

**Адаптация LIT методологии оценки производительности x86 архитектур под
отличающуюся архитектуру**

А.В. Марамзин

Московский физико-технический институт (государственный университет)

АО «Интел А/О»

Основная часть исследований в области процессорных архитектур проводится с помощью программной симуляции. Программная модель обладает большей гибкостью и позволяет провести проверку идей и решений относительно быстро. Однако, программной симуляции присущи свои трудности. Исполнение приложения на программной модели на несколько порядков медленнее, чем исполнение того же приложения на настоящем моделируемом микропроцессоре. Если исполнение стандартного SPEC теста производительности (*англ.* benchmark) на настоящем процессоре составляет порядка 100 секунд [1], то симуляция поведения того же теста на потактовой модели может занять десятки лет. Для решения этой проблемы была предложена методология семплирования. Из полной трассы инструкций приложения выбирается репрезентативный интервал (*англ.* sample). Путём симуляции и замера производительности процессора на семплированном интервале делаются оценки относительно поведения всего теста. Существует огромное количество подходов к семплированию: путём анализа профиля исполнения приложения [2][3], путём периодических или случайных выборок [4], и.т.д.

В данной работе описано применение LIT (*англ.* Long Instruction Trace) методологии, специально разработанной для оценки инкрементальных улучшений x86 архитектур, с целью изучения исследуемой архитектуры. Кратко, LIT трасса представляет собой образ памяти и архитектурное состояние в определенный момент исполнения бенчмарка. Также, для точного повторения пути трассируемой программы при симуляции, предоставляется LIT-сценарий (трасса внешних событий: прерывания, DMA, I/O). Для воссоздания корректного (такого, как если бы трасса исполнялась с самого начала) микроархитектурного состояния к определенному моменту используется технология разогрева кешей посредством трассировки обращений в память.

Моменты прихода внешних событий заданы в LIT-сценарии через количество выполненных x86 команд, что вносит определенную сложность при оценке архитектур с микроархитектурно-оптимизируемым кодом, когда в результате оптимизаций количество выполненных команд может отличаться от оригинального x86 кода. Однако, для поддержания совместимости с x86 архитектурой новые архитектуры должны

поддерживать семантически эквивалентное оригинальному коду исполнение, в частности, трасса обращений в память является инвариантом при преобразовании x86 кода в новый.

Вышеописанная проблема была решена путём перепривязки ЛИТ-событий к номерам доступов в память. В качестве валидации было проведено проигрывание ЛИТ-сценария, заданного количеством обращений в память, на оригинальной x86 трассе. Методология была успешно использована для оценки производительности на стандартных ЛИТ-трассах в двух проектах по исследованию замены x86 микроархитектуры совместимым программно-аппаратным решением.

Литература

1. Данные организации SPEC (<https://www.spec.org/>).
2. Lieven Eeckhout, Koen De Bosschere, Henk Neefs. Synthetic analysis through synthetic trace generation. IEEE, 2000.
3. Pradeep K. Dubey and Ravi Nair. Profile-driven generation of trace samples. IEEE, 1996.
4. Ronald E. Wunderlich, Thomas F. Wenisch, Babak Falsafi, James C. Hoe. Statistical sampling of microarchitecture simulation. ACM transactions on modeling and computer simulation, 2006, Vol.16, No. 3, Pages 197-224.