

Практическое Задание №3

Практическое задание №3 состоит из 5 упражнений:

- Первое в файле `03_01_detect_sequence_using_fsm.sv`
- Второе в файле `03_02_detect_sequence_using_shift_reg.sv`
- Третье в файле `03_03_serial_divisibility_using_fsm.sv`
- Четвертое и пятое в папке `03_04_sqrt_formula_fsms`

В каждом упражнении есть секция `Example` с модулем для примера, секция `Task` с описанием задания, в которой нужно описать ваше решение, а так же секция `Testbench`, которая осуществляет минимальную проверку работоспособности вашего решения.

Предисловие

В процессе работы с упражнениями 1-3, возможно запускать проверку вашего решения отдельно с помощью команды

```
iverilog -g2005-sv {SOLUTION_FILENAME}.sv && vvp a.out
```

В секции `Testbench` любого из заданий можно убрать комментарий у строк `$dumpfile;` и `$dumpvars;` для генерации `dump.vcd` файла при запуске. В файле содержится временная диаграмма описывающая изменения на всех проводах и регистрах во время симуляции работы модуля.

Можно воспользоваться командой `gtkwave dump.vcd` для просмотра файла, либо убрать комментарий у этой строки в файле скрипта `run_all_using_iverilog`.

Упражнение 1. Распознавание бинарной последовательности с помощью FSM

Ознакомиться с примером детектирования 4-х битной последовательности.

Задание: Реализовать модуль для детектирования 6-ти битной последовательности `110011` используя конечный автомат.

Упражнение 2. Распознавание бинарной последовательности с помощью Shift Register

Ознакомиться с примером детектирования 4-х битной последовательности.

Задание: Реализовать модуль для детектирования 6-ти битной последовательности `110011` используя сдвиговый регистр.

Упражнение 3. Последовательная проверка делимости числа

Ознакомиться с примером детектирования делимости числа на 3.

Ниже приведён пример работы модуля, вывод и внутреннее состояние в процессе. На вход модуля подаётся лишь самый правый бит:

| binary number | Div by 3 | State |
|---------------|----------|-------|
| 0 | yes | mod_0 |
| 01 | no | mod_1 |
| 011 | yes | mod_0 |
| 0110 | yes | mod_0 |
| 01101 | no | mod_1 |
| 011010 | no | mod_2 |
| 0110100 | no | mod_1 |
| 01101001 | yes | mod_0 |

Задание: Реализовать модуль последовательного детектирования делимости числа на 5 используя конечный автомат.

Упражнения 4 и 5. Вычисление формулы с помощью FSM

Введение

Директория `03_04_sqrt_formula_fsms` содержит примеры, тестбенчи, заготовки решений и вспомогательный код для 4-го и 5-го упражнений.

Для выполнения упражнений, необходимо использовать готовый модуль `isqrt.sv` в качестве чёрного ящика и написать FSM для вычисления двух формул.

Модуль `isqrt.sv` вычисляет целочисленный квадратный корень (integer square root) с фиксированной латентностью (временем в тактах между поступлением аргумента на вход и получением результата на выходе).

Модуль начинает вычисление при выставлении сигнала `x_vld`, и сообщает о готовности (валидности) результата выставляя сигнал `y_vld`.

Структура директории следующая:

```

├─ black_boxes // готовый модуль isqrt
│   ├── isqrt.sv
│   ├── isqrt_slice_comb.sv
│   └─ isqrt_slice_reg.sv
├─ testbenches
│   ├── formula_tb.sv // Основной код тестбенча
│   ├── isqrt_fn.svh  // Математическая формула isqrt для верификации
│   └─ tb.sv          // Запуск трёх тестбенчей для разных формул
├─ formula_1_fn.svh  // Эталонная формула 1 (используется для верификации)
├─ formula_1_impl_1_fsm.sv // Пример реализации формулы 1
├─ formula_1_impl_1_fsm_style_2.sv // Альтернативная реализация формулы 1
├─ formula_1_impl_1_top.sv
├─ formula_1_impl_2_fsm.sv // файл с Упражнением 4
├─ formula_1_impl_2_top.sv
├─ formula_2_fn.svh  // Эталонная формула 2 (используется для верификации)
├─ formula_2_fsm.sv  // файл с упражнением 5
├─ formula_2_top.sv
├─ run_all_using_iverilog_under_linux_or_macos_brew.sh
└─ run_all_using_iverilog_under_windows.bat

```

Замечание: Создавать экземпляры модуля `isqrt` самостоятельно запрещается. Необходимо работать с модулем через входы и выходы `isqrt_x` и `isqrt_y` модуля упражнения.

Упражнение 4

Необходимо ознакомиться с формулой в файле `formula_1_fn.svh` и примером конечного автомата для последовательного вычисления этой формулы в файле `formula_1_impl_1_fsm.sv` или `formula_1_impl_1_fsm_style_2.sv`.

Задание:

В файле `formula_1_impl_2_fsm.sv`, имплементировать вычисление Формулы 1 используя два модуля `isqrt` одновременно. Необходимо вычислить два из трёх значений параллельно. Далее, вычислить оставшееся значение и предоставить результат суммы.

Упражнение 5

Необходимо ознакомиться с формулой в файле `formula_2_fn.svh`.

Задание:

В файле `formula_2_fsm.sv`, имплементировать последовательное вычисление Формулы 2 используя один модуль `isqrt`.