

- **Definiciones básicas**

 - Autómata de Mealy

 - Autómata de Moore

- **Formas de descripción de un sistema secuencial**

 - Diagrama de estado

 - Tabla de flujo

 - Tabla de estado-salida

 - Tabla de transición

- **Proceso de análisis de sistemas secuenciales**

- **Problemas estructurales en circuitos asíncronos. Circuitos síncronos.**

- **Elementos de Memoria**

- **Proceso de Análisis y síntesis de circuitos con elementos de memoria**

- **Reducción de tablas de estado**

 - Obtención de pares compatibles mediante la carta de implicación

 - Obtención del conjunto compatible máximo

- **Reglas de asignación de variables de estado**

Definiciones básicas

- **Definiciones básicas.**

En un circuito secuencial, el valor de la salida en un instante de tiempo t depende del valor de las entradas, y de la historia previa en la evolución del sistema.

La historia en la evolución de un sistema secuencial, es mantenida por medio de sus *estados internos*.

• Un circuito secuencial está formado por:

- a) Un conjunto finito de n señales de entrada (x_1, x_2, \dots, x_n) , que generan un conjunto finito de 2^n estados de entrada E .

$$E = \{\underline{x}^1, \underline{x}^2, \dots, \underline{x}^{2^n}\} = \{e_1, e_2, \dots, e_{2^n}\} \quad \text{donde } x^i = (x_1, x_2, \dots, x_n)$$

- b) Un conjunto finito de p señales de salida (z_1, z_2, \dots, z_p) , que generan un conjunto finito de 2^p estados de salida S .

$$S = \{\underline{z}^1, \underline{z}^2, \dots, \underline{z}^{2^p}\} = \{s_1, s_2, \dots, s_{2^p}\} \quad \text{donde } z^i = (z_1, z_2, \dots, z_p)$$

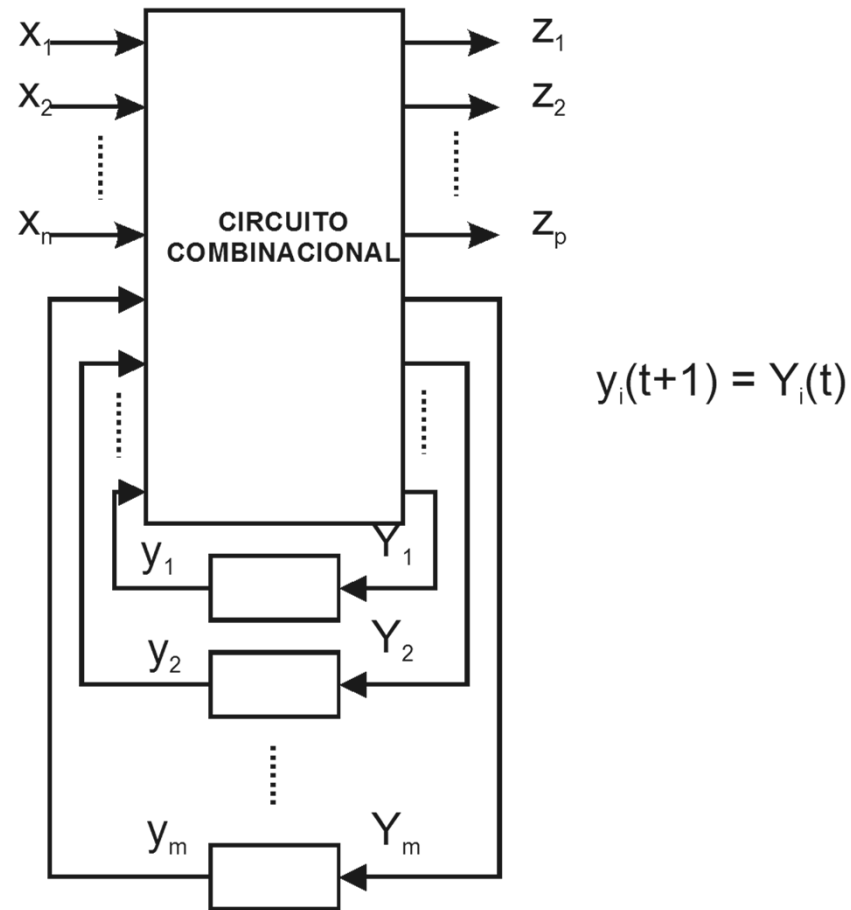
- c) Un conjunto finito de m variables internas (y_1, y_2, \dots, y_m) , que generan un conjunto finito de 2^m estados internos Q .

$$Q = \{\underline{y}^1, \underline{y}^2, \dots, \underline{y}^{2^m}\} = \{q_1, q_2, \dots, q_{2^m}\} \quad \text{donde } y^i = (y_1, y_2, \dots, y_m)$$

- d) Un circuito combinacional.
- e) Elementos de memoria incluidos en los lazos de realimentación.

El valor de $Q_i = (Y_1, Y_2, \dots, Y_m)$ determina el *estado próximo*.

El valor de $q_i = (y_1, y_2, \dots, y_m)$ determina el *estado presente*.



Dependiendo de la evolución de los estados internos, los circuitos secuenciales se dividen en:

Asíncronos. Para una combinación de las variables de entrada, el sistema evoluciona sin control externo, hasta que se llega a un estado interno estable.

Síncronos. El sistema es controlado mediante impulsos externos (reloj). El estado interno evoluciona sólo cuando se aplica el impulso.

Se denomina máquina secuencial, a la operación que produce un determinado sistema secuencial.

Autómatas

- **Autómata de Mealy**

- a) Alfabeto de entrada: $E = \{\underline{x}^1, \underline{x}^2, \dots, \underline{x}^{2^n}\} = \{e_1, e_2, \dots, e_{2^n}\}$
- b) Alfabeto de estados internos: $Q = \{\underline{y}^1, \underline{y}^2, \dots, \underline{y}^{2^m}\} = \{q_1, q_2, \dots, q_{2^m}\}$
- c) Alfabeto de salida: $S = \{\underline{z}^1, \underline{z}^2, \dots, \underline{z}^{2^p}\} = \{s_1, s_2, \dots, s_{2^p}\}$
- d) Función de salida: $f: E \times Q \rightarrow S$
- e) Función de transición: $t: E \times Q \rightarrow Q$
- f) Por tanto para una entrada e_i y un estado interno q_i tendremos que:

$$f(e_i, q_i) = s_i$$
$$t(e_i, q_i) = q_m$$

- **Autómata de Moore**

- a) Alfabeto de entrada: $E = \{ \underline{x}^1, \underline{x}^2, \dots, \underline{x}^{2^n} \} = \{ e_1, e_2, \dots, e_{2^n} \}$
- b) Alfabeto de estados internos: $Q = \{ \underline{y}^1, \underline{y}^2, \dots, \underline{y}^{2^m} \} = \{ q_1, q_2, \dots, q_{2^m} \}$
- c) Alfabeto de salida: $S = \{ \underline{z}^1, \underline{z}^2, \dots, \underline{z}^{2^p} \} = \{ s_1, s_2, \dots, s_{2^p} \}$
- d) Función de salida: $f: Q \rightarrow S$
- e) Función de transición: $t: E \times Q \rightarrow Q$
- f) Por tanto para una entrada e_i y un estado interno q_i tendremos que:

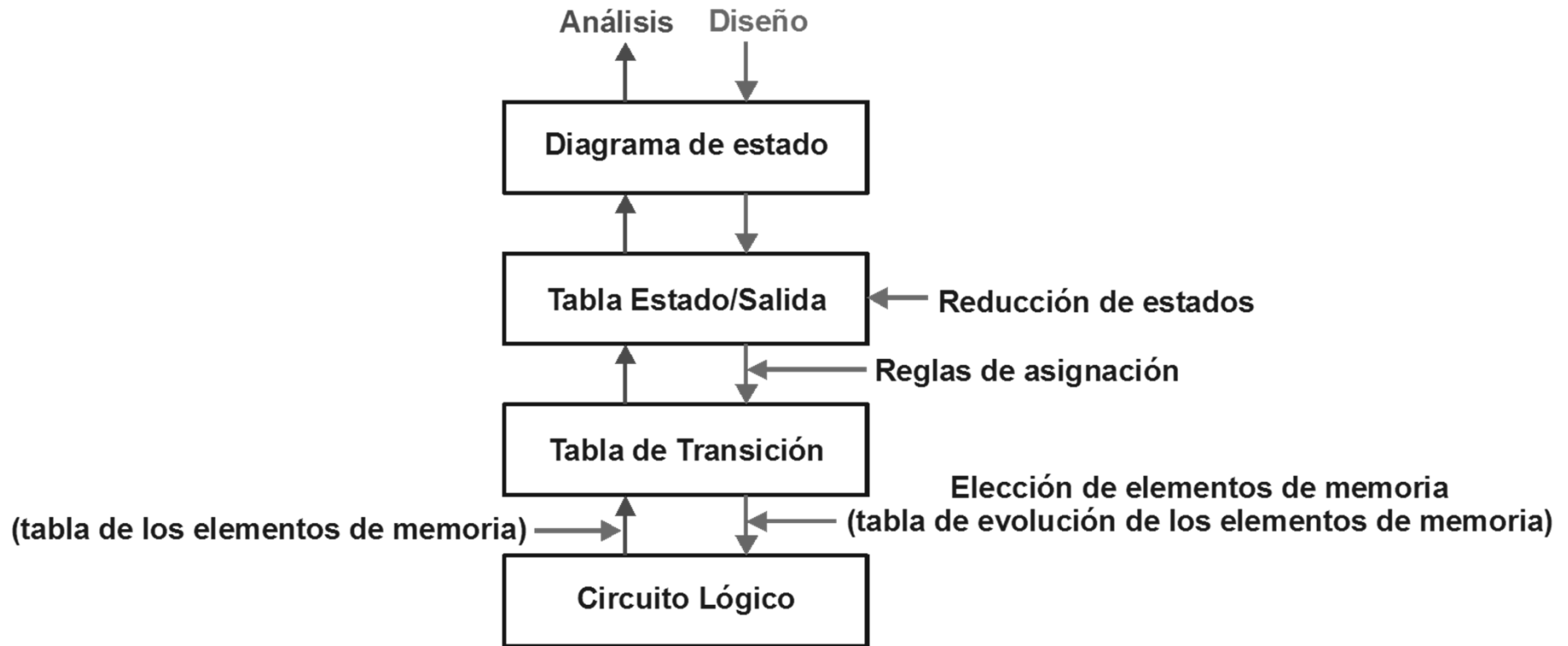
$$f(q_i) = s_i$$
$$t(e_i, q_i) = q_m$$

- **Importante**

Todo sistema secuencial admite una representación mediante un **autómata de Mealy** y una representación mediante un **autómata de Moore**

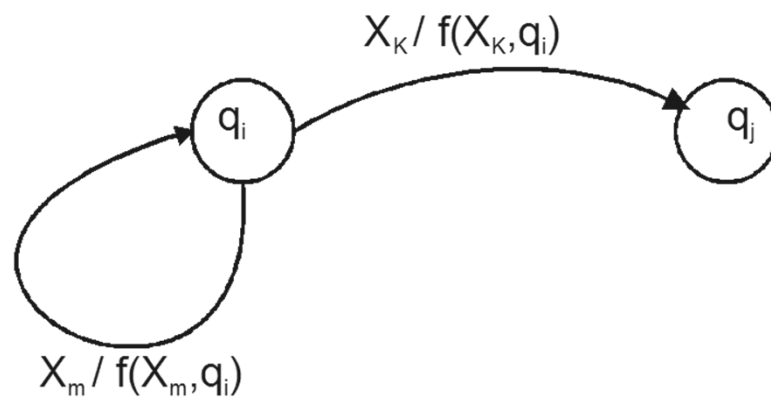
Formas de descripción de un sistema secuencial

- Formas de representación en el proceso de diseño y análisis de Circuitos Secuenciales

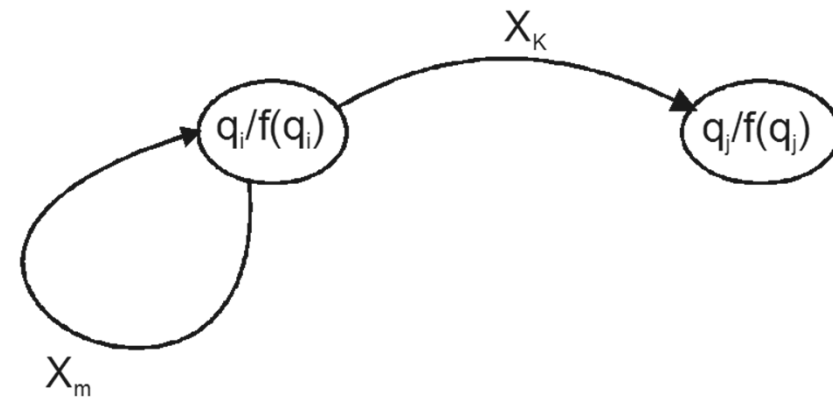


- Diagrama de estado

El *diagrama de estados* (también llamado *diagrama de transición de estados*) está constituido por un conjunto de nudos ó vértices que se corresponden a los estados y una serie de arcos dirigidos que indican las transiciones a los próximos estados, en función de las entradas y el estado presente.



Máquina de Mealy



Máquina de Moore

- **Tipos de estados**

- a) *Estado terminal*

- Estado en el que se permanece independientemente de la secuencia que se de en las entradas.

- b) *Estado inaccesible*

- Estado al que no se puede llegar desde ningún estado, aunque se aplique cualquier secuencia en las variables de entrada.

- c) *Estado estable* (para una combinación dada en las entradas)

- Estado en el que su próximo estado coincide con él mismo.

- d) *Estado inestable* (para una combinación dada en las entradas)

- Estado en el que su próximo estado NO coincide con él mismo.

- **Tabla de estado-salida**

Es una representación tabular de la función de estado (*próximo estado*) y de la función de salida (*valor de la salida*) en función del valor de las *entradas* y el *estado presente*.

En las columnas se representa el alfabeto de entrada x_i , y en las filas los estados presentes q_i .

- Tabla estado-salida para una máquina Mealy

		E		
		x_0	x_1
Q	q_0	$t(x_0, q_0), f(x_0, q_0)$	$t(x_1, q_0), f(x_1, q_0)$
	q_1	$t(x_0, q_1), f(x_0, q_1)$	$t(x_1, q_1), f(x_1, q_1)$

	q_{m-1}	$t(x_0, q_{m-1}), f(x_0, q_{m-1})$	$t(x_1, q_{m-1}), f(x_1, q_{m-1})$

- Tabla estado-salida para una máquina Moore

		E				S
		x_0	x_1	x_{n-1}	
Q	q_0	$t(x_0, q_0)$	$t(x_1, q_0)$	$t(x_{n-1}, q_0)$	$f(q_0)$
	q_1	$t(x_0, q_1)$	$t(x_1, q_1)$	$t(x_{n-1}, q_1)$	$f(q_1)$

	q_{m-1}	$t(x_0, q_{m-1})$	$t(x_1, q_{m-1})$	$t(x_{n-1}, q_{m-1})$	$f(q_{m-1})$

- **Tabla de flujo**

Es una representación tabular del *próximo estado* en función del valor de las *entradas* y de el *estado presente*, pero en la que sólo figuran los *estados estables*. Los estados inestables son sustituidos por el estado estable al que se transitará.

En las columnas se representa el alfabeto de entrada x_i , y en las filas los estados presentes q_i .

- **Tabla de transición.**

Es una representación tabular de los *valores próximos* de las variables de estado en función del valor de las *entradas* y del *valor presente* de las variables de estado.

En las columnas se representa el alfabeto de entrada x_i , y en las filas los *valores presentes* de las variables de estado.

Los estados estables, se marcan rodeándolos con un círculo.

	$x_1, x_2, \dots, x_i, \dots, x_n$
$y_1, y_2, \dots, y_i, \dots, y_m$	$Y_1, Y_2, \dots, Y_i, \dots, Y_m$

Proceso de análisis de sistemas secuenciales

Una característica común a todos los circuitos secuenciales, es la existencia de lazos ó bucles de realimentación, pero su presencia no es una condición suficiente para ser un circuito secuencial

Los lazos de realimentación pueden aparecer de forma explícita en el circuito o dentro de los elementos de memoria (se verán más adelante)

El proceso de análisis consta de los siguientes pasos

- a) Verificar que es un circuito secuencial.
- b) Romper los lazos de realimentación, insertando en cada uno de ellos un elemento de memoria.

Tendremos tantas variables de estado como rupturas efectuemos.

- c) Obtener las expresiones algebraicas de la función de transición y de la función de salida.
- d) Construimos la tabla de transición
- e) Construimos la tabla de estado-salida
- f) Construimos el diagrama de estado

Problemas estructurales en circuitos asíncronos. Circuitos síncronos

Cuando se produce un cambio en las entradas, la respuesta del circuito puede ser distinta a la esperada, debido a que no todas las variables de estado cambian simultáneamente

a) Azar extrínseco

La diferencia de los tiempos de propagación de la señal es debida a que las puertas con las que está construido el circuito tienen distintos tiempos de propagación

b) Azar intrínseco

La diferencia de los tiempos de propagación de la señal es debida a distinta longitud de los caminos de propagación de las señales

c) Carrera

Una situación no determinista motivada por el cambio de varias variables de estado

- **Tipos de carreras**

- a) Carrera crítica

- Cuando se llega a un estado estable no previsto, debido a una carrera

- b) Carrera no crítica

- Cuando se llega al mismo estado estable, independientemente del camino que se siga, debido a una carrera

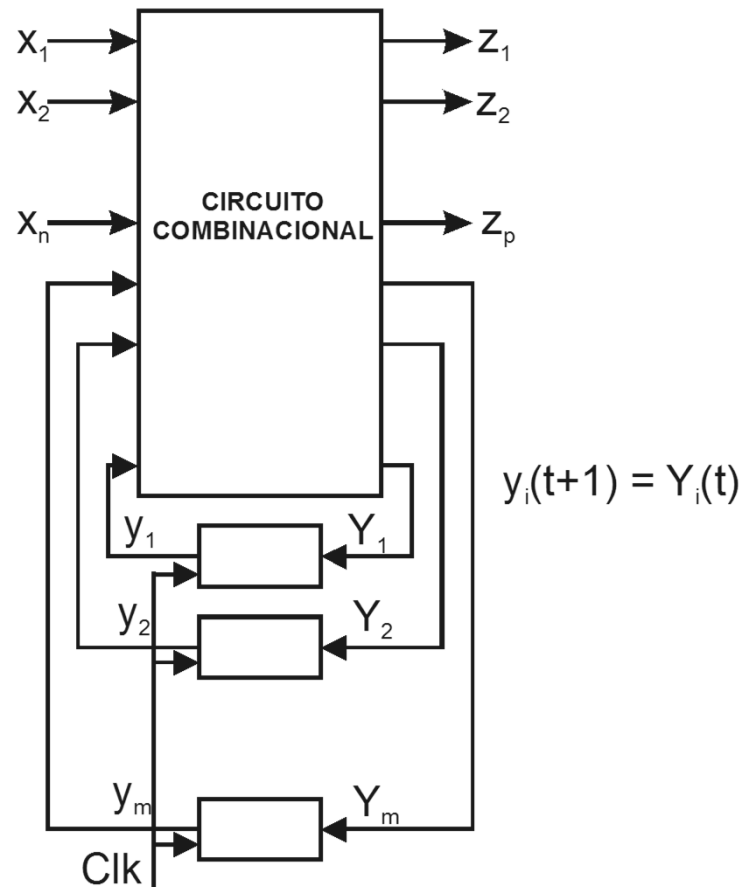
La existencia de dos ó más estados estables en una misma columna de la tabla de transición, es condición necesaria, pero no suficiente, para la existencia de carreras críticas

Otro posible problema es cuando para una determinada combinación de las variables de entrada, no hay ningún estado estable, por lo que tendríamos una máquina secuencial no determinista, y sería imposible predecir el estado al que tiende la máquina para dicha combinación de entrada

Circuitos secuenciales síncronos

Están libres de todos los problemas que presentan los circuitos secuenciales asíncronos, para lo cual presentan las siguientes restricciones:

- a) Cuando las entradas cambian, los lazos de realimentación están abiertos
- b) Cuando las entradas son estables, se cierran los lazos y se producen los cambios de estado
- c) Durante el proceso de apertura y cierre de los lazos de realimentación, se mantiene memoria del estado presente
- d) Sólo se permite un cambio de estado en cada apertura y cierre de los lazos de realimentación
- e) Los lazos se abren y cierran de acuerdo a una señal de control, generalmente periódica (reloj)



Como consecuencia, NO hay estados inestables, ya que cada próximo estado, será el estado siguiente al que se llegará, ya que sólo se permite una transición por cada pulso de reloj

Elementos de Memoria

Biestable SR

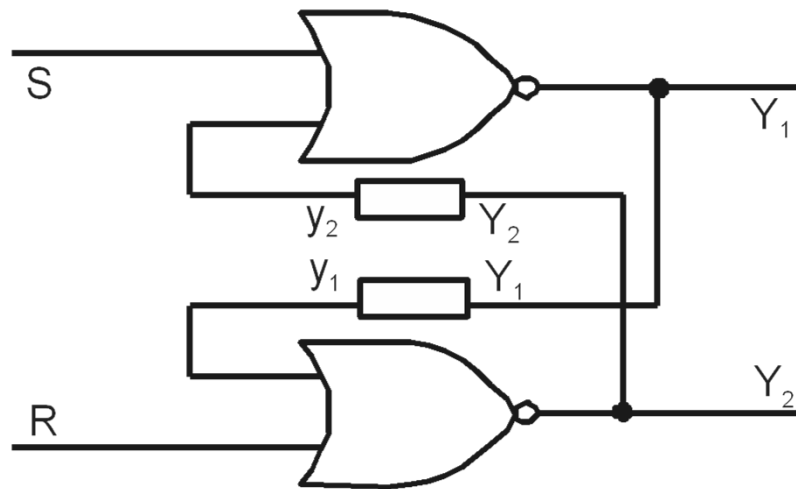


Diagrama lógico

		SR			
		00	01	11	10
y_1y_2	00	11	10	00	01
	01	01	00	00	01
	11	00	00	00	00
	10	10	10	00	00
		Y_1Y_2			

Tabla de transición

Carreras críticas para SR=00

$$00 \rightarrow 11 \Rightarrow \begin{cases} 00 \rightarrow 01 \\ 00 \rightarrow 10 \end{cases} \quad 11 \rightarrow 00 \Rightarrow \begin{cases} 11 \rightarrow 10 \\ 11 \rightarrow 01 \end{cases}$$

Prohibiendo la entrada SR=11, no podremos estar nunca en los estados $y_1y_2 = 00$, $y_1y_2 = 11$ con lo que eliminaremos dichas carreras críticas.

Posibles estados accesibles:

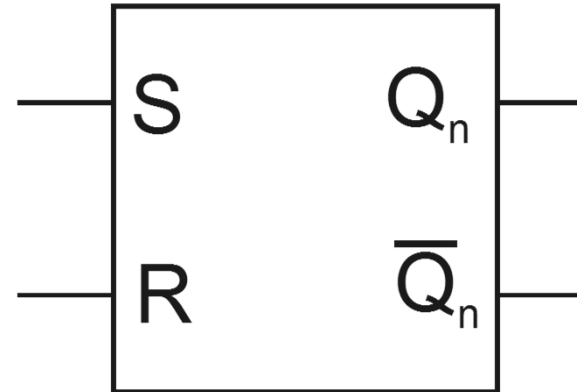
$$y_1y_2 = 01$$

$$y_1y_2 = 10$$

Tabla de transición simplificada

S	R	Y_2
0	0	y_2
0	1	0
1	0	1
1	1	no permitido

$$\begin{aligned}\overline{Q}_{n+1} &= Y_1 \\ Q_{n+1} &= Y_2 \\ \overline{Q}_n &= y_1 \\ Q_n &= y_2\end{aligned}$$



S	R	Q_{n+1}
0	0	Q_n
0	1	0
1	0	1
1	1	no permitido

Tabla del biestable SR

$Q_n \rightarrow Q_{n+1}$	S	R
$0 \rightarrow 0$	0	-
$0 \rightarrow 1$	1	0
$1 \rightarrow 0$	0	1
$1 \rightarrow 1$	-	0

Tabla de evolución del biestable SR

Biestable T

T	Q_{n+1}
0	Q_n
1	\bar{Q}_n

Tabla del biestable T

$Q_n \rightarrow Q_{n+1}$	T
$0 \rightarrow 0$	0
$0 \rightarrow 1$	1
$1 \rightarrow 0$	1
$1 \rightarrow 1$	0

Tabla de evolución del biestable T

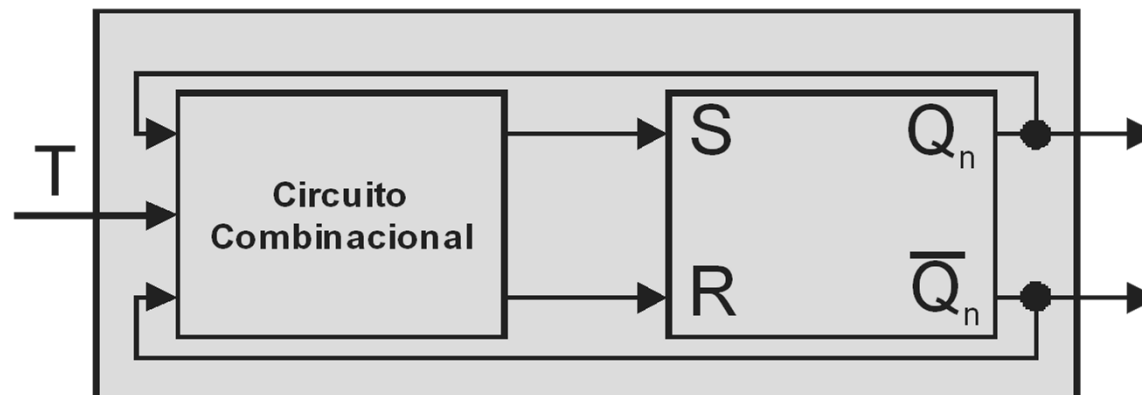


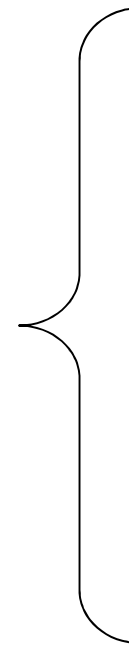
Tabla de Transición

		T	
		0	1
Q_n	0	0	1
	1	1	0

Q_{n+1}

		T	
		0	1
Q_n	0	0 -	1 0
	1	- 0	0 1

S R



		T	
		0	1
Q_n	0	0	1
	1	-	0

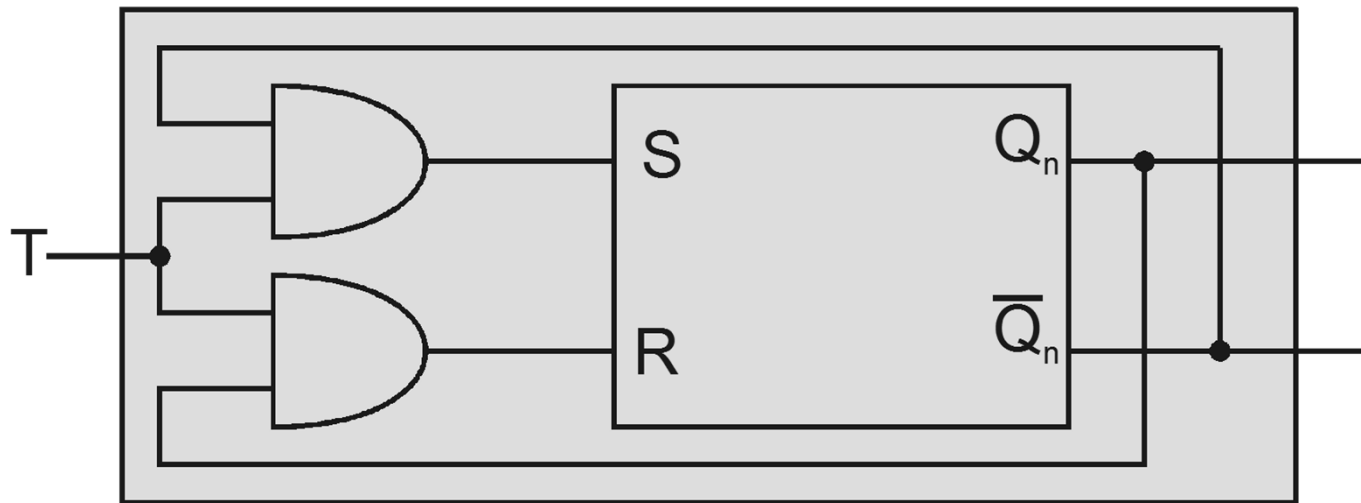
S

		T	
		0	1
Q_n	0	-	0
	1	0	1

R

$$S = T\bar{Q}_n$$

$$R = TQ_n$$



Biestable JK

J K	Q_{n+1}
0 0	Q_n
0 1	0
1 0	1
1 1	\bar{Q}_n

Tabla del biestable JK

$Q_n \rightarrow Q_{n+1}$	JK
$0 \rightarrow 0$	0 -
$0 \rightarrow 1$	1 -
$1 \rightarrow 0$	- 1
$1 \rightarrow 1$	- 0

Tabla de evolución del biestable JK

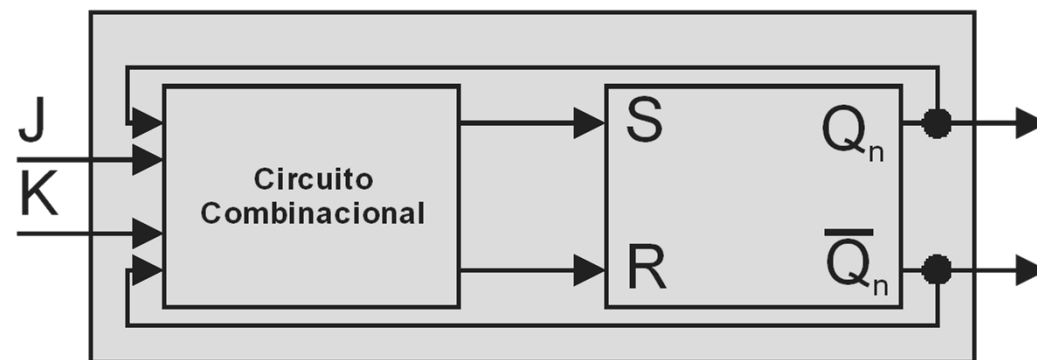


Tabla de transición

		JK			
		00	01	11	10
Q_n	0	0	0	1	1
	1	1	0	0	1

Q_{n+1}

		JK			
		00	01	11	10
Q_n	0	0 -	0 -	1 0	1 0
	1	- 0	0 1	0 1	- 0

SR

		JK			
		00	01	11	10
Q_n	0	0	0	1	1
	1	-	0	0	-

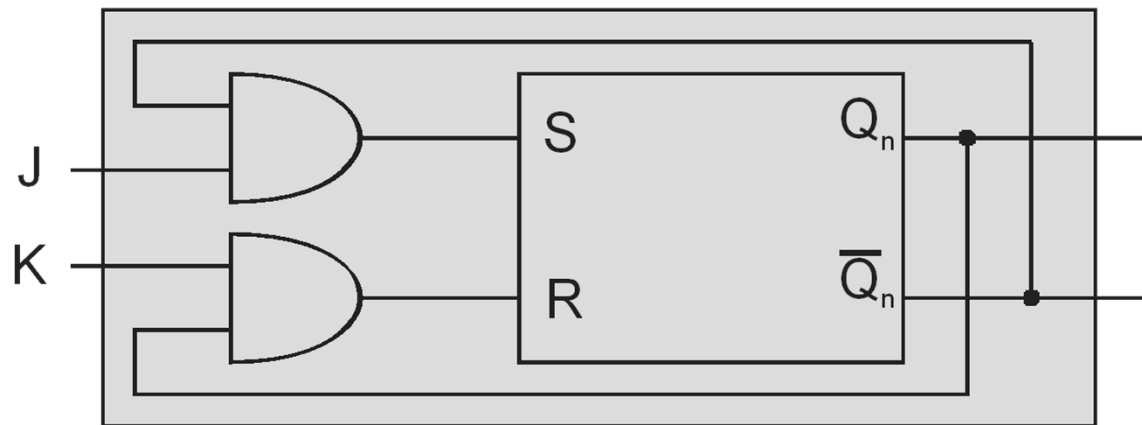
S

$S = J\bar{Q}_n$

		JK			
		00	01	11	10
Q_n	0	-	-	0	0
	1	0	1	1	0

R

$R = KQ_n$



Biestable D

D	Q_{n+1}
0	0
1	1

Tabla del biestable D

$Q_n \rightarrow Q_{n+1}$	D
$0 \rightarrow 0$	0
$0 \rightarrow 1$	1
$1 \rightarrow 0$	0
$1 \rightarrow 1$	1

Tabla de evolución del biestable D

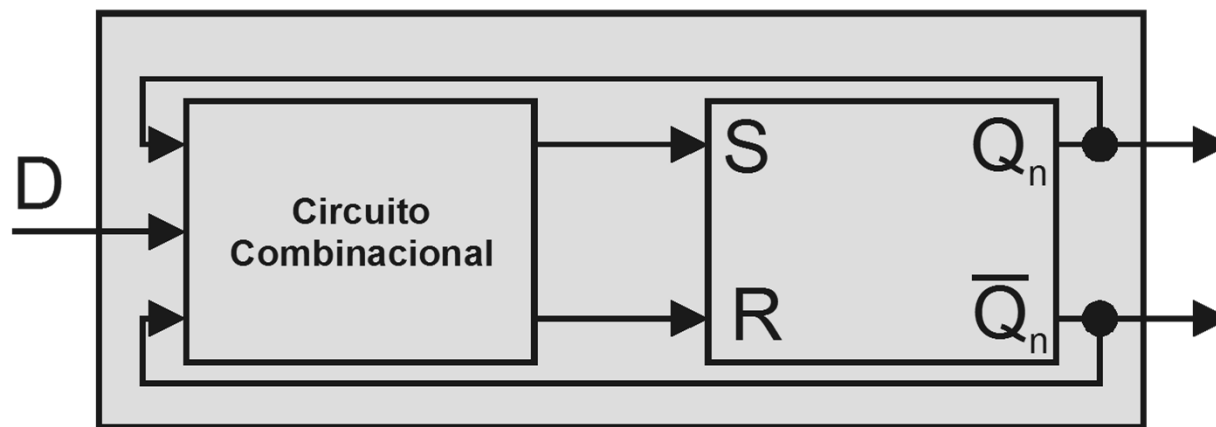


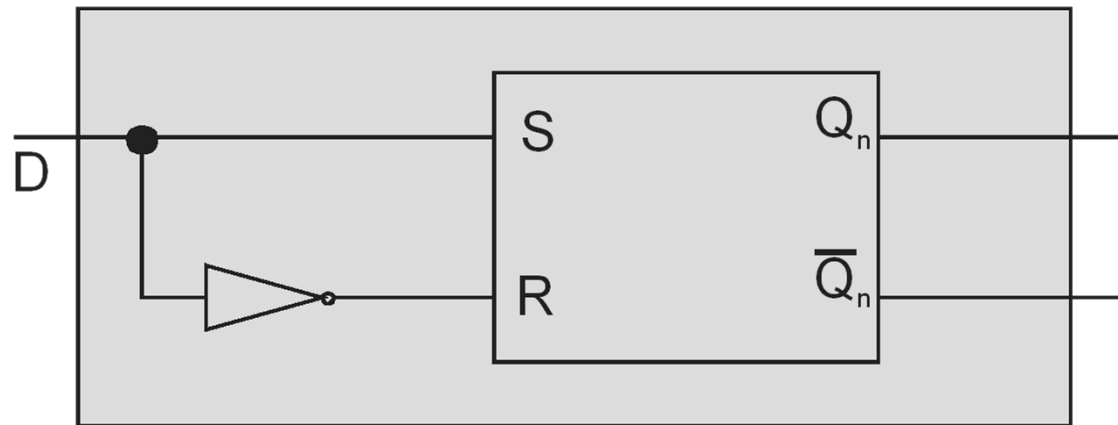
Tabla de transición

		D			
		0	1		
Q_n	0	0	1		
	1	0	1		
		Q_{n+1}			

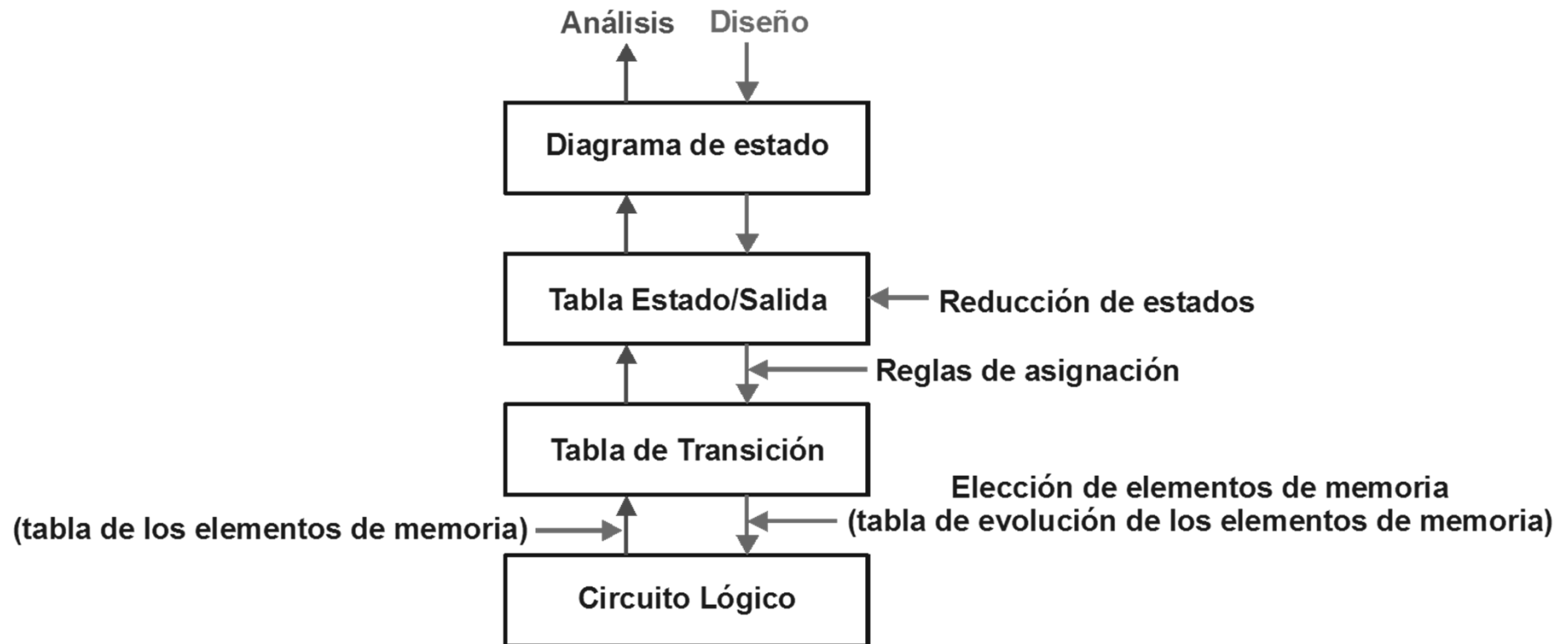
		D			
		0	1		
Q_n	0	0 -	1 0		
	1	0 1	- 0		
		S R			

		D			
		0	1		
Q_n	0	0	1		
	1	0	-		
		S		$S = D$	

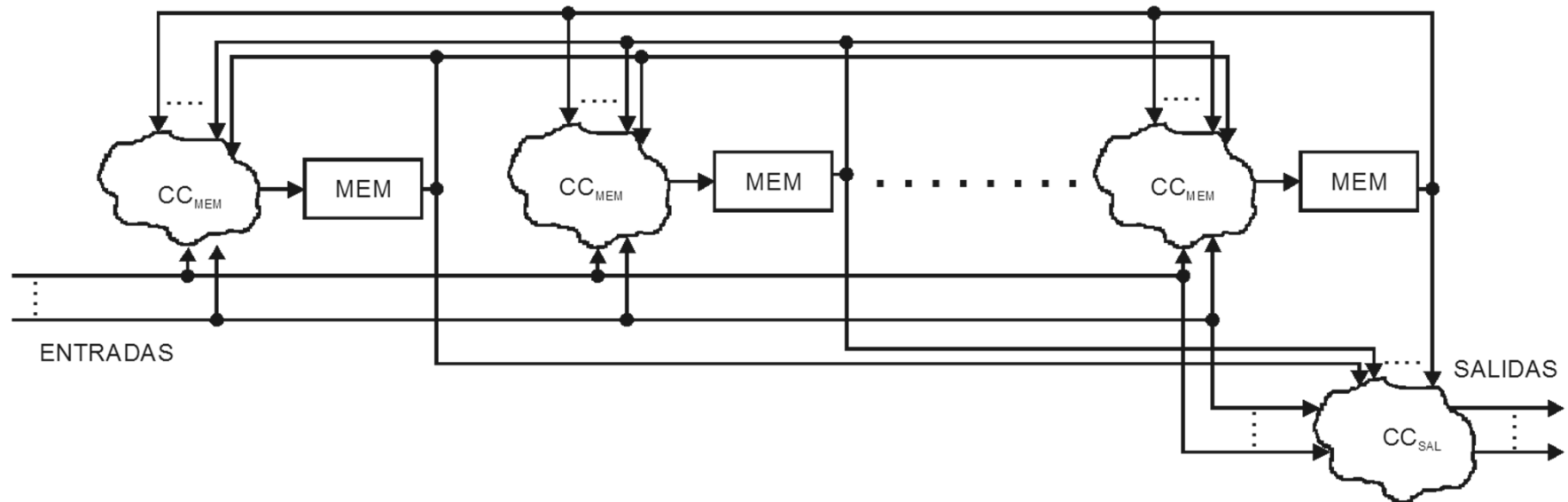
		D			
		0	1		
Q_n	0	-	0		
	1	1	0		
		R		$R = \bar{D}$	



Proceso de Análisis y síntesis de circuitos con elementos de memoria



- Estructura general de un circuito secuencial construido con elementos de memoria



Reducción de Tablas de Estado

- **Obtención de pares compatibles mediante la carta de implicación**

En general, una reducción de la tabla de estado, de un sistema secuencial, conduce a una reducción del coste lógico en su implementación

DEF: Un estado S_A de una tabla de estados A , se dice que *cubre a un estado* S_B de una tabla de estados B , si para cualquier secuencia de entrada finita comenzando en el estado S_A de la tabla A , obtiene la misma salida que la tabla B , comenzando en el estado S_B , siempre que dicha salida esté especificada

DEF: Una tabla de estados A se dice que *cubre a una tabla* de estados B , si todo estado de B es cubierto por al menos un estado de A

PROP: Si una tabla A cubre a otra tabla B , la implementación de la tabla A puede efectuar todas las operaciones de la implementación de la tabla B

DEF: Dos estados de una tabla de estados A, se dicen *compatibles*, si ambos estados pueden ser cubiertos por un único estado de una tabla de estados B

PROP: Dos estados de una tabla de estados serán *compatibles*, si:

- a) Para cualquier entrada producen la misma salida.
- b) Los próximos estados de ambos, son los mismos ó son compatibles

El procedimiento sistemático para encontrar los pares de estados compatibles se realiza mediante la *Carta de Implicación*

Ejemplo de simplificación de tabla de estados.

	I_1	I_2	I_3	I_4
1	1,1	6,1	1,1	2,0
2	3,1	2,0	2,1	2,0
3	3,1	5,1	3,1	2,0
4	4,1	6,1	4,1	2,0
5	1,1	5,1	4,1	5,0
6	1,1	6,1	3,1	6,0

Tabla de Estado/Salida

2	X *				
3	5-6	X *			
4		X *	5-6		
5	2-5 X 5-6 * 1-4	X *	1-3 X 3-4 * 2-5	1-4 X 5-6 * 2-5	
6	1-3 X 2-6 *	X *	1-3 X 5-6 * 2-6	1-4 X 3-4 * 2-6	3-4
	1	2	3	4	5

Pares de estados compatibles. { (1,3) , (1,4) , (3,4) , (5,6) }

- **Obtención del conjunto compatible máximo**

DEF: Conjunto compatible máximo es aquel conjunto de estados compatibles, que no es subconjunto de otro conjunto de estados compatibles. Se denota por MCS

Si construimos un tabla usando el MCS, normalmente se conseguirá una reducción de estados

Iniciar el proceso con los conjuntos

$C = \{ n \}$ donde n es el número de estados de la tabla.

$I = \{ \}$

Para todas las columnas, proceder desde la derecha hacia la izquierda con el siguiente proceso. Dada la columna i .

a) Formar el conjunto S_i de todos los estados cuyas entradas en dicha columna no tienen X

b) Formar los conjuntos:

$$I = \{ (S_i \cap C) \cup i \}$$

$$C = \{ C \cup I \}$$

c) Eliminar de C cualquier elemento duplicado ó contenido en otro

Cuando se termina el proceso con todas las columnas, el conjunto final C es el MCS

i	S_i	I	C	C (simplificado)
		\emptyset	6	6
5	6	56	56 , 6	56
4	\emptyset	4	4 , 56	4 , 56
3	4	34	34 , 4 , 56	34 , 56
2	\emptyset	2	2 , 34 , 56	2 , 34 , 56
1	34	134	134 , 2 , 34 , 56	134 , 2 , 56

$$MCS = \{ 134 , 56 , 2 \}$$

Renombramos los estados: $A = 134$ $B = 56$ $C = 2$

	I_1	I_2	I_3	I_4
A	A,1	B,1	A,1	C,0
B	A,1	B,1	A,1	B,0
C	A,1	C,0	C,1	C,0

Reglas de asignación de variables de estado

DEF: El asignamiento de variables de estado para dos estados se dice *adyacente*, si dicho asignamiento, provoca que las variables de estado asignadas difieran solamente en una

El objetivo en la asignación de estados para circuitos secuenciales síncronos, es la obtención de un coste mínimo en el circuito combinacional adicional a los elementos de memoria

Las reglas se deben aplicar en el orden jerárquico:

$$1.a \rightarrow 1.b \rightarrow 1.c \rightarrow 2 \rightarrow 3.$$

REGLA 1

- a) Los estados que tienen idénticos próximos estados, para todas las entradas (en las mismas entradas), deben tener asignamientos adyacentes
- b) Los estados que tienen idénticos próximos estados, para todas las entradas (en distintas entradas), deben tener asignamientos adyacentes si dichos próximos estados pueden ser dados como asignamientos adyacentes
- c) Los estados que tienen idénticos próximos estados, para algunas entradas, deben tener asignamientos adyacentes. Los estados con más columnas idénticas tendrán mayor prioridad

REGLA 2

Los próximos estados de un mismo estado, deberán tener asignamientos adyacentes

REGLA 3

Los estados que tengan la misma salida para una cierta entrada deben tener asignamientos adyacentes