

-- **** STUDENT: 64000225.....	2
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	2
-- **** STUDENT: 64200100.....	5
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	5
-- **** STUDENT: 64200163.....	7
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	7
-- **** STUDENT: 64200238.....	9
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	9
-- **** STUDENT: 64200288.....	13
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	13
-- **** STUDENT: 64200296.....	15
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	15
-- **** STUDENT: 64200385.....	18
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	18
-- **** STUDENT: 64210113.....	21
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	21
-- **** STUDENT: 64210132.....	24
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	24
-- **** STUDENT: 64210290.....	27
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	27
-- **** STUDENT: 64210382.....	32
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	32
-- **** STUDENT: 64210386.....	34
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	34
-- **** STUDENT: 64210455.....	37
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	37
-- **** STUDENT: 64210457.....	40
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	40
-- **** STUDENT: 64240430.....	43
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	43
-- **** STUDENT: 64210384.....	46
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	46
-- **** STUDENT: 64210445.....	49
-- KOMENTARJI K OCENI NALOGE -- Matej Možek: Ni pripomb.....	49

```
-- *****
-- **** STUDENT: 64000225
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
```

```
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
```

```
-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
-- USE ieee.numeric_std.ALL;
```

```
ENTITY dff_tb IS
END dff_tb;
```

```
ARCHITECTURE behavior OF dff_tb IS
```

```
    -- Component Declaration for the Unit Under Test ( UUT )
```

```
    COMPONENT dff
    PORT(
        D : IN std_logic;
        clk : IN std_logic;
        nPRESET : IN std_logic;
        nCLEAR : IN std_logic;
        Q : OUT std_logic
    );
    END COMPONENT;
```

```
    FOR ALL: dff USE ENTITY WORK.dff( dff_fdc );
```

```
    -- Inputs
```

```
    signal D : std_logic := '0';
    signal clk : std_logic := '0';
    signal nPRESET : std_logic := '1';
    signal nCLEAR : std_logic := '1';
```

```
    -- Outputs
```

```

signal          Q : std_logic;

    -- Clock period definitions
constant  clk_period : time := 10 ns;

BEGIN

    -- Instantiate the Unit Under Test ( UUT )
    uut: dff PORT MAP (
    D => D,
    clk => clk,
    nPRESET => nPRESET,
    nCLEAR => nCLEAR,
    Q => Q
    );

    -- Clock process definitions
    clk_process :process
    begin
        clk    <= '0';
        wait for clk_period/2;
        clk    <= '1';
        wait for clk_period/2;
    end process;

    -- Stimulus process
    stim_proc: process
    begin
        -- hold reset state for 100 ns.
        wait for 100 ns;

        wait for clk_period*10;

        D    <= '1';
        wait for clk_period*3;
        D    <= '0';
        wait for clk_period*3;
        D    <= '1';
        wait for clk_period*1.7;
        nCLEAR <= '0';
    end process;

```

```
wait for clk_period/10;  
nCLEAR <= '1';
```

```
wait;  
end process;
```

```
END;
```

```

-- *****
-- **** STUDENT: 64200100
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY dff_tb IS
END dff_tb;

ARCHITECTURE behavior OF dff_tb IS

COMPONENT dff
PORT(
D : IN std_logic;
clk : IN std_logic;
nPRESET : IN std_logic;
nCLEAR : IN std_logic;
Q : OUT std_logic );
END COMPONENT;

FOR ALL: dff USE ENTITY WORK.dff( dff_fdc );

signal      D : std_logic := '0';
signal      clk : std_logic := '0';
signal      nPRESET : std_logic := '1';
signal      nCLEAR : std_logic := '1';
signal      Q : std_logic;
constant    clkP : time := 10 ns;

BEGIN
uut: dff PORT MAP (
D => D,
clk => clk,
nPRESET => nPRESET,
nCLEAR => nCLEAR,
Q => Q );

```

```
clk_process :process
begin
clk    <= '0';
wait for clkP/2;
clk    <= '1';
wait for clkP/2;
end process;
```

```
sproc: process
begin

wait for 100 ns;
wait for clkP*10;
D      <= '1';
wait for clkP*3;
D      <= '0';
wait for clkP*3;
D      <= '1';
wait for clkP*1.7;
nCLEAR <= '0';
wait for clkP/10;
nCLEAR <= '1';
    wait;
end process;
END;
```

```

-- *****
-- **** STUDENT: 64200163
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY dff_tb IS
END dff_tb;

ARCHITECTURE behavior OF dff_tb IS
    COMPONENT dff
        PORT(
            D : IN std_logic;
            clk : IN std_logic;
            nPRESET : IN std_logic;
            nCLEAR : IN std_logic;
            Q : OUT std_logic
        );
    END COMPONENT;

    signal      D : std_logic := '0';
    signal      clk : std_logic := '0';
    signal      nPRESET : std_logic := '1';
    signal      nCLEAR : std_logic := '1';
    signal      Q : std_logic;

    constant    clk_period : time := 10 ns;

    FOR ALL: dff USE ENTITY work.dff( dff_fdp );

BEGIN
    uut: dff PORT MAP(
        D => D,
        clk => clk,
        nPRESET => nPRESET,
        nCLEAR => nCLEAR,
        Q => Q
    );

```

```

);
clk_process : process
begin
    clk    <= '0';
    wait for clk_period/2;
    clk    <= '1';
    wait for clk_period/2;
end process;

stim_proc: process
begin
wait for clk_period*2.2;
    D      <= '1';
    wait for clk_period*5;
    D      <= '0';
    wait for clk_period*5;
    nPRESET <= '0';
    wait for clk_period*5;
    nPRESET <= '1';
    wait for clk_period*5;
    D      <= '0';
    wait for clk_period*5;
    D      <= '1';
    wait for clk_period*5;
    nCLEAR <= '0';
    wait for clk_period*5;
    nCLEAR <= '1';

wait;
end process;
END;

```



```

-- *****
-- **** STUDENT: 64200238
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --
-- Company:
-- Engineer:
--
-- Create Date: 20:46:53 12/20/2024
-- Design Name:
-- Module Name: C:/Users/Simon/Desktop/faks/NDS/4_domaca/FIFO/dff_tb.vhd
-- Project Name: FIFO
-- Target Device:
-- Tool versions:
-- Description:
--
-- VHDL Test Bench Created by ISE for module: dff
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
-- Notes:
-- This testbench has been automatically generated using types std_logic and
-- std_logic_vector for the ports of the unit under test. Xilinx recommends
-- that these types always be used for the top-level I/O of a design in order
-- to guarantee that the testbench will bind correctly to the post-implementation
-- simulation model.
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

```

```
-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
-- USE ieee.numeric_std.ALL;
```

```
ENTITY dff_tb IS
END dff_tb;
```

```
ARCHITECTURE behavior OF dff_tb IS
```

```
    -- Component Declaration for the Unit Under Test ( UUT )
```

```
    COMPONENT dff
    PORT(
        D : IN std_logic;
        clk : IN std_logic;
        nPRESET : IN std_logic;
        nCLEAR : IN std_logic;
        Q : OUT std_logic
    );
    END COMPONENT;
```

```
FOR ALL: dff USE ENTITY WORK.dff( dff_fdc );
```

```
    -- Inputs
```

```
    signal      D : std_logic := '0';
    signal      clk : std_logic := '0';
    signal      nPRESET : std_logic := '0';
    signal      nCLEAR : std_logic := '0';
```

```
    -- Outputs
```

```
    signal      Q : std_logic;
```

```
    -- Clock period definitions
```

```
    constant    clk_period : time := 10 ns;
```

```
BEGIN
```

```
    -- Instantiate the Unit Under Test ( UUT )
```

```
    uut: dff PORT MAP (
        D => D,
        clk => clk,
```

```

nPRESET => nPRESET,
nCLEAR => nCLEAR,
Q => Q
);

    -- Clock process definitions
clk_process :process
begin
    clk    <= '0';
    wait for clk_period/2;
    clk    <= '1';
    wait for clk_period/2;
end process;

    -- Stimulus process
stim_proc: process
begin
    -- hold reset state for 100 ns.
    wait for 100 ns;

    wait for clk_period*10;

    D    <= '1';
    wait for clk_period * 10;
    D    <= '0';
    wait for clk_period * 10;

    nCLEAR    <= '0';
    wait for clk_period * 10;
    D    <= '1';
    wait for clk_period * 10;
    D    <= '0';
    wait for clk_period * 10;
    nCLEAR    <= '1';
    wait for clk_period * 10;

    nPreset    <= '0';
    wait for clk_period * 10;
    D    <= '1';
    wait for clk_period * 10;

```

```

nPreset    <= '1';
wait for clk_period * 10;

D    <= '1';
nCLEAR    <= '0';
nPreset    <= '0';
wait for clk_period * 10;
D    <= '0';
nCLEAR    <= '1';
nPreset    <= '1';
wait for clk_period * 10;

D    <= '0';
nCLEAR    <= '1';
nPreset    <= '0';
wait for clk_period * 10;
D    <= '1';
nCLEAR    <= '0';
nPreset    <= '1';
wait for clk_period * 10;

nCLEAR    <= '0';      -- Reset active
nPreset    <= '0';      -- Preset active
wait for clk_period * 10;
nCLEAR    <= '1';      -- Reset deasserted
nPreset    <= '1';      -- Preset deasserted
wait for clk_period * 10;

D    <= '1';
nCLEAR    <= '1';
nPreset    <= '1';
wait for clk_period * 50;

wait;
end process;

END;
```

```

-- *****
-- **** STUDENT: 64200288
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY dff_tb IS
END dff_tb;

ARCHITECTURE behavior OF dff_tb IS

COMPONENT dff
PORT(
    D : IN std_logic;
    clk : IN std_logic;
    nPRESET : IN std_logic;
    nCLEAR : IN std_logic;
    Q : OUT std_logic );
END COMPONENT;

FOR ALL: dff USE ENTITY WORK.dff( dff_fdc );

signal      D : std_logic := '0';
signal      clk : std_logic := '0';
signal      nPRESET : std_logic := '1';
signal      nCLEAR : std_logic := '1';
signal      Q : std_logic;
constant    ClockPeriode : time := 10 ns;

BEGIN
uut: dff PORT MAP (
    D => D, clk => clk, nPRESET => nPRESET, nCLEAR => nCLEAR, Q => Q
);

clk_process :process
begin
clk    <= '0';

```

```
wait for ClockPeriode/2;  
clk    <= '1';  
wait for ClockPeriode/2;  
end process;  
  
sproc: process  
begin  
  
wait for 100 ns;  
wait for ClockPeriode*10;  
D      <= '1';  
wait for ClockPeriode*10;  
D      <= '0';  
wait for ClockPeriode*10;  
D      <= '1';  
wait for ClockPeriode*10;  
nCLEAR <= '0';  
wait for ClockPeriode/10;  
nCLEAR <= '1';  
    wait;  
end process;  
END;
```

```

-- *****
-- **** STUDENT: 64200296
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --
-- Company:
-- Engineer:
--
-- Create Date: 19:05:19 12/20/2024
-- Design Name:
-- Module Name: C:/Users/Tim/Downloads/FIFO/FIFO/dff_tb.vhd
-- Project Name: FIFO
-- Target Device:
-- Tool versions:
-- Description:
--
-- VHDL Test Bench Created by ISE for module: dff
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
-- Notes:
-- This testbench has been automatically generated using types std_logic and
-- std_logic_vector for the ports of the unit under test. Xilinx recommends
-- that these types always be used for the top-level I/O of a design in order
-- to guarantee that the testbench will bind correctly to the post-implementation
-- simulation model.
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

```

```
-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
-- USE ieee.numeric_std.ALL;
```

```
ENTITY dff_tb IS
END dff_tb;
```

```
ARCHITECTURE behavior OF dff_tb IS
```

```
    -- Component Declaration for the Unit Under Test ( UUT )
```

```
    COMPONENT dff
    PORT(
        D : IN std_logic;
        clk : IN std_logic;
        nPRESET : IN std_logic;
        nCLEAR : IN std_logic;
        Q : OUT std_logic
    );
    END COMPONENT;
```

```
    -- Inputs
```

```
    signal      D : std_logic := '0';
    signal      clk : std_logic := '0';
    signal      nPRESET : std_logic := '0';
    signal      nCLEAR : std_logic := '0';
```

```
    -- Outputs
```

```
    signal      Q : std_logic;
```

```
    -- Clock period definitions
```

```
    constant    clk_period : time := 10 ns;
```

```
    FOR ALL: dff USE ENTITY WORK.dff( dff_fdp );
```

```
BEGIN
```

```
    -- Instantiate the Unit Under Test ( UUT )
```

```
    uut: dff PORT MAP (
        D => D,
```



```

clk => clk,
nPRESET => nPRESET,
nCLEAR => nCLEAR,
Q => Q
);

    -- Clock process definitions
clk_process :process
begin
    clk    <= '0';
    wait for clk_period/2;
    clk    <= '1';
    wait for clk_period/2;
end process;

    -- Stimulus process
stim_proc: process
begin
    -- hold reset state for 100 ns.
    wait for 100 ns;

    wait for clk_period*10;

    -- insert stimulus here
    D    <= '1';
    wait for clk_period*3;
    D    <= '0';
    wait for clk_period*3;
    D    <= '1';
    wait for clk_period*1.7;
    nCLEAR <= '0';
    wait for clk_period/10;
    nCLEAR <= '1';

    wait;
end process;

END;

```

```

-- *****
-- **** STUDENT: 64200385
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --
-- Company:
-- Engineer:
--
-- Create Date: 18:38:03 12/21/2024
-- Design Name:
-- Module Name: /home/ise/NDV/DN/4 DN/dff/dff_tb.vhd
-- Project Name: dff
-- Target Device:
-- Tool versions:
-- Description:
--
-- VHDL Test Bench Created by ISE for module: dff
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
-- Notes:
-- This testbench has been automatically generated using types std_logic and
-- std_logic_vector for the ports of the unit under test. Xilinx recommends
-- that these types always be used for the top-level I/O of a design in order
-- to guarantee that the testbench will bind correctly to the post-implementation
-- simulation model.
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

```

```
-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
-- USE ieee.numeric_std.ALL;
```

```
ENTITY dff_tb IS
END dff_tb;
```

```
ARCHITECTURE behavior OF dff_tb IS
```

```
    -- Component Declaration for the Unit Under Test ( UUT )
```

```
    COMPONENT dff
    PORT(
        D : IN std_logic;
        clk : IN std_logic;
        nPRESET : IN std_logic;
        nCLEAR : IN std_logic;
        Q : OUT std_logic
    );
    END COMPONENT;
```

```
        for all: dff use entity work.dff( dff_fdc );
```

```
    -- Inputs
```

```
    signal      D : std_logic := '0';
    signal      clk : std_logic := '0';
    signal      nPRESET : std_logic := '0';
    signal      nCLEAR : std_logic := '0';
```

```
    -- Outputs
```

```
    signal      Q : std_logic;
```

```
    -- Clock period definitions
```

```
    constant    clk_period : time := 10 ns;
```

```
BEGIN
```

```
    -- Instantiate the Unit Under Test ( UUT )
```

```
    uut: dff PORT MAP (
        D => D,
```

```

clk => clk,
nPRESET => nPRESET,
nCLEAR => nCLEAR,
Q => Q
);

    -- Clock process definitions
clk_process :process
begin
    clk    <= '0';
    wait for clk_period/2;
    clk    <= '1';
    wait for clk_period/2;
end process;

    -- Stimulus process
stim_proc: process
begin
    -- hold reset state for 100 ns.
    wait for 100 ns;

    wait for clk_period*10;

    -- insert stimulus here
    D      <= '1';
    wait for clk_period;
    D      <= '0';
    wait for clk_period;
    D      <= '1';
    wait for clk_period;
    nCLEAR <= '0';
    wait for clk_period;
    nCLEAR <= '1';
    wait for clk_period;
    D      <= '0';

    wait;
end process;

END;

```

```

-- *****
-- **** STUDENT: 64210113
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****

```

```

LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
USE ieee.numeric_std.ALL;
LIBRARY work;

```

```

ENTITY dff_tb IS
END dff_tb;

```

```

ARCHITECTURE behavior OF dff_tb IS

```

```

    -- Component Declaration for the Unit Under Test ( UUT )

```

```

COMPONENT dff
PORT(
D : IN std_logic;
clk : IN std_logic;
nPRESET : IN std_logic;
nCLEAR : IN std_logic;
Q : OUT std_logic
);
END COMPONENT;

```

```

    -- Inputs

```

```

signal      D : std_logic := '0';
signal      clk : std_logic := '0';
signal      nPRESET : std_logic := '1';
signal      nCLEAR : std_logic := '1';

```

```

    -- Outputs

```

```

signal      Q : std_logic;

```

```

    -- Clock period definitions

```

```

constant    clk_period : time := 10 ns;

```

```
BEGIN
```

```
    -- Instantiate the Unit Under Test ( UUT )
```

```
    uut: dff PORT MAP (  
        D => D,  
        clk => clk,  
        nPRESET => nPRESET,  
        nCLEAR => nCLEAR,  
        Q => Q  
    );
```

```
    -- Clock process definitions
```

```
    clk_process :process  
    begin  
        clk    <= '0';  
        wait for clk_period/2;  
        clk    <= '1';  
        wait for clk_period/2;  
    end process;
```

```
    -- Stimulus process
```

```
    stim_proc: process  
    begin
```

```
        nPRESET    <= '0'; nCLEAR    <= '1'; D    <= '0';  
        WAIT FOR 20 ns;  
        nPRESET    <= '1';
```

```
        nCLEAR     <= '0';  
        WAIT FOR 20 ns;  
        nCLEAR     <= '1';
```

```
        D    <= '0';  
        WAIT FOR clk_period * 2;
```

```
        D    <= '1';  
        WAIT FOR clk_period * 2;
```

```
        D    <= '0';
```

```
WAIT FOR clk_period * 2;
```

```
D      <= '1';
```

```
WAIT FOR clk_period * 2;
```

```
nCLEAR      <= '0';
```

```
WAIT FOR 20 ns;
```

```
nCLEAR      <= '1';
```

```
WAIT;
```

```
end process;
```

```
END;
```

```

-- *****
-- **** STUDENT: 64210132
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --
-- Company:
-- Engineer:
--
-- Create Date: 00:29:59 12/23/2024
-- Design Name:
-- Module Name: C:/Users/blazk/Desktop/NDV/DN/fifo/FIF01/dff_tb.vhd
-- Project Name: FIF01
-- Target Device:
-- Tool versions:
-- Description:
--
-- VHDL Test Bench Created by ISE for module: dff
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
-- Notes:
-- This testbench has been automatically generated using types std_logic and
-- std_logic_vector for the ports of the unit under test. Xilinx recommends
-- that these types always be used for the top-level I/O of a design in order
-- to guarantee that the testbench will bind correctly to the post-implementation
-- simulation model.
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;
use ieee.numeric_std.all;

```



```

-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
-- USE ieee.numeric_std.ALL;

ENTITY dff_tb IS
END dff_tb;

ARCHITECTURE behavior OF dff_tb IS

    -- Component Declaration for the Unit Under Test ( UUT )

    COMPONENT dff
    PORT(
        D : IN std_logic;
        clk : IN std_logic;
        nPRESET : IN std_logic;
        nCLEAR : IN std_logic;
        Q : OUT std_logic
    );
    END COMPONENT;

    for all: dff use entity work.dff( dff_fdr );

    -- Inputs
    signal      D : std_logic := '0';
    signal      clk : std_logic := '0';
    signal      nPRESET : std_logic := '0';
    signal      nCLEAR : std_logic := '0';

    -- Outputs
    signal      Q : std_logic;

    -- Clock period definitions
    constant    clk_period : time := 10 ns;

    signal      test_sig : unsigned( 2 downto 0 ) := "000";

BEGIN

```

```

    -- Instantiate the Unit Under Test ( UUT )
uut: dff PORT MAP (
D => D,
clk => clk,
nPRESET => nPRESET,
nCLEAR => nCLEAR,
Q => Q
);

    -- Clock process definitions
clk_process :process
begin
    clk    <= '0';
    wait for clk_period/2;
    clk    <= '1';
    wait for clk_period/2;
end process;

    -- Stimulus process
stim_proc: process
begin

wait for clk_period*10.2;

    -- insert stimulus here
    for i in 0 to 7 loop
        test_sig    <= test_sig+1;
        D           <= test_sig( 2 );
        nCLEAR <= test_sig( 1 );
        nPRESET    <= test_sig( 0 );
        wait for clk_period*2;
    end loop;

wait;
end process;

END;

```

```

-- *****
-- **** STUDENT: 64210290
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --
-- Company:
-- Engineer:
--
-- Create Date: 09:00:33 12/10/2024
-- Design Name:
-- Module Name: E:/Master/NDV/DN/4DN/FIFO/FIFO/dff_tb.vhd
-- Project Name: FIFO
-- Target Device:
-- Tool versions:
-- Description:
--
-- VHDL Test Bench Created by ISE for module: dff
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
-- Notes:
-- This testbench has been automatically generated using types std_logic and
-- std_logic_vector for the ports of the unit under test. Xilinx recommends
-- that these types always be used for the top-level I/O of a design in order
-- to guarantee that the testbench will bind correctly to the post-implementation
-- simulation model.
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

```

```
-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
-- USE ieee.numeric_std.ALL;
```

```
ENTITY dff_tb IS
END dff_tb;
```

```
ARCHITECTURE behavior OF dff_tb IS
```

```
    -- Component Declaration for the Unit Under Test ( UUT )
```

```
    COMPONENT dff
    PORT(
        D : IN std_logic;
        clk : IN std_logic;
        nPRESET : IN std_logic;
        nCLEAR : IN std_logic;
        Q : OUT std_logic
    );
    END COMPONENT;
```

```
        for all: dff use entity work.dff( dff_syn_set_rst );
```

```
    -- Inputs
```

```
    signal      D : std_logic := '0';
    signal      clk : std_logic := '0';
    signal      nPRESET : std_logic := '0';
    signal      nCLEAR : std_logic := '0';
```

```
    -- Outputs
```

```
    signal      Q : std_logic;
```

```
    -- Clock period definitions
```

```
    constant    clk_period : time := 10 ns;
```

```
BEGIN
```

```
    -- Instantiate the Unit Under Test ( UUT )
```

```
    uut: dff PORT MAP (
        D => D,
```

```

clk => clk,
nPRESET => nPRESET,
nCLEAR => nCLEAR,
Q => Q
);

-- Clock process definitions
clk_process :process
begin
    clk    <= '0';
    wait for clk_period/2;
    clk    <= '1';
    wait for clk_period/2;
end process;

-- Stimulus process
stim_proc: process
begin
    -- hold reset state for 100 ns.
    wait for 100 ns;

    nCLEAR <= '1';
    nPRESET <= '1';

    -- synchronous data '1'
    D      <= '1';
    wait for clk_period;

    -- synchronous data '0'
    D      <= '0';
    wait for clk_period;

    -- asynchronous data '1'
    wait for clk_period/4;
    D      <= '1';
    wait for clk_period;

    -- asynchronous data '0'
    D      <= '0';
    wait for clk_period/4;

```

```

        wait for clk_period;

-- synchronous nPreset '0'
    nPRESET    <= '0';
    wait for clk_period;

-- synchronous nPRESET '1'
    nPRESET    <= '1';
    wait for clk_period;

-- asynchronous nPRESET '0'
    wait for clk_period/4;
    nPRESET    <= '0';
    wait for clk_period;

-- asynchronous nPreset '1'
    nPRESET    <= '1';
    wait for clk_period/4;
    wait for clk_period;

    D          <= '1';
    wait for clk_period;

-- synchronous nCLEAR '0'
    nCLEAR <= '0';
    wait for clk_period;

-- synchronous nCLEAR '1'
    nCLEAR <= '1';
    wait for clk_period;

-- asynchronous nCLEAR '0'
    wait for clk_period/4;
    nCLEAR <= '0';
    wait for clk_period;

-- asynchronous nCLEAR '1'
    nCLEAR <= '1';
    wait for clk_period/4;
    wait for clk_period;

```

```
wait for clk_period*10;  
    -- insert stimulus here  
wait;  
end process;  
END;
```

```

-- *****
-- **** STUDENT: 64210382
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY dff_tb IS
END dff_tb;

ARCHITECTURE behavior OF dff_tb IS

    COMPONENT dff
    PORT(
        D : IN std_logic;
        clk : IN std_logic;
        nPRESET : IN std_logic;
        nCLEAR : IN std_logic;
        Q : OUT std_logic
    );
    END COMPONENT;

    FOR ALL: dff USE ENTITY WORK.dff( dff_fdc );

    signal          D : std_logic := '0';
    signal          clk : std_logic := '0';
    signal          nPRESET : std_logic := '1';
    signal          nCLEAR : std_logic := '1';

    signal          Q : std_logic;

    constant        clk_period : time := 10 ns;

BEGIN

    uut: dff PORT MAP (
        D => D,

```



```

clk => clk,
nPRESET => nPRESET,
nCLEAR => nCLEAR,
Q => Q
);

clk_process :process
begin
    clk    <= '0';
    wait for clk_period/2;
    clk    <= '1';
    wait for clk_period/2;
end process;

stim_proc: process
begin
wait for 100 ns;

wait for clk_period*10;

    D      <= '1';
    wait for clk_period*3;
    D      <= '0';
    wait for clk_period*3;
    D      <= '1';
    wait for clk_period*1.7;
    nCLEAR <= '0';
    wait for clk_period/10;
    nCLEAR <= '1';

wait;
end process;

END;

```

```

-- *****
-- **** STUDENT: 64210386
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --
-- Company:
-- Engineer:
--
-- Create Date: 12:31:23 12/10/2024
-- Design Name:
-- Module Name: /home/ise/Xilinx_shared/domaca_naloga_4/dff_tb.vhd
-- Project Name: domaca_naloga_4
-- Target Device:
-- Tool versions:
-- Description:
--
-- VHDL Test Bench Created by ISE for module: dff
--
-- Dependencies:
--
-- Revision:
-- Revision 0.01 - File Created
-- Additional Comments:
--
-- Notes:
-- This testbench has been automatically generated using types std_logic and
-- std_logic_vector for the ports of the unit under test. Xilinx recommends
-- that these types always be used for the top-level I/O of a design in order
-- to guarantee that the testbench will bind correctly to the post-implementation
-- simulation model.
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --      --      --      --      --      --      --      --      --      --
--      --      --      --      --      --
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

```

```
-- Uncomment the following library declaration if using
-- arithmetic functions with Signed or Unsigned values
-- USE ieee.numeric_std.ALL;
```

```
ENTITY dff_tb IS
END dff_tb;
```

```
ARCHITECTURE behavior OF dff_tb IS
```

```
    -- Component Declaration for the Unit Under Test ( UUT )
```

```
    COMPONENT dff
    PORT(
        D : IN std_logic;
        clk : IN std_logic;
        nPRESET : IN std_logic;
        nCLEAR : IN std_logic;
        Q : OUT std_logic
    );
    END COMPONENT;
```

```
    for all: dff use entity WORK.dff( dff_fdc );
```

```
    -- Inputs
```

```
    signal      D : std_logic := '0';
    signal      clk : std_logic := '0';
    signal      nPRESET : std_logic := '1';
    signal      nCLEAR : std_logic := '1';
```

```
    -- Outputs
```

```
    signal      Q : std_logic;
```

```
    -- Clock period definitions
```

```
    constant    clk_period : time := 10 ns;
```

```
BEGIN
```

```
    -- Instantiate the Unit Under Test ( UUT )
```

```
    uut: dff PORT MAP (
        D => D,
```

```

clk => clk,
nPRESET => nPRESET,
nCLEAR => nCLEAR,
Q => Q
);

    -- Clock process definitions
clk_process :process
begin
    clk    <= '0';
    wait for clk_period/2;
    clk    <= '1';
    wait for clk_period/2;
end process;

    -- Stimulus process
stim_proc: process
begin
    -- hold reset state for 100 ns.
    wait for 100 ns;

    wait for clk_period*10;

    -- insert stimulus here

    D      <= '1';
    wait for clk_period*3;
    D      <= '0';
    wait for clk_period*3;
    D      <= '1';
    wait for clk_period*1.7;
    nCLEAR <= '0';
    wait for clk_period/5;
    nCLEAR <= '1';

    wait;
end process;

END;

```

```

-- *****
-- **** STUDENT: 64210455
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY dff_tb IS
END dff_tb;

ARCHITECTURE behavior OF dff_tb IS

    COMPONENT dff
    PORT(
        D : IN std_logic;
        clk : IN std_logic;
        nPRESET : IN std_logic;
        nCLEAR : IN std_logic;
        Q : OUT std_logic
    );
    END COMPONENT;

    FOR ALL: dff USE ENTITY WORK.dff( dff_fdc );

    -- Inputs
    signal D : std_logic := '0';
    signal clk : std_logic := '0';
    signal nPRESET : std_logic := '1';
    signal nCLEAR : std_logic := '1';

    -- Outputs
    signal Q : std_logic;

    -- Clock period definitions
    constant clk_period : time := 10 ns;

BEGIN

```

```
    -- Instantiate the Unit Under Test ( UUT )
```

```
uut: dff PORT MAP (  
D => D,  
clk => clk,  
nPRESET => nPRESET,  
nCLEAR => nCLEAR,  
Q => Q  
);
```

```
    -- Clock process definitions
```

```
clk_process :process  
begin  
    clk    <= '0';  
    wait for clk_period/2;  
    clk    <= '1';  
    wait for clk_period/2;  
end process;
```

```
    -- Stimulus process
```

```
stim_proc: process  
begin  
    -- hold reset state for 100 ns.  
    wait for 100 ns;  
  
    wait for clk_period*10;
```

```
    -- insert stimulus here
```

```
        D    <= '1';  
        wait for clk_period*3;  
        D    <= '0';  
        wait for clk_period*3;  
        D    <= '1';  
        wait for clk_period*1.7;  
        nCLEAR <= '0';  
        wait for clk_period/10;  
        nCLEAR <= '1';
```

```
wait;  
end process;
```

END;

```

-- *****
-- **** STUDENT: 64210457
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY dff_tb IS
END dff_tb;

ARCHITECTURE behavior OF dff_tb IS

    COMPONENT dff
    PORT(
        D : IN std_logic;
        clk : IN std_logic;
        nPRESET : IN std_logic;
        nCLEAR : IN std_logic;
        Q : OUT std_logic
    );
    END COMPONENT;

        FOR ALL: dff USE ENTITY WORK.dff( dff_fdc );

-- Inputs
signal      D : std_logic := '0';
signal      clk : std_logic := '0';
signal      nPRESET : std_logic := '1';
signal      nCLEAR : std_logic := '1';

-- Outputs
signal      Q : std_logic;

-- Clock period definitions
constant    clk_period : time := 10 ns;

BEGIN

```



```
    -- Instantiate the Unit Under Test ( UUT )
```

```
uut: dff PORT MAP (  
D => D,  
clk => clk,  
nPRESET => nPRESET,  
nCLEAR => nCLEAR,  
Q => Q  
);
```

```
    -- Clock process definitions
```

```
clk_process :process  
begin  
    clk    <= '0';  
    wait for clk_period/2;  
    clk    <= '1';  
    wait for clk_period/2;  
end process;
```

```
    -- Stimulus process
```

```
stim_proc: process  
begin  
    -- hold reset state for 100 ns.  
    wait for 100 ns;  
  
    wait for clk_period*10;
```

```
    -- insert stimulus here
```

```
        D    <= '1';  
        wait for clk_period*3;  
        D    <= '0';  
        wait for clk_period*3;  
        D    <= '1';  
        wait for clk_period*1.7;  
        nCLEAR <= '0';  
        wait for clk_period/10;  
        nCLEAR <= '1';
```

```
wait;  
end process;
```

END;

```

-- *****
-- **** STUDENT: 64240430
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY dff_tb IS
END dff_tb;

ARCHITECTURE behavior OF dff_tb IS

    COMPONENT dff
    PORT(
        D : IN std_logic;
        clk : IN std_logic;
        nPRESET : IN std_logic;
        nCLEAR : IN std_logic;
        Q : OUT std_logic
    );
    END COMPONENT;

    for all: dff use entity work.dff( dff_fdc );

    -- Inputs
    signal D : std_logic := '0';
    -- signal clk : std_logic := '0';
    signal nPRESET : std_logic := '1';
    signal nCLEAR : std_logic := '1';

    -- Outputs
    signal Q : std_logic;

    -- Clock period definitions
    signal clock : std_logic;
    constant clock_period : time := 100 ns;

BEGIN

```

```

    -- Instantiate the Unit Under Test ( UUT )
uut: dff PORT MAP (
D => D,
clk => clock,
nPRESET => nPRESET,
nCLEAR => nCLEAR,
Q => Q
);

    -- Clock process definitions
clock_process :process
begin
    clock <= '0';
    wait for clock_period/2;
    clock <= '1';
    wait for clock_period/2;
end process;

    -- Stimulus process
stim_proc: process
begin

    -- prvi slucaj kada je nCLEAR == '0', postavlja q=0
    -- bez obzira kakav je clock, da li je ulazna ili silazna ivica, rezultat q=0
    nCLEAR <= '0';

    D    <= '0';
    wait for clock_period;
    D    <= '1';
    wait for clock_period;

    -- drugi slucaj kada je nCLEAR == '1'
    -- u ovom slucaju vrijednost Q = D
    -- na dijagramu vidimo kada je D = 1 i uzlazna ivica clocka, kao rezultat imamo      q=1
nCLEAR    <= '1';

    D    <= '0';
    wait for clock_period;
    D    <= '1';

```

```
        wait for clock_period;  
        wait;  
    end process;  
END;
```

```

-- *****
-- **** STUDENT: 64210384
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY diff_tb IS
END diff_tb;

ARCHITECTURE behavior OF diff_tb IS

    COMPONENT dff
    PORT(
        D : IN std_logic;
        clk : IN std_logic;
        nPRESET : IN std_logic;
        nCLEAR : IN std_logic;
        Q : OUT std_logic
    );
    END COMPONENT;

        FOR ALL: dff USE ENTITY WORK.dff( dff_fdc );

    -- Inputs
    signal      D : std_logic := '0';
    signal      clk : std_logic := '0';
    signal      nPRESET : std_logic := '1';
    signal      nCLEAR : std_logic := '1';

    -- Outputs
    signal      Q : std_logic;

    -- Clock period definitions
    constant    clk_period : time := 10 ns;

BEGIN

```

```
    -- Instantiate the Unit Under Test ( UUT )
```

```
uut: dff PORT MAP (  
D => D,  
clk => clk,  
nPRESET => nPRESET,  
nCLEAR => nCLEAR,  
Q => Q  
);
```

```
    -- Clock process definitions
```

```
clk_process :process  
begin  
    clk    <= '0';  
    wait for clk_period/2;  
    clk    <= '1';  
    wait for clk_period/2;  
end process;
```

```
    -- Stimulus process
```

```
stim_proc: process  
begin  
    -- hold reset state for 100 ns.  
    wait for 100 ns;  
  
    wait for clk_period*10;
```

```
    -- insert stimulus here
```

```
    D    <= '1';  
    wait for clk_period*3;  
    D    <= '0';  
    wait for clk_period*3;  
    D    <= '1';  
    wait for clk_period*2;  
    nCLEAR <= '0';  
    wait for clk_period/10;  
    nCLEAR <= '1';
```

```
wait;  
end process;
```

END;


```

-- *****
-- **** STUDENT: 64210445
-- *****
-- KOMENTARJI K OCENI NALOGE
-- Matej Možek: Ni pripomb
-- *****
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY diff_tb IS
END diff_tb;

ARCHITECTURE behavior OF diff_tb IS

    COMPONENT dff
    PORT(
        D : IN std_logic;
        clk : IN std_logic;
        nPRESET : IN std_logic;
        nCLEAR : IN std_logic;
        Q : OUT std_logic
    );
    END COMPONENT;

    FOR ALL: dff USE ENTITY WORK.dff( dff_fdr );

    -- Inputs
    signal D : std_logic := '0';
    signal clk : std_logic := '0';
    signal nPRESET : std_logic := '0';
    signal nCLEAR : std_logic := '0';

    -- Outputs
    signal Q : std_logic;

    -- Clock period definitions
    constant clk_period : time := 10 ns;

BEGIN

```

```
    -- Instantiate the Unit Under Test ( UUT )
```

```
uut: dff PORT MAP (  
D => D,  
clk => clk,  
nPRESET => nPRESET,  
nCLEAR => nCLEAR,  
Q => Q  
);
```

```
    -- Clock process definitions
```

```
clk_process :process  
begin  
    clk    <= '0';  
    wait for clk_period/2;  
    clk    <= '1';  
    wait for clk_period/2;  
end process;
```

```
    -- Stimulus process
```

```
stim_proc: process  
begin  
    -- hold reset state for 100 ns.  
    wait for 100 ns;
```

```
    -- insert stimulus here
```

```
        nCLEAR <= '1';  
        wait for clk_period;
```

```
        D      <= '1';  
        wait for clk_period;
```

```
        D      <= '0';  
        wait for clk_period;
```

```
        D      <= '1';  
        wait for clk_period;
```

```
        nCLEAR <= '0';  
        wait for clk_period;
```

```
wait;  
end process;  
  
END;
```

