

摘 要

无线接入是未来通信网络极其重要的接入方式,尤其是基于 IEEE 802.11 协议标准的无线局域网(WLAN),由于其带宽高、布网简单、频谱无需分配,已成为最为重要的宽带无线接入技术之一,并是未来可重配置终端必不可少的配置技术。本论文的目的就是研究如何在 DSP 平台上,设计和开发 802.11 标准的媒体服务控制(MAC)层协议。

论文总结笔者参与江苏省高科技项目“下一代泛在网络端到端重配置关键技术”的研究心得,深入研究和探讨在 ARM、DSP 和 FPGA 的协作环境下,IEEE 802.11MAC 层协议软件的设计和实现技术。

论文首先介绍了无线局域网的发展,并对其进行技术分析,引出了 IEEE802.11 标准。然后系统地分析了 IEEE 802.11 标准的媒质访问控制(MAC)层总体功能,详细介绍了其中的分布协调功能(DCF)和点协调功能(PCF),并对其中的关键技术作了重点阐述。接着给出了采用 TMS320C5410 DSP 实现的 IEEE 802.11MAC 层协议设计方案,设计并编写了 MAC 层的调度模块、MAC 层与物理层及 LLC 层的接口模块。最后详细讨论了 STA MAC 层协议的 DSP 实现技术,给出了 MAC 层各功能模块的 C 语言软件设计。

论文提出的 IEEE 802.11 MAC 层协议栈 DSP 实现方案有良好的可扩展性,为项目后续实现终端 WLAN 和 WiMAX 的重配置奠定了基础。

南京邮电大学
硕士学位论文摘要

学科、专业：工 学 通信与信息系统

研究 方 向： IP 与宽带网络技术

作 者： 2005 级研究生 朱 军

指导教师 糜正琨

题 目： IEEE 802. 11MAC 层协议实现技术研究

英文题目： The Study of Implementation Technology of IEEE
802. 11 MAC Protocol

主 题 词： 无线局域网 IEEE 802. 11 分布协调功能
点协调功能 数字信号处理

Keywords: Wireless Local Area Network(WLAN)
IEEE 802. 11
Distributed Coordination Function(DCF)
Point Coordination Funtion(PCF)
Digital Signal Processing(DSP)

Abstract

Wireless access is an extremely important future communication network access technology, in which IEEE 802.11 based WLAN is of the most importance owing to its high bandwidth, simple networking and no need for spectrum allocation. It is also an indispensable reconfiguration technology for the future reconfigurable terminals. This paper is aimed at the study of the design and development of 802.11 MAC protocol on DSP platform.

Based on the author's research experience in the Jiangsu Provincial high-tech project "Research on the key technologies of end-to-end reconfiguration for next generation ubiquitous communication networks", The paper makes in-depth study on software design and implementation technology of IEEE 802.11 MAC protocol under an ARM, DSP and FPGA collaborative environment.

Following an introduction of the development of WLAN and its technical analysis, IEEE 802.11 standard is briefed. Systematic analysis is then given to 802.11 MAC layer functionality with detailed discussion of its distributed coordination function (DCF) and point coordination function (PCF) and the involved key technologies. Furthermore a design scheme is proposed of IEEE 802.11 MAC layer protocol using TMS320C5410 DSP. The MAC layer scheduler module, the interface module between MAC layer and physical layer, and that between MAC and LLC layer is designed and programmed. Finally detailed discussion is provided of the DSP implementation technology of STA MAC protocol with software design in C of each functional module of MAC layer.

The proposed DSP implementation scheme of IEEE 802.11 MAC layer protocol is of good scalability and has laid solid foundation for the follow-up project to realize the WLAN and WiMAX reconfiguration in the terminals.

南京邮电大学
硕士学位论文摘要

学科、专业：工 学 通信与信息系统

研究 方 向： IP 与宽带网络技术

作 者： 2005 级研究生 朱 军

指导教师 糜正琨

题 目： IEEE 802. 11MAC 层协议实现技术研究

英文题目： The Study of Implementation Technology of IEEE
802. 11 MAC Protocol

主 题 词： 无线局域网 IEEE 802. 11 分布协调功能
点协调功能 数字信号处理

Keywords: Wireless Local Area Network(WLAN)
IEEE 802. 11
Distributed Coordination Function(DCF)
Point Coordination Funtion(PCF)
Digital Signal Processing(DSP)

南京邮电大学学位论文独创性声明

本人声明所呈交的学位论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知,除了文中特别加以标注和致谢的地方外,论文中不包含其他人已经发表或撰写过的研究成果,也不包含为获得南京邮电大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

研究生签名: 牛军 日期: 2008.4.11

南京邮电大学学位论文使用授权声明

南京邮电大学、中国科学技术信息研究所、国家图书馆有权保留本人所送交学位论文的复印件和电子文档,可以采用影印、缩印或其他复制手段保存论文。本人电子文档的内容和纸质论文的内容相一致。除在保密期内的保密论文外,允许论文被查阅和借阅,可以公布(包括刊登)论文的全部或部分内容。论文的公布(包括刊登)授权南京邮电大学研究生部办理。

研究生签名: 牛军 导师签名: 廖正波 日期: 2008.4.11

第一章 引言

1.1 无线局域网发展概述

无线局域网(Wireless Local Area Network, WLAN)技术在 20 世纪 80 年代中期提出,源于美国联邦通信委员会(Federal Communication Committee, FCC)对工业、科研和医学(Industrial Scientific Medical, ISM)频段公共应用的授权,它使用 ISM 频段为普通用户提供宽带局域网服务。

无线局域网利用无线通信技术在一定范围内建立计算机设备间的无线通信网络,利用无线电波在空间发送和接收数据,是计算机网络和无线通信技术相结合的产物。与有线通信悠久的历史相比,无线网络的历史并不长,特别是充分发挥无线通信的“可移动性”特点的无线局域网是 20 世纪 90 年代才开始实际应用。

1.2 IEEE 802.11 协议的引入

随着应用的进一步发展, WLAN 正逐渐从传统意义上的局域网技术发展成为“公共无线局域网”,成为国际互联网 INTERNET 宽带接入手段。在 WLAN 迅猛发展的同时, WLAN 的标准之争也成为众多制造商和运营商非常关注的一个话题。

WLAN 标准主要是针对物理层(Physical Layer, PHY)和媒质访问控制(Media Access Control, MAC)层,涉及到所使用的无线频率范围、空中接口通信协议等技术规范与技术标准。迄今为止,在已制定的众多 WLAN 标准中有 IEEE(Institute of Electrical and Electronics Engineers,美国电子电气工程师协会)802.11 系列(如: IEEE 802.11、IEEE 802.11b、IEEE 802.11a、IEEE 802.11g 等)、ETSI(European Telecommunication Standards Institute,欧洲电信标准化组织)提出的标准 HiperLan 和 HiperLan2, HomeRF 工作组的两个标准 HomeRF 和 HomeRF2。目前,基于 IEEE 802.11 系列标准的无线局域网具有比较广泛的应用,得到世界上绝大多数芯片制造商和设备制造商的支持。

国际电子电气工程师协会(IEEE)是全球公认的局域网权威机构,其中 IEEE 802 工作组负责局域网标准的开发,如 802.3 Ethernet 协议、802.5 Token Ring 协议等。IEEE 802.11 工作组的任务是在 IEEE802 委员会已建立好的框架下制定 WLAN 标准,并解决不同 WLAN 设备之间的兼容性问题。在 1997 年, IEEE 802.11 工作组经过 7 年的工作,完成了 IEEE

802.11 标准的制定,它定义了与 IEEE 802.3 有线物理层(Physical Layer, PHY)和媒质访问控制层(MAC)等等的无线 PHY 和 MAC 层,这是在无线局域网领域内第一个国际上被认可的协议标准。但是由于 IEEE 802.11 所提供的数据速率较低(理论最高速率只有 2Mbps),无法满足用户对高速数据传输的需求,在随后的时间里 IEEE 802.11 得到不断的改进,IEEE 相继推出了 802.11b, 802.11a, 802.11g 等一系列标准,使无线局域网的物理层技术不断的得到完善,也使无线局域网的应用得到了很好的推动。

1.3 本人所做的工作

本文研究源于江苏省工业高科技研究项目“下一代泛在通信网络中端到端重配置关键技术研究”。通过对 IEEE 802.11 MAC 协议的系统性研究,给出了基于 ARM(ARM9)、DSP(TMS320C5410)和 FPGA 的协作环境下 MAC 层协议的实现方案,设计编写了 MAC 层的核心调度模块及 MAC 层与物理层及 LLC 层的接口模块,并使用 C 语言对 MAC 层协议的功能模块进行了实现及联合调试,验证了相关模块及算法的正确性及有效性。

第一章首先回顾了无线局域网的发展,引出 IEEE 802.11 标准,并介绍了本文研究背景。第二章讨论了无线局域网的组网结构、优点及应用;简要介绍了目前比较流行的四种无线局域网技术标准,并进行了比较;简单介绍了 802.11(a/b/g 等)系列协议。第三章对无线局域网的体系结构及组成元素进行了阐述,对 IEEE 802.11 的服务进行了详细地描述,讨论了服务的关系及支持服务的消息内容。系统地分析了 MAC 子层的总体功能结构,详细介绍了其中的分布协调功能(Distributed Coordination Function, DCF)和点协调功能(Point Coordination Function, PCF)功能,并对其中的关键技术作了重点阐述。这些分析为第五章程序的编写打下了基础。第四章给出了基于 ARM(ARM9)、DSP(TMS320C5410)和 FPGA 的无线接入点 MAC 层协议栈实现方案,描述了终端的系统结构及工作流程,设计并实现了 MAC 层基于 DSP(TMS320C5410)的调度模块及 MAC 层与物理层、LLC 层的接口模块,并对接口模块进行了调试和分析。第五章参照 IEEE 802.11 MAC 协议流程,简单介绍了 MAC 层协议程序框图,对其中的九个软件模块进行了介绍及实现,对整个 MAC 层协议栈的实现进行了调试和测试。第六章总结了本论文的工作情况,说明了后期进一步工作的主要内容和计划安排。

第二章 无线局域网技术

采用无线传输方式的计算机局域网——无线局域网(WLAN)[1]是相当便利的数据传输系统,凭借自身简单的存取结构,使得用户能够通过它实现“信息随身化,便利走天下”的理想境界。WLAN 利用射频(Radio Frequency, RF)技术取代了传统有线网络中的铜缆双绞线和光纤等传输介质,并被看作有线网络的有效延伸,在某些环境还可以替代传统的有线网络。

2.1 无线局域网的技术要求[1]

无线局域网与以往的基于蜂窝电话网、专用分组交换网及其它技术的无线通信相比,有许多本质上的区别。首先无线局域网必须支持高速突发数据业务,在室内使用时要解决包括多径衰落、相邻子网间串扰等问题。下面列出无线局域网必须解决的技术难点。

1. 可靠性

无线局域网的信道误比特率应尽可能低,这样保证了通信系统的可靠性和稳定性。否则,当误比特率过高而又不能被纠错码纠正时,该错误分组将被安排重发。这样,大量的重发分组会使网络吞吐量显著下降。据实验数据表明,如果系统分组丢失率 $\leq 10^{-5}$,或信道误比特率 $\leq 10^{-8}$,可以保证较满意的网络性能。

2. 兼容性

对室内应用的无线局域网,应尽可能与现有有线局域网兼容,现有的网络操作系统和网络软件应能在无线局域网上不加修改地正常运行。

3. 数据速率

为了满足局域网的业务环境,无线局域网至少应具备 1Mbps 以上的数据速率。

4. 通信保密

由于无线局域网的数据经无线媒质发往空中,要求有较高的通信保密能力。无线局域网可在不同层次采取措施来保证通信的安全性。首先,采取适当的传输措施,例如采用扩展频谱技术,使盗听者难以从空中捕获到有用信号。其次,为防止不同无线局域网间干扰与数据泄漏,需采取网络隔离或设置网络认证措施。最后,在同一网中应设置严密的用户口令及认证措施,防止非法用户接入。还应设置用户可选的数据加密方案,即使信号被窃听也难于理解其中的数据内容。

5. 移动性

无线局域网中的站分为全移动站与半移动站两类。全移动站指在网络覆盖范围内该站可在移动状态下保持与网络的通信,例如蜂窝电话网的移动站(手机)即是一种全移动站。半移动站指在网络覆盖范围内网中的站可自由移动,但仅在静止状态下才能与网络通信。支持全移动站的网络称为全移动网络,而支持半移动站的网络称为半移动网络。按以上分类,目前的无线局域网大都属于覆盖范围极小的(几米~几百米)的全移动网络。为了扩大覆盖范围和提高频带利用率,必然导致引入蜂窝或微蜂窝网络结构。

6. 节能管理

由于无线局域网要面向便携机(其电池能量有限)使用,为节省便携机内电池的消耗,网络应具有节能管理功能。即当某站不处于数据收发状态时,应使机内收发机处于休眠状态,当要收发数据时,再激活收发机。

7. 小型化、低价格

这是无线局域网能够实用并普及的关键所在。这取决于大规模集成电路,尤其是高性能、高集成度砷化镓技术的进步。

8. 电磁环境、无线电频段的使用规范

在室内使用的无线局域网,应考虑电磁波对人体健康的损害及其它电磁环境的影响。无线电管理部门应规定无线局域网能够使用的频段,规定发射功率及带外辐射等各项技术指标。

2.2 无线局域网的特点

WLAN 以无线多址信道作为传输媒质,提供传统有线局域网 LAN 的功能,能够使用户真正实现随时、随地、随意的宽带网络接入,因而各团体、企事业单位广泛地采用了 WLAN 技术来构建其办公网络。

相对于有线局域网,无线局域网在无线组网及安全性等方面有如下的优点:

1. 通信可靠

在有线网络中,线缆的故障常常是网络瘫痪的主要原因。使用无线网络技术由于没有线缆,就彻底避免了由于线缆故障造成的网络瘫痪等问题。

另外,无线局域网采用扩频传输技术和补偿编码键控调制编码技术进行无线通信,具有抗射频干扰强的特点。同时采用的智能放大器和智能天线产品,具有理想的接收灵敏度,并可提供强大的、可靠的无线传输。

2. 成本低, 设备安装快速、简单

使用 WLAN 可以避免安装线缆的高成本费用、租用线路的费用以及当设备需要移动而增加的相关费用, 所以无线局域网可以极大地降低组网成本。由于无线局域网系统消除布线的繁琐工作, 网络可遍及有线不能到达的地方。

3. 灵活性及扩展能力

由于没有线缆的限制, 用户可以随心所欲地增加工作站或重新配置工作站。无线局域网可组成多种拓扑结构, 容易从少数用户的对等网络模式扩展到上千用户的结构化网络。

4. 移动性

在服务区域, WLAN 设置允许用户在任何时间、任何地点访问网络数据, 不需要指明访问地点, 因此用户可以在网络中漫游。

5. 高吞吐量

WLAN 可以实现 54Mb/s 的数据传输速率。如此的高吞吐量, 加上灵活性和移动性的特点, 通过无线局域网能很大地提高工作效率。

2.3 无线局域网的分类[2]

在 WLAN 迅猛发展的同时, WLAN 的标准之争也成为众多制造商和运营商非常关注的一个话题。下面简要介绍几种目前 WLAN 产品所采用的技术标准, 并在此基础上分析各自的优缺点。

IrDA红外数据通信[3]

红外线数据标准协会IrDA (Infrared Data Association)是致力于建立无线传播连接的国际标准组织。IrDA是一种低成本的红外线点对点通信的技术, 其相应的软件和硬件技术都已经比较成熟。它的优点是体积小、功率低、适合设备移动的需要、传输速率高(可达16Mbps)、成本低、应用普遍。

IrDA技术的致命缺陷在于: 首先IrDA是一种视距传输技术, 只有当红外收发设备点对点视线直接对准时才能进行数据传输, 不能穿透阻挡物; 通信只限于在两个设备之间进行, 如果是多个设备就必须彼此调整角度和位置等; 没有提供链路级安全性能, 底层的安全只依赖于红外的视线直接对准的通信特点, 可能被窃听; IrDA设备中的核心部件—红外线LED不十分耐用。这些缺点限制了IrDA技术的发展和应用。

HomeRF[4]

家庭射频HomeRF工作组成立于1998年, 致力于创建用于无线数据传输的开放性规范。

工作组认识到有线连接带来的移动困难、铺设新线路的高费用将阻碍家庭网络技术的普及,为此工作组开发了SWAP (Shared Wireless Access Protocol)协议,用于家庭范围内的计算机与家用电子设备间的无线通信。

HomeRF主要为家庭网络设计,是IEEE 802.11和DECT(数字无绳电话标准)的结合,旨在降低语音数据成本。其工作频段是2.4GHz,采用跳频扩频技术,跳频带宽为1MHz,在不同调制方式下,其数据传输速率为1M bps或4Mbps,支持数据和语音通信,传输范围超过100m。HomeRF的主要适用范围局限于家庭无线局域网,由于该技术没有完全公开,并且在抗干扰等方面相对其它技术尚有缺陷,因此其市场应用前景并不乐观。

HiperLAN2标准[5]

欧洲电信标准化协会(Europe Telecommunication Standardize Institute, ETSI)批准了HiperLAN2标准的核心技术规范,这意味着,最终用户将能以54 Mbps的速率通过无线方式接入互联网(原始物理层吞吐率高达54 Mbps,网络层的吞吐率最低也能保持在20 Mbps左右),使用各种多媒体和即时视频服务。HiperLAN2是一种新的高性能无线传输技术,工作频率为5GHz。该系统能够迅速且简单地安装,并可以IP、以太网、PPP、ATM及IEEE 1394等多种核心网络技术互连互通。它应该还包括一个面向UMTS(Universal Mobile Telecommunications System)的界面,可以作为其它面向第三代移动系统IMT-2000系列的界面基础。

IEEE 802.11标准

IEEE 802.11是无线局域网中最普遍使用的标准,也是本论文研究探讨的重点。

2.4 IEEE 802.11 系列协议

IEEE 802.11 标准工作组成立开始的任务是研究 1Mbit/s 和 2Mbit/s 的数据速率、工作在 2.4Ghz 开放频段的无线设备和网络发展的全球标准,主要用于难于布线的环境和移动环境中计算机的无线接入。现在 802.11 是一个系列协议,最初为 802.11[6],其后又相继开发了 802.11b,a,g 和 n。此外,IEEE 802.11 工作组正在为这些基础标准提出一些改善和拓展方案,其下设的多个特别工作组正在为各种标准的扩展和增强而努力,以期能带来新的功能,提高 QoS 及安全性等,这些工作对于最终用户有着重要的意义。

1999 年 802.11a[7]标准制定完成,其工作频段为 5.15~5.825GHz,数据传输速率达到 54Mbps/72Mbps(Turbo),传输距离控制在 10~100m。采用正交频分复用(Orthogonal Frequency Division Multiplexing, OFDM)的独特扩频技术;可提供 25Mbps 的无线 ATM 接口和

10Mbps 的以太网无线帧结构接口、以及 TDD/TDMA(Time Division Duplex Time Division Multiple Access)的空中接口;支持语音、数据、图像等业务;一个扇区可接入多个用户,每个用户可带多个用户终端。

1999 年 9 月 802.11b[8]被正式批准,该标准规定无线局域网工作频段在 2.4~2.4835GHz,数据传输速率达到 11Mbps,这给移动用户带来了以太网级的网络性能、速率和可用性。802.11b 是对 802.11 的一个补充,采用点对点模式和基本模式两种运作模式,在 802.11 的基础上增加了两种更高的通信速率:5.5Mb/s 和 11Mb/s,可因环境变化,在 11Mb/s、5.5Mb/s、2Mb/s 和 1Mb/s 之间动态切换,且在 2Mb/s 和 1Mb/s 速率时可与 802.11 DSSS 系统兼容。802.11b 使用直接序列扩频技术(Direct Sequence Spread Spectrum, DSSS)作为射频(Radio Frequency, RF)传输标准。目前技术最为成熟、市场应用最多的是 802.11b(即 Wi-Fi 标准)。本论文也侧重于 802.11b 协议标准的研究。

802.11g[9]标准是对流行的 802.11b 的提速(11Mb/s 提高到 54Mb/s),而且能兼容 802.11b。802.11e[10]标准对无线局域网 MAC 层协议提出改进,支持多媒体传输及无线局域网无线广播接口的 QoS 机制。802.11f[11]定义访问节点之间的通信,支持接入点互操作协议(Inter-Access Points Protocol, IAPP)。802.11h[12]用于 802.11a 的频谱管理技术,是对 802.11a 的扩展,用来兼容其他 5GHz 频段的标准。802.11i[13]标准是为 802.11a/b/g 进一步提供加密技术的无线数据网安全协议,它引入了新的加密密钥协议:动态密钥完整性协议(Temporal Key Integrity Protocol, TKIP)和高级加密标准(Advanced Encryption Standard, AES),解决了有线等效加密(Wired Equivalent Protocol, WEP)协议缺少足够的安全性的问题。

此外,新兴的 802.11n 标准结合了 MIMO (Multi-In-Multi-Out,多入多出)和 OFDM 等多种技术,以便形成很高的速率(高达 600Mbps),同时又能与 802.11b/g 设备通信。802.11n 被看作是下一代的无线网络技术,可提供对带宽最为敏感的应用所需的速率、范围和可靠性。遗憾的是,802.11n 标准尚未得到 IEEE 的正式批准,仍处于草案阶段,面临着一种“标准滞后、产品早产”的尴尬境地。

除了 802.11n,802.11s 将是 2008 年无线领域又一大热点。802.11s 定义了应用于网状网络的无线通信规范,将极大地规范 Mesh 产品的互操作性,并消除现有无线 Mesh 网络移动漫游方面的技术障碍。

2.5 无线局域网的应用

根据无线局域网的特点,其应用可分为两类:一类作为半移动网络应用,一类作为全移动网络应用。

1. 半移动应用

在半移动应用环境下,又可分为室内应用和室外应用。

在室内应用场景下,无线局域网可以替代有线局域网,或者作为有线局域网的补充,与有线局域网并存。因此,企业可自己建立面向企业内部用户的 WLAN 网络。无线局域网在以下应用情况可发挥其无线特长:大型办公室;车间;超级市场、智能仓库;临时办公室、会议室;证券市场等。这类应用可以显著提高企业的信息化程度,促进企业的发展;而随着企业对信息化的重视,这类应用必将得到迅速发展。

在难于布线的室外环境下,无线局域网可充分发挥其高速率、便携性、安装简易、组网灵活之优点。尤其在公共通信网不发达的情况下,可作为区域网(覆盖范围几十公里)使用。下面列出几种应用情况:城市建筑群间通信;学校校园网络;工矿企业厂区自动化控制与管理网络;银行、金融证券城区网络;城市交通信息网络;矿山、水利、油田等区域网络;港口、码头、江河湖坝区网络;野外勘测、实验等流动网络;军事、公安流动网络等。

2. 全移动网络应用

无线局域网与有线主干网构成移动计算网络。这种网络传输速率高、覆盖面大,是一种可传输多媒体信息的个人通信网络。这是无线局域网的发展方向。

作为新兴技术, WLAN 经过几年的推进和发展,其标准和产品已经日见成熟,应用也日渐广泛。不仅在销售、物流、电力、服务、教育,证券、商务应用等领域有了很大的发展,全球无线城市、热区的建设业已受到世界各国网络运营商的关注。在国外,由计算机设备供应商、通信运营商和专业服务提供商在内的庞大阵营正积极进军这一领域,无线局域网的热潮正在冲击全球的市场。在国内市场上,无线局域网的应用也已经进入关键阶段,中国移动、中国电信、中国网通等电信运营商纷纷结合各自网络开通了无线局域网业务,为应用市场带来了生机,也为无线网络产业带来无限商机。

从市场发展看,目前中国无线局域网整体市场规模不大,但市场发展速度很快,巨大的市场潜力被各大 WLAN 设备和芯片厂商看好。同时,2008 年是奥运年,也是 IT 技术的奥运盛会,对中国的 IT 业来说,其重要性无法估量。正如 2000 年的互联网让奥委会发出了

“奥运会走到了互联网的十字路口”的慨叹，WLAN 在 2002 年世界杯上的处子秀至今是 WLAN 厂商津津乐道的谈资，无疑，2008 年将成为未来 4 年中国 WLAN 产业的一针长效兴奋剂。

第三章 IEEE 802.11 无线局域网标准

3.1 IEEE 802.11 标准

IEEE 802.11系列规范主要从ISO/OSI(开放系统互联)基本参考模型中定义的物理层(PHY)和介质访问控制层(MAC)两个层面来制订所使用的无线频率范围、空中接口通信协议等技术规范与技术标准。物理层标准规定无线数据传输的信号特征和调制方法等基础规范；而介质访问控制层标准是在物理层之上定义应用要求规范。

3.1.1 IEEE 802.11 概述

IEEE 802.11在ISO/OSI基本参考模型中所处位置如图3-1所示。这种与上层协议的无关性定义使得基于不同底层网络，高层可以使用相同的协议进行操作。图中逻辑链路层(LLC)子层协议IEEE 802.2[14]规定了错误处理/成帧/流量控制以及网络层服务接口等规范。

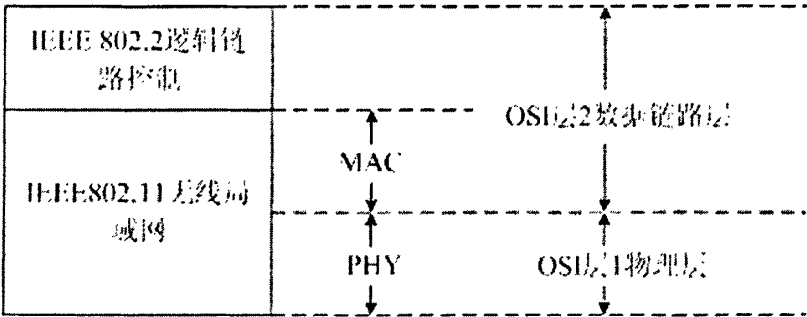


图3-1 IEEE 802.11协议规范模型

IEEE 802.11 MAC层协议详尽地阐述了访问共享物理层信道所应遵循的基本规范(如寻址方式、访问协调等)。由于在无线网络中冲突检测较困难，IEEE 802.11 MAC层协议采用了冲突避免(Collision Avoidance, CA)协议，而不是冲突检测(Collision Detection, CD)[15]。IEEE 802.11物理层的无线媒体(Wireless Media, WM)决定了它具有独特的媒体访问控制机制，各站点以载波侦听多路访问/冲突避免(Carrier Sense Multi-Access/Collision Avoidance, CSMA/CA)[16]的方式共享无线媒体。

不同的物理层在IEEE 802.11标准中有不同的定义，如跳频扩频、直接序列扩频、红外线、窄带调制四种物理层实现技术，都由三个协议功能组成：

- 1) 物理层管理(Physical Layer Management)：为物理层提供管理功能，它与

- MAC层管理相连。
- 2) 物理层汇聚协议 (Physical Layer Convergence Protocol, PLCP)子层：将MAC层协议数据单元(MPDU)映射成适合在PMD系统传送的格式为PLCP协议数据单元(PPDU)，从而降低MAC层对PMD层的依赖程度。
- 3) 物理介质相关(Physical Medium Dependent, PMD)子层：PMD的功能是定义两个或多个工作站之间通过无线介质WM实现物理层数据的发送和接收的方法和特性。为了实现以上功能，PMD需直接面向无线介质，并对数据进行调制和解调。

3.1.2 IEEE 802.11b 网络拓扑结构

本论文侧重于IEEE 802.11b标准协议的研究。其网络体系结构包括几个部分，它们通过交互来为网络上层提供支持站点移动性的无线局域网，如图3-2所示。其中无线站点STA是指接入无线媒质的部分，它包含MAC实体和PHY实体。STA可以是移动的，也可以是固定的。

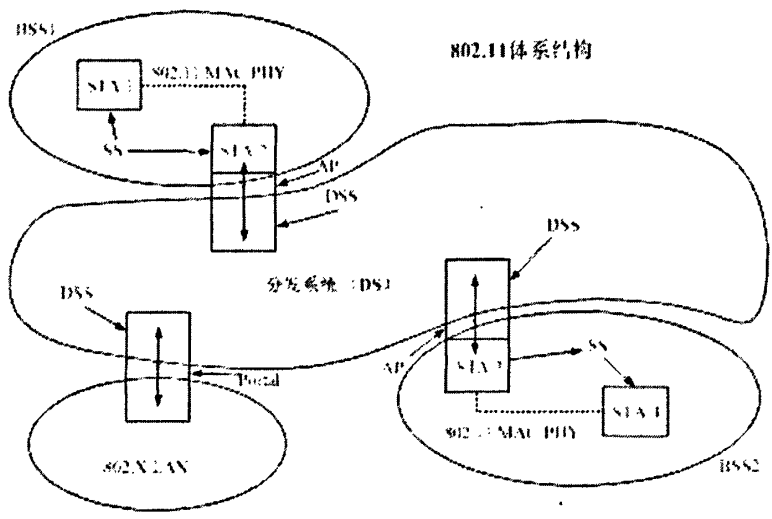


图3-2 IEEE 802.11b体系结构

基本服务集(Basic Service Set,BSS)是IEEE 802.11b局域网的基本组件。可以将BSS理解为一个覆盖区域，在该覆盖区域内BSS的成员站点之间可以保持相互通信，如果一个站点移出了BSS，它就不能和该BSS内的其它成员站点直接通信。

STA和BSS的关联是动态的，STA可以自由地开启、关闭，进入或离开BSS覆盖范围，这也涉及到分发系统服务(Distribution System Service, DSS)的使用。

分发系统(Distribution System, DS)通过互联多个BSS来构成一个扩展的网络，这样可以间接地增加WLAN的覆盖范围。IEEE 802.11在逻辑上把无线媒质WM同分发系统媒质

(Distribution System Medium, DSM)分开。DSM可以是各种类型不同的媒质或其组合；DS也可以采用不同的逻辑结构或其组合。

AP(Access Point)可以作为普通的STA工作，同时也提供DS服务，使BSS可以接入DS。DS和BSS可以组成一个任意大小和复杂的WLAN，被称作扩展服务集(Extended Service Set, ESS)网络，如图3-4所示。

连接802.11网络和传统有线局域网的逻辑结构称作入口(Portal)。入口是一个逻辑点，来自非802.11 LAN的MSDU(MAC Service Data Unit)通过该入口进入DS，如图3-2所示。所有来自非802.11 LAN的数据必须通过一个入口进入802.11 LAN。一个STA有可能同时提供AP和入口的功能。

IEEE 802.11标准有以下三种拓扑结构[17]：IBSS(Independent BSS)网络、Infrastructure BSS网络、ESS网络。

1. IBSS网络

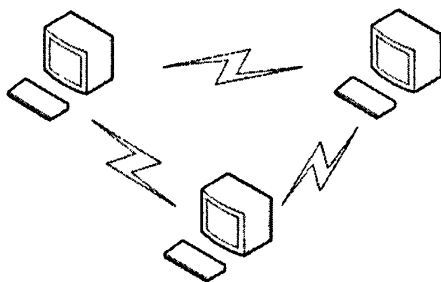


图3-3 IBSS网络

IBSS网络是一个独立的BSS，没有AP作为连接的中心，是最基本的网络类型，也被叫做对等网(Peer-to-Peer)或者非结构组网(Ad Hoc)。如图3-3所示，该网络类型存在的前提是站点之间能够直接通信，一个最小IBSS网络可以只包含两个STA。在IBSS网络中，只有一个公共广播信道，各站点间都可竞争信道，采用CSMA/CA MAC协议。该结构的优点是抗毁性好、建网容易且费用较低。IBSS网络在只需实现无线设备间相互通信的环境中特别有用，常用于组建Ad Hoc网络。IBSS只有一个BSS，没有物理DS，没有入口，没有有线LAN，没有DSS。

2. Infrastructure BSS网络

在Infrastructure BSS网络中，要求有一个无线AP充当中心站点，所有站点对网络的访问均由其来控制。该拓扑结构的弱点是抗毁性差，无线AP的故障容易导致整个网络瘫痪。在实际应用中，WLAN往往与有线网结合起来使用，无线接入点则充当WLAN与有线网的转换器。

3. ESS网络

为了实现跨越BSS范围, IEEE 802.11标准中定义了一个ESS LAN, 如图3-4所示。ESS网络有多个和有线网络连接的AP, 还包括一系列的无线终端。一个ESS是由两个或者多个BSS构成的一个单一子网。近似与蜂窝移动电话中的小区, 但和小区有明显的区别。在一个ESS内不同BSS的AP之间通过DS提供连接和数据传输, 但DS的构成未作定义。

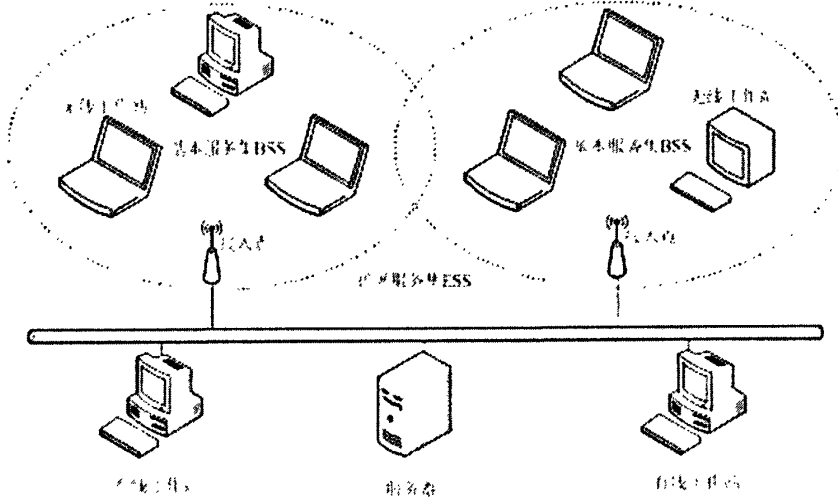


图3-4 ESS网络

3.2 IEEE 802.11服务

3.2.1 服务概述

IEEE 802.11标准没有明确规定DS实施的细节, 而是规定了服务(Service)。这些服务分为两类, 一类是每个STA都应该有的服务, 称为站点服务(Station Service, SS), 包括认证(Authentication)、取消认证(Deauthentication)、加密(Privacy)、MSDU传输(MSDU Delivery)四个服务; 另外一类是DS所独有的DSS服务, 包括关联(Association)、重关联(Reassociation)、分发(Distribution)、集成(Integration)和去关联(Disassociation)五个服务。SS和DSS由MAC子层实施。

分发: 只要工作在基础网络的STA传输数据, 就会使用到分发服务。参考图3-2所示网络, 考虑有数据需要从STA1传到STA4, 数据由STA1发出, 被STA2(输入AP)接收, 该AP把消息传给DS的分发服务。分发服务的任务就是在DS内, 通过合适的路径, 将消息送达适当的目的地。在这个例子中, 消息传到STA3(输出AP), 再通过WM发送到STA4。如果消息的发送者和接收者都在同一个BSS中, 那么输出和输入AP就是同一个AP。

集成: 集成服务是由DS提供, 它将DS连接到一个非802.11网络。

关联：STA在通过AP传输数据消息之前，必须通过关联服务来获得关联识别码 (Association Identity, AID)，实现同AP的关联。关联服务将STA映射到AP，DS使用这种映射信息来完成对STA的消息分发服务。一个STA不能同时与多个AP相关联，一旦关联完成，STA就可以通过AP使用DS服务进行通信。关联总是由STA发起，而不是AP。在同一时刻，AP可以和多个STA关联。

重关联：重关联服务可以使STA从一个BSS转移到处于同一个ESS的另外一个BSS，或者使STA在一个BSS内部移动可以改变关联属性。重关联总是由移动STA发起。

去关联：关联终止时调用去关联服务。在ESS中，去关联服务告诉DS可以丢弃现存的关联信息。通过DS传输消息给一个没有关联的STA是不会成功的。去关联可以由相互关联的任何一方(STA或者AP)发起。去关联是一种告示，而不是请求，去关联不能被关联的任何一方拒绝。

认证：802.11通过认证服务来提供对WLAN接入的控制，相互通信的STA使用这个服务在对方建立自己的身份。如果两个STA之间没有认证，就不能建立关联。

为了无线链接达到有线链接的安全标准，802.11在STA之间提供链接级认证，不提供端到端(消息源到消息目的地)或用户到用户级的认证。在开放认证系统中，任何STA都可以通过认证。802.11也支持Shared Key认证，这种认证就是WEP，在这种认证系统里，身份识别是通过一个共享的、秘密的WEP密钥来完成。管理信息库(Management Information Base, MIB)提供标准的认证过程。在任何时刻，一个STA可以同其它多个STA建立认证关系。

取消认证：终止一个现存的认证关系时调用取消认证服务。在ESS中，由于在关联前必须进行认证，所以取消认证也就引起去关联，取消认证可以由存在认证关系的双方发起。取消认证是一种通知，而不是请求，因此不能被对方拒绝。一旦AP给关联的STA发取消认证通知，那么它们之间关联关系也同时被终止。

加密：在一定范围内，所有兼容的802.11 STA都可能听到特定的PHY上信息的传输，这样，连接到有线LAN的简单无线链接(没有加密)就有可能严重破坏有线LAN的安全性。为了使无线LAN达到有线LAN所固有的安全性能，802.11提供了对消息内容进行加密的功能，该功能由加密服务提供。为了达到有线LAN“同等”的安全性，802.11提供了一种可选的加密策略——WEP。

MSDU传输：站点提供MSDU传输服务，将数据传至目的站点。

3.2.2 各服务之间的关系

一个STA通过WM为与其直接通信的每个STA保存两个状态变量:

- 1) 认证状态变量: 其值表示STA是否认证过。
- 2) 关联状态变量: 其值表示STA是否关联过。

这两个状态变量产生了每一对STA所可能具有的三种本地状态:

- 1) State1: 初始状态, 未认证, 未关联。
- 2) State2: 已认证, 未关联。
- 3) State3: 已认证, 已关联。

源STA和目的STA之间的状态决定了该STA对之间可以传输什么类型的帧: 在State1状态时, 只可以传输Class1类帧; 在State2状态时, 可以传输Class1类和Class2类帧; 而在State3状态时, 可以传输所有类帧。图3-5给出了这些状态和相关的服务之间的关系。

1. Class1类帧: 允许在State1、State2、State3状态中使用。

1) 控制帧

控制帧中属于Class1类帧的有RTS(Request To Send)、CTS(Clear To Send)、ACK(Acknowledgment)、CF-End+ACK、CF-End。

2) 管理帧

管理帧属于Class1类帧的有Probe Request/Response、Beacon、Authentication、Deauthentication、ATIM(Announcement Traffic Indication Message)。其中, 如果Authentication帧认证成功, STA可以传输Class2类帧, 不成功的认证使STA停留在State1。在State2或者State3状态, Deauthentication帧的取消认证通知使STA的状态变为State1, STA在传输Class2类帧之前必须重新认证。

3) 数据帧

该类数据帧的控制FC(Frame Control)域中“ToDS”和“FromDS”位都设为0。

2. Class2类帧: 允许在State2和State3状态中使用。

1) 管理帧

管理帧中属于Class2类帧的有Association Request/Response(该帧成功后可以传Class3类帧, 不成功STA则停留在State2)、Reassociation Request/Response(该帧成功后可传输Class3类帧, 不成功STA则停留在State2)。

如果STA1收到从STA2发来的一个Class2类帧, 而两个STA未通过认证, 并且为单播传

输(Unicast)帧，则STA1向STA2发出一个取消认证帧。

IBSS中可以有任意多个成员，它们之间只能传输Class1和Class2类帧。

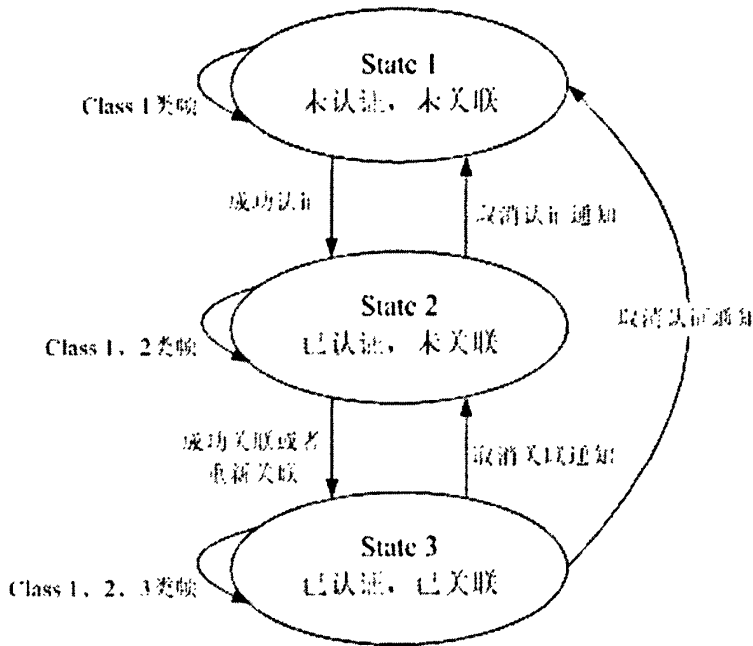


图3-5 状态变量和服务之间的关系

3. Class3类帧：允许在State3状态中使用。

1) 数据帧

该类数据帧的控制FC域中“ToDS”或者“FromDS”比特值设为1，以便可以利用DSS服务传输。IBSS中的服务都是SS。

2) 管理帧

管理帧中属于Class3类帧的有取消认证帧：在State3状态下，取消认证意味着去关联，STA的状态从State3变到State1。

3) 控制帧

控制帧中属于Class3类帧的有PS-Poll帧。

如果STA1收到从STA2发来的一个Class3类帧，而两个STA没有关联但是认证过，并且帧为单播传输(Unicast)帧，则STA1向STA2发出一个去关联帧。如果STA1收到从STA2发来的一个Class3类帧，而两个STA没有经过认证，并且帧为单播传输(Unicast)帧，则STA1向STA2发出一个取消认证帧。

3.3 IEEE 802.11MAC 层逻辑结构

IEEE 802.11 MAC 子层的功能是为用户提供可靠的数据传输，实现对共享媒质访问的

公共控制。它提供 2 种访问机制：分布协调功能(Distributed Coordination Function, DCF)是 802.11 的基本访问机制，作用于 BSS 和基本网络构架中，支持竞争型异步业务,可在所有站点中实现；集中控制访问机制，即点协调功能(Point Coordination Function, PCF)支持无竞争型时限业务和无竞争异步业务,PCF 的功能是以 DCF 为基础。其逻辑结构如图 3-6 所示。

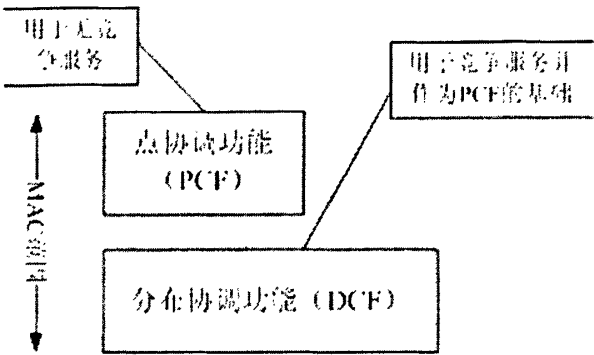


图 3-6 IEEE 802.11 MAC 层逻辑结构

802.11 MAC 子层的另一个功能就是对数据传输提供保护，保证所有数据的安全性达到有线 LAN 的要求。另外，802.11 MAC 子层还提供了两个强大的功能：CRC 校验和分段。

为提高成功发送的可能性，802.11MAC还提供了重新排序功能。只有在节电方式工作下的站点，且不处于激活状态时，才可先将MAC服务数据单元(MSDU)缓存起来，等站点激活时再发送出去，并对缓存数据进行重新排序。

3.4 分布协调功能(DCF)

DCF 使用 CSMA/CA 协议和随机退避机制，从而允许相互兼容的 PHY 自动共享媒质。CSMA/CA 协议用于减少 STA 进入媒质时发生碰撞的可能性。在媒质忙之后的空闲阶段，是最有可能发生冲突的，因为有可能有多个 STA 在等着使用媒质，这时就需要一个随机退避过程来解决竞争冲突。

802.11 同时使用物理载波侦听和虚拟载波侦听机制来实施载波侦听，其中虚拟载波侦听通过发布媒质预留声明信息来完成。所有在源 STA 或目的 STA 发送范围内的 STA 都能获得媒质的占用时间，这样 STA 即使不能从源 STA 接收数据帧，也能知道媒质的使用情况。

RTS/CTS 控制帧也具有避免传输冲突的功能，如果发送 RTS 的源 STA 没有能够接收到 CTS 帧，该 STA 就会重复发送 RTS。成功发送一个帧以后，需要在媒质空闲时进行随机退避，退避过程结束后才能发送其它帧。RTS/CTS 机制不能用于广播和多播数据帧的传输。

3.4.1 DCF 中的关键技术

1. 载波侦听机制

载波侦听机制包括物理载波侦听机制和虚拟载波侦听机制,用来决定当前媒质状态。当任一机制报告媒质忙,则认为媒质忙,否则认为媒质空闲。

物理载波侦听由 PHY 提供,具体由 PHY 规范定义。物理载波侦听结果由检测接收信号能量的强弱来确定。

虚拟载波侦听由 MAC 子层提供,使用网络分配矢量(Network Allocation Vector, NAV)。NAV 根据 RTS/CTS 帧或在竞争期间(Contention Period, CP),除 PS-Poll 帧外的所有帧的 Duration/ID 域的预留时间信息,对未来媒质上通信量进行预测。NAV 可以看作一个计数器,其值以固定速率递减,若减小到 0,虚拟载波侦听就指示媒质空闲;如果非 0,则表示忙。

2. 帧间隔

帧间的时间间隔称作帧间隔(InterFrame Space, IFS)。所有使用载波侦听功能的 STA 只有在检测到信道空闲时间大于 IFS 后,才认为信道空闲。MAC 层定义了四种 IFS: SIFS(Short IFS)、PIFS(PCF IFS)、DIFS(DCF IFS)、EIFS(Extended IFS),分别表示对无线媒质访问时不同的优先级,如图 3-7 所示。四种 IFS 定时都与 STA 的数据速率无关,而由物理层参数确定的,并且对于给定的 PHY,其值是固定不变的。

SIFS 用于 ACK 帧、CTS 帧、MSDU 的第二个或后续 MPDU 分段突发帧,以及 STA 在无竞争期间(Contention Free Period, CFP)对 PCF 轮询的应答,点协调器(Point Coordinator, PC)传输任意类型帧时也可使用。SIFS 间隔从空中接口看是从上一个帧的最后一个符号到下一帧的第一个符号之间的时间间隔。在 802.11 的实现中 SIFS 在标准值上下波动范围不能超过 PHY 所使用的 aSlotTime 值的 10%。SIFS 是最短的 IFS,在 STA 已经获取媒质控制权并且需要保持它以完成帧交换序列时使用,可以阻止需要等特更长媒质空闲时间的其它 STA 对媒质的占用,使得正在进行的帧交换序列优先完成。

PIFS 只能由工作在 PCF 下的 STA,在 CFP 的起始时间为获得媒质访问权时使用,除此之外,PIFS 不用于其它任何场合。使用 PCF 的 STA,如果其载波侦听机制在 PIFS 时间片的边界处认为媒质空闲,则可以进行无竞争传输。

DIFS 是工作在 DCF 下的 STA,为发送 MPDU 和 MAC 层管理协议数据单元 (MMPDU) 而使用的时间间隔。使用 DCF 的 STA,接收到一个帧后的随机退避时间结束时,如果其载波侦听机制在 DIFS 时间片的边界处认为媒质空闲,则可以进行有竞争传输。

EIFS 用于工作在 DCF 下的 STA。当 PHY 指示未正确接收到的一个完整的 MAC 帧或者 FCS 校验出现错误时，即使 STA 认为媒质空闲，在此之后的一个 EIFS 内都不会进行数据发送。当侦听到错误帧并且 PHY 显示媒质空闲后，不管虚载波侦听的状态，EIFS 开始计时。在 EIFS 阶段，只有在接收到一个无错帧后，EIFS 才会终止，否则一直到 EIFS 完全计时结束。

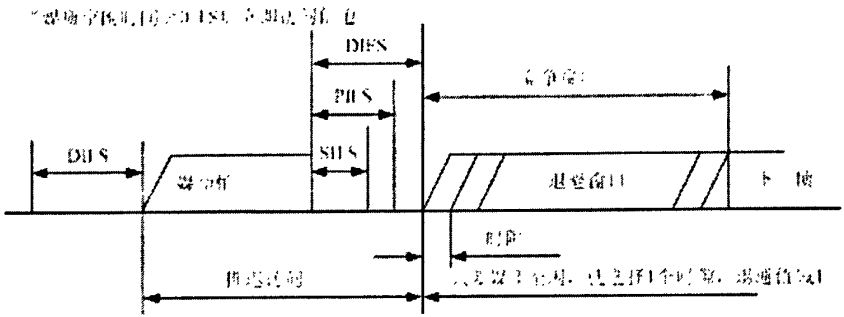


图 3-7 不同 IFS 间的关系

3. 随机退避时间

当 STA 希望传输数据帧和管理帧时，使用载波侦听机制看媒质是否空闲，如果媒质忙，则开始延时，直至媒质空闲时间为 DIFS(最近一次检测到的帧被正确接收)或者 EIFS(最近一次检测到的帧未被正确接收)。如图 3-7，STA 在这个延时过程结束后执行一个随机退避过程(如果此时退避计时器为零则不需要执行这个退避过程)以推迟发送。该过程最小化了多个 STA 由于同样原因延迟之后进行媒质竞争时出现冲突的可能性。

退避时间按照下式计算： $BackoffTime = Random() * aSlotTime$

其中，Random()是[0, CW]之内服从均匀分布的随机整数，CW 是介入两个 PHY 参数 aCWmin(最小竞争窗口)和 aCWmax(最大竞争窗口)之间的整数，时隙时间 aSlotTime 是相应 PHY 对应的参数值。

竞争窗口(Contention window,CW)的初始值为 aCWmin。每一个 STA 都有一个短重传计数器(STA Short Retry Count, SSRC)，一个长重传计数器(STA Long Retry Count, SLRC)，它们的初始值都为零。当与任何 MSDU 相关的短重传计数器递增时，SSRC 递增；当与任何 MSDU 相关的长重传计数器递增时，SLRC 也递增。当不成功传输出现而引起任意一个计数器增加时，CW 就会赋值为可取值序列中的下一个值，CW 达到 aCWmax 后，就会保持在 aCWmax，直到被复位。CW 序列的值为(CW₀、CW₁、...CW_i、CW_{i+1}、...CW_n)，其中：CW₀ = aCWmin，CW_{i+1} = 2CW_i + 1，CW_n = aCWmax。

如果站点的 SLRC 达到 aLongRetryLimit(长重传次数限制)，或 SSRC 达到 dot11ShortRetryLimit(短重传次数限制)，在成功传输一个 MPDU 或 MMPDU 之后，CW 就

复位到 aCWmin。如果收到一个 CTS 帧或者收到 ACK 帧(传送的 MDPU 或 MMPDU 的长度小于 dot11RtsThreshold), 或者传输一个以 Address1(地址 1)为组地址的帧, SSRC 都会复位到零。如果收到一个 ACK 帧(传送的 MDPU 或 MMPDU 的长度超过 dot11RtsThreshold), 或者传输一个以 Address1 为组地址的帧, SLRC 都会复位到零。

3.4.2 DCF 的接入过程

1. 基本接入机制

基本接入机制是 STA 决定是否可以进行传输时所用到的核心机制。如图 3-7 所示, 在没有 PC 的情况下; 或者工作在 PCF 下处于 CP 阶段; 或者 DCF 下的 STA 确定媒质空闲时间大于等于 DIFS; 或者不小于 EIFS(接收到一个 FCS 校验错误的 MAC 帧后), 就会开始一个退避过程(CFP 阶段除外)。

2. 退避过程

当媒质忙或发送不成功时就需要一个退避过程。为开始一个退避过程, STA 需要按照 3.4.1 节中的方程给退避定时器(Backoff Timer)赋值。所有的退避时隙紧跟 DIFS 或 EIFS 之后, 在 DIFS 或 EIFS 计时内, 媒质必须空闲。

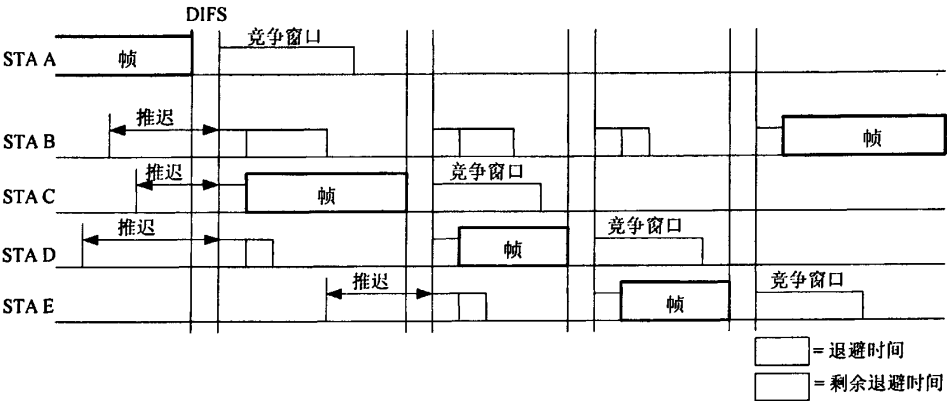


图 3-8 退避 过程

如图 3-8 所示, 五个 STA 都有数据需要传输。STA A 首先发送数据完毕, 经过 DIFS 后开始一个竞争窗口; 在这个 DIFS 之后, 所有的 STA 开始退避过程, STA C 的剩余退避时间最小, 则首先获得媒质传输数据, 在 STA C 传输数据的时候, 其它 STA 的退避过程挂起; 当 STA C 发送完数据, 间隔一个 DIFS 之后, 所有的 STA 都开始竞争媒质, 这时 STAD 退避过程首先结束, 从而获得媒质; 之后依次是 STAE 和 STAB 获得媒质。

在每个退避时隙中, STA 需要不停地检测是否有其它的 STA 在使用媒质, 如果没有, 则其退避时间递减一个 aSlotTime; 如果在一个退避时隙中发现媒质重新变忙, 则 STA 不

减去这个时间片，将退避过程挂起，当媒质空闲时间重新达到 DIFS 或 EIFS 后，再次激活退避过程。当退避时间为零时，传输马上开始。

任意一种 MoreFragments 位为 0 的 MPDU 帧传输(数据帧、PS-Poll 控制帧、管理帧)完成之后，即使没有其它 MSDU 帧在排队等待使用媒质，退避过程也立即开始。在接收到 ACK 帧之后，或者在传输不成功情况下，在规定的 ACK 应答时间间隔过去之后仍然没有收到 ACK，立即开始退避过程。

总之，当多个 STA 要求发送数据时，都要进入退避过程，拥有最小退避时隙的 STA 将最先获得媒质的使用权。

3. 差错恢复过程和重传限制

差错恢复由发起帧交换的 STA 完成。好多情况引发差错，例如：RTS 帧之后没有 CTS 的应答，这可能是因为在发送 RTS 或 CTS 帧时同其它的传输发生冲突；或者接收到 RTS 的 STA，发现自己的虚载波侦听指示媒质现在忙。发现传输差错之后，就应该在适当的时间进行数据重传，直到传送成功或达到重试次数上限。STA 对每个等待发送的 MSDU 或者 MMPDU 分别设置一个短重传计数器和长重传计数器。这些计数器相互之间是独立的。

发送完 RTS 之后，STA 开始等待 CTS。如果 RTS 发送失败，则 STA 对 MSDU 或者 MMPDU 的短重传计数器 SSRC 加 1，直到 MSDU 或者 MMPDU 的短重传计数器达到 dot11ShortRetryLimit(短重传次数域值)。

发送需确认的帧之后，STA 开始等待 ACK。当帧长不大于 dot11RtsThreshold 的 MSDU 或 MMPDU 帧发送失败，相应帧的短重传计数器 SSRC 将加 1，如果发送成功则短重传计数器 SSRC 复位置零。当帧长大于 dot11RtsThreshold 的 MSDU 或 MMPDU 的帧发送失败，相应帧的长重传计数器和 SLRC 将加 1，如果发送成功则这些计数器复位置零。所有的 MSDU 或 MMPDU 帧发送之后(不管长度如何)，如果没有接收到表示传输成功的 ACK 确认一次或多次，则置数据或管理帧中的 Retryfield 为 1。

发送失败后，进行帧的重传，直到短重传计数器达到 dot11ShortRetryLimit 或长重传计数器达到 dot11LongRehyLimit 时重传结束，将该帧丢弃。

4. 设置和重置 NAV

STA 接收到有效帧以后按照该有效帧中的 Duration/ID 域对 NAV 进行更新(只有在该 NAV 值比本地当前 NAV 值大，并且这个有效帧的后继帧的目的地不是该 STA 时才更新)。图 3-9 表示了一些 STA 可能接收到 RTS 而另外一些 STA 只接收到 CTS 帧。

如果 STA 使用 RTS 中的信息作为更新其 NAV 的基础，在出现 RTS 帧的物理层接收结束指示原语 PHY-RXEND.indication 后，如果经历 $(2 * aSIFSTime) + (CTS_Time)$

PHY-RXSTART.indication 情况下可以对其 NAV 进行复位。CTS 帧时间长度 CTS_Time 应该按照 CTS 帧长及最近接收到的 RTS 帧中的数据速率计算。

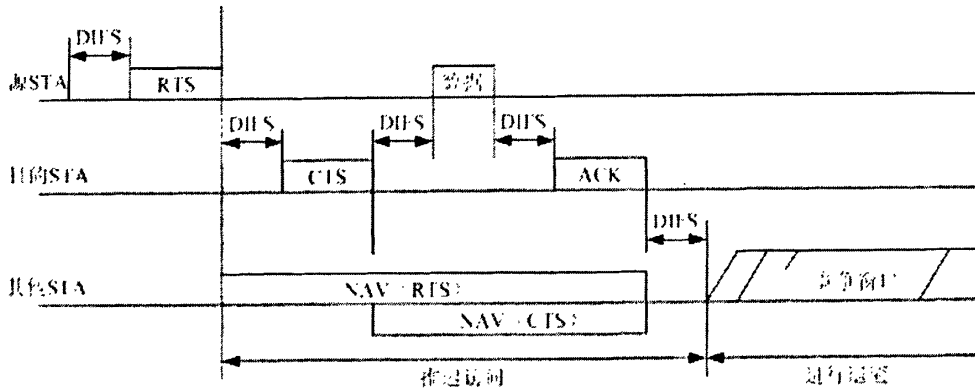


图 3-9 NAV 的设置

5. STA 对媒质的竞争

SIFS 可以用来提供高效的 MSDU 传输。一旦竞争到媒质,那么 STA 就会一直发送帧,直到出现下列三种情况之一:单个 MPDU 或 MMPDU 的所有分段帧都发送完毕;期望的一个 ACK 帧没有接收到;该 STA 由于驻留时间界限而被限制发送。如果是因为一个 ACK 帧没有出现,或者是因为受到驻留时间界限而停止了发送,当 STA 再次获得一个发送机会就会重新从上次停止的地方开始发送。

当源 STA 发送完一个帧,就会释放媒质并立即使听媒质等待 ACK 帧。在收到目的 STA 发送的 ACK 帧,附加一个 SIFS 后,发送另外一个分段帧。如果源 STA 获得 ACK 后,由于驻留边界的限制而没有足够的时间发送另外一个帧和对应 ACK 帧,那么该源 STA 就会在下一个驻留时间的开始竞争媒质。如果源 STA 没有接收到 ACK 帧,就会在经过退避过程和竞争过程后重新发送传送失败的 MPDU 或另外一个可行的 MPDU。

图 3-10 表示了使用 SIFS 传输多段 MSDU 的情况。

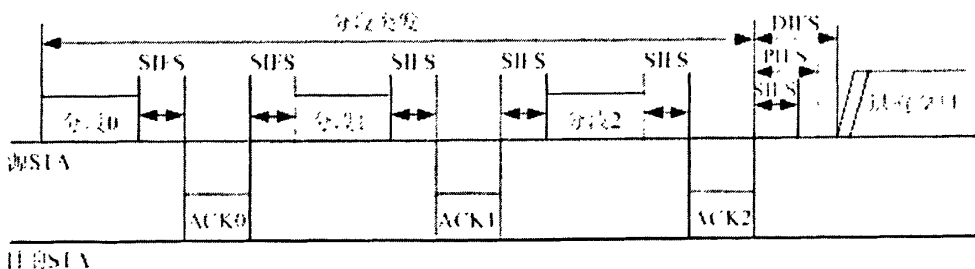


图 3-10 使用 SIFS 传输多段 MSDU

STA 在以下情况会经过 SIFS 间隔后可以数据进行发送:

- ◆ 目的 STA 收到一个需要 ACK 的帧。
- ◆ 源 STA 收到 ACK，并且有该 MSDU 的剩余分段需要发送，且在其驻留时间边界之前有足够的时间发送下一个分段和接收一个 ACK 帧。

同时也遵守以下规则：

- ◆ 如果 STA 发送了 MSDU 的最后一个分段，那么在接收到这个帧的 ACK 帧之后，该 STA 在未进行退避过程之前不进行其它发送。
- ◆ 如果一个 MSDU 被成功发送，或所有重传次数已达门限，而该 STA 还有 MSDU 要发送，则该 STA 需要一个退避过程。
- ◆ 只有没有被应答的帧才会被重传。

6. RTS/CTS 的使用

下面是使用 RTS/CTS 机制对 MSDU 或 MMPDU 进行分段传输的描述，见图 3-11。每一个帧中都有下一次传输所需的时间信息，RTS 中的 Duration/ID 域用于更新 NAV，表示媒质忙直到 ACK₀ 的结束；而 CTS 中的 Duration/ID 域也用于更新 NAV，也表示媒质忙直到 ACK₀ 的结束；分段 0 和 ACK₀ 都含有信息用于更新 NAV，表示媒质忙直到 ACK₁ 的结束，这是通过使用数据帧和 ACK 帧中的 Duration/ID 域而获得的，这个过程一直持续到最后一个分段，这个帧中的 Duration/ID 域值是一个 ACK 时间加上一个 SIFS 时间，并且这个 ACK 帧中的 Duration/ID 域被置零。在 RTS/CTS 开始帧交换之后，尽管随后的帧有可能大于 dot11RTSTreshold，也不需要产生另外的 RTS/CTS 帧进行握手。

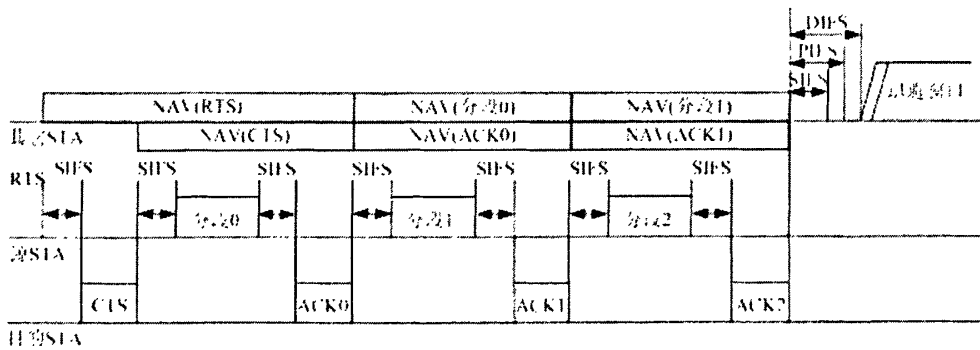


图 3-11 RTS/CTS 机制与 MSDU 的分段传输

目的 STA 收到源 STA 发出的 RTS 帧后，经过 SIFS，如果其 NAV 显示现在媒质空闲，就会发 CTS 帧对源 STA 进行应答；如果此时媒质不空闲，则不作应答。CTS 帧中的 RA(Recive Address)为 RTS 帧中的 TA (Transmission Address)。CTS 帧中的 Duration/ID 域应该为接收到的 RTS 帧相应域中的值减去 aSIFSTime 及根据 RTS 帧所使用的数据速率计算出来用于发送该 CTS 帧的时间。

发送完 RTS 帧, STA 就会在 PHY-TXEND.confirm 之后开始一个 CTSTimeout 的等待, 如果在这个间隔内没有出现 PHY-RXSTART.indication 信号, 该 STA 就会判断发送 RTS 失败。CTSTimeout 到期后, STA 开始一个退避过程继续竞争信道。如果在这个时间间隔内出现了 PHY-RXSTART.indication 信号, STA 就会等待 PHY-RXEND.indication 的出现, 进而判断发送 RTS 是否成功。如果接收到的 CTS 帧有效, 则认为应答成功, 就可以开始数据帧的发送。如果发现不是 CTS 帧, 不管是不是其它有效帧, 都认为 RTS 发送失败, 并且开始一个退避过程继续竞争信道。

3.4.3 MPDU 的传输过程

只有当定向 MPDU 的长度大于 dot11RtsThreshold 所规定的门限值时, STA 才采用 RTS/CTS 机制进行传输。dot11RtsThreshold 是 MAC 管理信息库 MIB 中的管理参数, 可以被 MAC 层的管理实体(Layer Management Entity, LME)设置和查询, 该值为 0 表示在所有 MPDU 帧的传输前都需要 RTS/CTS 机制。

如果 dot11RtsThreshold 值比所有的 MPDU 长度值都大, 则所有 MPDU 的传输都不需要 RTS/CTS 机制。如果使用 RTS/CTS 机制, 在 CTS 帧和一个 SIFS 间隔之后, 开始异步数据帧的传输。如果不使用 RTS/CTS 机制, 异步数据帧的传输根据基本接入过程的规则进行。无论使用或不使用 RTS/CTS 机制, 目的 STA 都要对异步数据帧进行应答(ACK)。

在没有 PCF 的情况下, ToDS 位为零的 STA 要进行 MPDU 广播或组播传送, 只能使用基本接入过程。不管广播或组播帧的长度是否大于门限值, 都不使用 RTS/CTS 机制, 并且, 任何 STA 收到广播或组播帧以后, 都不进行 ACK 响应。ToDS 位为 1 的 STA 要进行 MPDU 广播或组播传送, 除了要遵循 CSMA/CA 的基本接入过程, 还要使用 RTS/CTS 机制。这是因为 MPDU 是发给 AP 的。广播和组播 MPDU 将会发送到整个 ESS。除非 ToDS 位为 1, 否则在 MAC 层里没有对广播或组播帧进行差错恢复的机制。由于有可能因为干扰、冲突、或媒质时变而造成帧丢失, 相对于定向单播传输, 这种传输的可靠性要低。

3.4.4 ACK 应答过程

在成功接收到一个需要应答的帧和一个 SIFS 之后, 不管媒质的忙闲状态, STA 都会发送 ACK 帧, 如图 3-11 所示。

源 STA 在等待 ACKTimeout(ACK 超时)时间后仍然没有接收到 ACK 帧, 就会认为这个 MPDU 发送失败。在发送了一个需要应答的 MPDU 帧之后, STA 就从 PHY-TXEND.

confirm 信号开始一个 ACKTimeout 时间的等待。如果在这个时间内没有出现 PHY-RXSTART.indication 信号, STA 就会判断发送 MPDU 失败, 在 ACKTimeout 到期以后开始一个退避过程。如果在这个时间间隔内出现了 PHY-RXSTART.indication 信号, STA 就会等待 PHY-RXEND.indication 的出现, 以判断发送 MPDU 是否成功。在 PHY-RXEND.indication 之后, 如果判断接收到的 ACK 帧有效, 就认为 MPDU 发送成功, 就可以按照正常的帧交换顺序开始后续帧的发送, 如果没有后继帧需要发送就结束本次传输。如果不是 ACK 帧, 不管是否是其它有效帧, 都认为 MPDU 发送失败, 开始一个退避过程, 同时对接收到的帧进行处理。唯一的例外是对于收到 PS-Poll 帧的 STA 所发出的合法数据帧被认为是对 PS-Poll 帧的应答。

3.4.5 副本检测和恢复

由于协议规定了 MAC 层的应答和重传机制, 这就有可能出现一个帧被多次接收, 多余的副本帧应该在目的 MAC 里被滤除。副本滤除是利用了 MAC 报头的序列控制域(包含序列号和分段号), 同一个 MSDU 的所有 MPDU 都有相同的序列号, 不同的 MSDU 以较高的概率拥有不同的序列号。序列号由源 STA 按照整数的增序产生。

接收 STA 应该对最近接收到的帧设置一个三元列表, 元素从 MAC 报头提取获得, 含如下三项 <Address2,sequence-number,fragment-number>。接收 STA 应该有最新 <Address2,sequence-number>对的入口, 并且把最新收到的、符合这个对的 fragment-number 保存下来。

目的 STA 检查收到的帧, 发现如果帧中控制域的 Retry 位为 1, 且 <Address2,sequence-number,fragment-number>和三元列表中的某项相同, 则滤除此帧。根据这种匹配检查方法, 也有可能把一个帧错误地滤除, 不过这个概率很小, 结果会丢失一个帧(同其它 LAN 协议中的 FCS 错误而引起的帧丢失相似)。目的 STA 会对所有接收成功的需要应答的帧进行 ACK 响应, 不管这个帧是否因为副本滤除而被丢弃。

3.5 点协调功能(PCF)

PCF 通过轮询和应答机制为无竞争帧传输服务, PC 驻留在 AP 中, 所有 STA 本质上都服从 PCF 的媒质接入规则(因为这些规则是基于 DCF 的), 并且在每一个 CFP 的开始设置各自的网络分配矢量(NAV)。在 PCF 工作模式下, 所有 STA 都能在一个有 PC 工作的 BSS 中正常运作。如果是和一个点协调 BSS 关联, 这些 STA 就能在 PCF 控制下接收到所

有发送出来的帧。STA 也可以选择是否对由 PC 发出的无竞争轮询帧(CF-Poll)进行响应。能够响应 CF-Poll 的 STA 称作是无竞争可轮询的。无竞争可轮询 STA 和 PC 在 CFP 中不使用 RTS/CTS 机制。

当被 PC 轮询时，可轮询 STA 可能只发送一个 MPDU，这个 MPDU 可以发送到任意目的地(包括 PC)，也可以稍带对 PC 发出帧的应答信息。如果该数据帧没有被应答，STA 只有再次被 PC 轮询才会重传该帧，或者决定在 CP 期间重传。如果 CFP 内发送帧的目的 STA 不可轮询，该目的 STA 就使用 DCF 方式下的应答规则进行应答，而且 PC 保持对媒质的控制。PC 使用无竞争帧可能仅仅是为传输帧到 STA(包含不可轮询的 STA)，但绝不会轮询不可轮询的 STA。

PC 在 CFP 期间重传未得到应答的帧时，可以先执行一个退避过程。轮询列表由 PC 维护，当某一个关联识别码(AID)再次处于轮询列表顶端时，PC 可以重传未被该 AID 应答的帧。

在同一物理信道上有多个点协调 BSS 存在，并且这些 BSS 在空间上有重叠部分时，这些相互独立的 PC 控制下的 PCF 传输就有可能发生冲突。如图 3-6 所示，PCF 构建于以 CSMA/CA 为基础的 DCF 之上，并利用 DCF 提供的接入优先级规则实现接入。一个活动 PC 被内置于 AP 中，同时也就限制了 PCF 操作只能在 Infrastructure BSS 中。AP 可以通过设置 MLMEStart.request 原语中 CF 参数集中的 CFPMaDuration 参数为非零值来激活 PCF 工作模式。

3.5.1 CFP 的结构和定时

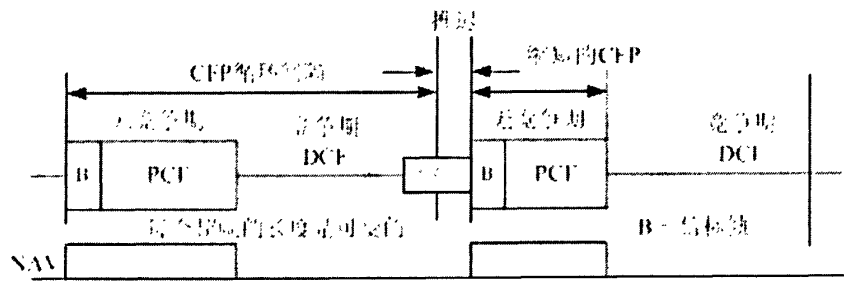


图 3-12 CFP 和 CP 的交替示意图

PCF 控制在 CFP 中帧的传输，CFP 与随后的 CP 交错进行一起组成超帧，DCF 控制 CP 期间帧的传输，如图 3-12 所示。PCF 发起数据传输的等待时间间隔称为 PIFS，PIFS 介于 SIFS 和 DIFS 之间，因此 PCF 比 DCF 的优先级高。一个含有交互传输指示信息(Delivery Traffic Indication Message, DTIM)的 Beacon 帧表示了 CFP 的开始(同时也是一个超帧的开

始), CFP 以特定的周期重复出现, 该频率和下文描述的 Beacon 间隔同步。

PC 周期性地产生信标帧, 下一个 Beacon 帧到来的时间间隔称为目标信标发送时间(Target Beacon Transmission Time, TBTT)。PC 根据 CF 参数集中 CFPRate 参数决定 CFP 的重复频率, CFPRate 以 DTIM 间隔为单位, 通过 Beacon 帧 CF 参数集中的 CFPPeriod 域发送到 BSS 的其它 STA。只有 PC 发送的 Beacon 和 Probe Response 帧才含有 CF 参数集。

CFP 的长度由 PC 控制, 其最大值由 CF 参数集中的 CFPPMaxDuration 决定, 最大长度和实际长度(PC 通过发送 CF-End 或 CF-End+ACK 控制帧来指示)都不限制为 Beacon 间隔的整数倍。CFPPMaxDuration 最小值为两个 MaxMPDUTime 加上发送开始 Beacon 帧和 CF-End 帧所需的时间。

如果 CFP 长度大于一个 Beacon 间隔, PC 会在 CFP 的适当时刻(由于媒质忙而产生的传输时迟之后)发送 Beacon。在所有表示 CFP 开始的 Beacon 帧的 CF 参数集中, CFPPDurRemaining 参数都是一个非零值, 该值以 TU(Time Unit)为单位, 指定从发送 Beacon 到该 CFP 结束的最大持续时间。在 CP 内 PC 发送的 Beacon 帧中的 CFPPDurRemaining 域应该为 0。在媒质忙的情况下 Beacon 发送会被推迟, 可以根据延迟时间值预先减少 CFP 的持续时间。在由于 DCF 传输而引起的媒质忙情况下, Beacon 应该延迟使媒质完成该 DCF 帧交换。在 Beacon 被延迟发送时, 表示 CFP 开始的 Beacon 帧中 CFPPDurationRemaining 参数值应该保证该 CFP 的结束时间不晚于 TBTT 与 aCFPPMaxDuration 之和的时间值。

3.5.2 PCF 的接入过程

一个 BSS 中有一个 AP, 且 PC 控制下的轮询规则是无竞争传输协议的基础。PC 在 CFP 的开始获得媒质的控制权, 由于 PCF 发起数据传输的等待时间间隔 PIFS, 比 DCF 控制下的两次传输之间的间隔(DIFS)短, 所以在整个 CFP 期间, PC 试图保持对媒质的控制。一个 BSS 中所有 STA(除了 PC)在 CFP 的开始时间都将其 NAV 置为 CFPPMaxDuration, 这就可以通过禁止 STA(无论是否可轮询)非轮询的传输来防止大部分的竞争。在 CFP 期间发送可以通过 Data+CF-ACK、CF-ACK、Data+CF-ACK+CF-Poll(只能由 PC 产生)或者 CF-ACK+CF-Poll(只能由 PC 产生)帧进行应答, 这样避免了单一 ACK 帧传输的开销。在 CFP 期间, 非可轮询或没有被轮询到的可轮询 STA 的应答应使用 DCF 的 ACK 过程。

在 CFP 的开始, PC 对媒质进行侦听, 如果媒质空闲时间长达一个 PIFS, PC 将发送一个含有 CF 参数集和 DTIM 的 Beacon 帧。在发出初始化 CFP 的 Beacon 帧后, PC 将等待至少一个 SIFS, 然后发送如下的一个帧: 数据帧、CF-Poll 帧、Data+CF-Poll 帧或者 CF-End

帧。如果 CFP 为空, 例如 PC 处没有缓冲的数据帧, 也不需要轮询, 则 PC 在初始化 CFP 的 Beacon 帧之后立即发送一个 CF-End 帧。

STA 从 PC 处接收到一个正确帧后, 应该在一个 SIFS 以后, 按照 3.5.3 节定义的传输过程进行应答。如果该 STA 不可轮询, 接收到有效帧后以 ACK 帧进行应答。

除了 PC, 每个 STA 都会在 CFP 开始的 TBTT 时刻将它们的 NAV 预置为 CFPMaDuration。每一个非 PC 的 STA 都会根据收到的 Beacon 帧(包括从重叠的 BSS 接收到的 Beacon 帧)的 CFPDurRemaining 参数值更新其 NAV。这些措施防止了 STA 在 CFP 期间对媒质的控制, 也降低了隐藏站点风险(在 CFP 期间, 隐藏 STA 认为媒质空闲时间大于 DIFS, 故可能会破坏正在进行的传输)。

正在加入 BSS 的 STA 在进行任何发送前, 都应先根据接收到的 Beacon 帧或 Probe Response 帧中的 CFPDurRemaining 参数值对其 NAV 进行更新。PC 将会在 CFP 结束时发送一个 CF-End 或 CF-End+ACK 帧, STA 接收到任何一个 BSS 发送的这种帧都应该复位其 NAV。

3.5.3 PCF 的传输过程

PCF 下帧传输的典型过程就是: AP/PC 发送帧和接收帧到 AP/PC 交替进行。在 CFP 期间, 这种发送的顺序, 以及被允许在任何给定时刻发送帧到 PC 的 STA, 都应该受到 PC 控制。

在 DCF 下的发送数据帧使用的子类型包括简单数据帧或 Null 帧。在 PCF 下发送的数据帧或轮询应答的数据帧应该按照如下规则:

- ◆ Data+CF-Poll、Data+CF-Ack+CF-Poll、CF-Poll、CF-Ack+CF-Poll 只能由 PC 发出。
- ◆ Data、Data+CF-Ack、Null、CF-Ack 可由 PC 或任意可轮询 STA 发出。
- ◆ PC 在初始化 CFP 的 Beacon 帧和结束 CFP 的 CF-End 帧之间以 SIFS 间隔进行传输

如果收到的数据帧子类型是 Data、Data+CF-Ack、Data+CF-Poll、Data+CF-Ack+CF-Poll, STA 会认为这个数据帧是报告给 LLC 的。无竞争可轮询 STA 对收到的所有数据帧的子类型域中每一位都进行“解释”, 只有子类型为 Data+CF-Ack、Data+CF-Ack+CF-Poll 或者 Data+CF-Poll 才会检查其帧体部分。

在 CFP 期间, PC 可以传输数据或管理帧给非可轮询、非节能 STA。这些 STA 在接收到帧之后的一个 SIFS 后需要用 DCF 方式进行应答。在 CFP 期间, PC 也可以传输广播或组播帧。由于启动 CFP 的 Beacon 帧中含有 DTIM, 如果相关联的 STA 使用节能模式, 在

一个含有 TIM(其中的 DTIM 域为 0)的 Beacon 后,等待传输的广播或组播帧就会马上开始发送。

可轮询 STA 接收到任意一个包含 CF-Poll 的数据帧以后,就会发送一个 Data+CF-ACK 对 CF-Poll 进行应答。PC 可以使用 CF-ACK 对一个接收帧进行应答,尽管含有 CF-ACK 的数据帧有可能是发送给其它 STA 的,而不是发送给正被应答的 STA。在发送帧给 PC 之后的一个 SIFS, STA 就会对由 PC 发送的任意类型帧进行解析。如果一个非可轮询 STA 接收到一个数据帧后,就会在一个 SIFS 之后发送一个 ACK 帧,而不会对 CF-Poll 帧作任何解析。

如果 PC 发送完任意一种类型的 CF-Poll 帧,经过 SIFS 之后没有一个可轮询 STA 对其进行响应,或者,在 PC 发送完需要应答的帧后,经过 SIFS 后,没有非可轮询 STA 发送 ACK 帧对此响应,PC 就会在它的上一次发送结束后,等待 PIFS,重新获得媒质的控制并且发送下一个帧。可轮询 STA 对含有其 MAC 地址的 CF-Poll 都应该响应。如果此时 STA 没有帧需要发送,就会发送一个 Null 帧。如果没有帧需要发送但是需要对接收到的包含 CF-Poll 的帧进行应答,就发送一个 CF-ACK 帧。

当从 CFP 开始的 Beacon 帧开始之后到 CFPDurRemaining 时间耗尽,或者 PC 没有帧需要发送且没有 STA 需要轮询,则该 CFP 结束。这两种情况 PC 都会在 CFP 结束时发送一个 CF-End 帧,如果在发送 CF-End 之前收到一个需要响应的帧,那么 PC 就会发送一个 CF-End+CF-ACK 帧。所有的 STA 在收到 CF-End+ CF-ACK 帧或 CF-End 帧之后都会对其 NAV 进行复位,以便在 CP 阶段竞争媒质。

3.6 分段(Fragmentation)

MAC 层可以对定向 MSDU 或者 MMPDU 执行分段和分段重组,分段和重组机制允许分段的重传。

所有分段的长度应是相同字节数,除了最后一个分段长度可能会短一些。同样,所有分段的长度应是偶数字节,除了最后一个分段可以是奇数字节。所有分段长度都由 aFragmentationThreshold 和刚创建的分段 MPDU 决定,不应超过 aFragmentationThreshold。如果该分段经过 WEP 扩展处理,其长度可能会大于 aFragmentationThreshold。对于 STA 应能够接收任意长度的分段。

如果 STA 的某个分段需要重传,则在该分段所属的 MSDU 或者 MMPDU 的生命周期内,该分段帧实体内容和长度保持不变。在发送 MSDU 或者 MMPDU 时,每个分段的帧

顺序号保持不变,按段顺序号从低到高的顺序进行发送。第一个分段的段顺序号值为 0,随后传送的每个分段的段顺序号依次增 1。帧控制域的更多分段(More Fragments)位用来表示该分段是否是最后一个分段。

aMaxTransmitMSDULifetime 表示 MSDU 允许发送的最长持续时间。STA 为每个 MSDU 的发送维护一个发送定时器。定时器从发送 MSDU 的第一个分段开始计时,如果其值超过 aMaxTransmitMSDULifetime,则该 MSDU 的未被发送的分段将被丢弃。

3.7 重组(Defragmentation)

为了使目的 STA 能对 MSDU 或者 MMPDU 完成重组,每个分段在其头部应包含下述的信息:帧类型、从 Address2 域获取的发送地址、目的 STA 地址、顺序控制域、更多分段指示位。根据顺序控制域,目的 STA 会检测收到的分段是否属于同一个 MSDU 或 MMPDU,以及这些分段之间的先后顺序。更多分段指示位指示目的 STA 当前收到的分段是否是 MSDU 或 MMPDU 的最后一个分段。当该分段是最后一个分段,或者该 MSDU 或 MMPDU 只有一个分段时,该域值被置为 0,其他分段该位都置为 1。

目的 STA 按照顺序控制域中段顺序号的顺序来重组 MSDU 或 MMPDU。只有在 STA 收到的更多分段指示位的值为 0 的分段时,才认为该 MSDU 或 MMPDU 已经接收结束。

STA 为每个接收到的 MSDU 或 MMPDU 维护一个接收定时器。由于所有的 STA 都应支持并发接收至少 3 个 MSDU 或 MMPDU,所以每个 STA 必须支持至少 3 个接收定时器,或者更多。目的 STA 会丢弃没有接收定时器的 MSDU 或 MMPDU 的所有分段。aMaxReceiveLifetime 规定允许接收一个 MSDU 的最长持续时间。STA 从收到 MSDU 或 MMPDU 的第一个分段就开始为其计时,如果接收定时器超过 aMaxReceiveLifetime 规定的时间,STA 将丢弃已接收到的该 MSDU 或 MMPDU 的所有分段。此后,如果 STA 仍然收到该 MSDU 或 MMPDU 的分段,则 STA 将丢弃该分段,并同时对该分段进行应答。

为了正确重组一个 MSDU 或 MMPDU,目的 STA 将丢弃重复接收到的分段,但是仍然要对该分段进行应答。

第四章 基于 DSP 的 MAC 层系统设计

通过前面对 IEEE 802.11MAC 层协议的研究,对基于 IEEE 802.11 技术的无线局域网有了一定的认识,我们下一步的工作就是根据参与项目的要求和整体安排,结合市场上各种应用芯片的性价比,确定 802.11 MAC 层协议实现方案。

项目最后提出并确定了 ARM、DSP 和 FPGA 三种处理器相结合的终端体系结构,通过 ARM 对目标及环境进行建模、运算,应用 DSP 进行数据调度、运算和处理,最后形成所需的调幅、调相、调频等控制字,通过 FPGA 控制收发器芯片产生射频模拟信号。利用数字芯片之间的通用性,ARM 与 DSP 间的通信,不仅能实时处理接收和发送的数据,还可以适应不同移动网络的具体要求,同时方便加载新的程序。FPGA 数字频率合成技术以其在频率捷变速度、相位连续性、相对带宽、高分辨率以及集成化等方面的优异性能,为终端射频信号模拟的实现方式提供了选择。

4.1 终端的系统结构

终端的系统结构如图 4-1 所示,包括 ARM 主控模块、DSP 实时数据处理模块和 FPGA 信号生成模块三个主要部分,还含有数据采集控制、主从系统及其外围器件设计、ARM 和 DSP 接口、ARM 和 FPGA 的接口、FPGA 和 DSP 的接口等部分。

ARM 主控模块负责系统监控和管理,协调系统的工作。ARM 装配有 Linux 操作系统,通过裁剪和修改的 Linux 内核实现了底层系统的应用程序接口以及 HPI(Host Port Interface)底层驱动程序(ARM 部分)。通过 DSP 自带的 HPI 并行接口,ARM 可以直接访问 DSP 的存储空间,包括映射的外围设备。嵌入式操作系统 Linux 的内核裁剪和修改的任务由项目组其他同学完成。

DSP 实时数据处理模块通过提供完整的 HPI 驱动程序(DSP 部分)及 IEEE 802.11 MAC 层控制协议(包括 DCF、PCF、分段和重组等),增加相应数据处理控制算法程序,利用 HPI 并行接口与主机 ARM 进行通信,实现各种实时处理和控制在。一般情况下,数据的复杂运算由 DSP 来实现,但在算法非常稳定后,可以用 FPGA 代替实现,以减少 DSP 的负担。因此,FPGA 也可以视为 DSP 的缓存器。DSP 与 FPGA 之间通过数据总线、地址总线进行通信。

FPGA 信号生成模块主要负责接收来自 DSP 的数据和控制参数后,通过 MII(Media

Independent Interface)接口控制收发器完成数据的发送, 或者是缓冲收发器接收到的数据, 以中断方式通知 DSP 进行数据的接收。根据 DSP 提供的控制参数, FPGA 进行信号提取, 并通过初步的数据运算, 如信道编解码、CRC 附着、交织、调制解调等, 来降低数据速率。FPGA 和 ARM 的连接, 可以视为一个备选方案, ARM 与 FPGA 之间的通信由 DSP 控制。当 FPGA 的数据处理速度足够满足本项目的需求时, 可以不通过 DSP 的进一步处理, 因此可以直接由 ARM 提供 FPGA 的控制参数。

图 4-1 中所示的 ARM 芯片[18]选择了三星电子公司生产的 S3C2410。S3C2410 是基于 ARM920T 处理器核心、采用 0.18mm 制造工艺的微处理器, 具有 16KB 指令和 16KB 数据 Cache、内存管理单元(MMU)、支持 TFT 的 LCD 控制器、NAND 闪存控制器、3 路 UART、4 路 DMA、4 路带 PWM 的 Timer 、I/O 口、RTC、8 路 10 位 ADC、Touch Screen 接口、IIC2BUS 接口、IIS2 BUS 接口、2 个 USB 主机、1 个 USB 设备、SD 和 MMC 接口和 2 路 SPI, S3C2410 处理器最高可运行在 268MHz。

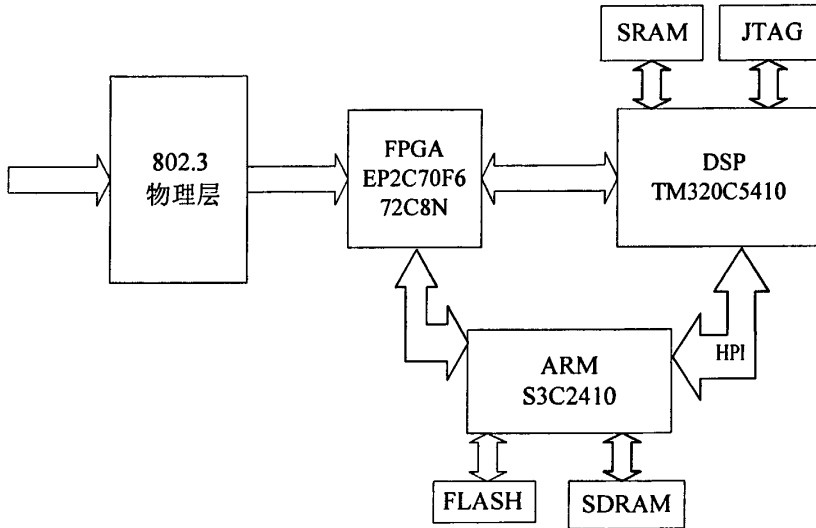


图 4-1 终端 的系统结构

本系统 DSP 芯片选用了 TI 公司的定点型 DSP TMS320C5410[19], 其特点如下: 操作速率可达 100Mips(百万条指令每秒); 具有先进的多总线结构(1 条程序总线、3 条数据总线和 4 条地址总线); 40bit 算术逻辑单元(ALU); 可进行非流水线式单周期 MAC 运算; 内置 64K RAM、16K ROM; 带有软件可编程等待状态发生器、锁相环 PLL 时钟发生器; 一个 8 位并行主机接口、3 个多通道缓冲串行口等。该芯片采用 3.3V 和 2.5V 双电源供电, 其中 IO 采用 3.3V 电源供电, 内核采用 2.5V 电源供电。

在 FPGA 的选择上, Cyclone II 器件[20]系列是成本最低的 FPGA 解决方案, 提供空前的设计灵活性和高性能系统集成能力。Cyclone II 器件架构提供多达 68416 个逻辑单元

(LE); 采用了 TSMC 领先的 90nm 低电介工艺技术; 基于流行的 M4K 存储器块, 提供多达 1.1 兆比特的嵌入式存储器, 可以支持配置为广泛的操作模式, 包括 RAM、ROM、先入先出(FIFO)缓冲器以及单端口和双端口模; 提供最多 150 个 18×18 比特乘法器, 可用于实现通用 DSP 功能; 提供高级外部存储器接口支持, 数据速率最高可达 668 Mbps; 支持串行总线和网络接口(如 PCI 和 PCI-X), 快速访问外部存储器件, 同时还支持大量通讯协议, 包括以太网协议和通用接口; 支持最多达四个可编程锁相环(PLL)和最多 16 个全局时钟线; Nios II 嵌入式处理器给低成本分立式微处理器提供了一个理想的替代方案; 可以采用 Altera 的低成本串行配置器件进行配置, 来提供最大 64 兆比特的 Flash 存储器。

此外, 图 4-1 中 802.3 物理层即以太网收发器的出现是由于仿真调试所需, 它采用 Intel 公司的 LXT972A[21]快速以太网物理层协议芯片, 支持速率 10Mbps 和 100Mbps 的全双工通信, 并提供三种工作模式: 自动协商、并行检测、手动控制。LXT972A 通过 MII 接口与 FPGA 进行通信。

此方案的制定具有相当大的灵活性, 对于软件或者硬件的开发都可按照需要进行设计和完善, 而且在必要时可自行升级以符合环境需要。

4.2 终端的工作流程

基于图 4-1 所示的硬件平台, 装配有 ARM 主控模块、DSP 实时数据处理模块和 FPGA 信号生成模块的终端, 在这三个主要模块的协同工作下, 实现了 IEEE 802.11 MAC 层协议功能。模块间协同工作过程如图 4-2 所示。

通过 HPI 并行接口, ARM 系统将来自上层的用户信息和控制参数传给 DSP, 也可以接收由 DSP 上传的 LLC 数据。DSP 从 HPI 接口接收到来自 ARM 的数据后, 便将数据传递给 MAC 层控制协议程序, 以便其对数据进行如下处理, 如添加头部、根据消息长度决定是否分段等。接着通过数据总线、地址总线, 将处理完的数据和一些控制参数交付给 FPGA。一旦 FPGA 获知 DSP 有数据需要发送, 便根据相应的控制参数对数据进行编码等工作, 以使其适合在信道上传输。在完成编码等一系列运算后, 便通过 MII 接口控制以太网收发器将数据发送到信道上。

在物理信道上捕获到有效接收信号后, 以太网收发器便通知 FPGA 开始数据的接收。在一帧数据接收的开始和结束, FPGA 都会通过中断引脚发送中断信号给 DSP。根据相应的寄存器, DSP 在获知一帧数据接收结束后, 就通过数据和地址总线读取 FPGA 的接收缓冲 RAM 中缓存的数据。接着 MAC 层控制协议程序对接收到的数据进行相应的处理, 如

去除头部、分段的重组、重复帧的过滤、应答当前帧等，并从接收的数据中获取一些有用的信息。数据处理结束后，DSP 通过 HPI 并行接口将处理完的数据，也就是 LLC 数据交付到 ARM。

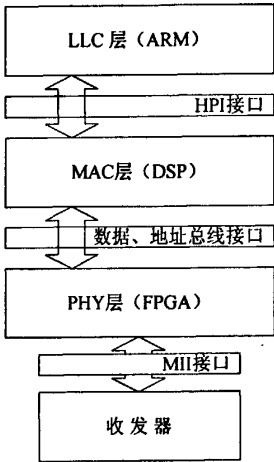


图 4-2 终端 的工作流程

4.3 MAC 层的调度模块

由于在 DSP 中实现 IEEE 802.11MAC 层协议缺乏操作系统的支持，本论文设计了一个 MAC 层的核心调度模块。根据系统不断产生的消息，调度模块进行相应的操作，来协调整个系统的正常运作。MAC 层调度模块是整个系统运作的一个关键模块。该模块主要通过一个 while 循环不断查询消息队列来处理消息，如果没有则一直处于查询状态，直到有新消息需要处理。消息结构定义如下：

```
struct MESSAGE
{
    enum    EMESSAGEID    MessageID;
    short   Priority;
    void*   pParam;
    struct  MESSAGE*      pNextMsg;
    enum    SYSTEMMODULE  ToModule;
    enum    SYSTEMMODULE  FromModule;
};
```

其中 EMESSAGEID 枚举类型成员 MessageID 定义了消息 ID，包括管理消息、接收相关消息、发送相关消息等；Priority 表示消息处理的优先级控制信息，数据范围为 0-65535，Priority

的数值越低，相应消息的处理优先级越高；空类型指针 `pParam` 用来指向内存中的某段数据，也就是该消息携带的数据净荷；结构体 `MESSAGE` 类型指针 `pNextMsg` 用来指向消息队列中下一个消息，一旦处理完成，调度模块就会从消息队列中删除该消息；`SYSTEMMODULE` 枚举类型成员 `ToModule`、`FromModule` 分别表示该消息来自的源模块和将要处理它的目标模块。`MAC` 层调度模块根据消息 `ID`、`ToModule`、`FromModule` 来确定处理消息需要调用的程序模块。

消息队列采用链表结构实现，`struct MESSAGE` 类型指针 `header` 指向消息队列的头部，`struct MESSAGE` 类型指针 `tail` 指向消息队列的尾部。消息队列的操作主要由 4 个函数来实现。初始化函数 `CreateMsgHeaderTail()` 中用来初始化消息队列。`MessageListInsert()` 函数实现新消息节点的内存分配，消息队列的插入和更新(如消息的优先级的修改)。在消息处理函数 `MessageNodeFree()` 中，首先在消息队列中查找一个优先级最高的信号并处理，将产生的新消息插入消息队列，然后从消息队列中删除已处理过的信号，更新消息队列。函数 `AllMsgFree()` 用来删除消息队列中的所有消息，并释放消息节点所占的内存。

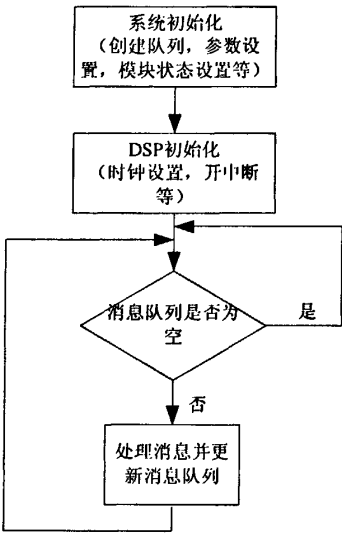


图 4-3 MAC 层调度模块流程图

图 4-3 所示为 `MAC` 层调度模块流程图。首先对整个 `MAC` 层协议系统进行初始化，包括创建相应队列结构(消息队列、传输队列、接收队列等的初始化)、系统参数设置、终端工作模式的选择(站点或者 `AP`)、模块初始状态设置等。接着初始化 `DSP`，设置时钟，开中断。然后进入 `while` 循环，不断查询消息队列是否为空。如果消息队列非空，读取消息队列中优先级最高的消息并作处理。对于在消息处理过程中产生的新的消息，在设置这些消息的优先级后将其插入到消息队列，并更新优先级队列。如果消息队列为空，则继续查询该消息队列，直到有消息可处理。

4.4 MAC 层协议与上下层的接口

4.4.1 接口要求

MAC 层与物理层，即 DSP 与 FPGA 之间以总线的方式相连。此时，FPGA 被当成是 DSP 的外部存储器。在进行数据访问和控制时，接口部分共定义了四个 16 位的控制寄存器。FPGA 在接收到有效数据后以中断方式通知 DSP。DSP 可以直接访问 FPGA 的内部存储器空间，也可以响应 FPGA 产生的中断请求(外部中断引脚 INT0)。

MAC 层与 LLC 层间的通信，也就是 DSP 与 ARM 之间的通信的核心是 DSP 器件本身带有的 HPI 并行接口。DSP 部分提供完整的 HPI 驱动程序(DSP 部分)，从 ARM 获取数据，然后交给通信协议及相应数据处理程控制算法程序；ARM 部分提供 HPI 底层驱动程序，传递上层命令给 DSP。DSP 利用 HPI 并行接口与主机 ARM 进行数据交换。

4.4.2 MAC 层与物理层的接口

MAC 层与物理层之间通信接口(MAC_PHY 接口)负责协调和规范 MAC 层和物理层的信息的交互的过程和方式。MAC_PHY 接口主要分成两部分：DSP 部分负责向 FPGA 的发送缓冲写入数据和从 FPGA 的接收缓冲读取数据；FPGA 部分在发送时负责将发送缓冲中的数据发送到空中接口，在接收时负责缓存接收数据并通知 DSP。DSP 与 FPGA 的硬件总线连接设计如图 4-4 所示。

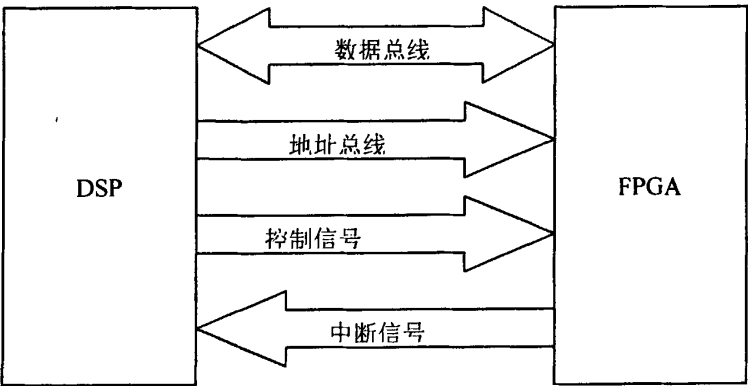


图 4-4 DSP 和 FPGA 的总线接口

为了实现 DSP 与 FPGA 之间的信息的交互，我们定义了 4 个 16 位的控制寄存器和 3 个宽度为 4bit，深度为 4096 的缓冲 RAM。控制寄存器主要用来存储控制信息，包括总体控制寄存器 ACNTL、发送相关寄存器 TCNTL1、发送相关寄存器 TCNTL2、接收相关寄

寄存器 RCNTL。ACNTL 寄存器主要存储系统的状态信息和部分控制参数,如信道忙/闲状态、中断响应状态等; TCNTL1 和 TCNTL2 寄存器用来存储 DSP 发送数据时给 FPGA 的一些指示信息,如发送数据的长度、调制方式、发送速率等; RCNTL 存储 DSP 从 FPGA 接收数据时需要的一些参数信息,如接收缓冲 RAM 指示信息、接收数据长度等。

三个缓冲 RAM 分别为接收缓冲 RAM0 和 RAM1、发送缓冲 RAM2,接收缓冲 RAM 用来缓冲 FPGA 从信道中接收的有效数据,发送缓冲 RAM 被 FPGA 用来暂存 DSP 发送的数据。这三个缓冲 RAM 均映射到 DSP 的地址空间, DSP 可以通过映射地址直接读写它们。在接收时, DSP 根据 RCNTL 的第 14 比特位来获知接收缓冲 RAM (0 或 1)。

当终端处于发送状态时,在完成退避并且媒体空闲时间满足要求后, DSP 在写入数据到发送缓冲 RAM2 后,将控制信息写入寄存器 ACNTL、TCNTL1 及 TCNTL2,通知 FPGA 有数据需要发送。FPGA 在查询到有数据要发送后,调用编码模块对数据进行编码,最后通过收发器将数据发送至信道。

当终端处于接收状态时, FPGA 将从信道中接收的数据存储于接收缓冲 RAM0(1),并以中断方式通知 DSP。DSP 在中断信号的中断处理程序中,根据 RCNTL 寄存器的指示到相应 RAM 映射地址读取数据。接着处理数据,处理过程包括辨别当前帧是否是广播帧或者目的地址是否是本站点、重复帧的过滤、去除 MSDU 头部、分段的重组、收到帧的应答等。如果收到的是不需重组或者是重组已完成且正确接收的数据帧或者管理帧,则通过 HPI 并行接口将数据发送至 ARM(LLC 层)。

4.4.3 MAC 层与 LLC 层的接口

MAC 层与 LLC(Logical Link Control)层的接口(MAC_LL C 接口)用来协调和规范 MAC 层和 LLC 层信息交互的过程和方式,该接口也分成两部分: DSP 部分完成从 ARM 获取数据,并交给通信协议及相应数据处理程控制算法程序的工作; ARM 部分负责传递上层命令给 DSP 和从 DSP 接收 LLC 数据。

DSP 与 ARM 之间通信主要通过 HPI 并行接口,通过该接口 ARM 微控制器可以直接访问 DSP 的内部存储器空间或地址映射到存储器空间的外围设备,还可以响应 DSP 发过来的中断请求,对于 ARM 来说, DSP 系统芯片就相当于一个外部设备。具体的硬件连接设计如图 4-5 所示。

TMS320VC5410 芯片中 HPI 接口模块由 8 位数据线 HD[7:0]和 10 条控制线构成。10 条控制线中,两条地址线 HCNTRL[1:0]负责对 HPI 口的内部三个寄存器(控制寄存器 HPIC、

地址寄存器 HPIA 和数据锁存器 HPID)进行寻址，从而访问 DSP 内存空间；HBIL 为字节区分信号线；HR/W 为读写控制；HCS 为 DSP 的片选；HRDY 为 DSP 就绪状态；HDS1、HDS2 用于和 HCS 生成内部读写控制逻辑；HINT 为中断请求；HAS 为主机地址使能，通常不使用，接高电平即可。

当高层有数据需要传输时，ARM 通过 HPI 接口将数据写入 HPI RAM，并置位 HPIC 中的 EINT0 位来中断 DSP。中断响应后，DSP 对接收的数据进行处理后交付到 FPGA 进行编码。当以太网收发器有数据到达时，FPGA 通知 DSP 对其进行处理。DSP 将处理过的数据写入 HPI RAM 中，然后通过将 HPIC 的 HINT 引脚置位低电平向主机 ARM 产生中断请求，告诉 ARM 有数据到达。

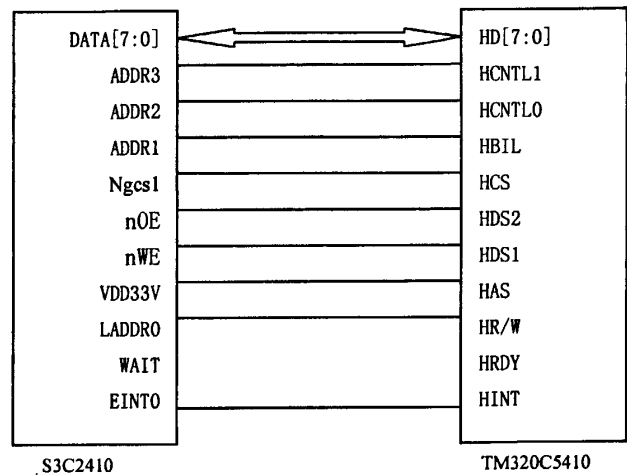


图 4-5 ARM 和 DSP 的总线接口

4.4.4 接口 调试及分析

在 MAC 层与上下层之间的接口部分的软件设计和实现后，我们确定了调试方案，并编写了基于图 4-1 的终端硬件平台的调试程序，主要用来调试接口的工作情况。经过调试，对于 MAC 层与 LLC 层的接口部分，ARM 和 DSP 都能正确读写，并且都能产生和响应中断。对于 MAC 层与物理层的接口部分，DSP 和 FPGA 之间通信正常，FPGA 能准确地产生中断，DSP 也能及时地去响应中断。在调试接口部分的数据收发时，令发送终端在模拟的无线信道上连续发送 100 帧数据，接收端对正确接收到的有效数据帧进行计数。结果发现，接收端正确接收到的有效数据帧为 100 帧，不存在数据丢失现象。

第五章 MAC 层部分功能模块的软件实现

在无线局域网的体系结构中，工作站STA和接入点AP是两个基本的组成单元。由于STA和AP在无线局域网网络结构中所起的作用不同，实现的功能也不尽相同，因此构成STA的IEEE 802.11 MAC层协议栈的实现和构成AP的IEEE 802.11 MAC层协议栈的实现有比较大的区别。

本章首先简要介绍STA的MAC层协议软件框图，并对其七个模块进行了简单介绍，然后分别编码实现了MAC_Data_Service模块、MPDU_Generation_STA模块、Protocol_Control_STA模块、Transmission模块以及Reception模块。其中Protocol_Control_STA、Transmission以及Reception三个模块协作完成IEEE 802.11的媒体接入控制核心协议——分布协调功能(DCF)和点协调功能(PCF)。

5.1 MAC 层软件框图

MAC层协议的软件实现有多种方法。[22][23]中给出了使用SDL来构造MAC层软件的方法。本文中直接使用不同的状态机和C\C++代码来构造软件[24][25]实现。

为了规范编程，我们采用了模块化的软件设计方法。图5-1是构成STA的IEEE 802.11 MAC层协议栈的基本模块划分，从实现的功能上分可以分为七个模块。本论文仅对STA的IEEE 802.11 MAC层协议栈中的五个功能模块进行了设计和实现。

1. **MAC_Data_Service**: 该模块从LLC层接收数据包，添加MAC头部，产生MSDU，然后送至MPDU_Generation_STA模块；从MPDU_Generation_STA模块接收MSDU，除去MAC头部，然后送至LLC层。
2. **MPDU_Generation_STA**: 将需要发送的MAC服务数据单元(MSDU)和MAC管理协议数据单元(MMPDU)转换为MAC协议数据单元(MPDU)，然后送至Protocol_Control_STA模块。如果需要，应对数据帧进行加密。
3. **MAC_Management_Service**: 该模块管理MAC的管理信息库(MIB)；处理MIB的访问；处理管理实体(Mlme)的请求和应答。
4. **MLME_STA**: 包括发起IBSS；产生scan、join、beacon/dwell、awake/doze定时；产生重关联、关联、去关联、认证、解认证服务；管理站点电源状态。
5. **Protocol_Control_STA**: 包括DCF功能；RTS/CTS机制；响应CF-Poll；产生ACK和

CF-ACK: 管理各类帧的重传; 处理Atim和PS-Poll。

6. **Transmission:** 从Protocol_Control_STA接收MPDU, 产生FCS, 然后发送比特流到WM中。
7. **Reception:** 从WM中接收比特流, 检查比特流是否有效, 进行解密, 滤除重复帧, 对分段帧进行重新组合, 侦听媒质状态(物理和虚拟载波侦听)。

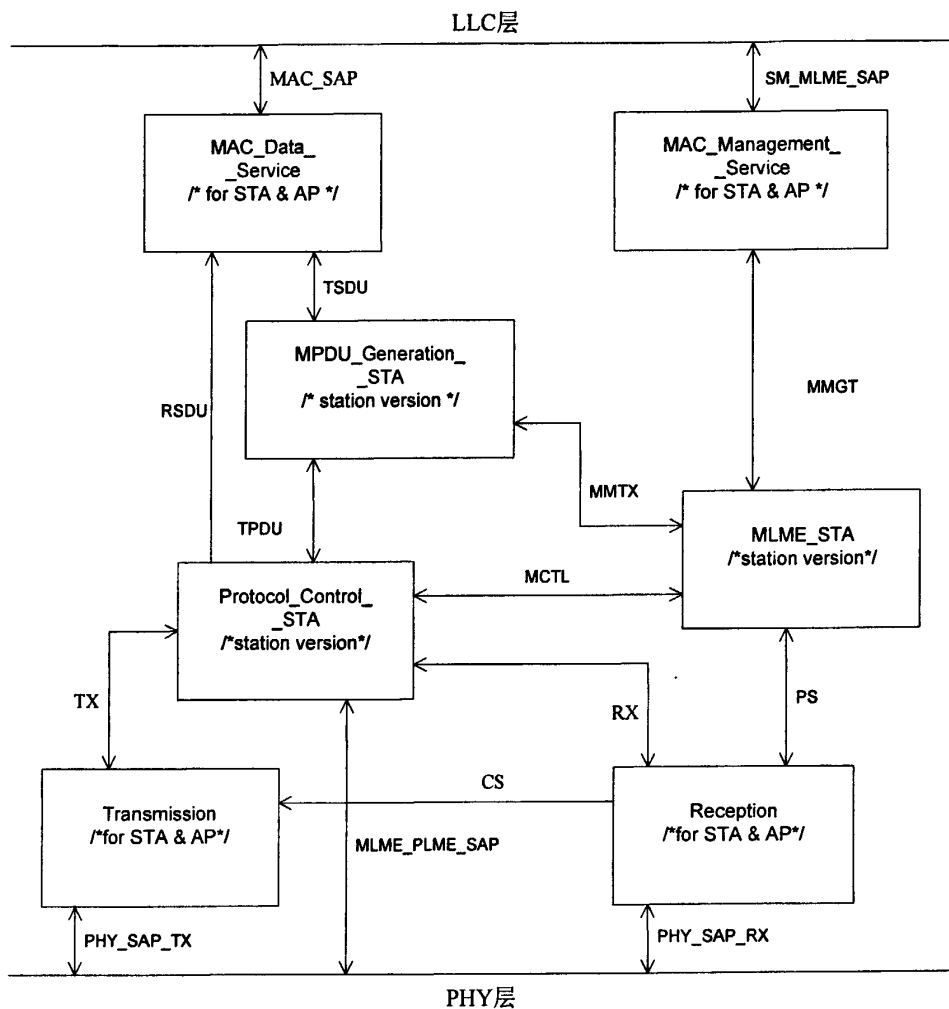


图 5-1 STA 的 MAC 层协议软件框图

STA的802.11MAC的七个模块之间必须进行必要的信息交互才能完成协议栈的功能。信息的交互由4.3中的MAC层调度模块来完成。

Protocol_Control_STA、Transmission、Reception三个模块与物理层相联系，从PHY接收数据、向PHY发送数据；MAC_Data_Service模块则与LLC层之间进行数据的交互；MPDU_Generation_STA模块负责将接收到的MSDU或MMPDU转换成MPDU，串联了Protocol_Control_STA和MAC_Data_Service模块。下面具体介绍这五个模块的软件实现，

编程语言采用标准C语言，主要为了便于今后软件代码的移植。需要说明的是，这五个模块的完全功能实现还需要PHY硬件驱动程序和良好操作系统的支持。

5.2 MAC_Data_Service 模块

该模块由两个子模块：MSDU_to_LLC、MSDU_from_LLC模块组成，提供MAC_SAP功能。MSDU_to_LLC模块负责从MPDU_Generation_STA模块接收MSDU，获取地址和状态信息，除去MAC头部，然后将数据送至LLC层。

MSDU_from_LLC模块负责从LLC层接收数据包，如果LLC层请求参数有效，则添加MAC头部，产生MSDU，并送至MPDU_Generation_STA的Prepare_MPDU子模块；如果参数无效或者传输尝试失败，则通知LLC层。该模块填充MAC头部的部分参数域，其余参数域在发送前设置。

5.3 MAC_Generation_STA 模块

本模块是MAC层接入控制的适配模块，主要将需要发送的MSDU或MMPDU转换为MPDU，然后送至Protocol_Control_STA模块。如果需要，应对数据帧进行分段和加密。该模块由两个子模块：Prepare_MPDU、PM_Filter_STA模块组成。

5.3.1 Prepare_MPDU 模块

如果需要发送的MSDU或MMPDU的长度大于dot11FragmentationThreshold参数，Prepare_MPDU模块将该MSDU或MMPDU转换为一个或多个分段MSDU。如果dot11PrivacyOptionImplemented参数设置成true，Prepare_MPDU模块应对数据帧进行加密。本论文中该参数被设置为false。

如果站点既没有被关联，也没有通过认证，则该模块一接收到MSDU或者MMPDU就向MAC_Data_Service模块报告。一旦经过分段和加密，就将MPDU报告给PM_Filter_STA模块。

该模块的软件实现流程如图5-2所示，交互过程如下：

```
if(mAssoc==true && mActingAsAp==false)
{
    memcpy(h->dh_3a,h->dh_ra,ETHER_ADDR_LEN*2);
```

```
memcpy(h->dh_ra,mbssid,ETHER_ADDR_LEN*2);

h->dh_fc.fc_to_ds=1;           //如果已关联则设置MAC头相关域
}

else if(mActingAsAp==false && mibss==false)
{
    //站点未通过认证
    **设置消息参数nobss
    **产生消息向MAC_Data_Service模块报告
}

** 如果长度大于dot11FragmentationThreshold则进行分段
**产生消息向PM_Filter_STA模块报告
```

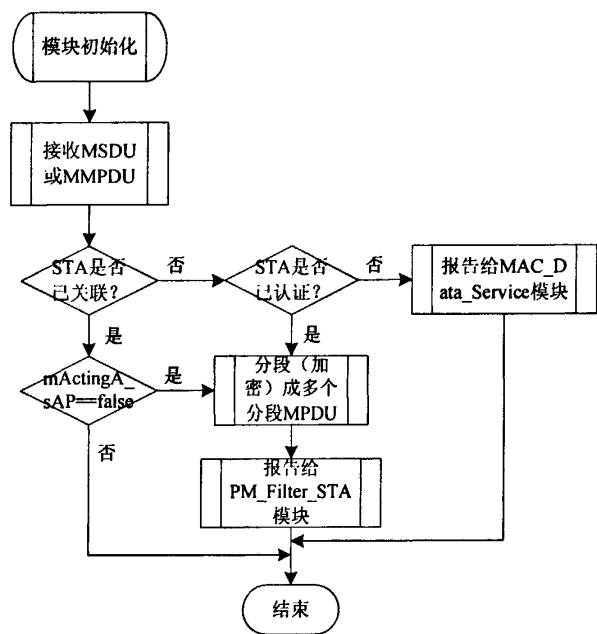


图5-2 Prepare_MPDU模块流程图

5.3.2 PM_Filter_STA 模块

PM_Filter_STA模块主要负责在无竞争期间，为BSS中的无竞争可轮询站点欲发送的帧以及IBSS中的需要通告的帧提供队列服务。该模块从Prepare_MPDU模块接收分段MSDU或向其报告分段MSDU，并向Protocol_Control_STA模块报告MPDU或从其接收MPDU。

5.4 Protocol_Control_STA 模块

本模块是MAC层接入控制的核心模块，主要完成分布式协调功能(DCF)，对CF-Poll进行响应，接收到MPDU后进行响应，实施RTS/CTS握手机制，进行各类帧的重传，处理Atim和PS-Poll等。该模块由两个子模块：Tx_Coordination_STA、Rx_Coordination_STA组成。

5.4.1 Tx_Coordination_STA 模块

Tx_Coordination_STA子模块的主要功能是实施调度控制策略，利用WM把MPDU在合适的时刻发送出去，与之协同工作的是Backoff_Procedure退避过程子模块。该模块的软件实现流程如图5-3所示。

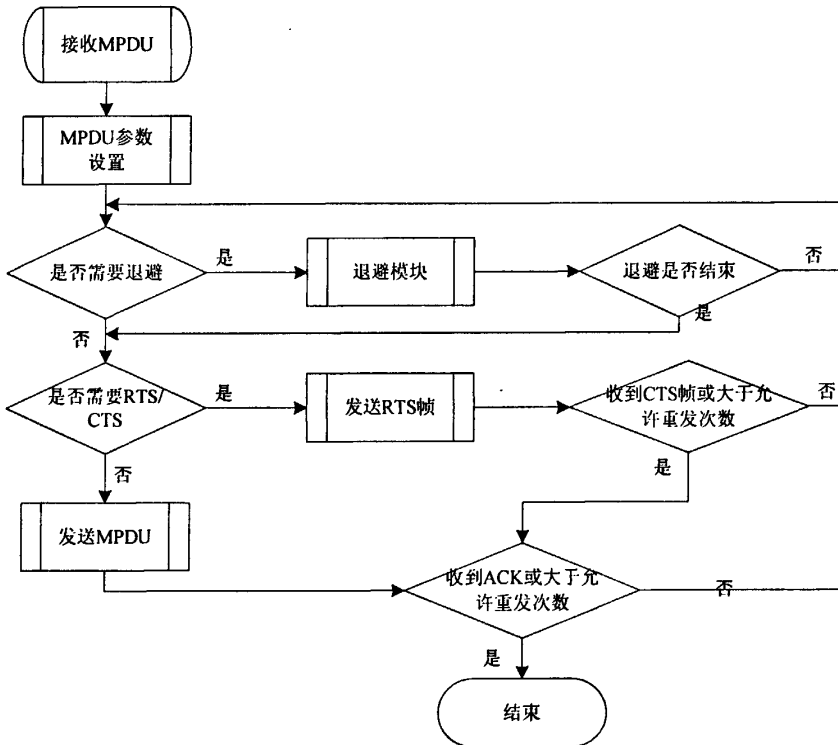


图5-3 Tx_Coordination_STA模块流程图

软件实现内容包括：MPDU在发送前期进行参数的初始化、实施发送交互过程、发送结束时对参数进行恢复。参数的初始化和参数恢复主要是对一些变量和定时器进行设置。发送交互过程有两种模式：Data/ACK和RTS/CTS/Data/ACK模式，其中Data/ACK模式是一般的数据帧交互模式，RTS/CTS/Data/ACK交互模式是为了保证传输质量、提高传输的可靠性而采用的交互模式。

软件模块主要包括RTS/CTS帧交互模块、Data/ACK帧交互模块、Backoff_Procedure

退避过程模块等。RTS/CTS帧交互是802.11 MAC层针对较长MPDU而设计的一种提高网络性能的措施。管理信息库MIB中的dot11RtsThreshold决定了MPDU在发送之前是否需使用RTS/CTS帧交互,如果该值为0,则表示所有MPDU在发送前都需先使用RTS/CTS。如果待发送MPDU长度大于dot11RtsThreshold(非0),则源站点首先需向目标站点发送RTS帧,目标站点在收到RTS帧后,等待一个SIFS发送一个CTS帧进行应答,对其它STA宣告媒质已被占用。其它站点此时可以利用CTS帧里的Duration/ID域的值来调整本地NAV值。

RTS/CTS协议的工作机制如下:

1. 源站点在发送信息前先发送一个发送请求控制帧,即RTS帧给目标站点。
2. 如果信道空闲,目标站点发送响应控制帧,即CTS帧;如果检测到信道忙,不发送CTS。这样就可以避免了不同工作站点同时向同一目标站点发送信息的可能。
3. 如果源站点收到CTS帧,则经过一个SIFS间隔后就可以继续发送数据帧。
4. 如果该数据帧需要确认,目标站点在成功接收后,经过一个SIFS间隔后就发送ACK确认帧。如果在规定的時間间隔后,源站点未能收到ACK,那么它就可判断信息发送失败,然后根据信道情况进行重传。

RTS/CTS交互过程如下:

```
if((length(mpdu)+sCrcLng) > dot11RtsThreshold && !isGroup(p_macheader)
tpdu.retry!=true && tpdu.fragCur!=0)
{
    //如果不是超长帧并且不是组播就产生RTS帧
    rtsdur.duration=3*aSifsTime+ceiling((stuff(aMpduDurationFactor(
aPlcpHdLength+aPreambleLength,sMaxMpduLength),8*( length(mpdu)+
aCrcLength))+2*stuff(aMpduDurationFactor( aPlcpHdLength+
aPreambleLength,sMaxMpduLength),sAckCtsLength)+aPlcpHdLength+
aPreambleLength)/txRate));//根据传输速率和传输比特长度计算传输时间值
BackoffDone(); //等待退避完成
TxQequest( rtsdur,txRate);//发送RTS帧
** 设置定时器Trsp=now()+dUsec(aSifsTime +calcDur(txRate,
**stuff(aMpduDurationFactor,sAckCtsLng))+aPlcpHeaderLength+aPreambleL
**ength+aSlotTime)
**等待CTS帧
if(ccw!=aCWmax)ccw=2*ccw+1; //退避窗口取下一个序列值
** Backoff( ccw,-1) //进行退避
```

```
slrc++;  
mpdu.lrc++;    //MPDU重传长计数器增1  
if(slrc==aLongRetryLimit)ccw=aCWmin; //如果到达特定上限，窗口值复位  
}
```

RTS/CTS帧交互后即可进行Data/ACK帧交互过程，源站点发送MPDU给目标站点，目标站点在接收到MPDU并等待一个帧间隔SIFS后，向源站点发送ACK帧进行确认。

如果源站点得知发送的MPDU帧丢失或存在错误，则进入退避过程，准备重新发送此MPDU；如果目标站点已正确接收，则本次发送过程结束。

Data/ACK交互的过程如下：

```
** TxRequest(tpdu,txRate); //发送Data帧  
**设置定时器  
**等待ACK帧  
**如果在ACKTimeout内没有收到ACK则执行以下操作  
{  
if(ccw!=aCWmax) ccw=2*ccw+1; //退避窗口取下一个序列值  
** Backoff(ccw,-1);  
p_macheader->fc.fc_retry=1;  
mpdu.resume= 1;  
if(length(mpdu)+sCrcLng>dot11RtsThreshold) {    //判断MPDU长度  
slrc++;    //站点重传计数器加1  
mpdu.lrc++;  
if(slrc == aLongRetryLimit)ccw=aCWmin;  
if(mpdu.lrc==aLongRetryLimit)**丢弃此MPDU;  
}  
else{  
ssrc++;    //站点短重传计数器加1  
mpdu.src++;    // MPDU的短重传计数器加1  
if(ssrc==aShortRetryLimit) ccw=aCWmin;  
if(mpdu.src==aShortRetryLimit)**丢弃此MPDU;  
}}
```

5.4.2 Rx_Coordination_STA 模块

Rx_Coordination_STA 功能模块根据接收到 MPDU 的类型进行不同的处理。

- (1) 如果接收到的帧是 ACK 应答帧, 则向 Tx_Coordination_STA 模块报告。Tx_Coordination_STA 模块根据 ACK 帧判断前面的数据帧是否传输成功, 然后作相应处理;
- (2) 如果接收到的帧是 CFEnd_ACK 帧, 则向 Tx_Coordination_STA 模块报告。Tx_Coordination_STA 模块根据 CFEnd_ACK 帧判断 CF 是否结束及前一个数据帧是否传输成功, 然后作相应处理;
- (3) 如果接收到的帧是 CFEnd 帧, 则向 Tx_Coordination_STA 模块报告。Tx_Coordination_STA 模块根据 CFEnd 帧判断 CF 结束;
- (4) 如果是 Beacon 帧、Probe Request 帧, 则向 MAC_Data_Service 模块报告;
- (5) 如果接收到的是 RTS 帧, 则根据 NAV 值决定是否进行应答;
- (6) 如果是 CTS 帧, 向 MAC_Data_Service 模块报告前一个发送成功;
- (7) 如果是认证、解认证、ATIM、Probe Response 帧, 则首先判断是否为组播地址, 如果是则不作处理, 如果不是组播地址, 则向 MAC_Data_Service 模块报告 MsduIndicate 消息;
- (8) 如果是数据帧, 则判断发送数据帧的源 STA 是否在 IBSS 中, 如果在, 则向 MAC_Data_Service 模块报告; 如果不在, 则报告给管理实体。

软件实现流程如图 5-4 所示, 主要是根据接收到帧的类型作相应处理。下面的 switch 语句简单地描述了该模块的软件实现结构。

```
switch(Subtype)
{
case MAC_Subtype_CFEnd_CFAck:
    **产生 ACK( 0,0);
case MAC_Subtype_CFEnd:
    ** 产生 CFEnd;
case MAC_Subtype_ACK:
    ** 产生 Ack(endRx, rxRate)
case MAC_Subtype_PR_Req:
case MAC_Subtype_Beacon:
    **产生 MsduIndicate( pdu,endRx ,strTs,noerr)
```

```
case MAC_Subtype_RTS:
    ** import( mNavEnd )
    if( mNavEnd<=now())
        **CTS 响应
case MAC_Subtype_CTS:
    **产生 Cts(endRx,rxRate);
case MAC_Subtype_Data:
    **产生 MsduIndicate(pdu);
case ...
Default break;
}
```

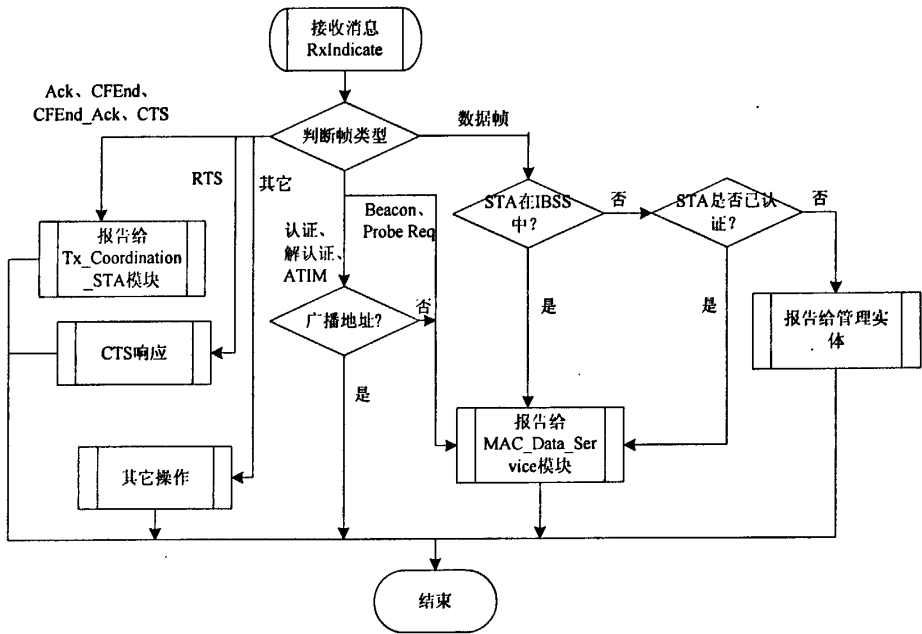


图 5-4 Rx_Coordination_STA 模块流程图

5.5 Transmission 模块

在数据包从 MAC 到 PHY 的传输过程中，Data_Pump 子模块对 MPDU 实施字节级的处理；对数据包进行 CRC 校验；对特定管理帧中的时间戳进行设置；在必要的时候，Backoff_Procedure 子模块实施退避过程。下面将分析其中的 Data_Pump 子模块和 Backoff_Procedure 子模块。

5.5.1 Data_Pump 模块

Data_Pump 模块是发送端与物理层之间的接口,主要功能是对 MPDU 进行帧校验,生成并添加 32bits 的 FCS 校验和序列。如果是 Beacon 帧或者 Probe Response 帧时,将 MAC 头部第 sTsOctet 个字节时戳域设置为 TSF+PHY 传输时延,该域为 1 字节。同步定时功能(Timing Synchronization Function,TSF)用来使 BSS 内的所有站点的时间保持同步,且所有站点都维持一个本地 TSF 定时器。

当 Protocol_Control_STA 模块判断发送所需要的条件都已经具备时,才向 Data_Pump 模块发送 TxRequest。在该模块接收到 TxRequest 消息后,开始把 FCS 序列和 MPDU 发送到物理层。发送最主要的条件就是判断媒质已经空闲。这个过程需要 Backoff_Procedure 模块的参与。软件实现流程如图 5-5 所示。

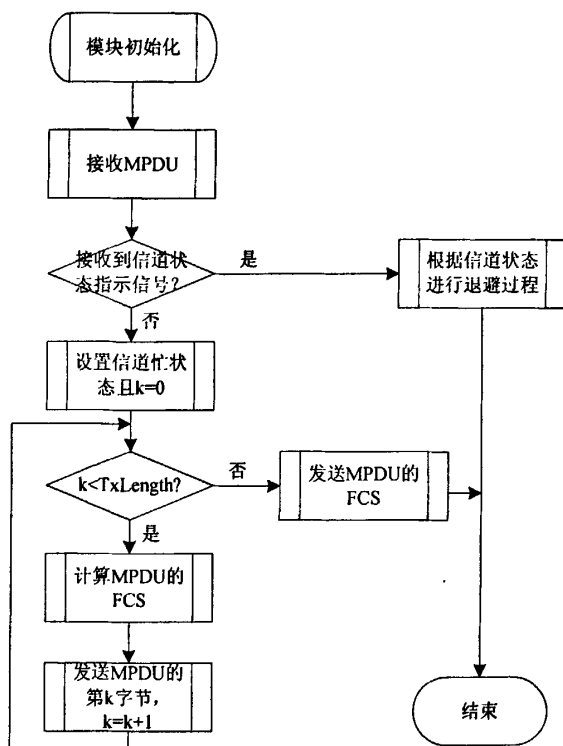


图 5-5 Data_Pump 模块流程

Data_Pump 模块在发送数据的时候,需要和物理层交互信息。Data_Pump 模块将要发送每一个数据包都缓存到 ToPHY 缓冲寄存器中,一个数据包为一字节大小。物理层通过 PhyData_request 和 PhyData_confirm 消息的交互从 MAC 接收数据,逐字节计算 FCS 序列并进行发送。物理层将长度为 TxLength+4 字节的数据写入 FPGA 的发送缓冲区 RAM2,并通过设置发送控制寄存器 TCNTL1 和 TCNTL2 以及 ACNTL 来控制收发器完成数据的发

送。

在计算 FCS 帧校验序列是我们利用 CRC32 来实现。论文中 CRC32 校验采用了查表法，该方法以空间换取时间，在效率上优于一般的直接模仿硬件的算法。查表法主要包含两个子函数：crc32table()和 do_crc32()，其中 crc32table()子函数生成 8bits 的 CRC 校验和列表 crc32Table[]，do_crc32()生成已知长度字符串的 32bits 的 CRC 校验和。子函数 crc32table()定义一个 CRC32 生成多项式 crcpoly，然后采用按位相异或的方式来生成 CRC 校验和列表 crc32Table[]。

```
void crc32table (void)
{
    unsigned long crc32ti,crcpoly;
    int i,j;
    crcpoly=0xEDB88320L; // CRC32 生成多项式
    for (i =0;i< 256;i++ )    //生成 0-255 的 CRC32 校验和表
    {
        crc32ti= i;
        for (j = 8; j>0;j--)
        {
            if(crc32ti&1) crc32ti =(crc32ti >>1)^crcpoly;//循环右移，按位异或
            else crc32ti=>>1;
        }
        crc32Table[i] = crc32ti;
    }
}
```

do_crc32()子函数根据 CRC 校验和列表 crc32Table[]，对长度 len、首地址为 msgAddr 的字符串进行 CRC 计算，生成 32bits 的字符串校验和序列。

```
unsigned long do_crc32 (unsigned char *msgAddr,int len)
{
    unsigned long crc32;
    int k ;
    crc32table();    //调用子函数 crc32table()生成校验和列表
    crc32=0xFFFFFFFF;
```

```
while (len>0)
{
k= *msgAddr;
msgAddr ++ ;
crc32=((crc32>>8)&0x00FFFFFF)^ crc32Table[(crc32^k)&0xFF];
len--;
}
return (crc32^0xFFFFFFFF);
}
```

5.5.2 Backoff_Procedure 模块

Tx_Coordination_STA 子模块在发送 MPDU 前必须进入 Backoff_Procedure 子模块进行退避过程，在 MPDU 发送失败，也必须进行退避过程。退避过程是为了降低媒质发生冲突的概率。软件实现流程如图 5-6 所示。

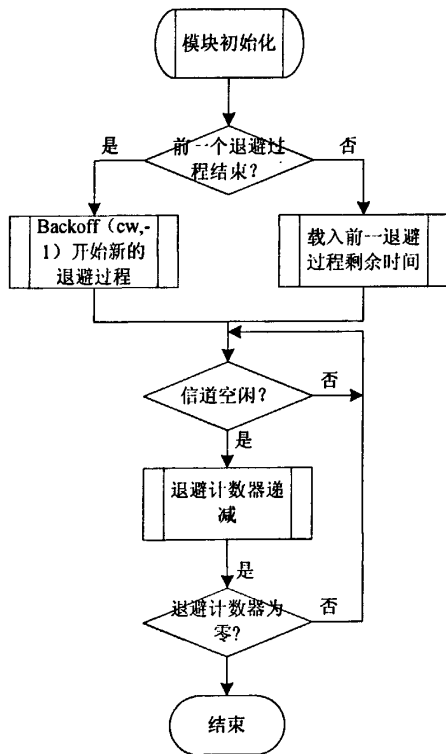


图 5-6 Backoff_Procedure 模块流程图

在退避过程中，退避窗口 CW 在最小值 aCWmin 与最大值 aCWmax 之间取值，与重传次数 N 有关。CW 参数初始值取 aCWmin。每个 STA 都有一个 STA 短重传计数器(SSRC)

和一个 STA 长重传计数器(SLRC)，初始值均为 0。如果有引起 MSDU 重传计数器增加的事件发生，SSRC 或 SLRC 就加 1。如果传输的帧长度大于 dot11RtsThreshold，这次重传就是长重传；如果传输的帧长度小于等于 dot11RtsThreshold，这次重传就是短重传。每次重传都会引起 SSRC 或 SLRC 的增加。如果是因为传输不成功而引起任意一个重传计数器增加 1，CW 就取序列中的下一个值，直到 CW 达到 aCWmax。若 CW 达到了 aCWmax，则保持该值直到被重置。确定了 CW 的值以后，就可以在[0,CW-1]产生一个随机数来初始化退避计数器的值，具体计算关系式为：

$$\text{Backoff_Counter} = \text{Random}()$$

其中，Random()是[0,CW-1]上服从均匀分布的随机整数， $aCW_{\min} \leq CW \leq aCW_{\max}$ ，aCWmin 及 aCWmax 由特定 PHY 特定决定。退避时间 BackoffTime 由 Backoff_Counter 乘以 aSlotTime 可得。对于不同 STA，这个随机值应该以很高概率相互无关，这就保证了在竞争信道时，发生冲突的可能性很小。

5.6 Reception 模块

本模块从物理层接收数据包，先由 Validate_MPDU 子模块对数据包进行有效性检查，再经过 Filter_MPDU 子模块的副本帧过滤，如果实施了 WEP 则还需要进行解密处理，然后由 Defragment 子模块完成分段帧的重组。子模块 Channel_State 用来对信道状态进行实时检测。

在数据帧的接收过程中，首先由通过 MAC_PHY 接口从物理层接收数据帧 MPDU，并计算 32bits 校验和序列 FCS，然后与接收到的 MPDU 帧结构中发送端计算的校验和序列 FCS 进行比较。如果比较结果一致，则把 MPDU 交给 Filter_MPDU 子模块进行副本帧过滤处理；如果不一致，则丢弃此 MPDU。Filter_MPDU 模块完成对 MPDU 的过滤，根据 MPDU 类型的不同而采用不同的处理方式，同时保留并修改相关站点参数。帧重组 Defragment 子模块的功能与 Prepare_MPDU 子模块中的帧分段过程恰好相反，它是把分段帧重新组成长帧并交给 Rx_Coordination 子模块处理。Rx_Coordination 子模块在接收到数据帧 MPDU 后，对发送端进行应答。媒质状态侦听子模块 Channel_State 实时检测媒质状态并将检测结果交给 Data_Pump 子模块，再由 Data_Pump 子模块来通知退避过程子模块 Backoff_Procedure 媒质的忙闲状态。

5.6.1 Validate_MPDU 模块

Validate_MPDU 子模块是接收端与物理层之间的接口，完成从物理层(PHY)接收数据帧 MPDU，根据接收到的 MPDU 重新计算 32bits 的帧校验和序列 FCS，并与接收到的 MPDU 帧结构中 FCS 域的数值进行比较，如果比较结果一致，则接受此 MPDU，如果比较结果不一致，丢弃此 MPDU。

本模块还向 Channel_State 模块通报 DIFS、EIFS、RTS 未响应定时信号。如果该模块在接收 MPDU 过程中发现 MPDU 长度大于 sMaxMpduLng、接收结束时发现实际接收长度和物理层传递的参数不一致、或者接收到的 MPDU 的协议版本与站点不兼容，就会向模块 Channel_State 报告。本模块同物理层直接相连，模块的实现需要和硬件接口相配合。软件实现流程如图 5-7 所示。

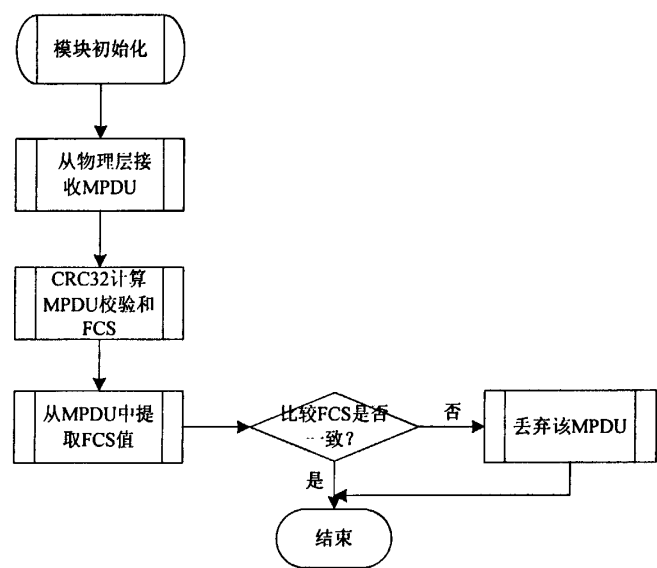


图 5-7 Validate_MPDU 模块流程图

5.6.2 Filter_MPDU 模块

本模块通过接收帧的目的地址或者组播目的地址 BSSID 对 MPDU 进行过滤，同时保持接收 MPDU 的分段号和序列号。

对需要应答的数据和管理帧，本模块向 Protocol_Control_STA 模块发送 NeedAck 消息，同时向 Defragment 模块报告 RxMpdu 消息；如果数据帧稍带了 CfAck 和 CfPoll 信息，则分别向 Protocol_Control_STA 模块发送 RxCfack 信号和 RxCfpoll 信号；如果是接收到 Data+CF-Poll 或者 Data+CF-Poll+CF-Ack 类型数据帧而向 Protocol_Control_STA 模块发送

的 RxCfpoll 信号，则在 SIFS 间隔内需 Tx_Coordination_STA 模块发送 NeedAck 信号。

该模块还负责将预留时间(Duration)和 CFP 剩余时间送给 Channel_State 模块，或向 Mlme 管理实体报告节电模式。

在接收 MPDU 时，该模块首先进行接收地址的过滤。如果接收到的 MPDU 目的地址与本地 MAC 地址相同，则把此 MPDU 存入接收队列，并根据帧类型的不同而进行不同的处理；如果接收到的 MPDU 目的地址与本地 MAC 地址不同，则读取此 MPDU 的 MAC 头部中 dh_duration 域的值来更新本地 NAV 的值，然后丢弃此 MPDU。

如果接收到的 MPDU 目的地址与本地 MAC 地址相同，就执行 MPDU 副本过滤处理过程。读接收到的 MPDU 中 MAC 头部控制域的 fc_retry 位，根据如果此位为 1，则搜索接收队列，看是否存在与此 MPDU 的序列号和分段号一样的 MPDU，如果存在，则删除此 MPDU；如果不存在，则把此 MPDU 放入接收队列，并更新接收队列。软件实现流程如图 5-8 所示。

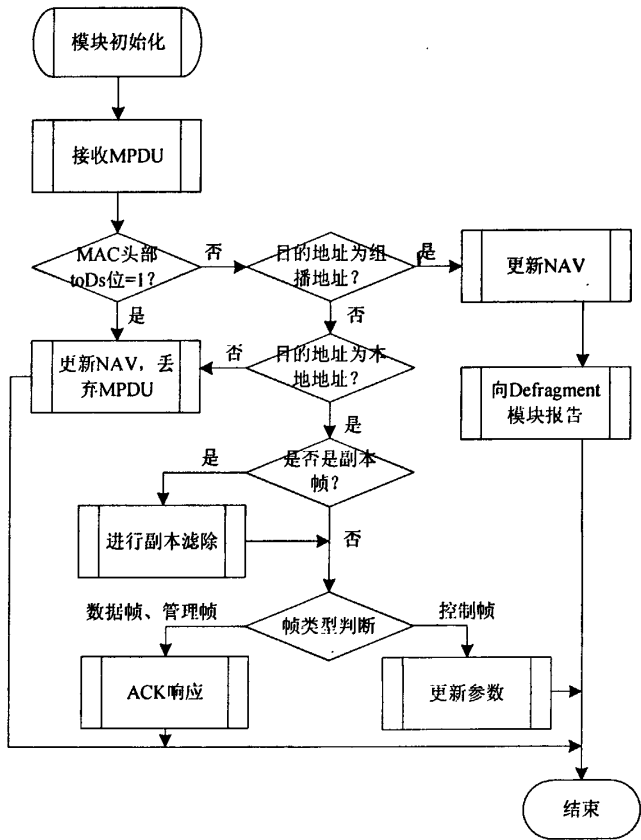


图 5-8 Filter_MPDU 模块流程图

下面是对 MPDU 进行副本过滤的软件实现。软件实现的主要工作是对接收队列的管理和维护，包含三个子函数：clearTupleCache()、updateTupleCache()、searchTupleCache()。

其中, `clearQueue()` 函数释放接收队列。 `searchTupleCache()` 函数在接收队列中搜索与接收到的分段帧具有相同序列号和分段号的分段帧。函数 `updateTupleCache()` 在接收队列中添加 MPDU; 设置接收队列的定时器; 如果定时时间到, 而此 MPDU 还有分段没有接收到, 则删除该 MPDU 已接收的所有的分段, 并释放接收队列。

接收队列中每个 MPDU 的主要参数通过一个结构体 `Tuple` 来管理, 其具体定义如下:

```
struct Tuple{  
    bool    full;                //队列是否为空  
    unsigned char macaddr [ETHER_ADDR_LEN]; //发送地址  
    int     seqnum;              //MPDU 序列号  
    int     fragnum;            //MPDU 分段号  
    long    tRx_low;            //定时器  
    long    tRx_high;  
}
```

5.6.3 Defragment 模块

如果接收到的分段帧实施了 WEP, 而且站点支持 WEP 功能, 也就是 `dot11PrivacyOptionImplemented` 参数值为真, Defragment 模块将先解密该分段帧, 恢复出明文。其中控制帧或者管理帧(认证过程中的加密帧除外)不能实施 WEP 功能, 如果分段帧是认证过程中的加密帧, 或者是控制域 `fc_wep` 位为 1 的数据帧, 就需要对其进行解密。如果站点的 `aExcludeUnencrypted` 参数被设置为真, 一旦接收到未实施 WEP 的分段帧, 就丢弃该帧。

接着, Defragment 模块对解密后的分段帧进行重组, 也就是把接收到的具有相同序列号的所有分段帧重新组装成一个长帧。如果是最后一个分段, 则将生成的 MSDU 报告给 `Rx_Coordination_STA` 模块, 并更新接收缓冲队列。如果是中间分段, 则重组该分段帧并更新接收缓冲队列, 释放所有超时的分段。

Defragment 模块的软件实现流程如图 5-9 所示。本模块软件设计主要包含三个子函数: `psAge()`、`psFree()` 和 `psSearch()`。其中, `psAge()` 子函数设置定时器; `ArFree()` 子函数的功能是释放数据帧 MPDU 的缓存队列; `psSearch()` 子函数的功能是在缓存队列中搜索具有相同序列号且分段号符合要求的分段帧。

接收缓存队列的数据结构是 `PartialSDU`, 定义如下:

```
typedef struct rxpkt{
    struct rxpkt    *rnext;
    long            reol;                //now+aMaxReceiveLifetime
    unsigned short  srcaddr[ETHER_ADDR_LEN]; //addr2, 发送地址
    short           rseqnum;             //MPDU 的序列号
    short           rToRvnum;           //期望接收的下一个分段的分段号
    short           datalen;             //已接收到的数据长度
    unsigned short  llcdata[aMaxLLCDataLen]; //缓存已经接收到的数据
}PartialSDU;
```

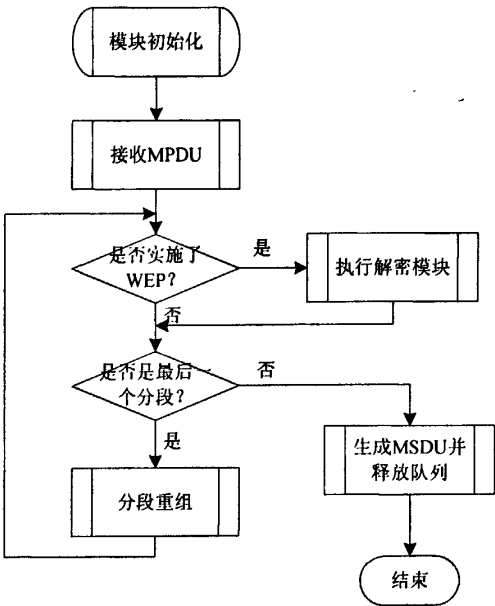


图 5-9 Defragment 模块流程图

5.6.4 Channel_State 模块

IEEE 802.11无线局域网采用CSMA/CA(载波侦听多路访问冲突避免)协议控制共享信道中信息的传送，由PHY层和MAC层共同完成载波侦听和冲突避免。

Channel_State模块同时使用物理载波侦听和虚拟载波侦听两种方法来确定媒质的忙闲状态。媒质的物理载波侦听检测有忙(BUSY)和闲(IDLE)两种状态，这直接由物理层报告。物理层使用信道空闲评估(Clear Channel Assessment,CCA)算法来决定信道是否空闲，CCA通过测量天线口能量值来决定接收信号强度(received signal strength indication,RSSI)。如果这个接收信号的强度低于规定阈值，那么信道就被当作是空闲的，同时MAC层获得一个信

道空闲的信号用于数据的传输。如果这个接收信号的强度高于阈值，数据的传输就被推迟。FPGA根据以太网收发器LXT972A芯片的49引脚给出的RX_DV信号，来报告DSP当前媒质的忙闲状态。虚载波侦听机制由MAC层提供，根据网络分配矢量NAV确定媒质的忙闲状态。

本模块状态主要有四种：CS_NAV(物理载波侦听检测为忙，虚拟载波侦听检测也为忙)、CS_noNAV(物理载波侦听检测为忙，虚拟载波侦听检测为闲)、noCS_NAV(物理载波侦听检测为闲，虚拟载波侦听检测为忙)、noCS_noNAV(物理载波侦听检测为闲，虚拟载波侦听检测也为闲)。其中只有物理载波侦听和虚载波侦听检测都为空闲的时候，媒质状态才被确定为空闲状态，其它三种状态都被确定为忙状态。

5.7 MAC 层协议各模块联合调试

根据项目实际情况，我们初步编写了 MAC 协议各个模块的验证程序，并在三块实验硬件平台之间，通过 Hub 和网线连通模拟来无线网络，然后在该网络上传递数据，进行模块之间的联合调试。

在模拟的环境中，三个站点分别进行同一数据(整数 1~27)的发送和接收，不管是三个站点同时发送，还是只有一个站点发送，在各站点的 FPGA 接收缓冲 RAM 中都有正确的数据。在 PC 机上，我们使用逻辑分析仪(SignalTap II logic Analyzer)来采集 FPGA 芯片数据并进行逻辑分析。SignalTap II logic Analyzer 由 Qutartus II 软件免费提供，通过 JTAG 口对数据进行采样和分析。图 5-10 所示是采样 FPGA 的数据总线和地址总线获得的数据截图，图中 addr 为地址总线信号，data 为数据总线信号且与发送的数据(整数 1~27)一致。

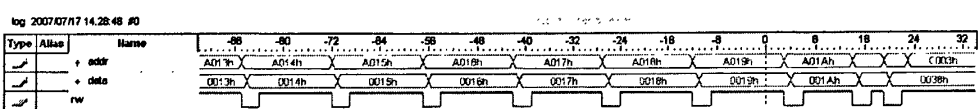


图 5-10 采样 截图

虽然模拟的环境与实际情况差别比较大，但也初步验证了模块编写的正确性，以及一些核心算法的正确性和有效性。接下来，可以通过对程序进行修改和补充，完成与硬件开发板的联调和测试。

第六章 结束语

随着无线接入技术的不断发展,嵌入式处理器的广泛推广以及应用需求的逐渐提升,基于嵌入式处理器的无线局域网(WLAN)通信系统得到了普遍应用。其中,基于 IEEE 802.11 标准的 WLAN,已成为目前主流的宽带无线接入技术,并且在未来可重配置终端中也将扮演重要的角色。

本论文的目的就是深入研究和探讨在 ARM、DSP 和 FPGA 的协作环境下,IEEE 802.11 MAC 层协议软件的设计和实现技术。

本文作者主要完成了以下工作:

- ◆ 研究了 IEEE 802.11 协议标准。系统地分析了 MAC 层的功能和控制协议,深入研究了 DCF 的运行机制和 CSMA/CA 接入方式,以及 PCF 的具体实现过程。
- ◆ 给出了采用 TMS320C5410 DSP 实现的 IEEE 802.11 MAC 层协议设计方案,设计并编写了 MAC 层的调度模块、MAC 层与物理层、LLC 层的接口模块,并对接口部分进行了调试和测试,获得的测试结果基本符合设计要求。
- ◆ 讨论了 STA MAC 层协议的 DSP 实现技术,并对 MAC_Data_Service、MPDU_Generation_STA、Protocol_Control_STA、Transmission、Reception 模块进行了模块化软件设计和代码实现。在模拟的 WM 环境中进行了上述五个模块的初步联调,验证了五个模块及一些核心算法的正确性和有效性。

需进一步完成的工作:

- ◆ 进一步完善 MAC 层的调度模块,改进调度策略,提高工作效率。
- ◆ 进行 MAC 层的管理模块的调试,并与已实现的五个模块之间进行联调。
- ◆ 在硬件环境调试成功后,对主程序进行修改和补充,并完成与硬件开发板的联调和测试。
- ◆ 根据项目的需要,在本论文工作的基础上,对重配置终端中的 WLAN 的设计和实现作进一步的研究和探讨。

致谢

论文工作即将结束，借此机会，我衷心地感谢所有曾经给予我无私关怀和帮助的老师 and 同学们。

首先我要感谢我的导师糜正琨教授在这三年研究生阶段给我的巨大帮助和教诲，从生活上的关怀、学习上的教导，到学术科研工作中的指导，点点滴滴。糜老师严谨的治学态度、不懈的钻研精神、和蔼的指导风格深深地影响着我。这将成为使我终身受益的宝贵人生财富，并激励我今后踏实工作、努力学习不止。

感谢项目组的李曙光、李萧潇、王成法、黄智海以及其他同学。在研究过程中，他们提出了很好的意见和建议，开拓了我的思路，给了我很大的启发和帮助。和他们在一起的日子，我感到很愉快。

我要特别感谢多年来父母的支持和鼓励，没有他们不离不弃的支持，就没有我的今天。

最后，还要感谢参加我的论文答辩的专家和老师，以及其他帮助过我的老师和同学。

参考文献

- [1] 钟章队, 赵红礼, 吴昊, 黄清编著:《无线局域网》, 科学出版社, 2004 年 2 月.
- [2] 金纯, 陈林星, 杨吉云编著:《IEEE 802.11 无线局域网》, 电子工业出版社, 2004 年 8 月.
- [3] 张晓红, SasanSaadat, 乔为民, 敬岚, 苟世哲, 刘军科, “红外通信 IrDA 标准与应用”, 光电子技术, 2003 年第 4 期.
- [4] 方晓农, 何朝霞, “基于 HomeRF 的家庭宽带网络”, 电子工程师, 2001 年第 10 期.
- [5] 李明, 黄载禄, “HiperLAN/2 下一代的无线局域网技术”, 通信技术, 2002 年 04 期.
- [6] The Institute of Electrical and Electronics Engineers, Inc. “Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer(PHY) pecifications”,ANSI/IEEE Std 802.11, 1999 Edition (R2003).
- [7] The Institute of Electrical and Electronics Engineers, Inc. “Part 11: Wireless LAN Medium Access Control(MAC) and Physical Layer (PHY) specifications:High-speed Physical Layer in the 5GHZ Band”,IEEE Std 802.11a-1999.
- [8] The Institute of Electrical and Electronics Engineers, Inc. “Part 11: Wireless LAN Medium Access Control(MAC) and Physical Layer(PHY) specifications:Higher- Speed Physical Layer Extension in the 2.4GHz Band”,IEEE Std 802.11a-1999.
- [9] The Institute of Electrical and Electronics Engineers, Inc.“Part 11:Wireless LAN Medium Access Control(MAC) and Physical Layer(PHY) specifications Amendment 4: Further Higher Data RateExtension in the 2.4 GHz Band”,IEEE Std 802.11g-2003.
- [10] The Institute of Electrical and Electronics Engineers, Inc.“Part 11: Wireless LAN Medium ccess Control(MAC) and Physical Layer (PHY) specifications Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements”,IEEE Std 802.11e-2005.
- [11] The Institute of Electrical and Electronics Engineers,Inc.“IEEE trial-use recommended practice for multi-vendor access point interoperability via an inter-access point protocol across distribution systems supporting ieee 802.11 operation”,IEEE Std 802.11F-2003.
- [12] The Institute of Electrical and Electronics Engineers, Inc.“Part 11: Wireless LAN Medium Access Control(MAC) and Physical Layer (PHY) specifications Amendment 5: Spectrum and Transmit Power Management Extensions in the 5 GHz band in Europe”,IEEE Std 802.11h-2003.

- [13] The Institute of Electrical and Electronics Engineers, Inc. "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 6: Medium Access Control (MAC) Security Enhancements", IEEE Std 802.11i-2004.
- [14] The Institute of Electrical and Electronics Engineers, Inc. "Part 2: Logical Link Control", ANSI/IEEE Std 802.2, 1998 edition.
- [15] The Institute of Electrical and Electronics Engineers, Inc. "Carries Sense Multiple Access with Collision Detection (CSMA/CD) access method and physical layer specifications", IEEE Std 802.3-2000.
- [16] 刘利民, 徐志伟, "一种基于 CSMA/CA 的无线局域网协议的实现方法", 计算机工程与设计, 2002 年 7 月, Vol.23, No.7.
- [17] O'Reilly, "802.11® Wireless Networks: The Definitive Guide", Matthew Gast Publisher, April 2002.
- [18] Samsung Electronics Co., Ltd. "S3C2410X 32-Bit RISC Microprocessor User's Manual, Revision 1.2", 2003.
- [19] 清源科技编著: 《TMS320C54x DSP 应用程序设计教程》, 机械工业出版社, 2004 年 1 月.
- [20] EDA 先锋工作室, 吴继华, 王诚编著: 《Alter FPGA/CPLD 设计》, 人民邮电出版社, 2007 年 6 月.
- [21] Intel Corporation, "LXT972A 3.3V Dual-Speed Fast Ethernet Transceiver Datasheet", January 2001.
- [22] HANNIKAINEN M, KNUUTILA J, HAMALAINEN T, et al. "Using SDL for implementing a wireless medium access control protocol[A]", 2000 Proc of Int Symp on Multimedia Software Engineering[C]. Taipei, 2000. 229 -236.
- [23] Youjin Kim, Haewon Jung, Hyeong Ho Lee, Kyoung Rok Chott, "MAC Implementation for IEEE 802.11 Wireless LAN", Electronics and Telecommunications Research Institute, Korea.
- [24] 官洪运, 徐金娣, 李德敏, "无线局域网 802.11 协议的分析及其 MAC 层实现", 东华大学学报(自然科学版), VOL.30, NO.4, Aug 2004.
- [25] 陈立全, 胡爱群, 胡骏, 李伟征, "IEEE802.11 MAC 层协议栈的均衡多线程实现方法", 应用科学学报, Vol.23, No.2, Mar 2005.