第35卷 第23期 计 算 机 工 程 2009年12月
Vol.35 No.23 Computer Engineering December 2009
·开发研究与设计技术· 文章编号：1000—3428(2009)23—0284—02 文献标识码：A 中图分类号：TP311

面向多目标的指令集模拟技术

杨义彬，蒋烈辉，尹 青，何红旗，宋彭涛
(解放军信息工程大学信息工程学院，郑州 450002)

摘 要：针对传统指令集模拟器的目标依赖性，提出基于虚指令集的多目标指令集模拟技术。采用解释型模拟策略，利用虚指令集技术优化虚指令集构造，进一步设计出面向多目标的指令集模拟器。实验结果证明该技术具有快速、灵活、可重用等特性，该模拟器在后续工作和兼容性方面优于传统模拟器。

关键词：虚指令集；模拟策略；多目标

Instruction Set Simulation Technique for Multi-target

YANG Yi-bin, JIANG Lie-hui, YIN Qing, HE Hong-qi, SONG Peng-tao
(College of Information Engineering, PLA Information Engineering University, Zhengzhou 450002)

【Abstract】Aiming at target dependence of traditional instruction set simulator, this paper proposes multi-target instruction set simulation technique based on virtual instruction set. It designs an instruction set simulator for multi-target further by using interpreted simulation strategy and imposing virtual instruction set technique to optimize virtual instruction set construction. Experimental result proves that the technique is fast, flexible and reusable, the simulator is superior to traditional simulator on the sides of follow-up work and compatibility.

【Key words】virtual instruction set; simulation strategy; multi-target

1 概述

指令集模拟器(Instruction Set Simulator, ISS)是在本地计算机环境下模拟其他处理器上程序运行过程的工具。前者被称作宿主机，后者被称作目标机，目标机通常是不存在或者是当时不可用的，它通过模拟每条指令在目标处理器上的执行效果来模拟目标机程序，是目标处理器的软件模拟器[1]。

指令集模拟器是处理器和嵌入式系统设计中的重要工具，是进行系统体系结构软件设计评估、系统软件设计开发及软硬件协同开发的有利工具。一个传统的指令集模拟器只能针对一个或几个同型号的处理器，而不同的处理器需要不同的模拟器，这是制约其发展的重要因素。本文通过设计一种通用的虚拟指令集，摆脱对目标指令集的依赖，实现针对多目标可扩展的指令集模拟[2]。

2 模拟策略

在经典的Van Neumann结构的计算机中，程序执行过程可分为4步：取指令，分析指令，执行和获得下一条指令地址。在使用软件模拟时，核心过程是分析指令和执行指令，将要执行的目标指令集代码译码转换为本机支持的指令序列，在本地执行代码。

指令集模拟器根据实现方式的不同，分为解释型模拟器和编译型模拟器[3]。

解释型模拟方法是在宿主机内存中建立一个数据结构来表示目标机的状态，宿主机根据目标机的指令对该数据结构进行相应操作，代表目标机改变状态，从而模拟目标机的具体实现过程。其主体是一个取指、译码、执行的循环，即从内存中读出一个指令字，分析指令并生成指令的操作码，通过调度和根据指令语义来运算并更新处理器状态。其流程如图1所示。

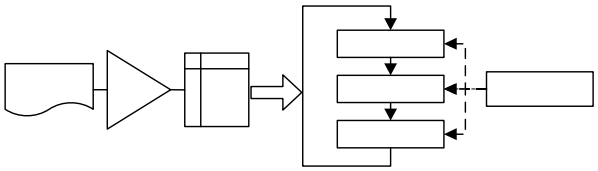


图1 基于解释型的模拟流程

基于编译型模拟的方法类似于交叉编译器，先将目标处理器指令翻译为一组宿主机指令，再通过执行宿主机指令来实现模拟，被模拟的机器状态保存在宿主机的全局存储空间中。其流程如图2所示。

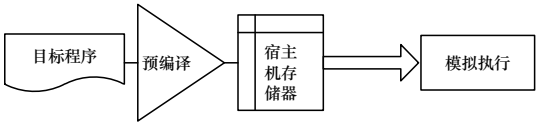


图2 基于编译型的模拟流程

编译型模拟器由于预先进行了一系列预处理，模拟时直接运行宿主机指令，其性能较高，但编译型模拟器一般仅能模拟静态代码，对于在运行时动态产生或修改代码的应用程序，如操作系统等无法正常模拟。解释型模拟器由于采用“解释”的形式逐条模拟指令，性能一般较低，但解释型模拟器仿真灵活性较高，便于调试跟踪，易于实现，因此，大多数指令集模拟器都采用解释型模拟策略。考虑上述优点，本文

基金项目：国家“863”计划基金资助项目(2006AA01Z409)
作者简介：杨义彬(1982－)，男，硕士研究生，主研方向：计算机仿真模拟；蒋烈辉，教授；尹 青，副教授；何红旗，讲师；宋彭涛，硕士研究生
收稿日期：2009-05-11 **E-mail：**yangyibin369@163.com

采用解释型模拟策略。

3 虚指令集构造的关键技术

虚拟指令集(Virtual Inalrucllon Set, VIS)是一种抽象的指令集系统，其设计思想来源于 Java 虚拟机。它是对各种运行于实际硬件之上的指令集系统的高度概括，将各种指令集翻译为统一的虚拟指令。在设计多目标模拟器时，虚拟指令充当目标代码和模拟器中间的桥梁，为模拟器屏蔽各种指令集的复杂性，作为后端模拟器只须考虑如何模拟虚拟指令的执行，简化了模拟器的设计，提高了系统的可扩展性。

根据虚拟指令的定义，虚拟指令集应该包含所有指令集的功能，但并不表示虚拟指令集就是所有指令集的超集。一般，指令集中包括一部分最基本的指令来实现底层的基本功能，其他的指令通过对基本指令的嵌套调用来实现复杂操作。

3.1 虚拟指令集构造原则

在设计构造虚拟指令集时，要遵循以下原则[4]:

(1)支持目标支持的数据结构和结构特征。虚拟指令集只有支持目标需要的数据结构和结构特征，才能正确有效地表示和模拟相应状态的操作和变化。

(2)足够的复杂度。虚拟指令集应具有一定的复杂度，这样才能直接表示大部分指令，从而实现应有的功能操作。

(3)最优化的指令集子集。虚拟指令集要能直接实现大部分常用指令，要避免虚拟指令的反复嵌套，而影响模拟器执行的效率。虚拟指令的条数不宜太多，要减少指令冗余，以利于简单地描述语义，降低模拟难度。

3.2 虚拟指令集设计

根据指令集的特性，在构建虚拟指令集时可以根据指令的不同特点分别对待。采用分层的设计思想，利用基本指令与自定义指令相结合的策略来构建虚拟指令集，将整个指令集划分为若干层次，高层次指令由低层次指令间接实现[5]。

在设计虚拟指令集的架构时，将指令分为基本指令 L1 和自定义指令 L2 2 层。基本指令属于底层指令，包含原子操作 L1.1 与常用指令 L1.2，能实现常见指令的基本功能。自定义指令通常是一些复杂指令，需要进行多种复杂运算。其中，原子操作是最底层的操作，包括基本运算指令、逻辑指令、比特操作指令和程序控制等指令，它是整个指令集的基础。常用指令相比原子指令复杂但使用率高，对于此类通用且复杂度较低的常用指令，可以由数条原子操作指令的组合来实现。为保证整个系统的效率，一些复杂度较高或者处理器特有的指令，不宜用大量基本操作实现。针对此类特殊指令，可以自定义一条指令并通过单独的模拟来实现其功能操作。指令集分层示意图如图 3 所示。

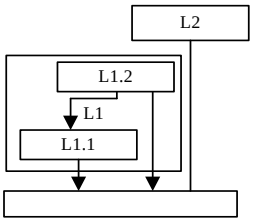


图 3 指令集分层示意图

确定虚拟指令集的最小基本指令类可以参考 RISC 指令集，遵循以下步骤[6]:

(1)确定最小运算类。各种处理器都有运算指令，包括算术运算和逻辑运算，算术运算又可分为无符号数运算、有符号数运算、浮点运算和整型运算等。在虚拟指令中要包含一些最基本的运算指令，如加法指令、减法指令等，一些复杂的运算指令就可以由此类最基本的运算指令组合得到。

(2)确定最小寻址方式。各种处理器的寻址方式不同，例如，8086 有 10 种寻址方式[3]，而 MC68000 有多达 18 种寻址方式，如果将如此多的寻址方式都包含到虚拟指令集中，那么会增加后端模拟器识别操作数寻址方式的难度。根据汇编指令中操作数的存放方式，操作数只能存放在寄存器、存储器和指令中，虚拟指令确定了 4 种基本寻址方式：寄存器寻址，立即数寻址，存储器寻址和位寻址。其他各种寻址方式可以通过运算转化为上述 4 种基本寻址方式。

(3)确定最小控制类。控制类指令是以改变程序计数器 PC 中的内容为目标，控制程序执行流程。虚拟指令中的基本控制类指令有无条件转移指令、条件转移指令、子程序调用和返回指令，其他各种控制类指令都可以归到上述 4 类指令中。

4 实验

基于本文提出的基于虚指令集的多目标指令集模拟技术，实现了一个基于虚指令集的模拟器。为了测试该模拟器的性能，以 MCS-51 和 Intel IA32 的指令集为测试指令集，并选用 translation simulator 和 explanation simulator 作为比较，分别测试它们的性能。测试环境是 1 Hz 处理器，1 GB 内存，Windows Server 2003 系统，均为缺省设置。实验结果如图 4 所示。

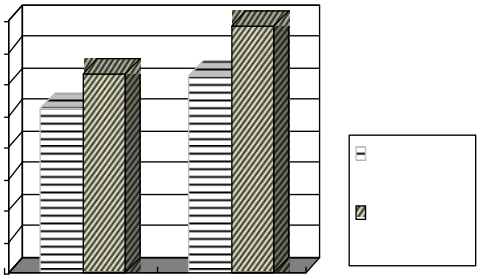


图 4 性能比较

5 结束语

实验数据表明，普通的指令集模拟器相比基于虚拟指令集的模拟器在性能上高出 20%左右，但基于虚指令集的多目标指令集模拟技术能够更好地兼容多指令系统，实现多目标。

参考文献

[1] Rowson J A. Hardware/software Co-simulation[C]//Proceedings of the 31st Design Automation Conference. San Diego, CA, USA: [s. n.], 1994: 439-440.

[2] Zhu Jianwen, Gajski D D. An Ultra-fast Instruction Set Simulator[J]. IEEE Translations on Very Large Scale Integration Systems, 2002, 10(3): 363-373.

[3] Larsson F, Magnusson P, Wemer B. SimGen: Development of Efficient Instruction Set Simulators[D]. Kista, Sweden: Swedish Institute of Computer Science, 1997.

[4] 王晓红, 王 旭, 王 雷, 等. 指令集仿真器自动生成技术的研究[J]. 计算机工程与应用, 2003, 39(2): 129-130.

[5] 乔亚男, 王换招, 方 舟. 指令分层过滤体系在通用CPU模拟器设计中的应用[J]. 计算机工程与应用, 2005, 41(3): 119-121.

[6] 唐遇星. 面向动态二进制翻译的动态优化和微处理器体系结构支撑技术研究[D]. 长沙: 国防科技大学, 2005.

编辑 陆燕菲