МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«Вятский государственный университет»**

Факультет автоматики и вычислительной техники

Кафедра электронных вычислительных машин

**Проектирование управляющего устройства на базе ПЛИС Altera**

Отчет по лабораторной работе № 2

дисциплины «Системы автоматизированного проектирования»

Выполнили студенты группы ИВТ-41\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /Кудяшев Я.Ю./

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Кашина В.С/

Проверил доцент кафедры ЭВМ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /Скворцов А.А./

Киров 2023

1 Описание задачи

При включении питания автомат находится в режиме выбора товара. На индикаторе «Выберите товар». После нажатия кнопки «8» автомат ожидает оплаты за воду. Пять литров воды стоит 15 руб. Кнопка «7» соответствует внесению в автомат 1 руб, «8» – 5 руб, «9» – 10 руб. Как только накопится не менее 15 руб., автомат выдает сообщение «Вставьте тару». После нажатия кнопки «8» автомат выдает сообщение «Нажмите Налить». После нажатия кнопки «#» происходит наливание воды в тару в течение 10 секунд, которое имитирует горящий красный светодиод. В это время автомат не реагирует на нажатие кнопок. По истечению этого времени красный светодиод гаснет, на индикаторе сообщение «Заберите тару». После нажатия кнопки «8» автомат переходит в первоначальный режим – режим выбора товара, на индикаторе «Выберите товар». Кнопка «0» имитирует кнопку «Отмена» автомата розлива воды. Ее нажатие отменяет все произведенные действия и переводит автомат в первоначальный режим выбора товара.

2 Структурная схема управляющего устройства

Структурная схема управляющего устройства представлена на рисунке 1.



Рисунок 1 – Структурная схема управляющего устройства

3 Функциональная схема управляющего устройства

Функциональная схема управляющего устройства представлена на рисунке 2.

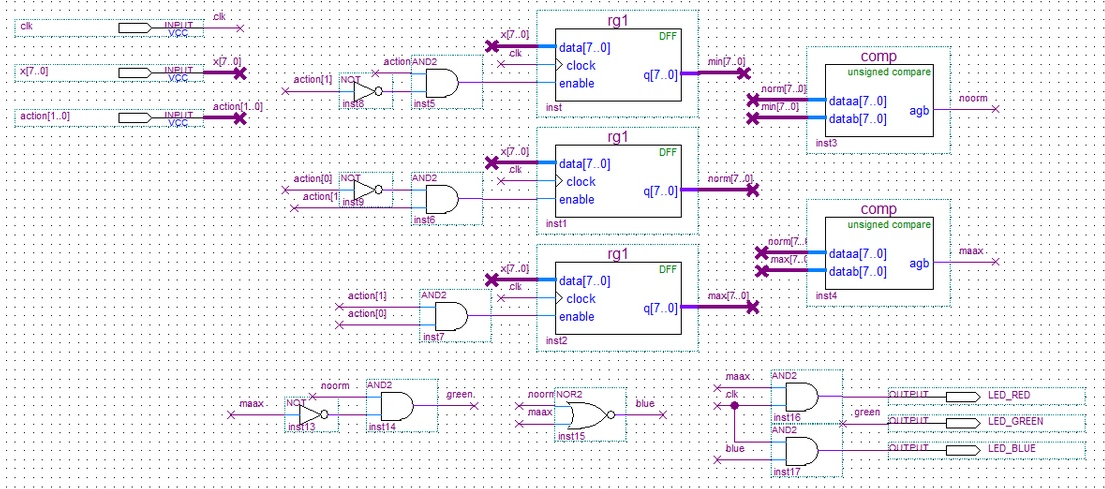


Рисунок 2 – Функциональная схема управляющего устройства

Обозначения входных и выходных элементов, представленных на функциональной схеме.

* х[7..0] – входная шина данных;
* action[1..0] – входная управляющая шина (00 – нет операции, 01 – запись нижней границы, 11 – запись верхней границы);
* clk – входной сигнал синхронизации;
* LED\_RED – выходной сигнал красного светодиода;
* \_GREEN – выходной сигнал зелёного светодиода;
* LED\_BLUE – выходной сигнал синего светодиода;

4 Временные диаграммы

Временные диаграммы, подтверждающие корректность работы устройства представлена на рисунке 3.

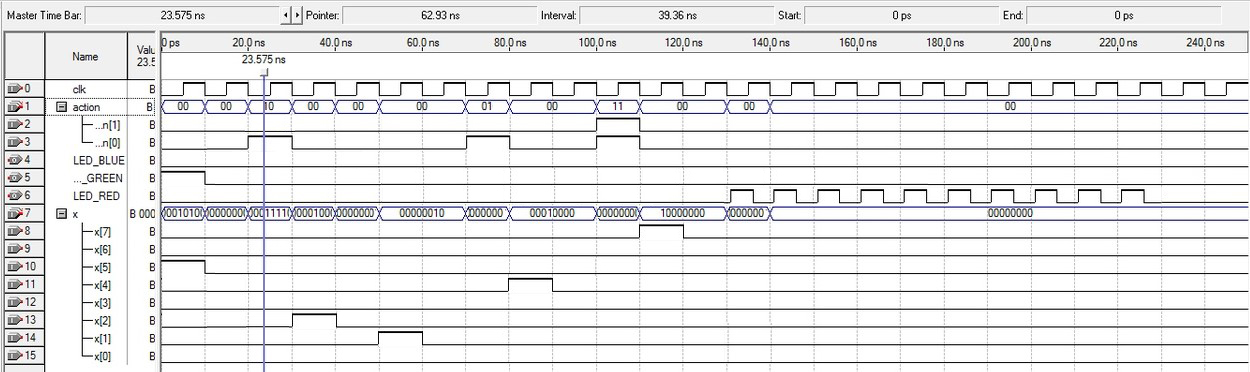


Рисунок 3 – Временные диаграммы

Сигналы n1 и n0 служат для перехода из одного состояния в другое. Сигналы x0-x7 активируют функции клавиш. Программа начинается с состояния 00 и инкрементируется до 11, после чего сбрасывается в начальное состояние 00. По приходу нового состояния на n, по x устанавливается в 1 в следующем такте.

Состояние 00 является начальным, в этом состоянии производится выбор товара, после подачи сигнала на x5 происходит переход на следующее состояние. В состоянии 01 производится оплата товара, по сигналам x1, x2, x3. После оплаты происходит переход в состояние 10, в котором необходимо вставить тару. Далее, при подтверждении по сигналу x4 меняется состояние, в котором необходимо подтвердить пуск воды по сигналу x7. Если приходит сигнал 1 на x7, то происходит переключение красного светодиода в режим мигания, на счётчике отсчитывается 10 тактов, после чего происходит переход в начальное состояние.

5 Листинг программы

Листинг программы представлен в приложении А.

Приложение А

(обязательное)

Листинг программы

-- PROGRAM "Quartus II"

-- VERSION "Version 9.1 Build 350 03/24/2010 Service Pack 2 SJ Web Edition"

-- CREATED "Thu Feb 28 18:14:41 2019"

LIBRARY ieee;

USE ieee.std\_logic\_1164.all;

LIBRARY work;

ENTITY lab IS

PORT

(

clk : IN STD\_LOGIC;

action : IN STD\_LOGIC\_VECTOR(1 DOWNTO 0);

x : IN STD\_LOGIC\_VECTOR(7 DOWNTO 0);

LED\_RED : OUT STD\_LOGIC;

LED\_GREEN : OUT STD\_LOGIC;

LED\_BLUE : OUT STD\_LOGIC

);

END lab;

ARCHITECTURE bdf\_type OF lab IS

COMPONENT rg1

PORT(clock : IN STD\_LOGIC;

enable : IN STD\_LOGIC;

data : IN STD\_LOGIC\_VECTOR(7 DOWNTO 0);

q : OUT STD\_LOGIC\_VECTOR(7 DOWNTO 0)

);

END COMPONENT;

COMPONENT comp

PORT(dataa : IN STD\_LOGIC\_VECTOR(7 DOWNTO 0);

datab : IN STD\_LOGIC\_VECTOR(7 DOWNTO 0);

agb : OUT STD\_LOGIC

);

END COMPONENT;

SIGNAL blue : STD\_LOGIC;

SIGNAL green : STD\_LOGIC;

SIGNAL maax : STD\_LOGIC;

SIGNAL max : STD\_LOGIC\_VECTOR(7 DOWNTO 0);

SIGNAL min : STD\_LOGIC\_VECTOR(7 DOWNTO 0);

SIGNAL noorm : STD\_LOGIC;

SIGNAL norm : STD\_LOGIC\_VECTOR(7 DOWNTO 0);

SIGNAL SYNTHESIZED\_WIRE\_0 : STD\_LOGIC;

SIGNAL SYNTHESIZED\_WIRE\_1 : STD\_LOGIC;

SIGNAL SYNTHESIZED\_WIRE\_2 : STD\_LOGIC;

SIGNAL SYNTHESIZED\_WIRE\_3 : STD\_LOGIC;

SIGNAL SYNTHESIZED\_WIRE\_4 : STD\_LOGIC;

SIGNAL SYNTHESIZED\_WIRE\_5 : STD\_LOGIC;

BEGIN

b2v\_inst : rg1

PORT MAP(clock => clk,

enable => SYNTHESIZED\_WIRE\_0,

data => x,

q => min);

b2v\_inst1 : rg1

PORT MAP(clock => clk,

enable => SYNTHESIZED\_WIRE\_1,

data => x,

q => norm);

SYNTHESIZED\_WIRE\_2 <= NOT(maax);

green <= noorm AND SYNTHESIZED\_WIRE\_2;

blue <= NOT(maax OR noorm);

LED\_RED <= maax AND clk;

LED\_BLUE <= clk AND blue;

b2v\_inst2 : rg1

PORT MAP(clock => clk,

enable => SYNTHESIZED\_WIRE\_3,

data => x,

q => max);

b2v\_inst3 : comp

PORT MAP(dataa => norm,

datab => min,

agb => noorm);

b2v\_inst4 : comp

PORT MAP(dataa => norm,

datab => max,

agb => maax);

SYNTHESIZED\_WIRE\_0 <= action(0) AND SYNTHESIZED\_WIRE\_4;

SYNTHESIZED\_WIRE\_1 <= SYNTHESIZED\_WIRE\_5 AND action(1);

SYNTHESIZED\_WIRE\_3 <= action(1) AND action(0);

SYNTHESIZED\_WIRE\_4 <= NOT(action(1));

SYNTHESIZED\_WIRE\_5 <= NOT(action(0));

LED\_GREEN <= green;

END bdf\_type;