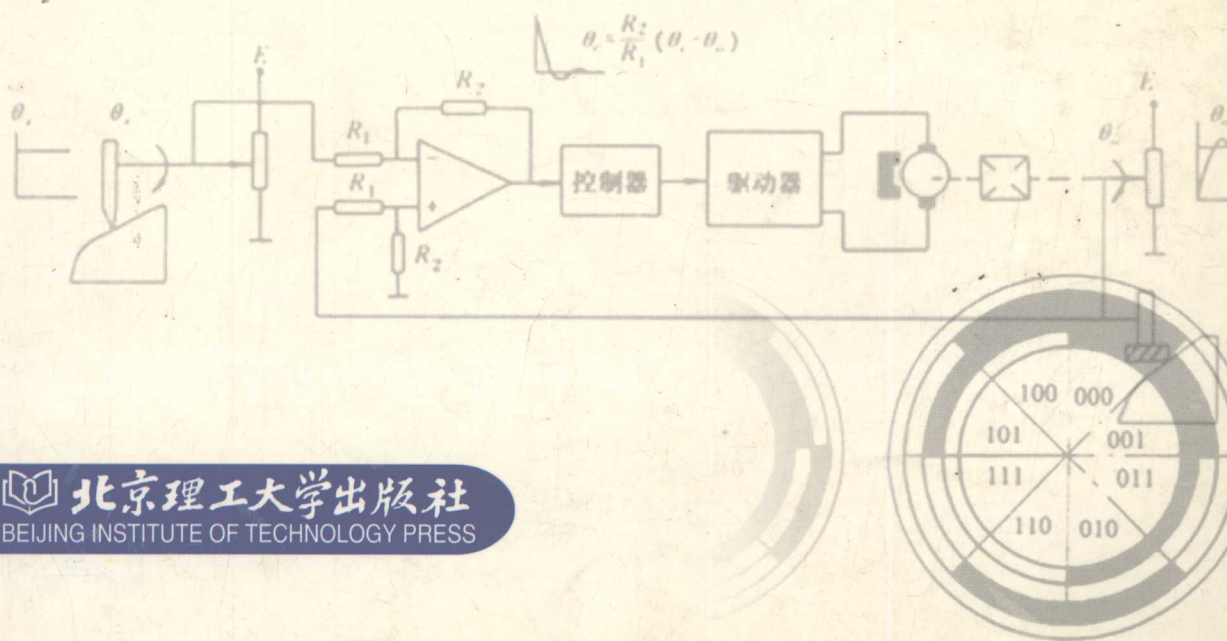


JiSuan.JiKongzhiXitong

计算机控制系统

(修订版)

张宇河 董 宁 编著



 北京理工大学出版社
BEIJING INSTITUTE OF TECHNOLOGY PRESS



计算机控制系统

(修订版)

张宇河 董 宁 编著

 **北京理工大学出版社**
BEIJING INSTITUTE OF TECHNOLOGY PRESS

图书在版编目 (CIP) 数据

计算机控制系统 (修订版) / 张宇河, 董宁编著. —2 版. —北京: 北京理工大学出版社, 2002.9 (2004.1 重印)

ISBN 7-81045-036-0

I. 计… II. ①张… ②董… III. 电子计算机—控制系统—高等学校—教材 IV. TP271

中国版本图书馆 CIP 数据核字 (96) 第 01509 号

出版发行 / 北京理工大学出版社

社 址 / 北京市海淀区中关村南大街 5 号

邮 编 / 100081

电 话 / (010)68914775(办公室) 68912824(发行部)

网 址 / <http://www.bitpress.com.cn>

电子邮箱 / chiefedit@bitpress.com.cn

经 销 / 全国各地新华书店

印 刷 / 北京圣瑞伦印刷厂

开 本 / 787 毫米 × 1092 毫米 1/16

印 张 / 15.5

字 数 / 374 千字

版 次 / 2002 年 9 月第 2 版 2004 年 1 月第 4 次印刷

印 数 / 9501~14500 册

责任校对 / 郑兴玉

定 价 / 20.00 元

责任印制 / 刘京凤

修订版说明

《计算机控制系统》是为自动控制专业本科生编写的必修课教材。本书第一版由张宇河、金钰共同编著，北京工业大学肖春林教授主审，于1996年由北京理工大学出版社出版，在北京理工大学自动控制专业本科生、大专生以及机械电子工程专业硕士研究生教学中使用，并被有关兄弟院校选用。

根据多年来的教学体会和使用者反馈的意见，我们对原书进行修订，修订工作由张宇河、董宁完成。

此次修订保留了原教材最基本部分，删除了部分章节，补充了新的内容。如系统总线和电磁兼容性设计，并针对北京理工大学自动控制专业的特点将内容侧重于计算机控制伺服系统，使教材不仅适用于自动化专业本科生也适用于机械电子工程专业的硕士研究生，内容更加切合工程实践的需要。

修订后的教材仍为九章。第一章概述计算机控制伺服系统的一般概念；第二章通道接口技术和第三章系统总线主要讲解计算机控制系统的硬件设计；第四章到第六章分别介绍实用的数字控制器的设计方法，它们是：PID方法，连续化方法，离散化方法；第七章介绍数控和可编程序控制器；第八章讨论控制系统的电磁兼容性问题；第九章介绍几个应用实例。

感谢北京理工大学胡祐德教授为修订版审稿并提出宝贵意见。

限于我们的水平与经验，书中难免还有错误和欠妥之处，敬请读者和同行们批评指正。

编著者

2002年4月

目 录

第一章 概述	(1)
1.1 伺服系统的一般概念	(1)
1.2 计算机控制伺服系统的组成及工作过程	(3)
1.3 计算机控制伺服系统的结构形式	(4)
习题与思考题	(6)
第二章 通道接口技术	(7)
2.1 计算机对外围通道的控制	(7)
2.2 模拟量输出通道	(19)
2.3 模拟量输入通道	(37)
习题与思考题	(59)
第三章 系统总线	(61)
3.1 总线的一般概念	(61)
3.2 系统总线	(63)
3.3 PC 总线模板举例	(79)
习题与思考题	(85)
第四章 数字 PID 控制器设计	(86)
4.1 数字 PID 算法	(86)
4.2 数字 PID 控制器参数选择	(88)
4.3 数字 PID 的改进算法	(90)
4.4 采样周期 T_s 的选择	(99)
习题与思考题	(101)
第五章 数字控制器的连续设计方法	(102)
5.1 使用连续化设计方法的条件	(102)
5.2 连续化设计步骤	(103)
5.3 $D(s)$ 的离散化方法	(105)
5.4 各种离散化方法的比较	(113)
5.5 设计举例	(116)
习题与思考题	(125)
第六章 数字控制器的离散设计方法	(127)
6.1 解析设计法	(127)
6.2 根轨迹设计法	(142)
6.3 频率响应设计法	(154)
6.4 状态空间设计法	(161)
习题与思考题	(173)
第七章 数控与可编程控制器	(174)
7.1 数控	(174)
7.2 可编程控制器	(183)
习题与思考题	(194)

第八章 计算机控制系统的电磁兼容性设计 (195)

8.1 电磁兼容性的概述..... (195)

8.2 噪声源..... (196)

8.3 噪声抑制技术..... (197)

8.4 数字信号的传输..... (208)

习题与思考题 (214)

第九章 实 例 (215)

9.1 喷砂机器人..... (215)

9.2 随动系统通用数字控制器..... (226)

9.3 飞行仿真头位跟踪视景显示系统..... (231)

附录 Z—变换简表 (238)

参考文献 (240)

第一章 概述

1.1 伺服系统的一般概念

自动控制系统可以分成调节系统和伺服系统两类。调节系统要求被控对象的状态保持不变，一般输入信号不作频繁调节；而伺服系统则要求被控对象的状态能自动地、连续地、精确地复现输入信号的变化。“伺服 (Servo)”一词是拉丁语，指“奴隶”的意思，它包含使系统像奴隶一样忠实地按照命令动作，而命令是根据需要不断变化的，因此伺服系统又叫做随动系统。对于机械运动控制系统，被控对象状态主要有速度和位置，如速度伺服系统、位置伺服系统。本书主要讨论位置伺服系统。

自动控制系统按其构成方式又有开环系统和闭环系统之分。开环系统简单，但不能保证控制质量，被控变量易受外界条件变化的影响。而闭环系统通过引入被控变量的反馈，利用偏差进行控制，使得被控变量紧紧跟踪输入信号，从而保证了控制质量。因此自动控制系统一般采用闭环控制方式。

典型的位置伺服系统有机械制造行业里的机床进给系统，工业机器人关节运动控制系统。数控机床按加工图纸编程，通过插补计算，产生伺服系统的给定信号，控制机床各运动轴伺服系统运行，实现零件加工。由于加工的零件是各式各样的，插补计算出的各进给轴的控制信号的轨迹是很复杂的。工业机器人是在数控基础上发展起来的，由于机器人机构形式的特殊性，使得机器人伺服机构的给定信号的产生要经过复杂的坐标变换。和数控机床一样，各轴伺服系统的给定信号的轨迹也是很复杂的，包含有时间的各阶导数。

在军事上，位置伺服系统用得更普遍。例如防空火力控制系统中的雷达天线的自动瞄准跟踪系统、火炮的自动跟踪系统等。武器伺服系统的给定信号是来自被跟踪的运动目标，而被跟踪的运动目标的运动规律是未知的、机动的，因此武器伺服系统的给定信号是随机的，同样包含有时间的各阶导数。

对位置伺服系统设计的要求，概括起来有以下几个方面：

① 对运动速度和运动加速度的要求。指控制对象应能达到的最大速度、最大加速度、最低速度以及在低速下的运动平稳性。

② 对动态指标的要求。指闭环控制系统在阶跃输入信号作用下输出响应的超调量、过渡过程时间、振荡次数等。也可以是在正弦信号输入作用下控制系统的通频带和振荡度，或者是控制系统的剪切频率、相位裕度和增益裕度。

③ 对系统精度的要求。指当输入为某个固定给定值时，输出与输入之间的最大位置差，称为静态误差；输入为最大速度（等速输入）时，输出与输入之间的位置误差，称为速度误差；输入为最大等加速度时的位置误差，称为加速度误差。

在扰动作用下（如火炮射击时后坐力），控制系统也将产生误差，因此还需要考虑在干扰力矩作用下控制系统的误差。

④ 对环境的要求。指环境的温度、湿度、腐蚀性和防爆性等。

控制系统设计分静态设计和动态设计两部分。静态设计是根据负载要求选择执行机构，

设计驱动器，根据精度要求，选择传感器，构成一个闭环控制系统；动态设计是根据系统品质指标要求设计控制器，使系统满足稳定性、快速性和跟踪精度的要求。伺服系统的稳态精度指标包括静态误差、速度误差和加速度误差。某些设备由于运动范围受限，等速信号给出后，系统还未达到稳态已运动到限位位置，因此速度误差无法检验。即使系统运动范围不受限，由于系统速度最大值受限，等加速输入时，在短时间内速度会达到最大值，系统加速度达不到稳态值，加速度误差也无法测量。因此在伺服系统中常采用等效正弦输入的正弦误差来检验，即输入转角为时间的正弦函数时，达到稳态后，输出与输入之间的位置误差。

设输入为时间 t 的正弦函数

$$\theta_i(t) = A \sin\left(\frac{2\pi}{T} t\right) \quad (1-1)$$

求一次导数

$$\theta'_i(t) = \Omega_i(t) = \frac{2\pi A}{T} \cos\left(\frac{2\pi}{T} t\right) \quad (1-2)$$

求二次导数

$$\theta''(t) = \epsilon_i(t) = -\frac{4\pi^2 A}{T^2} \sin\left(\frac{2\pi}{T} t\right) \quad (1-3)$$

令

$$\frac{2\pi A}{T} = \Omega_m, \quad \frac{4\pi^2 A}{T^2} = \epsilon_m$$

即对最大速度和最大加速度的要求。

于是可以求得

$$A = \frac{\Omega_m^2}{\epsilon_m} \quad (1-4)$$

$$T = \frac{2\pi \Omega_m}{\epsilon_m} \quad (1-5)$$

即可以以振幅为 A ，周期为 T 的正弦输入来检验系统正弦误差，等效检验了系统的速度误差和加速度误差。

一个简化的位置伺服系统如图 1-1 所示。这是个靠模机床的例子，传感探头靠着模型移动，通过电位计检测移动的位置，由伺服系统带动刀具跟踪传感探头并进行工件的加工。

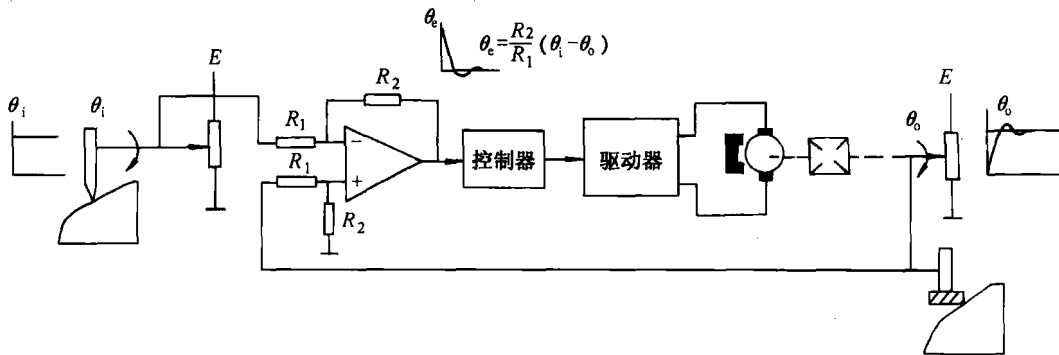


图 1-1 位置伺服系统

刀具位置检测也是电位计。这是一个时间连续控制系统，其工作过程是：将输出 θ_o 反馈到输入端与给定值 θ_i 比较，得到偏差 θ_e ，通过控制器的决策，控制系统以满意的动态过程消除偏差 θ_e ，使 θ_o 跟随 θ_i 。因此时间连续控制系统是随时都在监视误差，随时进行决策，随时进行控制，使控制系统以最理想的路线跟踪输入。

1.2 计算机控制伺服系统的组成及工作过程

利用计算机代替常规的模拟控制器，而使它成为控制系统的一个组成部分，我们把这种有计算机参加控制的系统简称为计算机控制系统。也就是说计算机控制系统是由强调计算机作为控制系统的一个组成部分而得名的。计算机控制系统有时也称为数字控制系统，这是强调在控制系统中包含有数字信号。

引入计算机控制的伺服系统叫做计算机控制伺服系统，也可以称为数字伺服系统。在图 1-1 伺服系统中引入计算机代替误差的求取和控制器的功能，构成计算机控制伺服系统，如图 1-2 所示。由于计算机输入输出只能是数字信号，所以要加入 A/D（模拟量 - 数字量转换）、D/A（数字量 - 模拟量转换）环节。

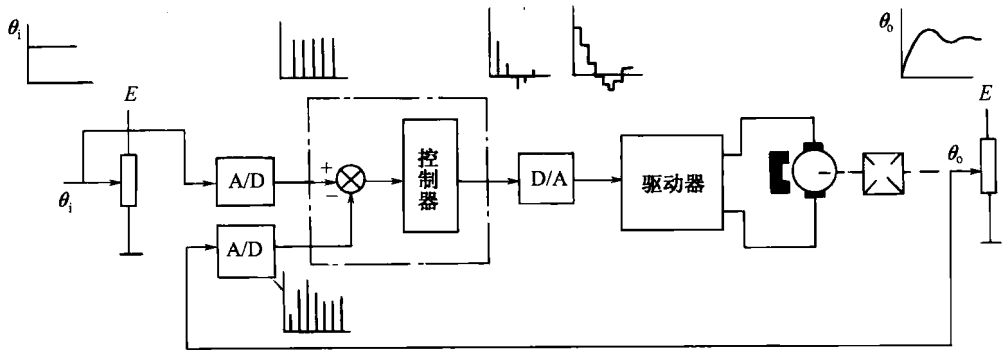


图 1-2 计算机控制伺服系统

计算机控制伺服系统由计算机通过模拟量输入通道（A/D）采集被控对象的状态，与给定值的数字量比较，获得误差，然后经控制器的算法程序进行信息加工，做出相应的控制和处理决策，形成控制信息，通过模拟量输出通道（D/A），转变成被控对象可以接受的模拟信号，通过驱动器带动系统跟踪输入变化。因此计算机控制伺服系统由计算机、模拟量输入通道、模拟量输出通道以及被控对象组成。计算机控制伺服系统的被控对象一般为运动部件，为系统安全起见，常要求系统启动工作时，计算机与被控对象相互间“握一次手”，互相访问一下，都准备就绪了才开始工

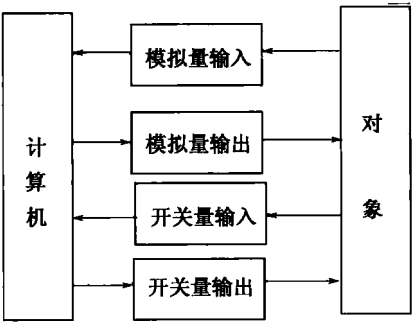


图 1-3 计算机控制伺服系统的组成

作。因此计算机控制伺服系统中还应该有开关量的输入、输出通道。这样计算机控制伺服系统的组成如图 1-3 所示。而其工作过程是：

实时数据采集：对被控参数的瞬时值进行检测、转换并输入到计算机中；

实时决策：对采集到的表征被控参数的状态变量进行分析，并按已给的控制规律进行计算，决定进一步的控制策略；

实时控制：根据决策的结果，适时地对控制机构发出控制信号。

计算机控制伺服系统就是不断重复上面三个步骤，控制整个系统按一定的品质指标进行工作，并对系统的异常状态进行监视和处理。控制过程的三个步骤对计算机来说实际上只是执行算术、逻辑运算和输入、输出操作。

所谓“实时”是指信号的输入、计算和输出都是在一定时间范围内完成。亦即计算机对输入信息以足够快的速度进行处理，并在一定的时间内作出反应或控制。实时性指标取决于下列延时：检测延时、输入延时、计算延时、输出延时等等，如图 1-4 所示。由上面三步所构成的循环周期就是实时时间，也称为采样周期 T_s 。实时时间的长短随控制对象的不同而不同，对于计算机控制伺服系统，因为控制对象是个快变化对象，因此要求实时时间要短，一般为 ms 级。控制系统的实时时间常由定时中断产生。

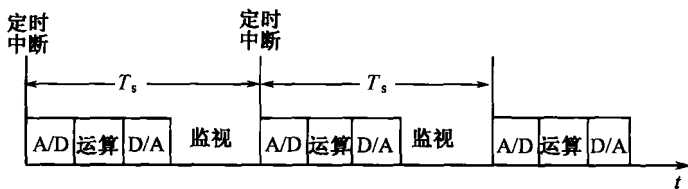


图 1-4 实时时间

1.3 计算机控制伺服系统的结构形式

现在常见的计算机控制伺服系统有如下三种结构形式：

1. 开环位置伺服系统

如图 1-5 所示，它以步进电动机作为执行元件。这是一种没有位置反馈的位置伺服系统，它能按计算机发出的位移指令驱动机械作相应的运动，不需要对机械的实际位移量或转角进行检测，它的位置控制精度完全靠传动机构的精度来保证。

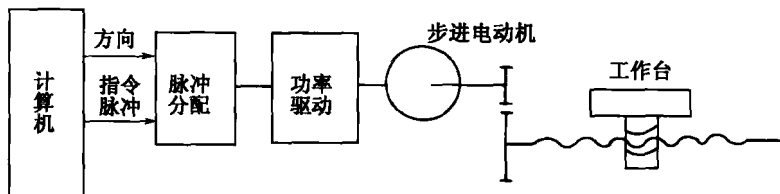


图 1-5 开环位置伺服系统

步进电动机实质上是一种同步电动机，每当向步进电动机发出一个进给脉冲时，步进电动机的转子就在此脉冲所产生的同步转矩作用下旋转一个固定的角度，通常称为步距角。因此它是一种将电脉冲变为角位移或线位移的电磁装置。其特点是定位精度高，但转动速度不快。步进电动机的步距角的大小与它的结构和控制方式有关，最常用的一般为 1.5° 、 0.75° 。步进电动机再经减速器带动丝杠，通过丝杠、螺母的相对转动，最后形成工作台的移动。这样工作台的位移量与进给脉冲的数量成正比，而工作台的移动速度与进给脉冲的频率，即单位时间的脉冲数成正比，显然这种开环位置伺服系统的位置精度完全取决于步进电动机的步距角精度和齿轮、丝杠等传动部件的精度。

这种开环控制系统由于结构简单，造价低，调试方便，仍然有广泛的应用场合。因此步进电动机和驱动器也在向集成化、模块化、细分方面不断发展。并有专门配套的控制模块、驱动模块（含细分）可供选用。

这种开环控制系统存在因负载惯性而引起的丢步问题。实际使用时需要加进升降速控制。

2. 基于基准脉冲工作方式的硬件数字伺服系统

为了克服使用步进电动机的开环位置伺服系统存在的丢步问题，又产生了一种由计算机或由硬件数字电路实现的、采用伺服电机闭环控制的、按步进电动机的控制方式工作的伺服系统，称为基准脉冲系统，又叫做脉冲—脉冲系统。由硬件数字电路实现又称为硬件数字伺服系统，即指位置闭环控制与调节采用数字技术由一系列数字电路来完成。一种采用偏差计数器—D/A 型的位置伺服系统如图 1-6 所示。图中 SM 为伺服电动机，PG 为光电脉冲发生器（Optical Pulse Generator），F/V 为频率电压变换。

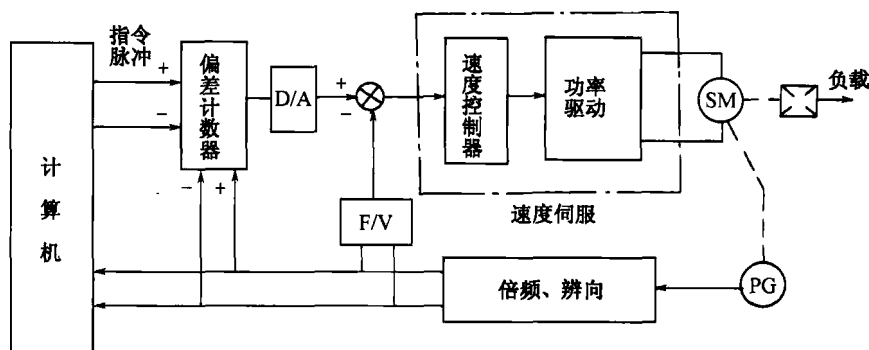


图 1-6 硬件数字伺服系统

增量式光电脉冲编码器是系统的反馈元件，由它产生位置反馈脉冲。用可逆计数器作数字比较器，它记录的是基准指令脉冲数与反馈脉冲数之差，即为位置误差。当代表坐标轴转向、转速和位移量的指令脉冲序列进入可逆计数器中进行计数，经双极性的 D/A 转换器将其转换成模拟量电压，就成为速度伺服单元的速度指令电压。电机按指令方向运转，电机一转，装在电机轴上的编码器产生比例于电机转角的脉冲数（脉冲频率比例于电机的转速），去减可逆计数器中积存的脉冲数。指令脉冲连续供给，上面的动作连续进行。当反馈脉冲频率与指令脉冲频率相等时，可逆计数器保持一定的积存量，电机以固定的转速连续运转。

当指令脉冲停止供给,由编码器来的反馈脉冲去减可逆计数器的积存脉冲数后,D/A转换器输出的速度指令电压也减小,电机慢慢减速。当可逆计数器积存的脉冲数为零时,电机停止转动,这样电机转到了与输入指令脉冲数成比例的转角,就停止转动。这种系统在停止时,由于有编码器来的反馈脉冲,可以产生停止位置的修正动作,因而系统具有电气嵌位能力。

这种系统的硬件结构,使得它的各调节器参数在机电联调整定后就被固定下来,不能改变,这对负载惯量变化不大的位置伺服系统可以获得较高的控制性能。而对于某些负载惯量变化较大的伺服系统,采用这种参数固定、结构不变的控制系统,很难在整个负载惯量变化范围内都取得同样令人满意的控制效果。

如前所述,这种位置伺服系统是在采用步进电机开环结构位置伺服系统基础上发展起来的,它的控制思想与开环系统相似,指令输入也采用脉冲形式,但和步进电机相比可以克服丢步问题。

这种系统又称为半闭环系统,因为它的位置检测器与伺服电机同轴,直接检测到的是电动机轴旋转的角位移,负载轴的位移是推算出来的,为间接检测,所以称为半闭环系统。这种系统机械传动的间隙和传动误差将成为系统误差。在计算机控制的系统中间隙和传动误差可以通过补偿方式来控制精度。

3. 软件数字伺服系统

上节介绍的计算机控制伺服系统是由计算机参与系统的闭环控制。控制器由计算机软件来实现,因此又称为软件数字伺服系统。这种系统控制更加灵活,许多先进的控制思想可以由控制器来实现,控制器的结构和参数也可以按照工作环境自动进行切换,使系统适应负载变化的能力显著增加。

软件数字伺服系统是本门课的研究重点。我们将主要介绍计算机控制伺服系统的硬件设计和控制器设计。

习题与思考题

1. 请说明伺服系统与调节系统相比有什么不同特点?
2. 画出计算机控制伺服系统的框图,说明其工作过程,比较与连续控制系统有什么异同点?
3. 一个计算机控制系统最小的采样周期受到什么因素限制?
4. 计算机控制系统的组成中为什么需要有开关量的输入、输出通道?
5. 说明硬件数字伺服系统工作原理。指出这种系统会不会出现丢步,为什么?
6. 一套数字伺服系统,要求最大跟踪角速度 $\Omega_m = 140^\circ/\text{s}$, 最大跟踪角加速度 $\epsilon_m = 200^\circ/\text{s}^2$ 。请计算等效正弦的振幅 A 和周期 T 。

第二章 通道接口技术

本章主要介绍计算机控制系统的硬件构成，包括计算机对外围通道的控制、模拟量输入通道、模拟量输出通道。由于开关量的输入输出可归结为模拟量输入输出通道中的某一位，因此本章不专门介绍。

2.1 计算机对外围通道的控制

计算机对外围通道的控制，一般说来其工作基础是中断系统，具体实施是通过接口电路进行的，计算机的输入输出指令及其有关的逻辑提供了 CPU 使用外围通道的全部控制信号。因此本节重点介绍指令、中断、接口三个问题。

2.1.1 输入输出指令

1. 输入输出接口的两种编址方式

计算机与外围通道的连接一般要用到地址总线、数据总线、控制总线。如图 2-1 所示。当外围通道地址确定之后，CPU 可以对该地址进行读写。执行读指令，可以把外围通道的数据传送到 CPU；执行写指令，CPU 的数据传给外围通道。

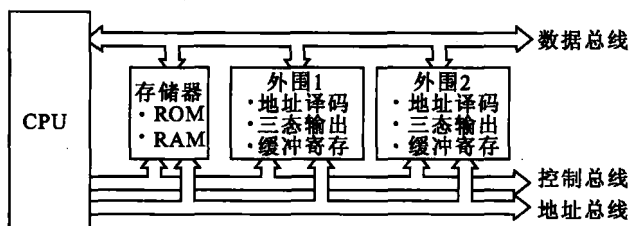


图 2-1 三总线

外围通道的地址对 CPU 来说有两种编址方式：一种是按存储器布局，即将外围通道的地址分配在存储器的地址空间，因此

又称为存储器映像 I/O。对于 M6800, 6502, 8031, 8098 等 CPU 的计算机只能采用这种编址方式。此时计算机对外围通道的输入输出操作就像对一个存储单元进行读写操作一样，所有访问内存的指令均可以适用于输入输出。由于这种编址方式不必区分是输入输出还是访问内存，因此 CPU 的指令系统中没有设置输入、输出指令。

我们以 Z80 指令为例，

LD A, (nn) 指令可以实现输入操作，

LD (nn), A 指令可以实现输出操作。

一种简单实用的输入、输出与存储器统一编址方式是用地址总线最高位 A_{15} 作为选择线，控制信号的连接如图 2-2 所示。

当 $A_{15}=0$ 时，(0000H~7FFFH) 为 32K 内存地址空间；

当 $A_{15}=1$ 时，(8000H~FFFFH) 为 32K I/O 地址空间。

这种输入输出的存储器编址方式的优点是访问内存的指令多，而且功能强，因此输入输出寄存器可以直接参与运算；可与存储器共用地址译码与控制电路；I/O 地址数几乎是不受

限制的。缺点是 I/O 要占用内存地址空间；需要全字长的地址译码；程序中较难区分是否为 I/O 操作。

第二种是按输入、输出布局。这种方式外围通道与存储器分开编址，又称隔离 I/O。这种编址方式，CPU 有专门的输入输出指令，并有相应的控制线指示是 I/O 操作。例如 Z80 的 I/O 请求信号 $\overline{\text{IORQ}}$ ，8086 的 $\text{M}/\overline{\text{IO}}$ 。80 系列的 CPU 如 Z80，8085，8086 等均可采用这种编址。图 2-3 示出了 Z80 CPU I/O 编址方式所用的控制信号。

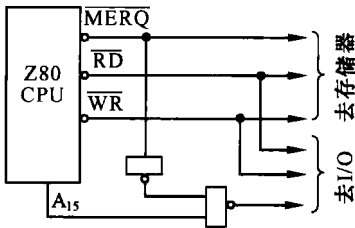


图 2-2 存储器编址

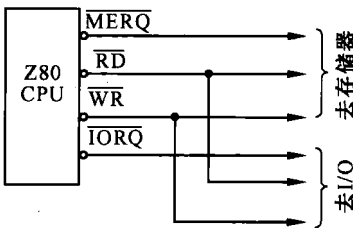


图 2-3 I/O 编址

输入输出布局的优点是 I/O 地址线少，译码简单；使用 I/O 指令，程序中易于识别。缺点是只能使用 I/O 指令，处理能力不如存储器布局强。

2. 输入输出指令和时序

了解了外围通道的两种编址方式之后，我们知道选用不同编址方式需要使用不同指令。对于使用 68 系列、8031 等 CPU 的计算机，只能采用存储器布局，相应的只能使用访问存储器指令。而 80 系列等 CPU，可由设计者根据需要任意选用。

下面介绍指令时序。了解时序的目的是使大家知道一条指令执行的过程中，CPU 能供给用户哪些信号，这些信号相互之间时间配合上的关系，要求接口电路设计者所选用的输入三态门电路和输出寄存器的动作时序要与指令时序相一致。

以 Z80 CPU 直接寻址的输入、输出指令为例。指令格式是：

IN A, (n); A ← (n) 输入指令
OUT (n), A; (n) ← A 输出指令

n 是通道的端口地址，它为一个字节，在指令执行时，端口地址要出现在地址总线的低八位上。输入指令的功能是把端口地址为 n 的数据(或状态)送到计算机的累加器 A；输出指令的功能是把累加器 A 的数据送到端口地址为 n 的寄存器中。

每条指令的执行都是在统一时钟脉冲控制下一个节拍一个节拍地进行。指令执行过程所需的时钟脉冲称为指令周期。上述输入输出周期的时序如图 2-4 所示。

当 CPU 从程序计数器所指定的内存单元中取出的指令是输入或输出指令时，经指令译码后确定的操作是 T_1 时钟周期上升沿后，输入或输出端口地址出现在地址总线的低八位，在 T_2 周期的上升沿之后，输入输出请求线 $\overline{\text{IORQ}}$ 和读 $\overline{\text{RD}}$ 或者写 $\overline{\text{WR}}$ 变为有效。紧接着 CPU 在 I/O 操作时自动插入等待周期 T_w^* ，这是因为在 $\overline{\text{IORQ}}$ 有效到 T_2 下降沿采样 $\overline{\text{WAIT}}$ 线的时间间隔十分短，在没有 T_w^* 周期时，即使 I/O 端口要求等待也没有足够时间发出 $\overline{\text{WAIT}}$ 申请，同时没有 T_w^* 周期，也很难设计 MOS I/O 装置能在这样的速度下工作。这样 CPU 在自动插入的 T_w^* 周期的下降沿再采样 $\overline{\text{WAIT}}$ 线以决定是否要插入等待周期 T_w^* 。在输入时，

以 $\overline{\text{IORQ}}$ 和 $\overline{\text{RD}}$ 作为通道的选通信号，使端口数据上到数据总线，CPU 在 T_3 周期的下降沿将数据输入到累加器 A。从时序图上可见， $\overline{\text{IORQ}}$ 和 $\overline{\text{RD}}$ 有效到数据上到数据总线允许约 $2T$ 时间，对于 2.5MHz 时钟的 CPU，要求输入端口三态缓冲器的响应时间要小于 $2T = 800\text{ns}$ 。在输出时，CPU 在 T_1 周期的下降沿后把输出的数据送到数据总线，并且保持整个指令周期。当 $\overline{\text{IORQ}}$ 和 $\overline{\text{WR}}$ 信号使通道寄存器选通时，在 T_3 周期的下降沿之后 $\overline{\text{WR}}$ 信号失效，将数据送入通道寄存器。对于 2.5MHz 时钟的 CPU， $\overline{\text{WR}}$ 信号有效时间约为 1 000ns， $\overline{\text{WR}}$ 失效后有效数据约保持 120ns 后再消失，这就要求通道的寄存器能适应这一关系。

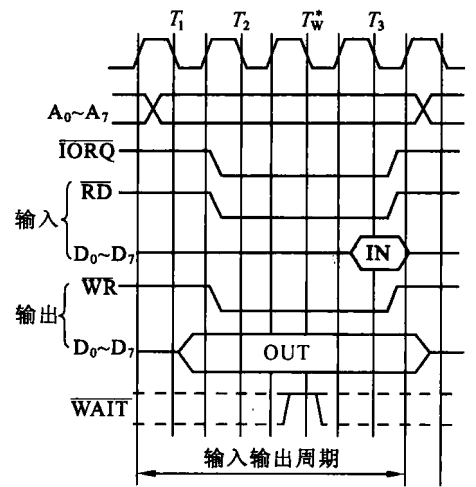


图 2-4 输入输出时序

下面再介绍一种以存储器布局方式工作的 8098 单片机对外部存储器或 I/O 操作的时序。8098 单片机有多种运行方式，最常用的是标准总线方式，它使用 16 位多路复用地址和 8 位数据总线。CPU 通过与 8 位数据总线复用的 16 位地址总线 $\text{AD}_0 \sim \text{AD}_7$ 、 $\text{A}_8 \sim \text{A}_{15}$ ，来对外部存储器或 I/O 进行寻址。为了将总线上的地址和数据信号分离，8098 提供了地址锁存允许信号 ALE，利用其下降沿给一个锁存器（例如 74LS373）送一个信号，用以锁存来自 $\text{AD}_0 \sim \text{AD}_7$ 的低 8 位地址。

图 2-5 为 8098 对外部存储器或 I/O 进行操作的理想定时波形。当对外存储器或 I/O 进行操作时，ALE 变高，16 位地址从 $\text{AD}_0 \sim \text{AD}_7$ 和 $\text{A}_8 \sim \text{A}_{15}$ 引脚送出。其后 ALE 变低， $\text{AD}_0 \sim \text{AD}_7$ 被锁存而 $\text{A}_8 \sim \text{A}_{15}$ 一直有效。

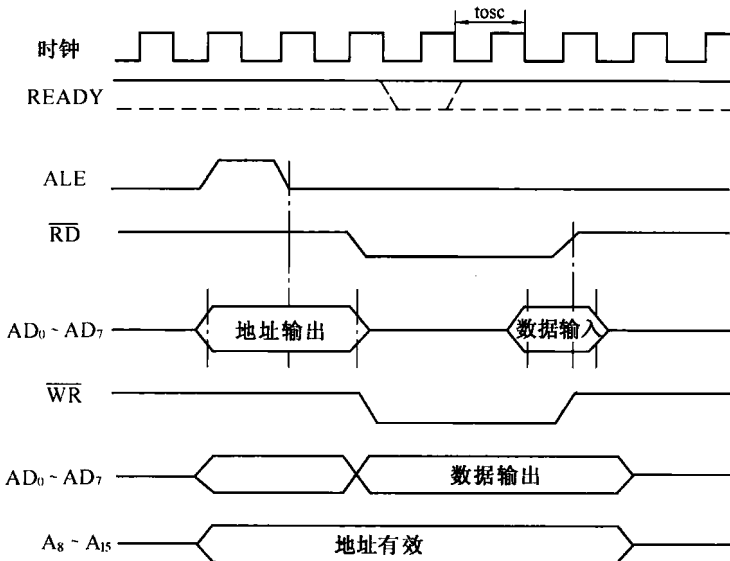


图 2-5 8098 I/O 时序

读操作：在读外部存储器或 I/O 期间，存储器或 I/O 的数据必须在 $\overline{RD}$ 信号上升沿出现之前放到 $AD_0 \sim AD_7$ 总线上并稳定下来。这一稳定时间称为数据建立时间，8098 在 $\overline{RD}$ 信号的上升沿将总线上的数据读入片内。

写操作：对外部存储器的写操作时序与读操作相同，在 $\overline{WR}$ 信号的下降沿后，8098 从总线上撤消地址信号的同时送出所要写入的数据。当 $\overline{WR}$ 变为高电平时，则数据被锁存到外部存储器。

为了实现对低速存储器或 I/O 的存取，8098 还提供一个准备就绪 (READY) 信号，用以展宽读和写信号的宽度。假如 READY 信号在 ALE 信号下降沿后的规定时间内变为低电平，那么在时钟信号的下降沿到来之前，8098 总线上的状态保持不变（即插入了等待状态）。当 READY 信号变为高电平后，在下一个时钟信号的下降沿，总线操作将继续进行。

从上面两个例子可以看出，尽管时序的形式不同，但是对 I/O 的接口要求是一样的。

第三个有代表性的时序是系统总线时序，以 IBM PC 总线的 I/O 时序为例，大家可以看出和 Z80 CPU 的时序很相似。

IBM PC 机 CPU 是 INTEL8088，其主机板上有用于扩展各种接口的扩展槽，即为系统总线。它们是经过主机板逻辑处理后，由 CPU 提供给用户使用的信号线。其 I/O 时序如图 2-6 所示。

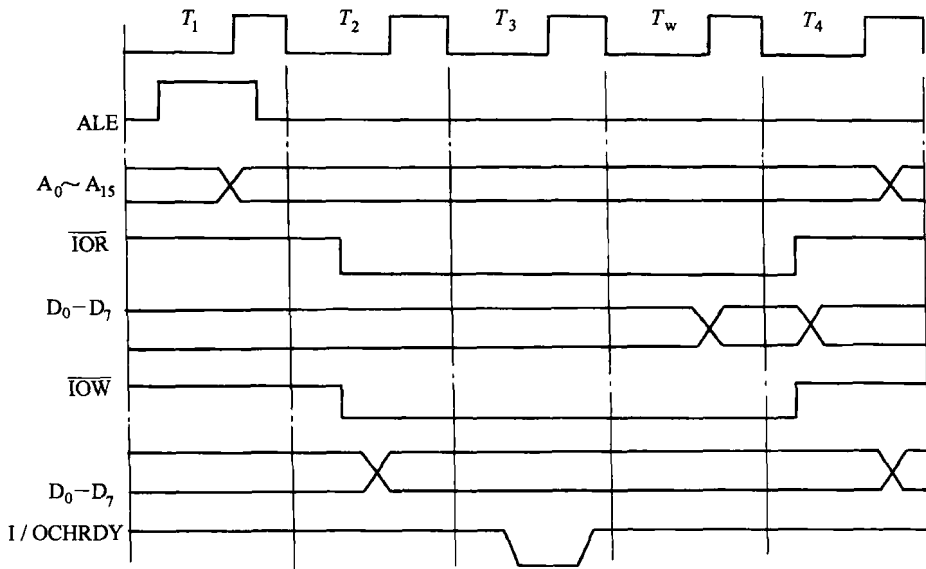


图 2-6 PC 总线的 I/O 时序

这些总线周期通常是 4 个时钟周期长，但是 PC 设计成可自动插入一个额外的 T_w （等待）时钟，因此在 PC 中，所有 I/O 总线周期最小均为 5 个时钟周期或者长度均为 $1.05\mu s$ 。总线周期能够通过控制在系统总线上的准备就绪信号 (I/O CHRDY) 进一步延长。由 8088 硬件所决定的，I/O 总线周期中 PC 地址引线 $A_{16} \sim A_{19}$ 是无效的。

8088 处理器执行 IN 指令时，启动 I/O 读总线周期，在 T_1 期间 ALE 信号有效，它表明在它的后沿，地址总线 $A_0 \sim A_{15}$ 包含有效的 I/O 端口地址。在 T_2 期间总线控制信号 $\overline{IOR}$ 有

效，它表明被访问的输入端口须将其内容送到数据总线上作为响应。在 T_4 时钟的起始部分，处理器对数据总线上的数据采样，然后 $\overline{IOR}$ 信号失效。

8088 处理器执行 OUT 指令时，启动 I/O 写总线周期，在 T_1 期间 ALE 信号有效，表明在它的后沿地址总线包含有效的 I/O 端口地址，随后在 T_2 期间总线控制信号 $\overline{IOW}$ 有效，标志被访问的端口须从数据总线上取数据。在 T_2 周期的后半部，8088 处理器驱动数据总线将数据送到输出端口。在 T_4 周期开始后， $\overline{IOW}$ 信号失效，8088 处理器从总线中移去数据。

2.1.2 中断

计算机与外围通道之间传送数据的方式通常有两种：DMA 传送和程序传送。

DMA 传送较复杂，需要有专门的 DMA 控制器，控制外围设备与存储器之间直接存取。这种传送方式适用于高速、大批量数据传送，大多用于数据采集系统，一般的控制系统较少采用。

程序传送可分：握手方式；查询方式；中断控制方式。

握手方式如图 2-7 所示。CPU 向外围通道发出启动命令之后，外围通道开始工作，进入忙状态，将通道忙信号引入 CPU，使 CPU 在通道工作期间一直处于等待状态。待外围通道工作结束，解除 CPU 等待状态，CPU 立即读出数据。这种方式适用于外围通道工作时间在几十微秒以下的系统，不适用于低速系统，因为使用这种方式 CPU 等待时间将很可观。

查询方式是 CPU 启动外围通道工作之后，就不停地读进外围通道的工作状态，一旦发现“忙”状态解除，就立即读进通道的数据。为了不使 CPU 的时间白白浪费在查询上，也可以在 CPU 发出启动命令之后，就去执行别的程序，每隔适当的程序步再去查询外围通道的工作状态。这两种作法，要么浪费计算机机时，要么响应不及时。

中断方式是 CPU 的一种处理外界实时信息的功能。CPU 在正常运行程序时，允许由外部电路随时迫使 CPU 暂停当前运行程序而转向另一个称为中断服务程序的程序段。该程序段往往是为外围通道服务的。当执行完中断服务程序后，再继续运行被中断的程序。

有了中断系统之后，可以克服前面提到的握手方式和查询方式的缺点，特别是直接面向生产过程的外围通道，由于生产系统的特点，决定着外围通道工作速度可以不必很高，这就更有必要用中断控制方式来协调快速 CPU 与慢速外围通道之间的关系。CPU 启动外围通道工作之后就以自己固有规律工作，当外围通道完成任务之后，向 CPU 发出中断请求，要求 CPU 暂停自己的工作，转去为外围通道服务。这样就可以实现 CPU 使用外围通道而不等待外围通道。这种中断工作方式的流程图如图 2-8 所示。

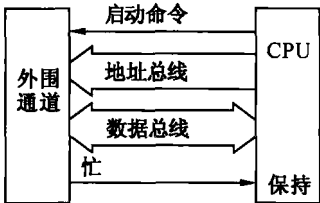


图 2-7 握手方式

下面以 Z80 中断方式 0 这种简单的中断方式来介绍一般的中断过程，以便了解如何安排中断控制方式。

CPU 响应中断的条件是：

首先要有中断申请信号，而且这个信号还要能保持住。当 CPU 响应这个中断之后，还要能清除这个中断申请。因此要求外围通道的接口电路要设置一个中断申请触发器。

其次是该中断申请不被屏蔽。一般为了增加中断的控制灵活性，在通道接口电路中设置