Tema 2

Análisis y diseño de circuitos digitales

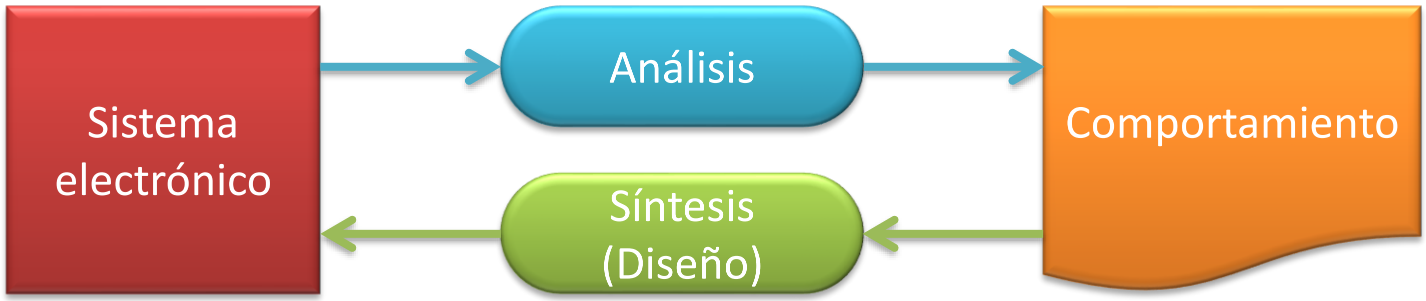
Contenido

Contenido

1. Introducción
2. Análisis de circuitos combinacionales
3. Análisis de circuitos secuenciales
4. Diseño de circuitos secuenciales (recordatorio)

- En el manejo de circuitos en general, y de circuitos digitales en particular, existen dos tipos de procesos:
  - Análisis
  - Diseño
- Análisis: consiste en la obtención de una descripción del comportamiento de un circuito descrito de forma estructural
- Diseño: consiste en pasar una descripción de comportamiento (alto nivel) a una descripción estructural (en base a componentes)

Sistemas electrónicos: análisis y diseño



- No linealidad
- Ruido
- Distorsión
- Retrasos

- Más o menos ideal
- Modelos muy simplificados

El proceso de análisis extrae el comportamiento de un sistema electrónico

El proceso de diseño construye un circuito que realiza el comportamiento deseado

Contenido

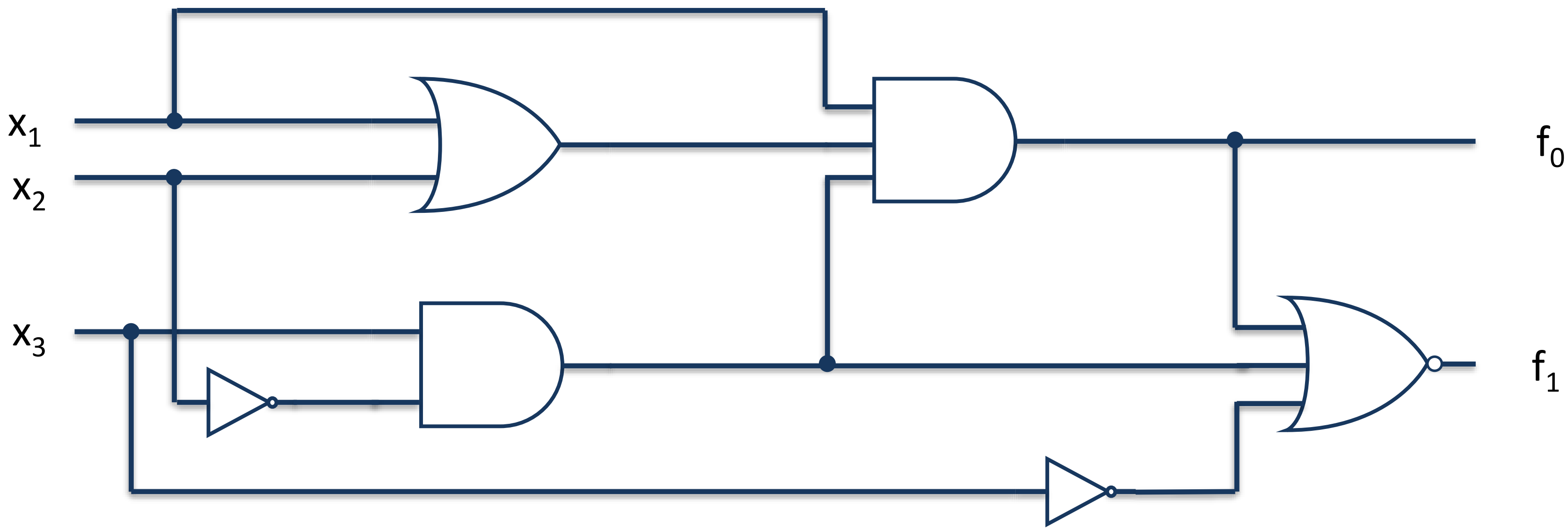
1. Introducción
2. Análisis de circuitos combinacionales
3. Análisis de circuitos secuenciales
4. Diseño de circuitos secuenciales (recordatorio)

- Análisis lógico: Determinación de la función lógica que realiza un determinado circuito lógico
  - Obtiene la expresión algebraica que representa su comportamiento
  - La relación entre las entradas y las salidas
  - Sólo puede realizarse considerando comportamientos ideales

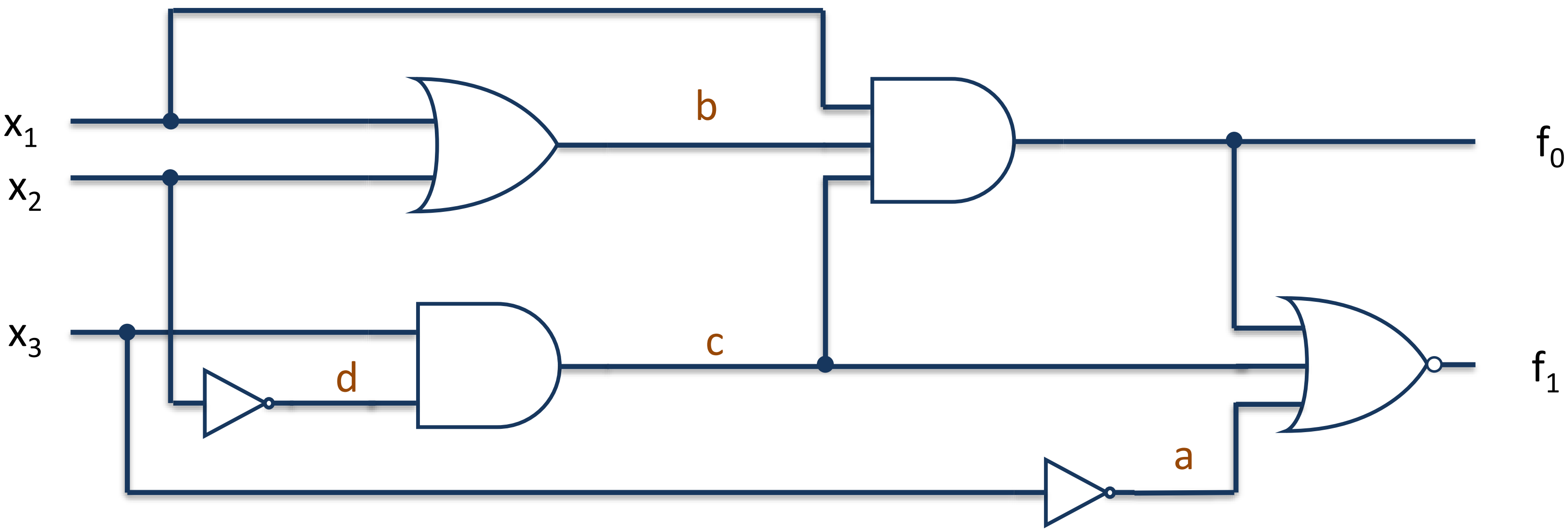
Proceso de análisis

- Es sencillo si se procede de forma ordenada:
  1. Identificar entradas y salidas
  2. Asignar nombres a las salidas de las puertas
  3. Obtener la relación matemática entre las salidas de las puertas y sus entradas
  4. Manipular las relaciones obtenidas hasta obtener las salidas del circuito en función de las entradas

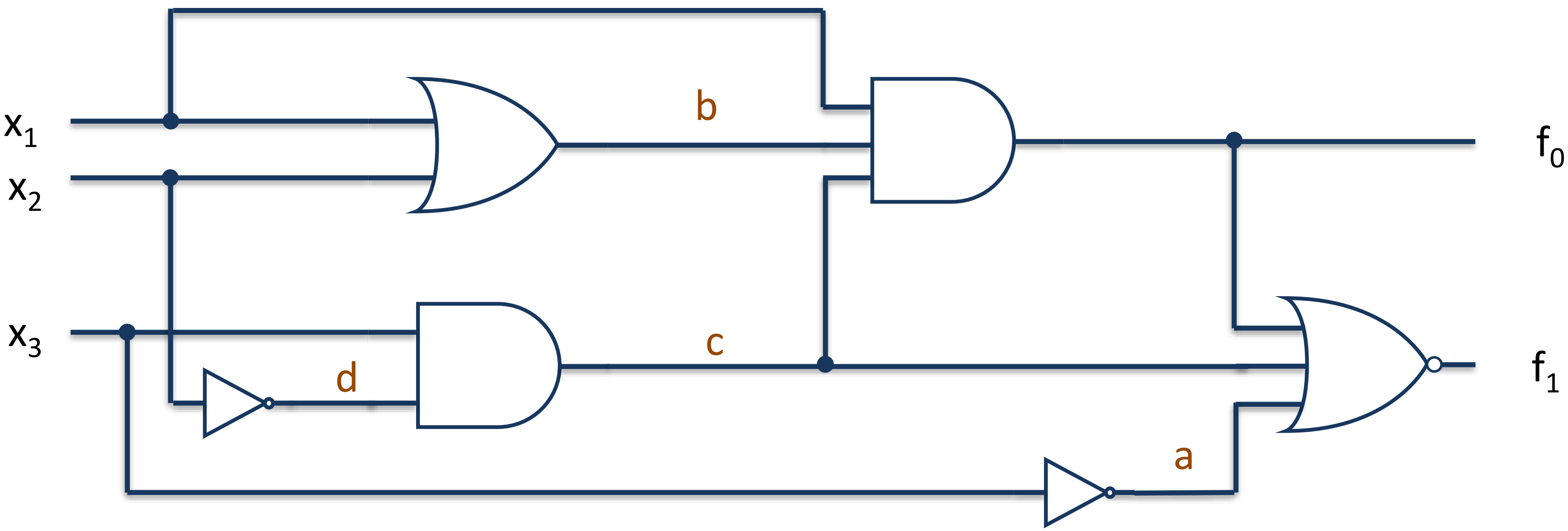
- Analizar el siguiente circuito:



A) Nombramos las salidas de las puertas



B) Obtenemos la expresión de la salida de cada puerta en función de sus entradas



$$a = \overline{x_3}$$

$$b = x_1 + x_2$$

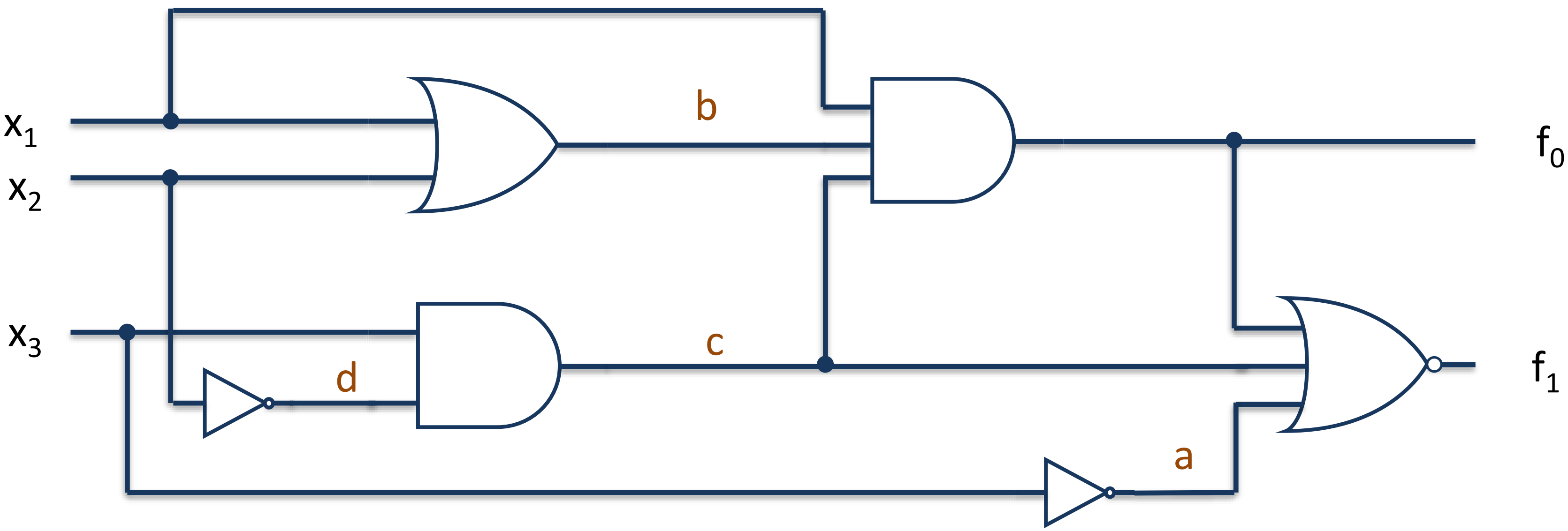
$$c = x_3 \cdot d$$

$$d = \overline{x_2}$$

$$f_0 = x_1 \cdot b \cdot c$$

$$f_1 = \overline{f_0 + c + a}$$

C) Sustituimos las señales intermedias en función de las entradas



$a = \overline{x_3}$

$b = x_1 + x_2$

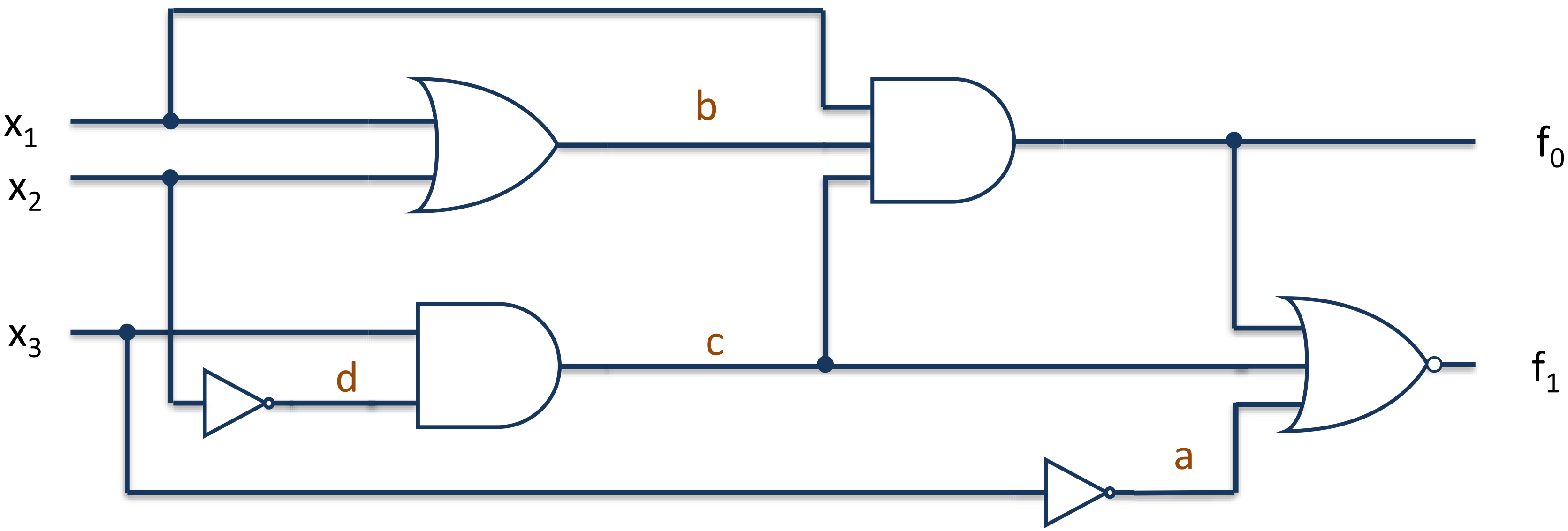
$c = x_3 \cdot d = x_3 \cdot \overline{x_2}$

$d = \overline{x_2}$

$f_0 = x_1 \cdot b \cdot c = x_1 \cdot (x_1 + x_2) \cdot (x_3 \cdot \overline{x_2})$

$$f_1 = \overline{f_0 + c + a}$$
$$= \overline{(x_1 \cdot (x_1 + x_2) \cdot (x_3 \cdot \overline{x_2})\ ) + (x_3 \cdot \overline{x_2}) + \overline{x_3}}$$

D) Operamos para dejarlo en forma de SP



$a = \overline{x_3}$

$b = x_1 + x_2$

$c = x_3 \cdot d = x_3 \cdot \overline{x_2}$

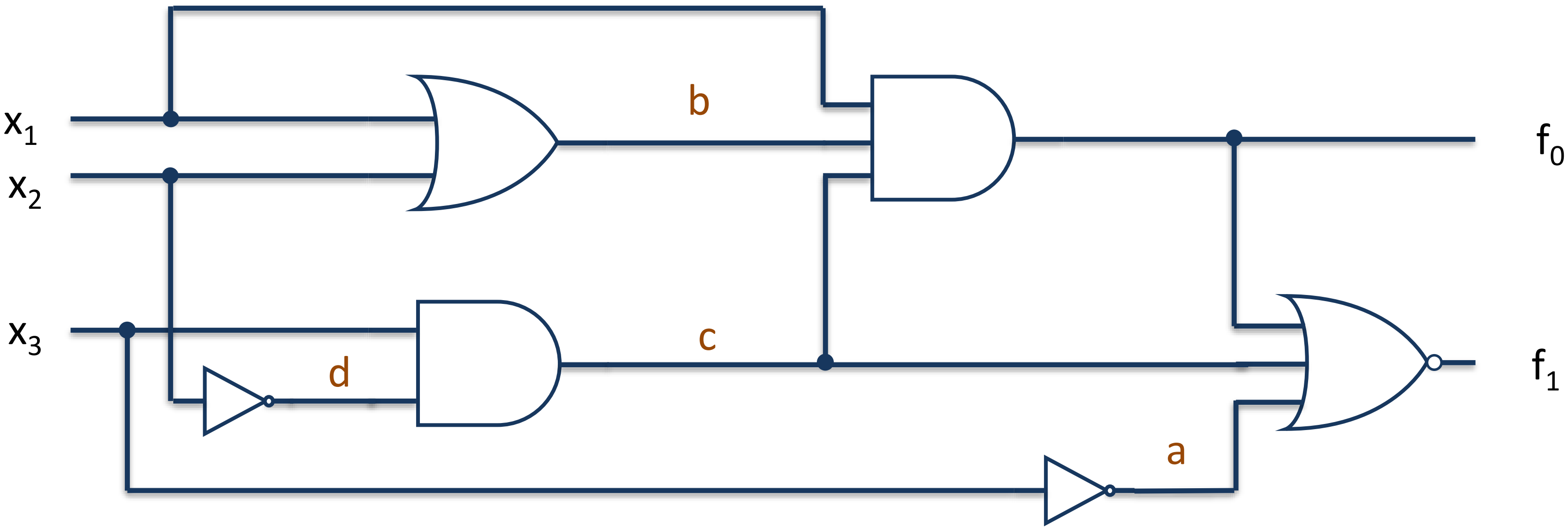
$d = \overline{x_2}$

$f_0 = x_1 \cdot b \cdot c = x_1 \cdot (x_1 + x_2) \cdot (x_3 \cdot \overline{x_2}) = \ x_1 \cdot \overline{x_2} \cdot x_3$

$$f_1 = \overline{f_0 + c + a}$$
$$= \overline{(x_1 \cdot (x_1 + x_2) \cdot (x_3 \cdot \overline{x_2})\ ) + (x_3 \cdot \overline{x_2}) + \overline{x_3}}$$
$$= x_2 \cdot x_3$$

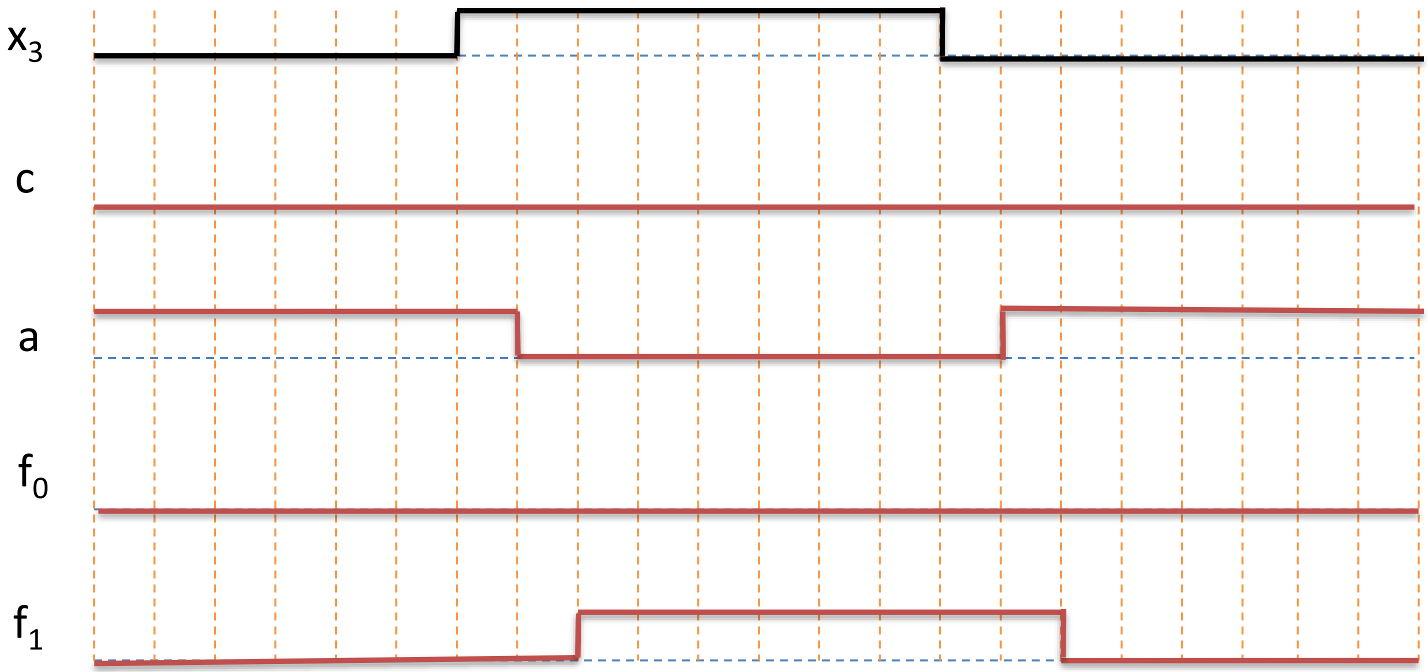
- Si el circuito tiene retrasos no puede obtenerse la expresión algebraica:
  - Hay que obtener la evolución en el tiempo

- Obtener la evolución de las salidas del circuito en el tiempo:
  - Cuando todas las puertas tienen un retraso D
  - Cuando $x_1 = x_2 =$ ‘1’ y x_3 cambia de forma periódica



Proceso de análisis con retraso

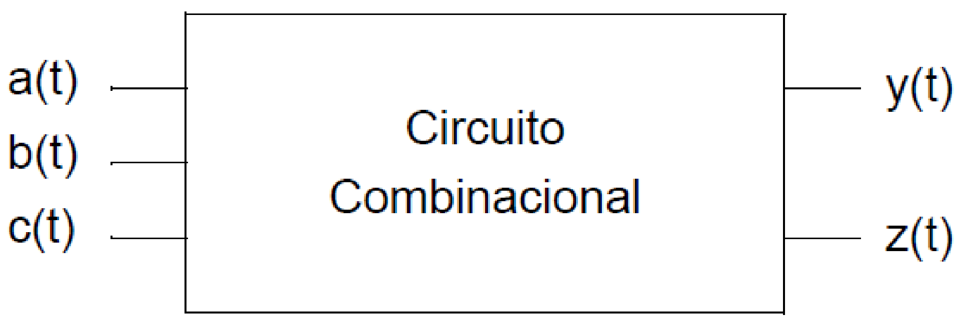
Con estos valores de las entradas b = ‘1’ y d = ‘0’ por lo que no se dibujan



Contenido

1. Introducción
2. Análisis de circuitos combinacionales
3. **Análisis de circuitos secuenciales**
4. Diseño de circuitos secuenciales (recordatorio)

- Las salidas en un determinado instante sólo dependen del valor de las entradas presentes en ese momento
  - Cada combinación de entradas sólo da un valor de salida (tabla de verdad)

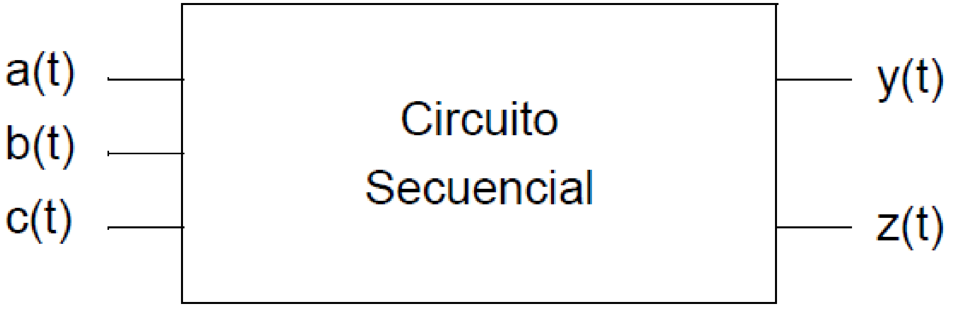


$$y(t_0) = f_1(a(t_0), b(t_0), c(t_0))$$

$$z(t_0) = f_2(a(t_0), b(t_0), c(t_0))$$

Circuitos secuenciales (recordatorio)

- Las salidas en un determinado instante dependen del valor de las entradas hasta ese momento
  - Una misma combinación de entradas puede dar salidas diferentes en función de su historia pasada
  - Tienen que "recordar" esa historia pasada: necesitan **elementos de memoria**

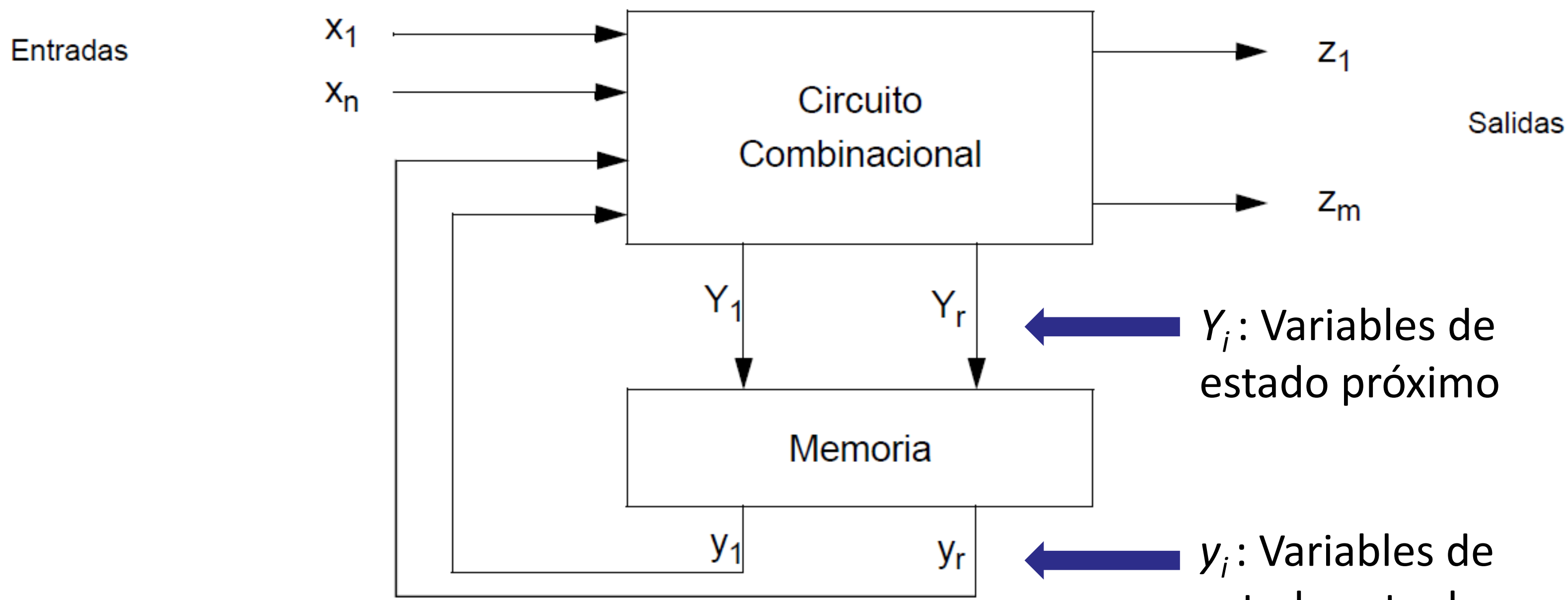


$$y(t_0) = f_1(a(t), b(t), c(t))$$

$$z(t_0) = f_2(a(t), b(t), c(t))$$

(para todo t menor o igual que t_0)

- Su estructura puede verse como una parte combinacional y unos elementos de memoria



Y_i : Variables de estado próximo

y_i : Variables de estado actual

$z_i(t) = f\,[x_1(t),\ldots, x_n(t), y_1(t),\ldots, y_r(t)]$

$Y_k(t) = f_Y\,[x_1(t),\ldots, x_n(t), y_1(t),\ldots, y_r(t)]$

Las salidas dependen de las entradas y del estado actual

Necesidad de los circuitos secuenciales

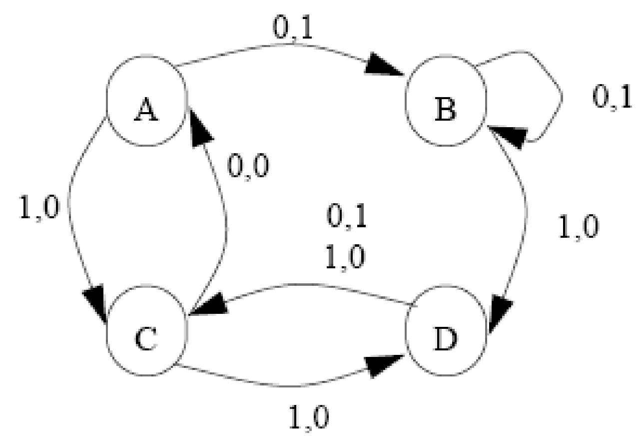
- Se utilizan principalmente cuando
  - El comportamiento del sistema depende intrínsecamente del tiempo
    - Ej: El circuito para controlar un semáforo (tiempo que lleva encendido cada luz)
  - Existe una operación fundamental que se repite *n* veces (procesos iterativos)
    - Ej: sumadores de muchos bits donde el elemento de memoria almacene resultados intermedios

Formas de representación

- La tabla de verdad o la expresión algebraica no son válidas
- Existen dos modos de representación:
  - La tabla de estados
  - El diagrama de estados

	0	1
A	B, 1	C, 0
B	B, 1	D, 0
C	A, 0	D, 0
D	C, 1	C, 0

- Tabla de estados:
  - Aparecen las entradas, el estado presente, el próximo estado y la salida
- Diagrama de estados:
  - Se representan los estados del sistemas y las posibles transiciones para cada combinación de entradas con la salida correspondiente



Circuitos secuenciales síncronos

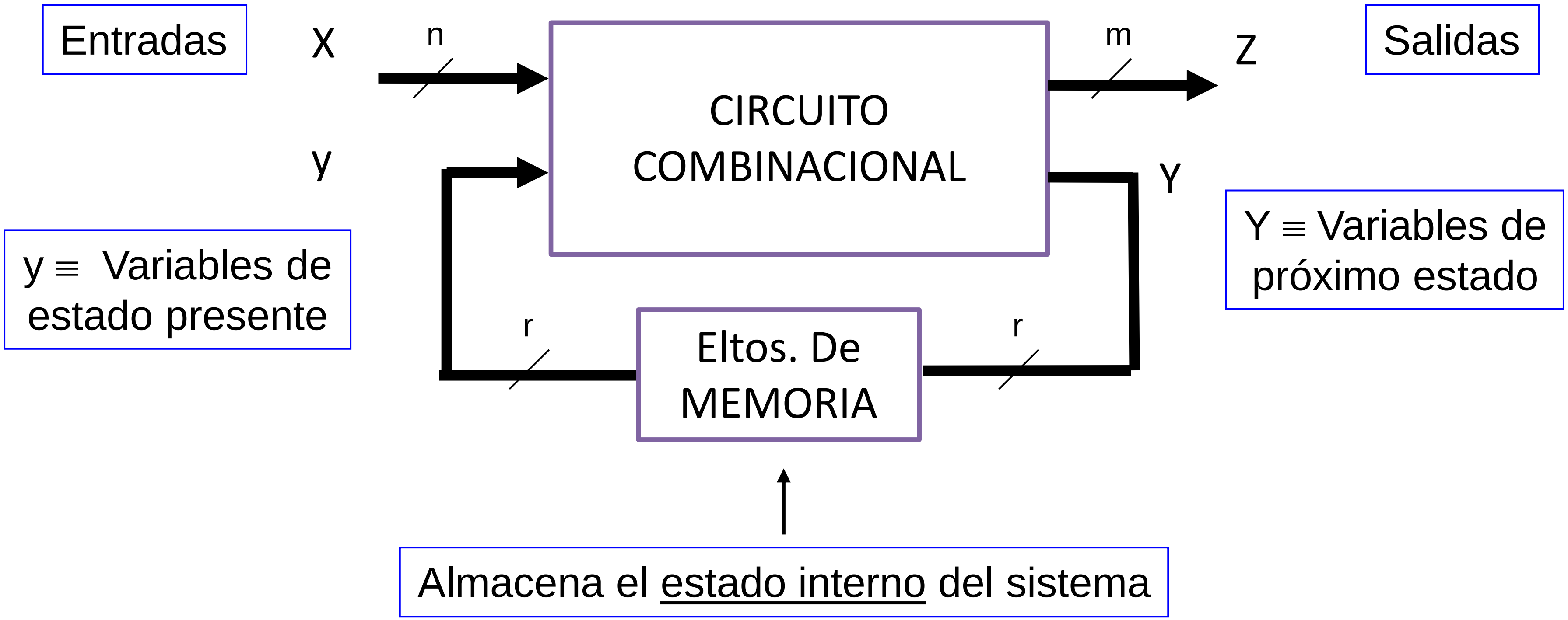
- Los circuitos secuenciales se pueden clasificar en:
  - Circuitos secuenciales **asíncronos**:
    Las entradas actúan en todo instante sobre el estado del circuito
  - Circuitos secuenciales **síncronos**:
    Las entradas actúan sobre el estado del circuito sólo en determinados instantes, que vienen determinados por la señal de reloj
- El diseño de circuitos secuenciales lo vamos a hacer en base a diseño de **máquinas de estados finitos** o **máquinas de estado**

NOTA: En nuestro concepto de circuito secuencial síncrono todos los biestables van a estar disparados por la misma señal de reloj y por el mismo flanco

Concepto de estado

- **Estado**: Se puede definir como el conjunto de valores almacenado en los biestables durante un determinado ciclo de reloj
  - En general, un circuito con *n* biestables podrá tener 2^n estados posibles
- Los estados contienen la información relevante que debe almacenar el circuito

Máquinas de estado



$$z_1(t) = f[x_1(t),\ldots, x_n(t), y_1(t),\ldots, y_r(t)]$$

$$Y_k(t) = f_Y[x_1(t),\ldots, x_n(t), y_1(t),\ldots, y_r(t)]$$

Definición de máquina de estados finitos (FSM)

- Una máquina de estados representa un sistema como un conjunto de **estados**, de **transiciones** entre ellos junto con las **entradas y salidas** asociadas
  – Es una representación del funcionamiento de un circuito secuencial particular
  – Pueden usarse para muchas otras cosas, más allá del diseño lógico y la arquitectura de computadores
  – Es siempre un circuito síncrono. Una única señal de reloj

Máquina de estados finitos (FSM)

- Cualquier circuito con memoria puede verse como una máquina de estados
- El diseño de FSMs involucra:
  - Definir estados
  - Definir transiciones entre estados
  - Optimizar / minimizar estados (fuera de esta asignatura)
  - Definir los valores de las salidas en cada momento
- Este enfoque es práctico sólo para FSMs pequeñas

Máquinas de estado: definiciones

- Diagrama de estados:
  - Muestra la forma y la función de la máquina de estados
  - Normalmente un diagrama de círculos y flechas
- Estado presente
  - Identifica la situación del circuito digital en el ciclo actual de reloj
- Próximo estado
  - El estado al que irá la máquina de estados en la siguiente transición
  - La transición se produce son la señal de reloj (flanco)
  - Depende de los valores de las **entradas** y del **estado presente**
- Rama
  - Indica un cambio del estado presente al próximo estado

- Máquina de Moore:
  - Las salidas dependen sólo del estado presente
  - Las entradas intervienen en la decisión del próximo estado

$$Z = f(y)$$

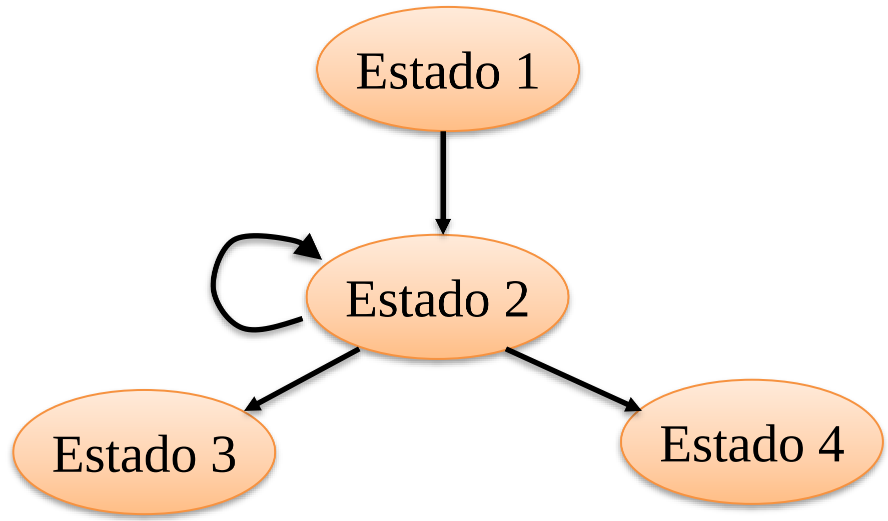
- Máquina de Mealy:
  - Las salidas dependen del estado presente y del valor de las entradas

$$Z = f(y,X)$$

- Su diferencia está únicamente en la dependencia de las salidas
  - Cualquier diseño se puede hacer como M. de Moore o como M. de Mealy
  - La descripción como M. de Mealy puede tener menos estados que la de M. Moore (pero no siempre)

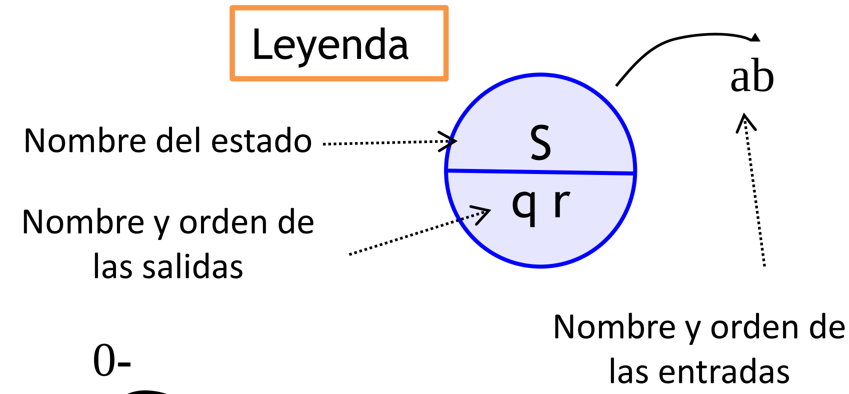
Diagramas de estado

- Para cualquier estado, hay un número finito de posibles próximos estados
- En cada ciclo de reloj la máquina cambia al siguiente estado
- Uno de los posibles próximos estados se convierte en el nuevo estado presente
  - La transición depende del estado presente y de las entradas



En un diagrama de estados bien dibujado, todas las posibles transiciones están visibles, incluyendo los bucles sobre el mismo estado.

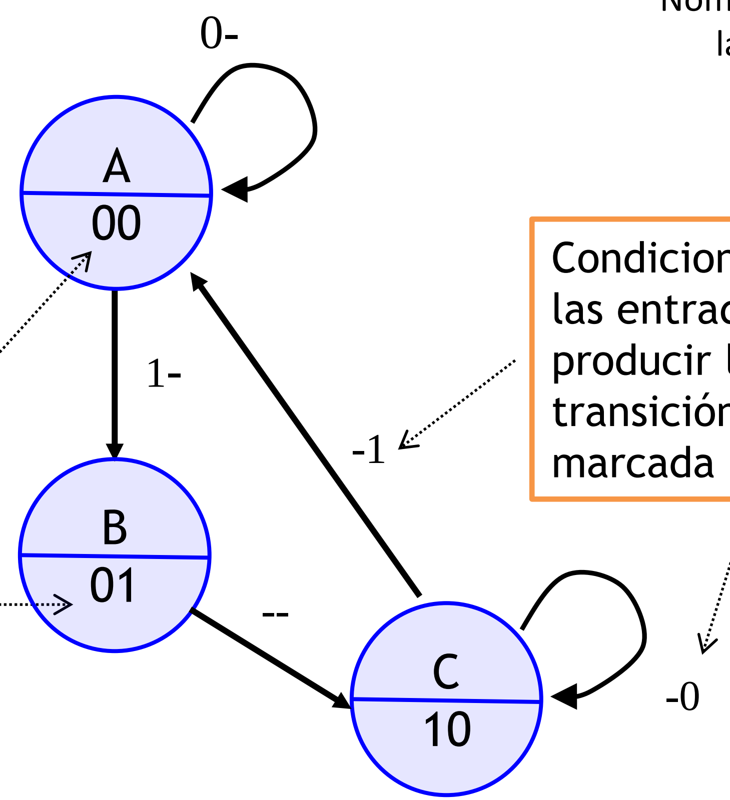
En este diagrama se puede deducir que si el estado presente es el 2, el estado previo era el 1 o el 2 y el próximo estado será el 2, el 3 ó el 4



Las salidas de las máquinas de Moore se muestran dentro del círculo del estado

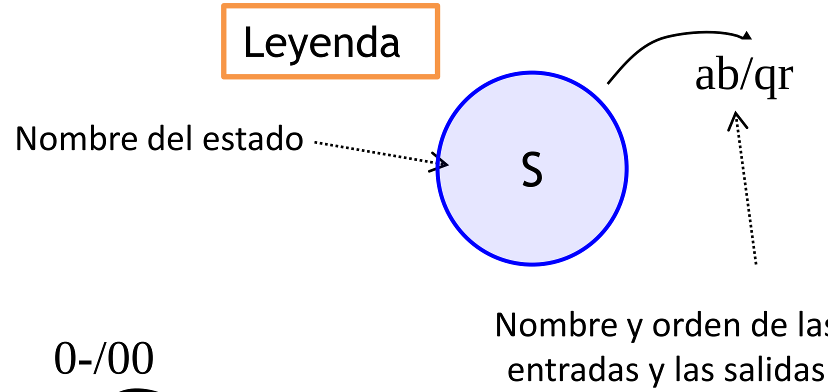
La salida está asociada al estado y sólo cambia cuando cambia el estado

Valores de las salidas cuando se está en ese estado



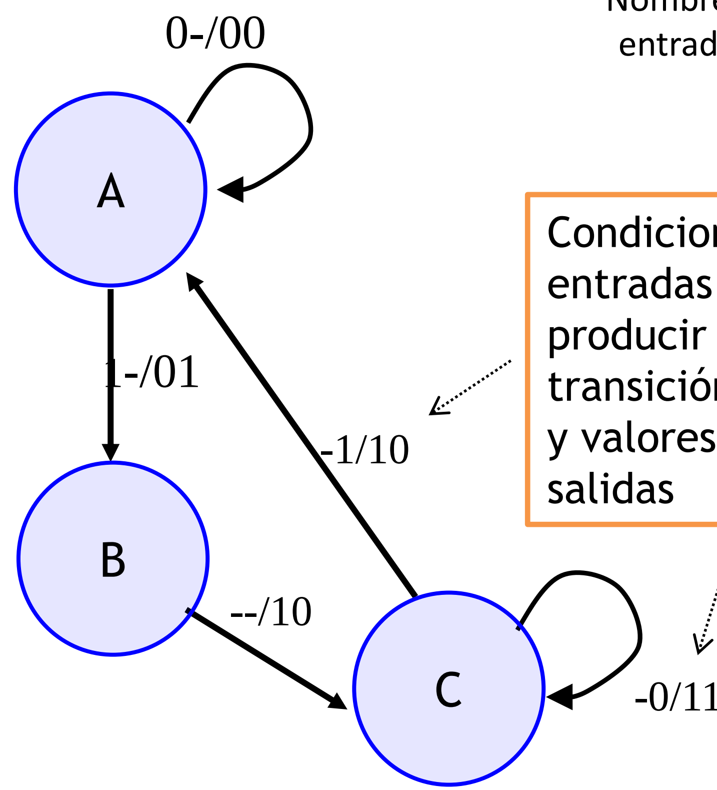
Condiciones en las entradas para producir la transición marcada

Diagramas de estado de máquinas de Mealy



Las salidas de las máquinas de Mealy se muestran en la transición

La salida está asociada al estado y a la entrada. Cambia cuando cambia el estado y/o cuando cambia la entrada



Condiciones en las entradas para producir la transición marcada y valores de las salidas

Tablas de estado

- Es una forma alternativa al diagrama de estados
  - Contiene la misma información (estado presente, próximo estado y valores de la salida) pero en formato de tabla
- Tiene un cierto parecido al mapa de Karnaugh, pero su interpretación es diferente

Tablas de estado: ejemplos

M. Mealy

S \ a	0	1
A	A,0	B,0
B	C, 1	D, 1
C	A, 0	C, 1
D	D, 1	B, 0

NS, z

M. Moore

S \ a	0	1	z
A	A	B	0
B	B	C	1
C	C	A	0

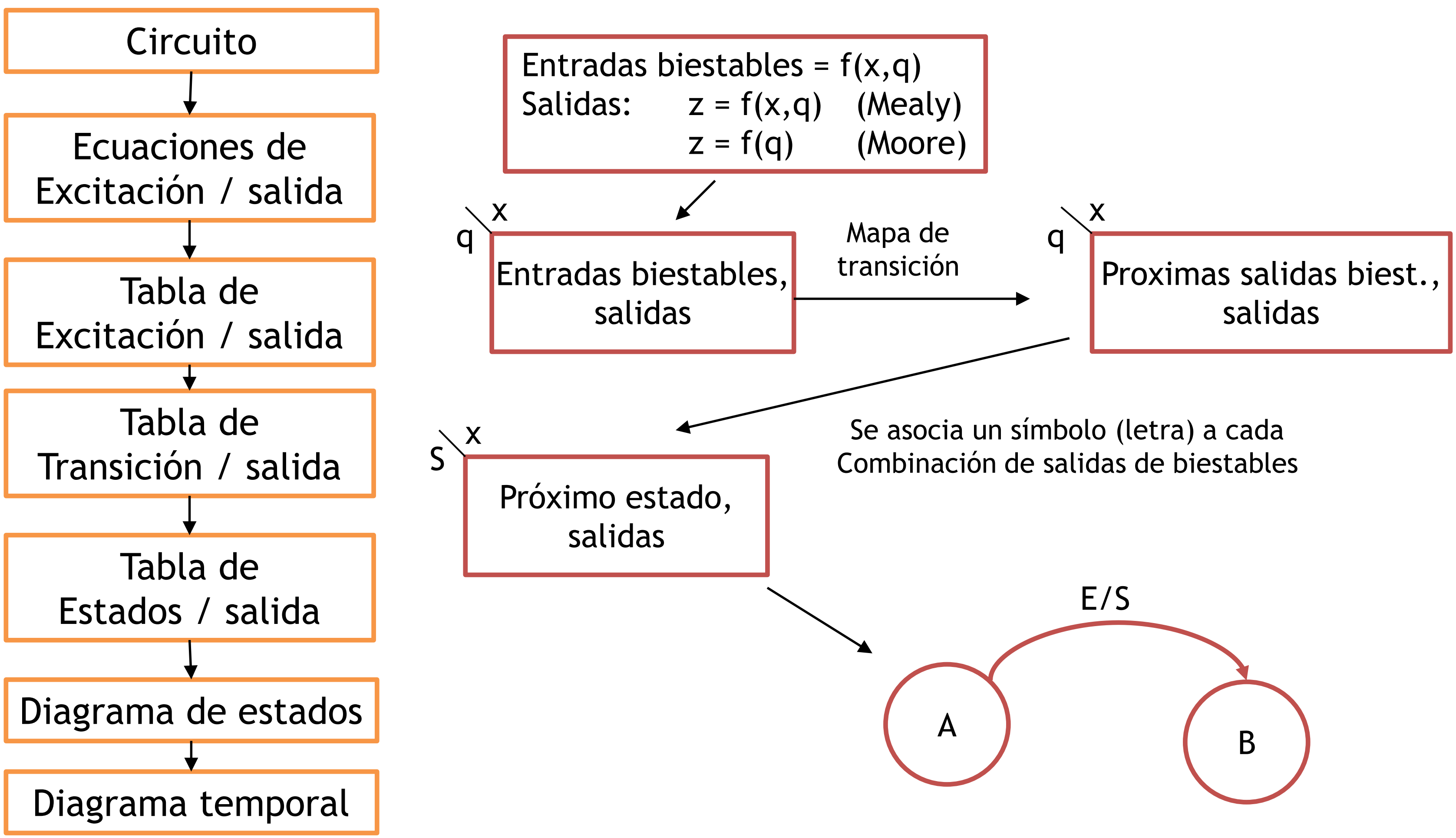
NS

S: Estado
NS: Próximo estado
a: entrada
z: salida

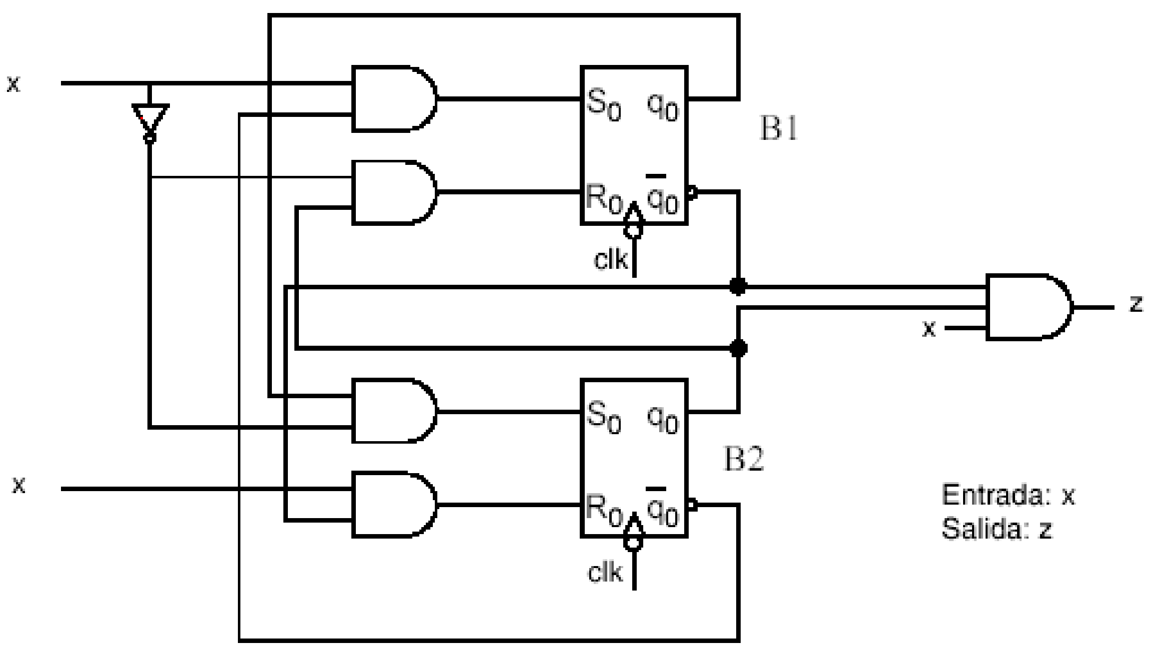
En la M. de Mealy cada celda contiene el próximo estado y la(s) salida(s)
En la máquina de Moore hay una columna aparte para la(s) salida(s)

- El proceso de análisis consiste en obtener el comportamiento de un circuito a partir de su esquemático:
  - Obtener el diagrama o la tabla de estados
  - Obtener su comportamiento temporal para unas entradas dadas

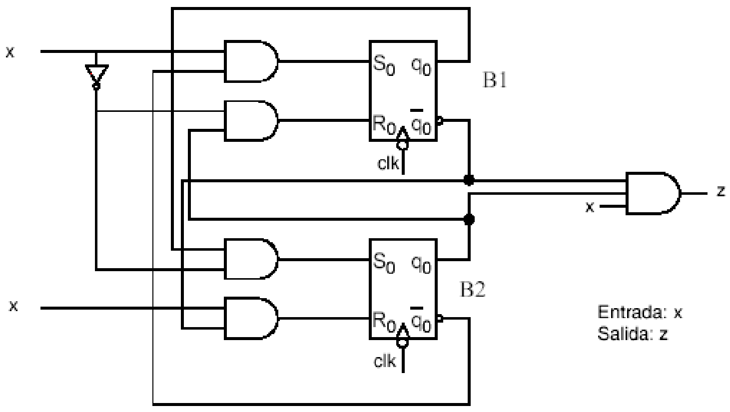
Pasos del proceso de análisis



- Analizar el comportamiento del siguiente circuito:



Paso 1: Ecuaciones de excitación / salida



Se ponen las entradas de los biestables en función de las entradas y de las propias salidas de los biestables

B1: $S_0 = x\, q_1'$

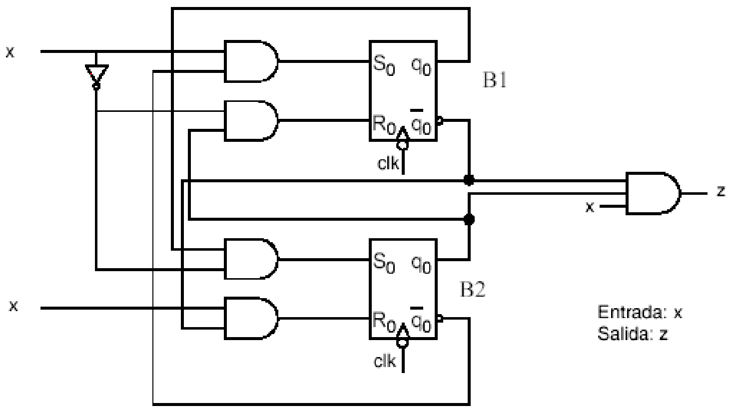
$R_0 = x'\, q_1$

B2: $S_1 = x'\, q_0$

$R_1 = x\, q_0'$

Salida: $z = x\, q_0'\, q_1$

Paso 2: Tabla de excitación / salida



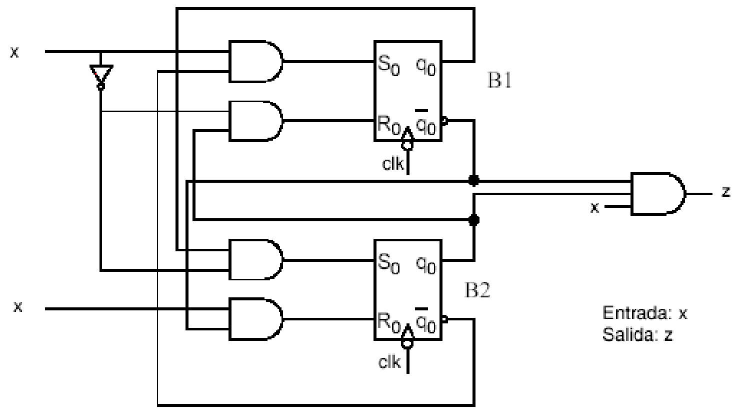
Las ecuaciones se convierten en una tabla

q_1q_0 \ X	0	1
00	00,00;0	10,01;0
01	01,00;0	00,01;0
11	01,10;0	00,00;0
10	00,10;0	10,00;1

Tanto las entradas como las "q" se ponen en codificación gray (obligatorio para hacer diseño)

Es importante mantener el mismo orden en los subíndices → R_1S_1, R_0S_0, z

Paso 3: Tabla de transición / salida



Haciendo uso de la tabla de transición del biestable se sustituyen las entradas de los biestables por los próximos estados

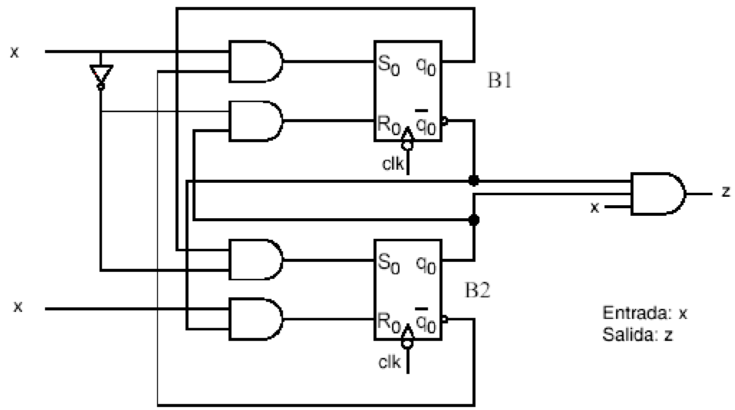
q \ RS	00	01	11	10
0	0	1	-	0
1	1	1	-	0

Q

q_1q_0 \ X	0	1
00	00;0	01;0
01	11;0	01;0
11	10;0	11;0
10	10;0	00;1

Q_1Q_0, z

Paso 4: Tabla de estados / salida



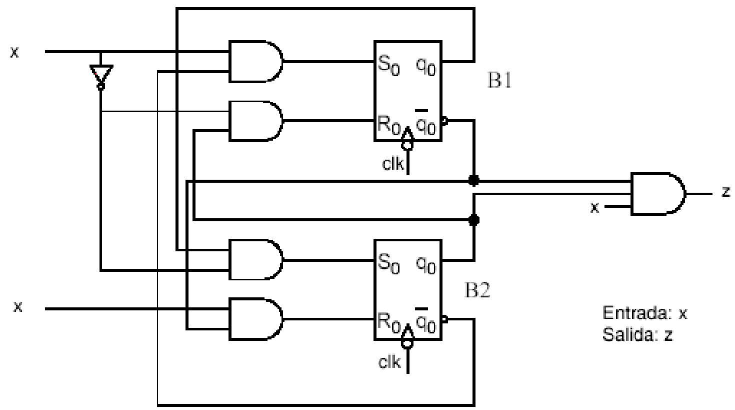
A cada combinación de las salidas de los biestables se le asigna un nombre (normalmente una letra)

Asignación de estados

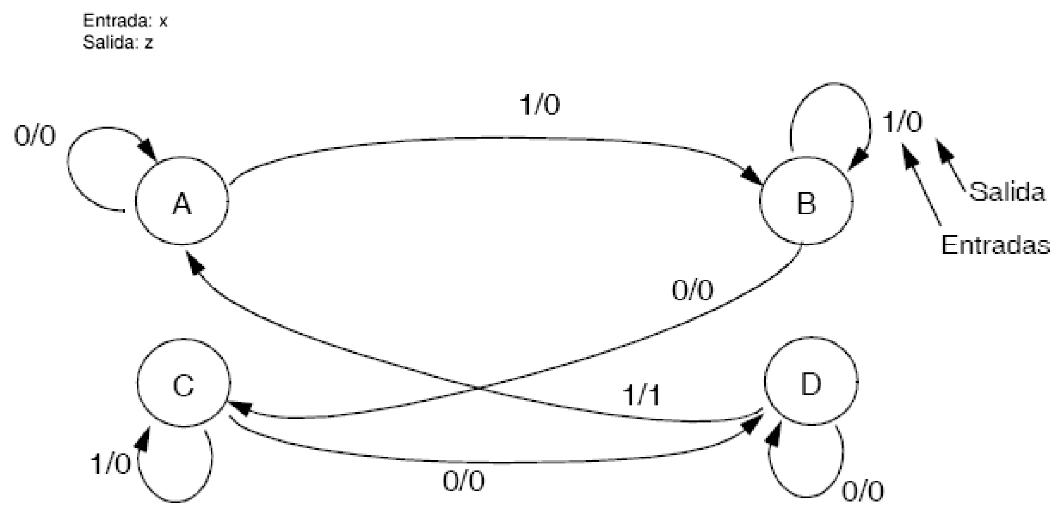
q_1q_0	
00	a
01	b
11	c
10	d

S \ X	0	1
a	a;0	b;0
b	c;0	b;0
c	d;0	c;0
d	d;0	a;1

NS, z



A partir de la tabla de estados se puede dibujar el diagrama de estados



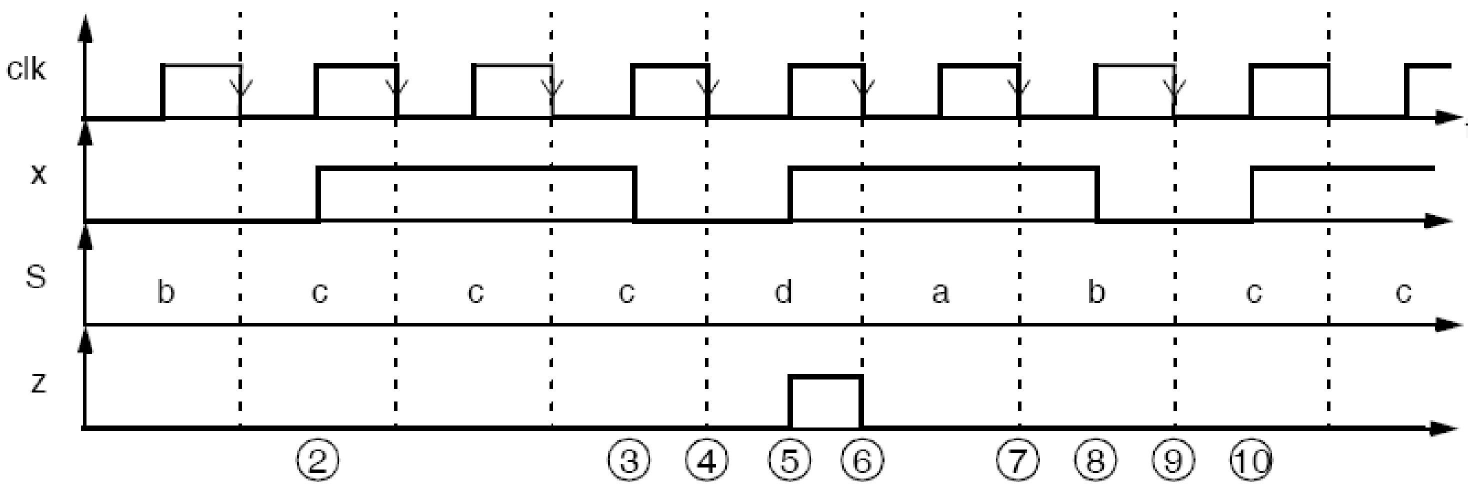
Evolución temporal

S \ X	0	1
a	a;0	b;0
b	c;0	b;0
c	d;0	c;0
d	d;0	a;1

NS, z

A partir de la tabla de estados se puede conocer la evolución del circuito para unas entradas dadas

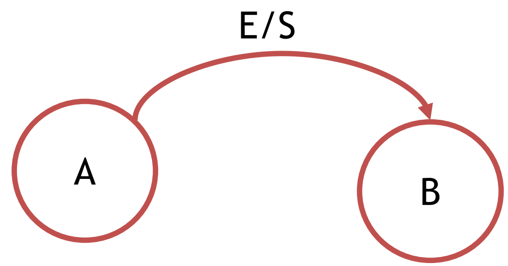
Los estados sólo pueden cambiar en los flancos activos de reloj
La salida en una máquina de Mealy cambia al cambiar la entrada



- El proceso de diseño de una máquina de estados parte de una descripción en lenguaje natural del problema a resolver
- A partir de ahí los pasos son:
  - Definir entradas y salidas (así como su significado)
  - Optar por M. Moore o M. Mealy
  - Definir estados (buscar un estado de partida)
  - Definir transiciones entre estados
  - Definir los valores de las salidas en cada momento
- Una vez obtenido el diagrama de estados se sigue el proceso hasta llegar al circuito

Pasos del proceso de diseño

Descripción no formal
↓
Diagrama de estados
↓
Tabla de Estados / salida
↓
Tabla de Transición / salida
↓
Tabla de Excitación / salida
↓
Ecuaciones de Excitación / salida
↓
Circuito



S \ x: Próximo estado, salidas

Se asocia una combinación de salidas de biestables a cada estado

q \ x: Proximas salidas biest., salidas

Se elije el biestable a utilizar

q \ x: Entradas biestables, salidas

Entradas biestables = f(x,q)
Salidas: z = f(x,q) (Mealy)
z = f(q) (Moore)

A partir de las ecuaciones se dibuja el circuito

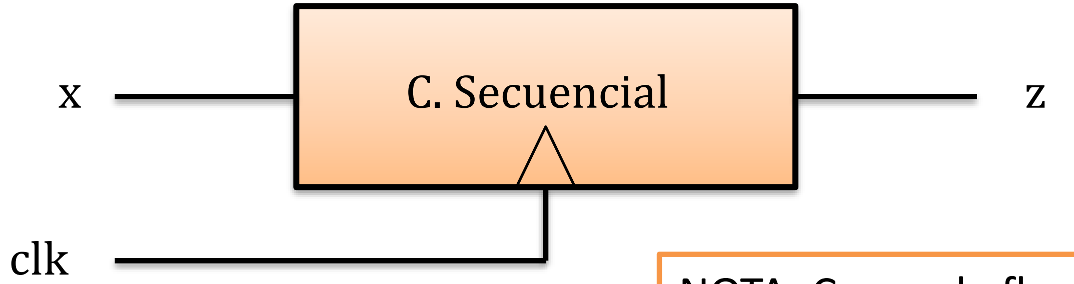
Ejemplo de diseño

- Diseñar un circuito con una entrada y una salida que tenga el siguiente comportamiento:
  - La salida se pone a 1 cuando por su entrada se reciben tres o más unos consecutivos

Vamos a hacer dos diseños:

  - Máquina de Mealy
  - Máquina de Moore

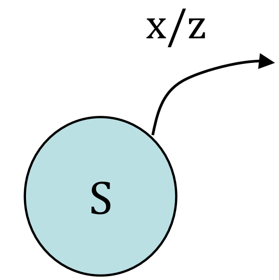
- El circuito cuenta con una entrada:
  - La llamaremos X
- El circuito cuenta con una salida:
  - La llamaremos Z
- Diseño como M. de Mealy



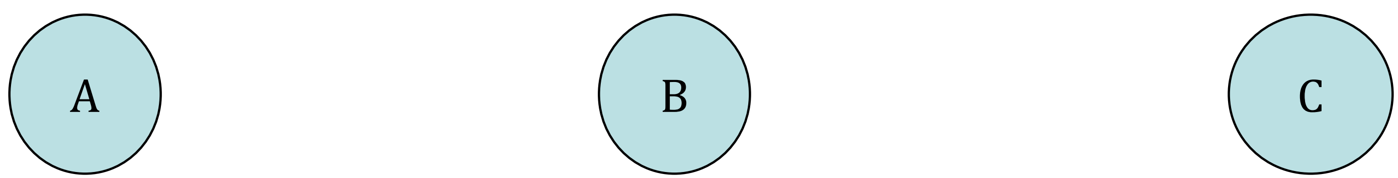
NOTA: Con cada flanco activo de reloj se mira el valor de la entrada

Planteamiento de estados

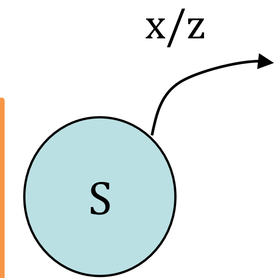
Buscamos una situación (estado) de partida para desarrollar el problema



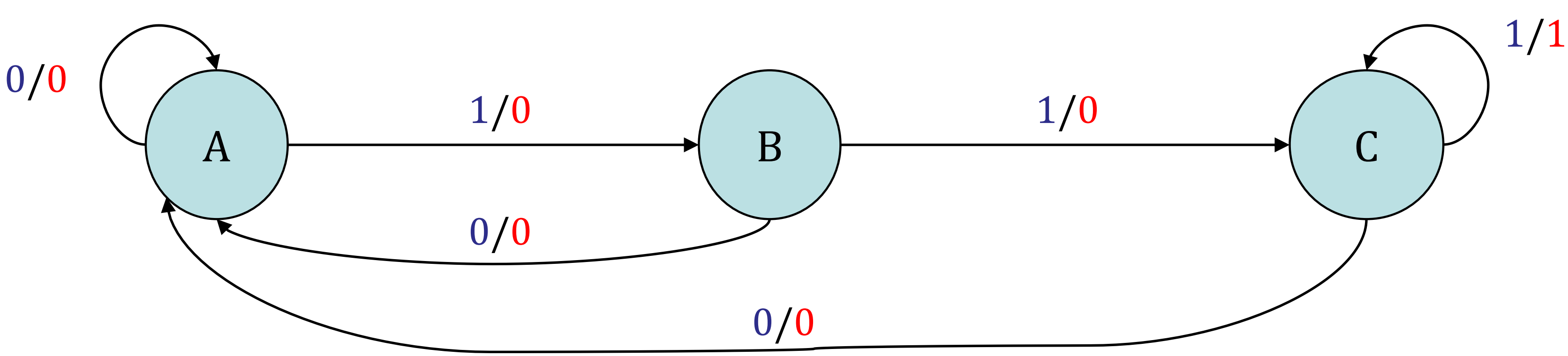
Estado A: No ha llegado ningún ‘1’ (ha llegado un ‘0’)
Estado B: Ha llegado un ‘1’
Estado C: Han llegado dos o más ‘1’ consecutivos



Planteamiento de transiciones



Cuando estamos en A y recibimos un ‘0’, seguimos en A. La salida será ‘0’
Cuando estamos en A y recibimos un ‘1’ pasamos a B. La salida será un ‘0’
Cuando estamos en B y la entrada es ‘0’ pasamos a A. La salida está a ‘0’
Cuando estamos en B y la entrada es ‘1’ pasamos a C. La salida está a ‘0’
Cuando estamos en C y la entrada es ’0’ pasamos a A. La salida está a ‘0’
Cuando estamos en C y la entrada es ‘1’ podemos permanecer en C con la salida a ‘1’, indicando que han llegado tres o más ‘1’ consecutivos



- Lo expresado en el diagrama de estados se puede expresar en forma de tabla:

S \ x	0	1
A	A,0	B,0
B	A,0	C,0
C	A,0	C,1

NS, z

Paso 3: Tabla de transición y salidas

- Se asigna a cada estado una combinación de salidas de los biestables
  - Como tenemos 3 estados, necesitamos dos biestables

Asignación de estados

s	q_1q_0
A	00
B	01
C	10

q_1q_0 \ x	0	1
00	00,0	01,0
01	00,0	10,0
11	--,-	--,-
10	00,0	10,1

Q_1Q_0, z

Paso 4: Tabla de excitación y salidas

- Se elige el biestable
  - En el ejemplo un biestable D
- En cada casilla se Introduce el valor de próximo estado del biestable

Tabla de excitación

$q \rightarrow Q$	D
$0 \rightarrow 0$	0
$0 \rightarrow 1$	1
$1 \rightarrow 0$	0
$1 \rightarrow 1$	1

q_1q_0 \ x	0	1
00	00,0	01,0
01	00,0	10,0
11	--,-	--,-
10	00,0	10,1

D_1D_0, z

- Utilizando la tabla de excitación como un mapa de Karnaugh, se obtienen las expresiones de las entradas del biestable y de la salida

q_1q_0 \ x	0	1
00	0	0
01	0	1
11	-	-
10	0	1

D_1

$D_1 = xq_1 + xq_0$

q_1q_0 \ x	0	1
00	0	1
01	0	0
11	-	-
10	0	0

D_0

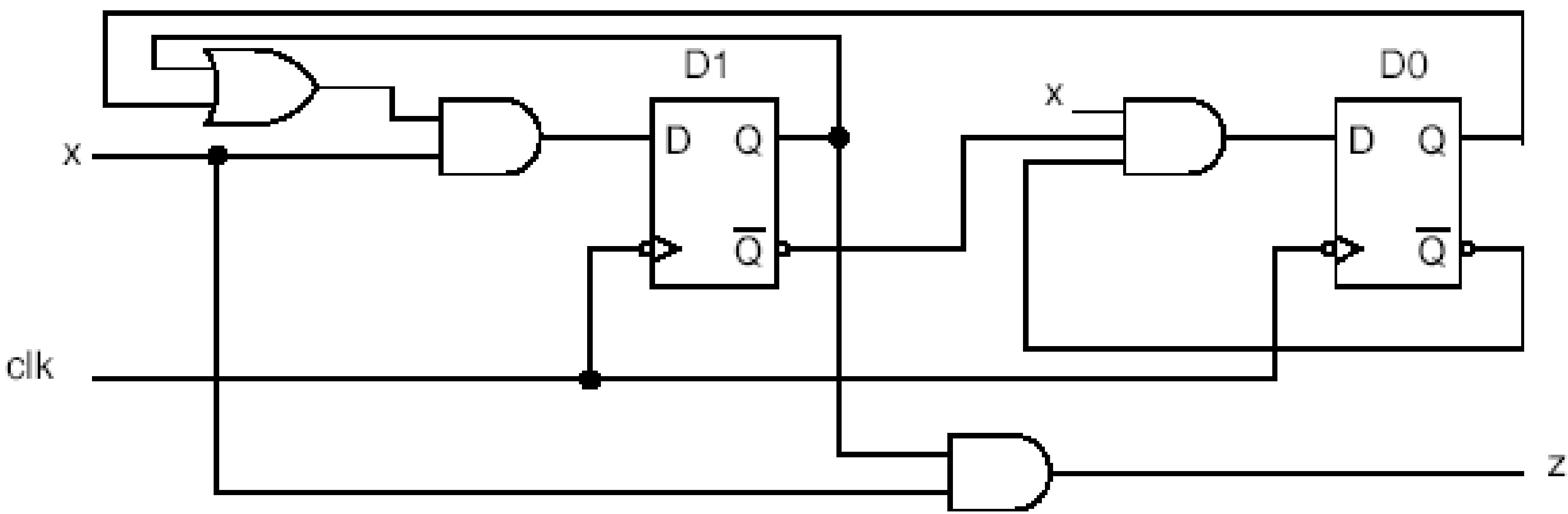
$D_0 = xq_1'q_0'$

q_1q_0 \ x	0	1
00	0	0
01	0	0
11	-	-
10	0	1

z

$z = xq_1$

Paso 6: Dibujar el circuito

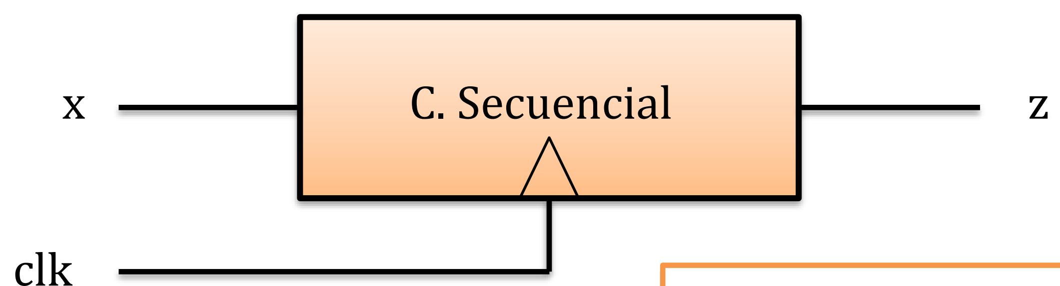


Diseño como M. de Moore

- Mismo problema que antes:
  – Diseñar un circuito con una entrada y una salida que tenga el siguiente comportamiento: La salida se pone a 1 cuando por su entrada se reciben tres o más unos consecutivos
- Recordar:
  – En la M. de Moore las salidas van asociadas al estado
  – Se pondrán dentro del círculo del estado

Entradas y salidas

- El circuito cuenta con una entrada:
  - La llamaremos X
- El circuito cuenta con una salida:
  - La llamaremos Z
- Diseño como M. de Moore

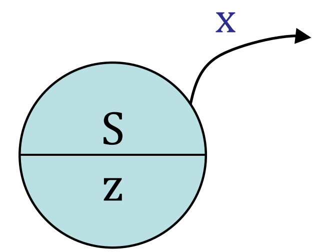


NOTA: Con cada flanco activo de reloj se mira el valor de la entrada

Planteamiento de estados

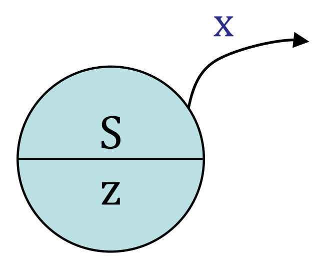
Estado A: No ha llegado ningún “1” (ha llegado un “0”)
Estado B: Ha llegado un “1”
Estado C: Han llegado dos “1” consecutivos
Estado D: Han llegado tres o más “1” consecutivos

NOTA: Se necesita un estado más que con M. Mealy



Planteamiento de transiciones

Cuando estamos en A y recibimos un ‘0’, seguimos en A
Cuando estamos en A y recibimos un ‘1’ pasamos a B
Cuando estamos en B y la entrada es ‘0’ pasamos a A
Cuando estamos en B y la entrada es ‘1’ pasamos a C
Cuando estamos en C y la entrada es ’0’ pasamos a A
Cuando estamos en C y la entrada es ‘1’ pasamos a D
Cuando estamos en D y la entrada es ‘0’ pasamos a A
Cuando estamos en D y la entrada es ‘1’ seguimos en D



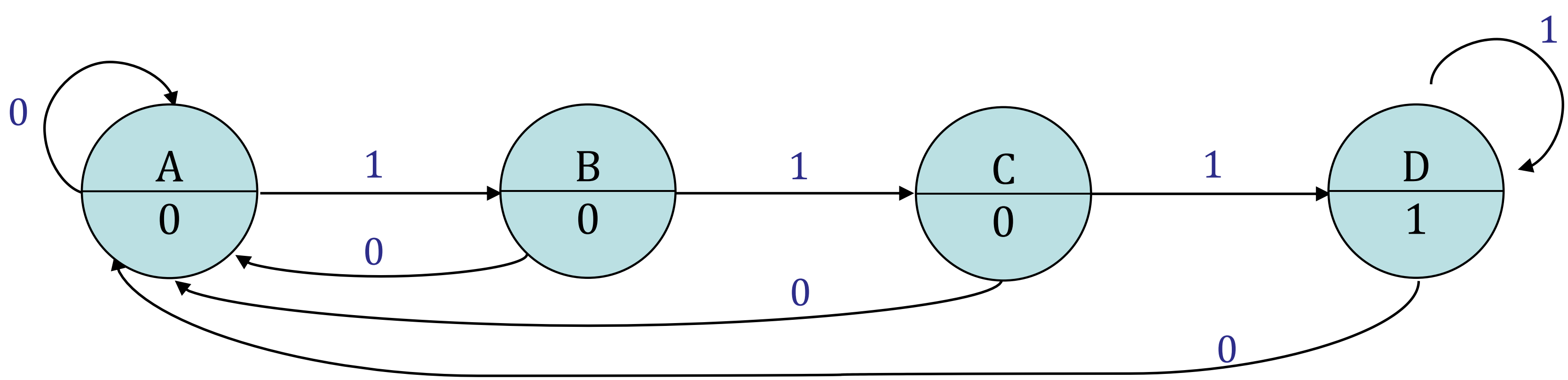


Tabla de estados y salidas

- Lo expresado en el diagrama de estados se puede expresar en forma de tabla:

S \ x	0	1	z
A	A	B	0
B	A	C	0
C	A	D	0
D	A	D	1

NS

- Se asigna a cada estado una combinación de salidas de los biestables
  - Como tenemos 4 estados, necesitamos dos biestables
  - La combinación de estados hay que ponerla en código gray

Asignación de estados

s	q_1q_0
A	00
B	01
C	10
D	11

S \ x	0	1	z
00	00	01	0
01	00	10	0
11	00	11	1
10	00	11	0

Q_1Q_0

Paso 4: Tabla de excitación y salidas

- Se elige el biestable
  - En este ejemplo un biestable JK
- En cada casilla se Introduce el valor de próximo estado del biestable

Tabla de excitación

$q \rightarrow Q$	JK
$0 \rightarrow 0$	0-
$0 \rightarrow 1$	1-
$1 \rightarrow 0$	-1
$1 \rightarrow 1$	-0

q_1q_0 \ x	0	1	z
00	0-,0-	0-,1-	0
01	0-,-1	1-,-1	0
11	-1,-1	-0,-0	1
10	-1,0-	-0,1-	0

$J_1K_{1,}$ J_0K_0

Paso 5: Ecuaciones de excitación y salidas

- Utilizando la tabla de excitación como un mapa de Karnaugh, se obtienen las expresiones de las entradas del biestable y de la salida

$q_1q_0 \backslash x$	0	1
00	0	0
01	0	1
11	-	-
10	-	-

$J_1 = xq_0$

$q_1q_0 \backslash x$	0	1
00	-	-
01	-	-
11	1	0
10	1	0

$K_1 = x'$

$q_1q_0 \backslash x$	0	1
00	0	1
01	-	-
11	-	-
10	0	1

$J_0 = x$

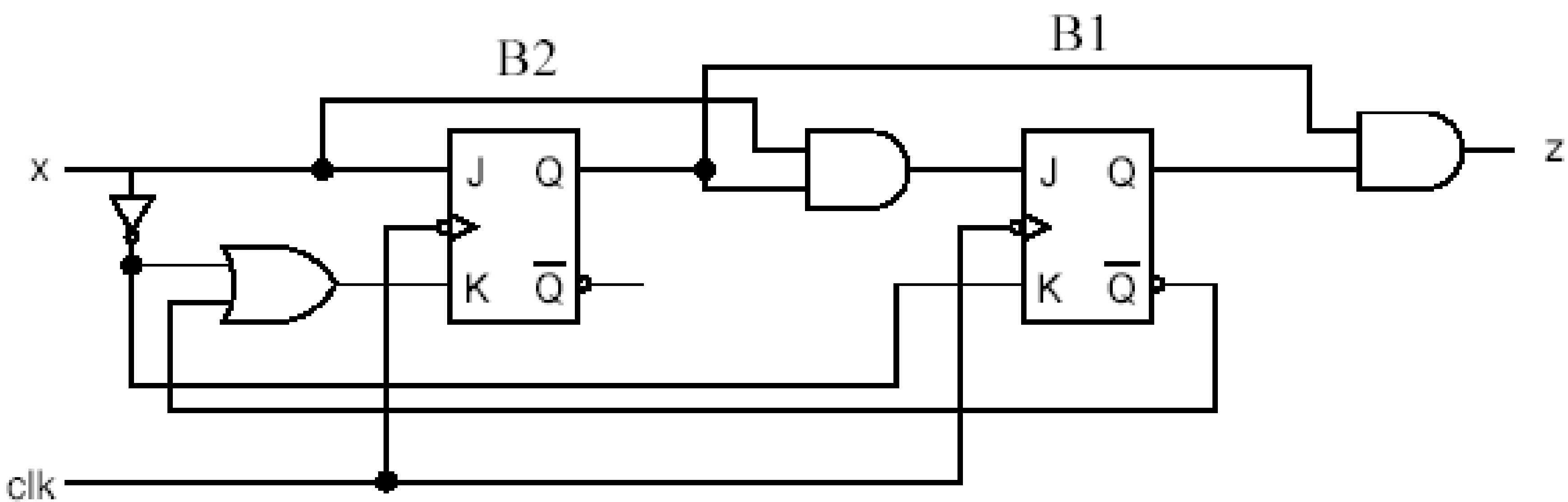
$q_1q_0 \backslash x$	0	1
00	-	-
01	1	1
11	1	0
10	-	-

$K_0 = x' + q_1'$

$q_1q_0 \backslash x$	z
00	0
01	0
11	1
10	0

$z = q_1q_0$

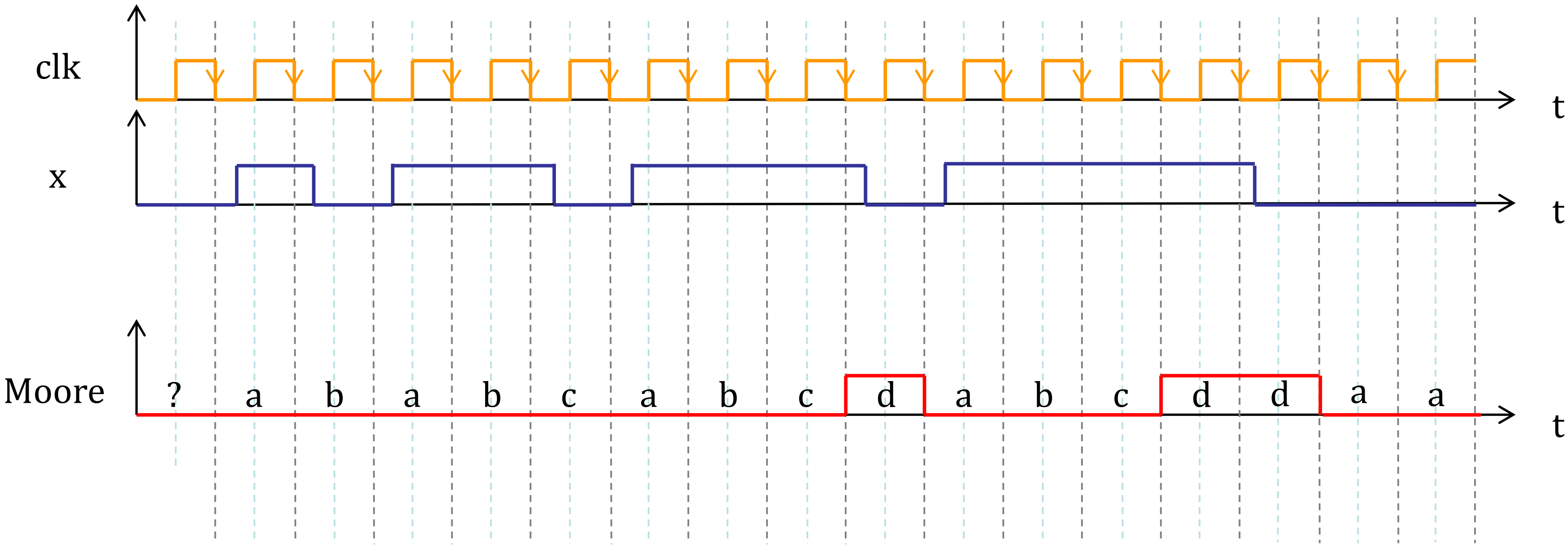
Paso 6: Dibujar el circuito



Evolución temporal

- Vamos a comprobar el funcionamiento de los dos diseños realizados
- Vamos a ver el comportamiento a partir de la tabla de estados
  - Para ello introduciremos un pulso de un ciclo de duración, otro de dos y otro de tres
  - La salida sólo se tiene que poner a ‘1’ con el pulso de tres ciclos de duración

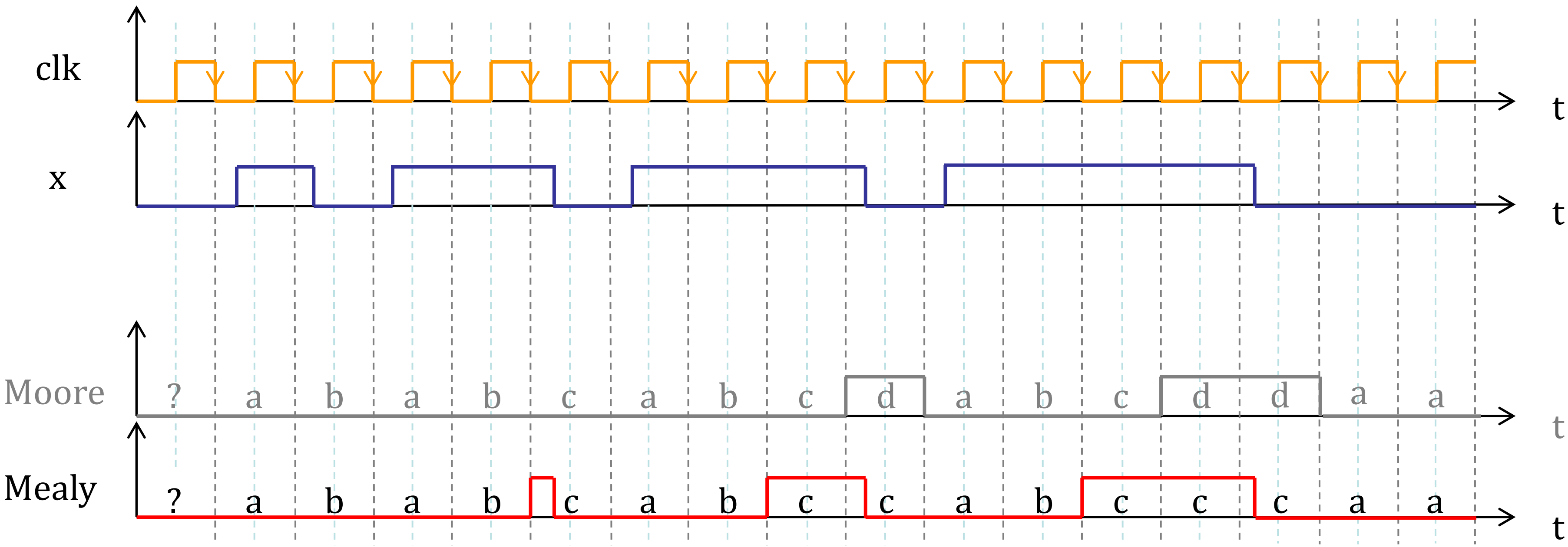
Evolución temporal



S \ x	0	1	z
A	A	B	0
B	A	C	0
C	A	D	0
D	A	D	1

NS

Evolución temporal



S \ x	0	1	z
A	A	B	0
B	A	C	0
C	A	D	0
D	A	D	1

NS

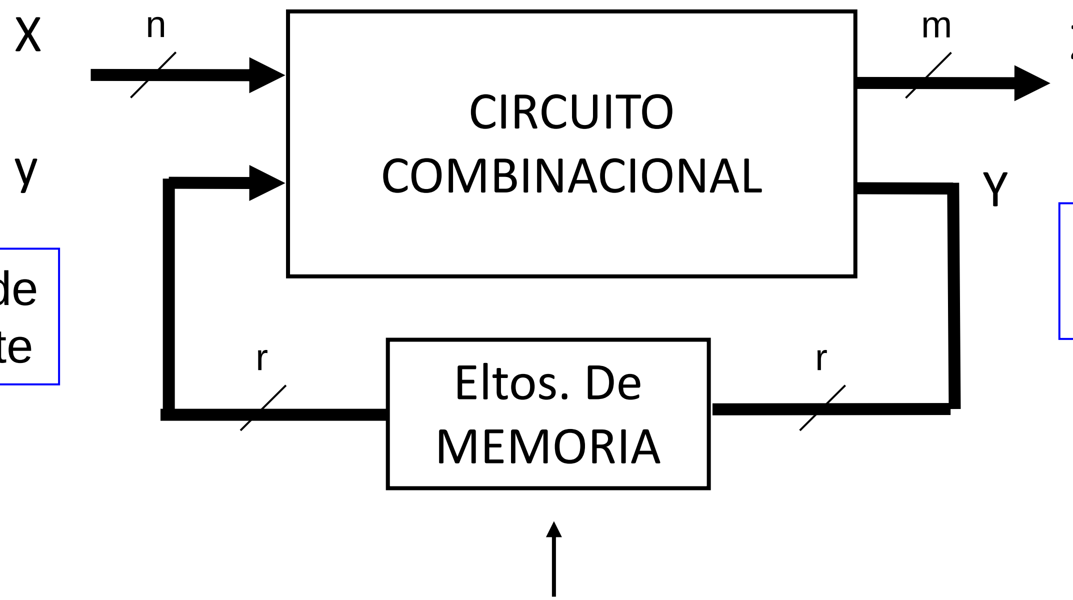
S \ x	0	1
A	A,0	B,0
B	A,0	C,0
C	A,0	C,1

NS, z

Máquinas de estado (recordatorio)

Entradas

Salidas



y ≡ Variables de estado presente

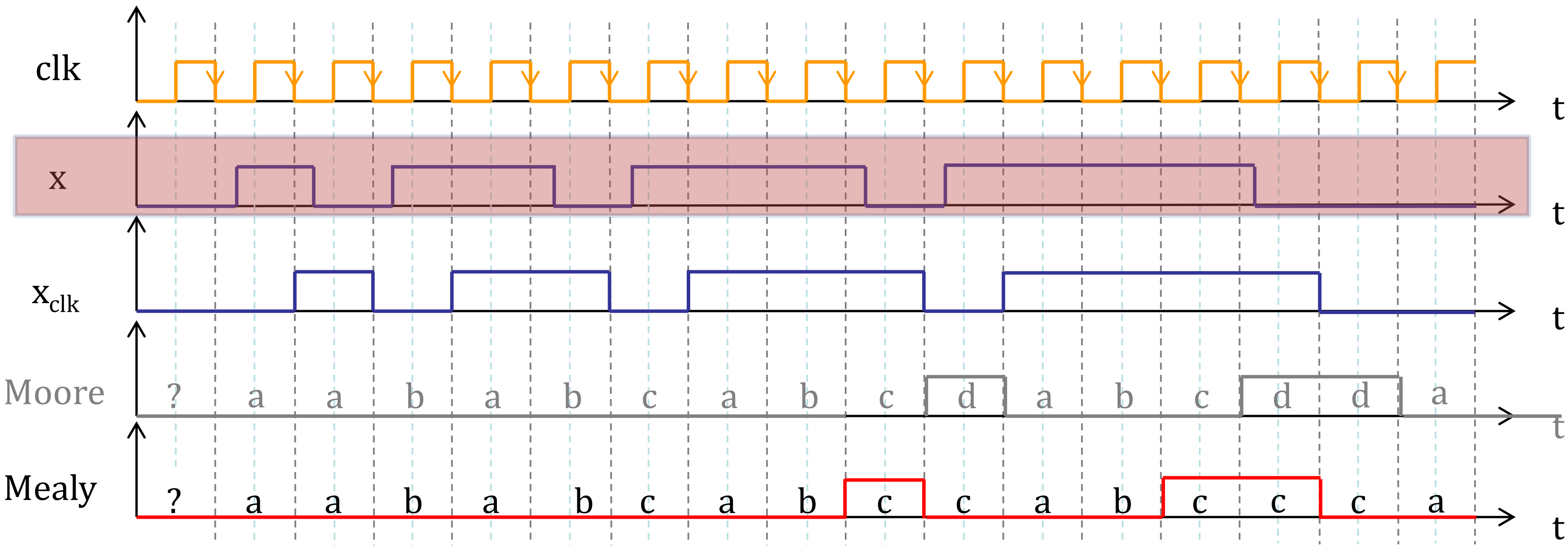
Y ≡ Variables de próximo estado

Almacena el <u>estado interno</u> del sistema

$$z_1(t) = f[x_1(t),..., x_n(t), y_1(t),..., y_r(t)]$$

$$Y_k(t) = f_Y[x_1(t),..., x_n(t), y_1(t),..., y_r(t)]$$

Evolución temporal (entrada sincronizada)



S \ x	0	1	z
A	A	B	0
B	A	C	0
C	A	D	0
D	A	D	1

NS

S \ x	0	1
A	A,0	B,0
B	A,0	C,0
C	A,0	C,1

NS, z

Observaciones al proceso de diseño

- En general, para un sistema dado, los estados del autómata de Mealy no coinciden con los del autómata de Moore
  - Normalmente, el número de estados del autómata de Moore es mayor que el número de estados del autómata de Mealy
- La elección de M. de Moore o M. de Mealy no es fácil
  - En general depende de la experiencia del diseñador
  - En problemas se darán algunas ideas

Directrices para la construcción de diagramas de estado

- En primer lugar construya algunas secuencias de entrada y de salida de ejemplo, para asegurarse de que comprende el enunciado del problema
- Determine bajo qué condiciones, si es que hay alguna, el circuito debe reinicializarse y volver a un estado de partida
- Busque un estado de partida, un estado a partir del cual pueda hacer la evolución del circuito
- Cada vez que se añade una nueva flecha al diagrama de estados hay que determinar si puede ir a uno de los estados previamente definidos o, por el contrario, es preciso añadir un nuevo estado
- Compruebe el diagrama para asegurarse de que para la combinación de valores de las variables de entrada y salida existe una y sólo una ruta saliendo de cada estado
- Cuando el diagrama esté completo, compruébelo aplicando secuencias de entrada y asegurándose de que las secuencias de salida son correctas