

IEEE802.11MAC 协议 DCF 接收模块的软件实现

刘晓明, 谢李蓉, 邓 锋

(重庆大学 通信工程学院, 重庆 400044)

摘 要: IEEE802.11MAC 的基本访问方法是分布式协调功能 DCF, 各站点 STA 之间处于竞争状态, 允许兼容的物理层之间通过使用 CSMA/CA 和无线媒介忙条件下的随机退避定时来自动进行媒介共享。研究了 IEEE802.11MAC 协议, 在 Windows 环境下, 基于移植到 VC++ 6.0 下的 μ C/OS-II 嵌入式操作系统, 用标准 C 语言对 MAC 站点状态机的接收模块 (Reception 模块及 rotocol_Control_STA 模块中的 Rx_Coordination 子模块) 进行了设计, 并通过了调试与验证, 实现了维护当前站点状态并对物理层接收到的数据进行过滤和分段重组。文中的研究为生产具有自主知识产权的 802.11g 芯片做了铺垫工作。

关键词: 802.11; MAC 层协议; μ C/OS-II 操作系统; 接收模块; DCF 模式

中图分类号: TP311.52

文献标识码: A

文章编号: 1673-629X(2008)11-0172-04

Implementation of Reception Module by Software via DCF in IEEE802.11 MAC Protocol

LIU Xiao-ming, XIE Li-rong, DENG Feng

(College of Communication Engineering, Chongqing University, Chongqing 400044, China)

Abstract: The basic medium access of IEEE802.11 MAC protocol is a DCF that allows for automatic medium sharing between compatible PHYs through the use of CSMA/CA and a random backoff time following a busy medium condition. Studies IEEE802.11MAC protocol. The system is based on μ C/OS-II embedded operation system which is transplanted to VC++ 6.0 under Windows environment. It implements the Reception Module (Reception Module and Rx_Coordination Module described in protocol) by software via DCF with language C. Through the debugging and test, the project is verified that the Reception module maintains the state of the channel, completes the filter and defragmentation of data received by PHY. Makes matting for producing 802.11g chips possessed independent intellectual property.

Key words: 802.11; MAC protocol; μ C/OS-II operation system; reception module; DCF

0 引 言

IEEE802.11 是现在最普及的无线标准之一, 它在宽带无线局域网 (WLAN) 领域已经占据绝对优势, 根据 IEEE802.11g 标准, 在 2.4GHz 频带上可以实现最大 54Mbit/s 的数据传送, 对 WLAN 的发展起到了很大的推动作用。

现在 802.11g 标准芯片的制造技术已经成熟, 并且在成本上具有优势, 802.11g 仍会继续作为无线网络市场的主流技术标准。但是目前的 802.11g 芯片都是国外公司设计的, 通过对 802.11g MAC 层协议进行

研究发现生产具有自主知识产权的 802.11g 芯片是可行的。文中在 Windows 环境下, 基于移植到 VC++ 6.0 下的 μ C/OS-II 嵌入式操作系统, 用标准 C 语言对 DCF 接收模块进行了设计, 为生产具有自主知识产权的 802.11g 芯片做了铺垫工作。

1 MAC 协议中 DCF 模式下接收模块的系统组成

1.1 概 述

分布式协调功能 (DCF) 是 IEEE802.11MAC 的基本访问方法, 各站点 STA 之间处于竞争状态, 允许兼容的物理层之间通过使用 CSMA/CA 和无线媒介忙条件下的随机退避定时来自动进行媒介共享^[1]。在 DCF 模式下, 接收模块维护当前站点状态并对物理层接收到的数据进行过滤和分段重组^[2,3]。

收稿日期: 2008-02-04

基金项目: 国家发改委 CNGI2005 示范工程项目 (CNGI-04-4-2D)

作者简介: 刘晓明 (1963-), 男, 重庆人, 教授, 博士后, 硕士研究生导师, 研究方向为软件无线电、计算机测控、信号与图像处理等。

接收模块与物理层 PHY 接口,接收物理层传来的数据信息及信道状态信息。检查数据有效性,对数据进行解密,地址过滤,重组。由物理层状态信息(Phy-Cca.indication)及当前网络分配矢量 NAV 的值来标识信道状态,完成载波检测,并进行帧间间隔定时。本课程题实现的接收模块还包括 Protocol_Control_STA 中的接收协调 Rx_Coordination 部分,根据收到的帧的类型和参数值生成本站点需反馈的应答帧^[4]。

接收模块 Reception 由 Validate_MPDU、Filter_MPDU、Defragment 和 Channel_State 四个小模块组成,整个系统流程图参考 ANSI/IEEE Std 802.11, 1999 Edition(R2003)。

1.2 各模块功能简介

Validate_MPDU 模块完成 MAC 层对 PHY 接收数据,并进行相关处理,如重置 MAC 层的 rts 响应时间,将数据整合成帧的分段,最后计算校验域的合法性,如果合法则送到过滤器 Filter_MPDU。

Filter_MPDU 模块对 24 位 MAC 头中的信息进行处理,过滤有效的接收帧,保存接收帧,更新重传计数器及缓冲的检测帧。

Defragment 模块完成分段重组。通过检查帧控制域中的多段标志位(moreFrag),控制是在缓冲区中暂存该分段,或是将该分段通过消息 RxIndicate 直接发送给上层模块 Rx_Coordination。

Channel_State 模块基于物理和虚拟载波侦听,获取信道状态进行状态机转换。将状态信息 Busy, Idle 或 Slot 通知发送模块 Transmission。仅当物理载波检测及虚拟载波检测都表明媒介空闲时,才通知发送方当前信道空闲。信道有五种状态,分别为 Wait_IFS(等待帧间间隔),Cs_Nav(物理虚拟忙),Cs_noNav(物理忙虚拟闲),noCs_Nav(物理闲虚拟忙),noCs_noNav(物理闲虚拟闲)。状态转移关系如图 1 所示。

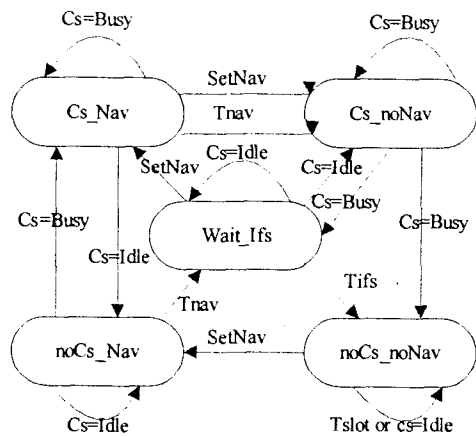


图 1 状态转移图

Rx_Coordination 模块实现接收协调功能。将 Filter_MPDU 发来的 NeedAck, RxCfAck, RxCfPoll 消息根据本 STA 的当前状态进行进一步处理后,以消息 TxCfAck, Ack, CfPoll 的形式通知 Tx_Coordination 发送响应帧。接收 Defragment 模块发来的数据 RxIndicate。根据帧类型对不同的帧分别处理,将数据帧以消息 MsduIndicate 的形式传送给上层模块 MAC_Data_Service,将管理帧和某些控制帧以消息 MmIndicate 的形式传送给 MLME_STA,并在需要时向 Tx_Coordination 发送 Ack, Cts, Cfend, 以便发送响应帧。另外, Rx_Coordination 还与 MLME_STA 交互站点的关联信息。

2 基于 $\mu C/OS-II$ 的 MAC 协议中接收模块的 C 语言实现

2.1 任务划分

根据 I/O 依赖的原则,周期执行功能的原则,以及 $\mu C/OS-II$ 的调度机制,本系统将任务划分为五个: Channel_State, Validate_MPDU, Filter_MPDU, Defragment, Rx_Coordination。与 MAC 状态机系统框图中的小模块划分相一致。

2.2 任务调度及优先级设定

本系统任务间的通信用消息队列的方式实现,一个模块维护一个接收消息队列,一旦消息到达或等待超时,任务就开始执行,否则挂起。考虑到消息接收的即时性和 Channel_State 模块的时间关键性,本系统五个任务的优先级自高向低设定如下: Rx_Coordination, Channel_State, Defragment, Filter_MPDU, Validate_MPDU。这样,高优先级任务在多任务开始执行时就因等待资源而挂起,而一旦低优先级任务完成当前数据处理,向高优先级发出消息后,高优先级任务便可以马上抢占 CPU,达到数据即时处理的目的,效率也比较高。将 Channel_State 设为较高的优先级,可以即时地将信道状态信息通知发送模块,避免不必要的信道闲置或冲突^[5]。

2.3 VC++ 6.0 下的工程构建及文件组织

本课程选择在 Visual C++ 6.0 平台上用 $\mu C/OS-II$ 实现 MAC 层接收模块。VC++ 中,构建好工程后,就可以向工程中添加源程序文件了。为了使系统的结构更为清晰,将源程序文件分为三个部分:在工程根目录下新建文件夹 mac_header,用于存放实现 MAC 状态机的数据类型和算子定义的 C 文件和相应的 package.h 头文件;在工程根目录下新建文件夹 mac,用于存放实现 MAC 状态机各模块的 C 程序文件及相

应的.h头文件;而创建任务,消息队列等需要的 main.c,系统测试所需的 test.c,以及相应的 includes.h 和 test.h,直接放在工程根目录下。这样,整个文件结构层次分明、清晰,有良好的可读性^[6]。

3 系统调试及性能测试

3.1 外围模块的模拟及调试

对系统进行调试时,需要模拟与外围模块的通信。与接收模块有消息交互的模块有 PHY 层, Transmission 模块中的 Data_Pump 子模块,发送协调模块 Tx_Coordination, MAC 子层站点管理实体 MLME_STA, MAC_Data_Service 模块的 MSDU_to_LLC 子模块。

模拟时,将这五个外围模块写成单独任务,任务中以 OSQPend (XX_Q,0,&err)函数等待相应接收消息队列中的消息,并向接收模块发送相应的反馈消息。

整个系统要运行,还需要编写 main 函数,定义任务,分配任务堆栈空间,定义消息队列,为消息队列分配内存,并建立最高优先级任务 AppStartTask,在该任务中初始 CPU,建立完成系统功能所需的各任务及消息队列,任务 AppStartTask 执行后进入睡眠态,不再运行。

3.2 仿真及测试结果分析

MAC 站点状态机接收模块主要完成信道状态维护及从 PHY 层接收数据并进行分段重组的功能。也就是说,作系统测试时,要检验的内容是 Busy, Idle, Slot 这三类消息的发送和接收是否及时准确,以及 PHY 层发送的字节型数据经接收模块过滤后发送的相关响应消息是否正确,重组后的帧数据是否无误。

测试时,在模拟的 PHY 层任务 DownTask 中,以手动方式,用 scanf 函数输入 MAC 头,帧实体数据固定。为了检验结果的正确性,采用打印输出的方式,即利用 μ C/OS-II 系统中的 OS_Printf 函数。测试表明,系统运行状况与 IEEE802.11MAC 协议中描述的站点状态机的 SDL 流图一致,且数据无误。

刚开机时,外围模块的模拟任务都被挂起, Rx_Coordination 模块刚开机时处于 No_Bss 状态,而 mDisable 设为 false,故进入 Rx_Idle 状态。Channel_State 处于初始态,并向 PHY 模块发送 PhyCcarst.request 原语,模拟 PHY 层的 DownTask 任务开始执行,向 Channel_State 模块发送 PhyCcarst.confirm 原语及 PhyCca.indication 原语,传递信道状态为 busy,忙。输入 rts 帧时的程序运行结果如图 2~图 5 所示。

如 2 图所示, b4 为 rts 帧,帧控制域各位都设为 0,地址域都与测试程序中设定的地址相匹配, Duration/ID 为任意合法数据。

```
RxC is running!
No_Bss!
RxC is running!
RxC_Idle!
Mlmetask is running!
channelstate is running!
MediaState is Channel_Start_State!
PhyCcarstrequest is Transmitted!
PHY has received PhyCcarstrequest from ChannelState!
PHY has transmitted PhyCcarstconfirmM to ChannelState!
PHY has transmitted PhyCcaindicationM(busy) to ChannelState!
Please input the Type of the Frame(1):
b4
Please input the Frame control Bits(1):
0
Please input the Duration/ID(2):
11 22
Please input Addr1 of the Frame(6):
0 1 2 3 4 5
Please input Addr2 of the Frame(6):
6 7 8 9 a b
Please input Addr3 of the Frame(6):
6 7 8 9 a b
Please input the sequence control(2):
0 0
```

图 2 设置输入 rts 帧图

```
Please input Addr3 of the Frame(6):
6 7 8 9 a b
Please input the sequence control(2):
0 0
PHY has transmitted PhyRxStartindicationM to ValidateMpdu!
Busy!
Data_Pump has received Busy!
DataPumptask is running!
channelstate is running!
MediaState is CS_NONAV!
channelstate has received PhyCcarstconfirm !
channelstate is running!
MediaState is CS_NONAV!
channelstate has received PhyCcaindication(busy)!
channelstate is running!
MediaState is CS_NONAV!
Defragment is running!
DefragState is DefragStartState!
DefragState is DefragInactive!
Defragment is running!
DefragState is DefragIdle!
FilterMpdu is running!
ValidateMpdu is running!
ValidateMpdu has received PhyRxStartindication from PHY!
ValidateMpdu is running!
```

图 3 PHY 处理过程图

如图 3,PHY 层开始向 ValidateMpdu 模块发送数据,收到 PhyCca.indication(busy)原语后, Channel_State 模块跳转到 Cs_noNav(物理忙虚拟闲)状态,并通知发送模块的 Data_Pump。Defragment 模块开机时为 DefragInactive 状态,由于 mDisable 设为 false 而进入 DefragIdle 状态。

如图 4, ValidateMpdu 模块收到 rts 帧后,重置 rts 超时计时器的值,并通知 Channel_State 等待帧间间隔,即发送消息 UseDifs,同时将分段以 RxMpdu 消息的形式发送给 FilterMpdu 模块。FilterMpdu 模块收到分段后,检查其帧控制域,将功率管理模式以消息 PsIndicate 消息的形式通知 MLME_STA 模块。当前分段经地址过滤后,地址匹配, FilterMpdu 将数据发给

Defragment。Rts 帧为控制帧,无需重组,直接以消息 RxIndicate 的形式发送给 Rx_Coordination。Rx_Coordination 在 Rx_C_Idle 状态下收到 RxIndicate 后,检查帧类型为 rts 帧,则生成一个用于交换的 cts 帧,按协议规定,需等待一个短帧帧间隔后向 Data_Pump 发送 TxRequest,然后进入 Wait_TxDone,即等待 cts 帧发送完成的状态。

```

ValidateMpd is running!
PHY has transmitted PhyRxEndindicationM to ValidateMpd!
ValidateMpd is running!
ValidateMpd has received PhyRxEndindication from PHY!
ValidateMpd has received rts Frame,reset Irts!
ValidateMpd has transmit UseDifs to ChannelState!
channelstate has received UseDifs!
ValidateMpd has transmit RxMpd to FilterMpd!
FilterMpd has transmit PsIndicate(sta_active)!
Mlne has received PsIndicateM message ,it's powermanagemode is 0
The macaddr is :6 7 8 9 a b
FilterMpd has transmit RxMpd to Defragment!
The type of the Frame is control or management,no need to defrag!
RxIndicate is Transmitted!
RxC has received RxIndicate
pdu is rts
RxC is running!
Wait_Sifs!
post TxRequest to DataPump
Data_Pump has received TxRequest message !
c4 0 63 0 6 7 8 9 a b
Data_Pump has transmitted TxConfirm!
DataPumptask is running!
RxC is running!
Wait_TxDone!

```

图 4 接收运行过程图

如图 5,收到 TxConfirm 后,Rx_Coordination 模块重又跳转到 Rx_Coordination 状态,任务挂起,直到队列中有可用消息。PHY 层通知 MAC 层信道空闲,Channel_State 模块跳转到 Wait_Ifs 状态,一个帧间隔后,向 Data_Pump 发送消息 Idle,之后跳转到 noCs_noNav 状态,并每隔一个时隙时间,就向 Data_Pump 发送消息 Slot,以便发送模块完成退避规程^[7]。

通过对测试结果分析证明接收 rts 帧后的处理和标准的流程要求一致,另对输入 beacon 帧和数据帧的测试结果也与标准一致,此处不再给出截图。

4 结束语

在 Windows 操作系统中,利用移植到 VC++ 6.0 上的 $\mu C/OS-II$,实现了 MAC 层接收模块的基本功

能,即数据功能和信道状态控制,并通过了调试。IEEE802.11 无线局域网以其灵活性和可移动性,作为宽带有线接入网的延伸和补充,有极好的应用前景。

```

Wait_TxDone!
RxC has received TxConfirm
RxC is running!
RxC_Idle!
Defragment is running!
DefragState is DefragIdle!
FilterMpd is running!
PHY has transmitted PhyCcaindicationM(idle) to ChannelState!
PHY is running!
channelstate has received PhyCcaindication(idle)!
channelstate is running!
MediaState is Wait_Ifs!
ValidateMpd is running!
Tifs or noTifs!
Idle!
Data_Pump has received Idle!
DataPumptask is running!
channelstate is running!
MediaState is NoCs_NoNav!
Slot!
Data_Pump has received Slot!
DataPumptask is running!
channelstate is running!
MediaState is NoCs_NoNav!
Slot!

```

图 5 接收处理完成图

参考文献:

- [1] Brenner P. A Technical Tutorial on the IEEE 802.11 Protocol [R]. [s.l.]: Breezecom Wireless Communications, 1997: 3-22.
- [2] LAN MAN Standards Committee of the IEEE Computer Society. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications [S]. IEEE-SA Standards Board. ANSI/IEEE Std 802.11, 1999.
- [3] 阮加勇, 黄本雄. IEEE 802.11 DCF 延迟性能分析[J]. 华中科技大学学报: 自然科学版, 2006, 34(4): 27-29.
- [4] 金 纯, 陈林星, 杨吉云. IEEE 802.11 无线局域网[M]. 北京: 电子工业出版社, 2004.
- [5] Labrosse J J. 嵌入式实时操作系统 uCOS-II[M]. 第 2 版. 邵贝贝, 等译. 北京: 北京航空航天大学出版社, 2006.
- [6] Kernighan B W, Ritchie D M. C 程序设计语言[M]. 第 2 版. 徐宝文, 李 志译. 北京: 机械工业出版社, 2004.
- [7] 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[J]. 东华大学学报: 自然科学版, 2004, 30(4): 32-36.

(上接第 158 页)

- [4] 王艳辉, 王相海. 基于提升方案小波的半脆弱水印图像认证算法[J]. 计算机工程与设计, 2007, 28(20): 4955-4958.
- [5] Maeno K, Sun Q, Chang S F. New Semi-Fragile Image Authentication Watermarking Techniques Using Random Bias and Nonuniform Quantization[J]. IEEE Transactions on Multimediam, 2006, 8(1): 32-45.
- [6] 王向阳, 陈利科. 一种新的自适应半脆弱水印算法[J]. 自

动化学报, 2007, 33(4): 361-366.

- [7] Yen Jui-Cheng. Watermark Embedded in Permuted Domain [J]. IEEE Trans. Electronics Letters, 2001, 37(2): 80-81.
- [8] Yen Jui-Cheng. Watermarks embedded in the permuted image[C]//Proc of 2001 IEEE International Conference on Circuits and Systems: Symposium. Sydney, NSW: IEEE, 2001: 53-56.