CASE Statement

General syntax:

CASE expression IS
    WHEN constant-value =>
        statement;
        {statement;}
    WHEN constant-value =>
        statement;
        {statement;}
    . . .
    WHEN OTHERS =>
        statement;
        {statement;}

• Constant-value can be:
  • Single value (e.g., “0000”)
  • List of values separated by | operator (e.g., “001” | “101”)
  • A range (e.g., “000” TO “100”)
• Similar to the SELECT assignment statement, all possible values of the expression used for the WHEN clause must be listed (use WHEN OTHERS). Otherwise, a syntax error will occur.
CASE statement for D FF with MUX:

LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY CaseMuxFF IS
PORT ( D0,D1,sel,clock : IN STD_LOGIC;
Q : OUT STD_LOGIC);
END CaseMuxFF;

ARCHITECTURE Behavior OF CaseMuxFF IS
BEGIN
PROCESS
BEGIN
WAIT UNTIL clock'Event AND clock = '1';
CASE sel IS
WHEN '0' =>
Q <= D0;
WHEN OTHERS =>
Q <= D1;
END CASE;
END PROCESS;
END Behavior;
CASE Statement for Combinatorial Circuit:

LIBRARY ieee ;
USE ieee.std_logic_1164.all ;

ENTITY mux2to1 IS
    PORT ( w0, w1, s : IN
           STD_LOGIC ;
           f : OUT   STD_LOGIC ) ;
END mux2to1 ;

ARCHITECTURE Behavior OF mux2to1 IS

BEGIN
    PROCESS ( w0, w1, s )
    BEGIN
        CASE s IS
            WHEN '0' =>
                f <= w0 ;
            WHEN OTHERS =>
                f <= w1 ;
        END CASE ;
    END PROCESS ;
END Behavior ;
LIBRARY ieee;
USE ieee.std_logic_1164.all;
ENTITY priority IS
  PORT (w : IN STD_LOGIC_VECTOR(3 DOWNTO 0);
        y : OUT STD_LOGIC_VECTOR(1 DOWNTO 0);
        z : OUT STD_LOGIC);
END priority;

ARCHITECTURE Behavior OF priority IS
BEGIN
  PROCESS (w)
  BEGIN
    IF w(3) = '1' THEN
      y <= "11";
    ELSIF w(2) = '1' THEN
      y <= "10";
    ELSIF w(1) = '1' THEN
      y <= "01";
    ELSE
      y <= "00";
    END IF;
  END PROCESS;
z <= '0' WHEN w = "0000" ELSE '1';
END Behavior;
Alternative code for the priority encoder

LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY priority IS
  PORT (w : IN STD_LOGIC_VECTOR(3 DOWNTO 0);
       y : OUT STD_LOGIC_VECTOR(1 DOWNTO 0);
       z : OUT STD_LOGIC);
END priority;

ARCHITECTURE Behavior OF priority IS
BEGIN
  PROCESS (w)
  BEGIN
    y <= "00";
    IF w(1) = '1' THEN y <= "01"; END IF;
    IF w(2) = '1' THEN y <= "10"; END IF;
    IF w(3) = '1' THEN y <= "11"; END IF;
    z <= '1';
    IF w = "0000" THEN z <= '0'; END IF;
  END PROCESS;
END Behavior;
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY ImpliedMemoryIF IS
    PORT ( sel, x2 : IN STD_LOGIC ;
           f     : OUT STD_LOGIC ) ;
END ImpliedMemoryIF ;

ARCHITECTURE Behavior OF ImpliedMemoryIF IS BEGIN
    PROCESS (sel, x2)
    BEGIN
        IF sel = '1' THEN
            f <= x2;
        END IF;
    END PROCESS ;
END Behavior ;
LIBRARY ieee;
USE ieee.std_logic_1164.all;
ENTITY ImpliedMemoryCASE IS
    PORT ( w0, w1, s : IN STD_LOGIC;
           f,g : OUT STD_LOGIC );
END ImpliedMemoryCASE;

ARCHITECTURE Behavior OF ImpliedMemoryCASE IS
BEGIN
    PROCESS ( w0, w1, s )
    BEGIN
        CASE s IS
            WHEN '0' =>
                f <= w0 ;
            WHEN OTHERS =>
                g <= w1 ;
        END CASE;
    END PROCESS;
END Behavior;

➢ The value of f is not specified when s not equal to ‘0’
➢ The value of g is not specified when s = ‘0’.
➢ The default values for those cases is “HOLD”.
    Thus, there is an implied memory (See .vwf file).
➢ f and g will be implemented as latches.
ImpliedMemoryCASE warning messages

From ImpliedMemoryCASE.vwf file
If use both (Resetn, Clock) in sensitivity list and don’t use Clock’EVENT:

LIBRARY ieee ;
USE ieee.std_logic_1164.all ;

ENTITY DFFAsynchRNoEvent IS
    PORT ( D, Resetn, Clock : IN   STD_LOGIC ;
           Q   : OUT   STD_LOGIC) ;
END DFFAsynchRNoEvent ;

ARCHITECTURE Behavior OF DFFAsynchRNoEvent IS
BEGIN
    PROCESS ( Resetn, Clock )
    BEGIN
        IF Resetn = '0' THEN
            Q <= '0' ;
        ELSIF Clock = '1' THEN -- did not use Clock’EVENT
            Q <= D ;
        END IF ;
    END PROCESS ;
END Behavior ;
If use both (Resetn, Clock) in sensitivity list and don’t use Clock’EVENT: