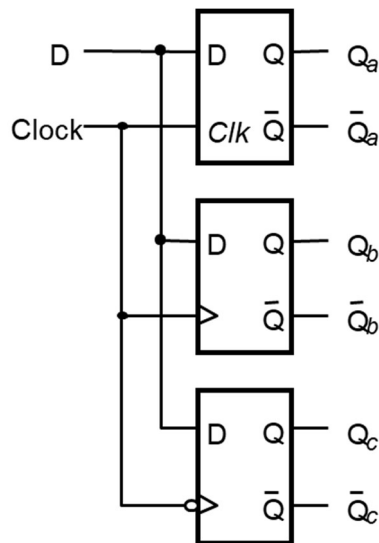


LABORATÓRIO 11

O objetivo deste laboratório observar os diversos comportamentos dos LATCH e dos FF tipo D.

ATIVIDADE 1

Escrever um código VHDL para verificar o funcionamento do Latch D com clock, do FF sensível à borda de subida e do FF sensível à borda de descida. OS CÓDIGOS DEVERÃO SER COMENTADOS. Usar a figura abaixo, como o projeto hierárquico superior:



A simulação deverá ter os sinais de entrada, como abaixo:



Comentar o resultado da simulação, com relação às saídas Qa, Qb, Qc.

Fazer cada elemento (Latch, FFD de subida, FFD de descida, como um componente, e instanciá-los em no arquivo superior hierarquicamente. Usar package.

ATIVIDADE 1.1: Usar PROCESS e IF ..THEN.. ELSEIF (entregar 7 arquivos vhd e o print da simulação).

ATIVIDADE 1.2: Usar WAIT UNTIL (entregar 7 arquivos vhd e o print da simulação).

OBSERVAÇÃO: Para as atividades 1.1 e 2.2, usar como base os slides 53 até 55 do capítulo 7 da teoria.

ATIVIDADE 2

Escrever um código VHDL para verificar o funcionamento do FF tipo D sensível à borda de subida o sinal de com RESET. OS CÓDIGOS DEVERÃO SER COMENTADOS.

ATIVIDADE 2.1: Verificar o funcionamento de um FF D sensível à borda de subida, com RESET, usando IF ..THEN..ELSE. (Entregar 3 arquivos vhd e o print da simulação). Comente.

ATIVIDADE 2.2: Verificar o funcionamento de um FF D sensível à borda de subida, com RESET, usando WAIT UNTL. (Entregar 3 arquivos vhd e o print da simulação). Comente.

OBSERVAÇÃO: Para as atividades 2.1 e 2.2 usar como base os slides 56 e 57 do capítulo 7 da teoria.

CÓDIGOS

Código para um latch D com clock

```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;

ENTITY latch IS
    PORT ( D, Clk : IN    STD_LOGIC ;
          Q       : OUT   STD_LOGIC );
END latch ;

ARCHITECTURE Behavior OF latch IS
BEGIN
    PROCESS (D, Clk )
    BEGIN
        IF Clk = '1' THEN
            Q <= D ;
        END IF ;
    END PROCESS ;
END Behavior ;
```

Código para um FF D sensível a borda de subida

```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;

ENTITY flipflop IS
    PORT ( D, Clock : IN    STD_LOGIC ;
          Q          : OUT  STD_LOGIC ) ;
END flipflop ;

ARCHITECTURE Behavior OF flipflop IS
BEGIN
    PROCESS (Clock )
    BEGIN
        IF Clock'EVENT AND Clock = '1' THEN
            Q <= D ;
        END IF ;
    END PROCESS ;
END Behavior ;
```

Código para um FF D sensível a borda de subida – WAIT...UNTIL

```
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY flipflop IS
    PORT ( D, Clock : IN    STD_LOGIC ;
          Q          : OUT  STD_LOGIC ) ;
END flipflop ;

ARCHITECTURE Behavior OF flipflop IS
BEGIN
    PROCESS
    BEGIN
        WAIT UNTIL Clock'EVENT AND Clock = '1' ;
        Q <= D ;
    END PROCESS ;
END Behavior ;
```

Código para um FF D sensível a borda de subida reset assíncrono

```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;

ENTITY flipflop IS
    PORT ( D, Resetn, Clock : IN      STD_LOGIC ;
          Q                  : OUT    STD_LOGIC) ;
END flipflop ;

ARCHITECTURE Behavior OF flipflop IS
BEGIN
    PROCESS ( Resetn, Clock )
    BEGIN
        IF Resetn = '0' THEN
            Q <= '0' ;
        ELSIF Clock'EVENT AND Clock = '1' THEN
            Q <= D ;
        END IF ;
    END PROCESS ;
END Behavior ;
```

Código para um FF D sensível a borda de subida reset síncrono

```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;

ENTITY flipflop IS
    PORT ( D, Resetn, Clock : IN      STD_LOGIC ;
          Q                  : OUT    STD_LOGIC) ;
END flipflop ;

ARCHITECTURE Behavior OF flipflop IS
BEGIN
    PROCESS
    BEGIN
        WAIT UNTIL Clock'EVENT AND Clock = '1' ;
        IF Resetn = '0' THEN
            Q <= '0' ;
        ELSE
            Q <= D ;
        END IF ;
    END PROCESS ;
END Behavior ;
```
