Experiment No: 3

**Vhdl code for mod n counter simulator**

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

use IEEE.STD\_LOGIC\_unsigned.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 modncount is

Port ( clk : in STD\_LOGIC;

clr : in STD\_LOGIC;

q : inout STD\_LOGIC\_VECTOR (2 downto 0));

end modncount;

architecture Behavioral of modncount is

signal count: std\_logic\_vector(2 downto 0);

begin

process(clk)

begin

if (clr='1') then count <= "000";

elsif (rising\_edge (clk)) then

if (count="100")

then count <= "000";

else

count<=count+ 1;

end if;

end if;

end process;

q<=count;

end Behavioral;

**Testbench:**

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 hhh IS

END hhh;

ARCHITECTURE behavior OF hhh IS

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

COMPONENT modncount

PORT(

clk : IN std\_logic;

clr : IN std\_logic;

q : INOUT std\_logic\_vector(2 downto 0)

);

END COMPONENT;

--Inputs

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

signal clr : std\_logic := '0';

--BiDirs

signal q : std\_logic\_vector(2 downto 0);

-- Clock period definitions

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

BEGIN

-- Instantiate the Unit Under Test (UUT)

uut: modncount PORT MAP (

clk => clk,

clr => clr,

q => q

);

-- Clock process definitions

clk\_process :process

begin

clk <= '0';

wait for 10 ns;

clk <= '1';

wait for 10 ns;

end process;

-- Stimulus process

stim\_proc: process

begin

clr<='1';

-- hold reset state for 100 ns.

wait for 20 ns;

clr<='0';

-- hold reset state for 100 ns.

wait for 20 ns;

-- hold reset state for 100 ns.

-- wait for 100 ns;

--

-- wait for clk\_period\*10;

-- insert stimulus here

wait;

end process;

END;

**UCF**

Net clk loc=”p182”;

Net clr loc=”p102”;

Net q(2) loc=”p165”;

Net q(2) loc=”p166”;

Net q(2) loc=”p167”;







