implicant: \( x'y'z' \)
minterm: \( x'y'z' \)

**Exercise 5.6**

(a) \( E(w, x, y, z) = \prod M(1, 3, 4, 7, 10, 13, 14, 15) = \sum m(0, 2, 5, 6, 8, 9, 11, 12) \)

minimal sum of products: \( w'y'z' + wx'z' + w'x'z' + w'yz' + w'xy'z' \)

minimal product of sums: \((w + x + z')(w' + y' + z)(x' + y' + z')(w' + x' + z')(w + x' + y + z)\)
(b) $E(w, x, y, z) = \sum m(0, 4, 5, 9, 11, 14, 15), dc(w, x, y, z) = \sum m(2, 8)$

\[
\begin{array}{c|c|c|c}
| & z & 0 & 0 \\
\hline
w & 1 & 0 & 0 \\
\hline
w & 1 & 1 & 0 \\
\hline
w & 0 & 0 & 1 \\
\hline
w & 0 & 0 & 1 \\
\hline
\end{array}
\]

minimal SP: $w'y'z' + w'y'x + wx'z + wxy$

\[
\begin{array}{c|c|c|c}
| & z & 0 & 0 \\
\hline
w & 1 & 0 & 0 \\
\hline
w & 1 & 1 & 0 \\
\hline
w & 0 & 0 & 1 \\
\hline
w & 0 & 0 & 1 \\
\hline
\end{array}
\]

minimal PS: $(w+x+z')(w+y')(x+y'+z)(w'+x'+y)$

(c) $E(x, y, z) = \sum m(0, 1, 4, 6) = \prod M(2, 3, 5, 7)$

\[
\begin{array}{c|c|c|c}
| & z & 0 & 0 \\
\hline
x & 1 & 1 & 0 \\
\hline
x & 1 & 0 & 0 \\
\hline
\end{array}
\]

minimal sum of products: $x'y' + xz'$

\[
\begin{array}{c|c|c|c}
| & z & 0 & 0 \\
\hline
x & 1 & 1 & 0 \\
\hline
x & 1 & 0 & 0 \\
\hline
\end{array}
\]

minimal product of sums: $(x + y')(x' + z')$
Exercise 5.7
\[ f(w, x, y, z) = \text{one}_\text{set}(1, 5, 7, 8, 9, 10, 14) \]

(a) prime implicants are:
\( (w + z), (w + x + y'), (x + y' + z'), (w' + y' + z'), (w' + x' + y'), (x' + y + z) \)

(b) essential prime implicate is: \((w + z)\)

(c) a minimal product of sums expression that implements \( f(w, x, y, z) \) is:
\[ E(w, x, y, z) = (w + z)(x + y' + z')(w' + x' + z')(x' + y + z) \]

the solution is not unique because there are other ways to cover the 0-cells (not covered by the essential prime implicate) with the same number of terms.

Exercise 5.8
Input: \((a, b, c, d)\), with \(a, b, c, d \in \{0, 1\}\)
Output: \(y \in \{0, 1\}\)
Function:
\[
y = \begin{cases} 
1 & \text{if } (8a + 4b + 2c + d) \text{ is prime} \\
0 & \text{otherwise}
\end{cases}
\]

<table>
<thead>
<tr>
<th>input value</th>
<th>abcd</th>
<th>y</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0000</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0001</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0010</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>0011</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>0100</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0101</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>0110</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>0111</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>1000</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>1001</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>1010</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>1011</td>
<td>1</td>
</tr>
<tr>
<td>12</td>
<td>1100</td>
<td>0</td>
</tr>
<tr>
<td>13</td>
<td>1101</td>
<td>1</td>
</tr>
<tr>
<td>14</td>
<td>1110</td>
<td>0</td>
</tr>
<tr>
<td>15</td>
<td>1111</td>
<td>0</td>
</tr>
</tbody>
</table>
From the Kmap we get the following prime implicants: $a'b'd, \ b'cd, \ a'b'c, \ a'cd,$ and $bc'd$

The essential prime implicants are: $b'cd, \ a'b'c,$ and $bc'd$

A minimal sum of products for function $y$ is:

$$y = b'cd + a'b'c + bc'd + a'cd$$

and the gate network that implements this expression is shown in Figure 5.1.

Figure 5.1: AND-OR gate network for “prime detector” (Exercise 5.8)

From the Kmap we get the following prime implicants: $(b' + d), (a' + d), (b + c), (c + d)$ and $(a' + b' + c')$. Only the $(c + d)$ prime implicate is not essential. So, the minimal product of sums in this case is:

$$y = (b' + d)(a' + d)(b + c)(a' + b' + c')$$

and the gate network that implements this expression is shown in Figure 5.2. Notice that the cost of the product of sums is lower.
Figure 5.2: OR-AND gate network for “prime detector” (Exercise 5.8)

Exercise 5.9
To repeat the Exercise 5.8 using the Quine-McCluskey minimization method we make use of two tables. The first table is used to obtain the prime implicants and the second to select the minimum cover.

The one-set for this function is one-set = \{2, 3, 5, 7, 11, 13\}

<table>
<thead>
<tr>
<th>Minterms</th>
<th>3-literal Prods</th>
<th>2-literal Prods</th>
<th>1-literal Prods</th>
</tr>
</thead>
<tbody>
<tr>
<td>0010 N</td>
<td>001</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0011 N</td>
<td>-101</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0101 N</td>
<td>-011</td>
<td>0-11</td>
<td></td>
</tr>
<tr>
<td>0111 N</td>
<td></td>
<td>01-1</td>
<td></td>
</tr>
<tr>
<td>1011 N</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1101 N</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The second table consider the prime-implicants obtained in the previous table:

<table>
<thead>
<tr>
<th>Prime Implicants</th>
<th>2</th>
<th>3</th>
<th>5</th>
<th>11</th>
<th>13</th>
<th>Essential</th>
</tr>
</thead>
<tbody>
<tr>
<td>001-</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td>•</td>
</tr>
<tr>
<td>-101</td>
<td></td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td>•</td>
</tr>
<tr>
<td>-011</td>
<td></td>
<td></td>
<td>x</td>
<td></td>
<td></td>
<td>•</td>
</tr>
<tr>
<td>0-11</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>01-1</td>
<td></td>
<td></td>
<td>x</td>
<td></td>
<td>x</td>
<td></td>
</tr>
</tbody>
</table>

Based on the table, the function is represented by 3 essential terms and the minterm 7 must be covered with either 0-11 or 01-1. The minimal SP expressions are:

\[ y = d'b'c + bc'd + b'cd + d'cd \]

or

\[ y = d'b'c + bc'd + b'cd + d'bd \]
Exercise 5.10 A high-level description for this system is:
Input: \( a, b, c, d, e \in \{0, 1\} \)
Output: \( f \in \{0, 1\} \)
Function:
\[
f = \begin{cases} 
1 & \text{if } (a + b + c + d + e) \geq 3 \\
0 & \text{otherwise}
\end{cases}
\]
Using Quine-McCluskey minimization method we obtain the following table:

<table>
<thead>
<tr>
<th>Minterms</th>
<th>4-literal products</th>
<th>3-literal products</th>
</tr>
</thead>
<tbody>
<tr>
<td>00111 N</td>
<td>0-111 N</td>
<td>--111</td>
</tr>
<tr>
<td>01011 N</td>
<td>-0111 N</td>
<td>-1-11</td>
</tr>
<tr>
<td>10011 N</td>
<td>01-11 N</td>
<td>1--11</td>
</tr>
<tr>
<td>01101 N</td>
<td>10-11 N</td>
<td>-11-1</td>
</tr>
<tr>
<td>10101 N</td>
<td>1-011 N</td>
<td>1-1-1</td>
</tr>
<tr>
<td>11001 N</td>
<td>011-1 N</td>
<td>11--1</td>
</tr>
<tr>
<td>01110 N</td>
<td>-1101 N</td>
<td>-11--1</td>
</tr>
<tr>
<td>10110 N</td>
<td>101-1 N</td>
<td>1--1-1</td>
</tr>
<tr>
<td>11010 N</td>
<td>1-101 N</td>
<td>11-1--1</td>
</tr>
<tr>
<td>11100 N</td>
<td>110-1 N</td>
<td>111--1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

To obtain a cover for the minterms we use a second table to identify the essential prime implicants, as shown below:

| 7  | 11 | 13 | 14 | 15 | 19 | 21 | 22 | 23 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | Essential Prime |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----------------|
| --111 | x  | x  | x  | x  |     |    |    |    |    |    |    |    |    |    |    | \( \bullet \)   |
| -1-11 | x  | x  | x  | x  | x  |    |    |    |    |    |    |    |    |    |    | \( \bullet \)   |
| 1--11 | x  | x  | x  | x  |     |    |    |    |    |    |    |    |    |    |    | \( \bullet \)   |
| -11-1 | x  | x  | x  | x  | x  | x  |    |    |    |    |    |    |    |    |    | \( \bullet \)   |
| 1-1-1 | x  | x  | x  | x  | x  | x  | x  |    |    |    |    |    |    |    |    | \( \bullet \)   |
| 11--1 | x  | x  | x  | x  | x  | x  | x  |    |    |    |    |    |    |    |    | \( \bullet \)   |
| -111- | x  | x  | x  | x  | x  | x  | x  | x  |    |    |    |    |    |    |    | \( \bullet \)   |
| 1-1-1 | x  | x  | x  | x  | x  |     |    |    |    |    |    |    |    |    |    | \( \bullet \)   |
| 11-1- | x  | x  | x  | x  |     |    |    |    |    |    |    |    |    |    |    | \( \bullet \)   |
| 111-- | x  | x  | x  | x  |     |    |    |    |    |    |    |    |    |    |    | \( \bullet \)   |

All the terms obtained in the first table are essential prime implicants. The minimal expression is:
\[
f = abc + abd + abc + acd + ace + ade + bcd + bce + bde + cde
\]
and the corresponding two-level gate network is shown in Figure 5.3 on page 63. Note that the OR gate has 10 inputs, which might make a two-level implementation impractical.

![Majority function - Exercise 5.10](image)

**Exercise 5.11** A high-level specification for the error detector is:

Input: $x$ is a 2-out-of-5 code represented as $x = (a, b, c, d, e)$, where $a, b, c, d, e \in \{0, 1\}$.

Output: $f \in \{0, 1\}$.

Function:

$$f = \begin{cases} 0 & \text{if the number of 1s in the input is 2} \\ 1 & \text{otherwise} \end{cases}$$

The synthesis of this function using Quine-McCluskey minimization method is shown in Table 5.1. The obtained 3-literal products generate a Prime-implicant chart similar to the one shown in Exercise 5.10, where it was concluded that all products are essential. The minimal sum of products is:

$$f = a'b'c'd' + abc + abd + acd + bcd + abe + ace + ade + bce + bde + cde + d'b'c'e' + d'b'd'c' + d'c'd'e' + b'c'd'e'$$

The gate network that implements the function $f$ is shown in Figure 5.4. Note that the OR gate has 15 inputs, which might make a two-level implementation impractical.