*Федеральное государственное бюджетное образовательное учреждение* *высшего образования*

|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | ***«Московский государственный технический университет  имени Н.Э. Баумана***  ***(национальный исследовательский университет)»***  ***(МГТУ им. Н.Э. Баумана)*** |

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

**Отчет**

**по лабораторной работе №2**

**Дисциплина:**

Основы проектирования устройств ЭВМ

**Название домашней работы:**

Проектирование цифровых устройств на основе ПЛИС

Студент гр. ИУ6-62Б  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.С. Бурлаков**

(Подпись, дата) (И.О. Фамилия)

Преподаватель  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

(Подпись, дата) (И.О. Фамилия)

Москва, 2019

# ВВедение

## Цель работы

Закрепление на практике теоретических сведений, полученных при изучении методики проектирования цифровых устройств на основе программируемых логических интегральных схем (ПЛИС), получение необходимых навыков работы с системой автоматизированного проектирования ISE WebPack 9.1 устройств на основе ПЛИС фирмы Xilinx, изучение аппаратных и программных средств моделирования, макетирования и отладки устройств на основе ПЛИС

# основная часть

**Задание 1.** Выполнить кодирование состояний автомата, представленного на рисунке 3, в соответствии с индивидуальным вариантом из таблицы 1. Для этого заполнить таблицу 2.

**Таблица 1 – Индивидуальный вариант задания**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Вариант | Набор | St0 | St1 | St2 | St3 |
| 5 | XC3S200 | 00 | 11 | 10 | 01 |

**Таблица 2 – Таблица переходов**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | St0 | St1 | St2 | St3 |
| Q1Q0 | 00 | 11 | 10 | 01 |
| CNT | 0 | 1 | 1 | 0 |
| CNT\_EN | 0 | 1 | 0 | 1 |
| CNT\_CLR | 1 | 0 | 1 | 0 |

CNT = Q1

CNT\_EN = Q0

CNT\_CLR = \_Q0;

Для определения функций D0=f(Q[20]). На ,COUNT,Q1,Q0) и D1=f(Q[20]). На ,COUNT,Q1,Q0) заполнить таблицу 3.

**Таблица 3 – Сигналы D1 и D2**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| count | Q[20] | Q1(t) | Q0(t) | D1 | D0 | Описание события |
| 0 | X | 0 | 0 | 0 | 0 | Ожидание нажатия кнопки |
| 1 | X | 0 | 0 | 1 | 1 | Нажатие кнопки |
| X | 0 | 1 | 1 | 1 | 1 | Ожидания окончания счета |
| X | 1 | 1 | 1 | 1 | 0 | Конец счета |
| 1 | X | 1 | 0 | 1 | 0 | Ожидание отпускания |
| 0 | X | 1 | 0 | 0 | 1 | Отпускание кнопки |
| x | 0 | 0 | 1 | 0 | 1 | Ожидание окончания счета |
| x | 1 | 0 | 1 | 0 | 0 | Конец счета |

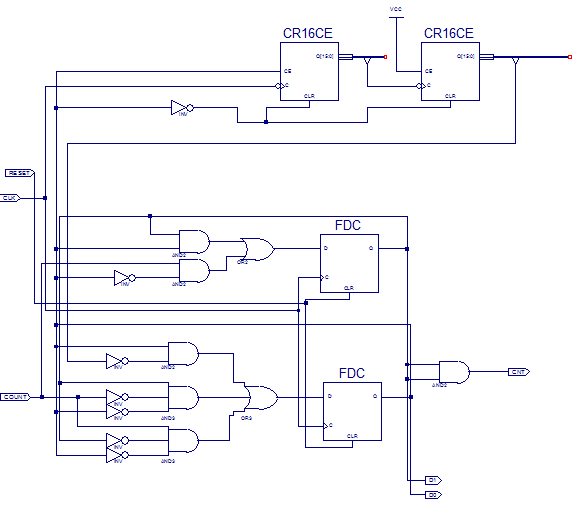
**Задание 2.** Собрать модель полученного устройства в редакторе схем САПР ISE WebPack 9.1.

Рисунок 1 – Схема устройства

**Задание 3.** В интегрированном редакторе тестов САПР ISE WebPack 9.1 разработать тест для полученного устройства и выполнить моделирование его работы в ISE Simulator.

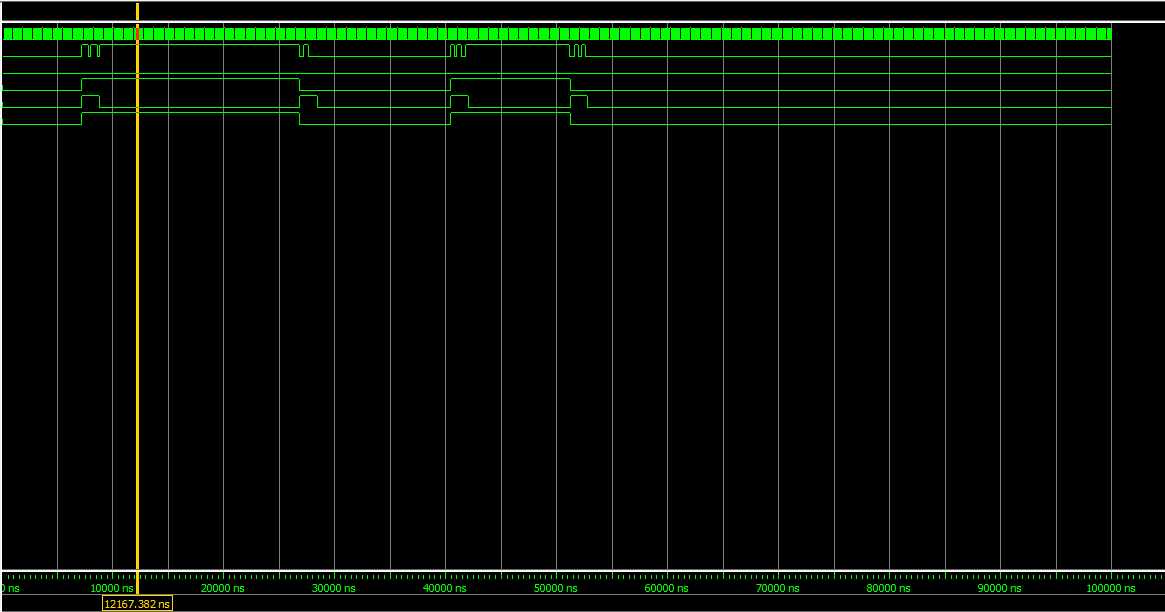
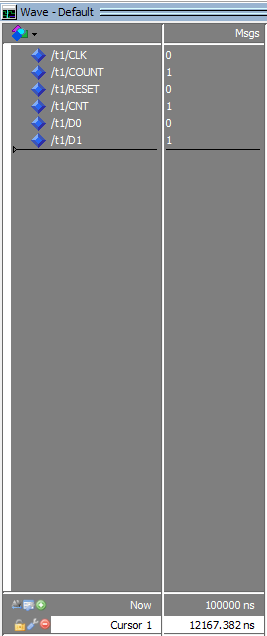


Рисунок 2 – Результаты моделирования схемы

**Задание 4.** Разработать устройство управления, принимающее 16-разрядное информационное слово Q[0..15] На и управляющее их последовательной выдачей по шине D[0..3] на декодер 7-сегментных индикаторов в соответствии с показанной на рисунке 4 диаграммой.

Листинг 1 – Устройство управления

LIBRARY ieee;

USE ieee.std\_logic\_1164.ALL;

USE ieee.std\_logic\_arith.ALL;

ENTITY Seven\_Segment\_Driver IS

PORT(

clk\_div : IN std\_logic;

Q : IN std\_logic\_vector(15 DOWNTO 0);

rst : IN std\_logic;

D : OUT std\_logic\_vector(3 DOWNTO 0);

A : INOUT std\_logic\_vector(3 DOWNTO 0));

END ENTITY Seven\_Segment\_Driver;

ARCHITECTURE Struct OF Seven\_Segment\_Driver IS

BEGIN

A\_drive: PROCESS (clk\_div,rst)

BEGIN

IF (rst = '1') THEN

A<="1110";

ELSIF (clk\_div'EVENT AND clk\_div='1') THEN

A(3)<=A(2);

A(2)<=A(1);

A(1)<=A(0);

A(0)<=A(3);

END IF;

END PROCESS A\_drive;

D(0) <= (Q(0) AND NOT(A(0)))

OR (Q(4) AND NOT(A(1)))

OR (Q(8) AND NOT(A(2)))

OR (Q(12) AND NOT(A(3)));

D(1) <= (Q(1) AND NOT(A(0)))

OR (Q(5) AND NOT(A(1)))

OR (Q(9) AND NOT(A(2)))

OR (Q(13) AND NOT(A(3)));

D(2) <= (Q(2) AND NOT(A(0)))

OR (Q(6) AND NOT(A(1)))

OR (Q(10) AND NOT(A(2)))

OR (Q(14) AND NOT(A(3)));

D(3) <= (Q(3) AND NOT(A(0)))

OR (Q(7) AND NOT(A(1)))

OR (Q(11) AND NOT(A(2)))

OR (Q(15) AND NOT(A(3)));

END ARCHITECTURE Struct;

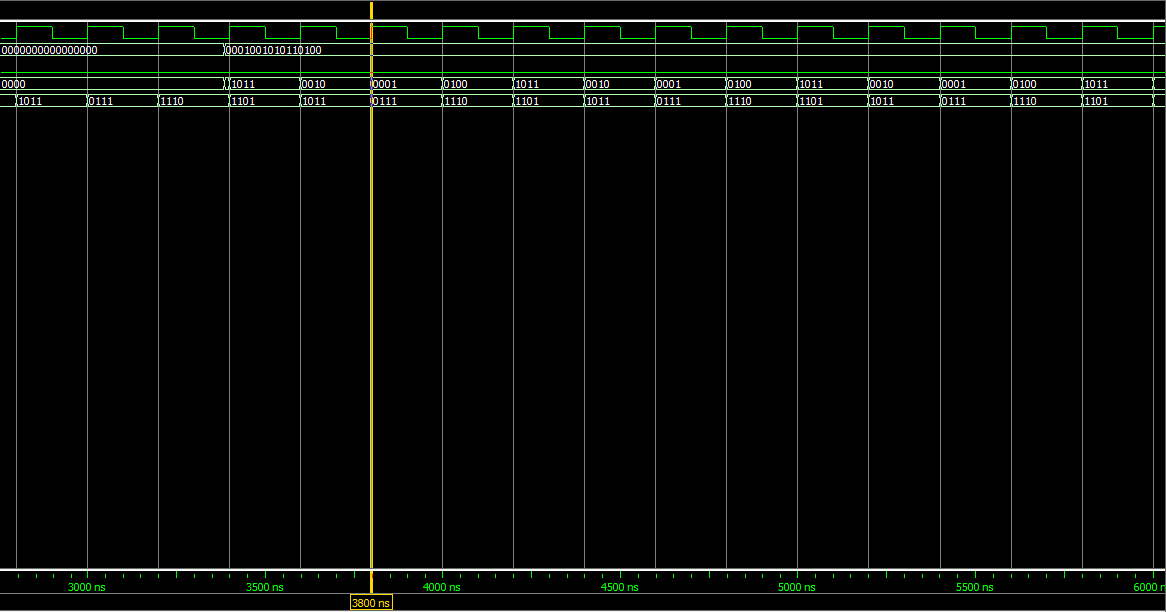
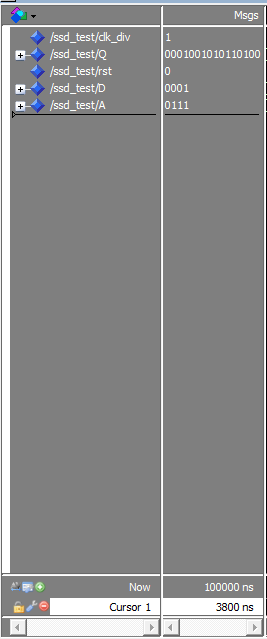


Рисунок 3 – Результаты моделирования

**Задание 5.** Разработать поведенческое VHDL описание схемы преобразования четырехразрядного информационного кода D[0..3] в код активизации 7-сегментного индикатора LED[0..7] в соответствии с таблицей 4. Подключить устройство к библиотеке компонентов текущего проекта.

LIBRARY ieee;

USE ieee.std\_logic\_1164.ALL;

USE ieee.std\_logic\_arith.ALL;

entity LEDD is

PORT(

D: IN std\_logic\_vector(3 DOWNTO 0);

LED: OUT std\_logic\_vector (7 DOWNTO 0);

)

end entity LEDD;

architecture struct of LEDD is

begin

if D ='0000' then LED <= '0000001'

elsif D = '0001' then LED <= '1001111'

elsif D = '0010' then LED <= '0010010'

elsif D = '0011' then LED <= '0000110'

elsif D = '0100' then LED <= '1001100'

elsif D = '0101' then LED <= '0100100'

elsif D = '0110' then LED <= '0100000'

elsif D = '0111' then LED <= '0001111'

elsif D = '1000' then LED <= '0000000'

elsif D = '1001' then LED <= '0000100'

elsif D = '1010' then LED <= '0001000'

elsif D = '1011' then LED <= '1100000'

elsif D = '1100' then LED <= '0110001'

elsif D = '1101' then LED <= '1000010'

elsif D = '1110' then LED <= '0110000'

elsif D = '1111' then LED <= '0111000';

end architecture struct;

**Задание 6.** В редакторе схем САПР ISE WebPack 9.1. собрать модель полученного устройства, включающего схему подавления дребезга, схему управления 7-сегментными индикаторами, 16-разрядный счетчик и схему деления частоты для активизации 7- сегментных индикаторов (библиотечные элементы CB16CE и BUFGCE).

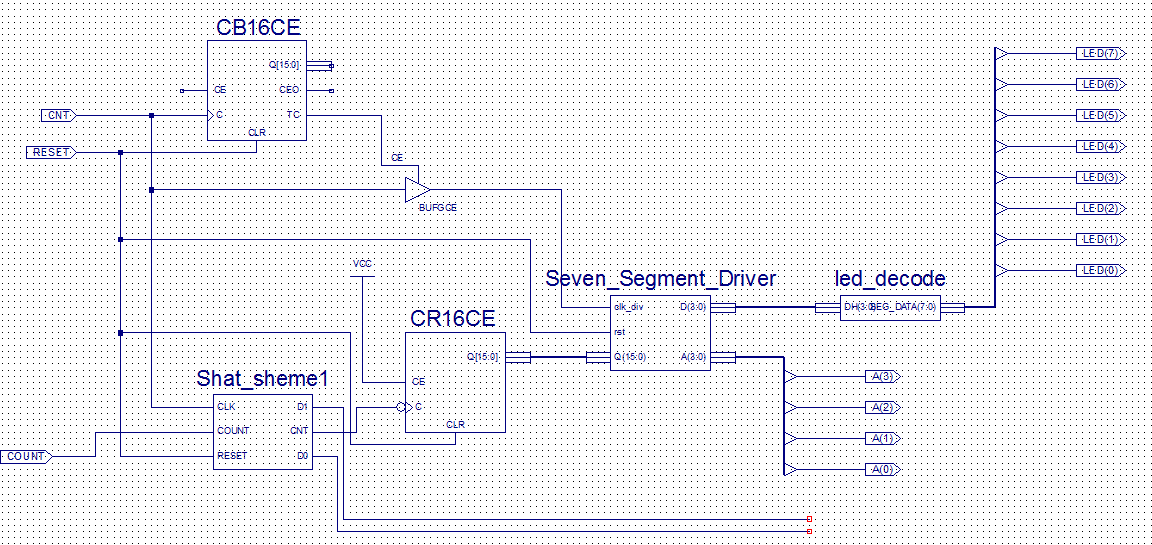


Рисунок 4 – Схема основного модуля

**Задание 7.** В САПР ISE WebPack 9.1 выполнить автоматический синтез технологической схемы, размещение и трассировку полученного устройства на кристалле Spartan3 XC3S200 ft256 (или для Spartan 3E-500 FG320), генерировать файл конфигурации ПЛИС (\*.bin)

-----------------

All values displayed in nanoseconds (ns)

Setup/Hold to clock CNT

------------+------------+------------+------------------+--------+

| Setup to | Hold to | | Clock |

Source | clk (edge) | clk (edge) |Internal Clock(s) | Phase |

------------+------------+------------+------------------+--------+

COUNT | 0.449(R)| 1.255(R)|CNT\_IBUFG | 0.000|

------------+------------+------------+------------------+--------+

Clock CNT to Pad

------------+------------+------------------+--------+

| clk (edge) | | Clock |

Destination | to PAD |Internal Clock(s) | Phase |

------------+------------+------------------+--------+

A<0> | 8.882(R)|XLXN\_14 | 0.000|

A<1> | 8.553(R)|XLXN\_14 | 0.000|

A<2> | 8.704(R)|XLXN\_14 | 0.000|

A<3> | 8.799(R)|XLXN\_14 | 0.000|

D0 | 9.120(R)|CNT\_IBUFG | 0.000|

D1 | 8.817(R)|CNT\_IBUFG | 0.000|

LED<0> | 12.715(R)|XLXN\_14 | 0.000|

LED<1> | 12.944(R)|XLXN\_14 | 0.000|

LED<2> | 12.000(R)|XLXN\_14 | 0.000|

LED<3> | 12.173(R)|XLXN\_14 | 0.000|

LED<4> | 12.075(R)|XLXN\_14 | 0.000|

LED<5> | 13.085(R)|XLXN\_14 | 0.000|

LED<6> | 13.028(R)|XLXN\_14 | 0.000|

------------+------------+------------------+--------+

Clock to Setup on destination clock CNT

---------------+---------+---------+---------+---------+

| Src:Rise| Src:Fall| Src:Rise| Src:Fall|

Source Clock |Dest:Rise|Dest:Rise|Dest:Fall|Dest:Fall|

---------------+---------+---------+---------+---------+

CNT | 1.836| | 1.844| 1.494|

---------------+---------+---------+---------+---------+

Analysis completed Sat Mar 30 17:57:05 2019

# Заключение

На практике закреплены теоретические сведения, полученные при изучении методики проектирования цифровых устройств на основе программируемых логических интегральных схем (ПЛИС), получены необходимые навыки работы с системой автоматизированного проектирования ISE WebPack 9.1 устройств на основе ПЛИС фирмы Xilinx, изучение аппаратных и программных средств моделирования, макетирования и отладки устройств на основе ПЛИС.