

# Electrónica Digital

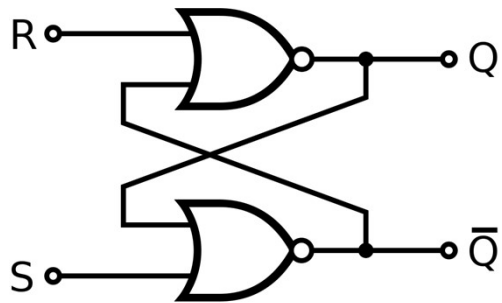
## Clase 15

Flip Flops y Máquinas de Estados Finitos

## 2 Latch Set-Reset (SR)

- **Latch:** es un tipo de dispositivo lógico **biestable asíncrono**.
  - **Biestable:** Que tienen **dos estados estables** (HIGH – LOW) (“0” – “1”)
  - El **paso de un estado** a otro se realiza **variando** sus **entradas**.
  - Cambian su salida con un nivel de voltaje (**level sensitive**) en sus entradas sin depender de un **clock (asíncrono)**.
- Los **biestables** pueden ser síncronos o asíncronos dependiendo de sus entradas:
  - **Asíncrono:** Solamente tienen entradas de control.
  - **Síncrono:** Posee además de las entradas de control, un **clock**.
- Un **latch S-R** con entrada activa en alto se compone de 2 compuertas NOR o 2 compuertas NAND acopladas como se muestra en la figura.

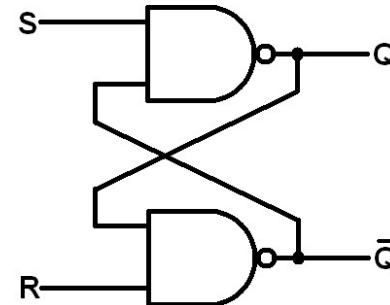
*Circuito lógico*



*Tabla de Verdad*

<i>S</i>	<i>R</i>	<i>Q</i>
0	0	<i>No cambia</i>
0	1	0
1	0	1
1	1	<i>No válido</i>

*Circuito lógico*

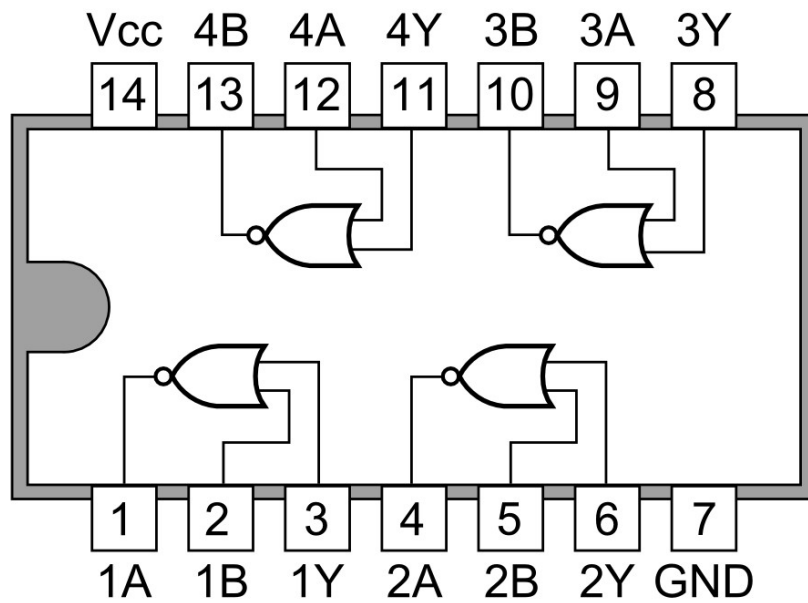


*Tabla de Verdad*

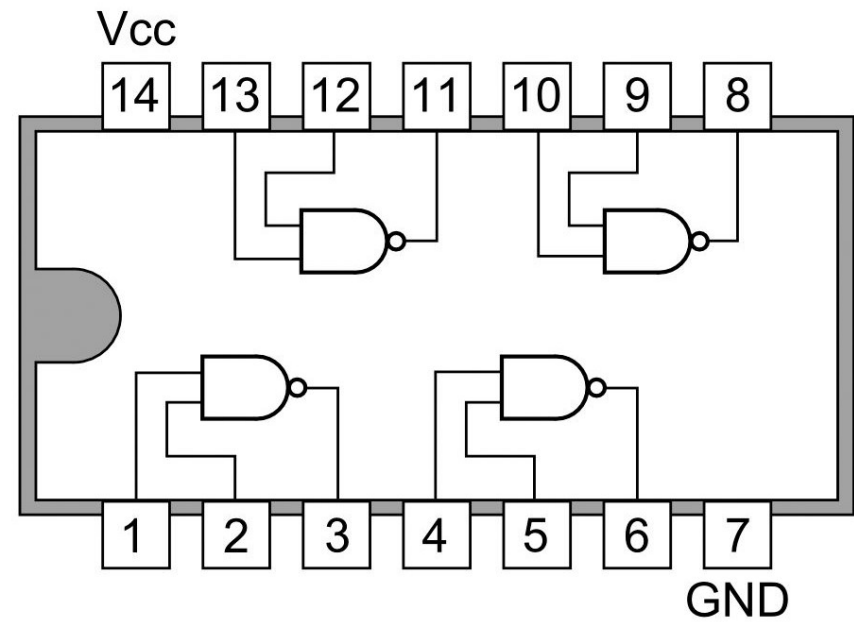
<i>S</i>	<i>R</i>	<i>Q</i>
0	0	<i>No cambia</i>
0	1	1
1	0	0
1	1	<i>No válido</i>

# 3 Compuerta NOR (74LS02) y NAND (74LS00)

*NOR – 74LS02*

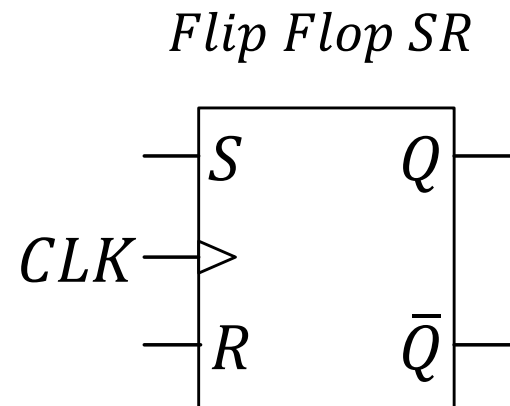
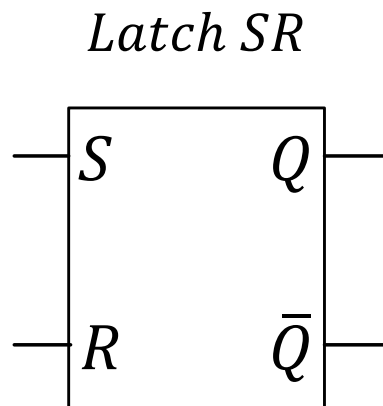


*NAND – 74LS00*



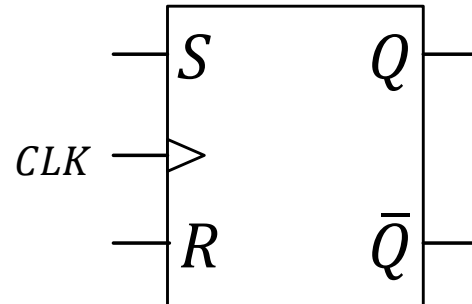
# 4 Flip-Flops disparados por flanco

- Los flip-flops son dispositivos **síncronos** de **dos estados**.
- También conocidos como **multivibradores biestables (circuito capaz de generar una onda cuadrada)**.
- Se crearon para eliminar las deficiencias de los latches:
  - Estos cambian con un flanco en el reloj (**edge sensitive**), no con un nivel de voltaje (**level sensitive**) en sus entradas:
  - Flanco de subida (cambio de voltaje bajo a alto) ó flanco de bajada (cambio de voltaje alto a bajo)
- El termino **síncrono** significa que la salida cambia de estado únicamente en un instante específico de una entrada de disparo denominada (**CLK**).
- Los cambios en la salida se producen **sincrónicamente** con el **reloj (CLK)**. Este produce lo que llamamos un **refresh**.



# Flip-Flop S-R

*Símbolo*



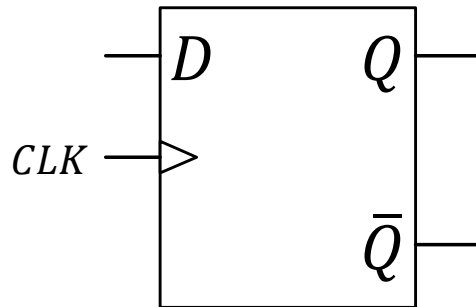
*Tabla de Verdad*

<i><b>S</b></i>	<i><b>R</b></i>	<i><b>CLK</b></i>	<i><b>Q</b></i>	<i><b><math>\bar{Q}</math></b></i>	
0	0	<i>X</i>	<i><b>Q<sub>0</sub></b></i>	<i><b>Q<sub>0</sub></b></i>	<i>No cambio</i>
0	1	↑	0	1	<i>RESET</i>
1	0	↑	1	0	<i>SET</i>
1	1	↑	?	?	<i>No válido</i>

# 6 Flip-Flop tipo D

- Útil cuando se necesita almacenar un único bit de datos (1 o 0).
- Si se añade un inversor (NOT) a un flip-flop S-R obtenemos un flip-flop D básico.

*Símbolo*



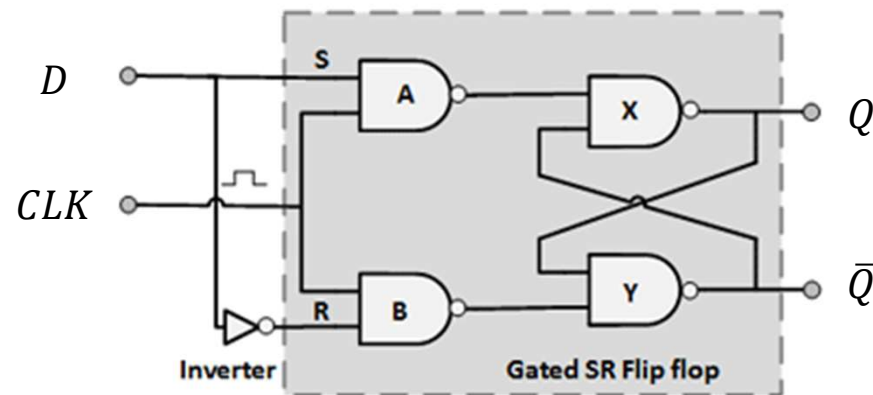
*Ecuación*

$$Q_{siguiente} = D$$

*Tabla de verdad*

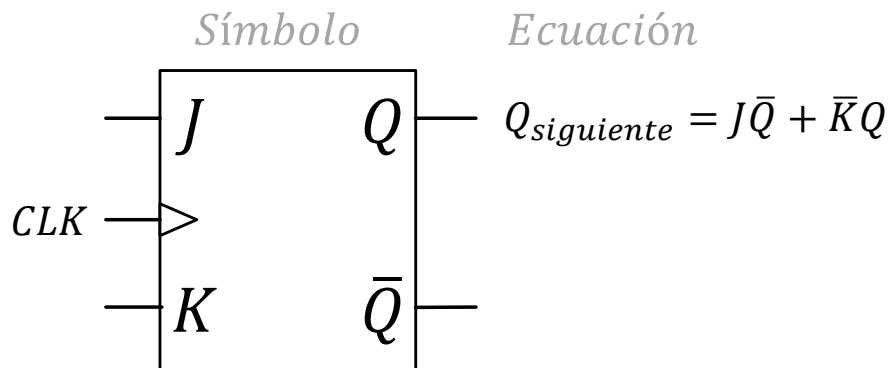
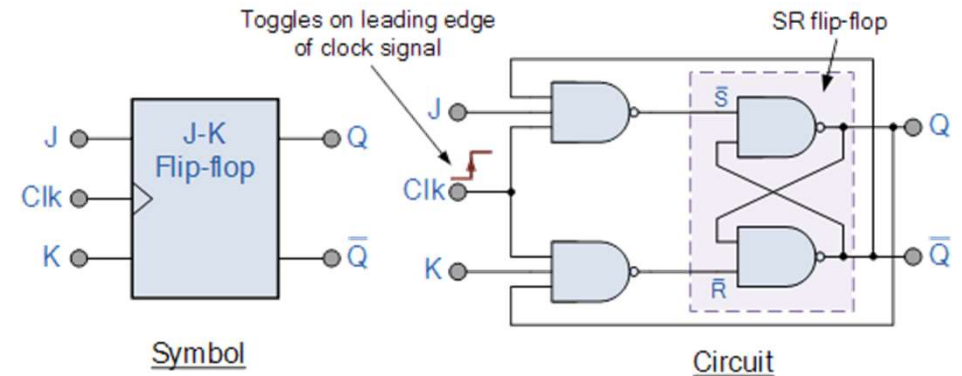
<i>D</i>	<i>CLK</i>	<i>Q</i>	$\bar{Q}$	
1	↑	1	0	<i>SET</i>
0	↑	0	1	<i>RESET</i>

*Circuito Lógico*



# 7 Flip-Flop J-K

- El flip flop mas utilizado.
- Igual al S-R pero sin condiciones no válidas.
- Entradas de control:
  - J: Grabado o SET.
  - K: Borrado o RESET.



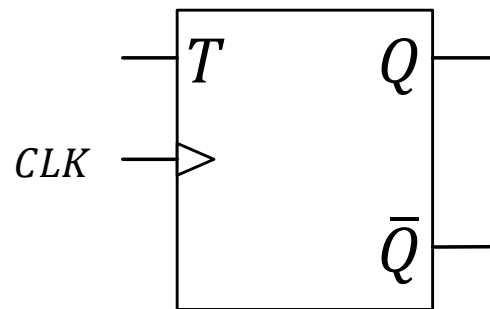
*Tabla de Verdad*

<i>J</i>	<i>K</i>	<i>CLK</i>	<i>Q</i>	<i>Q̄</i>	
0	0	↑	$Q_0$	$\bar{Q}_0$	No cambio
0	1	↑	0	1	RESET
1	0	↑	1	0	SET
1	1	↑	$\bar{Q}_0$	$Q_0$	Basculación ó Inversión

# 8 Flip-Flop tipo T

- Tipo Toggle (T).
- Cambia de estado ("toggle" en inglés) cada vez que la entrada de **CLK** se dispara mientras la entrada T está a nivel alto.
- Si la entrada T está a nivel bajo, el biestable retiene el nivel previo.

*Símbolo*



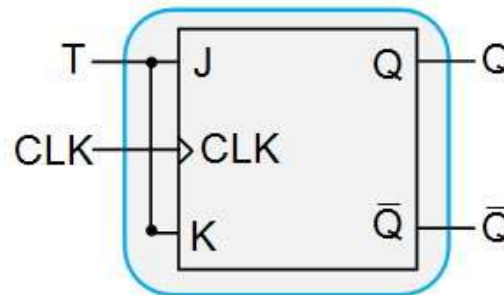
*Ecuación*

$$Q_{siguiente} = T \oplus Q$$

*Tabla de verdad*

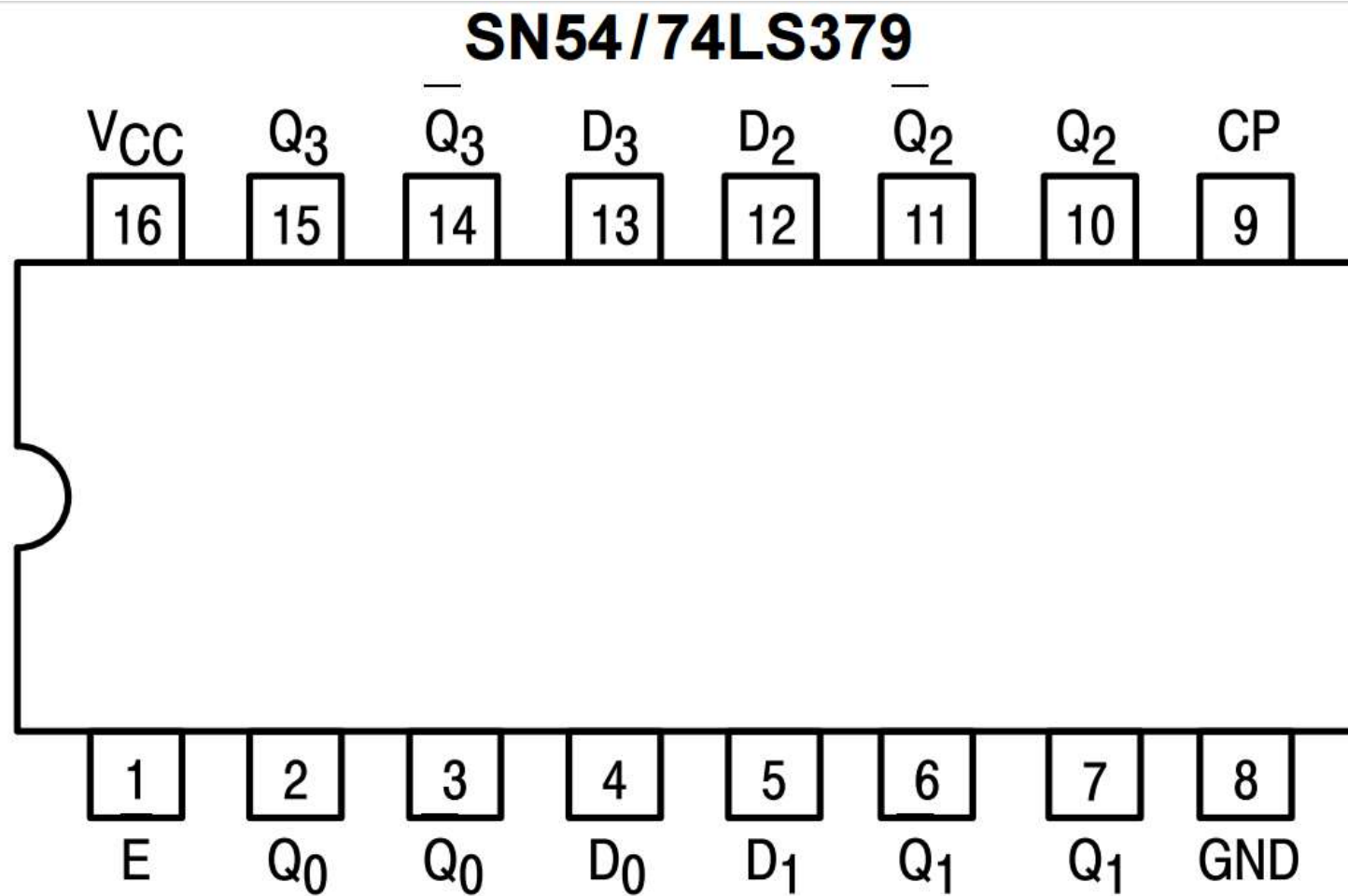
$T$	$Q$	$\bar{Q}$
0	$Q_0$	$\bar{Q}_0$
1	$\bar{Q}_0$	$Q_0$

*Circuito Lógico*

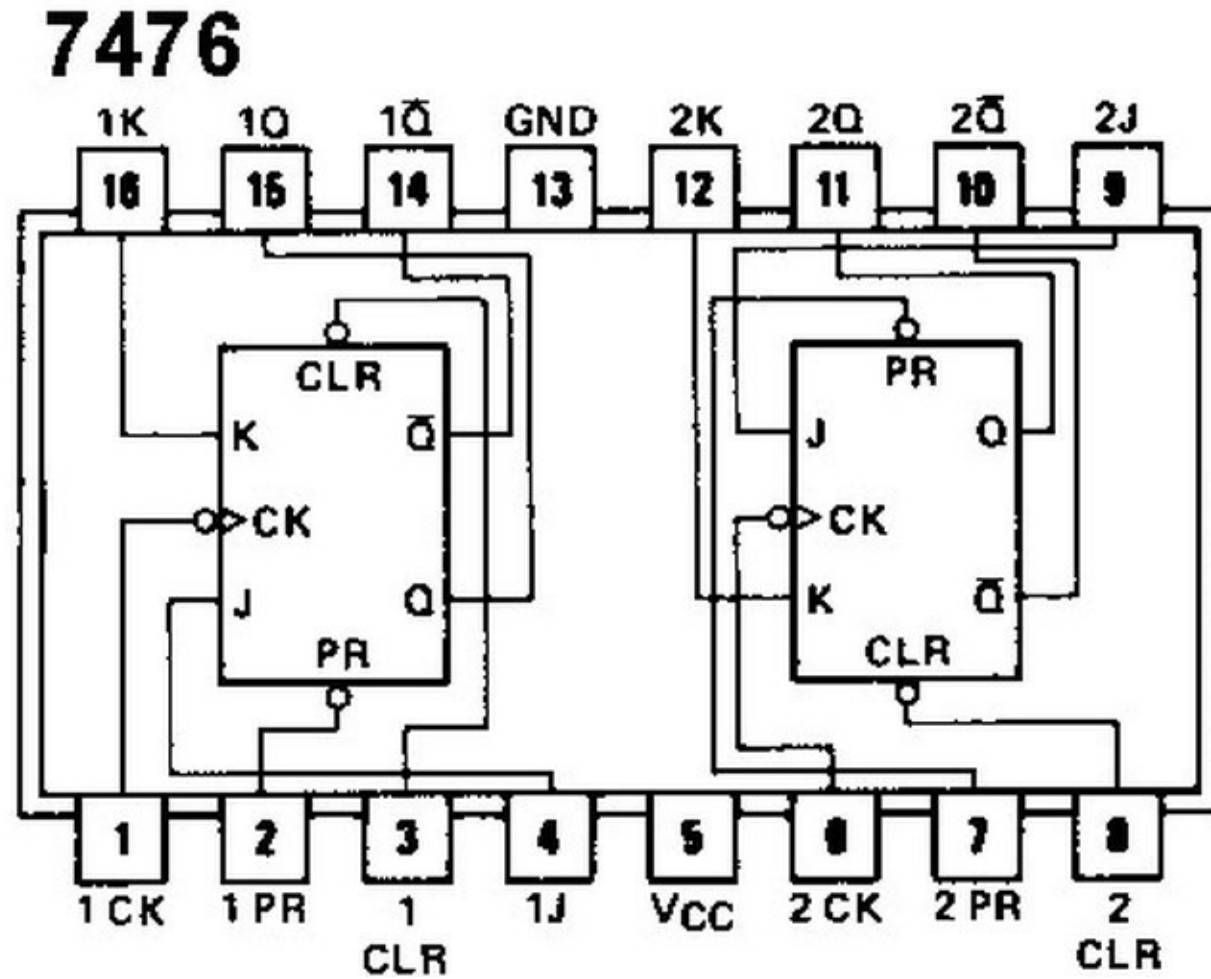




# Flip-Flop D (74LS379) Pinout

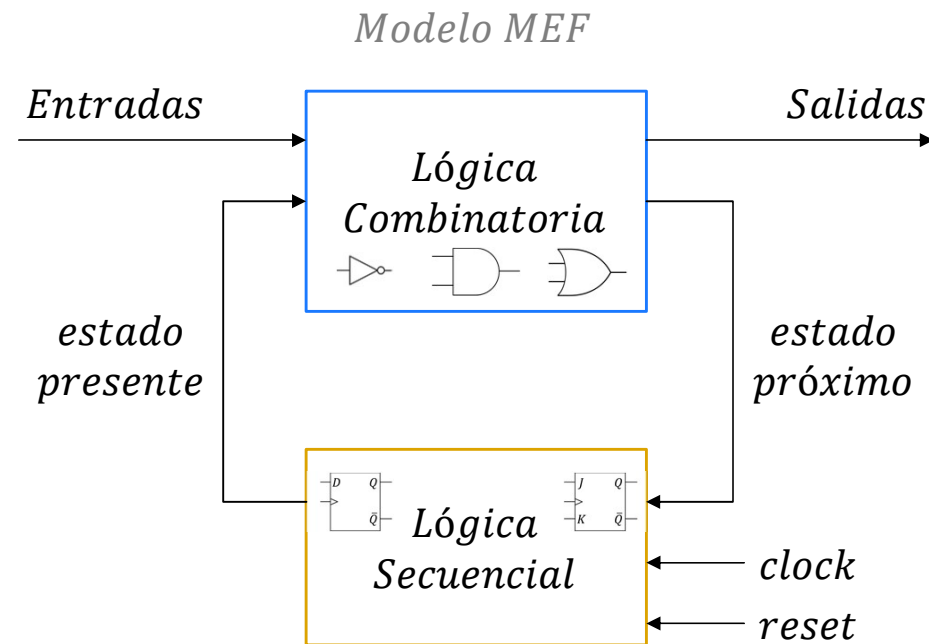


# Flip-Flop J-K (74LS76) Pinout



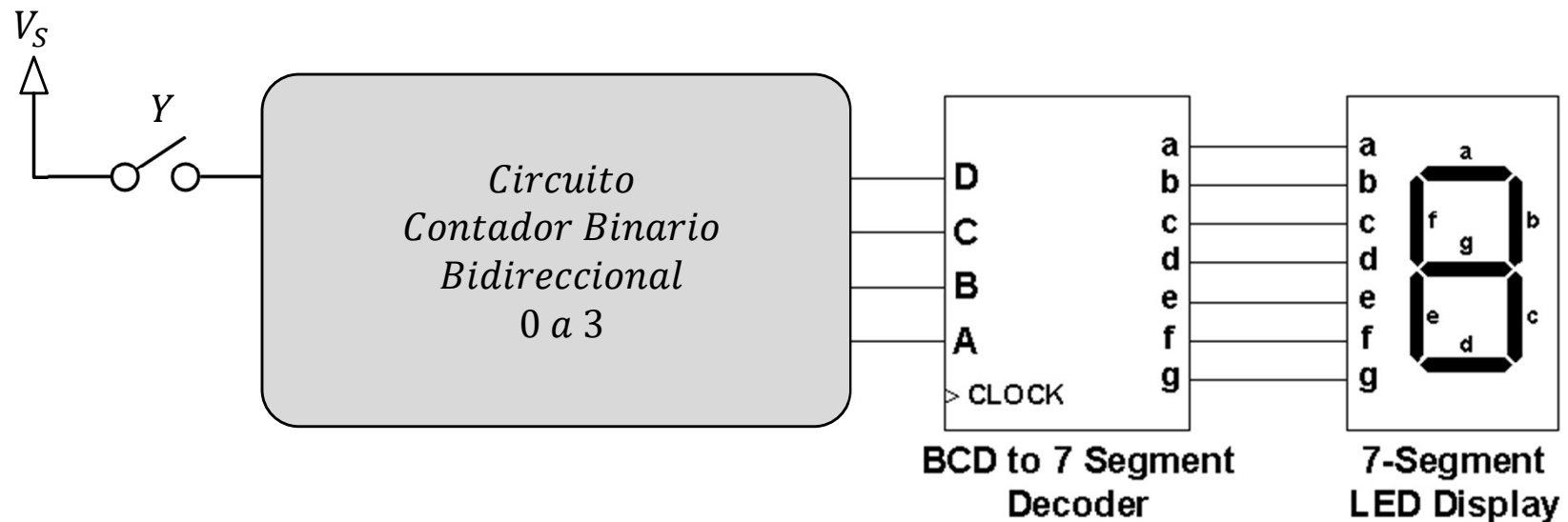
# Circuito Secuencial con MEF

- ▶ Los circuitos secuenciales son circuitos cuya función de salida depende del estado de las entradas y del estado anterior.
- ▶ Se solucionan usando compuertas lógicas y flip – flops.
- ▶ Los cambios de estado del circuito se controlan por medio de una señal de reloj.
- ▶ **Estado Presente:** Es la información actualmente almacenada en los Flip Flops.
- ▶ **Estado Próximo:** Es la información que los Flip Flops deben almacenar en un flanco del clock futura.



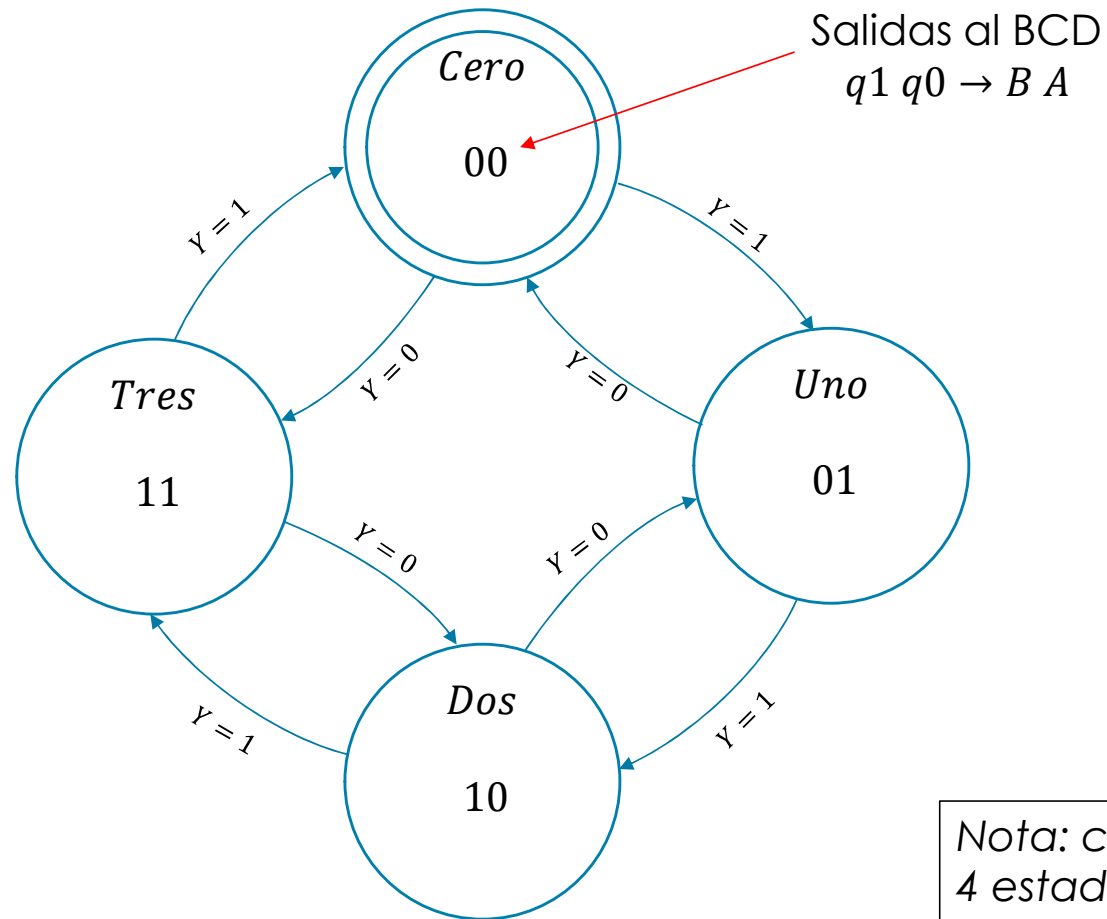
# Ejemplo Circuitos Secuenciales - MEF

- ▶ Diseñar un contador binario de 0 a 3 de tal manera que el usuario seleccione mediante un suiche si el contador funciona en forma ascendente (0 a 3) o descendente (3 a 0).
- ▶ Con el suiche en uno, funciona en forma ascendente y en cero, el contador funciona en forma descendente.



## Ej – Paso 1: Realizar diagrama de estados

- Este diagrama muestra la progresión de los estados por los que el contador avanza o retrocede cuando se aplica una señal de reloj.



Nota: como se requieren  
4 estados, es necesario  
 $\log_2 \text{estados} = \log_2 4 = 2 \text{ flip flops}$

## Ej – Paso 2: Realizar tabla de transiciones

- Esta tabla enumera cada estado del contador (Estado actual) junto con el correspondiente estado siguiente.
- El estado siguiente es el estado al que el contador pasa desde su estado actual, al aplicar un pulso de reloj.
- Se recomienda tener a la mano la tabla de excitación del Flip Flop con el que se va a implementar la lógica secuencial

ENTRADAS			SALIDAS							
Estado Presente			Y	Estado Próximo			J1	K1	J0	K0
Nombre	q1	q0		q1	q0	Nombre				
Cero	0	0	0	1	1	Tres	1	x	1	x
			1	0	1	Uno	0	x	1	x
Uno	0	1	0	0	0	Cero	0	x	x	1
			1	1	0	Dos	1	x	x	1
Dos	1	0	0	0	1	Uno	x	1	1	x
			1	1	1	Tres	x	0	1	x
Tres	1	1	0	1	0	Dos	x	0	x	1
			1	0	0	Cero	x	1	x	1

Tabla de excitación JK

Transiciones de Salida		Entradas del flip-flop J/K	
Q <sub>N</sub>	Q <sub>N+1</sub>	J	K
0	→ 0	0	X
0	→ 1	1	X
1	→ 0	X	1
1	→ 1	X	0

# Ej – Paso 3: Realizar los Mapas de Karnaugh

- Los mapas de Karnaugh se usan para determinar la lógica requerida para las entradas J y K de cada flip-flop del contador.
- Se debe utilizar un mapa de Karnaugh para la entrada J y otro para la entrada K de cada flip-flop.

*Para J0*

	$Q_1 \backslash Q_0$	0	1
00	1	1	
01	X	X	
11	X	X	
10	1	1	

$J0 = 1$

*Para K0*

$Q_1 \backslash Q_0$	0	1
00	X	X
01	1	1
11	1	1
10	X	X

$K0 = 1$

*Para J1*

	$Q_1 \backslash Q_0$	0	1
00	1	0	
01	0	1	
11	X	X	
10	X	X	

$q0 \cdot Y$

$\overline{q0} \cdot \overline{Y}$

$J1 = q0 \cdot Y + \overline{q0} \cdot \overline{Y}$

iguales

$$K1 = q0 \cdot Y + \overline{q0} \cdot \overline{Y}$$

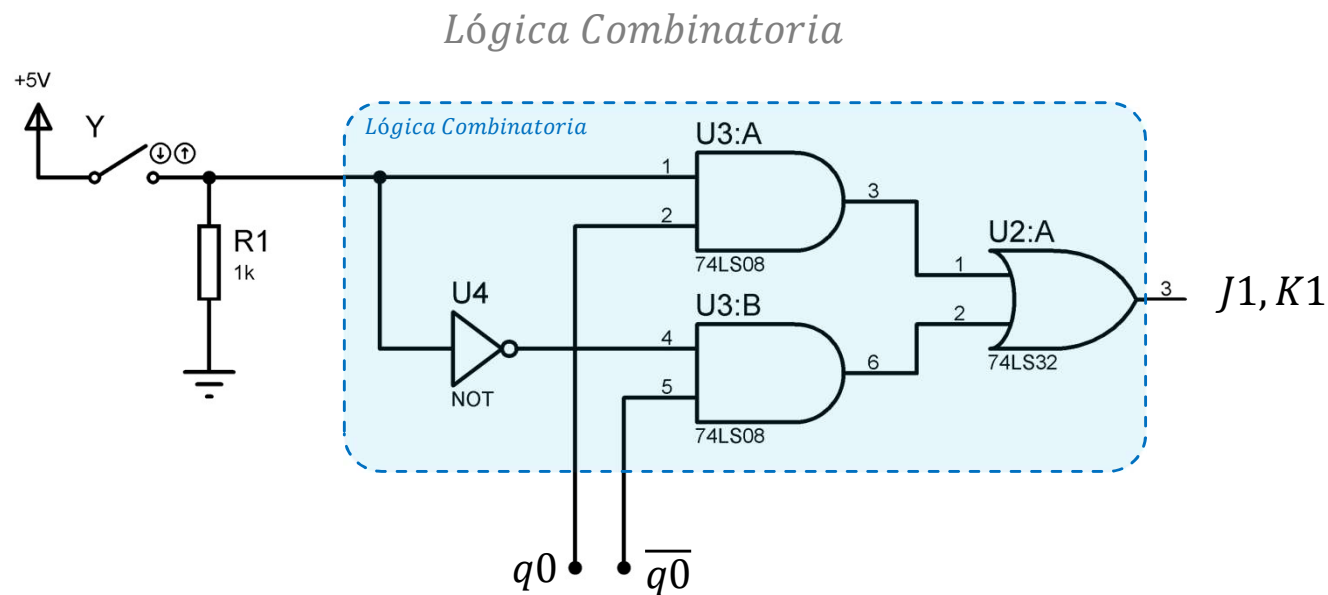
	$Q_1 \backslash Q_0$	0	1
00	X	X	
01	X	X	
11	0	1	
10	1	0	

$\overline{Q_1} \cdot \overline{Q_0}$  (points to 00 row)  
 $Q_1 \cdot \overline{Q_0}$  (points to 01 row)  
 $Q_1 \cdot Q_0$  (points to 11 row)  
 $\overline{Q_1} \cdot Q_0$  (points to 10 row)

$q0 \cdot Y$  (points to 01 row, 1 column)  
 $\overline{q0} \cdot \overline{Y}$  (points to 10 row, 0 column)

## Ej – Paso 3: Realizar los Mapas de Karnaugh

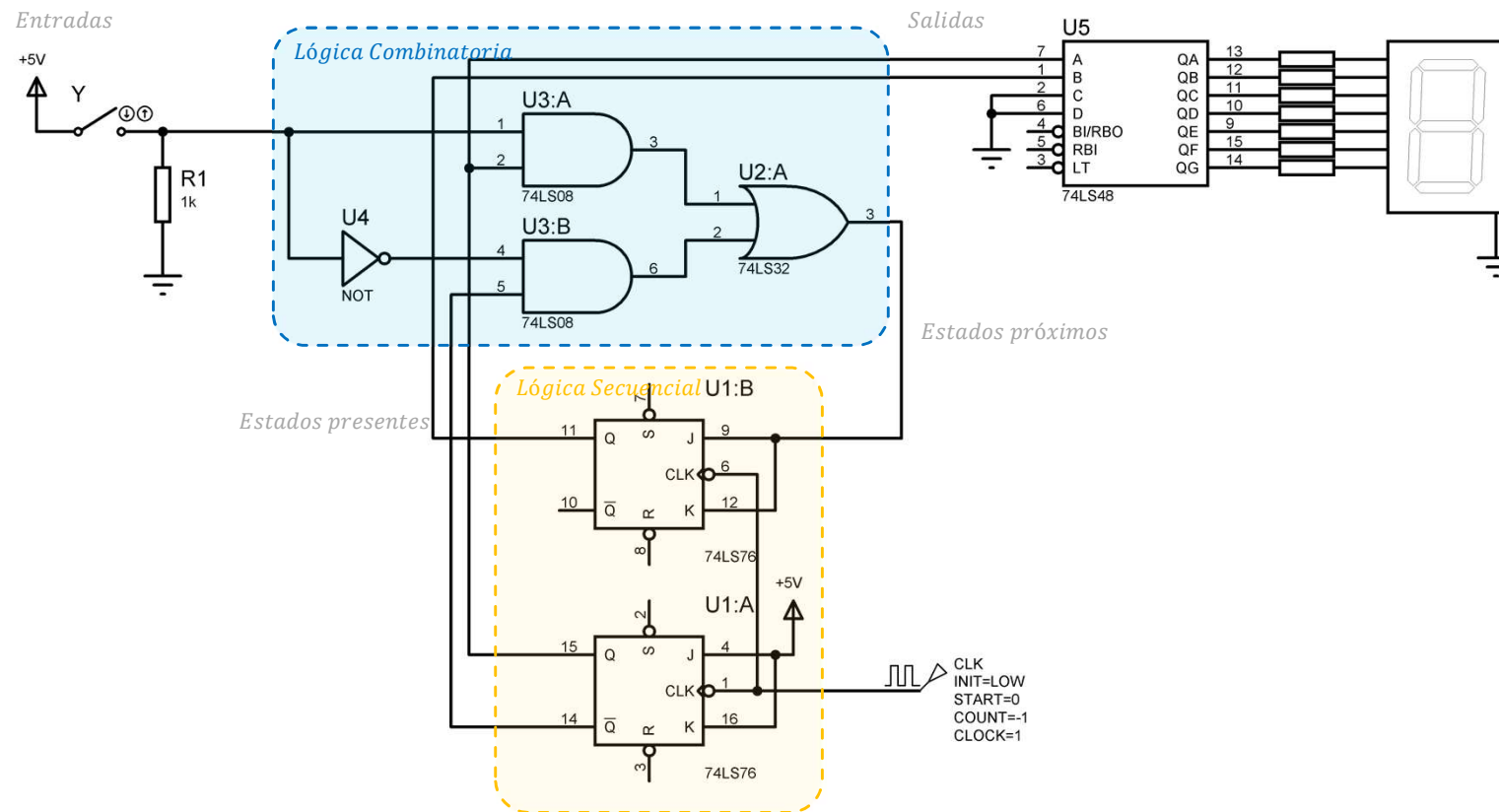
- Después de la simplificación de las expresiones en los mapas de Karnaugh se obtienen las expresiones lógicas para las entradas a los flip-flops.
  - $J0 = K0 = 1$  (Conexión directa a +5V).
  - $J1 = K1 = q0 \cdot Y + \overline{q0} \cdot \overline{Y}$





## Ej – Paso 4: Implementar MEF

- El paso final consiste en implementar la lógica combinacional y la lógica secuencial a partir de las expresiones J y K, y conectar los flip-flops para conseguir el contador ascendente descendente de 2 bits.



MUCHAS GRACIAS