Национальный исследовательский университет «МЭИ»

Институт информатики и вычислительной техники

Лабораторная работа №4

**Тема:** **ОРГАНИЗАЦИЯ МОДЕЛЬНОГО ЭКСПЕРИМЕНТА СРЕДСТВАМИ ЯЗЫКА VHDL.**

**Выполнил:**

**Студент** Швердяков М. А.

**Группа** А-06-21

**Вариант 10**

1) Построить схему "стенда изучения схемы", который состоит из схемы генератора, самой исследуемой схемы и наблюдателя.

2. Для заданного варианта схемы (Двухступенчатый JK-триггер c запрещающими связями)

построить:

а) модель этой схемы;

**entity** MS\_JK\_FF **is**

**generic**(T01: time := 5 ps; T10: time := 7 ps );

**port**(C, J, K: **in** bit; Q, NQ: **inout** bit);

**end** MS\_JK\_FF;

**architecture** arch\_MS\_JK\_FF **of** MS\_JK\_FF **is**

**signal** Q\_buf: bit;

**begin**

**process**(C, J, K)

**begin**

-- if (rising\_edge(C)) then

-- if (J = '0' and K = '1') then

-- Q\_buf <= '0';

-- elsif (J = '1' and K = '0') then

-- Q\_buf <= '1';

-- elsif (J = '1' and K = '1') then

-- Q\_buf <= (not Q\_buf);

-- end if;

-- end if;

**if** rising\_edge(C) **then**

**assert** **not** (C = '0' **or** (J = '0' **and** K = '0')) **report** ("storage mode") **severity** warning;

Q\_buf <= ((J **and** (**not** Q\_buf)) **or** ((**not** K) **and** Q\_buf));

**end** **if**;

**if** (Q\_buf = '0' **and** Q = '1') **then**

Q <= Q\_buf **after** T10;

NQ <= (**not** Q\_buf) **after** T01;

**elsif** (Q\_buf = '1' **and** Q = '0') **then**

Q <= Q\_buf **after** T01;

NQ <= (**not** Q\_buf) **after** T10;

**else**

Q <= Q\_buf;

NQ <= (**not** Q\_buf);

**end** **if**;

**end** **process**;

**end** arch\_MS\_JK\_FF;

б) модель внешней среды, включающей:

модель генератора входных сигналов, создающую в процессе

моделирования тестовую последовательность импульсов для данной

схемы;

**entity** generator **is**

**port** (C, J, K: **out** bit);

**end** generator;

**architecture** gener **of** generator **is**

**begin**

**process** **begin**

C <= '0'; J <= '1'; K <= '0'; **wait** **for** 100 ps;

C <= '1'; J <= '1'; K <= '0'; **wait** **for** 100 ps;

C <= '0'; J <= '1'; K <= '0'; **wait** **for** 100 ps;

C <= '1'; J <= '0'; K <= '1'; **wait** **for** 100 ps;

C <= '0'; J <= '0'; K <= '1'; **wait** **for** 100 ps;

C <= '1'; J <= '1'; K <= '0'; **wait** **for** 100 ps;

C <= '0'; J <= '1'; K <= '0'; **wait** **for** 100 ps;

C <= '1'; J <= '1'; K <= '0'; **wait** **for** 100 ps;

C <= '0'; J <= '1'; K <= '0'; **wait** **for** 100 ps;

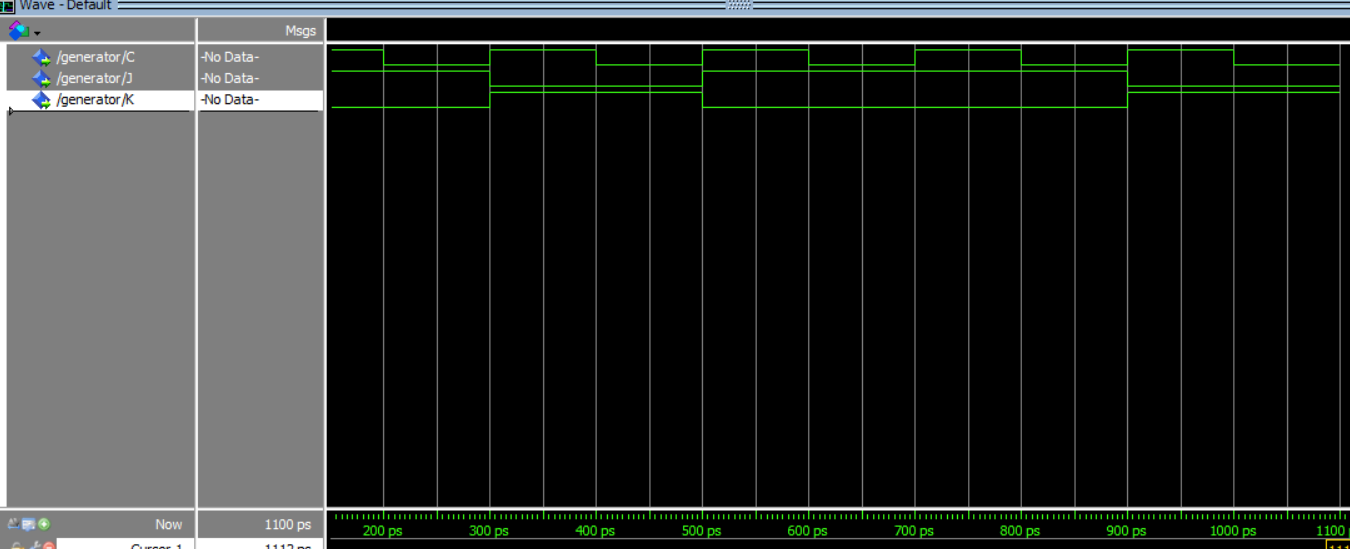
C <= '1'; J <= '0'; K <= '1'; **wait** **for** 100 ps;

C <= '0'; J <= '0'; K <= '1'; **wait** **for** 100 ps;

**wait**;

**end** **process**;

**end**;



г) модель наблюдателя - регистратора результатов эксперимента с

моделью.

**entity** observer **is**

**port** (C, J, K, Q, NQ: **in** bit;

condition: **out** bit\_vector(1 **to** 4);

outJ, outK, outQ, outNQ: **out** bit);

**end** observer;

**architecture** observer **of** observer **is**

**begin**

**process**(C, J, K, Q, NQ) **begin**

**if** (J = '1' **and** K = '1' **and** (C'EVENT **and** C = '0')) **then** condition <= "1000"; -- toggle

**elsif** (falling\_edge(C) **and** J = '0' **and** K = '1') **then** condition <= "0010"; -- reset

**elsif** (falling\_edge(C) **and** J = '1' **and** K = '0') **then** condition <= "0001"; -- set

**elsif** ((C'EVENT **and** C = '0') **or** (falling\_edge(C) **and** J = '0' **and** K = '0')) **then** condition <= "0100"; **end** **if**; -- storage

outJ <= J;

outK <= K;

outQ <= Q;

outNQ <= NQ;

**end** **process**;

**end**;

3. Занести построенные модели в рабочую библиотеку, для проверки их

работоспособности провести автономную отладку каждой модели.

4. Провести исследование (по аналогии с п.2 лабораторной работы №3)

разработанных в пп.2 а, б моделей как единой модели – "стенда изучения

схемы".

**entity** stend **is**

**port** (condition: **out** bit\_vector(1 **to** 4);

outJ, outK, outQ, outNQ: **out** bit);

**end** stend;

**architecture** stend **of** stend **is**

**component** generator

**port**(C, J, K:**out** bit);

**end** **component**;

**component** MS\_JK\_FF

**port** (C, J, K:**in** bit; Q,NQ:**inout** bit);

**end** **component**;

**component** observer

**port** (C, J, K, Q, NQ: **in** bit;

condition: **out** bit\_vector(1 **to** 4);

outJ, outK, outQ, outNQ: **out** bit);

**end** **component**;

**signal** C, J, K, Q, NQ: bit;

**begin**

Gen: generator **port** **map**(C, J, K);

Trigger: MS\_JK\_FF **port** **map**(C, J, K, Q, NQ);

Observ: observer **port** **map**(C, J, K, Q, NQ, condition, outJ, outK, outQ, outNQ);

**end**;



Вывод:

Смоделировал учебный стенд, понял, как создавать генератор сигнала, и как наглядно обозначать необходимые условия. Теперь понимаю как можно переделать стенды в лаборатории схемотехники на компьютерное моделирование.