

# 基于 UML 建模的通用水下探测系统开发

李 博,周穗华

(海军工程大学 兵器工程系,湖北 武汉 430033)

**摘 要:**传统水下探测系统的集中控制结构影响了系统的扩展和兼容能力,采取分布式结构的目标探测系统设计将有效解决系统存在的诸多问题,但传统的系统设计方法已无法适用于新结构的系统开发。在详细分析水下探测系统的功能需求的基础上,利用 UML 统一建模语言实现了对于通用水下探测系统的分析建模,着重分析了系统的静态结构和动态行为,开发了系统的分层式软件结构。相比以往系统设计方法,本方法提高了通用水下探测系统的软件设计效率以及系统的维护性和复用性。

**关键词:**统一建模语言;水下探测系统;分析建模

**中图分类号:**TP311

**文献标识码:**A

**文章编号:**1673-629X(2011)04-0238-04

## Multi-purpose Underwater Detection System Design Based on UML

LI Bo, ZHOU Sui-hua

(Dept. of Weaponry Engineering, Naval Univ. of Engineering, Wuhan 430033, China)

**Abstract:** Centralized control structure of underwater detection system (UDS) restricted expansion and compatibility. With the analysis of design detects existing in UDS, used the UML to establish an analysis model on the basis of the analysis of definition and function requirement of multi-purpose UDS, and analysed the static structure and dynamic behavior of the system. At last, introduced a layered software architecture for system design. Compared with previous design, the maintenance, reusability of software products would be improved based on using UML to analyze and design the system.

**Key words:** unified modeling language; underwater detection system; analysis model

### 0 引 言

水下探测系统在海洋环境探测、水下目标识别等方面具有广阔的应用前景。当前广泛应用的探测系统大量采用嵌入式设备作为系统的控制部件,信号处理能力得到显著提升。但水下探测系统目前采用集中式控制方式却限制了系统的扩展和兼容能力的提升,影响了系统的通用化和智能化的发展。若将当前工业系统设计中普遍采用的分布式控制系统结构应用于水下探测系统设计,并结合传统系统设计的模块化结构,将更有利于系统的调试和装配,并且采用通用接口和通信协议进行模块化的通用水下探测系统设计,将能促进系统的扩展性、可靠性和鲁棒性的提升<sup>[1]</sup>。

随着水下探测系统的性能要求的不断提升,系统的复杂程度越来越高、系统的软硬件协同设计难度也越来越大。传统的水下探测系统采用的是基于自上而

下的设计模式,针对具体的设计需求,逐步完成软件设计,软件的专用性强,可移植性差。而面向对象技术内在地支持了对系统的抽象、分层和复用技术,这将有助于水下探测系统分的开发。UML (Unified Modeling Language) 是面向对象的可视化建模语言,是面向对象领域中的重要成果,UML 建模提供了新的软件模型描述方法,其统一过程的迭代式生命周期适应于需求的不断变化,为降低软件开发的风险,提高重用性提供了保证<sup>[2,3]</sup>。目前 UML 已经广泛的应用在软件开发中,并已成为可视化描述软件结构和行为的业内标准语言。相对于传统的软件开发方法,UML 与统一过程更适合于构建水下探测系统的软件模型。文中将软件工程应用到水下探测系统软件开发之中,通过 UML 分析建模,设计了分层式结构的通用水下探测系统系统软件框架,基于模块的软件设计提升了系统的可维护性和软件模块复用性。

### 1 水下探测系统功能分析

根据水下探测系统的特点,可以定义其基本功能

收稿日期:2010-09-14;修回日期:2010-12-23

基金项目:十一五预研课题(4010709010301)

作者简介:李 博(1979-),男,博士生,研究方向为水下探测系统集成、控制与仿真;周穗华,教授,研究方向为武器系统与运用工程。

如下:

(1) 水下综合测量识别功能——水下探测系统能够实现对多种水下物理量的同时测量,对所探测物体进行区分并全面的放映出物体的运动规律。

(2) 自检、自校和自控制功能——水下探测系统长期处于水下工作状态,系统根据事先设定的程序实现自我控制功能。系统在接通电源时进行自检,诊断测试以确定模块组件有无故障,系统工作过程中,针对外界物理量变换,可实现对于内部存储的特征数据量的在线校正。

(3) 数据存储和传输功能——系统可实现探测数据的存储,并可通过遥控收发装置实现数据传输以及接收控制指令以实现相应的功能如系统休眠、系统工作、测量数据的输出等。

(4) 系统扩展功能——系统能够针对不同的探测需求实现对于水下探测系统的基本功能的扩展或性能的提升,如远程探测、特征识别、自身姿态检测等。

根据系统的功能分析,可确定系统的行为者和相关用例,图1用例模型描述了用例与行为者以及用例与用例之间的关联关系。系统的行为者是指与系统交互的外部实体<sup>[4,5]</sup>。对于水下探测系统,从系统需求可以看出,系统的行为者包括操作员、系统自检预置装置、保险器、遥控收发装置和系统的外部执行器。系统的上电复位组网用例包括了模块自检、参数配置两个子用例。

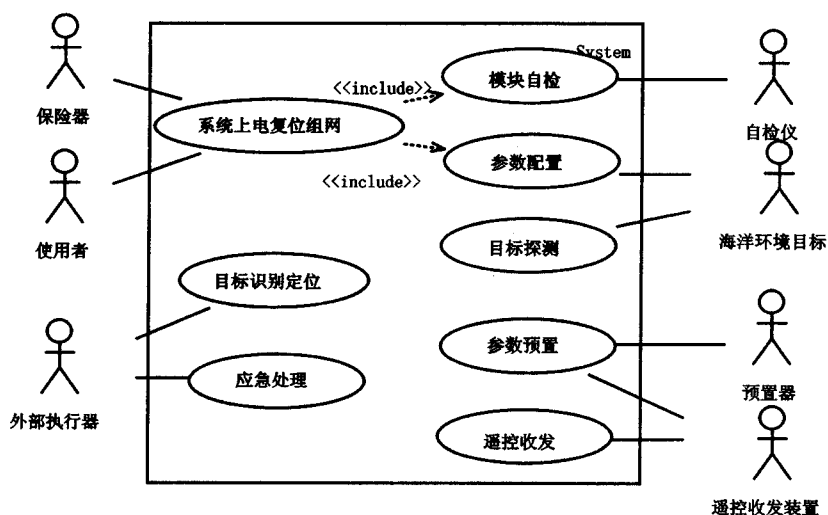


图1 水下探测系统用例模型图

## 2 系统分析建模

### 2.1 静态结构

由用例模型驱动建立对象与类模型,通过系统的

静态模型将系统中的类模型化来处理问题的静态结构,由于水下探测系统内部信息密集,因此对于静态建模就显得特别重要。在目标探测系统中,真实世界的类主要是物理 I/O 设备,即各种传感器和指示按键,系统参与者通过它与系统进行交互,在确定这些类后,将这些 I/O 设备看作是系统静态模型中的外部类,以类图的形式来描述。

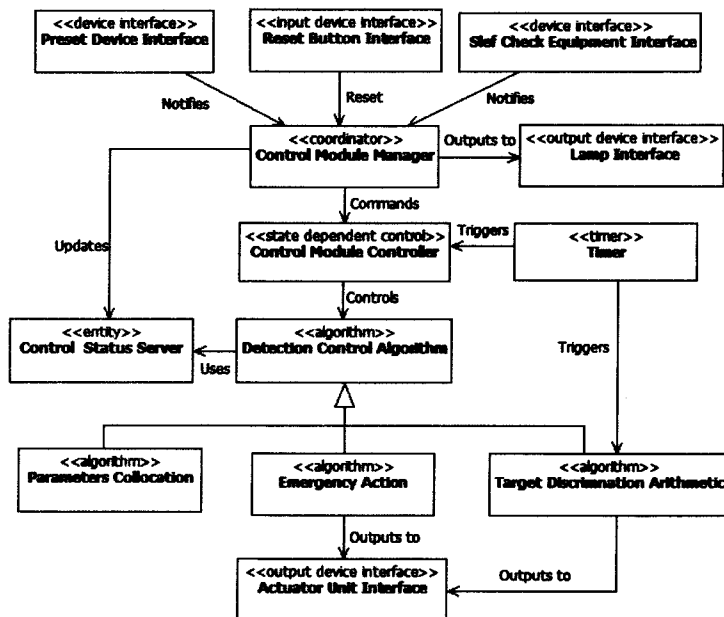


图2 控制模块子系统静态模型

分析过程是高度循环渐进的。由于用例图中的每个用例是通过各种对象的一系列活动的完成来实现系统的功能需求,因此,一旦用例确定,设计者就可确定每个用例的参与对象,并从对象之间的交互中发现对象间的关系。两个对象间若有交互,则必有关联,UML 的协作图通过对象之间发送或接收信息实现对于对象之间动态交互关系的描述,分析对象的行为,由此可标识对象属性与行为。文中重点讨论类的抽取与类间关联,因此不展开分析协作图。

从协作图出发开发解域的类图,协作图上的对象映射能实例化出它们的类。协作图上对象之间的连接,在类图上都有对应的关系。图2、图3分别为系统中的控制模块子系统和传感器模块子系统的静态模型。类之间的关系大部分是关联关系,在共享的某些公共特性的类之间,才存在泛化关系。在控制模块子系统中,Detection Control Algorithm 类是其他三个算法类:Parameters Collection 类、Emergency Action 类和

Target Discrimination Arithmetic 类的超类。在传感器模块子系统中, Parameters Arithmetic 类是算法类; Self Check 类、Zero Point Caculation 类和 Target Detection Arithmetic 类的超类。

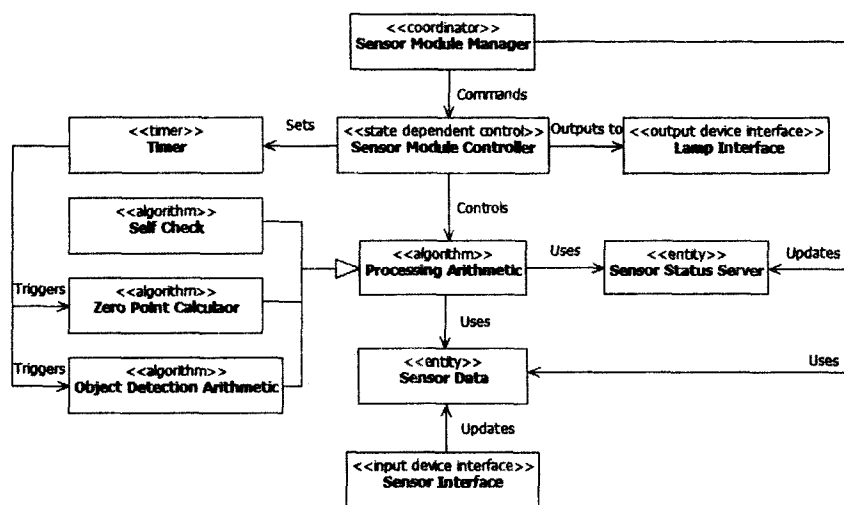


图3 传感器模块子系统静态模型

## 2.2 动态行为

类图展现了系统的静态特征,但系统不是静止的,对象在生存期中会通过状态的改变实现对于事件和时间的响应<sup>[6,7]</sup>。在每个模块中都有一个状态控制对象 Module Controller,其实现对于外部设备(外部执行器、自检仪、显示灯等)控制、显示,以及模块内部状态的控制,图4给出了 Module Controller 对象的状态图。

Module Controller 对象包含空闲状态、系统运行状态、指示执行器动作状态。其中系统运行状态是一个超状态,包含三个子状态:

(1) 初始化状态。当系统上电后,系统进入初始化状态,在此过程中,系统中各节点模块进行模块自检,并通过指示灯显示自检结果。

(2) 组网状态。当系统自检完成,或者系统运行过程中,节点出现故障,系统进入到此状态,在组网状态中,系统各模块间通过信息交互,确定系统构成,并通过指示灯显示组网状态。

(3) 系统探测识别状态。此状态本身也是一个超状态,包含以下几个子状态:

- 参数配置状态。当系统完成系统组网,或者系

统运行过程中需要重新进行零点检测,系统进入此状态。

- 目标探测状态。
- 目标识别状态。

## 2.3 子系统间消息通信

系统设计充分的考虑到了局部自治、与物理数据源尽量接近的分布式构件配置准则,系统中每个模块子系统都是自治构件,在提供相关服务的特定节点上运行。在系统运行的绝大多数时间,这些构件模块都可以独立运行,即使其中的部分构件不可用,它们也能独立运行,因为每个模块构件都可成为控制构件,具有一定的可预测性能。

由于系统的运行需要各个分布式模块之间通过消息通信协同实现,因此需要考

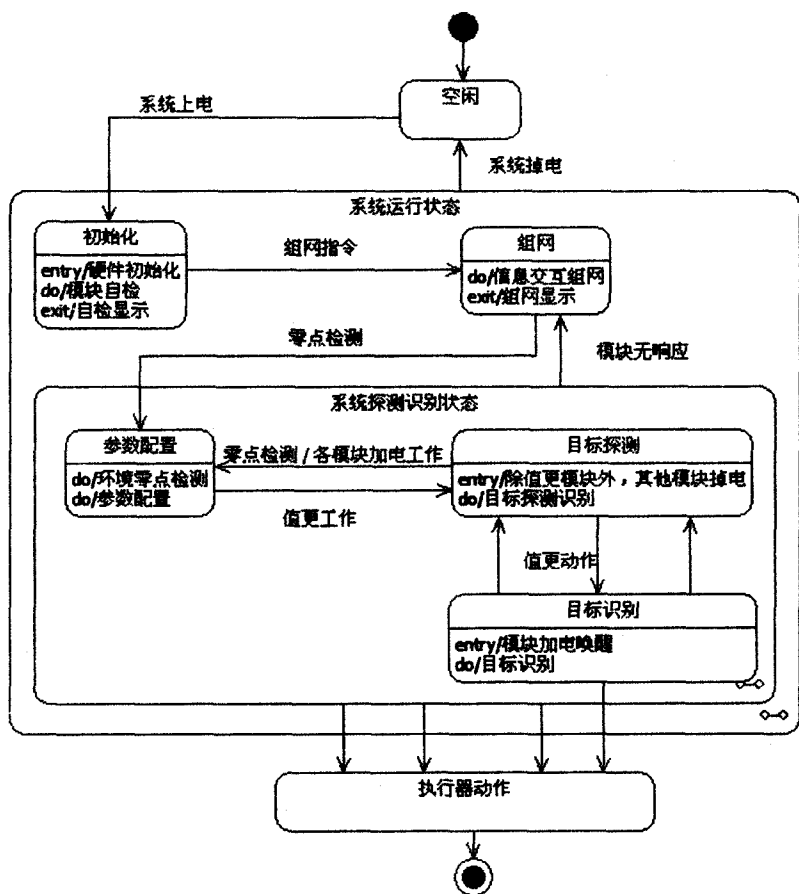


图4 Module Controller 对象的状态图

虑各个模块构件子系统之间接口的消息通信,如图5为控制模块与传感器模块之间的任务接口消息通信。

由于控制模块与传感器模块的处理任务的优先级不同,因此当控制模块向传感器模块发送控制指令之

后,不可能一直挂起等待应答,因此在 Control Module Controller 任务与 Sensor Module Manager 任务之间使用的是松耦合的消息通信,由于消息的生产者和消费者的处理速度不同,可在他们之间建立一个 FIFO 消息队列,进行信息缓冲。

当传感器模块针对相关控制指令,实现系统的自检、参数配置、目标探测等功能,传输控制模块所需的数据和相关状态信息,这种情况下, Sensor Module Controller 任务与 Control Module Manager 任务之间同样使用的是松耦合的消息通信。传感器模块在多数情况下,可能出于值更自主运行状态下,目标识别状态下,模块将以紧急信息通知控制模块,并唤醒系统其他传感器模块,这种情况下, Sensor Module Controller 任务与 Control Module Manager 任务是有应答的紧耦合,因为此时传感器模块必须确认控制模块接收到紧急信息,能够继续后续的目标识别定位等任务处理。

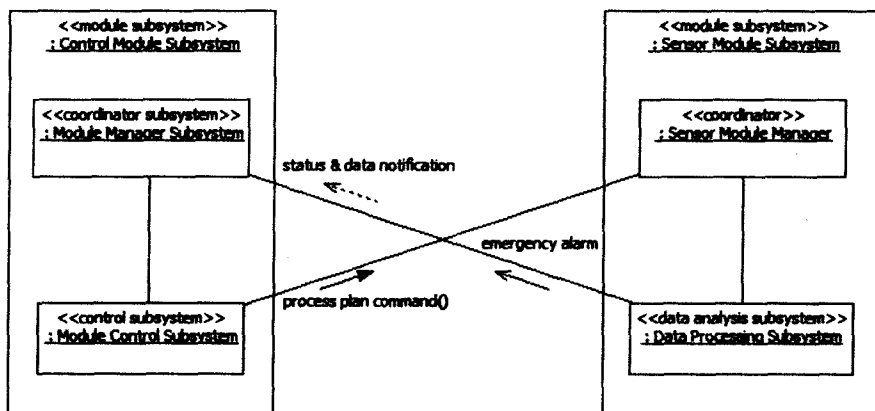


图5 子系统间任务接口

### 3 分层式软件体系结构

水下探测系统的分布式结构设计趋向于现有的组件和通用的结构,系统可根据具体应用不同进行适当的裁剪、修改或进行附加的开发。因此系统设计应借鉴开放式系统体系结构的设计理念<sup>[8-12]</sup>,即按照开放系统接口标准,通过建立系统所需的接口,使硬件和软件模块的替换成为可能,从而保证了未来的系统仅需对现有系统进行最少的再设计就能进行升级和维护。针对水下探测系统不同子系统的静态建模和系统协作图,设计目标探测系统的分层式软件结构,如图6所示。

该结构模型采用分层概念,整个软件结构分为四层:系统功能层、系统控制层、实时通信软件和硬件模块抽象层,其中系统功能层和系统控制层可统称为系统管理软件,位于软件结构的应用层。层与层之间通过标准的接口进行信息交互,这样保证了每层的独立性。

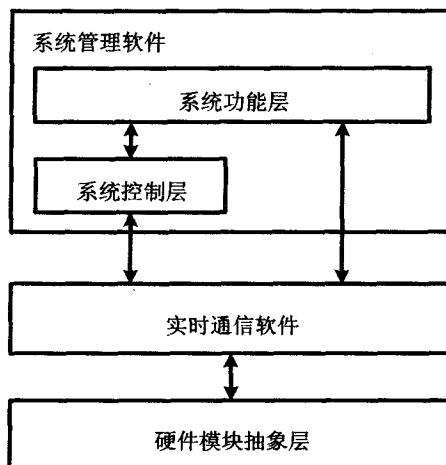


图6 水下探测系统分层式软件结构

系统功能层提供系统的通用管理和控制功能,负责内部功能模块的调度和管理,实现水下探测系统所需具备的探测、识别等基本功能以及相应的附加功能。

系统功能层对于水下探测系统的控制属于通用逻辑层面的控制,不涉及具体的型号特征,对于水下探测系统的具体控制则是通过调用系统控制层来实现的。系统功能层通过系统控制接口与系统控制层进行信息交互。

实时通信软件负责管理节点模块之间的通信,使上层应用可以方便地使用系统提

供的各种服务。其通常包括实时通信协议、分布式执行控制子层和服务接口。硬件模块抽象层是硬件调用层,与硬件具有密切关系,在其接口定义中包含了接口数据传输特性、接口配置属性、接口定义函数等系统或硬件所需的硬件支持功能,实时通信软件通过硬件模块抽象层调用硬件资源。由于硬件抽象子层屏蔽了系统硬件特征,因此可以适应嵌入式硬件多样性,软件结构中的高层内容和具体的硬件是相互独立的,硬件的改变不会影响应用层软件,上层软件的可移植性大大提高,系统的可扩展性得以提升。

### 4 结束语

水下探测系统在海洋环境探测、水下目标识别领域应用广泛。文中在给出水下探测系统的功能需求的基础上,采用UML方法对水下探测系统进行建模,进行了系统的静态建模和动态行为分析,针对水下探测系统的通用性要求,设计了分层式的系统软件结构,在水下探测系统的原理样机进行了试验,运行情况良好。

(下转第245页)

处。单步运行的具体实现方法如下:

#### 1) 单步进入。

单步运行主要通过单步中断来实现。监控程序接收到上位机的单步调试命令,保存监控程序的运行环境,还原用户程序的运行环境,打开 CPU 状态寄存器的 T 标志(即:置 T 标志为 1),以实现单步运行后进入单步中断服务程序。还原用户保护模式下运行环境,通过 IRET 指令跳转到用户程序上次到执行的位置,继续执行。CPU 在执行一条用户指令后,进入保护模式下的 T 中断服务处理程序。T 中断服务处理程序和断点中断处理程序类似,需要完成用户保护模式环境与监控程序实模式环境的切换,以及用户运行环境和调试信息的储存。

#### 2) 单步跳跃。

单步跳跃的实现方法与运行到断点类似,需要使用断点中断和单步中断共同完成。因为单步跳跃需要执行完整个函数或过程,所以上位机需要将跳跃地址(函数或过程体的下一条指令的起始地址)传输给下位机,下位机使用保护模式下的单步中断将该地址处的用户代码替换为 CC,还原用户程序的运行环境并继续执行。当用户程序执行到断点处时执行断点中断服务程序,切换到实模式下的监控程序,便实现了单步跳跃功能。

## 4 结束语

远程调试器是嵌入式开发的一个重要工具,文中给出了一个在实模式远程调试器的基础上实现保护模

式下调试功能的方案,使远程调试器能够实现实模式和保护模式两种环境下的目标程序的调试功能。

#### 参考文献:

- [1] 李红卫. 嵌入式远程调试工具的研究与实现[J]. 微计算机信息, 2009, 25(1-2): 87-89.
- [2] 乔 容, 彭思鹏, 柏桂枝. 嵌入式软件源码级交叉调试器的设计与实现[J]. 舰船电子工程, 2004, 24(3): 56-60, 101.
- [3] Azzerini B, Lopriore L. Program debugging environments: design and utilization[M]. New York: Ellis Horwood, 1992.
- [4] 章 辉, 桥 井, 高 桥, 等. 基于仿真器的源码级调试器设计与实现[J]. 计算机工程与设计, 2010, 31(8): 1685 - 1688.
- [5] 包 磊, 姚放吾. 基于远程设备的汇编源码调试器的设计与实现[J]. 计算机技术与发展, 2009, 19(11): 155-158.
- [6] 张 欣, 栾 新. 基于 ARM 芯片的嵌入式交叉调试系统[J]. 微计算机信息, 2007, 6-2: 128-130.
- [7] 王亚磊, 姚放吾, 罗 威, 等. 基于 386EX 的嵌入式软件调试器的设计与实现[J]. 计算机技术与发展, 2007, 17(s): 264-266.
- [8] 吴志雄. 一种易于扩展的交叉调试器设计及其实现[J]. 电子技术, 2007, 36(11): 26-29.
- [9] 杨季文. 80x86 汇编语言程序设计教程[M]. 北京: 清华大学出版社, 1998.
- [10] Rosenberg J. How Debuggers Work[M]. New York: Wiley Computer Publishing, 1996.
- [11] 曲玉昭, 陆华奇, 于忠清. 基于事件和周期任务的嵌入式系统设计[J]. 计算机技术与发展, 2006, 16(10): 175 - 177.

(上接第 241 页)

在水下探测系统软件开发的各阶段,使用 UML 对系统进行建模,可以通过建立不同的模型对系统进行描述,取得良好效果。这种标准化、独立性的模块化软件设计方法能够有效地促进水下探测系统的规范化、通用化设计,使得软件模块重用成为可能。

#### 参考文献:

- [1] 李 博, 周穗华. 水下探测系统内部总线设计[J]. 仪器仪表学报, 2008, 29(8): 30-34.
- [2] 吴 际, 金茂忠. UML 面向对象分析[M]. 北京: 北京航空航天大学出版社, 2002: 2-6.
- [3] Booch G, Rumbaugh J, Jacobson I. The Unified Modeling Language User Guide[M]. [s. l.]: Addison Wesley Longman, 1999: Appendix B.
- [4] 冀振燕. UML 系统分析设计与应用案例[M]. 北京: 人民邮电出版社, 2003: 2-10.
- [5] 王小平, 宣乐飞, 张 蔚. 基于 UML 的嵌入式实时控制系统的建模与实现[J]. 计算机技术与发展, 2006, 16(7): 239 - 244.
- [6] 王建新, 姚放吾. 基于 UML 的软硬件协同设计方法[J]. 计算机技术与发展, 2006, 16(1): 96-98.
- [7] Goma H. Designing Concurrent, Distributed, and Real-Time Applications with UML[M]. [s. l.]: Pearson Education, Inc. 2000.
- [8] 李宏伟. 开放式数控系统分布式体系结构及其实现策略研究[D]. 天津: 天津大学机械工程学院, 2005.
- [9] 史先传. 基于 CAN 总线的开放式数控系统硬件设计与实现[D]. 南京: 南京航空航天大学, 2001.
- [10] 陈宁宁. 开放式网络控制系统的研究与应用[D]. 杭州: 浙江大学电气工程学院, 2006.
- [11] 杨家龙. 基于网络环境的舰船机舱动力装置监控系统技术研究[D]. 哈尔滨: 哈尔滨工业大学, 2006.
- [12] 谢林柏. 网络化控制系统中若干问题的研究[D]. 武汉: 华中科技大学, 2004.