

张培仁 朱东杰 马云 史久根 著

自动控制技术和应用

— 监控网络设计 —

中国科学技术大学出版社

自动控制技术和应用

——监控网络设计

张培仁

朱东杰 马云 史久根 著

(火灾科学国家重点实验室)

中国科学技术大学出版社

2001·合肥

内 容 简 介

本书围绕自动控制技术的原理和应用,主要介绍近十年来嵌入式微控制器的发展和应用、传感器的新技术、采用新方法设计的 A/D 变换器,以及可编程逻辑阵列、大型远程复杂监控网的设计技术等。由于半导体工艺、通信技术、计算机技术、自动控制技术的飞速发展和相互融合,近年来自动控制技术的理论和应用有了很大的发展。本书中所说的自动控制技术,总的来说是指如何进行控制,用什么方法控制,而较少涉及自动控制系统中的控制理论问题。

本书的特点是强调自动控制技术的实用性和先进性,各章的内容都是近年来自动控制技术领域表现十分活跃的理论问题、技术综述和技术详解及其设计思想,书中结合了我们近年来最新的科研成果进行了详尽的阐述,使具有一定专业背景的读者能够在短期内提高驾驭控制技术的水平。

本书适合于从事自动控制技术开发研制的工程技术人员阅读,特别适合于有一定理论和实践经验的研究人员,也可以作为高等学校理工科大学或研究生学习自动控制技术的教学用书。

图书在版编目(CIP)数据

自动控制技术和应用:监控网络设计/张培仁等著. —合肥:中国科学技术大学出版社, 2001. 9

ISBN 7-312-01301-5

I. 自… II. 张… III. 劳动—安全监察—自动控制系统 IV. X924.3

中国版本图书馆 CIP 数据核字(2001)第 048455 号

中国科学技术大学出版社出版发行

(安徽省合肥市金寨路 96 号,邮编:230026)

中国科学技术大学印刷厂印刷

全国新华书店经销

开本:787×1092/16 印张:17 字数:400 千

2001 年 10 月第 1 版 2001 年 10 月第 1 次印刷

印数:1—1500 册

ISBN 7-312-01301-5/TP·276 定价:25.00 元

前 言

多年来我们一直从事计算机控制技术的研究,特别是对大型远程复杂监控系统和嵌入式微控制器更有兴趣。它们大量应用在水坝水库、火灾安全、大型远程控制系统中。目前国外控制技术已开始从集散式控制系统(DCS)向现场总线控制系统(FCS)过渡,21世纪控制系统将是现场总线的天下。自动控制技术、计算机技术、通信技术和传感器技术,相互融合,又相互促进。它们的硬件基础(超大规模集成电路)也有飞快的发展,反过来又促进了自动控制技术的进步。对于从事自动控制技术的工程技术人员和科研人员,在系统设计思想、采用的技术路线、使用的开发工具以及使用芯片的功能和集成度等,与十年前有很大的不同。因此,我们认为有必要编写一本能反映当前自动控制技术发展现状的书。这样,一方面是我们研究室长期科研工作的总结,另一方面也是目前大型远程监测和自动控制技术教学的需要。

本书共分16章。全书策划、审稿由张培仁教授负责。第1章、第2章、第3章、第6章、第9章由张培仁教授亲自执笔。第5章、第14章由朱东杰老师执笔。第8章、第10章由史久根执笔。第7章由都改欣执笔。第11章由刘蓬执笔,第12章、第13章由孙占辉执笔,第4章由徐燕执笔,第16章由周延萍执笔。第15章由马云老师执笔。

本书所有硬件实验由马云老师负责。

朱东杰老师负责全书编辑、排版工作,马云老师负责大部分文字输入工作。

对为本书的出版有贡献的黄竣、王洪波、高飞、陈智勇、王铜、罗国娟、吴岱昶等同志表示衷心的感谢!

中国科学技术大学范维澄副校长、火灾科学国家重点实验室王清安教授和张和平副主任,对本书的编写过程给予了热情支持和指导,在此表示衷心的感谢!

中国科技大学自动化系博士生导师孙德敏教授也为本书做了指导性工作,并为完善CAN现场总线提供了实验室和实验设备。

由于时间和水平有限,本书不足之处,请读者谅解。

张培仁

2001年6月20日于中国科大

目 录

第 1 章 嵌入式微控制器技术概述	1
1.1 嵌入式微控制器的新进展	1
1.2 嵌入式微处理器的开发平台	2
第 2 章 智能传感器和智能执行机构的原理及技术	5
2.1 智能传感器技术的发展和特点	5
2.2 传感器的基本性能	6
2.3 智能传感器功能实现方法	8
2.4 智能执行机构	11
第 3 章 A/D 转变换器新发展及 $\Sigma-\Delta$ A/D 设计实例	15
3.1 主要 A/D 变换器的特点	15
3.2 $\Sigma-\Delta$ 模数转换器(ADC)基本原理	18
3.3 AD7710 接口设计	24
第 4 章 EPAC 技术概论及其应用	31
4.1 引言	31
4.2 EPAC 技术概述	31
4.3 EPAC 应用实例 IMP50E10 与桥电路的连接	34
4.4 小结	36
第 5 章 FPGA 技术概论及其应用	39
5.1 绪论	39
5.2 FPGA 技术概述	42
5.3 FPGA 应用实例	45
5.4 FPGA 的前景展望	47
第 6 章 现场总线控制系统概述和技术特点	49
6.1 控制系统发展简介	49
6.2 现场总线的定义、特点及它与 DCS 的区别	49
6.3 ISA/SP50	53
6.4 FCS 需要解决的主要问题	54
6.5 控制器局部网(CAN)	63

6.6 局部操作网络 I/O 的特点	68
第 7 章 CAN 现场总线技术概述	81
7.1 控制器局域网(CAN)	81
7.2 CAN 总线数值	81
7.3 非破坏性的总线仲裁	82
7.4 多主与多节点接收	83
7.5 传送速率与距离	83
7.6 数据安全性	84
7.7 帧类型及其格式	85
第 8 章 CAN 现场总线扩展技术	87
8.1 问题的提出	87
8.2 可行性分析	87
8.3 系统的结构模型	88
8.4 系统软件开发平台	90
8.5 系统硬件开发平台	92
8.6 简易在线仿真平台	94
第 9 章 基于 16 位微处理器的 CAN 网关设计与实现	99
9.1 软硬件设计平台	99
9.2 网关的基本用途	99
9.3 网关的软硬件基本架构	100
9.4 基于 XA 的实时多任务内核的设计	104
9.5 上位机与网关间通信接口硬件的设计	119
9.6 在系统编程(ISP)实现	120
9.7 CAN 与 XA 的接口设计	122
9.8 485 总线硬件接口设计	123
9.9 实时时钟/日历芯片 MC146818 软硬件接口设计	124
9.10 CAN 中继器的设计与实现	128
第 10 章 总线互连技术及实现	133
10.1 若干约定	133
10.2 CAN 总线链路层设计	134
10.3 485 总线链路层设计	137
10.4 网络层协议	141
10.5 上位机与网关之间的接口通信协议与软件设计	148

第 11 章 CAN 总线与 PC 机的 EPP 接口	153
11.1 独立的 CAN 接口芯片	153
11.2 增强型并行口(EPP)	159
11.3 虚拟设备驱动程序 VxD	167
11.4 PC 机的监控节点程序的设计	172
11.5 附录	175
第 12 章 CAN 总线接口技术	187
12.1 CAN 串行链接 I/O 器件 P82C150	187
12.2 CAN 总线远程控制节点	197
12.3 CAN 总线控制步进电机节点的具体实现	207
第 13 章 CAN 现场总线系统的智能接口模块	211
13.1 概述	211
13.2 智能接口的通信协议及其公共部分	212
13.3 开关量输入/输出模块	218
13.4 模拟量输入/输出模块	223
13.5 应用实例	227
第 14 章 CAN 总线扩展应用实例	229
14.1 系统设计背景及总体思路	229
14.2 实验系统的设计	230
14.3 用户控制的实现	237
14.4 实验系统的安装与调试	239
14.5 系统评估及展望	240
第 15 章 本安系统及远距离二线制总线供电	243
15.1 现场总线与本质安全	243
15.2 现场总线物理层等效电路分析	245
15.3 远距离二线制总线供电及信号传输	248
第 16 章 安全系统应用设计的实例	257
16.1 二总线火灾报警系统的应用	257
16.2 QA840119 在二总线火灾报警系统中的应用	257
参考文献	263

第1章 嵌入式微控制器技术概述

1.1 嵌入式微控制器的新进展

嵌入式控制机主要分两类:一是嵌入式微控制器,二是嵌入式微处理器。嵌入式微控制器(Embedded Microcontrollers)常在一硅片上集成CPU、定时器、中断控制器、I/O口、串行口、看门狗、存储器及其它专用功能。如A/D、D/A、PWM口、I²C口和CAN网络口等。我国称为单片机系统。国内这种叫法主要是侧重于功能来讲的。因为构成计算机的三要素CPU、存储器、外设集成于一个芯片之中,所以称单片计算机。它追求的目标是整体化、小型化和廉价。

嵌入式微处理器(Embedded Microprocessors)与前者区别在于:后者是原微处理器发展起来的,它不附加内部存储器。它依靠与片外存储器经过三总线连接起来。指令系统也是原微处理器的指令系统,不需要重新学习。这类芯片有80188,80186,8680,V20,V40等。数字信号处理(DSP)也属于后者。本书主要讲嵌入式微控制器。

目前世界上集成电路生产厂家几乎都有自己的微控制器系列产品。不同厂家微控制器的指令不同,开发装置也不兼容。但是从近年的发展来看,8051系列可能最终形成微控制器的工业标准。这是由于Intel公司向不同厂家转让8051微处理器的生产权。目前生产的8051系列有百余种派生芯片。它们既保留8051核心结构又增加各个厂家赋予的专用功能,或在原来功能的基础上给予扩充。这样,芯片速度有所提高,功耗低,封装尺寸上也有很大改进,这样就形成了广泛的系列。同时大部分用户已习惯使用它,市场上有大量的软件包和硬件接口可供用户选用。

90年代以后逐步开发的16位微控制器80C251和80C51XA与8051指令兼容,这有利于以前开发的程序库重复使用和推广。

微控制器8051系列的C编译器经过多年实践日趋成熟,达到专业化水平的C编译器,也已被广大用户所认可。另外一个原因是只有8051系统被多家公司成功研制出实时多任务操作系统(RTOS)。它给用户有效管理应用程序的运行、合理分配CPU时间和资源都带来极大的方便。为用户解决复杂应用系统、软件产业化等问题带来好处。开发8051系列的公司很多,目前芯片价格很低,因此可选择性也很强。

16位微控制器MCS-8096是在80年代中期制造出来,世界著名的微控制器生产商Intel和Philips于90年代初分别推出16位机80C251和80C51XA两个系列。Intel公司又把原来8096进一步改造,产生了80C196/296的新产品。Siemens生产的RISC系列的SAB167和面向工业控制的C166系列。

嵌入式微控制器在处理速度、寻址能力、更好的兼容性、支持多任务等方面都有很大进步。在速度上,已达到20MHz以上的处理速度。这主要是因为结构上采用流水线设计,一般同时可以执行几条指令,单周期执行也缩短到100ns,中断响应超过400ns,有的CPU还增加了DSP处理器。在寻址能力上,各公司都达到24位线性空间寻址。在寻址方式上,支持间接、扩

展、相对和位寻址。一般都提供两个地址空间。即程序、数据空间。芯片中资源越来越丰富,有 FLASH ROM, RAM, A/D, PWM, I²C, CAN 等专用功能模块。OTP EPROM 广泛使用。RAM 最多可以到几 K, 有的芯片有二个双工 UART。芯片的电源管理越来越合理, 功耗越来越低。FLASH ROM 一般写 1000 次没有什么问题。工作电压具有灵活性, 一般 2.7V~5V 都可以工作。为了适应移动通信的低电压, 芯片开始采用片上外设接口的电源智能化管理技术。

微控制器性能大幅度的提高, 由此也带来单片机开发水平的提高。

1.2 嵌入式微处理器的开发平台

单片机自 70 年代末出现以来, 其应用得到了迅速的发展。以往开发单片机所用的基本工具是仿真器、汇编器、编程器。80 年代的 8051 系列、6805 系列、96/98 系列其内部资源基本在 64K 代码范围内, 而大多数应用程序的代码在十几 K 以内。进入 90 年代, 随着大规模集成电路技术的发展, 世界各主要半导体公司推出面向 21 世纪的微控制器, 如 INTEL296, 251, PHILIPS P51XA, MOTOROLA68300 系列等。这些芯片采用流水线结构和宽总线(16 条数据总线或是 2 个内部总线), 寻址能力可达到 6M~32M 字节, 片内 ROM/EPROM/FLASH ROM 可达到 28K~512K 字节。为了充分有效地开发这些微控制器的内部资源, 提高应用系统的性能指标, 对单片机开发平台的要求也有了很大提高。

目前单片机开发平台指的是什么? 由实时多任务操作系统(RTOS)、基于嵌入式系统的工程 C 语言编译器(COMPILER)、实时在线仿真器(ICE)组成的高性能工具集合即单片机开发平台。单片机开发平台是使走向标准化、产业化、降低风险、提高效率的有效手段。以往要实现单片机的开发工作, 对人才有多方面的要求: 既要精通软件、硬件又要具有系统设计概念, 还要有长期工作经验, 具备一定电子线路工艺方面的知识和经验。一般要大学毕业又工作多年的工程师才能胜任。在开发单片机中另一个大问题是因基于汇编语言的编程、开发, 只能孤军作战, 设计出来的代码可读性差、可继承性差, 是一种手工小生产方式。基于汇编的开发, 参加的人增多, 效率反而下降。这种开发对于比较简单的系统还可以应付, 但要面对 21 世纪的高性能微控制器, 将会遇到很大的困难。这是由于开发工具的落后。采用单片机开发平台, 就是 RTOS 替代了单片机应用程序中的主程序, 可以使多个程序员并行工作, 写出相对独立的应用任务, 也可以在 RTOS 的基础上购买专家库函数, 组成高效、可靠、有竞争力的嵌入式系统。在开发方式上产生根本的变化, 将系统分析员、系统设计员与程序员分离。系统分析定义应用系统的功能要求, 系统设计员设计电路硬件并在 RTOS 基础上定义软件功能模块。程序员只要有 C 语言的基础, 就可以在开发平台上完成复杂的应用系统。这样对程序员的设计要求降低了, 分工更加明确, 一个复杂系统可以由多人共同开发。使嵌入式软件程序的开发进入规模和产业化生产。高性能的实时在线 ICE 具备实时映象存储器(RAM), 测量数据跟踪功能, 对于复杂程序调试是必不可少的。

嵌入式系统工业是一种技术密集、资金密集、然而却又是高度分散的工业, 它不可能像 PC 软件产业那样垄断在少数大公司手中。这就给我们带来巨大机会和发展前景。

1.2.1 实时多任务操作系统(RTOS)

目前嵌入式应用一个发展倾向是采用实时多任务操作系统(Real Time Operation System, RTOS)。过去一个单片机应用程序所控制的外设和执行的任务不多,采用一个主循环和几个顺序调用的子程序模块即可满足要求。目前单片机芯片本身的性能大幅度提高。硬件可以适应复杂系统的要求,问题在于软件上。一个微控制器同时控制/监视很多外设,要求实时响应,有很多任务要处理,各个任务之间有多种信息传递,如果还用主循环若干子程序(或中断)解决会存在两个问题:其一是某一中断可能得不到响应或者处理时间过长,这对于一些控制场合是不允许的。例如,微控制在同时接收几个通道的并行或串行数据通信时,因为某一个通道通信时间过长,而其它通道来不及响应而丢失数据。这时解决的方法就是降低整个通信流量。其二是系统任务多,要考虑各种可能性也多,各种资源如调度不适当,就会发生死机,降低了软件的可靠性。实时操作系统是一段在嵌入式系统启动后,首先执行的背景程序,用户的应用程序是运行于RTOS之上的各个任务。RTOS根据各个任务的要求,进行资源管理、消息管理、任务调度、异常处理等工作。各个任务有一个优先级,RTOS根据各个任务的优先级,动态地切换各个任务,保证对实时性的要求。工程师在编程序时,可以分别编写各个任务,不必同时记住所有任务运行的各种可能情况,这大大减少了程序编写的工作量,而且减少了出错的可能性。

实时多任务操作系统,以分时方式运行多任务,看上去却好像是多任务“同时”运行。任务之间的切换当以优先级为依据,只有优先级服务方式的RTOS才是真正的实时操作系统,时间片方式和协作方式的RTOS并不是真正的“实时”。一般RTOS其主要功能块包括任务管理、事件管理、定时器管理、报文管理、循环队列管理、固定存储块管理、UART管理、自动掉电管理等。这些都是RTOS应具有的基本功能。RTOS也体现一种开放式软件框架。工程师更改某一个任务的程序对其它任务影响很小。并且一个复杂任务可以由多个工程师同时开发。各人之间只要制定好规则和协议即可。这样既缩短开发时间又减少软件产品对某个人的依赖性。另外RTOS的可继承性也比较好,例如,以前人们所开发的软件或者已成为产品的函数库,现在任何一个编程工程师都可以继续使用和充实,使软件越来越完善。从而使嵌入式的软件从“个体劳动”向“大生产方式”转变。

实时多任务操作系统一般以源代码方式供应用户,最终在用户代码中占有一部分。例如CMX的RTOS在80C51XA的最终代码只占3310字节。在30MHz主频情况下最大中断潜伏时间为 $2.9\mu\text{s}$ 。

嵌入式系统应用复杂化的一个要求是C语言编译器。目前国际上C编译器已作为标准产品提供给工程师使用。RTOS一般也是要在C语言编译器下使用的。把C语言从PC机移植到嵌入式微控制器上会遇到两个问题。其一是微控制器种类繁多,一个系列中的品种就很多,而对各种微控制器可通用的C编译器却很少。一般是一种CPU配一种编译器。其二是要考虑编译效率。即完成同样的任务、产生代码的数量、产生系统数据存储器的数据、执行时间等几个指标。嵌入式微控制器一般要求实时性要好,存储容量要大的C编译器。为了提高C编译器的效率,只有针对具体单片机的具体特点设计C语言编译器才能保证编译的效率。此外编译效率对不同问题会有不同的效率。因此往往要用多种程序进行测试,综合比较,最后作出一个综

合评价。目前流行 TASKING 4.0, KEIL5.02, AR4.23, PLC2.22 等 C 编译器。TASKING 和 KEIL 的 C 编译器效率较高。

1.2.2 实时在线仿真器(ICE)

在调试嵌入式 MCU 应用程序时会出现编程错误。其一是程序格式错误,或使用非法语句等语法性的错误。这些错误在编译时可以发现并可以纠正。其二是非语法性错误。如 I/O 口定义和使用错误、逻辑顺序的错误以及硬件接口使用错误。这些错误只有在目标系统上调试才能确认和定位。因此需要一个性能优越的仿真器来进行调试。当前 16 位 MCU 的仿真器种类很多,高中低档都有,连接方式一般是通过 PC 机的串行口或并行口与仿真器硬件相连。

一般应用仿真器的用户所关心的问题如下:

能否进行实时在线仿真。

仿真器占用 MCU 资源有多少。

对 C/C++ 及 RTOS 的支持程度。

不同的仿真器在这几个问题上差别很大。一般仿真器采取通用商业芯片作为仿真 CPU,在仿真的同时要处理与 PC 机之间的通信,做不到实时仿真,而且还要占用一定的系统资源。如串行口、中断、I/O 等。这种仿真方法是用户应用程序和监控用户仿真器程序交替工作,控制权绝大部分时间掌握在仿真器上。用户程序执行时并不实时,而是执行一段用户应用程序,返回监控一段时间进行与 PC 机通信,再执行一段用户应用程序,这样循环下去。另外一种仿真器是用微控制器厂家生产的专用芯片作为 MCU。该芯片增加引脚将芯片内部关键数据引到片外,在不影响芯片运行情况下监测其运行情况。这样仿真器可以在不占用系统任何资源的情况下实现全速实时仿真。

目前高档的仿真器一般可以实现实时在线仿真,有专用静态 RAM 作为目标系统的存储器的映象,使工程师在不干扰目标系统全速运行的情况下观察到存储器中数据的实时变化。此外,仿真器还有跟踪的功能。记录总线上的所有信号和指令的执行情况。记录的数据存入跟踪存储器中,在调试界面的窗口看到地址数据、总线状态、外部被跟踪信号、I/O 口数据传送情况等信息。还可以设置复杂的触发逻辑,给用户指令执行报告,指出各程序段的准确执行时间、执行顺序等。

总之,新型 16 位微控制器针对的是工业应用复杂化的要求而设计特点突出在:流水结构, DSP 处理器,大的寻址空间和丰富的外设。用 C 语言编译的仿真器,实时操作系统及有关函数库等开发工具组成了 16 位单片开发机的平台。

第2章 智能传感器和智能执行机构的原理及技术

2.1 智能传感器技术的发展和特点

传感器是获取信息的工具。信息技术的三大支柱是传感器和控制技术、通信技术与计算机技术。传感器应用遍及军事、科研、工业、农业等各个领域,它是非常重要的基本元件。

以前的传感器主要是手工制造或者用常规机械工艺加工,所以存在一些严重不足。主要表现在:尺寸大;频率响应特性差;输入—输出特性存在非线性,随时间有温度漂移;信噪比较低;大部分都是模拟信号的输出。

自动控制系统的飞速发展对传感器进一步提出了数字化、智能化、网络化、标准化的要求。集成电路的发展,半导体技术的进步,各种 CPU 的小型化,这些新技术与原有传感器的传统技术相结合,形成了智能传感器的技术特点。

现场总线在 20 世纪八十年代末,九十年代的发展,又使传感器不仅仅带有 CPU(微处理器功能),而且很容易与监控网接口。这使智能传感器又有了一个重大变革。

2.1.1 智能传感器应具有如下主要功能和特点

智能传感器是高精度的。它可以通过自动校正去除零点的漂移。与标准的参考基准实时地对比以自动进行整体系统标定,即自标定自校正的功能。它还可以通过采样数据进行统计处理来消除随机误差,从而提高信号信噪比。智能传感器与上位机通讯经常使用数字通讯,因此也就消除了因传送而产生的误差。采用这些技术的智能传感器比传统的传感器具有更高的精度。

2.1.2 智能传感器的高可靠性和高稳定性

传感器受环境影响是很大的。例如,环境温度变化,可能会使传感器产生温度漂移,这种漂移包括零点漂移和灵敏度放大倍数的变化。智能传感器除了自校正自标定以外,还能实时地自动进行自我检验、分析、判断所采集数据是否合理,并对异常情况进行处理、提示。也可以自动改变量程,得到最合适的(即最好精度的)数据。

智能传感器具有存储、记忆与信息处理功能,因此可以通过 CPU 进行编程、数字滤波以及相关分析等处理,去除输入数据中的噪声,提取真正有用的信号。它可以通过数据融合,消除多参数相互交叉影响,从而保证高信噪比、高分辨能力。

智能传感器自适应能力较强。它具有多种供电方式,又使系统的整体功耗大大降低。对于不同状态下的灵敏度、输入方式、数据滤波调整以及输出走向,可以通过网络上软件对不同已编好的软件做相应调整。当然因为智能传感器本身也带有 CPU(微处理器),所以也可以做一

些局部小的调整。

智能传感器性能价格比高。从整个系统来分析,智能传感器价格并不比传统的传感器高,而且大大减少了中间控制器的开销。另外智能传感器与监控网的都一般只有二条线。传感器,执行机构以及网桥连接是由二条线连接起来。这样可以减少 40%~60% 的连接线,这对大型远程复杂的监控网更有意义。

智能传感器实质上是一个系统。它是传感器、放大器、A/D 变换器、微处理器、通信技术的结合。当前它主要有两个研究方向,即智能传感器和微传感器。前者重点在“智能”上,后者既要“智能”又要“微型”。前者是把传感器,调理器,A/D 及 CPU 控制器,网络接口集成组合在一起,当然还有相应软件程序。后者把全部或者部分集中在一个硅片上。两者功能上有很多相同、相似的地方。但集成程度不一样。后者实质上把传感器技术与半导体工艺结合在一起。本节主要从功能上讲前者。前后两者并无明显界限,而且随着时间发展,两者既相互借鉴又相互融合。

综合上述,传感器技术是一个多门学科的综合技术,对应用传感器而不是设计传感器的技术人员来说,只有对智能传感器的发展、功能及特点有了一定的了解,才能更好地使用它。

2.2 传感器的基本性能

对传感器的基本性能的研究主要基于两个目的:其一,用作一个测量系统。只有了解其基本特性才能从输出信号推断出输入信号的变化规律。也就是了解其基本特性以便更好地使用传感器。其二,用于传感器本身的设计和研究。大部分工程技术人员主要是使用和选用传感器,而很少设计一个新的传感器。为此我们简单介绍传感器静态特性和动态特性,而不介绍制造传感器所用的工艺。

2.2.1 传感器的静态特性

传感器的静态特性主要体现在迟滞、重复性、线性度、精度、温度系数与温度附加的误差。

◆ 迟滞是指传感器输出的数据由小变大,再由大变小两种静态特性不一致的程度。

◆ 重复性是指系统输入量按同一方向作全量程,连续多次变动时,静态特性不一致的程度。

◆ 线性度是表示系统静态特性对选定拟合直线的接近程度。拟合直线的方法其中以理论线性度与最小二乘法线性度应用最普遍。

◆ 精度:系统总精度是指在其量程范围内,基本误差与满度值的百分比数表示。基本误差由系统误差、随机误差两部分组成。迟滞和线性误差为系统误差,重复性所表示的误差为随机误差。系统的总精度 = (系统误差的绝对值 + 随机误差的绝对值) / 满度值。

任何传感器设计都会努力地增加线性度,减少迟滞,提高重复性,以得到较高的精度。但是这里并没有考虑温度对传感器的影响。对于不是用半导体材料制作的传感器和用半导体材料制作的传感器的探头,温度对其影响可相差十倍。

◆ 温度系数与温度附加的误差。

零位温度系数是指温度每改变 1 度,零位值改变量与满度值之百分比。灵敏度温度系数及

其温度附加值,它是指由于灵敏度随温度漂移,温度在数值上每改变1度,灵敏度相对变化量的百分数。

2.2.2 传感器动态特性

大量被测量物理量是随时间变化的动态信号。系统的动态特性反映测量动态信号的能力。

智能传感器中的传感器本身有与频率相关的动态特性。其后面的调理器也是在一个频率范围内是稳定的,传感器与CPU、A/D的接口、分布电容、电磁环境也会影响整个传感器系统的动态特性。另外传感器本身一般都有一个固有频率。传感器在使用时要避开固有频率(也就是可产生共振的频率)。在使用传感器前必须清楚我们要测量系统的频率特性和传感器的动态特性曲线以及放大器的动态特性。只有全面考虑这三种情况才能设计和使用传感器系统。

2.2.3 设计和使用一个传感器系统应该注意的问题

首先要选择一个结构、尺寸、参数与工艺合适的传感器。例如,测压力的应变片尺寸中等,灵敏度较小,输出阻抗一般较低,但温度漂移比较小。而用半导体制成的测压力传感器,体积小,灵敏度高,而温度漂移大。又如,选用热电偶测温度,热电偶有几种金属组成。不同的热电偶传感器,测量温度的范围是不一样的,各段温度的线性度也是不一样的。但热电偶传感器一般输出阻抗低,精度高。输出信号一般在几毫伏或几十毫伏范围内。而半导体测量温度,传感器测输出信号一般在0.1V~1V之间,灵敏度比较高,但温度漂移比较大。

利用差动对称结构或电路来消除温度、电源、电流对传感器精度的影响。

对称结构在一定程度上也可以提高传感器的线性度。利用电桥测量时,应当注意电源电压或恒流源对电桥测量的影响。后级放大器一般用共模系数比较高,高输入阻抗和低输出阻抗的运算放大器。一般,第一级放大器的放大倍数不会超过十倍。

在电桥设计时,要尽可能消除电桥中除传感器以外的电阻、电容、离散度,确保电阻、电容的精度,一般用不低于1%精度的元件。最好选用温度漂移较小的金属膜精密电阻。设计时最好使用电位器可以方便调零校准电路。使整个系统有可调整的地方。这样会给整体调试带来方便。

对传感器系统如何进行补偿。对于产生系统误差的迟滞,一般可以用找出拟合曲线的方法来消除。对于产生线性度不好的传感器,应首先考虑用户使用的一段是不是该传感器线性度较好的一段。如果已经是较好的一段,但线性度还不够好,就需要采用二种方法校准。其一是,知道传感器的特性曲线的方程,通过拟合找出线性度好的直线。其二是,可以把以前已测出的一系列值或者该热电偶列表中的数据以文件形式存入内存或外存中,测到的数据通过查表找到相应的温度值。例如,一种热电偶从零下200度至零上1300的列表中,每隔0.1度就有一个该热电偶的毫伏(mV)值。例如,125.1度对应3.450mV。要存储这种热电偶的温度与毫伏表的数据,需要 $(1300+200) \times 10 \times 3 = 4.5$ 万字节,所以需要有一个40K字节的数据文件。如果进一步提高线性度,应该用线性度较好的传感器来替换。如果进一步提高精度,可以用插值法求更精确的温度值。

采用先进测量方法来提高整个系统的精度。在传感器系统中采用信号测量法就是一种很

有效的方法。二信号法是让被测量信号与参考基准信号,经过相同的路径由相同的系统进行测量。

测量被测量信号时输出为 $U_1 = (U_x + E_0) \times A$,

测量参考信号时输出 $U_2 = (U_r + E_0) \times A$,其中 A 为增益, E_0 为系统误差。

最后求出 $U_x = \frac{U_1}{U_2} U_r + E_0 \times \left(\frac{U_1}{U_2} - 1 \right)$ 。

实际上,我们用比值计算 U_x 公式是: $U_x = \frac{U_1}{U_2} \times U_r$ 。

故通过比值法求 U_x 绝对误差是: $U_x = E_0 \times \left(\frac{U_1}{U_2} - 1 \right)$,

相对误差是: $\frac{\Delta U_x}{U_x} = \frac{E_0}{U_x} \times \left(\frac{U_1}{U_2} - 1 \right)$,

而不是用二信号法求 U_x 的相对误差: $\frac{\Delta U_1}{U_1} = \frac{A E_0}{U_1} = \frac{E_0}{U_x}$ 。

所以用二信号法是不用二信号法的相对误差的 $\left(\frac{U_1}{U_2} - 1 \right)$ 倍。采用的参考信号 U_r 越接近被测的 U_x 时,输出 U_1 也越接近 U_2 ,相对误差越小。而且消除了系统增益变化产生的误差。二信号法要想使得到的 U_x 范围较大,而精度较高是很困难的。所以又引入了三信号法以及三步法测量方法。

例如,目前有三个信号要测。一个信号是标准零信号(接地),一个是标定信号,一个是真正要测量的 U_x 信号。测试系统先测零,再标定,最后测 U_x 。测完后,经过相减相比,简单计算: $U_x = \frac{U_3 - U_1}{U_2 - U_1} \times U_r$ 。

其中 U_1 是测零信号的值, U_2 是测标定信号 U_r 后得出的值。 U_3 是 U_x 的输出值。

这种方法要求系统增益在三步测量时无变化,因此时间变化并没有引入到系统的误差中。 E_0 (系统误差)在测定时间内不变,系统误差完全可以消除。

使用三步测量法时,为了确保精度,必须能在短时间内保持增益不变,电源稳定,接地良好,标定高应是按与被测目标参量同类属性的标准来提供(如标准恒温器等)。三步法因为总的测量时间很短,一般在几十毫秒完成,所以在这样短的时间内系统的温度漂移是可以不计的。因此环境温度产生的误差也可以消除。

三步法测量也会带来新的误差。用电子开关对三种输入进行转换时,电子开关的一致性,以及它们所引起的误差也要给予注意。

2.3 智能传感器功能实现方法

“智能”目前还没有准确的定义。在国内外杂志上常出现:智能控制、智能调节、智能传感器、智能机器人、智能通信等等。智能至少应有两种含义:其一是采用“人工智能”的理论、方法和技术。这包括专家系统、知识工程、模式识别等方面的理论和方法。其二是具有“拟人智能”的特性或功能。例如,自适应、自校正、自诊断等。人们讲智能传感器是从狭义上讲的,只是有一定的拟人功能。有 CPU 才有可能做到智能。但只有电脑、CPU 还是不行的。必须有相应的软件实现某种功能。

智能传感器功能主要指非线性自校正技术、自校零、自校准技术、自检测、自诊断、自选量程、增益控制、数字信号处理、图形数字信号预处理等。

实现传感器各项智能功能和建立智能传感器系统,是克服传感器本身不足,获得高稳定性、高可靠性、高精度、高分辨率与高自适应能力的必由之路。采用的技术可以是硬件技术也可以是软件技术。

测量系统的线性度是影响系统精度的重要因素之一。智能传感器具有非线性自动校正功能,这样可以消除系统的非线性误差,提高精度。但这要有一个前提,就是系统的输入/输出特性要具有重复性。

2.3.1 校正非线性方法

查表法是一种分段线性插值法。它是根据精度要求对非线性曲线进行分段,用若干段折线逼近曲线。将折点坐标存入数据表中,测量首先要明确对应输入被测量电压值在哪一段,然后根据那一段的斜率进行线性插值,即输出值。当然分的段数越多精度越高,误差越小。它的最终表达式为:

$$Y = X_K + \frac{X_{K+1} - X_K}{U_{K+1} - U_K} \times (U_i - U_K),$$

其中 K 为段数, U_i, U_K 为输出电压值, X_{K+1}, X_K 为被测量。 Y 为经过非线性校正后的输出值。

用查表法进行分段插值会有一定的误差,这误差大小主要决定于分段的多少,以及各段直线拟合曲线所产生的误差。

另外一种拟合方法是曲线拟合法。

假设仅非线性传感器输入输出拟合方程为:

$$X(u_i) = a_0 + a_1 u_i + a_2 u_i^2 + a_3 u_i^3 + \dots + a_n u_i^n,$$

如果确定 $n=3$ 已达到拟合误差:

$$X(U_i) = a_0 + a_1 u_i + a_2 u_i^2 + a_3 u_i^3.$$

通过输入标定点 X_1, X_2, X_3 得到定标输出值 U_1, U_2, U_3 等值。从而通过矩阵方程求出来系数值 $a_0 \sim a_3$ 。即得到拟合曲线的多项式方程。得到方程式:

$$X(u_i) = a_3 u^3 + a_2 u^2 + a_1 u + a_0 = [(a_3 u + a_2) \times u + a_1] \times u + a_0.$$

每次把采样值 U 代入上式循环运算三次,再加上 a_0 就求出。

这种方法求解在噪声不太大是有效的。因为噪声可能使 $a_0 \sim a_3$ 求解受阻。

2.3.2 自校零与自校技术

传感器测量系统中常常因温度变化使放大增益发生改变。在输入信号为 0 时,输出并不是一个固定不变值,即产生零漂移。当然,环境变化也会使系统的灵敏度产生漂移。

在传统传感器系统中,使用各种方法尽量减少产生温度漂移的可能性。选用温度漂移小的元器件。这样相对来说代价是比较高的。

智能传感器系统,它另辟途径,采用自动校正的零位漂移,灵敏度漂移。对于线性较好的传

传感器只要经过三步法就可以自动校正零位漂移和灵敏度漂移。

三步法即：

第一步校零。输入信号为零时，输出值 $Y_0 = A_0$ 。

第二步标定。输入一个标准值 U_r ，这时输出 Y_r 。

第三步测量。这时

$$U_x = \frac{Y_x - Y_0}{Y_r - Y_0} \times U_r = \frac{Y_x - A_0}{Y_r - A_0} \times U_r,$$

即输出值 Y_x 与输入值 U_x 关系式为： $Y_x = \frac{Y_r - A_0}{U_r} \times U_x + A_0$ 。

从而达到校准的目的。

如果传感器输入输出关系是非线性的，那么只有输入信号为 0 和一个标定点的标定是不够的。一般标定点至少三个，再用最小二乘法的曲线拟合方法，分别求出各段拟合曲线。再分段建立输入与输出信号的关系。从而达到自动校正的目的。

自动补偿主要指传感器的温度补偿和传感器频率特性补偿。使受温度和频带影响而产生的误差得到纠正。

2.3.3 自选量程与增益控制

智能传感器系统的增益设置问题，是要在系统自身数据容量与被测量的范围，系统的精度与信噪比，系统的灵敏度与要求的分辨率等诸多因素之间折衷选择来确定的。若增益太低，数据的信息容量就会浪费，信噪比可能很低，测量误差大从而不能满足要求；反之，若增益过大信息也会因为系统内的数据容量不够而损失掉。通过选择，分析系统误差而用一个合适的增益。这里没有一个统一的定理，只有若干考虑因素。如传感器灵敏度，信噪比，A/D 的输入范围，输出精度，在传感器与 A/D 之间选择合适的放大器。当然也要考虑放大器输入阻抗，输出阻抗对前后的影响。

2.3.4 智能传感器的自检验

自检是传感器自我检测过程。通过自检测，给予相应的提示，从而提高系统的可靠性。

一般开机上电先自检测，也可以进行周期自检或键入自检。自检内容主要是传感器的 ROM 和 RAM，CPU 的指令系以及 A/D，D/A，I/O。这时自检只能是部分自检，不可能把所有可能性都自检到。例如，一般 CPU 指令有 100 条左右，如果把每一条指令全都要检查一次还有可能，但是 100 多条指令排列组成的各种可能性是无法自检的，因这几乎是天文数字。一般只对智能传感器特定几种情况进行自检，并认为自检通过。例如，RAM 的自校全写入 0，再读出，看是否全 0。全写入 FFH，读出看是否全是 FFH。一般根据 ROM 的数据求和的值是否有改变来判断 ROM 的好坏。

本小节简单地叙述了一下智能传感器要达到的一些功能。例如自检，自校，自动补偿，补偿一些非线性等。如果对某一个功能特别有兴趣，有特殊用处，还要进一步查找有关算法，再进一步分析比较。最后通过硬软件实现。