Ejercicio 4

Se desea realizar un circuito que convierta un número binario de 4 bits en su complemento a dos.

- Exprese el valor de cada bit de salida en función de los mintérminos de los bits de entrada.
- Exprese el valor de cada bit de salida en forma simplificada.
- Dibuje el circuito lógico resultante utilizando compuertas AND, OR, NOT.
- Implemente el circuito resultante en Verilog.

Para obtener el complemento a dos de un numero binario se invierte el valor de cada una de sus cifras, es decir, se realiza el complemento a uno, y luego se le suma uno al número resultante de la inversión.

Su utilidad principal se encuentra en las operaciones matemáticas con números binarios. En particular, la resta de números binarios se facilita enormemente utilizando el complemento a dos: la resta de dos números binarios puede obtenerse sumando al minuendo el complemento a dos del sustraendo.

Además, llamaremos mintérmino m_i a aquel que se forma multiplicando (AND lógico) todas las variables, negando aquellas que valen 0 en la combinación para la cual queremos que el mintérmino valga 1. Para N variables booleanas, existen 2^N mintérmino, uno para cada posible combinación de ellas.

A continuación se realiza la tabla de verdad para un numero binario de 4 bits. Donde a cada número binario $(x_1x_2x_3x_4)$ le corresponde su respectivo complemento a dos a la salida $(f_1f_2f_3f_4)$.

x_1	x_2	x_3	x_4	f_1	f_2	f_3	f_4	m_i
0	0	0	0	0	0	0	0	m_0
0	0	0	1	1	1	1	1	m_1
0	0	1	0	1	1	1	0	m_2
0	0	1	1	1	1	0	1	m_3
0	1	0	0	1	1	0	0	m_4
0	1	0	1	1	0	1	1	m_5
0	1	1	0	1	0	1	0	m_6
0	1	1	1	1	0	0	1	m_7
1	0	0	0	1	0	0	0	m_8
1	0	0	1	0	1	1	1	m_9
1	0	1	0	0	1	1	0	m_{10}
1	0	1	1	0	1	0	1	m_{11}
1	1	0	0	0	1	0	0	m_{12}
1	1	0	1	0	0	1	1	m_{13}
1	1	1	0	0	0	1	0	m_{14}
1	1	1	1	0	0	0	1	m_{15}

Table 1: Complemento a dos $(f_1f_2f_3f_4)$ del bit de entrada $(x_1x_2x_3x_4)$

Se expresa la salida como función de los mintérminos. Para expresar la función en mintérminos tomamos donde la función sea 1 y unimos los mintérminos con sumas:

$$f_1(m_i) = m_1 + m_2 + m_3 + m_4 + m_5 + m_6 + m_7 + m_8$$

$$f_2(m_i) = m_1 + m_2 + m_3 + m_4 + m_9 + m_{10} + m_{11} + m_{12}$$

$$f_3(m_i) = m_1 + m_2 + m_5 + m_6 + m_9 + m_{10} + m_{13} + m_{14}$$

$$f_4(m_i) = m_1 + m_3 + m_5 + m_7 + m_9 + m_{11} + m_{13} + m_{15}$$

Se reemplaza por los valores de entrada,

$$
\begin{aligned}
f_1(x_1,x_2,x_3,x_4) &= \overline{x_1}\,\overline{x_2}\,\overline{x_3}x_4+\overline{x_1}\,\overline{x_2}x_3\overline{x_4}+\overline{x_1}\,\overline{x_2}x_3x_4+\overline{x_1}x_2\overline{x_3}\,\overline{x_4}+\overline{x_1}x_2\overline{x_3}x_4+\overline{x_1}x_2x_3\overline{x_4}+\overline{x_1}x_2x_3x_4+x_1\overline{x_2}\,\overline{x_3}\,\overline{x_4}\\
f_2(x_1,x_2,x_3,x_4) &= \overline{x_1}\,\overline{x_2}\,\overline{x_3}x_4+\overline{x_1}\,\overline{x_2}x_3\overline{x_4}+\overline{x_1}\,\overline{x_2}x_3x_4+\overline{x_1}x_2\overline{x_3}\,\overline{x_4}+x_1\overline{x_2}\,\overline{x_3}x_4+x_1\overline{x_2}x_3\overline{x_4}+x_1\overline{x_2}x_3x_4+x_1x_2\overline{x_3}\,\overline{x_4}\\
f_3(x_1,x_2,x_3,x_4) &= \overline{x_1}\,\overline{x_2}\,\overline{x_3}x_4+\overline{x_1}\,\overline{x_2}x_3\overline{x_4}+\overline{x_1}x_2\overline{x_3}x_4+\overline{x_1}x_2x_3\overline{x_4}+x_1\overline{x_2}\,\overline{x_3}x_4+x_1\overline{x_2}x_3\overline{x_4}+x_1x_2\overline{x_3}x_4+x_1x_2x_3\overline{x_4}\\
f_4(x_1,x_2,x_3,x_4) &= \overline{x_1}\,\overline{x_2}\,\overline{x_3}x_4+\overline{x_1}\,\overline{x_2}x_3x_4+\overline{x_1}x_2\overline{x_3}x_4+\overline{x_1}x_2x_3x_4+x_1\overline{x_2}\,\overline{x_3}x_4+x_1\overline{x_2}x_3x_4+x_1x_2\overline{x_3}x_4+x_1x_2x_3x_4
\end{aligned}
$$

Finalmente, utilizando mapas de Karnaugh, se simplifican las ecuaciones. Los mapas de Karnaugh reducen la necesidad de hacer cálculos extensos para la simplificación de expresiones booleanas.

El mapa de Karnaugh consiste en una representación bidimensional de la tabla de verdad de la función a simplificar. Puesto que la tabla de verdad de una función de N variables posee 2^N filas, el mapa K correspondiente debe poseer también 2^N cuadrados. Las variables de la expresión son ordenadas en función de su peso y siguiendo el código Gray, de manera que sólo una de las variables varía entre celdas adyacentes. La transferencia de los términos de la tabla de verdad al mapa de Karnaugh se realiza de forma directa, albergando un 0 ó un 1, dependiendo del valor que toma la función en cada fila.

En la Figura 1 podemos observar el mapa de Karnaugh de f_1 con cuatro conjuntos encerrados por distintos colores:

• La función del conjunto amarillo se produce porque se observa que ninguna x_i varia durante su conjunto. Como x_2, x_3 y x_4 no varian pero valen 0, entonces obtenemos como función del conjunto $x_1\overline{x_2}\,\overline{x_3}\,\overline{x_4}$.

• La función del conjunto rojo se produce al observar que x_1 y x_2 no varian. Sin embargo, como $x_1$es cero, obtenemos como función del conjunto $\overline{x_1}x_2$.

• En el caso del conjunto verde, x_1 y x_3 no varian pero x_1 es cero, por lo que resulta $\overline{x_1}x_3$.

• Finalmente, en el conjunto azul, x_1 y x_4 no varian pero x_1 es cero, por lo que resulta $\overline{x_1}x_4$.

De esta manera,

$$f_1(x_1,x_2,x_3,x_4) = x_1\overline{x_2}\,\overline{x_3}\,\overline{x_4}+\overline{x_1}x_2+\overline{x_1}x_3+\overline{x_1}x_4 = x_1\overline{x_2}\,\overline{x_3}\,\overline{x_4}+\overline{x_1}(x_2+x_3+x_4)$$

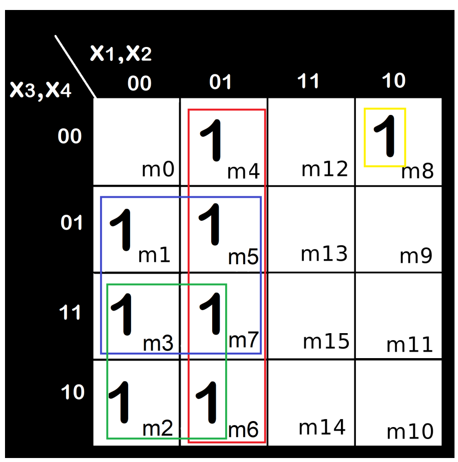

Figure 1: Mapa de Karnaugh $f_1(x_1,x_2,x_3,x_4)$

El análisis para los siguientes casos se realiza de la misma manera, pero no será detallado. Solo se procederá a mostrar las funciones simplificados y sus respectivos mapas de Karnaugh.

$$f_2(x_1,x_2,x_3,x_4) = x_2\overline{x_3}\,\overline{x_4}+\overline{x_2}x_3+\overline{x_2}x_4 = x_2\overline{x_3}\,\overline{x_4}+\overline{x_2}(x_3+x_4)$$

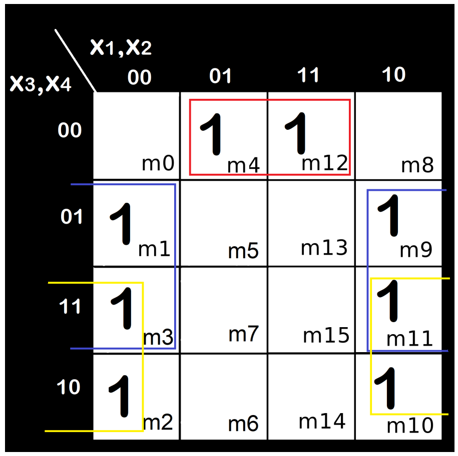

Figure 2: Mapa de Karnaugh $f_2(x_1, x_2, x_3, x_4)$

$$f_3(x_1, x_2, x_3, x_4) = \overline{x_3}x_4 + x_3\overline{x_4}$$

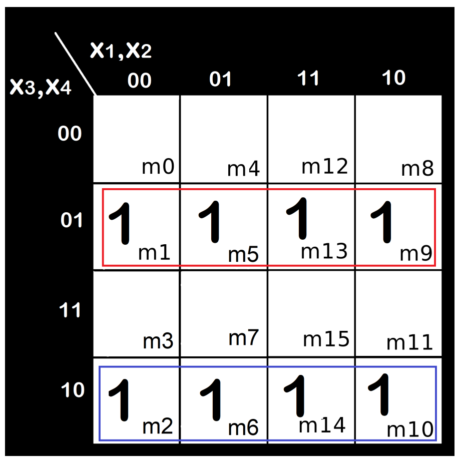

Figure 3: Mapa de Karnaugh $f_3(x_1, x_2, x_3, x_4)$

$$f_4(x_1, x_2, x_3, x_4) = x_4$$

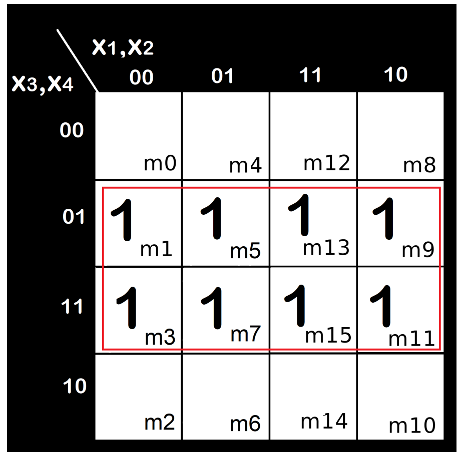

Figure 4: Mapa de Karnaugh $f_4(x_1, x_2, x_3, x_4)$

El circuito lógico resultante se muestra en la Figura 5, donde los Xi corresponden a la entrada y los Fi a la salida del circuito. Respetando la consigna, solo se utilizaron compuestas NOT, AND y OR.

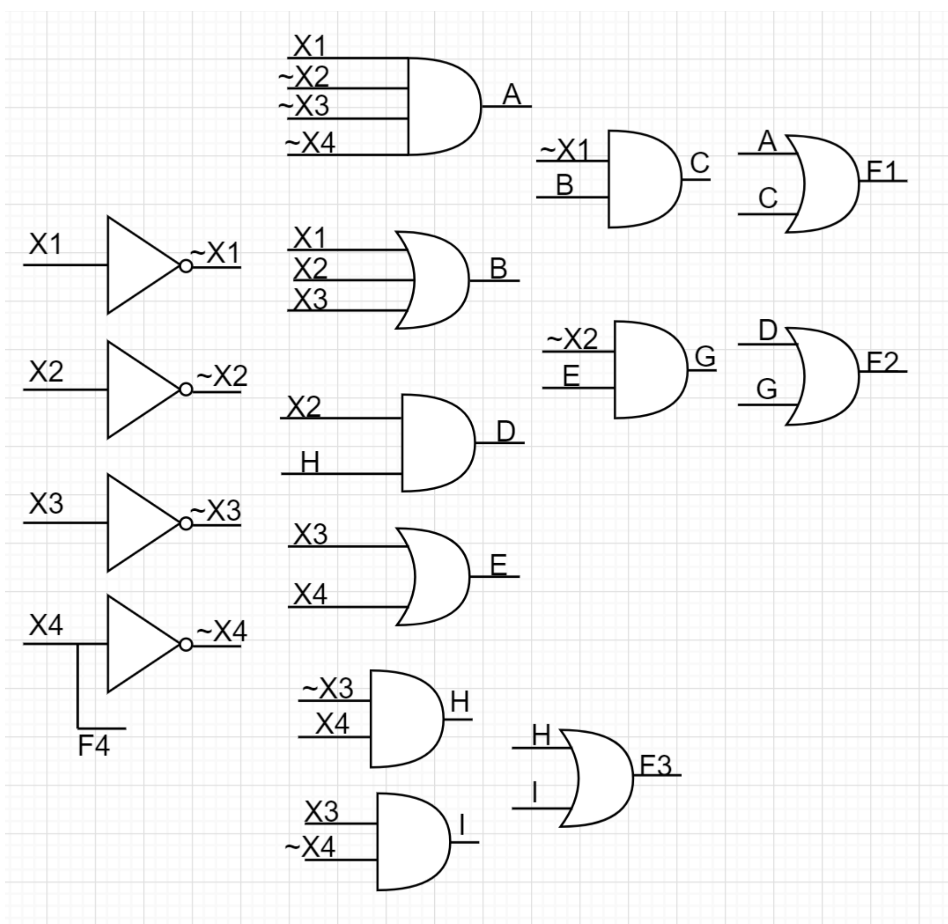

Figure 5: Circuito resultante realizado en draw.io