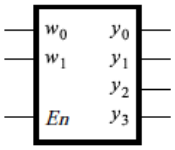
1.

a) The graphical symbol and truth table of a 2-to-4 decoder is shown below.

En	w_1	w_0	y_0	y_1	y_2	y_3
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1
0	x	x	0	0	0	0

(a) Truth table



(b) Graphical symbol

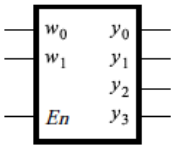
Complete the behavioral VHDL of 2-to-4 decoder (entity named **dec2to4**) using **IF-THEN-ELSE only.**

```
ENTITY dec2to4 IS
    PORT ( w : IN STD LOGIC VECTOR(1 DOWNTO 0) ;
           En : IN STD LOGIC ;
           y : OUT STD LOGIC VECTOR(0 TO 3) ) ;
END dec2to4 ;

ARCHITECTURE Behavior OF dec2to4 IS

BEGIN

    --- use IF-THEN-ELSE only ----

    if En = '1' then
           if w = "00" then y <= "1000"
           elsif w = ''01'' then y <= "0100";
           elsif  w = ''10'' then y <= "0010";
           elsif  w = ''11'' => y <= "0001";
    else
           y <= "0000";
    end if;

END Behavior ;
```

b) The graphical symbol and truth table of a 2-to-4 decoder is shown below.

En	w_1	w_0	y_0	y_1	y_2	y_3
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1
0	x	x	0	0	0	0

(a) Truth table



(b) Graphical symbol

Complete the behavioral VHDL of 2-to-4 decoder (entity named **dec2to4**) using IF and then **CASE.**

```
ENTITY dec2to4 IS
    PORT ( w : IN STD LOGIC VECTOR(1 DOWNTO 0) ;
           En : IN STD LOGIC ;
           y : OUT STD LOGIC VECTOR(0 TO 3) ) ;
END dec2to4 ;
```

ARCHITECTURE Behavior OF **dec2to4** IS

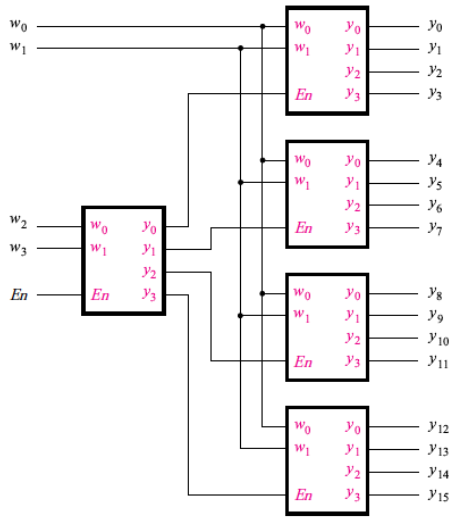
BEGIN

--- use IF and then **CASE** ----

```
if En = '1' then
        case (w) is
                WHEN ''00'' => y <= "1000";
                WHEN ''01'' => y <= "0100";
                WHEN ''10'' => y <= "0010";
                WHEN ''11'' => y <= "0001";
        end case;
else
        y <= "0000";
end if;
```

END Behavior ;

c) Complete the behavioral VHDL of 4-to-16 decoder (entity named **dec4to16**).




```
ENTITY dec4to16 IS
    PORT ( w : IN STD LOGIC VECTOR(3 DOWNTO 0) ;
           En : IN STD LOGIC ;
           y : OUT STD LOGIC VECTOR(0 TO 15) ) ;
END dec4to16 ;

ARCHITECTURE Structure OF dec4to16 IS

---- component declaration ------

COMPONENT dec2to4
    PORT ( w : IN STD LOGIC VECTOR(1 DOWNTO 0) ;
           En : IN STD LOGIC ;
           y : OUT STD LOGIC VECTOR(0 TO 3) ) ;
END COMPONENT ;

SIGNAL m : STD LOGIC VECTOR(0 TO 3) ;
```

```
BEGIN
    Dec1: dec2to4 PORT MAP ( w(3 DOWNTO 2), En, m(0 TO 3) ) ;
    Dec2: dec2to4 PORT MAP ( w(1 DOWNTO 0), m(0), y(0 TO 3) ) ;
    Dec3: dec2to4 PORT MAP ( w(1 DOWNTO 0), m(1), y(4 TO 7)) ;
    Dec4: dec2to4 PORT MAP ( w(1 DOWNTO 0), m(2), y(8 TO 11) ) ;
    Dec5: dec2to4 PORT MAP ( w(1 DOWNTO 0), m(3), y(12 TO 15) ) ;

    Or

    Dec1: dec2to4 PORT MAP ( w(3 DOWNTO 2), En, m(0 TO 3) ) ;

    G1: FOR i IN 0 TO 3 GENERATE
            Dec_right: dec2to4 PORT MAP ( w(1 DOWNTO 0), m(i), y(4*i TO 4*i+3) );
    END GENERATE G1 ;

END Structure ;
```