

Исполняющие и неисполняющие модели. Симуляция многопроцессорных систем

Курс «Программное моделирование вычислительных систем»

Григорий Речистов
grigory.rechistov@phystech.edu

14 сентября 2014 г.

- 1 Два класса моделей
- 2 Косимуляция
- 3 Практический пример
- 4 Литература
- 5 Конец

На этой лекции

- Совместная работа с моделью процессора.
- Работа с несколькими процессорами сразу.

Какие типы моделей нам известны

- Интерпретация

Какие типы моделей нам известны

- Интерпретация: процессоры.

Какие типы моделей нам известны

- Интерпретация: процессоры.
- Очередь событий

Какие типы моделей нам известны

- Интерпретация: процессоры.
- Очередь событий: таймер.

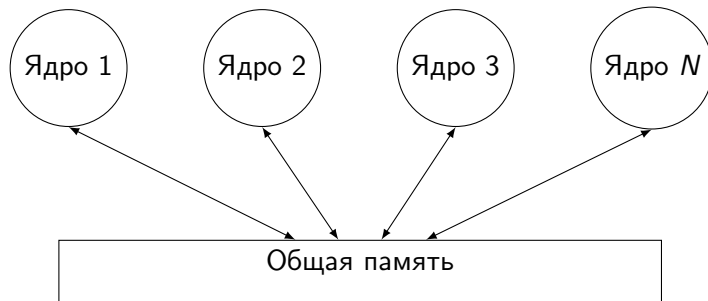
Какие типы моделей нам известны

- Интерпретация: процессоры.
- Очередь событий: таймер.
- «Мгновенная» модель «стимул — отклик»

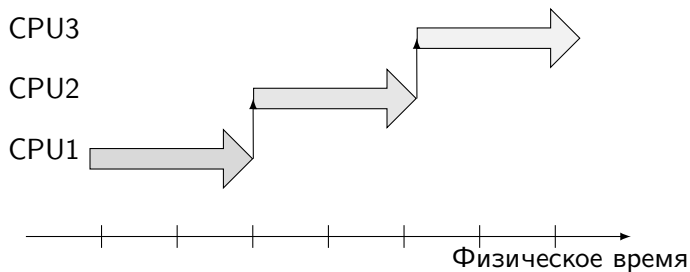
Какие типы моделей нам известны

- Интерпретация: процессоры.
- Очередь событий: таймер.
- «Мгновенная» модель «стимул — отклик»: ОЗУ.

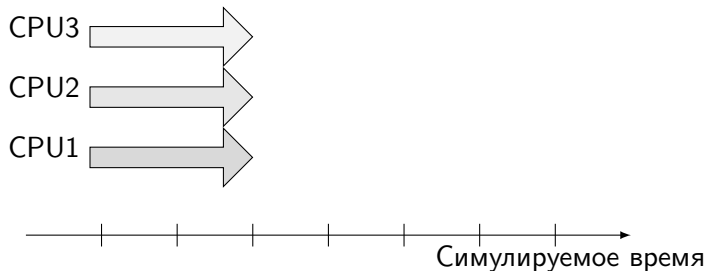
Модель многопроцессорной системы



Квотированная симуляция



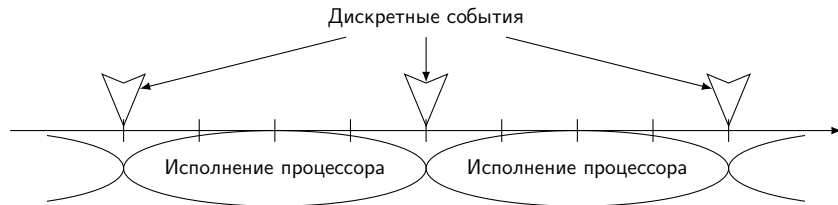
Квотированная симуляция



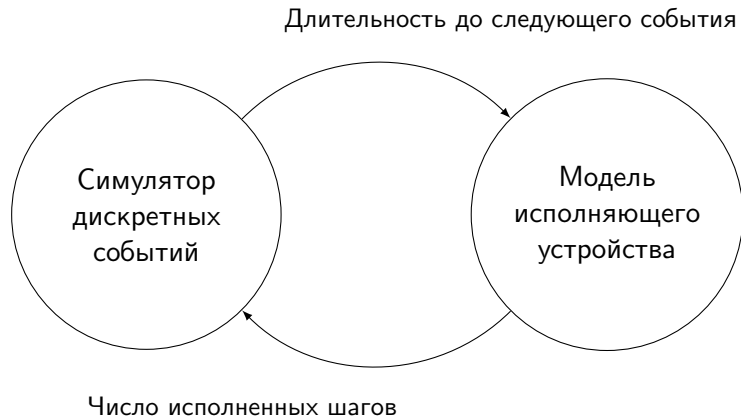
Размер квоты

- Процессор может исполнить меньше инструкций, чем содержится в выданной ему квоте.
- Не следует увлекаться излишне большими квотами, пытаясь ускорить исполнение.
- В DES-модели могут быть реализованы псевдо-события, обработка которых вызывает переключение текущего процессора.

Совместная симуляция DES и исполняющей модели



Косимуляция



Пример на модели viper-busybox.simics

Квота на восьмиядерной конфигурации:

```
simics> cpu-switch-time
Current time quantum:      0.0001 s
  200000.0  viper.mb.cpu0.core[0][0]
  200000.0  viper.mb.cpu0.core[0][1]
  200000.0  viper.mb.cpu0.core[1][0]
  200000.0  viper.mb.cpu0.core[1][1]
  200000.0  viper.mb.cpu0.core[2][0]
  200000.0  viper.mb.cpu0.core[2][1]
  200000.0  viper.mb.cpu0.core[3][0]
  200000.0  viper.mb.cpu0.core[3][1]
Default time quantum not set yet
```


Пример на модели viper-busybox.simics

Симулируемое время на восьмиядерной конфигурации:

```
simics> ptime -all
```

processor	steps	cycles	time [s]
viper.mb.cpu0.core[0][0]	5082940459	6000000000	3.000
viper.mb.cpu0.core[0][1]	5791600000	6000000000	3.000
viper.mb.cpu0.core[1][0]	5791600000	6000000000	3.000
viper.mb.cpu0.core[1][1]	5791600000	6000000000	3.000
viper.mb.cpu0.core[2][0]	5791600000	6000000000	3.000
viper.mb.cpu0.core[2][1]	5791600000	6000000000	3.000
viper.mb.cpu0.core[3][0]	5791600000	6000000000	3.000
viper.mb.cpu0.core[3][1]	5791600000	6000000000	3.000

Литература I



Wind River. Simics Processor Integration Guide.

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. Материалы доступны по лицензии Creative Commons Attribution-ShareAlike (Атрибуция — С сохранением условий) 4.0 весь мир (в т.ч. Россия и др.). Чтобы ознакомиться с экземпляром этой лицензии, посетите

<http://creativecommons.org/licenses/by-sa/4.0/>