

# PROYECTO N°1

Implementación del diagrama lógico de un circuito  
controlador - TEMA B

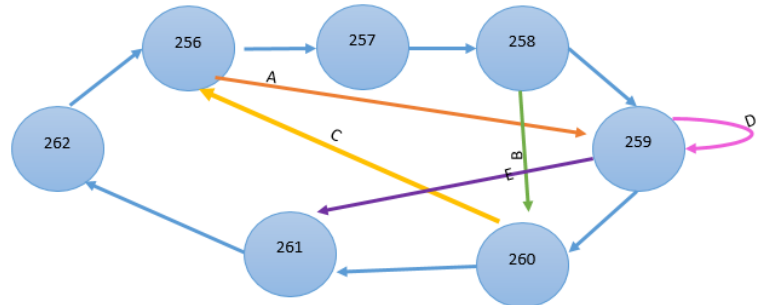
Arquitectura de  
Computadoras

## Informe del Proyecto

### Introducción:

El circuito es un controlador de estados que permite pasar entre los estados de acuerdo a la figura enunciada.

Las entradas (A-E) representan las líneas externas que, en caso de estar alguna activa, cambia el controlador a su modo alternativo.



Este modo alternativo realiza saltos especiales a estados específicos, a diferencia del normal que cicla automáticamente por todos los estados en orden desde el 256 al 262.

El circuito fue implementado en la versión Logisim Evolution 3.8.0

### Etapa 1:

Tabla de estados del circuito:

A	B	C	D	E	Estado Actual	Codificación Actual	Próximo Estado	Próxima Codificación
0					256	000	257	001
1					256	000	259	011
					257	001	258	010
	0				258	010	259	011
	1				258	010	260	100
			0	0	259	011	260	100
			0	1	259	011	261	101
			1	0	259	011	259	011
		0			260	100	261	101
		1			260	100	256	000
					261	101	262	110
					262	110	256	000

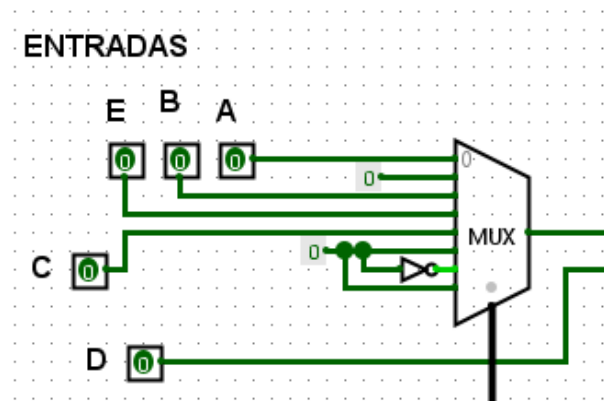
Las columnas “Codificación Actual” y “Próxima Codificación” las adoptamos como una optimización al circuito. Esto nos permite recorrer todo el circuito utilizando únicamente los últimos 3 bits, dejando los primeros 6 como constantes. Es decir, cada estado resulta de la suma de 256 y nuestra codificación

### Premisas del proyecto:

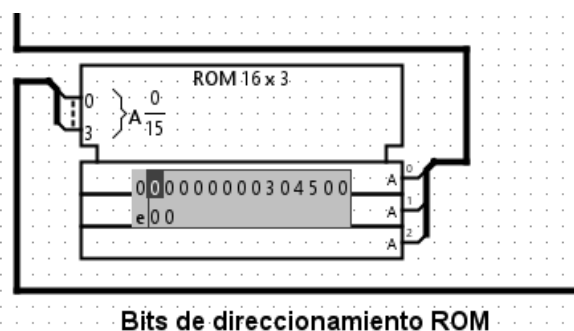
- Se asume que no se pueden activar dos líneas externas en simultáneo.
- Codificamos la “Salida” del circuito en base decimal para que sea más fácil de verificar y visualizar.
- El contador se mantiene constantemente activo por defecto, excepto cuando se encuentra en el estado 259 y se activa la línea D, donde se apaga el enable del contador para que se mantenga en el mismo estado.

### Decisiones tomadas en el contador:

- La entrada “Enable” está cableada a la negación de la salida 110 y la compuerta “D”, ya que en dicha situación el contador se mantiene en el estado actual. En cualquier otro momento el contador trabaja normalmente.
- La entrada “Up/Down” está cableada a una constante con valor “1” ya que el circuito siempre realiza un cambio de estado hacia adelante, o bien realiza un salto utilizando la entrada “Load”.
- Las entradas “Carry” y “Clear” no están cableadas ya que no es necesaria su utilización.
- La entrada “Load” se encuentra cableada a una compuerta OR, utilizando un MUX, el cual activa esta entrada cuándo se debe realizar un salto en el Modo Alternativo, y también se utiliza el bit más significativo de la salida del contador para poder codificar el inicio del contador en el estado 256.
- Los bits de salida del contador son utilizados como líneas de selección del MUX, como bits de entrada de la ROM y como los bits de salida de todo el circuito.



ROM: siempre codifica los saltos especiales que debe realizar el circuito, los cuáles van a ser activados cuando se active la entrada “Load” del contador. La ROM codifica dichos saltos en hexadecimal, tomando como entrada 4 bits, el primer bit representa el más significativo del contador y es utilizado para distinguir cuándo el contador se encuentra en 0 y es necesario realizar el salto al estado inicial 256. Los 3 bits restantes respetan la convención adoptada de codificación para cargar el estado al que debe saltar el contador.



MUX: es utilizado para activar la entrada “Load” del contador cuándo el circuito entra en Modo Alternativo. Mientras que la ROM codifica el estado al cuál se va a realizar el salto, el MUX determina cuándo se va a realizar. Las líneas de selección corresponden a los 3 bits menos significativos del estado actual del contador.

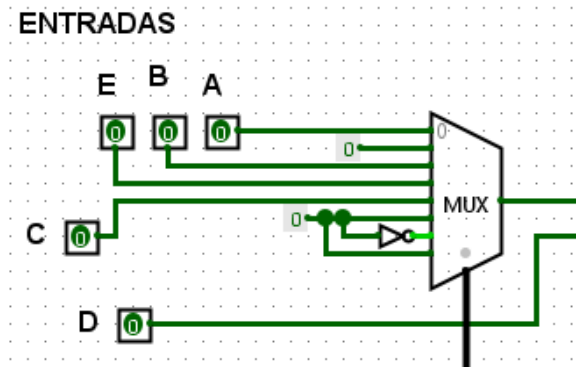
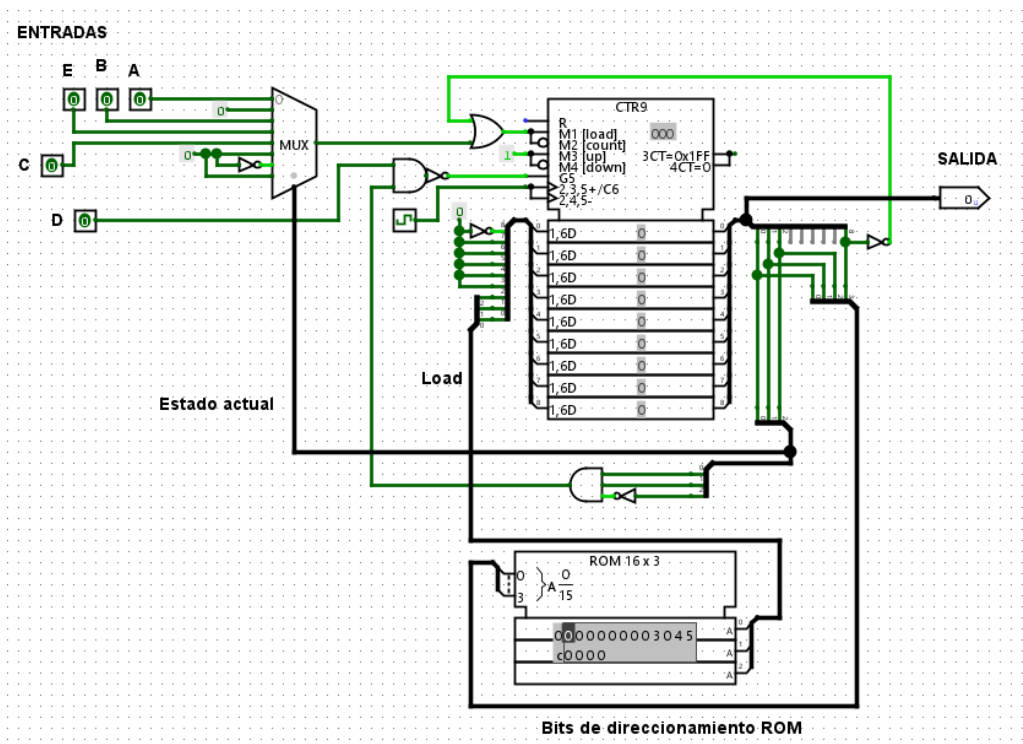


Diagrama del circuito:



### Conclusión:

El proyecto fue de utilidad para comprender más a fondo el funcionamiento de un contador, aplicándolo particularmente a este controlador de estados. Además, pudimos combinarlo con otros circuitos, como la ROM y el MUX para que estos trabajen en conjunto y en sincronía, mediante el uso de un reloj.

Podríamos haber hecho uso de un contador de menos bits, usando solo los últimos tres bits. Esta solución permite ahorrar en hardware, pero resulta poco extensible en el caso de necesitarlo.