

## 基于ARM的IEEE 802.11b MAC层协议IP核设计

李安新 陈曦 郑继荣 周祖成

清华大学电子工程系(100084)

2008-08-11

**摘要:** 介绍了IEEE 802.11b MAC层协议的IP核设计,提出了基于32位微处理器ARM7TDMI的系统设计方案,阐述了系统硬件平台的设计、结构及主要模块单元的功能;给出了利用形式描述语言SDL进行MAC层协议设计开发的完整设计流程;阐述了软件的层次结构,并针对设计中遇到的代码生成器的选择、设计优化、与实时操作系统(RTOS)的集成和环境函数编写等问题进行了深入讨论。

**关键词:** MAC层协议 无线局域网 SDL ARM

目前各种协议的设计实现,大多数是基于微处理器、微控制器或DSP的嵌入式系统。ARM是ARM公司推出的高性能32位RISC微处理器,具有业界领先的体系结构,被广泛应用于各种系统设计中。基于ARM的开发通常使用C、C++或汇编语言。笔者采用了更高级的形式语言SDL,大大缩短了协议的开发周期并提高了设计的可移植性。无线局域网是目前通信领域的一个研究热点,无线局域网的协议是非常典型的协议。本文将详细讨论使用SDL进行无线局域网802.11b MAC层协议的设计以及基于微处理器ARM7TDMI的系统实现方案。其设计方法具有普遍意义。

### 1 系统硬件平台设计及功能描述

整个系统硬件平台的设计主要分MAC层和物理层两部分。硬件平台的结构框图如图1所示。其中MAC层部分主要围绕32位微处理器ARM7TDMI和AMBA总线设计,其主要的单元模块和功能如下:

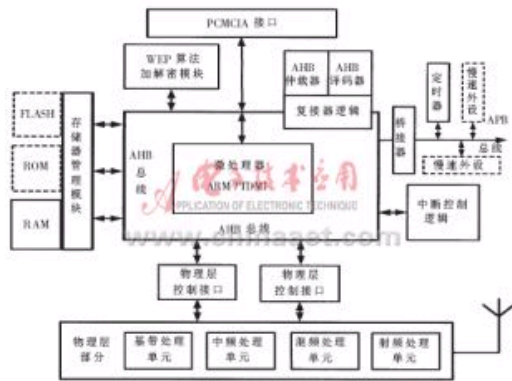


图1 硬件平台结构框图

- PCMCIA接口,此接口为硬件平台和主机的通信接口。其设计遵循PC Card标准(版本5)。
- WEP算法加解密模块,此模块用硬件实现IEEE 802.11b MAC层协议定义的有线网等效加/解密算法。
- 物理层数据接口,此接口用于完成物理层与MAC的数据交互操作,内部设计有发送和接收FIFO,用于完成数据的接收发送缓存。
- 物理层控制接口,此接口用于完成MAC层对物理层的控制功能。例如无数据收发时,可以通过此接口控制物理层部分转入节能状态。
- 存储器管理模块,此模块用于实现对系统所有存储器(如FLASH、ROM、RAM)的管理,处理器通过此模块对存储器进行访问。
- 中断控制逻辑,此模块用于对系统中各模块产生的中断信号进行控制和管理。
- 微处理器单元ARM7TDMI,用于完成与主机的通信,负责整个系统的控制和管理。

物理层部分的设计主要分为四个单元:

- 基带处理单元,主要用于完成基带信号的处理操作,如Rake接收、均衡、数/模、模/数转换等。
- 中频处理单元,主要用于完成信号的调制解调处理。
- 混频处理单元,主要用于完成射频中频的变频处理。
- 射频处理单元,主要完成射频信号的功率放大处理。

MAC层的主要模块单元(外部存储器单元除外)用Verilog硬件描述语言设计并用Xilinx的FPGA Virtex II xc2v3000编程实现。物理层部分则主要用Intersil公司的PRISM芯片组及少量外围电路设计实现。

### 2 SDL及软件开发平台

SDL是一种层次化的描述语言,采用结构化和自顶向下的设计原则,把系统规范分为系统、块、子块、进程、服务和过程几个层次进行描述。系统、块和子块是静态描述,用于描述系统的结构;进程、服务和过程是动态描述,用于描述系统的行为。SDL的理论模

### 热点专题

- 信心09,冬天来了,春天还会远吗?
- 低功耗技术,是鸡还是蛋?
- 华北计算机系统工程研究所(电子六所)总结表彰暨春节联欢会
- Powerwise高效能解决方案
- 2008Security China中国国际社会公共安全产品博览会
- 视频信号处理技术
- 2008嵌入式技术创新及...
- 2008飞思卡尔技术论坛
- Altera公司SOPC...
- 第十届高交会电子展
- 科技闪耀北京奥运
- ADLINK DAY—2008年量测与自动化技术国际高峰论坛
- 中国电子学会Xilinx杯开放源码硬件创新大赛
- 赛灵思公司Virtex-5系列FPGA
- 3G知识
- IPTV
- 触摸屏技术
- RoHS

### 杂志精华

- 基于CC2430的无线传感器...
- 无线传感器网络应用系统综述
- 无线传感器网络在野外测量中的...
- 基于竞争的无线传感器网络
- 用于矿井环境监测的无线传感器...
- 具有自适应通信能力的无线传感...
- 基于传感器网络技术的深孔测径...
- 基于无线传感器网络的家庭安防...
- 基于ATmega128L与C...
- 无线传感器网络中移动节点设备...

型是通信扩展有限状态机,每个进程都是一个通信扩展有限状态机。

SDL与常用的高级语言有很好的接口,如可以从SDL描述的系统规范直接导出C、CHILL甚至VHDL语言,以嵌入式系统和软硬件混合系统实现;在进行规范定义时,SDL又允许嵌入高级语言,如C/C++语言、OMT/UML对象模型、ASN.1或CORBA/IDL数据类型定义等。因而在多个层次上对系统进行准确的规范和描述。因为SDL的上述特性,目前已被越来越多地用于协议软件的开发实现<sup>[2][3]</sup>。

本文选用Telogic公司的SDT4.3和ARM公司的ADS1.2作为主要的软件开发工具。使用SDT进行协议软件的开发步骤如图2所示。



图2 用SDT工具进行协议软件开发流程

在系统行为定义阶段,要特别注意代码生成器对SDL设计的约束,例如对于C advanced生成器不支持信道子结构、信号细化等;对于C micro生成器不支持连续信号、服务、优先输入/输出等。在系统行为分析仿真调试阶段,使用C basic/C advanced代码生成器产生系统代码,与SDT提供的仿真内核一起编译链接,得到系统行为的仿真模型,然后可以使用text、SDL、MSC等多种跟踪方式进行仿真调试。在系统行为验证时,使用相同的代码只不过编译链接时加入SDT提供的验证内核,可得到系统行为的验证模型,可以使用自动状态空间遍历、覆盖率分析等方式进行系统行为的验证。仿真和验证都无环境函数,由仿真器和验证器充当系统的环境,产生和接收与系统交互的信号。在系统行为经仿真验证正确后,可以应用C advanced/C micro生成器产生面向应用的系统代码和环境函数。

应用SDT生成的代码经过适当修改和处理后可以输入ARM开发工具ADS,进行嵌入式系统的开发,其方法和设计流程详见后。

### 3 系统软件的设计和开发

系统的软件设计主要分为三部分:协议软件、驱动软件和接口软件。其中协议软件部分主要用于实现IEEE 802.11bMAC层协议定义的各种服务(如授权、关联等)和算法(如DCF、PCF、时钟同步算法等)。这一部分软件采用图2所示的设计流程,完全使用形式描述语言SDL进行设计实现,并使用SDT的代码生成器将SDL的系统描述转换成面向应用的C/C++代码。驱动软件部分主要用于实现对硬件设备的驱动功能。如PCMCIA接口驱动,这一部分软件用C/C++语言进行设计实现。接口软件部分主要完成SDL转换出的系统代码与RTOS及硬件平台的接口功能。这一部分软件借助于代码生成器产生的环境函数,用C/C++语言设计实现。软件部分的层次结构如图3所示。



图3 软件协议层次结构图

### 4 与ADS接口及软件后端开发

从SDL转化出C/C++代码后,可使用ARM的开发工具ADS进行后续的软件开发。其与SDT工具的接口及开发流程如图4所示。

由SDL描述转换出的C/C++代码,与环境函数、Runtime库以及C/C++库一起用ARM的编译器编译,产生面向ARM的可执行程序。其中,环境函数主要用描述系统运行的具体物理环境。由SDT工具根据用户所作的系统描述自动生成一个结构框架,然后用户根据所采用的具体硬件平台环境编辑这个文件,以描述真实的系统工作环境。Runtime库主要包含SDL预定义的数据类型、操作符的实现、调度函数、运行错误处理等信息。SDT工具提供简单的Runtime库。C advanced/C micro代码生成器都有各自对应的Runtime库。C/C++库是ADS本身携带的函数库,主要包含ISO标准定义的C/C++库函数。

在使用ARM编译器编译后,产生ARM的目标文件(.o文件)。如果还有用ARM汇编指令编写的汇编程序,可用汇编器(armasm)汇编,产生相应的目标文件。把所有的目标文件用链接器[armlink]链接,便可得到能在ARM7TDMI处理器上执行的映像文件(.aof文件)。这时可以用ADS提供的调试工具AXD进行程序的调试。

因为SDL的系统设计在高层进行了仿真和验证,所以调试的主要工作集中在驱动、中断和环境函数的调试上。对SDL系统的调试主要是通过仿真确定对系统性能影响严重的模块并对其进行优化以及系统在实时运行状态下能否满足设计要求。如果在调试中发现问题需要修改SDL的系统设计,可重新执行如图4所示的流程,直到满足设计要求。

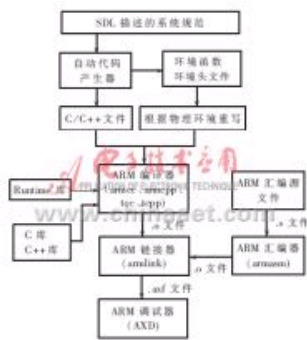


图4 使用SDF与ADS进行协议软件的开发流程

## 5 问题及分析

(1)代码生成器的选择问题<sup>[4]</sup>。SDT提供三种代码生成器,即C basic、C advanced和C micro。其中C basic是最简单的代码产生器,一般只用于在SDT开发环境中仿真系统的行为。C advanced和C micro是面向应用的代码产生器,可以产生高效的代码。C advanced支持几乎所有的SDL概念,对SDL设计的约束较少。C micro可以产生性能更优越、占用存储器空间更小的代码,代价是对SDL设计的约束较多,例如不支持使能条件、连续信号、过程的继承等<sup>[5]</sup>。

(2)设计优化问题。在进行系统设计时,应注意的设计要点有:当输出信号时,应带上接收进程的PID,这样可以减少对信号进行路由的开销;信号应尽量少带占用大量存储空间的参数,因为在信号传递时同时复制信号的参数,占用大量存储空间的参数将占用过多存储空间并引起附加延时;两个状态之间的传输操作不宜过多,否则会带来较大延时(可以用实时仿真确定影响时延的关键路径并进行优化);如果系统中有比较复杂的模块,对时延又有严格要求,可以用C/C++或汇编单独编写,也可用硬件完成,如图1中的WEP算法模块。

(3)与RTOS的集成问题。用户可以不使用RTOS,而使用SDT提供的缺省内核程序,也可以自己编写所需的调度算法、内存管理、中断处理等。SDT工具直接支持的RTOS有Solaris(Posix 4)、Win32、VxWorks和OSE del ta。SDT提供三种与RTOS的集成方式,即松散集成、线程集成和紧集成。松散集成把整个系统映射为OS的一个任务,使用SDT提供的标准内核进行调度,每次执行一个完整的传输。因此松散集成的最大延时是SDL设计中状态之间传输的最长时间。紧集成把每个进程映射为一个OS的任务,可以使用OS的调度算法,给不同的任务以不同的优先级执行,因而性能好于松散集成。线程集成则是两者的折衷。

(4)环境函数的编写。环境函数主要是完成四个函数的编写。xInitEnv():主要用于完成系统的初始化操作。xInEnv():主要用于接收来自硬件或RTOS的信号并转换成SDL系统所需要的信号。调度器每隔一段时间轮询一次xInEnv()函数,检查是否有信号输入。如果有信号输入则发送适当信号给SDL系统。xInEnv()函数中不能使用阻塞函数,如getchar()等。阻塞函数会妨碍调度器处理SDL系统。xOutEnv():主要用于接收来自SDL系统的信号并转换成对RTOS的信号或对硬件的操作。当SDL系统有信号输出时,则调用xOutEnv()函数,根据用户编写的代码产生相应的物理信号或硬件操作。xCloseEnv():用来完成关闭环境的操作。

## 参考文献

- 1 ITU-T Recommendation Z.100. Programming Languages: CCITT Specification and Description Language[SDL],1993
- 2 J.M.Alvarez,M.Diaz,L.Llopis,et al. Deriving hard realtime systems implementations directly from SDL specifications. Proceedings of the Ninth International Symposium on Hardware/Software Codesign,2001:128~133
- 3 M.Hannikainen,J.Knuutila,T.hamalainen,et al. Using SDL for implementing a wireless medium access control Protocol. International Symposium on Multimedia Software Engineering,2000:229~236
- 4 M.Hannikainen,A.Takko,J.knuutila,et al. SDL-to-C conversion for implementing embedded WLAN protocols, Industrial Electronics Society,2000. IECON 2000. 26th Annual Conference of the IEEE,2000;(4):2455~2460
- 5 Telelogic Tau 4.3 User' s Manual.Telelogic AB. Sweden, 2001

## 在线联系

[添加到收藏夹](#)

关于“[基于ARM的IEEE 802.11b MAC层协议IP核设计](#)”,我有如下需求或意向:

用户名:  密码:  验证码:  5829 [欢迎注册](#)

## 相关应用

- 如何在SDL中调用C语言函数
- 32位ARM嵌入式处理器的调试技术
- ARM CPU S3C44BOX与C54X DSP的接口设计
- 内嵌ARM核的FPGA芯片EPXA10及其在图像驱动和处理方...
- 基于ARM-μCLinux嵌入式系统启动引导的实现
- 基于ARM的CAN总线智能节点的设计

《电子技术应用》编辑部版权所有

地址：北京海淀区清华东路25号电子六所大厦

联系电话：82306084 / 82306085 传真：62311179 京ICP备05053646号

推荐分辨率1024\*768 IE6.0版本

