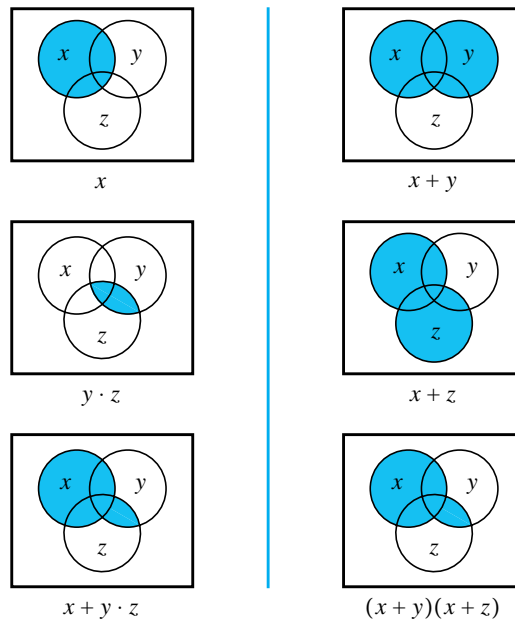# Chapter 2

2.1. The proof is as follows:

$$
\begin{aligned}
(x + y) \cdot (x + z) &= xx + xz + xy + yz \\
&= x + xz + xy + yz \\
&= x(1 + z + y) + yz \\
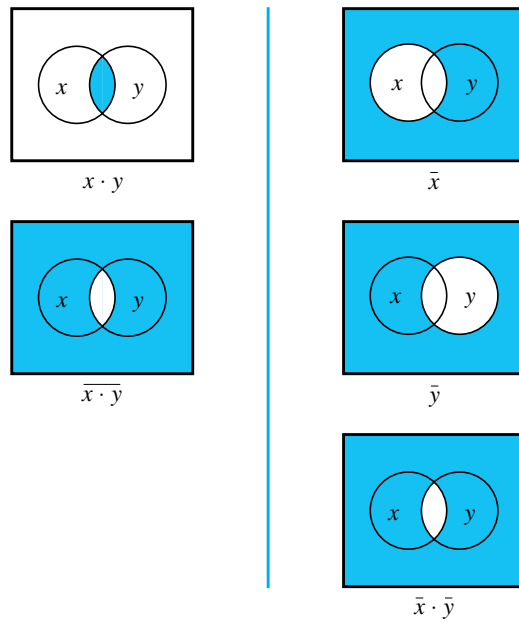&= x \cdot 1 + yz \\
&= x + yz
\end{aligned}
$$

2.2. The proof is as follows:

$$
\begin{aligned}
(x + y) \cdot (x + \overline{y}) &= xx + xy + x\overline{y} + y\overline{y} \\
&= x + xy + x\overline{y} + 0 \\
&= x(1 + y + \overline{y}) \\
&= x \cdot 1 \\
&= x
\end{aligned}
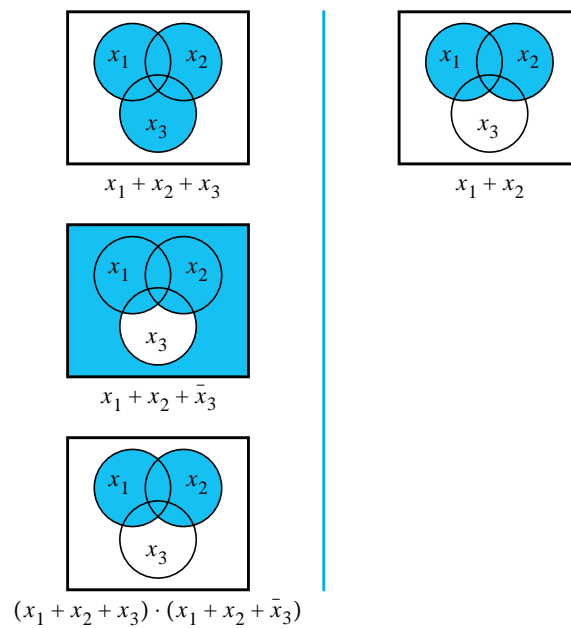$$

2.3. Proof using Venn diagrams:

2.4. Proof of $15a$ using Venn diagrams:

$x \cdot y$

$\bar{x}$

$\overline{x \cdot y}$

$\bar{y}$

$\bar{x} \cdot \bar{y}$

A similar proof is constructed for $15b$.

2.5. Proof using Venn diagrams:

$x_1 + x_2 + x_3$

$x_1 + x_2$

$x_1 + x_2 + \bar{x}_3$

$(x_1 + x_2 + x_3) \cdot (x_1 + x_2 + \bar{x}_3)$

2.6. A possible approach for determining whether or not the expressions are valid is to try to manipulate the left and right sides of an expression into the same form, using the theorems and properties presented in section 2.5. While this may seem simple, it is an awkward approach, because it is not obvious what target form one should try to reach. A much simpler approach is to construct a truth table for each side of an expression. If the truth tables are identical, then the expression is valid. Using this approach, we can show that the answers are:

(a) Yes
(b) Yes
(c) No

2.7. Timing diagram of the waveforms that can be observed on all wires of the circuit:
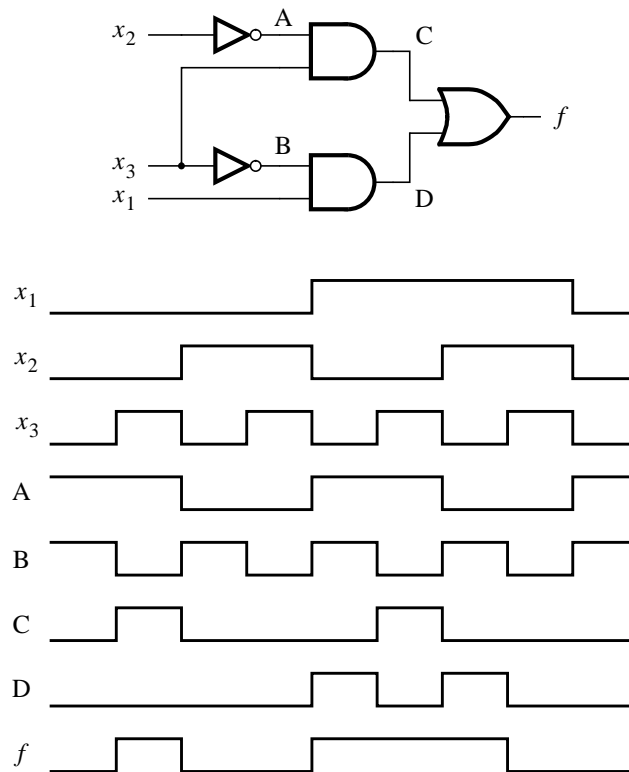
2.8. Timing diagram of the waveforms that can be observed on all wires of the circuit:
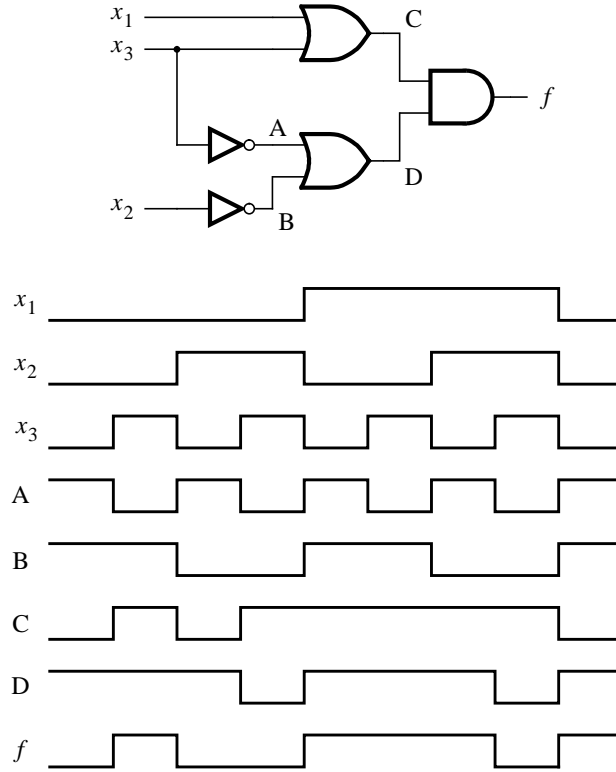
2.9. Starting with the canonical sum-of-products for f get

$$
\begin{aligned}
f &= \overline{x}_1\overline{x}_2 x_3 + \overline{x}_1 x_2\overline{x}_3 + \overline{x}_1 x_2 x_3 + x_1\overline{x}_2\overline{x}_3 + x_1\overline{x}_2 x_3 + x_1 x_2\overline{x}_3 + x_1 x_2 x_3 \\
&= x_1(\overline{x}_2\overline{x}_3 + \overline{x}_2 x_3 + x_2\overline{x}_3 + x_2 x_3) + x_2(\overline{x}_1\overline{x}_3 + \overline{x}_1 x_3 + x_1\overline{x}_3 + x_1 x_3) \\
&\quad + x_3(\overline{x}_1\overline{x}_2 + \overline{x}_1 x_2 + x_1\overline{x}_2 + x_1 x_2) \\
&= x_1(\overline{x}_2(\overline{x}_3 + x_3) + x_2(\overline{x}_3 + x_3)) + x_2(\overline{x}_1(\overline{x}_3 + x_3) + x_1(\overline{x}_3 + x_3)) \\
&\quad + x_3(\overline{x}_1(\overline{x}_2 + x_2) + x_1(\overline{x}_2 + x_2)) \\
&= x_1(\overline{x}_2 \cdot 1 + x_2 \cdot 1) + x_2(\overline{x}_1 \cdot 1 + x_1 \cdot 1) + x_3(\overline{x}_1 \cdot 1 + x_1 \cdot 1) \\
&= x_1(\overline{x}_2 + x_2) + x_2(\overline{x}_1 + x_1) + x_3(\overline{x}_1 + x_1) \\
&= x_1 \cdot 1 + x_2 \cdot 1 + x_3 \cdot 1 \\
&= x_1 + x_2 + x_3
\end{aligned}
$$

2.10. The canonical product-of-sums for f is

$$
\begin{aligned}
f &= (x_1 + x_2 + x_3)(x_1 + x_2 + \overline{x}_3)(x_1 + \overline{x}_2 + x_3)(x_1 + \overline{x}_2 + \overline{x}_3) \cdot \\
&\quad (\overline{x}_1 + x_2 + x_3)(\overline{x}_1 + x_2 + \overline{x}_3)(\overline{x}_1 + \overline{x}_2 + x_3)
\end{aligned}
$$

It can be manipulated as follows:

$$
\begin{aligned}
f &= (x_1(1 + x_2 + x_3)(1 + x_2 + \overline{x}_3)(1 + \overline{x}_2 + x_3)(1 + \overline{x}_2 + \overline{x}_3)) \cdot \\
&\quad (x_2(x_1 + 1 + x_3)(x_1 + 1 + \overline{x}_3)(\overline{x}_1 + 1 + x_3)(\overline{x}_1 + 1 + \overline{x}_3)) \cdot \\
&\quad (x_3(x_1 + x_2 + 1)(x_1 + \overline{x}_2 + 1)(\overline{x}_1 + x_2 + 1)(\overline{x}_1 + \overline{x}_2 + 1))
\end{aligned}
$$

$$= (x_1 \cdot 1 \cdot 1 \cdot 1 \cdot 1)(x_2 \cdot 1 \cdot 1 \cdot 1 \cdot 1)(x_3 \cdot 1 \cdot 1 \cdot 1 \cdot 1)$$
$$= x_1 x_2 x_3$$

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

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

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

$$
\begin{aligned}
f &= x_1 \overline{x}_2 \overline{x}_3 + x_1 x_2 x_4 + x_1 \overline{x}_2 x_3 \overline{x}_4 \\
&= x_1 \overline{x}_2 \overline{x}_3(\overline{x}_4 + x_4) + x_1 x_2 x_4 + x_1 \overline{x}_2 x_3 \overline{x}_4 \\
&= x_1 \overline{x}_2 \overline{x}_3 \overline{x}_4 + x_1 \overline{x}_2 \overline{x}_3 x_4 + x_1 x_2 x_4 + x_1 \overline{x}_2 x_3 \overline{x}_4 \\
&= x_1 \overline{x}_2 \overline{x}_3 + x_1 \overline{x}_2(\overline{x}_3 + x_3)\overline{x}_4 + x_1 x_2 x_4 \\
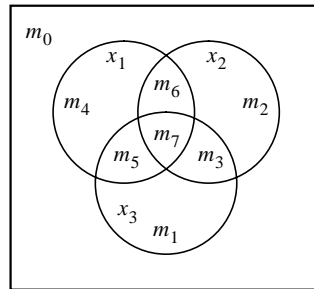&= x_1 \overline{x}_2 \overline{x}_3 + x_1 \overline{x}_2 \overline{x}_4 + x_1 x_2 x_4
\end{aligned}
$$

2.13. The simplest POS expression is derived as

$$
\begin{aligned}
f &= (x_1 + x_3 + x_4)(x_1 + \overline{x}_2 + x_3)(x_1 + \overline{x}_2 + \overline{x}_3 + x_4) \\
&= (x_1 + x_3 + x_4)(x_1 + \overline{x}_2 + x_3)(x_1 + \overline{x}_2 + x_3 + x_4)(x_1 + \overline{x}_2 + \overline{x}_3 + x_4) \\
&= (x_1 + x_3 + x_4)(x_1 + \overline{x}_2 + x_3)((x_1 + \overline{x}_2 + x_4)(x_3 + \overline{x}_3)) \\
&= (x_1 + x_3 + x_4)(x_1 + \overline{x}_2 + x_3)(x_1 + \overline{x}_2 + x_4) \cdot 1 \\
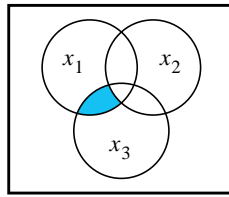&= (x_1 + x_3 + x_4)(x_1 + \overline{x}_2 + x_3)(x_1 + \overline{x}_2 + x_4)
\end{aligned}
$$

2.14. Derivation of the minimum product-of-sums expression:

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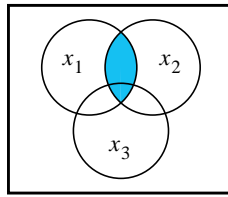\end{aligned}
$$

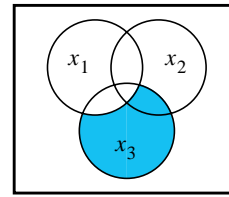2.15. (a) Location of all minterms in a 3-variable Venn diagram:

(b) For $f = x_1\overline{x}_2 x_3 + x_1 x_2 + \overline{x}_1 x_3$ have:

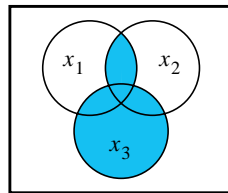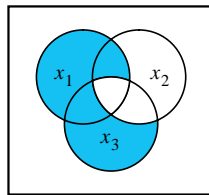$$x_1 \cdot \overline{x}_2 \cdot x_3 \qquad\qquad x_1 \cdot x_2 \qquad\qquad \overline{x}_1 \cdot x_3$$

Therefore, f is represented as:
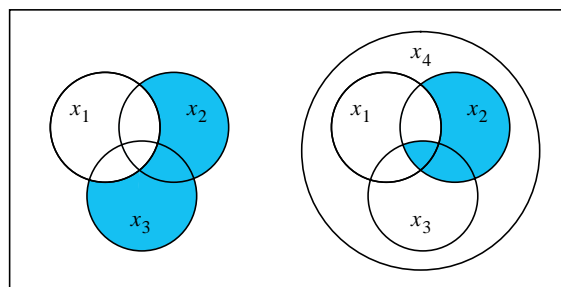

$$f = x_3 + x_1 x_2$$

2.16. The function in Figure 2.18 in Venn diagram form is:

2.17. In Figure P2.1a it is possible to represent only 14 minterms. It is impossible to represent the minterms $\overline{x}_1\overline{x}_2 x_3 x_4$ and $x_1 x_2 \overline{x}_3 \overline{x}_4$.
In Figure P2.1b, it is impossible to represent the minterms $x_1 x_2 \overline{x}_3 \overline{x}_4$ and $x_1 x_2 x_3 \overline{x}_4$.

2.18. Venn diagram for $f = \overline{x}_1 \overline{x}_2 x_3 \overline{x}_4 + x_1 x_2 x_3 x_4 + \overline{x}_1 x_2$ is

2.19. The simplest SOP implementation of the function is

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

2.20. The simplest SOP implementation of the function is

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

Another possibility is

$$
f = \overline{x}_1 x_3 + x_1 \overline{x}_3 + x_1 x_2
$$

2.21. The simplest POS implementation of the function is

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

2.22. The simplest POS implementation of the function is

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

2.23. The lowest cost circuit is defined by

$$
f(x_1, x_2, x_3) = x_1 x_2 + x_1 x_3 + x_2 x_3
$$

2.24. The truth table that corresponds to the timing diagram in Figure P2.3 is

x_1	x_2	x_3	f
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

The simplest SOP expression is $f = \overline{x}_1\overline{x}_2\overline{x}_3 + \overline{x}_1 x_2 x_3 + x_1\overline{x}_2 x_3 + x_1 x_2\overline{x}_3$.

2.25. The truth table that corresponds to the timing diagram in Figure P2.4 is

x_1	x_2	x_3	f
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

The simplest SOP expression is derived as follows:

$$
\begin{aligned}
f &= \overline{x}_1\overline{x}_2 x_3 + \overline{x}_1 x_2\overline{x}_3 + \overline{x}_1 x_2 x_3 + x_1\overline{x}_2\overline{x}_3 + x_1 x_2 x_3 \\
&= \overline{x}_1(\overline{x}_2 + x_2)x_3 + \overline{x}_1\overline{x}_2(\overline{x}_3 + x_3) + (\overline{x}_1 + x_1)x_2 x_3 + x_1\overline{x}_2\overline{x}_3 \\
&= \overline{x}_1 \cdot 1 \cdot x_3 + \overline{x}_1\overline{x}_2 \cdot 1 + 1 \cdot x_2 x_3 + x_1\overline{x}_2\overline{x}_3 \\
&= \overline{x}_1 x_3 + \overline{x}_1\overline{x}_2 + x_2 x_3 + x_1\overline{x}_2\overline{x}_3
\end{aligned}
$$

2.26. (a)

x_1	x_0	y_1	y_0	f
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

(b)　$f = (x_1 + \overline{y}_1)(\overline{x}_1 + y_1)(x_0 + \overline{y}_0)(\overline{x}_0 + y_0)$

2.27. (a)

x_1	x_0	y_1	y_0	f
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

(b) The canonical SOP expression is

$$
\begin{aligned}
f &= \overline{x}_1\overline{x}_0\overline{y}_1\overline{y}_0 + \overline{x}_1 x_0\overline{y}_1\overline{y}_0 + \overline{x}_1 x_0\overline{y}_1 y_0 + x_1\overline{x}_0\overline{y}_1\overline{y}_0 + x_1\overline{x}_0\overline{y}_1 y_0 + x_1\overline{x}_0 y_1\overline{y}_0 \\
&\quad + x_1 x_0\overline{y}_1\overline{y}_0 + x_1 x_0\overline{y}_1 y_0 + x_1 x_0 y_1\overline{y}_0 + x_1 x_0 y_1 y_0
\end{aligned}
$$

(c) The simplest SOP expression is

$$f = x_1 x_0 + \overline{y}_1 \overline{y}_0 + x_1 \overline{y}_0 + x_0 \overline{y}_1$$

2.30.
```
                    LIBRARY ieee ;
                    USE ieee.std_logic_1164.all ;

                    ENTITY prob2_30 IS
                        PORT ( x1, x2, x3, x4  : IN    STD_LOGIC ;
                               f1, f2          : OUT  STD_LOGIC ) ;
                    END prob2_30 ;

                    ARCHITECTURE LogicFunc OF prob2_30 IS
                    BEGIN
                        f1 <= (x1 AND NOT x3) OR (x2 AND NOT x3) OR
                              NOT x3 AND NOT x4) OR (x1 AND x2) OR
                              x1 AND NOT x4) ;
                        f2 <= (x1 OR NOT x3) AND (x1 OR x2 OR NOT x4) AND
                              x2 OR NOT x3 OR NOT x4) ;
                    END LogicFunc ;
```

2.31. For the functions given in this question, it is not true that $f_1 = \overline{f}_2$. The function f_1 is given in the form (SOP-term) AND (SOP-term). If these same two SOP terms are used for the *different* function $f_1 = $ (SOP-term) OR (SOP-term) then for this new f_1 it is true that $f_1 = \overline{f}_2$. Complete VHDL code using this new function f_1 is shown below.

```
                    LIBRARY ieee ;
                    USE ieee.std_logic_1164.all ;

                    ENTITY prob2_31 IS
                        PORT ( x1, x2, x3, x4  : IN    STD_LOGIC ;
                               f1, f2          : OUT  STD_LOGIC ) ;
                    END prob2_31 ;

                    ARCHITECTURE LogicFunc OF prob2_31 IS
                    BEGIN
                        f1 <= ((x1 AND x3) OR (NOT x1 AND NOT x3)) OR
                              ((x2 AND x4) OR (NOT x2 AND NOT x4)) ;
                        f2 <= (x1 AND x2 AND NOT x3 AND NOT x4) OR
                              (NOT x1 AND NOT x2 AND x3 AND x4) OR
                              (x1 AND NOT x2 AND NOT x3 AND x4) OR
                              (NOT x1 AND x2 AND x3 AND NOT x4);
                    END LogicFunc ;
```