

Research on Power-Saving Algorithms in ZigBee Network Based on Z-Stack^{*}

GAO Xiang, DENG Yongli, LÜ Yuanyuan, LU Qiyong^{*}

(Department of Electronic Engineering, Fudan University, Shanghai 200433, China)

Abstract: ZigBee technology is one of the important techniques for wireless sensor network. Among the implements of ZigBee, Z-Stack is one of the software architecture released by TI Corporation, which has been widely used. The article used CC2530 as hardware platform and Z-Stack as software platform to analyze Z-Stack. It points out that there is power consumption issue on current Z-Stack in practical use. The corresponding power-saving algorithms are also put forward. They are power optimization algorithm for end-device nodes and low power consumption routing algorithm for router nodes. Both algorithms are verified by experiments. The results show that the power-saving algorithms based on Z-Stack proposed in the article can save power consumption effectively when the nodes work and then improve the overall life of the network.

Key words: wireless sensor network; ZigBee network; Z-Stack; low power consumption

EEACC: 6210Q

doi: 10.3969/j.issn.1004-1699.2014.11.017

基于 Z-Stack 协议栈的 ZigBee 网络节能算法的研究^{*}

高翔, 邓永莉, 吕愿愿, 陆起涌^{*}

(复旦大学电子工程系, 上海 200433)

摘要: ZigBee 技术是实现无线传感器网络的重要手段之一, 其中, Z-Stack 协议栈是 TI 公司推出的一种实现 ZigBee 技术的软件架构, 已被广泛使用。以 CC2530 为硬件平台, Z-Stack 协议栈为软件平台, 通过对 Z-Stack 协议栈的分析研究, 指出了现有的 Z-Stack 协议栈在实际应用条件下存在的功耗问题, 提出了相应的节能算法: 针对终端节点的功率优化调节算法和针对路由器节点的低功耗算法, 并通过实验进行验证。实验结果表明, 与原始的 Z-Stack 协议相比, 本文提出的基于 Z-Stack 协议栈的节能算法可以有效地减少节点在工作过程中的能耗, 从而提高网络的整体寿命。

关键词: 无线传感器网络; ZigBee 网络; Z-Stack 协议栈; 低功耗

中图分类号: TN915

文献标识码: A

文章编号: 1004-1699(2014)11-1534-05

无线传感器网络(WSN)是由部署在监测区域内大量的廉价微型传感器节点组成, 通过无线通信方式形成的一个多跳的自组织的网络系统^[1], 从而将感知区域内对象的信息传送给观察者。2003 年 2 月, 美国著名的《技术评论》杂志将无线传感器网络列为对人类生活产生深远影响的十大新兴技术之首^[2]。ZigBee 是一种低速无线个域网技术, 它适用于通信数据量不大, 数据传输速率相对较低, 分布范围较小, 但对数据的可靠性、成本和功耗有要求的场合^[3]。由于 ZigBee 技术广阔的应用前景, 使得世界各大半导体生产厂商纷纷提出了不同的 ZigBee 解决方案。其中, Z-Stack 协议栈是 TI(德州仪器)公司开发的 ZigBee 协议栈, 并且经过了 ZigBee 联盟的

认可而为全球众多开发商所广泛采用。TI 的 Z-Stack 协议栈的源码总体上由三个部分组成: 硬件抽象层、操作系统抽象层和 ZigBee 协议各层^[4]。硬件抽象层提供定时器、I/O 口、串口等资源的 API; 操作系统抽象层负责任务管理; ZigBee 协议各层集成所有任务事件^[5]。操作系统抽象层提供一种支持多任务运作的系统资源分配机制负责调度各个任务的运行, 如果有事件发生了, 则会调用相应的事件处理函数进行处理。因此, 仅从程序运行的角度而言, 对于 Z-Stack 协议栈的理解, 可以认为是操作系统执行的过程。本文通过对 Z-Stack 协议栈的分析研究, 指出现有 Z-Stack 协议栈在实际使用中的局限性, 并提出相应的改进算法, 包括针对终端的节点功率优

项目来源: 国家 863 计划项目(2011AA100701)

收稿日期: 2014-06-23 修改日期: 2014-10-04

化调节算法和针对路由器节点的低功耗路由算法。这两种算法均能减少基于 Z-Stack 协议栈的 ZigBee 节点的整体能耗。本文所研究的 Z-Stack 协议栈版本为 ZStack-CC2530-2.3.0-1.4.0, 基于 CC2530^[8] 硬件平台, 符合 ZigBee2007^[9] 规范, 具有研究和实际应用价值。

1 基于 Z-Stack 协议栈的终端节点功率优化调节算法

1.1 算法提出的背景

ZigBee 终端节点在 Z-Stack 中采用默认的发射功率进行数据发送, 而根据 Friis 传输公式^[8], 当节点的接收灵敏度一定时, 假设终端节点与协调器 (或者路由器) 之间的距离比较近或者系统损耗较小的时候, 发射功率可以适当减小, 以减少不必要的功率损耗。正是由于这种考虑, 当 ZigBee 网络中节点的布局范围比较小的时候, 原始 Z-Stack 协议栈中固定功率的配置还有进一步改进的空间。基于此, 本文提出了基于 Z-Stack 协议栈的终端节点功率优化调节算法。

1.2 算法描述

本文提出的基于 Z-Stack 协议栈的功率调节优化算法工作于网络的初始化阶段, 算法将功率等级的划分为 16 级, 其中等级 1 对应的是最低发射功率 -22 dBm, 等级 16 对应的是最高发射功率 4.5 dBm。

算法采用功率探测数据包的方式, 如图 1 所示, ZigBee 终端节点在入网成功之后, 按照功率等级发送不同等级的功率数据包, 从最低等级 (level 1) 开始, 并开启轮询, 等待协调器节点的响应。若发送的功率探测数据包在若干个轮询周期内 (可以设置为 5~10 个周期) 未得到响应, 则提高一个功率等级继续发送, 直至协调器返回了数据包的功率等级数值。为了进一步提高功率调节后网络的可靠性, 终端节点需要检测协调器返回的 Data_level_n 数据包的

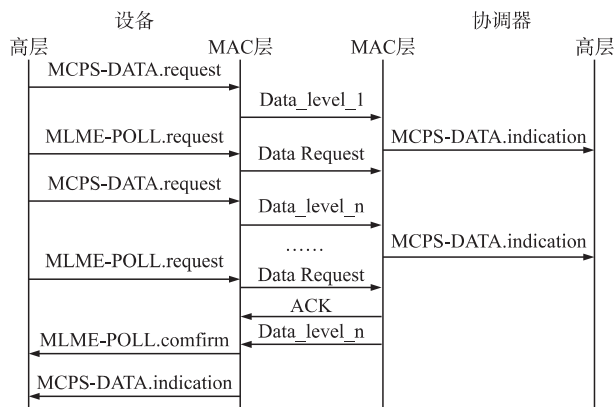


图 1 Z-Stack 协议栈中功率优化调节算法流程

RSSI 值是否大于 -85 dBm, 若协调器返回的确认功率等级数据包的 RSSI 值大于 -85 dBm, 才可以认定此功率数值为最低可靠发射功率等级。之后, 终端节点可以进入睡眠, 待唤醒时间到则可以采用最低可靠功率发送所需数据。

1.3 算法验证

算法验证在 CC2530 最小系统下进行, 其硬件结构如图 2 所示, 测试用锂电池电容量约为 210 mA·h。



图 2 算法验证测试 ZigBee 节点

在本节的算法验证中, 节点的拓扑结构和放置的位置如图 3 所示。其中, 测试环境为室内, 图中所示位置 1 与协调器节点距离约为 20 cm (中间无障碍物), 图中所示位置 2 与协调器节点的距离约为 4 m~5 m (中间有墙体和障碍物)。网络拓扑结构为星型拓扑结构, 节点初始化之后运行功率调节优化算法并进入低功耗模式, 之后定时唤醒并发送“Hello”测试数据包。测试过程中终端节点的唤醒周期为 $T=100$ ms。

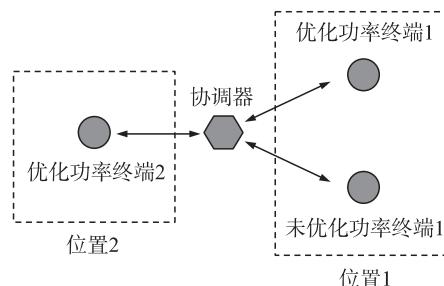


图 3 功率优化调节算法测试网络结构

最终功率优化调节算法的测试结果如图 4 所示, 通过监测测试节点工作过程中锂电池电容量与时间的关系可以看到, 即使终端节点离协调器节点的距离很近 (位置 1), 未运行功率优化调节算法的

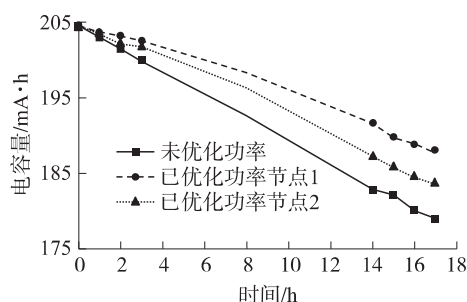


图 4 功率优化调节算法的验证

等待一定的时间 t_0 , 然后根据 SYN_MSG 消息中的统一睡眠时间值配置睡眠定时器, 设置定时唤醒时间为 t_1 , 再进入睡眠模式。

而终端节点收到 SYN_MSG 命令之后将关闭 Data Poll, 设置定时唤醒的时间为 t_2 (该时间也即网络周期性采集数据时间), 直接进入睡眠模式, 在终端节点唤醒之前, 无法收到任何的来自其他节点的消息。至此, 网络中除了协调器以外的所有节点都进入了低功耗模式。

在低功耗路由算法中, 需要保证的原则是: 路由器节点先于终端节点唤醒, 晚于终端节点睡眠。睡眠唤醒后的路由器节点需要通过软件重启恢复与其父节点的连接, 为了减少重新入网的通信开销, 在算法中采用了 Z-Stack 协议栈中的 NV (Non Volatile) 启动方式。NV 指的是非易失性存储器, 也即系统断电后, 存储在该存储器中的数据也不会丢失。在 Z-Stack 协议栈中, NV 存储器主要用于保存网络的配置参数, 如网络地址等。因此, 在低功耗路由算法中的路由器可以通过 NV 启动的方式保持其网络状态与唤醒前一致。

需要强调的是, 在低功耗路由算法中, 每个路由器节点计算出的 t_1 的值是不同的, 本文给出其公式, 如式 (1) 所示:

$$t_1 = t_2 + n \cdot T - \xi \quad (\xi \geq n_{\max} \cdot T + \delta) \quad (1)$$

其中, n 表示该节点的梯度值, 为网络的最大梯度, T 表示一个梯度补偿周期, ξ 表示最大的误差补偿, 称为最大前向失步补偿因子, δ 则用来弥补各个节点之间的时钟误差, 称为最小前向失步补偿因子。一般而言, 最小前向失步补偿因子越大, 越能提高低功耗路由的可靠性, 同时会带来一定路由器节点的能量损耗。但若最小前向失步补偿因子设置过小, 则可能会引起网络中某些时钟过快的节点成为孤儿节点。同时, 最小前向失步补偿因子的设定决定了网络中新节点入网的信标请求周期。若一个终端节点或者路由器节点需要加入网络, 而此时正好处于网络的睡眠周期, 在原始的 Z-Stack 协议栈下, 节点将耗费大量的功耗, 直至网络的睡眠周期结束。事实上, 这样的请求数据包发送的频率过于频繁, 低功耗路由延长了其信标请求的时间。但是, 如果这个信标请求的周期过长, 甚至大于上一级路由器节点的唤醒时间的话, 则将无法保证新节点在第一个同步周期内入网, 从而会加大新节点入网的时间并降低新节点成功入网的概率。因而在低功耗路由算法下, 新节点的信标请求周期应遵循 (2) 式, 以保证新节点在下一个唤醒周期内可以入网。

$$T_{\text{new}} \leq \delta \quad (2)$$

2.3 算法验证

低功耗路由算法验证的测试节点和锂电池同本文 1.3 节中描述的一致。在本节的算法验证中, 事件 A 定义为定时触发, 时间为 30 s; 事件 B 定义为按键触发, 即需要网络管理员通过按键启动整个网络的休眠过程和时间同步。其中, t_0 设置为 5 s, t_2 设置为 10 min, 最小前向失步补偿因子错误! 未找到引用源。设置为 1 min, 梯度补偿周期 $T = 10$ s。算法验证采用 1 个协调器节点、2 个路由器节点和 2 个终端节点, 构建如图 6 所示的 ZigBee 链状网络进行低功耗路由算法的测试。

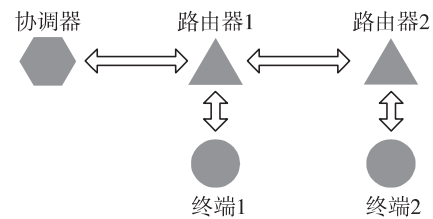


图 6 低功耗路由算法验证的 ZigBee 网络结构

本文提出的低功耗路由算法测试结果如图 7 所示。

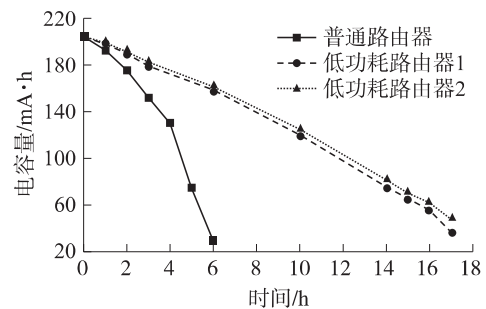


图 7 低功耗路由算法的验证

从图中可以看到, 低功耗路由器节点 1 和低功耗路由节点 2 的功耗均远小于普通低功耗路由器节点的功耗, 而这两个低功耗路由节点的功耗差别来源于梯度补偿周期产生的额外功耗。最后, 测试结果得到了普通路由器、低功耗路由器 1 和低功耗路由器 2 三者的电量耗尽时间。其中, 普通路由器 7 h 耗尽所有测试电池电量, 低功耗路由器 1 使用了 20 h 耗尽测试电池电量, 而低功耗路由器 2 使用了 21 h 耗尽测试电池电量。算法验证结果表明, 本文提出的低功耗路由算法可以有效地减少路由器节点的功耗, 从而达到网络节能的目的。

2.4 对比实验

现有关注 ZigBee 低功耗路由算法的文献多数引入节点能量代价来改进无线传感器网络中的中转路径, 从而达到低功耗路由的目的 [9-11]。其中, 参考文献 [10-11] 均提出并通过仿真验证了传统 Ad-hoc 网络中的 MMBR (Min-Max Battery Cost Routing) 算

法^[2]应用于 ZigBee 网络的可行性,该算法将节点的剩余电池电量作为路径选择的依据,并使用能量路径选择的方法延长 ZigBee 网络中节点的工作时间。本节将该算法在 CC2530 平台上进行模拟,假设与工作路由器节点平行的备选路由器节点个数为 3 个,剩余电池电量采用文献^[3]和^[4]提出的电压在线测量方法,网络根据路由器最大剩余能量的原则为终端节点分配路由器,不工作的路由器则可以进入低功耗模式,依次进行轮换。实验时睡眠路由器节点的电量估计周期为 10 s,路由选择的时间周期为 3 min,将 MMBCR 算法得到的功耗曲线和本文提出的低功耗路由算法进行对比,得到的结果如图 8 所示。

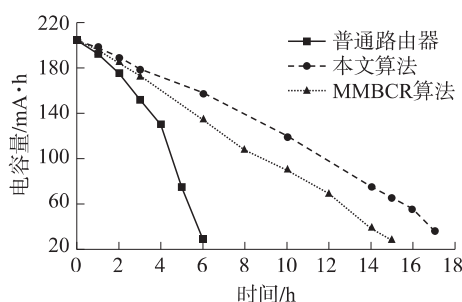


图8 对比实验的结果

对比实验的结果表明,基于同步休眠机制的低功耗路由算法比基于能量选择机制的低功耗路由算法在对局部路由器节点的节能效果更有效。事实上,MMBCR 算法解决了网络中节点能量分布不均衡的问题,从而避免了网络中出现局部“热点”;而本文提出的低功耗路由算法则充分利用了网络节点周期唤醒的特性,通过使路由器节点定时睡眠来节省能耗,具有更好的节能效果。

3 结论

本文通过对现有 Z-Stack 协议栈的分析研究,指出了现有 Z-Stack 协议栈中存在着发射功率固定和

无法实现低功耗路由的问题,并进行了改进,提出了针对终端节点的功率优化调节算法和针对路由器节点的低功耗路由算法,并在 CC2530 平台上进行了这两种节能算法的验证,实验结果证明了本文提出的算法可以有效地减少基于 Z-Stack 协议栈的 ZigBee 节点的能耗,从而延长网络寿命。本文提出的算法易于实现且具有通用性,具有实用价值。

参考文献:

- [1] 孙利民,李建中,陈渝,等. 无线传感器网络 [M]. 北京:清华大学出版社,2005:4-5.
- [2] Wade R, Mitchell W M, Petter F. Ten Emerging Technologies that Will Change the World [J]. Technology Review, 2003, 106(1): 22-49.
- [3] 郭渊博,杨奎武,赵俭,等. ZigBee 技术与应用 [M]. 北京:国防工业出版社,2010:6-10.
- [4] 张奇松,尹航. Z-Stack 剖析及其在无线测温网络中的应用 [J]. 计算机系统应用, 2009, 18(2): 103-105.
- [5] 曾宝国. Z-STACK 协议栈应用开发分析 [J]. 物联网技术, 2011(5): 71-73.
- [6] Texas Instruments. CC2530 Datasheet [EB/OL]. <http://www.ti.com.cn/cn/lit/ds/symlink/cc2530.pdf>, 2010-10-05.
- [7] ZigBee Alliance. ZigBee Specification 2007 [EB/OL]. <http://www.zigbee.org>, 2008.
- [8] Theodore S Rappaport. Wireless Communications: Principles and Practice(Second Edition) [M]. Prentice Hall, 2002: 73-74.
- [9] 董亮,张灵,陈云华. 基于限制广播的 ZigBee 分布式动态能量均衡协议 [J]. 传感技术学报, 2014, 27(8): 1120-1124.
- [10] Tekbiyik N, Uysal-Biyikoglu E. Energy Efficient Wireless Unicast Routing Alternatives for Machine-to-Machine Networks [J]. Journal of Network and Computer Applications, 2011, 34(5): 1587-1614.
- [11] Zhang Z, Zhou F, Liang Y. ZigBee Routing Algorithm Based on Energy Awareness and Energybalance [C] // Consumer Electronics, Communications and Networks(CECNet), 2012 2nd International Conference on. IEEE, 2012: 747-750.
- [12] Toh C K. Maximum Battery Life Routing to Support Ubiquitous Mobile Computing in Wireless Ad Hoc Networks [J]. Communications Magazine, IEEE, 2001, 39(6): 138-147.
- [13] 吕涛,施伟斌,范坤坤,等. WSN 节点电池供电性能测试研究 [J]. 传感技术学报, 2013, 26(10): 1457-1462.
- [14] 唐慧强,李超. 基于 CC2530 的锂电池电量检测系统的设计 [J]. 化工自动化及仪表, 2011(3): 350-353.



高翔(1989-),男,上海人,复旦大学硕士研究生,主要从事数字系统与通信、无线传感器网络方面的研究, vernegao@163.com;



陆起涌(1966-),男,江苏太仓人,复旦大学教授,硕士生导师,主要从事智能控制、嵌入式系统、物联网技术等方面的研究, lqyong@fudan.edu.cn.