

基于 UML 的 SoC 建模设计方法研究

张海涛, 龚龙庆

(西安微电子技术研究所, 陕西 西安 710075)

摘要:随着集成电路制造工艺的发展, 嵌入式计算机应用向着 SoC 的方向发展。为了适应制造工艺对 SoC 设计能力的要求, 提高 SoC 的设计效率, 成为了很紧迫的必要任务。采用统一的 SoC 系统级建模语言 SystemC、软/硬件协同设计技术、基于 IP 核复用等技术的 SoC 设计流程, 在一定程度上满足了 SoC 设计要求。在现有 SoC 设计流程基础上, 结合 UML 的模型驱动框架(MDA)设计方法, 在当前的 SoC 设计流程的系统需求规约描述、硬件实时反应式系统建模、软件模块设计实现中采用 UML 针对 SoC 的轻量型扩展特性, 可以很大程度地改进提高 SoC 的设计流程效率。

关键词: SoC 设计; UML; SystemC

中图分类号: TP39

文献标识码: A

文章编号: 1673-629X(2008)03-0145-03

Research of SoC Modeling Design Method Based on UML

ZHANG Hai-tao, GONG Long-qing

(Xi'an Institute of Microelectronics Technology, Xi'an 710075, China)

Abstract: As the development of the IC manufacture technic, the application of embedded computer system developed in a trend to SoC (system-on-a-chip) application. To fit a higher SoC design ability which required by manufacture technic, improving the SoC design efficiency becomes an urgent and essential task. Involving many technologies such as unified SoC system-level-modeling-language, SystemC, the hardware/software co-design technology, and IP core based reuse technology, the current SoC design flow basically meet the SoC design requirement. Introducing the mode-driven-architecture(MDA) design method of UML into the current SoC design flow, and according to the UML 2.0 profile for SoC v1.1 specification, using the UML modeling language in the system-requirement-specification describing, hardware real-time reactive system modeling, software module designing and implementation, the UML modeling language greatly improved the SoC design flow and the SoC design efficiency.

Key words: SoC design; UML; SystemC

0 引言

嵌入式计算机的设计正在从单立元器件向高集成度的系统级芯片设计方向发展, 而目前的集成电路制造工艺进入深亚微米、超深亚微米工艺后, 为系统级芯片的生产制造提供了可能。System-on-a-chip (SoC), 这种系统级芯片又称为片上系统, 是集成硬件模块和软件模块, 以处理器为核心, 附带相应外围接口模块、嵌入式实时操作系统、嵌入式应用软件的一种集成电路芯片。SoC 中的硬件模块由可复用的 IP 核代替原来构成嵌入式计算机的各种分立功能模块, 软件则由可以用于片上系统的可剪裁嵌入式操作系统及应用程序构成。

SoC 的设计, 既有专用集成电路、VLSI 芯片的设计特点, 又有嵌入式计算机的系统设计特点。总体来说, 系统级设计体现着 SoC 的最大设计特征, 同时采用可复用的 IP 核以搭积木的方式来构建系统, 软硬件的统一描述、软硬件的协同设计是 SoC 设计的另外一些重要特征。目前的 SoC 设计, 既要解决软硬件统一描述的问题, 又要解决可执行的系统规约在验证通过后的硬件逐步细化、软件的细化实现、软硬件协同设计验证的问题。

SystemC 语言的出现, 较好地解决了 SoC 的软硬件统一建模问题^[1]。UML 语言在提出针对嵌入式应用的建模扩展之后, 又针对 SoC 的设计, 提出了专门为 SoC 进行设计建模的扩展机制。

1 SoC 的传统设计方法

嵌入式计算机的发展, 越来越向微型化、高集成度的趋势发展。集成电路的制造工艺, 向深亚微米、超

收稿日期: 2007-06-08

作者简介: 张海涛(1975-), 男, 陕西商洛人, 硕士研究生, 研究方向为嵌入式计算机、软硬件协同设计、SoC 设计等; 龚龙庆, 研究员, 主要研究方向为嵌入式计算机设计、软硬件协同设计、SoC 设计等。

深亚微米发展,给集成电路的高集成度提供了可能,同时也给嵌入式计算机的发展、设计、制造带来新的变化和影响。系统级芯片 SoC 的出现,是未来嵌入式计算机应用的一个方向。

SoC 的设计,和 VLSI 与 ASIC 的设计,具有很大的相似性。不同于纯集成电路 VLSI 和 ASIC 的设计,SoC 是一种软件和硬件结合更为紧密的嵌入式计算机系统。在 SoC 上,可以集成多个不同类型的系统子模块,比如通用处理器、数字信号处理器、浮点处理单元、构成电子系统中的硬件部件 IP 核、片上系统的嵌入式操作系统(EOS)、面向特定应用的应用软件模块等。目前的 SoC 设计,软硬件协同设计方法在 SoC 的设计过程中,起着极为重要的作用。通常的 SoC 设计流程如图 1 所示。

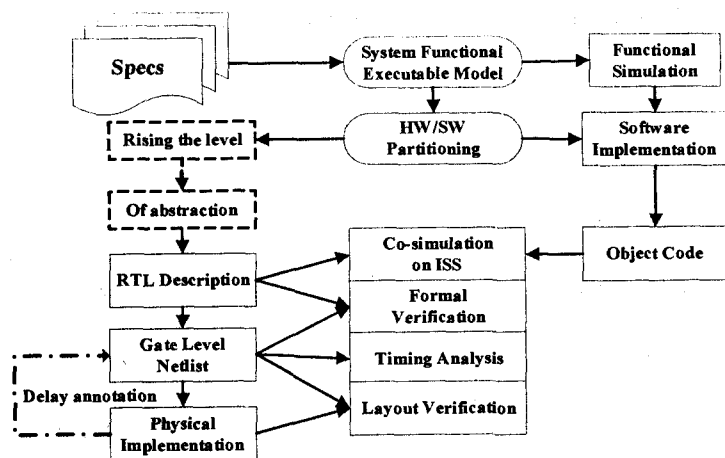


图 1 传统的 SoC 设计流程

图 1 是 STMicroElectronics 公司使用的传统 SoC 设计流程^[2]。对于 SoC 的设计,一般需要经过以下流程:先根据特定的应用需求,定义系统需求规约(System requirements specification),以某一种可执行的算法语言来建立系统的可执行功能模型(System Functional Executable Model),然后进行系统的性能评价,对系统划分为软件和硬件两大子系统,分别进行软件、硬件的实现。软件则进行代码编写、编译到硬件的目标处理器上可以运行的目标代码。硬件则使用硬件供述语言(Hardware Description Language, HDL)在 RTL 级实现描述后与软件的编译目标代码进行指令集(ISS)级别的协同仿真。协同仿真通过后,硬件可以继续门级综合、物理的实现。当硬件的 RTL 描述、门级综合、布局布线这几个步骤,还需要一些形式化的验证(Formal Verification)、时序分析(Timing Analysis)、布局验证(Layout Verification)等工作来保证 RTL 级的描述到布局布线的物理实现没有错误发生。

目前,采用 SystemC 语言进行 SoC 的系统级设计。

由 OSCI(Open SystemC Initiative)组织联合众多 EDA 厂商提出的 SystemC 语言,在 C/C++ 语言基础上,扩展了硬件的描述能力后,使 SystemC 语言具有了对嵌入式计算机进行统一系统描述的能力。使用 SystemC 语言进行系统级的需求规约建模,进行编译执行验证通过了系统的需求规约后,就可以使用 SystemC 分别进行系统的硬件实现、软件实现。统一的系统描述语言 SystemC,解决了系统需求描述、硬件的功能实现、软件代码的功能编写都采用同一种语言来进行描述建模的问题。同时,促进了系统需求分析人员、硬件设计人员、软件开发人员之间的无障碍交流,改进了 SoC 系统的设计过程。

2 UML 在 SoC 开发中的扩展

统一建模语言(Unified Model Language, UML)是由 OMG(Object Management Group)组织最早为规范面向对象的软件开发过程,而提出的软件模型进行统一建模的描述语言。在 2006 年,OMG 组织为了应对 SoC 设计过程中的建模需要,提出了应用于 SoC 设计的 UML Profile for System on a Chip (SoC) v1.0.1 规范(见表 1)。通过一种称为轻量型(Light-weight),定义 UML 的 Profile 作为 UML 扩展集的一种方言(dialect),定义一系列的构造型、标记值、约束定义等,来进行系统的扩展定义。UML 进行扩展后,可以使用用例图、状态图、顺序图、协作图等,SoC 的设计人员就可以通过创建类图和 SoC 结构图(SoC Structure Diagrams)来分别对 SoC 的系统需求建模、硬件设计、软件设计建模。

表 1 对应于 SoC 的 UML Profile 元类

SoC Model element	Stereotype	UML metaclass
Module	SoCModule	Class
Process	SoCProcess	Operation
Data	Data	Class
Controller	Controller	Class
Protocol Interface	SoCInterface	Interface
Channel	SoCChannel	Class
Protocol	SoCProtocol	Collaboration
Port	SoCPort	Port/Class
Module Part	SoCModuleProperty	Property
Channel Part	SoCChannel Property	Property
Connector	SoCConnector	Connector
Clock Port	SoCClock	Port
Reset Port	SoCReset	Port
Clock Channel	SoCClockChannel	Property
Reset Channel	SoCResetChannel	Property
Data Type	SoCDataType	Dependency

UML针对SoC扩展的UML构造型元类,显示在表1中^[3]。更为详细的UML Profile System on a Chip扩展,请参考UML Profile for System on a Chip V1.0.1规范。

3 结合UML的SoC设计流程改进

通过UML的Profile for SoC设计,可以在UML中对SoC的系统元素进行可视化的建模。采用类似于SystemC这样的具有系统级功能描述特性的建模语言,就可以在图1中的传统SoC设计流程中,用SystemC语言来建立系统的可执行的功能模型,并在软/硬件划分之后使用UML的SystemC Profile来进行硬件子系统的无时序功能/有时序功能建模、事务级、行为级、总线周期精确级(BCA)、RTL级的多个层次建模。同时,对于软件子系统的程序实现,也可以通过UML针对相关软件开发语言的Profile来进行软件的开发与实现。

结合UML的SoC设计流程如图2所示。在这样的一个SoC设计流程中,用SystemC语言进行系统级的可执行功能模型描述,并进行功能仿真,以验证系统的概念正确性。软/硬件划分之后,对于SoC系统的硬件模块,将可以通过UML对硬件子系统的无时序/有时序的功能模块、事务级、行为级、总线周期精确级(BCA)以及RTL级进行可视化的图形方式建模,而不是直接以相对低层次的SystemC程序代码的方式进行建模。以图形方式建立的硬件各层次模型,可以通过UML profile for SoC,来通过UML建模工具(如Rational Rose)转换成SystemC语言的代码模型。结合UML的SoC设计流程,将可以把UML使用在SoC设计的三个阶段:建立SoC可执行规约时的需求用例分析、将硬件子系统从功能模型到RTL实现的模型建

立、软件子系统实现时的静态结构模型建立这三个不同的阶段。

在硬件和软件划分之后的分别实现阶段,使用UML对于实现的具体要求,依照需求规约,可以先建立静态结构模型,然后再通过UML Profile for SoC扩展机制,将可视化的设计模型,转换成相应的SystemC语言(或者其它的系统级设计语言,比如SystemVerilog)。这样,就将SoC的设计过程,从系统需求规约直接到低层次代码(a lower level by means of "coding")的设计方式,提升到以可视化的且易于系统设计人员与用户、系统设计不同阶段的设计人员之间交流的模式描述方式。

这种把SystemC代码的描述形式提升到UML的模型描述方式,可以得到多种建模的优势:可视化的建模代替相对低级的代码描述、模型的重用、模型化的系统集成、文档的自动生成与UML文档维护、有利的系统模型分析与建立、模型到SystemC或者SystemVerilog的自动代码生成等。除此之外,这种将UML加入到SoC设计流程中的方式,在系统级功能验证之后的进一步验证,将可以较早地把硬件与软件验证引入到事务级验证。在带有精确定时信息的事务级硬件级别上,将软件与硬件进行较早的集成验证,可以较早地发现和解决硬件与软件的接口不兼容问题。

OMG组织对UML建模能力,还在进行着不断的扩展。目前提出的模型驱动开发(MDA),通过建立系统的与平台无关模型(PIM模型),进行系统的功能概念验证^[4]。在功能概念验证通过后,将系统的模型再转换成与平台相关的PSM模型。通过建立系统模型,然后再逐步实现映射到具体平台的开发方法,在很大程度上提高、促进着系统的开发过程。

4 相关的研究实例

使用UML在嵌入式计算机、SoC设计方法方面的建模研究,已经有相关的研究论文出现。在文献[5]中,作者以UML中的类图和SystemC的结合论述了将UML和SystemC语言使用于嵌入式系统开发方法。在文献[6]中,作者在对物流公司的车辆系统的GPS调度、定位车队过程中,通过建立车辆终端的用例图对系统的功能进行建模,通过车辆终端的顺序图对系统中的动态行为进行建模。在文献[2]中,作者还通过了

(下转第160页)

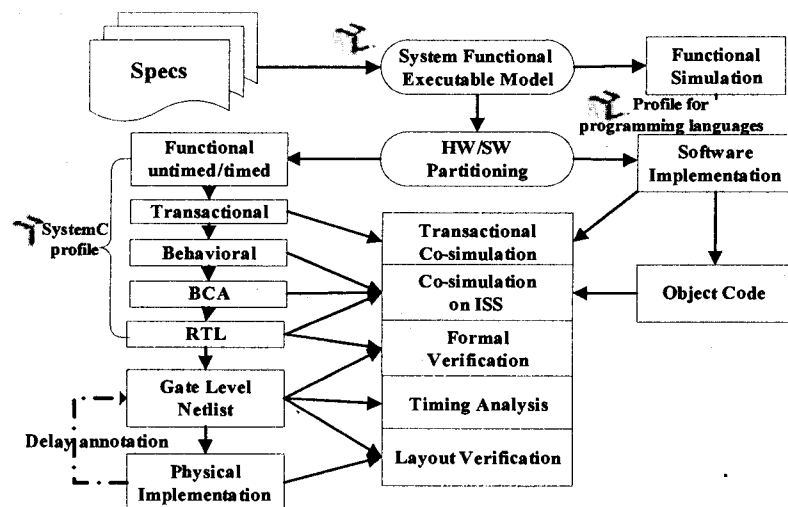


图2 结合UML的SoC设计流程

报现象。在实际应用中,可以根据网络状况相应地设置阈值,如果网络的路由变化相对频繁,则可以将阈值设置稍高;反之,则设置稍低。而在路由不可能变化的情况下,将阈值设为零,可最大程度地提高精度。

将各个阈值下的精度做个平均值(见表 1)。从平均值表上可以看出,当阈值较小时候由于误报较多影响了精度,而阈值高了,精度受漏报增多的影响也不高。当阈值取 3 的时候,漏报与误报的影响达到一个平衡值,取得一个相对较好的精度。在实际应用的时候,可以先对欲保护的主机的日常流量进行一段时间的监视(如 24 小时),然后将其经常访问的若干节点的 TTL 值的变动进行分析,用上述方法进行各个阈值下的精度比对,从而确定最佳阈值,并可定期进行调整。

表 1 所取阈值与精度关系

阈值	精度(+0)	精度(+1)	精度(+2)	精度(+3)	精度(avg)
0	99.68%	60.32%	60.32%	60.32%	70.16%
0.5	99.68%	60.32%	60.32%	60.32%	70.16%
1	98.72%	98.72%	61.28%	61.28%	80.00%
1.5	97.84%	97.84%	97.84%	62.16%	88.92%
2	97.84%	97.84%	97.84%	62.16%	88.92%
2.5	97.84%	97.84%	97.84%	62.16%	88.92%
3	96.92%	96.92%	96.92%	96.92%	96.92%
3.5	95.88%	95.88%	95.88%	95.88%	95.88%
4	95.88%	95.88%	95.88%	95.88%	95.88%
4.5	95.88%	95.88%	95.88%	95.88%	95.88%
5	95.16%	95.16%	95.16%	95.16%	95.16%

6 优缺点和改进方向

在实际应用过程中,当目标主机是存在于同一个子网中的主机时,可以对这个子网中任意一台主机发

出主动检测数据包,用其返回的 TTL 值作为该子网所有主机的 TTL 期望值,这样可以大幅减少主动检测数据包的发送量,提高检测效率。但是当发送伪造数据包的主机与目标之间的条数若恰好等于源主机与目标机之间的条数时,该方法就会失效。如何克服这一弱点,是今后实际应用研究工作的重点。

今后将结合其他的伪造报文检测手段,以期弥补该方法的不足,使之精度更高,误报和漏报率更低。

参考文献:

- [1] Bellovin S. Security Problems in the TCP/IP Protocol Suite [C]//ACM SIGCOMM Computer Communications Review. New York: ACM Press, 1989: 32-48.
- [2] 母军臣 朱长江. 基于概率 TTL 终值的 IP 欺骗 DDOS 防御策略[J]. 河南大学学报: 自然科学版, 2006(4): 96-99.
- [3] Kurose J F, Ross K W. Computer Networking: A top-down Approach Featuring the Internet[M]. 北京: 机械工业出版社, 2006.
- [4] Doyle J. CCIE #1919. Routing TCP/IP Volume 1[M]. 北京: 人民邮电出版社, 2003.
- [5] Stevens W R. TCP/IP 详解. 卷一. 协议[M]. 范建华等译. 北京: 机械工业出版社, 2004.
- [6] Tanenbaum A S. 计算机网络[M]. 第 4 版. 潘爱民译. 北京: 清华大学出版社, 2004.
- [7] 包怀忠. 马季. IP 网络路由追踪技术研究[J]. 微电子学与计算机, 2004(8): 59-62.
- [8] 佚名. IP 欺骗原理精解和防范手段综述[EB/OL]. 2005-04-03. <http://www.gipsky.com/modules/wfsection/article.php?articleid=5>.

(上接第 147 页)

UML 的静态结构类图,通过 UML 对 FIFO 的建模并转换成 SystemC 语言,来说明 UML 与 SystemC 结合进行 SoC 建模的特性。

5 结论

通过 UML profile for SoC,在传统的设计流程中结合 UML 的可视化建模能力,将不可见的代码对系统进行功能描述建模的能力,提升到了一种以可视化的图形方式对系统的功能进行建模,不仅有利于系统设计人员与用户之间的沟通,而且也方便了系统设计人员之间的交流沟通。UML profile for SoC 的扩展能力,在系统级描述语言(比如 SystemC)进行系统描述的过程中,通过静态结构类图对 SoC 的系统结构进行建模,并方便地将 UML 的图形化描述方式转换成系统级语言描述。在模型驱动开发(MDA)这种新的开发模式下,将更有利于系统的模型建立、系统的文档维护,提

高系统的开发效率。

参考文献:

- [1] IEEE-SA Standards Board. IEEE Standard SystemC Language Reference Manual[EB/OL]. 2005-12-06. <http://www.systemc.org>.
- [2] Riccobene E, Scandurra P, Rosti A, et al. A SoC Design Methodology Involving a UML 2.0 Profile for SystemC[M]. [s.l.]: IEEE, 2005.
- [3] OMG. UML Profile for System on a Chip (SoC)[EB/OL]. 2006-08-01. <http://www.omg.org>.
- [4] 陈燕. 基于 UML 的嵌入式系统系统级设计方法研究[D]. 上海: 复旦大学, 2005.
- [5] 石柯. 基于 UML 和 SystemC 的嵌入式系统集成开发方法的研究[J]. 高技术通讯, 2003(11): 44-47.
- [6] 王建新, 姚放吾. 基于 UML 的软硬件协同设计方法[J]. 计算机技术与发展, 2006, 16(1): 96-98.