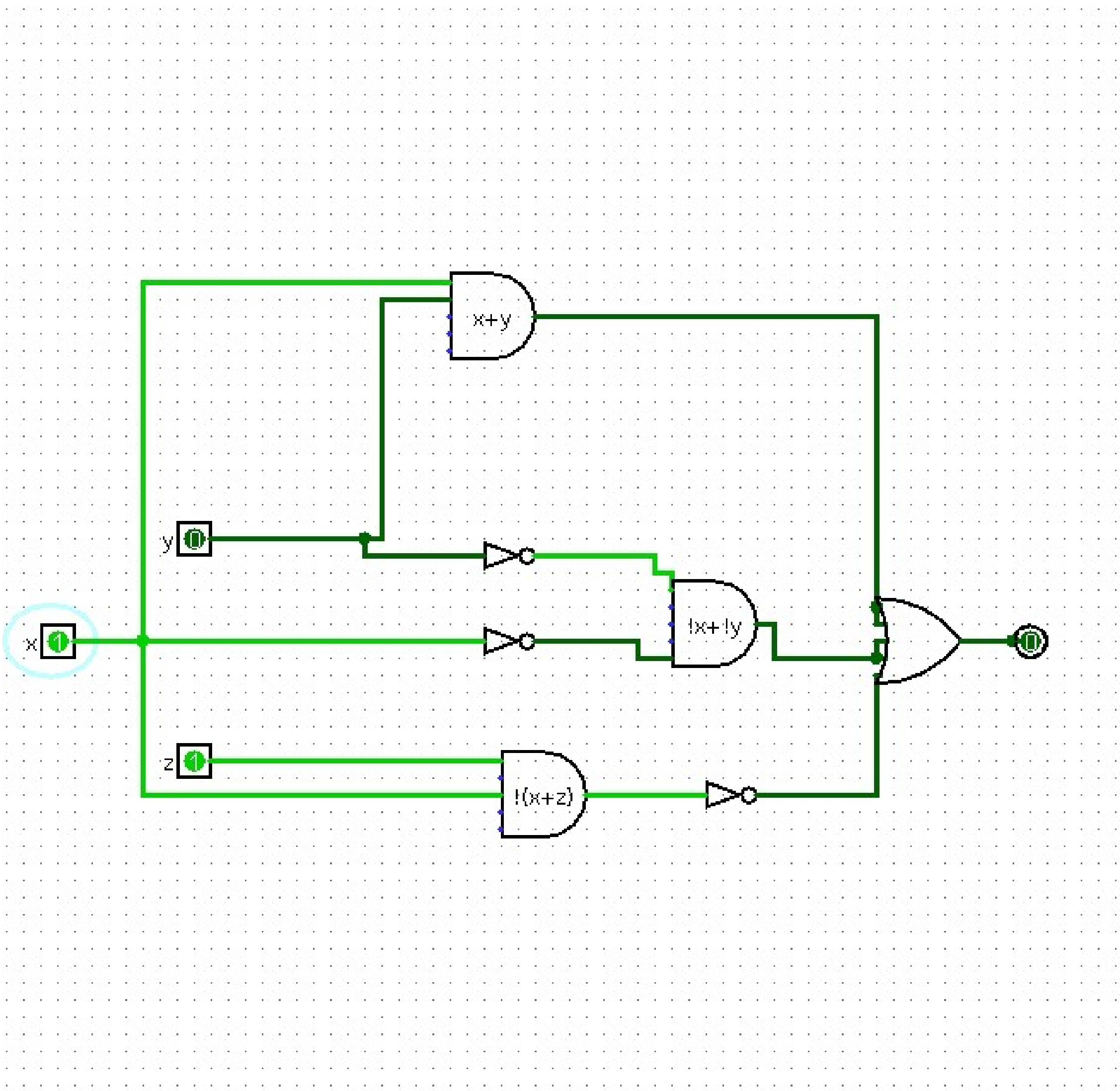
Fundamentos de los computadores
Practica 2: Álgebra de Boole

Ejercicio 1:

Circuito implementado en Logisim (archivo ej1.circ) para la función

$f(x,y,z)=(x+y)(\overline{x}+\overline{y})(\overline{x+z})$



Ejercicio 2:

Simplificamos la siguiente expresión: $f = \left[a\bar{b}(c + bd) + \bar{a}\bar{b}\right]c$

$$f = \left[a\bar{b}c + \bar{a}\bar{b}\right]c$$

$$f = a\bar{b}c + \bar{a}\bar{b}c$$

$$f = \bar{b}c$$

Tabla de verdad para $f = \left[a\bar{b}(c + bd) + \bar{a}\bar{b}\right]c$

a	*b*	*c*	*d*	*S*
0	0	0	0	0
0	0	0	1	0
0	0	1	0	**1**
0	0	1	1	**1**
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	**1**
1	0	1	1	**1**
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	0

Circuito implementado en Logisim (archivo *ej2-original.circ)*

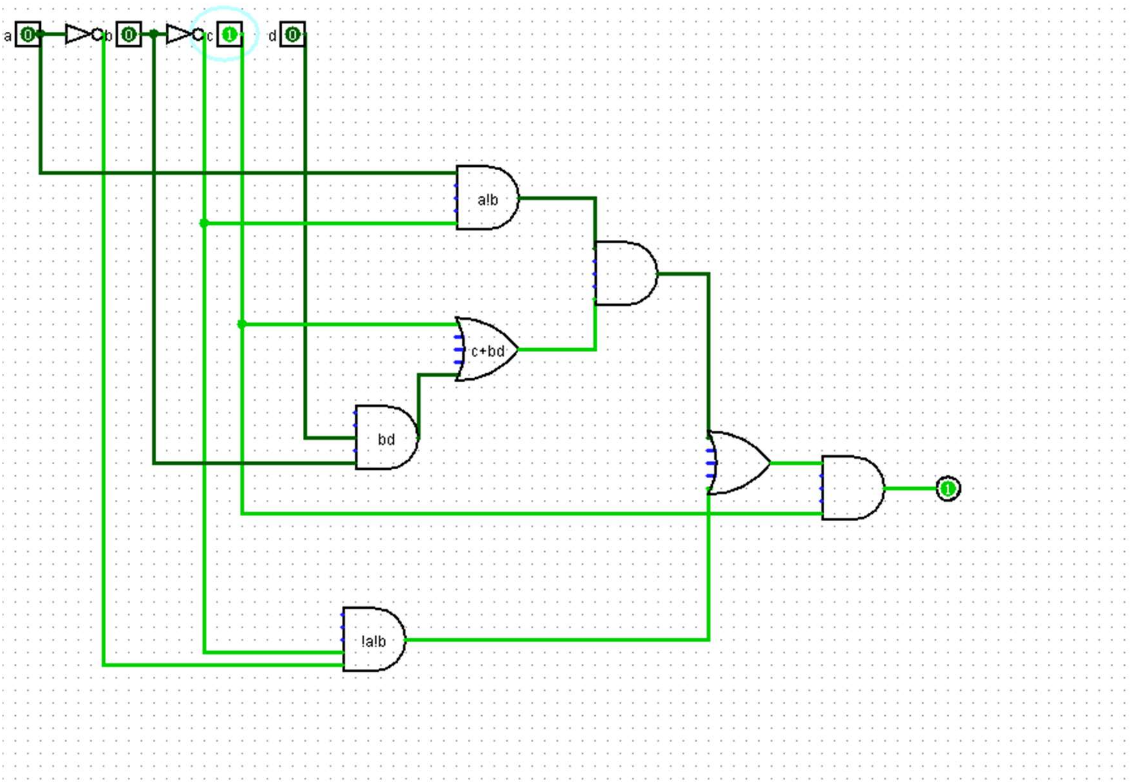
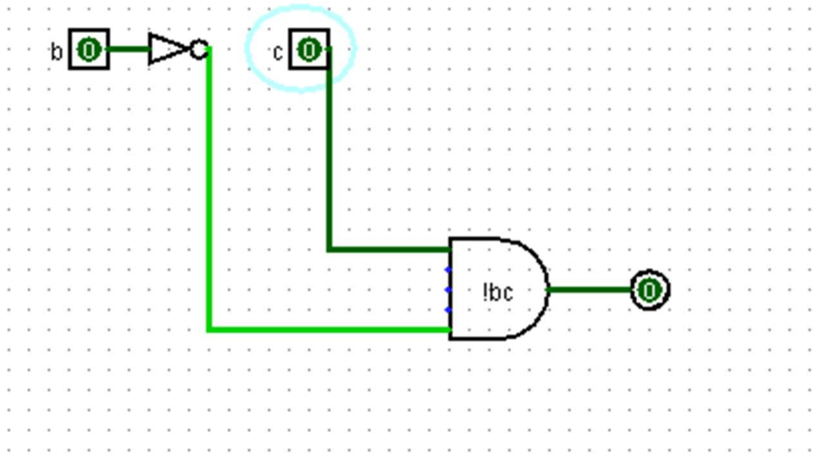


Tabla de verdad para $f = \overline{b}c$

b	***c***	***S***
0	0	0
0	1	**1**
1	0	0
1	1	0

Como Podemos ver en ambas tablas de verdad, la única situación que da como salida 1 es cuando b=0 y c=1, sin importar el valor de las otras variables.

Circuito implementado en Logisim (archivo *ej2-simplificada.circ)*



Ejercicio 3:

Apartado a:

Entradas				*Salida*
a	b	c	d	S
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

Tabla 1

Dada la siguiente tabla averiguamos:

La función algebraica en forma de suma de productos:

$$f(a,b,c,d) = \sum_{4} (0,2,4,5,6,8,10,11,12,13,14)$$

y en forma de producto de sumas:

$$f(a,b,c,d) = \prod_{4} (1,3,7,9,15)$$

Apartado b:

La tabla de Karnaugh y la expresión en forma de suma de productos:

ab\cd	**00**	**01**	**11**	**10**
00	1	0	0	1
01	1	1	0	1
11	1	1	0	1
10	1	0	1	1

$$f = a\bar{b}c + b\bar{c} + a\bar{d} + \bar{a}\bar{d}$$

Apartado c:

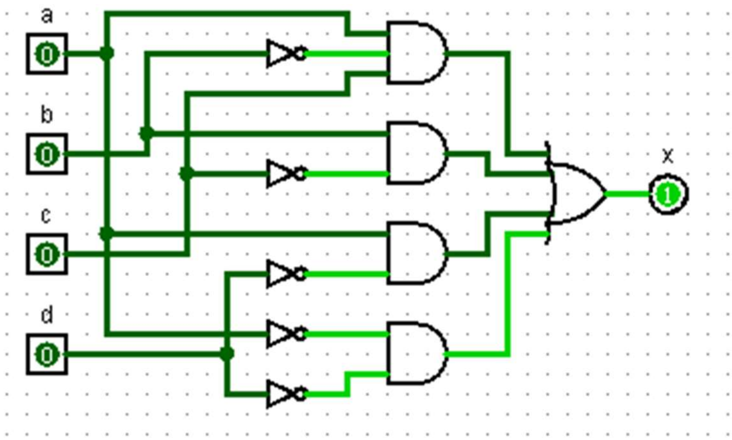
La tabla de Karnaugh y la expresión en forma de producto de sumas:

ab\cd	00	01	11	10
00	1	0	0	1
01	1	1	0	1
11	1	1	0	1
10	1	0	1	1

$$f = (a + \bar{c} + \bar{d})(\bar{b} + \bar{c} + \bar{d})(b + c + \bar{d})$$

Apartado d:

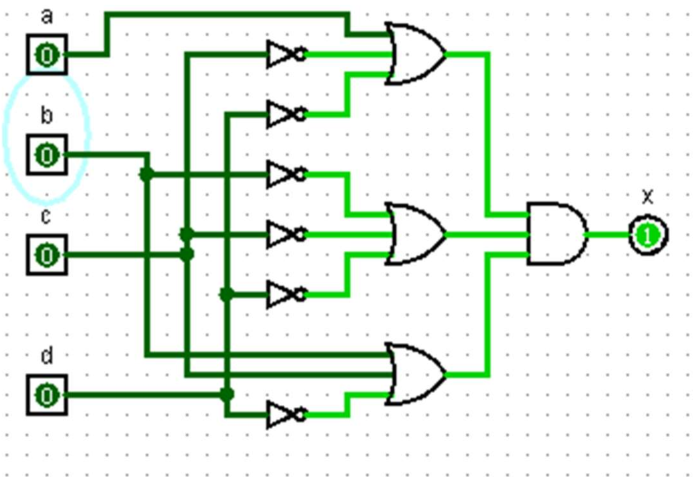
Circuito implementado en Logisim en forma de suma de productos (*archivo ej3d-sop.circ)*



Y su tabla de verdad:

a	b	c	d	x
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

Circuito implementado en Logisim en forma de producto de sumas (archivo ej3d-pos.circ)



Y su tabla de verdad

a	b	c	d	x
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0

Ya que ambas tablas de verdad coinciden, ambas formas de implementar el circuito son correctas.

Ejercicio 4:

$$f = \prod_4 (1,2,5,6,7,9,12,14,15)$$

Apartado a:

Obtenemos la tabla de Karnaugh para obtener su expresión mínima en forma de suma de productos:

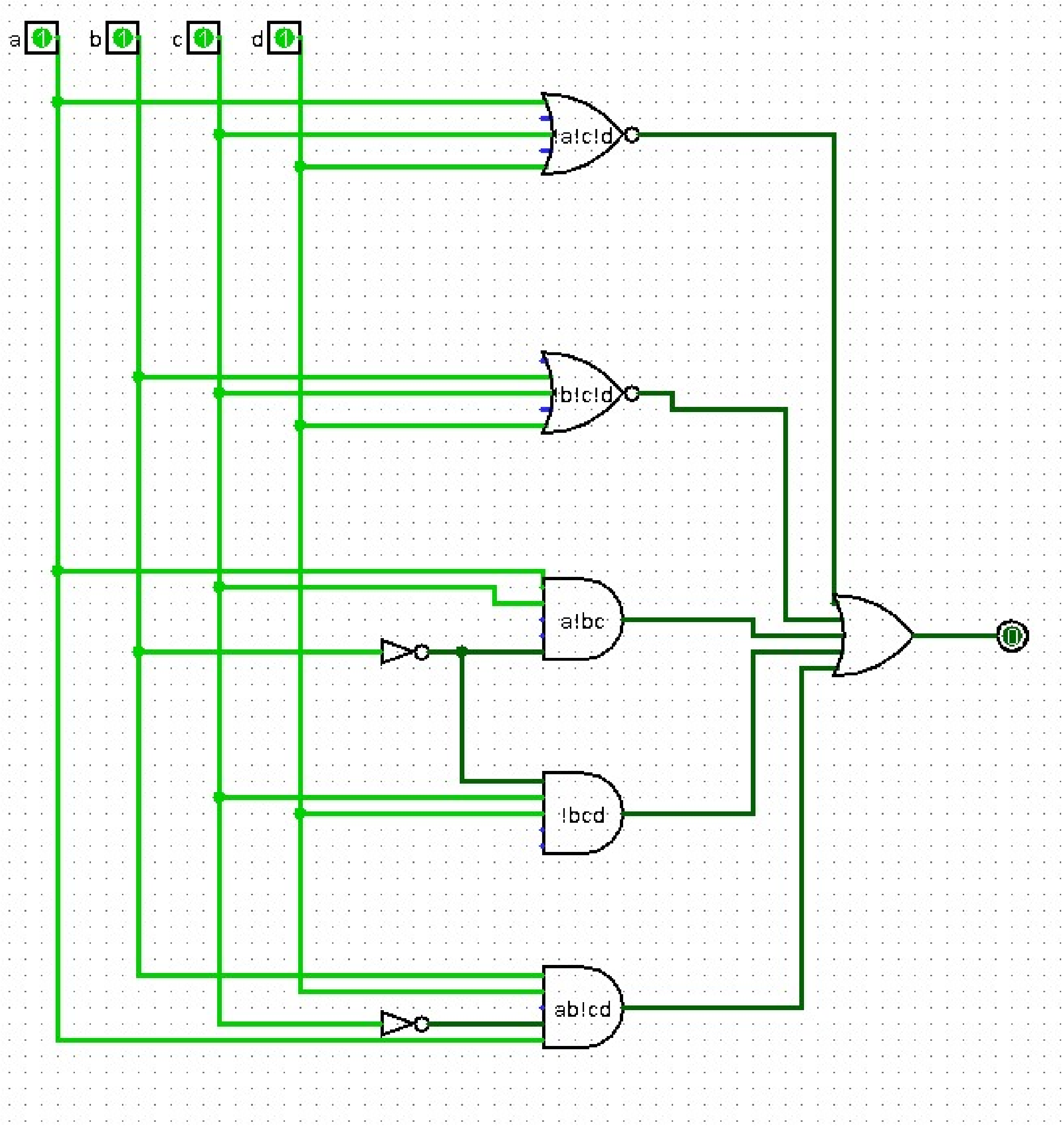
ab\cd	00	01	11	10
00	1	0	1	0
01	1	0	0	0
11	0	1	0	0
10	1	0	1	1

Su expresión en forma de suma de productos:

$$f = \bar{a}\bar{c}\bar{d} + \bar{b}\bar{c}\bar{d} + a\bar{b}c + \bar{b}cd + ab\bar{c}d$$

Apartado b:

Circuito implementado en Logisim (archivo ej4b.circ) para la expresión anterior.



Y obtenemos su tabla de verdad:

a	b	c	d	x
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	0
1	1	1	1	0

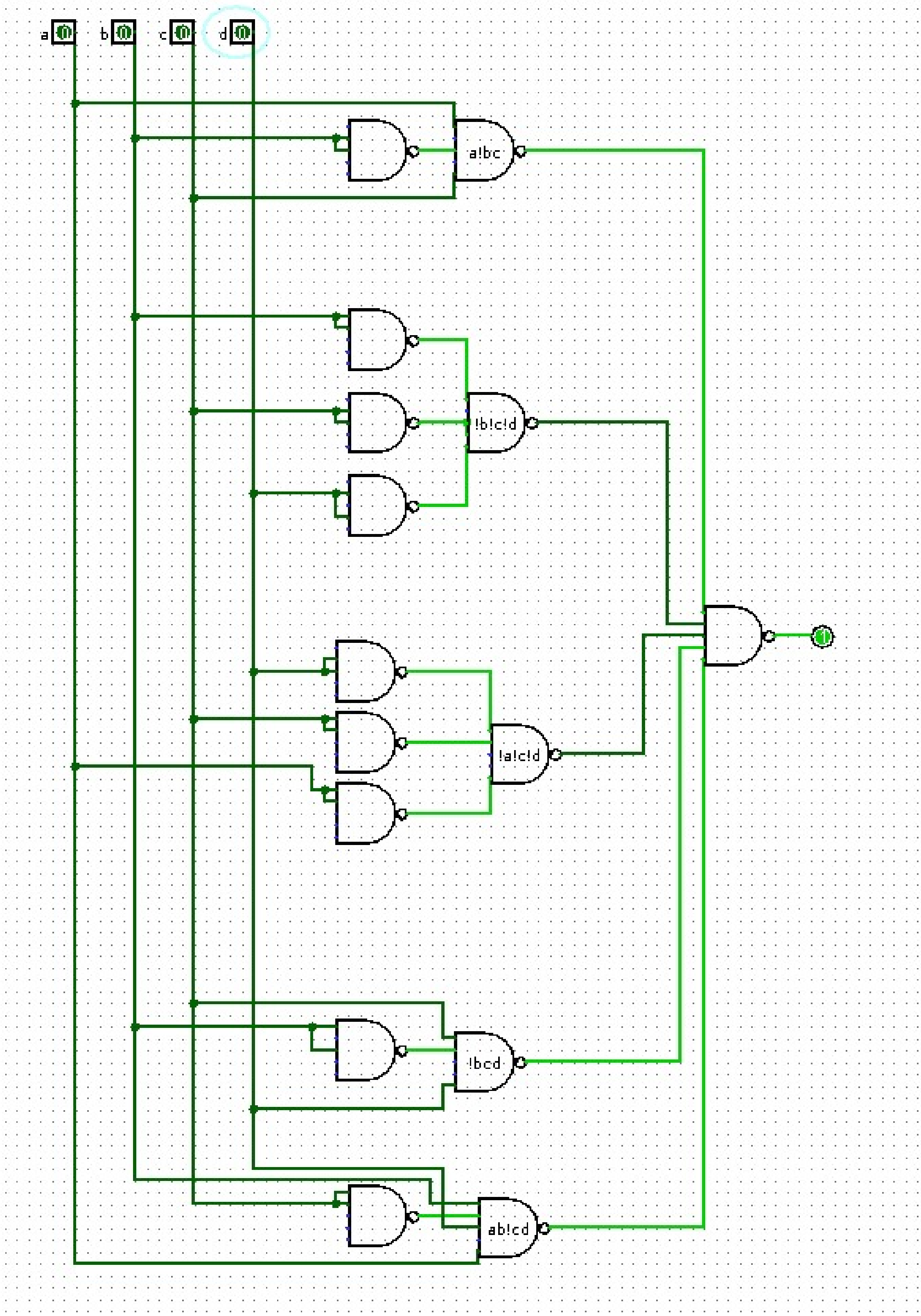
Apartado c:

Primero obtenemos la expresión para que pueda ser implementada solo con puertas NAND:

$$f = \overline{\overline{\bar{a}\bar{c}\bar{d} + \bar{b}\bar{c}\bar{d} + a\bar{b}c + \bar{b}cd + ab\bar{c}d}}$$

$$f = \overline{\overline{\bar{a}\bar{c}\bar{d}} * \overline{\bar{b}\bar{c}\bar{d}} * \overline{a\bar{b}c} * \overline{\bar{b}cd} * \overline{ab\bar{c}d}}$$

Circuito implementado en Logisim (archivo ej4c.circ) utilizando solo puertas NAND:



Y su tabla de verdad:

a	b	c	d	x
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	0
1	1	1	1	0

Ya que ambas tablas de verdad coinciden, ambas formas de implementar el circuito son correctas.

Ejercicio 5:

Para la función:

$$f = \sum_{5}(1,4,6,7,8,9,11,14,17,20,22,23,25,30,31) + \sum_{0}(2,13,15,18,24)$$

Primero obtenemos su tabla de verdad:

a	b	c	d	e	S
0	0	0	0	0	0
0	0	0	0	1	1
0	0	0	1	0	**X**
0	0	0	1	1	0
0	0	1	0	0	1
0	0	1	0	1	0
0	0	1	1	0	1
0	0	1	1	1	1
0	1	0	0	0	1
0	1	0	0	1	1
0	1	0	1	0	0
0	1	0	1	1	1
0	1	1	0	0	0
0	1	1	0	1	**X**
0	1	1	1	0	1
0	1	1	1	1	**X**
1	0	0	0	0	0
1	0	0	0	1	1
1	0	0	1	0	**X**
1	0	0	1	1	0
1	0	1	0	0	1
1	0	1	0	1	0
1	0	1	1	0	1
1	0	1	1	1	1
1	1	0	0	0	**X**
1	1	0	0	1	1
1	1	0	1	0	0
1	1	0	1	1	0
1	1	1	0	0	0
1	1	1	0	1	0
1	1	1	1	0	1
1	1	1	1	1	1

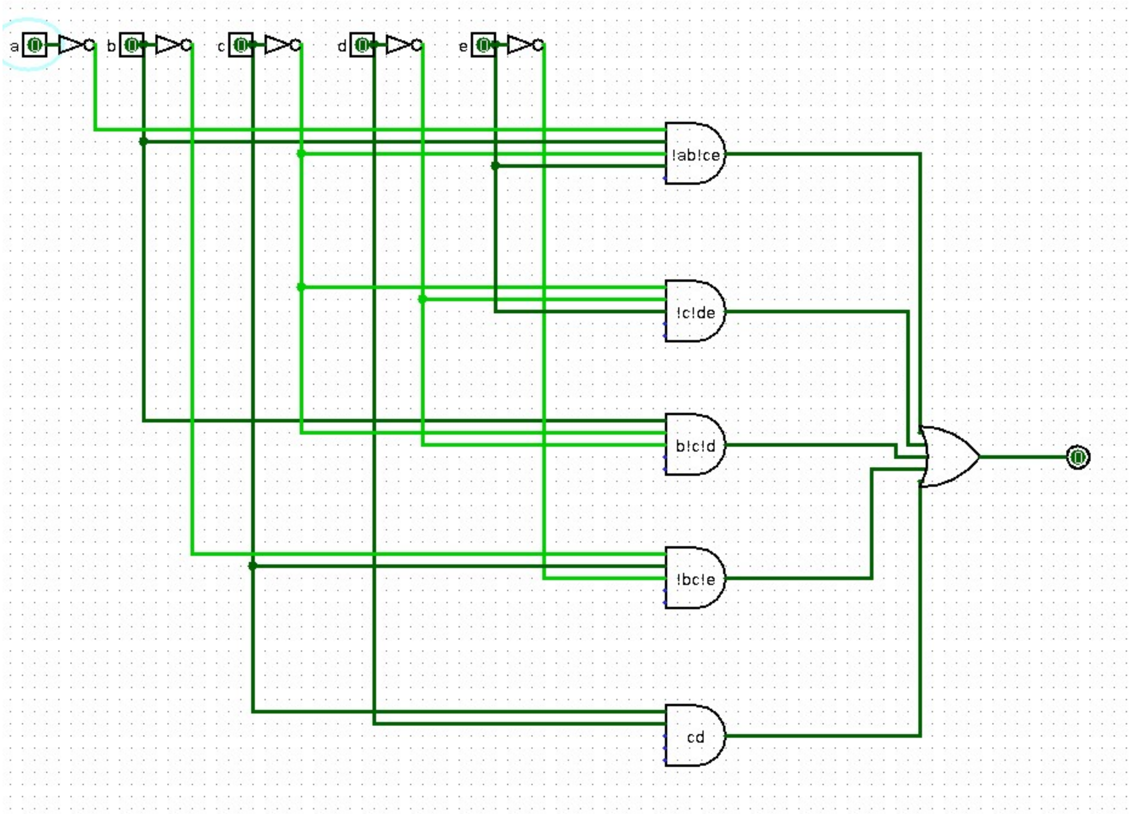
Lo que ocurre con las indiferencias (marcadas como X) es que pueden ser utilizadas tanto como 1 o 0 en la tabla de Karnaugh, lo cual nos facilita sacar las expresiones de suma de productos o producto de sumas.

ab\cde	000	001	011	010	110	111	101	100
00	0	1	0	x	1	1	0	1
01	1	1	1	0	1	x	x	0
11	x	1	0	0	1	1	0	0
10	0	1	0	x	1	1	0	1

Y su expresión en forma de suma de productos:

$$f = \bar{c}\bar{d}e + b\bar{c}\bar{d} + \bar{a}b\bar{c}e + cd + \bar{b}c\bar{e}$$

Circuito implementado en Logisim (archivo ej5.circ)



Y su tabla de verdad:

a	b	c	d	e	x
0	0	0	0	0	0
0	0	0	0	1	1
0	0	0	1	0	0
0	0	0	1	1	0
0	0	1	0	0	1
0	0	1	0	1	0
0	0	1	1	0	1
0	0	1	1	1	1
0	1	0	0	0	1
0	1	0	0	1	1
0	1	0	1	0	0
0	1	0	1	1	1
0	1	1	0	0	0
0	1	1	0	1	0
0	1	1	1	0	1
0	1	1	1	1	1
1	0	0	0	0	0
1	0	0	0	1	1
1	0	0	1	0	0
1	0	0	1	1	0
1	0	1	0	0	1
1	0	1	0	1	0
1	0	1	1	0	1
1	0	1	1	1	1
1	1	0	0	0	1
1	1	0	0	1	1
1	1	0	1	0	0
1	1	0	1	1	0
1	1	1	0	0	0
1	1	1	0	1	0
1	1	1	1	0	1
1	1	1	1	1	1

Como lo podemos ver en esta tabla, los valores de salida X que hemos contado como 1 en la tabla de Karnaugh dan salida 1, y los que no hemos utilizado dan salida 0.