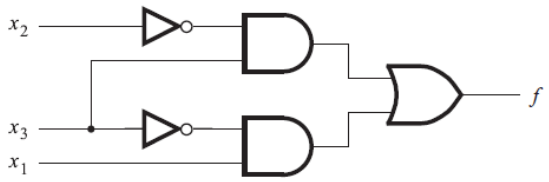
CEG3155: DGD 5
Assignment 1 Solution

Question 1 (Browns 2.8)

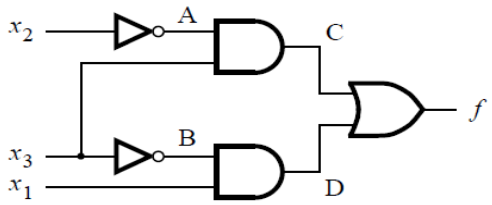
Draw a timing diagram for the circuit in the figure below. Show the waveforms that can be observed on all wires in the circuit.

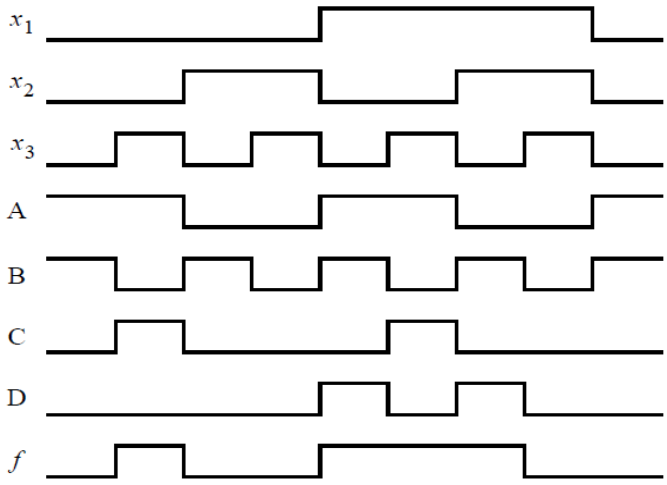


(a) A minimal sum-of-products realization

Solution

2.8. Timing diagram of the waveforms that can be observed on all wires of the circuit:





Question 2 (Browns 2.12)

Use algebraic manipulation to find the minimum sum-of-products expression for the function $f = x1\ x3 + x1\ \overline{x2} + \overline{x1}\ x2\ x3 + \overline{x1}\ \overline{x2}\ \overline{x3}$

Solution

2.12. Derivation of the minimum sum-of-products expression:

$$\begin{aligned} f &= x_1x_3 + x_1\overline{x}_2 + \overline{x}_1x_2x_3 + \overline{x}_1\overline{x}_2\overline{x}_3 \\ &= x_1(\overline{x}_2 + x_2)x_3 + x_1\overline{x}_2(\overline{x}_3 + x_3) + \overline{x}_1x_2x_3 + \overline{x}_1\overline{x}_2\overline{x}_3 \\ &= x_1\overline{x}_2x_3 + x_1x_2x_3 + x_1\overline{x}_2\overline{x}_3 + \overline{x}_1x_2x_3 + \overline{x}_1\overline{x}_2\overline{x}_3 \\ &= x_1x_3 + (x_1 + \overline{x}_1)x_2x_3 + (x_1 + \overline{x}_1)\overline{x}_2\overline{x}_3 \\ &= x_1x_3 + x_2x_3 + \overline{x}_2\overline{x}_3 \end{aligned}$$

Question 3 (Chu 4.1)

Add an enable signal, en, to a 2-to-4 decoder. When en is ' 1 ', the decoder functions as usual. When en is '0', the decoder is disabled and output becomes "0000". Use the conditional signal assignment statement to derive this circuit. Draw the conceptual diagram.

Solution

```
1  ----------------------------------------------------
2  -- a 2-to-4 decoder with enable
3  ----------------------------------------------------
4  library ieee;
5  use ieee.std_logic_1164.all;
6
7  entity decoder4_en is
8     port(
9          sel: in  std_logic_vector(1 downto 0);
10         en: in std_logic;
11         x: out std_logic_vector(3 downto 0)
12    );
13 end decoder4_en ;
14
15 architecture cond_arch of decoder4_en is
16 begin
17      x <= "0000" when (en='0') else
18           "0001" when (sel="00") else
19           "0010" when (sel="01") else
20           "0100" when (sel="10") else
21           "1000";  -- sel="11"
22 end cond_arch ;
```

Question 4 (Chu 4.4)

Consider a comparator with two 8-bit inputs, a and b. The a and b are with the std-logic-vector data type and are interpreted as unsigned integers. The comparator has an output, agtb, which is asserted when a is greater than b. Assume that only a single bit comparator is supported by synthesis software. Derive the circuit with concurrent signal assignment statements.

Solution

```
1  ----------------------------------------------
2  -- a gt b for unsigned number
3  ----------------------------------------------
4  library ieee;
5  use ieee.std_logic_1164.all;
6
7  entity gt_unsigned is
8     port(
9        a,b: in std_logic_vector(7 downto 0);
10       agtb: out std_logic
11    );
12 end gt_unsigned ;
13
14 architecture prim_arch of gt_unsigned is
15 begin
16    -- if a(7) /= b(7), return 1 if a(7)=1,b(7)=0
17    --                  return 0 if a(7)=0,b(7)=1
18    -- otherwise checking next bit
19
20    agtb <= a(7) when (a(7) /= b(7)) else
21            a(6) when (a(6) /= b(6)) else
22            a(5) when (a(5) /= b(5)) else
23            a(4) when (a(4) /= b(4)) else
24            a(3) when (a(3) /= b(3)) else
25            a(2) when (a(2) /= b(2)) else
26            a(1) when (a(1) /= b(1)) else
27            a(0) when (a(0) /= b(0)) else
28            '0';
29 end prim_arch ;
30
```

Question 5

The system has one input called pushbutton (PB) and 4 outputs LED1, LED2, LED3 and LED4. When the PB is not pressed (it is logic zero) only LED1 and LED3 are on. When the push button is pressed (it is logic one), only LED2 and LED4 are on. Implement this system in VHDL using **if** or **case** statement.

Solution

Using if statement:

```
1  library ieee;
2  use ieee.std_logic_1164.all;
3
4  entity Question5 is
5     port(
6        PB: in std_logic;
7        LED: out std_logic_vector(3 downto 0)
8     );
9  end Question5 ;
10
11 architecture Question5_arch of Question5 is
12 begin
13    process(PB)
14       if(PB = '0') then
15          LED <= "1010";
16       else
17          LED <= "0101";
18       end
19    end process;
20 end Question5_arch;
21
```

Using case statement:

```
1  library ieee;
2  use ieee.std_logic_1164.all;
3
4  entity Question5 is
5     port(
6        PB: in std_logic;
7        LED: out std_logic_vector(3 downto 0)
8     );
9  end Question5 ;
10
11 architecture Question5_arch of Question5 is
12 begin
13    process(PB)
14       case PB is
15          when '0' =>
16             LED <= "1010";
17          when '1' =>
18             LED <= "0101";
19          when others =>
20             LED <= "0000";
21       end case;
22    end process;
23 end Question5_arch;
```