

# Circuitos Secuenciales

## Apunte N° 6

# Circuitos Combinacionales

Hasta ahora hemos desarrollado sistemas y circuitos combinacionales donde las salidas dependían únicamente de las entradas en ese momento. A una misma combinación de entradas responden siempre con la misma salida.

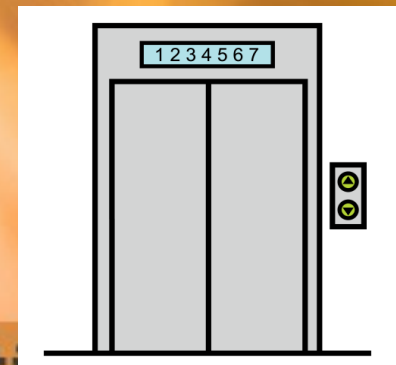
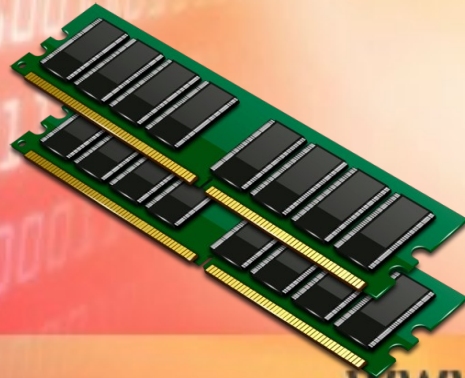
Los circuitos combinacionales tienen muchas limitaciones debido a que no son capaces de reconocer el orden en que se van presentando las combinaciones de entradas con respecto al tiempo, es decir, no pueden reconocer una secuencia de combinaciones, ya que no poseen una manera de almacenar información pasada, es decir no poseen memoria.

En los circuitos combinacionales las salidas son funciones que dependen únicamente de las entradas actuales al circuito. Pero sin embargo, en muchos sistemas digitales esto no es suficiente, siendo necesario además circuitos capaces de almacenar información, puesto que en ellos las salidas no quedan definidas únicamente por las entradas actuales, sino que dependen también de cuáles fueron los valores de éstas en el pasado. Se trata de los circuitos secuenciales.

# Circuitos Secuenciales

En los circuitos secuenciales, también llamados Maquinas de Estados Finitos, la salida no solo depende de las entradas presentes, también dependerá de la historia pasada, de lo que sucedió antes. La historia de las entradas anteriores en un momento dado se encuentra resumida en el estado del circuito, el cual se expresa en un conjunto de variables de estado.

El circuito secuencial debe ser capaz de mantener su estado durante algún tiempo, para ello se hace necesario el uso de dispositivos de memoria. Los dispositivos de memoria utilizados en circuitos secuenciales pueden ser tan sencillos como un simple retardador (inclusive, se puede usar el retardo natural asociado a las compuertas lógicas) o tan complejos como un circuito completo de memoria denominado multivibrador biestable o Flip-Flop.



# Circuitos Secuenciales

Un sistema secuencial dispone de elementos de memoria cuyo contenido puede cambiar a lo largo del tiempo.

El estado de un sistema secuencial viene dado por el contenido de sus elementos de memoria.

Es frecuente que en los sistemas secuenciales exista una señal que inicia los elementos de memoria con un valor determinado: señal de inicio (reset).

La señal de inicio determina el estado del sistema en el momento del arranque (normalmente pone toda la memoria a cero).

La salida en un instante concreto viene dada por la entrada y por el estado anterior del sistema.

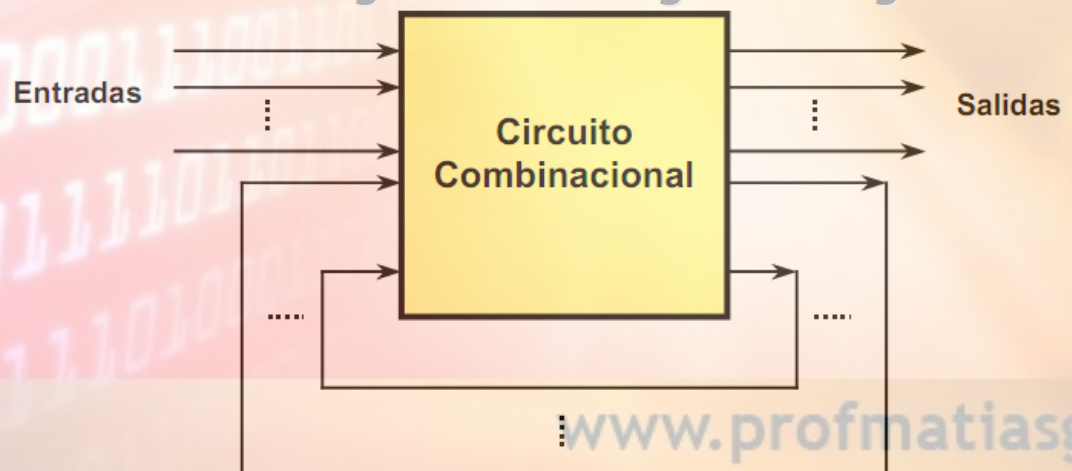
El estado actual del sistema, junto con la entrada, determinará el estado en el instante siguiente realimentación.

# El factor tiempo

El circuito secuencial debe ser capaz de mantener su estado durante algún tiempo, para ello se hace necesario el uso de dispositivos de memoria. Los dispositivos de memoria utilizados en circuitos secuenciales pueden ser tan sencillos como un simple retardador (inclusive, se puede usar el retardo natural asociado a las compuertas lógicas) o tan complejos como un circuito completo de memoria denominado multivibrador biestable o Flip-Flop.

Entra un factor que no se había considerado en los circuitos combinacionales, dicho factor es el tiempo.

Un circuito secuencial puede entenderse simplemente como un circuito combinacional en el cual las salidas dependen tanto de las entradas como de las salidas en instantes anteriores, esto implica una retroalimentación de las salidas como se muestra en diagrama de la siguiente figura



# Tipos de Circuitos Secuenciales

Los circuitos secuenciales se clasifican de acuerdo a la manera como manejan el tiempo en Circuitos Secuenciales Síncronos y Circuitos Secuenciales Asíncronos.

- ◆ **Síncronos:** sólo pueden cambiar de estado en determinados instantes de tiempo, es decir, están “sincronizados” con una señal de reloj (Clock). El sistema sólo hace caso de las entradas en los instantes de sincronismo.
- ◆ **Asíncronos:** pueden cambiar de estado en cualquier instante de tiempo en función de cambios en las señales de entrada, no usan elementos especiales de memoria, pues se sirven de los retardos propios (tiempos de propagación) de las compuertas lógicas usados en ellos.  
Esta manera de operar puede ocasionar algunos problemas de funcionamiento, ya que estos retardos naturales no están bajo el control del diseñador y además no son idénticos en cada compuerta lógica.

# Sistemas Síncronos o con Clock

Son sistemas que actúan bajo un control de tiempo, este control se denomina reloj (clock).

**Clock:** es una señal que se alterna entre los valores lógicos 0 y 1 en un periodo regular.

Clock 1



Clock 2



**Tipos de sincronismo:**

- ◆ **Sincronismo por nivel (alto o bajo):** el sistema hace caso de las entradas mientras el reloj esté en el nivel activo (alto o bajo).
- ◆ **Sincronismo por flanco (de subida o de bajada):** el sistema hace caso de las entradas y evoluciona justo cuando se produce el flanco activo (de subida o de bajada).

# El Clock

El Periodo (T): es el tamaño en tiempo de un ciclo.

La Frecuencia (f): es el inverso del periodo,  $1/T$  y está dada en Hertz (Hz).

Ejemplo:

Una señal con frecuencia de 200 MHz, corresponde a una señal que tenga un periodo de 5 ns.

En la mayoría de los sistemas sincrónicos, los cambios ocurren en las transiciones donde la señal cambia de 0 a 1 ó de 1 a 0

## Clock





# Tablas y diagramas de estados

Ejemplo de un sistema secuencial:

EJ: Un sistema con una entrada  $x$  y una salida  $z$ , de tal forma que  $z = 1$ , si  $x$  ha sido 1 por tres pulsos de clock consecutivos.

Para este ejemplo, el sistema debe almacenar en memoria la información de los últimos tres estados de la entrada y producir una salida basada en esa información.

Estado: Lo que se almacena en la memoria es el estado del sistema.

En este ejemplo, la salida depende únicamente del estado del sistema y que se haya seguido el patrón definido en la entrada del sistema.

En este tipo de Máquinas de Estado que sólo dependen del estado actual del sistema son llamadas de Modelos Moore ó Máquinas Moore, debido a Edward F. Moore, un pionero de las Máquinas de estados, quien escribió Gedanken-experiments on Sequential Machines, pp 129 – 153, Automata Studies, Annals of Mathematical Studies, no. 34, Princeton University Press, Princeton, N. J., 1956.

# Tablas y diagramas de estados

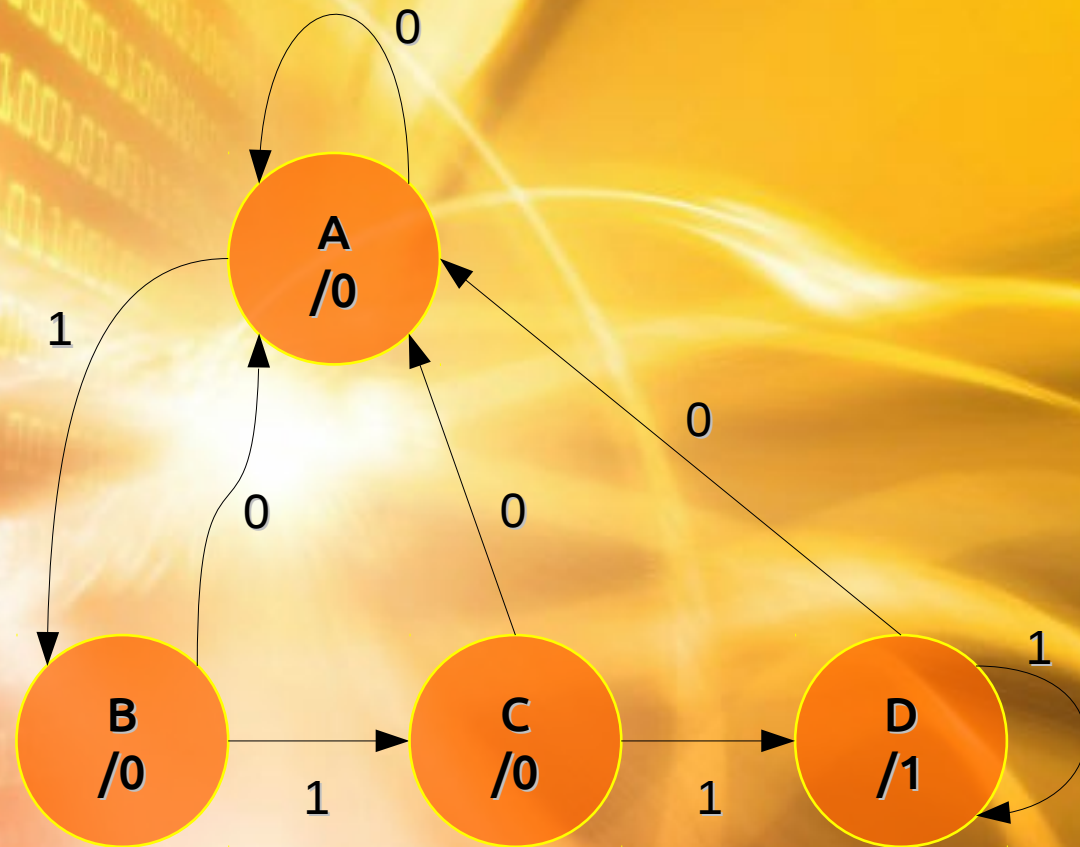
No abordaremos todavía el diseño de un sistema secuencial, pero daremos las herramientas necesarias para describirlo.

- ◆ **Tabla de Estados:** es una tabla que describe las transiciones de una máquina de estados finitos, en otras palabras, muestra las relaciones funcionales entre las entradas, salidas y estados de la memoria. Para cada combinación y cada estado, indica cual será la salida y cual será el próximo estado después del siguiente pulso de clock.
- ◆ **Diagrama de Estados:** Es una representación gráfica del comportamiento del sistema, mostrando cada combinación de entrada y cada estado, de la misma forma muestra el resultado de la salida y el valor del estado siguiente después de un pulso de clock.

A continuación veremos la tabla y el diagrama de estados para el EJ.

# Tablas y diagramas de estados

| Estado Presente | Estado Siguiete |       | Salida |
|-----------------|-----------------|-------|--------|
|                 | x = 0           | x = 1 |        |
| A               | A               | B     | 0      |
| B               | A               | C     | 0      |
| C               | A               | D     | 0      |
| D               | A               | D     | 1      |



En el futuro nos referiremos al Estado Presente por el símbolo  $q$  y el Estado Siguiete por el símbolo  $q^{n+1}$ .

# Timing Trace (rastreo en el tiempo)

Un timing trace, es un conjunto de valores para las entradas y salidas arreglados en una forma consecutiva con relación a los pulsos de clock. Es usado normalmente para explicar o clarificar el comportamiento de un sistema.

|          |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|----------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| <i>x</i> | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 |   |   |
| <i>q</i> | ? | A | B | C | A | B | C | D | A | A | B | A | B | C | D | D | D | A | A | ? |
| <i>z</i> | ? | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 |

# Biestables

Un biestable es un dispositivo capaz de almacenar un bit (1 ó 0).

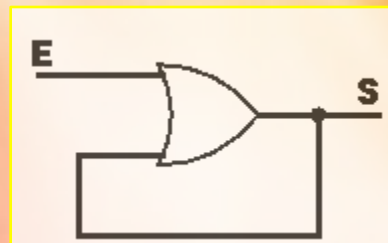
Principio de funcionamiento de un biestable: Utilizando realimentación entre puertas se puede mantener ( almacenar ) un valor estable hasta que cambien las condiciones de entrada.

Un circuito Flip-Flop puede mantener un estado binario en forma indefinida (en cuanto se suministre potencia al circuito) hasta que recibe la dirección de una señal de entrada para cambiar de estado.

La diferencia principal entre los diversos tipos de Flip-Flops esta en el número de entradas que poseen y en la manera en la cual las entradas afectan el estado binario.



Estas dos puertas NOT mantienen un valor estable (no puede modificarse porque no hay entradas)

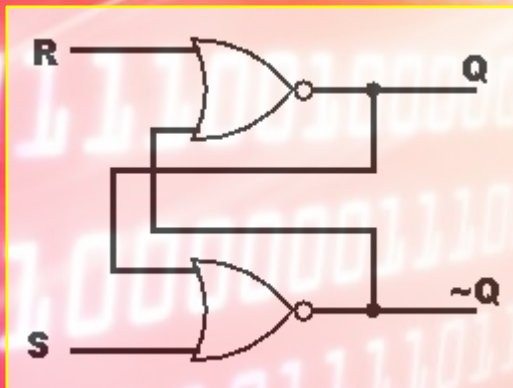


$$S_{(t+1)} = S_{(t)} + E$$

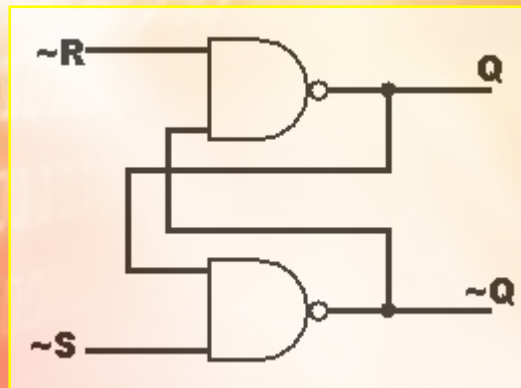
Al poner E a 1, S valdrá 1 (ese 1 ya no se puede borrar)

# Latch

Un latch es un tipo de dispositivo lógico biestable o multivibrador de almacenamiento, construido con dos o más compuertas con realimentación. Un latch S-R (Set-Reset) con entrada activa a nivel ALTO se compone de dos puertas NOR acopladas; un latch S-R con entrada activa a nivel BAJO está formado por dos puertas NAND conectadas. Observe que la salida de cada puerta se conecta a la entrada de la puerta opuesta. Esto origina la realimentación (feedback) regenerativa característica de todos los latches y Flip-Flops.



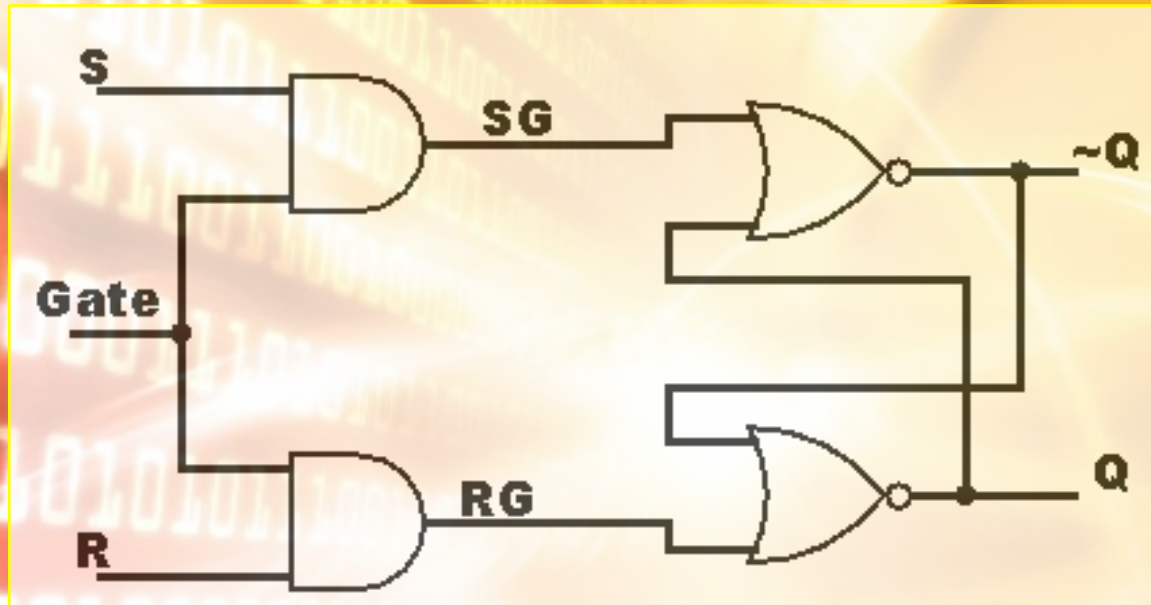
Latch S-R con  
entrada activa a  
nivel Alto



Latch  $\bar{S}\bar{R}$  con  
entrada activa a  
nivel Bajo

S = Set  
R = Reset

# Latch con gatillo (Gated)



En este latch, cuando la señal del gate es inactiva, tanto SG y RG serán 0 y el latch permanece sin cambios. Únicamente cuando la señal del gate es 1 el latch podrá recibir el valor 0 ó 1 así como el latch anterior.

# Biestables Flip-Flop

Los circuitos secuenciales básicos que funcionan también como unidades de memoria elementales se denominan multivibradores biestables (por tener dos estados estables –alto y bajo-), también conocidos como Flip-Flops.

El Flip-Flop es un dispositivo de almacenamiento binario con colck.

Bajo operaciones normales este dispositivo almacenará un 1 ó un 0 y sólo cambiarán estos valores en el momento que ocurra una transición del clock.

Las transiciones que pueden producir cambios en el sistema pueden ser cuando el clock va de 0 a 1, disparo por rampa de subida (leadign-edge triggered), o cuando el clock va de 1 a 0, disparo por rampa de bajada (trailing-edge triggered).

Estos circuitos pueden construirse con compuertas NAND o dos compuertas NOR. Cada circuito forma un Flip-Flop básico, La conexión y el acoplamiento cruzado mediante la salida de una compuerta a la entrada de otra constituye una trayectoria de retroalimentación. Por esta razón los circuitos se clasifican como secuenciales asíncronos. Cada Flip-Flop tiene dos salidas Q y Q negada.

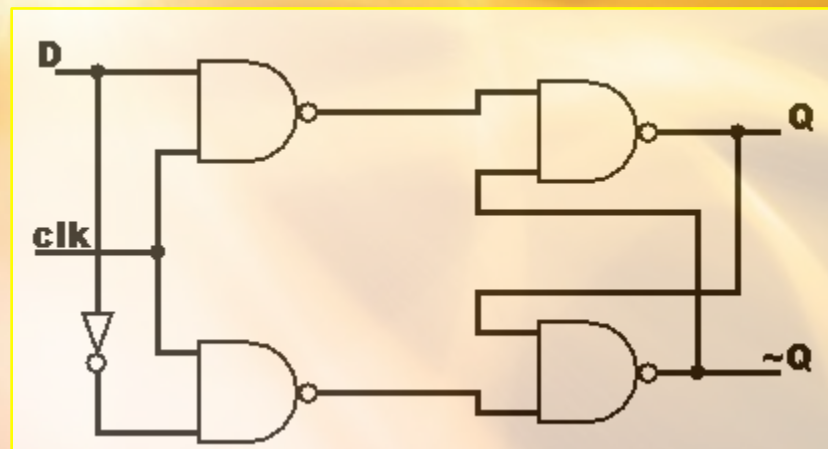
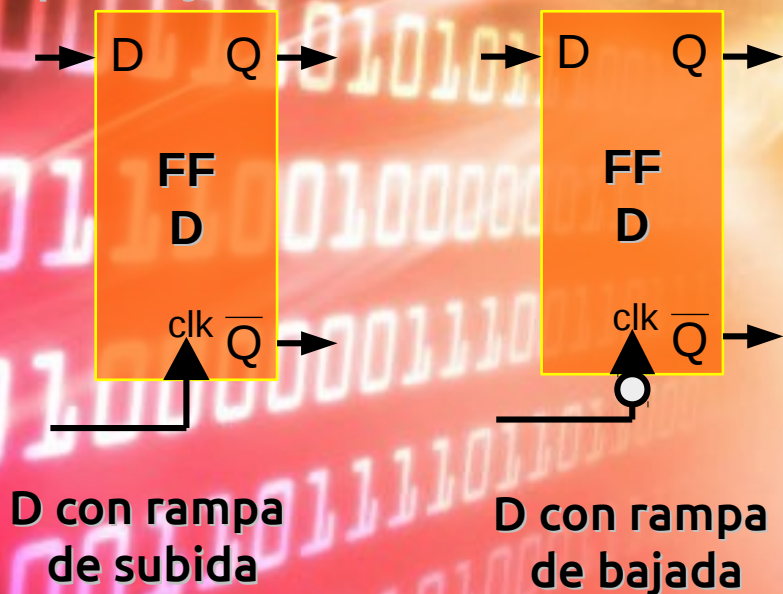


# Flip-Flop tipo D

El Flip-Flop tipo D es el más usado y es encontrado comúnmente en dispositivos lógicos programables.

Cuenta con una única entrada D que se copia al interior del biestable en los instantes de sincronismo.

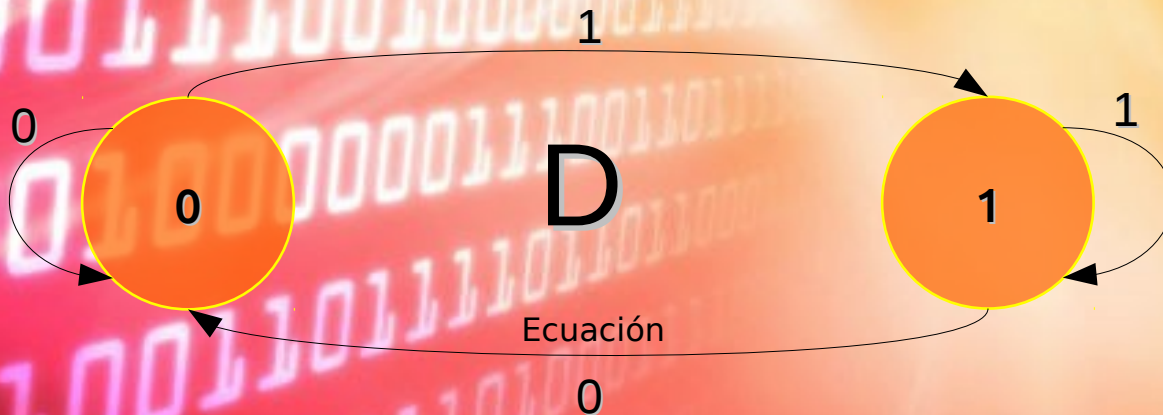
El nombre proviene de Delay (retardo), ya que su salida es un reflejo de lo que hay en la entrada con un retardo de un ciclo de clock.



# Flip-Flop tipo D – Tabla de comportamiento y diagrama de estados

| D | q | q <sup>n+1</sup> |
|---|---|------------------|
| 0 | 0 | 0                |
| 0 | 1 | 0                |
| 1 | 0 | 1                |
| 1 | 1 | 1                |

| D | q <sup>n+1</sup> |
|---|------------------|
| 0 | 0                |
| 1 | 1                |



$$q^{n+1} = D$$

Ecuación

La entrada D es activa por nivel alto.

# Comportamiento de un Flip-Flop D con rampa de bajada

Diagrama de tiempo



La salida no se verá afectada, ya que el valor de la entrada D solo es relevante en el instante de la rampa de bajada

# Flip-Flop RS (Reset-Set)

El FF RS es un dispositivo con dos entradas (Set y Reset) y una variable de estado o salida (Q) capaz de “guardar” un bit de información y funciona como sigue:

- ◆ Si su entrada Set se activa su estado Q se pone en Alto
- ◆ Si su entrada Reset se activa su estado Q se pone en Bajo
- ◆ Si no se activa ni Set ni Reset su estado no cambia
- ◆ Por supuesto, no se permite activar Set y Reset simultáneamente.

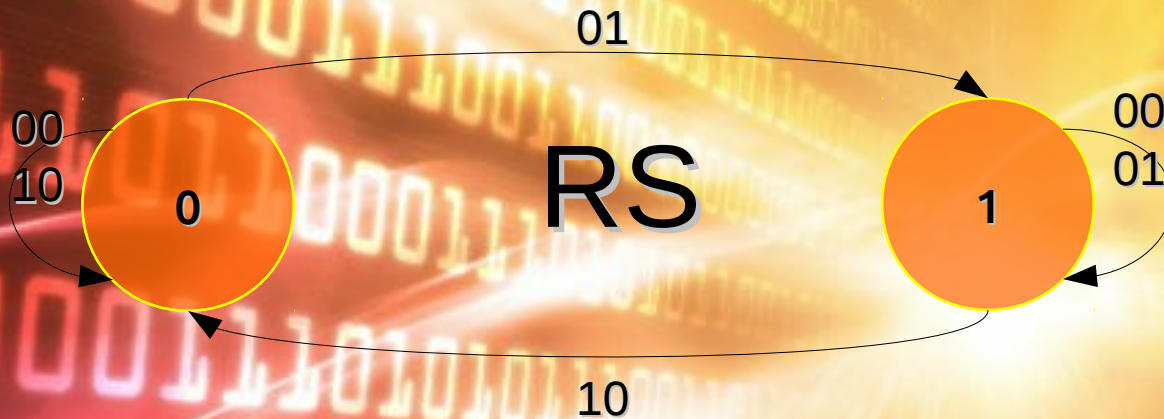
Tabla de comportamiento

| R | S | q | q <sup>n+1</sup> |
|---|---|---|------------------|
| 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 | -                |
| 1 | 1 | 1 | -                |

| R | S | q <sup>n+1</sup> |
|---|---|------------------|
| 0 | 0 | q                |
| 0 | 1 | 1                |
| 1 | 0 | 0                |
| 1 | 1 | -                |

Tabla de estados

# Flip-Flop RS – Diagrama de estados y Ecuación



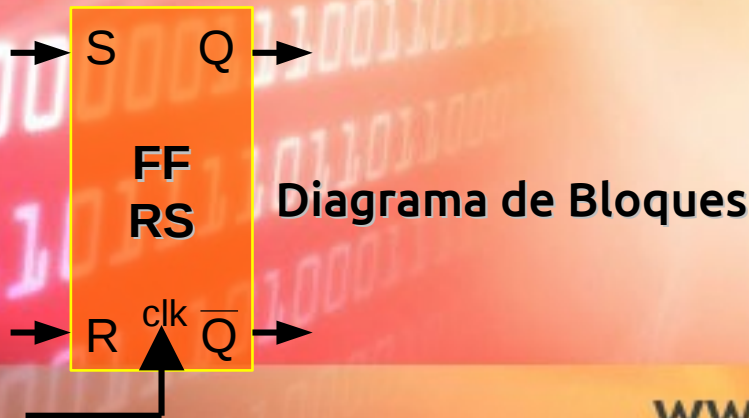
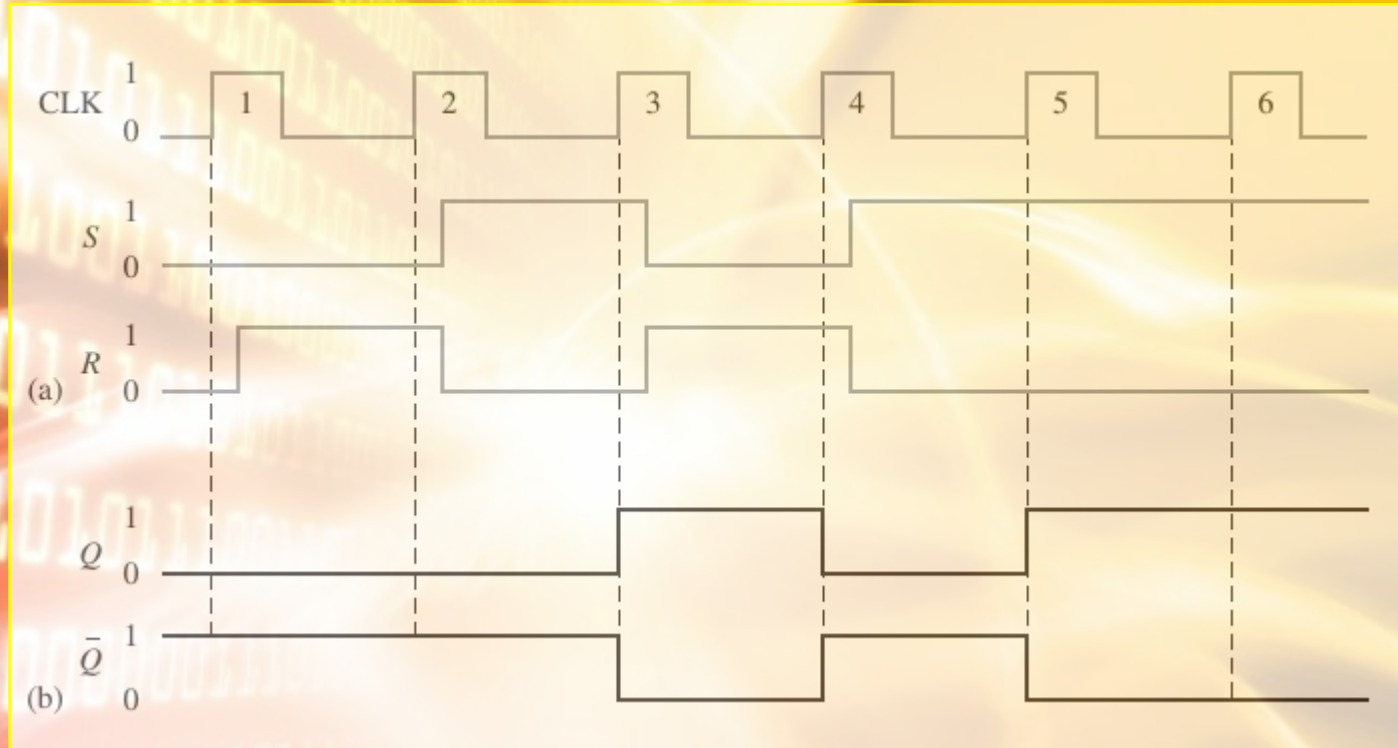
$$q^{n+1} = S + R\bar{q}$$

| q \ RS | 00 | 01 | 11 | 10 |
|--------|----|----|----|----|
| 0      | 0  | 0  | X  | 1  |
| 1      | 1  | 1  | X  | 1  |

| q | q <sup>n+1</sup> | R | S |
|---|------------------|---|---|
| 0 | 0                | X | 0 |
| 0 | 1                | 0 | 1 |
| 1 | 0                | 1 | 0 |
| 1 | 1                | 0 | X |

Tabla de Transiciones

# Flip-Flop RS – Diagrama de tiempo



# Flip-Flop T (Toggle)

Tiene una entrada T, de tal forma que si  $T = 1$ , el Flip-Flop cambia el valor del estado actual y si  $T = 0$ , el estado permanece sin cambios.

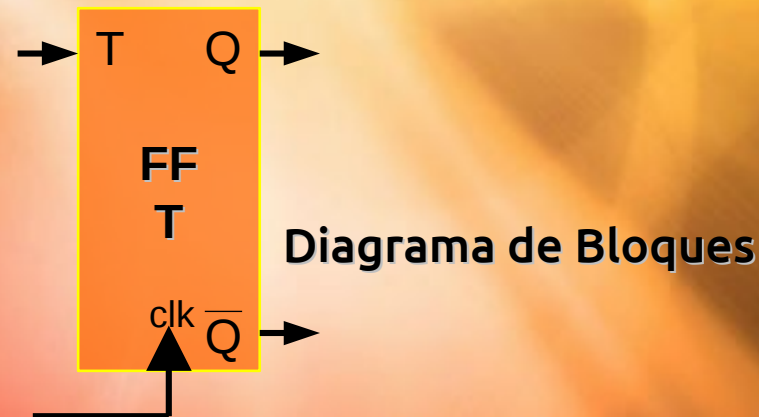
Tablas de Comportamiento

Tabla de comportamiento

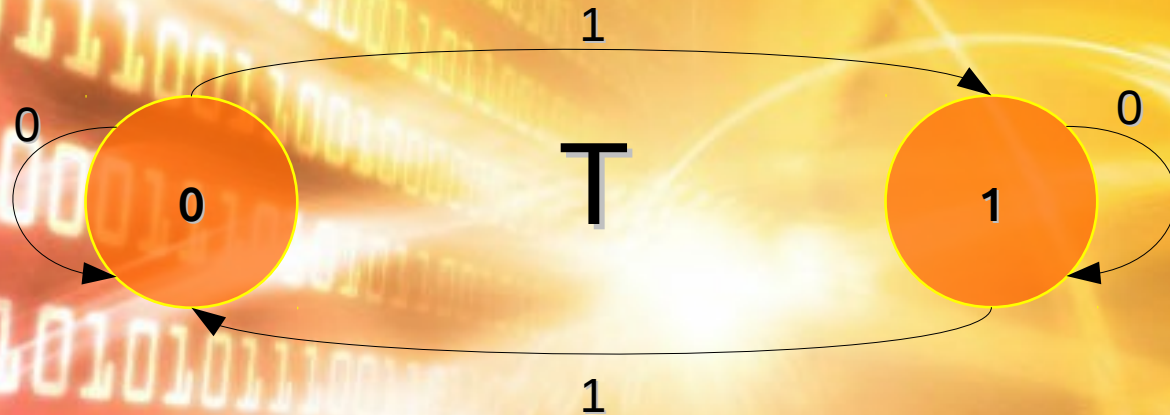
| T | q | q <sup>n+1</sup> |
|---|---|------------------|
| 0 | 0 | 0                |
| 0 | 1 | 1                |
| 1 | 0 | 1                |
| 1 | 1 | 0                |

Tabla de estados

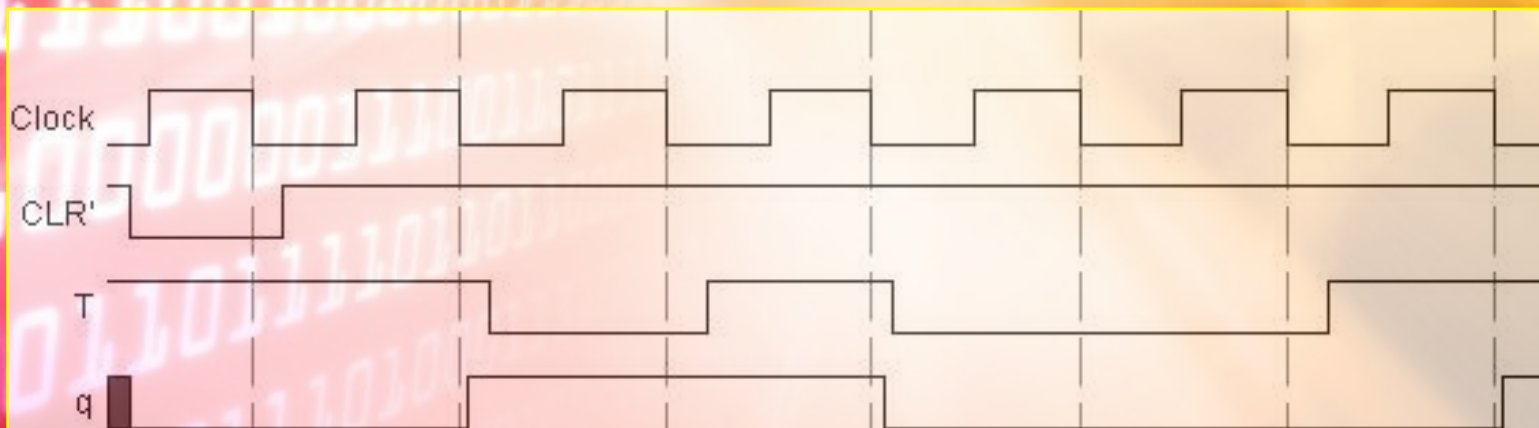
| T | q <sup>n+1</sup> |
|---|------------------|
| 0 | q                |
| 1 | $\bar{q}$        |



# Flip-Flop T – Diagrama de estados y de tiempo



*Ecuación para el comportamiento  $q^{n+1} = T \oplus q$*





# Flip-Flop JK

Es una combinación del RS y del T, siendo así, su comportamiento es como el RS, con excepción cuando sus entradas  $J = K = 1$  provoca que el Flip-Flop cambie de estado, como si fuera un Flip-Flop T, basculación.

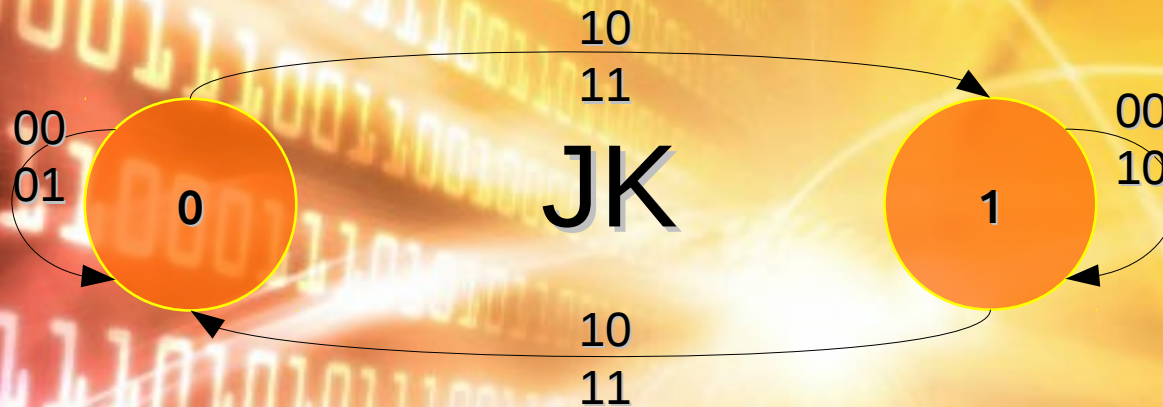
Tabla de comportamiento

| J | K | q | q <sup>n+1</sup> |
|---|---|---|------------------|
| 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 de estados

| J | K | q <sup>n+1</sup> |
|---|---|------------------|
| 0 | 0 | q                |
| 0 | 1 | 0                |
| 1 | 0 | 1                |
| 1 | 1 | $\bar{q}$        |

# Flip-Flop JK – Diagrama de estados y Ecuación



| q\JK | 00 | 01 | 11 | 10 |
|------|----|----|----|----|
| 0    |    |    | 1  | 1  |
| 1    | 1  |    |    | 1  |

$$q^{n+1} = J\bar{q} + \bar{K}q$$

# Flip-Flop JK – Diagrama de tiempo

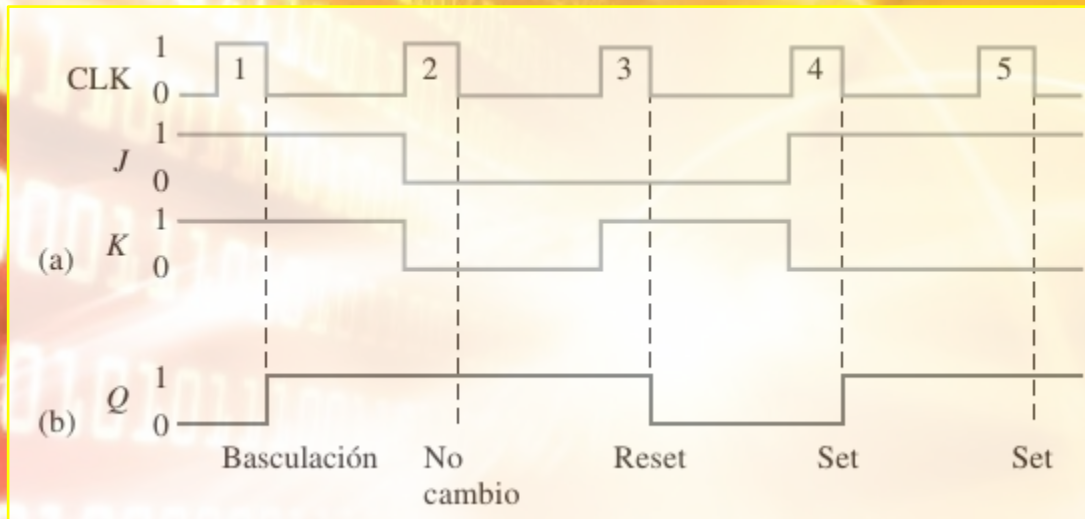


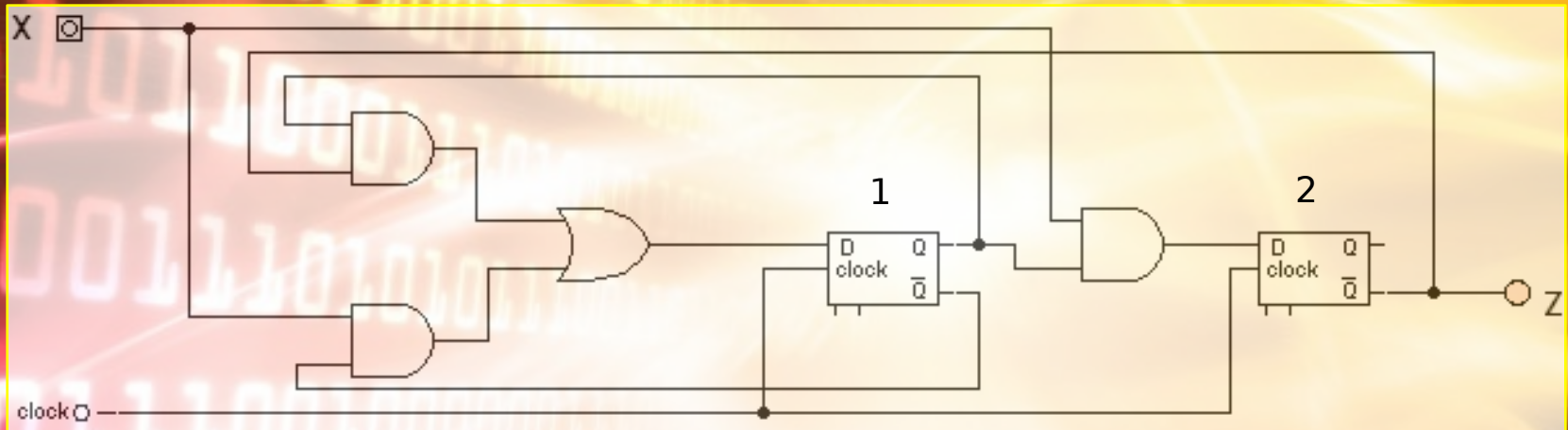
Diagrama de Bloques

| q | q <sup>n+1</sup> | J | K |
|---|------------------|---|---|
| 0 | 0                | 0 | X |
| 0 | 1                | 1 | X |
| 1 | 0                | X | 1 |
| 1 | 1                | X | 0 |

Tabla de Transiciones

# Análisis de Sistemas Secuenciales

Modelo tipo Moore con Flip-Flops tipo D



Del circuito encontramos:

$$D_1 = q_1 \cdot \bar{q}_2 + X \cdot \bar{q}_1$$

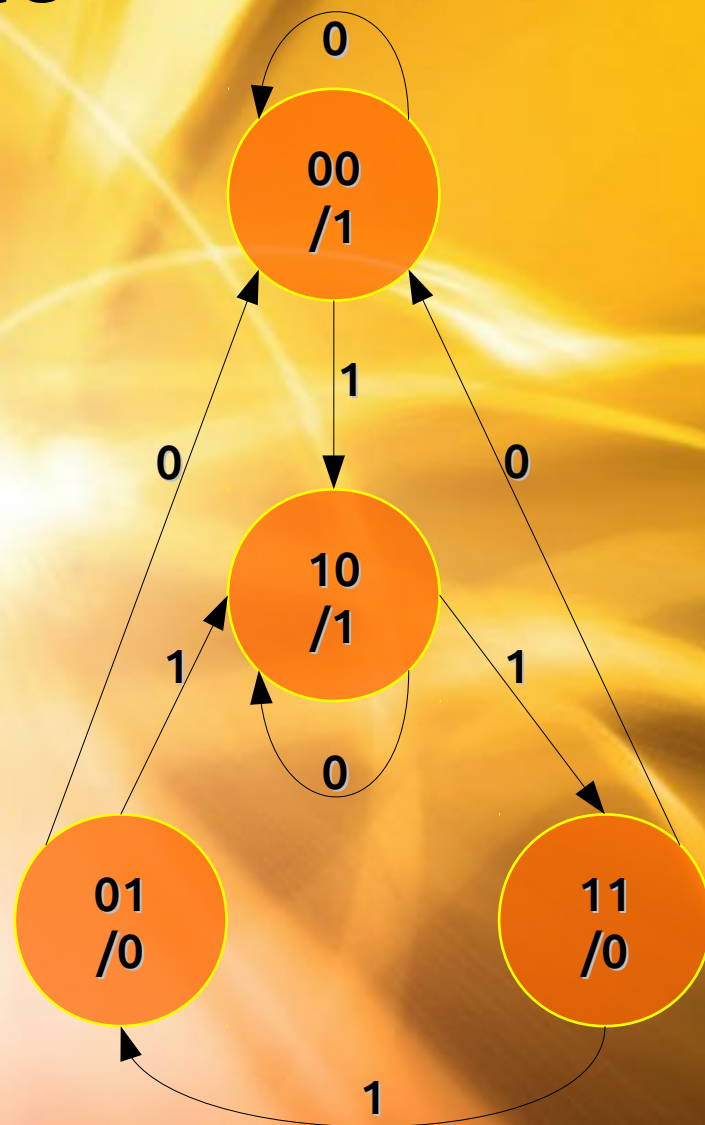
$$D_2 = X \cdot q_1$$

$$Z = \bar{q}_2$$

# Análisis de Sistemas Secuenciales

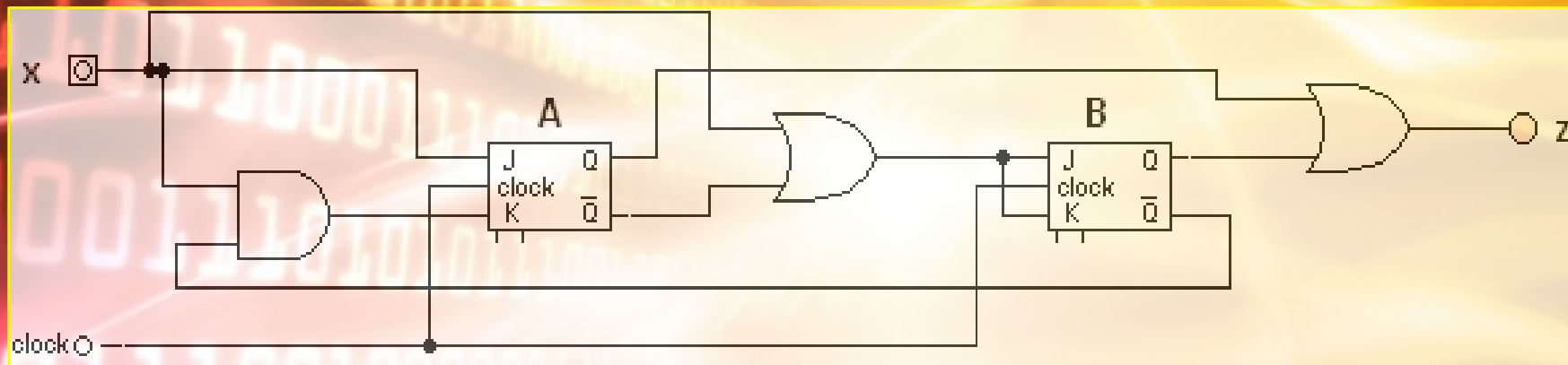
Tabla y diagrama de estados del circuito

| $q_1$ | $q_2$ | $q_1^{n+1} q_2^{n+1}$ |       | Z |
|-------|-------|-----------------------|-------|---|
|       |       | x = 0                 | x = 1 |   |
| 0     | 0     | 00                    | 10    | 1 |
| 0     | 1     | 00                    | 10    | 0 |
| 1     | 0     | 10                    | 11    | 1 |
| 1     | 1     | 00                    | 01    | 0 |



# Análisis de Sistemas Secuenciales

## Modelo tipo Moore con Flip-Flops tipo JK



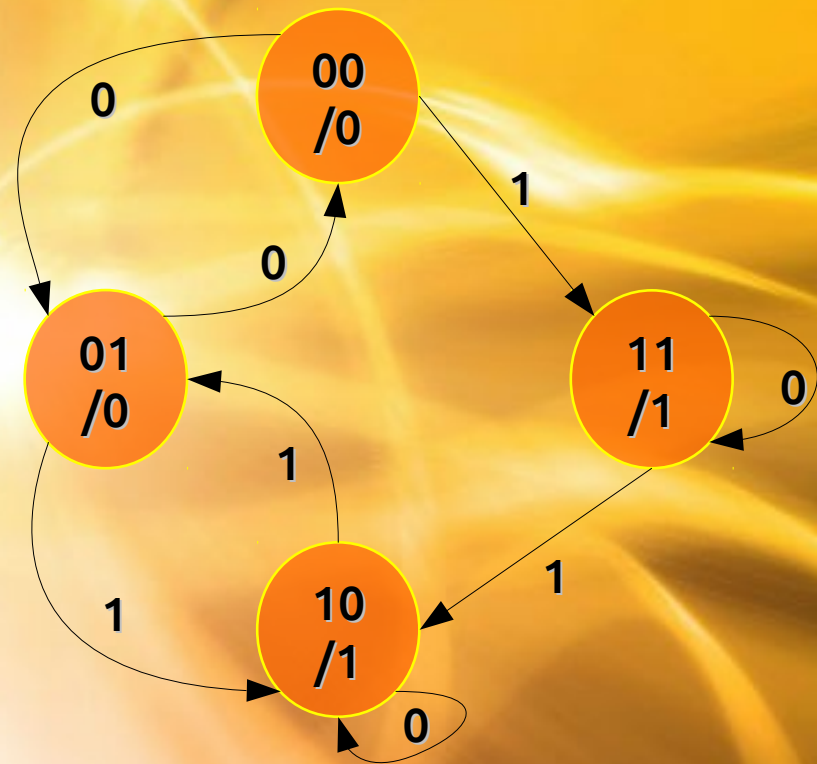
Este es un circuito de modelo tipo Moore, ya que la salida  $z$ , que es igual a  $A + B$ , es una función del estado, o sea, el contenido de los Flip-Flops, y no de la entrada  $x$ .

$$\begin{aligned} J_A &= X \\ K_A &= X \cdot \overline{Q_B} \\ J_B &= K_B = X + \overline{Q_A} \\ Z &= Q_A + Q_B \end{aligned}$$

# Análisis de Sistemas Secuenciales

Tabla y diagrama de estados del circuito

| A | B | $A^{n+1} B^{n+1}$ |       | Z |
|---|---|-------------------|-------|---|
|   |   | x = 0             | x = 1 |   |
| 0 | 0 | 0 1               | 1 1   | 0 |
| 0 | 1 | 0 0               | 1 0   | 1 |
| 1 | 0 | 1 0               | 0 1   | 1 |
| 1 | 1 | 1 1               | 1 0   | 1 |

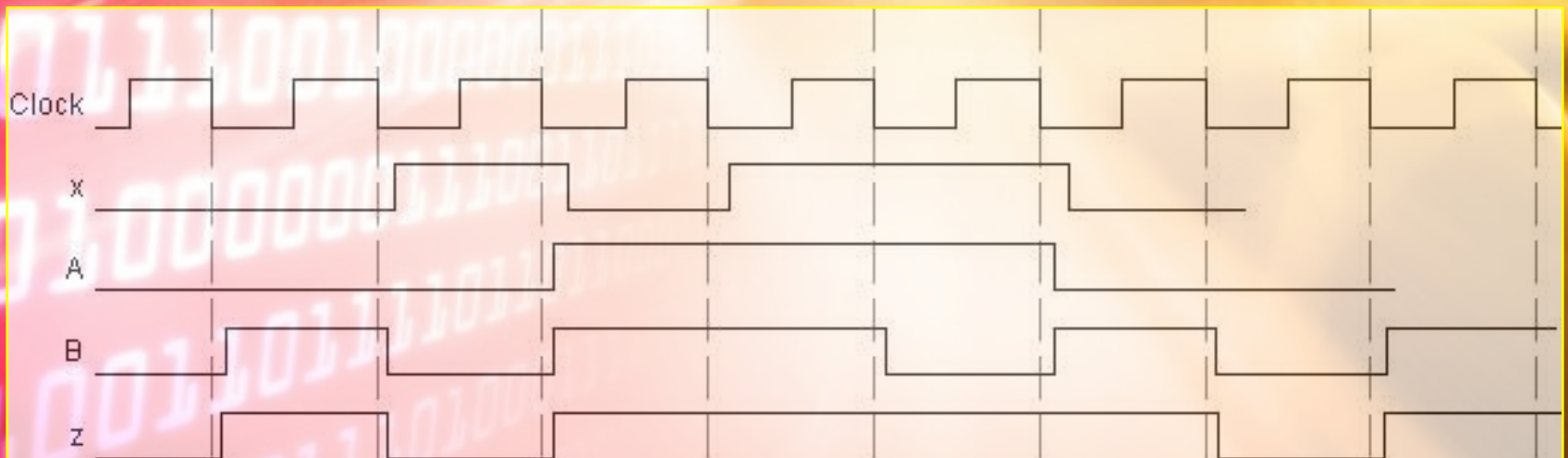


Para completar la tabla hay que tener en cuenta las ecuaciones de entrada de los Flip-Flops y el funcionamiento de cada uno de ellos para determinar el estado siguiente.

# Análisis de Sistemas Secuenciales

Trazado en el tiempo y Diagrama de tiempos

|   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|
| x | 0 | 0 | 1 | 0 | 1 | 1 | 0 |   |   |
| A | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 |   |
| B | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 |
| z | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 |

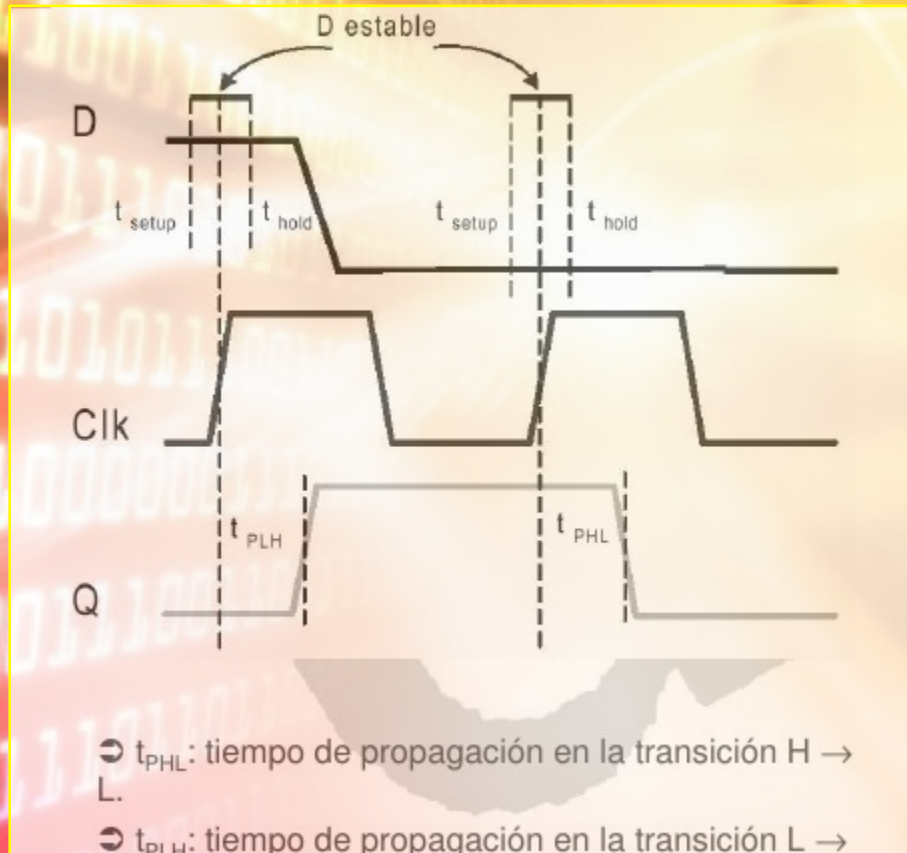




# Parámetros temporales de los biestables

- ◆ Tiempo de propagación o retardo (delay time): Tiempo necesario para que el efecto de un cambio en la entrada se haga estable en la salida.
- ◆ Tiempo de establecimiento (setup time): Tiempo mínimo anterior al flanco de disparo en que las entradas no deben variar (tiempo necesario para que el biestable asiente las entradas antes del flanco).
- ◆ Tiempo de mantenimiento (hold time): Es el tiempo máximo posterior al flanco de disparo en que las entradas no deben variar (tiempo necesario para que el biestable procese las entradas).
- ◆ Anchura del reloj  $t_{WH}$  y  $t_{WL}$ : Duración mínima necesaria para los pulsos de nivel alto y bajo respectivamente.
- ◆ Frecuencia máxima  $f_{max}$ : Máxima frecuencia permitida al reloj del biestable. Si se supera, el biestable puede funcionar mal.
- ◆ Tiempo de preset y clear: Es el tiempo mínimo que debe durar el nivel activo de las entradas asíncronas de puesta a 1 y puesta a 0 para que el biestable tome el valor pertinente. Este tiempo suele estar incluido en  $t_{pLH}$  y  $t_{pHL}$  respectivamente.

# Parámetros temporales de los biestables



# Flip-Flops con “Clear” y “Preset”

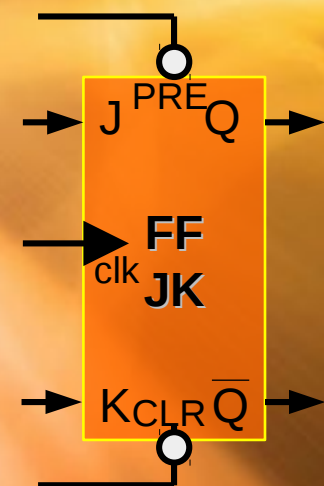
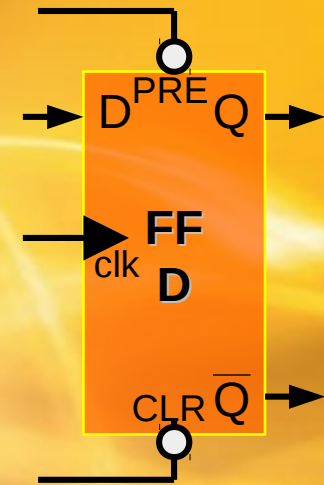
En los Flip-Flops que acabamos de estudiar, el RS, el D y el JK, se dice que sus entradas son entradas síncronas, ya que los datos de estas entradas condicionan la salida de los Flip-Flops sólo durante el flanco de disparo del impulso de reloj; esto significa que los datos se transfieren sincronizados con la señal de reloj.

La mayoría de los circuitos integrados Flip-Flops tienen también entradas asíncronas. Estas son entradas que pueden variar el estado del Flip-Flop independientemente del reloj. Generalmente, los fabricantes las denominan de inicialización, preset (PRE) y borrado, clear, (CLR), o de activación directa (SD, direct SET) y desactivación directa (RD, direct RESET). Un nivel activo en la entrada de inicialización del Flip-Flop (preset) pone a SET el dispositivo, y un nivel activo en la entrada de borrado (clear) lo pone en estado RESET.

Las entradas asíncronas de un biestable actúan al margen de las síncronas y prevalecen sobre ellas. Son muy útiles para iniciar o reiniciar el sistema con un estado inicial determinado.

# Flip-Flops con "Clear" y "Preset"

| PRE' | CLR' | D | q | q <sup>n+1</sup> |                     |
|------|------|---|---|------------------|---------------------|
| 0    | 1    | X | X | 1                | Constante inmediata |
| 1    | 0    | X | X | 0                |                     |
| 0    | 0    | X | X | -                | Invalido            |
| 1    | 1    | 0 | 0 | 0                | Normal              |
| 1    | 1    | 0 | 1 | 0                |                     |
| 1    | 1    | 1 | 0 | 1                |                     |
| 1    | 1    | 1 | 1 | 1                |                     |



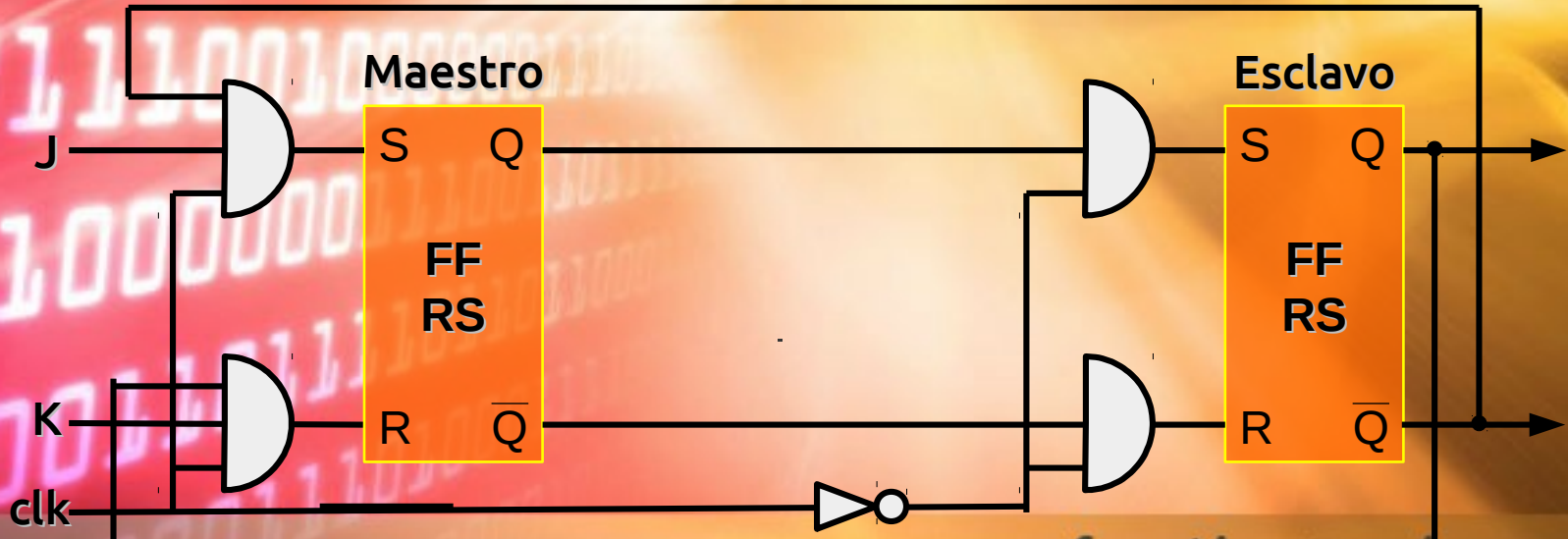
# Flip-Flop Maestro-Esclavo

Un Flip-Flop maestro-esclavo se construye con dos Flip-Flops, uno sirve de maestro y otro de esclavo.

Durante la subida del pulso de reloj se habilita el maestro y se deshabilita el esclavo. La información de entrada es transmitida hacia el Flip-Flop maestro. Cuando el pulso baja nuevamente a cero se deshabilita el maestro lo cual evita que lo afecten las entradas externas y se habilita el esclavo. Entonces el esclavo pasa al mismo estado del maestro. El comportamiento del Flip-Flop maestro-esclavo que acaba de describirse hace que los cambios de estado coincidan con la transición del flanco negativo del pulso.

Este tipo de biestables controlados por flanco se diseñan a partir de dos biestables sin entradas de control, conectados en cascada, más cierta lógica combinacional. Cuando el Clock está a 1, la información de entrada pasa al biestable maestro, mientras que el esclavo permanece cerrado, con lo que la salida no sufre variación. Cuando el Clock baja a 0 (flanco de bajada), la información del maestro pasa al esclavo y se cierra el maestro, con lo cual, los datos que en este instante estén en la entrada no pueden progresar, mientras que los datos que capturó el maestro y que ahora se transfieren al esclavo, son los que aparecen a la salida. Cuando el Clock vuelva a pasar a 1, se cerrará el esclavo con la información transferida anteriormente del maestro y que será la que esté presente en la salida, y éste se volverá a abrir. Luego la transferencia completa de la información, desde la entrada a la salida, sólo tendrá lugar durante los flancos de bajada de la señal del Clock.

# Flip-Flop Maestro-Esclavo



# Bibliografía y Licencia

- ◆ Acha, Santiago, Castro, Manuel, Rioseras, Miguel, "Electrónica Digital Introducción a la Lógica Digital" 2da Ed. (Ra-Ma 2010)
- ◆ Floyd, Thomas, "Fundamentos de sistemas digitales" 9na Ed. (Pearson 2006)
- ◆ Gonzalez Gomez, Juan, "Circuitos y Sistemas Digitales" (Madrid 2002)
- ◆ Este documento se encuentra bajo Licencia Creative Commons 2.5 Argentina (BY-NC-SA), por la cual se permite su exhibición, distribución, copia y posibilita hacer obras derivadas a partir de la misma, siempre y cuando se cite la autoría del Prof. Matías E. García y sólo podrá distribuir la obra derivada resultante bajo una licencia idéntica a ésta.
- ◆ Autor:

***Matías E. García***

Prof. & Tec. en Informática Aplicada  
[www.profmatiasgarcia.com.ar](http://www.profmatiasgarcia.com.ar)  
[info@profmatiasgarcia.com.ar](mailto:info@profmatiasgarcia.com.ar)

 **creative  
commons**



[www.profmatiasgarcia.com.ar](http://www.profmatiasgarcia.com.ar)