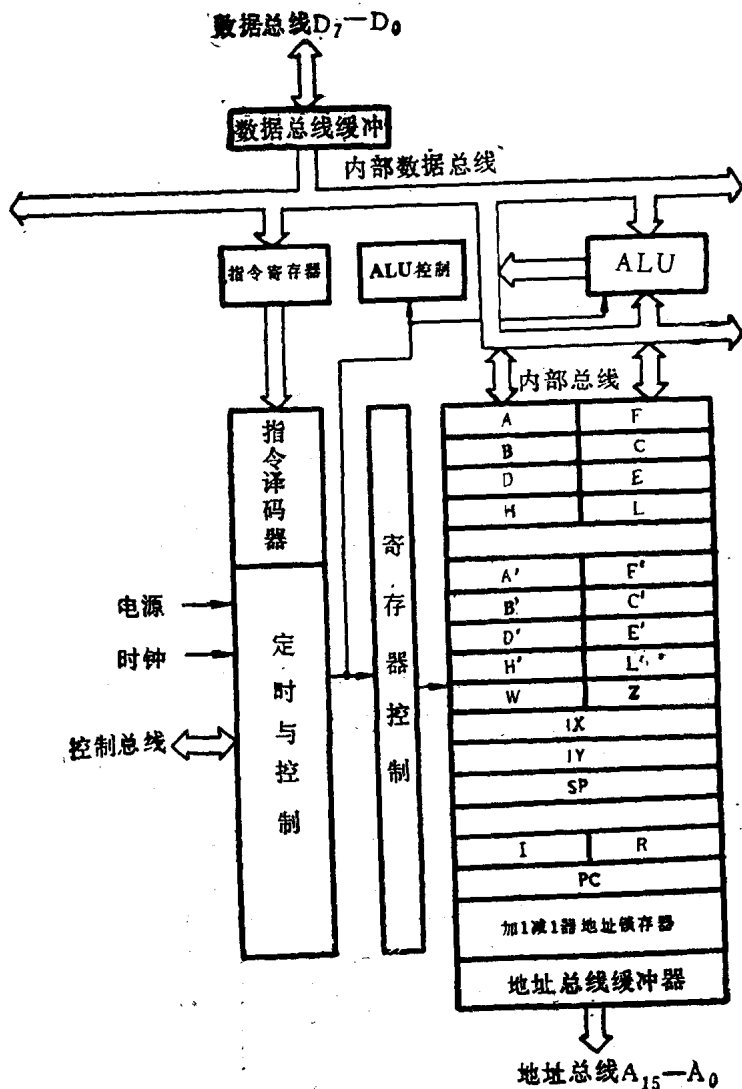


图 1-3 Z80 内部结构框图

比较图 1-2~图1-4可见,虽然其内部结构不尽相同,但都包括下列三个基本功能部件:

1. 算术逻辑部件(ALU——Arithmetic and Logical Unit),又称运算器,用来执行各种算术逻辑操作。

2. 指令寄存器、指令译码以及定时与控制部件,合称控制器,用来根据指令的操作码产生各种操作控制信号,包括脉冲和电平,以协调微处理器内部,并通过控制总线协调微型计算机内各部件间的动作。

3. 寄存器阵列,包括通用寄存器和各种专用寄存器,用来暂时存放数据、地址与状态。

以上几个部分常通过内部数据总线互连在一起。整个微处理器又通过外部总线(即前述系统总线)与微型计算机的其它部件互连。外部系统总线通常包含数据总线、地址总线与控制总线三部分。系统总线再加上电源线和时钟线(ϕ)构成了 μP 芯片的全部引线。图 1-5、图 1-6 和图 1-7 分别列出了 Intel 8080、Z80 和 MC 6800 的引线图。

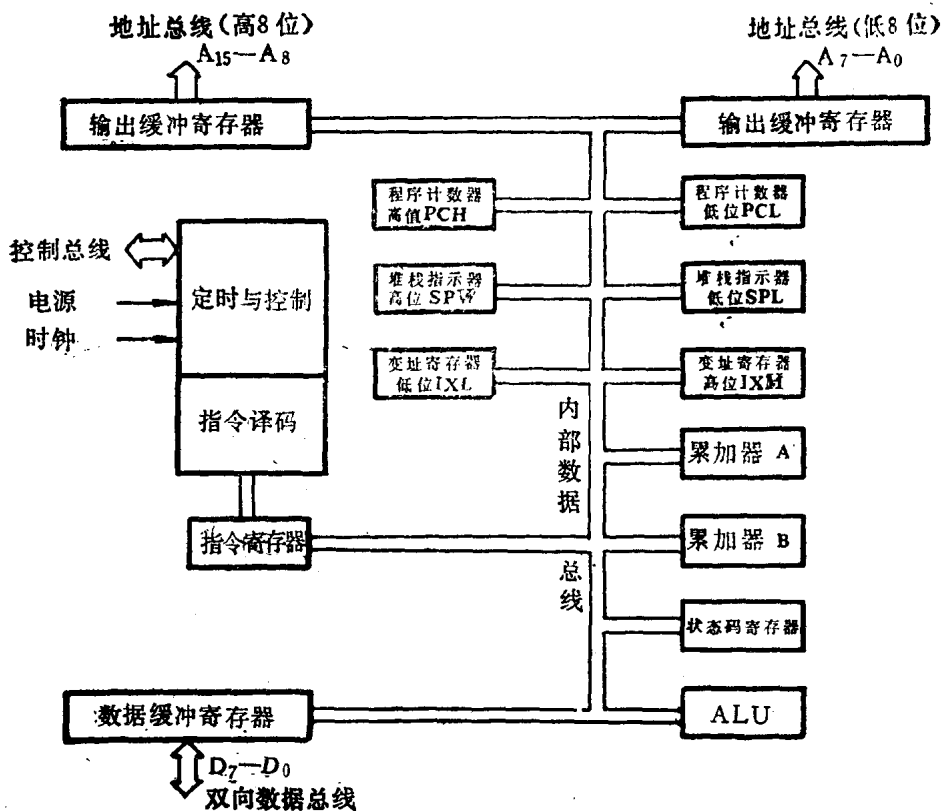


图 1-4 MC 6800 内部结构框图

这三个 8 位微处理器芯片都是 40 脚的，其中 $D_i (i=0\sim7)$ 、 $A_j (j=0\sim15)$ 分别代表 8 位数据总线和 16 位地址总线，除了排列位置不同外是完全一样的。但是时钟线和控制总线对这三个微处理器来说就不完全相同了，以它们为核心来构成微型计算机的连接方法也就不一样。此外，由于这三个微处理器芯片的内部结构不一样，它向用户提供的功能集中表现在指令系统上也就不一样。关于 Z80 和 Intel 8086/8088 的指令系统可参见附录一和附录二。但是，由于它们内部结构如前所述有许多共同点，其指令系统也有许多类似的地方，掌握了一种微机的指令系统后，再举一反三学习另一种微机的指令系统也就不困难了，尤其对于同一档的微机（例如同为 8 位机）更是如此。

微型计算机的内存贮器也是由大规模集成电路芯片构成的，通常包括随机访问存贮器 (RAM——Random Access Memory) 和

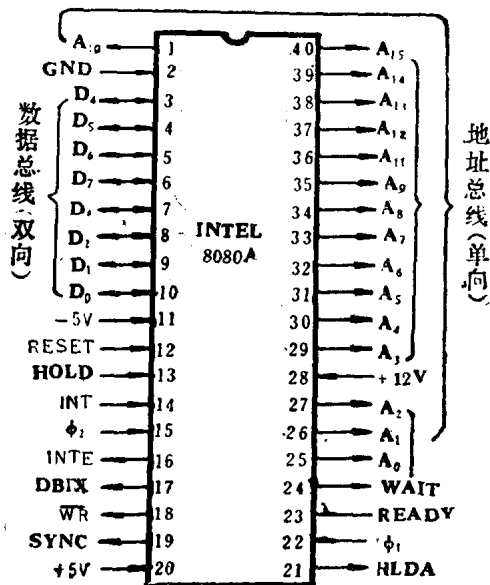


图 1-5 Intel 8080 芯片的引线图

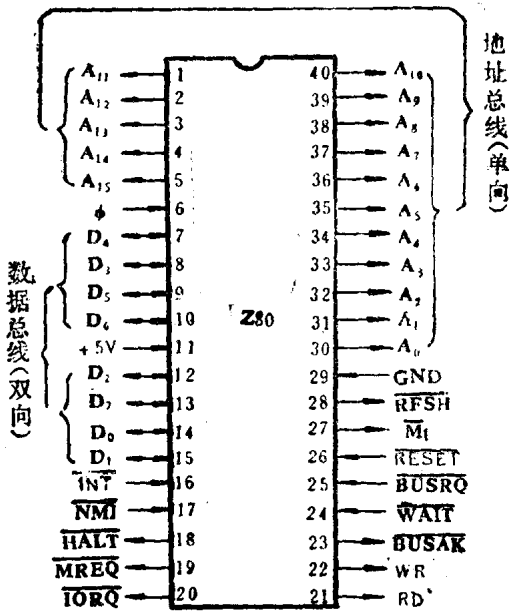


图 1-6 Z80 芯片的引线图

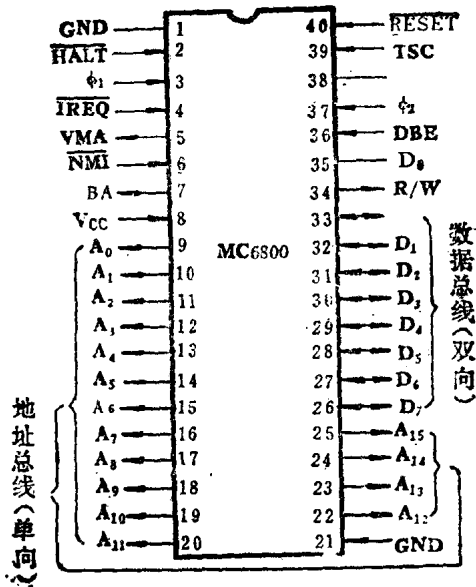


图 1-7 MC 6800 芯片的引线图

只读存储器(ROM——Read Only Memory)两部分。微机中系统软件的核心部分常是固化在 ROM 中的。微机有多种系统总线与不同类型的接口电路，如何软硬结合利用它们，再连接形形色色的外部设备以针对特定的应用要求构成一个实用的微型计算机系统是本书要讲解的重点。操作系统是系统软件的核心，但在一些简单的微型计算机系统中也可以没有操作系统，而只有一个更为简单的监控程序(例如在一些单板微机系统中就是这样)。语言处理程序包括汇编语言的汇编程序、BASIC 语言的解释程序和其它高级语言的编译程序等。实用程序是指如编辑程序、排错程序、诊断程序等这类通用的工具性程序。应用程序则是指用户为特定的应用需要而开发的程序。

§ 1.2 接口的基本原理

1.2.1 接口与通信在微机系统结构中的作用

什么是接口与通信？简单地说，接口(Interface)就是两个部件之间的连接点或边界。例如，微处理器和各种外部设备间的接口、微处理器和内存贮器间的接口等。两部件之间的连接包括物理电路上的连接，也包括逻辑上的连接，即信息的交换(例如数据、命令、状态等的交换)。前者是通过接口电路来实现的，后者是广义上的通信。有的时候，我们也把近距离的连接称为接口，而把远距离的连接称为通信。

以微处理器为核心的微型计算机系统，具有体积小、功耗低、使用方便灵活、造价低廉等许多优点。在工业控制、数据处理、数据通信、仪表、人们的日常生活等方面都得到了广泛的应用。在各种不同的应用中，要通过不同的外部设备与外界交换信息。目前，大规模集成电路技术已经把整个 CPU 集成在一块芯片上，为了与其配合连接不同的存储器与外部设备，以按照特定的要求构成各种用途的微型计算机和系统，各个制造厂商还提供了品种繁多、功