

Лабораторная работа № 7.

Исследование счетчиков.

Цель работы: изучение функционирования схем синхронных счетчиков. Сравнительный анализ синхронных счетчиков.

В работе предлагаются для изучения Verilog *HDL* – программы описания функционирования синхронных счетчиков: два варианта четырехразрядного счетчика-делителя частоты и двоичного четырехразрядного реверсивного счетчика. По функциональным диаграммам делаются выводы о работе счетчиков.

Порядок выполнения работы.

1. Создать проект, открыть Verilog *HDL* - файл и записать программу 7.1, отражающую функционирование 4-разрядного двоичного счетчика-делителя частоты. Откомпилировать файл.
2. Построить функциональные диаграммы работы счетчика, задав **интервал синхроимпульсов 20нс, а reset сформировать коротким «0» на интервале от 0нс до 10нс при базовом уровне «1», сигнал разрешения установить в «1»**. По диаграммам сделать вывод о коэффициенте деления частоты и форме полученного сигнала синхронизации на выходе.
3. Открыть новый Verilog *HDL* - файл и записать программу 7.2, отражающую функционирование недвоичного счетчика - делителя. Откомпилировать файл.
4. Построить функциональные диаграммы работы счетчика с такими же входными параметрами, сделать выводы о коэффициенте деления и форме полученного сигнала на выходе.

Программа 7.1

```
module count_div
# (parameter N=4)
(input wire clk, reset, ena,
output wire sync,
output wire [N-1:0]q);
reg [N-1:0] cnt;
reg [N-1:0] cnt_next;
assign q=cnt;
always@(posedge clk, negedge reset)
    if(!reset)
        cnt<=0;
    else
        cnt<=cnt_next;
always@*
```

```

        if(ena)
            cnt_next=cnt + 1;
        else
            cnt_next=cnt;
assign sync=(cnt>=2** $(N-1)$ )?1'b1:1'b0;
endmodule

```

Программа 7.2

```

module count_div1
# (parameter N=4, M=10)
(input wire clk, reset,
output wire sync,
output wire [N-1:0]q);
reg [N-1:0] cnt;
wire [N-1:0] cnt_next;
assign q=cnt;
always@(posedge clk, negedge reset)
    if(!reset)
        cnt<=0;
    else
        cnt<=cnt_next;
assign cnt_next = (cnt==(M-1)) ? 0 : (cnt+1);
assign sync = (cnt==(M-1)) ? 1'b1 : 1'b0;
endmodule

```

5. Открыть новый Verilog *HDL* - файл и записать программу 7.3, отражающую функционирование 4-разрядного реверсивного счетчика с возможностями асинхронного и синхронного сброса и остановки счета. Откомпилировать файл.

6. Построить функциональные диаграммы работы счетчика, задав **интервал изменения синхроимпульсов 20нс;**
интервал изменения реверса – 500нс;
асинхронный сброс - reset сформировать коротким «0» на интервале от 0нс до 10нс при базовом уровне «1»;
сигнал синхронного сброса установить короткой «1» в интервале от 80 до 110нс;
сигнал разрешения установить в «1» и вывести короткий «0» на интервале от 170 до 190нс.

Программа 7.3

```

module count
# (parameter N=4)
(input wire clk, reset, res, ena, rev,

```

```
output wire [N-1:0] q;
reg [N-1:0] cnt;
reg [N-1:0] cnt_next;
assign q=cnt;
always@ (posedge clk, negedge reset)
    if(!reset)
        cnt<=0;
    else
        cnt<=cnt_next;
    always@*
        if(res)
            cnt_next=0;
        else if(ena&!rev)
            cnt_next=cnt + 1;
        else if(ena&rev)
            cnt_next=cnt-1;
        else
            cnt_next=cnt;

endmodule
```