



Escuela de Ingenierías Industriales

Titulación: Grado en Ingeniería Electrónica, Robótica y Mecatrónica

Asignatura: Electrónica Digital

Tema 8: Análisis y Síntesis de Sistemas Secuenciales Síncronos



TEMA 8: ANÁLISIS Y SÍNTESIS DE SISTEMAS SECUENCIALES SÍNCRONOS

8.1. Análisis de sistemas secuenciales síncronos.

8.1.1 Introducción. Modelo de Huffman para sistemas síncronos. Variables y ecuaciones booleanas.

8.1.2 Funciones de entrada de los biestables. Funciones de salida.

8.1.3 Tabla de estados: Tablas características de los biestables. Tabla de Transición de estados.

8.1.4 Diagramas de estados. Cronogramas.

8.1.5 Modelos de Mealy y de Moore para los sistemas secuenciales síncronos.

8.2. Diseño de sistemas secuenciales síncronos.

8.2.1 Tabla de excitación de los biestables.

8.2.2 Reducción de estados: Tablas de implicación. Estados equivalentes y estados compatibles.

8.2.3 Asignación de estados.

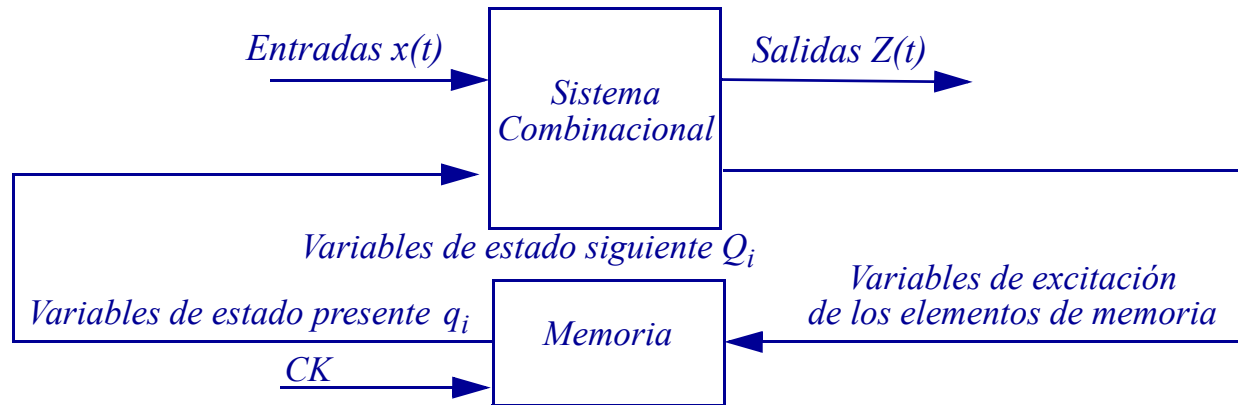
8.2.4 Ejemplos de diseño.

8.2.5

8.3. Diagramas de estados en HDL.

Análisis de Sistemas digitales secuenciales síncronos

- El análisis de un sistema digital secuencial síncrono se inicia identificando, sus entradas y salidas, así como el número y tipo de elementos de memoria que contiene, todo ello siguiendo el modelo de Huffman.



La salida de cada elemento de memoria está asociada a una variable binaria ($q_i(t)$).

En conjunto reciben el nombre de **variables de estado presente**, y codifican el **estado del sistema**.

La entradas de los elementos de memoria se denominan **variables de excitación de los elementos de memoria**, las cuales determinan el valor de las **variables de estado siguiente** (Q_i).

La **tabla de comportamiento** de los elementos de memoria establece la relación entre ambas variables, esto es, indica el estado al que evoluciona el elemento de memoria según el valor de su entrada.

La transición de estados se realiza cuando se produce el instante activo de la señal de reloj **CK**.

Un sistema con **n** elementos de memoria puede encontrarse, en un instante dado, en uno de los 2^n estados distintos posibles.

Análisis de Sistemas digitales secuenciales síncronos

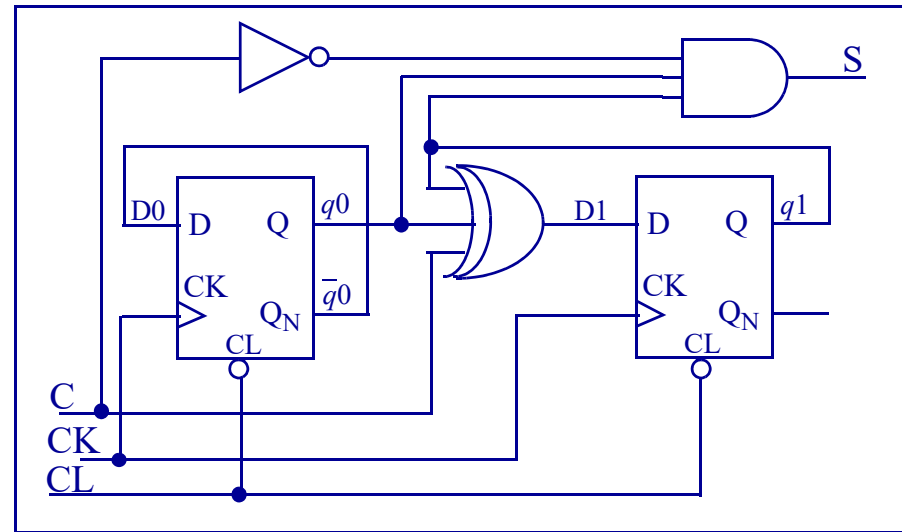
- La tarea de **análisis** consiste en **determinar el comportamiento del sistema** estableciendo, para cualquier estado presente, cuál es su salida y cuál el estado siguiente al que se evoluciona como respuesta a cualquiera de las posibles combinaciones de las variables de entrada.

El **comportamiento** de un sistema digital secuencial síncrono se describe:

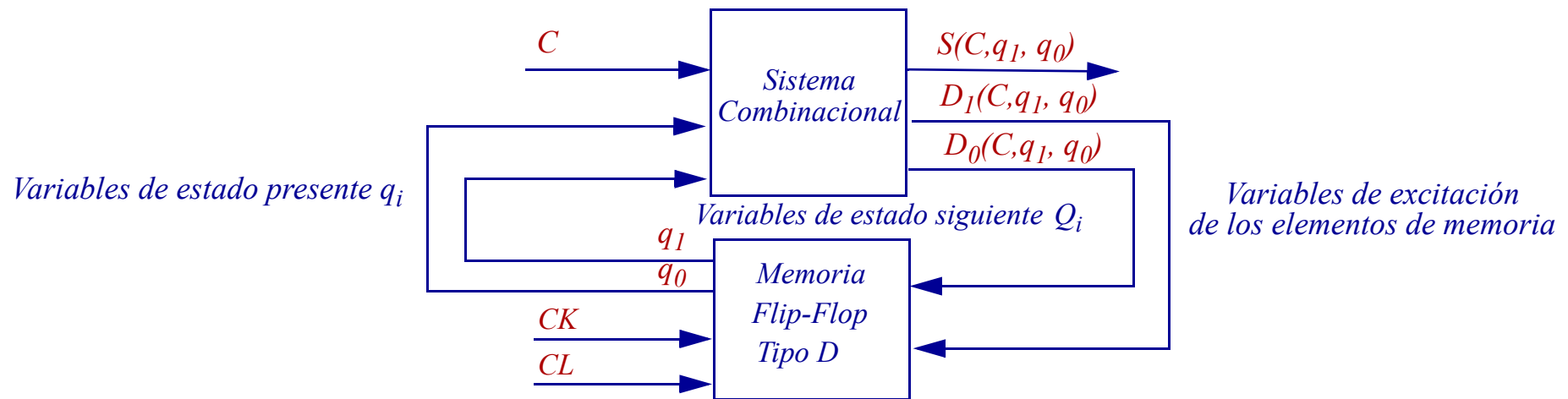
- a) Especificando dos **conjuntos de funciones booleanas**, que dependen, en general, de las entradas y de las variables de estado presente:
 - Las funciones que definen el comportamiento de las salidas, **funciones booleanas de salida**.
 - Las funciones que definen el estado siguiente, **funciones de excitación de los elementos de memoria**.
- b) Construyendo una **tabla de transición de estados**. Tabla que recoge para cada combinación de entrada y estado presente, cual será el estado siguiente y la salida del sistema.
- c) Construyendo un **diagrama de transición de estados**. Grafo orientado en el que se recoge la misma información que aparece en la tabla de transición de estados, pero de una forma gráfica y más intuitiva.
- d) Construyendo un **cronograma**. En el que se representa la misma información que en los instrumentos anteriores aunque de forma gráfica indicando expresamente la evolución temporal del sistema.
En los cronogramas de sistemas síncronos, de forma esquemática, se suele emplear como referencia la señal de reloj.

Análisis de Sistemas digitales secuenciales síncronos. Máquinas de estados

- Ejemplo1: Determina el comportamiento del siguiente sistema secuencial.



a) Identificación del sistema: Entradas, salidas, elementos de memoria que lo constituyen y variables de estado.

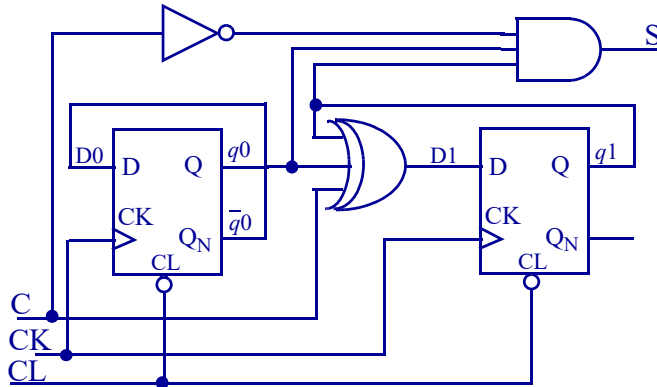


La Tabla de comportamiento de los flip flops D determinan el estado siguiente Q_i a partir de q_i y D_1D_0

Análisis de Sistemas digitales secuenciales síncronos. Máquinas de estados

● Ejemplo1: Continuación

b) *Funciones booleanas de salida y de excitación de los elementos de memoria.*



Función booleanas de salida

$$S(C, q_1, q_0) = q_1 \cdot q_0 \cdot \overline{C}$$

Función excitación de los FF

$$D_1(C, q_1, q_0) = q_1 \oplus q_0 \oplus C$$

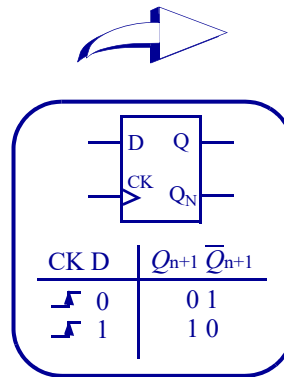
$$D_0(C, q_1, q_0) = \overline{q_0}$$

Cada flanco de subida $q_0 = D_0$ y $q_1 = D_1$

La activación de CL hace $q_0 = 0$ y $q_1 = 0$

c) *Descripción del comportamiento por medio de la tabla de transición de estados.*

Entrada C	Estado presente $q_1 q_0$	Entradas de FF D1 D0	Salida S
0	00	01	0
0	01	10	0
0	10	11	0
0	11	00	1
1	00	11	0
1	01	00	0
1	10	01	0
1	11	10	0



Entrada C	Estado presente $q_1 q_0$	Estado siguiente $Q_1 Q_0$	Entradas de FF D1 D0	Salida S
0	00	01	01	0
0	01	10	10	0
0	10	11	11	0
0	11	00	00	1
1	00	11	11	0
1	01	00	00	0
1	10	01	01	0
1	11	10	10	0

Análisis de Sistemas digitales secuenciales síncronos. Máquinas de estados

● Ejemplo1: Continuación

Función booleanas de salida

$$S(C, q_1, q_0) = q_1 \cdot q_0 \cdot \bar{C}$$

Función excitación de los FF

$$D_1(C, q_1, q_0) = q_1 \oplus q_0 \oplus C$$

$$D_0(C, q_1, q_0) = \bar{q}_0$$

Cada flanco de subida $q_0 = D_0$ y $q_1 = D_1$

La activación de CL hace $q_0 = 0$ y $q_1 = 0$

Tabla de transición de estados

Entrada C	Estado presente $q_1 q_0$	Estado siguiente $Q_1 Q_0$	Salida S
0	0 0	0 1	0
0	0 1	1 0	0
0	1 0	1 1	0
0	1 1	0 0	1
1	0 0	1 1	0
1	0 1	0 0	0
1	1 0	0 1	0
1	1 1	1 0	0

Entrada C	Estado presente $q_1 q_0$	Estado siguiente $Q_1 Q_0$	Entradas de FF D1 D0	Salida S
0	0 0	0 1	0 1	0
0	0 1	1 0	1 0	0
0	1 0	1 1	1 1	0
0	1 1	0 0	0 0	1
1	0 0	1 1	1 1	0
1	0 1	0 0	0 0	0
1	1 0	0 1	0 1	0
1	1 1	1 0	1 0	0

Tabla de transición de estados

Ep	Es		Salidas	
	C=0 C=1		C=0 C=1	
			S S	
A	B	D	0	0
B	C	A	0	0
C	D	B	0	0
D	A	C	1	0

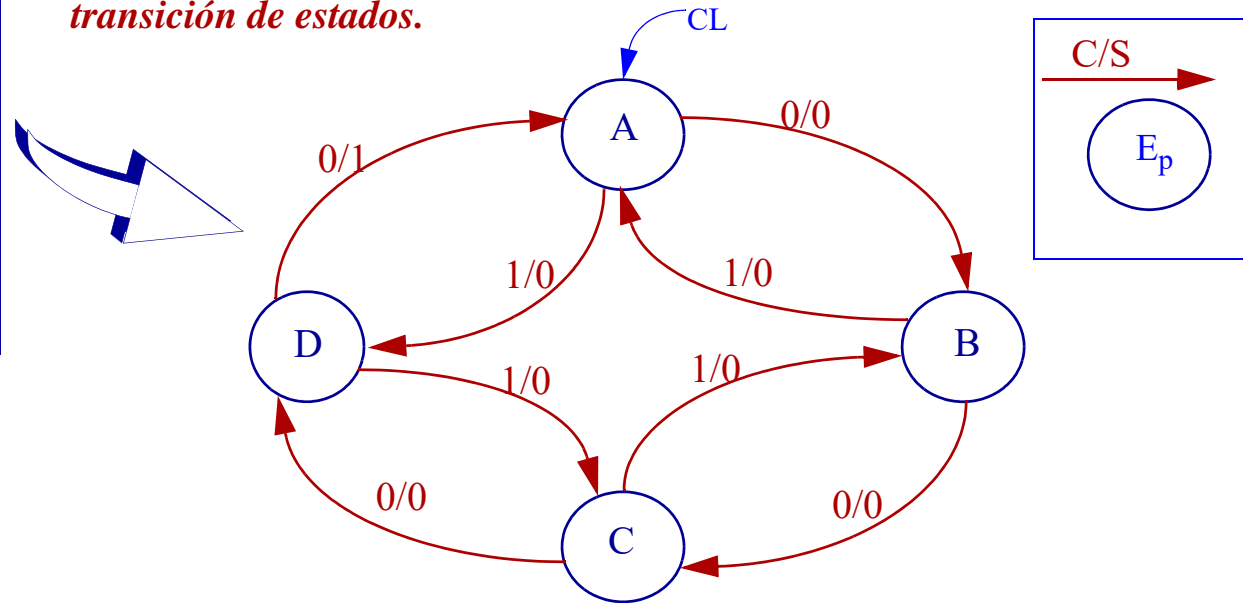
Estados	Códigos $q_1 q_0$
A	0 0
B	0 1
C	1 0
D	1 1

Análisis de Sistemas digitales secuenciales síncronos. Máquinas de estados

● Ejemplo1: Continuación

Ep	Es		Salidas	
	C=0 C=1		C=0 C=1	C=0 C=1
			S	S
A	B	D	0	0
B	C	A	0	0
C	D	B	0	0
D	A	C	1	0

d) Descripción del comportamiento por medio de un diagrama de transición de estados.



El sistema tiene cuatro estados.

Se asume que la entrada CL, entrada clear de los flip-flops, lleva al sistema al estado A ($q_1q_0 = 00$)

independientemente del valor de las entradas C y CK, aunque a veces no se refleje en el diagrama por simplicidad.

Se asume que las transiciones entre estados se realizan de manera síncrona cada flanco de subida del reloj CK.

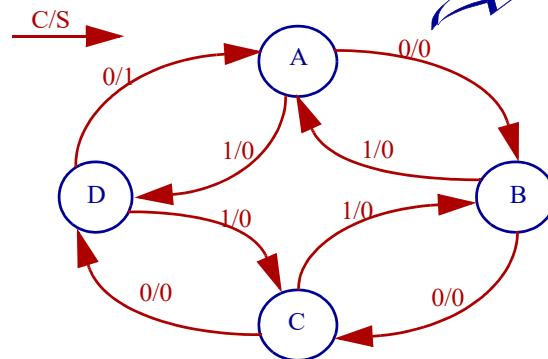
El sistema evoluciona transitando entre estados en sentido de las agujas del reloj si $C=0$, siguiendo la secuencia estados A, B, C, D, A ...; mientras que si $C=1$ la evolución es en sentido contrario a las agujas del reloj, siguiendo la secuencia A, D, C, B, A La secuencia puede invertirse en cualquier momento cambiando el valor de la entrada C

La salida del sistema S sólo toma valor uno si el estado presente es el estado D y la entrada es $C=0$.

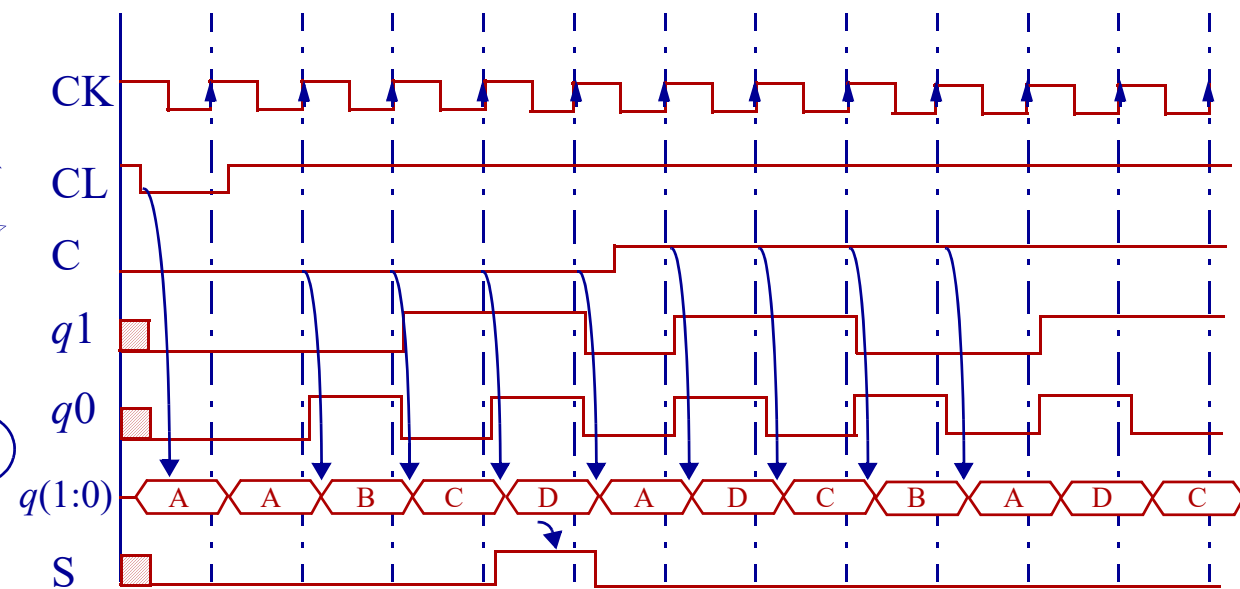
Análisis de Sistemas digitales secuenciales síncronos. Máquinas de estados

● Ejemplo1: Continuación

Ep	Es		Salidas	
	C=0	C=1	C=0 S	C=1 S
A	B	D	0	0
B	C	A	0	0
C	D	B	0	0
D	A	C	1	0



e) Descripción del comportamiento por medio de un cronograma



Tras la activación de la señal de clear ($CL=0$), los elementos de memoria, que inicialmente se marcan como desconocidos, se ponen a cero y por lo tanto el sistema se lleva al estado A.

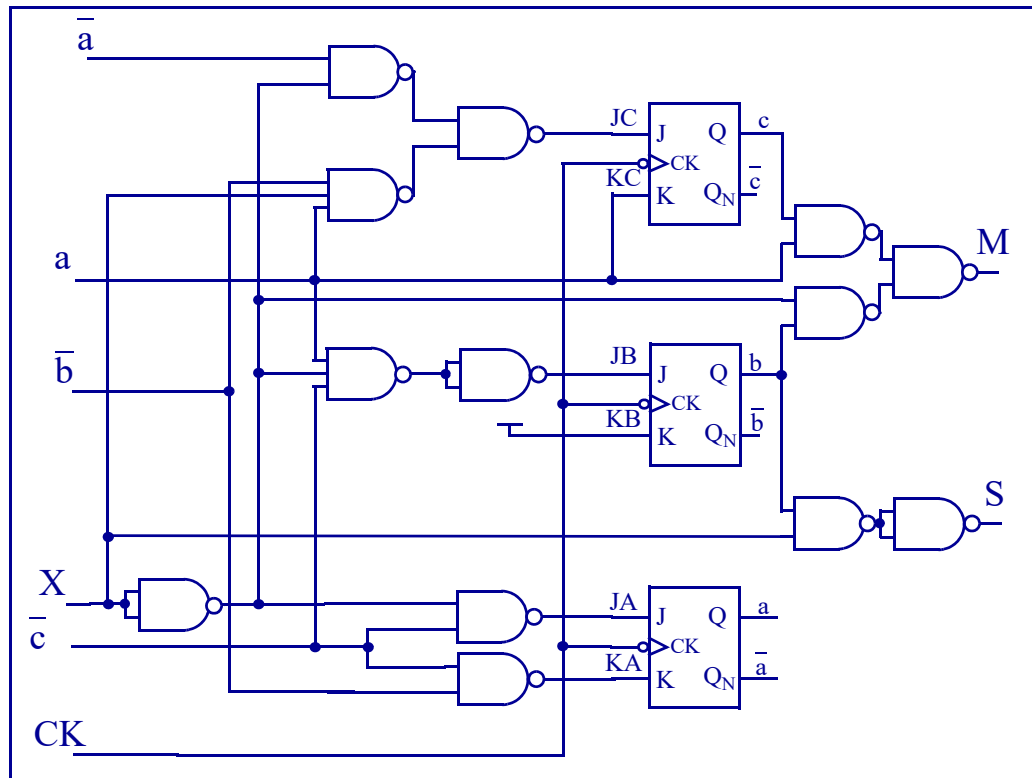
Tras la desactivación de CL, y dado que la entrada C está a cero durante los cuatro ciclos de reloj siguientes, con cada flanco de subida de la señal de reloj las variables de estado cambian, transitando el sistema entre estados $A \rightarrow B \rightarrow C \rightarrow D \rightarrow A$. La salida S se activa durante la permanencia del sistema en el estado D.

A partir de aquí la señal de entrada C cambia uno y permanece el resto del tiempo considerado en este cronograma. Por tanto, el gráfico muestra cómo, en este caso, el sistema evoluciona siguiendo la secuencia descendiente $D \rightarrow C \rightarrow B \rightarrow A \rightarrow D \rightarrow C \rightarrow \dots$, mientras que la salida S permanece siempre a cero.

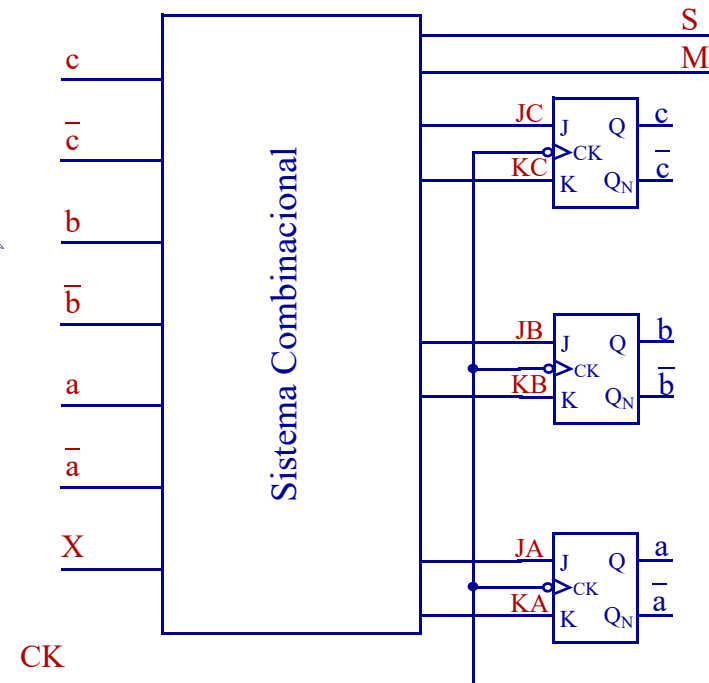
Este sistema es un caso particular de un tipo más general de sistemas que se conocen con el nombre de **generadores de secuencia o contadores**.

Análisis de Sistemas digitales secuenciales síncronos. Máquinas de estados

- Ejemplo2: Determina el comportamiento del siguiente sistema secuencial.



a) *Identificación del sistema: Entradas, salidas, elementos de memoria y variables de estado.*



b) *Funciones booleanas de salida y de excitación de los elementos de memoria.*

Función booleanas de salida

$$M(X, c, b, a) = (b \cdot \bar{X}) + (c \cdot a) \quad S(X, c, b, a) = (b \cdot X)$$

Función excitación de los FF

$$JA(X, c, b, a) = X + c \quad KA(X, c, b, a) = c + b$$

$$JB(X, c, b, a) = \bar{c} \cdot a \cdot \bar{X} \quad KB(X, c, b, a) = 1$$

$$JC(X, c, b, a) = \bar{a} \cdot \bar{X} + \bar{b} \cdot a \cdot X \quad KC(X, c, b, a) = a$$

Los estados siguientes A,B,C se actualizan según la tabla de comportamiento

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

Análisis de Sistemas digitales secuenciales síncronos. Máquinas de estados

● Ejemplo2: Continuación

Función booleanas de salida

$$M(X,c,b,a) = (b \cdot \bar{X}) + (c \cdot a) \quad S(X,c,b,a) = (b \cdot X)$$

Función excitación de los FF

$$JA(X,c,b,a) = X + c \quad KA(X,c,b,a) = c + b$$

$$JB(X,c,b,a) = \bar{c} \cdot a \cdot \bar{X} \quad KB(X,c,b,a) = 1$$

$$JC(X,c,b,a) = \bar{a} \cdot \bar{X} + \bar{b} \cdot a \cdot X \quad KC(X,c,b,a) = a$$

c) Tabla de transición de estados.

CK JK	Q_{n+1} \bar{Q}_{n+1}
0 0	Q_n
0 1	0 1
1 0	1 0
1 1	\bar{Q}_n

Entrada X	Estado presente c b a	Entradas de FF JC KC JB KB JA KA	Estado siguiente C B A	Salidas S M
0	0 0 0	1 0 0 1 0 0	1 0 0	0 0
0	0 0 1	0 1 1 1 0 0	0 1 1	0 0
0	0 1 0	1 0 0 1 0 1	1 0 0	0 1
0	0 1 1	0 1 1 1 0 1	0 0 0	0 1
0	1 0 0	1 0 0 1 1 1		0 0
0	1 0 1	0 1 0 1 1 1		0 1
0	1 1 0	1 0 0 1 1 1		0 1
0	1 1 1	0 1 0 1 1 1		0 1

Entrada X	Estado presente c b a	Entradas de FF JC KC JB KB JA KA	Salidas S M
0	0 0 0	1 0 0 1 0 0	0 0
0	0 0 1	0 1 1 1 0 0	0 0
0	0 1 0	1 0 0 1 0 1	0 1
0	0 1 1	0 1 1 1 0 1	0 1
0	1 0 0	1 0 0 1 1 1	0 0
0	1 0 1	0 1 0 1 1 1	0 1
0	1 1 0	1 0 0 1 1 1	0 1
0	1 1 1	0 1 0 1 1 1	0 1
1	0 0 0	0 0 0 1 1 0	0 0
1	0 0 1	1 1 0 1 1 0	0 0
1	0 1 0	0 0 0 1 1 1	1 0
1	0 1 1	0 1 0 1 1 1	1 0
1	1 0 0	0 0 0 1 1 1	0 0
1	1 0 1	1 1 0 1 1 1	0 1
1	1 1 0	0 0 0 1 1 1	1 0
1	1 1 1	0 1 0 1 1 1	1 1

Se completa la columna de Estado siguiente con la ayuda de la tabla de comportamiento del FF - JK

Análisis de Sistemas digitales secuenciales síncronos. Máquinas de estados

● Ejemplo2: Continuación

Entrada X	Estado presente <i>c b a</i>	Entradas de FF <i>JC KC JB KB JA KA</i>						Estado siguiente <i>C B A</i>	Salidas <i>S M</i>
0	0 0 0	1	0	0	1	0	0	1 0 0	0 0
0	0 0 1	0	1	1	1	0	0	0 1 1	0 0
0	0 1 0	1	0	0	1	0	1	1 0 0	0 1
0	0 1 1	0	1	1	1	0	1	0 0 0	0 1
0	1 0 0	1	0	0	1	1	1	1 0 1	0 0
0	1 0 1	0	1	0	1	1	1	0 0 0	0 1
0	1 1 0	1	0	0	1	1	1	1 0 1	0 1
0	1 1 1	0	1	0	1	1	1	0 0 0	0 1
1	0 0 0	0	0	0	1	1	0	0 0 1	0 0
1	0 0 1	1	1	0	1	1	0	1 0 1	0 0
1	0 1 0	0	0	0	1	1	1	0 0 1	1 0
1	0 1 1	0	1	0	1	1	1	0 0 0	1 0
1	1 0 0	0	0	0	1	1	1	1 0 1	0 0
1	1 0 1	1	1	0	1	1	1	0 0 0	0 1
1	1 1 0	0	0	0	1	1	1	1 0 1	1 0
1	1 1 1	0	1	0	1	1	1	0 0 0	1 1



Tabla de transición de estados

Entrada X	Estado presente <i>c b a</i>	Estado siguiente <i>C B A</i>	Salidas <i>S M</i>
0	0 0 0	1 0 0	0 0
0	0 0 1	0 1 1	0 0
0	0 1 0	1 0 0	0 1
0	0 1 1	0 0 0	0 1
0	1 0 0	1 0 1	0 0
0	1 0 1	0 0 0	0 1
0	1 1 0	1 0 1	0 1
0	1 1 1	0 0 0	0 1
1	0 0 0	0 0 1	0 0
1	0 0 1	1 0 1	0 0
1	0 1 0	0 0 1	1 0
1	0 1 1	0 0 0	1 0
1	1 0 0	1 0 1	0 0
1	1 0 1	0 0 0	0 1
1	1 1 0	1 0 1	1 0
1	1 1 1	0 0 0	1 1

Análisis de Sistemas digitales secuenciales síncronos. Máquinas de estados

● Ejemplo2: Continuación

Entrada X	Estado presente c b a	Estado siguiente C B A	Salidas S M
0	0 0 0	1 0 0	0 0
0	0 0 1	0 1 1	0 0
0	0 1 0	1 0 0	0 1
0	0 1 1	0 0 0	0 1
0	1 0 0	1 0 1	0 0
0	1 0 1	0 0 0	0 1
0	1 1 0	1 0 1	0 1
0	1 1 1	0 0 0	0 1
1	0 0 0	0 0 1	0 0
1	0 0 1	1 0 1	0 0
1	0 1 0	0 0 1	1 0
1	0 1 1	0 0 0	1 0
1	1 0 0	1 0 1	0 0
1	1 0 1	0 0 0	0 1
1	1 1 0	1 0 1	1 0
1	1 1 1	0 0 0	1 1

Tabla de transición de estados

Ep	Es		Salidas			
	X=0 X=1		X=0 X=1			
			S M	S M		
A	E	B	0 0	0 0		
B	D	F	0 0	0 0		
C	E	B	0 1	1 0		
D	A	A	0 1	1 0		
E	F	F	0 0	0 0		
F	A	A	0 1	0 1		
G	F	F	0 1	1 0		
H	A	A	0 1	1 1		

Estados	Códigos c b a
A	0 0 0
B	0 0 1
C	0 1 0
D	0 1 1
E	1 0 0
F	1 0 1
G	1 1 0
H	1 1 1

-Los estados C, G y H (sombreados en la tabla) no aparecen en la columna de estados siguientes, esto quiere decir que nunca serán alcanzables durante el funcionamiento normal del sistema.

Por tanto se puede decir que este sistema posee solamente cinco estados alcanzables, estos son los estados A, B, D, E, F, que son los únicos que deben ser considerados para entender en comportamiento del sistema.

La salida S solamente toma el valor lógico 1 para el estado alcanzable D, siempre y cuando la entrada X tome el valor 1.

La salida M toma el valor 1 en el estado alcanzables F, independientemente del valor de la entrada X; y en el estado D, siempre y cuando la entrada X tome el valor 0.

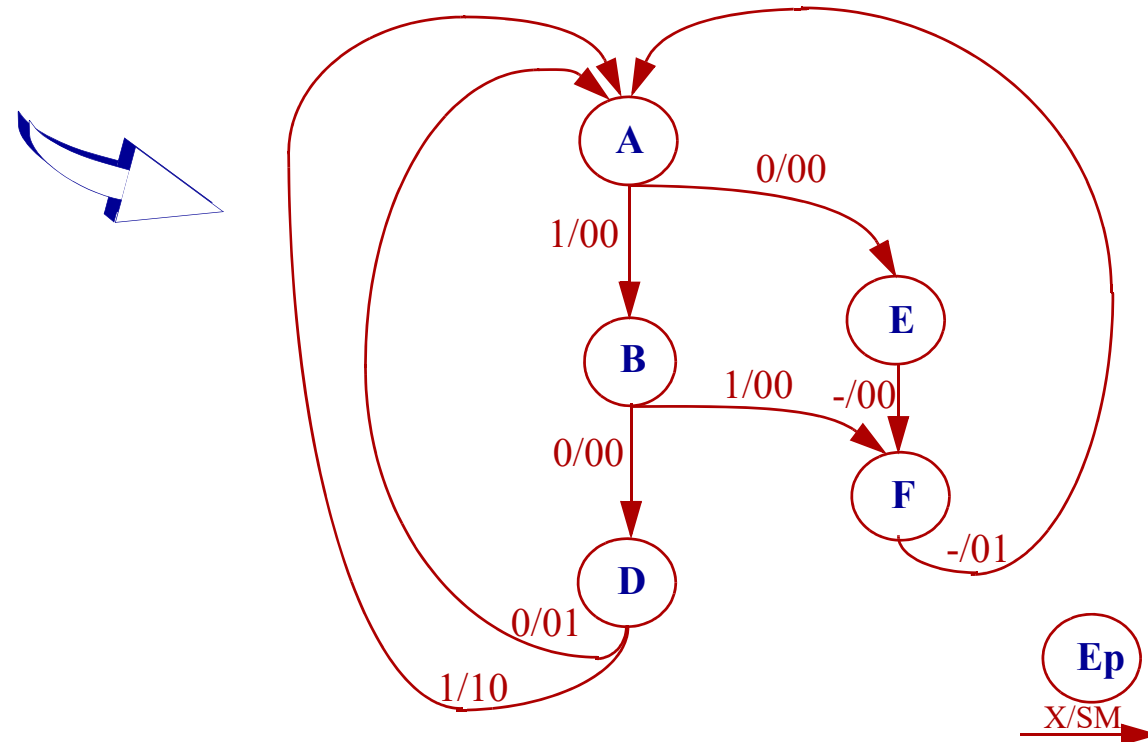
Análisis de Sistemas digitales secuenciales síncronos. Máquinas de estados

● Ejemplo2: Continuación

Tabla de transición de estados

Ep	Es		Salidas	
	X=0	X=1	X=0	X=1
			S M	S M
A	E	B	0 0	0 0
B	D	F	0 0	0 0
C	E	B	0 1	1 0
D	A	A	0 1	1 0
E	F	F	0 0	0 0
F	A	A	0 1	0 1
G	F	F	0 1	1 0
H	A	A	0 1	1 1

d) Descripción del comportamiento por medio de un diagrama de transición de estados.



El sistema es capaz de detectar la secuencia de entrada de tres bits X: 1->0->1 en cuyo caso activa la señal S.

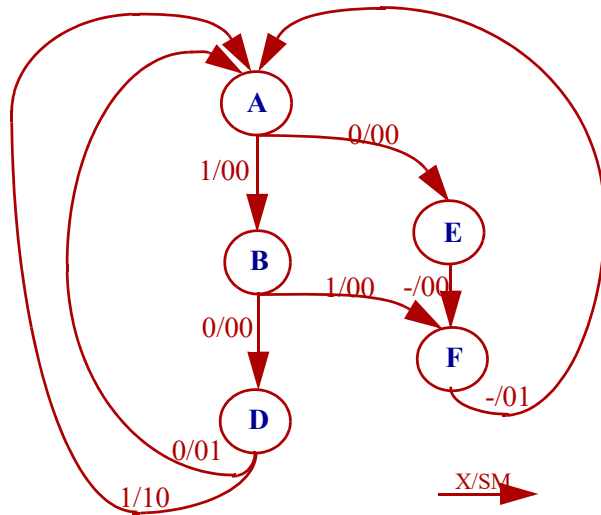
Si la secuencia de entrada es otra cualquiera entonces activa la salida M.

Tanto S como M se activan coincidiendo con la llegada del tercer bit de la secuencia.

Este tipo de sistemas secuenciales que se denominan **detectores de secuencias binarias**

Análisis de Sistemas digitales secuenciales síncronos. Máquinas de estados

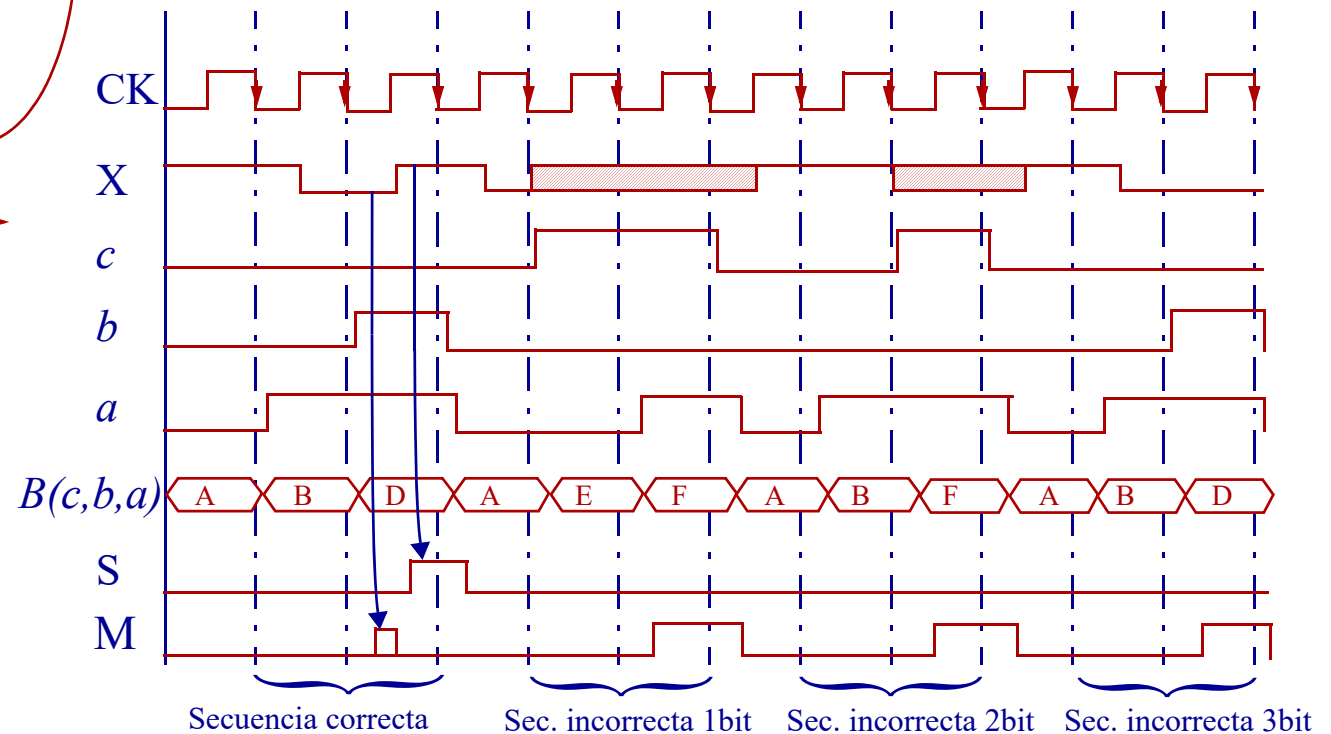
● Ejemplo2: Continuación



Estados	Códigos c b a
A	0 0 0
B	0 0 1
C	0 1 0
D	0 1 1
E	1 0 0
F	1 0 1
G	1 1 0
H	1 1 1

e) Descripción del comportamiento por medio de un cronograma

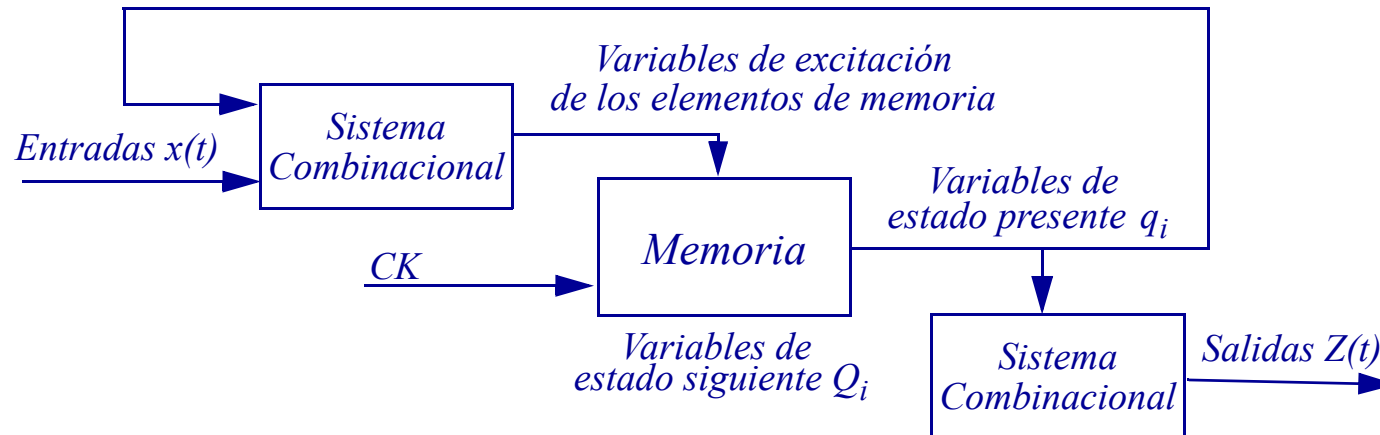
Este cronograma ejercita todos los estados posibles del sistema.



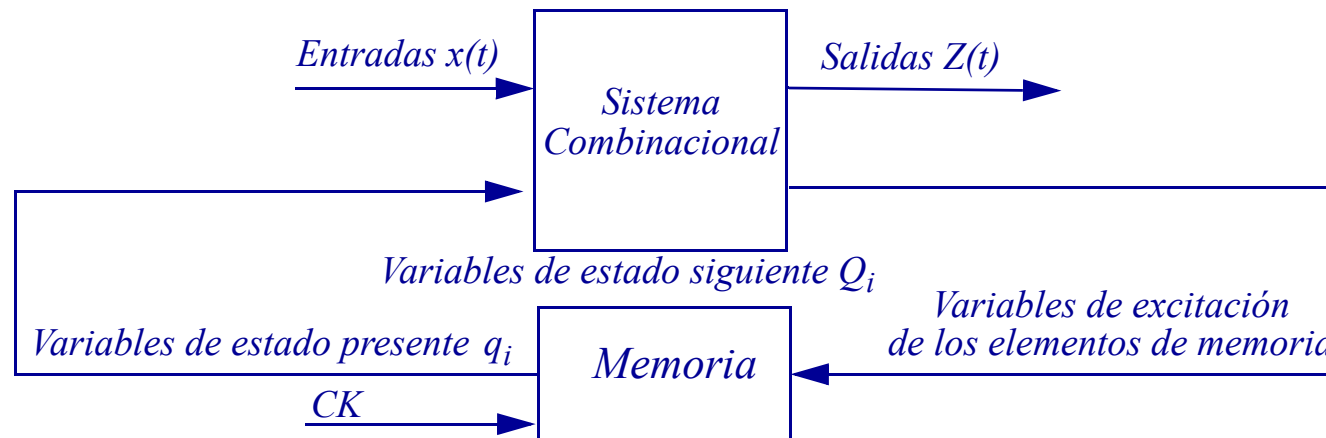
(En este cronograma hemos supuesto que el sistema ha sido llevado inicialmente al estado A)

Sistemas digitales secuenciales. Modelos de Mealy y de Moore

- El **modelo de Moore** es un caso particular del modelo de Huffman para sistemas secuenciales. Las máquinas de estados que siguen la estructura del modelo de Moore se caracterizan por que **las funciones de salida dependen exclusivamente de las variables de estado presente**.

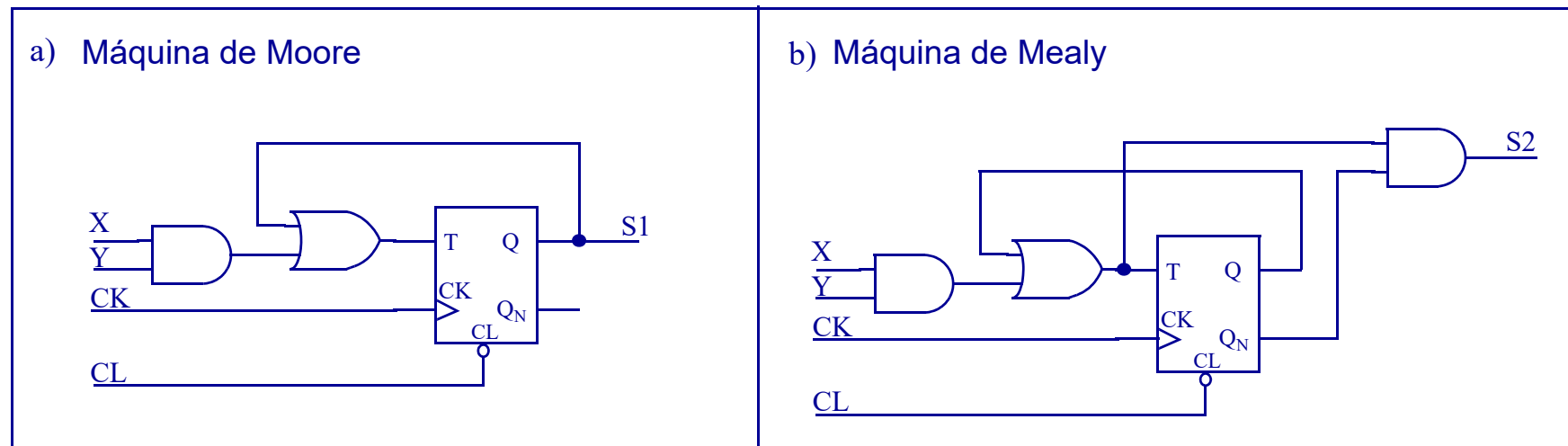


- El **modelo de Mealy** es el modelo de Huffman particularizado para sistemas secuenciales. En las máquinas de estados que siguen la estructura del modelo de Mealy **las funciones de salida dependen de las entradas y de las variables de estado presente**.



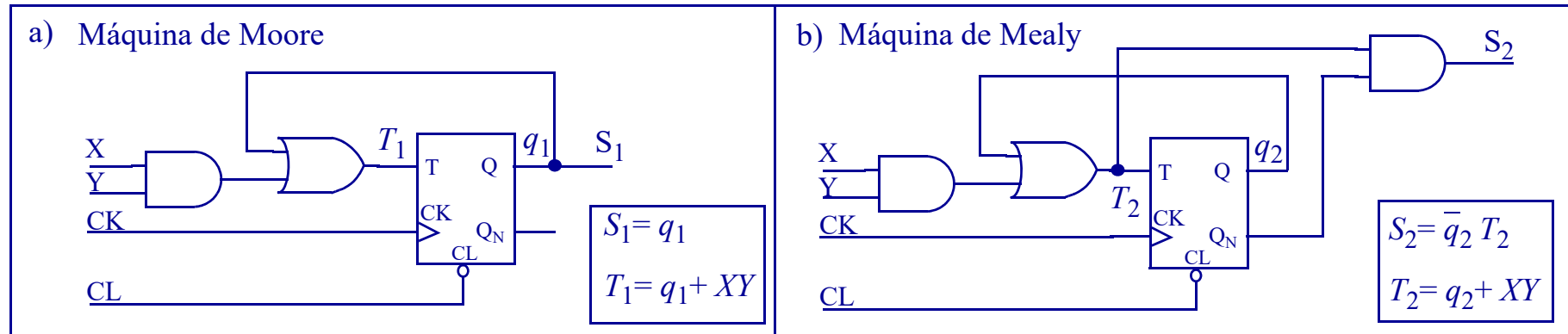
Sistemas digitales secuenciales. Modelos de Mealy y de Moore

- Ambos modelos son funcionalmente equivalentes, en el sentido de que cualquier máquina de estados construida según el modelo de Mealy, también puede ser construida siguiendo el modelo de Moore.
Sin embargo puede existir una diferencia apreciable en cuanto a la complejidad de la implementación de un sistema particular, siguiendo uno u otro modelo, principalmente en lo que respecta a nº de estados. En general el nº de estados de una máquina de estados realizada con un modelo de Moore será mayor o igual al de una máquina de estados realizada con un modelo Mealy.
Por contra, las funciones de salida de una máquina Moore son más simples que las del modelo Mealy. Además los cambios en las salidas en las máquinas de Moore son síncronos con la señal de reloj, y sus salidas están libres de la influencia de los cambios transitorios o espúreos en las entradas.
- **Ejemplo3: Comparación de un sistema diseñado siguiendo el modelo de Moore o de Mealy.**



Sistemas digitales secuenciales. Modelos de Mealy y de Moore

● Ejemplo3: Continuación.



A es $q_1 = 0$

B es $q_1 = 1$

Diagrama Moore

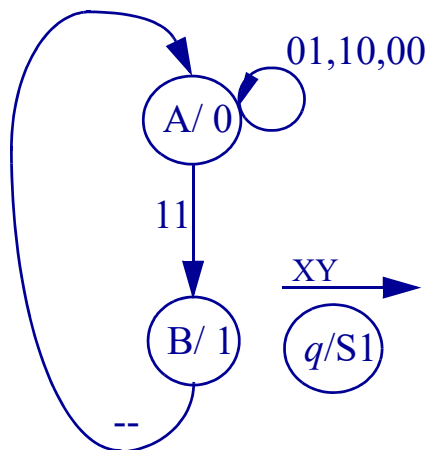
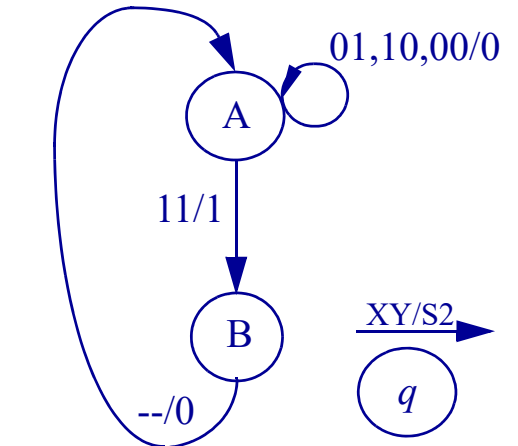


Tabla modelo Moore					Tabla modelo Mealy				
Ep	Ent	Es	Ent de FF	Sal	Ep	Ent	Es	Ent de FF	Sal
q_1	X Y	Q_1	T1	S1	q_2	X Y	Q_2	T2	S2
0	0 0	0	0	0	0	0 0	0	0	0
0	0 1	0	0	0	0	0 1	0	0	0
0	1 0	0	0	0	0	1 0	0	0	0
0	1 1	1	1	0	0	1 1	1	1	1
1	0 0	0	1	1	1	0 0	0	1	0
1	0 1	0	1	1	1	0 1	0	1	0
1	1 0	0	1	1	1	1 0	0	1	0
1	1 1	0	1	1	1	1 1	0	1	0

A es $q_2 = 0$

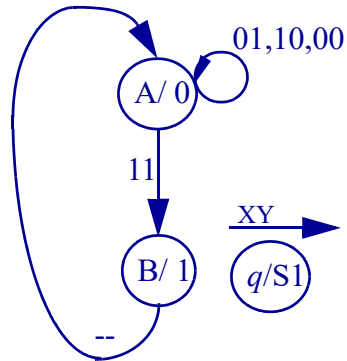
B es $q_2 = 1$

Diagrama Mealy



Sistemas digitales secuenciales. Modelos de Mealy y de Moore

● Ejemplo3: Continuación. Diagrama Moore



CRONOGRAMA

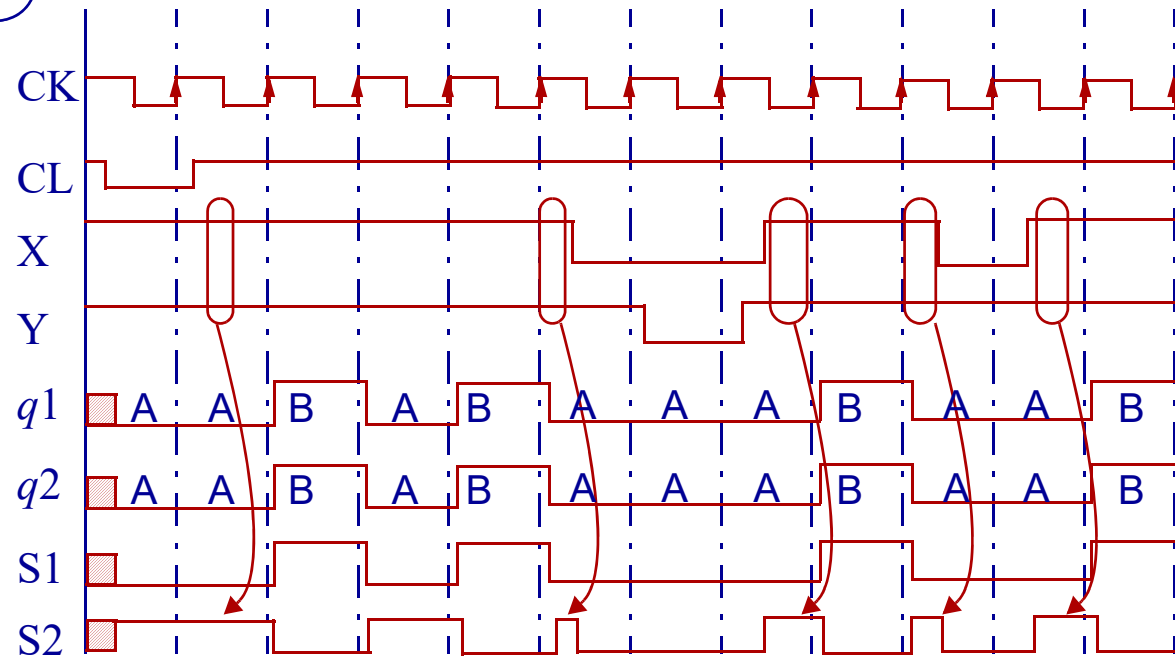
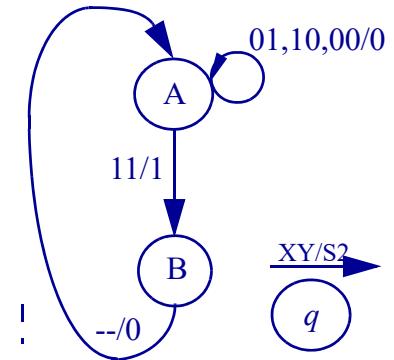


Diagrama Mealy



Aunque la secuencia de estados es la misma, no lo es la salida.

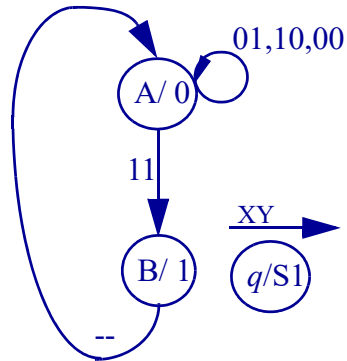
En el circuito de Moore la salida S1 cambia siguiendo las transiciones del estado q1,

En el circuito de Mealy, los cambios en las entradas se propagan directamente a la salida S2.

Sistemas digitales secuenciales. Modelos de Mealy y de Moore

● Ejemplo3: Continuación.

Diagrama Moore



b) Máquina de Mealy

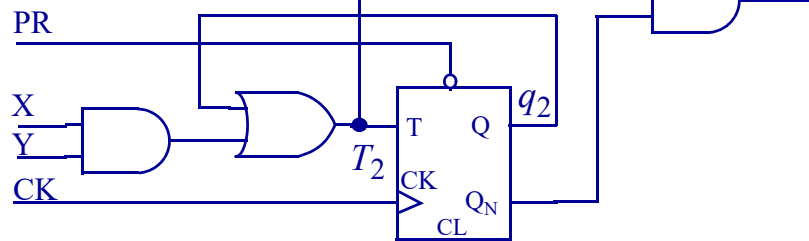
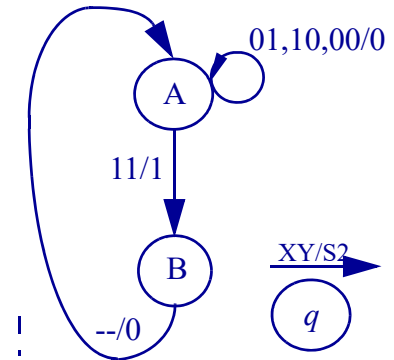
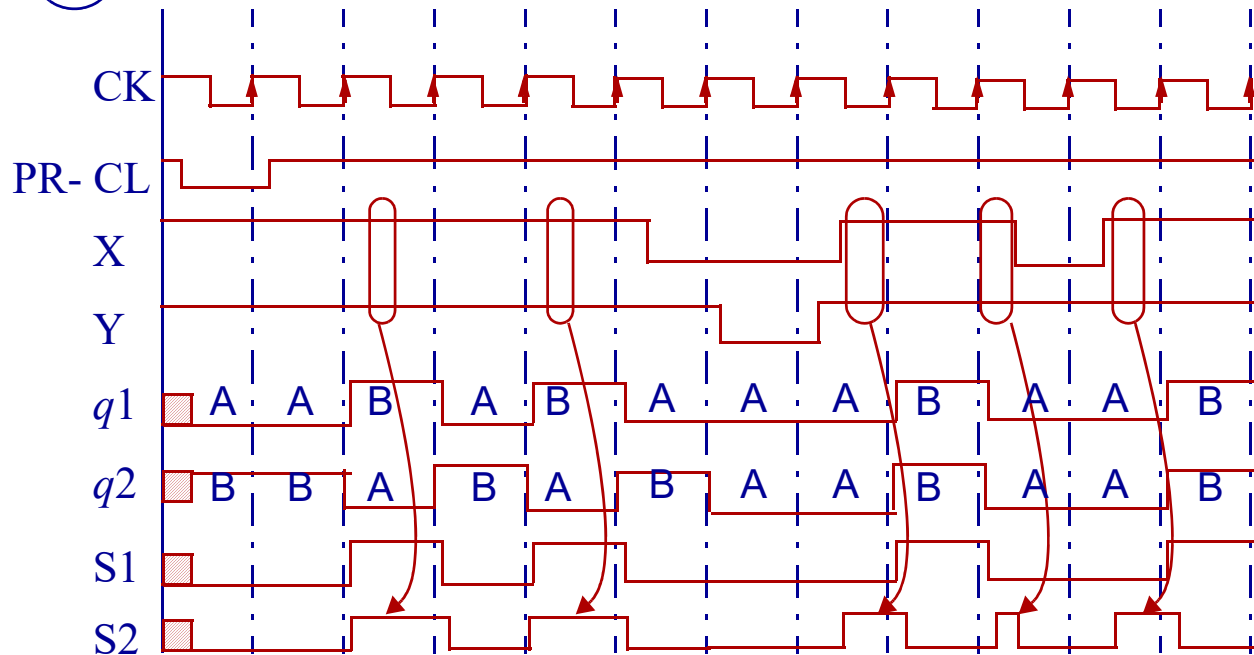


Diagrama Mealy



CRONOGRAMA



Si en el circuito de Mealy, cambiamos el CL por Preset su secuencia de estados comienza en el estado B de modo que ahora la salida en ambos circuitos coincide durante los primeros ciclos, sin embargo después la secuencia de estados se iguala y se produce un desfase de un ciclo.



Escuela de Ingenierías Industriales

Titulación: Grado en Ingeniería Electrónica, Robótica y Mecatrónica

Asignatura: Electrónica Digital

Tema 8: Análisis y Síntesis de Sistemas Secuenciales Síncronos



TEMA 8: ANÁLISIS Y SÍNTESIS DE SISTEMAS SECUENCIALES SÍNCRONOS

8.1. Análisis de sistemas secuenciales síncronos.

8.1.1 Introducción. Modelo de Huffman para sistemas síncronos. Variables y ecuaciones booleanas.

8.1.2 Funciones de entrada de los biestables. Funciones de salida.

8.1.3 Tabla de estados: Tablas características de los biestables. Tabla de Transición de estados.

8.1.4 Diagramas de estados. Cronogramas.

8.1.5 Modelos de Mealy y de Moore para los sistemas secuenciales síncronos.

8.2. Diseño de sistemas secuenciales síncronos.

8.2.1 Tabla de excitación de los biestables.

8.2.2 Reducción de estados: Tablas de implicación. Estados equivalentes y estados compatibles.

8.2.3 Asignación de estados.

8.2.4 Ejemplos de diseño.

8.2.5

8.3. Diagramas de estados en HDL.

Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

- El procedimiento a seguir puede resumirse en los siguientes puntos:
 - a) Identificar el número de **entradas** y **salidas** para cumplir con los requerimientos de diseño.
 - b) Construir una tabla que reúna las especificaciones del problema, de modo que, para cada una de las combinaciones de entrada posibles, se clarifique cuál será la acción a llevar a cabo; así como la respuesta o salida del sistema.
 - c) Construir un **Diagrama de transición de estados** en la que se concrete el nº de estados del sistema, su evolución y el comportamiento desde el punto de vista de entrada/salida.
 - d) Construir, a partir del diagrama de transición de estados, una **Tabla de transición de estados**.
 - e) Emplear las técnicas de **minimización de estados** para obtener una tabla de transición de estados que contenga un nº de estados mínimo y que siga cumpliendo con la especificaciones del problema.
 - f) Determinar las variables necesarias para **codificar los estados** del sistema y realizar una **asignación adecuada de códigos**.
 - g) **Seleccionar** el tipo de **flip-flops** para realizar dichas variables de estado.
 - h) Sustituir en la tabla de transición de estados los códigos asignados a los estados y completar la columna **Entrada de los flip-flops** a partir de la correspondiente **Tabla de excitación**.
 - i) Obtener las **funciones de salida** y las **funciones de entrada de los flip-flops** irreducibles mínimas empleando las técnicas de minimización combinacional y obtener el **diagrama lógico** del sistema.
 - j) Si hay combinaciones de las variables de estado no utilizadas, verificar que su aparición accidental en el sistema no lleva a situaciones erróneas irreversibles.

Tablas de excitación de los elementos de memoria.

- La **Tabla de excitación de un biestable o flip-flop**, indica qué combinación de entrada permite que en el elemento de memoria se produzca una transición de estado dada. Durante el proceso de diseño se utiliza para completar la columna **Entradas de los flip-flops**.

**Tabla de excitación
del flip-flop SR**

Q_n	Q_{n+1}	S	R
0	0	0	x
0	1	1	0
1	0	0	1
1	1	x	0

**Tabla de excitación
del flip-flop RS**

Q_n	Q_{n+1}	\bar{S}	\bar{R}
0	0	1	x
0	1	0	1
1	0	1	0
1	1	x	1

**Tabla de excitación
del flip-flop JK**

Q_n	Q_{n+1}	J	K
0	0	0	x
0	1	1	x
1	0	x	1
1	1	x	0

**Tabla de excitación
del flip-flop D**

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

**Tabla de excitación
del flip-flop T**

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

Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 1:

Se bombea agua a una depósito mediante dos bombas hidráulicas P1 y P2.

Ambas bombas deben activarse cuando el nivel de agua es inferior al Nivel1, y deben permanecer en marcha hasta que el agua alcance el Nivel2, momento en el que la bomba P1 se desactiva y permanece desactivada hasta que el agua vuelva a bajar del Nivel1 otra vez.

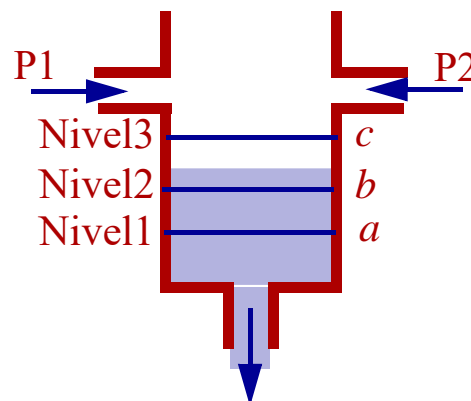
La bomba P2 permanece activada hasta que se alcanza el Nivel3, y entonces se desactiva también, y permanece así hasta que el nivel de agua cae de nuevo por debajo del Nivel1.

Se usan sensores de nivel para la detección del nivel de agua de la siguiente manera:

- Señal $a = 1$; cuando el agua está al Nivel1 o por encima. Si no $a = 0$.
- Señal $b = 1$; cuando el agua está al Nivel2 o por encima. Si no $b = 0$.
- Señal $c = 1$; cuando el agua está al Nivel3 o por encima. Si no $c = 0$.

Diseñar un circuito digital síncrono que permita controlar las bombas P1 y P2 de acuerdo con las especificaciones del problema.

La siguiente figura ilustra la situación:



Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 1: Continuación

a) *Identificar el número de entradas y salidas necesarias para cumplir con los requerimientos del diseño propuesto.*

El sistema propuesto ha de poseer tres entradas, asociadas a las señales binarias a , b y c . Si llamamos N al nivel de agua en el depósito en un instante dado y $N1$ $N2$ $N3$ a las correspondientes referencias de nivel, se tiene:

$a = 1$	si	$N \geq N1$	$a = 0$	si	$N < N1$
$b = 1$	si	$N \geq N2$	$b = 0$	si	$N < N2$
$c = 1$	si	$N \geq N3$	$c = 0$	si	$N < N3$

Para controlar el encendido y apagado de las dos bombas el sistema tendrá que disponer de al menos dos salidas; llamaremos $P1$ y $P2$ a las señales que controlan las bombas 1 y 2.

Asignaremos $P_i = 1$ a la situación bomba i funcionando y $P_i = 0$ en caso contrario.

Del enunciado del problema se desprende que las bombas de agua se pueden encontrar en tres situaciones distintas, cada una de las cuales estará asociada a una combinación de las variables de salida. De aquí se puede intuir que el sistema de control pedido puede presentar tres estados distintos.

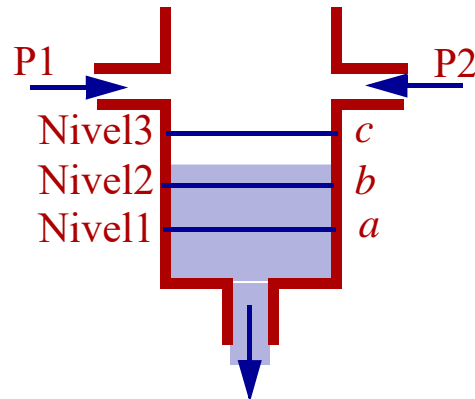
Vamos a resumir esas tres situaciones

$P1$	$P2$	Situación
0	0	A ambas bombas apagadas
0	1	B solo funciona la bomba 2
1	1	C ambas bombas funcionando
1	0	D situación no contemplada

Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 1: Continuación

b) Construir una tabla que resuma las especificaciones del problema de modo que para cada una de las combinaciones de entrada posibles se clarifique cual será la acción a llevar a cabo, así como la respuesta o salida del sistema.



Cod	Entradas <i>a b c</i>	Significado	Acción	
E0	0 0 0	$N < N1$	$P1 = 1$	$P2 = 1$
E1	0 0 1	No se da	$P1 = 0$	$P2 = 0$
E2	0 1 0	No se da	$P1 = 0$	$P2 = 0$
E3	0 1 1	No se da	$P1 = 0$	$P2 = 0$
E4	1 0 0	$N1 \leq N < N2$	$P1_{n+1} = P1_n$	$P2_{n+1} = P2_n$
E5	1 0 1	No se da	$P1 = 0$	$P2 = 0$
E6	1 1 0	$N2 \leq N < N3$	$P1 = 0$	$P2_{n+1} = P2_n$
E7	1 1 1	$N \geq N3$	$P1 = 0$	$P2 = 0$

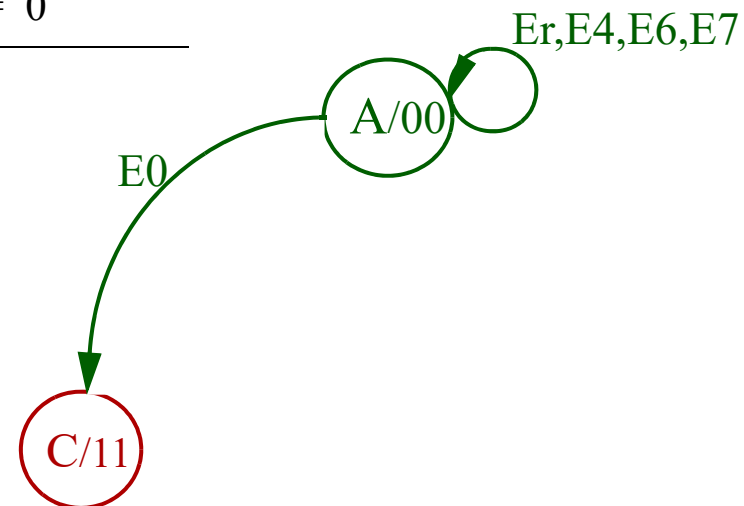
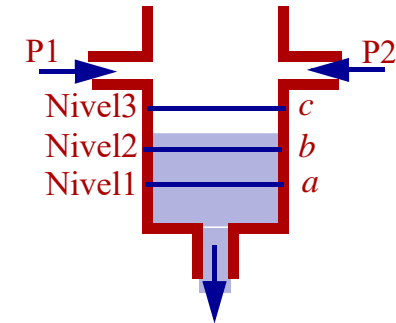
Llamaremos **Er** al conjunto de entradas que suponemos que no se van a dar $Er = \{E1, E2, E3, E5\}$

Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 1: Continuación

c) Construir, a partir de la información de la anterior tabla, un Diagrama de transición de estados en el que se concrete el n° de estados del sistema, su evolución y el comportamiento desde el punto de vista de entrada/salida.

Cod	a b c	Sign.	Acción	
E0	0 0 0	$N < N1$	$P1 = 1$	$P2 = 1$
Er		No se da	$P1 = 0$	$P2 = 0$
E4	1 0 0	$N1 \leq N < N2$	$P1_{n+1} = P1_n$	$P2_{n+1} = P2_n$
E6	1 1 0	$N2 \leq N < N3$	$P1 = 0$	$P2_{n+1} = P2_n$
E7	1 1 1	$N \geq N3$	$P1 = 0$	$P2 = 0$

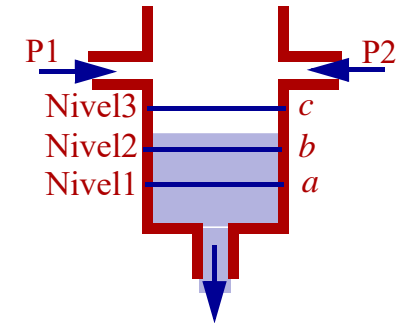


P1 P2	Estado
0 0	A ambas bombas apagadas
0 1	B solo funciona la bomba 2
1 1	C ambas bombas funcionando
1 0	D situación no contemplada

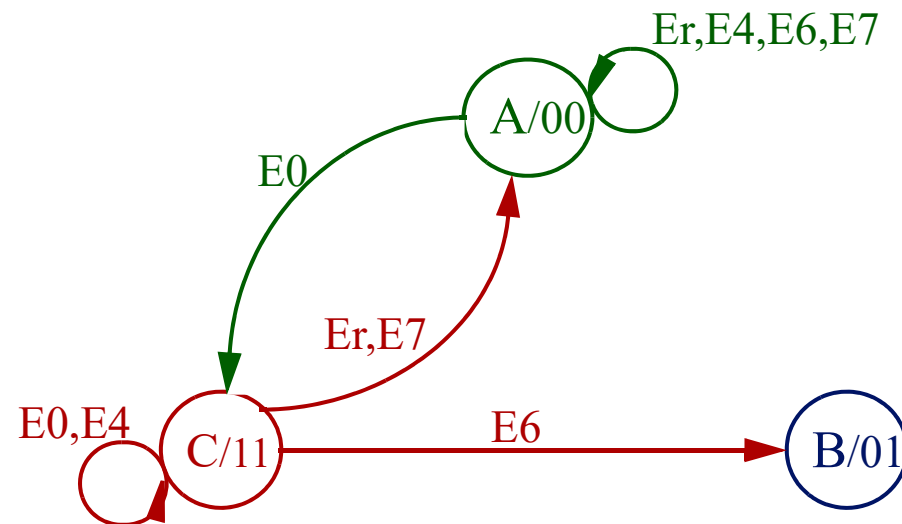
Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 1: Continuación

Cod	a b c	Sign.	Acción	
E0	0 0 0	$N < N1$	$P1 = 1$	$P2 = 1$
Er		No se da	$P1 = 0$	$P2 = 0$
E4	1 0 0	$N1 \leq N < N2$	$P1_{n+1} = P1_n$	$P2_{n+1} = P2_n$
E6	1 1 0	$N2 \leq N < N3$	$P1 = 0$	$P2_{n+1} = P2_n$
E7	1 1 1	$N \geq N3$	$P1 = 0$	$P2 = 0$



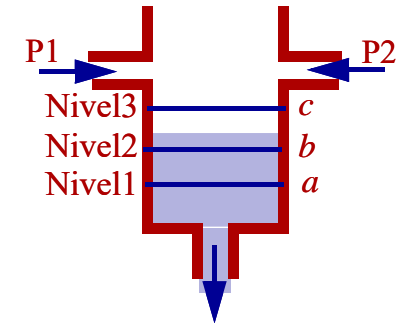
P1 P2	Estado
0 0	A ambas bombas apagadas
0 1	B solo funciona la bomba 2
1 1	C ambas bombas funcionando
1 0	D situación no contemplada



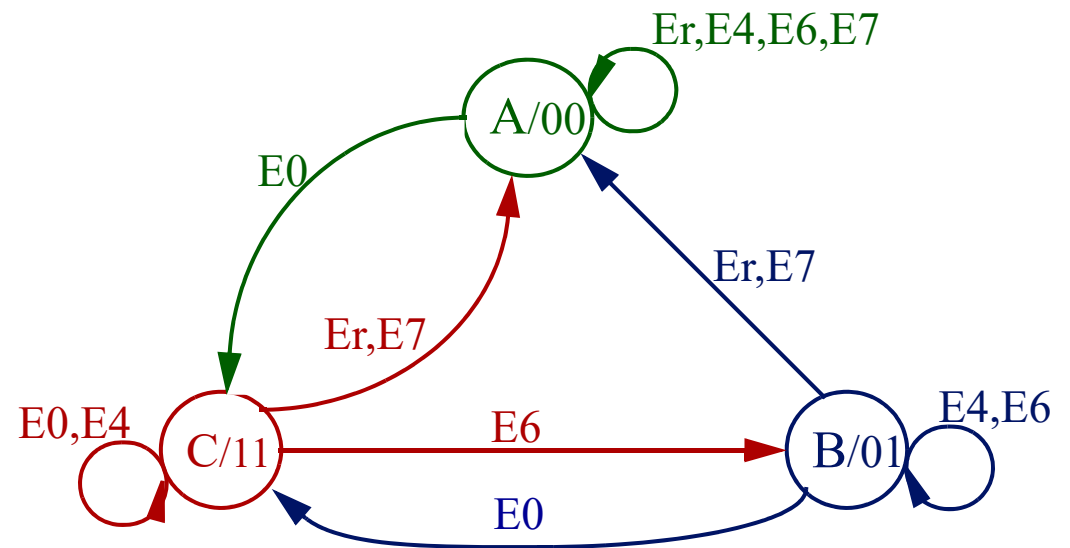
Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 1: Continuación

Cod	a b c	Sign.	Acción	
E0	0 0 0	$N < N1$	$P1 = 1$	$P2 = 1$
Er		No se da	$P1 = 0$	$P2 = 0$
E4	1 0 0	$N1 \leq N < N2$	$P1_{n+1} = P1_n$	$P2_{n+1} = P2_n$
E6	1 1 0	$N2 \leq N < N3$	$P1 = 0$	$P2_{n+1} = P2_n$
E7	1 1 1	$N \geq N3$	$P1 = 0$	$P2 = 0$



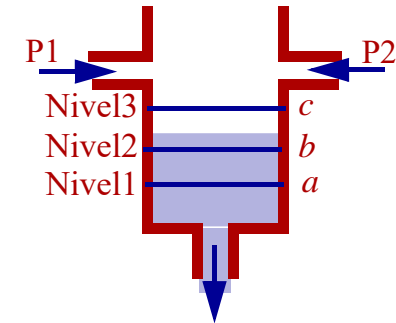
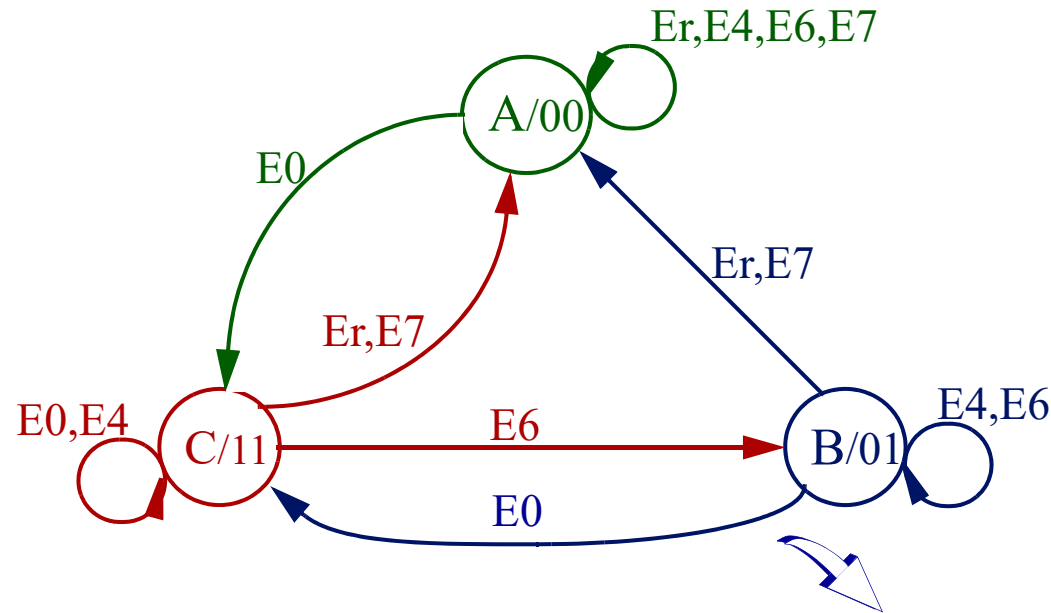
P1 P2	Estado
0 0	A ambas bombas apagadas
0 1	B solo funciona la bomba 2
1 1	C ambas bombas funcionando
1 0	D situación no contemplada



Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 1: Continuación

d) Construir, a partir del diagrama de transición de estados, una Tabla de transición de estados.



P1	P2	Estado
0	0	A ambas bombas apagadas
0	1	B solo funciona la bomba 2
1	1	C ambas bombas funcionando
1	0	D situación no contemplada

Tabla de transición de estados

Ep	Es								Salidas	
	E0	E1	E2	E3	E4	E5	E6	E7	P1	P2
A	C	A	A	A	A	A	A	A	0	0
B	C	A	A	A	B	A	B	A	0	1
C	C	A	A	A	C	A	B	A	1	1

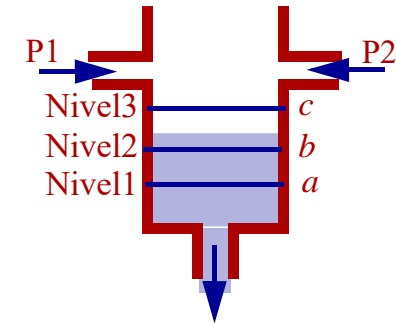
Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 1: Continuación

e) Emplear las técnicas de minimización de estados para obtener una tabla de transición de estados que contenga un n° de estados mínimo y que cumpla con la especificaciones del problema.

Tabla de transición de estados mínima

Ep	Es								Salidas	
	E0	E1	E2	E3	E4	E5	E6	E7	P1	P2
A	C	A	A	A	A	A	A	A	0	0
B	C	A	A	A	B	A	B	A	0	1
C	C	A	A	A	C	A	B	A	1	1



P1	P2	Estado
0	0	A ambas bombas apagadas
0	1	B solo funciona la bomba 2
1	1	C ambas bombas funcionando
1	0	D situación no contemplada

f) Definir las variables binarias necesarias para codificar los estados del sistema y realizar una asignación adecuada de códigos.

Para simplificar las funciones codificamos los estados de forma que $P1 = q1$ y $P2 = q2$

Asignación de código a los estados		
ESTADOS	q1	q2
A	0	0
B	0	1
C	1	1

g) Seleccionar el tipo de flip-flops a utilizar para realizar dichas variables de estado.

En este ejemplo usaremos dos elementos diferentes para ilustrar cómo trabajar con ellos.

Usaremos un flip-flop D para la variable q2 y un JK para la variable q1

Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 1: Continuación

h) Sustituir en la tabla de transición de estados los códigos asignados a los estados y obtener la columna de Entrada de los flip-flops haciendo uso de las correspondientes Tablas de excitación

	Estado presente <i>q1 q2</i>		Entradas <i>a b c</i>			Estado siguiente <i>Q1 Q2</i>		Salidas <i>P1 P2</i>	
	<i>q1</i>	<i>q2</i>	<i>a</i>	<i>b</i>	<i>c</i>	<i>Q1</i>	<i>Q2</i>	<i>P1</i>	<i>P2</i>
A	0	0	0	0	0	1	1	0	0
	0	0	0	0	1	0	0	0	0
	0	0	0	1	0	0	0	0	0
	0	0	0	1	1	0	0	0	0
	0	0	1	0	0	0	0	0	0
	0	0	1	0	1	0	0	0	0
	0	0	1	1	0	0	0	0	0
	0	0	1	1	1	0	0	0	0
B	0	1	0	0	0	1	1	0	1
	0	1	0	0	1	0	0	0	1
	0	1	0	1	0	0	0	0	1
	0	1	0	1	1	0	0	0	1
	0	1	1	0	0	0	1	0	1
	0	1	1	0	1	0	0	0	1
	0	1	1	1	0	0	1	0	1
	0	1	1	1	1	0	0	0	1

Ep	Es								Salidas	
	E0	E1	E2	E3	E4	E5	E6	E7	P1	P2
A	C	A	A	A	A	A	A	A	0	0
B	C	A	A	A	B	A	B	A	0	1
C	C	A	A	A	C	A	B	A	1	1

ESTADOS	<i>q1 q2</i>
A	0 0
B	0 1
C	1 1

Estado presente <i>q1 q2</i>	Entradas <i>a b c</i>	Estado siguiente <i>Q1 Q2</i>	Salidas <i>P1 P2</i>
1 1	0 0 0	1 1	1 1
1 1	0 0 1	0 0	1 1
1 1	0 1 0	0 0	1 1
1 1	0 1 1	0 0	1 1
1 1	1 0 0	1 1	1 1
1 1	1 0 1	0 0	1 1
1 1	1 1 0	0 1	1 1
1 1	1 1 1	0 0	1 1

Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 1: Continuación

**Tabla de excitación
del flip-flop JK**

Q_n	Q_{n+1}	J	K
0	0	0	x
0	1	1	x
1	0	x	1
1	1	x	0

**Tabla de excitación
del flip-flop D**

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

Estado presente $q1$ $q2$	Entradas a b c	Estado siguiente $Q1$ $Q2$	Entradas de FF J1 K1 D2	Salidas P1 P2
0 0	0 0 0	1 1	1 x 1	0 0
0 0	0 0 1	0 0	0 x 0	0 0
0 0	0 1 0	0 0	0 x 0	0 0
0 0	0 1 1	0 0	0 x 0	0 0
0 0	1 0 0	0 0	0 x 0	0 0
0 0	1 0 1	0 0	0 x 0	0 0
0 0	1 1 0	0 0	0 x 0	0 0
0 0	1 1 1	0 0	0 x 0	0 0
0 1	0 0 0	1 1		0 1
0 1	0 0 1	0 0		0 1
0 1	0 1 0	0 0		0 1
0 1	0 1 1	0 0		0 1
0 1	1 0 0	0 1		0 1
0 1	1 0 1	0 0		0 1
0 1	1 1 0	0 1		0 1
0 1	1 1 1	0 0		0 1

A

B

Sistemas digitales secuenciales síncronos.

Síntesis de máquinas de estados.

Ejemplo de diseño 1: Continuación

Estado presente $q1\ q2$	Entradas $a\ b\ c$	Estado siguiente $Q1\ Q2$	Entradas de FF J1K1 D2	Salidas P1 P2
00	000	11	1 x 1	00
00	001	00	0 x 0	00
00	010	00	0 x 0	00
00	011	00	0 x 0	00
00	100	00	0 x 0	00
00	101	00	0 x 0	00
00	110	00	0 x 0	00
00	111	00	0 x 0	00
01	000	11	1 x 1	01
01	001	00	0 x 0	01
01	010	00	0 x 0	01
01	011	00	0 x 0	01
01	100	01	0 x 1	01
01	101	00	0 x 0	01
01	110	01	0 x 1	01
01	111	00	0 x 0	01
11	000	11	x 0 1	11
11	001	00	x 1 0	11
11	010	00	x 1 0	11
11	011	00	x 1 0	11
11	100	11	x 0 1	11
11	101	00	x 1 0	11
11	110	01	x 1 1	11
11	111	00	x 1 0	11

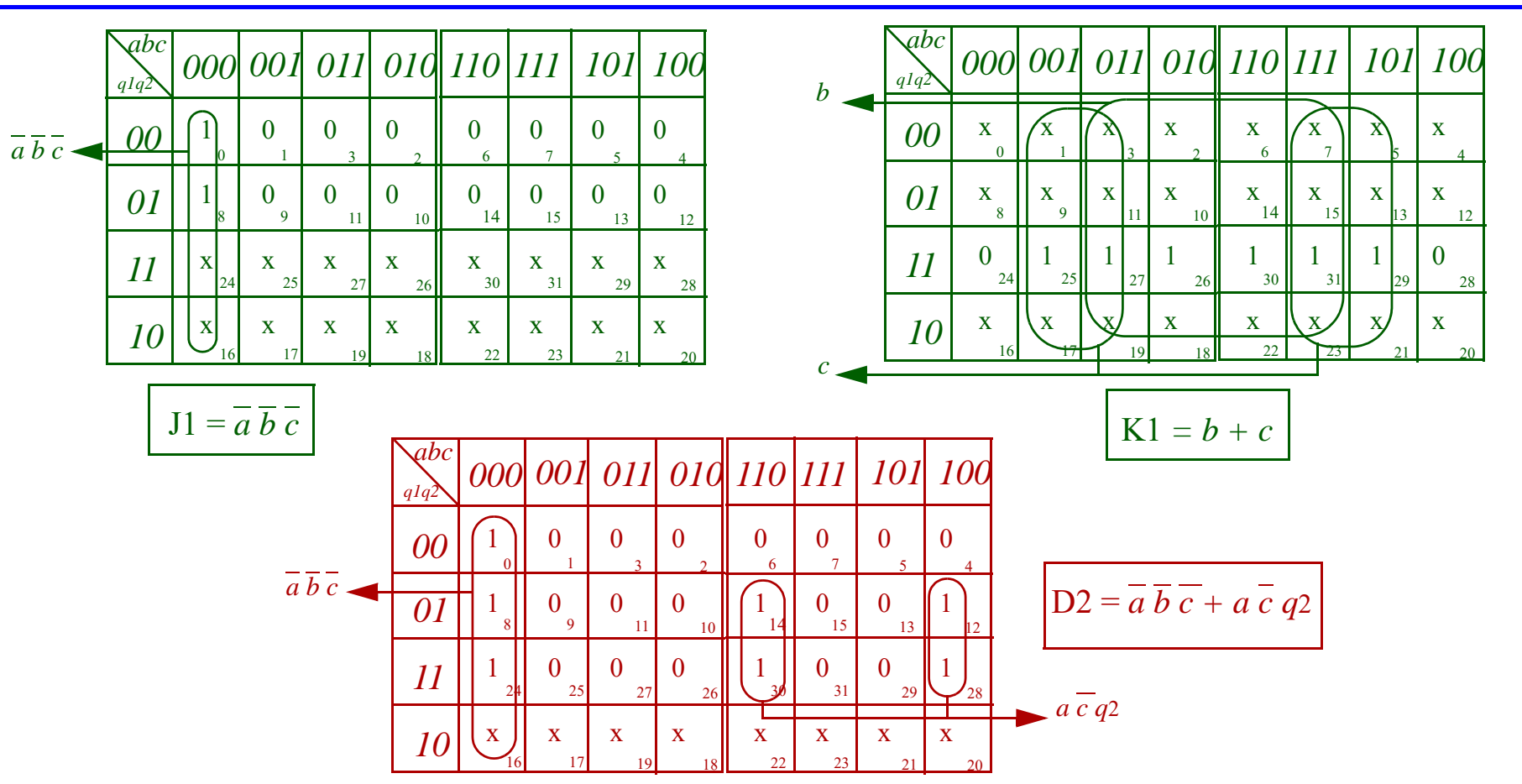
A { Rows 1-8 (00 state)
B { Rows 9-16 (01 state)
C { Rows 17-24 (11 state)

Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 1: Continuación

i) *Obtener las funciones de salida y las funciones de entrada de los flip-flops mínimas empleando las técnicas de minimización combinacional y obtener el diagrama lógico del sistema.*

Gracias a la asignación de estados realizada, las variables de salida coinciden con las variables de estado presente. Para obtener las funciones de entrada de las flip-flop J1, K1 y D2 emplearemos las técnicas de minimización de los mapas de Karnaugh, todas ellas son funciones de cinco variables.



Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 1: Continuación

$$J1 = \bar{a} \bar{b} \bar{c}$$

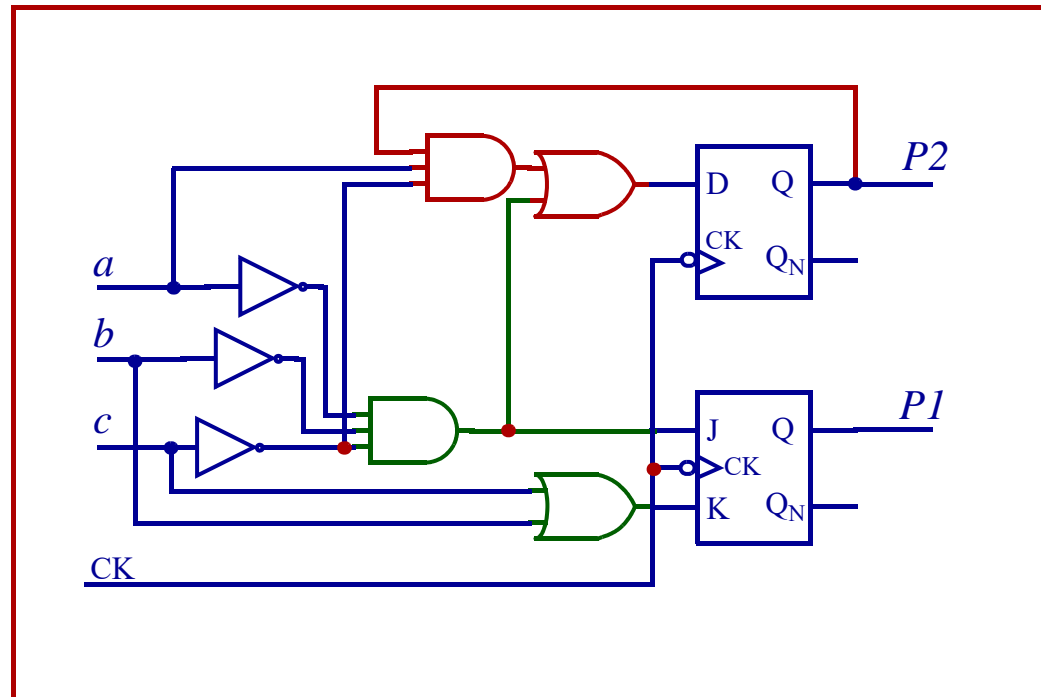
$$K1 = b + c$$

$$D2 = \bar{a} \bar{b} \bar{c} + a \bar{c} q2$$

$$P1 = q1$$

$$P2 = q2$$

El diagrama lógico del sistema resulta



Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 1: Continuación

j) Si hay combinaciones de las variables de estado no utilizadas, verificar que su aparición accidental en el sistema no lleva a situaciones erróneas irreversibles.

La combinación $q1q2 = 10$ de las variables de estado corresponden a un estado no alcanzable por el sistema (D).

Es conveniente, y en la práctica se hace necesario, comprobar que el sistema implementado es capaz de recuperar su funcionamiento normal si por alguna circunstancia fortuita éste alcanza dicho estado.

Estado presente $q1 \ q2$	Entradas $a \ b \ c$	Estado siguiente $Q1 \ Q2$	Entradas de FF $J1K1 \ D2$	Salidas $P1 \ P2$
1 0	0 0 0	1 1	1 0 1	1 0
1 0	0 0 1	0 0	0 1 0	1 0
1 0	0 1 0	0 0	0 1 0	1 0
1 0	0 1 1	0 0	0 1 0	1 0
1 0	1 0 0	1 0	0 0 0	1 0
1 0	1 0 1	0 0	0 1 0	1 0
1 0	1 1 0	0 0	0 1 0	1 0
1 0	1 1 1	0 0	0 1 0	1 0

D {

$$J1 = \bar{a} \bar{b} \bar{c} \quad K1 = b + c$$

$$D2 = \bar{a} \bar{b} \bar{c} + a \bar{c} q2$$

$$P1 = q1 \quad P2 = q2$$

Habría que modificar la función de salida, para evitar que la bomba P1 se active indebidamente.

$$P1 = q1 \cdot q2$$

El estado siguiente a D es un estado alcanzable para todas las combinaciones de entrada salvo para la combinación $abc = 100$. Gracias al cambio realizado en las salidas, esta circunstancia no resulta perjudicial.

Si el nivel de agua baja ($abc=000$), estaremos en la situación anterior y el sistema se recupera.

Y si el nivel de agua sube ($abc=110$), el estado siguiente será $q1q2 = 00$, ambas bombas estarán paradas, y el sistema se recupera puesto que permanecerán paradas hasta que el nivel baje por debajo del nivel N1.

Por lo tanto el sistema es capaz de recuperarse (sistema tolerante a fallos) y no es necesario realizar más modificaciones, salvo la de las funciones de salida para garantizar que en D ambas bombas esten paradas.

Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 1: Continuación

$$J1 = \bar{a} \bar{b} \bar{c}$$

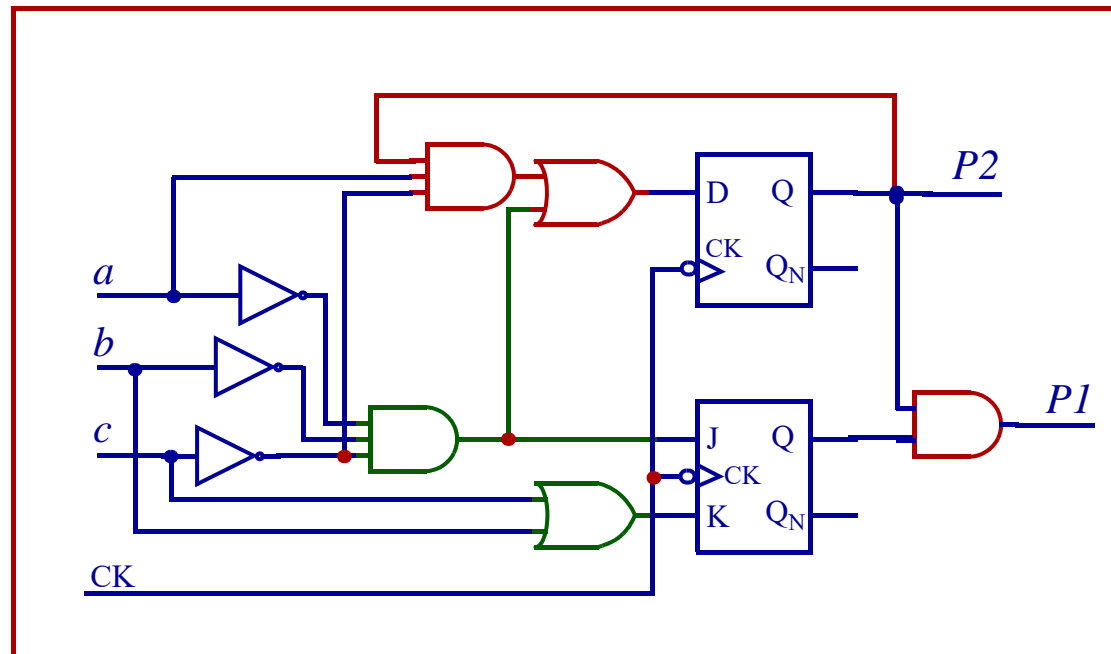
$$K1 = b + c$$

$$D2 = \bar{a} \bar{b} \bar{c} + a \bar{c} q2$$

$$P1 = q1 \cdot q2$$

$$P2 = q2$$

El diagrama lógico final del sistema resulta



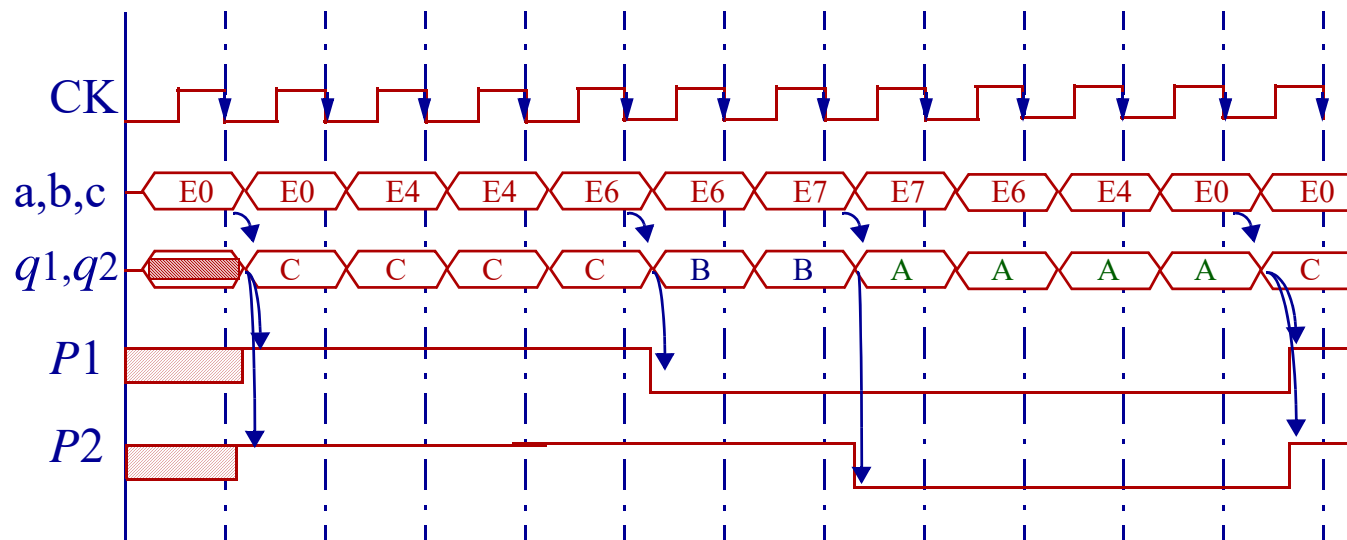
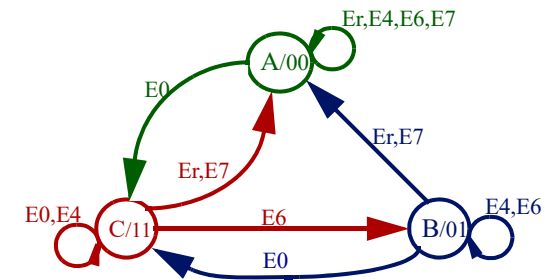
Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 1: Continuación. Conogramas (simulación)

Para verificar el funcionamiento del diseño mediante un simulador habría que excitar el circuito con todas las secuencias de entrada posibles. Una simulación exhaustiva supone considerar todas las imaginables. A continuación se muestran, a modo de ejemplo, los cronogramas correspondientes a algunas de ellas.

Sec. 1: Llenado y vaciado del depósito

E0 -> E4 -> E6 -> E7 -> E6 -> E4 -> E0

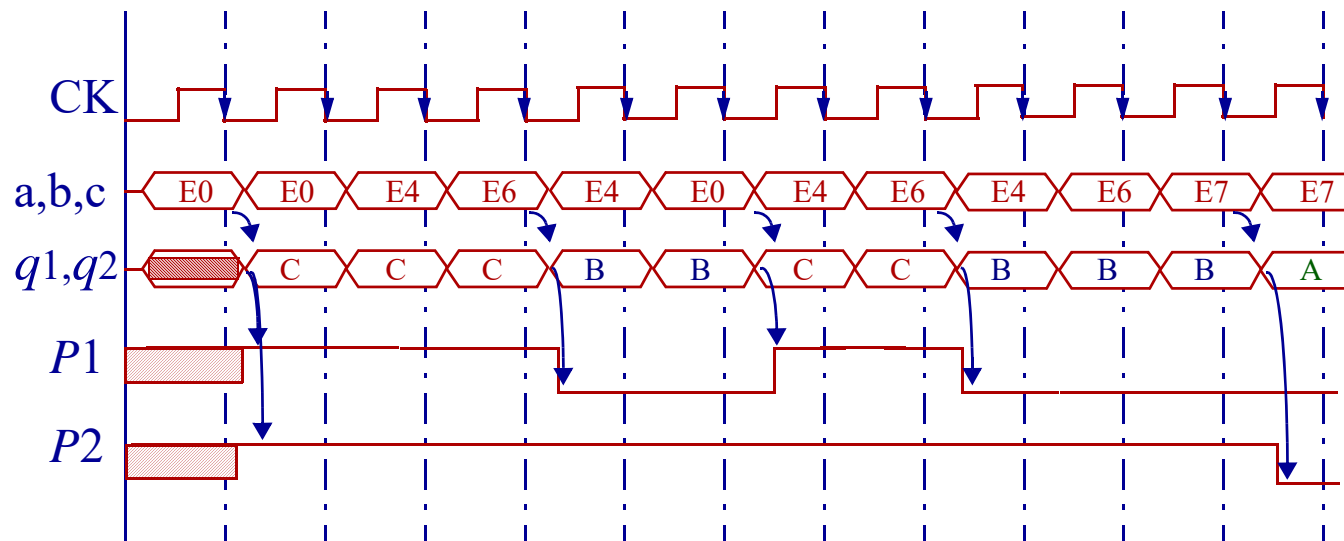
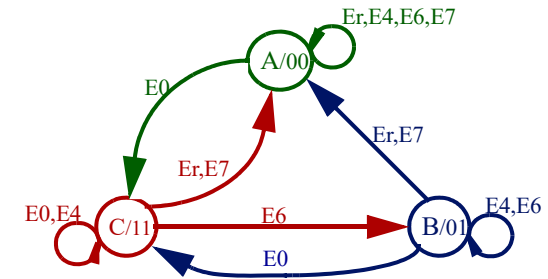


Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 1: Continuación. Conogramas (simulación)

Sec. 2: Llenado del depósito con subidas y bajadas intermedias

E0 -> E4 -> E6 -> E4 -> E0 -> E4 -> E6 -> E4 -> E6 -> E7





Escuela de Ingenierías Industriales

Titulación: Grado en Ingeniería Electrónica, Robótica y Mecatrónica

Asignatura: Electrónica Digital

Tema 8: Análisis y Síntesis de Sistemas Secuenciales Síncronos



TEMA 8: ANÁLISIS Y SÍNTESIS DE SISTEMAS SECUENCIALES SÍNCRONOS

8.1. Análisis de sistemas secuenciales síncronos.

8.1.1 Introducción. Modelo de Huffman para sistemas síncronos. Variables y ecuaciones booleanas.

8.1.2 Funciones de entrada de los biestables. Funciones de salida.

8.1.3 Tabla de estados: Tablas características de los biestables. Tabla de Transición de estados.

8.1.4 Diagramas de estados. Cronogramas.

8.1.5 Modelos de Mealy y de Moore para los sistemas secuenciales síncronos.

8.2. Diseño de sistemas secuenciales síncronos.

8.2.1 Tabla de excitación de los biestables.

8.2.2 Reducción de estados: Tablas de implicación. Estados equivalentes y estados compatibles.

8.2.3 Asignación de estados.

8.2.4 Ejemplos de diseño.

8.3. Diagramas de estados en HDL.

Reducción de tablas de transición de estados. Estados equivalentes.

- En un circuito secuencial la **reducción de coste** está ligada a la **reducción** tanto de **puertas lógicas** (sistema combinacional), como de **elementos de memoria**.

El problema de la reducción de elementos de memoria está relacionado directamente con el problema de encontrar el conjunto de estados mínimo que siga garantizando el correcto funcionamiento del sistema desde el punto de vista de los requerimientos de entrada/salida.

Los procedimientos para la reducción de estados toman como punto de partida la **Tabla de transición de estados**, y el concepto de estados equivalentes.

*“Se dice que **dos estados son equivalentes** si, para todas y cada una de las combinaciones de entrada, ambos proporcionan la misma salida y envían al circuito a un mismo estado siguiente, o a estados equivalentes entre sí.”*

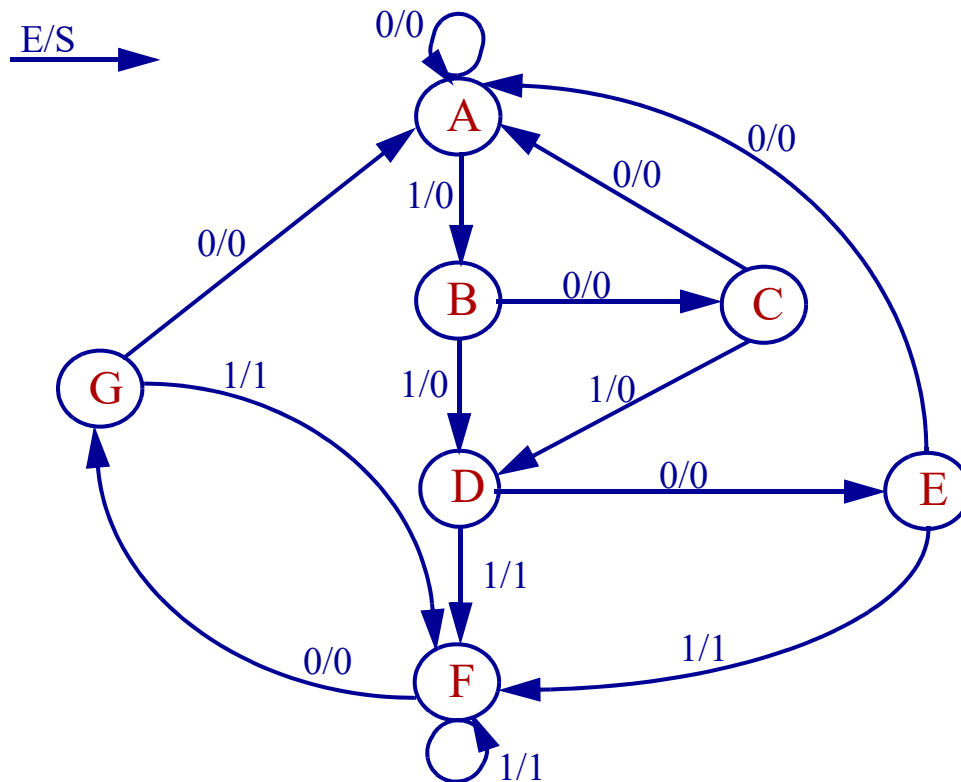
Cuando dos estados son equivalentes, uno de ellos puede ser eliminado de la tabla de transición de estados, sustituyendolo por el otro en cualquier ocurrencia, conservandose la relación entrada/salida en el sistema.

El algoritmo general de reducción estados establece una relación de equivalencia, en base a la cual se **clasifica cada estado de la tabla de estados en una clase de equivalencia**.

El conjunto de estados mínimo resulta de elegir el representante canónico de cada clase de equivalencia.

Reducción de tablas de transición de estados. Estados equivalentes.

● **Ejemplo 1:** Sea el sistema descrito por el siguiente diagrama de transición de estados.



Hay un número infinito de secuencias que pueden aplicarse al sistema, cada una de ellas dará como resultado una secuencia de salida.

Supongamos que el sistema está inicialmente en el estado A y que se aplica la secuencia

01010110100,

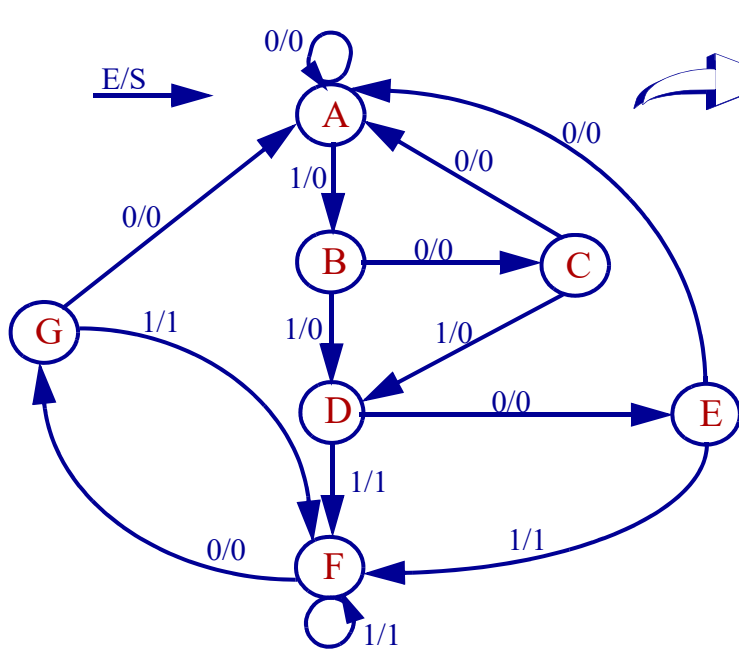
siguiendo el diagrama se tendrá la siguiente secuencia de estados y de salida:

Estado:	A	A	B	C	D	E	F	F	G	F	G	A
Entrada:	0	1	0	1	0	1	1	0	1	0	0	
Salida	0	0	0	0	0	1	1	0	1	0	0	

El problema de simplificación consiste en encontrar un sistema secuencial que genere la misma secuencia de salida pero que contenga un menor número de estados.

Reducción de tablas de transición de estados. Estados equivalentes.

● Ejemplo 1: Reducir el número de estados del sistema.



Ep	Es		Salida S	
	X=0	X=1	X=0	X=1
A	A	B	0	0
B	C	D	0	0
C	A	D	0	0
D	E	F	0	1
E	A	F	0	1
F	G	F	0	1
G	A	F	0	1

E y G son equivalentes:

$E \xrightarrow{0/0} A$

$E \xrightarrow{1/1} F$

y

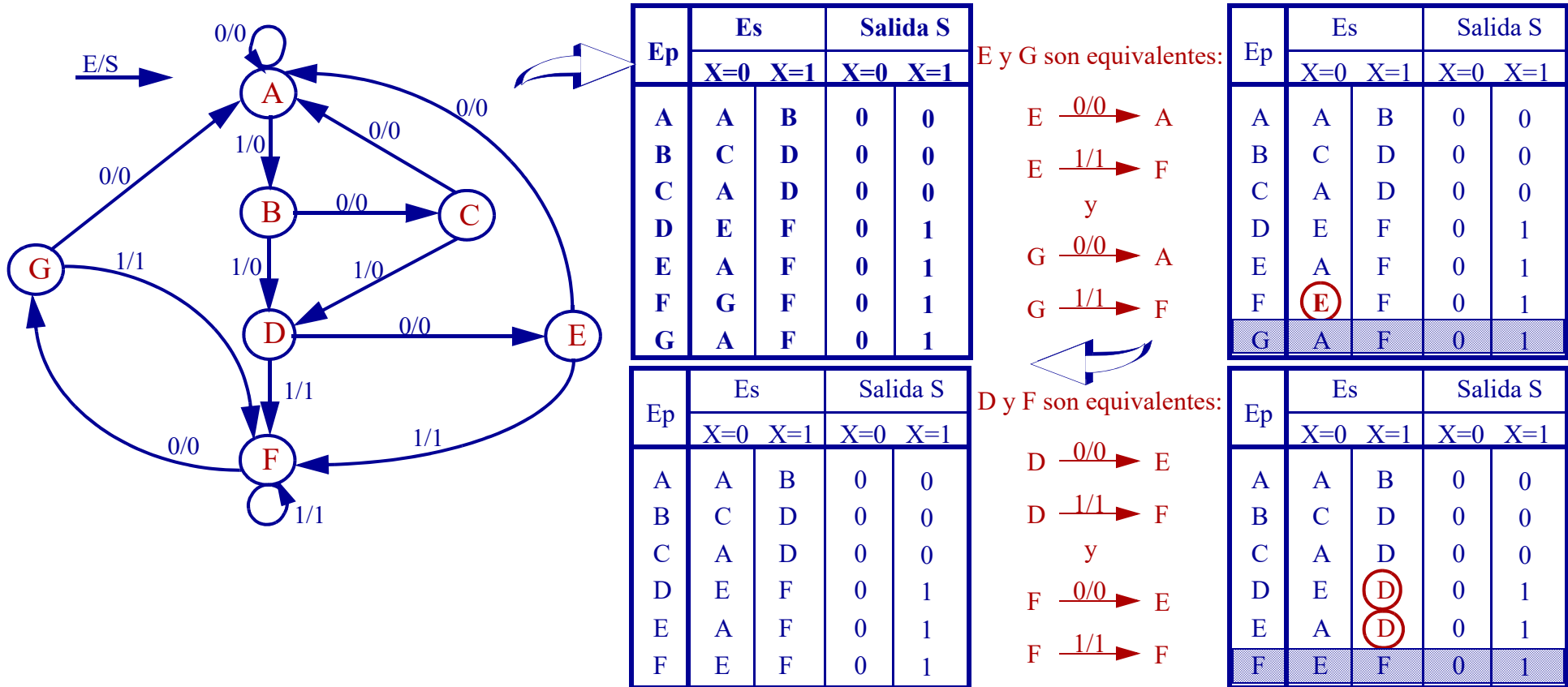
$G \xrightarrow{0/0} A$

$G \xrightarrow{1/1} F$

Ep	Es		Salida S	
	X=0	X=1	X=0	X=1
A	A	B	0	0
B	C	D	0	0
C	A	D	0	0
D	E	F	0	1
E	A	F	0	1
F	E	F	0	1
G	A	F	0	1

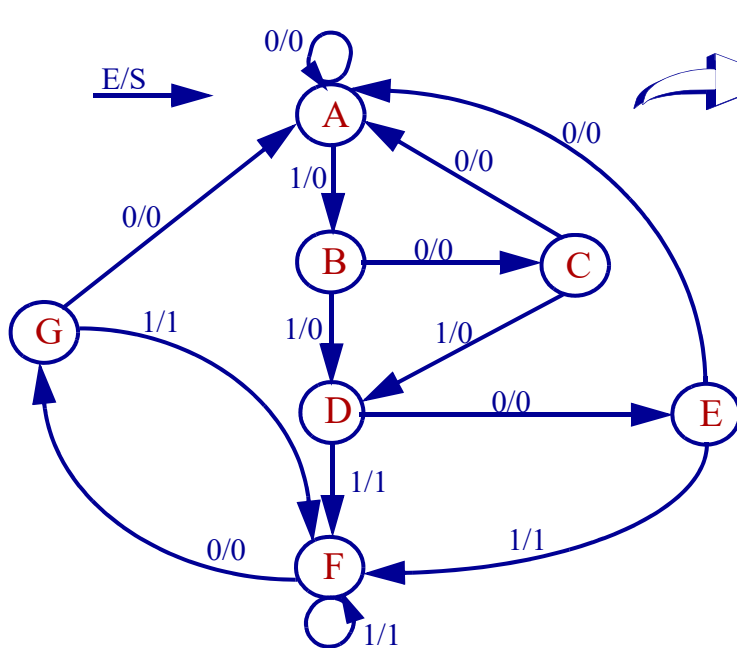
Reducción de tablas de transición de estados. Estados equivalentes.

● Ejemplo 1: Reducir el número de estados del sistema.



Reducción de tablas de transición de estados. Estados equivalentes.

● Ejemplo 1: Reducir el número de estados del sistema.



Ep	Es		Salida S	
	X=0	X=1	X=0	X=1
A	A	B	0	0
B	C	D	0	0
C	A	D	0	0
D	E	F	0	1
E	A	F	0	1
F	G	F	0	1
G	A	F	0	1

E y G son equivalentes:

E $\xrightarrow{0/0}$ A

E $\xrightarrow{1/1}$ F

y

G $\xrightarrow{0/0}$ A

G $\xrightarrow{1/1}$ F

Ep	Es		Salida S	
	X=0	X=1	X=0	X=1
A	A	B	0	0
B	C	D	0	0
C	A	D	0	0
D	E	F	0	1
E	A	F	0	1
F	E	F	0	1

D y F son equivalentes:

D $\xrightarrow{0/0}$ E

D $\xrightarrow{1/1}$ F

y

F $\xrightarrow{0/0}$ E

F $\xrightarrow{1/1}$ F

Ep	Es		Salida S	
	X=0	X=1	X=0	X=1
A	A	B	0	0
B	C	D	0	0
C	A	D	0	0
D	E	F	0	1
E	A	F	0	1
F	E	F	0	1
G	A	F	0	1

Ep	Es		Salida S	
	X=0	X=1	X=0	X=1
A	A	B	0	0
B	C	D	0	0
C	A	D	0	0
D	E	D	0	1
E	A	D	0	1
F	E	F	0	1

Ep	Es		Salida S	
	X=0	X=1	X=0	X=1
A	A	B	0	0
B	C	D	0	0
C	A	D	0	0
D	E	D	0	1
E	A	D	0	1

Tabla de transición de estados reducida resulta

Reducción de tablas de transición de estados. Estados equivalentes.

● Ejemplo 1: Reducir el número de estados del sistema.

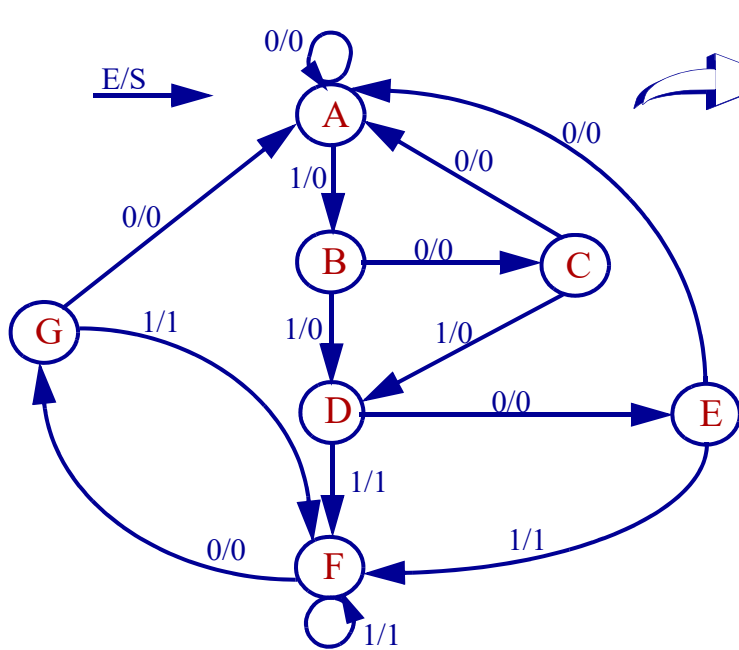
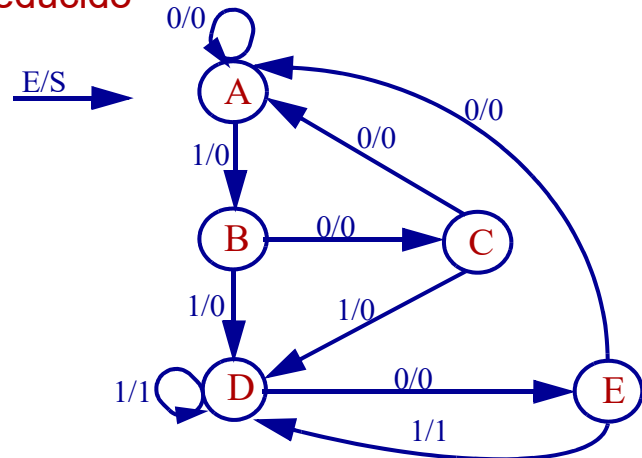


Diagrama de Transición de estados reducido



Ep	Es		Salida S	
	X=0	X=1	X=0	X=1
A	A	B	0	0
B	C	D	0	0
C	A	D	0	0
D	E	F	0	1
E	A	F	0	1
F	G	F	0	1
G	A	F	0	1

E y G son equivalentes:

E $\xrightarrow{0/0}$ A

E $\xrightarrow{1/1}$ F

y

G $\xrightarrow{0/0}$ A

G $\xrightarrow{1/1}$ F

Ep	Es		Salida S	
	X=0	X=1	X=0	X=1
A	A	B	0	0
B	C	D	0	0
C	A	D	0	0
D	E	F	0	1
E	A	F	0	1
F	E	F	0	1

D y F son equivalentes:

D $\xrightarrow{0/0}$ E

D $\xrightarrow{1/1}$ F

y

F $\xrightarrow{0/0}$ E

F $\xrightarrow{1/1}$ F

Ep	Es		Salida S	
	X=0	X=1	X=0	X=1
A	A	B	0	0
B	C	D	0	0
C	A	D	0	0
D	E	F	0	1
E	A	F	0	1
F	E	F	0	1
G	A	F	0	1

Ep	Es		Salida S	
	X=0	X=1	X=0	X=1
A	A	B	0	0
B	C	D	0	0
C	A	D	0	0
D	E	D	0	1
E	A	D	0	1
F	E	F	0	1

Ep	Es		Salida S	
	X=0	X=1	X=0	X=1
A	A	B	0	0
B	C	D	0	0
C	A	D	0	0
D	E	D	0	1
E	A	D	0	1

Tabla de transición de estados reducida resulta

Reducción de tablas de transición de estados. Estados equivalentes.

● Ejemplo 1: Continuación

Diagrama original

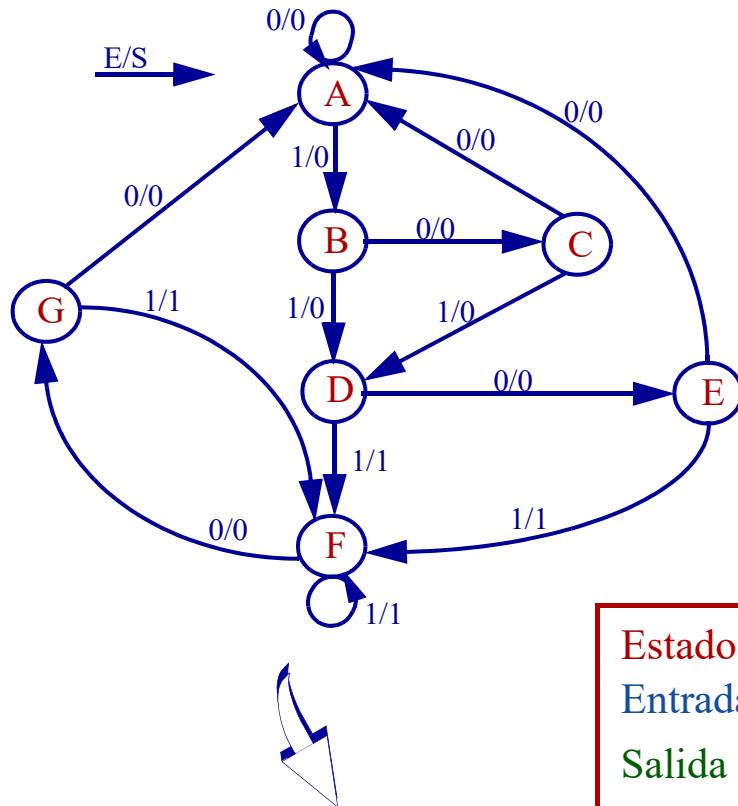
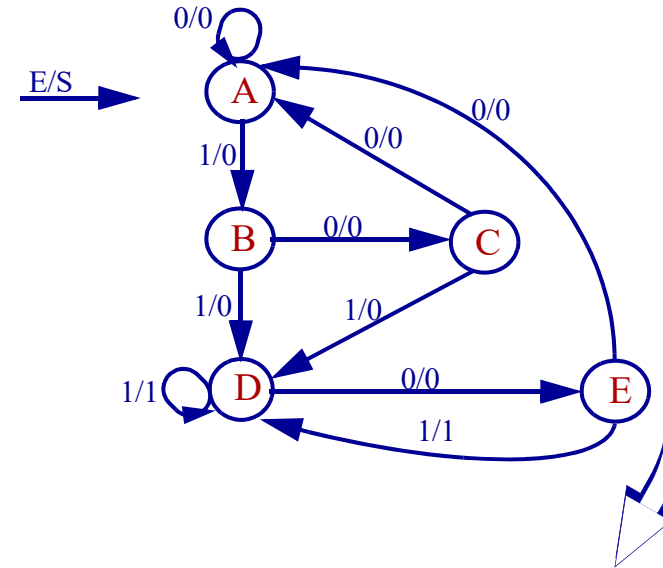


Diagrama reducido



Estado:	A	A	B	C	D	E	D	D	E	D	E	A
Entrada:	0	1	0	1	0	1	1	0	1	0	0	
Salida	0	0	0	0	0	1	1	0	1	0	0	

Estado:	A	A	B	C	D	E	F	F	G	F	G	A
Entrada:	0	1	0	1	0	1	1	0	1	0	0	
Salida	0	0	0	0	0	1	1	0	1	0	0	

Se observa que la respuesta del sistema es la misma.

Reducción de tablas de transición de estados. Tablas de implicación.

- Al buscar estados equivalentes sobre una **Tabla de transición de estados** es frecuente encontrar dos estados (A y B), que proporcionan la misma salida para todas las combinaciones de entrada pero transitan a estados siguientes distintos (E y D). Dichos estados (A y B) no son equivalentes entre sí, salvo que los correspondientes estados siguientes lo sean, en cuyo caso también lo serán los primeros. En este caso, se dice que la pareja formada por los segundos (E y D) implica a los primeros (A y B). Si se cumple que (E,D) implica a (A,B) y que (A,B) implica a (E,D), entonces los cuatro estados son equivalentes entre sí.
- El proceso de búsqueda de estados equivalentes o parejas de estados implicantes puede sistematizarse con la ayuda de la **Tabla de implicación**.

Si A,B,C,D,E,F y G son los estados de un sistema, podemos contruir la siguiente tabla, en la que cada celda representa una pareja de estado, y relizar el siguiente algoritmo de búsqueda de estados equivalentes:

B						
C						
D						
E						
F						
G						
	A	B	C	D	E	F

Tabla de implicación

- Cada celda se marca con:
 - X Si los estados correspondientes no son equivalentes.
 - v Si los estados correspondientes son equivalentes.
- Si existen parejas de estados implicantes se anotan en la celda correspondiente.
 - El proceso de marcar celdas repite y se detiene cuando todas ellas estén marcadas con X o con v.

Reducción de tablas de transición de estados. Tablas de implicación.

● Ejemplo 1: Reducir el número de estados del sistema.

Tabla de T. de estados

Ep	Es		Salida S	
	X=0	X=1	X=0	X=1
A	A	B	0	0
B	C	D	0	0
C	A	D	0	0
D	E	F	0	1
E	A	F	0	1
F	G	F	0	1
G	A	F	0	1

Tabla de implicación

B						
C						
D						
E						
F						
G						
	A	B	C	D	E	F

- Se marcan con **X** las celdas correspondientes a parejas de estados no equivalentes por tener distinta salida.

- Se marcan con **v** las celdas correspondientes a parejas de estados equivalentes.

- Se rellenan el resto de celdas con implicaciones.

B	A,C B,D					
C	B,D	A,C				
D	X	X	X			
E	X	X	X	A,E		
F	X	X	X	E,G	A,G	
G	X	X	X	A,E	v	A,G
	A	B	C	D	E	F

Reducción de tablas de transición de estados. Tablas de implicación.

● Ejemplo 1: Reducir el número de estados del sistema.

Tabla de T. de estados

Ep	Es		Salida S	
	X=0	X=1	X=0	X=1
A	A	B	0	0
B	C	D	0	0
C	A	D	0	0
D	E	F	0	1
E	A	F	0	1
F	G	F	0	1
G	A	F	0	1

Tabla de implicación

B						
C						
D						
E						
F						
G						
	A	B	C	D	E	F

- Se marcan con **X** las celdas correspondientes a parejas de estados no equivalentes por tener distinta salida.

- Se marcan con **v** las celdas correspondientes a parejas de estados equivalentes.

- Se rellenan el resto de celdas con implicaciones.

- Se marcan con **X** las celdas correspondientes a parejas de estados que no son equivalentes por tener como implicantes parejas de estados no equivalentes.



B	A,C B,D					
C	B,D	A,C				
D	X	X	X			
E	X	X	X	A,E		
F	X	X	X	E,G	A,G	
G	X	X	X	A,E	v	A,G
	A	B	C	D	E	F

Reducción de tablas de transición de estados. Tablas de implicación.

● Ejemplo 1: Reducir el número de estados del sistema.

Tabla de T. de estados

Ep	Es		Salida S	
	X=0	X=1	X=0	X=1
A	A	B	0	0
B	C	D	0	0
C	A	D	0	0
D	E	F	0	1
E	A	F	0	1
F	G	F	0	1
G	A	F	0	1

Tabla de implicación

B						
C						
D						
E						
F						
G						
	A	B	C	D	E	F

- Se marcan con **X** las celdas correspondientes a parejas de estados no equivalentes por tener distinta salida.

- Se marcan con **v** las celdas correspondientes a parejas de estados equivalentes.

- Se rellenan el resto de celdas con implicaciones.

- Se marcan con **X** las celdas correspondientes a parejas de estados que no son equivalentes por tener como estados implicantes parejas de estados no equivalentes.

- Se marcan con **v** las celdas correspondientes a parejas de estados que son equivalentes por tener como estados implicantes parejas de estados equivalentes.

- El proceso termina cuando todas las celdas de la **Tabla de implicación** están marcadas con **X** o con **v**.

B	A C B D					
C	B D	A C				
D	X	X	X			
E	X	X	X	A E		
F	X	X	X	E G	A G	
G	X	X	X	E G	v	A G
	A	B	C	D	E	F

Reducción de tablas de transición de estados. Tablas de implicación.

● Ejemplo 1: Reducir el número de estados del sistema.

Tabla de T. de estados

Ep	Es		Salida S	
	X=0	X=1	X=0	X=1
A	A	B	0	0
B	C	D	0	0
C	A	D	0	0
D	E	F	0	1
E	A	F	0	1
F	G	F	0	1
G	A	F	0	1

Tabla de implicación

B	A C B ,D					
C	B D	A C				
D	X	X	X			
E	X	X	X	A E		
F	X	X	X	A E,G	A G	
G	X	X	X	A E	A G	A G
	A	B	C	D	E	F

Se tiene que E es equivalente a G,
y que F lo es a D, por tanto:

En el conjunto de estados {A,B,C,D,E,F,G}
pueden establecerse **cinco** clases de equivalencia

{A},{B},{C},{D,F} y {E,G}.

Escogiendo un representante de cada clase de equivalencia se tiene el conjunto de estados reducidos mínimo, en este caso cinco estados, a partir de los cuales se contruye la tabla de estados reducida: {A,B,C,D,E,}

Tabla de T. de estados reducida

Ep	Es		Salida S	
	X=0	X=1	X=0	X=1
A	A	B	0	0
B	C	D	0	0
C	A	D	0	0
D	E	D	0	1
E	A	D	0	1

Reducción de tablas de transición de estados. Tablas de implicación.

● Ejemplo 2: Reducir el número de estados del sistema.

Tabla de T. de estados

Ep	Es		Salida S	
	X=0	X=1	X=0	X=1
A	D	B	0	0
B	E	A	0	0
C	G	F	0	1
D	A	D	1	0
E	A	D	1	0
F	C	B	0	0
G	A	E	1	0

Tabla de implicación

B						
C						
D						
E						
F						
G						
	A	B	C	D	E	F

Reducción de tablas de transición de estados. Tablas de implicación.

● Ejemplo 2: Reducir el número de estados del sistema.

Tabla de T. de estados

Ep	Es		Salida S	
	X=0	X=1	X=0	X=1
A	D	B	0	0
B	E	A	0	0
C	G	F	0	1
D	A	D	1	0
E	A	D	1	0
F	C	B	0	0
G	A	E	1	0

Tabla de implicación

B	D,E					
C	X	X				
D	X	X	X			
E	X	X	X	v		
F	X	X	X	X	X	
G	X	X	X	D,E	D,E	X
	A	B	C	D	E	F

Se tiene obtenen las siguientes parejas de estados equivalentes:

(A,B) , (D,E), (D,G) y (E,G)

De (D,E), (D,G) y (E,G) se tiene que (D,E,G) son tres estados equivalentes.

En el conjunto de estados {A,B,C,D,E,F,G} pueden establecerse cinco clases de equivalencia:

{A,B}, {C}, {D,E,G} y {F}.

Escogiendo un representante de cada clase se tiene el conjunto de estados mínimo: {A, C, D, F}.

La **Tabla de transición de estados reducida** resulta:

Ep	Es		Salida S	
	X=0	X=1	X=0	X=1
A	D	A	0	0
C	D	F	0	1
D	A	D	1	0
F	C	A	0	0

Reducción de tablas de T. E. incompletamente especificadas. Estados compatibles.

- Cuando en un sistema secuencial ciertas combinaciones o secuencias de entrada no están especificadas, porque no pueden ocurrir nunca, debido a restricciones internas o externas, o a que carecen de sentido para el funcionamiento normal del sistema; la Tabla de transición de estados resultante estará **incompletamente especificada**.

Los estados o transiciones de estado no especificadas pueden tratarse como indiferencias y pueden ser usadas para reducir el número de estados alcanzables.

Ejemplo de Tabla de transición de estados incompletamente especificada.

Ep	Es				Salida S			
	x y				x y			
	00	01	11	10	00	01	11	10
a	c	a	b	-	-	0	-	-
b	-	a	b	e	-	-	1	-
c	c	a	-	d	0	-	-	-
d	c	-	b	d	-	-	-	0
e	f	-	b	e	-	-	-	1
f	f	a	-	e	1	-	-	-

La simplificación de tablas de transición de estados incompletamente especificadas se basa en el concepto de **estados compatibles**:

“Se dice que dos estados son compatibles, si para cada combinación de entrada especificada para ambos el sistema produce la misma salida, y sus estados siguientes, cuando estén ambos especificados, son compatibles entre sí.”

Reducción de tablas de T. E. incompletamente especificadas. Estados compatibles.

- El procedimiento a seguir para encontrar estados compatibles y reducir la tabla de transición de estados es el siguiente:
 - 1º Se identifican las parejas de estados compatibles usando una tabla de implicación.
 - 2º Se construye el conjunto de **conjuntos de máximos estados compatibles** CMC_E .
Este es un conjunto cuyos elementos son conjuntos de estados compatibles, cada uno de los cuales contiene el máximo nº de estados compatibles entre sí.
 - 3º Se construye un conjunto que contenga el número mínimo de conjuntos de máximos estados compatibles $MCMC_E$, y que cumpla las **condiciones de cobertura y cierre**.
 - La condición de cobertura exige que todos los estados del sistema de partida pertenezca a alguno de los conjuntos incluidos en $MCMC_E$.
 - La condición de cierre exige que cualquier pareja de estados, implicados por parejas de estados que pertenecen a un mismo conjunto de estados compatibles, debe, a su vez, pertenecer al mismo conjunto de estados compatibles.
 - 4º Se elige un representante de cada elemento del $MCMC_E$, para formar la tabla de transición de estados reducida.
 - 5º Finalmente, se completa la tabla reducida agrupando todas las situaciones que están ahora especificadas para cada conjunto de estados compatibles.

Reducción de tablas de T. E. incompletamente especificadas. Estados compatibles.

● Ejemplo 1: Reducir el número de estados del sistema.

Tabla de T. de estados

Ep	Es				Salida S			
	x y				x y			
	00	01	11	10	00	01	11	10
a	c	a	b	-	-	0	-	-
b	-	a	b	e	-	-	1	-
c	c	a	-	d	0	-	-	-
d	c	-	b	d	-	-	-	0
e	f	-	b	e	-	-	-	1
f	f	a	-	e	1	-	-	-

Tabla de implicación

b					
c					
d					
e					
f					
	a	b	c	d	e



b					
c					
d					
e				X	
f			X		
	a	b	c	d	e

b	v				
c	v				
d	v		v		
e		v		X	
f		v	X		v
	a	b	c	d	e



b	v				
c	v	e,d			
d	v	e,d	v		
e	c,f	v	e,d c,f	X	
f	c,f	v	X	e,d c,f	v
	a	b	c	d	e



- Se marcan con **X** las celdas correspondientes a parejas de estados no compatibles por tener distinta salida para combinaciones de entrada especificadas en ambos estados
- Se marcan con **v** las celdas correspondientes a parejas de estados compatibles.
- Se rellenan el resto de celdas con implicaciones.

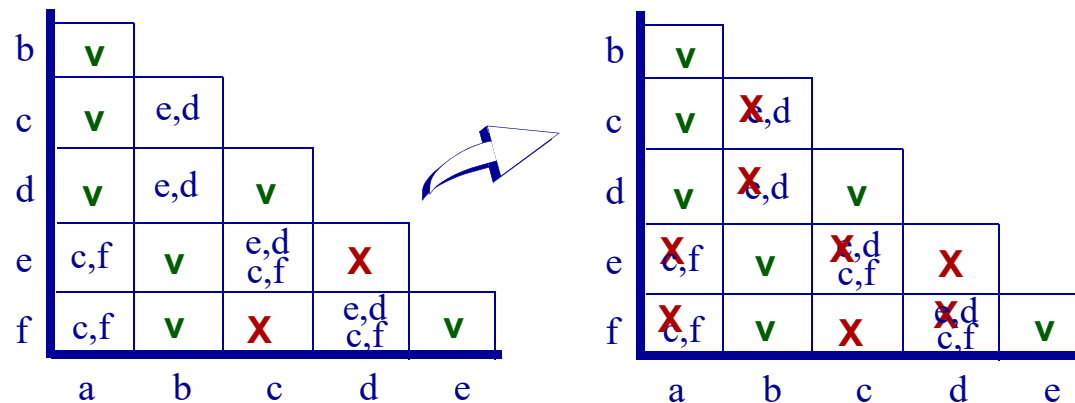
Reducción de tablas de T. E. incompletamente especificadas. Estados compatibles.

● Ejemplo 1: Continuación.

Tabla de T. de estados

Ep	Es				Salida S			
	x y				x y			
	00	01	11	10	00	01	11	10
a	c	a	b	-	-	0	-	-
b	-	a	b	e	-	-	1	-
c	c	a	-	d	0	-	-	-
d	c	-	b	d	-	-	-	0
e	f	-	b	e	-	-	-	1
f	f	a	-	e	1	-	-	-

- Se marcan con **X** las celdas correspondientes a parejas de estados no compatibles por tener como implicación parejas de estados no compatibles.
- Se marcan con **v** las celdas correspondientes a parejas de estados compatibles por tener como implicación parejas de estados compatibles.
- El proceso termina cuando todas las celdas de la **Tabla de implicación** están marcadas con **X** o con **v**.



Reducción de tablas de T. E. incompletamente especificadas. Estados compatibles.

● Ejemplo 1: Continuación.

- Se han obtenido las siguientes parejas de estados compatibles:

(a,b) , (a,c), (a,d), (b,e), (b,f), (c,d) y (e,f).

- De (a,c), (a,d) y (c,d) se tiene que (a,c,d) son tres estados compatibles entre sí dos a dos.

- De (b,e), (b,f) y (e,f) se tiene que (b,e,f) son tres estados compatibles entre sí dos a dos.

- Del conjunto $E=\{a,b,c,d,e,f\}$ se obtiene el siguiente conjunto de conjuntos máximos compatibles :

$$CMC_E = \{\{a,b\}, \{a,c,d\}, \{b,e,f\}\}.$$

- Escogiendo el conjunto $MCMC_E\{\{a,c,d\}, \{b,e,f\}\}$ se cumplen las condiciones de cobertura y cierre:

Cobertura: Todos los elementos de E pertenecen al menos a un conjunto de estados compatibles máximos.

Cierre: Se cumple, pues ninguna de las parejas (a,c), (a,d),(c,d); ni (b,e), (b,f), (e,f) tienen parejas implicantes.

- Se elige un representante por cada elemento de $MCMC_E$ para construir la **Tabla de estados reducida**.

b	v				
c	v	X,d			
d	v	X,d	v		
e	X,f	v	X,d	X	
f	X,f	v	X	X,d	v
	a	b	c	d	e

Tabla de T. de estados

Ep	Es				Salida S			
	x y				x y			
	00	01	11	10	00	01	11	10
a	c	a	b	-	-	0	-	-
b	-	a	b	e	-	-	1	-
c	c	a	-	d	0	-	-	-
d	c	-	b	d	-	-	-	0
e	f	-	b	e	-	-	-	1
f	f	a	-	e	1	-	-	-

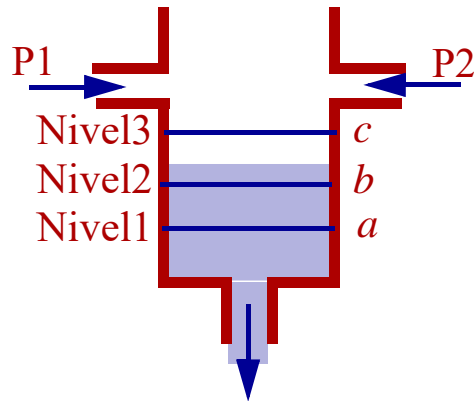


Ep	Es				Salida S			
	x y				x y			
	00	01	11	10	00	01	11	10
a	a	a	b	a	0	0	-	0
b	b	a	b	b	1	-	1	1

Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 2:

Diseñar un circuito digital síncrono que permita controlar las bombas P1 y P2 de acuerdo con las especificaciones del problema.



Cod	$a \ b \ c$	Sign.	Acción	
E0	0 0 0	$N < N1$	$P1 = 1$	$P2 = 1$
Er		No se da	$P1 = 0$	$P2 = 0$
E4	1 0 0	$N1 \leq N < N2$	$P1_{n+1} = P1_n$	$P2_{n+1} = P2_n$
E6	1 1 0	$N2 \leq N < N3$	$P1 = 0$	$P2_{n+1} = P2_n$
E7	1 1 1	$N \geq N3$	$P1 = 0$	$P2 = 0$

Vamos a resolver el problema del Ejemplo de diseño 1, de forma que obtengamos una implementación que siga el modelo de una máquina de estados finitos del tipo Mealy.

En esta solución, los estados memorizan el nivel de agua alcanzado en el depósito y no el estado de las variables de salida. Por otra parte, la evolución entre estados y la salida en cada instante, la determina la información que proporcionan los sensores.

Supondremos además que la frecuencia del reloj empleado es lo suficientemente alta para que el sistema que se diseña sea capaz de detectar la máxima velocidad de cambio del nivel de agua en el depósito, esto es $f > dN/dt$. Esto permitirá considerar que a la entrada no van a aparecer secuencias que supongan una transición entre niveles de agua no consecutivos.

Asumiremos entonces en el diseño que para esas secuencias de entrada “no esperadas”, el estado siguiente al que se evoluciona es una indiferencia, aunque al igual que en la solución anterior forzaremos que, dado el caso, la salida del sistema sea tal que los motores se detengan.

Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 2: Continuación

Cod	Significado	Acción
A	$N < N1$	Con entrada E0 o E4 $P1 = 1$ $P2 = 1$
B	$N1 \leq N < N2$ subiendo desde A	

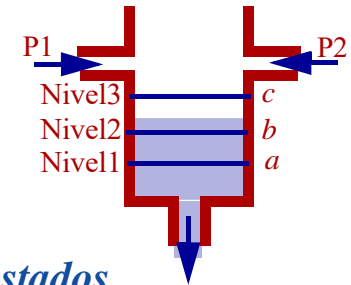
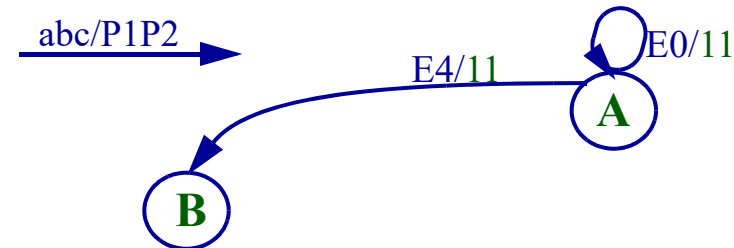


Diagrama de transición de estados



Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 2: Continuación

Cod	Significado	Acción
A	$N < N1$	Con entrada E0 o E4 $P1 = 1$ $P2 = 1$
B	$N1 \leq N < N2$ subiendo desde A	Con entrada E4 $P1 = 1$ $P2 = 1$ Con entrada E6 $P1 = 0$ $P2 = 1$
C	$N2 \leq N < N3$ subiendo desde B	

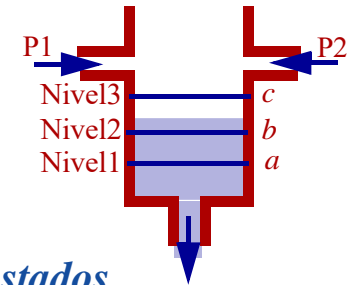
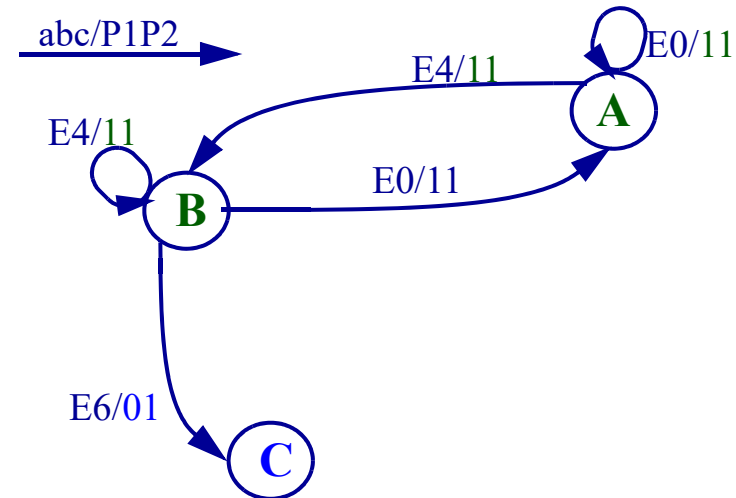


Diagrama de transición de estados



Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 2: Continuación

Cod	Significado	Acción
A	$N < N1$	Con entrada E0 o E4 $P1 = 1$ $P2 = 1$
B	$N1 \leq N < N2$ subiendo desde A	Con entrada E4 $P1 = 1$ $P2 = 1$ Con entrada E6 $P1 = 0$ $P2 = 1$
C	$N2 \leq N < N3$ subiendo desde B	Con entrada E4 o E6 $P1 = 0$ $P2 = 1$ Con entrada E7 $P1 = 0$ $P2 = 0$
D	$N \geq N3$, subiendo desde C	
G	$N1 \leq N < N2$, bajando desde C	

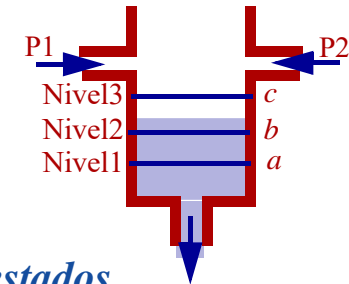
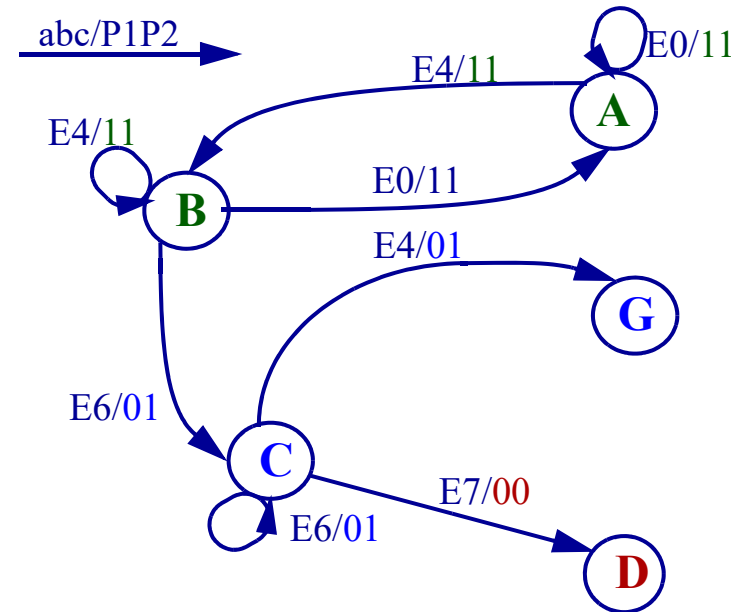


Diagrama de transición de estados



Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 2: Continuación

Cod	Significado	Acción
A	$N < N1$	Con entrada E0 o E4 $P1 = 1$ $P2 = 1$
B	$N1 \leq N < N2$ subiendo desde A	Con entrada E4 $P1 = 1$ $P2 = 1$ Con entrada E6 $P1 = 0$ $P2 = 1$
C	$N2 \leq N < N3$ subiendo desde B	Con entrada E4 o E6 $P1 = 0$ $P2 = 1$ Con entrada E7 $P1 = 0$ $P2 = 0$
D	$N \geq N3$, subiendo desde C	Con entrada E6 o E7 $P1 = 0$ $P2 = 0$
E	$N2 \leq N < N3$, bajando desde D	
G	$N1 \leq N < N2$, bajando desde C	

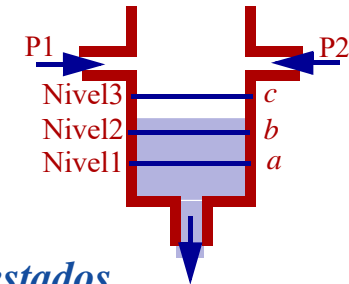
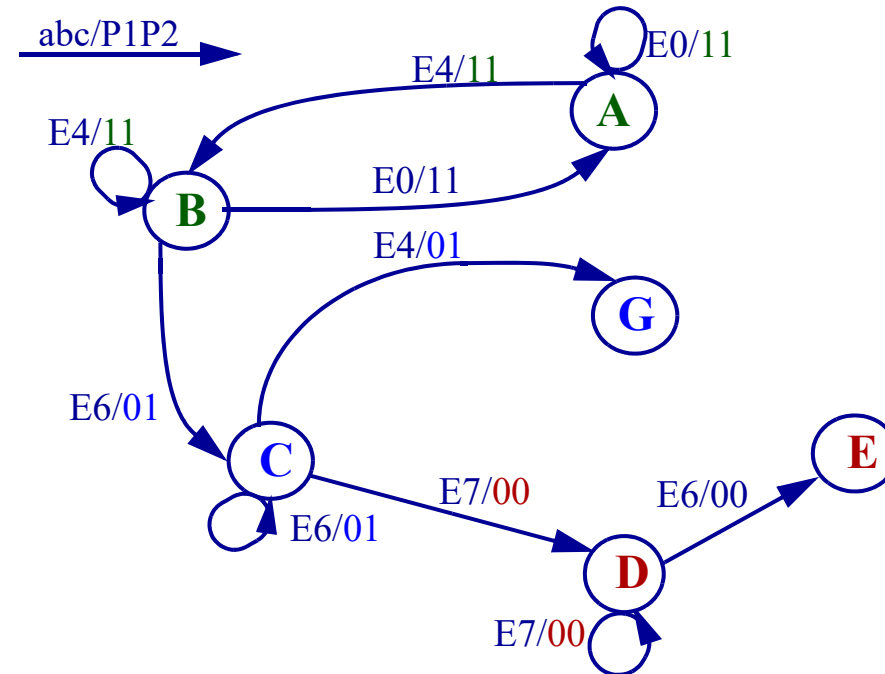


Diagrama de transición de estados



Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 2: Continuación

Cod	Significado	Acción
A	$N < N1$	Con entrada E0 o E4 $P1 = 1$ $P2 = 1$
B	$N1 \leq N < N2$ subiendo desde A	Con entrada E4 $P1 = 1$ $P2 = 1$ Con entrada E6 $P1 = 0$ $P2 = 1$
C	$N2 \leq N < N3$ subiendo desde B	Con entrada E4 o E6 $P1 = 0$ $P2 = 1$ Con entrada E7 $P1 = 0$ $P2 = 0$
D	$N \geq N3$, subiendo desde C	Con entrada E6 o E7 $P1 = 0$ $P2 = 0$
E	$N2 \leq N < N3$, bajando desde D	Con entrada E4 o E6 $P1 = 0$ $P2 = 0$
F	$N1 \leq N < N2$, bajando desde E	
G	$N1 \leq N < N2$, bajando desde C	

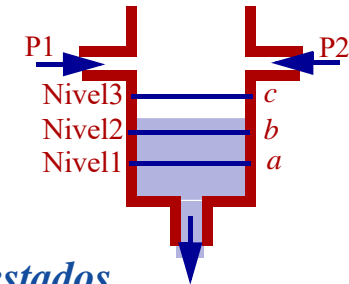
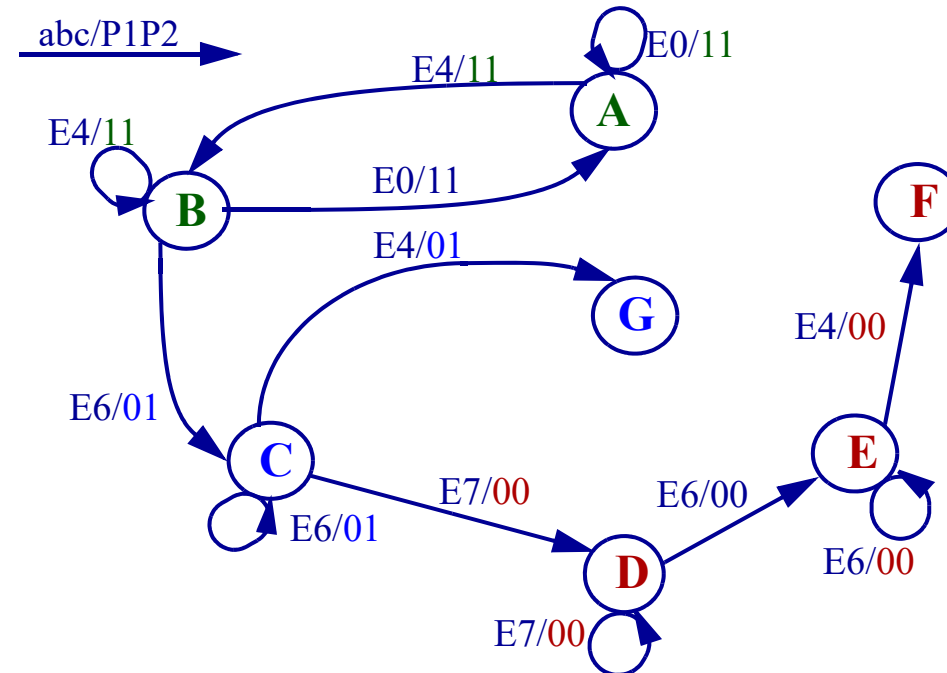


Diagrama de transición de estados



Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 2: Continuación

Cod	Significado	Acción
A	$N < N1$	Con entrada E0 o E4 $P1 = 1$ $P2 = 1$
B	$N1 \leq N < N2$ subiendo desde A	Con entrada E4 $P1 = 1$ $P2 = 1$ Con entrada E6 $P1 = 0$ $P2 = 1$
C	$N2 \leq N < N3$ subiendo desde B	Con entrada E4 o E6 $P1 = 0$ $P2 = 1$ Con entrada E7 $P1 = 0$ $P2 = 0$
D	$N \geq N3$, subiendo desde C	Con entrada E6 o E7 $P1 = 0$ $P2 = 0$
E	$N2 \leq N < N3$, bajando desde D	Con entrada E4 o E6 $P1 = 0$ $P2 = 0$
F	$N1 \leq N < N2$, bajando desde E	Con entrada E4 $P1 = 0$ $P2 = 0$ Con entrada E0 $P1 = 1$ $P2 = 1$
G	$N1 \leq N < N2$, bajando desde C	

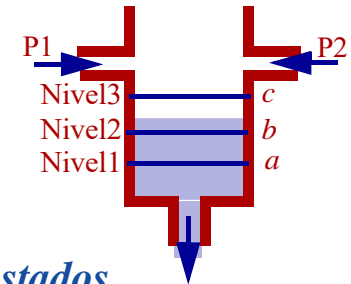
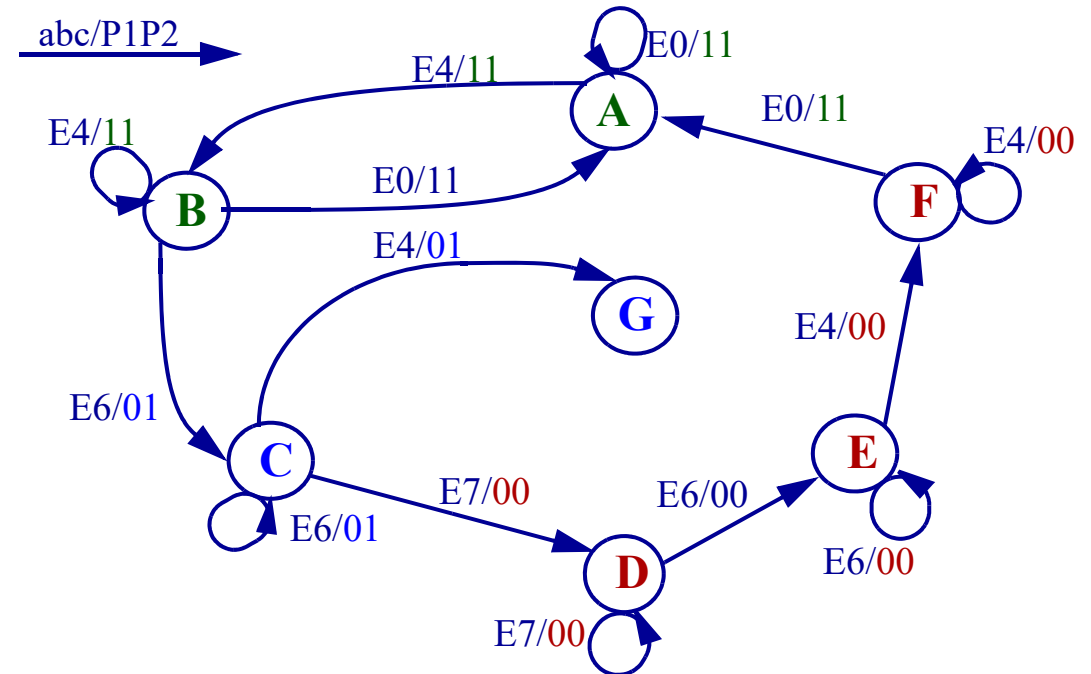


Diagrama de transición de estados



Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 2: Continuación

Cod	Significado	Acción
A	$N < N1$	Con entrada E0 o E4 $P1 = 1$ $P2 = 1$
B	$N1 \leq N < N2$ subiendo desde A	Con entrada E4 $P1 = 1$ $P2 = 1$ Con entrada E6 $P1 = 0$ $P2 = 1$
C	$N2 \leq N < N3$ subiendo desde B, o subiendo desde G	Con entrada E4 o E6 $P1 = 0$ $P2 = 1$ Con entrada E7 $P1 = 0$ $P2 = 0$
D	$N \geq N3$, subiendo desde C	Con entrada E6 o E7 $P1 = 0$ $P2 = 0$
E	$N2 \leq N < N3$, bajando desde D	Con entrada E4 o E6 $P1 = 0$ $P2 = 0$
F	$N1 \leq N < N2$, bajando desde E	Con entrada E4 $P1 = 0$ $P2 = 0$ Con entrada E0 $P1 = 1$ $P2 = 1$
G	$N1 \leq N < N2$, bajando desde C	Con entrada E4 o E6 $P1 = 0$ $P2 = 1$ Con entrada E0 $P1 = 1$ $P2 = 1$

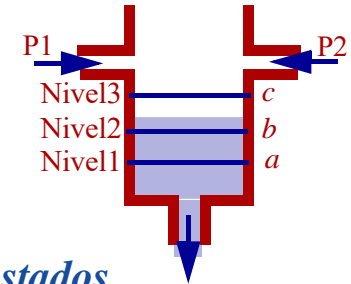
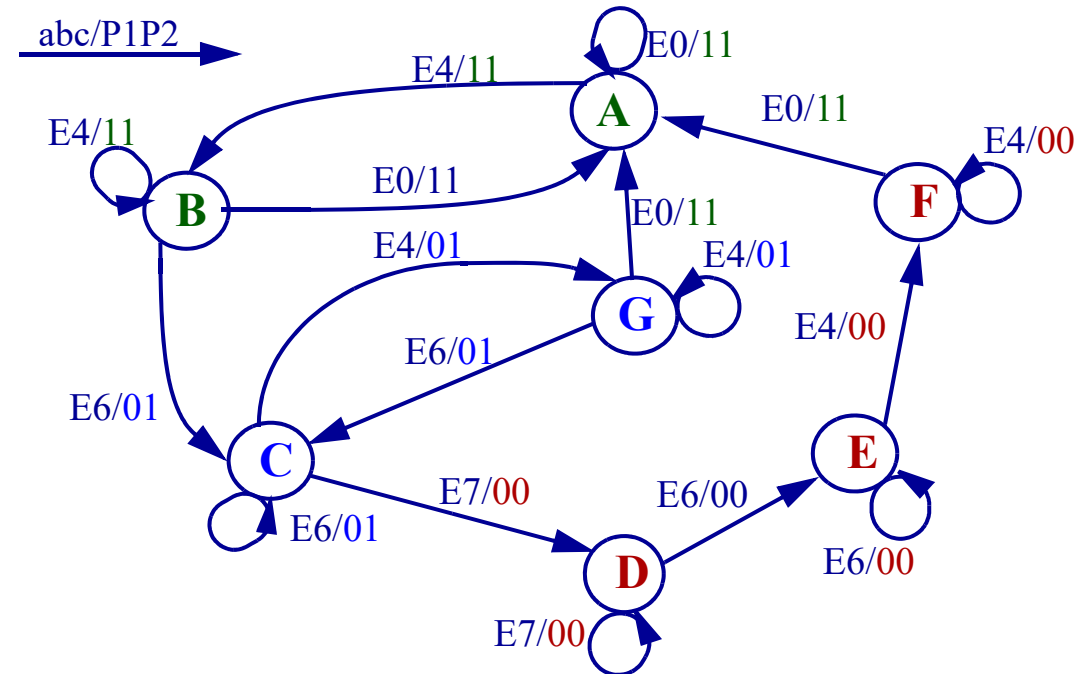


Diagrama de transición de estados



Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 2: Continuación

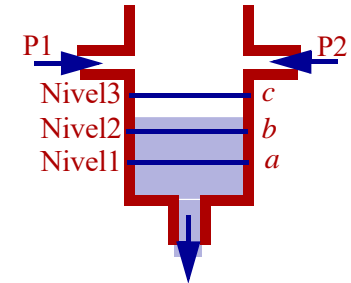
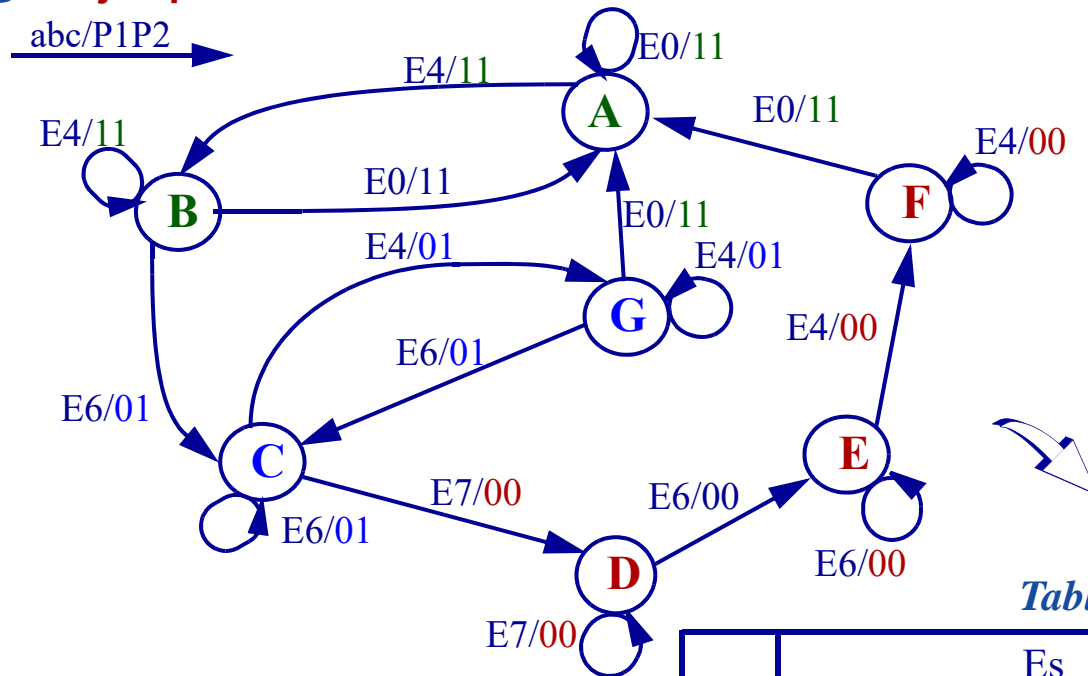


Tabla de transición de estados

Ep	Es								Salidas P1 P2							
	E0	E1	E2	E3	E4	E5	E6	E7	E0	E1	E2	E3	E4	E5	E6	E7
A	A	-	-	-	B	-	-	-	11	00	00	00	11	00	--	--
B	A	-	-	-	B	-	C	-	11	00	00	00	11	00	01	--
C	-	-	-	-	G	-	C	D	--	00	00	00	01	00	01	00
D	-	-	-	-	-	-	E	D	--	00	00	00	--	00	00	00
E	-	-	-	-	F	-	E	-	--	00	00	00	00	00	00	--
F	A	-	-	-	F	-	-	-	11	00	00	00	00	00	--	--
G	A	-	-	-	G	-	C	-	11	00	00	00	01	00	01	--

Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 2: Continuación

Tabla de transición de estados

Ep	Es								Salidas P1 P2							
	E0	E1	E2	E3	E4	E5	E6	E7	E0	E1	E2	E3	E4	E5	E6	E7
A	A	-	-	-	B	-	-	-	11	00	00	00	11	00	--	--
B	A	-	-	-	B	-	C	-	11	00	00	00	11	00	01	--
C	-	-	-	-	G	-	C	D	--	00	00	00	01	00	01	00
D	-	-	-	-	-	-	E	D	--	00	00	00	--	00	00	00
E	-	-	-	-	F	-	E	-	--	00	00	00	00	00	00	--
F	A	-	-	-	F	-	-	-	11	00	00	00	00	00	--	--
G	A	-	-	-	G	-	C	-	11	00	00	00	01	00	01	--

Tabla de implicación

B	v					
C	X	X				
D	v	X	X			
E	X	X	X	v		
F	X	X	X	v	v	
G	X	X	v	X	X	X
	A	B	C	D	E	F

- Elegimos los estados A, C y D como representantes para construir

Tabla de transición de estados reducida

Ep	Es								Salidas P1 P2							
	E0	E1	E2	E3	E4	E5	E6	E7	E0	E1	E2	E3	E4	E5	E6	E7
A	A	-	-	-	A	-	C	-	11	00	00	00	11	00	01	--
C	A	-	-	-	C	-	C	D	11	00	00	00	01	00	01	00
D	A	-	-	-	D	-	D	D	11	00	00	00	00	00	00	00

- Las parejas de estados compatibles son:
(A,B),(A,D),(C,G),(D,E),(D,F) y (E,F).

- El conjunto de conjuntos máximo compatibles resulta:

$$CMC_E = \{ \{A,B\}, \{A,D\}, \{D,E,F\}, \{C,G\} \}$$

- Por tanto

$$MCMC_E = \{ \{A,B\}, \{C,G\}, \{D,E,F\} \}$$

Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 2: Continuación

Tabla de transición de estados reducida

Ep	Es								Salidas P1 P2							
	E0	E1	E2	E3	E4	E5	E6	E7	E0	E1	E2	E3	E4	E5	E6	E7
A	A	-	-	-	A	-	C	-	11	00	00	00	11	00	01	--
C	A	-	-	-	C	-	C	D	11	00	00	00	01	00	01	00
D	A	-	-	-	D	-	D	D	11	00	00	00	00	00	00	00

Codificación de estados

Estados $q_1 q_2$

A = 11

C = 01

D = 00

Ep $q_1 q_2$	Es= $Q_1 Q_2$								Salidas P1 P2							
	E0	E1	E2	E3	E4	E5	E6	E7	E0	E1	E2	E3	E4	E5	E6	E7
11	11	-	-	-	11	-	01	-	11	00	00	00	11	00	01	--
01	11	-	-	-	01	-	01	00	11	00	00	00	01	00	01	00
00	11	-	-	-	00	-	00	00	11	00	00	00	00	00	00	00

Sustituimos estados por su código

En este ejemplo usaremos:

- un flip-flop D para la variable q_2
- y un JK para la variable q_1

Obtenemos funciones entrada de FF.

Ep $q_1 q_2$	J1K1,D2								Salidas P1 P2							
	E0	E1	E2	E3	E4	E5	E6	E7	E0	E1	E2	E3	E4	E5	E6	E7
11	x0,1	xx,x	xx,x	xx,x	x0,1	xx,x	x1,1	xx,x	11	00	00	00	11	00	01	--
01	1x,1	xx,x	xx,x	xx,x	0x,1	xx,x	0x,1	0x,0	11	00	00	00	01	00	01	00
00	1x,1	xx,x	xx,x	xx,x	0x,0	xx,x	0x,0	0x,0	11	00	00	00	00	00	00	00

Tabla de excitación del flip-flop JK

Q_n	Q_{n+1}	J	K
0	0	0	x
0	1	1	x
1	0	x	1
1	1	x	0

Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 2: Continuación

*Tabla de verdad de las funciones de entrada de los F:F J1, K1 y D2
y de las funciones de salida P1 y P2*

Ep q1q2	J1K1,D2								Salidas P1 P2							
	E0	E1	E2	E3	E4	E5	E6	E7	E0	E1	E2	E3	E4	E5	E6	E7
11	x0,1	xx,x	xx,x	xx,x	x0,1	xx,x	x1,1	xx,x	11	00	00	00	11	00	01	--
01	1x,1	xx,x	xx,x	xx,x	0x,1	xx,x	0x,1	0x,0	11	00	00	00	01	00	01	00
00	1x,1	xx,x	xx,x	xx,x	0x,0	xx,x	0x,0	0x,0	11	00	00	00	00	00	00	00

abc q1q2	000	001	011	010	110	111	101	100
00	1 0	x 1	x 3	x 2	0 6	0 7	x 5	0 4
01	1 8	x 9	x 11	x 10	0 14	0 15	x 13	0 12
11	x 24	x 25	x 27	x 26	x 30	x 31	x 29	x 28
10	x 16	x 17	x 19	x 18	x 22	x 23	x 21	x 20

$$J1 = \bar{a}$$

abc q1q2	000	001	011	010	110	111	101	100
00	x 0	x 1	x 3	x 2	x 6	x 7	x 5	x 4
01	x 8	x 9	x 11	x 10	x 14	x 15	x 13	x 12
11	0 24	x 25	x 27	x 26	1 30	x 31	x 29	0 28
10	x 16	x 17	x 19	x 18	x 22	x 23	x 21	x 20

$$K1 = b$$

Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 2: Continuación

*Tabla de verdad de las funciones de entrada de los F:F J1, K1 y D2
y de las funciones de salida P1 y P2*

Ep q1q2	J1K1,D2								Salidas P1 P2							
	E0	E1	E2	E3	E4	E5	E6	E7	E0	E1	E2	E3	E4	E5	E6	E7
11	x0,1	xx,x	xx,x	xx,x	x0,1	xx,x	x1,1	xx,x	11	00	00	00	11	00	01	--
01	1x,1	xx,x	xx,x	xx,x	0x,1	xx,x	0x,1	0x,0	11	00	00	00	01	00	01	00
00	1x,1	xx,x	xx,x	xx,x	0x,0	xx,x	0x,0	0x,0	11	00	00	00	00	00	00	00

abc q1q2	000	001	011	010	110	111	101	100
00	1 0	x 1	x 3	x 2	0 6	0 7	x 5	0 4
01	1 8	x 9	x 11	x 10	1 14	0 15	x 13	1 12
11	1 24	x 25	x 27	x 26	1 30	x 31	x 29	1 28
10	x 16	x 17	x 19	x 18	x 22	x 23	x 21	x 20

$$D2 = \bar{a} + \bar{c} q2$$

Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 2: Continuación

*Tabla de verdad de las funciones de entrada de los F:F J1, K1 y D2
y de las funciones de salida P1 y P2*

Ep q1q2	J1K1,D2								Salidas P1 P2							
	E0	E1	E2	E3	E4	E5	E6	E7	E0	E1	E2	E3	E4	E5	E6	E7
11	x0,1	xx,x	xx,x	xx,x	x0,1	xx,x	x1,1	xx,x	11	00	00	00	11	00	01	--
01	1x,1	xx,x	xx,x	xx,x	0x,1	xx,x	0x,1	0x,0	11	00	00	00	01	00	01	00
00	1x,1	xx,x	xx,x	xx,x	0x,0	xx,x	0x,0	0x,0	11	00	00	00	00	00	00	00

abc q1q2	000	001	011	010	110	111	101	100
00	1 ₀	0 ₁	0 ₃	0 ₂	0 ₆	0 ₇	0 ₅	0 ₄
01	1 ₈	0 ₉	0 ₁₁	0 ₁₀	0 ₁₄	0 ₁₅	0 ₁₃	0 ₁₂
11	1 ₂₄	0 ₂₅	0 ₂₇	0 ₂₆	0 ₃₀	x ₃₁	0 ₂₉	1 ₂₈
10	x ₁₆	x ₁₇	x ₁₉	x ₁₈	x ₂₂	x ₂₃	x ₂₁	x ₂₀

$\bar{a} \bar{b} \bar{c} \leftarrow$ (points to cell 0)
 $q1 \bar{b} \bar{c} \leftarrow$ (points to column 0)

$$P1 = \bar{a} \bar{b} \bar{c} + q1 \bar{b} \bar{c}$$

abc q1q2	000	001	011	010	110	111	101	100
00	1 ₀	0 ₁	0 ₃	0 ₂	0 ₆	0 ₇	0 ₅	0 ₄
01	1 ₈	0 ₉	0 ₁₁	0 ₁₀	1 ₁₄	0 ₁₅	0 ₁₃	1 ₁₂
11	1 ₂₄	0 ₂₅	0 ₂₇	0 ₂₆	1 ₃₀	x ₃₁	0 ₂₉	1 ₂₈
10	x ₁₆	x ₁₇	x ₁₉	x ₁₈	x ₂₂	x ₂₃	x ₂₁	x ₂₀

$\bar{a} \bar{b} \bar{c} \leftarrow$ (points to cell 0)
 $q2 a \bar{c} \leftarrow$ (points to column 14)

$$P2 = \bar{a} \bar{b} \bar{c} + q2 a \bar{c}$$

Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 2: Continuación

En resumen hemos obtenido:

Funciones de entrada de los FF:

$$J1 = \bar{a}$$

$$K1 = b$$

$$D2 = \bar{a} + \bar{c} q_2$$

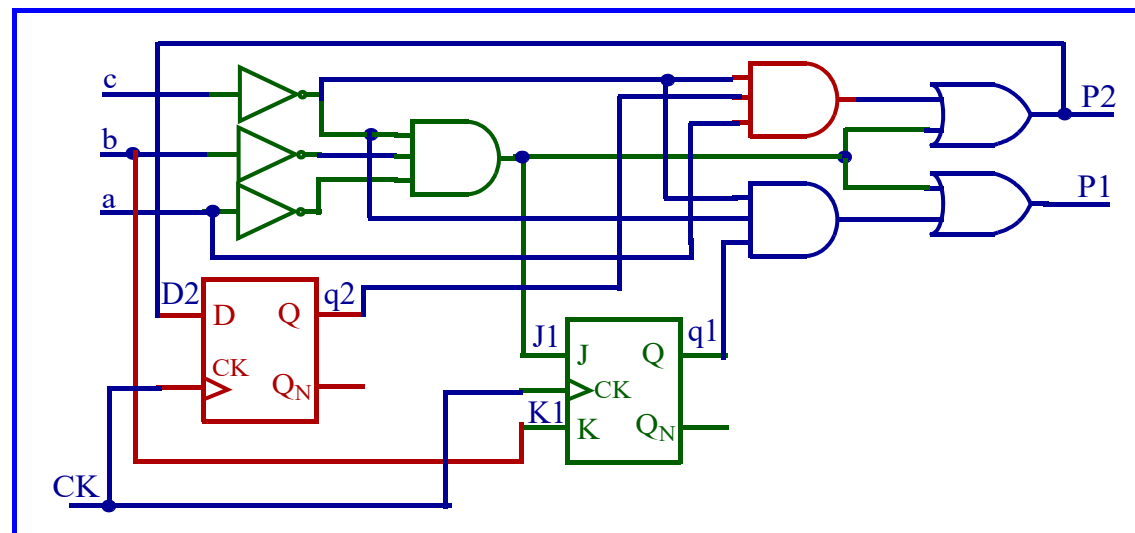
Funciones de Salida:

$$P1 = \bar{a} \bar{b} \bar{c} + q_1 \bar{b} \bar{c}$$

$$P2 = \bar{a} \bar{b} \bar{c} + q_2 a \bar{c}$$

De comparar los mapas K de J1 y P1 observamos que podemos escoger $J1 = \bar{a} \bar{b} \bar{c}$, con lo que resulta $P1 = J1 + q_1 \bar{b} \bar{c}$. Y de comparar los mapas K de D2 y P2 observamos que también podemos escoger $D2 = P2 = J1 + q_2 a \bar{c}$.

Haciendo esto tendremos el diagrama lógico.

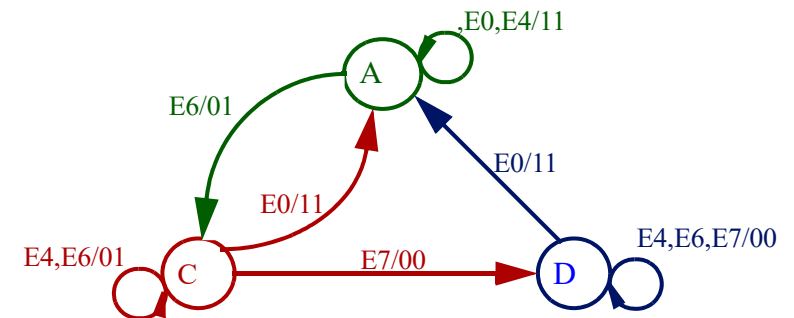


Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

● Ejemplo de diseño 2: Continuación. Conogramas (simulación)

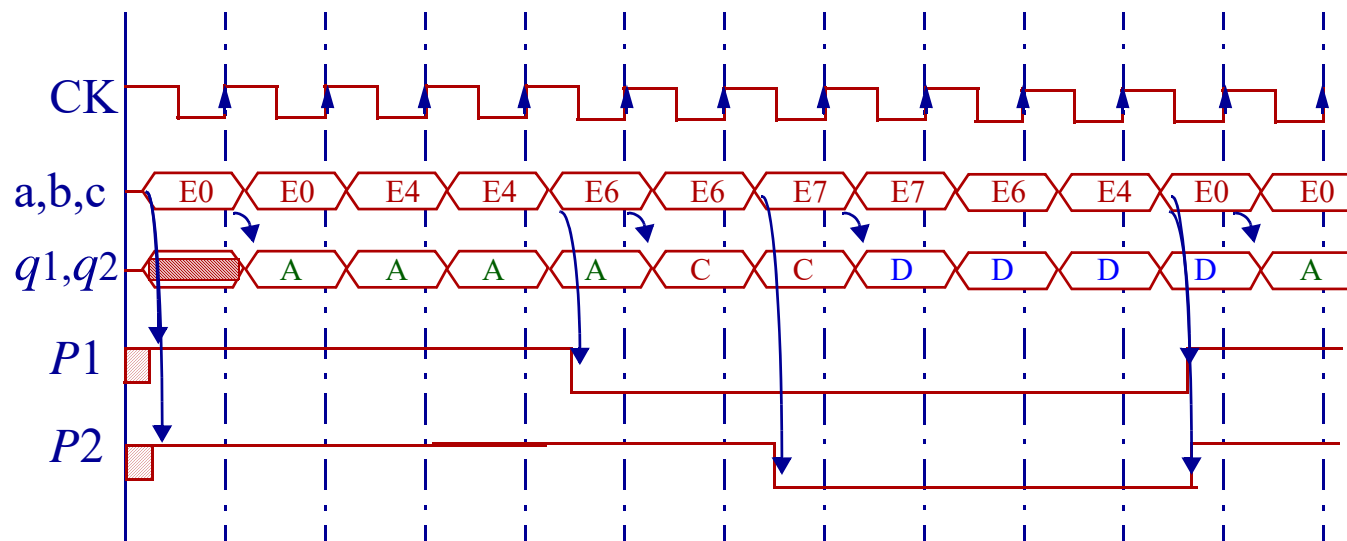
Para verificar el funcionamiento del diseño mediante un simulador habría que excitar el circuito con todas las secuencias de entrada posibles. Una simulación exhaustiva supone considerar todas las imaginables. A continuación se muestran, a modo de ejemplo, los cronogramas correspondientes a algunas de ellas.

Diagrama/Tabla de estados reducida



Sec. 1: Llenado y vaciado del depósito

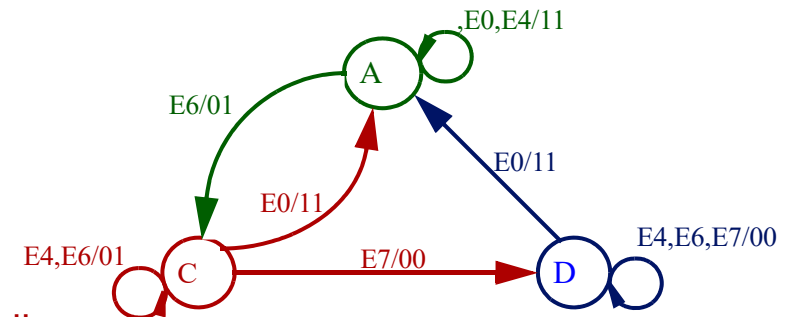
E0 -> E4 -> E6 -> E7 -> E6 -> E4 -> E0



Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

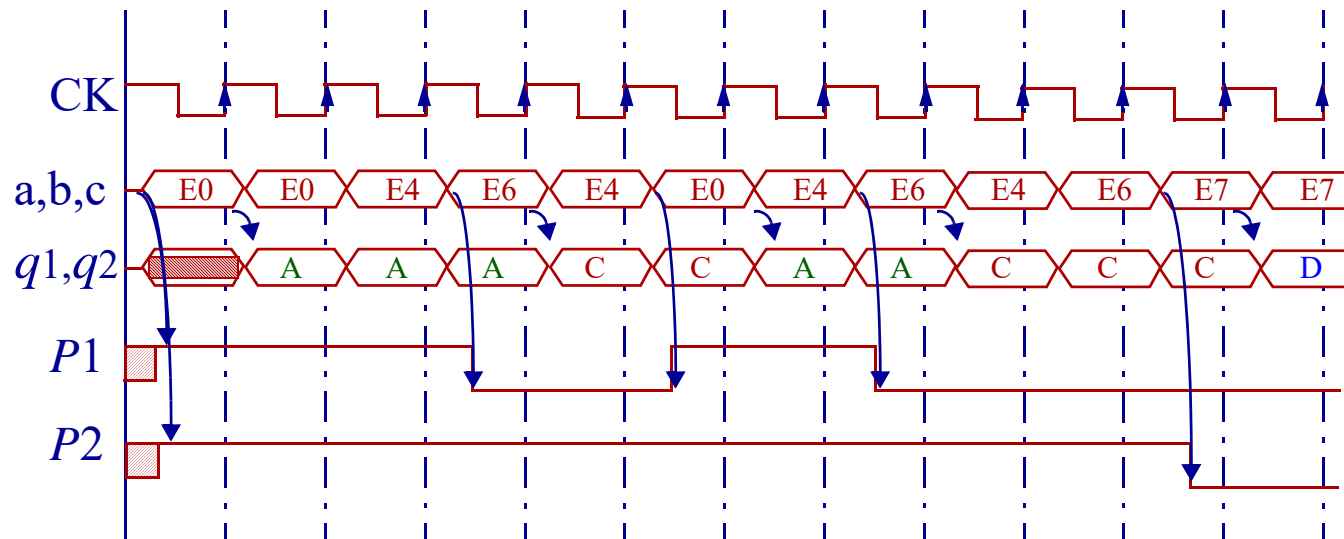
● Ejemplo de diseño 2: Continuación. Conogramas (simulación)

Diagrama/Tabla de estados reducida



Sec. 2: Llenado del depósito con subidas y bajadas intermedias

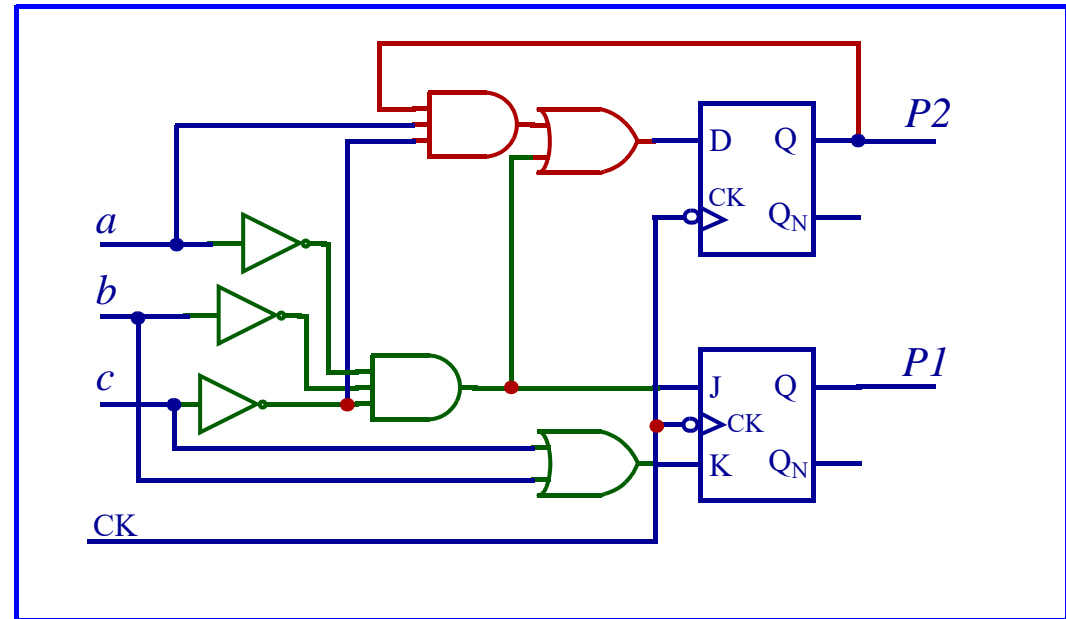
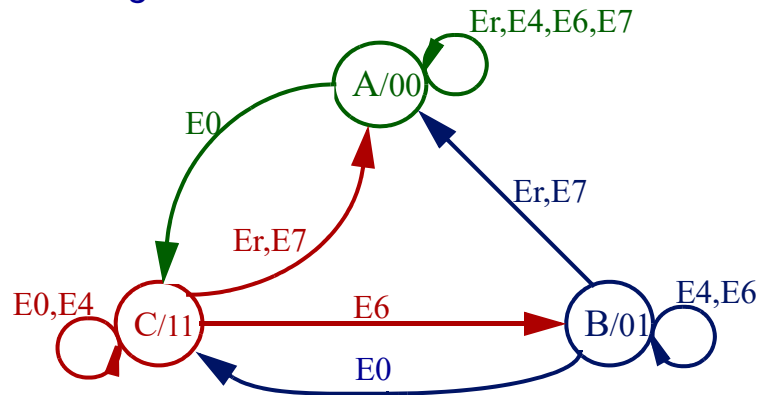
E0 -> E4 -> E6 -> E4 -> E0 -> E4 -> E6 -> E4 -> E6 -> E7



Sistemas digitales secuenciales síncronos. Síntesis de máquinas de estados.

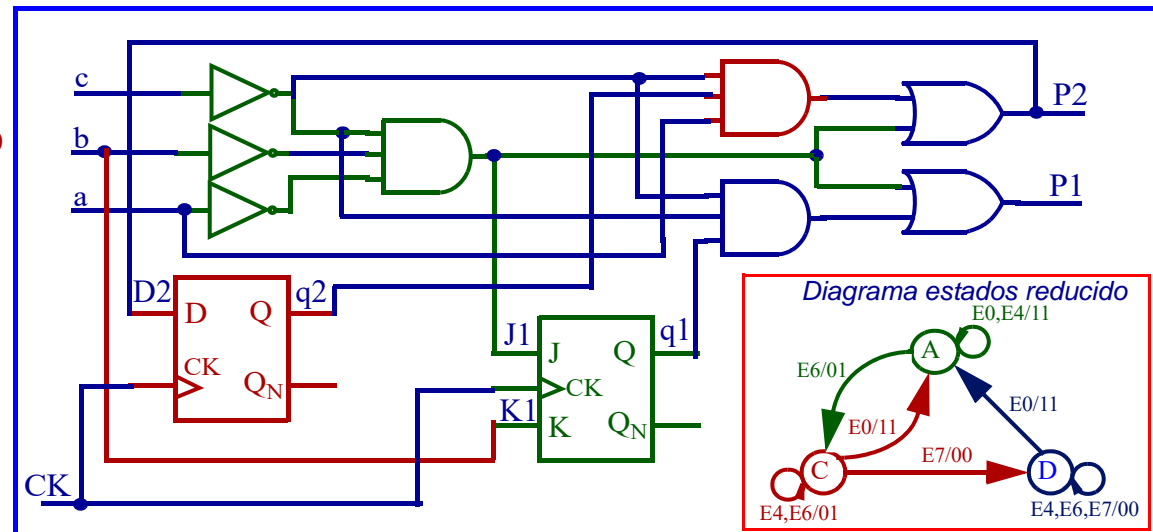
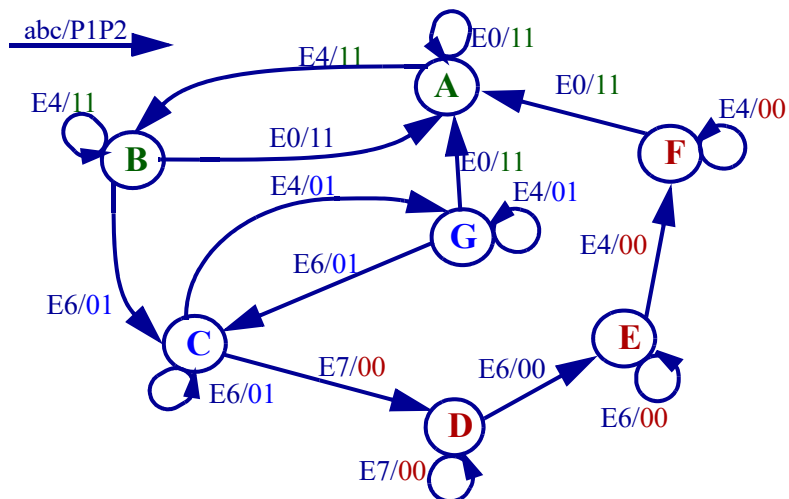
Solución: máquina de estados de Moore

Diagrama de estados

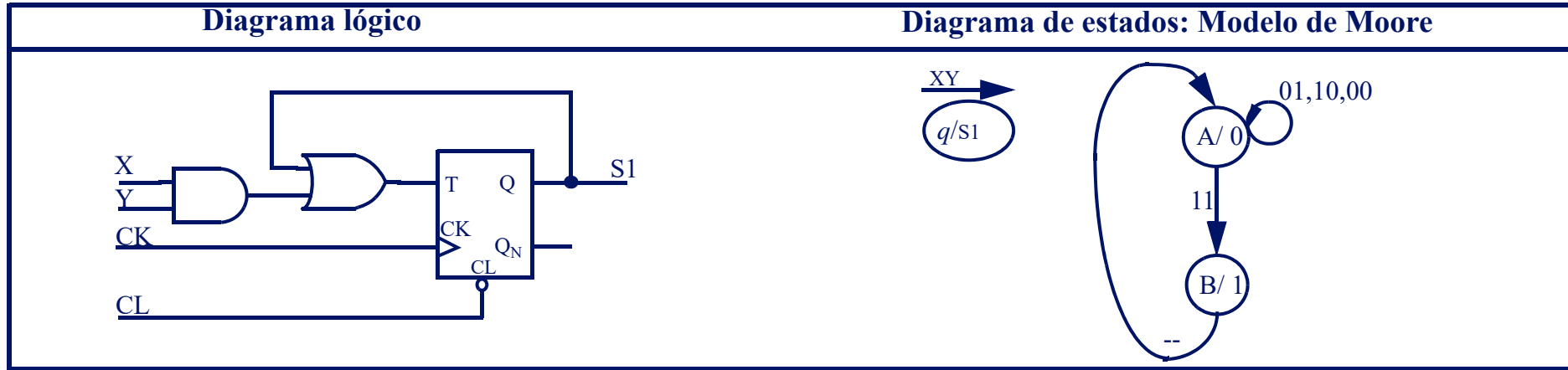


Solución: máquina de estados de Mealy

Diagrama de estados



Diagramas de estados en HDL



Descripción máquina de estados Sec_Moore.vhd

```

library IEEE;
use IEEE.std_logic_1164.all;
entity Sec_Moore is
    port (
        CK, CL, X, Y : in STD_LOGIC;
        S1 : out STD_LOGIC;
    );
end Sec_Moore;
architecture Sec_Moore of Sec_Moore is
    type state_type is (A, B);
    signal present_state, next_state : state_type;
begin
    sreg : process (CK, CL)
    begin
        if CL = '0' then
            present_state <= A;
        elseif CK'event and CK = '1' then
            present_state <= next_state;
        end if;
    end process;

```

```

C1 : process (present_state, X, Y)
begin
    case present_state is
        when A =>
            if X = '1' and Y = '1' then
                next_state <= B;
            else
                next_state <= A;
            end if;
        when B =>
            next_state <= A;
        when others =>
            null;
        end case;
    end process;

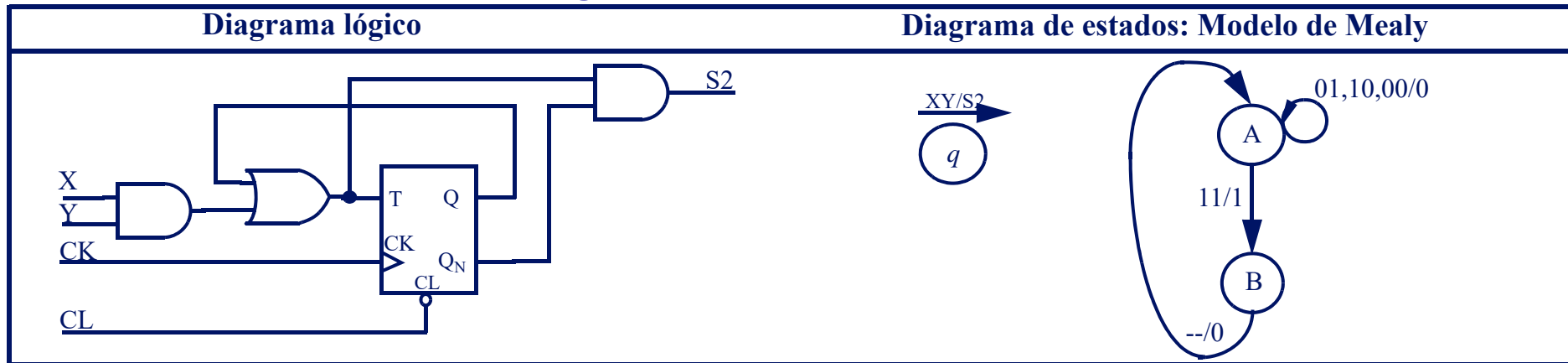
```

```

C2 : process (present_state)
begin
    if present_state = B then
        S1 <= '1';
    else
        S1 <= '0';
    end if;
end process;
end Sec_Moore;

```

Diagramas de estados en HDL



Descripción máquina de estados Sec_Mealy.vhd

```

library IEEE;
use IEEE.std_logic_1164.all;
entity Sec_Mealy is
    port (
        CK, CL, X, Y : in STD_LOGIC;
        S1 : out STD_LOGIC;
    );
end Sec_Mealy;
architecture Sec_Mealy of Sec_Mealy is
    type state_type is (A, B);
    signal present_state, next_state : state_type;
begin
    sreg : process (CK, CL)
    begin
        if CL = '0' then
            present_state <= A;
        elsif CK'event and CK = '1' then
            present_state <= next_state;
        end if;
    end process;

```

```

    C1 : process (present_state, X, Y)
    begin
        case present_state is
            when A =>
                if X = '1' and Y = '1' then
                    next_state <= B;
                else
                    next_state <= A;
                end if;
            when B =>
                next_state <= A;
            when others =>
                null;
            end case;
        end process;

```

```

    C2 : process (present_state, X, Y)
    begin
        if present_state = A then
            if X = '1' and Y = '1' then
                S2 <= '1';
            else
                S2 <= '0';
            end if;
        else
            S2 <= '0';
        end if;
    end process;
end Sec_Mealy;

```