

火电厂热工自动控制技术丛书

过程计算机控制及 先进控制策略的实现

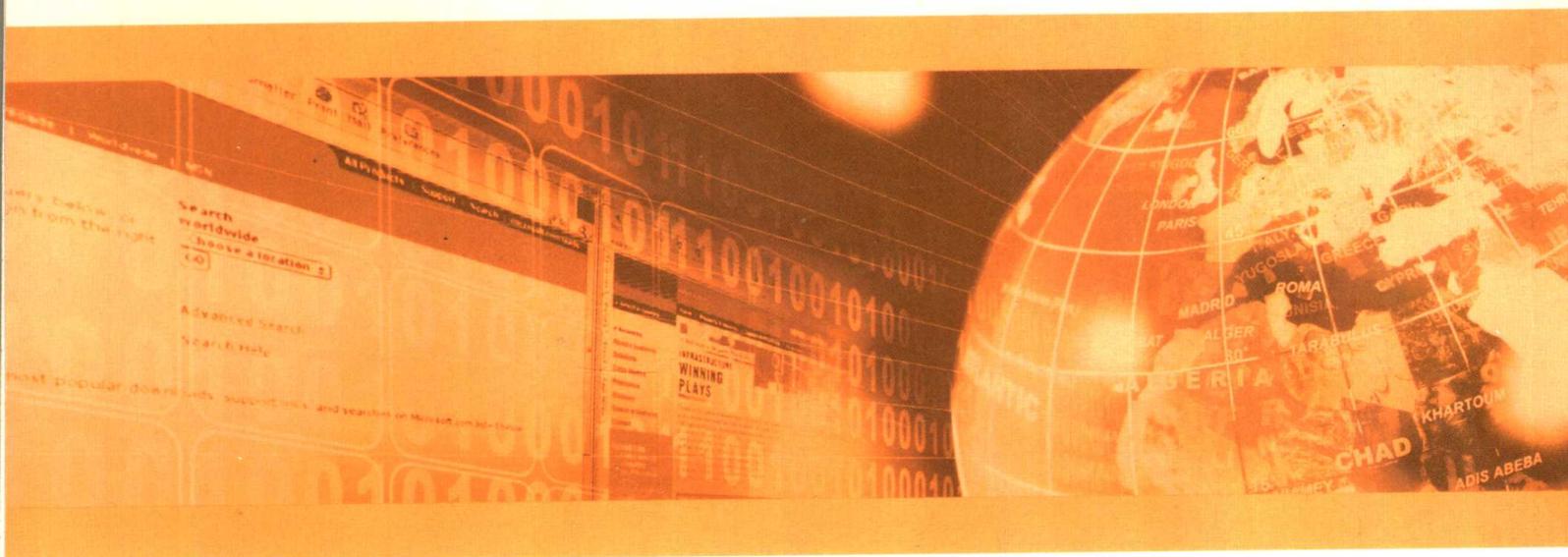


田涛 编著



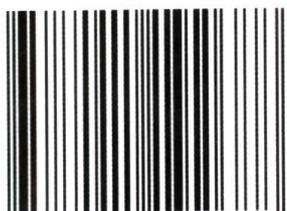
机械工业出版社
CHINA MACHINE PRESS

- ISBN 7-111-19268-0
- 策划编辑：吉 玲
- 封面设计：王伟光



上架指导：工业技术 / 自动化技术

ISBN 7-111-19268-0



9 787111 192688 >

定价：33.00 元

编辑热线：(010)88379768

地址：北京市百万庄大街22号

邮政编码：100037

联系电话：(010) 68326294

网址：<http://www.cmpbook.com>

(010) 68993821

E-mail: online@cmpbook.com

火电厂热工自动控制技术丛书

过程计算机控制及先进控制策略的实现

田 涛 编著



机械工业出版社

全书共分 10 章。其中有关计算机控制方面的内容,以 PC/PCI 总线工业控制计算机为基础展开,同时介绍了有关通信网络、分散控制系统和现场总线控制系统等知识;不仅介绍了 PID 算法、最小拍算法等常规控制算法,还介绍了自适应控制、模糊控制、神经控制等先进的控制策略及其计算机实现;以 3 类不同形式的火电厂热工控制工程实例阐述了控制算法的计算机实现方法。

本书适合于自动化相关专业的工程技术人员、大学本专科高年级学生及研究生阅读。

图书在版编目 (CIP) 数据

过程计算机控制及先进控制策略的实现 / 田涛编著. —北京: 机械工业出版社, 2006.6

ISBN 7-111-19268-0

I. 过... II. 田... III. 计算机控制 IV. TP273

中国版本图书馆 CIP 数据核字 (2006) 第 057404 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

责任编辑: 吉 玲 (E-mail: jiling@mail.machineinfo.gov.cn)

责任印制: 李 妍

三河市汇鑫印务有限公司印刷

2007 年 1 月第 1 版第 1 次印刷

184mm×260mm·19 印张·468 千字

0001—4000 册

定价: 33.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

本社购书热线电话 (010) 68326294

编辑热线: (010) 88379768

[Http://www.machineinfo.gov.cn/book/](http://www.machineinfo.gov.cn/book/)

封面无防伪标均为盗版

前 言

本书面向生产过程自动化领域，以火电厂热工自动控制为背景，系统地介绍了计算机控制系统方面的原理、设计方法及工程应用，适合于自动化相关专业的工程技术人员，大学本科高年级学生及研究生作为培训、教学用书或技术参考用书。

计算机控制是自动控制技术、计算机技术、通信网络技术等多学科结合的综合性的工程技术学科。从计算机应用的角度出发，自动控制工程是其重要的一个应用领域；而从自动控制工程来看，计算机技术又是一个重要的实现手段。

全书内容共分为 10 章。其中有关计算机控制装置方面的内容，以 PC/PCI 总线工业控制计算机为基础展开，同时介绍了有关通信网络、分散控制系统和现场总线控制系统等方面的基础知识；有关计算机控制算法部分，除了介绍数字 PID 算法、最小拍算法等常规控制算法外，还介绍了自适应控制、模糊控制、神经控制等先进的控制策略及其计算机实现；计算机控制系统的设计部分，是以 3 类不同形式的火电厂热工控制工程实例来阐明控制算法的计算机实现方法。本书内容按照由外而内、由硬件到软件的顺序逐步展开，先介绍控制计算机的抗干扰、输入/输出通道、IPC、控制网络基础等，再分析核心的控制算法，然后通过工程实例使读者对计算机控制系统的设计方法有一个总体的了解和把握。

本书在编写过程中力求理论分析计算与应用技术并重，强调计算机控制理论的实际运用，注重软件与硬件的有机结合，重视对近年来出现的新技术、先进控制方法的及时引入。同时，根据作者从事火电厂热工自动控制方面科研工作的亲身体会和经验，有机地融入了作者参加的国家 863 项目的科研成果，着重于对读者的分析问题能力、实际动手能力和创新思维能力的培养和训练。

本书作为教材使用，先修课程应包括微机原理及应用、自动控制原理、数字电路、模拟电路等，学时安排可以根据选用的章节灵活掌握。如果作为自动控制专业技术基础课程，一般不包含第 7 章分散控制系统与现场总线控制系统部分、第 9 章自适应控制部分（另设选修课程）以及第 10 章部分内容，参考学时大约 60 学时。

在本书的编写过程中，得到了华北电力大学白焰教授、大连轻工业学院林敏教授的热情帮助和大力支持，在此向他们表示诚挚的谢意！本书吸纳了许多兄弟院校计算机控制方面教材的长处，在此一并表示由衷的感谢！

由于作者水平有限，加之计算机控制技术的发展日新月异，书中不足或缺点在所难免，敬请广大读者批评指正，提出宝贵意见。

编 者

于华北电力大学

目 录

前言

第 1 章 计算机控制系统概述	1
1.1 计算机控制系统的结构与组成.....	1
1.1.1 计算机控制系统的一般结构.....	1
1.1.2 计算机控制系统的硬件组成.....	2
1.1.3 计算机控制系统的软件组成.....	3
1.2 计算机控制系统的分类.....	4
1.2.1 按功能及结构分类.....	4
1.2.2 按控制规律分类.....	8
1.2.3 按控制设备分类.....	8
1.3 计算机控制系统的特点.....	11
1.4 计算机控制系统的主要研究内容.....	11
1.4.1 输入数据处理.....	11
1.4.2 输出处理.....	12
1.4.3 控制功能.....	13
1.4.4 人-机接口功能.....	14
1.4.5 通信功能.....	14
1.4.6 自诊断功能.....	14
1.4.7 冗余技术.....	14
1.5 计算机控制系统的发展.....	15
1.5.1 计算机控制技术的发展历程.....	15
1.5.2 计算机控制技术的发展动向.....	16
思考题.....	17
第 2 章 过程计算机控制的理论基础	18
2.1 信号变换原理.....	18
2.1.1 计算机控制系统的信号流程.....	18
2.1.2 脉冲采样器和采样过程.....	19
2.1.3 采样定理.....	20
2.1.4 过程计算机控制系统中采样周期的选择.....	23
2.1.5 信号复现与零阶保持器.....	24
2.2 连续模型与离散模型间的转换.....	26
2.2.1 传递函数与 z 传递函数间的转换.....	26
2.2.2 连续与离散状态方程之间的相互转换.....	30
2.3 采样控制系统的稳态及动态分析.....	32

2.3.1	采样控制系统的稳态分析	32
2.3.2	采样控制系统的动态分析	33
2.4	线性离散控制系统的稳定性分析	35
2.4.1	S 平面与 Z 平面的映射关系	36
2.4.2	线性离散系统的稳定域	36
2.4.3	线性离散系统的稳定性判据	37
	思考题	41
第3章	过程计算机系统的抗干扰技术	43
3.1	干扰的来源与传播途径	43
3.1.1	干扰的来源	43
3.1.2	干扰的传播途径	44
3.2	串模干扰、共模干扰和长线传输干扰	46
3.2.1	串模干扰及其抑制	46
3.2.2	共模干扰及其抑制	47
3.2.3	长线传输干扰及其抑制	50
3.3	供电技术与接地技术	52
3.3.1	供电技术	52
3.3.2	接地技术	55
3.4	输入、输出通道信号隔离技术	58
3.4.1	数字信号输入、输出通道的隔离	59
3.4.2	模拟量输出通道的隔离	60
3.4.3	模拟量输入通道的隔离	61
3.4.4	常用隔离放大器	62
3.5	控制计算机的可靠性技术	64
3.5.1	故障自诊断技术	65
3.5.2	冗余技术	66
3.5.3	CPU抗干扰技术	66
3.6	数字滤波技术	68
3.6.1	平均值滤波	69
3.6.2	限幅滤波与限速滤波	70
3.6.3	惯性滤波	70
	思考题	71
第4章	过程输入、输出通道	72
4.1	过程输入、输出通道的控制方式	72
4.1.1	过程输入、输出通道与CPU交换的信息类型	72
4.1.2	过程通道的编址方式	72
4.1.3	CPU对过程通道的控制方式	73
4.1.4	过程通道接口设计应考虑的问题	74
4.2	模拟量输出通道	75

4.2.1	D/A 转换器	75
4.2.2	D/A 通道的输出电路	82
4.3	模拟量输入通道	86
4.3.1	信号调理与放大	87
4.3.2	多路模拟开关	89
4.3.3	采样保持器	90
4.3.4	A/D 转换器工作原理	92
4.3.5	逐位逼近式 A/D 转换器及其接口	98
4.4	数字量输入、输出通道	106
4.4.1	数字量输入通道	106
4.4.2	数字量输出通道	111
4.5	量化误差与字长选择	115
4.5.1	量化误差来源	115
4.5.2	A/D、D/A 字长的选择	117
4.6	数据预处理	118
4.6.1	系统误差的自动校准	118
4.6.2	标度变换	119
4.6.3	越限报警处理	124
	思考题	125
第 5 章	总线和总线式工控机	127
5.1	总线及其分类	127
5.1.1	总线的分类	127
5.1.2	总线的模板化结构	128
5.2	工业控制计算机的系统总线	128
5.3	总线式工业控制计算机	131
5.3.1	IPC 的硬件组成	131
5.3.2	IPC 的特点	132
5.4	IPC 产品简介	133
5.4.1	工业计算机机箱	133
5.4.2	工业级底板	134
5.4.3	工业级 CPU 卡	134
5.4.4	I/O 卡	136
5.5	远程数据采集与控制模块	138
	思考题	142
第 6 章	控制网络技术基础	143
6.1	数据通信基础	143
6.1.1	数据通信系统	143
6.1.2	数据传输编码	145
6.1.3	多路复用技术	147

6.1.4 通信同步技术	149
6.1.5 常用传输介质	151
6.2 通信网络技术	151
6.2.1 网络拓扑结构	151
6.2.2 网络控制方法	152
6.2.3 差错控制技术	153
6.3 OSI 模型	155
6.4 IEEE 802 标准	158
6.5 TCP/IP	160
6.5.1 TCP/IP 的网络层次结构	160
6.5.2 IP 地址	162
6.6 串行通信总线	163
6.6.1 RS-232C 通信总线	163
6.6.2 RS-422/485 通信总线	169
思考题	171
第 7 章 分散控制系统与现场总线控制系统	173
7.1 分散控制系统概述	173
7.1.1 分散控制系统的结构	173
7.1.2 分散控制系统的特点	175
7.1.3 分散控制系统的分散方式	178
7.1.4 部分 DCS 产品	178
7.2 分散控制系统的典型设备	179
7.2.1 过程控制站	179
7.2.2 运行员操作站	183
7.2.3 工程师工作站	185
7.3 现场总线控制系统概述	189
7.3.1 现场总线的概念	189
7.3.2 现场总线控制系统的构成	190
7.3.3 现场总线控制系统的特点	192
7.4 几种典型的现场总线	196
7.4.1 CAN 总线	196
7.4.2 LonWorks 总线	196
7.4.3 Profibus 总线	197
7.4.4 WorldFIP 总线	197
7.4.5 HART 总线	197
7.4.6 FF 总线	198
思考题	198
第 8 章 常规数字控制器的设计	199
8.1 数字控制器连续化设计	199

8.1.1	数字控制器的连续化设计步骤	199
8.1.2	数字 PID 控制器的设计	200
8.1.3	数字 PID 算法的改进	204
8.1.4	数字 PID 参数的整定	208
8.2	数字控制器离散化设计	211
8.2.1	数字控制器的离散化设计步骤	211
8.2.2	最少拍控制系统的设计	212
8.3	纯滞后控制	220
8.3.1	大林 (Dahlin) 控制算法	220
8.3.2	施密斯 (Smith) 预估控制算法	223
	思考题	224
第 9 章	先进控制技术基础	226
9.1	自适应控制	226
9.1.1	自校正控制	227
9.1.2	参考模型自适应控制系统	244
9.2	模糊控制	257
9.2.1	模糊控制器的输入、输出变量及其模糊化	258
9.2.2	建立模糊控制规则	260
9.2.3	模糊关系与模糊推理	262
9.2.4	模糊控制向量的模糊判决	263
9.2.5	模糊控制表	263
9.2.6	确定实际的控制量	264
9.2.7	模糊控制算法的工程实现	264
9.3	神经网络控制	265
9.3.1	神经元数理模型及其学习算法	265
9.3.2	BP 神经网络及其学习算法程序设计	268
9.3.3	Hopfield 神经网络	270
9.3.4	神经网络在控制工程中的应用	273
9.4	控制策略的渗透和结合	274
9.5	先进控制策略的实现	275
9.5.1	在 DCS、FCS 中组态的实现	275
9.5.2	在 DCS、FCS 中编程实现的途径	277
9.5.3	编程实现时需要注意的两个问题	279
第 10 章	控制算法的计算机实现工程实例	281
10.1	常规控制算法的工控机实现	281
10.2	常规控制算法的 DCS 实现	282
10.2.1	主蒸汽温度控制方案	282
10.2.2	主蒸汽温度控制系统的 Infi-90 实现	284
10.3	模型参考自适应预估控制的 DCS 实现	286

10.3.1	模型参考自适应预估算法的基本设计思想	286
10.3.2	模型参考自适应预估基本算法	287
10.3.3	算法在 Infi-90 中的编程实现	288
10.3.4	在火电厂燃烧协调控制系统中的应用	288
参考文献		293

第 1 章 计算机控制系统概述

计算机控制系统是自动控制理论、计算机技术以及现代通信网络技术紧密结合的产物，过程计算机控制是关于计算机技术如何应用于工业生产过程自动化的一门综合性学科。控制理论的发展，尤其是现代控制理论的发展，与计算机技术息息相关。利用计算机快速而又强大的数值计算、逻辑判断等信息加工能力，计算机控制系统可以实现比常规控制更复杂、更全面的控制方案。现代通信网络技术的发展更进一步扩展了计算机的信息处理能力。计算机和现代通信网络为现代控制理论的应用提供了有力的工具。同时，计算机控制系统应用于工业控制领域又产生了一系列理论与工程上的问题，进一步推动了控制理论和计算机技术的共同发展。

本书侧重于计算机对连续生产过程的控制，诸如电力、石油、化工、冶金、轻工、建材和制药等行业，其被控量主要是温度、压力、流量、物位和成分等各种连续模拟量。当然，对于机械制造与加工过程，诸如转速、位置等被控量的计算机控制系统，在原理与结构上也大致相同，只是在输入通道与输出通道上有些差异。

本章主要介绍了计算机控制的结构与组成、系统分类、系统特点以及发展概况等内容。

1.1 计算机控制系统的结构与组成

1.1.1 计算机控制系统的一般结构

计算机控制系统是融计算机技术与工业过程控制于一体的综合性技术，它是在常规仪表控制系统的基础上发展起来的。若将自动控制系统中控制器的功能用计算机或数字控制装置来实现，就构成了计算机控制系统，其基本原理框图如图 1-1 所示。因此，简单说来，计算机控制系统就是由各种各样的计算机参与控制的一类系统。

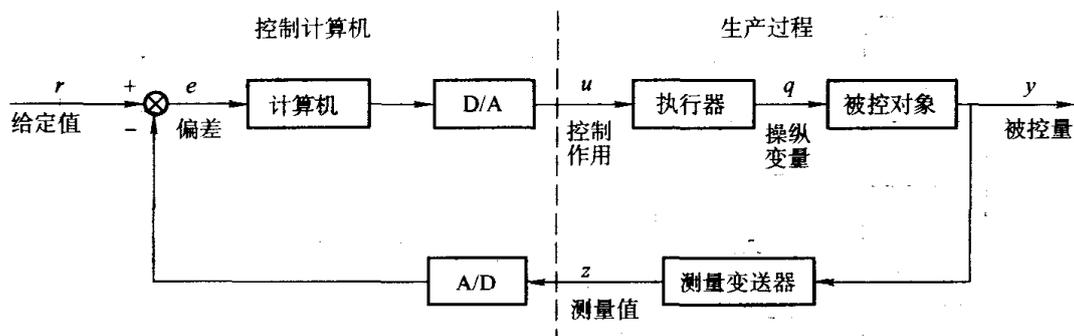


图 1-1 计算机控制系统基本原理框图

在一般的模拟控制系统中，控制规律是由硬件电路产生的，要改变控制规律就要更改硬件电路。而在计算机控制系统中，控制规律是用软件实现的，计算机执行预定的控制程序，就能实现对被控参数的控制。因此，要改变控制规律，只要改变控制程序就可以了。这就使

计算机控制系统的设计更加灵活、方便，特别是可以利用计算机强大的计算、逻辑判断、记忆、信息传递功能，实现更为复杂的控制规律，如非线性控制、逻辑控制、自适应控制及智能控制等。

在计算机控制系统中，计算机的输入和输出信号都是数字量，因此在这样的系统中，需要将反映工业对象连续信息的模拟量转换成计算机能接收的数字量，这由 A/D 转换器实现，同时将计算机控制输出的数字量转换为工业现场执行机构可接收的模拟量，这由 D/A 转换器实现。

计算机控制系统的控制过程一般可归纳为 3 个步骤：

- 1) 实时数据采集，对被控参数的瞬时值实时采集，并输入计算机；
- 2) 实时决策控制，对采集到的表征被控参数的状态量进行分析，并按已确定的控制规律，确定进一步的控制行为；
- 3) 实时控制输出，根据做出的控制决策，适时地向执行机构发出控制信号，在线、实时地实施控制。

以上过程不断地呈周期性重复，使整个系统能按照一定的动态性能指标进行工作。此外，计算机控制系统还能对被控参数和设备本身可能出现的异常状态进行及时监督和处理。这里的所谓“实时”，是指信号的输入、运算处理和输出能在一定的时间内完成，即要求控制计算机对输入信号要以足够快的速度进行测量与处理，并在一定的时间内反应或产生相应的控制作用，超过这个时间，就会失去控制时机。为了实现上述计算机控制的功能，需要控制计算机在硬件和软件两大部分的支持。

1.1.2 计算机控制系统的硬件组成

计算机控制系统的硬件一般由主机、常规外部设备、过程输入输出设备、操作台和通信设备等组成，如图 1-2 所示。

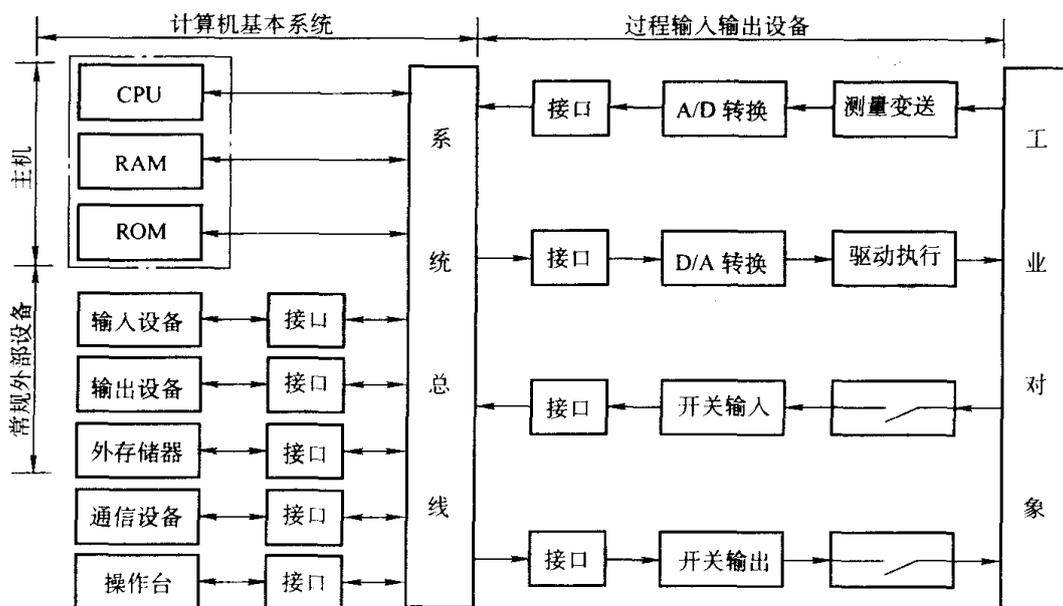


图 1-2 计算机控制系统硬件组成框图

与普通计算机硬件相比,控制计算机硬件具有如下3方面的特点:

1. 包含过程输入输出设备

过程输入输出设备是在计算机与工业对象之间起着信息传递和转换作用的装置,除了其中的测量变送单元和信号驱动单元属于自动化仪表的范畴外,主要是指过程输入输出通道,简称过程通道。过程输入通道包括模拟量输入通道(简称A/D通道)和数字量输入通道(简称DI通道),分别用来输入模拟量信号(如温度、压力、流量、液位等)和开关量信号(继电器触点、行程开关、按钮等)或数字量信号(如转速、流量脉冲、BCD码等)。过程输出通道包括模拟量输出通道(简称D/A通道)和数字量输出通道(简称DO通道),D/A通道把数字信号转换成模拟信号后再输出,DO通道则直接输出开关量信号或数字量信号。

2. 可靠性高和可维护性好,能适应工业控制现场恶劣的运行环境

为了适应工业过程控制对系统的可靠性要求,控制计算机的硬件普遍采取了有效的抗干扰、冗余、自诊断等可靠性技术。例如,典型的总线式工业控制计算机(IPC),就采用了加固型工业机箱、正压送风、工业电源、无源底板、一体化主板(主要部件采用工业级芯片)、看门狗(WatchDog)技术、电子盘存储技术等措施来提高系统的可靠性,以适应高温、高湿、腐蚀、振动、冲击、灰尘等恶劣的工业环境,且具有较高的电磁兼容性。

3. 具有高可靠性的实时通信网络

现代化工业生产过程的规模一般比较大,其控制与管理也很复杂,往往需要几台或几十台计算机才能分级完成控制和管理任务。这样,在不同地理位置、不同功能的计算机之间就需要通过通信设备连接成网络,以便进行信息交换。与普通的通信网络相比,控制网络对通信的可靠性要求很高,如典型的DCS或FCS大多采用冗余的通信总线结构,同时与外部网络的连接也采用硬防火墙等网络隔离设备。此外,由于控制网络有实时性方面的要求,因此对网络通信的速度也有较高的要求。

1.1.3 计算机控制系统的软件组成

上述硬件只能构成计算机的裸机,仅为计算机控制系统的躯体。要使计算机正确地运行以解决各种问题,必须为它编制各种程序。软件是各种程序的统称,是控制系统的灵魂。因此,软件的优劣直接关系到计算机的正常运行、硬件功能的充分发挥及其推广和应用。控制计算机的软件通常分为系统软件和应用软件两大类。

系统软件是一系列支持系统开发、测试、运行和维护的工具软件,核心是操作系统。在计算机控制系统中,为了满足实时处理的要求,通常采用实时多任务操作系统。目前,控制计算机常用的操作系统主要有UNIX、Windows NT、Windows 2000、Windows XP、RMX DOS等具有抢先能力的多任务操作系统,在一些要求不高的场合也有使用Windows 98等不具有抢先能力的多任务操作系统,甚至是DOS等单任务系统。在实时多任务操作环境下,要求将应用系统中的各种功能划分成若干任务,并按其重要性赋予不同的优先级,各任务的运行进程及相互间的信息交换由实时多任务操作系统协调控制。另外,系统软件还包括编程语言等辅助工具,编程语言一般为面向过程或对象的专用语言或编译类语言。系统软件一般由计算机厂商以产品形式向用户提供。

应用软件是系统设计人员利用编程语言或开发工具编制的可执行程序。对于不同的控制对象,控制和管理软件的复杂程度差别很大。在一般的计算机控制系统中,以下几类功能模

块是必不可少的，即过程输入模块、基本运算模块、控制算法模块、报警限幅模块、过程输出模块及数据管理模块等。

作为计算机控制系统的设计人员，只有首先了解并会使用系统软件，才能编制出较好的应用软件。设计开发应用软件，已成为当前计算机控制应用领域中最重要的一个方面。

1.2 计算机控制系统的分类

计算机控制系统的分类方法很多，可以按照系统的功能、控制规律或控制设备等进行分类。

1.2.1 按功能及结构分类

计算机控制系统按照功能及结构一般可分为数据采集系统、操作指导控制系统、直接数字控制系统、监督计算机控制系统、分散控制系统、现场总线控制系统及以太控制网络系统等几类。

1. 数据采集系统

数据采集系统 (DAS, Data Acquisition System) 是计算机应用于生产过程控制最早、也是最基本的一种类型，如图 1-3 所示。生产过程中的大量参数由仪表发送并经 A/D 通道或 DI 通道巡回采集后送入计算机，由计算机对这些数据进行分析处理，并按操作要求进行屏幕显示、制表打印和越限报警。该系统可以代替大量的常规显示、记录报警仪表，对整个生产过程进行集中监视。因此，该系统对于指导生产以及建立或改善生产过程的数学模型，具有重要的作用。

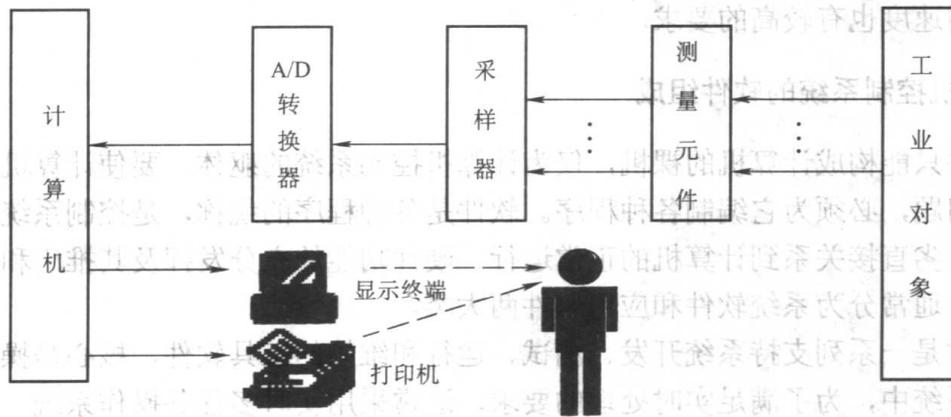


图 1-3 数据采集系统

2. 操作指导控制系统

操作指导控制 (OGC, Operation Guide Control) 系统是基于数据采集系统的一种开环系统，如图 1-4 所示。计算机根据采集到的数据以及工艺要求进行最优化计算，计算出的最优操作条件，并不直接输出到控制生产过程，而是显示或打印出来，操作人员据此去改变各个控制器的给定值或操作执行器，以达到指导操作的目的。显然，这属于计算机离线最优控制的一种形式。

操作指导控制系统的优点是结构简单、控制灵活和安全；缺点是要由人工操作，速度受

到限制，不能同时控制多个回路。因此，操作指导控制系统常用于计算机控制系统设置的初级阶段，或用于试验新的数学模型、调试新的控制程序等场合。

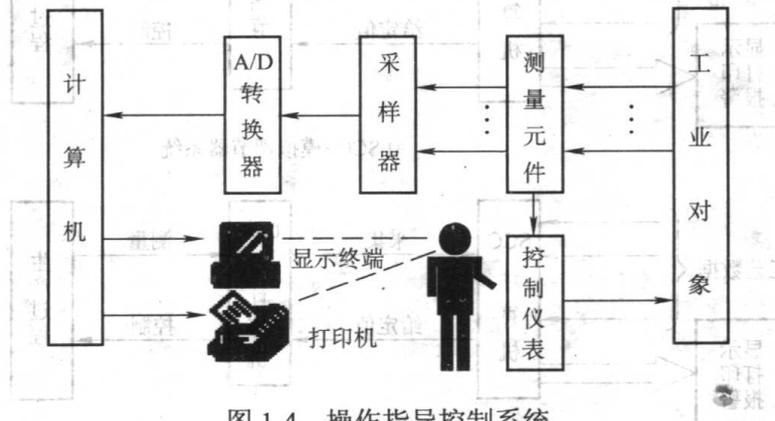


图 1-4 操作指导控制系统

3. 直接数字控制系统

直接数字控制（DDC, Direct Digital Control）系统是用计算机不仅完成对多个被控参数的数据采集，而且能按一定的控制规律进行实时决策，并通过过程输出通道发出控制信号，实现对生产过程的闭环控制，如图 1-5 所示。为了操作方便，DDC 系统还配置一个包括给定、显示、报警等功能的操作控制台。

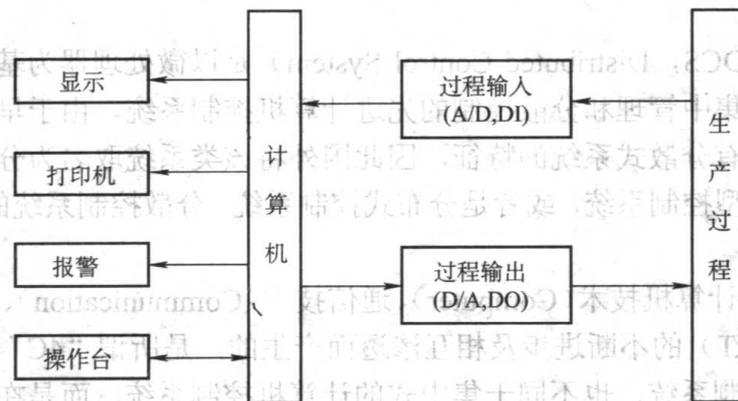


图 1-5 直接数字控制系统

在 DDC 系统中，一台计算机就可以完全取代多个模拟调节器，而且在各个回路的控制方案上，可以不改变硬件而只通过改变软件程序，就能有效地实现各种复杂控制。因此，DDC 系统是计算机在工业生产中最普遍的一种应用方式。

4. 监督计算机控制系统

监督计算机控制（SCC, Supervisory Computer Control）系统是 OGC 系统与常规仪表控制系统或 DDC 系统综合而成的两级系统，如图 1-6 所示。SCC 系统有两种不同的结构形式：一种是 SCC+模拟调节器系统（也可称计算机设定值控制系统，即 SPC 系统）；另一种是 SCC+DDC 控制系统。其中，作为上位机的 SCC 计算机按照描述生产过程的数学模型，根据原始工艺数据与实时采集的现场变量计算出最佳动态给定值，送给作为下位机的模拟调节器或 DDC 计算机，由下位机控制生产过程。这样，系统就可以根据生产工况的变化，不断地修正给定值，使生产过程始终处于最优工况。显然，这属于计算机在线最优控制的一种形式。

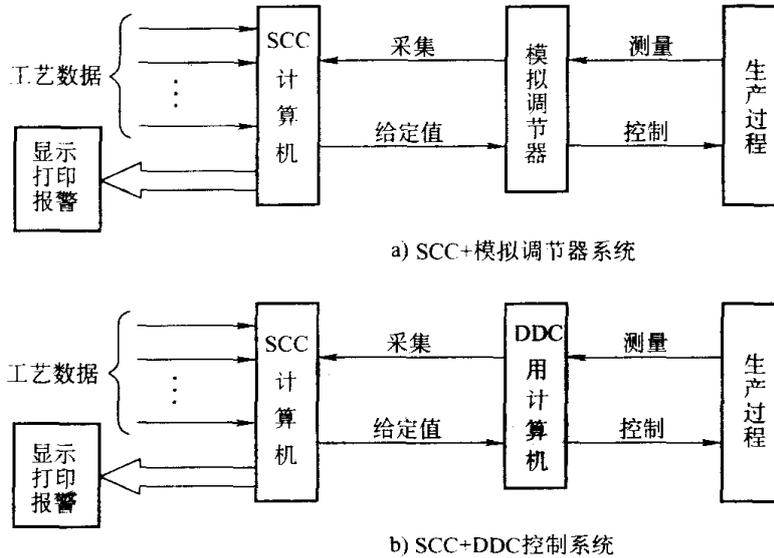


图 1-6 监督计算机控制系统的两种结构形式

另外，当上位机出现故障时，系统要求下位机可以独立完成控制功能。下位机直接参与生产过程控制，要求其实时性好、可靠性高和抗干扰能力强；而上位机承担高级控制与管理任务，应配置数据处理能力强、存储容量大的高档计算机。

5. 分散控制系统

分散控制系统（DCS, Distributed Control System）是以微处理器为基础，借助于计算机网络对生产过程进行集中管理和分散控制的先进计算机控制系统。由于早期开发的分散控制系统在体系结构上具有分散式系统的特征，因此国外将该类系统取名为分散控制系统。国内也有人将其称为集散型控制系统，或者是分布式控制系统。分散控制系统的典型结构如图 7-1 所示。

DCS 是随着现代计算机技术(Computer)、通信技术(Communication)、控制技术(Control)和图形显示技术(CRT)的不断进步及相互渗透而产生的，是所谓“4C”技术的结晶。它既不同于分散的仪表控制系统，也不同于集中式的计算机控制系统，而是在吸收了两者优点的基础上发展起来的具有崭新结构体系和独特技术风格的新型自动化系统。DCS 通过计算机网络将每个分散的过程控制装置和各种操作管理装置有机结合起来，不仅具有先进、可靠的控制性能和集中化的监视、操作功能，而且还有强大的信息处理能力、数据交换能力，以及灵活的构成方式，因而能够适应工业生产过程的各种需要，表现出顽强的生命力和显著的优越性。

6. 现场总线控制系统

现场总线控制系统（FCS, Fieldbus Control System）是新一代分布式控制结构。该系统改进了 DCS 成本高和由于各厂商的产品通信标准不统一而造成的不能互连等弱点，采用集管理、控制功能于一身的工作站与现场总线智能仪表的二层结构模式，把原 DCS 控制站的功能分散到智能型现场仪表中去。每个现场仪表（如变送器、执行器）均作为一个智能节点，都带有 CPU 单元，可分别独立完成测量、校正、调节、诊断等功能，靠网络协议把它们连接在一起统筹工作。这种彻底的分散控制模式使系统更加可靠，现场总线控制系统的典型结构如图 7-12~图 7-14 所示。

FCS 的核心是现场总线，它将当今网络通信与管理的概念引入工业控制领域。从本质上说，现场总线是一种数字通信协议，是连接智能现场设备和自动化系统的数字式、双向传输、多分支结构的串行通信网络。FCS 代表了今后工业控制体系结构发展的一种方向。

7. 以太控制网络系统

图 1-7 是基于以太控制网络技术的多总线集成控制系统的典型结构。目前，以太控制网络在工业自动化和过程控制领域得到了迅猛的发展。

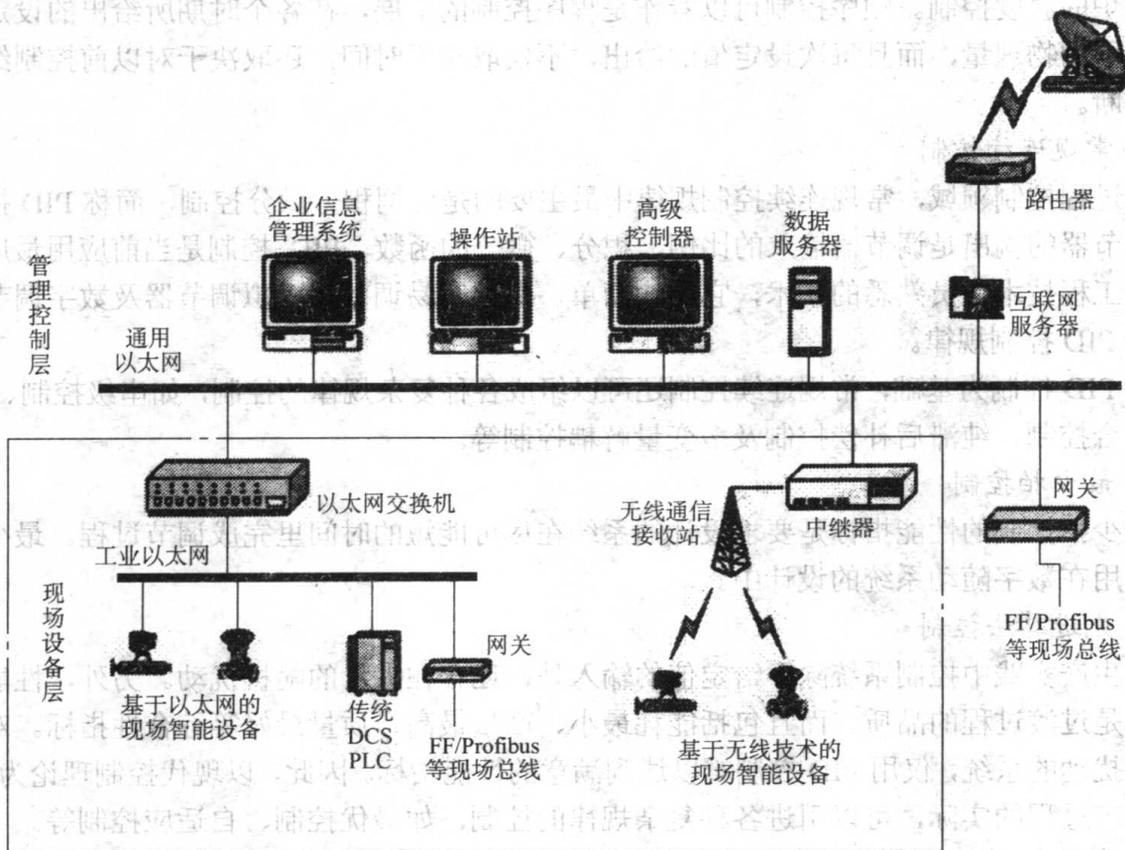


图 1-7 基于以太控制网络技术的多总线集成控制系统

以太网与其他控制网络相比具有很多优点，主要体现在以下几个方面：

- 1) 开放性：采用公开的标准和协议；
- 2) 平台无关性：具有伸缩性，可以选择不同厂家、不同类型的设备和服务；
- 3) 信息服务：能提供 E-mail、WWW、FTP 等多种信息服务；
- 4) 图形用户界面：具有统一、友好、规范化的图形界面，操作简单、易学易用；
- 5) 信息传递速度：信息传递快速、准确；
- 6) 多现场总线的集成能力：能够相互包容，易于集成多种现场总线协同完成测控任务；
- 7) 多系统集成能力：实现现场总线控制系统与传统控制系统的集成、各种现场总线控制系统之间的集成；以太控制网络易与信息网络集成，组成统一的企业网络。集成主要体现在现场通信协议的相容、不同系统数据的交换以及组态、监控、操作界面的统一；
- 8) 多技术集成能力：易于实现设备互操作性技术、OPC (OLE for Process Control) 技术、Ethernet 技术、TCP/IP 技术、Web 技术、现场总线设备管理技术和无线通信技术的集成。

1.2.2 按控制规律分类

计算机控制系统按照控制规律主要可分为程序控制和顺序控制、常规连续控制、最少拍控制、先进算法控制及智能控制等几类。

1. 程序控制和顺序控制

在程序控制中，被控量按照一定的、预先规定的时间函数变化，被控量是时间的函数，如单晶炉的温度控制。顺序控制可以看作是程序控制的扩展，在各个时期所给出的设定值可以是不同的物理量，而且每次设定值的给出，不仅取决于时间，还取决于对以前控制结果的逻辑判断。

2. 常规连续控制

在过程控制领域，常规连续控制规律中最主要的是比例积分微分控制，简称 PID 控制。这里调节器的输出是调节器输入的比例、积分、微分的函数。PID 控制是当前应用最广、最为广大工程技术人员熟悉的技术，它结构简单、参数容易调整，模拟调节器及数字调节器大多使用 PID 控制规律。

以 PID 控制为基础，常规连续控制还可以组成各种复杂规律的控制，如串级控制、前馈-反馈复合控制、纯滞后补偿控制及多变量解耦控制等。

3. 最少拍控制

最少拍控制的性能指标是要求设计的系统在尽可能短的时间里完成调节过程。最少拍控制通常用在数字随动系统的设计中。

4. 先进算法控制

在生产实践中控制系统除了给定值的输入外，还存在大量的随机扰动。另外，性能指标也不单是过渡过程的品质，而且包括能耗最小、产量最高、质量最好等综合性指标。对于存在随机扰动的系统，仅用 PID 控制难以达到满意的性能指标。因此，以现代控制理论为基础，针对生产过程的实际，可以引进各种复杂规律的控制，如最优控制、自适应控制等。

5. 智能控制

智能控制理论是一种把先进的方法学理论与解决当前技术问题所需要的系统理论结合起来的学科。智能控制理论可以看作是人工智能、运筹学和控制理论这三个主要理论领域的交叉或汇合，典型的智能控制系统有模糊控制、神经网络控制及遗传算法等。

1.2.3 按控制设备分类

计算机控制技术发展至今，为适应不同行业、不同工艺设备的需求，计算机制造厂家形成了几种典型的常用机型，这样按计算机控制设备来分，又可以将计算机控制系统分为可编程序控制器、可编程调节器、单片微型计算机、总线式工控机、分散控制系统、现场总线控制系统、工业以太控制网络系统等几大类。其中后三种系统已在前面介绍过了，下面主要对由可编程序控制器、可编程调节器、单片微型计算机和总线式工控机构成的控制系统从结构、原理和应用特点等方面作一简单介绍。

1. 可编程序控制器

可编程序控制器 (Programmable Logic Controller, PLC) 是计算机技术和继电逻辑控制概念相结合的产物，其低端为常规继电逻辑控制的替代装置，而高端为一种高性能的工业控

制计算机。

1985年1月,IEC(国际电工委员会)作了如下定义:PLC是一种数字运算操作的电子系统,专为工业环境下应用而设计,它采用可程序的存储器,用来在其内部存储执行逻辑运算、顺序控制、定时、计数和算术操作的指令,并通过数字式、模拟式的输入和输出,控制各种类型的机械或生产过程。可程序控制器及其有关设备,都应按易于使工业控制系统形成一个整体、易于扩充其功能的原则设计。

PLC在目前工业控制中应用最为广泛,它具有以下鲜明的特点。

1) 系统构成灵活,扩展容易,以开关量控制为其特长;也能进行连续过程的PID回路控制;并能与上位机构成复杂的控制系统,如DDC和DCS等,实现生产过程的综合自动化。

2) 使用方便,编程简单,采用简明的梯形图、逻辑图或语句表等编程语言,而无需复杂的计算机知识,因此系统开发周期短,现场调试容易。另外,可在线修改程序,改变控制方案而不必拆动硬件。

3) 能适应各种恶劣的运行环境,抗干扰能力强,可靠性强,远优于其他各种控制机型。

2. 可编程调节器

可编程调节器(Programmable Controller),又称单回路调节器、智能调节器、数字调节器。它主要由MPU(Micro Processor Unit)单元、过程I/O单元、面板单元、通信单元、硬手操单元和编程单元等组成。

可编程调节器实际上是一种仪表化了的微型控制计算机。它既保留了仪表面板的传统操作方式,易于为现场人员接受;又发挥了计算机软件编程的优点,可以方便、灵活地构成各种过程控制系统。但是,它又不同于一般的控制计算机,系统设计人员在硬件上无需考虑接口、信号传输和转换等问题,在软件编程上也只需使用一种面向问题的POL(Problem Oriented Language)语言。

这种POL组态语言为用户提供了几十种常用的运算和控制模块。其中,运算模块不仅能实现各种组合的四则运算,还能完成各种函数运算。通过控制模块的系统组态编程能够实现各种复杂的控制过程,诸如PID、串级、比值、前馈、选择、非线性及程序控制等。这种系统组态方式既简单易学,又便于修改与调试,因此,极大地提高了系统设计的效率。

同时,可编程调节器一般还具有断电保护和自诊断功能,系统的可靠性可以得到很好的保证。另外,可编程调节器的通信单元(通信接口)使之能与集中监视操作站、上位机进行通信,组成多级计算机控制系统,可实现各种高级控制和管理。因此,可编程调节器不仅可以作为大型分散控制系统中最基层的控制单元,而且可以在一些重要场合下单独构成复杂控制系统,完成1~4个控制回路,其在过程控制中的广泛应用是不言而喻的。

3. 单片微型计算机

随着微电子技术与超大规模集成电路技术的发展,诞生了计算机技术的另一个分支——超小型化的单片微型计算机(Single-chip Microcomputer),简称单片机。它抛弃了以通用微处理器为核心构成计算机的模式,充分考虑到控制的需要,将CPU、存储器、串-并行I/O接口、定时/计数器,甚至A/D转换器、脉宽调制器、图形控制器等功能部件全都集成在一块大规模集成电路芯片上,构成了一个完整的具有一定控制功能的微控制器。

单片机从结构上一般可分为两种:一种是将程序存储器和数据存储器分开,分别编址的Harvard结构,如MCS-51系列单片机;另一种是对两者不作逻辑上的区分,统一编址的

Princeton 结构，如 MCS-98/96 系列单片机。

由于单片机具有体积小、功耗低、性能可靠、价格低廉、功能扩展容易、使用方便灵活、易于产品化等诸多优点，特别是它所具有的强大的面向控制的能力，使它在工业控制、智能仪表、外设控制、家用电器、机器人、军事装置等方面得到了极为广泛的应用。

但单片机自身的特点和应用场合又决定了单片机应用系统的开发与一般计算机不同。由于单片机是面向控制设计的，专用性强、内存容量小、人-机接口功能不强，因此，单片机本身不具备自开发能力，必须借助于仿真器或开发系统与单片机联机，才能进行硬、软件的开发与调试。

单片机的应用软件多采用面向机器的汇编语言，而汇编语言的通用性与可移植性差，开发应用软件需要具备较深的计算机软件和硬件知识。随着高效率结构化语言的发展，其软件开发环境已得到逐步改善。目前，市场上已推出了面向单片机结构的高级语言，如 Intel 公司的 PL/M 结构化程序设计语言和 C 语言等。

单片机的应用从 4 位机开始，历经 8 位、16 位、32 位四种。但在小型测控系统与智能化仪器仪表的应用领域里，8 位和 16 位单片机因其品种多、功能强、价格低廉，目前仍然是单片机系列的主流机种。

4. 总线式工控机

随着计算机硬件设计的日益科学化、标准化与模块化，一种总线系统和开放式体系结构的概念应运而生。总线即是一组信号线的集合，一种传送规定信息的公共通道。它定义了各引线的信号特性、电气特性和机械特性。按照这种统一的总线标准，计算机厂家可设计制造出若干具有某种通用功能的模板，而系统设计人员则可以根据不同的生产过程，选用相应的功能模板组合成自己所需的计算机控制系统。

采用总线技术研制生产的控制计算机系统被称为总线式工控机。总线式工控机一般是在一块无源的并行底板总线上，插接多个功能模板构成。除了构成计算机基本系统的 CPU、RAM/ROM 和人-机接口板外，还有 A/D、D/A、DI、DO 等数百种工业 I/O 接口和通信接口板可供选择，其选用的各个模板彼此通过总线相连，均由 CPU 通过总线直接控制数据的传送和处理。

这种系统结构的开放性方便了用户的选用，从而大大提高了系统的通用性、灵活性和可扩展性。模板结构的小型化，使它的机械强度高，抗振动能力强；模板功能的单一，则便于对系统故障的诊断与维修；模板的线路设计布局合理，即由总线缓冲模块到功能模块，再到 I/O 驱动输出模块，使信号流向基本为直线，这些都大大提高了系统的可靠性和可维护性。另外，在结构配置上还采取了很多措施，如密封机箱正压送风、使用工业电源及带有 WatchDog 系统支持等。

总之，总线式工控机具有小型化、模板化、组合化、标准化的设计特点，可满足不同层次、不同控制对象的需要，又能在恶劣的工业环境中可靠地运行，因而其应用极为广泛。目前国内工控领域的机型主要有 STD 总线工控机和 PC/PCI 总线工控机两大类，由于 PC/PCI 总线工控机与普通 PC 软、硬件资源完全兼容，使其逐渐成为了工控领域的主流机型。

随着微电子技术的发展，上述这些不同的机型在性能与应用上已有互相交叉和覆盖的趋势。由于 PLC、可编程调节器、单片机等均作为一种应用广泛而独特的控制装置在有关课程中讲授，因此本书中的计算机控制系统主要是指总线式工控机，同时，这部分知识又是 DCS、

FCS 以及工业以太网系统的基础, 为了使知识系统更加完整, 本书同时也对 DCS、FCS 等系统进行了简要介绍。

1.3 计算机控制系统的特点

计算机控制系统与模拟控制系统相比, 主要具有以下特点。

1. 数字、模拟混合的系统

在连续控制系统中, 各处的信号是连续模拟信号, 而在计算机控制系统中, 除仍有连续模拟信号外, 还有离散信号、数字信号等多种信号。因此, 计算机控制系统是模拟和数字的混合系统。

2. 灵活方便、适应性强

一般的模拟控制系统中, 控制规律是由硬件电路实现的, 控制规律越复杂, 所需要的模拟电路往往越多, 如果要改变控制规律, 一般就必须更改硬件电路。而在计算机控制系统中, 控制规律是由软件实现的, 计算机只需执行预定的控制程序, 就能实现对被控参数的控制。需要改变控制规律时, 一般不对硬件电路做改动, 只要改变控制程序就可以了。

3. 可实现复杂控制规律

计算机具有丰富的指令系统和很强的逻辑判断功能, 能够实现模拟电路不能实现的复杂控制规律。

4. 离散控制

在连续控制系统中, 给定值与反馈值的比较是连续进行的, 控制器对产生的偏差也是连续调节的。而在计算机控制系统中, 计算机每隔一段时间间隔, 向 A/D 转换器发出启动转换信号, 并对连续信号进行采样获得离散时间信号, 经过计算机处理后, 产生的控制时间信号通过 D/A 转换器将离散信号转换成连续时间信号输出, 作用于被控对象。因此, 计算机控制系统并不是连续控制的, 而是离散控制的。

5. 可以采用分时控制

在连续控制系统中, 一般是一个控制器控制一个回路。而在计算机控制系统中, 由于计算机具有高速的计算处理能力, 一个数字控制器经常可以采用分时控制的方式, 同时控制多个回路。

6. 易于实现管控一体化

采用计算机控制系统可实现控制信息的全数字化, 易于建立集成企业经营管理、生产管理和过程控制于一体的管控一体化系统, 即建立集成了生产过程控制系统、生产执行系统和企业资源计划 (ERP) 系统的综合自动化系统。

1.4 计算机控制系统的主要研究内容

计算机控制系统内含各种丰富的功能, 以下内容是设计或应用过程中经常遇到的问题。

1.4.1 输入数据处理

对模拟量来说, 一般要进行采样、增益最佳化、信号的规格化、合理性检查、零偏校正、

热电偶冷端补偿、非线性校正、越限处理、工程量变换、数字滤波、开方计算、上下限报警等处理，以及对脉冲序列进行瞬时值变换及累积计算等。

1) 数据的采样：按系统不同的需要，采取不同的扫描采样频率，进行模数转换 (A/D)。

2) 增益最佳化：模拟量信号在 A/D 转换之前要进行前置放大，以使被转换量落在 A/D 转换线性范围之内 (通常在 50%~100%满度范围内)，以提高通道的相对测量精度。因此要选择合适量程，增益最佳化即能自动挑选最佳增益。

3) 信号的规格化：模拟量信号的规格化是指 1~5V 的模拟信号经 A/D 转换后变成规格化的数字量。

4) 合理性检查：假如 A/D 转换器超出限定时间或接到指令后根本未进行转换，则应根据故障置位给出不合理标志。如果是 A/D 转换器超量程或欠量程 (小于下限值)，该数将被进一步处理，给出读数不合理标志。

5) 零偏校正：由温度、电源等环境因素变化引起放大器零点漂移，可通过软件进行校正。通常是把输入短路时采集的放大器零原码取平均值存入内存，然后在当前测量结果中扣除此零漂值。

6) 工程量转换：上位机需显示或打印时，还应将规格化的数据转换成工程量单位值。

7) 越限处理：参数超限时，一般均需报警，通常分为绝对值报警、偏差报警、速度报警以及累计值报警。

8) 热电偶冷端补偿：对于安装在现场多路切换箱中的热电偶，其冷端温度自动补偿是通过一支专用的冷端温度检测热电偶进行的；输入处理时先接通一次，测量工作电动势；再短路一次，测量短路电动势，两者相减即可消除外电路影响。

9) 非线性校正：对于温度与热电动势 (mV) 数值或热电阻 (Ω) 数值间的非线性关系，可通过折线近似或曲线拟合的方法加以校正。采用曲线拟合法时，多采用高次方程。

10) 开方计算：对于具有二次方特性的数据，常需进行开方处理，才能使测量数据与被测物理量成线性关系，如节流式流量计的差压信号需开方才能转换成流量。

11) 热电偶开路检查：测点在被切断之前，用一小电流流过模拟信号线、多路切换电路和转换器，假如通过外电路的压降超过正常值，就表明热电偶已经开路。

12) 数字滤波：数字滤波就是为克服随机干扰引入的误差而进行的数据平滑处理。常用的数字滤波算法有程序判断法、中位值法、算术平均滤波、递推平均滤波、加权平均滤波、一阶惯性滤波和复合滤波等。

1.4.2 输出处理

计算机系统的输出一般分为模拟量输出和开关量输出两类。

模拟量输出时，CPU 算出的数字结果再经 D/A 转换成 4~20mA 信号送端子板输出，在输出带限幅信号时可经限幅处理。在 D/A 转换之前，数据先与限幅信号进行比较，正常时将输出送入 D/A 转换器，反之将限幅值送入 D/A 转换器。

在开关量输出时，由主机电路送出数字信号，先存入输出锁存器，再经驱动电路进行功率放大，送控制现场执行机构。数字输出可有三种不同的形式，即瞬时输出、延时输出及锁定输出。

1.4.3 控制功能

控制部分是计算机控制系统的核心。按其所具有回路数的多少,可分为单回路、双回路和多回路控制。多回路的控制最多达到数百个回路。控制算法的数目从几十种到上百种。随着计算机技术和控制理论的发展,控制功能变得越来越强大。控制方式从早期只有连续控制功能的传统控制,已发展到普遍具有逻辑控制、顺序控制、批量控制和各种智能控制方式。

1. 连续控制

严格说来,计算机控制全部是离散控制,但为区别于顺序控制和逻辑控制,还是称它为连续控制。连续就是调节器随输入信号的不断变化按一定规则输出,不间断地修正输出值的大小。

连续控制算法一般有PID控制、反馈-前馈复合控制、自适应控制、非线性控制、Smith预估控制、多变量解耦控制等常规传统控制算法,还有模糊控制和神经网络控制等智能控制算法。

除了各种算法之外,连续控制还必须具有一些附加功能才可以完成实际的过程控制。这些附加功能如下。

1) 手动方式 (MAN: MANUAL)。

2) 自动方式 (AUT: AUTO),以设定值为基准进行运算,实现闭环控制方式。

3) 串级方式 (CAS),以另一个控制回路的输出值作为本回路的设定值进行自动运算,实现串级控制方式。

4) 计算机方式 (COMP),由数据通信线路接收计算机的输出数据,作为本回路设定值进行自动运算的控制方式 (SPC),或者直接接收计算机的输出数据作为输出值的控制方式 (DDC)。

5) PV跟踪, PID算法具有PV跟踪功能,即在手动状态时,使本回路的设定值不再保持原来的设定值,而跟踪PV值。PV跟踪时,从手动切换到自动偏差总为零。因此, PID输出值不会产生扰动,实现了平滑切换。切换到自动后,再逐步把设定值调整到所要求的数值。

6) 输出值跟踪, PID算法在设置PV跟踪的同时,还需要设置输出值跟踪功能,即在手动状态时,使内存上次输出单元的数值跟踪手操输出值。这样,在切换到自动时,由于输出单元数值与手操输出值相等,从而实现无扰动切换。

7) 反算,当串级调节主回路PID算法输出插有静态偏置时,算法必须具有反算功能,即算法块(如加法功能块)必须反算出(逆运算)一个平衡无扰的输入值,作为主回路的输出。

2. 逻辑控制

根据输入变量的状态,按逻辑关系进行的控制称为逻辑控制。在计算机控制系统中,由逻辑块实现逻辑控制功能。逻辑运算包括AND(与)、OR(或)、XOR(异或)、LINK(连接)、OFFDELAY(停止延时)、ONDELAY(进行延时)、FILP-FLOP(触发器)及PULSE(脉冲)等。它们可直接用于过程控制,进行工艺连锁,也可作为顺序控制中的功能块,进行条件判断、状态转换等。

3. 顺序控制

顺序控制是根据预定顺序或逻辑,逐步进行各阶段信息处理的控制方法。在顺序控制中兼有连续控制、逻辑控制和输入/输出监控的功能。

4. 批量控制

批量控制就是利用顺序程序控制一个间断的生产过程，以得到规定的产品。例如，研制生产一种催化剂溶液，需经投料、加放一定量溶剂、搅拌并加热到一定温度、保温至规定时间，最后过滤排放这一过程。每个操作都不是连续的，但有相应的要求，每步的转移又都必须满足一定的条件。这里除了要进行连续常规的温度、流量闭环调节外，还需要执行打开阀门、启动搅拌等开关控制及判断，要用顺序程序把这些操作按次序连接起来，需要定义每步操作的具体条件和要求，直接指挥有关的现场控制设备，以得到满意的产品。因此，批量控制是顺序控制的一种应用。

1.4.4 人-机接口功能

按照现场控制的连接方式，计算机控制系统的人-机接口可分为现场简易接口和 CRT 操作站两种。人-机接口可以直接与现场装置相连，也可以经过通信网络和现场装置相连。简易操作接口，如开关、电位器、按键、数码显示器等器件，使现场工作人员能完成对系统进行操作、监视和设定的基本功能。CRT 操作站不仅能在屏幕上、键盘上实现上述基本功能，还能自动地对信息进行分析、判断和综合。人-机接口的主要功能归纳起来有以下几个方面：

- 1) 对全部过程变量进行各种格式的显示，并允许操作人员对过程进行干预，如参数调整、状态切换及紧急处理等；
- 2) 对过程变量的历史数据进行处理和存储；
- 3) 显示和打印过程报警；
- 4) 编制和调用各种用户显示画面；
- 5) 编制和打印各种报表；
- 6) 组态和装载监视、控制和管理软件；
- 7) 对通信总线上所连接的设备进行组态，并将组态结果装载到对应设备；
- 8) 完成系统的自诊断和故障维修。

1.4.5 通信功能

通信功能是计算机控制系统的重要支柱。分散控制的各单元以及各级人-机接口，都要靠通信系统连成一体。控制计算机的通信系统是一个高通信速率、低误码率、快速响应的网络，具有组织灵活、易于扩展、资源共享等特点。

1.4.6 自诊断功能

为了提高可靠性，计算机控制系统的各装置具有较强的自诊断功能。系统投运前，用离线诊断程序检查各部分工作状态；系统投运中，各设备不断执行在线自诊断程序，一旦发现错误，立即切换到备用设备，同时经过通信网络在 CRT 上显示出故障代码，等待及时处理。通常故障代码可以定位到插件板，用户只需更换插卡。

1.4.7 冗余技术

自诊断可以及时检查出故障，但要保证计算机控制系统的运行不受故障的影响，这主要靠冗余技术的应用。冗余有两种方式：工作冗余和后备冗余，俗称“热备用”和“冷备用”。

操作站常采用工作冗余的方式。对现场控制器,根据冗余方式的不同,有的采用1:1冗余,也有的为N:1冗余,但均采用无中断自动切换方式。

计算机控制系统还特别重视供电系统的可靠性,除了正常的交流供电外,还要有备用电源、不间断电源(UPS)以及各种掉电保护措施。

计算机控制的网络系统中大多采用两套相同通信网络的冗余形式,但具体工作情况有所不同,工作冗余和后备冗余两种情况都有。

除了硬件冗余外,计算机系统还采用了信息冗余技术,就是在发送信息的末端增加信息位,以提高检错及纠错能力。

1.5 计算机控制系统的发展

1.5.1 计算机控制技术的发展历程

20世纪50年代初,计算机就开始用于工业生产过程控制。控制理论与计算机的结合,产生了新型的计算机控制系统,为自动控制系统的 application 与发展开辟了新的途径。

从美国的计算机控制技术发展状况来看,大体分为以下三个阶段。

1. 试验阶段

1965年以前,是试验阶段。1952年,在化工生产中实现了计算机自动测量和数据处理。1954年,开始用计算机构成开环控制系统。1957年,在石油蒸馏过程控制中采用了计算机构成的闭环系统。1959年,在一个炼油厂建成了第一台闭环计算机控制装置。1960年,在合成氨和丙烯腈生产过程中实现了计算机监督控制。1962年,在一个乙烯工厂实现了直接数字控制(DDC)系统。

2. 实用普及阶段

1965~1969年是计算机控制进入实用普及的阶段。由于小型计算机的出现,使其可靠性不断提高,成本逐年下降,计算机在生产控制中的应用得到迅速发展。但这个阶段仍然主要是集中型的计算机控制系统,在高度集中的控制系统中,若计算机出现故障,将对整个装置和生产系统带来严重影响。虽然采用多机并用可以提高集中控制的可靠性,但会增加成本。

3. 推广分级控制阶段

1970年以后计算机控制进入了大量推广分级控制阶段。将计算机分散到生产装置中去,实现小范围的局部控制和某些特殊控制规律。这种控制方式称为“分散型计算机控制系统”或称“集散控制系统”。特别是由于微型机具有可靠性高、价格便宜、体积小、使用方便、灵活等特点,为分散型计算机控制系统的发展创造了良好的条件。

20世纪90年代以后,随着现场总线控制技术的逐渐成熟和以太网技术的逐步普及,以及智能化与功能自治性的现场设备的广泛应用,使嵌入式控制器、智能现场测控仪表和传感器可以方便地接入现场总线和工业以太网,直至与Internet相连。

纵观控制系统的发展历史,人们一般把20世纪50年代前的气动信号控制系统(PCS)称为第一代;把4~20mA等电动模拟信号控制系统称为第二代;把数字计算机集中式控制系统称为第三代;把20世纪70年代中期以来的集散式分布控制系统(DCS)称为第四代;把20世纪90年代发展起来的现场总线控制系统(FCS)称为第五代;而把正在出现的工业以太

网控制系统称为第六代。

在发展过程中由于市场的推动和应用的需要，计算机技术总在技术上领先于控制技术，一般来说计算机技术所产生的新概念、新理论也逐渐被控制技术所吸收，并在时间上大约滞后 10 年。当前，计算机控制系统正逐渐形成 Internet、Intranet、Infranet (Infrastructure networks 的缩写) 相互集成的发展趋势。

1.5.2 计算机控制技术的发展动向

纵观目前的计算机控制技术的发展，其动向主要体现在以下几个方面。

1. PLC 向微型化、网络化、PC 化和开放性方向发展

微型化、网络化、PC 化和开放性是 PLC 未来发展的主要方向。随着软 PLC 控制组态软件的进一步完善和发展，安装有软 PLC 组态软件和基于 PC 控制的 PLC 市场份额将逐步得到增长。当前，过程控制领域最大的发展趋势之一就是 Ethernet 技术的扩展，PLC 也不例外。现在，越来越多的 PLC 供应商开始提供 Ethernet 接口。

2. 工业控制网络向有线和无线相结合方向发展

计算机网络技术、无线技术以及智能传感器技术的结合，产生了“基于无线技术的网络化智能传感器”的全新概念。这种基于无线技术的网络化智能传感器使得工业现场的数据能够通过无线链路直接在网络上传输、发布和共享。无线局域网技术能够在工厂环境下，为各种智能现场设备、移动机器人以及各种自动化设备之间的通信提供高带宽的无线数据链路和灵活的网络拓扑结构，在一些特殊环境下有效地弥补了有线网络的不足，进一步完善了工业控制网络的通信性能。

3. 工业控制软件向先进控制方向发展

工业控制软件主要包括人-机界面软件、基于 PC 的控制软件以及生产管理软件等。目前，中国已开发出一批具有自主知识产权的实时监控软件平台、先进控制软件、过程优化控制软件等成套应用软件。作为工控软件的一个重要组成部分，近几年，国内人-机界面组态软件研制方面取得了较大的进展，软件和硬件相结合，为企业测、控、管一体化提供了比较完整的解决方案。

4. 工业自动化仪器仪表技术正在向数字化、智能化、网络化及微型化方向发展

仪器仪表向智能化方向发展，将产生智能仪器仪表；测控设备 PC 化，使虚拟仪器技术得到迅速发展；仪器仪表网络化，将产生网络仪器与远程测控系统。

5. 现代通信与网络技术在现代控制领域广泛渗透

现代通信与网络技术在现代控制领域中的广泛渗透对计算机控制网络提出了新的技术要求。主要体现在：系统的开放性与数字式互连网络性，实现系统的全分散智能控制；现场设备的智能化与功能自治性；互操作与互用性；对现场环境的适应性；很高的实时性与良好的时间确定性；很强的容错能力与高可靠性、高安全性；高度的集成性（包括系统的集成与技术的集成），而设备的功能块技术和现场总线描述技术为设备的互操作性奠定了基础；OPC 技术则使监控或者管理软件与现场设备之间的接口标准化，建立了统一的数据存取规范；工业控制网络采用 Ethernet 技术，将成为工厂底层控制网络的信息传输主干，用以连接系统监控设备和现场智能设备，使工业控制网络融入计算机网络的发展主流，形成面向自动控制领域的 Ethernet，产生了一种基于控制和信息的协议（CIP）的新型 Ethernet（即工业 Ethernet），

它专门为工业场合设计了应用层协议，提供了访问数据和控制设备操作的服务能力；TCP/IP 进入工业现场使得通过 Internet 远程监控生产过程和进行远程系统调试、设备故障诊断成为现实，最为典型的是 Ethernet+TCP/IP 的传感器、变送器可以直接成为网络的节点；现场总线设备实时管理技术则全面、直观地反映现场设备状态，实现可预测性的设备管理与维护模式。

总之，及时、准确、可靠地获得现场设备的信息是计算机控制系统的基本要求，可靠、高效的现场控制网络则是迅速而有效地收集和传送现场生产与管理数据的基本保障。目前，网络技术的迅速发展引发了自动控制领域中深刻的技术变革。计算机控制系统的结构沿着网络化方向、控制系统体系沿着开放性方向发展将是计算机控制技术发展的潮流。网络化、开放化、智能化和集成化是工业控制技术发展的方向。现场总线技术、以太控制网络技术、分布式网络控制技术与企业网络技术的出现及其发展，将推动控制领域的全方位技术进步。

思考题

1. 计算机系统由哪些部分组成？试画出框图。
2. 计算机控制系统是怎样分类的？按功能和控制规律可分为几类？
3. 计算机控制系统的主要特点有哪些？
4. 计算机控制系统的发展方向主要表现在哪几个方面？

第 2 章 过程计算机控制的理论基础

由数字计算机构成的控制系统，在本质上是一个离散时间系统。因此，要研究一个离散时间系统，首先应建立该系统的数学模型，然后解决数学描述方法和分析工具的问题。离散系统与连续系统之间的根本区别在于，在连续系统中，各职能环节的控制信号、反馈信号和偏差信号都是时间上连续的模拟量；而计算机控制系统的信号是时间上离散的数字量，所以取自系统输出端的反馈信号和从数字控制器的输出信号，都是离散时间信号或二进制数字量。在工程上应用的计算机控制系统，通常是连续信号与数字信号混合在一起，通过计算机完成控制任务，因此需要对混合在同一系统中的两种不同类型的信号进行采样、信息变换、处理和恢复。

本章着重介绍计算机控制系统的信号变换原理、系统内部过程的数学模型及不同模型间的等效转换方法、线性离散系统的稳定性分析、采样控制系统的稳态和动态分析等，这是设计一个计算机控制系统的重要理论基础。

2.1 信号变换原理

2.1.1 计算机控制系统的信号流程

数字计算机只能接收和处理二进制代码“0, 1”的组合，这组二进制数可以表示为某一种物理量的大小、某个数据或某些事物，称为离散量或数字信号。在实际系统中，被控参数大多是在时间上的连续信号，称为连续量或模拟量。因此，计算机控制系统也可以称为数字控制系统、离散控制系统或采样控制系统。

在计算机控制系统中，被控对象的模拟量和计算机所能处理的数字量组合在同一系统中，系统中各点信号变换关系如图 2-1 所示。

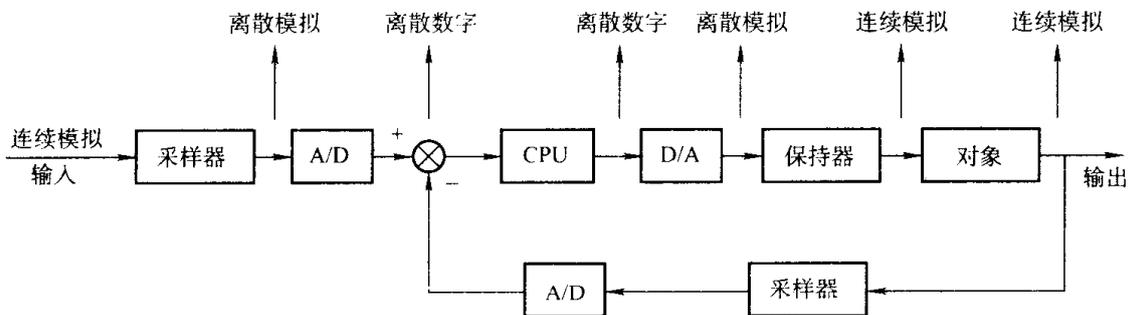


图 2-1 计算机控制系统信号变换关系

计算机控制系统的信号变换过程涉及到下面几个重要概念：

(1) 模拟信号是指在时间和幅值上都是连续的信号，在数学上用连续时间函数 $f(t)$ 表示；离散模拟信号是指在时间上是离散的，而在幅值上表示连续量信号的大小；数字信号表示在

时间上是离散的、在幅值上也是离散的信号。

(2) 采样是指利用采样开关，将模拟信号按照一定的时间间隔重复开闭抽样变成离散模拟信号的过程。采样开关重复开闭的时间间隔 T 称为采样周期。

(3) 量化是指采用一组二进制代码，用来逼近离散模拟信号的幅值并将其转换成数字信号，一般用 A/D 转换器完成。

一个原来在时间上和幅值上是连续的模拟信号，经过按一定采样周期闭合的采样器，把连续模拟信号变成离散模拟信号，经过 A/D 转换（同时完成量化与编码），就变成了数字计算机可以接收，在时间上离散、在幅值上也是离散的二进制数字信号。CPU 对输入信息进行运算处理，输出离散数字信号，经过 D/A 转换器和保持器就可以变成控制对象可以接受的信息。该控制信号是在时间上和幅值上都连续的模拟信号。上述就是计算机控制系统内信号转换的过程。

2.1.2 脉冲采样器和采样过程

计算机对某个随时间变化的模拟量进行采样，是利用定时器控制的开关，每隔一定时间使开关闭合而完成一次采样。开关重复闭合的时间间隔 T 称为采样周期。所谓采样过程，是指将一个连续的输入信号经开关采样后，转变为发生在采样开关闭合瞬时 $0, T, 2T, \dots, nT$ 的一连串脉冲输出信号。

实际采样器是一个模拟电子开关，将输入的连续量 $f(t)$ 转换成脉冲数值序列 $f(kT)$ ，也就是模拟离散量。采样器和信号的输入、输出关系如图 2-2 所示。

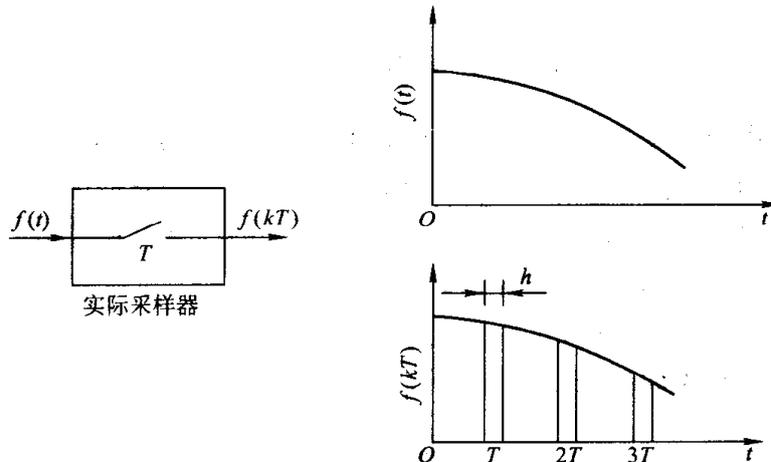


图 2-2 实际采样器和采样的数值序列

图中， T 为采样周期， h 为采样开关闭合时间。采样开关的输出是由一系列的矩形波的离散量所组成， h 决定了数值序列 $f(kT)$ 的宽度， h 应远远小于采样周期 T 。

在数字控制系统中，为了便于应用 z 变换、脉冲传递函数和差分方程等数学工具进行描述、分析和研究，引入了理想脉冲采样器的概念，即在实际采样器的输出端增加一个理想脉冲调制器，在输入端加一个时间函数 $f(t)$ ，则其输出将是一个脉冲序列 $f^*(t)$ 。其数学表达式为

$$f^*(t) = \sum_{k=0}^{\infty} f(kT)\delta(t-kT) \quad (2-1)$$

式中, $f^*(t)$ 为输出脉冲序列; $f(kT)$ 为输出脉冲数值序列; $\delta(t-kT)$ 为发生在 $t=kT$ 时刻上的单位脉冲。

脉冲采样器结构及波形如图 2-3 所示。需要指出, 具有无穷大幅值和持续时间为零的理想单位脉冲纯属数学上的假设, 不会在实际物理系统中发生。因此, 在实际应用中, 只有讲它的面积或强度才有意义。另外, 将采样开关当作理想单位脉冲发生器是近似的、有条件的, 即采样持续时间 h 应远远小于采样周期 T 及系统不可变部分的时间常数。上述条件在实际离散系统中通常是满足的。

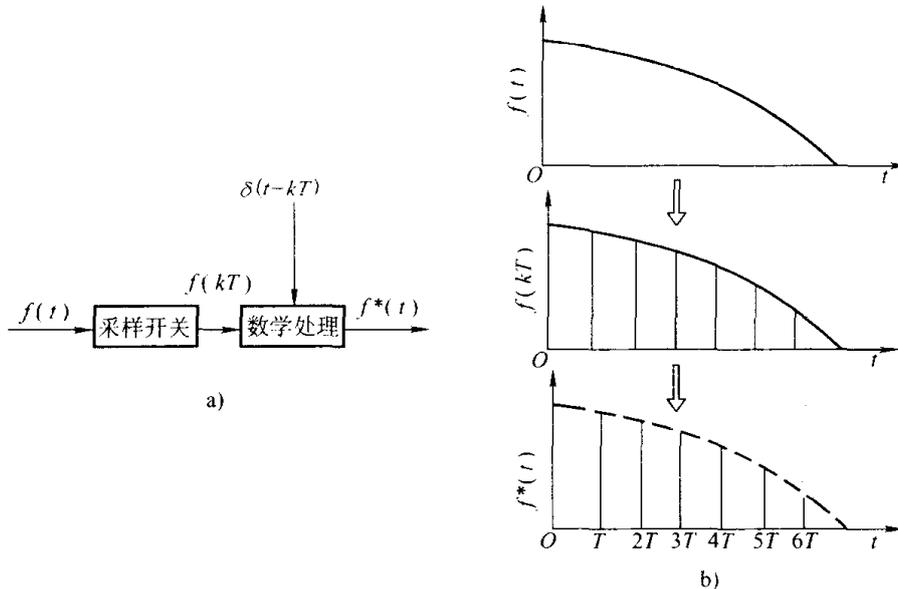


图 2-3 脉冲采样器结构及波形

图中, $f^*(t)$ 的物理意义为, 在每个采样时刻, 脉冲采样器输出一个脉冲, 这个脉冲的高度为无限大, 但它的强度是有限的, 为 $f(kT)$ 。因为 z 变换和脉冲函数在数学上只能处理脉冲序列, 为了应用这些数学工具分析数字控制系统, 把采样开关抽象成一个理想的脉冲采样器是为了方便数学分析, 在物理上是无法实现的。

根据理想单位脉冲函数的定义式 (2-1), 在采样开关闭合时, $f(kT)$ 与 $f(t)$ 的瞬时值相等, 式 (2-1) 还可改写成如下形式, 即

$$f^*(t) = f(t) \sum_{k=0}^{\infty} \delta(t - kT) \quad (2-2)$$

式 (2-2) 说明, 数字系统中的采样过程可以理解为脉冲调制过程。这里采样开关只起着理想脉冲发生器的作用, 通过它将连续信号 $f(t)$ 调制成脉冲序列 $f^*(t)$ 。

对连续时间信号采样的物理意义可以有两种解释, 如图 2-4 所示。图 2-4a 为连续时间信号被单位脉冲序列做了离散时间调制; 图 2-4b 为单位脉冲序列被连续时间信号作了幅值加权。上述两种解释是等价的。

2.1.3 采样定理

由图 2-1 可见, 一个数字控制系统的信息变换过程是先将连续模拟信息 $f(t)$ 通过采样器变成离散的脉冲序列 $f^*(t)$, 再经过 A/D 转换器变成二进制数字代码, 输入到 CPU 处理后,

再将离散数字量经 D/A 转换器及保持器还原成连续模拟量去驱动执行机构。

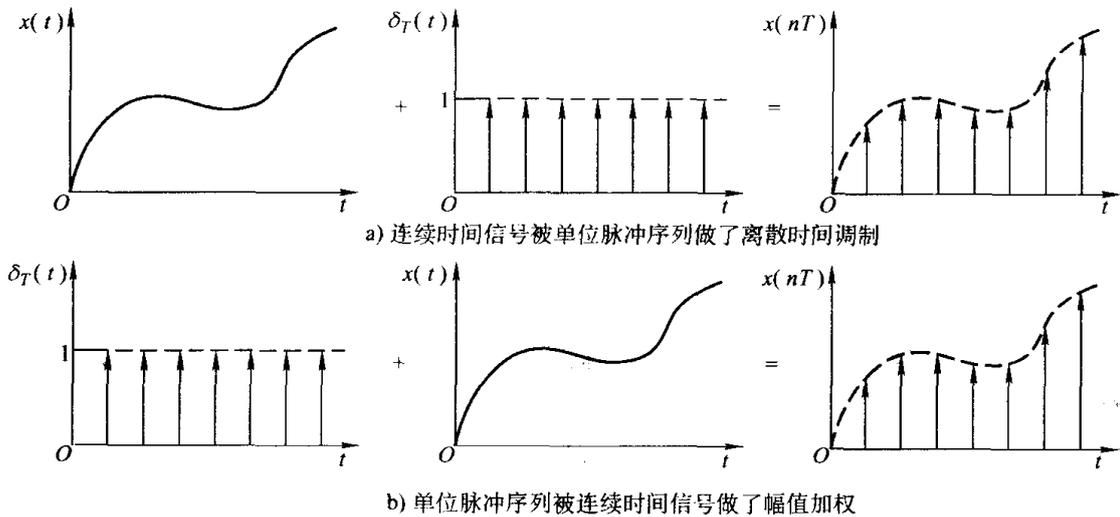


图 2-4 采样信号的物理意义

信息经过两种形式的一系列变换，是由脉冲采样器、A/D 和 D/A 转换器与保持器完成的。那么，在信息变换过程中，就存在一个信息在采样过程中能否完整地保留下来，而在输出控制时又能不失真地恢复连续信号的问题。香农 (Shannon) 采样定理给出了从采样的离散信号恢复到原连续信号所需要的最低采样频率，这个定理是解决计算机控制系统采样周期 T 选择问题的理论根据，它是任何数字控制系统信息变换的基本定理之一。

一个连续时间信号 $f(t)$ ，设其频带宽度是有限的，其最高频率为 ω_{\max} (或 f_{\max})，如果在等间隔点上对该信号进行连续采样，为了使采样后的离散信号 $f^*(t)$ 能包含原信号 $f(t)$ 的全部信息量，则采样角频率 ω_s 只有满足下面的关系

$$\omega_s \geq 2\omega_{\max} \quad (2-3)$$

采样后的离散信号 $f^*(t)$ 才能够无失真地复现 $f(t)$ 。

由式 (2-3) 可知，相邻两个采样点的周期为

$$T \leq \frac{1}{2\pi f_{\max}} \quad (2-4)$$

采样定理实际上表达了这样一种思想：对于连续模拟信号 $f(t)$ ，通常并不需要无限多个连续的时间点上的瞬时值来决定其变化规律，而只需要有各个等间隔点上的离散采样值就够了。这与作一条直线只需要知道两点的道理是类似的。

如果不满足采样定理的要求，为什么就不能恢复原连续信号呢？这是因为此时会产生频率混叠现象，下面采样定理的证明过程可以从理论上回答这一问题。

连续时间信号 $x(t)$ 与其傅里叶变换 $X(\omega)$ 如图 2-5 所示， $X(\omega)$ 的上限频率为 ω_a 。

将其以等间隔采样时间 T 采样，则采样信号 $x^*(t)$ 为

$$x^*(t) = \sum_{k=0}^{\infty} x(kT)\delta(t-kT) \quad (2-5)$$

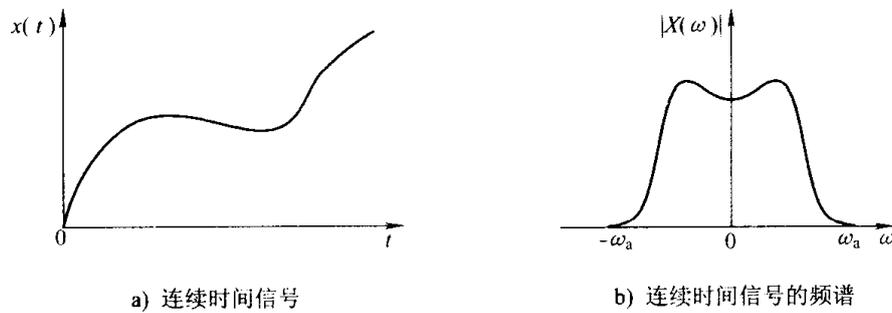


图 2-5 信号的时域表示与频域表示

由于调制信号为单位脉冲序列 $\delta_T(t)$ ，脉冲周期为 T ，是周期函数。对于周期函数可以表示为傅里叶级数

$$\delta_T(t) = \sum_{n=-\infty}^{\infty} C_n e^{-jn\omega_s t} \quad (2-6)$$

$$C_n = \frac{1}{T} \int_{-T/2}^{T/2} \delta_T(t) e^{-jn\omega_s t} dt = \frac{1}{T} \quad (2-7)$$

式中， ω_s 为与采样周期 T 相对应的采样频率，其关系为

$$\omega_s = 2\pi/T \quad (2-8)$$

所以，单位脉冲序列 $\delta_T(t)$ 的傅氏级数为

$$\delta_T(t) = \frac{1}{T} \sum_{n=-\infty}^{\infty} e^{-jn\omega_s t} \quad (2-9)$$

将式 (2-9) 代入采样信号的表达式 (2-5) 有

$$x^*(t) = \frac{1}{T} \sum_{k=-\infty}^{\infty} x(kT) e^{-jn\omega_s t} \quad (2-10)$$

基于拉氏变换的衰减定理有：如果 $x(t)$ ，其拉氏变换为 $X(s)$ ，则 $x(t)e^{-jn\omega_s t}$ ，其拉氏变换为 $X(s + jn\omega_s)$ 。

采样信号的拉氏变换为

$$X^*(s) = L[x^*(t)] = \frac{1}{T} \sum_{n=-\infty}^{\infty} X(s + jn\omega_s) \quad (2-11)$$

将 $s = j\omega$ 代入式 (2-11) 得到采样信号的傅氏变换为

$$X^*(\omega) = F[x^*(t)] = \frac{1}{T} \sum_{n=-\infty}^{\infty} X(j\omega + jn\omega_s) \quad (2-12)$$

式中， $X^*(\omega)$ 为 ω 的周期函数。

当 $n=0$ ，式 (2-12) 为主频谱分量，即

$$\frac{1}{T} X(j\omega)$$

主频谱分量除了幅值相差一个常数之外，与连续时间信号 $x(t)$ 的傅氏变换相同。因此，其频

谱形状相同，上限频率也是 ω_a 。

当 $n \neq 0$ 时，各周期项为主频谱的镜像频谱，其频谱形状与主频谱的形状相同，但是作为 ω 的周期函数，从主频谱分量的中心频谱 $\omega = 0$ 出发，以 ω_s 的整数倍向频率两端做频移。

如果满足条件 $\omega_s \geq 2\omega_a$ ，镜像频谱与主频谱相互分离，可以采用一个低通滤波器，将采样信号频谱中的镜像频谱滤除，用来恢复原连续时间信号 $x(t)$ ，如图 2-6 所示。

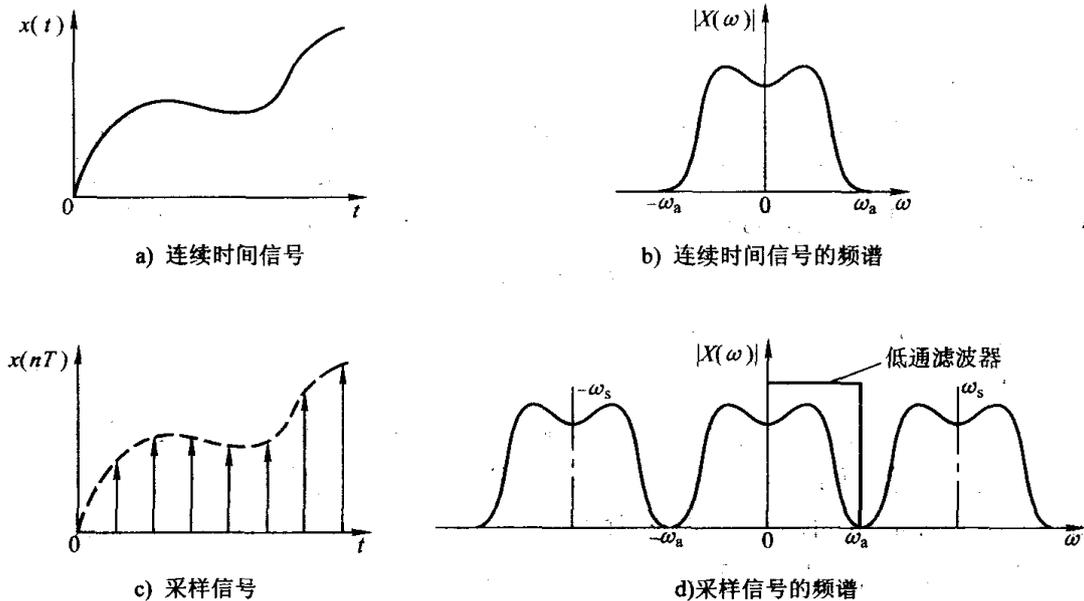


图 2-6 连续时间信号的频谱与采样信号的频谱

如果不满足条件 $\omega_s \geq 2\omega_a$ ，则采样信号频谱中的镜像频谱就会与主频谱混叠，采用低通滤波器的方法恢复的信号中仍混有镜像频谱成分，不能恢复为原连续时间信号 $x(t)$ ，所发生的信号混叠如图 2-7 所示。

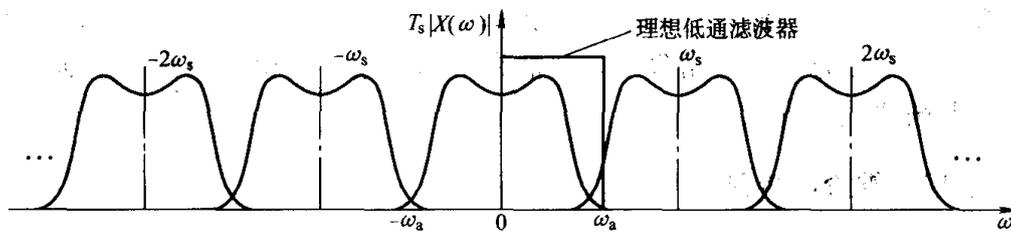


图 2-7 频谱混叠的说明

2.1.4 过程计算机控制系统中采样周期的选择

合理地选择采样周期 T ，是过程计算机控制系统设计的关键问题之一。由采样定理可知，当采样频率 $f_s \geq 2f_{\max}$ 时，系统可由离散的采样信号真实地恢复到原来的连续信号，这应当是选择采样周期 T 的理论基础和最低要求。从理论上讲，采样频率越高，失真越小。

从控制性能来考虑，采样周期越短越好。但是，考虑到计算机软件 and 硬件的制约，即硬件成本和控制软件的负荷，采样周期又不能太短。当然，随着计算机技术的飞速发展，这方面的制约将越来越小。

如果从过程控制方面的要求考虑，影响选择采样周期 T 的因素主要包括以下几方面。

1. 对象的动态特性

这主要与被控对象的惯性时间常数 T_p 和纯滞后时间 τ 有关。在不考虑计算机的制约因素条件下, 对具有自平衡能力的被控对象, 如果用 $G_p(s) = K_p e^{-\tau} / (1 + T_p s)$ 来近似描述对象的特性, 则采样周期的上限选择的经验公式为

$$T \approx \left(\frac{1}{5} \sim \frac{1}{15} \right) (T_p + \tau) \quad (2-13)$$

2. 扰动的特性

在控制系统中, 施加到系统的扰动一般包括两大类。一类是需要控制系统克服的频率较低的主要扰动; 另一类是频率较高的随机高频干扰, 如测量噪声等, 采样时要忽略。这样, 采样频率应选择在这两类干扰的频率之间。一般地, 采样频率的下限应满足

$$f_s \approx (5 \sim 10) f_b \quad (2-14)$$

式中, f_b 为需要克服的主要扰动的频率。

3. 控制算法

不同的控制算法对采样周期有不同的影响。例如, 采用连续化方法设计数字控制器时, 若忽略零阶保持器, 就要求系统具有足够高的采样频率, 以使采样控制系统更接近于连续系统。又如, 考虑到控制量的幅度都是受限的, 对某些直接离散化设计方法, 如最小方差控制、最少拍控制等算法, 采样周期又不能太小, 否则控制量容易超限。

4. 执行机构的速度

考虑到执行机构的响应速度都是有限的, 过高的采样频率对控制来说不仅无意义, 有时还会起到不好的作用。

5. 跟踪性能的要求

要求输出对参考输入具有很好跟踪响应性能的随动系统, 从采样原理出发, 就是要求输出能复现参考输入信号。因此, 这时的采样频率一般应大于 5~10 倍的参考输入信号频率。

由于生产过程千变万化, 非常复杂, 上面介绍的仅是一些粗略的设计原则, 实际的采样周期需要经过现场调试后确定。

2.1.5 信号复现与零阶保持器

从采样信号中恢复出连续时间信号称为信号的复现。采样定理从理论上指明了从采样信号 $x^*(t)$ 中恢复原连续时间信号 $x(t)$ 的条件可见, 信号的复现需要通过一个理想的低通滤波器才可以实现。同时, 对于频谱丰富的时间信号, 频谱成分的上限频率是不存在的。工程上将采样信号恢复为连续信号的装置称为保持器, 保持器是可以起到近似低通滤波器作用的工程器件。信号恢复与保持的实现依据的是信号的定值外推理论。

计算机作为信息处理装置, 输出结果需要把数字信号转换成模拟信号, 这中间要经过 D/A 转换器、采样开关和保持器才能完成。

保持器的作用主要有两个方面: 一是由于采样信号仅在采样开关闭合时有输出, 而在其余的时间输出为零, 所以在两次采样开关闭合期间, 存在一个采样信号如何进行保持的问题, 从数学的角度来说, 就是要解决在两个采样点之间进行插值的问题。二是保持器还要对采样时刻所产生的高频分量进行部分滤波。考虑到在硬件制作方面容易实现, 工程上大都采用零

阶保持器。

零阶保持器的特性为

$$f_h(kT+t') = f(kT) \quad (2-15)$$

式中, t' 为两个采样点之间的时间。式 (2-15) 说明, 零阶保持器输出值仅在每个采样时刻发生变化, 而在两个采样点之间的数值是不变的, 其对应工作波形如图 2-8 所示。

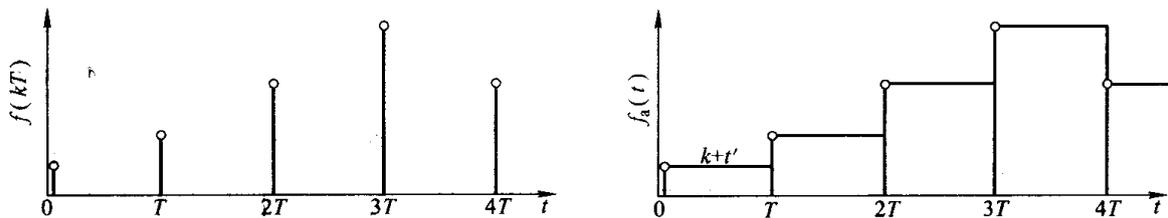


图 2-8 零阶保持器输入、输出波形

如果给零阶保持器输入一个单位脉冲, 由其输入、输出特性可知, 其输出为一个宽度为 T 、幅值为 1 的矩形波, 可以看作两个单位阶跃函数的叠加, 如图 2-9 所示, 即

$$g_h(t) = 1(t) - 1(t-T) \quad (2-16)$$

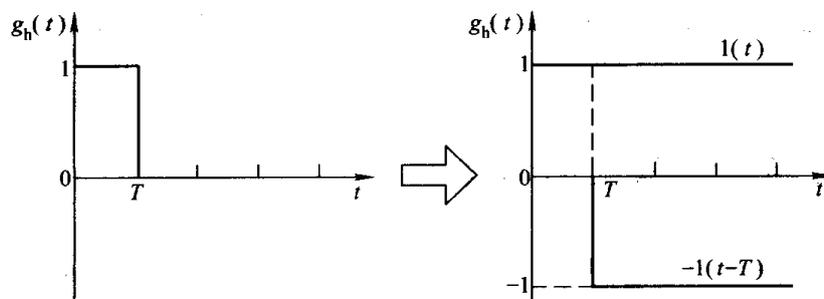


图 2-9 零阶保持器的传递关系

根据传递函数的其中一种定义方法, 即传递函数等于单位脉冲响应的拉氏变换, 将式 (2-16) 进行拉氏变换即可求得零阶保持器的传递函数

$$G_h(s) = L[1(t) - 1(t-T)] = \frac{1}{s} - \frac{1}{s} e^{-Ts} = \frac{1 - e^{-Ts}}{s} \quad (2-17)$$

用 $j\omega$ 代替式 (2-17) 中的 s , 可得零阶保持器的幅频、相频特性分别为

$$|G_h(j\omega)| = \frac{|1 - e^{-j\omega T}|}{|j\omega|} = T \frac{\sin \frac{\omega T}{2}}{\frac{\omega T}{2}} \quad (2-18)$$

$$\angle G_h(j\omega) = -\arctan \frac{1 - \cos \omega T}{\sin \omega T} = -\frac{\omega T}{2} \quad (2-19)$$

根据式 (2-18) 和式 (2-19) 画出的幅频、相频特性如图 2-10 所示。由图 2-10 可以看出, 零阶保持器具有明显的低通滤波特性, 但不是理想的低通滤波器。因此, 用于将数字信号恢复到原连续信号是有差别的, 但采样周期 T 取值越小, 差别也就越小。另外, 零阶保持器的滞后特性在离散系统中将引起附加的相位移, 使闭环系统的稳定性有所降低。在设计微机控

制系统的校正装置时，需要考虑这种滞后影响。

零阶保持器可以将当前采样点的幅值保持至下一个采样点时刻，从而使得两个采样点之间不为零值。采样信号经零阶保持器后，成为阶梯波形信号，如图 2-11 所示。如果取两个采样点的中点并使之平滑，平滑后的信号与原连续时间信号相比有 $1/2$ 个采样间隔的滞后。

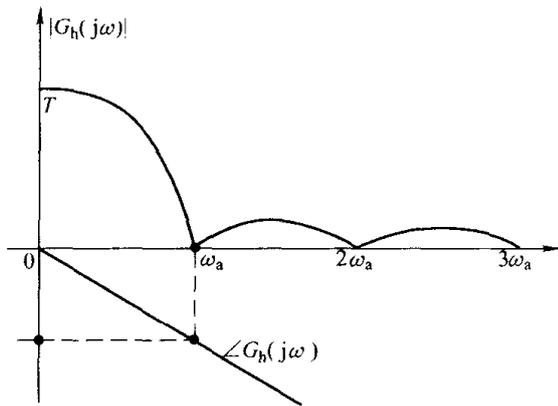


图 2-10 零阶保持器的幅频与相频特性

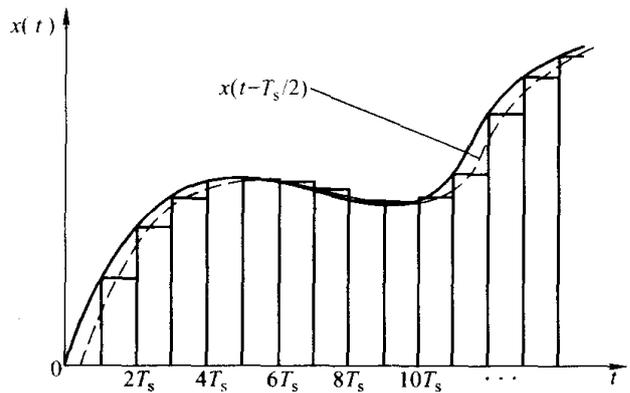


图 2-11 信号的零阶保持

最后需要指出，在相位上产生滞后现象是各阶保持器所具有的共性，与一阶和高阶保持器相比，零阶保持器具有最小的相位滞后。这也是在闭环离散控制系统中，大多采用零阶保持器的原因之一。

2.2 连续模型与离散模型间的转换

控制系统的数学模型可以分为连续模型和离散模型。连续模型可以用状态空间表达式、传递函数或传递矩阵来表示。与此相仿，离散模型也可以用离散状态空间表达式、 z 传递函数或 z 传递矩阵来表示。

在计算机控制系统中经常需要进行各种模型之间的相互转换。例如，计算机控制系统中经常包含连续环节和离散环节，为了分析方便，需要将连续模型转换为离散模型；又如，控制系统中对象的辨识往往得到的是离散模型，为了取得其连续模型，需要进行离散到连续模型转换。

一般地，连续模型与离散模型间的转换不能直接采用 z 变换或 z 反变换的方法求取。这是因为连续模型的输入信号是时间函数的连续信号，不是 z 变换中所要求的脉冲串信号。因此，将连续模型转换为离散模型采用 z 变换时，必须对两次采样间隔的输入信号进行某种形式的外推。这样获得的离散模型只能近似地等于连续模型，有时又称为离散等效模型，近似的程度取决于采样周期的大小；采样周期越小，近似程度越高。

2.2.1 传递函数与 z 传递函数间的转换

传递函数与 z 传递函数间的相互转换，也可以看作是模拟滤波器与数字滤波器间的转换，通常有向后差分变换法、双线性变换法、零阶保持器法和零极点匹配法等。

1. 向后差分变换法

从本质上看，传递函数 $G(s)$ 是微分方程的一种表达， z 传递函数 $G(z)$ 是差分方程的一种

表达, 这样就可以用数值微积分的方法将微分方程变换为差分方程, 其中最简单的是向后差分变换法。

向后差分变换中 s 与 z 之间的关系为

$$s \approx (1 - z^{-1})/T \quad (2-20)$$

式中, T 为采样周期, 这样向后差分变换法可以简单地表达为

$$G(z) \approx G(s) \Big|_{s=(1-z^{-1})/T} \quad (2-21)$$

例 2-1 设模拟滤波器 $G(s) = a/(s+a)$, 试用向后差分变换法, 设计等效的数字滤波器 $G(z)$ 。

$$\text{解: } G(z) = \frac{Y(z)}{E(z)} \approx G(s) \Big|_{s=\frac{1-z^{-1}}{T}} = \frac{a}{\frac{1-z^{-1}}{T} + a} = \frac{aT}{1+aT-z^{-1}}$$

由 $G(z)$ 可得到差分方程为

$$y(kT) \approx \frac{1}{1+aT} y(kT-T) + \frac{aT}{1+aT} e(kT)$$

无论原传递函数有多么复杂, 向后差分变换法的使用都非常简便, 无需对传递函数进行因式分解, 无论是手工计算还是计算机编程都很容易。

同时, 若 $G(s)$ 稳定, 向后差分变换后 $G(z)$ 也稳定; 但是, 如进行反方向的变换, 由 $G(z)$ 到 $G(s)$ 则不一定, 即向后差分的逆变换不能保持原函数的稳定特性。因此, 向后差分变换法一般只用正方向的变换。

需要注意的是, 这种变换的近似程度相对比较差, 特别是采样周期较大时。一般地, 向后差分变换法主要适用于精确度要求不高、传递函数较简单的场合, 如惯性滤波器、PID 控制器的离散化等。

2. 双线性变换法

根据 z 变换的定义, 拉氏变换算子 s 与 z 变换算子 z 之间有如下关系, 即

$$z = e^{sT} = \frac{e^{sT/2}}{e^{-sT/2}} \quad (2-22)$$

由泰勒级数可知

$$e^{sT/2} \approx 1 + sT/2 \quad (2-23)$$

$$e^{-sT/2} \approx 1 - sT/2 \quad (2-24)$$

可得

$$z \approx \frac{1+sT/2}{1-sT/2} \quad (2-25)$$

或

$$s \approx \frac{2}{T} \cdot \frac{1-z^{-1}}{1+z^{-1}} \quad (2-26)$$

根据式 (2-25) 和式 (2-26) 的转换关系, 就可以实现 $G(s)$ 和 $G(z)$ 之间的转换了, 简单地可用如下两个公式表达, 即

$$G(z) \approx G(s) \Big|_{s = \frac{2}{T} \frac{1-z^{-1}}{1+z^{-1}}} \quad (2-27)$$

$$G(s) \approx G(z) \Big|_{z = \frac{1+Ts/2}{1-Ts/2}} \quad (2-28)$$

例 2-2 设已知 $G(s) = a/(s+a)$ ，试用双线性变换法求 $G(z)$ 。

$$\begin{aligned} \text{解: } G(z) &\approx G(s) \Big|_{s = \frac{2}{T} \frac{1-z^{-1}}{1+z^{-1}}} \\ &= \frac{a}{\frac{2}{T} \frac{1-z^{-1}}{1+z^{-1}} + a} \\ &= \frac{a(1+z^{-1})}{(2/T+a) \left(1 + \frac{a-2/T}{a+2/T} z^{-1} \right)} \end{aligned}$$

例 2-3 设已知 $G(z) = \frac{1+z^{-1}+z^{-2}}{1+2z^{-1}+3z^{-2}}$ ，试用双线性变换法求 $G(s)$ 。

$$\begin{aligned} \text{解: } G(s) &\approx G(z) \Big|_{z^{-1} = \frac{1-Ts/2}{1+Ts/2}} \\ &= \frac{12+T^2s^2}{24-8Ts+2T^2s^2} \end{aligned}$$

与向后差分变换法一样，双线性变换法使用也非常简便，无需对传递函数进行因式分解，无论是手工计算还是计算机编程都很容易实现。

双线性变换将 s 平面的左半平面（ s 平面的稳定区域）映射到 z 平面的单位圆内（ z 平面的稳定区域）；因此，双线性变换不改变系统的稳定区域，即变换前后 $G(s)$ 与 $G(z)$ 的稳定特性不变，这是一种很有用的特性。有时为了判断离散系统的稳定性，可以利用双线性变换的逆变换，首先将 $G(z)$ 变换为 $G(s)$ ，然后就可以用连续系统的判别系统稳定性的各种方法了。

双线性变换实际上是基于用梯形法则对微分方程进行差分近似的方法，有时也称为梯形积分法或图斯汀（Tustin）变换法。当采样周期不大时，双线性变换具有很好的近似程度。

双线性变换法除了在计算机控制系统的设计中有着广泛的应用外，还可用于快速数字仿真及数字滤波器设计等其他方面。

3. 零阶保持器法

在计算机控制系统中常需要将一个连续的被控对象离散化，相当于在系统中一个连续的传递函数 $G(s)$ 前加有零阶保持器，然后将其离散化成 $G(z)$ ，如图 2-12 所示。

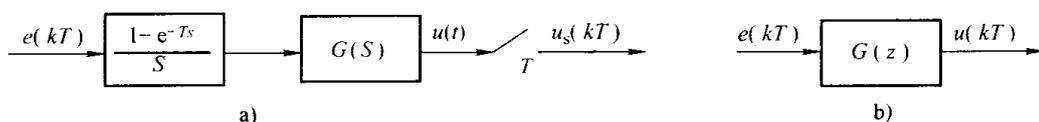


图 2-12 零阶保持器法的模型转换

对于带有零阶保持器的 $G(s)$, 其输出量 $u(t)$ 的采样值 $u_s(kT)$ 应该与离散化以后的输出响应 $u(kT)$ 相等, 即 $u(kT) = u_s(kT)$, 于是

$$\begin{aligned} G(z) &= \mathcal{Z} \left[\frac{1 - e^{-sT}}{s} G(s) \right] \\ &= (1 - z^{-1}) \mathcal{Z} \left[\frac{G(s)}{s} \right] \end{aligned} \quad (2-29)$$

例 2-4 已知 $G(s) = \frac{1}{s(s+1)}$, 试用零阶保持器法求 $G(z)$ 。

$$\begin{aligned} \text{解: } G(z) &= (1 - z^{-1}) \mathcal{Z} \left[\frac{G(s)}{s} \right] \\ &= (1 - z^{-1}) \mathcal{Z} \left[\frac{1}{s^2(s+1)} \right] \\ &= (1 - z^{-1}) \mathcal{Z} \left[\frac{1}{s^2} - \frac{1}{s} + \frac{1}{s+1} \right] \end{aligned}$$

查 z 变换表可得

$$\begin{aligned} G(z) &= \frac{T}{z-1} - 1 + \frac{z-1}{z-e^{-T}} \\ &= \frac{(T-1+e^{-T})z - (T+1)e^{-T} + 1}{z^2 - (1+e^{-T})z + e^{-T}} \end{aligned}$$

采用零阶保持器法进行模型转换时, 常需要对连续传递函数做因式分解, 以便查 z 变换表得到变换结果, 该方法仅适用于连续传递函数比较简单、可以手工计算的情况。

对于前面带有零阶保持器的连续传递函数的离散化采用上述方法变换时, 离散后响应序列与原连续传递函数的响应在采样时刻的值是一样的, 从这个意义上讲, 这种变换是一种精确的变换。

4. 零、极点匹配法

在传递函数 $G(s)$ 的离散近似中, 若直接利用映射关系将 $G(s)$ 的零、极点全部对应映射到 z 平面中去, 就可直接得到 $G(z)$ 。

设 $G(s)$ 的零、极点已知

$$G(s) = \frac{K_s (s+z_1)(s+z_2)\cdots(s+z_m)}{(s+p_1)(s+p_2)\cdots(s+p_n)} \quad (2-30)$$

根据定义 $z = e^{Ts}$, 可以直接将 S 平面的零点 ($s = -z_i$) 和极点 ($s = -p_i$) 一一对应地映射为 Z 平面上的零点 ($z = e^{-z_i T}$) 和极点 ($z = e^{-p_i T}$), 其中 T 为采样周期。

通常, 式 (2-30) 中 $n > m$ (即极点个数大于零点个数), 这时相当于在 S 平面的无穷远处, 即 $s = j\omega$ 且 ω 趋于无穷大, 存在 $n - m$ 个零点。由于 $z = e^{j\omega T}$, z 是 ω 的周期函数, 若系统工作在主频区, 即 $-\pi/T \leq \omega \leq \pi/T$, 所以 ω 趋于无穷大, 就可看作 ω 趋于 π/T , 相当于 z 趋于 -1 。因此 S 平面上的无穷远零点, 可用 Z 平面上的 $z = -1$ 来匹配。

因此,用零点、极点匹配法变换式(2-30),可得到

$$G(z) = \frac{K_z(z - e^{-z_1 T})(z - e^{-z_2 T}) \cdots (z - e^{-z_m T})(z + 1)^{(n-m)}}{(z - e^{-p_1 T})(z - e^{-p_2 T}) \cdots (z - e^{-p_n T})} \quad (2-31)$$

若在离散传递函数要求有一拍迟延,则式(2-31)分子中最后一项 $(z + 1)^{n-m}$ 应改为 $(z + 1)^{n-m-1}$ 。

z 传递函数中的增益 K_z 可以用某个特征频率(如 $\omega = 0$) 处,使用变换前后两者的增益匹配的方法来确定。如取稳态增益相等,则

$$G(s)|_{s=0} = G(z)|_{z=1} \quad (2-32)$$

对于复数零点、极点 $(s + a \pm jb)$,有时直接变换成如下形式,使用起来可以更方便些。

$$(s + a)^2 + b^2 \rightarrow z^2 - 2ze^{-aT} \cos bT + e^{-2aT} \quad (2-33)$$

例 2-5 已知 $G(s) = \frac{s+1}{s^2 + 0.2s + 1}$, 试用零点、极点匹配法求 $G(z)$ ($T=1s$)。

解: 对 $G(s)$ 的零点、极点做等效变换

$$\text{零点} \quad s + 1 \rightarrow z - e^{-T} = z - 0.368$$

$$\begin{aligned} \text{极点} \quad s^2 + 0.2s + 1 &= (s + 0.1)^2 + 0.995^2 \\ &\rightarrow z^2 - 2ze^{-0.1 \times 1} \cos 0.995 + e^{-2 \times 0.1} = z^2 - 0.985z + 0.819 \end{aligned}$$

补充零点 $z + 1$

根据零点、极点匹配法的变换公式,得

$$G(z) = \frac{K_z(z - 0.368)(z + 1)}{z^2 - 0.985z + 0.819}$$

式中, K_z 为待定的增益,由 $G(s)|_{s=0} = G(z)|_{z=1}$ 知

$$1 = \frac{K_z \times 0.632 \times 2}{1 - 0.985 + 0.819}$$

解得 $K_z = 0.660$, 代入上式得

$$G(z) = \frac{0.660(z - 0.368)(z + 1)}{z^2 - 0.985z + 0.819}$$

零、极点匹配法由于变换前后零、极点在 S 平面中与在 Z 平面中一一对应,因此,与双线性变换法一样,不改变系统的稳定区域,即变换前后 $G(s)$ 与 $G(z)$ 的稳定特性不变。同时,该方法还可以保证变换前后特征频率处的增益不变。

但是,采用零、极点匹配法时,要先求出连续传递函数的全部零点、极点,计算也稍显复杂,因此该方法主要适合于连续传递函数比较简单、便于进行手工计算的情况。

2.2.2 连续与离散状态方程之间的相互转换

连续与离散状态方程之间的相互转换,最常用的主要有零阶保持器法等,为了简单,这

里仅给出采用零阶保持器法时的变换公式，具体的推导过程读者可以查阅有关的参考文献。

1. 连续状态方程转换为离散状态方程

设过程的连续状态方程为

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) \end{cases} \quad (2-34)$$

式中， A 为状态矩阵； B 为控制矩阵； C 为输出矩阵。同时，该连续环节前有一个零阶保持器。

当采样周期为 T 时，按零阶保持器法可以得到其相对应的离散状态方程

$$\begin{cases} x(kT + T) = Fx(kT) + Gu(kT) \\ y(kT) = Cx(kT) \end{cases} \quad (2-35)$$

其中

$$F = e^{AT}$$

$$G = \left[\int_0^T e^{At} dt \right] B$$

这样，转换的问题可归结为计算矩阵指数 F 及其积分 G 。

例 2-6 设连续环节的状态方程为

$$\begin{cases} \begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u(t) \\ y(t) = x_2(t) \end{cases}$$

试求连续环节的离散状态空间表达式。

解：由连续状态方程得

$$A = \begin{bmatrix} -1 & 0 \\ 1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$\text{则有 } F = e^{AT} = \mathcal{L}^{-1} \left[(sI - A)^{-1} \right]_{t=T} = \begin{bmatrix} e^{-T} & 0 \\ 1 - e^{-T} & T \end{bmatrix}$$

$$\begin{aligned} G &= \left[\int_0^T e^{At} dt \right] B \\ &= \int_0^T \begin{bmatrix} e^{-t} & 0 \\ 1 - e^{-t} & t \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} dt \\ &= \int_0^T \begin{bmatrix} e^{-t} \\ 1 - e^{-t} \end{bmatrix} dt \\ &= \begin{bmatrix} 1 - e^{-T} \\ T - 1 + e^{-T} \end{bmatrix} \end{aligned}$$

于是可得离散方程为

$$\begin{cases} \begin{bmatrix} x_1(kT+T) \\ x_2(kT+T) \end{bmatrix} = \begin{bmatrix} e^{-T} & 0 \\ 1-e^{-T} & T \end{bmatrix} \begin{bmatrix} x_1(kT) \\ x_2(kT) \end{bmatrix} + \begin{bmatrix} 1-e^{-T} \\ T-1+e^{-T} \end{bmatrix} u(kT) \\ y(kT) = x_2(kT) \end{cases}$$

2. 离散状态方程转换为连续状态方程

设离散模型的状态方程为

$$\begin{cases} x(kT+T) = Fx(kT) + Gu(kT) \\ y(kT) = Cx(kT) \end{cases}$$

式中, T 为采样周期。

若连续模型前带有零阶保持器, 则与该离散模型相对应的连续状态方程为

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) \end{cases}$$

其中

$$A = \frac{1}{T} \ln F$$

$$B = (F - I)^{-1} AG$$

从上面的公式可见, 连续与离散状态方程之间的变换, 需要进行复杂的矩阵运算, 包括矩阵的指数、对数、逆和积分等计算。若用手工计算, 计算工作量很大, 而且容易出错, 精确度也十分有限; 一般采用计算机编制相应的计算程序来解决上述问题, 具体的编程算法读者可以查阅有关的参考文献。

2.3 采样控制系统的稳态及动态分析

2.3.1 采样控制系统的稳态分析

考虑如图 2-13 所示的单位反馈系统, 其中 $G(s)$ 为广义对象。下面讨论在典型输入信号下系统的稳态误差, 首先求出误差的采样信号 $e^*(t)$ 的 z 变换 $E(z)$ 与输入采样信号 $r^*(t)$ 的 z 变换 $R(z)$ 之间的关系。

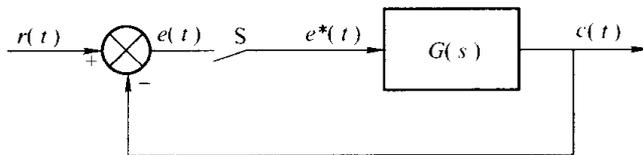


图 2-13 单位反馈系统

由
$$e(t) = r(t) - c(t) \quad (2-36)$$

即
$$e^*(t) = r^*(t) - c^*(t) \quad (2-37)$$

或
$$E(z) = R(z) - C(z) \quad (2-38)$$

以及
$$C(z) = G(z)E(z) \quad (2-39)$$

可得

$$E(z) = \frac{R(z)}{1+G(z)} \quad (2-40)$$

利用 z 变换的终值定理, 可求得系统稳态误差为

$$e(\infty) = \lim_{k \rightarrow \infty} e(k) = \lim_{z \rightarrow 1} \left(\frac{z-1}{z} \right) \frac{R(z)}{1+G(z)} \quad (2-41)$$

根据 $G(z)$ 中包含有 $z=1$ 的极点个数, 可以将系统分成 0 型、1 型、2 型等。

1. 单位阶跃输入

$$R(z) = \frac{z}{z-1} \quad (2-42)$$

$$e(\infty) = \lim_{z \rightarrow 1} \left[\frac{z-1}{z} \cdot \frac{1}{1+G(z)} \cdot \frac{z}{z-1} \right] = \frac{1}{1+G(1)} = \frac{1}{K_p} \quad (2-43)$$

式中, K_p 为位置误差系数; $K_p = 1+G(1)$ 。

对于 0 型系统, K_p 为有限值, 故有 $e(\infty) = 1/K_p$; 对于 1 型或高于 1 型的系统, $G(1) = \infty$; 所以有 $e(\infty) = 0$, 系统无稳态误差。

2. 单位斜坡输入

$$R(z) = \frac{Tz}{(z-1)^2} \quad (2-44)$$

$$e(\infty) = \lim_{z \rightarrow 1} \left[\frac{z-1}{z} \cdot \frac{1}{1+G(z)} \cdot \frac{Tz}{(z-1)^2} \right] = T \lim_{z \rightarrow 1} \left[\frac{1}{(z-1)} \cdot \frac{1}{G(z)} \right] = \frac{T}{K_v} \quad (2-45)$$

式中, $K_v = \lim_{z \rightarrow 1} (z-1)G(z)$ 为速度误差系数。

对于 0 型系统, $K_v = 0$, 故有 $e(\infty) \rightarrow \infty$; 对于 1 型系统, K_v 为有限值, $e(\infty) = T/K_v$; 对于 2 型及其以上的系统, $K_v \rightarrow \infty$, $e(\infty) = 0$ 。

综上所述, 采样控制系统的稳态误差与广义对象所对应的脉冲传递函数 $G(z)$ 中所含 $z=1$ 的极点个数密切相关, 在非阶跃输入时还和采样周期有关。

2.3.2 采样控制系统的动态分析

采样控制系统的脉冲传递函数可以写成:

$$G(z) = \frac{K(z-z_1)(z-z_2)\cdots(z-z_m)}{(z-p_1)(z-p_2)\cdots(z-p_n)} \quad (2-46)$$

式中, z_1, z_2, \dots, z_m 称为系统的零点; p_1, p_2, \dots, p_n 称为系统的极点。利用部分分式法, 可将 $G(z)$ 展开成

$$G(z) = \frac{A_1}{1-p_1z^{-1}} + \frac{A_2}{1-p_2z^{-1}} + \cdots + \frac{A_n}{1-p_nz^{-1}} \quad (2-47)$$

由此可见, 采样系统的时间响应是它各个极点时间响应的线性叠加。如果了解位于任意位置的一个极点所对应的时间响应, 则整个系统的时间响应也容易获得。

与连续系统类似, 采样系统的零点和极点在 Z 平面上的分布对系统的瞬态响应起着决定性的作用。特别是系统的极点不但决定了系统的稳定性, 还决定了系统响应速度。下面考虑只有一个实极点或只有一对共轭复数极点的脉冲传递函数, 在极点位置不同时它们的单位脉冲响应。

显然, 对于单位脉冲序列 $\delta(k)$, 它的 z 变换 $Z[\delta(k)] = 1$ 。在单位脉冲序列的作用下系统的动态过程称为系统的单位脉冲响应。设系统输入为 $R(z)$, 输出为 $C(z)$, 系统脉冲传递函数为 $G(z)$ 。由于在单位脉冲作为输入时, 有 $R(z) = 1$ 。这时系统输出为

$$C(z) = G(z)R(z) = G(z) \quad (2-48)$$

因此, 若设系统单位脉冲响应序列为 $h(k)$, 则有

$$h(k) = Z^{-1}[G(z)] \quad (2-49)$$

即系统脉冲传递函数 $G(z)$ 的 z 反变换即为系统的单位脉冲响应函数。

1. 实轴上单极点所对应的脉冲响应

设系统有一个位于 p_i 的单极点, 则在式 (2-47) 所示的系统脉冲传递函数的部分分式中必存在 $A_i/(1 - p_i z^{-1})$ 的一项, 在单位脉冲作用下, 对应于这一项的输出序列为 $h(k) = A_i p_i^k$ 。当 p_i 位于 Z 平面不同位置时, 它所对应的脉冲响应序列如图 2-14 所示。

- 1) $p_i > 1$ 时, $h(k)$ 为发散序列;
- 2) $p_i = 1$ 时, $h(k)$ 为等幅脉冲序列;
- 3) $0 < p_i < 1$ 时, $h(k)$ 为单调衰减脉冲序列, 且 p_i 越接近 0, 衰减越快;
- 4) $-1 < p_i < 0$ 时, $h(k)$ 为交替变号的衰减脉冲序列;
- 5) $p_i = -1$ 时, $h(k)$ 为交替变号的等幅脉冲序列;
- 6) $p_i < -1$ 时, $h(k)$ 是交替变号的发散脉冲序列。

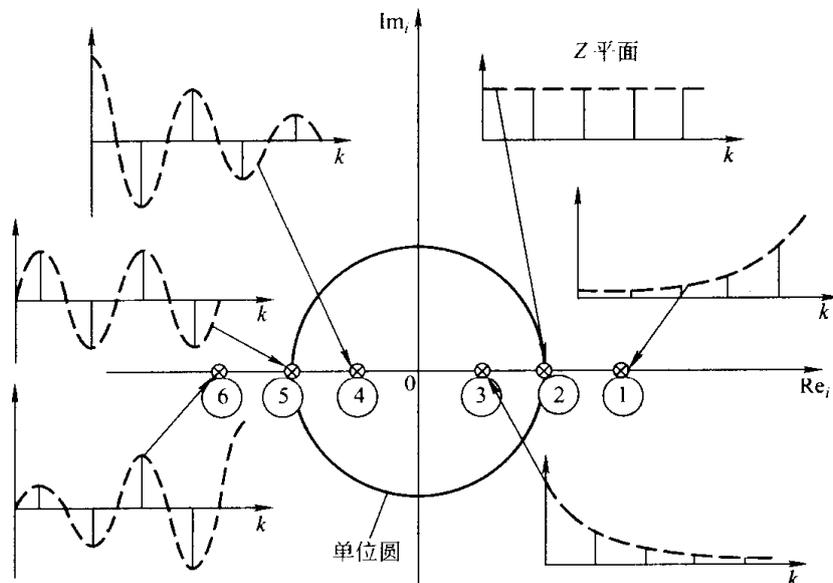


图 2-14 不同位置的实数极点与脉冲响应的关系

2. 一对共轭复数极点对应的脉冲响应

设系统有一对位于 $p_i, p_{i+1} = a + jb$ 的共轭复数极点, 则系统脉冲传递函数的部分分式

中必然有一项

$$\frac{A_i}{1-p_i z^{-1}} + \frac{\bar{A}_i}{1-\bar{p}_i z^{-1}} = \frac{(B_i + C_i z^{-1})}{1-2az^{-1} + (a^2 + b^2)z^{-2}} \quad (2-50)$$

式中, B_i 和 C_i 是由 A_i 、 \bar{A}_i 和 a 、 b 等计算得到的系数。式 (2-50) 的 z 反变换为

$$h(k) = ar^k \sin(k\theta + \varphi) \quad (2-51)$$

式中, $r = \sqrt{a^2 + b^2}$, $\theta = \arctan \frac{b}{a}$ 。当 a 、 b 位于 Z 平面的不同位置时所对应的单位脉冲响应由图 2-15 给出。

令 $r = \sqrt{a^2 + b^2}$ 为复数极点的模, 它表示极点到原点之间的距离, 如图 2-15 所示。

- 1) 当 $r > 1$ 时, $h(k)$ 为发散振荡序列;
- 2) 当 $r = 1$ 时, $h(k)$ 为等幅振荡序列;
- 3) 当 $r < 1$ 时, $h(k)$ 为衰减振荡序列, 且 r 越小, 衰减越快。

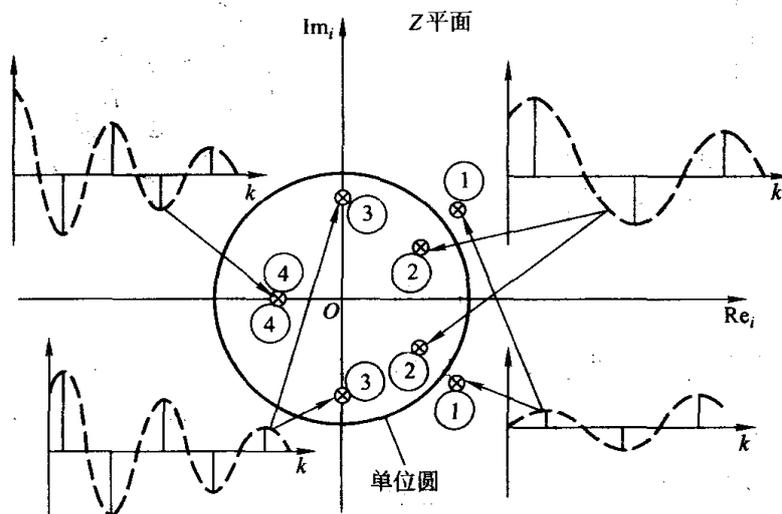


图 2-15 不同位置的共轭复数极点与脉冲响应的关系

虽然以上只分析了单个实极点和共轭复数极点的情况, 由式 (2-47) 可知, 整个系统的单位脉冲响应实际是这些项的脉冲响应之和; 因此, 采样系统脉冲传递函数的极点在 Z 平面上的位置, 决定了系统动态响应的速度。其中极点的模, 即极点与原点的距离, 决定了系统脉冲响应序列是发散还是衰减, 即决定了系统的稳定性。如果系统所有的极点的模都小于 1, 或者说系统所有的极点都位于 Z 平面上以原点为圆心、以 1 为半径的单位圆内, 则式 (2-51) 中各项都对应着衰减的脉冲响应序列, 随着 $k \rightarrow \infty$, 各项都趋向于零。因此, 系统是渐近稳定的。反之, 若系统中有模大于 1 的极点, 则当 $k \rightarrow \infty$ 时, 即使式 (2-51) 中的其他项都趋向于零, 但是由于相应于模大于 1 的极点的项的时间响应趋向于无穷大, 造成系统单位脉冲响应也趋向于无穷大, 因此系统为不稳定系统。

2.4 线性离散控制系统的稳定性分析

在控制系统的分析与设计过程中, 稳定性分析总是最为重要的内容, 计算机控制系统也

不例外。计算机控制系统一般是离散/模拟混合系统，为了变换的方便，常需要变换到离散域去分析计算机控制系统的稳定性。为了简单起见，下面主要讨论线性离散系统的稳定性。

2.4.1 S平面与Z平面的映射关系

在定义 z 变换时，有 $z = e^{Ts}$ ，其中 s 、 z 均为复变量， T 为采样周期。

设 $s = \sigma + j\omega$ ，则 $z = e^{(\sigma + j\omega)T} = e^{\sigma T} e^{j\omega T}$ ，显然， $|z| = e^{\sigma T}$ ， $\angle z = \omega T$ 。

当 $\sigma = 0$ 时， $|z| = 1$ ，即 S 平面上的虚轴映射到 Z 平面上，是以原点为圆心的单位圆周；当 $\sigma < 0$ 时， $|z| < 1$ ，即 S 平面上的左半部分映射到 Z 平面上，是以原点为圆心的单位圆内部分；当 $\sigma > 0$ 时， $|z| > 1$ ，即 S 平面上的右半部分映射到 Z 平面上，是以原点为圆心的单位圆外部分。

S 平面与 Z 平面的映射关系如图 2-16 所示。

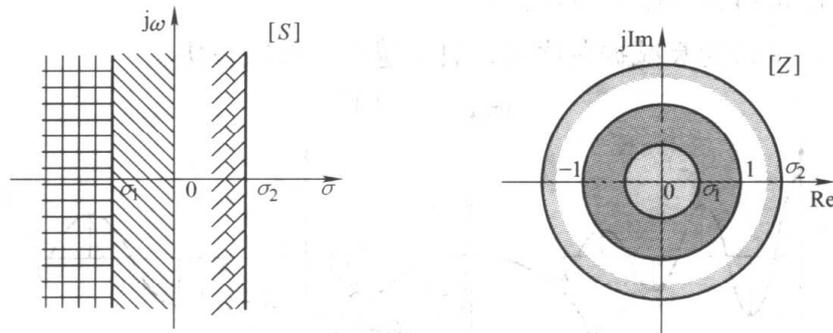


图 2-16 S 平面与 Z 平面的映射关系

设 S 平面上左半部分有直线 $\sigma_1 < 0$ ，映射到 Z 平面上，是以原点为圆心，以 $e^{\sigma_1 T}$ 为半径的圆周，显然 $|e^{\sigma_1 T}| < 1$ ；类似地，设 S 平面上右半部分有直线 $\sigma_2 > 0$ ，映射到 Z 平面上，是以原点为圆心，以 $e^{\sigma_2 T}$ 为半径的圆周，显然 $|e^{\sigma_2 T}| > 1$ 。

2.4.2 线性离散系统的稳定域

大家知道，一个线性连续系统稳定的充分必要条件是，其闭环特征方程 $G_c(s) = 0$ 的根全部位于 S 平面的左半部分。根据上面 S - Z 平面的映射关系易知，对线性离散系统的特征根来说，其稳定区域一定是在单位圆内。这一结论也可以直接通过分析特征方程所代表的线性齐次差分方程的解得到验证，此处就不再具体分析，读者可以查阅相关文献。

线性离散系统稳定的充分必要条件是特征方程的全部根或闭环 z 传递函数 $G_c(z)$ 的全部极点 z_i 都分布在 z 平面上以原点为圆心的单位圆内，即 $|z_i| < 1$ 。

与 S 平面的虚轴类似， Z 平面的单位圆周（即 $|z_i| = 1$ ）是稳定的临界线。需要特别说明的是，单位圆上 $z = 1$ 点是一个特殊的点，它对应于 S 平面的原点，可以看作是稳定的。

在线性离散控制系统的设计中，为使闭环系统具有满意的过渡过程，闭环零点应尽量避免分布在 Z 平面单位圆内的左半部分，尤其不要靠近负实轴。闭环极点最好分布在单位圆内的右半部分，理想的位置是在单位圆内正实轴并靠近原点，因为这时 $|z_i|$ 值很小，暂态分量衰减快，离散系统具有快速响应输入信号的能力。

例 2-7 设线性离散系统的特征方程为

$$(z^2 - 0.1z - 0.3)(z^2 - 0.1z - 0.56)(z - 0.9) = 0$$

试判断系统的稳定性。

解：由特征方程可得系统的特征根为

$$z_1 = -0.5, \quad z_2 = 0.6, \quad z_3 = -0.7, \quad z_4 = 0.8, \quad z_5 = 0.9$$

可知系统的全部特征根均位于单位圆内，所以系统是稳定的。

2.4.3 线性离散系统的稳定性判据

从前面的介绍可以看出，线性离散系统的稳定与否实质上是判断特征根或闭环极点的模的大小。当离散系统的阶次较低或已经分解为因式乘积形式时，可以直接通过求特征根来判别系统的稳定性。但是当系统的阶次较高时，就很难直接找出特征根。与连续系统可以应用代数判据不求特征根判别稳定性的方法类似，下面介绍几种常用的适合线性离散系统的代数稳定性判据。

1. ω 平面上的劳斯判据

线性连续系统可以通过劳斯判据来分析闭环极点在 S 平面右半部分的个数，来判别系统是否稳定。为了将连续系统的劳斯判据引入到离散系统中，需要对离散系统的特征方程进行一种称为 ω 变换的双线性坐标变换，在变换后的 ω 平面上可以使用劳斯判据。

如果通过

$$z = \frac{\omega + 1}{\omega - 1} \quad (2-52)$$

的变换，则可以将 Z 平面的单位圆 ($|z|=1$) 映射到 ω 平面的虚轴上，将单位圆内 ($|z|<1$) 的区域映射到 ω 平面的左半部分，这样就与连续系统 S 平面的情形类似了。根据 ω 变换后的特征方程

$$a_n \omega^n + a_{n-1} \omega^{n-1} + \dots + a_1 \omega + a_0 = 0 \quad (2-53)$$

应用劳斯判据就可判断离散系统的稳定性。

例 2-8 设某闭环离散系统的特征方程为

$$45z^3 - 117z^2 + 119z - 39 = 0$$

试判断该系统的稳定性。

解：将 $z = \frac{\omega + 1}{\omega - 1}$ 代入上式，得

$$45 \frac{(\omega + 1)^3}{(\omega - 1)^3} - 117 \frac{(\omega + 1)^2}{(\omega - 1)^2} + 119 \frac{(\omega + 1)}{(\omega - 1)} - 39 = 0$$

将上式整理并化简后，得到

$$\omega^3 + 2\omega^2 + 2\omega + 40 = 0$$

应用劳斯判据，列出劳斯阵列表

$$\begin{array}{cccc} \omega^3 & 1 & 2 & 0 \\ \omega^2 & 2 & 40 & 0 \\ \omega^1 & -18 & 0 & \\ \omega^0 & 40 & & \end{array}$$

由于劳斯阵列表中第一列系数有两次符号改变,说明有两个闭环特征根在 ω 平面的右半部分,也即系统有两个特征根在 Z 平面单位圆外,所以闭环离散系统不稳定。

例 2-9 已知闭环离散控制系统如图 2-17 所示。

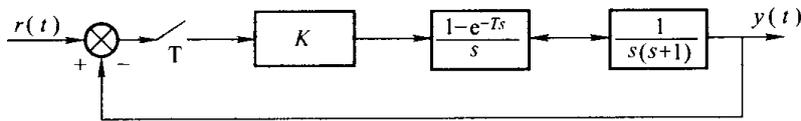


图 2-17 闭环离散控制系统框图

试判断采样周期 T 分别为 $1s$ 和 $5s$ 时,使闭环离散控制系统稳定的比例控制器 K 的取值范围。

解:由零阶保持器法可知,图中系统的开环 Z 传递函数为

$$\begin{aligned} G(z) &= (1-z^{-1})\mathcal{Z}\left[\frac{G(s)}{s}\right] \\ &= (1-z^{-1})\mathcal{Z}\left[\frac{K}{s^2(s+1)}\right] \\ &= K(1-z^{-1})\mathcal{Z}\left[\frac{1}{s^2}-\frac{1}{s}+\frac{1}{s+1}\right] \\ &= \frac{K[e^{-T}z+(1-e^{-T}-Te^{-T})]}{(z-1)(z-e^{-T})} \end{aligned}$$

对应闭环系统的特征方程为

$$1+G(z)=0$$

经整理并化简后,得

$$z^2 + [(K-1)e^{-T} + K(T-1) - 1]z + (1-K-KT)e^{-T} + K = 0$$

应用 ω 变换,将 $z = \frac{\omega+1}{\omega-1}$ 代入上述方程,得

$$(\omega+1)^2 + [(K-1)e^{-T} + K(T-1) - 1](\omega^2 - 1) + [(1-K-KT)e^{-T} + K](\omega-1)^2 = 0$$

经整理并化简得

$$KT(1-e^{-T})\omega^2 + [(2-2K) - 2(1-K-KT)e^{-T}]\omega + [(2+2K-KT) + (2-2K-KT)e^{-T}] = 0$$

这是一个二阶系统,应用劳斯判据可知,系统稳定的条件是各项系数同符号,即

$$\begin{cases} KT(1-e^{-T}) > 0 \\ (2-2K) - 2(1-K-KT)e^{-T} > 0 \\ (2+2K-KT) + (2-2K-KT)e^{-T} > 0 \end{cases}$$

当 $T=1s$ 时, 可解上述不等式得比例控制器 $0 < K < 2.4$ 时, 系统稳定。

当 $T=5s$ 时, 可解上述不等式得比例控制器 $0 < K < 0.652$ 时, 系统稳定。

从上述例题可见, 尽管连续的二阶系统只要 $K > 0$ 总是稳定的, 但用计算机实现时, 该离散二阶系统随着比例增益 K 的增大, 系统可能变得不稳定, 而且随着采样周期的增大, K 的稳定范围越来越小。一般地, 对一个离散采样控制系统来说, 随着采样周期的增大, 系统的稳定性一般会降低, 因此, 在可能的情况下, 应尽量采用小的采样周期, 这样更接近于连续系统的控制效果。

需要指出的是, 如果离散系统一旦得到了经过 ω 变换后的系统方程式, 那么凡适用于线性连续系统的稳定判据, 如赫尔维兹判据、奈奎斯特判据等, 都可以应用到线性离散控制系统中, 这里就不再一一介绍了。

2. 鞠利判据

设线性离散控制系统的特征方程按降幂方式表达

$$F(z) = a_n z^n + a_{n-1} z^{n-1} + \cdots + a_1 z + a_0 = 0 \quad (a_n > 0) \quad (2-54)$$

则可以直接在 Z 平面上, 借助如下的鞠利阵列来判断系统的稳定性。

	z^0	z^1	\cdots	z^k	\cdots	z^{n-2}	z^{n-1}	z^n
1	a_0	a_1	\cdots	a_k	\cdots	a_{n-2}	a_{n-1}	a_n
2	a_n	a_{n-1}	\cdots	a_{n-k}	\cdots	a_2	a_1	a_0
3	b_0	b_1	\cdots	b_k	\cdots	b_{n-2}	b_{n-1}	
4	b_{n-1}	b_{n-2}	\cdots	b_{n-k-1}	\cdots	b_1	b_0	
5	c_0	c_1	\cdots	c_k	\cdots	c_{n-2}		
6	c_{n-2}	c_{n-1}	\cdots	c_{n-k-2}	\cdots	c_0		
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots			
$2n-5$	p_0	p_1	p_2	p_3				
$2n-4$	p_3	p_2	p_1	p_0				
$2n-3$	q_0	q_1	q_2					

其中 $b_k = \begin{vmatrix} a_0 & a_{n-k} \\ a_n & a_k \end{vmatrix}$ $c_k = \begin{vmatrix} b_0 & b_{n-k-1} \\ b_{n-1} & b_k \end{vmatrix}$

$q_0 = \begin{vmatrix} p_0 & p_3 \\ p_3 & p_0 \end{vmatrix}$ $q_2 = \begin{vmatrix} p_0 & p_1 \\ p_3 & p_2 \end{vmatrix}$

上述鞠利阵列的偶数行是奇数行的逆序, 而奇数行各元素则是由上两行的第一列与本列的元素构成的行列式计算得到。

得到上述鞠利阵列后, 系统稳定的充要条件为满足如下 $n+1$ 个条件, 即

$$\begin{cases} F(1) > 0 \\ (-1)^n F(-1) > 0 \\ |a_0| < a_n \\ |b_0| > |b_{n-1}| \\ |c_0| > |c_{n-2}| \\ \vdots \\ |q_0| > |q_2| \end{cases} \quad (2-55)$$

对于二阶系统，上述条件可简化为

$$\begin{cases} F(1) > 0 \\ F(-1) > 0 \\ |a_0| < a_2 \end{cases} \quad (2-56)$$

例 2-10 设离散系统的特征方程为

$$F(z) = 3z^3 + 3z^2 + 2z + 1$$

试判别其稳定性。

解：由特征方程可构成如下鞠利阵列

z^0	z^1	z^2	z^3
1	2	3	3
3	3	2	1
-8	-7	-3	

可得

$$\begin{cases} F(1) = 9 > 0 \\ (-1)^3 F(-1) = 1 > 0 \\ |a_0| = 1 < a_3 = 3 \\ |b_0| = 8 > |b_2| = 3 \end{cases}$$

根据鞠利判据可知系统是稳定的。

3. 离散状态方程表达系统的稳定性分析

对于由如下状态方程表达的线性离散系统

$$x(kT + T) = Ax(kT) + Bu(kT) \quad (2-57)$$

一般可以通过其特征方程 $|zI - A| = 0$ ，如同前面介绍的方法一样分析系统的稳定性；还可以直接通过计算矩阵 A 的特征值 $|a_i| < 1$ 判别系统稳定与否；此外，也可以由李亚普诺夫方程进行直接判定。

对于上面的线性定常系统，若存在正定对称矩阵 Q 和 P ，并满足

$$A^T P A - P = -Q \quad (2-58)$$

则系统是稳定的。式(2-58)称为李亚普诺夫方程。

矩阵 P 为正定的充要条件是, 矩阵 P 的所有各阶主、子行列式大于零, 且对任意的向量 $X \neq 0$, 不等式 $X^T P X > 0$ 成立。

例 2-11 试判别下面系统的稳定性

$$X(k+1) = \begin{bmatrix} 0.8 & -0.4 \\ 1.2 & 0.2 \end{bmatrix} X(k)$$

解: 选 Q 为单位阵, 由李亚普诺夫方程得

$$\begin{bmatrix} 0.8 & 1.2 \\ -0.4 & 0.2 \end{bmatrix} \begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix} - \begin{bmatrix} 0.8 & -0.4 \\ 1.2 & 0.2 \end{bmatrix} \begin{bmatrix} p_{11} & p_{12} \\ p_{12} & p_{22} \end{bmatrix} = - \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

解上述方程, 得

$$P = \begin{bmatrix} 3.904 & -1.057 \\ -1.057 & 2.223 \end{bmatrix}$$

由于矩阵 P 是正定的, 故系统稳定。

思考题

1. 将连续传递函数 $G(s)$ 离散化为脉冲传递函数 $G(z)$ 主要有哪些常用方法? 这些方法各自有什么特点?
2. 试述线性离散系统稳定的充分必要条件。
3. 已知下列连续传递函数 $G(s)$, 试分别用向后差分 and 双线性变换法离散化为脉冲传递函数 $G(z)$ 。

① $1/s$

② $1/s^2$

③ $\frac{a}{s(s+a)}$

④ $\frac{1}{(s+a)^2}$

⑤ $\frac{s}{s^2+a^2}$

⑥ $\frac{ab}{s(s+a)(s+b)}$

4. 已知对象的连续传递函数 $G(s)$, 试求广义对象的脉冲传递函数 $HG(z)$ 。

① $\frac{K}{s(s+a)}$

② $\frac{K}{s(s+a)(s+b)}$

③ $\frac{K}{(s+a)^2}$

④ $\frac{K(s+c)}{s(s+a)(s+b)}$

5. 已知线性离散系统的特征方程, 试判断闭环系统的稳定性。

① $45z^3 - 117z^2 + 110z - 30 = 0$

② $z^3 - 1.5z^2 - 0.25z + 0.4 = 0$

③ $z^3 - 1.001z^2 - 0.3356z + 0.00535 = 0$

④ $z^2 - z + 0.65 = 0$

⑤ $(z+1)(z+0.5)(z-2) = 0$

6. 已知单位反馈系统的开环 z 传递函数, 试判断闭环系统的稳定性。

$$\textcircled{1} \frac{0.368z + 0.264}{z^2 - 1.368z + 0.368}$$

$$\textcircled{2} \frac{z + 0.7}{(z - 1)(z - 0.368)}$$

$$\textcircled{3} \frac{10z^2 + 21z + 2}{z^3 - 1.5z^2 + 0.5z - 0.04}$$

$$\textcircled{4} \frac{10z}{z^2 - z + 0.5}$$

7. 已知线性离散系统的状态方程, 试判断系统的稳定性。

$$\textcircled{1} x(k+1) = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 0 \end{bmatrix} x(k)$$

$$\textcircled{2} x(k+1) = \begin{bmatrix} 0 & 1 \\ -0.75 & 1 \end{bmatrix} x(k) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(k)$$

$$\textcircled{3} x(k+1) = \begin{bmatrix} 1.2 & 0.7 \\ 1 & 0.6 \end{bmatrix} x(k) + \begin{bmatrix} 1.2 \\ 0.8 \end{bmatrix} u(k)$$

第3章 过程计算机系统的抗干扰技术

在构成计算机控制系统时，被控变量往往分布在生产现场的各个地方，因而计算机的工作环境恶劣，遭受频繁干扰。这些干扰将会影响系统的测控精度，降低系统的可靠性，甚至导致系统的运行混乱，进而造成生产事故。因此，抗干扰技术是计算机控制系统中一个十分重要的问题。

所谓干扰，是指有用信号以外的噪声或造成计算机控制系统的设备不能正常工作的破坏因素。干扰是客观存在的，研究干扰的目的是抑制干扰进入计算机，制定相应的、行之有效的抑制或消除干扰的措施。本章在分析干扰的来源与传播途径的基础上，讨论抑制或消除干扰的各种有效措施，主要内容包括信号隔离技术、CPU抗干扰技术、接地技术、供电技术等硬件抗干扰措施以及数字滤波技术等软件抗干扰措施。

3.1 干扰的来源与传播途径

3.1.1 干扰的来源

计算机控制系统干扰的来源是多方面的，有时甚至是错综复杂的。干扰有的来自外部，有的来自内部。

外部干扰与系统结构无关，仅由使用条件等外部环境因素决定。外部干扰环境如图 3-1 所示，有天电干扰，如雷电或大气电离作用以及其他气象引起的干扰电波；有天体干扰，如

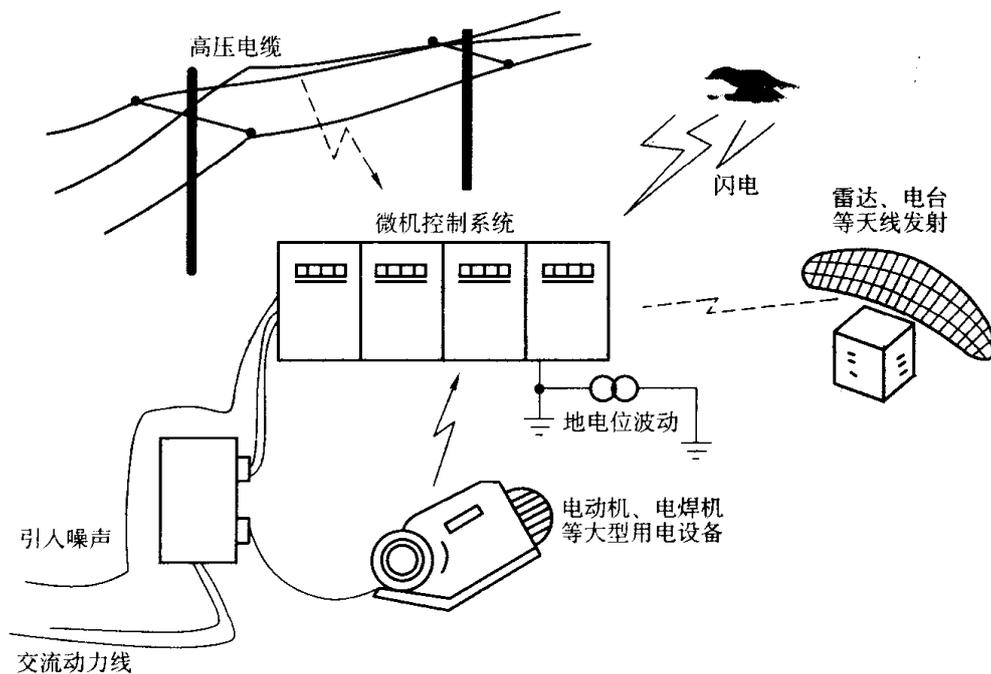


图 3-1 外部干扰环境

太阳或其他星球辐射的电磁波；有电气设备的干扰，如广播电台或通信发射台发出的电磁波，动力机械、电动机整流子、高频炉、电焊机等都会产生干扰；此外，荧光灯、开关、电流断路器、过载继电器、指示灯等具有瞬变过程的设备也会产生很大的干扰；另外来自电源的工频干扰也可视为外部干扰。

内部干扰则是由系统结构布局、制造工艺所引入的。内部干扰环境如图 3-2 所示，有分布电容、分布电感引起的耦合感应，电磁场辐射感应，长线传输造成的波反射；多点接地造成的电位差引入的干扰；装置及设备中各种寄生振荡引入的干扰，以及热噪声、闪变噪声、尖峰噪声等引入的干扰；甚至元器件产生的噪声等。

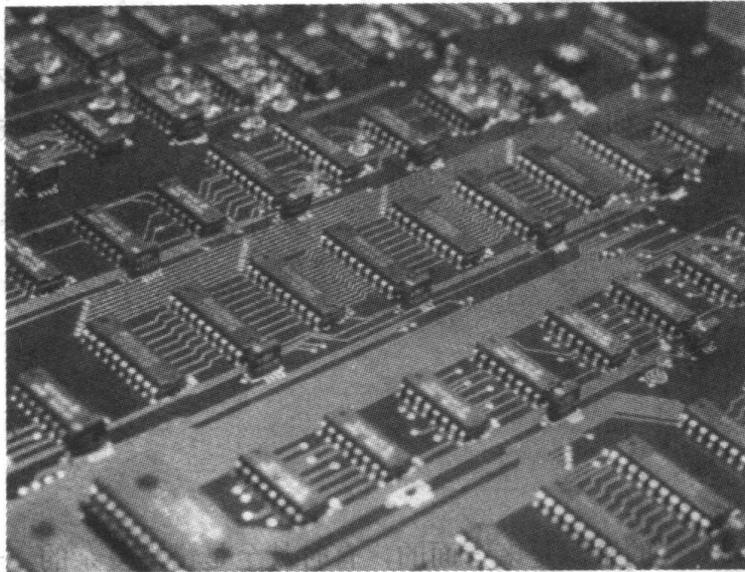


图 3-2 内部干扰环境

从机理上看，外部干扰和内部干扰的物理性质相同，因而消除或抑制它们的方法没有本质上的区别。

3.1.2 干扰的传播途径

无论什么样的干扰，总要以某种途径进入过程计算机控制系统。干扰传播的途径主要有静电耦合、磁场耦合及公共阻抗耦合。

1. 静电耦合

静电耦合是电场通过电容耦合途径窜入其他线路的。两根导线之间会构成分布电容，印制电路板上各印制电线之间、变压器线匝之间和绕线之间也都会构成分布电容。图 3-3 为两根平行导线之间的静电耦合示意图， C_{12} 是两根导线之间的分布电容， C_{1g} 和 C_{2g} 分别是两根导线对地的电容， R 是导线 2 的对地电阻。如果导线 1 上有干扰源 U_1 存在，由于分布电容 C_{12} 的存在，对频率为 ω 的干扰信号提供 $j\omega C_{12}$ 的电抗通道，导线 2 作为接受干扰的导体，则导线 2 上将出现干扰电压 U_n ，且

$$U_n = \frac{j\omega RC_{12}}{1 + j\omega R(C_{12} + C_{2g})} U_1 \quad (3-1)$$

当导线 2 对地电阻 R 很小时， $U_n \approx j\omega RC_{12} U_1$ ，即干扰电压 U_n 的大小与干扰频率成正比，

R 和 C_{12} 越小对抗干扰越有利；当导线 2 对地电阻 R 很大时，干扰电压 U_n 由电容 C_{12} 和 C_{2g} 的分压关系确定，即 $U_n \approx \frac{C_{12}}{C_{12} + C_{2g}} U_1$ ，此时的干扰比前一种情况大得多。

2. 磁场耦合

空间磁场耦合是通过导体间互感耦合进来的。在任何载流导体周围空间中都会产生磁场，而交变磁场则对其周围闭合电路产生感应电动势。在设备内部，线圈或变压器的漏磁会引起干扰；在设备外部，当两根导线平行架设时，也会产生干扰，如图 3-4 所示。

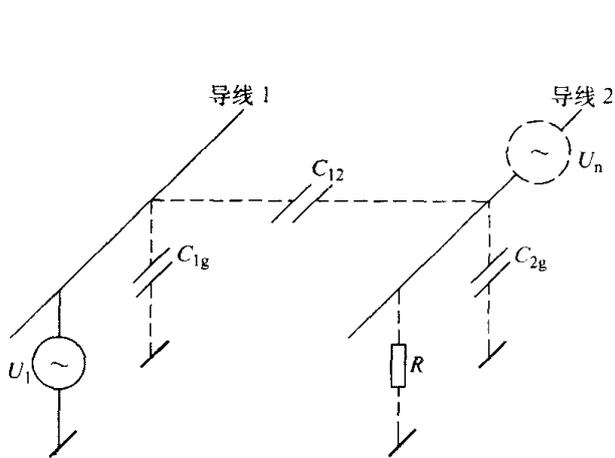


图 3-3 两平行导线间的静电耦合

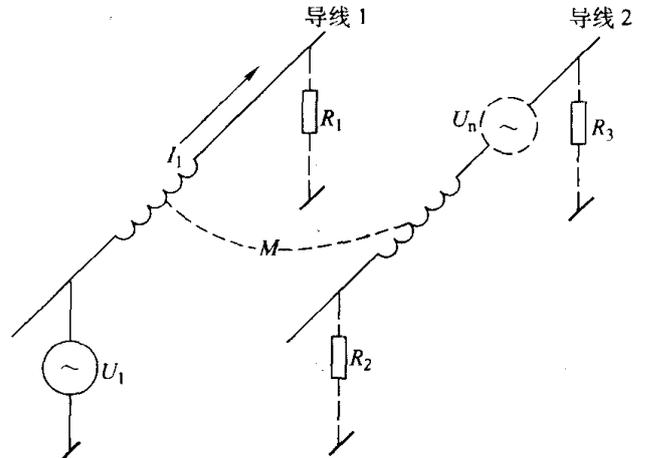


图 3-4 两导线间的磁场耦合

如果导线 1 为承载 10kVA、220V 的交流输电线，导线 2 为与之相距 1m 并平行、走线 10m 的信号线，假设两线间的互感 M 为 $4.2\mu\text{H}$ ，则干扰电压大小为

$$U_n = \omega M I_1 \approx 60\text{mV} \quad (3-2)$$

如果导线 2 是连接热电偶的信号线，那么这几十毫伏的干扰噪声足以淹没热电偶传感器的有用信号。正因为这个原因，过程控制系统中信号电缆与动力电缆必须分别敷设，且要求相距足够远的距离。

电磁场辐射也会造成干扰的耦合。当高频电流流过导体时，在该导体周围产生向空间传播的电磁波。此时整个空间充满了从长波到微波范围的电磁波，一般称为无线电波干扰。这些干扰极易通过电源线和长信号线耦合到计算机中。另外，长信号线具有天线效应，即能辐射干扰波或接收干扰波。当作为接收天线时，它与电磁波的极化面有密切的关系。例如，在大功率的广播发射台周围，当垂直极化波的电场强度为 100mV/m 时，长度为 10cm 的垂直导体便可以产生 5mV 的感应电动势，这也是一个不可小视的数字。

3. 公共阻抗耦合

公共阻抗耦合发生在两个电路的电流流经一个公共阻抗时，一个电路在该阻抗上的电压降会影响到另一个电路，从而产生相互干扰。

图 3-5 是一个公共电源线的阻抗耦合示意图。

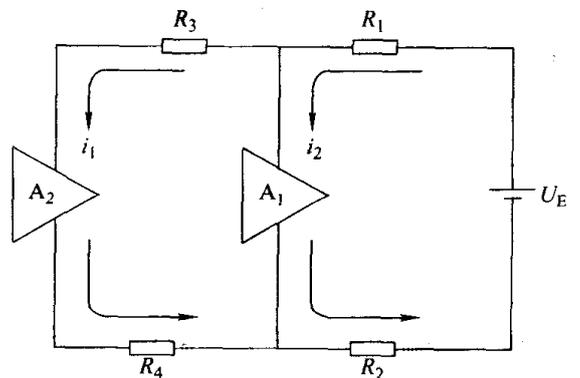


图 3-5 公共电源线的阻抗耦合

在一块印制电路板上,运算放大器 A_1 和 A_2 是两个独立的回路,但都接到一个公共电源 U_E 上,电源回流线的等效电阻为 R_1 , R_2 是两个回路的公共阻抗。当回路电流 i_1 变化时,在 R_1 、 R_2 上产生的电压降变化就会影响到另一个回路电流 i_2 ; 反之,也是如此。

在过程计算机控制系统中,常包含有模拟信号和数字信号,为了避免频繁变化的数字信号对模拟信号的干扰,一般要求模拟信号和数字信号分别接地,目的就是为了减小数字信号回路与模拟信号回路间的公共阻抗干扰。

3.2 串模干扰、共模干扰和长线传输干扰

按干扰的作用方式不同,可以分为串模干扰、共模干扰和长线传输干扰三类。这三类干扰的特点具有很大的不同,针对它们各自的特点,抑制干扰的措施也各不相同。

3.2.1 串模干扰及其抑制

1. 串模干扰

所谓串模干扰,是指叠加在被测信号上的干扰噪声,即串联于信号源回路之中,也称横向干扰或正态干扰。其表现形式如图 3-6a 所示,其中 U_s 为信号源, U_n 为串模干扰电压。

产生串模干扰的原因有分布电容的静电耦合、空间电磁场的磁场耦合、长线传输的互感以及 50Hz 的工频干扰等。在图 3-6b 中,邻近导线(干扰线)有交变电流 I_a 流过,由 I_a 产生的电磁干扰信号就会通过分布电容 C_1 和 C_2 的耦合,引至计算机控制系统的输入端。

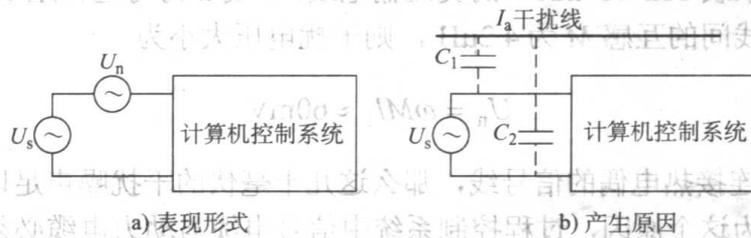


图 3-6 串模干扰

2. 串模干扰的抑制

对串模干扰的抑制较为困难,因为干扰 U_n 直接与信号源 U_s 串联在一起。过程计算机控制系统中串模干扰的抑制措施主要从两个方面考虑,一是切断串模干扰的传播途径;二是根据串模干扰信号的频率特点,采用滤波的办法,把相对低频的有用信号分离出来。

(1) 切断串模干扰的传播途径。串模干扰主要是来源于空间电磁场干扰,切断这类串模干扰的传播途径最有效的方法是屏蔽,工程上常采用带有金属编织物或护套屏蔽层的双绞线或同轴电缆作信号引线。为了保证良好的电磁屏蔽效果,过程计算机控制系统中,屏蔽层一般需遵守

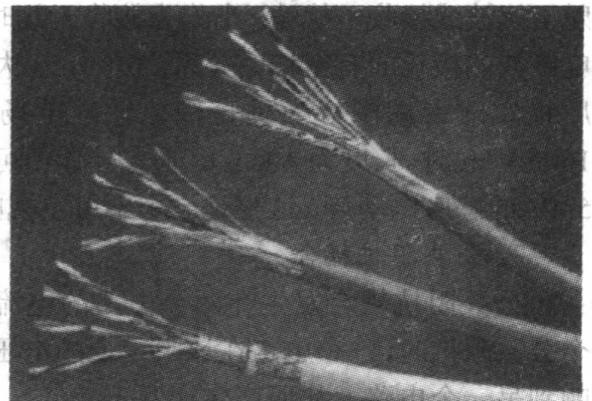


图 3-7 双绞线

单点接地的原则。

采用双绞线的目的是为了进一步减少电磁感应，使外界电磁场在双绞线相邻的小环路上形成相反方向的感应电动势，从而相互抵消减弱干扰作用。双绞线相邻的扭绞处之间为双绞线的节距，双绞线节距不同会对串模干扰起到不同的抑制效果，即节距越小，干扰的衰减比越大，抑制效果越好。

(2) 滤波。采用滤波的办法抑制串模干扰是一种常用的方法。根据串模干扰频率与被测信号频率的分布特性，决定选用具有低通、高通或带通等传递性的滤波器。

过程计算机控制系统中，一般采用低通滤波，滤波器可以是硬件的，也可以是本章后面将介绍的由软件实现的数字滤波器。硬件滤波器一般是采用电阻 R 、电容 C 、电感 L 等无源元件构成。图 3-8 所示是最常用的一种二阶 RC 滤波器，其缺点是信号有很大的衰减。为了把增益和频率特性结合起来，可以采取以反馈放大器为基础的有源滤波器，如图 3-9 所示。这对于小信号尤其重要，它不仅可以达到滤波的效果，而且能够提高信号增益；其缺点是电路稍复杂。

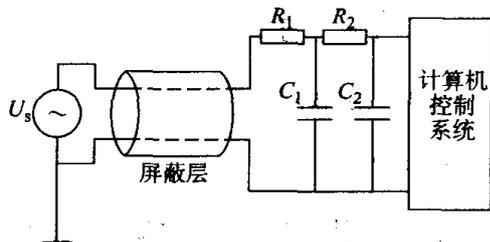


图 3-8 无源阻容低通滤波

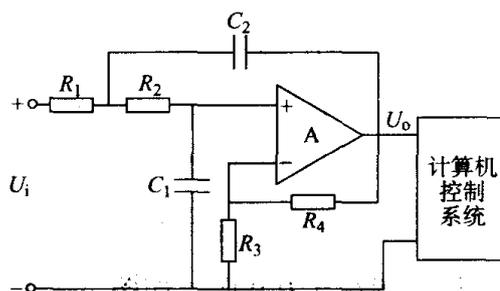


图 3-9 有源低通滤波

3.2.2 共模干扰及其抑制

1. 共模干扰

所谓共模干扰，是指信号放大器两个输入端上共有的干扰电压，可以是直流电压也可以是交流电压，其幅值达几伏甚至更高，这取决于产生干扰的现场环境和计算机等设备的接地情况。共模干扰也称纵向干扰或共态干扰，其表现形式如图 3-10a 所示。

产生共模干扰的原因是计算机的地或信号放大器的地与现场信号源的地之间，通常长达几十米以至几百米，在两地之间往往存在着一个电位差 U_{cm} ，如图 3-10b 所示。显然， U_{cm} 是放大器输入端上共有的干扰电压，故称共模干扰电压。

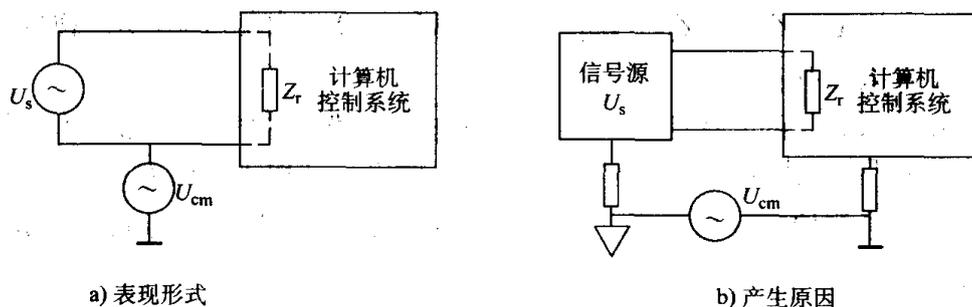


图 3-10 共模干扰

为了了解共模电压 U_{cm} 对输入系统的影响，人们常将共模电压转换成串模干扰的形式而加入到系统的输入端来分析。定量地衡量一个输入系统抑制共模干扰的能力，常用共模抑制比（CMRR, Common Mode Rejection Ratio）来表示，即

$$CMRR = 20 \lg \frac{U_{cm}}{U_n} \quad (3-3)$$

式中， U_{cm} 是共模干扰电压； U_n 是由 U_{cm} 转化成的串模干扰电压。

在计算机控制系统中，被测信号有单端对地输入和双端不对地输入两种输入方式，图 3-11 分别讨论在这两种输入方式时共模干扰的引入情形。

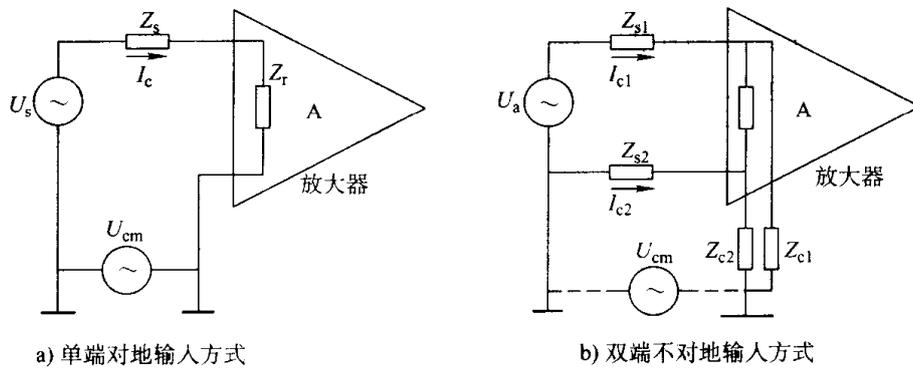


图 3-11 两种输入方式时共模电压的引入

采用单端输入方式时，如图 3-11a 所示，由共模电压 U_{cm} 引入到放大器输入端的串模干扰电压 U_{n1} 为

$$U_{n1} = I_c Z_s = \frac{Z_s}{Z_s + Z_r} U_{cm} \approx \frac{Z_s}{Z_r} U_{cm} \quad (3-4)$$

式中， Z_s 是信号源内阻（含信号引线电阻）； Z_r 是放大器输入阻抗， $Z_r \gg Z_s$ 。

显然， Z_r 越大越有利于提高抗共模干扰的能力。

当采用双端输入时，如图 3-11b 所示，由共模电压 U_{cm} 引入放大器输入端的串模干扰 U_{n2} 为

$$U_{n2} = I_{c1} Z_{s1} - I_{c2} Z_{s2} = \frac{Z_{s1}}{Z_{s1} + Z_{c1}} U_{cm} - \frac{Z_{s2}}{Z_{s2} + Z_{c2}} U_{cm} \approx \left(\frac{Z_{s1}}{Z_{c1}} - \frac{Z_{s2}}{Z_{c2}} \right) U_{cm} \quad (3-5)$$

式中， Z_{s1} 、 Z_{s2} 为信号源内阻（含信号引线电阻）； Z_{c1} 、 Z_{c2} 为放大器输入端对地的漏阻抗。

显然，为了提高抗共模干扰的能力，信号引线要尽量短， Z_{c1} 、 Z_{c2} 则要尽量大，而且阻值要尽量相近。理论上，若 $Z_{s1}/Z_{c1} = Z_{s2}/Z_{c2}$ ，则 $U_{n2} = 0$ 。由此可见，双端输入时，抗共模干扰能力比单端输入时强得多。

显然，单端输入方式的 CMRR 较小，说明它的抗共模干扰能力较差；而双端输入方式，由 U_{cm} 引入的串模干扰电压 U_n 较小，CMRR 较大，所以抗共模干扰能力很强。在计算机控制的输入系统中，考虑到抑制共模干扰，大都尽量不采用单端对地输入方式，一般都使用双端不对地输入方式。

2. 共模干扰的抑制

通过前面的分析可知，共模干扰产生的主要原因是不同“地”之间存在共模电压，以及

模拟信号系统对地的漏阻抗。除了采用双端输入方式外，抑制共模干扰最有效的方法就是切断不同“地”之间的电信号联系，共模干扰的抑制措施主要有变压器隔离、光电隔离和浮地屏蔽。

(1) 变压器隔离。利用变压器把模拟信号电路与数字信号电路隔离开来，即把“模拟地”与“数字地”断开。被测信号通过变压器磁场耦合获得通路，而共模干扰电压 U_{cm} 由于不能形成回路而得到有效的抑制。另外，隔离前和隔离后应分别采用两组互相独立的电源，切断电源部分的地线联系。如图 3-12 所示，被测信号 U_s 经放大后，首先通过调制器变换成交流信号，经隔离变压器 T 传输到二次侧，然后用解调器再将它变换为直流信号 U_{s2} ，再对 U_{s2} 进行 A/D 转换。

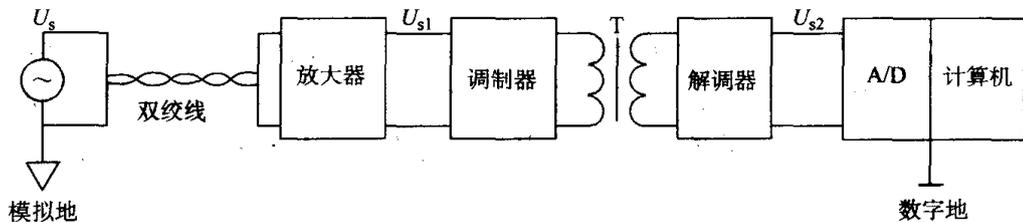


图 3-12 变压器隔离

(2) 光电隔离。利用光耦合器件进行信号隔离是目前计算机控制系统中最常用的一种抗干扰方法。光耦合器件是由发光二极管和光敏晶体管封装在同一个管壳内组成的，发光二极管两端为信号输入端，光敏晶体管的集电极和发射极为光耦合器的输出端，它们之间的信号传输是靠发光二极管在信号电压的控制下发光，传送由光敏晶体管来完成的。其输入、输出具有类似普通晶体管的输入、输出特性，即存在着截止区、饱和区与线性区三部分。利用光耦合器的开关特性（即工作在截止区、饱和区），可对数字信号进行隔离；利用光耦合器的线性放大区，也可实现对模拟信号的隔离。如图 3-13 所示。

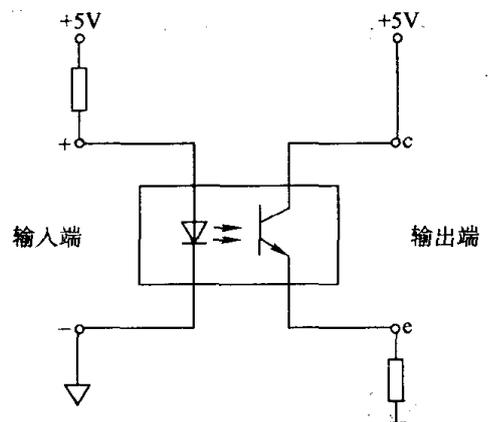


图 3-13 光耦合器

利用光耦合器进行信号隔离最突出的优点是，由于是靠光传送信号，完全切断了输入与输出电路之间地线的联系；同时，发光二极管是由电流驱动的，且动态电阻非常小，而干扰源的内阻一般很大，即使是在干扰电压幅值较高的情况下，由于没有足够的能量，仍不能使发光二极管发光，从而有效地抑制干扰信号。

考虑到变压器隔离和光电隔离技术在计算机控制系统输入、输出抗干扰方面的重要性，第 3.4 节中将进一步详细讨论不同输入、输出形式下如何进行信号的隔离。

(3) 浮地屏蔽。浮地屏蔽是利用屏蔽层让输入信号的“模拟地”浮空，使共模输入阻抗大为提高，共模电压在输入回路中引起的共模电流减小，从而达到抑制共模干扰的目的。

图 3-14 给出了一种浮地输入双层屏蔽放大电路。这里，计算机部分采用内、外双层屏蔽，且内屏蔽层对外屏蔽层（机壳地）是浮空的，而内层与信号源及信号线屏蔽层是在信号端单点接地的，被测信号到控制系统中的放大器是采用双端差动输入方式。图 3-14b 中， Z_{s1} 、 Z_{s2} 为信号源内阻及信号引线电阻， Z_{s3} 为信号线的屏蔽电阻， Z_{c1} 、 Z_{c2} 为放大器输入级对内屏

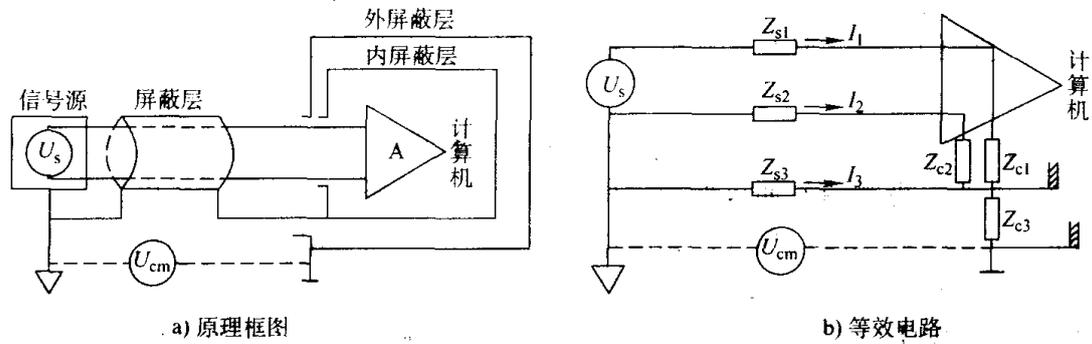


图 3-14 浮地输入双层屏蔽放大电路

蔽层的漏阻抗， Z_{c3} 为内屏蔽层与外屏蔽层之间的漏阻抗。由于 Z_{c1} 、 Z_{c2} 和 Z_{c3} 达到数十兆欧以上，而 Z_{s1} 、 Z_{s2} 和 Z_{s3} 一般很小，远小于前者，这样模拟地和数字地之间的共模电压 U_{cm} 就不会直接引入放大器，而是先经 Z_{s3} 和 Z_{c3} 产生共模电流 I_3 。由于 Z_{s3} 较小，故 I_3 在 Z_{s3} 上的压降 U_{s3} 也很小，可把它看成一个已受到抑制的新的共模干扰源 U_{n1} ，即

$$U_{n1} = U_{s3} = U_{cm} \frac{Z_{s3}}{Z_{c3} + Z_{s3}} \approx U_{cm} \frac{Z_{s3}}{Z_{c3}} \quad (3-6)$$

这时，放大器输入端之间所受到的共模电压的影响 U_{n2} 即为 Z_{s1} 、 Z_{s2} 上产生的干扰电压 U_{s1} 、 U_{s2} 的差值，即

$$\begin{aligned} U_{n2} &= U_{s1} - U_{s2} = U_{n1} \left(\frac{Z_{s1}}{Z_{s1} + Z_{c1}} - \frac{Z_{s2}}{Z_{s2} + Z_{c2}} \right) \\ &\approx U_{n1} \left(\frac{Z_{s1}}{Z_{c1}} - \frac{Z_{s2}}{Z_{c2}} \right) \\ &\approx U_{cm} \frac{Z_{s3}}{Z_{c3}} \left(\frac{Z_{s1}}{Z_{c1}} - \frac{Z_{s2}}{Z_{c2}} \right) \end{aligned} \quad (3-7)$$

显然，上述系统可以近似看作前面介绍的单端对地输入和双端不对地输入两种方式的合成，相当于对共模干扰进行了两次抑制，当然其共模抑制比也高得多。

3.2.3 长线传输干扰及其抑制

1. 长线传输干扰

计算机控制系统是一个从生产现场的传感器到计算机，再到生产现场执行机构的庞大系统。由生产现场到计算机的连线往往长达几十米，甚至数百米；即使在中央控制室内，各种连线也有几米到十几米。对于采用高速集成电路的计算机来说，长线的“长”是一个相对的概念，是否为“长线”取决于集成电路的运算速度。例如，对于纳秒级的数字电路来说，1m 左右的连线就应当作长线来看待。信号在长线中传输会遇到 3 个问题：一是长线传输会受到外界干扰，二是具有信号延时，三是高速度变化的信号在长线中传输时，还会出现波反射现象。

信号在长线中传输时，由于传输线的分布电容和分布电感的影响，信号会在传输线内部产生正向前进的电压波和电流波，称为入射波；另外，如果传输线的终端阻抗与传输线的阻抗不匹配，那么当入射波到达终端时，便会引起反射；同样，反射波到达传输线始端时，如果始端阻抗不匹配，也会引起新的反射。这种信号的多次反射现象，使信号波形产生严重的

畸变，并且引起干扰脉冲。

2. 长线传输干扰的抑制

对长线传输易于受到外界干扰的问题，一般可以通过采用带屏蔽的信号电缆来抑制；解决传输迟延的问题，没有简便的方法，只能是尽量减少信号线的长度；而波反射现象引起的干扰可以通过终端阻抗匹配或始端阻抗匹配的方法，将干扰抑制到最低限度。

(1) 波阻抗的测量。为了进行阻抗匹配，必须事先知道传输线的波阻抗 R_p ，波阻抗的测量如图 3-15 所示。图中信号传输线为双绞线，在输入线始端通过与非门加入标准矩形波信号，用示波器观察门 D_A 的波形，调节传输线终端的可变电阻 R ，当门 D_A 输出的波形不发生畸变时，即传输线的波阻抗与终端阻抗完全匹配，波反射完全消失，这时的 R 值就是该传输线的波阻抗。

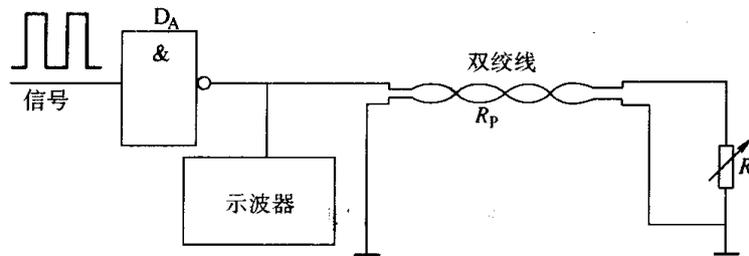


图 3-15 传输线波阻抗的测量

为了避免外界干扰的影响，在计算机中常采用双绞线和同轴电缆作信号线。双绞线的波阻抗一般在 $100\sim 200\Omega$ 之间，绞花越密，波阻抗越低。同轴电缆的波阻抗一般在 $50\sim 100\Omega$ 范围内。根据传输线的基本理论，无损耗导线的波阻抗 R_p 为

$$R_p = \sqrt{L_0 / C_0} \quad (3-8)$$

式中， L_0 为单位长度的电感 (H)； C_0 为单位长度的电容 (F)。

(2) 终端匹配。最简单的终端匹配方法如图 3-16a 所示。如果传输线的波阻抗是 R_p 时，那么当 $R = R_p$ 时，便实现了终端匹配，消除了波反射。此时终端波形和始端波形的形状一致，只是时间上滞后。由于并联了终端匹配电阻，相当于加大了负载，使波形的高电平下降，从而降低了高电平的抗干扰能力，但对波形的低电平没有影响。

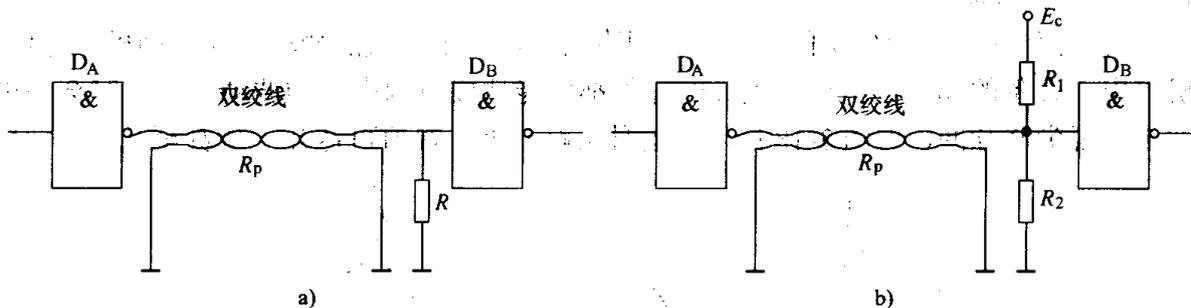


图 3-16 终端阻抗匹配

为了克服上述匹配方法的缺点，可采用图 3-16b 所示的终端匹配方法。其等效电阻 R 为

$$R = R_1 R_2 / (R_1 + R_2) \quad (3-9)$$

适当调整 R_1 和 R_2 的阻值，可使 $R=R_p$ 。这种匹配方法也能消除波反射，优点是波形的高

电平下降较少, 缺点是低电平抬高, 从而降低了低电平的抗干扰能力。为了同时兼顾高电平和低电平两种情况, 可选取 $R_1=R_2=2R_p$ 。实践中宁可使高电平降低得稍多点, 而让低电平抬高得少点, 可通过适当选取电阻 R_1 和 R_2 , 并使 $R_1 > R_2$ 来达到此目的, 当然还要保证等效电阻 $R=R_p$ 。

(3) 始端匹配。在传输线始端串入电阻 R , 如图 3-17 所示, 也能基本上消除反射, 达到改善波形的目的。一般选择始端匹配电阻 R 为

$$R = R_p - R_{sc} \quad (3-10)$$

式中, R_{sc} 为门 D_A 输出低电平时的输出阻抗。

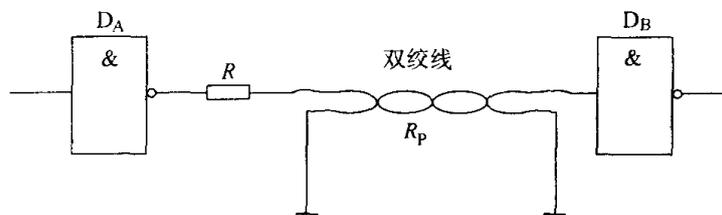


图 3-17 始端匹配

这种匹配方法的优点是波形的高电平不变, 缺点是波形低电平会抬高, 原因是终端门 D_B 的输入电流在始端匹配电阻 R 上的压降所造成的。显然, 终端所带负载门个数越多, 则低电平抬高得越显著。

3.3 供电技术与接地技术

3.3.1 供电技术

计算机控制系统一般是由交流电网供电, 电网电压与频率的波动将直接影响到控制系统的可靠性与稳定性。实践表明, 电源的干扰是计算机控制系统的一个主要干扰, 抑制这种干扰的主要措施有以下几个方面。

1. 交流电源系统

理想的交流电应该是 50Hz 的正弦波。但事实上, 由于负载的变动, 如电动机、电焊机、鼓风机等电器设备的起停, 甚至荧光灯的开关都可能造成电源电压的波动, 严重时会使电源正弦波上出现尖峰脉冲, 如图 3-18 所示。这种尖峰脉冲, 幅值可达几十伏甚至几千伏, 持续

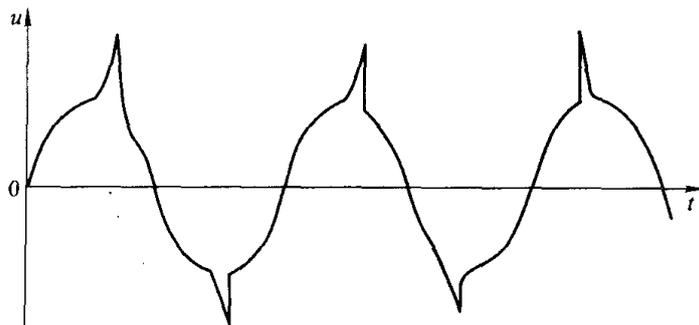


图 3-18 交流电源正弦波上的尖峰脉冲

时间也可达几毫秒之久,极易造成计算机的“死机”,甚至会损坏硬件,对系统威胁极大。在硬件上可以用以下方法加以解决。

(1) 选用供电比较稳定的进线电源。计算机控制系统的电源进线要尽量选用比较稳定的交流电源线,至少不要将控制系统接到负载变化大、晶闸管设备多或者有高频设备的电源上。

(2) 利用干扰抑制器消除尖峰干扰。干扰抑制器使用简单,利用干扰抑制器消除尖峰干扰的电路如图 3-19 所示。干扰抑制器是一种无源四端网络,目前已有产品出售。

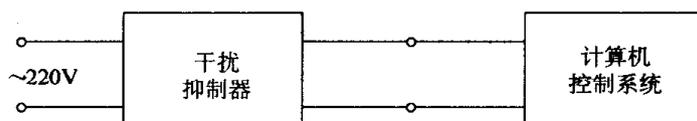


图 3-19 利用干扰抑制器的电源系统

(3) 采用交流稳压器稳定电网电压。计算机控制的一般交流供电系统如图 3-20 所示。图中交流稳压器是为了抑制电网电压的波动,提高计算机控制系统的稳定性,交流稳压器能把输出波形畸变控制在 5% 以内,还可以对负载短路起限流保护作用。低通滤波器是为了滤除电网中混杂的高频干扰信号,以保证 50Hz 基波通过。

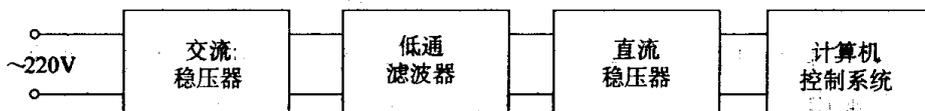


图 3-20 一般交流供电系统

(4) 利用 UPS 保证不间断供电。电网瞬间断电或电压突然下降等掉电事件会使计算机系统陷入混乱状态,是可能产生严重事故的恶性干扰。对于要求更高的计算机控制系统,可以采用不间断电源(UPS)向系统供电,如图 3-21 所示。

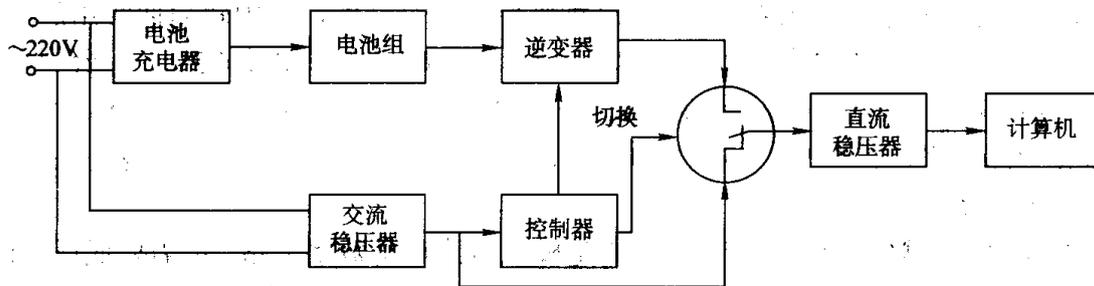


图 3-21 不间断电源(UPS)供电系统

正常情况下,由交流电网通过交流稳压器、切换开关、直流稳压器供电至计算机系统;同时交流电网也给电池组充电。所有的 UPS 设备都装有一个或一组电池和传感器,并且也包括交流稳压设备。如果交流供电中断,系统中的断电传感器检测到断电后就会通过控制器将供电通路在极短的时间内(3ms)切换到电池组,从而保证计算机控制系统不因停电而中断。这里,逆变器能把电池直流电压逆变到正常电压频率和幅度的交流电压,具有稳压和稳频的双重功能,提高了供电质量。

(5) 掉电保护电路。对于没有使用 UPS 的计算机控制系统,为了防止掉电后 RAM 中的信息丢失,可以采用镍电池对 RAM 数据进行掉电保护。图 3-22 是某一计算机系统 64KB 存

储板所使用的掉电保护电路。系统电源正常工作时，由外部电源+5V 供电，A 点电平高于备用电池（3V）电压， V_{D2} 截止，存储器由主电源（+5V）供电。系统掉电时，A 点电位低于备用电池电压， V_{D1} 截止， V_{D2} 导通，由备用电池向 RAM 供电。当系统恢复供电时， V_{D1} 重新导通， V_{D2} 截止，又恢复主电源供电。

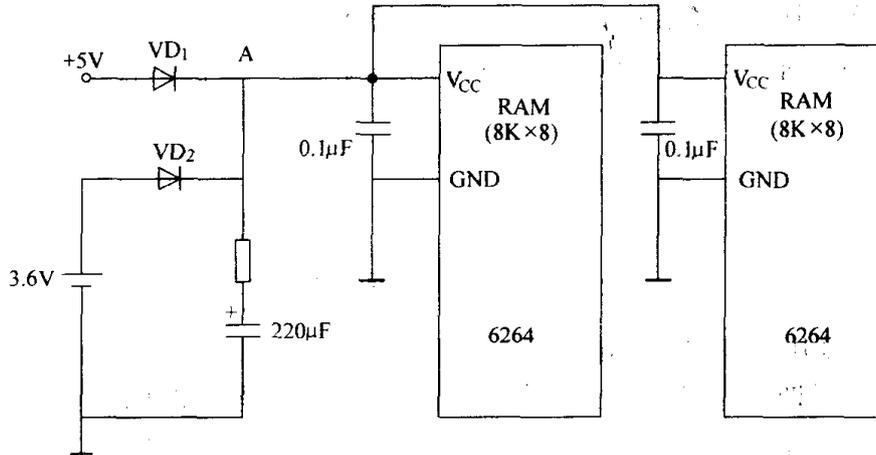


图 3-22 掉电保护电路

对于没有采用镍电池进行掉电保护的一些控制系统，至少应设置电源监控电路，即硬件掉电检测电路。在掉电电压下降到 CPU 最低工作电压之前应能提出中断申请（提前时间为几百微秒到数毫秒），使系统能及时对掉电作出保护反应——在掉电中断子程序中，首先进行现场保护，把当时的重要参数、中间结果以及输入、输出状态做出妥善处理，并在片内 RAM 中设置掉电标志。当电源恢复正常时，CPU 重新复位，复位后应首先检查是否有掉电标记。如果没有，则按一般开机程序执行，即首先初始化系统；如果有掉电标记，则说明本次复位是掉电保护之后的复位，不应将系统初始化，而应按与掉电中断子程序相反的方式恢复现场，以一种合理的安全方式使系统继续工作。这种监控电路有许多，其中较简便实用的一种应用电路如图 3-40 所示的 X5045 与 CPU 的接口电路。上电时，电压超过 4.5V 后，经过约 200ms 的稳定时间后 RESET 信号由高电平变为低电平；掉电时，当电源电压低于 4.5V 时，RESET 信号立即变为高电平，使 CPU 响应中断申请并转入掉电中断子程序，进行现场保护。

2. 直流电源系统

在自行研制的计算机控制系统中，无论是模拟电路还是数字电路，都需要低压直流供电。为了进一步抑制来自于电源方面的干扰，一般在直流电源侧也要采用相应的抗干扰措施。

(1) 交流电源变压器的屏蔽。把高压交流变成低压直流的简单方法是用交流电源变压器。因此，对电源变压器设置合理的静电屏蔽和电磁屏蔽，就是一种十分有效的抗干扰措施，通常将电源变压器的一次、二次绕组分别加以屏蔽，一次绕组屏蔽层与铁心同时接地，如图 3-23a 所示。在要求更高的场合，可采用层间也加屏蔽的结构，如图 3-23b 所示。

(2) 采用直流开关电源。直流开关电源是一种脉宽调制型电源，由于脉冲频率高达 20kHz，所以甩掉了传统的工频变压器，具有体积小、重量轻、效率高 (>70%)、适应电网电压变化范围大、电网电压变化时不会输出过电压或欠电压及输出电压保持时间长等优点。开关电源一次侧、二次侧之间具有较好的隔离，对于交流电网上的高频脉冲干扰有较强的隔离能力。现在已有许多直流开关电源产品，都有几个独立的电源，如 $\pm 5V$ 、 $\pm 12V$ 及 $\pm 24V$ 等。

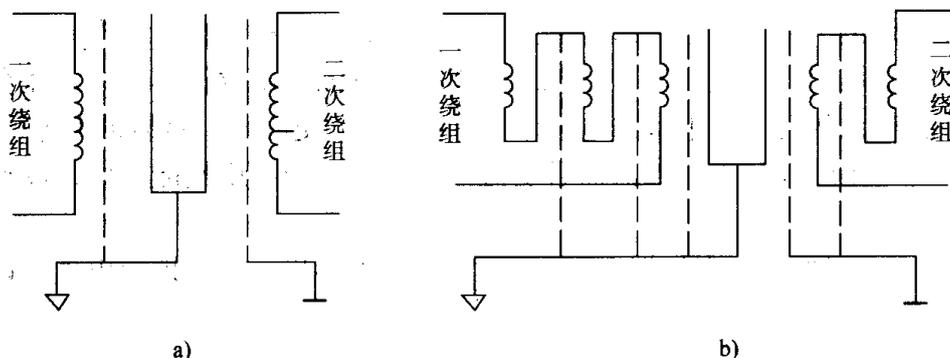


图 3-23 电源变压器的屏蔽

(3) 采用 DC—DC 变换器。如果系统供电电网波动较大，或者对直流电源的精确度要求较高，就可以采用 DC-DC 变换器，这种变换器可以将一种电压的直流电源，变换成另一种电压的直流电源，可分为升压型、降压型或升压/降压型。DC-DC 变换器具有体积小、性能价格比高、输入电压范围大、输出电压稳定（有的还可调）、环境温度范围广等一系列优点。

显然，采用 DC-DC 变换器可以方便地实现电池供电，从而可用于制造便携式或手持式计算机测控装置。

同时，过程计算机控制系统中也常把 DC-DC 变换器作为电源隔离器来使用。

(4) 为每块电路板设置独立的直流电源。当一台计算机测控系统有几块功能电路板时，为了防止板与板之间的相互干扰，可以对每块板的直流电源采取分散独立供电系统。在每块板上装一块或几块三端稳压集成块组成稳压电源，每个功能板单独对电压过载进行保护，不致因某个稳压块出现故障而使整个系统遭到破坏，而且也减少了公共阻抗的相互耦合，大大提高了供电的可靠性，也有利于电源散热。

(5) 集成电路块的 V_{CC} 端加旁路电容。集成电路的开关高速动作时会产生噪声，因此无论电源装置提供的电压多么稳定， V_{CC} 和 GND 端也会产生噪声。为了降低集成电路的开关噪声，在电路板上的每一块 IC 上都接有高频特性好的旁路电容，将开关电流经过的线路局限在板内一个极小的范围内。旁路电容可用 $0.01 \sim 0.1 \mu\text{F}$ 的陶瓷电容器，旁路电容器的引线要短而且紧靠需要旁路的集成器件的 V_{CC} 或 GND 端，否则会毫无意义。

3.3.2 接地技术

广义的接地包含两方面的含义，即接实地和接虚地。接实地指的是与大地连接；接虚地指的是与电位基准点连接，当这个基准点与大地电气绝缘，则称为浮地连接。正确、合理的接地技术对计算机控制系统极为重要，接地的目的有两个：一是为了保证控制系统稳定可靠地运行，防止地环路引起的干扰，常称为工作接地；二是为了避免操作人员因设备的绝缘损坏或下降遭受触电危险和保证设备的安全，称为保护接地。这里主要讨论工作接地技术。

在计算机控制系统中，大致有以下几种地线：模拟地、数字地、信号地、系统地、交流地和保护地。

模拟地作为传感器、变送器、放大器、A/D 和 D/A 转换器中模拟电路的零电位。由于模拟信号的信号比较小，而且与生产现场连接，因此对模拟信号有精度方面的严格要求。有时为区别远距离传感器的弱信号地与主机的模拟地关系，把传感器的地又叫信号地。

数字地作为计算机各种数字电路的零电位，应该与模拟地分开，避免模拟信号受数字脉

冲的干扰。系统地是上述几种地的最终回流点，直接与大地相连作为基准零电位。

交流地是计算机交流供电的动力线地或称零线，它的零电位很不稳定。在交流地上任意两点之间往往就有几伏乃至几十伏的电位差存在；另外，交流地也容易带来各种干扰。因此，交流地绝不允许与上述几种地相连，而且交流电源变压器的绝缘性能要好，以绝对避免漏电现象。

保护地也叫安全地、机壳地或屏蔽地，目的是使设备机壳与大地等电位，以避免机壳带电危及人身及设备安全。

以上这些地线如何处理，究竟是接地还是浮地，到底是一点接地还是多点接地，这些都是实际控制系统设计、安装、调试中的重要问题。

1. 单点接地与多点接地

根据接地理论分析，低频电路应单点接地，这主要是避免形成产生干扰的地环路；高频电路应该就近多点接地，这主要是为了避免“长线传输”引入的干扰。一般来说，当频率低于 1MHz 时，采用单点接地方式为好；当频率高于 10MHz 时，采用多点接地方式为好；而在 1~10MHz 之间，如果采用单点接地，其地线长度不得超过波长的 1/20，否则应采用多点接地方式。在工业过程控制系统中，信号频率大多小于 1MHz，所以通常采用单点接地方式，如图 3-24 所示。

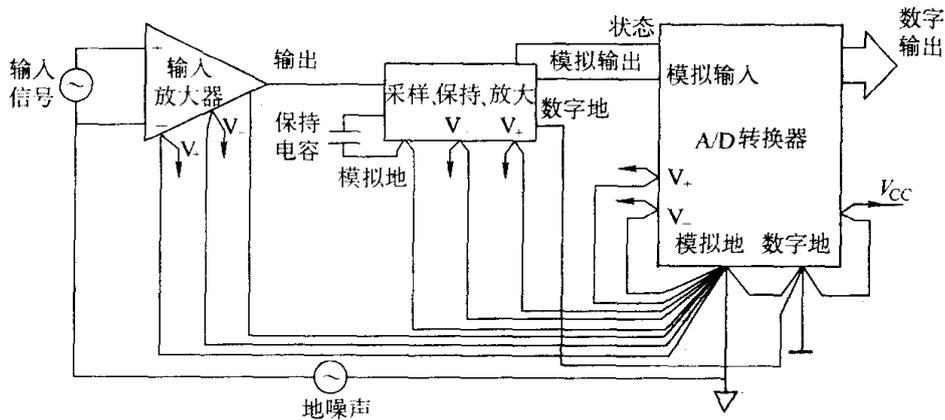


图 3-24 单点接地方式

在计算机控制系统中，各种地一般应采用分别回流法单点接地。模拟地、数字地、安全地的分别回流法如图 3-25 所示。汇流条由多层铜导体构成，截面呈矩形，各层之间有绝缘层。采用多层汇流条可以减少自感，从而减少干扰的窜入途径。在稍复杂的系统中，可以分别使用横向汇流条及纵向汇流条，机柜内各层机架之间分别设置汇流条，以最大限度减小公共阻抗的影响。在空间将数字地汇流条与模拟地汇流条间隔开，以避免通过汇流条间电容产生耦合。安全地（机壳地）始终与模拟地和数字地隔离开。这些地之间只是在最后才汇聚于一

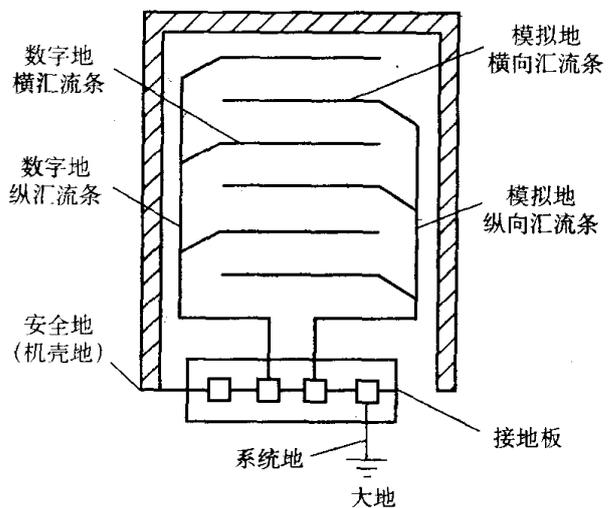


图 3-25 单点回流法接地方式

点，而且常通过铜接地板交汇，然后用截面积不小于 30mm^2 的多股软铜线焊接在接地板上并深埋地下。

2. 输入系统的接地

在计算机控制输入系统中，传感器、变送器和放大器通常采用屏蔽罩，而信号的传送往往使用屏蔽线。对于屏蔽层的接地要慎重，也应遵守单点接地原则。输入信号源有接地和浮地两种情况，接地电路也有两种情况。在图 3-26a 中，信号源端接地，而接收端放大器浮地，则屏蔽层应在信号源端接地（A 点）。而在图 3-26b 中却相反，信号源浮地，接收端接地，则屏蔽层应在接收端接地（B 点）。采用单点接地是为了避免在屏蔽层与地之间的回路电流，从而通过屏蔽层与信号线间的电容产生对信号线的干扰。一般输入信号比较小，而模拟信号又容易受到干扰。因此，对输入系统的接地和屏蔽应格外重视。

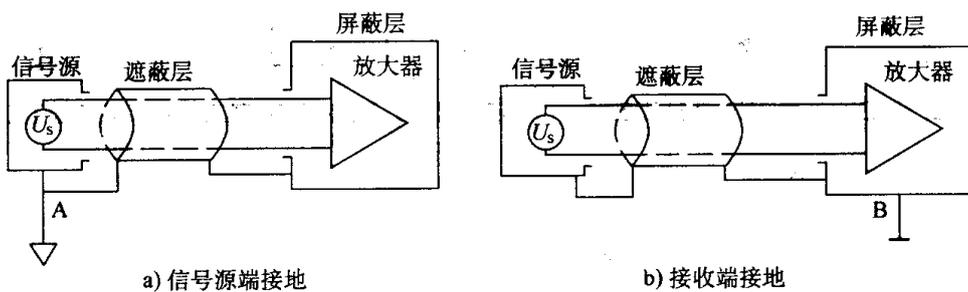


图 3-26 输入系统接地方式

高增益放大器常用金属罩屏蔽起来，但屏蔽罩的接地也要合理，否则将引起干扰。放大器与屏蔽罩间存在寄生电容，如图 3-27a 所示。由图 3-27a 的等效电路图 3-27b 可见，寄生电容 C_1 和 C_2 使放大器的输出端到输入端有一反馈通路，如不将此反馈消除，放大器可能产生振荡。解决的办法就是将屏蔽罩接到放大器的公共端，如图 3-27c 所示。这样便将寄生电容短路了，从而消除了反馈通路。

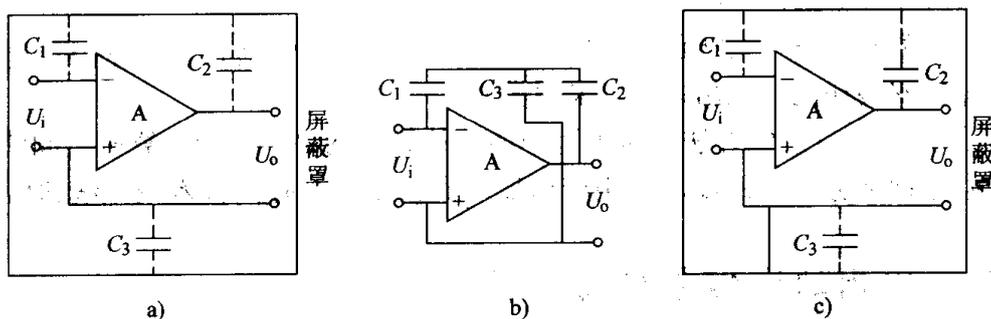


图 3-27 放大器公共端接屏蔽罩

3. 印制电路板的布线分布

设计印制电路板应遵守下列原则，以免系统内部地线产生干扰。

- 1) TTL、CMOS 器件的地线要呈辐射状，不能形成环形。
- 2) 印制电路板上的地线要根据通过的电流大小决定其宽度，不要小于 3mm ，在可能的情况下，地线越宽越好。
- 3) 旁路电容的地线不能过长，应尽量缩短。

4) 大电流的零电位地线应尽量宽, 而且必须和小信号的地线分开。

4. 主机系统的接地

计算机本身接地, 同样是为了防止干扰, 提高运行的可靠性。下面介绍三种主机接地方式。

(1) 全机一点接地。计算机控制系统的主机架内采用如图 3-25 所示的单点回流法接地方式。主机地与外部设备地的连接采用一点接地, 如图 3-28 所示。为了避免多点接地, 各机柜用绝缘板垫起来。这种接地方式安全可靠, 具有一定的抗干扰能力, 一般接地电阻选为 $4\sim 10\Omega$ 。接地电阻越小越好, 但接地电阻太小, 接地极的施工就会变得困难。

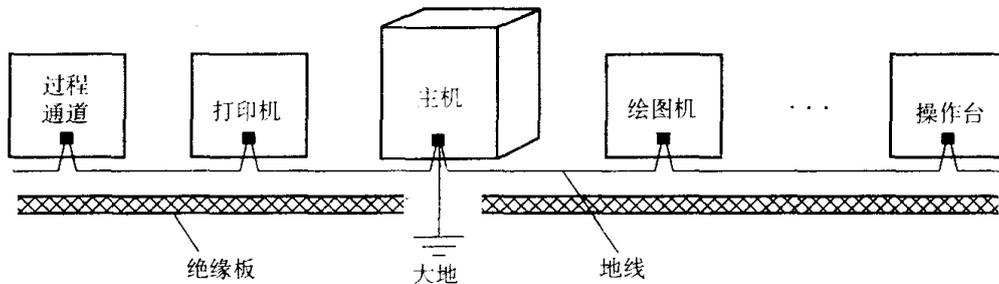


图 3-28 全机一点接地

(2) 主机外壳接地, 机心浮空。为了提高计算机系统的抗干扰能力, 将主机外壳作为屏蔽罩接地, 而把机内器件架与外壳绝缘, 绝缘电阻大于 $50M\Omega$, 即机内信号地浮空; 如图 3-29 所示。这种方法安全可靠, 抗干扰能力强, 但制造工艺复杂, 一旦绝缘电阻降低就会引入干扰。

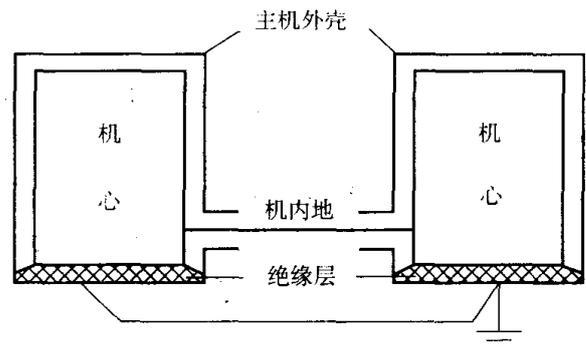


图 3-29 外壳接地机心浮空

(3) 多机系统的接地。在计算机网络系统中, 多台计算机之间相互通信, 资源共享。如果接地不合理, 将使整个网络系统无法正常工作。近距离的几台计算机安装在同一机房内, 可采用类似图 3-28 所示的全机一点接地方法。对于远距离的计算机网络, 多台计算机之间的数据通信, 通过隔离的办法把地分开, 例如, 采用变压器隔离技术、光隔离技术或无线通信技术。

3.4 输入、输出通道信号隔离技术

在过程计算机控制系统中, 输入、输出通道由于直接与工业生产现场相连, 所处环境常十分恶劣, 相应地对抗干扰方面的要求也很高, 除了前面介绍的采用屏蔽、阻抗匹配、滤波等技术外, 最重要的就是要对输入、输出信号进行隔离。无论是变压器隔离, 还是光电隔离, 隔离的核心都是要切断不同“地”之间的电信号联系, 主要起到抗共模干扰的作用。除了信号线是为避免“共地”需隔离外, 供电电源也是可能引起“共地”现象的原因之一, 因此隔离器的输入、输出端应当采用相互隔离的电源。下面针对数字信号和模拟信号输入、输出的不同情况下的隔离办法分别进行讨论。

3.4.1 数字信号输入、输出通道的隔离

数字信号隔离一般采用光电隔离的方法，与输入和输出通道隔离的原理类似，都是利用光耦合器的开关特性（即光敏晶体管工作在截止区和饱和区）来传输数字信号。

光耦合隔离器按其输出级不同可分为晶体管型、单向晶闸管型、双向晶闸管型等几种，如图3-30所示。它们的原理是相同的，即都是通过电、光、电这种信号转换，利用光信号的传送不受电磁场的干扰而完成隔离功能的。要注意的是，用于驱动发光管的电源与驱动光敏管的电源不应是共地的同一个电源，必须分开单独供电，才能有效避免输出端与输入端相互间的反馈和干扰。

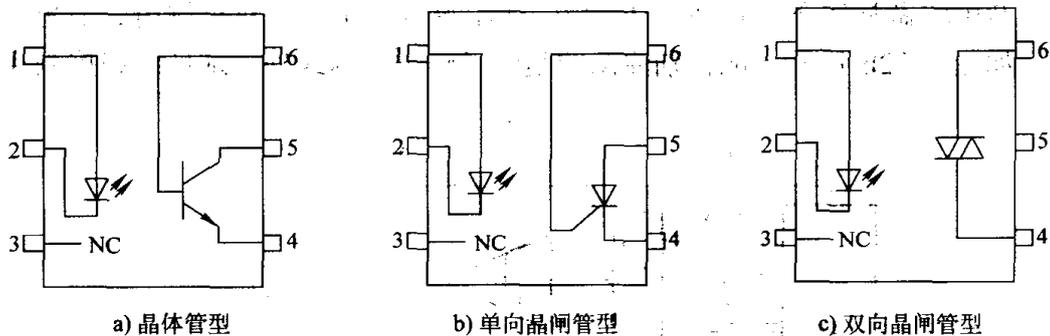


图 3-30 光耦合隔离器的几种类别

为了适应计算机控制系统的要求，目前已生产出各种集成的多路光耦合隔离器，如 TLP 系列就是常用的一种。典型的光耦合隔离电路有数字量同相传输与数字量反相传输两种，如图3-31所示。

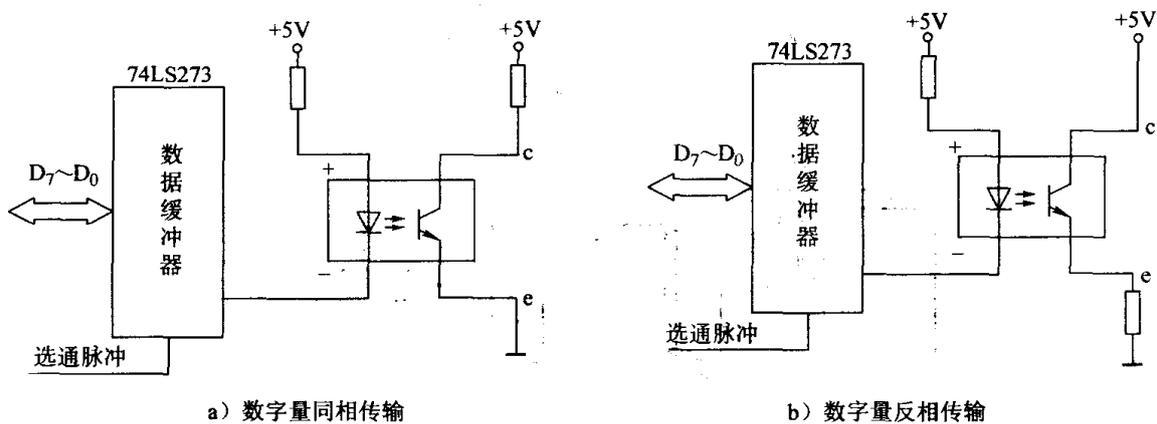


图 3-31 光电隔离耦合电路

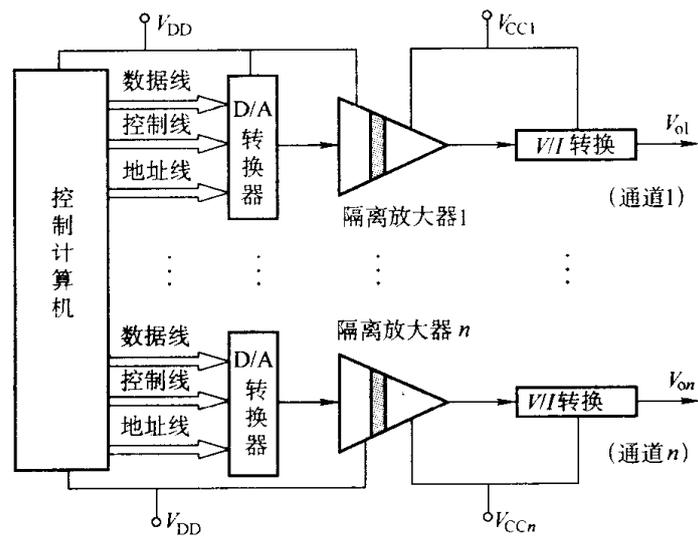
数字量同相传输如图3-31a所示，光耦合器的输入正端接正电源，输入负端接到与数据总线相连的数据缓冲器上，光耦合器的集电极c端通过电阻接另一个正电源，发射极e端直接接地，光耦合器输出端即从集电极c端引出，当数据线为低电平“0”时，发光管导通且发光，使得光敏管导通，输出端c接地而获得低电平“0”；当数据线为高电平“1”时，发光管截止不发光，则光敏管也截止使输出端c从电源处获得高电平“1”。如此，便完成了数字信号的同相传输。

数字量反相传输如图 3-31b 所示,与图 3-31a 不同的是光耦合器的集电极 c 端直接接另一个正电源,而发射极 e 端通过电阻接地,则光耦合器输出端从发射极 e 端引出。从而完成了数字信号的反相传输。

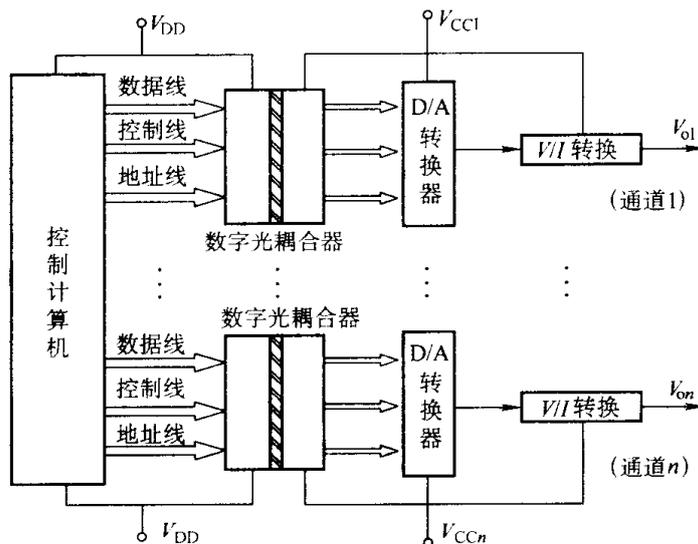
需要特别说明的是,某些数字量的输入、输出器件本身的输入、输出端就是隔离的,应用时可以不再进行隔离处理,只需注意输入、输出端应分别隔离供电,如固态继电器等。

3.4.2 模拟量输出通道的隔离

模拟量输出通道的隔离主要有数字隔离和模拟隔离两种形式,如图 3-32 所示。一种是在 D/A 转换器后的现场侧直接隔离模拟信号,如图 3-32a,隔离模拟信号一般是使用隔离放大器(在下一小节中将做具体介绍)。另一种是在 D/A 转换器前的计算机侧用数字光耦合器隔离数字信号,如图 3-32b 所示,应特别注意的是除了隔离数据线外,地址线和控制线同样也需要隔离。



a) 模拟隔离方式



b) 数字隔离方式

图 3-32 模拟量输出通道隔离的原理框图

这两种隔离方法各有优缺点。模拟信号隔离方法的优点是使用器件少，缺点是为保证 D/A 转换的精度和线性度需较高的成本；数字信号隔离方法的优点是调试简单，不影响系统的精度和线性度，缺点是使用光耦合器较多。但因光耦合器价格越来越低廉，数字信号隔离方法的优势突现出来，因而在工程中模拟量输出通道多使用数字隔离方式。模拟量输出通道用数字隔离方式的另一个原因是 D/A 转换一般采用一路输出对应一个 D/A 转换器，这样在有多路输出的情况下，只要供电电源是隔离的，数字隔离就可以实现 D/A 通道间与计算机间的完全隔离。如图 3-32b 所示的多路输出情况，为了使用单一电源供电，各个供电电源除了可以分别由相连的现场供电外，也可以通过 DC-DC 变换器实现。

3.4.3 模拟量输入通道的隔离

与模拟量输出通道的隔离类似，模拟量输入通道的隔离也可以采用数字隔离和模拟隔离两种不同的形式。一般地，在过程计算机控制中都有多个输入通道，而且常来自于生产现场的不同位置，这些信号之间的“地”也不同，因此输入系统的隔离除了需将控制计算机与现场侧隔离外，还需要对多路输入信号之间进行隔离，以达到抗共模干扰的目的，这也称为路间隔离。考虑到模拟量输入通道多数采用共用一个 A/D 转换器的形式，为了实现路间隔离，一般采用模拟隔离的方式，如图 3-33 所示。

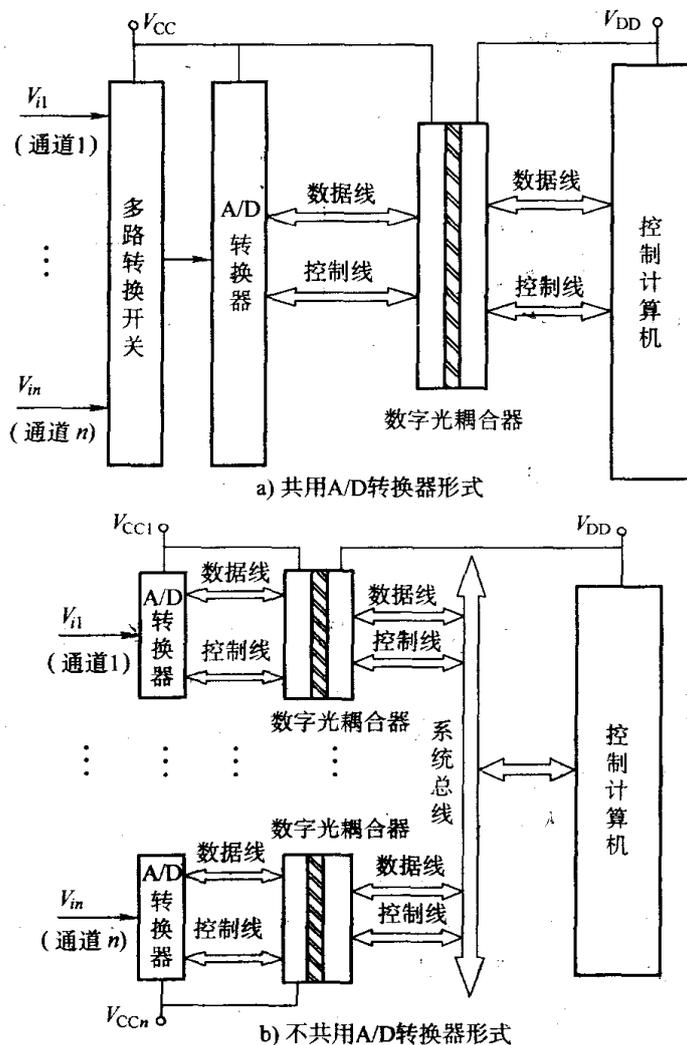


图 3-33 模拟量输入通道的数字隔离

与模拟量输出通道的数字隔离一样，A/D 通道的数字隔离一般也是使用数字光耦合器隔离计算机侧信号。这里有两种主要形式：图 3-33a 为共用一个 A/D 转换器的形式，由于多路转换开关将各输入通道的电信号连在了一起，因此无法实现路间隔离，抗共模干扰能力差，适合于各输入信号可共地的情形（如信号来自生产现场同一位置且由同一电源供电）或各输入信号引入前已进行了隔离处理。由于这种数字隔离方式不能有效抑制共模干扰，更多是起到保护控制计算机的目的，也称为“伪”隔离方式。图 3-33b 为不共用 A/D 转换器的形式，由于没有多路转换开关，只要各通道的供电电源是隔离的，就可以将计算机与输入通道以及各输入通道之间完全隔离开来，满足路间隔离的要求。随着 A/D 转换器价格的降低，这种使用多个 A/D 转换器进行数字隔离实现多路信号完全隔离的方式也越来越多。

A/D 通道隔离最常用的方式是使用隔离放大器进行模拟信号的隔离，如图 3-34 所示。这种隔离方式的优点是电路简单，共用 A/D 转换器也可方便地实现路间隔离；缺点是隔离放大器的价格比较高。注意，图中各路通道供电电源 V_{CCi} 间也应是隔离的，常采用 DC-DC 变换器；此外有些隔离放大器本身可以提供隔离电源，采用单一电源即可。

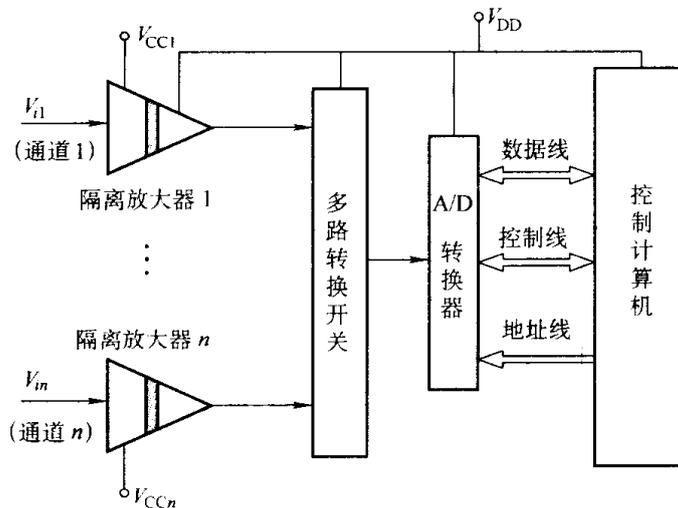


图 3-34 模拟量输入通道的模拟信号隔离

3.4.4 常用隔离放大器

模拟信号的隔离常使用隔离放大器，这是因为隔离放大器的线性和稳定性好，隔离电压和共模抑制比高，应用电路简单，频带较宽等。根据不同的耦合方式，隔离放大器可以分为变压器耦合隔离放大器和光耦合隔离放大器。下面介绍几种常用的隔离放大器。

1. AD202/AD204 隔离放大器

AD202/AD204 是一种低成本的变压器耦合隔离放大器。变压器耦合隔离放大器通常是由滤波、放大、增益调节、调制、隔离变压器、解调、滤波及输出级（或带增益调节）等部分组成，如图 3-35 所示。调制器的作用是把直流或低频信号 V_i 调制成较高频率（如 25kHz 或 200kHz）的交流信号，此交流信号幅值和相位反映了输入信号 V_i 的幅值和极性，经过高频变压器隔离，并经解调器把交流信号恢复成直流或低频信号 V_o ，而且幅值和极性取决于调制信号的幅值和相位。经过隔离放大器的信号 V_o 是输入信号 V_i 的 K 倍，即 $V_o = KV_i$ ， K 的大小取

决于增益调节电阻，这里需要指出的是，有的增益调节在输入侧，也有的增益调节在输出侧。

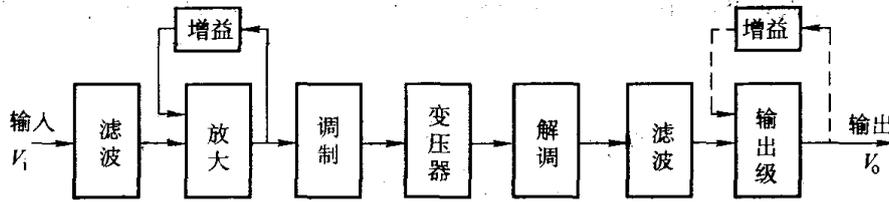


图 3-35 变压器耦合隔离放大器原理框图

AD202/AD204 的主要特点如下：

- 1) 高精度，最大非线性度为 $\pm 0.025\%$ ；
- 2) 高共模抑制比 (CMRR) 为 130dB；
- 3) 低功耗，75mW (AD202) 及 35mW (AD204)；
- 4) 宽频带，可达 5kHz (AD204)；
- 5) 高绝缘级别，CMV 为连续 $\pm 2000V$ 。

AD202 与 AD204 两者的区别是，AD202 直接由 +15V 直流电压供电，而 AD204 则是通过外部提供时钟（如 AD246）供电。

AD202/AD204 有两种封装形式。表 3-1 为两种封装的引脚说明，其功能框图如图 3-36 所示。由图可以看出，其输入、输出和电源 3 个部分都是相互隔离的，并通过低泄漏的变压器耦合在一起。其能量通道是，+15V 直流电源经直流/交流变换器变成交流电压后，一路作为解调器 (demod) 的电源电压，另一路经隔离变压器耦合到二次侧作为调制器 (modem) 的开关电源或经过整流滤波器恢复成直流电压电源输出。信号通道是，输入信号经放大器 A_1 放大后由调制器变为交流，通过隔离变压器耦合至解调器，恢复成原信号输出。采用这种方法可以提高微电流和低频信号的测量精度，减小漂移。

表 3-1 AD202/AD204 引脚功能

SIP 封装引脚	DIP 封装引脚	功能
1	1	正输入
2	2	输入/ V_{iso} 公共
3	3	负输入
4	38	输入反馈
5	37	$-V_{iso}$ 输出
6	36	$+V_{iso}$ 输出
31	20	+15V 输入口 (仅 AD202)
32	22	时钟/电源公共
33	21	时钟输入 (仅 AD204)
37	18	输出 LO
38	19	输出 HI

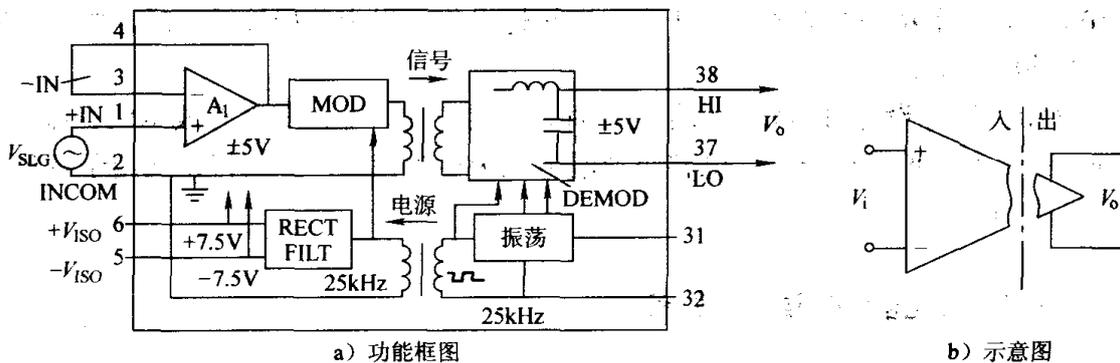


图 3-36 AD202/AD204 隔离放大器

图 3-37 为应用 AD202/AD204 构成的小信号放大隔离电路。该电路供电电流应足够大，这样才便于调制。其失调偏置由 OP-07 调整引脚来调节，增益则由反馈电阻 R_G 决定。

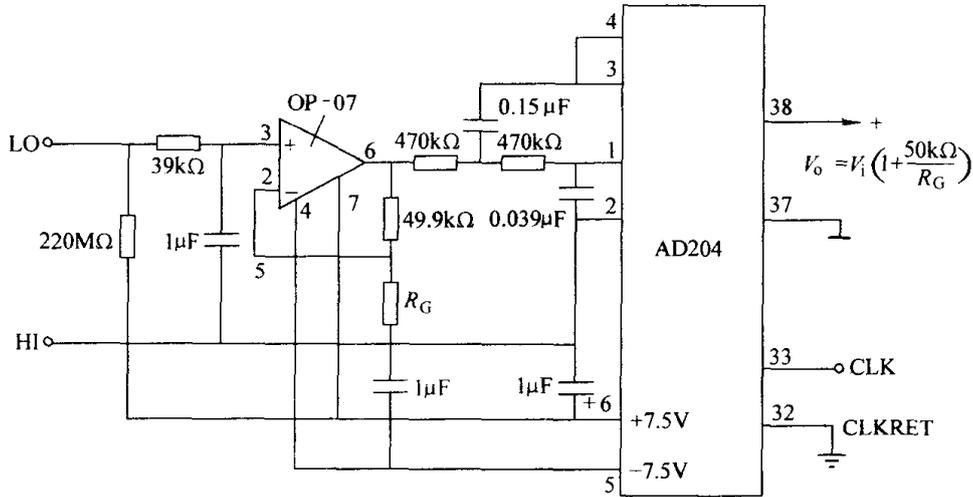


图 3-37 传感器信号输入放大隔离电路

2. ISO100 隔离放大器

ISO100 是一种小型廉价的光耦合隔离放大器，图 3-38 为其简化电路图。其工作电源为 $\pm 18V$ ，隔离电压达 2500V，输入电流最大为 $\pm 1mA$ ，频带宽达 60kHz，为 18 脚 DIP 封装。

由 3-38 可见，ISO100 是利用一个发光二极管和两个光敏二极管耦合，使输入信号与输出信号隔离开来。这样可将发光二极管的光反相送回输入端（负反馈），正相送至输出端，从而提高了放大器的精度、线性度和时间温度稳定性。其输入为电流信号，若进行电压输入，则利用一外接电阻即可实现，此时 $I_i = V_i / R_i$ 。

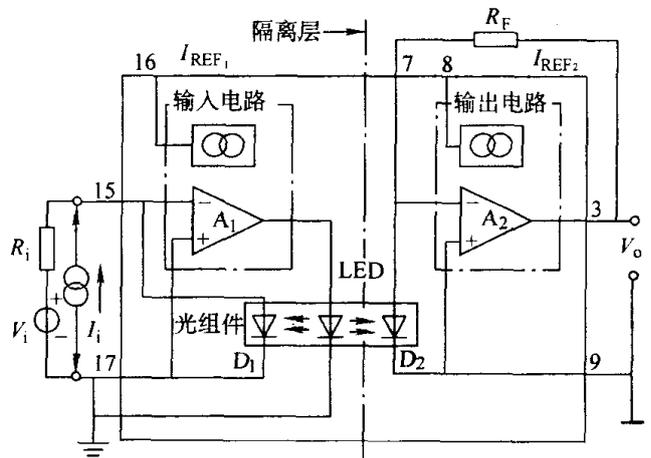


图 3-38 ISO100 隔离放大器简化电路

图 3-38 中 A_1 为单位增益放大器， A_2 为电流电压转换器，即在系统稳定时， $I_{D1} = I_{D2} \approx -I_i$ ，则输出 $V_o = -I_{D2} R_F = I_i R_F$ 。可见，只要改变外接电阻 R_F 的值，就能改变增益。

ISO100 中有两个精密电流源，用于完成双极性操作。当为单极性操作时，则不需精密电流源，可供外部使用。

利用 ISO100 进行设计只需外接少量元器件，非常方便，加之其具有成本低、体积小、隔离效果好、频带宽、偏移电压低、漂移小及漏电流极小等特点，因而特别适用于各种模拟输入通道的隔离和放大。

3.5 控制计算机的可靠性技术

控制计算机的可靠性技术包含很多方面的内容，其中应用最广泛的是故障自诊断技术、

冗余技术和 CPU 抗干扰技术。

3.5.1 故障自诊断技术

故障自诊断技术是用软件的办法迅速、准确地确定系统内部是否发生故障，以及故障发生的部位，指导运行维护人员及时发现故障、及时维修。利用计算机定时执行自诊断程序实现软、硬件故障部位诊断，通常有如下一些做法。

1. CPU 运算功能检查

在特定的存储区存储一组确定的数据，其中一个数据是其余数据经过作某些运算的结果。在诊断时，把参加运算的数据按预定的运算规律（如按位相加或异或等）进行运算，将运算结果与原来存储结果进行比较，如有差错就输出报警信号。

2. RAM 检查

自诊断程序定时向 RAM 中各区域分别写入一个随机数，然后又读出来与原写入数据进行比较，检查它们是否仍然一致，将该数据求反再进行一次存、取、比较的检查，若有错误就输出报警信号。

3. 输入、输出通道的检查

在设计系统的输入、输出通道时，通常给每一通道模板留下一对输入、输出通道作为检查通道，并将输出通道的输出端与输入通道的输入端连接起来。诊断时，计算机向检查通道的输出通道输出一个随机数，再从输入通道读取回来，然后对输出和输入数作比较，正常情况下对于数字量通道，这两个数应完全一致。对于模拟量通道，这两个数的误差应在精度允许的范围之内，否则就输出通道错的报警信号。

4. 控制软件及寄存器检查

编制一个软件和寄存器检查程序，对计算机内部应用程序逐条指令、逐个寄存器进行检查，发现错误，立即停机并显示故障点。

5. 设置监视定时器

对分散控制系统，给每个直接控制器设置一个监视定时器。正常工作时，由它定时向上位机发出一个脉冲，当某直接控制器的 CPU 出现故障或断电，便会停止发出定时脉冲，即上位机通过检查各下位机的“脉搏”，来判断它的工作正常与否。一旦发现某一控制器的“脉搏”停止，上位机立即发出报警信号。

6. 数据的有效性检查

输出、输入通道采集的数据及运算结果的数据，总是在一个有限的范围内，通过检查这些数据是否超限，便可以判断相关部分的硬件是否出现故障。

7. 设定软件模块出入口标志

当程序执行某一模块时，便在某存储单元存入该模块的标志，该模块执行完毕，通过读取该模块的出口标志并与原存入的入口标志比较，若不相符，说明程序执行过程中由于某些故障引起了出错。

8. 程序存储区的写保护

程序的存储区在正常情况下是只允许读不允许写，如果发生向这部分存储单元写入的操作，就说明出现了不正常的运行状态。所以在编制用户程序时，凡遇到对存储器的写操作，可先检查所写单元地址是否属于合法的地址，只有正确无误后，才能进行写操作，否则报警

停机。

3.5.2 冗余技术

随着大规模集成电子技术的成熟和硬件设备生产制作工艺技术的不断提高,使硬件质量、性能和工作可靠性都得到了充分的保证,在成本大大下降的前提下,为使系统有更高的运行可靠性,可进一步采用容错技术,使系统即便出现故障,仍然能维持正常的运行。冗余是容错技术的关键措施,所谓冗余是指系统满足一定功能要求下配置的设备超过系统实际需要的最低数量,如采用双 CPU 或双主机就是一种冗余。

冗余主要有两种方式,一是工作冗余,另一是后备冗余。工作冗余是对关键设备实行双重或三重的原则来重复配置,这些设备同时处于运行状态,工作过程中如果某一台出现故障,它便会自动脱离系统,并不影响系统的正常工作。后备冗余也用于关键设备,这种方式是使一台设备投入运行,另一台冗余设备处于热备用状态,但不投入运行,在线运行的设备一旦出现故障,后备的设备立即投入运行。如果系统需要多台同样的设备运行,为节省投资,可采用多台运行,一台备用的办法解决。故障设备的切除与后备设备的投入,其换接过程应当完全是自动完成的。

3.5.3 CPU 抗干扰技术

对于那些可能穿过过程输入、输出通道而进入 CPU 的干扰,可采取指令冗余、软件陷阱以及程序运行监视等措施来使 CPU 恢复正常工作。

1. 指令冗余技术

当计算机系统受到外界干扰时,破坏了 CPU 正常的工作时序,有可能造成程序计数器 PC 的值发生改变,跳转到随机的程序存储区。当程序跑飞到某一单字节指令上,程序便自动纳入正轨;当程序跑飞到某一双字节指令上,有可能落到其操作数上,则 CPU 会误将操作数当操作码执行;当程序跑飞到三字节指令上,因它有两个操作数,出错的概率会更大。

为了解决这一问题,可采用在程序中人为地插入一些空操作指令 NOP 或将有效的单字节指令重复书写,此即指令冗余技术。由于空操作指令为单字节指令,且对计算机的工作状态无任何影响,这样就会使失控的程序在遇到该指令后,能够调整其 PC 值至正确轨道,使后续指令得以正确执行。

通常不能在程序中加入太多的冗余指令,以免降低程序正常运行的效率。一般是在对程序流向起决定作用的指令之前以及影响系统工作状态的重要指令之前都应插入两、三条 NOP 指令,还可以每隔一定数目的指令插入 NOP 指令,以保证跑飞的程序迅速纳入正确轨道。

指令冗余技术可以减少程序出现错误跳转的次数,但不能保证在失控期间不干坏事,更不能保证程序纳入正常轨道后就太平无事了。解决这个问题还必须采用软件容错技术,使系统的误动作减少,并消除重大误动作。

2. 软件陷阱技术

指令冗余使跑飞的程序安定下来是有条件的,首先应使跑飞的程序必须落到程序区,其次必须执行到冗余指令。当跑飞的程序落到非程序区(如 EPROM 中未使用的空间、程序中的数据表格区)时,对此情况采取的措施就是设立软件陷阱。

软件陷阱,就是在非程序区设置拦截措施,使程序进入陷阱,即通过一条引导指令,强

行将跑飞的程序引向一个指定的地址，在那里有一段专门对程序出错进行处理的程序。如果把这段程序的入口标称为 ERROR 的话，软件陷阱即为一条“JMP ERROR”指令。为加强其捕捉效果，一般还在它前面加上两条 NOP 指令，因此真正的软件陷阱是由 3 条指令构成，即

```

NOP
NOP
JMP ERROR

```

软件陷阱安排在以下四种位置：未使用的中断向量区、未使用的大片 ROM 空间、程序中的数据表格区以及程序区中一些指令串中间的断裂点处。

由于软件陷阱都安排在正常程序执行不到的地方，故在不影响程序的执行效率，并且在当前 EPROM 容量不成问题的条件下，应多安插软件陷阱指令。

3. 程序运行监视技术

工业现场有时会出现瞬间的尖峰高能脉冲干扰，它可能会长驱直入作用到 CPU 芯片上，使正在执行的程序跑飞到一个临时构成的死循环中，这时指令冗余和软件陷阱技术也无能为力，系统将完全瘫痪。此时必须强制系统复位，摆脱死循环。由于操作者不可能一直监视系统，这就需要有一个独立于 CPU 之外的监视系统，在程序陷入死循环时，能及时发现并自动复位系统，这就是具有看守大门作用的程序运行监视系统，国外称为“WatchdogTimer”，即看门狗定时器或看门狗。

为了保证程序运行监视系统的可靠性，监视系统中必须包括一定的硬件部分，且应完全独立于 CPU 之外，但又要与 CPU 保持时刻的联系。因此，程序运行监视系统是硬件电路与软件程序的巧妙结合。图 3-39 给出了 WatchdogTimer 的工作原理。

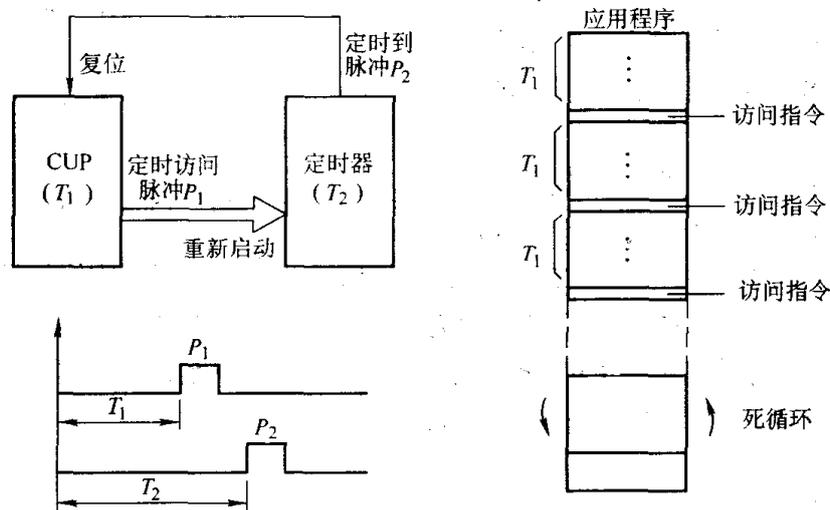


图 3-39 WatchdogTimer 工作原理

CPU 可设计成由程序确定的定时器 1，看门狗被设计成另一个定时器 2，它的计时启动将因 CPU 的定时访问脉冲 P_1 的到来而重新开始，定时器 2 的定时到脉冲 P_2 连到 CPU 的复位端。两个定时周期必须是 $T_1 < T_2$ ， T_1 就是 CPU 定时访问定时器 2 的周期，也就是在 CPU 执行的应用程序中每隔 T_1 时间安插一条访问指令。

在正常情况下, CPU 每隔 T_1 时间便会定时访问定时器 2, 从而使定时器 2 重新开始计时而不会产生溢出脉冲 P_2 ; 而一旦 CPU 受到干扰陷入死循环, 便不能及时访问定时器 2, 那么定时器 2 会在 T_2 时间到达时产生定时溢出脉冲 P_2 , 从而引起 CPU 的复位, 自动恢复系统的正常运行程序。

以前的 WatchdogTimer 硬件部分是用单稳电路或自带脉冲源的计数器构成, 一是电路有些复杂; 二是可靠性有些问题。美国 Xicor 公司生产的 X5045 芯片, 集看门狗功能、电源监测、 E^2 PROM 及上电复位等 4 个功能为一体, 使用该器件将大大简化系统的结构并提高系统的性能。

X5045 只有 8 根引脚, 其定义如下:

SCK——串行时钟;

SO——串行输出, 时钟 SCK 的下降沿同步输出数据;

SI——串行输入, 时钟 SCK 的上升沿锁存数据;

\overline{CS} ——片选信号, 低电平时 X5045 工作, 变为高电平时将使看门狗定时器重新开始计时;

\overline{WP} ——写保护, 低电平时写操作被禁止, 高电平时所有功能正常;

RESET——复位, 高电平有效。用于电源检测和看门狗超时输出;

V_{SS} ——地;

V_{CC} ——电源电压。

X5045 与 CPU 的接口电路如图 3-40 所示, X5045 的信号线 SO、SI、SCK、 \overline{CS} 与 CPU 的数据线 $D_0 \sim D_3$ 相连, 用软件控制引脚的读 (SO)、写 (SI) 及选通 (\overline{CS})。X5045 的引脚 RESET 与 CPU 的复位端 RESET 相连, 利用访问程序造成 \overline{CS} 引脚上的信号变化, 就算访问了一次 X5045。

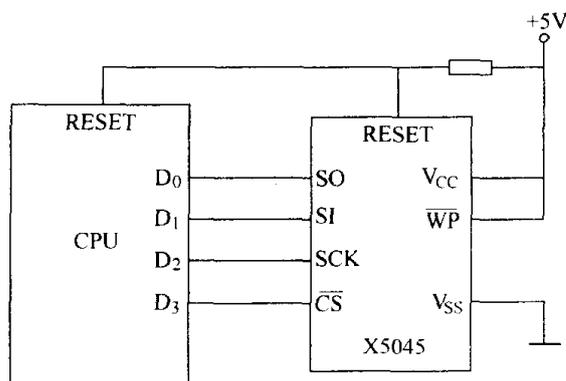


图 3-40 X5045 与 CPU 的接口电路

在 CPU 正常工作时, 每隔一定时间 (小于 X5045 的定时时间) 运行一次该访问程序, X5045 就不会产生溢出脉冲。一旦 CPU 陷入死循环, 便不再执行该程序也即不对 X5045 进行访问, 则 X5045 就会在 RESET 端输出宽度为 100~400ms 的正脉冲, 足以使 CPU 复位。

这里, X5045 中的看门狗对 CPU 提供了完全独立的保护系统, 并提供了 3 种定时时间: 200ms、600ms 和 1.4s, 可用于在编程时进行选择。

3.6 数字滤波技术

由于工业生产的现场环境非常恶劣, 各种干扰源很多, 计算机通过输入通道采集到的数据信号, 虽经硬件电路的滤波处理, 但仍会混有干扰噪声。因此, 为了提高系统性能, 达到准确地测量与控制的目的, 一般还需进行数字滤波处理。

数字滤波, 就是计算机系统对输入信号采样多次, 然后用某种计算方法对采样值进行数字处理, 以削弱或滤除干扰噪声从而获得真实信号的过程。这种滤波方法不需要硬件设备,

只需根据预定的滤波算法编制相应的程序即可达到目的,实质上是一种程序滤波。

数字滤波由于不涉及硬件设备,因此可靠性高,稳定性好,其滤波参数的修改也容易,便于灵活使用,而且一种滤波子程序可以被多个通道所共用,因而成本很低。另外,数字滤波还可以对各种干扰信号,甚至极低频率的信号进行滤波;其不足之处是占用 CPU 的机时。

总之,数字滤波与硬件滤波相比优点甚多,因此得到了普遍的应用。常用的数字滤波方法有平均值滤波、中值滤波、限幅滤波和惯性滤波等。

3.6.1 平均值滤波

平均值滤波就是对多个采样值进行平均的算法,这是消除随机误差最常用的方法,具体又可分为以下几种。

1. 算术平均滤波

算术平均滤波是在采样周期 T 内,对测量信号 y 进行 m 次采样,把 m 个采样值的算术平均值作为本次的有效采样值,即

$$\bar{y}(k) = \frac{1}{m} \sum_{i=1}^m y_i \quad (3-11)$$

这里,采样次数 m 决定了信号的平滑度和灵敏度。提高 m 值,可提高平滑度,但相应的灵敏度也随之降低, m 值大小应视具体情况选取。一般情况下,流量信号可取 8 左右,压力信号可取 4 左右,温度、成分等缓变信号可取 2 甚至不平均。在编制算法程序时, m 一般取 2、4、8 等 2 的整数幂,以使用移位来代替除法求得平均值。

2. 去极值平均滤波

算术平均滤波对周期性干扰信号有良好的抑制作用,但对工业场合经常遇到的尖脉冲干扰也进行平均计算,显然会造成较大偏差。为此,可采取先剔除脉冲干扰再平均计算的去极值平均滤波法,即对连续采样的 m 个数据进行比较,去掉其中的最大值与最小值,然后计算余下的 $m-2$ 个数据的算术平均值。

$$\bar{y}(k) = \frac{1}{m-2} \left(\sum_{i=1}^m y_i - y_{\max} - y_{\min} \right) \quad (3-12)$$

在编制算法程序时,为了便于用移位来代替除法求得平均值, m 一般取 4、6、10 等。

如果把连续 m 次采样值按大小进行排序,去掉其中最大的和最小的几个采样值,仅保留中间一个或几个采样值,再进行平均作为当前的采样结果,可能会获得更好滤除尖峰等随机干扰的效果。这种方法可看作去极值平均滤波的扩展形式,也称中值滤波。

3. 加权平均滤波

上述两种平均滤波方法都存在平滑度和灵敏度的矛盾。采样次数太少则平均效果差,次数太多则灵敏度下降。为协调两者关系,提高灵敏度,可采用加权平均滤波法,即

$$\bar{y}(k) = \sum_{i=1}^m C_i y_i \quad (3-13)$$

式中,第 m 次为最新采样值; C_i 为加权系数,且满足

$$\begin{aligned} C_1 + C_2 + \cdots + C_m &= 1 \\ C_m &> C_{m-1} > \cdots > C_1 > 0 \end{aligned}$$

C_1, C_2, \dots, C_m 的取值应视具体情况选取, 并通过调试确定。

4. 滑动平均滤波

前三种平均滤波算法有一个共同点, 即每取得一个有效采样值, 必须连续进行若干次采样, 当系统的采样速度较慢 (如双积分型 A/D 转换) 或采样信号变化较快时, 系统的实时性就无法得到保证。而滑动平均滤波算法能较好地解决这一问题。它在每个采样周期只采样一次, 将这一次采样值和过去的若干次采样值一起求平均, 所得结果即为有效采样值。

具体做法是: 可由循环队列结构方式来实现数据的存放, 比如取 m 个采样值求滑动平均, 只要在 RAM 中设置 m 个数据暂存区, 每次新采集一个数据便存入暂存区的队尾, 同时冲掉队首的一个数据, 这样在存储器队列中始终保持有 m 个最新的数据。

3.6.2 限幅滤波与限速滤波

由于大的随机干扰或采样器的不稳定, 可能使得采样数据偏离实际值太远, 为此, 可以通过限幅的办法进行滤波处理, 上、下限值的确定可根据工程实际量的可能变化范围求取。限幅滤波的具体方法为

$$\begin{cases} y(k) = y_H & (y(k) \geq y_H) \\ y(k) = y_L & (y(k) \leq y_L) \\ y(k) = y(k) & (y_L < y(k) < y_H) \end{cases} \quad (3-14)$$

限幅滤波可以剔除明显不合理的数据, 但剔除后只能用上、下限值替代, 有时效果不太好, 更好的办法是采用限速滤波。

限速滤波就是限制变化率, 把两次相邻的采样值相减, 求其增量的绝对值, 再与两次采样所允许的最大差值 Δy 进行比较, 如果不大于 Δy , 表示本次采样值 $y(k)$ 是真实的 (因为在其正常变化范围内), 则取 $y(k)$ 为有效采样值; 反之, $y(k)$ 是不真实的, 则取上次采样值 $y(k-1)$ 作为本次有效采样值, 即

$$\begin{cases} |y(k) - y(k-1)| \leq \Delta y, & y(k) = y(k) \\ |y(k) - y(k-1)| > \Delta y, & y(k) = y(k-1) \end{cases} \quad (3-15)$$

式中, $y(k)$ 为当前时刻的采样值; $y(k-1)$ 为前一时刻的采样值; Δy 为相邻两次采样值所允许的最大偏差, 其大小取决于采样周期 T 和信号 y 的正常变化率。

3.6.3 惯性滤波

惯性滤波是模拟硬件 RC 低通滤波器的数字实现。常用的 RC 低通滤波器的传递函数为

$$\frac{Y(s)}{X(s)} = \frac{1}{1 + T_f s} \quad (3-16)$$

式中, $T_f = RC$ 是滤波器的滤波时间常数, 其大小直接关系到滤波效果。

一般来说, T_f 越大, 则滤波器的截止频率越低, 滤出的电压纹波较小, 但输出滞后较大。由于大的时间常数及高精度的 RC 电路不易制作, 所以硬件 RC 滤波器不可能对极低频率的信号进行滤波。为此, 可以模仿硬件 RC 滤波器的特性参数, 用软件实现式 (3-16) 的低通数字滤波器, 亦称一阶惯性数字滤波器。

将式(3-16)写成差分方程,即

$$T_f \frac{y(k) - y(k-1)}{T} + y(k) = x(k) \quad (3-17)$$

整理得

$$y(k) = \frac{T}{T_f + T} x(k) + \frac{T_f}{T_f + T} y(k-1) = \alpha x(k) + (1 - \alpha) y(k-1) \quad (3-18)$$

式中, α 为滤波系数, $\alpha = T/(T_f + T)$, 且 $0 < \alpha < 1$; T 为采样周期。

一般地, 采样周期 T 远小于惯性时间 T_f , 则 $\alpha \ll 1$, 表明本次有效采样值主要取决于以前的有效采样值, 而本次采样值仅起到一些修正作用。当采样周期 T 足够小时, $\alpha \approx T/T_f$, 滤波算法的截止频率为

$$f = \frac{1}{2\pi RC} = \frac{\alpha}{2\pi T} \quad (3-19)$$

当采样周期一定时, 滤波系数 α 越小, 数字滤波器的截止频率就越低。例如, 当 $T = 0.5s$ (即每秒采样 2 次), $\alpha = 1/32$ 时, 有

$$f = (1/32)(2 \times 3.14 \times 0.5) \approx 0.01\text{Hz}$$

这对于变化缓慢的采样信号(如大型储水池的水位信号), 其滤波效果是很好的。

数字滤波器的程序编制可按式(3-18)进行。设计时, 应根据采样周期与截止频率适当选取 α 值, 使得滤波器的输出既无明显纹波, 又不太滞后。显然, 该算法比较简单, 比起平均值滤波法要快, 能很好地消除周期性干扰和较宽频率的随机干扰信号。

以上讨论的几种数字滤波方法, 各有特点。一般情况下, 算术平均滤波适用于周期性干扰; 加权平均滤波和滑动平均滤波的实时性更好; 限幅/限速滤波适用于偶然的脉冲干扰; 去极值平均滤波既可滤去脉冲干扰又可滤去小的随机干扰; 惯性滤波能很好地消除周期性干扰和较宽频率的随机干扰信号。在实际应用中, 究竟采用不采用以及采用哪一种数字滤波, 都应视具体情况而定。可能有的系统并不需要进行数字滤波, 如果应用得不恰当, 不但达不到滤波效果, 还会降低控制品质; 而有的系统采用复合滤波方法, 即把几种滤波方法结合起来使用, 可能会取得更好的滤波效果。

思考题

1. 简述干扰的来源与传播途径。
2. 串模干扰的成因是什么? 如何抑制串模干扰?
3. 共模干扰的成因是什么? 如何抑制共模干扰?
4. 过程输入、输出通道的隔离主要有哪几种形式? 各有什么特点?
5. 在过程计算机控制系统中, 主要有哪几种地? 最终如何接地?
6. 结合图 3-40, 简述程序运行监视系统的工作原理。
7. 简述数字滤波的特点。
8. 简述各种数字滤波方法的原理或算法及适用场合。

第 4 章 过程输入、输出通道

在过程计算机控制系统中，为了实现对生产过程的控制，要将对象的被控参数及运行状态，按要求的方式送入计算机，同时计算机经过计算、处理后，将结果以数字量的形式输出，此时也要将数字量变换为适合生产过程控制的量。因此在计算机和生产过程之间，必须设置信息的传递和变换装置，这个装置就称为过程输入、输出通道，也叫 I/O 通道。换句话说，过程输入、输出通道是计算机和工业生产过程相互交换信息的桥梁。

根据过程信息的性质及传递方向，过程输入、输出通道包括模拟量输入通道（A/D 通道）、模拟量输出通道（D/A 通道）、数字量（开关量）输入通道（DI 通道）和数字量（开关量）输出通道（DO 通道）。

4.1 过程输入、输出通道的控制方式

4.1.1 过程输入、输出通道与 CPU 交换的信息类型

过程输入、输出通道与 CPU 交换的信息类型有以下三种。

(1) 数据信息：反映生产现场的参数及状态的信息，它包括数字量、开关量和模拟量。

(2) 状态信息：又叫应答信息、握手信息，它反映过程通道的状态，如准备就绪信号等。

(3) 控制信息：用来控制过程通道的启动和停止等信息，如三态门的打开和关闭、触发器的启动等。

在过程输入、输出通道中，必须设置一个与 CPU 联系的接口电路，用于传送数据信息、状态信息和控制信息。

4.1.2 过程通道的编址方式

由于计算机控制系统一般都有多个过程输入、输出通道，因此需对每个输入、输出通道安排地址。过程通道编址方式主要有两种。

1. 过程通道与存储器统一编址方式

这种编址方式又称存储器映像方式，它从存储器空间划出一部分地址空间给过程通道，把过程通道的端口当作存储单元一样进行访问，对 I/O 端口进行输入、输出操作跟对存储单元进行读、写操作方式相同，只是地址不同。所有访问内存的指令同样都可用于访问 I/O 端口。采用这种方式的 CPU 主要有 Intel 8031 和 Intel 80196 系列单片机等。统一编址的最大优点是无需专门的 I/O 指令，这样便简化了指令系统的设计，并能省去相应的 I/O 操作的对外引线，而且 CPU 可直接对 I/O 数据进行算术和逻辑运算，且指令丰富。统一编址的不足之处在于 I/O 端口地址占用了一部分存储器空间；另外，访问内存的指令长度一般比专用的 I/O 指令长，因而取指令周期较长，又多占了指令字节。

2. 过程通道与存储器独立编址方式

这种编址方式将过程通道的端口地址单独编址，有自己独立的过程通道地址空间，而不是占用存储器地址空间。在过程通道地址空间中，每一个过程通道的端口有一个唯一对应的过程通道的端口地址。这种独立编址方式要求有专用的 I/O 指令（IN 及 OUT 指令）用于 CPU 与过程通道端口之间的数据传输。地址总线配合存储器操作信号实现存储器的访问控制，而地址总线与 I/O 操作信号配合则可访问过程通道。因此，实现这种编址方式的 CPU 应分别有存储器访问和 I/O 访问的指令及相应的控制信号，典型的微处理器 Z80 和 80X86 具有这种功能。

4.1.3 CPU 对过程通道的控制方式

计算机的外围设备及过程通道种类繁多，它们的传送速率又不相同，因此输入、输出将产生复杂的定时问题，也就是 CPU 采用什么控制方式向过程通道输入和输出数据。常用的控制方式主要有三种：程序查询方式、中断控制方式和直接存储器存取（DMA）方式。

1. 程序查询方式

采用程序查询方式时，数据在 CPU 和输入、输出通道之间的传送采用程序进行控制。传送前必须查询 I/O 通道的状态，准备就绪则传送，反之则等待。程序查询方式的流程如图 4-1 所示。该方式的主要优点是能保证主机与输入、输出通道之间协调工作；主要缺点是需要重复查询输入、输出通道是否“准备就绪”，从而浪费了 CPU 的时间。但由于这种传送方式硬件线路简单，程序容易实现，在计算机控制系统中是一种常用的过程通道控制方式。

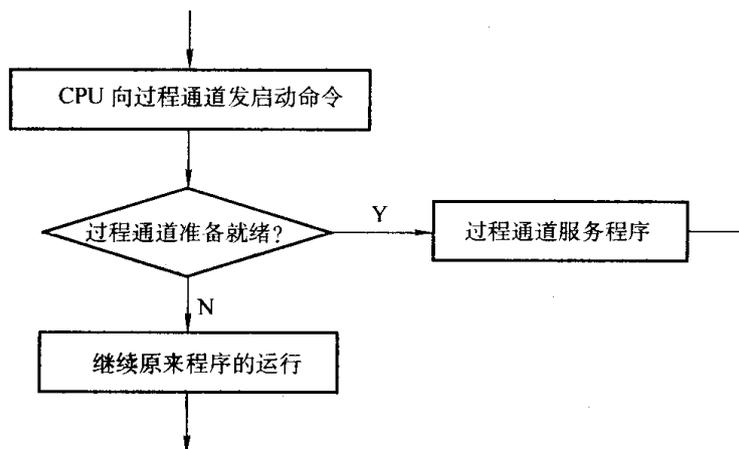


图 4-1 程序查询方式的流程

2. 中断控制方式

采用中断控制方式时，CPU 与 I/O 通道处于异步工作方式。当 CPU 与 I/O 通道需要传送数据时，过程通道做好准备后，主动向 CPU 请求中断，CPU 响应这一请求，并暂停正在运行的程序。系统一般用优先级来解决中断响应的先后顺序问题。

程序查询方式是由 CPU 来查询过程通道的状态，CPU 处于主动地位，外设处于被动地位；而中断控制方式则是过程通道处于主动地位，CPU 处于被动地位。过程通道需要传送数据时就向 CPU 发出中断请求信号，实时性比程序查询方式好。但由于为了能接收中断请求信号，CPU 内部需要有一些线路来控制。另外采用中断控制方式时，每传送一次数据就要中断

一次 CPU 原来的运行, CPU 响应中断后, 每次都要执行中断处理程序, 而且在其执行过程中都要保护断点、恢复断点, 浪费了很多不必要的 CPU 时间。故这种传送方式一般较适合于过程通道传送少量数据的场合。对于需要大量传送数据的场合则应采用高速的直接存储器存取 (DMA) 方式。

3. 直接存储器存取 (DMA) 方式

DMA (Direct Memory Access) 方式是一种完全由硬件完成输入、输出操作的工作方式。在这种方式下, I/O 通道和存储器之间不通过 CPU 而直接进行数据交换。DMA 方式的工作流程如图 4-2 所示。DMA 方式的主要优点是速度快, 数据传送速度只受存储器存取时间的限制; 其缺点是需要一个专用的芯片, 即 DMA 控制器, 来加以控制、管理, 硬件连接也稍显复杂。一般微处理器都设有用于 DMA 传送请求的应答联络线。

由于微电子技术的发展, 目前各种功能的接口电路都已制成集成芯片, 由集成接口芯片代替过去的分列数字电路。因此, 在系统的接口设计中, 通常不再需要复杂的电路参数计算, 而仅需要熟练地掌握和深入了解各类芯片的功能、特点、工作原理和使用方法, 并根据接口功能要求, 合理选择芯片, 把它们与 CPU 正确连接。由于大多数接口芯片功能是可编程的, 所谓“可编程接口”就是指其功能可由程序指令 (接口芯片功能设定的初始化程序) 设定接口芯片的功能。故接口的设计与应用, 除了合理选择 I/O 接口芯片进行硬件设计外, 还应包括对接口芯片的功能初始化程序和接口程序的分析与设计。

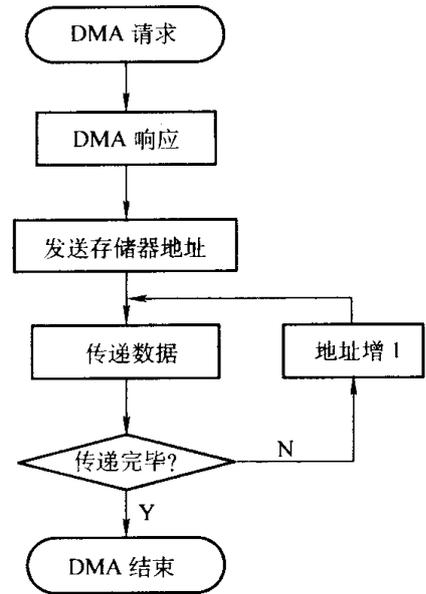


图 4-2 DMA 方式工作流程

4.1.4 过程通道接口设计应考虑的问题

在实时控制系统中, 输入通道实质上是一个信息检测—变换—传递—输入的通道, 输出通道实际上是信息变换—传递—输出的通道。两类通道通过计算机这个信息处理中心有机地构成一个完整的控制系统, 因此过程通道必须通过接口电路来实现与 CPU 的信息交换。接口电路起着连接过程通道与 CPU 的桥梁作用, 它的基本任务主要包括以下两点。

1) 控制信息的传递路径, 即根据控制的任务在众多的信息源中进行选择, 以确定该信息传送的路径和目的地。

2) 控制信息传送的顺序, 计算机控制的过程就是执行程序的过程。为确保进程正确无误, 接口电路应根据控制程序的要求, 适时地发出一组有序的门控信号。

在过程通道接口电路设计中应解决以下问题。

1. 触发方式

有序的门控信号的主要作用就是严格遵循系统工作时序要求, 适时对系统中某个或某些特定部件发出开启或关闭 (触发) 信号, 这必然涉及同步触发和异步触发这两种不同的工作方式。所谓同步触发是指系统的许多相关部件或功能块在同一门控信号作用下完成要求的操作, 如系统的复位信号就是确保系统中各相关部件或功能块回到初始状态的同步信号。异步触发则指各相关部件或功能块不需在同一信号控制下完成自己的操作。接口电路中的各相关

部件或功能块，其内部各单元在外部的同步信号作用下，要完成许多操作，这些操作可以是同步的，也可以是异步的，但必须满足时序要求。因此计算机控制系统是一种复合的触发方式，在同步触发中隐含异步触发，在异步触发中隐含同步触发，但其触发方式和触发时机必须遵循系统的工作时序。

2. 时序

控制逻辑的结构有组合控制逻辑与存储控制逻辑两种类型，不管哪种类型都要严格遵守规定的操作步骤，每一个操作步骤又都是在—组有序的控制信号驱动下实现的，所以进行接口电路设计时，首先要根据系统运行的要求，标出每个控制信号发生的时间顺序和相互之间的时间差，以及与系统时钟的关系，画出时序图，然后根据时序图来确定逻辑电路的结构。

3. 带负载能力

一旦控制逻辑确定后，系统能否可靠地运行与器件的选择关系密切，器件的选择除了要考虑电平的摆幅、数值、延时外，还应考虑器件所带负载是否匹配。

4.2 模拟量输出通道

模拟量输出通道的任务是把计算机处理后的数字量信号转换成模拟电压或电流信号，去驱动相应的执行器，从而达到控制的目的。模拟量输出通道一般是由接口电路、D/A 转换器和 V/I 变换器等构成，如图 4-3 所示，其核心是 D/A 转换器，通常也把模拟量输出通道称为 D/A 通道或 AO 通道。

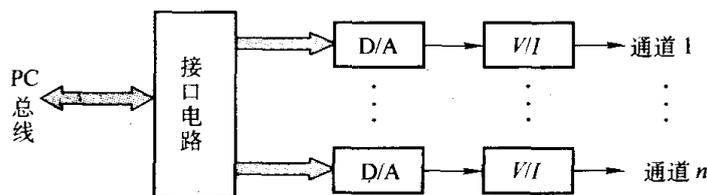


图 4-3 模拟量输出通道的结构组成

一般地，一路模拟量输出使用一个 D/A 转换器，由于 D/A 转换器芯片内部一般都带有数据锁存器，所以 D/A 转换器除承担数字信号到模拟信号的转换任务外，还兼有信号保持作用，即把主机在上一时刻对执行机构的控制作用维持到下一个输出时刻，这是一种数字信号保持方式。只要送给 D/A 转换器的数字信号不变，其模拟输出信号便保持不变。

4.2.1 D/A 转换器

D/A 转换器是一种能把数字量转换成模拟量的电子器件。D/A 转换器芯片类型很多，按位数有 8bit、10bit 和 12bit 等；按输出形式有：电流输出型，如 DAC0832、AD7502、DAC1210，电压输出型，如 AD558、AD3836 等；还有满足特殊需求的 D/A 转换器，如 DAC1420/DAC1422 的输出为 4~20mA，可以直接与控制仪表配套。

1. D/A 转换器的工作原理

现以 4bit D/A 转换器为例说明其工作原理。如图 4-4 所示，D/A 转换器主要由基准电压 V_{REF} 、 $R-2R/T$ 型电阻网络、位切换开关 S_i 和运算放大器 A 等四部分组成。

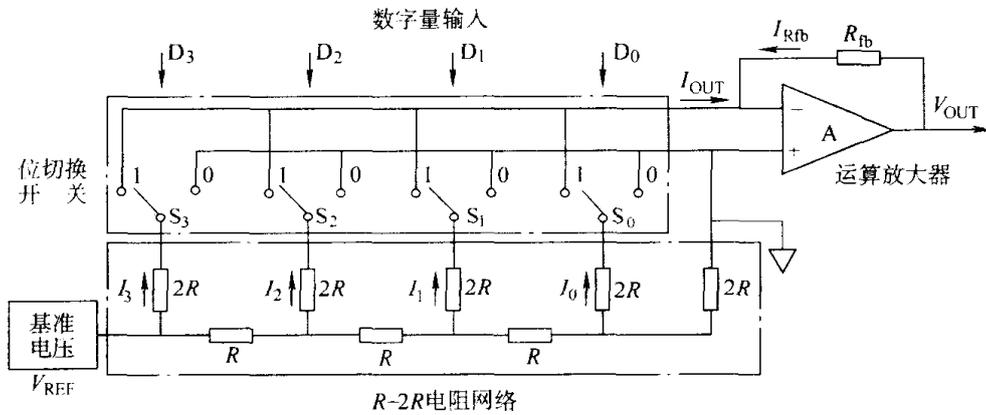


图 4-4 D/A 转换器原理框图

基准电压 V_{REF} 一般由外部稳压电源提供，位切换开关 $S_3 \sim S_0$ 分别接受要转换的二进制数 $D_3 \sim D_0$ 的控制，当某一位 $D_i=1$ ，则相应开关 S_i 切换到“1”端（虚地），就会把基准电压 V_{REF} 加在该分支电阻 $2R$ 上的电流 I_i 切换到放大器的反相端，此电流经反馈电阻 R_{fb} 直至输出端，从而把 $D_i=1$ 转换成相应的模拟电压 V_{OUT} 输出；而当 $D_i=0$ 时， S_i 切换到“0”端（地），则电流 I_i 切换到放大器的正相端流入地中而对放大器输出不起作用。由于 T 型电阻网络中各节点向右看的等效电阻均为 $2R$ ，则各 $2R$ 支路上的电流就按 $1/2$ 系数进行分配，即在各 $2R$ 支路上产生与二进制数各位的权成比例的电流，并经运算放大器 A 相加，从而输出成比例关系的模拟电压 V_{OUT} 。其转换公式推导如下：

$$\begin{aligned} I_{OUT} &= D_3 I_3 + D_2 I_2 + D_1 I_1 + D_0 I_0 \\ &= D_3 \times \frac{V_{REF}}{R} \times \frac{1}{2} + D_2 \times \frac{V_{REF}}{R} \times \frac{1}{4} + D_1 \times \frac{V_{REF}}{R} \times \frac{1}{8} + D_0 \times \frac{V_{REF}}{R} \times \frac{1}{16} \\ &= (D_3 \times 2^3 + D_2 \times 2^2 + D_1 \times 2^1 + D_0 \times 2^0) \times \frac{V_{REF}}{2^4 R} \end{aligned}$$

考虑到放大器反相端为虚地，可得

$$\begin{aligned} V_{OUT} &= I_{R_{fb}} R_{fb} = -I_{OUT} R_{fb} \\ &= -(D_3 \times 2^3 + D_2 \times 2^2 + D_1 \times 2^1 + D_0 \times 2^0) \cdot \frac{V_{REF}}{2^4} \cdot \frac{R_{fb}}{R} \end{aligned}$$

对于 n 位 D/A 转换器，它的输出电压 V_{OUT} 与输入二进制数 B 的关系式可写成

$$V_{OUT} = -(D_{n-1} \times 2^{n-1} + D_{n-2} \times 2^{n-2} + \dots + D_1 \times 2^1 + D_0 \times 2^0) \cdot \frac{V_{REF}}{2^n} \cdot \frac{R_{fb}}{R} = -B \cdot \frac{V_{REF}}{2^n} \cdot \frac{R_{fb}}{R} \quad (4-1)$$

由上述推导可见，输出电压除了与输入的二进制数有关，还与运算放大器的反馈电阻 R_{fb} 以及基准电压 V_{REF} 有关。

2. D/A 转换器性能指标

D/A 转换器性能指标是衡量芯片质量的重要参数，是选用 D/A 转换器芯片型号的依据。

(1) 分辨率。分辨率是指 D/A 转换器能分辨的最小输出模拟增量，即当输入数字发生单位数码变化时所对应输出模拟量的变化量，它取决于能转换的二进制位数。分辨率与二进制位数 n 呈下列关系，即

$$\text{分辨率} = \text{满刻度值} / (2^n - 1) = V_{\text{REF}} / 2^n$$

分辨率实际就是与输入的数字量的最低有效位 (LSB) 相当的输出模拟电压, 简称 1LSB。显然, 数字量位数越多, 分辨率也就越高, 即 D/A 转换器对输入量变化的敏感度也就越高。实际应用时, 应根据分辨率的要求来选定 D/A 转换器的位数。

(2) 转换精度。转换精度是指转换后所得的实际值和理论值的接近程度。它和分辨率是两个不同的概念。例如, 满量程时的理论输出值为 10V, 实际输出值是在 9.99~10.01V 之间, 其转换精度为 $\pm 10\text{mV}$ 。分辨率很高的 D/A 转换器并不一定具有很高的精度。

(3) 偏移量误差。偏移量误差是指输入数字量时, 输出模拟量对于零的偏移值。此误差可通过 D/A 转换器的外接 V_{REF} 和电位器加以调整。

(4) 稳定时间。稳定时间是描述 D/A 转换速度快慢的一个参数, 指从输入数字量变化到输出模拟量达到终值误差 1/2 LSB 时所需的时间。显然, 稳定时间越长, 转换速度越慢。对于输出是电流的 D/A 转换器来说, 稳定时间是很短的, 约为几微秒; 而对于输出是电压的 D/A 转换器, 其稳定时间主要取决于运算放大器的响应时间。

3. 8bit D/A 转换器及其接口

(1) 8bit DAC0832 芯片。8bit D/A 转换器的种类很多, 下面以其中一种典型的芯片——DAC0832 为例进行分析讨论。

DAC0832 芯片是一个 8bit D/A 转换器, 采用电流输出方式, 稳定时间为 $1\mu\text{s}$, 为 20 脚双列直插式封装。同系列芯片还有 DAC0830、DAC0831, 它们可以相互代换。DAC0832 的原理框图及引脚如图 4-5 所示。DAC0832 主要由 8bit 输入寄存器、8bit DAC 寄存器、8bit D/A 转换器以及输入控制电路四部分组成。8bit 输入寄存器用于存放主机送来的数字量, 使输入数字量得到缓冲和锁存, 由 $\overline{\text{LE}}_1$ 加以控制; 8bit DAC 寄存器用于存放待转换的数字量, 由 $\overline{\text{LE}}_2$ 加以控制; 8bit D/A 转换器输出与数字量成正比的模拟电流; 由与门、非与门组成的输入控制电路来控制两个寄存器的选通或锁存状态。

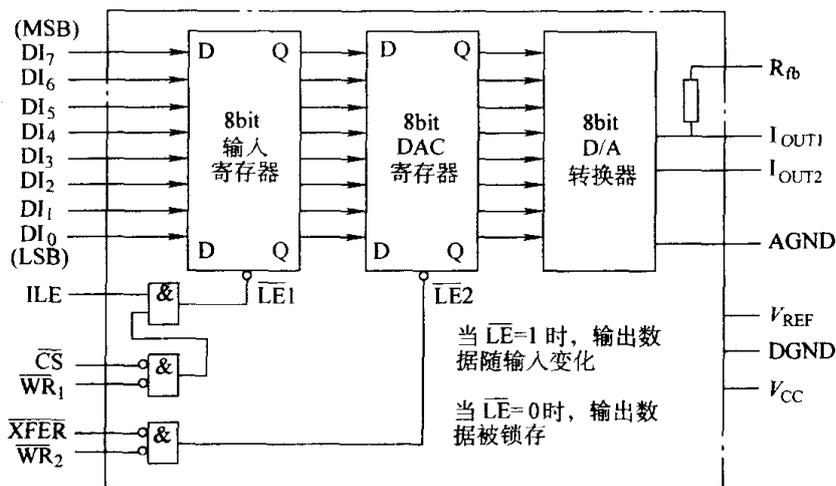


图 4-5 DAC0832 原理框图及引脚

各引脚功能如下。

$DI_0 \sim DI_7$: 数据输入线, 其中 DI_0 为最低有效位 (LSB), DI_7 为最高有效位 (MSB)。

$\overline{\text{CS}}$: 片选信号, 输入线, 低电平有效。

\overline{WR}_1 : 写信号 1, 输入线, 低电平有效。

ILE: 输入允许锁存信号, 输入线, 高电平有效。

当 ILE、 \overline{CS} 和 \overline{WR}_1 同时有效时, 8bit 输入寄存器 $\overline{LE1}$ 端为高电平“1”, 此时寄存器的输出端 Q 跟随输入端 D 的电平变化; 反之, 当 $\overline{LE1}$ 端为低电平“0”时, 原 D 端输入数据被锁存于 Q 端, 在此期间 D 端电平的变化不影响 Q 端。

\overline{WR}_2 : 写信号 2, 输入线, 低电平有效。

\overline{XFER} : 传送控制信号, 输入线, 低电平有效。

当 \overline{WR}_2 和 \overline{XFER} 同时有效时, 8bit DAC 寄存器 $\overline{LE2}$ 端为高电平“1”, 此时 DAC 寄存器的输出端 Q 跟随输入端 D (即输入寄存器 Q 端) 的电平变化; 反之, 当 $\overline{LE2}$ 端为低电平“0”时, 第一级 8bit 输入寄存器 Q 端的状态则锁存到第二级 8bit DAC 寄存器中, 以便第三级 8bit DAC 转换器进行 D/A 转换。

一般情况下为了简化接口电路, 可以把 \overline{WR}_2 和 \overline{XFER} 直接接地, 使第二级 8bit DAC 寄存器的输入端到输出端直通, 只有第一级 8bit 输入寄存器置成可选通、可锁存的单缓冲输入方式。特殊情况下可采用双缓冲输入方式, 即把两个寄存器都分别接成受控方式, 例如, 要求多个 D/A 转换器同步工作时, 首先将要转换的数据依次置入每个 8bit 输入寄存器, 然后用统一信号 (\overline{WR}_2 和 \overline{XFER}) 再同时打开多个 8bit DAC 寄存器, 以便实现多个 D/A 转换器同步输出。

I_{OUT1} : DAC 电流输出端 1, 一般作为运算放大器差动输入信号之一。

I_{OUT2} : DAC 电流输出端 2, 一般作为运算放大器另一个差动输入信号。

R_{fb} : 固化在芯片内的反馈电阻连接端, 用于连接运算放大器的输出端。

V_{REF} : 基准电压源端, 输入线, $-10 \sim +10V$ (DC)。

V_{CC} : 工作电压源端, 输入线, $+5 \sim +15V$ (DC)。

AGND: 模拟电路地。

DGND: 数字电路地。

这是两种不同的地, 但在一般情况下, 这两个地最后总有一点接在一起, 以便提高抗干扰能力。

(2) DAC0832 接口电路。由于 DAC0832 内部有输入寄存器, 所以它的数据总线可直接与主机的数据总线相连, 图 4-6 为 DAC0832 与 PC 总线的单缓冲接口电路, 它是由 DAC0832 转换芯片、运算放大器以及 74LS138 译码器和门电路构成的地址译码电路组成。在图 4-6 中, DAC0832 内的 DAC 寄存器控制端的 \overline{WR}_2 和 \overline{XFER} 直接接地, 使 DAC 寄存器的输入到输出始终直通; 而输入寄存器的控制端分别受地址译码信号与输入/输出指令控制, 即 PC 的地址线 $A_9 \sim A_0$ 经 74LS138 译码器和门电路产生接口地址信号作为 DAC0832 的片选信号 \overline{CS} , 输入/输出写信号 \overline{IOW} 作为 DAC0832 的写信号 \overline{WR}_1 。

当需要进行 D/A 转换时, 把被转换的数据 (如 DATA) 送入累加器 AL, 端口地址 (如 220H) 送入 DX, 然后执行一条 OUT 输出指令, 则 \overline{CS} 和 \overline{WR}_1 (\overline{IOW}) 同为低电平, 且 $\overline{LE1}$ 为“1”, 此时主机输出的数据写入 DAC0832 内的 8bit 输入寄存器, 再直通送入 D/A 转换器进行转换, 当 \overline{IOW} 恢复为高电平时, 使 $\overline{LE1}$ 为“0”, 则要转换的数据锁存在输入寄存器中,

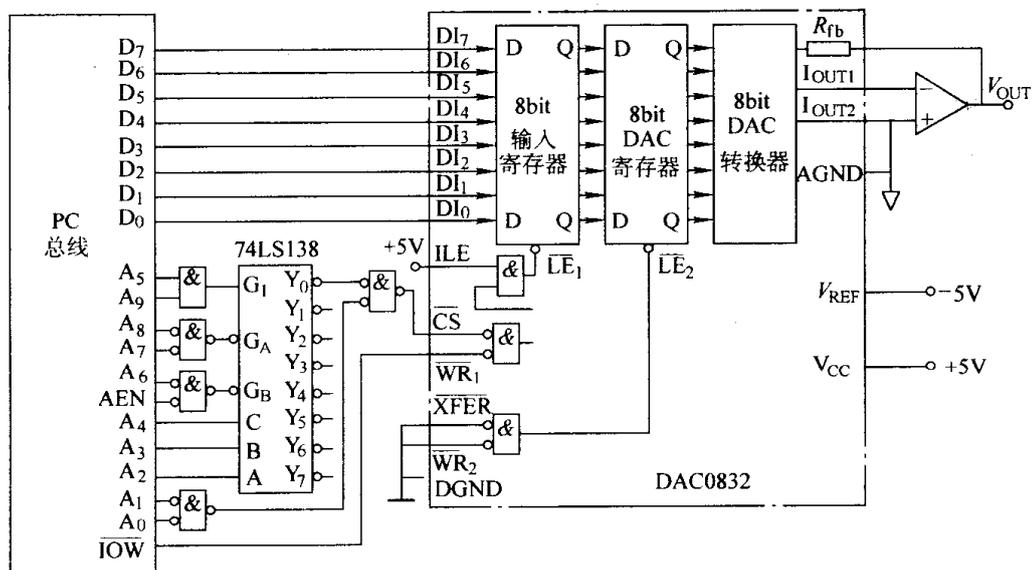


图 4-6 DAC0832 的单缓冲接口电路

使 D/A 转换的输出也保持不变。其接口程序为

```
MOV DX, 220H
MOV AL, [DATA]
OUT DX, AL
```

图 4-7 给出了 8 路 8bit D/A 转换模板的结构组成框图，它是按照总线接口逻辑、I/O 功能逻辑和 I/O 电气接口三部分布局电子元器件的。在图 4-7 中，总线接口逻辑部分主要由数据缓冲与地址译码电路组成，完成 8 路通道的分别选通与数据传送；I/O 功能逻辑部分由 8 片 DAC0832 组成，完成 D/A 转换；而 I/O 电气接口部分由运放与 V/I 变换电路组成（参见 4.2.2 节中的相关部分），实现电压或电流信号的输出。

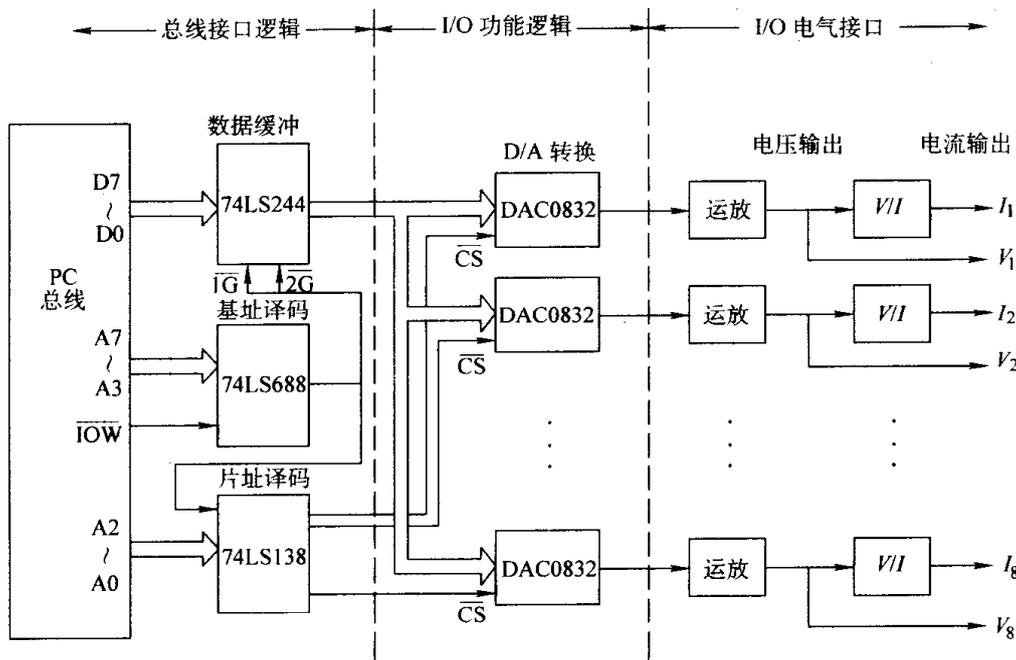


图 4-7 8 路 D/A 转换模板的结构框图

设 8 路 D/A 转换的 8 个输出数据存放在内存数据段 BUF0~BUF7 单元中，主过程已装填 DS，8 片 DAC0832 的通道口地址为 38H~3FH，分别存放在从 CH0 开始的 8 个连续单元中，该 D/A 转换模板的接口子程序如下：

```

DOUT  PROC  NEAR
MOV   CX, 8
MOV   BX, OFFSET BUF0
NEXT: MOV  AL, [BX]
OUT   CH0, AL
INC   CH0
INC   BX
LOOP  NEXT
RET
DOUT  ENDP

```

4. 高于 8bit 的 D/A 转换器及其接口

为了提高转换精度，可选用更多位数的 D/A 转换器，如 10bit、12bit、16bit 等。其转换原理与 8bit D/A 转换器基本一样，不同的是在数据线位数较少的计算机（如 8bit 数据总线的系统）进行接口连接时，数据要分为两次或三次输入。下面以 DAC1210 为例进行分析说明。

(1) 12bit DAC1210 芯片。DAC1210 是一个 12bit D/A 转换器，采用电流输出方式，其结构原理与控制信号功能与 DAC0832 基本类似。由于它比 DAC0832 多了 4 条数据输入线，故有 24 条引脚，DAC1210 内部原理框图如图 4-8 所示，同系列芯片 DAC1208、DAC1209 可以相互代换。

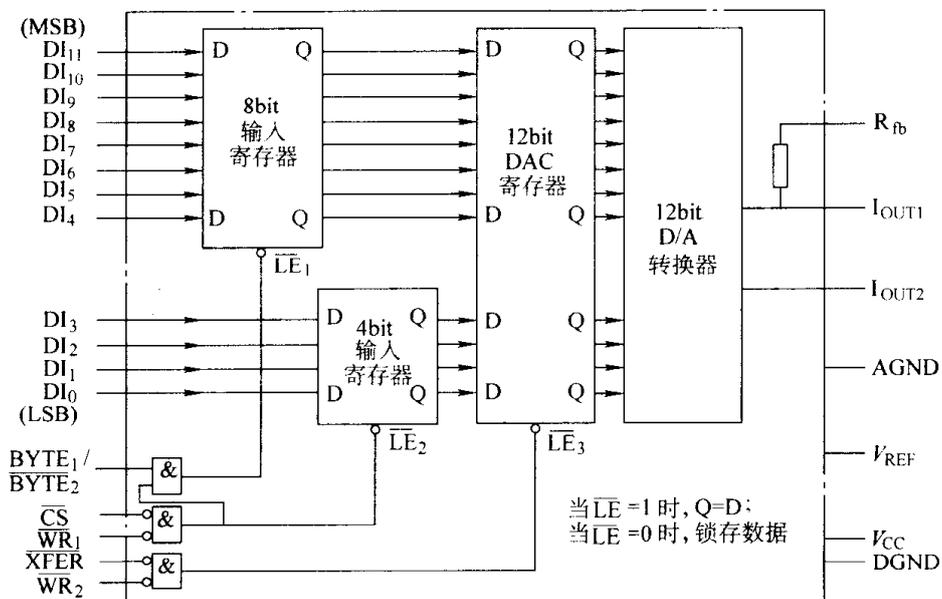


图 4-8 DAC1210 原理框图及引脚

DAC1210 内部有 3 个寄存器：一个是 8bit 输入寄存器，用于存放 12bit 数字量中的高 8 位 $DI_{11} \sim DI_4$ ；另一个是 4bit 输入寄存器，用于存放 12bit 数字量中的低 4 位 $DI_3 \sim DI_0$ ；还

有一个是 12bit DAC 寄存器，存放上述两个输入寄存器送来的 12bit 数字量；12bit D/A 转换器用于完成 12bit 数字量的转换。由与门、非与门组成的输入控制电路来控制 3 个寄存器的选通或锁存状态。其中引脚 \overline{CS} （片选信号，低电平有效）、 \overline{WR}_1 （写信号，低电平有效）和 $\overline{BYTE}_1/\overline{BYTE}_2$ （字节控制信号）的组合，用来控制 8bit 输入寄存器和 4bit 输入寄存器。

当 \overline{CS} 、 \overline{WR}_1 为低电平“0”， $\overline{BYTE}_1/\overline{BYTE}_2$ 为高电平“1”时，与门的输出口 \overline{LE}_1 、 \overline{LE}_2 为“1”，选通 8bit 和 4bit 两个输入寄存器，将要转换的 12bit 数据全部送入寄存器；当 $\overline{BYTE}_1/\overline{BYTE}_2$ 为低电平“0”时， \overline{LE}_1 为“0”，8bit 输入寄存器锁存刚传送的 8bit 数据，而 \overline{LE}_2 仍为“1”，4bit 输入寄存器仍为选通，新的低 4 位数据将刷新刚传送的 4bit 数据。因此，在与计算机接口的电路中，计算机必须先送高 8 位后再送低 4 位。 \overline{XFER} （传送控制信号，低电平有效）和 \overline{WR}_2 （写信号，低电平有效）用来控制 12bit DAC 寄存器，当 \overline{XFER} 和 \overline{WR}_2 同为低电平“0”时，与门输出 \overline{LE}_3 为“1”，12bit 数据全部送入 DAC 寄存器，当 \overline{XFER} 和 \overline{WR}_2 有一个为高电平“1”时，与门输出 \overline{LE}_3 即为“0”，则 12bit DAC 寄存器锁存住数据使 12bit D/A 转换器开始 D/A 转换。

(2) DAC1210 接口电路。DAC1210 内部虽有输入寄存器，但用 PC 中 8bit 数据总线与 12bit D/A 转换器组成接口电路时，数据总线一般需要采用复用的形式。图 4-9 是 12bit D/A 转换器 DAC1210 与 PC 总线的一种接口电路，它是由 DAC1210 转换芯片、运算放大器以及地址译码电路组成。与 8bit DAC0832 接口电路不同的是，除了数据总线 $D_7 \sim D_0$ 与 DAC1210 高 8 位 $DI_{11} \sim DI_4$ 直接相连，位 $D_3 \sim D_0$ 还要与 DAC1210 低 4 位 $DI_3 \sim DI_0$ 复用，因而控制电路也略微复杂。

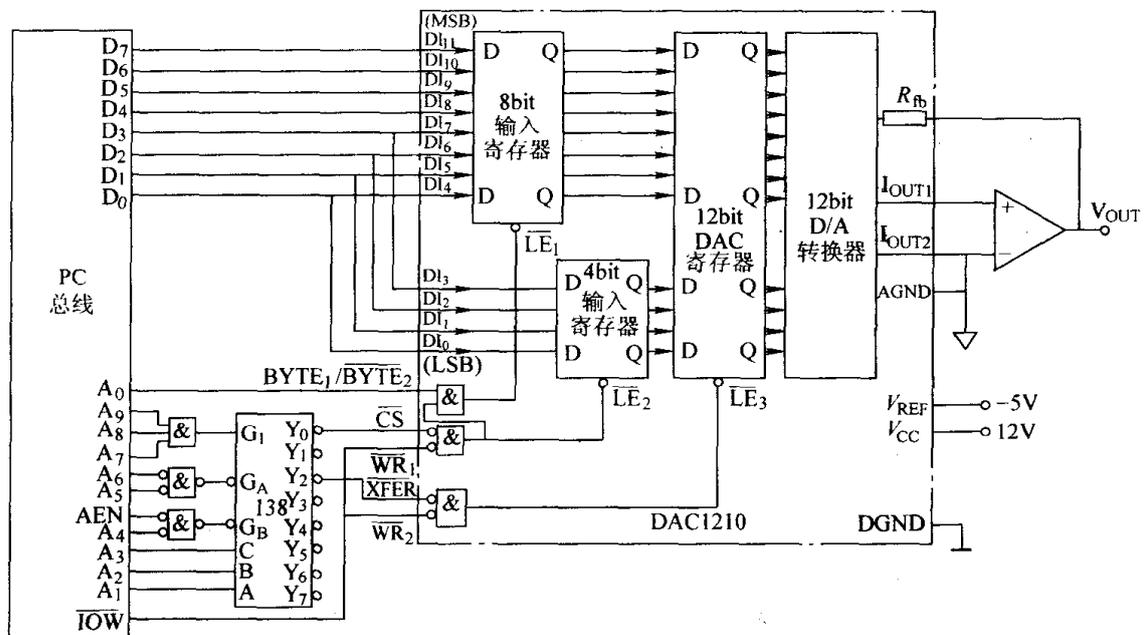


图 4-9 DAC1210 接口电路

在图 4-9 中， \overline{CS} 、 \overline{WR}_1 和 $\overline{BYTE}_1/\overline{BYTE}_2$ 组合，用来依次控制 8bit 输入寄存器（ \overline{LE}_1 ）和 4bit 输入寄存器（ \overline{LE}_2 ）的选通与锁存， \overline{XFER} 和 \overline{WR}_2 用来控制 DAC 寄存器（ \overline{LE}_3 ）的

选通与锁存, \overline{IOW} 与 $\overline{WR_1}$ 、 $\overline{WR_2}$ 连接, 用来在执行输出指令时获得低电平有效, 译码器的两条输出线 $\overline{Y_0}$ 、 $\overline{Y_2}$ 分别连到 \overline{CS} 和 \overline{XFER} , 一条地址线 A_0 连到 $\overline{BYTE_1}/\overline{BYTE_2}$, 从而形成三个端口地址: 低 4 位输入寄存器为 380H; 高 8 位输入寄存器为 381H; 12 位 DAC 寄存器为 384H。

在软件设计中, 为了实现 8bit 数据线 $D_7 \sim D_0$ 传送 12bit 被转换数, 主机必须分两次传送被转换数。首先将被转换数的高 8 位传给 8 位输入寄存器 $DI_{11} \sim DI_4$, 再将低 4 位传给 4bit 输入寄存器 $DI_3 \sim DI_0$, 然后再打开 DAC 寄存器, 把 12bit 数据送到 12 位 D/A 转换器去转换。当输出指令执行完后, DAC 寄存器自动处于锁存状态以保持数模转换的输出不变。设 12bit 被转换数的高 8 位存放在 DATA 单元中, 低 4 位存放在 DATA+1 单元中。其转换程序为:

```

DAC: MOV    DX, 0381H
      MOV    AL, [DATA]
      OUT    DX, AL           ; 送高 8 位数据
      DEC    DX
      MOV    AL, [DATA+1]
      OUT    DX, AL         ; 送低 4 位数据
      MOV    DX, 0384H
      OUT    DX, AL         ; 完成 12 位数据转换
  
```

4.2.2 D/A 通道的输出电路

多数 D/A 转换芯片输出的是弱电流信号, 要驱动后面的自动化装置, 需在电流输出端外接运算放大器。根据不同控制系统中自动化装置需求的不同, 输出方式可以分为电压输出、电流输出以及自动/手动切换输出等多种方式。

1. 电压输出方式

由于系统要求不同, 电压输出方式又可分为单极性输出和双极性输出两种形式。

(1) 单极性输出。D/A 转换器单极性输出方式如图 4-10 所示。

由式 (4-1), 假设 $R_{fb} = R$, 可得输出电压 V_{OUT} 的单极性输出表达式为

$$V_{OUT} = -B \times V_{REF} / 2^n \quad (4-2)$$

式中, B 为转换的二进制数码; $V_{REF} / 2^n$ 为参考电源和转换器位数所确定的常数。

显然, V_{OUT} 和 B 成正比关系: 当输入数字量 B 为 00H 时, V_{OUT} 也为 0; 输入数字量 B 为 FFH 即 255 时, V_{OUT} 为与 V_{REF} 极性相反的最大值。

(2) 双极性输出方式。D/A 转换器双极性输出方式如图 4-11 所示。 V_{OUT1} 为单极性电压输出, V_{OUT2} 为双极性电压输出。

A_1 和 A_2 为运算放大器。 A 点为虚地, 故可得

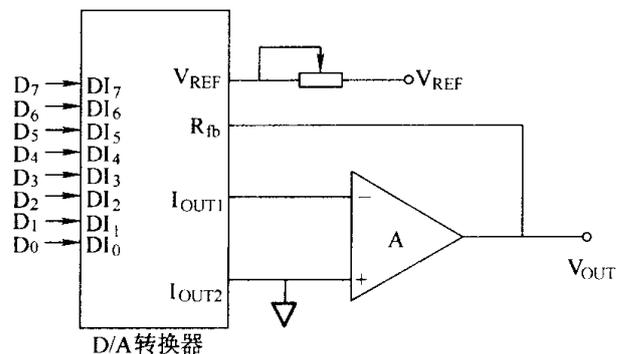


图 4-10 D/A 转换器单极性输出方式

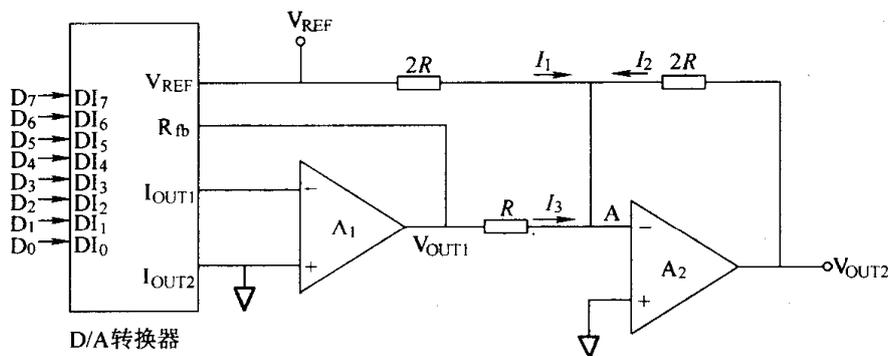


图 4-11 D/A 转换器双极性输出方式

$$I_1 + I_2 + I_3 = 0$$

$$V_{OUT1} = -B \times V_{REF} / 2^n$$

$$I_1 = \frac{V_{REF}}{2R}$$

$$I_2 = \frac{V_{OUT2}}{2R}$$

$$I_3 = \frac{V_{OUT1}}{R}$$

解上述方程可得双极性输出表达式为

$$V_{OUT2} = (B - 2^{n-1}) \times V_{REF} / 2^{n-1} = V_{REF} (B / 2^{n-1} - 1) \quad (4-3)$$

在图 4-11 中，运放 A₂ 的作用是将运放 A₁ 的单向输出变为双向输出。当输入数字量小于 2ⁿ⁻¹ 时，输出模拟电压为负；当输入数字量大于 2ⁿ⁻¹ 时，输出模拟电压为正。

2. 电流输出方式

因为电流信号易于远距离传送，且不易受到干扰，特别是在过程控制系统中，自动化仪表接收的是电流信号，所以在计算机控制输出通道中常以电流信号来传送信息，这就需要将电压信号转换成毫安级的电流信号，实现电流输出方式的电路称为 V/I 变换电路。

实现 V/I 变换可以采用专用的电流输出型运放 F3080 和 F3094，也可以利用普通运放，还可以使用高精度的集成 V/I 变换器。下面介绍几种常用电路。

(1) 普通运放 V/I 变换电路有以下几种情况

1) 0~10V/0~10mA 变换电路。图 4-12 为 0~10V/0~10mA 的典型变换电路，由运放 A 和晶体管 VT₁、VT₂ 组成，R₁ 和 R₂ 是输入电阻，R_f 是反馈电阻，R_L 是负载的等效电阻。

输入电压 V_{in} 经输入电阻进入运算放大器 A，放大后进入晶体管 VT₁、VT₂。由于 VT₂ 射极接有反馈电阻 R_f，得到反馈电压 V_f 加至输入端，形成运放 A 的差动输入信号。该变

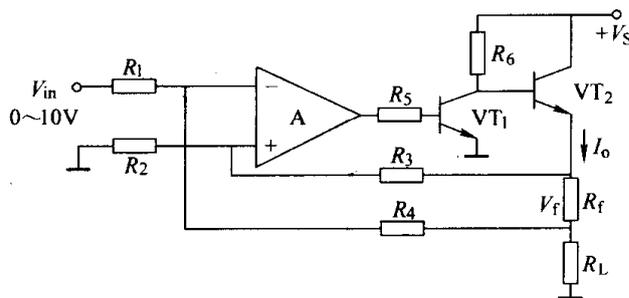


图 4-12 0~10V/0~10mA 转换电路

换电路由于具有较强的电流反馈，所以有较好的恒流性能。

输入电压 V_{in} 和输出电流 I_o 之间的关系如下：

若 R_3 、 $R_4 \gg R_f$ 、 R_L ，则可以认为 I_o 全部流经 R_f ，由此可得

$$V_- = V_{in} R_4 / (R_1 + R_4) + I_o R_L R_1 / (R_1 + R_4)$$

$$V_+ = I_o (R_L + R_f) R_2 / (R_2 + R_3)$$

对于运放，有 $V_- \approx V_+$ ，若取 $R_1 = R_2$ 、 $R_3 = R_4$ ，则整理得

$$I_o = V_{in} R_3 / (R_1 R_f) \quad (4-4)$$

可以看出，输出电流 I_o 和输入电压 V_{in} 呈线性对应的单值函数关系， $R_3 / (R_1 R_f)$ 为一常数，与其他参数无关。若取 V_{in} 为 $0 \sim 10V$ ， $R_1 = R_2 = 100k\Omega$ ， $R_3 = R_4 = 20k\Omega$ ， $R_f = 200\Omega$ ，则输出电流 I_o 为 $0 \sim 10mA$ 。

2) $1 \sim 5V/4 \sim 20mA$ 变换电路。图 4-13 为 $1 \sim 5V/4 \sim 20mA$ 的变换电路，两个运放 A_1 、 A_2 均接成射极输出形式。在稳定工作时， $V_{in} = V_1$ ，所以 $I_1 = V_1 / R_1 = V_{in} / R_1$ 。又因为 $I_1 \approx I_2$ ，所以

$$V_{in} / R_1 = I_2 = (V_s - V_2) / R_2$$

即

$$V_2 = V_s - V_{in} \cdot R_2 / R_1$$

在稳定状态下， $V_2 = V_3$ ， $I_f \approx I_o$ ，故

$$I_o \approx I_f = (V_s - V_3) / R_f = (V_s - V_2) / R_f$$

将 V_2 代入上式得

$$I_o = V_{in} R_2 / (R_1 R_f) \quad (4-5)$$

式中， R_1 、 R_2 、 R_f 均为精密电阻，所以输出电流 I_o 线性正比于输入电压 V_{in} ，且与负载无关，接近于恒流。若 $R_1 = 5k\Omega$ ， $R_2 = 2k\Omega$ ， $R_3 = 100\Omega$ ，当 V_{in} 为 $1 \sim 5V$ 时，输出电流 I_o 为 $4 \sim 20mA$ 。

(2) 集成芯片 V/I 变换电路。图 4-14 是集成 V/I 变换器 ZF2B20 的引脚，采用单正电源供电，电源电压范围为 $10 \sim 32V$ ，ZF2B20 的输入电阻为 $10k\Omega$ ，动态响应时间小于 $25\mu s$ ，非线性为 $\pm 0.025\%$ 。

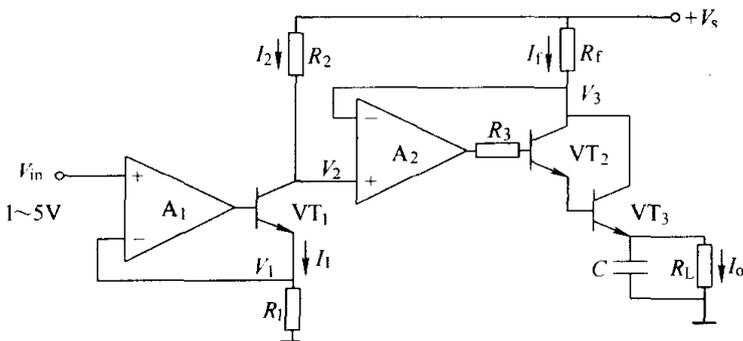


图 4-13 $1 \sim 5V/4 \sim 20mA$ 的变换电路

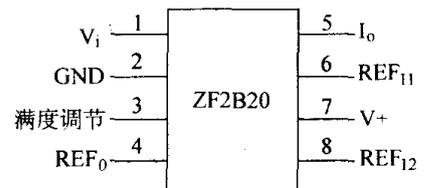


图 4-14 ZF2B20 引脚

通过 ZF2B20 可以产生一个与输入电压成比例的输出电流，其输入电压范围是 0~10V，输出电流是 4~20mA。它的特点是低漂移，在工作温度为 -25~85℃ 范围内，最大温漂为 0.005%/℃。

利用 ZF2B20 实现 V/I 变换的电路非常简单，图 4-15a 所示电路是一种带初值校准的 0~10V/4~20mA 的变换电路；图 4-15b 则是一种带满度校准的 0~10V/0~10mA 的变换电路。

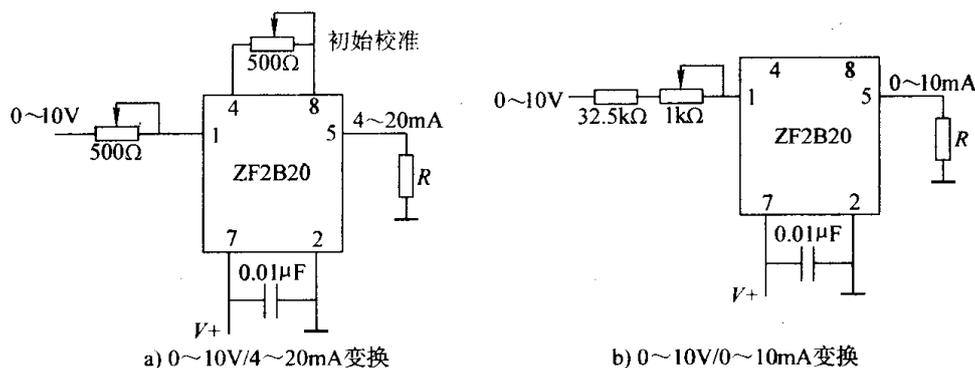


图 4-15 ZF2B20 V/I 变换电路

3. 自动/手动输出方式

计算机过程控制中，除了正常运行时的自动控制方式外，在开、停机或发生事故的情况下还需要进行手动控制。在自动/手动控制方式切换的瞬间前后，必须保持系统的输出信号不变，才能对执行器的现有位置不产生扰动，即达到双向切换时平衡对位的无扰动切换。下面讨论这种具有自动/手动无扰动切换功能的 V/I 变换电路。

在如图 4-16 所示电路中，是在普通运放 V/I 变换电路的基础上，增加了自动、手动切换开关 S₁、S₂、S₃ 和手动增减电路与输出跟踪电路。

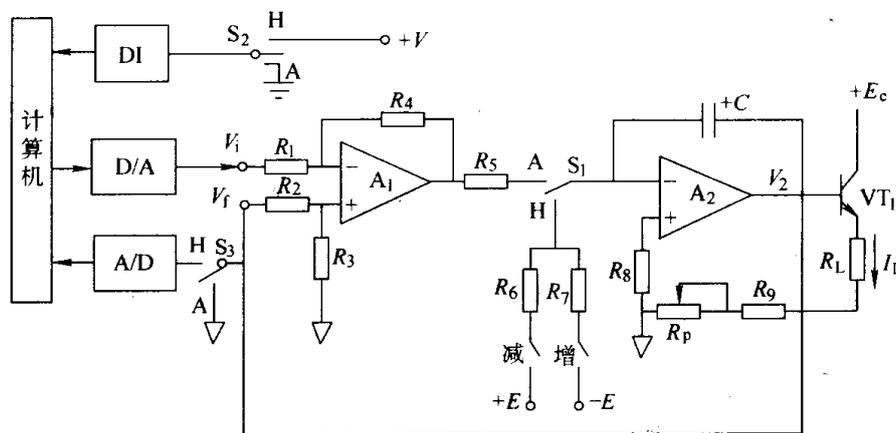


图 4-16 带自动/手动切换的 V/I 变换电路

(1) 自动/手动状态下的 V/I 变换

1) 当开关处于自动 (A) 状态时，运放 A₂ 与 A₁ 接通，形成一个电压比较型跟随器。当 V_f ≠ V_i 时，电路能自动地使输出电流增大或减小，最终使 V_f = V_i，于是有

$$I_L = V_i / (R_9 + R_p) \tag{4-6}$$

从式 (4-6) 可见，只要电阻 R₉、R_p 稳定性好，A₁、A₂ 具有较好的增益，该电路就有较高的线性精度。当 R₉ + R_p = 500Ω 或 250Ω 时，输出电流 I_L 就以 0~10mA 或 4~20mA 的直

流电流信号线性地对应 V_i 的 $0\sim 5V$ 或 $1\sim 5V$ 的直流电压信号。

2) 当开关处于手动(H)状态时,此时运放 A_2 与 A_1 断开,成为一个保持型反相积分器。当按下“增”按钮时, V_2 以一定的速率上升,从而使 I_L 也以同样的速率上升;当按下“减”按钮时, V_2 以一定的速率下降, I_L 也以同样的速率下降。负载 R_L (一般为电动调节阀)上的电流 I_L 的升、降速率取决于 R_6 、 R_7 、 C 和电源电压 $\pm E$ 的大小,而手动操作按钮的时间长短决定输出电流 I_L 的大小。

(2) 自动/手动双向无扰动切换

1) 自动到手动的切换:当开关 S_1 、 S_2 、 S_3 都从自动(A)切换为手动(H)时,“增”、“减”两按钮处于断开状态,运放 A_2 为一高输入阻抗保持器,则 A_2 的输出 V_2 几乎保持不变,从而维持输出电流 I_L 恒定。因此,自动到手动的无扰动切换是由手动操作电路的硬件实现的。

2) 手动到自动的切换:当开关 S_1 、 S_2 、 S_3 从手动(H)切换为自动(A)时,要做到无扰动切换必须使自动输出电路具有跟踪手动输出的功能,即在手动状态下,使来自计算机D/A转换电路的自动输入信号 V_i 始终等于反映手动输出的信号 V_f (V_f 与 I_L 总是一一对应的)。要达到这个目的,必须有如图4-16所示的硬件电路与相应的跟踪程序配合(软件跟踪说明见第8章有关内容)。其工作过程是这样的:在每个控制周期,计算机首先由数字量输入通道(DI)读入开关 S_2 的状态,以判断输出电路是处于手动状态或是自动状态。若是自动状态,则程序执行本回路预先规定的控制运算,输出 V_i 并通过 V/I 变换输出电流 I_L ;若为手动状态,则首先由A/D通道读入 V_f 并转换为数字信号,然后原封不动地将此数字信号送出,由D/A转换为电压信号送至输出电路的输入端 V_i ,这样使 V_i 始终与 V_f 相等。当开关 S_1 从手动(H)切换为自动(A)时,因 $V_i \equiv V_f$, V_2 与 I_L 会保持不变,从而实现了手动到自动方式的无扰动切换。

4.3 模拟量输入通道

在过程计算机控制系统中,输入信号多是模拟量,那么控制计算机是如何通过输入通道采集模拟信号的呢?

模拟量输入通道的任务是把被控对象的过程参数如温度、压力、流量、液位等模拟量信号转换成计算机可以接收的数字量信号。典型模拟量输入通道的结构组成如图4-17所示。来自于工业现场传感器或变送器的多个模拟量信号首先需要进行信号调理,然后经多路模拟开

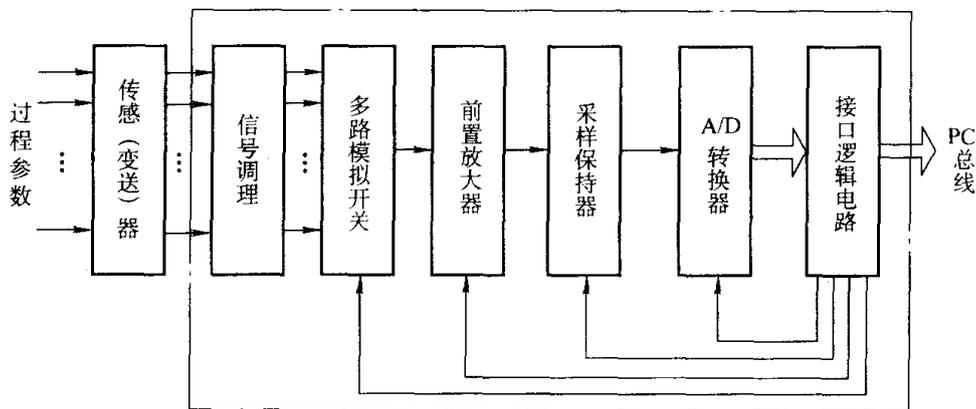


图 4-17 模拟量输入通道的结构组成

关, 分时切换到后级进行前置放大、采样保持和 A/D 转换, 通过接口逻辑电路以数字量信号进入主机系统, 从而完成对过程参数的巡回检测任务。根据实际情况的不同, 也常将前置放大器放在多路模拟开关之前, 这时需要使用与通道数量相同的多个前置放大器。显然, 这里的核心是 A/D 转换器, 通常把模拟量输入通道称为 A/D 通道或 AI 通道。

4.3.1 信号调理与放大

在模拟量输入通道中, 对现场可能引入的各种干扰必须采取相应的技术措施以保证 A/D 转换的精度, 所以首先要在通道之前设置输入信号调理电路。根据通道需要, 可以采取不同的信号调理技术, 如信号滤波、光电隔离、电平转换、过电压保护、反电压保护、 I/V 变换及放大等。下面主要介绍模拟量输入通道中常用的 I/V 变换和放大技术, 其余部分参见本章以及第 3 章的有关内容。

1. 信号调理电路

在控制系统中, 对被控量的检测往往用到各种类型的测量变送器, 当它们的输出信号为 $0\sim 10\text{mA}$ 或 $4\sim 20\text{mA}$ 的电流信号时, 一般是采用电阻分压法把现场传送来的电流信号转换为电压信号, 图 4-18 所示为两种常用的无源 I/V 变换和有源 I/V 变换电路。

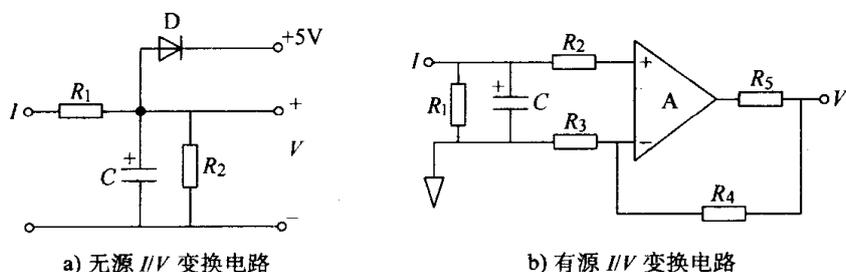


图 4-18 I/V 变换电路

2. 前置放大器

前置放大器的任务是将模拟输入小信号放大到 A/D 转换的量程范围之内, 如 $0\sim 5\text{V}$ (DC)。

对单纯的微弱信号, 可用一个运算放大器进行单端同相放大或单端反相放大。如图 4-19 所示, 若信号源的一端接放大器的正端, 则为同相放大, 同相放大电路的放大倍数 $G=1+R_2/R_1$; 若信号源的一端接放大器的负端, 则为反相放大, 反相放大电路的放大倍数 $G=-R_2/R_1$ 。当然, 这两种电路都是单端放大, 所以信号源的另一端是与放大器的另一个输入端共地。

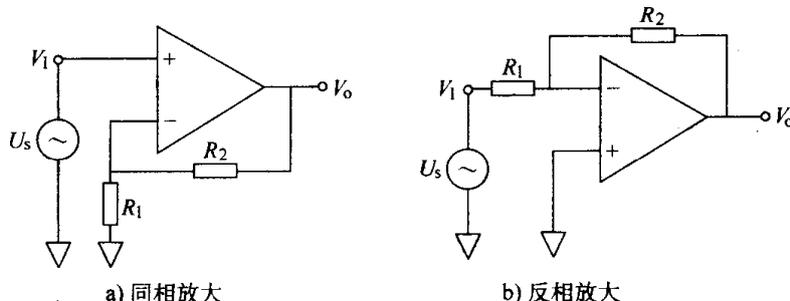


图 4-19 放大电路

但是来自生产现场的传感器信号往往带有较大的共模干扰, 而单个运放电路的差动输入

端难以起到很好的抑制作用。因此，A/D 通道中的前置放大器常采用由一组运放构成的测量放大器，也称仪表放大器，如图 4-20a 所示。

经典的测量放大器是由三个运放组成的对称结构，测量放大器的差动输入端 V_{IN+} 和 V_{IN-} 分别是两个运放 A_1 、 A_2 的同相输入端，输入阻抗很高，而且完全对称地与被测信号相连，因而有着极强的抑制共模干扰的能力。

在图 4-20a 中， R_G 是外接电阻，专门用来调整放大器增益，输入、输出间有如下关系：

$$G = \frac{V_{OUT}}{V_{IN+} - V_{IN-}} = \frac{R_s}{R_2} \left(1 + \frac{2R_1}{R_G} \right) \quad (4-7)$$

目前，这种测量放大器的集成电路芯片有多种，如 AD521/AD522、INA102 等。

在 A/D 转换通道中，多路被测信号经常共用一个测量放大器，而各路输入信号大小往往不同，但都要放大到 A/D 转换器的同一量程范围。因此，对应于各路不同大小的输入信号，测量放大器的增益也应不同。具有这种性能的放大器称为可变增益放大器或可编程放大器，如图 4-20b 所示。

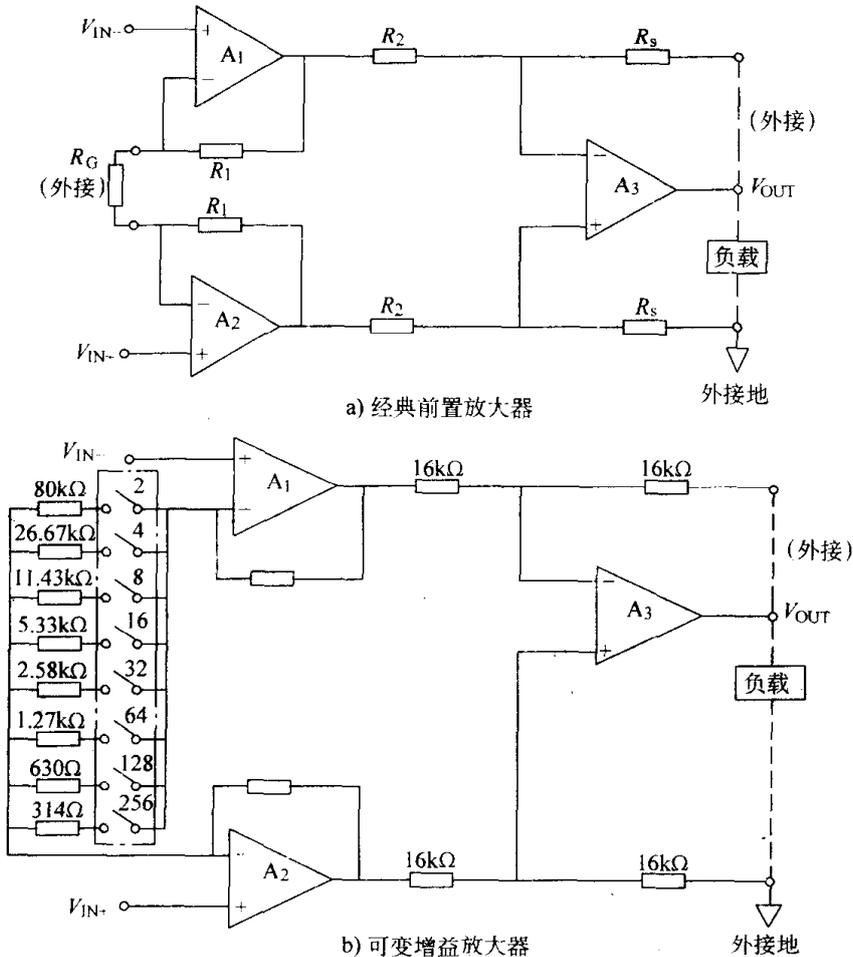


图 4-20 前置放大器

把图 4-20a 中的外接电阻 R_G 换成一组精密的电阻网络，每个电阻支路上有一个开关，通过支路开关依次通、断就可改变放大器的增益，根据开关支路上的电阻值与增益公式，就可算得支路开关自上而下闭合时的放大器增益分别为 2、4、8、16、32、64、128、256 倍。显

然，这一组开关如果用多路模拟开关（类似于 CD4051）就可方便地进行增益可变的计算机数字程序控制。此类可变增益放大器的集成电路芯片有 AD612、AD614 等。

4.3.2 多路模拟开关

由于计算机的工作速度远快于被测参数的变化，因此一台计算机系统可供几十个检测回路使用，但由于一般均采用一个 A/D 转换器的结构，计算机在某一时刻只能接收一个回路的信号。所以，必须通过多路模拟开关实现多选一的操作，将多路输入信号依次地切换到后级。

目前，计算机控制系统使用的多路开关种类很多，并具有不同的功能和用途。如集成电路芯片 CD4051（双向、单端、8 路）、CD4052（单向、双端、4 路）及 AD7506（单向、单端、16 路）等。所谓双向，就是该芯片既可以实现多到一的切换，也可以完成一到多的切换；而单向则只能完成多到一的切换。双端是指芯片内的一对开关同时动作，从而完成差动输入信号的切换，以满足抑制共模干扰的需要。需要注意的是，多路模拟开关不能使各输入信号的电信号完全隔离开来，如需进行路间隔离必须在此之前分别对各输入信号进行变压器隔离或光电隔离。

现以常用的 CD4051 为例，介绍模拟开关的结构原理，如图 4-21 所示。CD4051 由电平转换、译码驱动及开关电路三部分组成。当禁止端 $\overline{\text{INH}}$ 为“1”时，前、后级通道断开，即 $S_0 \sim S_7$ 与 S_m 端不可能接通；当 $\overline{\text{INH}}$ 为“0”时，则通道可以被接通，通过改变控制输入端 C、B、A 的数值，就可选通 8 个通道 $S_0 \sim S_7$ 中的一路。比如，当 $CBA=000$ 时，通道 S_0 选通；当 $CBA=001$ 时，通道 S_1 选通；……，当 $CBA=111$ 时，通道 S_7 选通。其真值表如表 4-1 所列。

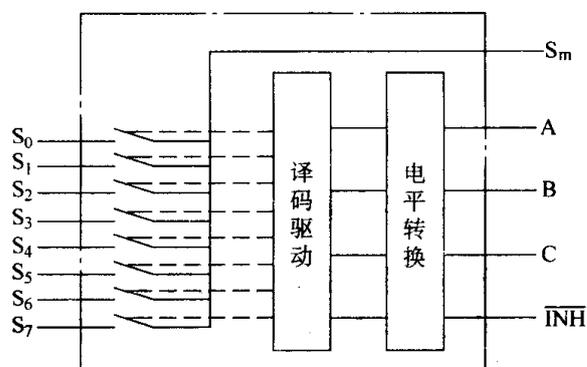


图 4-21 CD4051 结构原理

表 4-1 CD4051 的真值表

$\overline{\text{INH}}$	输 入			所选通道
	C	B	A	
0	0	0	0	S_0
0	0	0	1	S_1
0	0	1	0	S_2
0	0	1	1	S_3
0	1	0	0	S_4
0	1	0	1	S_5
0	1	1	0	S_6
0	1	1	1	S_7
1	×	×	×	无

注：表中×表示 1 或 0 都可以。

在实际应用中，输入信号很多时，常需要用多片集成模拟开关对通道数进行扩展；同时，由于单端输入地线共用，由地线引起的干扰大，为此可选用差动输入的芯片或采用多片单端

输入芯片构成双端输入电路，以抑制共模干扰。图 4-22 所示就是利用 4 片 CD4051 构成的一个 16 路差动输入的转换开关扩展电路。

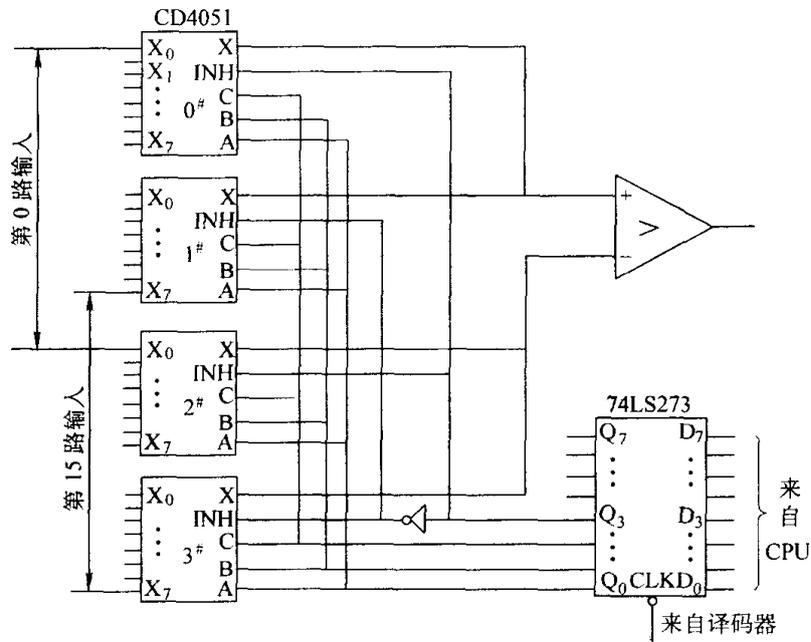


图 4-22 16 路差动输入模拟开关扩展电路

4.3.3 采样保持器

采样保持器 (S/H) 的主要作用在于保证 A/D 转换器进行转换期间，输入电压保持不变，以免引起 A/D 转换器的转换误差。它有两个工作模式：一是采样，另一是保持。在采样状态时，其输出能跟随输入电压的变化而变化，故该状态亦称跟踪状态；而当处于保持状态时，其输出将保持在进入保持状态瞬间的输入电压的值不变。

图 4-23a 所示为典型的采样保持器的基本电路。它是由模拟开关 K、保持电容 C 和两个运算放大器 A₁、A₂ 组成。其中 A₁、A₂ 都接成跟随器形式，以增大采样保持器的输入阻抗，降低其输出阻抗。当控制信号 V_c 为高电平时，S/H 处于“采样”状态，这时开关 K 闭合，输入电压 V_i 经过 A₁ 向电容 C 充电。因 A₁ 的输出阻抗小，即充电时间常数小，故 C 上的电压可迅速跟踪上输入信号 V_i 的变化，并经 A₂ 输出，即 V_o ≈ V_i；当控制信号 V_c 为低电平时，S/H 处

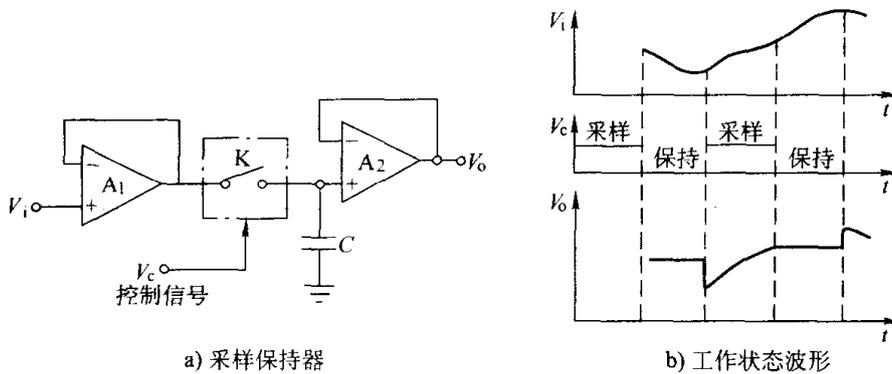


图 4-23 采样保持器工作原理

于“保持”状态，这时开关 K 断开，由于 A_2 的输入阻抗很大，漏电流很小，电容 C 放电时间常数很大，这样电容 C 就将开关 K 断开前瞬间的输入电压 V_i 值保持一段时间，并经 A_2 继续输出。图 4-23b 所示为其各工作状态的波形。

由以上分析可知，电容 C 对采样保持的精度有很大影响。C 值小，则采样状态时充电时间常数小，即电容 C 充电快，输出对输入信号的跟随特性好，但在保持状态时放电时间常数也小，即电容 C 放电快，故保持性能差；反之，C 值大，保持性能好，但跟随特性差。因此，在选择电容时，应均衡考虑其容量的大小。

A/D 转换器完成一次完整的转换过程所需的时间称转换时间。对变化快的模拟信号来说，转换期间将引起转换误差，这个误差称为孔径误差，如图 4-24 所示。

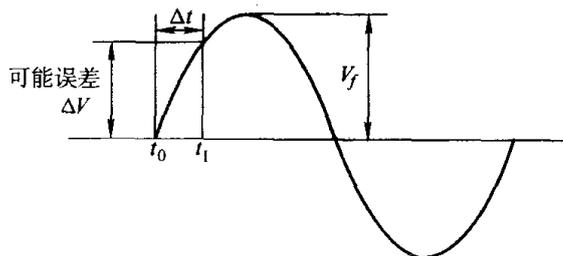


图 4-24 孔径误差

$$\text{设模拟信号为} \quad v_f = V_f \sin 2\pi f t \quad (4-8)$$

$$\text{其微分为} \quad dv_f / dt = 2\pi f V_f \cos 2\pi f t \quad (4-9)$$

$$\text{最大变化率为} \quad \left. \frac{dv_f}{dt} \right|_{\max} = 2\pi f V_f \quad (4-10)$$

在信号与横坐标交点处，信号变化率最大，可能引起最大的信号误差，设孔径时间为 Δt ，这时最大误差为

$$\Delta V = V_f 2\pi f \Delta t \quad (4-11)$$

为满足 A/D 转换精度要求，希望在 Δt 时间内，信号变化最大幅度应小于 A/D 转换器的量化误差 ΔE 。对于一个 12bit A/D 转换器，若转换时间为 $100\mu\text{s}$ ，基准电压为 10.24V ，其量化误差为

$$\Delta E = \frac{1}{2} \text{LSB} = \frac{1}{2} \times \frac{10.24 \times 10^3}{2^{12}} = 1.25(\text{mV}) \quad (4-12)$$

若 $V_f = 5\text{V}$ ，由此要求输入信号的最高变化频率

$$f_{\max} \leq \frac{1}{2\pi V_f} \cdot \frac{\Delta E}{\Delta t} \approx 0.5\text{Hz} \quad (4-13)$$

由此可见，当转换时间越长时，不影响转换精度所允许的信号最高频率就越低，这将大大地限制 A/D 转换器的工作频率范围。在 A/D 转换前加入采样保持器，可以在满足转换精度的条件下提高信号允许的工作频率。

在选择采样保持器时，一般应考虑以下几个主要参数。

(1) 孔径时间：电路接到保持信号后，模拟开关由导通转变为断开所需的时间。显然，对于一个动态的模拟输入信号在此期间会发生变化，这将导致 A/D 转换产生不确定性误差（孔径误差）。

(2) 捕捉时间：电路接到采样控制信号后，输出电压 V_o 达到指定跟踪误差范围内所需的时间。A/D 转换器的采样周期应大于捕捉时间。

(3) 保持时间：模拟开关 K 断开的时间，由采样速率确定。

(4) 变化率 dV_o/dt ：反映在保持阶段，由于保持电容 C 漏电或放大器 A_2 的漏电流所引

起的保持电压的变化。为此，选取泄漏电阻比较大的电容（一般均采取外接形式），如聚苯乙烯或聚四氟乙烯等电容，并且 A_2 应采用高输入阻抗的运算放大器。

同时要注意，模拟输入线尽可能远离逻辑控制线，以避免由 V_c 信号的突变所产生的干扰。

常用的集成采样保持器有 AD582、LF198/LF298/LF398 等，其内部结构和引脚如图 4-25a、b 所示。这里，用 TTL 逻辑电平控制采样和保持状态，如 AD582 的采样电平为“0”，保持电平为“1”，而 LF198 的则相反。

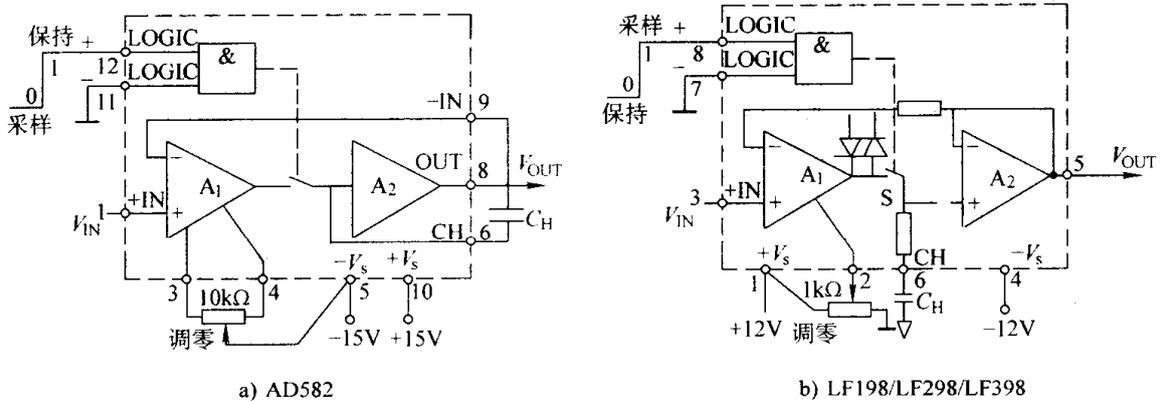


图 4-25 集成采样保持器

在 A/D 通道中，采样保持器的采样和保持电平应与后级的 A/D 转换器相配合，该电平信号既可以由其他控制电路产生，也可以由 A/D 转换器直接提供。总之，保持器在采样期间，不启动 A/D 转换器，而一旦进入保持期间，则立即启动 A/D 转换器，从而保证 A/D 转换时的模拟输入电压恒定，以确保 A/D 转换精度（可参见图 4-38 所示的 8 路 12bit A/D 转换接口电路）。

考虑到一般在过程控制系统中，信号变化慢，频率较低，如果采用转换速度快的 A/D 转换器，如逐位逼近式 A/D 转换器的转换时间一般为几十 μs ，在可以保证转换精度的前提下，也可以在输入通道中不使用采样保持器。

4.3.4 A/D 转换器工作原理

A/D 转换器从原理上主要可分为逐位逼近式、双积分式、电压/频率式 (V/F 转换) 和 Σ - Δ 式等多种形式。其中双积分式和 Σ - Δ 式的 A/D 转换器，转换精度高，价格低廉，在仪器仪表中应用广泛，多用于数据采集系统及精度要求比较高的场合，但转换速度较慢；逐位逼近式 A/D 转换器既照顾了转换速度，又具有一定的精度，是目前应用最多的一种；V/F 转换则多用于需要远距离串行传送的场合。下面分别介绍其工作原理。

1. 逐位逼近式 A/D 转换原理

一个 n 位 A/D 转换器是由 n 位寄存器、 n 位 D/A 转换器、运算比较器、控制逻辑电路及输出锁存器等五部分组成。现以 4bit A/D 转换器把模拟量 9 转换为二进制数 1001 为例，说明逐位逼近式 A/D 转换器的工作原理。

如图 4-26 所示，当启动信号作用后，时钟信号在控制逻辑作用下，首先使寄存器的最高位 $D_3=1$ ，其余为 0，此数字量 1000 经 D/A 转换器转换成模拟量即 $V_0=8$ ，送到比较器输入端与被转换的模拟量 $V_{IN}=9$ 进行比较，控制逻辑根据比较器的输出进行判断。 $V_{IN} \geq V_0$ ，则保

留 $D_3=1$ ；再对下一位 D_2 进行比较，同样先使 $D_2=1$ ，与上一位 $D_3=1$ 一起即 1100 进入 D/A 转换器，转换为 $V_o=12$ 再进入比较器，与 $V_{IN}=9$ 比较，因 $V_{IN} < V_o$ ，则使 $D_2=0$ ；再下一位 D_1 位也是如此； $D_1=1$ 即 1010，经 D/A 转换为 $V_o=10$ ，再与 $V_{IN}=9$ 比较，因 $V_{IN} < V_o$ ，则使 $D_1=0$ ；最后一位 $D_0=1$ 即 1001 经 D/A 转换为 $V_o=9$ ，再与 $V_{IN}=9$ 比较，因 $V_{IN} \geq V_o$ ，保留 $D_0=1$ 。比较完毕，寄存器中的数字量 1001 即为模拟量 9 的转换结果，存在输出锁存器中等待输出。

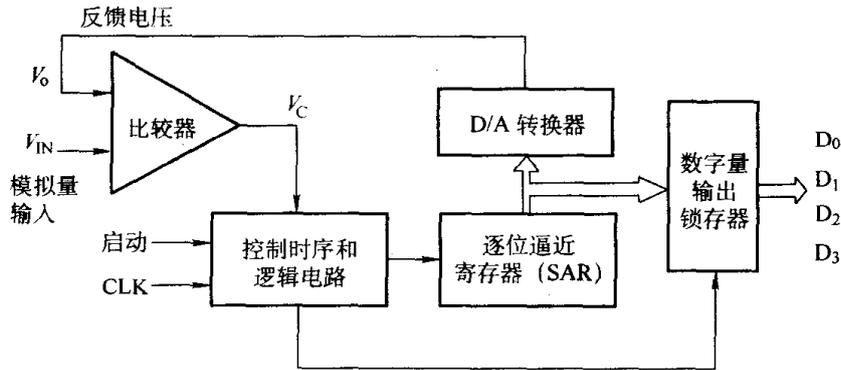


图 4-26 逐位逼近式 A/D 转换原理

一个 n 位 A/D 转换器的模数转换表达式为

$$B = \frac{V_{IN} - V_{R-}}{V_{R+} - V_{R-}} \times 2^n \quad (4-14)$$

式中， V_{R+} 、 V_{R-} 为基准电压源的正、负输入； V_{IN} 为待转换的输入模拟量； B 为转换后的输出数字量。当基准电压源确定之后， n 位 A/D 转换器的输出数字量 B 与要转换的输入模拟量 V_{IN} 成正比。

逐次逼近式 A/D 转换器的优点是精度较高、转换速度较快，而且转换时间是固定的，因而特别适合数据采集系统和控制系统的模拟量输入通道。它的缺点是抗干扰能力不够强。此类 A/D 转换器的常用品种有普通型 8bit 单路 ADC0801~ADC0805、8bit 8 路 ADC0808/ADC0809、8bit 16 路 ADC0816/ADC0817 等，混合集成高速型 12bit 单路 AD574A、AD1674、ADC803 等。

2. 双积分式 A/D 转换原理

双积分式 A/D 转换器的转换基础是测量两个时间：第一个时间是模拟电压向电容充电的固定时间，第二个时间是已知参考电压放电所需要的时间，模拟输入电压与参考电压的比值就是这两个时间值之比。图 4-27a 是这种 A/D 转换器的电路原理，图 4-27b 是这种 A/D 转换器的双积分原理。

双积分式 A/D 转换器的工作原理与过程说明如下。

电路中包括一个积分器，其输出电压反映了输入电压的积分时间。当输入电压为恒值时，积分器的输出以同输入电压值成正比的速率随时间不断地增加，即 $dv_o/dt = V_{IN}/RC$ 。因此，在“启动转换”信号控制下，模拟输入电压 V_{IN} 在固定时间内充 N 个计数脉冲，时间一到，控制逻辑就把模拟开关转换到与 V_{IN} 极性相反的参考电压 V_{REF} 上，使电容开始放电，放电期间计数器计数脉冲的多少反映了放电时间的长短，从而决定了模拟输入电压的大小。输入电压高则放电时间长，当比较器判定电容放电完毕时，输出信号使计数器停止计数，并由控制逻辑发出“转换结束”信号，计数器的值反映了输入电压 V_{IN} 在固定积分时间内的平均值。这

是因为，较大的输入电压（图 4-27b 中的虚线部分）将在固定时间 T_0 内引起更大的积分增量 Δv_0 。这样一来，直线下降部分所花费的时间 T_1 就相应变长，即计数值增大。因此，双积分式 A/D 转换器是通过测量时间间隔 T_1 来测量电压值的。由此可见，一个快速时钟和一个多位计数器就能很容易地使这种方法获得很高的分辨率。

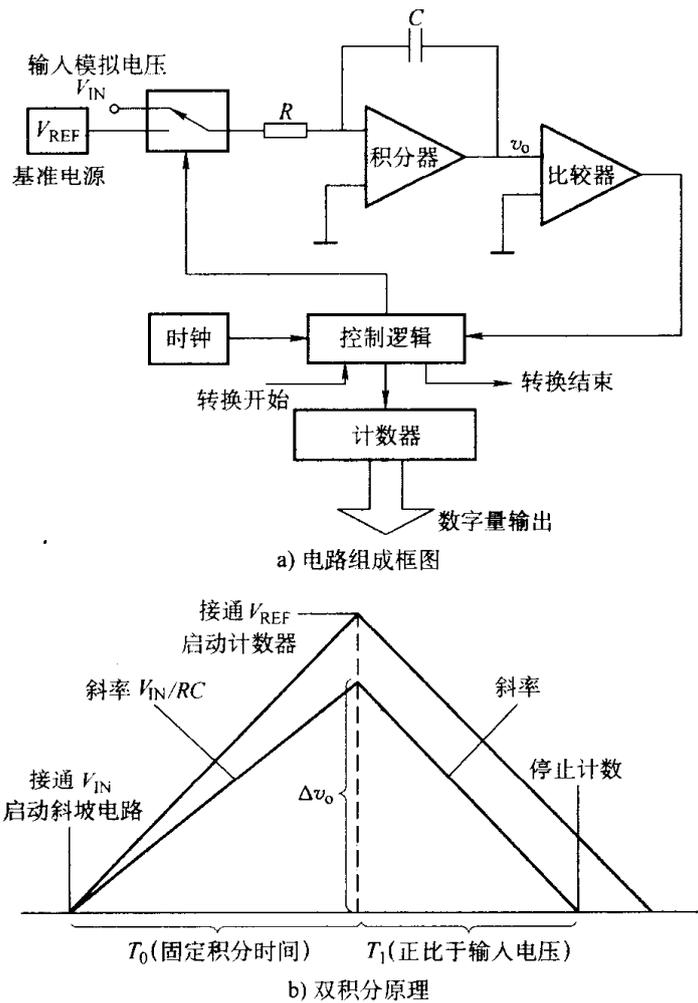


图 4-27 双积分式 A/D 转换原理

双积分式 A/D 转换器的电路具有固有的抑制噪声能力，而且抑制能力随着噪声频率的增加，线性地增强。这是因为在采样时间 T 内噪声已变化了几个周波，它对积分的贡献必然包括其正、负两部分，通过积分部分的作用使得噪声的影响趋于平均化，即噪声抑制能力随着噪声频率的增加而增加。

双积分式 A/D 转换器的优点是消除干扰和电源噪声的能力强、精度高；缺点是转换速度较慢。因此在信号变化缓慢、对模拟量输入速率要求较低、转换精度要求高，且现场干扰较严重的情况下，可以采用这种 A/D 转换器。

此种 A/D 转换器的常用品种有输出为 3 位半 BCD 码（二进制编码的十进制数）的 ICL7107、MC14433，输出为 4 位半 BCD 码的 ICL7135 等。

3. Σ - Δ 式 A/D 转换器

Σ - Δ 式 A/D 转换器以很低的采样分辨率（1 位）和很高的采样速率将模拟信号数字化，通

过使用采样、噪声整形和数字滤波等方法增加有效分辨率，然后对 A/D 转换器输出进行采样抽取处理以降低有效采样速率。

Σ - Δ 式 A/D 转换器的电路由非常简单的模拟电路（一个比较器、一个开关、一个或多个积分器及模拟求和电路）和复杂的数字信号处理电路构成。一个一阶 Σ - Δ A/D 转换电路的原理如图 4-28 所示，所谓一阶 Σ - Δ A/D 转换器是只采用一个积分器，它的工作原理说明如下。

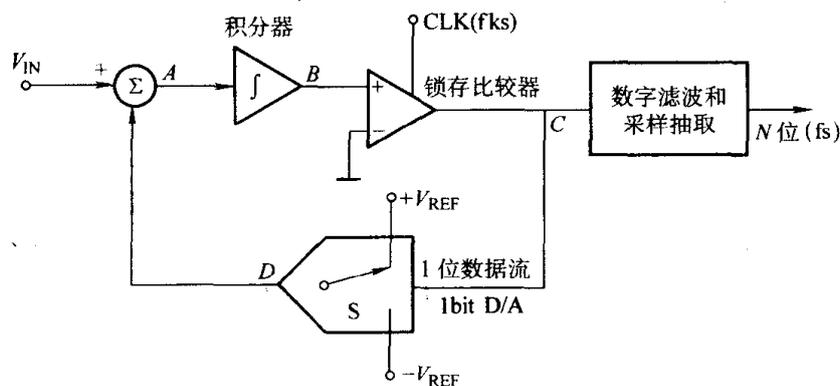


图 4-28 一阶 Σ - Δ A/D 转换器原理图

在“启动转换”信号控制下，先对积分器和锁存比较器初始化，假定初始状态为积分器输出为负，锁存比较器输出为“0”，开关 S 把 $+V_{REF}$ 接到积分器输入端，积分器对输入电压 V_{IN} 和 V_{REF} 之和进行积分，积分器输出从负向正线性增加。当积分器输出为 0 时，比较器输出要翻转，但这个翻转过程必须在时钟控制下进行，因而锁存比较器的状态变化比积分器输出为 0 时要晚，积分器输出大于 0。锁存比较器翻转后，开关 S 接通 $-V_{REF}$ ，使积分器反向积分，积分器输出由正向负方向变化，同样在积分器为 0 时也产生类似的翻转，使积分器回到前一个状态。不断重复上述过程，在锁存比较器输出端得到一串脉冲。图 4-29 所示为 Σ - Δ 调制器波形。

很明显，当 $V_{IN}=0$ 时，得到的输出为“0”、“1”相间的脉冲，若 $V_{IN}>0$ ，输出脉冲中“1”的个数要多于“0”的个数，若 $V_{IN}<0$ 。输出脉冲“0”的个数则多于“1”的个数。

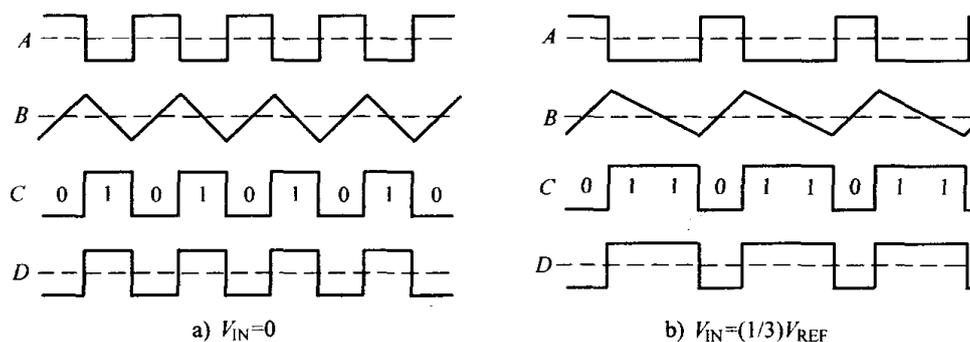


图 4-29 一阶 Σ - Δ 调制器波形图

为了得到很高的精度， Σ - Δ 调制器的采样频率（即时钟脉冲）要远高于信号变化频率（过采样），而实际输出的数字量采用 $1/M$ 采样抽取，即在锁存比较器输出的 M 个脉冲内取相同位置的脉冲作为数字量的一位（一般为首脉冲）。在采样周期内顺序获得的一串数据即为转化

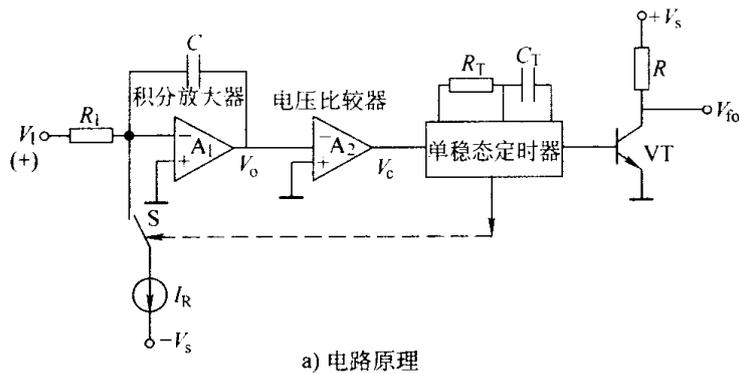
的数字量。

Σ - Δ /D 转换器具有双积分 A/D 转换器所具有的对串模干扰的抑制能力，转换精度高，对组成电路的元件要求较低，比较器的失调和漂移电压不会影响转换精度，目前在数字化测控系统中得到了广泛应用。

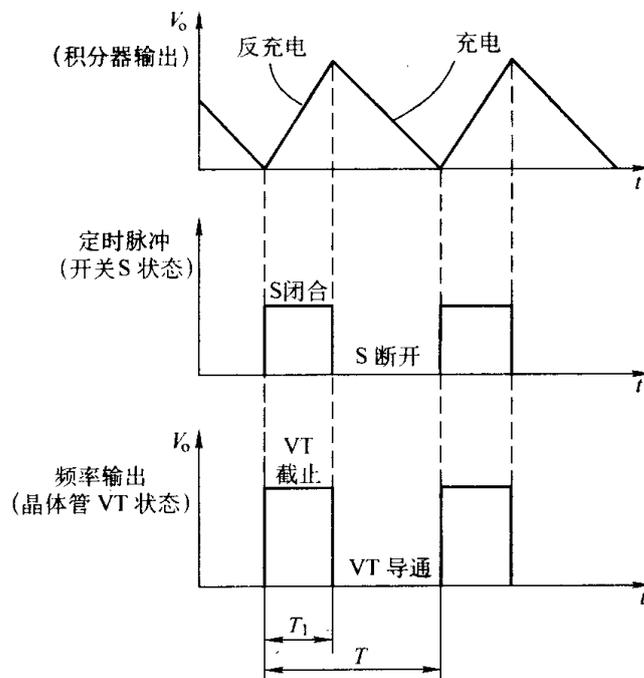
4. 电压/频率式 A/D 转换原理

电压/频率式 A/D 转换器简称 V/F 转换器，它是一种把模拟电压信号转换成频率信号的器件。实现 V/F 转换的方法很多，现以常见的电荷平衡 V/F 转换法说明其转换原理，如图 4-30a、b 所示。

在图 4-30 中， A_1 是积分放大器， A_2 为零电压比较器，恒流源 I_R 和开关 S 构成 A_1 的反充电网络，开关 S 由单稳态定时器触发控制。当积分放大器 A_1 的输出电压 V_o 下降到零时，零电压比较器 A_2 输出跳变，则触发单稳态定时器，即产生暂态时间为 T_1 的定时脉冲，并使开关 S 闭合；同时使晶体管 VT 截止，频率输出端 V_{fo} 输出高电平。



a) 电路原理



b) 波形

图 4-30 电荷平衡式 V/F 转换原理

在开关 S 闭合期间，恒流源 I_R 被接入积分器的“-”输入端。由于电路是按 $I_R > V_{lmax} / R_1$

设计的, 故此时电容 C 被反向充电, 充电电流为 $I_R - V_1/R_1$, 则积分器 A_1 输出电压 V_o 从零伏起线性上升。当定时 T_1 时间结束, 定时器恢复稳态, 使开关 S 断开, 反向充电停止, 同时使晶体管 VT 导通, V_{fo} 端输出低电平。

开关 S 断开后, 正输入电压 V_1 开始对电容 C 正向充电, 其充电电流为 V_1/R_1 , 则积分放大器 A_1 输出电压 V_o 开始线性下降。当 $V_o=0$ 时, 比较器 A_2 输出再次跳变, 又使单稳态定时器产生 T_1 时间的定时脉冲而控制开关 S 再次闭合, A_1 再次反向充电, 同时 V_{fo} 端又输出高电平。如此反复下去, 就会在积分器 A_1 输出端 V_o 、单稳态定时器脉冲输出端和频率输出端 V_{fo} 端产生如图 4-30b 所示的波形, 其波形的周期为 T 。

根据反向充电电荷量和正向充电电荷量相等的电荷平衡原理, 可得

$$(I_R - V_1/R_1)T_1 = V_1/R_1 \cdot (T - T_1) \quad (4-15)$$

整理得

$$T = I_R R_1 T_1 / V_1 \quad (4-16)$$

则 V_{fo} 端输出的电压频率为

$$f_o = 1/T = V_1 / (I_R R_1 T_1) \quad (4-17)$$

式中, f_o 就是由 V_1 转换而来的输出频率, 两者成线性比例关系。由式 (4-17) 可见, 要精确地实现 V/F 变换, 要求 I_R 、 R_1 和 T_1 应准确稳定。积分电容 C 虽没有出现在式 (4-17) 中, 但它的漏电流将会影响到充电电流 V_1/R_1 , 从而影响转换精度。为此应选择漏电流小的电容。

采用 V/F 容易实现信号隔离, 抗干扰能力较强, 占用计算机 I/O 端口较少, 因而在测控系统中有一定的应用。此种 V/F 转换器的常用品种有 VFC32、LM131/LM231/LM331、AD650 及 AD651 等。

5. A/D 转换器的性能指标

(1) 分辨率

分辨率指 A/D 转换器对微小输入信号变化的敏感程度。分辨率越高, 转换时对输入量微小变化的反应越灵敏。通常用数字量的位数来表示, 如 8bit、10bit、12bit 等。分辨率为 n , 表示它可以对满刻度的 $1/2^n$ 的变化量作出反应, 即

$$\text{分辨率} = \text{满刻度值} / 2^n$$

(2) 转换精度

A/D 转换器的转换精度可以用绝对误差和相对误差来表示。

所谓绝对误差是指对应于一个给定数字量 A/D 转换器的误差, 其误差的大小由实际模拟量输入值和理论值之差来度量。绝对误差包括增益误差、零点误差和非线性误差等。

相对误差是指绝对误差与满刻度值之比, 一般用百分数来表示, 对 A/D 转换器常用最低有效值的位数 (LSB, Least Significant Bit) 表示, $1 \text{ LSB} = 1/2^n$ 。

例如, 对于一个 8bit 0~5V 的 A/D 转换器, 如果其相对误差为 $\pm 1 \text{ LSB}$, 则其绝对误差为 $\pm 19.5 \text{ mV}$, 相对百分误差为 0.39%。一般来说, 位数 n 越大, 其相对误差 (或绝对误差) 越小。

(3) 转换时间

A/D 转换器完成一次转换所需的时间称为转换时间, 如逐位逼近式 A/D 转换器的转换时间为微秒级、双积分式 A/D 转换器的转换时间为毫秒级。

4.3.5 逐位逼近式 A/D 转换器及其接口

1. 8bit A/D 转换器及其接口

8bit A/D 转换器的种类很多, 下面以其中一种典型的芯片——ADC0809 为例进行说明。

(1) ADC0809 芯片

ADC0809 为 8bit 逐位逼近式 A/D 转换器, 其分辨率为 $1/2^8 \approx 0.39\%$, 模拟电压转换范围是 $0 \sim +5V$, 标准转换时间为 $100\mu s$, 采用 28 脚双立直插式封装, 其内部结构及引脚如图 4-31 所示。

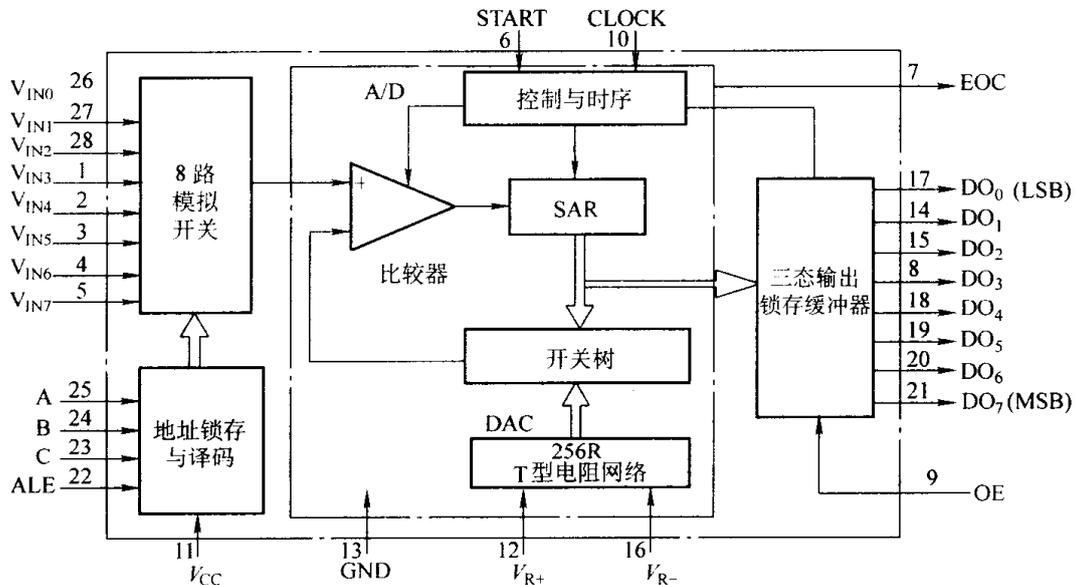


图 4-31 ADC0809 内部结构及引脚

ADC0809 在逐位逼近式 A/D 转换器的原理基础上, 增加了一个 8 路模拟开关、一个用来选择通道的地址锁存译码电路和一个三态输出锁存器, 其引脚功能如下。

$V_{IN0} \sim V_{IN7}$: 8 路模拟量输入端, 允许 8 路模拟量分时输入。

ALE: 地址锁存允许信号, 输入, 高电平有效, 要求信号宽度为 $100 \sim 200ns$, 上升沿时锁存 3 位通道选通信号。

A、B、C: 3 位地址线, 即模拟量通道选择线。ALE 为高电平时, 地址译码与通道选择相对应。例如, $A=0$ 、 $B=0$ 、 $C=0$ 时, 选通输入通道 V_{IN0} ; $A=0$ 、 $B=0$ 、 $C=1$ 时, 选通输入通道 V_{IN4} ; $A=1$ 、 $B=1$ 、 $C=1$ 时, 选通输入通道 V_{IN7} 。

START: 启动 A/D 转换信号, 输入, 高电平有效, 要求信号宽度为 $100 \sim 200ns$ 。上升沿时将转换器内部清零, 下降沿时启动 A/D 转换。

EOC: 转换结束信号, 输出, 高电平有效。平时 EOC 为高电平, A/D 转换期间为低电平, A/D 转换结束后又变为高电平。EOC 可用作向主机申请中断的信号, 或供主机查询 A/D 转换是否结束的信号。

OE: 输出允许信号, 输入, 高电平有效。该信号用来打开三态输出缓冲器, 将 A/D 转换得到的 8bit 数字量送到数据总线上。

$DO_0 \sim DO_7$: 8bit 数字量输出, DO_0 为最低位, DO_7 为最高位。由于有三态输出锁存, 可与主机数据总线直接相连。

CLOCK: 外部时钟脉冲输入端。当脉冲频率为 640kHz 时, A/D 转换时间为 100 μ s。

V_{R+} 、 V_{R-} : 基准电压源正端、负端。取决于被转换的模拟电压范围, 通常 $V_{R+}=+5V$ (DC), $V_{R-}=0V$ (DC)。

V_{CC} : 工作电源, +5V (DC)。

GND: 电源地。

了解 ADC0809 的内部转换时序非常重要, 这是设计分析硬件与软件时的主要依据。如图 4-32 所示, 具转换过程表述如下: 首先 ALE 的上升沿将地址代码锁存、译码后选通模拟开关中的某一路, 使该路模拟量进入到 A/D 转换器中。同时 START 的上升沿将转换器内部清零, 下降沿启动 A/D 转换, 即在时钟的作用下, 逐位逼近过程开始, 转换结束信号 EOC 即变为低电平, 当转换结束后, EOC 恢复高电平, 此时, 如果对输出允许 OE 输入一高电平命令, 则可读出数据。需要特别注意的是, START 信号下降沿启动 A/D 转换后, EOC 不会立即变为低电平, 在短时间内仍为高电平, 在目前计算机速度很快的情况下, 如此时就通过读 EOC 信号为高判断转换已结束是不正确的。一种解决的办法是在软件编程时, 启动转换命令发出后, 加入空指令进行延时, 再读 EOC 信号。

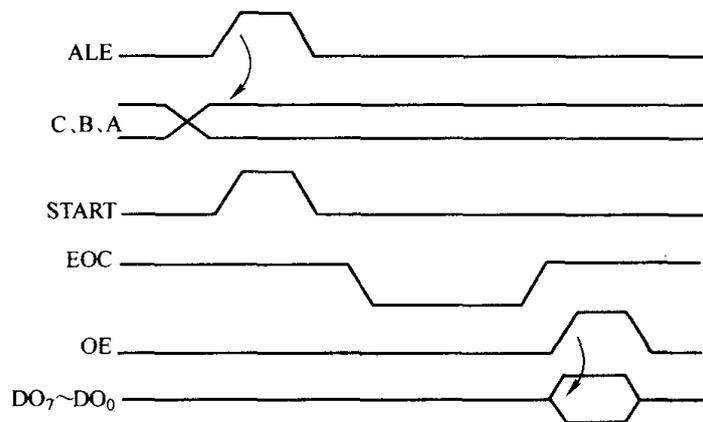


图 4-32 ADC0809 的转换时序

上述过程说明, 判断 A/D 转换器是否完成一次转换, 可以依据转换结束信号 EOC 电平的高低, 或者根据输入时钟频率计算出转换芯片的转换时间。因此, 读取 A/D 转换数就可以采用查询、定时、中断及 DMA 等多种方式。

另外, ADC0809 芯片输出端具有可控的缓冲锁存门电路, 易于直接与主机进行接口; 还有一类芯片内部没有缓冲锁存门电路, 不能直接与主机连接。这样, 在 A/D 转换器与主机之间的数据线连接上也出现了直接连接、通过 8255 或锁存器间接连接的几种情形。

(2) 接口电路

A/D 转换器的接口电路主要是解决主机如何分时采集多路模拟量输入信号的, 即主机如何启动 A/D 转换, 如何判断 A/D 完成一次转换, 如何读入并存放转换结果的。下面介绍两种典型的接口电路。

① 查询方式读 A/D 转换数。图 4-33 为采用程序查询方式的 8 路 8bit A/D 转换接口电路, 由 PC 总线、ADC0809, 以及 138 译码器、74LS02 或非门与 74LS126 三态缓冲器组成。在图 4-33 中, 假定启动转换的板址 PA=01000000B, 每一路的口地址分别为 000~111B, 故 8 路转换地址为 40H~47H。

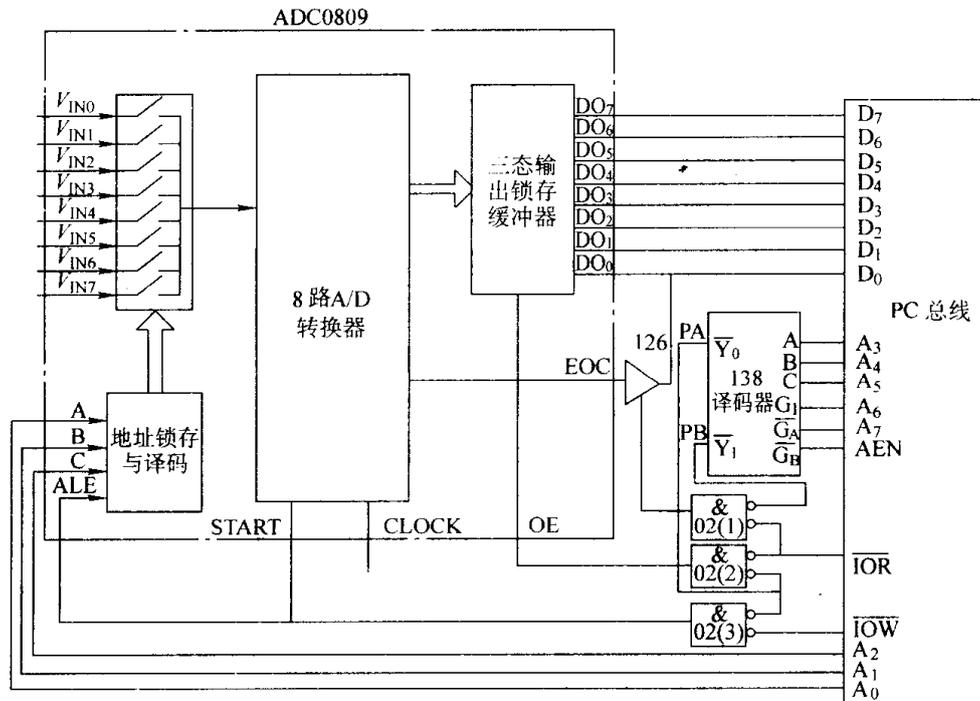


图 4-33 查询方式读 A/D 转换数电路组成

接口程序如下：

```

MOV    BX, BUFF           ; 置采样数据区首址
MOV    CX, 08H           ; 8 路输入
START: OUT    PA, AL       ; 启动 A/D 转换
REOC:  IN     AL, PB       ; 读 EOC
      RCR    AL, 01        ; 判断 EOC
      JNC   REOC          ; 若 EOC=0, 继续查询
      IN    AL, PA         ; 若 EOC=1, 读 A/D 转换数
      MOV   [BX], AL       ; 存 A/D 转换数
      INC  BX              ; 存 A/D 转换数地址加 1
      INC  PA              ; 接口地址加 1
      LOOP START          ; 循环

```

下面分析说明启动转换过程。首先主机执行一条启动转换第 0 路的输出指令，即把 AL 中的数据送到地址为 PA 的接口电路中，此时 AL 中的内容无关紧要，而地址 PA=40H 使 138 译码器的 \overline{Y}_0 输出一个低电平，连同 OUT 输出指令造成 \overline{IOW} 为低电平，从而使非与门 02(3) 产生脉冲信号到引脚 ALE 和 START，ALE 的上升沿将通道地址代码 000 锁存并进行译码，选通模拟开关中的第 0 路 V_{IN0} ，使该路模拟量进入到 A/D 转换器中；同时 START 的上升沿将 ADC0809 中的逐位逼近寄存器清零，下降沿启动 A/D 转换，即在时钟的作用下，逐位逼近的模数转换过程开始。

接着，主机查询转换结束信号 EOC 的状态，通过执行输入指令，即是把地址为 PB 的转换接口电路的数据读入 AL 中，此时地址 PB=01001000B (48H)，使 138 译码器的 \overline{Y}_1 端输出一个低电平，连同 IN 输入指令造成 \overline{IOR} 端为低电平，从而使非与门 02(1) 产生脉冲信号并

究竟采用何种接口方式要根据 A/D 转换器芯片而定。

2. 高于 8bit 的 A/D 转换器及其接口

8bit A/D 转换器的分辨率约为 0.39%，转换精度在 0.4% 以下，这对一些精度要求比较高的控制系统而言是不够的，因此要采用更多位的 A/D 转换器，如 10bit、12bit、14bit 等 A/D 转换器。下面以 AD574A 为例介绍 12bit A/D 转换器及其接口电路。

(1) AD574A 芯片

AD574A 是一种高性能的 12bit 逐位逼近式 A/D 转换器，分辨率为 $1/2^{12} \approx 0.024\%$ ，转换时间为 $25\mu s$ ，适合于在高精度快速采样系统中使用。如图 4-35 所示，其内部结构大体与 ADC0809 类似，由 12bit A/D 转换器、控制逻辑、三态输出锁存缓冲器与 10V 基准电压源构成，可以直接与主机数据总线连接，但只能输入一路模拟量。AD574A 也采用 28 脚双立直插式封装，各引脚功能如下。

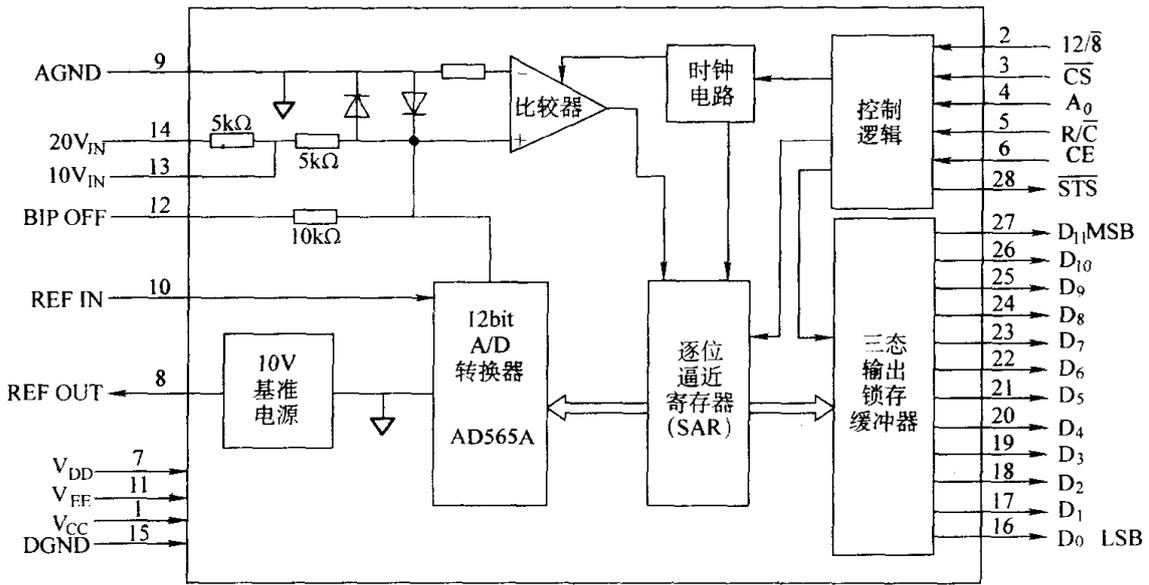


图 4-35 AD574A 原理框图及引脚

$10V_{IN}$ 、 $20V_{IN}$ 、BIP OFF：模拟电压信号输入端。单极性应用时，将 BIP OFF 接 0V，双极性时接 $10V$ 。量程可以是 $10V$ ，也可以是 $20V$ 。输入信号在 $10V$ 范围内变化时，将输入信号接至 $10V_{IN}$ ；在 $20V$ 范围内变化时，接至 $20V_{IN}$ 。模拟输入信号的几种接法如表 4-2 所列，相应电路如图 4-36 所示。

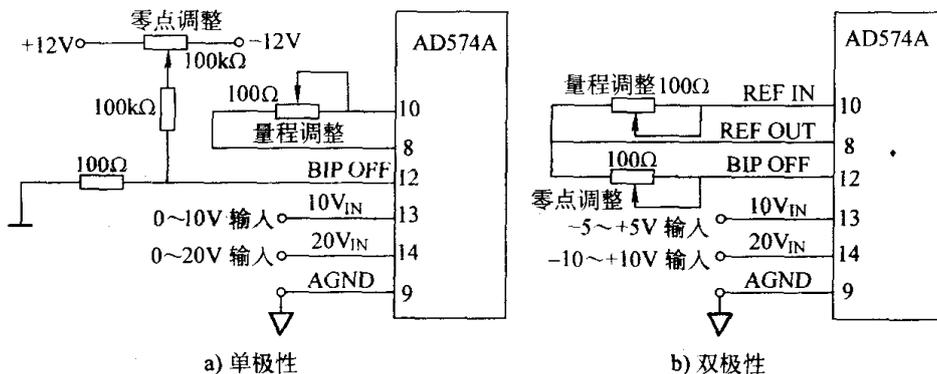


图 4-36 AD574A 的输入信号连接方法

V_{CC} : 工作电源正端, +12V (DC) 或+15V (DC)。

V_{EE} : 工作电源负端, -12V (DC) 或-15V (DC)。

V_{DD} : 逻辑电源端, +5V (DC)。虽然使用的工作电源为±12V (DC) 或±15V (DC), 但数字量输出及控制信号的逻辑电平仍可直接与 TTL 兼容。

DGND、AGND: 数字地、模拟地。

REF OUT: 基准电压源输出端, 芯片内部基准电压源为+10 (1±1%) V。

REF IN: 基准电压源输入端, 如果 REF OUT 通过电阻接至 REF IN, 则用来调量程。

\overline{STS} : 转换结束信号, 高电平表示正在转换, 低电平表示已转换完毕。

$D_0 \sim D_{11}$: 12 位输出数据线, 三态输出锁存, 可与主机数据线直接相连。

CE: 片能用信号, 输入, 高电平有效。

\overline{CS} : 片选信号, 输入, 低电平有效。

R/\overline{C} : 读/转换信号, 输入, 高电平为读 A/D 转换数据, 低电平为启动 A/D 转换。

$12/\overline{8}$: 数据输出方式选择信号, 输入, 高电平时输出 12bit 数据, 低电平时与 A_0 信号配合输出高 8 位或低 4 位数据。 $12/\overline{8}$ 不能用于 TTL 电平控制, 必须直接接至+5V (引脚 1) 或数字地 (引脚 15)。

A_0 : 字节信号, 在转换状态时, A_0 为低电平可使 AD574A 产生 12bit 转换, A_0 为高电平可使 AD574A 产生 8bit 转换; 在读数状态时, 如果 $12/\overline{8}$ 为低电平, A_0 为低电平时, 则输出高 8 位数, 而 A_0 为高电平时, 则输出低 4 位数; 如果 $12/\overline{8}$ 为高电平, 则 A_0 的状态不起作用。

CE、 \overline{CS} 、 R/\overline{C} 、 $12/\overline{8}$ 、 A_0 各控制信号的组合作用, 如表 4-3 所列。

表 4-2 模拟输入信号的几种接法

引脚	单极性	双极性
BIP OFF	0V	10V
10V _{IN}	0V~10V	-5~+5V
20V _{IN}	0V~20V	-10~+10V

表 4-3 AD574A 控制信号的作用

CE	\overline{CS}	R/\overline{C}	$12/\overline{8}$	A_0	功能
0	×	×	×	×	无
×	1	×	×	×	无
1	0	0	×	0	启动 12bit 转换
1	0	0	×	1	启动 8bit 转换
1	0	1	+5V	×	输出 12bit 数据
1	0	1	接地	0	输出高 8bit 数据
1	0	1	接地	1	输出低 4bit 数据

注: ×表示 1 或 0 均可。

AD574A 的启动和读操作时序如图 4-37 所示。

(2) 接口电路

12bit A/D 转换器 AD574A 与 PC 总线的接口有多种方式: 既可以与 PC 总线的 16bit 数据总线直接相连, 构成简单的 12bit 数据采集系统; 也可以只占用 PC 总线的低 8 位数据总线, 将转换后的 12bit 数字量分两次读入主机。同样, 在 A/D 转换器与 PC 总线之间的数据传送上也可以使用程序查询、软件定时或中断控制等多种方法。由于 AD574A 的转换速度很高, 一般多采用查询或定时方式读 A/D 转换数据。

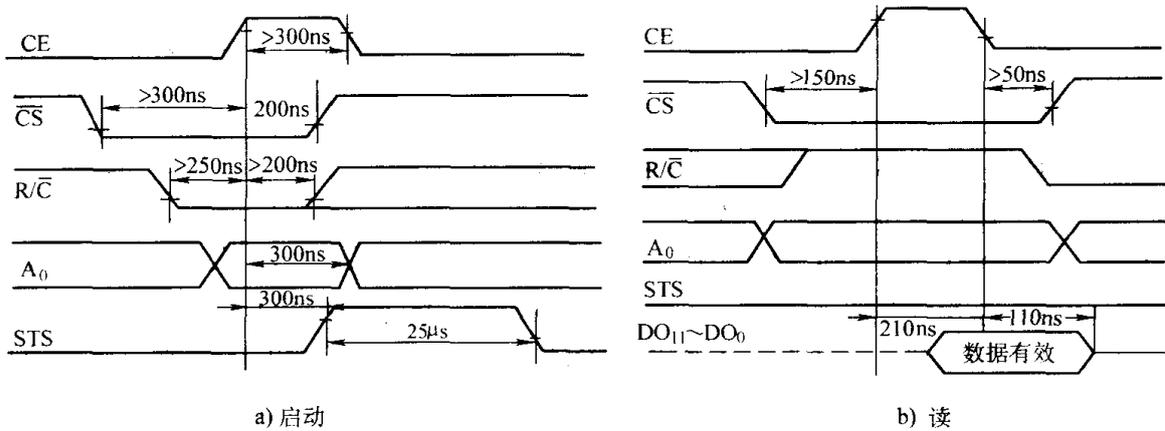


图 4-37 AD574A 工作时序

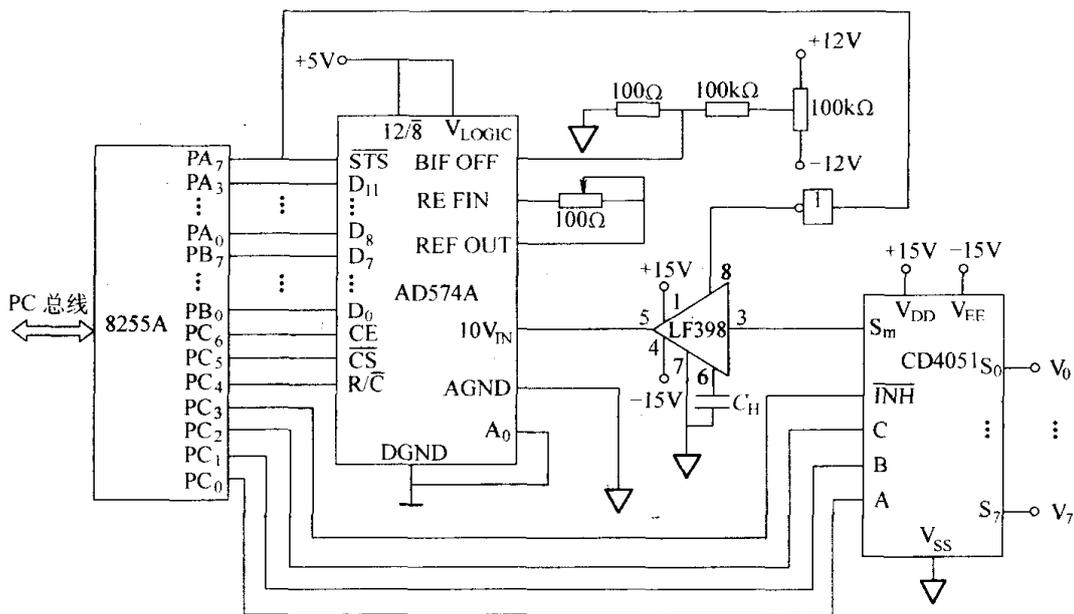


图 4-38 8 路 12bit A/D 转换接口电路

图 4-38 是一种使用 AD574A 并利用 8255A 扩展口构成的 8 路 12bit A/D 转换接口电路，其采集数据的过程如下。

(1) 通道选择：将模拟量输入通道号写入 8255A 的端口 C 低 4 位 (PC₃~PC₀)，可以依次选通 8 路通道。

(2) 采样保持控制：把 AD574A 的 \overline{STS} 信号通过反相器连到 LF398 的信号采样保持端，当 AD574A 未转换期间或转换结束时 STS=0，使 LF398 处于采样状态，当 AD574A 转换期间 STS=1，使 LF398 处于保持状态。

(3) 启动 AD574A 进行 A/D 转换：通过 8255A 的端口 PC₆~PC₄ 输出控制信号启动 AD574A。

(4) 查询 AD574A 是否转换结束：读 8255A 的端口 A，查问 STS 是否已由高电平变为低电平。

(5) 读取转换结果：若 \overline{STS} 已由高电平变为低电平，则读 8255A 端口 A、B，便可得到 12bit 转换结果。

设 8255A 的 A、B、C 端口与控制寄存器的地址为 2C0H~2C3H，主过程已对 8255A 初始化，且已装填 DS、ES（两者段基址相同），采样值存入数据段中的采样值缓冲区 BUF，另定义一个 8bit 内存单元 BUF1。该过程的数据采集程序框图如图 4-39 所示，数据采集程序如下：

```

AD574A  PROC  NEAR                ; 过程定义伪指令
          MOV  CX, 8                ; 计数器初始
          CLD                        ; 标志位 DF 清零
          MOV  AL, 00000000B
          MOV  BUF1, AL             ; CE=0,  $\overline{CS}$ =0,  $R/\overline{C}$ =0,  $\overline{INH}$ =C=B=A=0,
                                   ; 控制信号初始化, 通道号初始化
          LEA  BX, BUF              ; 置采样缓冲区首址
NEXTCH:  MOV  DX, 2C2H              ; 8255A 的 PC 口址
          MOV  AL, BUF1
          OUT  DX, AL               ; 送 PC 口控制信号与通道号
          NOP
          NOP
          OR   AL, 01000000B        ; CE=1
    
```

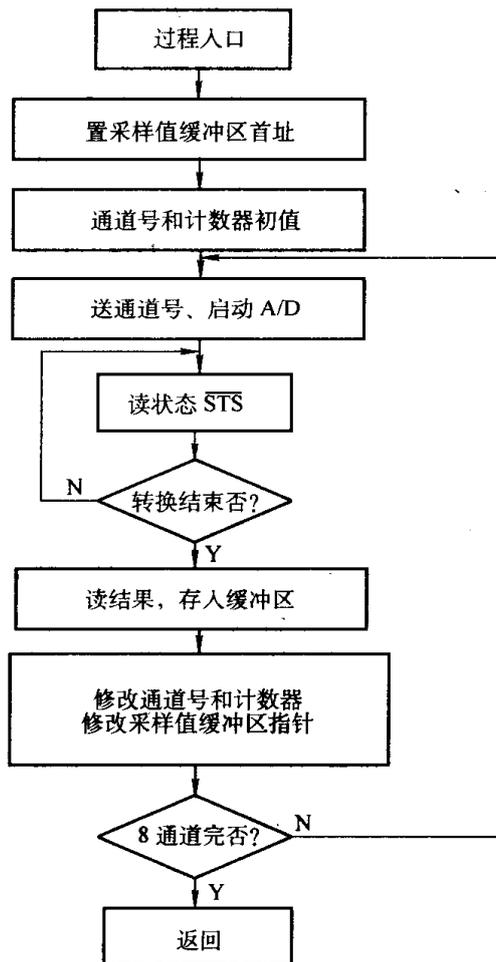


图 4-39 8 路数据采集程序框图

```

    OUT    DX, AL          ; 启动 A/D
    AND    AL, 10111111B   ; CE=0
    OUT    DX, AL
    MOV    DX, 2C0H       ; 8255A 的 PA 口地址
POLLING: IN    AL, DX
    TEST   AL, 80H
    JNZ    POLLING        ; 测试  $\overline{STS}$ 
    MOV    AL, BUF1
    OR     AL, 0101000B    ;  $R/\overline{C}=1$ 
    MOV    DX, 2C2H
    OUT    DX, AL         ; 输出 12bit 转换数据到 8255A
    MOV    DX, 2C0H
    IN     AL, DX         ; 读 8255A 的 PA 口
    AND    AL, 0FH
    MOV    AH, AL         ; 保留 PA 口的低 4 位 (12bit 数据中的高 4 位)
    INC    DX
    IN     AL, DX         ; 读 8255A 的 PB 口 (12bit 数据中的低 8 位)
    STOSW
    ; 12bit 数存入内存, 自动修改采样缓冲区指针
    INC    BUF1          ; 修改通道号
    LOOP   NEXTCH        ; 采集下一通道, 直到第 8 路
    MOV    AL, 00111000B   ; CE=0,  $\overline{CS}=1$ ,  $R/\overline{C}=1$ 
    MOV    DX, 2C2H
    OUT    DX, AL         ; 停止 AD574A 工作
    RET
AD574A  ENDP

```

4.4 数字量输入、输出通道

在计算机控制系统中,除了要处理模拟量信号外,还要处理另一种数字信号,包括开关信号、脉冲信号。它们是以二进制的逻辑“1”、“0”或电平的高、低出现的,如开关触点的闭合和断开、指示灯的亮和灭、继电器或接触器的吸合和释放、电动机的起动和停止、晶闸管的通和断、阀门的打开和关闭、仪器仪表的 BCD 码以及脉冲信号的计数和定时等。

计算机控制系统的输入数字信号来自于现场的信号传感器,输出数字信号又送回现场设备中。因此,现场的电磁干扰会通过数字量输入、输出通道串入到计算机系统中,这就需要采用通道隔离技术。数字信号最常用的隔离方法是光电耦合隔离技术,相关内容可参看本书第 3 章。

4.4.1 数字量输入通道

数字量输入通道又可称为开关量输入通道,该通道将双值逻辑的开关量(数字量)变换

为计算机能够接收的数字量，它的结构形式如图 4-40 所示。

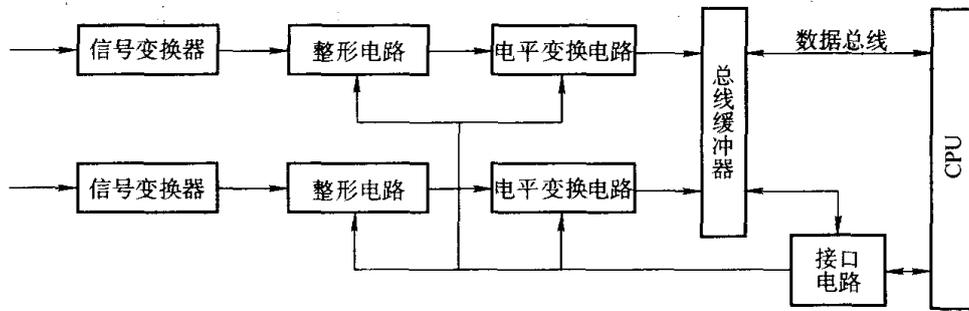


图 4-40 开关量输入通道结构框图

典型的开关量输入通道通常由以下几部分组成。

- ① 信号变换器：将过程的非电量开关量转换为电压或电流的双值逻辑值。
- ② 整形电路：将混有毛刺之类干扰的输入双值逻辑信号或其信号前后沿不符合要求的输入信号整形为接近理想状态的方波或矩形波，而后再根据系统要求变换为相应形状的脉冲信号。
- ③ 电平变换电路：将输入的双值逻辑电平转换为与 CPU 兼容的逻辑电平。
- ④ 总线缓冲器：暂存数字量信息并实现与 CPU 数据总线的连接。
- ⑤ 接口电路：协调通道的同步工作，向 CPU 传递状态信息并控制开关量到 CPU 的输入。

1. 开关量（数字量）形式及变换

开关量（数字量）大致可分为三种形式，即机械有触点开关量、电子无触点开关量和非电量开关量。不同的开关量要采用不同的变换方法。

(1) 机械有触点开关量

机械有触点开关量是工程中遇到的最典型的开关量，它由机械式开关（如继电器、接触器、开关、行程开关、阀门、按钮等）产生，有常开、常闭两种方式。机械有触点开关量的显著特点是无源，开闭时会产生抖动。同时这类开关通常安装在生产现场，在信号变换时应采取隔离措施。

机械有触点开关量的变换方法通常有以下三种。

- ① 控制系统自带电源方式。这种方法一般用于开关安装位置离计算机控制装置较近的场所，供电电源为直流 24V 以下，常用电路有串联和并联两种（见图 4-41）形式。对于并联电路，触点闭合时输出 V_o 为高电平，触点打开时 V_o 为低电平。串联电路正好与之相反。

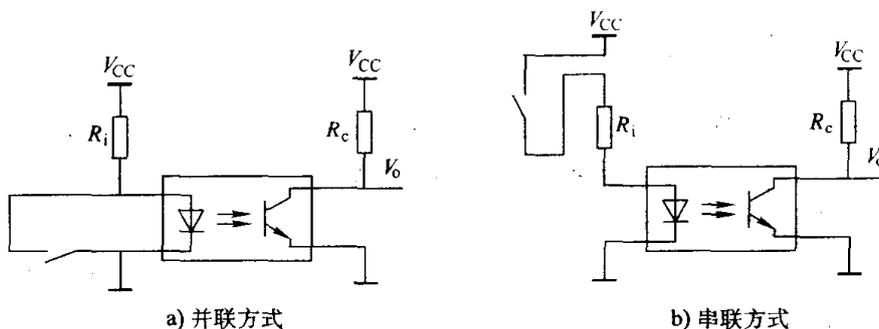


图 4-41 自带电源的开关量变换电路

② 外接电源方式。它适合于开关安装在离控制设备较远位置的场合。外接电源可采用直流或交流形式。采用直流电源形式的变换电路如图 4-42 所示。其中二极管为保护元件，对 R_i 的选择应考虑功率因数。

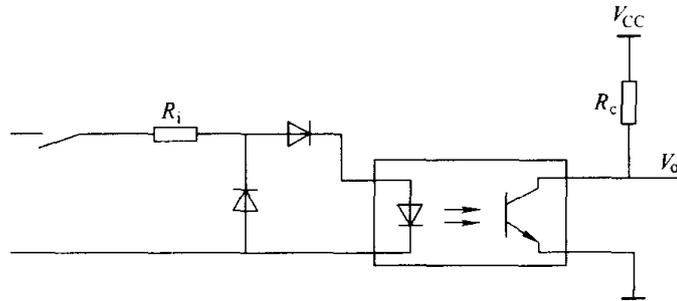


图 4-42 外接直流电源开关量变换电路

外接电源采用交流电路时一般使用变压器，将高压交流电（220V 或 110V）变为低压。这种电路的响应速度较慢，因而使用较少。

③ 恒流源方式。这种方式用于抗干扰能力要求高、传输距离较远的场合。电流一般取 0~10mA，即触点闭合时输出电流为 10mA，触点打开时输出电流为 0。

(2) 电子无触点开关量

电子无触点开关量指电子开关（如固态继电器、功率电子器件、模拟开关等）产生的开关量。由于无触点开关通常没有辅助机构，其开关状态与主电路没有隔离，因而隔离电路是它的信号变换电路的重要组成部分。

无触点开关量的采集可由两种方式实现。第一种方式与有触点开关处理方法相同，即把无触点开关当作有触点开关，按图 4-43 方式连接电路即可。需要注意的是连接极性不能随意更换。

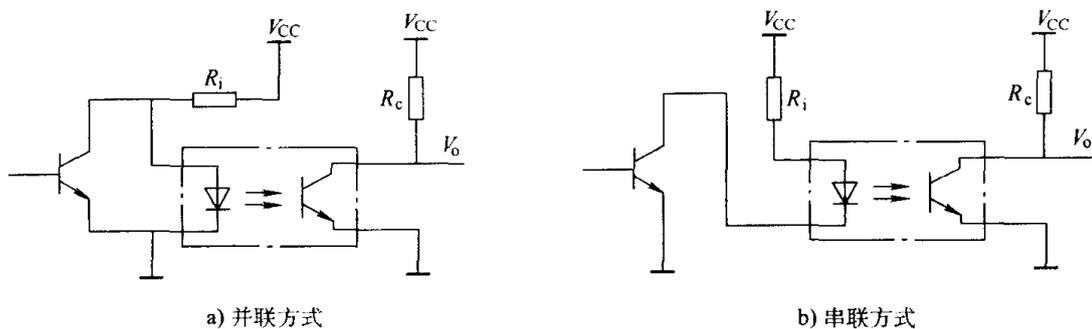


图 4-43 无触点开关变换电路

电子无触点开关量变换的第二种方法是从功率开关的负载电路采样法。它的原理电路框图如图 4-44 所示。这种方法直接反映负载电路工作状态，而对开关状态的采样是间接的。

(3) 非电量开关量（数字量）

通过采用磁、光、声等方式反映过程状态，在许多控制领域中得到广泛应用。这种非电量开关量（数字量）需要通过电量转换后才能以电的形式输出。实现非电量开关量（数字量）的信号变换电路由非电量/电量变换、放大（或检波）电路、光电隔离电路等组成，如图 4-45

所示。

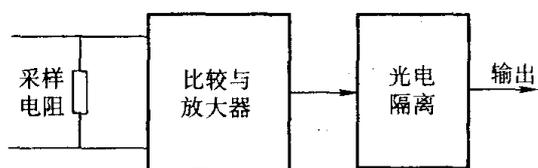


图 4-44 开关量采样变换原理电路框图

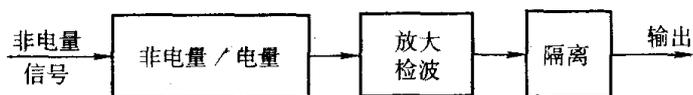


图 4-45 非电量开关量变换电路结构

在图 4-45 中，非电量/电量变换一般采用磁敏、光敏、声敏等元件，它将磁、光、声的变化以电压或电流形式输出。由于敏感元件输出信号较弱，输出电信号不一定是逻辑量（如可能是交流电压），因此对信号要进行放大和检波后才能变成具有一定驱动能力的逻辑电信号。隔离电路根据控制系统工作环境及信号拾取方式决定是否采用。

2. 整形与电平变换

各种过程开关量经信号变换后转换成逻辑电信号或脉冲信号，但这种信号脉冲宽度、脉冲波形形状、脉冲前后沿陡度及信号电平可能不很理想，通常需进行波形整形及电平变换才能输入到计算机。

(1) 波形整形

波形整形的目的是使逻辑信号变为较理想的电信号，并提高抗干扰能力。波形整形包括触点消抖、脉冲定宽、去除尖峰毛刺等。

1) 触点消抖。在机械有触点开关中，当触点闭合或打开时将产生抖动，使得开关量在动作瞬间的状态不稳，若是工作在计数方式或作为中断输入，将导致系统工作不正常，因此采用触点消抖是必要的。

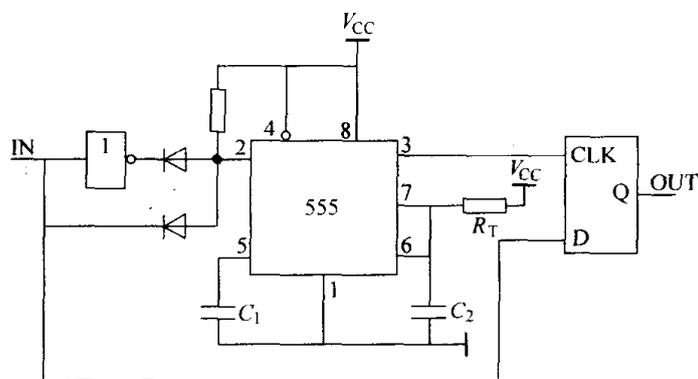


图 4-46 触点消抖电路

实现触点消抖的方法很多，图 4-46 为采用定时器 555 的一种消抖电路。定时器 555 输出作为 D 触发器的时钟，而 D 触发器的 D 端则与输入相连，555 工作在单稳态触发方式，当触点闭合或打开时将触发 555，产生一个脉冲宽度 $f = 0.632RC$ 的脉冲，只要该脉宽大于触点抖动时间，在 D 触发器输出端便能获得与输入状态相同、没有抖动的信号。

2) 脉冲定宽。在许多控制系统中，有时要求在开关量变化时提供一个脉冲宽度稳定的脉冲，如上跳时产生脉冲、下跳时产生脉冲、上/下跳变时都产生脉冲。采用单稳触发器很容易实现这个功能。图 4-47 为上、下跳变都产生脉冲的一种电路，脉冲定宽应注意的问题是：脉冲宽度应明显小于引起脉冲产生的开关量变化的周期。其中 555 定时器工作在单稳态触发器

状态。

3) 消除毛刺。由于受环境干扰的影响, 传输的开关量信号将产生毛刺。带有毛刺的开关量信号会对计算机控制系统的工作可靠性产生一些影响。消除毛刺通常采用史密特触发器(如 74LS14 等)或集成比较器。图 4-48 为采用比较器的整形电路及其电路特性。该电路具有与史密特触发器相似的特性, 门限电平由 R_{P1} 调节, 回环宽度由 R_F 调节。

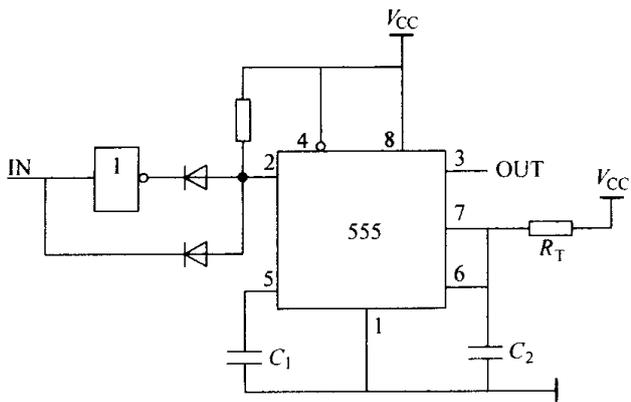


图 4-47 开、关状态产生定宽脉冲电路

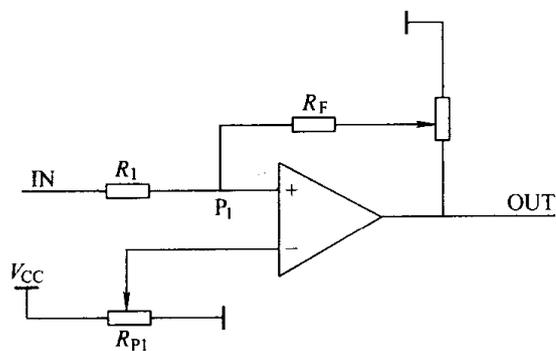


图 4-48 回环比较器

(2) 电平变换

在计算机控制系统中, CPU 一般只接受 TTL 电平信号, 当开关量变换后的信号为非 TTL 电平时, 则需要进行电平变换。电平变换可采用光电隔离、晶体管或 CMOS-TTL 电子变换芯片, 电路如图 4-49 所示, 其中光电隔离抗干扰性能好, 但反应速度较慢, 采用晶体管或 CMOS-TTL 电子变换芯片则速度较快。

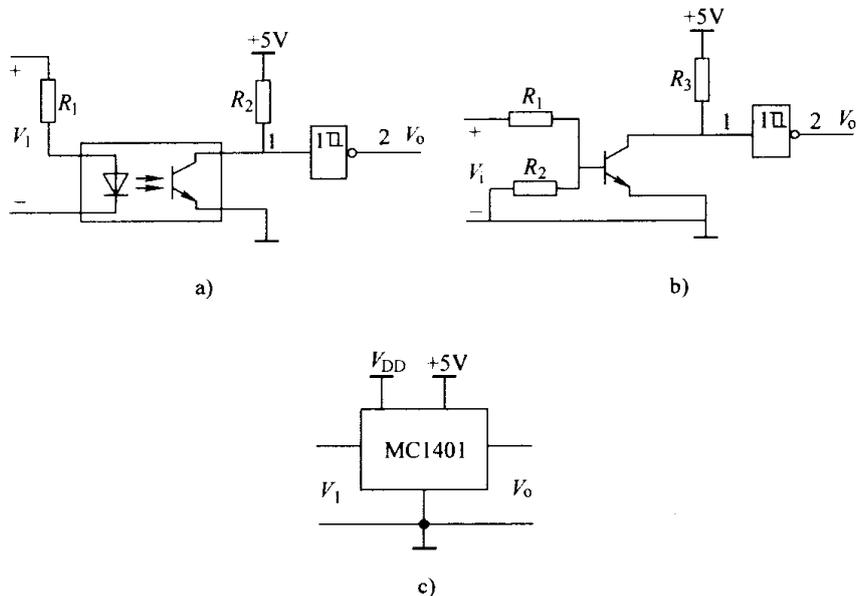


图 4-49 电平变换电路

3. 脉冲计数电路

根据计算机控制系统的功能要求, CPU 对开关量输入信号的处理形式主要有三种: 开关状态检测、脉宽测量和脉冲计数。下面仅介绍其中的脉冲计数电路接口。

有些用于检测流量、转速的传感器发出的是脉冲频率信号，对于大量程可以设计一种定时计数输入接口电路，即在一定的采样时间内统计输入的脉冲个数，然后根据传感器的比例系数可换算出所检测的物理量。

图 4-50 为一种定时计数输入接口电路，传感器发出的脉冲频率信号，经过简单的信号调理，引到 8254 芯片的计数通道 1 的 CLK_1 口。8254 是具有 3 个 16bit 计数器通道的可编程计数器/定时器。图 4-50 中，计数通道 0 工作于模式 3， CLK_0 用于接收系统时钟脉冲， OUT_0 输出一个周期为系统时钟脉冲 N 倍（ N 为通道 0 的计数初值）的连续方波脉冲，其高、低电平时段是计数通道 1 的采样时间和采样间隔时间，分别记为 T_s 、 T_w ；计数通道 1 和通道 2 均选为工作模式 2，且 OUT_1 串接到 CLK_2 ，使两者构成一个计数长度为 2^{32} 的脉冲计数器，以对 T_s 内的输入脉冲计数。

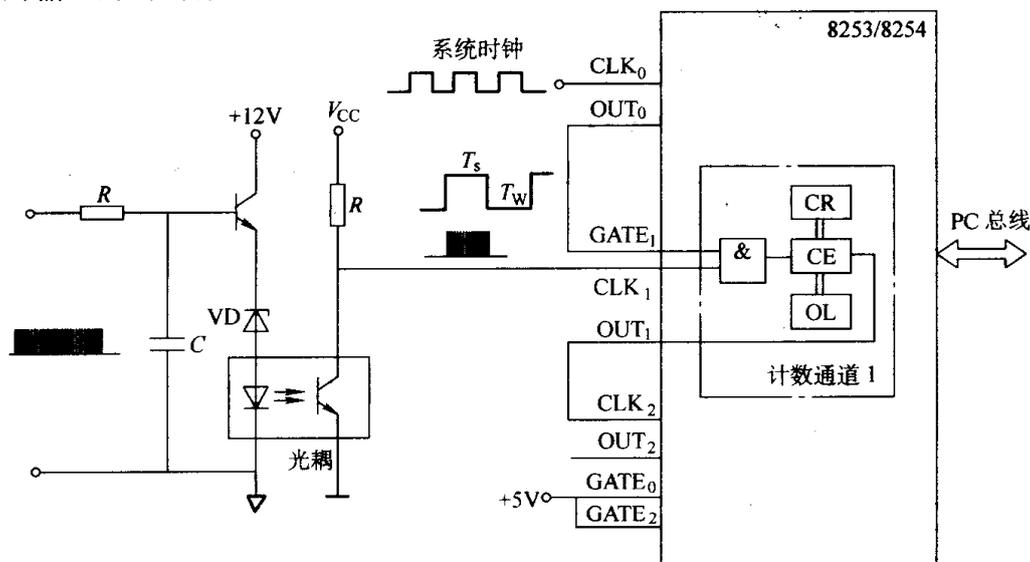


图 4-50 脉冲计数输入接口电路

如果获得 T_s 时间内的输入脉冲个数为 n ，则单位时间内的脉冲个数即脉冲频率为 n/T_s ，从而可换算出介质的流量或电机的转速值。比如，发出脉冲频率信号的是涡街流量计或磁电式速度传感器，它们的脉冲当量（即一个脉冲相当的流量或转数）为 K ，则介质的流量或电机的转数就为 $n/T_s \cdot K$ 。

4.4.2 数字量输出通道

数字量输出通道简称 DO 通道，它的任务是把计算机输出的微弱数字信号转换成能对生产过程进行控制的数字驱动信号。根据现场负荷功率的大小不同，可以选用不同的功率放大器件构成不同的开关量驱动输出通道。常用的有晶体管输出驱动电路、继电器输出驱动电路、晶闸管输出驱动电路及固态继电器输出驱动电路等。

1. 晶体管输出驱动电路

对于低压情况下的小电流开关量，用功率晶体管就可作开关驱动组件，其输出电流就是输入电流与晶体管增益的乘积。

(1) 普通晶体管驱动电路

当驱动电流只有几十毫安时，如驱动发光二极管、小功率继电器等器件，只要采用一个

普通的功率晶体管就能构成驱动电路。图 4-51 所示为驱动 LED 数码管的小功率晶体管输出电路，当 CPU 数据线 D_i 输出数字“0”即低电平时，经 7406 反相锁存器变为高电平，使 NPN 型晶体管 V 导通，集电极电流驱动 LED 数码管发光。

(2) 达林顿驱动电路

当驱动电流需要达到几百毫安时，如驱动中功率继电器、电磁开关等装置，输出电路必须采取多级放大或提高晶体管增益的办法。达林顿阵列驱动器是由多对两个晶体管组成的达林顿复合管构成，它具有高输入阻抗、高增益、输出功率大及保护措施完善等特点，同时多对复合管也非常适用于计算机控制系统中的多路负荷。

图 4-52 给出达林顿阵列驱动器 MC1416 的结构与每对复合管的内部结构，MC1416 内含 7 对达林顿复合管，每个复合管的集电极电流可达 500mA，截止时能承受 100V 电压，其输入/输出端均有钳位二极管，输出钳位二极管 VD_2 抑制高电位上发生的正向过冲， VD_1 、 VD_3 可抑制低电平上的负向过冲。

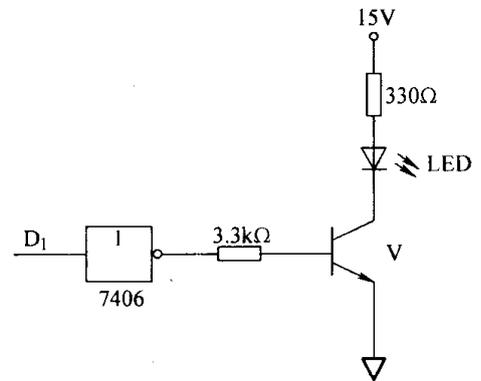
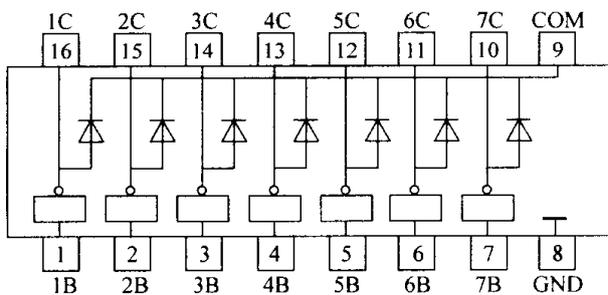
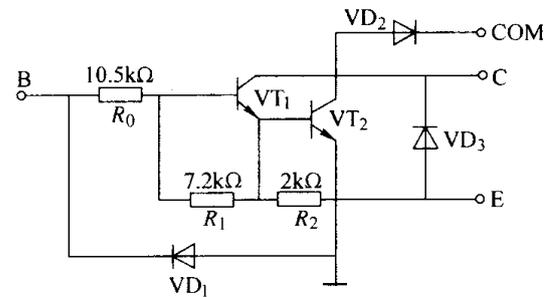


图 4-51 小功率晶体管输出电路



a) MC1416 结构



b) 复合管内部结构

图 4-52 MC1416 达林顿阵列驱动器

图 4-53 所示为达林顿阵列驱动中的一路驱动电路，当 CPU 数据线 D_i 输出数字“0”即低电平时，经 7406 反相锁存器变为高电平，使达林顿复合管导通，产生的几百毫安集电极电流足以驱动负载线圈，而且利用复合管内的保护二极管构成了负载线圈断电时产生的反向电动势的泄流回路。

2. 继电器输出驱动电路

电磁继电器主要由线圈、铁心、衔铁和触点等部件组成，简称为继电器，它分为电压继电器、电流继电器、中间继电器等几种类型。继电器方式的开关量输出是一种最常用的输出方式，通过弱电控制外界交流或直流的高电压、大电流设备。

继电器驱动电路的设计要根据所用继电器线圈的吸合电压和电流而定，控制电流一定要大于

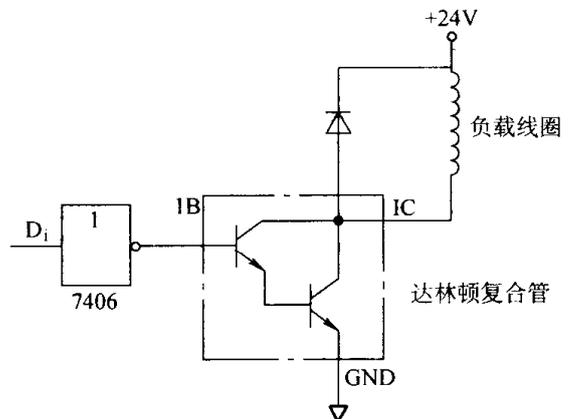


图 4-53 达林顿阵列驱动电路

继电器的吸合电流才能使继电器可靠地工作。图 4-54 所示为经光耦隔离器的继电器输出驱动电路，当 CPU 数据线 D_i 输出数字“1”即高电平时，经 7406 反相驱动器变为低电平，光耦隔离器的发光二极管导通发光，使光敏晶体管导通，继电器线圈 KA 得电，动合触点闭合，从而驱动大型负载设备。

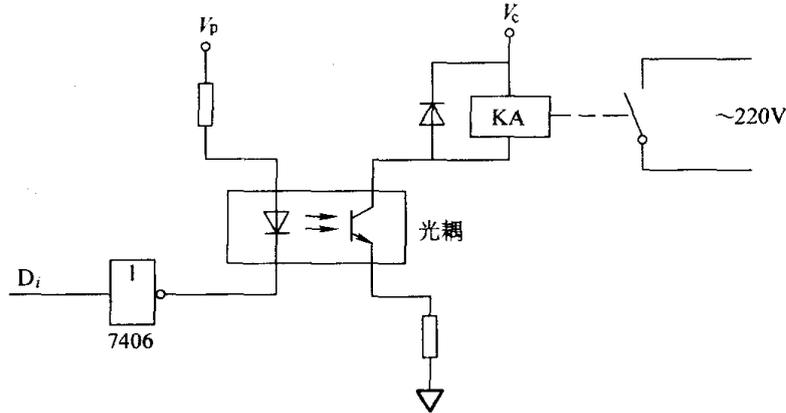


图 4-54 继电器输出驱动电路

由于继电器线圈是电感性负载，当电路突然关断时，会出现较高的电感性浪涌电压，为了保护驱动器件，应在继电器线圈两端并联一个阻尼二极管，为电感线圈提供一个电流泄放回路。

3. 晶闸管输出驱动电路

晶闸管又称可控硅 (SCR, Silicon Controlled Rectifier)，是一种大功率的半导体器件，具有用小功率控制大功率、开关无触点等特点，在交、直流电机调速系统、调功系统、随动系统中应用广泛。

晶闸管是一个三端器件，其符号表示如图 4-55 所示。图 4-55a 为单向晶闸管，有阳极 A、阴极 K、控制极 (门极) G 三个极。当阳、阴极之间加正压时，控制极与阴极两端也施加正压，使控制极电流增大到触发电流值时，晶闸管由截止转为导通；只有在阳、阴极间施加反向电压或阳极电流减小到维持电流以下，晶闸管才由导通变为截止。单向晶闸管具有单向导电功能，在控制系统中多用于直流大电流场合，也可在交流系统中用于大功率整流回路。

双向晶闸管也叫三端双向可控硅，在结构上相当于两个单向晶闸管的反向并联，但共享一个控制极，结构如图 4-55b 所示。当两个电极 T_1 、 T_2 之间的电压大于 1.5V 时，不论极性如何，都可利用控制极 G 触发电流控制其导通。双向晶闸管具有双向导通功能，因此特别适用于交流大电流场合。

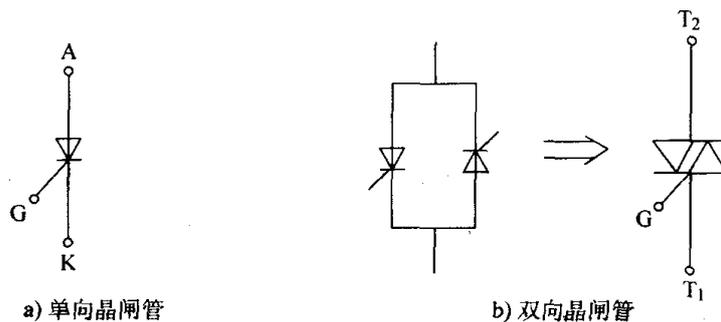


图 4-55 晶闸管结构符号

晶闸管常用于高电压、大电流的负载，不适于与 CPU 直接相连，在实际使用时要采用隔离措施，图 4-56 所示为经光耦隔离的双向晶闸管输出驱动电路，当 CPU 数据线 D_i 输出数字“1”时，经 7406 反相变为低电平，光耦二极管导通，使光敏晶闸管导通，导通电流再触发双向晶闸管导通，从而驱动大型交流负载设备 R_L 。

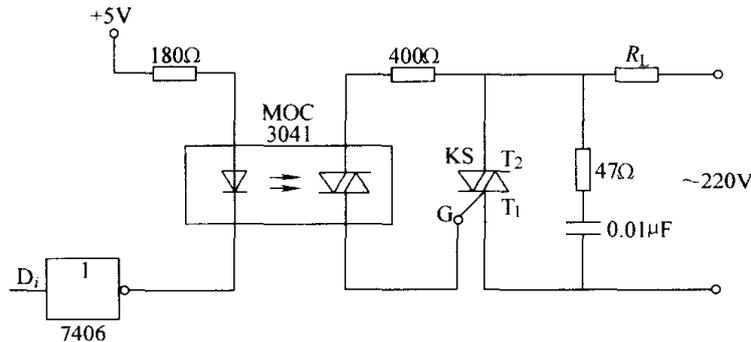


图 4-56 双向晶闸管输出驱动电路

4. 固态继电器驱动电路

固态继电器 (SSR, Solid State Relay) 是一种新型的无触点开关的电子继电器，它利用电子技术实现了控制回路与负载回路之间的电隔离和信号耦合，而且没有任何可动部件或触点，却能实现电磁继电器的功能，故称为固态继电器。它具有体积小、开关速度快、无机械噪声、无抖动和回跳、寿命长等传统继电器无法比拟的优点，在计算机控制系统中得到广泛的应用，大有取代电磁继电器之势。

SSR 是一个四端组件，有两个输入端、两个输出端，其内部结构类似于图 4-56 所示的晶闸管输出驱动电路。图 4-57 所示为其结构原理，它由五部分组成。光耦隔离电路的作用是在输入与输出之间起信号传递作用，同时使两端在电气上完全隔离；控制触发电路是为后级提供一个触发信号，使电子开关（晶体管或晶闸管）能可靠地导通；电子开关电路用来接通或关断直流或交流负载电源；吸收保护电路的功能是为了防止电源的尖峰和浪涌对开关电路产生干扰造成开关的误动作或损害，一般由 RC 串联网络和压敏电阻组成；零压检测电路是为交流型 SSR 过零触发而设置的。

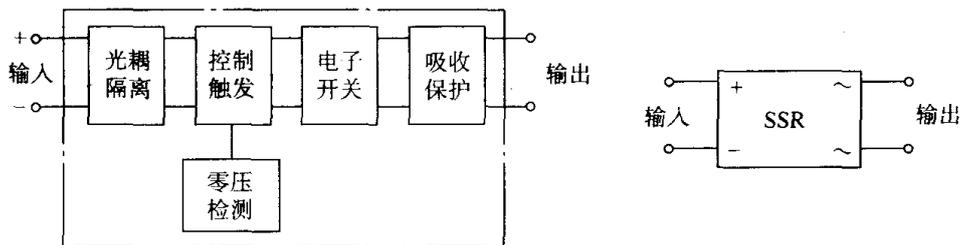


图 4-57 固态继电器结构原理及符号

SSR 的输入端与晶体管、TTL、CMOS 电路兼容，输出端利用器件内的电子开关来接通和断开负载。工作时只要在输入端施加一定的弱电信号，就可以控制输出端大电流负载的通断。

SSR 的输出端可以是直流也可以是交流，分别称为直流型 SSR 和交流型 SSR。直流型 SSR 内部的开关组件为功率晶体管，交流型 SSR 内部的开关组件为双向晶闸管。交流型 SSR

按控制触发方式不同又可分为过零型和移相型两种，其中应用最广泛的是过零型。

过零型交流 SSR 是指当输入端加入控制信号后，需等待负载电源电压过零时，SSR 才为导通状态；而断开控制信号后，也要等待交流电压过零时，SSR 才为断开状态。移相型交流 SSR 的断开条件同过零型交流 SSR 一样，但其导通条件简单，只要加入控制信号，不管负载电流相位如何，均可立即导通。

直流型 SSR 的输入控制信号与输出完全同步。直流型 SSR 主要用于直流大功率控制。一般取输入电压为 4~32V，输入电流为 5~10mA，它的输出端为晶体管输出，输出工作电压为 30~180V。

交流型 SSR 主要用于交流大功率控制。一般取输入电压为 4~32V，输入电流小于 500mA。它的输出端为双向晶闸管，一般额定电流在 1~500A 范围内，电压多为 380V 或 220V。图 4-58 所示为一种常用的固态继电器驱动电路，当数据线 D_i 输出数字“0”时，经 7406 反相变为高电平，使 NPN 型晶体管导通，SSR 输入端得电则输出端接通大型交流负载设备 R_L 。

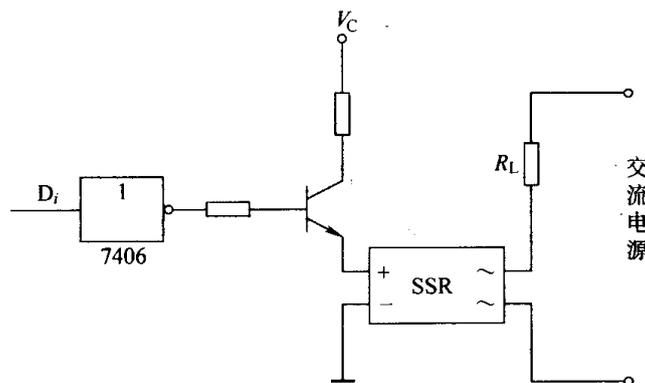


图 4-58 固态继电器输出驱动电路

当然，在实际使用中，要特别注意固态继电器的过电流与过电压保护以及浪涌电流的承受等工程问题，在选用固态继电器的额定工作电流与额定工作电压时，一般要远大于实际负载的电流与电压，而且输出驱动电路中仍要考虑增加阻容吸收组件。具体电路与参数请参考生产厂家有关手册。

4.5 量化误差与字长选择

在实现一个计算机控制系统时必须回答这样一些问题：转换器应有多高精度度？在计算中要求什么样的精度？计算应当用定点运算还是用浮点运算？为了回答这些问题，必须了解限制条件的影响并估计其对闭环系统造成的后果。这并不是一个无关紧要的问题，因为其答案与反馈、算法和采样速率的综合作用有关。当然这只需作粗略的估计即可。例如，分辨率应当是 10 位或 12 位吗？而字长应当为 24 位或 32 位吗？这样的问题也许可用简化的分析来回答。

4.5.1 量化误差来源

到目前为止，所讨论的计算机控制系统均只考虑了信号在时间上的离散化问题，而并未

考虑幅值上的量化效应, 图 4-59 画出了计算机控制系统的典型结构。为便于分析, 图中将采样过程单独画出, 而实际的 A/D 转换器中包含了采样过程。

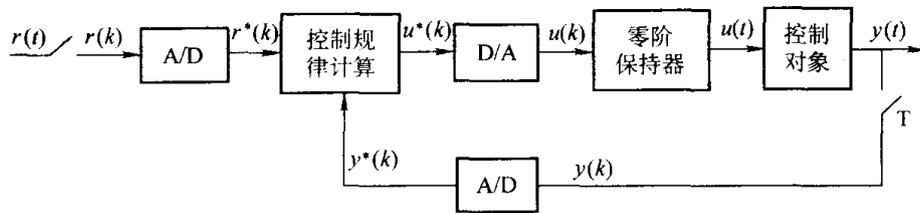


图 4-59 计算机控制系统的典型结构

1. 量化误差

设计算机字长为 n , 采用定点无符号整数, 则机内数的最小单位 q 称为量化单位, 即

$$q = \frac{1}{2^n - 1} \approx 2^{-n} \quad (4-18)$$

通过 A/D 转换可计算出模拟电压 x 相当于多少个整量化单位, 即

$$x = Lq + \varepsilon \quad (4-19)$$

式中, L 为整数, 对于余数 ε ($\varepsilon < q$) 可以用截尾或舍入来处理。

所谓截尾就是舍掉数值中小于 q 的余数 ε , 其截尾误差 ε_t 为

$$\varepsilon_t = x_t - x \quad (4-20)$$

式中, x 为实际数值; x_t 为截尾后的数值。显然 $-q < \varepsilon_t \leq 0$ 。

所谓舍入是指当被舍掉的余数 ε 不小于量化单位的一半时, 则最小有效位加 1; 而当余数 ε 小于量化单位的一半时, 则舍掉 ε 。这时舍入误差为

$$\varepsilon_r = x_r - x \quad (4-21)$$

式中, x 为实际数值; x_r 为舍入后的数值。显然 $-q/2 \leq \varepsilon_r \leq q/2$ 。

在计算机控制系统中数值误差源有三个: 首先被测参数(模拟量)经 A/D 转换器变成数字量时产生了第一次量化误差。在运算之前, 运算式的参数(如 PID 算式中的 K_p 、 T_i 、 T_d 等)必须预先置入指定的内存单元。由于字长有限, 对参数可采用截尾或舍入来处理。另外在运算过程中, 也会产生误差。这些是在 CPU 内产生的第二次量化误差。计算机输出的数字控制量经 D/A 转换器变成模拟量, 在模拟量输出装置内产生了第三次量化误差。

2. 量化误差来源

由图 4-59 可以看出, 产生量化误差的原因主要有以下几个方面。

(1) A/D 转换的量化效应

经过 A/D 转换, 它将模拟信号转换为时间上离散、幅值上量化的数字信号, 根据 A/D 转换装置不同的实现原理, 它将实现如图 4-60a 或 b 所示的两种输入、输出关系(图中以 $y(k)$ 为例), 其中图 4-60a 表示舍入的情况, 图 4-60b 表示截尾的情况。

图中量化单位 q 的大小取决于 A/D 转换信号的最大幅度及转换的字长, 设 $y(k)$ 的最大信号为 $\pm y_m$, 转换字长为 n , 若转换后的二进制数用原码表示, 则总共可表示 $2^n - 1$ 个数, 若用补码表示, 则总共可产生 2^n 个数。由于计算机中的数一般采用补码表示, 因此可以算得量化

单位 q 的大小为

$$q = \frac{2y_m}{2^n} = \frac{y_m}{2^{n-1}} \quad (4-22)$$

q 的大小反映了 A/D 转换装置的分辨能力, 通常 $q/2y_m = 2^{-n}$ 称为 A/D 转换的分辨率。典型的 A/D 转换的位数为 8bit、10bit、12bit 或 14bit, 其分辨率分别为 0.4%、0.1%、0.025% 或 0.006%。

图 4-60 所示的输入、输出关系显示了典型的非线性特性。当 n 比较大, 如 $n \geq 12$, 即 A/D 转换的分辨率较高时, A/D 转换的量化效应对系统性能的影响较小, 一般可以忽略它的影响; 但当 $n \leq 10$ 时, 将对系统性能产生影响。

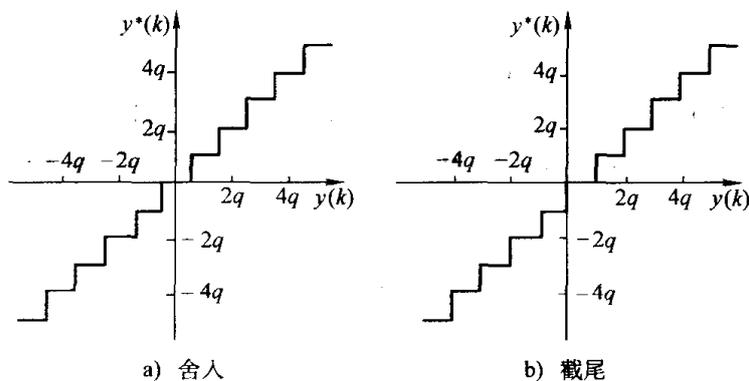


图 4-60 A/D 转换器的输入、输出关系

(2) 控制规律计算中的量化效应

根据图 4-59 中经过量化的数字信号送入计算机的中央处理单元进行控制规律的计算。设计算所用的字长为 n_1 , 一般大于 A/D 转换的字长, 即 $n_1 > n$, 由于计算所用字长是有限的, 因此计算过程中也会产生量化误差。另外, 在计算过程中, 采用定点还是浮点运算也是很关键的问题, 浮点运算一般量化误差很小, 可以忽略不计。但是单片机等进行浮点数运算时速度较慢, 而在计算机实时控制中, 常对计算速度要求很高, 因此, 很多地方仍需要采用定点运算。对于定点数表示, 加或减的运算是准确的, 关键是需选定合适的比例因子, 以避免出现上溢或下溢的问题。对于乘或除的运算, 结果产生双倍的字长, 但是结果数仍只能用单字长来表示, 因而这里也产生量化的问题。通常对于低位数也可采用舍入或截尾两种方法进行处理。

(3) 控制参数的量化效应

在进行控制规律的计算时, 其中的一些参数与要求的参数值也会存在一定的误差, 字长越长, 这种误差便越小。在工程上, 由于控制对象的模型通常是不准确的, 其参数误差有时高达 20%, 故一般说来也不必要求控制器参数非常准确。因此, 控制参数的量化效应通常可以忽略。但有时问题本身对控制器参数很灵敏, 因此参数的量化效应也能对系统性能产生很大的影响。

(4) D/A 转换的量化效应

由于计算所用的字长通常比 D/A 转换的字长要长, 因此, 经过 D/A 转换后, 从 $u^*(k)$ 到 $u(k)$ 之间, 也存在类似如图 4-60 所示的量化效应。

4.5.2 A/D、D/A 字长的选择

为了减少量化误差, 应尽量加大字长, 但这要受技术或硬件成本的制约。因此在误差允

许的条件下, 合理选择 A/D、D/A 转换字长就显得非常重要。

无论是 A/D 转换还 D/A 转换, 总是用一个范围为 $2^n - 1$ 内的整数与转换的模拟量相对应, 则转换的分辨率可表示为

$$D = 1/(2^n - 1) \quad (4-23)$$

若已知一个系统的分辨率或转换精度的要求, 可以用式 (4-24) 求得满足要求的 A/D 转换和 D/A 转换的字长, 即

$$n \geq \log_2(1 + \frac{1}{D}) \quad (4-24)$$

在计算机控制系统中, 常用的 A/D、D/A 转换器字长为 8bit、10bit、12bit、16bit, 按照上述公式估算出的字长取整后再选择一种。特殊被控对象, 可选用更高分辨率的器件。

例如, 某温度控制系统的温度范围为 $0 \sim 600^\circ\text{C}$, 要求分辨率为 0.5°C , 可求出 A/D 转换器字长

$$n \geq \log_2(1 + \frac{1}{D}) = \log_2(1 + \frac{600}{0.5}) \approx 10.23$$

因此, 取 A/D 转换器字长为 12bit。

又如, 某执行机构的最大输入值为 $u_{\max} = 20\text{mA}$, 最小输入值为 $u_{\min} = 4\text{mA}$, 灵敏度为 $\lambda = 0.05\text{mA}$, 参照式 (4-24) 可得 D/A 转换器的字长为

$$n \geq \log_2(1 + \frac{u_{\max} - u_{\min}}{\lambda}) = \log_2(1 + \frac{20 - 4}{0.05}) \approx 8.32$$

因此, 可取 D/A 转换器字长为 10bit 或 12bit。

4.6 数据预处理

数据预处理主要是指计算机控制系统中经 A/D 采样得到的原始数据需要经过一系列的预处理才能进一步进行控制计算、显示及存储等, 主要包括系统误差的自动校准、数字滤波、标度变换、越限报警处理等方面的内容, 其中数字滤波已在第 3 章进行了介绍。

4.6.1 系统误差的自动校准

系统误差是指在相同条件下, 经过多次测量, 误差的数值 (包括大小、符号) 保持恒定, 或按某种已知的规律变化的误差。这种误差的特点是, 在一定测量条件下, 其变化规律是可以掌握的, 产生误差的原因一般也是知道的。因此, 原则上讲, 系统误差是可以通过适当的技术途径来确定并加以校正的。在系统的测量输入通道中, 一般均存在零点偏移和漂移, 产生放大电路的增益误差及器件参数的不稳定现象, 它们会影响测量数据的准确性, 这些误差都属于系统误差。有时必须对这些误差进行校准。下面介绍一种实用的自动校正方法。

这种方法的最大特点是由系统自动完成, 不需要人的介入。其电路如图 4-61 所示。该电路的输入部分有一个多路开关。系统在刚通电时或每隔一定时间, 自动进行一次校准。这时, 先把开关接地, 测出此时的输入值 x_0 ; 然后把开关接基准电压 V_R , 测出输入值 x_1 , 设测量信号 x 与 y 呈线性关系, 即 $y = ax + b$, 由此得到两个误差方程:

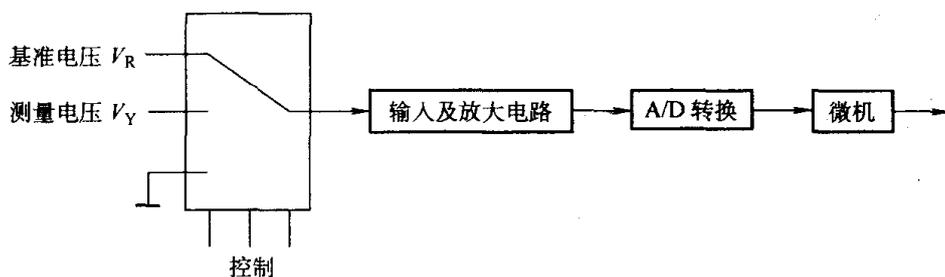


图 4-61 全自动校准电路

$$\begin{cases} V_R = ax_1 + b \\ 0 = ax_0 + b \end{cases} \quad (4-25)$$

解此方程组，得

$$\begin{cases} a = V_R / (x_1 - x_0) \\ b = V_R x_0 / (x_0 - x_1) \end{cases} \quad (4-26)$$

从而可得校正公式

$$y = V_R (x - x_0) / (x_1 - x_0) \quad (4-27)$$

采用这种方法测得的结果与放大器的漂移和增益无关，与 V_R 的精度也无关。这样可大大提高测量精度，降低对电路器件的要求。

4.6.2 标度变换

生产中的各种参数都有着不同的量纲和数值，但在计算机控制系统的采集、A/D 转换过程中已变为无量纲的数据，当系统进行显示、记录、打印和报警等操作时，必须把这些测得的数据还原为相应量纲的物理量，这就需要进行标度变换。

标度变换的任务是把计算机系统检测的对象参数的二进制数值还原变换为原物理量的工程实际值。图 4-62 所示为标度变换原理，这是一个温度测控系统，某种热电偶传感器把现场中的温度值 $0 \sim 1200^\circ\text{C}$ 转变为 $0 \sim 48\text{mV}$ 的信号，经输入通道中的运算放大器放大到 $0 \sim 5\text{V}$ ，再由 8bit A/D 转换器转换成 $00\text{H} \sim \text{FFH}$ 的数字量，这一系列的转换过程是由输入通道的硬件电路完成的。CPU 读入该数字信号后，在送到显示器进行显示以前，必须把这一无量纲的二进制数值再还原变换成原量纲为 $^\circ\text{C}$ 的温度信号。比如，最小值 00H 应变换对应为 0°C ，最大值 FFH 应变换对应为 1200°C 。

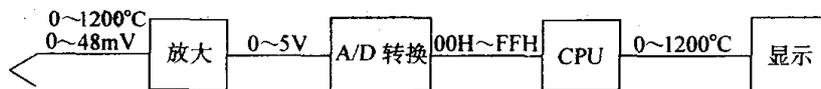


图 4-62 标度变换原理

这个标度变换的过程是由算法软件程序来完成的，标度变换有各种不同的算法，它取决于被测参数的工程量与转换后的无量纲数字量之间的函数关系。一般而言，输入通道中的放大器、A/D 转换器基本上是线性的，因此，传感器的输入、输出特性就大体上决定了这个函数关系的不同表达形式，也就决定了不同的标度变换方法。

标度变换的主要方法有线性式变换、非线性式变换、多项式变换及查表法。

1. 线性式变换

线性标度变换是最常用的标度变换方式,其前提条件是传感器的输出信号与被测参数之间呈线性关系,如图 4-63 所示。

数字量 N_x 对应的工程量 A_x 的线性标度变换公式为

$$A_x = (A_m - A_0) \frac{N_x - N_0}{N_m - N_0} + A_0 \quad (4-28)$$

式中, A_0 为一次测量仪表的下限(测量范围最小值); A_m 为一次测量仪表的上限(测量范围最大值); A_x 为实际测量值(工程量); N_0 为仪表下限所对应的数字量; N_m 为仪表上限所对应的数字量; N_x 为实际测量值所对应的数字量。由于

对一个特定的通道而言,其中 A_0 、 A_m 、 N_0 、 N_m 均为常数,则式(4-28)可进一步简化为

$$A_x = aN_x + b \quad (4-29)$$

$$\text{式中} \quad a = \frac{A_m - A_0}{N_m - N_0}, \quad b = A_0 - \frac{A_m - A_0}{N_m - N_0} N_0$$

例 4-1 某加热炉温度测量仪表的量程为 $200 \sim 800^\circ\text{C}$, 在某一时刻计算机系统采样并经数字滤波后的数字量为 CDH, 求此时的温度值是多少?(设该仪表的量程是线性的)

解: 根据式(4-28), 已知 $A_0 = 200^\circ\text{C}$, $A_m = 800^\circ\text{C}$, $N_x = \text{CDH} = 205$, $N_0 = 0$, $N_m = \text{FFH} = 255$, 所以此时的温度为

$$A_x = (A_m - A_0) \frac{N_x - N_0}{N_m - N_0} + A_0 = (800 - 200) \frac{205}{255} + 200 = 682 \quad (^\circ\text{C})$$

2. 非线性式变换

如果传感器的输出信号与被测参数之间呈非线性关系时,上面的线性变换式均不适用,需要建立新的标度变换公式。由于非线性参数的变化规律各不相同,故应根据不同的情况建立不同的非线性变换式,但前提是它们的函数关系可用解析式来表示。

例如,在差压法测流量中,流量与差压间的关系为

$$Q = K\sqrt{\Delta p}$$

式中, Q 为流体流量; K 为刻度系数(与流体的性质及节流装置的尺寸有关); Δp 为节流装置前、后的差压。

可见,流体的流量与被测流体流过节流装置前、后产生的压力差的平方根成正比,于是得到测量流量时的标度变换公式为

$$Q_x = (Q_m - Q_0) \sqrt{\frac{N_x - N_0}{N_m - N_0}} + Q_0 \quad (4-30)$$

式中, Q_0 为差压流量仪表的下限值; Q_m 为差压流量仪表的上限值; Q_x 为被测液体的流量测量值; N_0 为差压流量仪表下限所对应的数字量; N_m 为差压流量仪表上限所对应的数字量;

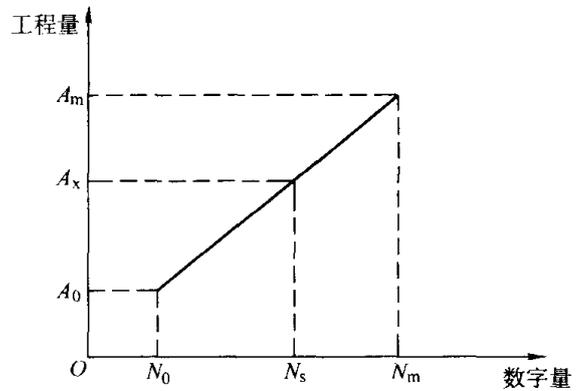


图 4-63 线性关系的标度变换

N_x 为差压流量仪表测得差压值所对应的数字量。

对于流量仪表, 一般下限 Q_0 皆为 0, 同时考虑到 Q_m 、 N_0 、 N_m 都是常数, 故以式 (4-30) 可简化为编程用的标度变换子程序公式, 即

$$Q_x = K_1 \sqrt{N_x - N_0} \quad (4-31)$$

式中, $K_1 = Q_m / \sqrt{N_m - N_0}$

3. 多项式变换

还有些传感器的输出信号与被测参数之间虽为非线性关系, 但它们的函数关系无法用一个解析式来表示, 或者解析式过于复杂而难以直接计算。这时可以采用一种计算既简便又能满足实际工程要求的近似表达式——插值多项式, 来进行标度变换。

插值多项式是用一个 n 次多项式来代替某种非线性函数关系的方法。其插值原理是: 被测参数 y 与传感器的输出值 x 具有的函数关系为 $y = f(x)$, 只知道在 $n+1$ 个相异点处的函数值为: $f(x_0) = y_0$, $f(x_1) = y_1$, \dots , $f(x_n) = y_n$ 。现构造一个 n 次多项式 $P_n(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$ 去逼近函数 $y = f(x)$, 把 $y = f(x)$ 中这 $n+1$ 个相异点处的值作为插值代入 n 次多项式 $P_n(x)$, 便可以获得 $n+1$ 个一次方程组

$$\begin{cases} a_n x_0^n + a_{n-1} x_0^{n-1} + \dots + a_1 x_0 + a_0 = y_0 \\ a_n x_1^n + a_{n-1} x_1^{n-1} + \dots + a_1 x_1 + a_0 = y_1 \\ a_n x_2^n + a_{n-1} x_2^{n-1} + \dots + a_1 x_2 + a_0 = y_2 \\ \vdots \\ a_n x_n^n + a_{n-1} x_n^{n-1} + \dots + a_1 x_n + a_0 = y_n \end{cases} \quad (4-32)$$

式中, x_0, x_1, \dots, x_n 是已知的传感器的输出值; y_0, y_1, \dots, y_n 是被测参数。可以求出 $n+1$ 个待定系数 a_0, a_1, \dots, a_n , 从而构造一个可代替这种函数关系的可插值多项式 $P_n(x)$ 。

下面利用热敏电阻测量温度的例子来说明这一过程。热敏电阻具有灵敏度高、价格低廉等特点, 但是热敏电阻的阻值与温度之间的关系是非线性的, 而且只能以表 4-4 所示的方式来表示。现构造一个三阶多项式 $P_3(R)$ 来逼近这种函数关系。

表 4-4 热敏电阻的温度-电阻特性

温度 $t/^\circ\text{C}$	电阻 $R/\text{k}\Omega$	温度 $t/^\circ\text{C}$	电阻 $R/\text{k}\Omega$	温度 $t/^\circ\text{C}$	电阻 $R/\text{k}\Omega$
10	8.0000	21	6.5574	32	5.5554
11	7.8431	22	6.4516	33	5.4793
12	7.6923	23	6.3491	34	5.4053
13	7.5471	24	6.2500	35	5.3332
14	7.4074	25	6.1538	36	5.2630
15	7.2727	26	6.0606	37	5.1946
16	7.1428	27	5.9701	38	5.1281
17	7.0174	28	5.8823	39	5.0631
18	6.8965	29	5.7970	40	5.0000
19	6.7796	30	5.7142		
20	6.6670	31	5.6337		

取三阶多项式为

$$t = P_3(R) = a_3 R^3 + a_2 R^2 + a_1 R + a_0$$

并取 $t=10$ 、 17 、 27 、 39 这 4 点为插值点，便可以得到

$$\begin{cases} 8.0000^3 a_3 + 8.0000^2 a_2 + 8.0000 a_1 + a_0 = 10 \\ 7.0174^3 a_3 + 7.0174^2 a_2 + 7.0174 a_1 + a_0 = 17 \\ 5.9701^3 a_3 + 5.9701^2 a_2 + 5.9701 a_1 + a_0 = 27 \\ 5.0631^3 a_3 + 5.0631^2 a_2 + 5.0631 a_1 + a_0 = 39 \end{cases}$$

解上述方程组，得

$$a_3 = -0.2346989, \quad a_2 = 6.120273, \quad a_1 = -59.28043, \quad a_0 = 212.7118$$

因此，所求的逼近多项式为

$$t = -0.2346989R^3 + 6.120273R^2 - 59.28043R + 212.7118$$

这就是用来标度变换的插值多项式，将采样测得的电阻值 R 代入上式，即可获得被测温度 t 。

显然，插值点的选择对于逼近的精度有很大的影响。通常在函数 $y = f(x)$ 的曲线上曲率大的地方应适当加密插值点。

一般来说，增加插值点和多项式的次数能提高逼近精度。但同时会增加计算时间，而且在某些情况下反而会造成误差的摆动；另一方面，对于那些带拐点的函数，如果用一个多项式去逼近，将会产生较大的误差。

为了提高逼近精度，且不占用过多的机时，较好的方法是采用分段插值法。分段插值法是将被逼近的函数根据其变化情况分成几段，然后将每一段区间分别用直线或抛物线去逼近。分段插值的分段点的选取可按实际曲线的情况灵活决定，既可以采用等距分段法，又可采用非等距分段法。

如上例热敏电阻温度 t 与电阻值 R 的插值多项式，其计算量较大，程序也较复杂。为使计算简单，提高实时性，可采用分段线性插值公式或称分段线性化的方法，即用多段折线代替曲线进行计算。

根据表 4-3 中的数据制成如图 4-64 所示的热敏电阻特性及分段线性化，图 4-64 中曲线为热敏电阻的负温度-电阻特性，折线 L_0 、 L_1 、 L_2 代替或逼近曲线。当获取某个采样值 R 后，

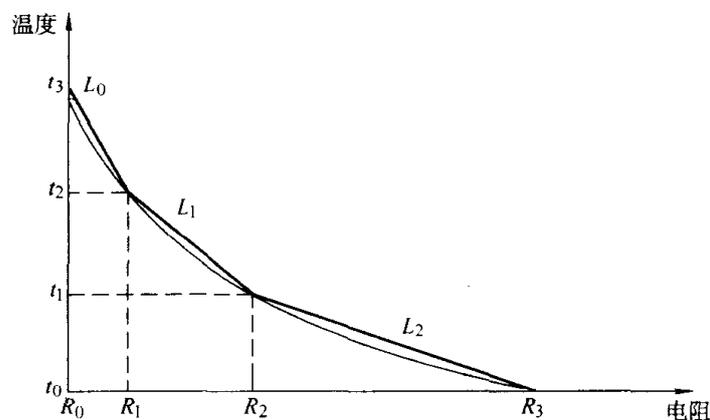


图 4-64 热敏电阻特性及分段线性化

先判断 R 的大小处于哪一折线段内, 然后就可按相应段的线性化公式计算出标度变换值。其计算公式为

$$t = \begin{cases} -k_0(R - R_0) + t_3 & R_0 \leq R \leq R_1 \\ -k_1(R - R_1) + t_2 & R_1 \leq R \leq R_2 \\ -k_2(R - R_2) + t_1 & R_2 \leq R \leq R_3 \end{cases}$$

式中, k_0 、 k_1 、 k_2 分别为线段 L_0 、 L_1 、 L_2 的斜率。

同样, 分段数越多, 线性化精度越高, 软件开销也相应增加。分段数应视具体情况和要求而定。当分段数多到使线段缩成一个点时, 实际上就是另一种方法——查表法。

4. 查表法

所谓查表法就是把事先计算或测得的数据按照一定顺序编制成表格, 查表程序的任务就是根据被测参数的值或者中间结果, 查出最终所需要的结果。它是一种非数值计算方法, 利用这种方法可以完成数据的补偿、计算、转换等各种工作。比如, 输入通道中对热电偶特性的处理, 可以用非线性插值法进行标度变换, 也可以采用精度更高、效果更好的查表法进行标度变换, 即利用热电偶的 $\text{mV} \cdot ^\circ\text{C}$ 分度表, 通过计算机的查表指令就能迅速、便捷地由电动势 (mV) 值查到相应的温度 ($^\circ\text{C}$) 值; 当然控制系统中还会有一些其他参数或表格也是如此, 如对数表、三角函数表、模糊控制表等。

查表程序的繁简程度及查询时间的长短, 除与表格的长短有关外, 很重要的因素在于表格的排列方法。一般来讲, 表格有两种排列方法: 无序表格, 即表格中的数据是任意排列的; 有序表格, 即表格中的数据按一定的顺序排列。表格的排列不同, 查表的方法也不尽相同。

常用的查表方法有顺序查表法、计算查表法及对分查表法等。

(1) 顺序查表法

顺序查表法是针对无序排列表格的一种方法。其查表方法类似人工查表。因为无序表格中所有各项的排列均无一定的规律, 所以只能按照顺序从第一项开始逐项寻找, 直至找到所要查找的关键字为止。

顺序查表法虽然比较“笨”, 但对于无序表格或较短表格而言, 仍是一种比较常用的方法。

(2) 计算查表法

在计算机数据处理中, 一般使用的表格都是线性表, 它是若干个数据元素 X_1, X_2, \dots, X_n 的集合, 各数据元素在表中的排列方法及所占的存储器单元个数都是一样的。因此, 要搜索的内容与表格的排列有一定的关系。只要根据所给的数据元素 X_i , 再通过一定的计算, 求出元素 X_i 所对应数值的地址, 然后将该地址单元的内容取出即可。

这种有序表格要求各元素在表中的排列格式及所占用的空间必须一致, 而且各元素是严格按顺序排列的。其关键在于找出一个计算表地址的公式, 只要公式存在, 查表的时间与表格的长度无关。正因为它对表格的要求比较严格, 并非任何表格均可采用。通常它适用于某些数值计算程序、功能键地址转移程序以及数码转换程序等。

(3) 对分查表法

在前面介绍的两种查表方法中, 顺序查表法速度较慢, 计算查表法虽然速度很快, 但对表格的要求比较挑剔, 因而具有一定的局限性。在实际应用中, 很多表格都比较长, 且难以用计算查表法进行查找, 但它们一般都满足从大到小或从小到大的排列顺序, 如热电偶 mV -

℃分度表、流量测量中差压与流量对照表等。对于这样的表格，通常采用快速而有效的对分查表法。

对分查表法的具体做法是：先取数组的中间值 $D=n/2$ 进行查找，与要搜索的 X 进行比较，若相等，则查到。对于从小到大的顺序来说，如果 $X>n/2$ 项，则下一次取 $n/2\sim n$ 间的中值，即 $3n/4$ 与 X 进行比较；若 $X<n/2$ 项，则取 $0\sim n/2$ 的中值，取 $n/4$ 与 X 进行比较。如此比较下去，则可逐次逼近要搜索的关键字，直至找到为止。

4.6.3 越限报警处理

为了实现安全生产，在计算机测控系统中，对于重要的参数和部位，都设置紧急状态报警系统，以便及时提醒操作人员注意或采取应急措施，使生产继续进行或在确保人身设备安全的前提下终止生产。其方法就是把计算机的采集数据在进行预处理、数字滤波、标度变换之后，与该参数的设定上限、下限值进行比较，如果高于上限值或低于下限值则进行报警，否则就作为采样的正常值，进行显示和控制。

在控制系统中，报警参数可以是被控参数、被测参数、输入偏差或控制量等，设需要判断的报警参数为 X ，该参数的上限、下限约束值分别为 X_H 和 X_L ，则越限报警有如下几种形式。

(1) 上限报警：若 $X_K>X_H$ ，则发出上限报警，否则继续执行原定操作。

(2) 下限报警：若 $X_K<X_L$ ，则发出下限报警，否则继续执行原定操作。

(3) 上、下限报警：若 $X_K>X_H$ ，则发出上限报警，否则判断 $X_K<X_L$ 否？若是则发出下限报警，否则继续执行原定操作。

具体设计报警程序时，为了避免测量值在极限值一点处来回摆动造成频繁报警，一般应在极限值附近设置一个回差带，如图 4-65 所示。

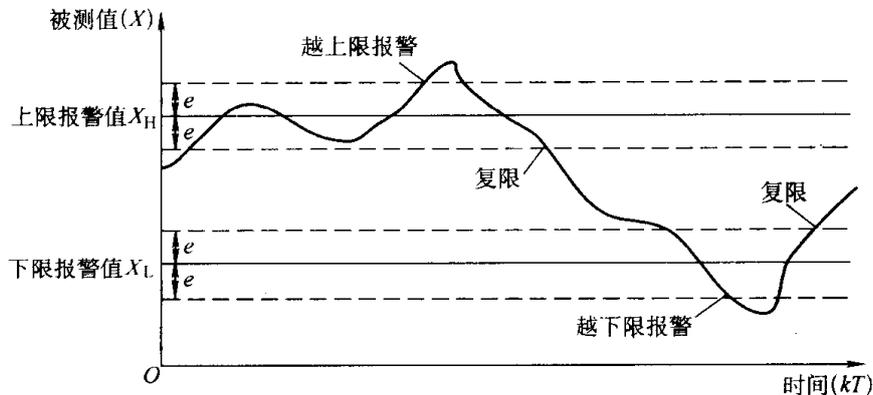


图 4-65 越限报警范围

图 4-65 中， X_H 、 X_L 是上限、下限约束值， $2e$ 为回差带宽。当被测值超越 X_H+e 时，才算越过上限报警值并设置相应的越上限标志（上限标志位置 1），同时输出越上限的声、光报警；当被测值下降到 X_H-e 以下时，则复位上限，这时应撤消越上限标志（上限标志位清 0）及相应的声光报警。同理，当被测值低于 X_L-e 点时，才算越过下限并设置相应的越下限标志（下限标志位置 1），同时输出越下限的声、光报警；当被测值上升到 X_L+e 以上时，则复位下限，这时应撤消越下限标志（下限标志位清 0）及相应的声光报警。如此，回差值 e 避免了测量值在极限值 X_H 或 X_L 一点处来回摆动造成频繁报警， e 值的大小可根据现场具体的被测参数设

定。越限报警程序流程如图 4-66 所示。

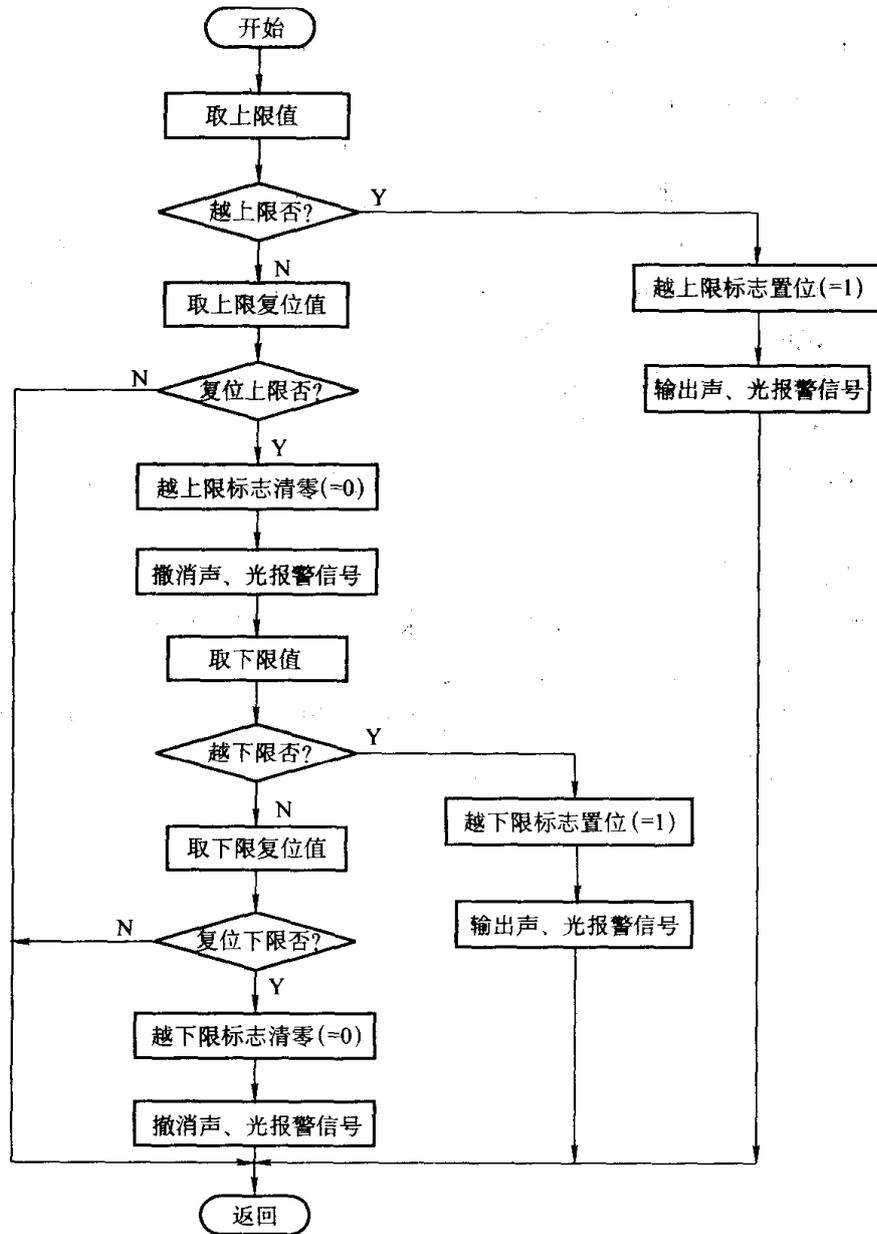


图 4-66 越限报警程序流程

思考题

1. 画图说明模拟量输出通道的功能、各组成部分及其作用。
2. D/A 转换器的性能指标有哪些?
3. 结合图 4-5 分析说明 DAC0832 的内部结构组成及其作用。
4. 结合图 4-7 分析说明由 DAC0832 组成的单缓冲接口电路的工作过程, 编写完成一次 D/A 转换的接口程序。
5. 结合图 4-8 分析说明由 DAC1210 组成的接口电路的工作过程, 编写完成一次 D/A 转

换的接口程序。

6. 简单说明 D/A 转换输出电路有几种输出方式。
7. 结合图 4-16 分析说明自动/手动双向无扰动切换过程。
8. 画图说明模拟量输入通道的功能、各组成部分及其作用。
9. 分析图 4-23 采样保持器的原理电路及工作过程。
10. 简述逐位逼近式、双积分式、 V/F 式以及 $\Sigma-\Delta$ 式的 A/D 转换原理。
11. 结合图 4-31 与图 4-32, 分析说明 ADC0809 的结构组成及其引脚作用。
12. 试分析图 4-33、图 4-34 中 ADC0809 接口电路的启动、转换、查询或定时读入数据的工作过程。比较说明这两种接口电路在硬件和软件上的异同点。
13. 分析说明图 4-38 的 8 路 12bit A/D 转换模板的工作原理。
14. 如何选择 A/D 或 D/A 转换的位数?
15. 在计算机控制系统中, 一般要对测量数据进行哪些预处理技术?
16. 如何进行系统的自动校准?
17. 结合图 4-62, 分析说明标度变换的概念及其变换原理。
18. 某温度测量系统(假设为线性关系)的测温范围为 $0\sim 150^{\circ}\text{C}$, 经 ADC0809 转换后对应的数字量为 $00\text{H}\sim\text{FFH}$, 试写出它的标度变换公式。
19. 在数据处理中, 何为查表法? 它能完成哪些功能? 一般有哪些查表方法?
20. 在计算机控制系统中, 如何设置越限报警?

第 5 章 总线和总线式工控机

随着计算机设计的日益科学化、合理化、标准化和模块化，计算机总线概念也逐渐形成和完善起来。在采用计算机控制系统的工业生产的过程中，特别是对于具有一定规模的工程项目而言，出于可靠性、实时性好，功能丰富、配套完善、扩充灵活及开发周期短的考虑，人们更愿意选用专用于工业控制的现成的通用控制计算机。总线式工业控制计算机（IPC）不仅在中、小型控制系统中担当主要控制装置，而且还可以作为大型网络控制系统中最基层的一种控制单元。

5.1 总线及其分类

总线是一组信号线的集合，是一种传送规定信息的公共通路，它定义了各引线的信号、电气和机械特性。利用总线可以实现芯片内部、印制电路板各部件之间、机箱内各模板之间、主机与外部设备之间或系统与系统之间的连接与通信。

定义总线标准的目的有两个：一是生产厂家能按照统一的标准设计制造计算机；二是用户可以依照某种标准总线选择不同生产厂家的各种模板或设备，灵活地构成各种用途的计算机系统。总线标准会直接影响整个计算机系统的性能、可靠性、扩展性和升级性。因而，总线是构成计算机应用系统的一种重要技术。

5.1.1 总线的分类

按总线在系统的不同层次位置，总线可分为如下几类。

1. 片内总线

片内总线是在集成电路芯片内部，用来连接各功能单元的信息通路。例如，CPU 芯片中的内部总线，它是算术逻辑部件、寄存器和控制器之间的信息通路。这种总线一般是由芯片生产厂家设计的，计算机系统设计者并不关心。但是随着微电子学的发展，出现了 ASIC 技术，用户也可以按自己的要求借助于 CAD 技术，设计自己所需的芯片。

2. 内部总线

内部总线是指总线式计算机系统内部各个功能模板之间的信息通路，通过机箱内的无源底板插槽，完成系统内各插件模板之间的信息传送，因此称为板级总线，也常称为系统总线。显然，系统总线对计算机设计者和计算机用户都是十分重要的一种设计标准。通常意义下的计算机总线概念即是指系统总线。

采用系统总线标准设计、组装的计算机模板与设备的兼容性很强。因为接插件的机械尺寸、各引脚的信号定义、逻辑关系、时序要求和驱动能力等都遵守统一的总线标准。

3. 外部总线

外部总线是指计算机系统与系统之间或计算机系统与外设之间的信息通路，常称为通信总线。这种总线的数据传输方式可以是并行的，也可以是串行的。并行总线的特点是多位数

据同时传输，其优点是传输速度快、接口简单，缺点是电缆线数多、成本较高。串行总线的特点是多位数据一位一位地按顺序传输，其优点是电缆线数少、适合于远距离通信，缺点是传输速度慢、接口复杂。通信总线常是计算机网络设计者所要考虑的总线标准。常用的外部总线有 RS-232C、RS-485、IEEE-488 和并行打印总线等。

4. 现场总线

现场总线 (Field Bus) 是用于过程自动化和制造自动化最底层的现场设备或现场仪表互联的通信网络，是现场通信网络与控制系统的集成。现场总线不仅是一种通信技术，也不仅是用数字仪表代替模拟仪表，而是开创了新一代的现场总线控制系统，以代替传统的分散控制系统。

5.1.2 总线的模板化结构

系统总线的概念是与计算机的模板化结构分不开的。为了提高计算机系统的通用性、灵活性和可扩展性，计算机的各部件采用模板化结构，再通过总线把各模板连接起来，称之为总线的模板化结构。它的核心是设计若干块通用功能模板。

控制计算机面向工业生产过程，不同行业的生产过程使用不同的原料，生产不同的产品，即使生产同一产品的生产过程，也有设备和工艺等区别。因此，不可能设计出多种固定配置的计算机来适用于各种不同的生产过程。为了解决这一难题，就需要对计算机的各种控制对象进行分析与综合，针对其共性，设计若干通用功能部件，并把这些部件按功能划分为几块，再按总线标准设计成模板。常用的模板有 CPU、RAM/ROM、A/D、D/A、DI、DO、PIO (并行输入、输出) 及 SIO (串行输入、输出) 等。

通过对模板品种和数量的选择与组合，就可以方便地配置成不同生产过程所需要的工业控制计算机。由于采用了模板化结构，方便了用户的选用。如果生产过程要扩大规模，改进工艺，并相应要求改变计算机的配置或增加功能，也由于模板化结构的开放性而容易得到满足。所以模板化设计的总线结构提高了系统的灵活性、通用性和可扩展性。

模板化设计也为系统的维修提供了方便。由于每块模板功能比较单一，一旦出现故障，很容易判断是哪一块模板的问题，在有备用模板的情况下，把坏的模板换下来，系统即可正常工作。由于模板的总线端都加了驱动与隔离，故障不会扩散到系统中的其他模板上。所以模板化设计的总线结构也大大提高了系统的可靠性和可维护性。

5.2 工业控制计算机的系统总线

系统总线是构成完整的计算机系统的内部信息枢纽。由于历史的原因，目前存在有多种总线标准，国际上已正式公布或推荐的总线标准主要有 VME 总线、MULTIBUS 总线、NUBUS 总线、UNIBUS 总线、FASTBUS 总线、FUTUREBUS 总线、STD 总线、PC 总线及 PCI 总线等。这些总线标准都是在一定的历史背景和应用范围内产生的。

尽管各种总线的引线数目不同，但按功能通常分为数据总线 DB、地址总线 AB、控制总线 CB 和电源总线 PB 等四大部分。

早期的工业控制计算机多采用 STD 总线，这是一种专为工业控制设计的一种 8bit 总线，具有小模板、开放式结构的特点，但其 8bit 结构以及总线速度的制约，后虽经改进可扩展到

16/32bit, 但已渐渐跟不上当前电子技术特别是 CPU 技术的发展, 逐步被更通用、软/硬件资源更丰富的 PC 及 PCI 总线所替代。

PC 总线是 IBM PC 总线的简称, PC 总线因 IBM PC 及其兼容机的广泛普及而成为全世界用户承认的一种事实上的标准。PC 总线工业控制计算机也称为工业 PC, 即 IPC。随着 CPU 的更新换代, PC 总线也随之扩充, 诸如支持 80286CPU 的 ISA 总线、支持 80486 和奔腾系列的 VESA 总线、PCI 总线等。目前, 工业控制用的 PC 总线现在已逐渐演变为一种总线系统的总称, 包括 ISA 总线 (PC/AT 总线)、PCI 总线和 PC104 总线等。

描述总线的指标主要有两个: 一个是总线宽度, 指总线一次操作可以传输的数据位数值, 如 IBM PC 总线为 8bit, PCI 总线为 32bit; 另一个是总线频率, 指总线工作时的最高时钟频率, 如 ISA 总线为 8MHz、PCI 总线为 33.3MHz。显然, 时钟频率越高, 单位时间可传送的数据量越大。

工业控制计算机目前常用的总线是 ISA 总线和 PCI 总线。ISA (Industrial Standard Architecture) 总线是 IBM 公司为 1984 年推出的 PC/AT 而建立的系统总线标准, 又称 AT 总线。它同时具有 8bit 与 16bit 扩展槽结构, 在 1993 年后在很多场合被 PCI 总线取代。PCI (外围部件互联, Peripheral Component Interconnect) 总线是随着系统速度不断提高, 针对计算机总线接口相对简单而制定的一种处理器局部总线。它可定义两种数据宽度: 32bit 和 64bit, 总线速度最高可达 66MHz 以上; 理论处理数据能力 32bit 为 264MB/s, 64bit 为 528MB/s。大多数计算机和操作系统都支持 PCI 总线, 非常适合在高速计算机和高速数据通信领域中应用。

下面主要介绍 PC/AT 总线。

IBM PC 总线共有 62 根引脚线, 其 CPU 是 Intel 公司准 16bit 的 8088 或 16bit 的 8086, 表 5-1 列出了 PC 总线的 62 根引脚分配。其中, 电源线和辅助电源线为 11 根, 数据总线为 8 根, 地址总线为 20 根 (寻址能力 1M), 控制总线为 21 根, 另外还有 2 根外态线, 即 I/O $\overline{\text{CHK}}$ (I/O 通道检查)、I/O CH RDY (I/O 通道就绪)。控制总线包括中断请求 ($\overline{\text{IRQ2}} \sim \overline{\text{IRQ7}}$)、DMA 请求 ($\overline{\text{DRQ1}} \sim \overline{\text{DRQ3}}$)、DMA 响应 ($\overline{\text{DACK0}} \sim \overline{\text{DACK3}}$)、存储器读和写 ($\overline{\text{MEMR}}$ 和 $\overline{\text{MEMW}}$)、I/O 端口读和写 ($\overline{\text{IOR}}$ 和 $\overline{\text{IOW}}$) 等。

表 5-1 IBM PC 总线的 62 个引脚定义

B 面引脚	符号	名称	A 面引脚	符号	名称
B1	GND	地	A1	I/O $\overline{\text{CHK}}$	I/O 通道校验
B2	RESET DRV	复位	A2	D7	8 条数据总线, 双向
B3	+5V	电源	A3	D6	
B4	IRQ2	中断请求 2	A4	D5	
B5	-5V	电源	A5	D4	
B6	DRQ2	DMA 请求 2	A6	D3	
B7	-12V	电源	A7	D2	
B8	RESERVED	保留	A8	D1	
B9	+12V	电源	A9	D0	
B10	GND	地	A10	I/O CH RDY	I/O 通道就绪
B11	$\overline{\text{MEMW}}$	存储器写	A11	AEN	DMA 控制

(续)

B 面引脚	符号	名称	A 面引脚	符号	名称
B12	$\overline{\text{MEMR}}$	存储器读	A12	A19	20 条地址线, 输出, 带锁存
B13	$\overline{\text{IOW}}$	IO 端口写	A13	A18	
B14	$\overline{\text{IOR}}$	IO 端口读	A14	A17	
B15	$\overline{\text{DACK3}}$	DMA 响应 3	A15	A16	
B16	DRQ3	DMA 请求 3	A16	A15	
B17	$\overline{\text{DACK1}}$	DMA 响应 1	A17	A14	
B18	DRQ1	DMA 请求 1	A18	A13	
B19	$\overline{\text{DACK0}}$	DMA 响应 0	A19	A12	
B20	CLOCK	时钟	A20	A11	
B21	IRQ7	中断请求 7	A21	A10	
B22	IRQ6	中断请求 6	A22	A9	
B23	IRQ5	中断请求 5	A23	A8	
B24	IRQ4	中断请求 4	A24	A7	
B25	IRQ3	中断请求 3	A25	A6	
B26	$\overline{\text{DACK2}}$	DMA 响应 2	A26	A5	
B27	T/C	终止计数	A27	A4	
B28	ALE	地址锁存	A28	A3	
B29	+5V	电源	A29	A2	
B30	OSC	主振信号	A30	A1	
B31	+GND	地	A31	A0	

注: 字母上面标有横线的表示低电平有效。

IBM PC 总线只有 8bit 数据线, 后来 IBM 公司将原来 8bit 的 IBM PC 总线扩展为 16bit 的 AT 总线, 它保持原来 PC 总线的 62 个引脚信号, 以便原先的 PC/XT 适配器板可以插在 AT 机的插槽上。同时为使数据总线扩展到 16bit, 地址总线扩展到 24bit, 而增加一个延伸的 36 引脚插槽。新增加的 36 个引脚定义列于表 5-2 中, 主要包括高位地址线 (新增 A20~A23 和重复 A17~A19, 不带锁存)、高位数据线 (D8~D15)、中断请求 (IRQ10~IRQ15)、DMA 请求 (DRQ0、DRQ5~DRQ7)、DMA 响应 ($\overline{\text{DACK0}}$ 、 $\overline{\text{DACK5}}$ ~ $\overline{\text{DACK7}}$)、存储器读和写 ($\overline{\text{SMEMR}}$ 和 $\overline{\text{SMEMW}}$, 16MB 范围有效)、16bit 存储器片选 ($\overline{\text{MEM CS16}}$)、16bit I/O 片选 ($\overline{\text{I/O CS16}}$) 等。

表 5-2 AT 总线新增加的 36 个引脚定义

C 面引脚	符号	名称	D 面引脚	符号	名称
C1	$\overline{\text{SBHE}}$	高字节允许	D1	$\overline{\text{MEM CS16}}$	存储器 16 位片选
C2	A23	高位地址, 双向, 不带锁存	D2	$\overline{\text{I/O CS16}}$	I/O 端口 16 位片选
C3	A22		D3	IRQ10	中断请求, 输入
C4	A21		D4	IRQ11	

(续)

C 面引脚	符号	名称	D 面引脚	符号	名称
C5	A20	高位地址, 双向, 不带锁存	D5	IRQ12	中断请求, 输入
C6	A19		D6	IRQ14	
C7	A18		D7	IRQ15	
C8	A17		D8	$\overline{\text{DACK0}}$	DMA 响应 0
C9	$\overline{\text{SMEMR}}$	存储器读, 双向	D9	DRQ0	DMA 请求 0
C10	$\overline{\text{SMEMW}}$	存储器写, 双向	D10	$\overline{\text{DACK5}}$	DMA 响应 5
C11	D8	数据总线高字节, 双向	D11	DRQ5	DMA 请求 5
C12	D9		D12	$\overline{\text{DACK6}}$	DMA 响应 6
C13	D10		D13	DRQ6	DMA 请求 6
C14	D11		D14	$\overline{\text{DACK7}}$	DMA 响应 7
C15	D12		D15	DRQ7	DMA 请求 7
C16	D13		D16	+5V	电源
C17	D14		D17	$\overline{\text{MASTER}}$	主控, 输入
C18	D15		D18	GND	地

注: 字母上面标有横线的表示低电平有效。

5.3 总线式工业控制计算机

工业 PC (IPC) 是以 PC 总线 (ISA、VESA、PCI 总线) 为基础构成的工业计算机。

5.3.1 IPC 的硬件组成

为了提高 IPC 的通用性、灵活性和可扩展性, IPC 的各部件均采用模板化结构, 即在一块无源的并行底板总线上, 插接多个功能模板即组成了一台 IPC 的硬件装置。其硬件组成框图如图 5-1 所示, 除了构成计算机基本系统的主机板、人-机接口、系统支持、磁盘系统、通信接口板外, 还有 AI、AO、DI、DO 等数百种工业 I/O 接口板可供选择。选用的各个模板彼此通过内部总线相连, 而由 CPU 通过总线直接控制数据的传送和处理。

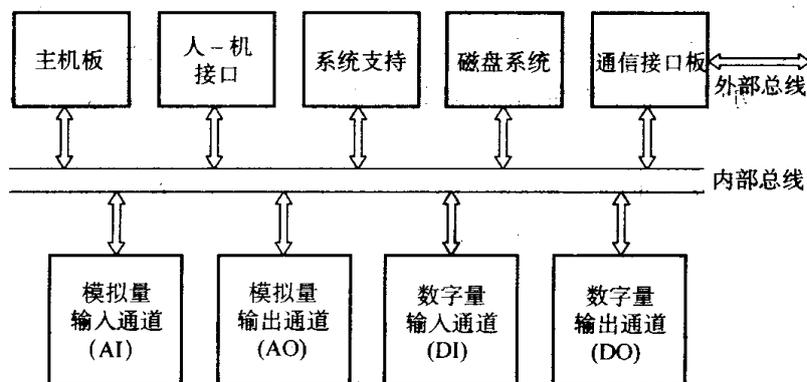


图 5-1 IPC 的硬件组成框图

5.3.2 IPC 的特点

普通的 PC 总线计算机是为数据处理,而不是针对工业控制而设计的,其本身有若干严重不足之处,如既没有总线仲裁能力、不支持多主系统,也不如 MULTI BUS、VME 总线设计得全面和周到;在结构上,其母板水平放置,抗冲击和抗振能力差,易积灰尘。总之,难于抵御工业现场的天敌(如发热、粉尘、振动、辐射、电磁场和腐蚀性气体等)的侵蚀,因而,难以完成 IPC 的基本使命。目前的 IPC 是对普通 PC 总线计算机进行“加固”而成的,除了具备 PC 的功能外,其结构上有如下特点。

1. 加固型工业机箱

由于工控机应用于较恶劣的工业现场环境,因此对机箱应采取一系列加固措施。一般工业用的 IPC 采用全钢标准机架机箱(19in),机箱密封,双冷却风扇正压送风散热(前风扇带空气过滤罩),使用防振可调节夹具固定模板,以达到防振、防冲击、防尘、通风散热性能良好的目的,并具有良好的电磁屏蔽能力。

2. 工业级抗干扰电源

一般 IPC 均采用工业级的开关电源,电源部分具有防浪涌冲击、过电压过电流保护功能、抗干扰能力强、适应输入电压范围广,输出功率一般大于 200W。

3. 无源总线底板

目前的 IPC 一般采用 PC/AT 总线和 PCI 总线兼容的无源母板作为底板,采用多层 PCB(多为 4 层),同时具有 PC/AT 插槽(黑色)和 PCI 插槽(白色),一般的插槽总数都大于 6 个。

4. 一体化主板

主机板是工控机的核心部件,目前有 80486、80586、Pentium1~Pentium 4 等各类主机板,板上所有元器件性能都达到了工业级标准。并且是一体化主板(ALL-IN-ONE)。目前先进的 ALL-IN-ONE 主板集成了几乎所有 PC 功能部件,包括基本 I/O 控制、存储器(如 RAM、ROM、EPROM、E²PROM、FLASH 等)、人-机接口(如键盘、鼠标接口等)、通信接口(RS-232C、打印口、USB 口、以太网口等),甚至包含显示器接口(即集成了显示卡的功能)。除此之外,ALL-IN-ONE 主板一般还包含:程序运行监视系统,即看门狗定时器,当系统出现异常时能使系统自动恢复;电源掉电检测,其目的是为了及时检测到电源掉电后,立即保护当时的重要数据和各寄存器的状态;保护重要数据的后备存储器,采用带有后备电池的 SRAM、NOVRAM 及 E²PROM,能在系统掉电后保证数据不丢失;实时日历时钟,用于定时自动执行某些控制功能和自动记录某个控制是在何时发生的;PC/PC-104 总线接口。

5. 显示设备

IPC 除了采用外接 CRT 或液晶屏外,还常使用显示屏与机箱一体的一体化机箱。同时,有些 IPC 也使用触摸屏,使其具有输入功能。

6. 电子盘

IPC 为了增强存储设备的可靠性,除了采用一般的软盘、硬盘等外存储设备外,还常广泛使用半导体虚拟存储器(如 RAM、ROM、EPROM、E²PROM、FLASH 等),即电子盘。电子盘由于不包含机械转动部件,具有更高的可靠性和使用寿命,更适合在工业现场的恶劣环境下使用。

7. 输入、输出模板

输入、输出模板是 IPC 和生产过程之间信号传递和变换的连接通道，它包括模拟量输入（AI 或 A/D）模板、模拟量输出（AO 或 D/A）模板、数字量输入（DI）模板及数字量输出（DO）模板等种类。由于输入或输出均涉及到生产现场被控参数的种类、个数、精度及干扰等，因而其该类模板是系统中性能差异最大、品种类型最多，也是用户选择最为丰富的一种。

IPC 除了具备上述硬件结构上的特点外，还具有丰富的控制软件。一般地，IPC 配备完整的操作系统、适合生产过程控制的工具软件以及各种控制软件包。工业控制软件正向结构化、组态化方向发展，如组态王 KingView、WinCC、Intouch 及 FIX 等。

IPC 在硬件结构上的特点，再配以相应的工业控制应用软件和原 PC 的巨大的软、硬件资源，使 PC 总线工控机在自动控制领域中得到了迅速普及，其发展速度之快，远超出了其他任何一种总线的工控产品。

5.4 IPC 产品简介

目前国内外生产总线式 IPC 的专业厂家很多，著名品牌比比皆是，如研华、研祥、凌华、中泰、康泰克、康拓、威达、华控及浪潮等。现以深圳研祥工控集团研制的 PC 总线工业计算机产品为例进行说明。

研祥的 PC 总线工业计算机共有 14 种系列：工业计算机机箱，工业一体化工作站，工业平板计算机/显示器，便携式工业计算机，工业级 CPU 卡，工业级底板，工业计算机外设，亚当（ADAM）系列远程数据采集与控制模块，基于 ISA 总线数据采集与控制卡，基于 PCI 总线数据采集与控制卡，端子板和附件，工业通信软件、应用软件等。每一系列又有几种至十几种规格型号。以下以研祥公司的几种产品为例对工业控制计算机产品的特点进行简单介绍。

5.4.1 工业计算机机箱

研祥的机箱按带不带底板和电源，安装形式是桌面、壁挂还是上架以及尺寸长度不同等，又可分为二十几种型号。图 5-2 所示为 IPC-810/811 型号的 14 槽上架型计算机机箱。

其产品特性为：19in 上架型，符合 EIA RS-310C 标准，14 槽 ISA/PCI/PICMG 无源底板，提供 3 个 3.5in 和 2 个 5.25in 磁盘驱动器空间，前面板带电源开关，CPU 复位按钮，键盘锁开关，电源开关 LED 和 HDD LED，防尘带锁门保护控制部件，前、后面板均带键盘接口，防止插卡振动的防振压条，支持 PS/2 规格电源，尺寸为 177mm（高）× 482mm（宽）× 452mm（深），支持 ATX 母板。

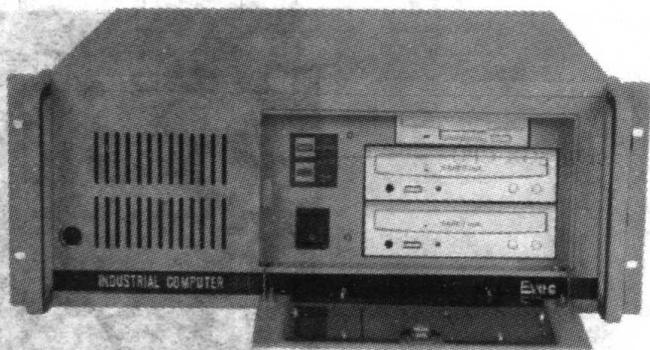


图 5-2 IPC-810/811 机箱

5.4.2 工业级底板

研祥的底板按插槽种类、槽数多少、电源种类以及配置不同等，又可分为三十几种型号。图 5-3 所示为 IPC-6114P4 型号的 14 槽底板。

产品规格为：最大扩展槽 14，其中 ISA 槽 9 个，PCI 槽 4 个，PICMG 槽 1 个，电源±5V、±12V、+3V，尺寸为 323mm×300mm。其中 PICMG 插槽是一种符合 PICMG 标准的控制总线插槽，在同一个 PICMG 插槽内包含了 PCI 和 ISA 总线的插槽或两个 PCI 总线的插槽，提供了 PCI 总线与 ISA 总线间或两个 PCI 总线间相互连接的桥梁，一般用于全长 PCI 总线 CPU 卡的插接。

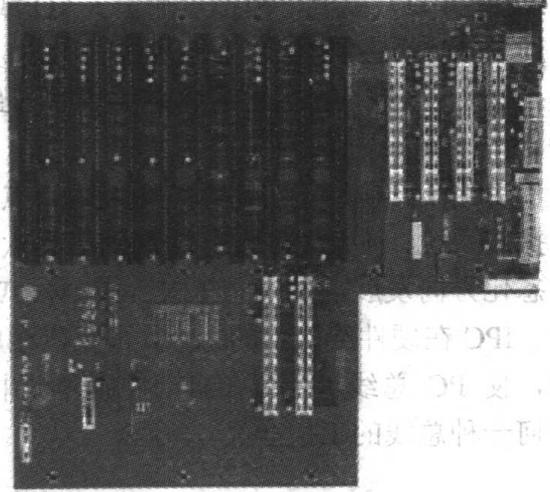


图 5-3 IPC-6114P4 型 14 槽 PCI/ISA 底板

5.4.3 工业级 CPU 卡

1. ISA 总线 CPU 卡

研祥的 CPU 卡按 CPU 种类、存储器容量、显示方式、主板长短以及 I/O 接口不同等，又可分为十几种型号。

图 5-4 所示为 IPC-586VDNH (GX) CPU 卡。IPC-586VDNH (GX) 采用美国国家半导体公司低功耗 586 级处理器 GX1 系列设计，单 5V 供电，板上集成了 VGA/LCD 控制器，支持 18bitTFT 显示；作为 486 及以下主板替代产品，IPC-586VDNH (GX) 支持 168 线内存（代替 30 线、72 线旧产品），网络接口升速至 100Mbit/s，增加了两个广泛使用的 USB 接口，处理器达到 586 多媒体级性能，在 0~60℃工作范围内无需风扇，极大提高了系统的可靠性。IPC-586VDNH (GX) 既可通过工业底板机箱组成工业级应用产品，也可作为嵌入式单板使用。可广泛应用于工业产品、仪器仪表、Internet 设备及各种智能仪器仪表。

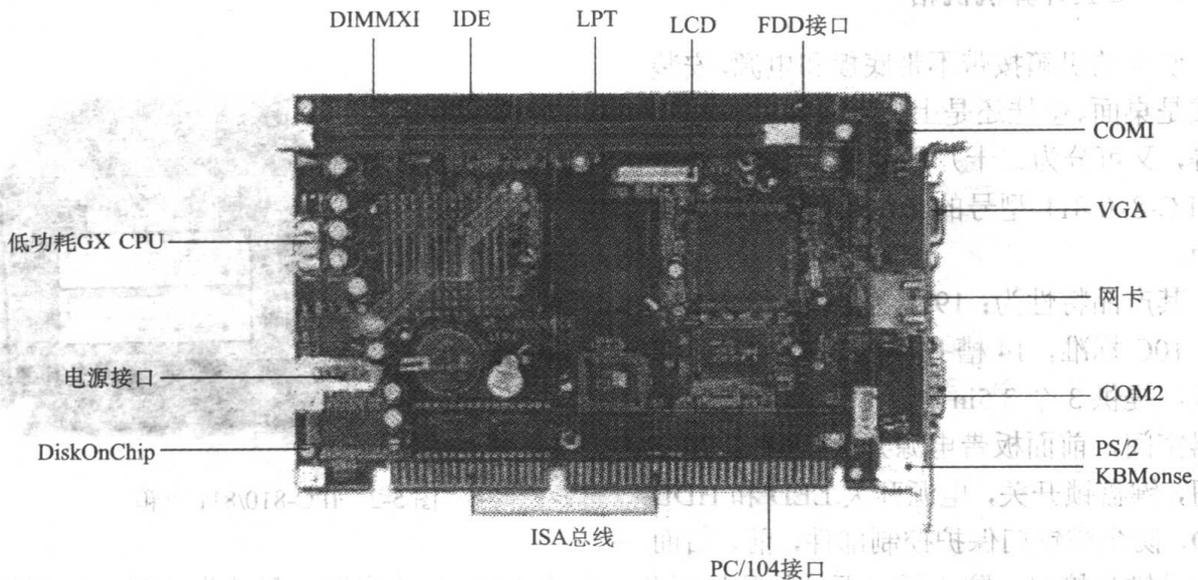


图 5-4 IPC-586VDNH (GX) CPU 卡

具体的产品规格如下。

总线类型: ISA 总线半长卡;

处理器: 板上增强型低功耗 NS_GXLV2.2V 200/233MHz CPU, 无需 CPU 风扇也可正常工作;

系统芯片集: NS_Cx5530A;

系统内存: 168 线 DIMM×1, 最大内存 128MB;

BIOS: Award PnP BIOS;

在板视频: NS_CX5530 支持 CRT/LCD 显示, 显示内存为 1.5~4MB, 分辨率为 1024×768×24 (CRT) 和 1024×768×18 (LCD), 18bit TFT LCD 接口;

在板 LAN: RTL8139C 10/100MB Ethernet 控制器, RJ-45 接口;

固态盘接口: M-System DiskOnChip flash 盘;

IDE 控制器: 一个 Ultra DMA33 通道, 可接 2 个设备;

USB 接口: 2 个 USB 接口;

多 I/O 接口: PC97317 I/O 接口芯片; 一个 FDD 接口, 一个并口; 一个 RS-232 接口, 一个 RS-232/422/485 接口, 一个键盘, 一个鼠标接口, 一个 115kbit/s IrDA 接口;

电源: +5V 单电源;

扩充总线: PC/104 扩充总线;

外形尺寸: 185mm×122mm;

工作温度: 0~60℃;

相对湿度: 5%~90%, 非凝结。

2. PCI 总线 CPU 卡

图 5-5 所示为研祥的 FSC-1718VN CPU 卡。采用 Intel 875P+6300ESB 系统芯片组的工业级 PICMG1.2 总线全长 CPU 卡, 性能卓越, 速度非凡, 支持高达 800MHz 系统总线, 支持 Socket478 Intel 超线程 Pentium 4 及 PRESCOTT 处理器, 配备两条 DDR 266/333/400 双通道 DIMM 插槽, 可支持最大达 2GB 系统存储器, Intel 独有的 PAT 技术支持。板上集成了一个 CSA 架构千兆以太网控制器、两个串行 ATA IDE 接口、四个 USB 2.0 接口、ATA 66/100 IDE 接口、二串一并等常用接口。FSC-1718VN 支持 64bit PCI-X 总线, 通过 PICMG1.2 规范的底板可连接 64bit 高速 PCI 设备, 是高速宽带应用性价比极高的解决方案, 可满足对高带宽应用日益增多的要求。FSC-1718VN 板载 ATI Radeon 9200SE 8X 高性能 AGP 图形加速控制器, 性能更优越, 64MB DDR 显示缓存。

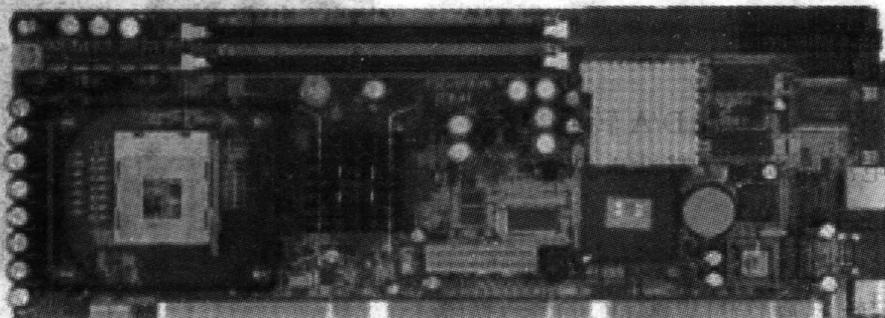


图 5-5 FSC-1718VN CPU 卡

具体的产品规格如下。

总线类型：完全兼容 PICMG 1.2 PCI/PCI-X 总线标准；

处理器：800MHz 系统总线，Socket478 Pentium 4 处理器；

系统芯片集：Intel 875P+6300ESB 芯片组；

系统内存：两条 184-pin DDR DIMM 插槽，支持双通道，系统存储器最大为 2GB；

BIOS：AMI BIOS，支持 ACPI 电源管理功能；

在板视频：以 ATI RADEON 9200SE 8X 为核心的 2D/3D 图形处理引擎，64MB DDR 显示缓存，最高刷新频率达 200Hz，最高分辨率可以达到 2048×1536 像素的 32bit 真彩显示，完全支持 DirectX 8.1、DirectX9.0 和 OpenGL 1.3 及 Pentium 4 的 SSE 内建 HDTV 解码器及硬件级的动态补偿技术

在板 LAN：Intel 1000Mbit/s 以太网控制器，RJ-45 接口；

IDE 控制器：两个 ATA 33/66/100 通道；

串行 IDE：两个 SATA IDE 150Mbit/s；

多 I/O 接口：一个 FDD 接口、四个 USB、一个并口、两个 RS-232、一个 IrDA 红外接口、一个 PS/2 键盘/鼠标接口；

看门狗定时器：256 级，可编程 1s（或 min）分辨率，可编程超时中断或系统复位；

硬件监测：对主要工作电压、CPU 温度和风扇转速进行监测；

电源：+5V、+12V，支持 ATX 电源；

外形尺寸：338mm×122mm；

工作温度：0~60℃；

相对湿度：5%~90%，非凝结。

5.4.4 I/O 卡

1. ISA 总线 I/O 卡

研祥的 ISA 总线的 I/O 卡，按模拟量输入路数、极性、采样速率、模拟量输出范围以及数字 I/O 通道的不同等，可分为十余种型号。图 5-6 所示为 PCL-728 隔离 2 通道模拟量输出卡。PCL-728 带两个独立的 D/A 转换器，可以同时被更新。每个 DAC 上有 12bit 分辨率和 16kHz 的吞吐量。每一个模拟输出通道都具备高电压隔离。光隔离器的 500V（DC）隔离保护 PC 和外设免受侵入高电压的损坏。在系统开机或复位时，两个通道的单极性和双极性输出电压都被复位为 0V。

(1) 特性

- ① 12bit 分辨率，双缓冲 D/A 转换器；
- ② 多电压/电流范围：双极性 ±5V，±10V；单极性 0~+5V，0~+10V；
- ③ 电流：0~20mA 或 4~20mA 电流环；
- ④ 提供稳定输出操作的集成 DC-DC 转换器；
- ⑤ 紧凑型，半长 PCB；
- ⑥ 4 层 PCB，带内部地层。

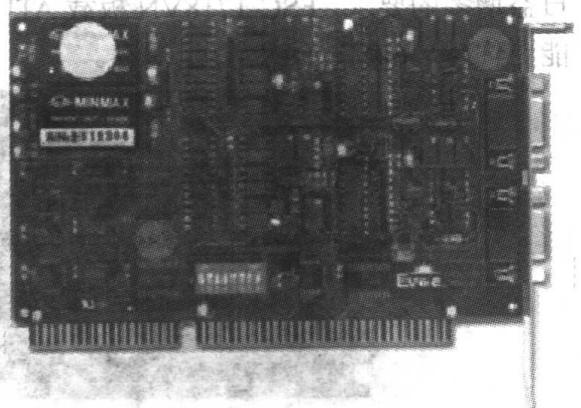


图 5-6 PCL-728 隔离 2 通道模拟量输出卡

(2) 技术说明

- ① 模拟量输出 (D/A);
- ② 通道: 2 个隔离通道;
- ③ 分辨率: 12bit, 双缓冲;
- ④ 稳定时间: 小于 60 μ s;
- ⑤ 吞吐量: 16kHz;
- ⑥ 输出范围: 单极性 0~+10V, 0~+5V; 双极性 \pm 5V 或 \pm 10V;
- ⑦ 电流环路: 0~20mA, 4~20mA;
- ⑧ 参考电压: 内部为-5V 或-10V; 外部为+10V 或-10V, AC 或 DC;
- ⑨ 隔离电压: >5000Vrms (通道对通道, 输入对输出);
- ⑩ 精度: 满量程的 \pm 0.012%;
- ⑪ 非线性: \pm 1 LSB;
- ⑫ 温度系数: $5 \times 10^{-6}/^{\circ}\text{C}$ 。

(3) 通用规格

- ① 接口: 两个 9 芯 D 型接口;
- ② 工作温度: 0~55 $^{\circ}\text{C}$;
- ③ 储存温度: -20~80 $^{\circ}\text{C}$;
- ④ 工作湿度: 5%~95%, 无凝结;
- ⑤ 功耗: +5V, 780mA 典型值, 最大为 1A;
- ⑥ 尺寸: 163mm \times 107mm。

2. PCI 总线 I/O 卡

研祥的 PCI 总线的 I/O 卡, 按模拟量输入路数、极性、采样速率、模拟量输出范围以及数字 I/O 通道的不同等, 可分为十余种型号。图 5-7 所示为 PCI-64AD 系列 I/O 卡

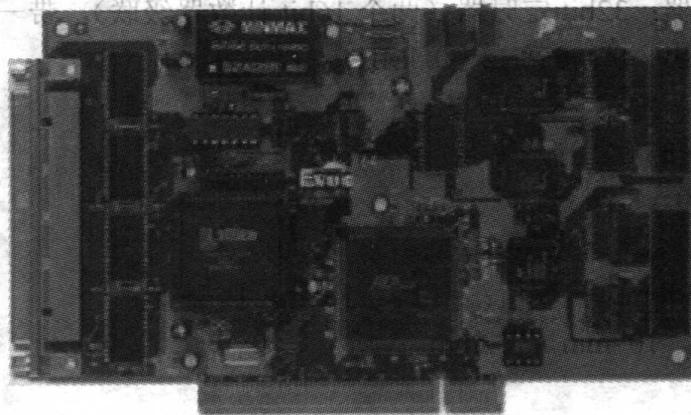


图 5-7 PCI-64AD 系列 I/O 卡

(1) 特性

- ① 32bit PCI 总线, 即插即用;
- ② 64 路单端或 32 路双端模拟输入通道;
- ③ 双极性模拟输入范围;
- ④ 板上 A/D 带有 1KB FIFO 内存;

- ⑤ 自动扫描通道选择;
- ⑥ 最高至 100kHz 采样速率;
- ⑦ 可编程增益 $\times 1$ 、 $\times 10$ 、 $\times 100$ 、 $\times 1000$;
- ⑧ 软件触发、定时器触发和外部触发 3 种触发模式;
- ⑨ 16 通道 DI 和 16 通道 DO;
- ⑩ 紧凑型, 半长 PCB。

(2) 技术说明

- ① 模拟量输入 (A/D);
- ② 转换器和分辨率: 12bit ADS774 或兼容芯片;
- ③ 通道数: 64 单端/32 双端;
- ④ 模拟信号输入范围 (软件控制);
- ⑤ 双极性: $\pm 10V$ 、 $\pm 1V$ 、 $\pm 0.1V$ 、 $\pm 0.01V$;
- ⑥ 转换时间: $8\mu s$;
- ⑦ 数据吞吐量: 最大为 100kHz;
- ⑧ 过电压保护: 连续, 最大 $\pm 35V$;
- ⑨ 精度: 增益为 1 或 10 时, 精度为 $1\% \pm 1 \text{ LSB}$; 增益为 100 或 1000 时, 精度为 $0.02\% \pm 1 \text{ LSB}$;
- ⑩ 输入阻抗: $10M\Omega$;
- ⑪ 触发模式: 软件、定时触发或外部触发;
- ⑫ 数据传输: 程序控制、中断;
- ⑬ FIFO 大小: 1024B。

(3) 可编程计数器

- ① 器件: 8254;
- ② A/D 定时触发器: 32bit 定时器 (两个 16bit 计数器级联), 带一个 2MHz 的时基;
- ③ 定时触发器输出: $0.00046\text{Hz} \sim 0.5\text{MHz}$ 。

(4) 通用规格

- ① 工作温度: $0 \sim 55^\circ\text{C}$;
- ② 储存温度: $-20 \sim 80^\circ\text{C}$;
- ③ 湿度: $5\% \sim 95\%$, 无凝结;
- ④ 功耗: +5V, 600mA 典型值; +12V, 100mA 典型值;
- ⑤ 尺寸: 紧凑型, $107\text{mm} \times 172\text{mm}$ 。

(5) 数字 I/O

- ① 输入通道数: 16DI;
- ② 输出通道数: 16DO;
- ③ 信号类型: TTL 兼容。

5.5 远程数据采集与控制模块

近几年开发出一种现场输入、输出模块, 由于其中内置了微处理器和采用了网络进行数

据通信,彻底改变了传统的模拟量 I/O、数字量 I/O 模板的结构方式,大大方便了分散型控制系统的构成。使用远程 I/O 模块时,模块可放在现场一次变送器处,将现场信号转换成数字信号,进行通信转换,而且各模块均是隔离的,其输出可以成组地连接在通信网络上,大大减少了现场接线成本。

比较具有代表性的远程智能数据转换模块是由研华工控公司生产的 ADAM-4000 系列(见图 5-8)和 ADAM-6000 系列产品,这两类产品类似,不同的是 ADAM-4000 系列的通信网络采用 RS-485 总线,而 ADAM-6000 系列采用工业以太网进行数据传输。表 5-3 列出了 ADAM-4000 系列的主要模块。

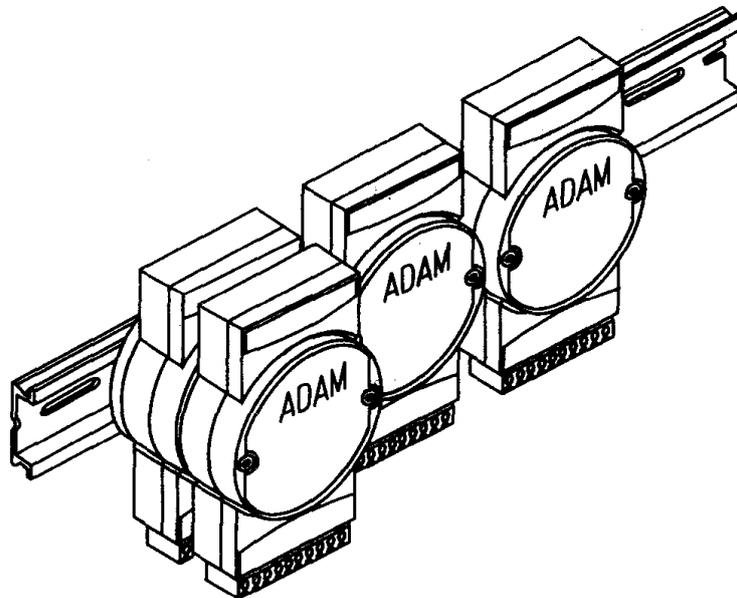


图 5-8 ADAM-4000 系列模块的导轨安装

表 5-3 ADAM-4000 系列远程 I/O 模块

ADAM-4011/4011D	热电偶输入模块 (mV、mA、V), 16bit, 双积分式 A/D 转换
ADAM-4012	模拟量输入模块 (mV、mA、V), 16bit, 双积分式 A/D 转换
ADAM-4014D	模拟量输入模块 (mV、mA、V), 16bit, Σ - Δ 式 A/D 转换
ADAM-4013	热电阻输入模块 (Pt、Ni), 3 通道, 双积分式 A/D 转换
ADAM-4016	应变输入
ADAM-4017/4018	8 路模拟量输入模块 (mV、mA、V), 16bit
ADAM-4021	模拟量输出 (mA、V), 12bit
ADAM-4050	数字量输入 (7 路) 输出 (8 路)
ADAM-4052	数字量输入 (8 路), 隔离
ADAM-4060	数字量输入 (4 路), 继电器输出 (4 路)
ADAM-4080/4080D	数字量输入 (2 路), 计数器/频率输入 (2 路)
ADAM-4520	RS-232/RS-422、485 转换器
ADAM-4510	RS-422/RS-485 中继器

注: 带 D 的型号, 具有 LED 显示功能; 单电源供电 DC 10~30V。

最基本的 ADAM 控制网络构成如图 5-9 所示，包括控制主机、RS-232/485 转换模块 ADAM-4520 和其他 ADAM I/O 模块。数据通信采用主从方式，PC 作为主机发出包含模块地址的预定义 ASCII 字符命令，而相应地址的模块对命令进行响应，并将处理结果以 ASCII 字符形式通过 RS-485 回送到主机。其中模块的数据采集是周期性地自动进行的，采集的结果保存在模块自己的缓冲区中，只有当主机需要数据时，模块才将当前的结果发送到主机。

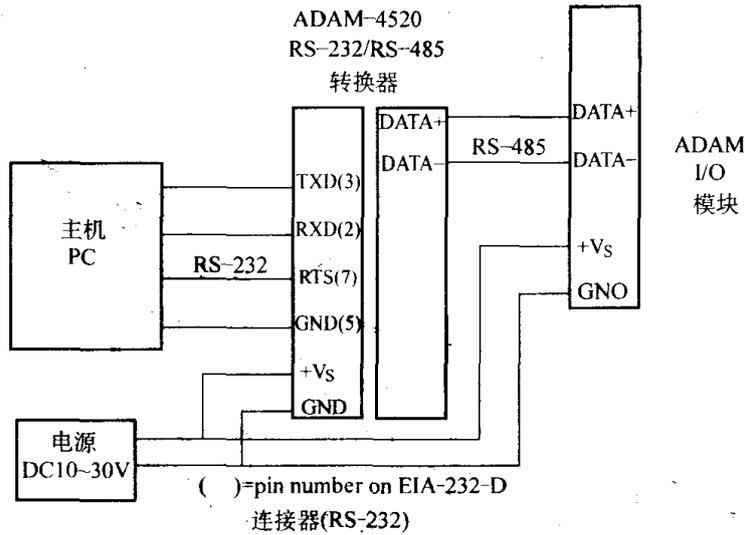


图 5-9 ADAM 模块与主机的网络连接

图 5-10 是由多个 ADAM 模块构成的 RS-485 控制网络的接线原理。

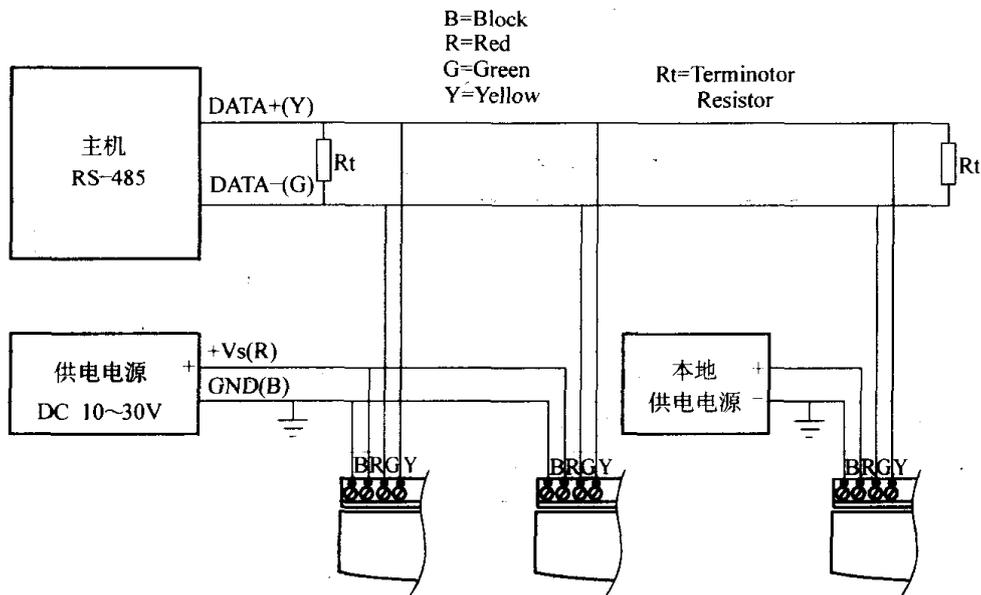


图 5-10 多个 ADAM 模块的连接

下面以最常用的模拟量输入模块 ADAM-4012 和模拟量输出模块 ADAM-4021 为例，对模块的内部结构及其性能指标具体给予介绍。

(1) 模拟量输入模块 ADAM-4012

图 5-11 所示为模拟量输入模块 ADAM-4012 的结构原理。

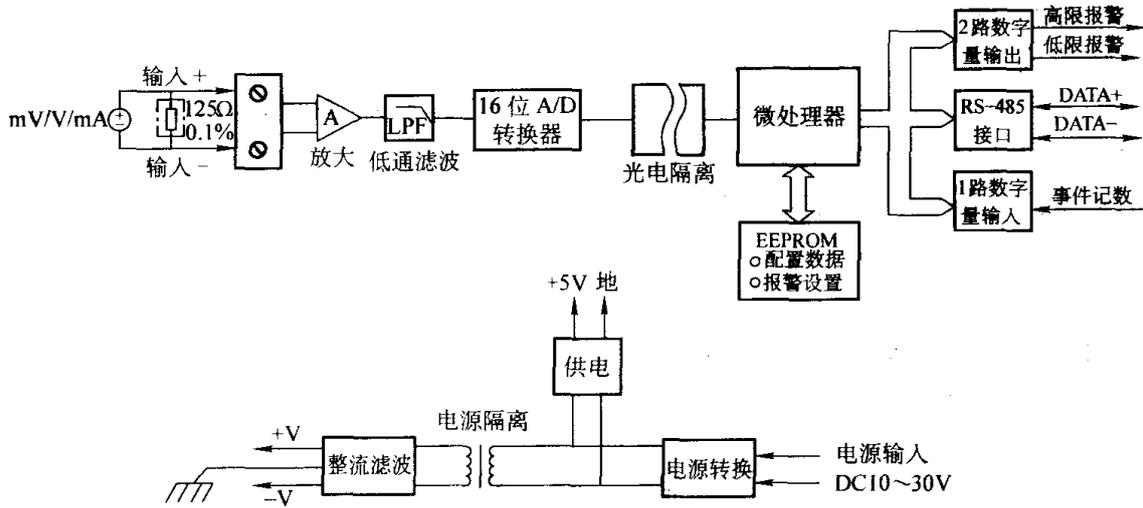


图 5-11 模块 ADAM-4012 的结构原理

主要性能指标:

- ① 输入范围: $\pm 150\text{mV}$ 、 $\pm 150\text{mV}$ 、 $\pm 1\text{V}$ 、 $\pm 5\text{V}$ 、 $\pm 10\text{V}$ 、 $\pm 20\text{mA}$ (接 125Ω 电阻);
- ② 输出: 通过 RS-485 输出 ASCII 字符结果, 传输速率为 1200bit/s 、 2400bit/s 、 4800bit/s 、 9600bit/s 、 19.2kbit/s 、 38.4kbit/s , 传输距离小于 1200m ;
- ③ 精度: $< \pm 0.05\%$;
- ④ 隔离电压: DC 3000V ;
- ⑤ $\text{CMR} > 150\text{dB}$;
- ⑥ $\text{NMR} > 100\text{dB}$;
- ⑦ 带宽: 4Hz ;
- ⑧ 转换速率: 10 次/s;
- ⑨ 数字量输出: 2 路;
- ⑩ 数字量输入: 1 路。

(2) 模拟量输出模块 ADAM-4021

图 5-12 所示为模拟量输出模块 ADAM-4021 的结构原理。

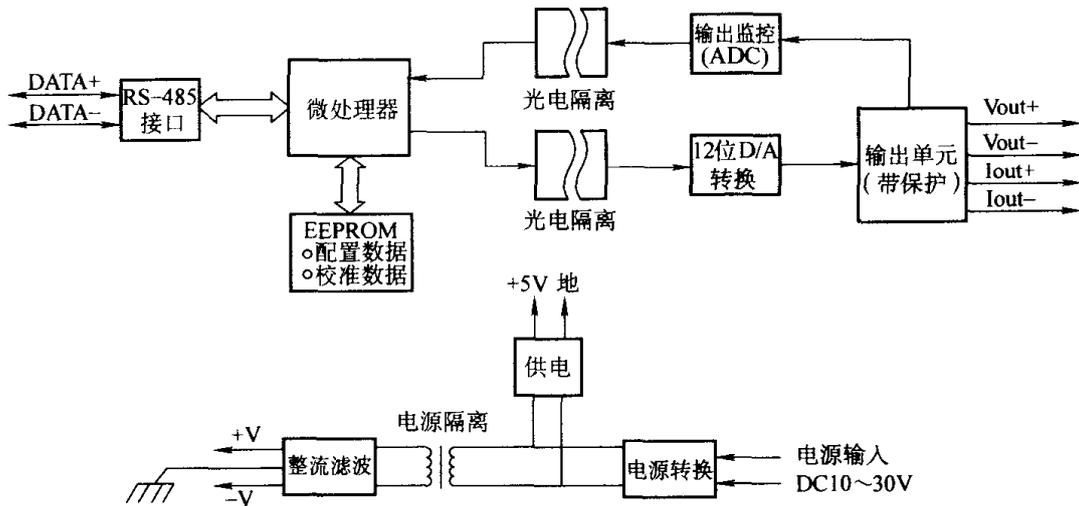


图 5-12 模块 ADAM-4021 的结构原理

主要性能指标如下。

- ① 输出范围：0~10V、0~20mA 或 4~20mA；
- ② 输入：通过 RS-485 输入 ASCII 字符命令，传输速率为 1200bit/s、2400bit/s、4800bit/s、9600bit/s、19.2kbit/s、38.4kbit/s，传输距离小于 1200m；
- ③ 精度：电压小于±0.2%、电流小于±0.1%；
- ④ 隔离电压：DC 3000V；
- ⑤ 输出阻抗：0.5Ω；
- ⑥ 电流输出带负载能力：0~500Ω；
- ⑦ 带宽：100 次/s。

思 考 题

1. 画图简要说明 IPC 的硬件组成。
2. 简述总线的概念。何为模板化结构？
3. 简要说明 IPC 与普通 PC 在硬件上有何不同？
4. 简述 ADAM 远程数据采集与控制模块的特点。

第 6 章 控制网络技术基础

随着计算机、通信、网络、控制等学科领域的发展，控制网络技术日益为人们所关注。控制网络，即网络化的控制系统，简称工控网。正是借助于计算机网络，多台原本单独存在的工业控制计算机被连成多处理器结构，使整个控制系统在实现分散控制的同时，还能够达到集中监视、集中管理和资源共享的目的。

控制网络技术源于计算机网络技术，因此，控制网络一方面与一般计算机网络有许多共同之处，另一方面又有其独特性，通常具有可靠性高、实时性好及互操作性好等特点。

6.1 数据通信基础

计算机网络就是利用通信线路和通信设备，把分布在不同地理位置上的具备独立功能的多台计算机、终端及其附属设备连接起来的一种网络，并配以相应的网络软件，使网络用户能够共享网络中的硬件、软件和数据等资源。

数据通信是计算机控制网络的基础。因此，首先应了解数据通信的基本概念。

6.1.1 数据通信系统

1. 通信系统构成

计算机与计算机或设备之间的数据交换称为数据通信。数据通信的实质是以计算机为中心，通过某些通信线路与设备，对二进制编码的字母、数字、符号以及数字化的声音、图像信息进行的传输、交换和处理。一个数据通信系统主要由以下 5 部分组成，如图 6-1 所示。

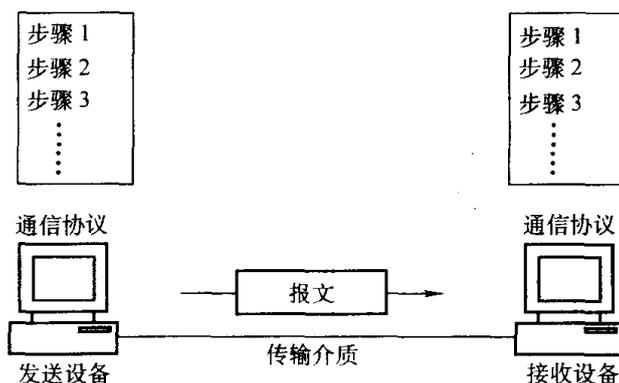


图 6-1 数据通信系统的构成

(1) 报文：报文即需要传送的数据，它可以是文本、数字、图像及声音等，或上述内容的组合。

(2) 发送设备：可以是计算机、工作站、电话机及摄像机等。

(3) 接收设备：可以是计算机、工作站、电话机及电视机等。

(4) 传输介质：发送设备与接收设备之间的物理通路，如双绞线、同轴电缆、光纤及无

线电波等。

(5) 通信协议：控制数据通信的一系列规则。发送与接收设备都要按相同的通信协议工作，如同两个谈话的人使用同一种语言一样。

数据通信过程一般按以下步骤进行。

(1) 数据打包：数据打包就是对需要传送的数据进行包装，形成数据包或报文。报文内除了数据本身外，还有报头、报尾等一些附加信息，如报文说明、长度、校验等。

(2) 数据转换与编码：数据转换与编码就是对报文作适当变换，以适应传输要求。如串行通信中的并-串转换，0、1 的传输编码（归零编码、不归零编码等）、信号电平的选择，以及信号的调制形式等。

(3) 数据传输：经过转换与编码后，将代表报文的信号（数字信号或模拟信号）放到传输介质上，发往接收设备。

(4) 数据转换与译码：接收设备将接收到的信号经转换与译码后，形成报文。

(5) 数据解包：接收设备根据数据打包时的协议从报文中去除附加信息，得到最终需要的数据。

2. 数据通信方式

数据通信的基本传输方式有并行通信和串行通信两种。

(1) 并行通信

并行通信是指所传送数据的各位同时发送或接收。数据有多少位，就需要多少根传输线，如图 6-2a 所示。计算机中表示数据的最基本单位是位 (bit)，但为了处理的方便与快捷，往往将多个位一起使用，形成一个字 (word) 或字节 (byte)。并行通信时常以字节为单位进行信号连接与数据传送，由于 1byte 为 8bit，故最少需要 8 条数据线。并行通信的特点是传送速度快，但由于连线较多致使成本高，只适合于近距离计算机或设备之间的数据通信，如计算机与打印机之间通常采用并行通信方式。另外，在过程控制系统中，处于同一机柜内的主计算机（上位机）与通道计算机（下位机）之间也常用并行通信方式传送数据。

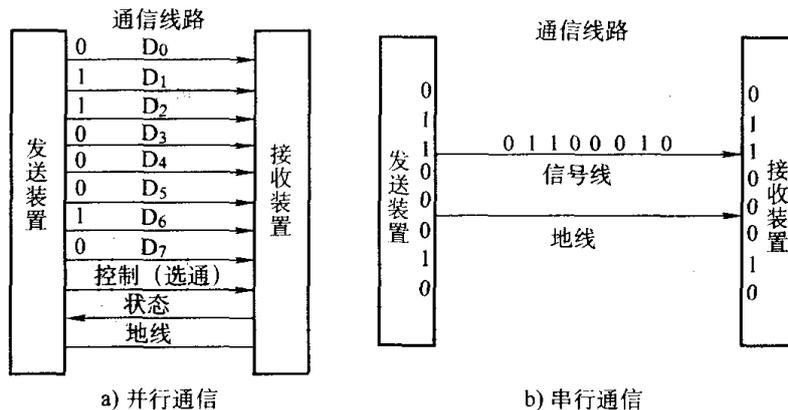


图 6-2 并行通信与串行通信

(2) 串行通信

串行通信是指所传送数据的各位按顺序一位一位地发送或接收，如图 6-2b 所示。其特点是只需一对传输线，适合于长距离传输，但通信速度较并行通信时慢。随着通信技术的发展，串行通信速度不断提高，使得计算机网络通信普遍采用串行通信方式。计算机控制系统中各

站间的数据传递及与信息管理系统间的数据交换都采用了串行通信方式。

波特率 (Baud Rate) 是串行通信中的一个重要指标。它定义为每秒传送二进制数码的位数, 单位是位每秒, 用 bit/s 表示。

在串行异步通信中, 波特率为每秒传送的字符数与每个字符位数的乘积。例如, 如果每秒传送的速率为 120 字符/s, 而每个字符包含 10bit (1 个起始位、7 个数据位、1 个位奇偶校验位、1 个停止位), 则波特率为

$$120 \text{ 字符/s} \times 10\text{bit/字符} = 1200\text{bit/s}$$

现在异步通信的波特率可达 100Mbit/s, 当采用光纤作为传输介质时, 传输波特率可以更高。

3. 数据通信制式

按通信线路上信息传送方向与时间的关系, 可分为 3 种通信制式, 分别为单工通信制式、半双工通信制式和全双工通信制式, 如图 6-3 所示。

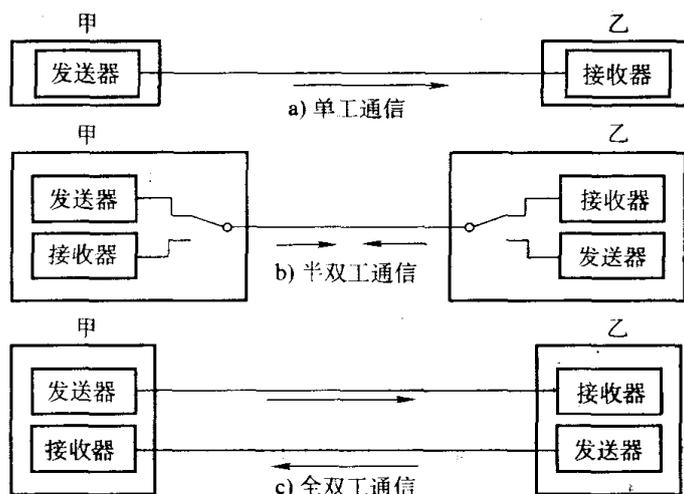


图 6-3 3 种通信制式

(1) 单工制式

在单工制式下, 通信线的一端接发送器, 另一端接接收器, 它们形成单向连接, 只允许数据按照一个固定的方向传送。如图 6-3a 所示, 数据只能由甲站传送到乙站。

(2) 半双工制式

在半双工制式下, 系统中的两个站都由一个发送器和一个接收器组成, 通过收、发开关接到一根通信线上, 如图 6-3b 所示。在这种制式下, 数据能从甲站传送到乙站, 也能从乙站传送到甲站, 但是不能同时在两个方向上传送, 只能交替地发送和接收。其收、发开关是由软件控制的电子开关, 通过半双工协议进行功能切换。

(3) 全双工制式

在全双工制式下, 两个站的每端都含有发送器和接收器, 通过两条通信线可以同时传送两个方向的数据流, 而不是交替传送, 如图 6-3c 所示。

6.1.2 数据传输编码

数据通信中需要传输数据信息, 而信息必须转换为数字的或模拟的信号后才能通过通信

线路传送，这种信息转换称为传输编码。它有多种方式，这里只讨论数字信息的数字信号编码与数字信息的模拟信号编码两种。

1. 数字信息的数字信号编码

在众多数字信息的数字信号编码方法中，这里仅介绍常用的几种：单极性编码方法、不归零编码方法及归零编码方法中的曼彻斯特码与差动曼彻斯特码，如图 6-4 所示。

单极性编码如图 6-4a 所示，它是编码中最简单、最原始的一种。单极性码用一种电平代表“1”，用另一种电平代表“0”，而且通常其中的一种电平（如“0”对应的电平）为 0V。但由于单极性码具有直流成分且不含同步信息，影响了单极性码的应用。

双极性编码如图 6-4b 所示，它使用正、负两种电平，因此有效地减小了传输线上的直流分量，但不归零码仍然没有彻底解决信号同步问题。

归零编码如图 6-4c 所示，它使用正、负、零 3 种电平，信号在数据位的中间发生变化。“正”到“零”的跳变代表“1”，“负”到“零”的跳变代表“0”。归零码较好地解决了信号同步问题，但由于每一位数据都要产生两次跳变，因此需占用更多的带宽。

曼彻斯特码如图 6-4d 所示，它也是在数据位的中间产生跳变，用该跳变的方向表示数值。“负”到“正”的跳变代表“1”，“正”到“负”的跳变代表“0”。该跳变还被用作信号同步，即编码数据中自带时钟信息，保证了收、发双方的绝对同步。以太网中就使用了曼彻斯特码。

差动曼彻斯特码如图 6-4e 所示，它用数据位中间的跳变携带同步信息，但由数据位起始处是否有跳变来传递数值。有跳变表示“0”，无跳变表示“1”。令牌环网中就使用了差动曼彻斯特码。

图 6-4f、g 分别表示接收端从编码数据 6-4e 中分离出的解码时钟与解码数据。

2. 数字信息的模拟信号编码

数字信号传送时要求传输线的频带很宽，而在长距离通信时，通常是用电话线进行传送的，其带宽往往不能满足要求，这时会产生信号的畸变，而且数据通信速率越高信号畸变越严重。

一种解决方案是采用数字信息的模拟信号转换技术将数字信号传输变为模拟信号传输，称为信号调制。在数据接收端还要将模拟信号复原为数字信息，称为解调，如图 6-5 所示。完成调制与解调的装置称为调制解调器 (MODEM)。

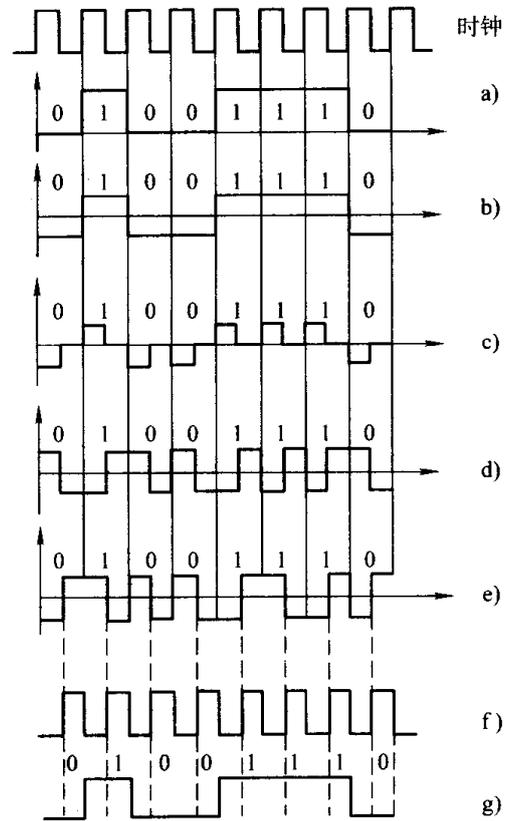


图 6-4 数字信息的数字信号编码

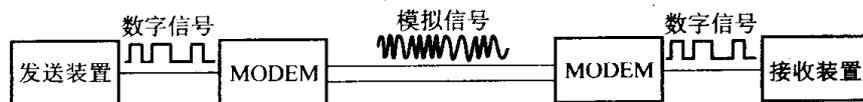


图 6-5 调制与解调

将数字信号调制为模拟信号有3种方式，即调幅、调频与调相，如图6-6所示。

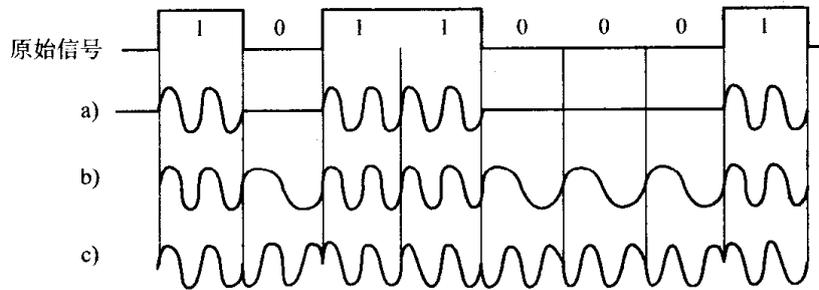


图 6-6 三种调制信号

调幅又称调幅键控 (ASK, Amplitude Shift Keying), 就是用原始数字信号去控制载波的振幅变化, 这种调制是利用数字信号的 1 或 0 去接通或断开连续的载波。图中, 正常幅度的正弦波表示 1, 幅度为 0 的表示 0。调幅方式很容易受干扰信号的影响, 因此仅用于非常低速的信号传输, 或与其他调制方式 (如相位调制) 结合使用。

调频又称调频键控 (FSK, Frequency Shift Keying), 就是用原始数字信号去控制载波的频率变化, 如用一种频率的正弦波信号表示 1, 而用另一种频率 (通常为低频) 的正弦波信号表示 0。调频方式可以削弱干扰对信号的影响, 但它会占用较宽的频带, 受传输介质制约也较大。

调相又称调相键控 (PSK, Phase Shift Keying), 就是用原始数字信号去控制载波的相位变化, 最简单的调相方法是使正弦波相位相差 180° 来表示 0 和 1。图中, 相位差为 0° 的正弦波信号表示 1, 相位差为 180° (反相) 的正弦波信号表示 0, 这种方式称为 2-PSK 方式。调相方式比调幅方式有较好的抗干扰能力, 比调频方式节省带宽, 因此得到广泛应用。

更复杂的调制称为正交调制 (QAM), 它是 ASK 与 PSK 结合形成的一种调制方法。QAM 方式可以有效提高数字信号的传送速率, 也是目前 MODEM 中普遍使用的方法。

3. 基带传输与频带传输

按照传输线上信号的种类, 数据通信可以分为基带传输与频带传输。

所谓基带是指电信号所固有的频带。基带传输是指直接用电脉冲信号代表数字信号 0 或 1 进行传输。基带传输的优点是安装、维护投资小, 但存在信息传送容量小, 每条传输线只可传送一路信号且传送距离短的缺点。

所谓频带传输是指用基带信号对载波信号调制后进行传输。利用载波传输信号可以减小线路干扰对传输信号的影响, 实现远距离传输。采用这种传输技术, 可以在一条通信线路上, 通过频分复用 (FDM) 将其划分为几个信道, 从而支持多路信号的传输。频带传输的安装维护费用较基带传输高, 但是它具有较高的信息传递量, 且覆盖范围较大。

6.1.3 多路复用技术

所谓多路复用技术是指把多路独立信号在一条信道上进行传输的技术, 其作用相当于把单条传输信道划分成多个子信道, 以实现网络中若干节点共享通信信道的目的, 提高通信线路的利用率。通常的多路复用技术主要有两种, 为频分多路复用和时分多路复用。

1. 频分多路复用

频分多路复用 (FDM) 技术是将单个物理信道的频谱分割成若干个互不重叠的小频段,

每个小频段都可以看作是一个子信道,如图 6-7 所示。从图中可见,每个小频段是互不重叠的,而且相邻频段之间留有一空闲频段,以保证数据在各自频段上可靠地传输。这种方法实现时需要使用多个 MODEM。

采用多路复用技术时,数据在各个子信道上以并行方式传输,即可以在各个子信道上同时传输不同的信号。当然,也可以将一个符号的各位在不同的信道上进行传输,相当于一个符号以并行方式进行传输。

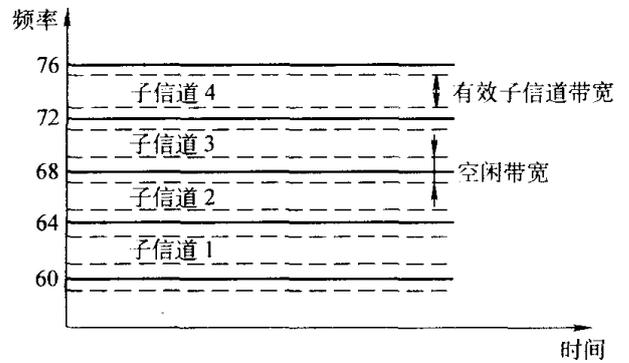


图 6-7 物理信道的频谱分割

传输信号的带宽与分配给每个子信道的带宽之间的转换是采用调制技术进行的。从图 6-8 可见,一个频率多路复用系统是由若干个并行通路组成的,每个通路均由调制器与相应的滤波器组成,由于各个通道是独立的,所以,一个通道发生故障不至于影响其他通道。终端的输出首先进入一个低通滤波器,目的在于抑制信号中高次谐波的作用,然后再进入调制器。信号经过调制后进入带通滤波器,带通滤波器的作用在于防止各个相邻子信道间发生干扰现象。各个子信道上的信号同时馈送至通信信道上,从而实现多路信号共享一个公用的通信信道的目的。

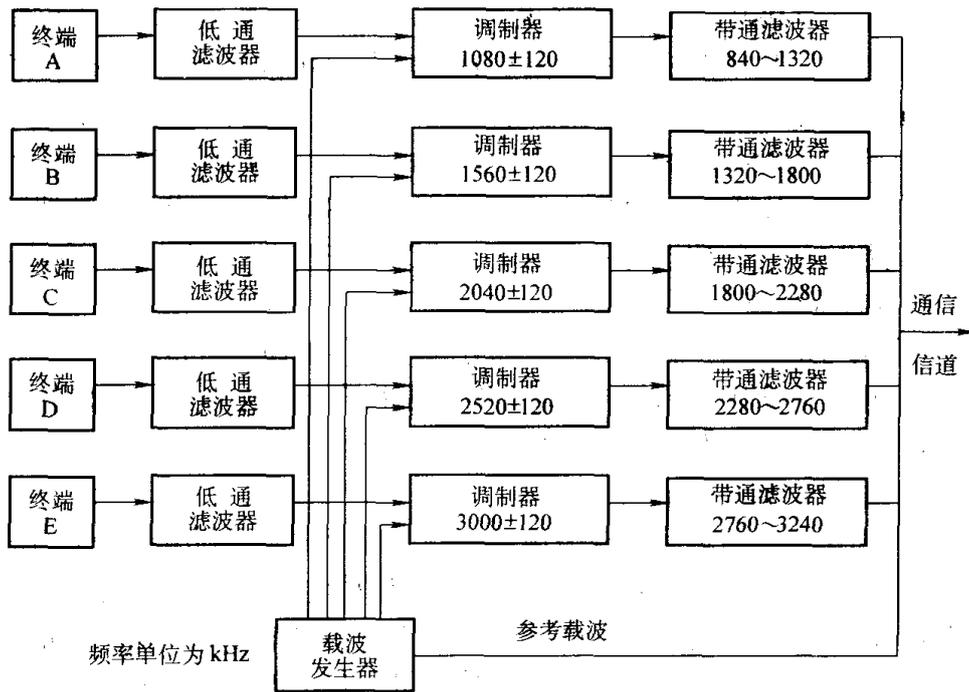


图 6-8 频分多路复用系统的组成

2. 时分多路复用

时分多路复用 (TDM) 技术是把信道的传输时间分隔成许多时间段,在一个时间段内,一路信号占用整个信道的带宽进行传输,其原理如图 6-9 所示。信号的传输可以按比特、按符号或按成组方式进行。为了在接收端能够对复合信号进行正确的分离,接收端与发送端的时序必须严格同步,否则将造成信号间的混淆。

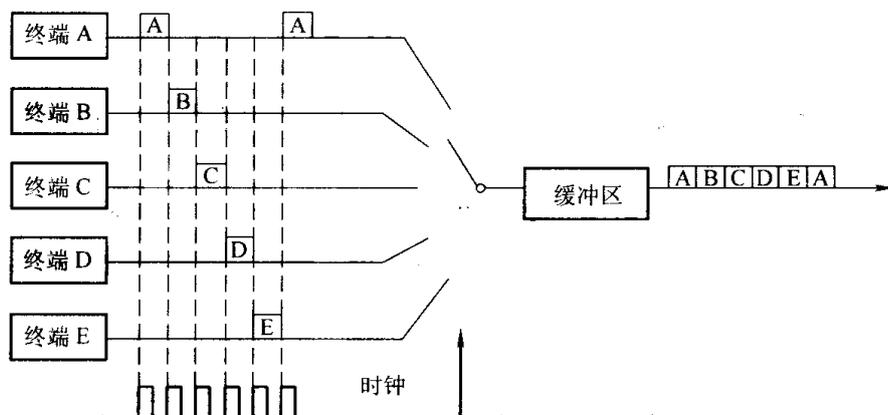


图 6-9 时分多路复用系统原理

3. 两种复用技术的比较

(1) TDM 比 FDM 易于实现, 而且随着大规模集成电路的发展, TDM 的价格会有明显下降。

(2) TDM 较 FDM 具有较高的传输速度。因为在 TDM 中, 可以充分利用信道的全部带宽。

(3) 在 TDM 中, 只需要一个 MODEM 就可以了; 而在 FDM 中, 每个通道均需要一个 MODEM。

(4) 在 FDM 中, 通常需要 A/D 转换设备; 而在 TDM 中, 由于具有明显的数字形式, 因此特别适用于与计算机直接相连的系统。

(5) TDM 能混合不同速率和同步方式的终端, 能适应新型数据通信网。

(6) 在进行数据传输的差错控制和校正操作时, TDM 比 FDM 会产生较多的时间延迟。

FDM 技术在模拟通信系统中应用较多, 在数字通信系统中应用较少。这是因为, 在不同的调制解调器载波频带之间应有一定的间隔, 以防止相互干扰, 从而使线路的利用率降低。另外由于超大规模集成电路技术的发展, 研制 TDM 器件比研制 FDM 器件容易, 成本较低, 故在数字通信系统中, 将主要采用 TDM 技术。

采用多路复用技术有许多优点: 一是成本较低; 二是时间延迟较少 (因为线路中用的缓冲部件较少); 三是透明性好, 从主机或通信处理机到终端是透明的, 用户并未感到多路器的存在。其不足之处是采用多路复用技术后, 线路的带宽和传送速率未能充分利用, 另外对主机 I/O 端口的要求并未减少。

6.1.4 通信同步技术

无论是并行通信还是串行通信, 数据都是按时间顺序传送出去的。为保证发送与接收过程正确无误, 必须使用同步技术, 即使接收端和发送端在时间上取得一致; 否则将会导致通信质量下降, 甚至完全不能工作。

1. 通过控制线实现收、发双方同步

所谓并行同步通信方式, 是指在并行通信中通过控制线实现收、发双方同步。数据收、发双方除了数据线相连外, 还有若干控制信号线, 用来传送发送与接收装置的状态。对于图 6-2a 所示并行通信连接方法, 进行一次数据传送的过程如下。

(1) 发送装置在发送前，首先检查接收装置的状态。

(2) 若接收装置处于非就绪状态，返回(1)继续等待；若接收装置处于就绪状态，表示可以接收数据，发送装置将数据放到数据线上，并发出控制信号，告诉接收装置数据已准备好。

(3) 接收装置收到控制信号后，从数据线上将数据取走。在取数据过程中，状态线置为非就绪状态，取数完成后，状态线重新处于就绪状态，等待下一次接收数据。

(4) 发送装置将控制信号撤回，准备发送下一次数据。

而在串行通信中，往往只有一对数据线，没有控制信号线。此时收、发双方的同步是靠同步信号来实现的，而且一对传输线同时完成传送数据与同步信号双重任务。串行通信中的同步方式又有两种：一种是启停同步方式，与其相对应的传输方式称为异步通信方式；另一种是自同步方式，与其相对应的传输方式称为同步通信方式。

2. 串行异步通信

串行异步通信 (ASYNC, Asynchronous Data Communication) 是一个字符一个字符地按帧传送数据的方式，其传送一帧的格式如图 6-10 所示。开头一个起始位“0”，接着是 5~8 个数据位，且规定低位在前、高位在后，然后是一个奇偶校验位，最后加上一个停止位“1”表示字符的结束。若数据没有准备好则以空闲字符“1”来填充，直到数据准备好形成下一帧。

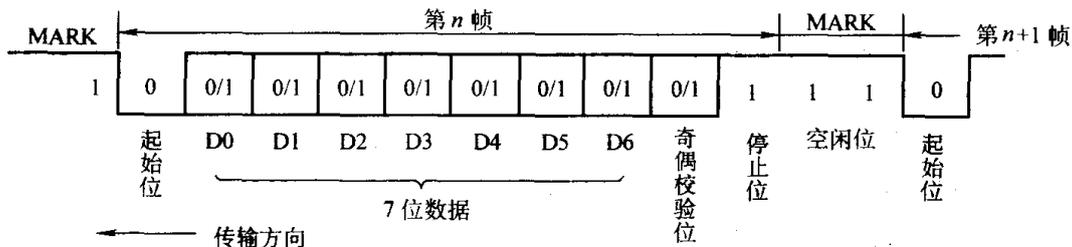


图 6-10 串行异步通信的信息帧格式

图 6-10 中说明，一帧信息包含 1 个起始位、5~8 个数据位、1 个奇偶校验位、1 或 2 个停止位。当无信号传送时，为停止位（高电平）状态，当出现起始位（低电平）时，表示数据传送开始。因此停止位到起始位的电平转换，即为同步信息。

进行异步通信时，收、发双方必须有两项约定：一是帧信息格式，即字符的编码形式、奇偶校验形式、起始和停止位的格式等；二是传送速率。

串行异步通信在向外发送字符时，由于在字符的首尾分别附加了一个起始位和停止位，因此，传送效率较低，传送速率较慢。能够完成异步通信的硬件称为 UART (Universal Asynchronous Receiver/Transmitter)，典型的 UART 接口电路芯片有 Intel8250 及 MC6850 等。

3. 串行同步通信

串行同步通信 (SYNC, Synchronous Data Communication) 是一种连续传送数据块的方式，每次传送 n 个字节的数据块。用 1 或 2 个同步字符表示数据传送的开始，接着是 n 个字节的数据块，字符之间不允许留空隙，当没有字符可发送时，则连续发送同步字符，如图 6-11 所示。

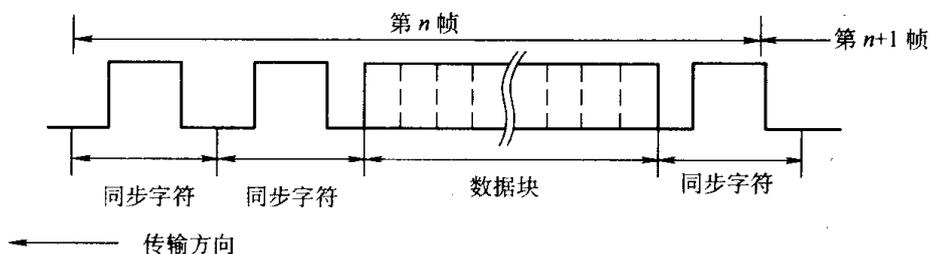


图 6-11 串行同步通信的信息帧格式

同步字符可以选择一个特殊的 8bit 二进制码, 如 01111110 作为同步字符(称单同步字符), 或两个连续的 8bit 二进制码作为同步字符(称双同步字符)。为了保证收、发双方同步, 收、发双方必须使用相同的同步字符, 而且往往采用可以生成解码时钟的编码方式。

串行同步通信的传送速率高于串行异步通信, 且传送的数据块越长, 越能显示其优越性。能够完成同步通信的硬件称为 USRT (Universal Synchronous Receiver/Transmitter)。既能够完成异步通信, 又能够完成同步通信的硬件称为 USART, 典型的 USART 接口电路芯片有 Intel8251 等。

6.1.5 常用传输介质

传输介质是连接站与站间的物理信号通路。目前使用的传输介质主要有 3 种, 即双绞线、同轴电缆和光纤。表 6-1 给出了这几种传输介质主要性能的比较。

表 6-1 传输介质主要性能比较

项 目 \ 类 型	双绞线	基带同轴电缆	宽带同轴电缆	光 纤
传输信号	数字、模拟	数字	数字、模拟	模拟
最大带宽	100kHz~1MHz	10~50MHz	300~400MHz	实际不受限制
互连复杂性	不复杂	不太复杂	较复杂	复杂
噪声抑制能力	外层有屏蔽, 较好	好	好	非常好
最大传输距离	100m	2.5km	300km	100km

6.2 通信网络技术

按网络作用范围的大小, 计算机网络可以分成两大类: 一类是分布在广大地理范围内的广域网; 另一类为分布在一定区域内的局域网。计算机控制系统的通信网络属于后一类。决定局域网特性的要素有传输介质、拓扑结构和介质访问控制方法。

6.2.1 网络拓扑结构

在计算机网络中, 抛开网络中的具体设备, 将工作站、服务器等网络单元抽象为“点”, 将网络中的电缆等通信介质抽象为“线”, 这样从拓扑学 (topology) 的观点观察计算机通信网络系统, 就形成了点和线组成的几何图形, 从而抽象出了网络系统的具体结构。因此, 计算机通信网络的拓扑结构, 是指网络中的各台计算机以及设备之间相互连接的方式。常见的

网络拓扑结构有星形、环形和总线型 3 种，如图 6-12 所示。

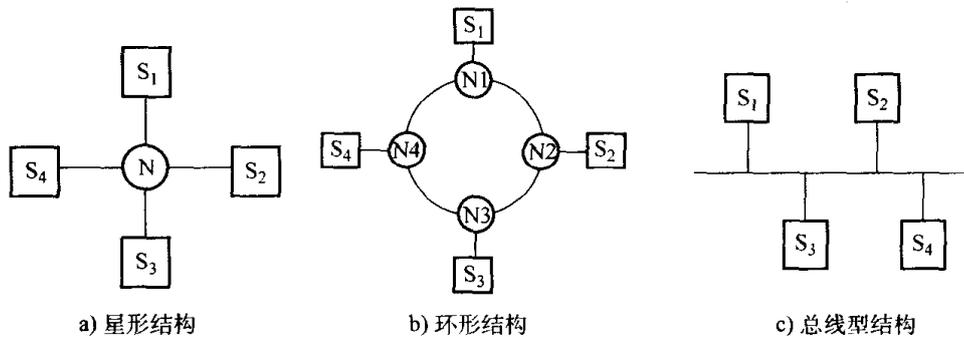


图 6-12 网络拓扑结构

星形结构是将分布于各处的多个站 ($S_1 \sim S_n$) 连到处于中心位置的中央节点 (N) 上，任何两个站的通信都要通过中央节点。这种连接方式简单直观，但中央节点负荷较重，势必造成中央节点复杂。另外，一旦中央节点发生故障将造成系统通信中断。为提高网络的可靠性，常采取中央节点冗余的方法。

环形结构中每个站都是通过节点（或称中继器）连接到环形网上，所有的节点共享一条物理通道，信息沿单方向围绕环路进行循环，按点对点方式传输。由一个工作站发出的信息传递到相邻的下一节点，该节点对信息进行检查，若不是信息目的站，则依次向下一节点传递，直至到达目的站。为避免某节点发生故障会阻塞信息通路，环形网各节点应有旁路措施。

总线型结构中，网上所有节点通过硬件接口直接连到一条公共通信线路上，任何站都可随机发送信息，并能被其他站所接收。总线型网络具有易于扩充、可靠性高等特点。但因为总线为所有站共同使用，为避免发送冲突，应规定介质访问控制协议来分配信道，以保证在任一时刻只有一个节点发送信息。

6.2.2 网络控制方法

网络控制方法就是研究在通信网络中，信息如何从源站迅速、正确地传递到目的站。网络控制方法与所使用的网络拓扑结构有关，常用的有查询、令牌传送、CSMA/CD、时间片寄存器输入及存储转发等，下面仅介绍其中几种。

1. 查询法

查询法用于主从结构网络中，如星形网络或具有主站的总线型网络。主站依次询问各站是否需要通信，收到通信应答后再控制信息的发送与接收。当多个从站要求通信时，按站的优先级安排发送。

2. CSMA/CD

CSMA/CD (Carrier Sense Multiple Access With Collision Detection)，即带有冲突检测的载体监听多重访问技术，是一种竞争方式，适用于总线型网络结构。在这种方式中，网上各站是平等的，任何一个站在任何时刻均可以广播的形式向网上发送信息。信息中包含有目的站地址，其他各站接收到后确定是否为发给本站的信息。由于总线型结构网络中线路是公用的，因此竞争发送所要解决的问题是当有多个站同时发送信息时的协调问题。CSMA/CD 采取的控制策略是竞争发送、广播式传输、载体监听、冲突检测、冲突后退及再试发送。

当工作站有数据需要发送时,首先监听线路是否空闲,若空闲,则该站就可发送数据。载体监听技术虽然能够减少线路冲突,但还不能完全避免冲突。如两个工作站同时监听到线路空闲时,会同时发送数据,造成数据作废。解决的办法是在发送数据的同时,发送站还进行冲突检测,当检测到冲突发生时,工作站将等待一段随机时间再次发送。

CSMA/CD 遵守 IEEE802.3 标准。

3. 令牌传送

这种方式中,有一个称为令牌(token passing)的信息段在网络中各节点间依次传递。令牌有空、忙两种状态,开始时为空。节点只有得到空令牌时才具有信息发送权,同时将令牌置为忙。令牌绕节点一周且信息被目标节点取走后,令牌被重新置为空。

令牌传送既适合于环形网(称为令牌环——token ring),又适合于总线型网(称为令牌总线——tokenbus)。在总线型网情况下,各站被赋予一逻辑位置,所有站形成一个逻辑环。令牌环遵守 IEEE802.5 标准,令牌总线遵守 IEEE802.4 标准。

令牌传送具有效率高、信息吞吐量大及实时性好等特点。

4. 存储转发式

存储转发式的信息传送过程为:源节点发送信息,到达它的相邻节点;相邻节点将信息存储起来,等到自己的信息发送完,再转发这个信息,直到把此信息送到目的节点;目的节点加上确认信息(正确)或否认信息(出错),向下发送直至源站;源节点根据返回信息决定下一步动作,如取消信息或重新发送。

存储转发式不需要交通指挥器,允许有多个节点在发送和接收信息,信息延时小,带宽利用率高。

6.2.3 差错控制技术

由于通信线路周围电磁干扰的存在,以及收发器件噪声的影响,信息在发送、接收及传递过程中难免出现差错。通信网络的差错控制技术就是要及时将差错检测出来,并采取适当的纠正措施,以确保所接收信息的准确性。差错控制技术包括检验错误与纠正错误。

1. 检验错误

检测传输错误最简单的方法是将同一数据发送多次,在接收端进行逐位比较。但这样做是很不经济的。一方面降低了传送速率,另一方面逐位比较也浪费时间。当前检错中普遍使用的冗余技术也利用了这种思路,只不过冗余数据量要小得多。

源节点在发送数据时,除基本数据外,还包含附加校验位,附加校验位与基本数据有一定关系,如为基本数据按指定规则的运算结果。目的节点接收到数据后,仍按相同规则对基本数据进行计算并将计算结果与接收到的附加校验位相比较,若二者相同,则认为所接收数据正确,否则认为所接收数据错误。

不难理解,校验位越多,校验准确性越高,但传送效率越低。

常用的校验方法有垂直冗余码校验或奇偶校验(VRC)、纵向冗余码校验(LRC)、循环冗余校验(CRC)以及校验和(check sum)等几种。下面简要介绍奇偶校验和循环冗余校验的原理和基本方法。

(1) 奇偶校验

奇偶校验是在传递字节最高位后附加一个校验位。该校验位根据字节内容取 1 或 0,奇

校验时传送字节与校验位中“1”的数目为奇数，偶校验时传送字节与校验位中“1”的数目为偶数。接收端按同样的校验方式对收到的信息进行校验。如发送时规定为奇校验时，若收到的字符及校验位中“1”的数目为奇数，则认为传输正确；否则，认为传输错误。

例如，采用偶校验方法传送一个字节 0110 0001，发送端在字节最高位后添加校验位 1，使带校验位数据中 1 的个数为偶数，即 10110 0001。接收端收到该数据后，同样检查数据中 1 的个数。若为偶数，认为数据正确，将 0110 0001 取走；若为奇数，表明有传输错误，需采取进一步措施。

不难发现，奇偶校验只能检测出单个信息位出错而不能确定出错位置，因此这种校验方式检错能力低。

(2) 循环冗余校验

采用循环冗余校验时，发送端发送的信息由基本信息位与校验冗余位两部分组成。发送端在发送基本信息位的同时，发送端的 CRC 校验位生成器自动生成 CRC 校验位（由基本信息除以所谓生成多项式 $G(x)$ 而得），一旦基本信息位发送完，就将 CRC 校验位紧随其后发送。接收端用接收到的基本信息及校验位除以同一多项式 $G(x)$ ，如果这种除法的余数为 0，即能被除尽，则认为传输正确；否则为传输错误。

与奇偶校验不同，循环冗余校验是一个数据块校验一次。在同步串行通信中，几乎都采用循环冗余校验，如对磁盘信息读/写的校验等。

常用的生成多项式 $G(x)$ 主要为

$$\text{CRC-16:} \quad G(x) = x^{16} + x^{15} + x^2 + 1$$

$$\text{SDLC:} \quad G(x) = x^{16} + x^{12} + x^5 + 1$$

任何一个由二进制数组成的代码都可以和一个只包含 0 和 1 两个系数的多项式建立一一对应关系，如代码 1011011 对应的多项式为

$$1x^6 + 0x^5 + 1x^4 + 1x^3 + 0x^2 + 1x^1 + 1x^0 = x^6 + x^4 + x^3 + x + 1$$

而多项式 $x^5 + x^4 + x^2 + x$ 与代码 110110 对应。一般地，一个 n 位代码和一个 $n-1$ 次多项式对应。

现假定， k 位要发送的信息对应的 $k-1$ 次多项式为 $K(x)$ ， r 位 CRC 校验位对应的 $r-1$ 次多项式为 $R(x)$ 。由信息位产生 CRC 校验位的过程是一个已知 $K(x)$ 求 $R(x)$ 的过程。具体做法是：用一个事先约定的 r 次生成多项式 $G(x)$ 去除 $x^r K(x)$ ，把所得的余式作 $R(x)$ 。由此构成的 $k+r$ 位发送代码所对应的 $k+r-1$ 次多项式为

$$T(x) = x^r K(x) + R(x)$$

注意，这里的除法是指所谓的二进制除法。

设 $G(x)$ 除 $x^r K(x)$ 的商式为 $Q(x)$ ，则 $x^r K(x) = G(x)Q(x) + R(x)$ 。若传输无误，则接收到的代码所对应的多项式也为 $T(x)$ ，因为

$$T(x) = x^r K(x) + R(x) = G(x)Q(x) + R(x) + R(x) = G(x)Q(x)$$

即 $T(x)$ 能被 $G(x)$ 整除。这样接收端的校验过程可用 $G(x)$ 去除收到的代码的多项式来实现。

CRC 校验码在发送端的产生和接收端的校验，目前一般都是由硬件 CRC 校验电路自动实现的，当然也可以通过软件实现，但此时通信速度会受到软件执行时间的限制。

2. 纠正错误

常用的纠错方式有3种,即重发纠正错误、自动纠正错误及混合纠正错误。

(1) 重发纠正错误:当接收端检测出接收错误时,会以适当方式将检测结果反馈给发送端,发送端重新发送该信息。这种过程可以重复多次,直至接收端接收正确为止。

(2) 自动纠正错误:发送端在发送数据时,还带有能够纠正错误的信息码。接收端检测出错误后,按纠错码自动进行纠正。这种方式要考虑纠错能力与发送效率间的权衡。

(3) 混合纠正错误:这是上述两种方式的综合。当接收端检测出错误时,若判断为在纠错能力之内,则进行自动纠错;否则进行重发纠错。

理论上,任何错误都是可以自动纠正的。但实际上由于自动纠错需要较多的冗余信息,且纠错算法复杂,因此自动纠错仅限于错误位数较少时的情况。

6.3 OSI 模型

计算机网络就是要实现资源共享。但是,由于挂在网上的计算机或设备可能出自于不同的生产厂,型号也不尽相同,这样硬件和软件上的差异就会给通信带来困难。因此,在网络中应有一系列供全网“成员”共同遵守的有关信息传递的人为约定,以实现正常通信和共享资源,这就是通信网络协议或称规范,其最好的组织方式是层次结构模型。因此,计算机网络层次结构模型与各层协议的集合被定义为计算机网络体系结构。在计算机网络的发展过程中,许多制造厂商均发表了各自的网络体系结构以支持本公司的计算机产品的联网,但其通用性差,不便于不同厂商的网络产品进行互联。

为此,国际标准化组织 ISO 于 1977 年成立了专门机构研究该问题。不久,他们就提出了一个试图使各种计算机在世界范围内互联成网的标准框架,这就是著名的开放系统互联参考模型(OSI/RM, Open System Interconnection/Reference Model),简称为 OSI,从而形成了网络体系结构的国际标准,使得任何两个遵守 OSI 协议的系统可以相互连接。

OSI 模型将数据传输过程分解为一系列功能元素,把相关的功能组合在一起称为层,每一层完成一项通信子功能,并且下层为上层提供服务。分层结构具有易于理解和使用灵活的特点,更重要的是,OSI 模型使得不兼容系统之间的通信是透明的。

OSI 参考模型由七层组成,从下至上分别为物理层、数据链路层、网络层、传输层、会话层、表示层及应用层。如图 6-13 所示。

在传送一组数据(报文)时,要经过第 7~2 层的封装,到第 2 层形成的信息整体称为帧(frame)。所谓封装实际上就是在原始数据上附加报头 H 或报尾 T,一般在第 7、6、5、4、3、2 层上加报头,在第 2 层上加报尾。封装后的信息由物理层放到通信线路上传输。接收过程中分别去除报头与报尾,最后获得所需数据。图 6-13 中的 L3~L7 表示相应层的数据,H7~H2 代表相应层上附加的报头,T2 为第 2 层上附加的报尾。

下面简要介绍 OSI 模型各层的基本含义。

1. 物理层

物理层用来提供通信设备的机械特性、电气特性、功能特性和过程特性,并在物理线路上传输数据位流,如规定“1”与“0”的电平值、1 位数据的宽度、连接器引脚数及其含义、数据收发的时序等。物理层标准的例子有 RS-232C、RS-499/422/423/485 等。

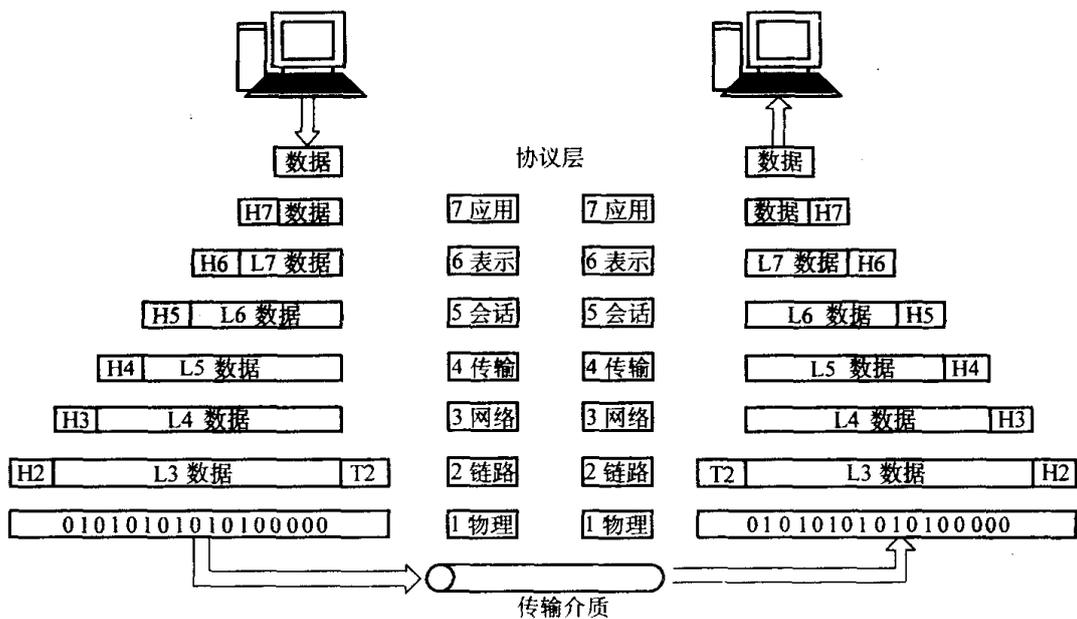


图 6-13 OSI 模型

物理层所关心的内容主要包括以下几点。

- (1) 线路结构。两个或多个设备是如何物理相连的？线路共享还是独占？
- (2) 数据传送方式。两设备间是单向传递还是双向传递？
- (3) 网络拓扑结构。网络设备是如何布局的？设备间是直接传递数据还是要通过中间设备？
- (4) 信号及编码。用什么信号传送信息？0 与 1 是如何表示的？
- (5) 介质。用什么介质传送数据？

2. 链路层

链路层负责将被传送的数据按帧结构格式化，从一个站无差错地传送到下一个站。该层从第 3 层接收数据，加上报头与报尾形成数据帧，其中包含地址及其他控制信息。

链路层的主要职责如下。

- (1) 节点至节点的数据发送。
- (2) 地址功能。报头与报尾中含有当前站与下一站的物理地址，保证使数据从发送站经过中间站到达目的站。
- (3) 存取控制。当两个以上设备连在同一条线路上时，链路层协议负责确定某时间段内哪一个设备获得线路控制权。
- (4) 流量控制。调节数据通信流量。
- (5) 差错控制。具有检错与纠错功能，当发现传输错误时，一般是要求重新发送完整信息。
- (6) 同步。报头中的同步信息向接收端表明数据已经到来，同时还可以使接收端调整接收时钟；报尾包含差错控制位及指示数据帧结束位。

数据链路层协议的例子有高级数据链路控制协议 (HDLC) 及逻辑链路控制协议 (LLC) 等。

3. 网络层

网络层负责将数据通过多种网络从源地址发送到目的地址，并负责多路径下的路径选择和拥挤控制。

网络层提供两种相关服务，即线路交换和路由。线路交换是指将物理线路暂时接通，就像用电话通话一样。通过交换使发送站与接收站直接接通进行数据传递。路由是指在多个路径方案中选择最佳路径，如速度、造价、可靠性等。每个数据块可能选择不同的路径，在到达目的站后再重新组装成完整的数据。

网络层添加的报头中包含数据包源地址与目的地址的信息。这些地址与数据链路层中的地址是不同的，前者是当前站与下一个要经过站的物理地址，传输过程中是不断改变的，网络层中的地址是逻辑地址，在传输过程中是不变的。

4. 传输层

传输层负责源端到目的端完整数据的传送，在这一点上与网络层是有区别的，网络层只负责数据包的传送，它并不关心数据包之间的关系。

计算机通常是多任务的，同时有若干个程序在运行。因此，源地址到目的地址的数据发送不仅是从一个计算机发送到另一个计算机上，具体地说是从一个计算机的应用程序发送到另一个计算机的应用程序上。传输层的数据头中包含了服务点（端口地址或套接字地址）的信息。也就是说，网络层负责把数据包传送到正确的计算机，而传输层则是把完整的数据传送到计算机的应用程序上。

当传输层从会话层接到数据后，将其分解为适合传输的数据段，在数据头中标明数据段的顺序，以便目的站的数据恢复。

为了提高数据的安全性，传输层可以建立源站与目的站之间的“连接”。所谓“连接”是一个连接源站与目的站的逻辑通路，一个信息中的所有数据段都从这一通路通过，此时传输层还要考虑更多的是顺序控制、流量控制及差错控制等。

5. 会话层

会话层为网络的会话控制器，负责通信设备间交互作用的建立、维护与同步，同时还负责每一会话的正常关闭，即不会造成会话的突然中断。例如，一用户需要发送长度为 200 页的文件，但当发送到 52 页时突然中断。这时有两种处理方法：一种方法是取消本次传送，重新建立会话并从第 1 页开始新一次发送；另一种方法是将大的会话分割成若干子会话，如每 10 页为一个子会话，当重新建立会话后，则从第 51 页开始新一次发送。会话层中采用后一种方法。

会话层通过在会话中插入校验点的方法将其划分为子会话，这样可保证当出现故障时不用再从头开始。根据不同传输任务的要求，校验点可能非常重要，也可能根本不用。

会话层完成的其他工作还有用户的身份检查（如口令与登录名等）和控制数据的交换方向（双向或单向）等。

会话层的数据头包含的控制信息有数据的类型及同步点信息等。

6. 表示层

表示层使数据格式不同的设备之间可以进行通信，如设备分别采用不同的编码，表示层具有代码翻译功能，使设备间能够互相理解。

表示层还具有数据的加密、解密、压缩、解压等功能。

表示层的数据头包含传输类型、参数及长度等信息。

7. 应用层

应用层是面向用户的，为用户程序（或进程）提供访问 OSI 环境的服务，如通信服务、

虚拟终端服务、网络文件传送及网络设备管理等。该层还具有相应的管理功能，支持分布应用的通用机制，解决了数据传输的完整性问题或收/发设备的速度匹配问题。

根据上述介绍，可以得出信息在 OSI 中各层传递过程的特点为：物理层面向“位”，链路层面向“帧”，网络层面向“信息包”，而在传输层则是以“报文”为单位，一个报文可分为几个信息包向下传递。一般分类是低四层负责用户数据的透明传输，高三层负责对数据进行分析、转换和利用。

分层模型的主要优点在于便于结构化设计的实现、修改和扩充。OSI 不是网络协议标准，它仅仅是为协议标准提供了一种主体结构，供各种协议进行标准化选择。其中选用最多的是物理层和数据链路层，其他各层按需要选用，并把第 3 层及其以上各层称为高层。例如，目前应用的局域网只选用了物理层和数据链路层，其余统称为高层。

实际使用的网络通信协议有很多，在计算机控制系统中有着广泛的应用，包括目前应用最广的局域网（LAN）的网络协议、各种 DCS 的通信协议、各种 FCS 的通信协议、工业以太网以及串行通信总线的通信协议等，下面分别介绍其中使用较多的 IEEE 802 标准、TCP/IP 以及串行通信总线。

6.4 IEEE 802 标准

电气与电子工程师协会（IEEE, Institute of Electrical and Electronics Engineers）于 1980 年 2 月成立了 IEEE 802 课题组，于 1981 年底提出了 IEEE 802 局域网标准，它相当于 OSI 模型的第一层与第二层，而在高层与 OSI 模型保持兼容。

IEEE 802 标准所提供的功能是局域网所应完成的最小的基本通信功能。在使用通信网提供的服务中，按照 OSI 模型的规定，其第一层至第三层的简要描述如下。

第一层，物理层：负责物理传输介质上的数据流的传输，涉及到信号电压的大小、极性和比特的传输时间等参数，以及为建立、维持和终止物理链路所需的机械、电气和规程方面的特性。

第二层，数据链路层：提供通过物理链路的可靠的数据传送，为高层正确无误地传输数据包，它的主要内容是规定介质存取控制技术、错误校验方式和逻辑链路管理方式。该层要把信包变成线路传输单位，对于同步方式而言就是帧。

第三层，网络层：该层的主要功能是信包的路径选择以及链路的协调管理，包括如何选择路径以防止死锁和阻塞。

局域网具有以下两个特点：用带地址的帧来传送数据；不存在中间交换，所以也不要求有路由选择（在环形局域网和基带总线局域网中使用转发器，但它们都不包含交换和路由选择功能）。因此，相对于 OSI 模型，第一层，即物理连接是需要的；第二层也是必要的，因为通过局域网传输的数据必须组成帧，并进行一定的控制。对于第三层，从概念上讲也是需要的，因为从局域网上所连的设备角度来看，它本身是被连接到了能与多个设备通信的网络中的某个访问点，并通过该访问点将所发送的电文传送给网络上许多端点中的一个。但对局域网而言，由于任意两点间有直接的链路可用，因此并不需要路由选择功能；其他由网络层提供的功能，如寻址、排序、流量控制及差错控制等，对存在单个直接链路的两个节点而言，也可以在第二层上实现，而局域网的任意节点之间的情况也是如此。由上所述，局域网所应

提供的基本通信功能可在 OSI 模型的第一层和第二层上实现。

IEEE 802 协议规定了逻辑链路控制 (LLC)、介质存取控制 (MAC) 和物理层 (PS) 3 个层次的内容, 它们相当于 OSI 模型的第二层和第一层, 即数据链路层和物理层。该协议中关于体系结构的定义、寻址方法和其他综述在 IEEE 802.1 (综述和体系结构) 标准文本中提供。

OSI 协议中的数据链路层是由高级同步数据链路协议 (HDLC) 完成的, 而在 IEEE 802 标准中, 要求在相对于 OSI 数据链路层的层次上, 完成以下功能:

- 1) 提供一个或多个服务访问点 (SAP)。所谓服务访问点, 是指两个相邻点之间的逻辑接口, 这里是指与高层的接口。
- 2) 发送时将信包组装成带有地址和差错检测字段的帧。
- 3) 接收时拆卸帧, 完成地址识别和差错检测。
- 4) 管理链路上的通信。

在 IEEE 802 标准中, 上述第 (1) 项功能被包含在 LLC 子层中。后三项功能被看成是独立的一层, 叫介质存取控制 (MAC) 子层。IEEE 802 标准之所以将数据链路层划分为两个子层, 是由以下因素决定的:

1) 在 OSI 模型的数据链路层中缺少对包含多个源和多个目的地的链路进行访问管理所需的逻辑功能。在 IEEE 802 的 LLC 子层中对此作了针对于局域网的规定。IEEE 802 的 LLC 子层的规定体现在 IEEE 802.2 标准中。

2) 对于同一 LLC 子层, 可以提供几种 MAC 方式。IEEE 802 委员会认为, 单一的一种结构不可能满足所有的应用场合, 因此选定了带有冲突检测的载波侦听多路存取 (CSMA/CD)、令牌总线 (token bus)、令牌环 (token ring) 等 3 种网络结构, 并分别体现在 3 种 MAC 方式中。因此, 有必要将 LLC 和 MAC 分开定义。该协议链路层结构如图 6-14 所示。

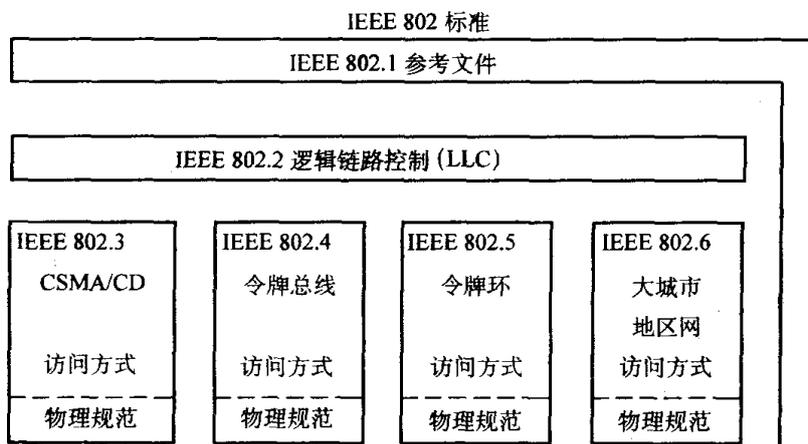


图 6-14 IEEE 802 标准的方案

IEEE 802 的 3 种 MAC 定义及对应的物理层定义分别体现在 IEEE 802.3 (CSMA/CD 介质存取控制方法和物理层规范)、IEEE 802.4 (令牌总线介质存取控制方法和物理层规范)、IEEE 802.5 (令牌环介质存取控制方法和物理层规范) 等 3 个标准中。

此外, IEEE 802 局域网标准中还包含 IEEE 802.6 城域网标准、IEEE 802.7 宽带技术咨询组、IEEE 802.8 光纤技术咨询组和 IEEE 802.9 数据和语音综合网络标准等。

6.5 TCP/IP

TCP/IP 是一组不同层次上的多个协议组合成的一个协议族，是发展至今最成功的通信协议，它被用于当今所构筑的最大的开放式网络系统 Internet 之上就是其成功的明证。TCP/IP 建立的质量标准是开放式、抗毁性和可靠性。这些特性包括可靠传输数据、自动检测和避免网络发生错误的能力。此外，重要的是 TCP/IP 是一个开放式通信协议，开放性意味着在任何组合间，不管这些设备的物理特征有多大差异，都可以进行通信。

与 OSI 参考模型不同，TCP/IP 模型更侧重于互联设备间的数据传送，而不是严格的功能层次划分。它通过解释功能层次分布的重要性来做到这一点，但它仍为设计者具体实现协议留下很大的余地。因此，OSI 参考模型在解释互联网络通信机制上比较适合，而 TCP/IP 成为了互联网络协议的市场标准。

6.5.1 TCP/IP 的网络层次结构

TCP/IP 分为 4 个层次，每一层具有不同的功能。

(1) 链路层。有时也称为数据链路层或网络接口层，通常包括操作系统中的设备驱动程序和计算机中对应的网络接口卡。它们一起处理与电缆（或其他任何传输媒介）的物理接口细节。

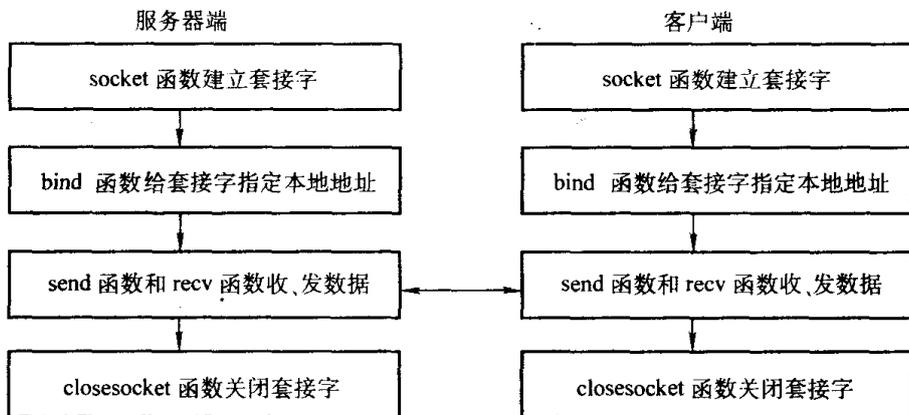
(2) 网络层。有时也称为互联网层，用于处理分组在网络中的活动，如分组的选路。在 TCP/IP 族中，网络层协议包括 IP（网际协议）、ICMP（Internet 互联网控制报文协议）及 IGMP（Internet 组管理协议）等。IP 是网络层上的主要协议，同时被 TCP 和 UDP 使用。TCP 和 UDP 的每组数据都通过端系统每个中间路由器中的 IP 层在互联网中进行传输。应用程序直接访问 IP 的情况很少。尽管 ICMP 主要被 IP 使用，但应用程序也有可能访问它。两个流行的诊断工具 Ping 和 Traceroute，它们都使用了 ICMP。IGMP 是 Internet 组管理协议，它用来把一个 UDP 数据报传播到多个主机。

(3) 传输层。主要为两台主机上的应用程序提供端到端的通信。在 TCP/IP 族中，有两个互不相同的传输协议：TCP（传输控制协议）和 UDP（用户数据报协议），TCP 为两台主机提供高可靠性的数据通信。它所做的工作包括把应用程序交给它的数据，分成合适的小块交给下面的网络层，确认接收到的分组，设置发送最后确认分组的超时时钟等。由于传输层提供了高可靠性的端到端的通信，因此应用层可以忽略所有这些细节。而另一方面，UDP 则为应用层提供了一种非常简单的服务。它只是把称为数据报的分组从一台主机发送到另一台主机，但并不能保证该数据报一定到达另一端，需由应用层来提供各种必要的保证。这两种传输层协议分别在不同的应用程序中有不同的用途。

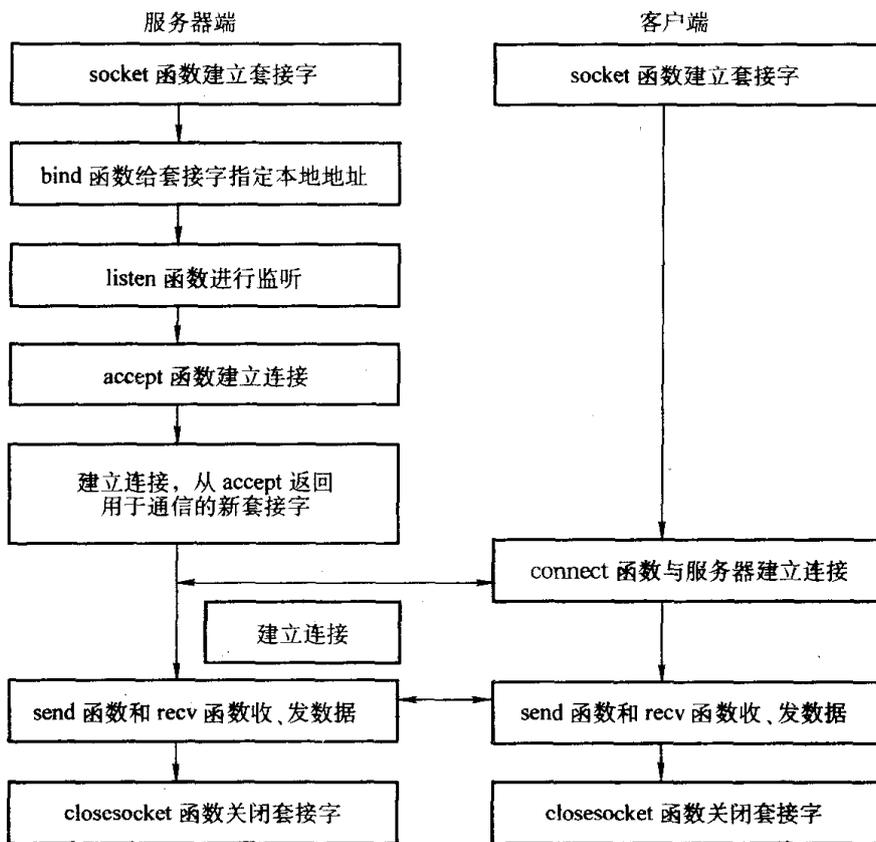
图 6-15a 是面向 UDP 的套接字通信流程，图 6-15b 是面向 TCP 的套接字通信流程。

(4) 应用层。负责处理特定的应用程序细节。在各种具体软件实现中提供 Telnet（远程登录）、FTP（文件传输协议）、SMTP（简单文件传送协议）和 SNMP（简单网络管理协议）等一些功能。

图 6-16 是一个在以以太网为基础的局域网中进行 FTP 通信的例子，图中列出了该过程所涉及到的所有协议。



a) 面向 UDP



b) 面向 TCP

图 6-15 基于 TCP/UDP 的套接字通信流程

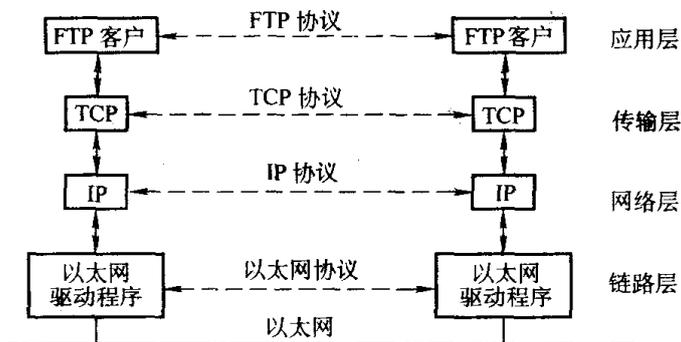


图 6-16 TCP/IP 通信的实例

6.5.2 IP 地址

一个设备的地址是它在网络中的惟一识别标志。网络地址一般分为网络地址和节点地址。根据 IPv4 (IP 第 4 版) 标准的规定, IP 使用 32bit 二进制地址对网络、网络设备和连接到网络的机器进行地址的定位。IP 地址由 Internet 网络信息中心 (InterNIC) 管理。

共有 5 类 IP 地址, 分别用字母 A、B、C、D、E 标识。每个地址由网络地址和主机地址两部分组成。5 类地址表示所支持的网络数和主机数间不同的折衷方案。通常把这些地址用带点的十进制数表示成四个字节, 如 222.222.222.45 等。

A 类 IP 地址的第一字节首位 (二进制) 是 0, 后七位表明网络号。后面 3B 共 24bit 表明主机地址。因此: A 类地址是从 1.0.0.0 到 127.255.255.255。实际的每个 A 类地址支持 16 774 214 个用户地址。

B 类 IP 地址的第一字节首二位 (二进制) 是 10, 后 14 位表明网络号。后面 2B 共 16bit 表明主机地址。即 B 类地址是从 128.1.0.0 到 191.255.255.255。实际的每个 B 类地址支持 65534 个惟一主机地址。

C 类 IP 地址的第一字节首三位 (二进制) 是 110, 后 21 位表明网络号。后面 1B 共 8bit 表明主机地址。即 C 类地址是从 192.0.0.0 到 233.255.255.255。实际的每个 C 类地址支持 254 个惟一主机地址。

D 类 IP 地址的第一字节首四位 (二进制) 是 1100, 它主要用于多信道广播, 每路广播地址有惟一的网络地址。因此, D 类地址是从 224.0.0.0 到 239.255.255.254。

E 类 IP 地址由 InterNIC 保留, 用于自己的研究, 因此, 一般用户是不能使用 E 类 IP 地址的。

在 A、B、C、D 类 IP 地址中也都有部分地址是保留的。例如, 用于多信道寻址或系统保留, 用户实际上不能使用。例如, A 类的 127 地址被用于回路测试及主机与自己的通信, 因此, 一旦安装完 TCP/IP, A 类主机就具有 127 地址。IP 地址要求每个在网络上的计算机等设备有各自惟一的地址。由于 Internet 的飞速发展, 尤其是 Internet 的商业化, 使互联网用户以前所未有的速度增长, 虽然理论上 IPv4 可提供约 40 亿个地址, 但 IP 地址的不足和对新的网络层服务的要求推动了新版本的发展, 新版本 IPv6 对 IP 地址作了极大扩展。它采用 128 位二进制地址来表明网络, 理论上可提供 340, 282, 366, 920, 938, 463, 463, 374, 607, 431, 768, 211 456 个地址。此外, 对网络的实时通信、网络层安全等服务也在新版本中得到了解决。

为了正确寻址网络上的设备, 连接在同一网络上的每个设备必须使用同样的网络地址。同样, 具有同一网络地址的每个设备必须有不同的节点 (或主机) 号来相互区分。在网络中有两类不同的地址: 网络层地址 (即 IP 地址) 及介质存取控制 (MAC) 层地址。IP 地址由网络的拓扑结构来分配。MAC 层地址又称为物理地址或链路地址, 通常它被网卡或网络设备的制造厂商设置在硬件中。

在 Internet 的通信中, 首先应识别 IP 地址是否与主机在同一子网内, 如果是, 则不用寻址, 直接可在本子网内传输信息; 如果不是, 则信息先传给所连接的网关, 通过网关进入路由器, 由路由器发送, 接收端在接收到数据后, 检查目的地址, 如果是该站的 IP 地址则接收, 如果不是则寻找有关路径再传送给其他网关。

由于在同一网络上所有主机的网络号相同，而本网的网络号已知，为识别目的站 IP 地址是否与本网的网络号一致，常采用“子网掩码”技术。子网掩码用于把数据传送到实体内不同的子网，而不必把数据传送到整个网络。子网掩码也是 32 位数，它用于快速识别目的站 IP 地址中的网络号是否与本网的网络号一致，以便判别是否属于同一子网。例如，主机 IP 地址是 222.222.221.45，其子网掩码是 255.255.255.0，则主机的网络号是 222.222.221，主机在该网络的节点号是 45。

6.6 串行通信总线

计算机控制网络中的计算机之间、计算机与远程终端、计算机与外部设备以及计算机与测量仪表之间的通信，多数情形下采用串行通信方式，而且需借助于标准的物理层接口——串行通信总线。到目前为止，串行通信总线有很多种，如 RS-232C、RS-422、RS-485、SPI 总线、I²C 总线、SMBus 总线以及现场总线等。

下面主要介绍两种适用于中、小型工业控制系统的常用串行通信总线标准，即 RS-232C 和 RS-422/485。

6.6.1 RS-232C 通信总线

RS-232C 总线是由美国电子工业协会 EIA 于 1969 年修定的一种通信接口标准，专门用于数据终端设备和数据通信设备之间的串行通信。数据终端设备 (DTE, Data Terminal Equipment) 是数据的源点或归宿，通常是指输入、输出设备和传输控制器或者计算机等数据处理装置及其通信控制器。数据通信设备 (DCE, Data Communication Equipment) 的任务是实现由源点到目的点的传输，通常是指自动呼叫应答设备、调制解调器以及其他一些中间装置的集合。目前 RS-232C 接口已成为计算机的标准配置，如串行口 COM1、COM2 均为 RS-232C 总线接口标准。

1. 信号定义

标准的 RS-232C 接口定义了 25 个信号针，采用 25 针接插件 DB-25，并规定 DTE 的接插件为凸形，DCE 的接插件为凹形，如图 6-17a 所示。对不需要 25 针的系统来说，常用 9 针的简化接插件，如图 6-17b 所示。表 6-2 给出了常用的 9 根引脚的信号功能。

表 6-2 常用的 RS-232 总线接口信号

符号	25 针引脚	9 针引脚	信号流向	功能
TxD	2	3	输出	发送数据
RxD	3	2	输入	接收数据
RTS	4	7	输出	请求发送
CTS	5	8	输入	清除发送
DSR	6	6	输入	数据装置准备好
GND	7	5		信号地
DCD	8	1	输入	数据载体检测
DTR	20	4	输出	数据终端准备好
RI	22	9	输入	振铃指示

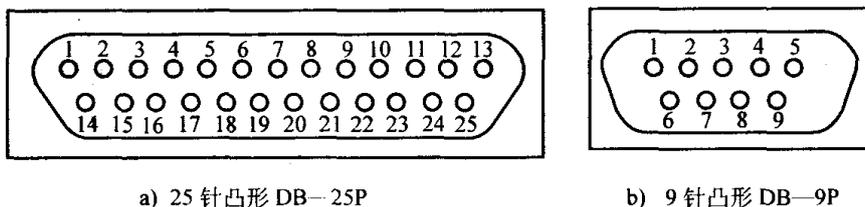


图 6-17 RS-232C 接插件

2. 电气特性

由于 RS-232C 是早期为促进公用电话网络进行数据通信而制定的标准。为了增加信号在线路上的传输距离和提高抗干扰能力，RS-232C 采用了较高的传输电平，且为双极性、公共地和负逻辑，即规定逻辑“1”状态电平为-15~-5V，逻辑“0”状态电平为+5~+15V，其中-5~+5V 用作信号状态的变迁区。

而计算机均采用 TTL 逻辑电平。TTL 电平规定低电平“0”在 0~+0.8V 之间，高电平“1”在+2.4~+5V 之间，因此在 TTL 电路与 RS-232C 总线之间要进行电平的转换及正、反逻辑的转换，否则将使 TTL 电路烧毁。

3. 接口电路

这种电平与逻辑的转换是用专门的集成电路芯片来完成的，早期常用 MC1488 和 MC1489 作发送器和接收器。如图 6-18 所示，发送器 MC1488 可实现 TTL 到 RS-232C 的电平转换，所用正、负电源分别是±12V；接收器 MC1489 可实现 RS-232C 到 TTL 的电平转换，所用电源是+5V。由于需要±12V 与+5V 供电电压，因此人们现在更愿意使用一种新的单一电源供电的 MAX232 芯片。

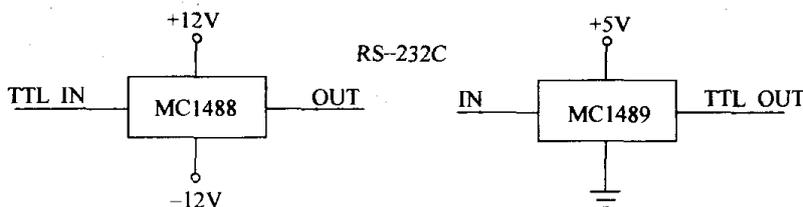


图 6-18 RS-232C 发送和接收器

MAX232 芯片的引脚结构及发送/接收过程如图 6-19 所示，它是一个含有两路发送器和接收器的 16 脚 DIP/SO 封装型工业级 RS-232C 标准接口芯片。芯片内部有一个电源电压变换

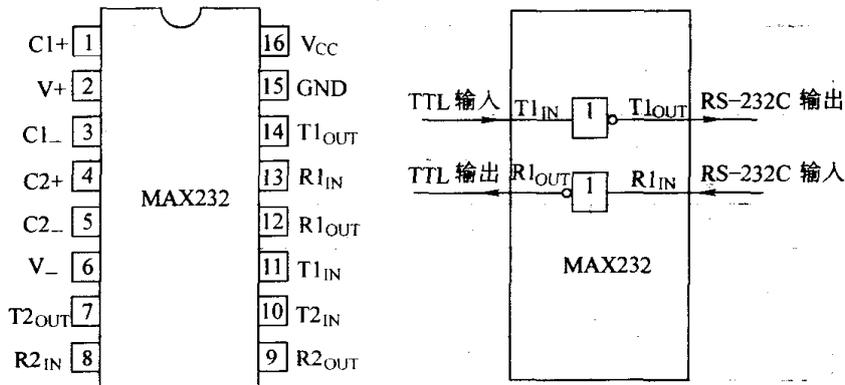


图 6-19 MAX232 芯片的引脚结构及发送/接收过程

器,可以把输入的+5V电源电压变换为RS-232C输出电平所需的 $\pm 10\text{V}$ 电压。所以,采用此芯片接口的串行通信系统只需单一的+5V电源就可以。图中给出了其中的一路发送器和接收器,T1_{IN}引脚为TTL电平输入端,转换后的RS-232C电平由T1_{OUT}送出;而R1_{IN}引脚接收RS-232C电平,转换后的TTL电平由R1_{OUT}输出。如此便完成了TTL到RS-232C(发送)以及RS-232C到TTL(接收)的电平与逻辑的转换。

由于采用单端输入和公共信号地线,容易引进干扰。为了保证数据传输的正确,RS-232C总线的传送距离一般不超过15m,传送信号速率不大于20kbit/s。

4. 电流环接口

为了提高RS-232C通信的抗干扰能力,常采用一种称为20mA电流环的技术,进行DTE与DCE之间的光电隔离。20mA电流环规定,以回路中有电流表示逻辑“1”,而无电流表示逻辑“0”。它的基本电路结构如图6-20所示,其中包括一个发光二极管和一个光敏晶体管。如果从TTL输出逻辑电平,转换成20mA电流环,通常可以将二极管通过限流电阻接+5V电源,而其阳极则与输入信息引线连接,其中限流电阻的数值要视二极管阈值电流大小而定,即为“1”时,应使电流达20mA左右,反之为零。另外要注意,要使外设相连接路上满足20mA电流要求,一般是由外设提供电源及调整手段。

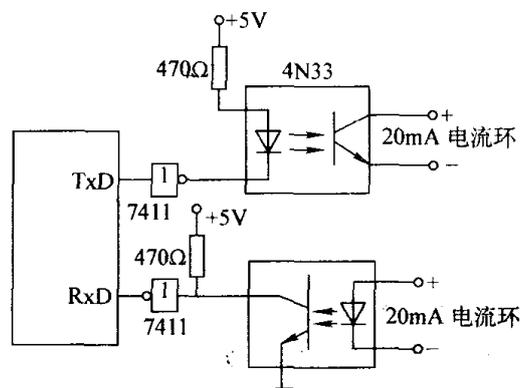


图 6-20 光隔离电流环接口电路

5. 通信连接形式

需要指出的是,一般PC如Intel的x86系列的各种CPU内均没有串行接口,因此在进行串行通信时,都需配备适当的接口适配器,如Intel8250、Intel8251等,以完成CPU与串行接口之间发送数据的并/串转换与接收数据的串/并转换。作为自动化系统的设计者,一般并不关心已成为计算机标准配置的RS-232C接口的内部电路,更关心如何利用PC的RS-232C接口构建与外部其他计算机(包括本身带有串行接口的单片微型计算机)或数据通信设备的串行通信问题。

RS-232C总线标准中包含两个信道,即主信道和次信道,表6-2中仅给出了常用的主信道接口信号。根据具体的应用场合不同,RS-232C通信主要有以下几种连接方式。

(1) PC与PC之间使用MODEM连接

计算机之间通过MODEM或其他DCE使用一条电话线进行通信,如图6-21所示。图中,计算机DTE向远程DTE发送数据的过程如下:首先DTE向本地DCE(MODEM)发出DTR=“1”和RTS=“1”的信号,表示DTE为本地和远程DCE之间建立通道开了绿灯,同时请求发送数据;此时DCE发回信号DSR=“1”,表示DCE已做好发送数据准备,又向DTE发回CTS=“1”,表示准备向DTE发送数据。只有当DTE收到从本地DCE发回肯定的DSR和CTS信号后,DTE才能由TxD线向DCE发送数据。因此,RTS、DTR、DSR、CTS四个信号同时为“1”是TxD发送数据的条件。

当接收数据时,DTE先向本地DCE发出DTR=“1”信号,表示本地和远程DCE之间可以建立通道;一旦通道建立好,DCE向DTE发出DSR=“1”信号;这时,数据就可以通过

RxD 线传到 DTE。因此，RxD 信号产生的条件是 DTR 和 DSR 两个信号同时为“1”。至于 RxD 线上是否有信号，取决于远程 DTE 是否发送数据。

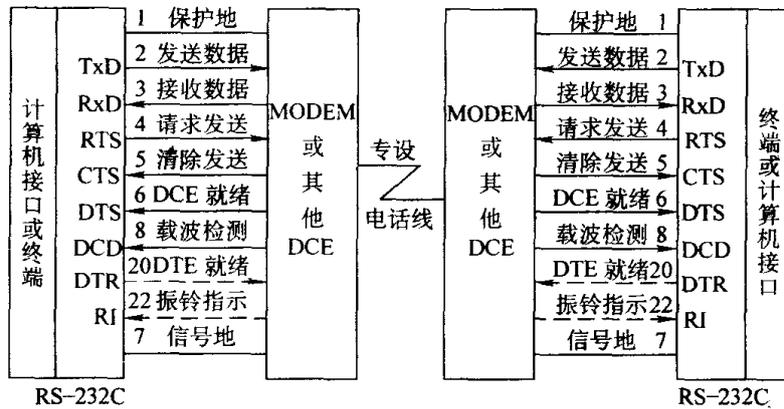


图 6-21 使用 MODEM 时 RS-232C 引脚的连线

(2) PC 与 PC 之间直接连接

当计算机和终端之间不使用 MODEM 或其他 DCE 而直接通过 RS-232C 接口连接时，一般只需要 5 根线（不包括保护地线以及本地 4、5 之间的连线），但其中多数应采用反馈与交叉相结合的连接法，如图 6-22 所示。

图中，2→3 交叉线为最基本的连线，以保证 DTE 和 DCE 之间能正常地进行全双工通信。20→6 也是交叉线，用于两端的通信联络，使两端相互检测出对方“数据已就绪”的状态。4→5 为反馈线，使传送请求总是被允许的。由于是全双工通信，这根反馈线意味着任何时候都可以双向传送数据，用不着再去发“请求发送”（RTS）信号。这种没有 MODEM 的串行通信方式，一般只用于近程通信。

(3) PC 与 PC 之间三线连接法

这是一种最简单的 RS-232C 连线，只需 2→3 交叉连接线以及信号地线，而将各自的 RTS 和 DTR 分别接到自己的 CTS 和 DSR 端，如图 6-23 所示。

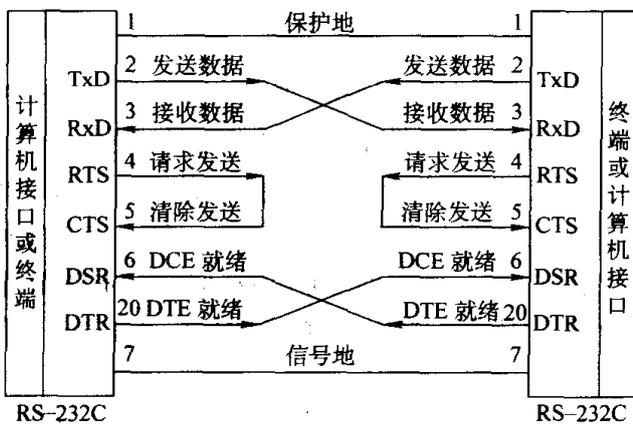


图 6-22 使用 RS-232C 的直接连接法

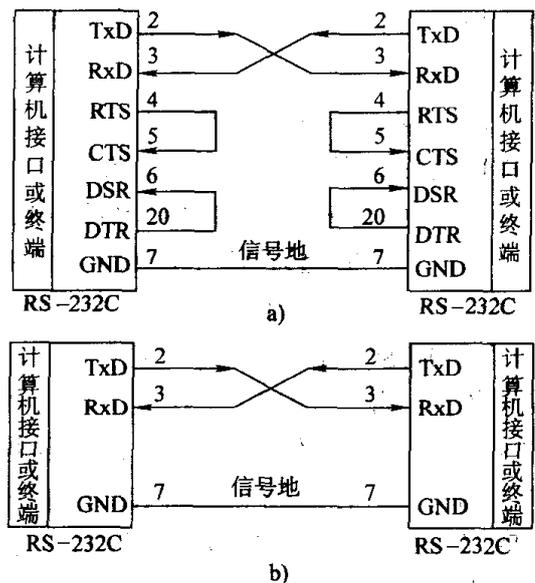


图 6-23 最简单的 RS-232C 连接方式

在图 6-23a 中, 只要一方使自己的 RTS 和 DTR 为“1”, 那么它的 CTS、DSR 也就为“1”, 从而进入了发送和接收的就绪状态, 这种接法常用于一方为主动设备, 而另一方为被动设备的通信中, 如计算机与打印机或绘图仪之间的通信。这样, 被动的一方 RTS 与 DTR 常置“1”, 因而 CTS、DSR 也常置“1”, 因此, 使其长期处于接收就绪状态, 只要主动一方令线路就绪 (DTR=“1”), 并发出发送请求 (RST=“1”), 即可立即向被动的一方传送信息。

图 6-23b 为更简单的连接方法, 如果说图 6-23a 所示的连接方法在软件设计上还需要检测“清除发送”(CTS)和“数据设备就绪”(DSR)的话, 那么图 6-23b 所示的连接方法则完全不需要检测上述信号, 随时都可发送和接收。这种连接方法无论在软件还是在硬件上, 都是最简单的一种方法。

上述几种情形都是 PC 与 PC 之间, 应用 RS-232C 总线接口标准进行串行通信的。而在中、小型计算机控制系统中, 常用的一种控制网络模式是一台 PC 与若干台单片机系统构成的分散型测控系统。

(4) PC 与单片机之间的通信网络

这种模式是把以单片机为核心的智能式测控仪表作为从机 (又称下位机), 完成对工业现场的数据采集和控制任务, 而 PC 作为主机 (又称上位机) 将上传数据和下达指令以实现集中管理和最优控制。

图 6-24 给出了 PC 与多个单片机构成的 RS-232C 通信网络示意, PC 作主机、 n 个单片机智能仪表为从机, 构成了主从方式的 RS-232C 串行总线网络。PC 串行口给出的已是标准的 RS-232C 电平, 而单片机则为 TTL 电平, 采用 MAX232 芯片就可实现电平的转换和驱动 (见图 6-19)。

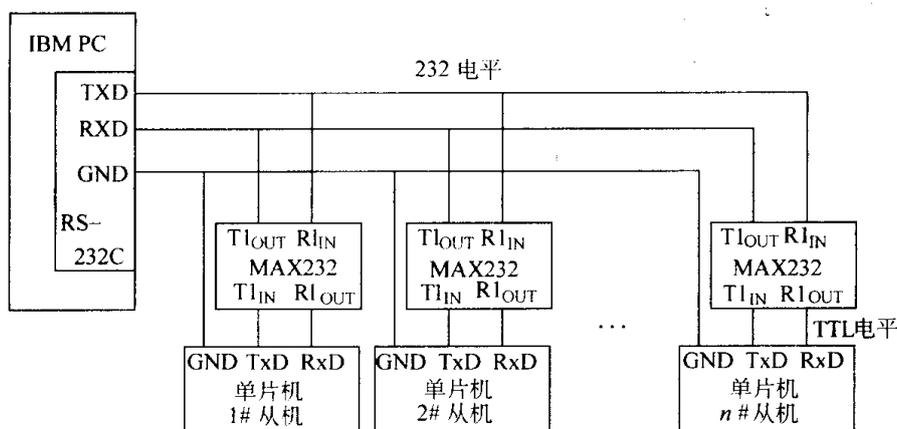


图 6-24 PC 与多个单片机构成的 RS-232C 通信网络

6. 软件编程实现

RS-232C 接口的软件编程方法很多, 主要有以下几种方式: 直接读/写硬件端口 (如直接对 Intel8250、8251 编程)、DOS 下调用 BIOS 或 DOS 中断服务、Windows 使用 API 函数或调用 Comm 控件等。下面简单介绍其中一种在 Windows 下的软件编程方法, 即通过调用微软提供的称为 MSCOMM32.OCX 的 Active X 控件使用 VC++ 进行串行异步通信的编程, 假设这里硬件按图 6-23b 所示的方式进行连接。

使用 Active X 进行编程的第一步, 首先要进行控件的注册, 将 MSCOMM32.OCX 注册到

Windows 操作系统。然后，在 VC 的集成编程环境下添加该控件，如使用 MFC，则会自动生成两个程序，如 MYComm.h 和 MYComm.cpp，这两段程序提供了 MSCOMM32.OCX 的标准接口，同时主程序框架下加入如下代码：

```
int CMainFrame::OnCreate (LPCREATESTRUCT lpCreateStruct)
{
    .....
    if (!MYComm.Create (NULL, 0, CRect (0, 0, 0, 0), this, IDC_MYCOMM))
    { return -1;}
    MYComm.SetCommPort (0);
    MYComm.SetRThreshold (1);
    MYComm.SetInputMode (1);
    MYComm.SetSThreshold (0);
    MYComm.SetPortOpen (TRUE);
    .....
}
```

其中 MYComm.Create () 用于控件的产生；初始化部分：MYComm.SetCommPort (0) 指定通信端口为 COM1，MYComm.SetRThreshold (1) 设置接收线程，MYComm.SetSThreshold (0) 设置发送线程，MYComm.SetInputMode (1) 设置中断接收方式，MYComm.SetPortOpen (TRUE) 打开串口。串行通信波特率和帧格式采用默认形式（即 9600bit/s，8 为字长，1 位终止位，无校验位），如需改变设置可使用 MYComm.SetSettings (LPCTSTR lpszNewValue) 函数实现，字符串 lpszNewValue 的格式为“波特率，奇偶校验，字长，停止位”。

发送数据 var 可以通过如下代码实现：

```
MYComm.SetOutput (var);
while (MYComm.GetOutBufferCount () > 0);
```

采用中断方式接收数据时，计算机系统通过中断自动调用如下代码实现：

```
void CMainFrame::OnCommEvent ()
{
    CString result;
    ...
    switch (MYComm.GetCommEvent ())
    {
        case 1: // vbMSCommEvSend:
            ...
            break;
        case 2: // vbMSCommEvReceive:
            result=MYComm.GetInput ();
            break;
        ...
    }
}
```

6.6.2 RS-422/485 通信总线

RS-232C 虽然使用很广，但由于推出时间比较早，所以在现代通信网络中已暴露出明显的缺点。主要表现在：传输速率不够快；传输距离不够远；未明确规定连接器；接口使用非平衡发送器和接收器；接口处各信号间容易产生串扰。所以 EIA 在 1977 年作了部分改进，制定了新标准 RS-449，其中除了保留与 RS-232C 兼容外，还在提高传输速率、增加传输距离、改进电气特性等方面做了很多努力，增加了 RS-232C 没有的环测功能，对连接器作了明确规定，解决了机械接口问题。

在 RS-449 标准下，推出的子集有 RS-423A/RS-422A，以及 RS-422A 的变型 RS-485。

1. RS-423A/RS-422A

RS-423A/RS-422A 总线标准的数据线也是负逻辑且参考电平为地，与 RS-232C 规定为 $-15\sim+15\text{V}$ 有所不同，这两个标准规定为 $-6\sim+6\text{V}$ 。

与 RS-232C 的单端驱动非差分接收方式相比，RS-423A 是一个单端驱动差分接收方式，而 RS-422A 则是平衡驱动差分接收方式。差分平衡结构能从地线的干扰中分离出有效信号。实际上，差分接收器可以区分 0.2V 以上的电位差，因此，可不受参考电平波动及共模电磁干扰的影响，具有更强的抗干扰能力，数据传输速率与传输距离也更快、更远。RS-423A 在传输速率为 1kbit/s 时，传输距离可达 1200m ，在速率为 100kbit/s 时，距离可达 90m ；而 RS-422A 可以在 1200m 距离内把传输速率提高到 100kbit/s ，或在 12m 内提高到 10Mbit/s 。图 6-25 给出了 RS-232C/RS-423A/RS-422A 的数据传输电气接口电路。

图 6-25a 为 RS-232C 所采用的单端发送、单端接收电路。该电路的特点是传送信号只用一根导线，对于多路信号线，其地线是公共的。因此，它是最简单的连接结构，但缺点是易受干扰信号的影响。而 RS-423A 采用了单端发送、双端接收的传送方式，如图 6-25b 所示，利用差分接收器的另一端接发送端的信号地，因而大大减少了地线的干扰。RS-422A 则更进一步采用了双端发送、双端接收的传送方式，如图 6-25c 所示，这种平衡驱动和差分接收方法从根本上消除了地线干扰。这种发送器相当于两个单端发送器，它们的输入是同一个信号，而一个发送器的输出正好与另一个反相。当干扰信号作为共模信号出现时，接收器则接收差分输入电压。只要接收端具有足够的抗共模电压工作范围，它就能识别这两种信号而正确接收传送信号。

RS-423A/RS-422A 的另一个优点是允许传输线上连接多个接收器。虽然在 RS-232C 系统中可以使用多个接收器循环工作，但它每一时刻只允许一个接收器工作，RS-423A/RS-422A

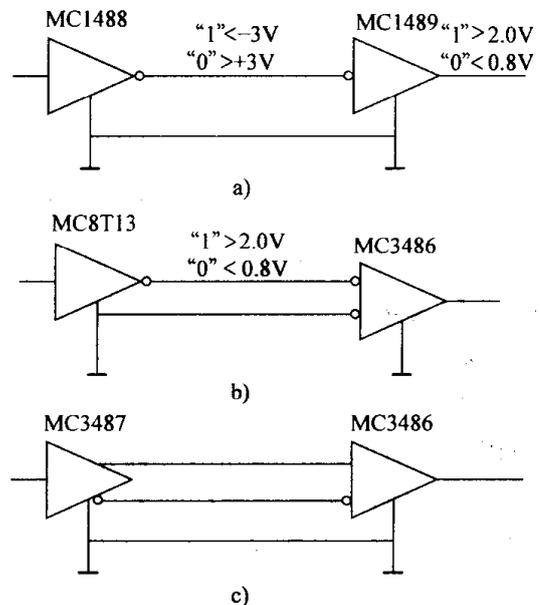


图 6-25 RS-232C/RS-423A/RS-422A 接口电路

可允许 10 个以上接收器同时工作。

2. RS-485

在许多工业过程控制中，往往要求用最少的信号线来完成通信任务。目前广泛应用的 RS-485 串行接口总线就是为适应这种需要应运而生的。它实际就是 RS-422 总线的变形，二者不同之处在于：RS-422 为全双工，而 RS-485 为半双工；RS-422 采用两对平衡差分信号线，RS-485 只需其中的一对。RS-485 更适合于多站互联（已经具备了现场总线的概念），一个发送驱动器最多可连接多于 32 个负载设备，负载设备可以是被动发送器、接收器和收发器。其电路结构是在平衡连接的电缆上挂接发送器、接收器或组合收发器，且在电缆两端各挂接一个终端电阻用于消除两线间的干扰。

图 6-26 给出了 RS-485 与 RS-422 总线的两种数据传送方式。图 6-26a 为 RS-485 半双工连接方式，任一时刻只能有一个站发送数据，一个站接收数据。因此，其发送电路必须由使能站加以控制。图 6-26b 由于是全双工连接方式，故两站都可以同时发送和接收。

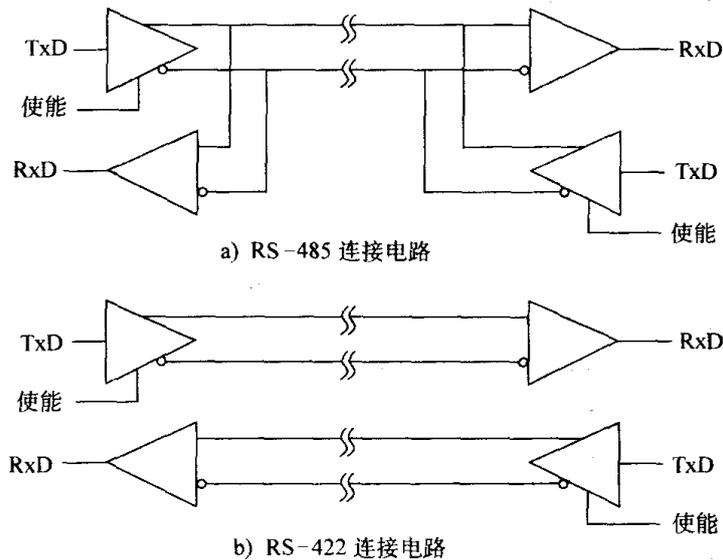


图 6-26 RS-485 与 RS-422 总线的的数据传送方式

与 RS-232C 标准总线一样，RS-422 和 RS-485 两种总线也需要专用的接口芯片完成电平转换。MAX481E/ MAX488E 分别是只用+5V 电源的 RS-485/RS-422 的 8 引脚收发器，其结构及引脚如图 6-27 所示。两个芯片的共同点是都含有一个发送器 D 和一个接收器 R，其中 DI 是发送输入端，RO 是接收输出端。不同的是，图 6-27a 中只有两根信号线 A 和 B，信号线 A 为同相接收器输入和同相发送器输出，信号线 B 为反相接收器输入和反相发送器输出，由于是半双工，所以有发送与接收的使能端 DE 与 \overline{RE} 引脚。而在图 6-27b 中，有两对 4 根信号线 A、B 和 Y、Z，其中 A、B 专用作接收器输入，A 为同相、B 为反相；而 Y、Z 专用作发送器输出，Y 为同相、Z 为反相，所以构成了全双工通信。

图 6-28 所示为以 PC 作主机，n 个单片智能设备为从机、工作于主从方式的 RS-485 总线网络的结构。利用 PC 配置的 RS-232C 串行端口，外配一个 RS-232C/RS-485 转换器，可将 RS-232C 信号转换为 RS-485 信号。每个从机通过 MAX487E 芯片构建 RS-485 通信接口，就可挂接在 RS-485 总线网络上，总线端点处并接的两个 120Ω电阻用于消除两线间的干扰。

RS-485 总线网络传输距离最远可达 1200m (速率 100kbit/s)、传输速率最高可达 10Mbit/s (距离 12m)。至于在网络上最多允许挂接多少个从机, 这主要取决于 RS-232/485 转换器的驱动能力与 RS-485 接口芯片的输入阻抗与驱动能力, 如果再加上中继站, 可以增加更多的从机数量。

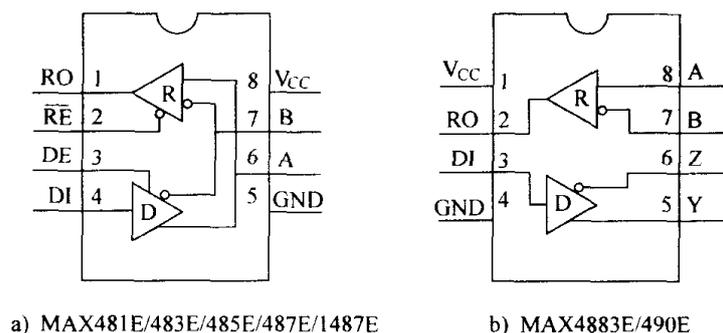


图 6-27 MAX481E/MAX488E 的结构及引脚

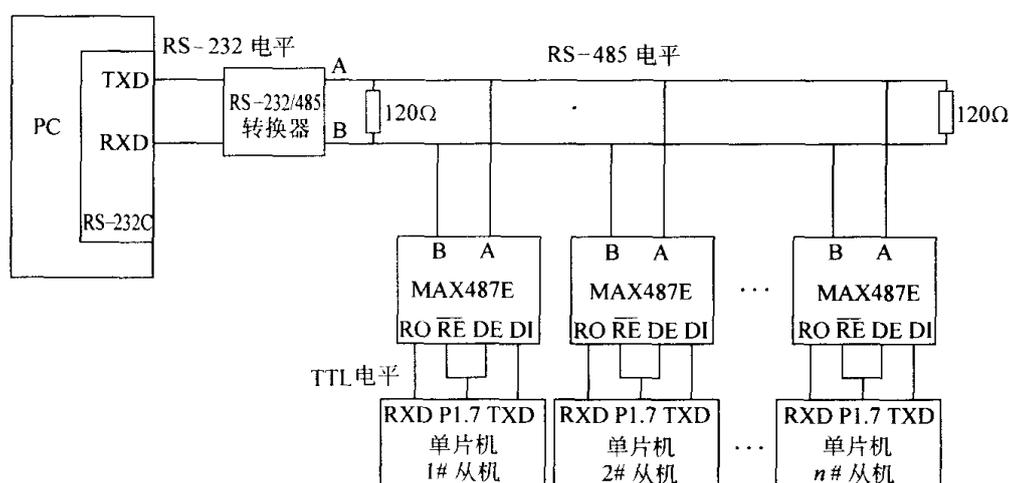


图 6-28 PC 与多个单片机构成的 RS-485 通信网络

RS-422/485 是一种平衡传输的电气标准, 不包含软件接口协议, 但它的支持 SDLC 等同步协议, 可对帧中的站地址进行识别, 实现链路寻址, 构成 RS-422/485 网络。

目前市场上已有许多 RS-232/485 转换器出售, 可以有效地利用微型计算机标准配置的 RS-232 接口, 实现 RS-485 通信。

思 考 题

1. 数据通信的实质是什么? 比较说明并行通信和串行通信的概念及其特点。
2. 画图说明串行通信的 3 种制式。
3. 串行异步通信和同步通信如何实现收、发双方的同步, 两者有何区别?
4. CSMA/CD 和令牌传送在原理和控制策略上有什么不同?
5. 频分多路与时分多路是如何实现多路复用的? 试比较两种方式的特点。

6. 设要发送的基本信息位为 11010111, 附加 16 位 CRC 校验信息位, 生成多项式为 $G(x) = x^{16} + x^{15} + x^2 + 1$, 求发送代码多项式 $T(x)$ 。
7. OSI 参考模型的各层分别是什么? 完成什么功能?
8. 对比说明 RS-232C 总线标准与 TTL 逻辑电平的电气特性, 它们之间如何进行接口?
9. RS-232C 总线在实际应用中有几种接线方式? 都应用在何种场合?
10. 结合图 6-19 与图 6-24, 分析说明 PC 与多个单片机构成的 RS-232C 通信网络。
11. 结合图 6-28, 分析说明 PC 与多个单片机构成的 RS-485 通信网络。

第 7 章 分散控制系统与现场总线控制系统

随着计算机技术、通信技术、控制技术、大规模集成电路技术、图形显示技术、多媒体技术、人工智能技术以及其他高新技术的发展，分散型控制系统和现场总线控制系统，已成为工业生产过程控制、管理和决策的核心，广泛应用于电力、石油、化工、冶金、建材、制药等各种工业生产过程。在面对日趋激烈的市场竞争条件下，分散控制系统和现场总线控制系统在保持生产过程的安全稳定，提高生产企业工艺系统的经济效益，实现全厂范围内的优化运行，并且在管理层的决策支持方面发挥着越来越重要的作用。

7.1 分散控制系统概述

分散控制系统（DCS, Distributed Control System）是继直接作用式气动仪表、气动单元组合仪表、电动单元组合仪表和组件组装式仪表之后的新一代控制系统。分散控制系统是以大型工业生产过程及其相互关系日益复杂的控制对象为前提，从生产过程综合自动化的角度出发，按照系统工程中分解与协调的原则研制开发出来的，是一种以微处理机为核心，结合了控制技术、通信技术和 CRT 显示技术的新型控制系统。

7.1.1 分散控制系统的结构

分散控制系统是纵向分层、横向分散的大型综合控制系统。它以多层计算机网络为依托，将分布在全厂范围内的各种控制设备和数据处理设备连接在一起，实现各部分的信息共享和协调工作，共同完成各种控制、管理及决策功能。

图 7-1 所示为一个分散控制系统的典型结构，系统中的所有设备分别处于四个不同的层

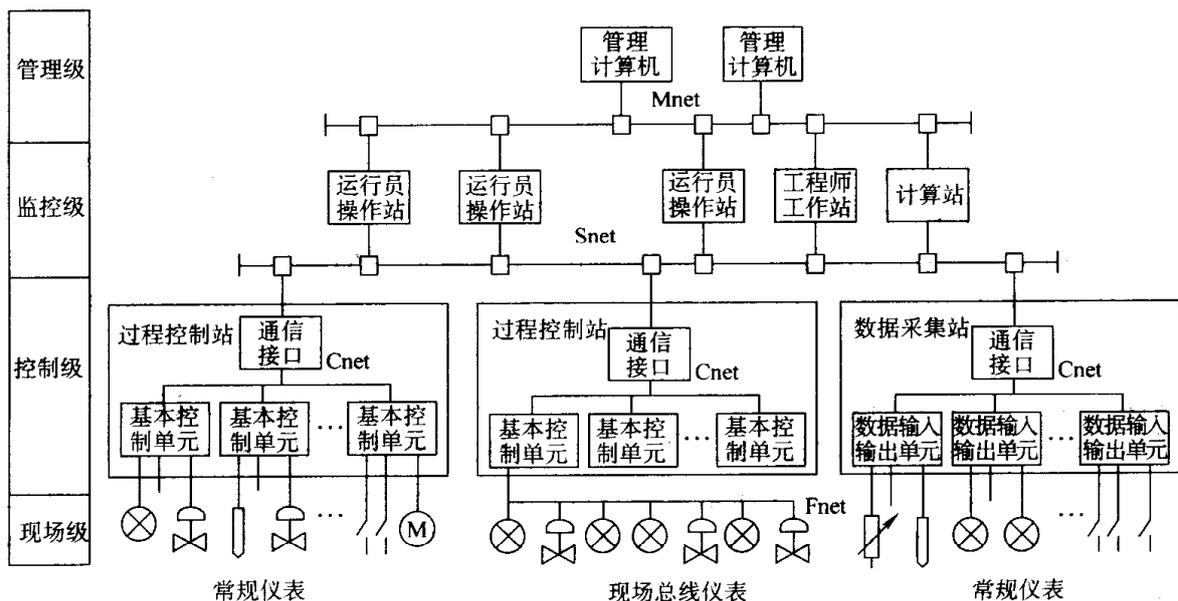


图 7-1 分散控制系统的典型结构

次, 自下而上分别是现场级、控制级、监控级和管理级。对应着这四层结构, 分别由四层计算机网络, 即现场网络 (Fnet, Field Network)、控制网络 (Cnet, Control Network)、监控网络 (Snet, Supervision Network) 和管理网络 (Mnet, Management Network) 把相应的设备连接在一起。

1. 现场级

现场级设备一般位于被控生产过程的附近。典型的现场级设备是各类传感器、变送器和执行器, 它们将生产过程中的各种物理量转换为电信号。例如, 4~20mA 的电信号 (一般变送器) 或符合现场总线协议的数字信号 (现场总线变送器), 送往控制站或数据采集站, 或者将控制站输出的控制量 (4~20mA 的电信号或现场总线数字信号) 转换成机械位移, 带动调节机构, 实现对生产过程的控制。

目前现场级的信息传递有三种方式, 一种是传统的 4~20mA (或者其他类型的模拟量信号) 模拟量传输方式; 另一种是现场总线的全数字量传输方式; 还有一种是在 4~20mA 模拟量信号上, 叠加上调制后的数字量信号的混合传输方式。现场信息以现场总线为基础的全数字传输是今后的发展方向。这方面的内容将在后面的现场总线控制系统中详细介绍。

按照传统观点, 现场设备不属于分散控制系统的范畴, 但随着现场总线技术的飞速发展, 网络技术已经延伸到现场, 微处理机已经进入变送器和执行器, 现场信息已经成为整个系统信息中不可缺少的一部分。因此, 通常将其并入分散控制系统体系结构中。

2. 控制级

控制级主要由过程控制站和数据采集站构成。一般在电厂中, 把过程控制站和数据采集站集中安装在位于主控室后的电子设备室中。过程控制站接收由现场设备, 如传感器、变送器传来的信号, 按照一定的控制策略计算出所需的控制量, 并送回到现场的执行器中去。过程控制站可以同时完成连续控制、顺序控制或逻辑控制功能, 也可能仅完成其中的一种控制功能。

数据采集站与过程控制站类似, 也接收由现场设备送来的信号, 并对其进行一些必要的转换和处理之后再送到分散型控制系统中的其他部分, 主要是监控级设备中去。数据采集站接收大量的过程信息, 并通过监控级设备传递给运行人员。数据采集站不直接完成控制功能, 这是它与过程控制站的主要区别。

3. 监控级

监控级的主要设备有运行员操作站、工程师工作站和计算站。其中运行员操作站安装在中央控制室, 工程师工作站和计算站一般安装在电子设备室。

运行员操作站是运行员与分散型控制系统相互交换信息的人-机接口设备。运行人员通过运行员操作站来监视和控制整个生产过程。运行人员可以在运行员操作站上观察生产过程的运行情况, 读出每一个过程变量的数值和状态, 判断每个控制回路是否工作正常, 并且可以随时进行手动/自动控制方式的切换, 修改给定值, 调整控制量, 操作现场设备, 以实现生产过程的干预。另外, 还可以打印各种报表, 拷贝屏幕上的画面和曲线等。

为了实现以上功能, 运行员操作站是由一台具有较强图形处理功能的微型机, 以及相应的外部设备组成, 一般配有 CRT 显示器、大屏幕显示装置 (选件)、打印机、复印机、键盘、鼠标或球标。

工程师工作站是为了使控制工程师对分散控制系统进行配置、组态、调试、维护所设置

的工作站。工程师工作站的另一个作用是对设计文件进行归类和管理,以形成各种设计文件,如各种图样、表格等。

工程师工作站一般由 PC 配置一定数量的外部设备所组成,如打印机、绘图机等。

计算站的主要任务是实现对生产过程的监督控制,如机组运行优化和性能计算、先进控制策略的实现等。由于计算站的主要功能是完成复杂的数据处理和运算功能,因此,对它主要是运算能力和运算速度的要求。一般,计算站由超级微型机或小型机构成。

机组运行优化也可以由一套独立的控制计算机和优化软件构成,只是在机组控制网络上设一接口,利用优化软件的计算结果去改变控制系统的给定值或偏置。

4. 管理级

管理级包含的内容比较广泛,一般来说,它可能是一个发电厂的厂级管理计算机,也可能是若干个机组的管理计算机。它所面向的使用者是厂长、经理、总工程师、值长等行政管理或运行管理人员。厂级管理系统的主要任务是监测企业各部分的运行情况,利用历史数据和实时数据预测可能发生的各种情况,从企业全局利益出发辅助企业管理人员进行决策,帮助企业实现其规划目标。

对管理计算机的要求是,能够对控制系统做出快速反应的实时操作系统。大量数据的高速处理与存储,能够连续运行可冗余的高可靠性系统,能够长期保存生产数据,并且具有优良的、高性能的、方便的人-机接口,还具有丰富的数据库管理软件、过程数据收集软件、人-机接口软件以及生产管理系统生成等工具软件,实现整个工厂的网络化和计算机的集成化。

管理级是属厂级的,也可分成实时监控和日常管理两部分。实时监控是全厂各机组和公用辅助工艺系统的运行管理层,承担全厂性能监视、运行优化、全厂负荷分配和日常运行管理等任务,主要为值长服务;日常管理承担全厂的管理决策、计划管理、行政管理等任务,主要是为厂长和各管理部门服务。

5. 通信网络系统

DCS 各级之间的信息传输主要依靠通信网络系统来支持。通信网络分成低速、中速、高速,其中低速网络主要用于分散系统的现场级与控制级的通信(即 Fnet),中、高速网络主要用于控制级与监控级、监控级与管理级之间及其内部的通信。

用于 DCS 的计算机网络在很多方面的要求不同于通用的计算机网络。它是一个实时网络,也就是说网络需要根据现场通信的实时性要求,在确定的时限内完成信息的传送。目前,监控级与管理级之间的通信由于 DCS 开放性的要求越来越多地采用了通用的以太网技术。

根据网络的拓扑结构,DCS 的计算机网络大致可分为星型、总线型和环型结构三种,并多采用冗余形式。DCS 中最常用的网络结构是总线型网和环型网,在这两种结构的网络中,各个节点可以说是平等的,任意两个节点之间的通信可以直接通过网络进行,而不需要其他节点的介入。在比较大的分散控制系统中,为了提高系统性能,也可以把集中网络结构合理地运用于一个系统中,以充分利用各网络结构的优点。

7.1.2 分散控制系统的特点

由于分散控制系统采用了先进的计算机控制技术和分级分散式的体系结构,所以与常规控制系统和集中式计算机控制系统相比,它具有很多优点,下面从几个侧面分别讨论一下分散控制系统的特点。

1. 适应性和扩展性

分散控制系统在结构上采用了常规控制系统的模块化设计方法,无论是硬件还是软件都可以根据实际应用的需要去灵活地加以组合。对于小规模的生产过程,可以只用一、两个过程控制站或数据采集站,配以简单的人-机接口装置,即可以实现生产过程的直接数字控制。对于大规模的生产过程可以采用几十个、甚至上百个过程控制站或数据采集站以及各种实现优化控制任务的高层计算站、运行员操作站、工程师工作站等人-机接口设备,组成一个具有管理和控制功能的大型分级计算机控制系统,这一点,集中式计算机控制系统是无法做到的。一个按照小规模生产过程设计的集中式计算机控制系统,由于主机存储容量、运算速度和带外部设备能力等诸多因素的限制,很难把它应用于大规模生产过程中。同样,一个按照大规模生产过程设计的集中式计算机控制系统,如果将其用于小规模的生产过程,则会造成巨大的浪费。

模块化设计方法带来的另一个优点是系统的扩展性。分散控制系统可以随着生产过程的不断发展,逐渐扩充系统的硬件和软件,以期达到更大的控制范围和更高的控制水平。分散控制系统的可扩展性具有两个明显的特征:一个是它的递进性,即扩充新的控制范围或控制功能时,并不需要摒弃已有的硬件和软件;另一个是它的整体性,也就是说,分散控制系统在扩展时,并不是让新扩充的部分形成一个与原有部分毫无联系的孤岛,而是通过通信网络把它们联系起来,形成一个有机的整体。这一点对于现代化的大型工业生产过程来说尤为重要。

2. 控制能力

常规控制系统的控制功能是用硬件实现的,因而要改变系统的控制功能,就要改变硬件本身,或者改变硬件之间的连接关系。在分散控制系统中,控制功能主要是由软件实现的,因此它具有高度的灵活性和完善的控制能力。它不仅能够实现常规控制系统的各种控制功能,而且还能完成各种复杂的优化控制算法和各种逻辑推理及逻辑判断。它不但保留了数字控制系统的全部优点,而且还克服了集中式计算机控制系统由于功能过分集中所造成的可靠性太低的缺点。因此,它的控制能力是常规控制系统所不可比拟的。分散控制系统集中了连续控制、批量控制、逻辑顺序控制及数据采集等多种功能。

3. 人-机联系手段

分散控制系统具有比常规控制系统更先进的人-机联系手段,其中最重要的一点,就是采用了 CRT 图形显示和键盘操作。人-机联系按照信息的流向分为“人→过程”联系和“过程→人”联系。在常规控制系统中,“人→过程”联系是通过各种操作器、定值器、开关和按钮等设备实现的,运行人员通过这些设备调整和控制生产过程;“过程→人”联系是通过各种显示仪表、记录仪表、报警装置、信号灯等设备实现的,运行人员通过它们了解生产过程的运行情况。所有这些传统的人-机联系设备都是安装在控制盘或者控制台上的。当生产过程的规模比较大、复杂程度比较高时,这些设备的数量会迅速增加,甚至达到令人无法应付的程度。例如,一台 600MW 的发电机组,如果采用常规控制系统,其控制盘的长度竟达 10m 以上。在如此庞大的监视和操作面中要迅速、准确地找到需要监视和操作的对象是比较困难的,也容易出错。这种情况暴露了常规控制系统的人-机联系手段的双向分散这一弱点。

在分散控制系统中,由于采用了 CRT 显示和键盘操作技术,人-机联系手段得到了根本性的改善。“过程→人”的信息直接显示在 CRT 屏幕上,运行人员可以随时调用他所关心的

显示画面来了解生产过程中的情况，同时，运行人员还可以通过键盘输入各种操作命令，对生产过程进行干预。由此可见，在分散控制系统中，所有的过程信息都被“浓缩”在CRT屏幕上，所有的操作过程也都“集中”在键盘上。因此，分散控制系统的人-机联系手段是双向集中的。除上述特点之外，分散控制系统还具有人-机联系一致性比较好的特点，因为键盘操作使许多操作过程得到统一，而遵循统一的操作规律是防止误操作的有力措施。

4. 可靠性

分散控制系统的可靠性比以往任何一种控制系统的可靠性都要高，这主要反映在以下几个方面。

(1) 由于系统采用模块化结构，每个过程控制站仅控制少数几个控制回路，个别回路或单元故障不会影响全局，而且元器件的高度集成化和严格的筛选有效地保证了控制系统的可靠性。

(2) 分散控制系统广泛采用了各种冗余技术，如对电源、通信系统、过程控制站等都采用了冗余技术。尽管常规控制系统也可以采用某些冗余技术，但由于其故障判断和系统切换都不易处理，所以常规控制系统的冗余往往只限于变送器或操作器。分散控制系统由于采用了计算机技术，因此上述问题很容易得到解决。原则上说，分散控制系统中的任何一个组成部分都可以采用冗余技术，这样就为设计出具有高可靠性的系统创造了条件。

(3) 分散控制系统采用软件模块组态方法形成各种控制方案，取消了常规系统中各种模件之间的连接导线，因此大大减少了由连接导线和连接端子所造成的故障。

5. 可维修性

可维修性反映了系统部件发生故障后对其进行维修的难易程度。可维修性差的系统需要较长的维修时间和较高的维修费用。常规控制系统的可维修性最差。由于它的部件种类繁多，稳定性较差，又缺少必要的诊断功能，所以维修工作十分困难。集中式计算机控制系统的可维修性比常规控制系统要好些，但由于它有一个庞大的、相互关联十分密切的硬件和软件系统，所以也要求维修人员具有较高的技术水平。分散控制系统的可维修性明显优于上述两类系统。它采用少数几种统一设计的标准模件，每一种模件包含的硬件比较简单。因为整个系统的控制功能不是由一台计算机包揽，而是由许多微处理机分别完成的，每台微处理机只担负着少量的控制任务，因此对它的要求并不是很高。另外，分散控制系统采用了比较完善的在线故障诊断技术，大多数系统的故障诊断定位准确度都可以达到模件级。通过各种人-机接口设备，运行员或工程师能够迅速发现系统设备故障的性质和位置，并且可以在不中断被控过程的情况下更换故障模件。

6. 安装费用

控制系统的安装费用主要包括电缆、导线的安装敷设费用和控制室、电子设备室的建筑费用。常规控制系统的安装费用比较高，这是因为由变送器、传感器和执行器到控制系统机柜之间需要很长的电缆，各种模件之间也要通过导线的连接组成不同的控制方案。另外，各种机柜和控制盘、台也要占用大量的建筑空间。

在分散控制系统中，控制方案的实现主要靠软件功能块的连接，因此大大减少了模件之间的连接线。另外，过程控制站可以采用地理分散的方式安装在被控过程的附近，这样就大大减少变送器、传感器和执行器与控制系统之间的连接电缆，不仅节省了导线、电缆的安装敷设费用，而且减少了控制系统在中央控制室所占用的空间。根据有关资料介绍，分散控制

系统的安装工作量仅为常规控制系统的 30%~50%，而控制室建筑面积仅为常规控制系统的 60%左右。可见，采用分散控制系统所取得的经济效益是十分显著的。

由于分散控制系统具有以上特点，所以它代表了当前计算机控制系统发展的主流和方向。目前，国外新建电厂和老厂改造几乎毫无例外地采用了分散控制系统，国内近期由国外引进的发电机组也大多采用分散控制系统。随着分散控制系统在研究、制造、推广和应用等方面的不断深入发展，它必将在电厂热工过程自动化中发挥更大的作用。

7.1.3 分散控制系统的分散方式

分散控制系统包含了功能分散、物理分散及地理分散三个不同的概念，下面以火电厂 DCS 中的功能分散为例进行具体说明。深入理解分散的概念，对于了解分散控制系统的本质是非常必要的。

火力发电厂的热力系统是一个复杂的大系统，为了便于理解、分析、控制和管理这样一个系统，常把它划分为若干个子系统，如制粉系统、燃烧系统、汽水系统等。同样，一个大系统的控制功能也要分解为一些基本的控制功能，这就是系统的功能分散。

一个单元机组的控制功能在纵向上可分为四个级，即单元机组级、功能组级、子功能组级和过程 I/O 级。

每一级在横向上再根据被控生产过程的特点分成若干个块，每块对应着一部分生产过程，这样就形成了图 7-2 所示的金字塔式的系统功能分层分块结构。

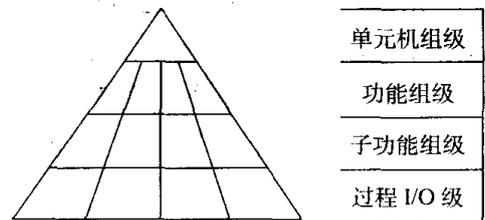


图 7-2 系统功能的分层分块结构

1. 单元机组级

单元机组级主要实现与机组协调控制系统高层控制任务有关的控制功能，如单元机组主控指令的形成、单元机组负荷给定值的形成、单元机组负荷给定值的控制、单元机组的协调控制以及单元机组的负荷能力计算，机组自动起动和停止的顺序控制等。

2. 功能组级

功能组级主要实现子系统中较高层次的顺序控制和连续控制。对于顺序控制，在功能组级形成控制子功能组的起动或停止命令。对于连续控制，功能组级完成某些控制回路的主控任务，向下一级控制器发出主控信号。功能组级往往对应着某些子系统，如燃烧系统、风烟系统、给水系统、过热蒸汽系统、再热蒸汽系统、高/低压旁路系统及凝结水系统等。

3. 子功能组级

它主要实现每个被控对象的顺序控制和连续控制。许多基本的控制作用均在这一级实现。在有些情况下，将这一级的功能移到上一级，即功能组级去实现。在控制功能比较简单情况下，子功能组级与功能组级可合并为一级。

4. 过程 I/O 级

它是分散控制系统与被控生产过程之间的桥梁，是每一个模拟量或开关量的入口和出口，它们直接与现场的变送器、执行器，继电器及电动装置等现场设备相连。

7.1.4 部分 DCS 产品

目前在国内的电力、石油、化工、纺织、建材、造纸、制药等行业已装备了上千套 DCS，

其中国外著名品牌占据多数。近年来，国内也正式推出了自行设计和制造的分散控制系统，并正在大力推广使用。表 7-1 列举了当前在国内应用较多的部分 DCS 产品。

表 7-1 国内外部分 DCS 产品

产品名称	生产厂家
WDPF-II, OVation	Westhouse (美国西屋公司)
INFI-90, INFI-90 OPEN	ABB Bailey Control (美国 ABB 贝利公司)
MAX1000+PLUS	MAX Control Sytem (美国 MCS 公司)
HIACS-3000/5000	日立 HITACHI (日本日立公司)
TDC-3000	Honeywell (美国霍尼威尔公司)
CENTUM	YOKOGAWA (日本横河机电公司)
Teleperm, SIMATIC PCS7	Siemens (德国西门子公司)
Spectrum, I/A Series	Foxboro (美国福克斯波罗公司)
HS-2000	中国北京和利时自动化工程有限公司
XDPS-400+	中国上海新华控制技术有限公司
SUPCON JX	浙大中控自动化有限公司

7.2 分散控制系统的典型设备

7.2.1 过程控制站

过程控制站是分散控制系统中实现过程控制的重要设备。根据控制方式的不同，过程控制站可以分为直接数字控制站、顺序控制站和批量控制站。其中，直接数字控制站主要用于生产过程中连续量（又称模拟量，如温度、压力、流量等）的控制；顺序控制站主要用于生产过程中离散量（又称开关量，如电机的起/停、阀门的开/关等）的控制；而批量控制站既可以实现连续量的控制，又可以实现离散量的控制。目前大多数分散型控制系统中的过程控制站均能同时实现连续控制、顺序控制和逻辑控制功能。因此，在没有必要加以区别时，统称为过程控制站。

1. 过程控制站的结构

过程控制站一般是标准的机柜式结构，如图 7-3 所示。

机柜的顶部装有风扇组件，其目的是带走机柜内部电子部件所散发出来的热量，机柜内部设若干层模块安装单元。图中上层安装处理器模块和通信模块，中间安装 I/O 模块，最下边安装电源组件，它为整个

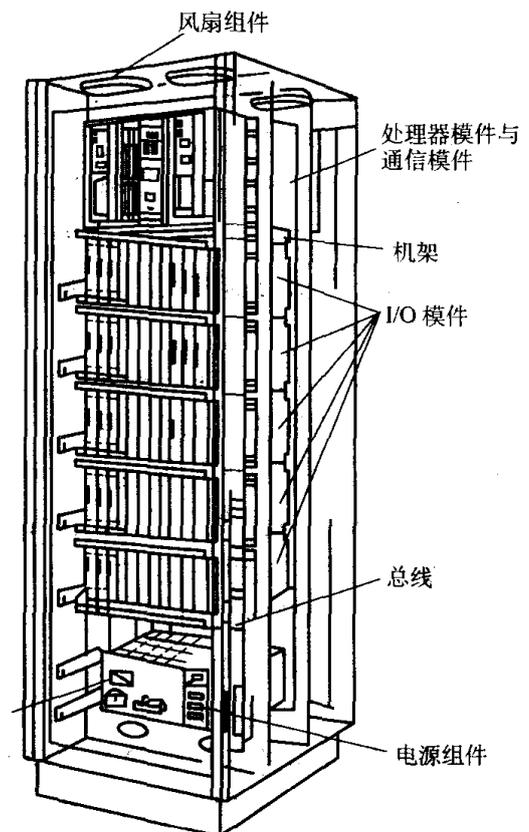


图 7-3 典型的现场过程控制站机柜

机柜提供电源。

机柜内还设有各种总线，如电源总线、接地总线、数据总线、地址总线及控制总线等。有些总线是由模件安装单元背后的印制电路板构成的，有些总线是由模件安装单元之间的扁平电缆或其他专用电缆构成的，有些是由装设在机柜侧面的汇流条构成的。

由 I/O 模件输入或输出的信号，经过机柜中的端子板与现场设备相连，通信模件与监控网络之间连接通过专用通信电缆实现。

过程控制站的系统结构如图 7-4 所示。

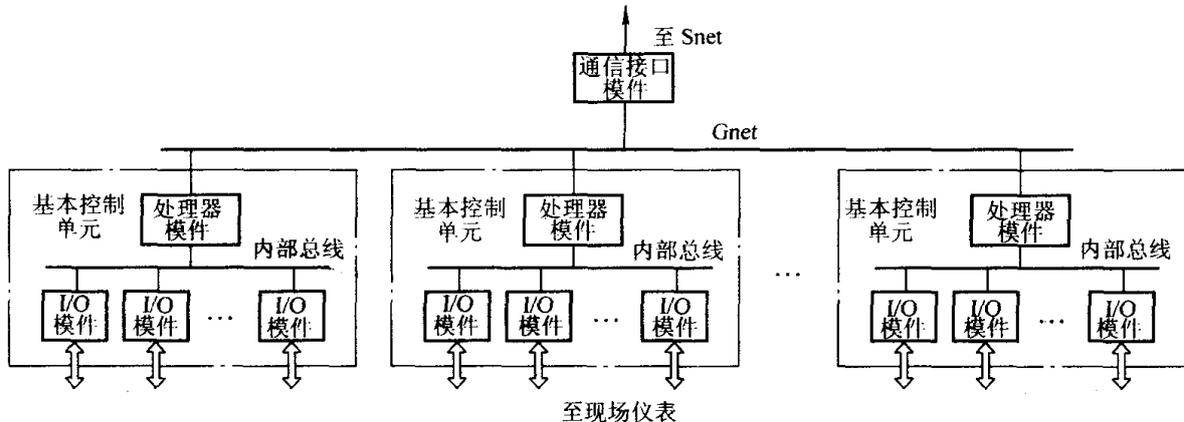


图 7-4 过程控制站的系统结构

一个过程控制站中可能包含一个或多个基本控制单元，基本控制单元是由一个完成控制或数据处理任务的处理器模件以及与其相连的若干个 I/O 模件所构成的。基本控制单元之间，通过控制网络 Cnet 连接在一起，Cnet 网络上的上传信息通过通信模件，送到监控网络 Snet，同理 Snet 的下传信息，亦通过通信模件和 Cnet 传到各个基本控制单元。在每一个基本控制单元中，处理器模件与 I/O 模件之间的信息交换由内部总线完成。内部总线可能是并行总线，也可能是串行总线。近年来，越来越多的分散型控制系统在处理器模件和 I/O 模件之间采用了串行总线，简化了系统结构，提高了信息传输的可靠性。

2. 基本控制单元

基本控制单元 (BCU, Basic Control Unit) 是分散控制系统中直接控制生产过程的那一部分硬件和软件的统称。基本控制单元接受来自传感器或变送器的过程变量，按照一定的控制策略，计算出所需要的控制变量，并把这些控制变量传送到生产过程中去，通过执行机构去调整生产过程中的温度、压力、流量、液位等被控变量。对于不同的分散控制系统，基本控制单元的结构和功能也有很大差异，一般可将基本控制单元划分为单回路型、功能分离型、多回路型等三种类型，如表 7-2 所示。

基本控制单元是由一个完成控制算法或数据任务的处理器模件以及与其相连的其他 I/O 模件所组成的。这些模件包括模拟量输入模件、模拟量输出模件、开关量输入模件、开关量输出模件、脉冲量输入模件等基本模件。还有专门用于汽轮机电液控制系统的液压伺服模件，用于连接现场总线的现场总线模件等专用模件。通信接口模件不一定与基本控制单元有一一对应的关系，在有些分散控制系统中，可能存在若干个基本控制单元共用一个通信接口模件的情况。

表 7-2 基本控制单元的结构对比

项目 \ 类型	单回路型	功能分离型	多回路型
控制规模	小, 一个 PID 控制回路或一台电动机	中, 几个控制回路, 小型的逻辑控制和顺序控制系统	大, 几十个控制回路, 大型的逻辑控制或顺序控制系统
控制功能	连续控制和顺序控制均可实现	连续控制和顺序控制分别由不同的控制器实现	连续控制和顺序控制均可实现
适用范围	适用于从小到各种规模的系统	即使在小型系统中, 也常需要用两种类型的控制单元	不适用于规模太小的系统
硬件要求	用廉价的硬件就可以达到所要求的性能	由于功能分离, 可以合理地使用硬件	由于控制功能很强, 对硬件的要求很高
通信要求	实现复杂控制需要的控制信号通信量大, 人-机接口通信量小	功能分离造成不同类型控制单元之间的通信量较大	控制单元之间的通信量小, 人-机接口通信量大
安全措施	每个控制单元只控制一个回路, 手动后备即可满足要求	在可靠性要求较高的情况下, 要采取冗余措施	因控制规模很大, 在大多数情况下均需采取冗余措施

处理器模块是完成过程控制的核心模块, 它完成用户所设计的各种控制策略。控制策略以组态文件的形式存储于非易失性存储器 (NVRAM) 中。由于有后备电池的支持, 即使在系统失电的情况下, 组态数据也不会丢失。鉴于处理器模块的重要性, 大多数分散型控制系统中的处理器模块都是可冗余设置的, 即同时设置两个或多个处理器模块, 一个工作, 另一个 (或几个) 备用。工作的处理器模块与备用的处理器模块具有相同的组态, 当主处理器模块工作时, 备用的处理器模块不断监视着主处理器模块, 一旦主处理器模块发生故障, 备用处理器模块就会自动接替控制任务, 保证生产过程的安全。

图 7-5 是处理器模块的原理框图, 它是由以下几部分所组成的。

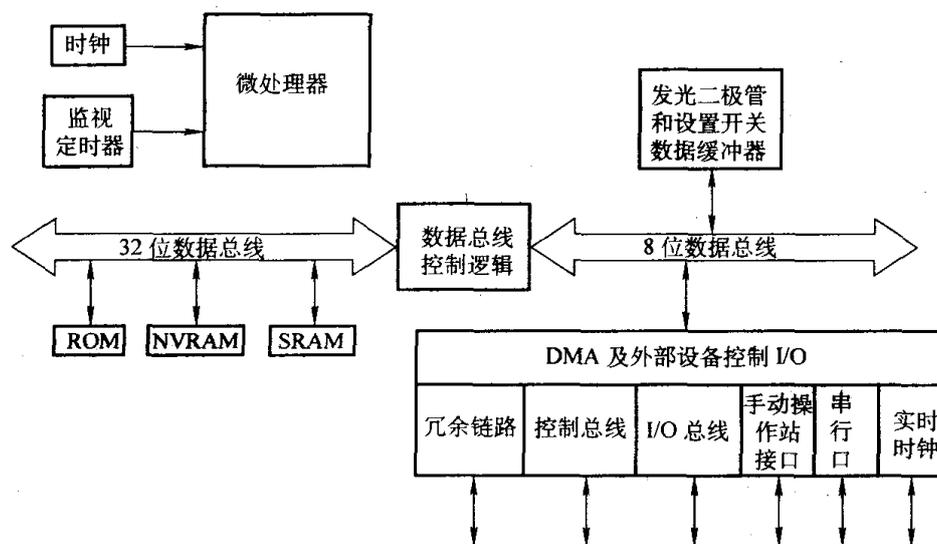


图 7-5 处理器模块原理框图

(1) 微处理器: 微处理器负责模块的操作和控制。目前一般采用 32bit 的微处理器。微处理器可以同步访问 32bit 字长的存储器, 或者异步访问各种字长的端口。由于微处理器负责模

件的操作,因此它与模件中的所有其他部分交换信息。微处理器的操作系统和功能块库,驻留于只读存储器(ROM)中。微处理器完成由功能块组态而建立的各种控制策略。

(2) 时钟和实时时钟:时钟为微处理器和相关的外部设备提供时钟信号,而实时时钟(RTC)为系统提供实时时间。

(3) 存储器:存储器包含只读存储器(ROM)、静态随机存储器(SRAM)和非易失存储器(NVRAM)。ROM保存微处理器的操作系统;SRAM用作存储暂存数据以及系统组态的副本;NVRAM保存系统组态(用功能块设计的控制策略)、批处理文件以及UDF和高级语言程序。NVRAM中存储的信息在系统失电时也不丢失。

(4) DMA控制器:直接数据存储(DMA)控制部分使各种通信链路能够直接与RAM交换数据,而不需要微处理器的干预。I/O扩展总线、双冗余的链路模拟量控制站或开关量控制站串行链路以及控制网络Cnet等通信链路,均支持DMA。采用DMA处理方式大大减轻了微处理器的负担,并且显著地提高了处理器模件的工作速度,因为它不需要再频繁地处理数据交换任务了。

(5) 控制网络:控制网络是一个高速通信网络,处理器模件利用这个网络来与过程控制站中的其他基本控制单元交换信息。控制网络一般是可冗余的,有两个独立的通道,处理器模件可以同时通过两个通道来发送和接收信息,并检查两个通道的一致性,这样就可以及时发现故障,使通信故障造成的影响减至最低。

(6) 冗余链路:处理器模件是可冗余组态的。冗余链路为主处理器模件和备用处理器模件提供两个并行链路,当主模件执行控制任务时,备用模件通过冗余链路,获得主模件的块输出信息,不管主模件因何原因发生故障,备用模件都可以在不中断被控过程的情况下自动接替控制任务。

由于主模件和备用模件在硬件上处于平等地位,并且可以相互切换,所以设置了并行链路。需要注意的是主模件和备用模件必须使用同一版本的固件,否则就会发生错误。

(7) I/O总线:处理器模件与I/O模件之间的数据传输通过I/O总线进行。I/O总线可能是并行总线也可能是串行总线。每一个挂在I/O总线上的I/O模件均有自己独立的地址,处理器模件可以通过I/O总线对I/O模件进行各种操作,如对模件进行读、写操作。

在处理器模件面板上有一些发光二极管用来表示模件的工作状态和错误信息。模件上的开关用来设定模件的地址和工作方式,这些信息均通过数据缓冲器和锁存器与微处理器交换信息。

(8) 手动操作站接口:当微处理器发生故障时,可以通过手动操作站对生产过程进行手动操作。因此,在处理器模件中,设置了手动操作站接口,可以通过该接口与手动操作站相连接。该接口也可以把相关的过程变量、给定值、报警状态等信息传送到手动操作站,并通过手动操作站显示出来。

(9) 串行口:处理器模件还可以设置串行口,以便通过标准的串行口(如RS-232或RS-485)与其他设备进行通信。串行口支持高级语言,如C语言和BASIC语言编程,并且具有光电隔离措施,以提高可靠性,防止模件的损坏。

3. 基本控制单元的软件

由于基本控制单元采用了以微处理器为基础的控制技术,它的硬件只能起到把信号输入到计算机或把信号从计算机中输出,并为程序的执行提供环境的作用。因此,要实现复杂的

控制功能，必须有软件的支持。基本控制单元的控制和计算功能是由程序存储器中的程序以及工作存储器中的参数决定的。在 BCU 中存储的程序和参数都是以二进制的形式存在的，这和其他任何一种计算机装置都相同。然而，用户并不希望直接同二进制数打交道，因此，必须有一种合适的语言来使用户能够方便地描述 BCU 所要完成的控制和计算功能，这就是基本控制单元的语言。

根据国际电工委员会制定的 IEC1131—3 标准，DCS 中常用 3 种为图形化编程的方法，即功能块图 (FBD, Function Block Diagram)、梯形图 (LD, Ladder Diagram)、顺序功能图 (SFC, Sequential Function Chart)，以及另外两种文本化语言，即指令表 (IL, Instruction List) 和结构化文本 (ST, Structured Text)。除此之外，在许多分散控制系统中还支持面向问题的语言 (POL, Problem Oriented Language) 和通用的高级语言，如 BASIC、Fortran 和 C 的编程，用户可以利用这些高级语言来实现一些特殊的控制算法。由于各 DCS 厂家推出的系统均使用自己的图形化编程语言，因此缺少通用性。

在目前的分散控制系统中，功能块是在 BCU 一级最流行的方法。功能块是一种预先编好程序的软件模块，用户确定它的参数，并且通过组态将其连接在一起。每一个功能块完成一种或几种基本的控制功能，如 PID 控制、开方运算、乘除运算等。功能块很像常规控制系统中的单元仪表或模件仪表，所以在有些分散控制系统中把功能块称为“内部仪表”，只不过这些内部仪表的功能是由软件实现的。每个内部仪表对应着 ROM 中的一段程序，而不是一个真正的“硬件仪表”。

在不同的分散控制系统中，功能块的描述方法是不同的，但一般可以归纳为以下几个要点：用一个矩形框表达一个功能块；功能块的输入、输出信号用有向线段来表示；矩形框内的符号代表功能块所实现的功能。要在系统中确定一个功能块的连接关系和控制功能，必须定义功能码（说明功能块所完成的功能）、块地址（运算输出结果的存放地址）、输入说明表和参数说明表等三类参数。

厂家把所有的控制和计算功能块都编好程序存放在基本控制单元的 ROM 中，用户只要选择适当的功能块，把它们连接在一起，并设置好必要的参数，就可以组成所需要的控制系统。功能块不仅能够实现许多常规仪表所完成的普通控制功能，而且能够实现许多复杂的高级控制功能，如自适应功能块可根据外部过程的情况自动调整功能块中的某些参数。

7.2.2 运行员操作站

分散控制系统的监控级是面向现场操作员和系统工程师的，与此相应的系统设备称为运行员操作站和工程师工作站。一般地，运行员操作站和工程师工作站是两类分离的不同结构，但也有一些分散控制系统将运行员操作站和工程师工作站的功能集成到统一的硬件上，只在操作权限上加以区分。为了方便，下面将运行员操作站和工程师工作站分别加以介绍。

分散控制系统的运行员操作站是处理一切与运行操作有关的人-机界面功能的网络节点，其主要功能是使运行员可以通过操作站及时了解现场运行状态、各种运行参数的当前值、是否有异常情况发生等，并可通过输出设备对工艺过程进行控制和调节，以保证生产过程的安全、可靠、高效和高质量。

1. 运行员操作站的硬件

运行员操作站对于硬件的可靠性要求很高，为了保证运行操作的安全可靠，分散控制系

统中均采用多个操作站并列运行的方式，运行员操作站与现场级的连接具有不同的结构，如图 7-6 所示。其中图 7-6a 中各个操作站与现场级通过监控网络（Snet）一一对应地连接；图 7-6b 的结构中，一台主机可以带多个 CRT 和外设（操作站）；图 7-6c 中与现场级的连接是冗余结构的多台服务器，服务器的任务是可靠地向所有客户机提供数据，其他操作站作为这些服务器的客户机，起监控作用的客户机的任务是建立操作员与系统之间的接口，图 7-6c 与图 7-6b 的区别是其中一台服务器的故障不会影响到操作站的正常运行。

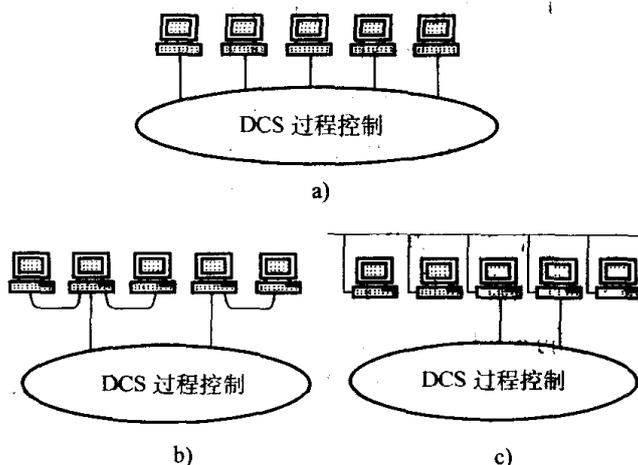


图 7-6 运行员操作站的不同构成方式

构成运行员操作站的硬件早期通常是由 DCS 的生产厂家自己设计，当前则往往由通用的高档微机、IPC 或工作站配以 DCS 厂家的通信接口设备组成，这样操作站的计算机软、硬件可以紧跟计算机技术的发展而随时更新。

运行员操作站一般由 IPC 或工作站、工业键盘和鼠标、大屏幕图形显示器、操作控制台、通信接口设备以及打印机、绘图仪等构成。有些运行员操作站也采用触摸屏。

2. 运行员操作站的软件

运行员操作站的功能是进行正常运行的工艺监视和运行操作，以 GUI 图形用户界面为平台，主要由总貌画面、分组画面、点画面、流程图画面、趋势曲线画面、报警显示画面和操作指导画面等构成，而这些功能都是通过运行员操作站的软件实现的。

运行员操作站的低层操作系统一般采用实时多任务的操作系统，如 RMX-DOS、UNIX、Windows NT、Windows 2000、Windows XP 等。运行员操作站通过监控网络（Snet）获取现场级的各种过程信息，建立并维护系统的实时数据库（不同的 DCS，实时数据库形式可能不同，有的分散于各个现场控制站，有的由运行员操作站建立维护，也有的由独立于运行员操作站的服务器建立维护）。这些网络通信和数据库软件是分散控制系统的用户系统软件的核心，各厂家的系统有很大的不同，但目前大多数均采用客户机/服务器（C/S）结构、DDE（动态数据交换）或 OPC（OLE for Process Control，用于过程控制的对象链接嵌入）接口技术等，下与现场级、上与管理级交换信息。

运行员操作站的用户软件部分，虽然各厂家的 DCS 提供的软件形式各不相同，但都要实现对过程进行监控的基本功能，主要包括过程画面的显示（包括操作功能）、报警管理、记录与报表、历史数据库的检索与处理等。其中最为重要的是通过过程画面的监视以及相应的实

施操作,实现对控制过程的监控。运行员操作站的过程画面主要有如下几种类型:工艺流程画面、单独的 I/O 点显示(包括过程状态信息、操作指导与处理信息、设计组态信息等,如图 7-7 所示)、控制设备的显示、趋势画面及帮助指导画面等。

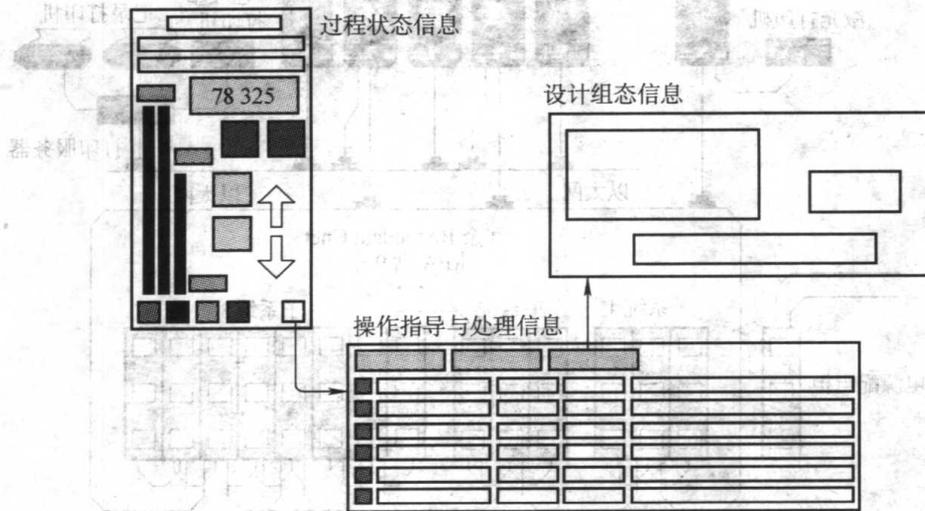


图 7-7 过程信息显示

7.2.3 工程师工作站

工程师工作站(EWS, Engineering Work Station)是分散控制系统中用于系统设计的工作平台,它的主要功能是为系统设计工程师提供各种设计工具,使工程师利用它们来组合、调用 DCS 的各种资源,将 DCS 的各种设备组织起来以发挥分散控制系统的各项功能。

工程师工作在硬件结构上与运行员操作站基本相同,但由于它的主要任务是系统设计,且系统运行时并不一定要求必须在线,这样对其可靠性的要求可以稍微降低,可以使用普通的计算机(操作系统可以是普通多任务系统),在系统中的优先级别往往设计成比其他网络节点低。

工程师工作站与运行员操作站的最主要的区别在软件上。工程师工作站的软件系统的主要功能包括系统组态、系统调试与设计更改、在线监督、运行记录及文件整理等。

1. 系统组态

分散控制系统对系统的设计过程是利用工程师工作站来组合 DCS 中所提供的控制算法或画面符号,而不是编制具体的计算机程序或软件,也不是用来描绘制造或安装用的图纸,所以习惯上把这种设计过程称为组态或组态设计。

系统组态设计的主要任务是利用 DCS 提供的所有控制、监视功能来设计实际的过程控制系统,组态设计包括很多方面。

(1) 系统硬件构成的总体设计

DCS 中使用的所有硬件设备,它们之间的电气、通信上的联系以及逻辑上的关系等设计,都可以在 EWS 上完成。设计者在 EWS 上可以很方便地把它们调出来,组成系统。

所谓组成系统就是画出它们之间的联系图、接线图。用 EWS 设计这些系统,应该像搭积木一样方便,而不必借助其他别的画图软件,如 AutoCAD,因为 EWS 是在组合现成的设备。

通过这样的设计结果，使 DCS 的使用者能够一目了然地了解 DCS 的总体布置情况。

图 7-8 表示了 DCS 的一幅总貌，而图 7-9 是系统数据库组态界面。

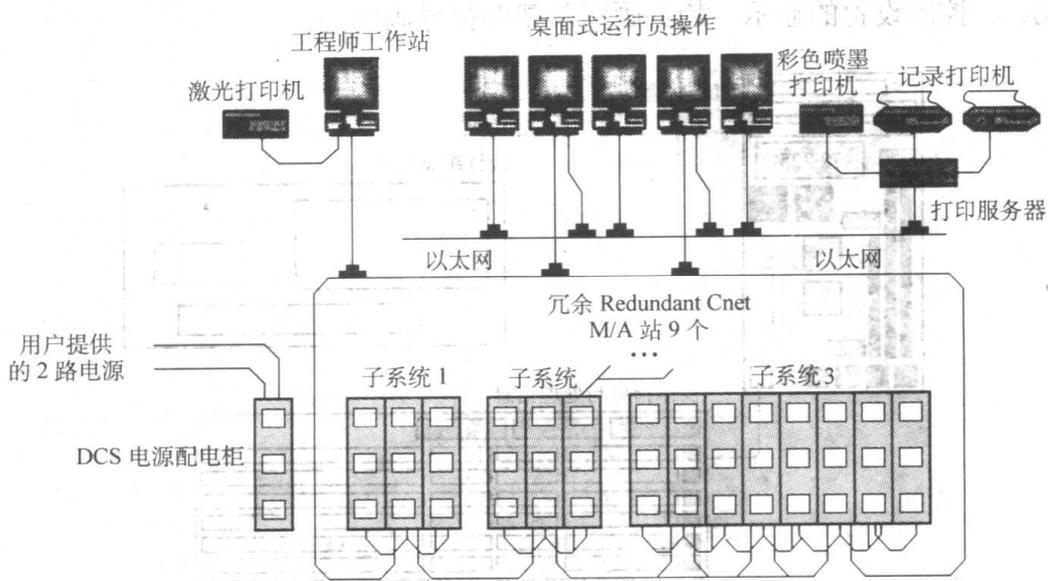


图 7-8 DCS 的总貌

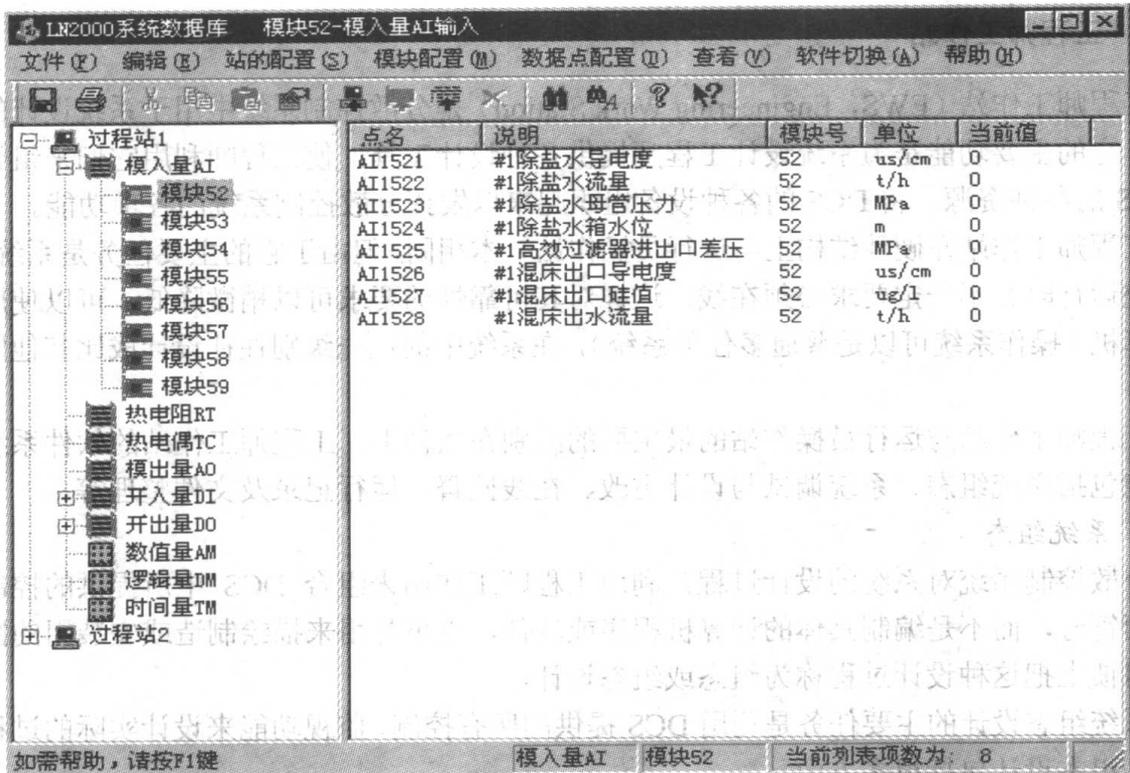


图 7-9 系统数据库组态界面

(2) 系统的硬件设计

硬件设计的结果是安装、组装和接线所用的工程图样。这通常包括过程控制单元中的模块布置、电源分配、现场 I/O 的连接、屏蔽与接地等方面的设计，使用 EWS 应可以很方便地

根据 DCS 的设计规范和过程控制的具体要求设计出这些接线图、配置图。

(3) 控制逻辑设计

控制逻辑的设计过程是根据 DCS 中给出的控制元件（算法，称功能码）组成控制方案的过程。简单说来，它包括采集现场信号的设计、控制运算、决策的设计、控制输出的设计、通信与传递信号的设计、系统诊断与处理方面的设计等。这方面的功能设计可以包括很多内容。图 7-10 所示为某电厂的算法功能块组态图。

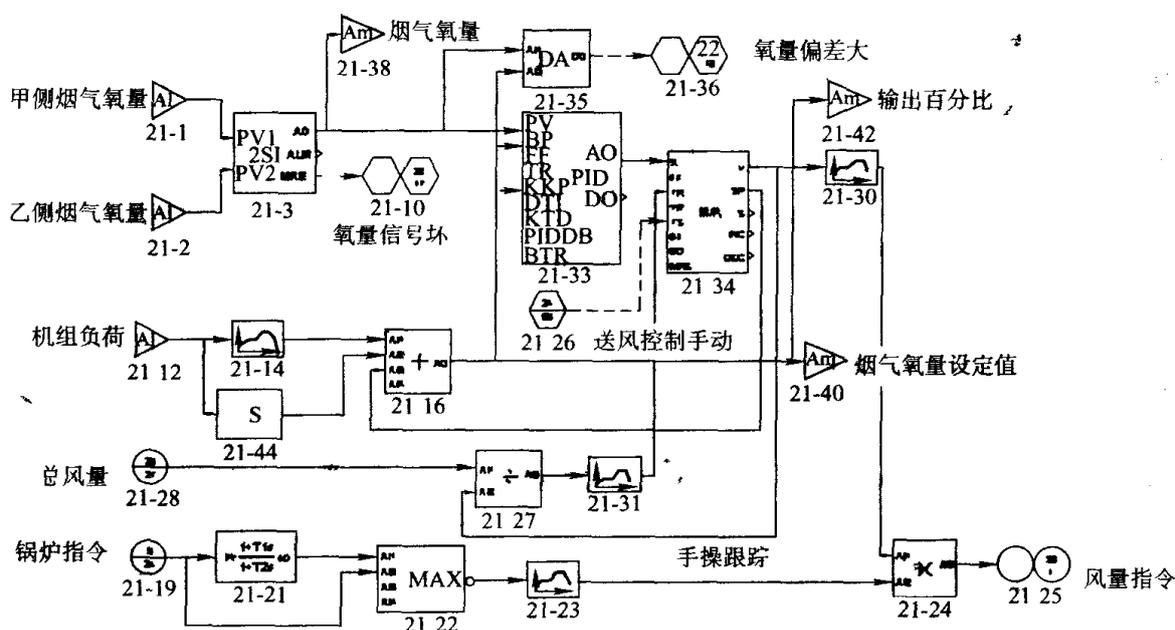


图 7-10 算法功能块组态图

(4) 人-机接口站组态设计

这部分设计是根据运行、维护等日常由运行员完成的任务，通过 DCS 的运行员操作站所提供的功能，设计并组成一个监控系统的过程。大体包括画面设计、报警设计、数据库设计、记录设计及帮助指导设计等。与过程控制的组态不同的是，人-机接口的组态的逻辑性不强，某幅画面中的设备画得大一些、小一些并不影响过程的安全，因此这部分设计常得不到习惯于逻辑思维的控制工程师的重视，然而事实上这部分设计是不能忽视的，从信息的组织方法这个意义上说，人-机接口系统设计得好坏，体现了设计者是否了解自己的 DCS，是否了解过程。图 7-11 所示为对运行员操作站进行的图形组态界面。

(5) 文件组态

这部分组态指的是为使 DCS 正常运行而编制的各种说明指导性文件。DCS 应是自我完善的，DCS 的安装、使用、应用方法、故障处理方法都应该在 DCS 中提供，根据问题性质的不同，或者在人-机接口站上提供，或者在工程师工作站上提供，而不应该让使用者“离线”查阅资料。

2. 系统调试与设计更改

由于 EWS 的设计中心的地位，所以在 DCS 的系统调试与设计更改过程中，EWS 发挥了最重要的作用。调试过程中，可通过 EWS 来确认控制组态是否执行得正确，这需要在线运行 EWS，以实时监控过程中的动态数据，发现问题后，修改组态。这些工作不是在其他入-机接

口上可以完成的，要在 EWS 上完成。虽然在运行员接口上可以看到一些问题，但是，对组态的调整应在一个点进行，是在工程师站上完成的。在下载了人-机接口的组态之后，如果有问题，也要在 EWS 上修改。最典型的功能是具有对组态的在线修改功能，如上限、下限定值的改变和控制参数的修整及对检测点甚至对某个现场 I/O 站的离线直接操作。

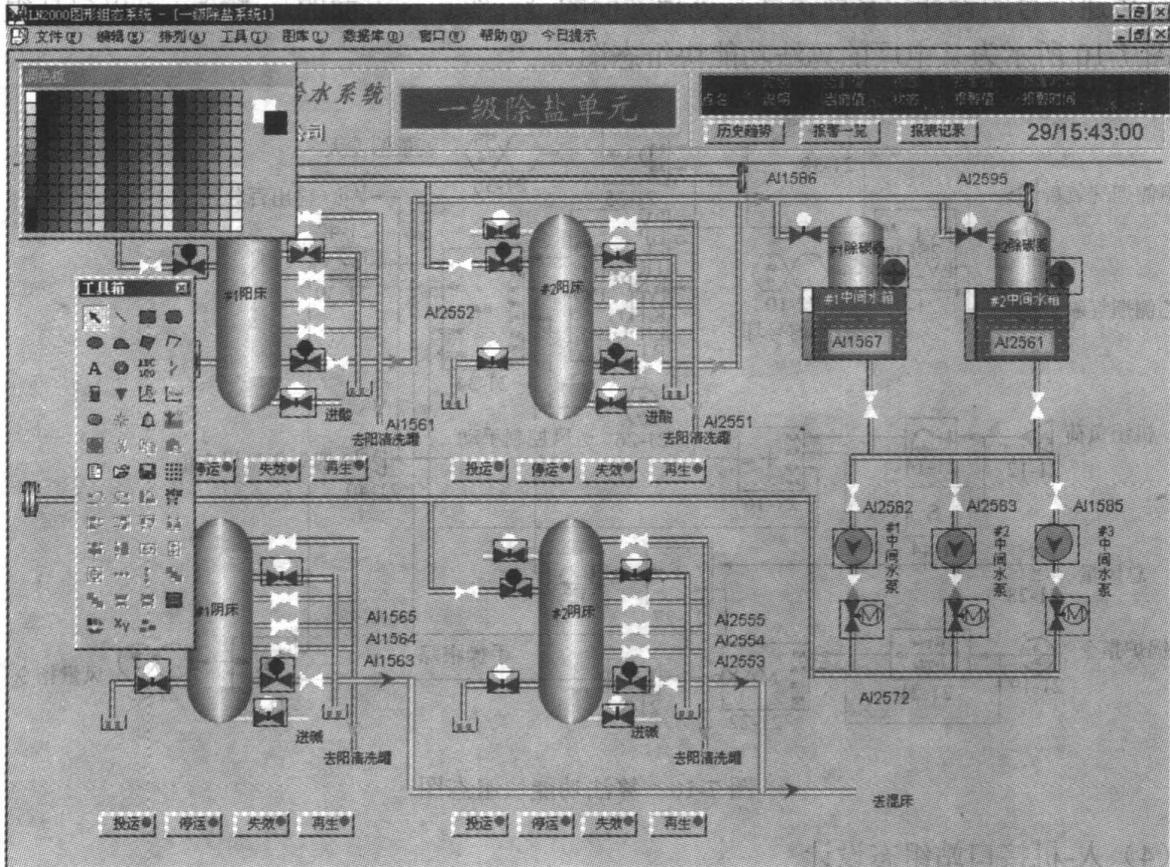


图 7-11 图形组态界面

3. 在线监督

与运行员操作站不同，工程师工作站必须对 DCS 本身的运行状态进行监视，包括各个现场 I/O 控制站的运行状态、各运行员操作站的运行情况、网络通信情况等。一旦发现异常，系统工程师必须及时采取措施，进行维修或调整，以使 DCS 能保证连续正常地运行，不会因对生产过程的失控造成损失。

4. 运行记录

EWS 的另一项作用是在调试结束之后，系统进入正常运行时，对某些特殊的过程参数做记录。这些特殊的参数包括：一些操作员并不关心而对系统的控制效果很重要的参数，尤其是一些控制回路之间相交联的变量的耦合程度；系统运行过程中 DCS 的负荷参数、通信负荷率、运算负荷率等。这些参数一方面对控制系统的整定很有必要，同时，它们又不是在较短期的调试过程中可以看得出来的，需要用相对长一些的时间来观察，所以在 EWS 上监视这些参数很有必要。运行员在人-机接口上监视当然也可以，但作为系统的设计者，要注意把这些用于控制或系统分析的信息与运行员日常处理的信息分开，以避免这两种不同类型的工作相互干扰，同时这些参数往往是不易确定的、灵活的，所以，在 EWS 上由工程师来监视比运行

员在操作站上监视要方便得多。再有, EWS 上对数据做分析的软件比运行员接口站的软件要丰富一些, 用这些软件对控制系统做分析会更有效。

5. 文件整理

EWS 既然作为设计中心而存在, 那么, EWS 上的文件对于 DCS 来说就非常重要了。同时, 按照前面提到的 DCS 应“自我完善”的概念, EWS 也应作为向用户提供 DCS 全部设计文件与系统说明文件的中心。实际上, 如果 EWS 管理得好, 利用得有效的話, 它还应提供系统调试、运行、管理过程中相关的全部文件, 使得与 DCS 相关的文件都可以在 EWS 上找到。这样会促使人们把有关的问题与 DCS 的设计、应用联系起来, 也便于获得解决问题的信息。目前很多人把这些工作放到 EWS 之外, 也就是 DCS 之外去解决, 使问题的解决过程与问题本身在两套系统上进行, 这不是一种高效率的方法。

7.3 现场总线控制系统概述

随着控制技术、计算机技术和通信技术的飞速发展, 数字化作为一种趋势正在从工业生产过程的决策层、管理层、监控层和控制层一直渗透到现场设备。现场总线的出现, 使数字通信技术迅速占领工业过程控制系统中模拟量信号的最后一块领地。一种全数字化的、全开放式的、可互操作的新型控制系统——现场总线控制系统, 正在向人们走来。现场总线控制系统的出现代表了工业自动化领域中一个新纪元的开始, 并将对该领域的发展产生深远的影响。

7.3.1 现场总线的概念

现场总线(FieldBus)是用于过程自动化或制造自动化中的、实现智能化现场设备(如变送器、执行器、控制器)与高层设备(如主机、网关、人-机接口设备)之间互连的、全数字、串行、双向的通信系统。通过它可以实现跨网络的分布式控制。按照国际电工委员会(IEC)标准和现场总线基金会(FF)的定义: 现场总线是连接智能现场设备和自动化系统的数字式、双向传输、多分支结构的通信网络。

现场总线的本质含义表现在以下几个方面。

1. 现场通信网络

现场总线作为一种数字式通信网络一直延伸到生产现场中的现场设备, 使过去采用点对点式的模拟量信号传输或开关量信号的单向并行传输变为多点一线的双向串行数字式传输。

2. 设备互连

现场设备是指位于生产现场的传感器、变送器和执行器等。这些现场设备可以通过现场总线直接在现场实现互连, 相互交换信息。而在 DCS 系统中, 现场设备之间是不能直接交换信息的。

3. 互操作性

现场设备种类繁多, 一个制造商可能不能提供一个工业生产过程所需要的全部设备。另外, 用户也不希望受制于某一制造商。这样, 就有可能在一个现场总线控制系统中, 连接多个制造商生产的设备。所谓互操作性是指来自不同厂家的设备可以相互通信, 并且可以在多厂家的环境中完成功能的能力。它体现在: 用户可以自由地选择设备, 而这种选择独立于供

应商、控制系统和通信协议；制造商具有增加新的、有用的功能的能力；不需要专有协议和特殊订制驱动软件和升级软件。

4. 分散功能块

现场总线控制系统把功能块分散到现场仪表中执行，因此取消了传统的 DCS 系统中的过程控制站。例如，现场总线变送器除了具有一般变送器的功能之外还可以运行 PID 控制功能块。类似地，现场总线执行器除了具有一般执行器的功能之外，还可以运行 PID 控制功能块和输出特性补偿块，甚至还可以实现阀门特性自校验和阀门故障自诊断功能。

5. 现场总线供电

现场总线除了传输信息之外，还可以完成为现场设备供电的功能。总线供电不仅简化了系统的安装布线，而且还可以通过配套的安全栅实现本质安全系统，为现场总线控制系统在易燃易爆环境中应用奠定了基础。

6. 开放式互连网络

现场总线为开放式互连网络，既可与同层网络联连，也可与不同层网络互连。现场总线协议是一个完全开放的协议，它不像 DCS 那样采用封闭的、专用的通信协议，而是采用公开化、标准化、规范化的通信协议。这就意味着来自不同厂家的现场总线设备，只要符合现场总线协议，就可以通过现场总线网络连接成系统，实现综合自动化。

7.3.2 现场总线控制系统的构成

现场总线控制系统作为新一代过程控制体系结构目前还处在发展阶段，各种不同的现场总线控制系统层出不穷，其系统结构形态各异，有的是按照现场总线体系结构的概念设计的新型控制系统，有的是在现有的 DCS 系统上扩充了现场总线的功能。为了便于讨论，这里将重点放在监控级、控制级和现场级。监控级之上的管理级、决策级等不予考虑。因此可以把 FCS 分为三类：一类是由现场设备和人-机接口组成的两层结构的 FCS；另一类是由现场设备、控制站和人-机接口组成的三层结构的 FCS；还有一类是由 DCS 扩充了现场总线接口模块所构成的 FCS。

1. 具有二层结构的 FCS

具有二层结构的 FCS 如图 7-12 所示。它是由现场设备和人-机接口两部分所组成。现场设备包括符合现场总线通信协议的各种智能仪表，如现场总线变送器、转换器、执行器和分析仪表等。由于系统中没有单独的控制器，系统的控制功能全部由现场设备完成，如常规的 PID 控制算法可以在现场总线变送器或执行器中实现。人-机接口设备一般有运行员操作站和工程师工作站。运行员操作站或工程师工作站通过位于机内的现场总线接口卡和现场总线与

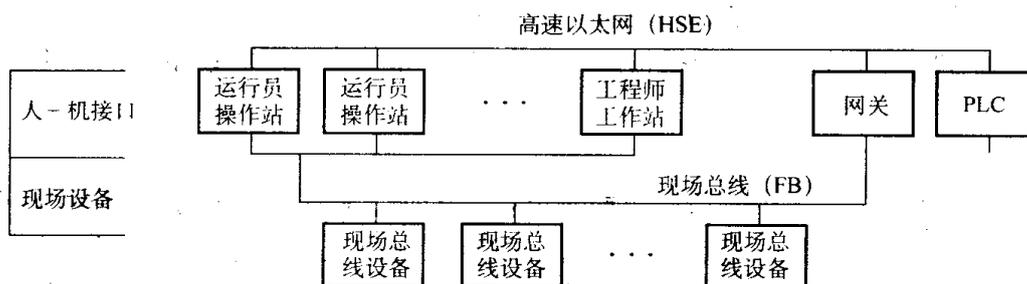


图 7-12 具有二层结构的现场总线控制系统

现场设备交换信息，人-机接口之间的或与更高层设备之间的信息交换，通过高速以太网（HSE）实现。高速以太网上还可以连接需要高速通信的现场设备，如可编程逻辑控制器（PLC）等。低速现场总线还可以通过网关连接到高速现场总线上，通过高速现场总线与人-机接口设备或其他高层设备交换信息。

这种现场总线控制系统结构适合于控制规模相对较小、控制回路相对独立、不需要复杂协调控制功能的生产过程。在这种情况下，由现场设备所提供的控制功能即可以满足要求。因此在系统结构上取消了传统意义上的控制站，控制站的控制功能下放到现场，简化了系统结构。但带来的问题是不便于处理控制回路之间的协调问题，一种解决办法是将协调控制功能放在运行员操作站或者其他高层计算机上实现；另一种解决办法是在现场总线接口卡上实现部分协调控制功能。

2. 具有三层结构的 FCS

具有三层结构的 FCS 如图 7-13 所示。它由现场设备、控制站和人-机接口三层所组成。其中现场设备包括各种符合现场总线通信协议的智能传感器、变送器、执行器、转换器和分析仪表等；控制站可以完成基本控制功能或协调控制功能，执行各种控制算法；人-机接口包括：运行员操作站和工程师工作站，主要用于生产过程的监控以及控制系统的组态、维护和检修。系统中其余各部分的功能同前所述，故不赘述。

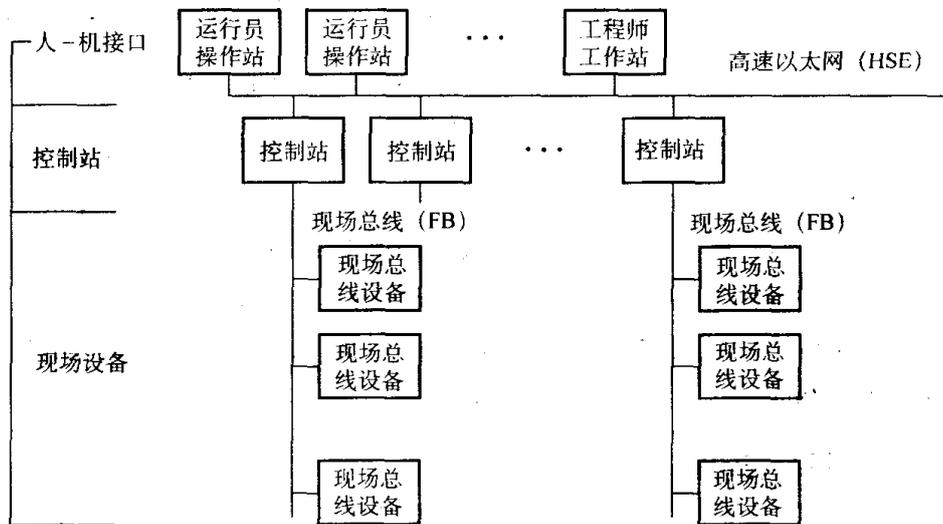


图 7-13 具有三层结构的现场总线控制系统

这种现场总线控制系统的结构虽然保留了控制站，但控制站所实现的功能与传统的 DCS 有很大区别。在传统的 DCS 中，所有的控制功能，无论是基本控制回路的 PID 运算，还是控制回路之间的协调控制功能均由控制站实现。但在 FCS 中，低层的基本控制功能一般是由现场设备实现的，控制站仅完成协调控制或其他高级控制功能。当然，如有必要，控制站本身是完全可以实现基本控制功能的。这样就可以让用户有更加灵活的选择。具有三层结构的 FCS 适合用于比较复杂的工业生产过程，特别是那些控制回路之间关联密切、需要协调控制功能的生产过程，以及需要特殊控制功能的生产过程。

3. 由 DCS 扩充而成的现场总线控制系统

现场总线作为一种先进的现场数据传输技术正在逐渐渗透到新兴产业中的各个领域。

DCS 系统的制造商同样也在利用这一技术改进现有的 DCS 系统，他们在 DCS 系统的 I/O 总线上挂接现场总线接口模件，通过现场总线接口模件扩展出若干条现场总线，然后经现场总线与现场智能设备相连。如图 7-14 所示。

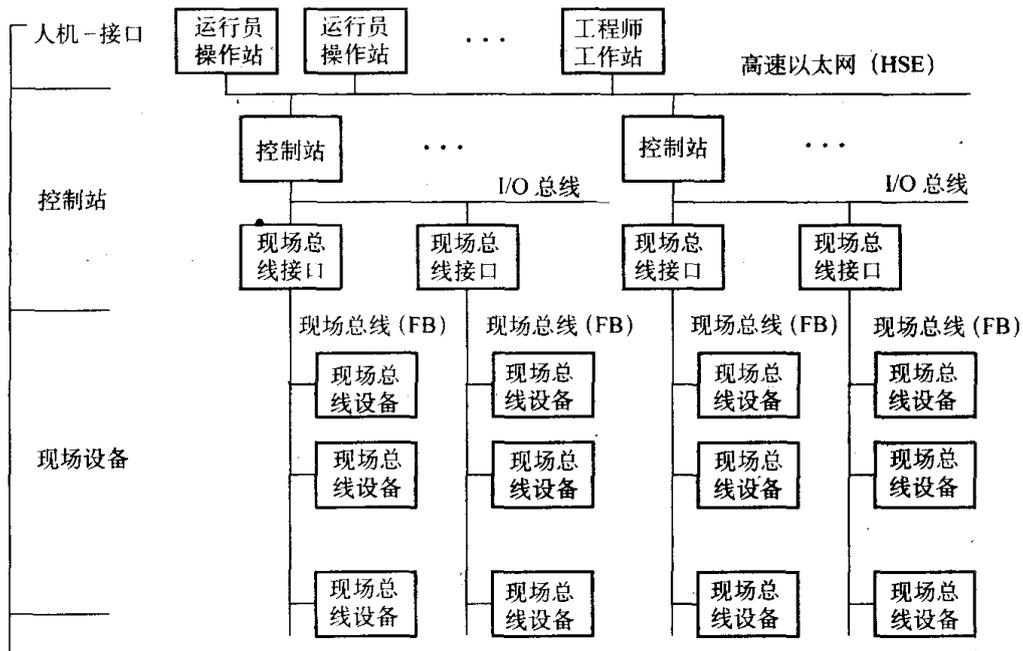


图 7-14 由 DCS 扩充而成的现场总线控制系统

这种现场总线控制系统是由 DCS 演变而来的。因此，不可避免地保留了 DCS 的某些特征。例如，I/O 总线 and 高层通信网络可能是 DCS 制造商的专有通信协议，系统开放性要差一些。现场总线装置的组态可能需要特殊的组态设备和组态软件，即不能在 DCS 原有的工程师工作站上对现场设备进行组态等。这种类型的系统比较适合于在用户已有的 DCS 系统中进一步扩展应用现场总线技术，或者改造现有 DCS 系统中的模拟量 I/O，提高系统的整体性能和现场设备的维护管理水平。

7.3.3 现场总线控制系统的特点

现场总线是现场仪表所采用的双向数字通信方式，是自动化仪表的最新技术成果，它将取代当今现场仪表中广泛使用的 4~20mA 标准模拟通信方式。

现场总线具有以下特点。

- 1) 一根双绞线可连接多台设备，从而减少了导线数量，降低配线成本。
- 2) 由于采用数字传输方式，可以实现高精度的信息处理，提高控制质量。
- 3) 由于实现了多重通信，除了可以传送过程变量 PV、控制变量 MV 值之外，还可以传送大量的现场设备管理信息。
- 4) 现场仪表之间可以通信，实现了现场仪表的自律分散控制。
- 5) 由于现场总线仪表具有互操作性，不同厂家的仪表可以自由组合，为用户提供了更广泛的选择余地。
- 6) 实现了测量仪表、电气仪表、分析仪表的综合化。
- 7) 在控制室就可以对现场仪表进行调试、校验、诊断和维护。

下面就将详细讨论这些特点。

1. 通信方式

目前现场仪表有三种通信方式，即模拟通信方式、混合通信方式和数字通信方式。数字通信方式，即现场总线通信方式，与模拟通信方式和混合通信方式比较，具有信息精度高、传送速度快、传送数据量大、可以实现双向通信等优点。

(1) 模拟通信方式

模拟通信方式是用 4~20mA 直流模拟信号传送信息，其拓扑结构为一对一方式，即一对线只能接 1 台现场仪表，信息只能单方向传输。因此，由现场设备接收信息的信号线和给现场设备发送控制信号的控制线是分开的。

(2) 混合通信方式

混合通信方式是在 4~20mA 模拟信号上叠加数字信号的通信方式，其中模拟信号用于传送过程变量或控制变量；数字信号用于传输现场仪表的调整信息、诊断信息和状态信息，以便进行现场仪表的远程设定和设备管理。

由于混合通信方式是厂家自主开发的，不同厂家的设备之间不能进行信息交换。虽然混合通信方式实现了数字通信，但仍然以 4~20mA 模拟通信为主体，因此混合通信方式的数据传输速度和传输容量仍比现场总线通信方式要低。

(3) 现场总线通信方式

现场总线通信方式与模拟通信方式和混合通信方式不同，它是完全的数字通信方式。现场总线通信方式可以进行双向通信，因此与模拟通信方式和混合通信方式不同，它可以传送多种数据。在模拟通信方式中，一对配线只能接一台现场仪表，现场总线通信方式没有这种限制，一根现场总线配线可以连接多台现场仪表。

现场总线通信方式推进了国际标准化，确保了互操作性。使用现场总线能够解决在混合通信方式中传送速度慢、互操作性差等问题。

各种通信方式的比较见表 7-3。

表 7-3 各种通信方式的比较

	现场总线	混合	模拟
拓扑结构	多点	点对点	点对点
传输方式	数字传输	模拟加数字传输	模拟传输
传输方向	双向	单向（模拟） 双向（数字）	单向
信号种类	多重信号	部分多重信号	单信号
标准化	是	不是	是

2. 节省导线、电缆及其安装费用

采用现场总线技术，可以实现多仪表互连、多变量检测和多变量传送。

(1) 多仪表互连

一根线上连接多台现场仪表就称为多仪表互连，多仪表互连的最大优点就是节省大量的导线、电缆及其安装费用。图 7-15 是多仪表互连的图示。

在模拟通信方式中，一对导线只能连接一台现场仪表。但现场总线通信方式中采用多仪

表互连方式，一对导线可以连接多台现场仪表，并且还可以在配线好的线上补接现场仪表，而不必重新敷设电缆。这样，就为在工程中修改和扩充控制系统提供了极大的方便。

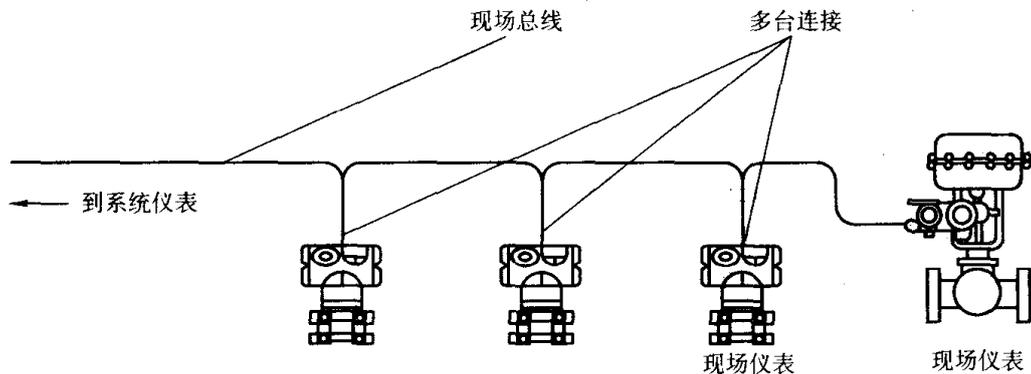


图 7-15 多仪表互连

过去，现场仪表的连接需要大量的导线和电缆。若使用现场总线通信方式，因采用了多仪表互连方式，导线和电缆的用量大大减少，其敷设成本也显著降低，许多现场仪表可以连接到一条现场总线上，控制系统的规模很容易扩大，装置的自动化水平得以提高。

(2) 多变量检测

所谓多变量检测是指一台现场仪表可以同时检测多个过程变量，也称为多变量测量。在过去的模拟通信方式中，测量一个变量就需要一对导线，因此，每台现场仪表只能测量一个过程变量；而采用了现场总线通信方式后，一台现场仪表就可以同时检测多个过程变量。如图 7-16 所示。

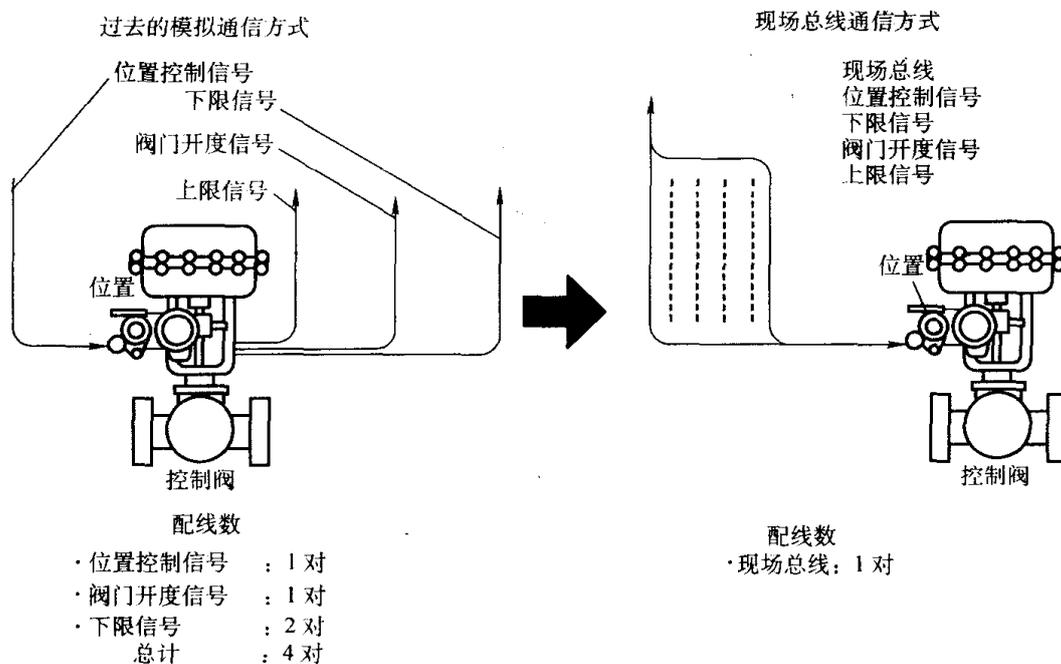


图 7-16 单变量检测和传输与多变量检测和传输

(3) 多变量传送

现场总线可以实现多变量传送，因此一台测量多变量的现场仪表只要用一对导线，就可以把该现场仪表测量的变量全部传送出去。参见图 7-16。

由图 7-16 可见,一个气动调节阀需要传送一个控制信号、一个开度信号和两个限位信号。如果采用传统的现场仪表,则需要 4 对导线;而采用现场总线仪表,则只需要 1 对导线即可。

另外,利用现场总线仪表的多变量检测和多变量传送特性还可以实现一些特殊的系统功能,如变送器周围环境温度的监测、变送器导压管堵塞的监测等。

3. 传输精度的提高

引入现场总线可消除模拟通信方式中数据传输时产生的误差,提高传输精度。

模拟通信方式中产生误差的原因有以下 3 个方面:

- 1) 现场仪表中 D/A 转换产生误差。
- 2) 模拟信号传输产生误差。
- 3) 系统仪表的 A/D 转换产生误差。

在模拟通信方式中,传输装有微处理器的现场仪表数据时,数据进行 A/D、D/A 转换时产生误差。同时,模拟信号在传输过程中也会产生误差。使用现场总线,可以消除转换误差和传输误差。现场总线是用数字信号传输数据,数字信号传输与模拟信号传输的不同之处在于:前者不产生信号传输过程中所带来的误差。现场总线中的数据以数字状态传送,不需要 A/D、D/A 转换,因此,也不会产生转换误差。引入现场总线,就消除了上述 3 种误差,提高了传送精度。

4. 多层次的信息传输

现场总线除了可以传输过程变量 PV 和控制变量 MV 之外,还可以传输其他各种信息。现场仪表之间也可以通过现场总线交换信息。由于现场总线实现了多种数据的双向传送,与模拟通信方式相比,它具有更多的功能。

模拟通信方式只能传输 PV 值和 MV 值;混合通信虽然实现了多种数据的传输,但它的传输速度慢,系统仪表和现场仪表只能是一对一的通信;现场总线解决了混合通信存在的问题,它的传输速度快,系统仪表和现场仪表之间可以实现多对多的通信。它可以双向传递各种数据,实现底层现场仪表和高层系统仪表,以及底层现场仪表之间的、多层次的信息传输。

5. 系统控制功能的分散化

采用现场总线,实现了控制系统的综合管理和自律分散控制。

(1) 现场仪表具有综合管理功能

使用现场总线不但可以传输 PV 值和 MV 值,而且还可以传输很多用于设备管理的信息。所以,现场仪表能够实现更多的功能。例如,具有温度、压力校正的现场总线流量变送器,具有阀门流量特性补偿的现场总线阀门定位器等。

(2) 系统控制功能的自律分散

现场仪表具有高层次功能,在某种程度上承担了系统仪表的控制功能。这样就有利于实现现场仪表控制功能的自律分散化,从而进一步提高整个系统的可靠性。

(3) 系统控制功能的下移

随着现场仪表的高功能化以及控制功能的分散化,今后系统仪表的部分控制功能会向下移动,进入现场总线仪表。例如, PID 控制功能既可以在现场仪表中实现,也可以在系统仪表中实现,根据不同的控制对象,用户可以自由选择。当控制回路之间的关联密切、需要协调时,可以将 PID 控制功能放在系统仪表中实现;相反,当控制回路之间的独立性较强时,可以将其放在现场仪表中实现。

6. 互操作性

现场总线通信方式正在向国际化推进。标准化确保了互操作性的实现。不同厂家的设备可以混合使用,控制系统的组合是自由的。

过去的混合通信方式能够传送数字信号,但厂家使用自己开发的通信方法,不同厂家的设备间相互交换信息十分困难。

现场总线推进了国际化,确保了互操作性。因此,凡是符合现场总线通信协议的现场总线设备,不管是哪一个厂家生产的,都可以相互交换信息。这样,用户就不必围绕着一家公司选择设备,控制系统构成的自由度大大增加。用户能够以最优的性能/价格比构成符合自己要求的控制系统。

7.4 几种典型的现场总线

自 20 世纪 80 年代末以来,有几种类型的现场总线技术已经发展成熟并且广泛地应用于特定的领域。这些现场总线技术各具特点,有的已经逐渐形成自己的产品系列,占有相当大的市场份额。下面就是几种典型的现场总线。

7.4.1 CAN 总线

CAN (Control Area Network) 是控制局域网的缩写,它是由德国 Bosch 公司推出、最早用于汽车内部监测部件与控制部件的数据通信网络。现在已经逐步发展应用于其他控制领域。CAN 规范现已被国际标准化组织采纳,成为 ISO11898 标准。CAN 协议也是建立在 ISO/OSI 模型基础上的,它采用了 OSI 底层的物理层、数据链路层和高层的应用层,其信号传输介质为双绞线。最高通信速率为 1Mbit/s (通信距离 40m),最远通信距离可达 10km (通信速率为 5kbit/s),节点总数可达 110 个。

CAN 的信号传输采用短帧结构,每一帧的有效字节数为 8 个,因而传输的时间短,受干扰的概率低,每帧信息均有 CRC 校验和其他检错措施,通信误码率极低。CAN 节点在错误严重的情况下,具有自动关闭总线的功能,这时故障节点与总线脱离,使其他节点的通信不受影响。

7.4.2 LonWorks 总线

LonWorks (Local Operating Network) 是局部操作网络的缩写。它是由美国 Echelon 公司研制,于 1990 年正式公布的现场总线网络。它采用了 ISO/OSI 模型中完整的七层通信协议,采用了面向对象的设计方法,通过网络变量把网络通信设计简化为参数设置,其最高通信速率为 1.25Mbit/s (通信距离不超过 130m),最远通信距离为 27000m (通信速率为 78Kbit/s),节点总数可达 32000 个。网络的传输介质可以是双绞线、同轴电缆、光纤、射频、红外线及电力线等。

LonWorks 的信号传输采用可变长帧结构。每帧的有效字节可为 0~288 个。LonWorks 所采用的 LonTalk 通信协议被封装在称之为 Neuron 的神经元芯片中。芯片中有 3 个 8bit CPU,一个用于实现 ISO/OSI 模型中的第 1 层和第 2 层的功能,称为媒体访问控制处理器;第二个用于完成 3~6 层的功能,称为网络处理器;第三个应用于第 7 层,称为应用处理器。芯片中

还具有信息缓冲区, 以实现 CPU 之间的信息传递, 并作为网络缓冲区和应用缓冲区。

7.4.3 Profibus 总线

Profibus 是过程现场总线 (Process Field Bus) 的缩写, 它是德国国家标准 DIN 19245 和欧洲标准 EN 50170 所规定的现场总线标准。Profibus 由 3 个兼容部分组成, 即 Profibus-DP、Profibus-PA 和 Profibus-FMS。其中 Profibus-DP 是一种高速低成本通信系统。它按照 ISO/OSI 参考模型定义了物理层、数据链路层和用户接口; Profibus-PA 专为过程自动化设计, 可使变送器与执行器连接在一根总线上, 并提供本质安全和总线供电特性。Profibus-PA 采用扩展的 Profibus-DP 协议, 另外还有现场设备描述的 PA 行规; Profibus-FMS 根据 ISO/OSI 参考模型定义了物理层、链路层和应用层, 其中应用层包含了现场总线消息规范 (FMS, Fieldbus Message Specification) 和低层接口 (LLI, Lower Layer Interface), 最高通信速率为 12Mbit/s (通信距离不超过 100m), 最大通信距离为 1200m (通信速率为 9.6kbit/s)。如果采用中继器可延长至 10km, 其传输介质可以是双绞线或光缆。每个网络可挂 32 个节点, 如带中继器, 最多可挂 127 个节点。

Profibus 采用定长或可变长帧结构, 定长帧一般为 8B, 可变长帧每帧的有效字节数为 1~244 个。近年来, 多家公司联合开发 Profibus 通信系统的专用集成电路芯片, 目前已经能将 Profibus-DP 协议全部集成在一块芯片之中, 如被称为 Profibus 控制器的 SPC3 芯片、主站控制器 PBM 芯片、从站控制器 PBS01 芯片等。

7.4.4 WorldFIP 总线

WorldFIP 意为世界工厂仪表协议 (World Factory Instrument Protocol)。最初由 Cegelec 等几家法国公司在原有通信技术的基础上根据用户的要求所制定, 随后即成为法国标准, 后来又采纳了 IEC 物理层国际标准 (IEC61158-2), 并命名为 WorldFIP。WorldFIP 是欧洲现场总线标准 EN50170-3。WorldFIP 组织成立于 1987 年, 目前包括有 ALSTOM、Schneider、Honeywell 等世界著名大公司在内的 100 多个成员。WorldFIP 协议按照 ISO/OSI 参考模型定义了物理层、数据链路层和应用层。WorldFIP 采用有调度的总线访问控制。通信速率分别为 31.35kbit/s、1Mbit/s、2.5Mbit/s, 对应的最大通信距离分别为 5000m、1000m、500m, 其通信介质为双绞线。如果采用光纤, 其最大通信距离可达 40km。每段现场总线的最大节点数为 32 个, 使用分线盒可连接 256 个节点。整个网络最多可以使用 3 个中继器, 连接 4 个网段。

WorldFIP 采用可变长帧结构, 每帧的最大字节数为 256 个。适合于包括 TCP/IP 在内的各种类型的协议数据单元。WorldFIP 可以提供各种专用通信芯片, 如具有总线仲裁器功能的 FULLFIP2、具有总线仲裁器功能并且支持双处理器结构的 FIPIU2, 以及无总线仲裁器功能 MICROFIP 等。

7.4.5 HART 总线

HART (Highway Addressable Remote Transducer) 是可寻址远程传感器数据通路的缩写。最早由 Rosemount 公司开发, 得到了 80 多家仪表公司的支持, 并于 1993 年成立了 HART 通信基金会。HART 协议参考了 ISO/OSI 参考模型的物理层、数据链路层和应用层。其主要特点是采用基于 Bell 202 通信标准的频移键控 FSK 技术。在现有的 4~20mA 模拟信号上叠加

FSK 数字信号, 以 1200Hz 的信号表示逻辑 1, 以 2200Hz 的信号表示逻辑 0, 通信速率为 1200bit/s, 单台设备的最大通信距离为 3000m, 多台设备互联的最大通信距离为 1500m, 通信介质为双绞线, 最大节点数为 15 个。

HART 采用可变长帧结构, 每帧最长为 25 个字节, 寻址范围为 0~15。当地址为 0 时, 处于 4~20mA 与数字通信兼容状态。而当地址为 1~15 时, 则处于全数字状态。HART 协议的应用层规定了三类命令: 第一类是通用命令, 适用于遵循 HART 协议的所有产品; 第二类称为普通命令, 适用于遵循 HART 协议的大多数产品; 第三类称为特殊命令, 适用于遵循 HART 协议的特殊设备。另外, HART 还为用户提供了设备描述语言 (DDL, Device Description Language)。

7.4.6 FF 总线

FF (Fieldbus Foundation) 是现场总线基金会的缩写。现场总线基金会是国际公认的、惟一不附属于某企业的、非商业化的国际标准化组织。其宗旨是制定单一的国际现场总线标准。FF 协议的前身是以美国 Fisher-Rosemount 公司为首, 联合 Foxboro、Yokogawa、ABB、Siemens 等 80 家公司制定的 ISP 协议, 和以 Honeywell 公司为首、联合欧洲等地的 150 家公司制定的 World FIP 协议。迫于用户的压力, 支持 ISP 和 World FIP 的两大集团于 1994 年 9 月握手言和, 成立了 FF。FF 以 ISO/OSI 参考模型为基础, 取其物理层、链路层和应用层为 FF 通信模型的相应层次, 并在此基础上增加了用户层。基金会现场总线分为低速现场总线 and 高速现场总线两种通信速率。低速现场总线 H1 的传输速率为 31.25Kbit/s, 高速现场总线 HSE 的传输速率为 100Mbit/s, H1 支持总线供电和本质安全特性。最大通信距离为 1900m (如果加中继器可延长至 9500m), 最多可直接连接 32 个节点 (非总线供电), 13 个节点 (总线供电), 6 个节点 (本质安全要求)。如果加中继器最多可连接 240 个节点。通信介质为双绞线、光缆或无线电。

FF 采用可变长帧结构, 每帧的有效字节数为 0~251 个。目前已经有 Smar、Fuji、National、Semiconductor、Siemens、Yokogawa 等 12 家公司可以提供 FF 的通信芯片。

目前, 全世界已有 120 多个用户和制造商成为现场总线基金会的成员。基金会董事会囊括了世界上最主要的自动化设备供应商。基金会成员所生产的自动化设备占世界市场的 90% 以上。基金会强调中立与公正。所有的成员均可以参加规范的制订和评估, 所有的技术成果由基金会拥有和控制。由中立的第三方负责产品的注册和测试等。因此, 基金会现场总线具有一定的权威性、广泛性和公正性。

思 考 题

1. 画图简要说明 DCS 的四层体系结构及其功能作用。
2. 简要说明分散系统中过程控制站、运行员操作站、工程师工作站的主要功能。
3. DCS 有哪些主要特点?
4. 画图简要说明 FCS 的几种典型结构形式。
5. FCS 有哪些主要特点?
6. 目前较为流行的现场总线有哪些?

第 8 章 常规数字控制器的设计

一个自动控制系统的核心是控制器。控制器的任务是按照一定的控制规律，产生满足工艺要求的控制信号，以输出驱动执行器，达到自动控制的目的。在传统的模拟控制系统中，控制器的控制规律或控制作用是由仪表或电子装置的硬件电路完成的，而在微机控制系统中，除了计算机装置以外，更主要的体现在软件算法上，即数字控制器的设计上。

对于单回路的常规控制系统，数字控制器的设计主要有连续化设计和直接离散化设计两大类设计方法。

8.1 数字控制器连续化设计

数字控制器的连续化设计是把整个控制系统看成是模拟系统，利用模拟系统的理论和方法进行分析和设计，得到模拟控制器后，再通过某种近似，将模拟控制器离散化为数字控制器，并由计算机来实现。这对于对象特性不太清楚，采样周期比较小的场合，可以获得满意的控制效果。由于广大工程技术人员对 S 平面比对 Z 平面更为熟悉，因此这种方法在工程技术领域得到广泛的应用。

8.1.1 数字控制器的连续化设计步骤

在如图 8-1 所示的单回路计算机控制系统中， $G(s)$ 是被控对象的传递函数， $H_0(s)$ 是零阶保持器， $D(z)$ 是数字控制器。现在的设计问题是如何根据被控对象 $G(s)$ ，设计出满足系统性能指标要求的数字控制器 $D(z)$ ，其设计的步骤主要包括下面几个方面。

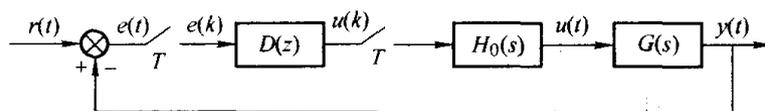


图 8-1 计算机控制系统的结构

1. 设计假想的连续控制器 $D(s)$

设计控制器 $D(s)$ ，一种方法是事先确定控制器的结构，如后面将要重点介绍的 PID 算法等，然后通过其中控制参数的整定完成设计；另一种设计方法是应用连续控制系统的设计方法，直接设计出控制器的结构和参数，如频率特性法、根轨迹法等。关于连续控制系统设计的各种方法，读者可参考有关自动控制原理方面的文献，这里就不再讨论。

无论采用哪种设计方式，设计时都需要知道广义被控对象，如图 8-1 所示，广义被控对象应当是包含零阶保持器的传递函数 $H_0(s)G(s)$ ，但是由于零阶保持器的传递函数 $H_0(s) = (1 - e^{-Ts})/s$ 比较特殊，这对某些连续控制系统设计方法来说，直接处理起来比较困难，为此常采用如下两种解决办法。

一种方法是忽略控制回路中零阶保持器对控制系统的影响，直接按 $G(s)$ 进行设计。大家

知道, 零阶保持器在控制回路中将对控制信号产生附加的相移, 大约相当于半个采样周期的时间滞后环节, 这样会减小所设计系统的相角裕量。为了将这种不利影响控制在较小范围, 在选择采样周期时, 一般应满足 $T \leq (0.15 \sim 0.5) / \omega_c$, 其中 ω_c 是连续控制系统的穿越频率, 即采样周期必须足够小。

另一种方法是所谓的 ω 变换法, 即先利用零阶保持器法将 $G(s)$ 离散化为 $G(z) = (1 - z^{-1})Z \left[\frac{G(s)}{s} \right]$, 再利用 ω 变换将 $G(z)$ 变换为 $G(\omega) = G(z) \Big|_{z = \frac{1 + \omega T/2}{1 - \omega T/2}}$, 然后利用 $G(\omega)$ 进行连续系统的设计。采用这种迂回的办法, 除了考虑了零阶保持器外, 得到的 $G(\omega)$ 为 ω 的有理函数, 同时还解决了系统有纯时间滞后环节给设计带来的不便。应当说明的是, ω 变换是一种近似等效的关系, 存在着所谓频率扭曲现象, 在采用对数频率特性法设计时, 有时需要进行频率校正, 特别是采样周期较大时。

2. 将 $D(s)$ 离散化为 $D(z)$

将 $D(s)$ 离散化, 得到其离散形式 $D(z)$, 可采用前面在第 2 章中介绍的各种近似的离散化方法, 如双线性变换法、零阶保持器法、向后差分法、零/极点匹配法等。

3. 设计由计算机实现的控制算法

设数字控制器 $D(z)$ 的一般形式为

$$D(z) = \frac{U(z)}{E(z)} = \frac{b_0 + b_1 z^{-1} + \dots + b_m z^{-m}}{1 + a_1 z^{-1} + \dots + a_n z^{-n}} \quad (8-1)$$

式中, $n \geq m$, 各系数 a_i 、 b_i 为实数。

式 (8-1) 可写为

$$U(z) = (-a_1 z^{-1} - a_2 z^{-2} - \dots - a_n z^{-n})U(z) + (b_0 + b_1 z^{-1} + \dots + b_m z^{-m})E(z) \quad (8-2)$$

式 (8-2) 用时域可表示为

$$u(k) = -a_1 u(k-1) - a_2 u(k-2) - \dots - a_n u(k-n) + b_0 e(k) + b_1 e(k-1) + \dots + b_m e(k-m) \quad (8-3)$$

利用式 (8-3) 即可由计算机编程实现, 式 (8-3) 也称为数字控制器 $D(z)$ 的控制算法。

4. 校验

控制器 $D(z)$ 设计完并得到控制算法后, 须按图 8-1 所示的计算机控制系统检验其闭环性能是否符合设计要求, 这一步可由计算机控制系统的数字仿真来验证。如果满足设计要求则设计结束, 否则应重新修改设计。

8.1.2 数字 PID 控制器的设计

PID 控制是连续系统中技术最成熟、应用最为广泛的一种控制方式, 有资料表明, 目前 90% 以上的工业控制回路均采用各种形式的 PID 控制算法。PID 是 Proportional (比例)、Integral (积分)、Differential (微分) 三者的缩写。PID 控制的实质是根据反馈后计算得到的输入偏差值, 按比例、积分、微分的函数关系进行运算, 其运算结果用以输出控制。

PID 控制算法主要具有以下优点。

1) 算法蕴涵了动态控制过程中过去、现在和将来的主要信息, 而且其配置达到最优。其

中，比例（P）代表了当前的信息，起纠正偏差的作用，使过程反应迅速；微分（D）在信号变化时有超前控制作用，代表了将来的信息。在过程开始时强迫过程进行，过程结束时减小超调，克服振荡，提高系统的稳定性，加快系统的过渡过程；积分（I）代表了过去积累的信息，它能消除静差，改善系统静态特性。此三种作用配合得当，可使动态过程快速、平稳、准确，收到良好的效果。

2) PID 控制适应性好，有较强的鲁棒性。对各种工业应用场合都可在不同的程度上应用。

3) 算法简单明了，形成了完整的设计和参数调整方法，很容易为工程技术人员所掌握。

下面从最基本的模拟 PID 控制原理出发，讨论数字 PID 控制的计算机实现方法。

1. 理想微分 PID 控制算法

在连续控制系统中，经常采用的一种 PID 控制算法，其控制规律可表示为

$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt} \right] \quad (8-4)$$

或写成传递函数形式

$$G(s) = \frac{U(s)}{E(s)} = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) \quad (8-5)$$

式中， K_p 为比例增益或比例系数； T_i 为积分时间； T_d 为微分时间。

所谓理想的数字 PID 就是上述连续系统 PID 控制规律的离散化数字实现。为了用计算机软件程序实现 PID 控制规律，当采样周期足够小时，可以用求和代替积分、用向后差分代替微分，将微分方程式（8-4）离散化为差分方程，即作如下近似

$$\int_0^t e(t) dt \approx T \sum_{i=1}^k e(i) \quad (8-6)$$

$$\frac{de(t)}{dt} \approx \frac{e(k) - e(k-1)}{T} \quad (8-7)$$

式中， T 为采样周期； k 为采样序号。

由式（8-4）、式（8-6）、式（8-7）可得如下数字 PID 控制算式：

$$u(k) = K_p \left[e(k) + \frac{T}{T_i} \sum_{i=1}^k e(i) + T_d \frac{e(k) - e(k-1)}{T} \right] \quad (8-8)$$

或

$$u(k) = K_p e(k) + K_i \sum_{i=1}^k e(i) + K_d [e(k) - e(k-1)] \quad (8-9)$$

式中， K_p 为比例系数； K_i 称为积分系数； K_d 称为微分系数。

在式（8-8）和式（8-9）中的输出 $u(k)$ 同调节阀的开度（位置）是一一对应的，因此也称为理想 PID 的位置型算式。

由式（8-8）或式（8-9）可见，位置型算式使用不够方便，这是因为要累加偏差 $e(i)$ ，不仅要占用较多的存储单元，而且不便于编写程序，最好能转换为某种递推的形式。为此，同样可以采用向后差分法直接对式（8-5）进行离散化。

$$\begin{aligned}
G(z) &= \frac{U(z)}{E(z)} \approx G(s) \Big|_{s=(1-z^{-1})/T} \\
&= K_p \left[1 + \frac{T}{T_i(1-z^{-1})} + \frac{T_d(1-z^{-1})}{T} \right] \\
&= \frac{1}{1-z^{-1}} K_p \left[(1-z^{-1}) + \frac{T}{T_i} + \frac{T_d}{T}(1-z^{-1})^2 \right] \\
&= \frac{1}{1-z^{-1}} K_p \left[\left(1 + \frac{T}{T_i} + \frac{T_d}{T}\right) - \left(1 + 2\frac{T_d}{T}\right)z^{-1} + \frac{T_d}{T}z^{-2} \right] \quad (8-10)
\end{aligned}$$

也即（注意在推导过程中保留分母的 $(1-z^{-1})$ 因子）

$$(1-z^{-1})U(z) = K_p \left[\left(1 + \frac{T}{T_i} + \frac{T_d}{T}\right) - \left(1 + 2\frac{T_d}{T}\right)z^{-1} + \frac{T_d}{T}z^{-2} \right] E(z) \quad (8-11)$$

写成差分方程的形式为

$$\Delta u(k) = u(k) - u(k-1) = q_0 e(k) + q_1 e(k-1) + q_2 e(k-2) \quad (8-12)$$

$$\text{其中, } \begin{cases} q_0 = K_p \left(1 + \frac{T}{T_i} + \frac{T_d}{T}\right) \\ q_1 = -K_p \left(1 + \frac{2T_d}{T}\right) \\ q_2 = K_p \frac{T_d}{T} \end{cases}$$

式(8-12)称为理想数字PID的增量型算式,这是一种递推的算式,避免了位置型算式中累加求和的麻烦,仅需记录三个历史数据,即 $u(k-1)$ 、 $e(k-1)$ 和 $e(k-2)$,占用存储空间小,运算速度快,且可以通过对增量的限幅在一定程度上避免控制输出大的冲击或计算错误。

2. 实际微分PID控制算法

理想微分PID控制的实际控制效果并不理想,特别在干扰作用下执行机构常动作频繁,计算的控制输出甚至会超过执行机构的上、下限。但是,使用模拟控制器却没有上述现象,其中主要是因为模拟电路本身特性的限制,模拟控制器无法实现理想的微分控制项(即 $T_d s$),而是用一种称为实际微分来近似实现。为了保持模拟控制器良好的控制效果,在数字PID控制中也常采用实际微分的算法,其实质是模仿模拟控制器在理想算法的基础上增加一个对微分作用的低通滤波环节(通常为一阶惯性环节)。

式(8-13)为一种常见的实际微分PID控制算式,即

$$G(s) = \frac{U(s)}{E(s)} = K_p \frac{1}{1+T_f s} \left(1 + \frac{1}{T_i s} + T_d s \right) \quad (8-13)$$

式中, T_f 为惯性时间。

这里仍采用向后差分法对式(8-13)进行离散化,即

$$G(z) = \frac{U(z)}{E(z)} \approx G(s) \Big|_{s=(1-z^{-1})/T}$$

$$\begin{aligned}
 &= K_p \frac{1}{1+T_f(1-z^{-1})/T} \left[1 + \frac{T}{T_i(1-z^{-1})} + \frac{T_d(1-z^{-1})}{T} \right] \\
 &= \frac{1}{(1-z^{-1})} \frac{K_p T}{(T+T_f)-T_f z^{-1}} \left[\left(1 + \frac{T}{T_i} + \frac{T_d}{T}\right) - \left(1 + 2\frac{T_d}{T}\right)z^{-1} + \frac{T_d}{T}z^{-2} \right]
 \end{aligned}$$

即

$$\left[(T_f + T) - T_f z^{-1} \right] \Delta U(z) = K_p T \left[\left(1 + \frac{T}{T_i} + \frac{T_d}{T}\right) - \left(1 + 2\frac{T_d}{T}\right)z^{-1} + \frac{T_d}{T}z^{-2} \right] E(z)$$

写成差分方程的形式为

$$\begin{cases} \Delta u(k) = C_1 \Delta u(k-1) + C_2 e(k) + C_3 e(k-1) + C_4 e(k-2) \\ u(k) = u(k-1) + \Delta u(k) \end{cases} \quad (8-14)$$

其中 $C_1 = \frac{T_f}{T+T_f}$, $C_2 = \frac{K_p T}{T+T_f} \left(1 + \frac{T}{T_i} + \frac{T_d}{T}\right)$,

$$C_3 = -\frac{K_p T}{T+T_f} \left(1 + 2\frac{T_d}{T}\right), \quad C_4 = \frac{K_p T_d}{T+T_f}$$

实际微分 PID 算法可以获得比理想微分 PID 算法更好的控制效果,可以通过对比如图 8-2 所示两种算法的阶跃响应进行说明。

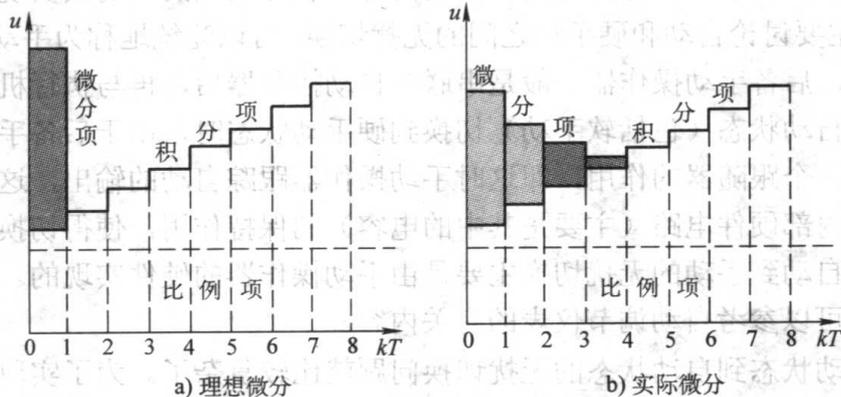


图 8-2 PID 阶跃响应对比

由图 8-2 可知:

1) 理想微分 PID 算法的微分作用仅局限于一个采样周期有一个大幅度的输出,在实际使用时这会产生两方面的问题。一是控制输出可能超过执行机构或 D/A 转换的上、下限;二是执行机构的响应速度可能跟不上,无法在短时间内跟踪这种较大的微分输出。这样在大的干扰作用情况下,一方面会使算法中的微分不能充分发挥作用,另一方面也会对执行机构产生一个大的冲击作用。相反地,实际微分 PID 算法由于惯性滤波的存在,使微分作用可持续多个采样周期,有效地避免了上述问题的产生,因而具有更好的控制性能。

2) 由于微分对高频信号具有放大作用,采用理想微分容易在系统中引入高频的干扰,引起执行机构的频繁动作,降低机构的使用寿命。而实际微分 PID 算法中包含有一阶惯性环节,具有低通滤波的特性,抗干扰能力较强。

由于上面分析的这些原因,在实际的工业控制应用中,特别是在微分作用较强时,数字

PID 控制大多采用实际微分算法。

实际微分 PID 算法除了式 (8-13) 所示的形式外, 还具有其他多种形式, 式 (8-15) 和式 (8-16) 是其中两种以传递函数表示的控制算式, 感兴趣的读者可以用向后差分或双线性变换法进行离散化。

$$G(s) = \frac{U(s)}{E(s)} = K_p \frac{1 + T_d s}{1 + \frac{T_d}{K_d} s} \left(1 + \frac{1}{T_i s} \right) \quad (8-15)$$

$$G(s) = \frac{U(s)}{E(s)} = K_p \left(1 + \frac{1}{T_i s} + \frac{T_d s}{1 + \frac{T_d}{K_d} s} \right) \quad (8-16)$$

式中, K_d 为微分增益。

3. 手动/自动跟踪与无扰切换

在自动控制系统中, 控制作用主要包含手动和自动两类状态, 而手动又可分为由计算机实现的所谓“软手动”和由后备手动操作器实现的所谓“硬手动”。上述三种控制状态之间常需要进行切换, 并且要求切换过程中控制输出不会产生扰动, 称为无扰切换, 这种无扰切换是通过状态间的相互跟踪实现的。

由于自动状态和软手动状态都是由计算机内部的软件实现的, 切换实现比较简单, 为了简化问题, 这里主要讨论自动和硬手动之间的无扰切换, 可以笼统地称为手动/自动无扰切换。

控制系统中, 后备手动操作器一般是串联在自动控制器后, 再与执行机构相连的。系统从计算机实现的自动状态 (包括软手动) 切换到硬手动状态时, 由于后备手动操作器在自动状态时实际上起一个跟随器的作用, 即这时手动操作器跟踪自动的输出, 这样在切换的瞬间由于手动操作器内部硬件电路 (主要是其中的电容) 的保持作用, 使得切换没有扰动产生。可以简单地说, 自动到手动的无扰切换主要是由手动操作器的硬件实现的。有关手动操作器的硬件结构读者可以参考自动调节仪表的有关内容。

相反地, 手动状态到自动状态的无扰切换问题就比较复杂了。为了实现这种无扰切换, 计算机需要采样获得手动操作器或执行机构输出的所谓阀位值。但仅仅这样是不够的, 因为这只是使 $u(k-1)$ 能够跟踪阀位, 这里同时还需要对控制算法的软件进行处理。由式 (8-12) 和式 (8-14) 可知, 要使手动到自动实现无扰切换就是要使 $\Delta u(k) = 0$, 这样在手动状态下 (包括软手动和硬手动), 就必须使算法中 $e(k-1)$ 、 $e(k-2)$ 和 $\Delta u(k-1)$ 等历史状态清零, 同时使切换时偏差 $e(k)$ 也为零。为使 $e(k) = 0$, 就要求手动状态时的设定值跟踪过程的反馈量 (即所谓 SP 跟踪 PV)。这样处理后, 就可以保证 $\Delta u(k) = 0$ 。但是, 采用上述方式实现无扰切换有一个问题, 就是投入自动后还需要由运行人员调整设定值, 有些不太方便。另一种方式是 SP 不跟踪 PV, 这时不能保证 $e(k) = 0$, 投入自动时必须将偏差控制在较小的范围内, 以保证切换过程中扰动很小, 这样的好处是可以不再需要调整设定值。总之, 手动到自动的无扰切换主要是通过计算机的软件实现的。

8.1.3 数字 PID 算法的改进

用计算机实现 PID 控制, 不只是简单地把 PID 控制规律数字化, 而是进一步与计算机的

强大运算能力、存储能力和逻辑判断能力结合起来,使PID控制更加灵活多样,更能够满足对控制系统提出的各种要求。下面介绍的数字PID算法的改进主要包括积分项的改进、微分项的改进和带死区的PID等几方面。

1. 积分分离算法

一般的PID控制算法中,当有较大的扰动或大幅度改变设定值时,由于短时间内出现大的偏差,加上系统本身具有的惯性和滞后,在积分的作用下,将引起系统过量的超调和长时间的波动。特别是对于温度、成分等大惯性、大滞后的系统,这一现象更为严重。考虑到积分的主要作用是消除系统的稳态偏差,在偏差较大的过程中,积分的作用并不明显,为此可以通过下面的积分分离措施来改变这一情况。

积分分离措施可以采用如下一种简便处理办法,即当偏差较大时取消积分作用,当偏差较小时才投入积分作用,也即

当 $|e(k)| > \beta$ 时,采用PD控制;

当 $|e(k)| \leq \beta$ 时,采用PID控制。

积分分离值 β 应根据具体对象及要求确定。若 β 值过大,达不到积分分离的目的,如图8-3中的曲线a所示;若 β 值过小,一旦被控量 y 脱离了积分分离区,只进行PD控制,有可能无法消除残差,如图8-3中的曲线c所示。

积分分离除了采用上述简单的积分“开关”控制外,还可以采用所谓变速积分的算法。变速积分的基本思想是设法改变积分增益的大小,使其与输入偏差的大小相对应。偏差越大,积分作用越弱;反之则越强。下面介绍一种变速积分的算法,如图8-4所示。

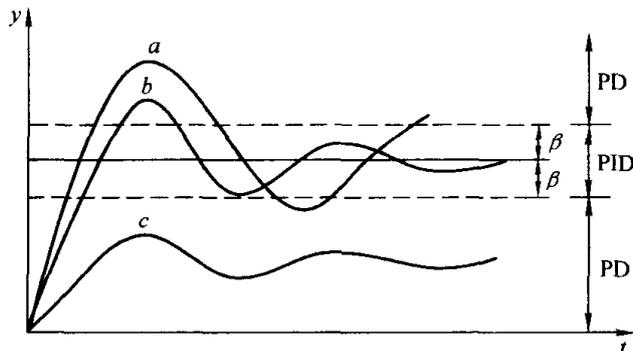


图 8-3 不同积分分离值下的系统响应曲线

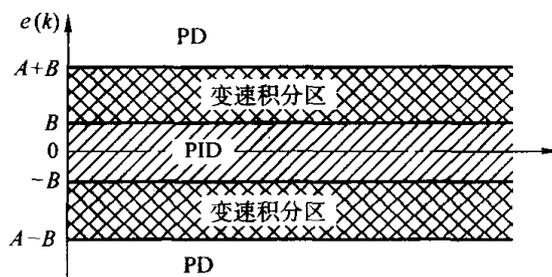


图 8-4 变速积分系数与偏差大小的关系

这里

$$\Delta u_i(k) = K_i f[e(k)]e(k) \quad (8-17)$$

f 与偏差当前值 $|e(k)|$ 的关系为

$$f[e(k)] = \begin{cases} 1 & |e(k)| \leq B \\ \frac{A - |e(k)| + B}{A} & B < |e(k)| \leq A + B \\ 0 & |e(k)| > A + B \end{cases} \quad (8-18)$$

$f[e(k)]$ 在 $0 \sim 1$ 区间内变化,当偏差大于所给分离区间 $A+B$ 后, $f[e(k)]=0$,即积分项不再继续累加当前值 $e(k)$;当偏差小于 B 时,加当前值 $e(k)$,即积分项与普通PID积分项相同,积分动作达到最高速;而当偏差在 B 与 $A+B$ 之间时,则累加进的是部分当前值,其值在 $0 \sim e(k)$

之间且随着 $e(k)$ 的大小而反向变化。

与普通 PID 算法相比,积分分离算法的优点是可以减小系统的超调量,容易使系统稳定,提高了控制系统的品质。

2. 抗积分饱和算法

虽然 PID 控制系统是作为线性系统来分析处理的,但在某些情况下往往存在不可避免的非线性因素,如所有的执行机构、阀门以及 D/A 转换输出都有限幅,具有上、下限的限制。控制系统在运行过程中,控制量输出是一个动态过程(不是与当前的被控量一一对应的),有时会不可避免地使控制输出达到系统的限幅值。这时的执行器将保持在极限位置而与过程变量无关,相当于控制系统处于开环状态。此时,若控制器具有积分作用,输入偏差的存在可能导致持续积分,积分项可能会进一步使 PID 计算的输出超出系统的限幅值。当偏差反向时,系统需要很长的时间才能使积分作用返回有效的正常值。这一现象称为积分饱和,积分饱和现象会使控制系统的品质变差。

从上面积分饱和现象的分析,很容易得到一种简单的抗积分饱和的办法,即当出现积分饱和时,通过停止积分作用的方法来抑制积分的饱和。具体的办法是,当控制输出达到系统的上、下限限幅值时,停止对某一方向的积分。设控制器输出满足 $u_{\min} \leq u(k) \leq u_{\max}$,其中 u_{\max} 和 u_{\min} 分别为控制量容许的上、下限值,当 $u(k)$ 超出此范围时,采取停止积分的措施。以采用正作用的 PID 控制为例,若 $u(k) \geq u_{\max}$,且 $e(k) > 0$,则令积分增益 $K_i = 0$ 停止积分,防止计算控制量 u 的继续增加;类似地,若 $u(k) \leq u_{\min}$,且 $e(k) < 0$,同样积分增益 $K_i = 0$ 停止积分,防止计算控制量 u 的继续减小。当然,在要求不高时,也可以不考虑偏差 $e(k)$ 的方向,只要达到控制量容许的上、下限值,就停止积分。

这里要特别注意,是否采取抗积分饱和措施的关键是判断控制系统最终的控制输出是否超出了系统要求的限幅值。在串级控制系统中积分饱和现象有时非常严重,这时控制最后的输出是副调节器的输出,当它已达到了执行机构容许的上、下限值时,不仅副调节器要采取抗积分饱和措施,更重要的是主调节器要采取抗积分饱和算法。例如,在火电厂主蒸汽温度的串级控制中,一般主调节器就必须采取抗积分饱和的算法,这里的主调节器是否采取抗积分饱和措施与其本身的输出无关。

从形式上看,尽管积分分离算法和抗积分饱和算法都是通过停止积分作用实现的,但它们判断停止积分的条件完全不同。积分分离算法进行分离的依据是 PID 控制器的输入偏差 e ,而抗积分饱和算法的抗积分饱和依据是抗积分饱和算法系统最终的控制输出量 u 。如果用一句通俗简单的话来总结积分分离算法和抗积分饱和算法的特点,就是“大偏差时不积分(积分分离),输出超限时也不积分(抗饱和)”。

3. 微分项的改进

数字 PID 控制算法对微分项的改进,其中最重要就是前面介绍过的实际微分 PID 算法,该算法不仅可以抑制干扰,还可以克服理想微分算法对系统有大幅度冲击的缺点。

从抗干扰的观点出发,为了抑制微分作用容易引入干扰的不足,除了可以像实际微分算法一样在 PID 算法的输出侧对控制输出进行惯性滤波外,同样也可以通过对 PID 控制的输入偏差 e 进行低通滤波处理,以抑制高频的过程干扰对系统的不利影响,如采用数字滤波算法中均值滤波、去极值滤波、限幅限速滤波等,其中采用四点中值差分法处理输入偏差 e 就是其中一种典型的方法,感兴趣的读者可以参考有关文献,这里不再赘述。

为了减小设定值变化时，数字 PID 控制算法中微分作用对系统的冲击作用，避免微分动作导致控制量 u 的大幅度变化，工业控制过程中还常采用一种称为微分先行的控制算法。该算法的基本原理很简单，就是只对过程的被控量 y 进行微分，不对设定值 r 进行微分，这样可以使设定值变化时控制作用的变化较为平缓。

需要注意的是，对串级控制的副回路而言，因副调节器的给定值是主调节器提供的，故上述仅对测量值微分的做法并不适用。

4. 带死区的数字 PID 算法

在许多实际的控制系统中，并不要求被控量十分精确地与设定值相等，完全无差，而是容许偏差在一定范围内变化。在这种情况下，计算机控制中为了避免控制动作过于频繁，以消除由于执行机构或阀门的频繁动作所引起的系统振荡，有时采用所谓带死区的 PID 算法，也称带不灵敏区的算法。如图 8-5 所示，该算法是在原 PID 算法的前面增加一个不灵敏区的非线性环节来实现的，即

$$p(k) = \begin{cases} e(k) & |e(k)| > B \\ 0 & |e(k)| \leq B \end{cases} \quad (8-19)$$

在图 8-5 中，死区 B 是一个可调参数，其大小一般应根据控制系统对被控量稳态精度的要求以及现场的试验结果来确定。

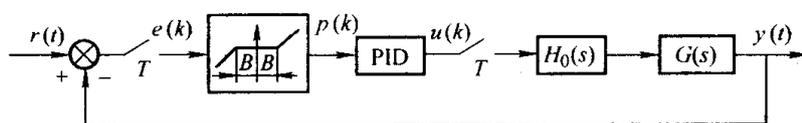


图 8-5 带死区的计算机 PID 控制系统

需要指出的是，死区是一个非线性环节，不能像线性环节一样随便移到 PID 控制器的后面，对控制量输出设定一个死区，这样做的效果是完全不同的。在生产现场有时为了延长执行机构或阀门使用寿命，有一种错误的做法，即不按设计规范的要求片面地增大执行机构或阀门的回程误差（简称回差），希望能避免执行机构或阀门频繁动作，这就相当于将死区移到了 PID 控制器后面，这样有时会得到适得其反的效果。

图 8-6 所示是带有回程误差阀门的控制系统可能出现的响应曲线。在正常情况下，控制输出不会恰好使被控量与设定值相等，这时在偏差的存在情况下，系统如果具有较强的积分作用，偏差的连续积分会使控制器产生一个超过回差的有效输出变化 $\Delta u(k)$ ，在 $\Delta u(k)$ 的作用下被控量完全有可能又产生一个反向的偏差，这一反向偏差的积分会使控制器产生一个有效的反向输出变化 $-\Delta u(k)$ ，再次使被控量产生正向偏差，这样不断反复，实际有效的控制输出会形成一串不断变化的矩形波，这样不仅不能避免控制输出的频繁变

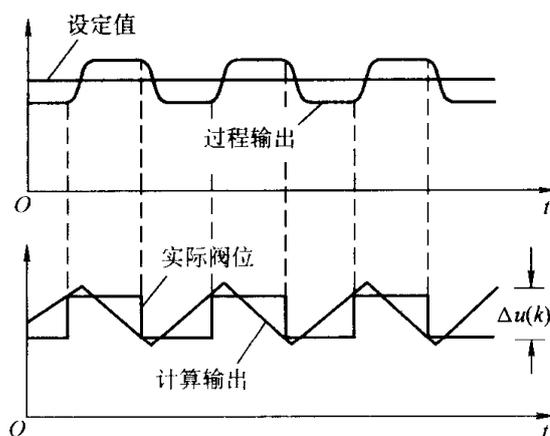


图 8-6 具有回差的控制系统可能出现的响应曲线

化,相反可能使动作更频繁了。使用在 PID 前带有死区的算法,在一定程度上可以抑制如图 8-6 所示的由于执行机构存在回差而可能产生的控制量来回振荡现象。

8.1.4 数字 PID 参数的整定

对由数字 PID 构成的控制系统的设计,由于控制器的结构和控制规律已经确定,系统控制质量的好坏主要取决于参数是否合理,设计确定 PID 参数的工作也称为 PID 的参数整定。

一般的生产过程都具有较大的时间常数,而数字 PID 控制系统的采样周期则要小得多,所以数字 PID 的参数整定,完全可以按照模拟调节器的各种参数整定方法进行分析 and 综合。整定 PID 控制参数的方法很多,但可归纳为理论整定法与工程整定法两大类。

理论整定法以被控对象的数学模型为基础,通过理论计算(如根轨迹、频率特性等)直接求得控制器参数。理论整定需要知道被控对象的精确数学模型,否则整定后的控制系统难以达到预期的效果。而实际问题的数学模型往往都是一定条件下的近似,所以这种方法主要用于理论分析,在工程上用的并不是很多。

工程实际中应用最多的整定方法是工程整定法,它实际上是一种近似的经验方法。由于其方法简单,便于实现,特别是不必依赖控制对象的数学模型,且能解决控制工程中的实际问题,因而被工程技术人员广泛采用。下面将主要介绍在数字 PID 整定中应用较多的扩充临界比例带法与扩充响应曲线法,同时对数字 PID 整定的参数仿真寻优方法也进行简单介绍。

考虑到计算机控制系统还一般具有零阶保持器,当采样周期较大时,整定有时需要考虑保持器相角滞后的影响,相当于在被控对象的纯迟延时间基础上增加了 $1/2$ 采样周期。

数字控制器与模拟控制器相比,除了需要整定 PID 参数,即比例系数 K_p 、积分时间 T_i 和微分时间 T_d 外,还有一个重要参数——采样周期 T ,相关内容参见本书第 2 章。

1. 扩充临界比例带法

扩充临界比例带法是模拟调节器中使用的临界比例带法(也称稳定边界法)的扩充,是一种闭环整定的实验经验方法。按该方法整定 PID 参数的步骤如下。

1) 选择一个足够短的采样周期 T_{\min} 。所谓足够短,具体地说就是采样周期选择为对象的纯滞后时间的 $1/10$ 以下。

2) 将数字 PID 控制器设定为纯比例控制,并逐步减小比例带 δ ($\delta = 1/K_p$),使闭环系统产生临界振荡。此时的比例带和振荡周期称为临界比例带 δ_k 和临界振荡周期 T_k 。

3) 选定控制度。所谓控制度,就是以模拟调节器为基准,将 DDC 的控制效果与模拟调节器的控制效果相比较。控制效果的评价函数通常采用 $\int_0^{\infty} e^2(t)dt$ (最小的误差平方积分)表示。

$$\text{控制度} = \frac{\left[\int_0^{\infty} e^2(t)dt \right]_D}{\left[\int_0^{\infty} e^2(t)dt \right]_A} \quad (8-20)$$

实际应用中并不需要计算出两个误差的平方积分,控制度仅表示控制效果的物理概念。例如,当控制度为 1.05 时,就是指 DDC 控制与模拟控制效果基本相同;控制度为 2.0 时,是指 DDC 控制比模拟控制效果差。

4) 根据选定的控制度查表 8-1, 求得 T 、 K_p 、 T_i 、 T_d 的值。

表 8-1 按扩充临界比例带法整定 T 、 K_p 、 T_i 、 T_d

控制度	控制规律	T	K_p	T_i	T_d
1.05	PI	$0.03T_k$	$0.53\delta_k$	$0.88T_k$	—
	PID	$0.014T_k$	$0.63\delta_k$	$0.49T_k$	$0.14T_k$
1.2	PI	$0.05T_k$	$0.49\delta_k$	$0.91T_k$	—
	PID	$0.043T_k$	$0.47\delta_k$	$0.47T_k$	$0.16T_k$
1.5	PI	$0.14T_k$	$0.42\delta_k$	$0.99T_k$	—
	PID	$0.09T_k$	$0.34\delta_k$	$0.43T_k$	$0.20T_k$
2.0	PI	$0.22T_k$	$0.36\delta_k$	$1.05T_k$	—
	PID	$0.16T_k$	$0.27\delta_k$	$0.40T_k$	$0.22T_k$
模拟调节器	PI	—	$0.57\delta_k$	$0.83T_k$	—
	PID	—	$0.70\delta_k$	$0.50T_k$	$0.13T_k$
Ziegler-Nichols 整定法	PI	—	$0.45\delta_k$	$0.83T_k$	—
	PID	—	$0.60\delta_k$	$0.50T_k$	$0.125T_k$

5) 按求得的整定参数投入运行, 在投运中观察控制效果, 再适当调整参数, 直到获得满意的控制效果。

2. 扩充响应曲线法

与上述闭环整定方法不同, 扩充响应曲线法是一种开环整定方法。如果可以得到被控对象的动态特性曲线, 那么就可以与模拟调节系统的整定一样, 采用扩充响应曲线法进行数字 PID 的整定。其步骤如下。

1) 断开数字控制器, 使系统在手动状态下工作。将被控量调节到给定值附近, 当达到平衡时, 给对象一个阶跃输入信号。

2) 记录被调量在此阶跃作用下的变化过程曲线 (即广义对象的飞升特性曲线), 如图 8-7 所示。

3) 根据飞升特性曲线, 求得被控对象纯滞后时间 τ 和等效惯性时间常数 T_p , 以及它们的比值 T_p/τ 。

4) 由求得的 T_p 和 τ 以及它们的比 T_p/τ , 选择某一控制度, 查表 8-2, 即可求得数字 PID 的整定参数 T 、 K_p 、 T_i 、 T_d 的值。

5) 按求得的整定参数投入运行, 在投运中观察控制效果, 再适当调整参数, 直到获得满意的控制效果。

3. 仿真寻优法

如果能像扩充响应曲线法一样得到被控对象的动态特性曲线, 就可以利用计算机系统的强大计算能力, 通过辨识、仿真及寻优等过程, 整定获得一定意义下最优的 PID 参数。其主

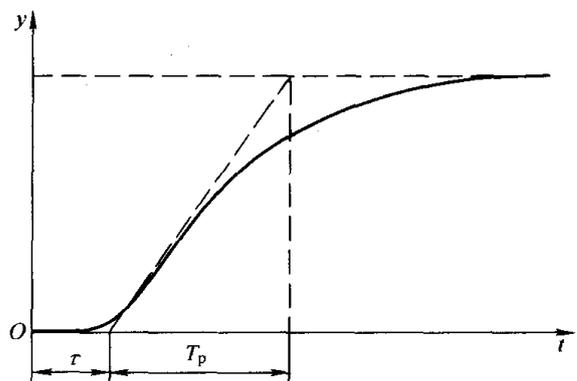


图 8-7 广义对象的阶跃飞升特性曲线

要步骤如下。

表 8-2 按扩充响应曲线法整定 T 、 K_p 、 T_i 、 T_d

控制度	控制规律	T	K_p	T_i	T_d
1.05	PI	0.1τ	$0.84T_p/\tau$	3.4τ	—
	PID	0.05τ	$1.15T_p/\tau$	2.0τ	0.45τ
1.2	PI	0.2τ	$0.78T_p/\tau$	3.6τ	—
	PID	0.16τ	$1.0T_p/\tau$	1.9τ	0.55τ
1.5	PI	0.5τ	$0.68T_p/\tau$	3.9τ	—
	PID	0.34τ	$0.85T_p/\tau$	1.6τ	0.65τ
2.0	PI	0.8τ	$0.57T_p/\tau$	4.2τ	—
	PID	0.6τ	$0.6T_p/\tau$	1.5τ	0.82τ
模拟调节器	PI	—	$0.9T_p/\tau$	3.3τ	—
	PID	—	$1.2T_p/\tau$	2.0τ	0.4τ
Ziegler-Nichols 整定法	PI	—	$0.9T_p/\tau$	3.3τ	—
	PID	—	$1.2T_p/\tau$	3.0τ	0.5τ

1) 与扩充响应曲线法类似, 通过被控对象的阶跃实验, 获得广义对象的飞升特性曲线。

2) 通过各种适合飞升特性曲线的辨识方法 (如面积法等), 得到被控对象的粗略控制模型, 如对象的静态增益 K_p 、纯滞后时间 τ 和等效惯性时间常数 T_p 等 (即对象模型用

$$G_p(s) = \frac{K_p e^{-\tau}}{1 + T_p s} \text{ 描述}。$$

3) 如需要将控制模型离散化, 与数字 PID 控制器一起, 编程实现一个单回路的计算机仿真系统, 也可以采用 Matlab 等仿真工具。

4) 在上述仿真系统的基础上, 选择某一积分型的性能指标函数, 选用各种合适的优化方法, 如单纯形法、梯度法等, 通过寻优得到在选定性能指标下的优化 PID 参数。常用的积分型性能指标函数主要有

$$\text{ISE} = \int_0^{\infty} e^2(t) dt \quad (8-21)$$

$$\text{IAE} = \int_0^{\infty} |e(t)| dt \quad (8-22)$$

$$\text{ITAE} = \int_0^{\infty} t |e(t)| dt \quad (8-23)$$

$$J = \int_0^{\infty} [e^2(t) + \rho u^2(t)] dt \quad (8-24)$$

式中, $\rho > 0$ 为控制量的加权系数。

最优的整定参数应使这些积分指标最小, 不同的积分指标所对应的系统输出被控量响应曲线稍有差别。一般情况下, ISE 指标的超调量大, 上升时间快; IAE 指标的超调量适中, 上升时间稍快; ITAE 指标的超调量小, 调整时间也短; 加入控制量加权函数的指标, 是为了限制控制量过大, 以减小控制量的频繁波动。

5) 考虑到实际控制系统的复杂性, 适当改变仿真对象的参数, 通过仿真验证控制效果, 使控制系统对模型一定程度的失配具有一定的鲁棒性。

6) 按求得的优化整定参数投入运行, 在投运中观察控制效果, 再适当调整参数, 直到获得满意的控制效果。

8.2 数字控制器离散化设计

随着辨识技术的发展, 某些对象的特性可以精确获得, 这时可以一开始就把系统看成是数字系统, 然后按采样控制理论, 以 z 变换为工具, 以脉冲传递函数为数学模型, 直接设计满足指标要求的数字控制器 $D(z)$, 这称为数字控制器的直接离散化设计法, 或称直接解析设计法。

8.2.1 数字控制器的离散化设计步骤

研究如图 8-8 所示的典型计算机控制系统。

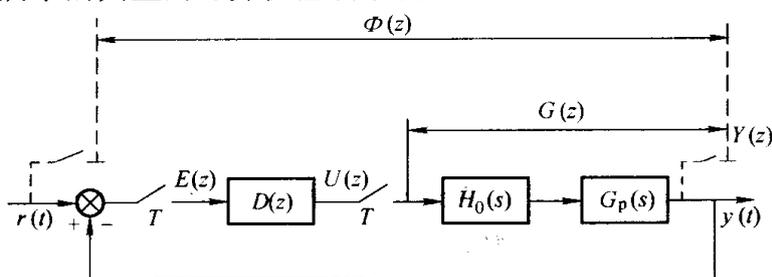


图 8-8 典型的计算机控制系统结构

图中, $G_p(s)$ 为被控对象; $H_0(s) = \frac{1 - e^{-Ts}}{s}$ 为零阶保持器; $G(z)$ 是 $H_0(s)$ 和 $G_p(s)$ 相乘后的等效脉冲传递函数; $D(z)$ 是需要设计的数字控制器。该系统的闭环脉冲传递函数为

$$\Phi(z) = \frac{Y(z)}{R(z)} = \frac{D(z)G(z)}{1 + D(z)G(z)} \quad (8-25)$$

误差脉冲传递函数为

$$\Phi_e(z) = \frac{E(z)}{R(z)} = \frac{1}{1 + D(z)G(z)} = 1 - \Phi(z) \quad (8-26)$$

直接离散化设计的目标就是根据预期的控制指标, 直接设计满足要求的数字控制器 $D(z)$, 而预期的控制指标通常是由理想的闭环脉冲传递函数或误差脉冲传递函数来体现的。由此可得出数字控制器的离散化设计步骤如下:

1) 当设计要求一旦确定, 即根据控制系统的性能指标要求和其他约束条件, 确定所需的闭环脉冲传递函数 $\Phi(z)$ 或误差脉冲传递函数 $\Phi_e(z)$ 。

2) 根据被控对象和零阶保持器的传递函数求出广义对象的脉冲传递函数 $G(z)$ 。

3) 由式 (8-25)、式 (8-26) 可推出数字控制器的脉冲传递函数 $D(z)$

$$D(z) = \frac{\Phi(z)}{G(z)[1 - \Phi(z)]} = \frac{\Phi(z)}{G(z)\Phi_e(z)} \quad (8-27)$$

上述设计的基本思想与人们熟悉的模拟化设计方法有很大不同。模拟化设计方法是根据被控对象, 结合期望的控制性能, 设计出合适的控制器结构及参数, 最后再验证闭环的控制效果是否满足性能指标的要求。而直接离散化设计是先根据期望的控制性能指标, 设计出满足性能指标的闭环脉冲传递函数, 然后再逆推出控制器 $D(z)$ 。

下面将分别介绍两种直接离散化数字控制器的设计方法, 即最少拍控制算法和大林控制算法。

8.2.2 最少拍控制系统的设计

所谓最少拍控制系统是指系统在典型输入信号作用下, 具有最快的响应速度。也就是说, 系统经过最少个采样周期(或节拍), 就能结束瞬态过程, 使稳态偏差为零。由于最少拍控制系统主要用于随动系统, 因此也称之为最少拍随动系统。最少拍控制系统又可分为有纹波最少拍系统和无纹波最少拍系统。

1. 最少拍控制系统 $D(z)$ 的设计

设计最少拍系统的数字控制器 $D(z)$, 最重要的就是要研究如何根据性能要求, 构造一个理想的闭环脉冲传递函数。

由误差表达式

$$E(z) = \Phi_e(z)R(z) = e_0 + e_1z^{-1} + e_2z^{-2} + \dots \quad (8-28)$$

可知, 要实现无静差、最少拍, $E(z)$ 应在最短时间内趋近于零, 即 $E(z)$ 应为有限项多项式。因此, 在输入 $R(z)$ 一定的情况下, 必须对 $\Phi_e(z)$ 提出要求。

最少拍系统典型的输入信号常用的主要有以下几种形式:

$$\textcircled{1} \text{ 单位阶跃输入} \quad r(t) = 1(t) \quad R(z) = \frac{1}{1-z^{-1}} \quad (8-29)$$

$$\textcircled{2} \text{ 单位速度输入} \quad r(t) = t \quad R(z) = \frac{Tz^{-1}}{(1-z^{-1})^2} \quad (8-30)$$

$$\textcircled{3} \text{ 单位加速度输入} \quad r(t) = \frac{1}{2}t^2 \quad R(z) = \frac{T^2z^{-1}(1+z^{-1})}{2(1-z^{-1})^3} \quad (8-31)$$

$$\text{输入信号的一般表达式可表示为} \quad R(z) = \frac{A(z)}{(1-z^{-1})^N} \quad (8-32)$$

将式(8-32)代入误差表达式, 得

$$E(z) = \Phi_e(z)R(z) = \frac{\Phi_e(z)A(z)}{(1-z^{-1})^N} \quad (8-33)$$

要使式(8-33)中 $E(z)$ 为有限项多项式, $\Phi_e(z)$ 应能被 $(1-z^{-1})^N$ 整除, 即 $\Phi_e(z)$ 应取 $(1-z^{-1})^N F(z)$ 的形式。要实现最少拍, $E(z)$ 应尽可能简单, 故取 $F(z) = 1$ 。这样, 经过简单计算可以容易地得到在不同典型输入情况下, $\Phi_e(z)$ 或 $\Phi(z)$ 的表达式, 进而设计出最少拍控制器 $D(z)$, 如表 8-3 所示。从表中还可以看到, 在单位阶跃、单位速度和单位加速度输入的情况下, 最少拍系统分别经过一拍 (T)、二拍 ($2T$) 和三拍 ($3T$) 的调整时间后, 系统偏差就可消失, 且过渡时间最短。

表 8-3 各种典型输入下的最少拍系统

典型输入 $r(t)$	误差脉冲传递函数 $\Phi_e(z)$	闭环脉冲传递函数 $\Phi(z)$	最少拍调节器 $D(z)$	调节时间
$1(t)$	$1-z^{-1}$	z^{-1}	$\frac{z^{-1}}{(1-z^{-1})G(z)}$	T
t	$(1-z^{-1})^2$	$2z^{-1}-z^{-2}$	$\frac{2z^{-1}-z^{-2}}{(1-z^{-1})^2G(z)}$	$2T$
$\frac{1}{2}t^2$	$(1-z^{-1})^3$	$3z^{-1}-3z^{-2}+z^{-3}$	$\frac{3z^{-1}-3z^{-2}+z^{-3}}{(1-z^{-1})^3G(z)}$	$3T$

例 8-1 在如图 8-8 所示系统中, 设被控对象的传递函数 $G_p(s) = \frac{10}{s(s+1)}$, 采样周期 $T = 1\text{s}$, 试在单位速度输入下设计一个最少拍数字控制器 $D(z)$ 。

解: 被控对象与零阶保持器的等效脉冲传递函数为

$$\begin{aligned} G(z) &= (1-z^{-1})\mathcal{Z}\left[\frac{G_p(s)}{s}\right] = (1-z^{-1})\mathcal{Z}\left[\frac{10}{s^2(s+1)}\right] \\ &= 10(1-z^{-1})\mathcal{Z}\left[\frac{1}{s^2} - \frac{1}{s} + \frac{1}{s+1}\right] \\ &= 10(1-z^{-1})\left[\frac{z^{-1}}{(1-z^{-1})^2} - \frac{1}{1-z^{-1}} + \frac{1}{1-e^{-1}z^{-1}}\right] \\ &= \frac{3.68z^{-1}(1+0.718z^{-1})}{(1-z^{-1})(1-0.368z^{-1})} \end{aligned}$$

根据最少拍系统设计的要求, 对单位速度输入应选 $\Phi_e(z) = (1-z^{-1})^2$, 代入式 (8-27) 可得

$$\begin{aligned} D(z) &= \frac{1-\Phi_e(z)}{G(z)\Phi_e(z)} \\ &= \frac{1-(1-z^{-1})^2}{\frac{3.68z^{-1}(1+0.718z^{-1})}{(1-z^{-1})(1-0.368z^{-1})} \cdot (1-z^{-1})^2} \\ &= \frac{0.543(1-0.5z^{-1})(1-0.368z^{-1})}{(1-z^{-1})(1+0.718z^{-1})} \end{aligned}$$

此时输出

$$\begin{aligned} Y(z) &= \Phi(z)R(z) = [1-\Phi_e(z)]R(z) \\ &= (2z^{-1}-z^{-2})\frac{z^{-1}}{(1-z^{-1})^2} = 2z^{-2} + 3z^{-3} + 4z^{-4} + \dots \end{aligned}$$

误差

$$E(z) = \Phi_e(z)R(z) = (1-z^{-1})^2 \cdot \frac{z^{-1}}{(1-z^{-1})^2} = z^{-1}$$

输出和误差变化的波形如图 8-9 所示。从图中可见，系统经过了两个采样周期以后，输出完全跟踪了输入，稳态误差为零。

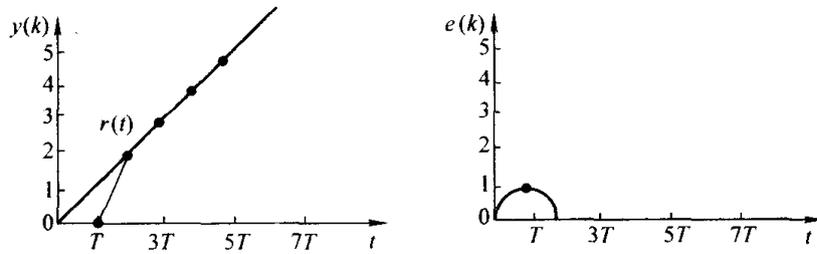


图 8-9 单位速度输入下输出和误差变化波形

该系统是针对单位速度输入设计的最少拍系统，那么这个系统对其他输入是否还能成为最少拍呢？下面对单位阶跃和单位加速度输入下该系统的响应情况进行分析。

在单位阶跃输入时，输出量为

$$Y(z) = \Phi(z)R(z) = (2z^{-1} - z^{-2}) \cdot \frac{1}{1-z^{-1}} = 2z^{-1} + z^{-2} + z^{-3} + \dots$$

即输出序列为 $y(0) = 0$ ， $y(1) = 2$ ， $y(2) = y(3) = \dots = 1$

输出响应如图 8-10 所示，从图中可见，该系统在单位阶跃输入下，经过两个采样周期就稳定在设定值上，但在第一个采样点上有 100% 的超调量。

单位加速度输入时，输出量为

$$Y(z) = \Phi(z)R(z) = (2z^{-1} - z^{-2}) \cdot \frac{z^{-1}(1+z^{-1})}{2(1-z^{-1})^3} = z^{-2} + 3.5z^{-3} + 7z^{-4} + 11.5z^{-5} + \dots$$

即输出序列为 $y(0) = 0$ ， $y(1) = 2$ ， $y(2) = 1$ ， $y(3) = 3.5$ ， $y(4) = 7$ ， $y(5) = 11.5 \dots$ 此时单位加速度输入 $\frac{1}{2}t^2$ 的采样函数 $r(kT) = \frac{1}{2}(kT)^2$ ，输入序列为 $r(0) = 0$ ， $r(1) = 0.5$ ， $r(2) = 2$ ， $r(3) = 4.5$ ， $r(4) = 8$ ， $r(5) = 12.5 \dots$ 。从第二拍开始，输出与输入的误差为 1，跟踪波形如图 8-11 所示。

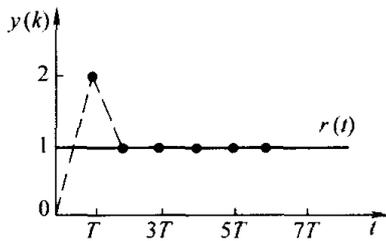


图 8-10 单位阶跃输入下输出响应

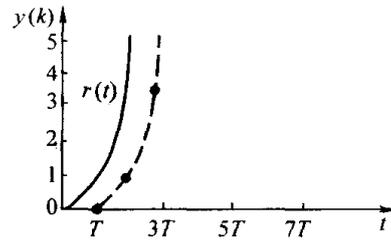


图 8-11 单位加速度输入下输出响应

由上面的分析，进一步可以得出结论：按照某种典型输入设计的最少拍系统，对其他输入函数的响应并不是最少拍系统，可见最少拍系统对输入函数的适应性很差。一般地，按某一种典型输入设计的最少拍系统，用于阶次较低的输入函数时，系统将出现较大的超调，同时响应时间也增加，但是还能保持在采样时刻稳态误差为零。相反地，当用于阶次较高的输入函数时，输出不能完全跟踪输入，存在静差。

2. 最少拍控制器 $D(z)$ 设计的限制条件

前面在讨论最少拍控制器 $D(z)$ 的设计时,并没有考虑控制器 $D(z)$ 物理上是否能实现以及控制器是否稳定的问题,而仅是由式(8-27)简单地求得,显然位于分母的被控对象 $G(z)$ 将影响到控制器 $D(z)$ 的物理可实现性和稳定性。

设被控对象 $G(z)$ 具有如下一般形式

$$G(z) = \frac{Kz^{-r} \prod_{i=1}^m (1 - z_i z^{-1})}{\prod_{i=1}^n (1 - p_i z^{-1})} \quad (8-34)$$

式中, r 为纯滞后; K 为静态增益; z_i 和 p_i 分别为对象的零点和极点; n 和 m 分别为分母和分子的阶次。

$$D(z) = \frac{z^r \Phi(z) \prod_{i=1}^n (1 - p_i z^{-1})}{K \Phi_e(z) \prod_{i=1}^m (1 - z_i z^{-1})} \quad (8-35)$$

由式(8-35)可见,若 $D(z)$ 中存在 z^r 环节,则表示数字调节器具有超前特性,即在环节施加输入信号之前的 r 个采样周期就有输出,这样的超前环节在物理上是不可能实现的。所以,当 $G(z)$ 分子中含有 z^{-r} 因子时,就必须使闭环脉冲传递函数 $\Phi(z)$ 的分子中也含有 z^{-r} 因子,抵消 $G(z)$ 中的 z^{-r} 因子,以免 $D(z)$ 中存在超前环节 z^r 。

在式(8-35)中,若在 $\prod_{i=0}^m (1 - z_i z^{-1})$ 中,存在单位圆上 ($z_i = 1$ 除外) 和单位圆外的不稳定零点 z_i 时,则 $D(z)$ 将是发散、不稳定的,因此, $D(z)$ 中不容许包含 $G(z)$ 的这类零点,这样只能把 $G(z)$ 中 $|z_i| \geq 1$ ($z_i = 1$ 除外) 的零点作为 $\Phi(z)$ 的零点,从而保证 $D(z)$ 的稳定。当然,这样将会使最少拍系统的调节时间加长。

由式(8-27)可知,最少拍系统的闭环脉冲传递函数可表示为

$$\Phi(z) = D(z)G(z)\Phi_e(z) \quad (8-36)$$

若对象 $G(z)$ 的极点 $\prod_{i=0}^n (1 - p_i z^{-1})$ 中,存在单位圆上 ($p_i = 1$ 除外) 和单位圆外的不稳定极点时,从形式上看可由 $D(z)$ 或 $\Phi_e(z)$ 的零点抵消掉。但实际上,不可能由控制器 $D(z)$ 的不稳定零点完全抵消对象 $G(z)$ 的不稳定极点,这是因为数字系统实现时总是具有截断误差,对象 $G(z)$ 的模型也不可能完全准确,且实际对象也是时变的,任何小的误差随时间的积累都会使闭环系统不稳定。这样, $G(z)$ 的不稳定极点只能由误差传递函数 $\Phi_e(z)$ 的零点来抵消。同样,由于要求 $\Phi_e(z)$ 的零点包含 $G(z)$ 的不稳定极点,会使 $\Phi_e(z)$ 变得复杂,误差 $E(z)$ 的展开项数增加,这样将会使最少拍系统的过渡过程时间加长。

根据上面的分析,设计最少拍系统时,考虑到控制器的可实现性和系统的稳定性,必须考虑以下几个条件。

1) 为实现无静差调节,选择 $\Phi_e(z)$ 时,必须针对不同的输入选择不同的形式,通式为

$$\Phi_e(z) = (1 - z^{-1})^N F(z)。$$

2) 保证系统的稳定性, $\Phi_e(z)$ 的零点应包含 $G(z)$ 的所有不稳定极点。

3) 为保证控制器 $D(z)$ 物理上的可实现性, $G(z)$ 的所有不稳定零点和滞后因子均应包含在闭环脉冲传递函数 $\Phi(z)$ 中。

4) 为实现最少拍控制, $F(z)$ 应尽可能简单。 $F(z)$ 的选择要满足恒等式

$$\Phi_e(z) + \Phi(z) = 1 \quad (8-37)$$

例 8-2 在如图 8-8 所示单位反馈线性离散系统中, 设被控对象的传递函数

$$G_p(s) = \frac{10}{s(0.1s+1)(0.05s+1)}, \text{ 采样周期 } T = 0.2s, \text{ 试在单位阶跃输入下设计最少拍数字控制器 } D(z)。$$

器 $D(z)$ 。

解: 被控对象与零阶保持器的等效脉冲传递函数为

$$\begin{aligned} G(z) &= (1 - z^{-1})Z\left[\frac{G_p(s)}{s}\right] = (1 - z^{-1})Z\left[\frac{10}{s^2(0.1s+1)(0.05s+1)}\right] \\ &= \frac{0.76z^{-1}(1+0.05z^{-1})(1+1.065z^{-1})}{(1-z^{-1})(1-0.135z^{-1})(1-0.0185z^{-1})} \end{aligned}$$

式中, 有一个零点 ($z = -1.065$) 在单位圆外和一个滞后因子 z^{-1} 。

根据设计最少拍系统的限制条件, 可假设

$$\Phi_e(z) = (1 - z^{-1})F(z) \quad (8-38)$$

$$\Phi(z) = az^{-1}(1+1.065z^{-1}) \quad (8-39)$$

由 $\Phi_e(z) = 1 - \Phi(z)$ 可知, $\Phi_e(z)$ 、 $\Phi(z)$ 应当是同阶次多项式, 且尽可能简单, 故可取

$$F(z) = (1 + bz^{-1}) \quad (8-40)$$

式 (8-39) 和式 (8-40) 中的 a 和 b 为待定系数。将式 (8-38)、式 (8-39) 和式 (8-40) 分别代入恒等式 $\Phi_e(z) + \Phi(z) = 1$, 可得

$$az^{-1}(1+1.065z^{-1}) + (1 - z^{-1})(1 + bz^{-1}) = 1$$

$$\text{解得} \quad \begin{cases} a = 0.484 \\ b = 0.516 \end{cases}$$

可知

$$\Phi(z) = 0.484z^{-1}(1+1.065z^{-1})$$

$$\Phi_e(z) = (1 - z^{-1})(1 + 0.516z^{-1})$$

由式 (8-27) 得

$$\begin{aligned} D(z) &= \frac{\Phi(z)}{G(z)\Phi_e(z)} \\ &= \frac{0.484z^{-1}(1+1.065z^{-1})}{\frac{0.76z^{-1}(1+0.05z^{-1})(1+1.065z^{-1})}{(1-z^{-1})(1-0.135z^{-1})(1-0.0185z^{-1})} \cdot (1-z^{-1})(1+0.516z^{-1})} \\ &= \frac{0.636(1-0.0185z^{-1})(1-0.135z^{-1})}{(1+0.05z^{-1})(1+0.516z^{-1})} \end{aligned}$$

系统经 $D(z)$ 数字校正后, 在单位阶跃输入作用下, 系统输出响应为

$$\begin{aligned} Y(z) &= \Phi(z)R(z) = 0.484z^{-1}(1+1.065z^{-1}) \cdot \frac{1}{1-z^{-1}} \\ &= 0.484z^{-1} + z^{-2} + z^{-3} + \dots \end{aligned}$$

该式说明输出响应 $y(k)$ 经两拍后, 完全跟踪输入, 稳态误差为零。显然, 由于有单位圆外的零点, 响应时间与表 8-4 相比, 增加了一拍。

系统误差

$$\begin{aligned} E(z) &= \Phi_e(z)R(z) = (1-z^{-1})(1+0.516z^{-1}) \cdot \frac{1}{1-z^{-1}} \\ &= 1+0.516z^{-1} \end{aligned}$$

3. 最少拍无纹波控制器的设计

最少拍控制器的设计方法虽然简单, 但也存在一定的问题: 一是对输入信号的变化适应性差; 二是通过扩展 z 变换方法可以证明, 最少拍系统虽然在采样点处可以实现无静差, 但在采样点之间却有偏差, 通常称之为纹波。这种纹波不但影响系统的控制质量, 还会给系统带来功率损耗和机械磨损。为了准确地设计一个无纹波最少拍系统, 下面通过一个例子分析最少拍系统中纹波产生的原因和解决办法。

例 8-3 在如图 8-8 所示系统中, 设被控对象的传递函数 $G_p(s) = \frac{10}{s(s+1)}$, 采样周期

$T = 1\text{s}$ 。要求:

- 1) 在单位阶跃输入下, 设计一个最少拍数字控制器;
- 2) 分析纹波产生的原因及解决的办法;
- 3) 设计一个无纹波的数字控制器。

解: 被控对象与零阶保持器的等效脉冲传递函数为

$$\begin{aligned} G(z) &= (1-z^{-1})\mathcal{Z}\left[\frac{G_p(s)}{s}\right] = (1-z^{-1})\mathcal{Z}\left[\frac{10}{s^2(s+1)}\right] \\ &= 10(1-z^{-1})\mathcal{Z}\left[\frac{1}{s^2} - \frac{1}{s} + \frac{1}{s+1}\right] \\ &= \frac{3.68z^{-1}(1+0.718z^{-1})}{(1-z^{-1})(1-0.368z^{-1})} \end{aligned}$$

(1) 根据最少拍系统的设计准则, 在单位阶跃输入下, 应取误差传递函数

$$\Phi_e(z) = (1-z^{-1})F_1(z) \quad (8-41)$$

闭环脉冲传递函数

$$\Phi(z) = z^{-1}F_2(z) \quad (8-42)$$

在满足 $\Phi_e(z) = 1 - \Phi(z)$ 时, $F_1(z)$ 、 $F_2(z)$ 的最简单形式是 $F_1(z) = 1$, $F_2(z) = 1$ 。分别代入式 (8-41) 和式 (8-42) 后, 得数字控制器

$$D(z) = \frac{\Phi(z)}{G(z)\Phi_e(z)} = \frac{z^{-1}}{\frac{3.68z^{-1}(1+0.718z^{-1})}{(1-z^{-1})(1-0.368z^{-1})} \cdot (1-z^{-1})}$$

$$= \frac{0.272(1-0.368z^{-1})}{(1+0.718z^{-1})}$$

此时输出 $Y(z) = \Phi(z)R(z) = z^{-1} \frac{1}{1-z^{-1}} = z^{-1} + z^{-2} + z^{-3} + z^{-4} + \dots$

误差 $E(z) = \Phi_e(z)R(z) = (1-z^{-1}) \cdot \frac{1}{1-z^{-1}} = 1 = z^0 + 0 \cdot z^{-1} + 0 \cdot z^{-2} + \dots$

对应 $y(k)$ 和 $e(k)$ 波形如图 8-12 所示。

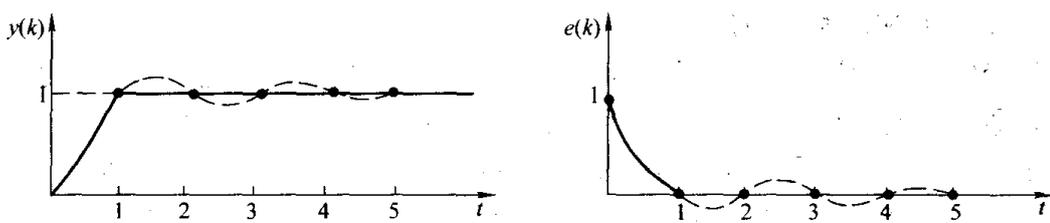


图 8-12 最少拍有纹波控制下 $y(k)$ 和 $e(k)$ 的波形

(2) 分析纹波产生的原因及解决办法

从图 8-12 中可见, 系统经过一拍以后就进入了稳定状态, 但实际上此时控制器的输出

$$U(z) = D(z)E(z) = \frac{0.272(1-0.368z^{-1})}{1+0.718z^{-1}} \cdot 1$$

$$= 0.272 + 0.295z^{-1} - 0.27z^{-2} + 0.248z^{-3} - 0.227z^{-4} + \dots$$

上式说明, 系统输出进入稳态后, 控制器的输出并没有进入稳态, 它作用到被控对象后就形成了纹波, 即在采样点之间存在误差, 输出在平衡点附近出现波动, 如图 8-12 中的虚线所示。那么用什么办法使 $U(z)$ 在有限拍内结束过渡过程呢? 根据 $U(z) = D(z)\Phi_e(z)R(z)$, 可以证明, 只要 $D(z)\Phi_e(z)$ 是关于 z^{-1} 的有限项多项式, 那么在三种典型输入下, $U(z)$ 一定能在有限拍内结束过渡过程, 实现无纹波。

下面以单位阶跃输入和单位速度输入两种情况加以分析说明。

1) 当输入为单位阶跃时, 即 $R(z) = \frac{1}{1-z^{-1}}$, 如果 $D(z)\Phi_e(z) = a_0 + a_1z^{-1} + a_2z^{-2}$ 为有限项多项式, 则

$$U(z) = D(z)\Phi_e(z)R(z) = \frac{a_0 + a_1z^{-1} + a_2z^{-2}}{1-z^{-1}}$$

$$= a_0 + (a_0 + a_1)z^{-1} + (a_0 + a_1 + a_2)(z^{-2} + z^{-3} + z^{-4} + \dots)$$

即从第二个采样周期开始, $u(k)$ 就稳定于一个常数。

2) 当输入为单位速度时, 即 $R(z) = \frac{Tz^{-1}}{(1-z^{-1})^2}$, 设 $D(z)\Phi_e(z) = a_0 + a_1z^{-1} + a_2z^{-2}$ 为有限项

多项式, 则

$$\begin{aligned} U(z) &= D(z)\Phi_e(z)R(z) = (a_0 + a_1z^{-1} + a_2z^{-2}) \cdot \frac{Tz^{-1}}{(1-z^{-1})^2} \\ &= a_0Tz^{-1} + T(2a_0 + a_1)z^{-2} + T(3a_0 + 2a_1 + a_2)z^{-3} + T(4a_0 + 3a_1 + 2a_2)z^{-4} + \dots \end{aligned}$$

由此可见, 对 $u(k)$ 来说, 从第三拍开始, $u(k) = u(k-1) + T(a_0 + a_1 + a_2)$, 即 $u(k)$ 按固定斜率增加且稳定。上述分析是取 $D(z)\Phi_e(z)$ 的项数为三项时的特例。实际上, 当 $D(z)\Phi_e(z)$ 为其他有限项时, 或输入为单位加速度输入时, 仍有上面的结论。

下面讨论使 $D(z)\Phi_e(z)$ 为有限项多项式时所必须满足的条件。

由 $D(z) = \frac{\Phi(z)}{G(z)\Phi_e(z)}$ 知

$$D(z)\Phi_e(z) = \frac{\Phi(z)}{G(z)} \quad (8-43)$$

设被控对象 $G(z) = \frac{Q(z)}{P(z)}$, $P(z)$ 和 $Q(z)$ 分别是 $G(z)$ 的分母和分子多项式, 且无公因子。

代入式 (8-43) 中得 $D(z)\Phi_e(z) = \Phi(z) / \frac{Q(z)}{P(z)} = \frac{\Phi(z)P(z)}{Q(z)}$ 。

显然, 只要闭环脉冲传递函数 $\Phi(z)$ 中包含 $G(z)$ 的全部零点 $Q(z)$, 则 $\Phi(z)P(z)$ 就可以被 $Q(z)$ 整除, 从而 $D(z)\Phi_e(z)$ 必定为有限项。因此可以得出设计无纹波最少拍系统的全部条件。

1) 为实现无静差调节, 应取 $\Phi_e(z) = (1-z^{-1})^N F(z)$, N 可根据三种典型输入分别取 1、2、3。

2) 为保证系统的稳定性, $\Phi_e(z)$ 的零点应包含 $G(z)$ 的所有不稳定极点。

3) 要实现无纹波控制, 闭环脉冲传递函数 $\Phi(z)$ 应包含 $G(z)$ 的全部零点和滞后因子。

4) 为实现最少拍控制, $F(z)$ 应尽可能简单。

(3) 无纹波数字控制器设计

因为被控对象等效脉冲传递函数 $G(z) = \frac{3.68z^{-1}(1+0.718z^{-1})}{(1-z^{-1})(1-0.368z^{-1})}$, 则根据无纹波系统的设计

条件, 可取

$$\Phi_e(z) = (1-z^{-1})(1+az^{-1})$$

$$\Phi(z) = bz^{-1}(1+0.718z^{-1})$$

式中, a 和 b 为待定系数。

上两式代入 $\Phi(z) = 1 - \Phi_e(z)$ 中可解得 $a = 0.418$, $b = 0.582$

即 $\Phi_e(z) = (1-z^{-1})(1+0.418z^{-1})$

$$\Phi(z) = 0.582z^{-1}(1+0.718z^{-1})$$

数字控制器 $D(z) = \frac{\Phi(z)}{G(z)\Phi_e(z)} = \frac{0.158(1-0.368z^{-1})}{1+0.418z^{-1}}$

此时系统输出 $Y(z) = \Phi(z)R(z) = 0.582z^{-1}(1+0.718z^{-1}) \cdot \frac{1}{1-z^{-1}}$

$$= 0.582z^{-1} + z^{-2} + z^{-3} + z^{-4} + \dots$$

采样点的输出 $y(0) = 0, y(1) = 0.582, y(2) = y(3) = y(4) = \dots = 1$

误差 $E(z) = \Phi_e(z)R(z) = (1 - z^{-1})(1 + 0.418z^{-1}) \cdot \frac{1}{1 - z^{-1}} = 1 + 0.418z^{-1}$

采样点的误差 $e(0) = 1, e(1) = 0.418, e(2) = e(3) = e(4) = \dots = 0$

图 8-13 所示为最少拍无纹波控制下 $y(k)$ 和 $e(k)$ 的波形, 可见系统经过两拍后, 实现了无静差完全跟踪。要验证系统是否有纹波, 只要看其控制器输出能否在有限拍内结束过渡过程即可。

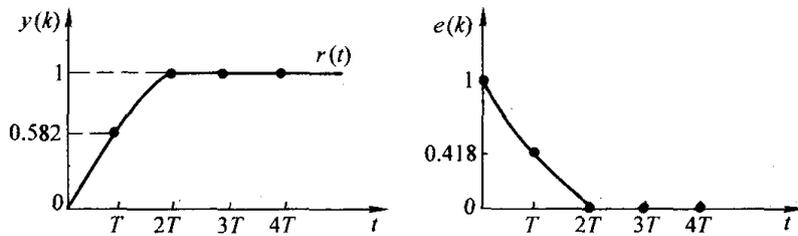


图 8-13 最少拍无纹波控制下 $y(k)$ 和 $e(k)$ 的波形

此时控制器输出

$$U(z) = D(z)E(z) = D(z)\Phi_e(z)R(z)$$

$$= \frac{0.158(1 - 0.368z^{-1})}{1 + 0.418z^{-1}} \cdot (1 + 0.418z^{-1}) = 0.158 - 0.0581z^{-1}$$

可见, 控制信号在第二拍后, $u(2) = u(3) = \dots = 0$, 进入稳态, 故保证了系统输出无纹波。当然, 由于引入了无纹波条件, 其过渡过程时间 ($2T$) 比普通的最少拍系统增加了一拍。

8.3 纯滞后控制

工业过程中的许多对象具有纯滞后特性。例如, 物料经传送带传送到秤体, 蒸汽在长管道内流动至加热罐, 都要经过一定的时间后才能将控制作用送达被控量。这个时间滞后使控制作用不能及时达到效果, 扰动作用不能及时被察觉, 会延误控制, 引起系统的超调和振荡。分析表明, 时间滞后因素 $e^{-\tau s}$ 将直接进入闭环系统的特征方程, 使系统的设计十分困难, 极易引起系统的不稳定。

研究表明, 当对象的纯滞后时间 τ 与主过程对象的惯性时间常数 T 之比, 即 $\tau/T \geq 0.5$ 时, 常规的 PID 控制很难获得良好控制效果。长期以来, 人们对纯滞后对象的控制作了大量的研究, 比较有代表性的方法有大林控制算法和施密斯预估控制算法。

8.3.1 大林 (Dahlin) 控制算法

1968 年, 美国 IBM 公司的大林 (E.B.Dahlin) 提出了一种控制算法, 对被控对象具有纯滞后的过程控制, 具有良好的效果。

1. 大林算法的基本形式

设有一阶惯性的纯滞后对象 $G(s) = \frac{Ke^{-\tau s}}{T_1 s + 1}$, 其中 T_1 为被控对象的时间常数, τ 为纯滞后时间, 且 τ 为采样周期 T 的整数倍, 即 $\tau = NT$ 。

大林算法的设计目标是: 设计一个合适的数字控制器 $D(z)$, 使系统在单位阶跃函数的作用下, 整个系统的闭环传递函数为一个延迟环节 (考虑系统的物理可实现性) 和一个惯性环节 (使输出平滑解决超调) 相串联的形式, 即理想的闭环传递函数为 $\Phi(s) = \frac{e^{-\tau s}}{T_0 s + 1}$, T_0 为闭环系统的等效时间常数。由于是在 Z 平面上讨论数字控制器的设计, 如采用零阶保持器, 且采样周期为 T , 则闭环系统的脉冲传递函数

$$\begin{aligned}\Phi(z) &= (1 - z^{-1}) \mathcal{Z} \left[\frac{e^{-NTs}}{s(T_0 s + 1)} \right] \\ &= z^{-N} (1 - z^{-1}) \cdot \frac{(1 - e^{-T/T_0}) z^{-1}}{(1 - z^{-1})(1 - e^{-T/T_0} z^{-1})} \\ &= \frac{z^{-(N+1)} \cdot (1 - e^{-T/T_0})}{(1 - e^{-T/T_0} z^{-1})}\end{aligned}\quad (8-44)$$

类似地, 可得被控对象的脉冲传递函数为

$$G(z) = K \frac{z^{-(N+1)} \cdot (1 - e^{-T/T_1})}{(1 - e^{-T/T_1} z^{-1})}\quad (8-45)$$

根据直接离散化设计的原理可得

$$\begin{aligned}D(z) &= \frac{\Phi(z)}{G(z)[1 - \Phi(z)]} \\ &= \frac{\frac{z^{-(N+1)} \cdot (1 - e^{-T/T_0})}{(1 - e^{-T/T_0} z^{-1})}}{\frac{K z^{-(N+1)} \cdot (1 - e^{-T/T_1})}{(1 - e^{-T/T_1} z^{-1})} \left[1 - \frac{z^{-(N+1)} \cdot (1 - e^{-T/T_0})}{(1 - e^{-T/T_0} z^{-1})} \right]} \\ &= \frac{(1 - e^{-T/T_0})(1 - e^{-T/T_1} z^{-1})}{K \cdot (1 - e^{-T/T_1}) \left[1 - e^{-T/T_0} z^{-1} - (1 - e^{-T/T_0}) z^{-(N+1)} \right]}\end{aligned}\quad (8-46)$$

式 (8-46) 即为被控对象为带有纯滞后的一阶惯性环节时, 大林控制器的表达式, 显然 $D(z)$ 可由计算机直接实现。

对带有纯滞后的二阶惯性环节的被控对象, 即 $G(s) = \frac{Ke^{-\tau s}}{(T_1 s + 1)(T_2 s + 1)}$, 设闭环脉冲传递函数仍为式 (8-44), 则

$$D(z) = \frac{(1 - e^{-T/T_0})(1 - e^{-T/T_1}z^{-1})(1 - e^{-T/T_2}z^{-1})}{K \cdot (c_1 + c_2z^{-1})[1 - e^{-T/T_0}z^{-1} - (1 - e^{-T/T_0})z^{-(N+1)}]} \quad (8-47)$$

其中 $c_1 = 1 + \frac{1}{T_2 - T_1}(T_1e^{-T/T_1} - T_2e^{-T/T_2})$

$$c_2 = e^{-T(1/T_1+1/T_2)} + \frac{1}{T_2 - T_1}(T_1e^{-T/T_2} - T_2e^{-T/T_1})$$

2. 振铃现象及消除方法

人们发现，直接用上述控制算法构成闭环控制系统时，计算机的输出 $U(z)$ 常会以 $1/2$ 采样频率大幅度上下振荡。这一振荡将使执行机构的磨损增加，而且影响控制质量，甚至可能破坏系统的稳定，必须加以消除。通常这一振荡现象被称为振铃现象。

为了衡量振荡的强烈程度，可引入振铃幅度 RA 的概念。RA 的定义为：在单位阶跃输入作用下，数字控制器 $D(z)$ 的第 0 次输出与第 1 次输出之差为振铃幅度，即 $RA = U(0) - U(1)$ 。表 8-4 给出了 $D(z)$ 在不同形式下的振铃特性。

从表中可以看出：当极点 $z = -1$ 时，振铃幅度 $RA = 1$ ；当极点 $z = -0.5$ 时，振铃幅度 $RA = 0.5$ 。当右半 Z 平面上有极点时，振铃减轻；当右半 Z 平面上有零点时，振铃加剧。可以证明，振铃的根源就是 $z = -1$ 附近的极点所致，且 $z = -1$ 处振铃最严重。

表 8-4 单位阶跃输入下的振铃特性

数字控制器 $D(z)$	输出 $u(k)$	振铃幅度 RA	输出序列图
$\frac{1}{1+z^{-1}}$	1 0 1 0 1	1	
$\frac{1}{1+0.5z^{-1}}$	1.0 0.5 0.75 0.625 0.646	0.5	
$\frac{1}{(1+0.5z^{-1})(1-0.2z^{-1})}$	1.0 0.7 0.89 0.803 0.848	0.3	
$\frac{(1-0.5z^{-1})}{(1+0.5z^{-1})(1-0.2z^{-1})}$	1.0 0.2 0.5 0.37 0.46	0.8	

为了消除振铃，大林提出了一个切实可行的办法，就是先找到 $D(z)$ 中可能产生振铃的极

点 ($z = -1$ 附近的极点), 然后令该极点的 $z = 1$ 。这样, 既取消了这个极点, 又不影响系统的稳态输出。因为根据终值定理, 系统的稳态输出 $Y(\infty) = \lim_{z \rightarrow 1} (z-1)Y(z)$, 显然系统进入稳态后

$z = 1$ 。

下面讨论消除振铃后数字控制器的形式。将式 (8-46) 的分母进行分解, 得

$$D(z) = \frac{(1 - e^{-T/T_0})(1 - e^{-T/T_1}z^{-1})}{K \cdot (1 - e^{-T/T_1})(1 - z^{-1}) \left[1 + (1 - e^{-T/T_0})(z^{-1} + z^{-2} + z^{-3} + \dots + z^{-N}) \right]}$$

显然, 式中极点 $z = 1$, 不会引起振铃。引起振铃的可能因子是 $\left[1 + (1 - e^{-T/T_0})(z^{-1} + z^{-2} + z^{-3} + \dots + z^{-N}) \right]$ 项。

当 $N = 0$ 时, 此因子不存在, 无振铃可能。

当 $N = 1$ 时, 有一个极点 $z = -(1 - e^{-T/T_0})$ 。

在 $T_0 \ll T$ 时, $z \rightarrow -1$, 存在严重的振铃现象。为消除振铃, 可令 $z = 1$, 因子变为 $1 + (1 - e^{-T/T_0})z^{-1} = 2 - e^{-T/T_0}$, 此时

$$D(z) = \frac{(1 - e^{-T/T_0})(1 - e^{-T/T_1}z^{-1})}{K \cdot (1 - e^{-T/T_1})(1 - z^{-1})(2 - e^{-T/T_0})} \quad (8-48)$$

当 $N = 2$ 时, 因子变为 (令 $z = 1$) $1 + (1 - e^{-T/T_0})(z^{-1} + z^{-2}) = 3 - 2e^{-T/T_0}$, 此时

$$D(z) = \frac{(1 - e^{-T/T_0})(1 - e^{-T/T_1}z^{-1})}{K \cdot (1 - e^{-T/T_1})(1 - z^{-1})(3 - 2e^{-T/T_0})} \quad (8-49)$$

式 (8-48) 和式 (8-49) 就是对纯滞后对象 ($N = 1$ 和 $N = 2$ 时) 用大林算法设计出的数字控制器 $D(z)$ 。

8.3.2 施密斯 (Smith) 预估控制算法

1957 年施密斯 (Smith) 提出了一种纯滞后的补偿模型, 但当时的模拟仪表无法实现, 直至后来利用计算机可以完成大滞后时间补偿的预估控制。

如图 8-14 所示, $G_p(s)$ 和 τ 分别为控制对象的不包含滞后环节的传递函数和纯滞后时间, 该算法的核心是控制回路中增加 Smith 预估器 $G_p(s)(1 - e^{-\tau s})$, 与常规控制器 $D(s)$ 并联共同组成纯滞后补偿控制器, 即

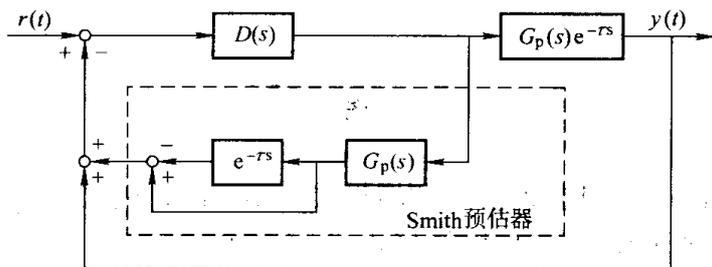


图 8-14 Smith 预估控制方框图

$$D'(s) = \frac{D(s)}{1 + D(s)G_p(1 - e^{-\tau s})}$$

经补偿后的系统闭环传递函数为

$$\Phi(s) = \frac{D'(s)G_p(s)e^{-\tau s}}{1 + D'(s)G_p(s)e^{-\tau s}} = \frac{D(s)G_p(s)}{1 + D(s)G_p(s)}e^{-\tau s} \quad (8-50)$$

式(8-50)说明,对常规控制器 $D(s)$ 来说,包含原控制对象 $G_p(s)e^{-\tau s}$ 与Smith预估器的广义被控对象只相当于 $G_p(s)$,即纯滞后环节 $e^{-\tau s}$ 被放在了闭环控制回路之外。拉氏变换的位移定理说明, $e^{-\tau s}$ 仅将控制作用在时间坐标上推移了一个时间,控制系统的过渡过程及其他性能指标都与对象特性为 $G_p(s)$ 时完全相同。因此将Smith预估器与控制器并联,理论上可以使控制对象的时间滞后得到完全补偿。

采用模拟调节仪表实现上述Smith预估器比较困难,这是因为对象模型 $G_p(s)$ 各式各样,纯滞后环节由模拟电路 $e^{-\tau s}$ 模拟也很麻烦。相反地,由计算机来实现Smith预估器却很容易。考虑到计算机控制系统中控制输出后具有零阶保持器,为了与离散化的被控对象对应,Smith预估器的离散化也采用零阶保持器法,设Smith预估器的等效脉冲传递函数为 $G_r(z)$,则

$$\begin{aligned} G_r(z) &= (1 - z^{-1})\mathcal{Z}\left[\frac{G_r(s)}{s}\right] = (1 - z^{-1})\mathcal{Z}\left[(1 - e^{-\tau s})\frac{G_p(s)}{s}\right] \\ &= (1 - z^{-1})(1 - z^{-N})\mathcal{Z}\left[\frac{G_p(s)}{s}\right] \end{aligned} \quad (8-51)$$

式中, $N = (\text{int})(\tau/T)$,一般地采样周期 T 取纯滞后时间 τ 的整倍数关系。

上述Smith预估器 $G_r(z)$ 的输入为控制器 $D(z)$ 的输出,式(8-51)中后移算子 z^{-1} 、 z^{-N} 可以通过计算机存储单元的移位方便地实现。而数字控制器 $D(z)$ 除了最常用的PID外,还可以是其他的控制算法。

一般认为,Smith预估补偿方法是解决大滞后问题的有效方法,预估系统在模型基本准确时表现出良好的性能,但预估器对模型的精度或运行条件的变化十分敏感,对预估模型的精度要求较高,鲁棒性较差。研究表明,简单PID控制系统承受对象参数变化的能力要强于带有Smith预估器的系统。正是由于上述Smith预估器对模型误差敏感的原因,限制了Smith预估补偿方法在工业过程控制系统中的推广及应用。为了克服Smith预估器对模型误差敏感、鲁棒性差的不足,国内、外控制界针对Smith预估器还提出了各种各样的改进算法,这里就不一一介绍了。

思 考 题

1. 什么是数字PID位置式控制算法和增量式控制算法?试比较它们的优、缺点。
2. 什么是积分饱和现象?它是怎样引起的?如何消除?
3. 什么是数字PID的积分分离算法?它有何优点?
4. 试分析说明理想微分PID与实际微分PID输出控制作用有何区别?实际微分PID有何优

越性?

5. 为什么可以把带死区的PID控制算法称为非线性的PID控制算法?
6. 何为PID控制器参数的整定? 整定的实质是什么? 采样周期如何确定?
7. 设采样周期 $T=1$, 试根据实际微分PID算式

$$G(s) = \frac{U(s)}{E(s)} = K_p \frac{1}{1+T_f s} \left(1 + \frac{1}{T_i s} + T_d s \right)$$

采用双线性变换法离散化, 推导其数字算式。

参考答案:
$$\begin{cases} \Delta u(k) = C_1 \Delta u(k-1) + C_2 e(k) + C_3 e(k-1) + C_4 e(k-2) \\ u(k) = u(k-1) + \Delta u(k) \end{cases}$$

其中:
$$C_1 = \frac{2T_f - 1}{2T_f + 1}, \quad C_2 = \frac{K_p}{2T_i(1+2T_f)}(1+2T_i+4T_iT_d),$$

$$C_3 = \frac{K_p}{2T_i(1+2T_f)}(2-8T_iT_d), \quad C_4 = \frac{K_p}{2T_i(1+2T_f)}(1-2T_i+4T_iT_d)$$

8. 最少拍系统设计是否意味着采样周期越小, 系统的调节时间就会越短? 最少拍有纹波设计与无纹波设计的主要差别是什么?

9. 设被控对象的传递函数 $G_p(s) = \frac{5}{s(0.1s+1)(0.05s+1)}$, 采样周期 $T=0.1s$, 试在单位速度输入下设计最少拍有纹波和无纹波数字控制器 $D(z)$ 。

10. 什么是大林算法? 振铃现象产生的原因是什么? 如何消除?

11. 已知被控对象 $G(s) = \frac{e^{-s}}{0.1s+1}$, 设采样周期 $T=0.5s$, 试按大林算法设计数字控制器 $D(z)$, 并分析系统是否会产生振铃现象? 若有试求出RA, 并消除之。

第9章 先进控制技术基础

传统的控制策略隐含着两个前提，一是要求对象的模型是精确的、不变化的，且是线性的；二是操作条件和运行环境是确定的、不变的。一般的工业系统只是粗略地、近似地满足这些条件，在要求不高的情况下是可行的。随着工业应用领域的扩大，控制精度和性能要求的提高，必须考虑控制对象参数乃至结构的变化、非线性的影响、运行环境的改变以及环境干扰等时变的和不确定因素，才能得到满意的控制效果。在实际应用需求的激励下，在计算机的高速、小型、大容量、低成本所提供的良好物质条件下，一系列新型控制策略应运而生，并迅速在实际中得到应用、改进和发展。

由于这些先进控制算法本身的复杂性，依靠常规的控制仪表难于实现，一般都是由计算机来实现的，从这个意义上讲这些算法都可以归到计算机控制算法这一大类中。作为计算机控制基本方法的扩展，本章将对工业应用较多的几个有代表性的控制策略进行简要介绍，主要包括自适应控制、模糊控制和神经网络控制等，在10.3节中有这方面应用的一个实例作为进一步的介绍。

9.1 自适应控制

自适应控制是针对对象特性的变化、漂移和环境干扰对系统的影响而提出来的，它的基本思想是通过在线辨识使这种影响逐渐降低以至消除。20世纪50年代末，美国麻省理工学院提出了第一个自适应控制方案，但由于实现上的困难和计算上的复杂，没有得到充分的发展。70年代后，计算机技术特别是微型机的发展和普及，随机控制和系统辨识等的成熟，使得对自适应控制的研究又活跃起来。数十年来，经国内外众多研究和应用人员的努力，已形成了较为完整的理论，并获得了许多成功的应用。

在反馈控制和最优控制系统中，通常假设被控对象的数学模型是已知的而且是确定性的线性系统。然而，实际上多数的被控对象或过程的数学模型事先是难以确知的，尤有甚者，由于原料品种、设备工作状况和环境条件等发生变化，也会引起系统参数的变化。被控对象实质上是非线性的，这种非线性对控制系统的工作带来很大的困难，它是一般反馈控制和最优控制所难以解决的。为了使系统仍能自动地保持在某种意义下的最优状态，采用随着参数变化而自动改变控制策略的自适应控制就成为必要的了。

“一个自适应控制系统必须提供出被控对象当前状态的连续信息，也就是要辨识对象，并用当前系统的性能与期望的或最优的性能相比较，并作出使系统趋向期望的或最优的性能的决策。最后它必须对控制器进行适当的修正以驱使系统走向最优状态。”

由此可见，自适应控制系统应该包含下列三个基本功能。

1) 辨识被控对象的结构和参数或性能指标的变化，以便精确地建立被控对象的数学模型，或当前的实际性能指标。

2) 综合出一种控制策略或控制规律，以便确保被控系统达到期望的性能指标。

3) 自动地修正控制器参数以保证所综合出的控制策略在被控对象上得到实现。

目前所研究的自适应控制系统最基本的有两大类，即参考模型自适应控制系统和具有被控对象数学模型在线辨识的自适应控制系统。

9.1.1 自校正控制

1. 具有被控对象数学模型在线辨识的自适应控制系统

这类系统的特点是根据系统运行数据 $y(t)$ 、 $u(t)$ 对被控对象进行在线辨识，然后再根据辨识出来的模型参数和事先指定的性能指标在线地综合出最佳的控制作用。通常在设计辨识算法时，已考虑了随机扰动和量测噪声的影响，所以它属于随机自适应控制系统这一类。如图 9-1 所示，其中 $y(t)$ 是参考输入， $\xi(t)$ 是随机噪声， $\hat{\theta}(t)$ 和 $\hat{x}(t)$ 分别为参数和状态的估计值。

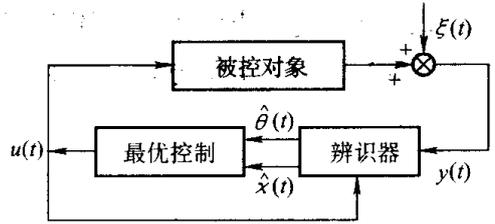


图 9-1 带有模型在线辨识的自适应控制系统

此类系统工作原理是：辨识器根据 $\{u(t)\}$ 、 $\{y(t)\}$ 序列用一定的估计算法在线地对被控对象的未知参数和未知状态进行估计，得到未知参数估计值 $\hat{\theta}(t)$ 和未知状态 $\hat{x}(t)$ 的估计值。再利用 $\hat{\theta}(t)$ 和 $\hat{x}(t)$ 及事先指定的性能指标综合出最优控制作用 $u(t)$ 。这样经过不断地辨识和控制，系统的实际性能指标将逐渐地达到最优。

这类系统中用得最为广泛的是自校正控制系统，它的理论基础是系统辨识和随机最优控制理论。根据所采用的参数估计方法的不同和控制目标函数的不同，原则上可以构成复杂度各不相同的自校正调节器。但是，从工业应用角度，应当在基本满足控制性能的前提下力求调节器简单可靠。在实际应用中，常以递推最小二乘为参数估计方法，以最小方差为控制目标函数。下面首先介绍参数估计的最小二乘法，然后介绍最小方差控制和自校正调节器。

2. 参数估计的最小二乘法

在自适应控制系统中，系统辨识都是通过数字计算机来实现的，所以下面讨论线性动态离散模型的参数估计。离散动态系统参数估计众多方法中，从理论上和实践上以最小二乘法最为成熟而且应用广泛。这里主要介绍最小二乘法的基本内容，由基本最小二乘法派生出来的比较复杂的算法，如广义最小二乘法等，读者可参看有关系统辨识方面的书籍。

(1) 一次完成最小二乘估计

设系统由下列 n 阶差分方程描述：

$$y(k) + a_1 y(k-1) + \dots + a_n y(k-n) = b_0 u(k) + \dots + b_n u(k-n) + e(k) \quad (9-1)$$

或写成

$$\begin{cases} A(z^{-1})y(k) = B(z^{-1})u(k) + e(k) \\ A(z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_n z^{-n} \\ B(z^{-1}) = b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_n z^{-n} \end{cases} \quad (9-2)$$

式中， z^{-1} 为延迟算子，如 $z^{-1}y(k) = y(k-1)$ ， $z^{-2}y(k) = y(k-2)$ ， \dots ； $\{u(k)\}$ 、 $\{y(k)\}$ 是输入与输出量的不同时刻的测量值组成的离散时间序列； $\{e(k)\}$ 为不可测量的随机干扰序列。假设方程的阶数 n 是已知的，而参数 $a_1, a_2, \dots, a_n, b_0, b_1, \dots, b_n$ 是未知的。现在的问题是如何通过带

有噪声的输出和输入观测数据估计模型的参数。

把待估计的模型参数和 k 时刻以前的 n 次采样数据记为向量形式, 即有

$$\theta = [a_1, a_2, \dots, a_n, b_0, b_1, \dots, b_n]^T$$

$$\phi^T(k-1) = [-y(k-1), \dots, -y(k-n), u(k), \dots, u(k-n)]$$

则式 (9-2) 可以写成

$$y(k) = \phi^T(k-1)\theta + e(k) \quad (9-3)$$

式中, $e(k)$ 为观测噪声和模型不准等形成的误差。把 $k = n+1, n+2, \dots, n+N$ 共 N 步采样数据, 代入式 (9-3) 组成 N 个方程, 并且用矩阵形式表达为

$$Y_N = \Phi_N \theta + \varepsilon_N \quad (9-4)$$

其中

$$Y_N = \begin{bmatrix} y(n+1) \\ y(n+2) \\ \vdots \\ y(n+N) \end{bmatrix}_{N \times 1} \quad \varepsilon_N = \begin{bmatrix} e(n+1) \\ e(n+2) \\ \vdots \\ e(n+N) \end{bmatrix}_{N \times 1}$$

$$\Phi_N = \begin{bmatrix} \phi^T(n+1) \\ \phi^T(n+2) \\ \vdots \\ \phi^T(n+N) \end{bmatrix} = \begin{bmatrix} -y(n) & \cdots & -y(1) & u(n+1) & \cdots & u(1) \\ -y(n+1) & \cdots & -y(2) & u(n+2) & \cdots & u(2) \\ \vdots & & \vdots & \vdots & & \vdots \\ -y(n+N-1) & \cdots & -y(N) & u(n+N) & \cdots & u(N) \end{bmatrix}_{N \times (2n+1)}$$

最小二乘法是选择模型参数 $\hat{\theta}$, 使模型输出与所有实际观测值总体上最接近, 所以最小二乘估计的性能指标取为

$$\min J = \sum_{k=n+1}^{n+N} e^2(k) = \varepsilon_N^T \varepsilon_N = (Y_N - \Phi_N \theta)^T (Y_N - \Phi_N \theta) \quad (9-5)$$

由数学中的极值条件 $\left. \frac{\partial J}{\partial \theta} \right|_{\theta=\hat{\theta}} = 0$, 可以求得使 J 最小的 $\hat{\theta}$ 。由式 (9-5) 得

$$J = (Y_N - \Phi_N \theta)^T (Y_N - \Phi_N \theta)$$

$$= Y_N^T Y_N - 2\theta^T \Phi_N^T Y_N + \theta^T \Phi_N^T \Phi_N \theta$$

$$\left. \frac{\partial J}{\partial \theta} \right|_{\theta=\hat{\theta}} = -2\Phi_N^T Y_N + 2\Phi_N^T \Phi_N \theta \Big|_{\theta=\hat{\theta}} = 0$$

$$\Phi_N^T \Phi_N \hat{\theta} = \Phi_N^T Y_N$$

若 $\Phi_N^T \Phi_N$ 满秩, 则动态离散模型参数的最小二乘估计 $\hat{\theta}$ 为

$$\hat{\theta} = (\Phi_N^T \Phi_N)^{-1} \Phi_N^T Y_N \quad (9-6)$$

为了保证 $\Phi_N^T \Phi_N$ 满秩, 要求 $N \geq 2n+1$, 一般取测量数据的组数 N 为 100~200。

(2) 递推最小二乘估计

上面介绍的参数估计方法是观测到 N 组输入、输出数据以后, 再根据式 (9-6) 求得参数

估计值 $\hat{\theta}$ ，所以称为一次完成法。但是，自适应控制系统的参数估计是在线进行的，新的观测数据源源不断而来，希望利用这些新的观测信息来不断改进参数估计，如果利用一次估计公式 (9-6) 不断进行运算，显然是不现实的，这是因为矩阵 Φ_N 的维数将随着数据个数的增加而不断增加，矩阵 $\Phi_N^T \Phi_N$ 的求逆运算越来越困难，甚至是不可能的。另外，由于老数据要保留，而新数据不断增加，所以存储量也要不断增加。因此对于在线估计，必须用递推算法才能实现。事实上，当观测数据较多时，即使对于离线估计矩阵求逆也是很困难的，所以经常采用递推算法。

在进行 $n + N$ 次观测后，根据一次完成法可以构成输出向量 Y_N 和数据矩阵 Φ_N ，得到最小二乘估计 $\hat{\theta} = (\Phi_N^T \Phi_N)^{-1} \Phi_N^T Y_N$ ，当又获得了一组新的观测数据 $\{y(n + N + 1), u(n + N + 1)\}$ 后，同样可构成 Y_{N+1} 和 Φ_{N+1} ，得到最小二乘估计

$$\hat{\theta}_{N+1} = (\Phi_{N+1}^T \Phi_{N+1})^{-1} \Phi_{N+1}^T Y_{N+1} \quad (9-7)$$

通过分析第 N 次和第 $N+1$ 次参数估计间的关系，得出如下最小二乘递推算式

$$\begin{cases} \hat{\theta}(k) = \hat{\theta}(k-1) + \mathbf{K}(k)[y(k) - \phi^T(k)\hat{\theta}(k-1)] \\ \mathbf{K}(k) = \frac{P(k-1)\phi(k)}{1 + \phi^T(k)P(k-1)\phi(k)} \\ P(k) = [I - \mathbf{K}(k)\phi^T(k)]P(k-1) \end{cases} \quad (9-8)$$

其中， $\hat{\theta}_N$ 简记为 $\hat{\theta}(k-1)$ ， $\hat{\theta}_{N+1}$ 简记为 $\hat{\theta}(k)$ ， $\phi^T(n + N + 1)$ 简记为 $\phi^T(k)$ ，且记 $P(k-1) = (\Phi_N^T \Phi_N)^{-1}$ ， $y(n + N + 1)$ 简记为 $y(k)$ 。

公式 (9-8) 表明，新的参数估计 $\hat{\theta}(k)$ 是用新的实际测量值 $y(k)$ 与基于老模型进行预测得到的量 $\phi^T(k)\hat{\theta}(k-1)$ 之偏差，对前面的参数估计 $\hat{\theta}(k-1)$ 加以修正得到的，修正系数阵为 $\mathbf{K}(k)$ 。 $P(k-1)$ 的物理意义是参数估计误差的方差，作为 $n + N$ 时刻参数估计精度的一种度量。 $P(k-1)$ 越大表示参数估计值越不准确， $P(k-1)$ 越小表示参数估计值越接近真值。

若已知 $\hat{\theta}(k-1)$ 、 $P(k-1)$ 的初始值 $\hat{\theta}_0$ 、 P_0 ，即可递推算出以后的参数估计值。设置初始值有很多方法，在实践中常用下述简便的方法。若估计前知道参数的大致范围，则可以在此范围内设置初值，若完全不知道 θ 的范围时，可任意设置 $\hat{\theta}_0$ ，为简便通常取 $\hat{\theta}_0 = 0$ 。由于 θ 的初始值是任意设定的，因此 P_0 应设置得很大，表示初始估计的精度很差。通常取 $P_0 = \alpha^2 I$ ，其中 α 为充分大的数，一般取计算机容许的最大值，如 $\alpha^2 = 10^6$ 。

可以证明，对于任意设置的初值，当递推到 $2n$ 步时，得到的 $\hat{\theta}$ 和 P 与一次完成法得到的结果很接近，所以这种设置初值的方法在工程实践中被广泛采用。

应用上述的递推公式进行一次辨识，所需的计算量大大减少，数据的存储量也大大减少。若系统是 n 阶，采样数据个数为 N ，用一次完成法，计算机需要存储 $n+N$ 组数据，而用递推算法则只需存储 $n+1$ 组数据。因为 N 一般为上百甚至上千，所以，两者存储的数据量相差很大。

根据递推算法和一次完成算法各自的优缺点，递推算法常用于在线辨识和实时控制，而一次完成法辨识精度较高，常用于离线建模。

(3) 带遗忘因子的递推最小二乘估计

递推算法(9-8)的一个显著特点是:对所有观测数据的加权是相同的,这就意味着老数据和新数据对于当前参数估计所提供的信息同等重要。显然,如果辨识的系统是定常系统,这种考虑是合理的,可以得到更多的信息,从而提高辨识精度。但对于时变系统,老的观测数据只能反映系统变化前的特性,数据越陈旧,偏离当前动态的可能性越大。而当前数据最能反映时变系统当前时刻的动态特性。因此,平等地使用参数变化前后的观测数据辨识变化后的系统参数,显然是不合理的。

在工程实践中,常采用带有遗忘因子的递推最小二乘估计算法辨识时变系统。这种方法的基本思想是充分重视当前的数据,而削弱老数据的作用,即将过去的的数据逐渐“遗忘”掉。具体作法是在递推计算过程中,每取得一个新的数据 $y(n+N+1)$ 时,将以前所有的数据都乘以一个加权因子 $\rho(0 < \rho < 1)$ 后参加辨识,即

$$Y_{N+1} = \begin{bmatrix} \rho Y_N \\ \dots \\ y(n+N+1) \end{bmatrix} \quad \Phi_{N+1} = \begin{bmatrix} \rho \Phi_N \\ \dots \\ \phi^T(n+N+1) \end{bmatrix}$$

显然,这样使得数据的作用以指数规律 ρ^k 衰减,因为第 k 次观测之前第一个观测值乘以 ρ 后构造 Y 和 X ,之前第二个测量值乘以 ρ^2 构造 Y 和 X ,……。由于 $0 < \rho < 1$,所以越是老的数据作用越小。

仿照前面的推导递推算式(9-8)的过程,并令 $\beta = \rho^2(0 < \beta < 1)$, β 称为“遗忘因子”,可得到带遗忘因子的递推最小二乘估计的算式为

$$\begin{cases} \hat{\theta}(k) = \hat{\theta}(k-1) + K(k)[y(k) - \phi^T(k)\hat{\theta}(k-1)] \\ K(k) = \frac{P(k-1)\phi(k)}{\beta + \phi^T(k)P(k-1)\phi(k)} \\ P(k) = \frac{1}{\beta} [I - K(k)\phi^T(k)]P(k-1) \end{cases} \quad (9-9)$$

遗忘因子 β 值的大小对参数估计结果有很大影响。 β 越小表明遗忘得越快,越重视当前数据,越能反映当前系统的变化,这适合于参数变化速度相对于辨识速度较快的时变系统。 β 取得越大,重视了更多的历史数据,可以得到更多的系统信息,因此辨识的模型精度较高,适合于参数变化速度远低于辨识速度的慢时变系统。否则,由于没有充分利用老数据中所含的系统信息,辨识精度较低。一般 β 在 $0.95 \sim 0.995$ 的范围内选取。当 $\beta=1$ 时,表示没有“遗忘”,式(9-9)就成为式(9-8)。

在前面讨论线性离散系统的参数估计时,实际上都是假设已知模型的阶次和滞后步数,事实上模型的阶次和滞后步数也需要由观测数据根据某个准则辨识得到,这个过程通常称为“定阶”。定阶一般是按“假设-检验”的步骤反复进行的,即由低阶向高阶逐次假定系统的阶次,分别估计模型的参数,然后对得到的模型进行检验,以满足要求的最低阶作为所确定的模型阶次。低阶模型对系统描述粗糙,但分析设计容易,而高阶模型对系统描述精度高,但分析设计复杂,尤其是不利于在线辨识与自适应控制。

(4) 闭环系统的可辨识性

前面介绍的参数估计方法, 明显地或者隐含地要求系统的输入信号与输出端的噪声是不相关的。然而, 除了少数系统是开环控制外, 大多数被辨识对象是闭环系统中的一部分。例如, 下面介绍的自校正控制系统都是反馈系统, 由于存在反馈, 系统输入信号与被控对象输出端的噪声将是相关的。在很多情况下, 根据闭环条件下的观测数据辨识对象的模型参数时, 可能会得到错误的结果。20世纪70年代以后, 许多学者开始研究闭环系统的辨识问题, 研究结果表明, 在一定条件下可以在闭环状态下辨识对象模型。下面简单介绍最小二乘法的闭环辨识条件。

考察图9-2所示闭环控制系统, 由扰动 $e(k)$ 产生的输出 $y(k)$ 为

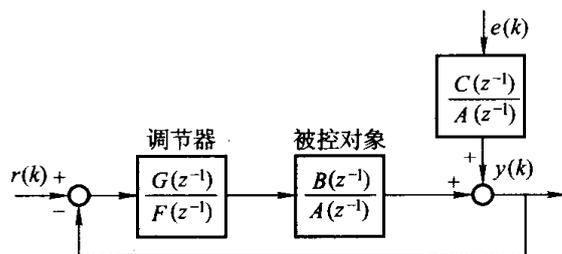


图9-2 闭环系统的可辨识性

$$y(k) = \frac{F(z^{-1})C(z^{-1})}{A(z^{-1})F(z^{-1}) + B(z^{-1})G(z^{-1})} e(k) = \frac{Q(z^{-1})}{P(z^{-1})} e(k) \quad (9-10)$$

其中

$$A(z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_n z^{-n}$$

$$B(z^{-1}) = b_1 z^{-1} + b_2 z^{-2} + \dots + b_n z^{-n}$$

$$C(z^{-1}) = 1 + c_1 z^{-1} + c_2 z^{-2} + \dots + c_n z^{-n}$$

$$F(z^{-1}) = 1 + f_1 z^{-1} + f_2 z^{-2} + \dots + f_\mu z^{-\mu}$$

$$G(z^{-1}) = g_0 + g_1 z^{-1} + g_2 z^{-2} + \dots + g_\nu z^{-\nu}$$

$$P(z^{-1}) = A(z^{-1})F(z^{-1}) + B(z^{-1})G(z^{-1}) = 1 + p_1 z^{-1} + p_2 z^{-2} + \dots + p_l z^{-l}$$

$$Q(z^{-1}) = C(z^{-1})F(z^{-1}) = 1 + q_1 z^{-1} + q_2 z^{-2} + \dots + q_\gamma z^{-\gamma}$$

$$l = n + \max(\mu, \nu)$$

$$\gamma = n + \mu$$

闭环系统可辨识的第一个条件是闭环稳定, 否则, $\{y(k)\}$ 不是平稳随机过程。因此下面设闭环系统是稳定的, 由 $P(z^{-1}) = A(z^{-1})F(z^{-1}) + B(z^{-1})G(z^{-1})$ 得

$$\begin{aligned} & 1 + p_1 z^{-1} + p_2 z^{-2} + \dots + p_l z^{-l} \\ &= (1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_n z^{-n})(1 + f_1 z^{-1} + f_2 z^{-2} + \dots + f_\mu z^{-\mu}) \\ & \quad + (b_1 z^{-1} + b_2 z^{-2} + \dots + b_n z^{-n})(g_0 + g_1 z^{-1} + g_2 z^{-2} + \dots + g_\nu z^{-\nu}) \end{aligned} \quad (9-11)$$

比较式(9-11)两边 z^{-1} 的同次幂的系数, 可得下列方程组

$$\begin{cases} a_1 + b_1 g_0 = p_1 - f_1 \\ a_1 f_1 + a_2 + b_1 g_1 + b_2 g_0 = p_2 - f_2 \\ a_1 f_2 + a_2 f_1 + a_3 + b_1 g_2 + b_2 g_1 + b_3 g_0 = p_3 - f_3 \\ \dots \\ a_1 f_{j-1} + \dots + a_n f_{j-n} + b_1 g_{j-1} + \dots + b_n g_{j-n} = p_j - f_j \end{cases} \quad (9-12)$$

其中, $f_i = 0; g_i = 0, i < 0 f_0 = 1$
 写成矩阵形式

$$S\theta = \alpha^* \tag{9-13}$$

其中

$$S = \begin{bmatrix} 1 & 0 & g_0 & 0 \\ f_1 & 1 & g_1 & g_0 \\ f_2 & f_1 & \vdots & g_1 \\ \vdots & f_2 & \ddots & \vdots \\ f_\mu & \vdots & f_1 & g_\nu \\ & f_\mu & f_2 & g_1 \\ & & \vdots & \vdots \\ & & & g_\nu \\ 0 & f_\mu & 0 & \end{bmatrix}; \theta = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \\ b_1 \\ \vdots \\ b_n \end{bmatrix}; \alpha^* = \begin{bmatrix} p_1 - f_1 \\ p_2 - f_2 \\ \vdots \\ p_\mu - f_\mu \\ p_{\mu+1} \\ \vdots \\ p_{2n} \end{bmatrix}$$

式 (9-12) 或式 (9-13) 描述了调节器、闭环系统和被控对象三者参数之间的关系。根据线性方程组解的存在性可知, 当 S 是 $2n \times 2n$ 方阵且秩等于 $2n$ 时 θ 有惟一解 $\theta = S^{-1}\alpha^*$; 当 S 的行数大于 $2n$ 而秩等于 $2n$ 时 θ 有惟一最小二乘解 $\theta = (S^T S)^{-1} S^T \alpha^*$ 。总之, 参数 θ 有解的条件是 S 的秩等于 $2n$, 即 $\text{rank}[S] = 2n$ 。从 S 的结构不难看出, 只有当 μ 或 $\nu \geq n$ 时, S 的前 $2n$ 行才不存在全 0 行, 这样才有可能使 S 的秩为 $2n$ 。因此, 被控对象参数 θ 闭环可辨识的条件是调节器的阶次 μ 或 ν 不小于被控对象的阶次 n 。

上述推导过程中假定被控对象的分子、分母均为 n 阶, 若被控对象的分子、分母的阶次分别为 n_b 和 n_a , 观察式 (9-12) 或式 (9-13) 的形式, 可得到更一般的最小二乘类辨识算法的闭环可辨识条件是调节器的阶次 μ 或 ν 的 2 倍应不小于被控对象分子、分母阶次的和 $n_a + n_b$ 。如果考虑到一般过程计算机控制系统均具有纯滞后时间 d , 则相应的闭环可辨识条件是 2μ 或 2ν 应不小于 $n_a + n_b - d$, 这与通常控制中纯滞后不利于控制的结论不同, 纯滞后的存在是有利于系统闭环辨识的。

在确定自校正调节器的结构时, 必须注意到闭环可辨识条件, 并以此来设计控制器的阶次。

3. 最小方差自校正调节器

自校正调节器适用于过程结构已知, 但参数缓慢变化的随机系统。这种自校正调节器因具有算法简单、用一台微型计算机就可实现的特点, 所以在某些工业过程控制中有一定实用价值。其工作原理是根据输入序列 $\{u(k)\}$ 和输出序列 $\{y(k)\}$ 的数据, 不断地对被控对象参数进行估计, 得出估计值 $\hat{\theta}$, 然后根据参数估计值 $\hat{\theta}$ 进行自校正调节器控制参数 θ_c 的计算, 如图 9-3 所示。最后再根据当时量测所得到的输出 $y(k)$ 算出下一步应有的控制作用 $u(k)$, 以控制被控对象使之达到满意的工作状态。随着生产过程的不断进行, 自校正调节器将不断进行采样、估计、修正和控制, 直到控制系统达到或接近最优, 这就是自校正控制的过程。

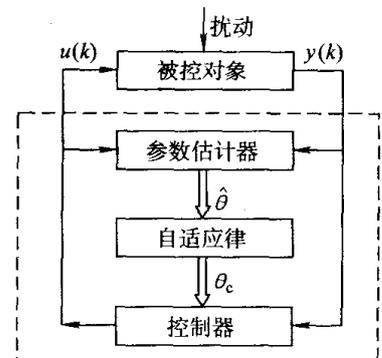


图 9-3 自校正调节器

(1) 自校正调节器的控制策略

假定被控对象可用下面的差分方程(9-14)表示。在一般情况下,可假设 $n_a = n_b = n_c = n$,这时称被控对象的模型是 n 阶的。

在工业过程控制问题中,要求系统的输出对其给定值的波动应尽可能小,对于上述控制系统就意味着要使输出的稳态方差尽可能小。所以,自校正调节器一般采用最小方差控制策略。

对由模型

$$y(k) = \frac{B(z^{-1})}{A(z^{-1})}u(k-d) + \frac{C(z^{-1})}{A(z^{-1})}w(k) \quad (9-14)$$

描述的被控对象,假设:

- 1) 被控对象的纯延迟时间 d ,以及多项式 $A(z^{-1})$ 、 $B(z^{-1})$ 、 $C(z^{-1})$ 的阶次和系数都是已知的;
- 2) 被控对象模型是最小相位系统,即多项式 $B(z^{-1})$ 的所有零点位于单位圆内;
- 3) 多项式 $C(z^{-1})$ 的所有零点位于单位圆内;
- 4) $\{w(k)\}$ 是均值为零,方差为 σ^2 的白噪声序列。

以下要确定在给定值恒定不变,也就是相当于 $r(k) = 0$ 的条件下使目标函数

$$J = E\{y^2(k+d)\} \quad (9-15)$$

为最小的最小方差控制律 $u^*(t)$ 。

现在假定所求的容许控制是输出序列 $y(k)$ 、 $y(k-1)$ 、...和输入序列 $u(k-1)$ 、 $u(k-2)$ 、...的线性函数。在控制和扰动的共同作用下,系统方程可写为

$$y(k+d) = \frac{B(z^{-1})}{A(z^{-1})}u(k) + \frac{C(z^{-1})}{A(z^{-1})}w(k+d) \quad (9-16)$$

式中, $\frac{C(z^{-1})}{A(z^{-1})}w(k+d)$ 可以展开成 $w(k+d)$ 、 $w(k+d-1)$ 、...、 $w(k+1)$ 、 $w(k)$ 、 $w(k-1)$ 、...等的线性组合, k 及 k 以前的随机变量 $w(k)$ 、 $w(k-1)$...与观测量 $y(k)$ 、 $y(k-1)$ 、...之间有一定的依赖关系,它们不是互相独立的,但 k 以后的随机变量 $w(k+1)$ 、 $w(k+2)$ 、...、 $w(k+d)$ 与 $y(k)$ 、 $y(k-1)$ 、...之间是独立的。现将 $\frac{C(z^{-1})}{A(z^{-1})}$ 分解为两部分,即

$$\frac{C(z^{-1})}{A(z^{-1})} = D(z^{-1}) + \frac{z^{-d}E(z^{-1})}{A(z^{-1})} \quad (9-17)$$

式中, $D(z^{-1})$ 为商式,而 $z^{-d}E(z^{-1})$ 则为余式。若 $D(z^{-1})$ 的阶次为 $d-1$,则 $D(z^{-1})w(k+d)$ 是随机向量 $w(k+1)$ 、 $w(k+2)$ 、...、 $w(k+d)$ 的线性组合。又由于 $E(z^{-1})$ 的阶次一定要比 $A(z^{-1})$ 的阶次小1,否则又能被 $A(z^{-1})$ 整除,所以 $E(z^{-1}) = e_0 + e_1z^{-1} + \dots + e_{n-1}z^{-(n-1)}$,可见 $z^{-d}E(z^{-1})w(k+d)$ 是随机量 $w(k)$ 、 $w(k-1)$ 、...、 $w(k-n+1)$ 的线性组合。把式(9-17)代入式(9-16)可得

$$y(k+d) = D(z^{-1})w(k+d) + \frac{E(z^{-1})}{A(z^{-1})}w(k) + \frac{B(z^{-1})}{A(z^{-1})}u(k) \quad (9-18)$$

又从式 (9-14) 知

$$w(k) = \frac{A(z^{-1})}{C(z^{-1})} y(k) - \frac{B(z^{-1})}{C(z^{-1})} z^{-d} u(k) \quad (9-19)$$

把式 (9-19) 代入式 (9-18) 可得

$$y(k+d) = D(z^{-1})w(k+d) + \left[\frac{B(z^{-1})}{A(z^{-1})} - \frac{B(z^{-1})E(z^{-1})}{A(z^{-1})C(z^{-1})} z^{-d} \right] u(k) + \frac{E(z^{-1})}{C(z^{-1})} y(k) \quad (9-20)$$

此式等号右边第二项为

$$\begin{aligned} \left[\frac{B(z^{-1})}{A(z^{-1})} - \frac{B(z^{-1})E(z^{-1})}{A(z^{-1})C(z^{-1})} z^{-d} \right] u(k) &= \frac{B(z^{-1})}{C(z^{-1})} \left[\frac{C(z^{-1})}{A(z^{-1})} - z^{-d} \frac{E(z^{-1})}{A(z^{-1})} \right] u(k) \\ &= \frac{B(z^{-1})}{C(z^{-1})} D(z^{-1}) u(k) \end{aligned} \quad (9-21)$$

代回原式可得

$$y(k+d) = D(z^{-1})w(k+d) + \frac{B(z^{-1})}{C(z^{-1})} D(z^{-1}) u(k) + \frac{E(z^{-1})}{C(z^{-1})} y(k) \quad (9-22)$$

把式 (9-22) 代入式 (9-15) 中可得

$$\begin{aligned} E\{y^2(k+d)\} &= E\left\{ \left[D(z^{-1})w(k+d) \right]^2 \right\} + E\left\{ \left[\frac{E(z^{-1})}{C(z^{-1})} y(k) + \frac{B(z^{-1})D(z^{-1})}{C(z^{-1})} u(k) \right]^2 \right\} \\ &\quad + 2E\left\{ D(z^{-1})w(k+d) \left[\frac{E(z^{-1})}{C(z^{-1})} y(k) + \frac{B(z^{-1})D(z^{-1})}{C(z^{-1})} u(k) \right] \right\} \end{aligned} \quad (9-23)$$

上面已经证明 $D(z^{-1})w(k+d)$ 是 $w(k+1), w(k+2), \dots, w(k+d)$ 的线性组合, 而由于 $w(k+1), w(k+2), \dots, w(k+d)$ 是零均值的, 式 (9-23) 成为

$$E\{y^2(k+d)\} = E\left\{ \left[D(z^{-1})w(k+d) \right]^2 \right\} + E\left\{ \left[\frac{E(z^{-1})}{C(z^{-1})} y(k) + \frac{B(z^{-1})D(z^{-1})}{C(z^{-1})} u(k) \right]^2 \right\} \quad (9-24)$$

式 (9-24) 的右边第一项与 $u(k)$ 无关, 若要使 $E\{y^2(k+d)\}$ 为最小, 只有使式 (9-24) 等号右边第二项为零, 即

$$u^*(k) = -\frac{E(z^{-1})}{B(z^{-1})D(z^{-1})} y(k) \quad (9-25)$$

此即为最小方差控制律。这时输出量的最小方差由式 (9-24) 可知为

$$E\{y^2(k+d)\} = E\left\{ \left[D(z^{-1})w(k+d) \right]^2 \right\}$$

$$\text{即} \quad E\{y^2(k+d)\} = (1 + d_1^2 + d_2^2 + \dots + d_{d-1}^2) \sigma^2 \quad (9-26)$$

式中, σ^2 是 $w(k)$ 的方差。在最小方差控制的控制条件下, 输出量的误差 $\bar{y}(k+d)$ 也就是 $y(k+d)$, 根据式 (9-22) 应为

$$\bar{y}(k+d) = y(k+d) = D(z^{-1})w(k+d)$$

$$\text{即} \quad \bar{y}(k+d) = w(k+d) + d_1 w(k+d-1) + \dots + d_{d-1} w(k+1) \quad (9-27)$$

(2) 自校正调节器调节系统的算法

在被控对象参数未知的条件下, 根据最小方差控制策略, 不难预料, 要获得较好的控制效果, 自校正控制策略的形式也应具有式 (9-25) 的形式, 即

$$u(k) = -\frac{E(z^{-1})}{B(z^{-1})D(z^{-1})}y(k)$$

若令 $E(z^{-1}) = G(z^{-1})$; $B(z^{-1})D(z^{-1}) = F(z^{-1})$, 而

$$G(z^{-1}) = g_0 + g_1z^{-1} + \cdots + g_{n_g}z^{-n_g}$$

$$F(z^{-1}) = f_0 + f_1z^{-1} + \cdots + f_{n_f}z^{-n_f}$$

则
$$u(k) = -\frac{G(z^{-1})}{F(z^{-1})}y(k) \quad (9-28)$$

根据隐式算法的原则, 被控对象的估计模型最好用所希望控制策略的参数作为自己的参数, 以被控系统的实际输入和输出作为自己的可用信息, 故被控对象的估计模型为

$$y(k+d) = (g_0 + g_1z^{-1} + \cdots + g_{n_g}z^{-n_g})y(k) + (f_0 + f_1z^{-1} + \cdots + f_{n_f}z^{-n_f})u(k) + w(k+d) \quad (9-29)$$

或
$$y(k+d) = G(z^{-1})y(k) + F(z^{-1})u(k) + w(k+d) \quad (9-30)$$

式中, $G(z^{-1})$ 和 $F(z^{-1})$ 的阶次分别用 $\deg G(z^{-1})$ 和 $\deg F(z^{-1})$ 表示, 且

$$\begin{cases} \deg G(z^{-1}) = n_g = n_a - 1 \\ \deg F(z^{-1}) = n_f = n_b + d - 1 \end{cases} \quad (9-31)$$

所以式 (9-31) 中需要估计的参数个数为

$$n_a - 1 + n_b + d - 1 = n_a + n_b + d - 2$$

若记

$$\begin{aligned} \theta &= [g_0, g_1, \cdots, g_{n_g}, f_1, \cdots, f_{n_f}]^T \\ \phi^T(k) &= [y(k), \cdots, y(k-n_g), u(k-1), \cdots, u(k-n_f)] \end{aligned}$$

则估计模型最后可表达为

$$y(k+d) = b_0u(k) + \phi^T(k)\theta + w(k+d)$$

或
$$y(k) - b_0u(k-d) = \phi^T(k-d)\theta + w(k) \quad (9-32)$$

式中, $b_0 = f_0$, 是由式 $F(z^{-1}) = D(z^{-1})B(z^{-1})$ 得到的。在闭环可辨识条件推导中知, 要使全部未知参数可辨识的条件之一是使多项式 $F(z^{-1})$ 的首项系数已知, 所以 b_0 可先假定为某一值。根据式 (9-32) 知, 只要控制策略相应变为

$$u^*(k) = -\frac{1}{b_0}\phi^T(k)\theta \quad (9-33)$$

把式 (9-33) 代入式 (9-32) 即可得到

$$y(k+d) = w(k+d)$$

因而, 实现了最小方差控制的要求。

根据最小二乘法, 就可对式 (9-32) 中未知参数向量 θ 进行估计, 引入记号

$$\hat{\theta} = [\hat{g}_0, \hat{g}_1, \dots, \hat{g}_{n_g}, \hat{f}_1, \dots, \hat{f}_{n_f}]^T \quad (9-34)$$

它代表参数向量 θ 的最小二乘估计, 则参数向量估计的递推公式为

$$\begin{cases} \hat{\theta}(k) = \hat{\theta}(k-1) + K(k)[y(k) - b_0 u(k-d) - \phi^T(k-d) \hat{\theta}(k-1)] \\ K(k) = \frac{P(k-1)\phi(k-d)}{\beta + \phi^T(k-d)P(k-1)\phi(k-d)} \\ P(k) = \frac{1}{\beta} [I - K(k)\phi^T(k-d)]P(k-1) \end{cases} \quad (9-35)$$

式中, β 为遗忘因子。

用上述递推公式所得的参数向量估计 $\hat{\theta}$ 代替式 (9-33) 中的参数 θ , 就得到了自校正调节器的控制策略

$$u^*(k) = -\frac{1}{b_0} \phi^T(k) \hat{\theta} \quad (9-36)$$

或

$$u^*(k) = -\frac{\hat{G}(z^{-1})}{\hat{F}(z^{-1})} y(k) \quad (9-37)$$

式中

$$\hat{G}(z^{-1}) = \hat{g}_0 + \hat{g}_1 z^{-1} + \dots + \hat{g}_{n_g} z^{-n_g}$$

$$\hat{F}(z^{-1}) = \hat{f}_0 + \hat{f}_1 z^{-1} + \dots + \hat{f}_{n_f} z^{-n_f}$$

综上所述, 自校正调节器算法包括以下几个步骤:

- 1) 通过采样获取新的观测输出 $y(k)$;
- 2) 组成观测数据向量 $\phi(k)$ 和 $\phi(k-d)$;
- 3) 用递推最小二乘估计式 (9-35) 计算最新参数向量 $\hat{\theta}$ 估计;
- 4) 用式 (9-36) 计算出当时所需的自校正控制策略 $u^*(k)$ 进行控制;
- 5) 使 $k \Rightarrow k+1$ 重复进行上面 4 个步骤, 直到达到最小方差控制 $y(k) = w(k)$ 为止。

(3) 最小方差自校正调节系统仿真例

假定被控对象是最小相位的, 带有白噪声的, 对象差分方程为

$$y(k) - 1.5y(k-1) + 0.7y(k-2) = u(k-1) + 0.5u(k-2) + e(k)$$

可知: $A(z^{-1}) = 1 - 1.5z^{-1} + 0.7z^{-2}$; $B(z^{-1}) = 1 + 0.5z^{-1}$; $d = 1$; $e(k)$ 白噪声用伪随机信号代替。

用上述 5 个步骤进行在线辨识和自校正调节, 假设 $b_0 = 1$, $y(1) = 1$, 遗忘因子 $\beta = 0.99$ 。而参数估计 $\hat{\theta}$ 为

$$\hat{\theta} = [\hat{g}_0, \hat{g}_1, \dots, \hat{g}_{n_g}, \hat{f}_1, \dots, \hat{f}_{n_f}]^T$$

式中, $n_g = 1$, $n_f = 1$ 。

仿真结果如图 9-4 所示。当递推次数达到 30 时, 参数估计收敛到 $\hat{g}_0 \rightarrow -1.5$, $\hat{g}_1 \rightarrow 0.7$, $\hat{f}_0 \rightarrow 0.5$, 而 $y(k) = e(k)$, 自校正调节系统达到最小方差控制。

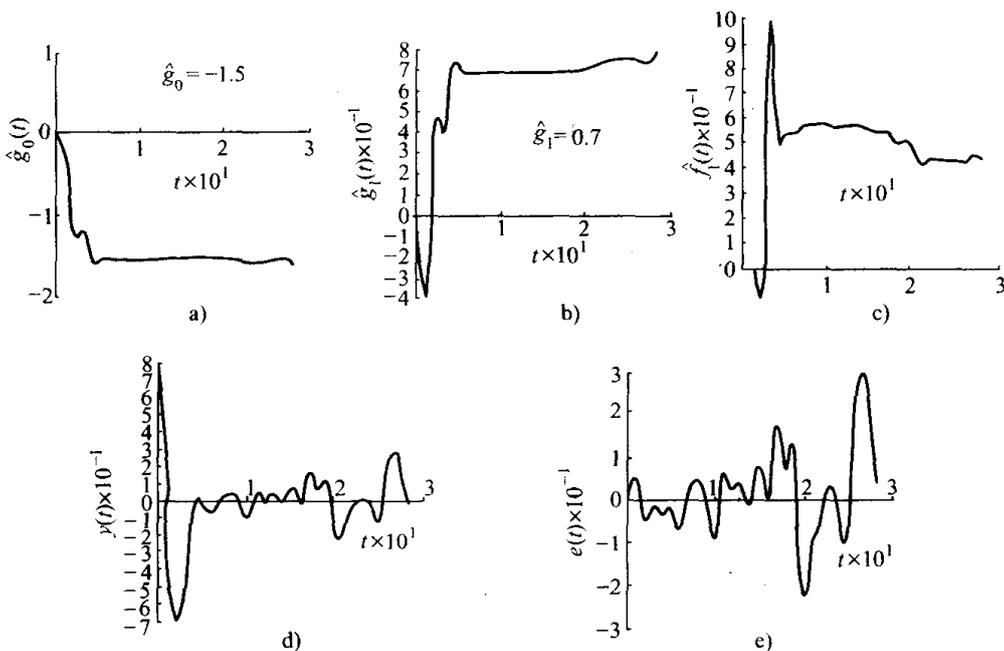


图 9-4 仿真结果

(4) 自校正调节器在应用中应注意的问题

自校正调节器因其原理简单，实现也比较方便，所以越来越多地用在工业过程控制中。它的控制质量比 PID 调节器高，但也存在着一些问题，需要进一步加以解决。

1) 控制信号可能过大。由式 (9-36)

$$u^*(k) = -\frac{1}{b_0} \phi^T(k) \theta$$

可知，当被控对象的增益 $|b_0|$ 过小时，最小方差调节器增益 $|1/b_0|$ 可能很大，这时为了达到输出方差最小，可能要求很大的控制信号，这在工程上是极不现实的。控制信号可能过大是最小方差调节器的致命缺点。

2) 对于非最小相位系统，采用自校正调节器还会带来控制系统的不稳定，即上述自校正调节器不适用于非最小相位系统。

4. 广义最小方差自校正控制器

(1) 对控制量加以约束的最小方差控制器

为了克服非最小相位系统对自校正调节器带来的不稳定，也为了克服自校正调节器可能出现的过大的控制信号，提出了广义最小方差控制策略。它要求把最小方差控制的性能指标 $J = E\{y^2(k+d)\} = \min$ ，改为广义最小方差控制指标

$$J = E\{y^2(k+d) + \Lambda u^2(k)\} = \min \tag{9-38}$$

式中， Λ 为对控制作用 $u(k)$ 的加权系数。和前面自校正调节器一样，也假定输出量的设定值 $y_r(k) = 0$ 。

与讨论最小方差控制时一样，也可以导出关系式

$$y(k+d) = D(z^{-1})w(k+d) + \frac{B(z^{-1})}{C(z^{-1})}D(z^{-1})u(k) + \frac{E(z^{-1})}{C(z^{-1})}y(k) \tag{9-39}$$

把式(9-39)代入式(9-38)可得

$$J = E \left\{ \left[D(z^{-1})w(k+d) + \frac{E(z^{-1})}{C(z^{-1})}y(k) + \frac{B(z^{-1})D(z^{-1})}{C(z^{-1})}u(k) \right]^2 + \lambda u^2(k) \right\} \quad (9-40)$$

式(9-40)右边括号内的 $D(z^{-1})w(k+d)$ 为 $k+1$ 时刻之后的信息,与右边其他项(属 k 及 k 以前的信息)是无关的,所以 J 可以分解为

$$J = E \left\{ \left[D(z^{-1})w(k+d) \right]^2 \right\} + E \left\{ \left[\frac{E(z^{-1})}{C(z^{-1})}y(k) + \frac{B(z^{-1})D(z^{-1})}{C(z^{-1})}u(k) \right]^2 \right\} + E \{ \lambda u^2(k) \}$$

为了求得使 J 最小的 $u(k)$,可求 J 关于 $u(k)$ 的导数,并令其为零。由于上式右边第一项与 k 时刻的 $u(k)$ 无关,所以对 $u(k)$ 求导数为零,因此

$$\frac{\partial J}{\partial u(k)} = 0 + 2 \left[\frac{E(z^{-1})}{C(z^{-1})}y(k) + \frac{B(z^{-1})D(z^{-1})}{C(z^{-1})}u(k) \right] b_0 + 2\lambda u(k) = 0$$

整理后可得

$$u(k) = - \frac{E(z^{-1})}{B(z^{-1})D(z^{-1}) + \frac{\lambda}{b_0}C(z^{-1})} y(k) \quad (9-41)$$

应该注意的是,在推导过程中用到 $u(k), u(k-1), u(k-2), \dots$ 之间是线性独立的假定,否则得不到式(9-41)的结果。根据式(9-41)可知被控对象组成如图9-5所示的框图。

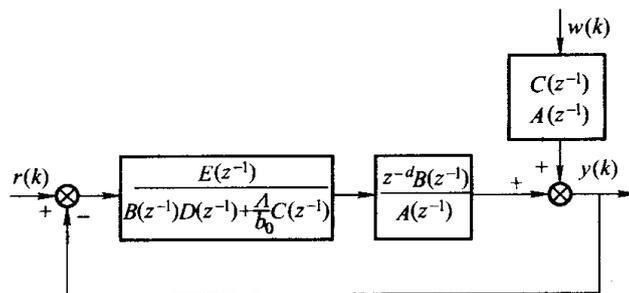


图9-5 广义最小方差控制系统

由闭环系统方块图获得闭环的特征方程为

$$1 + \frac{z^{-d} E(z^{-1}) B(z^{-1})}{\left[B(z^{-1}) D(z^{-1}) + \frac{\lambda}{b_0} C(z^{-1}) \right] A(z^{-1})} = 0$$

$$\text{即} \quad A(z^{-1}) B(z^{-1}) D(z^{-1}) + \frac{\lambda}{b_0} C(z^{-1}) A(z^{-1}) + z^{-d} E(z^{-1}) B(z^{-1}) = 0 \quad (9-42)$$

把式(9-17)代入式(9-42),经整理后得特征方程

$$C(z^{-1}) \left[B(z^{-1}) + \frac{\lambda}{b_0} A(z^{-1}) \right] = 0 \quad (9-43)$$

当 $\lambda = 0$ 时,可得一般最小方差控制的闭环特征方程为

$$C(z^{-1})B(z^{-1}) = 0 \quad (9-44)$$

比较式(9-43)和式(9-44)可知,当 $B(z^{-1})$ 是非最小相位时,也就是多项式 $B(z^{-1})$ 有零点在单位圆外或圆上时,由式(9-44)知,最小方差控制的闭环是不稳定的;但由式(9-43)看,尽管 $B(z^{-1})$ 是非最小相位的,广义最小方差控制系统却不一定是不稳定的。

当 $\Lambda \rightarrow 0$ 时,特征方程 $\left[B(z^{-1}) + \frac{\Lambda}{b_0} A(z^{-1}) \right] = 0$ 的零点趋于 $B(z^{-1})$ 的零点,只要 $B(z^{-1})$ 是非最小相位的,闭环系统就不稳定。当 $\Lambda \rightarrow \infty$ 时,特征方程 $\left[B(z^{-1}) + \frac{\Lambda}{b_0} A(z^{-1}) \right] = 0$ 的零点趋于 $A(z^{-1})$ 的零点,即趋于被控对象的极点。已知被控对象是稳定的, $A(z^{-1})$ 的零点在单位圆内,因此, Λ 增大, $\left[B(z^{-1}) + \frac{\Lambda}{b_0} A(z^{-1}) \right] = 0$ 的零点向 $A(z^{-1})$ 的零点趋近,即使 $B(z^{-1})$ 是非最小相位的,闭环系统仍可以稳定,这是广义最小方差控制的一个优点。

广义最小方差控制的另一个优点是当 $|b_0|$ 很小时,不会使控制量 $u(k)$ 像最小方差控制那样大。由式(9-41)等号右边可知,由于分母第一项 $B(z^{-1})$ 中的 b_0 与分母第二项的 b_0 ,对 $u(k)$ 的作用正好相反,因而 b_0 过小而引起控制量 $u(k)$ 过大的问题也解决了。

这种形式的广义方差控制策略尚需解决稳态偏差问题,因为采用式(9-38)的广义最小方差性能指标中含有 $\Lambda u^2(k)$ 项,稳态时必将使 $y^2(k+d)$ 产生某一数值,来补偿 $\Lambda u^2(k)$ 的值,而在稳态时 $y(k+d) \neq 0$,产生了稳态偏差。为了解决这个问题,又研究提出了具有辅助模型的广义最小方差自校正控制器。

(2) 带有辅助变量的最小方差控制器

为了减小稳态误差,除了对控制量 $u(k)$ 加以约束外,还可以输入一负的伺服信号 $\psi y_r(k)$,构成所谓辅助变量 $x(k)$,即

$$x(k) = \Gamma y(k) - \psi y_r(k-d) + \Lambda u(k-d) \quad (9-45)$$

或

$$x(k+d) = \Gamma y(k+d) - \psi y_r(k) + \Lambda u(k) \quad (9-46)$$

辅助变量 $x(k)$ 中的 Γ 、 Ψ 、 Λ 为加权系数,可以是常数也可以是多项式。此时性能指标可表示为

$$J = E \{ x^2(k+d) \} = \min \quad (9-47)$$

考虑到对象的模型 $y(k+d) = \frac{B(z^{-1})}{A(z^{-1})} u(k) + \frac{C(z^{-1})}{A(z^{-1})} w(k+d)$, 以及利用丢番图方程将扰动

$\frac{C(z^{-1})}{A(z^{-1})} w(k+d)$ 进行分解,与前面最小方差控制的推导一样,可得

$$y(k+d) = D(z^{-1})w(k+d) + \frac{B(z^{-1})}{C(z^{-1})} D(z^{-1})u(k) + \frac{E(z^{-1})}{C(z^{-1})} y(k) \quad (9-48)$$

代入式(9-46)可得

$$x(k+d) = \Gamma D(z^{-1})w(k+d) + \Gamma \frac{E(z^{-1})}{C(z^{-1})} y(k) + \Gamma \frac{B(z^{-1})}{C(z^{-1})} D(z^{-1})u(k) + \Lambda u(k) - \psi y_r(k) \quad (9-49)$$

考虑到式 (9-49) 等号右边第一项 $\Gamma D(z^{-1})w(k+d)$ 与其他各项不相关, 所以性能指标函数可以简化为

$$J = E\{x^2(k+d)\} = E\left\{\left[\Gamma D(z^{-1})w(k+d)\right]^2 + E\left\{\left[\Gamma \frac{E(z^{-1})}{C(z^{-1})}y(k) + \Gamma \frac{B(z^{-1})}{C(z^{-1})}D(z^{-1})u(k) + \Lambda u(k) - \psi y_r(k)\right]^2\right\}\right\} \quad (9-50)$$

使 J 为最小只需使式 (9-50) 等号右边可控的第二项为零, 即可得到广义最小方差控制律为

$$u(k) = -\frac{\Gamma E(z^{-1})y(k) - \psi C(z^{-1})y_r(k)}{\Gamma B(z^{-1})D(z^{-1}) + \Lambda C(z^{-1})} \quad (9-51)$$

下面进行稳定性分析和稳态误差的分析。

当式 (9-50) 等号右边可控的第二项为零时, 性能指标为

$$J = E\{x^2(k+d)\} = E\left\{\left[\Gamma D(z^{-1})w(k+d)\right]^2\right\} = \Gamma^2(1 + d_1^2 + \dots + d_{d-1}^2)\sigma^2$$

式中, σ^2 为白噪声序列 $\{w(k)\}$ 的方差。而根据定义, 辅助变量应为

$$x(k+d) = \Gamma y(k+d) - \psi y_r(k) + \Lambda u(k) = \Gamma D(z^{-1})w(k+d) \quad (9-52)$$

把式 (9-51) 代入到被控对象方程可得

$$y(k) = \frac{z^{-d}B(z^{-1})}{A(z^{-1})} \left\{ -\frac{\Gamma E(z^{-1})y(k) - \psi C(z^{-1})y_r(k)}{\Gamma B(z^{-1})D(z^{-1}) + \Lambda C(z^{-1})} \right\} + \frac{C(z^{-1})}{A(z^{-1})}w(k) \quad (9-53)$$

由式 (9-53) 可以构成广义最小方差的控制系统, 如图 9-6 所示。

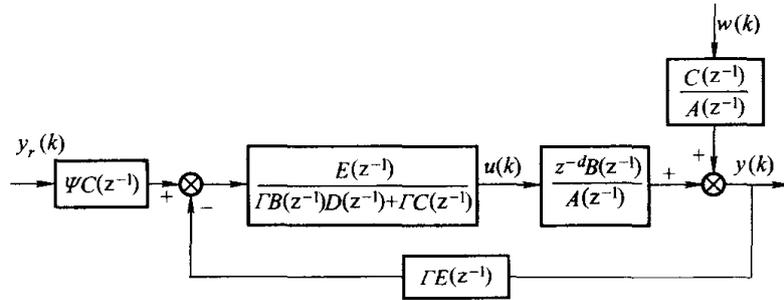


图 9-6 广义最小方差的控制系统

式 (9-53) 经整理后可得系统的闭环传递函数为

$$y(k) = \frac{z^{-d}B(z^{-1})\psi}{\Lambda A(z^{-1}) + \Gamma B(z^{-1})}y_r(k) + \frac{\Lambda C(z^{-1}) + \Gamma B(z^{-1})D(z^{-1})}{\Lambda A(z^{-1}) + \Gamma B(z^{-1})}w(k) \quad (9-54)$$

可知系统的闭环特征方程为

$$\Lambda A(z^{-1}) + \Gamma B(z^{-1}) = 0 \quad (9-55)$$

不管 $A(z^{-1})$ 、 $B(z^{-1})$ 是否是稳定多项式, 但只要适当地选择 Λ 和 Γ , 就可以使特征方程的根在单位圆内, 此时闭环系统稳定。

又由式 (9-54) 知系统在设定值扰动时的输出为

$$y(k) = \frac{z^{-d}B(z^{-1})\psi}{\Lambda A(z^{-1}) + \Gamma B(z^{-1})} y_r(k) \quad (9-56)$$

系统的稳态误差为

$$\begin{aligned} e_{ss} &= \lim_{k \rightarrow \infty} [y(k) - y_r(k)] = \lim_{\substack{k \rightarrow \infty \\ z \rightarrow 1}} \frac{z^{-d}B(z^{-1})\psi - \Lambda A(z^{-1}) - \Gamma B(z^{-1})}{\Lambda A(z^{-1}) + \Gamma B(z^{-1})} y_r(k) \\ &= \frac{B(1)\psi - \Lambda A(1) - \Gamma B(1)}{\Lambda A(1) + \Gamma B(1)} y_r(\infty) \end{aligned} \quad (9-57)$$

为使稳态误差 e_{ss} 为零, 只需令 $B(1)\psi - \Lambda A(1) - \Gamma B(1) = 0$, 在选定 Λ 、 Γ 后, 有

$$\psi = \frac{\Lambda A(1)}{B(1)} + \Gamma \quad (9-58)$$

(3) 带有辅助变量的自校正控制算法

$$\text{若令} \quad H(z^{-1}) = \psi C(z^{-1}) = h_0 + h_1 z^{-1} + \cdots + h_{n_h} z^{-n_h}$$

$$G(z^{-1}) = \Gamma B(z^{-1})D(z^{-1}) + \Lambda C(z^{-1}) = g_0 + g_1 z^{-1} + \cdots + g_{n_g} z^{-n_g}$$

$$M(z^{-1}) = \Gamma E(z^{-1}) = m_0 + m_1 z^{-1} + \cdots + m_{n_m} z^{-n_m}$$

则式 (9-51) 可以写为

$$G(z^{-1})u(k) + M(z^{-1})y(k) - H(z^{-1})y_r(k) = 0$$

利用式 (9-52) 可得

$$x(k+d) = G(z^{-1})u(k) + M(z^{-1})y(k) - H(z^{-1})y_r(k) + \Gamma D(z^{-1})w(k+d)$$

$$\text{或} \quad x(k+d) = \phi^T(k)\theta + w'(k+d) \quad (9-59)$$

式中, $\phi^T(k) = [y(k), \dots, y(k-n_m); u(k), \dots, u(k-n_g); y_r(k), \dots, y_r(k-n_h)]$

$$\theta^T = [m_0, \dots, m_{n_m}; g_0, \dots, g_{n_g}; h_0, \dots, h_{n_h}]$$

其中, $n_m = n_a - 1$ 。由于观测数据向量 $\phi^T(k)$ 与拟合误差 $\Gamma D(z^{-1})w(k+d)$ 不相关, 可直接用噪声信号 $w'(k+d)$ 表达, 所以参数估计是无偏的。

根据最小二乘原理, 则可得参数向量估计的递推公式为

$$\begin{cases} \hat{\theta}(k) = \hat{\theta}(k-1) + K(k)[x(k) - \phi^T(k-d)\hat{\theta}(k-1)] \\ K(k) = \frac{P(k-1)\phi(k-d)}{\beta + \phi^T(k-d)P(k-1)\phi(k-d)} \\ P(k) = \frac{1}{\beta} [I - K(k)\phi^T(k-d)]P(k-1) \end{cases} \quad (9-60)$$

控制律为

$$u(k) = \frac{\hat{H}(z^{-1})y_r(k) - \hat{M}(z^{-1})y(k)}{\hat{G}(z^{-1})} \quad (9-61)$$

$$\text{或} \quad u(k) = -\frac{1}{\hat{g}_0} \left[\hat{H}(z^{-1})y_r(k) - \hat{M}(z^{-1})y(k) + \sum_{i=1}^{n_g} \hat{g}_i u(k-i) \right] \quad (9-62)$$

与最小方差自校正调节器不同的是，广义最小方差自校正控制可以通过适当选择加权系数 λ 来限制控制信号 $u(k)$ 的幅值，并在非最小相位情况下保持系统稳定，另外也可以方便地处理设定值的问题。

5. 极点配置的自校正调节器

最小方差自校正调节器，由于估计误差的存在，可能造成非最小相位系统的不稳定；另外，当被控对象的增益很小时，最小方差自校正调节器所需要的控制作用又太大，利用极点配置方法设计自校正控制系统也可以克服上述这些缺点。

(1) 极点配置自校正调节系统基本原理

设对象（过程）方程为

$$y(k) = \frac{B(z^{-1})}{A(z^{-1})} z^{-d} u(k) + \frac{C(z^{-1})}{A(z^{-1})} w(k) \quad (9-63)$$

其中， $A(z^{-1}) = 1 + a_1 z^{-1} + \dots + a_{n_a} z^{-n_a}$ ， $B(z^{-1}) = b_0 + b_1 z^{-1} + \dots + b_{n_b} z^{-n_b}$ ；

$C(z^{-1}) = 1 + c_1 z^{-1} + \dots + c_{n_c} z^{-n_c}$ ， d 为纯迟延时间。

闭环控制系统如图 9-7 所示。

设调节器的传递函数为 $G(z^{-1})/F(z^{-1})$ ，其中

$$G(z^{-1}) = g_0 + g_1 z^{-1} + \dots + g_\nu z^{-\nu}, \quad \deg G(z^{-1}) = \nu$$

$$F(z^{-1}) = 1 + f_1 z^{-1} + \dots + f_\mu z^{-\mu}, \quad \deg F(z^{-1}) = \mu$$

由扰动 $w(k)$ 到输出 $y(k)$ 的闭环传递函数为

$$\frac{y(k)}{w(k)} = \frac{F(z^{-1})C(z^{-1})}{A(z^{-1})F(z^{-1}) + z^{-\tau}B(z^{-1})G(z^{-1})} = \frac{Q(z^{-1})}{P(z^{-1})} \quad (9-64)$$

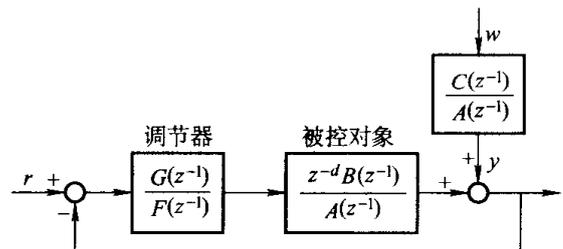


图 9-7 闭环控制系统

极点配置调节器的设计原则是使闭环传递函数等于所希望的形式，即 $Q(z^{-1})/P(z^{-1})$ ，多项式 $P(z^{-1}) = 1 + p_1 z^{-1} + \dots + p_{n_p} z^{-n_p}$ 的结构由设计者根据工艺的要求加以设计，以便使系统的极点分布在要求的位置上。对于 $Q(z^{-1})$ 可不作严格的规定，但为了使设计简化可取 $Q(z^{-1}) = F(z^{-1})$ ，这样可得

$$\frac{F(z^{-1})C(z^{-1})}{A(z^{-1})F(z^{-1}) + z^{-\tau}B(z^{-1})G(z^{-1})} = \frac{F(z^{-1})}{P(z^{-1})}$$

$$\text{即} \quad P(z^{-1})C(z^{-1}) = A(z^{-1})F(z^{-1}) + z^{-d}B(z^{-1})G(z^{-1}) \quad (9-65)$$

令式 (9-65) 等式两端 z^{-1} 的同次幂的系数相等，可以获得以 $\{g_0, g_1, \dots, g_\nu, f_1, f_2, \dots, f_\mu\}$ 为未知数的联立方程组。根据闭环系统的可辨识条件，要求调节器多项式的阶次 μ 或 ν 应不小于过程的阶次。取 $n_a = n_b = n_c = n$ ，且设

$$\begin{cases} \mu \geq n_b + d - 1 \\ v \geq n_a - 1 \end{cases} \quad (9-66)$$

式(9-65)的右边的阶应当为 $2n + d - 2$ ，而且右边的未知系统的个数亦为 $2n + d - 2$ 个（常数项为 1），因此只要方程(9-66)等于的左边的阶不超过右边的阶，则上述以 $g_0, g_1, \dots, g_v, f_1, f_2, \dots, f_\mu$ 为未知数的联立方程组就有解。为此，在设定多项式 $P(z^{-1})$ 时，应当保证 $P(z^{-1})$ 的阶 n_p 满足条件

$$n_p \leq n + d - 1 \quad (9-67)$$

(2) 极点配置自校正调节器的算法

在参数未知的情况下，根据自校正控制原理，首先应建立一个估计模型，然后由递推最小二乘法对该模型进行参数估计，并把所得参数估计值代替式(9-65)。

假设 $C(z^{-1}) = 1$ ，被控系统的模型可表示为

$$y(k) = [1 - A(z^{-1})]y(k) + z^{-d}B(z^{-1})u(k) + w(k) \quad (9-68)$$

定义

$$\theta = [a_1, a_2, \dots, a_{n_a}; b_0, b_1, \dots, b_{n_b}]^T$$

$$\phi^T(k) = [-y(k-1), -y(k-2), \dots, -y(k-n_a); u(k-d), u(k-d-1), \dots, u(k-d-n_b)]$$

则式(9-68)可以写成最小二乘格式

$$y(k) = \phi^T(k)\theta + w(k) \quad (9-69)$$

式中， $w(t)$ 为拟合误差； $\phi^T(t)$ 为观测数据向量。

利用最小二乘估计方法，就可对式(9-69)进行估计了，其递推公式如下：

$$\begin{cases} \hat{\theta}(k) = \hat{\theta}(k-1) + \mathbf{K}(k)[y(k) - \phi^T(k)\hat{\theta}(k-1)] \\ \mathbf{K}(k) = \frac{\mathbf{P}(k-1)\phi(k)}{\beta + \phi^T(k)\mathbf{P}(k-1)\phi(k)} \\ \mathbf{P}(k) = \frac{1}{\beta}[\mathbf{I} - \mathbf{K}(k)\phi^T(k)]\mathbf{P}(k-1) \end{cases} \quad (9-70)$$

用上述的递推公式所得的估计参数 $\hat{\theta}$ 即 $\hat{A}(z^{-1})$ 和 $\hat{B}(z^{-1})$ 代入极点配置公式(9-65)中可得

$$\hat{A}(z^{-1})\hat{F}(z^{-1}) + z^{-d}\hat{B}(z^{-1})\hat{G}(z^{-1}) = P(z^{-1}) \quad (9-71)$$

令式(9-71)两边同幂次的系数相等，再解所得联立方程组，求出 $\hat{F}(z^{-1})$ 和 $\hat{G}(z^{-1})$ 的各个系数，就得到极点配置自校正调节器的控制策略

$$u(k) = -\frac{\hat{G}(z^{-1})}{\hat{F}(z^{-1})}y(k) \quad (9-72)$$

这种方法是间接地估计出控制律的参数，故称为间接自校正控制。

由于实际系统 $C(z^{-1}) = 1$ 的条件不一定都成立，但根据自校正调节系统的自校正特性（这

里不作证明), 对于 $C(z^{-1}) \neq 1$ 的被控系统, 当采用上述极点配置自校正算法时, 虽然该算法是在 $C(z^{-1}) = 1$ 的条件下导出的, 但只要算法收敛, 就必定收敛于 $y(k) = \frac{F(z^{-1})}{P(z^{-1})} w(k)$ 的闭环特性, 这时拟合误差将等于系统的噪声 $w(k)$ 。

9.1.2 参考模型自适应控制系统

从最优控制理论可知, 当被控对象用式 (9-73) 线性定常状态方程描述时, 即

$$\dot{x} = Ax + Bu \tag{9-73}$$

式中, x 为被控对象的状态向量; u 是控制输入; A 和 B 是常数阵。可以找到一个最优控制 u , 使得二次性能指标

$$J = \int_0^t (x^T Q x + u^T R u) d\tau \tag{9-74}$$

为极小。式中, R 是正定阵; Q 至少为非负定阵。

但实际情况是: 能够正确地用线性定常微分方程式来描述的动态过程的数目是不多的, 只有在状态向量的稳定值附近, 才能用线性定常的假定对系统的动态特性进行描述; 二次性能指标 J 中矩阵 Q 和 R , 需要在现场整定中逐步确定; 系统的动态参数 A 和 B , 往往是未知的; 动态向量 x 的某些分量, 很难直接得到。

此外, 随着系统的维数越高, 选取 Q 和 R 变得越来越复杂, 用 Q 和 R 阵值来确定广泛采用的性能指标, 如上升时间、超调量和阻尼系数等是困难的和不直观的。但是, 如果这些指标用一个称为参考模型的理想化的控制系统的性能来规定, 就要容易得多。参考模型自适应控制系统, 就是基于上述原理发展起来的一种最优控制系统。

参考模型自适应控制系统的原理框图如图 9-8 所示, 由参考模型、可调系统和自适应律等三个部分组成。参考模型是一个理想化的系统模型, 按要求的性能指标预先设计好作为一个样板。它的输出值 (或状态) x_m 按要求的动态特性设计。可调系统是由实际被控对象、负反馈控制器所组成的闭环控制系统, 它的输出状态反映了当时可调系统的动态特性, 在自适应控制系统尚未调好时, 可调系统输出状态 x 与参考模型输出状态 x_m 是会有差别的, 差别的大小用状态广义误差 $e = x_m - x$ 表示。自适应律以一定的性能指标, 根据 e 的变化情况去调节可调系统, 最后保证

$$\lim_{t \rightarrow \infty} [x_m(t) - x(t)] = \lim_{t \rightarrow \infty} e = 0$$

根据自适应律的输出作用到达可调系统的不同位置, 分为可调参数自适应和信号综合自适应两种。当自适应律的输出去调整可调系统控制器参数时, 称为参数自适应控制; 而当自适应律的输出作为可调系统的一个辅助输入时, 则称为信号综合自适应控制。这两种情况的算法是不同的。

由于实际控制对象常常是时变的、非线性的系统, 要使 $\lim_{t \rightarrow \infty} e = 0$, 必须采用解决非线性系统的方法来设计

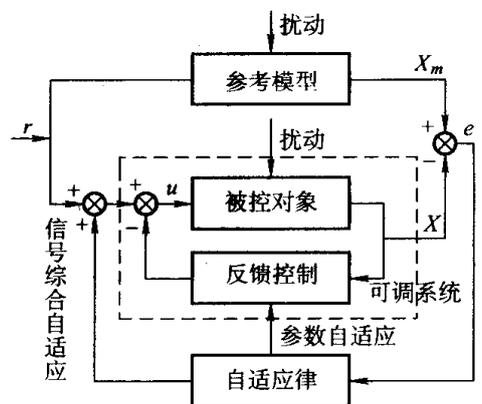


图 9-8 参考模型自适应控制系统

参考模型自适应控制系统。在设计参考模型自适应控制系统中，广泛应用了三种基本方法：局部参数最优化理论（梯度法、共轭梯度法等）、李雅普诺夫稳定性理论和超稳定性与正实性原理等。由于篇幅的限制，这里只介绍两种参考模型自适应控制系统的设计方法。

1. 被控对象全部状态能直接获取的自适应控制系统

- 假设：① 参考模型是一个理想化了的定常线性系统；
 ② 参考模型和可调系统的维数相同，并且维数是已知的；
 ③ 可调系统的所有参数对于自适应作用是可达的；
 ④ 状态广义误差 e 是可测量的；
 ⑤ 系统是单输入、单输出的。

被控对象的状态方程和观测方程表示为

$$\dot{x}_p(t) = A_p(t)x_p(t) + B_p(t)u(t) \tag{9-75}$$

$$y_p(t) = h_p(t)x_p(t) \tag{9-76}$$

式中， $A_p(t)$ 是 $n \times n$ 未知参数矩阵； $B_p(t)$ 是 $n \times 1$ 未知参数矩阵； $h_p(t)$ 是 $1 \times n$ 未知参数向量； $x_p(t)$ 是被控对象的状态向量，为 n 维的； $u(t)$ 是被控对象的输入控制作用，为标量；参考模型的状态方程和观测方程为

$$\dot{x}_m(t) = A_m x_m(t) + B_m r(t) \tag{9-77}$$

$$y_m(t) = h_m x_m(t) \tag{9-78}$$

式中， A_m 、 B_m 和 h_m 为给定的常数矩阵和向量，其阶次同 $A_p(t)$ 、 $B_p(t)$ 和 $h_p(t)$ ，且 A_m 、 B_m 是稳定矩阵， A_m 、 B_m 和 h_m 的选择应使参考模型的输出响应 $y_m(t)$ 实现所希望的特性； $x_m(t)$ 是参考模型的状态向量，为 n 维的； $y_m(t)$ 是参考模型的输出，是标量； $r(t)$ 是参考输入，也是标量。

由参考模型和可调系统组成的控制系统如图 9-9 所示。由图 9-9 知，如果 g 为可调系统的前馈增益， $r(t)$ 为参考输入， F 为可调系统的反馈增益向量。

由图 9-9 可得

$$u(t) = r(t)g - Fx_p(t) \tag{9-79}$$

$$\begin{aligned} \dot{x}_p(t) &= A_p(t)x_p(t) + B_p(t)u(t) \\ &= A_p(t)x_p(t) + B_p(t)[r(t)g - Fx_p(t)] \\ &= [A_p(t) - B_p(t)F]x_p(t) + B_p(t)gr(t) \end{aligned}$$

和式 (9-77) 比较，只有使

$$\begin{cases} A_p(t) - B_p(t)F = A_m \\ B_p(t)g = B_m \end{cases} \tag{9-80}$$

才有可能使可调系统对参考输入 $r(t)$ 的动态响应与参考模型一致。

参考模型自适应控制，就是要根据得到的有关对象和参考模型的信息（如 $x_p(t)$ 、 $x_m(t)$ 、 $r(t)$ 和 $e(t)$ 等），设计一个自适应律，使之自动地调节控制器的 F 和 g 的值，满足关系式 (9-80)。此外，要求参数 F 和 g 的调整过程是稳定的，即要求达到渐近稳定，也即

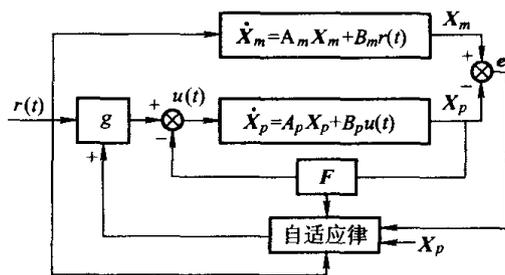


图 9-9 参考模型自适应控制系统

$$\text{当 } t \rightarrow \infty, \begin{cases} e(t) \rightarrow 0, \text{ 即 } x_p(t) \rightarrow x_m \\ A_p(t) - B_p(t)F \rightarrow A_m \\ B_p(t)g \rightarrow B_m \end{cases}$$

由于系统往往是非线性的，用李雅普诺夫稳定性理论和波波夫的超稳定性理论，是解决上述问题很有用的工具。

下面以简单的一阶系统为例来说明自适应控制系统的设计方法。

例 9-1 设被控对象的状态方程为

$$\dot{x}_p(t) = -a_p(t)x_p(t) + b_p(t)u(t) \quad (9-81)$$

所用的符号如前所述，所不同的是这里的变量都是标量。参考模型的状态方程为

$$\dot{x}_m(t) = -a_m x_m(t) + b_m r(t) \quad (9-82)$$

式中， $a_m > 0$ ， $b_m > 0$ ， $a_p(t)$ 决定系统的时间常数； $b_p(t)$ 相当于对象的放大倍数。如令 $g(t)$ 和 $f(t)$ 分别为可调系统的前馈增益和反馈增益。由式 (9-80) 知，只有调整 $g(t)$ 和 $f(t)$ ，使

$$\begin{cases} a_p(t) - b_p(t)f(t) = a_m \\ b_p(t)g(t) = b_m \end{cases} \quad (9-83)$$

才能使可调系统的动态响应与参考模型一致。

假设参数 $a_p(t)$ 和 $b_p(t)$ 的变化比可调系统和参考模型动态响应过渡过程时间要慢，也比 $g(t)$ 和 $f(t)$ 自适应调整过程慢得多，那么在调整过程中认为 $a_p(t)$ 和 $b_p(t)$ 是常数，并用 a_p 和 b_p 表示。可以导出

$$\dot{x}_p(t) = -[a_p + b_p f(t)]x_p(t) + b_p g(t)r(t) \quad (9-84)$$

因广义状态偏差

$$e(t) = x_m(t) - x_p(t) \quad (9-85)$$

可得

$$\begin{aligned} \dot{e}(t) &= \dot{x}_m(t) - \dot{x}_p(t) \\ &= -a_m x_m + b_m r(t) + [a_p + b_p f(t)]x_p(t) - b_p g(t)r(t) \\ &= -a_m e(t) - [a_m - a_p - b_p f(t)]x_p(t) + [b_m - b_p g(t)]r(t) \end{aligned} \quad (9-86)$$

令

$$\phi(t) = a_m - a_p - b_p f(t) \quad (9-87)$$

$$\varphi(t) = b_m - b_p g(t) \quad (9-88)$$

把式 (9-87) 和式 (9-88) 代入式 (9-86)，得

$$\dot{e}(t) = -a_m e(t) - \phi(t)x_p(t) + \varphi(t)r(t) \quad (9-89)$$

根据式 (9-83) 和 $\lim_{t \rightarrow \infty} e = 0$ 的要求，自适应的任务是设计一个参数 $f(t)$ 和 $g(t)$ 的自适应调整律，使式 (9-89) 渐近稳定，即要求

当 $t \rightarrow \infty$ 时， $e(t) \rightarrow 0$ ； $\phi(t) \rightarrow 0$ 和 $\varphi(t) \rightarrow 0$

李雅普诺夫渐近稳定定理：如果在包含原点在内的某个域 S 内，存在李雅普诺夫函数 $V(x, t) > 0$ ，该函数对时间的导数 $\dot{V}(x, t) \leq 0$ ，则一切由初始状态 $x(0) \neq 0$ 出发的运动轨线，

将收敛于 $\dot{V}(x, t) \equiv 0$ 的 x 的某一不变子集 M (即指状态空间中满足 $\dot{V}(x, t) \equiv 0$ 的状态 x 的集合)。不难想像, 当 M 缩小成状态空间的原点时, 系统对原点这个平衡状态就是渐近稳定的。

令 Σ 为包含变量 e 、 ϕ 、 φ 的一个三维偏差空间, 设 $\varepsilon^T = (e, \phi, \varphi)$ 为空间 Σ 的向量。选择李雅普诺夫函数为

$$V(\varepsilon) = \frac{1}{2} \left(b_p e^2 + \frac{1}{\lambda_1} \phi^2 + \frac{1}{\lambda_2} \varphi^2 \right) \quad (9-90)$$

式中, $\lambda_1 > 0$, $\lambda_2 > 0$, 设 $b_p > 0$, 则

$$V(\varepsilon) > 0 \quad \text{对所有 } \varepsilon \neq 0$$

而

$$\dot{V}(\varepsilon) = b_p e \dot{e} + \frac{1}{\lambda_1} \phi \dot{\phi} + \frac{1}{\lambda_2} \varphi \dot{\varphi} \quad (9-91)$$

用式 (9-89) 代入式 (9-91) 可得

$$\dot{V}(\varepsilon) = -b_p a_m e^2 + \phi \left(\frac{1}{\lambda_1} \dot{\phi} - b_p e x_p(t) \right) + \varphi \left(\frac{1}{\lambda_2} \dot{\varphi} + b_p e r(t) \right)$$

如果选择

$$\begin{cases} \dot{\phi} = \lambda_1 b_p e x_p(t) \\ \dot{\varphi} = -\lambda_2 b_p e r(t) \end{cases} \quad (9-92)$$

则

$$\dot{V}(\varepsilon) = -b_p a_m e^2 \leq 0 \quad b_p > 0, a_m > 0 \quad (9-93)$$

这时, 因为 $V(\varepsilon) > 0$, $\dot{V}(\varepsilon) \leq 0$, 只要按式 (9-92) 决定参数 $\phi(t)$ 和 $\varphi(t)$, 就可保证偏差方程 (9-89) 是稳定的。若要使式 (9-89) 的偏差方程渐近稳定, 尚需要附加一些条件。根据李雅普诺夫渐近稳定定理, 一切由初始状态 $\varepsilon(0)$ 出发的轨线, 都将收敛到不变集合 M (即使 $\dot{V}(\varepsilon, t) \equiv 0$ 的 ε 的集合), 有

$$M = \{ \varepsilon : \dot{V}(\varepsilon) \equiv 0 \} = \{ \varepsilon : e \equiv 0 \} = \{ \varepsilon : e = 0, \dot{e} = 0 \} = \{ \varepsilon : e = 0, -\phi x_p + \varphi r = 0 \}$$

当 M 缩小到原点, 即由 $-\phi x_p + \varphi r = 0$ 所决定的平面缩小到原点, 偏差方程 (9-89) 才能渐近稳定。实际上, 由于 $r(t)$ 与 x_p 是线性无关的, 要使 $-\phi x_p + \varphi r = 0$ 成立, 必须是 $\phi = 0$, $\varphi = 0$ 才行。这样, 误差空间 Σ 中由 $\varepsilon(0) \neq 0$ 出发的轨线, 将收敛到 Σ 空间原点 ($e = 0$, $\phi = 0$, $\varphi = 0$), 微分方程 (9-89) 达到渐近稳定。

由式 (9-87) 和式 (9-88) 对时间求导数, 可得

$$\dot{\phi}(t) = -b_p(t) \dot{f}(t)$$

$$\dot{\varphi}(t) = -b_p(t) \dot{g}(t)$$

把它代入式 (9-92), 得到

$$\begin{cases} \dot{f}(t) = -\lambda_1 e(t) x_p(t) \\ \dot{g}(t) = \lambda_2 e(t) r(t) \end{cases} \quad (9-94)$$

式 (9-94) 即为可调系统反馈增益 $f(t)$ 和前馈增益 $g(t)$ 的自适应调整律。利用这个规律调整可调系统, 最后可达到, 当 $t \rightarrow \infty$ 时,

$$e = 0$$

$$\phi(t) = 0, \text{ 即 } a_p + b_p f(t) = a_m$$

$$\varphi(t) = 0, \text{ 即 } b_p g(t) = b_m$$

可调系统输出的动态响应和参考模型输出一致。

2. 根据对象输入、输出设计参考模型自适应控制系统

被控对象的全部状态要准确得到常常是很困难的, 因此按被控对象输入和输出直接设计自适应控制系统, 自然会引起人们的很大兴趣。这个问题的研究正朝着两个方向努力, 一个称为直接法, 它是根据对象的输入、输出直接设计自适应律, 以调节某一特定结构控制器的可调参数, 使由被控对象和控制器组成的可调系统的传递函数与给定的参考模型的传递函数完全匹配; 另一个方向是利用被控对象的输入、输出设计一个自适应观测器, 实时地给出被控对象未知参数和状态的估计, 然后利用这些估计值综合控制作用, 使被控对象的输出准确地跟踪参考模型的输出, 或者使某一性能指标优化, 这种方法称为间接法。这里仅讨论直接法。

(1) 设计问题的提法

设被控对象是一个线性时不变、单输入单输出的动态系统, 假定其结构已知而参数未知, 具体地说就是其传递函数的分子和分母多项式的阶已知, 但多项式的系数未知, 对象的动态特性可由以下状态方程来描述:

$$\dot{x}_p = A_p x_p + B_p u$$

$$y_p = h^T x_p$$

式中, A_p 是 $n \times n$ 常数矩阵; B_p , h 是 $n \times 1$ 维常向量; u , y_p 分别表示对象的输入和输出, 它们都是标量。

对象的传递函数可以写成

$$W_p(s) = h^T (sI - A_p)^{-1} b_p = k_p \frac{Z_p(s)}{R_p(s)}$$

式中, $R_p(s)$, $Z_p(s)$ 分别是 s 的 n 和 m 阶多项式, 其首项系数均为 1, 另外设 $m \leq n-1$; k_p 是对象的增益, 假定其符号已知, 数值未知。

根据对控制系统动态性能的要求选择一个参考模型, 模型的输入为系统的参考输入 $r(t)$, 模型的输出 $y_m(t)$ 代表了希望的系统输出响应, 参考模型的传递函数可以写成以下形式

$$W_m(s) = k_m \frac{Z_m(s)}{R_m(s)}$$

式中, $R_m(s)$, $Z_m(s)$ 分别是 s 的 n 阶和 r 阶多项式, $r \leq n$, 为简单起见, 以下选择模型与对象的结构相同, 即 $r = m$ 。

为了保证控制器可靠工作, 不致放大外来的噪声干扰, 希望尽量不采用微分器, 因此, 当控制器和被控对象组成可调系统时的极点个数与零点个数之差不小于 $n-m$, 这样才能保证控制器中不包含纯微分环节。

设对象输出与模型输出的偏差为 $e_1(t)$, 即 $e_1(t) = y_p(t) - y_m(t)$, 自适应控制问题可归结为:

已知被控对象传递函数的结构 (分母和分子的阶 n, m) 和输入、输出数据组 $\{u(t), y_p(t)\}$; 参考模型的结构 (与被控对象相同) 和输入、输出数据组 $\{r(t), y_m(t)\}$ 。

要求设计一个自适应控制器，使得

$$\lim_{t \rightarrow \infty} |e_1(t)| = \lim_{t \rightarrow \infty} |y_p(t) - y_m(t)| = 0$$

在设计时还附加一个条件，就是不容许应用微分器。或者说，控制器传递函数分子的阶不能大于分母的阶，否则认为系统在物理上是不能实现的。在设计自适应控制系统时，有两个根本问题需要很好地解决。

第一个问题是要解决控制器的存在性问题，即要考虑一个合理的控制器结构，在这个结构下，能够保证组成一个与参考模型完全匹配的可调系统。如果令可调系统的传递函数为 $W(s)$ ，参考模型的传递函数为 $W_m(s)$ ，那么要求控制器在设定的结构下存在一组参数 θ^* ，使得可调系统的传递函数 $W(s) = W_m(s)$ 。

第二个问题是自适应律的设计问题，即根据可以获得的输入、输出设计一个自适应律，自动调节控制器的参数，使得在对象参数未知或者参数发生变化的条件下，还能使对象输出与模型输出一致。

(2) 控制器的结构

为了实现可调系统与参考模型的完全匹配，首先，自适应控制器必须具有足够多的可调参数。假定被控对象传递函数分母为 n 阶，分子为 $n-1$ 阶，那么对象的未知参数最多可能有 $2n$ 个，其中分母有 n 个系数，分子有 $n-1$ 个系数再加上增益 k_p 。因此，控制器也应当有 $2n$ 个可调参数与之对应。假定整个控制器是一个多输入和单输出的动态系统，其结构如图 9-10 所示，其中控制器的输入包括参考输入 $r(t)$ 、对象输入 $u(t)$ 、对象输出 $y_p(t)$ ，这些信号都是可以测量的，控制器的输出就是对象的控制输入 $u(t)$ ，控制器内部有两个辅助信号发生器，又称为状态滤波器，分别接在对象的输入端和输出端，辅助信号发生器输出的信号以及对象的输出信号 $y_p(t)$ ，分别经过可调增益 c^T 、 d^T 、 d_0 综合成子系统 F_1 、 F_2 的输出信号 $w^{(1)}$ 和 $w^{(2)}$ 。信号 $w^{(1)}$ 和 $w^{(2)}$ 再加上由可调增益 k_0 输出的信号 $k_0 r(t)$ ，共同组成控制输入 $u(t)$ ，加到对象入口。

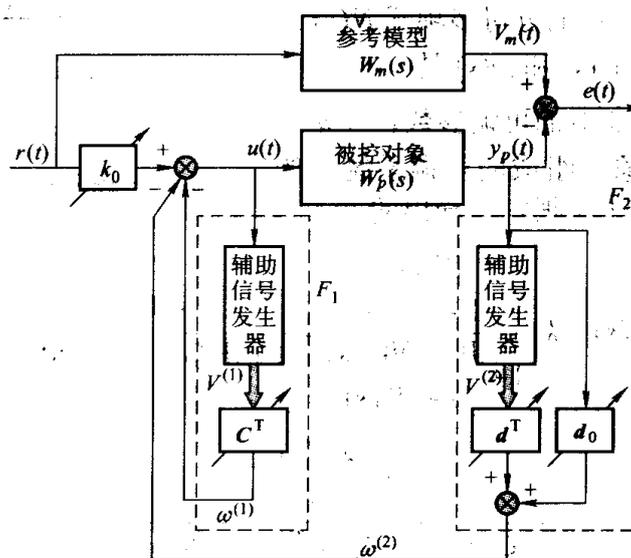


图 9-10 自适应控制系统结构

由辅助信号发生器组成的子系统 F_1 、 F_2 实质上是两个 $(n-1)$ 维的动态系统。 F_1 、 F_2 的

状态空间描述如下:

$$\text{对 } F_1: \begin{cases} \dot{\mathbf{v}}^{(1)} = \mathbf{A}\mathbf{v}^{(1)} + \mathbf{b}u \\ \mathbf{w}^{(1)} = \mathbf{c}^T \mathbf{v}^{(1)} \end{cases}$$

$$\text{对 } F_2: \begin{cases} \dot{\mathbf{v}}^{(2)} = \mathbf{A}\mathbf{v}^{(2)} + \mathbf{b}y_p \\ \mathbf{w}^{(2)} = d_0 y_p + \mathbf{d}^T \mathbf{v}^{(2)} \end{cases}$$

其中, $\mathbf{v}^{(1)}$ 、 $\mathbf{v}^{(2)}$ 分别表示 F_1 、 F_2 的状态向量; $w^{(1)}$ 、 $w^{(2)}$ 分别表示 F_1 、 F_2 的输出, 都是标量。

$\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ \dots & \dots \\ -l_{n-1} & \dots & -l_1 \end{bmatrix}$ 为 $(n-1) \times (n-1)$ 稳定矩阵, 所谓稳定矩阵是指特征值都在左半平面的矩阵。

$$\mathbf{b}^T = [0, \dots, 0, 1]$$

$$\mathbf{c}^T = [c_1(t), \dots, c_{n-1}(t)]$$

$$\mathbf{d}^T = [d_1(t), \dots, d_{n-1}(t)]$$

\mathbf{c}^T 、 \mathbf{d}^T 是控制器中两个可调参数向量, 此外还有两个可调参数 d_0 、 k_0 , 为了书写简便, 控制器中 $2n$ 个可调参数用向量 $\boldsymbol{\theta}(t)$ 表示, 即

$$\boldsymbol{\theta}(t) = [k_0(t), \mathbf{c}^T(t), d_0(t), \mathbf{d}^T(t)]^T$$

当 $\boldsymbol{\theta}(t)$ 为某一常向量时, 子系统 F_1 、 F_2 的传递函数 $W_1(s)$ 、 $W_2(s)$ 分别可以写成

$$W_1(s) = \mathbf{c}^T (s\mathbf{I} - \mathbf{A})^{-1} \mathbf{b} = \frac{C(s)}{N(s)}$$

$$W_2(s) = d_0 + \mathbf{d}^T (s\mathbf{I} - \mathbf{A})^{-1} \mathbf{b} = \frac{D(s)}{N(s)} + d_0$$

其中, 传递函数 $W_1(s)$ 、 $W_2(s)$ 的分母多项式 $N(s)$ 为 s 的 $n-1$ 阶多项式, 分子 $C(s)$ 和 $D(s)$ 是 s 的 $n-2$ 阶多项式, 其系数就是控制器的可调参数 \mathbf{c}^T 、 \mathbf{d}^T 。

由控制器和被控对象所组成的可调系统的传递函数 $W(s)$, 利用图 9-11 是很容易导出的。

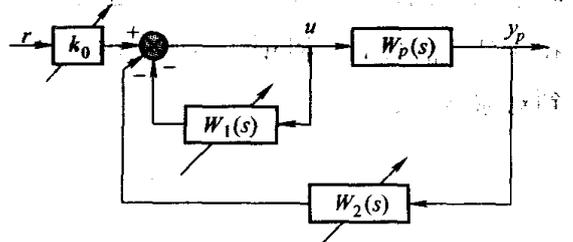


图 9-11 可调系统的结构

$$\begin{aligned} W(s) &= k_0 \frac{W_p(s)}{1 + W_1(s)} = \frac{k_0 W_p(s)}{1 + \frac{W_p(s)W_2(s)}{1 + W_1(s)}} \\ &= \frac{k_0 k_p \frac{Z_p(s)}{R_p(s)}}{1 + \frac{C(s)}{N(s)} + k_p \frac{Z_p(s)}{R_p(s)} \left[d_0 + \frac{D(s)}{N(s)} \right]} \\ &= \frac{k_0 k_p Z_p(s) N(s)}{[N(s) + C(s)] R_p(s) + k_p Z_p(s) [D(s) + d_0 N(s)]} \end{aligned}$$

为了实现可调系统与参考模型的完全匹配, 要求 $W(s) = W_m(s)$, 为此, 只要选择:

- 1) $N(s) = Z_m(s)$
- 2) $[N(s) + C(s)]R_p(s) + k_p Z_p(s)[D(s) + d_0 N(s)] = R_m(s)Z_p(s)$
- 3) $k_0 = k_m / k_p$

显然, 实现上述条件中的①、③是不成问题的, 但能否实现其中的②项就不那么明显了。为此, 需要应用以下定理。

定理 设 $R(s)$ 和 $Z(s)$ 为两个互质多项式, $R(s)$ 为 n 阶, $Z(s)$ 不大于 $n-1$ 阶, 则只要适当选择另外两个 $n-1$ 阶多项式 $\alpha_1(s)$ 、 $\alpha_2(s)$, 总可以使和式 $\alpha_1(s)R(s) + \alpha_2(s)Z(s)$ 为任意 $2n-1$ 阶多项式。

应用这个定理, 其中 $[N(s) + C(s)]$ 和 $[D(s) + d_0 N(s)]$ 都是 $n-1$ 阶可调多项式, 相当于定理中的 $\alpha_1(s)$ 、 $\alpha_2(s)$, 又设 $R_p(s)$ 和 $Z_p(s)$ 分别为 n 阶和 $n-1$ 阶多项式, 相当于定理中的 $R(s)$ 和 $Z(s)$, 因此条件②的左端可以构成任意 $2n-1$ 阶多项式。再看条件②的右端 $R_m(s)Z_p(s)$ 正好是 $2n-1$ 阶, 因此, 条件②是完全能够实现的。

综上所述, 要实现可调系统与参考模型传递函数的完全匹配, 应当要求:

1) 被控对象传递函数分子多项式 $Z_p(s)$ 是 $n-1$ 阶稳定多项式, 如果 $Z_p(s)$ 不是稳定多项式, 就不能实现 $W(s)$ 中关于 $Z_p(s)$ 的零极相消。也就是说, 在控制器中将出现不稳定极点, 这当然是不希望的。

2) 辅助信号发生器传递函数的分母多项式 $N(s)$ 应当选择等于 $Z_m(s)$, 由于要求 $N(s)$ 是稳定多项式, 所以 $Z_m(s)$ 也应当是稳定多项式。当然, 参考模型传递函数的分母多项式 $R_m(s)$ 也必须是稳定多项式, 这是保证系统具有希望的动态性能的起码要求。关于对参考模型传递函数进一步的要求, 将在后面讨论自适应律的设计时再逐步提出。

通过以上的分析, 集中说明了一个事实, 即采用图 9-10 所示的控制器结构, 完全能够保证实现可调系统与参考模型传递函数的完全匹配, 这样也就解决了关于自适应控制系统设计中的第一个基本问题, 即控制器的存在问题。

(3) 自适应律的设计

为了应用李雅普诺夫稳定理论, 设计控制器可调参数的自适应调整律, 首先需要将参考模型和可调系统用状态空间加以表达, 找出两者状态偏差的动态方程, 进而求出输出偏差的动态方程, 自适应律的设计就是要使系统与模型的输出偏差渐近稳定。

令 $w(t)$ 表示可调系统中的信号向量, 即

$$w^T(t) = [r(t), v^{(1)}(t), y_p(t), v^{(2)}(t)]$$

令 $\theta(t)$ 表示可调系统中的可调参数向量, 即

$$\theta(t) = [k_0(t), c^T(t), d_0(t), d^T(t)]^T$$

那么, 被控对象的输入控制作用 $u(t)$ 可以写成

$$u(t) = \theta^T(t)w(t)$$

因此, 整个可调系统可用以下状态方程组来描述

$$\dot{x}_p = Ax_p + b_p u$$

$$\dot{v}^{(1)} = \Lambda v^{(1)} + bu$$

$$\dot{v}^{(2)} = \Lambda v^{(2)} + bh^T x_p$$

$$y_p = h^T x_p$$

令增广状态向量 $x = \begin{bmatrix} x_p \\ v^{(1)} \\ v^{(2)} \end{bmatrix}$ ，则上述方程组可简写成向量矩阵形式

$$\begin{bmatrix} \dot{x}_p \\ \dot{v}^{(1)} \\ \dot{v}^{(2)} \end{bmatrix} = \begin{bmatrix} A_p & 0 & 0 \\ 0 & \Lambda & 0 \\ bh^T & 0 & \Lambda \end{bmatrix} \cdot \begin{bmatrix} x_p \\ v^{(1)} \\ v^{(2)} \end{bmatrix} + \begin{bmatrix} b_p \\ b \\ 0 \end{bmatrix} \theta^T w(t) \quad (9-95)$$

令 $\theta(t) = \theta^* + \phi(t)$ ，其中 θ^* 表示可调系统与参考模型完全匹配时的一组参数。 $\phi(t)$ 表示参数向量 $\theta(t)$ 中与参考模型失配的部分。即如果模型完全匹配，则 $\phi(t) = 0$ ，所以 $u(t)$ 又可简化成：

$$\begin{aligned} u(t) &= \theta^T(t) w(t) \\ &= [\theta^* + \phi(t)]^T w(t) \\ &= k_0^* r + c^{*T} v^{(1)} + d_0^* h x_p + d^{*T} v^{(2)} + \phi^T(t) w(t) \end{aligned}$$

将这个结果代入到可调系统的状态方程式 (9-95) 中，经过整理得以下增广状态方程，即

$$\dot{x} = A_c x + b_c [k_0^* r + \phi^T(t) w(t)] \quad (9-96)$$

其中

$$\begin{aligned} x^T &= [x_p^T, v^{(1)T}, v^{(2)T}] \\ A_c &= \begin{bmatrix} A_p + d_0^* b_p h^T & b_p c^{*T} & b_p d^{*T} \\ b d_0^* h & \Lambda + b c^{*T} & b d^{*T} \\ b h^T & 0 & \Lambda \end{bmatrix} \\ b_c &= \begin{bmatrix} b_p \\ b \\ 0 \end{bmatrix} : (3n-2) \text{ 维向量} \end{aligned}$$

$$b^T = [0, \dots, 0, 1] : (n-1) \text{ 维向量}$$

注意，当 $\phi(t) = 0$ 时， $\theta(t) = \theta^*$ 。因此，式 (9-96) 也可以看成是参考模型的状态方程，如果令 x_{mc} 表示参考模型的状态向量，即

$$x_{mc}^T = [x_m^T, v_m^{(1)T}, v_m^{(2)T}]$$

x_{mc} 有 $3n-2$ 维与 x 的维数相对应，参考模型的状态方程可写成

$$\begin{aligned} \dot{x}_{mc} &= A_c x_{mc} + b_c k_0^* r \\ y_m &= h_c^T x_{mc} = h^T x_m \end{aligned} \quad (9-97)$$

式中， $h_c^T = [h^T, 0, 0]$ 。

由于被控对象参数未知，即 A_c 、 b_c 未知，因此，不能利用式 (9-97) 来决定参考模型，但是

利用式(9-97)可以很方便地导出状态偏差的动态方程,为此,将式(9-96)与式(9-97)相减,再令:

$$\mathbf{e} = \mathbf{x} - \mathbf{x}_{mc} = \begin{bmatrix} \mathbf{x}_p - \mathbf{x}_m \\ \mathbf{v}^{(1)} - \mathbf{v}_m^{(1)} \\ \mathbf{v}^{(2)} - \mathbf{v}_m^{(2)} \end{bmatrix}$$

则状态偏差 \mathbf{e} 的微分方程可写成

$$\dot{\mathbf{e}} = \mathbf{A}_c \mathbf{e} + \mathbf{b}_c [\boldsymbol{\phi}^T(t) \mathbf{w}(t)]$$

其中
$$\mathbf{b}_c = \begin{bmatrix} \mathbf{b}_p \\ \mathbf{b} \\ \mathbf{0} \end{bmatrix}$$

令系统输出与模型输出的偏差为 e_1 ,即

$$e_1 = y_p - y_m$$

则
$$e_1 = \mathbf{h}_c^T \mathbf{e}$$

其中
$$\mathbf{h}_c^T = [\mathbf{h}^T, \mathbf{0}, \mathbf{0}]$$

如果把 $\mathbf{u}(t) = \boldsymbol{\phi}^T(t) \mathbf{w}(t)$ 看成是偏差模型的输入, e_1 看成是偏差模型的输出, \mathbf{e} 看成是偏差模型的状态,则偏差模型的状态方程、输出方程、传递函数分别表示如下:

$$\dot{\mathbf{e}} = \mathbf{A}_c \mathbf{e} + \mathbf{b}_c [\boldsymbol{\phi}^T(t) \mathbf{w}(t)]$$

$$e_1 = \mathbf{h}_c^T \mathbf{e}$$

$$W_e(s) = \mathbf{h}_c^T (s\mathbf{I} - \mathbf{A}_c)^{-1} \mathbf{b}_c$$

根据式(9-97)可求得参考模型的传递函数 $W_m(s)$ 如下:

$$W_m(s) = \mathbf{h}_c^T (s\mathbf{I} - \mathbf{A}_c)^{-1} \mathbf{b}_c k_0^* = W_e(s) k_0^*$$

已知: $k_0^* k_p = k_m$

所以有
$$W_e(s) = \frac{k_p}{k_m} W_m(s) \quad (9-98)$$

偏差模型也可以用图9-12所示的结构表示。

其中的输入信号是多变量,即 $\mathbf{w}(t) = [r(t), \mathbf{v}^{(1)}(t), y_p(t), \mathbf{v}^{(2)}(t)]^T$, $\boldsymbol{\phi}(t)$ 是可调参数向量, $e_1(t)$ 是系统与模型之间的输出偏差。

在确定了控制器的结构和偏差模型结构之后,剩下的问题就是如何设计一个可调参数的自适应调整律,使得偏差模型渐近稳定,由于对象的状态 \mathbf{x}_p 不能直接获取,偏差模型的状态亦不能直接获取,所以不能直接应用前面介绍的直接设计方法。但是,在图9-12所示的偏差模型中,输入信号

$\mathbf{w}(t) = [r(t), \mathbf{v}^{(1)}(t), y_p(t), \mathbf{v}^{(2)}(t)]^T$ 和输出信号 $e_1(t)$ 都是可以直接获取的,以这些信号为依据就可以确定自适应律。

在自适应律的推导过程中要用到正实函数和两个有关系统稳定性的定理及一个引理,下

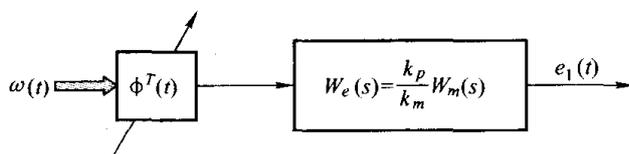


图9-12 偏差模型结构

面分别介绍。

定义 凡满足以下两个条件的实有理函数 $W(s)$ 称为正实函数:

- ① $W(s)$ 的极点在 s 的左半平面 (虚轴上只容许有一阶极点, 且其留数为正);
- ② 对任意 ω , $\operatorname{Re}[W(j\omega)] \geq 0$ (如 $\operatorname{Re}[W(j\omega)] > 0$, 称为严正实函数)。

根据上述定义, 很容易判断以下有理函数的正实性。 $W(s) = \frac{1}{s+a}$, 若 $a > 0$, 可知 $W(s)$

是正实函数; $W(s) = \frac{1}{s^2 + as + 1}$, $a > 0$, 则不满足上述条件, 不是正实函数。

定理 对于一般输入、输出间有惯性的动态系统, 传递函数 $W(s)$ 是正实函数的充分必要条件是在存在正定对称矩阵 P 和 Q , 并满足关系式

$$\begin{cases} A^T P + PA = -Q \\ Pb = c \end{cases} \quad (9-99)$$

式中, A 是 $n \times n$ 矩阵; b 是 $n \times 1$ 矩阵; c 是 $1 \times n$ 矩阵。

定理 如果误差模型的传递函数 $W_e(s) = h_c^T (sI - A_c)^{-1} b_c$ 为严正实函数 (以下简称严正实), 而且模型的输入信号 $w(t)$ 是有界的和分段连续的向量函数, 则方程组:

$$\begin{cases} \dot{e} = A_c e + b_c \phi^T \omega \\ e_1 = h_c^T e \\ \dot{\phi} = -\Gamma e_1 w \end{cases} \quad (9-100)$$

是稳定的, 而且 $t \rightarrow \infty$ 时 $|e_1(t)| \rightarrow 0$ 。

证明 选择以下二次型函数为可能的李雅普诺夫函数:

$$V(e, \phi) = \frac{1}{2} [e^T P e + \phi^T \Gamma^{-1} \phi]$$

式中, P 、 Γ 都是正定对称阵, 根据卡尔曼-雅库波维奇定理, 如果传递函数 $W_e(s) = h_c^T (sI - A_c)^{-1} b_c$ 是正实函数, 那么, 一定存在正定对称阵 P 、 Q , 并使得以下关系式成立:

$$\begin{cases} A_c^T P + PA_c = -Q \\ Pb_c = h_c \end{cases}$$

下面求 $V(e, \phi)$ 对时间的导数:

$$\begin{aligned} \dot{V}(e, \phi) &= \frac{1}{2} e^T (A_c^T P + PA_c) e + \frac{1}{2} e^T P b_c \phi^T w + \frac{1}{2} w^T \phi b_c^T P e + \phi^T \Gamma^{-1} \dot{\phi} \\ &= \frac{1}{2} e^T (A_c^T P + PA_c) e + e^T P b_c \phi^T w + \phi^T \Gamma^{-1} \dot{\phi} \end{aligned}$$

将关系式

$$\begin{aligned} P b_c &= h_c \\ \dot{\phi} &= -\Gamma e_1 w \end{aligned}$$

代入到 $\dot{V}(e, \phi)$ 的表达式中, 得

$$\dot{V}(e, \phi) = -\frac{1}{2} e^T Q e + e^T h_c \phi^T w - \phi^T e_1 w = -\frac{1}{2} e^T Q e \leq 0$$

由于 $\dot{V}(\mathbf{e}, \phi)$ 是负半定的, 所以, 微分方程式 (9-100) 是稳定的, 也就是说, 只要初始条件 $\mathbf{e}(t_0)$ 、 $\phi(t_0)$ 是有界的 (实际情况大都如此), 则 $\mathbf{e}(t)\phi(t)$ 也应该是有界的。

可以进一步证明: 当 $t \rightarrow \infty$ 时 $|e_1(t)| \rightarrow 0$ 。

由于 $\dot{V}(\mathbf{e}, \phi)$ 是时间的非增函数, 并且有下界, 假定 $V(\mathbf{e}, \phi)$ 收敛于某一有限值 $V(\infty)$, 又由于初始状态 $\mathbf{e}(0)$ 、 $\phi(0)$ 是有界的, 所以 $V(0)$ 有界, 即

$$\lim_{t \rightarrow \infty} \int_0^{\infty} \dot{V}(t) dt = V(\infty) - V(0) = \text{有限值}$$

因为 $\dot{V}(t)$ 是一致连续的, 则

$$\lim_{t \rightarrow \infty} \dot{V}(t) = 0$$

亦即

$$\lim_{t \rightarrow \infty} -\mathbf{e}^T(t) \mathbf{Q} \mathbf{e}(t) = 0$$

所以

$$t \rightarrow \infty \text{ 时, } \mathbf{e}(t) \rightarrow 0$$

又已知:

$$e_1(t) = \mathbf{h}_c^T \mathbf{e}(t)$$

因此

$$t \rightarrow \infty \text{ 时, } e_1(t) \rightarrow 0 \quad (\text{证毕})$$

最后附带说明一下, 如果输入信号 $\omega(t)$ 由不同的频率分量组成, 则方程组 (9-100) 是渐近稳定的, 即当 $t \rightarrow \infty$ 时 $\|\phi(t)\| \rightarrow 0$ 。

上述定理为设计参数的自适应调整律提供了重要依据。它说明, 只要偏差模型的传递函数 $W_e(s) = \mathbf{h}_c^T (s\mathbf{I} - \mathbf{A}_c)^{-1} \mathbf{b}_c$ 是严正实的, 就可以用实际能够获得的输出偏差 $e_1(t)$ 代替不易直接获取的状态偏差 $\mathbf{e}(t)$ 来组成自适应律, 即如果选择自适应律为

$$\dot{\phi} = -\Gamma e_1 w \quad (9-101)$$

就可以保证 $t \rightarrow \infty$ 时 $e_1(t) \rightarrow 0$ 。但是, 能否应用式 (9-101) 作为参数的自适应律, 关键在于 $W_e(s)$ 是否为严正实的。根据前面的推导已知:

$$W_e(s) = \frac{k_p}{k_m} W_m(s)$$

所以 $W_e(s)$ 是否是严正实, 又取决于 $W_m(s)$ 是否是严正实。前面关于对控制器结构和模型匹配问题的分析, 已提出了对参考模型传递函数 $W_m(s)$ 的要求, 即要求其分子、分母都是稳定多项式, 分子比分母低一阶。现在从应用式 (9-101) 为参数的自适应律, 并使系统渐近稳定的角度, 进一步要求 $W_m(s)$ 不仅是稳定的和最小相位的, 而且还是严正实的, 当这些条件都能满足时, 就可以由式 (9-101) 式构成自适应律, 从而完成了整个控制系统的设计。

自适应控制系统的结构仍如图 9-10 所示。

由于

$$\theta(t) = \theta^* + \phi(t)$$

$$\dot{\theta}(t) = \dot{\phi}(t)$$

所以图 9-10 中可调参数向量 $\theta^T = [k_0, c^T, d_0, d^T]$ 的自适应律也采用

$$\dot{\theta}(t) = -\Gamma e_1(t) w(t)$$

的形式。

以上的分析是基于这样的假定, 即参考模型的传递函数 $W_m(s)$ 是严正实的, 而且 $W_m(s)$ 与 $W_p(s)$ 的结构相同, 都是分母比分子高一阶。读者可能会提出这样的问题, 对参考模型提出

严正实的要求会不会影响模型的动态响应？当 $W_m(s)$ 与 $W_p(s)$ 的分母的阶比分子的阶高二阶以上时怎么处理？对第一个问题，实践证明，尽管对模型提出了严正实的要求，但仍然可以使模型具有良好的动态响应。对于第二个问题的解决，目前存在着不同的处理办法，这里只介绍一种最简单的方法。

(4) 自适应律设计的推广

为简单起见，这里只讨论最常见的情况，即对象传递函数分母比分子高二阶的情形。至于对象传递函数分母比分子高二阶以上的情形，也可以用类似的办法处理。

假定参考模型的结构与对象相同。即

$$W_m(s) = k_m \frac{Z_m(s)}{R_m(s)}$$

$$W_p(s) = k_p \frac{Z_p(s)}{R_p(s)}$$

式中， $R_m(s)$ 、 $R_p(s)$ 都是 n 阶稳定多项式； $Z_m(s)$ 、 $Z_p(s)$ 都是 $n-2$ 阶稳定多项式。

常见的对象 $W_p(s) = \frac{k_p}{s^2 + 2\xi\omega_0 s + \omega_0^2}$ 就属于这种情形。此时， $W_m(s)$ 与 $W_p(s)$ 都不满足严正实的条件，因此，不能直接应用式 (9-101) 作为参数的自适应调整律。但是，只要将加到可调系统入口的参考输入信号 $r(t)$ ，先经过滤波器 $\frac{1}{s+\sigma}$ 滤波，（其中 $\sigma > 0$ ），如图 9-13a 所示，再经过简单的结构变换，使系统转换成图 9-13b 所示。

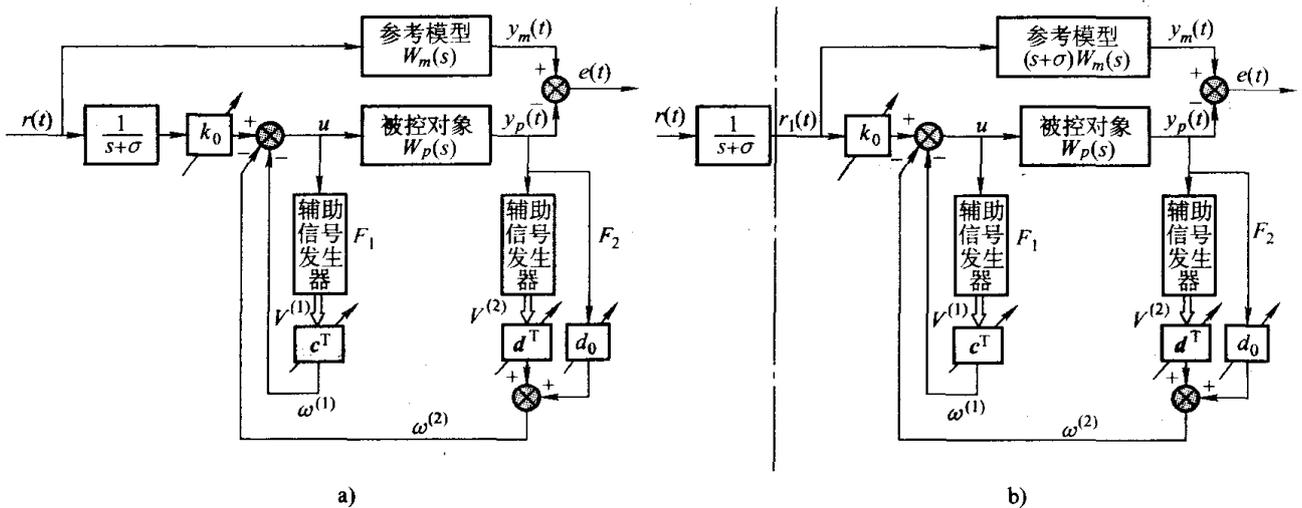


图 9-13 $W_m(s)$ 分母比分子大二阶时自适应控制系统结构

由图 9-13b 不难看出，如果把 $r_1(t)$ 看成等效系统（见图 9-13b 虚线以右部分）的输入， $W_{m1}(s) = (s + \sigma)W_m(s) = \frac{k_m(s + \sigma)Z_m(s)}{R_m(s)}$ 看成是新的等效参考模型，只要适当选择 $(s + \sigma)$ 总可以保证 $W_{m1}(s) = (s + \sigma)W_m(s)$ 为严正实函数。这样，自适应控制器的结构基本保持与原来一样，因此仍旧可以采用式 (9-101) 作为控制器参数的自适应调整律，即

$$\dot{\theta}(t) = -\Gamma e_1(t)w(t)$$

或者写成

$$\begin{bmatrix} \dot{k}_0(t) \\ \dot{c}^T(t) \\ \dot{d}_0(t) \\ \dot{d}^T(t) \end{bmatrix} = -\Gamma e_1(t) \begin{bmatrix} r_1(t) \\ v^{(1)}(t) \\ y_p(t) \\ v^{(2)}(t) \end{bmatrix}$$

由于此时 $Z_m(s)$ 、 $Z_p(s)$ 为 $n-2$ 阶，与前述情况不同。因此，还需要检验一下可调系统与参考模型之间的匹配条件是否还能满足。根据图 9-13 所示的结构，不难导出新的可调系统的传递函数如下：

$$W(s) = k_0 \frac{1}{s + \sigma} \cdot \frac{\frac{W_p(s)}{1 + W_1(s)}}{1 + \frac{W_p(s)W_2(s)}{1 + W_1(s)}}$$

经过简化后得

$$W(s) = \frac{k_0 k_p Z_p(s) N(s)}{[N(s) + C(s)] R_p(s) + k_p Z_p(s) [D(s) + d_0 N(s)]} \cdot \frac{1}{s + \sigma}$$

选择

- 1) $N(s) = Z_m(s)(s + \sigma)$
- 2) $[N(s) + C(s)] R_p(s) + k_p Z_p(s) [D(s) + d_0 N(s)] = R_m(s) Z_p(s)$
- 3) $k_0 = k_m / k_p$

则 $W(s)$ 可简化成

$$W(s) = \frac{k_m Z_p(s) Z_m(s) (s + \sigma)}{R_m(s) Z_p(s) (s + \sigma)}$$

由于 $(s + \sigma)$ 、 $Z_p(s)$ 都是稳定多项式，可以进行零极相消，因此，只要适当调整 $\theta^T = [k_0, c^T, d_0, d^T]$ ，就可以使 $W(s) = W_m(s)$ 。

对于 $W_p(s)$ 的分母比分子高二阶以上时，只要将参数输入信号的前置滤波器由一阶改成 $n-m-1$ 阶（其中 n 是 $R_p(s)$ 的阶， m 是 $Z_p(s)$ 的阶），然后用与上述方法类似的办法处理，即选择：

$$N(s) = Z_m(s)(s + \sigma_1) \cdots (s + \sigma_i), \quad i = n - m - 1$$

并保证 $W_{m1}(s) = (s + \sigma_1) \cdots (s + \sigma_i) W_m(s)$ 为严正实函数，则自适应律为

$$\begin{bmatrix} \dot{k}_0(t) \\ \dot{c}^T(t) \\ \dot{d}_0(t) \\ \dot{d}^T(t) \end{bmatrix} = -\Gamma e_1(t) \begin{bmatrix} \frac{r(t)}{(s + \sigma_1) \cdots (s + \sigma_i)} \\ v^{(1)}(t) \\ y_p(t) \\ v^{(2)}(t) \end{bmatrix}$$

9.2 模糊控制

模糊控制是用语言归纳操作人员的控制策略，运用语言变量和模糊集合理论形成控制算

法的一种控制。1974年 Mamdani 首次用模糊逻辑和模糊推理实现了第一台试验性的蒸汽机控制，开始了模糊控制在工业中的应用。模糊控制不需要建立控制对象精确的数学模型，只要把现场操作人员的经验和数据总结成较完善的语言控制规则，因此它能绕过对象的不确定性、不精确性、噪声，以及非线性、时变性、时滞等影响。系统的鲁棒性强，尤其适用于非线性、时变、滞后系统的控制。

模糊控制器的基本结构如图 9-14 所示。

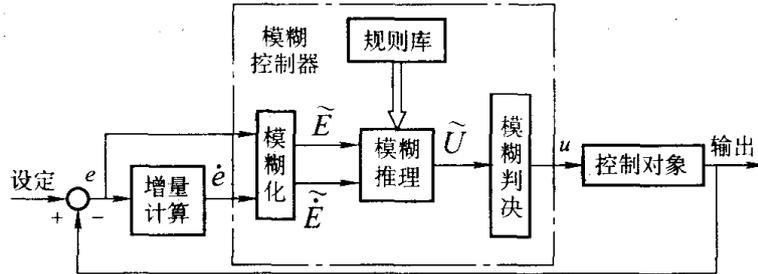


图 9-14 模糊控制器基本结构

最简单的模糊控制器需要完成的主要功能如下：

- 1) 把精确量（一般是系统的误差及误差变化率）转化成模糊量；
- 2) 按总结的语言规则（如图 9-14 的规则库中）进行模糊推理；
- 3) 推理的结果从模糊量转化成可以用于实际控制的精确量。

模糊控制器是模糊控制系统的核心。它是基于模糊条件语句描述的语言控制规则，所以又称为模糊语言控制器。本节以一个典型的二维模糊控制器为例，扼要介绍模糊控制器的工程设计及应用方法。必须指出，本节给出的二维模糊控制器本身具有相当的普遍性和实用性。

9.2.1 模糊控制器的输入、输出变量及其模糊化

1. 模糊控制器的输入、输出变量

模糊控制器是模仿人的一种控制。因此，可以考察一下人工控制的过程。在人工控制过程中，一般根据被控量的误差、误差的变化（EC）和误差变化的变化即误差变化的速率（ER）进行决策。人对误差最敏感，其次是误差的变化，再次是误差变化的速率。因此，模糊控制器的输入变量通常取 E 或 E 和 EC 或 E 、 EC 和 ER ，分别构成所谓一维、二维、三维模糊控制器。一维模糊控制器的动态性能不佳，通常用于一阶被控对象，二维模糊控制器的控制性能和控制复杂性都比较好，是目前广泛采用的一种形式。一般选择控制量的增量作为模糊控制器的输出变量。

2. 描述输入和输出变量的词集

在模糊控制中，输入、输出变量大小是以语言形式描述的，因此要选择描述这些变量的词汇。人们的日常语言中对各种事物和变量的描述，总是习惯于分为三个等级，如物体的大小分为大、中、小；运动的速度分为快、中、慢；年龄的大小分为老、中、青；人的身高分为高、中、矮；产品的质量分为优、中、劣（或一、二、三等）。所以，一般都选用“大、中、小”三个词汇来描述模糊控制器的输入、输出变量的状态，再加上正、负两个方向和零状态，共有七个词汇，即

{负大, 负中, 负小, 零, 正小, 正中, 正大}

一般用这些词的英文字头缩写为

$$\{NB, NM, NS, O, PS, PM, PB\}$$

一般情况下,选择上述七个词汇比较合适,但也可以多选或少选。选择较多的词汇可以精确描述变量,提高控制精度,但使控制规则变得复杂;选择的词汇过少使变量的描述太粗糙,导致控制器性能变坏。

为了提高系统稳态精度,通常在误差接近于零时增加分辨率,将“零”又分为“正零”和“负零”,因此,描述误差变量的词集一般取为

$$\{\text{负大, 负中, 负小, 负零, 正零, 正小, 正中, 正大}\}$$

用英文字头简记为

$$\{NB, NM, NS, NO, PO, PS, PM, PB\}$$

注意,上述“零”、“负零”、“正零”和其他词汇一样,都是描述了变量的一个区域。

3. 变量的模糊化

某个变量变化的实际范围称为该变量的基本论域。记误差的基本论域为 $[-x_e, x_e]$,误差变化的基本论域为 $[-x_c, x_c]$,模糊控制器的输出变量(系统的控制量)的基本论域为 $[-y_u, y_u]$ 。显然,基本论域内的量是精确量,因而模糊控制器的输入和输出都是精确量,但是模糊控制算法需要模糊量。因此,输入的精确量(数字量)需要转换为模糊量,这个过程称为“模糊化”(Fuzzification);另一方面,模糊算法所得到的模糊控制量需要转换为精确的控制量,这个过程称为“清晰化”或者“反模糊化”(Defuzzification)。

比较实用的模糊化方法是将基本论域分为 n 个档次,即取变量的模糊子集论域

$$\{-n, -n+1, \dots, 0, \dots, n-1, n\}$$

从基本论域 $[a, b]$ 到模糊子集论域 $[-n, n]$ 的转换公式为

$$y = \frac{2n}{b-a} \left[x - \frac{a+b}{2} \right] \quad (9-102)$$

增加论域中的元素个数可提高控制精度,但增大了计算量,而且模糊控制效果的改善并不显著。一般选择模糊论域中所含元素个数为模糊语言词集总数的2倍以上,确保诸模糊集能较好地覆盖论域,以避免出现失控现象。例如,在选择上述七个词汇情况下,可选择 E 和 EC 的论域均为

$$\{-6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6\}$$

选择模糊控制器的输出变量即系统的控制量 U 的论域为

$$\{-7, -6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6, 7\}$$

4. 隶属度

为了实现模糊化,要在上述离散化了的精确量与表示模糊语言的模糊量之间建立关系,即确定论域中的每个元素对各个模糊语言变量的隶属度。

隶属度是描述某个确定量隶属于某个模糊语言变量的程度。例如,在上述 E 和 EC 的论域中,+6隶属于 PB (正大),隶属度为1.0;+5也隶属于 PB ,但隶属度要比+6差,可取为0.8;+4隶属于 PB 的程度更小,隶属度可取为0.4;显然乃 $0 \sim -6$ 就不隶属于 PB 了,所以隶属度取为0。

确定隶属度要根据实际问题的具体情况。实验研究结果表明,人进行控制活动时的模糊

概念一般可以用正态型模糊变量描述。下面给出常用的确定模糊变量隶属度 μ 的赋值表，如表 9-1 至表 9-3 所示。

表 9-1 模糊变量 E 的赋值表

μ E \ e	-6	-5	-4	-3	-2	-1	-0	+0	+1	+2	+3	+4	+5	+6
PB	0	0	0	0	0	0	0	0	0	0	0.1	0.4	0.8	1.0
PM	0	0	0	0	0	0	0	0	0	0.2	0.7	1.0	0.7	0.2
PS	0	0	0	0	0	0	0	0.3	0.8	1.0	0.5	0.1	0	0
PO	0	0	0	0	0	0	0	1.0	0.6	0.1	0	0	0	0
NO	0	0	0	0	0.1	0.6	1.0	0	0	0	0	0	0	0
NS	0	0	0.1	0.5	1.0	0.8	0.3	0	0	0	0	0	0	0
NM	0.2	0.7	1.0	0.7	0.2	0	0	0	0	0	0	0	0	0
NB	1.0	0.8	0.4	0.1	0	0	0	0	0	0	0	0	0	0

表 9-2 模糊变量 EC 的赋值表

μ EC \ ec	-6	-5	-4	-3	-2	-1	0	+1	+2	+3	+4	+5	+6
PB	0	0	0	0	0	0	0	0	0	0.1	0.4	0.8	1.0
PM	0	0	0	0	0	0	0	0	0.2	0.7	1.0	0.7	0.2
PS	0	0	0	0	0	0	0	0.9	1.0	0.7	0.2	0	0
O	0	0	0	0	0	0.5	1.0	0.5	0	0	0	0	0
NS	0	0	0.2	0.7	1.0	0.9	0	0	0	0	0	0	0
NM	0.2	0.7	1.0	0.7	0.2	0	0	0	0	0	0	0	0
NB	1.0	0.8	0.4	0.1	0	0	0	0	0	0	0	0	0

表 9-3 模糊变量 U 的赋值表

μ U \ u	-7	-6	-5	-4	-3	-2	-1	0	+1	+2	+3	+4	+5	+6	+7
PB	0	0	0	0	0	0	0	0	0	0	0	0.1	0.4	0.8	1.0
PM	0	0	0	0	0	0	0	0	0	0.2	0.7	1.0	0.7	0.2	0
PS	0	0	0	0	0	0	0	0.4	1.0	0.8	0.4	0.1	0	0	0
O	0	0	0	0	0	0	0.5	1.0	0.5	0	0	0	0	0	0
NS	0	0	0	0.1	0.4	0.8	1.0	0.4	0	0	0	0	0	0	0
NM	0	0.2	0.7	1.0	0.7	0.2	0	0	0	0	0	0	0	0	0
NB	1.0	0.8	0.4	0.1	0	0	0	0	0	0	0	0	0	0	0

9.2.2 建立模糊控制规则

模糊控制是语言控制，因此要用语言归纳专家的手动控制策略，从而建立模糊控制规则表。手动控制策略一般都可用条件语句加以描述。条件语句的基本类型为

if A or B and C or D then E

例如，水温控制规则之一为：若水温高或偏高，且温度上升快或较快，则加大冷水流量。用条件语句可表达为

if E=NB or NM and EC=NB or NM then U=PB

下面推荐一种根据系统输出的误差及误差的变化趋势，以消除误差的模糊控制规则。该规则用下述 21 条模糊条件语句来描述，基本总结了众多的被控对象在手动操作过程中，各种可能出现的情况和相应的控制策略，其中误差 E 、误差变化 EC 及控制量 U 对于不同的被控对象有着不同的物理意义。例如，锅炉的压力与加热的关系；汽轮机转速与阀门开度的关系；反应堆的热交换关系；飞机、轮船的航向与舵的关系；卫星的姿态与作用力的关系等。

- (1) if E=NB or NM and EC=NB or NM then U=PB
- (2) if E=NB or NM and EC=NS or O then U=PB
- (3) if E=NB or NM and EC=PS then U=PM
- (4) if E=NB or NM and EC=PM or PB then U=O
- (5) if E=NS and EC=NB or NM then U=PM
- (6) if E=NS and EC=NS or O then U=PM
- (7) if E=NS and EC=PS then U=O
- (8) if E=NS and EC=PM or PB then U=NS
- (9) if E=NO or PO and EC=NB or NM then U=PM
- (10) if E=NO or PO and EC=NS then U=PS
- (11) if E=NO or PO and EC=O then U=O
- (12) if E=NO or PO and EC=PS then U=NS
- (13) if E=NO or PO and EC=PM or PB then U=NM
- (14) if E=PS and EC=NB or NM then U=PS
- (15) if E=PS and EC=NS then U=O
- (16) if E=PS and EC=O or PS then U=NM
- (17) if E=PS and EC=PM or PB then U=NM
- (18) if E=PM or PB and EC=NB or NM then U=O
- (19) if E=PM or PB and EC=NS then U=NM
- (20) if E=PM or PB and EC=O or PS then U=NB
- (21) if E=PM or PB and EC=PM or PB then U=NB

上述 21 条模糊条件语句可以归纳为模糊控制规则表 9-4。

表 9-4 模糊控制规则表

E \ EC \ U	EC						
	PB	PM	PS	O	NS	NM	NB
PB	NB	NB	NB	NB	NM	O	O
PM	NB	NB	NB	NB	NM	O	O
PS	NM	NM	NM	NM	O	PS	PS
PO	NM	NM	NS	O	PS	PM	PM
NO	NM	NM	NS	O	PS	PM	PM
NS	NS	NS	O	PM	PM	PM	PM
NM	O	O	PM	PB	PB	PB	PB
NB	O	O	PM	PB	PB	PB	PB

9.2.3 模糊关系与模糊推理

模糊控制规则实际上是一组多重条件语句，可以表示为从误差论域到控制量论域的模糊关系矩阵 R ，通过误差的模糊向量 E 和误差变化的模糊向量 EC ，与模糊关系 R 的合成进行模糊推理，得到控制量的模糊向量，然后采用“清晰化”方法将模糊控制向量转换为精确量。

根据模糊集合和模糊关系理论，对于不同类型的模糊规则可用不同的模糊推理方法。下面仅介绍其中的对“if A then B”类型的模糊规则的推理。

若规则为已知输入为 A ，则输出为 B ；若现在已知输入为 A' ，则输出 B' 用合成规则求取

$$B' = A' \circ R \quad (9-103)$$

其中模糊关系 R 定义为

$$\mu_R(x, y) = \min[\mu_A(x), \mu_B(y)] \quad (9-104)$$

例如，已知当输入的模糊集合 A 和输出的模糊集合 B 分别为

$$A = 1.0/a_1 + 0.8/a_2 + 0.5/a_3 + 0.2/a_4 + 0.0/a_5$$

$$B = 0.7/b_1 + 1.0/b_2 + 0.6/b_3 + 0.0/b_4$$

这里采用模糊集合的 Zadeh 表示法，其中 a_i 、 b_i 表示模糊集合所对应的论域中的元素，而 μ_i 表示相应的隶属度，“/”不表示分数的意思。

$$R = A \times B = \begin{bmatrix} 1.0 \cap 0.7 & 1.0 \cap 1.0 & 1.0 \cap 0.6 & 1.0 \cap 0.0 \\ 0.8 \cap 0.7 & 0.8 \cap 1.0 & 0.8 \cap 0.6 & 0.8 \cap 0.0 \\ 0.5 \cap 0.7 & 0.5 \cap 1.0 & 0.5 \cap 0.6 & 0.5 \cap 0.0 \\ 0.2 \cap 0.7 & 0.2 \cap 1.0 & 0.2 \cap 0.6 & 0.2 \cap 0.0 \\ 0.0 \cap 0.7 & 0.0 \cap 1.0 & 0.0 \cap 0.6 & 0.0 \cap 0.0 \end{bmatrix}$$

$$= \begin{bmatrix} 0.7 & 1.0 & 0.6 & 0.0 \\ 0.7 & 0.8 & 0.6 & 0.0 \\ 0.5 & 0.5 & 0.5 & 0.0 \\ 0.2 & 0.2 & 0.2 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 \end{bmatrix}$$

则当输入 $A' = 0.4/a_1 + 0.7/a_2 + 1.0/a_3 + 0.6/a_4 + 0.0/a_5$ ， B' 由下式求取

$$B' = A' \circ R = \begin{bmatrix} 0.4 \\ 0.7 \\ 1.0 \\ 0.6 \\ 0.0 \end{bmatrix}^T \circ \begin{bmatrix} 0.7 & 1.0 & 0.6 & 0.0 \\ 0.7 & 0.8 & 0.6 & 0.0 \\ 0.5 & 0.5 & 0.5 & 0.0 \\ 0.2 & 0.2 & 0.2 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 \end{bmatrix}$$

$$= [(0.4 \cap 0.7) \cup (0.7 \cap 0.7) \cup (1.0 \cap 0.5) \cup (0.6 \cap 0.2) \cup (0.0 \cap 0.0),$$

$$(0.4 \cap 1.0) \cup (0.7 \cap 0.8) \cup (1.0 \cap 0.5) \cup (0.6 \cap 0.2) \cup (0.0 \cap 0.0),$$

$$(0.4 \cap 0.6) \cup (0.7 \cap 0.6) \cup (1.0 \cap 0.5) \cup (0.6 \cap 0.2) \cup (0.0 \cap 0.0),$$

$$(0.4 \cap 0.0) \cup (0.7 \cap 0.0) \cup (1.0 \cap 0.0) \cup (0.6 \cap 0.0) \cup (0.0 \cap 0.0)]$$

$$\begin{aligned}
 &= [(0.4 \ 0.7 \ 0.5 \ 0.2 \ 0.0), (0.4 \ 0.7 \ 0.5 \ 0.2 \ 0.0), \\
 &\quad (0.4 \ 0.6 \ 0.5 \ 0.2 \ 0.0), (0.0 \ 0.0 \ 0.0 \ 0.0 \ 0.0)] \\
 &= (0.7, 0.7, 0.6, 0.0)
 \end{aligned}$$

则 $B' = 0.7/b_1 + 0.7/b_2 + 0.6/b_3 + 0.0/b_4$

在上述运算中，“ \cap ”为取小运算，“ \cup ”为取大运算。

由于系统的控制规则库是由若干条规则组成的，对于每一条推理规则都可以得到一个相应的模糊关系。 n 条规则就有 n 个模糊关系： R_1, R_2, \dots, R_n ，对于整个系统的全部控制规则所对应的模糊关系 R 可对 n 个模糊关系 R_i ($i=1, 2, \dots, n$) 取“并”操作得到，即

$$R = R_1 \cup R_2 \cup \dots \cup R_n = \bigcup_{i=1}^n R_i \quad (9-105)$$

9.2.4 模糊控制向量的模糊判决

由上述得到的控制量是一个模糊集合，需要采用“清晰化”方法将模糊控制向量转换为精确量。下面介绍两种简单、实用的方法。

1. 最大隶属度法

这种方法是在模糊控制向量中，取隶属度最大的控制量作为模糊控制器的控制量。例如，当得到模糊控制向量为

$$U' = 0.1/2 + 0.4/3 + 0.7/4 + 1.0/5 + 0.7/6 + 0.3/7$$

由于控制量隶属于等级 5 的隶属度为最大，所以取控制量为 $U = 5$ 。

这种方法的优点是简单易行，缺点是完全排除了其他隶属度较小的控制量的影响和作用，没有充分利用取得的信息。

2. 加权平均判决法

为了克服最大隶属度法的缺点，可以采用加权平均判决法，也称垂心法即

$$U = \frac{\sum_{i=1}^n \mu(u_i) u_i}{\sum_{i=1}^n \mu(u_i)} \quad (9-106)$$

例如 $U' = 0.1/2 + 0.8/3 + 1.0/4 + 0.8/5 + 0.1/6$

则 $U = \frac{2 \times 0.1 + 3 \times 0.8 + 4 \times 1.0 + 5 \times 0.8 + 6 \times 0.1}{0.1 + 0.8 + 1.0 + 0.8 + 0.1} = 4$

9.2.5 模糊控制表

模糊关系、模糊推理以及模糊判决的运算可以离线进行，最后得到模糊控制器输入量的量化等级 E 、 EC 与系统控制量的量化等级 U 之间的确定关系，这种关系通常称为“控制表”。对应于前面的 21 条控制规则的“控制表”，如表 9-5 所示。

模糊控制表可以离线求出，作为文件存储在计算机中，计算机实时控制时只要将 A/D 得到的误差 e 和误差的变化 ec 进行量化，得到相应的等级 E 和 EC ，然后从文件中直接查询所需采取的控制策略。

表 9-5 模糊控制表

EC \ E	-6	-5	-4	-3	-2	-1	0	+1	+2	+3	+4	+5	+6
-6	7	6	7	6	7	7	7	4	4	2	0	0	0
-5	6	6	6	6	6	6	6	4	4	2	0	0	0
-4	7	6	7	6	7	7	7	4	4	2	0	0	0
-3	7	6	6	6	6	6	6	3	2	0	-1	-1	-1
-2	4	4	4	5	4	4	4	1	0	0	-1	-1	-1
-1	4	4	4	5	4	4	1	0	0	0	-3	-2	-1
-0	4	4	4	5	1	1	0	-1	-1	-1	-4	-4	-4
+0	4	4	4	5	1	1	0	-1	-1	-1	-4	-4	-4
+1	2	2	2	2	0	0	-1	-4	-4	-3	-4	-4	-4
+2	1	2	1	2	0	-3	-4	-4	-4	-3	-4	-4	-4
+3	0	0	0	0	-3	-3	-6	-6	-6	-6	-6	-6	-6
+4	0	0	0	-2	-4	-4	-7	-7	-7	-6	-7	-6	-7
+5	0	0	0	-2	-4	-4	-6	-6	-6	-6	-6	-6	-6
+6	0	0	0	-2	-4	-4	-7	-7	-7	-6	-7	-6	-7

9.2.6 确定实际的控制量

显然，实际的控制量 u 应为从控制表中查到的量化等级 U 乘以比例因子。设实际的控制量 u 的变化范围为 $[a, b]$ ，量化等级为 $\{-n, -n+1, \dots, 0, \dots, n-1, n\}$ ，则实际的控制量应为

$$u = \frac{a+b}{2} + \frac{b-a}{2n}U \quad (9-107)$$

若 $a = -y_u$ ， $b = y_u$ ，则

$$u = \frac{y_u}{n}U \quad (9-108)$$

例如，在上述二维模糊控制器中当 E 和 EC 的量化等级分别为 -3 和 $+1$ 时，由模糊控制表查得 $U = 3$ ，模糊控制器输出的实际控制量应为 $u = \frac{3}{7}y_u$ 。

9.2.7 模糊控制算法的工程实现

模糊控制由于其计算较复杂，一般都是通过计算机来实现，它是一种以模糊数学为基础的计算机数字控制算法。模糊控制系统的组成结构类似于一般的数字控制系统，如图 9-15 所示。



图 9-15 模糊控制系统的计算机实现框图

与一般的计算机控制系统相比,模糊控制系统的控制器是模糊控制器。根据上述模糊控制原理,可以用多种方法实现模糊控制算法。

1. 查表法

查表法是模糊控制应用最早、最广的一种方法。这种方法是采用离线方式完成模糊推理,得到模糊控制表,如表 9-5,然后将模糊控制表存入计算机,在线控制时只要进行简单的查表操作,一般的单片机就能完成,而且实时性好。目前模糊控制家电产品大都采用这种方法。查表法的缺点是当改变模糊控制规则和隶属函数时,则需要重新计算模糊控制表。

2. 软件模糊推理法

这种方法是将模糊控制的全过程都用软件实现,在线进行输入量模糊化、模糊推理、模糊决策过程。目前,美、日、德等国已经研制出多种模糊控制软件,以供各种应用程序的移植。

3. 模糊控制器专用芯片

用硬件实现模糊控制的特点是实时性好、控制精度高。目前,模糊控制器专用芯片已经商品化,在伺服系统、机器人、汽车等控制中得到广泛应用。随着模糊控制的广泛应用,控制专用芯片的价格将不断降低。

模糊控制由于理论研究较成熟,实现较简单,适应面广而获得了广泛的应用。从复杂的水泥回转窑,到单回路的温度控制,以及洗衣机等很普及的家用电器都有所应用。现在,许多公司和生产厂家都能生产定型的模糊控制器,提供各种型号和功能的模糊控制芯片,从而大大地促进了模糊控制技术的广泛应用。

模糊控制从某种意义上可以看作是一种非线性分段的 PD 控制,因此算法本身不能保证消除稳态误差,现在已有许多改进算法或与其他具有积分作用的算法结合能够克服模糊控制这一弱点。

模糊控制存在的主要问题是:要有较好的效果,必须具有完善的控制规则,然而总结出的控制规则一般不具有普遍意义,故通用性较差。对于某些复杂的工业过程,有时难以总结出较完整的经验;并且当对象动态特性发生变化,或者受到随机干扰时都会影响模糊控制的效果。

9.3 神经网络控制

神经网络是由众多简单的神经元连接而成的一个网络。尽管每个神经元结构、功能都不复杂,但网络的整体动态行为则是极为复杂的,可以组成高度非线性动力学系统,从而可以表达很多复杂的物理系统。由于神经网络具有大规模并行处理能力和自适应、自组织、自学习能力,以及采用分布式存储等特点,在自动控制领域中展现了广阔的应用前景。

9.3.1 神经元数理模型及其学习算法

1. 生物神经元结构

从生物控制与信息处理的角度看,神经元结构如图 9-16 所示。

神经元细胞体由细胞核、细胞质、细胞膜等组成。由细胞体向外伸出的最长的一条分支称为轴突,即神经纤维。轴突是用来传递和输出信息的,其端部的许多轴突末梢为信号输出

端子，用于将神经冲动传出到其他神经元。由细胞体向外伸出的其他许多较短的分支称为树突。树突相当于细胞的输入端，树突的全长各点都能接收其他神经元的冲动。神经冲动只能由前一级神经元的轴突末梢传向下一级神经元的树突或细胞体，不能反方向传递。

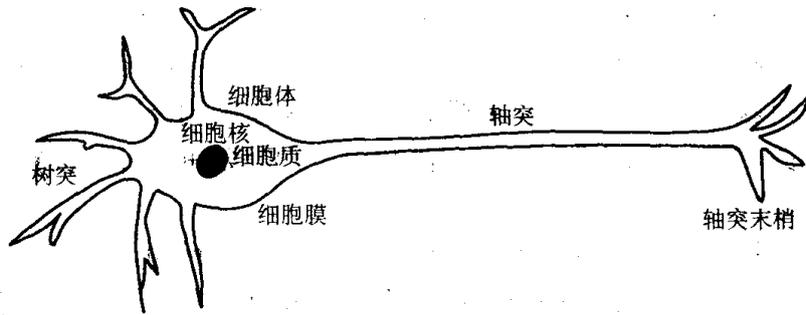


图 9-16 神经元的结构

神经元具有两种常规工作状态，即兴奋与抑制，即满足“0-1”律。当传入的神经冲动使细胞膜电位升高超过阈值时，细胞进入兴奋状态，产生神经冲动并由轴突输出；当传入的冲动使膜电位下降低于阈值时，细胞进入抑制状态，没有神经冲动输出。

2. 神经元数学模型

根据神经元的结构和功能，从 20 世纪 40 年代开始先后提出的神经元模型有几百种之多。下面介绍神经元的一种所谓标准、统一的数学模型。它由三部分组成，即加权求和、线性环节和非线性函数映射，如图 9-17 所示。图中， $y_i(t)$ 为神经元的输出， θ_i 为神经元的阈值， a_{ij}, b_{ik} 为权值， $u_k(t) (k = 1, 2, \dots, M)$ 为外部输入， $y_j(t) (j = 1, 2, \dots, N)$ 为其他神经元的输出。

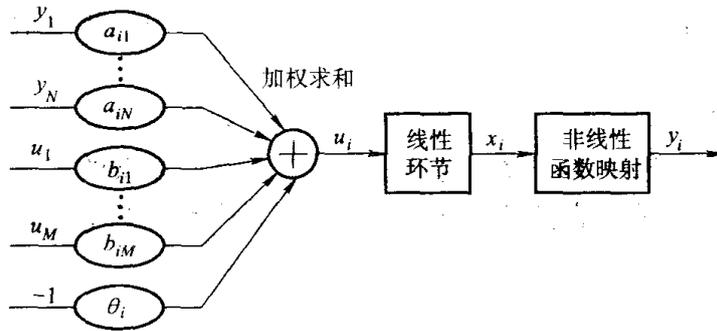


图 9-17 神经元数学模型

(1) 加权求和

$$v_i(t) = \sum_{j=1}^N a_{ij} y_j(t) + \sum_{k=1}^M b_{ik} u(k) - \theta_i \tag{9-109}$$

将式 (9-97) 记为矩阵形式

$$V(t) = AY(t) + BU(t) - \theta \tag{9-110}$$

式中， $A = \{a_{ij}\}_{N \times N}$ ， $B = \{b_{ik}\}_{N \times M}$ ， $V = [v_i, \dots, v_N]^T$ ， $Y = [y_i, \dots, y_N]^T$ ，

$$U = [u_i, \dots, u_M]^T, \theta = [\theta_i, \dots, \theta_N]^T。$$

(2) 线性环节的传递函数描述

$$X_i(s) = H(s)V_i(s) \tag{9-111}$$

式中, $H(s)$ 通常取为: $1, \frac{1}{s}, \frac{1}{Ts+1}, e^{-\tau s}$ 。

(3) 常用的非线性函数

阶跃函数
$$f(x_i) = \begin{cases} 1 & x_i > 0 \\ 0 & x_i \leq 0 \end{cases}$$

分段线性函数
$$f(x_i) = \begin{cases} 1 & x_i \geq B \\ ax_i + b & A \leq x_i < B \\ 0 & x_i < A \end{cases}$$

s 型函数
$$f(x_i) = \frac{1}{1 + e^{-x_i + C}}$$

3. 神经网络的学习和训练

人的大脑神经系统是进行学习和记忆的最重要的智能器官。要模拟人脑神经网络的学习功能, 必须使人工神经网络具有学习功能并对其进行训练。即从应用环境中选出一些样本数据, 通过不断地调整权矩阵, 直至得到合适的输入、输出关系为止。下面介绍几种常用的神经网络学习规则。

(1) 联想式学习规则 (Hebb 学习规则)

心理学家 Hebb 认为, 如果神经网络中某一神经元与另一直接与其相连的神经元同时处于兴奋状态, 那么这两个神经元间的连接强度应该加强, 如图 9-18 所示。

从神经元 u_j 到神经元 u_i 的连接强度, 即权重变化 Δw_{ij} 可用式 (9-100) 表达, 即

$$\Delta w_{ij} = G[a_i(k), t_i(k)] \times H[\bar{y}_j(k), w_{ij}] \quad (9-112)$$

式中, $t_i(k)$ 是神经元 u_i 的教师信号; G 是神经元 u_i 的活性度 $a_i(k)$ 和教师信号 $t_i(k)$ 的函数; H 是神经元 u_j 输出 $\bar{y}_j(k)$ 和连接权重 w_{ij} 的函数。输出 $\bar{y}_j(k)$ 与神经元 u_j 的活性度 $a_j(k)$ 之间满足非线性关系 $\bar{y}_j(k) = f_j[a_j(k)]$ 。

当上述的教师信号 $t_i(k)$ 没有给出时, 函数 H 只与输出 $\bar{y}_j(k)$ 成正比, 于是式 (9-112) 可变为更简单的形式

$$\Delta w_{ij} = \eta a_i(k) \bar{y}_j(k) \quad (9-113)$$

式中, $\eta > 0$, 为学习速率。

式 (9-113) 表明, 对一个神经元较大的输入或该神经元活性度较大的情况, 它们之间的连接权重会更大。

(2) 误差传播式学习

根据 Hebb 学习规则, 考虑到式 (9-112) 中函数 G 与教师信号 $t_i(k)$ 和神经元 u_i 的活性度 $a_i(k)$ 的差值成比例, 函数 H 与神经元的输出 $\bar{y}_j(k)$ 成比例, 即

$$G[a_i(k), t_i(k)] = \eta_1 [t_i(k) - a_i(k)]$$

$$H[\bar{y}_j(k), w_{ij}] = \eta_2 \bar{y}_j(k)$$

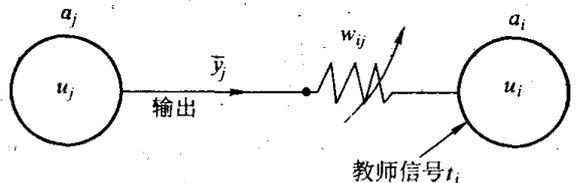


图 9-18 联想式学习规则示意

则连接权重变化可表示为

$$\Delta w_{ij} = \eta [t_i(k) - a_i(k)] \bar{y}_j(k)$$

式中, $\eta = \eta_1 \eta_2$, 为学习速率。

在上式中, 若将教师信号 $t_i(k)$ 作为期望输出 d_i , 而把 $a_i(k)$ 理解为实际输出 y_i , 则该式变为

$$\Delta w_{ij} = \eta [d_i - y_i] \bar{y}_j(k) = \eta \delta \bar{y}_j(k) \quad (9-114)$$

式中, $\delta = d_i - y_i$, 为期望输出与实际输出的差值, 又称误差修正规则 (δ 规则)。根据这个规则反复迭代计算, 直至求出使 δ 达到最小的 w_{ij} 值。

9.3.2 BP 神经网络及其学习算法程序设计

BP 神经网络是多层前向网络, 其结构如图 9-19 所示。

设 BP 神经网络具有 m 层, 第一层称为输入层, 最后一层称为输出层, 中间各层称为隐层。输入信息由输入层向输出层逐层传递。各个神经元的输入、输出关系函数是 f , 由 $k-1$ 层的第 j 个神经元到 k 层的第 i 个神经元的连接权重为 w_{ij} , 输入、输出样本为 $\{x_{si}, y_i\}, i=1, 2, \dots, n$ 。并设第 k 层第 i 个神经元输入的总和为 u_i^k , 输出为 y_i^k , 则各变量之间的关系为

$$y_i^k = f(u_i^k)$$

$$u_i^k = \sum_j w_{ij} y_j^{k-1} \quad k=1, 2, \dots, m \quad (9-115)$$

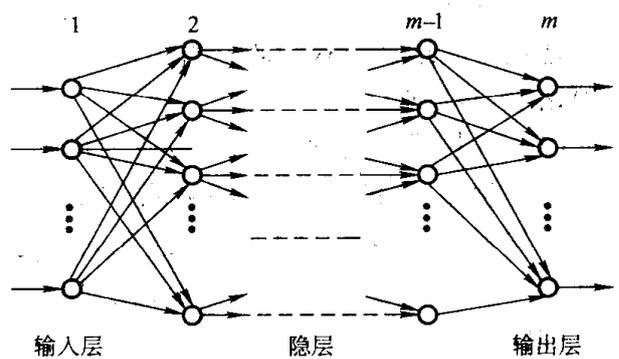


图 9-19 BP 神经网络结构

BP 学习算法是通过反向学习过程使误差最小, 因此选择目标函数为

$$J = \frac{1}{2} \sum_{j=1}^n (y_j^m - y_j)^2 \quad (9-116)$$

即选择神经网络权重使期望输出 y_j^m 与实际输出 y_j 之差的平方和最小。这种学习算法实际上是求误差函数 J 的极小值, 约束条件是式 (9-115), 可以利用非线性规划中的“最快下降法”, 使权重沿误差函数的负梯度方向改变, 因此, 权重的修正量为

$$\Delta w_{ij} = -\varepsilon \frac{\partial J}{\partial w_{ij}} \quad (9-117)$$

式中, ε 为学习步长。

下面推导 BP 学习算法。先求 $\frac{\partial J}{\partial w_{ij}}$, 即有

$$\frac{\partial J}{\partial w_{ij}} = \frac{\partial J}{\partial u_i^k} \cdot \frac{\partial u_i^k}{\partial w_{ij}} = \frac{\partial J}{\partial u_i^k} \cdot \frac{\partial}{\partial w_{ij}} \left(\sum_j w_{ij} y_j^{k-1} \right) = \frac{\partial J}{\partial u_i^k} y_j^{k-1}$$

$$\text{令} \quad d_i^k = \frac{\partial J}{\partial u_i^k} = \frac{\partial J}{\partial y_i^k} \cdot \frac{\partial y_i^k}{\partial u_i^k}$$

则由式(9-117)得

$$\Delta w_{ij} = -\varepsilon d_i^k y_j^{k-1}$$

下面推导 d_i^k 的计算公式:

$$d_i^k = \frac{\partial J}{\partial u_i^k} = \frac{\partial J}{\partial y_i^k} \cdot \frac{\partial y_i^k}{\partial u_i^k} = \frac{\partial J}{\partial y_i^k} \cdot \frac{df(u_i^k)}{du_i^k}$$

取 $f(\cdot)$ 为 s 型函数, 即

$$y_i^k = f(u_i^k) = \frac{1}{1 + e^{-u_i^k}}$$

$$\text{即} \quad \frac{\partial y_i^k}{\partial u_i^k} = \frac{df(u_i^k)}{du_i^k} = \frac{e^{-u_i^k}}{[1 + e^{-u_i^k}]^2} = y_i^k (1 - y_i^k)$$

$$d_i^k = y_i^k (1 - y_i^k) \frac{\partial J}{\partial y_i^k}$$

下面分为两种情况求 $\frac{\partial J}{\partial y_i^k}$ 。

1) 当 i 为输出层 (m 层) 的神经元, 即 $k = m$, $y_i^k = y_i^m$ 。由误差定义式得

$$\frac{\partial J}{\partial y_i^k} = \frac{\partial J}{\partial y_i^m} = y_i^m - y_i$$

$$\text{则} \quad d_i^m = y_i^m (1 - y_i^m) (y_i^m - y_i)$$

2) 若 i 为隐单元层 k , 有

$$\frac{\partial J}{\partial y_i^k} = \sum_l \frac{\partial J}{\partial u_l^{k+1}} \frac{\partial u_l^{k+1}}{\partial y_i^k} = \sum_l w_{ly} d_l^{k+1}$$

$$\text{则} \quad d_i^k = y_i^k (1 - y_i^k) \sum_l w_{ly} d_l^{k+1}$$

综上所述, BP 学习算法可以归纳为

$$\begin{cases} \Delta w_{ij} = -\varepsilon d_i^k y_j^{k-1} \\ d_i^m = y_i^m (1 - y_i^m) (y_i^m - y_i) \\ d_i^k = y_i^k (1 - y_i^k) \sum_l w_{ly} d_l^{k+1} \end{cases} \quad (9-118)$$

从以上公式可见, 求 k 层的误差信号 d_i^k , 需要上一层的 d_l^{k+1} , 因此, 误差函数的求取是一个始于输出层的反向传播的递归过程, 所以称为反向传播学习算法。通过多个样本的学习, 修改权重, 不断减少偏差, 最后达到满意的结果。BP 学习算法的程序框图如图 9-20 所示。

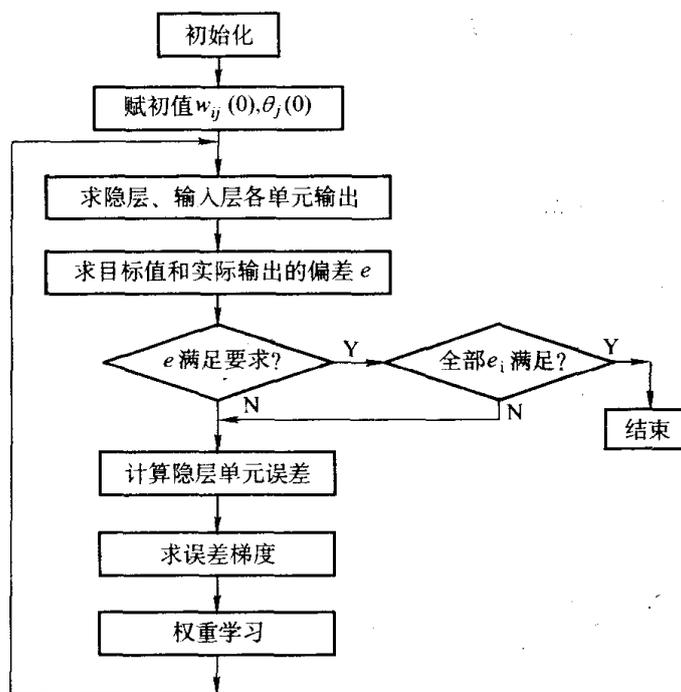


图 9-20 BP 算法程序框图

9.3.3 Hopfield 神经网络

20 世纪 80 年代初，霍普菲尔德先后提出了离散型 Hopfield 神经网络和连续型 Hopfield 神经网络，引入“计算能量函数”的概念，给出了网络稳定性判据，尤其是给出了 Hopfield 神经网络的电子电路实现，为神经计算机的研究奠定了基础，同时开拓了神经网络用于联想记忆和优化计算的新途径，从而有力地推动了神经网络的研究。

1. 离散型 Hopfield 神经网络

Hopfield 神经网络是全互联反馈神经网络，它的每一个神经元都和其他神经元相连接。

n 阶离散型 Hopfield 神经网络 N ，可由一个 $n \times n$ 阶矩阵 $w = [w_{ij}]$ 和一个 n 维向量 $\theta = [\theta_1, \dots, \theta_n]^T$ 所惟一确定，记为 $N = (w, \theta)$ ，其中， w_{ij} 表示神经元 i 与 j 的连接强度， θ_i 表示神经元 i 的阈值。若用 $X_i(t)$ 表示 t 时刻神经元所处的状态（可能为 1 或 -1，即 $X_i(t) = \pm 1$ ），那么神经元 i 的状态随时间变化的规律（又称演化律）为

$$X_i(t+1) = \text{sgn}(H_i(t)) = \begin{cases} 1, & H_i(t) \geq 0 \\ -1, & H_i(t) < 0 \end{cases} \quad (9-119)$$

式中， $H_i(t) = \sum_{j=1}^n w_{ij} X_j(t) - \theta_i$ ， $1 \leq i \leq n$ 。

Hopfield 神经网络是一个多输入、多输出带阈值的二态非线性动力学系统，因此存在一种所谓能量函数。在满足一定的参数条件下，该能量函数值在网络运行过程中不断降低，最后趋于稳定的平衡状态。Hopfield 引入这种能量函数作为网络计算求解的工具，因此常称它为计算能量函数。

离散型 Hopfield 神经网络的计算能量函数定义为

$$E = -\frac{1}{2} \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N w_{ij} x_i x_j + \sum_{i=1}^N \theta_i x_i \quad (9-120)$$

式中, x_i 、 x_j 是各个神经元的输出。

下面考察第 m 个神经元的输出变化前后, 能量函数 E 值的变化。

设 $x_m = 0$ 的能量函数值为 E_1 , 有

$$E_1 = -\frac{1}{2} \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N w_{ij} x_i x_j + \sum_{i=1}^N \theta_i x_i$$

将 $i = m$ 项分出来, 并注意到 $x_m = 0$, 得

$$E_1 = -\frac{1}{2} \sum_{\substack{i=1 \\ i \neq m}}^N \sum_{\substack{j=1 \\ j \neq i}}^N w_{ij} x_i x_j + \sum_{\substack{i=1 \\ i \neq m}}^N \theta_i x_i \quad (9-121)$$

类似地, 当 $x_m = 1$ 时的能量函数值为 E_2 , 则有

$$E_2 = -\frac{1}{2} \sum_{\substack{i=1 \\ i \neq m}}^N \sum_{\substack{j=1 \\ j \neq i}}^N w_{ij} x_i x_j + \sum_{\substack{i=1 \\ i \neq m}}^N \theta_i x_i - \sum_{\substack{j=1 \\ j \neq m}}^N w_{mj} x_j + \theta_m \quad (9-122)$$

当神经元状态由“0”变为“1”时, 能量函数 E 值的变化量 ΔE 为

$$\Delta E = E_2 - E_1 = -\left(\sum_{\substack{j=1 \\ j \neq m}}^N w_{mj} x_j - \theta_m \right) \quad (9-123)$$

由于此时神经元的输出是由 0 变为 1, 因此满足神经元兴奋条件

$$\sum_{\substack{j=1 \\ j \neq m}}^N w_{mj} x_j - \theta_m > 0 \quad (9-124)$$

所以由式 (9-123) 得 $\Delta E < 0$ 。

当神经元状态由“1”变为“0”时, 能量函数 E 值的变化量 ΔE 为

$$\Delta E = E_1 - E_2 = \sum_{\substack{j=1 \\ j \neq m}}^N w_{mj} x_j - \theta_m \quad (9-125)$$

由于此时神经元的输出是由“1”变为“0”, 因此

$$\sum_{\substack{j=1 \\ j \neq m}}^N w_{mj} x_j - \theta_m < 0 \quad (9-126)$$

所以也有 $\Delta E < 0$ 。

综上所述, 总有 $\Delta E < 0$, 这表明神经网络在运行过程中能量将不断降低, 最后趋于稳定的平衡状态。

2. 连续型 Hopfield 神经网络

1984 年, Hopfield 又提出一种连续时间神经网络模型及其电子线路实现, 如图 9-21 所示。

其中，每一个神经元由电阻 R_i 和电容 C_i ，以及具有饱和非线性特性的运算放大器模拟，输出也同时还反馈至其他神经元，但不反馈自身。 u_i 表示神经元 i 的膜电位状态； v_i 表示它的输出； C_i 表示细胞膜输入电容； R_i 表示细胞膜的传递电阻；电阻 R_i 和电容 C_i 并联模拟了生物神经元输出的时间常数，而输出 v_i 对 u_j ($j = 1, 2, \dots, N$) 的影响则模拟了神经元之间互联的突触特性；运算放大器模拟神经元的非线性特性。

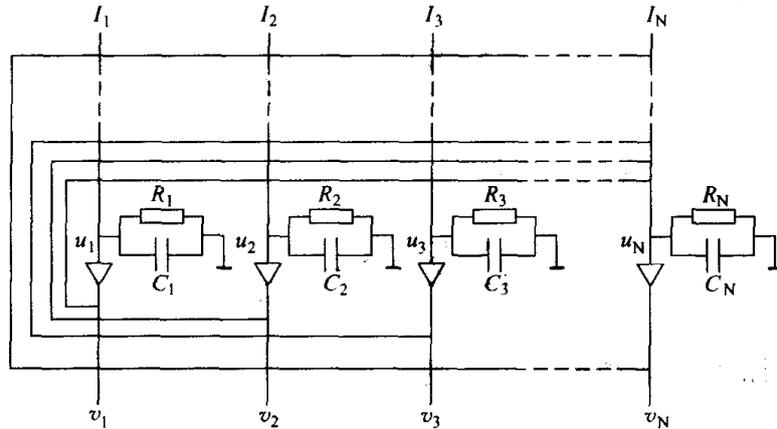


图 9-21 连续型 Hopfield 神经网络其电子线路

由基尔霍夫电流定律，连续型 Hopfield 神经网络动力学系统方程为

$$\frac{1}{R_i} u_i + C_i \frac{du_i}{dt} = I_i + \sum_{j=1}^N w_{ij} v_j \quad (9-127)$$

$$v_i = f(u_i) = \frac{1}{1 + e^{-2u_i/u_0}} \quad i = 1, 2, \dots, n$$

式中， I_i 表示系统外部的输入； $w_{ij} = \frac{1}{R_{ij}}$ 模拟神经元之间互联的突触特性； $f(u_i)$ 是放大器的非线性饱和特性，近似于 s 型函数。

连续型 Hopfield 神经网络模型，在简化了生物神经元性质的同时，重点突出了以下重要特性：

- 1) 神经元作为一个输入、输出变换，其传输特性具有 s 型函数的特性；
- 2) 细胞膜具有时空整合作用；
- 3) 神经元之间存在着大量的兴奋和抑制性连接，这种连接主要是通过反馈来实现的；
- 4) 具有既代表产生动作电位的神经元，又代表按渐近方式工作的神经元的能力。

因此，连续型 Hopfield 神经网络准确地保留了生物神经网络的动态和非线性特征，有助于理解大量神经元之间的协同作用是怎样产生巨大的计算能力的。

连续型 Hopfield 神经网络的计算能量函数 $E(t)$ 定义为

$$E(t) = -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n w_{ij} V_i(t) V_j(t) - \sum_{i=1}^n V_i(t) I_i + \sum_{i=1}^n \frac{1}{R_i} \int_0^{V_i(t)} f^{-1}(v) dv \quad (9-128)$$

与离散型的证明过程类似，可以证明，对于连续型 Hopfield 神经网络，若 $f^{-1}(\bullet)$ 为单调递增的连续函数， $C_i > 0$ ， $w_{ij} = w_{ji}$ ，那么有 $\frac{dE(t)}{dt} \leq 0$ 。而且仅当 $\frac{dV_i(t)}{dt} = 0$ ， $1 \leq i \leq n$ ，则有

$\frac{dE(t)}{dt} = 0$ 。这表明, Hopfield 神经网络的能量函数的值随着时间的推移总是在不断地减少, 神经网络趋于某一平衡状态, 平衡点就是 $E(t)$ 的极小值点。

9.3.4 神经网络在控制工程中的应用

神经网络是用来模拟脑神经的结构和思维、判断等脑功能的一种信息处理系统。神经网络之所以为自动控制界关注, 是由于它具有下述颇具吸引力的特点。

1) 很强的自学习和自组织能力, 能进行在线或离线的学习。

2) 并行处理及其带来的高速处理能力, 而且处理的时间与问题的复杂程度只是成比例关系, 而不是如串行处理中的几何数量级关系。

3) 很强的处理非线性问题的能力, 能逼近任意的非线性函数。因而适于处理那些难以用模型或规则描述的过程或系统。

4) 很强的信息综合能力, 能同时处理大量的、不同类型的定量和定性信息, 便于进行多种信息的融合。

5) 分布式存储信息和容错能力。每个神经元存储多种信息的部分内容, 部分神经元的损坏和信息破坏只会导致网络部分功能减弱。

神经网络在控制工程中具有许多不同应用形式, 包括性能优化、故障检测等。下面主要介绍两种最基本的形式, 即用于模型辨识和用于直接控制。

1. 基于神经网络的系统辨识

神经网络具有逼近任意非线性函数的能力, 所以可以用它建立非线性系统的模型。尽管还有很多关键的理论问题尚待解决, 但已有结果展示出神经网络在非线形系统建模方面的广阔应用前景。

(1) 前向模型辨识

神经网络前向建模 (forward modeling) 就是利用系统的输入、输出数据训练一个神经网络, 使神经网络具有与系统相同的输入、输出关系, 其结构如图 9-22 所示。

神经网络模型与被建模的对象并联, 建模对象输出与网络输出的差 (预测误差) 作为网络的训练信号。这种学习结构是监督学习 (supervised learning), 被建模的对象是教师, 直接地提供下个目标值 (系统输出)。

设系统由下列非线性差分方程描述为

$$y^p(t+1) = f[y^p(t), \dots, y^p(t-n+1); u(t), \dots, u(t-m+1)] \quad (9-129)$$

对象在 $t+1$ 时刻的输出值 $y^p(t+1)$, 取决于过去 n 个输出值和 m 个输入值, 选择神经网络的输入、输出结构与建模对象的输入、输出结构相同, 记网络的输出为 y^m , 则有

$$y^m(t+1) = \hat{f}[y^p(t), \dots, y^p(t-n+1); u(t), \dots, u(t-m+1)] \quad (9-130)$$

(2) 反向模型辨识

动态系统的反向 (逆) 模型在自动控制中是非常重要的。基于神经网络的反向建模 (inverse modeling) 方法如图 9-23 所示。图中, 作为对象 P 的逆模型的神经网络 C 位于对象之前, 网

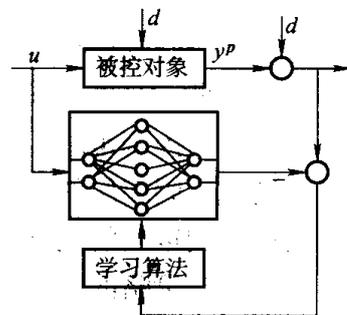


图 9-22 前向模型辨识结构

络模型的输出 u 作为被控对象的输入。若 C 为 P 的逆模型, 则应有 $y^p = r$, 否则, 学习算法根据其偏差调整神经网络 C 的权重, 使 $y^p \approx r$ 。

在该结构中也可以再包含一个被控对象的前向模型。误差信号可以取 $E = r - y^p$; 当存在噪声时也可以取 $E = r - y^m$ 。因为学习过程是基于对象的理想输出和实际输出的偏差, 所以是有目的的学习。

2. 神经网络控制器

控制系统的控制器可以由一个神经网络实现, 其一般结构如图 9-24 所示。

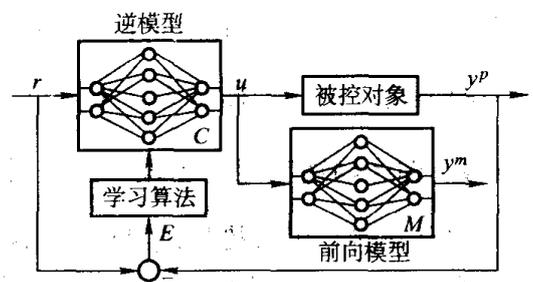


图 9-23 反向模型辨识结构

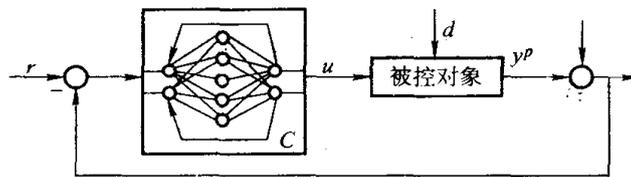


图 9-24 神经网络控制器的结构

在这种结构中, 可以用变结构理论来建造控制器, 采用学习算法进行自适应训练。Chi 在 1990 年利用 Hopfield 网络的优化计算能力, 成功地实现了对一时变线性系统的自适应控制。Guez 在 1988 年利用 Hopfield 网络的联想记忆能力, 实现了根据系统状态变化调整 PID 参数。

9.4 控制策略的渗透和结合

除了上述介绍的自适应控制、模糊控制、神经网络控制外, 常用的先进算法还包括鲁棒控制、预测控制及遗传算法等。每种控制策略的算法各有不同, 且各自有其特长, 但都在某方面存在不足。因此, 一种必然的发展趋势是各种控制策略互相渗透, 取长补短, 互济优势, 结合成复合的控制策略。这些复合控制策略克服了单独策略的不足, 具有更优良的性能, 能更好地满足不同应用的不同要求, 因而获得了更广泛的应用。可以说, 复合(混合)控制模式是先进控制策略的发展方向。

复合控制策略的类型很多, 而且随着研究工作的进展还在不断地增加和变化, 其中几种常见、应用广泛的控制策略主要有模糊 PID 复合控制、模糊预测控制、模糊神经网络控制及专家 PID 控制等。

控制策略的渗透和结合有下述特点。

- 1) 渗透面最广的是模糊控制, 它具有的复合控制策略适用性强的特点, 应用也最广。
- 2) PID 控制形成的复合控制策略吸收了其他策略性能上的优点, 保持了自身应用上的优势, 仍是一种工业上的主要控制方式。
- 3) 十分有前途的一种方法是在模糊控制和神经网络结合的基础上, 再与其他智能或传统控制策略结合形成更高层次的、性能更优良的控制策略。
- 4) 神经网络和遗传算法对其他控制策略的渗透在研究上具有很大的意义, 能形成性能十分优秀的控制策略, 在应用上尚有待进一步扩展。

一方面由于过程控制系统普遍具有大滞后、强耦合、本质非线性和存在大量的未知干扰

的特点,以及连续生产过程对安全性的特殊要求;另一方面先进控制策略本身具有的复杂性、理论研究的不完善,以及算法的普遍适用性不强等方面的问题,使得先进控制策略在过程控制中的应用仍处于一个发展完善的阶段。对于先进控制策略在过程控制中的应用与研究,有以下几点认识:

1) PID 及其变形(积分分离、抗积分饱和、带死区、自整定等)以及与先进控制策略的综合(鲁棒 PID、智能 PID 等),仍然是有效的控制策略。

2) 传统控制策略,如前馈、补偿、解耦算法往往十分有效,甚至能取得事半功倍的效果。

3) 系统结构与算法设计是构成控制系统的两大要素,控制结构的优化在某种程度上比对先进控制算法的研究更具有实际意义。一个好的控制结构不仅能够提高系统的控制性能,还能在一定程度上降低控制算法的复杂性,提高运算效率,降低运行成本。

对控制系统结构的优化可根据工艺过程的特点,采用多种方式,从不同的角度入手。以火电厂的热工过程控制为例,对基本控制回路的结构优化,就主要包括反馈信号的选取与构造、不同控制方式的切换(给水单冲量、三冲量切换;锅炉跟随、汽机跟随、机炉协调控制方式切换等)、快变过程与慢变过程的相互补偿及多种控制手段之间的协同工作(再热蒸汽温度的挡板调节、喷燃摆角调节、喷水调节的相互协调等)。

4) 从控制目标出发,综合运用各种控制方法是构成先进控制系统的有效途径。

5) 由于先进控制算法本身具有的复杂性,在其应用过程中要特别注意系统运行的可靠性,同时还要充分考虑系统的实现方式和维护成本。

9.5 先进控制策略的实现

先进控制策略的算法与传统的控制算法相比复杂得多,同时一般通用性较差,常是针对特定的控制对象而专门设计的,因此采用常规的控制仪表一般难以实现。随着一些先进控制算法的逐渐成熟,目前其中一些算法已固化在 DCS、FCS 中,用户可以像常规控制算法一样,方便地通过组态的办法进行实现。但是由于先进控制算法一般针对性较强,更多的时候需要自己编程实现。

一种编程实现的方法是采用工业控制计算机等专用设备,采用先进控制算法的系统自身构成一个独立的闭环采样控制系统。如果先进控制算法是连续形式的,实现过程中需要对算法进行离散化处理,便于计算机实现,同时注意信号隔离、采样周期选择、手/自动切换等问题。这种实现方法一般适合小型系统及要求不太高的场合,不是这里主要讨论的内容。

另一种编程实现的方式是在 DCS、FCS 等现有的成熟设备上实现先进控制算法,这样可以充分利用 DCS、FCS 的丰富软、硬件资源,构成的系统可靠性也更有保障。在目前 DCS、FCS 已在过程控制领域中普遍采用的情况下,DCS 和 FCS 已成为先进控制策略应用的主要载体,如何将先进控制算法在 DCS 和 FCS 中实现就是一个必须面对的重要问题。

9.5.1 在 DCS、FCS 中组态的实现

通过组态方式在 DCS、FCS 中实现先进控制算法相对比较简单,用户可以像常规控制算法一样,通过调用包含先进控制算法组态功能模块,按要求设置其中的有关参数,如果需要可以在线调整某些设置参数,就可以方便地实现。下面以 Ovation 系统中的 FUZZYCTR¹ 模块

为例进行具体说明。通过使用运行在 Ovation 控制器上的算法，Ovation 可以向用户提供一些高级功能，如动态前馈、Smith 预测器、模糊控制、神经网络和多变量控制等。

FUZZYCTR 算法计算的是模糊控制器的控制输出。这个算法的控制原理和可变增益的 PID 类似。通常在手工模式下，当操作员觉得用 PID 控制器调整已经难以有效地控制过程时，采用 FUZZYCTR 算法来进行控制。算法的输入是给定值和过程输出。算法的输出则是控制输出。跟踪和无振荡传送都采用与 PID 算法一样的方式进行处理。

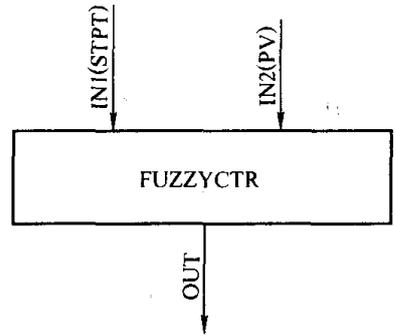


图 9-25 FUZZYCTR 功能模块

在该模块中定义的语言变量为大正数 (LP)、中等正数 (MP)、小正数 (SP)、零 (ZE)、小负数 (SN)、中等负数 (MN)、大负数 (LN)。用户可以任意选择两种常用的隶属度函数类型，包括梯形和钟形（三角形是特殊的梯形）的一种。用户通过 X5 字段（字节）的第 0~3 位来指定自己的选择。

如果是梯形隶属度函数，如图 9-26 所示，每个函数都由 4 个参数惟一标识：中心点 x_0 ，下底线的两个端点 X_{min} 和 X_{max} ，上底线的宽度 w 。这样，函数基本上就可以由两个线性方程来决定：

$$y = 1 + k_1[x - x_0(w/2)]$$

$$y = 1 + k_2[x + x_0(w/2)]$$

式中，斜率 $k_1 = \frac{1}{x_0 - 0.5w - X_{min}}$ ； $k_2 = \frac{1}{x_0 - 0.5w - X_{max}}$ 。

可以假设，中心点 x_0 、两个端点 X_{min} 和 X_{max} ，以及上底线宽 w 是可以调整的。这样，当 $w=0$ 时，梯形就变成了三角形。

如果是钟形隶属度函数，曲线的形状是由高斯分布来描述的。这样就只需要两个参数（中值和变值）。例如，当钟形曲线的中心位于 $(x_0, 1)$ 时，这个隶属度函数可以描述为如图 9-27 所示。

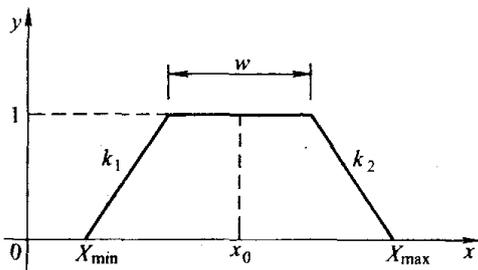


图 9-26 梯形隶属度函数

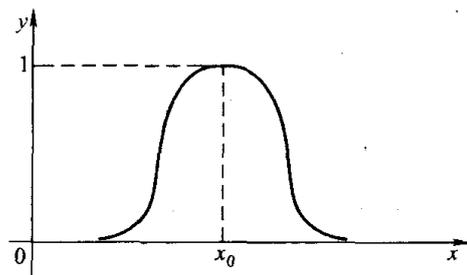


图 9-27 钟形隶属度函数

下面是一个简单定义和模糊逻辑公式的描述。

误差： $e(k) = \text{给定值 } r(k) - \text{过程输出 } y(k)$

误差偏量： $\Delta e(k) = e(k) - e(k-1)$

控制输出为 $u(k)$

以下是一些控制规则的示例：

1) 如果 $e = LP$ 并且 Δe 为任意值，则 $u = LP$

- 2) 如果 $e=LP$ 并且 $\Delta e=SP$ 或 ZE , 则 $u=SP$
- 3) 如果 $e=ZE$ 并且 $\Delta e=SP$ 或 ZE , 则 $u=ZE$
- 4) 如果 $e=Ze$ 并且 $\Delta e=SN$ 或 ZE , 则 $u=Ze$
- 5) 如果 $e=SN$ 并且 $\Delta e=SN$ 或 ZE , 则 $u=SN$
- 6) 如果 $e=LN$ 并且 Δe 为任意值, 则 $u=LN$

在这种设置下, 通常不推荐在一个单独的算法中同时使用两种隶属度函数类型 (梯形和钟形)。由于这里当前设置包括 7 个成员函数 (LP、MP、SP、ZE、SN、MN、LN), 因此需要 $4 \times 7 = 28$ 个记录字段 (类型: REAL32) 来描述梯形函数, 需要 $2 \times 7 = 14$ 个记录字段 (类型: REAL32) 来描述钟形函数。

Defuzzification (近似向心的) 控制输出如下计算:

$$u = \frac{\sum_{i=1}^n (\mu U_i)}{\sum_{i=1}^n \mu_i}$$

式中, i 是规则数目; U_i 是一个相关波纹控制输出。

规则表是可以在线调整的, 这样操作人员就可以按照实际需要添加或删除规则, 每个规则占一个算法记录的字段 (类型: INT32)。字段中的每一位的含义如下:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
规则个数				误差				0	误差偏量			控制			

比特位 12~15: 规则个数, 在当前设置下至少分配了 16 条规则。

比特位 7: 连接符, 0 表示 AND, 1 表示 OR。

比特位 8~11、3~6、0~2, 模糊度如下所示:

- 0000—LP; 0001—MP; 0010—SP; 0011—ZE; 0100—SN; 0110—MN; 0111—LN;
 - 1000—ANY; 1001—LP/MP; 1010—MP/SP; 1011—SP/ZE; 1100—SN/MN; 1101—MN/LN
- 注意: “控制” 只有 3 个比特位 (0~2)。

9.5.2 在 DCS、FCS 中编程实现的途径

由于国内在过程控制领域中使用的 DCS、FCS 种类繁多, 在工程上实施先进算法的软、硬件条件各不相同, 因此先进控制算法编程实现的途径也是多种多样的。如果根据实施算法的硬件载体来分类, 大致可以分为外挂式、嵌入式、网络混合式等三大类。

1. 外挂式

该方法是通过外挂工控机等计算机系统, 在外挂的计算机中实现先进控制算法, 将 DCS 或 FCS 作为数据采集源和输出信号的转接器, 外挂的计算机通过模拟信号硬接线连入 DCS 或 FCS, 如图 9-28 所示。

这时, 一方面要从 DCS 或 FCS 的模拟信号输入端由外挂工控机的 A/D 采集现场模拟量信号, 另一方面把计算得到的控制量信号由工控机的 D/A 送到 DCS 或 FCS 的模拟信号输入端。DCS 或 FCS 所作的就是把采集到的控制量信号转送到现场的执行机构,

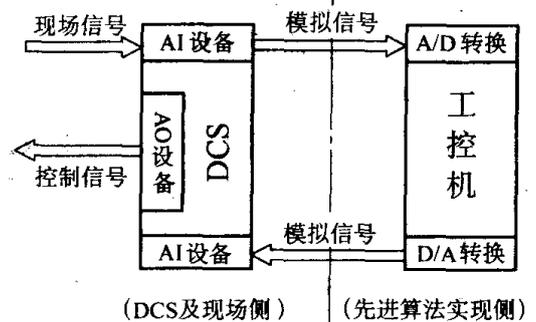


图 9-28 外挂式的先进控制算法实现形式

起到一个转接器的作用。从严格意义上说,这种方式并不是在 DCS 或 FCS 中实现了先进的控制算法,因为算法完全是由工控机完成,与 DCS 或 FCS 基本无关。这种方式的优点是不受 DCS 或 FCS 种类的限制,基本不需要 DCS 或 FCS 软、硬件的支持,适用范围广,适合于其他两类方式不容易实现时的情况;缺点是无法充分利用分散控制系统本身具有的丰富的软、硬件资源(如组态设计软件等),无法共享系统已有数据信息,同时需要进行硬接线,使安全可靠较差。

2. 嵌入式

该方法是将包含先进算法的软件直接嵌入到 DCS 或 FCS 的现场控制单元功能模块中,如 Infi-90 系统的多功能处理器 MFP、HIACS-3000 系统的 H04-M 控制器、WDPF-II 系统的分布式处理单元 DPU、MAX-1000 系统的数字处理器 DSP、IA 系统的控制处理模块 CP 及 Teleperm-ME 系统的功能模块 AS 等。这种方式的优越性是显而易见的,不仅充分利用了 DCS 或 FCS 的软、硬件资源,同时控制的实时性也能很好满足,可以使先进的控制算法较好地融入原 DCS 或 FCS 中。

这里,软件编程一般在工程师工作站上进行,运行时还需要下装到现场控制单元功能模块中。软件编程工具主要包括组态软件(如 Infi-90 的 EWS CAD 软件)、汇编语言、通用高级语言等。其中通过组态软件调用厂家已定义功能块(功能码)的方式,一般只能实现一些相对简单的先进算法,同时技术保密性较差,应用范围有限。采用汇编语言编程比较烦琐,同时支持这种编程的 DCS 不是很多,如国外大的厂家仅有 WDPF 等。与此不同的是,绝大多数 DCS 均不同程度地支持通用高级语言编程,常见的编程语言有 C、FORTRAN、PASCAL、BASIC 等。一般地,先进控制算法都较为复杂,完全由控制器组态实现不太容易,多数情况下需要使用高级语言编程,同时这样也可以方便地进行软件加密。这里支持高级编程语言的 DCS 软件开发平台一般是专用的,编程也必须遵守其特殊的规范。

3. 网络混合式

DCS 和 FCS 的一个重要特点就是具有很强的网络通信能力;可以通过网络实现整个系统的信息共享。从技术上讲,可以把一台工业控制计算机作为 DCS 或 FCS 网络的一个通信站点,一方面从网络中获取被控对象的输入信息,再利用高性能的处理器实现高级算法语言编写的先进控制算法,然后将计算输出作为上层的控制指令再由网络传输到下层的现场控制单元进行输出以控制执行机构。但是,由于连续生产过程对系统的安全性要求很高,出于对计算机网络安全考虑(如担心网络病毒、黑客程序的入侵等),DCS、FCS 的网络系统与外部系统的通信一般都有单向性的要求,即 DCS、FCS 的网络只允许数据输出,不允许输入。这样实现先进控制算法的计算机可以通过网络获取控制现场的采集数据信息,但计算得到的控制输出一般还必须通过模拟信号连接方式送入 DCS 或 FCS 的方式完成控制任务,构成所谓网络混合模式的实现形式。

通过网络获取控制现场的采集数据信息,可以充分利用 DCS 或 FCS 本身的软件资源,如控制软件、辅助设计软件(CAD 等),而硬件上投入也不大。这种方式要求系统的通信协议具有一定的开放性。为了保证这种通信的单向性,一般在 DCS 或 FCS 网络与实现先进控制算法的计算机之间需要增加带硬防火墙的路由设备进行网络隔离,保证数据不能由外部网络进到 DCS 或 FCS 中,通信协议常采用 UDP 协议。

实现先进控制算法的模拟信号控制输出一般可以由常规的 D/A 转换设备送入 DCS 或 FCS

的模拟信号输入端口,另外也可以通过 DCS 的专用 I/O 设备 RTU(远程终端设备)实现,RTU 在 DCS 内部实际也是通过模拟信号连接的。

图 9-29 所示为采用网络混合式在 HIACS-5000C 系统中实现先进控制算法的系统结构工程实例。

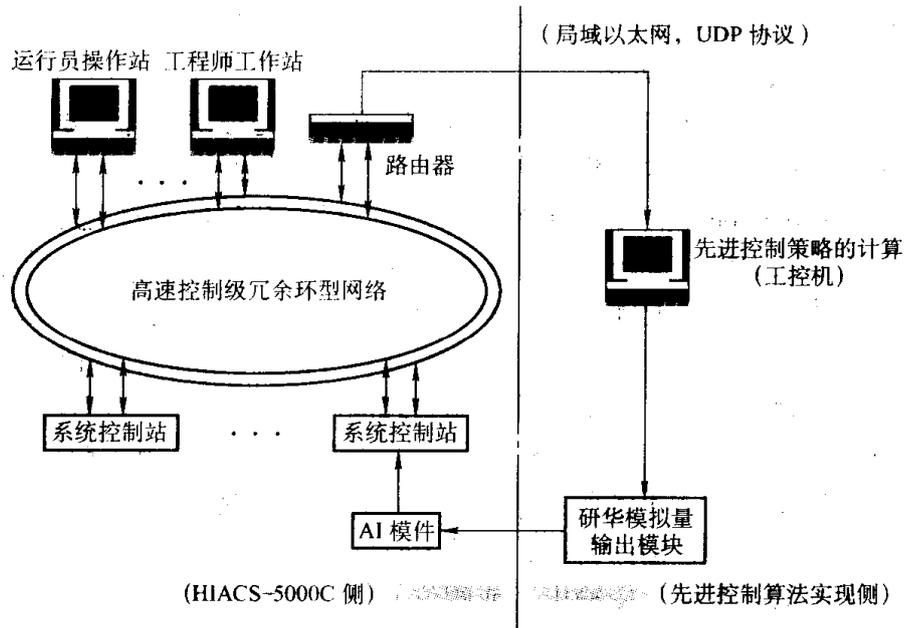


图 9-29 网络混合式在 HIACS-5000C 中实现先进控制算法的系统结构

DCS 或 FCS 所提供的网络接口可以有不同的层次。例如,底层的可以是根据厂家提供的数据库层协议,通过 NETBIOS 调用,或低层 API 函数调用直接读取网络数据。较高级一点的方式是原系统直接提供读、写实时数据库的函数,这样只需了解其实时数据库结构,在原系统设计上增加一些 I/O 点或用数据库定义一些实时数据,就可以方便地获取工业现场的实时信息。更先进的方法是基于 OPC 技术开放的接口标准,即过程控制中的对象链接嵌入技术,实现与采用 OPC 技术规范 of DCS 或 FCS 通信接口。由于 OPC 技术规范具有通用和开放的特点,这一方法普遍适用于符合 OPC 技术规范 of 分散控制系统和现场总线系统。

以 Infi-90 系统的操作员工作站系统 Conductor NT 为例,由于其采用了开放的 OPC 技术规范,底层操作系统软件为普通的 Windows NT,可以以操作员工作站作为 OPC 服务器,与普通微机(OPC 客户机)实现双向数据接口。这一接口实现硬件基础是建立在现有成熟的局域网(LAN)技术上,通信软件采用 TCP/IP 协议。具体地,以操作员工作站作为 OPC 服务器,利用其提供的一套可以在 Excel 中调用的函数,通过 VB 或 VC 的编程调用,读、写 Excel 中的过程数据,在操作员工作站上建数据读取服务器,分别对来自局域网的客户机的标签和数据读请求进行响应。OPC 客户机是对标签和过程数据有需求的一方,用户可以开发对现场数据和标签数据进行读取的 API,实现在 VC、VB、C++ 等语言环境里的自由调用。这样由通用网卡实现的通信接口,其速率较之原有的 RS-232 接口有极大的提高。

9.5.3 编程实现时需要注意的两个问题

1. 采样周期的选取

如何合理地选择采样周期是设计计算机控制系统必定会遇到的一个实际问题。一般说来,

增大采样周期将使系统的性能变差，但是可以降低对于计算机的要求。因此，工程上选择采样周期的一个基本原则是：在满足系统性能的前提下，尽可能选择较长的采样周期。

在 DCS 和 FCS 的实际应用中，如果原系统只能采用统一的采样频率，这个采样频率一般比较高，这样在实现先进控制算法时就需要进行一些特殊处理，否则计算的工作量将很大，不利于系统的稳定，可以采用多次采样平均的计算方法加以处理。

2. 多种控制方式间的无扰切换

在常规控制系统中，手动控制方式与自动控制方式间进行切换时，要求是无扰的。如果是由计算机实现的 PID 控制，自动到手动的无扰切换主要是通过手操器（或 DCS 中控制站）的硬件电路（切换时由跟随器变为保持器）实现的；而手动到自动的无扰切换除了控制器在手动状态下要跟踪手操输出（或阀位）外，控制软件还需要对算法中历史数据进行初始化处理，即主要通过软件编程保证无扰切换。在 DCS 或 FCS 中实现先进控制算法时无扰切换的问题将变得更为复杂，这是因为从安全可靠等方面考虑，一般仍需保留常规的 PID 控制算法，这样无扰切换问题就变成了至少三种控制方式间的切换，共六种切换形式。

为了简化问题，通常可以把手动状态作为中间过渡状态，这样就可简化为四种切换形式，即手动到 PID、PID 到手动、手动到先进算法以及先进算法到手动。先进控制算法与手动控制间的无扰切换问题与前面介绍的自动、手动无扰切换并无本质的不同，除了实现先进算法的软件在手动状态下需要跟踪手操输出（或阀位）外，对算法中历史数据进行初始化处理可能比 PID 算法要复杂一些，主要是要找到类似电感、电容的储能单元。同时，其他两种状态到手动的无扰切换，手操器（或 DCS 中的控制站）在切换前跟随的是 PID 算法输出和先进算法输出的二选一信号。

第 10 章 控制算法的计算机实现工程实例

本章通过三个工程实例，具体介绍以计算机为主要控制设备构成的控制系统工程实现，实现的控制算法包括常规控制算法和复杂的自适应控制算法，采用的主要硬件平台包括工业控制计算机和分散控制系统两类。

10.1 常规控制算法的工控机实现

工业控制计算机是控制计算机的主要机型之一，目前以 PC/PCI 总线为基础，将 PC 丰富的软、硬件资源与工控机结构融为一体的 IPC 已成为工业控制计算机的主流，在此基础上构架的计算机控制系统应用也越来越广泛，特别在组成一些中、小型系统方面有其独特的优越性。下面以某电厂锅炉汽包水位控制系统为例，说明如何利用工业控制计算机和 RS-485 总线构成一个常规控制算法的计算机控制系统。

图 10-1 为该锅炉汽包水位控制系统的原理框图，这是一个典型的带前馈的串级控制系统，其中汽包水位是主调节量，给水流量为副调节量，蒸汽流量为前馈量，给水指令作为控制量去控制给水泵转速以保持汽包水位的稳定。

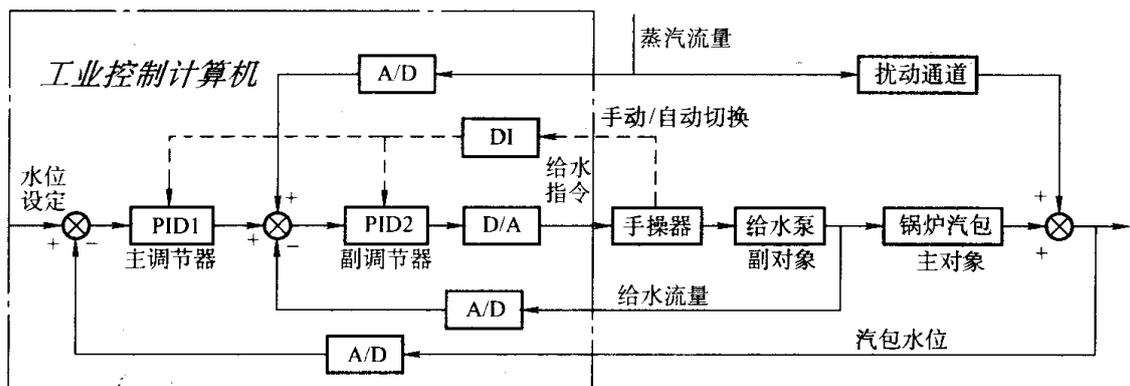


图 10-1 锅炉汽包水位计算机控制系统的原理框图

根据上述原理框图，组成如图 10-2 所示的计算机控制系统，其中控制计算机采用研华工控机 IPC-610（包含 RS-232 串口），I/O 通道选用研华 ADAM 系列远程数据采集模块实现模拟量输入、输出及数字量输入。与 PC 总线采集板相比，ADAM 系列模块具有接线方便、各模块均隔离、安全可靠的特点。这里，ADAM-4520 为 RS-232/485 转换器；ADAM-4012 为 16 位模拟量输入模块，分别采集汽包水位、给水流量、蒸汽流量和给水阀位信号；ADAM-4052 为 8 路隔离数字量输入模块，用于手动/自动切换信号输入。各个模块通过 RS-485 总线连接到通信网络上，大大减少了现场接线成本。由 RS-485 构成的通信网络采用主从式结构，IPC-610 和 ADAM-4520 作为通信的主机，发送数据采集和控制输出指令，其他 ADAM 模块作为从设备响应主机的指令，实际使用中采集和控制周期均为 1s。

在控制软件设计上,采用通用的 Windows 98 作为操作系统,控制软件采用 VC++编写而成,并利用了 IPC-610 主板提供的看门狗功能。软件通信部分,采用 MSComm 控件编程进行串行通信。软件定时部分编程采用了操作系统提供的多媒体定时器,以便实现高精度实时控制的要求。考虑到流量信号的脉动较大,包含很多高频的随机扰动,对蒸汽流量和给水流量信号进行了软件的低通滤波处理。大型火电厂从运行安全性考虑,对水位信号一般还采取冗余技术,多测点采样对信号进行多选一处理。控制软件部分,考虑到汽包水位信号干扰较强,主调节器采用了带死区的 PI 控制算法,副调节器采用 PI 或 P 调节;同时引入蒸汽流量前馈信号构成一个典型的串级三冲量给水控制系统;在手动/自动切换状态下,主调节器输出(包含蒸汽流量前馈)跟踪采样的给水流量信号,主调节器设定跟踪汽包水位,副调节器输出跟踪给水阀位信号,实现整个控制系统无扰切换。

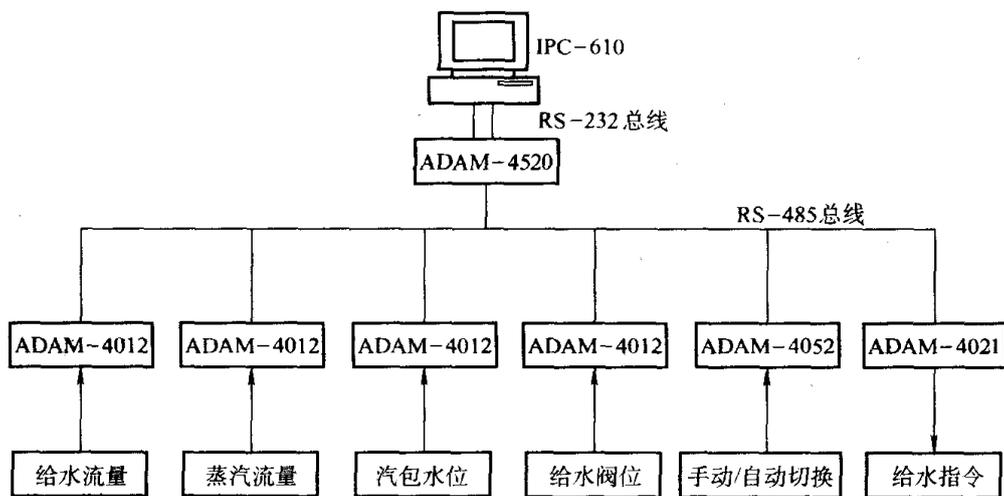


图 10-2 汽包水位控制计算机硬件组成结构框图

10.2 常规控制算法的 DCS 实现

近年来,DCS 在火电厂过程控制领域的应用已经相当普及,应用水平得到了迅速提高,DCS 从单一功能向多功能、一体化方向发展,已经实现了包括数据采集(DAS)、模拟量控制(MCS)、开关量控制(SCS)、汽轮机控制(DEH)、旁路控制(BPS)及电气控制(ECS)等多项功能,在减轻运行维护人员的劳动强度、提高火电厂的综合自动化水平、改善火电机组运行安全经济性等多方面发挥了极为重要的作用。

图 10-3 所示为某 300MW 单元机组锅炉控制部分采用 Infi-90 系统的硬件配置。下面就以上述 Infi-90 系统在锅炉主蒸汽温度控制中应用为例,给出一个在火电厂 DCS 中实现常规控制算法的工程应用实例。

10.2.1 主蒸汽温度控制方案

主蒸汽温度是单元机组主要的安全经济参数,在正常运行工况下主蒸汽温度的偏差要求控制在 $\pm 2^{\circ}\text{C}$ 范围内,动态情况下的偏差不能超过额定值的 $+5\sim-10^{\circ}\text{C}$,对控制性能要求比较高。为了克服主蒸汽温度被控对象的迟延惯性大的影响,增强系统抗干扰能力,大型单元机

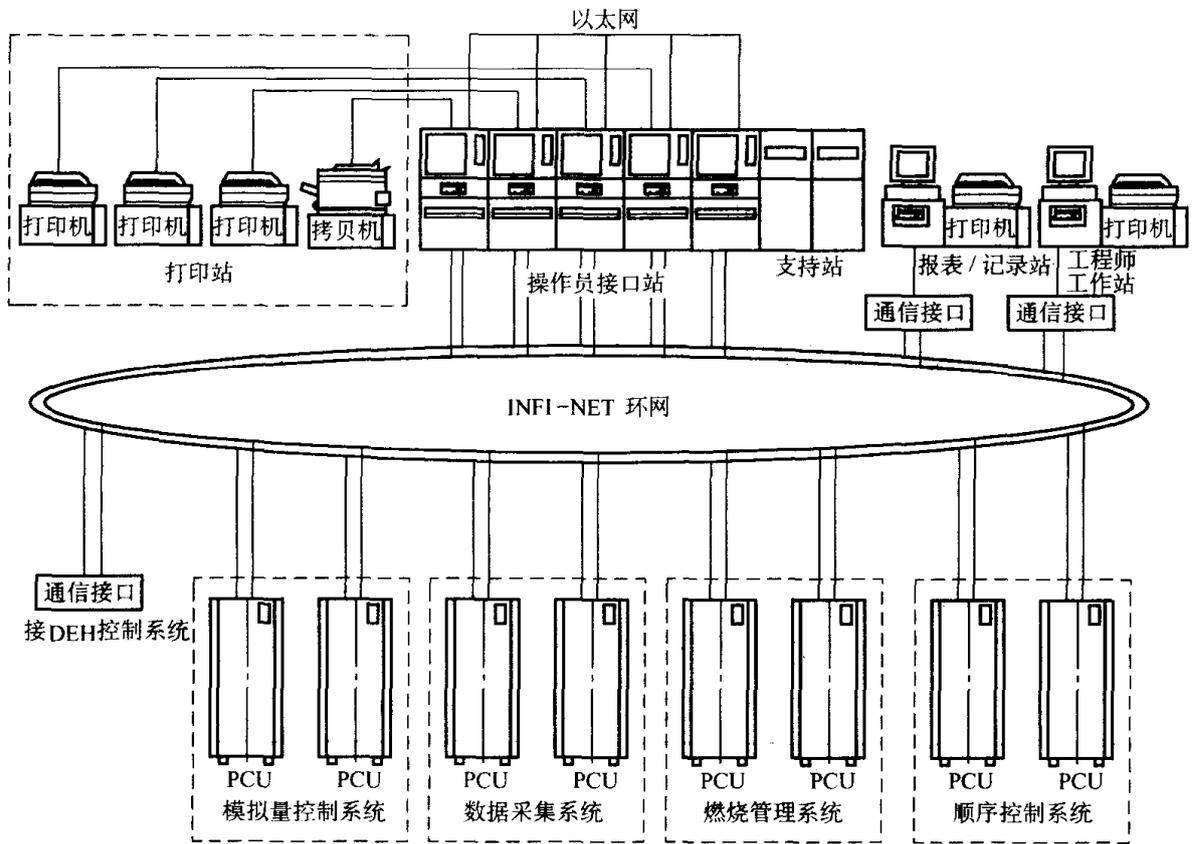


图 10-3 某 300MW 单元机组锅炉控制 Infi-90 系统硬件配置

组的主蒸汽温度控制一般采用二级喷水减温的调温方式（一级减温相当于初调，二级减温相当于细调），同时又分为甲、乙两侧进行分别控制，这样共有四个结构类似的控制回路。为了进一步克服迟延和惯性对控制的不良影响，每侧每级的喷水调节均采用了串级控制方式，图 10-4 所示为采用喷水调节的串级温度控制系统原理结构。

除了减温水量以外，影响主蒸汽温度的其他主要因素还有蒸汽量扰动和烟气量扰动，统称为外部干扰。为了提高控制系统抵御外部干扰的能力，主蒸汽温度控制系统中还采用了前馈的方法。图 10-5 为机组实际二级减温控制系统的结构（SAMA 图）；图中给出了控制回路的基本结构及调节器跟踪、手动/自动切换逻辑。

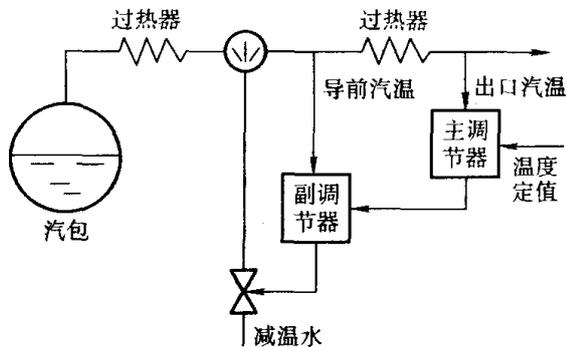


图 10-4 串级温度控制系统原理结构

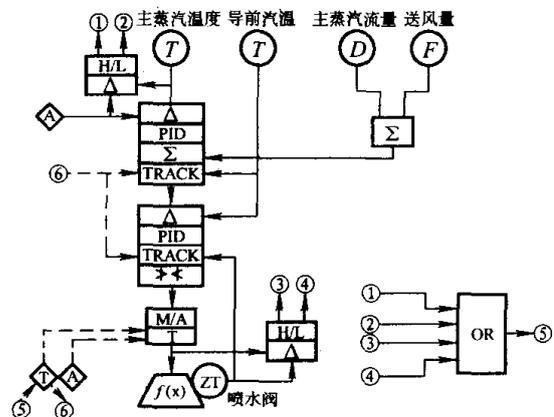


图 10-5 主蒸汽温度控制 SAMA 图

10.2.2 主蒸汽温度控制系统的 Infi-90 实现

用 Infi-90 实现上述温度控制系统时, 需要完成输入、输出信号连接、控制回路组态、数据库组态及画面组态等几方面的工作, 涉及如图 10-3 所示硬件结构中的模拟量控制系统 (MCS)、工程师工作站 (EWS) 和操作员接口站 (OIS) 等几部分。

1. 输入/输出信号连接

在上述温度控制回路中有 5 个输入信号 (即主蒸汽温度、喷水后温度、主蒸汽流量、送风量和阀位信号) 以及 1 个输出信号 (阀位指令)。在 Infi-90 系统中, 对所有的 I/O 信号都要分配 I/O 模块与端子单元, 端子单元与 I/O 模块相对应。该系统中涉及的 I/O 模块及其端子单元如表 10-1 所示。

这里使用 IMASI03 作为热电偶输入模块, 相应端子单元为 NTAI06, 用于输入主蒸汽温度信号和喷水后温度信号; 使用 IMFBS01 作为电流信号输入模块, 相应端子单元为 NTAI05, 用于输入主蒸汽流量信号、送风量信号和阀位信号; 使用 IMASO01 作为模拟量输出模块, 相应端子单元为 NTDI01, 用于输出阀位指令信号。

表 10-1 常见的 I/O 模块及其端子单元

I/O 模块	端子单元	通道数	说明
IMASI03	NTAI06	16	通用信号输入模块
IMFBS01	NTAI05	15	4~20mA/1~5V 输入模块
IMASO01	NTDI01	14	4~20mA/1~5V 输出模块
IMDSI02	NTDI01	16	开关量输入模块
IMDSO14	NTDI01	16	开关量输出模块

2. 控制模块组态

系统中采用的 Infi-90 控制模块为 IMMFP02, 它可与若干个 I/O 模块相连。控制模块中固化有 200 余种算法模块, 用户通过组态的方式生成自己的控制回路。

控制模块的组态是在工程师工作站 EWS 上通过运行组态软件来进行的。组态的过程是以 CAD 图的形式将相应模块连接起来, 生成若干页组态图。将这些组态图编译后下装到控制模块后, 控制模块就可以执行组态时指定的功能。

一般来说, 组态图中包含 I/O 模块组态 (如上述输入模块 IMASI03、IMFBS01 和输出模块 MASO01 的组态)、控制回路组态、例外报告组态及趋势组态等内容。

图 10-6 为某电厂 300MW 机组主蒸汽温度控制系统的控制回路简化 CAD 组态图。其中 APID (即功能码 FC156) 为改进的 PID 控制算法, 是一种具有相当完善功能的数字 PID 算法, 具有完善的跟踪、抗积分饱和、高低限幅、前馈输入等功能; M/A (即功能码 FC80) 为控制接口站, 提供与数字量控制站、操作员接口单元、管理命令系统和计算机接口单元等装置之间的接口, 它可以实现基本、串级和比率设定点控制以及手动/自动站转换。

上述主蒸汽温度控制采用了典型的串级控制方式, 其中主调节器采用 PID 控制, 副调节器采用 PI 控制, 有利于克服蒸汽温度对象的大惯性、大滞后特性。由于导前蒸汽温度 (喷水后温度) 的滞后时间和惯性时间常数与出口蒸汽温度 (主蒸汽温度) 相比相对较小, 副回路

作为一个快速回路，能尽快消除内扰（减温水流量）的影响，实现对出口蒸汽温度的初调，同时也有利于消除外扰影响。

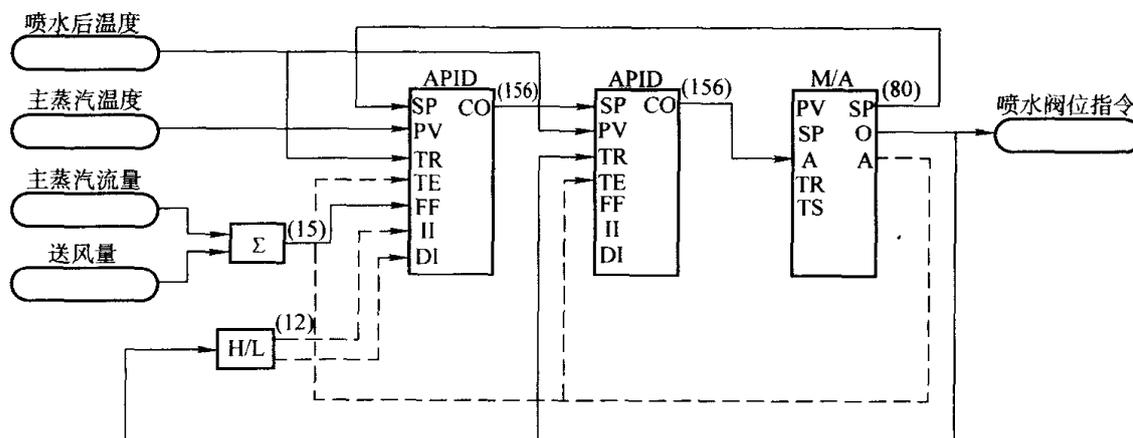


图 10-6 主蒸汽温度系统控制回路简化组态

同时，系统还引入了主蒸汽流量和送风量信号作为主调节器的前馈信号。当负荷或风量发生变化时，预先调整减温水流量，以尽快消除外扰影响。前馈系数根据风量及负荷对蒸汽温度对象的扰动试验进行整定。

此外，主调节器还采取了抗积分饱和措施，这是通过对喷水阀位指令的高低限幅块 H/L（即 FC12）的输出连接到主调节器（FC156）的 II 和 DI 端实现的。当饱和时积分作用被切除。

3. 数据库组态

凡是需要在操作员接口站上显示操作的参数都必须在数据库中进行定义，表 10-2 所示为蒸汽温度控制的标签数据库示例。

表 10-2 标签数据库

TAGINDEX 标签索引	TAGDESC 标签描述	TAGTYPE 标签类型	NUMDECP 小数位数	LOOP 环路号	PCU PCU 号	MODULE 模件号	BLOCK 块号	ALMGROUG 报警组	...
100	1MAINTEMP	ANALOG	2	1	10	5	1010	1	
102	1DESUPTM	ANALOG	2	1	10	5	1012	1	
103	1STMFLOW	ANALOG	2	1	10	5	1110	1	
104	1AIRFLOW	ANALOG	2	1	10	5	1112	2	
105	1VALVEPOS	ANALOG	2	1	10	5	1114	2	
106	1VALVEINS	ANALOG	2	1	10	5	1310	2	

4. 画面组态

Infi-90 中，操作员接口站上的所有显示操作画面均可通过工程师工作站上的图形组态软件来制作。显示操作画面中主要包括静态图形、动态参数及操作器等，通过图形组态软件中相应的工具可以方便地实现。图 10-7 为针对本例所作的一个简单的主蒸汽温度系统显示操作画面。

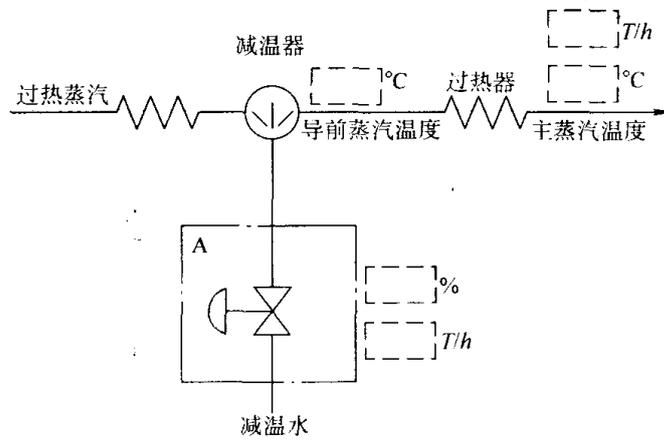


图 10-7 主蒸汽温度系统显示操作画面

10.3 模型参考自适应预估控制的 DCS 实现

10.3.1 模型参考自适应预估算法的基本设计思想

一般认为 Smith 预估方案是克服时滞影响的一种有效方法，但是其对过程模型精度和参数时不变的要求，限制了它的实际应用。在此基础上，将 Smith 预估方案与模型参考自适应技术 (MRAPC) 结合在一起，杨志远教授等提出了如图 10-8 所示的模型参考自适应预估控制方案。其中， K_p 和 τ_m 分别为增益和时滞补偿器； $F(s, T)$ 是传递函数为 $1/(Ts + 1)^2$ 的二阶惯性补偿器； $G_c(s)$ 为控制器； K_p 、 $F(s, T)$ 、 $KG(s)e^{-\tau s}$ 及 $G_m(s)e^{-\tau_m s}$ 分别是广义对象和参考模型。当广义对象与参考模型相匹配时输出反馈基于预估的特性，此时闭环系统的控制品质将等同于由 $G_m(s)$ 与 $G_c(s)$ 构成的无滞后过程的控制。在这里模型的作用是双重的，一方面作为 Smith 预估器的预估模型，另一方面作为模型参考自适应控制算法的参考模型。MRAPC 算法与典型的模型参考自适应方法不同，在一般模型参考自适应方法中参考模型是自适应控制闭环后的参考轨迹，而 MRAPC 中参考模型是自适应控制的开环被控对象之一。

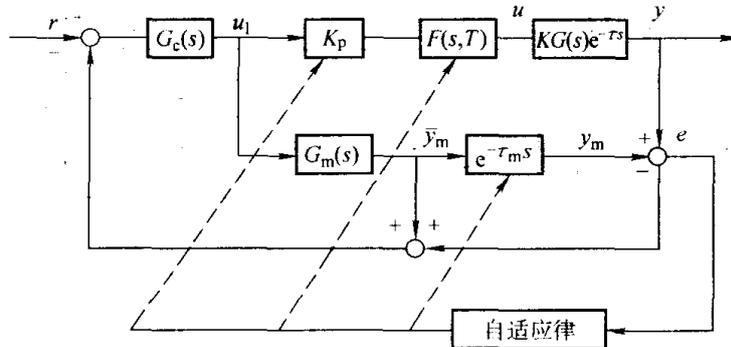


图 10-8 模型参考自适应预估控制算法基本原理框图

上述模型参考自适应预估控制方案的主要特点如下：

- 1) 利用模型降阶和模型参考自适应技术实现系统的在线模型跟踪。利用三个可表征工业

工程主要特性的自适应可调参数——时滞、增益和惯性时间，对未知或时变的被控过程参数进行在线、同时补偿，具有很强的工程实用性。

2) 该系统既无需辨识过程的所有参数，也不对控制器参数进行在线调整，具有算法简便、易于实现的特点。特别对时滞进行了在线估计和补偿，消除其对系统控制品质的影响。

3) 由于具有 Smith 预估参考模型和采用了参数在线估计算法，使得控制器参数的设计非常简单，如果采用 PID 算法，则其参数的整定可以离线进行设定，且仅仅与参考模型的惯性时间有关。

10.3.2 模型参考自适应预估基本算法

由图 10-8 可知，模型参考自适应系统的广义模型输出误差为

$$E(s) = Y(s) - Y_m(s) = U(s) \left[\frac{K_p}{(Ts+1)^2} KG(s)e^{-\tau s} - G_m(s)e^{-\tau_m s} \right] \quad (10-1)$$

式中， K_p 、 T 和 τ_m 为自适应可调参数，则可定义可调参数矢量

$$\theta(t) = [K_p(t), \tau_m(t), T(t)]^T \quad (10-2)$$

选择广义模型输出误差平方的积分作为目标函数

$$J = \frac{1}{2} \int_{t_0}^t e^2(\theta, t) dt \quad (t > t_0) \quad (10-3)$$

根据最速下降法，为了使 J 达到极小，自适应参数的调整律为

$$\dot{\theta}(t) = -c(t)\lambda \left. \frac{\partial J}{\partial \theta(t)} \right|_{\theta(t)=\theta(t_0)} \quad (10-4)$$

这里 $c(t) > 0$ 为自适应增益， $\lambda = \text{diag}[\lambda_i] \in \mathbf{R}^{3 \times 3}$ ($\lambda_i > 0$) 为加权系数矩阵， t_0 是 $\theta(t)$ 上一次被调整时间，则

$$\begin{aligned} \left. \frac{\partial J}{\partial \theta(t)} \right|_{\theta(t)=\theta(t_0)} &= \int_{t_0}^t \left[\frac{\partial y(t)}{\partial K_p(t)}, -\frac{\partial y_m(t)}{\partial \tau_m(t)}, \frac{\partial y(t)}{\partial T(t)} \right]^T e(t) dt \\ &= \int_{t_0}^t h(t) e(t) dt \end{aligned} \quad (10-5)$$

这里， $h(t)$ 是自适应系统当 $\theta(t) = \theta(t_0)$ 时的敏感度函数。

经推导可得

$$h(t) = \left[\frac{1}{K_p(t)} y(t, \theta(t)), \dot{y}_m(t, \theta(t)), -2y'(t, \theta(t)) \right]^T \quad (10-6)$$

其中， $y'(t) = L^{-1} \left[\frac{s}{Ts+1} Y(s) \right]$ 是过程输出 $y(t)$ 的实际微分。

上述模型参考自适应预估算法的优点主要有以下两点：

1) 自适应滤波表达式中的敏感度函数只含有一阶实际微分环节，有效排除了其他算法中高阶纯微分环节对敏感度函数的影响，使算法克服干扰噪声的能力较强。

2) 采用模型误差的积分型性能指标，使算法对噪声具有很强的鲁棒性。

10.3.3 算法在 Infi-90 中的编程实现

为了将上述自适应预估算法在 Infi-90 中实现, 这里采用嵌入的方式, 将包含算法的软件直接嵌入到 Infi-90 的多功能处理器 (MFP) 中。这种实现方式的优越性是显而易见的, 不仅充分利用了分散系统的软、硬件资源, 同时控制的实时性也能很好满足, 可以使先进的控制算法较好地融入原分散系统中。这里, 算法的编程在 Infi-90 的工程师工作站上进行, 运行时还需要下装到多功能处理器中。软件编程工具主要包括 EWS CAD 组态软件和专用的 C 语言编程工具 SCCU01。下面具体对在 Infi-90 系统中用 C 语言开发实现先进控制算法的方法进行介绍。

Infi-90 的 C 语言开发软件 SCCU01 包括 Microtec ANSI C Cross Compiler 和 Bailey C Utility Program 两大部分, 通过该软件可以向 Infi-90 的控制器 (如 MFP01、MFP02、MFP03 和 MFC03 等) 下载 C 语言编写的执行代码, 这样控制器可以通过调用这些代码实现各种用户自定义的控制算法。如工程师工作站上的 CAD 组态软件可以像调用其他功能模块一样调用 C 语言编写的执行代码程序, 其典型的组态和调用过程如图 10-9 所示。由图中可见, C 语言程序与 Infi-90 的功能块之间可以很方便地进行 I/O 操作, C 语言程序可以读出任意 Infi-90 功能块的输出, 这样在开发先进的控制算法时就可以利用 Bailey 的 CAD 组态软件提供的功能, 大大缩短了系统的开发周期。

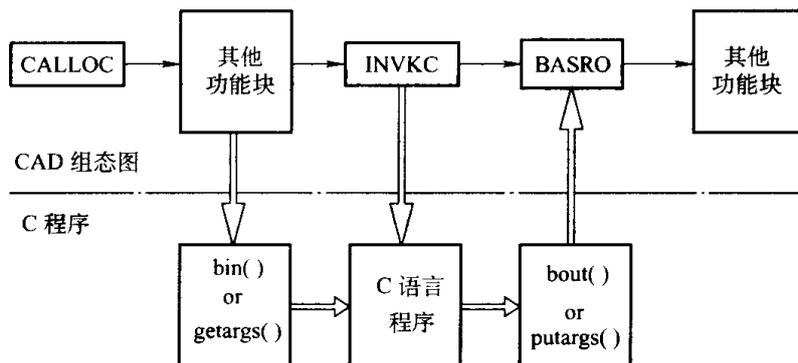


图 10-9 典型的 C 语言程序与 CAD 组态间相互调用过程示意

10.3.4 在火电厂燃烧协调控制系统中的应用

下面以某电厂 300MW 机组锅炉控制系统 (采用 Infi-90 系统) 进行的 AGC (自动发电控制) 改造为例, 对自适应预估控制算法在火电厂燃烧协调系统中的应用进行简要介绍。该改造项目的目标是使锅炉控制各子系统符合原电力部大型火电机组热控“验评标准”、功率指令接受网调 AGC 调度。这里的关键是使各控制子系统能够适应大范围负荷变动的工况要求, 长期稳定运行, 满足 AGC 调度下对锅炉响应快速性、稳定性的特殊要求, 其中的主要难点在燃烧协调控制系统。

机组实现 AGC 负荷调度控制的关键, 就是要通过大幅度提高燃烧控制系统的品质, 能满足 AGC 调度对机组负荷跟踪快速性、稳定性的要求。在燃烧协调控制系统中, 由于送风控制、引风控制和汽机控制等, 都是惯性、迟延较小的快速系统, 采用常规 PID 控制策略就可以获得较好控制效果, 而锅炉的主蒸汽压力系统是控制惯性、迟延均较大的慢系统, 惰性很强,

是实现机组负荷快速跟踪的瓶颈,该系统控制品质的好坏直接关系到 AGC 负荷调度控制的效果,是需要重点解决的问题。为此需要在燃烧协调系统的主蒸汽压力控制回路中实施模型参考自适应预估控制方案,以解决这一典型的大滞后、大惯性控制问题,满足 AGC 调度对控制品质的高要求。

1. 主蒸汽压力系统动态特性实验及模型的辨识

机组主要运行方式为锅炉跟随控制 (BF) 和以锅炉跟随为基础的协调控制 (CC) 两种方式。在 BF 方式下,由 DEH 系统闭环控制汽轮机调门来调节机组输出,主蒸汽压力由锅炉燃烧系统控制;但在 CC 方式下,DEH 系统开环,整个汽轮机作为一个随动系统由 CCS 系统的燃烧系统控制调门,达到调节机组输出的目的,主蒸汽压力仍由锅炉燃烧系统控制。主蒸汽压力系统实验时,汽轮机侧手动保持调门开度不变,锅炉侧在手动方式下给煤率指令作 3% 的阶跃扰动,记录给煤率和主蒸汽压力的变化。

图 10-10 是某负荷下主蒸汽压在给煤率阶跃扰动下的动态特性实验曲线。经辨识得到的对象模型如下。

$$\text{给煤对象: } G_1(s) = \frac{2.3}{(3s+1)^2} e^{-3s} \quad (\text{给煤率/给煤指令, 量纲: t/h}\%)$$

$$\text{主蒸汽压对象: } G_2(s) = \frac{0.13}{(100s+1)^3} e^{-80s} \quad (\text{主蒸汽压/给煤率指令, 量纲: MPa}\%)$$

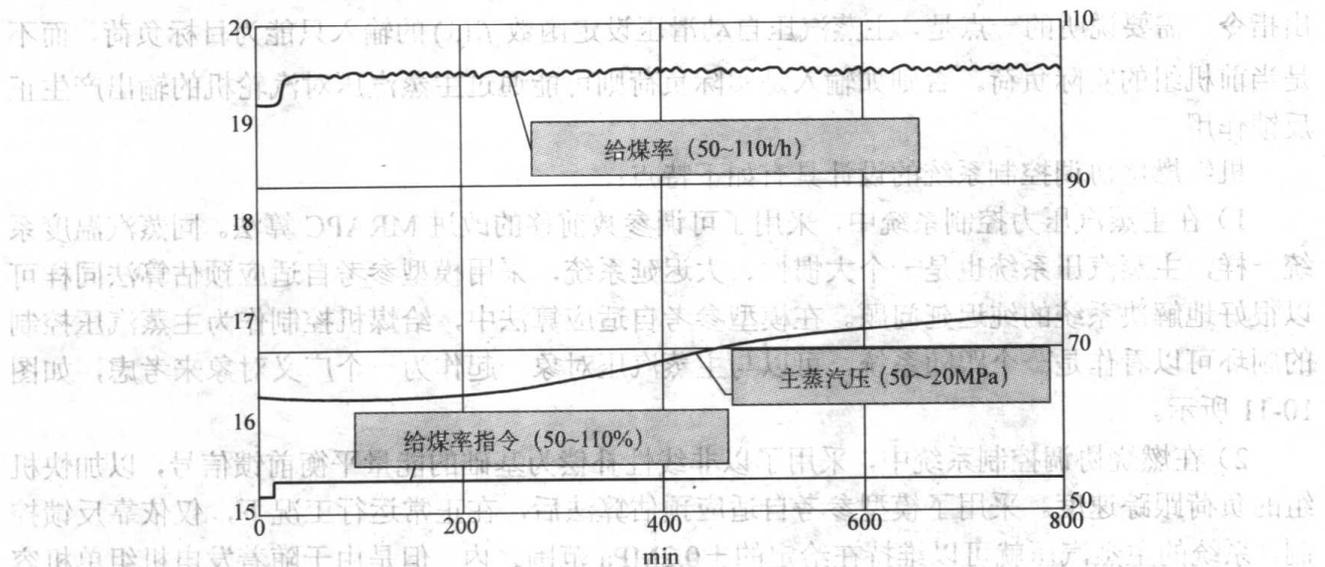


图 10-10 主蒸汽压在给煤率阶跃扰动下特性实验曲线

2. 燃烧协调控制系统的设计

该机组锅炉是直吹式的煤粉炉,主要运行方式为锅炉跟随控制 (BF 方式) 和以锅炉跟随为基础的协调控制 (CC 方式),其中协调控制是 AGC 控制的必备条件之一,机组能否适应 AGC 调度的快速和稳定的要求,主要就取决于燃烧控制系统。上述两种方式对主蒸汽压控制系统来说闭环反馈部分是基本相同的,采用了同一个 PID 主调节器;不同部分只是前馈量不同,前者采用了直接能量平衡信号(即代表汽轮机能量需求的压力比信号 $P_1 \cdot P_0 / P_1$),后者是间接能量平衡信号。机组在协调方式下主蒸汽压力自适应预估控制系统的基本原理框图如图 10-11 所示。

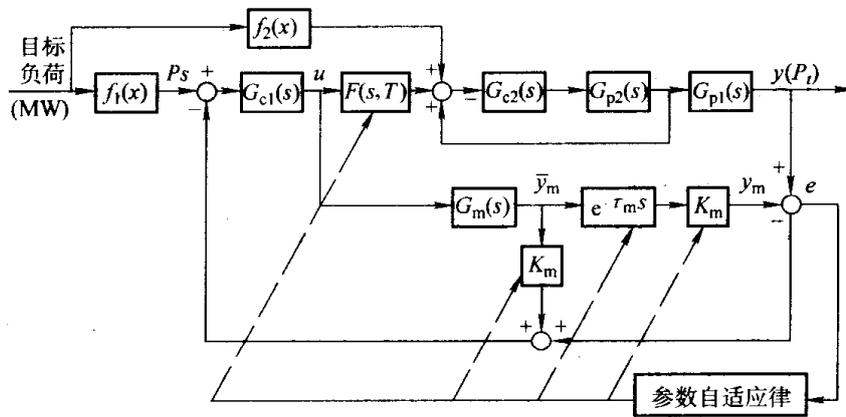


图 10-11 主蒸汽压力自适应控制系统原理框图

图中， $G_{c1}(s)$ 和 $G_{c2}(s)$ 分别为蒸汽压主调节器和给煤机副调节器， $G_{p1}(s)$ 和 $G_{p2}(s)$ 分别为主蒸汽压对象和给煤机对象， $f_1(x)$ 为主蒸汽压根据负荷指令的自动滑压设定函数， $f_2(x)$ 为负荷前馈函数发生器， y 为被控的出口蒸汽压， $K_m G_m(s) e^{-\tau_m s}$ 为自适应预估模型，根据特性实验求得其初始模型为 $\frac{0.13}{(100s+1)^3} e^{-80s}$ ，此时自适应预估模型的输入是主调节器的给煤量输出指令。需要说明的一点是，主蒸汽压自动滑压设定函数 $f_1(x)$ 的输入只能为目标负荷，而不是当前机组的实际负荷，否则如输入是实际负荷则可能通过主蒸汽压对汽轮机的输出产生正反馈作用。

机组燃烧协调控制系统的设计具有如下特点：

1) 在主蒸汽压力控制系统中，采用了可调参数前移的改进 MRAPC 算法。同蒸汽温度系统一样，主蒸汽压系统也是一个大惯性、大迟延系统，采用模型参考自适应预估算法同样可以很好地解决系统的纯迟延问题。在模型参考自适应算法中，给煤机控制作为主蒸汽压控制的副环可以看作是一个跟随系统，可以与主蒸汽压对象一起作为一个广义对象来考虑，如图 10-11 所示。

2) 在燃烧协调控制系统中，采用了以非线性补偿为基础的能量平衡前馈信号，以加快机组的负荷跟踪速度。采用了模型参考自适应预估算法后，在正常运行工况下，仅依靠反馈控制，系统的主蒸汽压就可以维持在给定的 $\pm 0.2\text{MPa}$ 范围之内。但是由于随着发电机组单机容量的增大，锅炉蒸汽机的相对热容量减少，试图利用机组本身的热容通过调节蒸汽机跟踪较大负荷的快速变化已经不太实现了，从根本上说，负荷的变化必须通过燃烧系统给煤率的变化来平衡。为了满足在变负荷条件下 AGC 调度对机组快速性的特殊要求，加快系统对负荷变化的响应速度，在燃烧系统中适当引入了能量平衡前馈信号就非常有必要了。由于有反馈作用的存在，无论是引入直接能量平衡信号，还是间接能量平衡信号，前馈量均不能采用完全补偿或过补偿的方式，否则，会使系统超调量过大甚至不稳定。因此对能量平衡前馈信号的整定，使反馈与前馈之间相互协调，以满足 AGC 调度对燃烧系统快速性和稳定性两方面的要求，是燃烧协调系统现场调试的主要工作。由于在不同负荷段机组热效率不同，这样在相同的负荷变化情况下不同负荷段能量平衡所需的给煤率变化是不同的。同时，该机组在协调方式下运行时 250MW 以下为自动滑压运行、250MW 以上为定压运行，机组在上述两个负荷段

内如果采用相同的前馈平衡信号, 系统对负荷变动的响应速度是不同的, 滑压运行时由于反馈作用较大, 动态过程中系统的给煤量变化较大, 可能引起超调。为了解决上述两个问题, 这里改变了原系统在协调方式运行时采用简单比例前馈的方案, 通过一个函数发生器 $f_2(x)$ 以实现非线性的前馈补偿, 在能量平衡前馈信号的整定过程中分别考虑不同负荷段前馈的不同情况。

3) 在协调运行方式下, 汽轮机控制中采用了常规的压力回拉回路, 以协调汽轮机与锅炉输出快慢的矛盾。由于在协调方式下汽轮机 DEH 部分处于开环状态, CCS 控制系统可以把整个汽轮机作为一个广义被控对象, 在其控制过程中应充分考虑锅炉是相对较慢的系统, 这样汽轮机的控制作用不能太快, 同时为了防止由此而引起的主蒸汽压变化过大, 在汽轮机控制回路中采用了压力回拉回路。

3. 机组燃烧协调控制系统的投运与 AGC 调度

燃烧协调控制系统中采用改进的自适应预估控制方案后, 经反复调试和一年多时间的运行表明, 控制系统的主要指标不仅达到了原部颁标准, 而且在 AGC 闭环控制过程中能很好地适应 AGC 调度对负荷的快速变化的要求。

图 10-12 是系统在协调运行方式下进行的负荷扰动试验曲线。

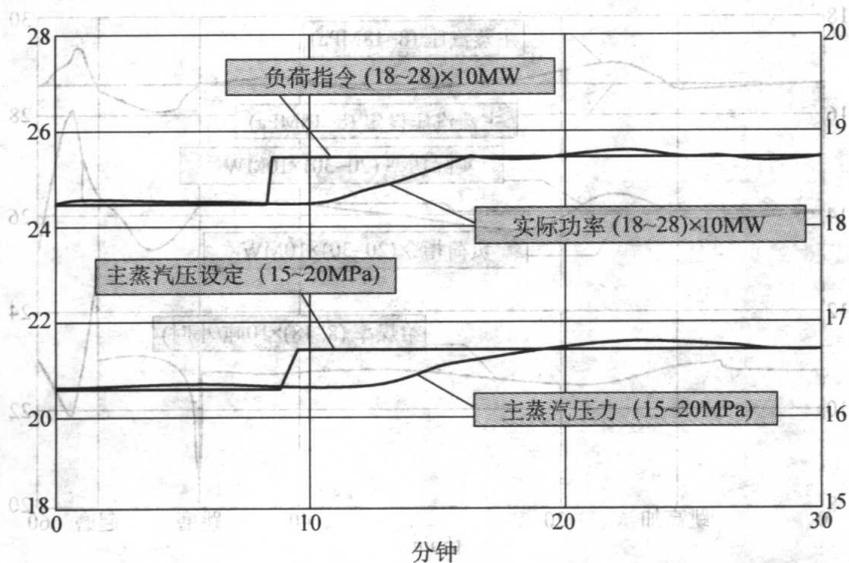


图 10-12 协调方式下机组负荷扰动运行趋势

图 10-13 所示为 AGC 闭环调度运行曲线。锅炉控制系统经自适应改造后, 燃烧和其他主要控制系统不仅能满足 AGC 调度快速性方面的要求, 而且可以在 AGC 调度情况下, 快速地跟踪负荷指令。

为了检验机组克服扰动的能力, 在各控制系统投运处于自动的条件下, AGC 投试过程中除了进行大范围的负荷扰动试验外, 机组还进行了跳一台磨和高加解列两种情况下的极限扰动试验, 试验表明蒸汽温度、燃烧等各主要控制系统均能很好地满足系统的要求, 在这种异常工况下系统在自动方式下能很快地消除干扰, 各被控参数的偏差均在要求的范围内。图 10-14 即为极限扰动试验的异常工况下, 燃烧协调系统的响应曲线。高加解列时负荷与主蒸汽压力在自动控制作用下主蒸汽压波动均较小; 在跳一台磨给煤率骤减 20t/h 的情况下, 在自动控制作用下主蒸汽压波动小于 0.5MPa, 负荷波动小于 8MW。

Images have been losslessly embedded. Information about the original file can be found in PDF attachments. Some stats (more in the PDF attachments):

```
{
  "filename": "MTE3NjA5Nzluemlw",
  "filename_decoded": "11760972.zip",
  "filesize": 33883792,
  "md5": "17a80d3cfd9184cf6be59b35c56db45d",
  "header_md5": "cf297ff7d72c69d800c16d71c7a95bdb",
  "sha1": "14a4359e18bda31f08d282d11584779ec7ebcfa1",
  "sha256": "7e16a931435e8cac430dfcf58f246173805115d2823e9bfd579a7f582c794a2b",
  "crc32": 1566538204,
  "zip_password": "",
  "uncompressed_size": 35444815,
  "pdg_dir_name": "\u2563\u00b2\u2502\u2560\u255d\u255e\u2566\u03c0\u2557\u00b7\u2510\u256a\u2553\u255e\u255d\u2591\u2567\u255a\u255c\u00b0\u2510\u256a\u2553\u255e\u2593\u2580\u252c\u2558\u2561\u2500\u2569\u2561\u2567\u2553_11760972",
  "pdg_main_pages_found": 292,
  "pdg_main_pages_max": 292,
  "total_pages": 303,
  "total_pixels": 1924927488,
  "pdf_generation_missing_pages": false
}
```