**Министерство науки и высшего образования Российской Федерации**

федеральное государственное автономное образовательное учреждение   
высшего образования

«Национальный исследовательский Томский политехнический университет»

|  |  |
| --- | --- |
| Школа / филиал | Инженерная школа неразрушающего контроля и безопасности |
| Обеспечивающее подразделение | Отделение электронной инженерии |
| Направление подготовки | 11.03.04 «Электроника и наноэлектроника» |
| Образовательная программа | Прикладная электронная инженерия |

**ОТЧЕТ**

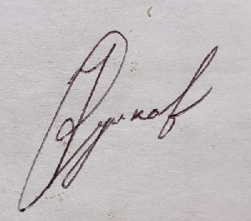
**ПО ЛАБОРАТОРНОЙ РАБОТЕ №1**

по дисциплине «Основы ПЛИС»



Выполнил:

студент гр. 1А22 17.10.2024 \_\_\_\_\_\_\_\_\_\_\_\_\_ Кравцов О.К.



Проверил:

ассистент ОЭИ 17.10.2024 \_\_\_\_\_\_\_\_\_\_\_\_\_ Сизиков Ф.А.

# ЦЕЛЬ РАБОТЫ

Изучить основные инструменты САПР *Vivado*, научиться моделировать в ней работу схем на основе простых логических элементов и познакомиться с платой *Basys 3*.

# ПОСТАНОВКА ЗАДАЧИ

1. Изучить методические рекомендации, ознакомиться с интерфейсом и особенностями САПР *Vivado*.
2. Выбрать логическое выражение согласно варианту и произвольно расставить скобки в нём.
3. Составить таблицу истинности для полученной логической функции.
4. Написать код её реализации на *HDL* (согласно варианту).
5. Написать тестовый файл на том же *HDL* и промоделировать работу кода в двух режимах: «*Run Post-Implementation Functional Simulation*» и «*Run Post-Implementation Timing Simulation*».
6. Сравнить результаты моделирования с составленной таблицей истинности, при необходимости исправить допущенные ошибки.
7. Назначить входам и выходам ножки, к которым на плате *Basys 3* к ПЛИС подключены движковые переключатели и светодиод (согласно варианту).
8. Продемонстрировать результаты моделирования и назначенные ножки преподавателю.
9. Проверить работу написанного кода на плате *Basys 3*, соответствие результатов с составленной в п.3 таблицей истинности.
10. Продемонстрировать работу преподавателю.
11. Сделать несколько фотографий работы кода на плате.

# ТАБЛИЦА ИСТИННОСТИ

В соответствии с 11 вариантом была взята логическая функция четырёх переменных:

В ней были расставлены скобки случайным образом (по своему усмотрению):

Для полученной функции (2) была составлена таблица истинности (табл. 1)

Таблица 1 – Таблица истинности

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Входы** | | | | **Выход** |
| **A** | **B** | **C** | **D** | **X** |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 |

# КОД НА HDL

Согласно заданию 11 варианта необходимо было описать реализацию функции (2) на *HDL* *SystemVerilog.*

Листинг программы на языке *SystemVerilog:*

module logic\_function (  
 input logic A,  
 input logic B,  
 input logic C,  
 input logic D,  
 output logic X  
);  
  
 always\_comb begin  
 X = (A ^ B) & (C | D);  
 end  
  
endmodule

# ФУНКЦИОНАЛЬНАЯ СХЕМА

На рисунке 1 представлена схема, синтезированная *Vivado* на основе кода. Она полностью соответствует описываемой функции и коду: имеет 4 входа (*A-D*), 3 логических вентиля (*AND, OR, XOR*) и один выход – *X*.

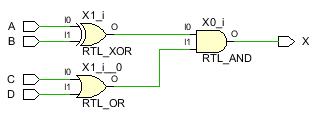


Рисунок 1 – Функциональная схема

# МОДЕЛИРОВАНИЕ

Моделирование кода даёт возможность посмотреть работу заданной лог. функции, не используя плату. Для проведения симуляции был написан тестовый файл.

Листинг тестового файла на языке *SystemVerilog:*

`timescale 1ns / 1ps  
  
module SIM\_VOLK();  
 logic [3:0] ABCD\_reg;  
 logic X;  
 initial begin  
 ABCD\_reg <= 4'd0;  
 end  
 always begin  
 #50 ABCD\_reg += 4'b1;  
 end  
 VOLK test(.A(ABCD\_reg[3]), .B(ABCD\_reg[2]), .C(ABCD\_reg[1]), .D(ABCD\_reg[0]), .X(X));  
endmodule

На рисунках 2 и 3 представлены скриншоты симуляций в двух разных режимах. Видно, что полученные сфазированные временные диаграммы состояний входов и выходов полностью соответствуют таблице истинности, что свидетельствует о верно написанном коде.

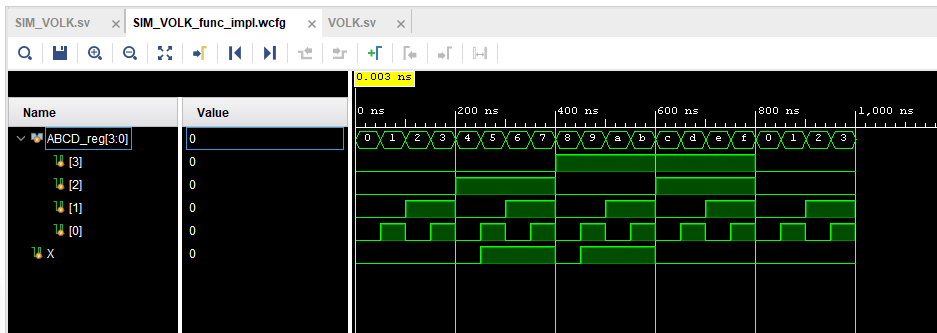


Рисунок 2 – Моделирование поведения модуля без учёта реальных задержек в кристалле

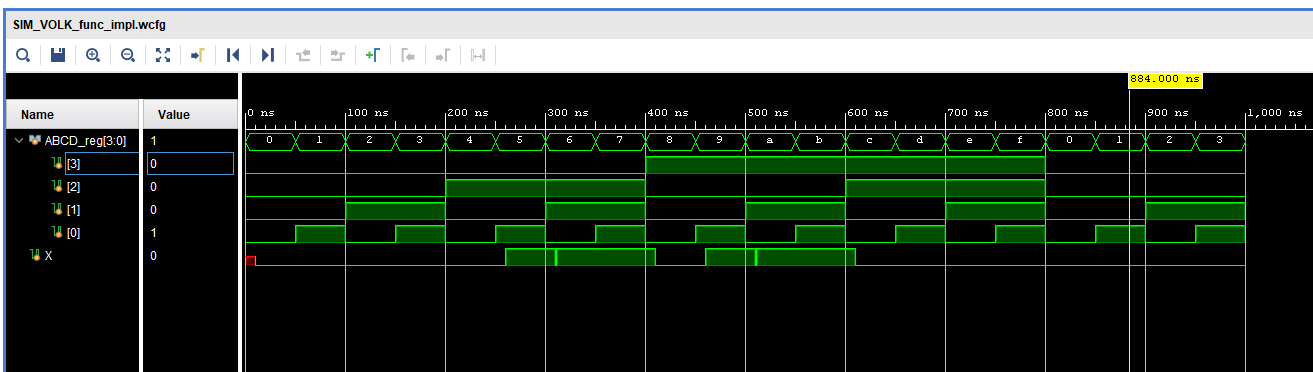


Рисунок 3 – Моделирование поведения модуля с учётом реальных задержек в кристалле

# НАзначение ножек плис

С помощью вкладки «*I/O Ports»* были описаны используемые порты ввода-вывода. Для всех пинов был выбран стандарт «*LVCMOS 3.3*» - КМОП 3,3 В. Входы модуля были подключены к пинам R2, T1, U1, W1, соответствующие крайним левым движковым переключателям. Выход модуля был подан на пин *U16*, к котором на плате подключен крайний правый светодиод *LD0.*

Содержимое файла «*my-xdc.xdc*»:

set\_property IOSTANDARD LVCMOS33 [get\_ports A]  
set\_property IOSTANDARD LVCMOS33 [get\_ports B]  
set\_property IOSTANDARD LVCMOS33 [get\_ports C]  
set\_property IOSTANDARD LVCMOS33 [get\_ports D]  
set\_property IOSTANDARD LVCMOS33 [get\_ports X]  
set\_property PACKAGE\_PIN U16 [get\_ports X]  
set\_property PACKAGE\_PIN R2 [get\_ports A]  
set\_property PACKAGE\_PIN T1 [get\_ports B]  
set\_property PACKAGE\_PIN U1 [get\_ports C]  
set\_property PACKAGE\_PIN W2 [get\_ports D]

# Фотографии макета

На рисунках 4-6 представлены фотографии макета при различном положении переключателей.

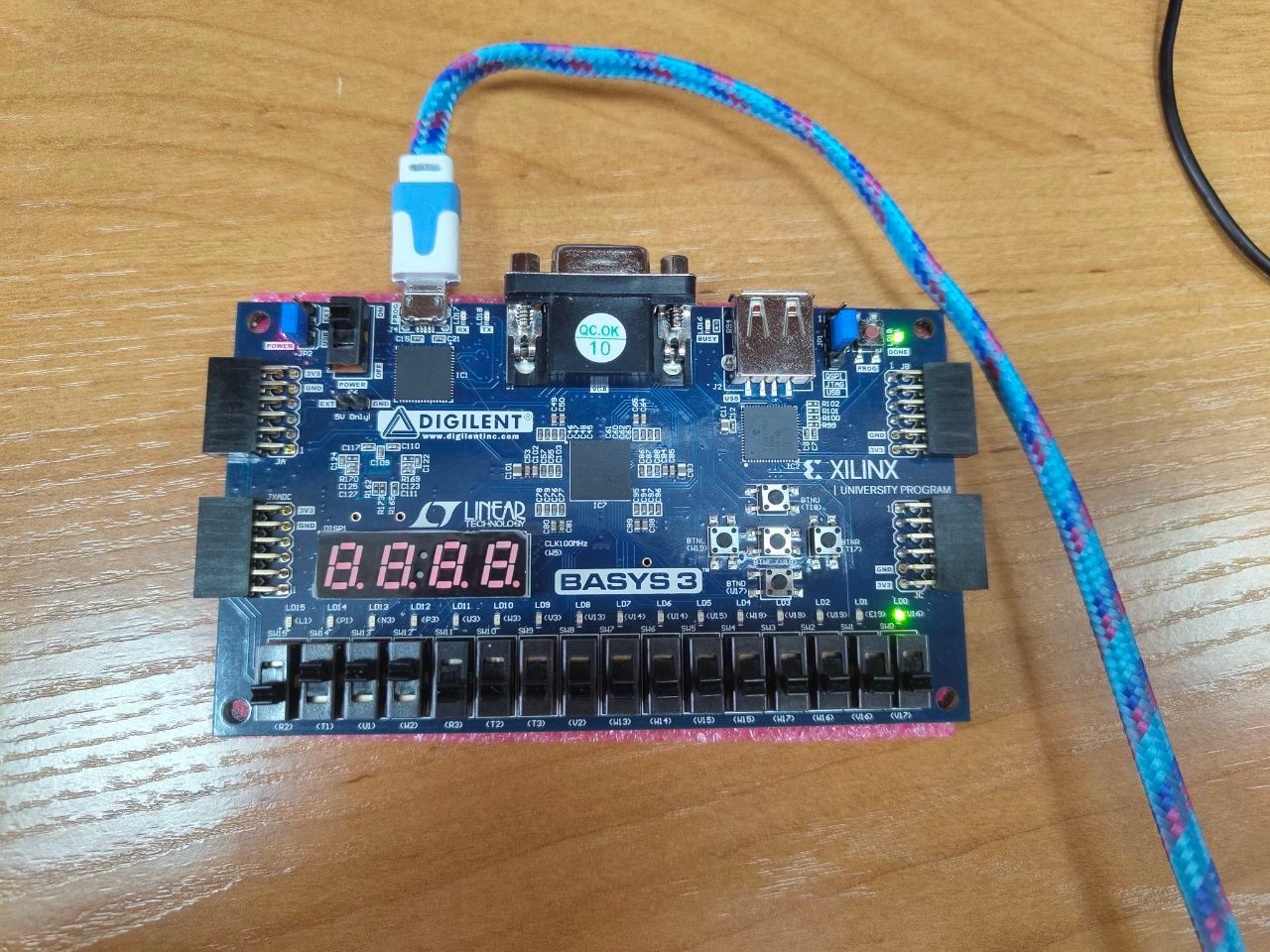


Рисунок 4 – Фотография макета: A = 0, B = 1, C = 1, D = 1

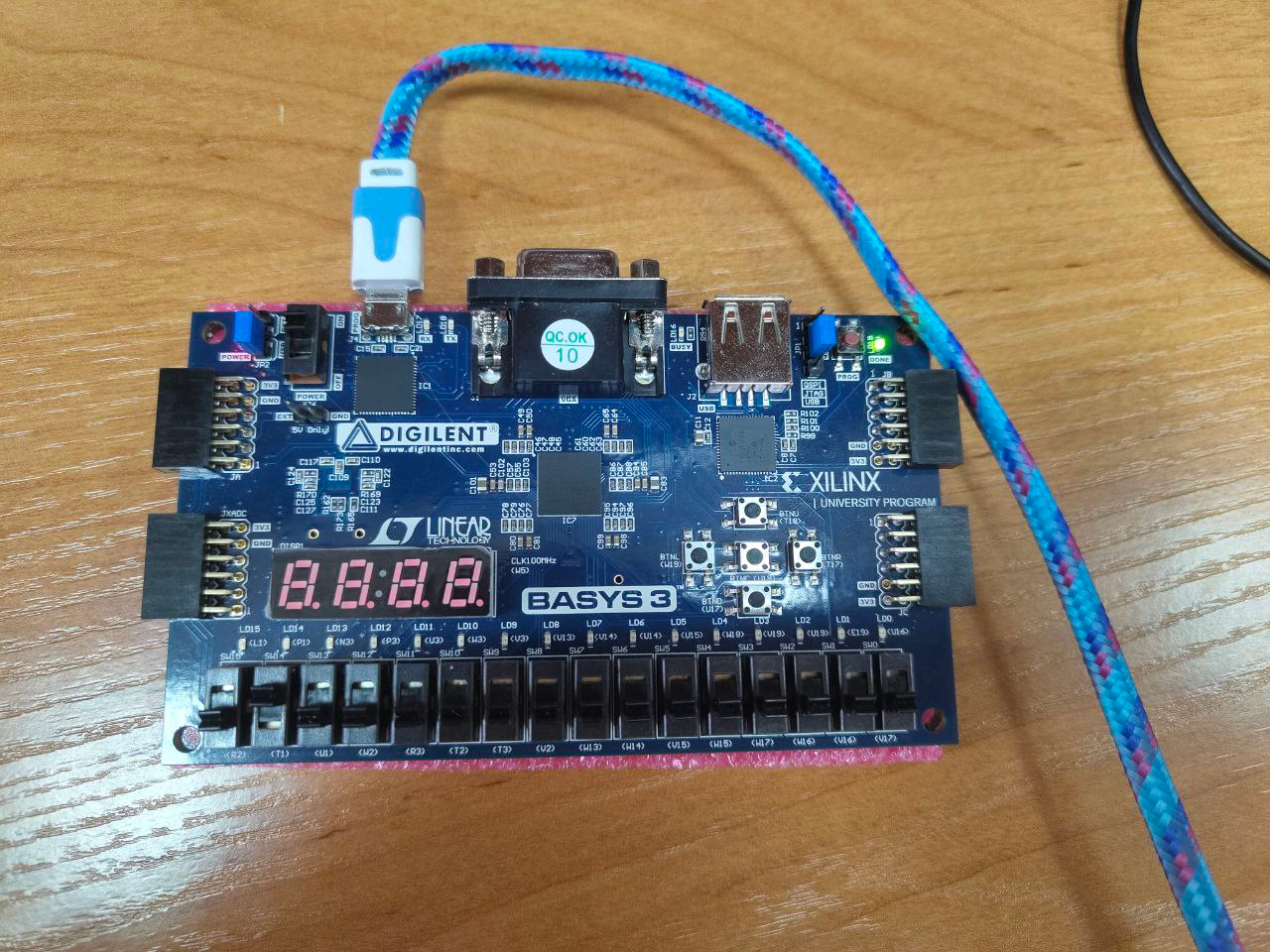


Рисунок 5 – Фотография макета: A = 0, B = 1, C = 0, D = 0

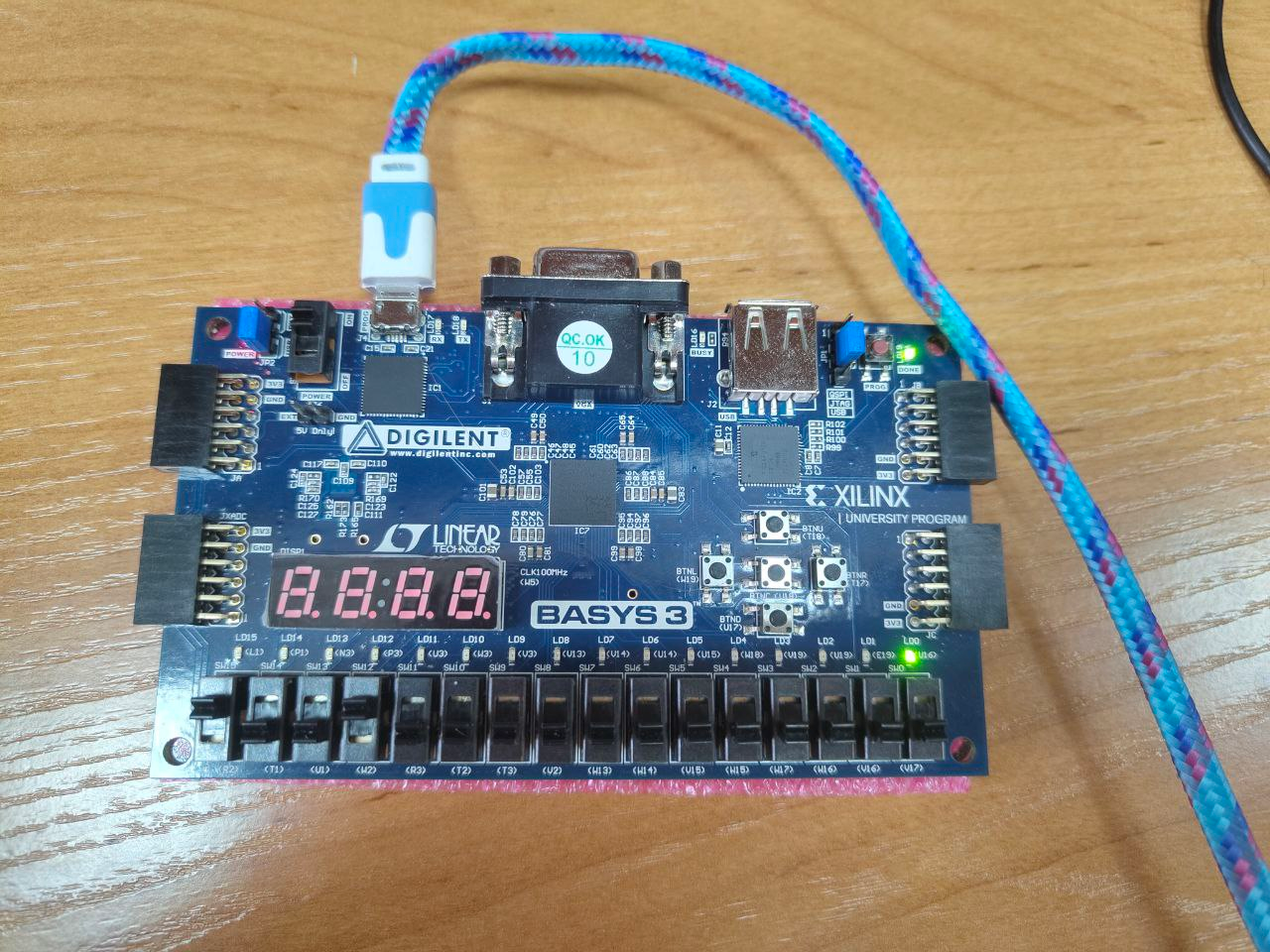


Рисунок 6 – Фотография макета: A = 1, B = 0, C = 0, D = 1

# Выводы

Была выбрана и проанализирована логическая функция согласно варианту задания. Произвольная расстановка скобок в этой функции привела к получению уникального выражения, для которого была составлена таблица истинности.

Реализация функции на языке описания аппаратуры *SystemVerilog* позволила приобрести практический опыт написания кода для ПЛИС. Проведение моделирования в двух режимах — «*Run Post-Implementation Functional Simulation*» и «*Run Post-Implementation Timing Simulation*» — подтвердило корректность работы разработанной схемы и соответствие результатов составленной таблице истинности.

Последующее тестирование на плате *Basys 3* дало возможность на практике убедиться в правильности работы схемы. Фотографии макета при различных комбинациях входных сигналов наглядно демонстрируют соответствие полученных результатов теоретическим ожиданиям.

В процессе работы были приобретены ценные навыки: от написания HDL-кода и проведения симуляций до практической реализации и отладки схем на аппаратуре. Это заложило прочную основу для дальнейшего изучения программируемых логических интегральных схем и разработки более сложных цифровых устройств.