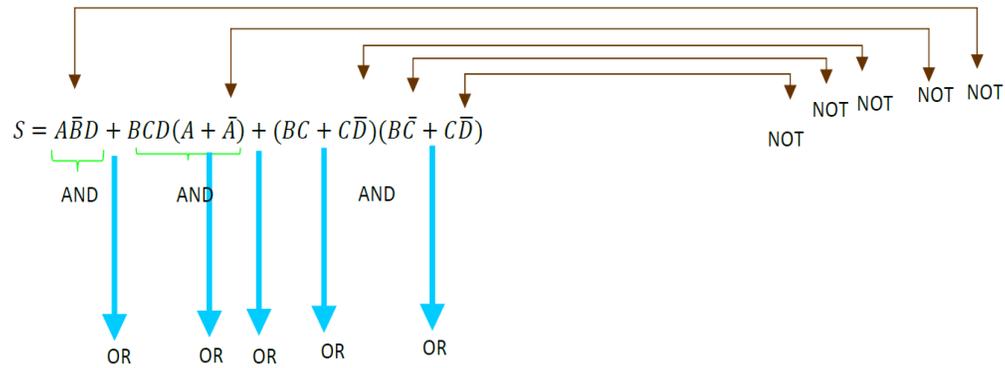


Resolución de funciones lógicas por medio de álgebra de Boole

Unidad de Apoyo para el Aprendizaje

Segunda forma de solución del ejemplo 3.



En este ejemplo se tienen 3 compuertas AND, 5 compuertas OR y 5 compuerta NOT, se utilizarán los teoremas del algebra de Boole para simplificar esa función y posteriormente poder armar el circuito.

1. Realización de las primeras operaciones para quitar los paréntesis.

$$S = A\bar{B}D + BCDA + BCD\bar{A} + BCB\bar{C} + BCC\bar{D} + C\bar{D}B\bar{C} + C\bar{D}C\bar{D}$$

2. Aplicando el teorema, resulta:

$$S = A\bar{B}D + BCDA + BCD\bar{A} + 0 + B\bar{C}\bar{D} + 0 + C\bar{D}$$

3. Factorizando elementos

$$S = A\bar{B}D + BCD(A + \bar{A}) + C\bar{D}(B + 1)$$

Resolución de funciones lógicas por medio de álgebra de Boole

Unidad de Apoyo para el Aprendizaje

$$S = A\bar{B}D + BCD + C\bar{D}$$

4. Última factorización posible

$$S = A\bar{B}D + C(BD + \bar{D})$$

Diagram illustrating the factorization of the Boolean expression $S = A\bar{B}D + C(BD + \bar{D})$. The expression is shown with green brackets under $A\bar{B}D$ and $C(BD + \bar{D})$, both labeled "AND". Blue arrows point from these brackets to "OR" labels below. A brown line with arrows at the end branches from the top of the expression to "NOT" labels under the B and D terms in the second AND term.

Con esto podemos observar que la cantidad de compuertas lógicas se ha reducido de 3 a 2 AND, de 5 a 2 OR y de 5 a 2 NOT.

La simplificación ahora es un poco más corta pero el resultado es el mismo que el anterior.