

基于 $\mu\text{C}/\text{OS}-\text{II}$ 的 IEEE802.11 MAC 协议的实现

刘晓明, 邓 锋, 谢李蓉

(重庆大学 通信工程学院, 重庆 400044)

摘 要: IEEE802.11 协议是目前最具影响力的无线局域网标准,但是目前的芯片都是国外公司设计的,本项目的目的是为了研制具有自主知识产权的 802.11 芯片。介绍了 SDL 描述的 IEEE802.11 MAC 协议,并在理解协议的基础上,提出了在 $\mu\text{C}/\text{OS}-\text{II}$ 实时操作系统中用 C 语言来实现 MAC 协议的方法,并由此在 $\mu\text{C}/\text{OS}-\text{II}$ 操作系统中实现了整个 IEEE802.11 MAC 协议,为生产出具有自主知识产权的 802.11 无线网络芯片奠定了基础。

关键词: $\mu\text{C}/\text{OS}-\text{II}$; 无线局域网; MAC 协议; SDL

中图分类号: TP311

文献标识码: A

文章编号: 1673-629X(2008)08-0026-03

Realization of IEEE802.11 MAC Protocol Based on $\mu\text{C}/\text{OS}-\text{II}$

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

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

Abstract: IEEE802.11 protocol is the most influential wireless LAN standard, but all chips of it are designed by foreign companies. The purpose of this project is to develop own intellectual property rights 802.11 chips. In this paper, the SDL description of the IEEE802.11 MAC protocol is introduced, and on the basis of understanding the protocol, achieved the IEEE802.11 MAC protocol by made in $\mu\text{C}/\text{OS}-\text{II}$ real-time operating system using C language, and laying a foundation for produce own intellectual property rights of 802.11 wireless networking chips.

Key words: $\mu\text{C}/\text{OS}-\text{II}$; WLAN; MAC; SDL

0 引 言

随着信息技术的飞速发展,人们对网络通信的需求不断提高,传统局域网络已经越来越不能满足人们的需求,而无线局域网(WLAN)因其具备可移动性、接入灵活、维护方便等特点,得到了越来越广泛的应用。

IEEE802.11 协议是目前最具影响力的无线局域网标准^[1]。基于 IEEE802.11 协议的芯片制造技术现在已经非常成熟,但是目前的芯片都是国外公司设计的,通过对 802.11 协议的研究发现生产具有自主知识产权的 802.11 芯片是完全可行的。媒体访问控制(MAC, Medium Access Control)协议是 IEEE802.11 的一个重要组成部分,文中提出了使用开源的实时操作系统 $\mu\text{C}/\text{OS}-\text{II}$ 来实现 IEEE802.11 MAC 协议的一种方法,并在此基础上,使用 $\mu\text{C}/\text{OS}-\text{II}$ 验证并实现了

IEEE802.11 MAC 协议,为下一步生产出具有自主知识产权的 802.11 无线网络芯片(MAC 层和 PHY 层)奠定了基础。

1 MAC 协议简介

IEEE802.11 协议主要由两部分组成:物理层(PHY)和数据链路层的 MAC 子层。MAC 子层负责提供在不可靠的无线媒质上可靠的传输用户数据的机制,它的主要功能包括对无线媒质的公平访问、终端同步、用户认证、数据加密、协议数据单元编址、定义帧的格式、错误校验、分割和拼接等问题。

MAC 协议主要包括基本的分布式协调功能(DCF)和可选的点协调功能(PCF)。DCF 类似于传统的分组网,支持异步数据传输等异步业务,所有要传输数据的用户拥有平等接入网络的机会,通过竞争模式以取得资料传输的权利,而 PCF 通过某终端或服务站点来主控分配权,主要用于传输实时业务。所有站点均支持 DCF。在 Ad-hoc 网中,DCF 独立工作;在基本结构网中,DCF 可独立工作也可与 PCF 共同工作^[2]。

收稿日期:2007-11-18

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,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。

2 MAC 协议的 SDL 描述

IEEE802.11 MAC 协议采用 SDL(Specification and Description Language)描述。

SDL 是一种抽象化的、面向对象的、系统级的功能规格与描述语言,采用文字和图形结合的方式提供清晰准确的系统说明和描述,主要用于电信领域。采用 SDL 描述的系统包含若干个功能块(block),每个功能块内至少有一个进程(process),进程中的 SDL 行为用扩展有限状态机描述,它由一系列的状态、输入信号和定时器信号组成。外部的输入信号和进程内的定时器都可以导致进程中的状态转移,在此过程中可能产生输出信号,从而触发其他进程的事件^[3]。

IEEE802.11 MAC 协议的 SDL 描述可以分为以下几个功能模块:

1)MAC 数据服务模块:向 LLC 层提供 MAC 层的数据服务接口。模块主要由 MSDU_to_LLC 和 MSDU_from_LLC 这两个进程组成。即发送 MAC 层服务数据单元 MSDU 给 LLC 层,和从 LLC 层接收 MAC 服务数据单元 MSDU。

2)MPDU 生成模块: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 进行分割、封装,生成一个或多个符合 IEEE802.11 MAC 数据帧结构的协议数据单元 MPDU。如果需要,还将对数据帧进行加密操作,以提高数据传输的可靠性。此外,模块还完成 MPDU 的排队管理功能。该模块主要包括 Prepare_MPDU 进程和 PM_Filter 进程。

3)协议控制模块:协议控制模块是 MAC 协议的核心模块,分为 Tx_Coordination 进程和 Rx_Coordination 进程。其中 Tx_Coordination 进程负责 RTS 帧和 ATIM 帧的生成,发送来自上层的 MAC 协议数据单元,它是整个数据帧发送流程中最关键的部分,其主要功能是通过一种控制调度过程最大限度地利用信道把 MPDU 在最短的时间里发送出去,与之协同工作的是 Backoff_Procedure 退避过程子模块。Rx_Coordination 进程根据接收到 MPDU 类型的不同而进行不同处理,如果是 RTS 帧和 Data 帧,必须分别给出 CTS 和 ACK 帧进行确认,并且传递数据帧到达 MAC 数据服务模块,传递管理帧到达 MLME 管理实体,生成接收到 ACK、CTS 以及 CF-Poll 等帧的指示信息给 Tx_Coordination 进程。

4)发送模块:将 MAC 帧以字节流的形式发送到物理层,完成实际的发送过程。这个模块中要完成对发送帧产生 CRC 校验,向发送的 Beacon 帧中加入时间戳用来进行时间同步等。同时这个模块还负责处理底层获得的当前信道的状态,完成协议要求的随机退避

功能。该模块主要包括两个进程:Data_Pump 进程和 Backoff_Procedure 进程。

5)接收模块:对从物理层接收到的帧进行 CRC 校验,如果正确接收的话,则进行地址过滤,丢弃目的地址不是自己的帧。如果数据是经过分段的,需要进行数据的重组,然后将该帧送往协议控制模块的接收协调模块进行分类处理。同时,这个模块还要提取接收帧中的信道保留信息,并把信道的状态送往发送模块来协调退避功能的完成。该模块主要包括以下几个进程:Channel_State 进程,Validate_MPDU 进程,Filter_MPDU 进程,以及 Defragment 进程^[4]。

在对协议进行充分的研究和理解的基础上,决定采用实时操作系统 $\mu\text{C}/\text{OS}-\text{II}$ 来实现 IEEE 802.11 的 MAC 协议。

3 在 $\mu\text{C}/\text{OS}-\text{II}$ 中的实现

$\mu\text{C}/\text{OS}-\text{II}$ 是一个源代码公开的实时操作系统内核。它具有结构简单、执行效率高、占用空间小,实时性能优良,系统移植简便等优点^[5],因此,选择了 $\mu\text{C}/\text{OS}-\text{II}$ 作为实现 IEEE802.11 MAC 协议的软件平台。

IEEE802.11 MAC 协议是采用 SDL 描述的,SDL 是面向对象的描述性语言,而 C 语言是面向过程的编程语言,因此,在 $\mu\text{C}/\text{OS}-\text{II}$ 中用 C 语言来实现 MAC 协议时,需要做一些灵活的处理。

3.1 数据类型的实现

SDL 描述的 MAC 协议中大部分的数据类型都能在 C 语言中找到对应的数据类型,这主要由它的位数来决定,如 Octet 对应着 INT8U(即 unsigned char)、Integer 对应着 int 等。C 语言中没有只占 1 位的数据类型,可以用 INT8U 来处理 SDL 中的 BIT 类型。

Octetstring 在协议中是非常常用的一个数据类型,它是一个长度不确定的 8 位数组,C 语言中定义数组必须给出它的长度,因此不能直接用数组来定义这个类型。可以通过定义这样一种自定义类型来处理它:

```
typedef struct  
{  
    Integer length;  
    INT8U * ptr;  
} Octetstring;
```

在使用时,先定义一个数组(其长度根据具体情况确定),然后将 Octetstring 类型变量的指针指向该数组的第一个元素,并对 Octetstring 变量的 length 域赋值。

对于各种数据类型新增的操作,用函数来予以实现。需要注意的是,如果函数的返回值是 Octetstring

类型,可能会因为数组的复制而造成函数的效率低,因此采用了这样的方法:在函数中增加一个指向 Octetstring 类型变量的指针作为参数,函数直接修改这个指针指向的 Octetstring 型变量,而返回值为 void。

如 SDL 描述中为 Octetstring 类型增加了一个操作叫 mkOS,其描述为:

```
mkOS : Integer, Integer -> Octetstring;
```

在用 C 语言实现时,通过建立一个名为 mkOS 的函数来实现这样的功能,函数的原型为:

```
void mkOS (Integer i, Integer k, Octetstring * os);
```

3.2 进程的实现

SDL 中的进程是借鉴的操作系统的概念,因此可以直接用 μ C/OS-II 的任务来处理它。根据对协议的研究分析,把整个 MAC 协议分为 MSDU_to_LLC, MSDU_from_LLC, Prepare_MPDU, PM_Filter, Tx_Coordination, Rx_Coordination, Data_Pump, Backoff_Procedure, Channel_State, Validate_MPDU, Filter_MPDU, Defragment 等 12 个任务。为了调试的方便,又建立了两个任务 LLC 和 PHY,用来分别模拟 LLC 层和 PHY 层,向 MAC 层发送所需要的消息。对于多任务实时操作系统来说,任务优先级的分配往往对系统的性能有很大的影响。在实现 MAC 协议时,任务运行完后都是通过 OSQPend() 来等待消息队列中的消息而挂起的(见第 3.3 节),所以处于就绪态的任务通常只有刚接收到消息的那一个任务,这样,任务的优先级问题就显得并不那么重要,在后来的调试中也证实了这一点。

每个任务都有很多不同的状态,可以把每个状态分别写成一个函数,然后为每个状态分配一个 ID,设置一个变量为 status_ID,程序无限循环的判断变量 status_ID 的值,并调用此值对应的状态函数。状态间的转移通过修改 status_ID 来实现。

3.3 进程间通信的实现

进程间的信号传递可以用消息队列来实现。为每个任务建立一个消息队列和一个用于临时保存消息的数组,其他任务发给该任务的消息都发到这个消息队列,任务需要等待某个消息时,会先在消息数组中查找是否存在此消息,如果存在,就将消息取出并进行相应的处理,如果不存在,就用 OSQPend() 将自己挂起。当该任务的消息队列收到消息时,任务就会立即转为就绪态。任务运行后,先判断此消息是否为当前需要的消息,如果是,则进行相应的处理,如果不是,则把这个消息保留在消息数组中,因为这个消息可能会在任务处于下一个状态时需要。

消息结构体的定义如下:

```
typedef struct
{
    MsgType sig_id;
    union
    {
        void * par_ptr;
        Integer par_copy[MSG_BORDER_LEN];
    };
} Message;
```

sig_id 指示消息的类型,任务根据它来判断该消息是否就是任务当前所需要的消息。消息的参数由联合体给出,如果参数少,就可以将参数放在数组 par_copy[] 中,如果参数较复杂,那么就可以通过指针来实现。如消息 TxRequest 有两个参数:tpdu 和 txrate,可以通过定义如下结构体:

```
typedef struct
{
    Frame * rtsdu_ptr;
    Rate txrate_ptr;
} TxRequest;
```

然后将此类型变量的地址赋给消息的 par_ptr 来初始化这个消息。

3.4 消息优先级的实现

在 IEEE802.11 MAC 协议中,消息有两种不同的优先级:普通优先级和高优先级。而在 μ C/OS-II 中,消息是没有优先级的,不过 μ C/OS-II 提供了 OSQPost() 和 OSQPostFront() 两个不同的函数来发送消息。OSQPost() 函数和 OSQPostFront() 函数的区别在于:OSQPost() 发送的消息是 FIFO(先入先出)的,而 OSQPostFront() 发送的消息是 LIFO(后入先出)的,这是因为 OSQPostFront() 在插入新的消息到消息队列时,使用 OSQOut 作为指向下一个插入消息的单元的指针,而不是 OSQIn。

这样,可以利用这两个函数的不同来实现消息优先级的区分。对于普通优先级的任务,使用 OSQPost() 函数发送,而对于高优先级的消息,使用 OSQPostFront() 函数来发送。这样,当 OSQPend() 函数从消息队列中取消息时,就会最先取到由 OSQPostFront() 函数发送的消息,这样就相当于提高了这条消息的优先级。

3.5 进出口值的处理

在 IEEE802.11 MAC 协议中,有一些参数是任务间共享的,如 mFxiP、cNack 等,SDL 的处理规则是由共享此参数的任务 export,访问此参数的任务 import,每当一个任务要共享另一个任务的某一参数时,都将

(下转第 31 页)

阈值提升小波去噪图像,图 5 为平滑梯度图像后的分水岭变换图像。图 6 为合并后的图像。可以看出,提

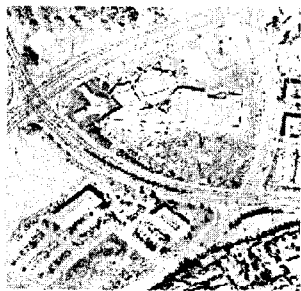


图 2 遥感原图像



图 3 含噪图像



图 4 提升小波去噪图像

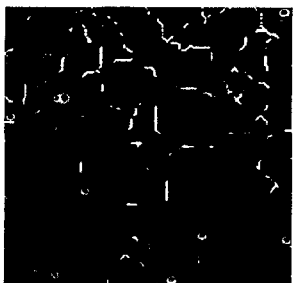


图 5 平滑梯度图像后的分水岭变换



图 6 合并后的图像

升小波去噪图像的噪声点明显减少,但图像有模糊不清,将分水岭分割图像嵌入提升小波去噪后的图像能够得到更好的效果。实验结果表明,该方法不但能够降低图像的噪声点,而且能够使图像重要特征变得清晰,大大提高了图像的去噪效果和清晰度。

参考文献:

- [1] 余锦华,陈建华,施心陵.基于提升小波和 Context 模型的图像无失真编码方案[J].计算机工程与应用,2004,40(3):49-51.
- [2] 罗文斌,张立明,陈鑫.第二代小波在医学图像无损压缩中的应用[J].计算机工程,1999,25(10):80-82.
- [3] Fahmy G F, Panchanathan S. A lifting based system for compression and classification trade off in the JPEG2000 framework[J]. J Vis. Commun. Image R., 2004, 15: 145-162.
- [4] Ooninx P J, de Zeeuw P M. Adaptive lifting for shape-based image retrieval[J]. Pattern Recognition, 2003, 36: 2663-2672.
- [5] Ercelebi E. Electrocardiogram signals de-noising using lifting-based discrete wavelet transform[J]. Biology and Medicine, 2004, 34: 479-493.
- [6] 王志武,丁国清,颜国正,等.自适应提升小波变换与图像去噪[J].红外与毫米波学报,2002,21(6):447-450.
- [7] 孙红星,王蓉,赵楠楠,等.基于小波提升和形态学的图像边缘检测方法[J].系统仿真学报,2006,18(S2):869-871.
- [8] Hua Chun-Kuang, Yana Wen-Ming, Chung Kuo-Liang. Efficient cache-based spatial combinative lifting algorithm for wavelet transform[J]. Signal Processing, 2004, 84: 1689-1699.

(上接第 28 页)

进行一次信号的交换。而在 C 语言中,可以简单地把这些参数设置成全局变量^[6]。实际测试表明,这样的简化是完全可行的。

4 结束语

在 $\mu C/OS-II$ 中使用 C 语言来实现 SDL 描述的 IEEE802.11 MAC 协议时,应该在充分理解协议的基础上,根据功能进行灵活的处理,而不是简单地寻求 SDL 与 C 语言间的一一对应关系。给出了一些实现的具体方法和思路,并由此在 $\mu C/OS-II$ 操作系统中实现了整个 IEEE802.11 MAC 协议,为下一步与 PHY 层一起生产出具有自主知识产权的 802.11 无线网络芯片奠定了基础。

参考文献:

- [1] 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.
- [2] 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.
- [3] 胡田立,杨洋,金德鹏,等.基于 SDL 的无线局域网 MAC 协议设计与实现[J].计算机工程,2006,32(7):267-268.
- [4] 徐会明.基于 IEEE 802.11 协议的无线局域网 MAC 层研究与实现[D].武汉:华中师范大学,2001.
- [5] Labrosse J J. 嵌入式实时操作系统 $\mu C/OS-II$ [M]. 第 2 版.邵贝贝等译.北京:航空航天大学出版社,2006.
- [6] 和诚凯,申振宁,刘乃安,等.802.11 协议固件实现中的 SDL 与 C 语言之间的映射[J].计算机工程,2004,30(9):97-98.