

WEICHULIJIWENYUDA

微处理机问与答



徐景村 朱胜利 申良凤 编

山东教育出版社

微处理机问与答

徐景村
朱胜利 编
申良凤

山东教育出版社

1992年·济南

鲁新登字2号

微处理机问与答

徐景村 朱胜利 申良凤 编

*

山东教育出版社出版

(济南经九路胜利大街)

山东省新华书店发行 山东新华印刷厂印刷

*

787毫米×1092毫米32开本 6.25印张 129千字

1993年7月第1版 1993年7月第1次印刷

印数 1—1,100

ISBN 7—5328—1517—7/G·1355

定价 2.60 元

前 言

当今世界已进入信息的时代，各类信息的收集、处理和传递都离不开电子计算机，所以计算机教育是一项既面对现实又面向未来的教育，是迎接经济竞争和新技术革命挑战的重要教育，目前它已经普及到中学的教学当中。现在关于计算机语言、计算机编程等软件方面的书很多，而关于计算机的结构原理等硬件方面的参考书则较少，尤其是这方面普及性的书则更少。鉴于这一情况，我们编写了本书。本书可做为大专院校学生及高中生学习计算机的参考书及其他计算机爱好者的自学读物。

微处理机是计算机及许多工业设备中必不可少的重要组成部分，它同传统的逻辑系统相比，其主要优点仅仅是改变控制它的程序指令，就可改变它的工作模式或工作状态。前几年出现的单片微机，虽然体积小，价格低廉，但能提供很复杂的控制功能，特别适用于处理大批数据的场合。

本书就微处理机的硬件方面作了介绍，前面几章讲的是微处理机的内部结构和操作，后面几章介绍的是微处理机和外部的接口问题，之所以介绍这方面的内容是因为接口是微处理机发挥功能所必不可少的重要组成部分。在第六章“输入/输出”，第七章“支持电路”中，我们结合流行的PC计算机，加进了一些新内容。这些内容介绍的是PC计算机中的

CPU所需的部分支持电路，如8255并行输入输出电路、8253定时器电路、8259中断管理电路、8237DMA电路等。最后介绍了一下微处理机系统的硬件开发、软件开发及应用。

编 者
1992年5月

目 录

第一章	概述.....	1
第二章	存储器.....	25
第三章	判断、分支和寻址方式.....	53
第四章	算术和逻辑.....	63
第五章	子程序和中断.....	88
第六章	输入/输出.....	101
第七章	支持部件	125
第八章	硬件开发	172
第九章	软件开发	182

第一章 概 述

一、什么是微处理机？

微处理机是一种大规模集成电路（LSI），这种电路含有数字计算机的中央处理单元（CPU）所需的复杂的数字逻辑电路。一组或两、三组大规模集成电路元件构成的CPU也可称为微处理机。

微处理机本身用处不大，必须在它周围配上其它电路才能组成一台可工作的数字计算机。图1—1表明了组成一台数字计算机的基本单元，这些计算机可以是象ZX81那样的小微机，也可以是象DECVAX II 那样的大型主机架计算机。

计算机的核心是含有各种逻辑电路的CPU，这些逻辑电路控制着计算机的时序和操作，以及执行计算或逻辑功能。

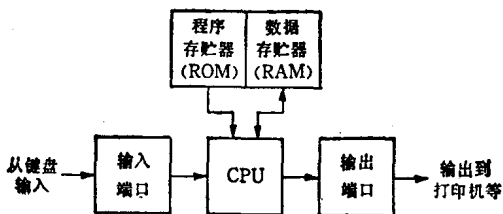


图1—1 数字计算机的一般结构

除CPU之外，计算机还有存贮部件，它们用于存放待处理的数据以及存放称为程序的指令序列，这些程序告诉计算机做什么。

计算机必须能与外界交换信息，这一点是通过称为“端口（port）”的输入输出通道来实现的。这些端口允许计算机从各种输入源接受数据，例如开关、键盘等。这些端口也输出数据给打印机、绘图机或视频显示器（VDU）。

二、微计算机与微处理机的区别

微计算机习惯上指使用微处理机作它的CPU的小型计算机系统。微计算机一般使用若干块称为“芯片”的集成电路块，每一块“集成电路”都制作在一小块硅片上。有时微计算机的所有部件都制作在一块集成电路上，这类计算机称为“单片微计算机”，以便与普通微处理机区别开来。

一台单片微计算机包括构成CPU的控制和计算电路，一些贮存数据和指令的存贮器，以及一组与外界交换信息用的输入输出信号线。单片微计算机最理想的是用于许多小型专用设备中。因为它们硬件非常简单并且可减少产品的尺寸和成本。

单片微计算机应用的一个例子是人们所熟悉的袖珍电子计算器，其它的应用还有电子玩具和家庭用具的控制装置等。

三、什么是bit（位）？

bit是一位二进制数字（binary digit）的缩写，也是微处理机所处理的最小数据单位。

一个bit拥有0和1两种状态之一。“0”状态另外可采用的名字指“低（low）”或“假（false）”。同样地，“1”

状态指“高 (higt)”或“真 (true)”。

一个数据位采用两个状态适合于电子系统。例如,在一台微处理机中,一个状态表示开关合上,则另一个状态就表示开关断开。采用这两种状态信号的逻辑称为模二 (binary) 逻辑,所有通用的微处理机和微计算机都采用这种逻辑。

在典型的微处理机中,“0”状态表示信号电压在 $0 \sim 0.5V$,而“1”状态表示信号电压在 $3 \sim +5V$ 。

四、什么是数据字?

单个bit仅能表示数字0和1,为了表示高于1的数字,可将一组bit组织在一起构成一个数据字。如果我们用4个bit构成一个字,把这四个bit看作一组时,就会有16种不同的0和1状态的组合。这16种组合状态可代表16个不同的数字,例如代表 $0 \sim 15$ 。

任意个bit可组合在一起构成一个数据字,并且bit的数目越大,该数据字可表示的不同的数字范围也越大。这是因为单个bit有两种状态,而每增加一个bit就使可组合的数字加倍。于是8个bit可提供256个组合值,9个bit可提供512个组合值,16个bit能提供的组合值超过65000个。

在微处理机系统中,字长通常取4、8、16和32个bit,但也有少数微处理机采用12个bit长的数据字。

五、什么是字节?

8个bit的数据字称为一个“字节 (byte)”。通常术语byte并不专指8bit的数据字。

一个字节可表示 $0 \sim 255$ 之间的任何一个数字,字节还习惯地表示文体符号的字长。常用的文体字符中有96个大写符号和小写符号,这些符号包括数字和专用的记号。此外还

有32个不同的控制代码，总共有128个字符码，这些字符码可由一个8位的字节中的7位来表示。

nybble或nibble一般习惯指一个4位的字(半个字节)。这个术语也可表示一个字中4位的段。因此一个字节可分成高半字节和低半字节。

六、怎样编制程序指令？

处理机的指令是由一串数字组成的，这些数字以二进制的形式放在微计算机系统的存贮器中。每条指令都含有一个操作码(opcode)，这些操作码规定了当指令执行时，CPU必须完成的操作类型。典型的操作如LOAD(传送)，STORE(存数)，ADD(相加)，INCREMENT(增1)等等。

每个不同的操作码由存贮器中一个唯一的数字表示，这样我们就可以用96表示LOAD指令，而97表示STORE指令。

除操作码外，大部分指令还有一个操作数码，这个操作数码在存贮器中是紧随在操作码后存放的，它占一个或两个字节，这个操作数码可以代表该指令所要处理的数据或者代表一个存贮单元位置的地址，在该存贮单元中存放的或者是某个数据，或者是另一条指令。

虽然指令操作码只是存贮器中的一个数，但编写程序时实际上应用助记符号来代替它，这样可使程序更易为程序设计员理解。于是把“一个数送进累加器”的操作码可以写成LDA。类似地，与该操作码有关的数据和地址也可以用一名字代替而使程序更易理解。一般的概念如图1—2所示，该图给出了一个助记语言表和实际的存贮内容，这是取

自Motorola MC6800微处理器中的一段程序。

助记语言程序

```
LDAA  数据1  将数据1送入累加器A中
ADDA   #5    加5到累加器
ASLA                   将累加器A向左移动1位
SUBA  数据2  从累加器中减去数据2
STAA  结果    将累加器中的内容存到RESULT 存
                贮单元中
```

数据1贮存在0211单元，数据2存在0212单元，而结果存储在0213的单元。

程序在存储器中的存放如图1—2所示。

存储地址 (十六进制)	存储内容 (十六进制)	
0100	B6	LDAA 操作码
0101	02	地址 0211
0102	11	
0103	8B	
0104	05	ADDA 操作码
0105	48	数据5
0106	B0	ASLA 操作码
0107	02	SUBA 操作码
0108	12	
0109	B7	
010A	02	STAA 操作码
010B	13	地址 0213

图1—2 表示6800型微处理器的机器数码程序的一部分，以及它在存储器中是怎样存放的

在通用微处理机中,可以有50~200个不同的操作码来定义不同的操作。实际上一条指令(例如LOAD)可有几种不同的表示方法。每一种表示法都有不同的操作码并且用不同的方式来执行。

多数处理机,都采用8位字长的指令操作码,但也有些处理机采用16位字长的程序指令。在这些16位处理机中,操作码仅占一条完整指令字的一部分,指令字的其它部分可以定义各种选择条件以及提供地址信息。

七、字长怎样影响微处理机?

对于象硬币处理机、玩具、计算器和家庭用具控制器这类简单的控制来说,4位字长是很方便的。这样长度的一个字能迅速提供从0~9的数字并可附加5或6个可定义计算机要完成的操作类型的代码。许多单片微计算机(例如德克萨斯仪器公司的TMS1000系列)虽然它的指令为8位字长,但是数据码却是4位的。

与文字信息处理有关的应用以及通用台式终端或个人计算机,一般使用8位字长的数据字。许多流行的微处理机,例如Intel 8085, Motorola 6800和6809系列, zilog Z80 和 Mos Technology 6500系列,其数据和指令都采用8位字长。

当处理机用于大系统或具有非常复杂计算的场合时,采用16位或32位字长是有利的。

采用较长字的一个优点是:8位处理机的每条指令通常用2或3个字节,而16位或32位机只需用1个字就可以了,这使得在取指令和执行指令时可以节省时间。总之,机器的字长越长,则运行的越快并且越有效。

微处理机或微型计算机的内部组成编排，称为它的“结构”。微处理机的逻辑是非常复杂的，因而一个结构图通常仅能表示逻辑系统的主要功能块。结构图还标出可被程序中的指令使用的所有工作寄存器。

图 1—3 表明了一种典型通用微处理机的结构，类似的结构被用于象APPLE、PET或sinclair ZX81 这样的通用计算机系统中。

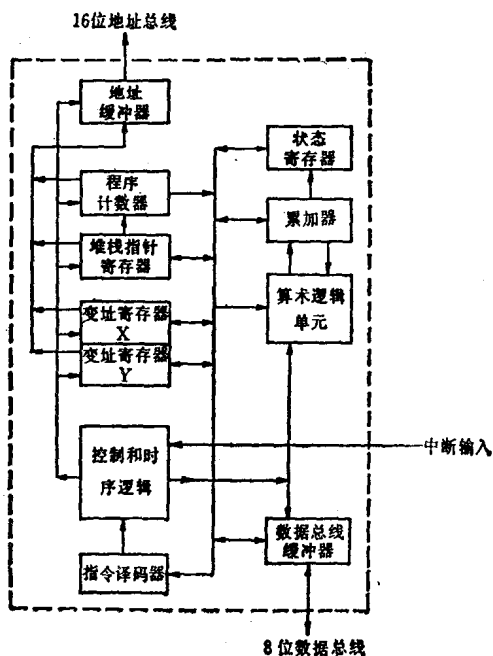


图 1—3 典型的 8 位微处理机的内部结构

逻辑可大致分成控制操作顺序的部分和通过执行算术和逻辑运算来处理数据的部分。这两部分关系密切并且在某些情况下它们的功能重迭。

控制部分的核心是一个称为程序计数器的寄存器。有必要解释一下，寄存器是一种逻辑电路，该电路用来保存一个数据字内各位的状态。程序计数寄存器用于保存存贮单元的地址，该存贮单元中含有要执行的下一条指令的操作码。在执行期间操作码保存在一个指令寄存器中，CPU内的指令译码器为CPU执行必要的操作提供了所需的逻辑时序。

控制系统的其他部分包括一个堆栈（stack）或一个堆栈指针寄存器，一个变址寄存器或数据指针寄存器和提供中断操作的某些逻辑（所谓中断就是用外部信号来打断程序的正常运行）。

数据处理是由算术逻辑单元（ALU）完成的，ALU工作中用到一个叫累加器的寄存器。某些处理机（如Motorola 6800系列）有两个累加器。与累加器有关的还有状态寄存器，状态寄存器中的各位用来指示处理机的状态。

CPU、存贮器和输入/输出接口间的通信联系是通过一系列并行的总线（buses）来实现的。通常有三组总线，即数据总线、地址总线和控制总线。数据通过数据总线去CPU或者从CPU来，而地址总线传递来自CPU的地址信号，用这个信号选择一个特定的存贮单元或一个输入/输出通道，并且将此通道与数据总线相连。控制总线（正如它的名字所表示的那样）用于在系统的不同部分间传递一组控制和时序信号。

图1—4所示为一种典型的单片微型计算机结构，这种

结构可用于袖珍计算器。该单片机内设有程序存储器和数据存储器，还有几个用来处理输入和输出信号的附加寄存器。

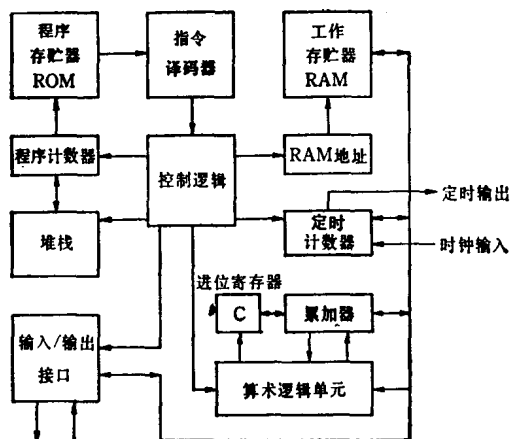


图 1—4 单片微处理机的典型结构

图中所示另一部分逻辑是时钟脉冲发生器，由它提供该微处理机所需的时序信号。图中的地址寄存器是用来驱动地址总线的。许多单片微计算机内有若干个独立的地址寄存器以及独立的用于程序存储器和数据存储器的总线系统，数据地址可通过程序指令来提供。

八、数据总线是如何控制的？

因为数据即可以送到CPU，也可以来自CPU，所以，正确地控制数据的传递方向很重要。这一点是通过来自CPU的读/写控制线来完成的。当一个数据字从CPU送出时，P/W线处于写状态，此时，CPU驱动数据总线。而当一个

字被送到CPU时，则R/W线处于读状态。此时，或者是存储器或者是输入/输出设备将驱动数据总线。

大多数微处理机采用三态逻辑电路驱动数据总线。除了0和1逻辑电平之外，这种逻辑电路还有第三种状态，即使CPU与数据总线脱离联系的状态。处于第三种状态时，CPU释放了数据总线，此时数据总线可被其它设备使用。这种方案很有用，特别是在几个微处理机一起操作并且共用同一组数据总线时。

三态逻辑门电路的组成，工作原理及表示符号是什么？

三态逻辑门电路的电路组成原理如图1—5(a)所示，它的表示符号如图1—5(b)所示：

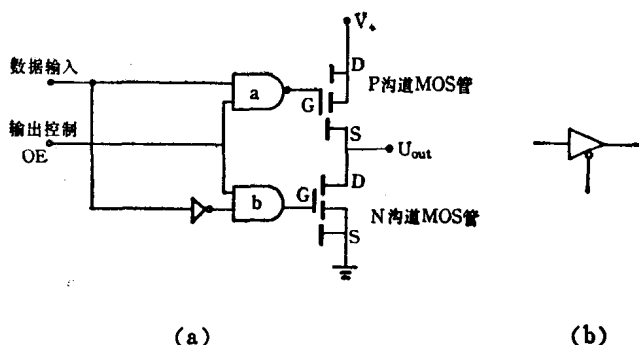
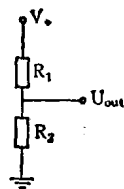


图1—5 (a) 三态门电路组成 (b) 三态门表示符号

工作原理是：由P沟道MOS管和N沟道MOS管组合成一个CMOS电路，只要输出控制信号OE为低电平，不论数据输入为何种电平，与非门a输出总为高电平，与门b输出总为低电平，此时两个MOS管的沟道都不会形成，它们的漏

极D和源极S之间都呈现高阻抗状态,对于电源回路而言,两个MOS管相当于两个无穷大的电阻串联。如图1—6所示,图中 R_1 、 R_2 相当于两个MOS管的漏源极之间的电阻。

$$U_{out} = \frac{V_+ \cdot R_2}{R_1 + R_2}, \quad V_+ \text{一般是} +5\text{V}, \quad U_{out}$$



取值范围是 $0.8\text{V} < U_{out} < 3\text{V}$ 。

当OE为高电平(一般高电平为 $> 3\text{V}$ 图1—6

的电平)时,如果数据输入也为高电平,那么与非门a输出低电平,此时满足PMOS管沟道形成的条件,P管导通, V_+ 电压加到 U_{out} 输出端,与此同时与门b输出为0,NMOS管不

能形成沟道,由此 $R_1 \approx 0$, $R_2 \rightarrow \infty$, $U_{out} = \frac{V_+ \times R_2}{R_1 + R_2} \approx V_+$ 。

即输出信号反应了输入信号的状态。

当OE为高电平时,如果数据输入端输入的信号电平为低电平,低电平一般认为是0电平,此时与非门a输出高电平,P沟道管不能形成沟道,DS之间呈现无穷大电阻,即 $R_1 \rightarrow \infty$,而与门b输出高电平,NMOS管满足沟道形成的条件,所以NMOS管导通,NMOS管的DS之间相当于短路,即 $R_2 \rightarrow 0$,此时 U_{out} 不难看出约为0V。再次说明输出信号的状态反应了输入信号的状态。

三态逻辑门电路常用于数据总线中或地址总线中,用以传送数据信号或地址信号。常见的三态逻辑门电路的型号有74AS230, 74AS231, 74LS240, 74LS241—245。根据功能不同,有的作三态总线驱动器用,有的作三态总线变换器用,还有的作三态总线传送器用。图1—7所示为常用的三