******

**JK Lakshmipat University**

[**Computer**](https://canvas.instructure.com/courses/8496533) **Organization and Architecture**

**ASSIGNMENT 3**

**Name – Indrajit Roy**

**Section – B**

**Roll No.- 2023BTECH037**

**Date of Submission – 15th November, 2024**

**Submitted To – DR. Pranab Roy Sir**

**D Flip Fop**

***● VHDL Code:***

***----------------------------------------------------------------------------------***

***-- Company:***

***-- Engineer:***

***--***

***-- Create Date: 17:21:24 10/15/2024***

***-- Design Name:***

***-- Module Name: d\_ff - arch\_diff***

***-- Project Name:***

***-- Target Devices:***

***-- Tool versions:***

***-- Description:***

***--***

***-- Dependencies:***

***--***

***-- Revision:***

***-- Revision 0.01 - File Created***

***-- Additional Comments:***

***--***

***----------------------------------------------------------------------------------***

***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;***

***-- Uncomment the following library declaration if instantiating***

***-- any Xilinx primitives in this code.***

***--library UNISIM;***

***--use UNISIM.VComponents.all;***

***entity d\_ff is***

***port ( d : in std\_logic;***

***clk : in std\_logic;***

***enable : in std\_logic;***

***reset : in std\_logic;***

***q : out std\_logic);***

***end d\_ff;***

***architecture arch\_dff of d\_ff is***

***begin***

***process (clk,reset)***

***begin***

***if reset ='1' then***

***q <='0';***

***elsif rising\_edge(clk) then***

***if enable ='1' then***

***q <= d;***

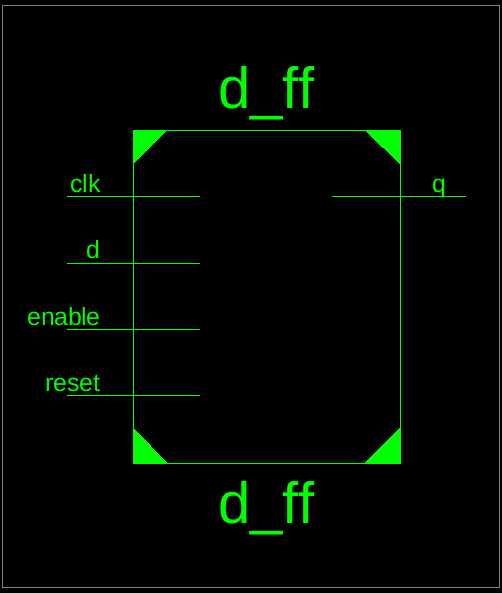
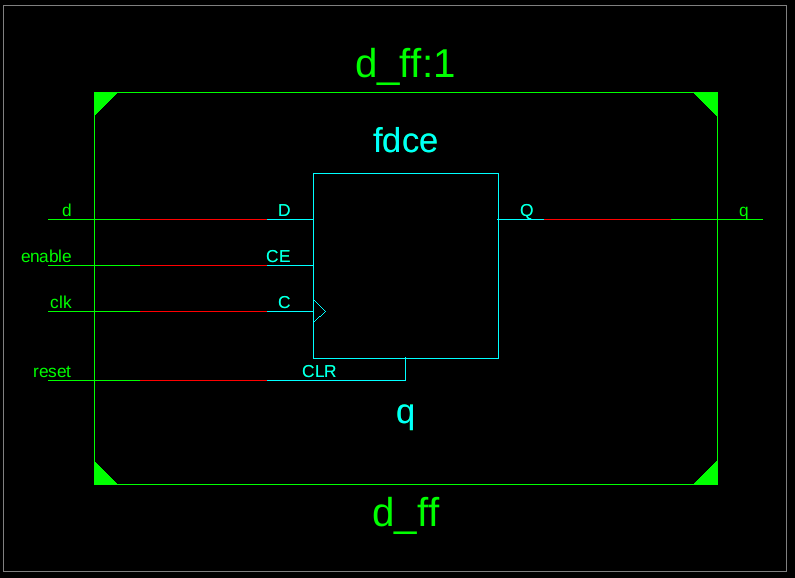
***end if;***

***end if;***

***end process;***

***end arch\_dff;***

***● RTL Diagram:***

***● Testbench Code:***

***--------------------------------------------------------------------------------***

***-- Company:***

***-- Engineer:***

***-- Create Date: 17:40:34 10/15/2024***

***-- Design Name:***

***-- Module Name: /home/ise/Assignment3/d\_ff\_tb.vhd***

***-- Project Name: Assignment3***

***-- Target Device:***

***-- Tool versions:***

***-- Description:***

***-- VHDL Test Bench Created by ISE for module: d\_ff***

***-- 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 d\_ff\_tb IS***

***END d\_ff\_tb;***

***ARCHITECTURE behavior OF d\_ff\_tb IS***

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

***COMPONENT d\_ff***

***PORT(***

***d : IN std\_logic;***

***clk : IN std\_logic;***

***enable : IN std\_logic;***

***reset : IN std\_logic;***

***q : OUT std\_logic***

***);***

***END COMPONENT;***

***--Inputs***

***signal d : std\_logic := '0';***

***signal clk : std\_logic := '0';***

***signal enable : std\_logic := '0';***

***signal reset : std\_logic := '0';***

***--Outputs***

***signal q : std\_logic;***

***BEGIN***

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

***uut: d\_ff PORT MAP (***

***d => d,***

***clk => clk,***

***enable => enable,***

***reset => reset,***

***q => q***

***);***

***-- Clock process definitions***

***clk\_process :process***

***begin***

***clk <= '0';***

***wait for 5 ns;***

***clk <= '1';***

***wait for 5 ns;***

***end process;***

***-- Stimulus process***

***stim\_proc: process***

***begin***

***d <= '0';***

***enable <= '0';***

***reset <= '0';***

***wait for 20 ns;***

***reset <= '1';***

***wait for 10 ns;***

***reset <= '0';***

***wait for 10 ns;***

***d <= '1';***

***enable <= '1';***

***wait for 20 ns;***

***d <= '0';***

***enable <= '0';***

***wait for 20 ns;***

***reset <= '1';***

***wait for 10 ns;***

***reset <= '0';***

***wait for 10 ns;***

***enable <= '1';***

***d <= '1';***

***wait for 20 ns;***

***enable <= '0';***

***d <= '0';***

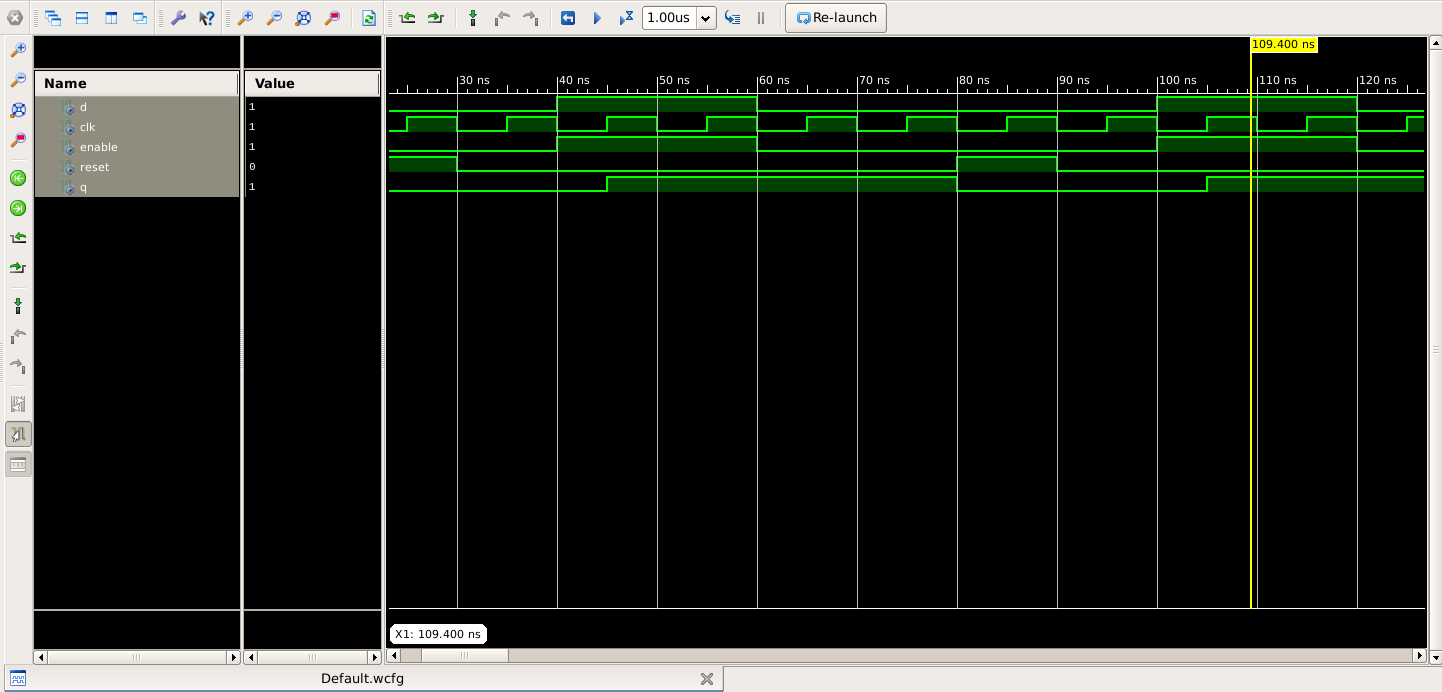
***wait for 20 ns;***

***wait;***

***end process;***

***END;***

***● Waveform:***

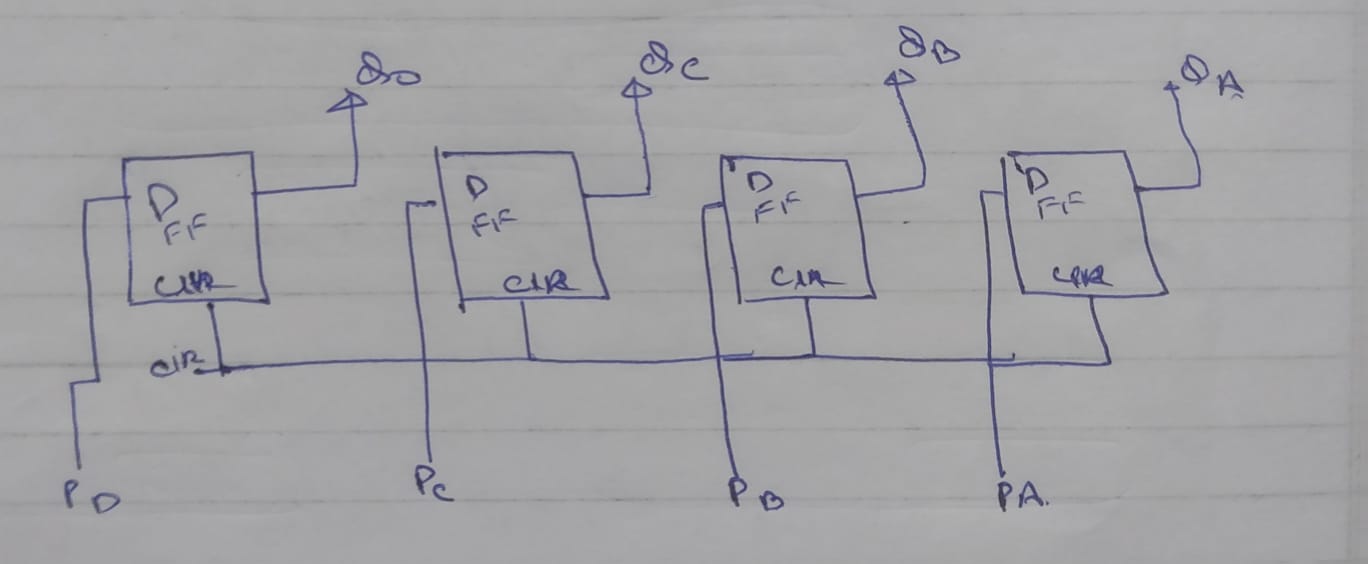


**Question 1**

**Develop a VHDL model to design a four bit Parallel in parallel out register using D flip flop components. Use an enable and reset input pin to enable and reset the registers.**

**Aim :** **To design and implement a 4-bit Parallel-In Parallel-Out (PIPO) register using D flip-flop components in VHDL, with functionality to enable data loading and reset the register to its initial state.**

**Block Diagram :**

***● VHDL Code:***

***----------------------------------------------------------------------------------***

***----------------------------------------------------------------------------------***

***-- Company:***

***-- Engineer:***

***--***

***-- Create Date: 17:18:25 10/15/2024***

***-- Design Name:***

***-- Module Name: pipo - Behavioral***

***-- Project Name:***

***-- Target Devices:***

***-- Tool versions:***

***-- Description:***

***--***

***-- Dependencies:***

***--***

***-- Revision:***

***-- Revision 0.01 - File Created***

***-- Additional Comments:***

***--***

***----------------------------------------------------------------------------------***

***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;***

***-- Uncomment the following library declaration if instantiating***

***-- any Xilinx primitives in this code.***

***--library UNISIM;***

***--use UNISIM.VComponents.all;***

***entity pipo is***

***port(***

***clk:in std\_logic;***

***enable:in std\_logic;***

***reset:in std\_logic;***

***data\_in:in std\_logic\_vector(3 downto 0);***

***data\_out:out std\_logic\_vector (3 downto 0)***

***);***

***end pipo;***

***architecture Behavioral of pipo is***

***component d\_ff***

***port ( d : in std\_logic;***

***clk : in std\_logic;***

***enable : in std\_logic;***

***reset : in std\_logic;***

***q : out std\_logic);***

***end component;***

***begin***

***step1: d\_ff Port map (d=> data\_in(0),clk=> clk,enable => enable,reset => reset,q=>data\_out(0));***

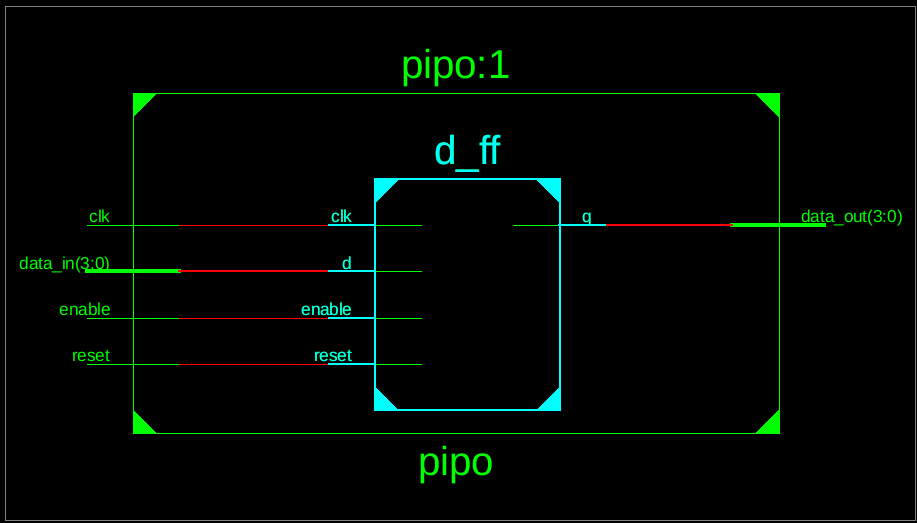
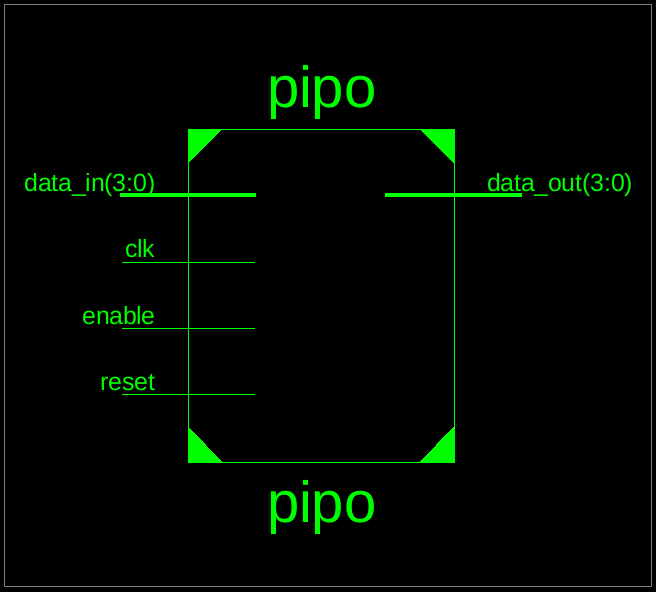
***step2: d\_ff Port map (d=> data\_in(1),clk=> clk,enable => enable,reset => reset,q=>data\_out(1));***

***step3: d\_ff Port map (d=> data\_in(2),clk=> clk,enable => enable,reset => reset,q=>data\_out(2));***

***step4: d\_ff Port map (d=> data\_in(3),clk=> clk,enable => enable,reset => reset,q=>data\_out(3));***

***end Behavioral;***

***● RTL Diagram:***



***● Testbench Code:***

***--------------------------------------------------------------------------------***

***-- Company:***

***-- Engineer:***

***-- Create Date: 17:31:19 10/15/2024***

***-- Design Name:***

***-- Module Name: /home/ise/Assignment3/pipo\_tb.vhd***

***-- Project Name: Assignment3***

***-- Target Device:***

***-- Tool versions:***

***-- Description:***

***--***

***-- VHDL Test Bench Created by ISE for module: pipo***

***--***

***-- 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 pipo\_tb IS***

***END pipo\_tb;***

***ARCHITECTURE behavior OF pipo\_tb IS***

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

***COMPONENT pipo***

***PORT(***

***clk : IN std\_logic;***

***enable : IN std\_logic;***

***reset : IN std\_logic;***

***data\_in : IN std\_logic\_vector(3 downto 0);***

***data\_out : OUT std\_logic\_vector(3 downto 0)***

***);***

***END COMPONENT;***

***--Inputs***

***signal clk : std\_logic := '0';***

***signal enable : std\_logic := '0';***

***signal reset : std\_logic := '0';***

***signal data\_in : std\_logic\_vector(3 downto 0) := (others => '0');***

***--Outputs***

***signal data\_out : std\_logic\_vector(3 downto 0);***

***BEGIN***

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

***uut: pipo PORT MAP (***

***clk => clk,***

***enable => enable,***

***reset => reset,***

***data\_in => data\_in,***

***data\_out => data\_out***

***);***

***-- Clock process definitions***

***clk\_process :process***

***begin***

***while true loop***

***clk <= '0';***

***wait for 1 ns;***

***clk <= '1';***

***wait for 1 ns;***

***end loop;***

***end process;***

***-- Stimulus process***

***stim\_proc: process***

***begin***

***reset <= '1';***

***wait for 200 ns;***

***reset <= '0';***

***wait for 200 ns;***

***enable <= '1';***

***data\_in <= "1010";***

***wait for 200 ns;***

***data\_in <= "1100";***

***wait for 200 ns;***

***enable <= '0';***

***wait for 200 ns;***

***reset <= '1';***

***wait for 200 ns;***

***reset <= '0';***

***wait for 200 ns;***

***enable <= '1';***

***data\_in <= "0110";***

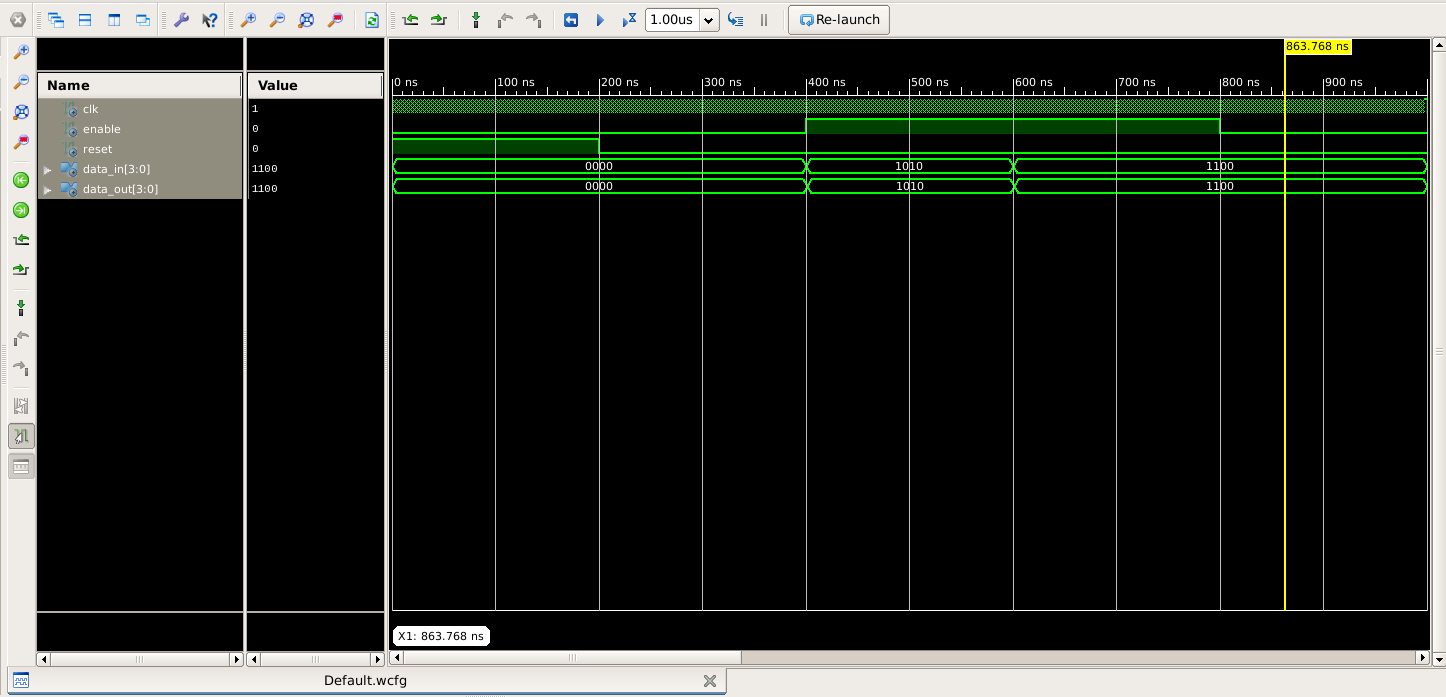
***wait for 200 ns;***

***wait;***

***end process;***

***END;***

***● Waveform:***

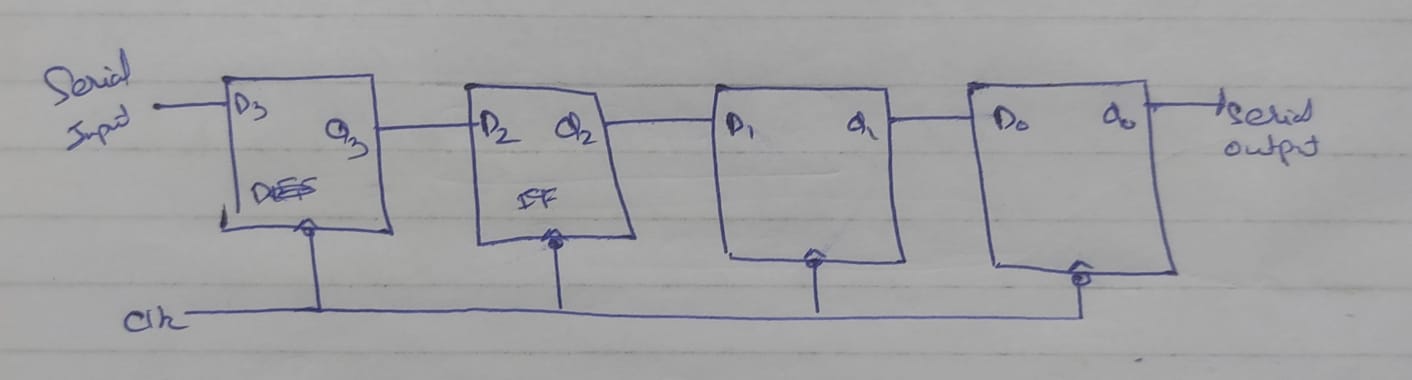


**Question 2**

**Develop a VHDL model to design a four bit Serial in Serial out register using D flip flop components. Use an enable and reset input pin to enable and reset the registers.**

**Aim : To develop a VHDL model for designing a 4-bit Serial-In Serial-Out (SISO) shift register using D flip-flop components. The design will include an enable input to control data shifting and a reset input to clear the register contents**

**Circuit Diagram :**



***● VHDL Code:***

***----------------------------------------------------------------------------------***

***-- Company:***

***-- Engineer:***

***--***

***-- Create Date: 18:07:21 10/15/2024***

***-- Design Name:***

***-- Module Name: siso - Behavioral***

***-- Project Name:***

***-- Target Devices:***

***-- Tool versions:***

***-- Description:***

***--***

***-- Dependencies:***

***--***

***-- Revision:***

***-- Revision 0.01 - File Created***

***-- Additional Comments:***

***--***

***----------------------------------------------------------------------------------***

***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;***

***-- Uncomment the following library declaration if instantiating***

***-- any Xilinx primitives in this code.***

***--library UNISIM;***

***--use UNISIM.VComponents.all;***

***entity siso is***

***port(***

***clk:in std\_logic;***

***enable:in std\_logic;***

***reset:in std\_logic;***

***serial\_in:in std\_logic;***

***serial\_out:out std\_logic***

***);***

***end siso;***

***architecture Behavioral of siso is***

***component d\_ff is***

***port ( d : in std\_logic;***

***clk : in std\_logic;***

***enable : in std\_logic;***

***reset : in std\_logic;***

***q : out std\_logic);***

***end component;***

***signal q0,q1,q2,q3:std\_logic;***

***begin***

***step1:d\_ff port map(d=>serial\_in,clk=>clk,enable=>enable,reset=>reset,q=>q3);***

***step2:d\_ff port map(d=>q3,clk=>clk,enable=>enable,reset=>reset,q=>q2);***

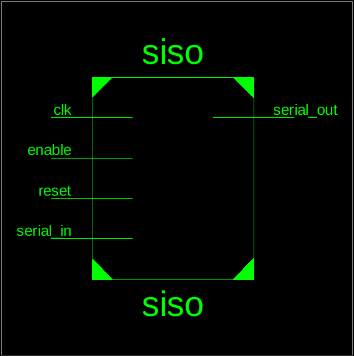
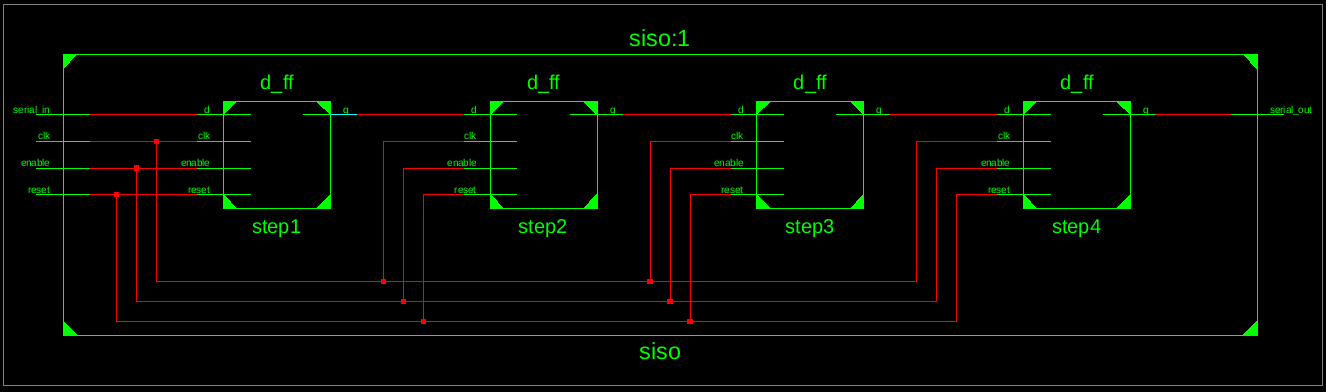
***step3:d\_ff port map(d=>q2,clk=>clk,enable=>enable,reset=>reset,q=>q1);***

***step4:d\_ff port map(d=>q1,clk=>clk,enable=>enable,reset=>reset,q=>q0);***

***serial\_out<=q0;***

***end Behavioral;***

***● RTL Diagram:***

***● Testbench Code:***

***--------------------------------------------------------------------------------***

***-- Company:***

***-- Engineer:***

***--***

***-- Create Date: 18:19:38 10/15/2024***

***-- Design Name:***

***-- Module Name: /home/ise/Assignment3/siso\_tb.vhd***

***-- Project Name: Assignment3***

***-- Target Device:***

***-- Tool versions:***

***-- Description:***

***--***

***-- VHDL Test Bench Created by ISE for module: siso***

***--***

***-- 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 siso\_tb IS***

***END siso\_tb;***

***ARCHITECTURE behavior OF siso\_tb IS***

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

***COMPONENT siso***

***PORT(***

***clk : IN std\_logic;***

***enable : IN std\_logic;***

***reset : IN std\_logic;***

***serial\_in : IN std\_logic;***

***serial\_out : OUT std\_logic***

***);***

***END COMPONENT;***

***--Inputs***

***signal clk : std\_logic := '0';***

***signal enable : std\_logic := '0';***

***signal reset : std\_logic := '0';***

***signal serial\_in : std\_logic := '0';***

***--Outputs***

***signal serial\_out : std\_logic;***

***BEGIN***

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

***uut: siso PORT MAP (***

***clk => clk,***

***enable => enable,***

***reset => reset,***

***serial\_in => serial\_in,***

***serial\_out => serial\_out***

***);***

***-- Clock process definitions***

***clk\_process :process***

***begin***

***clk <= '0';***

***wait for 50 ns;***

***clk <= '1';***

***wait for 50 ns;***

***end process;***

***-- Stimulus process***

***stim\_proc: process***

***begin***

***reset <= '1';***

***wait for 10 ns;***

***reset <= '0';***

***enable <= '1';***

***serial\_in <= '1'; wait for 10 ns;***

***serial\_in <= '0'; wait for 10 ns;***

***serial\_in <= '0'; wait for 10 ns;***

***serial\_in <= '1'; wait for 10 ns;***

***enable <= '0';***

***wait for 200 ns;***

***reset <= '1';***

***wait for 100 ns;***

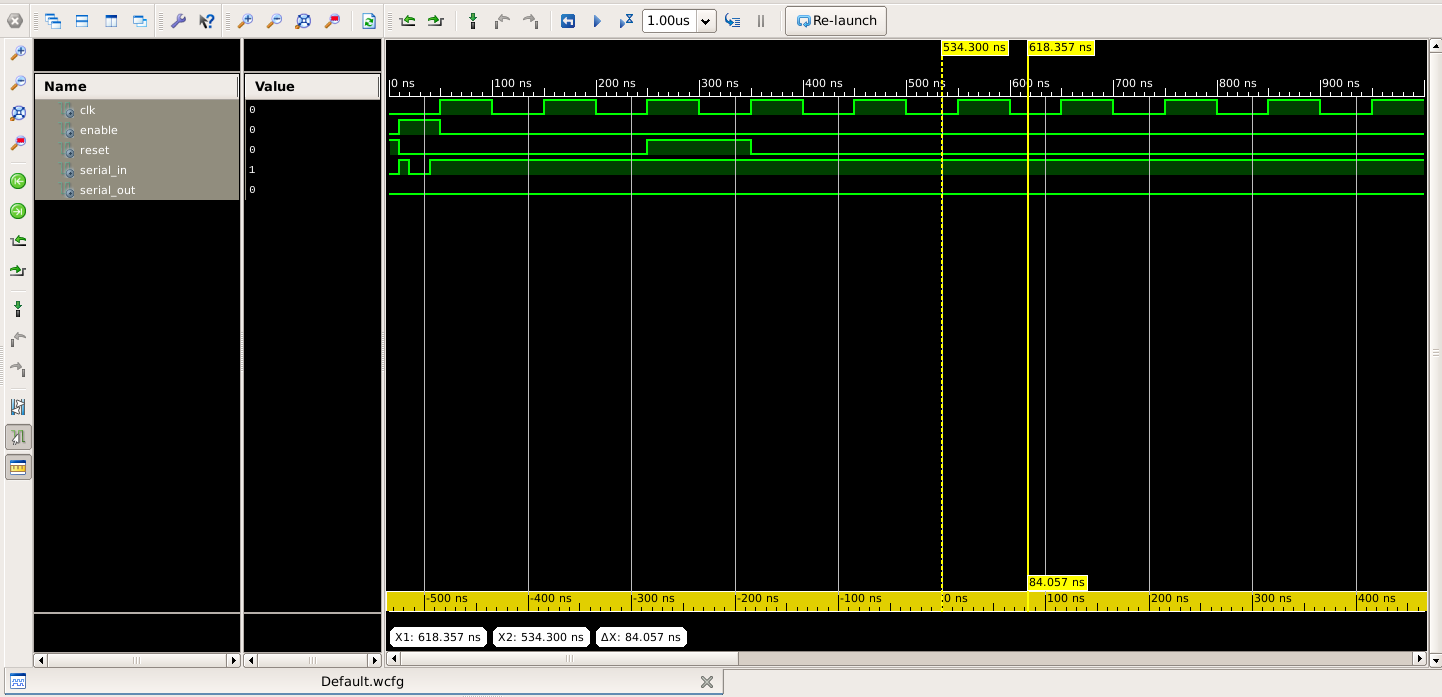
***reset <= '0';***

***wait;***

***end process;***

***END;***

***● Waveform:***

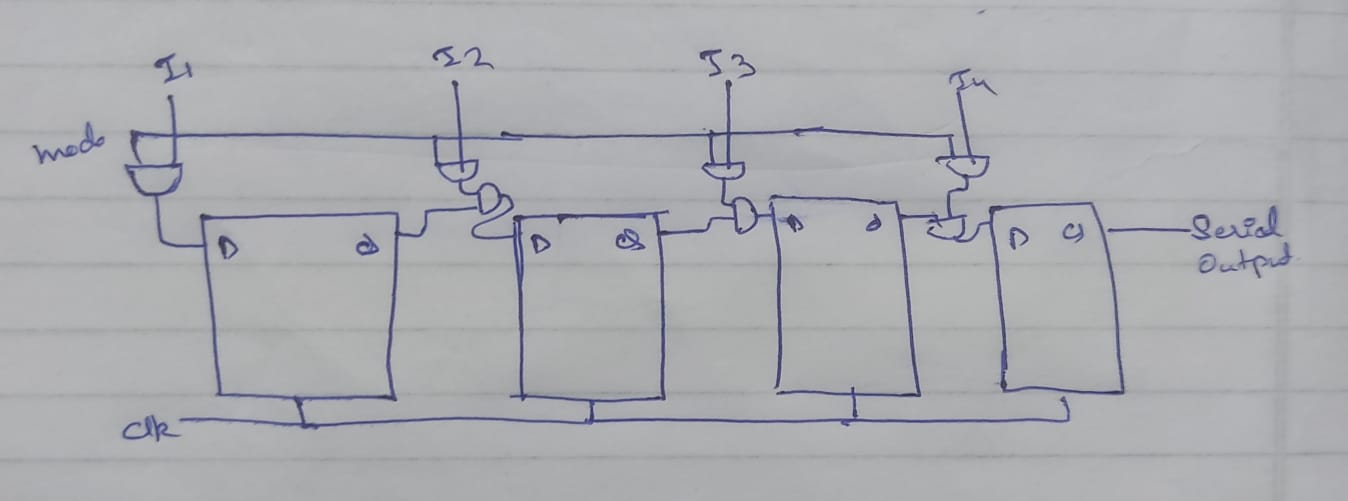


**Question 3**

**Develop a VHDL model to design a four bit Parallel/Serial in Serial out register using D flip flop components. Use an enable and reset input pin to enable and reset the registers. Use a mode input pin to decide whether the input mode is serial or parallel.**

**Aim : To develop a VHDL model for designing a 4-bit Parallel/Serial-In Serial-Out (PISO) shift register using D flip-flop components. The design will include an enable input to control data shifting, a reset input to clear the register contents, and a mode input to select between parallel or serial data input.**

**Block Diagram :**



***● VHDL Code:***

***----------------------------------------------------------------------------------***

***----------------------------------------------------------------------------------***

***-- Company:***

***-- Engineer:***

***--***

***-- Create Date: 18:25:52 10/15/2024***

***-- Design Name:***

***-- Module Name: piso - Behavioral***

***-- Project Name:***

***-- Target Devices:***

***-- Tool versions:***

***-- Description:***

***--***

***-- Dependencies:***

***--***

***-- Revision:***

***-- Revision 0.01 - File Created***

***-- Additional Comments:***

***--***

***----------------------------------------------------------------------------------***

***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;***

***-- Uncomment the following library declaration if instantiating***

***-- any Xilinx primitives in this code.***

***--library UNISIM;***

***--use UNISIM.VComponents.all;***

***entity piso is***

***port(***

***clk:in std\_logic;***

***reset: in std\_logic;***

***enable:in std\_logic;***

***mode : in std\_logic;***

***parallel\_in : in std\_logic\_vector(3 downto 0);***

***serial\_in : in std\_logic;***

***serial\_out: out std\_logic);***

***end piso;***

***architecture Behavioral of piso is***

***component d\_ff is***

***port ( d : in std\_logic;***

***clk : in std\_logic;***

***enable : in std\_logic;***

***reset : in std\_logic;***

***q : out std\_logic);***

***end component;***

***signal d : STD\_LOGIC\_VECTOR(3 downto 0);***

***signal t : STD\_LOGIC\_VECTOR(3 downto 0);***

***begin***

***step1:d\_ff port map(d(0),clk,enable,reset,t(0));***

***step2:d\_ff port map(d(1),clk,enable,reset,t(1));***

***step3:d\_ff port map(d(2),clk,enable,reset,t(2));***

***step4:d\_ff port map(d(3),clk,enable,reset,t(3));***

***process(clk,reset)***

***begin***

***if reset='1' then***

***d<=(others=>'0');***

***elsif rising\_edge(clk) then***

***if enable = '1' then***

***if mode = '0' then***

***d<=parallel\_in;***

***else***

***d(3 downto 1)<=t(2 downto 0);***

***d(0)<=serial\_in;***

***end if;***

***end if;***

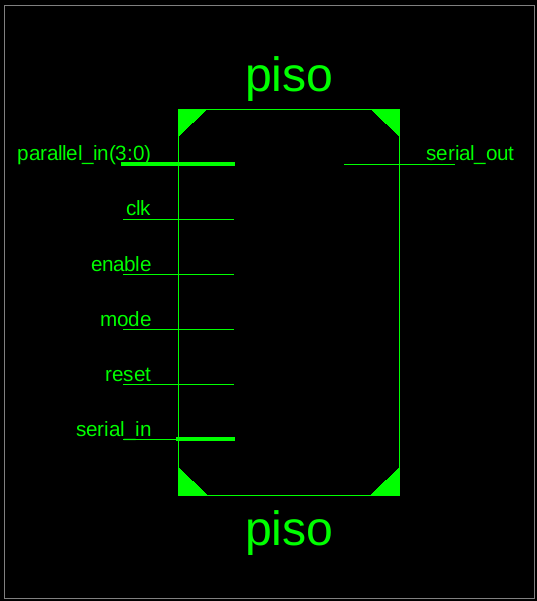
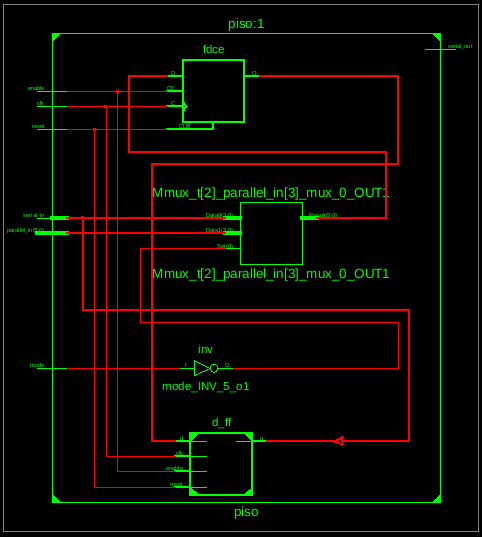
***end if;***

***end process;***

***serial\_out<=t(3);***

***end Behavioral;***

***● RTL Diagram:***

***● Testbench Code:   
--------------------------------------------------------------------------------***

***-- Company:***

***-- Engineer:***

***--***

***-- Create Date: 18:35:51 10/15/2024***

***-- Design Name:***

***-- Module Name: /home/ise/Assignment3/piso\_tb.vhd***

***-- Project Name: Assignment3***

***-- Target Device:***

***-- Tool versions:***

***-- Description:***

***--***

***-- VHDL Test Bench Created by ISE for module: piso***

***--***

***-- 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 piso\_tb IS***

***END piso\_tb;***

***ARCHITECTURE behavior OF piso\_tb IS***

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

***COMPONENT piso***

***PORT(***

***clk : IN std\_logic;***

***reset : IN std\_logic;***

***enable : IN std\_logic;***

***mode : IN std\_logic;***

***parallel\_in : IN std\_logic\_vector(3 downto 0);***

***serial\_in : IN std\_logic;***

***serial\_out : OUT std\_logic***

***);***

***END COMPONENT;***

***--Inputs***

***signal clk : std\_logic := '0';***

***signal reset : std\_logic := '0';***

***signal enable : std\_logic := '0';***

***signal mode : std\_logic := '0';***

***signal parallel\_in : std\_logic\_vector(3 downto 0) := (others => '0');***

***signal serial\_in : std\_logic := '0';***

***--Outputs***

***signal serial\_out : std\_logic;***

***-- Clock period definitions***

***constant clk\_period : time := 100 ns;***

***BEGIN***

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

***uut: piso PORT MAP (***

***clk => clk,***

***reset => reset,***

***enable => enable,***

***mode => mode,***

***parallel\_in => parallel\_in,***

***serial\_in => serial\_in,***

***serial\_out => serial\_out***

***);***

***-- 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***

***reset <= '1';***

***wait for 100 ns;***

***reset <= '0';***

***wait for 100 ns;***

***enable <= '1';***

***mode <= '0';***

***parallel\_in <= "0101";***

***wait for 100 ns;***

***mode <= '1';***

***serial\_in <= '1';***

***wait for 100 ns;***

***wait for 100 ns;***

***serial\_in <= '0';***

***wait for 100 ns;***

***serial\_in <= '1';***

***wait for 100 ns;***

***serial\_in <= '1';***

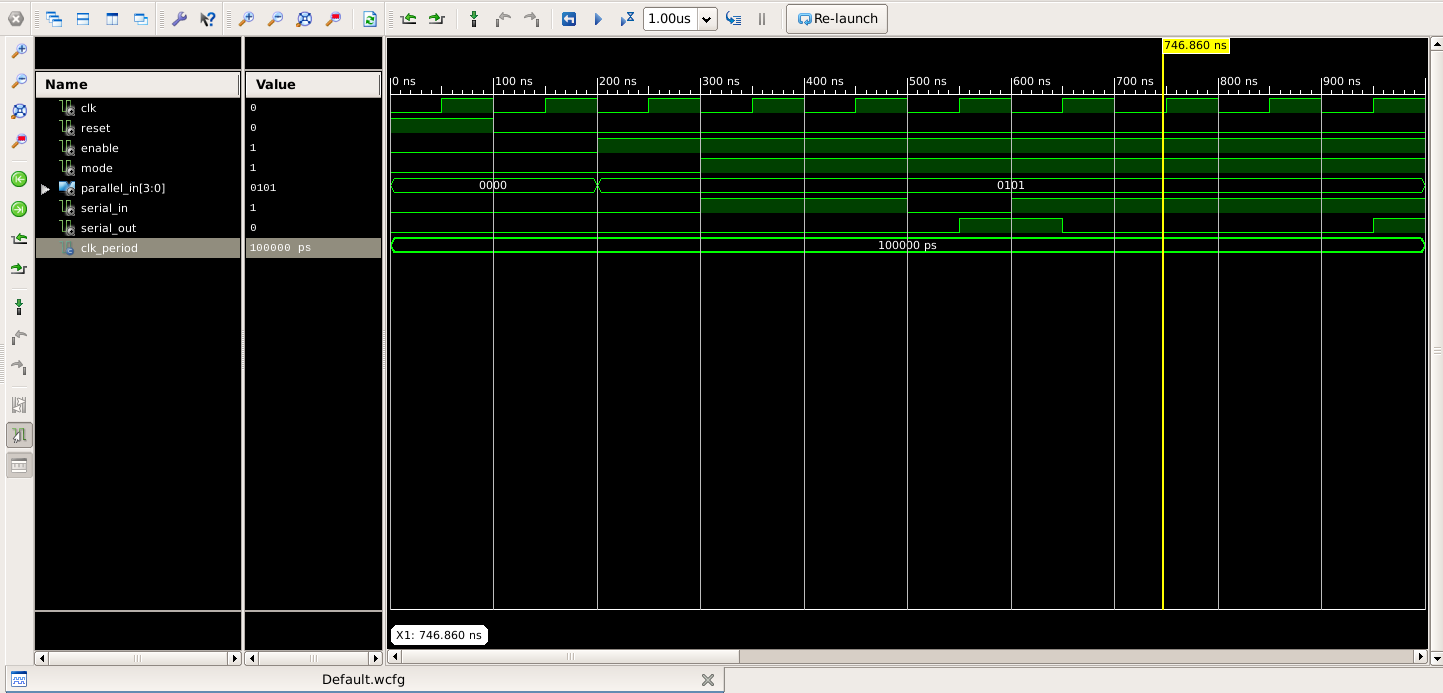
***wait for 100 ns;***

***wait;***

***end process;***

***END;***

***● Waveform:***

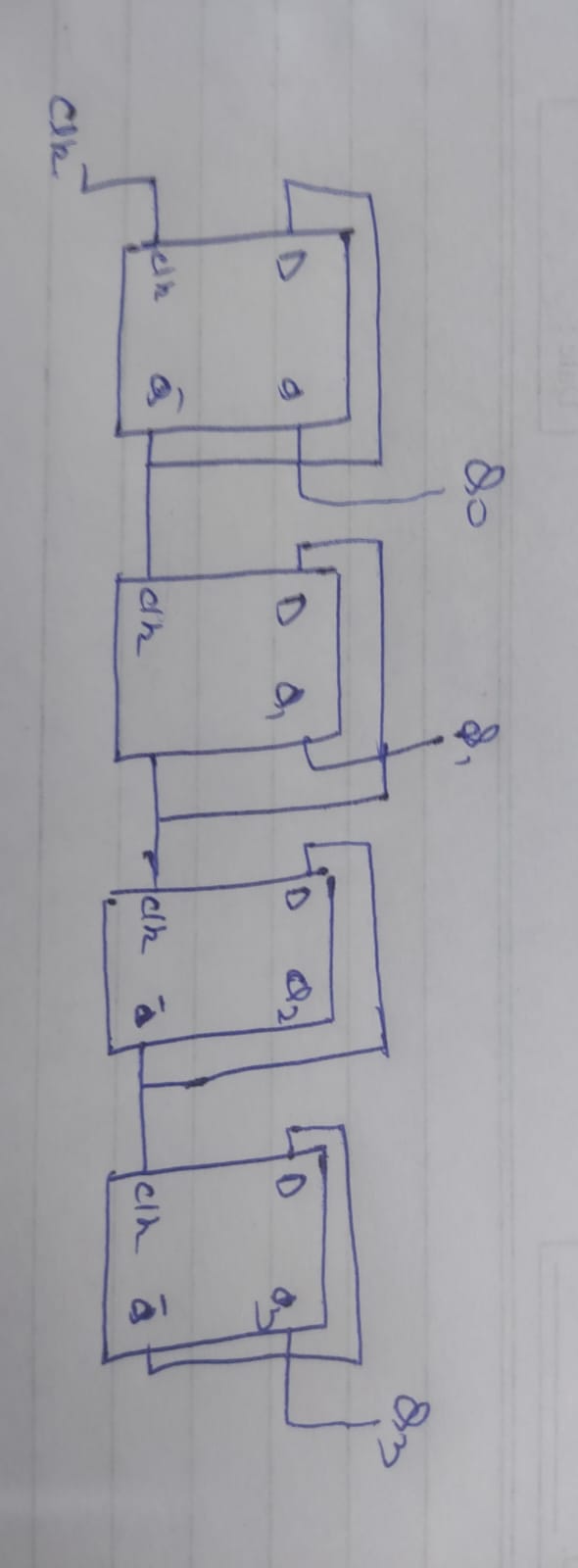


**Question 4**

**Develop a VHDL model to design a four bit ripple counter using D flip flop components. Use an enable and reset input pin to enable and reset the counter.**

**Aim : To develop a VHDL model for designing a 4-bit ripple counter using D flip-flop components. The design will include an enable input to control the counting operation and a reset input to asynchronously reset the counter to zero**

**Block Diagram :**



***● VHDL Code:***

***----------------------------------------------------------------------------------***

***----------------------------------------------------------------------------------***

***-- Company:***

***-- Engineer:***

***--***

***-- Create Date: 18:41:12 10/15/2024***

***-- Design Name:***

***-- Module Name: ripple\_counter - Behavioral***

***-- Project Name:***

***-- Target Devices:***

***-- Tool versions:***

***-- Description:***

***--***

***-- Dependencies:***

***--***

***-- Revision:***

***-- Revision 0.01 - File Created***

***-- Additional Comments:***

***--***

***----------------------------------------------------------------------------------***

***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;***

***-- Uncomment the following library declaration if instantiating***

***-- any Xilinx primitives in this code.***

***--library UNISIM;***

***--use UNISIM.VComponents.all;***

***entity ripple\_counter is***

***port(***

***clk : in STD\_LOGIC;***

***enable : in STD\_LOGIC;***

***reset : in STD\_LOGIC;***

***count : out STD\_LOGIC\_VECTOR(3 downto 0)***

***);***

***end ripple\_counter;***

***architecture Behavioral of ripple\_counter is***

***component d\_ff is***

***port ( d : in std\_logic;***

***clk : in std\_logic;***

***enable : in std\_logic;***

***reset : in std\_logic;***

***q : out std\_logic);***

***end component;***

***signal t:std\_logic\_vector (3 downto 0); signal d\_input : std\_logic\_vector(3 downto 0);***

***begin***

***d\_input(0) <= not t(0);***

***d\_input(1) <= not t(1);***

***d\_input(2) <= not t(2);***

***d\_input(3) <= not t(3);***

***step1:d\_ff port map(d=>d\_input(0),clk=>clk,enable=>enable,reset=>reset,q=>t(0));***

***step2:d\_ff port map(d=>d\_input(1),clk=>t(0),enable=>enable,reset=>reset,q=>t(1));***

***step3:d\_ff port map(d=>d\_input(2),clk=>t(1),enable=>enable,reset=>reset,q=>t(2));***

***step4:d\_ff port map(d=>d\_input(3),clk=>t(2),enable=>enable,reset=>reset,q=>t(3));***

***process(clk, reset)***

***begin***

***if reset = '1' then***

***count <= "0000";***

***elsif rising\_edge(clk) then***

***if enable = '1' then***

***count <= t;***

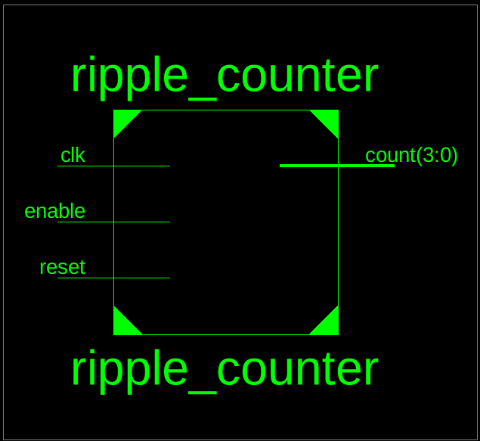
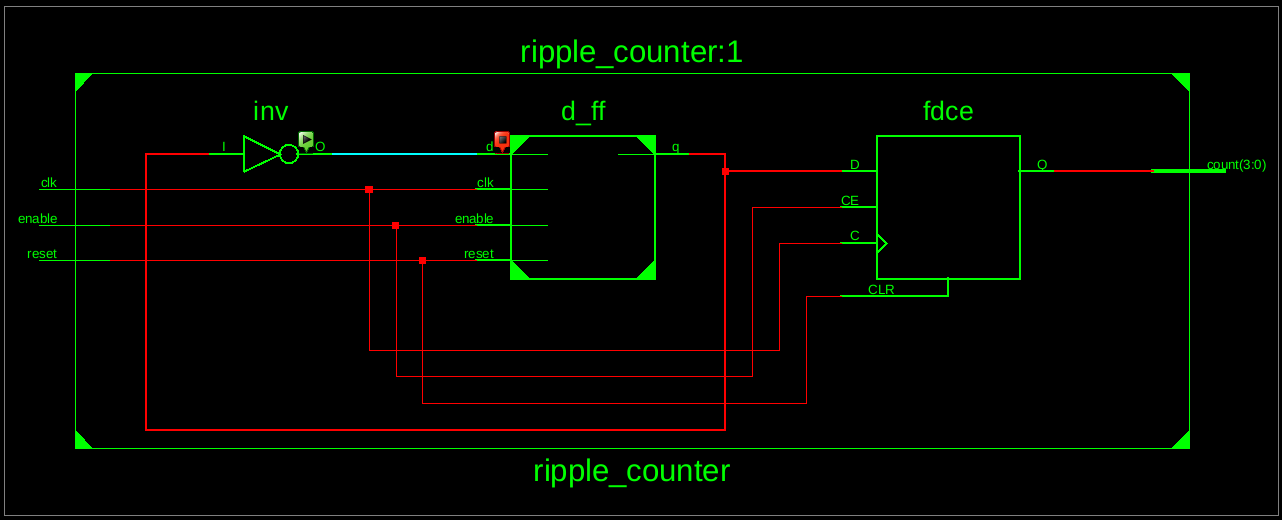
***end if;***

***end if;***

***end process;***

***end Behavioral;***

***● RTL Diagram:***

***● Testbench Code:   
--------------------------------------------------------------------------------***

***-- Company:***

***-- Engineer:***

***--***

***-- Create Date: 18:48:52 10/15/2024***

***-- Design Name:***

***-- Module Name: /home/ise/Assignment3/ripple\_counter\_tb.vhd***

***-- Project Name: Assignment3***

***-- Target Device:***

***-- Tool versions:***

***-- Description:***

***--***

***-- VHDL Test Bench Created by ISE for module: ripple\_counter***

***--***

***-- 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 ripple\_counter\_tb IS***

***END ripple\_counter\_tb;***

***ARCHITECTURE behavior OF ripple\_counter\_tb IS***

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

***COMPONENT ripple\_counter***

***PORT(***

***clk : IN std\_logic;***

***enable : IN std\_logic;***

***reset : IN std\_logic;***

***count : OUT std\_logic\_vector(3 downto 0)***

***);***

***END COMPONENT;***

***--Inputs***

***signal clk : std\_logic := '0';***

***signal enable : std\_logic := '0';***

***signal reset : std\_logic := '0';***

***--Outputs***

***signal count : std\_logic\_vector(3 downto 0);***

***-- Clock period definitions***

***constant clk\_period : time := 10 ns;***

***BEGIN***

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

***uut: ripple\_counter PORT MAP (***

***clk => clk,***

***enable => enable,***

***reset => reset,***

***count => count***

***);***

***-- Clock process definitions***

***clk\_process :process***

***begin***

***while true loop***

***clk <= '0';***

***wait for clk\_period/2;***

***clk <= '1';***

***wait for clk\_period/2;***

***end loop;***

***end process;***

***-- Stimulus process***

***stim\_proc: process***

***begin***

***reset <= '1';***

***enable <= '0';***

***wait for 20 ns;***

***reset <= '0';***

***wait for 20 ns;***

***enable <= '1';***

***wait for 50 ns;***

***enable <= '0';***

***wait for 20 ns;***

***enable <= '1';***

***wait for 50 ns;***

***reset <= '1';***

***wait for 20 ns;***

***reset <= '0';***

***wait for 20 ns;***

***enable <= '1';***

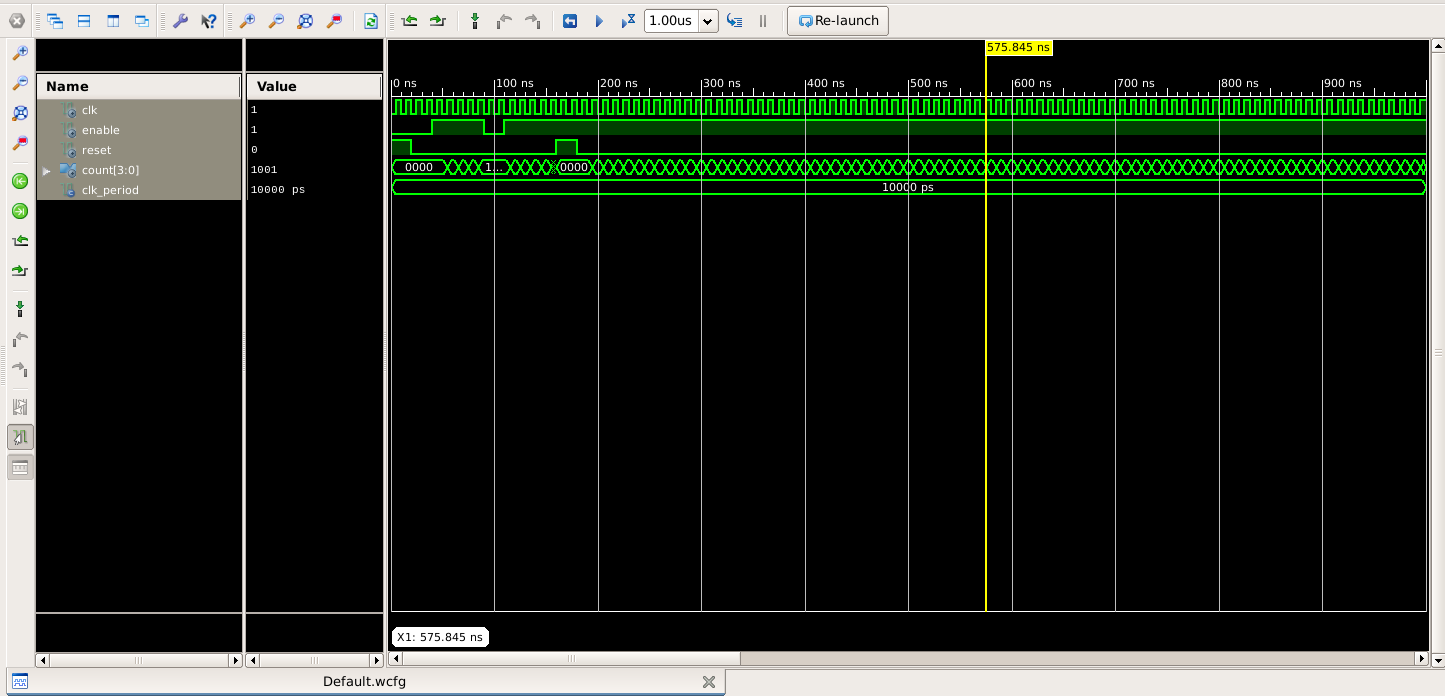
***wait for 50 ns;***

***wait;***

***end process;***

***END;***

***● Waveform:***



Proof That it Was Not Copied

***You can check My Github Account***

https://github.com/ij-roy/Semester-3/tree/main/COA/Assignment%203

***I had uploaded Every picture and code used above and you can also check the Date of Upload***

**Thank You**