**4 BIT BARREL SHIFTER USING STRUCTRAL**
library IEEE;
use IEEE.STD\_LOGIC\_1164.ALL;
use IEEE.STD\_LOGIC\_ARITH.ALL;
use IEEE.STD\_LOGIC\_UNSIGNED.ALL;

---- Uncomment the following library declaration if instantiating
---- any Xilinx primitives in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity MUX41CASE is
    Port ( SEL : in  STD\_LOGIC\_VECTOR (01 downto 0);
           D : in  STD\_LOGIC\_VECTOR (03 downto 0);
           Y : out  STD\_LOGIC);
end MUX41CASE;

architecture Behavioral of MUX41CASE is
begin
PROCESS (D,SEL)
BEGIN
    CASE SEL IS
     WHEN "00" => Y <= D(0);
     WHEN "01" => Y <= D(1);
     WHEN "10" => Y <= D(2);
     WHEN "11" => Y <= D(3);
     WHEN OTHERS => Y <= 'Z';
     END CASE;
END PROCESS;
end Behavioral;
----------------------------------------------------------------------------------
library IEEE;
use IEEE.STD\_LOGIC\_1164.ALL;
use IEEE.STD\_LOGIC\_ARITH.ALL;
use IEEE.STD\_LOGIC\_UNSIGNED.ALL;

---- Uncomment the following library declaration if instantiating
---- any Xilinx primitives in this code.
--library UNISIM;
--use UNISIM.VComponents.all;

entity barrel4structri8 is
    Port ( D : in  STD\_LOGIC\_VECTOR (03 downto 0);
           SEL : in  STD\_LOGIC\_VECTOR (01 downto 0);
           Y : out  STD\_LOGIC\_VECTOR (03 downto 0));
end barrel4structri8;

architecture STRUCTURAL of barrel4structri8 is
COMPONENT MUX41CASE is
  Port ( SEL : in  STD\_LOGIC\_VECTOR (01 downto 0);
           D : in  STD\_LOGIC\_VECTOR (03 downto 0);
           Y : out  STD\_LOGIC);
end COMPONENT;
signal t1,t2,t3: std\_logic\_vector(3 downto 0);
begin
t1 <= d(0) & d(3 downto 1);
t2 <= d(1 downto 0) & d(3 downto 2);
t3 <= d(2 downto 1) & d(0) & d(3);
U1 : MUX41CASE port map (SEL (1 downto 0), d(3 downto 0), y(0));
U2 : MUX41CASE port map (SEL (1 downto 0), t1 , y(1));
U3 : MUX41CASE port map (SEL (1 downto 0), t2 , y(2));
U4 : MUX41CASE port map (SEL (1 downto 0), t3 , y(3));
end STRUCTURAL;



FA USING HA AS A COMPONENT

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

use IEEE.STD\_LOGIC\_ARITH.ALL;

use IEEE.STD\_LOGIC\_UNSIGNED.ALL;

entity ha is

 port ( aa,ba :in std\_logic;

 ya,co : out std\_logic);

end ha;

architecture dataflow of ha is

begin

ya <= aa xor ba;

co <= aa and ba;

end dataflow;

---------------------------------------

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

use IEEE.STD\_LOGIC\_ARITH.ALL;

use IEEE.STD\_LOGIC\_UNSIGNED.ALL;

entity or1 is

 port ( ab,bb :in std\_logic;

 yb : out std\_logic);

end or1;

architecture dataflow of or1 is

begin

yb <= ab or bb;

end dataflow;

---------------------------------------

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

use IEEE.STD\_LOGIC\_ARITH.ALL;

use IEEE.STD\_LOGIC\_UNSIGNED.ALL;

entity fausingha is

 Port ( a,b,cin : in STD\_LOGIC;

 y,co : out STD\_LOGIC);

end fausingha;

architecture structural of fausingha is

component ha is

 port ( aa,ba :in std\_logic;

 ya,co : out std\_logic);

end component;

component or1 is

 port ( ab,bb :in std\_logic;

 yb : out std\_logic);

end component;

signal op1,co1,co2 : std\_logic;

begin

u1: ha port map (a,b,op1,co1);

u2: ha port map (op1,cin,y,co2);

u3: or1 port map (co2,co1,co);

end structural;



**BCD TO EXCESS3:**

library IEEE;

use IEEE.STD\_LOGIC\_1164.ALL;

use IEEE.STD\_LOGIC\_ARITH.ALL;

use IEEE.STD\_LOGIC\_UNSIGNED.ALL;

entity BCDTOEX3 is

 Port ( D : in STD\_LOGIC\_VECTOR (03 downto 0);

 Y : out STD\_LOGIC\_VECTOR (03 downto 0));

end BCDTOEX3;

architecture Behavioral of BCDTOEX3 is

begin

Y <= D + "0011";

end Behavioral;