

# 6. SISTEMAS SECUENCIALES

## 6. SISTEMAS SECUENCIALES

### 6.1 Circuitos biestables.

- Definición de sistema secuencial. Tipos y características: Asíncronos y síncronos.
- Biestables: R S, J K, T y D
- Tiempos característicos en biestables.

### 6.2 Registros de desplazamiento.

- Concepto de registro.
- Registros de desplazamiento. Entrada serie, salida serie. Entrada serie, salida paralelo. Entrada paralelo, salida serie. Entrada paralelo, salida paralelo. Bidireccional.
- Aplicaciones de los registros. Generador de secuencia.

### 6.3 Contadores.

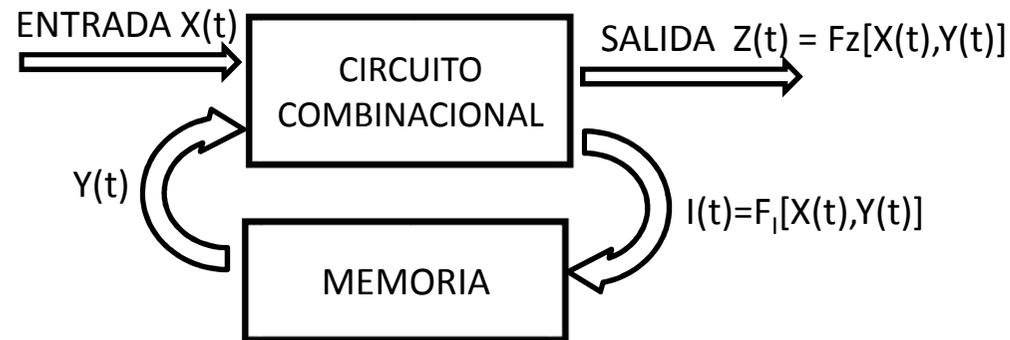
- Contadores digitales y sus aplicaciones.
- Contadores asíncronos. Contador de décadas.
- Contadores síncronos. Acarreo en serie y paralelo.
- Contador reversible.
- Contadores basados en registros de desplazamiento: en anillo, Johnson, con protección.

### 6.4 Análisis y diseño de circuitos secuenciales síncronos.

- Análisis de circuitos secuenciales síncronos.
- Tablas de transiciones y diagramas de estados: Máquina de Mealy y Máquina de Moore.
- Síntesis de sistemas secuenciales síncronos.

- **Definición de sistema secuencial**

Aquel sistema en que la información de salida es función de las entradas actuales y de la situación anterior del sistema (lo que se conoce como historia o memoria del sistema):



La memoria almacena la situación del sistema

- **Tipos de circuitos secuenciales**

a) Asíncronos: los cambios de las salidas se producen en los mismos instantes que se modifican las variables de entrada (salvo retardos propios de los dispositivos electrónicos)

b) Síncronos: requieren de la presencia de una señal particular, llamada sincronismo o reloj, para que las variables de entrada actúen sobre el circuito.

- **Definición de circuito biestable**

Los biestables (flip flop y latch) son circuitos lógicos elementales capaces de permanecer indefinidamente en uno de sus dos estados posibles (alto o bajo) mientras no haya orden de cambiar (lo que se conoce como disparo)

Esta capacidad de retener el estado, convierte los biestables en la célula elemental de memoria (almacena un 1 o un 0) y por tanto constituye la base para la construcción de sistemas secuenciales

- **Clasificación de los biestables**

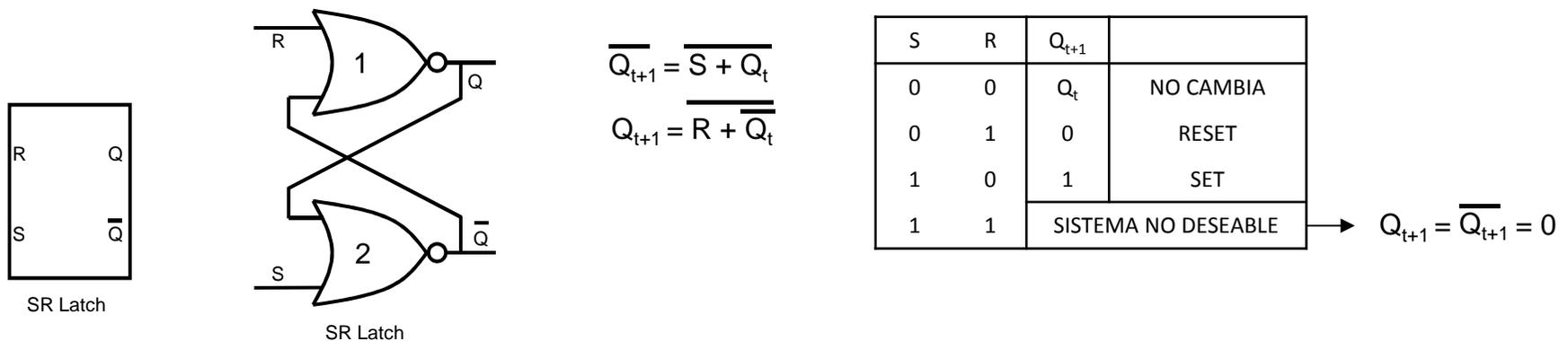
BIESTABLES	Asíncronos (latch)	R-S (*) J-K T	
		Activados por nivel (Latch Síncronos)	R-S J-K T D (*)
	Síncronos	Modo toggle activado por flanco (flip-flop)	R-S (*) J-K (*) T (se obtiene a partir del JK) D (*)
		Modo flanco (edge)	R-S J-K (*) T (se obtiene a partir del JK) D (*)

Los **asíncronos no usan reloj** (las señales de entrada son válidas en todo instante)  
 Los **síncronos sí usan reloj** (si son de tipo nivel las entradas son válidas durante el tiempo que el reloj está a valor alto y si son de flanco las entradas son válidas sólo en instantes en que la señal reloj sube o baja)

- Biastable RS**

Veremos dos versiones: RS-NOR y RS-NAND

**Biastable RS-NOR:**



R=RESET  $\Rightarrow$  R=1 y S=0  $\Rightarrow$  Q=0 y  $\bar{Q}$ =1 (el biestable almacena un 0 lógico)

S=SET  $\Rightarrow$  S=1 y R=0  $\Rightarrow$  Q=1 y  $\bar{Q}$ =0 (el biestable almacena un 1 lógico)

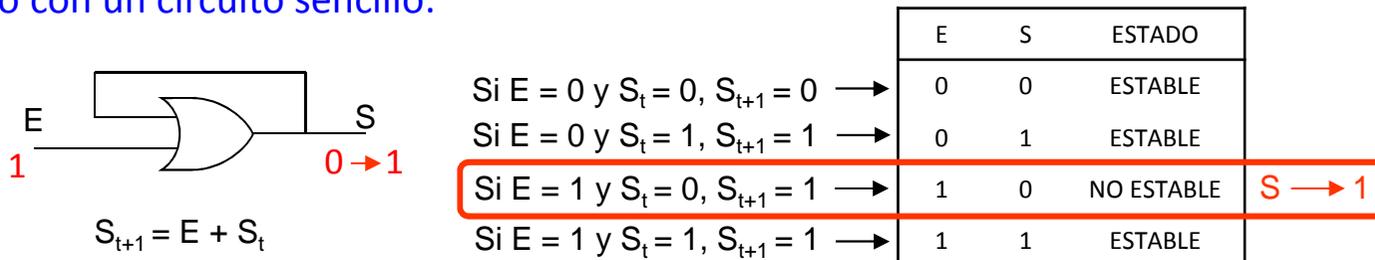
Cuando R=0 y S=0 no se manda ni SET ni RESET y el biestable mantiene su estado anterior. Si almacenaba un 0 lógico (Q=0 y  $\bar{Q}$ =1) mantendrá los valores de salida, y si almacenaba un 1 lógico (Q=1 y  $\bar{Q}$ =0), mantendrá también sus valores de salida

Normalmente Q y  $\bar{Q}$  son complementarias, pero cuando R=S=1  $\Rightarrow$  Q= $\bar{Q}$ =0 porque el biestable detecta que le llegan un reset y un set a la vez. **Este caso hay que evitarlo**

## Biestable RS-NOR

a) Estados estables: situaciones en las que un circuito permanece indefinidamente mientras no se alteren sus entradas

Ejemplo con un circuito sencillo:



Estados estables del biestable RS NOR  $\overline{Q}_{t+1} = \overline{S} + \overline{Q}_t$   $Q_{t+1} = R + \overline{Q}_t$

S	R	Q	$\overline{Q}$	ESTADO	
0	0	0	0	ESTABLE	$\overline{Q}=1 \Rightarrow Q=0$ y $S=0 \Rightarrow \overline{Q}=1$ $Q=1 \Rightarrow \overline{Q}=0$ y $R=0 \Rightarrow Q=1$
0	0	0	1		
0	0	1	0		
0	0	1	1		
0	1	0	0	ESTABLE	$R=1 \Rightarrow Q=0$ y $S=0 \Rightarrow \overline{Q}=1$
0	1	0	1		
0	1	1	0		
0	1	1	1		
1	0	0	0	ESTABLE	$S=1 \Rightarrow \overline{Q}=0$ y $R=0 \Rightarrow Q=1$
1	0	0	1		
1	0	1	0		
1	0	1	1		
1	1	0	0	ESTABLE	$S=1 \Rightarrow \overline{Q}=0$ y $R=1 \Rightarrow Q=0$
1	1	0	1		
1	1	1	0		
1	1	1	1		

## Biestable RS-NOR

b) Transiciones: un cambio en las entradas puede producir un cambio en las salidas.

Hay que tener en cuenta:

- Las puertas lógicas tienen un tiempo de propagación ( $t_p$ ), lo que supone un retardo en el biestable
- Los valores antes y después de la transición:

$Q_t$ : valor de Q antes de la transición

$Q_{t+1}$ : valor de Q después de la transición (transcurrido  $t_p$ )

Las posibles transiciones del biestable RS-NOR son:

Transición	S	R	$Q_t$	$\overline{Q}_t$	$S_t$	$R_t$	$Q_{t+1}$	$\overline{Q}_{t+1}$
1	0	0	0	1	0	1	0	1
2	0	0	0	1	1	0	1	0
3	0	0	0	1	1	1	0	0
4	0	0	1	0	0	1	0	1
5	0	0	1	0	1	0	1	0
6	0	0	1	0	1	1	0	0
7	0	1	0	1	0	0	0	1
8	0	1	0	1	1	0	1	0
9	0	1	0	1	1	1	0	0
10	1	0	1	0	0	0	1	0
11	1	0	1	0	0	1	0	1
12	1	0	1	0	1	1	0	0
13	1	1	0	0	0	0	INDET	
14	1	1	0	0	0	1	0	1
15	1	1	0	0	1	0	1	0

$Q_{t+1}$  y  $\overline{Q}_{t+1}$  se obtienen a partir del valor de las entradas S y R tras cambiarlas en el instante t ( $S_t$  y  $R_t$ ) y del valor, también en el instante t, de Q y  $\overline{Q}$  ( $Q_t$  y  $\overline{Q}_t$ )

Estos casos no interesan porque cuando las entradas S y R valen 1 y las salidas Q y  $\overline{Q}$  valen 0, como veremos en la siguiente diapositiva, da lugar a oscilaciones en el valor de las salidas

Un cambio en las entradas  $\Rightarrow$  el circuito pase por distintos estados hasta llegar un estado estable. A veces, como en el caso de la transición 13, no llega a un estado estable

# Circuitos biestables

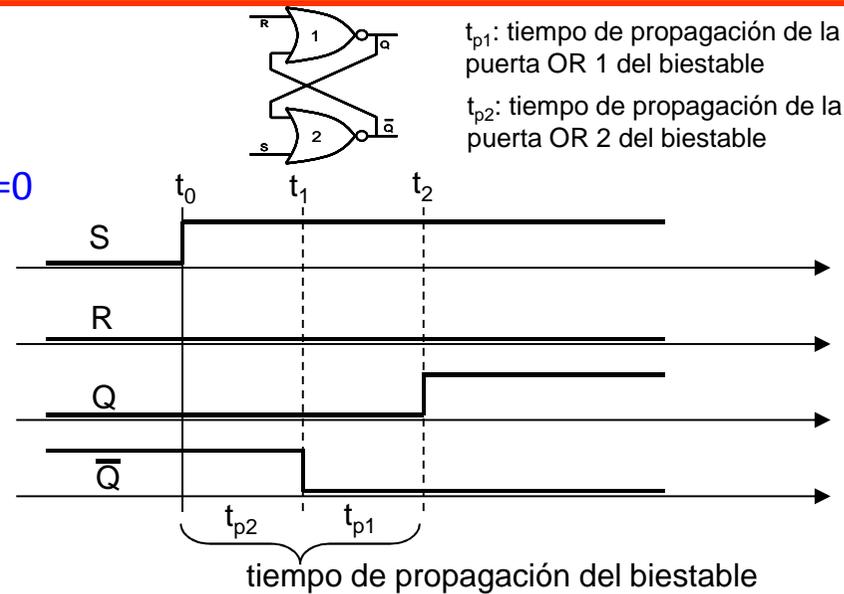
## Bi stable RS-NOR

Ejemplos de transiciones:

- Transición 2:  $S=0 R=0 Q=0 \bar{Q}=1 \rightarrow S=1 R=0 Q=1 \bar{Q}=0$

	S	R	Q	$\bar{Q}$		
$t_0 \rightarrow$	Inicial	0	0	0	1	
$t_1 \rightarrow$	Cambio $S=1$	1	0	0	1	$\bar{Q}_{t+1} = \bar{S} + \bar{Q}_t$
$t_2 \rightarrow$	$S=1 \Rightarrow \bar{Q}=0$	1	0	0	0	
	$\bar{Q}=0$ y $R=0 \Rightarrow Q=1$	1	0	1	0	ESTABLE

$Q_{t+1} = R + \bar{Q}_t$



- Transición 13:  $S=1 R=1 Q=0 \bar{Q}=0 \rightarrow S=0 R=0 Q$  y  $\bar{Q}$  INDETERMINADOS

a) Si  $t_{p1} = t_{p2}$  (puerta 1 igual de rápida que la 2)

	S	R	Q	$\bar{Q}$		
$t_0 \rightarrow$	Inicial	1	1	0	0	
$t_1 \rightarrow$	Cambio $S=0$ y $R=0$	0	0	0	0	
$t_2 \rightarrow$	$S=0$ y $Q=0 \Rightarrow \bar{Q}=1$	0	0	1	1	OSCILACIONES=INESTABLE
	$R=0$ y $\bar{Q}=0 \Rightarrow Q=1$	0	0	1	1	
	$Q=1, \bar{Q}=1 \Rightarrow Q=0, \bar{Q}=0$	0	0	0	0	

b) Si  $t_{p1} < t_{p2}$  (puerta 1 más rápida que la 2)

	S	R	Q	$\bar{Q}$		
$t_0 \rightarrow$	Inicial	1	1	0	0	
$t_1 \rightarrow$	Cambio $S=0$ y $R=0$	0	0	0	0	
	$R=0$ y $\bar{Q}=0 \Rightarrow Q=1$	0	0	1	0	ESTABLE

c) Si  $t_{p1} > t_{p2}$  (puerta 1 menos rápida que la 2)

	S	R	Q	$\bar{Q}$		
$t_0 \rightarrow$	Inicial	1	1	0	0	
$t_1 \rightarrow$	Cambio $S=0$ y $R=0$	0	0	0	0	
	$S=0$ y $Q=0 \Rightarrow \bar{Q}=1$	0	0	0	1	ESTABLE

## Bi stable RS-NOR

Agrupando las transiciones válidas se obtiene

Transición	ESTADO INICIAL				ESTADO FINAL			
	S	R	$Q_t$	$\overline{Q}_t$	$S_t$	$R_t$	$Q_{t+1}$	$\overline{Q}_{t+1}$
7	0	1	0	1	0	0	0	1
10	1	0	1	0	0	0	1	0
1	0	0	0	1	0	1	0	1
4	0	0	1	0	0	1	0	1
11	1	0	1	0	0	1	0	1
2	0	0	0	1	1	0	1	0
8	0	1	0	1	1	0	1	0
5	0	0	1	0	1	0	1	0

S	R	$Q_{t+1}$	
0	0	$Q_t$	NO CAMBIA
0	1	0	RESET
1	0	1	SET
1	1		SISTEMA NO DESEABLE

## Conclusión:

Tal y como se indicó al principio, el RS-NOR es un elemento de memoria:

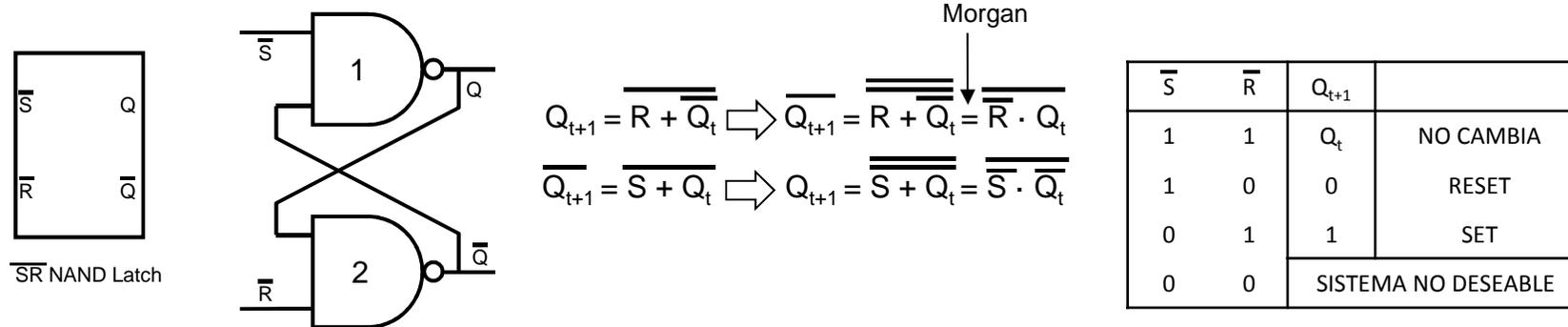
Para almacenar un 1 se pone el SET a 1  $\Rightarrow S = 1$  y  $R = 0 \Rightarrow Q_{t+1} = 1$

Al volver SET a 0  $\Rightarrow S = 0$  y  $R = 0 \Rightarrow Q_{t+1} = Q_t = 1$

Para almacenar un 0 (borrar) se pone el RESET a 1  $\Rightarrow S = 0$  y  $R = 1 \Rightarrow Q_{t+1} = 0$

Al volver RESET a 0  $\Rightarrow S = 0$  y  $R = 0 \Rightarrow Q_{t+1} = Q_t = 0$

## Biastable RS-NAND:



$\overline{R}$ =RESET  $\Rightarrow \overline{R}=0$  y  $\overline{S}=1 \Rightarrow \overline{Q}=1$  y  $Q=0$  (el biestable almacena un 0 lógico) }  $\overline{S}$  y  $\overline{R}$  activas a nivel bajo  
 $\overline{S}$ =SET  $\Rightarrow \overline{S}=1$  y  $\overline{R}=0 \Rightarrow Q=1$  y  $\overline{Q}=0$  (el biestable almacena un 1 lógico)

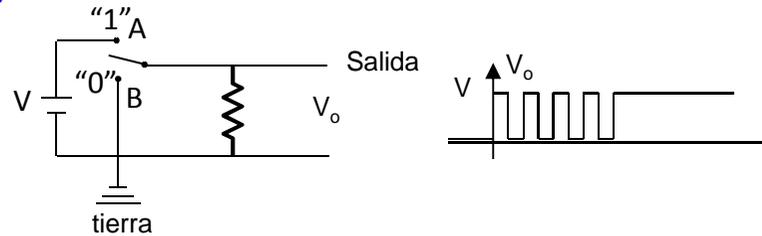
Cuando  $R=0$  y  $S=0$  no se manda ni SET ni RESET y el biestable mantiene su estado anterior. Si almacenaba un 0 lógico ( $Q=0$  y  $\overline{Q}=1$ ) mantendrá los valores de salida, y si almacenaba un 1 lógico ( $Q=1$  y  $\overline{Q}=0$ ), mantendrá también sus valores de salida

Normalmente  $Q$  y  $\overline{Q}$  son complementarias, pero cuando  $\overline{R}=\overline{S}=0 \Rightarrow Q=\overline{Q}=0$  porque el biestable detecta que le llegan un reset y un set a la vez. **Este caso hay que evitarlo**

## Aplicación biestable RS: conmutador sin rebote

Los conmutadores mecánicos rebotan varias veces antes de adoptar la posición definitiva

### a) Influencia en un circuito sin biestable

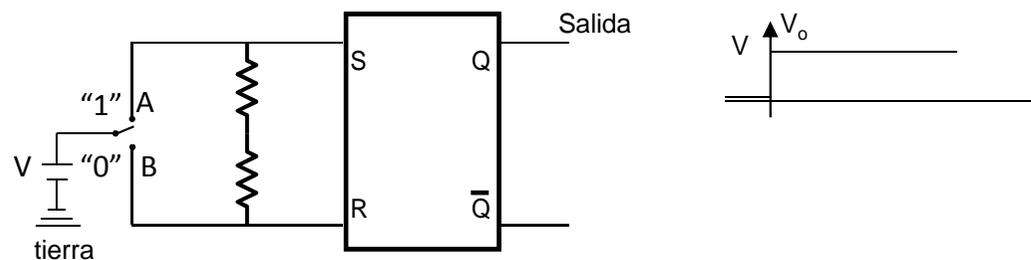


Al conmutar de B a A oscila entre dos posiciones hasta quedarse en A:

Conmutador en A  $\longrightarrow$   $S=V="1"$   
 Conmutador en el espacio entre A y B  $\longrightarrow$   $S= \text{tierra}="0"$

} La salida oscila entre 1 y 0

### b) Influencia en un circuito con biestable



Al conmutar de B a A oscila entre dos posiciones hasta quedarse en A:

Conmutador en A  $\longrightarrow$   $S=V="1"$  y  $R=\text{tierra}="0"$   $\longrightarrow$   $Q=1$   
 Conmutador en el espacio entre A y B  $\longrightarrow$   $S=\text{tierra}="0"$  y  $R=\text{tierra}="0"$   $\longrightarrow$   $Q(t+1)=Q(t)=1$

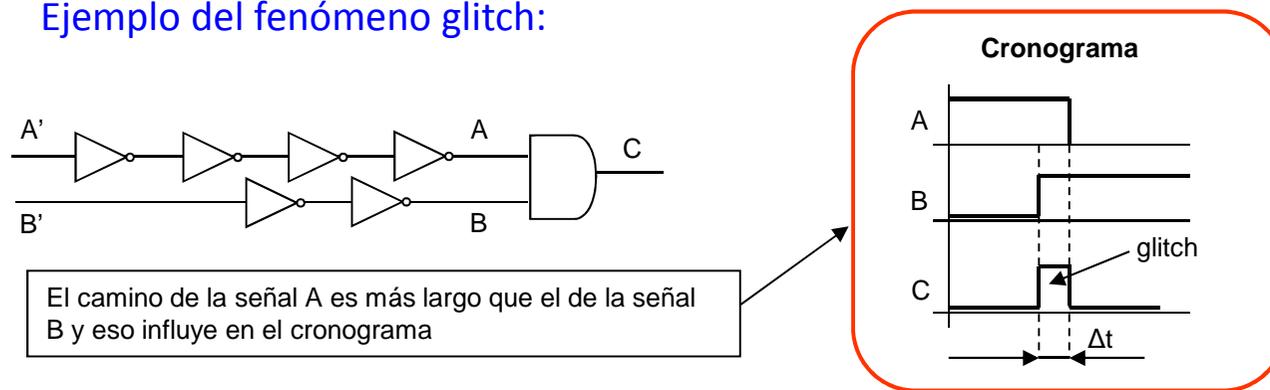
} La salida no oscila, se queda en 1

## Problemática de los biestables RS asíncronos

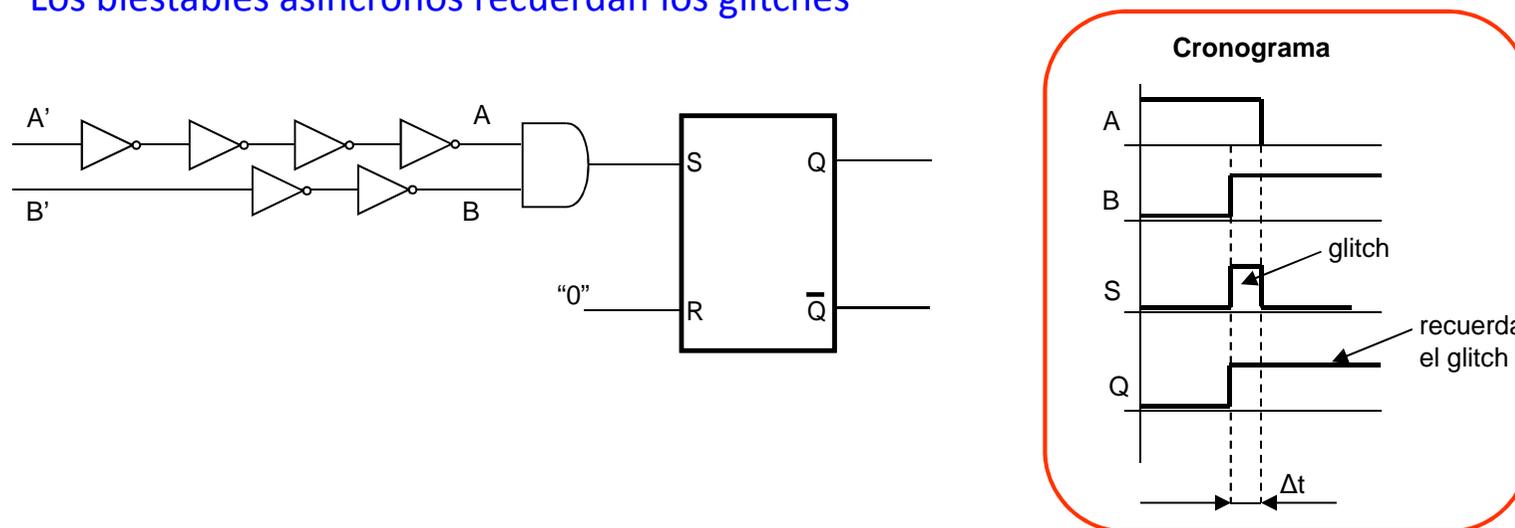
Los biestables RS-NAND y RS-NOR vistos hasta ahora no tienen entrada para señal de reloj y por eso se clasifican como biestables asíncronos.

Los biestables asíncronos tienen el problema de recordar los glitches, que son señales no deseadas originadas por retardos diferentes de las señales transmitidas por caminos diferentes

Ejemplo del fenómeno glitch:



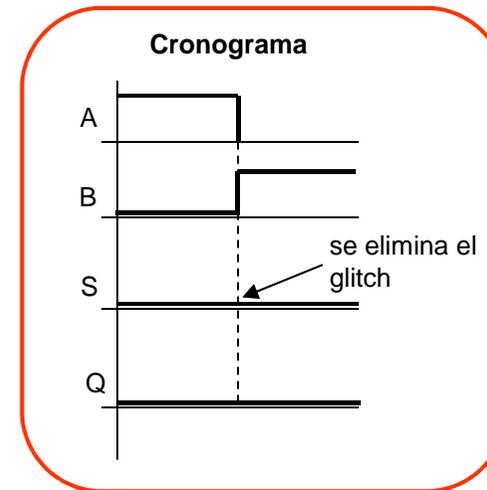
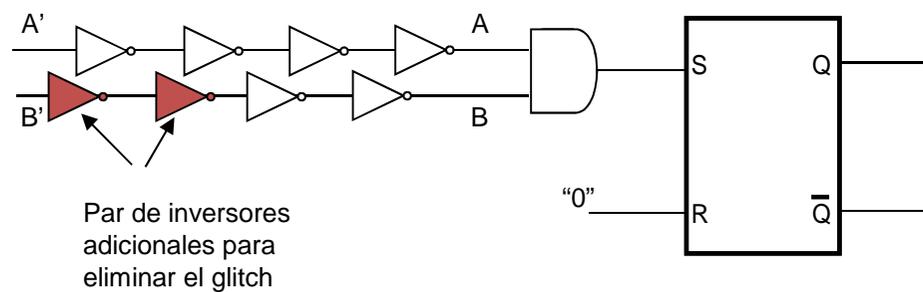
Los biestables asíncronos recuerdan los glitches



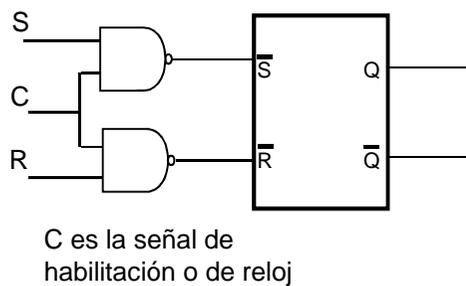
# Circuitos biestables

## Soluciones a la problemática de los biestables RS asíncronos

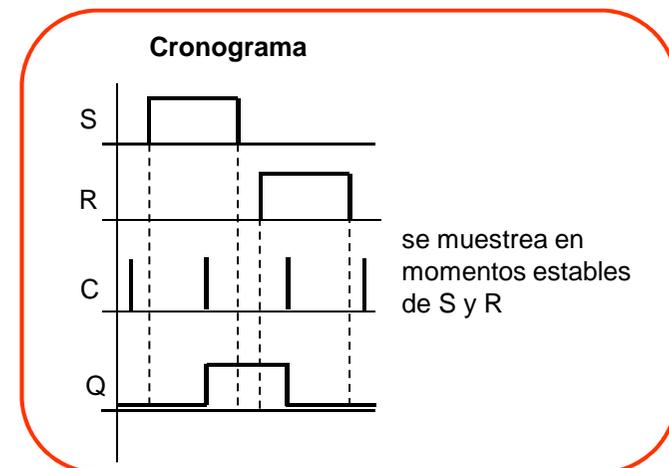
a) Retardar la señal que va por el camino más rápido mediante parejas de inversores (en el ejemplo de la diapositiva anterior mediante 2 inversores por la línea de la señal B)



b) Utilizar un biestable síncrono: un reloj fija los instantes de tiempo en que se permite la entrada de datos

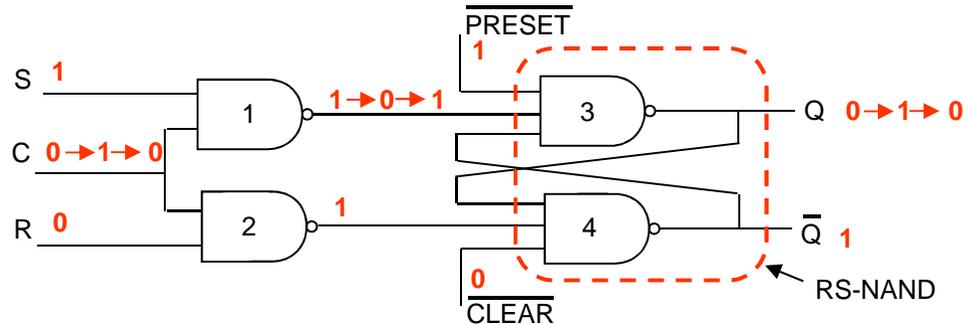


C	S	R	$Q_{t+1}$
0	X	X	$Q_t$
1	0	1	0 RESET
1	1	0	1 SET
1	0	0	$Q_t$
1	1	1	NO DESEABLE



## Biestable RS síncrono con entradas asíncronas

Además de las entradas síncronas S y R controladas por el reloj C, puede tener entradas asíncronas  $\overline{\text{PRESET}}$  y  $\overline{\text{CLEAR}}$  no controladas por reloj (ambas son activas a nivel bajo).



Las entradas asíncronas prevalecen sobre las síncronas (sirven para inicializar el biestable)

$\overline{\text{PRESET}} = 0$  (poner a SET)  $\Rightarrow Q=1$  ( $\overline{Q}=0$ )

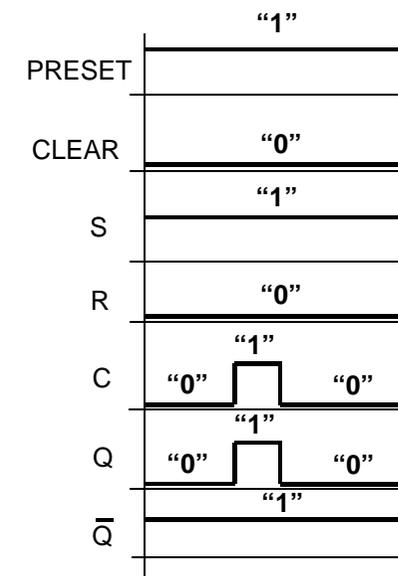
$\overline{\text{CLEAR}} = 0$  (borrar)  $\Rightarrow Q=0$  ( $\overline{Q}=1$ )

$\overline{\text{PRESET}} = \overline{\text{CLEAR}} = 1$   $\Rightarrow$  Actúan las síncronas (se habilitan puertas 3 y 4)

$\overline{\text{PRESET}} = \overline{\text{CLEAR}} = 0$   $\Rightarrow$  Situación no deseada (se manda SET y RESET a la vez – posibles oscilaciones)

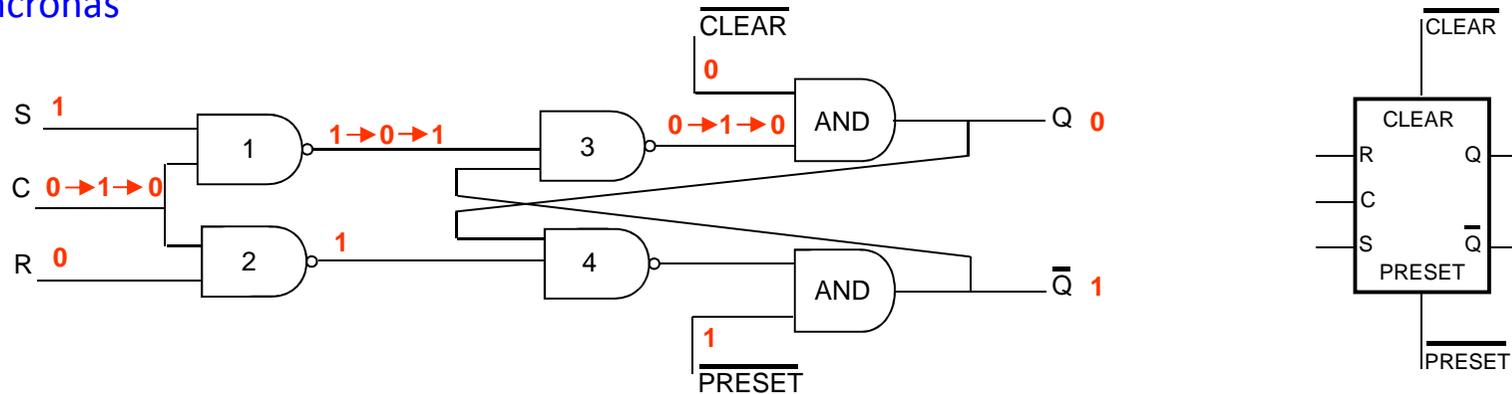
Anomalía: se propagan los impulsos de reloj a la salida

Si  $\overline{\text{PRESET}} = 1$  y  $\overline{\text{CLEAR}} = 0$ ,  $Q=0$ , pero en la realidad no es así



## Solución de anomalía del biestable RS síncrono con entradas asíncronas:

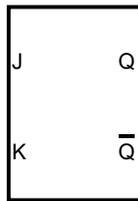
Utilizar puertas AND que bloquean el efecto de las entradas síncronas cuando se activan las asíncronas



$\overline{\text{PRESET}}$	$\overline{\text{CLEAR}}$	C	S	R	$Q_{t+1}$	$\overline{Q}_{t+1}$	
0	0	X	X	X	0*	0*	ESTADO INDETERMINADO AL VOLVER $\overline{\text{PRESET}} = \overline{\text{CLEAR}} = 1$
0	1	X	X	X	1	0	SET ( $\overline{\text{PRESET}} = 0$ )
1	0	X	X	X	0	1	RESET ( $\overline{\text{CLEAR}} = 0$ )
1	1		0	0	$Q_t$	$\overline{Q}_t$	NO CAMBIA
1	1		0	1	0	1	RESET (R=1)
1	1		1	0	1	0	SET (S=1)
1	1		1	1	INDETERMINADO		

- Biestable JK**

Es como el RS pero a la entrada R se le llama K, a la S se le llama J, y elimina la indeterminación que produce en RS cuando  $R=1$  y  $S=1$  (cuando  $J=1$  y  $K=1$  invierte la salida almacenada por el biestable)



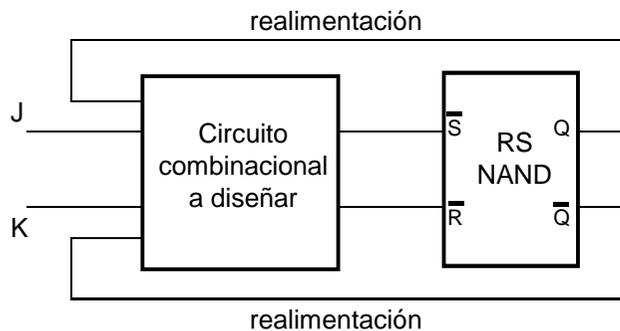
JK Latch

J = SET  
K = RESET

J	K	$Q_{t+1}$	
0	0	$Q_t$	NO CAMBIA
0	1	0	RESET
1	0	1	SET
1	1	$\overline{Q_t}$	INVIERTE

**Diseño de un JK a partir de un RS-NAND:**

Realimentando las salidas del RS-NAND a un circuito combinacional que también recibe las entradas JK y cuyas salidas se conectan a las entradas del RS-NAND. **El método vale para cualquier biestable**



**Tablas de verdad del JK y el RS-NAND:**

ENTRADAS			SALIDA
J	K	$Q_t$	$Q_{t+1}$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

ENTRADAS			SALIDA
$\overline{S}$	$\overline{R}$	$Q_t$	$Q_{t+1}$
0	0		SITUACIÓN NO DESEABLE
0	0		SITUACIÓN NO DESEABLE
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

En el RS-NAND, ¿qué valores S y R hacen que  $Q_t$  pase a  $Q_{t+1}$ ? Como ejemplo, en rojo se marcan los casos en que se pasa de  $Q_t=0$  a  $Q_{t+1}=0$ , que son  $\overline{S}=1, \overline{R}=0$  y  $\overline{S}=1, \overline{R}=1$ . Es decir  $\overline{S}=1, \overline{R}=X$ .

# Circuitos biestables

## Diseño de un JK a partir de un RS-NAND (continuación):

Tras haber deducido que para transitar de  $Q_t = 0$  a  $Q_{t+1} = 0$ ,  $S=1$  y  $R=X$ , se analizan el resto de casos y se obtiene la tabla de transiciones del RS-NAND:

$Q_t$	$Q_{t+1}$	$\overline{S}$	$\overline{R}$
0	0	1	X
0	1	0	1
1	0	1	0
1	1	X	1

ENTRADAS				SALIDAS	
J	K	$Q_t$	$Q_{t+1}$	$\overline{S}$	$\overline{R}$
0	0	0	0	1	X
0	0	1	1	X	1
0	1	0	0	1	X
0	1	1	0	1	0
1	0	0	1	0	1
1	0	1	1	X	1
1	1	0	1	0	1
1	1	1	0	1	0

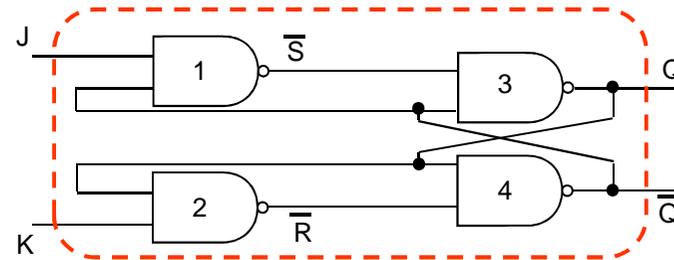
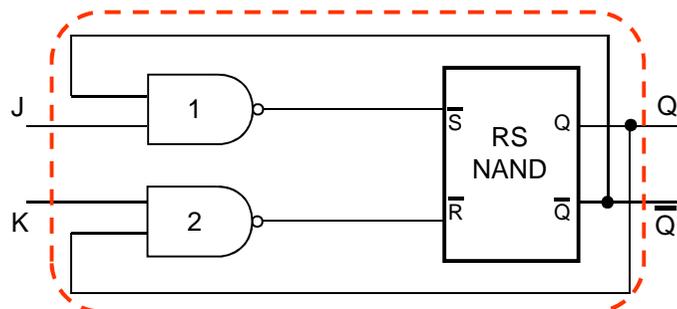
Simplificando por Karnaugh S y R:

$\overline{S}$	J/K	00	01	11	10
0	0	1	1	0	0
1	1	X	1	1	X

$$\overline{S} = \overline{J} + Q_t = \overline{J \cdot \overline{Q}_t}$$

$\overline{R}$	J/K	00	01	11	10
0	0	X	X	1	1
1	1	1	0	0	1

$$\overline{R} = \overline{K} + \overline{Q}_t = \overline{K \cdot Q_t}$$



$$Q_{t+1} = \overline{\overline{S} \cdot \overline{Q}_t}$$

$$\overline{Q}_{t+1} = \overline{\overline{R} \cdot \overline{Q}_t}$$

Problema: debido a la doble realimentación se producirán oscilaciones cuando  $J=K=1$

## Ejemplo de problema con oscilaciones en biestable JK

Análisis en el dominio del tiempo:

Slot de tiempo	J	K	$\bar{S}$	$\bar{R}$	Q	$\bar{Q}$
Inicial	0	0	1	1	0	1
Slot de tiempo 1	1	1	1	1	0	1
Slot de tiempo 2	1	1	0	1	0	1
Slot de tiempo 3	1	1	0	1	1	1
Slot de tiempo 4	1	1	0*	0*	1	0
Slot de tiempo 5	1	1	1	0	1	1
Slot de tiempo 6	1	1	0*	0*	0	1
Slot de tiempo 7	1	1	0	1	1	1

Se repite el lazo indefinidamente

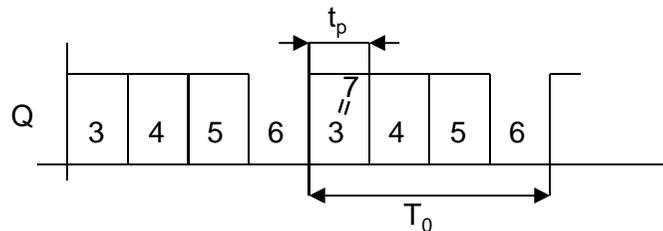
$$\bar{S} = \bar{J} + Q_t = \overline{J \cdot Q_t}$$

$$\bar{R} = \bar{K} + \bar{Q}_t = \overline{K \cdot \bar{Q}_t}$$

$$Q_{t+1} = \bar{S} \cdot \bar{Q}_t$$

$$\bar{Q}_{t+1} = \bar{R} \cdot Q_t$$

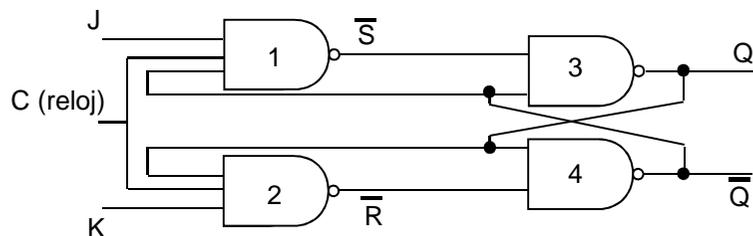
\* Situación no deseada



Se soluciona el problema con biestables más complejos

## Biestable JK síncrono elemental

Como en el biestable RS, la introducción de un reloj permite controlar cuándo se actualizan las salidas del biestable, pero no arregla el problema de oscilaciones, que no tiene que ver con el reloj



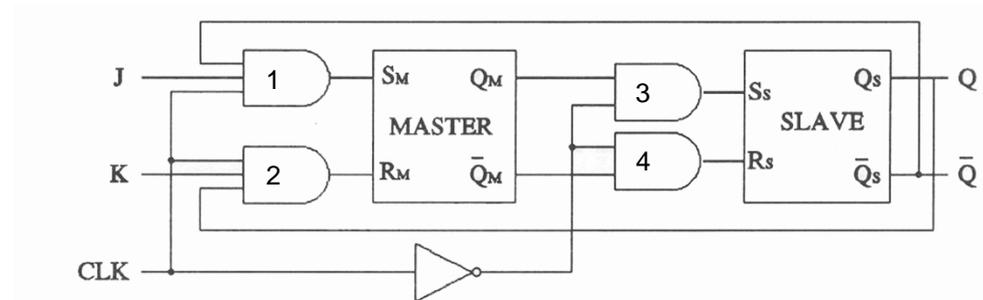
C=1 habilita NAND 1 y 2 y el circuito funciona como si no hubiera reloj  
C=0 inhabilita NAND 1 y 2, y pone  $\bar{S}$  y  $\bar{R}$  a 1 (no se actualizan las salidas Q y  $\bar{Q}$ )

# Circuitos biestables

## Biastable JK maestro-esclavo

Soluciona las oscilaciones del JK aislando la salida de la entrada para que no exista un camino de realimentación no deseado

En los JK maestro esclavo la lectura del dato y la actualización del biestable ocurren en diferente nivel de ciclo de reloj



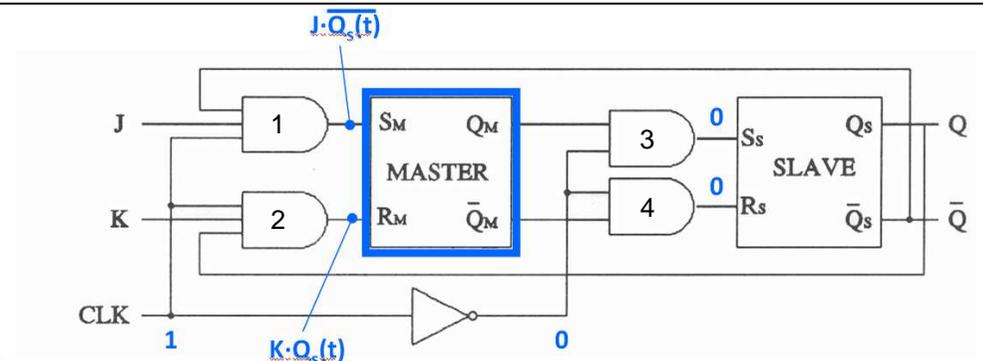
### a) Reloj a nivel alto

AND 1 y 2 habilitadas, luego:

$S_M = \bar{Q} \cdot J$  y  $R_M = Q \cdot K$  (se evita  $S_M = R_M = 1$  y el maestro adopta el estado ordenado por J y K)

AND 3 y 4 deshabilitadas, luego:

$S_s = R_s = 0$  (se mantiene la salida del esclavo)



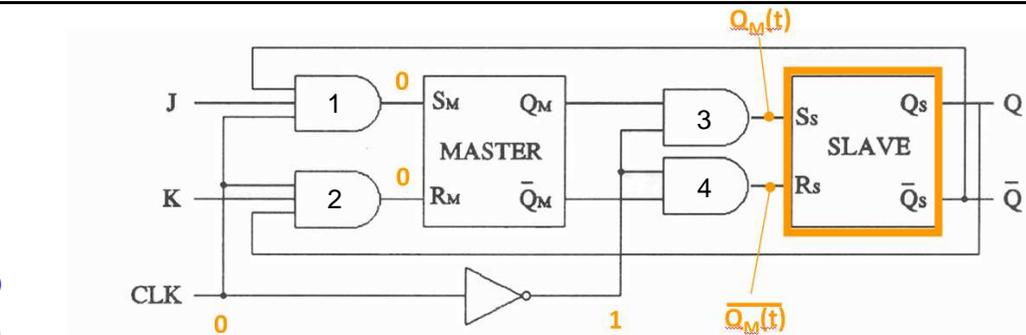
### b) Reloj a nivel bajo

AND 1 y 2 deshabilitados, luego:

$S_M = R_M = 0$  (el maestro mantiene su estado)

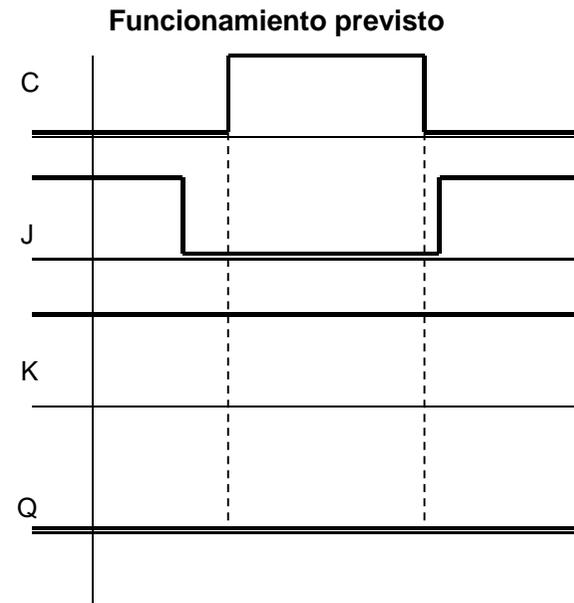
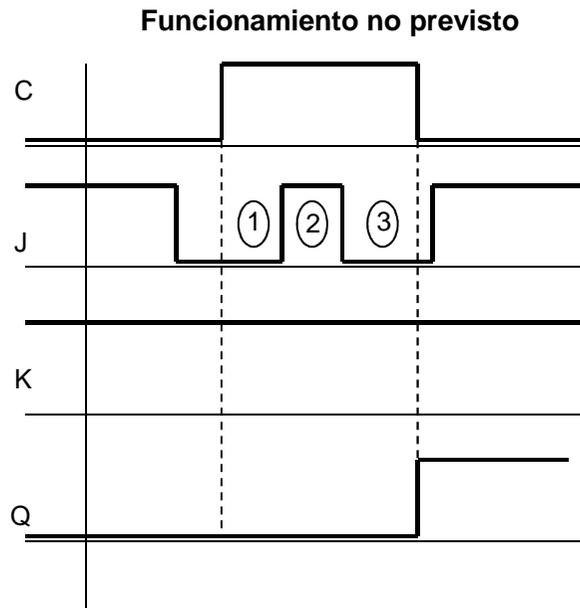
AND 3 Y 4 habilitadas, luego:

$S_s = Q_M$  y  $R_s = \bar{Q}_M$  (el esclavo coge del maestro su valor de salida y lo copia en su propia salida)



## Problema del biestable JK maestro-esclavo

Cuando en el intervalo en que la señal de reloj está a nivel alto se producen variaciones en la entrada J, se puede producirse un comportamiento anómalo



Funcionamiento previsto o esperado:

$J=0$   $K=1$  cuando  $C$  pasa de 1 a 0 las salidas se ponen:  $Q=0$  y  $\bar{Q}=1$

Funcionamiento no previsto o no esperado

- ①  $J=0$   $K=1$   $Q=0$  y  $\bar{Q}=1$ ,  $S_M=\bar{Q}\cdot J=0$  y  $R_M=Q\cdot K=0$   $\Rightarrow$  mantiene su estado el maestro ( $Q_M=0$  y  $\bar{Q}_M=1$ )
- ②  $J=1$   $K=1$   $Q=0$  y  $\bar{Q}=1$ ,  $S_M=\bar{Q}\cdot J=1$  y  $R_M=Q\cdot K=0$   $\Rightarrow$  se manda un set al maestro ( $Q_M=1$  y  $\bar{Q}_M=0$ )
- ③  $J=0$   $K=1$   $Q=0$  y  $\bar{Q}=1$ ,  $S_M=\bar{Q}\cdot J=0$  y  $R_M=Q\cdot K=0$   $\Rightarrow$  mantiene su estado el maestro ( $Q_M=1$  y  $\bar{Q}_M=0$ )  
y cuando  $C=0$   $S_S=Q_M=1$  y  $R_S=\bar{Q}_M=0$  se manda un set al esclavo ( $Q=1$  y  $\bar{Q}=0$ )

**Conclusión:** para un buen funcionamiento hace falta que  $C=1$  en un intervalo de tiempo corto

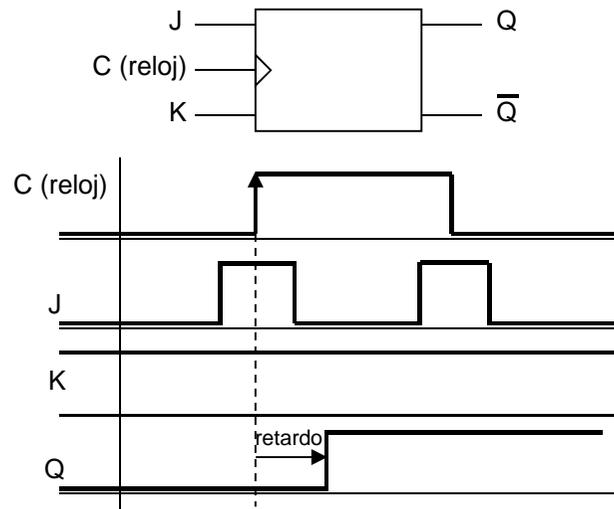
# Circuitos biestables

## Biestable JK disparado por flanco (en inglés edge)

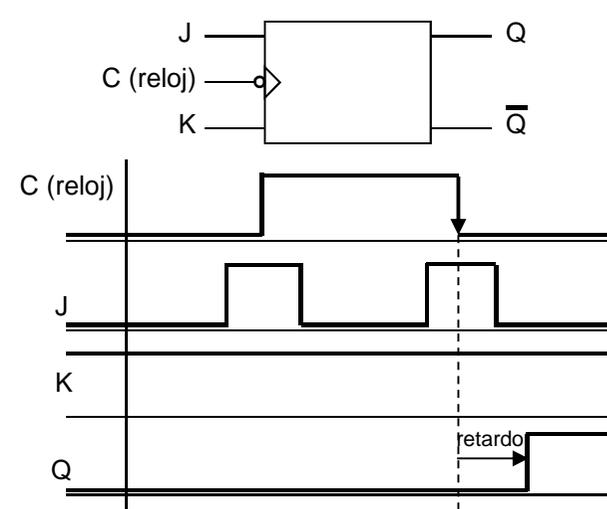
Soluciona el comportamiento no previsto del JK maestro esclavo realizando la lectura de las entradas y la actualización de las salidas en uno de los flancos del impulso de reloj, llamado flanco activo

Tipos:

a) Flanco ascendente

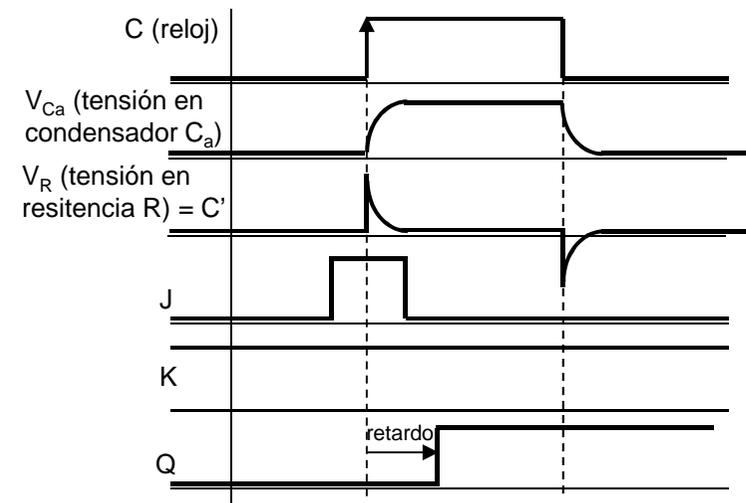
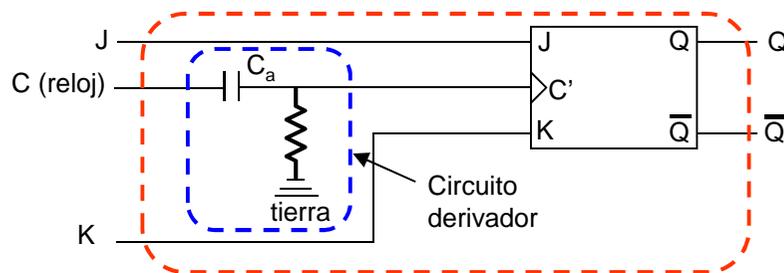


b) Flanco descendente



Las entradas deben estar estables durante un tiempo previo al flanco activo y la salida transita con un retardo respecto al flanco activo

Circuito biestable JK edge a partir de maestro-esclavo:

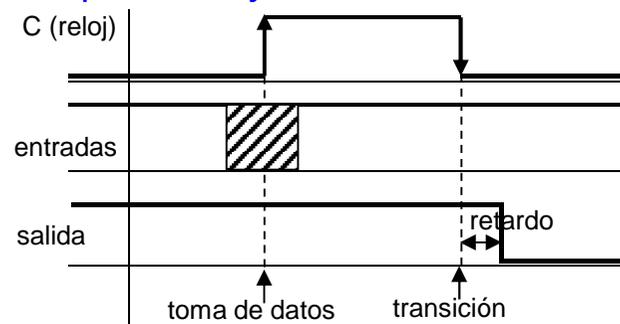


## Biestable JK maestro esclavo con cierre de datos SN74111 (<http://bit.ly/1sXylam>)

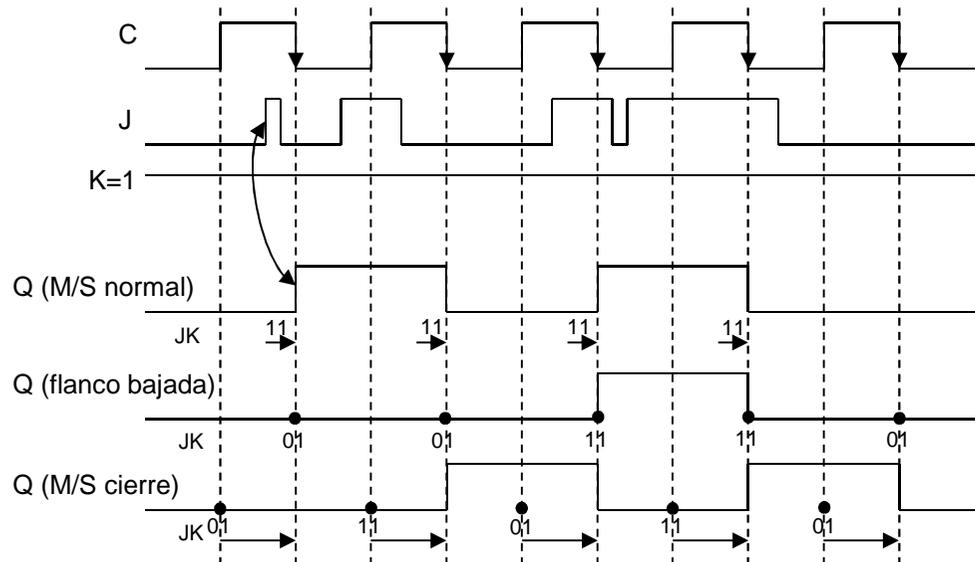
La lectura de las entradas (toma de datos) se realiza en el flanco ascendente (+) del impulso de reloj

La transición de la salida tiene lugar en el flanco descendente (-) del impulso de reloj (una vez transcurrido el tiempo de retardo)

A diferencia del maestro esclavo normal no le afectan los cambios en las entradas durante el intervalo en que el reloj está a nivel alto

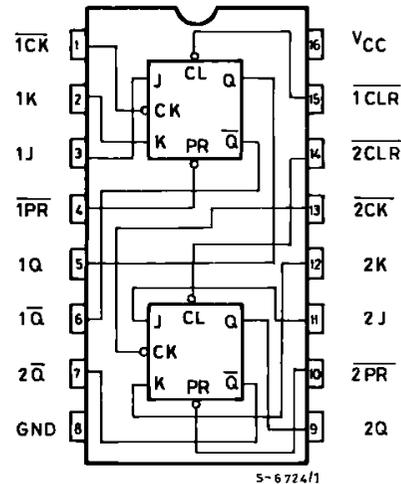
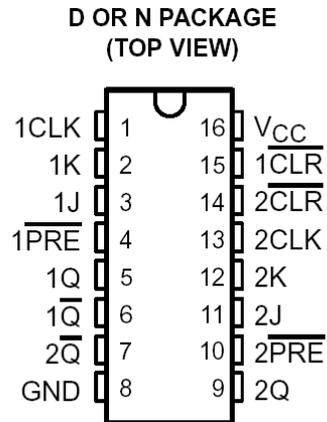


## Comportamiento de los distintos biestables JK



# Circuitos biestables

Biestable JK comercial disparado por flanco de bajada con clear y preset (<http://bit.ly/1p9Msiv>)



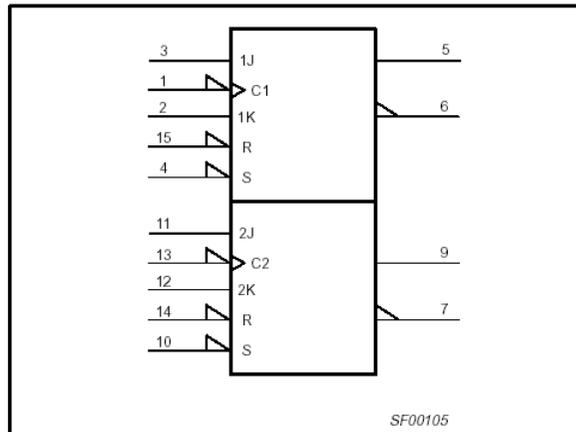
SN74F112  
DUAL NEGATIVE-EDGE-TRIGGERED J-K FLIP-FLOP  
WITH CLEAR AND PRESET  
SDFS048A - D2932, MARCH 1987 - REVISED OCTOBER 1993

FUNCTION TABLE

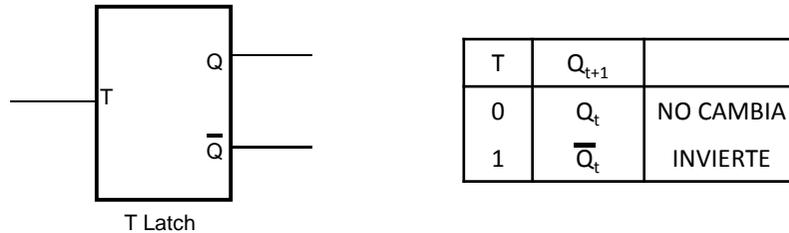
INPUTS					OUTPUTS	
$\overline{\text{PRE}}$	$\overline{\text{CLR}}$	CLK	J	K	Q	$\overline{\text{Q}}$
L	H	X	X	X	H	L
H	L	X	X	X	L	H
L	L	X	X	X	H <sup>†</sup>	H <sup>†</sup>
H	H	↓	L	L	Q <sub>0</sub>	$\overline{\text{Q}}_0$
H	H	↓	H	L	H	L
H	H	↓	L	H	L	H
H	H	↓	H	H	Toggle	
H	H	H	X	X	Q <sub>0</sub>	$\overline{\text{Q}}_0$

† The output levels in this configuration are not guaranteed to meet the minimum levels for  $V_{OH}$ . Furthermore, this configuration is nonstable; that is, it will not persist when either  $\overline{\text{PRE}}$  or  $\overline{\text{CLR}}$  returns to its inactive (high) level.

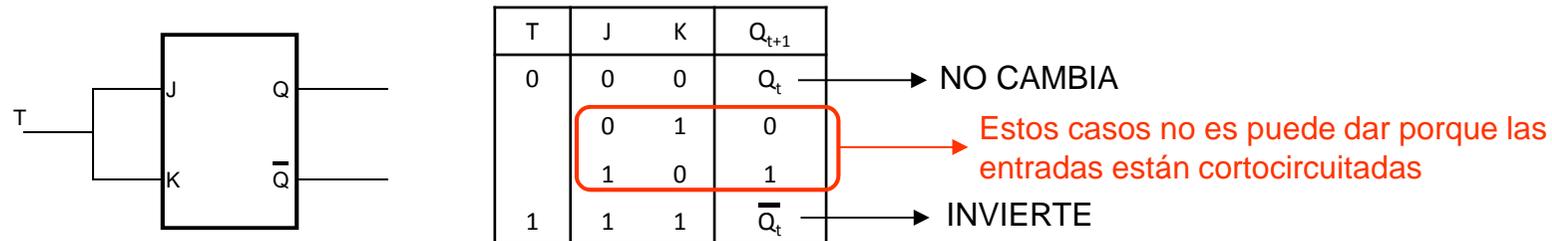
IEC/IEEE SYMBOL



- Biestable T (toggle o de conmutación)**

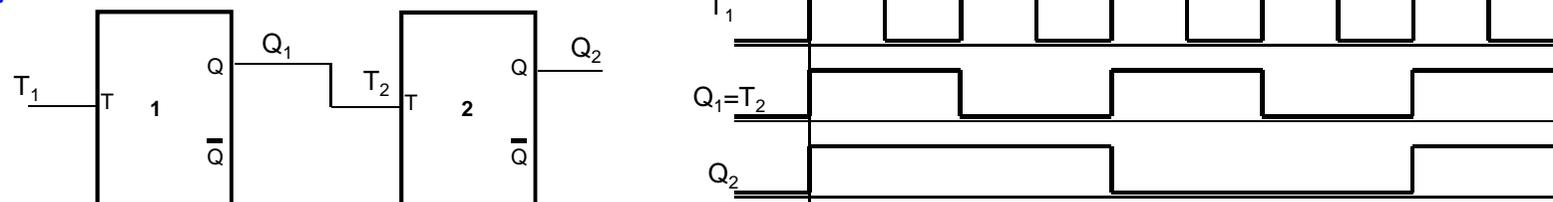


Se puede construir a partir de un biestable JK, uniendo las entradas J y K:

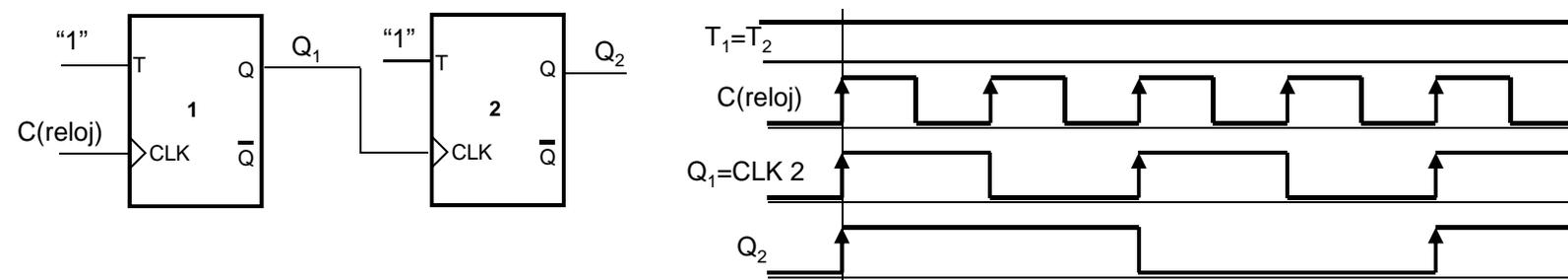


Aplicación de biestable T: divisor de frecuencia (cada biestable divide por 2 la frecuencia)

a) con un biestable T asíncrono

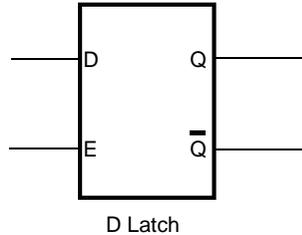


b) Con un biestable T síncrono por flanco de subida



No se consideran los retardos

- Bi stable D (de muestreo o retención)**



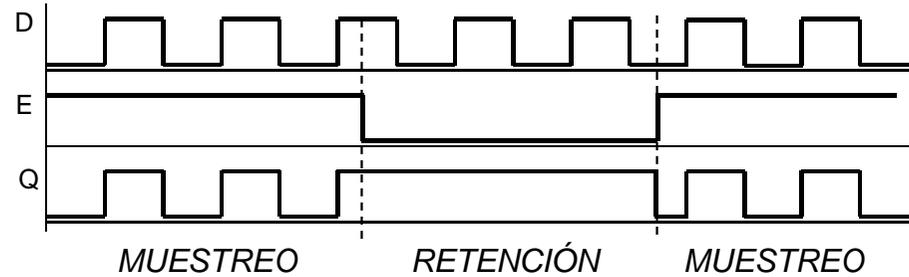
E	$Q_{t+1}$	
0	$Q_t$	NO CAMBIA (RETENCIÓN)
1	D	MUESTREALA ENTRADA

La tabla anterior se puede expresar también como:

E	D	$Q_t$	$Q_{t+1}$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

$E=0$  (modo retención)  
 $Q_{t+1}=Q_t$

$E=1$  (modo muestreo)  
 $Q_{t+1}=D$

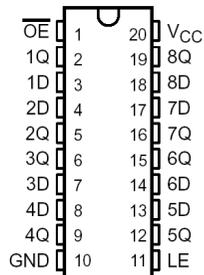


Nota: no se consideran los retardos

**Bi stable D tipo latch comercial SN74F373 (<http://bit.ly/1oQJujo>):**

**SN54F373, SN74F373**  
**OCTAL TRANSPARENT D-TYPE LATCHES**  
**WITH 3-STATE OUTPUTS**  
SDFS076A - D2932, MARCH 1987 - REVISED OCTOBER 1993

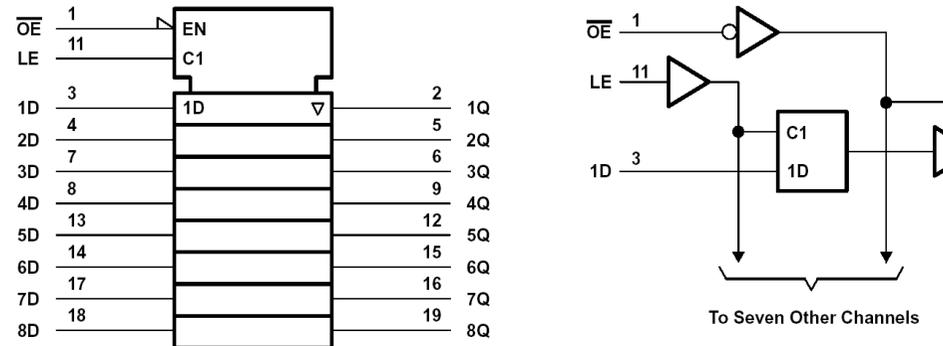
SN54F373 . . . J PACKAGE  
 SN74F373 . . . DB, DW, OR N PACKAGE  
 (TOP VIEW)



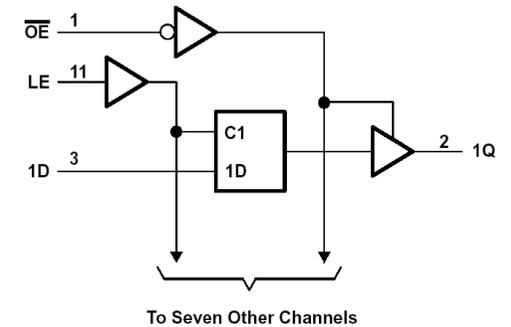
**FUNCTION TABLE**  
(each latch)

INPUTS			OUTPUT
$\overline{OE}$	LE	D	Q
L	H	H	H
L	H	L	L
L	L	X	$Q_0$
H	X	X	Z

logic symbol†

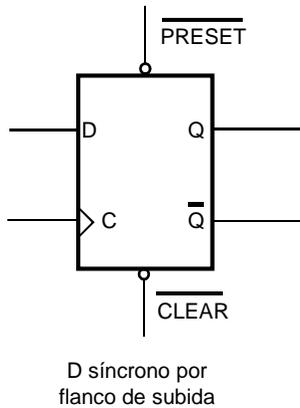


logic diagram (positive logic)



† This symbol is in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.

## Biestado D síncrono



PRESET	CLEAR	C	D	$Q_t$	$Q_{t+1}$
0	1	X	X	1	0
1	0	X	X	0	1
0	0	X	X	1	1
1	1		0	0	0
1	1		0	1	1
1	1		1	0	0
1	1		1	1	1
1	1	↑	0	0	0
1	1	↑	0	1	0
1	1	↑	1	0	1
1	1	↑	1	1	1

SET asíncrono  
 RESET asíncrono  
 No deseado

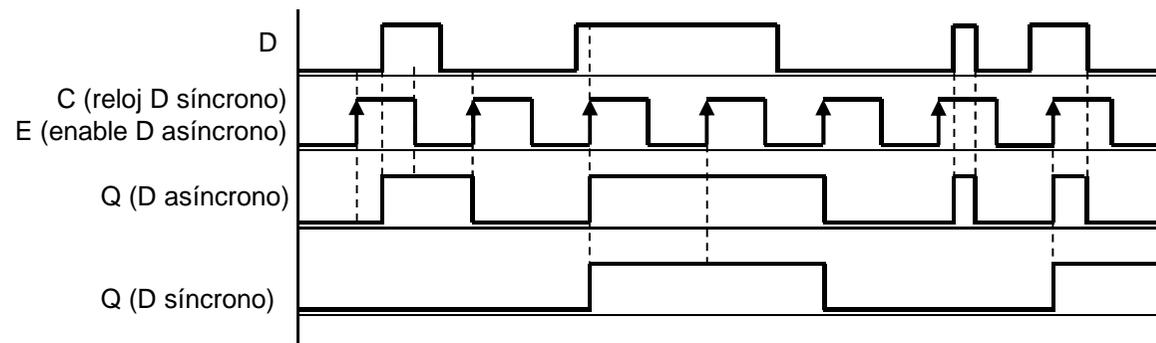
Predominan las entradas asíncronas

E=0 (modo retención)  
 $Q_{t+1}=Q_t$

E=1 (modo muestreo)  
 $Q_{t+1}=D$

Predominan las entradas síncronas

## Comparativa entre biestado asíncrono y biestado síncrono por flanco de subida



Nota: no se consideran los retardos

- **Tiempos característicos de los biestables**

**Tiempo de establecimiento (set-up time):** mínimo tiempo anterior al flanco activo de toma de datos durante el que los niveles lógicos de las señales de entrada no deben cambiar

**Tiempo de mantenimiento (hold time):** mínimo tiempo posterior al flanco activo de toma de datos durante el que las entradas no deben cambiar

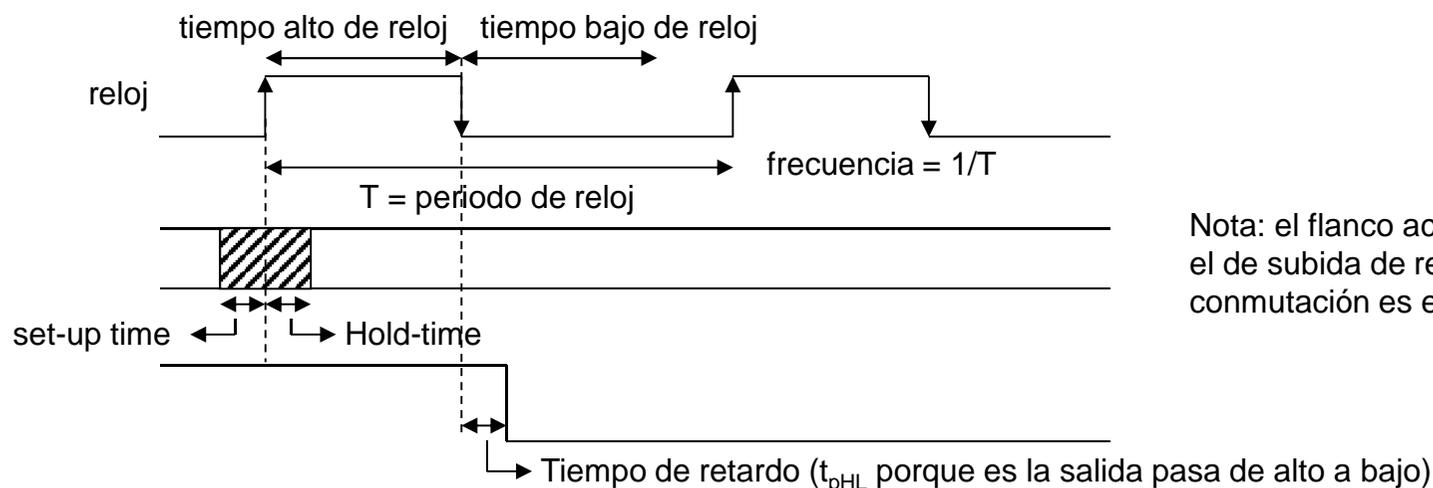
**Frecuencia máxima de reloj:** frecuencia máxima admisible de la señal de reloj para la que el fabricante garantiza el correcto funcionamiento del circuito

**Duración del tiempo alto de reloj:** tiempo mínimo que debe durar la parte alta del impulso de reloj

**Duración del tiempo bajo de reloj:** tiempo mínimo que debe durar la parte baja del impulso de reloj

**Tiempo bajo de  $\overline{\text{PRESET}}$  y  $\overline{\text{CLEAR}}$ :** tiempo mínimo que deben permanecer activadas las entradas asíncronas para garantizar un correcto funcionamiento

**Tiempo de retardo o de propagación:** tiempo que transcurre desde el flanco activo de reloj, que ordena la transición de las salidas hasta el momento que la transición tiene lugar (los fabricantes diferencian el tiempo de transición de alto a bajo  $t_{pHL}$  y el de bajo a alto  $t_{pLH}$ )



Nota: el flanco activo de toma de datos es el de subida de reloj, y el flanco activo de conmutación es el de bajada de reloj

- Tiempos característicos en un biestable comercial

timing requirements over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted)

		$V_{CC} = 5\text{ V},$ $T_A = 25^\circ\text{C}$		MIN	MAX	UNIT
		MIN	MAX			
$f_{\text{clock}}$	Clock frequency	0	110	0	100	MHz
$t_w$	Pulse duration	CLK high or low	4.5	5		ns
		$\overline{\text{CLR}}$ or $\overline{\text{PRE}}$ low	4.5	5		
$t_{\text{su}}$	Setup time, data before CLK↓	High	4	5		ns
		Low	3	3.5		
$t_h$	Hold time, data after CLK↓	High	0	0		ns
		Low	0	0		
$t_{\text{su}}$	Setup time, inactive state, data before CLK↓§	$\overline{\text{CLR}}$ or $\overline{\text{PRE}}$ high	4	5		ns

§ Inactive-state setup time is also referred to as recovery time.

### switching characteristics (see Note 3)

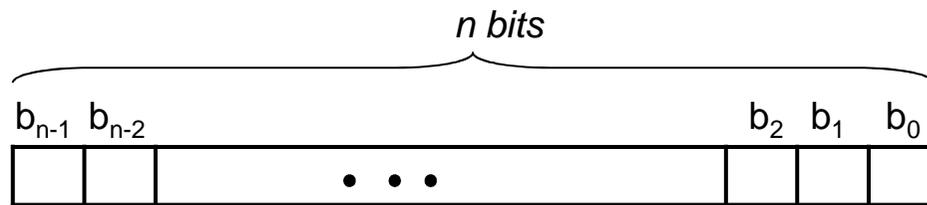
PARAMETER	FROM (INPUT)	TO (OUTPUT)	$V_{CC} = 5\text{ V},$ $C_L = 50\text{ pF},$ $R_L = 500\ \Omega,$ $T_A = 25^\circ\text{C}$			$V_{CC} = 4.5\text{ V to }5.5\text{ V},$ $C_L = 50\text{ pF},$ $R_L = 500\ \Omega,$ $T_A = \text{MIN to MAX}^\dagger$		UNIT
			MIN	TYP	MAX	MIN	MAX	
$f_{\text{max}}$			110	130		100		MHz
$t_{\text{PLH}}$	CLK	Q or $\overline{Q}$	1.2	4.6	6.5	1.2	7.5	ns
$t_{\text{PHL}}$			1.2	4.6	6.5	1.2	7.5	
$t_{\text{PLH}}$	$\overline{\text{PRE}}$ or $\overline{\text{CLR}}$	Q or $\overline{Q}$	1.2	4.1	6.5	1.2	7.5	ns
$t_{\text{PHL}}$			1.2	4.1	6.5	1.2	7.5	

† For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions.

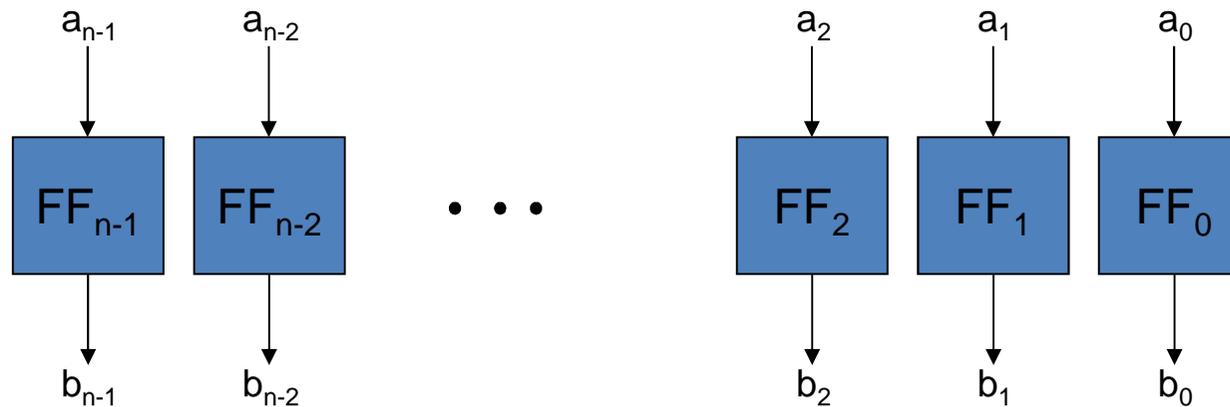
NOTE 3: Load circuits and waveforms are shown in Section 1.

- **Concepto de registro**

Son circuitos secuenciales contruidos con biestables que almacenan información (bits)



Un registro que almacena n bits está compuesto de n biestables (uno por cada bit):

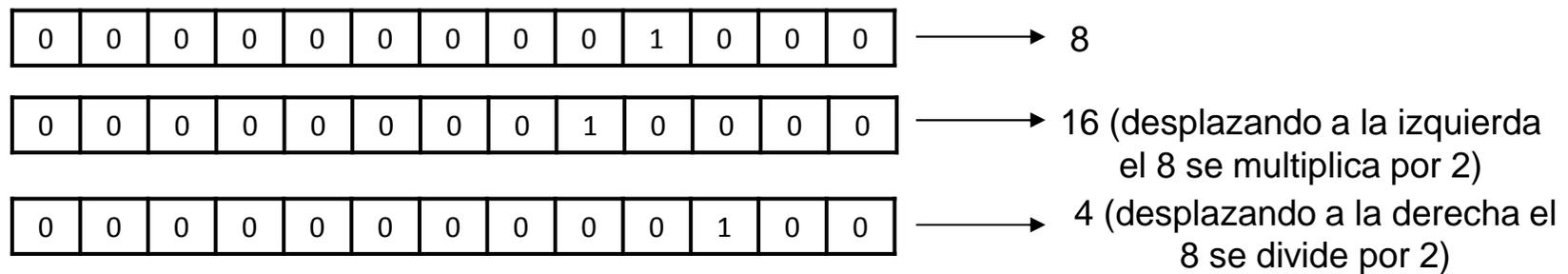


Los biestables contienen en sus salidas  $b_n b_{n-1} \cdots b_2 b_1 b_0$ , y estos valores dependen de las entradas de los biestables  $a_n a_{n-1} \cdots a_2 a_1 a_0$

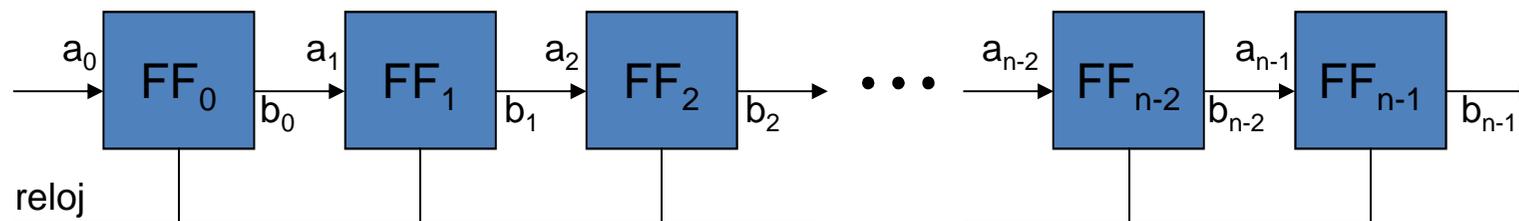
- Registros de desplazamiento**

Permiten desplazar la información (los bits) a lo largo del registro (de un biestable a otro).

Si el desplazamiento es hacia la izquierda se multiplica por 2, y si es hacia la derecha se divide por 2:



Conectando los biestables en cascada (la salida de cada biestable a la entrada del siguiente) se añade, a la funcionalidad de almacenar información de  $n$  bits, la de realizar operaciones básicas del aritmética binaria también con el mismo circuito: la multiplicación por 2 o la división por 2.

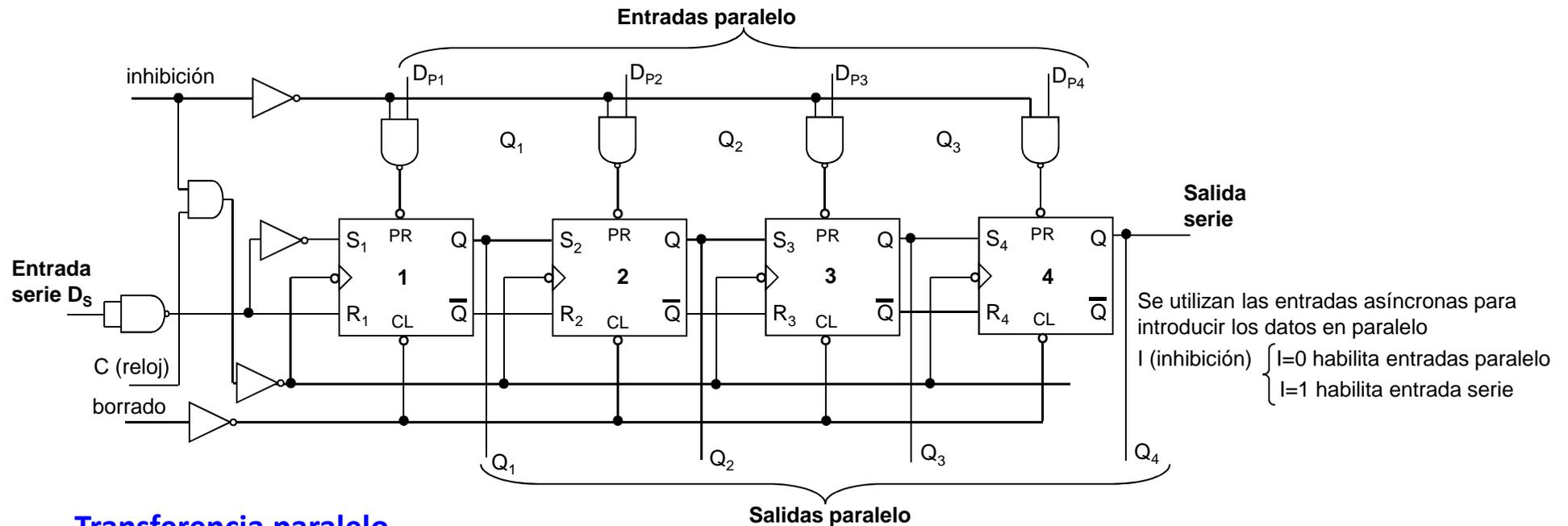


Se comprobará en las siguientes diapositivas que se puede integrar en un mismo circuito las tres funciones: almacenar  $n$  bits, desplazar a la izquierda y desplazar a la derecha



- Registros de desplazamiento: entrada serie/paralelo-salida serie/paralelo**

Ejemplo para 4 bits con 4 biestables RS síncronos por flanco descendente:



## Transferencia paralelo

1) Se borran los biestables: borrado=1  $\Rightarrow \overline{CL}_i=0 \Rightarrow CL_i=1$  }  $\Rightarrow Q_i=0$   
 inhibición=1  $\Rightarrow \overline{PR}_i=1 \Rightarrow PR_i=0$  }

2) Se introducen los datos en paralelo: borrado=0  $\Rightarrow CL_i=1$  (clear asíncrono inhabilitado)

inhibición=0  $\Rightarrow C=0$  (se inhabilita el reloj)

$DP_i=1 \Rightarrow \overline{PR}_i=0 \Rightarrow PR_i=1 \Rightarrow Q_i=1$

$DP_i=0 \Rightarrow \overline{PR}_i=1 \Rightarrow Q_i=0$  (se mantiene estado)

**Transferencia serie:** borrado=0  $\Rightarrow \overline{CL}_i=1 \Rightarrow CL_i=0$  } Funcionan las entradas síncronas. El circuito  
 inhibición=1  $\Rightarrow \overline{PR}_i=1 \Rightarrow PR_i=0$  } funciona como en la anterior diapositiva

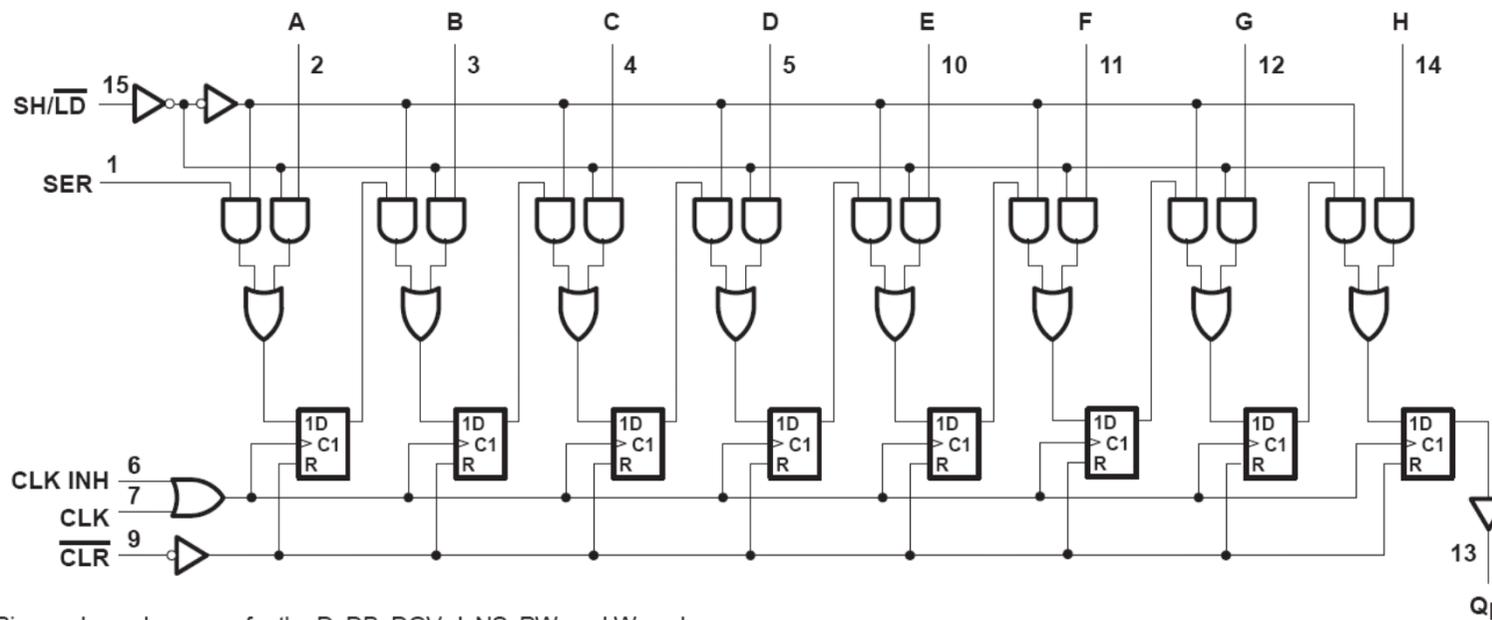
- **Registro desplazamiento comercial: entrada serie/paralelo-salida serie/paralelo**

Para 8 bits con 8 biestables D síncronos por flanco ascendente (<http://bit.ly/1kNPAkz>):

## SN54LV166A, SN74LV166A 8-BIT PARALLEL-LOAD SHIFT REGISTERS

SCLS456C - FEBRUARY 2001 - REVISED APRIL 2005

logic diagram (positive logic)



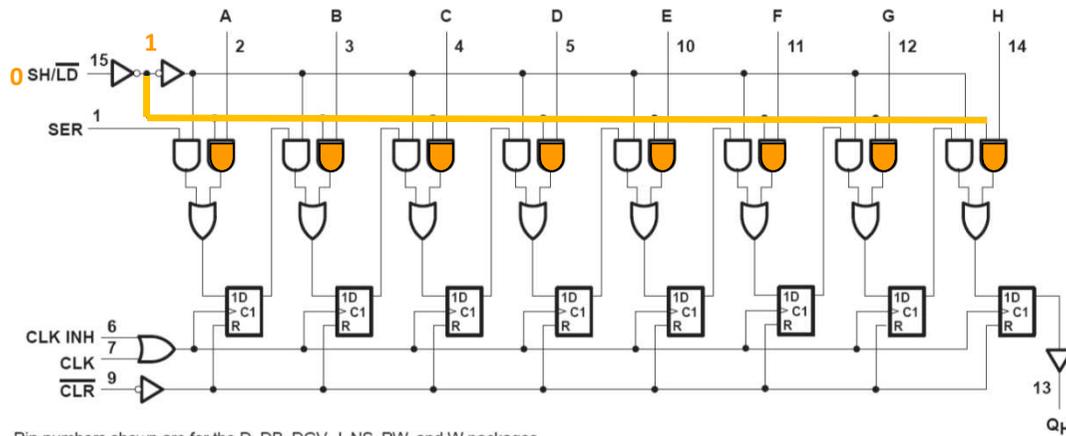
Pin numbers shown are for the D, DB, DGV, J, NS, PW, and W packages.

# Registros

- Registro desplazamiento comercial: entrada serie/paralelo-salida serie/paralelo**

La entrada SH/LD determina el modo de funcionamiento del circuito:

a)  $SH/\overline{LD}=0$  como carga de datos en paralelo

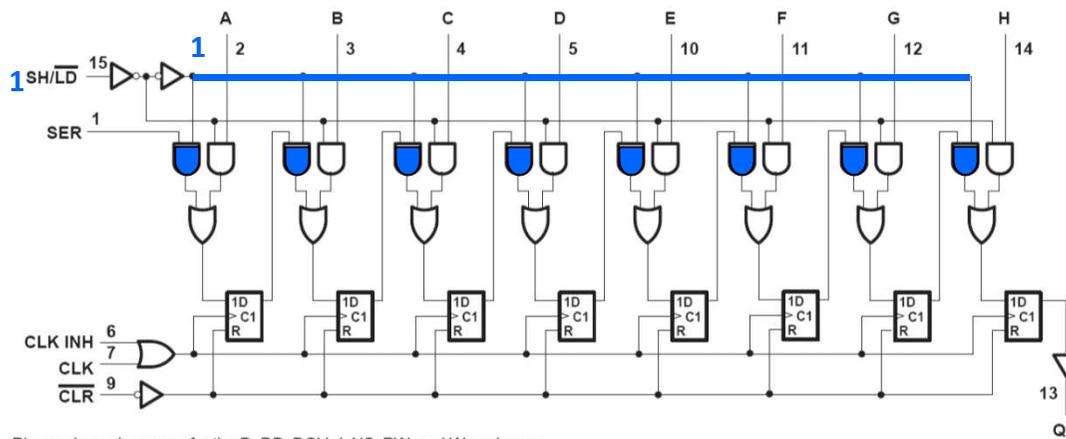


Pin numbers shown are for the D, DB, DGV, J, NS, PW, and W packages.

FUNCTION TABLE

INPUTS						OUTPUTS		
$\overline{CLR}$	$SH/\overline{LD}$	CLK INH	CLK	SER	PARALLEL A...H	INTERNAL QA	INTERNAL QB	QH
L	X	X	X	X	X	L	L	L
H	X	L	L	X	X	QA0	QB0	QH0
H	L	L	↑	X	a...h	a	b	h
H	H	L	↑	H	X	H	QA <sub>n</sub>	QG <sub>n</sub>
H	H	L	↑	L	X	L	QA <sub>n</sub>	QG <sub>n</sub>
H	X	H	↑	X	X	QA0	QB0	QH0

b)  $SH/\overline{LD}=1$  como registro de desplazamiento (cada ciclo del reloj CLK se desplazan los bits un lugar a la derecha)



Pin numbers shown are for the D, DB, DGV, J, NS, PW, and W packages.

FUNCTION TABLE

INPUTS						OUTPUTS		
$\overline{CLR}$	$SH/\overline{LD}$	CLK INH	CLK	SER	PARALLEL A...H	INTERNAL QA	INTERNAL QB	QH
L	X	X	X	X	X	L	L	L
H	X	L	L	X	X	QA0	QB0	QH0
H	L	L	↑	X	a...h	a	b	h
H	H	L	↑	H	X	H	QA <sub>n</sub>	QG <sub>n</sub>
H	H	L	↑	L	X	L	QA <sub>n</sub>	QG <sub>n</sub>
H	X	H	↑	X	X	QA0	QB0	QH0

- Registro desplazamiento hacia la izquierda**

El circuito de la figura 7.10 (muy similar al de la anterior diapositiva), se puede convertir en la figura 7.12 en un registro desplazamiento hacia la izquierda cableando las entradas y salidas

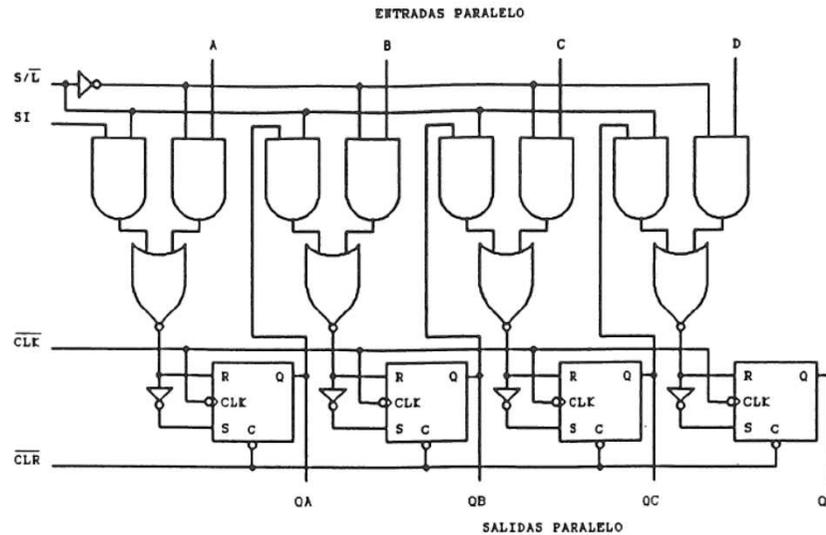


Fig. 7.10. Estructura de un registro de desplazamiento con entrada paralelo salida paralelo.

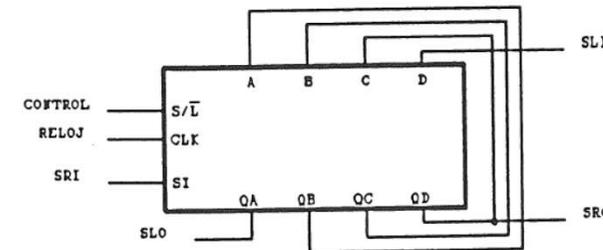


Fig. 7.12. Obtención de un registro de desplazamiento a la izquierda.

ENTRADAS								SALIDAS			
$\overline{CLR}$	$S/\overline{L}$	$\overline{CLK}$	SI	A	B	C	D	$Q_A$	$Q_B$	$Q_C$	$Q_D$
L	X	X	X	X	X	X	X	L	L	L	L
H	X	L o H	X	X	X	X	X	$Q_{A0}$	$Q_{B0}$	$Q_{C0}$	$Q_{D0}$
H	L	↓	X	a	b	c	d	a	b	c	d
H	H	↓	L	X	X	X	X	L	$Q_{An}$	$Q_{Bn}$	$Q_{Cn}$
H	H	↓	H	X	X	X	X	H	$Q_{An}$	$Q_{Bn}$	$Q_{Cn}$

H = Nivel alto. L = Nivel bajo.  
 X = Indiferente.  
 ↓ = Transición de nivel alto a bajo (flanco de bajada).  
 a...d = El nivel en las entradas A...D respectivamente.  
 $Q_{A0}$ ,  $Q_{B0}$ ,  $Q_{C0}$  y  $Q_{D0}$  = Estados de  $Q_A$ ,  $Q_B$ ,  $Q_C$  y  $Q_D$  respectivamente después de un flanco de bajada de la señal de reloj.  
 $Q_{An}$ ,  $Q_{Bn}$ ,  $Q_{Cn}$  y  $Q_{Dn}$  = nivel de  $Q_A$ ,  $Q_B$ ,  $Q_C$  y  $Q_D$  respectivamente antes de un flanco de bajada de la señal de reloj.

Fuente: Circuitos Electrónicos Digitales. Manuel Mazo et al. Universidad de Alcalá

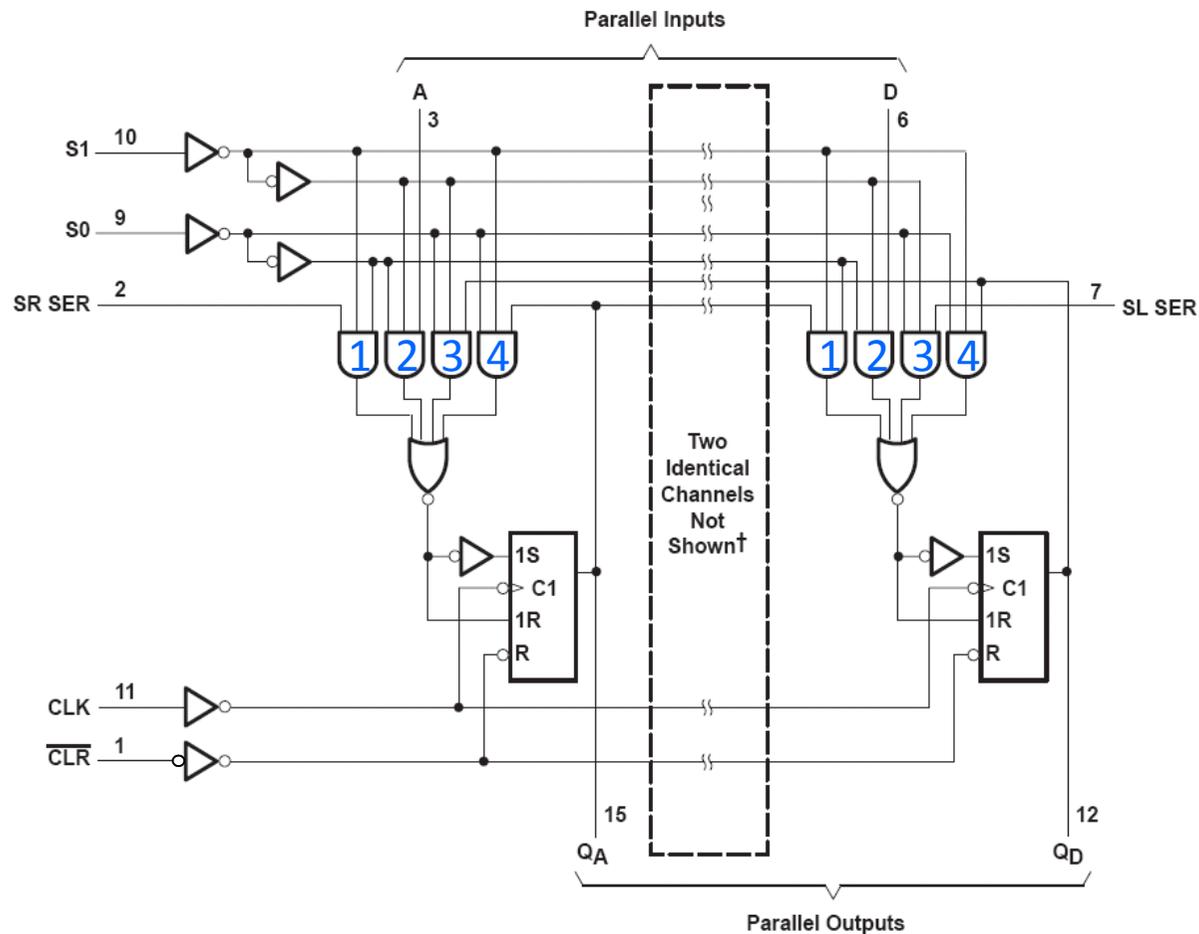
Sin embargo es una manera de obtener un registro de desplazamiento hacia la izquierda incómoda. Para conmutar del modo de funcionamiento hacia la derecha hacia la izquierda hace falta cablear. Para solucionarlo veremos el siguiente circuito

- Registro bidireccional de 4 bits (circuito)

**SN54AS194, SN74AS194**  
**4-BIT BIDIRECTIONAL UNIVERSAL SHIFT REGISTERS**

SDAS212A - DECEMBER 1983 - REVISED DECEMBER 1994

logic diagram (positive logic)

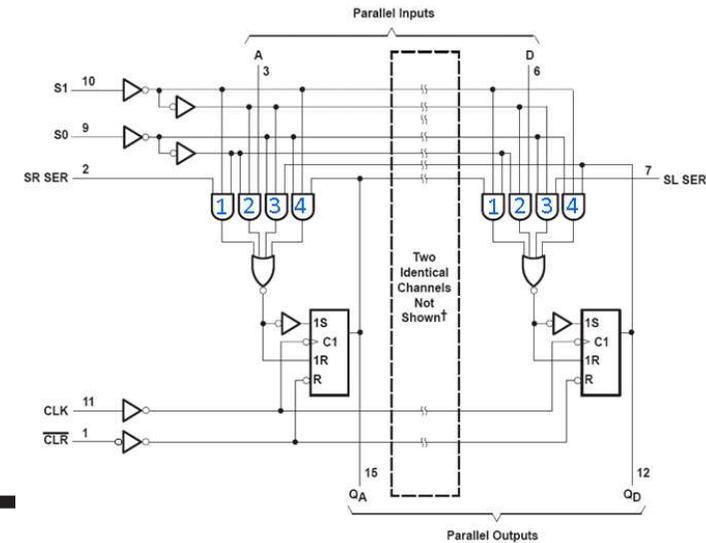


† I/O ports not shown: QB (14) and QC (13)  
Pin numbers shown are for the D, J, and N packages.

- Registro bidireccional de 4 bits (tabla de verdad)

## SN54AS194, SN74AS194 4-BIT BIDIRECTIONAL UNIVERSAL SHIFT REGISTERS

SDAS212A – DECEMBER 1983 – REVISED DECEMBER 1994



FUNCTION TABLE

			INPUTS							OUTPUTS			
$\overline{\text{CLR}}$	MODE		CLK	SERIAL		PARALLEL				QA	QB	QC	QD
	S1	S0		LEFT	RIGHT	A	B	C	D				
L	X	X	X	X	X	X	X	X	X	L	L	L	L
H	X	X	L	X	X	X	X	X	X	QA0	QB0	QC0	QD0
H	H	H	↑	X	X	a	b	c	d	a	b	c	d
H	L	H	↑	X	H	X	X	X	X	H	QAn	QBn	QCn
H	L	H	↑	X	L	X	X	X	X	L	QAn	QBn	QCn
H	H	L	↑	H	X	X	X	X	X	QBn	QCn	QDn	H
H	H	L	↑	L	X	X	X	X	X	QBn	QCn	QDn	L
H	L	L	X	X	X	X	X	X	X	QA0	QB0	QC0	QD0

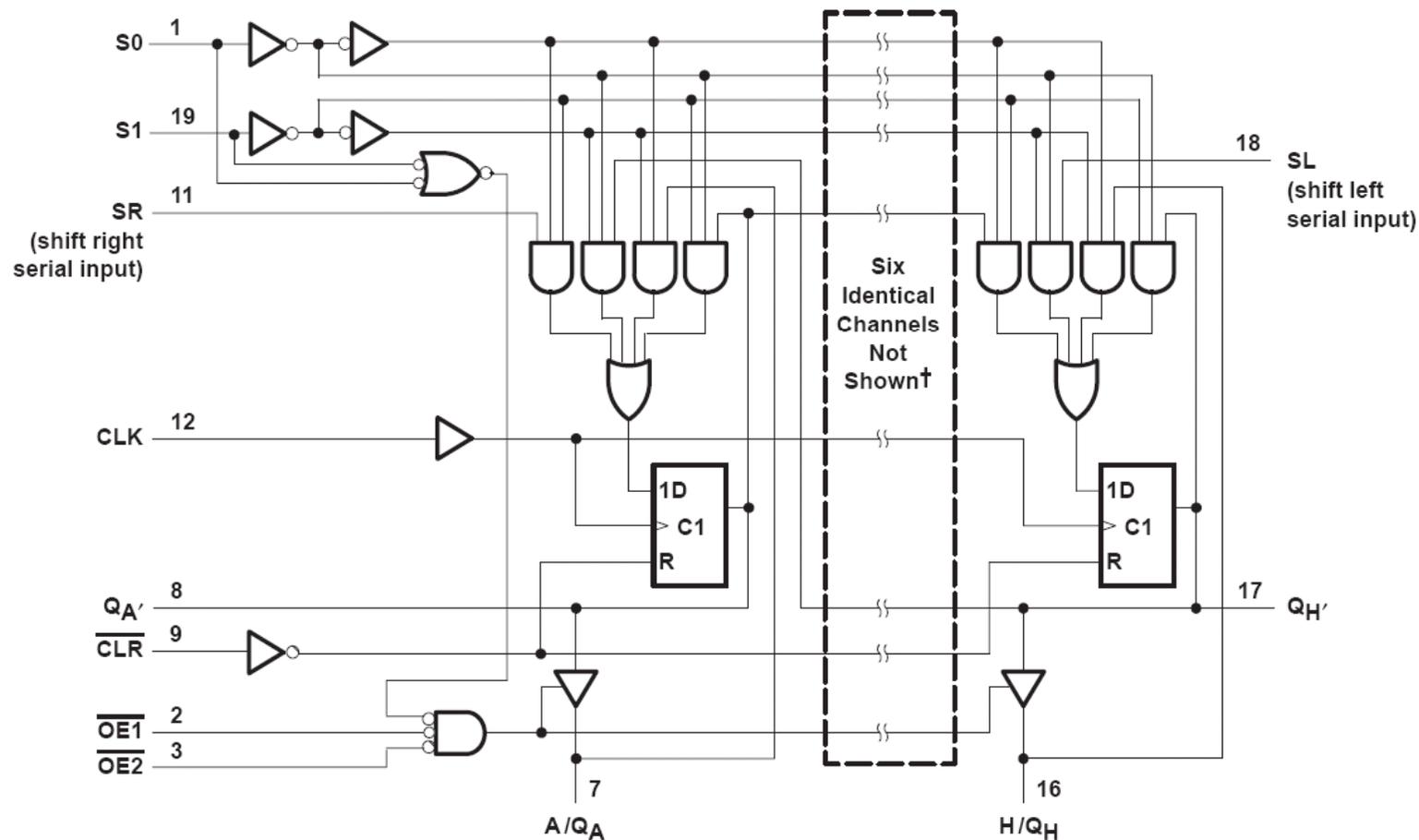
H = high level (steady state); L = low level (steady state); X = irrelevant (any input, including transitions); ↑ = transition from low to high level; a, b, c, d = the level of steady-state input at inputs A, B, C, or D, respectively; QA0, QB0, QC0, QD0 = the level of QA, QB, QC, or QD, respectively, before the indicated steady-state input conditions were established; QAn, QBn, QCn, QDn = the level of QA, QB, QC, respectively, before the most recent ↑ transition of the clock.

- Registro bidireccional de 8 bits (circuito)

## SN54ALS299, SN74ALS299 8-BIT UNIVERSAL SHIFT/STORAGE REGISTERS WITH 3-STATE OUTPUTS

SDAS220B – DECEMBER 1982 – REVISED DECEMBER 1994

logic diagram (positive logic)

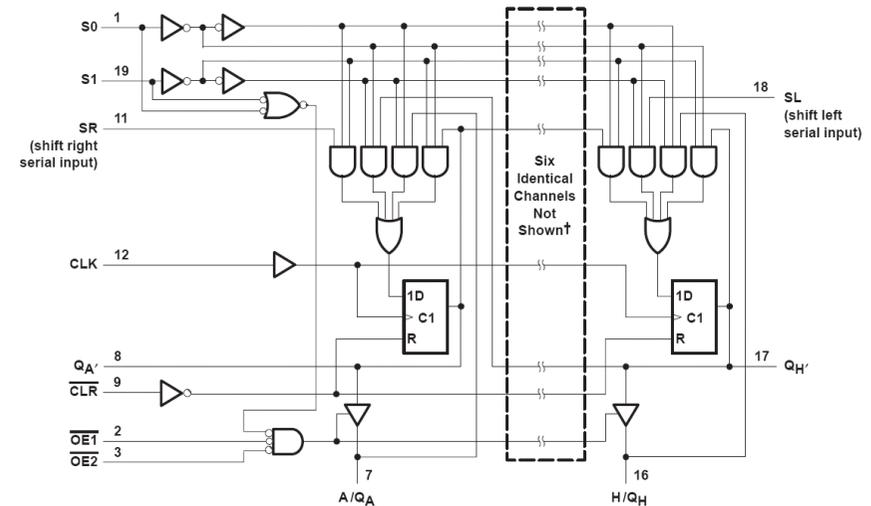


† I/O ports not shown: B/Q<sub>B</sub> (13), C/Q<sub>C</sub> (6), D/Q<sub>D</sub> (14), E/Q<sub>E</sub> (5), F/Q<sub>F</sub> (15), and G/Q<sub>G</sub> (4).

- Registro bidireccional de 8 bits (tabla de verdad)

## SN54ALS299, SN74ALS299 8-BIT UNIVERSAL SHIFT/STORAGE REGISTERS WITH 3-STATE OUTPUTS

SDAS220B – DECEMBER 1982 – REVISED DECEMBER 1994



FUNCTION TABLE

MODE	INPUTS								I/O PORTS								OUTPUTS			
	CLR	S1	S0	OE1†	OE2†	CLK	SL	SR	A/QA	B/QB	C/QC	D/QD	E/QE	F/QF	G/QG	H/QH	QA'	QH'		
Clear	L	X	L	L	L	X	X	X	L	L	L	L	L	L	L	L	L	L	L	L
	L	L	X	L	L	X	X	X	L	L	L	L	L	L	L	L	L	L	L	L
	L	H	H	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	L	L
Hold	H	L	L	L	L	X	X	X	QA0	QB0	QC0	QD0	QE0	QF0	QG0	QH0	QA0	QH0		
	H	X	X	L	L	L	X	X	QA0	QB0	QC0	QD0	QE0	QF0	QG0	QH0	QA0	QH0		
Shift Right	H	L	H	L	L	↑	X	H	H	QA <sub>n</sub>	QB <sub>n</sub>	QC <sub>n</sub>	QD <sub>n</sub>	QE <sub>n</sub>	QF <sub>n</sub>	QG <sub>n</sub>	H	QG <sub>n</sub>		
	H	L	H	L	L	↑	X	L	L	QA <sub>n</sub>	QB <sub>n</sub>	QC <sub>n</sub>	QD <sub>n</sub>	QE <sub>n</sub>	QF <sub>n</sub>	QG <sub>n</sub>	L	QG <sub>n</sub>		
Shift Left	H	H	L	L	L	↑	H	X	QB <sub>n</sub>	QC <sub>n</sub>	QD <sub>n</sub>	QE <sub>n</sub>	QF <sub>n</sub>	QG <sub>n</sub>	QH <sub>n</sub>	H	QB <sub>n</sub>	H		
	H	H	L	L	L	↑	L	X	QB <sub>n</sub>	QC <sub>n</sub>	QD <sub>n</sub>	QE <sub>n</sub>	QF <sub>n</sub>	QG <sub>n</sub>	QH <sub>n</sub>	L	QB <sub>n</sub>	L		
Load	H	H	H	X	X	↑	X	X	a	b	c	d	e	f	g	h	a	h		

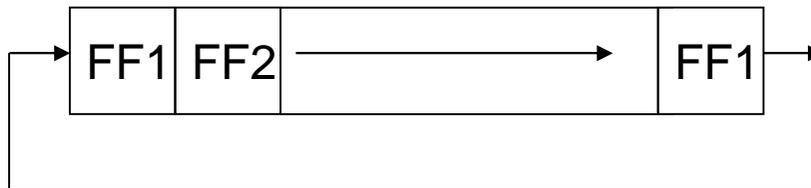
NOTE: a . . . h = the level of the steady-state input at inputs A through H, respectively. This data is loaded into the flip-flops while the flip-flop outputs are isolated from the I/O terminals.

† When one or both output-enable inputs are high, the eight I/O terminals are disabled to the high-impedance state; however, sequential operation or clearing of the register is not affected.

- **Registros de rotación**

Se les llama también registros de desplazamiento con conexión en anillo

La salida del último biestable está conectada a la entrada del primer biestable

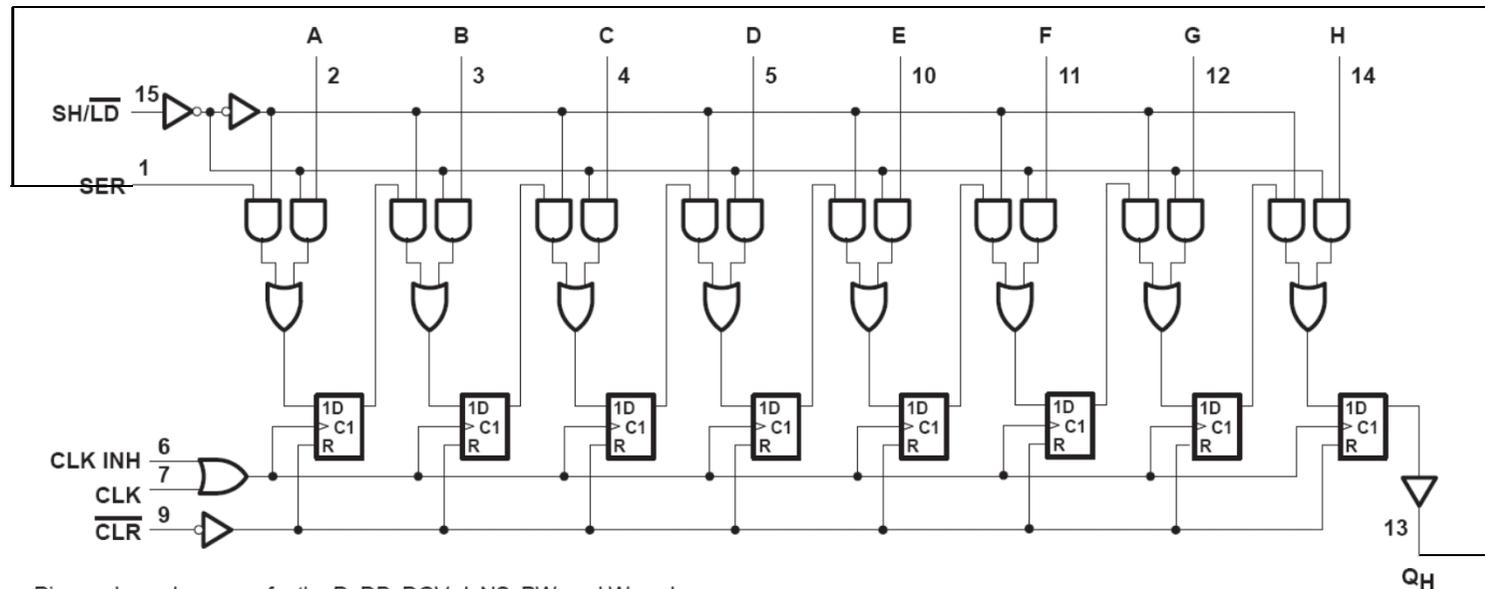


FF es flip flop, una forma de llamar a los biestables activados por flanco

La información de bits no se pierde; está rotando en el registro

Para que el siguiente circuito sea un registro de rotación basta con conectar la salida  $Q_H$  a la entrada A y poner a 1 la entrada SER.  $SH/\overline{LD}$  habrá que ponerlo primero a 0 para cargar los datos, y luego a 1

logic diagram (positive logic)



Pin numbers shown are for the D, DB, DGV, J, NS, PW, and W packages.

- **Contador digital**

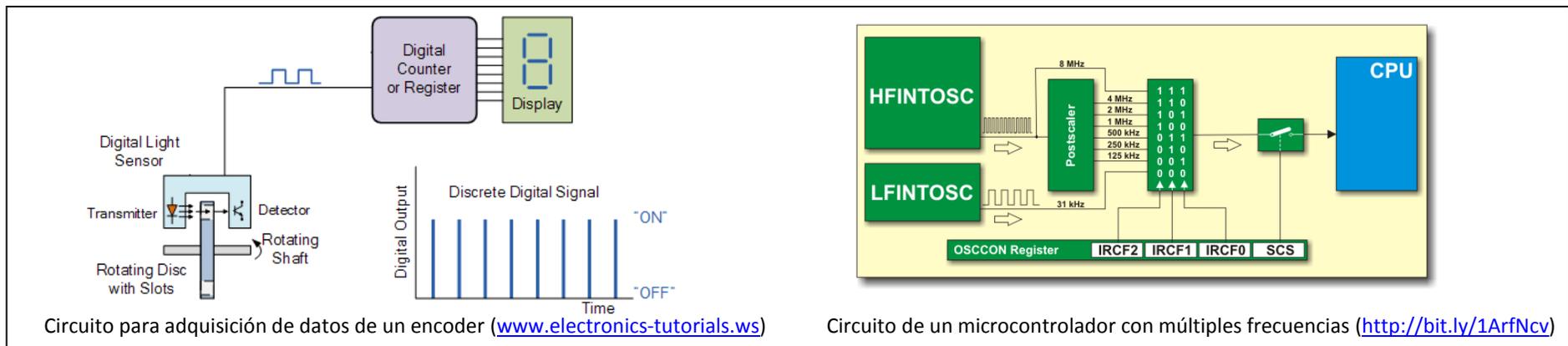
Es un circuito que recuerda el número de impulsos que le ha llegado

El módulo N de un contador es el número de impulsos que puede contar (ojo! si un contador cuenta de 0 a 5, no es módulo 5 sino módulo 6 ya que desde el 0 hasta el 5 van 6 números)

**Aplicaciones:**

División de frecuencia: en circuitos electrónicos es habitual emplear más de una señal de reloj, que se obtiene mediante un oscilador. Los osciladores son caros, por lo que resulta más económico obtener la señal de mayor frecuencia mediante un oscilador y mediante divisores de frecuencia generar el resto de señales de reloj de frecuencia más baja

Adquisición de datos: si cada impulso que recibe el contador corresponde a la detección de un objeto por parte de un sensor, se pueden obtener datos del número de detecciones realizadas



Temporización: si se cuenta el número de impulsos de una señal de reloj y se conoce la frecuencia del reloj, es posible calcular el tiempo que ha transcurrido cada vez que el contador pasa por 0. Mediante un diseño adecuado (frecuencia de la señal de entrada y módulo del contador) se pueden obtener una alarma que avisa de que a transcurrido un tiempo concreto.

- **Contadores asíncronos**

Al igual que con los circuitos de registros, para los contadores también se emplean biestables.

Los primeros que veremos son los asíncronos, que se caracterizan por que sus biestables no emplean la misma señal de reloj

Los contadores asíncronos se pueden obtener mediante distintos tipos de biestables. Nosotros veremos un método basado en biestables JK:

Paso 1:

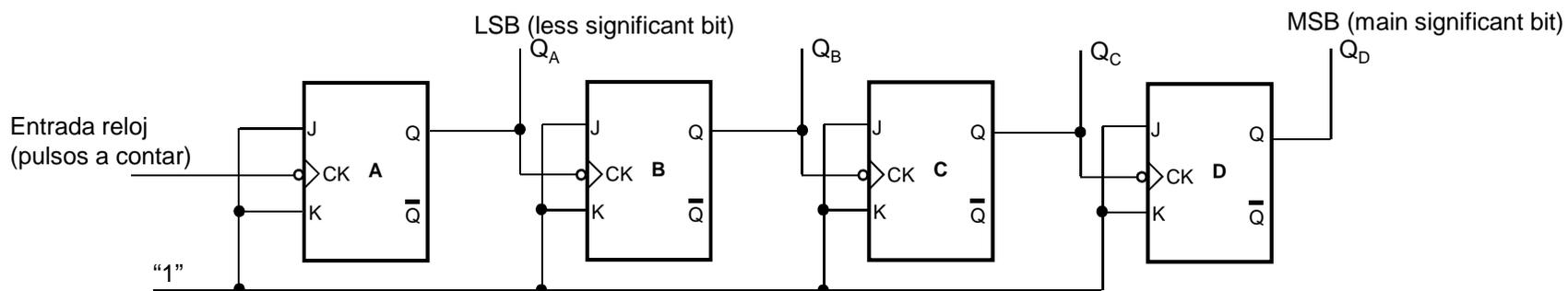
Determinar el número  $n$  de biestables necesarios para obtener el módulo  $N$ :  $2^{n-1} < N \leq 2^n$  (ejemplo: para un módulo 16 hacen falta 4 biestables ya que  $2^4=16$ )

Paso 2:

Se diseña un circuito contador módulo  $2^n$ , (si el módulo que se desea obtener es 16,  $2^n$  es 16, pero si el módulo que se desea obtener es 10, también  $2^n$  será 16)

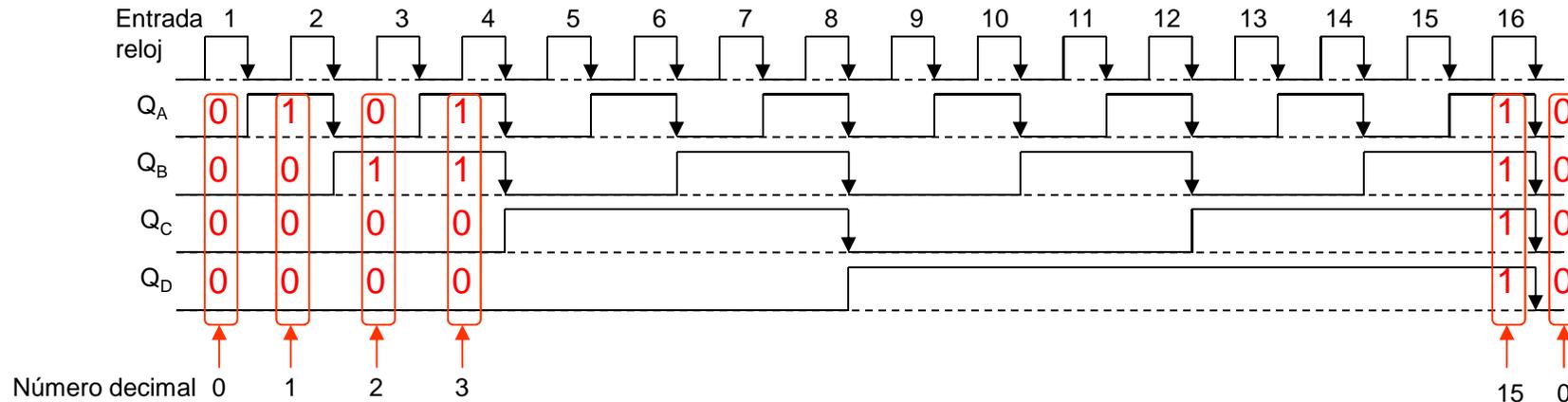
Paso 3:

Se conectan los biestables en cascada (cada biestable toma su señal de reloj de la salida del anterior biestable). El siguiente circuito es un módulo 16:



## Contadores asíncronos

Como cada biestable del circuito anterior tiene entradas  $J=K=1$  (igual a un biestable T con entrada  $T=1$ ), estos cambian su estado con cada flanco de bajada de reloj:



El circuito cuenta en binario natural hasta  $2^4=16$  impulsos, siendo 4 el número de biestables en cadena:

	$Q_D$	$Q_C$	$Q_B$	$Q_A$	Número decimal
Inicialmente	0	0	0	0	0
Después del impulso 1	0	0	0	1	1
Después del impulso 2	0	0	1	0	2
Después del impulso 3	0	0	1	1	3
Después del impulso 4	0	1	0	0	4
Después del impulso 5	0	1	0	1	5
Después del impulso 6	0	1	1	0	6
Después del impulso 7	0	1	1	1	7
Después del impulso 8	1	0	0	0	8
Después del impulso 9	1	0	0	1	9
Después del impulso 10	1	0	1	0	10
Después del impulso 11	1	0	1	1	11
Después del impulso 12	1	1	0	0	12
Después del impulso 13	1	1	0	1	13
Después del impulso 14	1	1	1	0	14
Después del impulso 15	1	1	1	1	15
Después del impulso 16	0	0	0	0	0

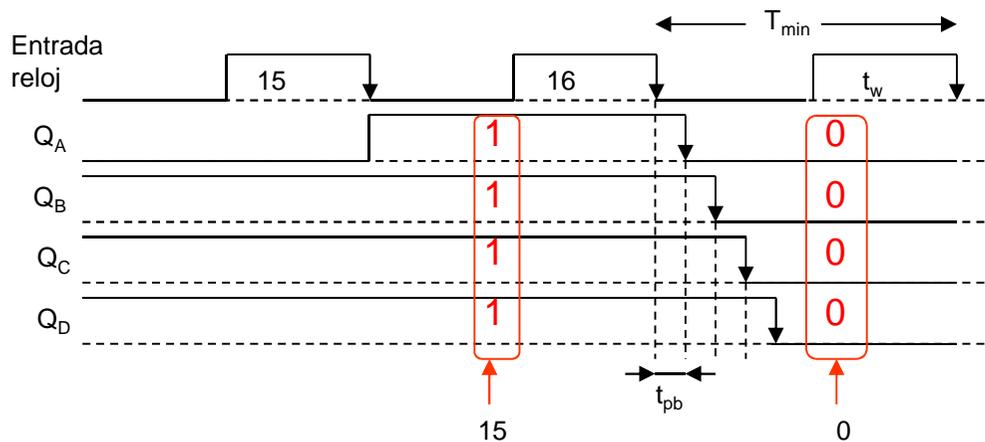
Si se compara  $Q_A$  con  $Q_B$ , el periodo de  $Q_B$  es el doble que el de  $Q_A$ . Lo mismo sucede entre  $Q_B$  y  $Q_C$ , y entre  $Q_C$  y  $Q_D$ .

Esto es porque la frecuencia de reloj se divide por 2 a la salida de cada biestable

- Contadores asíncronos**

En la práctica los biestables tienen retardos, lo que provoca que la conmutación del circuito, al estar conectados los biestables en cascada, sea secuencial:

De modo que la frecuencia máxima de la entrada de reloj del contador vendrá determinada por la transición más larga (cuando conmutan todos los biestables), que en el caso del contador de módulo 16 ocurrirá en dos casos: de 0111 a 1000 y de 1111 a 0000. Cogemos la segunda opción:



Si los JK son maestro/esclavo el  $t_{set-up} = t_w$  (ancho del pulso de reloj)

El  $t_{hold}$  se considera despreciable y el tiempo de propagación o retardo del biestable será  $t_{pb}$

El periodo más corto que puede tener la entrada de reloj será entonces:

$$T_{min} = n \cdot t_{pb} + t_{set-up} \quad (n \text{ es el número de biestables})$$

$$\text{Con lo que para 4 biestables: } T_{min} = 4 \cdot t_{pb} + t_{set-up} \Rightarrow f_{max} = 1/T_{min}$$

- **Contadores asíncronos**

Paso 4: hasta ahora se ha resuelto la parte de diseño de un contador módulo  $2^n$ , donde  $n$  es el número de biestables. Pero cuando el módulo del contador no es  $2^n$ , hace falta un paso adicional. Si por ejemplo se desea diseñar un contador módulo 10, se utilizará el contador módulo 16 del paso 3 y se obtendrá a partir de él el módulo 10.

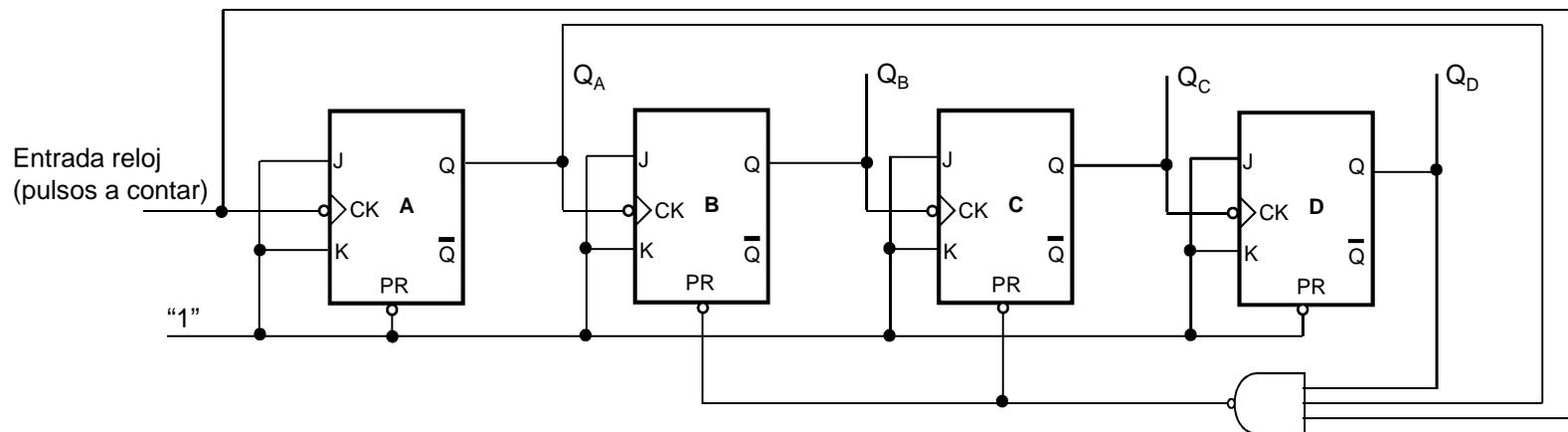
a) Se parte del último estado del contador ( $N-1$ ), en este caso  $10-1=9$

b) Se conectan las salidas de los biestables con  $Q=1$  y la entrada de impulsos a una puerta NAND

c) Se conecta la salida de la NAND a las PRESET de los biestables con  $Q=0$

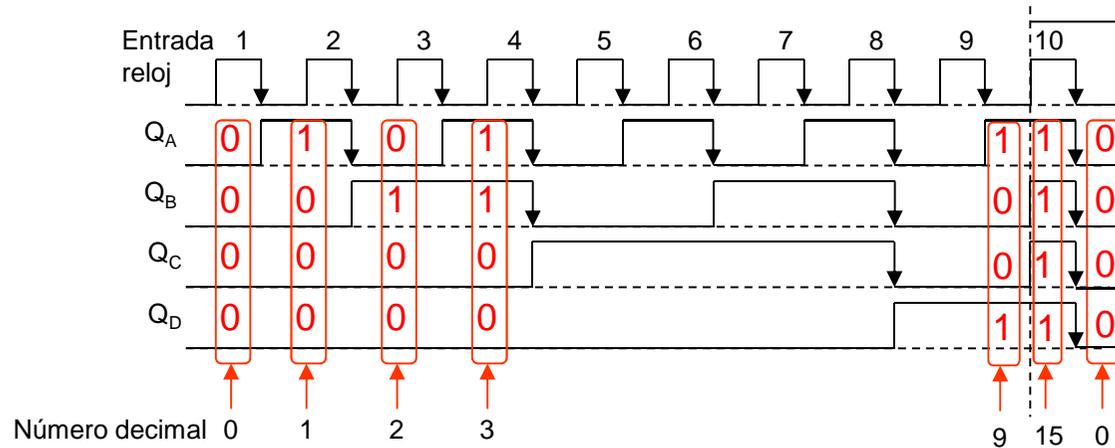
d) Se conecta a 1 los PRESET de los biestables con  $Q=1$

Así, en la subida del siguiente impulso de reloj los biestables conmutan a 1 y en la bajada conmutan a 0



## • Contadores asíncronos

El cronograma del contador asíncrono módulo 10 sin contar los tiempos de propagación es:



La salida de la puerta NAND se hace 0 y los biestables quedan a "1"

Durante el décimo impulso cuenta el 15, que es un número no deseado, lo que se soluciona en parte reduciendo el ancho de los pulsos

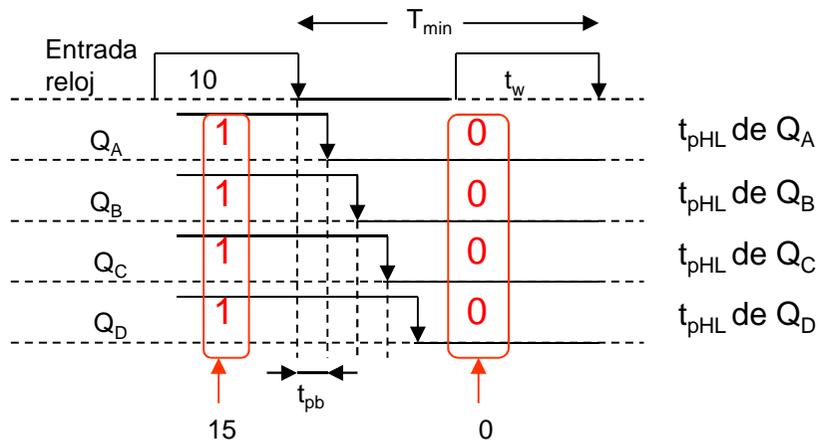
Diagrama de tiempos sin contar los retardos de propagación

	Q <sub>D</sub>	Q <sub>C</sub>	Q <sub>B</sub>	Q <sub>A</sub>	Número decimal
Inicialmente	0	0	0	0	0
Después del impulso 1	0	0	0	1	1
Después del impulso 2	0	0	1	0	2
Después del impulso 3	0	0	1	1	3
Después del impulso 4	0	1	0	0	4
Después del impulso 5	0	1	0	1	5
Después del impulso 6	0	1	1	0	6
Después del impulso 7	0	1	1	1	7
Después del impulso 8	1	0	0	0	8
Después del impulso 9	1	0	0	1	9
Después del impulso 10	0	0	0	0	0

## Inconvenientes de los contadores asíncronos

Analizando el contador módulo 10 se indican a continuación los problemas comunes de todos los contadores asíncronos

a) Igual que en el contador módulo 16, al producirse la transición en el flanco de bajada del décimo impulso todos los biestables conmutan a la vez y el retardo de cada biestable se acumula reduciendo la máxima frecuencia de trabajo del circuito.



b) Además el contador pasa por estados transitorios no deseados

	$Q_D$	$Q_C$	$Q_B$	$Q_A$	Número decimal
Después del flanco de bajada del impulso 9	1	0	0	1	9
Después de NAND = 0	1	1	1	1	15
Después de flanco de bajada de $Q_A$	1	1	1	0	14
Después de flanco de bajada de $Q_B$	1	1	0	0	12
Después de flanco de bajada de $Q_C$	1	0	0	0	8
Después de flanco de bajada de $Q_D$	0	0	0	0	0

Estados no deseados

c) Las transiciones no tardan el mismo tiempo y por eso hay que considerar la transición más larga para estar seguro de que el contador está en un estado estable

$Q_D$	$Q_C$	$Q_B$	$Q_A$		
0	0	0	0	→	0 0 0 1
0	0	1	1	→	0 1 0 0

→ 1 cambio     $1 t_{pLH}$   
 → 3 cambios     $1 t_{pLH} + 2 t_{pHL}$

- Contadores síncronos**

Intentan resolver los problemas de los contadores asíncronos

La señal de los impulsos se aplica a la vez a todos los biestables

Los biestables conmutan simultáneamente

Las entradas síncronas se utilizan para pasar de un estado a otro

La frecuencia de los impulsos o de utilización del contador es más alta que en los asíncronos.

**Método de diseño**

1) Calcular el número  $n$  de biestables necesarios: para un contador módulo  $N$ :  $2^{n-1} < N \leq 2^n$

2) Sacar la tabla de verdad del contador:

entradas: estados por los que ha de pasar el contador y estados siguientes después del impulso

salidas: excitaciones que se han de presentar en los biestables (en las entradas síncronas) para provocar el cambio de estado requerido

Tabla de transiciones biestable RS				Tabla de transiciones biestable JK				Tabla de transiciones biestable T			Tabla de transiciones biestable D		
$Q_t$	$Q_{t+1}$	S	R	$Q_t$	$Q_{t+1}$	J	K	$Q_t$	$Q_{t+1}$	T	$Q_t$	$Q_{t+1}$	D
0	0	0	X	0	0	0	X	0	0	0	0	0	0
0	1	1	0	0	1	1	X	0	1	1	0	1	1
1	0	0	1	1	0	X	1	1	0	1	1	0	0
1	1	X	0	1	1	X	0	1	1	0	1	1	1

3) Calcular, a partir de la tabla de verdad, el conexionado de las entradas síncronas de los biestables y de las salidas de las mismas

4) implementar el circuito

## Contador síncrono módulo 16

1) Módulo  $N=16$  (16 estados)  $\Rightarrow 2^{n-1} < 16 \leq 2^n \Rightarrow 4$  biestables

2) La tabla de verdad del contador (usando biestables JK por flanco de bajada)

Tabla de transiciones								Señales de control							
Estado actual (t)				Estado siguiente (t+1)				Entradas síncronas (t)							
$Q_D$	$Q_C$	$Q_B$	$Q_A$	$Q_D$	$Q_C$	$Q_B$	$Q_A$	$J_D$	$K_D$	$J_C$	$K_C$	$J_B$	$K_B$	$J_A$	$K_A$
0	0	0	0	0	0	0	1	0	X	0	X	0	X	1	X
0	0	0	1	0	0	1	0	0	X	0	X	1	X	X	1
0	0	1	0	0	0	1	1	0	X	0	X	X	0	1	X
0	0	1	1	0	1	0	0	0	X	1	X	X	1	X	1
0	1	0	0	0	1	0	1	0	X	X	0	0	X	1	X
0	1	0	1	0	1	1	0	0	X	X	0	1	X	X	1
0	1	1	0	0	1	1	1	0	X	X	0	X	0	1	X
0	1	1	1	1	0	0	0	1	X	X	1	X	1	X	1
1	0	0	0	1	0	0	1	X	0	0	X	0	X	1	X
1	0	0	1	1	0	1	0	X	0	0	X	1	X	X	1
1	0	1	0	1	0	1	1	X	0	0	X	X	0	1	X
1	0	1	1	1	1	0	0	X	0	1	X	X	1	X	1
1	1	0	0	1	1	0	1	X	0	X	0	0	X	1	X
1	1	0	1	1	1	1	0	X	0	X	0	1	X	X	1
1	1	1	0	1	1	1	1	X	0	X	0	X	0	1	X
1	1	1	1	0	0	0	0	X	1	X	1	X	1	X	1

Tomando como base la tabla de transiciones del biestable JK

Tabla de transiciones biestable JK			
$Q_t$	$Q_{t+1}$	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

3) Simplificando por Karnaugh cada una de las salidas:

Nota: las entradas del mapa de Karnaugh son las del estado actual, no las del siguiente

$J_D$	$Q_D$	$Q_B$	$Q_C$	00	01	11	10	
$Q_A$	00	0	4	0	12	X	8	X
01	1	0	5	0	13	X	9	X
11	3	0	7	1	15	X	11	X
10	2	0	6	0	14	X	10	X

$$J_D = Q_A \cdot Q_B \cdot Q_C$$

El resto de salidas son:

$$K_D = Q_A \cdot Q_B \cdot Q_C$$

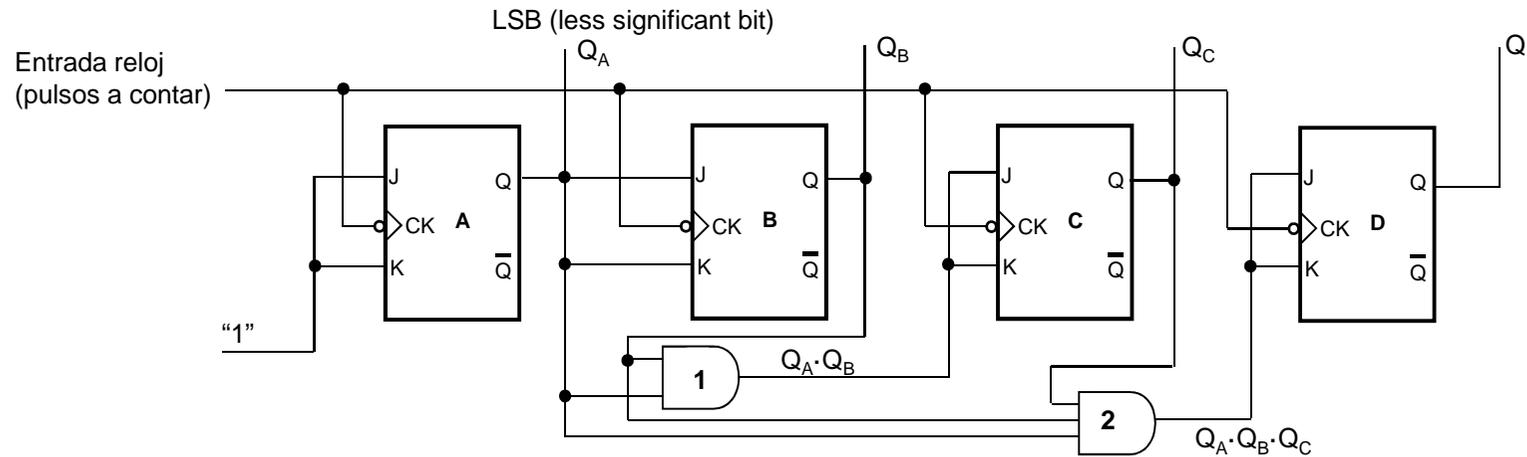
$$J_C = K_C = Q_A \cdot Q_B$$

$$J_B = K_B = Q_A$$

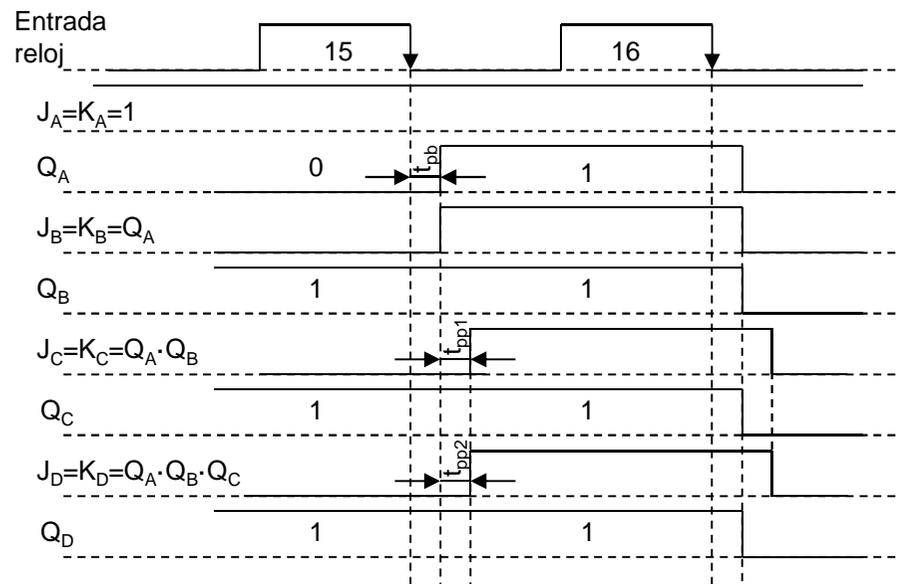
$$J_A = K_A = 1$$

## Contador síncrono módulo 16

### 4) Implementación del circuito (versión con acarreo paralelo)



### Cronograma (todos los biestables conmutan simultáneamente):



$t_{pp1}$  = tiempo de propagación de la puerta AND 1

$t_{pp2}$  = tiempo de propagación de la puerta AND 2

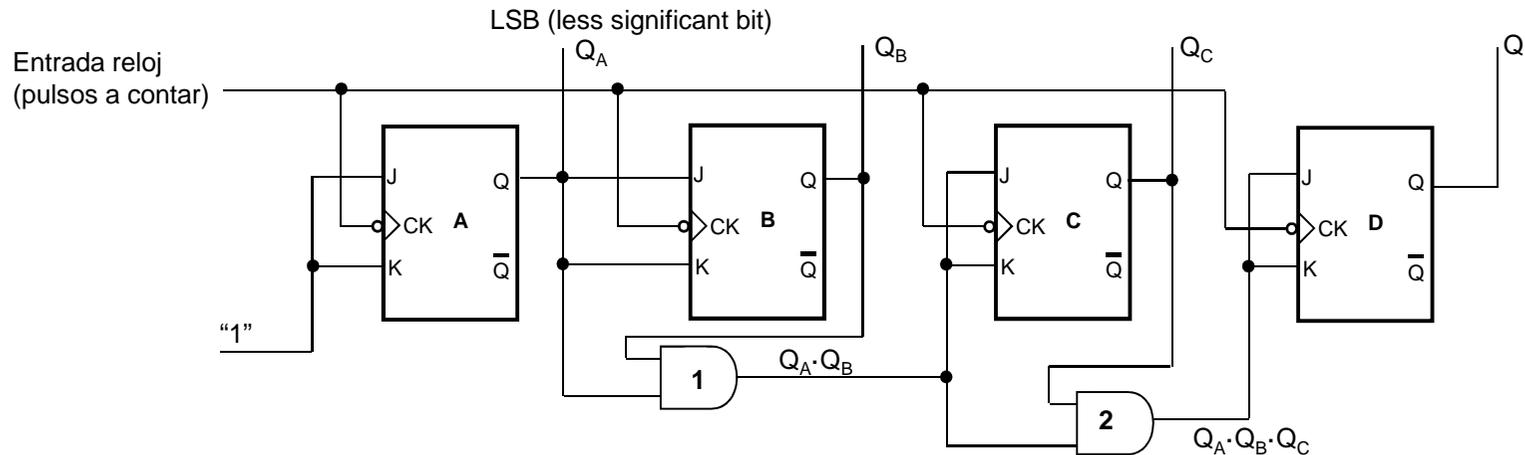
$t_{pb}$  = tiempo de propagación del biestable JK

$$T_{\min} = t_{pb} + t_{pp1} + t_{pp2} \Rightarrow f_{\max} = 1/T_{\min}$$

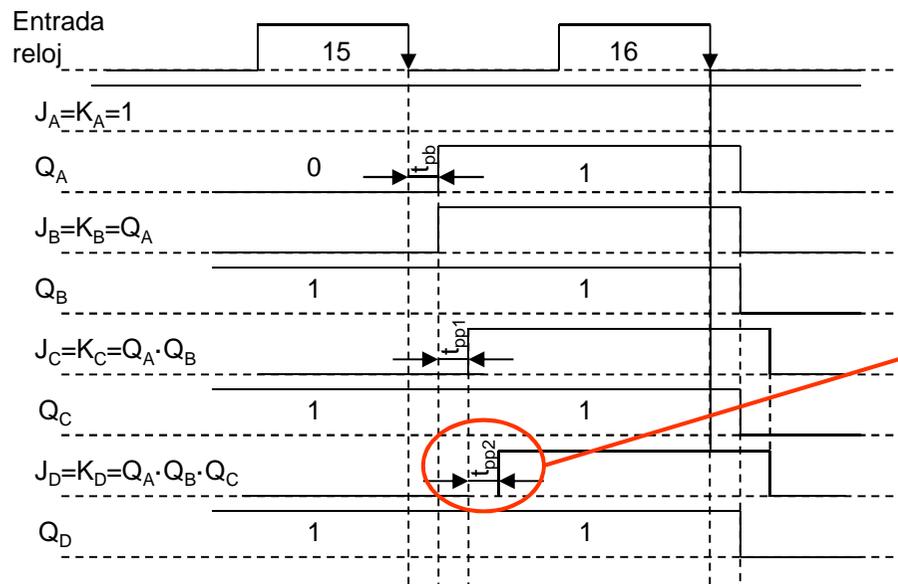
**El periodo mínimo es más pequeño que el del contador asíncrono  
Por eso las frecuencia máxima de trabajo es mayor**

## Contador síncrono módulo 16

### 4) Implementación del circuito (versión con acarreo serie)



### Cronograma (todos los biestables conmutan simultáneamente):



$t_{pp1}$  = tiempo de propagación de la puerta AND 1

$t_{pp2}$  = tiempo de propagación de la puerta AND 2

$t_{pb}$  = tiempo de propagación del biestable JK

$$T_{min} = t_{pb} + 2t_{pp} + t_{setup} \Rightarrow f_{max} = 1/T_{min}$$

Con el acarreo serie se evita usar puertas de más de 2 entradas, pero el biestable D responde con 1  $t_{pp}$  más de retardo que con el acarreo paralelo.

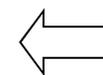
Aún así, el periodo mínimo es más pequeño que el del contador asíncrono (la frecuencia máxima de trabajo es mayor)

## Contador síncrono módulo 10

1) Módulo  $N=10$  (10 estados)  $\Rightarrow 2^{n-1} < 10 \leq 2^n \Rightarrow 4$  biestables

2) La tabla de verdad del contador (usando biestables JK por flanco de bajada)

Tabla de transiciones								Señales de control							
Estado actual (t)				Estado siguiente (t+1)				Entradas síncronas (t)							
$Q_D$	$Q_C$	$Q_B$	$Q_A$	$Q_D$	$Q_C$	$Q_B$	$Q_A$	$J_D$	$K_D$	$J_C$	$K_C$	$J_B$	$K_B$	$J_A$	$K_A$
0	0	0	0	0	0	0	1	0	X	0	X	0	X	1	X
0	0	0	1	0	0	1	0	0	X	0	X	1	X	X	1
0	0	1	0	0	0	1	1	0	X	0	X	X	0	1	X
0	0	1	1	0	1	0	0	0	X	1	X	X	1	X	1
0	1	0	0	0	1	0	1	0	X	X	0	0	X	1	X
0	1	0	1	0	1	1	0	0	X	X	0	1	X	X	1
0	1	1	0	0	1	1	1	0	X	X	0	X	0	1	X
0	1	1	1	1	0	0	0	1	X	X	1	X	1	X	1
1	0	0	0	1	0	0	1	X	0	0	X	0	X	1	X
1	0	0	1	0	0	0	0	X	1	0	X	0	X	X	1



Tomando como base la tabla de transiciones del biestable JK



Tabla de transiciones biestable JK			
$Q_t$	$Q_{t+1}$	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

3) Simplificando por Karnaugh cada una de las salidas:

$J_B$	$Q_D$	$Q_C$	$Q_B$	$Q_A$	00	01	11	10
00	0	0	0	0	0	4	12	8
01	1	1	0	0	1	5	13	9
11	3	1	1	0	X	7	15	11
10	2	0	1	1	X	6	14	10

$$J_B = Q_A \cdot \overline{Q_D}$$

$K_B$	$Q_D$	$Q_C$	$Q_B$	$Q_A$	00	01	11	10
00	0	0	0	0	X	4	12	8
01	1	1	0	0	X	5	13	9
11	3	1	1	0	1	7	15	11
10	2	0	1	1	0	6	14	10

Aunque se puede hacer un lazo más grande, así  $J_B=K_B$  y el circuito es más sencillo

$$K_B = Q_A \cdot \overline{Q_D}$$

El resto de salidas son:

$$J_D = Q_A \cdot Q_B \cdot Q_C$$

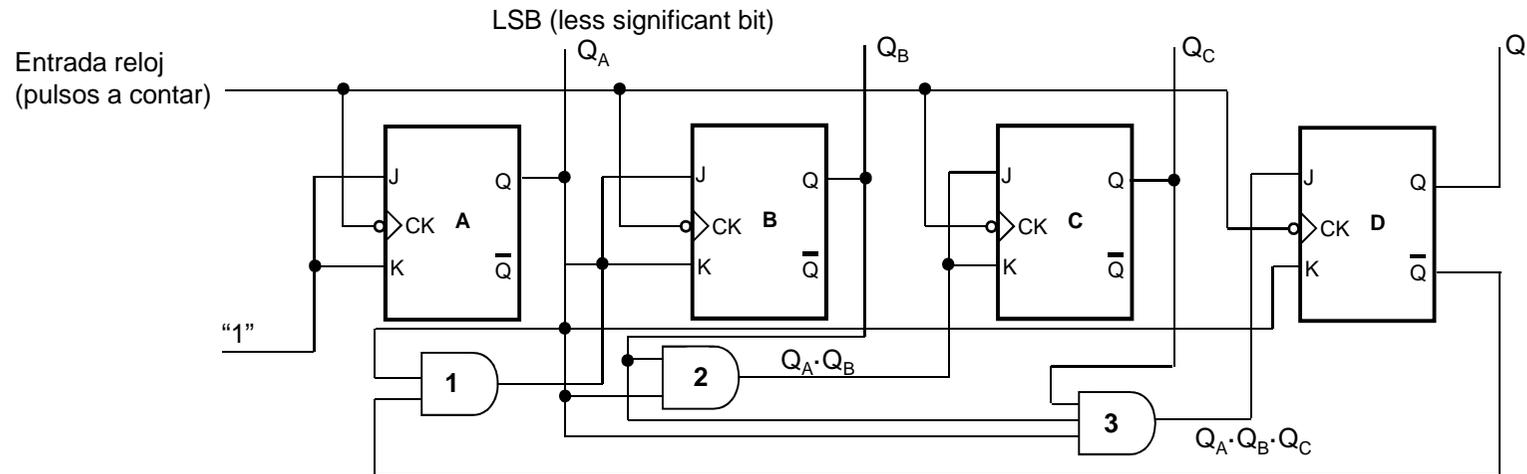
$$K_D = Q_A$$

$$J_C = K_C = Q_A \cdot Q_B$$

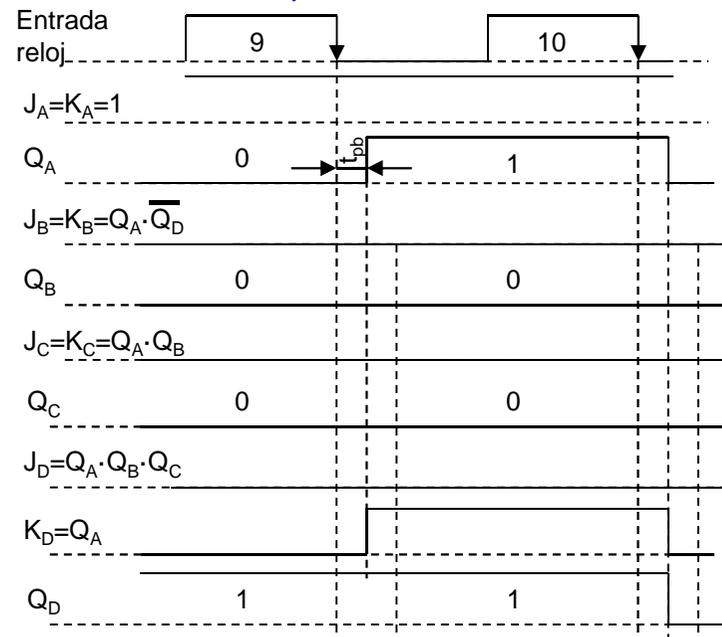
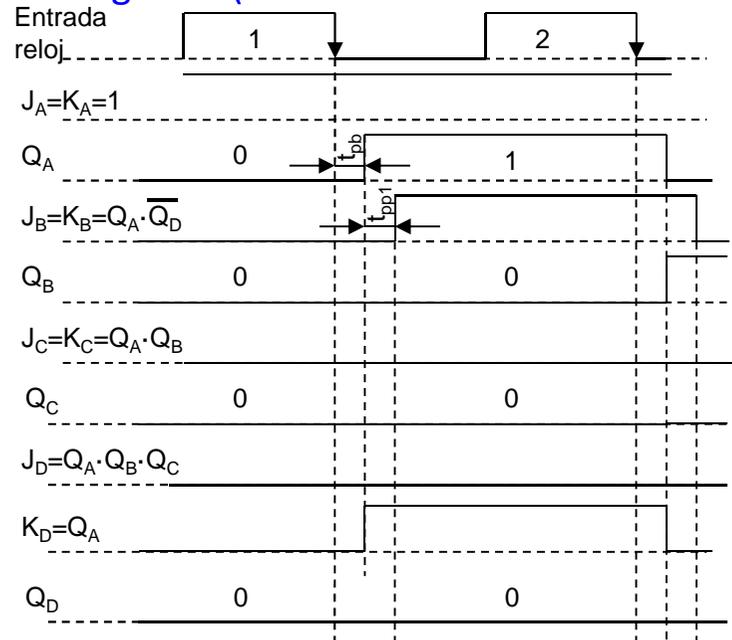
$$J_A = K_A = 1$$

## Contador síncrono módulo 10

### 4) Implementación del circuito (versión con acarreo paralelo)



### Cronograma (todos los biestables conmutan simultáneamente):



**Acarreo paralelo**



$$T_{\min} = t_{pb} + t_{pp} + t_{\text{setup}}$$



$$f_{\max} = 1/T_{\min}$$

- Contadores reversibles**

**Contadores binarios ascendentes:** cuentan en sentido creciente de los números (los vistos hasta ahora)

**Contadores binarios descendentes:** cuentan en sentido decreciente de los números. Descuentan:

Ascendente	Módulo 4	Descendente
0 – 00	Inicialmente	0 – 00
1 – 01	Tras primer impulso	3 – 11
2 – 10	Tras segundo impulso	2 – 10
3 – 11	Tras tercer impulso	1 – 01
0 – 00	Tras cuarto impulso	0 – 00

**Contadores reversibles (up and down counter):** cuenta y descuenta con el mismo circuito

Ejemplo: contador reversible módulo 6 con biestables JK disparados por flanco de bajada

1) Módulo N=6 (6 estados)       $2^{n-1} < 6 \leq 2^n$       3 biestables

2) La tabla de verdad del contador

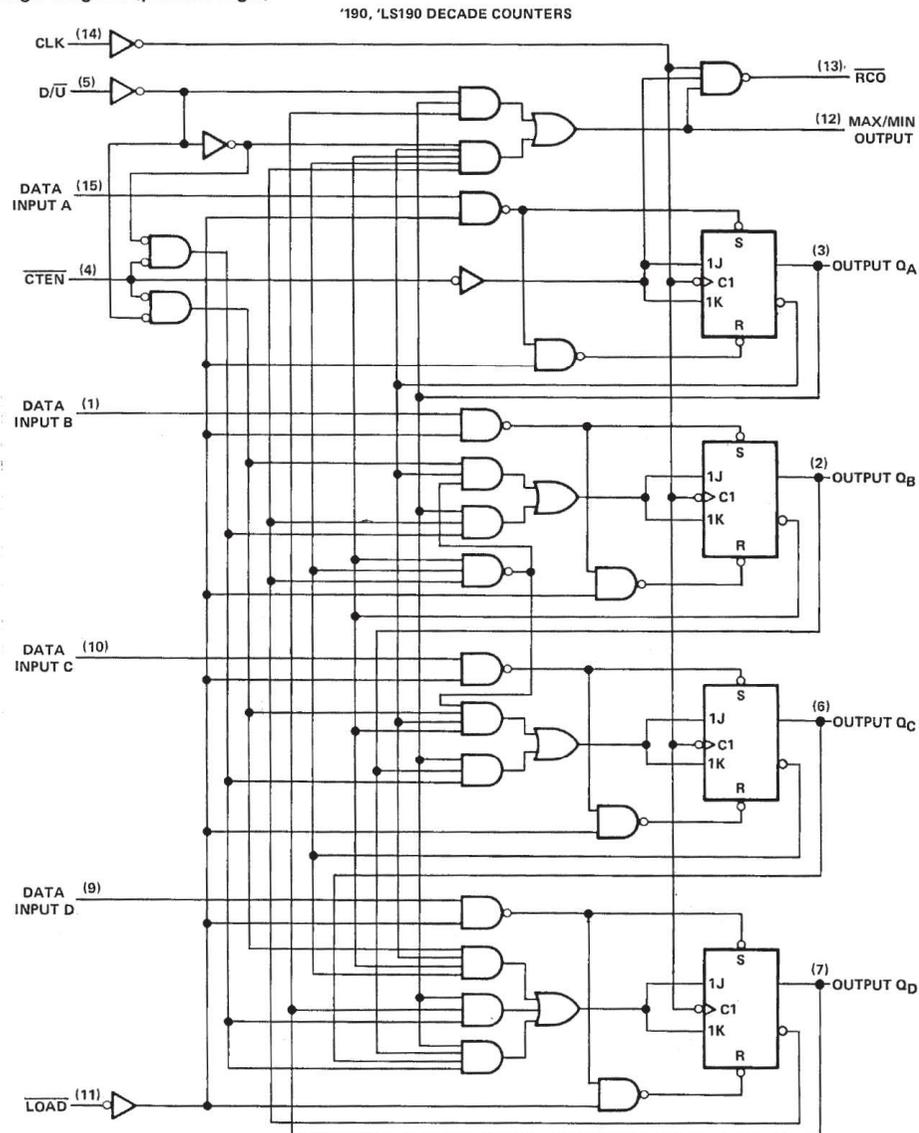
Entrada de control C	Tabla de transiciones						Señales de excitación					
	Estado actual (t)			Estado siguiente(t)			Entradas síncronas					
	Q <sub>C</sub>	Q <sub>B</sub>	Q <sub>A</sub>	Q <sub>C</sub>	Q <sub>B</sub>	Q <sub>A</sub>	J <sub>C</sub>	K <sub>C</sub>	J <sub>B</sub>	K <sub>B</sub>	J <sub>A</sub>	K <sub>A</sub>
0	0	0	0	1	0	1	1	X	0	X	1	X
0	0	0	1	0	0	0	0	X	0	X	X	1
0	0	1	0	0	0	1	0	X	X	1	1	X
0	0	1	1	0	1	0	0	X	X	0	X	1
0	1	0	0	0	1	1	X	1	1	X	1	X
0	1	0	1	1	0	0	X	0	0	X	X	1
1	0	0	0	0	0	1	0	X	0	X	1	X
1	0	0	1	0	1	0	0	X	1	X	X	1
1	0	1	0	0	1	1	0	X	X	0	1	X
1	0	1	1	1	0	0	1	X	X	1	X	1
1	1	0	0	1	0	1	X	0	0	X	1	X
1	1	0	1	0	0	0	X	1	0	X	X	1



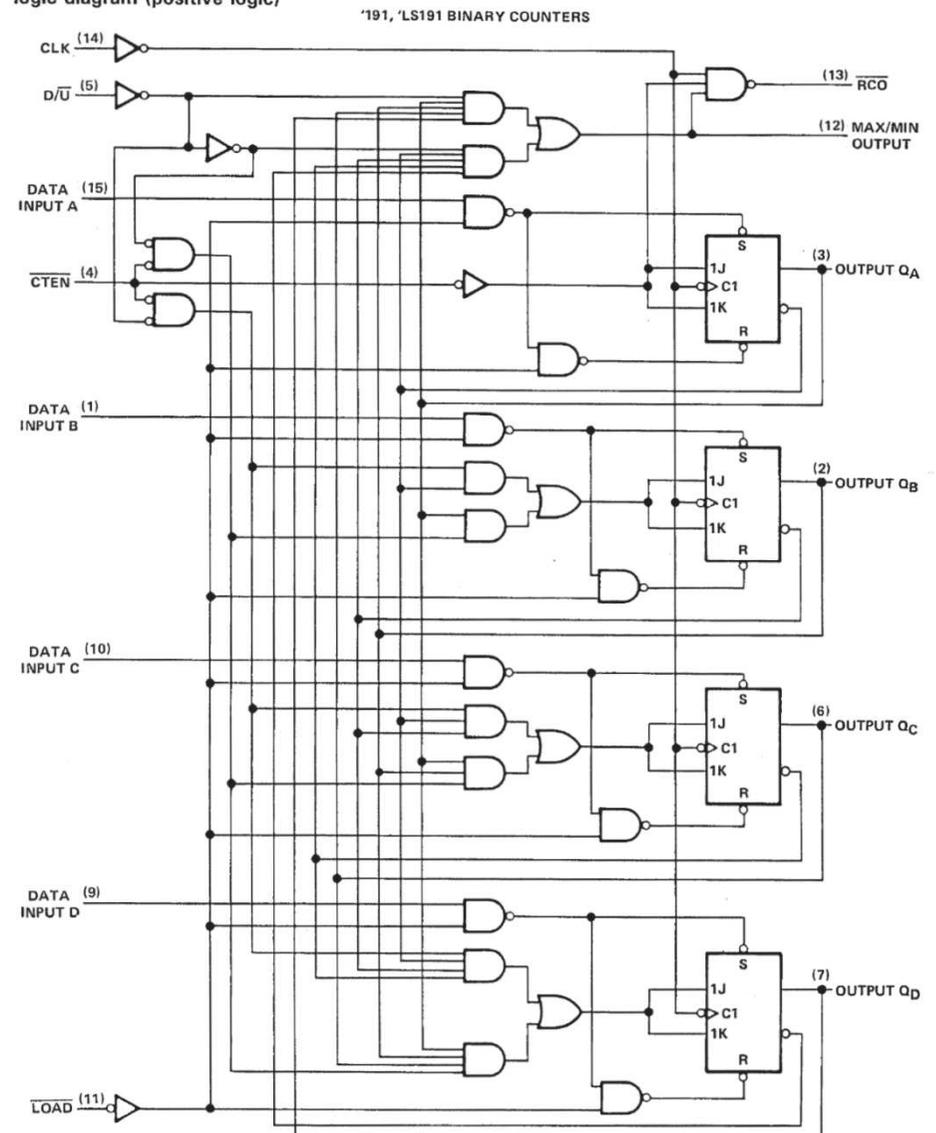
- Contadores reversibles comerciales

El SN74190 cuenta módulo 10 reversible y el SN74191 módulo 16 reversible (<http://bit.ly/1AMLWLK>)

logic diagram (positive logic)



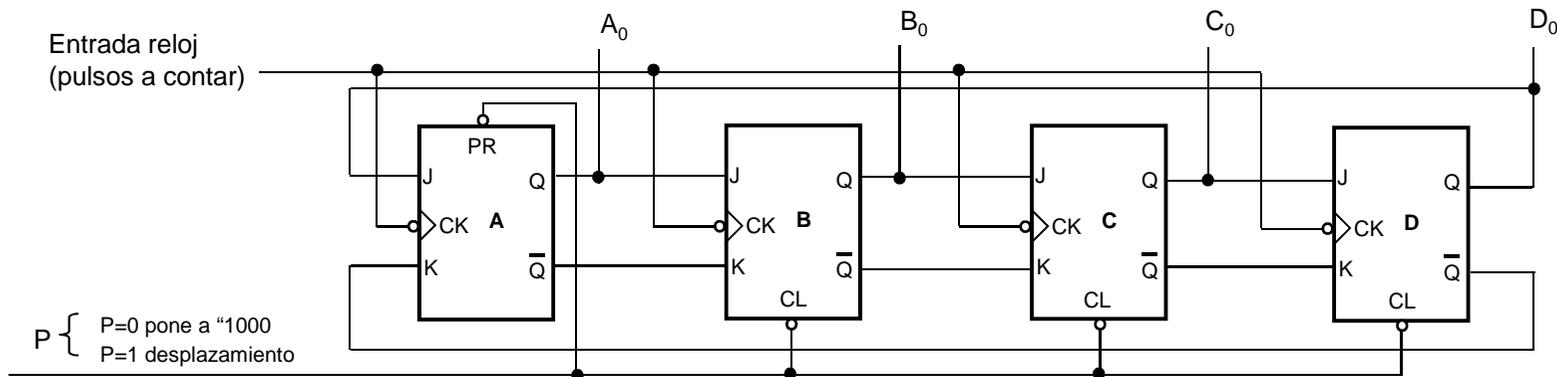
logic diagram (positive logic)



- **Contadores basados en registros de desplazamiento**

Son un tipo especial de contadores síncronos en los que no importa que no se cuente en binario natural y donde el número de códigos que se obtienen es el módulo del contador

a) Contador en anillo



Lo inicializamos con 1000 (para ello la entrada P=0)

Cada vez que llega un impulso el "1" se se desplaza a la derecha.

En el cuarto impulso el "1" vuelve al primer biestable y comienza el ciclo de nuevo

	A <sub>0</sub>	B <sub>0</sub>	C <sub>0</sub>	D <sub>0</sub>
Estado inicial	1	0	0	0
Después del primer impulso	0	1	0	0
Después del segundo impulso	0	0	1	0
Después del tercer impulso	0	0	0	1
Después del cuarto impulso	1	0	0	0

} Contador de módulo N=4

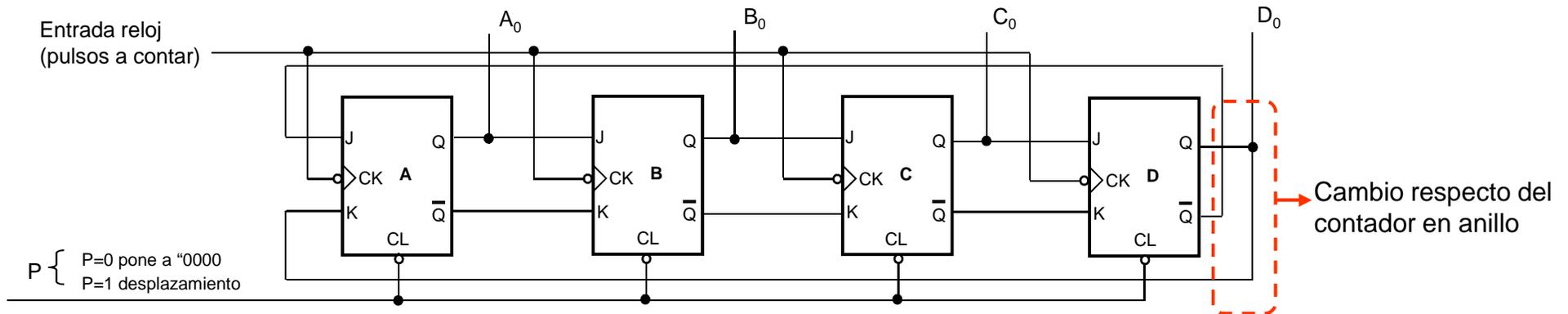
Para cada estado se necesita un biestable

La salida del contador ya está decodificada. Sólo aparece un "1" en las salidas (puede excitar un LED)

La posición del 1 indica el número de impulsos

- Contadores basados en registros de desplazamiento

- b) Contador Johnson (Möbius o anillo invertido)



Lo inicializamos con 0000 (para ello la entrada P=0)

Conforme llegan los impulsos se van poniendo los biestables a "1" y después a "0"

	A <sub>0</sub>	B <sub>0</sub>	C <sub>0</sub>	D <sub>0</sub>
Estado inicial	0	0	0	0
Después del primer impulso	1	0	0	0
Después del segundo impulso	1	1	0	0
Después del tercer impulso	1	1	1	0
Después del cuarto impulso	1	1	1	1
Después del quinto impulso	0	1	1	1
Después del sexto impulso	0	0	1	1
Después del séptimo impulso	0	0	0	1
Después del octavo impulso	0	0	0	0

Contador de módulo N=8  
El código que se obtiene es el código Johnson

Con el contador Johnson se obtiene el doble de estados que con el contador en anillo para el mismo número de biestables

Con n biestables: síncrono normal 2<sup>n</sup> estados, en anillo n estados y Johnson 2n estados diferentes

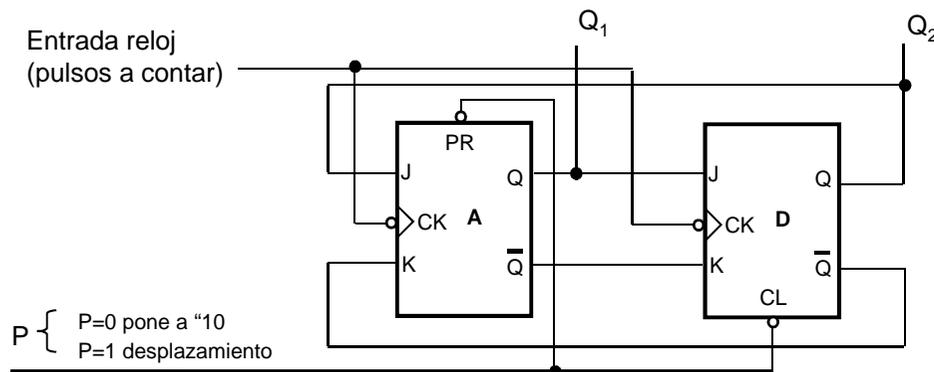
- Contadores basados en registros de desplazamiento

- c) Contador con protección (anti lockout)

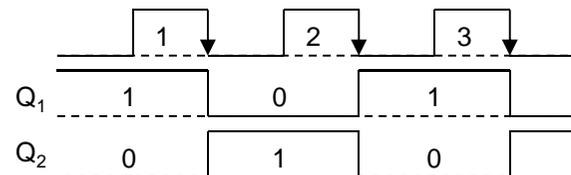
Los contadores que saltan estados (módulo  $N < 2^n$ ), a causa del ruido pueden caer en uno de los estados no deseados y no salir de él

Los contadores con protección son los que salen automáticamente de los estados no deseados

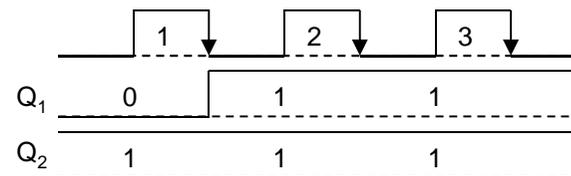
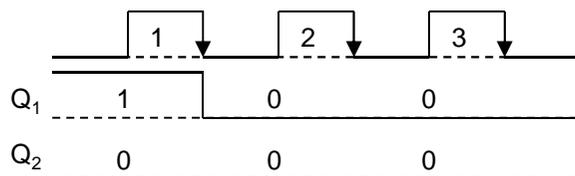
Ejemplo: Contador en anillo de módulo  $N=2$



	Q <sub>1</sub>	Q <sub>2</sub>
Estado inicial	1	0
Después del primer impulso	0	1
Después del segundo impulso	1	0



Si el contador cae en uno de los estados no deseados ("00", "11"), permanece en dicho estado indefinidamente



- Contadores basados en registros de desplazamiento

c) Contador con protección (anti lockout)

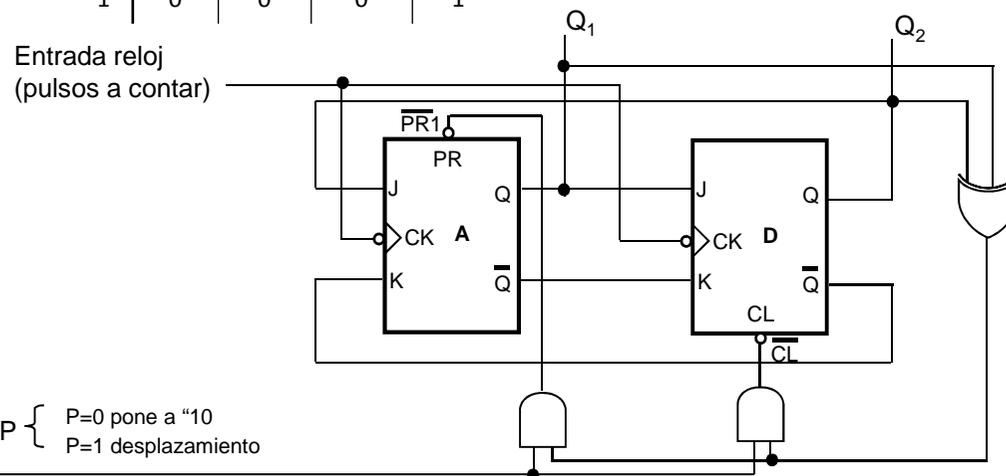
Mediante lógica adicional se consigue que el contador no se quede en el estado indeseado ("00" o "11") y se inicializa automáticamente a "10".

P	Estado actual		Estado siguiente		Entradas asíncronas		
	Q <sub>1</sub>	Q <sub>2</sub>	Q <sub>1</sub>	Q <sub>2</sub>	PR1	CL2	
0	0	0	1	0	0	0	} Inicialización manual
0	0	1	1	0	0	0	
0	1	0	1	0	0	0	
0	1	1	1	0	0	0	
1	0	0	1	0	0	0	→ Inicialización automática
1	0	1	1	0	1	1	} No actúan las entradas asíncronas
1	1	0	0	1	1	1	
1	1	1	1	0	0	0	→ Inicialización automática

$$\overline{PR1} = \overline{CL2}$$

P \ Q <sub>2</sub> \ Q <sub>1</sub>	00	01	11	10
0	0	2	6	4
1	1	3	7	5

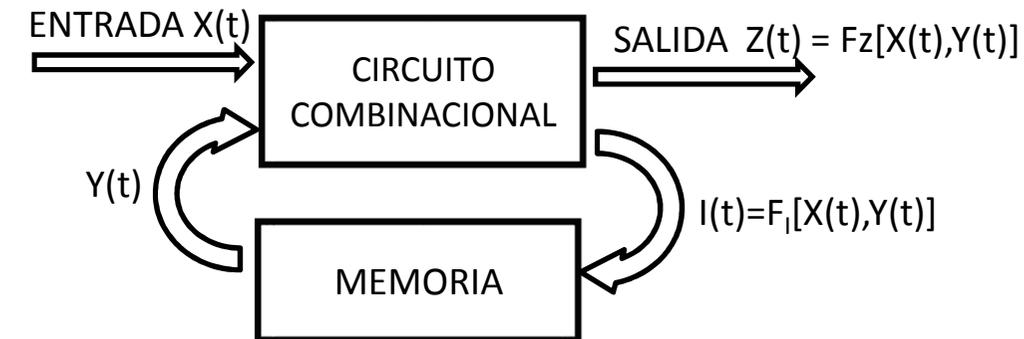
$$\overline{PR1} = \overline{CL2} = P \cdot Q_1 \cdot \overline{Q_2} + P \cdot \overline{Q_1} \cdot Q_2 = P \cdot (Q_1 \oplus Q_2)$$



# Análisis y diseño de circuitos secuenciales síncronos

## 0. Recordatorio

**Sistemas secuenciales:** La información de salida es función de las entradas actuales y de la situación (historia) del sistema



La memoria almacena la situación del sistema

**Sistemas secuenciales síncronos:** El contenido de los elementos de memoria sólo puede cambiar en presencia de pulsos de reloj

Hasta ahora se ha visto unos par de tipos concretos de circuitos secuenciales síncronos, que son los contadores síncronos y los registros de desplazamiento.

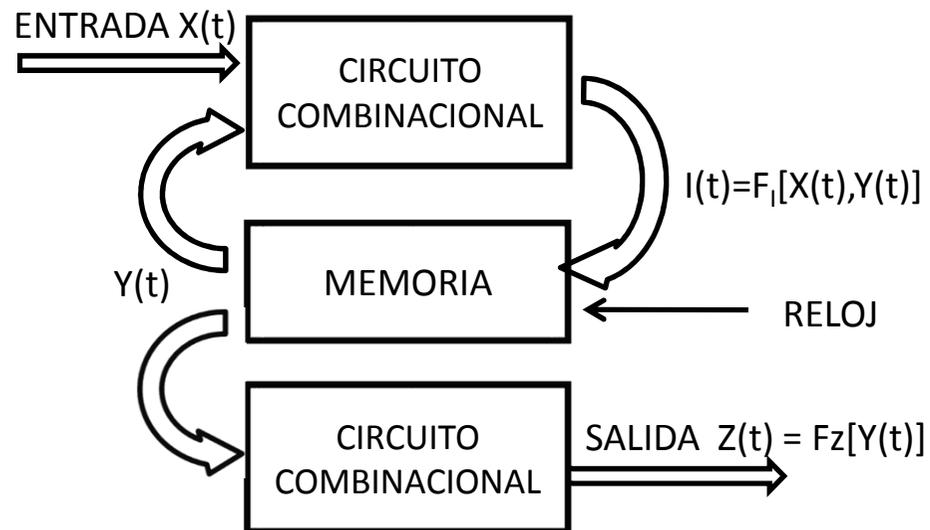
A continuación se explicarán las máquinas de Moore y de Mealy, que permiten implementar cualquier tipo de circuito secuencial síncrono

# Análisis y diseño de circuitos secuenciales síncronos

## 1. Introducción

**Máquina de Mealy:** la salida actual es función de las entradas actuales y de la situación actual (estado actual) del sistema

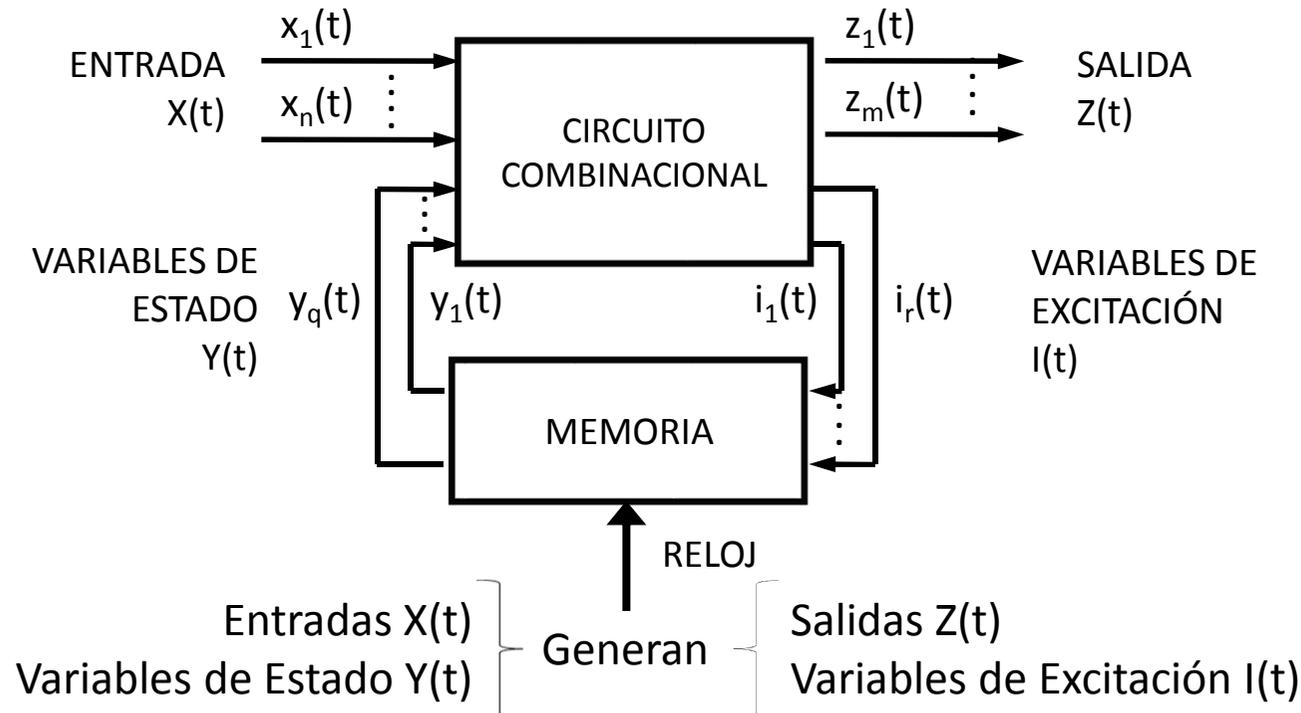
**Máquina de Moore:** la salida actual sólo es función de la situación actual (estado actual) del sistema



# Análisis y diseño de circuitos secuenciales síncronos

## 2. Análisis de Circuitos Secuenciales Síncronos

Modelo general de un sistema secuencial síncrono:

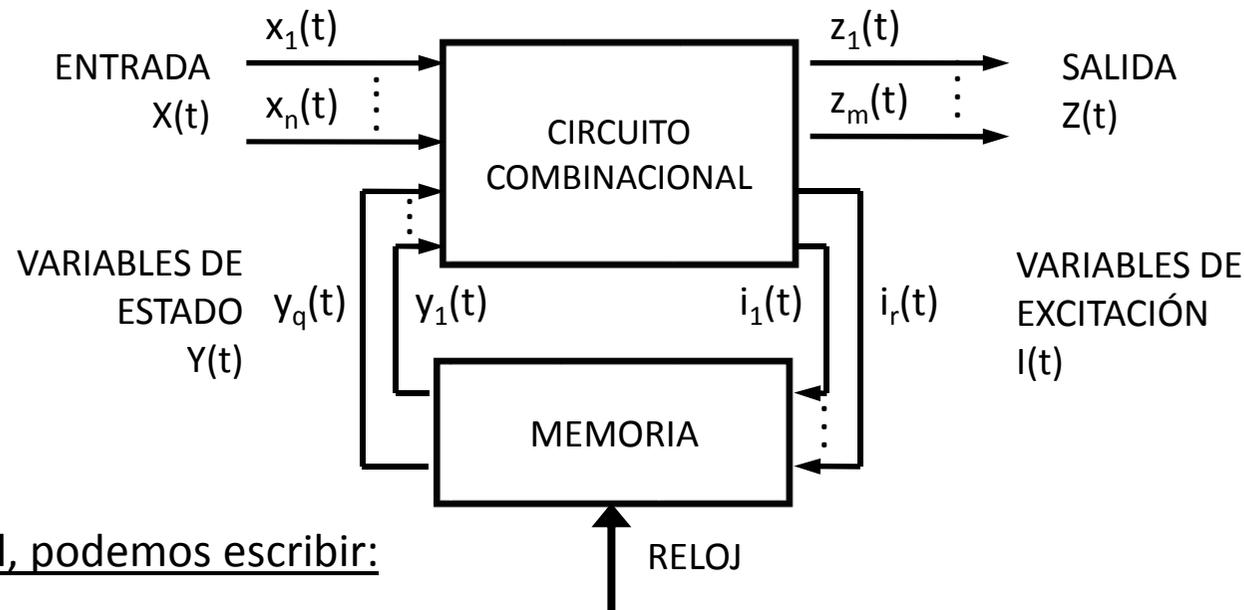


- Las señales de excitación  $I(t)$  definen el estado (siguiente) que almacenará la memoria cuando llegue el pulso de reloj
- La memoria va almacenando los estados o situaciones por las que va pasando el sistema

# Análisis y diseño de circuitos secuenciales síncronos

## 2. Análisis de Circuitos Secuenciales Síncronos

Modelo general de un sistema secuencial síncrono:



En general, podemos escribir:

$$\text{Entradas } X(t) = [x_1(t), x_2(t), \dots, x_n(t)]$$

$$\text{Estado actual } Y(t) = [y_1(t), y_2(t), \dots, y_q(t)]$$

$$\text{Salidas } Z(t) = [z_1(t), z_2(t), \dots, z_m(t)] = F_Z[X(t), Y(t)]$$

$$z_j(t) = f_j[x_1(t), x_2(t), \dots, x_n(t), y_1(t), y_2(t), \dots, y_q(t)]$$

$$\text{Excitaciones } I(t) = [i_1(t), i_2(t), \dots, i_r(t)] = F_I[X(t), Y(t)]$$

$$i_j(t) = f_j[x_1(t), x_2(t), \dots, x_n(t), y_1(t), y_2(t), \dots, y_q(t)]$$

$$\text{Estado siguiente: } Y(t+1) = F_Y[I(t)] = F_Y[X(t), Y(t)]$$

# Análisis y diseño de circuitos secuenciales síncronos

## 2. a) Ecuaciones de Estado y de Salida

### Ejemplo 1

Ecuación de Salida:  $z(t) = x(t) \cdot y_1(t) \cdot \bar{y}_2(t)$   
(Máquina de Mealy)

Ecuaciones de excitación de los Biestables:

$$J_1(t) = \bar{x}(t) + \bar{y}_1(t) \cdot y_2(t)$$

$$K_1(t) = \bar{x}(t)$$

$$J_2(t) = x(t)$$

$$K_2(t) = y_2(t)$$

Ecuaciones del estado siguiente en un Biestable:

S(t)	R(t)	Q(t)	Q(t+1)
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	X
1	1	1	X

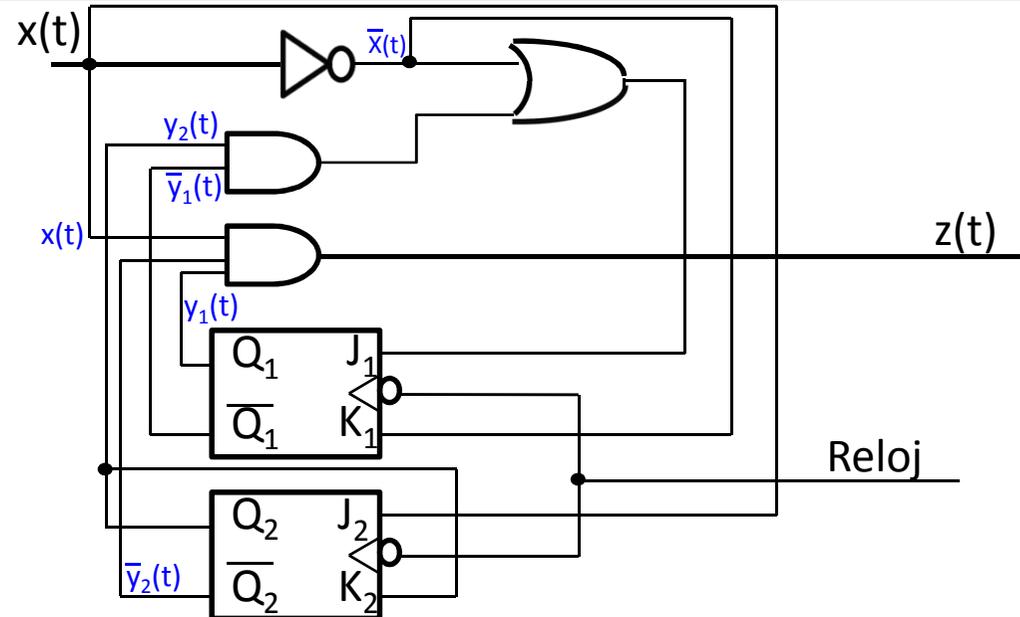
$$Q(t+1) = S(t) + \bar{R}(t) \cdot Q(t)$$

J(t)	K(t)	Q(t)	Q(t+1)
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

$$Q(t+1) = J(t) \cdot \bar{Q}(t) + \bar{K}(t) \cdot Q(t)$$

D(t)	Q(t)	Q(t+1)
0	0	0
0	1	0
1	0	1
1	1	1

$$Q(t+1) = D(t)$$



# Análisis y diseño de circuitos secuenciales síncronos

## 2. a) Ecuaciones de Estado y de Salida

### Ejemplo 1

Ecuación de Salida:  $z(t) = x(t) \cdot y_1(t) \cdot \bar{y}_2(t)$   
(Máquina de Mealy)

Ecuaciones de excitación de los Biestables:

$$\begin{aligned} J_1(t) &= \bar{x}(t) + \bar{y}_1(t) \cdot y_2(t) & K_1(t) &= \bar{x}(t) \\ J_2(t) &= x(t) & K_2(t) &= y_2(t) \end{aligned}$$

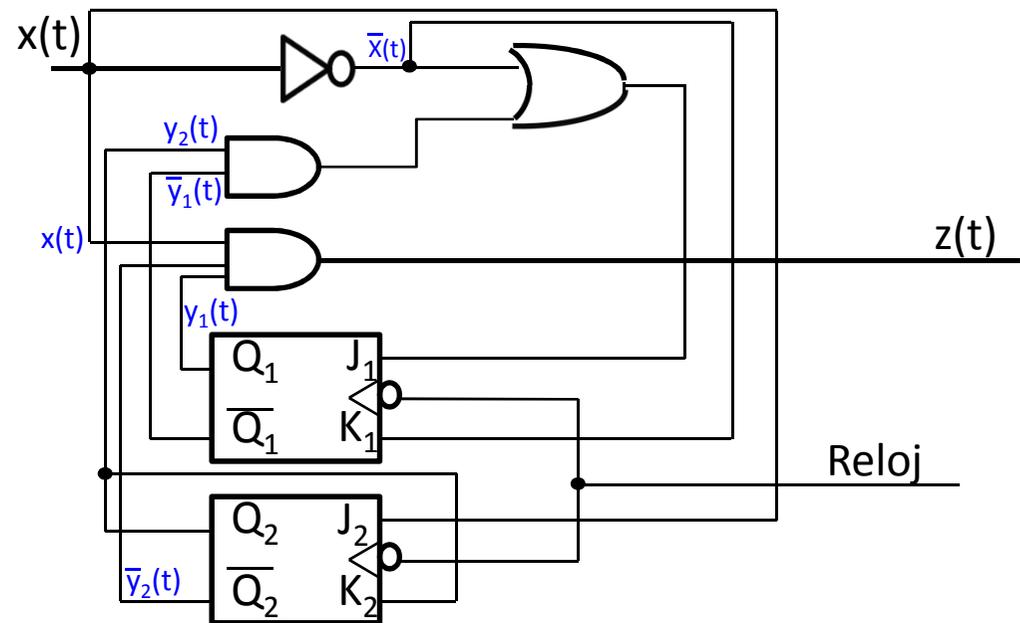
Ecuaciones del estado siguiente en un Biestable:

$$Q(t+1) = J(t) \cdot \bar{Q}(t) + \bar{K}(t) \cdot Q(t)$$

$$y_1(t+1) = J_1(t) \cdot \bar{y}_1(t) + \bar{K}_1(t) \cdot y_1(t) = [\bar{x}(t) + \bar{y}_1(t) y_2(t)] \bar{y}_1(t) + \bar{x}(t) y_1(t) = \bar{x}(t) \bar{y}_1(t) + \bar{y}_1(t) y_2(t) + x(t) y_1(t)$$

$$y_2(t+1) = J_2(t) \cdot \bar{y}_2(t) + \bar{K}_2(t) \cdot y_2(t) = x(t) \bar{y}_2(t) + \bar{y}_2(t) y_2(t) = x(t) \bar{y}_2(t)$$

- Estas ecuaciones caracterizan al sistema: podemos hallar directamente a partir del estado actual y de la entrada presente, cuál va a ser su siguiente estado y la salida actual



# Análisis y diseño de circuitos secuenciales síncronos

## 2. a) Ecuaciones de Estado y de Salida

### Ejemplo 1

Suponiendo que para  $t=0$  tenemos  $x(0)=0$   $y_1(0)=0$   $y_2(0)=0$

Entrada actual:  $X(0) = [x(0)] = [0]$

Estado actual:  $Y(0) = [y_1(0), y_2(0)] = [0, 0]$

Salida Actual:  $Z(0) = [z(0)] = x(0) \cdot y_1(0) \cdot \bar{y}_2(0) = [0]$

Excitaciones:  $I(0) = [i_1(0), i_2(0)] = [J_1(0), K_1(0), J_2(0), K_2(0)]$

$$i_1(0) = [\bar{x}(0) + \bar{y}_1(0)y_2(0), \bar{x}(0)] = [1, 1]$$

$$i_2(0) = [x(0), y_2(0)] = [0, 0]$$

Estado siguiente:  $y(1) = [y_1(1), y_2(1)]$

$$y_1(1) = \bar{x}(0)\bar{y}_1(0) + \bar{y}_1(0)y_2(0) + x(0)y_1(0) = 1$$

$$y_2(1) = x(0)\bar{y}_2(0) = 0$$

Resumen

$$z(t) = x(t) \cdot y_1(t) \cdot \bar{y}_2(t)$$

$$y_1(t+1) = \bar{x}(t)\bar{y}_1(t) + \bar{y}_1(t)y_2(t) + x(t)y_1(t)$$

$$y_2(t+1) = x(t)\bar{y}_2(t)$$

$$J_1(t) = \bar{x}(t) + \bar{y}_1(t) \cdot y_2(t) \quad K_1(t) = \bar{x}(t)$$

$$J_2(t) = x(t) \quad K_2(t) = y_2(t)$$

Entrada Actual	Estado Actual		F <sub>1</sub> [X(t),Y(t)]				F <sub>2</sub> [X(t),Y(t)]		FZ[X(t),Y(t)]
			Excitación Actual				Estado Siguiente		
X(t)	Y <sub>1</sub> (t)	Y <sub>2</sub> (t)	J <sub>1</sub> (t)	K <sub>1</sub> (t)	J <sub>2</sub> (t)	K <sub>2</sub> (t)	Y <sub>1</sub> (t+1)	Y <sub>2</sub> (t+1)	Z(t)
0	0	0	1	1	0	0	1	0	0
0	0	1							
0	1	0							
0	1	1							
1	0	0							
1	0	1							
1	1	0							
1	1	1							

# Análisis y diseño de circuitos secuenciales síncronos

## 2. b) Tablas de transiciones y Diagramas de Estado

- La finalidad de las tablas de transiciones y diagramas de estado es describir de manera sencilla el funcionamiento del circuito

### b1) Tablas de transiciones

#### \*Máquina de Mealy

- Columna = Entrada Actual
- Fila = Estado Actual
- Intersección = Estado siguiente / Salida actual

#### \*Máquina de Moore

- Columna = Entrada Actual
- Fila = Estado Actual / Salida Actual
- Intersección = Estado siguiente

\*Tabla de transiciones del Ejemplo 1 (Máquina de Mealy): intersección = [Estado siguiente] / Salida actual

Entrada Actual X(t) \ Estado Actual Y(t)	0	1
[0,0]	[1,0] / 0	[0,1] / 0
[0,1]	[1,0] / 0	[1,0] / 0
[1,0]	[0,0] / 0	[1,1] / 1
[1,1]	[0,0] / 0	[1,0] / 0

Si hacemos:

$$[0,0] = q_0$$

$$[0,1] = q_1$$

$$[1,0] = q_2$$

$$[1,1] = q_3$$



Entrada Actual X(t) \ Estado Actual Y(t)	0	1
q <sub>0</sub>	q <sub>2</sub> / 0	q <sub>1</sub> / 0
q <sub>1</sub>	q <sub>2</sub> / 0	q <sub>2</sub> / 0
q <sub>2</sub>	q <sub>0</sub> / 0	q <sub>3</sub> / 1
q <sub>3</sub>	q <sub>0</sub> / 0	q <sub>2</sub> / 0

# Análisis y diseño de circuitos secuenciales síncronos

## 2. b) Tablas de transiciones y Diagramas de Estado

### b2) Diagramas de estado

- Su finalidad es describir de manera sencilla el funcionamiento del circuito
- Permite conocer fácilmente la secuencia de salida y de los estados por los que pasa el sistema cuando le llega una secuencia a la entrada

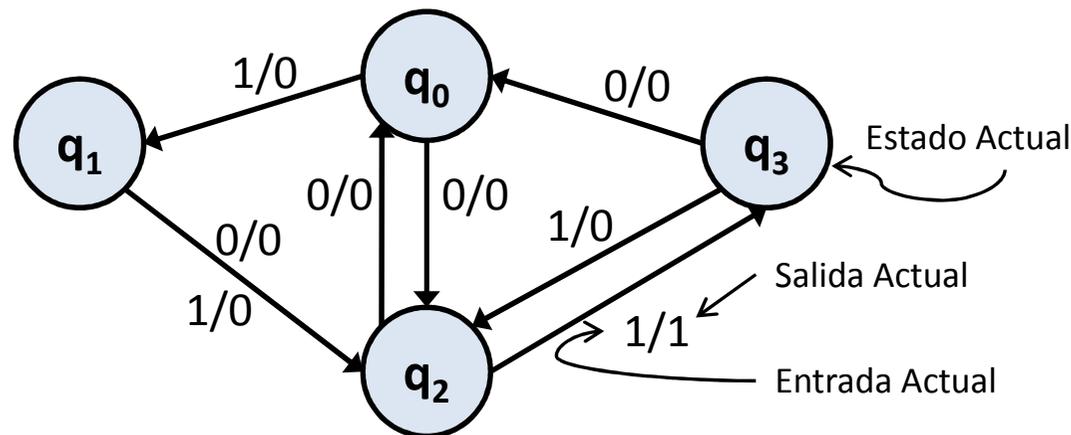
#### \*Máquina de Mealy

- Nodo = Estado
- Flecha = Transición con la entrada actual / salida actual

#### \*Máquina de Moore

- Nodo = Estado con salida actual
- Flecha = transición con entrada actual

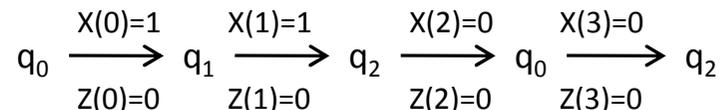
#### \*Diagrama de estados del Ejemplo 1 (Máquina de Mealy)



Ej: Sea la secuencia  $X(0)=1, X(1)=1, X(2)=0, X(3)=0$

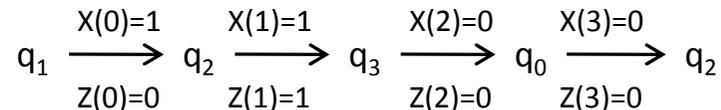
1) Si estado inicial =  $q_0$

Secuencia de estados:  
Secuencia de salidas:



2) Si estado inicial =  $q_1$ :

Secuencia de estados:  
Secuencia de salidas:



# Análisis y diseño de circuitos secuenciales síncronos

## 2. c) Tablas de transiciones y excitaciones de los biestables

Para el Ejemplo 1 la tabla completa de transiciones y excitaciones de los biestables es:

Entrada Actual	Estado Actual		$F_I[X(t),Y(t)]$				$F_Y[X(t),Y(t)]$		$F_Z[X(t),Y(t)]$
			Excitación Actual				Estado Siguiente		Salida Actual
X(t)	Y <sub>1</sub> (t)	Y <sub>2</sub> (t)	J <sub>1</sub> (t)	K <sub>1</sub> (t)	J <sub>2</sub> (t)	K <sub>2</sub> (t)	Y <sub>1</sub> (t+1)	Y <sub>2</sub> (t+1)	Z(t)
0	0	0	1	1	0	0	1	0	0
0	0	1	1	1	0	1	1	0	0
0	1	0	1	1	0	0	0	0	0
0	1	1	1	1	0	1	0	0	0
1	0	0	0	0	1	0	0	1	0
1	0	1	1	0	1	1	1	0	0
1	1	0	0	0	1	0	1	1	1
1	1	1	0	0	1	1	1	0	0

# Análisis y diseño de circuitos secuenciales síncronos

---

## PASOS A SEGUIR PARA DISEÑAR DE SISTEMAS SECUENCIALES SÍNCRONOS

### 1. Descripción del funcionamiento del sistema:

Consiste en definir e identificar correctamente las **entradas, salidas y estados** del sistema

### 2. Construcción de la tabla y el diagrama de estados

### 3. Minimización de estados

Consiste en utilizar el mínimo número de estados posibles para el correcto funcionamiento

### 4. Asignación y Codificación de estados

Determinar el número de biestables necesarios;  $N = n^{\circ}$  estados;  $2^{n-1} < N \leq 2^n$

Representar los estados mediante las salidas de los biestables

Realizar la asignación (arbitraria) de estados : asociar un estado a una codificación

### 5. Implementación del circuito

Obtención de la tabla de transiciones codificada

Ecuaciones de excitación de los biestables simplificadas

Ecuaciones de salidas simplificadas

Dibujo del circuito

# Análisis y diseño de circuitos secuenciales síncronos

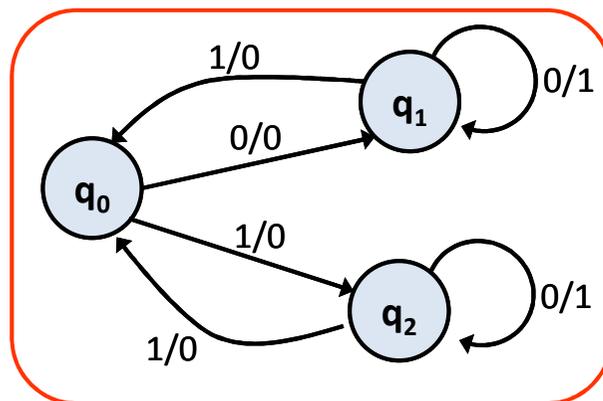
Antes de pasar a los ejemplos se explicará el paso 3 de diseño de circuitos secuenciales síncronos:

## 3. Minimización de estados

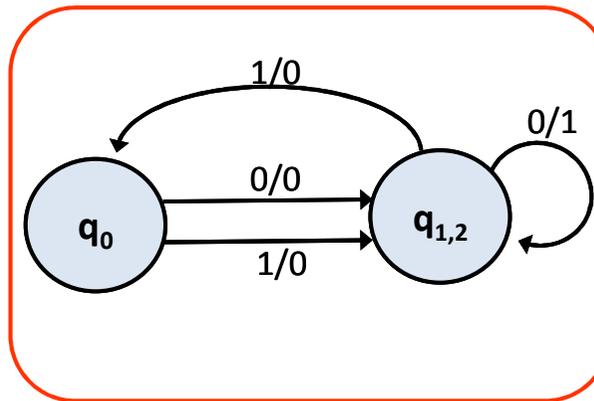
Consiste en utilizar el mínimo número de estados posibles para el correcto funcionamiento

Se trata de detectar ESTADOS EQUIVALENTES (indistinguibles a través de los valores de salida)

Ejemplo previo:



SISTEMA SECUENCIAL 1



SISTEMA SECUENCIAL 2

DESDE EL PUNTO DE VISTA DE UN OBSERVADOR EXTERIOR ESTOS DOS SISTEMAS REALIZAN LA MISMA FUNCIÓN

PERO EL SEGUNDO SISTEMA ES MÁS SIMPLE: SÓLO REQUIERE DOS ESTADOS

## Método de minimización

### 1. Agrupar los estados en clases:

Cada clase contendrá los estados que ante las mismas entradas tengan las mismas salidas

### 2. Comprobación de estados equivalentes (indistinguibles) dentro de la misma clase

Ante las mismas entradas, los estados tienen las mismas salidas y, además, sus estados siguientes están en la misma clase

### 3. Si la clase no es de estados equivalentes: subdivisión en subclases

# Análisis y diseño de circuitos secuenciales síncronos

## 3. Minimización de estados

Ejemplo

		ENTRADA	
		0	1
ESTADO ACTUAL	$q_0$	$q_0/1$	$q_4/0$
	$q_1$	$q_0/0$	$q_4/0$
	$q_2$	$q_1/0$	$q_5/0$
	$q_3$	$q_1/0$	$q_5/0$
	$q_4$	$q_2/0$	$q_4/1$
	$q_5$	$q_3/0$	$q_5/1$

1. Agrupar los estados en clases

2. Comprobación de estados equivalentes dentro de la misma clase

3. Si la clase no es de estados equivalentes: subdivisión en subclases

CLASE  $C_{10}=[q_0]$

CLASE  $C_{00}=[q_1, q_2, q_3]$

CLASE  $C_{01}=[q_4, q_5]$

# Análisis y diseño de circuitos secuenciales síncronos

## 3. Minimización de estados

1. Agrupar los estados en clases

2. Comprobación de estados equivalentes dentro de la misma clase

3. Si la clase no es de estados equivalentes: subdivisión en subclases

		ENTRADA	
		0	1
ESTADO ACTUAL	q <sub>0</sub>	q <sub>0</sub> /1	q <sub>4</sub> /0
	q <sub>1</sub>	q <sub>0</sub> /0	q <sub>4</sub> /0
	q <sub>2</sub>	q <sub>1</sub> /0	q <sub>5</sub> /0
	q <sub>3</sub>	q <sub>1</sub> /0	q <sub>5</sub> /0
	q <sub>4</sub>	q <sub>2</sub> /0	q <sub>4</sub> /1
	q <sub>5</sub>	q <sub>3</sub> /0	q <sub>5</sub> /1

CLASE	C <sub>10</sub> =[q <sub>0</sub> ]		C <sub>00</sub> =[q <sub>1</sub> , q <sub>2</sub> , q <sub>3</sub> ]						C <sub>01</sub> =[q <sub>4</sub> , q <sub>5</sub> ]			
ESTADO	q <sub>0</sub>		q <sub>1</sub>		q <sub>2</sub>		q <sub>3</sub>		q <sub>4</sub>		q <sub>5</sub>	
ENTRADA	0	1	0	1	0	1	0	1	0	1	0	1
EST. SIG.	q <sub>0</sub>	q <sub>4</sub>	q <sub>0</sub>	q <sub>4</sub>	q <sub>1</sub>	q <sub>5</sub>	q <sub>1</sub>	q <sub>5</sub>	q <sub>2</sub>	q <sub>4</sub>	q <sub>3</sub>	q <sub>5</sub>
CLASE EST. SIG.	C <sub>10</sub>	C <sub>01</sub>	C <sub>10</sub>	C <sub>01</sub>	C <sub>00</sub>	C <sub>01</sub>	C <sub>00</sub>	C <sub>01</sub>	C <sub>00</sub>	C <sub>01</sub>	C <sub>00</sub>	C <sub>01</sub>



i C<sub>00</sub>=[q<sub>1</sub>, q<sub>2</sub>, q<sub>3</sub>] hay que dividirlo en subclases !

# Análisis y diseño de circuitos secuenciales síncronos

## 3. Minimización de estados

		ENTRADA	
		0	1
ESTADO ACTUAL	q <sub>0</sub>	q <sub>0</sub> /1	q <sub>4</sub> /0
	q <sub>1</sub>	q <sub>0</sub> /0	q <sub>4</sub> /0
	q <sub>2</sub>	q <sub>1</sub> /0	q <sub>5</sub> /0
	q <sub>3</sub>	q <sub>1</sub> /0	q <sub>5</sub> /0
	q <sub>4</sub>	q <sub>2</sub> /0	q <sub>4</sub> /1
	q <sub>5</sub>	q <sub>3</sub> /0	q <sub>5</sub> /1

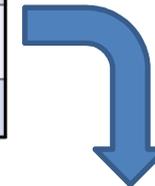
1. Agrupar los estados en clases

2. Comprobación de estados equivalentes dentro de la misma clase

3. Si la clase no es de estados equivalentes: subdivisión en subclases



CLASE	C <sub>10</sub> =[q <sub>0</sub> ]		C <sub>00</sub> =[q <sub>1</sub> , q <sub>2</sub> , q <sub>3</sub> ]						C <sub>01</sub> =[q <sub>4</sub> , q <sub>5</sub> ]			
ESTADO	q <sub>0</sub>		q <sub>1</sub>		q <sub>2</sub>		q <sub>3</sub>		q <sub>4</sub>		q <sub>5</sub>	
ENTRADA	0	1	0	1	0	1	0	1	0	1	0	1
EST. SIG.	q <sub>0</sub>	q <sub>4</sub>	q <sub>0</sub>	q <sub>4</sub>	q <sub>1</sub>	q <sub>5</sub>	q <sub>1</sub>	q <sub>5</sub>	q <sub>2</sub>	q <sub>4</sub>	q <sub>3</sub>	q <sub>5</sub>
CLASE EST. SIG.	C <sub>10</sub>	C <sub>01</sub>	C <sub>10</sub>	C <sub>01</sub>	C <sub>00</sub>	C <sub>01</sub>	C <sub>00</sub>	C <sub>01</sub>	C <sub>00</sub>	C <sub>01</sub>	C <sub>00</sub>	C <sub>01</sub>



CLASE	C <sub>10</sub> =[q <sub>0</sub> ]		C <sub>00</sub> '=[q <sub>1</sub> ]		C <sub>00</sub> ''=[q <sub>2</sub> , q <sub>3</sub> ]				C <sub>01</sub> =[q <sub>4</sub> , q <sub>5</sub> ]			
ESTADO	q <sub>0</sub>		q <sub>1</sub>		q <sub>2</sub>		q <sub>3</sub>		q <sub>4</sub>		q <sub>5</sub>	
ENTRADA	0	1	0	1	0	1	0	1	0	1	0	1
EST. SIG.	q <sub>0</sub>	q <sub>4</sub>	q <sub>0</sub>	q <sub>4</sub>	q <sub>1</sub>	q <sub>5</sub>	q <sub>1</sub>	q <sub>5</sub>	q <sub>2</sub>	q <sub>4</sub>	q <sub>3</sub>	q <sub>5</sub>
CLASE EST. SIG.	C <sub>10</sub>	C <sub>01</sub>	C <sub>10</sub>	C <sub>01</sub>	C <sub>00</sub> '	C <sub>01</sub>	C <sub>00</sub> '	C <sub>01</sub>	C <sub>00</sub> ''	C <sub>01</sub>	C <sub>00</sub> ''	C <sub>01</sub>

# Análisis y diseño de circuitos secuenciales síncronos

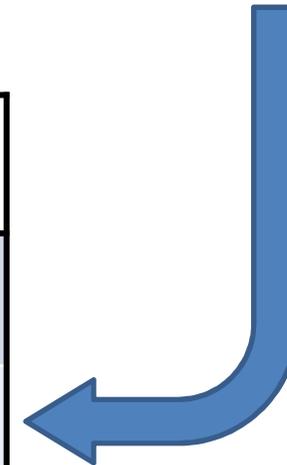
## 3. Minimización de estados

		ENTRADA	
		0	1
ESTADO ACTUAL	$q_0$	$q_0/1$	$q_4/0$
	$q_1$	$q_0/0$	$q_4/0$
	$q_2$	$q_1/0$	$q_5/0$
	$q_3$	$q_1/0$	$q_5/0$
	$q_4$	$q_2/0$	$q_4/1$
	$q_5$	$q_3/0$	$q_5/1$



CLASE	$C_{10}=[q_0]$		$C_{00}'=[q_1]$		$C_{00}''=[q_2, q_3]$		$C_{01}=[q_4, q_5]$					
ESTADO	$q_0$		$q_1$		$q_2$	$q_3$	$q_4$		$q_5$			
ENTRADA	0	1	0	1	0	1	0	1	0	1	0	1
EST. SIG.	$q_0$	$q_4$	$q_0$	$q_4$	$q_1$	$q_5$	$q_1$	$q_5$	$q_2$	$q_4$	$q_3$	$q_5$
CLASE EST. SIG.	$C_{10}$	$C_{01}$	$C_{10}$	$C_{01}$	$C_{00}'$	$C_{01}$	$C_{00}'$	$C_{01}$	$C_{00}''$	$C_{01}$	$C_{00}''$	$C_{01}$

		ENTRADA		CLASE	ESTADOS EQUIVALENTES
		0	1		
ESTADO ACTUAL	$p_0$	$p_0/1$	$p_3/0$	$C_{10}$	$q_0$
	$p_1$	$p_0/0$	$p_3/0$	$C_{00}'$	$q_1$
	$p_2$	$p_1/0$	$p_3/0$	$C_{00}''$	$q_2, q_3$
	$p_3$	$p_2/0$	$p_3/1$	$C_{01}$	$q_4, q_5$



# Análisis y diseño de circuitos secuenciales síncronos

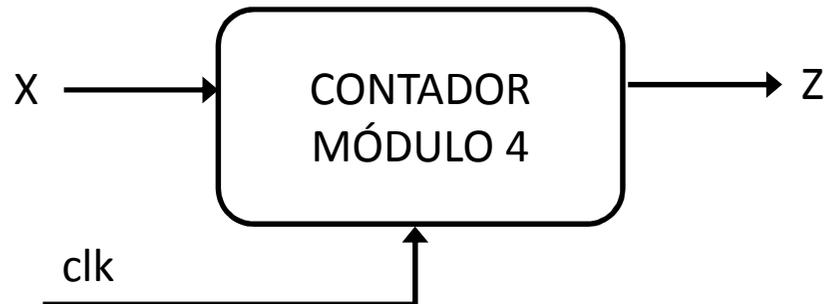
## EJEMPLO DE DISEÑO

Diseño de un contador síncrono de módulo 4.

Entrada de puesta a cero.

Salida que enciende un LED en los 2 últimos estados

1. Descripción del funcionamiento del sistema: definir las **entradas, salidas y estados**



X=1 puesta a 0 del contador

Z=1 enciende el LED

Estados del sistema:

$q_0$ : contador a 0. Salida asociada Z=0

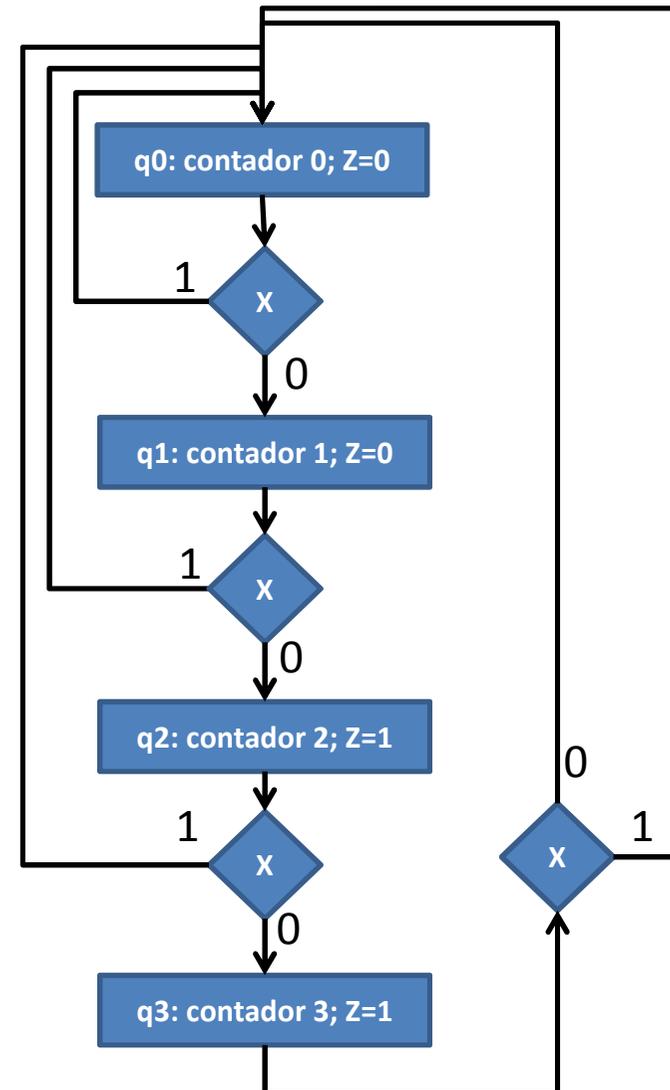
$q_1$ : contador a 1. Salida asociada Z=0

$q_2$ : contador a 2. Salida asociada Z=1

$q_3$ : contador a 3. Salida asociada Z=1

**MÁQUINA DE MOORE**

1. DESCRIPCIÓN
2. CONSTRUCCIÓN TABLA
3. MINIMIZACIÓN
4. ASIGNACIÓN ESTADOS
5. IMPLEMENTACIÓN



# Análisis y diseño de circuitos secuenciales síncronos

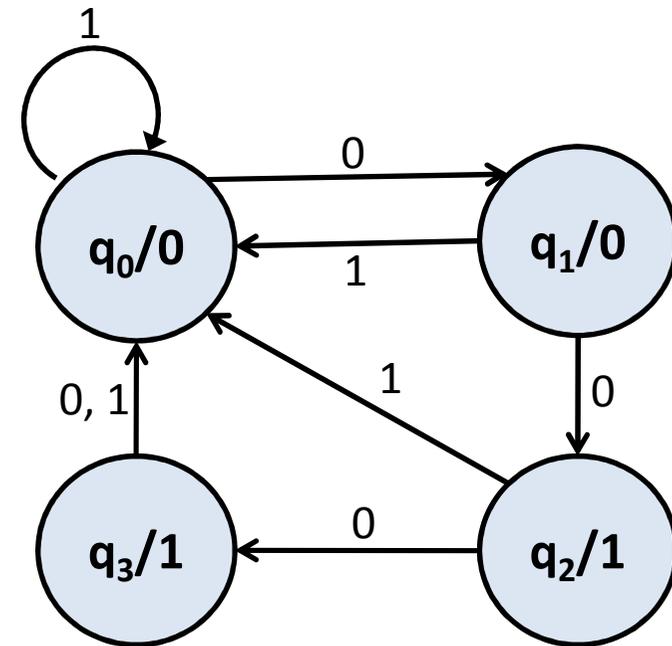
## 2. Construcción de la tabla y el diagrama de estados

1. DESCRIPCIÓN
2. CONSTRUCCIÓN TABLA
3. MINIMIZACION
4. ASIGNACIÓN ESTADOS
5. IMPLEMENTACIÓN

		ENTRADA X	
		0	1
EST. ACTUAL / SALIDA ACTUAL	$q_0/0$	$q_1$	$q_0$
	$q_1/0$	$q_2$	$q_0$
	$q_2/1$	$q_3$	$q_0$
	$q_3/1$	$q_0$	$q_0$

CLASE  $C_0 = [q_0, q_1]$

CLASE  $C_1 = [q_2, q_3]$



# Análisis y diseño de circuitos secuenciales síncronos

## 3. Minimización de estados

1. DESCRIPCIÓN
2. CONSTRUCCIÓN TABLA
3. MINIMIZACIÓN
4. ASIGNACIÓN ESTADOS
5. IMPLEMENTACIÓN

		ENTRADA X		
		0	1	
EST. ACTUAL / SALIDA ACTUAL	$q_0/0$	$q_1$	$q_0$	CLASE $C_0=[q_0, q_1]$
	$q_1/0$	$q_2$	$q_0$	
	$q_2/1$	$q_3$	$q_0$	CLASE $C_1=[q_2, q_3]$
	$q_3/1$	$q_0$	$q_0$	

CLASE	$C_0=[q_0, q_1, ]$				$C_1=[q_2, q_3]$			
ESTADO	$q_0$		$q_1$		$q_2$		$q_3$	
ENTRADA	0	1	0	1	0	1	0	1
EST. SIG.	$q_1$	$q_0$	$q_2$	$q_0$	$q_3$	$q_0$	$q_0$	$q_0$
CLASE EST. SIG.	$C_0$	$C_0$	$C_1$	$C_0$	$C_1$	$C_0$	$C_0$	$C_0$

Las clases  $C_0$  y  $C_1$  hay que dividir las en 2 subclases  $\rightarrow$  4 clases  $\rightarrow$  4 estados  
 No es posible minimizar más

# Análisis y diseño de circuitos secuenciales síncronos

## 4. Asignación y codificación de estados

1. DESCRIPCIÓN
2. CONSTRUCCIÓN TABLA
3. MINIMIZACIÓN
4. ASIGNACIÓN ESTADOS
5. IMPLEMENTACIÓN

		ENTRADA X	
		0	1
EST. ACTUAL / SALIDA ACTUAL	q <sub>0</sub> /0	q <sub>1</sub>	q <sub>0</sub>
	q <sub>1</sub> /0	q <sub>2</sub>	q <sub>0</sub>
	q <sub>2</sub> /1	q <sub>3</sub>	q <sub>0</sub>
	q <sub>3</sub> /1	q <sub>0</sub>	q <sub>0</sub>

N=4 estados (q<sub>0</sub>,q<sub>1</sub>,q<sub>2</sub>,q<sub>3</sub>)  
 $2^{n-1} < N \leq 2^n$

n=2 biestables (Y<sub>1</sub>=Q<sub>1</sub> ; Y<sub>2</sub>=Q<sub>2</sub>)

### CODIFICACIÓN ASIGNADA

ESTADO ACTUAL	CODIFICACIÓN ASIGNADA	
	Q <sub>1</sub>	Q <sub>2</sub>
q <sub>0</sub>	0	0
q <sub>1</sub>	0	1
q <sub>2</sub>	1	0
q <sub>3</sub>	1	1

### ENTRADA X

		ENTRADA X	
		0	1
EST. ACTUAL/SALIDA ACTUAL	00/0	01	00
	01/0	10	00
	10/1	11	00
	11/1	00	00

# Análisis y diseño de circuitos secuenciales síncronos

## 5. Implementación del circuito

1. DESCRIPCIÓN
2. CONSTRUCCIÓN TABLA
3. MINIMIZACIÓN
4. ASIGNACIÓN ESTADOS
5. IMPLEMENTACIÓN

		ENTRADA X	
		0	1
EST. ACTUAL/SALIDA ACTUAL	00/0	01	00
	01/0	10	00
	10/1	11	00
	11/1	00	00

De la tabla anterior se obtiene la  
**Tabla de transiciones y salidas** (ya codificada)

ENTRADA ACTUAL X(t)	EST. ACTUAL		EST. SIGUIENTE		SALIDA ACTUAL Z(t)
	Q1(t)	Q2(t)	Q1(t+1)	Q2(t+1)	
0	0	0	0	1	0
0	0	1	1	0	0
0	1	0	1	1	1
0	1	1	0	0	1
1	0	0	0	0	0
1	0	1	0	0	0
1	1	0	0	0	1
1	1	1	0	0	1

# Análisis y diseño de circuitos secuenciales síncronos

## 5. Implementación del circuito

1. DESCRIPCIÓN
2. CONSTRUCCIÓN TABLA
3. MINIMIZACIÓN
4. ASIGNACIÓN ESTADOS
5. IMPLEMENTACIÓN

ENTRADA ACTUAL X(t)	EST. ACTUAL Q1(t) Q2(t)		EST. SIGUIENTE Q1(t+1) Q2(t+1)		SALIDA ACTUAL Z(t)
0	0	0	0	1	0
0	0	1	1	0	0
0	1	0	1	1	1
0	1	1	0	0	1
1	0	0	0	0	0
1	0	1	0	0	0
1	1	0	0	0	1
1	1	1	0	0	1

Ecuación de la salida Z(t)

$Z(t) \times Q_1(t)$

Q <sub>2</sub> (t) \ Q <sub>1</sub> (t)	00	01	11	10
0	0	1	1	0
1	0	1	1	0

$$Z(t) = Q_1(t)$$

No depende de X es una Máquina de Moore

El resto de la implementación es exactamente igual al diseño de un **contador síncrono**

ENTRADA ACTUAL X(t)	EST. ACTUAL Q1(t) Q2(t)		EST. SIGUIENTE Q1(t+1)=D1(t) Q2(t+1)=D2(t)	
0	0	0	0	1
0	0	1	1	0
0	1	0	1	1
0	1	1	0	0
1	0	0	0	0
1	0	1	0	0
1	1	0	0	0
1	1	1	0	0

# Análisis y diseño de circuitos secuenciales síncronos

## 5. Implementación del circuito

1. DESCRIPCIÓN
2. CONSTRUCCIÓN TABLA
3. MINIMIZACIÓN
4. ASIGNACIÓN ESTADOS
5. IMPLEMENTACIÓN

ENTRADA ACTUAL X(t)	EST. ACTUAL		EST. SIGUIENTE		SALIDA ACTUAL Z(t)
	Q1(t)	Q2(t)	Q1(t+1)=D1(t)	Q2(t+1)=D2(t)	
0	0	0	0	1	0
0	0	1	1	0	0
0	1	0	1	1	1
0	1	1	0	0	1
1	0	0	0	0	0
1	0	1	0	0	0
1	1	0	0	0	1
1	1	1	0	0	1

$D_1(t) \times Q_1(t)$

Q <sub>2</sub> (t) \ Q <sub>1</sub> (t)	00	01	11	10
0	0	1	0	0
1	1	0	0	0

$$D_1(t) = \bar{X} (Q_1(t) \oplus Q_2(t))$$

$D_2(t) \times Q_1(t)$

Q <sub>2</sub> (t) \ Q <sub>1</sub> (t)	00	01	11	10
0	1	1	0	0
1	0	0	0	0

$$D_2(t) = \bar{X} \cdot \overline{Q_2(t)}$$

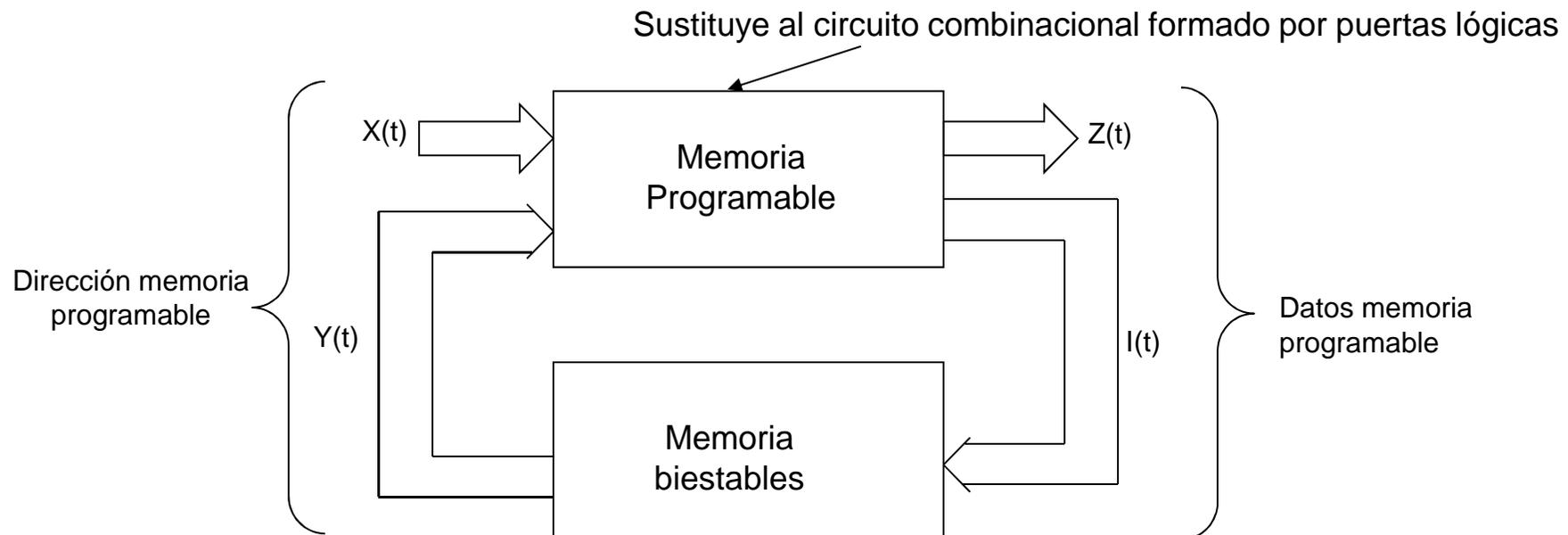


# Análisis y diseño de circuitos secuenciales síncronos

## 5. Implementación del circuito (mediante programación)

1. DESCRIPCIÓN
2. CONSTRUCCIÓN TABLA
3. MINIMIZACIÓN
4. ASIGNACIÓN ESTADOS
5. IMPLEMENTACIÓN

- Se graba en una memoria programable la tabla de decisiones del sistema
- Gran flexibilidad: los cambio en la tabla de decisiones sólo afectan al contenido de la memoria programable y no a la circuitería exterior
- El sistema evoluciona de acuerdo a una secuencia de lecturas que se efectúa sobre la memoria programable:
  - Se envía a la memoria la dirección, que contiene información sobre el estado actual, y las señales de entrada
  - Se realiza el ciclo de lectura
  - Se obtiene de memoria la información para deducir el estado siguiente y la salida



# Análisis y diseño de circuitos secuenciales síncronos

## 5. Implementación del circuito (mediante programación)

1. DESCRIPCIÓN
2. CONSTRUCCIÓN TABLA
3. MINIMIZACIÓN
4. ASIGNACIÓN ESTADOS
5. IMPLEMENTACIÓN

En el ejemplo del contador módulo 4 la tabla completa era:

ENTRADA ACTUAL X(t)	EST. ACTUAL Q1(t) Q2(t)		EST. SIGUIENTE Q1(t+1)=D1(t) Q2(t+1)=D2(t)		SALIDA ACTUAL Z(t)
0	0	0	0	1	0
0	0	1	1	0	0
0	1	0	1	1	1
0	1	1	0	0	1
1	0	0	0	0	0
1	0	1	0	0	0
1	1	0	0	0	1
1	1	1	0	0	1

$A_2$      $A_1$      $A_0$ 
 $D_1$      $D_0$ 
 $D_2$

Entradas de dirección
Entradas de datos

**La dirección tiene tres campos:**

$A_2$  para la señal de entrada  $x(t)$

$A_1 A_0 = Q1(t) Q2(t)$  información estado actual

**Los datos tiene tres campos:**

$D_2$  para la señal de salida  $z(t)$

$D_1 D_0 = Q1(t+1) Q2(t+1)$  información estado siguiente

