

934865

仪表和控制系统中的 微处理机

原著[英] S.A 莫尼

杨安禄 陈长龄 译

张世箕 杨安禄 审校



电子科技大学出版社

934865

TP368.1
34477

TP368.1
34477

仪表和控制系统中的 微 处 理 机

原著[英] S. A. 莫尼
杨安禄 陈长龄 译
张世箕 杨安禄 审校

电子科技大学出版社

• 1991 •

Microprocessors in Instrumentation and Control

[英] S. A. 莫尼

仪表和控制系统中的微处理机

杨安禄 陈长龄 译

张世箕 杨安禄 审校

*

电子科技大学出版社出版

(中国成都建设北路二段九号)

电子科技大学出版社印刷厂印刷

四川省新华书店经销

*

开本 850×1168 1/32 印张 8.75 版面字数 214千字

版次 1990年12月第一版 印次 1991年7月第二次印刷

印数 1—1200册

中国标准书号 ISBN 7-81016-297-7/TP·24

(15452·140)

定价: 4.30元

内 容 提 要

本书全面地论述了微处理机测试与控制系统的基本问题：微处理器结构、算法与编程、通用并行/串行输入输出接口、计数与定时器、中断与实时操作、模拟输入输出通道与数据采集、系统硬件与软件开发，特别是数字控制算法推导更具有新意。

全书内容丰富，自成体系，可供测试与控制工程技术人员参考，能作为大学非自动控制专业教学用书，也可供有关管理干部阅读。

译者序

由于微处理机具有成本低、体积小、功耗少、可靠性高和使用灵活等特点，而在信息自动化同过程控制相结合方面获得广泛应用，使得计算机测试与控制系统由集中控制向分散型微处理机控制迅速发展。微处理机在集总分布式测试与控制系统中的地位越来越重要，然而与迅猛发展的商业性应用相比，工业技术领域的微处理机实时测试与控制应用仍然是被忽视的课题。微处理机实时测试与控制要求用户兼有硬件和软件的知识，这方面人才的培养也还得不到应有的重视，因而很缺乏。我们翻译此书，正是为了向广大读者推荐急待解决的被忽视的课题。

本书译自[英国]工程师学会理事会技术工程师、[英国]电子工业工程师学会会员、英国计算机学会会员 S. A. 莫尼 (Money) 的著作 “Microprocessors in Instrumentation and Control”。内容涉及微处理机实时测试与控制系统的设计与编程、数据采集、数字控制算法、系统硬件与软件开发等基本问题。全书论及的问题较为全面，自成体系，由浅入深，循序渐进，几乎不借助其它参考书籍，一名对微处理机有了初步了解的读者就能读懂本书，进而付诸实际应用。

该书内容与特点使其适用面较宽，可供非自动控制专业人员学习，也能为自动控制专业人员提供参考；不但是工程技术人员技术实践的参考文献，也可作为有关管理干部的自学读物，还能选作大学非自动控制专业本科、专科生的教材或教学参考书。

本书由杨安禄翻译第一至五章，并审校第六至八章，陈长龄翻译第六至十二章，张世箕教授审校其余章节，并对全书进行统稿。

张世箕教授对全书提出了许多极宝贵的修改意见，陈玉珍工程师对全书做了许多绘图和文字修整工作，在此一并表示衷心的感谢！

限于译者的水平，书中还可能有不少漏误之处，恳切地希望得到批评指正。

杨安禄 陈长龄
于电子科技大学自动化系
1990年12月

前 言

近年来，微处理机的广泛应用，使我们可以生产出各种通用而且可靠的数据采集和数字控制系统。这些系统的早期设计，由于采用大量的小规模集成逻辑器件，使得系统往往过于庞大且通用性和实用性较差。尽管能利用小型计算机来实现一些数字控制，但是微处理器的出现却表明了控制器设备能造得更加紧凑，而不必依靠一套中央计算设备。

本书就微处理机在数据采集和控制系统中的应用做了详细的阐述。第一章简要介绍微处理机系统的基本硬件构成及工作原理，第二章将讨论针对各种任务的编程原理和技巧。CPU较为重要的任务之一是执行逻辑和算术运算，在第三章研究利用微处理机进行算术运算的原理，其中包括介绍浮点算术运算以及实现数字控制系统所需的积分、微分等函数运算。

在实际应用中，微处理机系统必须与外界通信，在第四章和第五章我们将研究微处理机系统的数字式数据输入和输出的有关技术。第四章将讨论各种并行输入/输出方案，包括目前许多仪器常用的IEEE-488通用仪器总线（GPIB）方案。第五章讨论用异步和同步两种数据格式进行数字信息串行传输。

对于数据采集和控制系统来说，许多输入和输出量是模拟形式的，第七章研究信号从模拟到数字以及从数字到模拟形式的变换技术，还将讨论设置和测试模拟输入和输出通道所需的软件设计。

利用计算机系统进行数据采集，需对输入信号进行离散采样，其中可能出现“混叠”的问题，即在采样过程中有可能产生

虚假信号，而微处理器很难将虚假信号与真实输入信号区分开来，这个问题将在第八章讨论，在第八章还研究利用微处理器建立数据采集系统的一些编程原理。

过去，大多数控制系统采用模拟式反馈技术实现控制器的功能，有了微处理器就可以采用数字技术来实现控制功能，从而研究出更加灵活易变的控制方案，只需给微处理器的控制算法方程输入新的数据就能改变控制方案的各种特性。第九章将讨论实现这类系统的基本原理。

在微处理器的许多应用中，计算操作必须是实时操作。为了更有效地利用CPU，要求有某种手段能暂时中断正常的程序序列而立即处理外部事件。微处理器器件一般都具备一些“中断”功能，因此能够满足上述要求。第十章讲述这类硬件中断方案的工作原理。

除非给微处理器编程使之做所希望的工作，否则微处理器本身毫无用处。给微处理器编写程序通常称之为软件开发，它包括程序的总体设计，编写实际的程序代码，以及最后对软件进行测试并修正操作中的错误。这个工作可以利用CPU指令助记符语言来完成，但更好的方法是使用高级语言，这样能使编程者集中精力于程序的作用而不必考虑CPU操作的细节。第十一章将讨论软件开发的原理和一些常见的高级语言的使用。

微处理器系统必须由硬件来执行程序软件所要求的功能，在早期的微处理器中，由一系列集成电路组合来进行硬件设计。今天，则用制造好的模块电路板，这些电路板能提供以微处理器为基础的系统所需要的各种不同的组成部分，从而简化硬件设计。对许多应用而言，则利用个人或小型计算机系统来实施所需的以微处理器为基础的计划，在选择硬件时所必须考虑的一些原则，将在第十二章进行讨论。

S.A. 莫尼

目 录

前 言

第一章	基本原理	(1)
第二章	编程技术	(19)
第三章	算术运算和逻辑运算	(43)
第四章	并行输入/输出	(65)
第五章	串行输入/输出	(93)
第六章	计数与定时操作	(118)
第七章	模拟输入/输出	(135)
第八章	数据采集	(171)
第九章	控制系统	(197)
第十章	中断与实时操作	(220)
第十一章	软件开发	(241)
第十二章	硬件设计	(260)

第一章 基本原理

近年来由于微处理器器件和微型计算机的发展，在电子仪器和控制设备领域内能设计出更先进的多功能设备。以往用模拟电路技术来设计这种系统，它能完成的功能是十分有限的。自六十年代以来，由于集成电路技术的发展，以及种类繁多的数字逻辑集成器件相继问世，使在仪器设备中引入数字技术成为可能。采用数字技术的优点是仪器设备或控制单元能获得更高的精度和范围更广的功能。

采用标准集成电路逻辑器件时，数字化电子设备的外观实体尺寸显得比较庞大，采用更复杂的专门设计的集成电路能降低所需器件的数目，从而使仪器的结构更加紧凑。对于批量不大的产品，若采用专门设计的电路，成本可能很高，因而这种方法是不经济的，除非设备需要大量生产。

另一种方法是用一个微处理器和少量标准支持芯片来代替专门设计的电路。采用微处理器的优点在于它是一种通用器件，价格低廉，并且可以通过编程来适应极不相同的各种仪器和控制的应用。其结果是，电子设备可以由少量的复杂的集成电路组成，优点是能够改变系统的操作以满足用户的各种特殊要求。

如今许多家用设备也以微处理器作为控制系统的一部分，其中一些采用相当简单的控制方案执行一段预先编制的操作程序。例如在一台洗衣机中，洗衣水必须加热到一定的预置温度，然后鼓轮转动，到了预定的洗涤时间加热器断开并进行一次喷淋处理，最后多半是完成甩干操作。在早期的自动洗衣机中，由马达驱动的开关闭件来控制洗衣过程，水温由一个简单的温控开关来

控制。现代的洗衣机则要求它具有各种洗涤程序，为了适应这些要求，洗衣机的机械开关系统就会很复杂，生产成本将很高。以微处理器为基础的系统允许用户预置广泛变化的各种洗涤程序，还可以用作温度控制和其他各种监控功能，而且成本极为低廉。

在录音机中，微处理器控制着卷带放带的过程，还监测磁带的速度，磁带的张力，磁带结束传感器以及其他一些运行参数，并能自动改变系统的工作状况以补偿工作参数的任何变化。现在微处理器已经用于控制汽车引擎运转之类的工作。采用了微处理器就有可能使引擎工作在最佳条件以节省燃料，平稳运行，而且其整体性能比早期采用的模拟式控制好得多。

在实验室中，大多数老式仪器都采用模拟测量技术，其测量结果在表头或图上显示。如果需要精确的结果，这类系统则达不到。采用了数字技术的测量仪器，可以达到更高的精确度，而且使用方便。含有微处理器的仪器十分灵活，具有很多特点，比如自动选择量程来适应被测信号的大小等。

测量结果往往需要按某种方式进行分析以获得所需的信息。在过去必须由人工采集大量的数据，然后送到主机或小型计算机系统去分析并打印出来。利用现代化的数据采集系统，连同适当的测量仪器，就可以自动采集实验数据并记录在诸如磁带之类的载体上。实验之后，这些记录下来的数据可以直接读进计算机进行分析，节约了大量的时间和可观的人力。如果使用一个以微处理器为基础的仪器系统，那么就有可能在实验进行之中作出所需的部分或全部分析，从而节省了时间，并且无需求助于大的计算机系统。

在过程控制中，我们的系统基本上是如录音机之类的器件所用系统的大型翻版。在一些原材料(比如钢铁或塑料)的处理过程中，要生产出最终的成品，可能需要极不相同的操作序列。这些

过程的每一步本身是相当复杂的，并需要对各种参数进行监视和控制。这里整个过程可以用一台单独的大型计算机来控制，但是若用多台微处理器对全过程分段进行控制，再用一台微计算机来管理整个系统，将会更有效、更经济。

象家用电器所用的微处理器，通常是为特殊需要专门设计的，它们能够大批量地生产，因此比通用控制机或计算机要廉价得多。对于实验室或工业控制和监控来说，以微处理器为基础的大众化家用计算机系统是很有用的。这类机器的优点是价格低廉且通用，一般的用户来编程也相当容易。这类机器同一些简单的接口电路结合起来，就能给出适当的结果，而且比用一套专用的仪器设备经济得多。

本书的重点在于微处理器器件的操作以及它们在仪器和控制领域内的一些应用。

数字逻辑和信号

在现实世界中，我们发现诸如长度、重量等物理量，其量值范围都可以无限制地变化，对于仪器和控制而言，这些物理参数一般都是通过适当的传感器被变成与被测物理参量成正比的电压或电流，这些电信号称为“模拟”信号。

模拟信号可以利用放大器和滤波器等进行处理，其结果可以在表头上显示或绘制出来，甚至有可能制造出使用模拟信号的计算机。模拟系统的主要问题是在处理电路中难以维持高准确度和高稳定性，模拟方案往往也相当死板，假若在它要完成的分析和计算中要作一点改动，通常都要改接电路以构成新的组态。

替代模拟法的另一途径是采用仅可能具有“通”和“断”两种状态的简单电路元件。这种元件仅能表示“0”和“1”两个数，不过采用这些简单开关电路的一个组合状态则可以表示任何

数值。这样，若取三个开关，如图 1.1 所示，便可能有 8 种通/断组合，这些组合可以表示 0~7 这 8 个数值。把开关的数量增加到 4

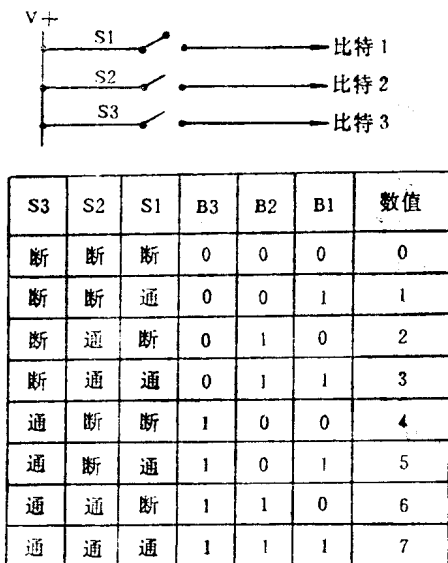


图 1.1 用通/断状态的组合来表示数值

个，就能给出 16 种组合，用 5 个开关则允许有 32 种组合。事实上，在一群开关中，每增加一个开关，可能具有的组合数目便翻一倍。

因为每个开关仅有两种可能的状态，这种系统称为“二进制”系统。每个开关代表整个数据的一个数位，称为一个“比特(bit)”(即二进制)。比特的两种状态常称为“0”态和“1”态。在更为熟悉的十进制数中，每个数位有十种可能的状态，其范围从 0~9，而且各位数按从右到左的顺序分别表示个位，十位，百位等等。在二进制系统中，每一位数不是“0”就是“1”，它们从最右边到最左边分别代表 1, 2, 4, 8 等等，如图 1.2 所示。

构成一个二进制数的一组比特通常称为一个“字(word)”，在

微型计算机中，字长可能由4比特(用于小型专用器件)直到32比特(用于大型通用微计算机系统)。虽然早期的微处理器用过12比特字长，但在实际工作中常见的典型字长是4，8，16和32比特。字

比特	比特	比特	比特	比特	比特	比特	比特
7	6	5	4	3	2	1	0
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1

各比特的位权值

例

$$101100 = 32 + 8 + 4 = 44$$

图1.2 在一个二进制数中各比特的数值加权

长决定了在单次微处理器操作中所能操作的数值范围。因此，8比特字可以表示的数在0~255范围之内，而16比特字则扩展了数的范围，给出0~65535的数值。

基本的微计算机结构

作为开头，我们首先来看一看微处理器和微计算机系统的一般构成。事实上，一台微计算机只不过是一台由为数不多的几片复杂集成电路组成的数字计算机系统。在某些情况下，一切关键功能都包括在单片集成电路之中。

典型的数字计算机系统，无论是微型计算机还是大型主机系统，其组织都是如图1.3所示的基本模式。系统的核心是中央处理单元或CPU，它包含了计算机系统中执行算术和逻辑功能以及控制各种操作的定时和排序所需要的复杂逻辑。

除了CPU之外，一个计算机系统还需要某些形式的存储器。事实上这是一个电子文件档案系统，它允许以数字形式把信息的各项存储在一边以备后续工作使用。存储器的一个重要功能是保存称为“程序”的一个指令清单，程序告诉CPU应该做什么工作。

存储器还被用来保存将要被处理的数据或处理数据后所产生的结果。在小型专用微计算机系统中，程序存储区和数据存储区可能是分开的，但在通用系统中程序和数据二者通常都被保存在一个公用的存储器之中。

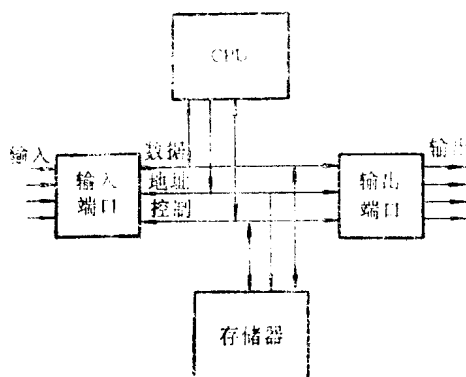


图1.3 一个数字式计算机系统的基本组成
部分的方框图

如果计算机要用于实际应用，那么重要的就是提供某种手段让计算机同外部世界通信，使计算机能够从外部设备接收数据，并且产生输出以便可以显示出来或用以控制外部设备的动作。对于大多数应用来说，重要的是用户能够同 CPU 通信，以便在程序中选择可供随意选用的部分，或者从某个外部来源把程序装进去。CPU和外部世界之间的这种数据或命令传递是通过一系列称为“端口”的输入或输出通道来完成的。

计算机系统的各个部分由三组总线连接在一起。每组总线都是由若干条互连的导线组成，在这些线上能够传送数字数据。一组线称为地址总线，用来选择某个特定的存储单元或输入输出通道，使之与主数据总线接通，以便让数据能够传入或传出CPU。在地址总线上，数据总是从CPU流向系统的其他部分。在单片微计算机中可能有两组分开的地址总线，一组用来选择程序存储器中

的存储单元，另一组则用于数据存储器 and 输入输出端口。

系统的各个部分之间是通过“数据”总线来传递数据的，这是一组双向总线，从而可使数据传入或传出CPU。在这组总线上数据流的方向由CPU本身来管理，而控制信号则被送到存储器或输入输出端口，指示它应该发送还是接收数据。通常用一个完整的数据字在数据总线上同时运动，以并行方式传递数据。按照CPU内部处理的数据字长来对微处理器进行分类，最常见的字长是4、8和16比特。有时候为了减少连线数目，数据总线可以复用，这样一个16比特字就可以分作两个8比特字节用8条线的数据总线来传送。这里虽然数据总线只有8比特宽，然而微处理器仍然属于16比特型。

第三组总线称为“控制”总线，它携带着计算机各个部分之间的各种控制和状态信号，其中一些线传送来自CPU的信号，而另一些线则把一些信号送回CPU。这组总线控制着计算机系统的定时和常规操作。

微处理器和微计算机之间的差别在于前者仅含有CPU，而后者则在同一集成电路封装体中还含有存储器和输入输出通道。微处理器总是需要附加支持电路才能构成一个完整的系统，而单片微计算机自身就能工作。

指令格式

程序告诉CPU该作什么操作，程序是由存储在计算机系统的存储区内的数字清单组成的。每条指令含一个操作码(opcode)，它规定所要进行的操作的类型。有些指令还包括一个操作数(operand)，它规定该指令所使用的数据。大多数情况下操作数都是指明某一个特定的存储单元，在其中能找到所用的数据。

对于8比特通用微处理器来说，操作码一般是一个8比特的数

据字，能提供多达256条不同的指令。如果需要操作数的话，则再由一个或两个8比特数据字构成，它们存放在操作码之后的相继存储单元之中。8比特微处理器中指令的典型存储分布示于图1.4中。这样，对某些指令来说，必须从指令存储器传送三个数据字才能把该指令传入CPU。这种方法是有缺点的，因为会降低计算机的操作速度。

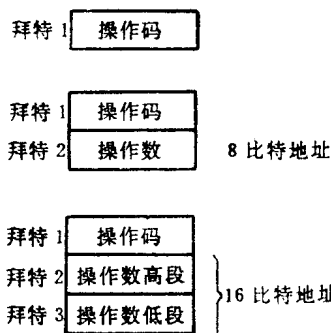


图1.4 8比特微处理器中指令的操作码和
操作数的典型数据分布

数据字宽度为16比特的较大的微处理器，可以把操作码和操作数合并成一个16比特的字，如图1.5所示。这只需从存储器传送一次数据就能建立一条完整的指令。象8比特型微处理器一样，某些指令需要更大的操作数，因而操作码保存在头一个字中，并在其后再增加1个或多个字作为操作数以构成一条完整的指令。

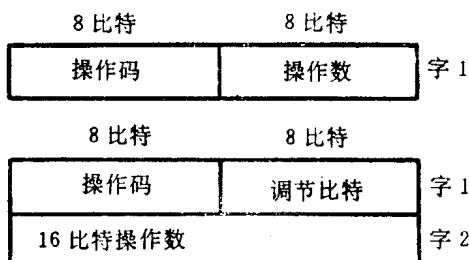


图1.5 16比特微处理器的典型指令格式