



iSCALARE



Лаборатория суперкомпьютерных технологий для биомедицины, фармакологии и малоразмерных структур

Языки разработки моделей и аппаратуры

Григорий Речистов

grigory.rechistov@phystech.edu

На предыдущих лекциях:

- Цели симуляции
- Алгоритмы моделирования
- Проблемы, связанные с симуляцией устройств

На этой лекции:

- Компоненты симулятора
- Языковые средства разработки моделей и аппаратуры

Вопрос

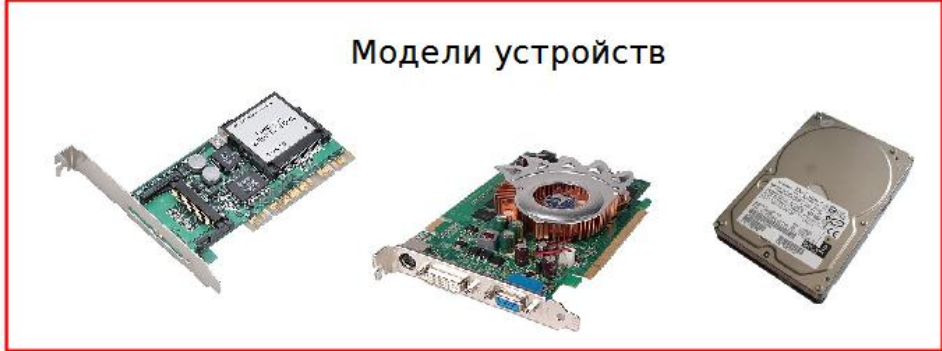
На каком языке
программирования должен быть
написан симулятор?

Классификация компонент

Графический интерфейс



Модели устройств



Интерфейс командной строки



Интерпретатор скриптов



Использование языков общего назначения (C, C++, Java...)

- Используется ООП
- Написание моделей «с нуля»
- Особенности языков
 - Специфика int, struct, endianness
 - thread safety
 - malloc/free

Абстракции аппаратуры (1/2)

- Сигналы — логический уровень (0, 1, X, Z)
- Шины — передача групп бит
- Операции над отдельными битами
- Транзакции — отображение направления сигнала
- Абстракции хранения данных: группы регистров, банки памяти

Абстракции аппаратуры (2/2)

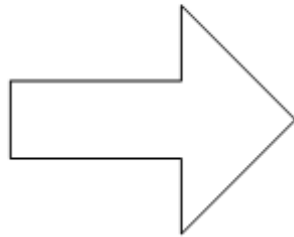
- Карты памяти
- Задержки событий — разные для различных действий

Решения

- Создание библиотек, реализующих общие примитивы моделирования (SystemC/TLM)
- Использование специализированных языков написания моделей (DML)

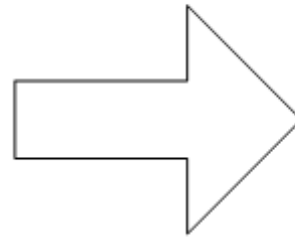
DML (1/2)

*.dml



DMLC

*.c



GCC

*.so



DML (2/2)

```
dml 1.2;
```

```
device simple_dml_device;
```

```
parameter desc = "Simple DML device";
```

```
parameter documentation = "This is an implementation of  
simple DML device.";
```

```
bank regs {
```

```
    parameter register_size = 4;
```

```
    register r1 @ 0x1000;
```

```
    register r2 @ 0x1004;
```

```
}
```

Разработка процессора (1/2)

Исходное описание



`int 13h`

$f(x)$



Дизассемблер

Симулятор

Документация

Примеры:

- LISA
- ISDL
- SimGen

Разработка процессора (2/2)

Недостатки:

- Генерируется не самый быстрый код
- Код может быть не компактен
- Модель может работать медленнее

Преимущества:

- Скорость создания/модификации
- **Согласованность**

SystemC/TLM (1)

Дата	Версия	Примечания
Sept 1999	0.9	First version; Cycled based
Mar 2000	1.0	Widely accessed major release
Aug 2002	2.0	Add channels & events; cleaner syntax
Apr 2002	2.0.1	Bug fixes; widely used
Dec 2005	2.1.v1	IEEE approves the IEEE 1666-2005 standard for SystemC
Jun 2008	2.2.05	TLM-2.0.0 library released
Nov 2011	2.3.0	IEEE approves the IEEE 1666-2011 standard for SystemC

SystemC/TLM (2)

Application

Written by the end user

Methodology- and technology-specific libraries
(SystemC verification library, bus models)

Core language

Modules
Ports
Exports
Processes
Interfaces

Predefined channels

Signal, clock, FIFO,
mutex, semaphore

Channels
Events

Utilities

Report handling,
tracing

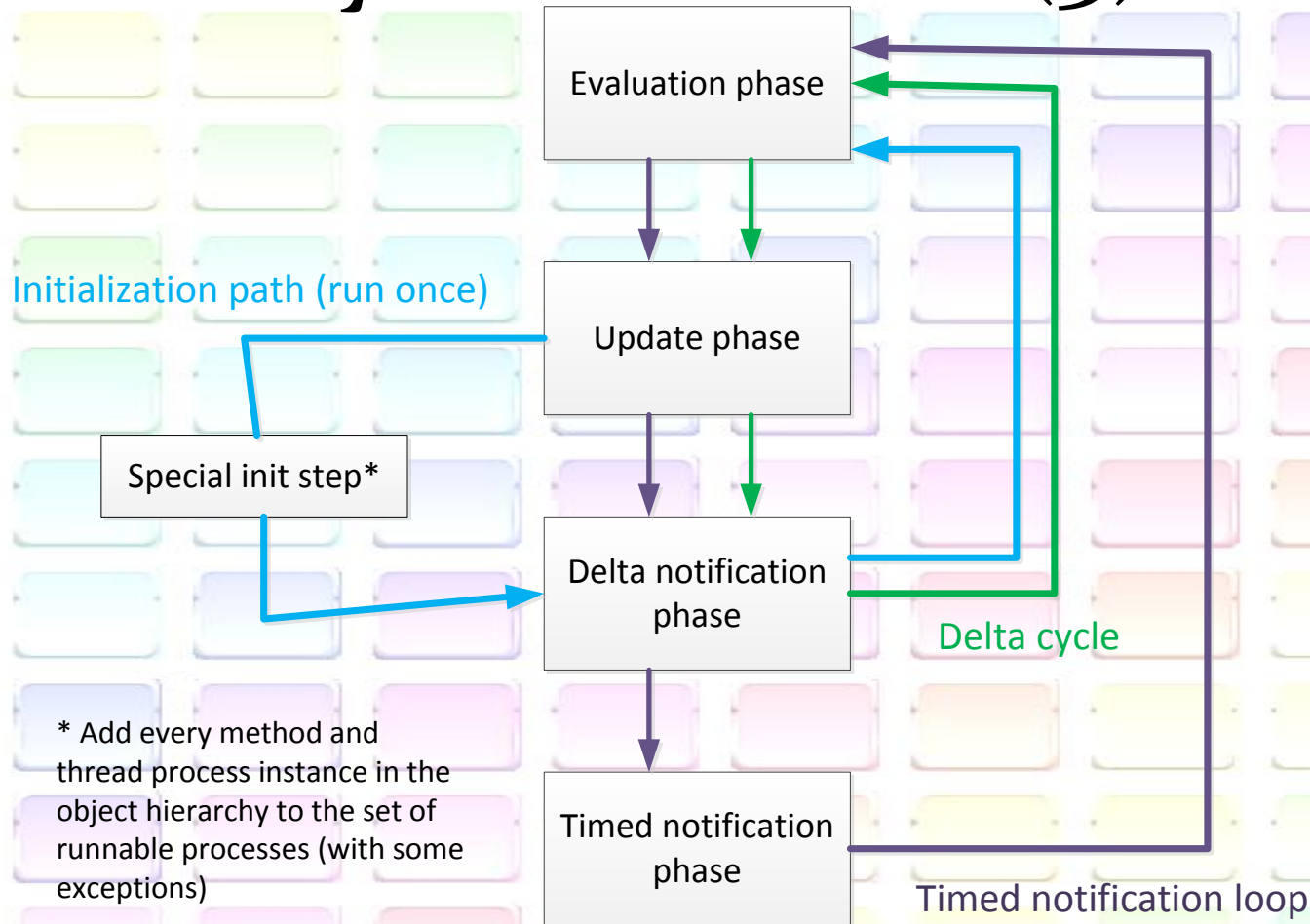
Finite-precision integers
Limited-precision integers
Fixed-point types

Data types

4-valued logic type
4-valued logic
vectors
Bit vectors

Programming language C++

SystemC/TLM (3)



* Add every method and thread process instance in the object hierarchy to the set of runnable processes (with some exceptions)

Verilog

Phil Moorby & Prabhu Goel

«Automated Integrated Design Systems», 1984 г.

Netlist — логически эквивалентное описание, состоящее из элементарных логических примитивов

Команды:

- Синтезируемые — представленные в аппаратуре
- Несинтезируемые — для отладки и симуляции

В 2009 стандарт объединен с Verilog

VHDL

- Был разработан в 1983 г. по заказу Министерства обороны США
- Первоначально предназначался для моделирования, но позже появилась и синтезируемое подмножество
- Стандарт многократно обновлялся

Литература

- O. Schliebusch et al. **Architecture implementation using the machine description language LISA** — 2002 — <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=994928>.
- Hadjiyiannis G., Hanono S., Devadas S. **ISDL: An Instruction Set Description Language for Reatargetability**. — 1997 — <http://www.caa.lcs.mit.edu/~devadas/pubs/isdl.ps>
- Rishiyur S Nikhil et al. **BSV by Example**. http://csg.csail.mit.edu/6.S078/6_S078_2012_www/resources/bsv_by_example.pdf
- Fredrik Larsson et al. **SimGen: Development of Efficient Instruction Set Simulators**. <ftp://ftp.sics.se/pub/SICS-reports/Reports/SICS-R--97-03--SE.ps.Z>

На следующей лекции:

Контрольная работа

Спасибо за внимание!

Все материалы курса выкладываются на сайте лаборатории:
http://iscalare.mipt.ru/material/course_materials/

Замечание: все торговые марки и логотипы, использованные в данном материале, являются собственностью их владельцев. Представленная здесь точка зрения отражает личное мнение автора, не выступающего от лица какой-либо организации.