

软硬件协同设计环境中性能分析工具的研究与实现^①

Study and Implementation of the Performance Analysis Tool in the HW/SW Co - design Environment

程 焯 (上海 华东师范大学计算机系 200062)
栾 静 (上海 华东师范大学计算机系 200062)
(乌鲁木齐 新疆师范大学数理信息学院 830054)
顾君忠 (上海 华东师范大学计算机系 200062)

摘要:软硬件协同设计中的性能分析常常使设计者将过多的精力用于对大量数据的整理和分析,从而降低了设计效率。集成的软硬件协同设计环境 iCDMdt 中的性能分析工具以验证平台采集的分析数据和用户设置为输入,在同一环境下以内嵌的分析图形输出,用简捷明了的方式帮助设计者实现性能分析。本文对 iCDMdt 环境中性能分析工具及其实现进行了详细叙述,并以普通电话机模型作为集成的软硬件协同设计环境 iCDMdt 的设计实例应用其性能分析工具。

关键词:软硬件协同设计 性能分析工具 可视化

1 引言

iCDMdt 环境中的性能分析工具采用通过建立虚拟原型系统的仿真方法来进行系统性能分析,利用仿真运行结果作为性能分析工具的输入,而以可视化分析图形作为输出,为用户提供直观明了的分析结果。本文介绍了 iCDMdt 中的性能分析工具及其实现,同时以普通电话机模型为设计实例,利用 iCDMdt 平台进行设计验证,并通过 iCDMdt 中的性能分析工具得到其性能的分析结果,为进一步优化做好准备。

2 iCDMdt 环境中性能分析工具

在软硬件协同设计系统层的设计目标可简单地表述为:在达到应用性能要求和实时性约束的前提下尽量降低系统成本。性能分析就是估算系统实现的性能,判断是否达到设计约束的设计步骤。系统性能分析通常使用基于调度的静态分析方法或者建立虚拟原型系统的仿真方法。在基于调度的静态分析评价方法中,系统功能被建模成为同步数据流图或者控制数据流图,然后采用数学分析的方法,根据静态调度方法得

到系统的性能评价;基于仿真的评价办法中,设计者构造一个系统的虚拟原型,然后加入一些典型的激励或者测试代码,利用仿真运行结果分析系统性能。虽然基于仿真的方法常常会因为涉及了系统实现的太多细节,仿真的速度非常慢,但是在设计的系统级进行设计空间探索、体系结构调整等宏观工作,不需要涉及到很多的实现细节,因此在较高的层次进行仿真可得到一个大致性能估计,同时也可以达到很高的仿真速度,从而允许实现多次仿真迭代以探索巨大的解空间,并能够满足这一个阶段系统设计对仿真精度的要求^[5]。因此在 iCDMdt 中选择基于仿真的性能分析方法。

2.1 iCDMdt 环境简介

软硬件协同设计技术是一个综合复杂的技术,它不仅是为设计产品提供虚拟的样本,完成软硬件的协同设计和划分、实现不同抽象模型之间的转换、验证和优化,合理配置软硬件的端口,同时还应建立组件库和数据库,提炼基本组件,提高设计质量和提早上市。为了实现上述设计原则,我们在研究软硬件协同设计过程中研

^① 本文受上海市科技发展基金项目“中德软硬件协同设计技术合作研究”资助,项目编号:025007014

制并开发了面向全过程、需求驱动的综合软硬件协同设计与测试环境 iCDMdt,从而使在该平台上设计的模型在制作芯片之前,能够很好的达到功能要求、满足时间约束条件。iCDMdt 具有如图 1 所示的体系结构。

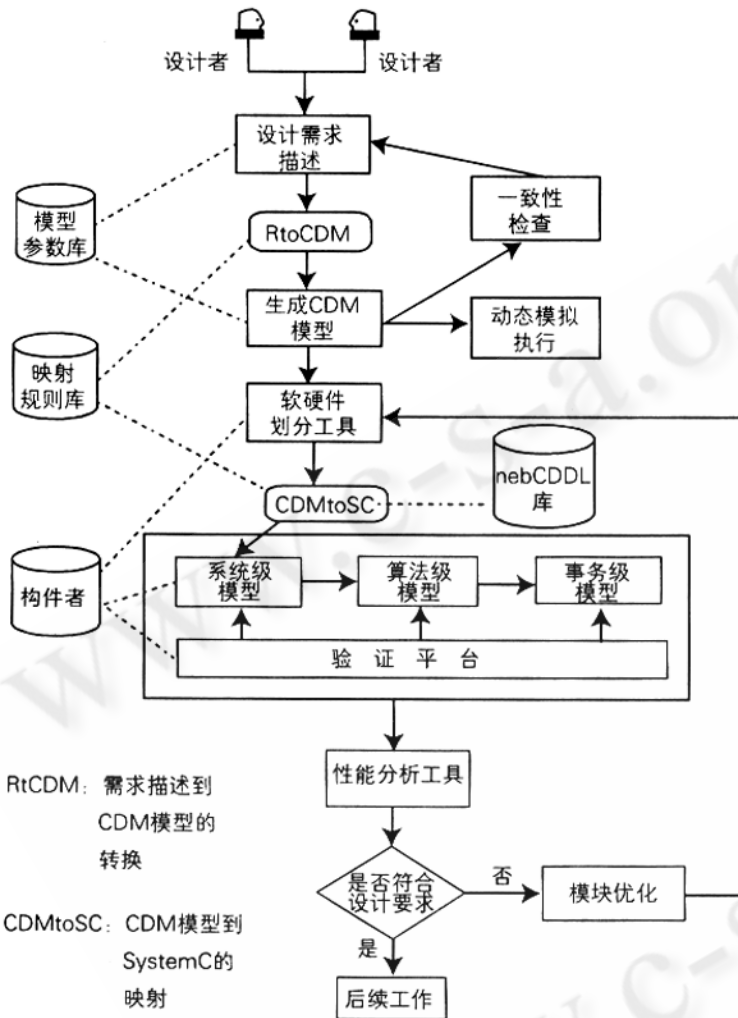


图 1 系统体系结构

验证是软硬件协同设计过程中的重要环节,可以在开发早期找出错误,从而避免时间、金钱的耗费及重复性的劳动,iCDMdt 将设计平台与验证平台无缝地结合,形成一个完整的平台,弥补了传统设计过程中设计与验证采用不同平台的不足。iCDMdt 环境中的验证平台采用非形式化验证,即利用仿真进程模拟激励,对设计模型进行系统级、算法级和事务级操作验证。iCDMdt 是利用 SystemC^[6]这种基于 C++ 的硬件建模语言来模拟系统的行为并产生相应的模拟结果。为实现设计与验证在同一环境中的无缝连接,iCDMdt 利用两

层描述模型之间的转换机制(RtoCDM 和 CDMtoSC)将用户需求通过中间模型转换成 SystemC 模型进行模拟验证(如图 1 所示)。除了验证设计模型的功能正确性,验证其是否满足条件约束也很重要,因此在 SystemC 模型中,用户可设置需要采集的性能数据,然后系统模型转换工具将 SystemC 模型编译成 SystemC 可执行程序。这些程序执行后用户设置的性能数据将以一定的格式输出到文档中,作为性能分析工具的仿真数据输入。性能分析工具还将结合用户对分析图形的设置,经过数据分析处理和图形的集成得到内嵌的分析图形,使用户能直观清晰的得到性能分析结果,便于快速找出设计中的错误和不符合条件约束之处。性能分析工具在整个设计验证过程中承接了设计与进一步优化的过程,成为保证优化并达到设计约束要求的关键,它具有交互性、分析结果可视化和分析图形嵌入等特点。

性能分析工具模型

由于在验证平台的仿真过程中,结合了用户对性能数据的设置,因此 iCDMdt 环境中的性能分析工具必须可以处理不规则的性能数据并考虑用户对性能分析的不同需要,最终用一目了然的分析图形,为用户快速做出分析结论和找出设计弊端提供有效参考。为达到上述目的,iCDMdt 环境中的性能分析工具采用如图 2 所示的方法,以用户设置和仿真数据作为输入,经过性能分析工具的各项数据处理和图形嵌入,最终以分析图形输出,用最简捷的方式帮助设计者完成性能分析过程。

2.2 性能分析工具模型

数据规格化处理模块将用户以多种形式设置的参数和性能指标以及杂乱的仿真数据转化成标准化性能数据,即系统可以识别的数据形式。数据处理模块利用 MATLAB 工具强大的数值计算功能,实现数据的预处理,分别提取数据单位和纯数据,通过计算和筛选,将有效数据存于相应的向量和矩阵中。可视化处理则利用了 MATLAB 的绘图功能,将数据以二维、三维或四维的图形表现,结果清晰明了。在获得可视化图形之后,嵌入平台模块利用了 MATLAB 的外部接口,将 MATLAB 图形集成到用 Visual C++ 开发的 iCDMdt 环境中。

数据规格化处理模块将用户以多种形式设置的参数和性能指标以及杂乱的仿真数据转化成标准化性能数据,即系统可以识别的数据形式。数据处理模块利用 MATLAB 工具强大的数值计算功能,实现数据的预处理,分别提取数据单位和纯数据,通过计算和筛选,将有效数据存于相应的向量和矩阵中。可视化处理则利用了 MATLAB 的绘图功能,将数据以二维、三维或四维的图形表现,结果清晰明了。在获得可视化图形之后,嵌入平台模块利用了 MATLAB 的外部接口,将 MATLAB 图形集成到用 Visual C++ 开发的 iCDMdt 环境中。

3 iCDMdt 中性能分析工具的实现

在进行性能分析与优化时,设计者常常不得不被冗烦的数据分析绊住手脚,而 iCDMdt 环境中的性能分析工具则充分利用了 MATLAB 的数值计算和可视化功能,将性能与约束条件分析的结果以直观的图形方式集成于平台之中,为设计者省去了大量性能数据分析、比较的时间,从而提高了设计效率,缩短了开发时间。

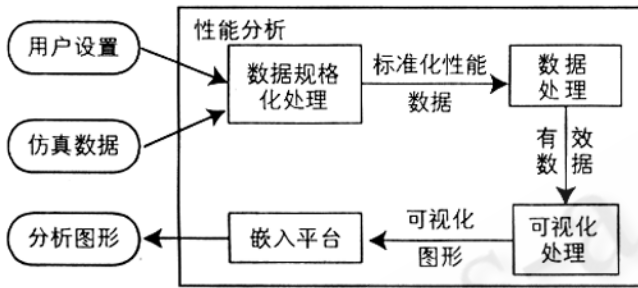


图 2 性能分析工具模型

3.1 数据预处理和可视化实现

因为大量而频繁地读文件和多层循环会导致 MATLAB 程序运行时间过长,影响执行效率,因而将实现数据规格化交给 iCDMdt 环境中的 standard 函数实现,用以删除其中不必要的空格和空行,统一单位并将各种形式的数据转换成统一形式。最后将经过规格化处理的数据以文件形式输出并作为数据预处理模块的输入。

在 MATLAB 中,向量与矩阵的构造、运算极为方便,不必像其他高级计算机语言那样要编写冗长的循环语句。数据预处理模块利用 MATLAB 这一特性,分别提取了单位和纯数据,通过计算和筛选,将有效数据存于相应的向量和矩阵中。

在分析大量数据时,很难直接从中找到内在规律及其含义,依据数据画出图形有时加上颜色可使人们便于找到问题的本质,发展的趋势及错误的的数据。MATLAB 的另一强大功能就在于它可将计算的数据及科学问题形象的显示出来,它可以有二维、三维甚至四维的图形表现,再通过对图形的线型、色彩、视角等属性的处理把数据可视结果表现的尽善尽美。在应用 MATLAB 对大量有效数据进行可视化处理时主要有三方面的工作:1)对离散的数据进行函数的插值。选用一种插值方法来平滑已知数据(插值的方法有多种:最近点插值法、线性插值法、样条插值法和立方插值法),这里采用最近点插值法,对数据进行细化;2)采用不同

图形表现形式,对所处理的数据进行图形控制,例如使用二维曲线和条形图;3)为方便用户理解,根据用户的设置为图形配有标题、坐标轴标识以及注释说明。

3.2 MATLAB 图形的嵌入

MATLAB 的外部接口可以完成与其他编程环境的交互,各取所长,充分发挥 MATLAB 的数值计算的长处,同时能利用其他语言的优势。iCDMdt 环境的设计与测试功能的完全集成充分利用了 MATLAB 这一优势,将 .m 文件的函数实现的性能分析图形集成到用 Visual C++ 开发的 iCDMdt 环境中。虽然 MATLAB 有很多与 C++ 语言混合编程的方法,但为实现 MATLAB 文件完全脱离 MATLAB 环境运行,在 iCDMdt 中我们利用 MATLAB 的 C++ 数学库和图形库来实现它。实现数据分析的 MATLAB 文件经过编译后可形成同名的 C++ 文件,其中含有同名函数。在图形嵌入模块中调用这一函数(本文称为函数体)即可,当然还需要在函数体执行前启动 C++ 数学库和图形库,在函数体执行后关闭它们。

4 应用:设计实例及其分析结果

本文以普通电话机模型作为应用 iCDMdt 环境的设计实例,以此来实现打电话、接电话的功能。为了比较准确的模拟人们日常使用普通电话机的过程,我们拟定了电话机模型的时间约束和信号约束条件,因为我们设计的 iCDMdt 环境是在高层对设计对象进行描述,因此主要考虑对时间的响应。因此通过 iCDMdt 环境中的模拟验证平台能模拟电话机模型在满足时间约束时,实现顺利完成打电话和接电话的过程或者满足某一信号约束条件中断。本例中的系统约束条件为:从摘机到接通电话的最大延迟 15s,从摘机到无人接听的最大延迟 8s,响铃接通电话的最大延迟为 8s,其它情况中断。为了验证模型设计的合理性,设计者可以按照自己的需要调整不同的设计参数,利用模拟验证平台采集各种约束条件下的执行相关时间。在 iCDMdt 环境下经过需求设计、模拟验证和性能分析后,得到的性能分析结果如图 3 所示。

由上图中的子图 1 可以看到设计模型在完成顺利打电话的过程中,多次采集的响应时间数据分布走势相同,并且未超出最大延迟 15s,说明设计模型的打电话相关模块设计较为合理,而由子图 2 可以看到不同

约束条件下的响应时间数据分布有一定差异,响应时间在 10s 内的执行次数比曲线很平缓,表示有较多次仿真执行都以中断结束,说明设计模型的接电话相关模块的设计或约束设置上不太合理。

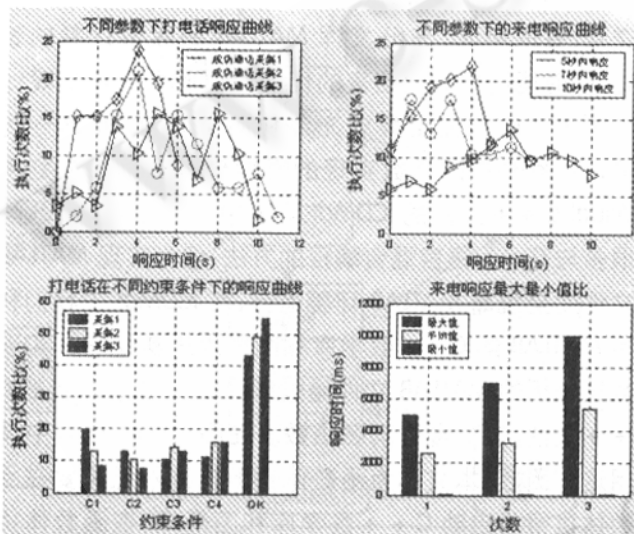


图 3 iCDMdt 环境中普通电话机实例的设置及性能分析图

5 小结

在我们研制开发的 iCDMdt 环境中,其性能分析工

具成为软硬件划分和进一步优化的关键。本文在简单介绍了继承的软硬件协同环境 iCDMdt 后,详细叙述了性能分析工具及其实现,并以普通电话机为设计实例,应用 iCDMdt 环境得到直观明了的性能分析图形。这一实例的实际执行证明了设计者不仅能在同一环境中完成设计、模拟、测试、分析全过程,而且能得到清晰直观的分析结果,从而大大提高了设计效率。

参考文献

- 1 何强、何英, Matlab 扩展编程(M), 北京 清华大学出版社, 2002。
- 2 尚涛、石端伟、安宁、张李义, 工程计算可视化与 MATLAB 实现(M), 武汉大学出版社, 2002。
- 3 刘志俭、潘献飞、连军想, MATLAB 外部程序接口 (6.X)(M), 北京 科学出版社, 2002。
- 4 栾静、吕钊、顾君忠, 中德软硬件协同设计合作研究项目研制报告(R), 华东师范大学应用研究所, 2004, 12。
- 5 徐明、夏新军、陈吉华, SoC 设计中一种软硬件划分的性能评价方法, 计算机工程, 2004, 11。
- 6 SystemC. Functional Specification For SystemC2. 0. [EB/OL] <http://www.systemc.org>, 2001.