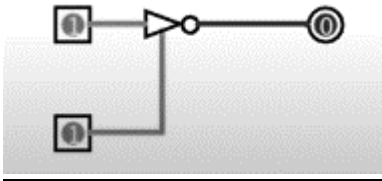


Ejercicios con Logisim

Implemente el siguiente circuito a modo de practica con logisim

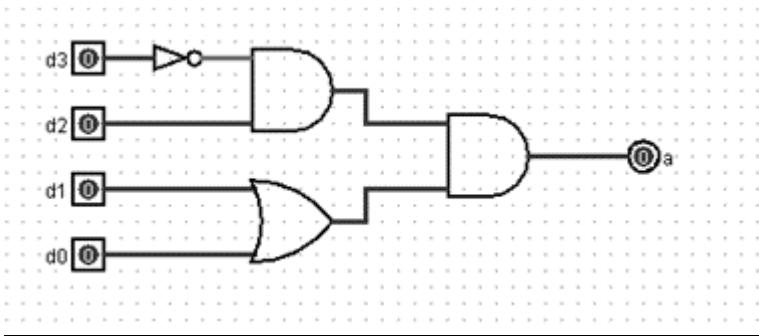
Ejemplo:

Modifique el pin de control del componente inversor controlado y vea que sucede en la salida.

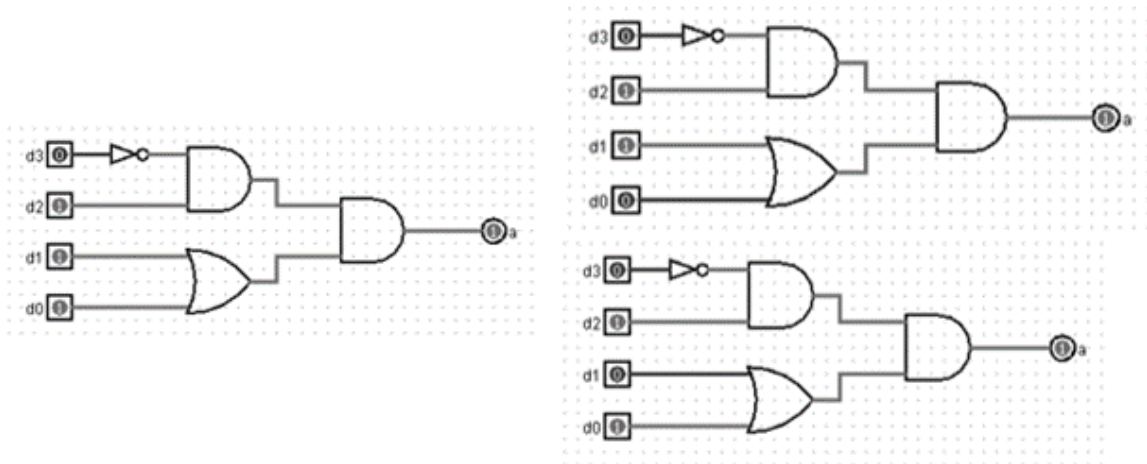


Realice el mismo circuito pero en esta ocasión cambie el componente inversor controlado por el componente buffer controlado y vea que sucede con la salida.

Ejemplo:



Modifique las entradas y vea que sucede en la salida.



Complete la siguiente tabla de verdad

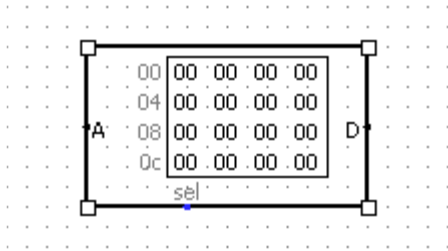
d3	d2	d1	d0	a
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

Conversión binaria, decimal y hexadecimal

Hexadecimal	Binary	Decimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
A	1010	10
B	1011	11
C	1100	12
D	1101	13
E	1110	14
F	1111	15

Memoria ROM

La memoria funciona igual que la RAM con la única diferencia es que esta solo tiene tres pines (A,D, sel).



La ROM puede almacenar hasta 4,096 valores (lo que se especifica en el atributo Número De Bits De Direccionamiento), y cada uno de ellos puede ser de hasta 32 bits (lo que se especifica en el atributo Número De Bits De Datos). Un circuito puede acceder a los valores de la ROM, pero no los puede modificar. El usuario puede cambiar los valores individualmente a través de la Herramienta De Cambio, o puede editarlos todos con la Herramienta De Menú.

A diferencia de la RAM, los contenidos de la ROM se almacenan como atributos del componente. Así, si un circuito que contiene una ROM se utiliza dos veces, entonces ambas ROM contendrán los mismos valores. También a causa de este comportamiento, los contenidos de la ROM se almacenan en archivos creados por Logisim.

Los valores de la ROM se muestran en el componente. Las direcciones se pueden ver en gris a la izquierda del área de visualización. Dentro, cada valor se proporciona en hexadecimal. El valor de la dirección seleccionada se mostrará en texto de color inverso (blanco sobre negro).

Pines

A sobre el borde oeste (entrada, el número de bits coincide con lo que indica el atributo Número De Bits De Direccionamiento)

Permite seleccionar a cual de los valores se está accediendo desde el circuito.

D sobre el borde este (entrada/salida, el número de bits coincide con lo que se indica en el atributo Número De Bits De Datos)

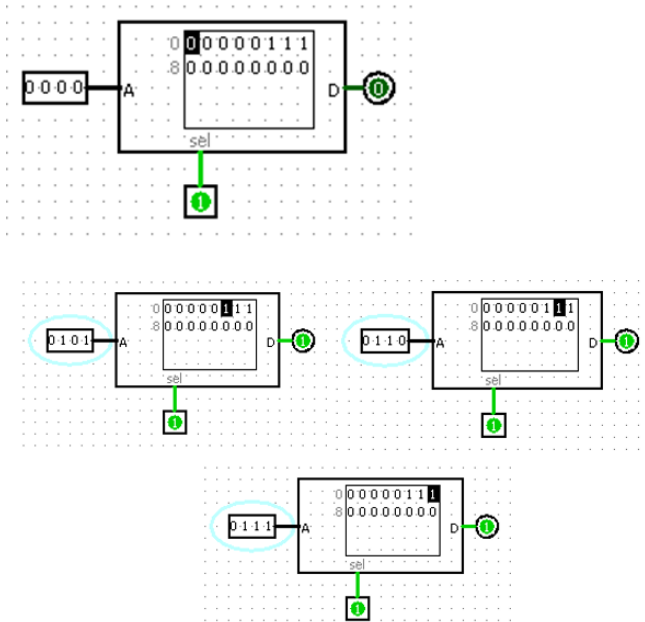
Proporciona en el pin *D* el valor de la dirección seleccionada si *sel* es 1 u oscilante; si *sel* es 0, entonces *D* será oscilante.

sel sobre el borde sur (entrada, 1 bit)

Para un solo módulo ROM esta entrada no tiene función, se puede ignorar. Para más de un módulo ROM en paralelo, se puede utilizar esta entrada para conectar y desconectar un módulo ROM en función de si su valor es 1 o 0. En otras palabras, cuando vale 0, no se proporciona ningún valor en el pin de salida *D*.

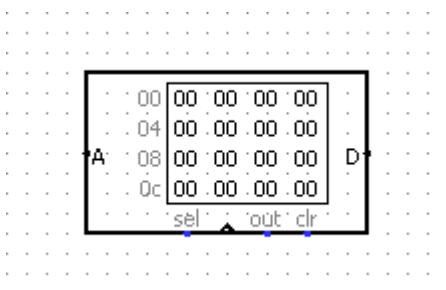
Ejemplo:

Implemente el siguiente circuito



En este circuito, se puede observar que se muestra la representación de la dirección de una ROM de 16 bits, se pueden ver las salidas a las que se activará el circuito y a las que no según la posición. Por ello es que en las posiciones 5, 6 y 7 hay un 1. Cuando en A se introduzca alguno de estos dígitos en binario, se marcará la respectiva posición y la salida será 1, es decir, el circuito se activará, esto se ve reflejado en la figura

Memoria RAM en Logisim



-**A**: sobre el borde oeste (entrada de los bits de dirección de la memoria RAM). Permite especificar a que valor de la memoria se accede.

-**D**: sobre el borde este (entrada o salida, según los valores de sel y out, de los bits de datos que carguemos en memoria).

-**Out** : sobre el borde sur (entrada, 1 bit) (Lectura de datos L ó R)

Especifica si la RAM debería proporcionar a la salida el valor de la dirección seleccionada (A). La opción de salida está seleccionada si out vale 1 o indefinido; si out vale 0, entonces D se comporta como una entrada de forma que se almacena su valor en los flancos de subida del reloj.

-**Sel**: en el borde sur (entrada, 1 bit) (**Triestado de salida OE**)

Para un sólo módulo RAM, se puede ignorar esta entrada. Si tienes más de un módulo

RAM en paralelo, puedes utilizar esta entrada para conectar o desconectar cada

módulo RAM, en función de si su valor es 0 o 1. En otras palabras, cuando vale 0, no se proporciona ningún valor en el pin de salida, *D*, y no se cambiará ningún valor de la

memoria cuando el reloj pase de 0 a 1.

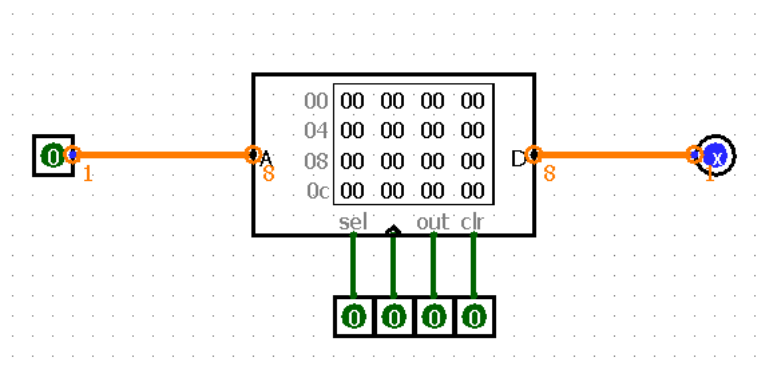
-Triangulo sobre el borde sur: (entrada, 1 bit) (**Carga de datos E o W**)

Entrada de reloj : Cuando *out* vale 0, y esta entrada pasa de 0 a 1 (y *sel* vale 1/indefinido y *clr* vale 0), entonces el valor de la posición de memoria de la dirección seleccionada pasa a ser el mismo que el proporcionado en el pin *D*. Sin embargo, mientras el reloj permanezca a 0 o a 1 el valor del pin *D* no será almacenado en memoria.

-**Clr** (clear): sobre el borde sur (entrada, 1 bit).

Cuando vale 1, y *sel* vale 1 o indefinido, se resetea la memoria, es decir, los valores de todas sus posiciones de memoria pasan a ser 0, sin importar lo que valgan las demás entradas.

Ejemplo: Conecte la memoria RAM de la siguiente manera como lo indica la figura, para luego modificarlo según requerimientos:



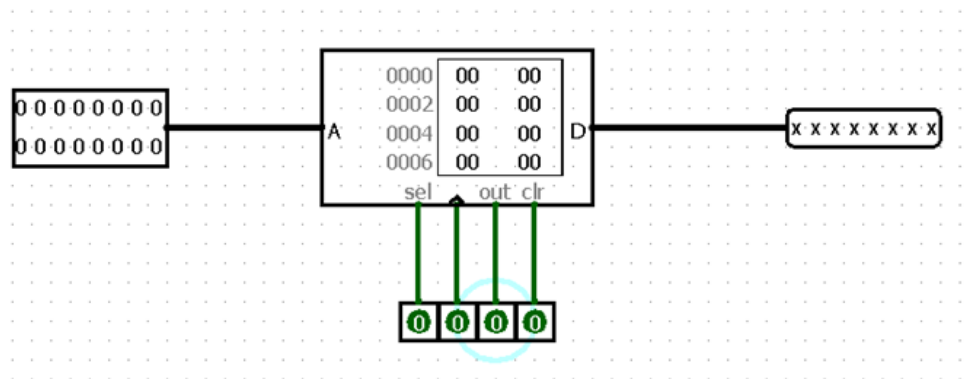
Modifique según los requerimientos

Ejercicio 1:

- Memoria RAM de 256x8 y realizar un circuito que permita leer en ella. Introducir en las direcciones 1Fh el dato 1C

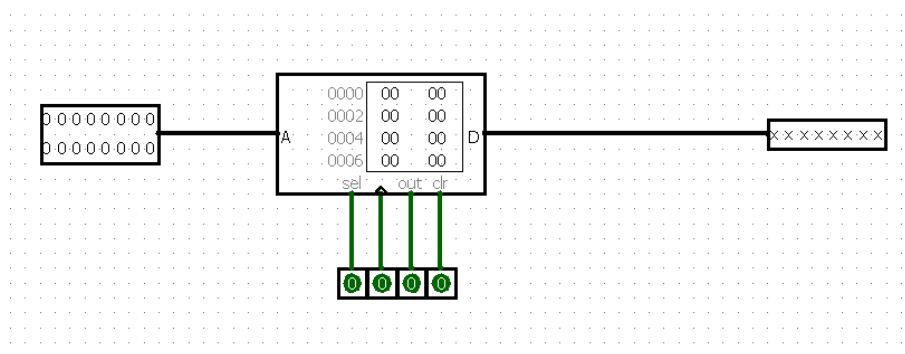
Ejercicio 2:

- Memoria RAM de 64Kx8 y realizar un circuito que permita leer la memoria. Introducir en las direcciones 0023h el dato 5A y en la dirección C120h el dato 2F.



Ejercicio 3:

Memoria RAM de 64Kx8 y realizar un circuito que permita escribir en ella. Introducir en las direcciones 0023h el dato 3B y en la dirección A120h el dato 25.



Ejercicio 4:

Memoria RAM de 64Kx8, realizar un circuito que permita leer y escribir en ella. Introducir en las direcciones 0023h el dato 3B y en la dirección A120h el dato 2A.

