

嵌入式精简 TCP/IPV6 协议栈的设计与实现

成继中, 赵欢

湖南大学计算机与通信学院, 长沙 (410082)

E-mail: chengjizhong@tom.com

摘 要: 本文通过对常用 IPV6 技术的研究和分析, 针对嵌入设备处理速度慢, 存储容量小等特点, 对常用 TCP/IPV6 协议栈进行了裁减和简化, 裁减掉一些不常用但不影响基本通信功能的协议模块, 同时对要保留下来要实现的各个协议进行简化, 只实现其基本功能。设计完成实现后的协议栈, 具有代码量少, 运行效率高和良好的可移植性等特点, 适合于各种嵌入式设备, 是一种解决嵌入式设备接入 IPV6 网络的可行方案。

关键词: IPV6; 嵌入式操作系统; 邻居发现; ICMPV6; 地址解释

中图分类号: TP311

1. 引言

嵌入式 Internet 技术是指把 Internet 技术应用于嵌入式设备, 实现嵌入式设备的信息交互, 是嵌入式技术与 Internet 技术的结合, 具有非常广阔的市场前景。目前不少厂商都在进行这方面研究, 并推出了不少嵌入式 Internet 解决方案, 比较常用的成熟的解决方案有, 瑞士计算机科学院 Adam Dunkels 写的 ulP 和 LWIP, 它们以 IPV4 技术为基础, 以精简为指导思想, 把复杂的 TCP/IP 技术引入嵌入式设备, 满足嵌入式设备接入网络的需求。而作为 IPV4 改良版本的 IPV6, 是对 IPV4 的升级和改进, 是下一代网络的核心, 如何以 IPV6 技术为基础, 设计一款和嵌入设备结合的具有代码量少, 功能简单的精简 TCP/IPV6 协议栈是一件非常现实意义的挑战, 也是本课题设计的目的所在。

2. IPV6 协议栈

IPV6 协议栈是基于 IPV6 网络层的协议, 和 IPV4 一样, 遵循现有互联网四层网络互联体系结构, 如图 1 所示。从图中我们可以看到, 协议栈分为网络接口层, 互联网

层, 传输层, 应用层四层。应用层直接面向用户, 并提供访问其它层服务的功能; 传输层用于提供源主机和目的主机上的对等实体对话; 网络接口层屏蔽了具体的硬件实

现细节, 负责底层数据的接收和发送; 网络层是整个 TCP/IP 体系结构的关键部分, 其主要功能是在网络上提供可靠的主机到主机的数据传送。IPv6 协议正是位于该层, 它包含的主要协议模块有 IPV6, ICMPV6, 邻居发现 ND, IPsec 等。

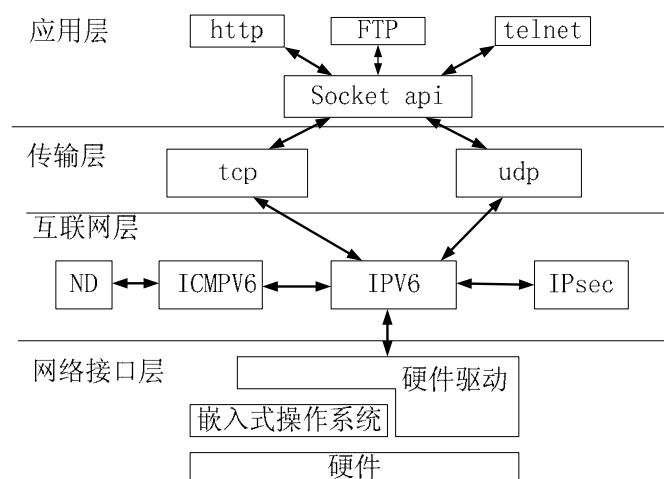


图1 IPV6协议栈体系结构

2.1 IPV6 协议

根据RFC2460对IPV6功能的描述, IPV6主要负责把上层来的数据段添加IPV6报头, 交由底层发送; 把下层接收到的报文经过处理和分析, 交给TCP, UDP或ICMPV6处理。和IPv4相比 IPv6的改变主要集中在以下几个方面: 地址容量的扩展, 报头格式的简化, 支持扩展和选项的改进, 数据流标签的能

¹ 本课题得到湖南省科技计划项目 (Science and Technology plan project of Hunan province No.05FJ3046) 的资助。

力, 认证和保密的能力等^[1]。

2.2 ICMPV6 协议

ICMPV6协议合并了IPv4中ICMP(控制报文协议), I- GMP(组成员协议)、ARP(地址解析协议)等多个协议的功能, 实现差错控制, 地址解释等功能, 并支持Mobile IPv6。ICMPV6报文封装在IP报文中, 是IP报文的有效载荷数据, 它通过它的各种错误报文和信息报文的交换来实现差错控制, 地址解释和路由前缀信息获取等功能。

2.3 邻居发现(Neighbor discovery)协议

邻居发现协议ND是IPv6协议栈中的核心协议, 是IPV6解决邻节点交互的一个重要协议。它定义了下列问题的解决机制: 路由发现, 前缀发现, 参数发现, 地址自动配置, 地址解释, 下一跳决定, 邻居不可达, 重复地址检测, 重定向。邻居发现的这些功能是通过5个ICMP报文(邻居请求/邻居通告报文, 路由器请求/路由器通告报文, 重定向报文)的交换来实现的。

3. IPV6 协议栈的精简

协议栈精简的核心是“微型化”, 我们对协议栈进行协议模块裁减和单个协议简化。

3.1 协议模块裁减

协议模块裁减是指在保障基本通信功能的前提下尽可能去掉一些协议模块, 节省系统资源。网络接口层我们只考虑 802.3 以太网协议(CSMA/CD, MAC, LLC), 不考虑面向 CAN, RS-232, RS-485, 射频, 蓝牙等相关的支持模块。接入方式上只考虑用路由器接入方式, 不考虑拨号连接方式, 去掉和拨号连接方式相关的面向点对点连接的 PPP 协议和 SLIP 协议, 这两个协议在网络接口层占用的代码量比较多; IP 层只实现基本的报头, 不实现扩展报头, 去掉基于认证头和封装安全载荷头选项的 IPsec 协议, 安

全控制交给其他层。ICMPV6 和 ND 是核心协议必须保留; 传输层 TCP 和 UDP 可以全部实现也可以只实现一种, 考虑的适应性, 本设计中都给予实现。因此协议模块裁减后要实现的核心协议族为 802.3, IPV6, ICMPV6, ND, TCP, UDP。

3.2 单个协议简化

单个协议简化是指以单个协议为目标, 进行功能和数据结构的简化。对 IPV6 协议来说, 只接收, 发送报文, 不支持报文的分片与重组, 不支持扩展报头选项, 对可靠连接传输来讲, 包过大得不到确认, 会根据拥塞控制机制和重传机制, 减少数据分组长度, 进行重新发送, 对大多数应用来说这不会产生其他严重问题。对 ICMPV6 来说, 只实现错误报文中的目的不可达报文, 信息报文中的应答回复报文, 不实现超时报文, 报文过大报文和应答请求报文, 一般包过大, 超时报文由路由器实现, 应答请求报文用于主动测试中发起测试的 PC 机一端。对邻居发现 ND 模块来说, 只实现邻居请求和邻居应答报文, 嵌入式设备刚接入网络, 它可以静态的等待网络上路由器定时发送的路由公告报文, 而不是主动发送路由请求报文来获取, 不需实现路由请求/路由应答报文。嵌入式设备连接的邻居接点, 路由一般简单, 传输量少, 不需重定向报文来进行路由定向。简化的大块在 TCP, TCP 是整个协议簇中最复杂, 代码量最多的协议。它的功能模块有: 滑动窗口, 流量控制, 拥塞控制, TCP 连接状态机, 往返时间估计, 重传协议。本协议栈的目标是有操作系统支持的嵌入式系统, 速度和存储量比 8 位和 16 位单片机都有提高, 不必采用分配固定缓冲区的形式进行接收一帧处理一帧, 可以考虑采用分配一个较大的缓冲区实现滑动窗口机制, 用来提高传输效率, 实验证明, 传输效率的提高是明显的, 往返时间估计和重传机制比较简单, 代码量不大, 可以实现, TCP 状态机表示 TCP 进程通信的状态迁移, 是 TCP 的核

心必须实现,可以不实现流量控制机制,因为流量不是很大。因此 TCP 模块实现的功能有: TCP 有限自动机,滑动窗口,往返时间估计,重传协议。忽略流量控制与拥塞控制模块,在可靠连接中,当因拥塞而发生数据丢失的时候,发送方收不到确认就采用重传机制重发数据^[2]。

4. 嵌入式精简 IPV6 协议栈的设计与实现

在设计协议栈过程中,我们在嵌入式操作系统基础上设计和实现一个操作系统模拟层,实现基本的时钟,消息管理和进程同步等基本操作系统功能。协议进程方面,把所有的协议栈封装到单独进程中,应用程序可以驻留在其中或作为一个单独的进程,这样既实现了与操作系统分离,又避免了层间切换。对于内存管理采用类 BSD buf 结构,把静态缓冲区和动态缓冲区链接起来^[3]。

4.1 IPV6

IPV6 模块主要用于完成对接收到的 IPV6 数据报进行处理,对需要发送的 IPV6 数据包进行构造并递交底层发送。当接收到一个数据包时,网络设备驱动调用 ip_input() 函数来对其 IP 报头进行检查,检查其版本号,报文长度,载荷长度,目的节点地址和下一报头,待检查无误后,根据下一包头的类型分别提交给不同的处理模块。当要发送数据时,必须要知道发送报文的下一跳 IPV6 地址,以及该地址的相对应 MAC 地址,ip_route() 函数就是为实现这样的功能而设计的,其获取下一跳 IPV6 地址与其对应 MAC 地址的处理流程如图 2 所示。

图中,目的缓存用来存储着一系列最近的报文流量与对应的下一跳 IP 地址的关系,前缀列表存储着一系列子网前缀和其他地址前缀及其对应的下一跳 IP 地址的关系,如果两者中都没有找到匹配的记录,则再从前缀列表中选择默认路由器作为传输的下一跳 IPV6 地址。

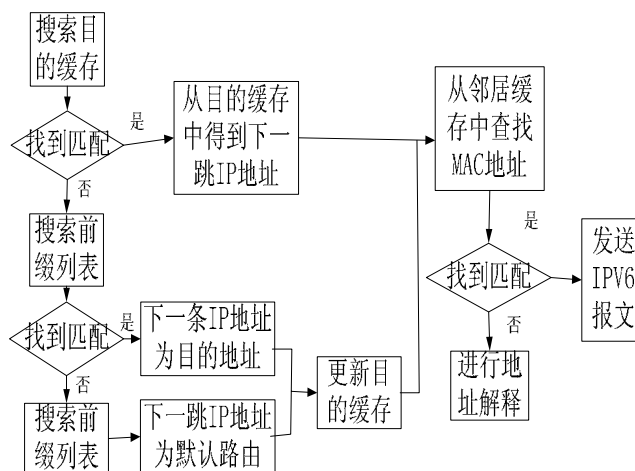


图2 下一条IP地址获取流程

在成功获取了下一跳 IPV6 地址后,数据就进入传输阶段,传输阶段由 ip_outputif() 函数控制, ip_output() 函数填充好报头,选择好发送网络接口,然后激活发送网络接口进行数据发送^[4]。

4.2 ICMPV6

ICMPV6 负责接收,解释和发送 ICMPV6 报文。收到报文后,如果是邻居信息报文则转交给邻居发现模块,如果是诊断报文则交给 ICMPV6 诊断模块。ICMPV6 模块只实现了应答回复报文,目的不可达报文。当处理到达的 IP 报文时,如果下一报头既不是 TCP,UDP 也不是 ICMPV6,那么表示在嵌入式设备端的协议栈的已经到达 IP 层,是端口不可达,发送目的不可达报文。当收到 ICMPV6 的应答请求报文时,就发送应答回复报文,其格式与请求报文相似,在收到的请求报文的基础上改变报文类型,重新计算校验和,

在 IP 报头中将源,目的地址对调就可以了。

4.3 邻居发现

邻居发现是精简 IPV6 协议簇最核心的协议,它利用邻居请求报文和邻居公告报文的交换,实现地址解释,地址重复性检测,以及地址自动配置功能。不实现路由器请求/路由器公告报文,和重定向报文。

●邻居请求报文

类型值为 135, 报文 IP 头的源地址域为发送邻居请求报文接口的地址或者未指定, 目的地址域为与被请求目标地址相关联的被请求节点组播地址, 或者就是被请求目标地址本身。ICMPV6 报头域中的目标地址域为被请求目标地址。选项域可以包含源链路层地址选项, 用来告诉对方发送请求节点的 MAC 地址, 当源地址为指定

地址时必须包含该选项。

●邻居公告报文

类型值为 136, 用来响应邻居请求报文, 或者用来告知节点其链路层地址的改变, 报文 IP 头的源地址为发送邻居公告报文的接口地址, 目的地址为发送邻居请求的单播地址, 或者是用来公告给所有邻居节点其链路层地址改变的全节点多播地址。目标地址就是被解释的 IPV6 地址, 或者在地址唯一性验证中将要采用的 IPV6 地址。

地址解释就是节点仅仅知道邻居节点 IP 地址的情况下, 通过发送邻居请求报文和接收邻居公告报文, 来得到对应节点链路层地址的过程, 是邻居发现模块中最重要的一个功能模块, 其处理过程如图 3 所示。

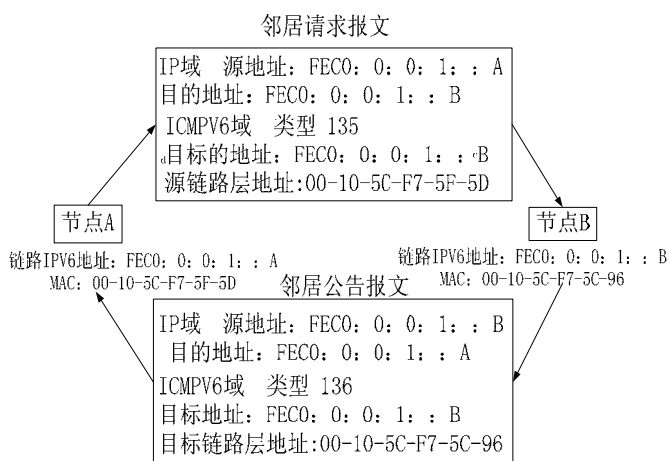


图3 邻居解释过程

节点 A 知道节点 B 的链路 IPV6 地址 FEC0:0:0:1::B 但不知道节点 B 的链路层地址 00-10-5C-F7-5C-96, 沿箭头方向, A 发送邻居请求报文, IP 域的目的地址是要求被解释

的目标地址 FEC0: 0: 0: 1: : B。节点 B 收到邻居请求报文后, 查看目标地址就是属于本机, 是则发送一个单播的邻居公告报文给 A, 在邻居公告报文的链路层地址选项里包含节点 B 的链路层地址 00-10-5C-F7-5C-96。这样

节点 A 知道了节点 B 的链路层地址, 地址解释过程完成^[5]。

5. 测试与验证

5.1 在 Altera De2 上的实现与测试

课题的开发环境: Altera De2 (硬件平台), Quartus II 5.1 和 Nios II 5.1 (软件平台), 整个开发过程以 LWIP1.1.0 为参考, 在理解了 LWIP 的结构后在结合开发环境改写。完成后对协议栈进行了测试和验证, 测试主要集中在网络层的 ND, IPV6, ICMPV6 模块。由于邻居发现模块建立在 IPV6, ICMPV6 基础上的, 对邻居模块的测试相当于对 IPV6 和 ICMPV6 也进行了测试, 很具有代表性^[6]。

受周围网络环境中无 IPV6 路由器所限, 测试在 IPV6 局域网上进行, Altera de2 通过以太网与 PC 机直接相连。测试对象电路板 MAC 地址为 00-10-5C-F7-5F-

5D, 其经过地址转换算法得到的本地 IPV6 地址为: fe80:210:5cff:fe7:5f5d, 当它接入网络时, 为了对自己将要配置的地址进行唯一性验证, 它要发送邻居请求报文, 通过 PC 端网络抓包工具 Sniffer, 我们抓到了由目标板发出的邻居请求报文, 如图 4 所示:

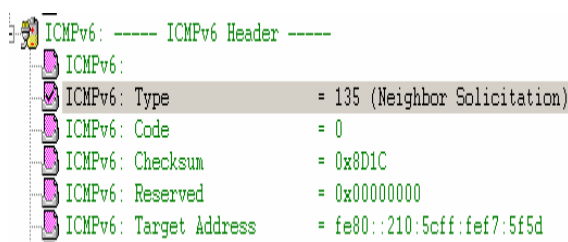


图4 邻居请求报文

从图中看到其报文的类型值为 135。目标地址为 fe80:210:5cff:fe7:5f5d。

测试协议栈在获取链路地址后,我们在 PC 机端执行 ping6 fe80::210:5cff:fe7:5f5d。这个过程中要知道目标板的链路层地址,于是发起针对目标板 IPV6 地址的地址解释。在地址解释过程中,我们抓到了目标协议栈发送的,包含自己链路层地址的单播邻居公告报文,如图 5 所示。

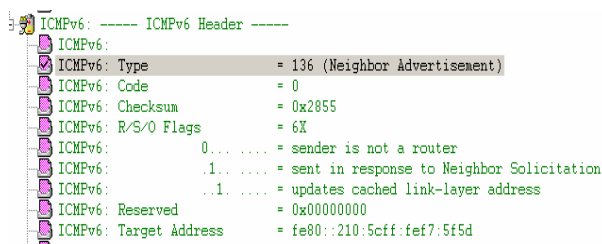


图 5 邻居公告报文

由图可得知,报文类型值为 136,目标地址为目

标板本地 IPV6 地址

fe80::210:5cff:fe7:5f5d。

5.2 在 s3c4410box 上的移植

移植目标平台:基于 s3c4410box 处理器的 ARM7 开发板,按照通用的方法,先移植了 uc/os-ii 嵌入式操作系统,在移植好的基础上用操作系统函数编写了操作系统模拟层,把网络接口层的函数指针指向电路板提供的网卡驱动程序,在系统启动初试化函数中添加针对 IPV6 协议栈的启动代码。完成这些后我们使用 altera de2 上一样的测

试方法进行测试,实验结果证明协议栈满足基本通信功能。证明协议栈可以在该电路板上进行移植^[7]。

6. 结束语

本文介绍了嵌入式精简 TCP/IPV6 的设计思想和实现方法,精简性和可移植性是其考虑的主要方面,该协议栈是一种解决了嵌入设备和接入 IPV6 网络的可行解决方案。

参考文献

- [1] Robert e f. Embedded Internet Systems Come Home[J]. IEEE Internet Computing,2001,5(1):52-53.
- [2] Ruhuarvi j,Mahonen P,Saaranen M J. providing Network Connectivity for Small appliances a Functionally Minimized Embedded Web Server[J].IEEE Communication Magazine,2001.39(10):74-79.
- [3] Soung S. Network-Driven layered multicast with IPV6[J],Lecture Notesin Computer Science, 2000 , Volume 18 :11.
- [4] Liu Li-feng,Zou Shi-hong. A congestion and rate control scheme based on directed diffusion in wireless sensor networks[J].Journal of Beijing University of Posts and Telecommunications,2006,29(2):54-58.
- [5] Chris M,Mailik T, A look at native Ipv6 multicast[J], IEEE Internet Computing,2004 Volume8 Issue4: 48
- [6] 周立功. SOPC 嵌入式系统实验教程. 深圳:北京航空航天大学出版社. 2006 :241-248
- [7] 姚羽, 石林. IPV6 技术项目实验指导书. 北京:电子工业出版社. 2007 :24-28

Design and achievement of Embedded Light-Weight TCP/IPV6 Stack

Cheng Jizhong, Zhao Huan

College of computer and communication, Hunan University, Changsha (410082)

Abstract

Via the research and analyse for the IPV6 technique in this article. In allusion to the MCU on embeded system is not fast, and the storage capability is low, we cut down the common IPV6 stack. In this design we cut down some unusuary used but not affect basic communication protcols. Besides, for the saved protocols we only realize it's basic function. After the achievment we find that this stack little-codes, efficiency-runing and have good grafted ability. So it fit for embeded system devices, and be considered as a feasible scheme for embeded system connecting to IPV6 network.

Keywords: PV6; Embedded Operating System; Neighbor Discovery; ICMPV6; Address Resolution.

作者介绍:

成继中(1981-), 男, 湖南娄底人, 硕士, 主要研究方向: 嵌入式系统结构;

赵欢 (1967-), 女, 教授/博士, 主要研究方向: 嵌入式语音合成。