



(12)发明专利申请

(10)申请公布号 CN 106130773 A
(43)申请公布日 2016. 11. 16

(21)申请号 201610531717.8

(22)申请日 2016.07.07

(71)申请人 苏州大学

地址 215123 江苏省苏州市工业园区仁爱路199号

(72)发明人 王宜怀 陈建明 陈瑞杰 陆伟国

(74)专利代理机构 南京利丰知识产权代理事务所(特殊普通合伙) 32256

代理人 王锋

(51) Int. Cl.

H04L 12/24(2006.01)

H04L 29/08(2006.01)

G06F 9/445(2006.01)

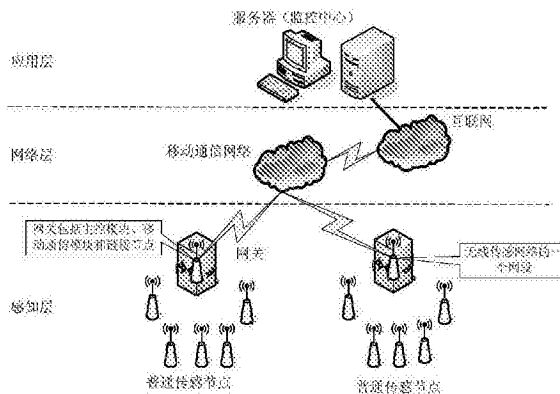
权利要求书2页 说明书14页 附图4页

(54)发明名称

一种基于MQX_Lite的WSN节点远程代码更新系统

(57)摘要

本发明在物联网三层体系结构的基础上设计开发一套可靠、高效、通用的远程代码更新解决方案。本发明具体涉及物联网感知层网关及无线传感节点的设计,以及应用层服务器相关设计。感知层无线传感节点远程更新的设计包括搭建无线传感节点硬件平台、驱动构件及监控程序的设计。感知层网关的远程更新的设计包括整体硬件设计、驱动构件及相应程序的设计。应用层服务器软件设计包括管理软件、数据库和通信软件的设计。



1. 一种基于MQX_Lite的WSN节点远程代码更新系统,其特征在于,包括:感知层、网络层以及应用层;所述感知层包括:组成无线传感网络的WSN节点、用于连接无线传感网络、移动通信网络和因特网的网关;所述网络层包括移动通信网络和因特网;所述应用层包括服务器;其中,

感知层的WSN节点基于无线代码更新协议(WCUCP)实现代码更新通信,并通过监控程序实现具体代码更新,可同时更新全部或个别节点的程序;

感知层的所述网关包括:主控模块、移动通信模块和链接节点;所述网关通过移动通信模块连接服务器进行双向通信,链接节点与普通传感节点间通过无线传感网络进行双向通信;所述网关与服务器之间基于远程代码更新协议RCUCP实现远程代码更新的双向通信;

应用层的服务器包括通信软件模块和管理软件模块;所述通信软件模块负责数据通信;所述管理软件模块负责对数据解析、处理。

2. 根据权利要求1所述的WSN节点远程代码更新系统,其特征在于,所述WSN节点正常状态下运行用户程序,当接收到更新命令时,通过软件复位转入监控程序运行;监控程序配合服务器运行远程代码更新软件,实现远程更新。

3. 根据权利要求2所述的WSN节点远程代码更新系统,其特征在于,WSN节点间的代码更新过程包括:根节点通过USB转串口的线与PC机相连接;PC机代码更新软件提取出待更新程序机器码中的有效数据,分包组帧后通过串口发送给根节点,根节点转发给WSN节点;同时也接收WSN节点返回的信息。

4. 根据权利要求3所述的WSN节点远程代码更新系统,其特征在于,所述WSN节点设置了辅助标志位和超时机制;辅助标志位用于辅助监控程序与用户程序间的跳转;超时机制用于判断WSN节点是否接收到代码更新相关数据帧,当计时器超过更新等待时间阈值UPDATE_TIME都没有收到相关数据帧,则自动转入到用户程序中运行,否则执行正常更新流程。

5. 根据权利要求4所述的WSN节点远程代码更新系统,其特征在于,PC机代码更新软件通过根节点将更新代码分组发送给WSN节点,WSN节点无需收到一帧数据回发一帧确认(ACK)帧,只要最后回发一帧包含丢帧信息的数据,PC机代码更新软件收到该帧解析出丢失的帧数据并重发。

6. 根据权利要求5所述的WSN节点远程代码更新系统,其特征在于,WSN节点与根节点采用CSMA/CA(Carrier Sense Multiple Access with Collision Avoidance,载波监听多址接入-碰撞避免)算法来避免数据接收时的冲突;在调用发送函数之前,先延时一个随机($2^{\wedge}BE$ 范围内,开始时 $BE=1$)的时间,然后判断信道是否空闲,如果空闲发送,否则在1到 $2^{\wedge}BE$ 间再延时一个随机时间,此时 $BE=\min(BE+1, \max BE)$,如此循环,最多可以循环($NB < \max NB$)次。

7. 根据权利要求6所述的WSN节点远程代码更新系统,其特征在于,所述服务器的管理软件模块负责远程更新系统中网关及WSN节点状态的监测和管理,以及将待更新的机器码传到服务器指定位置,并选择要更新节点的网络号和节点号,发送代码更新相关命令。

8. 根据权利要求7所述的WSN节点远程代码更新系统,其特征在于,所述服务器采用公网固定IP地址,各网关与服务器的通信软件模块正常数据通信前,都需作为客户端主动创建与服务器的点对点连接。

9. 根据权利要求8所述的WSN节点远程代码更新系统,其特征在于,服务器的通信软件通过维护一张哈希表,记录当前处于连接状态的网关和其对应的Socket句柄;每当与网关

建立连接后就在该表中添加一条记录；当网关断开连接后就删除表中对应的记录。

10. 根据权利要求9所述的WSN节点远程代码更新系统,其特征在于,每个网关设置了一个网络号,该网关控制同一网络号下的WSN节点;每个WSN节点都有唯一的网络号和网内地址,同时还需要设置部分特殊地址来实现广播或点对点数据传输。

一种基于MQX_Lite的WSN节点远程代码更新系统

技术领域

[0001] 本发明涉及物联网领域,具体涉及一种基于MQX_Lite的WSN节点远程代码更新系统。

背景技术

[0002] 物联网(Internet of Things)是指通过各种信息传感设备,实时采集任何需要监控、连接、互动的物体或过程等各种需要的信息,与互联网结合形成的一个巨大网络。其目的是实现物与物、物与人,所有的物品与网络的连接,方便识别、管理和控制。物联网从宏观上来看包含三个层次,分别是感知层,用来感知世界;网络层,用来传输数据;应用层,用来处理数据。感知层包括电子标签、摄像头、读卡器等综合识别与感知的数据采集设备。网络层包括2G、3G、4G等移动通信技术和有线网络。应用层主要涉及物联网包括智能医疗、环境监测、智能家居、智能物流、智能城市等应用。感知层的作用是感知和采集信息,从仿生学来看,感知层为“感觉器官”,可以感知自然界的各种信息。通过传感器、摄像头、红外感应器等数据采集设备获取外部物理世界的的数据,采用有线或无线等短距离传输技术传递数据,配合传感器网关和网络层通信。感知层所需要的关键技术包括传感器技术、短距离有线(RS485、CAN总线、嵌入式以太网等)和无线(蓝牙、红外、ZigBee、Wi-Fi、Sub-1G等)通信技术、RFID、传感器网络技术等。网络层作为纽带连接着感知层和应用层,它由各种私有网络、互联网、有线和无线通信网等组成,相当于人的神经中枢系统。它的作用是将感知层所获得的数据安全可靠地传输到应用层,这些数据可以通过移动通信网(无线移动通信网络、卫星通信网络、公众电话网等)、国际互联网、企业内部网、各类专网、局域网(以太网、无线局域网WLAN、Bluetooth等)等网络传输。网络层所需要的关键技术包括移动通信、接入网和承载网的技术等。应用层位于物联网三层结构中的最顶层,主要作用是进行数据处理计算、处理和知识挖掘,从而实现对物理世界的实时控制、精确管理和科学决策。该层按形态直观地划分为两个子层。一个是应用程序层,进行数据处理,它涵盖了国民经济和社会的每一领域,包括电力、医疗、银行、交通、环保、物流、工业、农业、城市管理、家居生活等,另一个是终端设备层,提供人机界面,例如个人计算机、PDA、智能手机等。

[0003] 无线传感网络是物联网感知层使用最广泛的技术,也是物联网发展的重点。无线传感网络(Wireless Sensor Networks,WSN)是由分布在不同区域的自组织的节点设备组成的无线网络。这些节点设备通过自身附带或连接的传感器能够实时的监测周围的物理或者环境状况,感知和采集覆盖区域中监测对象的信息,比如温度,湿度,光照度,压力等,并将此信息融合、汇聚至无线网络的中心管理者。每个无线传感节点均由各自的感知单元、通信单元和微控制单元等构成。WSN中的节点通过无线方式通信,因此网络设置灵活,设备位置可以随时更改,还可以跟互联网进行有线或无线方式的连接。除了具备传统无线移动网络的动态性、扩展性和间断性等共同特征外,WSN还具有低成本、高集成、低功耗、自组织等鲜明的特点。WSN简单的组网方式通常包括传感器节点(SensorNode,简称节点)、汇聚节点(网关)和任务管理节点。大量节点随机密集部署在监测区域中,通过自组织的方式构成网

络;节点在对所探测信息进行初步处理后,以多跳中继方式将数据传至网关。然后经互联网、移动通信网络、卫星等途径到达最终用户所在的管理节点。

[0004] 微机电系统(Micro-Electro-Mechanism System, MEMS)、片上系统(System on Chip, SoC)、无线通信和低功耗嵌入式技术的飞速发展促进了无线传感网络(Wireless Sensor Networks, WSN)的快速发展。在网络的运行过程中,基于修复漏洞、改变网络功能、调优系统参数等原因,经常需要对传感器节点运行的程序进行版本的更新。随着WSN的发展,WSN节点数量变得越来越多,更新和维护成为了重要问题,传统的人工现场更新方案无法再满足节点数量众多、分布复杂的新需求,因此通过无线方式实现代码更新变的格外重要。目前,国内外对基于物联网三层结构的远程代码更新关注度不够,对远程代码更新系统整体的设计与研究还不多;对WSN节点更新的局部相关技术研究虽较多,例如无线传感网络的重编程技术,但对重编程的研究集中在数据协议和代码分发机制上,着重关注于重编程编码、算法、可靠性和传输协议安全性的研究,对重编程的通用性、可移植性研究和异构无线传感网络的重编程研究较少,过于依赖于具体的执行环境。

发明内容

[0005] 为了克服现有技术的不足,本发明旨在物联网三层体系结构的基础上设计开发一套可靠、高效、通用的远程代码更新解决方案。本发明具体涉及物联网感知层网关及无线传感节点的设计,以及应用层服务器相关设计。感知层无线传感节点远程更新的设计包括搭建无线传感节点硬件平台、驱动构件及监控程序的设计。感知层网关的远程更新的设计包括整体硬件设计、驱动构件及相应程序的设计。应用层服务器软件设计包括管理软件、数据库和通信软件的设计。

[0006] 本发明所述的基于MQX_Lite的WSN节点远程代码更新系统,其特征在于,包括:感知层、网络层以及应用层;所述感知层包括:组成无线传感网络的WSN节点、用于连接无线传感网络、移动通信网络和因特网的网关;所述网络层包括移动通信网络和因特网;所述应用层包括服务器;其中,

[0007] 感知层的WSN节点基于无线代码更新协议(WCUCP)实现代码更新通信,并通过监控程序实现具体代码更新,可同时更新全部或个别节点的程序;

[0008] 感知层的所述网关包括:主控模块、移动通信模块和链接节点;所述网关通过移动通信模块连接服务器进行双向通信,链接节点与普通传感节点间通过无线传感网络进行双向通信;所述网关与服务器之间基于远程代码更新协议RCUCP实现远程代码更新的双向通信;

[0009] 应用层的服务器包括通信软件模块和管理软件模块;所述通信软件模块负责数据通信;所述管理软件模块负责对数据解析、处理。

[0010] 优选的是,所述WSN节点正常状态下运行用户程序,当接收到更新命令时,通过软件复位转入监控程序运行;监控程序配合服务器运行远程代码更新软件,实现远程更新。

[0011] 优选的是,WSN节点间的代码更新过程包括:根节点通过USB转串口的线与PC机相连接;PC机代码更新软件提取出待更新程序机器码中的有效数据,分包组帧后通过串口发送给根节点,根节点转发给WSN节点;同时也接收WSN节点返回的信息。

[0012] 优选的是,所述WSN节点设置了辅助标志位和超时机制;辅助标志位用于辅助监控

程序与用户程序间的跳转;超时机制用于判断WSN节点是否接收到代码更新相关数据帧,当计时器超过更新等待时间阈值UPDATE_TIME都没有收到相关数据帧,则自动转入到用户程序中运行,否则执行正常更新流程。

[0013] 优选的是,PC机代码更新软件通过根节点将更新代码分组发送给WSN节点,WSN节点无需收到一帧数据回发一帧确认(ACK)帧,只要最后回发一帧包含丢帧信息的数据,PC机代码更新软件收到该帧解析出丢失的帧数据并重发。

[0014] 优选的是,WSN节点与根节点采用CSMA/CA(Carrier Sense Multiple Access with Collision Avoidance,载波监听多址接入-碰撞避免)算法来避免数据接收时的冲突;在调用发送函数之前,先延时一个随机(2^{BE} 范围内,开始时 $\text{BE}=1$)的时间,然后判断信道是否空闲,如果空闲发送,否则在 1 到 2^{BE} 间再延时一个随机时间,此时 $\text{BE}=\min(\text{BE}+1, \max\text{BE})$,如此循环,最多可以循环($\text{NB}<\max\text{NB}$)次。

[0015] 优选的是,所述服务器的管理软件模块负责远程更新系统中网关及WSN节点状态的监测和管理,以及将待更新的机器码传到服务器指定位置,并选择要更新节点的网络号和节点号,发送代码更新相关命令。

[0016] 优选的是,所述服务器采用公网固定IP地址,各网关与服务器的通信软件模块正常数据通信前,都需作为客户端主动创建与服务器的点对点连接。

[0017] 优选的是,服务器的通信软件通过维护一张哈希表,记录当前处于连接状态的网关和其对应的Socket句柄;每当与网关建立连接后就在该表中添加一条记录;当网关断开连接后就删除表中对应的记录。

[0018] 优选的是,每个网关设置了一个网络号,该网关控制同一网络号下的WSN节点;每个WSN节点都有唯一的网络号和网内地址,同时还需要设置部分特殊地址来实现广播或点对点数据传输。

[0019] 可见,本发明的传感节点的远程代码更新可以解决物联网系统维护难的问题,通过已有的网络结构实现物联网感知层的传感节点的远程更新,减少人工现场对传感节点的维护,真正意义上的实现了代码的无线更新。远程代码更新中会实现服务器与网关、网关与传感节点的交互,间接地增加了节点和服务器的交互,可以实时监控节点的运行状态并进行相应的控制,提高了自动化程度。这样能有效地降低物联网系统更新和维护成本,使其更加可靠、稳定、灵活、可配置,适用于复杂多变的环境。远程代码更新的实现也为以后开发类似物联网系统提供了快捷的调试和测试手段,在系统开发和测试阶段都可以使用远程更新功能来更新代码,以后开发类似的系统时无需人工逐个的通过写入器来更新代码。当系统出现Bug需要修改程序时,也无需逐个更换程序。

附图说明

[0020] 为了更清楚地说明本发明实施例的技术方案,下面将对实施例中所需要使用的附图作简单地介绍,显而易见地,下面描述中的附图仅仅是本发明的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动的前提下,还可以根据这些附图获得其他的附图。

[0021] 图1是本发明所述远程代码更新系统的整体结构示意图;

[0022] 图2是本发明所述远程代码更新系统的网关硬件结构示意图;

[0023] 图3是本发明所述远程代码更新系统的服务器端结构示意图;

- [0024] 图4是本发明所述无线传感网络节点更新的系统结构示意图；
- [0025] 图5是本发明所述PC机软件解析.HEX文件并提取出有效数据的流程示意图；
- [0026] 图6是本发明所述根节点与PC机软件、WSN节点间数据通信流程；
- [0027] 图7是本发明所述代码更新的流程示意图；
- [0028] 图8是本发明所述传感节点程序的详细执行流程图；
- [0029] 图9是本发明所述远程更新系统的结构示意图；
- [0030] 图10是本发明所述服务器端通信软件和网关的Socket通信过程示意图；
- [0031] 图11是本发明所述通信软件与数据库及网关间的通信过程示意图。

具体实施方式

[0032] 下面结合附图详细阐述本发明所优选的具体实施方式。

[0033] 本发明基于物联网系统的远程代码更新软件设计主要包括以下三个部分：(1)感知层普通无线传感节点：感知层的传感器节点数量众多，本发明参考WSN重编程技术设计了无线代码更新的通信协议(Wireless Code Update Communication Protocol:WCUCP)和防冲突机制，WCUCP简单、方便、有效兼顾可靠性，具有很好的通用性和可移植性，防冲突机制可避免数据接收时的冲突。同时，为了配合代码更新，本发明还设计了无线传感节点的监控程序，可实现把机器码数据接收存放到临时区域，验证正确后拷贝到用户程序区，最后跳转到用户程序执行。(2)关于感知层网关的设计，网关是服务器和传感节点的联系纽带，包括三个实体部分：主控模块、移动通信模块和链接节点。网关接收服务器端发过来的程序镜像，并将其发送给普通无线传感节点实现代码更新。本发明选择了GPRS移动通信技术，设计了统一的接口将通信的细节隐藏起来。同时，本发明为网关与服务器的通信设计了远程代码更新协议(Remote Code Update Communication Protocol:RCUCP)，采用合理编址机制，为实现并行更新的方案提供了基础，大大提高了节点远程代码更新的效率。(3)应用层服务器软件设计。为使更新软件具有更好的灵活性，将负责通信的软件和负责对更新数据分析、处理、解析等操作的管理软件分离，两者之间通过实时数据库进行通信。通信软件采用TCP(Transmission Control Protocol,传输控制协议)通信，增加了数据传输的稳定性和可靠性。管理软件为用户提供远程更新的界面，主要包括更新网段和节点的选择。服务器端软件需要对更新的程序镜像进行解析、组帧、校验等操作，并和网关实时交互。

[0034] 本发明基于MQX_Lite实现WSN节点的远程代码更新。MQX是Precise Software Technologies公司1989年开发的一款嵌入式实时操作系统。MQX的主要特点是实时性高、内核精简、内核免费且有技术支持、开发工具成熟、外设驱动丰富。轻量级MQX实时操作系统(MQX_Lite RTOS)是标准MQX实时操作系统的轻量级版本，其内核精简，只包含标准MQX内核的部分必要组件及某些组件的轻量级版本，是标准MQX的一个真子集。它裁剪了标准MQX中的诸如I/O子系统、时间片轮转调度、显式任务队列调度、看门狗及命名服务等功能，而对时间、信号量、消息和定时器组件，只保留了它们的轻量级版本。主要针对资源有限(FLASH或RAM)的MCU，支持应用以低于4 KB RAM的空间运行。主要包含任务管理与调度、任务间通信和同步、存储管理、时间管理和中断处理、外设管理等核心功能。

[0035] 本发明基于物联网三层体系结构设计了远程代码更新系统，系统结构如图1所示。其中感知层主要包括组成无线传感网络的普通传感节点(可简称为WSN节点)；用于连接无

线传感网络、移动通信网络和因特网的网关。网络层主要是网络通信基础设施,包括移动通信网络和因特网。应用层主要涉及各种物联网的应用,例如环境监测、智能交通、远程医疗、智能城市管理和工业监控等。感知层的WSN节点选择基于IEEE 802.15.4标准的硬件,本发明为远程代码更新系统中无线传感网络中的代码更新设计了无线代码更新协议WCUCP,该协议简单、可靠,可实现轻量洪泛广播通信,不依赖于具体的操作系统和网络结构,具有较高的独立性,具有很好的通用性和可移植性。通过监控程序的设计来实现具体的代码更新,可同时更新全部或个别节点的程序。感知层的网关选择GPRS移动通信技术将感知层采集的数据传输到服务器。网关主要包括三个实体部分:主控模块、移动通信模块和链接节点。网关通过移动通信模块连接着服务器进行双向通信,链接节点与普通传感节点间通过无线传感网络进行双向通信。本发明为远程代码更新系统中网关与服务器间设计了远程代码更新协议RCUCP,包括编制机制和数据帧格式设计,为实现并行更新的方案提供了基础,大大提高了节点远程代码更新的效率。应用层选择面向连接的TCP作为通信手段,增加了数据传输的可靠性和稳定性。为了增加软件的灵活性和可扩展性,本发明将负责通信的软件和负责对数据分析、处理等操作的管理软件分离。通信软件需要时刻运行来和网关保持持续通信,而管理软件可由使用者自主打开或关闭。因此,将通信软件和管理软件分离设计,两者通过数据库进行交互,该方案更具有灵活性,易于开发。

[0036] WSN节点需选择基于IEEE 802.15.4标准的芯片,为使设计更加简单灵活,本文选择了现有的无线通信模块,并根据实际需要设计了扩展板模块。远程代码更新系统中,正常状态下传感节点运行用户程序,当接收到更新命令时,通过软件复位转入监控程序运行。监控程序通过RCUCP、WCUCP和代码分发机制配合服务器远程代码更新软件实现远程更新。为协调监控程序与用户程序的执行,防止传感节点异常复位转到监控程序中运行,监控程序中设计了超时机制,用于判断是否接收到远程更新相关数据帧,当计时器超过更新等待时间阈值都没收到,则自动转入到用户程序中运行。传感节点的软件设计主要包括四个部分:(1)存储器空间和机器码文件的分析,链接文件的分析与设计。存储空间的大小及分布取决于硬件的芯片选型,机器码文件的格式取决于开发工具支持的机器码类型及用户的选择。链接文件设定了中断向量表、代码内容等所处的存储空间。监控程序所处的FLASH空间和用户程序不能重合,因此需通过修改用户程序的链接文件重新定位其存储空间。(2)用于节点无线更新的WCUCP网络协议、丢帧重传机制、防冲突机制的设计和实现。其中,WCUCP简单、可靠,可实现轻量洪泛广播通信,不依赖于具体的操作系统和网络结构,具有较高的独立性,具有很好的通用性和可移植性。丢帧重传机制和防冲突机制保证了数据帧在不可靠无线环境中传输的正确性和可靠性。(3)内存分配的设计。为实现更新代码的可靠擦写机制,将FLASH空间分为四个部分:监控程序区、用户程序区、更新用户程序区和预留区。待更新的代码先存放在更新用户程序区,待接收完毕且校验正确后,拷贝覆盖到用户程序区。(4)监控程序的设计。监控程序的主要功能是与网关进行通信,通过WCUCP实现代码有效准确的传输,实现代码更新。监控程序设计主要包括跳转函数设计、跳转函数设计、通用性设计和程序保护。

[0037] 网关是服务器和该网段中WSN节点的联系纽带,整个硬件采用分块的形式进行设计,包括主控模块、移动通信模块和链接(Link)节点。它为WSN节点的更新提供了数据传输服务,是远程代码更新系统的重要部分。图2为网关的硬件结构示意图,其中链接节点的硬

件部分和普通传感节点的相同。主控模块通过移动通信模块与服务器进行双向通信。主控模块将要发送的数据通过移动通信模块发送给服务器,移动通信模块将从服务器端接收到的数据进行解析、处理后传给主控模块。主控模块通过Link节点与无线传感节点进行双向通信。Link节点负责连接和控制传感节点网络,具有传感网络中的路由节点的功能,负责发送初始数据,接收所需的返回信息。网关的软件设计包括以下部分:(1)主控模块的驱动构件和程序的设计。(2)移动通信模块的驱动构件和程序的设计。(3)链接节点的程序设计。(4)网关与服务器间通信协议RCUCP的设计。

[0038] 为了使服务器端软件更具有灵活性和扩展性,将其设计为两部分,分别为负责数据通信的通信软件模块和对数据解析、处理等操作的管理软件模块,两者之间通过实时数据库进行通信,如图3所示。将服务器端软件设计为两个的优点如下:(1)若将服务器端软件设计为单个软件,则对数据的解析和处理等操作只能在服务器上进行,具有局限性。并且该软件不能关闭,否则网关无法连接到服务器进行数据通信,系统无法正常运行。(2)将软件的通信和管理分开设计,降低了软件开发的难度,方便了软件的扩展和调试。管理软件将远程代码更新相关命令按照协议构建成各种数据帧,并将其放入数据库。除此之外,待更新的机器码是由管理软件模块直接上传到服务器的指定文件夹中,通信软件模块接收到更新命令后从服务器指定文件夹中读取解析机器码并分包发送出去,这样可以减少机器码存入数据库的操作,简单省时。管理软件可以随意开关,可以实现为客户端或Web形式,且无需运行在服务器上。通信软件获取数据库中未处理的命令帧,发送给对应的网关;同时接收网关上传的返回数据,存入数据库中。通信软件使用面向连接的TCP通信方式,数据传输稳定性高,需要时刻运行在服务器上保持对相应端口的侦听。通信软件实现形式为控制台或Service,不能被随意关闭。

[0039] 远程代码更新系统的设计基于物联网三层结构体系,涉及到服务器与网关、网关与传感节点的通信。下面对系统进行分解研究,首先研究WSN节点间的代码更新,主要验证网络协议、丢帧重传机制及监控程序等设计的可靠性。

[0040] 为实现无线传感网络节点的更新,本发明设计了一套结合串口和无线射频等技术的无线代码更新系统。该系统可通过PC机上位机软件选择相应WSN节点进行高效、可靠的程序更新。系统结构如图4所示。其中,根节点通过USB转串口的线与PC机相连接,USB端连接PC机,串口端连接根节点。PC机代码更新软件是用户对WSN节点进行控制的接口,主要功能为提取出待更新程序机器码中的有效数据,分包组帧后通过串口发送给根节点,根节点转发给WSN节点;同时也接收WSN节点返回的信息,例如数据包丢失、更新完成等。

[0041] 目前,单片机的机器码形式主要有BIN、.S19、.ELF和.HEX等,本发明使用Kinetis Design Studio开发工具进行微控制器的程序开发,程序编译后主要产生.ELF和.HEX两种形式的机器码,本发明选择其中的.HEX形式进行研究,下面简单介绍.HEX文件。.HEX(Intel HEX)文件是由一行行符合Intel HEX文件格式的文本所构成的ASCII文本文件。.HEX文件以行为单位,每一行是一个HEX记录,由十六进制数的机器码或数据常量组成。.HEX文件经常被用于将程序或数据传输存储到ROM、EPROM,大多数编程器和模拟器使用.HEX文件。一个.HEX文件可以包含任意多的十六进制记录,每行HEX记录除了开始标记“:”(冒号)外有5个部分组成。各部分含义见下表。

[0042] .HEX记录格式

[0043]

数据段	数据长度	地址	类型	数据段	校验和
长度	1 byte	2 byte	1 byte	n byte	1 byte
说明	HEX 记录数据部分的长度	数据类型记录, 该字段有效; 非数据类型记录, 该字段为 0000	00~05	取决于记录类型	开始标记之后字段的的所有字节之和的补码。

[0044] 其中,地址为该行 .HEX 记录的数据部分在 FLASH 中的存放首地址。 .HEX 记录共有 6 种类型 00~05,00 表示数据记录,01 表示文件结束记录,02 表示扩展段地址记录,03 表示开始段记录,04 表示扩展线性地址记录,05 表示链接开始地址记录。校验和用于检验 .HEX 记录的正确性,公式为:校验和 = 0xFF - (数据长度 + 偏移量 + 类型 + 数据段) + 0x01。

[0045] PC 机软件解析 .HEX 文件并提取出有效数据的流程如图 5 所示,提取 .HEX 文件内容,首先解析出程序的起始地址(即第一行数据记录的地址)赋值给变量 FLASH_ADDRESS 中,所有类型为 00 记录的有效数据,数据内容存储在 Image.HEXData 数组中,并在相同下标的 Image.Flag[] 数组中标记其是否为有效数据(1:有效;0:无效)。此外,用户代码中不可避免的会出现程序跳转,.HEX 文件中的地址会出现跳变,而被跳过的地址中存储的内容是无意义的,故给其赋值 0xff,但在代码更新时仍将其作为有效数据进行传输以简化传输过程。

[0046] 根节点是 PC 机与 WSN 节点间通信的桥梁,主要实现数据转发功能。根节点程序的设计基于 CMSIS(Cortex Microcontroller Software Interface Standard:ARM Cortex 微控制器软件接口标准),增加了该程序的可重用性和可移植性。本文中对根节点的设计加入了 MQX_Lite RTOS,按功能需求设计了五个任务:主任务(task_main)、小灯任务(task_light)、串口接收任务(task_uart0_re)、无线接收任务(task_rf_recv)、无线发送任务(task_rf_send),其中,主任务的优先级最高,是 MQX_Lite 的自启动任务,小灯任务的优先级较低,其他任务的优先级一样。程序中使用轻量级事件实现各任务间的同步,轻量级消息队列实现任务间的数据传输。

[0047] PC 机向根节点发送数据前会进行握手确保链路导通,握手过程:根节点串口接收到 PC 机更新软件发送过来的握手命令,进入串口的中断服务例程接收数据,对数据组帧放入轻量级消息队列,task_uart0_re 中检测到轻量级消息队列中有数据则取出判断,若是握手命令,通过串口向 PC 机软件回发确认命令,确保根节点跟 PC 机可正常通信。

[0048] 确保根节点与 PC 机软件链路连通后,就可实现系统的通信。PC 机软件通过串口向根节点发送数据,根节点的串口触发中断,执行串口 0 的中断服务例程 isr_uart0_re,组帧并将接收到的数据放入消息队列 uart_rev_queue 中。task_uart0_re 任务等待消息队列 uart_rev_queue 中消息,等到后解帧并根据操作命令将数据放入消息队列 rf_send_queue 中。task_rf_send 任务中等待消息队列 uart_rev_queue 中消息,等到调用 RFSendDataByCSMACA 将数据通过 Sub-1G 技术发给 WSN 节点。WSN 节点收到数据处理后,通过 Sub-1G 技术向根节点回发数据,触发根节点的 RF 中断,中断服务例程中进行组帧,成功后置 RF 接收事件位 EVENT_RF_RECV,task_rf_recv 任务一直等待该事件位,等到后将数据通过串口转发给 PC 机。根节点与 PC 机软件、WSN 节点间数据通信流程如图 6 所示。根节点的程序设计具有较强的通用性和扩展性,适用于 PC 软件与 WSN 节点通信的多种应用。

[0049] WSN 节点的程序模块包括以下部分:(1)存储映像解析及划分:进行更新代码首先需要了解 KW01 芯片的存储空间结构并对其进行合理划分。WSN 节点芯片具有 128KB(128*

1024字节)的FLASH程序空间,地址范围为0x0000_0000~0x0001_FFFF。本发明将芯片的FLASH区划分为四个部分:监控程序区(28KB)、用户程序区(45KB)、更新程序区(45KB)和预留区(10KB)。监控程序区存放监控程序代码及其中断向量表,其中监控程序不执行正常的用户程序的功能,而只是通过无线射频通信方式接收程序或数据,执行对用户程序的代码更新。用户程序区存放用户代码及其中断向量表,更新程序区用来临时存放接收到的待更新的程序,预留区用来存放一些标志位。FLASH区重新划分后,用户程序区的起始位置不是0x0000_0000地址,因此需要在用户程序的工程链接文件中对地址进行重新设置并对中断向量表重定位。(2)链接文件内存设计:链接文件提供了在链接过程中所需要的参数信息,给出了程序代码和常数的存放位置、堆栈大小以及地址范围等信息。在整个链接过程中,代码和数据的基本单位是“段”。用户将不同属性的内容放入不同的段中,链接器识别这些段,按照用户指定将各个段放入相应的存储单元中,完成链接。(3)中断向量表重定位:中断向量表是按照中断源的中断向量号顺序存放中断服务程序入口地址的一段存储区域。程序执行的第一条指令的地址以中断向量的形式给出,因此重定位中断向量表是成功跳转到用户程序执行的关键。监控程序的中断向量表起始地址为0x0000_0000,Cortex M0+内核机制中,复位时,MCU读取向量表的前两个字(前8字节),第一个字为MSP(主堆栈指针)的初始值,第二个字为复位向量(即程序执行的起始地址),程序执行从复位向量处开始。此处,中断向量表重定位指的是将本处于FLASH起始地址0x0000_0000处的用户程序中断向量表整体移到用户程序区的起始地址0x0000_7000处。事实表明,用户程序与中断向量表重定位后的用户程序的代码完全一致,区别只是代码在Flash中的存储位置不同,它们之间的转换公式为:重定位后的用户代码=0x0000_7000+用户代码。用户程序中需要对中断向量表起始地址寄存器SCB->VTOR的值进行设置,设为重定位后的地址0x0000_7000。(4)监控程序的设计(更新引导程序):监控程序的设计至关重要,关系着整体系统是否能可靠、高效的实现WSN节点的代码更新。监控程序的主要功能是接收PC机发送过来的命令帧和数据帧,解析出相应的命令及数据,将接收到的更新代码临时存放到更新用户程序区,待接收完成且校验正确后拷贝写入到用户程序区,最后通过PC指针跳转执行用户程序。更新流程图如图7所示。

[0050] 为协调监控程序与用户程序的执行顺序,防止节点异常复位后运行到监控程序中,此时系统并没有进行代码更新,导致节点空等数据帧的情况,本文设计了辅助标志位和超时机制。

[0051]

辅助标志位含义	
标志值	含义
0xff (初始值)	手动写入程序后重新上电, 监控程序检测到该标志值后直接跳转到用户应用程序。
0x01	用户程序收到更新命令帧时, 置该标志为0x01。
0x02	更新机器码接收完毕后, 置该标志为0x02。

[0052] 辅助标志位为FLASH的127扇区起始地址的内容,用于辅助监控程序与用户程序间的跳转,主要有0xff(初始值)、0x01和0x02三个值,详细介绍见上表。超时机制用于判断是否接收到代码更新相关数据帧,当计时器超过更新等待时间阈值UPDATE_TIME都没有收到相关数据帧,则自动转入到用户程序中运行,否则执行正常更新流程。传感节点程序的详细执行流程图见图8。

[0053] 关于辅助监控程序与用户程序间的跳转函数,WSN节点接收到的更新代码先存放在更新程序区,校验无误后通过调用FLASH扇区擦除函数FLASH_ErasePage对FLASH空间中的用户程序区进行擦除,并读取更新程序区的代码数据,使用FLASH扇区写入函数FLASH_Write将其逐扇区写入到用户程序区。写入正确后,需重设主堆栈指针、PC指针跳转实现程序跳转到更新代码处执行

[0054] 为了增加监控程序的通用性,节点的网络号和节点号在用户程序的总头文件中宏定义。执行用户程序时,用户程序首先调用扇区写入函数FLASH_Write将节点的网络号和节点写到FLASH的126号扇区处。当进行代码更新时,监控程序首先从126号扇区处读取中节点的网络号和节点号,然后与接收到的数据帧的网络号节点号进行对比,判断是否是该节点的数据。只有最初烧写到节点里的用户程序需要进行网络号和节点号的设置,以后的更新程序无需再设置,因为机器码发送到哪个节点是由PC机软件选择的。此种设计方法不仅使监控程序变得通用,也增强了用户程序通用性。

[0055] 关于程序保护,监控程序对代码更新至关重要,一旦被破坏,下次代码更新时就会出现错误,那么必须将节点收回重新烧写监控程序,这将会耗费巨大的人力和物力。针对这种的潜在问题,在监控程序设计时通过调用FLASH扇区保护函数FLASH_Protect()对监控程序的FLASH区进行保护设置,这样可以有效的保证监控程序不会被意外的破坏,增强了程序的可靠性。

[0056] 关于丢帧重传机制,当用户进行代码更新操作时,根节点需要通过Sub-1G技术与传感节点间进行交互,发送更新命令、机器码,接收确认帧等数据。由于根节点与WSN节点间的数据传输是通过Sub-1G技术实现的,WSN节点不能保证数据100%的接收,故本发明设计了一种简单高效的丢帧重传机制来保证通信的可靠性。PC机软件通过根节点将更新代码分组发送给传感节点,传感节点无需收到一帧数据回发一帧确认(ACK)帧,只要最后回发一帧包含丢帧信息的数据,PC机软件收到该帧解析出丢失的帧数据并重发。WSN节点判断丢帧方法:设置大小为180Bytes的缓冲区,最多可记录1440帧(每帧传输32字节的机器码,共可以传输45KB数据,即用户程序区大小)数据接收状态,缓冲区第1个字节的第七位flag_buf[0].7代表第1帧数据,flag_buf[0].6代表第2帧数据,flag_buf[0].0代表第8帧数据,以此类推。缓冲区初始化为0,成功接收到该帧数据时相应位置1。

[0057] 关于WCUCP网络协议的设计,通信协议(communications protocol)是指双方实体完成通信或服务所必须遵循的规则和约定。协议定义了数据单元使用的格式,信息单元应该包含的信息与含义,连接方式,信息发送和接收的时序,从而确保网络中数据顺利地传送到确定的地方。本发明为根节点与WSN节点间的通信设计了一套协议,即无线代码更新通信协议WCUCP,通信双方的帧格式一致,具有一定的通用性,设计如下表。

[0058]

WCUCP												
帧长	源地址			中继地址			目的地址			命令字	数据内容	校验码
	网络	路由	节点	网络	路由	节点	网络	路由	节点			
1B	2B	1B	2B	2B	1B	2B	2B	1B	2B	2B	NB	1B

[0059] 其中:

[0060] (1)帧长:包含从长度字段本身开始到校验码所有字节总长度。

[0061] (2)源地址、中继地址和目的地址分别为5个字节。源地址表示数据帧发送节点的

地址;中继地址表示转发此数据帧的中继节点的地址,初始值为源地址;目的地址表示数据帧接收节点的地址。例如,节点A要将数据帧发送给节点C,此时数据帧的源地址和中继地址都为节点A的地址,目的地址为节点C的地址。当节点B收到该数据帧时,会将数据帧的中继地址设为节点B的地址。数据传输过程中源地址和目的地址是不变的。

[0062] 路由表示节点收到数据帧是否转发,主要有0x00、0x01、0x02和0x03四种方式,0x00表示不路由,节点收到此类帧不再转发;0x01表示泛洪路由,节点收到此类帧转发一次;0x02表示递增路由,节点接收到帧的目的地址大于节点自身地址则转发一次,否则不转发;0x03表示递减路由,节点接收到帧的目的地址小于节点自身地址则转发一次,否则不转发。

[0063] (3)命令字:命令字有两个字节组成,命令字1用来区分不同的应用,命令字2是具体的操作命令,详见下表。表中命令都为字符型,方便识别,也可根据需求设计为数字。命令内容:命令内容长度待定,视具体命令确定。校验码:单字节的累加和校验。

[0064]

不同应用中的命令字				
具体应用	命令字1	命令字2	命令含义	数据内容
代码更新	C	U	更新命令	用户程序区首地址、机器码帧总数
		H	机器码数据	待更新机器码数据
		O	数据发送结束	无
		A	数据发送成功	无
		Y	接收成功、无丢帧	无
		N	丢帧重发	无
		K	程序更新成功	无
LED控制	L	W	向Flash写入数据	LED显示的字模
		O	数据帧发送结束	无
		P	显示的汉字数	汉字数
		Y	接收成功、无丢帧	无
		N	丢帧重发	无

[0065] 无线代码更新中,由于RF驱动构件的限制,每次发送的字节数最大为64,除去硬件过滤地址、有效长度、校验和等字段,最大可以发送60字节的有效数据。因此,帧协议中数据内容最长为41字节。

[0066] 关于防冲突机制,在无线代码系统中,若根节点发送一帧广播帧,WSN节点收到后会做出应答,此时根节点会同时收到多个WSN节点的数据,会产生冲突引起数据帧的丢失。因此,本发明采用CSMA/CA(Carrier Sense Multiple Access with Collision Avoidance,载波监听多址接入-碰撞避免)算法来避免数据接收时的冲突。在调用RF发送函数之前,先延时一个随机(2^{BE} 范围内,开始时 $BE=1$)的时间,然后判断信道是否空闲,如果空闲发送,否则在1到 2^{BE} 间再延时一个随机时间,此时 $BE=\min(BE+1, \max BE)$,如此循环,最多可以循环($NB < \max NB$)次。 NB 为循环变量; $\max NB$ 表示最多尝试发送的次数,如果最后还是发送失败了,那么会强制发送一次; $\max BE$:表示退避指数的最大值; BE 代表指数随机退避的指数值,随机退避的时间选择在 2^{BE} 和0之间。

[0067] 上面对WSN节点间的代码更新进行了详细研究,在此基础上实现远程代码的更新。在下面,首先给出远程代码更新的系统结构,然后介绍服务器端管理软件和通信软件、主控器各模块的底层驱动代码以及通信协议的设计。

[0068] 本文基于物联网的三层体系结构进行远程代码更新系统的设计,分别为服务器端

管理软件、通信软件和数据库的设计、网关各模块驱动程序及主控程序设计、WSN节点程序的设计,以及相关通信协议的设计。其中,WSN节点的程序与无线代码更新系统中WSN节点相同,因此WSN节点的程序设计此处不再介绍。远程更新系统如图9所示。

[0069] 关于服务器端监控软件设计,监控软件是远程代码更新系统服务器端的软件,运行于服务器上,是整个系统的重要组成部分。监控软件一方面需要与用户进行人机交互,另一方面需要与主控器进行实时通信。为增加监控软件的灵活性、可维护性和可扩展性,本文将监控软件分为管理软件和通信软件两部分。两个部分独立设计互不影响,并通过数据库进行数据的同步。其中,管理软件负责显示信息数据并提供用户直接控制的操作界面;通信软件与远程网关间进行数据通信,处理接收到的状态数据,并负责将用户通过操作管理软件产生的命令发送给网关。管理软件的开发主要有C/S(Client/Server)和B/S(Browser/Server)两种模式。C/S模式,即客户端/服务器模式,服务器通常采用高性能的PC、工作站或小型机,并采用大型数据库系统,如Oracle、Sybase、Informix或SQL Server,客户端需要安装专用的客户端软件。B/S模式,即浏览器/服务器模式,客户机上只要安装一个浏览器,如Internet Explorer或Google Chrome,服务器安装Oracle、Sybase、Informix或SQL Server等数据库,浏览器通过Web Server同数据库进行数据交互。C/S模式的优点是能充分发挥客户端PC的处理能力,但客户端都需要安装专用的客户端软件,安装和维护比较麻烦。B/S模式最大的优点就是可以在任何地方进行操作而不用安装任何专门的软件。只要有一台能上网的电脑就能使用,客户端零维护。因此,本发明的管理软件采用B/S模式,即管理软件设计为网页,用户交互界面通过浏览器实现,主要的事务逻辑在服务器端实现,数据的存储和提取则通过数据库实现。用户只需在网页中设置控制参数、点击执行按钮等操作,通过浏览器向服务器发起请求,服务器分析处理请求并计算,向对应SQL Server数据库表中存取数据。输入查询或控制参数,通过浏览器向Web服务器发起请求,Web服务器接受并处理请求,经分析计算,向相应的SQL Server数据库服务器存取数据,最后将结果返回给客户端浏览器,最后将结果返回给客户端浏览器。

[0070] 管理软件负责远程更新系统中网关及WSN节点状态的监测和管理,但最重要的功能是将待更新的机器码传到服务器指定位置,并选择要更新节点的网络号和节点号,发送代码更新相关命令。管理软件主要与数据库间进行交互,提供了友好的图形化界面供用户操作,减少了操作的复杂性,降低对操作人员专业知识的要求。通信软件主要负责网络数据的收发及解析,通过GPRS网络和网关直接进行通信,通过数据库和管理软件进行交互,在整个系统架构中起到了“桥梁”的作用。其获取数据库中未处理的命令,将这些命令按照通信协议封装成数据帧,并通过GPRS网络发送给网关;同时该软件也接收来自网关的各种状态信息的数据帧,按照通信协议解析后将状态数据存入数据库指定的表中。当通信软件获取到数据库中有远程代码更新命令时,将该命令封装成帧后发送给网关,并且解析服务器上指定文件夹中最新上传的机器码文件并分包组帧发送给网关,机器码的解析与有效数据的处理与无线代码更新系统中PC软件的设计一致,此处不再赘述。

[0071] 网关与服务器间通信需使用Socket,又称“套接字”,用于描述IP地址和端口号,应用程序通常通过“套接字”向网络发出请求或者应答网络请求。目前经常使用的Socket分为两种:流式Socket和数据报式Socket。流式Socket基于面向连接的TCP/IP协议进行通信,提供双向的、可靠的、有序的、不重复的数据流服务。通信双方必须建立一条通信连接,这样需

要较多的网络信道开支,但是保证了准确无误的信息传送。数据报式Socket基于无连接的UDP协议进行通信,提供不可靠、无连接的数据传输服务。因此,本系统中使用面向连接的流式Socket,即使用Socket实现TCP的通信机制。

[0072] 服务器采用公网固定IP地址,各网关与服务器通信软件正常数据通信前,都需作为客户端主动创建与服务器的点对点连接。通信软件通过维护一张哈希表,记录当前处于连接状态的网关和其对应的Socket句柄。每当与网关建立连接后就在该表中添加一条记录;当网关断开连接后就删除表中对应的记录。网关作为客户端通过GPRS网络与服务器端的通信软件进行通信,通信采用异步Socket方式。通信软件使用多线程异步Socket机制,使得服务器的同一个端口可以与多个网关进行通信。

[0073] 要实现网关和服务器的网络通信,首先必须在服务器端创建套接字。通信软件先进行数据库、系统参数、通信接口等的初始化,然后调用系统的Socket函数创建用于侦听的套接字,接着调用Bind方法将该Socket与本服务器地址和端口绑定。完成上述操作之后,通信软件调用Listen函数侦听网关的请求,等待网关的连接。

[0074] 网关同样调用系统的Socket函数创建套接字,接着通过Connect方法向服务器发起连接请求。当服务器端通信软件接收到网关的连接请求,就会通过BeginAccept方法接受请求并且通过EndAccept方法创建新的套接字,建立与指定的网关的连接。连接建立成功后,通信软件就可以用BeginReceive和BeginSend方法进行数据的异步传输。通信结束后,网关首先调用Close结束通信,通信软件检测到网关断开连接后也调用Close结束本次通信。服务器端通信软件和网关的Socket通信过程如图10所示。其中,通信软件绑定的本地端口号存放在数据库的SystemConfig表中,通信软件启动时从表中读取相应的端口号,这样端口号的修改比较方便,增强了通信软件的可移植性。通信软件的数据传输主要通过两个队列实现,分别为上行队列UplinkQueue和下行队列DownlinkQueue。UplinkQueue主要存放服务器指定端口收到的两种数据包:网关发送过来的数据;管理软件通过网口发送通知通信软件处理数据库中新命令的数据。DownlinkQueue主要存放管理软件下发的命令,管理软件先将命令存放于数据库表中,通信软件读取出命令放入下行队列中。通信软件与数据库及网关间的通信如图11所示。

[0075] 网关是远程代码更新系统的重要组成部分,是服务器与WSN节点通信的桥梁。网关主要接收远程服务器的控制和操作命令,解析命令监测和控制WSN节点并返回相应的应答信息。

[0076] 远程代码更新系统中有数量众多的网关和WSN节点,对它们进行合理的编址至关重要。服务器跟网关都有各自的IP地址,它们间的通信是通过TCP机制实现的。网关开始连接服务器时,将自身网络号和GPRS模块获取到的IP地址一起发给服务器。连接建立成功后,服务器通过网络号(IP地址)找到相应网关进行通信。为区分各网关,每个网关设置了一个网络号,该网关控制同一网络号下的WSN节点。每个WSN节点都有唯一的网络号和网内地址,各为2个字节,同时还需要设置部分特殊地址来实现广播或点对点数据传输。网络号占2个字节,最多可编址65536个网络,网络号在0到65535之间,其中0号网络作为预留不使用,65535号网络作为广播网络地址。网内地址也使用2个字节,最多可编址65536个节点,0号网内地址作为网关在该网络中的地址,65535号网内地址作为该网络中的广播地址。因此,RCUCP最多可编制65534个网段,每个网段内的节点数目最多有65534个,理论上节点数量最

多可为 $65534 \times 65534 = 4294705156$ 个,足以满足一般物联网系统的需求。网络特殊地址的定义如下表所示。

[0077]

网络号	网内地址	含义
0	任何	预留
1-65534	0	某网段中的网关的网内地址
1-65534	65535	某网段中的广播地址
65535	任何	该网段为广播网段, 全网广播

[0078] RCUCP编制机制将地址划分为网络号和网内地址两个部分,有效的支持网关并行的更新各网段内节点的代码,大大提高代码更新的效率。服务器可将机器码数据帧并行的发送给各个网关,网关并行的对该网络内的WSN节点进行代码更新。

[0079] RCUCP的数据帧格式设计下表。

[0080]

帧头	网络号	网内地址	路由	命令字	数据内容	校验码	帧尾
1B	2B	2B	1B	2B	NB	1B	1B

[0081] 为了防止帧数据部分丢失而导致全部数据读取错乱,本协议设计成帧头帧尾的形式,帧头固定为0x53,帧尾固定为0x54。帧的数据内容中也会出现0x53和0x54,因此本协议中增加了转义字符0x55,在帧数据内容中遇到0x53、0x54或0x55,都在其前面加0x55,分别转义为0x550x53、0x55 0x54、0x55 0x55。网络号和网内地址都由2个字节组成。路由为目的节点的路由方式,若该帧是发送给WSN节点的,则网关将该路由放入WCUCP中目的节点的路由字段中。命令字有两种类型,分别为网关命令和节点命令。网关命令是只对网关操作的命令;节点命令是最终对WSN节点的操作命令,需要网关程序将命令转化到WCUCP的命令字段中去。

[0082] 本发明直接通过网页将机器码传到服务器上的指定文件夹中,用户通过网页发送更新命令后,通信软件从服务器指定文件夹中找到相应机器码,解析提取出有效数据分包组帧发送给网关。若网页在本地解析提取机器码文件有效数据并分包存放在数据库表中,当通信软件接收到更新命令时,还需要再从数据库表中根据编号读取机器码并组帧发送,此过程比本文的方法繁琐,因此通信软件处理网页传到服务器上的机器码的方式是可取的。远程代码更新系统的更新过程为用户在系统网页上选择要更新的节点所属的网络号及节点号(节点号选项中有选择全部节点选项),点击自动更新按钮即可,此时用户的操作就已经完成。点击自动更新按钮,网页后台会将数据库相应表中的标志位会置1,通信软件查询到该变化则解析机器码中有效数据分包发送给网络号对应的网关。网关先间隔的向WSN节点发送两帧代码更新命令帧,确保WSN节点从用户程序跳转到监控程序中。随后,网关向WSN节点发送机器码数据包,并进行相应的丢帧重传。WSN节点接收机器码完成且校验正确后,将原本放在更新程序区中的代码拷贝到用户程序区中,最后将PC指针跳转到用户程序区执行。

[0083] 应当理解的是,以上所述仅为本发明的较佳实施例而已,并不足以限制本发明的技术方案,对本领域普通技术人员来说,在本发明的精神和原则之内,可以根据上述说明加

以增减、替换、变换或改进,而所有这些增减、替换、变换或改进后的技术方案,都应属于本发明所附权利要求的保护范围。

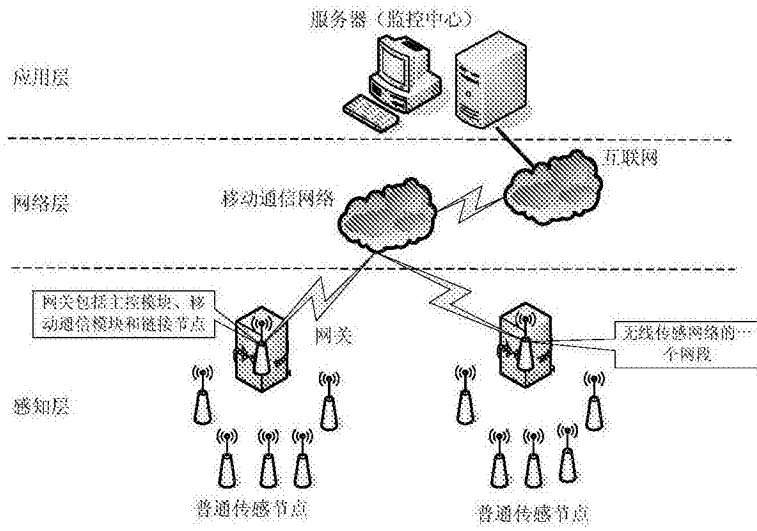


图1

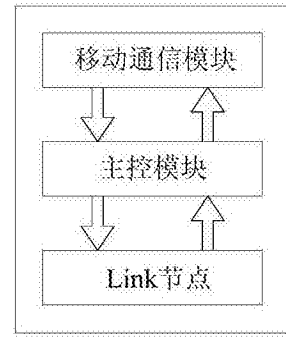


图2

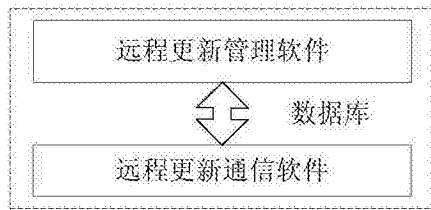


图3

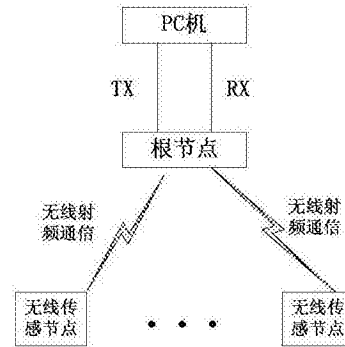


图4

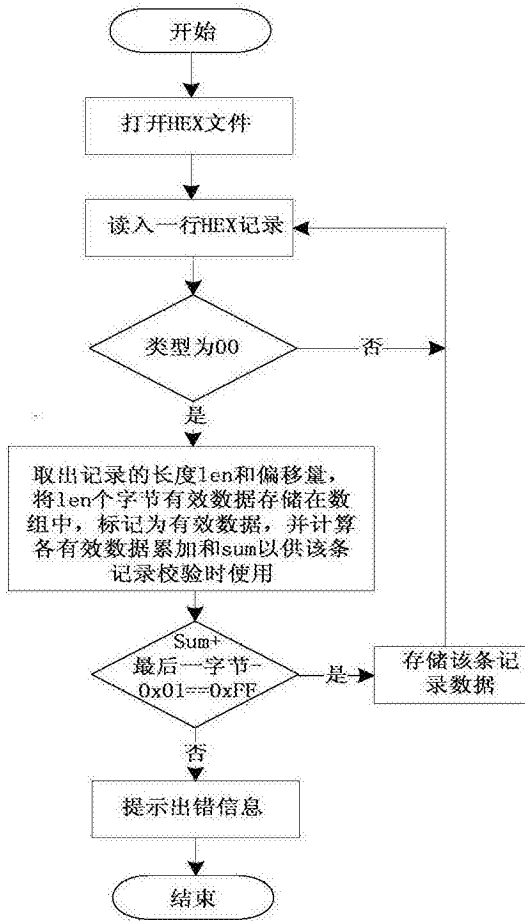


图5

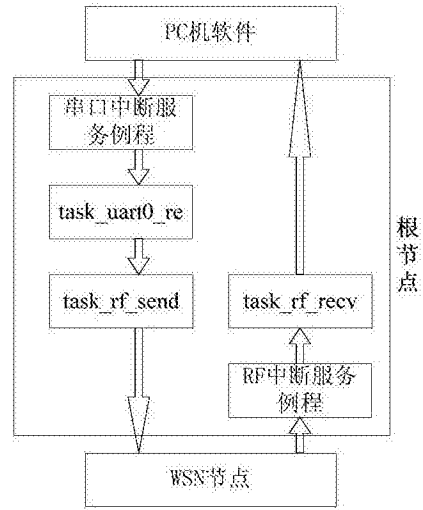


图6

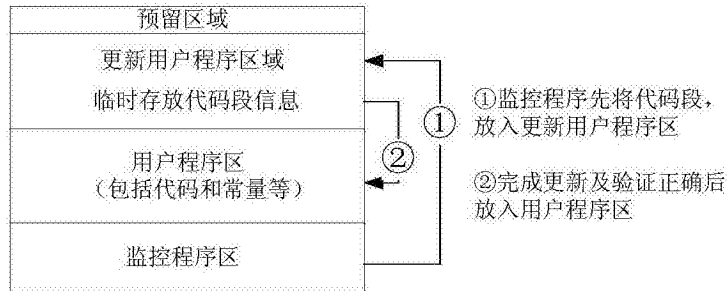


图7

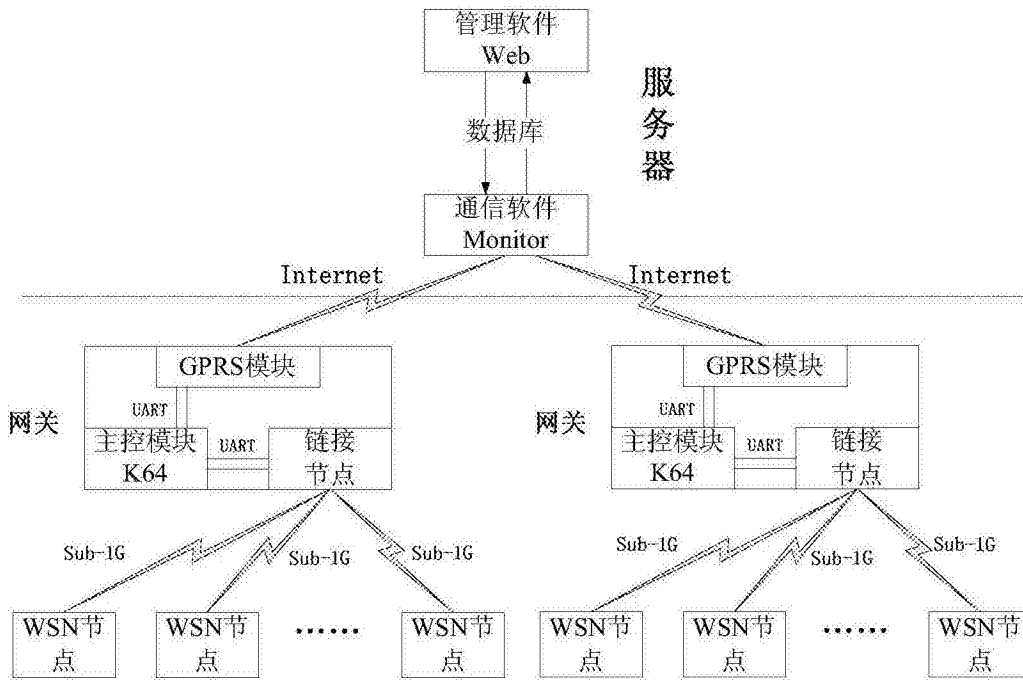


图9

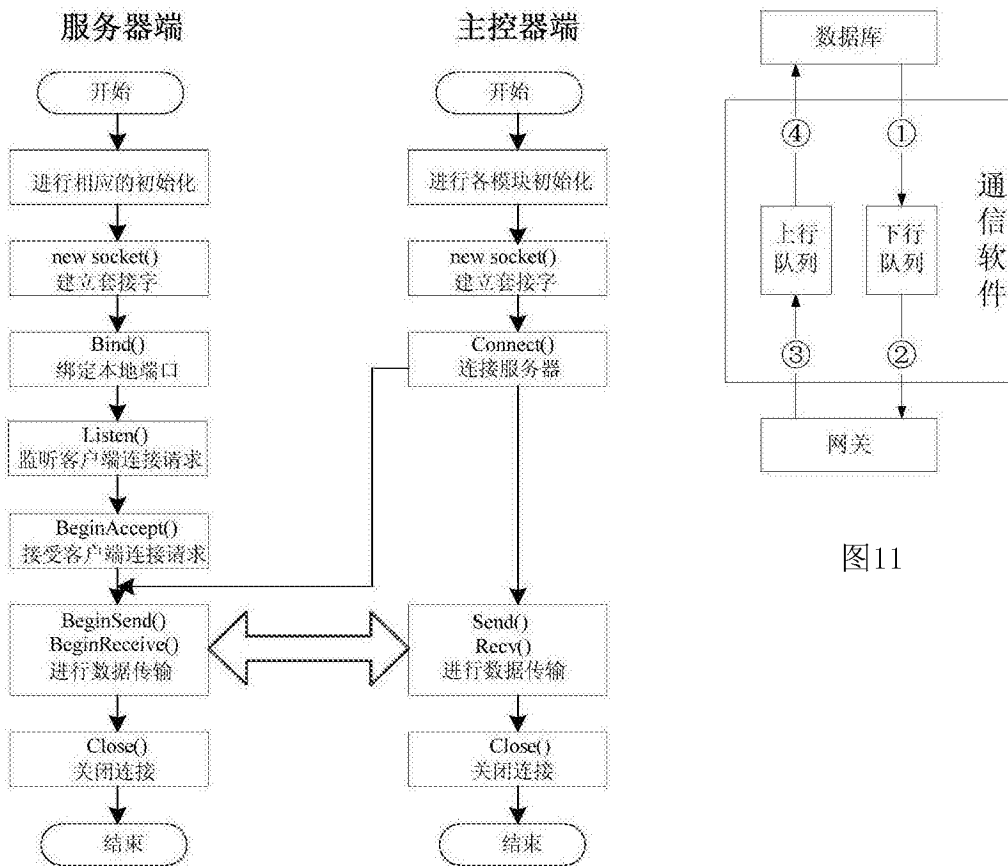


图11

图10