Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ»

Факультет компьютерных систем и сетей

Кафедра информатики

Дисциплина: архитектура вычислительных систем

Отчет по лабораторной работе №2

# ПЛИС-устройства и разработка на языке SystemVerilog

Выполнил: студент: гр. 053506

Слуцкий Никита Сергеевич

Проверил: ст. преподаватель Шиманский Валерий Владимирович

Минск 2022

**СОДЕРЖАНИЕ**

1. Введение
2. Постановка задачи и результаты выполнения
3. Выводы
4. Литература

**Введение**

**Цели данной работы:**

1. Научиться использовать инструменты программируемых логических интегральных схем (ПЛИС/FPGA)
2. Разрабатывать и симулировать описание аппаратуры на SystemVerilog, а затем синтезировать его для ПЛИС

В ходе выполнения данной лабораторной работы будут получены навыки проектирования простых и нетривиальных логических схем, изучены основы разработки схем конечных автоматов, изучены основы и не только языка SystemVerilog, а также получены навыки работы в ПО Intel Quartus Prime и ModelSim.

**Постановка задачи и результаты выполнения**

**Задание 1: Описание инструментов**

Это задание призвано ознакомить с процессом и инструментами разработки. С инструментами ознакомился, необходимые результаты получил. Так как это необязательное и ознакомительное задание, его формулировка и результаты опускаются (но тем не менее задание было выполнено).

**Задание 2: Разработка структурного описания конечных автоматов**

**Номер в журнале 21, Вариант 1**

Задание для этой лабораторной работы заключается в разработке конечного автомата в SystemVerilog для контроля задних огней автомобиля 1965 г. Ford Thunderbird. На каждой стороне есть три огня, которые зажигаются последовательно, тем самым указывая направление поворота. На рис. 1 показаны задние огни автомобиля, а на рис. 2 показана последовательность мигания для левых огней (a) и правых огней (b).

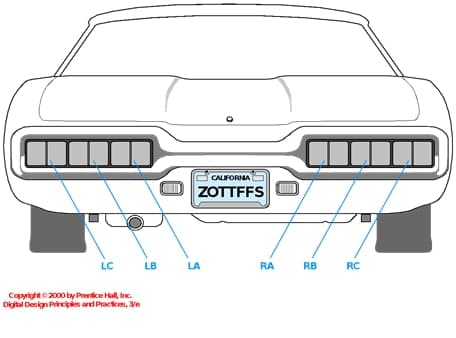


Рисунок 1

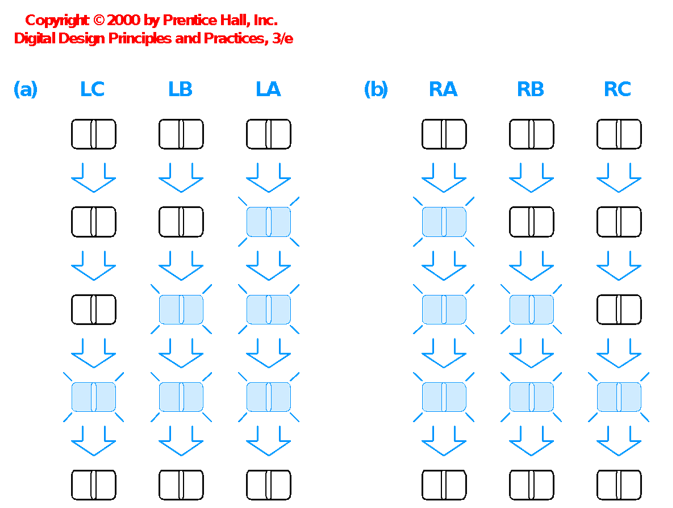


Рисунок 2

КА должен иметь два входных контакта, левый и правый, которые запускают последовательность мигания в цикле после их включения. В любой момент времени включется только один из двух входов (левый или правый). КА должен иметь шесть выходных контактов, LA, LB, LC, RA, RB и RC. После запуска последовательности она должна продолжаться, даже если сигнал на входе сброше. Когда последовательность завершится, она должна на цикл вернуться в состояние с выключенными огнями, прежде чем можно будет начать другую последовательность. Пример ожидаемого поведения см. в файле тестового вектора Thunderbird далее в этой лабораторной работе.

* Получить у преподавателя свой вариант включения поворотных сигналов. Составьте тестовые векторы.
* Составить диаграмму переходов состояний для вашего КА.
* Выбрать кодировку состояний
* Составть схему.
* Написать **структурный код SystemVerilog** для КА
* Смоделировать свой КА с помощью следующего самопроверяющегося тестового стенда и векторов. Изучить тестовый стенд и проследить, как он применяет входы и проверяет выходы

Выполнение:

В соответствии с вариантом задания №1 я построил автомат и прописал состояния в нём. Это представлено на рисунке ниже.

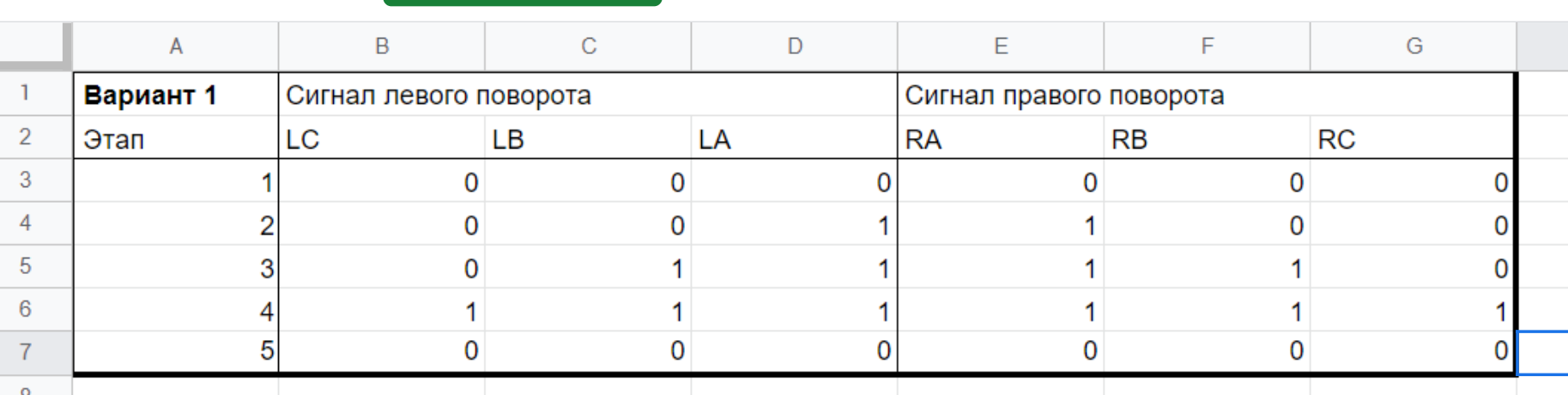


Рисунок 3

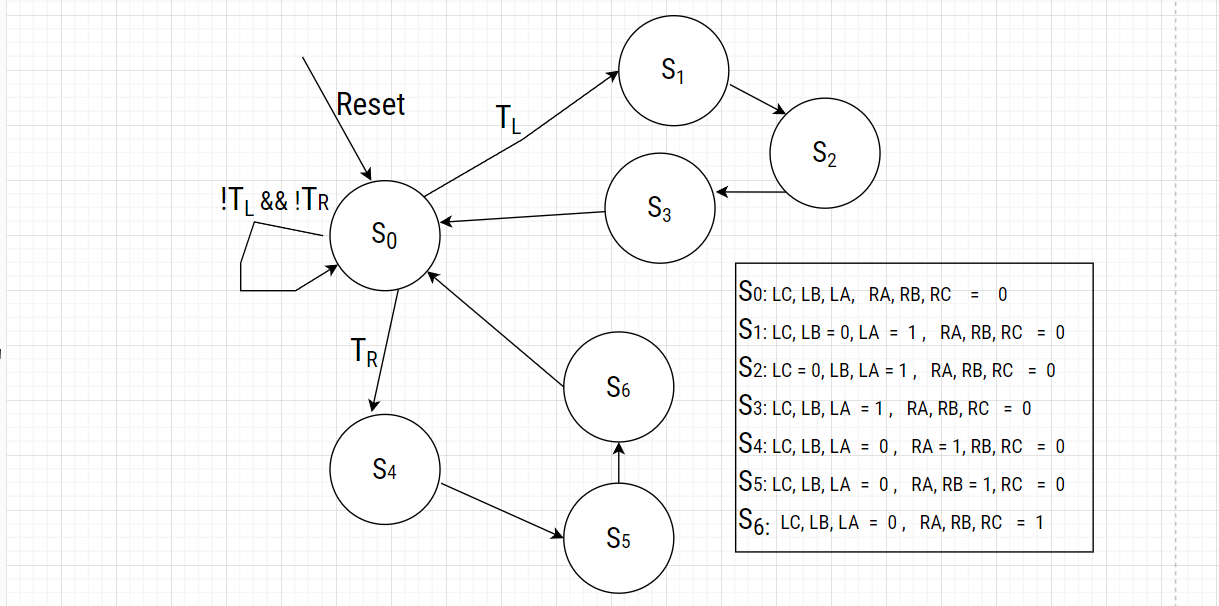
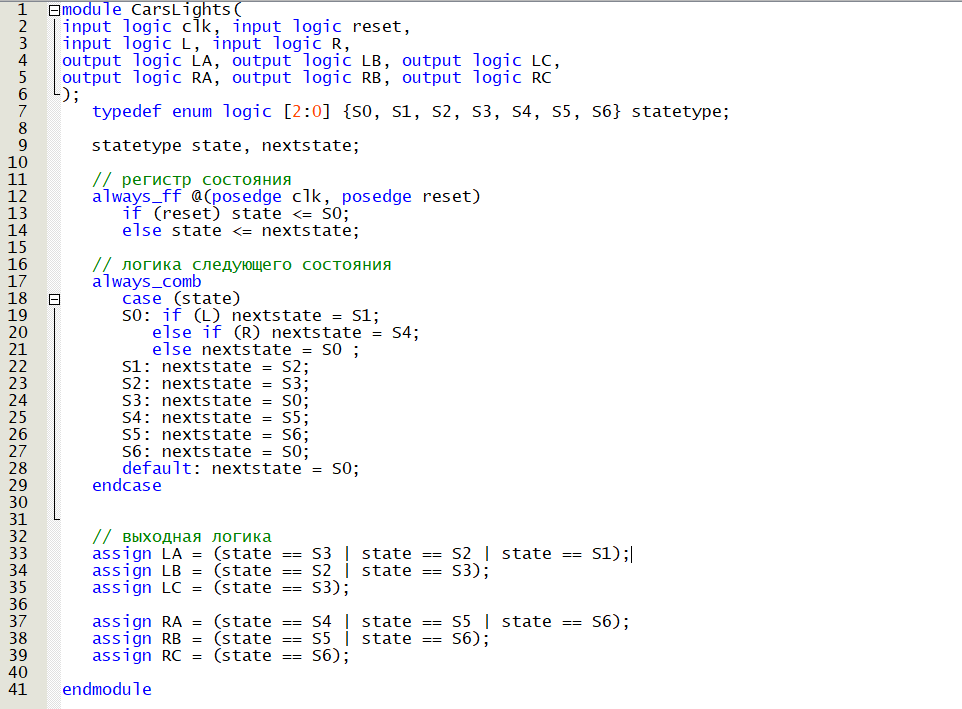


Рисунок 4

В соответствии с смоделированным на листе автоматом я написал модуль CarsLights на языке SystemVerilog:



Синтезированная в RTL Viewer программы Quartus схема имеет следующий вид:

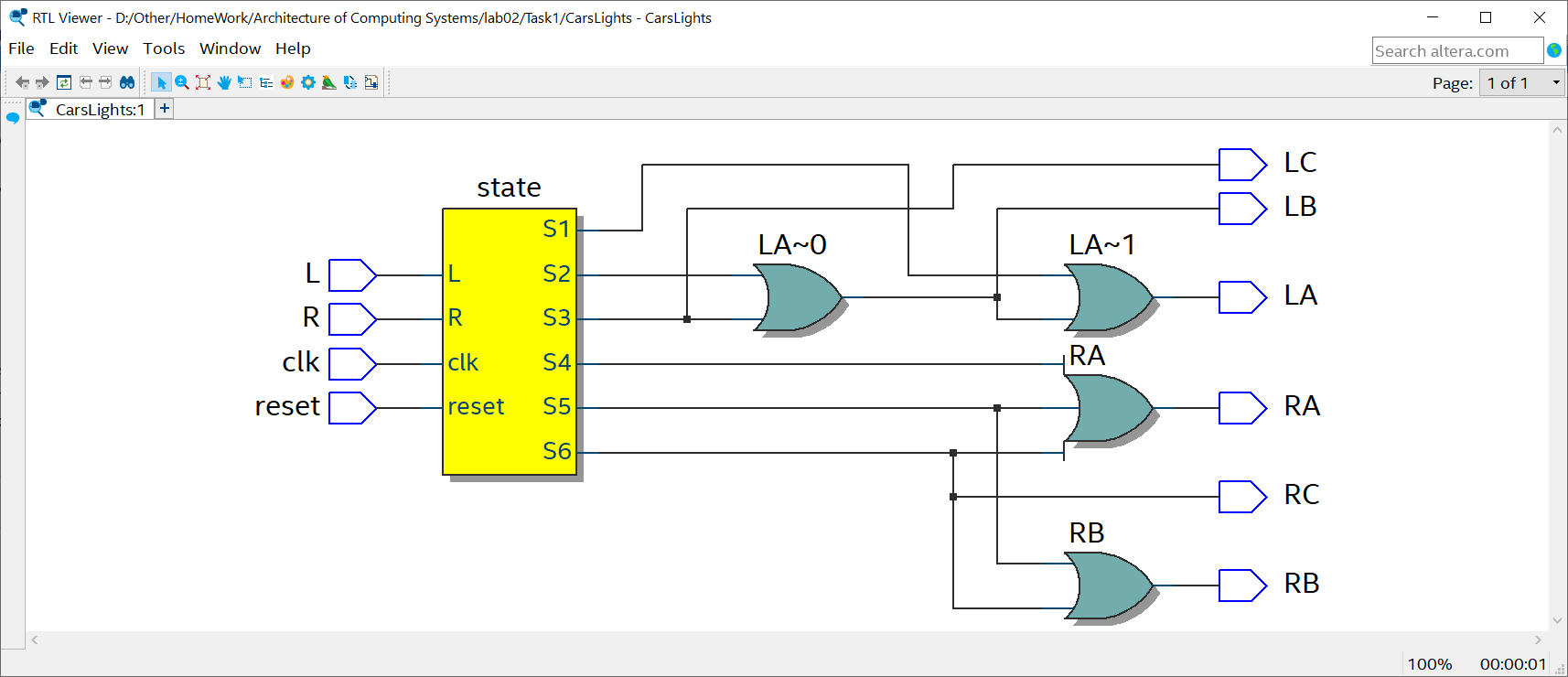


Рисунок 5

Соотношение сигналов и контактов на устройстве выполнено в соответствии с документацией от Intel. Оно представлено на скриншоте из окна Pin Planner программы Quartus:

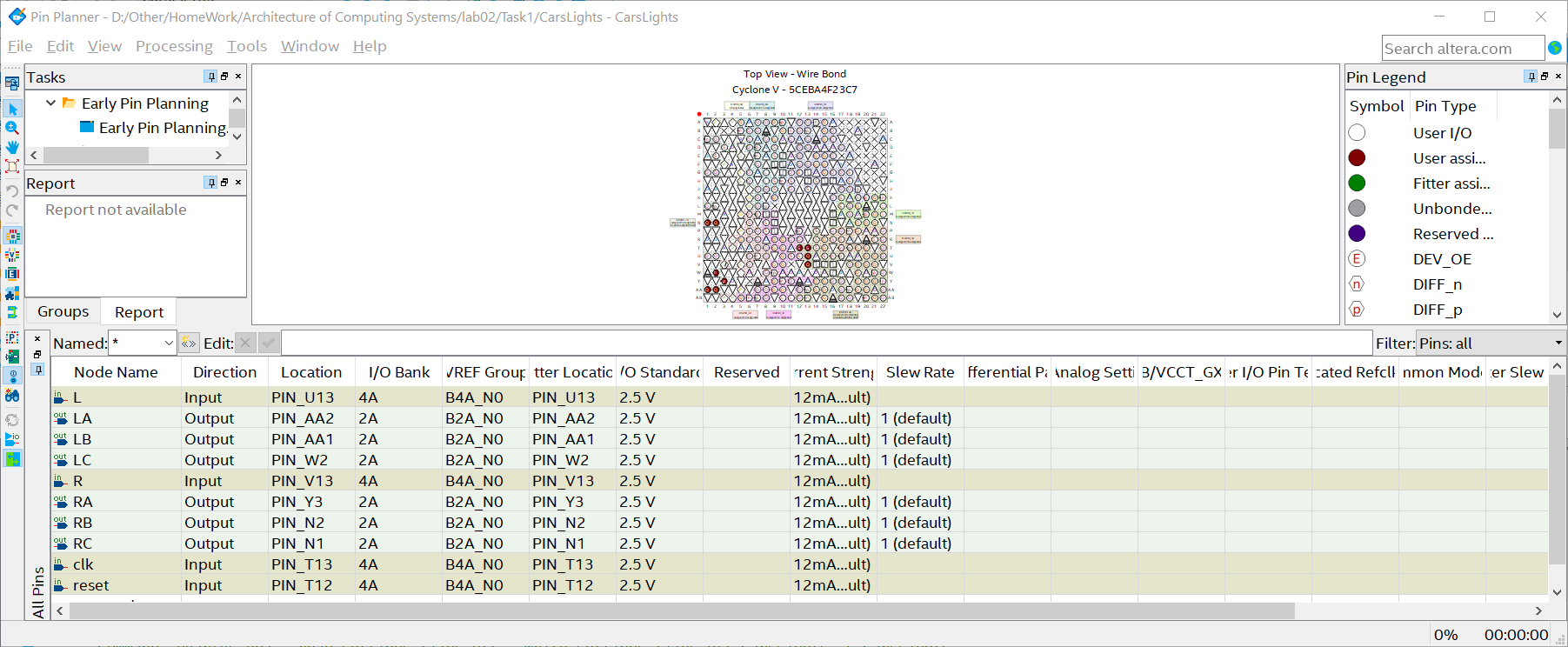
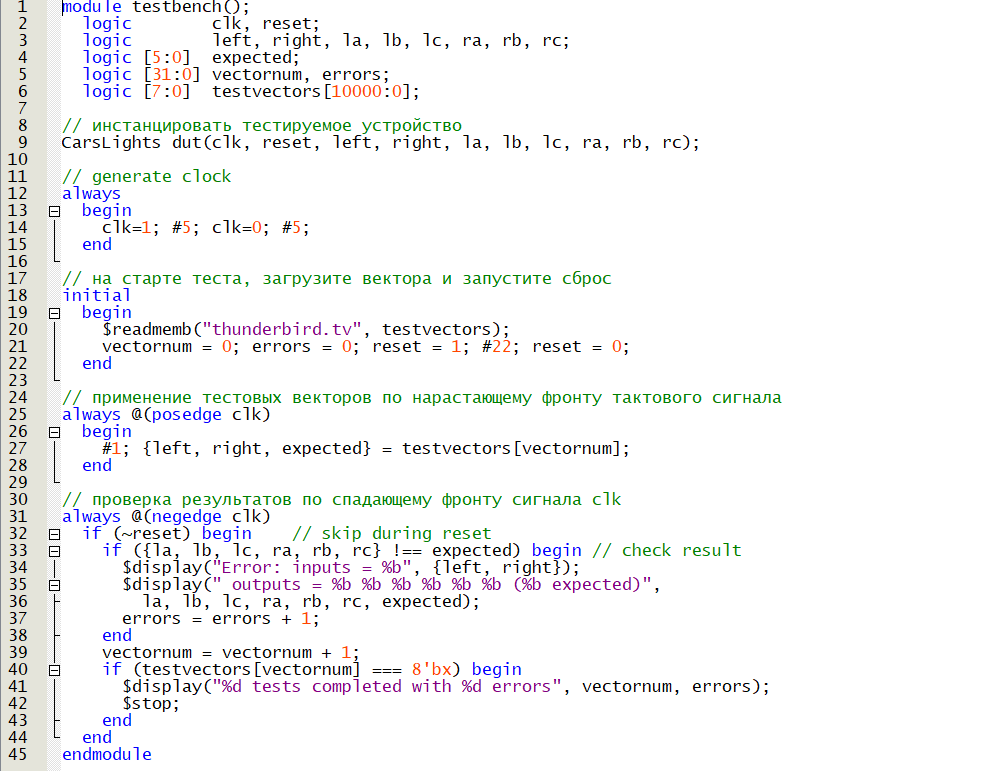


Рисунок 6

Далее я написал модуль testbench для синтеза и проверки в ModelSim.

Модуль имеет следующую реализацию:



В программе ModelSim были синтезированы файлы и проведена работа с помощью тестового стенда testbench. Скриншоты с осциллограммой и выводом про успешное прохождение тестов представлены на скриншотах ниже.

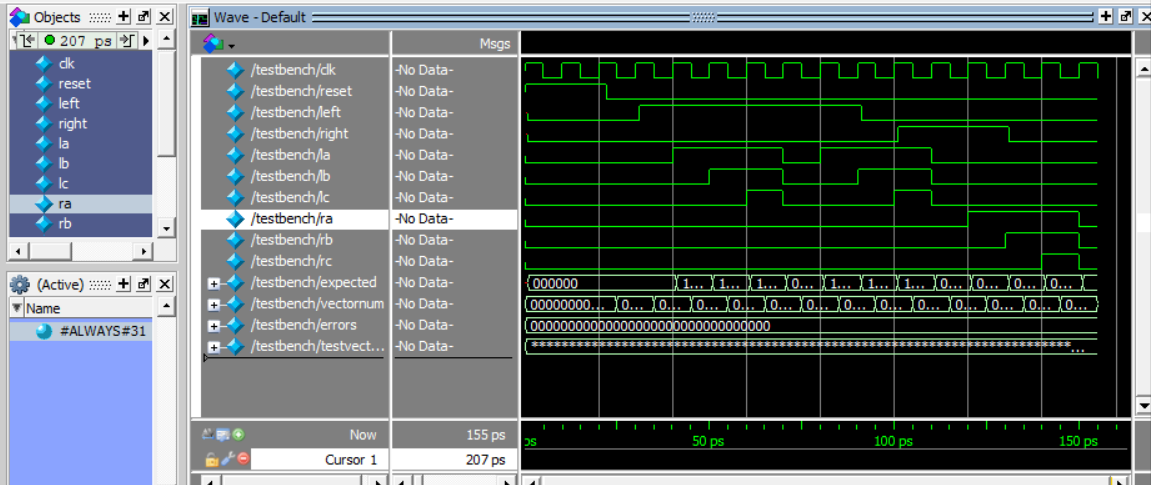


Рисунок 7

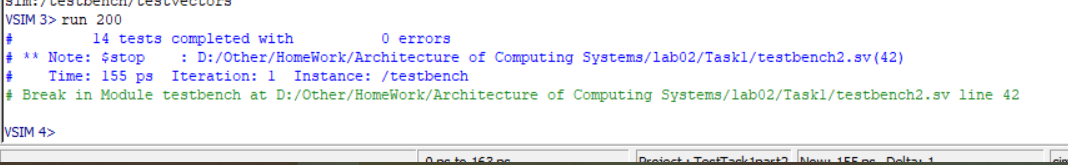


Рисунок 8

Делаю вывод, что модуль выполняет то, что от него ожидаю, что он соответствует спроектированному автомату. Благодаря RTL Viewer можно посмотреть, как реализовать модуль с использованием логических компонентов из изученной в первой лабораторной работы программы Logisim.

**Задание 2 Разработка поведенческой модели конечных автоматов**

Цель этого задания заключается в разработке конечного автомата, используя поведенческий SystemVerilog.

**Приключенческая игра**

Приключенческая игра-ĸвест, которую вы будете разрабатывать, имеет семь комнат и один объект - меч. Игра начинается в локации "Пещера ĸаĸофонии". Чтобы пройти игру, вы сначала должны последовать на восток через локацию "Извилистый туннель", а затем на юг ĸ "Стремительной реке". После чего, на западе в локации "Тайник с мечом" вы должны найти "Вострый меч". Меч позволит вам безопасно пройти на восток от "Стремительной реки" через "Логово дракона" в локацию "Победный свод", и на этом моменте вы с победой завершите игру. Если же вы пойдете через "Логово дракона" без "Вострого меча", то будете повержены опасным драконом и будете брошены на "Тёмное кладбище", где игра завершится с поражением. Игра останется на кладбище или в победном своде до тех пор, пока вы не перезапустите её. Общая карта игры показана на следующем изображении.

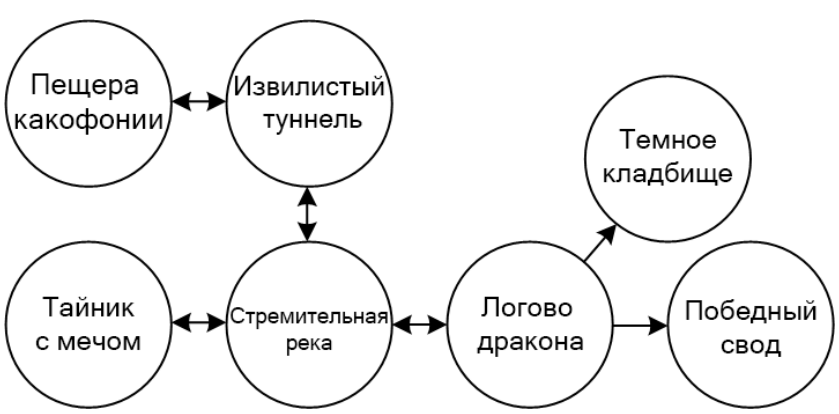
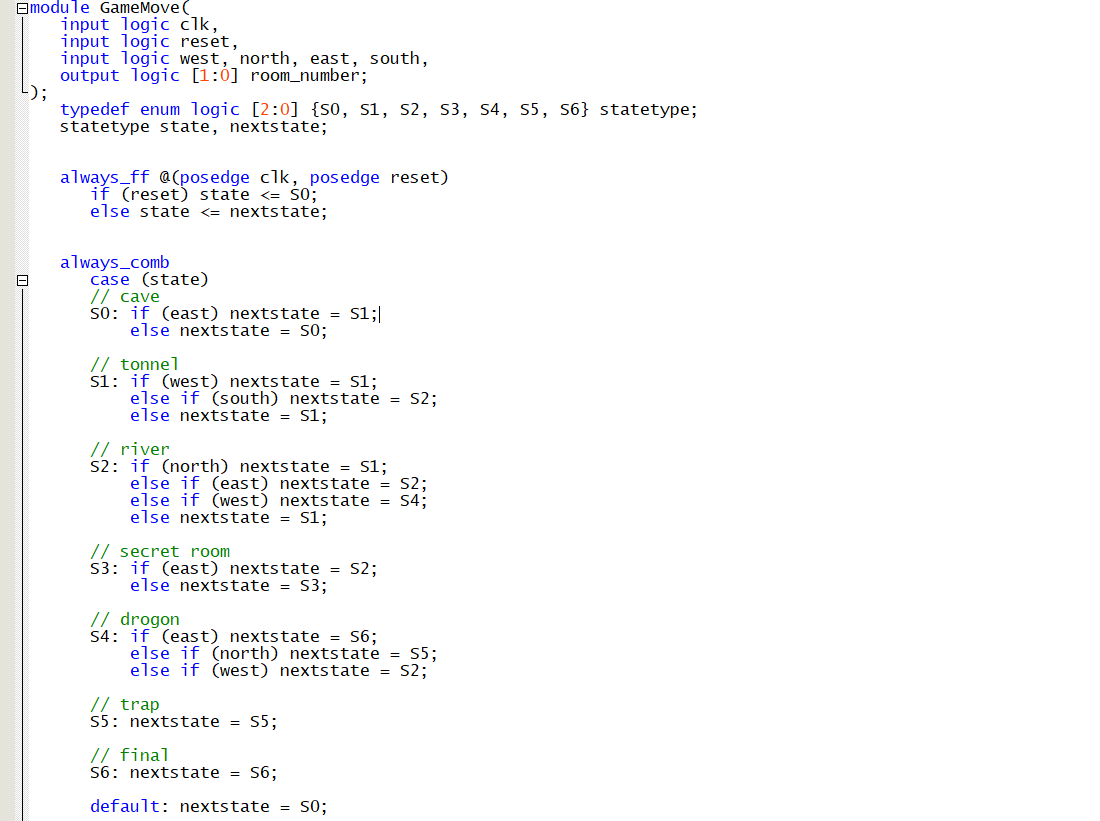
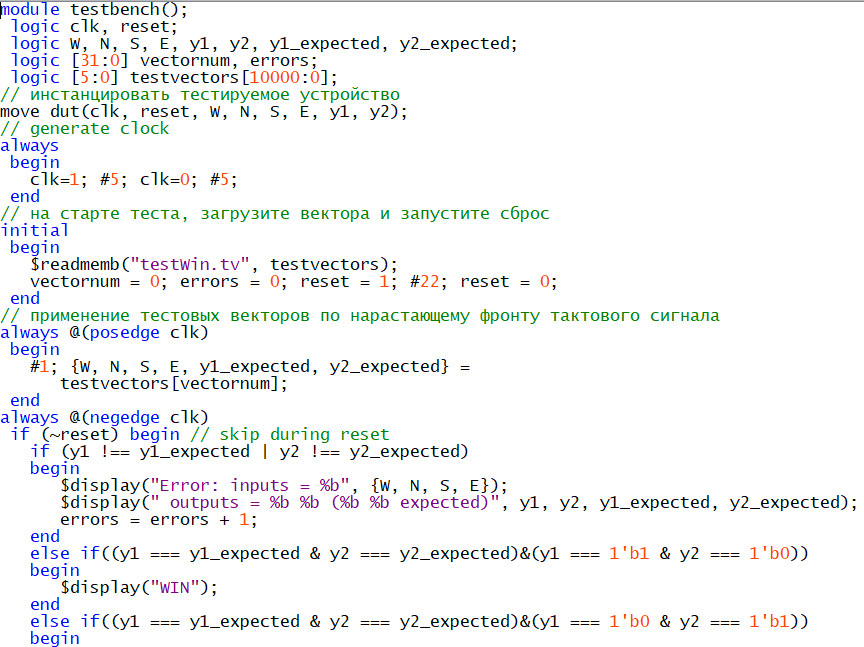


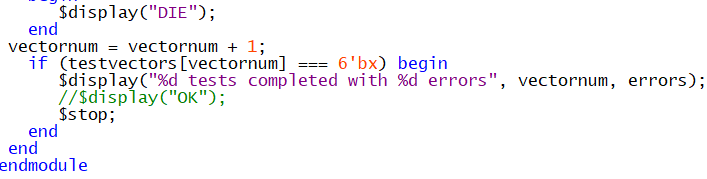
Рисунок 9

**Код конечного автомата для движения:**



**Тестовый код:**





**Выводы**

В результате выполнения лабораторной работы №2 на практике были изучены основы работы с программным продуктом Quartus и ModelSim и применены существующие знания для разработки простых и не очень логических схем. Были спроектированы автоматы для решения поставленных задач, а также уже реализованы на языке описания архитектуры SystemVerilog в среде Intel Quartus. Тесты пройдены, схемы синтезированы, реализованные модули готовы к поставленным в условиях задачам. Работа к сдаче готова.

**Литература**

Харрис, Дэвид; Харрис, Сара «Цифровая схемотехника и архитектура компьютера. RISC-V»

Джон Хопкрофт, Раджив Мотвани, Джеффри Ульман «Введение в теорию автоматов, языков, вычислений»