Chapter 4
Complements in Combinational Network Design

Abstract In this chapters, we overcome the limitations of the Karnaugh maps, whose application becomes impractical when applied to expressions with more than four/five variables. We present the Quine–McCluskey method, the first algorithms for minimizing Boolean expressions developed by Willard V. Quine and improved by Edward J. McCluskey. We present both the methods for synthesize single and multiple functions at the same time.

In past chapters, we addressed minimizing Boolean expressions, i.e., using Karnaugh maps to find an expression equivalent to the original but with fewer prime implicants. Unfortunately, the map method can only be applied to expressions with a maximum of four variables, extendable to five variables through 3-D maps or, in limited cases, even more through entered variables. For more than four variables, we cannot use "manual" methods but rather algorithmic methods implemented on a computer.

4.1 Minimizing Boolean Expressions with the Quine–McCluskey Method

One of the first algorithms for minimizing Boolean expressions was developed by Willard V. Quine (1908–2000) and improved by Edward J. McCluskey (1929–2016)[1] and is known as the "Quine–McCluskey Method" (hereinafter referred to as "QM–M").

The QM–M is an algorithm that translates the manual procedure of the Karnaugh maps, and it is made up of two phases. The first is the "expansion" phase where all the implicants of the function to be minimized (Karnaugh's "cubes") are generated. The prime implicants are identified, and the others are eliminated. The

[1]E.J.McCluskey, Minimization of Boolean Functions, The Bell System Technical Journal, November 1956.

G. Donzellini et al., *Introduction to Digital Systems Design*,
https://doi.org/10.1007/978-3-319-92804-3_4

second phase known as "covering" is where the smallest number of prime implicants needed to make the function equivalent to the starting function is chosen. That is, all the *minterms* of the function are "covered." There are tables for these two phases that help keep track of the steps in the algorithm and are easy to calculate.

4.1.1 The Expansion Phase

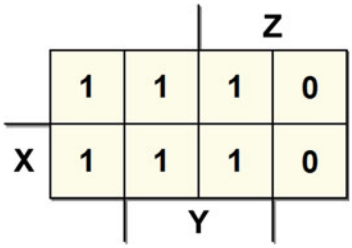
In the preparatory phase, the QM–M uses a simple approach to identify *minterms*: an n variable *minterm* is identified by an n-bit binary number where a direct variable is denoted with the value 1 and a negated variable, with 0. Let's look at this three-variable function as an example:

$$F(X, Y, Z) = \overline{X}\ \overline{Y}\ \overline{Z} + \overline{X}\ Y\ \overline{Z} + \overline{X}\ Y\ Z + X\ \overline{Y}\ \overline{Z} + X\ Y\ \overline{Z} + X\ Y\ Z$$

Minterms are identified by the binary numbers (000, 010, 011, 100, 110, 111), which in decimal are (0, 2, 3, 4, 6, 7), so we can use this encoding to write the compact form of the function:

$$F(X, Y, Z) = \Sigma(0, 2, 3, 4, 6, 7)$$

The latter can be described through the map below.



From here, we can easily derive the minimal expression:

$$F(X, Y, Z) = Y + \overline{Z}$$

At the beginning of the expansion phase, the QM–M lists all the *minterms* in a table, respecting the order that we will outline below. It then proceeds to pair them to obtain all the possible implicants with one variable less than the starting *minterm*.

3 Variables		
Terms	$X\ Y\ Z$	P
0	000	
2	010	
4	100	
3	011	
6	110	
7	111	

2 Variables		
Terms	$X\ Y\ Z$	P

1 Variable		
Terms	$X\ Y\ Z$	P

For example, *minterms* $m0$ and $m2$ can be combined, thus

$$\overline{X}\ \overline{Y}\ \overline{Z} + \overline{X}\ Y\ \overline{Z} = \overline{X}\ \overline{Z}\ (\overline{Y} + Y) = \overline{X}\ \overline{Z}$$

In QM–M notation, that would be:

$$\begin{array}{c} 000 + 010 \rightarrow\ 0{-}0 \\ 0\ +\ 2\ \rightarrow (0,2) \end{array}$$

The symbol "–" is used to show that the variable was eliminated through simplification.

When we combine all the possible *minterm* pairs, we get a new table with all the two-variable implicants (one variable less than originally).

3 Variables		
Terms	$X\ Y\ Z$	P
0	000	∨
2	010	∨
4	100	∨
3	011	∨
6	110	∨
7	111	∨

2 Variables		
Terms	$X\ Y\ Z$	P
0, 2	0–0	
0, 4	–00	
2, 3	01–	
2, 6	–10	
4, 6	1–0	
3, 7	–11	
6, 7	11–	

1 Variable		
Terms	$X\ Y\ Z$	P

At this point, we should analyze two expedients that have been used in the first phase of the algorithm.

The first expedient refers to the initial order of the *minterms* that are grouped by their number of negated variables, that is, the number of 0s in the corresponding binary number.[2] Group 1 has the *minterms* with all the negated variables (000). Group 2 has those with two negated variables (010, 100) and so on. It is actually impossible to combine *minterms* that differ by two or more variables, for example $\overline{X}\ \overline{Y}\ \overline{Z}$ (000) and $\overline{X}\ Y\ Z$ (011), because we would obtain no simplification. It is only possible to

[2]In the table, the groups are separated by one continuous line.

combine copies of *minterms* that differ by just one variable which is negative in one of them and direct in the other, for example: $X\ Y\ \overline{Z}$ (110) and $X\ Y\ Z$ (111). In other words, two *minterms* can be combined only if the corresponding binary numbers have a Hamming distance of 1. By grouping the numbers that contain the same number of 0s, we get homogeneous groups and can reduce the number of comparisons. Instead of comparing all the possible copies of *minterms*, we can check only those belonging to adjacent groups, those with smaller Hamming distances. The *minterm* of group 1 (000) can actually only be combined with that of group 2 (010, 100), which in turn can only be combined with that of group 3 (011, 110) and so on.[3] Note that not all combinations are possible: the term (100) belonging to group 2 can be combined with (110) but not with (011) from group 3 because their Hamming distance is two.

The second expedient will keep track of terms (implicants) that have been combined. If two implicants are combined to obtain an implicant with one less variable it means that those implicants were not prime and should not be considered in the covering phase. We know that a non-prime implicant can be substituted with a prime implicant if the prime covers it. Column P in the table does just this: there is an indication that the term has been combined so it is non-prime and can be overlooked in the covering phase.

By applying the same procedure to two-variable implicants, we get implicants where an additional variable has been eliminated.

3 Variables		
Terms	$X\ Y\ Z$	P
0	000	∨
2	010	∨
4	100	∨
3	011	∨
6	110	∨
7	111	∨

2 Variables		
Terms	$X\ Y\ Z$	P
0, 2	0–0	∨
0, 4	–00	∨
2, 3	01–	∨
2, 6	–10	∨
4, 6	1–0	∨
3, 7	–11	∨
6, 7	11–	∨

1 Variable		
Terms	$X\ Y\ Z$	P
0, 2, 4, 6	– –0	P_0
2, 3, 6, 7	–1–	P_1

Note that when combining two-variable implicants, we have two different ways to get the same one-variable implicant:

$$\begin{array}{c} (0,2) + (4,6) \rightarrow (0,2,4,6) \\ 0{-}0 \ + \ 1{-}0 \ \rightarrow \ {-}{-}0 \end{array}$$

or

$$\begin{array}{c} (0,4) + (2,6) \rightarrow (0,4,2,6) \\ {-}00 + {-}10 \ \rightarrow \ {-}{-}0 \end{array}$$

[3] If we had to compare all the possible pairs among n terms to see if they could be combined, we would have to make more than $n(n-1)/2$ comparisons, i.e., each term with every other bidirectionally.

Clearly, implicant $(0, 2, 4, 6)$ and implicant $(0, 4, 2, 6)$ identify the same term because they correspond to $(--0)$, which is the term $\overline{Z}$. This is why only one of these will be brought over to the right-most table.

When we can combine the implicants no further, as in this case where there are only two uncombined terms altogether, the expansion phase ends and all the unflagged terms are prime implicants.

We can now list the starting function's prime implicants, which are:

$$\begin{aligned} P0 &= (0, 2, 4, 6) = --0 = \overline{Z} \\ P1 &= (2, 3, 6, 7) = -1- = Y. \end{aligned}$$

4.1.2 The Covering Phase

If the goal in the expansion phase is to find all the prime implicants, the goal of the covering phase is to identify the lowest number of prime implicants that can cover the starting function. We must therefore be sure that all the *minterms* that defined the function to be minimized are covered by at least one of the identified prime implicants. We must also be sure to use as few implicants as possible to obtain the coverage.

To reach these goals, the QM–M used a "covering table" where the columns show all the *minterms* and the rows show all the prime implicants identified in the expansion phase. The "Xs" in the m_i column and in the P_j row show that m_i is covered by P_j:

	m_0	m_2	m_3	m_4	m_6	m_7
P_0	X	X		X	X	
P_1		X	X		X	X

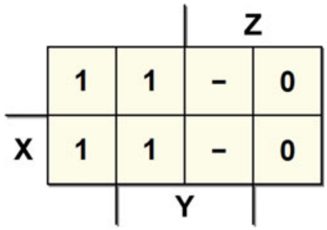
In this case, it is easy to deduce that both the prime implicants are needed to cover the function, so the minimized function is:

$$F(X, Y, Z) = P_0 + P_1 = \overline{Z} + Y$$

Note that if a column contains just one "X," it means that there is only one prime implicant that can cover the corresponding *minterm*. In this case, the implicant is a essential prime implicant because without it, the function could not be completely covered. In this last example, both prime implicants are essential because *minterms* m_0 and m_4 are covered only by P_0, while *minterms* m_3 and m_7 are only covered by P_1. Further on, we will see that covering tables can be much more complex than this, so after selecting the essential prime implicants, we will need to use an algorithm to select the remaining ones to achieve minimum coverage.

4.1.3 Incompletely Specified Functions

We often need to deal with incompletely specified functions like the one represented in the map below:



which is written in QM–M notation as:

$$F(X, Y, Z) = \Sigma(0, 2, 4, 6) + d(3, 7)$$

where $d()$ groups all the *minterms* corresponding to don't-care.

Here, the QM–M could be applied by simply treating the don't-cares as 1s in the expansion phase and as 0s in the covering phase. The basic idea is this: in the expansion phase whenever larger cubes are constructed, (implicants with ever fewer variables) it makes sense to use as many *minterms* as possible to raise the possibilities of simplification. In the covering phase, we want to avoid covering a *minterm* if it is not strictly necessary. Avoiding to use it in the coverage phase, we try to prevent the *minterm* from making any prime implicant superfluous.

For the function above, the expansion phase is identical to the previous case while the covering phase uses a table with no m_3 or m_7 don't-care *minterms*.

	m_0	m_2	m_4	m_6
P_0	X	X	X	X
P_1		X		X

From this table, we can immediately see that P_0 is an essential prime implicant (due to m_0 and m_4) and it is also able to cover the function, giving us:

$$F(X, Y, Z) = P_0 = \overline{Z}$$

as expected.

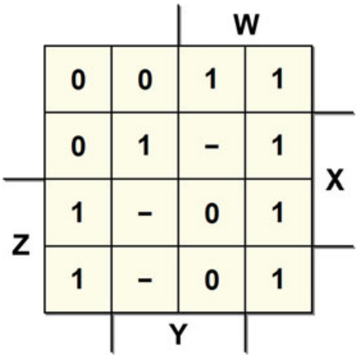
4.1.4 Optimizing the Covering Phase

The covering phase can be particularly complicated when there is a large number of prime implicants.

Let's assume that after the expansion phase and after identifying the essential prime implicants, k prime implicants remain $\{P_0, \ldots, P_{k-1}\}$ from which we will choose the minimum coverage.

To obtain minimum coverage, we must check through $2^k - 1$ different cases, all the possible combinations with 1, 2, 3, or 4 prime implicants, and so on until finding the right one: $\{P_0\}$, $\{P_1\}$, ..., $\{P_k\}$, $\{P_0, P_1\}$, $\{P_0, P_2\}$, $\{P_0, P_3\}$, and so on until $\{P_0, \ldots, P_{k-1}\}$.

The QM–M offers a more efficient alternative that usually requires far fewer comparisons. Let's assume we want to minimize the following function:



$$F(X, Y, W, Z) = \Sigma(1, 2, 3, 6, 9, 10, 11, 12) + d(5, 13, 14)$$

The expansion phase begins with grouping terms, as explained previously. Group 1 contains *minterms* with three negated variables, group 2 with two and group 3 with one, as shown in the following tables.

4 Variables		
Terms	$XYWZ$	P
1	0001	
2	0010	
3	0011	
5	0101	
6	0110	
9	1001	
10	1010	
12	1100	
11	1011	
13	1101	
14	1110	

3 Variables		
Terms	$XYWZ$	P

2 Variables		
Terms	$XYWZ$	P

Now we can begin comparing all the group 1 and group 2 terms. When it is possible to combine two four-variable terms, we get three-variable implicants.

4 Variables		
Terms	$XYWZ$	P
1	0001	∨
2	0010	∨
3	0011	∨
5	0101	∨
6	0110	∨
9	1001	∨
10	1010	∨
12	1100	
11	1011	
13	1101	
14	1110	

3 Variables		
Terms	$XYWZ$	P
1, 3	00−1	
1, 5	0−01	
1, 9	−001	
2, 3	001−	
2, 6	0−10	
2, 10	−010	

2 Variables		
Terms	$XYWZ$	P

Now we proceed to compare the terms of group 2 and group 3. See the tables below.

4 Variables		
Terms	$XYWZ$	P
1	0001	∨
2	0010	∨
3	0011	∨
5	0101	∨
6	0110	∨
9	1001	∨
10	1010	∨
12	1100	∨
11	1011	∨
13	1101	∨
14	1110	∨

3 Variables		
Terms	$XYWZ$	P
1, 3	00−1	
1, 5	0−01	
1, 9	−001	
2, 3	001−	
2, 6	0−10	
2, 10	−010	
3, 11	−011	
5, 13	−101	
6, 14	−110	
9, 11	10−1	
9, 13	1−01	
10, 11	101−	
10, 14	1−10	
12, 13	110−	
12, 14	11−0	

2 Variables		
Terms	$XYWZ$	P

A tic in column P regarding all the four-variable implicants (the *minterms*) indicates that none are prime.

The algorithm continues comparing the two groups of three-variable implicants to obtain two-variable implicants.

4 Variables		
Terms	$XYWZ$	P
1	0001	∨
2	0010	∨
3	0011	∨
5	0101	∨
6	0110	∨
9	1001	∨
10	1010	∨
12	1100	∨
11	1011	∨
13	1101	∨
14	1110	∨

3 Variables		
Terms	$XYWZ$	P
1, 3	00−1	∨
1, 5	0−01	∨
1, 9	−001	∨
2, 3	001−	∨
2, 6	0−10	∨
2, 10	−010	∨
3, 11	−011	∨
5, 13	−101	∨
6, 14	−110	∨
9, 11	10−1	∨
9, 13	1−01	∨
10, 11	101−	∨
10, 14	1−10	∨
12, 13	110−	P_0
12, 14	11−0	P_1

2 Variables		
Terms	$XYWZ$	P
1, 3, 9, 11	−0−1	P_2
1, 5, 9, 13	−−01	P_3
2, 3, 10, 11	−01−	P_4
2, 6, 10, 14	−−10	P_5

Note that we can combine both (1, 5) + (9, 13) → (1, 5, 9, 13) and (1, 9) + (5, 13) → (1, 9, 5, 13) but they produce the same term (−−01), $\overline{W}\ Z$, which is reported only once in the two-variable table. The pairs (1, 5), (9, 13) and (1, 9), (5, 13) both receive the tic because neither is a prime implicant.

In the end, there are six prime implicants: P_0 and P_1 with three variables (represented on the Karnaugh map as two cubes with two cells) and P_2, P_3, P_4, P_5 with two variables (represented on the Karnaugh map as four cubes with four cells).

The QM–M proceeds with the covering table to find the lowest number of implicants to cover the function.

	m_1	m_2	m_3	m_6	m_9	m_{10}	m_{11}	m_{12}
P_0								X
P_1								X
P_2	X		X		X		X	
P_3	X				X			
P_4		X	X			X	X	
P_5		X		X		X		

The table clearly indicates that P_5 is an essential prime implicant because it is the only one that covers m_6, so the minimum expression will certainly include it.

$$F(X, Y, W, Z) = P_5 + \cdots = W\ \overline{Z} + \cdots$$

So we can write a new table and eliminate row P_5, which has already been selected as well as columns m_2, m_6, m_{10}, which are covered by P_5:

	m_1	m_3	m_9	m_{11}	m_{12}
P_0					X
P_1					X
P_2	X	X	X	X	
P_3	X		X		
P_4		X		X	

The resulting table can be further simplified by analyzing coverage by the rows and columns and eliminating some of the prime implicants (the rows) or some of the *minterms* (the columns). Column m_i can be eliminated if it covers column m_j; that is, if for every X in column m_j there is an X in the corresponding row of column m_i. In this configuration, we can eliminate because the *minterm* m_i would be covered by one of the implicants that covers m_j, so there is no need to treat it.

In the table above, the columns m_9 and m_{11} can be eliminated because they cover m_1 and m_3, respectively, (in this case, they are actually equal):

	m_1	m_3	m_{12}
P_0			X
P_1			X
P_2	X	X	
P_3	X		
P_4		X	

Likewise, row P_i can be eliminated if it is covered by another implicant P_j; that is, if for every X in row P_i there is an X in the corresponding column of row P_j. We can eliminate P_i because the prime implicant P_j covers all the *minterms* covered by P_i (and possibly more).

In the table above, we can immediately see that P_1 covers P_0 and vice versa while P_2 covers P_3 and P_4. Thus, the final table is:

	m_1	m_3	m_{12}
P_0			X
P_2	X	X	

Now it is very easy to identify the optimal coverage, composed here by P_0 and P_2. By adding these two prime implicants to the already identified (P_5), we get the final result:

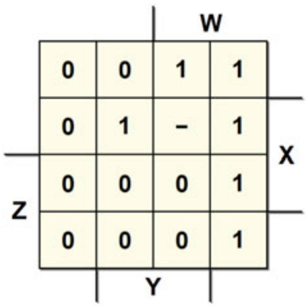
$$F(X, Y, W, Z) = P_0 + P_2 + P_5 = X\ Y\ \overline{W} + \overline{Y}\ Z + W\ \overline{Z}$$

In rare cases, not all the columns or rows can be eliminated in a coverage table. In this case, the table is referred to as "cyclic" and all the possible combinations of prime implicants must be checked to obtain optimal coverage. There are various ways (some optimal, some less so) to deal with this, but they go beyond the scope of this book.

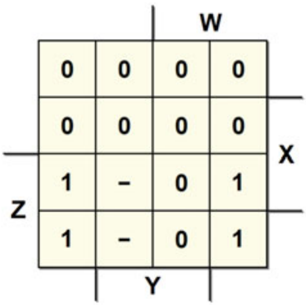
4.1.5 Simultaneous Optimization of Multiple Functions

In real digital systems, we must often create different combinational networks, each corresponding to a Boolean function in the same project. In a case like this, we can benefit in terms of circuit complexity from jointly optimizing functions to reuse some parts of circuits shared by multiple networks. This is why the QM–M was extended to optimize more than one Boolean function. The expansion and covering phases were changed to identify the prime implicants that could be used to cover the *minterms* of multiple functions.

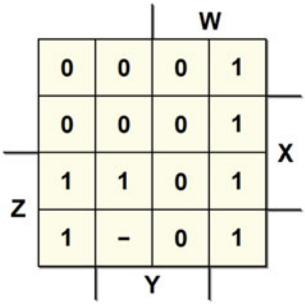
To understand how the multiple function method is extended, see the example below. Supposing we must optimize the following three functions:



$$F_1(X, Y, W, Z) = \Sigma(2, 3, 6, 10, 11, 12) + d(14)$$



$$F_2(X, Y, W, Z) = \Sigma(1, 3, 9, 11) + d(5, 13)$$



$$F_3(X, Y, W, Z) = \Sigma(1, 2, 3, 9, 10, 11) + d(5)$$

In the expansion phase, all the *minterms* of the three functions are reported in the first table. The difference from the case of only one function is that here, there is an added column that uses multi-bit masks to indicate how many functions to optimize and which ones contain the corresponding *minterm*.

4 Variables			
Terms	$XYWZ$	$F_1F_2F_3$	P
1	0001	011	
2	0010	101	
3	0011	111	
5	0101	011	
6	0110	100	
9	1001	011	
10	1010	101	
12	1100	100	
11	1011	111	
13	1101	010	
14	1110	100	

3 Variables			
Terms	$XYWZ$	$F_1F_2F_3$	P

2 Variables			
Terms	$XYWZ$	$F_1F_2F_3$	P

The *minterm* m_1, for example, appears in functions F_2 and F_3 but not in F_1. This is indicated by mask "011". Likewise, the *minterm* m_{14} appears in function F_1 but not in F_2 or F_3, and so the corresponding mask is "100".

In the expansion phase, the four-variable terms are combined to form three-variable terms. We must keep in mind that two terms can be combined only if they appear in the same function. For example, consider the combination of m_1 and m_3, i.e., (0001) and (0011), to obtain (00−1). This is possible only for functions F_2 and F_3, which contain both the terms. Function F_1 contains the term m_3 but not m_1, so it is impossible to simplify in this case. Thus, we will use mask "011" with (1, 3) to indicate that this term is present in only two out of the three functions: F_2 and F_3.

When applying the tic that indicates prime implicants, we must take into account which terms have been combined relative to which functions. The *minterm* m_1 has actually been simplified in all cases, so it is not a prime implicant, and the tic can be inserted in column P. The *minterm* m_3, however, has not been simplified with any other term in the function F_1, so the tic cannot be inserted because m_3 is still a potential prime implicant for this function at least during the expansion phase.

4 Variables			
Terms	$XYWZ$	$F_1F_2F_3$	P
1	0001	011	$\vee$
2	0010	101	
3	0011	111	
5	0101	011	
6	0110	100	
9	1001	011	
10	1010	101	
12	1100	100	
11	1011	111	
13	1101	010	
14	1110	100	

3 Variables			
Terms	$XYWZ$	$F_1F_2F_3$	P
1, 3	00−1	011	

2 Variables			
Terms	$XYWZ$	$F_1F_2F_3$	P

We can derive two simple rules from these considerations about combining terms and inserting the tic for implicants.

Rule 1

Two terms can be combined if we compute the bit-wise AND from the corresponding masks and at least one bit from the resulting mask differs from 0. We actually obtain this when there is a 1 in the same position in both masks; i.e., the terms being considered are both present in the same function. The resulting mask will be reported in the column of the simplified term. The previous case, where m_1 is combined with m_3 to obtain (1, 3): "011" AND "111" → "011".

Rule 2

If a term's mask is identical to the one resulting from the simplification, the corresponding *minterm* is definitely not a prime implicant, so the tic can be inserted. In the previous case, the resulting mask is "011" so the term m_1 is definitely not a prime implicant and we can insert the tic "$\vee$." The masks of m_3 and (1, 3) are not the same (they are "111" and "011", respectively), so m_3 is still a potential prime implicant.

By continuing the simplification, we get the following table with three-variable terms.

4 Variables			
Terms	$XYWZ$	$F_1F_2F_3$	P
1	0001	011	∨
2	0010	101	∨
3	0011	111	∨
5	0101	011	∨
6	0110	100	∨
9	1001	011	∨
10	1010	101	∨
12	1100	100	∨
11	1011	111	∨
13	1101	010	∨
14	1110	100	∨

3 Variabili			
Terms	$XYWZ$	$F_1F_2F_3$	P
1, 3	00−1	011	
1, 5	0−01	011	
1, 9	−001	011	
2, 3	001−	101	
2, 6	0−10	100	
2, 10	−010	101	
3, 11	−011	111	
5, 13	−101	010	
6, 14	−110	100	
9, 11	10−1	011	
9, 13	1−01	010	
10, 11	101−	101	
10, 14	1−10	100	
12, 14	11−0	100	

2 Variables			
Terms	$XYWZ$	$F_1F_2F_3$	P

Note that the *minterms* m_{12} and m_{13} were not combined because m_{12} is only present in function F_1 ("100") and m_{13}, only in function F_2 ("010").

Now we can combine the three-variable terms to obtain two-variable terms if possible.

4 Variables			
Terms	$XYWZ$	$F_1F_2F_3$	P
1	0001	011	∨
2	0010	101	∨
3	0011	111	∨
5	0101	011	∨
6	0110	100	∨
9	1001	011	∨
10	1010	101	∨
12	1100	100	∨
11	1011	111	∨
13	1101	010	∨
14	1110	100	∨

3 Variables			
Terms	$XYWZ$	$F_1F_2F_3$	P
1, 3	00−1	011	∨
1, 5	0−01	011	P_0
1, 9	−001	011	∨
2, 3	001−	101	∨
2, 6	0−10	100	∨
2, 10	−010	101	∨
3, 11	−011	111	P_1
5, 13	−101	010	∨
6, 14	−110	100	∨
9, 11	10−1	011	∨
9, 13	1−01	010	∨
10, 11	101−	101	∨
10, 14	1−10	100	∨
12, 14	11−0	100	P_2

2 Variables			
Terms	$XYWZ$	$F_1F_2F_3$	P
1, 3, 9, 11	−0−1	011	P_3
1, 5, 9, 13	−−01	010	P_4
2, 3, 10, 11	−01−	101	P_5
2, 6, 10, 14	−−10	100	P_6

Here as well, we must pay attention to the masks that indicate which functions we can continue simplifying and which are the prime implicants.

For example, when we combine (1, 3) with (9, 11) we get (1, 3, 9, 11) for functions F_2 and F_3 ("011"), and both starting terms are non-prime implicants.

However, when we combine (1, 9) with (3, 11) we get the same term (1, 3, 9, 11) for the same functions (F_2 and F_3) but (3, 11) is still a prime implicant because it was impossible to simplify it with another term in function F_1. The two masks are actually "011" and "111".

When the expansion phase is finished, we can move on to the covering phase, considering each function separately.

	F_1						F_2				F_3					
	m_2	m_3	m_6	m_{10}	m_{11}	m_{12}	m_1	m_3	m_9	m_{11}	m_1	m_2	m_3	m_9	m_{10}	m_{11}
P_0							X				X					
P_1		X			X			X		X			X			X
P_2						X										
P_3							X	X	X	X	X		X	X		X
P_4							X		X							
P_5	X	X		X	X							X	X		X	X
P_6	X		X	X												

Masks are useful here as well because they indicate what functions the prime implicant should be associated with. The prime implicant P_4, for example, covers the *minterms* m_1, m_5, m_9, and m_{13} only for function F_2, because the corresponding mask is "010", so Xs are not inserted for the other functions (F_1 and F_3).

As before, we go on to identify the essential prime implicants, in this case: P_2 (because it is the only one that covers the *minterm* m_{12} in F_1), P_3 (because it is the only one that covers the *minterm* m_9 of F_3), P_5 (because it is the only one that covers the *minterms* m_2 and m_{10} in F_3) and P_6 (because it is the only one that covers the *minterm* m_6 in F_1). Note that when a prime implicant is selected, it is selected for all the functions. The implicant P_5, for example, is selected because it is essential for F_3 but at that point it is also used to cover the *minterms* of F_1. In other words P_5, which corresponds to the term (2, 3, 10, 11) i.e., $\overline{Y}W$, certainly appears in the optimal expression of F_3 but can also be used for F_1. This way, the same logical network is used twice economizing on the circuit level.

By selecting P_2, P_3, P_5 and P_6 we cover all the *minterms* in all the functions and we now can write the resulting functions:

$$F_1(X, Y, W, Z) = P_2 + P_5 + P_6 = XY\overline{Z} + \overline{Y}W + W\overline{Z}$$
$$F_2(X, Y, W, Z) = P_3 = \overline{Y}Z$$
$$F_3(X, Y, W, Z) = P_3 + P_5 = \overline{Y}Z + \overline{Y}W$$

Note that the P_3 and P_5 combinational networks are used twice in different functions providing some savings in the overall complexity of the circuit.

4.2 Exercises

4.2.1 Quine–McCluskey: Single Function Synthesis

1. Synthesize the following Boolean function with the QM–M

$$F(A, B, C, D) = \Sigma(5, 7, 8, 9, 12, 13, 15) + d(4)$$

4 Variables		
Terms	$ABCD$	P

3 Variables		
Terms	$ABCD$	P

2 Variables		
Terms	$ABCD$	P

	m_5	m_7	m_8	m_9	m_{12}	m_{13}	m_{15}
P_0							
P_1							
P_2							
P_3							
P_4							
P_5							
P_6							
P_7							
P_8							
P_9							
P_{10}							
P_{11}							
P_{12}							
P_{13}							
P_{14}							
P_{15}							

2. Synthesize the following Boolean functions with the QM–M

$$F(A, B, C, D) = \Sigma(0, 1, 2, 3, 4, 5, 15) + d(10, 14)$$

4 Variables		
Terms	$ABCD$	P

3 Variables		
Terms	$ABCD$	P

2 Variables		
Terms	$ABCD$	P

	m_0	m_1	m_2	m_3	m_4	m_5	m_{15}
P_0							
P_1							
P_2							
P_3							
P_4							
P_5							
P_6							
P_7							
P_8							
P_9							
P_{10}							
P_{11}							
P_{12}							
P_{13}							
P_{14}							
P_{15}							

3. Synthesize the following Boolean function with the QM–M

$$F(A, B, C, D) = \Sigma(1, 9, 11, 13, 15) + d(0, 4, 5)$$

4 Variables		
Terms	$ABCD$	P

3 Variables		
Terms	$ABCD$	P

2 Variables		
Terms	$ABCD$	P

	m_1	m_9	m_{11}	m_{13}	m_{15}
P_0					
P_1					
P_2					
P_3					
P_4					
P_5					
P_6					
P_7					
P_8					
P_9					
P_{10}					
P_{11}					
P_{12}					
P_{13}					
P_{14}					
P_{15}					

4.2.2 *Quine–McCluskey: Jointly Synthesis of Multiple Functions*

1. Jointly synthesize the following Boolean functions with the QM–M

$$F_1(A, B, C, D) = \Sigma(5, 7, 13, 15) + d(4, 12)$$
$$F_2(A, B, C, D) = \Sigma(8, 9, 12, 13) + d(4, 5)$$
$$F_3(A, B, C, D) = \Sigma(4, 5, 12, 13)$$

4 Variables			
Terms	$ABCD$	$F_1F_2F_3$	P

3 Variables			
Terms	$ABCD$	$F_1F_2F_3$	P

2 Variables			
Terms	$ABCD$	$F_1F_2F_3$	P

	F_1				F_2				F_3			
	m_5	m_7	m_{13}	m_{15}	m_8	m_9	m_{12}	m_{13}	m_4	m_5	m_{12}	m_{13}
P_0												
P_1												
P_2												
P_3												
P_4												
P_5												
P_6												
P_7												
P_8												
P_9												
P_{10}												
P_{11}												
P_{12}												
P_{13}												
P_{14}												
P_{15}												

2. Jointly synthesize the following Boolean functions with the QM–M

$$F_1(A, B, C, D) = \Sigma(0, 1, 2, 15) + d(3)$$
$$F_2(A, B, C, D) = \Sigma(2, 3, 4, 5) + d(0, 1)$$
$$F_3(A, B, C, D) = \Sigma(0, 1, 4, 5)$$

4 Variables			
Terms	$ABCD$	$F_1 F_2 F_3$	P

3 Variables			
Terms	$ABCD$	$F_1 F_2 F_3$	P

2 Variables			
Terms	$ABCD$	$F_1 F_2 F_3$	P

	F_1				F_2				F_3			
	m_0	m_1	m_2	m_{15}	m_2	m_3	m_4	m_5	m_0	m_1	m_4	m_5
P_0												
P_1												
P_2												
P_3												
P_4												
P_5												
P_6												
P_7												
P_8												
P_9												
P_{10}												
P_{11}												
P_{12}												
P_{13}												
P_{14}												
P_{15}												

3. Jointly synthesize the following Boolean functions with the QM–M

$$F_1(A, B, C, D) = \Sigma(1, 4, 5) + d(13)$$
$$F_2(A, B, C, D) = \Sigma(1, 9, 13) + d(0, 5)$$
$$F_3(A, B, C, D) = \Sigma(9, 11, 13, 15) + d(1, 5)$$

4 Variables			
Terms	$ABCD$	$F_1F_2F_3$	P

3 Variables			
Terms	$ABCD$	$F_1F_2F_3$	P

2 Variables			
Terms	$ABCD$	$F_1F_2F_3$	P

	F_1			F_2			F_3			
	m_1	m_4	m_5	m_1	m_9	m_{13}	m_9	m_{11}	m_{13}	m_{15}
P_0										
P_1										
P_2										
P_3										
P_4										
P_5										
P_6										
P_7										
P_8										
P_9										
P_{10}										
P_{11}										
P_{12}										
P_{13}										
P_{14}										
P_{15}										

4.3 Solutions

4.3.1 Quine–McCluskey: Synthesis of a Single Function

1. $F(A, B, C, D) = B\ D + A\ \overline{C}$

4 Variables		
Terms	$ABCD$	P
4	0100	$\vee$
8	1000	$\vee$
5	0101	$\vee$
9	1001	$\vee$
12	1100	$\vee$
7	0111	$\vee$
13	1101	$\vee$
15	1111	$\vee$

3 Variables		
Terms	$ABCD$	P
5, 4	010–	$\vee$
9, 8	100–	$\vee$
12, 4	–100	$\vee$
12, 8	1–00	$\vee$
7, 5	01–1	$\vee$
13, 5	–101	$\vee$
13, 9	1–01	$\vee$
13, 12	110–	$\vee$
15, 7	–111	$\vee$
15, 13	11–1	$\vee$

2 Variables		
Terms	$ABCD$	P
13, 12, 9, 8	1–0–	P_0
3, 12, 5, 4	–10–	P_1
15, 13, 7, 5	–1–1	P_2

	m_5	m_7	m_8	m_9	m_{12}	m_{13}	m_{15}
P_0			X	X	X	X	
P_1	X				X	X	
P_2	X	X				X	X

2. $F(A, B, C, D) = A\ B\ C + \overline{A}\ \overline{C} + \overline{A}\ \overline{B}$

4 Variables		
Terms	$ABCD$	P
0	0000	$\vee$
1	0001	$\vee$
2	0010	$\vee$
4	0100	$\vee$
3	0011	$\vee$
5	0101	$\vee$
10	1010	$\vee$
14	1110	$\vee$
15	1111	$\vee$

3 Variables		
Terms	$ABCD$	P
1, 0	000–	$\vee$
2, 0	00–0	$\vee$
4, 0	0–00	$\vee$
3, 1	00–1	$\vee$
3, 2	001–	$\vee$
5, 1	0–01	$\vee$
5, 4	010–	$\vee$
10, 2	–010	P_0
14, 10	1–10	P_1
15, 14	111–	P_2

2 Variables		
Terms	$ABCD$	P
3, 2, 1, 0	00––	P_3
5, 4, 1, 0	0–0–	P_4

	m_0	m_1	m_2	m_3	m_4	m_5	m_{15}
P_0			X				
P_1							
P_2							X
P_3	X	X	X	X			
P_4	X	X			X	X	

3. $F(A, B, C, D) = A\ D + \overline{C}\ D$

4 Variables		
Terms	$ABCD$	P
0	0000	∨
1	0001	∨
4	0100	∨
9	1001	∨
5	0101	∨
11	1011	∨
13	1101	∨
15	1111	∨

3 Variables		
Terms	$ABCD$	P
1, 0	000−	∨
4, 0	0−00	∨
5, 1	0−01	∨
5, 4	010−	∨
9, 1	−001	∨
11, 9	10−1	∨
13, 5	−101	∨
13, 9	1−01	∨
15, 11	1−11	∨
15, 13	11−1	∨

2 Variables		
Terms	$ABCD$	P
5, 4, 1, 0	0−0−	P_0
13, 5, 9, 1	−−01	P_1
15, 13, 11, 9	1−−1	P_2

	m_1	m_9	m_{11}	m_{13}	m_{15}
P_0	X				
P_1	X	X		X	
P_2		X	X	X	X

4.3.2 *Quine–McCluskey: Joint Synthesis of Multiple Functions*

1. The solution is

$$F_1(A, B, C, D) = B\ \overline{C}$$

$$F_2(A, B, C, D) = A\ \overline{C}$$

$$F_3(A, B, C, D) = B\ D$$

4 Variables			
Terms	$ABCD$	$F_1F_2F_3$	P
4	0100	111	∨
8	1000	010	∨
5	0101	111	∨
9	1001	010	∨
12	1100	111	∨
7	0111	100	∨
13	1101	111	∨
15	1111	100	∨

3 Variables			
Terms	$ABCD$	$F_1F_2F_3$	P
5, 4	010−	111	∨
4, 12	−100	111	∨
8, 9	100−	010	∨
8, 12	1−00	010	∨
5, 7	01−1	100	∨
5, 13	−101	111	∨
9, 13	1−01	010	∨
12, 13	110−	111	∨
7, 15	−111	100	∨
13, 15	11−1	100	∨

2 Variables			
Terms	$ABCD$	$F_1F_2F_3$	P
4, 5, 12, 13	−10−	111	P_0
8, 9, 12, 13	1−0−	010	P_1
5, 7, 13, 15	−1−1	100	P_2

	F_1				F_2				F_3			
	m_5	m_7	m_{13}	m_{15}	m_8	m_9	m_{12}	m_{13}	m_4	m_5	m_{12}	m_{13}
P_0	X		X				X	X	X	X	X	X
P_1					X	X	X	X				
P_2	X	X	X	X								

2. The solution is

$$F_1(A,B,C,D) = A\ B\ C\ D\ + \overline{A}\ \overline{B}$$
$$F_2(A,B,C,D) = \overline{A}\ \overline{B} + \overline{A}\ \overline{C}$$
$$F_3(A,B,C,D) = \overline{A}\ \overline{C}$$

4 Variables			
Terms	$ABCD$	$F_1F_2F_3$	P
0	0000	111	∨
1	0001	111	∨
2	0010	110	∨
4	0100	011	∨
3	0011	110	∨
5	0101	011	∨
15	1111	100	P_0

3 Variables			
Terms	$ABCD$	$F_1F_2F_3$	P
0, 1	000−	111	P_1
0, 2	00−0	110	∨
0, 4	0−00	011	∨
1, 3	00−1	110	∨
1, 5	0−01	011	∨
2, 3	001−	110	∨
4, 5	010−	011	∨

2 Variables			
Terms	$ABCD$	$F_1F_2F_3$	P
0, 1, 2, 3	00−−	110	P_2
0, 1, 4, 5	0−0−	011	P_3

	F_1				F_2				F_3			
	m_0	m_1	m_2	m_{15}	m_2	m_3	m_4	m_5	m_0	m_1	m_4	m_5
P_0				X								
P_1	X	X							X	X		
P_2	X	X	X		X	X						
P_3							X	X	X	X	X	X

3. The solution is

$$F_1(A,B,C,D) = \overline{A}\ \overline{C}\ D + \overline{A}\ B\ \overline{C}$$
$$F_2(A,B,C,D) = \overline{C}\ D$$
$$F_3(A,B,C,D) = A\ D$$

4 Variables			
Terms	$ABCD$	$F_1F_2F_3$	P
0	0000	010	∨
1	0001	111	∨
4	0100	100	∨
5	0101	111	∨
9	1001	011	∨
11	1011	001	∨
13	1101	111	∨
15	1111	001	∨

3 Variables			
Terms	$ABCD$	$F_1F_2F_3$	P
0, 1	000−	010	P_0
1, 5	0−01	111	P_1
1, 9	−001	011	∨
4, 5	010−	100	P_2
5, 13	−101	111	P_3
9, 11	10−1	001	∨
9, 13	1−01	011	∨
11, 15	1−11	001	∨
13, 15	11−1	001	∨

2 Variables			
Terms	$ABCD$	$F_1F_2F_3$	P
1, 5, 9, 13	−−01	011	P_4
9, 11, 13, 15	1−−1	001	P_5

	F_1			F_2			F_3			
	m_1	m_4	m_5	m_1	m_9	m_{13}	m_9	m_{11}	m_{13}	m_{15}
P_0				X						
P_1	X		X	X						
P_2		X	X							
P_3			X			X			X	
P_4				X	X	X	X		X	
P_5							X	X	X	X