Dongyun Lee

ID: 112794190

PreLab5: Behavioral Style Combinational Design Using Case, If, and Loop Statements – XS3 to BCD Converter

ESE382-L01

Bench #4

File: X:/ESE382-Lab/Lab5/converter_xs3_to_BCD/xs3_to_BCD_case/src/xs3_bcd_case.vhd

```vhdl
1   ----------------------------------------------------------------------------
2   --
3   -- Title       : xs3_to_BCD_case
4   -- Design      : xe3_to_BCD_case
5   -- Author      : Dongyun Lee
6   -- Company     : Stony Brook University
7   --
8   ----------------------------------------------------------------------------
9   --
10  -- File        : \\Mac\Home\Documents\Aldec_Codes\test\test\src\test.vhd
11  -- Generated   : Sun Feb 25 18:09:46 2024
12  -- From        : interface description file
13  -- By          : Itf2Vhdl ver. 1.22
14  --
15  ----------------------------------------------------------------------------
16  --
17  -- Description : converts a 4-bit Excess-3 (XS-3) input (p, q, r, s) to its corresponding Binary-Coded Decimal (BCD) output (d, c, b, a) using a case statement for the mapping.
18  --
19  ----------------------------------------------------------------------------
20  library IEEE;
21  use IEEE.STD_LOGIC_1164.ALL;
22
23  entity converter_xs3_bcd is
24      port ( p,q,r,s : in  std_logic;
25             d,c,b,a : out  std_logic
26      );
27  end entity converter_xs3_bcd;
28
29  architecture xs3_bcd_case of converter_xs3_bcd is
30      signal temp : std_logic_vector(3 downto 0);
31
32  begin
33      (d,c,b,a) <= temp;
34      casey : process (p,q,r,s)
35      begin
36          case std_logic_vector'(p,q,r,s) is
37              when "0011" => temp <= "0000";
38              when "0100" => temp <= "0001";
39              when "0101" => temp <= "0010";
40              when "0110" => temp <= "0011";
41              when "0111" => temp <= "0100";
42              when "1000" => temp <= "0101";
43              when "1001" => temp <= "0110";
44              when "1010" => temp <= "0111";
45              when "1011" => temp <= "1000";
46              when "1100" => temp <= "1001";
47              when others => temp <= "----";
48
49          end case;
50      end process;
51
```

```
52  end architecture xs3_bcd_case;
```

File: X:/ESE382-Lab/Lab5/converter_xs3_to_BCD/xs3_to_BCD_case/src/xs3_bcd_case_vect.vhd

```
1   ------------------------------------------------------------------------
    ----
2   --
3   -- Title       : xs3_to_BCD_case_vect
4   -- Design      : xe3_to_BCD_case_vect
5   -- Author      : Dongyun Lee
6   -- Company     : Stony Brook University
7   --
8   ------------------------------------------------------------------------
    ----
9   --
10  -- File        : \\Mac\Home\Documents\Aldec_Codes\test\test\src\test.vhd
11  -- Generated   : Sun Feb 25 18:09:46 2024
12  -- From        : interface description file
13  -- By          : Itf2Vhdl ver. 1.22
14  --
15  ------------------------------------------------------------------------
    ----
16  --
17  -- Description : converts a 4-bit Excess-3 (XS-3) input (p, q, r, s) to its
    corresponding Binary-Coded Decimal (BCD) output (d, c, b, a) using a case
    statement for the mapping.
18  --               using vectors instead of scalars.
19  ------------------------------------------------------------------------
    ----
20  library IEEE;
21  use IEEE.STD_LOGIC_1164.ALL;
22  use ieee.numeric_std.all;
23
24  entity converter_xs3_bcd is
25      port ( pqrs : in  std_logic_vector(3 downto 0);
26             dcba : out  std_logic_vector(3 downto 0)
27      );
28  end entity converter_xs3_bcd;
29
30  architecture xs3_bcd_case_vect of converter_xs3_bcd is
31  begin
32      casey : process (pqrs)
33      begin
34          case pqrs is
35              when "0011" => dcba <= "0000";
36              when "0100" => dcba <= "0001";
37              when "0101" => dcba <= "0010";
38              when "0110" => dcba <= "0011";
39              when "0111" => dcba <= "0100";
40              when "1000" => dcba <= "0101";
41              when "1001" => dcba <= "0110";
42              when "1010" => dcba <= "0111";
43              when "1011" => dcba <= "1000";
44              when "1100" => dcba <= "1001";
45              when others => dcba <= "----";
46
47          end case;
48      end process;
49
50  end architecture xs3_bcd_case_vect;
```

```
1   ----------------------------------------------------------------------------
    ----
2   --
3   -- Title       : xs3_to_BCD_if_vect
4   -- Design      : xe3_to_BCD_if_vect
5   -- Author      : Dongyun Lee
6   -- Company     : Stony Brook University
7   --
8   ----------------------------------------------------------------------------
    ----
9   --
10  -- File        : \\Mac\Home\Documents\Aldec_Codes\test\test\src\test.vhd
11  -- Generated   : Sun Feb 25 18:09:46 2024
12  -- From        : interface description file
13  -- By          : Itf2Vhdl ver. 1.22
14  --
15  ----------------------------------------------------------------------------
    ----
16  --
17  -- Description : converts a 4-bit Excess-3 (XS-3) input (p, q, r, s) to its
    corresponding Binary-Coded Decimal (BCD) output (d, c, b, a) using if
    statement.
18  --               using vectors instead of scalars.
19  ----------------------------------------------------------------------------
    ----
20  library IEEE;
21  use IEEE.STD_LOGIC_1164.ALL;
22  use ieee.numeric_std.all;
23
24  entity converter_xs3_bcd is
25      port ( pqrs : in  std_logic_vector(3 downto 0);
26             dcba : out  std_logic_vector(3 downto 0)
27      );
28  end entity converter_xs3_bcd;
29
30  architecture xs3_bcd_case_if_vect of converter_xs3_bcd is
31  begin
32      casey : process (pqrs)
33      begin
34          dcba <= "----";
35
36          if pqrs = "0011" then
37              dcba <= "0000";
38          elsif pqrs = "0100" then
39              dcba <= "0001";
40          elsif pqrs = "0101" then
41              dcba <= "0010";
42          elsif pqrs = "0110" then
43              dcba <= "0011";
44          elsif pqrs = "0111" then
45              dcba <= "0100";
46          elsif pqrs = "1000" then
47              dcba <= "0101";
48          elsif pqrs = "1001" then
49              dcba <= "0110";
50          elsif pqrs = "1010" then
51              dcba <= "0111";
```

```
52          elsif pqrs = "1011" then
53              dcba <= "1000";
54          elsif pqrs = "1100" then
55              dcba <= "1001";
56          end if;
57      end process;
58
59  end architecture xs3_bcd_case_if_vect;
```

File: X:\ESE382-Lab\Lab5\converter_xs3_to_BCD\xs3_to_BCD_case\src\xs3_bcd_loop.vhd (/conv

```
1   ----------------------------------------------------------------------------
    -------
2   --
3   -- Title       : xs3_to_BCD_loop
4   -- Design      : xe3_to_BCD_loop
5   -- Author      : Dongyun Lee
6   -- Company     : Stony Brook University
7   --
8   ----------------------------------------------------------------------------
    -------
9   --
10  -- File        : \\Mac\Home\Documents\Aldec_Codes\test\test\src\test.vhd
11  -- Generated   : Sun Feb 25 18:09:46 2024
12  -- From        : interface description file
13  -- By          : Itf2Vhdl ver. 1.22
14  --
15  ----------------------------------------------------------------------------
    -------
16  --
17  -- Description : converts a 4-bit Excess-3 (XS-3) input (p, q, r, s) to
    its corresponding Binary-Coded Decimal (BCD) output (d, c, b, a) using
    looping.
18  --               using vectors instead of scalars.
19  --                  no don't cares
20  ----------------------------------------------------------------------------
    -------
21  library IEEE;
22  use IEEE.STD_LOGIC_1164.ALL;
23  use ieee.numeric_std.all;
24
25  entity converter_xs3_bcd is
26      port ( pqrs : in  std_logic_vector(3 downto 0);
27             dcba : out  std_logic_vector(3 downto 0)
28      );
29  end entity converter_xs3_bcd;
30
31  architecture xs3_bcd_loop of converter_xs3_bcd is
32  begin
33      comp: process (pqrs)
34
35      variable bcd_var : std_logic_vector(3 downto 0);
36      begin
37          bcd_var := "0000";  -- default value
38
39          for i in 0 to 9 loop
40              if unsigned(pqrs) = to_unsigned(i + 3, 4) then -- if i + 3 =
    pqrs then assgin to bcd_var
41                  bcd_var := std_logic_vector(to_unsigned(i, 4));
42                  exit;
43              end if;
44          end loop;
45
46          dcba <= bcd_var;
47      end process;
48
49  end architecture xs3_bcd_loop;
```