

Функциональная верификация цифровой аппаратуры

Лекция 1

Схемотехника

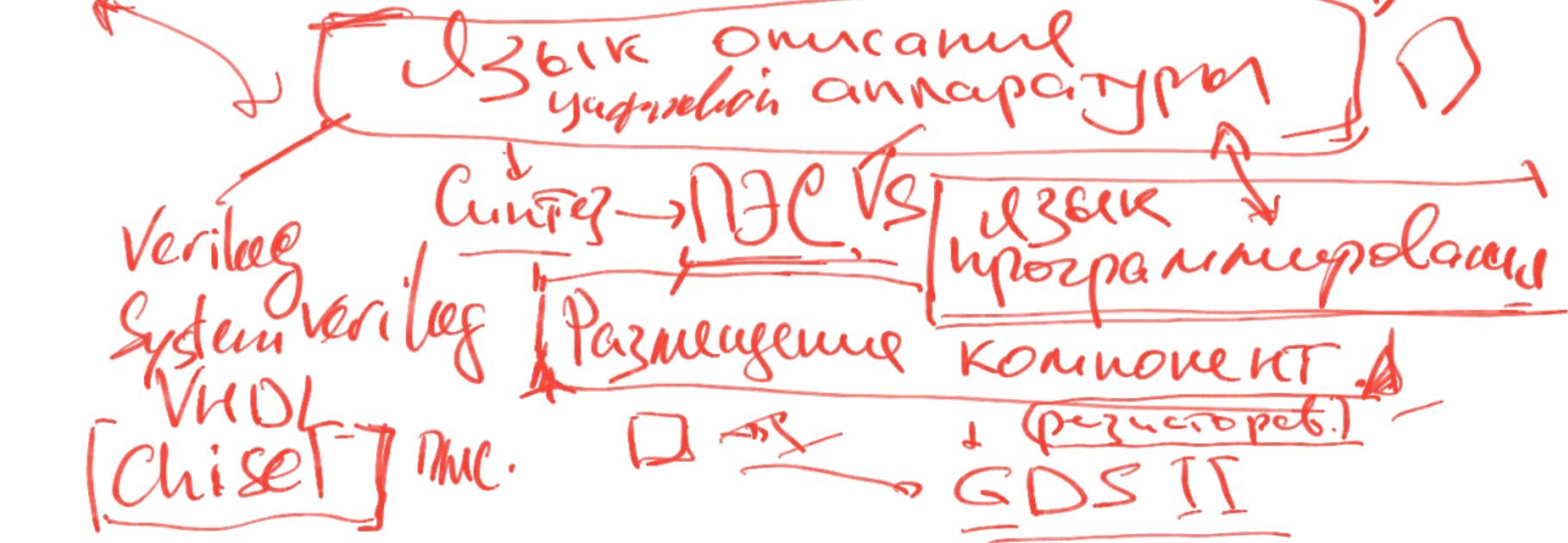
Фролов П.В.

September 2, 2025

Ларин

Дукель

Маршрут проектирования цифровой аппаратуры



Маршрут проектирования цифровой аппаратуры для изготовления микросхем "RTL-to-silicon"

Уровни абстракции и детализации

Представления проектируемой микросхемы на этапах маршрута проектирования

- Функциональная спецификация;
- алгоритмическое описание (микроархитектура);
- поведенческая модель (RTL-описание);
- схема соединений библиотечных элементов (вентилей). (принципиальная электрическая схема) (gate netlist);
- схема соединений транзисторов (заказное проектирование);
- план расположения транзисторов и соединений на слоях кристалла (GDSII: Graphic Database System).

Сверху вниз

- рост детализации;
- уменьшение абстракции.

Маршрут проектирования цифровой аппаратуры для ПЛИС

"RTL-to-FPGA"

Уровни абстракции и детализации

ПЛИС ?

- Функциональная спецификация;
- алгоритмическое описание (микроархитектура);
- поведенческая модель (RTL-описание);
- схема соединений элементов ПЛИС (принципиальная электрическая схема) (gate netlist);

FPGA ?

RTL-описание цифровой электрической схемы



- Что значит магические буквы RTL?
- Какая математическая модель соответствует этим трём буквам?

Уровень регистровых передач.

RTL-описание цифровой электрической схемы



RTL: Register Transfer Level / Уровень регистровых передач

- формальное описание поведения цифровой схемы (для моделирования);
- формальное описание функциональных блоков (для синтеза).

Математическая модель

- Время дискретно;
- состояние схемы $s \in S \subseteq 2^N$ определяется данными в регистрах (N – всех регистрах памяти устройства разрядностью n);
- RTL-описание определяет функцию перехода $f : (s \in S, v_{in})_t \rightarrow s'_{t+1} \in S; v_{in}$ – вектор значений входных сигналов устройства.

(Функциональная верификация в маршруте проектирования цифровой аппаратуры)

Маршрут проектирования

- функциональная спецификация;
- алгоритмическое описание (микроархитектура);
- поведенческая модель;
- принципиальная электрическая схема;
- схема соединений транзисторов;
- план расположения транзисторов и соединений на слоях кристалла.

Крепление.

Функциональная верификация в маршруте проектирования цифровой аппаратуры

Маршрут проектирования

- 
- Функциональная спецификация;
 - алгоритмическое описание (микроархитектура);
 - поведенческая модель;
 - принципиальная электрическая схема;
 - схема соединений транзисторов;
 - план расположения транзисторов и соединений на слоях кристалла.

Функциональная верификация – проверка соответствия поведенческой модели функциональной и алгоритмической спецификации.

Функциональная верификация в маршруте проектирования цифровой аппаратуры



Маршрут проектирования

- Функциональная спецификация;
- алгоритмическое описание (мираархитектура);
- поведенческая модель;
- принципиальная электрическая схема;
- схема соединений транзисторов;
- план расположения транзисторов и соединений на слоях кристалла.

Функциональная верификация – проверка соответствия поведенческой модели функциональной и алгоритмической спецификации.

Виды верификации

- Проверка модели
- Функциональная верификация
- Проверка временных характеристик (timing verification)
- Формальная верификация

Маршрут проектирования цифровой аппаратуры

Валидация

- Функциональная спецификация;
- алгоритмическое описание (микроархитектура);
- поведенческая модель;
- принципиальная электрическая схема;
- схема соединений транзисторов;
- план расположения транзисторов и соединений на слоях кристалла.

Маршрут проектирования цифровой аппаратуры

Валидация

- Функциональная спецификация;

- алгоритмическое описание (микроархитектура);

- поведенческая модель;

- принципиальная электрическая схема;

- схема соединений транзисторов;

- план расположения транзисторов и соединений на слоях кристалла.

Заказчик /
потенциальный
потребитель

Программо-
аппаратный
комплекс.

Валидация

подтверждение того, что требования для конкретного применения выполнены в реализации.

D

D

Как осуществлять функциональную верификацию?

Основные подходы к функциональной верификации



- Моделирование (тестирование): проверка корректности на одном входном векторе (тесте).
- Формальные методы: проверка эквивалентности, проверка свойств. Формальное описание требуемого (соответствующего спецификации) поведения (свойства) -> автоматическая (автоматизированная) проверка / доказательство того, что верифицируемая схема этому свойству удовлетворяет. Проверка корректности на множестве входных векторов.



Моделирование

Стандартный маршрут

EDA-tools.

1. Siemens

Mentor Graphics

RTB

FPGA

Совместное моделирование

Vivado.

2. Cadence.

Modelsim

Xcellium

iseas

VCS

verilator

Формальные методы

Примеры

- Статический анализ кода на предмет
 - соответствия формальному описанию языка;
 - соблюдению правил разработки;
 - отсутствия "подозрительных" мест.
- Формальная проверка выполнения поведенческой моделью свойств, заданными формулами временной (температуральной логики) (model checking).
- Формальная проверка корректности арифметических вычислений.
- Проверка эквивалентности двух разных моделей.

Формальные методы

Стандартный маршрут

Формальные методы

Ограничения

- Узкая область применения.
- Ошибки в ПО для формальной верификации.
- Ошибки в спецификации => проверка ошибочных свойств.
- Ресурсоёмкость – комбинаторный взрыв. Верификация устройства "по частям" – вопрос к корректности разбиения устройства "на части" и необходимость формулирования "частных" свойств.
- Ошибки формулирования свойств.

Основной принцип верификации – избыточность

Процесс верификации

- Преобразование спецификации в реализацию для верификации (тест, тестовое окружение, описание формально проверяемого свойства, модель для проверки эквивалентности и т.д.).
- Сравнение верифицируемой реализации (Design-under-verification) с реализацией для верификации.

Чем сильнее различаются верифицируемая реализация и реализация для верификации, тем меньше вероятность повторения в них одних и тех же ошибок.

Типы ошибок

- ошибки реализации;
- ошибки спецификации: неспецифицированная функциональность, противоречивость / несогласованность, нереализованная функциональность.

Избыточность и подходы к верификации

Моделирование

Реализация для верификации – элементы тестового окружения, ответственные за проверку поведения проверяемой реализации:

- чекеры тестового стенда;
- эталоны откликов.

Избыточность и подходы к верификации

Верификация формальными методами

Реализация для верификации:

- свойства;
- модели для проверки эквивалентности.

Маршрут функциональной верификации аппаратуры

Цикл функциональной верификации

Маршрут функциональной верификации аппаратуры

Цикл функциональной верификации

- ① планирование
- ② разработка тестового окружения (стенда и тестов) / окружения для верификации формальными методами
- ③ взаимная отладка RTL и верификационного окружения
- ④ регрессионные прогоны
- ⑤ верификация-на-кремнии (post-silicon verification)
- ⑥ escape analysis : определение причин необнаружения ошибок на стадии разработки RTL

Цикл функциональной верификации

Регрессионные прогоны

- регулярность
- полнота
- воспроизводимость
- случайность
- ресурсоемкость и избыточность

Цикл функциональной верификации

Готовность к выдаче проекта на фабрику

Цикл функциональной верификации

Готовность к выдаче проекта на фабрику

Критерии готовности проекта с финализированным физ. проектированием

- Отсутствие найденных неисправленных критических ошибок.
- Успешное прохождение регрессионных прогонов.
- Готовность тестов по тестпланам.
- Достижение заданного уровня тестового покрытия.
- Пройденные маршруты верификации формальными методами.

Цикл функциональной верификации

Верификация-на-кремнии (Post-silicon verification)

Цикл функциональной верификации

Разбор полётов (Escape analysis)

- Почему логические ошибки не были найдены вовремя?
- Что нужно улучшить в тестовом окружении для того, чтобы подобные ошибки (в т.ч. именно найденные) были найдены?

Валидация маршрута верификации применительно к данной разработке.

Взаимная отладка RTL-описания и тестового окружения

Взаимодействие с разработчиками

Документирование взаимодействия верификаторов и разработчиков RTL необходимо.

Жизненный цикл отчёта об ошибке

- NEW Верификатор открывает новый отчёт об ошибке.
- ASSIGNED Инженер подтверждает наличие ошибки и приступает к её вдумчивому исправлению.
- RESOLVED Работа со стороны инженера завершена.
 - FIXED Ошибка исправлена.
 - INVALID Ошибка не подтверждена (например, ошибка в тестовом окружении).
 - DUPLICATE Повторная ошибка.
- VERIFIED Исправление проверено.
- CLOSED Конец жизненного цикла ошибки.
- REOPENED Исправление проверено и в нём найдена ошибка. Либо повторное воспроизведение ошибки при регрессионных прогонах, либо недоисправленная ошибка с проявлением, не выявленном при первичной проверке.

Документированное взаимодействие верификаторов с разработчиками

Требования к оформлению отчёта об ошибке

Сообщение об ошибке RTL должно содержать

- исходные данные для воспроизведения ошибки:
 - номера редакций RTL-описания и тестового окружения ;
 - команды сборки RTL-модели и запуска моделирования;
 - команду запуска ;
- описание ошибочного поведения:
 - описание исходной ситуации (что делает тест перед ошибкой, с цитатами из журнала моделирования),
 - что ожидается (в соответствии с документацией),
 - что идёт не так (с цитатой из журнала моделирования);
- идентификаторы воспроизведения:
 - характерная цитата из журнала моделирования (например, время моделирования в тактах);
 - полный журнал или трасса сигналов.

Уровни (этапы) логической верификации

- Инженерная (designer-level)
- Модульная (автономная) (unit-level)
- Верификация библиотечного блока (core-level)
- Верификация микросхемы (chip-level)
- Верификация системного уровня (board-level, system-level): проверка взаимодействия проверенных компонент.
- Совместная верификация программно-аппаратного комплекса.

Автономная и системная верификация (МЦСТ)

Маршрут проектирования программно-аппаратного комплекса

Валидация и функциональная верификация

Литература

- William K. Lam. Hardware Design Verification: Simulation and Formal Method-Based Approaches
- Bruce Wile, John C. Goss, Wolfgang Roesner. Comprehensive Functional Verification: The Complete Industry Cycle.