БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ и РАДИОЭЛЕКТРОНИКИ

Факультет КСиС

Кафедра ЭВМ ФКСиС

Контроль и диагностика средств вычислительной техники

Лабораторная работа № 4

Разработка потоковых, структурных и поведенческих моделей и тестов контроля цифровых устройств с памятью

Вариант № 7

Ст. гр. 550502 Преподаватель:

Кессо П.И. доцент каф. ЭВМ ФКСиС

Золоторевич Л. А.

Минск 2018

**Вариант 7**

Особенности функционирования триггера:

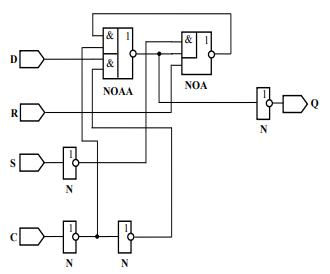
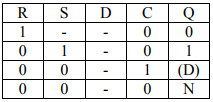


Рис 1.1 D-триггер со сбросом и установкой

Таблица функционирования:



Функциональный тест:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | R | S | D | C | Q |
|  | 0 | 0 | 0 | 0 | X |
| Установка 0 | 1 | 0 | 0 | 0 | 0 |
| Хранение 0 | 0 | 0 | 0 | 0 | 0 |
| Установка 1 | 0 | 1 | 0 | 0 | 1 |
| Хранение 1 | 0 | 0 | 0 | 0 | 1 |
| Установка 0 | 0 | 0 | 0 | 1 | 0 |
| Хранение 0 | 0 | 0 | 0 | 0 | 0 |
|  | 0 | 0 | 1 | 0 | 0 |
| Установка 1 | 0 | 0 | 1 | 1 | 1 |
| Хранение 1 | 0 | 0 | 0 | 0 | 1 |

Описание кода программы в среде VLSI-SIM:

circuit KESSO\_L4;

inputs D(1), R(1), S(1),C(1);

outputs Q(1);

GATES

A1 'N\_'(1) S(1);

A2 'N\_'(1) C(1);

B1 'NOA22'(1) C1(1), A2(1), D(1), B2(1);

B2 'N\_'(1) A2(1);

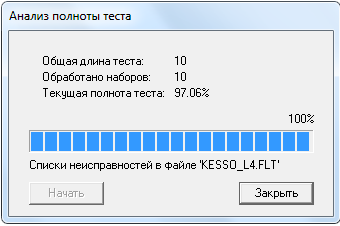
C1 'NOA2'(1) R(1), B1(1), A1(1);

Q 'N\_'(1) B1(1);

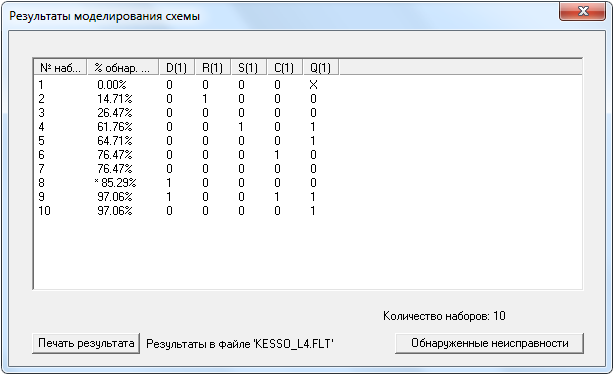
ENDGATES

END

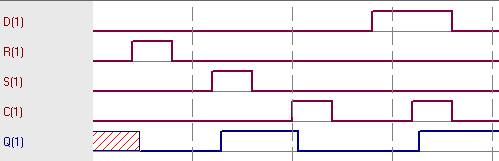
Определение тестового покрытия:



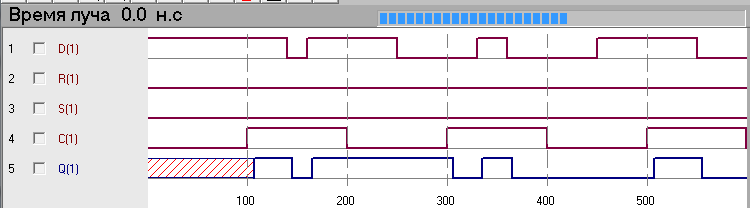
Анализ полноты теста:



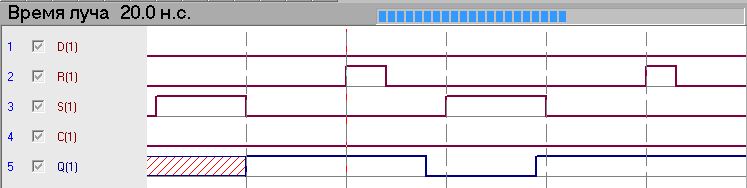
Результаты моделирования:



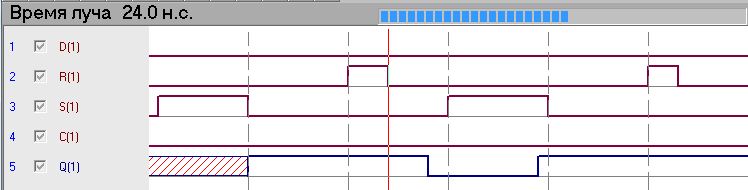
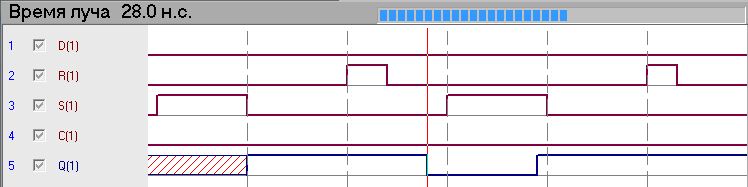
Результаты моделирования:



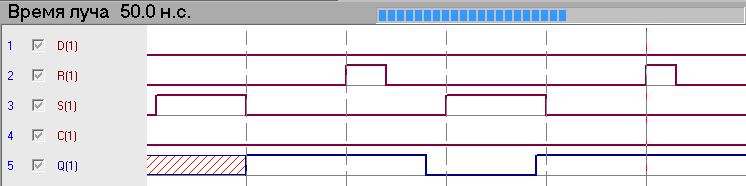
**Анализ времени удержания и время страбатывания сигнала R с помощью программы SCA-TIME**

****

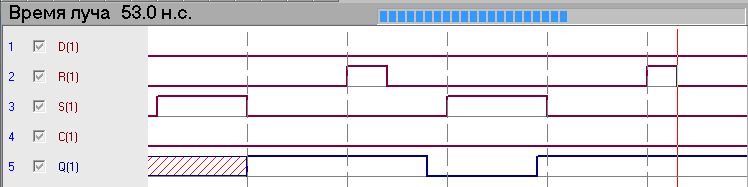
Начало R - 20 ns

****Окончание R - 24 ns****

Время сброса Q - 28 ns

****

Время установки R - 50ns

****

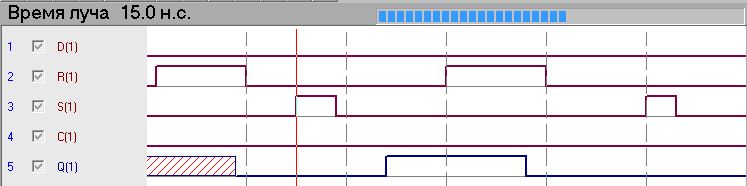
Время сброса R - 53ns

Проведя анализ мы выяснили, что :

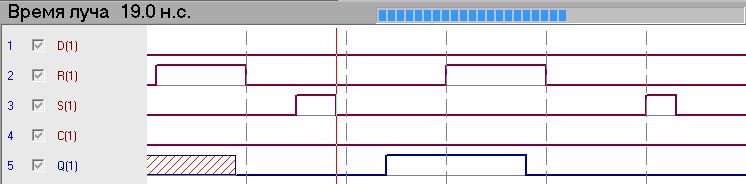
Время удержания R = 4ns

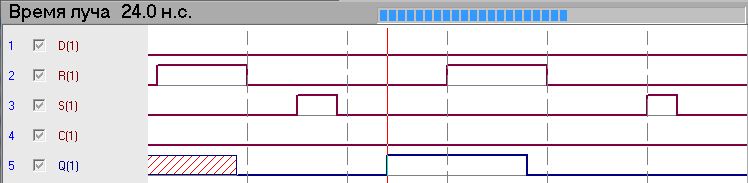
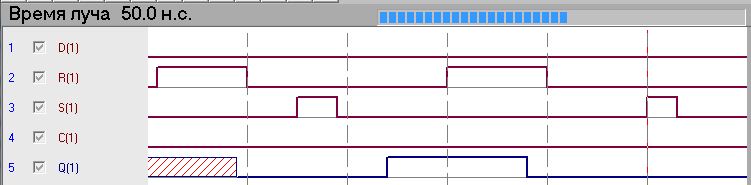
Время срабатывания R = 8 ns

**Анализ времени удержания и время срабатывания сигнала S с помощью программы SCA-TIME**

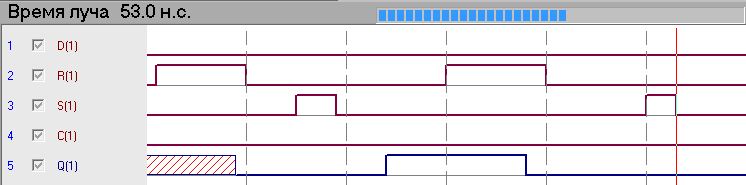
****

Время установки S - 15 ns

****

Время сброса S - 19 ns****Время установки Q - 24 ns****

Время установки S - 50 ns

****

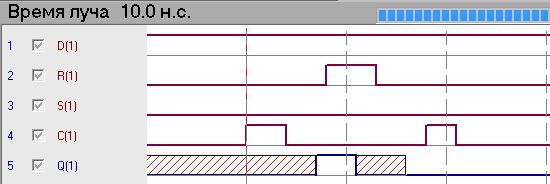
Время сброса S - 53 ns

Проведя анализ мы выяснили, что :

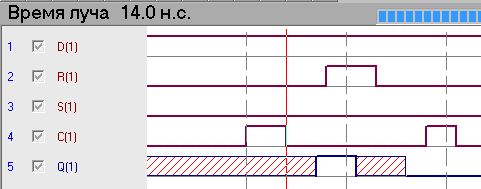
Время удержания S = 4ns

Время срабатывания S = 8 ns

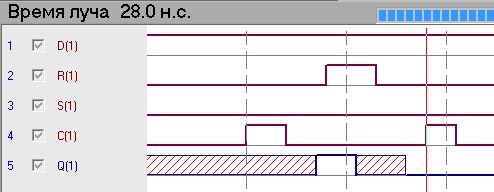
**Анализ времени удержания сигнала C при установке 1 с помощью программы SCA-TIME**



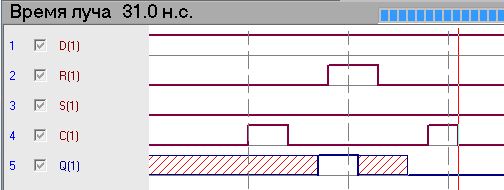
Время установки C - 10 ns



Время сброса С - 14 ns



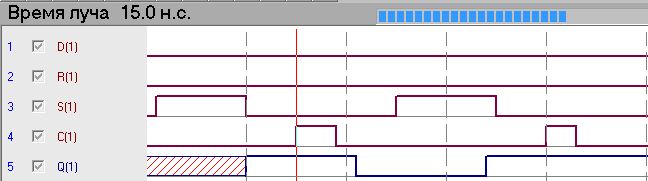
Время установки C - 28 ns



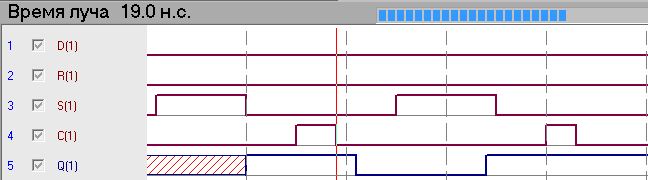
Время сброса С - 31 ns

Проведя анализ мы выяснили, что :

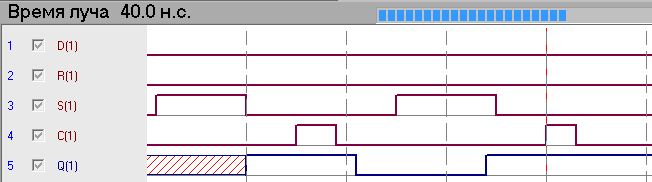
Время удержания С = 4ns

**Анализ времени удержания сигнала C при установке 0 с помощью программы SCA-TIME** 

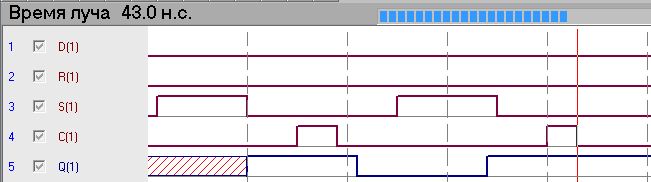
Время установки C - 15 ns



Время сброса C - 19 ns



Время установки C - 40 ns

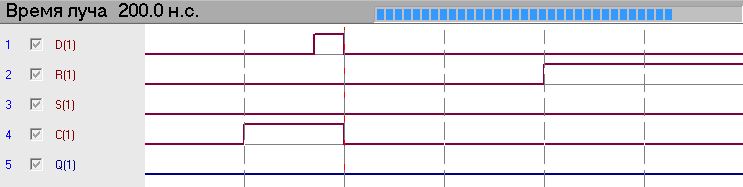
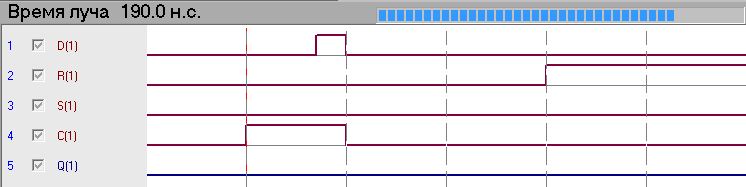
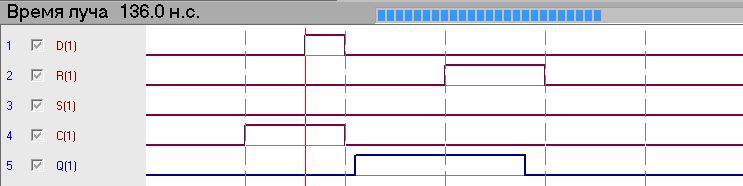
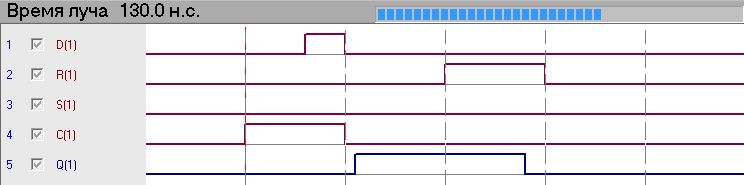
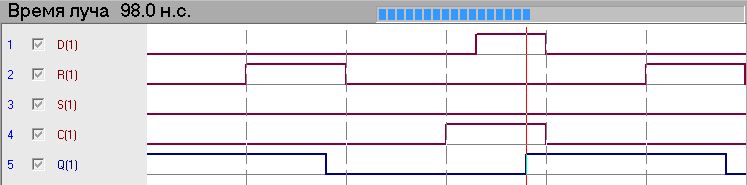
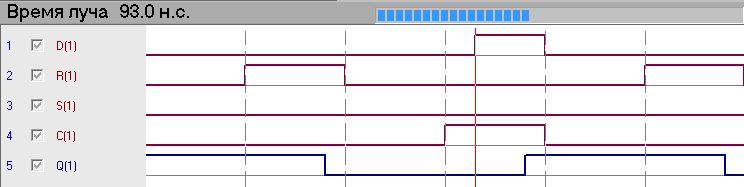
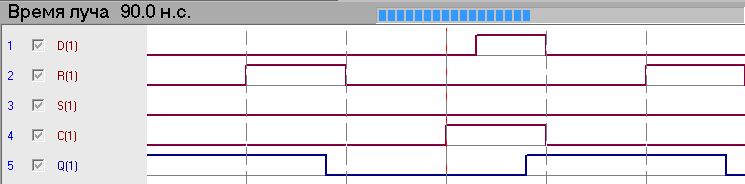
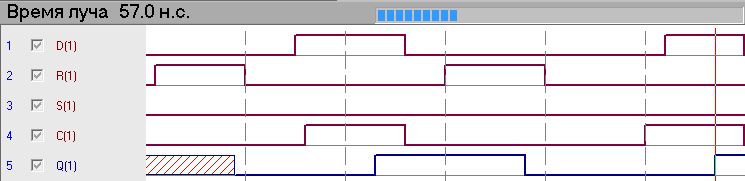
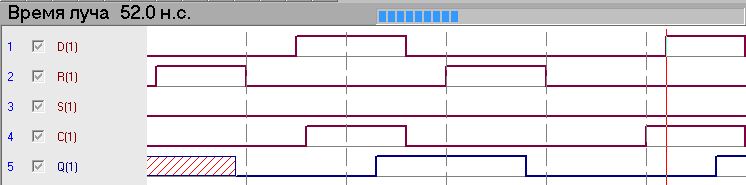
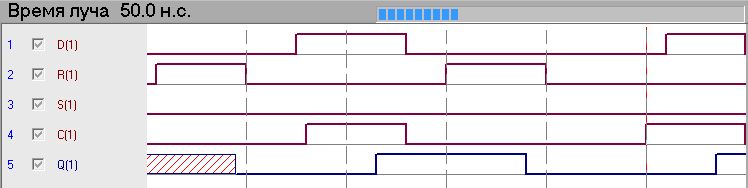
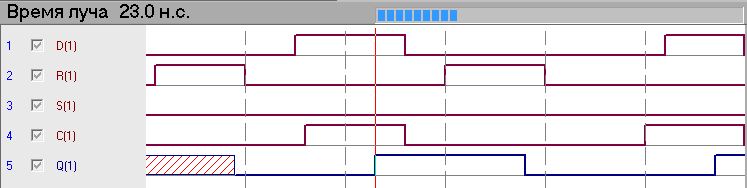
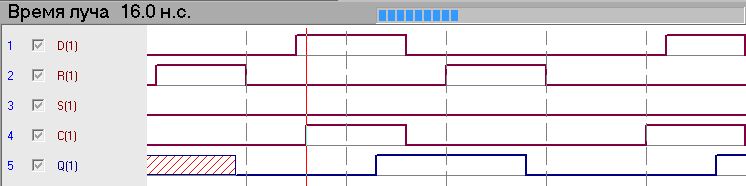
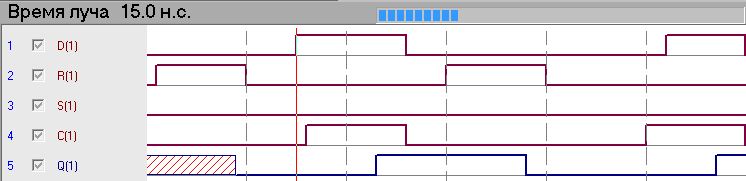


Время сброса C - 43 ns

Проведя анализ мы выяснили, что :

Время удержания С = 4ns

**Анализ времени удержания сигнала D с помощью программы SCA-TIME**



Проведя анализ мы выяснили, что :

Время удержания D = 4ns

Время предустановки C = 2ns

Время установки Q = 5ns

**Структурное описание схемы на языке VHDL**

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

ENTITY lr4 IS

PORT (

D : in STD\_LOGIC;

R : in STD\_LOGIC;

S : in STD\_LOGIC;

C : in STD\_LOGIC;

Q : out STD\_LOGIC

);

END lr4;

ARCHITECTURE Arch OF lr4 IS

component n

Port ( X : in STD\_LOGIC;

Y : out STD\_LOGIC);

end component;

component noa

Port ( X1 : in STD\_LOGIC;

X2 : in STD\_LOGIC;

X3 : in STD\_LOGIC;

Y : out STD\_LOGIC);

end component;

component noaa

Port ( X1 : in STD\_LOGIC;

X2 : in STD\_LOGIC;

X3 : in STD\_LOGIC;

X4 : in STD\_LOGIC;

Y : out STD\_LOGIC);

end component;

SIGNAl A1 : STD\_LOGIC;

SIGNAl A2 : STD\_LOGIC;

SIGNAl B1 : STD\_LOGIC;

SIGNAl B2 : STD\_LOGIC;

SIGNAl C1 : STD\_LOGIC;

SIGNAL C2 : STD\_LOGIC;

BEGIN

a\_1: n port map(S, A1);

a\_2: n port map(C, A2);

b\_1: noaa port map(C1, A2, D, B2, B1);

b\_2: n port map(A2, B2);

c\_1: noa port map(A1, B1, R, C1);

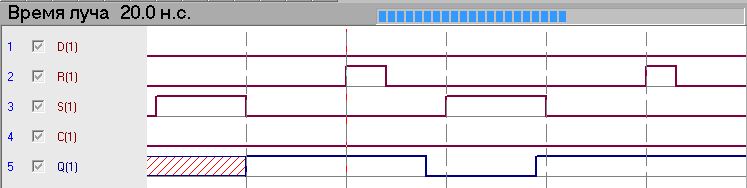
c\_2: n port map(B1, C2);

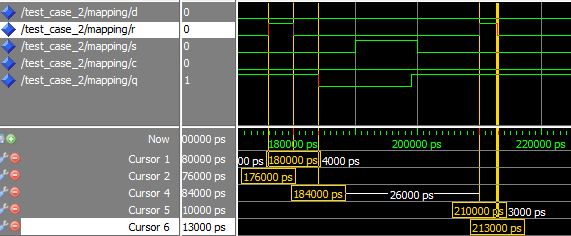
Q <= C2;

END Arch;

**Сравнение результатов моделирования VLSI-SIM и ModelSim**

**Анализ времени удержания и время срабатывания сигнала R**

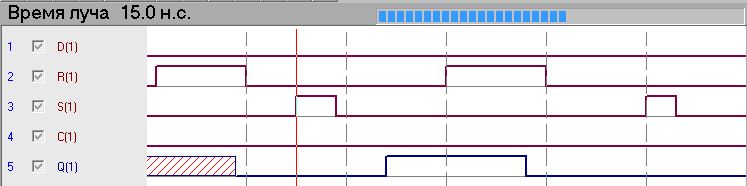
****

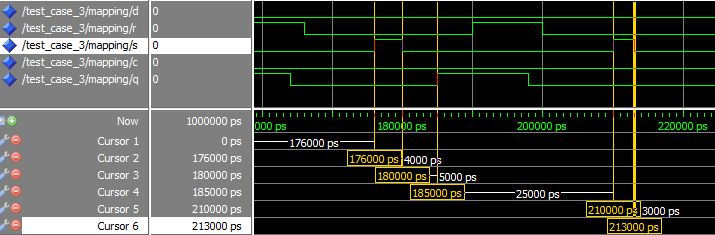
****

Время удержания R = 4ns

Время срабатывания R = 8 ns

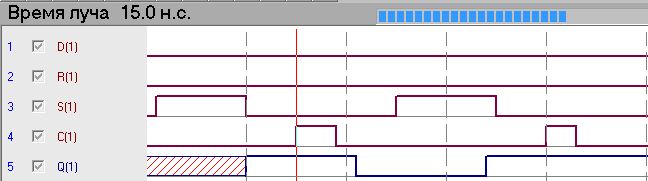
**Анализ времени удержания и время срабатывания сигнала S**

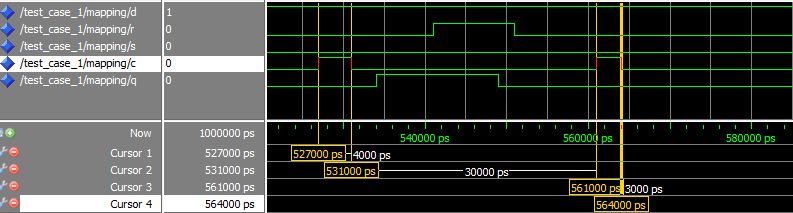
****



Время удержания S = 4ns

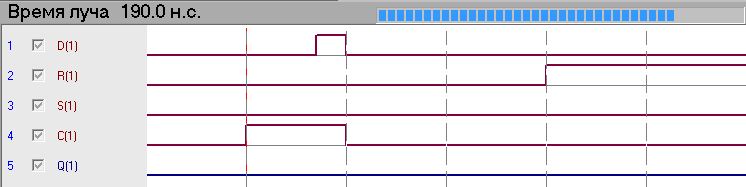
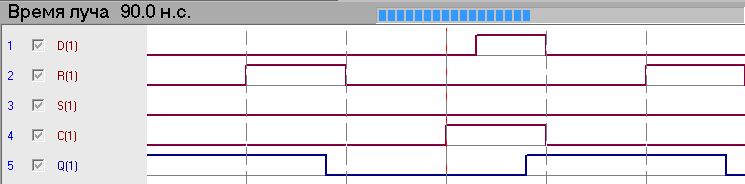
Время срабатывания S = 8 ns

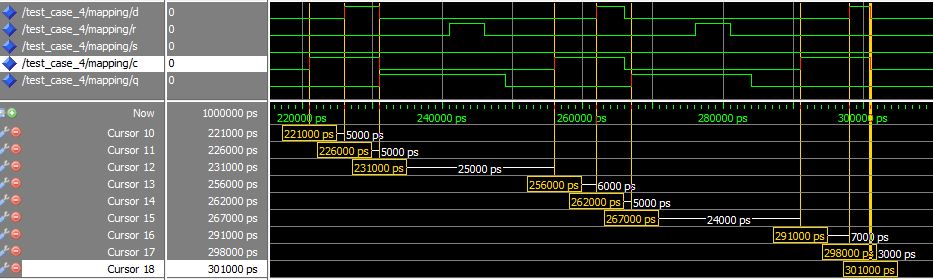
**Анализ времени удержания сигнала C** 



Время удержания С = 4ns

**Анализ времени удержания сигнала D**





Время удержания D = 4ns

Время предустановки С = 2ns

Время установки Q = 5ns

**Поведенческое описание схемы на языке VHDL**

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

ENTITY pov\_lr\_4 IS

PORT (

D : in STD\_LOGIC;

R : in STD\_LOGIC;

S : in STD\_LOGIC;

C : in STD\_LOGIC;

Q : out STD\_LOGIC

);

END pov\_lr\_4;

ARCHITECTURE Arch OF pov\_lr\_4 IS

constant reset\_time: time := 4 ns;

constant r\_retention\_time: time := 4 ns;

constant set\_time: time := 4 ns;

constant s\_retention\_time: time := 4 ns;

constant c\_preset\_time: time := 2 ns;

constant c\_retention\_time : time := 2 ns;

constant d\_retention\_time : time := 4 ns;

constant clk\_set\_time : time := 1 ns;

signal c\_preset\_event: STD\_LOGIC;

signal c\_event: STD\_LOGIC;

signal d\_event: STD\_LOGIC;

signal clk\_set: STD\_LOGIC;

signal clk\_set\_flag: STD\_LOGIC;

signal d\_temp: STD\_LOGIC;

signal r\_event: STD\_LOGIC;

signal reset: STD\_LOGIC;

signal r\_flag: STD\_LOGIC;

signal s\_event: STD\_LOGIC;

signal set: STD\_LOGIC;

signal s\_flag: STD\_LOGIC;

BEGIN

-- через 2 нс после фронта С устанавливается флаг проверки времени предустановки С

start\_c\_process: process(C)

begin

if C = '1' then

if c\_preset\_event = 'U' then

c\_preset\_event <= '1' after c\_preset\_time;

else

c\_preset\_event <= not c\_preset\_event after c\_preset\_time;

end if;

end if;

end process;

-- проверка предустановки С, через 2 нс после проверки устанавливается флаг проверки времени удержания С,

-- если С было установлено верно

c\_preset\_process: process(c\_preset\_event)

begin

if C'STABLE(c\_preset\_time) then

if c\_event = 'U' then

c\_event <= '1' after c\_retention\_time;

else

c\_event <= not c\_event after c\_retention\_time;

end if;

end if;

end process;

-- проверка времени удержания С

d\_preset\_process: process(c\_event)

begin

if C'STABLE(c\_retention\_time) then

if d\_event = 'U' then

d\_event <= '1' after d\_retention\_time - c\_retention\_time;

else

d\_event <= not d\_event after d\_retention\_time - c\_retention\_time;

end if;

end if;

end process;

-- прорверка времени удержания D

clk\_set\_process: process

begin

wait until D'STABLE(d\_retention\_time);

if C'DELAYED(1 ns) = '1' then

d\_temp <= D'DELAYED(2 ns);

clk\_set\_flag <= '1', '0' after clk\_set\_time;

if clk\_set = 'U' then

clk\_set <= '1' after clk\_set\_time;

else

clk\_set <= not clk\_set after clk\_set\_time;

end if;

end if;

end process;

-- проверка времени удержания S

start\_set\_process: process(S)

begin

if S = '1' then

if s\_event = 'U' then

s\_event <= '1' after s\_retention\_time;

else

s\_event <= not s\_event after s\_retention\_time;

end if;

end if;

end process;

-- проверка времени удержания S

set\_process: process(s\_event)

begin

if S'STABLE(s\_retention\_time) then

s\_flag <= '1', '0' after set\_time;

if set = 'U' then

set <= '1' after set\_time;

else

set <= not set after set\_time;

end if;

end if;

end process;

-- проверка времени удержания R

start\_reset\_process: process(R)

begin

if R = '1' then

if r\_event = 'U' then

r\_event <= '1' after r\_retention\_time;

else

r\_event <= not r\_event after r\_retention\_time;

end if;

end if;

end process;

-- проверка времени удержания R

reset\_process: process(r\_event)

begin

if R'STABLE(r\_retention\_time) then

r\_flag <= '1', '0' after reset\_time;

if reset = 'U' then

reset <= '1' after reset\_time;

else

reset <= not reset after reset\_time;

end if;

end if;

end process;

q\_process: process(reset, set, clk\_set)

begin

if r\_flag'event then

Q <= '0';

elsif s\_flag'event then

Q <= '1';

elsif clk\_set\_flag'event then

Q <= d\_temp;

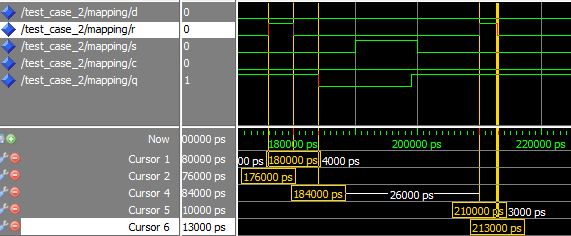
end if;

end process;

END Arch;

**Проверка поведенческого описания**

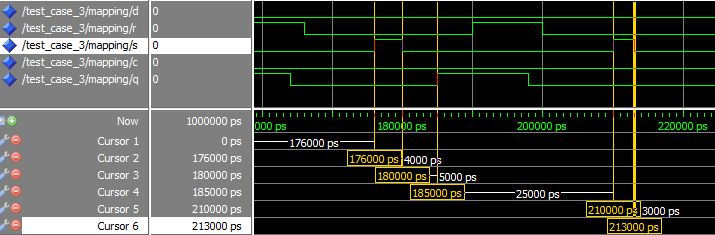
**Анализ времени удержания и время срабатывания сигнала R**

****

Время удержания S = 4ns

Время срабатывания S = 8 ns

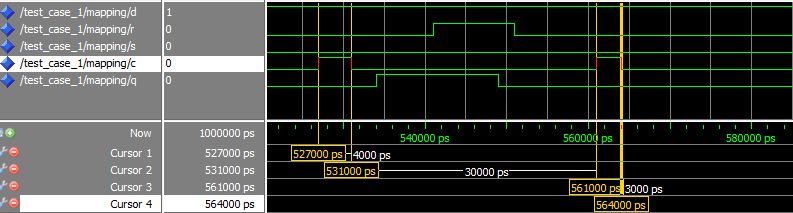
**Анализ времени удержания и время срабатывания сигнала S**



Время удержания S = 4ns

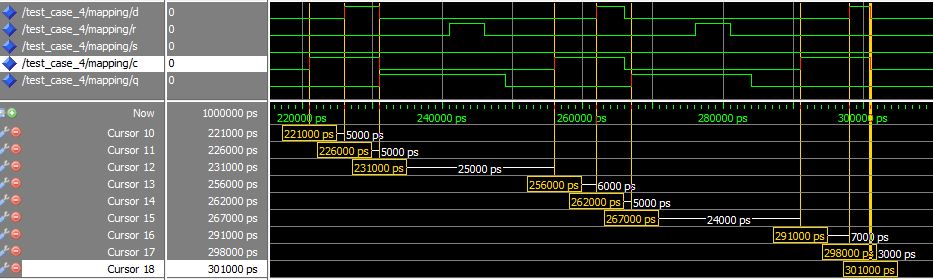
Время срабатывания S = 8 ns

**Анализ времени удержания сигнала C**



Время удержания С = 4ns

**Анализ времени удержания сигнала D**



Время удержания D = 4ns

Время установки Q = 5ns