

## Проектирование конечного автомата. (ИКПИ)

### Задание на курсовое проектирование.

Синтезировать конечный автомат с произвольной сменой состояний, каждое из которых отражает некоторое событие. Возможность перехода из одного состояния в другое зависит от сигнала “х”, который формируется с помощью счетчика тактов.

В синтез входит:

1. построение временной диаграммы работы счетчика тактов с указанием интервалов времени формирования управляющего сигнала «х»;
2. построение схем КЦУ управления автоматом, КЦУ связей автомата и КЦУ, преобразующего состояние автомата для вывода на индикаторы. Временные диаграммы их работы;
3. программа на VHDL для реализации автомата на макете в FPGA CycloneV ;
4. диаграмма переходов автомата.

В задании таблица переходов (Табл.1) отражает ход смены состояний с учетом времени удержания предшествующего состояния в тактах.

Таблица состояний (Табл.2) отражает событие, соответствующее полученному состоянию, и отображение состояния на выходной шине. Состояние выходной шины при реализации выводится на сегментные индикаторы.

#### Задание 1. Светофор.

Возможные состояния на выходной шине:

- красный (r);
- желтый (Y);
- красный+желтый (rY);
- зеленый (G);
- желтый +правый поворот (Y-);
- зеленый +правый поворот (G-);

- красный + левый поворот (- r).

### Вариант 1.1.

Табл. 1.

S0 (начальное)	Состояния		
x	S1	S2	S3
x=0	S1	S3	S3
x=1	S2	S3	S1
Время удержания (такт)	3	1	4

Табл. 2

состояние	S1	S2	S3
событие	5	7	9
выходная шина	r	Y	G

### Вариант 1.2.

Табл. 1

S0(начальное)	Состояния			
x	S1	S2	S3	S4
x=0	S1	S3	S3	S4
x=1	S2	S3	S4	S1
Время удержания(такт)	3	1	2	2

Табл. 2

состояние	S1	S2	S3	S4
событие	4	6	7	8
выходная шина	r	Y	G-	G

### Вариант 1.3

Табл. 1.

S0(начальное)	Состояния				
x	S1	S2	S3	S4	S5
x=0	S1	S3	S3	S5	S5
x=1	S2	S3	S4	S5	S1
Время удержания(такт)	3	1	2	1	3

Табл. 2.

состояние	S1	S2	S3	S4	S5
событие	1	2	7	8	10
выходная шина	r	Y	Y-	G-	G

### Вариант 1.4.

Табл. 1.

S0(начальное)	Состояния				
x	S1	S2	S3	S4	S5
x=0	S1	S2	S4	S4	S5
x=1	S2	S3	S4	S5	S1
Время удержания(такт)	2	2	1	2	2

Табл. 2.

состояние	S1	S2	S3	S4	S5
событие	10	4	5	8	6
выходная шина	- r	r	Y	Y-	G

### Вариант 1.5.

Табл. 1.

S0(начальное)	Состояния			
x	S1	S2	S3	S4
x=0	S1	S2	S4	S4
x=1	S2	S3	S4	S1
Время удержания(такт)	3	2	1	2

Табл. 2.

состояние	S1	S2	S3	S4
событие	3	6	8	9
выходная шина	r	- r	r Y	G

### Задание 2. Стиральная машина.

Автомат реализует три основных режима стиральной машины: «стирка», «полоскание» и «отжим». Отображение на индикаторах: С, П, О.

#### Вариант 2.1. Хлопчатобумажные ткани.

Табл. 1.

S0 (начальное)	Состояния		
x	S1	S2	S3
x=0	S1	S2	S3
x=1	S2	S3	S1
Время удержания (такт)	4	3	2

Табл. 2

состояние	S1	S2	S3
событие	10	11	12
выходная шина	C	П	O

**Вариант 2.2. Шерстяные ткани.**

Табл. 1.

S0 (начальное)	Состояния		
x	S1	S2	S3
x=0	S1	S2	S1
x=1	S2	S3	S1
Время удержания (такт)	2	3	1

Табл. 2

состояние	S1	S2	S3
событие	4	6	9
выходная шина	C	П	O

**Задание 3.****Работа кассового картридера с банковской картой.**

В этом задании выделяем 4 этапа:

- проверка наличия карты «П»;
- считывание кода «С»;
- обработка кода «O»;
- получение результата «Р».

Табл. 1.

S0(начальное)	Состояния			
x	S1	S2	S3	S4
x=0	S1	S2	S3	S4
x=1	S2	S3	S4	S0
Время удержания(такт)	2	2	3	2

Табл. 2

состояние	S0	S1	S2	S3	S4
событие	11	1	3	5	8
выходная шина	--	П	С	О	Р

После получения результата желательно приостановить работу устройства.

#### Задание 4.

##### Разметка прямоугольной фигуры.

Движение вперед – FO;

Движение вправо - r l;

Движение назад - rE;

Движение влево – LE.

Табл. 1.

S0(начальное)	Состояния			
x	S1	S2	S3	S4
x=0	S1	S2	S3	S4
x=1	S2	S3	S4	S0
Время удержания(такт)	5	2	5	2

Табл. 2

состояние	S0	S1	S2	S3	S4
событие	1	2	8	10	15
выходная шина	--	FO	r I	rE	LE

**Задание 5.**

**Последовательный вывод на индикатор многозначного числа.**

**Вариант 5.1.**

Число 55508991

Табл. 1.

S0(начальное)	Состояния				
x	S1	S2	S3	S4	S5
x=0	S1	S3	S4	S4	S1
x=1	S2	S3	S4	S5	S1
Время удержания(такт)	3	1	1	2	1

Табл. 2.

состояние	S1	S2	S3	S4	S5
событие	5	0	8	9	1
выходная шина (инд.)	5	0	8	9	1

**Вариант 5.2.**

Число 4223118

Табл. 1.

S0(начальное)	Состояния				
x	S1	S2	S3	S4	S5
x=0	S2	S2	S4	S4	S1
x=1	S2	S3	S4	S5	S1
Время удержания(такт)	1	2	1	2	1

Табл. 2.

состояние	S1	S2	S3	S4	S5
событие	4	2	3	1	8
выходная шина (инд.)	4	2	3	1	8

**Вариант 5.3.**

Число 772944443

Табл. 1.

S0(начальное)	Состояния				
x	S1	S2	S3	S4	S5
x=0	S1	S3	S4	S4	S1
x=1	S2	S3	S4	S5	S1
Время удержания(такт)	2	1	1	3	1

Табл. 2.

состояние	S1	S2	S3	S4	S5
событие	7	2	9	4	3
выходная шина (инд.)	7	2	9	4	3



**Вариант 5.4.**

Число 15333288

Табл. 1.

S0(начальное)	Состояния				
х	S1	S2	S3	S4	S5
х=0	S2	S3	S3	S5	S5
х=1	S2	S3	S4	S5	S1
Время удержания(такт)	1	1	3	1	2

Табл. 2.

состояние	S1	S2	S3	S4	S5
событие	1	5	3	2	8
выходная шина (инд.)	1	5	3	2	8

**Вариант 5.5.**

Число 92001115

Табл. 1.

S0(начальное)	Состояния				
х	S1	S2	S3	S4	S5
х=0	S2	S3	S3	S4	S1
х=1	S2	S3	S4	S5	S1
Время удержания(такт)	1	1	2	3	1

Табл. 2.

состояние	S1	S2	S3	S4	S5
событие	9	2	0	1	5
выходная шина	9	2	0	1	5

**Вариант 5.6.**

Число 26880777

Табл. 1.

S0(начальное)	Состояния				
x	S1	S2	S3	S4	S5
x=0	S2	S3	S3	S5	S5
x=1	S2	S3	S4	S5	S1
Время удержания(такт)	1	1	2	1	3

Табл. 2.

состояние	S1	S2	S3	S4	S5
событие	2	6	8	0	7
выходная шина (инд.)	2	6	8	0	7

**Вариант 5.7.**

Число 66899122

Табл. 1.

S0(начальное)	Состояния				
x	S1	S2	S3	S4	S5
x=0	S1	S3	S3	S5	S5
x=1	S2	S3	S4	S5	S1
Время удержания(такт)	2	1	2	1	2

Табл. 2.

состояние	S1	S2	S3	S4	S5
событие	6	8	9	1	2
выходная шина (инд.)	6	8	9	1	2

### Задание 6.

Последовательный вывод на индикатор слов, каждая буква слова выводится в течение 1 секунды.

Отображение слов: SLEEP, dEEr, rAbbIT

#### Вариант 6.1.

Табл. 1.

S0(начальное)	Состояния			
x	S1	S2	S3	S4
x=0	S2	S3	S3	S1
x=1	S2	S3	S4	S1
Время удержания(такт)	1	1	2	1

Табл. 2

состояние	S1	S2	S3	S4
событие	5	6	7	8
выходная шина	S	L	E	P

#### Вариант 6.2.

Табл. 1.

S0 (начальное)	Состояния		
x	S1	S2	S3
x=0	S2	S2	S1
x=1	S2	S3	S1
Время удержания (такт)	1	2	1

Табл. 2.

состояние	S1	S2	S3
событие	7	9	11
выходная шина	d	E	r

**Вариант 6.3.**

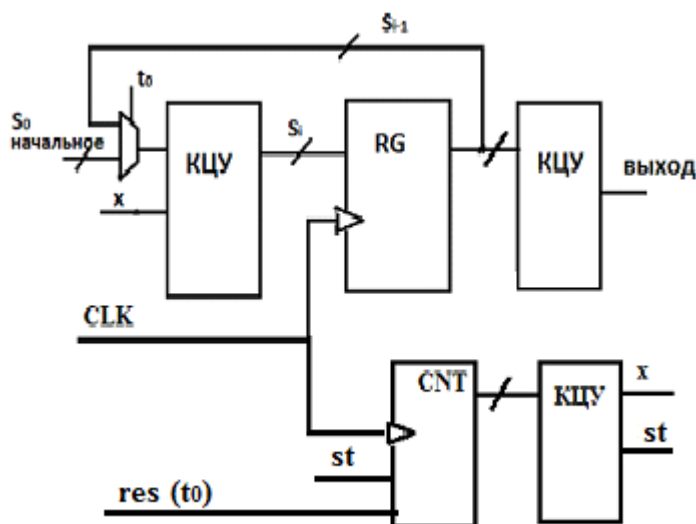
Табл. 1.

S0(начальное)	Состояния				
x	S1	S2	S3	S4	S5
x=0	S2	S3	S3	S5	S1
x=1	S2	S3	S4	S5	S1
Время удержания(такт)	1	1	2	1	1

Табл. 2.

состояние	S1	S2	S3	S4	S5
событие	7	8	10	12	13
выходная шина	r	A	b	I	T

Структура автомата может быть представлена следующим образом.



Синхронизация подается на схему от внешнего ввода в FPGA (50MHz, PIN AF14) через понижающий счетчик с возможностью его приостановки внешним сигналом с тумблера. Сигнал начальной установки счетчика в «0» может также подаваться от любого тумблера макета. При подаче этого сигнала автомат возвращается в начальное состояние, поэтому сигналы **res** и **t<sub>0</sub>** подаются от одного источника.

### Методика выполнения курсовой работы.

#### Занятие 1.

Рассмотрим пример.

В качестве примера возьмем светофор, в котором переход из красного цвета в зеленый и из зеленого в красный происходит через короткий промежуток желтого.

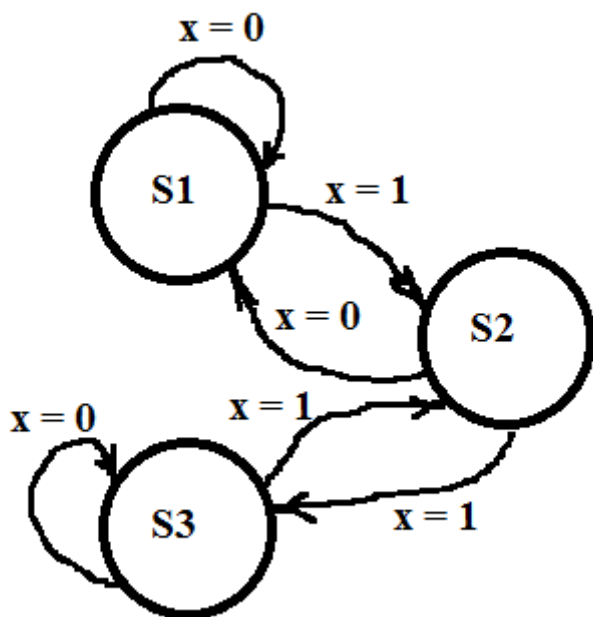
Табл. 1.

S0 (начальное)	Состояния		
x	S1	S2	S3
x=0	S1	S1	S3
x=1	S2	S3	S2
Время удержания (такт)	3	1	3

Табл. 2

состояние	S1	S2	S3
событие	4	9	12
выходная шина	r	Y	G

Граф работы автомата:

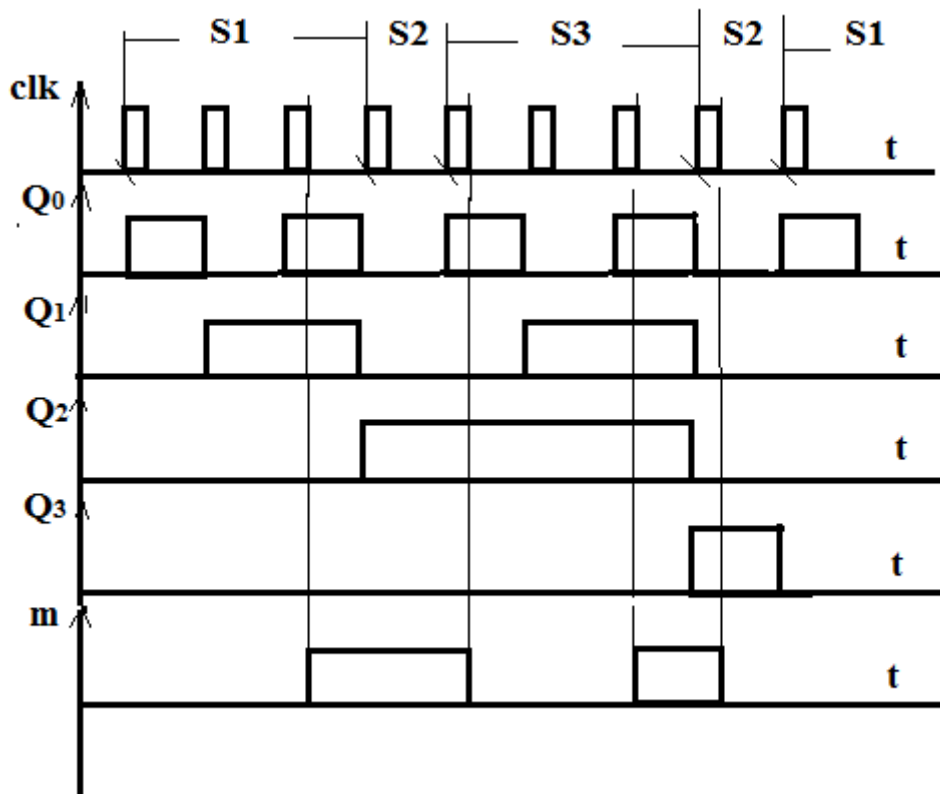


### 1. Порядок работы счетчиков.

Согласно заданию, наш счетчик должен держать автомат в состоянии S1 в течение 3-х тактов, а затем выводить в состояние S2. Значит, при переходе **счетчика** в состояние «3», управляющее КЦУ формирует «1» для входа “x” автомата. Таким образом, следующий импульс синхронизации позволит автомату перейти в состояние S2.

Состояние S2 держится 1 такт, но переход из него в S3 происходит по «1» на входе управления. Следовательно, управляющее КЦУ при состоянии **счетчика** «4» продолжает формировать «1». Формирование «0» на КЦУ происходит при приходе **счетчика** в состояние «5».

Удержание состояния S3 также происходит в течение 3-х тактов. Нетрудно определить, что в этом случае КЦУ формирует «1» после прихода **счетчика** в состояние «7». Следующее за этим состояние S2 держится 1 такт и переходит в S1 по управлению уровнем «0». Из состояния **счетчика** «8» КЦУ формирует «0», **счетчик** возвращается в состояние «1» и цикл повторяется.



Теперь построим управляющее КЦУ.

Нанесем условия на карту Карно

		<u>Q1</u>	
	<u>Q0</u>		
Q2   Q3	0	0 <sub>0</sub>	1 <sub>3</sub>
	1	0 <sub>5</sub>	1 <sub>7</sub>
	Φ <sub>12</sub>	Φ <sub>13</sub>	Φ <sub>15</sub>
	0 <sub>8</sub>	Φ <sub>9</sub>	Φ <sub>11</sub>
		0 <sub>2</sub>	0 <sub>6</sub>
		Φ <sub>14</sub>	Φ <sub>10</sub>

Получим зависимость для сигнала управления m:

$$m = Q_0 \wedge Q_1 \vee \bar{Q}_0 \wedge \bar{Q}_1 \wedge Q_2$$

По формуле можно построить схему КЦУ.

## Занятие 2.

Теперь можем записать программы функционирования счетчика и КЦУ.

Для счетчика:

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;
entity count is
port(clk_i,res:in std_logic;
d_o:out std_logic_vector(3 downto 0));
end count;
architecture prc of count is
signal cnt:std_logic_vector(3 downto 0);
begin
process(cnt,res)
begin
if(res='1')then cnt<=(others=>'0');
elsif(rising_edge(clk_i))then
cnt<=cnt+"0001";
if cnt="1000"then cnt<="0001";
end if;
end if;
end process;
d_o<=cnt;
end;
```

Для КЦУ:

```
library ieee;
use ieee.std_logic_1164.all;
entity cd is
port(Q:in std_logic_vector(3 downto 0);
m: out std_logic);
end cd;
architecture nn of cd is
begin
process(adr)
begin
case Q is
when "0000"=>m<='0';
when "0001"=>m<='0';
when "0010"=>m<='0';
```



```

when "0011"=>m<='1';
when "0100"=>m<='1';
when "0101"=>m<='0';
when "0110"=>m<='0';
when "0111"=>m<='1';
when "1000"=>m<='0';
when others=>m<='X';
end case;
end process;
end;

```

Описание программы работы автомата.

Описание КЦУ связей автомата.

```

library ieee;
use ieee.std_logic_1164.all;
entity moore is
port(
clk : in std_logic;
x : in std_logic;
reset : in std_logic;
output : out std_logic_vector(3 downto 0)
);
end entity;
architecture rtl of moore is
type state_type is (s0, s1, s2, s3);
signal state : state_type;
begin
process (clk, reset)
begin
if reset = '1' then
state <= s0;
elsif (rising_edge(clk)) then
case state is
when s0=>
if x = '1' then
state <= s1;
else
state <= s1;
end if;
when s1=>
if x = '1' then
state <= s2;

```

```
else
state <= s1;
end if;
when s2=>
if x = '1' then
state <= s3;
else
state <= s1;
end if;
when s3 =>
if x = '1' then
state <= s2;
else
state <= s3;
end if;
end case;
end if;
end process;
process (state)
begin
case state is
when s0 =>
output <= "1111";
when s1 =>
output <= "0100";
when s2 =>
output <= "1001";
when s3 =>
output <= "1100";
when others =>
output <= "1111";
end case;
end process;
end rtl;
```