

厦门大学博硕士学位论文摘要库

学校编码: 10384

分类号 \_\_\_\_\_ 密级 \_\_\_\_\_

学号: X2006230084

UDC \_\_\_\_\_

厦 门 大 学

硕 士 学 位 论 文

CAN 总线在工业用嵌入式控制系统中的

应用研究

Research on Application of CAN-BUS in Industrial  
Embedded Control System

余新良

指导教师姓名: 林 坤 辉 教授

专 业 名 称: 软 件 工 程

论文提交日期: 2008 年 11 月

论文答辩日期: 2008 年 11 月

学位授予日期: 2008 年 月

答辩委员会主席: \_\_\_\_\_

评 阅 人: \_\_\_\_\_

2008 年 11 月

## 厦门大学学位论文原创性声明

兹呈交的学位论文，是本人在导师指导下独立完成的研究成果。本人在论文写作中参考的其他个人或集体的研究成果，均在文中以明确方式标明。本人依法享有和承担由此论文产生的权利和责任。

声明人（签名）：  
年 月 日



## 摘要

CAN 总线采用了 ISO/OSI 的 7 层参考模型中的 3 层：物理层、数据链路层和应用层。它是一个多主的串行通讯协议，能有效地支持具有很高安全等级的分布式实时控制。但是，协议在数据链路层只实现简单的冲突仲裁和重发机制，当 CAN 总线网络负载突发性增长时，存在通讯冲突问题，发送失败的报文在其尝试重发的过程中会严重影响到网络的运行稳定，造成其它报文的延迟。

在工业分布式嵌入式控制系统中，系统工艺要求达到的控制目标是由系统中各个 CAN 节点协调工作而实现的。CAN 总线连接各节点，是控制信息和状态信息传递的通道。为了保证控制的实时性，本论文设计实现在应用层以时间片的方式来调度周期性报文的 CAN 总线应用协议，协调分布式系统中各个 CAN 节点对总线物理介质的占用。这样可以保证在周期时间范围内通讯的确定性，而时间的确定性正是实时控制的基本要求。

系统设计实现包括硬件和软件两个方面的工作。硬件平台采用 PXA270 和 LPC2194 的双处理器结构。PXA270 处理器负责控制逻辑的运行，产生和处理 CAN 总线传递的数据信息。ARM7 处理器 LPC2194 主要负责 CAN 总线应用协议的实现，以及检测和处理系统紧急事件。软件平台采用 Linux 操作系统和实时性性能好的  $\mu$ C/OS-II 操作系统。PXA270 处理器运行 Linux， $\mu$ C/OS-II 运行在处理器 LPC2194 中。需要设计专用 Linux 设备驱动程序和  $\mu$ C/OS-II 任务程序，保证两个处理器间可靠地互通信息，使控制参数和工况参数在 PXA270、LPC2194、CAN 总线、各分布式 CAN 节点间顺利交互，达到系统工艺要求的控制目标。

系统实现了四条独立 CAN 总线，每条总线最多可挂装 110 个 CAN 节点。各 CAN 节点协同工作，执行对温度、压力、湿度等模拟量的采集以及电磁阀、电机等开关量的控制和检测。系统的控制逻辑要求时间精度精确到 0.1 秒。所以系统信息量大，控制实时性要求高。

通过大量的测试工作，确认系统达到了两个主要的设计目标。第一个目标是基于 CAN 总线的系统控制数据流能在整个分布式控制系统中通畅传递；第二个目标是对于影响系统安全运行的紧急事件，在 20ms 内能够检测到并且作适当处理。

**关键词：**CAN 总线；Linux；嵌入式系统

## Abstract

CAN bus is subdivided into three different layers according to the ISO/OSI seven layers Reference Model: the Physical Layer, the Data Link Layer and the Application Layer. It is a multimaster serial communications protocol which efficiently supports distributed realtime control with a very high level of security. But the arbitration and retransmission mechanism in the Data Link Layer is simple, when the load burst, the bus access conflict is resulted, so many retransmissions of corrupted messages will seriously affect the performance of CAN bus network, resulting message transmission delay.

All CAN nodes linked by CAN bus cooperate with each other to fulfill control logic in the industrial distributed embedded control system. CAN bus is the channel of transmission of control and state messages. To ensure realtime performance, a method that schedule periodical messages by timeslice in the Application Layer is designed. Usage of physical medium by CAN nodes is arranged, so the message transmission delay is confined, which is the basic requirement of realtime control.

Steps on software and hardware should be taken to make certain for these requirements. The core of hardware platform is processor PXA270 and processor LPC2194, it is a two-processor-architecture. PXA270 runs the system control logic, generate and dispose the data message transmitted by CAN bus. LPC2194 is dedicated to communication task, implement CAN Application Protocol, detect and deal with system emergencies. The soft platform adopt Linux operating system and the realtime kernel  $\mu\text{C}/\text{OS}-\text{II}$ . PXA270 runs Linux, LPC2194 runs  $\mu\text{C}/\text{OS}-\text{II}$ .

Customized Linux device driver and  $\mu\text{C}/\text{OS}-\text{II}$  task should be designed to allow control and state message flow freely between PXA270, LPC2194, CAN bus and CAN nodes, fulfill system control logic.

Every system has four independent CAN buses, up to 110 CAN nodes can be hung on one CAN bus. The control point can be analog signals such as temperature, pressure and humidity or switch signals for electromagnetic valves and motors. The time precision requirement of system control logic is 0.1 second. So the amount of message data are great, and the realtime performance is critical.

Two targets of system design are confirmed through a great deal test work. The first target is to make system control data flow freely between every part of the

distributed control system based on CAN bus. The second is to make certain that system emergencies can be detected and disposed properly in twenty milliseconds.

**Key Words:** CAN bus; Linux; Embedded System

厦门大学博硕士学位论文摘要库

# 目 录

<b>第一章 绪论</b> .....	<b>1</b>
1.1 系统概况 .....	1
1.2 主要研究工作和创新点 .....	4
1.3 本文内容组织 .....	5
<b>第二章 现场总线选型</b> .....	<b>6</b>
2.1 工业以太网简介 .....	6
2.2 CAN 总线的特点 .....	7
2.3 CAN 总线选用原则 .....	9
<b>第三章 CAN 总线应用协议</b> .....	<b>11</b>
3.1 CAN 总线通讯调度方式 .....	11
3.2 通讯帧格式 .....	14
3.3 应用层数据包拆分格式 .....	16
3.4 应用层协议 .....	16
3.5 CAN 总线时间参数设定与分析 .....	17
3.6 下位机模块通讯控制策略 .....	18
3.7 嵌入式主机通讯控制策略 .....	19
<b>第四章 硬件设计</b> .....	<b>21</b>
4.1 PXA270 处理器简介 .....	21
4.2 LPC2194 处理器简介 .....	22
4.3 嵌入式主机硬件结构 .....	24
4.4 CAN 总线硬件结构设计 .....	25
4.5 CAN 总线通讯数据流路径设计 .....	28
<b>第五章 Linux 设备驱动程序设计</b> .....	<b>31</b>
5.1 设备驱动程序设计概述 .....	31
5.2 驱动程序的初始化 .....	34
5.3 驱动程序的 open 方法设计 .....	37
5.4 驱动程序的 write 方法设计 .....	41

5.5 驱动程序的 read 方法设计 .....	42
<b>第六章 控制流程基于 <math>\mu</math>C/OS-II 操作系统的实现.....</b>	<b>44</b>
6.1 基于 $\mu$ C/OS-II 的系统设计概述 .....	44
6.2 任务建立 .....	45
6.3 处理器 PXA270 和 LPC2194 间数据收发 .....	47
6.4 处理器 LPC2194 和 CAN 总线间数据收发 .....	48
<b>第七章 测试与分析 .....</b>	<b>51</b>
7.1 测试目的 .....	51
7.2 软硬件环境 .....	51
7.3 测试方案 .....	51
7.4 测试情况分析 .....	53
<b>第八章 总结与展望 .....</b>	<b>55</b>
<b>参考文献 .....</b>	<b>56</b>
<b>致谢 .....</b>	<b>58</b>

# Contents

<b>CHAPTER 1 Introduction .....</b>	<b>1</b>
1.1 System Overview .....	1
1.2 Main Works and Innovations .....	4
1.3 Organization of This Paper .....	5
<b>CHAPTER 2 Select A Fieldbus Type.....</b>	<b>6</b>
2.1 Brief Introduction of Industrial Ethernet.....	6
2.2 Characteristics of CAN Bus .....	7
2.3 Rules of Selection of CAN Bus .....	9
<b>CHAPTER 3 CAN Bus Application Protocol .....</b>	<b>11</b>
3.1 CAN Bus Communication Schedule Mode.....	11
3.2 Communication Frame Format.....	14
3.3 Data Packet Split Format of Application Layer .....	16
3.4 Application Layer Protocol.....	16
3.5 CAN Bus Time Parameter Setting and Analysis .....	17
3.6 Submodule Communication Control Strategy .....	18
3.7 Embedded Mainframe Communication Control Strategy.....	19
<b>CHAPTER 4 Hardware Design.....</b>	<b>21</b>
4.1 Brief Introduction of PXA270 Processor .....	21
4.2 Brief Introduction of LPC2194 Processor .....	22
4.3 Hardware Structure of Host Embedded Controller .....	24
4.4 Hardware Structure Design of CAN Bus.....	25
4.5 Communication Data Stream Path Design of CAN Bus .....	28
<b>CHAPTER 5 Implementation of Linux Device Driver .....</b>	<b>31</b>
5.1 Summarize of Linux Device Driver Design .....	31
5.2 Initialization of Linux Device Driver .....	34
5.3 Open Method of CAN Bus Linux Device Driver.....	37
5.4 Write Method of CAN Bus Linux Device Driver .....	41
5.5 Read Method of CAN Bus Linux Device Driver .....	42
<b>CHAPTER 6 Control Flow Implementation Based on <math>\mu</math>C/OS-II</b>	
<b>Operating System.....</b>	<b>44</b>

6.1 Summarize of System Design Based on $\mu$ C/OS- II .....	44
6.2 Establish Task .....	45
6.3 Transmitting and Receiving between Processor PXA270 and LPC2194 .....	47
6.4 Transmitting and Receiving between Processor LPC2194 and CAN Bus.....	48
<b>CHAPTER 7 Test and Analysis.....</b>	<b>51</b>
7.1 Aim of Test.....	51
7.2 Environment of Test .....	51
7.3 Scheme of Test.....	51
7.4 Result Analysis of Test.....	53
<b>CHAPTER 8 Conclusions .....</b>	<b>55</b>
<b>References .....</b>	<b>56</b>
<b>Acknowledgements .....</b>	<b>58</b>

厦门大学博硕士学位论文摘要

厦门大学博硕士学位论文摘要库

# 第一章 绪论

## 1.1 系统概况

随着现代社会经济的高度发展，环境问题越来越成为人们关注的焦点，燃煤电厂产生的烟气粉尘排放污染问题日益受到重视，排放控制要求越来越高。除尘设备是电厂除发电主机外最大的辅助设备。近年来，袋式除尘器技术发展迅速，适用性越来越广，在电力、水泥、钢铁、冶金和化工等行业得到普遍应用。

袋式除尘控制系统实现袋式除尘器的工艺要求。工艺控制点分散设备各个部位，最远两个控制点间的距离不超过 1 千米，监控的物理量包括温度、压力、湿度等模拟量和电磁阀、电机等开关量。该系统可采用自动、定时和手动来控制。当在自动控制时，由压差计采集滤袋内外的压差信号。当压差值达到设定的极限值时，嵌入式主机通过现场总线发出信号，下位机模块控制离线提升阀立即关闭出气挡板，使过滤停止，稍候脉冲阀打开，回转喷吹管中喷出的脉冲气体陆续地对滤袋进行清灰，使粉尘从滤袋上剥离下来。经过一段时间，嵌入式主机发出信号，脉冲阀关闭，停止喷吹，稍候提升阀（离线阀）提起，打开出气挡板。此时，设备清灰完成，恢复到过滤状态。系统也可以定时控制，即按顺序对各室进行定时清灰。当定时控制时，每室的喷吹时间，每室的间隔时间及全部喷吹完成时间均可以调节。

现场总线控制系统是继传统集散控制系统之后的新一代控制系统，以适应工业控制领域分散化、网络化和智能化的发展趋势，成为当前控制领域的热门技术。CAN 全称为 Controller Area Network，即控制器局域网，相比起各种形式的工业以太网或其他形式的现场总线，CAN 总线技术以高可靠性，长传输距离，低成本而具有巨大的应用潜力。

本论文论述的内容来源于项目《大型电站电袋或布袋除尘器嵌入式控制系统的开发与应用》。项目的主要目标是开发基于 ARM+Linux+CAN 的 32 位嵌入式分布式控制系统。项目成功后，可获得一个通用的开发应用平台，广泛应用于各种工业分布式控制场所。项目先期以电袋除尘器控制系统立项开发。

CAN 总线连接系统各节点，其可靠设计开发实现是整个项目开发的一个关键部分，决定项目的成败。

基于上述应用需求，决定采用目前工业控制领域热门的 32 位 ARM 处理器和 linux 操作系统为技术实现手段，立项开发基于 CAN 现场总线的分布式控制系统。选用高性能 Xscale 微架构处理器 PXA270 作为系统的控制核心芯片，linux 操作系统采用实时性能较好的 2.6 版本内核<sup>[1][2]</sup>。

系统采用功能模块化设计方式。功能模块间以 CAN 现场总线连接，每个功能模块都是一个 CAN 节点，实现分布式应用。考虑设计 4 条独立的 CAN 总线，可考虑按功能模块的种类占用独立 CAN 总线，也可按输入输出占用独立 CAN 总线。要求系统设计达到较高的可靠性和通用性，将来经后续开发和扩展，可以作为通用的工业用嵌入式分布式控制系统开发平台。系统结构图如图 1.1 所示。

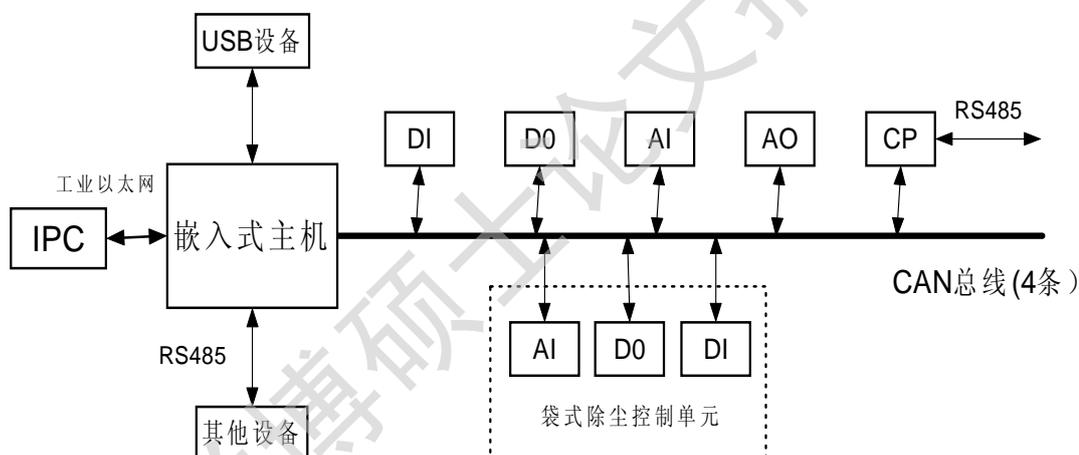


图 1.1 系统结构图

其中：DI 开关量输入模块    DO 开关量输出模块  
 AI 模拟量输入模块    AO 模拟量输出模块  
 CP 通讯模块（协议转换等）

系统功能由多个下位机模块在嵌入式主机的统一组织调度下协同工作而实现，从这个方面看，类似于大中型 PLC 的功能实现结构组织形式。从控制流程上看，系统采用事件驱动型设计。电磁阀开关到位，温度或压力超限等物理事件都会作为触发事件，触发某段功能程序运行，以实现系统预设的工艺控制逻辑。本系统程序运行方式为事件驱动型，与按大周期组织程序运行的 PLC 相比较有本质上的区别，更适用于实时性要求高的控制场合，从这点来看，控制理念更先进<sup>[3]</sup>。

嵌入式主机是整个系统的控制调度核心。响应键盘、CAN 总线及以太网通讯等输入事件。监控下位机插拔、维护功能单元各模块初始化等过程，维护系统信息，保证下位机按系统信息工作。运行控制程序，按功能单元维护和监视整个系统的运行，以达到系统工艺控制要求。对模块传来的数据在逻辑层按预定义的功能单元处理，归档，提供相关界面显示。

功能单元是嵌入式主机维护的一个逻辑层概念，硬件结构上每个功能单元由一个或多个功能模块组成，状态监测由输入模块处理回馈给主机，输出模块按主机的设定动作驱动执行部件，控制和协调由主机完成。各模块都有自己的 CPU，可按工况条件实现一定的智能处理。运行过程中，模块主动周期性地通过 CAN 现场总线汇报本模块运行状况。

在硬件设计方面，嵌入式主机采用 PXA270+ LPC2194 的双处理器结构。

主处理器 PXA270 是高性能 32 位嵌入式微处理器，基于 Intel XSCALE 架构，最高主频可达 624MHz，多个 DMA 通道支持外设数据收发，减轻 CPU 负担<sup>[4]</sup>。

LPC2194 是飞利浦公司基于 ARM7TDMI-S CPU 的 32 位微控制器，CPU 最大操作频率 60MHz，内含 4 个 CAN 控制器，可同时支持 4 条 CAN 总线的操作<sup>[5]</sup>。

下位机模块选用与嵌入式主机负责 CAN 通讯功能 ARM7 芯片 LPC2194 同一系列的处理器，设计双路 CAN 接口，适用于冗余系统（非冗余系统的下位机模块其中一路不焊接）。

在软件设计方面，采用 Linux 操作系统和  $\mu$ C/OS-II 操作系统。

因为要实现以太网、人机界面和文件系统功能等复杂功能，所以在主处理器 PXA270 上选用 Linux 操作系统，内核版本为 2.6.9。

嵌入式主机 LPC2194 处理器和下位机模块的 LPC2xxx 处理器的编程实现都选用实时性性能好的  $\mu$ C/OS-II 操作系统。保证高优先级任务调度运行的实时性，使影响系统安全的紧急事件得到及时处理。

电袋除尘系统的控制逻辑在主机 PXA270 处理器中运行，ARM7 处理器 LPC2194 专门负责 CAN 总线通讯的实现。设计规划好系统 CAN 通讯数据流路径，使控制参数及工况参数在 PXA270、LPC2194、CAN 总线、下位机模块间顺利交互，达到系统工艺要求的控制目标。

## 1.2 主要研究工作和创新点

CAN 总线通讯数据流贯穿嵌入式主机和下位机模块,使工业现场工艺控制逻辑得以实现。其实现效果决定项目的成败。本论文将详细论述 CAN 总线在系统中的实现的各方面相关内容。

### 1、主要研究工作

- (1) 作为工业现场控制,需要保证控制的实时性,本文设计专用的《CAN 总线通讯协议》,以时间片方式调度 CAN 总线上各节点对总线的分时占用,规定系统 CAN 总线应用层中通讯调度,应用层数据包拆分格式,数据链路层通讯帧格式等方面的内容;
- (2) CAN 总线通讯方面的硬件结构设计和实现。采用双处理器结构,需要很好规划主处理器 PXA270 和通讯处理用处理器 LPC2194 两者的功能分工。系统工艺逻辑实现程序和系统管理功能程序在主处理器 PXA270 中运行。实时性要求高的工作在处理器 LPC2194 中处理。两处理器间采用 SPI 接口通讯。
- (3) Linux 设备驱动程序设计。专门设计的 Linux 设备驱动程序实现两处理器间通讯,两处理器间采用 SPI 接口通讯速率为 3.75MHz,如果处理器 PXA270 采用中断方式接收每个字节数据,则 Linux 的运行将频繁因为接收数据而在内核态和用户态来回切换,运行效率差,甚至会因为中断来不及响应而造成数据丢失。所以 Linux 设备驱动程序设计时,使 PXA270 采用 DMA 方式实现和处理器 LPC2194 间数据的收发。
- (4) 研究工业环境下 CAN 总线通讯的可靠性,满足系统可靠性需求,测试系统性能是否能达到设计要求。

### 2、本文的创新点

- (1) 设计基于时间片调度的 CAN 总线通讯协议,实现由事件驱动的工业环境下的分布式控制;
- (2) 主处理器 PXA270 和通讯处理用处理器 LPC2194 两者在 CAN 总线通讯中分工协作的实现;
- (3) 专用 Linux 设备驱动程序设计,实现主处理器 PXA270 的 DMA 功能。

Degree papers are in the "[Xiamen University Electronic Theses and Dissertations Database](#)". Full texts are available in the following ways:

1. If your library is a CALIS member libraries, please log on <http://etd.calis.edu.cn/> and submit requests online, or consult the interlibrary loan department in your library.
2. For users of non-CALIS member libraries, please mail to [etd@xmu.edu.cn](mailto:etd@xmu.edu.cn) for delivery details.

厦门大学博硕士论文摘要库