

TEMA 4: Sistemas Digitales

Índice:

1. Circuitos Combinacionales

1.1 Definición

1.2 Circuitos Aritméticos

1.3 Decodificadores

1.4 Codificadores

1.5 Multiplexores

1.6 Demultiplexor

2. Circuitos Secuenciales

2.1 Definición

2.2 Biestables

2.3 Registros y Contadores

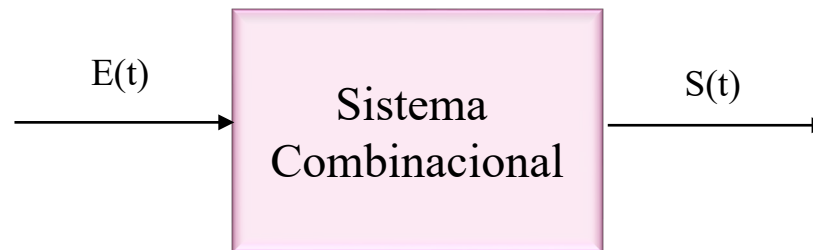
TEMA 4: Sistemas Digitales

Bibliografía:

- ❑ T.L.Floyd. Fundamentos de Sistemas Digitales.
 - Cap. 6: Funciones de la Lógica Combinacional
 - Cap. 8: Flip-Flops y Dispositivos Relacionados
 - Cap. 9: Contadores
 - Cap. 10: Registros de Desplazamiento
- ❑ C.Blanco. Fundamentos de Electrónica Digital.
 - Cap. 4: Circuitos Combinacionales.
 - Cap. 6: Registros
 - Cap. 7: Contadores
- ❑ J.M^a Angulo y J. García. Sistemas Digitales y Tecnología de Computadores.
 - Cap. 4: Análisis y Diseño de Sistemas Combinacionales.
 - Cap. 6: Elementos Aritméticos.

1.1 Definición de Sistema Combinacional.

Es un conjunto de dispositivos lógicos en el que las salidas dependen exclusivamente del valor existente en cada momento en las entradas.



- Todo sistema combinacional se puede definir mediante funciones lógicas.
- Habitualmente los sistemas combinacionales se encuentran integrados en bloques que realizan funciones específicas.
- A la hora de implementar un sistema combinacional utilizaremos un criterio de simplificación, buscando la menor cantidad de puertas lógicas básicas.

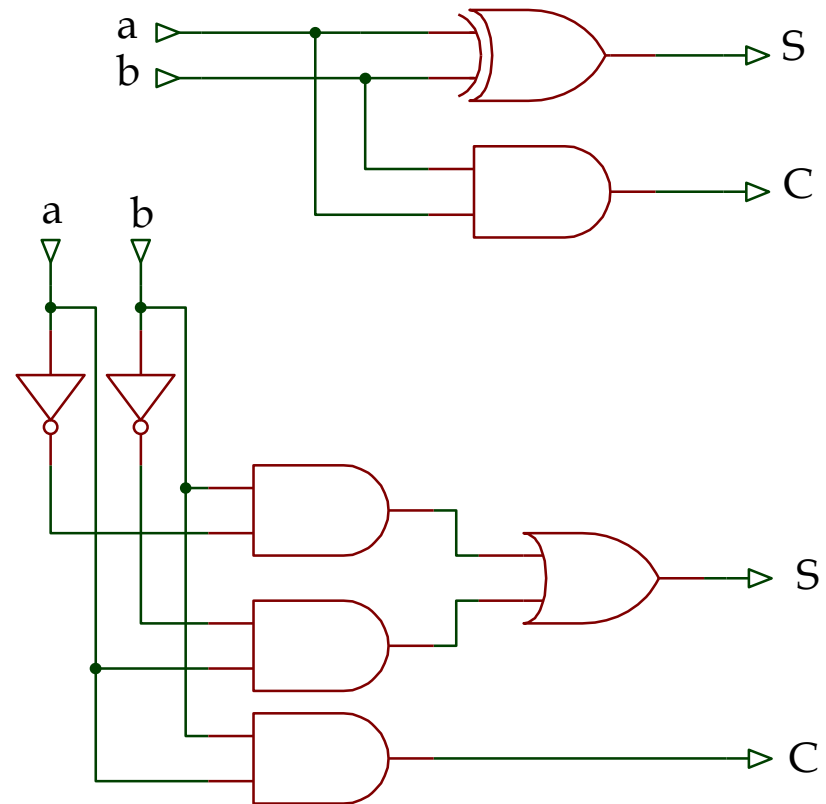
1.2. Circuitos Aritméticos. Semisumador

Es un circuito combinacional capaz de sumar dos dígitos binarios (a y b) proporcionando como salidas la suma (S) y el posible acarreo (C). Su tabla de verdad será:

a	b	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

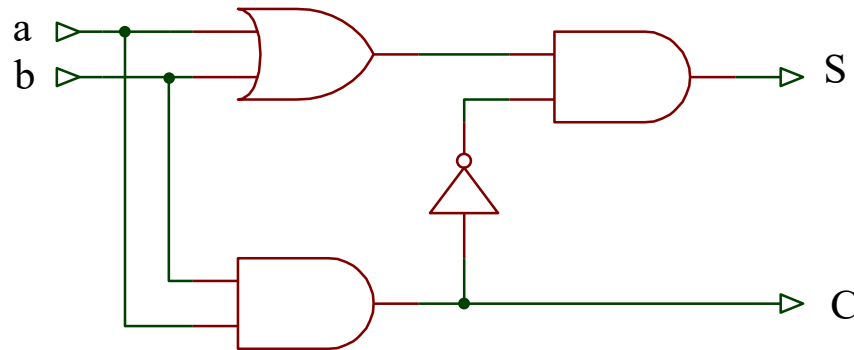
$$S = \bar{a}b + a\bar{b} = a \oplus b$$

$$C = ab$$

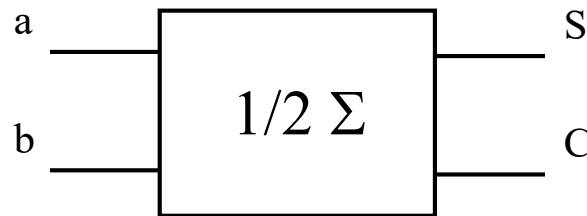


Si operamos la expresión de la suma:

$$S = a \oplus b = (a + b)(\bar{a} + \bar{b}) = (a + b)\bar{a}\bar{b} = (a + b)\bar{C}$$



Bloque Funcional:



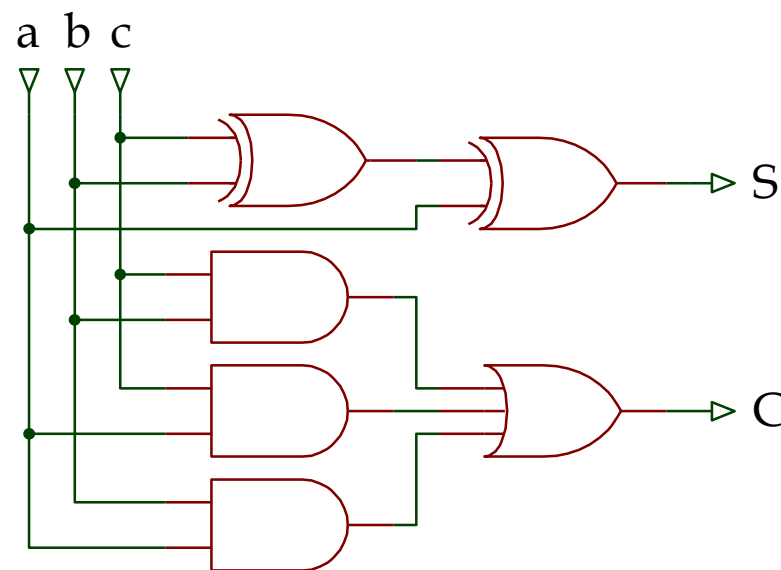
Sumador Completo

Es un circuito combinacional capaz de sumar dos dígitos binarios junto con el posible acarreo procedente de la etapa anterior y proporcionando como salidas la suma y el acarreo producido.

a	b	c	S	C
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

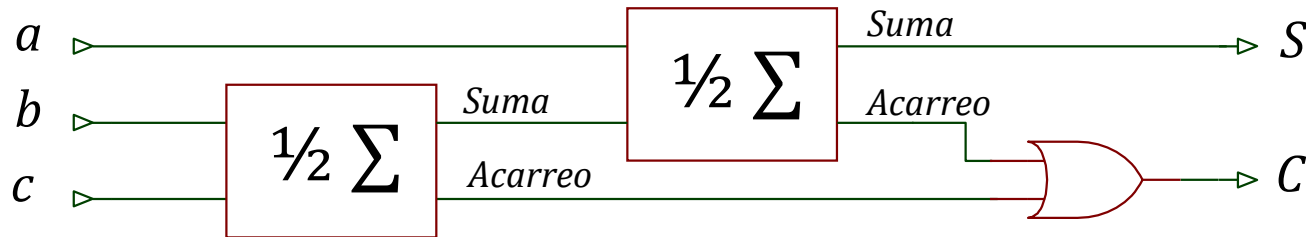
$$S = \bar{a}\bar{b}c + \bar{a}b\bar{c} + a\bar{b}\bar{c} + abc = a \oplus b \oplus c$$

$$C = ab + ac + bc$$

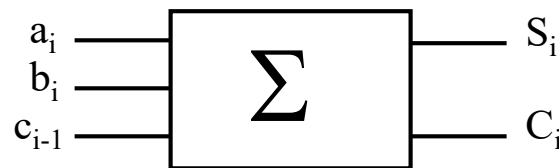


Sumador Completo

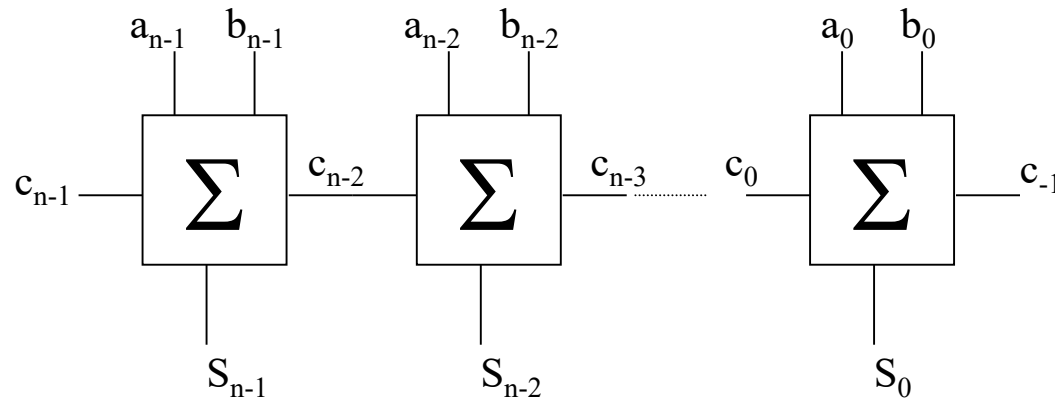
Construcción mediante Semisumadores



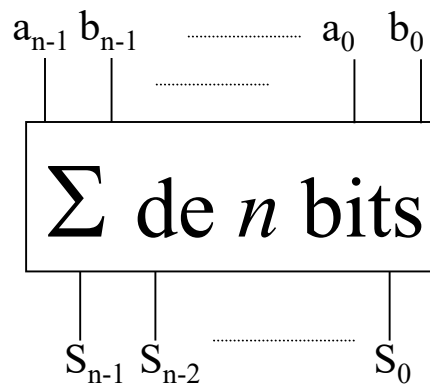
Bloque Funcional de un Sumador Completo de 1 Bit.



Podemos construir sumadores del número de bits que necesitamos simplemente encadenando sumadores completos de 1 bit



Bloque funcional del sumador paralelo de n bits

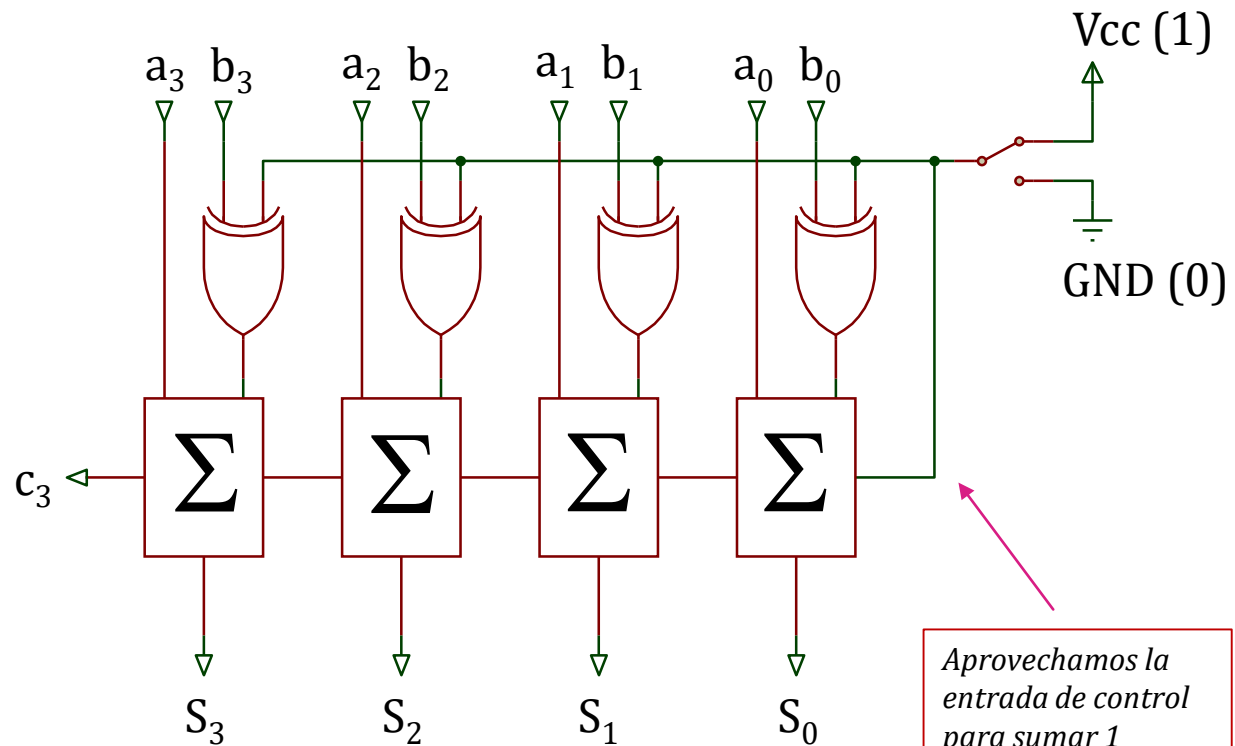


Sumador/Restador Binario

Es un circuito capaz de realizar las operaciones de suma y resta sobre números con signo. Si los números están representados en **Complemento a 2**, el circuito resultante sería:

\bar{s}/r	b	b'
0	0	0
0	1	1
1	0	1
1	1	0

$$b' = (\bar{s}/r) \oplus b$$



Carece de significado en C2

Aprovechamos la entrada de control para sumar 1

1.3. Decodificador. Definición

Podemos definir al **decodificador** como un circuito combinacional que consta de n entradas y 2^n salidas como máximo. Este circuito pone a uno la salida cuyo índice coincide con la combinación binaria presente en las entradas. Se suelen nombrar haciendo referencia al número de entradas y salidas que poseen. Así, un decodificador de **2 a 4 líneas** presenta la siguiente tabla de verdad:

a	b	S_0	S_1	S_2	S_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

$$S_0 = \bar{a}\bar{b}$$

$$S_1 = \bar{a}b$$

$$S_2 = a\bar{b}$$

$$S_3 = ab$$

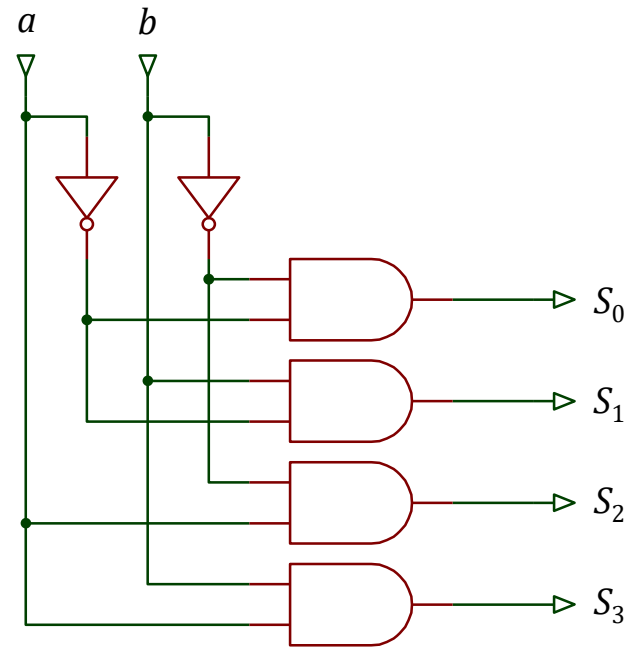
Cada salida hace referencia a uno de los posibles productos canónicos de las variables de entrada

$$S_0 = \bar{a}\bar{b}$$

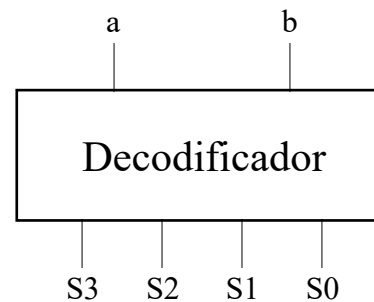
$$S_1 = \bar{a}b$$

$$S_2 = a\bar{b}$$

$$S_3 = ab$$



Bloque Funcional:



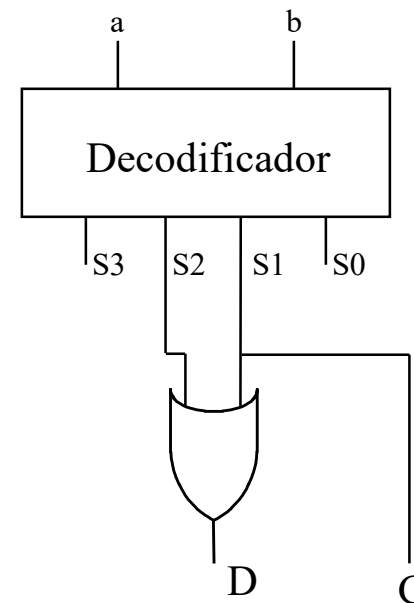
Puede utilizarse el decodificador de n entradas como circuito para implementar funciones lógicas de n variables lógicas. La idea se basa en utilizar los productos canónicos que produzcan un valor lógico 1 en la salida.

Ejemplo: Implementación de un semirestador.

a	b	D	C
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

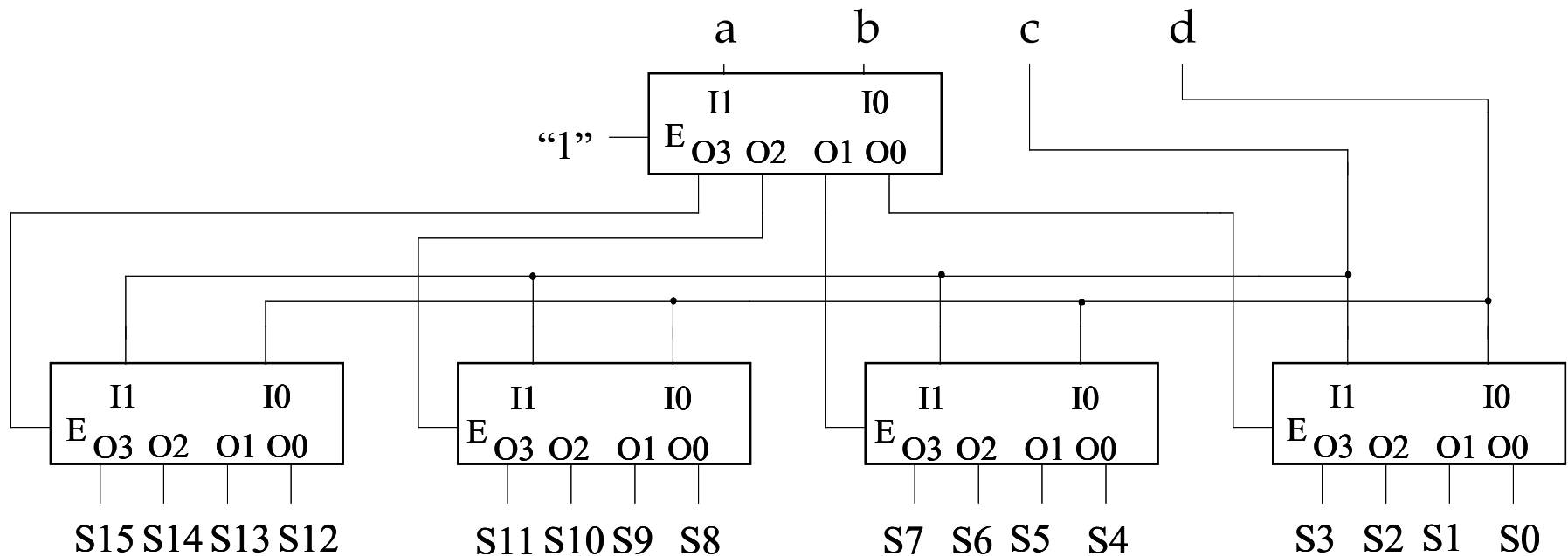
$$C = \sum_2 (1)$$

$$D = \sum_2 (1, 2)$$



Agrupación de Decodificadores

Decodificador de 4 a 16 líneas mediante el uso de la entrada de habilitación E



1.4. Codificador. Definición

Podemos definir al Codificador como un circuito combinacional que consta de 2^n entradas y n salidas. Este circuito coloca en sus salidas la combinación binaria correspondiente al índice de la entrada activada. Existen dos tipos: los Codificadores Prioritarios y los No Prioritarios.

Un Codificador Decimal-BCD **no prioritario** tiene la siguiente tabla de verdad:

E0	E1	E2	E3	E4	E5	E6	E7	E8	E9	DCBA	NA
1	0	0	0	0	0	0	0	0	0	0000	0
0	1	0	0	0	0	0	0	0	0	0001	0
0	0	1	0	0	0	0	0	0	0	0010	0
0	0	0	1	0	0	0	0	0	0	0011	0
0	0	0	0	1	0	0	0	0	0	0100	0
0	0	0	0	0	1	0	0	0	0	0101	0
0	0	0	0	0	0	1	0	0	0	0110	0
0	0	0	0	0	0	0	1	0	0	0111	0
0	0	0	0	0	0	0	0	1	0	1000	0
0	0	0	0	0	0	0	0	0	1	1001	0
0	0	0	0	0	0	0	0	0	0	0000	1

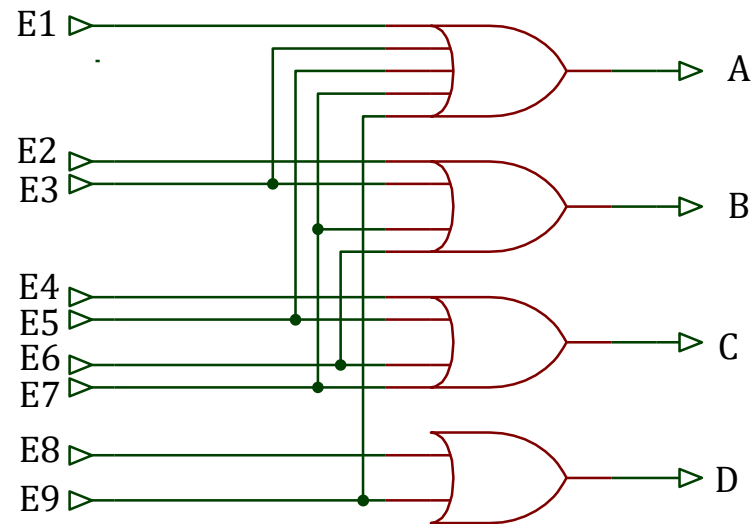
$$D = E8 + E9$$

$$C = E4 + E5 + E6 + E7$$

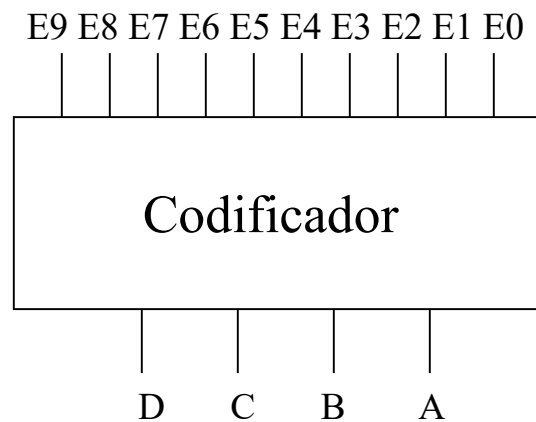
$$B = E2 + E3 + E6 + E7$$

$$A = E1 + E3 + E5 + E7 + E9$$

Esquema:



Bloque funcional:



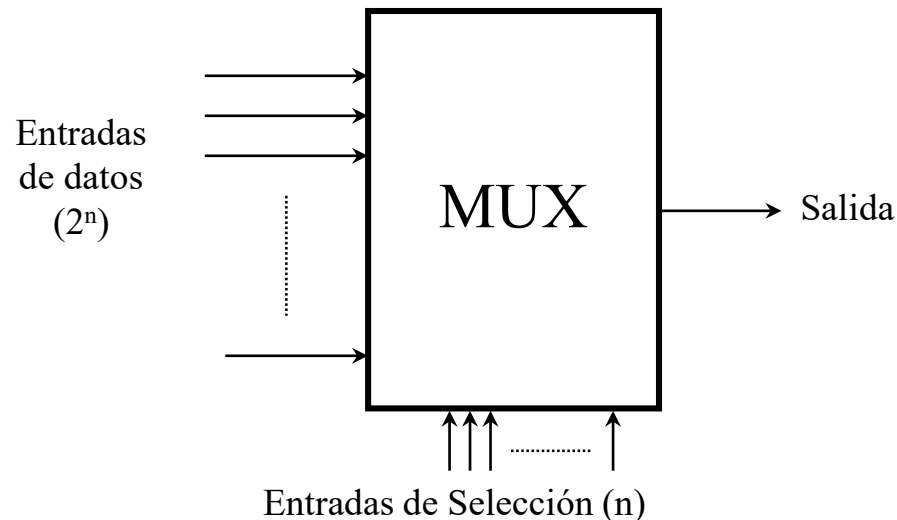
Si activamos más de una entrada simultáneamente en un codificador no prioritario se solaparán las combinaciones de salida que produce cada una de ellas.

Para evitar este problema se diseñan los **codificadores prioritarios**. En estos, cuando activamos más de una señal de entrada al mismo tiempo, solamente proporcionan la salida correspondiente a la entrada que está dotada de mayor prioridad, que generalmente es la que tiene asignada un mayor número decimal. El codificador Decimal-BCD anterior tendría la siguiente tabla de verdad:

E9	E8	E7	E6	E5	E4	E3	E2	E1	E0	DCBA	NA
0	0	0	0	0	0	0	0	0	1	0000	0
0	0	0	0	0	0	0	0	1	X	0001	0
0	0	0	0	0	0	0	1	X	X	0010	0
0	0	0	0	0	0	1	X	X	X	0011	0
0	0	0	0	0	1	X	X	X	X	0100	0
0	0	0	0	1	X	X	X	X	X	0101	0
0	0	0	1	X	X	X	X	X	X	0110	0
0	0	1	X	X	X	X	X	X	X	0111	0
0	1	X	X	X	X	X	X	X	X	1000	0
1	X	X	X	X	X	X	X	X	X	1001	0
0	0	0	0	0	0	0	0	0	0	0000	1

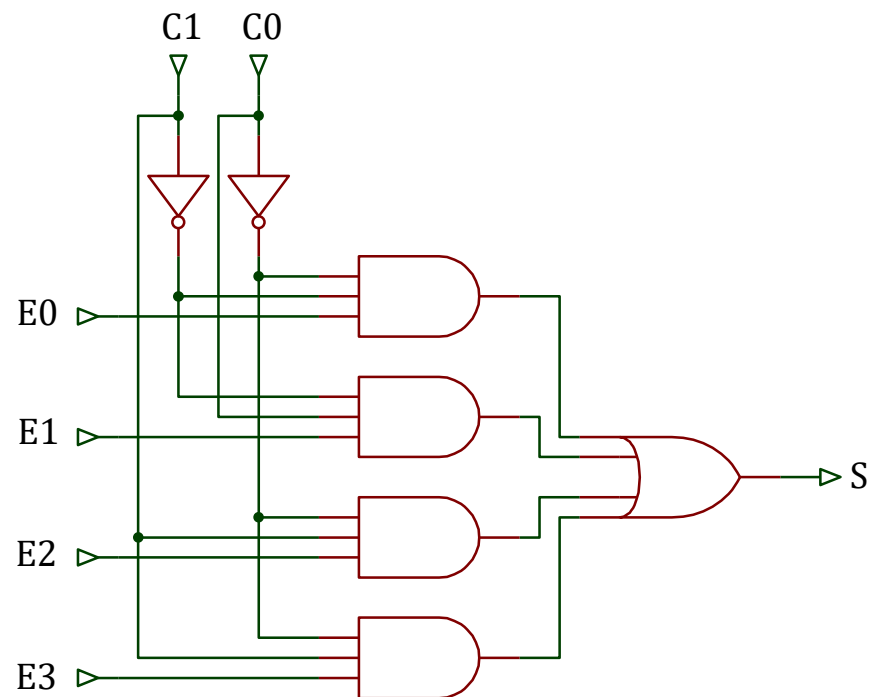
1.5. Multiplexor. Definición

Podemos definir al **multiplexor** o **selector de datos** como un circuito combinatorial que consta de 2^n entradas de datos, n entradas de selección y una salida. Este circuito coloca en su salida el valor de la entrada cuyo índice coincide con la combinación binaria presente en las entradas de selección.



Un multiplexor con dos entradas de selección y cuatro de información presenta la siguiente tabla de verdad:

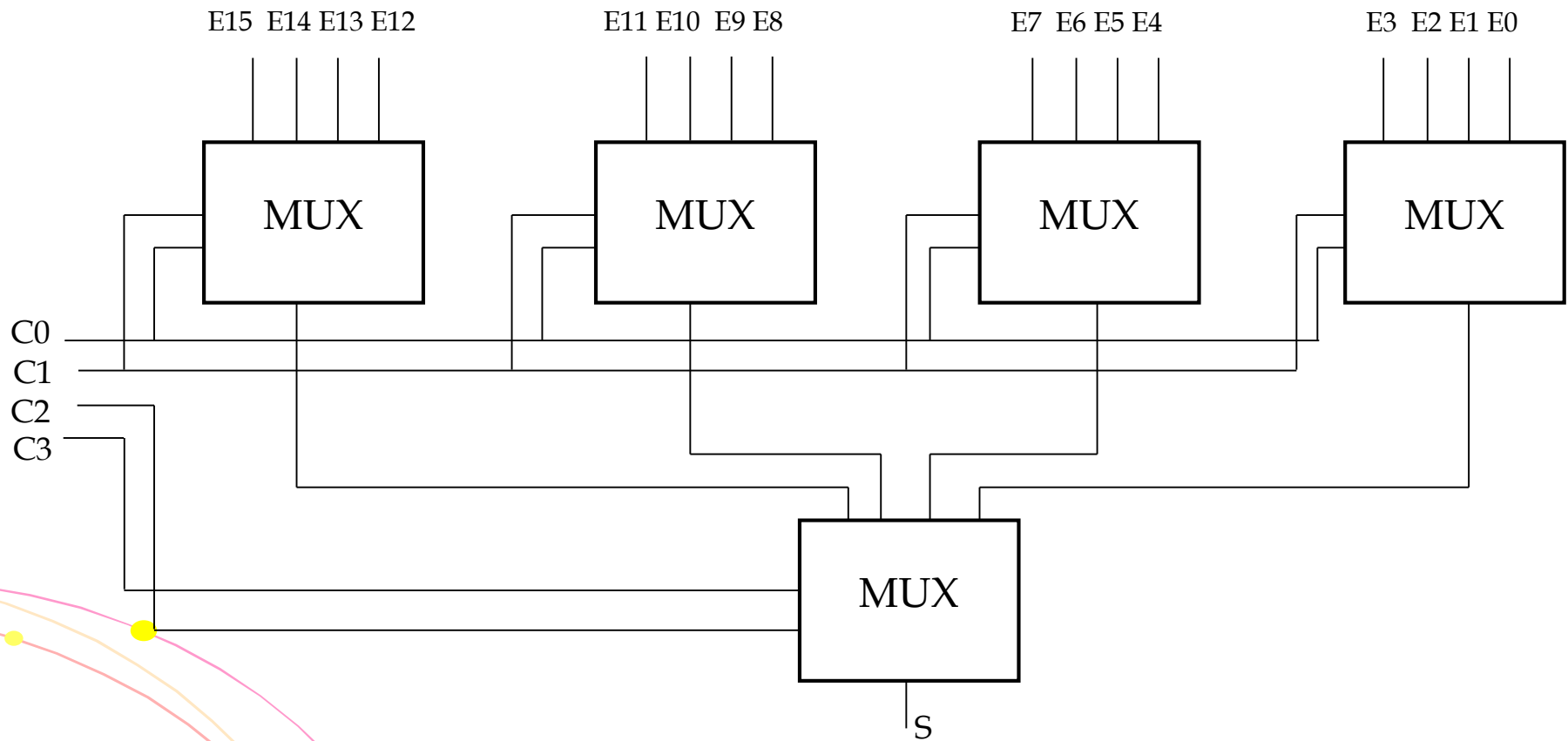
C1	C0	E3	E2	E1	E0	S
0	0	X	X	X	0	0
0	0	X	X	X	1	1
0	1	X	X	0	X	0
0	1	X	X	1	X	1
1	0	X	0	X	X	0
1	0	X	1	X	X	1
1	1	0	X	X	X	0
1	1	1	X	X	X	1



$$S = \overline{C0} \overline{C1} E0 + \overline{C1} C0 E1 + C1 \overline{C0} E2 + C1 C0 E3$$

Asociación de Multiplexores

Podemos construir multiplexores mayores asociando otros de menos entradas:



Implementación de Funciones

Puede utilizarse el multiplexor de n -entradas de selección como circuito para implementar funciones lógicas de $n+1$ variables lógicas. La idea consiste en asociar a las entradas de selección las variables lógicas exceptuando una de ellas que se utiliza para configurar la entrada de datos.

Se construye una tabla de verdad organizada de la siguiente manera:

n variables de selección				1 variable de datos		
				0	1	
2^n	0	0	...	0		→ selecciona entrada 0
	0	0	...	1		→ selecciona entrada 1
	1	1	...	1		→ selecciona entrada 2^{n-1}

Para cada combinación i posible en las entradas de selección tendremos 4 casos posibles en fila en la variable de datos:

0	0	→ Colocamos 0 en la entrada de datos E_i
0	1	→ Colocamos la variable de datos en E_i
1	0	→ Colocamos la variable de datos negada en E_i
1	1	→ Colocamos 1 en E_i

1.6. Demultiplexor. Definición

Podemos definir al demultiplexor como un circuito combinatorial que consta de 1 entrada de datos, n entradas de selección y 2^n salidas. Este circuito coloca en la salida cuyo índice coincide con la combinación binaria presente en las entradas de selección el valor de la entrada de datos.

Demultiplexor de 1 a 4

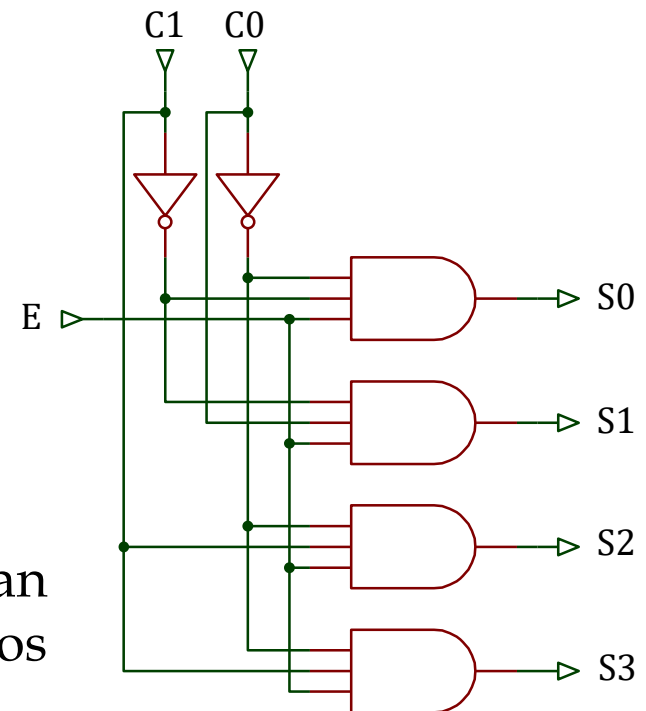
E	C1	C0	S3	S2	S1	S0
0	0	0	0	0	0	0
1	0	0	0	0	0	1
0	0	1	0	0	0	0
1	0	1	0	0	1	0
0	1	0	0	0	0	0
1	1	0	0	1	0	0
0	1	1	0	0	0	0
1	1	1	1	0	0	0

$$S0 = \overline{C0} \overline{C1} E$$

$$S1 = \overline{C1} C0 E$$

$$S2 = C1 \overline{C0} E$$

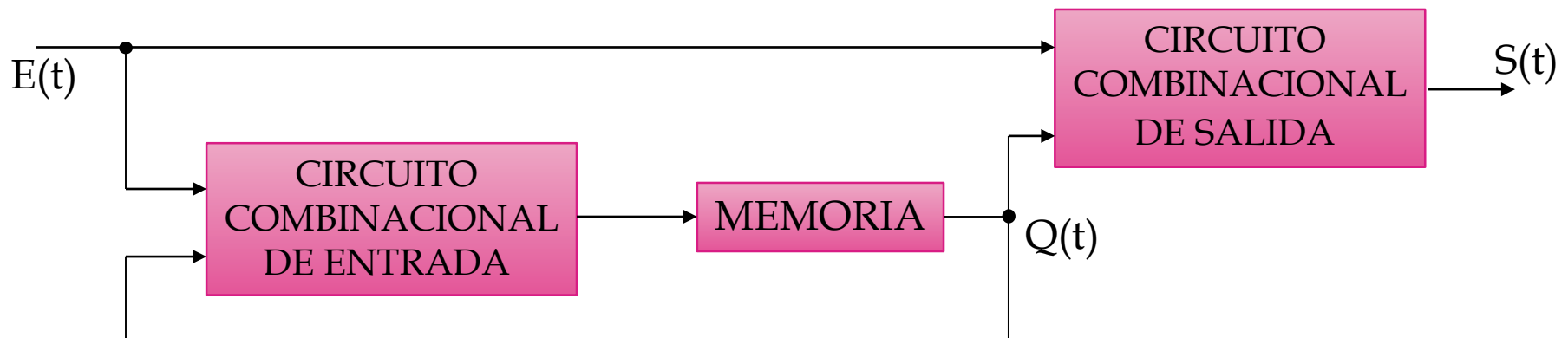
$$S3 = C1 C0 E$$



Por su comportamiento, los demultiplexores van incorporados en el mismo circuito integrado de los decodificadores

2.1. Definición de Sistema Secuencial.

Los sistemas secuenciales son aquellos cuya salida en un instante dado depende, no sólo de las entradas al sistema en ese instante, sino también de la evolución o historia anterior de las mismas, es decir de la secuencia de entradas a la que ha estado sometido.



$$S(t) = f_1[E(t), Q(t)] \quad Q(t + \Delta t) = f_2[E(t), Q(t)]$$

Si suprimimos los circuitos combinacionales de entrada y salida nos queda un circuito compuesto solamente por el elemento de memoria.

2.2 Biestables. Definición y Clasificación

Los biestables (flip-flops o básculas) son circuitos lógicos, con dos estados estables, capaces de permanecer indefinidamente en cualquiera de ellos, aun después de desaparecer la señal de entrada que provocó el paso al estado alcanzado. Característica principal: Realimentación.

Clasificación

Por su modo de funcionamiento hablaremos de:

- Biestable RS
- Biestable JK
- Biestable D

Por su modo de activación hablaremos de:

- Biestables Asíncronos (no existe señal de reloj)
- Biestables Síncronos (con señal de reloj) $\left\{ \begin{array}{l} \text{Por Nivel} \left\{ \begin{array}{l} \text{Alto} \\ \text{Bajo} \end{array} \right. \\ \text{Por Flanco} \left\{ \begin{array}{l} \text{De Subida} \\ \text{De Bajada} \end{array} \right. \end{array} \right.$

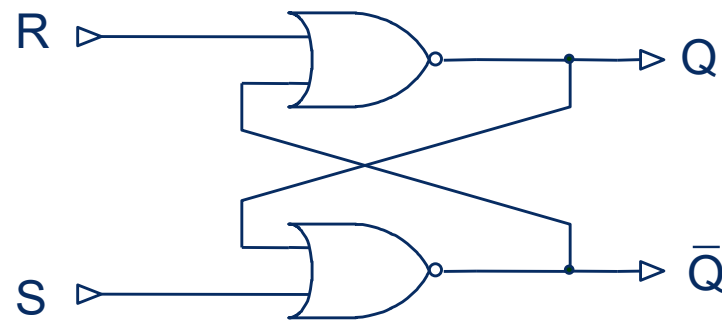
2.2 Biestable RS. Biestable RS Asíncrono

Posee dos entradas, llamadas **R** (Reset) y **S** (Set), y dos salidas complementarias, que denominaremos **Q** y \bar{Q} . Su modo de funcionamiento queda definido por la siguiente tabla de verdad:

R	S	Q(t)	Q(t+1)
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	No permitida
1	1	1	No permitida

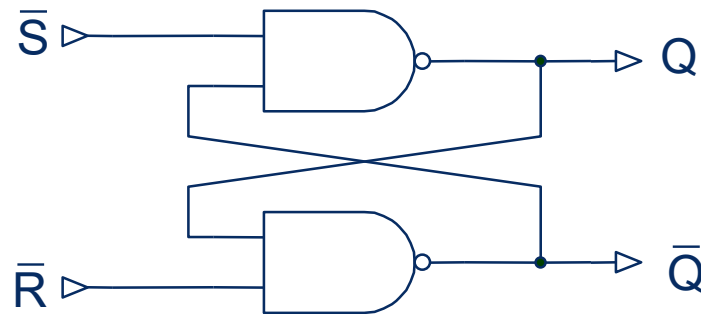
Tabla Resumen:

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

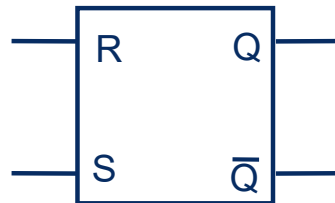


2.2 Biestable RS. Biestable RS Asíncrono

Alternativamente podemos construirlo con puertas NAND, si bien deberemos tener presente que sus entradas serán negadas



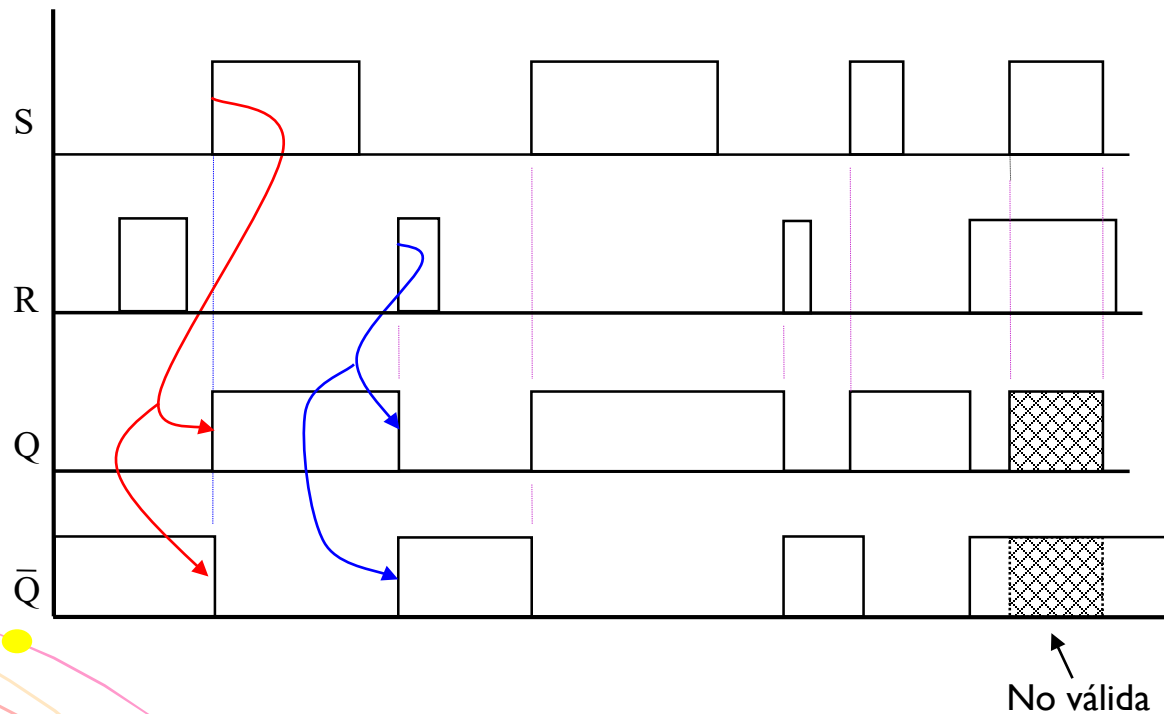
En cualquier caso, para utilizarlo emplearemos su bloque funcional, que siempre responde a la tabla de verdad original.



2.2 Biestable RS. Biestable RS Asíncrono

Diagramas Temporales (Cronogramas):

Biestable RS Asíncrono con puertas NOR:

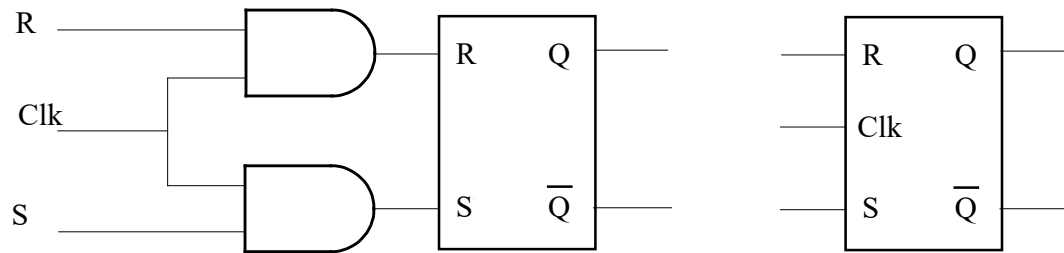


2.2 Biestable RS. Biestable RS Síncrono

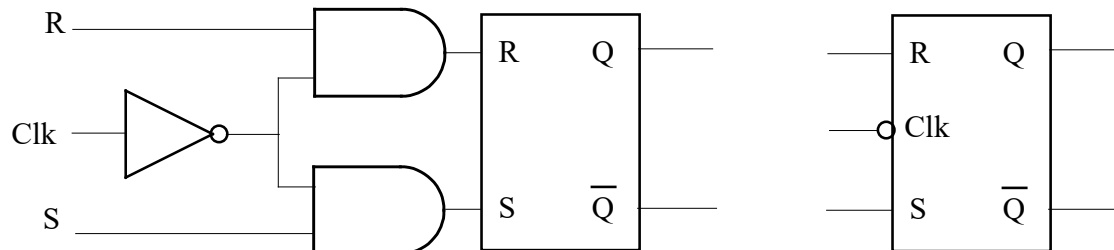
En un biestable síncrono la salida solamente puede cambiar de estado cuando se produzca el valor o cambio adecuado en la entrada de disparo, que denominaremos de *reloj* (Clk).

En un biestable activo por *nivel*, la entrada de reloj actúa como una entrada de habilitación.

Por nivel Alto:



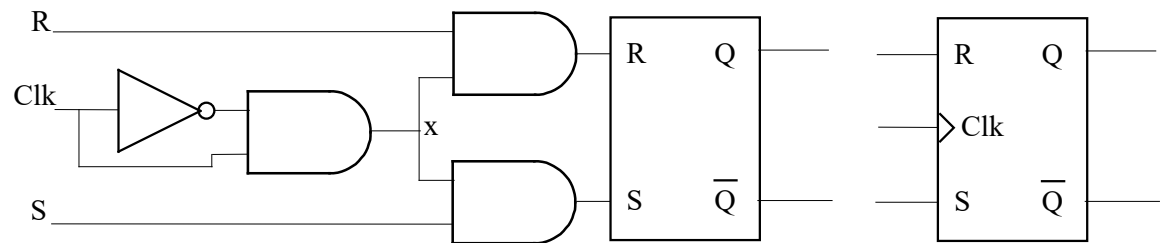
Por nivel Bajo:



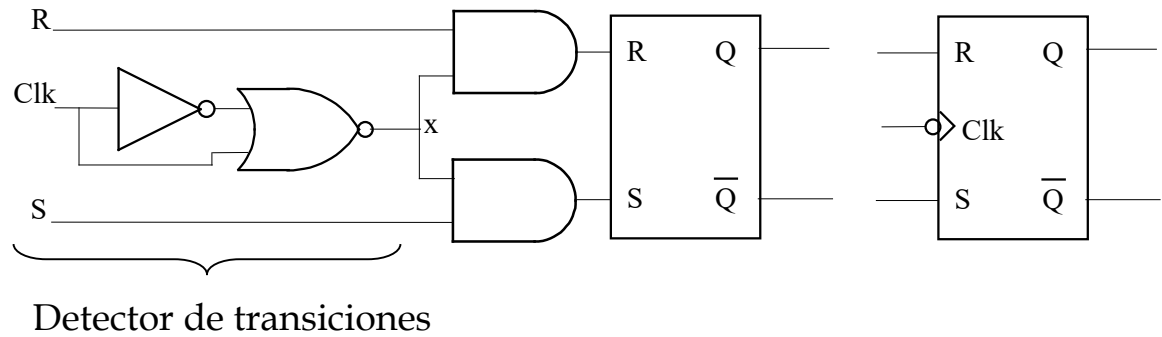
2.2 Biestable RS. Biestable RS Síncrono

Un biestable activo por *flanco*, solamente puede cambiar de estado si en su entrada de reloj se produce la transición adecuada: un cambio de Nivel BAJO a nivel ALTO, si es activo por *flanco de subida* o bien un cambio de Nivel ALTO a Nivel BAJO si es activo por *flanco de bajada*.

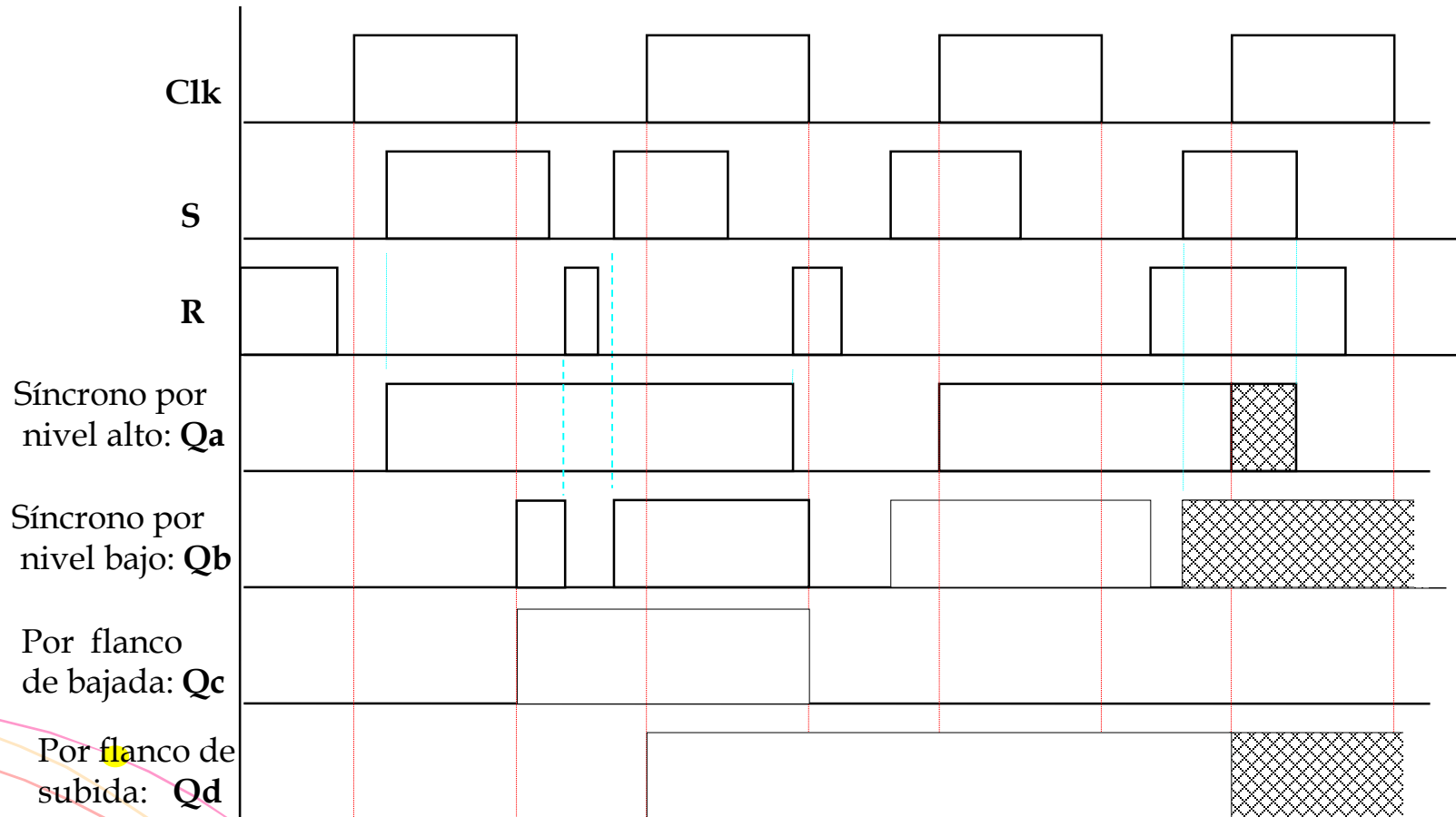
Por flanco de subida:



Por flanco de bajada:



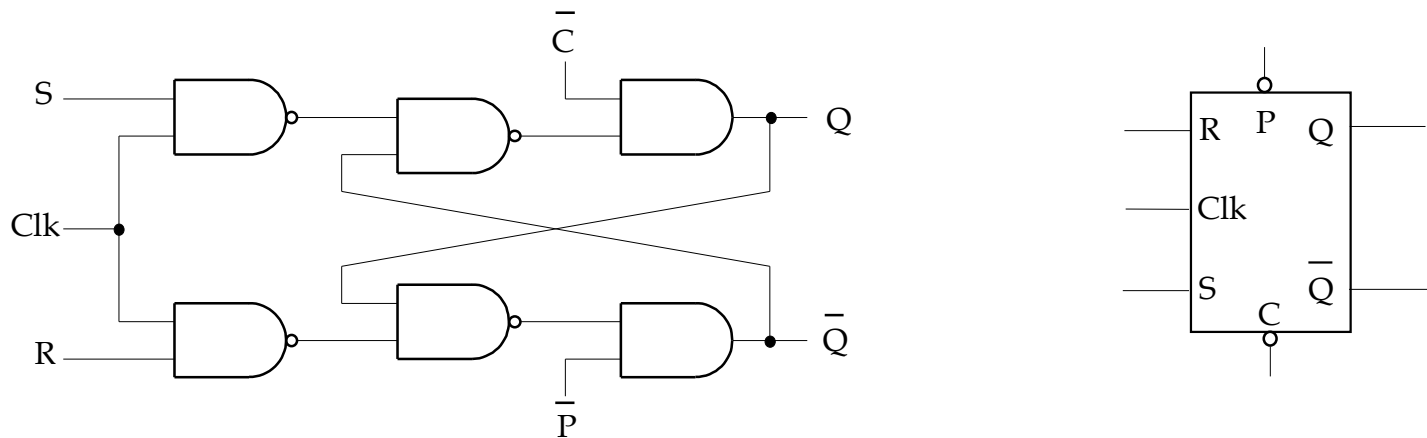
2.2 Biestable RS. Biestable RS Síncrono



2.2 Biestable RS. Biestable RS Síncrono con entradas Asíncronas

Las *entradas asíncronas* fuerzan la salida del biestable a un estado particular, independientemente del valor que coloquemos en las entradas S y R y de la entrada de Reloj. La activación de la entrada asíncrona **PRESET** (P) fuerza la aparición en la salida de un Nivel ALTO. La activación de la entrada **CLEAR** (C) fuerza la salida a Nivel BAJO.

Biestable RS con entradas asíncronas activas a NIVEL BAJO:



2.2. Biestables. Biestable JK.

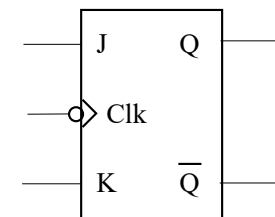
Posee dos entradas, llamadas J y K, y dos salidas complementarias, que denominaremos Q y \bar{Q} . Su modo de funcionamiento queda definido por la siguiente tabla de verdad. Siempre son activos por flanco.

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

Tabla Resumen:

J	K	Q(t+1)	Acción
0	0	Q(t)	No cambia
0	1	0	Reset
1	0	1	Set
1	1	$\bar{Q}(t)$	Basculación

Bloque funcional:



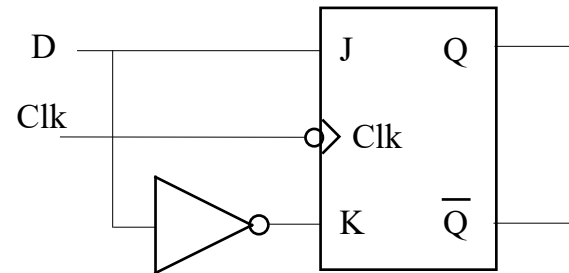
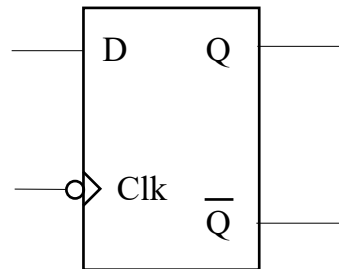
2.2. Biestables. Biestable D.

Posee una entrada, llamada D (Data), y dos salidas complementarias, que denominaremos Q y \bar{Q} . Su modo de funcionamiento queda definido por la siguiente tabla de verdad.

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

Tabla Resumen:

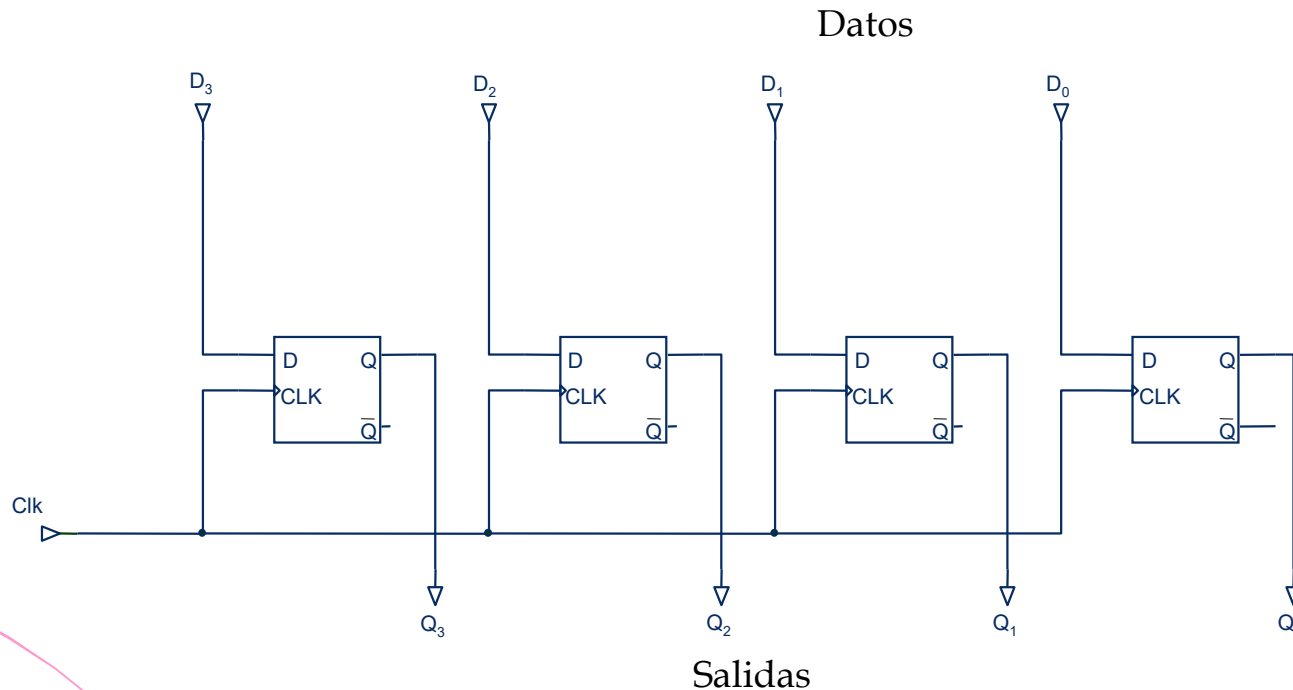
D	Q(t+1)
0	0
1	1



2.3 Registros. Definición. Tipos

Un registro es un grupo de biestables interconectados entre si. Los clasificamos, según su función, en registros de almacenamiento y registros de desplazamiento.

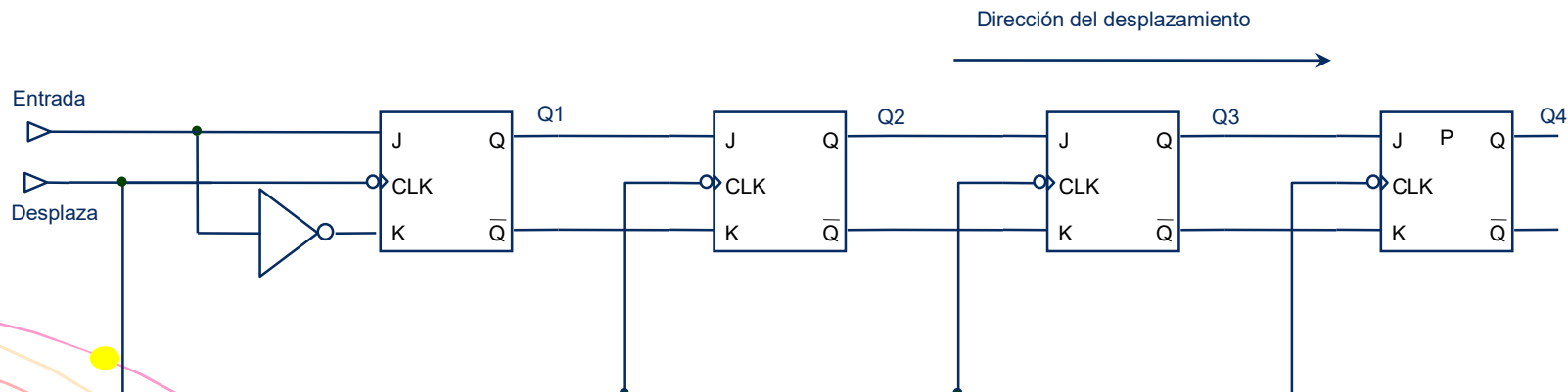
La finalidad de un *registro de almacenamiento* es la de mantener temporalmente la información para que pueda ser tratada. Todos los biestables que lo componen comparten la misma señal de reloj.



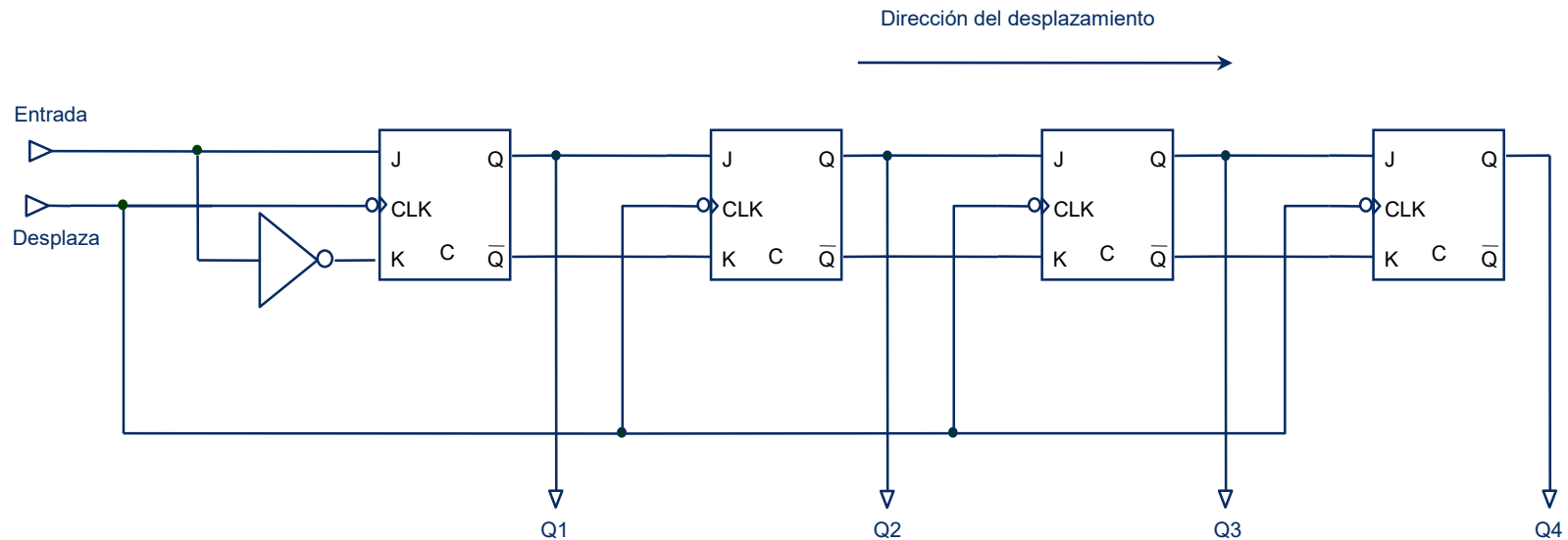
2.3 Registros. Definición. Tipos

Un *registro de desplazamiento* nos permitirá almacenar y mover la información entre las etapas que lo componen de una forma preestablecida y con una finalidad específica. Así, hablaremos de registros de desplazamiento con entrada y salida serie, con entrada serie y salida paralelo y de entrada paralelo y salida serie.

Registro de desplazamiento con entrada y salida serie



Registro de desplazamiento con entrada serie y salida paralelo



2.3 Contadores. Definición.

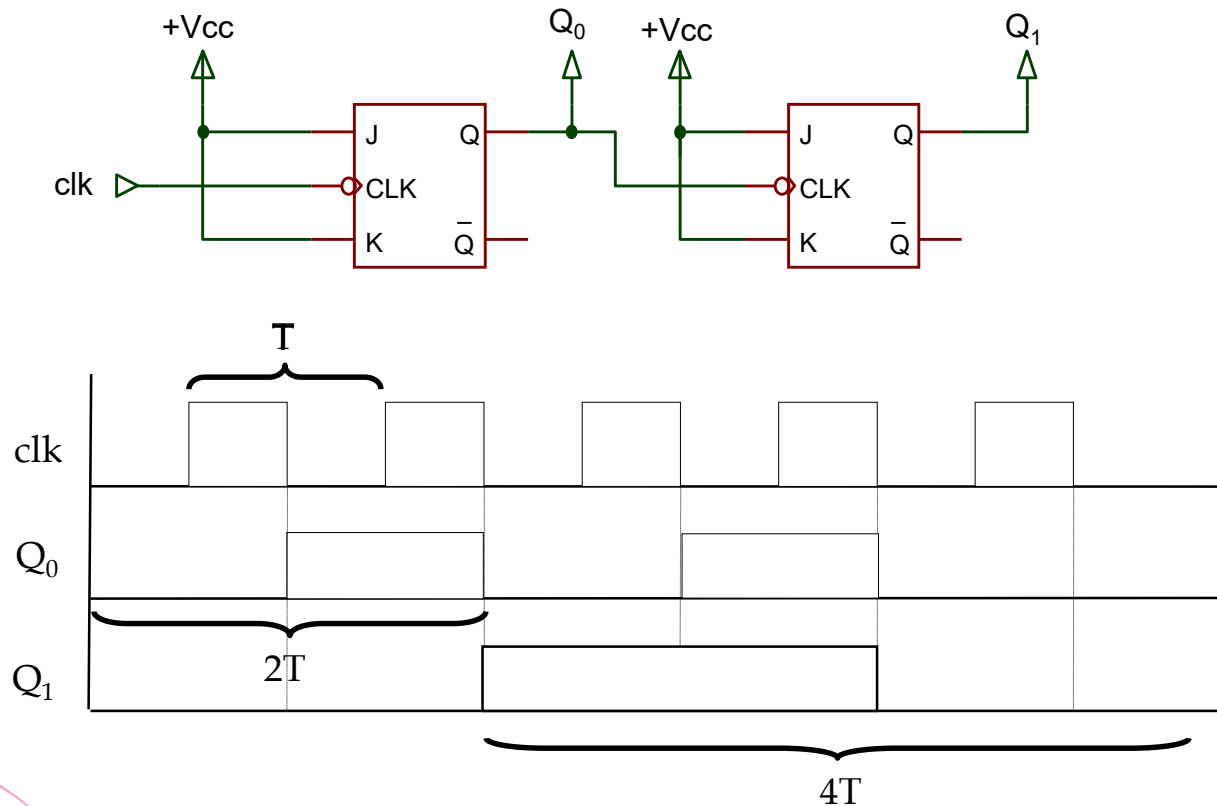
Un *contador* es un tipo de circuito secuencial que sigue una secuencia de estados preestablecida ante la aplicación de pulsos de reloj. Los dividiremos en *contadores síncronos* y *asíncronos*.

- En un contador asíncrono las entradas de reloj de los biestables que lo forman no son comunes para todos ellos. También se denominan *contadores de rizado*.
- En un contador síncrono solo existe una única entrada de reloj que activa simultáneamente todos los biestables que lo constituyen.

En general, construiremos un contador asíncrono de n -bits conectando en cascada la salida del biestable i (Q_i) a la entrada del reloj del siguiente (Clk_{i+1})

2.3 Contadores Asíncronos.

Una aplicación inmediata de los contadores asíncronos es su empleo como divisores de frecuencia. Con n biestables obtendremos una división de 2^n .



2.3 Contadores Asíncronos.

Con n biestables podremos tener hasta 2^n estados distintos. A la cantidad de estados que forman parte de la salida de un contador la denominaremos *módulo* del contador.

Para construir un contador de módulo K (con $K < 2^n$) haremos uso de las entradas asíncronas. Cuando lleguemos al valor límite, lo reiniciaremos mediante las entradas de Preset y Clear, según sea el caso.

Ejemplo: Contador de décadas (modulo 10):

