**(10 февраля) Вводная работа.**

**Знакомство с библиотекой** **SystemC. Создание первого проекта Hello world.**

Программа:

1. В среде NetBeans создать новый проект.
2. В настройках компоновщика (Linker) для проекта подключить библиотеку SystemC (через PkgConfig)
3. Используя исходный код программы из примера /usr/share/doc/systemc/examples/sysc/simple\_fifo, создать программу, моделирующую работу очереди.

<описание архитектуры проекта>

1. Выполнить компиляцию проекта, убедиться в правильности подключения библиотеки SystemC.
2. Запустить созданное приложение, наблюдая результаты моделирования устройства в консоли.
3. Модифицировать созданное устройство, изменив тип данных, с которыми работает очередь, на int. Соответствующим образом модифицировать тест.

**(17 февраля) Первая работа.**

**Знакомство с описанием синхронных устройств на RTL уровне на языке SystemC.**

Программа:

1. Скопировать в локальную папку проект с примером регистра.
2. Выполнить компиляцию проекта. Запустить созданное приложение, наблюдая результаты моделирования устройства в консоли. Проверить правильность работы устройства, открыв сгенерированный vcd файл в GTKWave.
3. Разработать собственные устройства: счетчик и сдвигающий регистр, - с дополнительными функциями по индивидуальному заданию.
4. Создать тесты с самопроверкой для всех основных и дополнительных функций разработанных устройств.

Основные входы/выходы счетчика:

* clk
* areset
* sreset\_n
* dout

Основные входы/выходы сдвигающего регистра:

* clk
* areset
* sreset
* cin
* cout
* dout

Варианты индивидуальных заданий

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № п/п | counter | | | shift\_register | | |
| ena | load | up/down | ena | load | left/right |
| 1 | + |  |  |  | + |  |
| 2 | + |  |  |  |  | + |
| 3 |  | + |  | + |  |  |
| 4 |  | + |  |  |  | + |
| 5 |  |  | + | + |  |  |
| 6 |  |  | + |  | + |  |

**(3 марта) Вторая работа.**

**Знакомство с описанием комбинаторных устройств на языке SystemC.**

Программа:

1. Скопировать в локальную папку проект с примером регистра (с подсчетом четности).
2. Выполнить компиляцию проекта. Запустить созданное приложение, наблюдая результаты моделирования устройства в консоли. Проверить правильность работы устройства, открыв сгенерированный vcd файл в GTKWave.
3. Разработать конечный автомат Мура по индивидуальному заданию.

<схема КА с делением на синхронную и комбинаторную части>

1. Создать тесты с самопроверкой для всех переходов конечного автомата.

<таблица с вариантами (таблица истинности для переходов и выходов КА)>

**(10 марта) Третья работа.**

**Знакомство с описанием параметризованных устройств на языке SystemC.**

Задачи:

1. Создать потактовое описание FIFO, работающего с 8-разрядными словами. Глубина FIFO должна задаваться в конструкторе.

Обязательные входы:

- clk

- sreset\_n

- data\_in

- push

- pop

Обязательные выходы:

- data\_out

- empty

- full

2. Создать на основе класса FIFO шаблон класса FIFOParam, в котором параметром выступает тип данных.

3. Унаследовать от созданного шаблона класса FIFOParam, шаблон класса FIFOParamExtended с дополнительными выходами:

- almost\_empty

- almost\_full

Для всех трёх описаний необходимо написать соответствующие тесты.