

文章编号: 1001-0920(2002)05-0541-04

基于嵌入式 Internet 技术的 Webit 体系结构研究与实现

金欢, 阮冠春, 徐凌宇, 赵海

(东北大学 信息科学与工程学院, 辽宁 沈阳 110004)

摘要: 基于嵌入式 Internet 技术, 提出一种面向设备动态驱动的异种网络互连体系结构——Webit, 并建立 Webit 模型。该体系结构采用 thin TCP/IP 协议栈技术建立 Embedded Web Server, 实现了将现场设备接入 Internet 的目的; 构建嵌入式 XML 中间件, 实现实时多任务系统与 Embedded Web Server 的集成, 使现场可编程设备的远程监控、管理及驱动程序动态下载得以实现。Webit 体系结构使工业生产自动化系统和智能家电的远程化成为可能, 并为其提供了技术上的支持。

关键词: 嵌入式 Internet; Webit; 实时多任务; 嵌入式 XML; thin TCP/IP

中图分类号: TP 393.1

文献标识码: A

Research and implementation of Webit architecture based on embedded internet

JIN Huan, RUAN Guan-chun, XU Ling-yu, ZHAO Hai

(School of Information Science and Engineering, Northeastern University, Shenyang 110004, China)

Abstract: Based on the technology of the embedded internet, the dynamic device-driver oriented architecture of connecting heterogeneous networks, called Webit, is proposed and the Webit model is established. This architecture introduces thin TCP/IP stack technology to upbuild embedded web server, and actualizes the connection of locale devices and internet. The design of embedded XML middleware realizes the integration of multi-task RTOS and embedded web server. The function of control, management, and downloading of programmable drivers can be implemented. The Webit architecture provides the possibility and technological assistance for industrial automation and remoteness of intelligent household appliances.

Key words: embedded internet; Webit; multi-task RTOS; embedded XML, thin TCP/IP

1 引言

随着计算机网络和通信技术的不断发展, 人们对以 IP 网络为基础的 Internet 网络应用提出了更

高的要求, 将来的 Internet 必将实现“利用 IP 技术传输控制行为”。嵌入式 Internet 技术的出现很大程度上解决了众多不同类型的网络和 Internet 的互连问题, 人们可通过现有的网络设施, 不受地域限制,

收稿日期: 2001-08-31; 修回日期: 2001-10-30

基金项目: 国家自然科学基金项目(69873007)

作者简介: 金欢(1976—), 男, 浙江杭州人, 硕士生; 从事嵌入式 Internet 技术研究; 赵海(1959—), 男, 辽宁沈阳人, 教授, 博士生导师, 从事嵌入式 Internet 技术、数据融合等研究。

实现对控制网络中各节点现场设备的实时监控。但单一的实时监控并不能满足现场动态环境的复杂变化。对于技术人员,更希望能够通过现有的公共通信网络,对现场可编程设备进行访问,如下载驱动程序、实施远程维护等^[1]。

Webit 体系结构的提出使现场可编程设备驱动远程下载及诊断与维护成为可能,并提供了技术上的支持。在该领域中,应用嵌入式 Internet 技术,将 Embedded Web Server 与实时多任务系统集成一体,通过嵌入式 XML 中间件,可将设备驱动字节代码下载到现场设备的微处理单元 MCU 上。实时多任务系统能够动态地创建新任务,并做一定目的的操作。该体系结构具有开放性和实时性的特点,为工业控制和信息家电的研制提供了基础。

在 Webit 模型中,本文构造了基于多线程并发机制的 Embedded Web Server 模型和嵌入式 XML 中间件,为 LAN, WAN 和 Internet 客户提供了透明服务;同时在 Embedded Web Server 上增加了安全机制,以保证工业设备数据的安全。

目前,许多大型家电集团公司,如长虹、海尔等,已采用了 Webit 体系结构来开发智能信息家电。

2 Webit 体系结构

Webit 体系结构是一种基于任务动态驱动的 emGateway 体系结构。它可以解决异种网络间的设备驱动与程序动态下载问题。而异种网络的互连取决于不同传输协议的转换,故需解决其通信问题,目的在于隐藏网络细节,向用户提供一致的通信服务,为设备任务的下载提供基础^[2]。

基于这一原则,Webit 体系结构主要分为 3 层:实时多任务系统应用层、嵌入式 XML 中间件层和

Embedded Web Server 层。其体系结构如图 1 所示。

2.1 实时多任务系统应用层

从结构上讲,实时多任务系统应用层包括两部分:一部分为系统内核,即实时执行程序^[3];另一部分为输入输出部分(I/O),负责与现场设备(如传感器、执行器和 PLC 等)相连。

在 Webit 模型中,实时多任务执行程序(iDCX51)负责任务的创建、删除和调度,并提供一种界面:外部调用接口,如 RAC 接口,这种外部接口可使任务由外界下载并使动态创建成为可能。

2.2 嵌入式 XML 中间件层

XML 的基本特征是具有最大可延伸性,它可任意定义新的标签和属性名称;文档结构可进行任意复杂层次嵌套;任何 XML 文档可包含可选语法描述,且这些描述可被需要进行结构合法性验证的应用所使用^[4]。

嵌入式 XML 是适用于特定用户需求的智能工具,用户可自行规定 XML 对象定义,这使得 Web 语言具有一定的针对性。嵌入式 XML 从 Embedded Web Server 层获取 XML 数据流,通过实时多任务系统应用层接口实现对设备的访问和指令操作;并将设备返回数据转化为 XML 格式数据流,提交给网络客户,实现系统的业务规则。嵌入式 XML 可从多种数据源中提取所需数据进行融合,并且这种异步数据源的数据提取操作对用户是透明的。

2.3 Embedded Web Server 层

Embedded Web Server 层是以 thin TCP/IP 协议栈为基础构建而成,实现了 HTTP, TCP 和 IP 协议。它具有 IP 地址,功能与一般意义的 Web Server 一样,通过网络接入设备便可接入 Internet。分布在世界各地的客户机,只要安装 Web 浏览器,就可通过 HTTP/TCP/IP 协议建立与 Embedded Web Server 的连接。

至此,Internet 客户机到实时多任务系统的通信信道就透明地建立起来,设备驱动和任务可由此通道下载至实时多任务系统。客户端可与本地客户一样透明的监控、管理、维护实时多任务系统的各个方面。

3 Webit 模型的实现及关键技术

3.1 Webit 模型的构建

Webit 模型中的实时多任务系统采用实时执行程序 iDCX51,运行于高性能的 8 位 MCS-51 系统微

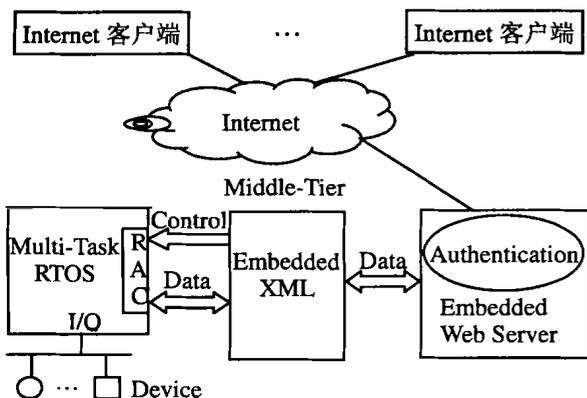


图 1 Webit 体系结构

控制器环境中。该环境的建立采用自行设计的 F2001 智能接口模块, 功能上具备 16 个可读 I/O 口和 8 个可写 I/O 口, 可连接任何类型的现场设备。Embedded Web Server 和嵌入式 XML 中间件是在以 8 位 AVR AT90S8515 微控制器为核心控制器及以太网接口控制器 RTL8019AS 基础上建立而成。外部设置了 Serial E²PROM 存储器, 用于存放用户编写的网页页面和图像。

3.2 Embedded Web Server 与 thin TCP/IP 协议栈

嵌入式 Internet 是一项成熟技术, 在此基础上, Embedded Web Server 的设计必须考虑其特殊性:

1) 最大程度地满足工业网络通信的实时性与灵活性; 2) 满足不同客户的随机访问, 保证互连的开放性; 3) 提供适用于工业设备的安全机制, 确保工业设备的数据安全。因此设计中采用了许多新技术, 如多线程并发与同步、网关技术、安全算法和 thin TCP/IP 协议栈技术等, 用以更好地实现 Embedded Web Server 功能, 提高其性能。

Embedded Web Server 的主要功能如下: 1) 响应不同客户的 HTTP 请求, 并建立连接; 2) 实现与嵌入式 XML 中间件的双向通信; 3) 实现不同客户之间对共享资源的同步操作; 4) 服务器管理以及数据安全性保证。

Embedded Web Server 设计上的难点在于 thin TCP/IP 协议栈的实现, 它需要尽量减小 HTTP/TCP/IP 协议栈而不失去其本身的一般标准性, 这样的服务器可称为“瘦”服务器。所有的 HTTP 协议都支持两种请求: 简单请求和完全请求。简单请求与完全请求的区别在于简单请求只需一行声明所需页面的 GET 行, 无需协议版本。在 Webit 模型中, 网络应用层以 HTTP 协议为主, 而进行的设备级操作针对性强, 因此简单请求即可达到对现场设备的监控及动态驱动程序的下装。为快速响应接收到的数据包, Web 服务器把它传送到 thin TCP/IP 协议栈的简单状态机上进行预处理计算, 减少数据冗余, 大大降低了复杂度。使用这种方法, 可节省微控制器对数据包的处理时间, 更快地响应客户的 HTTP 请求^[5, 6]。

3.3 嵌入式 XML 中间件的结构

Webit 模型中的核心部分——嵌入式 XML 是符合通用接口标准并能完成特定功能的中间件 (COM)。它一方面要从 Embedded Web Server 接受 XML 数据流, 并将其解释后与实时多任务系统的

RAC 接口通信; 一方面接受返回的结果, 按照预先定义好的 XML 对象, 定义封装成 XML 数据流提交给 Embedded Web Server, 由它将数据传送给客户, 完成一次客户请求响应。

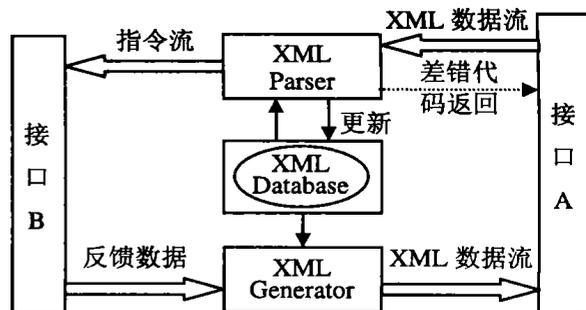


图 2 嵌入式 XML 中间件的结构

在结构上, 嵌入式 XML 主要分为 3 部分 (如图 2 所示): XML Parser, XML Generator 和 XML Database。XML Parser 面向 XML 数据流, 查找 XML Database 中的标签命令对应关系, 将用户自定义的 XML 标签翻译成 RAC 命令, 并将命令提交给 RAC 接口; XML Generator 的作用是将实时多任务系统的反馈数据封装成 XML 数据流返回; XML Database 中存放了 XML 对象定义集 (标签命令对应关系), 并且可以扩充修改。需要指出的是, XML Parser 功能还包括负责 XML 数据流差错信息的返回和 XML Database 的对象定义集的更新。其中, 关键部分在于 XML 对象定义集的遍历与翻译, 其核心代码如下:

```
int ParseXML() {
    int i = 0; // 初始化数据标记指针
    while(i < XMLStreamTags - Length) {
        if (SearchXMLStack(XMLStreamTags(i))) {
            // 出栈
        };
        int found = SearchXMLDatabase
        (XMLStreamTags(i));
        if (found) {
            // 转化为 RAC 指令
            // 压栈
            i++; // 指向下一标记
        }
        else {
            // 返回错误代码
            break;
        }
    }
}
```

};
}

3.4 RAC 接口

在 Webit 模型中, RAC 接口的功能是接受 RAC 控制命令, 并将反馈数据提交给嵌入式 XML。RAC 接口规程是一系列有明确规定、有具体内容的信息格式^[7]。在对任务进行管理时, 其主要手段就是正确发送和接受带有 RAC 参数的命令。

RAC 接口认可 iDCX51 的命令信息, 用户可存取 I/O 和存储器单元, 建立或删除任务, 获取有关硬件、固件或 DCM 系统远程设备上的任务信息。

通过发送一个 iDCX51 命令信息引用 RAC 任务服务, 该命令信息规定 RAC 任务(任务 0)作为它的目的, 并且还在信息头里的相应段中规定相应的 RAC 命令值^[8]。图 3 表示从一个用户任务发送命令信息到 RAC 任务接口的过程。

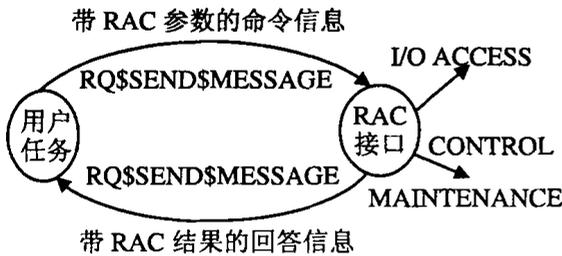


图 3 用户任务与 RAC 接口的通信过程

3.5 实时多任务系统的设计与实现

iDCX51 是分布式控制执行程序, 具有实时多任务操作系统的一切特点, 并且支持全部的 8051 指令系统。在本文模型中, iDCX51 运行在分布式多控制器环境下, 能提供所有实时控制和多任务管理所需的全部功能(其中包括多级控制功能)。任务采用‘抢占式’执行方式, 首先执行优先级高的任务, 从而保证任务的实时性。在实时多任务系统中, 任务 0 始

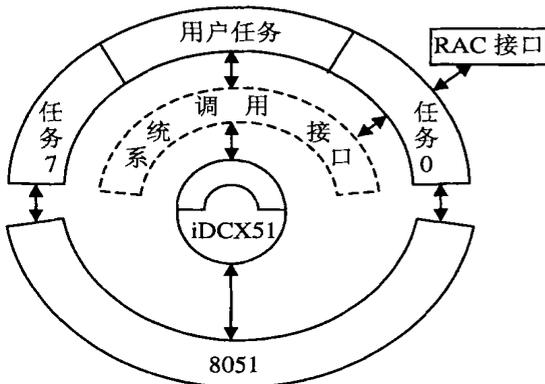


图 4 iDCX51 内核与各部分之间关系 Electronic Publishing House. All rights reserved. http://www.cnki.net (下转第 549 页)

终处于活动状态, 负责与 RAC 接口的通信任务。图 4 给出了 iDCX51 作为内核与 8051 单片机、系统调用及用户任务之间的关系。

4 Webit LAN/WAN 客户端设计与实现

4.1 通过浏览器对现场设备进行监控

Internet 客户是一种真正的虚拟客户。因此在装有浏览器的任何硬件平台上, 不依赖于操作系统的类型就可对现场设备进行监控, 如当前主要流行的 Windows 98/ NT, UNIX/Linux 以及 OS/2 等操作系统。Webit 客户端借用浏览器来监控现场设备, 无需依赖操作系统的类型, 真正实现了中立性和可移植性。

通过浏览器可完成对现场设备的控制, 包括: 1) 复位节点; 2) 查看指定节点的任务情况; 3) 读取指定 I/O 端口的数据; 4) 向指定 I/O 端口写数据; 5) 对指定 I/O 端口进行 AND, OR, NEG 和 XOR 操作。

4.2 客户端程序上载软件(EILoader)

EILoader 是针对 Webit 模型的应用而设计开发的一个客户端软件。它除了具备实时监控现场设备的功能外, 主要解决外部任务的上载、任务的动态驱动和删除问题。软件的设计采用事件驱动技术和模块化设计思想, 定义了一系列的标准函数, 保证了与 Web 服务器通信接口的通用性和开放性。

软件功能结构上分为 4 部分: 任务上载模块、任务创建模块、任务删除模块和获取实时多任务系统当前信息模块。

5 结 论

Webit 体系结构是基于嵌入式 Internet 技术的一种巧妙设计。Webit 模型实现了现场物理设备和最大虚拟网络的无缝连接, 将设备和网络真正融合。它不仅是传统意义上的数据或信息被 Internet 下载, 而且是设备功能的下载, 使 Action over IP 的思想得以实现。目前, Webit 体系结构已成功地应用于网宝(Webit)系列产品的研制开发中。

Webit 体系结构的应用前景十分广阔, 可以广泛应用于家用电器网络化操作、分布式设备监视器网络化、生产自动化系统、楼宇自动化和建筑物监视系统远程化以及电信系统, 因此具有现实意义。

于其它两种算法约 0.2。从实验结果可以看出, 当不能对数据集中全部属性进行离散化时, 离散化效果会下降, 例如在 Glass 和 Glass2 中, BFF 算法得到的是缺少一个属性的约简, 所以它的准确率低于 BCC 算法。可见, 在大数据集中能否对所有属性进行离散化, 将对离散化后的效果产生一定的影响。

2) 本文提出的 BCC 算法的离散化效果比较好, 而且离散化效果相对于基于频率函数的算法, 在 Glass 和 Glass2 两个初始区间较大的数据集中有一定的提高。

3) 由图 1(c) 可见, 各种算法的准确率并没有随记录数有规律地排列, 而是随初始区间数的增加而减少。另外, 在依赖度等指标上的实验也验证了 BCC 算法的有效性。可见, 当数据量较大时, 用 BCC 算法约简的 NS 算法能保持较好的离散化效果。

5 结 语

本文通过对 NS 算法的研究, 在粗糙集理论和属性频率函数的基础上提出了候选核和基于候选核的 BCC 算法, 解决了该离散化算法存在的某些属性不能离散化等问题。实验证明, 该算法具有较好的离

散化效果。本文在研究中总结出的一些规律对于这类算法具有一定的实际意义。

参考文献(References):

- [1] Dougherty J, Kohavi R, Sahami M. Supervised and unsupervised discretization of continuous features[A]. *Proc of 12th ICML[C]*. Morgan: Kauffman Publishers, 1995. 194-202.
- [2] Nguyen H S. Discretization of real value attributes: A boolean reasoning approach[D]. Warsaw: University of Poland, 1997.
- [3] Hu Keyun. Research on concept lattice and rough set based data mining methods[D]. Beijing: Tsinghua University, 2001.
- [4] Hu X H. Knowledge discovery in databases: An attribute-oriented rough set approach[D]. Regina: University of Regina, 1995.
- [5] Ronny Kohavi, Dan Sommerfield. MLC++: Machine learning library in C++ [CP/DK]. <http://www.sgi.com/technology/mlc>, 2001-06-01.
- [6] M Gawrys, J Sienkiewicz. rough set library user's manual [CP/DK]. <ftp://ftp.ii.pw.edu.pl/pub/Rough/>, 1993-09-16/2001-05-08.

(上接第 544 页)

参考文献(References):

- [1] Burton H Lee. Embedded internet system: Poised for takeoff[J]. *IEEE Internet Computing*, 1998, 14(2): 24-29.
- [2] 赵海, 陈飞鸣. Embedded Internet 的体系结构及其 DND C 模型的实现[J]. *东北大学学报*, 1999, 20(3): 257-260.
(Zhao Hai, Chen Feiming. Embedded internet architecture and establish of the ONDC module[J]. *J of Northeastern University*, 1999, 20(3): 257-260.)
- [3] Garvey A, Lesser V. Design to time real-time scheduling[J]. *IEEE Trans on Systems, Man and Cybernetics*, 1993, 23(6): 58-67.
- [4] Elliott e Rusty Harold. XML 实用大全[M]. 北京: 中国水利水电出版社, 2000. 83-85.
- [5] Ready J F. VRTX: A real-time operating system for embedded microprocessor applications[J]. *IEEE Micro*, 1986, 6(4): 8-17.
- [6] P Paulin, M Cornero, C Liem, et al. Trends in embedded systems technology[A]. *Hardware/Software Code-sign*, Kluwer Academic[C]. Publishers, 1996.
- [7] 赵海. 现场总线网络通信模型中的逻辑链路层[J]. *东北大学学报*, 1996, 17(6): 230-234.
(Zhao H. Logic link layer in the communication model of fieldbus network [J]. *J of Northeastern University*, 1996, 17(6): 230-234.)
- [8] 陈功富, 韩贤东. 计算机网设计与实现[M]. 北京: 人民邮电出版社, 1994. 146-167.