

Electrónica Digital 1

Lógica combinacional -tiempos de propagación

Ferney Alberto Beltrán Molina



UNIVERSIDAD
NACIONAL
DE COLOMBIA

Marzo 2020

Contacto

Nombre: Ferney Alberto Beltrán Molina, Ing, MSc, PhD(c)
Email: fabeltranm@unal.edu.co
oficina: Centro de Investigación e Innovación

Contenido

Recordando

ejemplo comparador 1bit

Mapas de karnaugh

Tiempos de propagación

Multiplexores / demultiplexores

Índice

Recordando

ejemplo comparador 1bit

Mapas de karnaugh

Tiempos de propagación

Multiplexores / demultiplexores

Tipos de circuitos digitales

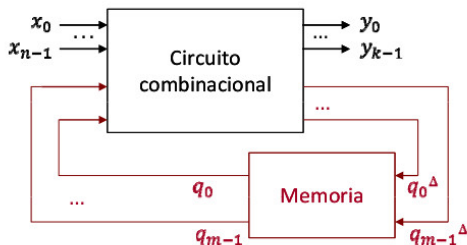
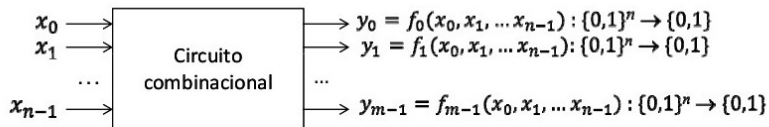
- ▶ **Circuitos combinacionales**

Las salidas del circuito en cada instante de tiempo dependen única de los valores de entrada. combina los valores de entrada en un instante de tiempo para calcular la salida

- ▶ **Circuitos secuenciales.**

Las salidas del circuito secuencial dependen tanto de los valores actuales como de los anteriores de las entradas; en otras palabras, depende de la secuencia de entrada.

Tipos de circuitos digitales



Álgebra de Boole propiedades

1 - Elemento inverso, $\bar{0}=1$, $\bar{1}=0$

2 - Idempotencia, $a+a=a$, $a \cdot a=a$

3 - Involución, $\overline{\overline{a}}=a$

4 - Asociatividad, $a+(b+c)=(a+b)+c$, $a \cdot (b \cdot c)=(a \cdot b) \cdot c$









5 - Absorción, $a+a \cdot b=a$, $a \cdot (a+b)=a$

6 - (sin nombre), $a+\bar{a} \cdot b=a+b$, $a \cdot (\bar{a}+b)=a \cdot b$

7 - de Morgan, $\overline{(a+b)}=\bar{a} \cdot \bar{b}$, $\overline{a \cdot b}=\bar{a}+\bar{b}$

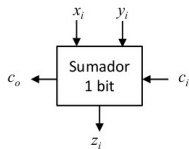
8 - de Morgan generalizada, $\overline{(a_1+a_2+\dots+a_n)}=\bar{a}_1 \cdot \bar{a}_2 \cdot \dots \cdot \bar{a}_n$, $\overline{a_1 \cdot a_2 \cdot \dots \cdot a_n}=\bar{a}_1+\bar{a}_2+\dots+\bar{a}_n$

Álgebra de Boole propiedades

nombre	símbolo	función
AND		$z = a \cdot b$
OR		$z = a + b$
INV		$z = \bar{a}$
NAND		$z = \overline{a \cdot b}$
NOR		$z = \overline{a + b}$
XOR		$z = a \oplus b = \bar{a} \cdot b + a \cdot \bar{b}$
XNOR		$z = \overline{a \oplus b} = \bar{a} \cdot \bar{b} + a \cdot b$
Tri-state		$z = a \text{ si } c = 1, z = HI \text{ si } c = 0$

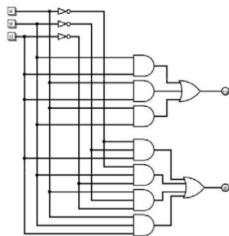
Funciones Booleanas - Resumiendo

- *Descripción Funcional*
- *Tabla de Verdad*
- *función(s) Booleana(s)*
- *Circuito Digital*



```
s <= x_i + y_i + c_i;  
if s = 0 then z_i <= 0; c_o = 0;  
  elsif s = 1 then z_i <= 1; c_o <= 0;  
    elsif s = 2 then z_i <= 0; c_o <= 1;  
      else z_i <= 1; c_o <= 1;  
    end if;  
  end if;  
end if;
```

x_i	y_i	c_i	c_o	z_i
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1



$$c_o = y_i \cdot c_i + x_i \cdot c_i + x_i \cdot y_i$$

$$z_i = \bar{x}_i \cdot \bar{y}_i \cdot c_i + \bar{x}_i \cdot y_i \cdot \bar{c}_i + x_i \cdot \bar{y}_i \cdot \bar{c}_i + x_i \cdot y_i \cdot c_i$$

respuesta - Ejemplo BCD2SSEG

- ▶ $a = x_1 + x_2 * x_0 + x_3 + \overline{x_2} * \overline{x_0}$
- ▶ $b = \overline{x_2} + \overline{x_1} * \overline{x_0} + x_1 * x_0$
- ▶ $c = \overline{x_1} + x_0 + x_2$
- ▶ $d = \overline{x_2} * \overline{x_0} + \overline{x_2} * x_1 + x_1 * \overline{x_0} +$
- ▶ $e = \overline{x_2} * \overline{x_0} + \overline{x_0} * x_1$
- ▶ $f = \overline{x_0} * \overline{x_1} + \overline{x_1} * x_2 + x_2 * \overline{x_0} + x_3$
- ▶ $g = \overline{x_0} * \overline{x_2} + \overline{x_1} * x_2 + x_1 * \overline{x_0} + x_3$

Índice

Recordando

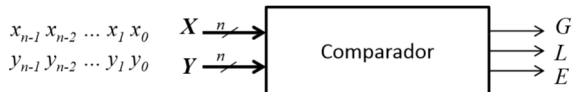
ejemplo comparador 1bit

Mapas de karnaugh

Tiempos de propagación

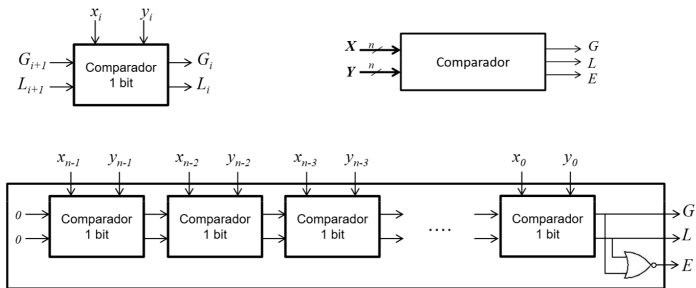
Multiplexores / demultiplexores

Comparador 1bit

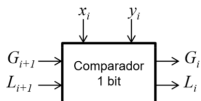


```
if X > Y then G <= 1;  
  elsif X < Y then L <= 1;  
    else E <= 1;  
  end if;  
end if;
```

Comparador 1bit



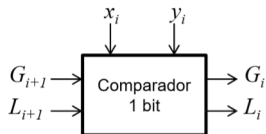
Comparador 1bit



	$n-1$	$n-2$	$n-3$	$n-4$	1	0
X:	1	0	1	1	0	...	0	0
Y:	1	0	1	0	1	...	1	0

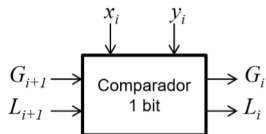
$G_n=0$								
$L_n=0$								

Comparador 1bit resultado de G_i



G_{i+1}	L_{i+1}	x_i	y_i	G_i	L_i
0	0	0	0	0	0
0	0	0	1	0	1
0	0	1	0	1	0
0	0	1	1	0	0
0	1	x	x	0	1
1	0	x	x	1	0
1	1	x	x	x	x

Comparador 1bit resultado de L_i

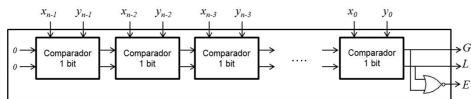


G_{i+1}	L_{i+1}	x_i	y_i	G_i	L_i
0	0	0	0	0	0
0	0	0	1	0	1
0	0	1	0	1	0
0	0	1	1	0	0
0	1	x	x	0	1
1	0	x	x	1	0
1	1	x	x	x	x

Comparador 1bit puertas lógicas

$$G_t = \bar{L}_{t+1} \cdot x_t \cdot \bar{y}_t + G_{t+1}$$

$$L_t = \bar{G}_{t+1} \cdot \bar{x}_t \cdot y_t + L_{t+1}$$



Índice

Recordando

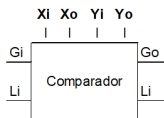
ejemplo comparador 1bit

Mapas de karnaugh

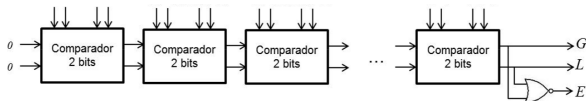
Tiempos de propagación

Multiplexores / demultiplexores

Ejemplo comparador 2bit



Gi	Li	Xi	Xo	Yi	Yo	Go	Lo
0	0	0	0	0	0	0	0
0	0	0	0	1	x	0	1
0	0	0	0	x	1	0	1
0	0	0	1	0	0	1	0
0	0	0	1	0	1	0	0
0	0	0	1	1	x	0	1
0	0	1	0	0	x	1	0
0	0	1	0	1	0	0	0
0	0	1	0	1	1	0	1
0	0	1	1	0	x	1	0
0	0	1	1	x	0	1	0
0	0	1	1	1	1	0	0
0	1	x	x	x	x	0	1
1	0	x	x	x	x	1	0
1	1	x	x	x	x	x	x



ejemplo comparador 2bit

Gi	Li	Xi	Xo	Yi	Yo	Go	Lo
0	0	0	0	0	0	0	0
0	0	0	0	1	x	0	1
0	0	0	0	x	1	0	1
0	0	0	1	0	0	1	0
0	0	0	1	0	1	0	0
0	0	0	1	1	x	0	1
0	0	1	0	0	x	1	0
0	0	1	0	1	0	0	0
0	0	1	0	1	1	0	1
0	0	1	1	0	x	1	0
0	0	1	1	x	0	1	0
0	0	1	1	1	1	0	0
0	1	x	x	x	x	0	1
1	0	x	x	x	x	1	0
1	1	x	x	x	x	x	x

Go		Xo	0	0	0	0	1	1	1	1
		Yi	0	0	1	1	1	1	0	0
		Yo	0	1	1	0	0	1	1	0
Gi	Li	Xi								
0	0	0								
0	0	1								
0	1	1								
0	1	0								
1	1	0								
1	1	1								
1	0	1								
1	0	0								

ejemplo comparador 2bit

Gi	Li	Xi	Xo	Yi	Yo	Go	Lo
0	0	0	0	0	0	0	0
0	0	0	0	1	x	0	1
0	0	0	0	x	1	0	1
0	0	0	1	0	0	1	0
0	0	0	1	0	1	0	0
0	0	0	1	1	x	0	1
0	0	1	0	0	x	1	0
0	0	1	0	1	0	0	0
0	0	1	0	1	1	0	1
0	0	1	1	0	x	1	0
0	0	1	1	1	1	0	0
0	1	x	x	x	x	0	1
1	0	x	x	x	x	1	0
1	1	x	x	x	x	x	x

L			Xo	0	0	0	0	1	1	1	1
			Yi	0	0	1	1	1	1	0	0
			Yo	0	1	1	0	0	1	1	0
			Gi	Li	Xi						
0	0	0									
0	0	1									
0	1	1									
0	1	0									
1	1	0									
1	1	1									
1	0	1									
1	0	0									

ejemplo comparador 2bit

G			Xo	0	0	0	0	1	1	1	1
			Yi	0	0	1	1	1	1	0	0
			Yo	0	1	1	0	0	1	1	0
Gi	Li	Xi									1
0	0	0									
0	0	1	1	1				1	1	1	
0	1	1									
0	1	0									
1	1	0	x	x	x	x	x	x	x	x	x
1	1	1	x	x	x	x	x	x	x	x	x
1	0	1	1	1	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	1	1	1	1

G			Xo	0	0	0	0	1	1	1	1
			Yi	0	0	1	1	1	1	0	0
			Yo	0	1	1	0	0	1	1	0
Gi	Li	Xi									1
0	0	0									
0	0	1	1	1				1	1	1	
0	1	1									
0	1	0									
1	1	0	x	x	x	x	x	x	x	x	x
1	1	1	x	x	x	x	x	x	x	x	x
1	0	1	1	1	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	1	1	1	1

G			Xo	0	0	0	0	1	1	1	1
			Yi	0	0	1	1	1	1	0	0
			Yo	0	1	1	0	0	1	1	0
Gi	Li	Xi									1
0	0	0									
0	0	1	1	1				1	1	1	
0	1	1									
0	1	0									
1	1	0	x	x	x	x	x	x	x	x	x
1	1	1	x	x	x	x	x	x	x	x	x
1	0	1	1	1	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	1	1	1	1

G			Xo	0	0	0	0	1	1	1	1
			Yi	0	0	1	1	1	1	0	0
			Yo	0	1	1	0	0	1	1	0
Gi	Li	Xi									1
0	0	0									
0	0	1	1	1				1	1	1	
0	1	1									
0	1	0									
1	1	0	x	x	x	x	x	x	x	x	x
1	1	1	x	x	x	x	x	x	x	x	x
1	0	1	1	1	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	1	1	1	1

Ejemplo comparador 2bit

$$G_o =$$

Ejemplo comparador 2bit

L			Xo	0	0	0	0	1	1	1	1
			Yi	0	0	1	1	1	1	0	0
			Yo	0	1	1	0	0	1	1	0
Gi	Li	Xi									
0	0	0		1	1	1	1	1	1		
0	0	1			1						
0	1	1	1	1	1	1	1	1	1	1	1
0	1	0	1	1	1	1	1	1	1	1	1
1	1	0	x	x	x	x	x	x	x	x	x
1	1	1	x	x	x	x	x	x	x	x	x
1	0	1									
1	0	0									

L			Xo	0	0	0	0	1	1	1	1
			Yi	0	0	1	1	1	1	0	0
			Yo	0	1	1	0	0	1	1	0
Gi	Li	Xi									
0	0	0		1	1	1	1	1	1		
0	0	1			1						
0	1	1	1	1	1	1	1	1	1	1	1
0	1	0	1	1	1	1	1	1	1	1	1
1	1	0	x	x	x	x	x	x	x	x	x
1	1	1	x	x	x	x	x	x	x	x	x
1	0	1									
1	0	0									

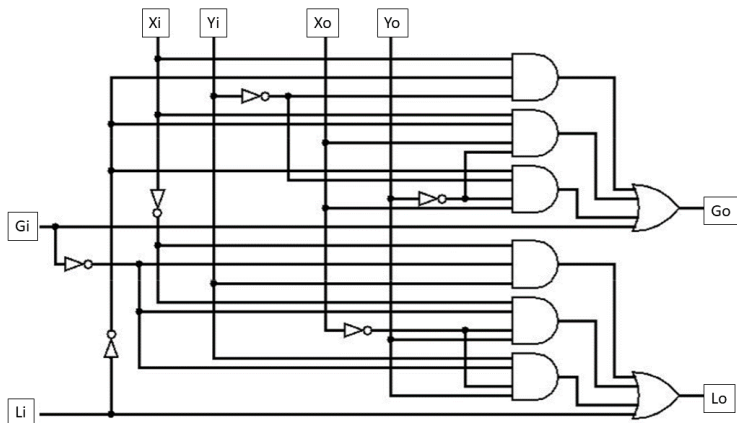
L			Xo	0	0	0	0	1	1	1	1
			Yi	0	0	1	1	1	1	0	0
			Yo	0	1	1	0	0	1	1	0
Gi	Li	Xi									
0	0	0		1	1	1	1	1	1		
0	0	1			1						
0	1	1	1	1	1	1	1	1	1	1	1
0	1	0	1	1	1	1	1	1	1	1	1
1	1	0	x	x	x	x	x	x	x	x	x
1	1	1	x	x	x	x	x	x	x	x	x
1	0	1									
1	0	0									

L			Xo	0	0	0	0	1	1	1	1
			Yi	0	0	1	1	1	1	0	0
			Yo	0	1	1	0	0	1	1	0
Gi	Li	Xi									
0	0	0		1	1	1	1	1	1		
0	0	1			1						
0	1	1	1	1	1	1	1	1	1	1	1
0	1	0	1	1	1	1	1	1	1	1	1
1	1	0	x	x	x	x	x	x	x	x	x
1	1	1	x	x	x	x	x	x	x	x	x
1	0	1									
1	0	0									

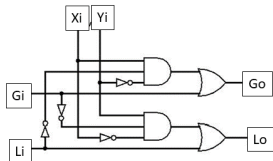
Ejemplo comparador 2bit

$$L_o =$$

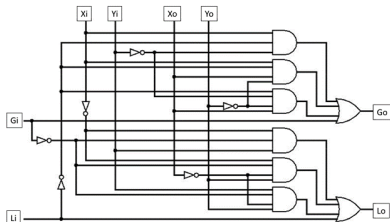
Resultado



Comparativa de puertas



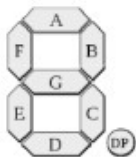
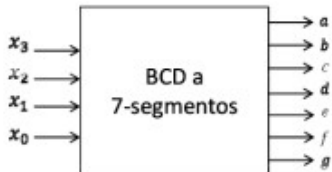
Comparador de 1 bit



Comparador de 2 bit

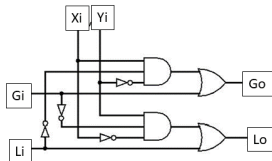
Número de puertas por cada implementación

mapas K - Ejemplo BCD2SSEG

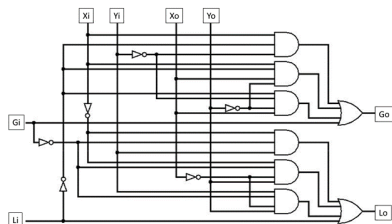


x_3	x_2	x_1	x_0	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	0	0	1	1
1	0	1	0							
1	0	1	1							
1	1	0	0							
1	1	0	1							
1	1	1	0							
1	1	1	1							

comparativa de tiempos



Comparador de 1 bit



Comparador de 2 bit

Tiempo de propagación en cada implementación ?

Índice

Recordando

ejemplo comparador 1bit

Mapas de karnaugh

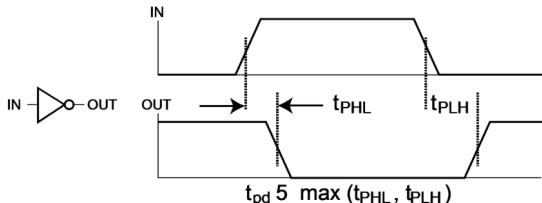
Tiempos de propagación

Multiplexores / demultiplexores

Tiempos de propagación

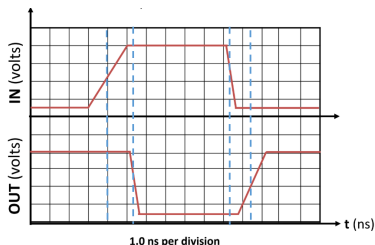
El tiempo de propagación es el tiempo que tarda un cambio en una entrada de una puerta para verse reflejado a la salida.

1. El retraso generalmente se mide al 50% con respecto a los niveles de voltaje de salida H y L.
2. La señal de salida de alto a bajo (t_{PHL}) y de bajo a alto (t_{PLH}). los cambios pueden tener diferentes retrasos de propagación.
3. Los cambio de alto a bajo (HL) y de bajo a alto (LH) son definido con respecto a la salida, no a la entrada.



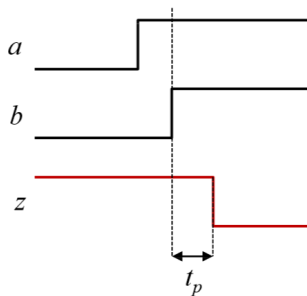
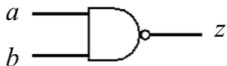
Ejercicio: Tiempos de propagación

Cuál es el tiempo de propagación de 4 inversores idénticos según la gráfica ?

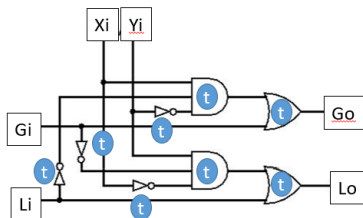


Tiempos de propagación

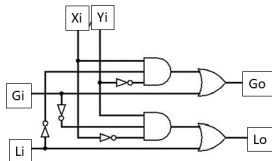
1. Toda puerta lógica tiene un tiempo de retraso en la salida respecto a la entrada



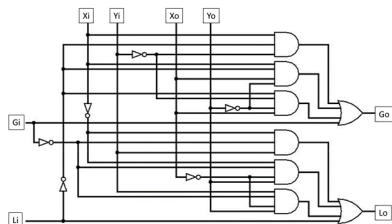
Tiempos de propagación



comparativa de tiempos



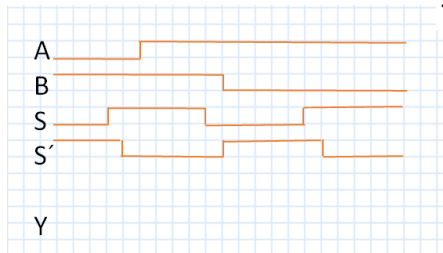
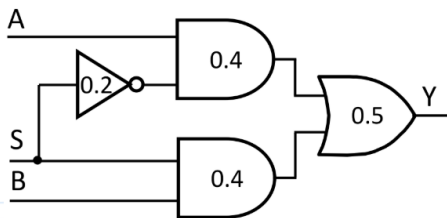
Comparador de 1 bit



Comparador de 2 bit

Tiempo de propagación en cada implementación ?

Ejercicio



Índice

Recordando

ejemplo comparador 1bit

Mapas de karnaugh

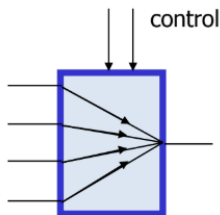
Tiempos de propagación

Multiplexores / demultiplexores

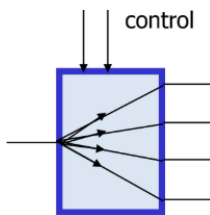
descripción

Son una conexiones directas punto a punto entre puertas

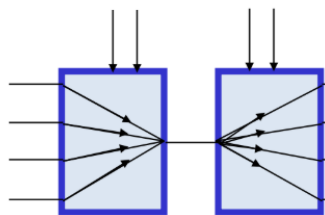
- ▶ **multiplexor** Enrutar una de muchas entradas a una sola salida
- ▶ **demultiplexor** Enrutar una sola entrada a una de las muchas salidas



multiplexer



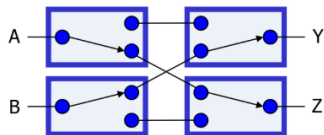
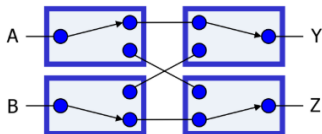
demultiplexer



4x4 switch

conmutación

Son una conexiones directas punto a punto entre puertas
Se puede usar para hacer redes de conmutación de tamaño arbitrario: se usa para implementar interconexión de múltiples fuentes / múltiples destinos



concepto general

- ▶ 2^n entradas de datos, n entradas de control (llamadas "selección"), 1 salida
- ▶ Se usa para conectar 2^n puntos a un solo punto
- ▶ El patrón de señal de control forma un índice binario de entrada conectada a la salida

$$Z = A' I_0 + A I_1$$

A	Z
0	I_0
1	I_1

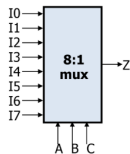
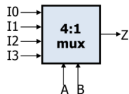
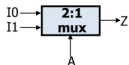
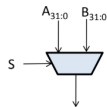
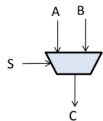
functional form

logical form

two alternative forms
for a 2:1 Mux truth table

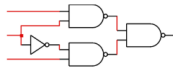
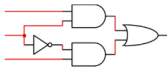
I_1	I_0	A	Z
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

concepto general

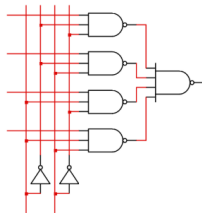
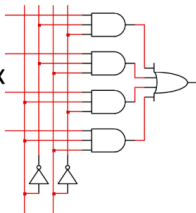


implementación con puertas

- 2:1 mux



- 4:1 mux



Tablas de verdad

A	B	C	F
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

$$\begin{aligned}F(A,B,C) &= m_0 + m_2 + m_6 + m_7 \\ &= A'B'C' + A'BC' + ABC' + ABC \\ &= A'B'(C') + A'B(C') + AB'(0) + \\ &\quad AB(1)\end{aligned}$$

1 —
0 —
1 —
0 —
0 —
0 —
1 —
1 —

Tablas de verdad

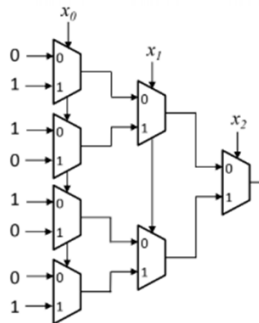
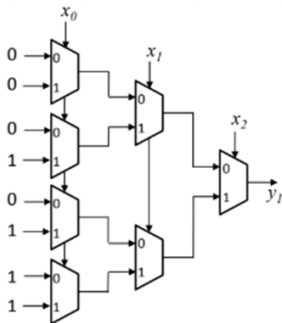
Toda función booleana puede implementarse con multiplexores 2-a-1 de 1 bit utilizando reiteradamente la siguiente regla (Ley de Shannon)

$$f(x_0, x_1, \dots, x_n) = \overline{x_0} * f(0, x_1, \dots, x_n) + x_0 * f(1, x_1, \dots, x_n)$$

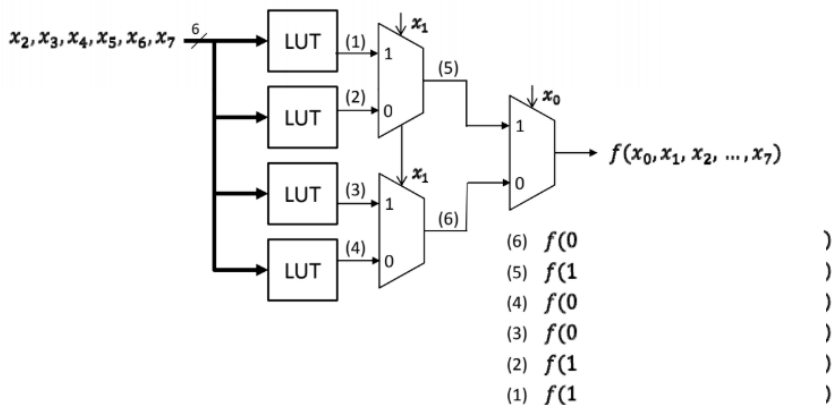
.

Tablas de verdad

x2	x1	x0	y1	y0
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

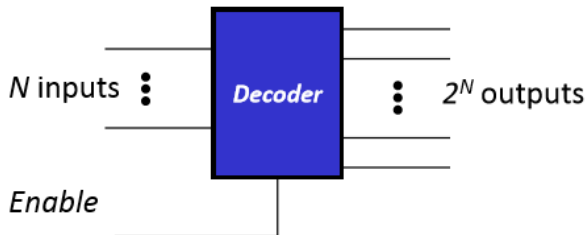


look Up Table

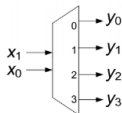


Decodificador/demultiplexor

- ▶ n entradas
- ▶ $m = 2^n$ salidas



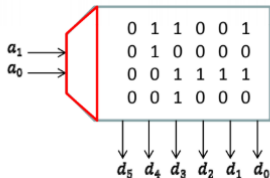
Decodificador/demultiplexor



$x_1 x_0$	$y_0 y_1 y_2 y_3$
0 0	1 0 0 0
0 1	0 1 0 0
1 0	0 0 1 0
1 1	0 0 0 1

$$y_0 = \bar{x}_1 \cdot \bar{x}_0 = m_0; \quad y_1 = \bar{x}_1 \cdot x_0 = m_1$$

$$y_2 = x_1 \cdot \bar{x}_0 = m_2; \quad y_3 = x_1 \cdot x_0 = m_3$$

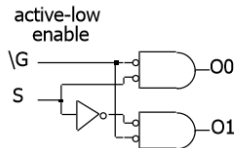
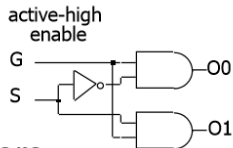


0	1	1	0	0	1
0	1	0	0	0	0
0	0	1	1	1	1
0	0	1	0	0	0

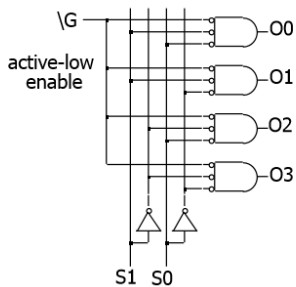
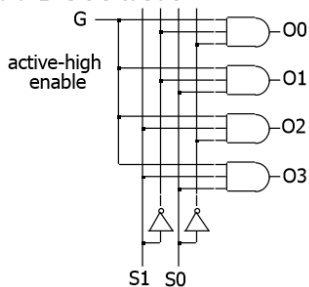
d_5 d_4 d_3 d_2 d_1 d_0

Decodificador con puertas lógicas

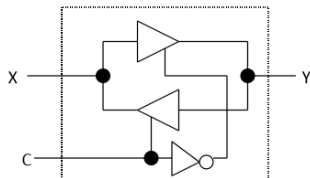
1:2 Decoders



2:4 Decoders



Bidireccional



Bi-direction circuit

PREGUNTAS