Example 41: Divide-by-2 Counter

Example 44: N-Bit Register (Generic)

Exercise 1: Using the Generic n-Bit Register component implement an 8-Bit Register (see Listing 7.8 on page 178) on the BASYS-2 Board. Use x7seg.vhd for display.
Clock Divide Component

use IEEE.STD_LOGIC_1164.all;
use IEEE.STD_LOGIC_unsigned.all;

entity clkdiv is
port (mclk : in STD_LOGIC;
clr : in STD_LOGIC;
clk25 : out STD_LOGIC;
clk190 : out STD_LOGIC;
clk3 : out STD_LOGIC);
end clkdiv;

architecture clkdiv of clkdiv is
signal q: STD_LOGIC_VECTOR(23 downto 0);
begin
process(mclk, clr)
begin
if clr = '1' then
q <= X"000000";
elsif mclk’event and mclk = '1' then
q <= q + 1;
end if;
end process;
clk25 <= q(0);    -- 25 MHz
clk190 <= q(17);  -- 190 Hz
clk3 <= q(23);    -- 3 Hz
end clkdiv;

Ring Counters and Shift Registers

architecture ring4 of ring4 is
signal qs: STD_LOGIC_VECTOR(3 downto 0);
begin
process(clk, clr)
begin
if clr = '1' then
qs <= "0001";
elsif clk’event and clk = '1' then
qs(3) <= qs(0);
qs(2 downto 0) <= qs(3 downto 1);
end if;
end process;
q <= qs;
end ring4;

Example 45: Shift Registers
Example 46: Ring Counter

entity ring4 is
port( clk : in STD_LOGIC;
clr : in STD_LOGIC;
q : out STD_LOGIC_VECTOR(3 downto 0);
end ring4;

architecture ring4 of ring4 is
begin
process(clk, clr)
begin
if clr = '1' then
qs <= "0001";
elsif clk’event and clk = '1' then
qs(3) <= qs(0);
qs(2 downto 0) <= qs(3 downto 1);
end if;
end process;
q <= qs;
end ring4;

Exercise 2 : Design and Implement on a BASYS-2 board an 8-Bit Ring Counter (based on Listing 7.11 ring4.vhd)

entity Ring8_top is
port(mclk: in STD_LOGIC;
btn : in STD_LOGIC_VECTOR(3 downto 0);
ld : out STD_LOGIC_VECTOR(7 downto 0) );
end Ring8_top;

architecture Ring8_top is
begin
Ring8_top will need 3 internal signals: clk25a, clk190a, and clk3a

architecture ring4 of ring4 is
signal qs: STD_LOGIC_VECTOR(3 downto 0);
begin
process(clk, clr)
begin
if clr = '1' then
qs <= "0001";
elsif clk’event and clk = '1' then
qs(3) <= qs(0);
qs(2 downto 0) <= qs(3 downto 1);
end if;
end process;
q <= qs;
end ring4;