

单片机入门

DANPIAN WEIJI RUMEN

黄德 周建民 郝炳焜 编著



单片机应用入门

黄 德 周建民 郝炳焜 编 著

安徽科学技术出版社

责任编辑:何宗华

单片微机应用入门

黄德等编著

*

安徽科学技术出版社出版

(合肥市九州大厦八楼)

邮政编码:230063

新华书店经销 蚌埠涂山印刷厂印刷

开本:787×1092 1/16 印张:12.25 字数:302千字

1997年1月第1版 1997年1月第1次印刷

印数:4 000

ISBN 7-5337-1449-0/TP·41 定价:13.50元

(本书如有倒装、缺页等问题向承印厂调换)

内 容 简 介

本书共八章,前三章系统阐述了 MCS-51 单片机的基本原理,软、硬件知识和指令系统。第四、五、六章阐述了 MCS-51 单片机的功能开发、软件设计及外部扩展,着重解决在实际应用中所应注意的中断处理和接口技术问题。第七章介绍了用 MCS-51 系列芯片设计单机系统和分布式计算机系统的方法。第八章设计了必要的四个实验,以便读者理论联系实际、学以致用、全面掌握 MCS-51 系列单片机的内容,并为通晓其它类型的单片机奠定基础。

本书通俗、简明、系统、实用,它既是一本单片机的入门书,也是一本单片机的简明教程。

本书可作为大、中专院校非计算机专业的教材,也可供计算机应用人员培训和自学使用。

前 言

电子计算机技术是本世纪最伟大的科学发明之一。自 1946 年世界上第一台电子计算机诞生以来的 50 年中,计算机技术取得了飞速的发展,其应用遍及各行各业,已成为现代工业水平的重要标志。

随着集成电路(IC)技术的发展,微型计算机于 70 年代问世。微型计算机的 CPU(中央处理单元)是将计算机的运算器、控制器集成在一起的一块芯片上。

单片微型计算机简称单片微机或单片机,它是将 CPU、程序存储器 ROM 或 EPROM、定时/计数器、输入/输出接口等多功能部件集成在一块芯片上。从这一意义上讲,单片机即微型计算机单片化。因其面向控制,所以又称微控制器。

单片机的发展大体经过了三个阶段。

1974 年 Fairchild 公司推出了八位的 F8 单片机。它是采用双片形式,功能也较简单;1976 年 Intel 公司将 MCS-48 单片机系列推向市场。这可为单片机发展的第一阶段。

MCS-48 单片机投向市场后,大大推动了单片机的发展。70 年代末至 80 年代初出现了高性能单片机,有代表性的是 Intel 公司的 MCS-51 系列和 Motorola 公司的 6801,其片内 ROM、RAM 容量增大,寻址达 64K 字节,有串行输入/输出接口,能进行多级中断处理。此为单片机发展的第二阶段。

80 年代中期以来为单片机发展的第三个阶段。这一时期的单片机,实时处理能力更强,集成度更高。如 Intel 公司的 MCS-96 系列单片机,其主频达 12MHz,ROM 有 8K 字节,RAM 有 232 字节,8 个中断源,多通道 10 位 A/D 转换,还有高速输入输出部件。特别是 1990 年 2 月美国推出了 i80860 超级单片机,其运算速度达 1.2 亿次/秒,可进行 32 位整数运算和 64 位浮点运算,片内还有一个三维图形处理器,可构成超级图形工作站。这为巨型计算机单片化展现了光辉的前景。

90 年代以来,单片机在向着大容量高性能、小容量低价格的方向发展。大容量高性能的单片机是指 CPU 的功能强、内存容量大,用于复杂控制的场所。如 MCS-51 系列单片机增加了一个布尔处理器,其位处理功能更强,输入/输出速度更快。小容量低价格单片机内存容量相对较小,输入/输出通道较少,CPU 指令比较精简,单片机的功能也相应地有所简化。这种单片机价格低廉,适合家用电器等控制比较简单的场合。

MCS-51 系列 8 位单片微机具有价格低、性能价格比高、应用软件齐全、开发应用方便等突出特点,用它研制机电一体化产品和测控系统时,便于实现智能化功能和多机通信功能,工业界和科技界仍普遍采用。因此,我们编写的《单片微机应用入门》一书,主要介绍 MCS-51 系列单片机。

在编写本书时,我们参考了国内一些同类著作,在此向有关作者谨表谢意。

本书第一、二、三章由黄德编写,第四、五、六章由周建民编写,第七、八章由周建民、郝炳焜编写。

限于编者水平,本书难免有不妥之处,望同行及读者指正。

前 言

作 者

1996.3

本书是作者从事计算机教学工作多年,结合教学实践,参考国内外有关文献,编写而成的。本书可作为高等院校计算机专业及相关专业教材,也可供从事计算机工作的工程技术人员参考。

本书共分八章。第一章介绍计算机的发展概况、分类及应用;第二章介绍计算机中数据的表示方法;第三章介绍运算方法和运算部件;第四章介绍中央处理器(CPU)的结构和功能;第五章介绍总线系统;第六章介绍指令系统;第七章介绍中央处理器的性能指标;第八章介绍中央处理器的应用。

本书在编写过程中,参考了国内外许多有关文献,在此表示衷心的感谢。由于编者水平有限,书中难免存在不足之处,恳请读者批评指正。

本书在编写过程中,得到了许多同行及朋友的帮助,在此表示衷心的感谢。

本书在编写过程中,得到了许多同行及朋友的帮助,在此表示衷心的感谢。

本书在编写过程中,得到了许多同行及朋友的帮助,在此表示衷心的感谢。

本书在编写过程中,得到了许多同行及朋友的帮助,在此表示衷心的感谢。

本书在编写过程中,得到了许多同行及朋友的帮助,在此表示衷心的感谢。

本书在编写过程中,得到了许多同行及朋友的帮助,在此表示衷心的感谢。

目 录

前 言	1
1 基础知识	1
1-1 计算机的基本工作原理	1
一、计算机的硬件	1
二、计算机的软件	2
三、计算机的技术指标	3
1-2 微型机与单片机	4
一、中央处理单元 CPU	4
二、微型计算机的构成	6
三、单片微型计算机	8
1-3 半导体存储器	8
一、存储器的分类	9
二、随机存储器 RAM	9
三、静态 RAM 的基本结构	11
四、只读存储器 ROM	12
五、存储器系统	14
1-4 输入/输出接口	16
一、I/O 接口的编址方式	16
二、CPU 与外设间传送信息的方式	16
1-5 中断	17
一、中断系统及其功能	17
二、中断源	18
三、中断分类	18
四、中断处理过程	18
五、嵌套中断	20
2 MCS-51 单片机系统的硬件结构	21
2-1 MCS-51 单片机的主要功能特点	21
2-2 MCS-51 芯片的内部结构	22
2-3 MCS-51 芯片的引脚功能	23
一、电源	23
二、时钟电路	23
三、控制信号	25
四、I/O 口	26
五、MCS-51 芯片引脚的应用特性	29

2-4	MCS-51 的存贮器配置	31
	一、程序存贮器	31
	二、数据存贮器	32
	三、专用寄存器 SFR	34
	四、外部数据存贮器	39
2-5	MCS-51 CPU 时序	39
	一、机器周期、状态、相位	39
	二、典型指令的取指和执行时序	39
	习 题	41
3	MCS-51 指令系统	42
3-1	概述	42
	一、指令系统与程序	42
	二、汇编语言	42
	三、汇编与反汇编	42
	四、伪指令	43
	五、指令系统中的具体规定	45
3-2	MCS-51 指令的寻址方式	47
	一、立即寻址	47
	二、直接寻址	48
	三、寄存器寻址	48
	四、寄存器间接寻址	49
	五、基址寄存器加变址寄存器间接寻址	49
	六、相对寻址	50
	七、位寻址	51
3-3	MCS-51 指令	51
	一、数据传送指令(共 29 条)	51
	二、算术运算指令(共 24 条)	61
	三、逻辑运算指令(共 24 条)	69
	四、控制转移指令(共 17 条)	75
	五、布尔处理类指令(共 17 条)	79
	习 题	84
4	P3 端口的第二种功能开发	87
4-1	MCS-51 的定时/计数器	87
	一、定时/计数器工作原理	87
	二、定时/计数器的模式寄存器和控制寄存器	87
	三、定时/计数器的 4 种工作模式	89
	四、初始化	90
	五、定时/计数器应用举例	91
4-2	MCS-51 的串行口	93

831	一、串行通信的两种基本方式	93
831	二、MCS-51 串行口结构	94
831	三、串行口工作模式	95
871	四、波特率的设计	98
131	五、串行口的应用	100
831	4-3 MCS-51 单片机中断系统	103
831	一、中断的一般功能	103
071	二、MCS-51 单片机的中断源	104
871	三、中断控制	105
471	四、中断响应过程	108
771	五、外部中断的响应时序与触发方式	108
881	六、多外部中断源的设计	109
	习 题	111
	5 软件设计	113
	5-1 简单程序举例	113
	5-2 分支程序	114
	一、简单分支程序	114
	二、N 路分支程序	115
	5-3 查表程序	117
	5-4 子程序的设计	120
	5-5 程序设计举例	124
	习 题	128
	6 8051 芯片的外部扩展	130
	6-1 存储器设计	130
	一、MCS-51 存储器系统配置	130
	二、程序存储器扩展设计	131
	三、数据存储器扩展设计	135
	四、存储器系统扩展设计	138
	6-2 并行 I/O 接口电路扩展设计及应用	138
	一、8155/8156 可编程 RAM/I/O 扩展器	139
	二、显示器接口	142
	6-3 单片机与数模(D/A)及模数(A/D)转换器接口	146
	一、D/A 接口	146
	二、A/D 接口	149
	习 题	152
	7 应用实例	154
	7-1 单片机单机系统——智能化蒸汽流量计	154
	一、LFXW 型微机蒸汽流量计的基本原理和设计	154
	二、LFXW 型微机蒸汽流量计	155

59	三、单片机系统的硬件设计	156
60	四、流量的补偿方法及软件概述	157
62	7-2 分布式系统——汽车安全检测系统	159
62	一、检测原理	159
60	二、国产全自动汽车检测线	161
8	实 验	168
63	实验 1 键盘操作、程序输入与断点设置	168
64	实验 2 程序调试	170
65	实验 3 程序设计与调试	172
68	实验 4 MCS-51 单片微机系统设计与电路连接	174
	附 录	177
	参考书目	186

1 基础知识

1-1 计算机的基本工作原理

一、计算机的硬件

一般来说,计算机由五个部分组成:即运算器、控制器、存储器、输入设备及输出设备。这些设备称为计算机的硬件,如图 1-1 所示。我们把运算器、控制器、存储器称为计算机的主机。而输入设备和输出设备则称为计算机的外部设备,简称外设。外设与主机之间通过输入输出接口(简称 I/O 接口)电路连接。如有必要,还可对计算机系统配备外部存储器,称之为外存,而把主机内的存储器称为内存。

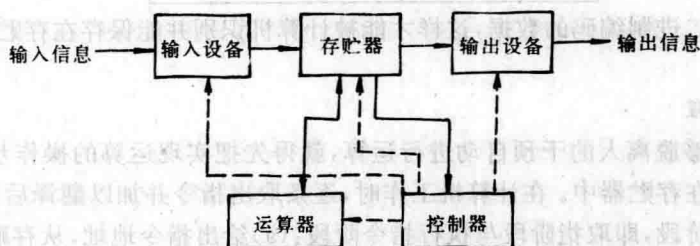


图 1-1 计算机的基本结构(实线表示程序、数据、状态信息,虚线表示控制信息)

控制器的主要功能是使计算机自动执行程序。它从内存中取出指令并译码,按指令要求执行以下控制任务:运算器对数据信息的处理与加工,输出运算结果,外部设备与主机之间的信息交换,计算机系统中随机事件的自动处理等。

运算器的主要功能是进行数据处理与加工。所谓数据处理是指数值运算或非数值运算,如进行加、减、乘、除等四则运算,或进行与、或、非、异或、移位、比较等逻辑运算。运算器又称为算术逻辑单元,简称 ALU。

存储器是计算机的记忆装置,用来存放各类数据和各种程序。每一个数据或每一条指令占用一个“单元”,为了区别这些不同的存储单元,通常给每一个单元编上一个号码(称为地址)。这样就可以根据确定的地址,存入或读出数据。

存储器有内存和外存之分,内存是任何一台计算机必不可少的组成部分,它主要用来存放计算机当前要执行的程序和有关的数据。内存容量较小,但存取速度较快。从功能上来说,内存又分为只读存储器 ROM 和随机存取存储器 RAM 两大类,分别用来存放程序和数据。外存存储器主要用来存放大量暂时不执行的程序和备用的数据。通常,外存的容量比内存大得多,又称为海量存储器,是计算机的后备信息仓库。目前常使用的有软磁盘、硬磁盘及磁带等。

输入和输出设备是人机联系的桥梁。输入设备的功能是把外部信息(如模拟量、开关量)转换成计算机能够接受的形式。常用的输入设备主要有:各类键盘、磁卡读入器、光电文字读入

器、磁带机、磁盘机、模/数(A/D)转换器等装置。输出设备的功能是把计算机的处理结果输出到外部,常用的输出设备有:显示器(CRT)、打印机、磁盘、数/模(D/A)转换器、绘图仪、复印机以及语音输出装置等。

计算机的输入/输出设备称为 I/O 设备,输入/输出作业被称为 I/O 操作。

二、计算机的软件

指挥计算机工作的各种程序,统称为软件。这里所说的程序是指为解决某一特定问题所编排的指令序列。

1. 指令和指令系统

所谓指令就是使计算机完成某种基本操作(如加、减、移位等)的命令。全部指令的集合成指令系统。

指令通常由两部分组成:第一部分为操作码,它表示计算机操作的性质(如加、减等);第二部分为操作数,它代表参加运算的数或存放该数的地址。指令的一般格式为:

操作码	操作数或地址
-----	--------

指令是一组以二进制编码的数据,这样才能被计算机识别并能保存在存储器中,称其为机器码或机器指令。

2. 指令执行过程

若要计算机能够脱离人的干预自动进行运算,就得先把实现运算的操作步骤以指令的形式一条一条地存放在存储器中。在计算机工作时,逐条取出指令并加以翻译后再执行。指令的执行过程分为两个阶段,即取指阶段与执行指令阶段:(a)给出指令地址,从存储器中取出指令并进行指令译码;(b)执行这条指令,然后再给出下一条指令地址,周而复始。

3. 程序设计语言

(1) 机器语言

用机器能够直接识别的二进制指令代码(即机器码)编写的程序称为机器语言。计算机的硬件结构能按照规定的逻辑,识别并执行这些指令码所规定的操作,于是计算机即可按机器语言编写的程序进行工作。由于机器语言无明显的操作特征,十分繁琐,又不便记忆,所以使用机器语言程序难写也难懂。

(2) 汇编语言

用机器指令系统的助记符(能反映指令特征和操作性质的英文单词或英文缩写),用符号代替操作数来编写的程序称为汇编语言程序。这种程序能反映指令功能,人们容易辨识、记忆和阅读。由于指令助记符与机器码有一一对应关系,因此它仍是一种依赖于机器硬件结构和指令系统的语言,是一种面向机器的语言。用汇编语言编写的程序不仅执行速度快,又可以有效地利用机器本身的专有特性,从而提高机器的工作效率。尽管如此,由于用汇编语言编写的程序面向机器,而不同机型的指令系统和硬件结构又互不相同,所以在一种机型上不能运行另一种机型的汇编程序,即是说汇编程序的通用性差,编程时必须深入了解机器的硬件结构,并熟悉机器指令系统。因此,用汇编语言编程仍是一项十分困难的工作。

(3) 高级语言、编译程序

为了方便用户,应当使程序语句与实际问题的更接近,即用户只考虑要解决的问题不必了解

具体机器的硬件结构就能编写程序,这就是面向问题的语言,如 BASIC、FORTRAN、PASCAL 等各种高级语言。

高级语言容易理解、学习和掌握,用户使用高级语言编写程序可大大减少工作量。但计算机执行时,必须将高级语言编写的源程序翻译成机器语言表示的目标代码方能执行,完成“翻译”功能的程序称为编译程序或解释程序。显然,高级语言的执行速度较慢,不太适合完成实时服务等功能。所以,在某些情况下汇编语言依然是必不可少的。

4. 系统软件

系统软件是用来提高计算机的使用效率、完善计算机的功能、简化程序设计、方便用户使用的特殊程序,一般由专门的计算机软件技术人员开发。如操作系统、监控程序、诊断程序、编译和解释程序统称系统软件。

5. 应用软件

利用各种程序设计语言和计算机系统软件编制的、用来解决用户各种实际问题的程序,统称应用软件。

三、计算机的技术指标

1. 字 长

所谓字长就是计算机的运算器(ALU)一次能处理(运算、存取)的二进制数的位数。字长越长,一个字所能表示的有效位就越多,计算的精度也就越高。

通常,八位二进制数称为一个字节(Byte),二个字节被称为一个字(Word),32 位的二进制数就是一个双字(Double word)。

一台计算机的字长是由运算一次能处理的二进制数长度所决定的。

2. 存储容量

存储容量是用来表示存储器存储二进制信息多少的一个技术指标。存储容量一般是以字节为单位计算的。存储容量越大,能存放的信息量就越大,计算机的功能也就越强。对于微型计算机而言,系统存储容量常见的有 64KB、128KB、256KB、512KB、640KB、1MB 以及更多等。

3. 指令系统

指令系统是计算机所有指令的集合。机器指令功能取决于计算机的硬件结构性能。

4. 指令执行时间

指令执行时间反映了计算机运算速度的快慢,它取决于系统的主时钟频率、指令系统的设计以及 CPU 的体系结构等。对于微型机而言,相应指标有主时钟频率以及执行每条指令所用的机器周期数。所谓机器周期就是计算机完成一次独立操作所持续的时间,这种独立操作是指存储器读或写、取指令操作码等操作。计算机的主频高,指令的执行时间就短,计算机的运算速度就快,系统的性能就好。指令执行时间是一项评价速度的重要技术指标。

5. 外部设备配置

外部设备是实现人机对话的设备。一台计算机所配置的外部设备种类多、型号齐全,人机对话的手段就多,系统的适应能力就强,通用性也就好。

6. 软件配置

所谓软件是指能完成各种功能的计算机程序的总和。软件是计算机的灵魂。计算机配置

的系统软件丰富、应用软件多、程序设计语言齐全,系统的功能就越强。

综上所述,对一台计算机性能的评价,要综合它的体系结构、指令系统、外设的多寡以及软件配置是否丰富等各项指标,才能正确评价与衡量其性能的优劣。

1-2 微型机与单片机

一、中央处理单元 CPU

中央处理单元将运算器和控制器集成在一块芯片上,它是整个微型计算机的核心,除了进行算术和逻辑运算外,还控制指挥其它部件协调工作。尽管不同类型的 CPU 在性能上各有不同,但一般都具备以下几种基本功能:

- (1) 执行算术、逻辑运算;
- (2) 与存储器或外部设备交换数据(接收或发送);
- (3) 暂存少量数据;
- (4) 对指令进行译码并执行指令所规定的操作;
- (5) 提供系统所需的定时和控制信号;
- (6) 响应其它部件发出的中断请求。

CPU 的结构可简单表示为图 1-2 的形式。

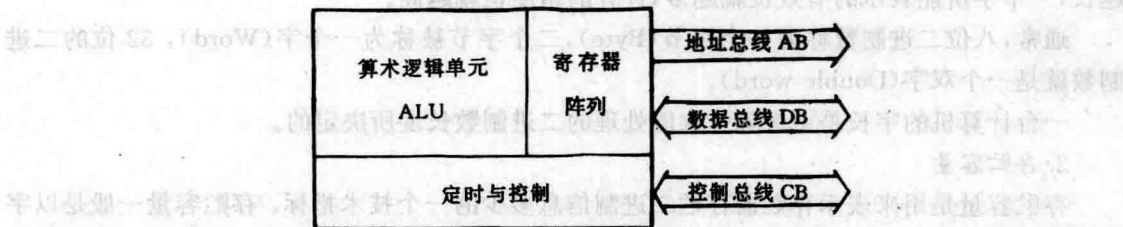


图 1-2 CPU 结构简图

由图可见,CPU 是由 ALU、寄存器阵列、定时与控制等部分组成,是实现上述基本功能的核心部件。而 CPU 是通过三条总线来实现与外部的联系。从应用角度来看,了解与掌握 CPU 的内部结构,可以最大限度地利用和发挥计算机的功能。

CPU 各部分的作用如下述:

1. 算术逻辑单元 ALU

算术逻辑单元是对数据信息进行加工的部件。数据加工包括:算术运算(加、减、乘、除),逻辑运算(与、或、非和异或),以及移位操作等。

2. 累加器 A

累加器 A 是 CPU 中最重要、使用最频繁的一个 8 位寄存器。它既是一个可供用户编程使用的通用寄存器,同时又是一个专用寄存器。算术运算和逻辑运算有一个操作数存放在累加器 A 中,而且最后的运算结果也都存放在累加器 A 中。在某些情况下,访问内存或执行 I/O 操作也只能使用累加器 A。

3. 标志寄存器 FR

标志寄存器也称程序状态字,是用来存放 ALU 运算结果的各种特征状态信息的,诸如运

算有无进(借)位、有无溢出、结果是否为零等。这些都可通过标志寄存器的相应位来反映。程序中经常要检测这些标志位的状态以决定下一步的操作。状态不同,操作处理方法就不同。微处理器内部都有一个标志寄存器,但不同型号的 CPU,其标志寄存器的标志数目和具体规定亦有不同。下面介绍几种常用的标志位:

(1)进位标志位 C 或 Cy

在做加法或减法运算时,如果最高位产生了进位或借位,该进位或借位就被保存在 C 中,有进(借)位 C 被置“1”,否则置“0”。另外,ALU 执行某些循环或移位操作也会影响该标志位。

(2)零标志位 Z

当 ALU 的运算结果为零时,零标志位 Z 即被置“1”,否则置“0”。一般加法、减法、比较与移位等指令会影响 Z 标志。

(3)符号标志位 S

符号标志位供有符号的数使用,它总和 ALU 运算结果的最高位的状态相同。在有符号数的运算中, $S=1$ 表示运算结果为负, $S=0$ 表示运算结果为正。

(4)奇偶标志位 P

奇偶标志位用来表示逻辑运算结果中“1”的个数为奇数还是偶数,一般规定“1”的个数为奇数,则 $P=1$,“1”的个数为偶数,则 $P=0$,但不同型号的机器规定亦有不同。

(5)溢出标志位 OV

在有符号的二进制算术运算中,如果其运算结果超过了机器数所能表示的范围,产生溢出时置 $OV=1$,反之 $OV=0$ 。

(6)辅助进位位 AC

辅助进位位亦称半进位标志位 H。当两个 8 位数进行加、减运算时,若 D_3 位向 D_4 位产生进位或借位,则该标志置“1”,否则置“0”。这个标志用于 BCD 码运算的十进制调整,供计算机判断使用。

4. 寄存器组

CPU 内部有一组寄存器,它相当于 CPU 里的小存储器,用于暂时存放数据、地址等信息。它一般分为通用寄存器组和专用寄存器组。

每种 CPU 的寄存器组均有不同,但对用户却十分重要。用户可以不关心 ALU 的具体构成,但对寄存器组的结构和功能必须清楚,这样才能充分利用通用寄存器的特性,简化程序设计,提高运算速度。

5. 程序计数器 PC

程序计数器 PC 又称指令地址指针,PC 总是指向要执行的下一条指令的地址;当程序是以指令的形式存放在内存中一个连续的区域中时,程序顺序执行,每取出一个指令字节,程序计数器便自动加“1”,即 PC 总是指向下一条指令的地址;当程序执行转移、调用或返回指令时,其目标地址自动被置入 PC,程序运行便产生转移。

6. 堆栈及堆栈指针

堆栈是用户在数据存储器中划分出的一个专用区域,用来存放一些数据,实际上是一个数据的暂存区。这种暂存数据的存储区域由堆栈指针 SP 中的内容决定,它有三个主要特点:

(1)按照顺序向栈顶写数据,按顺序从栈顶读出数据。

(2)SP 始终指向栈顶。

(3)堆栈的两种操作,压入(PUSH)和弹出(POP)往往成对进行。所谓压入就是将数据写入堆栈,弹出则是从堆栈中读出数据。

7. 指令寄存器、指令译码器和命令发生器

指令寄存器接收并寄存从存储器中取出的指令码,它指明 CPU 将执行何种操作。指令的操作码部分送指令译码器 ID 进行译码,产生相应的控制信号或脉冲,并送命令发生器,在定时时钟节拍脉冲的作用下,发出各种控制命令,使 CPU 完成指令所规定的操作。

二、微型计算机的构成

微型计算机以 CPU 为核心,通过总线将存储器、输入/输出接口电路等部件连接在一起。如图 1-3 所示。

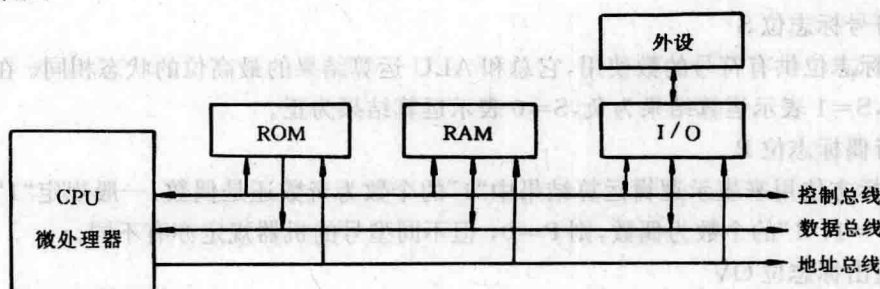


图 1-3 微型计算机框图

如果根据不同要求配备外部设备、电源及辅助电路,并根据系统要求在存储器 ROM 中驻留系统软件,就构成了微型计算机系统。

1. 总线

由图 1-3 可见,整个计算机采用了三总线结构,所有功能部件都连接在总线上,各个部件之间的数据和信息都通过总线传送。换言之,总线是将多个装置或功能部件连接起来,并用来传送信息的公共通道。实际上,总线就是一组导线,导线的数目取决于微处理器的结构,总线有三种类型:

(1)数据总线 DB

数据总线用来在微处理器、存储器以及输入/输出接口之间传送数据。如 CPU 可通过数据总线从 ROM 中读出数据,通过该总线对 RAM 读出或写入数据,亦可把运算结果通过 I/O 接口送至外部设备等。微处理器的位数与外部数据总线的位数一致。数据总线是双向三态的,数据即可从 CPU 中送出,也可从外部送入 CPU,通过三态控制使 CPU 内部数据总线与外部数据总线连接或断开。

(2)地址总线 AB

CPU 对各功能部件的存取是按地址进行的,地址总线用来传送 CPU 发出的地址信息,以存取被选择的存储器单元或 I/O 接口电路。地址总线是单向三态的,只要 CPU 向外送出地址即可;通过三态控制可使 CPU 内部地址总线与外部地址总线连接或断开。地址总线的位数决定了可以直接存取的存储单元(或 I/O 口)的最大可能数量(即容量)。

(3)控制总线 CB

控制总线比数据总线与地址总线复杂。可以是 CPU 发出的控制信号,也可以是其它部件

送给 CPU 的控制信号。对于某条具体的控制线，信号的传送方向则是固定的，不是从 CPU 输出，就是输入到 CPU。控制总线的位数与 CPU 的位数无直接关系，一般受 CPU 的控制功能与引脚数目的限制。

计算机采用三总线结构，不仅使系统中传送的信息有条理、有层次、便于进行检测，而且其结构简单、规则、紧凑、易于系统扩展。只要其它功能部件符合总线规范，就可以接入系统，从而扩展系统功能。但采用总线结构后，某一时刻，一种总线上只能传送一组信号，这就必须使用三态缓冲器。

2. 三态缓冲器

(1) 单向三态缓冲器

单向三态缓冲器电路如图 1-4 所示，真值表如表 1-1 所示。当三态控制端 TSC 为低电平时，即 $TSC=0$ 时，为使能允许控制，此时“ $D \rightarrow Y$ ”。若 $D=0$ ，则“或非门”2 输出高电平，“或非门”1 输出低电平，T1 管截止，而 T2 管导通， $Y=0$ ；反之，若 $D=1$ ，则 $Y=1$ 。数据单向传送。当 $TSC=1$ 时，为使能禁止，“或非门 1”“或非门 2”均输出低电平，使 T1、T2 管截止，Y 呈高阻状态，相当于断开。

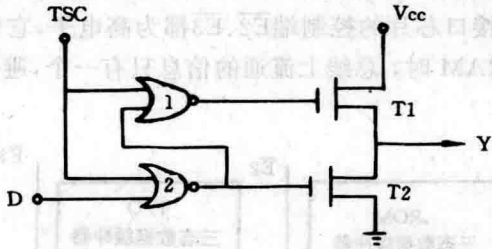


表 1-1 真值表

TSC	D	Y
0	0	0
	1	1
1	0	高阻态
	1	高阻态

图 1-4 单向三态电路图

其逻辑符号如图 1-5 所示。

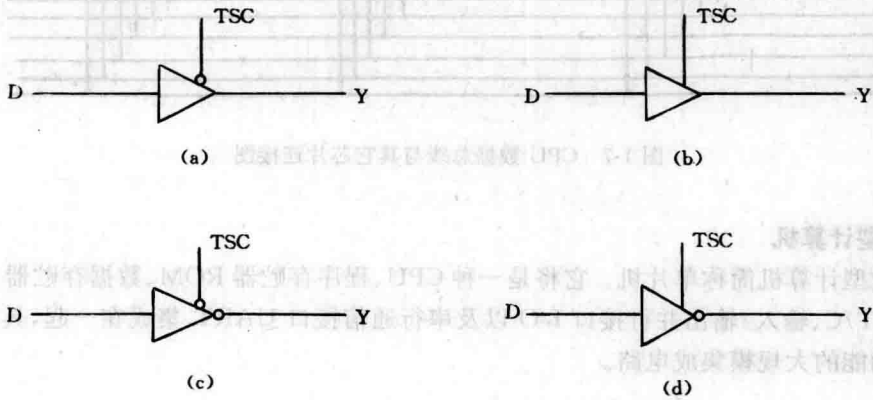


图 1-5 单向三态电路逻辑符号图

图 1-4 单向缓冲电路和逻辑符号如图 1-5(a)所示，为低电平使能，而高电平禁止。图 1-5 (b)所示符号为高电平使能，而低电平禁止，它们都是三态同相缓冲器，两者区别仅仅在于三