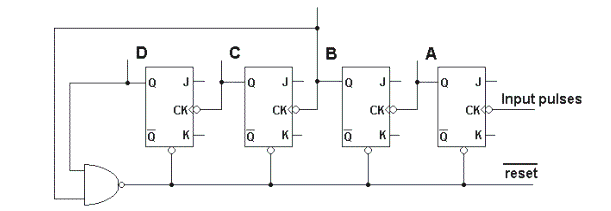
LAB -3

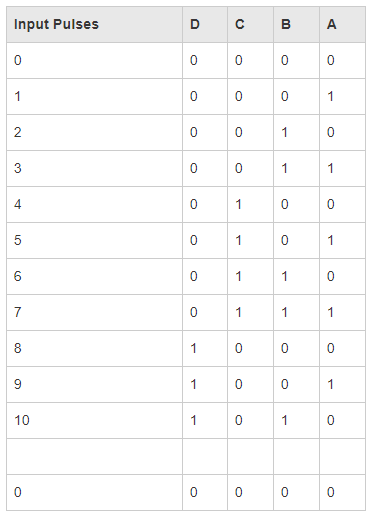
IMPLEMENTATION OF D FLIP FLOP AND BCD COUNTER USING VHDL

Theory

BCD Counter

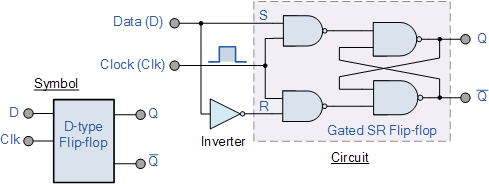
A binary coded decimal (BCD) is a serial digital counter that counts ten digits. And it resets for every new clock input. As it can go through 10 unique combinations of output, it is also called as “Decade counter”.

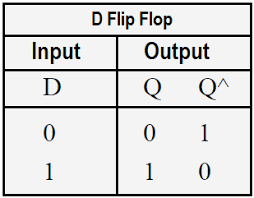




D flip flop

The **D Flip Flop** is by far the most important of the clocked flip-flops as it ensures that ensures that inputs S and R are never equal to one at the same time. The D-type flip flop are constructed from a gated SR flip-flop with an inverter added between the S and the R inputs to allow for a single D (Data) input.





VHDL code for D flip flop

library ieee;

use ieee. std\_logic\_1164.all;

use ieee. std\_logic\_arith.all;

use ieee. std\_logic\_unsigned.all;

entity D\_FF is

PORT( D,CLOCK: in std\_logic;

Q: out std\_logic);

end D\_FF;

architecture behavioral of D\_FF is

begin

process(CLOCK)

begin

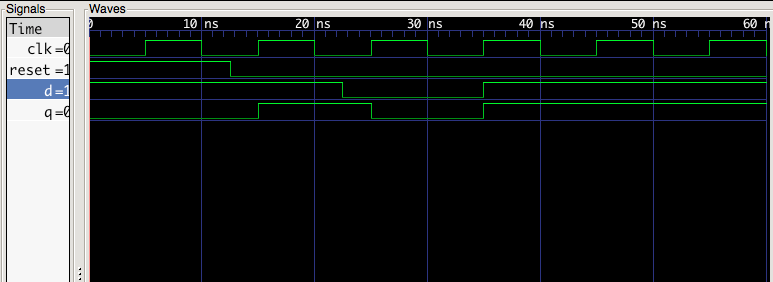
if(CLOCK='1' and CLOCK'EVENT) then

Q <= D;

end if;

end process;

end behavioral;



> VHDL code example for BCD counter:

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

use IEEE.STD\_LOGIC\_ARITH.ALL;

use IEEE.STD\_LOGIC\_UNSIGNED.ALL;

entity bcdcounter is

Port( rst : in STD\_LOGIC;

clk : in STD\_LOGIC;

count: out STD\_LOGIC\_VECTOR (3 DOWNTO 0)

);

end bcdcounter;

architecture Behavioral of bcdcounter is

TYPE state is (zero, one, two, three, four, five, six, seven, eight, nine);

SIGNAL pr\_state, nx\_state: state;

SIGNAL clk1: STD LOGIC:='0';

SIGNAL N: INTEGER RANGE O to 49999999:=0;

Begin

--clock design

process(clk)

begin

if(rising\_edge(clk)) then n<=n+1;

if(n=49999999) then clk1<=not clk1; n<=0;

end if;

end if;

end process;

--sequential part

process(rst,clk1)

begin

if(rst='1') then pr\_state<=zero;

elsif(rising\_edge(clk1)) then pr\_state<=nx\_state;

end if;

end process;

--combinational part

process(pr\_state)

begin

case pr\_state is

when zero=> count<="0000";

nx\_state<=one;

when one=> count<="0001";

nx\_state<=two;

when two=> count<="0010";

nx\_state<=three;

when three=> count<="0011";

nx\_state<=four;

when four=> count<="0100";

nx\_state<=five;

when five=> count<="0101";

nx\_state<=six;

when six=> count<="0110";

nx\_state<=seven;

when seven=> count<="0111";

nx\_state<=eight;

when eight=> count<="1000";.

nx\_state<=nine;

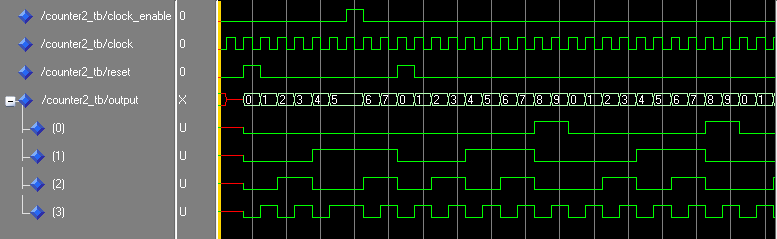
when nine=> count<="1001";

nx\_state<=zero;

end case;

end process;

end Behavioral;



Discussion And Conclusion:

In this lab we visualized bcd counter and D flip flop using VHDL and viewed their waveform.