

Языки описания схем

mk.cs.msu.ru → Лекционные курсы → Языки описания схем

Блок 20

Verilog:

Как на самом деле устроена симуляция

Лектор:

Подымов Владислав Васильевич

E-mail:

valdus@yandex.ru

ВМК МГУ, 2024/2025, осенний семестр

Напоминания

Программная модель кода \mathcal{V} выполняется пошагово (*как обычная программа*)

На каждом шаге симуляции определены, *в числе прочего*:

- ▶ Значения всех точек
- ▶ Текущее значение τ **модельного времени** и соответствующий **текущий регион**: значение τ вместе с совокупностью **действий**, которые следует выполнить в текущее время

В начале симуляции:

- ▶ $\tau = 0$
- ▶ Значение каждого разряда каждой переменной — \mathcal{X}
- ▶ Значение каждого разряда каждого соединения — \mathcal{Z}

Процессы, события, действия (общие слова)

Кодом в начале симуляции задаётся набор **процессов** — объектов, пошаговое параллельное (*конкурентное, асинхронное*) выполнение которых и образует симуляцию

Симуляция представляет собой поток **событий**, описывающих изменение значений времени и точек и состояний процессов

Событие представляет собой выполнение **действия** в заданный момент модельного времени

Во время симуляции могут **планироваться** новые действия: создаваться с указанием **региона** (**времени выполнения** действия)

Процессы, события, действия (общие слова)

Среди действий особо выделяются два вида:

- ▶ **Присваивание** значения `val` в переменную `x`: $x \leftarrow val$
 - ▶ Основной эффект выполнения: значение в точке `x` становится равным `val`
- ▶ **Вычисление** процесса
 - ▶ Основной эффект выполнения: состояние процесса изменяется согласно одному шагу вычисления и устройству процесса

Процессы могут **реагировать** на действия присваивания, порождая (планируя) действия своего вычисления

При выполнении действия вычисления могут планироваться новые действия присваивания и вычисления

Группы действий

Действия региона разбиваются на четыре группы, по убыванию приоритета: **активные**, **неактивные**, **отложенные** и **отслеживающие**

Порядок обработки действий текущего региона:

- ▶ Если в активной группе есть действия, то из неё **недетерминированно** выбирается и выполняется действие
 - ▶ Если есть хотя бы одно присваивание, то выбирается присваивание
 - ▶ Иначе выбирается вычисление
- ▶ Если активная группа пуста, но текущий регион непуст, то наиболее приоритетная группа региона становится активной
- ▶ Если текущий регион пуст, но есть непустой регион, то модельное время увеличивается так, чтобы регион, следующий по возрастанию времени, стал текущим
- ▶ Если все регионы пусты, то ничего не происходит (*ожидается команда на выполнение из консоли*)

«**Действие немедленно выполняется**» — так будем обозначать планирование активного действия в текущем регионе

Группы действий

Пример

Будем изображать регион как упорядоченную четвёрку групп по убыванию приоритета

Пусть в процессе симуляции все процессы завершились и получились такие значения точек и времени и регионы:

$\tau/1, x/0, y/0$

1: $(\{x \leftarrow 1, y \leftarrow 2\}, \{y \leftarrow 3\}, \{y \leftarrow 4\}, \emptyset);$

3: $(\emptyset, \emptyset, \{x \leftarrow 5\}, \emptyset);$

5: $(\emptyset, \emptyset, \{y \leftarrow 6\}, \emptyset)$

Тогда симуляция может быть продолжена так ...

Группы действий

Пример

$\tau/1, x/0, y/2$

1: ($\{x \leftarrow 1\}, \{y \leftarrow 3\}, \{y \leftarrow 4\}, \emptyset$); 3: ($\emptyset, \emptyset, \{x \leftarrow 5\}, \emptyset$); 5: ($\emptyset, \emptyset, \{y \leftarrow 6\}, \emptyset$)

$\tau/1, x/1, y/2$

1: ($\emptyset, \{y \leftarrow 3\}, \{y \leftarrow 4\}, \emptyset$); 3: ($\emptyset, \emptyset, \{x \leftarrow 5\}, \emptyset$); 5: ($\emptyset, \emptyset, \{y \leftarrow 6\}, \emptyset$)

$\tau/1, x/1, y/2$

1: ($\{y \leftarrow 3\}, \emptyset, \{y \leftarrow 4\}, \emptyset$); 3: ($\emptyset, \emptyset, \{x \leftarrow 5\}, \emptyset$); 5: ($\emptyset, \emptyset, \{y \leftarrow 6\}, \emptyset$)

$\tau/1, x/1, y/3$

1: ($\emptyset, \emptyset, \{y \leftarrow 4\}, \emptyset$); 3: ($\emptyset, \emptyset, \{x \leftarrow 5\}, \emptyset$); 5: ($\emptyset, \emptyset, \{y \leftarrow 6\}, \emptyset$)

$\tau/1, x/1, y/3$

1: ($\{y \leftarrow 4\}, \emptyset, \emptyset, \emptyset$); 3: ($\emptyset, \emptyset, \{x \leftarrow 5\}, \emptyset$); 5: ($\emptyset, \emptyset, \{y \leftarrow 6\}, \emptyset$)

$\tau/1, x/1, y/4$

1: ($\emptyset, \emptyset, \emptyset, \emptyset$); 3: ($\emptyset, \emptyset, \{x \leftarrow 5\}, \emptyset$); 5: ($\emptyset, \emptyset, \{y \leftarrow 6\}, \emptyset$)

$\tau/3, x/1, y/4$

3: ($\emptyset, \emptyset, \{x \leftarrow 5\}, \emptyset$); 5: ($\emptyset, \emptyset, \{y \leftarrow 6\}, \emptyset$)

$\tau/3, x/1, y/4$

3: ($\{x \leftarrow 5\}, \emptyset, \emptyset, \emptyset$); 5: ($\emptyset, \emptyset, \{y \leftarrow 6\}, \emptyset$)

$\tau/3, x/1, y/5$

3: ($\emptyset, \emptyset, \emptyset, \emptyset$); 5: ($\emptyset, \emptyset, \{y \leftarrow 6\}, \emptyset$)

$\tau/5, x/1, y/5$

5: ($\emptyset, \emptyset, \{y \leftarrow 6\}, \emptyset$)

$\tau/5, x/1, y/5$

5: ($\{y \leftarrow 6\}, \emptyset, \emptyset, \emptyset$)

$\tau/5, x/1, y/6$

5: ($\emptyset, \emptyset, \emptyset, \emptyset$)