

Языки описания схем

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

Блок 1

Организационные вопросы

лектор:

Подымов Владислав Васильевич

e-mail:

valdus@yandex.ru

Осень 2017

О чём этот курс

1. Основы цифровой схемотехники

- ▶ Как устроена цифровая аппаратура
- ▶ Как проектировать цифровую аппаратуру, и причём здесь дискретная математика

2. Язык Verilog

- ▶ Проектирование аппаратуры при помощи высокоуровневого описания поведения

3. Основы архитектуры процессоров

- ▶ Как спроектировать процессор от ассемблера вниз

Литература

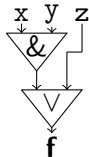
1. Harris, Harris. Digital design and computer architecture. 2nd ed. 2013.
 2. Перевод: Харрис, Харрис. Цифровая схемотехника и архитектура компьютера. 2-е изд. Открытая электронная версия.
 3. Patterson, Hennesy. Computer organization and design: the hardware/software interface. 5th ed. 2013.
 4. Thomas, Moorby. The Verilog hardware description language. 5th ed. 2008.
 5. Любой другой справочник по языку Verilog
- ▶ **Основы цифровой схемотехники:** 1, 2
 - ▶ **Язык Verilog:** 4, 5
 - ▶ **Основы архитектуры процессоров:** 1, 2, 3

Полезные ссылки

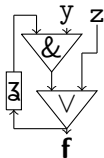
1. mk.cs.msu.ru → Спецкурсы → Программируемые логические интегральные схемы
2. mk.cs.msu.ru → Семинары → Практикум по пакетам проектирования сверхбольших интегральных схем (осенний семестр 2016 года)
3. asic-world.com
 - ▶ здесь можно найти много справочных материалов и полезных советов по проектированию схем в целом и языку Verilog в частности
4. edaplayground.com
 - ▶ здесь можно играть в написание схем на языке Verilog через web-интерфейс
5. opencores.org
 - ▶ просто посмотреть и испугаться: здесь можно найти готовые модули на языке Verilog

Рекомендуется вспомнить

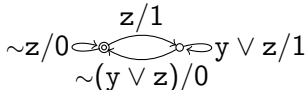
1. Булевы функции $f(0, 1, 0) = 0$
2. Булевы формулы $x \& y \vee z$
3. Схемы из функциональных элементов (СФЭ)



4. Элемент задержки (з)



5. Автоматы-преобразователи Мура и Мили



Как получить зачёт

Для получения зачёта необходимо и достаточно выполнить одно большое задание:

спроектировать и протестировать маленький процессор, который почти ничего не умеет, но всё же способен делать что-то полезное, на языке Verilog

Чуть более подробно, этот процессор должен

- ▶ уметь обрабатывать инструкции арифметики целых чисел
- ▶ уметь обрабатывать инструкции ветвления
- ▶ уметь работать с загрузкой, изображающей память
- ▶ содержать автомат, управляющий выполнением инструкций
- ▶ быть протестирован в функциональном симуляторе (программе на компьютере)

Конец блока 1