



(12)发明专利申请

(10)申请公布号 CN 108616855 A

(43)申请公布日 2018. 10. 02

(21)申请号 201810284191.7

(22)申请日 2018.04.02

(71)申请人 上海仪电数字技术股份有限公司
地址 200001 上海市黄浦区福州路666号27
号楼H室

(72)发明人 彭召旺 郭奕丰

(74)专利代理机构 上海弼兴律师事务所 31283
代理人 胡美强 罗朗

(51)Int.Cl.
H04W 4/80(2018.01)

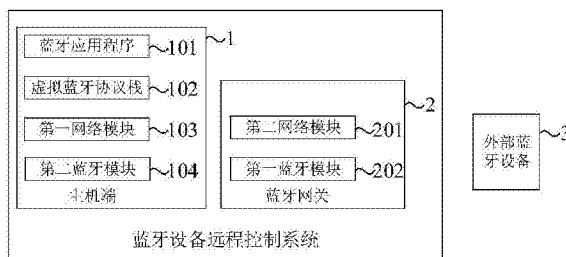
权利要求书2页 说明书11页 附图2页

(54)发明名称

蓝牙设备远程控制系统及方法

(57)摘要

本发明公开了一种蓝牙设备远程控制系统及方法,其中蓝牙设备远程控制系统包括主机端和蓝牙网关;主机端包括蓝牙应用程序、虚拟蓝牙协议栈和第一网络模块,蓝牙网关包括第二网络模块和第一蓝牙模块;第一网络模块用于与第二网络模块通信;蓝牙应用程序用于发送蓝牙控制命令至虚拟蓝牙协议栈,虚拟蓝牙协议栈用于发送蓝牙控制命令至第一网络模块;第一网络模块用于发送蓝牙控制命令至第二网络模块;第二网络模块用于调用第一蓝牙模块执行蓝牙控制命令以使得第一蓝牙模块与外部蓝牙设备通信。本发明的蓝牙设备远程控制系统中主机端与外部蓝牙设备之间的数据传输距离不受限制,蓝牙设备的应用范围更加广泛。



1. 一种蓝牙设备远程控制系统,其特征在于,所述蓝牙设备远程控制系统包括主机端和蓝牙网关;所述主机端用于通过所述蓝牙网关与外部蓝牙设备通信;

所述主机端包括蓝牙应用程序、虚拟蓝牙协议栈和第一网络模块,所述蓝牙网关包括第二网络模块和第一蓝牙模块;

所述第一网络模块用于与所述第二网络模块通信;

所述蓝牙应用程序用于发送蓝牙控制命令至所述虚拟蓝牙协议栈,所述虚拟蓝牙协议栈用于发送所述蓝牙控制命令至所述第一网络模块;

所述第一网络模块用于发送所述蓝牙控制命令至所述第二网络模块;

所述第二网络模块用于调用所述第一蓝牙模块执行所述蓝牙控制命令,所述第一蓝牙模块用于根据所述蓝牙控制命令与所述外部蓝牙设备通信。

2. 如权利要求1所述的蓝牙设备远程控制系统,其特征在于,所述蓝牙控制命令包括蓝牙搜索、蓝牙配对、蓝牙连接和蓝牙通信中的至少一种。

3. 如权利要求1所述的蓝牙设备远程控制方法,其特征在于,所述蓝牙控制命令包括调用方法名称和参数。

4. 如权利要求1所述的蓝牙设备远程控制系统,其特征在于,所述主机端还包括第二蓝牙模块,所述蓝牙应用程序用于判断所述外部蓝牙设备是否在短距通信范围内,若是,则通过所述第二蓝牙模块与所述外部蓝牙设备通信,若否,则通过所述蓝牙网关与所述外部蓝牙设备通信。

5. 如权利要求1所述的蓝牙设备远程控制系统,其特征在于,所述第一蓝牙模块用于将执行所述蓝牙控制命令后的结果发送至所述第二网络模块;

所述第二网络模块用于发送所述结果至所述第一网络模块;

所述第一网络模块用于通过所述虚拟蓝牙协议栈将所述结果发送至所述蓝牙应用程序;

所述蓝牙应用程序用于接收并处理所述结果。

6. 一种蓝牙设备远程控制方法,其特征在于,所述蓝牙设备远程控制方法基于如权利要求1-5任意一项所述的蓝牙设备远程控制系统实现;

所述蓝牙设备远程控制方法包括:

S1、所述第一网络模块与所述第二网络模块通信;

S2、所述蓝牙应用程序发送蓝牙控制命令至所述虚拟蓝牙协议栈,所述虚拟蓝牙协议栈发送所述蓝牙控制命令至所述第一网络模块;

S3、所述第一网络模块发送所述蓝牙控制命令至所述第二网络模块;

S4、所述第二网络模块调用所述第一蓝牙模块执行所述蓝牙控制命令,所述第一蓝牙模块根据所述蓝牙控制命令与所述外部蓝牙设备通信。

7. 如权利要求6所述的蓝牙设备远程控制系统,其特征在于,所述蓝牙控制命令包括蓝牙搜索、蓝牙配对、蓝牙连接和蓝牙通信中的至少一种。

8. 如权利要求6所述的蓝牙设备远程控制方法,其特征在于,所述蓝牙控制命令包括调用方法名称和参数。

9. 如权利要求6所述的蓝牙设备远程控制系统,其特征在于,当所述主机端包括第二蓝牙模块时,所述蓝牙应用程序判断所述外部蓝牙设备是否在短距通信范围内,若是,则通过

所述第二蓝牙模块与所述外部蓝牙设备通信,若否,则通过所述蓝牙网关与所述外部蓝牙设备通信。

10.如权利要求6所述的蓝牙设备远程控制系统,其特征在于,所述蓝牙设备远程控制方法还包括:

S5、所述第一蓝牙模块将执行所述蓝牙控制命令后的结果发送至所述第二网络模块;

S6、所述第二网络模块发送所述结果至所述第一网络模块;

S7、所述第一网络模块通过所述虚拟蓝牙协议栈将所述结果发送至所述蓝牙应用程序;

S8、所述蓝牙应用程序接收并处理所述结果。

蓝牙设备远程控制系统及方法

技术领域

[0001] 本发明涉及通信领域,尤其涉及一种蓝牙设备远程控制系统及方法。

背景技术

[0002] 作为一种无线通信方式,蓝牙可实现蓝牙主机与蓝牙设备之间的通信。蓝牙数据在蓝牙主机与蓝牙设备之间以无线方式进行传输,使用2.4GHz(一种频率单位)工作频段。虽然理论传输距离可达百米,但实际场景中,受发射功率、墙体阻隔等因素限制,消费级蓝牙无线信号的传输范围很小,能保证稳定传输的实际距离往往只有几米或者十几米。这限制了蓝牙主机的应用与蓝牙设备之间的数据传输距离。

[0003] 为扩大传输距离的传统方法包括增加发射功率,移除障碍物,从而使场景更开放,但增加发射功率,会导致功耗增加、设备电池寿命缩短;再者,国家相关法规对发射功率也有限定(2.4GHz频段不得超过100mW(毫瓦)或20dBm(分贝毫瓦)),因而不可能无限增加功率。

发明内容

[0004] 本发明要解决的技术问题是为了克服现有技术中蓝牙主机与蓝牙设备之间的数据传输距离短,应用范围受限制的缺陷,提供一种蓝牙设备远程控制系统及方法。

[0005] 本发明是通过下述技术方案来解决上述技术问题:

[0006] 一种蓝牙设备远程控制系统,所述蓝牙设备远程控制系统包括主机端和蓝牙网关;所述主机端用于通过所述蓝牙网关与外部蓝牙设备通信;

[0007] 所述主机端包括蓝牙应用程序、虚拟蓝牙协议栈和第一网络模块,所述蓝牙网关包括第二网络模块和第一蓝牙模块;

[0008] 所述第一网络模块用于与所述第二网络模块通信;

[0009] 所述蓝牙应用程序用于发送蓝牙控制命令至所述虚拟蓝牙协议栈,所述虚拟蓝牙协议栈用于发送所述蓝牙控制命令至所述第一网络模块;

[0010] 所述第一网络模块用于发送所述蓝牙控制命令至所述第二网络模块;

[0011] 所述第二网络模块用于调用所述第一蓝牙模块执行所述蓝牙控制命令,所述第一蓝牙模块用于根据所述蓝牙控制命令与所述外部蓝牙设备通信。

[0012] 可选地,所述蓝牙控制命令包括蓝牙搜索、蓝牙配对、蓝牙连接和蓝牙通信中的至少一种。

[0013] 可选地,所述蓝牙控制命令包括调用方法名称和参数。

[0014] 可选地,所述主机端还包括第二蓝牙模块,所述蓝牙应用程序用于判断所述外部蓝牙设备是否在短距通信范围内,若是,则通过所述第二蓝牙模块与所述外部蓝牙设备通信,若否,则通过所述蓝牙网关与所述外部蓝牙设备通信。

[0015] 可选地,所述第一蓝牙模块用于将执行所述蓝牙控制命令后的结果发送至所述第二网络模块;

- [0016] 所述第二网络模块用于发送所述结果至所述第一网络模块；
- [0017] 所述第一网络模块用于通过所述虚拟蓝牙协议栈将所述结果发送至所述蓝牙应用程序；
- [0018] 所述蓝牙应用程序用于接收并处理所述结果。
- [0019] 一种蓝牙设备远程控制方法,所述蓝牙设备远程控制方法基于如上所述的蓝牙设备远程控制实现；
- [0020] 所述蓝牙设备远程控制方法包括：
- [0021] S1、所述第一网络模块与所述第二网络模块通信；
- [0022] S2、所述蓝牙应用程序发送蓝牙控制命令至所述虚拟蓝牙协议栈,所述虚拟蓝牙协议栈发送所述蓝牙控制命令至所述第一网络模块；
- [0023] S3、所述第一网络模块发送所述蓝牙控制命令至所述第二网络模块；
- [0024] S4、所述第二网络模块调用所述第一蓝牙模块执行所述蓝牙控制命令,所述第一蓝牙模块根据所述蓝牙控制命令与所述外部蓝牙设备通信。
- [0025] 可选地,所述蓝牙控制命令包括蓝牙搜索、蓝牙配对、蓝牙连接和蓝牙通信中的至少一种。
- [0026] 可选地,所述蓝牙控制命令包括调用方法名称和参数。
- [0027] 可选地,当所述主机端包括第二蓝牙模块时,所述蓝牙应用程序判断所述外部蓝牙设备是否在短距通信范围内,若是,则通过所述第二蓝牙模块与所述外部蓝牙设备通信,若否,则通过所述蓝牙网关与所述外部蓝牙设备通信。
- [0028] 可选地,所述蓝牙设备远程控制方法还包括：
- [0029] S5、所述第一蓝牙模块将执行所述蓝牙控制命令后的结果发送至所述第二网络模块；
- [0030] S6、所述第二网络模块发送所述结果至所述第一网络模块；
- [0031] S7、所述第一网络模块通过所述虚拟蓝牙协议栈将所述结果发送至所述蓝牙应用程序；
- [0032] S8、所述蓝牙应用程序接收并处理所述结果。
- [0033] 本发明的积极进步效果在于：
- [0034] 本发明的蓝牙设备远程控制通过蓝牙应用程序利用虚拟蓝牙协议栈和第一网络模块将蓝牙控制命令发送至蓝牙网关,蓝牙网关利用第二网络模块接收蓝牙控制命令并调用第一蓝牙模块执行蓝牙控制命令,以使得第一蓝牙模块与外部蓝牙设备通信。本发明的蓝牙设备远程控制系统中主机端可远程通过蓝牙网关与外部蓝牙设备进行通信交互,使得主机端与外部蓝牙设备之间的数据传输距离不受限制,蓝牙设备的应用范围更加广泛。

附图说明

- [0035] 图1为本发明的实施例1的蓝牙设备远程控制系统的模块示意图。
- [0036] 图2为本发明的实施例1的蓝牙设备远程控制系统的远程过程调用的模块示意图。
- [0037] 图3为本发明的实施例2的蓝牙设备远程控制方法的流程图。

具体实施方式

[0038] 下面通过实施例的方式进一步说明本发明,但并不因此将本发明限制在所述的实施例范围之中。

[0039] 实施例1

[0040] 本实施例提供一种蓝牙设备远程控制系统,如图1所示,蓝牙设备远程控制系统包括主机端1和蓝牙网关2;主机端1用于通过蓝牙网关2与外部蓝牙设备3通信;

[0041] 主机端包括蓝牙应用程序101、虚拟蓝牙协议栈102、第一网络模块103和第二蓝牙模块104,蓝牙网关2包括第二网络模块201和第一蓝牙模块202。

[0042] 蓝牙应用程序101用于判断外部蓝牙设备3是否在短距通信范围内,若是,则通过第二蓝牙模块104与外部蓝牙设备3通信,若否,则通过蓝牙网关2与外部蓝牙设备3通信。

[0043] 当蓝牙应用程序101判断外部蓝牙设备3在短距通信范围内时,则通过第二蓝牙模块104与外部蓝牙设备3通信。

[0044] 第二蓝牙模块104包括蓝牙协议栈API(应用程序编程接口)、蓝牙驱动模块和蓝牙硬件,蓝牙应用程序101调用蓝牙协议栈API实现蓝牙设备发现过程、设备连接过程、通信过程(向外部蓝牙设备发送指令、获取蓝牙设备数据)等蓝牙功能。蓝牙协议栈API则调用蓝牙驱动模块的有关接口,向蓝牙硬件发出各类通信指令,执行具体相对应的动作以与外部蓝牙设备3通信。

[0045] 当蓝牙应用程序101判断没有搜索到外部蓝牙设备3时,蓝牙应用程序101用于通过蓝牙网关2与外部蓝牙设备3通信:

[0046] 第一网络模块103用于与第二网络模块201通信;

[0047] 蓝牙应用程序101用于发送蓝牙控制命令至虚拟蓝牙协议栈102,虚拟蓝牙协议栈102用于发送蓝牙控制命令至第一网络模块103;其中,蓝牙控制命令包括蓝牙搜索、蓝牙配对、蓝牙连接和蓝牙通信中的至少一种,更具体地,蓝牙控制命令包括调用方法名称和参数。

[0048] 第一网络模块103用于发送蓝牙控制命令至第二网络模块201;

[0049] 第二网络模块201用于调用第一蓝牙模块202执行蓝牙控制命令,第一蓝牙模块202用于根据蓝牙控制命令与外部蓝牙设备3通信。

[0050] 第一蓝牙模块202包括蓝牙协议栈API、蓝牙驱动模块和蓝牙硬件,第二网络模块201用于调用蓝牙协议栈API实现外部蓝牙设备3的发现过程、设备连接过程、通信过程(向外部蓝牙设备发送指令、获取外部蓝牙设备数据)等蓝牙功能。蓝牙协议栈API则调用蓝牙驱动模块的有关接口,向蓝牙硬件发出各类通信指令,执行具体相对应的动作以与外部蓝牙设备3通信。

[0051] 第一蓝牙模块202用于将执行蓝牙控制命令后的结果发送至第二网络模块201;

[0052] 第二网络模块201用于发送结果至第一网络模块103;

[0053] 第一网络模块103用于通过虚拟蓝牙协议栈102将结果发送至蓝牙应用程序101;

[0054] 蓝牙应用程序101用于接收并处理结果。

[0055] 本实施例中的第一网络模块和第二网络模块可包括使用TCP/IP(网络通讯协议)协议的互联网,也可包括但不限于局域网、Wi-Fi(一种允许电子设备连接到一个无线局域

网的技术)无线网,或GPRS/3G/4G/5G(均为网络接入的一种模式)移动互联网等;本实施例中,网络模块的协议采用TCP/IP网络协议。

[0056] 虚拟蓝牙协议栈102是对标准蓝牙协议栈的模拟实现,从功能上看它完全等同于标准蓝牙协议栈,基于虚拟蓝牙协议栈102封装的API,也完全等同于标准蓝牙协议栈API。不同的是,在此种情况下,虚拟蓝牙协议栈102不再直接调用蓝牙驱动模块,而只是将蓝牙应用程序101的蓝牙控制命令通过TCP/IP网络传输给远端的蓝牙网关2,由远端的蓝牙网关2执行实现标准蓝牙协议栈API,后者则会将返回值通过TCP/IP网络回传给虚拟蓝牙协议栈102,再由虚拟蓝牙协议栈102返回给蓝牙应用程序101。

[0057] 主机端1加蓝牙网关2的组合既可实现主机端1与外部蓝牙设备3之间的传统的近距离交互模式,也可以实现远程的交互模式,但蓝牙应用程序是通用的,当主机端不包括蓝牙模块或者超出外部蓝牙设备的直接通信范围时,蓝牙应用程序与远端蓝牙设备间的蓝牙控制命令的交互也可以间接通过虚拟蓝牙协议栈、主机端的TCP/IP网络、蓝牙网关的TCP/IP网络对蓝牙应用程序发出的蓝牙控制命令进行透传,以传送至蓝牙网关的蓝牙模块执行蓝牙控制命令,从而保证了蓝牙应用程序的兼容性,无需对蓝牙应用程序重新定制开发。

[0058] 本实施例中使用远程过程调用(RPC)的方式实现虚拟蓝牙协议栈的API方法。将被调用的方法名称和参数经TCP/IP网络流化后透传给蓝牙网关,蓝牙网关的第二网络模块在收到被调用方法的名称、参数后,调用标准蓝牙协议栈中的API方法,并将获得的结果经第二网络模块通过第一网络模块回传给虚拟蓝牙协议栈,由虚拟蓝牙协议栈转发给蓝牙应用程序。

[0059] 远程调用可以是双向的。虚拟蓝牙协议栈调用蓝牙网关,蓝牙网关也可以调用蓝牙应用程序设置的回调函数,具体的实现是,蓝牙网关里设置一个统一的回调接口,它将参数透传给虚拟蓝牙协议栈,虚拟蓝牙协议栈再调用蓝牙应用程序设置的回调函数将对应的返回数据发送给蓝牙应用程序。

[0060] 关于虚拟蓝牙协议栈的建立,只保留供应用层调用的基于标准蓝牙协议栈的全部API结构,即API命名、参数及返回值均与标准蓝牙协议栈的API保持一致。虚拟蓝牙协议栈的API不调用蓝牙驱动模块和蓝牙硬件,不与蓝牙驱动模块直接交互,而会直接将调用虚拟蓝牙协议栈的各个API方法经过打包封装并通过TCP/IP网络传递给远端的蓝牙网关执行。

[0061] 蓝牙网关端通过TCP/IP网络模块收到主机端发出的包含有蓝牙控制命令的消息后,调用蓝牙网关中的标准蓝牙协议栈执行相应的蓝牙控制命令。

[0062] RPC已非常成熟,已出现了大量的成熟RPC产品或实现。如著名的RPC实现框架有Java(一种面向对象编程语言)RMI(远程方法调用)、Alibaba Dubbo(阿里巴巴开源出来的一个分布式服务框架)、Apache Thrift(Facebook开源的跨语言的RPC通信框架)、Google GRPC(Google开源的一个高性能、跨语言的RPC框架)、Twitter Finagle(Twitter基于Netty开发的支持容错的、协议无关的RPC框架)等。这些产品或框架性能优越,在互联网中应用十分广泛。

[0063] 至于如何实现远程接口调用,作为实施例,可以利用Java的RMI框架。如图2其中RMI客户端从RMI注册簿中查询RMI服务端已注册的可用远程的方法,然后调用RMI stub上的方法,RMI stub将调用的方法的名称、参数打包后通过TCP/IP网络传递给RMI skeleton,RMI skeleton解包后得到被调用的方法和参数,然后调用RMI服务端中的相应的调用的方

法。RMI客户端与RMI服务端通过TCP/IP网络进行连接。

[0064] 具体到本实施例的蓝牙远程调用场景,虚拟蓝牙协议栈和第一网络模块相当于RMI stub,第二网络模块相当于RMI skeleton,主机端的蓝牙应用程序相当于RMI客户端,第一蓝牙模块相当于RMI服务端。具体工作过程如下:

[0065] 主机端的蓝牙应用程序调用虚拟蓝牙协议栈中的API方法;RMI stub(即虚拟蓝牙协议栈)将API方法(变量和方法名)打包,然后通过TCP/IP网络(第一网络模块)发送给蓝牙网关的RMI skeleton(即第二网络模块);RMI skeleton将RMI stub发送来的API方法解包,解析出被调用的方法、参数以及该方法所在对象;RMI skeleton调用第一蓝牙模块中对应的方法,并将执行结果返回给RMI skeleton,RMI skeleton将执行结果打包并发送给RMI stub,RMI stub解包得到执行结果并将执行结果返回给蓝牙应用程序。

[0066] 以蓝牙4.0Android平台主机端执行蓝牙扫描为例。

[0067] 主机端的蓝牙应用程序调用了BluetoothAdapter的API进行设备扫描的操作:

```
[0068]     startLeScan(new BluetoothAdapter.LeScanCallback() {  
[0069]         @Override  
[0070]         public void onLeScan(final BluetoothDevice device, final int rssi, byte[]  
[0071]         scanRecord)  
[0072]         {  
[0073]             .....  
[0074]         }  
[0075]     })
```

[0070] 其中BluetoothAdapter.LeScanCallback是主机端蓝牙应用程序设置的可监听扫描结果的类,这个类实现回调函数onLeScan()。

[0071] 主机端的蓝牙应用程序调用startLeScan()方法后,RMI stub向RMI注册表查询是否存在该方法。如果存在则将参数BluetoothAdapter.LeScanCallback和方法名startLeScan()打包并发送给RMI skeleton。RMI skeleton解包后,寻找对应的方法后调用并将返回值打包回传给RMI stub,返回值中有回调onLeScan()所需要的值;RMI stub解包后返回给主机端的蓝牙应用程序,包括回调LeScanCallback.onLeScan()。

[0072] 打包的API方法在传输过程中,以Java对象的形式传输。Java对象的远程传输必须序列化,即Java对象需要继承Serializable。以startLeScan()这个API方法为例,RMI skeleton将回传值以ScanCallBack对象的方式序列化并回传给RMI stub。

[0073]

```
public class ScanCallBack implements Serializable {
    Private BluetoothDevice device;
    private int rssi;
    private byte[] scanRecord;
    public ScanCallBack (BluetoothDevice device, int rssi, byte[]
scanRecord){
        .....
    }
    public BluetoothDevice getDevice() {
        .....
    }
    public int getRssi() {
    }
[0074] public byte[] getRecord () {
        .....
    }
}
```

[0075] RMI stub收到RMI skeleton回传的ScanCallBack对象后,从中提取相应的device (设备)、rssi (信号强度)、scanRecord (广播数据) 属性,调用蓝牙应用程序程序设置的LeScanCallback.onLeScan(),从而使蓝牙应用程序程序获得前述扫描结果。

[0076] 本实施例可有效地通过TCP/IP网络传输虚拟蓝牙协议栈所涉及的方法及参数,通过TCP/IP网络传输到远端蓝牙网关内的标准蓝牙协议栈,进而传递至蓝牙驱动模块、蓝牙硬件。这可使蓝牙应用程序、蓝牙设备之间工作距离从传统的几米、几十米,扩大至TCP/IP网络可达到的距离,如几十公里甚至成千上万公里,实现蓝牙应用程序对外部蓝牙设备的远程操控与数据获取。

[0077] 本实施例的蓝牙设备远程控制系统中主机端可远程通过蓝牙网关与外部蓝牙设备进行通信交互,使得主机端与外部蓝牙设备之间的数据传输距离不受限制,也可以近程直接通过主机端的蓝牙模块直接与外部蓝牙设备进行通信交互,使得蓝牙设备的应用范围更加广泛。

[0078] 实施例2

[0079] 本实施例提供一种蓝牙设备远程控制方法,蓝牙设备远程控制方法基于如实施例

1中的蓝牙设备远程控制系统实现。如图3所示,蓝牙设备远程控制方法包括:

[0080] 步骤300、蓝牙应用程序判断外部蓝牙设备是否在短距通信范围内,若是,则执行步骤301;若否,则执行步骤302;

[0081] 步骤301、蓝牙应用程序通过第二蓝牙模块与外部蓝牙设备通信。

[0082] 第二蓝牙模块包括蓝牙协议栈API(应用程序编程接口)、蓝牙驱动模块和蓝牙硬件,蓝牙调用蓝牙协议栈API实现蓝牙设备发现过程、设备连接过程、通信过程(向外部蓝牙设备发送指令、获取蓝牙设备数据)等蓝牙功能。蓝牙协议栈API则调用蓝牙驱动模块的有关接口,向蓝牙硬件发出各类通信指令,执行具体相对应的动作以与外部蓝牙设备通信。

[0083] 步骤302、蓝牙应用程序通过蓝牙网关与外部蓝牙设备通信。

[0084] 步骤302包括:

[0085] 步骤3021、第一网络模块与第二网络模块通信;

[0086] 本实施例中的第一网络模块和第二网络模块可包括使用TCP/IP协议的互联网,也可包括但不限于局域网、Wi-Fi无线网,或GPRS/3G/4G/5G移动互联网等;本实施例中,网络模块的协议采用TCP/IP网络协议。。

[0087] 步骤3022、蓝牙应用程序发送蓝牙控制命令至虚拟蓝牙协议栈,虚拟蓝牙协议栈发送蓝牙控制命令至第一网络模块;蓝牙控制命令包括蓝牙搜索、蓝牙配对、蓝牙连接和蓝牙通信中的至少一种,蓝牙控制命令包括调用方法名称和参数。

[0088] 步骤3023、第一网络模块发送蓝牙控制命令至第二网络模块;

[0089] 步骤3024、第二网络模块调用第一蓝牙模块执行蓝牙控制命令,第一蓝牙模块根据蓝牙控制命令与外部蓝牙设备通信。

[0090] 第一蓝牙模块包括蓝牙协议栈API、蓝牙驱动模块和蓝牙硬件,第二网络模块用于调用蓝牙协议栈API实现外部蓝牙设备的发现过程、设备连接过程、通信过程(向外部蓝牙设备发送指令、获取外部蓝牙设备数据)等蓝牙功能。蓝牙协议栈API则调用蓝牙驱动模块的有关接口,向蓝牙硬件发出各类通信指令,执行具体相对应的动作以与外部蓝牙设备通信。

[0091] 步骤3025、第一蓝牙模块将执行蓝牙控制命令后的结果发送至第二网络模块;

[0092] 步骤3026、第二网络模块发送结果至第一网络模块;

[0093] 步骤3027、第一网络模块通过虚拟蓝牙协议栈将结果发送至蓝牙应用程序;

[0094] 步骤3028、蓝牙应用程序接收并处理结果。

[0095] 虚拟蓝牙协议栈是对标准的蓝牙协议栈的模拟实现,从功能上看它完全等同于标准的蓝牙协议栈,基于虚拟蓝牙协议栈封装的API,也完全等同于标准的蓝牙协议栈API。不同的是,在此种情况下,虚拟蓝牙协议栈不再直接调用蓝牙驱动模块,而只是将蓝牙应用程序的蓝牙控制命令通过TCP/IP网络传输给远端的蓝牙网关,由远端的蓝牙网关执行实现标准的蓝牙协议栈API,后者则会将返回值通过TCP/IP网络回传给虚拟蓝牙协议栈,再由虚拟蓝牙协议栈返回给蓝牙应用程序。

[0096] 主机端加蓝牙网关的组合既可实现主机端与外部蓝牙设备之间的传统的近距离交互模式,也可以实现远程的交互模式,但蓝牙应用程序是通用的,当主机端不包括蓝牙模块或者超出蓝牙设备的直接通信范围时,蓝牙应用程序与远端蓝牙设备间的蓝牙控制命令的交互也可以通过虚拟蓝牙协议栈、主机端的TCP/IP网络、蓝牙网关的TCP/IP网络对蓝牙

应用程序发出的蓝牙控制命令进行透传,以传送至蓝牙网关的蓝牙模块执行蓝牙控制命令,从而保证了蓝牙应用程序的兼容性,无需对蓝牙应用程序重新定制开发。

[0097] 本实施例中使用远程过程调用(RPC)的方式实现虚拟蓝牙协议栈的API方法。将被调用的方法名称和参数经TCP/IP网络流化后透传给蓝牙网关,蓝牙网关的第二网络模块在收到被调用的方法名称和参数后,调用标准蓝牙协议栈中的API方法,并将获得的结果经第二网络模块通过第一网络模块回传给虚拟蓝牙协议栈,由虚拟蓝牙协议栈转发给蓝牙应用程序。

[0098] 远程调用可以是双向的。虚拟蓝牙协议栈调用蓝牙网关,蓝牙网关也可以调用蓝牙应用程序设置的回调函数,具体的实现是,蓝牙网关里设置一个统一的回调接口,它将参数透传给虚拟蓝牙协议栈,虚拟蓝牙协议栈再调用蓝牙应用程序设置的回调函数将对应的返回数据发送给蓝牙应用程序。

[0099] 关于虚拟蓝牙协议栈的建立,只保留供应用层调用的基于标准蓝牙协议栈的全部API结构,即API命名、参数及返回值均与标准蓝牙协议栈的API保持一致。虚拟蓝牙协议栈的API不调用蓝牙驱动模块和蓝牙硬件,不与蓝牙驱动模块直接交互,而会直接将调用虚拟蓝牙协议栈的各个API方法经过打包封装,并通过TCP/IP网络传递给远端的蓝牙网关执行。

[0100] 蓝牙网关端通过TCP/IP网络模块收到主机端发出的包含有蓝牙控制命令的消息后调用蓝牙网关中的标准蓝牙协议栈执行相应的蓝牙控制命令。

[0101] RPC已非常成熟,已出现了大量的成熟RPC产品或实现。如著名的RPC实现框架有Java(一种面向对象编程语言)RMI(远程方法调用)、Alibaba Dubbo(阿里巴巴开源出来的一个分布式服务框架)、Apache Thrift(Facebook开源的跨语言的RPC通信框架)、Google GRPC(Google开源的一个高性能、跨语言的RPC框架)、Twitter Finagle(Twitter基于Netty开发的支持容错的、协议无关的RPC框架)等。这些产品或框架性能优越,在互联网中应用十分广泛。

[0102] 至于如何实现远程接口调用,作为实施例,可以利用Java的RMI框架。如图2所示,其中RMI客户端从RMI注册簿中查询RMI服务端已注册的可用远程的方法,然后调用RMI stub上的方法,RMI stub将调用的方法的名称、参数打包后通过TCP/IP网络传递给RMI skeleton,RMI skeleton解包后得到被调用的方法和参数,然后调用RMI服务端中的相应的调用的方法。RMI客户端与RMI服务端通过TCP/IP网络进行连接。

[0103] 具体到本实施例的蓝牙远程调用场景,虚拟蓝牙协议栈相当于RMI stub,第二网络模块相当于RMI skeleton,主机端的蓝牙应用程序相当于RMI客户端,第一蓝牙模块相当于RMI服务端。具体工作过程如下:

[0104] 主机端的蓝牙应用程序调用虚拟蓝牙协议栈中的API方法;RMI stub(即虚拟蓝牙协议栈)将API方法(变量和方法名)打包,然后通过TCP/IP网络(第一网络模块)发送给蓝牙网关的RMI skeleton(即第二网络模块);RMI skeleton将RMI stub发送来的API方法解包,解析出被调用的方法、参数以及该方法所在对象;RMI skeleton调用第一蓝牙模块中对应的方法,并将执行结果返回给RMI skeleton,RMI skeleton将执行结果打包并发送给RMI stub,RMI stub解包得到执行结果并将执行结果返回给蓝牙应用程序。

[0105] 以蓝牙4.0Android平台主机端执行蓝牙扫描为例。

[0106] 主机端的蓝牙应用程序调用了BluetoothAdapter的API进行设备扫描的操作:

[0107]

```
startLeScan(new BluetoothAdapter.LeScanCallback() {  
    @Override  
    public void onLeScan(final BluetoothDevice device, final int rssi, byte[]  
scanRecord)  
    {  
        .....  
    }  
})
```

[0108] 其中BluetoothAdapter.LeScanCallback是主机端蓝牙应用程序设置的可监听扫描结果的类,这个类实现回调函数onLeScan()。

[0109] 主机端的蓝牙应用程序调用startLeScan()方法后,RMI stub向RMI注册簿查询是否存在该方法。如果存在则将参数BluetoothAdapter.LeScanCallback和方法名startLeScan()打包并发送给RMI skeleton。RMI skeleton解包后,寻找对应的方法后调用并将返回值打包回传给RMI stub,返回值中有回调onLeScan()所需要的值;RMI stub解包后返回给主机端的蓝牙应用程序,包括回调LeScanCallback.onLeScan()。

[0110] 打包的API方法在传输过程中,以Java对象的形式传输。Java对象的远程传输必须序列化,即Java对象需要继承Serializable。以startLeScan()这个API方法为例,RMI skeleton将回传值以ScanCallBack对象的方式序列化并回传给RMI stub。

[0111] `public class ScanCallBack implements Serializable {`

[0112]

```
    Private BluetoothDevice device;

    private int rssi;

    private byte[] scanRecord;

    public ScanCallback (BluetoothDevice device, int rssi, byte[]
scanRecord){
        .....
    }

    public BluetoothDevice getDevice() {
        .....
    }

    public int  getRssi() {
    }

    public byte[] getRecord () {
        .....
    }
}
```

[0113] RMI stub收到RMI skeleton回传的ScanCallback对象后,从中提取相应的device (设备)、rssi (信号强度)、scanRecord (广播数据) 属性,调用蓝牙应用程序程序设置的LeScanCallback.onLeScan(),从而使蓝牙应用程序程序获得前述扫描结果。

[0114] 本实施例可有效地通过TCP/IP网络传输虚拟蓝牙协议栈所涉及的方法及参数,通过TCP/IP网络传输到远端蓝牙网关内的标准蓝牙协议栈,进而传递至蓝牙驱动模块、蓝牙硬件。这可使蓝牙应用程序、蓝牙设备之间工作距离从传统的几米、几十米,扩大至TCP/IP网络可达到的距离,如几十公里甚至成千上万公里,实现蓝牙应用程序对外部蓝牙设备的远程操控与数据获取。

[0115] 本实施例的蓝牙设备远程控制方法中主机端可远程通过蓝牙网关与外部蓝牙设备进行通信交互,使得主机端与外部蓝牙设备之间的数据传输距离不受限制,也可以近程直接通过主机端的蓝牙模块直接与外部蓝牙设备之间进行通信交互,使得蓝牙设备的应用范围更加广泛。

[0116] 虽然以上描述了本发明的具体实施方式,但是本领域的技术人员应当理解,这仅是举例说明,本发明的保护范围是由所附权利要求书限定的。本领域的技术人员在不背离本发明的原理和实质的前提下,可以对这些实施方式做出多种变更或修改,但这些变更和

修改均落入本发明的保护范围。

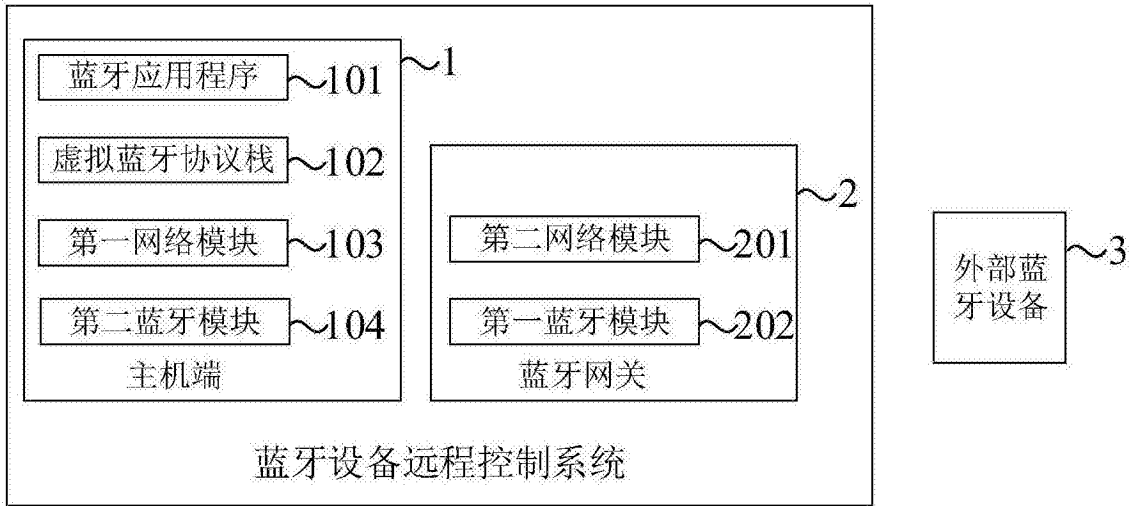


图1

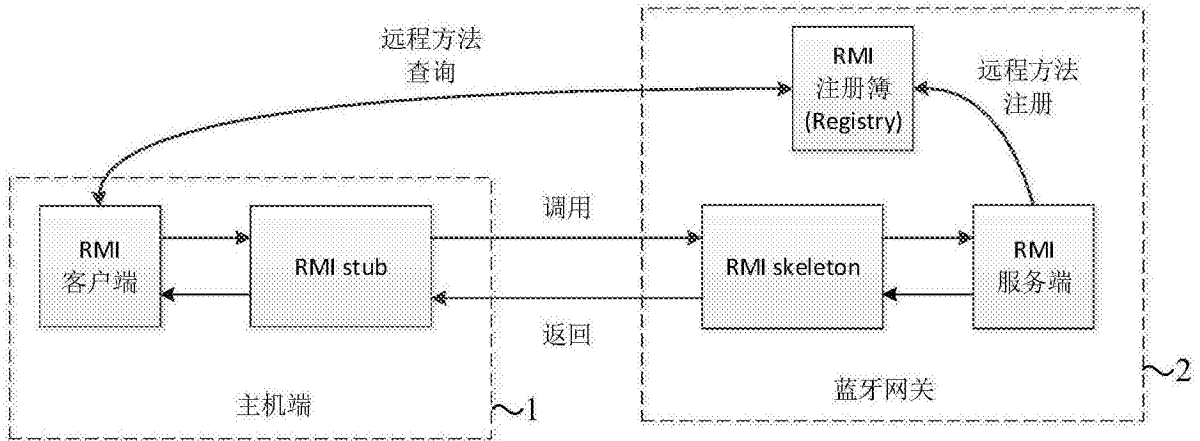


图2

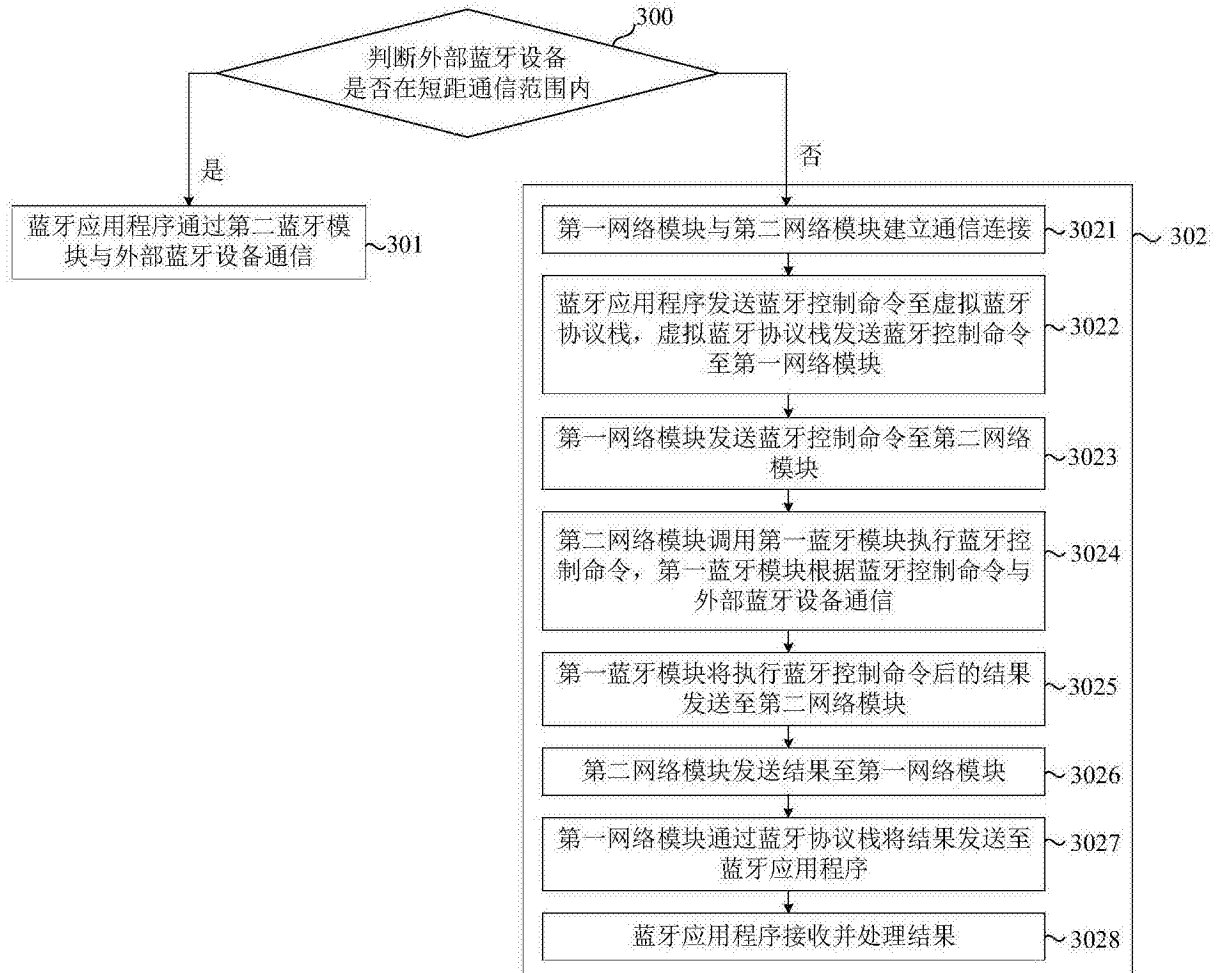


图3