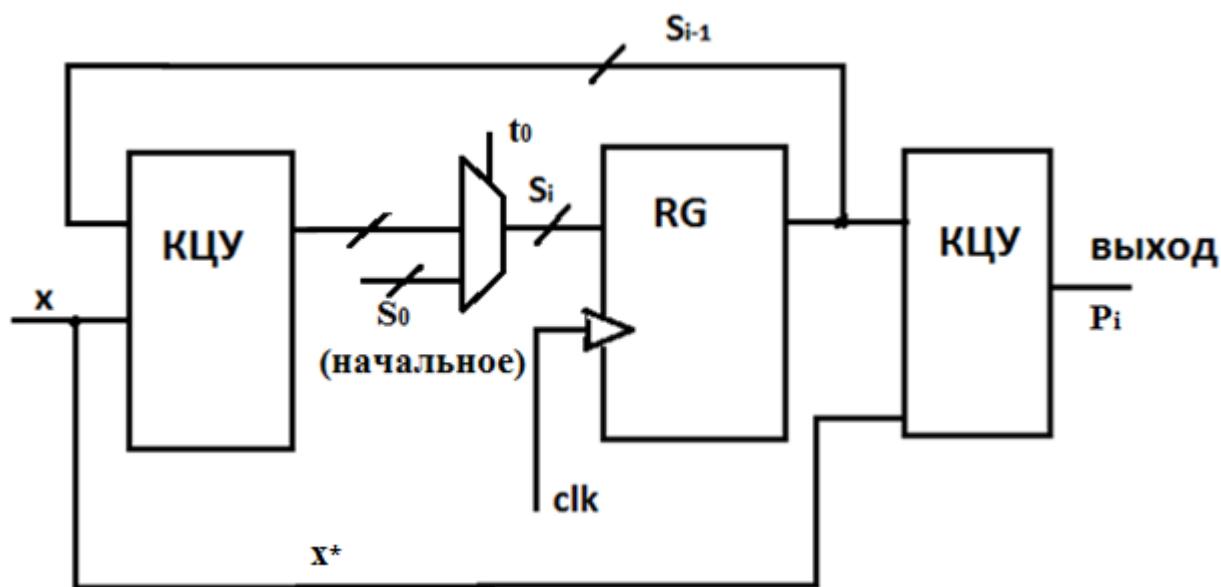


## Задание.



Синтезировать конечный автомат, работающий по принципу автомата Мили. В синтез входит построение схемы КЦУ выхода автомата, временные диаграммы его работы, программа на Verilog HDL для реализации автомата на макете в FPGA CycloneV и диаграмма переходов автомата. Вариант выбирается в соответствии с последней и предпоследней цифрами номера зачетной книжки (студенческого билета). Соответственно: последняя цифра – номер таблицы переходов, предпоследняя – номер таблицы состояний. Сигналы  $x$  и  $x^*$  определяются комбинацией связанных сигналов «a» и «b». Переходы регулирует «a», состояния – «b».

Примером такого автомата может служить автомат, регулирующий работу светофора со стрелками боковых поворотов. При трех основных переходах можно получить от 4 до 6-ти возможных состояний. Как переходы, так и события внутри состояний регулируются по времени удержания. Время удержания основного состояния между переходами поглощает время смены событий внутри состояний.

## Варианты заданий.

Таблица переходов 0								
x	S0	S1	S2	S3	S4	S5	S6	S7
X=0	S2	S3	S4	S5	S1	S7	S0	S6
X=1	S1	S4	S6	S7	S2	S3	S5	S0

Таблица переходов 1								
x	S0	S1	S2	S3	S4	S5	S6	S7
X=0	S7	S0	S1	S2	S3	S4	S5	S6
X=1	S1	S5	S6	S7	S0	S3	S4	S2

Таблица переходов 2								
x	S0	S1	S2	S3	S4	S5	S6	S7
X=0	S4	S5	S6	S7	S1	S2	S3	S0
X=1	S3	S5	S7	S1	S2	S4	S0	S6

Таблица переходов 3								
x	S0	S1	S2	S3	S4	S5	S6	S7
X=0	S6	S3	S4	S0	S7	S1	S2	S5
X=1	S5	S6	S0	S2	S3	S7	S4	S1

Таблица переходов 4								
x	S0	S1	S2	S3	S4	S5	S6	S7
X=0	S3	S6	S1	S7	S0	S2	S4	S5
X=1	S5	S4	S1	S7	S6	S3	S0	S2

Таблица переходов 5								
x	S0	S1	S2	S3	S4	S5	S6	S7
X=0	S1	S4	S6	S7	S2	S3	S5	S0
X=1	S2	S3	S4	S5	S1	S7	S0	S6

Таблица переходов 6								
x	S0	S1	S2	S3	S4	S5	S6	S7
X=0	S1	S5	S6	S7	S0	S3	S4	S2
X=1	S7	S0	S1	S2	S3	S4	S5	S6

Таблица переходов 7								
X	S0	S1	S2	S3	S4	S5	S6	S7
X=0	S3	S5	S7	S1	S2	S4	S0	S6
X=1	S4	S5	S6	S7	S1	S2	S3	S0

Таблица переходов 8								
X	S0	S1	S2	S3	S4	S5	S6	S7
X=0	S5	S4	S1	S7	S6	S3	S0	S2
X=1	S3	S6	S1	S7	S0	S2	S4	S5

Таблица переходов 9								
X	S0	S1	S2	S3	S4	S5	S6	S7
X=0	S3	S6	S1	S7	S0	S2	S4	S5
X=1	S1	S4	S6	S7	S2	S3	S5	S0

Таблица состояний 0								
X*	S0	S1	S2	S3	S4	S5	S6	S7
X*=0	0	1	2	3	4	5	6	7
X*=1	7	6	5	4	3	2	1	0

Таблица состояний 1								
X*	S0	S1	S2	S3	S4	S5	S6	S7
X*=0	1	2	3	4	5	6	7	0
X*=1	0	7	6	5	4	3	2	1

Таблица состояний 2								
X*	S0	S1	S2	S3	S4	S5	S6	S7
X*=0	2	3	4	5	6	7	0	1
X*=1	1	0	7	6	5	4	3	2

Таблица состояний 3								
X*	S0	S1	S2	S3	S4	S5	S6	S7
X*=0	3	4	5	6	7	0	1	2
X*=1	2	1	0	7	6	5	4	3

Таблица состояний 4								
X*	S0	S1	S2	S3	S4	S5	S6	S7
X*=0	4	5	6	7	0	1	2	3
X*=1	3	2	1	0	7	6	5	4

Таблица состояний 5								
X*	S0	S1	S2	S3	S4	S5	S6	S7
X*=0	5	6	7	0	1	2	3	4
X*=1	4	3	2	1	0	7	6	5

Таблица состояний 6								
X*	S0	S1	S2	S3	S4	S5	S6	S7
X*=0	6	7	0	1	2	3	4	5
X*=1	5	4	3	2	1	0	7	6

### Таблица состояний 7

$X^*$	S0	S1	S2	S3	S4	S5	S6	S7
$X^*=0$	7	0	1	2	3	4	5	6
$X^*=1$	6	5	4	3	2	1	0	7

### Таблица состояний 8

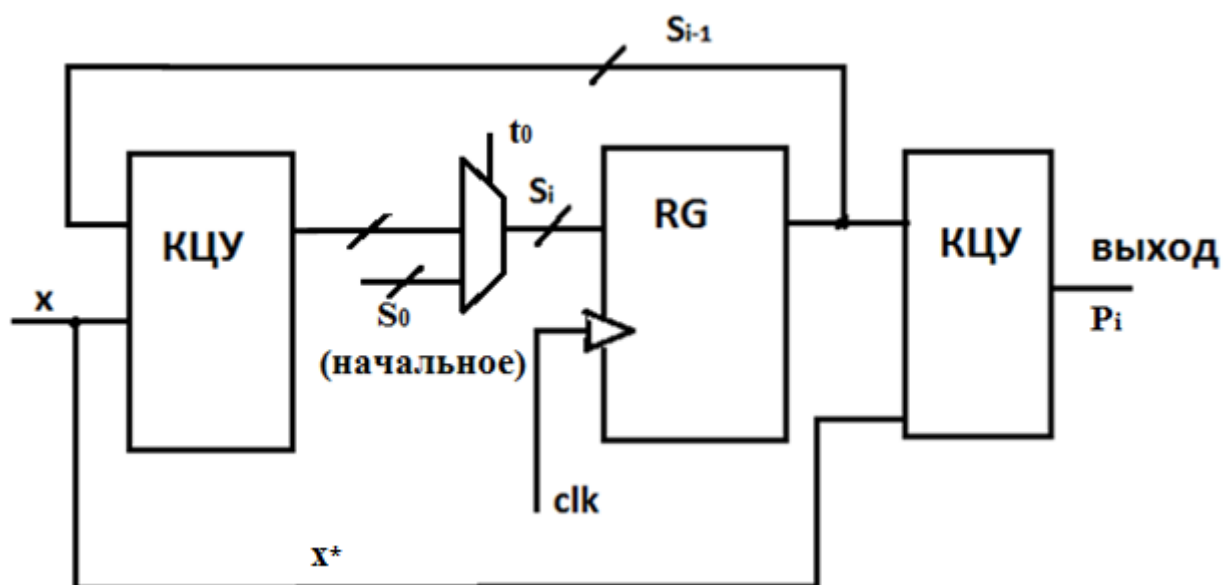
$X^*$	S0	S1	S2	S3	S4	S5	S6	S7
$X^*=0$	0	2	4	6	1	3	5	7
$X^*=1$	1	3	5	7	2	4	6	0

### Таблица состояний 9

$X^*$	S0	S1	S2	S3	S4	S5	S6	S7
$X^*=0$	0	7	6	5	4	3	2	1
$X^*=1$	1	2	3	4	5	6	7	0

### Пример синтеза.

Структура автомата может быть представлена следующим образом (повторение материала лекции 6).



Построим схему автомата Мили, имеющего 3 разряда.

Предположим, таблица переходов и таблица состояний автомата будут иметь следующий вид:

Таблица переходов								
x	S0	S1	S2	S3	S4	S5	S6	S7
X=0	S2	S3	S4	S5	S6	S0	S7	S1
X=1	S3	S5	S4	S1	S6	S7	S0	S2

Таблица состояний								
X*	S0	S1	S2	S3	S4	S5	S6	S7
X*=0	1	3	5	7	0	2	4	6
X*=1	2	4	6	0	1	3	5	7

Значит, при  $x=0$  автомат будет совершать переходы:

$S0 \rightarrow S2 \rightarrow S4 \rightarrow S6 \rightarrow S7 \rightarrow S1 \rightarrow S3 \rightarrow S5 \rightarrow S0 \rightarrow \dots$

А при  $x=1$

$S0 \rightarrow S3 \rightarrow S1 \rightarrow S5 \rightarrow S7 \rightarrow S2 \rightarrow S4 \rightarrow S6 \rightarrow S0 \rightarrow \dots$

При этом изменение  $x^*$  будет сказываться на выходе автомата следующим образом:

При  $x^*=0$  эквивалентность состояний следующая:

$S0==1, S1==3, S2==5, S3==7, S4==0, S5==2, S6==4, S7==6;$

При  $x^*=1$ :

$S0==2, S1==4, S2==6, S3==0, S4==1, S5==3, S6==5, S7==7;$

Принимая во внимание, что S всего лишь – State (состояние), по этим данным запишем таблицу истинности для второго КЦУ автомата, именуя входы  $S_i$  -  $M[2..0]$ , а выходы  $P_i$  -  $N[2..0]$ .

X*	M <sub>2</sub>	M <sub>1</sub>	M <sub>0</sub>	N <sub>2</sub>	N <sub>1</sub>	N <sub>0</sub>
0	0	0	0	0	0	1
0	0	0	1	0	1	1
0	0	1	0	1	0	1
0	0	1	1	1	1	1
0	1	0	0	0	0	0
0	1	0	1	0	1	0
0	1	1	0	1	0	0
0	1	1	1	1	1	0
1	0	0	0	0	1	0
1	0	0	1	1	0	0
1	0	1	0	1	1	0
1	0	1	1	0	0	0
1	1	0	0	0	0	1
1	1	0	1	0	1	1
1	1	1	0	1	0	1
1	1	1	1	1	1	1

Запишем выражения для выходов, используя карты Карно.

				$M_1$	
		$M_0$			
		00	01	11	10
$M_2$	00	1 0	1 1	1 3	1 2
	01	4	5	7	6
	11	1 12	1 13	1 15	1 14
	10	8	9	11	10

Для N<sub>0</sub>

$$N_0 = \overline{X} \& \overline{M_2} \vee X \& M_2$$

		$M_1$			
		$M_0$			
		00	01	11	10
$M_2$	00	0	1 1	1 3	2
	01	4	1 5	1 7	6
	11	12	1 13	1 15	14
	10	1 8	9	11	1 10
$X$					

Для  $N_1$

$$N_1 = \overline{X} \& M_0 \vee M_2 \& M_0 \vee X \& \overline{M_2} \& \overline{M_0}$$

		$M_1$			
		$M_0$			
		00	01	11	10
$M_2$	00	0	1	1 3	1 2
	01	4	5	1 7	1 6
	11	12	13	1 15	1 14
	10	8	1 9	11	1 10
$X$					

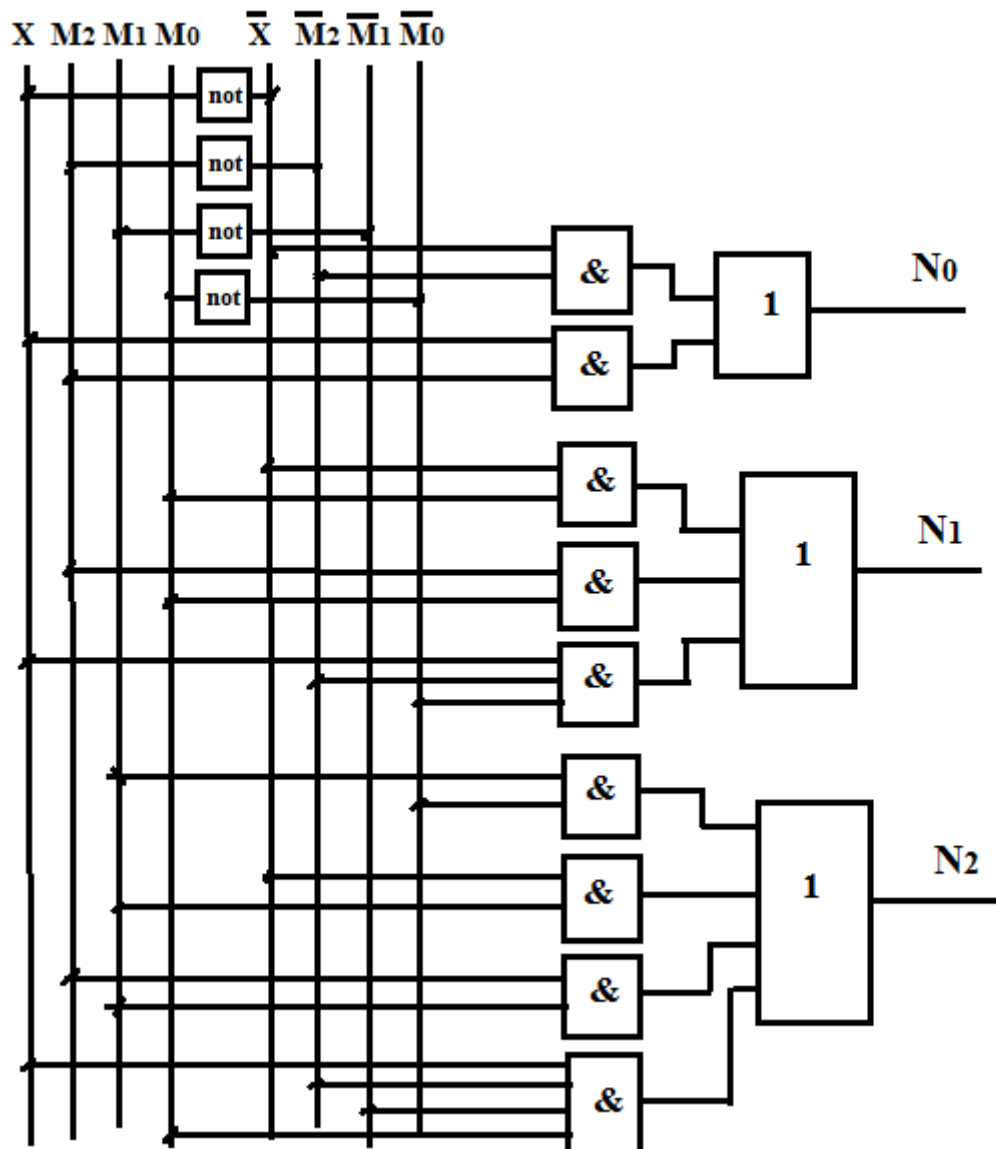
Для  $N_2$

$$N_2 = M_1 \& \overline{M_0} \vee \overline{X} \& M_1 \vee M_2 \& M_1 \vee X \& \overline{M_2} \& \overline{M_1} \& M_0$$

На основании выведенных уравнений построим структурную схему второго КЦУ.



## Структурная схема КЦУ.



Для построения функциональной диаграммы работы КЦУ записываем программу:

```
Module block2
```

```
(input x, input [2:0]m, output n0,n1,n2);
```

```
assign n0 = ~x&~m[2] | x&m2;
```

```
assign n1= ~x&m[0] | m[2]&m[0] | x&~m[2]&~m[0];
```

```
assign n2=m[1]&~m[0] | ~x&m[1] | m[2]&m[1] | x&~m[2]&~m[1]&m[0];
```

```
endmodule
```

Программа работы устройства будет состоять из двух частей: программирования переходов в следующее состояние на момент подачи импульса синхронизации (первое КЦУ), и программирование соответствия состояния определенному числу (второе КЦУ). Принимаем  $x == a$ ;  $x* == b$ .

```
module mealy_3
(input clk,a, b, reset,
 output reg [2:0] out);
reg [2:0] state;
assign in1=a&(~b) | a;
assign in2=b&(~a) | b;

parameter S0 = 0, S1 = 1, S2 = 2, S3 = 3, S4 = 4, S5 = 5, S6 = 6, S7 = 7;
always @ (posedge clk or posedge reset)
begin
    if (reset)
        state <= S0;
    else
        case (state)
            S0:
                if (in1)
                    begin
                        state <= S3;
                    end
                else
                    begin
                        state <= S2;
                    end
            S1:
                if (in1)
                    begin
                        state <= S5;
                    end
                else
                    begin
                        state <= S3;
                    end
            S2:
                if (in1)
                    begin
                        state <= S4;
                    end
                else
                    begin
                        state <= S4;
                    end
        endcase
    end
```

```
        end
S3:
if (in1)
begin
state <= S1;
end
    else
    begin
state <= S5;
end
S4:
if (in1)
begin
state <= S6;
end
    else
    begin
state <= S6;
end
S5:
if (in1)
begin
state <= S7;
end
    else
    begin
state <= S0;
end
S6:
if (in1)
begin
state <= S0;
end
    else
    begin
state <= S7;
end
S7:
if (in1)
begin
state <= S2;
end
    else
    begin
state <= S1;
```

```

                                end
                        endcase
end
always @ (state or in2)
begin
    case (state, in2)
        S0:
            if (in2)
                begin
                    out = 3'b010;
                end
            else
                begin
                    out = 3'b001;
                end
        S1:
            if (in2)
                begin
                    out = 3'b100;
                end
            else
                begin
                    out = 3'b011;
                end
        S2:
            if (in2)
                begin
                    out = 3'b110;
                end
            else
                begin
                    out = 3'b101;
                end
        S3:
            if (in2)
                begin
                    out = 3'b000;
                end
            else
                begin
                    out = 3'b111;
                end
        S4:
            if (in2)
                begin

```

```
        out = 3'b001;
    end
        else
        begin
        out = 3'b000;
        end
    S5:
    if (in2)
    begin
    out = 3'b011;
    end
        else
        begin
        out = 3'b010;
        end
    S6:
    if (in2)
    begin
    out = 3'b101;
    end
        else
        begin
        out = 3'b100;
        end
    S7:
    if (in2)
    begin
    out = 3'b111;
    end
        else
        begin
        out = 3'b110;
        end
    endcase
end
endmodule
```