



Titulación: Ingeniería Informática
Asignatura: Fundamentos de Computadores

Bloque 3: Sistemas secuenciales
Tema 8: Elementos de memoria

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



ÍNDICE

- Bibliografía
- Introducción
- Biestables asíncronos
- Biestables síncronos por nivel
- Biestables síncronos por flanco
 - *Flip-flop SR*
 - *Flip-flop JK*
 - *Flip-flop D*
 - *Flip-flop T*
- Síntesis de máquinas de estados con *flip-flops*
- Análisis de máquinas de estados con *flip-flops*
- Descripción de máquinas de estados en VHDL

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



BIBLIOGRAFÍA

- Román Hermida, Ana M^o del Corral, Enric Pastor, Fermín Sánchez
“Fundamentos de Computadores” , cap. 5
Editorial Síntesis
- Thomas L. Floyd
“Fundamentos de Sistemas Digitales”, cap. 8
Editorial Prentice Hall
- Daniel D. Gajski
“Principios de Diseño Digital”, cap. 6
Editorial Prentice Hall
- M. Morris Mano
“Diseño Digital”, cap. 6

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



INTRODUCCIÓN

- Los elementos de memoria básicos se llaman biestables, por que pueden estar en dos posibles estados: con la salida a '0' o a '1'.
- Según las entradas de datos de que disponen podemos establecer varios tipos de biestables:
 - S-R: entradas de puesta a 1 (*S, set*) y puesta a 0 (*R, reset*)
 - J-K: entradas de puesta a 1 (*J, set*) y puesta a 0 (*K, reset*)
 - D: entrada de datos (*D*)
 - T: entrada de inversión o basculamiento (*toggle*)
- Los biestables pueden ser asíncronos o síncronos:
 - Asíncronos (*latch*): cuando cambia cualquiera de las entradas puede cambiar la salida.
 - Síncronos: la salida puede cambiar sólo cuando la señal de sincronismo lo

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



INTRODUCCIÓN

- Además, dentro de los biestables síncronos se puede diferenciar entre:
 - Síncronos por nivel (*gated latch*): la salida puede cambiar cuando la señal de sincronismo está en un determinado nivel (alto o bajo).
 - Síncronos por flanco (*low/high edge-triggered flip-flop*): la salida puede cambiar cuando la señal de sincronismo pasa de un nivel a otro.
 - Cuando pasa de nivel bajo a alto => síncrono por flanco de subida.
 - Cuando pasa de nivel alto a bajo => síncrono por flanco de bajada.
- Los más utilizados son los **biestables síncronos por flanco**, y reciben el nombre de ***flip-flops***.
- Normalmente los biestables síncronos cuentan con entradas asíncronas que se utilizan para forzar un valor determinado en los mismos al margen del reloj (prevalecen sobre él).
 - Puesta a 0 asíncrona (*clear, reset*)

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



INTRODUCCIÓN

- Tabla de excitación de un biestable:
 - Muestra las entradas que hay que introducir en un biestable para gobernar sus transiciones entre estados.

Estado actual	Estado siguiente	Entradas			
		X0	X1	...	Xn
S _i	S _j	V0	V1	...	Vn
.	.			.	
.	.			.	
.	.			.	

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



ÍNDICE

- Bibliografía
- Introducción
- **Biestables asíncronos**
- Biestables síncronos por nivel
- Biestables síncronos por flanco
 - *Flip-flop SR*
 - *Flip-flop JK*
 - *Flip-flop D*
 - *Flip-flop T*
- Síntesis de máquinas de estados con *flip-flops*
- Análisis de máquinas de estados con *flip-flops*
- Descripción de máquinas de estados en VHDL

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

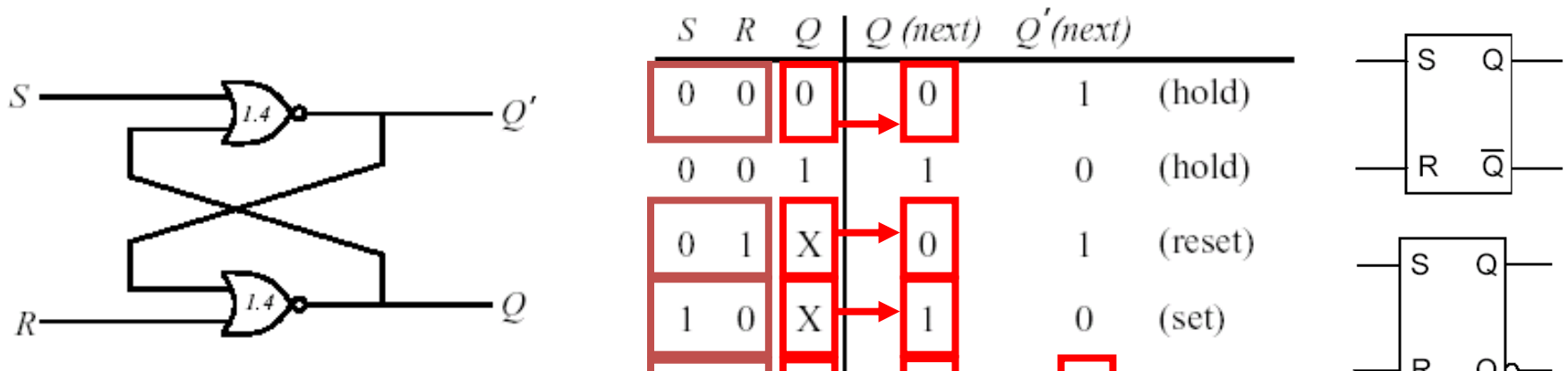
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



BIESTABLES ASÍNCRONOS

- El elemento de memoria más sencillo es el *latch* SR que tiene dos puertas NOR o NAND de dos entradas conectadas entre sí.
 - Es un biestable con un estado de *Set* (puesta a 1) y otro de *Reset* (puesta a 0).

Latch SR NOR Asíncrono:



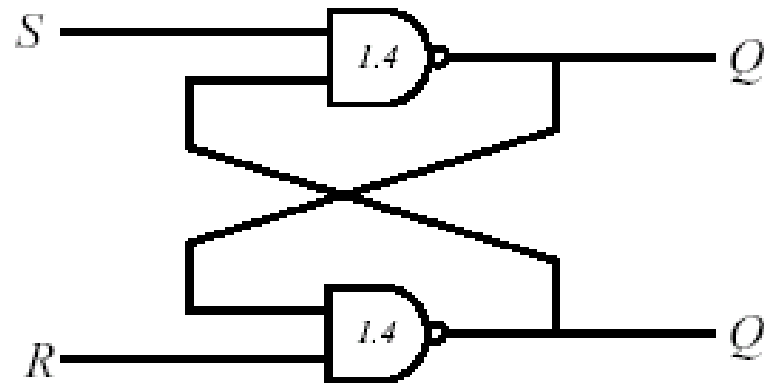
CLASES PARTICULARES, TUTORIAS TECNICAS ONLINE
 LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
 CALL OR WHATSAPP:689 45 44 70



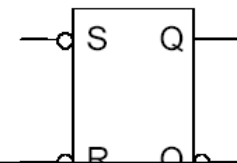
BIESTABLES ASÍNCRONOS

- **Latch SR NAND Asíncrono:**



S	R	Q	$Q (next)$	$Q' (next)$	
0	0	X	1	1	(?)
0	1	X	1	0	(set)
1	0	X	0	1	(reset)
1	1	0	0	1	(hold)

El *latch* SR NAND opera con señales activas bajas



Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



ÍNDICE

- Bibliografía
- Introducción
- Biestables asíncronos
- **Biestables síncronos por nivel**
- Biestables síncronos por flanco
 - *Flip-flop SR*
 - *Flip-flop JK*
 - *Flip-flop D*
 - *Flip-flop T*
- Síntesis de máquinas de estados con *flip-flops*
- Análisis de máquinas de estados con *flip-flops*
- Descripción de máquinas de estados en VHDL

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

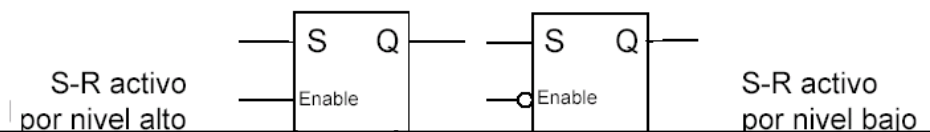
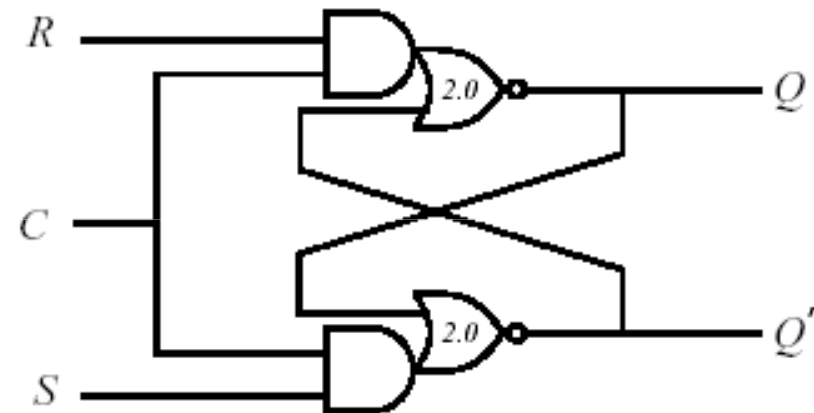
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



BIESTABLES SÍNCRONOS POR NIVEL

Latch SR NOR síncrono por nivel (*gated latch*): añadiendo unas puertas AND y una señal de sincronismo C, se puede convertir el *latch* SR NOR asíncrono en uno síncrono por nivel.

C	S	R	Q	Q (next)	
0	X	X	0	0	(inactive)
0	X	X	1	1	(inactive)
1	0	0	0	0	(hold)
1	0	0	1	1	(hold)
1	0	1	X	0	(reset)
1	1	0	X	1	(set)
1	1	1	X	NA	(?)

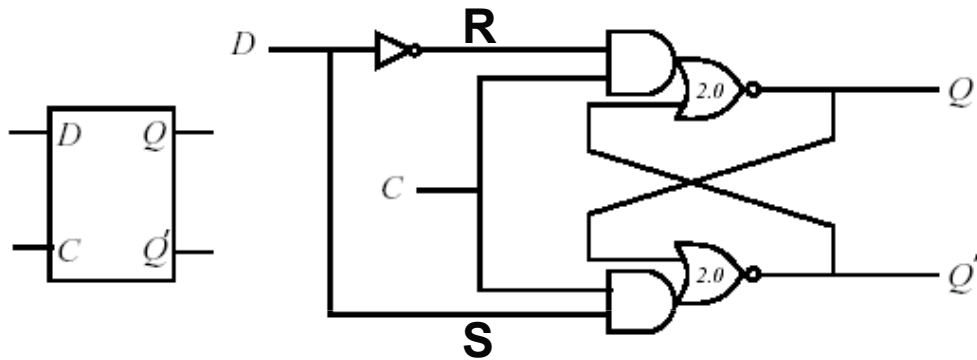


CLASES PARTICULARES, TUTORIAS TÉCNICAS ONLINE
 LLAMA O ENVÍA WHATSAPP: 689 45 44 70
 ...
 ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
 CALL OR WHATSAPP:689 45 44 70



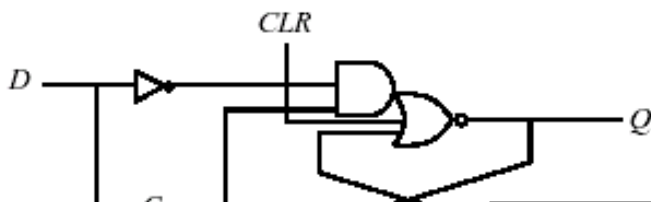
BIESTABLES SÍNCRONOS POR NIVEL

Latch D síncrono (gated latch): añadiendo un inversor para que la entrada R y la entrada S nunca tengan el mismo valor se elimina el estado prohibido, y se obtiene un **biestable D**.



C	D	Q	Q (next)
0	X	0	0
0	X	1	1
1	0	X	0
1	1	X	1

Latch D síncrono (gated latch) con Clear y Preset asíncronos: se pueden añadir entradas de *clear* y *preset* asíncronas conectándolas directamente al *latch* SR interno.



CLR	PRS	Q
0	0	ON
0	1	Preset (Q=1)



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



BIESTABLES SÍNCRONOS POR NIVEL

Comportamiento indeseado de los biestables disparados por nivel:

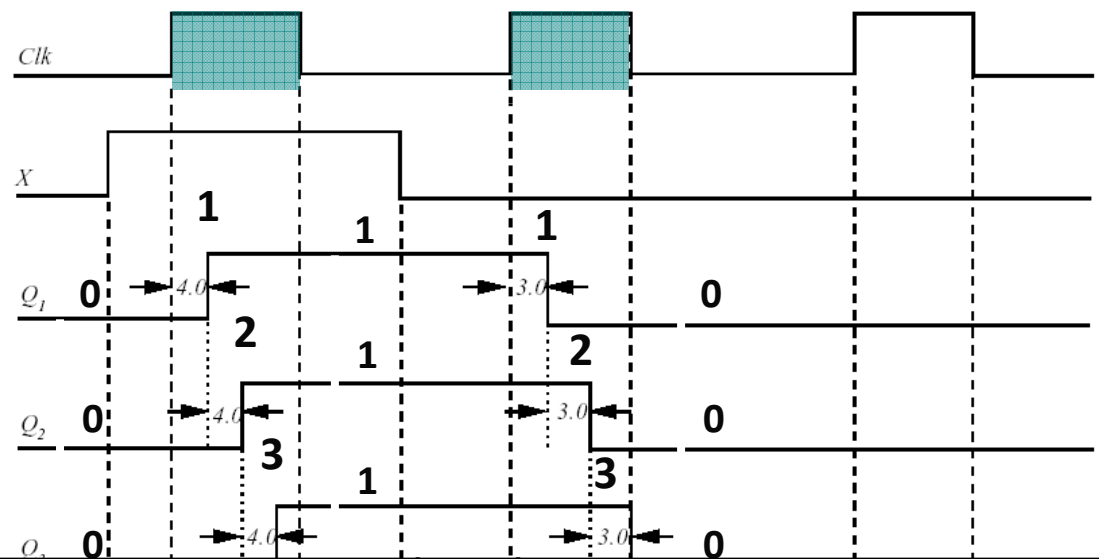
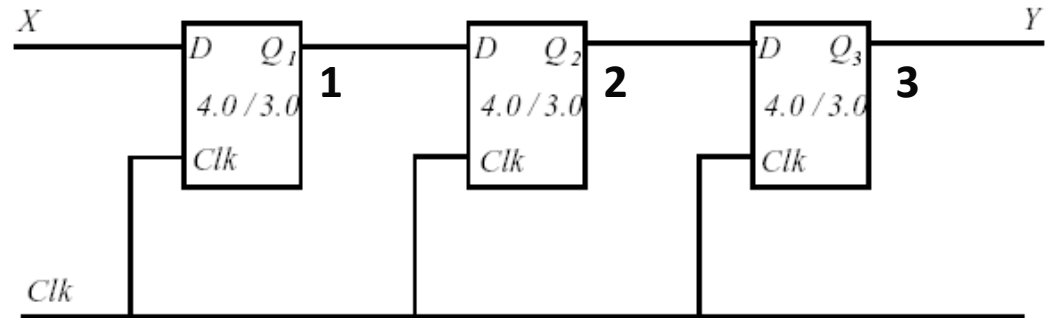
Desplazamiento de un valor alto $X=1$ por 3 latches.

Comportamiento deseado: 100, 010, 001(2)

Comportamiento obtenido: 111, 000, 000

Se debe a que los latches son transparentes y cuando $C=H$ los cambios en D se propagan

Solución:



CLASES PARTICULARES, TUTORIAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



ÍNDICE

- Bibliografía
- Introducción
- Biestables asíncronos
- Biestables síncronos por nivel
- **Biestables síncronos por flanco**
 - *Flip-flop SR*
 - *Flip-flop JK*
 - *Flip-flop D*
 - *Flip-flop T*
- Síntesis de máquinas de estados con *flip-flops*
- Análisis de máquinas de estados con *flip-flops*
- Descripción de máquinas de estados en VHDL

Cartagena99

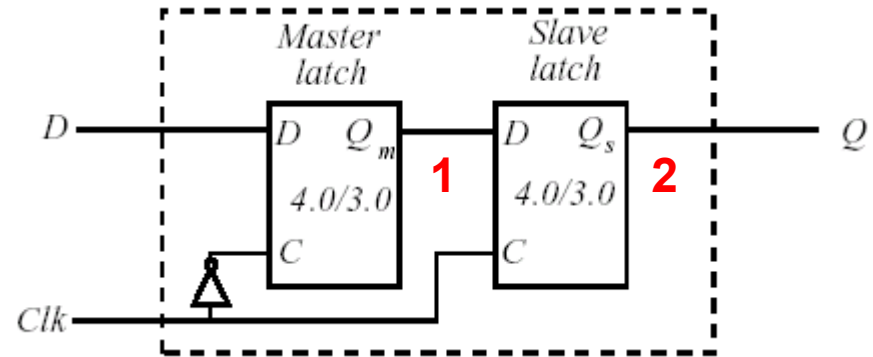
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



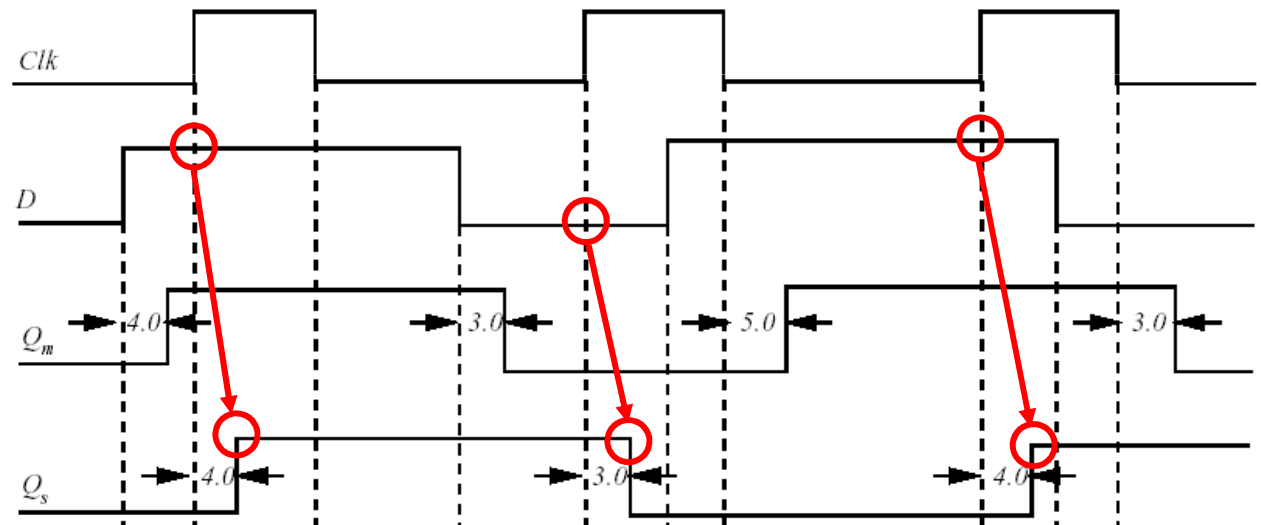
BIESTABLES SÍNCRONOS POR FLANCO

Flip-flop D master-slave: utilizando dos biestables síncronos por nivel se puede construir un biestable síncrono por flanco.



Con Clk=0 el primer D captura y el segundo D no cambia
Con Clk=1 al revés

Este modo de operación implica que **se captura D en el flanco de subida del reloj Clk**



Cartagena99

CLASES PARTICULARES, TUTORIAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

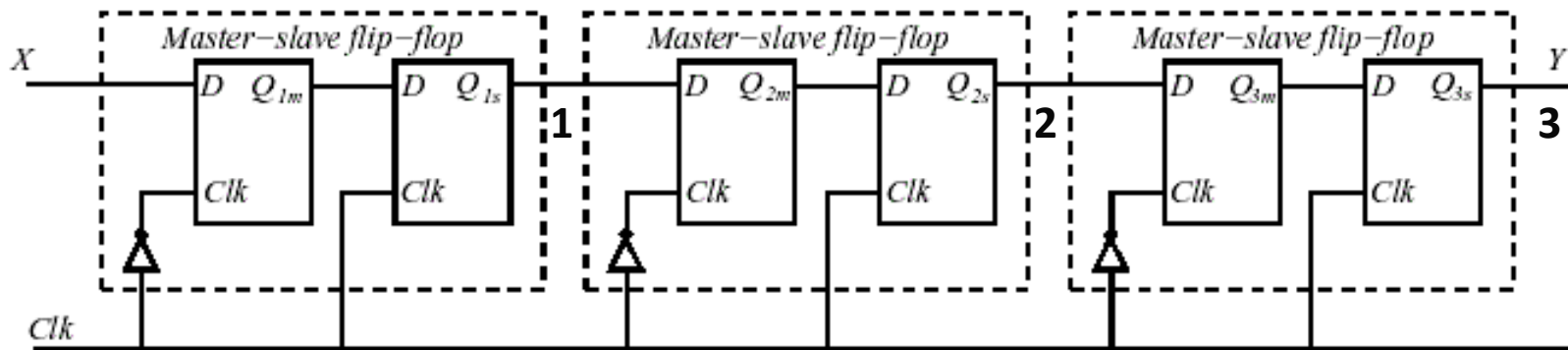
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



BIESTABLES SÍNCRONOS POR FLANCO

Comportamiento deseado de los biestables disparados por flanco:

Desplazamiento de un valor alto $X=1$ por 3 biestables *master-slave*.
Comportamiento deseado: 100, 010, 001(/2)



Cartagena99

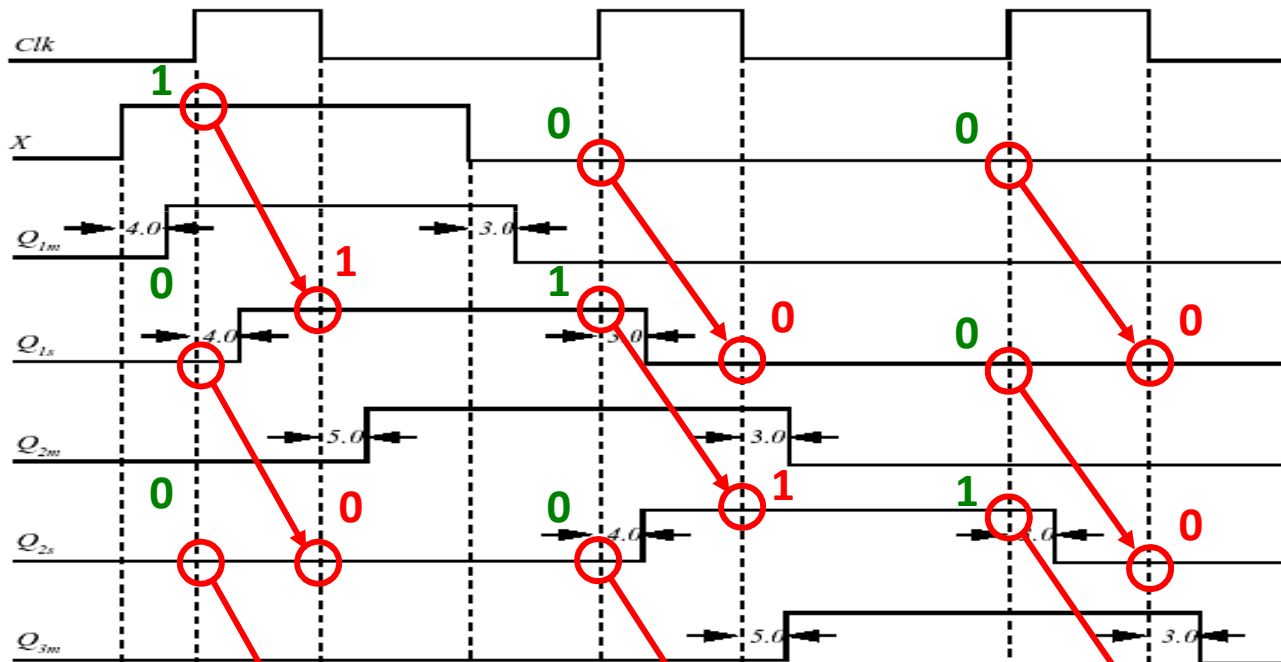
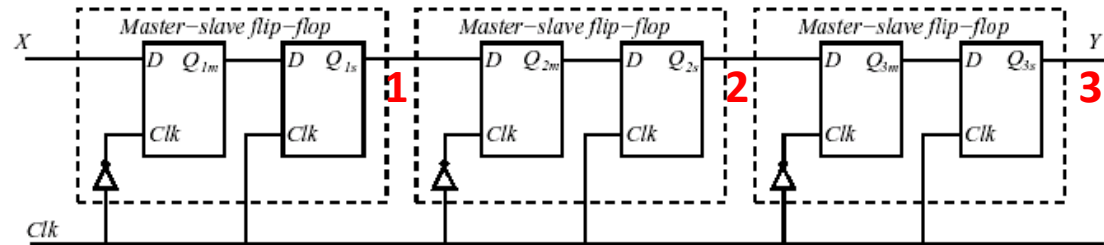
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

...

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP: 689 45 44 70



BIESTABLES SÍNCRONOS POR FLANCO



100, 010, 001

Desplazamiento

Cartagena99

CLASES PARTICULARES, TUTORIAS TECNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

...

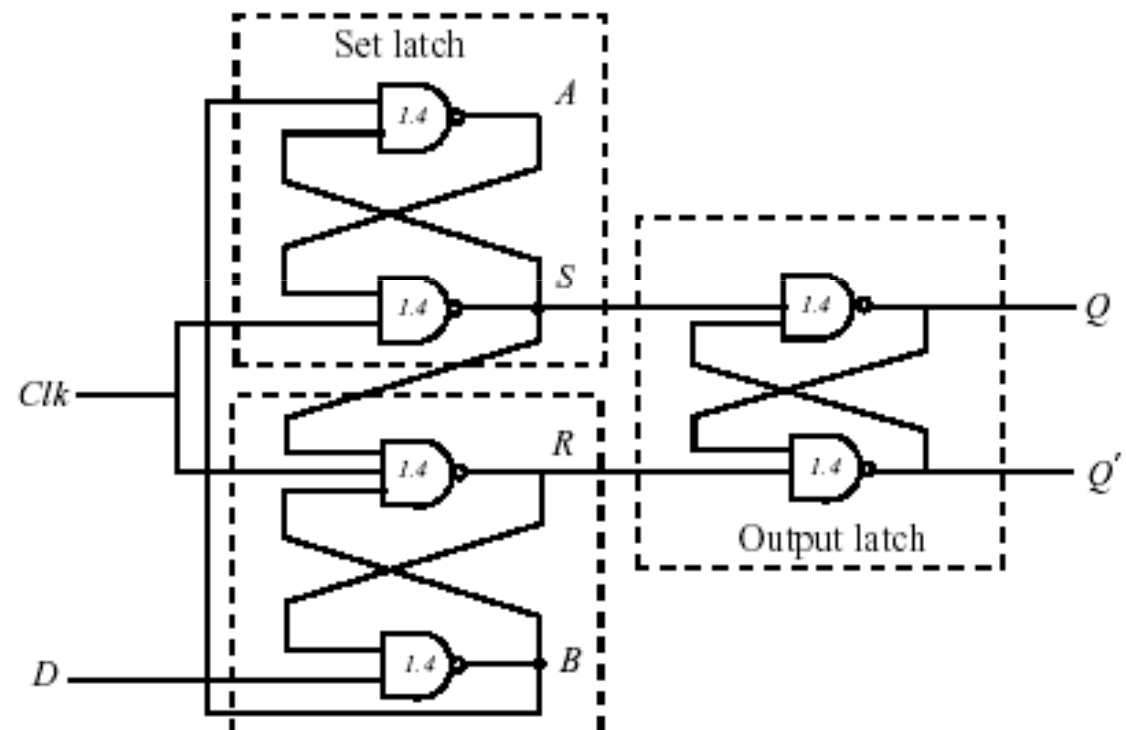
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



BIESTABLES SÍNCRONOS POR FLANCO

Flip-flop D disparado por flanco de subida del reloj:

Consiste en añadir al *latch* convencional dos *latches* que aseguren que en S y en R nunca se da la condición prohibida.



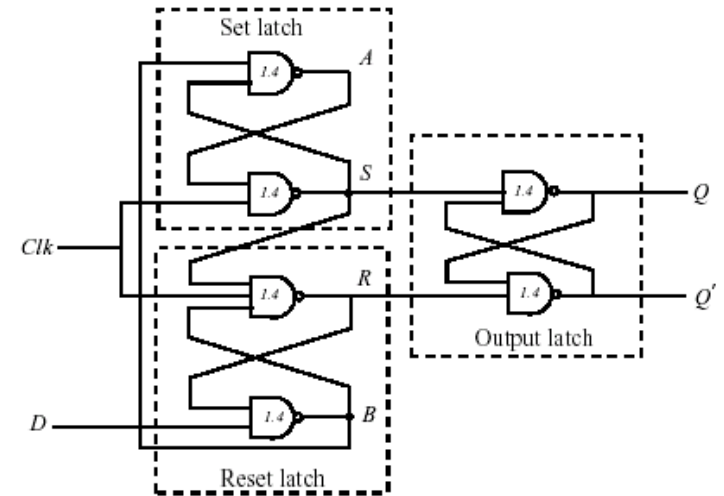
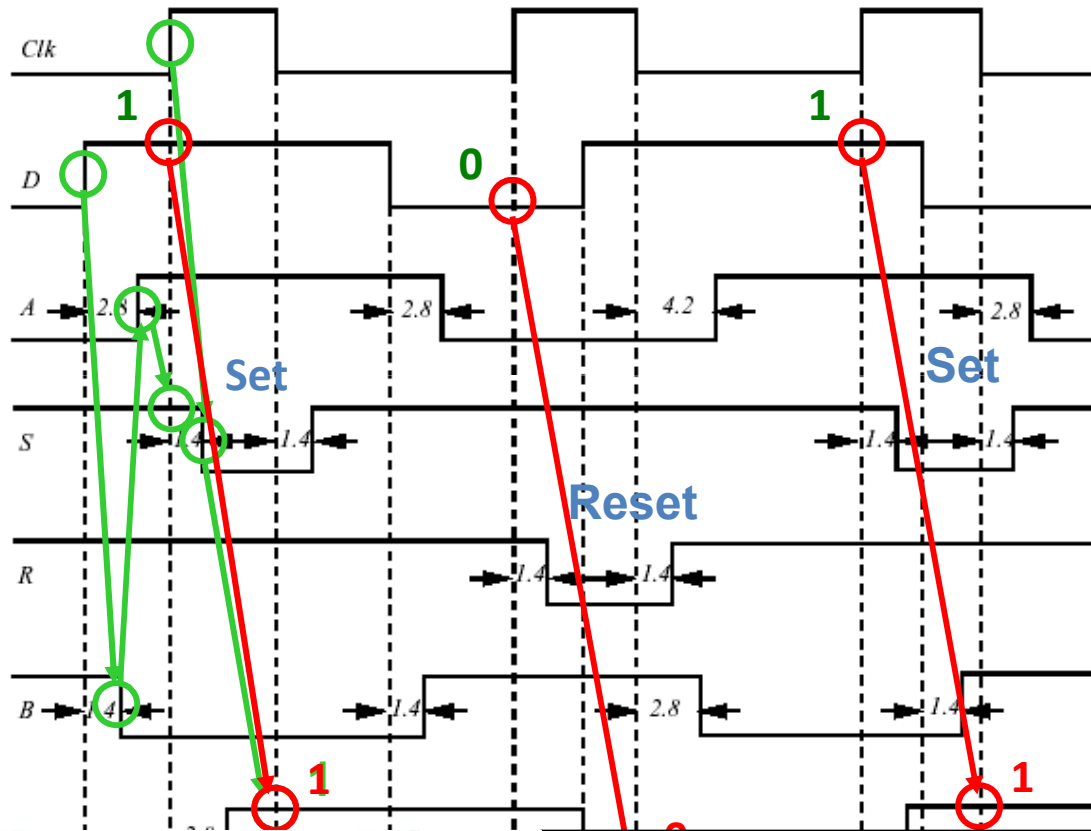
Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



BIESTABLES SÍNCRONOS POR FLANCO



Este modo de operación implica que se captura D en el flanco de subida del reloj Clk

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

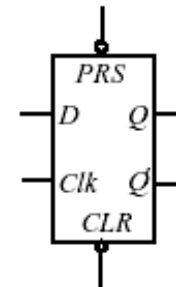
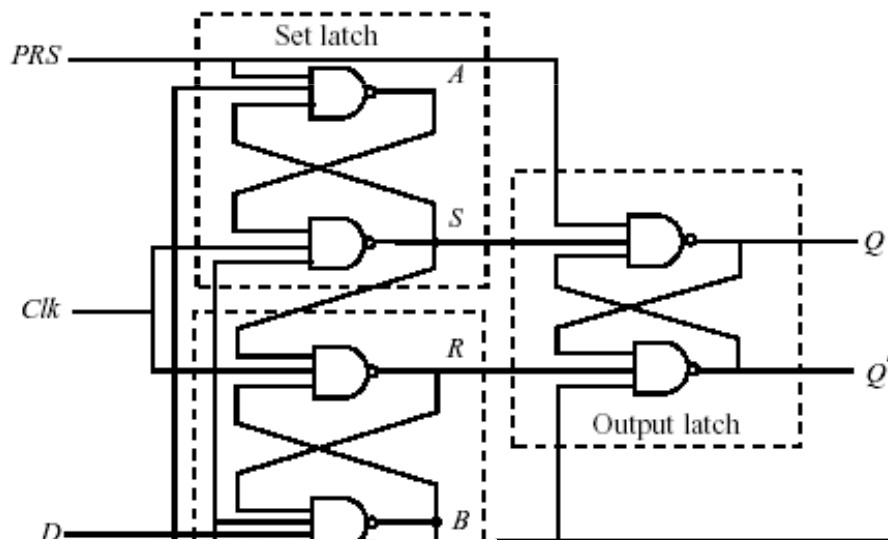
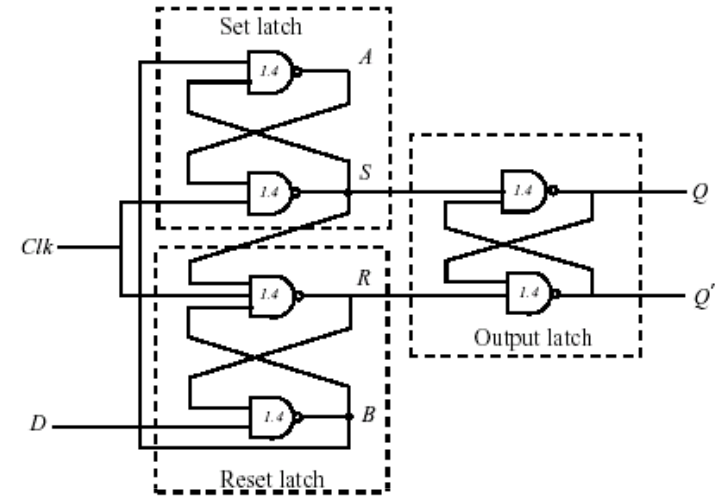
ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



BIESTABLES SÍNCRONOS POR FLANCO

Flip-flop D disparado por flanco de subida del reloj con *Clear* y *Preset*:

El *Flip-flop* D base + 2 entradas adicionales CLR y PRS en las NAND del *latch* de salida



CLR	PRS	
0	0	No permitido
0	1	Clear (Q=0)
1	0	Preset (Q=1)
1	1	ON

Cartagena99

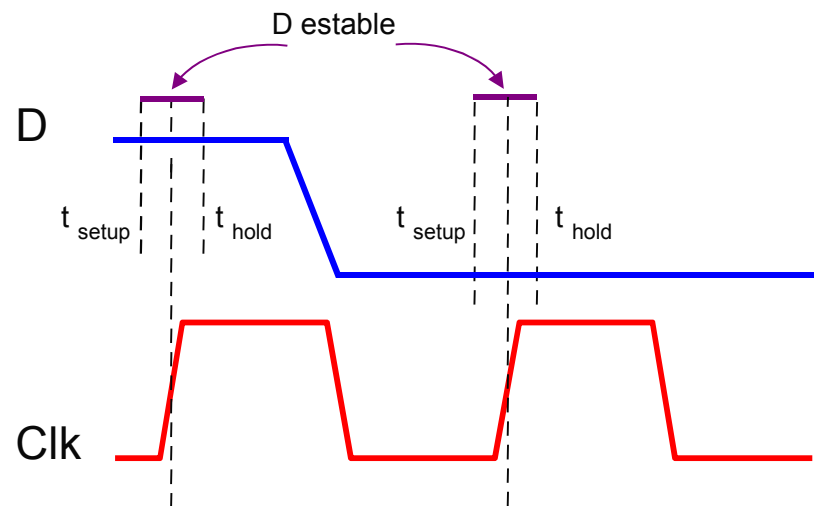
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



PARÁMETROS DE LOS BIESTABLES

- Para que un *flip-flop* funcione correctamente, hay que respetar dos tiempos:
 - Tiempo de preestabilización (*setup time*): tiempo previo al flanco activo en el que las entradas deben permanecer estables.
 - Tiempo de mantenimiento (*hold time*): tiempo posterior al flanco activo en el que las entradas deben permanecer estables.



Cartagena99

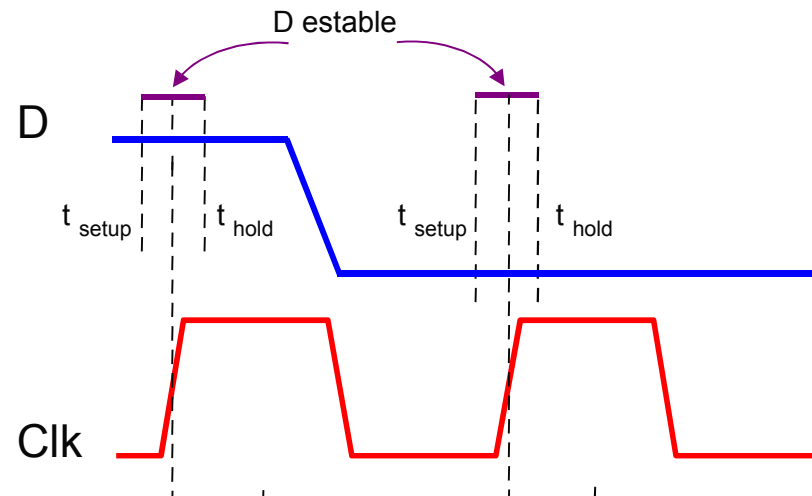
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



PARÁMETROS DE LOS BIESTABLES

- Otro tiempo importante es el retardo de propagación (*propagation delay*):
 - Es el tiempo transcurrido desde el flanco hasta que la salida del biestable presenta el nuevo estado.
 - El tiempo de cambio de valor alto a bajo (t_{pHL}) y el de bajo a alto (t_{pLH}) no tienen por qué ser iguales.
 - El retardo de propagación es siempre mayor que el tiempo de mantenimiento.



Cartagena99

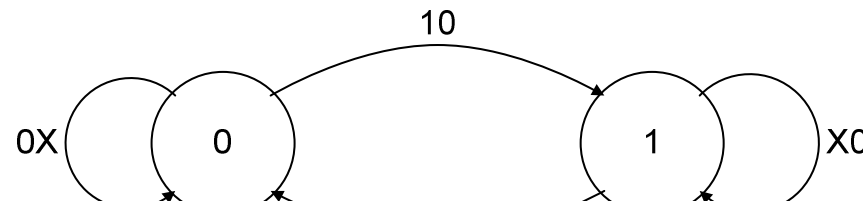
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



FLIP-FLOP SR

Flip-flop name	Flip-flop symbol	Characteristic table	Characteristic equation	Excitation table																																			
SR		<table border="1"> <thead> <tr> <th>S</th> <th>R</th> <th>Q (next)</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>Q</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>NA</td></tr> </tbody> </table>	S	R	Q (next)	0	0	Q	0	1	0	1	0	1	1	1	NA	$Q (next) = S + R'Q$ $SR = 0$	<table border="1"> <thead> <tr> <th>Q</th> <th>Q(next)</th> <th>S</th> <th>R</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td><td>X</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>X</td><td>0</td></tr> </tbody> </table>	Q	Q(next)	S	R	0	0	0	X	0	1	1	0	1	0	0	1	1	1	X	0
S	R	Q (next)																																					
0	0	Q																																					
0	1	0																																					
1	0	1																																					
1	1	NA																																					
Q	Q(next)	S	R																																				
0	0	0	X																																				
0	1	1	0																																				
1	0	0	1																																				
1	1	X	0																																				



CLASES PARTICULARES, TUTORIAS TÉCNICAS ONLINE
 LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
 CALL OR WHATSAPP:689 45 44 70



FLIP-FLOP SR: MODELO VHDL

```
library ieee;
use ieee.std_logic_1164.all;

entity flipflopSR is
  port (clk, rst, S, R: in std_logic;
        Q: out std_logic);
end flipflopSR;

architecture funcional of flipflopSR is
begin
  process (clk, rst)
  begin
    if rst = '1' then
      Q <= '0';
    elsif rising_edge (clk) then
      if S = '1' then
        Q <= '1';
      elsif R = '1' then
        Q <= '0';
      end if;
    end if;
  end process;
end architecture;
```

Cartagena99

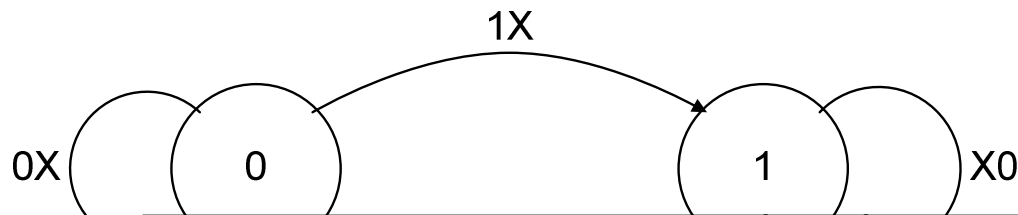
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



FLIP-FLOP JK

Flip-flop name	Flip-flop symbol	Characteristic table	Characteristic equation	Excitation table																																			
JK		<table border="1"> <thead> <tr> <th>J</th> <th>K</th> <th>Q (next)</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>Q</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>Q'</td></tr> </tbody> </table>	J	K	Q (next)	0	0	Q	0	1	0	1	0	1	1	1	Q'	$Q (next) = JQ' + K'Q$	<table border="1"> <thead> <tr> <th>Q</th> <th>Q(next)</th> <th>J</th> <th>K</th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td><td>X</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>X</td></tr> <tr><td>1</td><td>0</td><td>X</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>X</td><td>0</td></tr> </tbody> </table>	Q	Q(next)	J	K	0	0	0	X	0	1	1	X	1	0	X	1	1	1	X	0
J	K	Q (next)																																					
0	0	Q																																					
0	1	0																																					
1	0	1																																					
1	1	Q'																																					
Q	Q(next)	J	K																																				
0	0	0	X																																				
0	1	1	X																																				
1	0	X	1																																				
1	1	X	0																																				



Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



FLIP-FLOP JK: MODELO VHDL

```
library ieee;
use ieee.std_logic_1164.all;

entity flipflopJK is
  port (clk, rst, J, K: in std_logic;
        Q: out std_logic);
end flipflopJK;
```

```
architecture funcional of flipflopJK is
  signal JK: std_logic_vector(1 downto 0);
  signal Q_aux: std_logic;
begin
  JK <= J&K;
  process (clk, rst)
  begin
    if rst = '1' then
      Q_aux <= '0';
    elsif rising_edge(clk) then
      case JK is
        when "00" => Q_aux <= Q_aux;
        when "01" => Q_aux <= '0';
        when "10" => Q_aux <= '1';
        when others => Q_aux <= not Q_aux;
      end case;
    end if;
  end process;
```

Cartagena99

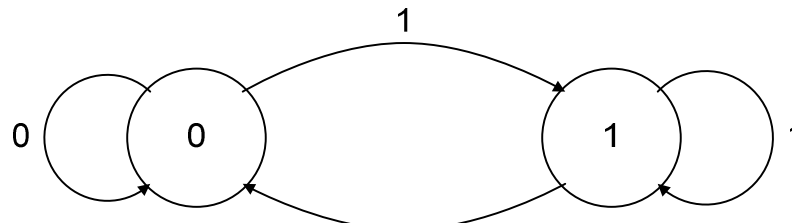
CLASES PARTICULARES, TUTORIAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



FLIP-FLOP D

Flip-flop name	Flip-flop symbol	Characteristic table	Characteristic equation	Excitation table																					
D		<table border="1"> <thead> <tr> <th>D</th> <th>Q (next)</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> </tr> </tbody> </table>	D	Q (next)	0	0	1	1	$Q (next) = D$	<table border="1"> <thead> <tr> <th>Q</th> <th>Q(next)</th> <th>D</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table>	Q	Q(next)	D	0	0	0	0	1	1	1	0	0	1	1	1
D	Q (next)																								
0	0																								
1	1																								
Q	Q(next)	D																							
0	0	0																							
0	1	1																							
1	0	0																							
1	1	1																							



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
 LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
 CALL OR WHATSAPP:689 45 44 70



FLIP-FLOP D: MODELO VHDL

```
library ieee;
use ieee.std_logic_1164.all;

entity flipflopD is
  port(clk, rst, D: in std_logic;
        Q: out std_logic);
end flipflopD;

architecture funcional of flipflopD is
begin
  process(clk,rst)
  begin
    if rst = '1' then
      Q <= '0';
    elsif rising_edge(clk) then
      Q <= D;
    end if;
  end process;
end funcional;
```

Cartagena99

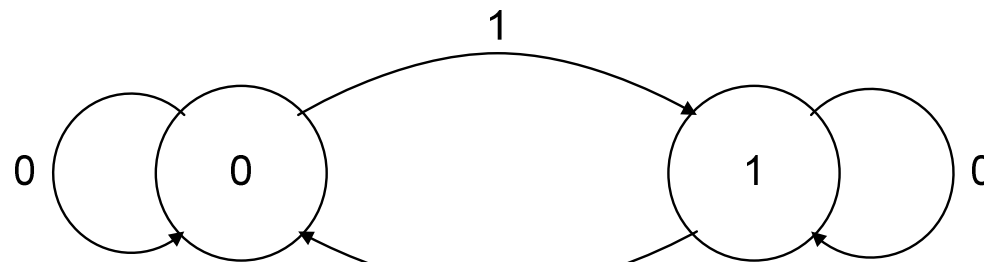
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



FLIP-FLOP T

Flip-flop name	Flip-flop symbol	Characteristic table	Characteristic equation	Excitation table																					
T		<table border="1"> <thead> <tr> <th>T</th> <th>Q (next)</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Q</td> </tr> <tr> <td>1</td> <td>Q'</td> </tr> </tbody> </table>	T	Q (next)	0	Q	1	Q'	$Q (next) = TQ' + T'Q$	<table border="1"> <thead> <tr> <th>Q</th> <th>Q(next)</th> <th>T</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	Q	Q(next)	T	0	0	0	0	1	1	1	0	1	1	1	0
T	Q (next)																								
0	Q																								
1	Q'																								
Q	Q(next)	T																							
0	0	0																							
0	1	1																							
1	0	1																							
1	1	0																							



Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



FLIP-FLOP T: MODELO VHDL

```
library ieee;
use ieee.std_logic_1164.all;

entity flipflopT is
  port (clk, rst, T: in std_logic;
        Q: out std_logic);
end flipflopT;

architecture funcional of flipflopT is
  signal Q_aux: std_logic;
begin
  process (clk, rst)
  begin
    if rst = '1' then
      Q_aux <= '0';
    elsif rising_edge (clk) then
      if T = '1' then
        Q_aux <= not Q_aux;
      end if;
    end if;
  end if;
end if;
```

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP: 689 45 44 70



ÍNDICE

- Bibliografía
- Introducción
- Biestables asíncronos
- Biestables síncronos por nivel
- Biestables síncronos por flanco
 - *Flip-flop SR*
 - *Flip-flop JK*
 - *Flip-flop D*
 - *Flip-flop T*
- **Síntesis de máquinas de estados con *flip-flops***
- Análisis de máquinas de estados con *flip-flops*
- Descripción de máquinas de estados en VHDL

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



SÍNTESIS DE FSM CON *FLIP-FLOPS*

- Hemos visto ya la síntesis de máquinas secuenciales utilizando *flip-flops* de tipo D.
- Revisaremos la síntesis, pero empleando ahora *flip-flops* de otras clases.
- La dificultad estriba en la síntesis de la función de transición.
 - Con *flip-flops* D basta con conocer el estado próximo, y suministrar a la entrada D el valor del estado buscado.
 - Con otros *flip-flops* hay que conocer el estado actual y el próximo, y en función de ambos establecer el valor de las entradas.
 - En los *flip-flops* SR y JK será preciso calcular dos funciones por cada variable de estado.
- Usaremos el mismo ejemplo que en el tema anterior.
 - Materializaremos únicamente máquinas de Mealy.

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



SÍNTESIS DE FSM CON *FLIP-FLOPS*

Diagrama de estados

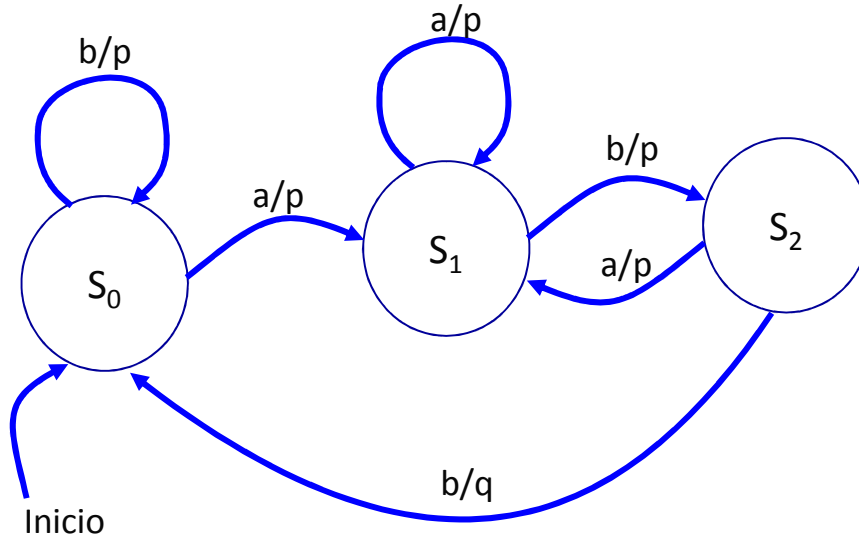


Tabla de estados y salidas

Estado actual	Entrada actual	
	A	b
S ₀	S ₁ /p	S ₀ /p
S ₁	S ₁ /p	S ₂ /p
S ₂	S ₁ /p	S ₀ /q
Estado siguiente / Salida		

Codificación binaria

Entrada		Salida		Estado		
x(t)	x ₀	z(t)	z ₀	s(t)	Q ₁	Q ₀

Tabla de estados y salidas

S(t)		X ₀	
Q ₁	Q ₀	0	1



CLASES PARTICULARES, TUTORIAS TÉCNICAS ONLINE
 LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
 CALL OR WHATSAPP:689 45 44 70



SÍNTESIS DE FSM CON *FLIP-FLOPS*: JK

- Si elegimos biestables JK, tenemos que crear una nueva tabla de excitación y salida.

S(t)		X ₀		S(t), X			S(t+1), Z			Entradas de excitación			
Q ₁	Q ₀	0	1	Q ₁	Q ₀	X	Q' ₁	Q' ₀	Z	J ₁	K ₁	J ₀	K ₀
0	0	01/0	00/0	0	0	0	0	1	0	0	X	1	X
0	1	01/0	10/0	0	0	1	0	0	0	0	X	0	X
1	0	01/0	00/1	0	1	1	1	0	0	1	X	X	1
1	1			1	0	0	0	1	0	X	1	1	X
				1	0	1	0	0	1	X	1	0	X
				1	1	0	X	X	X	X	X	X	X

Flip-flop name	Flip-flop symbol	Excitation table																
JK		<table border="1"> <thead> <tr> <th>Q</th> <th>Q(next)</th> <th>J</th> <th>K</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>X</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>X</td> </tr> <tr> <td>1</td> <td>0</td> <td>X</td> <td>1</td> </tr> </tbody> </table>	Q	Q(next)	J	K	0	0	0	X	0	1	1	X	1	0	X	1
Q	Q(next)	J	K															
0	0	0	X															
0	1	1	X															
1	0	X	1															

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
 LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
 CALL OR WHATSAPP:689 45 44 70





SÍNTESIS DE FSM CON *FLIP-FLOPS*: JK

- Es preciso materializar J_1, K_1, J_0, K_0 y Z .

S(t), X			S(t+1), Z			Entradas de excitación			
Q_1	Q_0	X	Q'_1	Q'_0	Z	J_1	K_1	J_0	K_0
0	0	0	0	1	0	0	X	1	X
0	0	1	0	0	0	0	X	0	X
0	1	0	0	1	0	0	X	X	0
0	1	1	1	0	0	1	X	X	1
1	0	0	0	1	0	X	1	1	X
1	0	1	0	0	1	X	1	0	X
1	1	0	X	X	X	X	X	X	X



Función de transición

$$J_1(Q_1, Q_0, X_0) = X_0 \cdot Q_0$$

$$K_1(Q_1, Q_0, X_0) = 1$$

$$J_0(Q_1, Q_0, X_0) = \overline{X_0}$$

$$K_0(Q_1, Q_0, X_0) = X_0$$

Función de salida

$$Z_0(Q_1, Q_0, X_0) = X_0 \cdot Q_1$$

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP: 689 45 44 70



SÍNTESIS DE FSM CON *FLIP-FLOPS*: SR

- Si elegimos biestables SR, la tabla de excitación y salida cambia.

S(t)		X ₀	
Q ₁	Q ₀	0	1
0	0	01/0	00/0
0	1	01/0	10/0
1	0	01/0	00/1



S(t), X			S(t+1), Z			Entradas de excitación			
Q ₁	Q ₀	X	Q' ₁	Q' ₀	Z	S ₁	R ₁	S ₀	R ₀
0	0	0	0	1	0	0	X	1	0
0	0	1	0	0	0	0	X	0	X
0	1	0	0	1	0	0	X	X	0
0	1	1	1	0	0	1	0	0	1
1	0	0	0	1	0	0	1	1	0
1	0	1	0	0	1	0	1	0	X
1	1	0	X	X	X	X	X	X	X

Flip-flop name	Flip-flop symbol	Excitation table			
		Q	Q(next)	S	R
SR		0	0	0	X
		0	1	1	0
		1	0	0	1
		1	1	X	0

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
 LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
 CALL OR WHATSAPP:689 45 44 70





ÍNDICE

- Bibliografía
- Introducción
- Biestables asíncronos
- Biestables síncronos por nivel
- Biestables síncronos por flanco
 - *Flip-flop SR*
 - *Flip-flop JK*
 - *Flip-flop D*
 - *Flip-flop T*
- Síntesis de máquinas de estados con *flip-flops*
- **Análisis de máquinas de estados con *flip-flops***
- Descripción de máquinas de estados en VHDL

Cartagena99

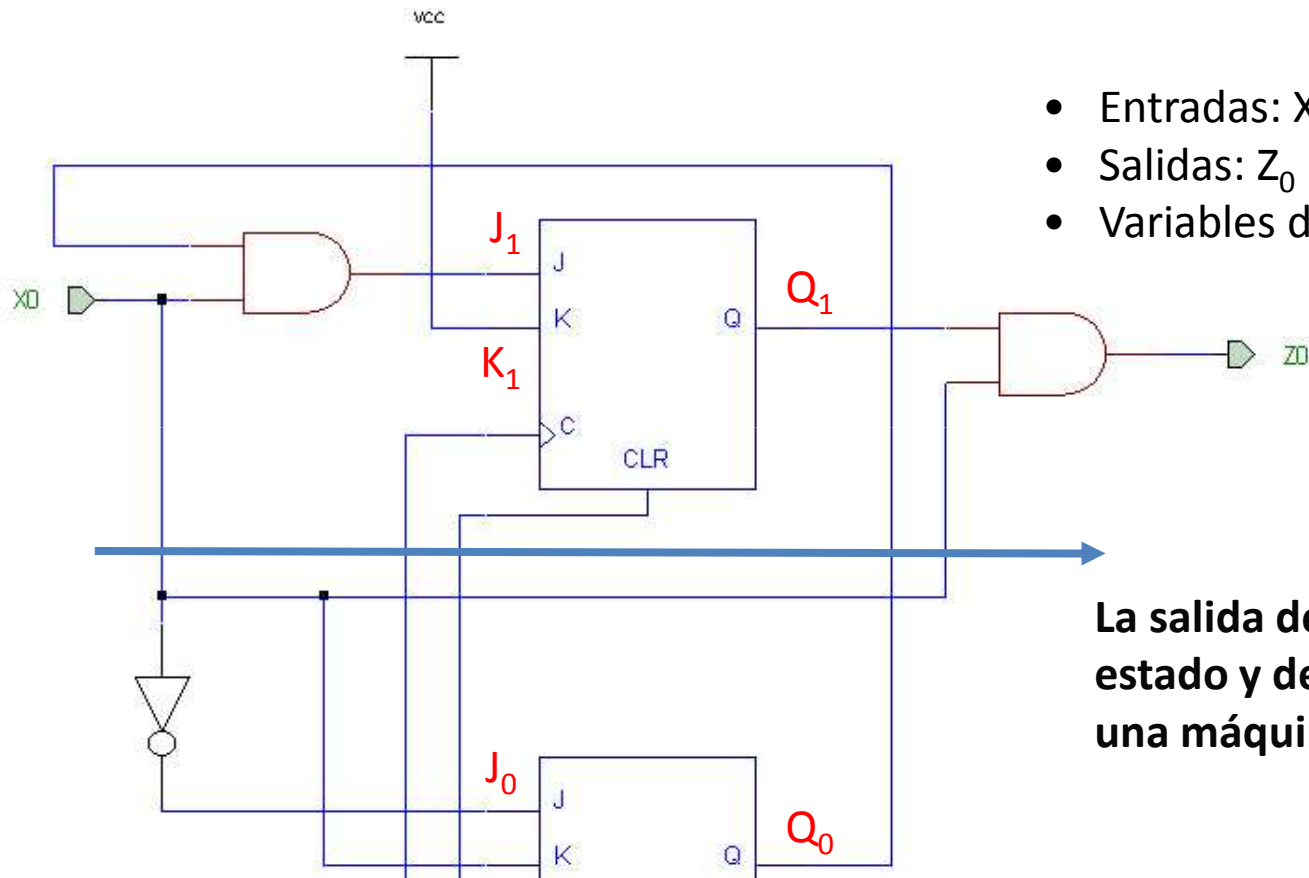
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



ANÁLISIS DE FSM CON *FLIP-FLOPS*

- Lo que cambia es la generación de la tabla de excitación y salida.



- Entradas: $X_0 \in \{0,1\}$
- Salidas: $Z_0 \in \{0,1\}$
- Variables de estado: 2 $\{Q_1, Q_0\}$

La salida depende del estado y de la entrada: es una máquina de Mealy.

Cartagena99

**CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70**

**ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70**



ANÁLISIS DE FSM: ECUACIONES Y TABLAS - MEALY

Función de transición

$$J_1(Q_1, Q_0, X_0) = X_0 \cdot Q_0$$

$$K_1(Q_1, Q_0, X_0) = 1$$

$$J_0(Q_1, Q_0, X_0) = \overline{X_0}$$

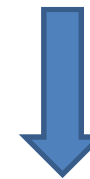
$$K_0(Q_1, Q_0, X_0) = X_0$$

Función de salida

$$Z_0(Q_1, Q_0, X_0) = X_0 \cdot Q_1$$



S(t), X			Entradas de excitación				S(t+1), Z		
Q ₁	Q ₀	X	J ₁	K ₁	J ₀	K ₀	Q' ₁	Q' ₀	Z
0	0	0	0	X	1	X	0	1	0
0	0	1	0	X	0	X	0	0	0
0	1	0	0	X	X	0	0	1	0
0	1	1	1	X	X	1	1	0	0
1	0	0	X	1	1	X	0	1	0
1	0	1	X	1	0	X	0	0	1
1	1	0	0	X	0	0	1	1	0
1	1	1	0	X	1	0	1	0	1



La salida depende del estado y de la entrada: es una máquina de Mealy.

S(t)		X ₀	
Q ₁	Q ₀	0	1
0	0	01/0	00/0
0	1	01/0	10/0
1	0	01/0	00/1



Cartagena99

CLASES PARTICULARES, TUTORIAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



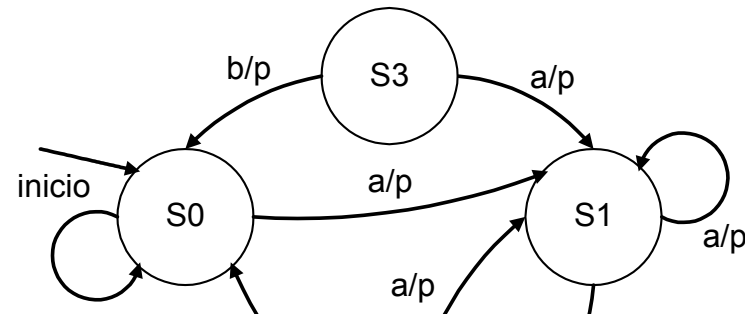
ANÁLISIS DE FSM: DECODIFICACIÓN Y REPRESENTACIÓN FORMAL - MEALY

- Asignamos nombres y valores a las entradas, los estados y la salida.

S(t)		X ₀		Entrada		Salida		Estado		
Q ₁	Q ₀	0	1	X ₀	X(t)	Z ₀	Z(t)	Q ₁	Q ₀	S(t)
0	0	01/0	00/0	0	a	0	p	0	0	S ₀
0	1	01/0	10/0	1	b	1	q	0	1	S ₁
1	0	01/0	00/1					1	0	S ₂
1	1	01/0	00/1					1	1	S ₃



Estado actual	Entrada actual	
	a	b
S ₀	S ₁ /p	S ₀ /p



CLASES PARTICULARES, TUTORIAS TÉCNICAS ONLINE
 LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
 CALL OR WHATSAPP:689 45 44 70



ÍNDICE

- Bibliografía
- Introducción
- Biestables asíncronos
- Biestables síncronos por nivel
- Biestables síncronos por flanco
 - *Flip-flop SR*
 - *Flip-flop JK*
 - *Flip-flop D*
 - *Flip-flop T*
- Síntesis de máquinas de estados con *flip-flops*
- Análisis de máquinas de estados con *flip-flops*
- Descripción de máquinas de estados en VHDL

Cartagena99

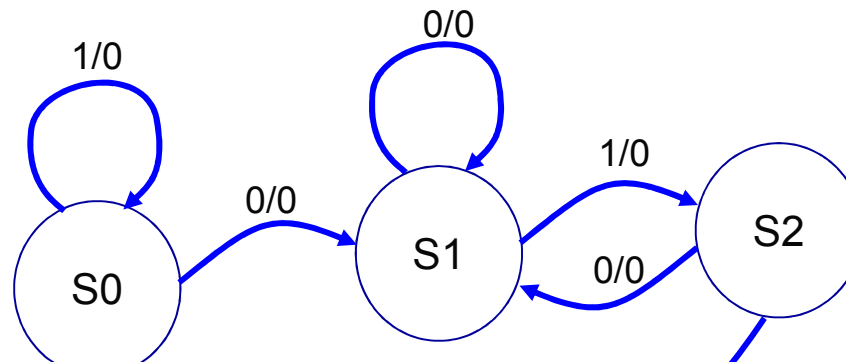
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



DESCRIPCIÓN DE MÁQUINAS DE ESTADOS EN VHDL

- Las máquinas de estados se pueden describir fácilmente en VHDL de forma funcional.
- Se describen utilizando 2 procesos, uno que modela el cálculo del estado siguiente y de la salida, y otro proceso que modela la actualización del estado.
- Se puede crear un tipo de datos enumerado para los estados, o se puede utilizar un vector de bits.
- Vamos a ver un ejemplo con la siguiente máquina de estados:



CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70

Cartagena99



DESCRIPCIÓN DE MÁQUINAS DE ESTADOS EN VHDL

- Ejemplo: entidad y proceso que modela el estado futuro y la salida.

```
library ieee;  
use ieee.std_logic_1164.all;  
  
entity fsm_funcional is  
    port(clk,rst,x: in std_logic;  
         z: out std_logic);  
end fsm_funcional;
```

```
architecture funcional of fsm_funcional is  
    type estados is (S0, S1, S2);  
    signal ns, ps: estados; --next state y present state  
begin  
    --proximo estado y salida  
    process(x,ps)  
    begin  
        case ps is  
            when S0 => if x = '0' then ns <= S1; z <= '0';  
                       else ns <= S0; z <= '0';  
                       end if;  
            when S1 => if x = '0' then ns <= S1; z <= '0';  
                       else ns <= S2; z <= '0';  
                       end if;
```

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



DESCRIPCIÓN DE MÁQUINAS DE ESTADOS EN VHDL

- Ejemplo (continuación): proceso que modela la actualización del estado.

```
--actualización del estado
  process (clk, rst)
  begin
    if rst = '1' then
      ps <= S0;
    elsif rising_edge(clk) then
      ps <= ns;
    end if;
  end process;

end funcional;
```

Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



DESCRIPCIÓN DE MÁQUINAS DE ESTADOS EN VHDL

- Ejemplo (continuación): test-bench.

```
library ieee;
use ieee.std_logic_1164.all;

entity test_fsm_funcional is
end test_fsm_funcional;

architecture test of test_fsm_funcional is
signal clk: std_logic := '0';
signal rst,x,z: std_logic;
begin
  clk <= not clk after 5 ns;
  rst <= '0', '1' after 10 ns, '0' after 20 ns;
  x <= '1', '0' after 30 ns, '1' after 60 ns, '0' after 70 ns, '1' after 80 ns;
  inst: entity work.fsm_funcional port map(clk,rst,x,z);
end test;
```

Cartagena99

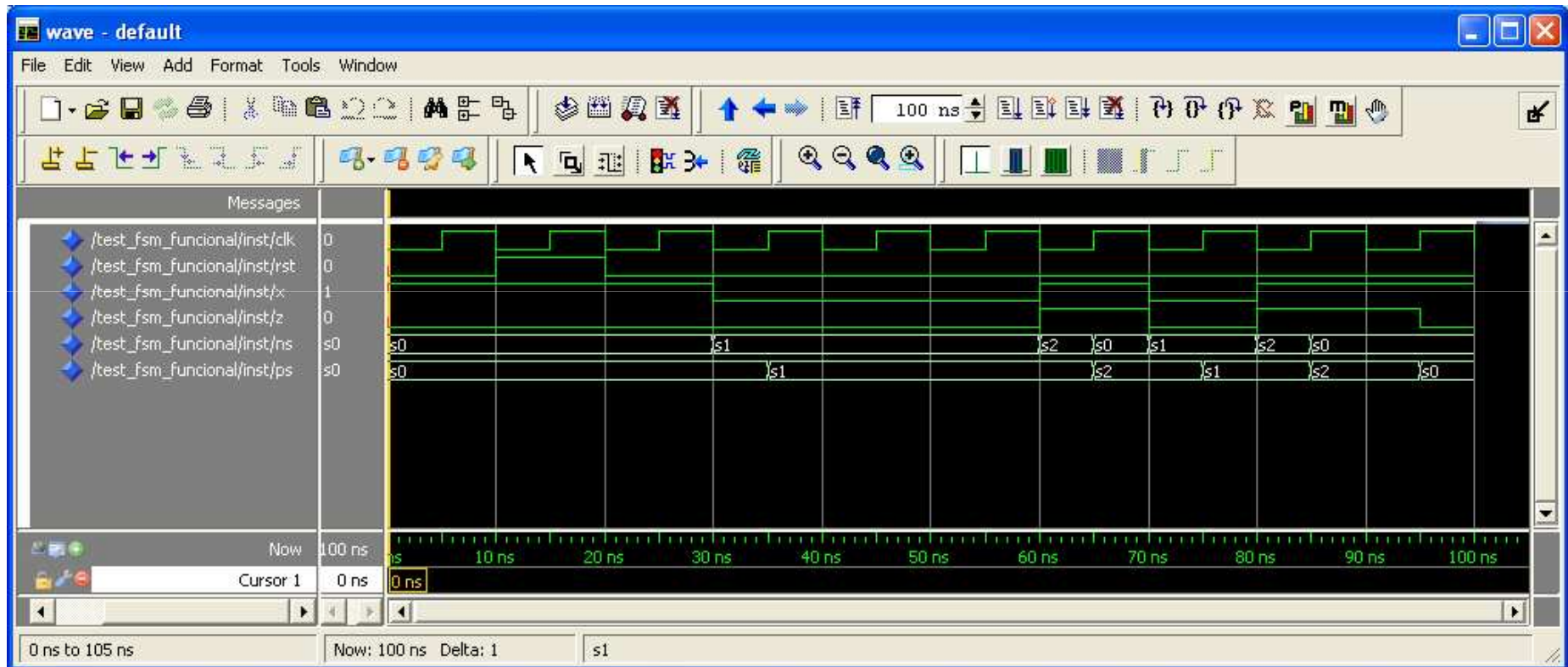
CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70



DESCRIPCIÓN DE MÁQUINAS DE ESTADOS EN VHDL

- Ejemplo (continuación): resultado de la simulación.



Cartagena99

CLASES PARTICULARES, TUTORÍAS TÉCNICAS ONLINE
LLAMA O ENVÍA WHATSAPP: 689 45 44 70

ONLINE PRIVATE LESSONS FOR SCIENCE STUDENTS
CALL OR WHATSAPP:689 45 44 70