

# 基于 RFID 实时监控系统的通信数据处理方案研究

高 飞

(重庆大学, 重庆 400044)

**摘 要:**扼要分析 RFID 技术在柔性制造系统(FMS)中的优势,通过介绍 RFID 在汽油发动机生产线监控执行系统中的实际应用,给出系统的整体参考架构,进一步重点分析系统中数据通信中间件技术的选择、边缘层的数据过滤技术及通信数据的持久化方案,从而对基于 RFID 系统中通信数据处理的实际设计方案给出了一种可行的、有价值的参考。

**关键词:**RFID;实时监控系统;通信中间件;数据过滤;数据持久化

**中图分类号:**TP311.13

**文献标识码:**A

**文章编号:**1673-629X(2007)11-0096-03

## Research on Communication Data Processing Program of RFID - Based Real - Time Monitoring System

GAO Fei

(Chongqing University, Chongqing 400044, China)

**Abstract:** Briefly analyses the advantage of the RFID technology in the flexible manufacturing system (FMS), then introduces the actual application of RFID in the petrol engine production line of monitoring execution system. Further analyses the choice of data communication middleware technology, data filtering technology of the edged server layer and the persistent program of data communication. In short, give a desirable and valuable reference of the actual designing program of the RFID-based data communication system.

**Key words:** RFID; real-time monitoring system; communication middleware; data filtering; data persistence

### 0 引 言

随着信息技术的广泛发展,生产技术发生了革命性的变化,柔性制造系统(FMS)已逐渐开始在离散制造业中崭露头脚。传统的生产系统由于缺乏信息监测手段,无法随生产条件及生产目标的改变而动态均衡调整生产,故障事件得不到及时处理,物流缺乏指导,工位连接不畅导致阻塞和缺料事件。条形码技术以其较低成本,较高的准确性、可靠性及输入速度快等优点得到了行业的广泛认可<sup>[1]</sup>。但随着高度柔性化系统对数据采集精确度日益提高,同时生产也要求物品实现真正的自动化管理,RFID(射频识别)技术应运而生。

与传统的条形码识别相比,射频识别技术有更大的读写距离和范围,可以直接识别生产流水线上移动中的智能吊架标签,并且避免了条形码可能受到外部灰尘、油污等环境污染而对识别造成的影响<sup>[2]</sup>。此外更重要的是射频识别技术不但可以将射频标签的数据读出,还可以通过终端来进行数据重复写入。

文中的应用背景正是基于离散制造业的生产过程实时监控执行系统,采用 RFID 的智能识别替代原有的条码识别技术,使车间管理由繁化简,更加体现数据采集的实时性及准确性,使生产自动化理念得到了彻底体现。

### 1 基于 RFID 实时监控系统的构成

图1描述的生产过程实时监控执行系统的 RFID 解决方案参考架构分六层实现:数据采集层、RFID 边缘服务器层、数据持久层、企业服务器层、集成层和应用层。

\* 数据采集层,位于架构的最底层,其阅读器由触发器控制,实时采集车间生产线上的产品标签。

\* RFID 边缘服务器层,边缘服务器定期轮询阅读器(例如,每秒两次),以消除重复操作,并执行过滤和设备管理。边缘服务器还产生 ALE 事件并将事件发送到上层应用。

\* 持久层,负责底层通信数据与上层 ERP 数据的持久化操作,并存储电子标签事件。

\* 企业服务器层,处理及控制企业应用层的业务逻辑,同时提供对持久层的集中管理。

收稿日期:2007-01-22

作者简介:高 飞(1981-),男,四川成都人,硕士研究生,主要研究方向为电子商务与现代物流。

\* 企业应用层, 提供人机交互界面, 实现对业务的操作和管理。

\* 集成层, 集成现有 ERP, 通过使用组件服务, 最大化已有的业务处理资源。通过抽取、处理、分析不同业务数据, 进行数据和业务的整合。

RFID 控制信息通过操作门户流入集成层, 然后流入企业服务器层、持久层、边缘层, 最后通过工位控制器流入阅读器; 而阅读器获取的 RFID 数据则在过滤后顺着这个链向上传送。

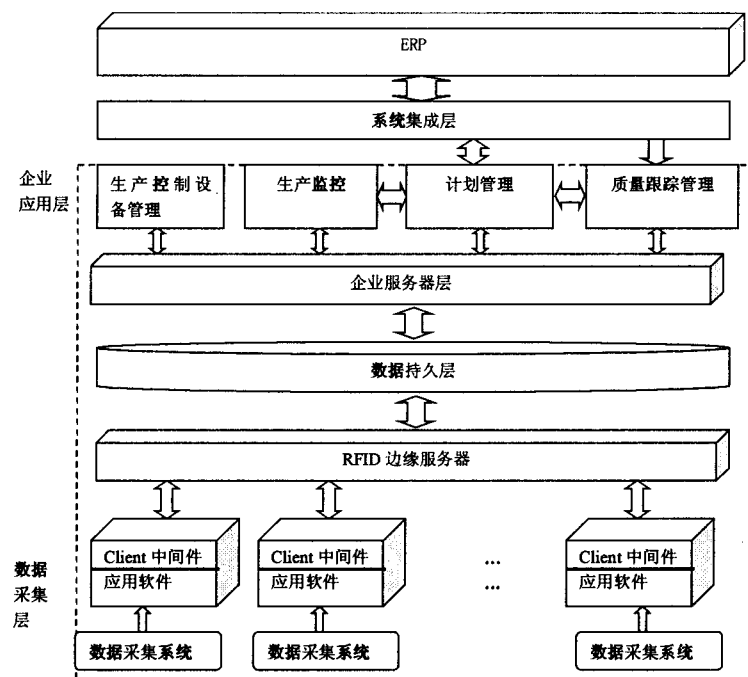


图 1 生产过程实时监控执行系统的 RFID 解决方案参考架构图

## 2 通信中间件技术模型选择

中间件是处于操作系统和应用程序之间的软件, 它是一种独立的服务程序, 分布式应用软件借助中间件在不同的技术之间共享资源<sup>[3]</sup>。网络通信中间件能在不同平台之间通信, 实现分布式系统中可靠的、高效的、实时的跨平台数据传输。通信中间件最常见的三种形式包括: 会话编程(如 TCP/IP 的 Socket)、远程过程调用(RPC)以及消息队列中间件(MOM, message-oriented middleware)。

### 2.1 远程过程调用(RPC)

远程过程调用(RPC)是一种同步对话方式模型<sup>[4]</sup>。RPC 为客户机提供向后端服务器申请服务的通信机制, 如果你把客户机/服务器应用程序想作是一个分割的程序, 则服务器运行数据访问部分, 因为它离数据最近, 客户机则运行数据表示和与用户交互的前端部分。这样, 远程过程调用可看作是把分割的程序

通过网络重组的部件。使用 RPC 通信实质上就是把程序分为主调程序和被调程序两类, 两者之间以典型的 C/S 模式进行信息交流。而且主/被调程序一般是同步、基于连接的。

### 2.2 会话编程

会话编程也是一种同步的对话模式。会话通信是在对等的通信双方之间存在一种基于应用程序的私有逻辑通信, 在整个会话过程中一直保持该逻辑连接被通信双方的应用程序独占。一个程序也可以同时参加

多个对话, 当逻辑连接与程序连接关系紧密时, 一般采用会话编程。会话编程要求程序间的耦合度很高, 因此编程时需过多关注底层通信的细节问题。

### 2.3 消息队列中间件

消息队列中间件采用的是异步通信模式, 它利用高效可靠的消息传递机制进行平台无关的数据交换, 并基于数据通信来进行分布式系统的集成。其实现原理是: 消息的发送者把自己想要发送的信息放入一个容器中(Message), 然后把它保存至一个系统公用空间的消息队列(Message Queue)中; 本地或者是异地的消息接收程序再从该队列中取出发给它的消息进行处理。消息传递机制中传递的消息由通信双方商议协定, 这样做的好处是: 一方面对数据进行了加密, 另一方面采用自己定义的通信格式可以节省传递量。此外消息也可以含有时间戳, 以

便于接收方对某些与时间相关的应用进行处理。消息还可以含有到期时间, 它表明如果在指定时间内消息还未到达则作废。

### 2.4 确定模型

RFID 读写器每秒能捕捉 120~400 个信号, 产生大量需要处理的数据, 为了处理庞大的数据流, 必须采用多线程多路技术。考虑到线程处理的异步及程序的完整、可移植性, 以及通信应用程序的松耦合度, 系统中通信机制采用了消息队列服务模型。采用消息队列的好处有: 由于是异步通信, 无论是发送方还是接收方都不用等待对方返回成功消息, 就可以执行余下的代码, 因而大大地提高了事物处理的能力; 在信息传送过程中, 信息发送机制具有一定功能的故障恢复能力; 消息传递机制使得消息通信的双方具有不同的物理平台成为可能。具体解决方案是采用阻塞 I/O 技术, 为每个客户端打开一个套接字, 并为每一客户创建一个唯一的线程, 通过批量信息传送, 将许多请求打包到一个数据包以减轻网络拥塞, 缩短多个请求通过安全层和

其他代码层所需的时间。

### 3 边缘服务器层数据流过滤技术

边缘服务器层(简称边缘层)位于企业应用层与通信中间件之间。以保障信息传送,以及执行任何所需的数据格式转换。由于通信中间件只保证数据流无差错实时传送并不负责对传输数据包的解析、分组等复杂工作。因此实时采集的原始数据流经过通信中间件到达边缘层或由企业应用层下发的指令、数据等信息将交给边缘层作统一处理。

#### 3.1 生成 ALE 事件

边缘层处理的数据来源有两个方向:一种数据导向来于 RFID 读写器采集的数据流,另一种来于企业应用层下发的指令及数据。在分布式事务处理系统中要处理大量事务,常常在系统中要同时做大量的事务。例如在联机事务处理系统(OLTP)中,每笔事务常常要多台服务器上的程序顺序地协调完成,一旦中间发生某种故障时,不但要完成恢复工作,而且要自动切换系统,达到系统永不停机,实现高可靠性运行;同时要使大量事务在应用服务器能实时并发运行,并进行负载均衡地调度,实现昂贵的可靠性机和大型计算机系统同等的功能,为了实现这个目标,要求系统具有监视和调度整个系统的功能<sup>[5]</sup>。因此作为一个事务处理平台,根据 X/OPEN 的参数模型规定,应由事务处理中间件、通信处理中间件以及数据存取管理中间件三部分组成。

本应用中边缘层将事务处理与数据存取管理做统一调度,通信处理则交由通信中间件执行。事务的生成与执行采用 ALE(应用级事件)标准(ALE 由 EPC-global 软件行动小组中超过 100 多家的领先企业研究开发,并建立了支持 EPC 的软件产品在收集、管理和安排 EPC 技术在供应链中生成的数据时所应采用的方法)。ALE 能有助 RFID 使用者选择希望接收的 RFID 数据,避免因接收过量不需要的数据而造成系统超负荷。

#### 3.2 消息传递机制

由于网络、异常、人为误操作等原因,数据可能会出现丢失、差错、重复等状况,因此不能保证消息传递的“一次发送正确性”。另一方面事务的执行过程:消息在事务化入队和出队中,若操作存在不一致性,即一个操作尚未完成,如消息入队,就不能保证它已经入队,同样也不能保证真的处理了出队。因而在消息传递中除了采用异步传输模式外,在边缘层还需设置处理时序,设定保障机制,即与实时更新数据库建立一种匹配,对数据进行轻量级侦听检测。若传输数据与数

据库中更新之间存在完美匹配,则将丢弃重复数据,若两者之间存在完全匹配,则修改丢失或差错数据项。若两者不存在匹配,则作为一次新的事务进行提交处理。

#### 3.3 单点故障的解决

为了减缓数据流写入中央数据库给系统带来的瓶颈,需要消除集成层的单点故障,可行的办法是将数据暂时存放在内存,事件完成后将之丢弃或将数据暂时保存在系统文件中,等待一个完整的 ALE 事件完成后,由应用服务器自动从内存或系统文件中提取数据写入中央存储数据库中。这样就可大大减少系统对数据库的依赖性。因此需要合理设置缓存堆栈大小,避免存储大量数据而导致堆栈溢出,同时应避免过度使用内存空间导致系统出现瓶颈,一般推荐缓存大小应设置为磁盘数据块的整数倍,本系统中设置为 16kB,当缓存数据容量达到 16kB 时,若一个 ALE 事件尚未完成,则先将缓存数据写入数据库,同时在中央数据库的临时表中记录当前完成操作的日志,以便于下一次提交时事务的确认。若一个 ALE 事件已经完成而缓存容量尚未达到 16kB,则也将数据提交写入数据库。

### 4 数据流的持久化

实时监控需要收集生产过程中大量的实时数据,并且需对实时事件能及时进行处理,车间实时监控需要同时与企业计划层和数据采集层保持双向通信能力,从上下两层接收相应数据并反馈处理结果和生产指令,因此底层与上层频繁的数据流交互成为系统伸缩和扩展的一个瓶颈,此外计划层或采集层业务需求的灵活变更也会给系统的维护带来一定的困难。在业务层和关系数据库之间增加一个健壮的持久层,负责实现对象和关系数据库之间的映射。利用这个持久层框架的机制,对象与关系数据库之间的转换就可以透明地进行,而不用去关心数据库连接、并发性和事务等问题,即业务层就可以直接获取或存储对象,中间的转换过程就交给框架处理了。这样就可以避免由于数据库结构或类中任何一方变化时系统的大幅修改,极大减少了系统层次之间的耦合性,同时高效的缓存数据处理和对象持久化策略可以极大消除系统的瓶颈<sup>[6]</sup>。

### 5 结 论

通过剖析现有离散制造业采用“RFID 电子标签”信息采集模式的优点,进一步通过分析 RFID 在生产过程实时监控执行系统中的应用,给出了一种通信数

(下转第 102 页)

```

CMP AL,24H
JL STEP2
MOV AL,00H
STEP2:MOV CH,AL
MOV AH,6
MOV DS,AX
LEA SI,OLD
MOV AX,354AH
INT 21H
MOV WORD PTR [SI],BX
MOV WORD PTR [SI+2],ES
STEP:MOV AH,7
INT 1AH
INT
CODE ENDS
INT 1AH
JC STEP1
MOV CX,3
SETT:MOV AH,2
MOV DL,8
INT 21H
LOOP SETT
MOV BX,CS
MOV DS,BX
MOV DX,OFFSET INTIA
21H END START

```

程序结构分为驻留和非驻留两部分。驻留部分是用户的报警中断程序,除鸣笛外,还要在退出前恢复

INT 4AH 中断向量。非驻留部分完成保护 INT 4AH,用 INT 1AH 的 7 号功能取消原报警时刻,同时设置 4AH 新的中断向量,使之指向用户报警程序,用 INT 1AH 的 6 号子功能设置报警时刻,然后使用中断 INT 27H 中断驻留退出。

#### 4 结 论

驻留内存程序是一种较特殊的程序。由于加载后一直占有系统资源,会影响系统的整体效率。所以,一般情况下尽量不要使用驻留程序。特别在 DOS 这类主要是在单任务下工作的操作系统中如使用不当会使系统工作效率降低或不能正常工作,甚至造成整个系统的崩溃。因此,在开发应用程序时,若必须使用驻留程序,其程序长度要尽量缩短,以减少驻留的内容。

#### 参考文献:

(上接第 95 页)

- [11] Lu Baoliang, Wang Kaian, Utiyama M, et al. A part - versus - part method for massively parallel training of support vector machines[C]//Proceedings of IEEE/INNS Int. Joint Conf. on Neural Networks (IJCNN2004). Budapest, Hungary: [s. n.], 2004: 735 - 740.

- [12] SMOBR - ASMO Program for Training SVMs[EB/OL]. 2001. <http://www.litc.cpdee.ufmg.br/~barros>.  
[13] Chang Chih - Chung, Lin Chih - Jen. LIBSVM: a library for support vector machines[EB/OL]. 2001. <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.

(上接第 98 页)

据流处理的解决方案。采用以上技术的解决方案已成功地应用于基于 RFID 的汽油发动机生产线监控执行系统中,并在应用中取得了较好的效果。但本方案还不完善,未能很好地解决安全性问题,如何在套接字接口实现安全认证、身份验证、授权和审计等还有待进一步研究。

#### 参考文献:

- [1] 周四军,王建宇. 基于条形码技术的车间监控系统的实时信息采集[J]. 电子技术应用, 2000(5): 21 - 23.

- [2] 王璐,秦汝祥,贾群. 基于 RFID 技术的门禁监控系统[J]. 微机发展, 2003, 13(11): 59 - 63.  
[3] 秦王景,储方杰,高文. 中间件技术研究[J]. 计算机应用研究, 2006, 23(2): 292 - 296.  
[4] 李俊平. 银行大机互连通信中间件的设计[J]. 计算机与数字工程, 2000, 28(6): 58 - 63.  
[5] 高建华,沈莹. 一种新的交易事务处理程序设计方法[J]. 计算机工程与科学, 2001, 23(5): 87 - 90.  
[6] 周函,周波,董金祥. 对象持久层高性能 Cache 实现技术[J]. 计算机应用研究, 2003, 20(11): 35 - 37.