

# УНИВЕРСИТЕТ ИТМО

Курс «Системы на кристалле»

## Лекция 4

# Технологии верификации, отладки и тестирования СнК

Быковский С.В

e-mail: [sergei\\_bykovskii@itmo.ru](mailto:sergei_bykovskii@itmo.ru)

Санкт-Петербург, 2019



# Ситуации нарушения функционирования

ГОСТ 27.002-2015 Надежность в технике. Основные понятия. Термины и определения

- **Отказ** - событие, заключающееся в нарушении работоспособного состояния объекта.
  - явный/скрытый отказ
  - перемежающийся отказ
  - деградационный отказ
  - и др.
- **Сбой** - Самоустраняющийся отказ или однократный отказ, устраняемый незначительным вмешательством оператора.

# Причины отказов

## 1. Ошибки проектирования

- Ошибки спецификации
- Ошибка в выборе проектных решений

## 2. Ошибки реализации

- Ошибки технологии производства
- Ошибки кодирования/программирования
- Инструментальная ошибка (работа компилятора, RTL-синтезатора и др.)

## 3. Условия эксплуатации

- Приобретаемые ошибки (старение, воздействие радиации и др.)
- Ошибки внешних воздействий (форматы данных, непрогнозируемая последовательность стимулов и др.)



## 1. Невычислительные отказы

- Отказы элементной базы

## 2. Вычислительные отказы

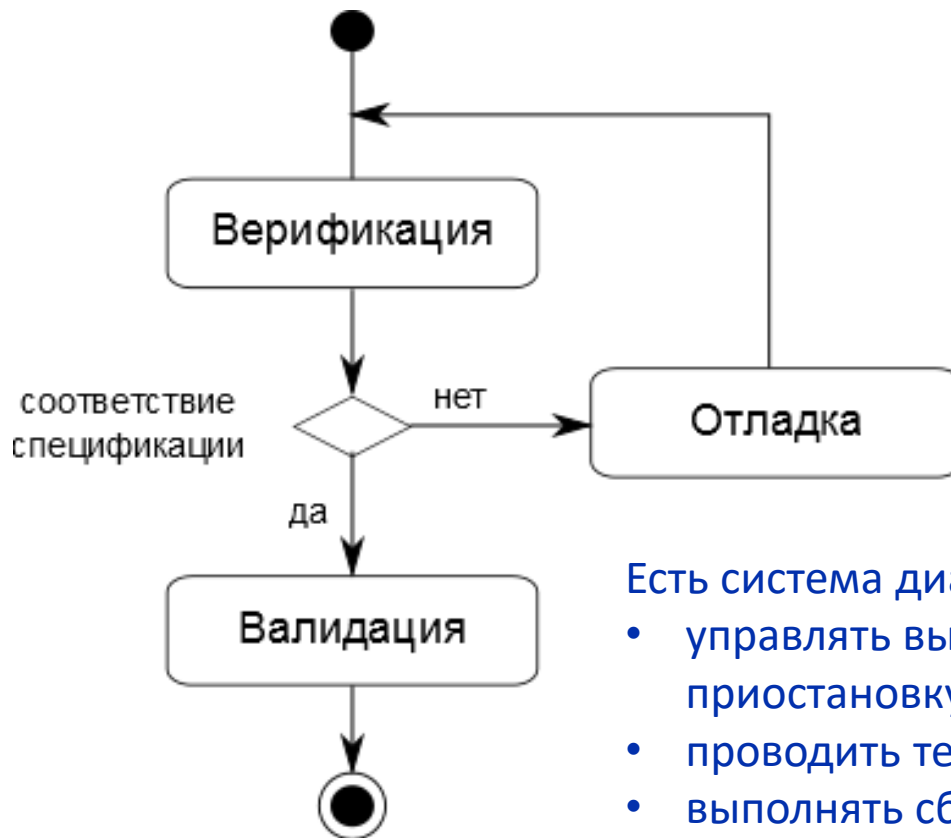
- Нарушение алгоритмов функционирования системы



## Верификация, валидация и тестирование

- ✓ **Верификация** – проверка соответствия параметров функционирования системы требованиям спецификации.
- ✓ **Валидация** – подтверждение того, что продукт соответствует требованиям эксплуатации.
- ✓ **Тестирование** – метод исследования системы посредством помещения её в различные ситуации и наблюдения за изменением её характеристик.

## Связь понятий: верификация, отладка, валидация



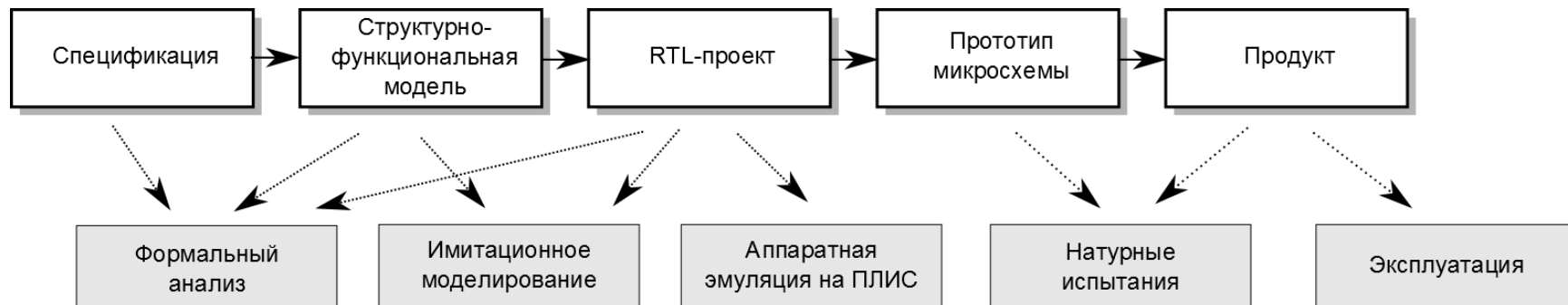
На практике нет трех систем:

- для верификации;
- для отладки;
- для валидации.

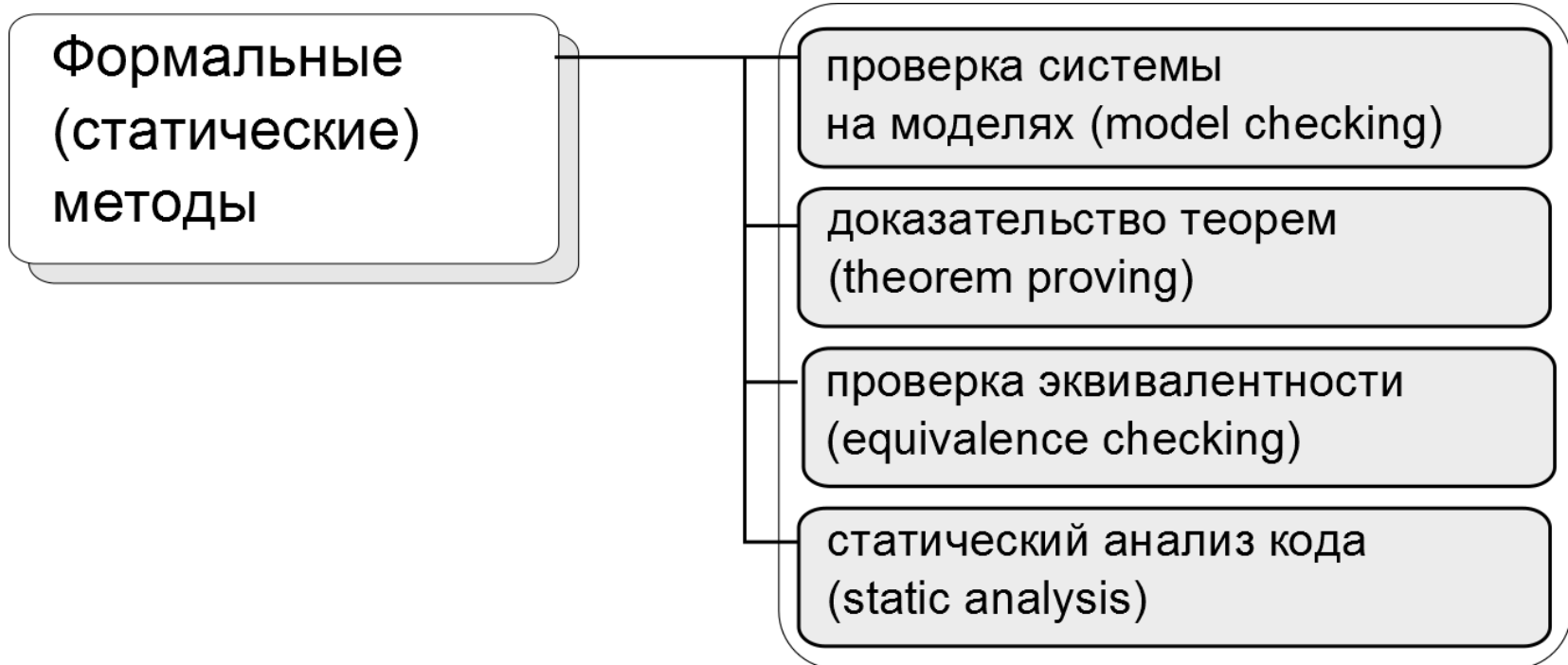
Есть система диагностики , которая позволяет:

- управлять вычислительным процессом (выполнять приостановку, запуск отдельных устройств),
- проводить тестирование
- выполнять сбор данных о работе системы

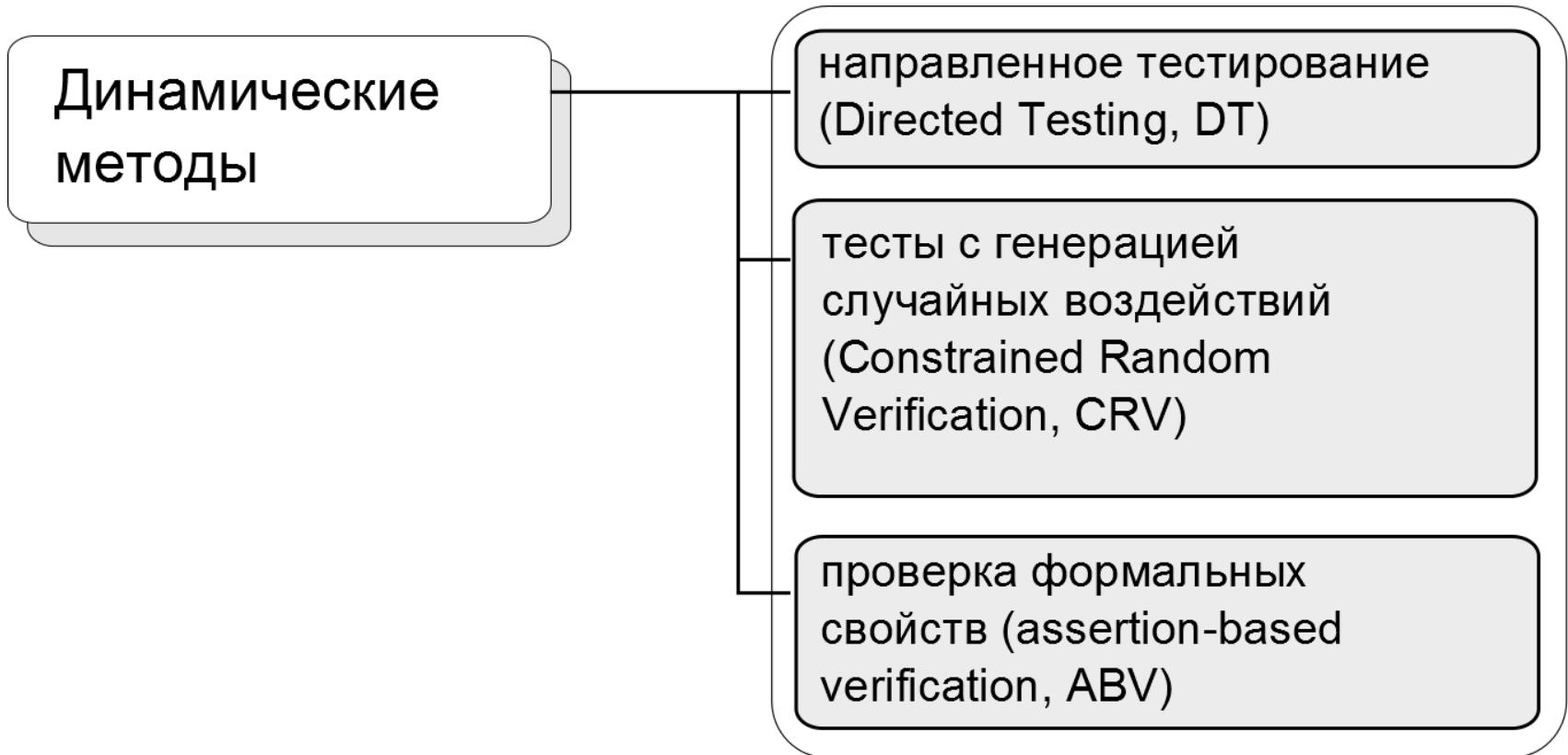
# Жизненный цикл проекта СнК



## Формальные методы



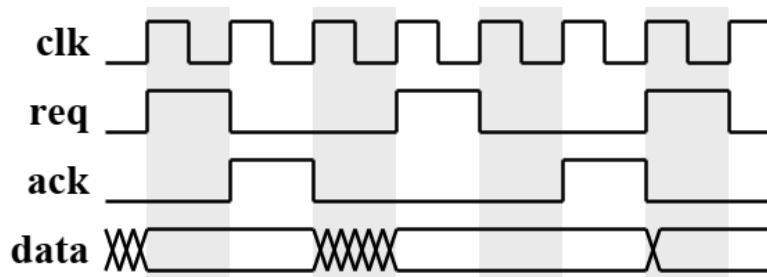
# Динамические методы верификации





## Пример задания утверждений (assertion)

Производится проверка, что после сигнала req должен следовать ack. Это должно быть обязательно выполнено перед любым следующим req.



```
sequence s_transfer;
  req ##1 !req [*1:max] ##0 ack;
endsequence
```

```
property p_transfer;
  @(posedge clk)
    disable iff (reset)
      req |-> s_transfer;
endproperty
```

```
a_transfer :
  assert property(p_transfer)
    else $error("illegal transfer");
```

## Пример задания рандомизированных значений

```
module test;
class randValues;
    rand int data_in;
    constraint c {
        data_in dist {
            [-30000: 30000] := 1,
            [70000: 100000] := 1
        };
    }
endclass
```

```
randValues r;

initial begin
    r = new();
    repeat(5) begin
        r.randomize();
        $display("R value = %d",
r.data_in);
    end
end

endmodule
```

## Варианты симуляции

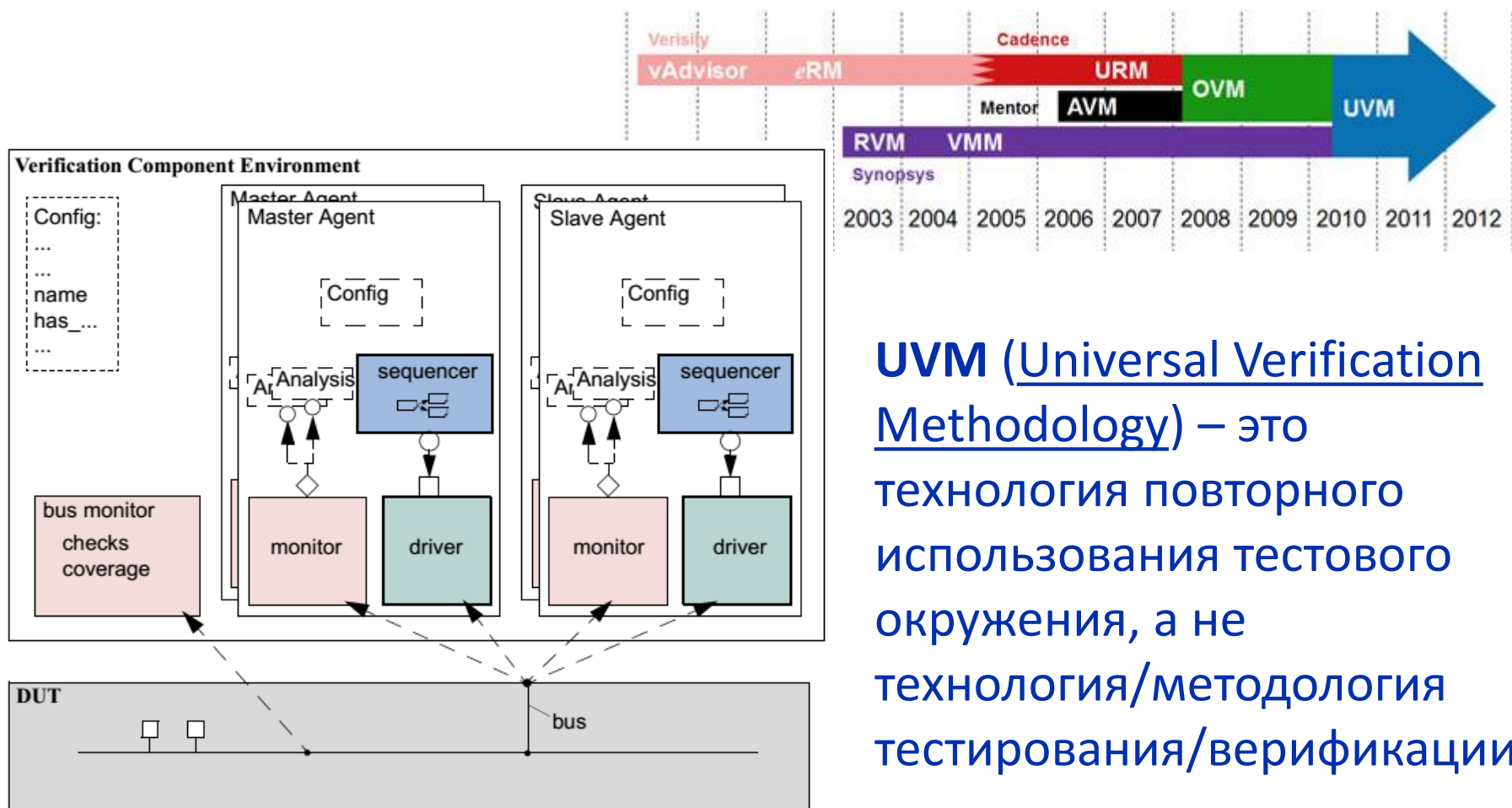
### Базовые:

- SIL (software in the loop) – тестируем программную модель аппаратуры. Окружение либо моделируем, либо берем реальное.
- HIL (hardware in the loop) – тестируется реальная аппаратура. Внешнее окружение задается моделью.

### Частные:

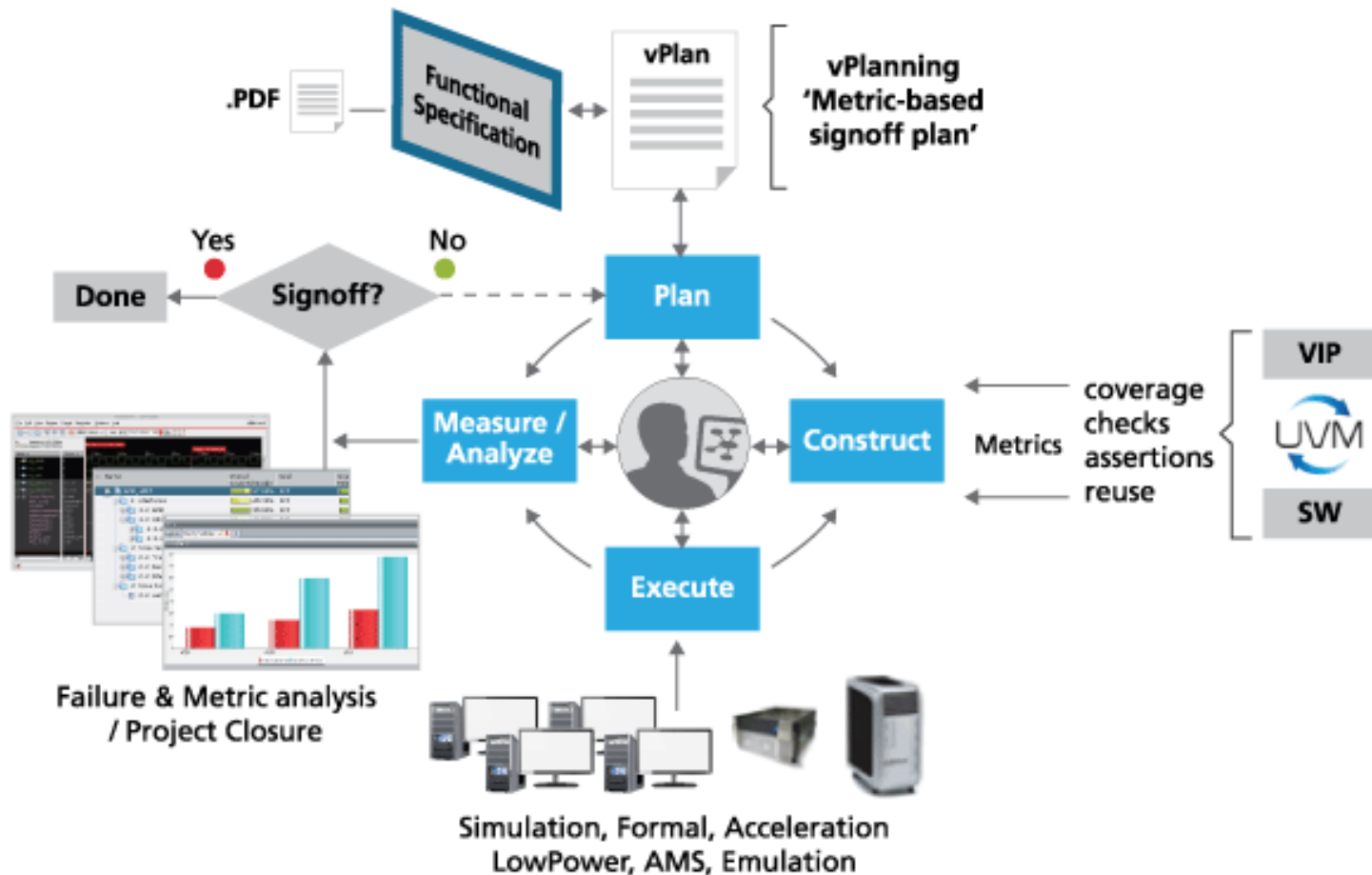
- MIL (model in the loop) – объектом тестирования является модель.
- PIL (processor in the loop) – объектом тестирования является реальный процессор.

# Инфраструктура тестирования (UVM)



**UVM (Universal Verification Methodology)** – это технология повторного использования тестового окружения, а не технология/методология тестирования/верификации

# Metric driven verification (Cadence)



## Доля пропущенных ошибок перед началом натурных испытаний



# Верификация СнК на этапе натурных испытаний

## Преимущества

- Система функционирует в реальном времени
- Система работает в реальных, либо максимально приближенным к реальным условиям

## Особенности

- Возможно наблюдать только за ограниченным количеством элементов системы
- Необходимо создавать специальные вычислительные средства для наблюдения, которые являются частью самой системы
- Один тест ни о чем не говорит. Необходимо проводить множественное тестирование.

# Средства натурной верификации СнК

## Внешние средства

- Логические анализаторы
- Осциллографы
- Анализаторы спектра

## Встроенные средства

- Встроенные логические анализаторы
- Встроенные средства протоколирования событий
- Мониторы-утверждения

## Преимущества

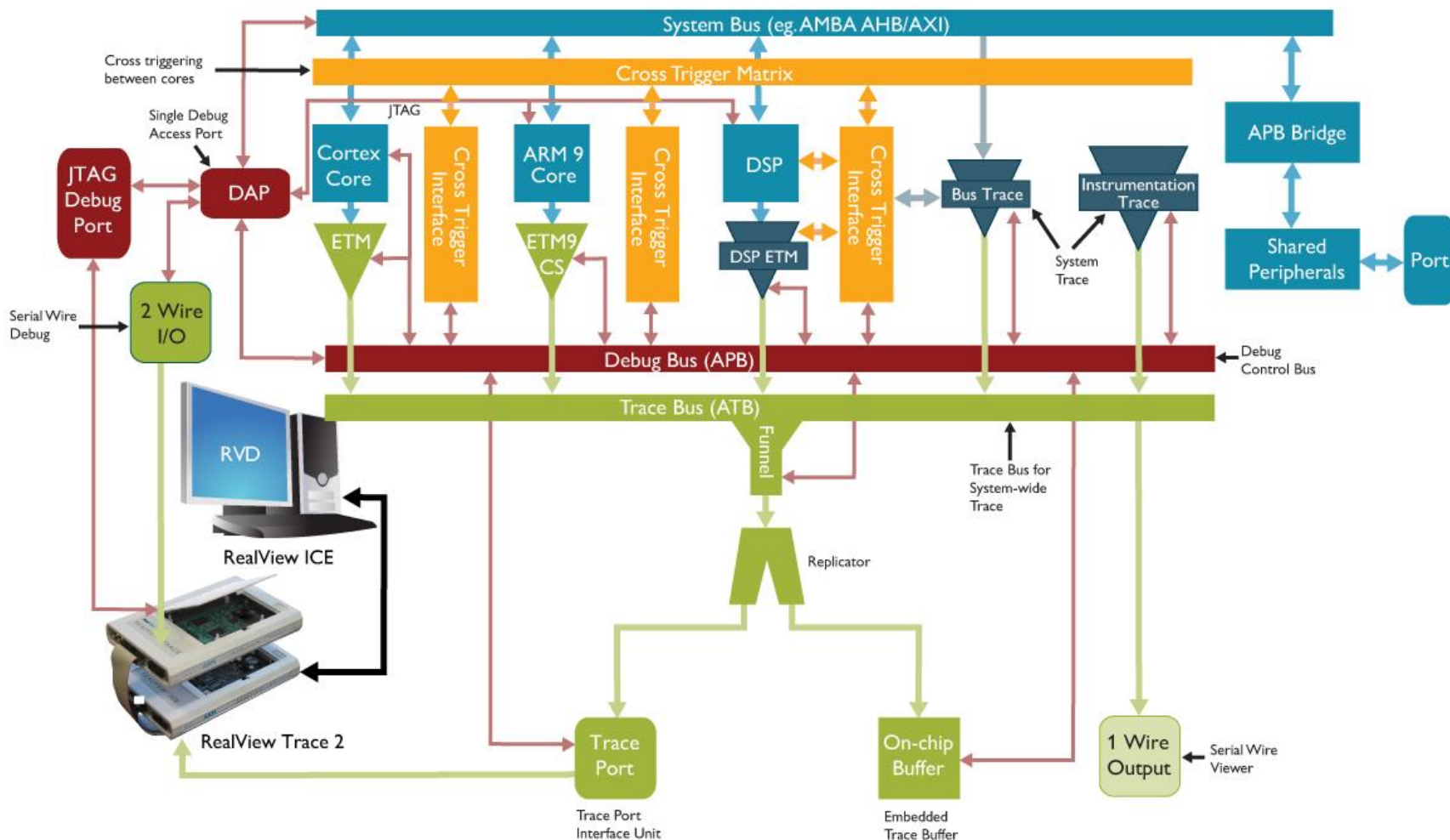
- Не надо проектировать диагностическую подсистему. Требуется иметь измерительный прибор.

## Преимущества

- Возможно наблюдать за «быстрыми» внутренними сигналами со скоростью работы системы



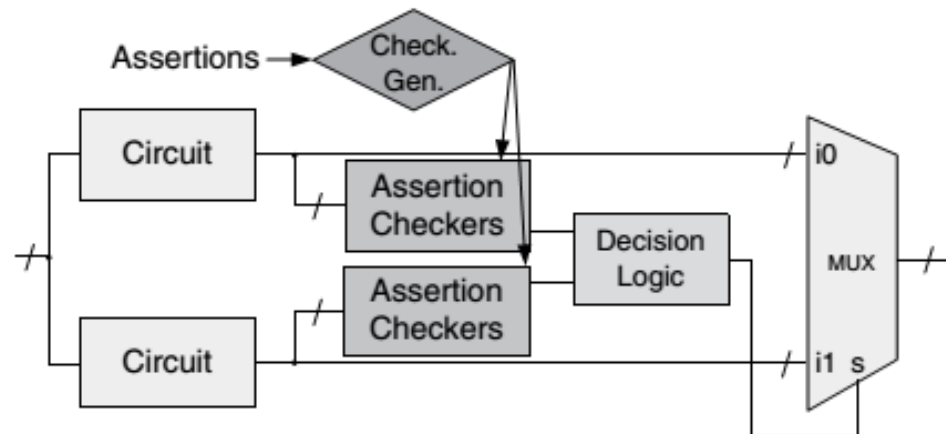
# ARM CoreSight



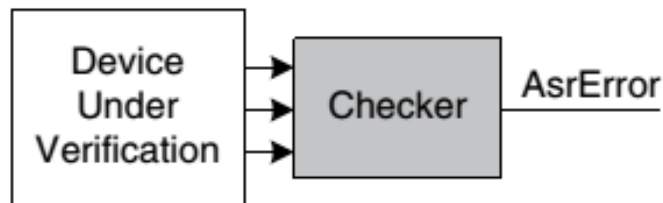
# Мониторы-утверждений (assertion checkers)

Компиляторы мониторов-утверждений:

- MBAC
- BusMOP
- FoCs

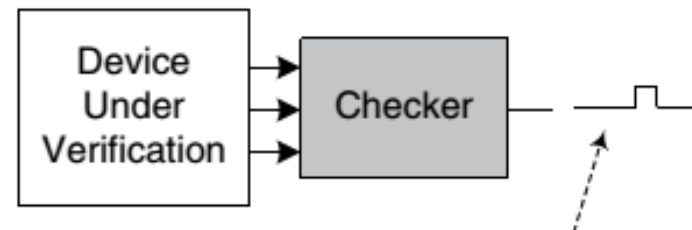


Checkers in Formal Verification



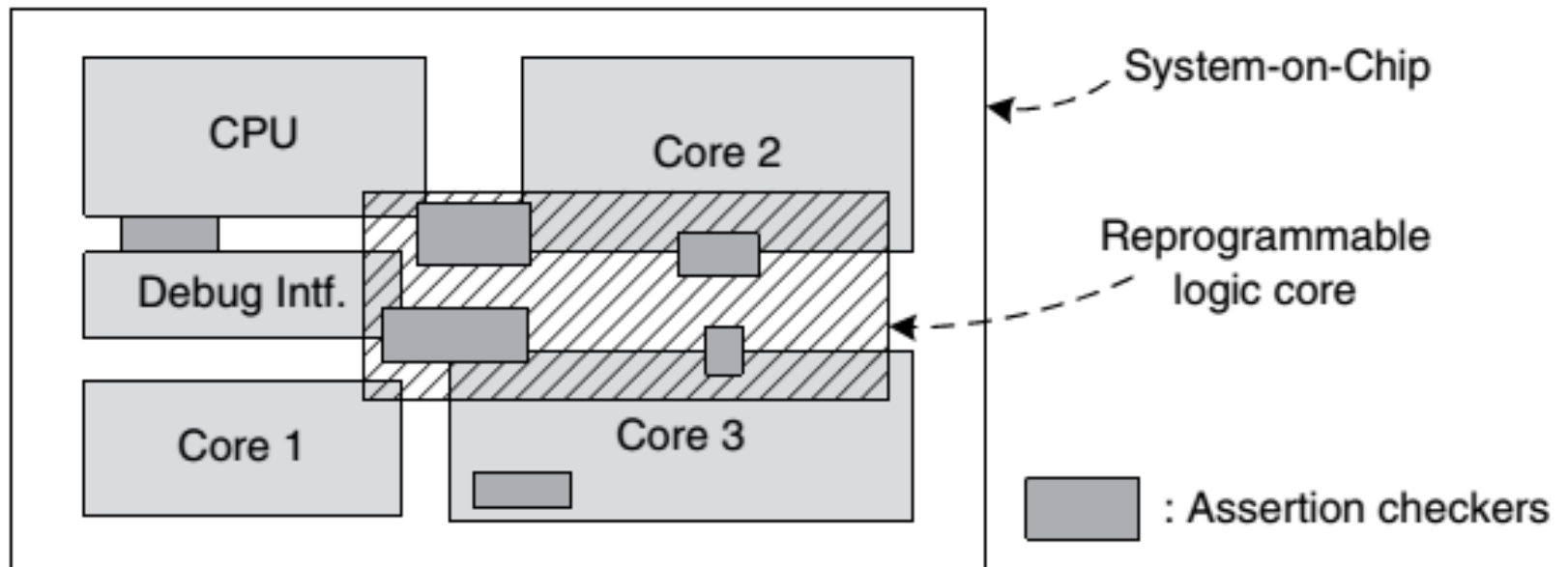
Model Check the Property:  $G \neg \text{AsrError}$

Checkers in Dynamic Verification

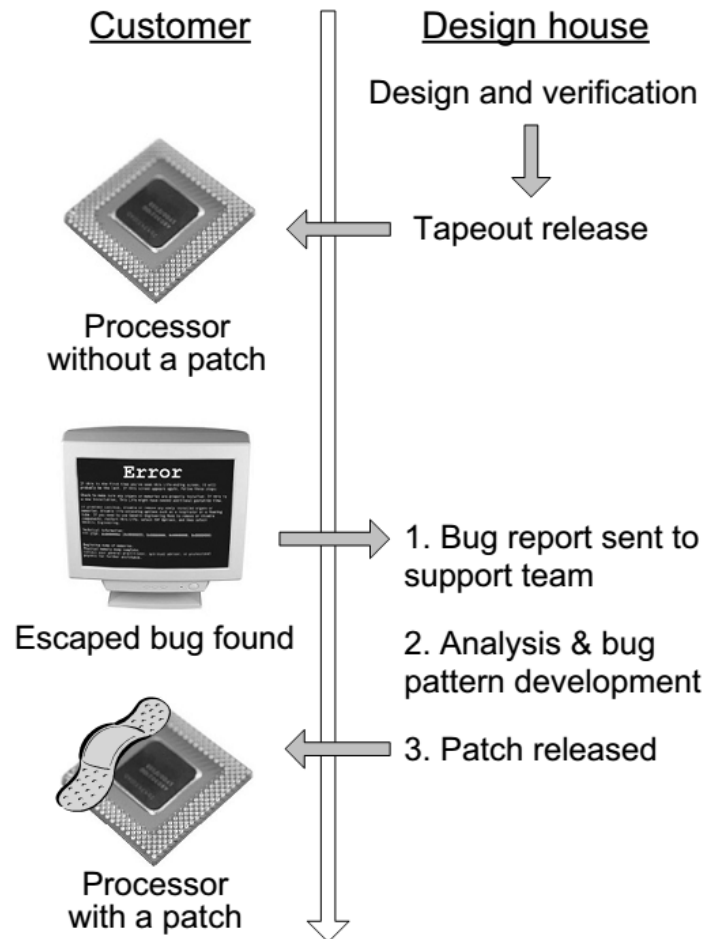


Simulate DUV+Checker: *trace*

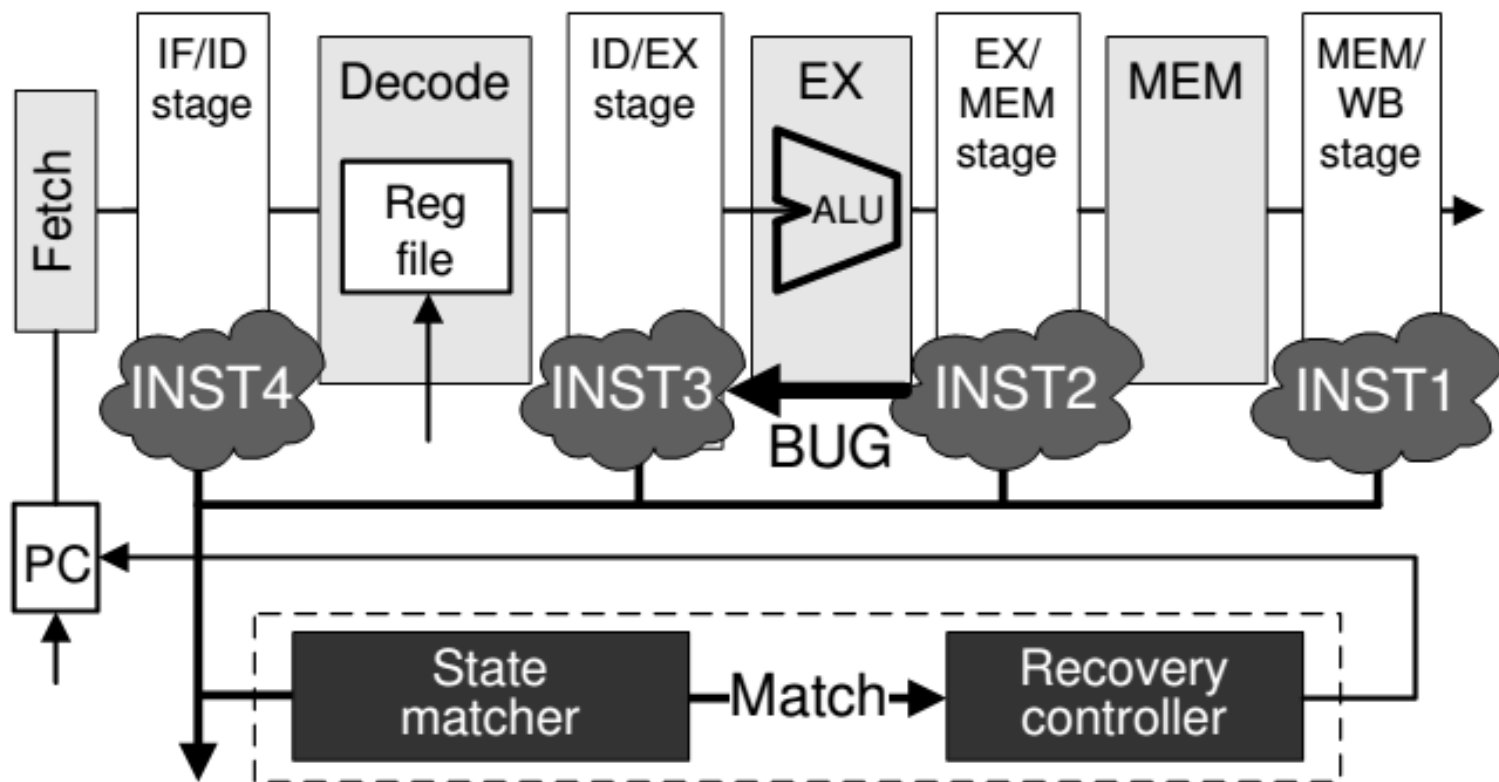
## Мониторы-утверждений



## Средства исправления ошибок на этапе эксплуатации (hardware patch)



## Средства исправления ошибок на этапе эксплуатации (hardware patch)





УНИВЕРСИТЕТ ИТМО

**Спасибо за внимание!**

[sergei\\_bykovskii@itmo.ru](mailto:sergei_bykovskii@itmo.ru)

Санкт-Петербург, 2019