Unit 11 Canonical forms.

- In a Boolean expression a **literal** is an input variable or its complement.
- A Boolean function is in canonical Sum of Products form when each product term contains each of the literals.
- The rows of the truth table for which $Q = 1$ give the canonical terms in the Sum of Products expression.
- A Boolean function is in canonical Product of Sums form when each Sum term contains each of the literals.
- The rows of the truth table for which $Q = 0$ give the canonical terms in the Product of Sums expression.

When a real world digital problem is specified, the most convenient form of the specification is a truth table in which all of the possible combinations of the n input variables are enumerated and the appropriate output action specified. For instance, if A represents a green traffic light, B represents a traffic free junction and C signifies that the car engine is running, then a truth table can be formulated which specifies the action, Q, to be taken, where $Q = 1$ means to drive forward.

This truth table would have only one row for which $Q = 1$ but there are many other truth tables for which $Q = 1$ for a number of different sets of conditions. The number of rows in the truth table is given by 2^n where n is the number of inputs to the system. In the following examples, due to lack of space for large truth tables, we will only consider truth tables having less than five input variables but usually, in any real world situation, there will be more than four input variables.

Consider, as an example, the following truth table for an arbitrary system.

A	B	C	Q	
0	0	0	0	
0	0	1	1	
0	1	0	0	
0	1	1	1	$\Leftarrow$
1	0	0	0	
1	0	1	0	
1	1	0	1	
1	1	1	1	

Consider the arrowed row, $\Leftarrow$, for which the input conditions are $A = 0$, $B = 1$ and $C = 1$. If $A = 0$ then $\overline{A} = 1$ and we obtain the Boolean identity for this row, $\overline{A}.B.C = 1$. Similarly, there are other rows for which $Q = 1$ in accordance with the requirements of the real world problem specification. We can therefore form the Boolean expression for this truth table as:

$$Q = \overline{A}.\overline{B}.C + \overline{A}.B.C + A.B.\overline{C} + A.B.C$$

This expression can be simplified, using Boolean algebra, as follows:

$$\begin{aligned} Q &= \overline{A}.\overline{B}.C + \overline{A}.B.C + A.B.\overline{C} + A.B.C \\ &= \overline{A}.C.(\overline{B} + B) + A.B.(\overline{C} + C) \\ &= \overline{A}.C.1 + A.B.1 \\ &= \overline{A}.C + A.B \end{aligned}$$

This means that the problem can be implemented in logic gate circuits in either of the two forms shown in Figure 11.1. It is evident that the second form will require fewer logic gates and also fewer interconnections and will therefore, be the preferred solution.

In the circuit we have also introduced a bus line and its complement, A and $\overline{A}$. This does not give any great simplification for this three variable system but, for larger numbers of input variables, the reduction of the number of gates required (by avoiding repeated inverter gates) is significant and it also gives a more systematic circuit diagram.

The truth table then converts to a digital logic circuit which is in the form of a two level AND-OR circuit and, when expressed in Boolean algebraic form, is a Sum of Products. In an effort to standardize the expression of the truth table we use what is called the **Canonical Form** of the Boolean expression. In the canonical form of a Sum of Products Boolean expression, each of the literals, that is each of the input variables or its complement,

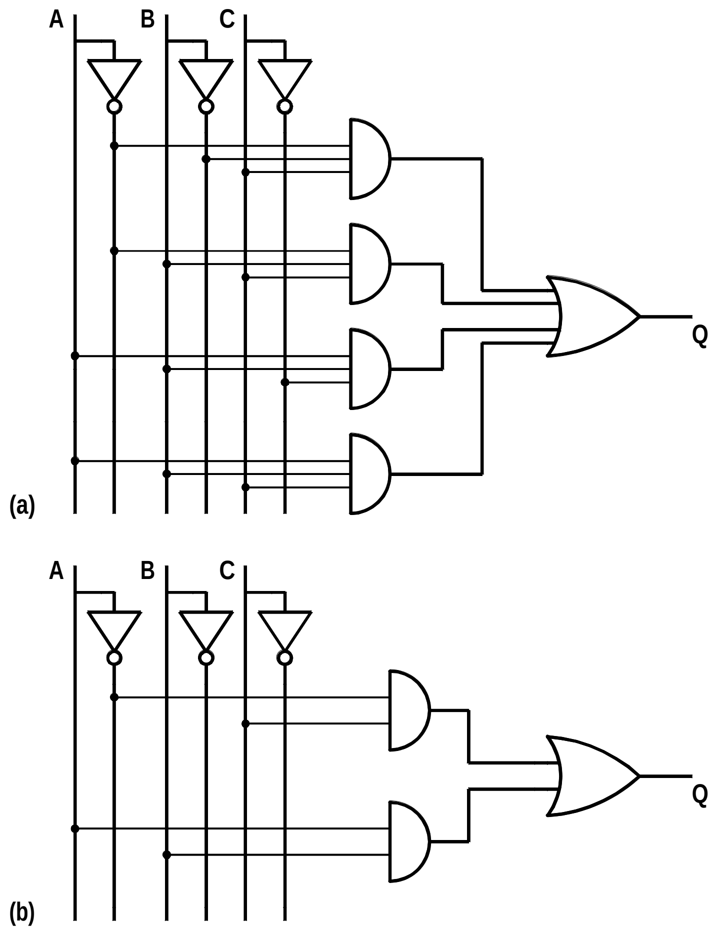

Figure 11.1: Sum of Products circuit (a) implementing the canonical form and (b) implementing the simplified expression.

appears once in each of the product terms of the expression and also the expression contains a product term for each logical 1 in the truth table output. These product terms are then summed to give a Sum of Products. The inputs may appear in either uncomplemented or complemented form, that is as either A or $\overline{A}$ in the canonical form. For the truth table shown above, the canonical form is:

$$Q = \overline{A}.\overline{B}.C + \overline{A}.B.C + A.B.\overline{C} + A.B.C$$

The Boolean expression can also be in a canonical form when each of the literals appears in each of the Sum terms of a Product of Sums Boolean expression. The following examples show how these two canonical forms are obtained from the truth tables.

11.1 Examples

11.1 Obtain the OR-AND (Product of Sums) form of the Boolean expression which represents the truth table shown below and draw the logic gate circuit.

A	B	C	Q	$\overline{Q}$
0	0	0	1	0
0	0	1	1	0
0	1	0	1	0
0	1	1	1	0
1	0	0	1	0
1	0	1	1	0
1	1	0	0	1
1	1	1	1	0

An extra column has been inserted in the truth table which represents $\overline{Q}$ and it can be seen that there is only one 1 in the $\overline{Q}$ column which occurs for $A = 1$, $B = 1$, $C = 0$ so that the truth table is represented by:

$$\overline{Q} = AB\overline{C}$$

By use of DeMorgan's theorem we obtain:

$$Q\overline{\overline{Q}} = \overline{AB\overline{C}} = \overline{A} + \overline{B} + \overline{\overline{C}} = \overline{A} + \overline{B} + C$$

This is implemented in the circuit shown below. Note that the single input AND gate has no effect on the signal.

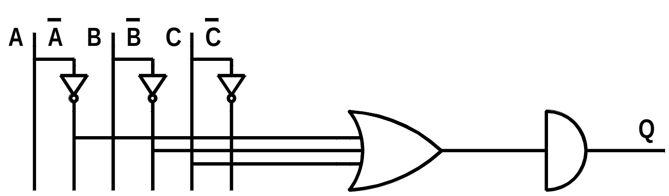

Figure 11.2: $Q = \overline{A} + \overline{B} + C$

11.2 Obtain the OR-AND Product-of-Sums form of the Boolean expression which represents the truth table shown below.

A	B	C	Q	$\overline{Q}$
0	0	0	1	0
0	0	1	1	0
0	1	0	0	1
0	1	1	1	0
1	0	0	1	0
1	0	1	1	0
1	1	0	0	1
1	1	1	1	0

An extra column has been inserted in the truth table which represents $\overline{Q}$ and there are only two 1s in the $\overline{Q}$ column which are specified by

$$\overline{Q} = \overline{A}B\overline{C} + AB\overline{C}$$

Successive application of DeMorgan's theorem gives:

$$\begin{aligned} Q = \overline{\overline{Q}} &= \overline{\overline{A}B\overline{C} + AB\overline{C}} \\ &= (\overline{\overline{A}B\overline{C}})(\overline{AB\overline{C}}) \\ &= (A + \overline{B} + C)(\overline{A} + \overline{B} + C) \end{aligned}$$

Which is implemented in the circuit shown below

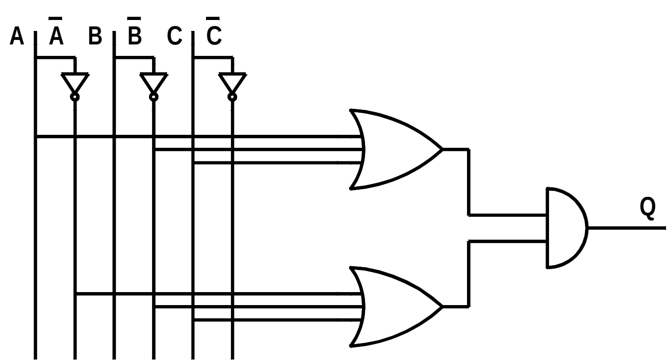

Figure 11.3: $Q = (A + \overline{B} + C)(\overline{A} + \overline{B} + C)$

11.3 Obtain the canonical Sum-of-Products (AND-OR) expression which represents this truth table.

A	B	C	Q
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

The generalized representation of a Sum-of-Products Boolean expression is:

$$\begin{aligned} Q &= \overline{A}.\overline{B}.\overline{C} + \overline{A}.\overline{B}.C + \overline{A}.B.\overline{C} + \overline{A}.B.C \\ &\quad + A.\overline{B}.\overline{C} + A.\overline{B}.C + A.B.\overline{C} + A.B.C \end{aligned}$$

When the product of inputs (uncomplemented or complemented) in a row of the truth table computes to a 1 the product term for that row is a 1 and the Sum-of-Products computes to a 1. But according to the information in the truth table some of these Product terms are known to compute to 0 and can therefore be eliminated from the Boolean Sum-of-Products expression so that the expression now reduces to:

$$\begin{aligned} Q &= 0 + \overline{AB}C + 0 + 0 \\ &\quad + A\overline{BC} + 0 + 0 + ABC \\ &= \overline{AB}C + A\overline{BC} + ABC \end{aligned}$$

11.4 Obtain the canonical Product-of-Sums (OR-AND) expression which represents this truth table.

A	B	C	Q
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

The generalized representation of a Product-of-Sums Boolean expression is:

$$\begin{aligned} Q &= (\overline{A}+\overline{B}+\overline{C})(\overline{A}+\overline{B}+C)(\overline{A}+B+\overline{C})(\overline{A}+B+C) \\ &\quad (A+\overline{B}+\overline{C})(A+\overline{B}+C)(A+B+\overline{C})(A+B+C) \end{aligned}$$

When the Sums of inputs (uncomplemented or complemented) in a row of the truth table computes to a 0 the sum term for that row is a 0 and the Product-of-Sums computes to a 0. But, according to the information in the truth table, some of these terms are known to compute to 1 and can therefore be eliminated from the Product-of-Sums expression so that the generalized expression shown above now reduces to:

$$\begin{aligned} Q &= (\overline{A}+\overline{B}+\overline{C})(\overline{A}+\overline{B}+C)(\overline{A}+B+\overline{C})(\overline{A}+B+C) \\ &\quad (A+\overline{B}+\overline{C})(A+\overline{B}+C)(A+B+\overline{C})(A+B+C) \\ &= (1)(1)(1)(\overline{A}+B+C) \\ &\quad (1)(A+\overline{B}+C)(A+B+\overline{C})(1) \\ &= (A+\overline{B}+\overline{C})(\overline{A}+B+\overline{C})(\overline{A}+\overline{B}+C) \end{aligned}$$

An alternative derivation starts from the truth table

$$\overline{Q} = \overline{A}BC + A\overline{B}C + AB\overline{C}$$

Then by repeated application of DeMorgan's theorem we get

$$\begin{aligned} Q &= \overline{\overline{\overline{A}BC + A\overline{B}C + AB\overline{C}}} \\ &= \overline{(\overline{A}BC)}\,\overline{(A\overline{B}C)}\,\overline{(AB\overline{C})} \\ &= (A+\overline{B}+\overline{C})(\overline{A}+B+\overline{C})(\overline{A}+\overline{B}+C) \end{aligned}$$

11.2 Problems

11.1 Calculate and write in the Boolean value of Q_x (0 or 1) for each of these terms:

(a) When $A = 1$, $B = 0$ and $C = 1$ then $Q_1 = A\overline{BC} =$

(b) When $A = 1$, $B = 0$ and $C = 1$ then $Q_1 = A\overline{B}C =$

(c) When $A = 1$, $B = 1$ and $C = 0$ then $Q_2 = AB\overline{C} =$

(d) When $A = 1$, $B = 0$ and $C = 1$ then $Q_3 = \overline{A} + \overline{B} + C =$

(e) When $A = 1$, $B = 0$ and $C = 1$ then $Q_4 = \overline{A} + B + \overline{C} =$

(f) When $A = 0$, $B = 1$ and $C = 0$ then $Q_5 = A + \overline{B} + C =$

(g) When $A = 1$, $B = 1$ and $C = 1$ then $Q_6 = \overline{A} + \overline{B} + \overline{C} =$

11.2 Expand the expression below to the canonical Sum-of-Products form (Hint Use P3 $X.1 = X$ and P4 $X + \overline{X} = 1$).

$$Q = A\overline{C} + \overline{B}C$$

11.3 Expand the expression below to the canonical Product-of-Sums form (Hint Use P22 $X + Y.Z = (X + Y)(X + Z)$, P3 $X + 0 = X$ and P4 $\overline{X}X = 0$).

$$Q = (A + \overline{B})(A + C)$$

11.4 Write down the canonical, Boolean Sum of Products expression which represents this truth table.

A	B	C	Q
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

11.5 Write down the canonical Boolean Product of Sums expression which represents this truth table.

A	B	C	Q
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

11.6 Draw the AND-OR circuit which implements this truth table.

A	B	C	Q
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

11.7 Draw the OR-AND circuit which implements this truth table.

A	B	C	Q
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

11.8 Use De Morgan's theorem to show that a Sum-of-Products, AND–OR circuit, can be converted to a circuit which uses only NAND gates. (Use the identity $\overline{\overline{A}} = A$.)