

Общие свойства симуляторов

Курс «Программное моделирование вычислительных систем»

Григорий Речистов
grigory.rechistov@phystech.edu

16 февраля 2015 г.

- 1 Требования
- 2 Точность
- 3 Скорость
- 4 Совместимость
- 5 Итоги

На прошлой лекции

Симуляторы

- сократили длительность цикла проектирования вычислительных систем
- увеличили эффективность труда проектировщика
- стали незаменимы при совместной разработке аппаратуры и программного обеспечения

Вопросы

- Что такое функциональный симулятор режима приложения?

Вопросы

- Что такое функциональный симулятор режима приложения?
- Почему программы и аппаратуру необходимо разрабатывать совместно?

Вопросы

- Что такое функциональный симулятор режима приложения?
- Почему программы и аппаратуру необходимо разрабатывать совместно?
- Может ли Dosbox загрузить Windows?

Вопросы

- Что такое функциональный симулятор режима приложения?
- Почему программы и аппаратуру необходимо разрабатывать совместно?
- Может ли Dosbox загрузить Windows? Да, если это Win 3.1

Требования к симуляторным решениям

- Точность
- Скорость
- Совместимость

Точность

Essentially, all models are wrong, but some are useful. *G. E. P. Box, N. R. Draper*

- Точность должна быть достаточной для выполнения целей, поставленных перед симулятором
- Излишняя точность — медленная работа, дольше разработка, больше ошибок

Откуда берутся данные для создания модели

- По спецификациям, документации: SDM, EAS, BWG...
- Из предыдущей модели + спецификации
- Из анализа работы реальной аппаратуры (reverse engineering)

Как проверить корректность модели?

- Функциональная корректность
- Адекватность результатов
- Сравнение с другой моделью этого же устройства
- Сравнение с реальной аппаратурой

Как доказать корректность модели?

- Формальная верификация
- Не всегда возможна
- Огромное число состояний: длина одной инструкции IA-32 до 15 байт, а это 2^{120} комбинаций входных значений

Скорость

- MIPS — миллионы гостевых инструкций, исполненных за одну хозяйскую секунду
- МГц — число гостевых тактов, исполненных за одну хозяйскую секунду

Скорость как замедление

Slowdown. Сравнение с физической системой со сходными параметрами и исполняющей аналогичный сценарий

- Необходима реальная система, с которой можно сравниваться
- Специальный случай: целевая архитектура = хозяйская
- Чаще всего модель работает медленнее аппаратуры

Замедление разных типов симуляции

	Тип модели	Замедление
Функциональная с аппаратным ускорением		1...5
Функциональная с двоичной трансляцией		10...100
Функциональная с интерпретацией		100...10 ³
	Потактовая	10 ³ ...10 ⁶

Как повысить скорость симулятора?

- Это мы будем обсуждать на многих последующих занятиях
- Эффективная симуляция и бездействие, аппаратная поддержка, параллельное исполнение...

Совместимость

- Отладчики, среды разработки, анализаторы трасс
- Форматы: образов дисков и памяти, трасс
- Автоматизация и расширяемость с помощью динамических языков: Perl, Python, Lua...
- Автоматическая работа без человеческого вмешательства: без GUI, для регулярного тестирования

Структура симулятора

- Симуляционное ядро
- Интерфейс командной строки
- Графический интерфейс
- Интерпретатор скриптов
- Публичный API

Итоги

- Точность
- Скорость
- Расширяемость

На следующей лекции

- Интерпретация

Спасибо за внимание!

Слайды и материалы курса доступны по адресу
<http://is.gd/ivuboc>

Замечание: все торговые марки и логотипы, использованные в данном материале, являются собственностью их владельцев. Представленная точка зрения отражает личное мнение автора.