

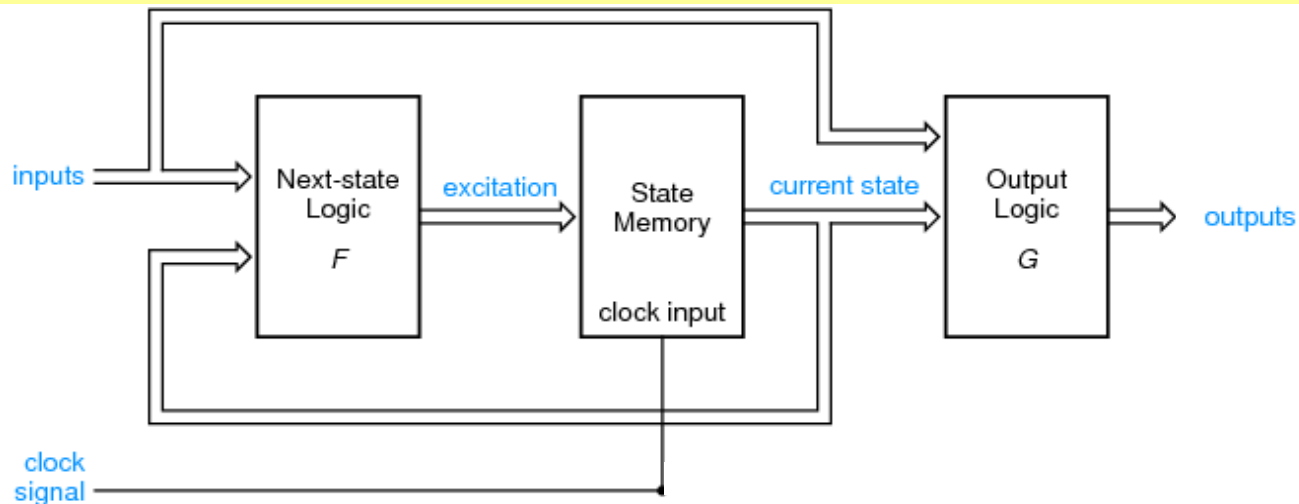
Sistemas secuenciales

Máquinas de estado síncronas
Análisis y diseño

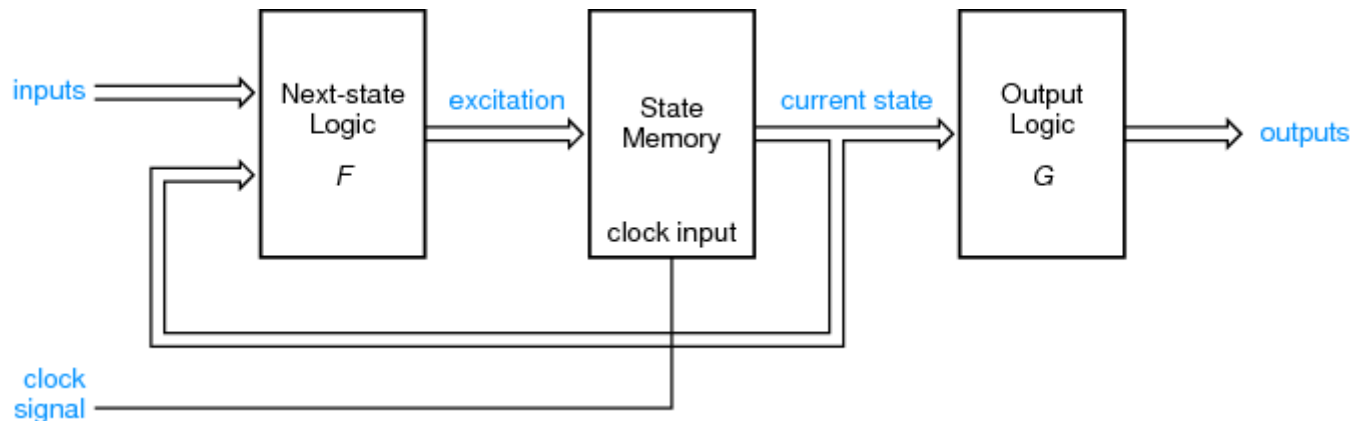
Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: Estructuras

Esquema de una máquina de estados (tipo Mealy). **Estado siguiente = $F(\text{estado actual}, \text{entrada})$**
Salida = $G(\text{estado actual}, \text{entrada})$

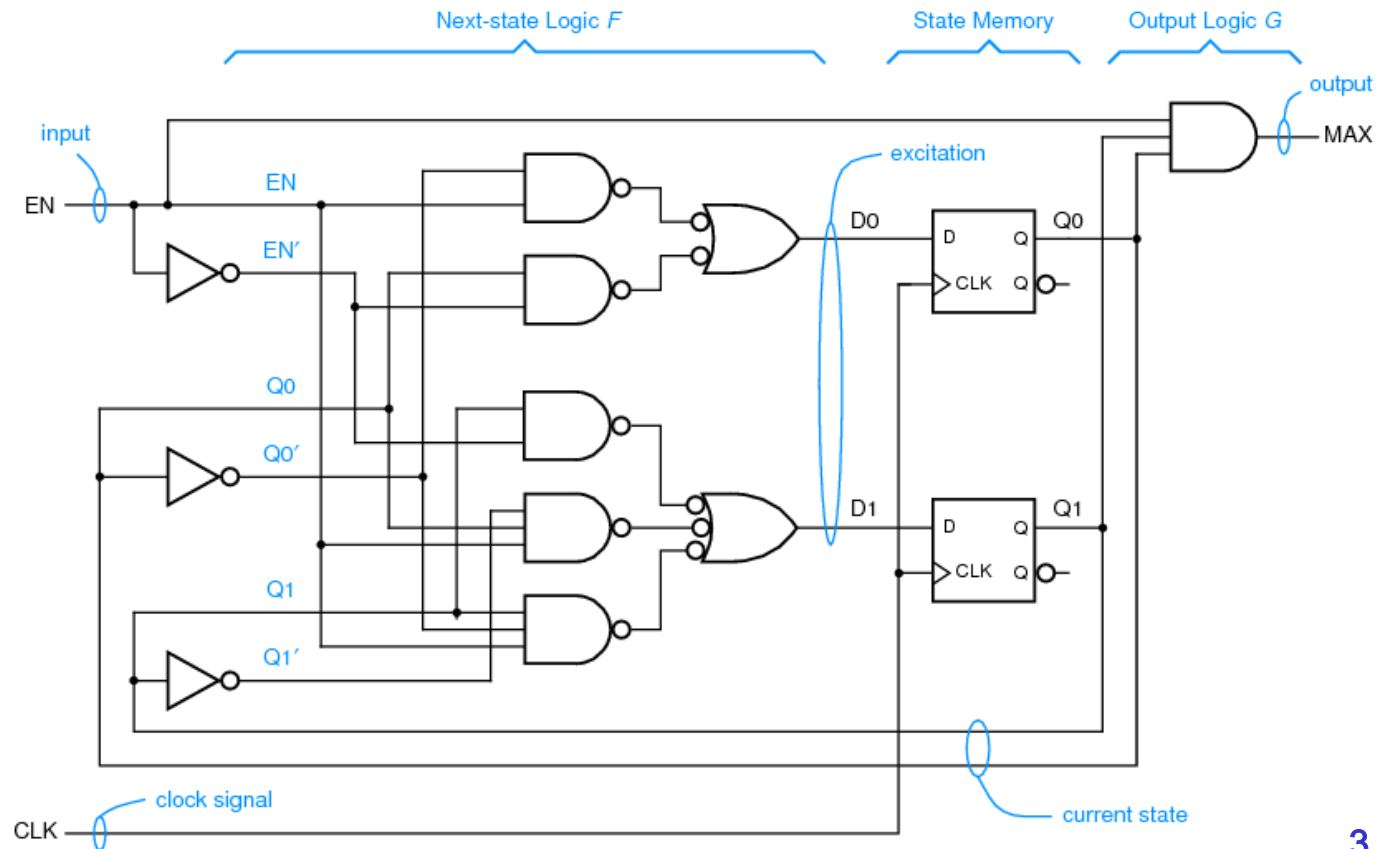
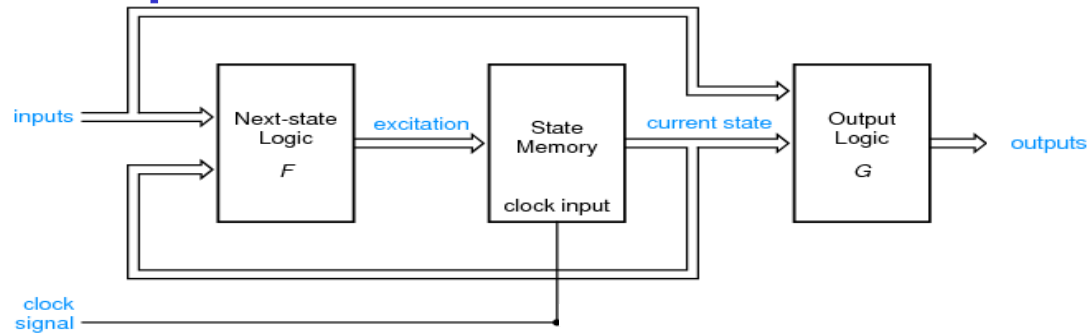


Esquema de una máquina de estados (tipo Moore). **Estado siguiente = $F(\text{estado actual}, \text{entrada})$**
Salida = $G(\text{estado actual})$



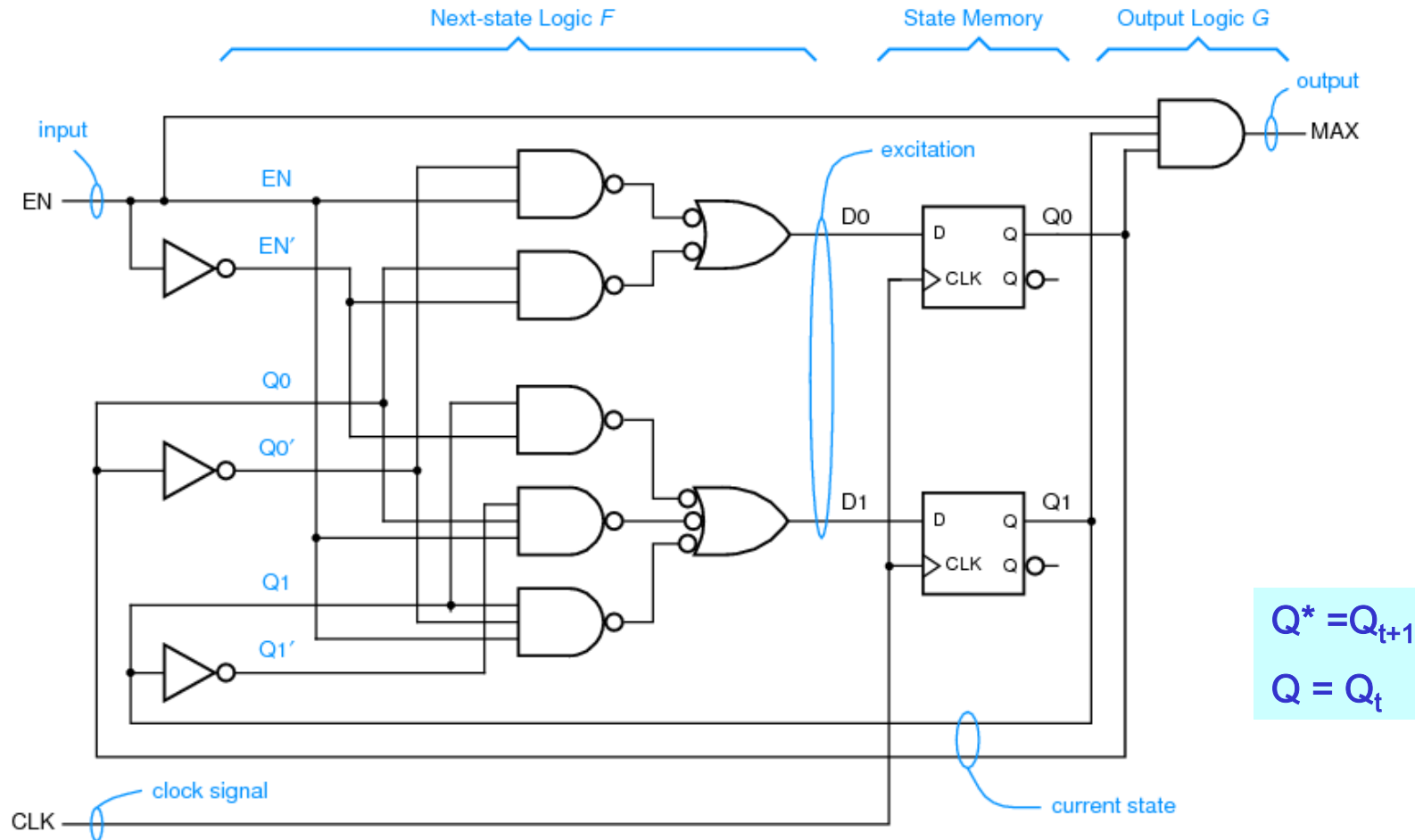
Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: Análisis



Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: Análisis



$$D_0 = Q_0 \cdot EN' + Q_0' \cdot EN$$

$$Q_0^* = D_0 = Q_0 \cdot EN' + Q_0' \cdot EN$$

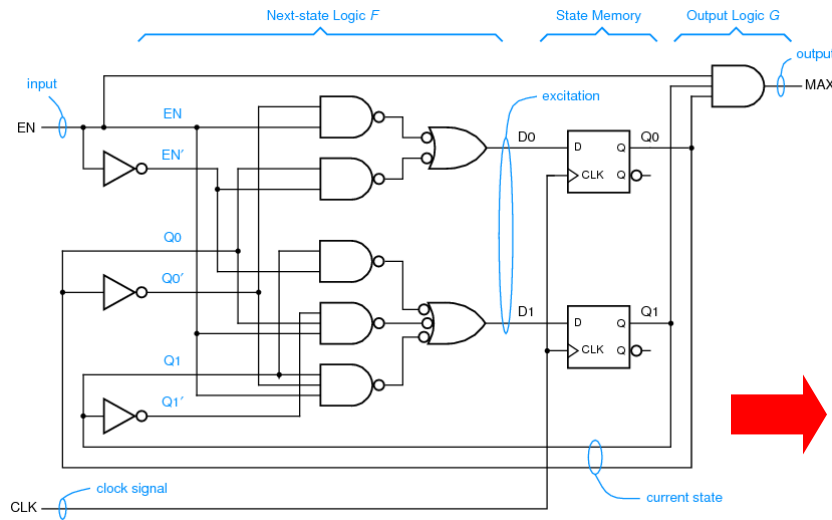
$$D_1 = Q_1 \cdot EN' + Q_1' \cdot Q_0 \cdot EN + Q_1 \cdot Q_0' \cdot EN$$

$$Q_1^* = D_1 = Q_1 \cdot EN' + Q_1' \cdot Q_0 \cdot EN + Q_1 \cdot Q_0' \cdot EN$$

$$MAX = EN \cdot Q_0 \cdot Q_1$$

Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: **Análisis**



$$D0 = Q0 \cdot EN' + Q0' \cdot EN$$

$$D1 = Q1 \cdot EN' + Q1' \cdot Q0 \cdot EN + Q1 \cdot Q0' \cdot EN$$

$$Q0^* = D0 = Q0 \cdot EN' + Q0' \cdot EN$$

$$Q1^* = D1 = Q1 \cdot EN' + Q1' \cdot Q0 \cdot EN + Q1 \cdot Q0' \cdot EN$$

Ecuaciones de estado (evolución)

$$MAX = EN \cdot Q0 \cdot Q1$$

Ecuación de salida

(a)

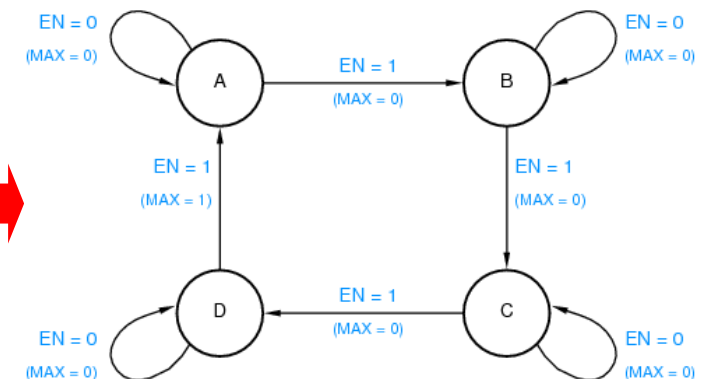
$Q1 \ Q0$	EN	
	0	1
00	00	01
01	01	10
10	10	11
11	11	00
$Q1^* \ Q0^*$		

(b)

S	EN	
	0	1
A	A	B
B	B	C
C	C	D
D	D	A
S^*		

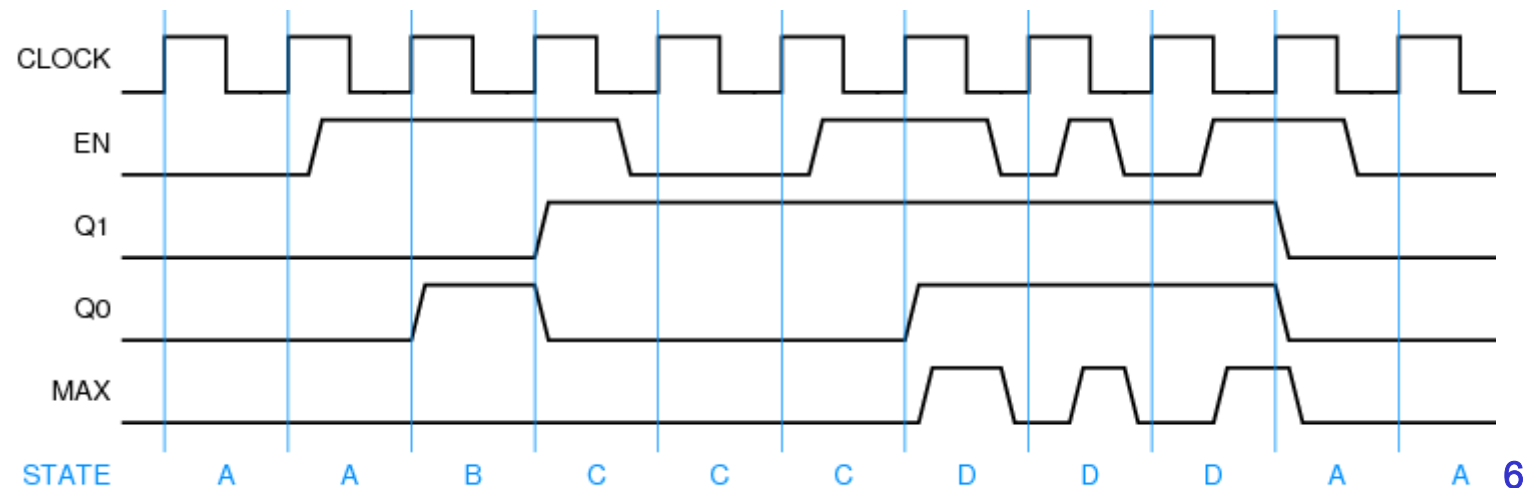
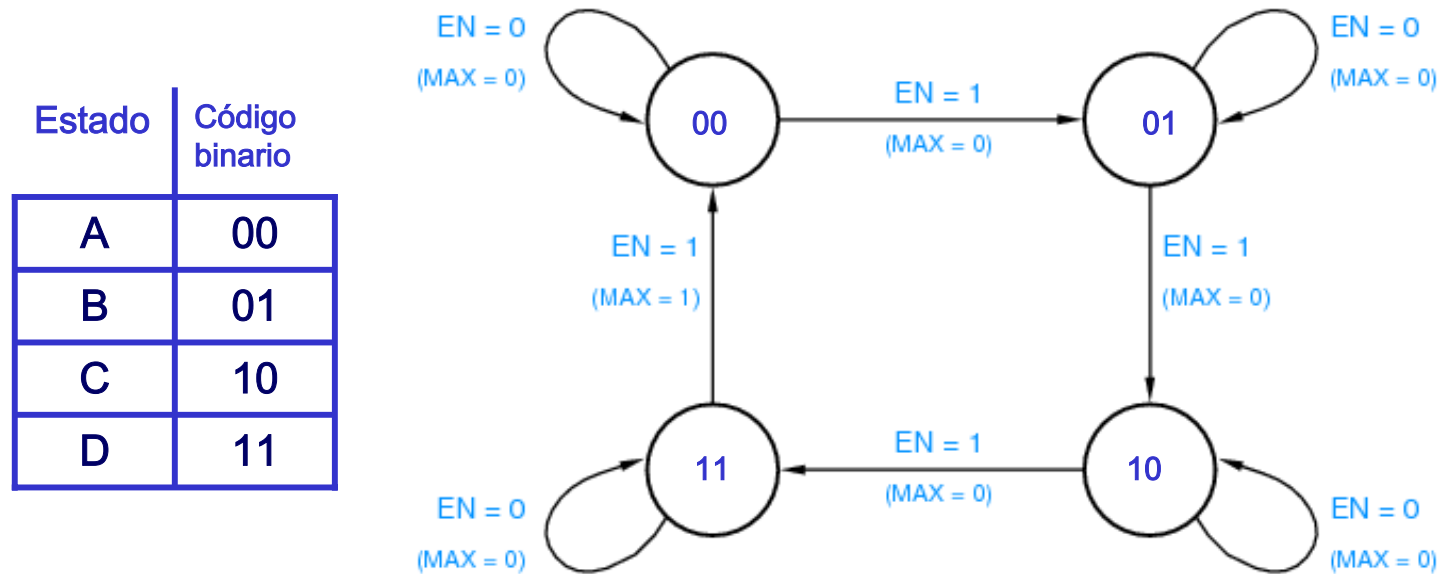
(c)

S	EN	
	0	1
A	A, 0	B, 0
B	B, 0	C, 0
C	C, 0	D, 0
D	D, 0	A, 1
S^*, MAX		



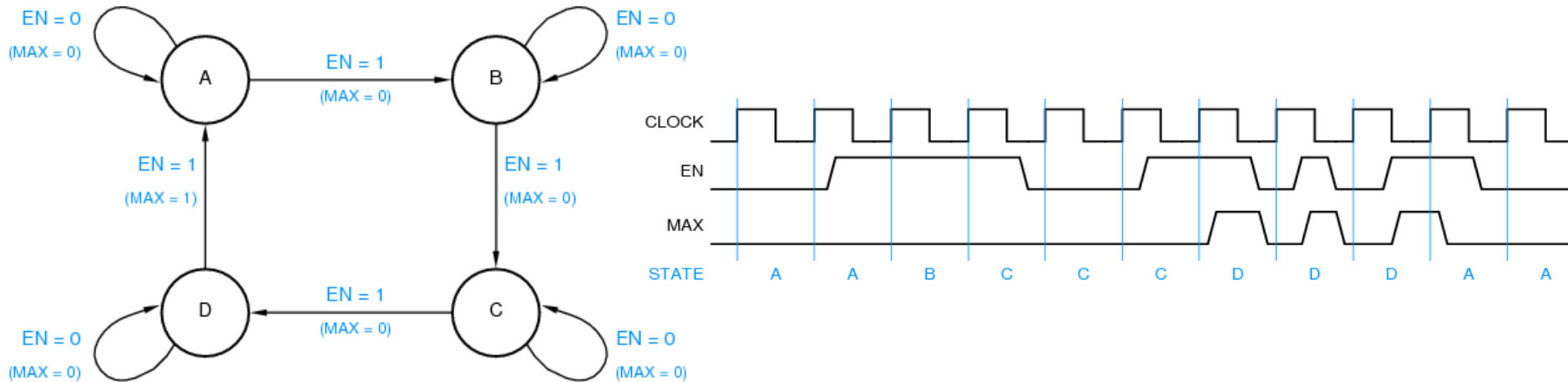
Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: **Análisis**



Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: **Análisis**



Ejemplo de lectura del diagrama de flujo:

... Si se está en el estado A y la entrada EN vale 0 en el momento del flanco activo, el siguiente estado será A.

... Si se está en el estado A y la entrada EN vale 1 en el momento del flanco activo, el siguiente estado será B.

.....y por ser Mealy $z_i = f(Q_i, x_i)$:

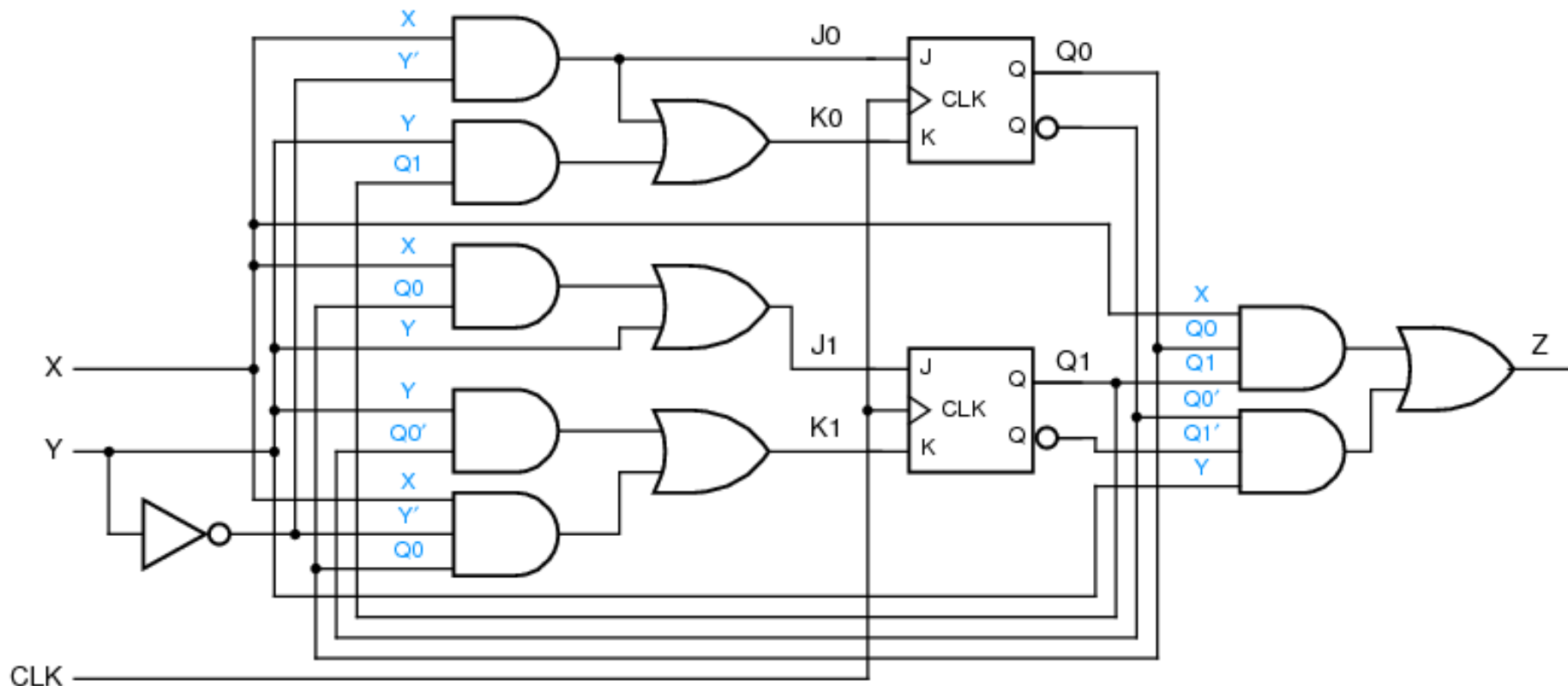
... Si estoy en el estado D, **mientras** la entrada vale 0 la salida vale 0, y **mientras** la entrada vale 1 la salida vale 1 (la salida se producirá con un pequeño retardo desde el cambio de la entrada).

Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: Análisis

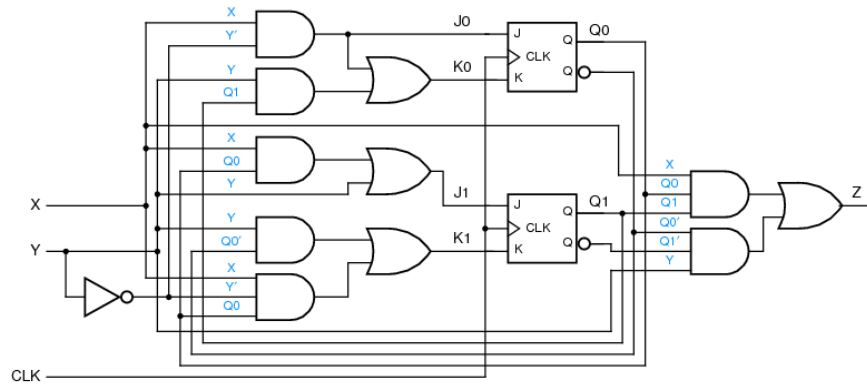
Otro ejemplo de análisis de circuito secuencial realizado con biestables JK activados por flanco ascendente:

La evolución entre estados se realizará dependiendo del valor de las entradas JK en el momento del flanco ascendente.



Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: Análisis



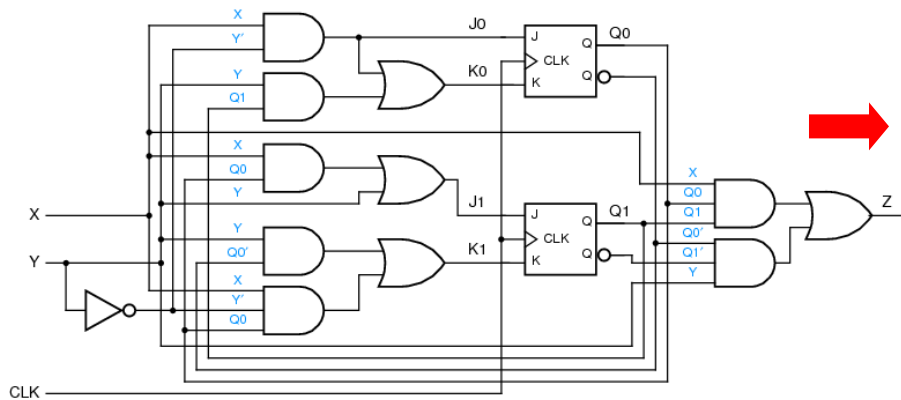
Q1	Q0	X	Y	J1	K1	J0	K0	Q1 _{t+1}	Q0 _{t+1}	Z
0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	0	1	0	1
0	0	1	0	0	0	1	1	0	1	0
0	0	1	1	1	1	0	0	1	0	1
0	1	0	0	0	0	0	0	0	1	0
0	1	0	1	1	0	0	0	1	1	0
0	1	1	0	1	1	1	1	1	0	0
0	1	1	1	1	0	0	0	1	1	0
1	0	0	0	0	0	0	0	1	0	0
1	0	0	1	1	1	0	1	0	0	0
1	0	1	0	0	0	1	1	1	1	0
1	0	1	1	1	1	0	1	0	0	0
1	1	0	0	0	0	0	0	1	1	0
1	1	0	1	1	0	0	1	1	0	0
1	1	1	0	1	1	1	1	0	0	1
1	1	1	1	1	0	0	1	1	0	1

Del circuito se pueden sacar las ecuaciones de J0, K0, J1 y K1, para sacar los estados siguientes $Q0_{t+1}$ y $Q1_{t+1}$ a los que avanzará al flanco activo, o se puede realizar directamente la tabla.

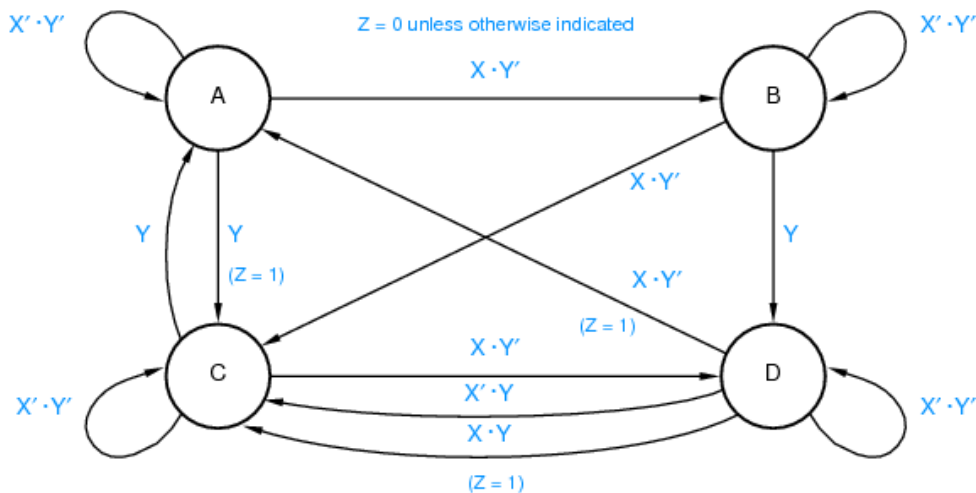
También es preciso obtener la salida, bien a partir de su ecuación o de la tabla.

Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: Análisis



Q1	Q0	X	Y	J1	K1	J0	K0	Q1 _{t+1}	Q0 _{t+1}	Z
0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	0	1	0	1
0	0	1	0	0	0	1	1	0	1	0
0	0	1	1	1	1	0	0	1	0	1
0	1	0	0	0	0	0	0	0	1	0
0	1	0	1	1	0	0	0	1	1	0
0	1	1	0	1	1	1	1	1	0	0
0	1	1	1	1	0	0	0	1	1	0
1	0	0	0	0	0	0	0	1	0	0
1	0	0	1	1	1	0	1	0	0	0
1	0	1	0	0	0	1	1	1	1	0
1	0	1	1	1	1	0	1	0	0	0
1	1	0	0	0	0	0	0	1	1	0
1	1	0	1	1	0	0	1	1	0	0
1	1	1	0	1	1	1	1	0	0	1
1	1	1	1	1	0	0	1	1	0	1



Se ha denominado (siendo estado = Q1 Q0):

Estado 00 = A

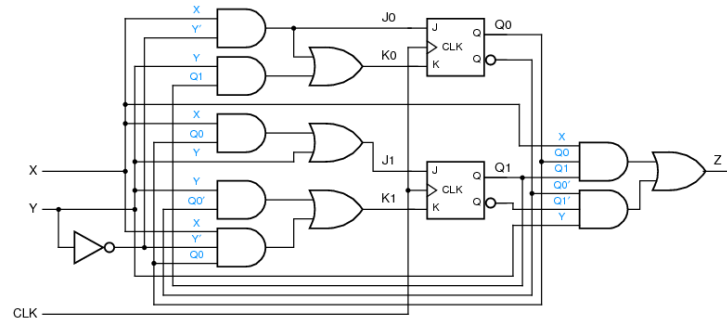
Estado 01 = B

Estado 10 = C

Estado 11 = D

Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: Análisis



Q1	Q0	X	Y	J1	K1	J0	K0	Q1 _{t+1}	Q0 _{t+1}	Z
0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	0	1	0	1
0	0	1	0	0	0	1	1	0	1	0
0	0	1	1	1	1	0	0	1	0	1
0	1	0	0	0	0	0	0	0	1	0
0	1	0	1	1	0	0	0	1	1	0
0	1	1	0	1	1	1	1	1	0	0
0	1	1	1	1	0	0	0	1	1	0
1	0	0	0	0	0	0	0	1	0	0
1	0	0	1	1	1	0	1	0	0	0
1	0	1	0	0	0	1	1	1	1	0
1	0	1	1	1	1	0	1	0	0	0
1	1	0	0	0	0	0	0	1	1	0
1	1	0	1	1	0	0	1	1	0	0
1	1	1	0	1	1	1	1	0	0	1
1	1	1	1	1	0	0	1	1	0	1

Se ha denominado (siendo estado = Q1 Q0):

Estado 00 = A

Estado 01 = B

Estado 10 = C

Estado 11 = D

a)

	XY			
Q1 Q0	00	01	10	11
00	00, 0	10, 1	01, 0	10, 1
01	01, 0	11, 0	10, 0	11, 0
10	10, 0	00, 0	11, 0	00, 0
11	11, 0	10, 0	00, 1	10, 1
	Q1*Q0*, Z			

b)

	XY			
S	00	01	10	11
A	A, 0	C, 1	B, 0	C, 1
B	B, 0	D, 0	C, 0	D, 0
C	C, 0	A, 0	D, 0	A, 0
D	D, 0	C, 0	A, 1	C, 1
	S*, Z			

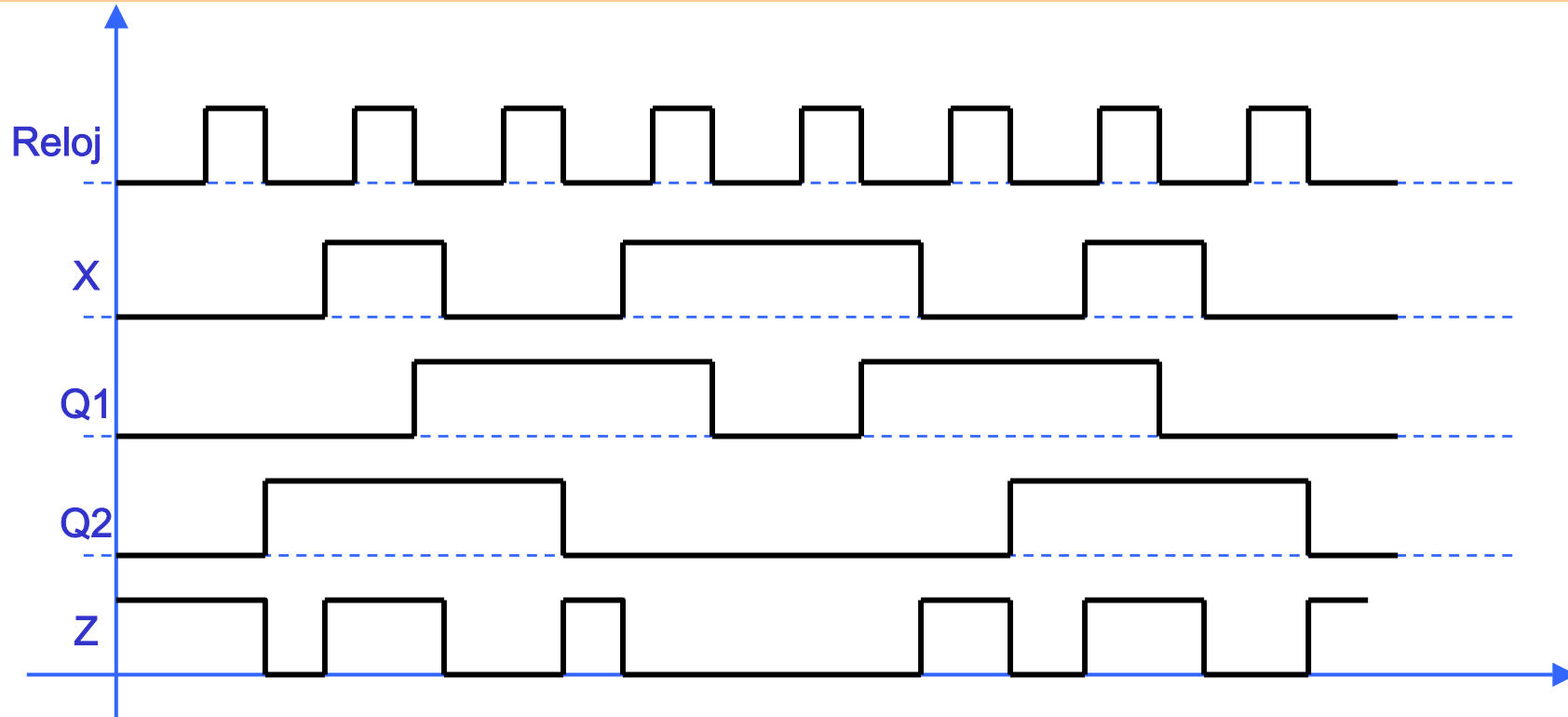


Otra forma de representación de la tabla de flujo

Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: Análisis

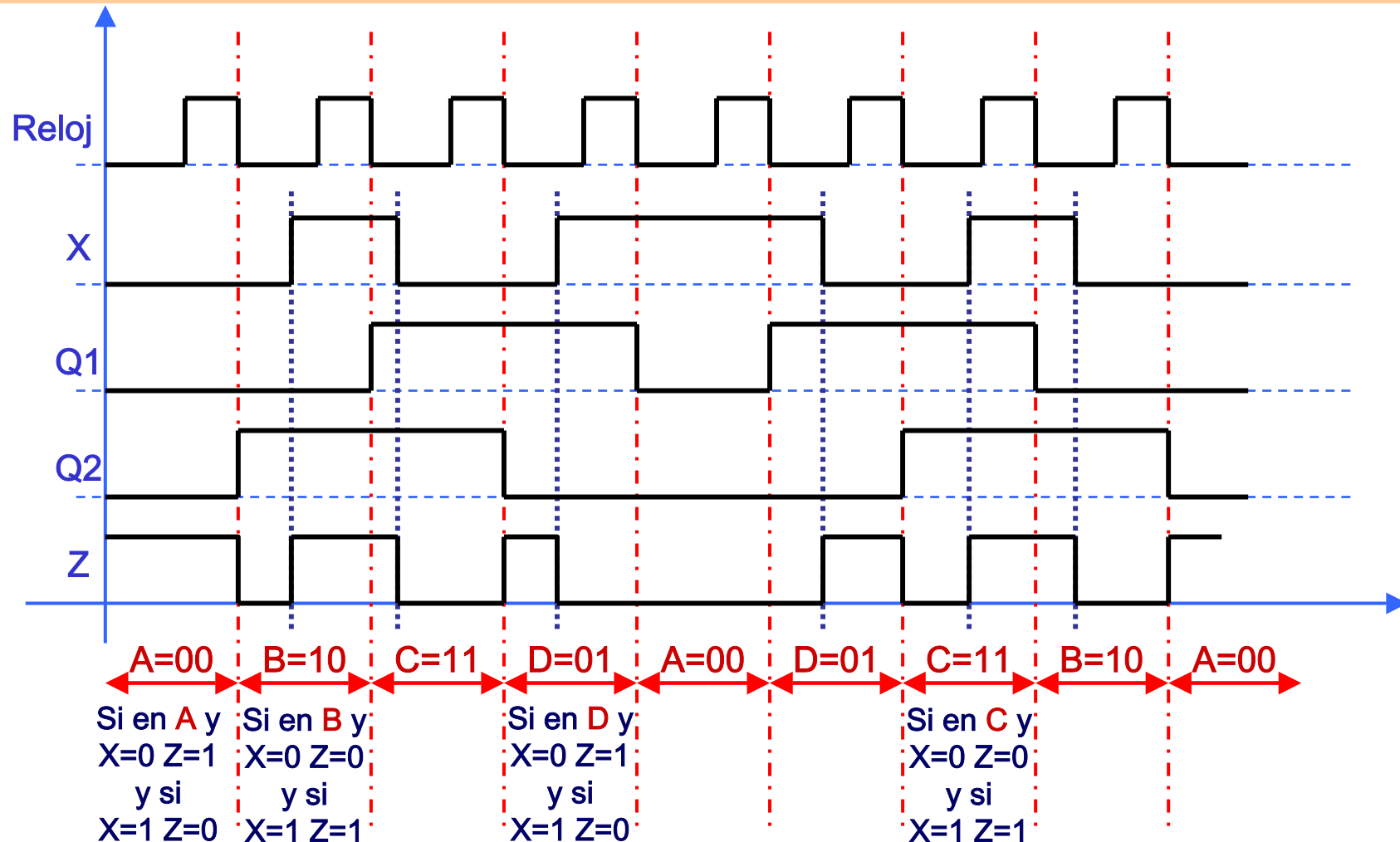
Construir el diagrama de flujo de un sistema que tiene una entrada de reloj, una entrada de dato X y una salida Z y que evoluciona como se indica en la figura del diagrama de tiempos. (Se señalan los valores de las variables de estado Q1 y Q2)



Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: Análisis

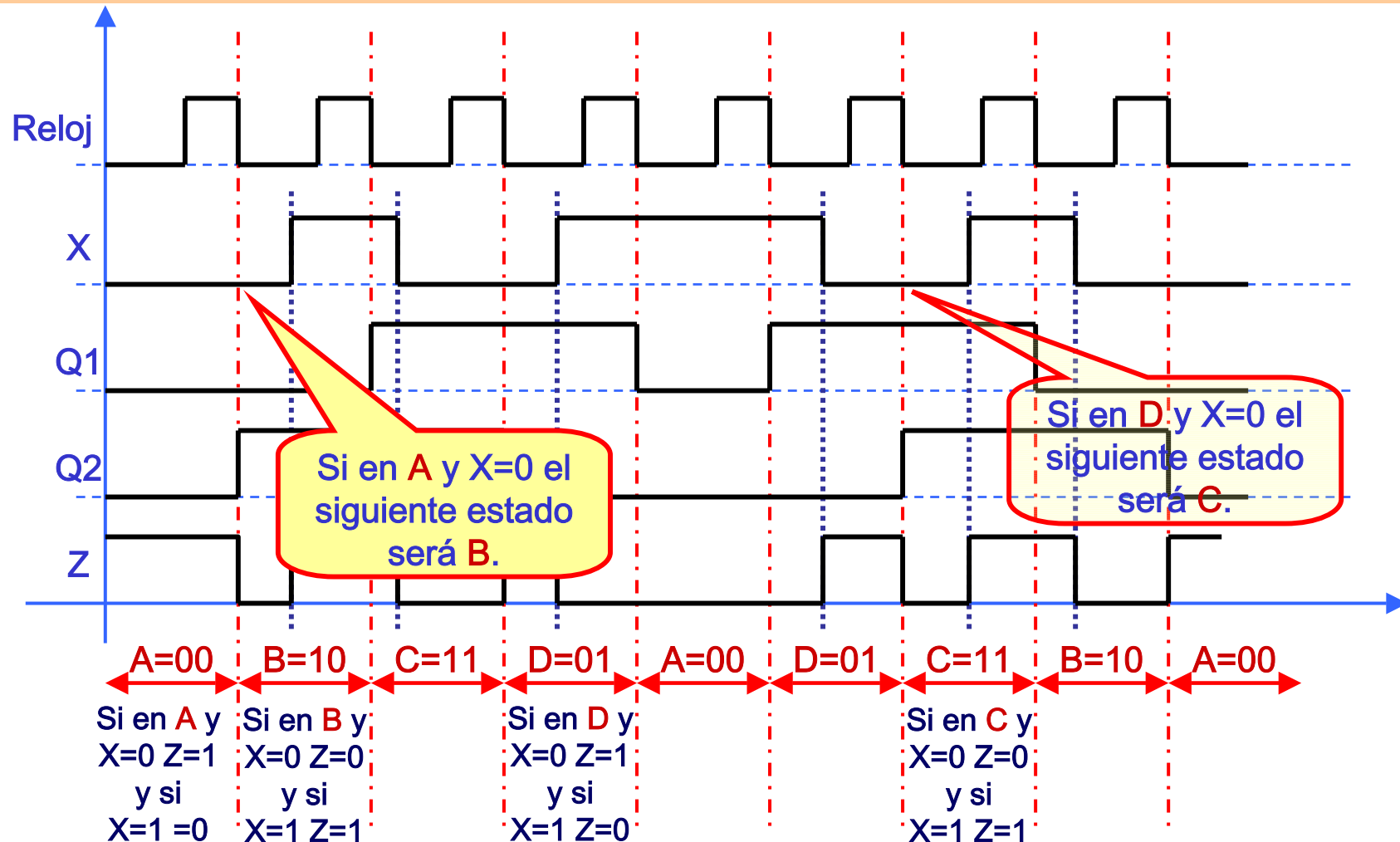
Construir el diagrama de flujo de un sistema que tiene una entrada de reloj, una entrada de dato X y una salida Z y que evoluciona como se indica en la figura del diagrama de tiempos. (Se señalan los valores de las variables de estado Q1 y Q2)



Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: Análisis

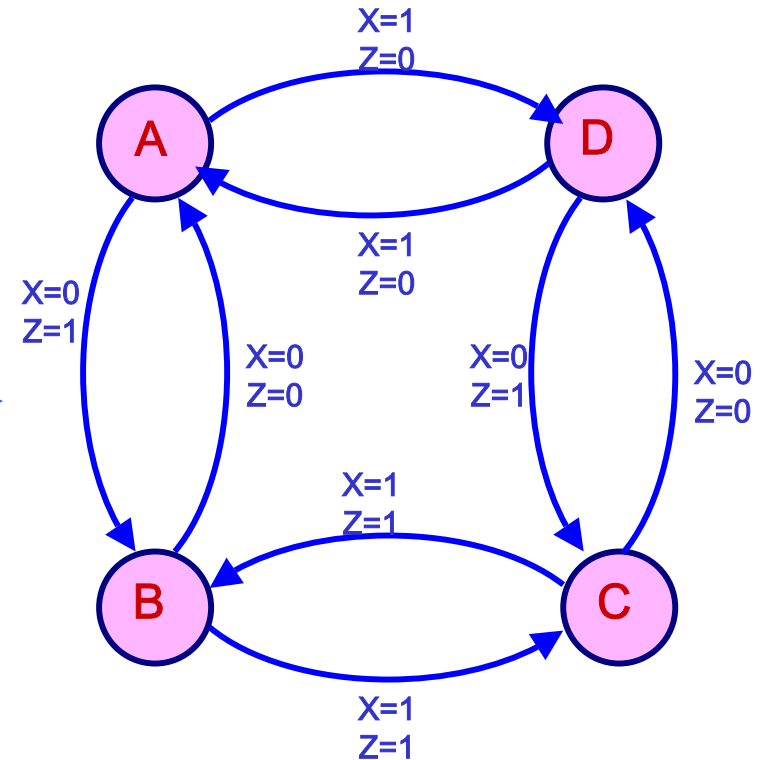
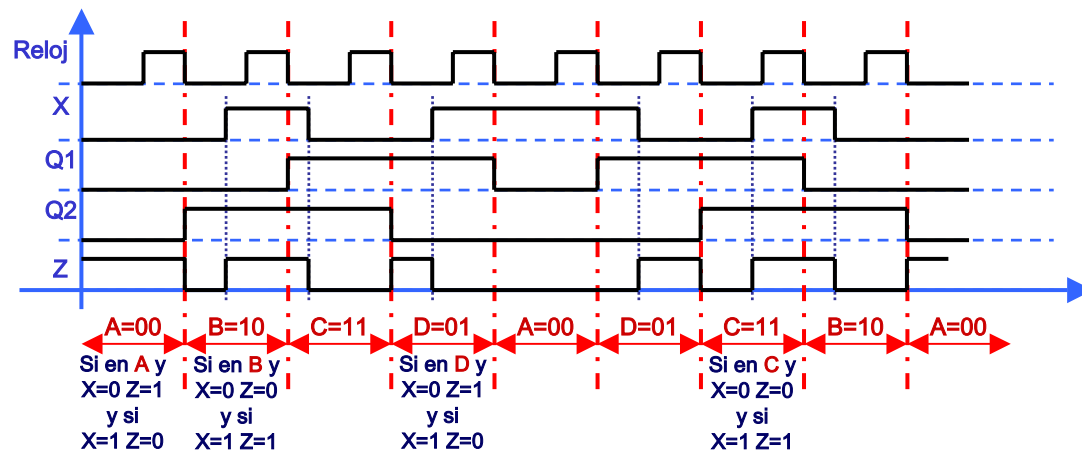
Construir el diagrama de flujo de un sistema que tiene una entrada de reloj, una entrada de dato X y una salida Z y que evoluciona como se indica en la figura del diagrama de tiempos. (Se señalan los valores de las variables de estado Q1 y Q2)



Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: Análisis

Construir el diagrama de flujo de un sistema que tiene una entrada de reloj, una entrada de dato X y una salida Z y que evoluciona como se indica en la figura del diagrama de tiempos. (Se señalan los valores de las variables de estado Q1 y Q2)



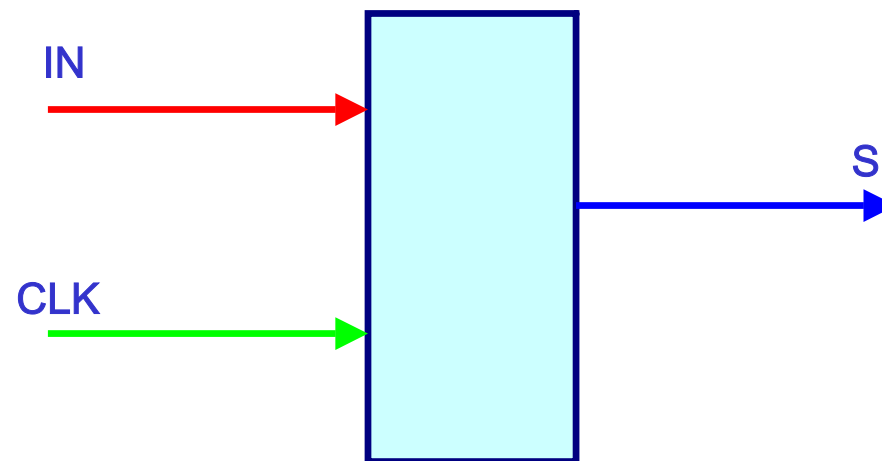
Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: **SÍNTESIS** (diseño)

Diseñar un sistema con una entrada IN y una salida S así como una entrada de reloj.

El sistema deberá sacar un 1 en la salida durante un ciclo de reloj, siempre que en la entrada se haya reconocido dos bits iguales en los dos flancos ascendentes consecutivos de la señal de reloj.

En caso contrario deberá sacar 0 por la salida S.



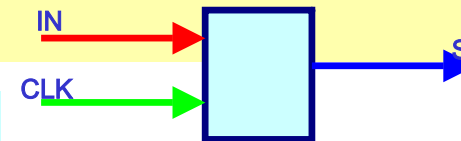
Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: **SÍNTESIS** (diseño)

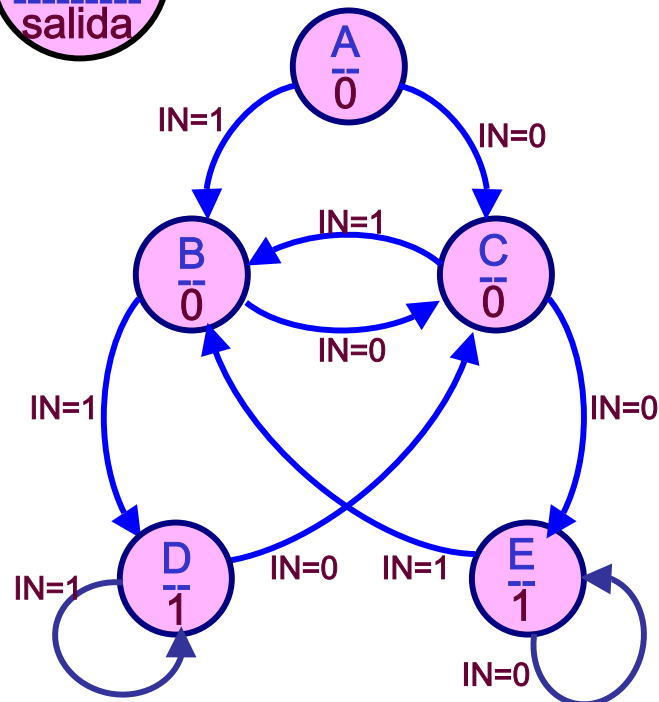
Diseñar un sistema con una entrada IN y una salida S así como una entrada de reloj.

El sistema deberá sacar un 1 en la salida durante un ciclo de reloj, siempre que en la entrada se haya reconocido dos bits iguales en los dos flancos ascendentes consecutivos de la señal de reloj.

En caso contrario deberá sacar 0 por la salida S.



Será un sistema Moore por tenerse que mantener la salida durante un ciclo de reloj, independientemente de que durante el ciclo cambie la entrada



Codificación de estados

A=000
B=001
C=010
D=011
E=100

La salida
mientras estoy en
el estado A (000)
es 0

Q2	Q1	Q0	IN	Q2*	Q1*	Q0*	S
0	0	0	0	0	1	0	0
0	0	0	1	0	0	1	0
0	0	1	0	0	1	0	0
0	0	1	1	0	1	1	0
0	1	0	0	1	0	0	0
0	1	0	1	0	0	1	0
0	1	1	0	0	1	0	1
0	1	1	1	0	1	1	1
1	0	0	0	1	0	0	1
1	0	0	1	0	0	1	1
1	0	1	0	X	X	X	X
1	0	1	1	X	X	X	X
....	X	X	X	X

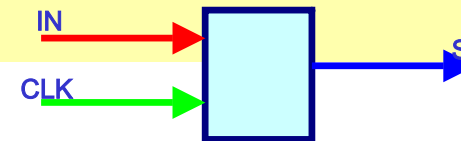
Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: **SÍNTESIS** (diseño)

Diseñar un sistema con una entrada IN y una salida S así como una entrada de reloj.

El sistema deberá sacar un 1 en la salida durante un ciclo de reloj, siempre que en la entrada se haya reconocido dos bits iguales en los dos flancos ascendentes consecutivos de la señal de reloj.

En caso contrario deberá sacar 0 por la salida S.



Q2	Q1	Q0	IN	Q2*	Q1*	Q0*	S
0	0	0	0	0	1	0	0
0	0	0	1	0	0	1	0
0	0	1	0	0	1	0	0
0	0	1	1	0	1	1	0
0	1	0	0	1	0	0	0
0	1	0	1	0	0	1	0
0	1	1	0	0	1	0	1
0	1	1	1	0	1	1	1
1	0	0	0	1	0	0	1
1	0	0	1	0	0	1	1
1	0	1	0	X	X	X	X
1	0	1	1	X	X	X	X
....	X	X	X	X

Precisa 3 variables de estado, luego precisa 3 biestables tipo D activados por flanco ascendente

Obtendremos la evolución de los estados siguientes si hacemos $Q2^* = D2$ (entrada al biestable 2)

Q2,Q1		Q0,IN			
		00	01	11	10
00		0	0	0	0
01		1	0	0	0
11		X	X	X	X
10		1	0	X	X

Lo mismo haremos para D1, D0 y S

Q2*

$$Q2^* = D2 = Q1 \cdot Q0' \cdot IN' + Q2 \cdot IN'$$

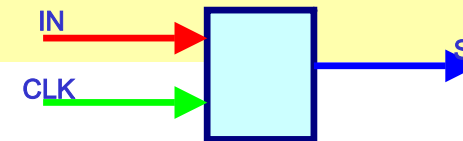
Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: **SÍNTESIS** (diseño)

Diseñar un sistema con una entrada IN y una salida S así como una entrada de reloj.

El sistema deberá sacar un 1 en la salida durante un ciclo de reloj, siempre que en la entrada se haya reconocido dos bits iguales en los dos flancos ascendentes consecutivos de la señal de reloj.

En caso contrario deberá sacar 0 por la salida S.



Q2	Q1	Q0	IN	Q2*	Q1*	Q0*	S
0	0	0	0	0	1	0	0
0	0	0	1	0	0	1	0
0	0	1	0	0	1	0	0
0	0	1	1	0	1	1	0
0	1	0	0	1	0	0	0
0	1	0	1	0	0	1	0
0	1	1	0	0	1	0	1
0	1	1	1	0	1	1	1
1	0	0	0	1	0	0	1
1	0	0	1	0	0	1	1
1	0	1	0	X	X	X	X
1	0	1	1	X	X	X	X
....	X	X	X	X

Lo mismo haremos para D1, D0 y S

Q0, IN

Q2, Q1

	00	01	11	10
00	1	0	1	1
01	0	0	1	1
11	X	X	X	X
10	0	0	X	X

Q1*

$$Q1^* = D1 = Q0 + Q2' \cdot Q1' \cdot IN$$

Q0, IN

Q2, Q1

	00	01	11	10
00	0	1	1	0
01	0	1	1	0
11	X	X	X	X
10	0	1	X	X

Q0*

$$Q0^* = D0 = IN$$

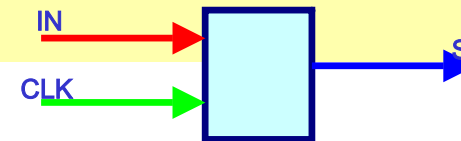
Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: **SÍNTESIS** (diseño)

Diseñar un sistema con una entrada IN y una salida S así como una entrada de reloj.

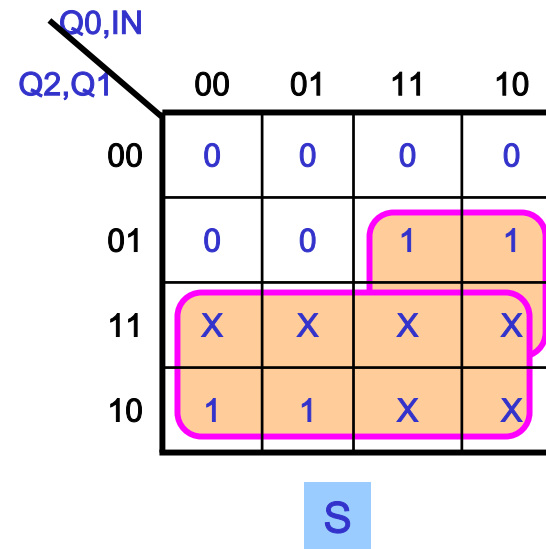
El sistema deberá sacar un 1 en la salida durante un ciclo de reloj, siempre que en la entrada se haya reconocido dos bits iguales en los dos flancos ascendentes consecutivos de la señal de reloj.

En caso contrario deberá sacar 0 por la salida S.



Q2	Q1	Q0	IN	Q2*	Q1*	Q0*	S
0	0	0	0	0	1	0	0
0	0	0	1	0	0	1	0
0	0	1	0	0	1	0	0
0	0	1	1	0	1	1	0
0	1	0	0	1	0	0	0
0	1	0	1	0	0	1	0
0	1	1	0	0	1	0	1
0	1	1	1	0	1	1	1
1	0	0	0	1	0	0	1
1	0	0	1	0	0	1	1
1	0	1	0	X	X	X	X
1	0	1	1	X	X	X	X
....	X	X	X	X

Lo mismo haremos para D1, D0 y S



$$S = Q2 + Q1 \cdot Q0$$

Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: **SÍNTESIS** (diseño)

Diseñar un sistema con una entrada IN y una salida S así como una entrada de reloj.

El sistema deberá sacar un 1 en la salida durante un ciclo de reloj, siempre que en la entrada se haya reconocido dos bits iguales en los dos flancos ascendentes consecutivos de la señal de reloj.

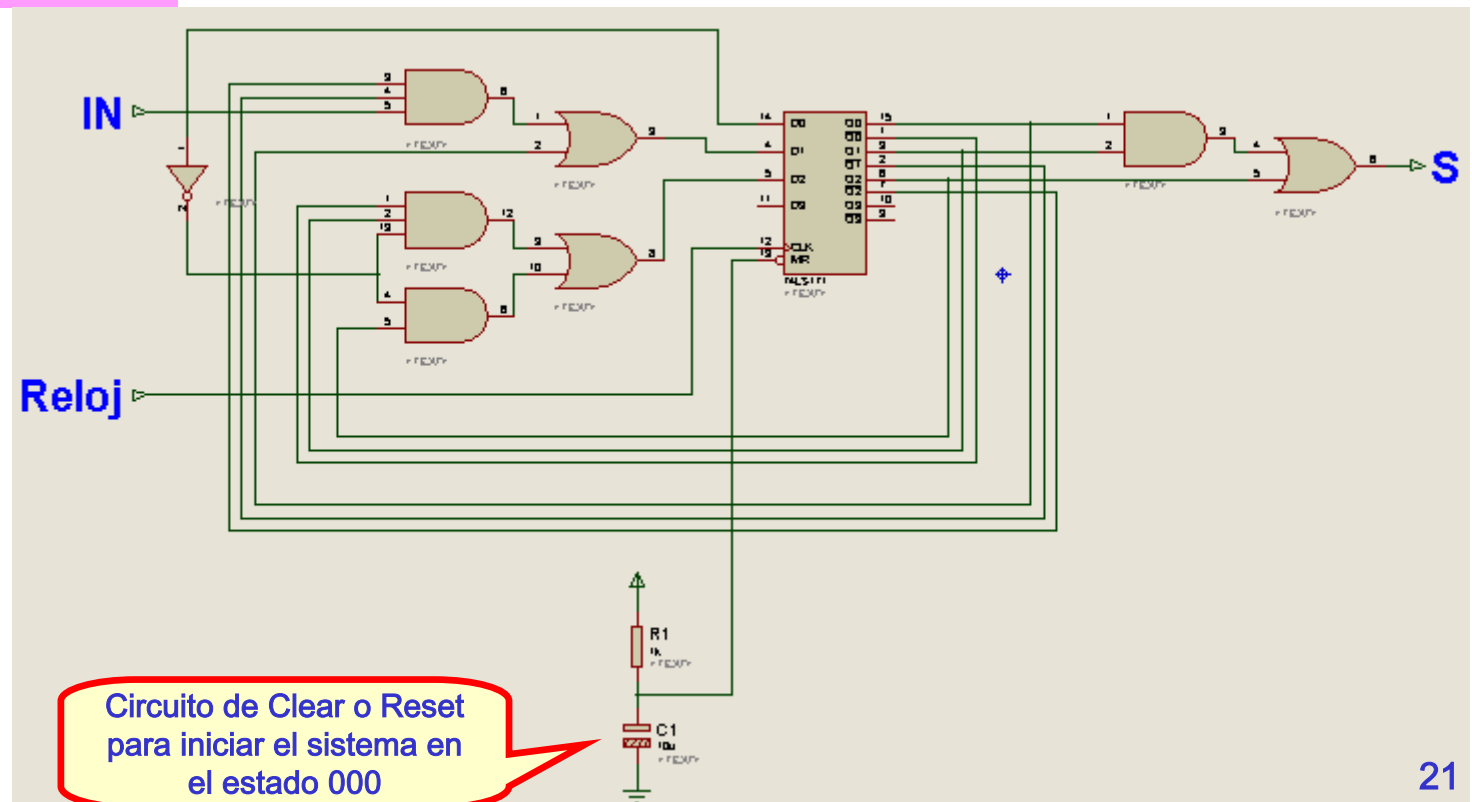
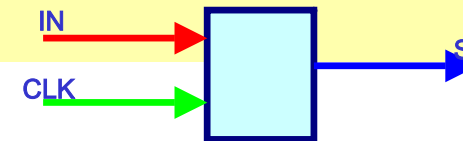
En caso contrario deberá sacar 0 por la salida S.

$$Q2^* = D2 = Q1 \cdot Q0' \cdot IN' + Q2 \cdot IN'$$

$$Q1^* = D1 = Q0 + Q2' \cdot Q1' \cdot IN$$

$$Q0^* = D0 = IN$$

$$S = Q2 + Q1 \cdot Q0$$



Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: **SÍNTESIS** (diseño)

Observaciones en el proceso de Diseño:

Codificación de estados:

Una vez diseñada la evolución de estados, es preciso dar un código a cada estado. Si hay m estados serán precisos n bits **al menos** para codificar los estados, tal que $2^n > m$.

Interesa que **estados consecutivos presenten códigos adyacentes** (cambio de un único bit en sus códigos) **siempre que sea posible**. Con ello se evita transiciones intermedias de los estados durante la evolución entre Q_t y Q_{t+1} que en la salida pueden dar transitorios no deseados.

Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: **SÍNTESIS** (diseño)

Observaciones en el proceso de Diseño:

Minimización del número de estados (simplificación de estados):

Una vez diseñada la evolución de estados, es preciso reconocer si existen estados equivalentes que pueden considerarse idénticos y así simplificar el diseño resultante.

Estados equivalentes: Aquellos que para las mismas entradas evolucionan a los mismos estados y producen las mismas salidas.

En ocasiones es difícil distinguir los estados equivalentes, ya que dos estados pueden evolucionar a estados diferentes, pero que son equivalentes a su vez, lo que hace que los estados origen serían también equivalentes.

Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: **SÍNTESIS** (diseño)

Observaciones en el proceso de Diseño:

Minimización del número de estados (simplificación de estados):

Metodología:

1. Generar clases agrupando los estados que tienen las mismas salidas para las mismas entradas, especificando la clase de los estados siguientes a los que evolucionan con las distintas entradas.
2. Generar nuevas clases dentro de las ya generadas en el punto anterior, agrupando aquellos estados que tienen las mismas clases en los estados a los que evolucionan.
3. Realizar la operación del punto 2 de forma repetida hasta que sea imposible el desdoblar las clases.
4. Los grupos de cada clase son grupos de estados equivalentes.

Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: **SÍNTESIS** (diseño)

Observaciones en el proceso de Diseño:

Minimización del número de estados (simplificación de estados): Ejemplo

Metodología:

1. Generar clases agrupando los estados que tienen las mismas salidas para las mismas entradas, especificando la clase de los estados siguientes a los que evolucionan con las distintas entradas.

Partiendo de una tabla de flujo como ésta realizamos la primera clasificación:

Q_t	Q_{t+1}/Z para $X=0$	Q_{t+1}/Z para $X=1$
Q_1	$Q_3/0$	$Q_5/0$
Q_2	$Q_4/0$	$Q_6/0$
Q_3	$Q_3/0$	$Q_5/1$
Q_4	$Q_4/0$	$Q_6/1$
Q_5	$Q_5/0$	$Q_1/0$
Q_6	$Q_6/0$	$Q_2/0$



Clase	Estados	Clases del próximo estado para $X=0$ $X=1$	
A	Q_1	B	A
	Q_2	B	A
	Q_5	A	A
	Q_6	A	A
B	Q_3	B	A
	Q_4	B	A

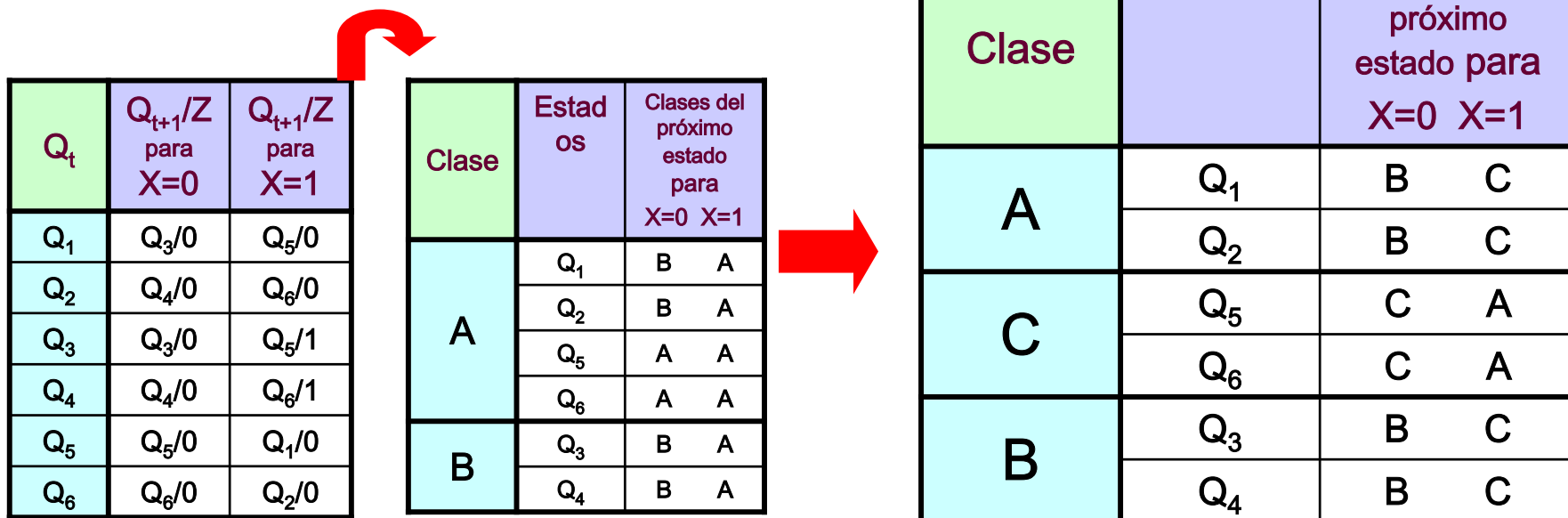
Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: **SÍNTESIS** (diseño)

Minimización del número de estados (simplificación de estados): Ejemplo

Metodología:

1. Generar clases agrupando los estados que tienen las mismas salidas para las mismas entradas, especificando la clase de los estados siguientes a los que evolucionan con las distintas entradas.
2. Generar nuevas clases dentro de las ya generadas en el punto anterior, agrupando aquellos estados que tienen las mismas clases en los estados a los que evolucionan.



Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: SÍNTESIS (diseño)

Minimización del número de estados (simplificación de estados): Ejemplo

Metodología:

1. Generar clases agrupando los estados que tienen las mismas salidas para las mismas entradas, especificando la clase de los estados siguientes a los que evolucionan con las distintas entradas.
2. Generar nuevas clases dentro de las ya generadas en el punto anterior, agrupando aquellos estados que tienen las mismas clases en los estados a los que evolucionan.

Clase	Estados	Clases del próximo estado para X=0 X=1	
A	Q ₁	B	A
	Q ₂	B	A
C	Q ₅	A	A
	Q ₆	A	A
B	Q ₃	B	A
	Q ₄	B	A

3. Realizar la operación del punto 2 de forma repetida hasta que sea imposible el desdoblar las clases.
4. Los grupos de cada clase son grupos de estados equivalentes.



Q1 es equivalente con Q2
Q5 es equivalente con Q6
Q3 es equivalente con Q4
y de la tabla original queda:

Q _t	Q _{t+1} /Z para X=0	Q _{t+1} /Z para X=1
Q ₁	Q ₃ /0	Q ₅ /0
Q ₅	Q ₅ /0	Q ₁ /0
Q ₃	Q ₃ /0	Q ₅ /1

Tabla equivalente simplificada

sustituyendo Q2 por Q1, Q6 por Q5 y Q4 por Q3

Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: **SÍNTESIS** (diseño)

Otro ejemplo de Diseño: Con minimización del número de estados (simplificación de estados):

Diseñar un cambiador de código serie que pase de código binario natural de tres bits a código Gray de tres bits. Los bits se reciben de mayor a menor peso. Una vez recibido el tercer bit, comienza la recepción del siguiente dato binario.

Realizarlo de tal forma que en todo momento haga la conversión, es decir que en el momento en que cambie la entrada, debe cambiar la salida.

BINARIO	GRAY
000	000
001	001
010	011
011	010
100	110
101	111
110	101
111	100

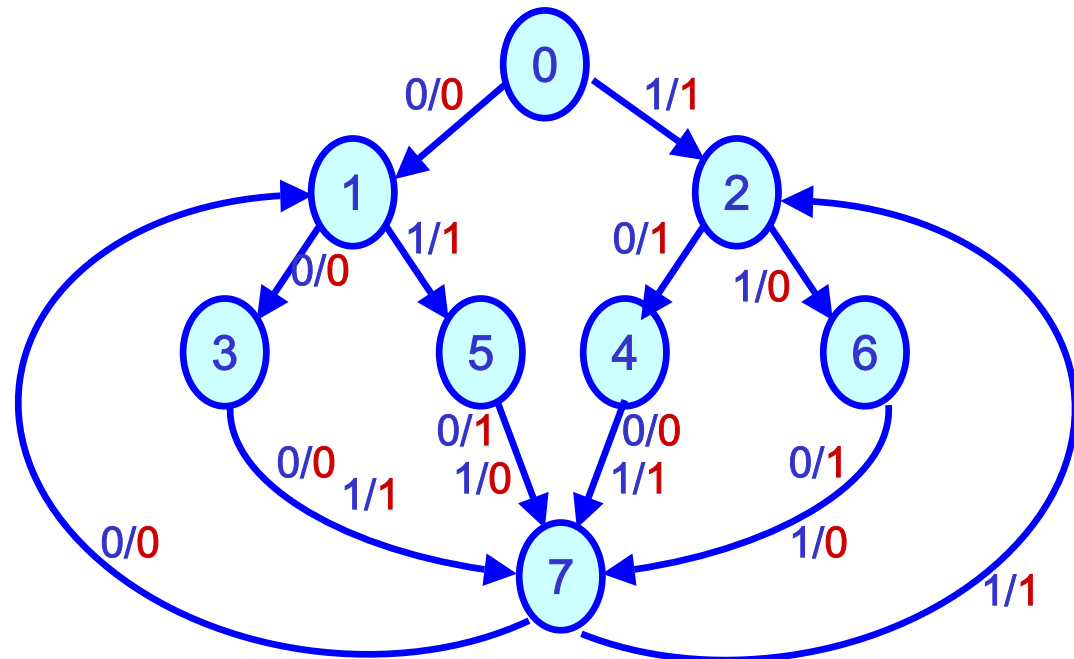
Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: **SÍNTESIS** (diseño)

Otro ejemplo de Diseño: Con minimización del número de estados (simplificación de estados):

Diseñar un cambiador de código serie que pase de código binario natural de tres bits a código Gray de tres bits. Los bits se reciben de mayor a menor peso. Una vez recibido el tercer bit, comienza la recepción del siguiente dato binario. Realizarlo de tal forma que en todo momento haga la conversión, es decir que en el momento en que cambie la entrada, debe cambiar la salida.

BINARIO	GRAY
000	000
001	001
010	011
011	010
100	110
101	111
110	101
111	100



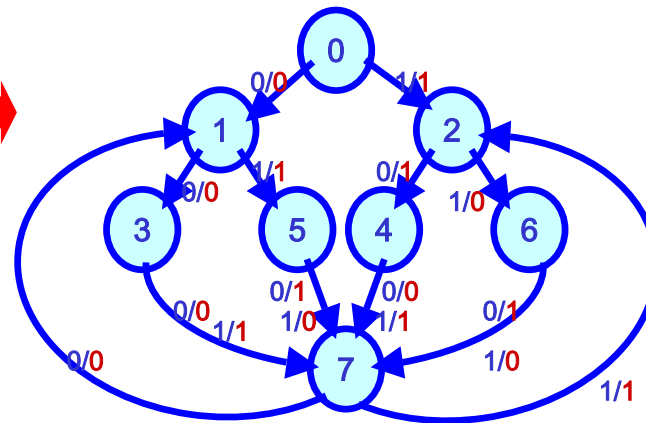
Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: **SÍNTESIS** (diseño)

Otro ejemplo de Diseño: Diseñar un cambiador de código serie que pase de código binario natural de tres bits a código Gray de tres bits. Los bits se reciben de mayor a menor peso. Una vez recibido el tercer bit, comienza la recepción del siguiente dato binario. Realizarlo de tal forma que en todo momento haga la conversión, es decir que en el momento en que cambie la entrada, debe cambiar la salida.

1º.-Clasificación por salidas iguales

BINARIO	GRAY
000	000
001	001
010	011
011	010
100	110
101	111
110	101
111	100



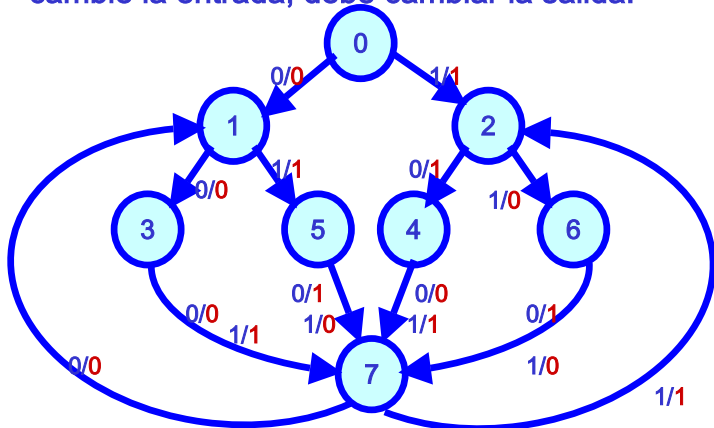
Estado actual	Salidas para IN=0 IN=1		Clase
0	0	1	A
1	0	1	A
2	1	0	B
3	0	1	A
4	0	1	A
5	1	0	B
6	1	0	B
7	0	1	A

Pertenecerán a la misma clase los estados que presentan para las mismas entradas las mismas salidas

Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: **SÍNTESIS** (diseño)

Otro ejemplo de Diseño: Diseñar un cambiador de código serie que pase de código binario natural de tres bits a código Gray de tres bits. Los bits se reciben de mayor a menor peso. Una vez recibido el tercer bit, comienza la recepción del siguiente dato binario. Realizarlo de tal forma que en todo momento haga la conversión, es decir que en el momento en que cambie la entrada, debe cambiar la salida.



Son clases diferentes aquellos estados que presentan diferentes estados siguientes para las mismas entradas

Clasificación y verificación de clases en próximo estado y nueva clasificación

Estado actual	Salidas para IN=0 IN=1		Clase
0	0	1	A
1	0	1	A
2	1	0	B
3	0	1	A
4	0	1	A
5	1	0	B
6	1	0	B
7	0	1	A

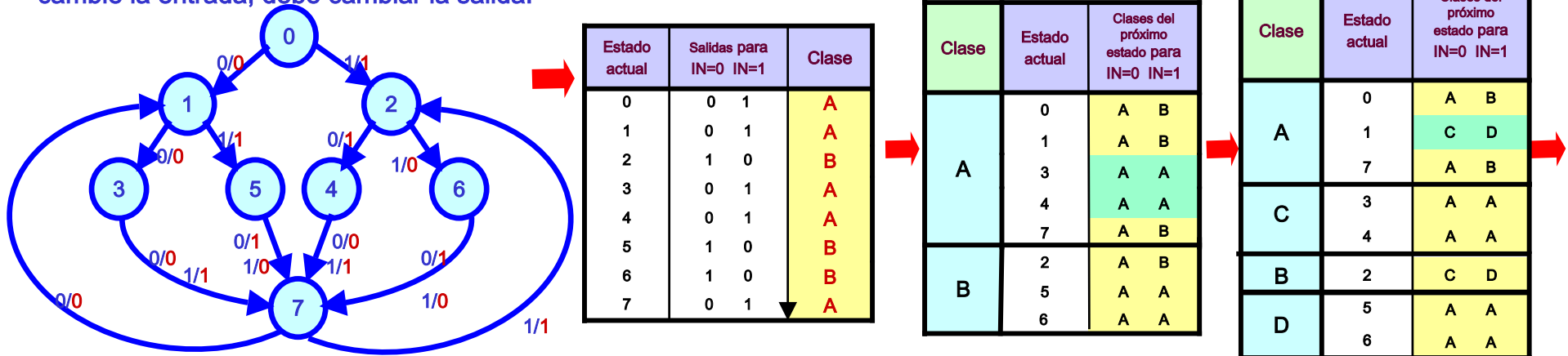
Clase	Estado actual	Clases del próximo estado para IN=0 IN=1	
A	0	A	B
	1	A	B
	3	A	A
	4	A	A
B	7	A	B
	2	A	B
	5	A	A
	6	A	A

Clase	Estado actual	Clases del próximo estado para IN=0 IN=1	
A	0	A	B
	1	C	D
	7	A	B
C	3	A	A
B	4	A	A
	2	C	D
D	5	A	A
	6	A	A

Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: **SÍNTESIS** (diseño)

Otro ejemplo de Diseño: Diseñar un cambiador de código serie que pase de código binario natural de tres bits a código Gray de tres bits. Los bits se reciben de mayor a menor peso. Una vez recibido el tercer bit, comienza la recepción del siguiente dato binario. Realizarlo de tal forma que en todo momento haga la conversión, es decir que en el momento en que cambie la entrada, debe cambiar la salida.



Clase	Estado actual	Clases del próximo estado para	
		IN=0	IN=1
A	0	E	B
E	7	E	B
C	3	A	A
B	2	C	D
D	5	A	A
	6	A	A

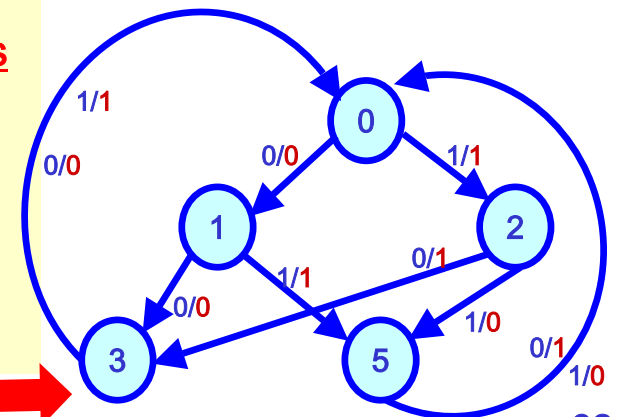
Una vez que no se pueden desdoblar clases se concluye con los **estados equivalentes** que tienen para las mismas entradas las mismas salidas y evolucionan a los mismos estados

Estados equivalentes

0 y 7

3 y 4

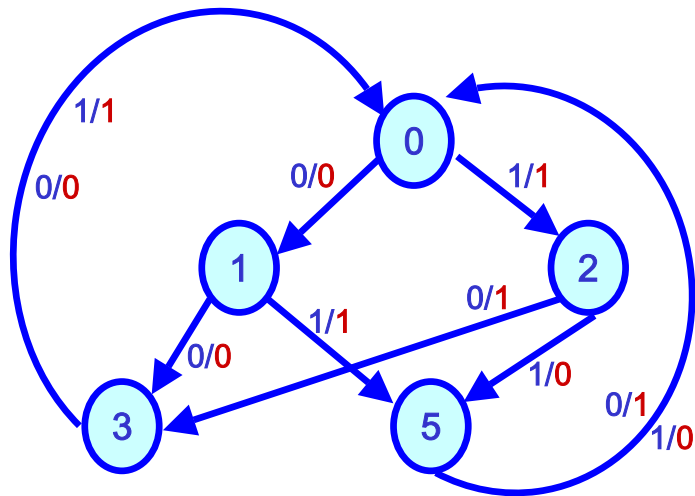
5 y 6



Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: **SÍNTESIS** (diseño)

Otro ejemplo de Diseño: Diseñar un cambiador de código serie que pase de código binario natural de tres bits a código Gray de tres bits. Los bits se reciben de mayor a menor peso. Una vez recibido el tercer bit, comienza la recepción del siguiente dato binario. Realizarlo de tal forma que en todo momento haga la conversión, es decir que en el momento en que cambie la entrada, debe cambiar la salida.



Codificando los estados:

0 = 000

1 = 001

2 = 010

3 = 011

5 = 101



Q2	Q1	Q0	IN	Q2*	Q1*	Q0*	OUT
0	0	0	0	0	0	1	0
0	0	0	1	0	1	0	1
0	0	1	0	0	1	1	0
0	0	1	1	1	0	1	1
0	1	0	0	0	1	1	1
0	1	0	1	1	0	1	0
0	1	1	0	0	0	0	0
0	1	1	1	0	0	0	1
1	0	0	0	X	X	X	X
1	0	0	1	X	X	X	X
1	0	1	0	0	0	0	1
1	0	1	1	0	0	0	0
1	0	0	0	X	X	X	X
1	0	0	1	X	X	X	X
1	0	1	0	X	X	X	X
1	0	1	1	X	X	X	X

Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: **SÍNTESIS** (diseño)

Otro ejemplo de Diseño: Diseñar un cambiador de código serie que pase de código binario natural de tres bits a código Gray de tres bits. Los bits se reciben de mayor a menor peso. Una vez recibido el tercer bit, comienza la recepción del siguiente dato binario. Realizarlo de tal forma que en todo momento haga la conversión, es decir que en el momento en que cambie la entrada, debe cambiar la salida.

Q2	Q1	Q0	IN	Q2*	Q1*	Q0*	OUT
0	0	0	0	0	0	1	0
0	0	0	1	0	1	0	1
0	0	1	0	0	1	1	0
0	0	1	1	1	0	1	1
0	1	0	0	0	1	1	1
0	1	0	1	1	0	1	0
0	1	1	0	0	0	0	0
0	1	1	1	0	0	0	1
1	0	0	0	X	X	X	X
1	0	0	1	X	X	X	X
1	0	1	0	0	0	0	1
1	0	1	1	0	0	0	0
1	0	0	0	X	X	X	X
1	0	0	1	X	X	X	X
1	0	1	0	X	X	X	X
1	0	1	1	X	X	X	X

Q0,IN
Q2,Q1

	00	01	11	10
00	0	0	1	0
01	0	1	0	0
11	X	X	X	X
10	X	X	0	0

Q2*

Q0,IN
Q2,Q1

	00	01	11	10
00	0	1	0	1
01	1	0	0	0
11	X	X	X	X
10	X	X	0	0

Q1*

Q0,IN
Q2,Q1

	00	01	11	10
00	1	0	1	1
01	1	1	0	0
11	X	X	X	X
10	X	X	0	0

Q0*

Q0,IN
Q2,Q1

	00	01	11	10
00	0	1	1	0
01	1	0	1	0
11	X	X	X	X
10	X	X	0	1

OUT

Sistemas Secuenciales Síncronos

Máquinas de estados síncronas: **SÍNTESIS** (diseño)

Otro ejemplo de Diseño: Diseñar un cambiador de código serie que pase de código binario natural de tres bits a código Gray de tres bits. Los bits se reciben de mayor a menor peso. Una vez recibido el tercer bit, comienza la recepción del siguiente dato binario. Realizarlo de tal forma que en todo momento haga la conversión, es decir que en el momento en que cambie la entrada, debe cambiar la salida.

