

# Языки описания схем

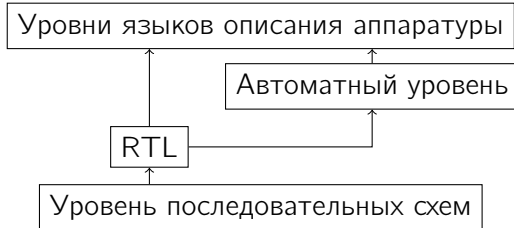
[mk.cs.msu.ru](http://mk.cs.msu.ru) → Лекционные курсы → Языки описания схем

## Блок 11

Verilog:  
общие вступительные слова

Лектор:  
**Подымов Владислав Васильевич**  
E-mail:  
**[valdus@yandex.ru](mailto:valdus@yandex.ru)**

ВМК МГУ, 2024/2025, осенний семестр



Разработка **последовательной схемы** и/или **RTL-описания схемы** — трудоёмкий и неустойчивый к ошибкам процесс:

- ▶ имеется *декларативное* описание поведения схемы
- ▶ из этого описания *методом пристального взгляда* извлекается основная масса триггеров/регистров
- ▶ схема *вручную* дополняется логическими вентилями/булевыми функциями, соединениями и вспомогательными триггерами/регистрами

В **языках описания аппаратуры** используются понятия и подходы, более близкие к декларативному описанию схем и позволяющие меньше задумываться о точной расстановке элементов схемы при её разработке

Verilog — это<sup>1</sup> один из двух самых популярных на данный момент языков описания цифровых микросхем<sup>2</sup>

Для краткости будем иногда писать «V» вместо «Verilog»

Изначально этот язык создавался для **программной симуляции** схем:

- ▶ схема разрабатывается другими средствами
- ▶ на языке Verilog описывается **программная модель**, поведение которой приблизительно соответствует поведению схемы
- ▶ модель запускается (*как обычная программа*) и выдаёт информацию об изменении значений сигналов во времени и другие отладочные данные

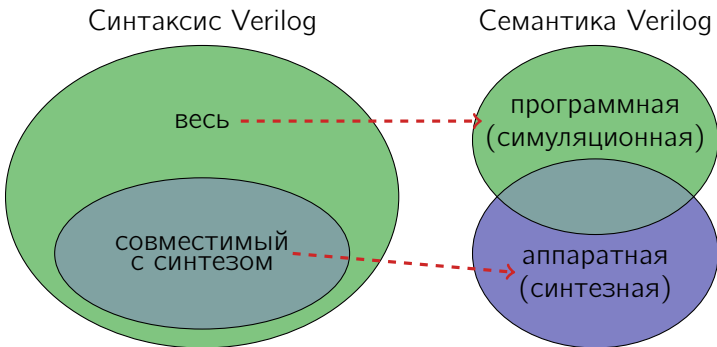
Язык оказался настолько удобным, что стал повсеместно применяться и для **синтеза** реальных схем

---

<sup>1</sup> Если считать его расширения и модификации, например, SystemVerilog

<sup>2</sup> Два самых популярных в мире языка описания аппаратуры — Verilog и VHDL

Эти языки похожи, при этом синтаксис Verilog проще, так что остановимся на нём



Два главных документа, описывающих  $\mathcal{V}$  как  
(1) средство симуляции и (2) средство синтеза:

1. IEEE Standard for Verilog Hardware Description Language  
(в курсе обсуждается версия **2005**)
2. IEEE Standard for Verilog Register Transfer Level Synthesis  
(в курсе обсуждается версия **2002**)

Синтаксис Verilog местами очень похож на синтаксис C/C++<sup>1</sup>

Это сходство позволит избежать долгих объяснений «с нуля», но следует иметь в виду, что оно поверхностно:

- ▶ Итог сборки кода
  - ▶ C/C++: машинный код, выполняемый процессором
  - ▶ Verilog: цифровая микросхема (*в том числе и сам процессор*)
- ▶ Трактовка переменных:
  - ▶ C/C++: последовательно изменяемые области памяти
  - ▶ Verilog: выделенные точки микросхемы
- ▶ Трактовка выражений и команд:
  - ▶ C/C++: команды машинного кода
  - ▶ Verilog: наборы логических вентилей и описание поведения подсхем
- ▶ ...

---

<sup>1</sup> И не просто так похож: создатели языка перенесли много синтаксических деталей из C, чтобы язык был *интуитивно понятнее*