

基于 XML 和二维码的家电识别和控制方法

田源¹, 丁香乾¹, 王晓东¹, 许晓伟¹, 陶冶²

(1. 中国海洋大学 信息科学与工程学院, 山东 青岛 266100;

2. 青岛科技大学 信息科学技术学院, 山东 青岛 266061)

摘要: 目前,越来越多的家电提供了智能终端的控制功能,但由于家电设备的功能控制指令因品牌、型号的差异而不同,因此向家庭控制系统中添加新的家电设备时就需要修改固化在原有软件系统中的设备控制指令集。为了避免对软件的重构工作,基于 XML 标签文本实现了描述家电操作指令的功能,并且使用二维码来进行设备识别。控制设备通过解析从二维码中取得的 XML 文档来获得设备的控制指令规则和数据组织规则。这种设备识别方式使得新的家电设备加入控制系统时,可以在不修改原有控制软件的前提下配置新家电的控制指令规则和数据组织规则,从而实现了设备的动态添加功能。

关键词: 数字家庭; 设备描述; 设备识别; XML 标签; 二维码

中图分类号: TP391

文献标识码: A

文章编号: 1673-629X(2013)12-0244-05

doi: 10.3969/j.issn.1673-629X.2013.12.058

Method of Home Appliances Recognition and Control Based on XML and 2D Barcode

TIAN Yuan¹, DING Xiang-qian¹, WANG Xiao-dong¹, XU Xiao-wei¹, TAO Ye²

(1. College of Information Science and Engineering, Ocean University of China, Qingdao 266100, China;

2. College of Information Science and Technology, Qingdao University of Science & Technology, Qingdao 266061, China)

Abstract: More and more appliances support the remote control by smart devices now. However, because of the difference cause by different brand and type for function control set of home devices, when adding new devices to domestic control system, a big modification on the control software would be brought in for the differences in the instruction and data arrangement of the devices. To avoid the unnecessary reconstruction of the software, provide a new kind of appliance recognition system based on XML tags and 2D-barcode. The control device uses the XML profile which is retrieved from the 2D-barcode to configure the rules of devices' instruction and data arrangement. It enables dynamical configuration for new devices without modifying the control software.

Key words: digital home; device description; device recognition; XML tag; 2D-barcode

1 概述

近年来,伴随计算机软硬件技术的飞速发展和智能手机等移动终端产品的普及,越来越多的研究开始偏向于通过智能设备或者高性能计算机来控制家用电器等传统设备。但是由于家用电器往往配置低端,多数并不带有操作系统,而是仅仅通过内部二进制指令或者控制电路来实现电器指令控制。因此智能设备如

何能够在不改造原有设备的基础上实现对家电的识别和配置就成为众多研究的方向之一。随着很多家电厂商将家庭控制中心的概念引入到家庭环境中,家用电器种类的多样性和家电控制指令的巨大差异性就成为对家电统一管理需要克服的难题。当有新的家电加入到已有的控制系统中时,往往会面临软件系统的重构,这大大降低了家庭控制中心的可扩展性。

收稿日期: 2013-02-22

修回日期: 2013-05-27

网络出版时间: 2013-08-28

基金项目: 国家科技支撑计划项目(2011BAH16B03, 2012BAH12F01, 2012BAH15F01, 2012BAF11B01, 2012BAF11B02, 2012BAF11B03, 2011BAH15B05); 青岛市科技计划基础研究项目(12-1-4-1-(9)-jch, 12-1-4-1-(10)-jch)

作者简介: 田源(1987-),男(土家族),山西大同人,硕士研究生,研究方向为嵌入式应用、物联网服务技术;丁香乾,研究生,教授,博士生导师,研究方向为计算智能。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20130827.1430.009.html>

文中考虑使用一种统一的方式来描述家电设备的功能信息,特别是家电控制指令。在现有的研究中,有很多针对设备描述的方法应用在家庭或者工业领域中。文献[1]基于 Java 的 RMI 分布式模型,介绍了基于 Web 应用程序的家电控制方式,虽然提出了在应用层的客户端(浏览器)与服务端(远程设备)的交互应用,但是具体的部署却需要基于支持 JVM 和 Web 服务器的家庭设备上实现。介于家庭设备普遍的低端特征,这种方式具有很大局限性。文献[2-5]均提出了在工业控制领域中,使用 XML 对于设备的描述方式。它们大多基于某种行业标准模型来对设备进行描述。如文献[2]中提到的 SCL 是在变电站设备中通用的一种语言。由于在工业应用中,其设备一般具有统一的设计标准,因此底层的指令标准和控制方式都具有一定的规范性。在设计设备控制软件或者中间件时,可以按照统一的标准进行开发。然而在家电产品中,不同的家电品牌,往往具有不同的指令格式和数据组织格式。比如两台不同品牌的家电,并不能像工业设备那样通过统一的标准进行规范。

在智能家庭领域中,对于异构设备之间的通信已经有了较为广泛的研究,如基于 UPnP^[6-7] 协议栈的家庭媒体中心设备的硬件架构和软件设计方案^[8-9]。它的系统模型可以实现在家庭环境中,媒体家电之间进行通信。但是文献中的系统模型并不能包含非媒体播放的家用电器,如冰箱、空调等。并且如果有新的设备要加入到系统中,则必须要支持系统所使用的通信协议栈,才可以实现通信,增加了系统的复杂度。那么能否通过一种通用语言实现设备的描述和识别呢?

文中就针对家庭环境下的设备识别问题提出了一种基于 XML 标签的家电识别方法与识别系统。目前很多关于 XML 对设备功能描述的研究,大多基于 Web 应用程序。比如文献[10]中,根据智能楼宇的应用需求,设计了基于 BACnet/XML 的设备描述与控制模型,而文献中所给出的部署方式均需要基于 PC 实现。而在家庭应用领域中,考虑到成本和复杂度,很少通过 PC 来构建家庭服务网络,而是通过嵌入式设备来实现家电控制。家电的控制往往只是底层的二进制数据的收发,通过一定的规则来提取二进制数据中的有效信息。例如表 1。如果可以将不同家电中的指令和数据规则通过某种描述方式被已有的系统识别,就解决了智能家居控制中加入新设备的问题。

文中根据描述设备功能的需求提出了 DPL(Device Processing Language),它由 DPWS^[11-12] 扩展而来。DPWS(Devices Profile for Web Services) 是由微软和相关团体开发的用于网络设备和计算机兼容设备进行通信的配置文件规范。它可以使互联网中异构的

设备之间通过统一的规范,进行数据交互,实现了设备互连的目的。总体而言,DPL 是对 WSDL、DPWS 规范的继承和扩充,在遵循上述规范约定的主要元素的基础上,针对家庭设备的特点,进行有针对性的扩展。DPL 主要用于描述家电控制指令信息。这里的指令信息特指传统家电最终所能识别的二进制指令,以及家电所返回的包含设备信息的二进制数据。之后,在实际应用场景中,文章提出结合二维码的方式进行识别,增加部署的便捷性和易用性。

表 1 家电指令规则示例

	空调 A	空调 B
指令码长度(十六进制)	36 位	48 位
调温功能(十六进制)	对第 14 位进行加 1 或减 1 操作	对第 22 位进行加 1 或者减 1 操作
开关功能(十六进制)	将第 3 位置 1(开)或 0(关)	将第 5 位置 3(开)或 0(关)

2 设备描述

文中采用 DPL 对设备的基本属性、功能属性等进行描述。DPL 标签库基于 XML 构建,可以将大量信息组织为具有确定意义的整体结构,从而提供描述结构化信息的功能。

DPL 约定了对设备基本属性和功能属性的描述方式。

2.1 基本属性描述

基本属性描述标签用来描述设备的实际特性和具体设备密切相关,主要包括一些不随时间和用户的使用而改变的设备特性。例如:设备的名字、厂家、功率、版本号、设备出厂号等。DPL 涉及的基本属性应涵盖设备描述模型中关于 <ThisDevice> 的规定范围。基本属性描述主要涵盖:型号属性、设备属性、设备类型。

设备的型号属性由 <ThisModel> 元素描述,该元素可扩展,该元素内包含多个子标签。内容包括:设备制造商名称(Manufacturer)、制造商网址(ManufacturerUrl)、设备型号名称(ModelName)、型号值(ModelNumber)、该型号设备网址(ModelUrl)。

设备属性由 <ThisDevice> 元素描述,该元素可扩展,描述结构与 <ThisModel> 类似。内容包括:设备名称(FriendlyName)、固件版本号(FirmwareVersion)、设备序列号(SerialNumber)。

设备类型由 <ThisType> 元素描述,该元素可扩展,描述结构与 <ThisModel> 类似,其内容为设备具体的类型。

下面是一段描述某冰箱设备的示例片段,片段包含“<This Model>”和“<This Device>”两部分信息:

```

<dpl: ThisModel >
< Manufacturer > XXX </Manufacturer >
< ManufacturerUrl > www. XXX. com </ManufacturerUrl >
< ModelName > LCDG - MB - 86 </ModelName >
< ModelNumber > LCDC - MB - 86 </ModelNumber >
</dpl: ThisModel >
<dpl: ThisDevice >
< FriendlyName > Intelligent Refrigerator </FriendlyName >
< FirmwareVersion > 0. 1 </FirmwareVersion >
< SerialNumber > 34281293791235237492 </SerialNumber >
</dpl: ThisDevice >

```

2.2 功能属性描述

功能描述是 DPL 的主要应用目的。DPL 对设备功能的描述基于 WSDL 标准元素进行扩展,主要涉及数据类型描述(types)、消息类型描述(messages)、服务描述(operation)。而服务描述中所对应的具体实现在 < process > 元素中定义。该元素包含多个 < action > 子标签,子标签中定义具体的操作实现。< operation > 标签下的 input 和 output 定义当前操作的输入和输出。一般对于家电,输入为一串十六进制指令码。设备收到指令码后会回复包含设备信息(如温度、电压、开关状态等)的十六进制指令码。脚本在这里负责处理这些十六进制指令码,从中提取有效信息,处理过程中有可能涉及到简单数学运算,具体内容根据设备不同而异。

在 DPL 中定义数据类型,通过 < types > 标签实现,内容包括数据名称与类型。具体类型可参照编程语言中的简单数据类型。如下面所给示例:

```

< wsdl: types >
< xs: schema xmlns: xs = " http://www. w3. org/2001/
XMLSchema" >
< xs: element name = " Address" type = " xs: int" / >
< xs: element name = " Temperature" type = " xs: int" / >
< xs: element name = " Voltage" type = " xs: int" / >
< xs: element name = " Current" type = " xs: int" / >
< xs: element name = " Power" type = " xs: int" / >
</xs: schema >
</wsdl: types >

```

虽然可以通过 XML 标签定义多种逻辑结构,包括选择、判断、循环、基本数学运算和数据类型转换等。但是考虑到 DPL 文档编写的复杂度,这里推荐通过脚本语言实体实现设备的具体功能,即在 < action > 标签中嵌入可扩展脚本(< script > </script > 标签)的方式。< script > 标签用于定义客户端脚本,包括 Perl、Bash、Python 等引擎支持的脚本。为了防止脚本中的代码和 DPL 标签中涉及到的特殊符号的冲突(如 <、>、:) ,应将脚本代码嵌入到 CDATA 部分中。< message > 标签所对应的域在解析的时候可作为脚本的输入参数或输出参数。

下面代码片段给出了功能“GetTemperature”的定义和其相关元素的定义:

```

< wsdl: message name = " GefTemperatureIn" / >
< wsdl: message name = " GefTemperatureOut" >
< wsdl: part name = " parameters" element = " tns: Tempera-
ture" / >
</wsdl: message >
< wsdl: portType name = " Refrigerator" wse: EventSource = "
false" >
< wsdl: operation name = " GetTemperature" >
< wsdl: input
name = " GetTemperatureIn"
message = " tns: GefTemperatureIn"
dpl: action = " GetTemperatureInProcess" / >
< wsdl: output
name = " GefTemperatureOut"
message = " tns: GefTemperatureOut"
dpl: action = " GetTemperatureOutProcess" / >
</wsdl: operation >
</wsdl: portType >
< dpl: process >
< dpl: action name = " GetTemperatureInProcess" >
< script type = <! -- script type -- > >
<! [CDATA[ <! -- script code -- > ] ] >
</script >
</dpl: action >
< dpl: action name = " GetTemperatureOutProcess" >
< script type = <! -- script type -- > >
<! [CDATA[ <! -- script code -- > ] ] >
</script >
</dpl: action >
</dpl: process >

```

上面示例代码除 dpl 命名空间下的标签以外,其他标签均遵循 DPWS 标准。而 process 标签部分则定义了系统在处理具体数据时需要执行的具体操作。以此来真正实现通过文档解析引擎对设备进行指令控制和数据处理的功能。

3 设备控制

文中所指的控制设备主要针对家庭环境下的媒体中心或者家庭物联网管理设备,该设备直接与底层家电通过某种方式相连接。由于目前大部分家庭设备并非智能设备,其操作方式较为单一,大多只接受直接的二进制指令。设备通过串口、RFID、Zigbee、红外线、蓝牙等方式接收或发送数据,少数家电支持 Wifi 无线网络。而各种家电的指令因生产厂商的不同,指令标准也大不相同,同时设备回传的数据内容格式也不同。比如表 1 中的空调 A 的增加温度的指令码为 0x10100...0A400... ,发出之前要检验第 14 位为有效值(即大

于零且小于 F); 而空调 B 增加温度的指令码为 0x5A0031...141100... ,发送之前要检验第 22 位为有效值(即大于等于 0 且小于 A)。如果要对设备进行统一管理,或者解析设备返回的数据,就需要首先对控制设备进行相应的配置。因此控制设备首先要读取设备的 XML 描述文件,解析其内容,获得指令和数据规则。再通过该描述文件生成设备控制指令码,以具体的连接方式将指令码发送到具体设备并接收设备返回信息。所有这些控制流程和数据格式均可通过 DPL 文档配置。

如下面示例代码片段,该片段定义了底层指令的具体操作内容,通过 Shell 脚本语句实现读取设备温度的数据处理。示例中的具体指令仿照某冰箱内部指令结构,因此数据为实际应用场景下的有效数据。

```
< dpl: process >
< dpl: action name = "GetTemperatureInProcess" >
< script type = "bash" >
<! [CDATA [echo "020A322711223C" ] ] >
< /script >
< /dpl: action >
< dpl: action name = "GetTemperatureOutProcess" >
< script type = "bash" >
<! [CDATA [TEMP = substring( $ 1 , 12 , 2)
echo $ TEMP ] ] >
< /script >
< /dpl: action >
< /dpl: process >
```

上面示例代码片段定义了 GetTemperature 的输入操作和输出操作。输入操作脚本中输出了实际发送的指令码,输出操作脚本提取设备返回数据串中的第 12 位和第 13 位,这里第 12 位和第 13 位表示冰箱保鲜区的温度。

4 二维码信息组织

使用二维码保存 DPL 文档地址和设备基本信息,用于设备识别。凡是具有摄像照相功能的设备均可用于二维码识读(如手机、PDA),避免了设备硬件改造。二维码具有成本低、识读方便、部署简单等特性。部署二维码有多种方式,可以在设备屏幕显示,也可印刷并粘贴于设备外壳。但二维码本身所能包含的信息量较少,所以这里二维码仅保留基本的设备信息,详细的设备指令信息保存在远端服务器。二维码中包含配置文件地址,设备通过读取该地址,可以通过互联网从远端服务器获取设备 DPL 配置文件。图 1 为系统操作流程示意。

下面给出一种二维码内容组织方式,该方式类似于 IP 包头的信息组织方式。基本内容包括:基本描述

信息(设备名称、设备厂商、设备型号、设备种类、其他信息)和获取设备功能的配置文件的网络链接地址。为了进一步缩减信息长度,可以使用短地址服务支持。

考虑到 DPL 配置文件会造成厂商的设备指令泄露,所以可在“其他信息”中加入安全验证码。该安全验证码可作为请求服务器提供 DPL 配置文件的凭证,在一定程度上提供了一定的信息安全性保证。

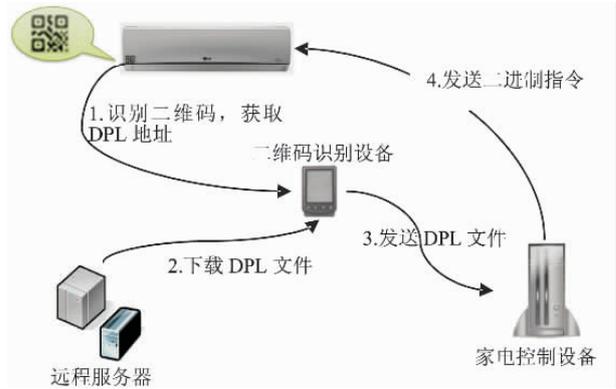


图 1 系统操作流程示意

由于同样的设备拥有相同的指令规则,为了避免 DPL 的重复解析,可以通过设备型号来判断当前设备指令是否在本地图集集中已存在。如果存在,则直接添加设备,如果不存在,则通过网络地址进行 DPL 解析过程。

5 系统实现

该识别系统的具体部署主要由三部分组成:XML 文档存储服务器、控制设备、二维码采集设备。这里的控制设备(如家庭媒体中心/感知设备网络控制器)是运行在家庭网络^[6]中的嵌入式系统控制设备。它可以通过有线的方式与底层家电相连,其部署位置可能为固定的,因此二维码采集设备和控制设备并非一定为同一设备。两者可通过 Wifi、蓝牙等方式互联。

考虑到信息的安全性,二维码采集设备应使用特定的识读软件,用于识别二维码中所包含的安全信息,并作为身份验证信息提供给服务器来获得 DPL 配置文件。

控制设备可以和实际家电进行通信,与家电的连接处于架构的硬件层,包括无线与有线两种方式。和实际家电的通信一般为直接的二进制指令和家电返回的二进制数据。

控制设备具有嵌入式操作系统,可支持多任务运行。由于涉及家电控制,因此会有一定的实时性要求。如 Android 系统的软实时即可满足需求。

在应用软件层包含了与家电设备的控制和识别相关的应用程序模块。

这里的二维码识读模块可以位于独立的设备中,

比如手机终端。DPL 解析引擎位于家电控制设备中,负责对 DPL 文档的解析工作。

服务器端用于存放和更新 DPL 配置文件资源,因此仅仅需要提供 HTTP 文件共享服务即可。

DPL 文件存放在远端服务器,这样有利于对 DPL 文件的维护和更新。获取 DPL 文件并驱动设备的工作流程如下:

a) 识读设备通过读取家电的二维码信息,获得 XML 描述文件的网络地址;

b) 访问该地址,获得存储在远端服务器的 XML 描述文件;

c) 识读设备将描述文件传输到控制设备上,控制设备启动相应的解析引擎解析描述文件,并将对应的家电加入到设备列表中。

在仿真试验中,程序运行在 Android 4.0.4 操作系统的手机上,通过设备自身的摄像头识别二维码,并且通过内置解析引擎实现设备控制指令的解析过程,见图 2。

设备列表中仅有冰箱一项,点击“添加”按钮进入二维码识别和 DPL 解析,见图 3。



图 2 识别空调二维码

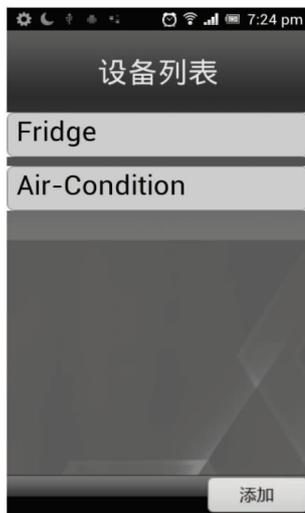


图 3 设备列表(识别后)

在仿真实验中,程序根据设备功能,基于简单的 Button 来实现具体的空调控制操作。

6 结束语

文中提出了一种基于 XML 与二维码的设备描述方法与识别系统。该识别系统针对家庭网络中设备多样性与复杂性所设计。文中给出了一种家庭电器功能描述规范建议,通过该规范,可以使控制设备在不改变原有软件的前提下识别控制新的家电设备。文中的二维码作为设备识别的一部分,提高了设备识别的便利性,同时也避免了对传统家电进行改造,降低系统部署成本。

在实际的应用中可以将已知的设备指令规则写入到 DPL 中,动态地添加家电设备。对于二维码中提到的安全码部分,目前仅是从信息安全的角度提出了设置的建议,可以根据未来的使用需求进一步改进。

参考文献:

- [1] 唐赞玉,刘宏,翟绍军.基于 JavaRMI 技术信息家电远程控制设计与实现[J].电脑知识与技术:学术交流,2007(17):1216-1217.
- [2] 尹家凡,王孙安,盛万兴.XML 语言在变电站设备描述中的应用[J].计算机工程与应用,2003,39(21):209-210.
- [3] 王忠锋,王宏.基于 XML 的现场总线设备描述[J].计算机工程,2004,30(8):183-185.
- [4] 马振芳,冯冬芹.XML 与功能块相结合的设备描述方案[J].计算机应用,2011,31(11):2947-2949.
- [5] 黄仁杰,刘枫.一种基于 XML 扩展的电子设备描述方法[J].西南师范大学学报(自然科学版),2005,30(6):1026-1029.
- [6] Miller B A, Nixon T, Tai C, et al. Home networking with universal plug and play [J]. IEEE communications magazine, 2001, 39(12): 104-109.
- [7] Universal plug and play device architecture 1.0 version 1.0. 1 [M]. [s. l.]: UPnP Forum 2003.
- [8] 吴非,吴浩,刘金玉.基于 UPnP 协议栈的嵌入式家庭媒体适配器的研究与设计[J].计算机应用研究,2008,25(5):1480-1482.
- [9] 陶林彬.基于闪联协议的家庭网络平台的研究与实现[D].成都:电子科技大学,2009.
- [10] 朱斌,蔡昭权,黄超.基于 BACnet/XML 的 BACnet/WS 集成模型的设计与实现[J].电子学报,2008,36(4):781-784.
- [11] Microsoft Corporation. Device profile for Web Service [EB/OL]. 2006-06. <http://specs.xmlsoap.org/ws/2006/02/devprof/devicesprofile.pdf>.
- [12] Bobek A, Zeeb E, Bohn H, et al. Device and service templates for the devices profile for Web Services [C]//Proc of 6th IEEE international conference industrial informatics, Daejeon, Korea [s. n.]: 2008: 797-801.