



Faculty
of
Computer
science
Higher School of Economics

Факультет компьютерных наук

National Research University Higher School of Economics

Программный проект

Прототип онлайн-генератора тестовых
программ для микропроцессоров

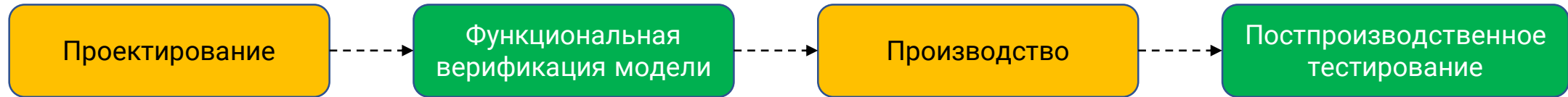
Prototype of Online Test Program
Generator for Microprocessors

Выполнил:

Литвинов Михаил Юрьевич, студент группы БПИ201

Научный руководитель:

Камкин Александр Сергеевич, к. ф.-м. н.,
в.н.с ИСП РАН, доцент ФКН НИУ ВШЭ



Динамическая верификация (тестирование):

- (Построение HDL-модели)
- (Симуляция тестируемого аппаратного обеспечения)
- Генерация тестовых сценариев
 - Коды на языке Ассемблера
- Проверка результатов выполнения тестовых сценариев



\ Spike

▶ Оффлайн-тестирование - генерация набора тестовых ситуаций без

использования системы, подлежащей тестированию

▶ Онлайн-тестирование - генерация набора тестовых ситуаций и их

непосредственное исполнение на системе, подлежащей тестированию

Функциональное тестирование на симуляторе аппаратного обеспечения:

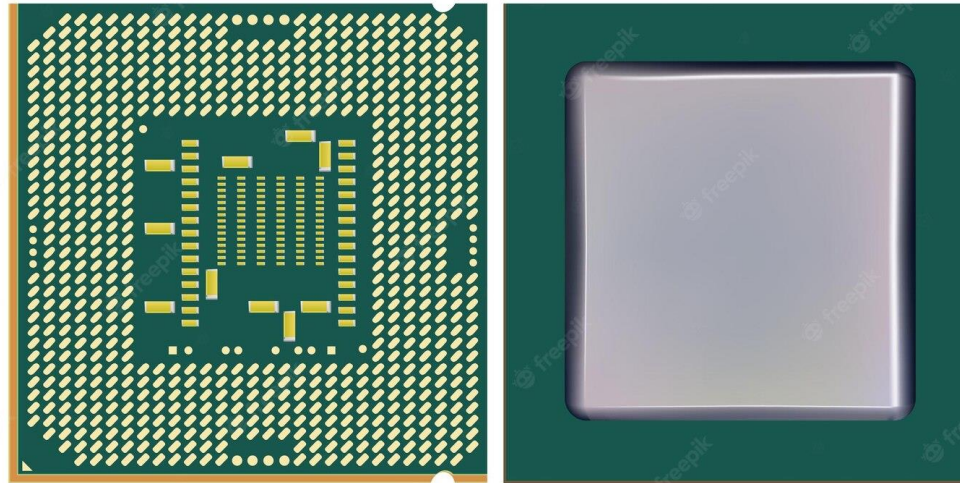
- Возможность изменения загружаемой модели поведения
- Требуется спроектированная HDL-модель
- Низкая скорость выполнения инструкций ($\sim 10^3$ операций / сек.)



QEMU

Функциональное тестирование на Сверхбольшой Интегральной Схеме (СБИС):

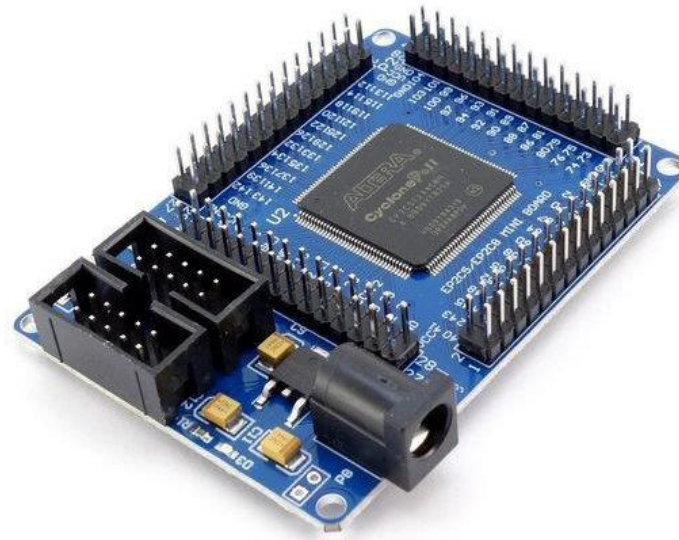
- Достигается целевая скорость выполнения операций ($\sim 10^8$ операций / сек.)
- Невозможность осуществления на ранних этапах тестирования
- Невозможность исправления ошибки проектирования на готовой схеме



Сверхбольшая Интегральная Схема

Функциональное тестирование на Программируемой Логической Интегральной Схеме (ПЛИС):





- Достигается высокая скорость выполнения операций ($>10^5$ операций / сек.)
- Возможность изменения модели микропроцессора
- Требуется спроектированная HDL-модель



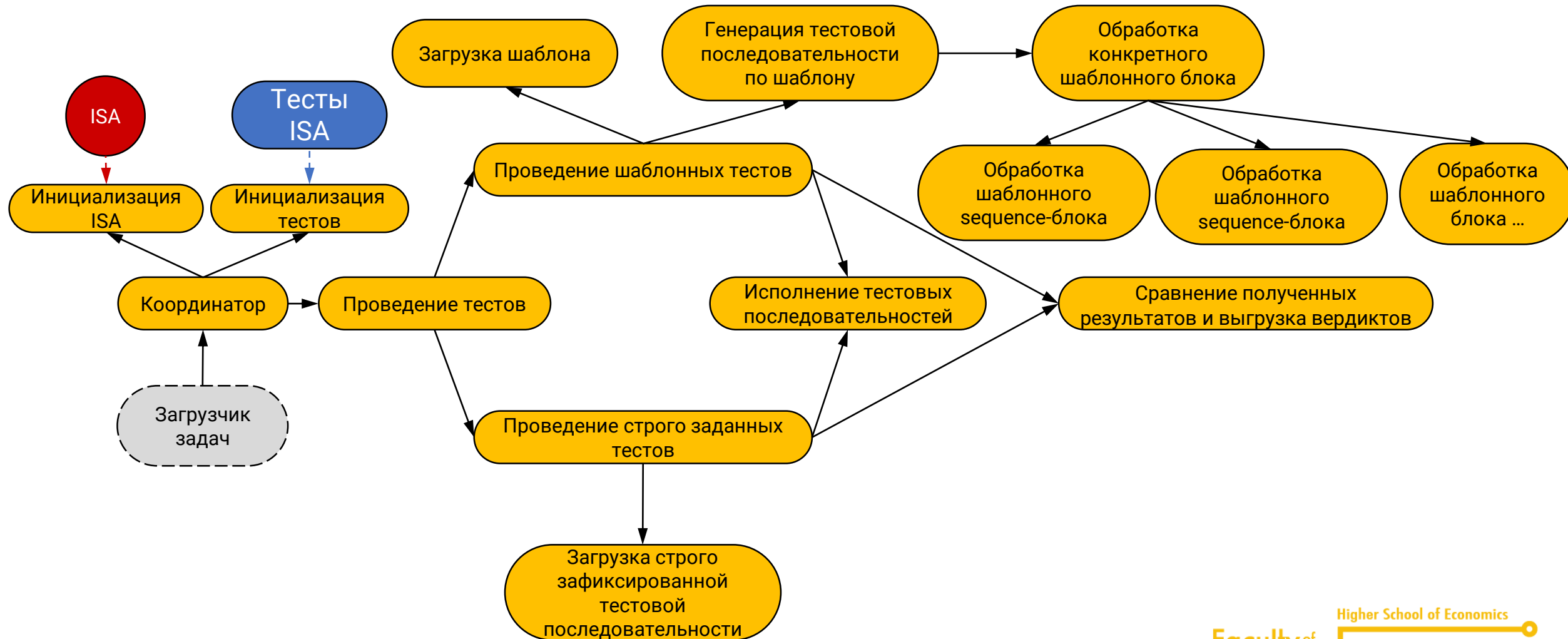
Программируемая Логическая Интегральная
Схема

➤ Создание упрощенного онлайн-генератора тестовых программ для микропроцессоров на основе частично заданных и строго зафиксированных описаний тестов, при этом успешно функционирующего как минимум в отношении некоторой RISC-V RV32I-совместимой модели микропроцессоров

- ▶ Углубленное изучение ISA RISC-V
- ▶ Изучение существующих проектных аналогов
- ▶ Проектирование общей архитектуры онлайн-генератора
- ▶ Проектирование упрощенного формата модели абстрактной ISA, достаточного для описания пользовательских инструкций с учетом пропущенных параметров
- ▶ Проектирование формата описания и алгоритма обработки шаблонов тестов
- ▶ Реализация RISC-V RV32I-совместимой модели ISA и набора тестов к ней на основе спроектированных форматов
- ▶ Реализация отдельных компонентов онлайн-генератора и их объединение в одном проектном решении
- ▶ Проведение тестирования и оценки качества проекта

-  Threadmill
 - Утилита-генератор тестовых программ в производственном тестировании (*post-silicon*) микропроцессоров
 - Закрытая программная реализация
-   MicroTESK
 - Оффлайн-генератор тестовых программ для микропроцессоров
 - Реализация с открытым исходным кодом
-  RISC-V-DV
 - Генератор случайных инструкций исключительно под работу с RISC-V
 - Требуется RTL-симулятор с поддержкой SystemVerilog и UVM 1.2
 - Реализация с открытым исходным кодом

- ▶ Использование пользовательских описаний целевой ISA, набора частично заданных и строго зафиксированных тестов и алгоритма функционально эквивалентного преобразования готовой тестовой последовательности
- ▶ Последовательная генерация и исполнение тестовых последовательностей для описанных шаблонов с заполнением пропущенных входных данных их инструкций
- ▶ Последовательная обработка и исполнение строго зафиксированных тестов
- ▶ Выгрузка итоговых вердиктов проведения тестов по заданному адресу ОЗУ с опциональным выводом на экран



- ▶ **InstructionGroup** – совокупность инструкций, описывающая тип, расположение и размер имеющихся у инструкции параметров
- ▶ **Instruction** – представление инструкции, описывающее постоянные характеристики вне зависимости от параметров (например, код операции)
- ▶ **PseudoInstruction** – инструкция с зафиксированными параметрами, реализующая определенную операцию (например, «mov» реализует «addi»)
- ▶ **ParametrizedInstruction** – представление некоторой (псевдо)инструкции с (частично) заполненными параметрами; используется в тестах

- Иерархическая структура блоков
- Использование итераторов для порождения инструкций

```
block {                                # upper level block
  block {                             # nested block
    add x0, x1, _, _                 # basic block
    add x1, x2, _, _
    add x2, x3, _, _
  }
  block {
    sub x0, x1, _, _
    sub x1, x2, _, _
    sub x2, x3, _, _
  }
  block {
    cmp x0, 0, _
    cmp x1, 0, _
    cmp x2, 0, _
  }
}
```

- ▶ Шаблонный блок - логически обособленная совокупность инструкций в рамках шаблона
- ▶ Дерево шаблона – иерархическая структура взаимосвязей между шаблонными блоками (вложенность относительно друг друга и упорядоченность для одного уровня вложенности)
- ▶ Корень дерева шаблонов – шаблонный блок, для которого не существует не вложенного для него блока
- ▶ В шаблоне всегда должна быть минимум одна достижимая инструкция возврата для успешного завершения тестовой последовательности

➤ Sequence – упорядоченный набор инструкций

➤ Pool – (не)упорядоченное (мульти)множество инструкций или блоков, из которого выбираются k инструкций или блоков

$$(1)seq \left\{ \begin{array}{l} (2)seq\{\dots \\ (3)pool(n = 1, BLOCKS) \left\{ \begin{array}{l} (4)seq\{\dots \\ (5)pool(n = 3, WITHREPEATS, NOORDER)\{\dots \\ (6)seq\{\dots \end{array} \right. \end{array} \right.$$

Общие средства реализации:

➤ Система сборки проектов на Си/C++ CMake (версия 3.18)

Средства реализации ISA-модели RISC-V RV32I:

➤ Кросс-компилятор языка Си (C99) из коллекции GCC для ISA RISC-V
(<https://github.com/riscv-collab/riscv-gnu-toolchain>)

➤ Программный эмулятор ISA RISC-V Spike
(<https://github.com/riscv-software-src/riscv-isa-sim>)

Особенности сборки исходных файлов проекта:

➤ Помимо необходимого для корректной сборки средствами CMake файла CMakeLists.txt было написано два bash-совместимых скрипта – общего скрипта compile.sh и скрипта для RISC-V RV32I-совместимой модели riscv.sh

- ▶ Спроектирован упрощенный формат модели абстрактной ISA, учитывающий частичное заполнение параметров инструкций
- ▶ Спроектирован формат древовидного описания шаблонов тестов, а также реализован алгоритм их обработки
- ▶ Реализован прототип RISC-V RV32I-совместимой модели микропроцессоров с набором соответствующих тестов в рамках спроектированных форматов
- ▶ Спроектирован и реализован прототип онлайн—генератора тестовых программ на основе частично заданных и строго зафиксированных тестов для микропроцессоров
- ▶ Проведено тестирование и оценка качества разработанного проекта на основе разработанного набора тестов с логгированием промежуточных шагов

- ▶ Поддержка компиляции под bare metal-системы
- ▶ Автоматизация генерации кода для описания ISA/тестов
- ▶ Добавление шаблонных блоков новых типов

- 1) Andrew Waterman, Krste Asanovic. The RISC-V Instruction Set Manual [Статья в Интернет]
Режим доступа: <https://riscv.org/wp-content/uploads/2017/05/riscv-spec-v2.2.pdf> , свободный.
(дата обращения: 14.11.2022)
- 2) CMake cross-platform build-process managing system. [Интернет-ресурс]
Режим доступа: <https://cmake.org/> , свободный. (дата обращения: 14.12.2022)
- 3) IBM Threadmill Post-Silicon Exerciser [Статья в Интернет]
Режим доступа: https://researcher.watson.ibm.com/researcher/view_group.php?id=1347 , свободный. (дата обращения: 13.11.2022)
- 4) MicroTESK configurable test program generator [Статья в Интернет]
Режим доступа: <http://www.microtesk.org/> , свободный. (дата обращения: 13.11.2022)

- 5) Random instruction generator for RISC-V processor verification [Интернет-ресурс] Режим доступа: <https://github.com/chipsalliance/riscv-dv> , свободный (дата обращения: 09.01.2023)
- 6) А.С. Камкин, А.М. Коцыняк, С.А. Смолов, А.Д. Татарников, М.М. Чупилко, А.А. Сортов.
Средства функциональной верификации микропроцессоров [Статья в Интернет]
Режим доступа: Сборник трудов ИСП РАН, <https://ispranproceedings.elpub.ru/jour/article/view/771>, свободный.
(дата обращения: 13.10.2022)

Благодарю вас за внимание!

myulitvinov@edu.hse.ru