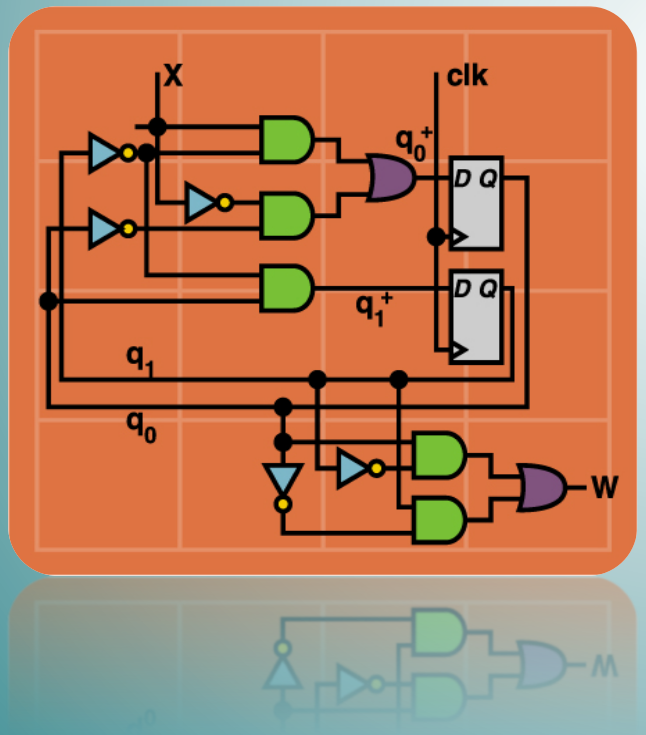
Sistemas Digitales

Tema 4. Circuitos Lógicos Secuenciales

«Digital Design and Computer Architecture» (Harris & Harris). Chapter 3 (3.1 - 3.5)



Pablo Abad
Pablo Prieto Torralbo

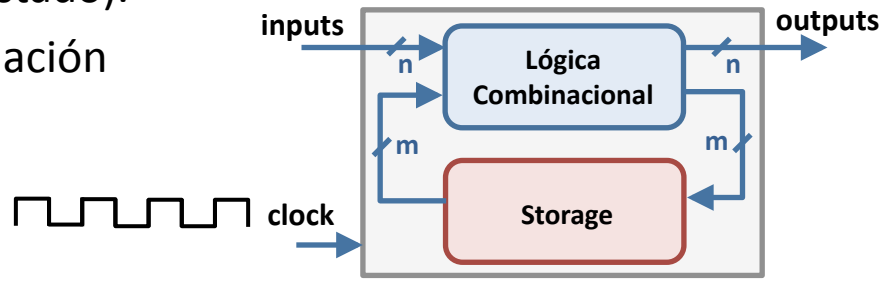
Departamento de Ingeniería Informática y Electrónica

Índice

- **Introducción:**
  - Definición de CLS.
  - Necesidad de Memoria y Sincronización.
- **Latches y Flip Flops.**
- **Máquinas de Estados:**
  - Mealy.
  - Moore.
- **Análisis.**
- **Síntesis:**
  - Mínimo número de biestables.
  - 1 biestable por estado.
- **Timing.**

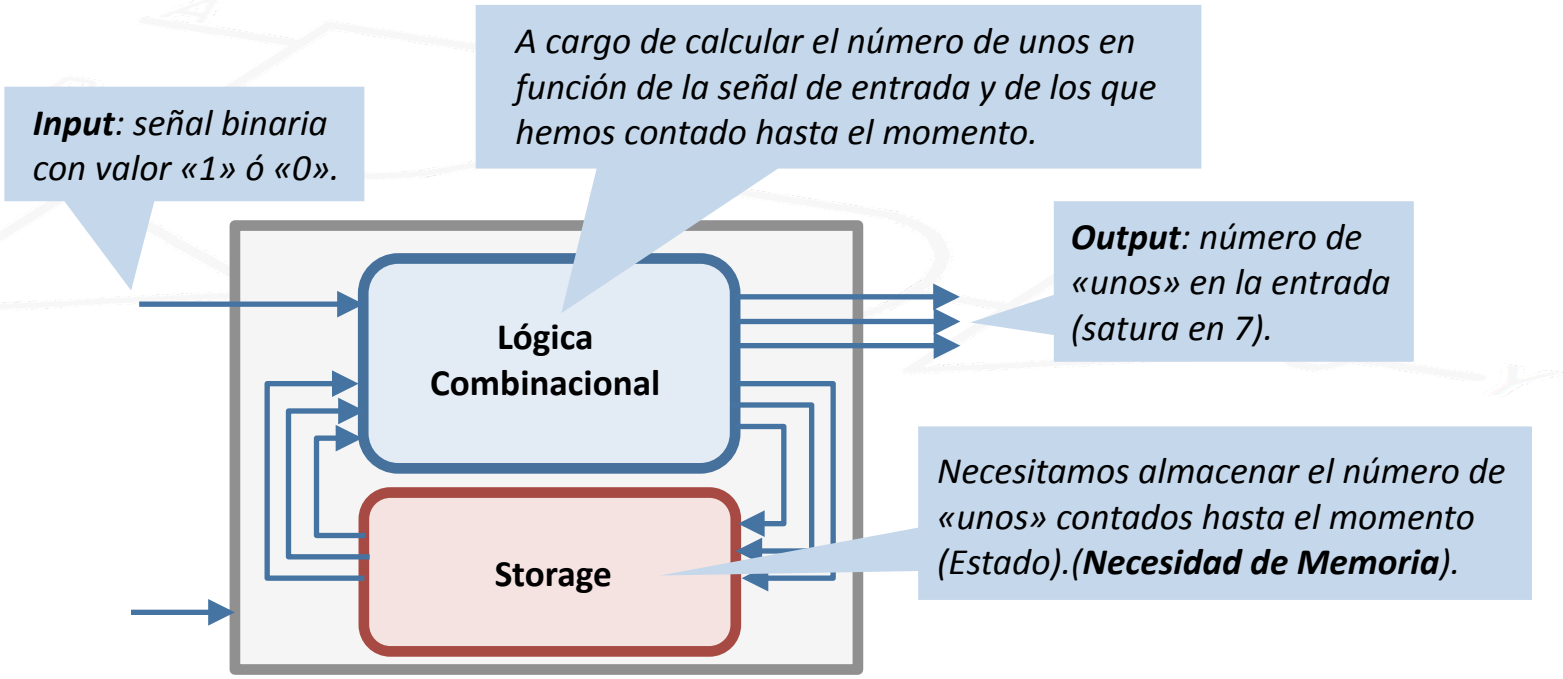
Introducción

- **Circuito Lógico Secuencial (CLS):**
  - Definición: circuito combinacional cuya salida depende de los valores actuales y pasados de las señales de entrada.
  - Se trata de circuitos en los que aparecen lazos de «feedback» (salidas del circuito pueden actuar como valores de entrada).
  - Los Componentes de un CLS son:
    - Señales de entrada y Salida (señales binarias).
    - Señal de Reloj (señal binaria con forma periódica).
    - Lógica Combinacional (determina la salida y el próximo estado).
    - Almacenamiento (mantiene información sobre el estado actual).



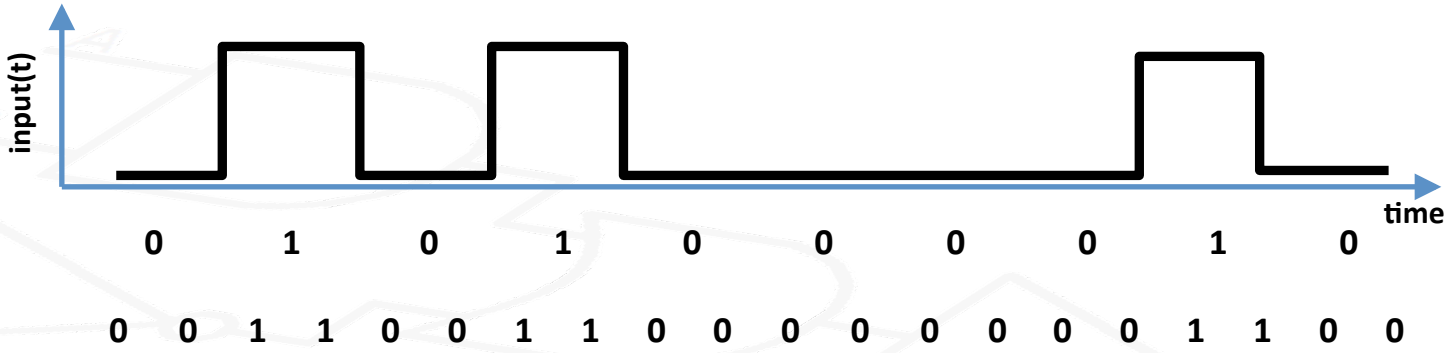
Introducción

- **Ejemplo: Circuito contador de «unos»:**
  - La salida depende del valor que tengo en input (valor actual) y de los «unos» que llevo contados hasta el momento (valor pasado).

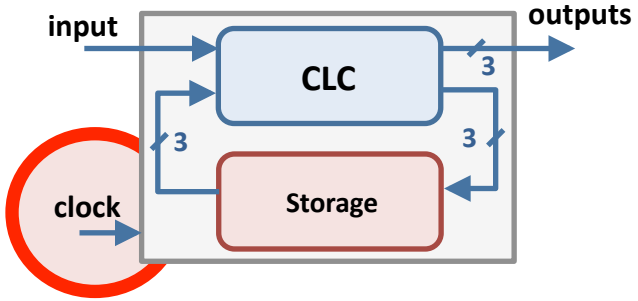


Introducción

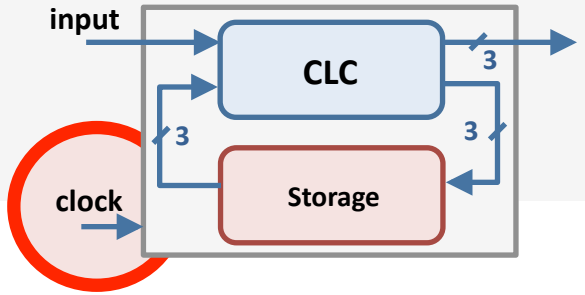
- **Necesidad de Sincronización (Reloj):**
  - **Pregunta:** Con la siguiente señal de entrada, ¿Cuántos unos contaría mi CLS?



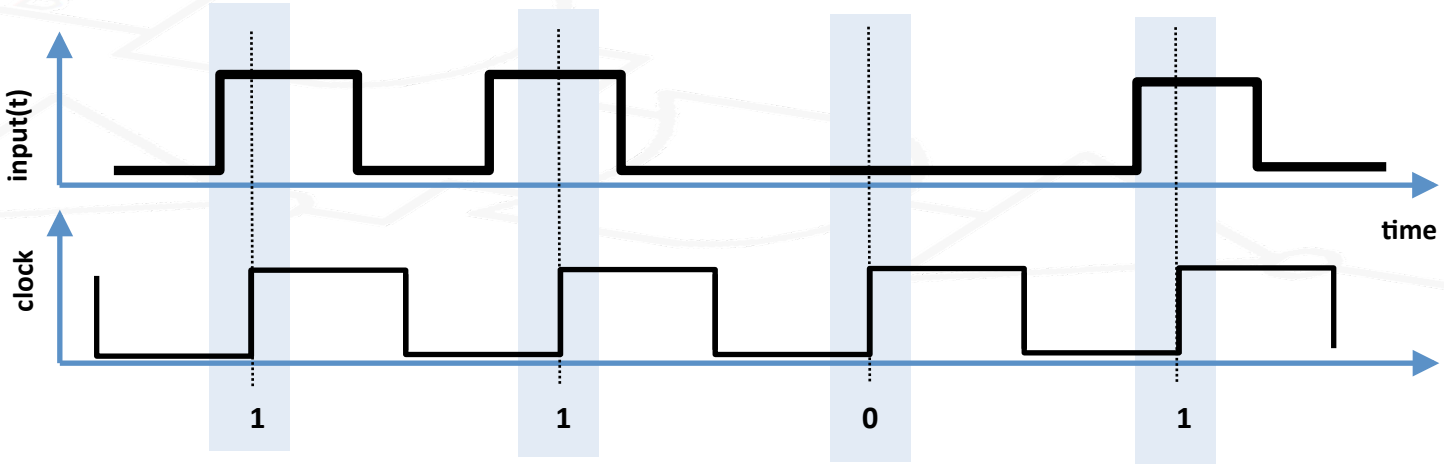
  - **Respuesta**: Nos falta información para poder considerar una respuesta como válida (3?, 6?…).



Introducción



- **Necesidad de Sincronización (Reloj):**
  - La información adicional la proporciona la señal de reloj, que marca en qué precisos instantes se lee la señal de entrada.
  - El reloj convierte la señal de entrada en una señal síncrona.



Señal Síncrona (Dato + Clk) = secuencia de bits.

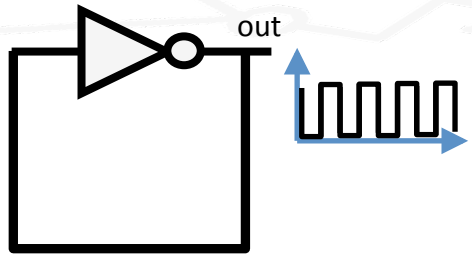
Índice

- **Introducción:**
  - Definición de CLS.
  - Necesidad de Memoria y Sincronización.
- **Latches y Flip Flops.**
- **Máquinas de Estados:**
  - Mealy.
  - Moore.
- **Análisis.**
- **Síntesis:**
  - Mínimo número de biestables.
  - 1 biestable por estado.
- **Timing.**

Latches y Flip Flops

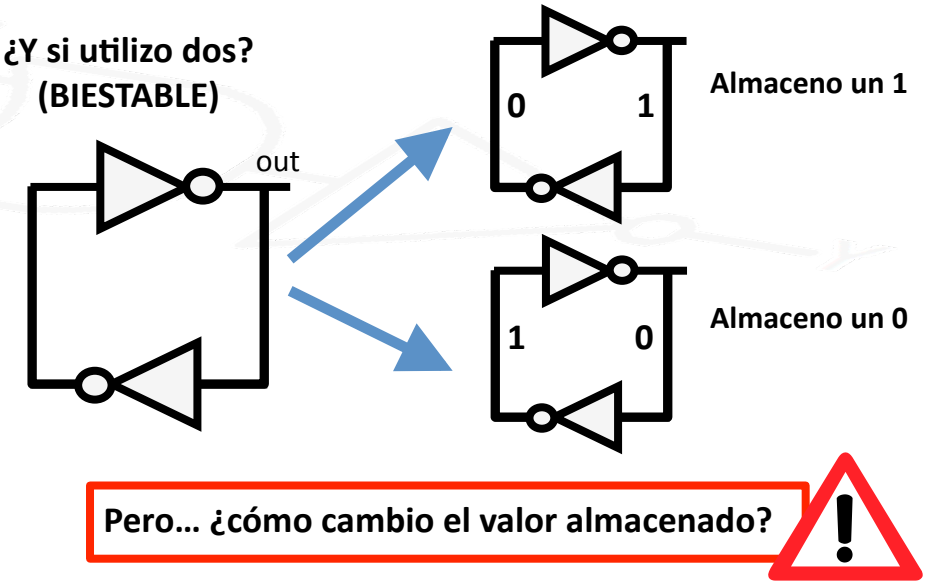
- **Un CLS debe almacenar información que se utilizará en el futuro (estado), nuevo elemento: Memoria.**
- **¿Cómo almacenamos un valor con lógica combinacional?:**
  - Feedback (realimentación): la salida se conecta a la entrada y conseguimos que la señal se mantenga en el tiempo.

¿Con un inversor?



NO, la señal de salida no tiene un valor estable.

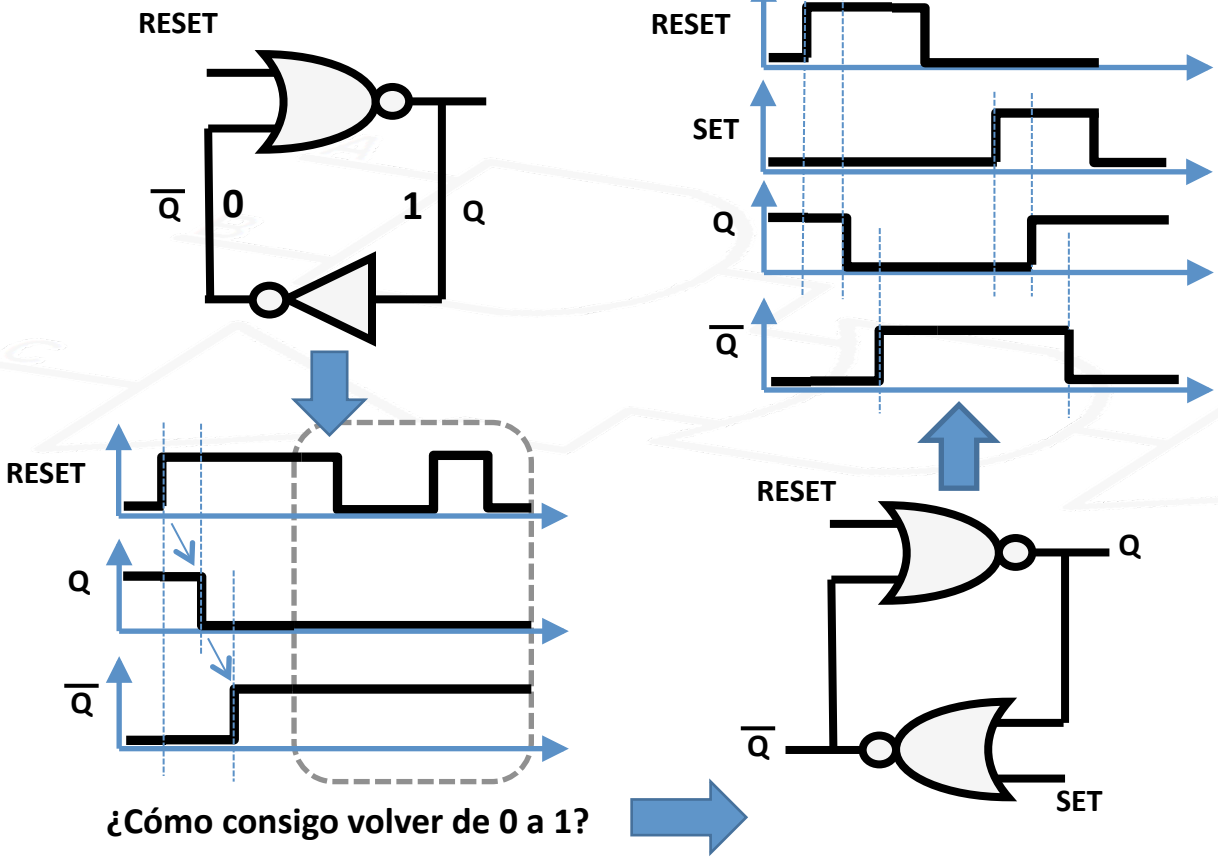
¿Y si utilizo dos? (BIESTABLE)



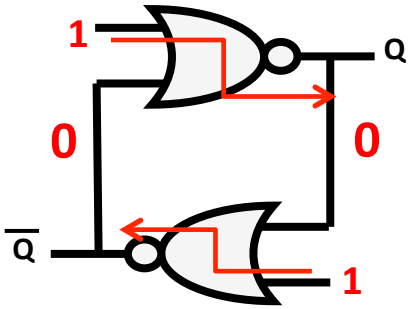
Pero… ¿cómo cambio el valor almacenado?

Latches y Flip Flops

- **Cambio del valor almacenado.**



ATENCIÓN

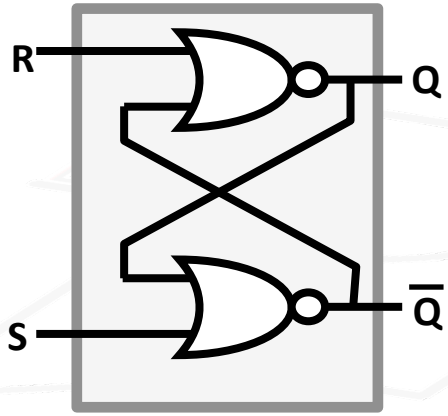


¿Q = $\overline{Q}$?

No es un biestable. El valor final no es predecible.
R = S = 1: prohibido.

Latches y Flip Flops

- **Latch SR:**
  - Elemento de memoria más simple en el diseño de sistemas digitales.



S	R	Q	Q(next)	$\overline{Q}$(next)
0	0	0	0	1
0	0	1	1	0
0	1	X	0	1
1	0	X	1	0
1	1	X	0	0

Biestable: el circuito almacena el valor previamente introducido (memoria). Q(next) = Q.

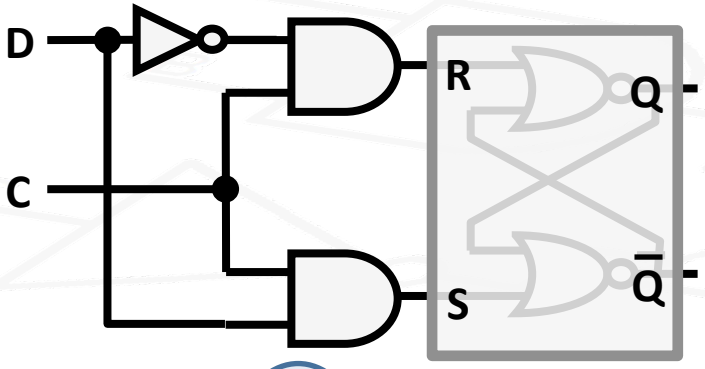
Set/Reset: se graba un nuevo valor de Q. Q(next) puede ser distinto a Q.

Don´t Care: el valor de entrada no influye en la salida (me da igual que sea un 1 o un 0.

Prohibido: valor final no biestable.

Latches y Flip Flops

- **Gated D-Latch:**
  - Eliminamos la posibilidad de alcanzar el estado prohibido.
  - Sigue funcionando como elemento de memoria.



C	D	Q	Q(next)	Q(next)	
0	X	0	0	1	Hold
0	X	1	1	0	
1	0	X	0	1	Set/Reset
1	1	X	1	0	

C	D	R	S
0	0	0	0
0	1	0	0
1	0	1	0
1	1	0	1

nunca 1,1

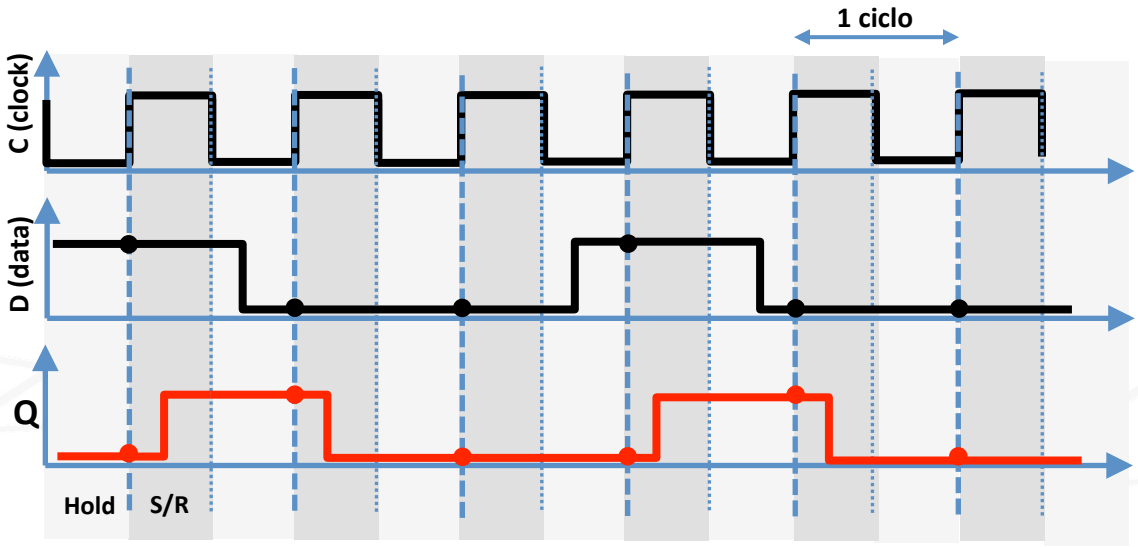
<u>Funcionamiento</u>:

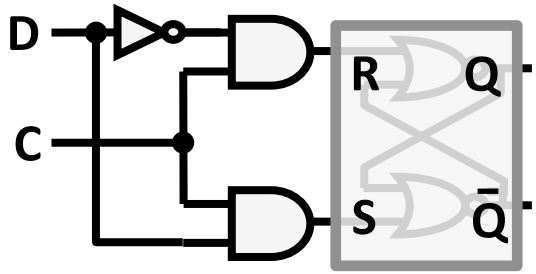
C = 0: mantengo el valor previamente almacenado.

C = 1: Copio el valor de D al interior del lazo.

Latches y Flip Flops

- **Ejemplo de Funcionamiento:**



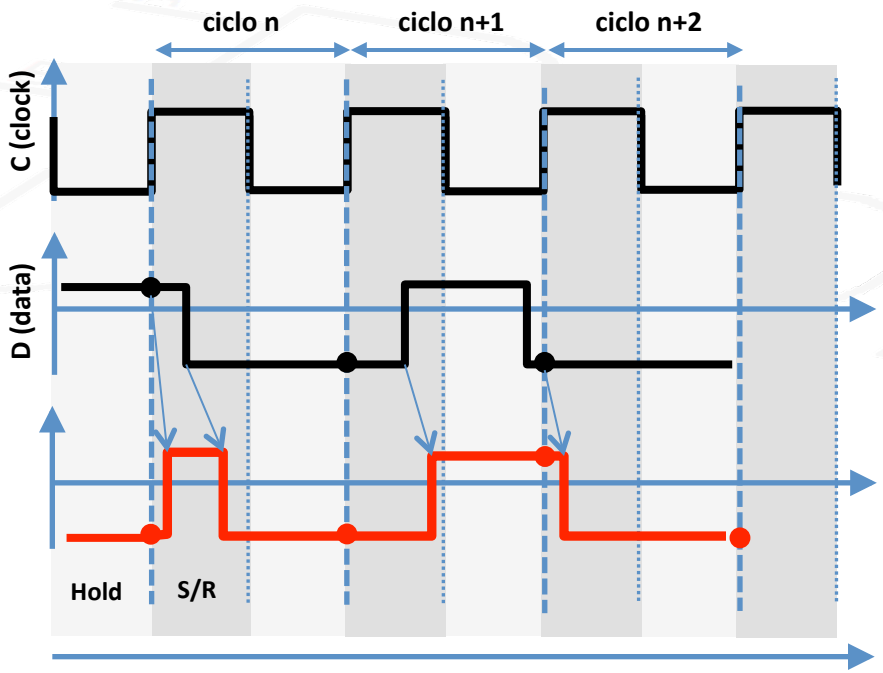


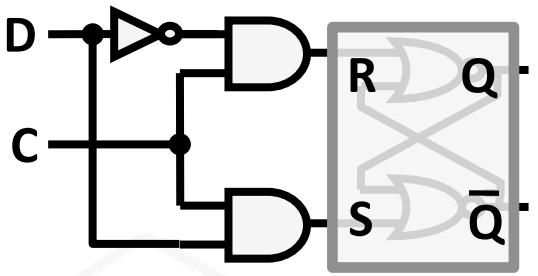
Ciclo	n	n+1	n+2	n+3	n+4	n+5
Bits (D)	1	0	0	1	0	0
Bits (Q)	0	1	0	0	1	0

El valor a la entrada en el *ciclo n* (presente) aparece en la salida en el *ciclo n+1* (futuro).

Latches y Flip Flops

- **Problema:**
  - El comportamiento deseado (Diapositiva anterior) solamente se cumple si la señal de entrada no cambia durante la fase de Set/Reset (clk = 1).



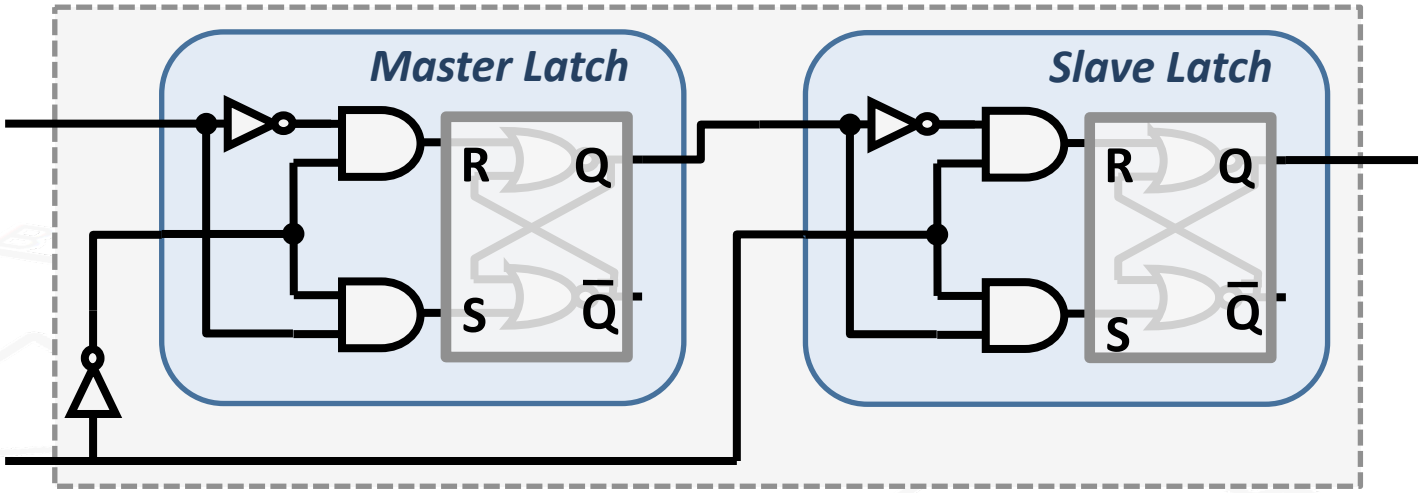


Ciclo	n	n+1	n+2
Bits (D)	1	0	0
Bits (Q)	0	0	1

¿SOLUCIÓN?

Latches y Flip Flops

- **Biestable D activado por flanco (Flip Flop):**

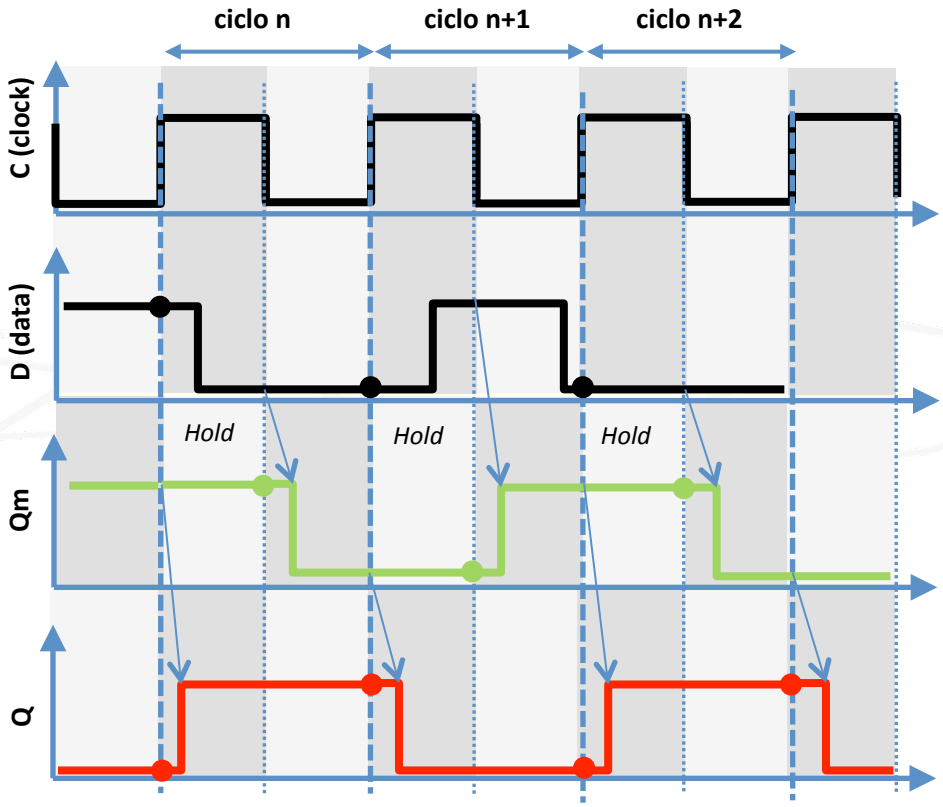


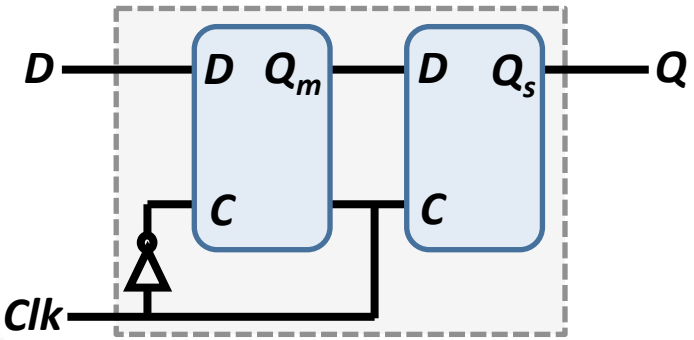
Segunda fase del ciclo (clk = 0):
el siguiente estado, en la entrada del circuito, es almacenado en el Master Latch.

Primera Fase del ciclo (clk = 1):
el estado previamente calculado se convierte en el estado actual, y es enviado a la salida.

Latches y Flip Flops

- **Biestable D activado por flanco (Flip Flop):**



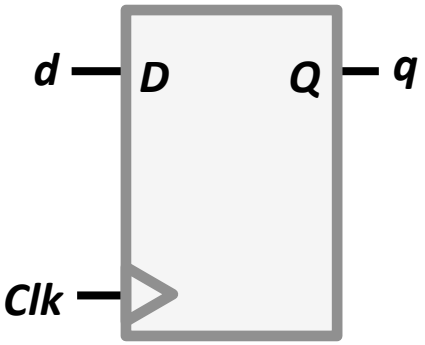


Ciclo	n	n+1	n+2
Bits (D)	1	0	0
Bits (Q)	0	1	0

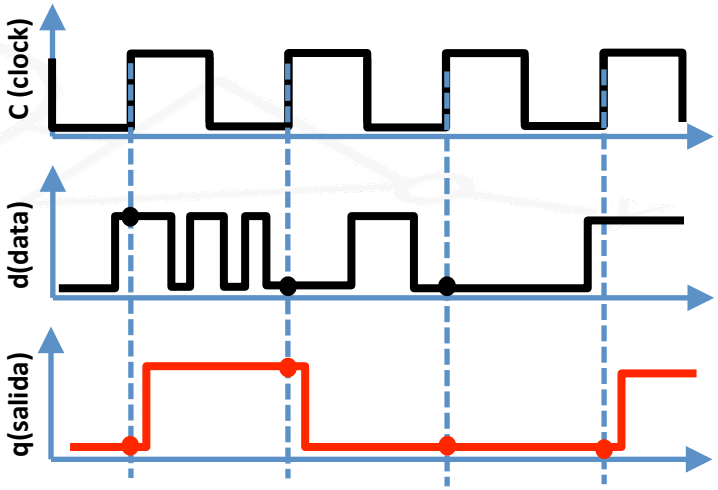
¡¡CORRECTO!!

Latches y Flip Flops

- **Biestable D activado por flanco (Flip Flop):**
  - Elemento básico de memorización (1 bit).
  - Ecuación característica: $\mathbf{q^+ = d}$
    - q+: estado siguiente (ciclo actual+1).
    - d: valor de entrada actual.

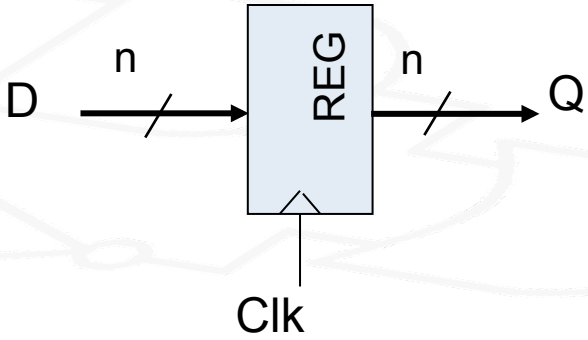


El valor presente en d cuando el reloj hace un flanco ascendente se copia en el Flip Flop, y se mantiene hasta el siguiente flanco ascendente (independientemente de lo que haga la señal de entrada).



Latches y Flip Flops

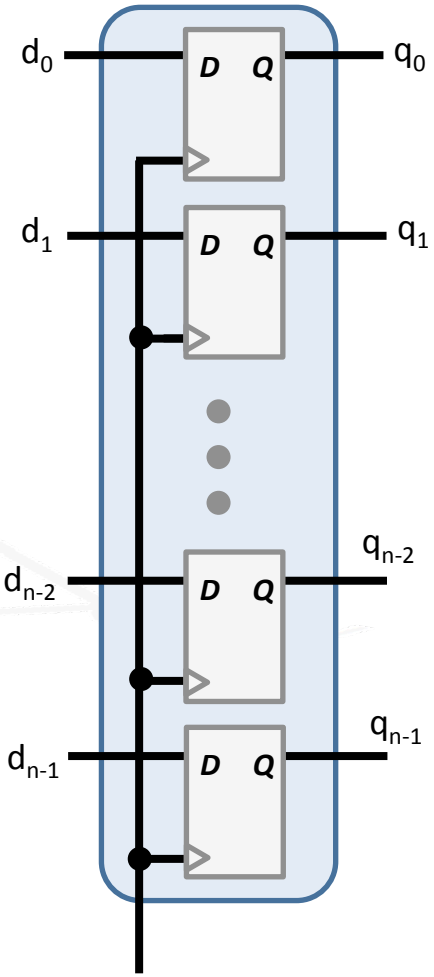
- **Registro:**
  - Elemento de memorización de n bits.
  - Formado por n Flip Flops conectados en paralelo.



$D = d_{n-1}, d_{n-2}, \ldots d_1, d_0$

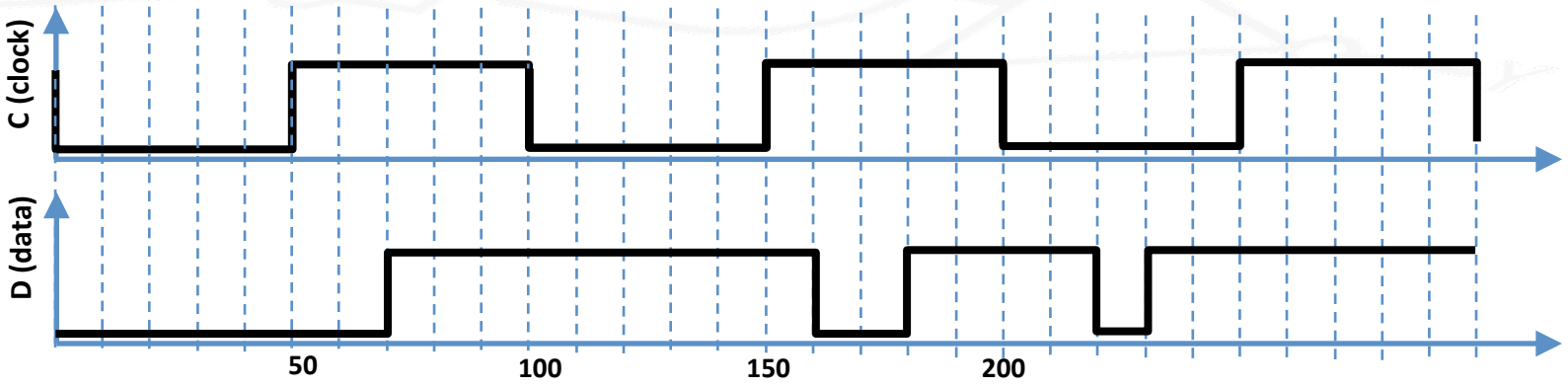
$Q = q_{n-1}, q_{n-2}, \ldots q_1, q_0$

$Q^+ = D$



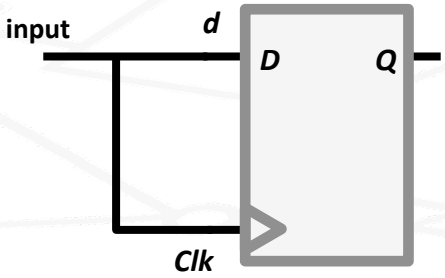
Latches y Flip Flops

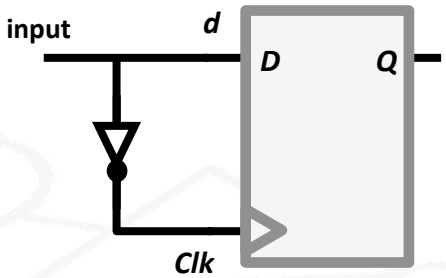
- **Ejercicio 1:**
  - Dadas las señales de reloj y de entrada del cronograma inferior, completa la forma de la señal de salida si el elemento de memoria utilizado es un latch-D. El tiempo de propagación de dicho elemento es 10 u.t.
  - Repite el proceso anterior para un Flip Flop con el mismo tiempo de propagación.
  - ¿Cómo cambia la señal de salida si tp = 0?

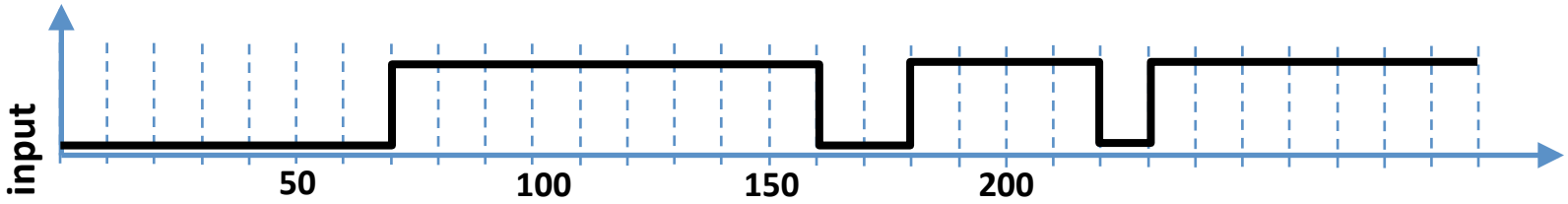


Latches y Flip Flops

- **Ejercicio 2:**
  - Dadas los CLS de las figuras inferiores y el cronograma de la señal de entrada, completa la forma de la señal de salida en ambos casos (delay NOT: 10 u.t., delay Flip Flop: 20 u.t.).





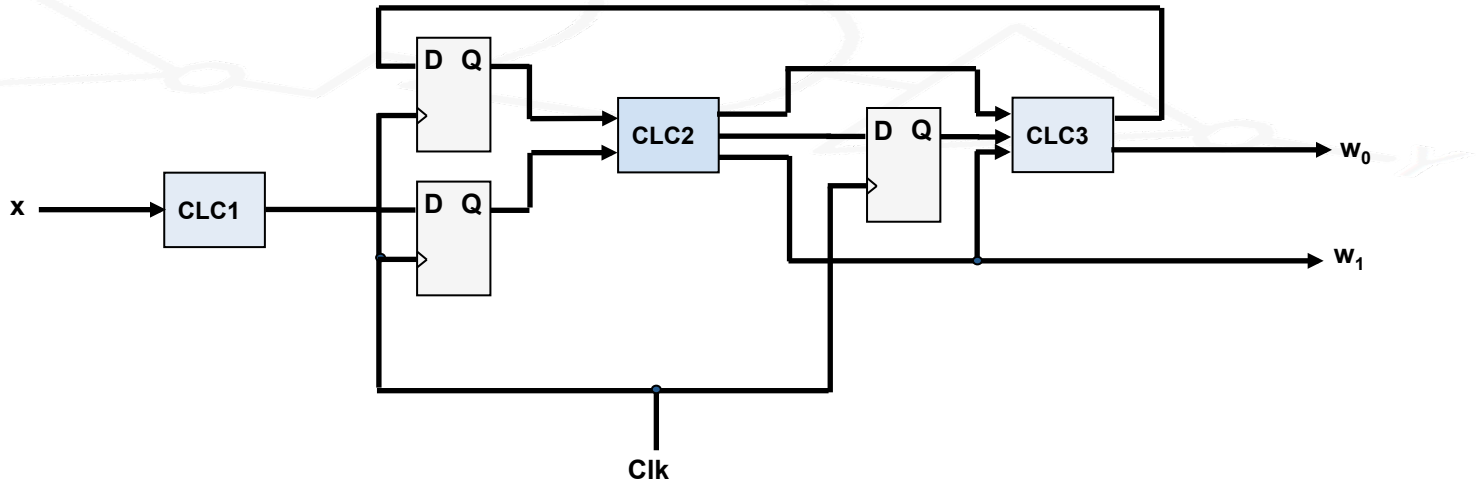


Índice

- **Introducción:**
  - Definición de CLS.
  - Necesidad de Memoria y Sincronización.
- **Latches y Flip Flops.**
- **Máquinas de Estados:**
  - Mealy.
  - Moore.
- **Análisis.**
- **Síntesis:**
  - Mínimo número de biestables.
  - 1 biestable por estado.
- **Timing.**

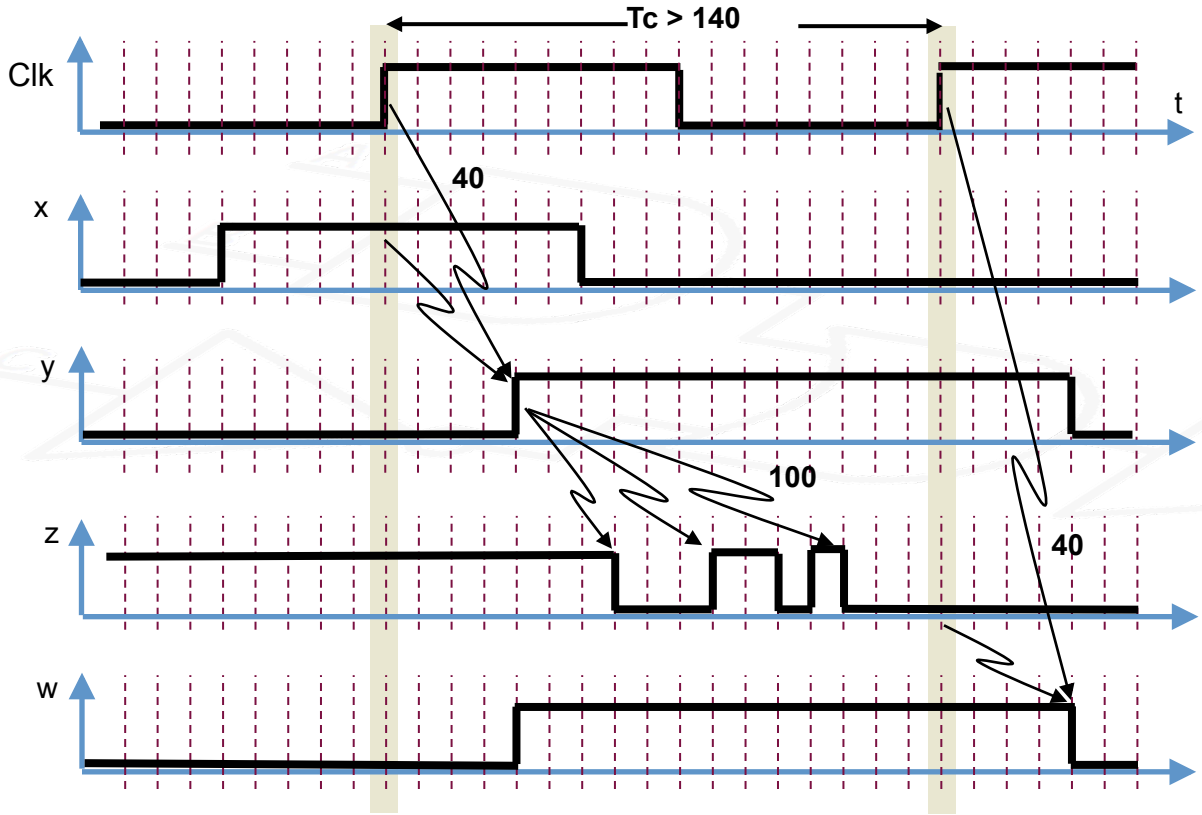
Máquinas de Estados

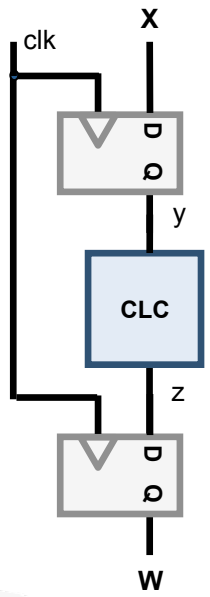
- **Circuito Secuencial Síncrono:**
  - Red de combinacionales y biestables conectados entre sí.
  - Puede haber caminos cíclicos, pero tienen que atravesar al menos un biestable.
  - Todos los biestables utilizan la misma señal de reloj.
  - Las señales de entrada se sincronizan con el mismo reloj.



Máquinas de Estados

- **Circuito Secuencial Síncrono: Tiempo de ciclo.**





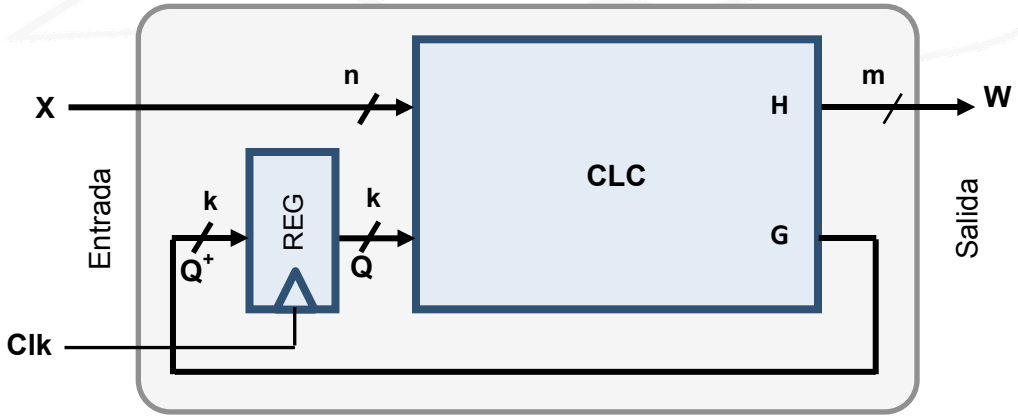
Tp(FF) = 40 u.t.

Tp(CLC) = 100 u.t.

Tc ≥ Tp(FF) + Tp(CLC)

Máquinas de Estados

- **Máquinas de Estados (FSM):**
  - Cualquier CLS se puede representar con un esquema como el de la figura inferior (**Modelo de Mealy**):
    - Agrupando todos los circuitos combinacionales en un único CLC y todos los biestables (Flip Flops) en un único REG.
  - Se denomina FSM porque un circuito con K registros solo puede estar en un número finito (2^K) de estados.



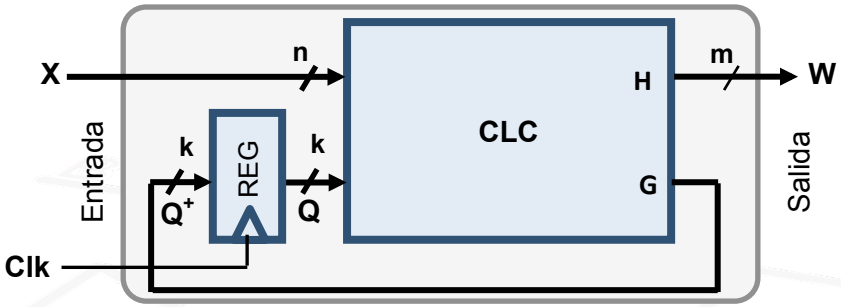
Modelo de Mealy:

$W = H(X,Q)$
Ec. de salida

$Q^+ = G(X,Q)$
Ec. estado siguiente

Máquinas de Estados

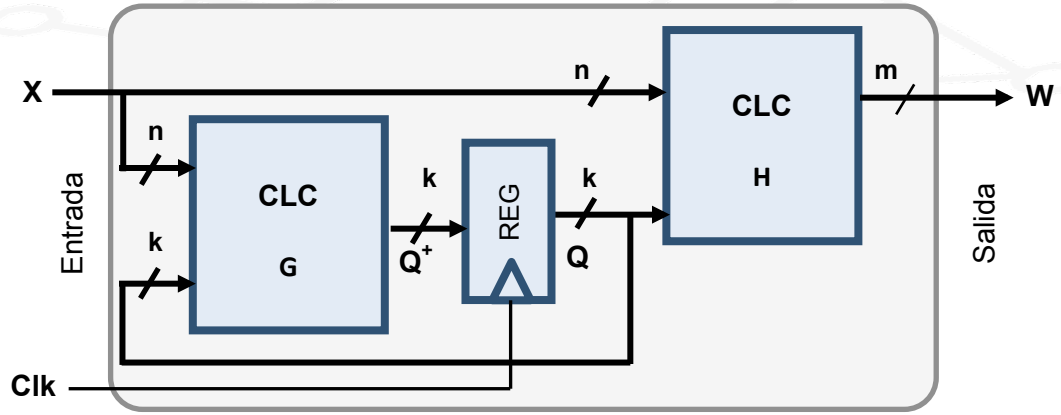
- **Representación alternativa del CLS de Mealy:**
  - Separando las funciones H y G.



Modelo de Mealy:

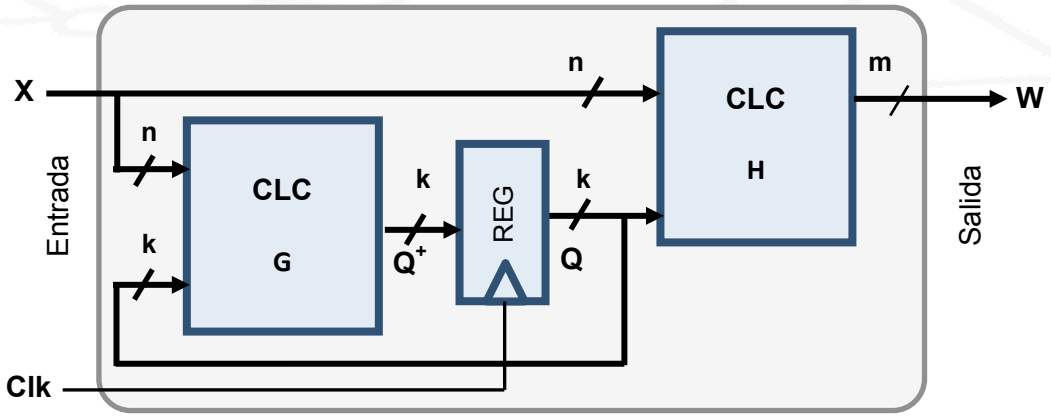
$W = H(X,Q)$
Ec. de salida

$Q^+ = G(X,Q)$
Ec. estado siguiente



Máquinas de Estados

- **Especificación de un CLS:**
  - **Qué datos necesitamos para caracterizar el circuito:**
    - número de entradas: $X=(x_{n-1}, x_{n-2},\ldots,x_1,x_0)$.
    - número de salidas: $W=(w_{m-1}, w_{m-2},\ldots,w_1,w_0)$.
    - número de estados: $Q=(q_{k-1}, q_{k-2},\ldots,q_1,q_0)$.
    - Tabla de verdad de W: CLC H.
    - Tabla de verdad de Q^+: CLC G.
    - Estado inicial: Valor inicial de Q.



Máquinas de Estados

- **Ejemplo de Especificación de un CLS:**
  - X=(x_0), W=(w_1,w_0), Q=(q_1,q_0), Estado inicial (0,0).

- Tabla de verdad de W

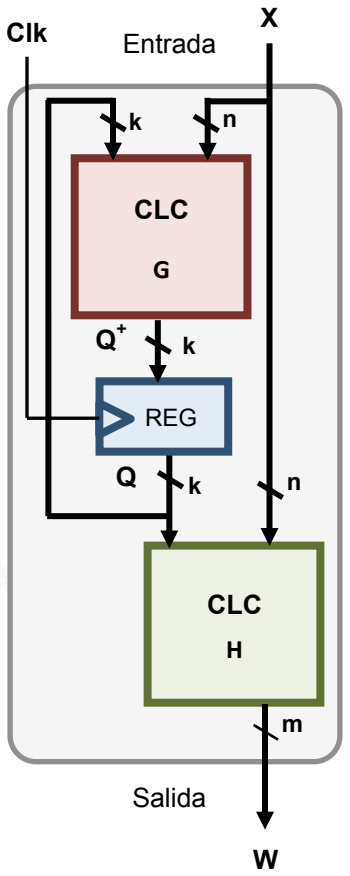
q_1	q_0	x_0	w_1	w_0
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	1	1
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	0

Tabla de Salida

- Tabla de verdad de Q^+

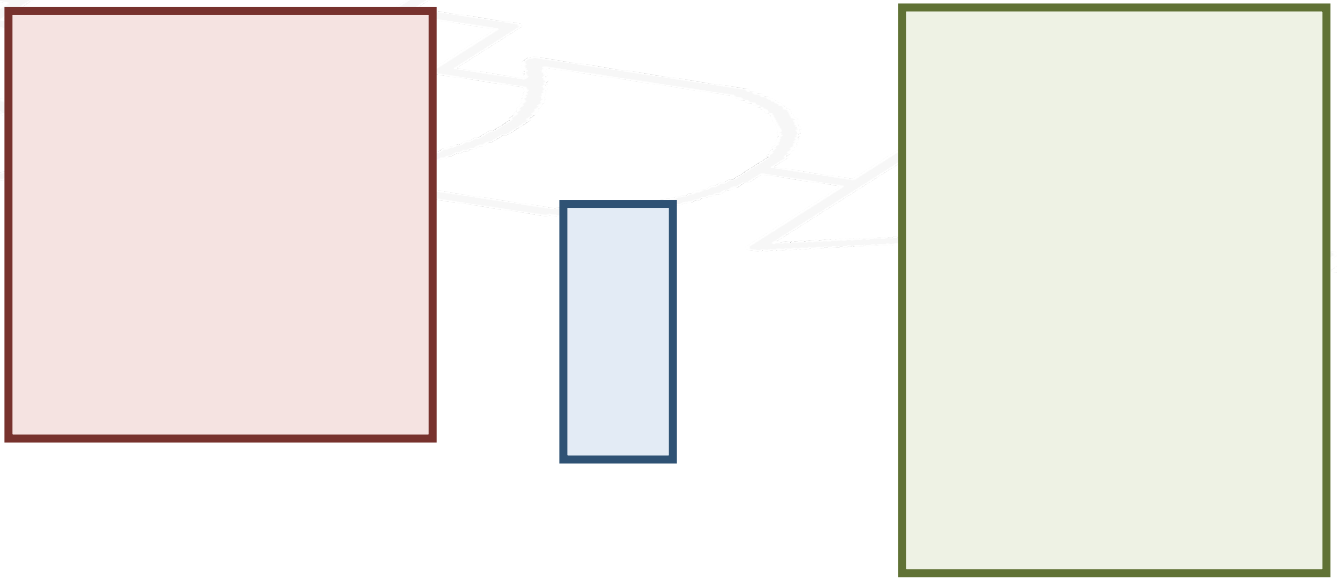
q_1	q_0	x_0	q_1^+	q_0^+
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	0	1
1	0	0	0	1
1	0	1	0	0
1	1	0	x	x
1	1	1	x	x

Tabla de Transición (estado siguiente)



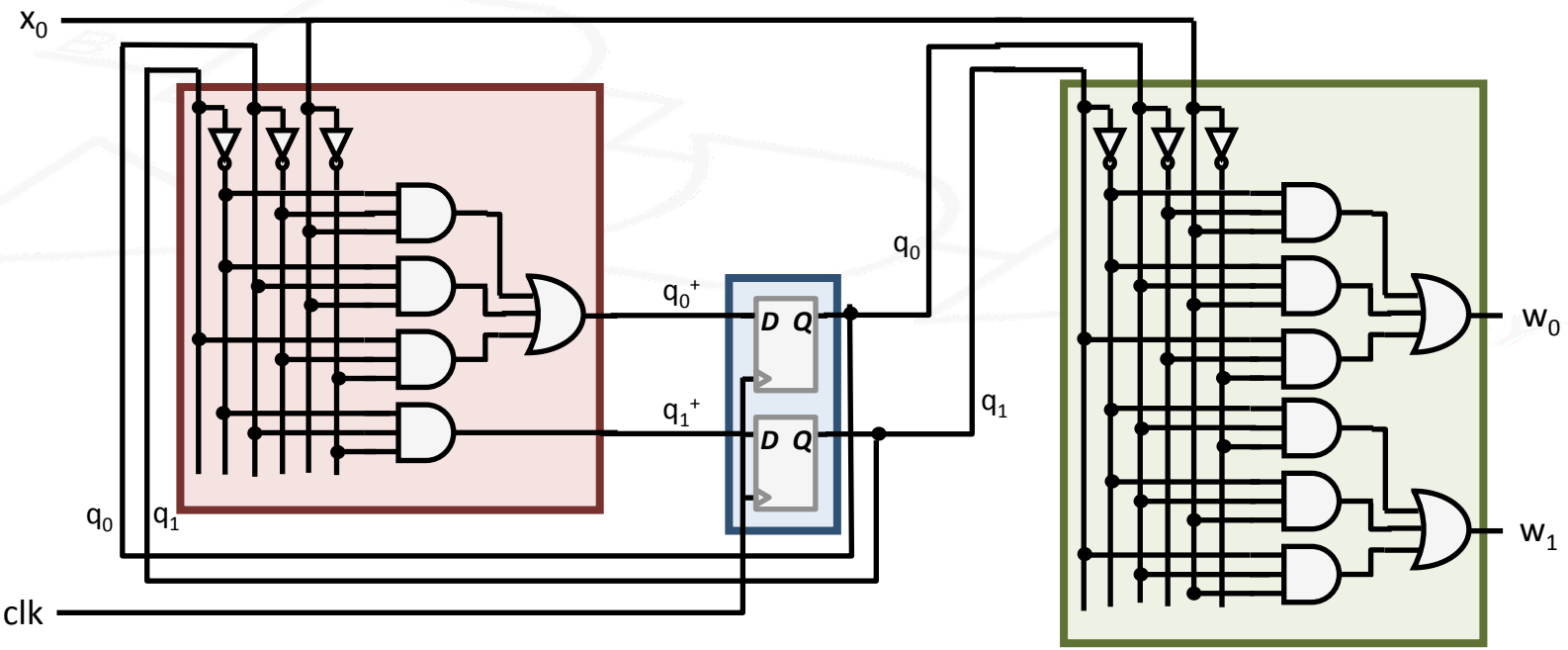
Máquinas de Estados

- **Ejercicio 1:**
  - Dada la especificación de un CLC de la diapositiva anterior, implementar el circuito con los siguientes componentes: puertas NOT, AND y OR de 3 entradas y Biestables D activados por flanco.



Máquinas de Estados

- **Ejercicio 1:**
  - Dada la especificación de un CLC de la diapositiva anterior, implementar el circuito con los siguientes componentes: puertas NOT, AND y OR de 3 entradas y Biestables D activados por flanco.



Máquinas de Estados

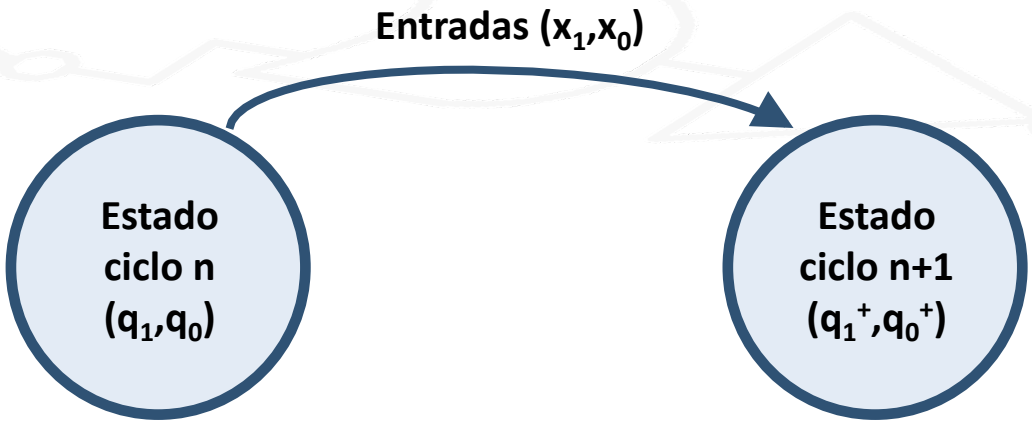
- **Ejercicio 2:**
  - Dada la especificación de un CLC de la diapositiva anterior, rellena la siguiente tabla con los valores de las señales de salida en cada ciclo de reloj.

Ciclo	n	n+1	n+2	n+3	n+4	n+5
x_0	0	1	1	0	1	1
w_0						
w_1						

Máquinas de Estados

- **El Grafo de Estados:**
  - Representación alternativa de la Tabla de Transición.
  - q_1^+(*ciclo n*)= q_1(*ciclo n+1*).
  - Facilita el proceso de síntesis: a partir de una descripción textual es más fácil inferir el grafo que la T.V.

Formato:

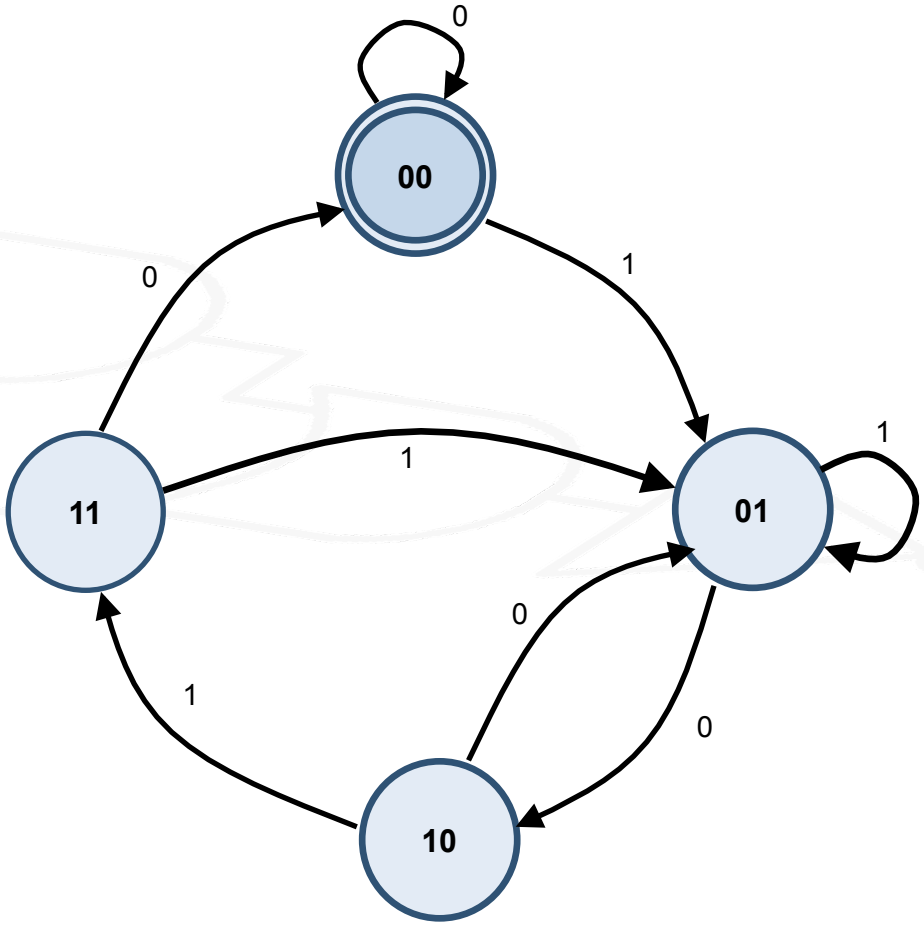


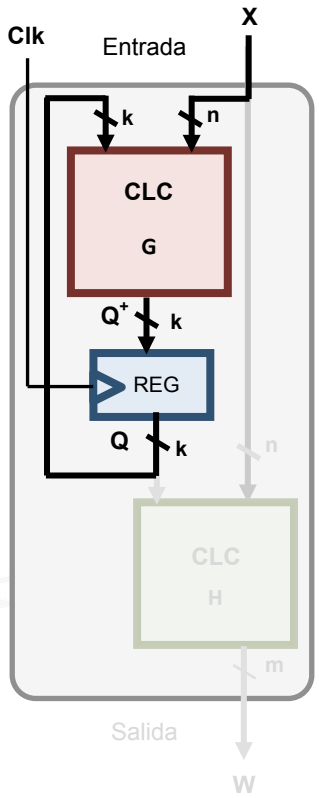
Máquinas de Estados

- **De la T.V. al Grafo de Estados:**

Tabla de Transición

q_1	q_0	x_0	q_1^+	q_0^+
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	0	1
1	0	0	0	1
1	0	1	1	1
1	1	0	0	0
1	1	1	0	1





Máquinas de Estados

- **Ejemplo de Especificación de un CLS:**
  - $X = (x_0)$, $W = (w_1,w_0)$, $Q = (q_1,q_0)$, Estado inicial (0,0).

- Tabla de verdad de W

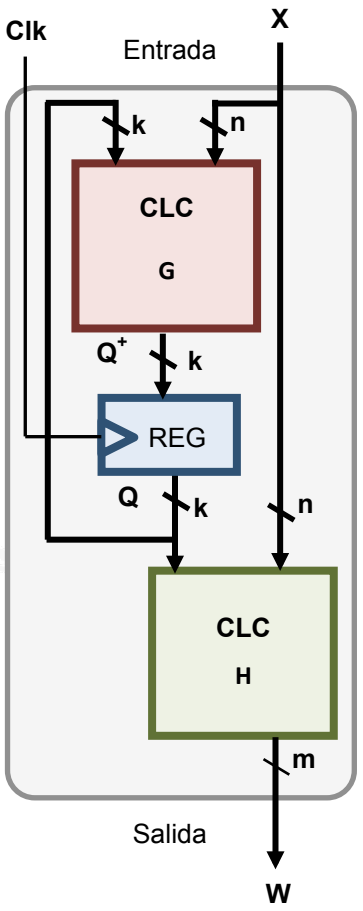
q_1	q_0	x_0	w_1	w_0
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	1	1
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	0

Tabla de Salida

- Tabla de verdad de Q^+

q_1	q_0	x_0	q_1^+	q_0^+
0	0	0	0	0
0	0	1	0	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	1	0
1	1	0	x	x
1	1	1	x	x

Tabla de Transición (estado siguiente)



Máquinas de Estados

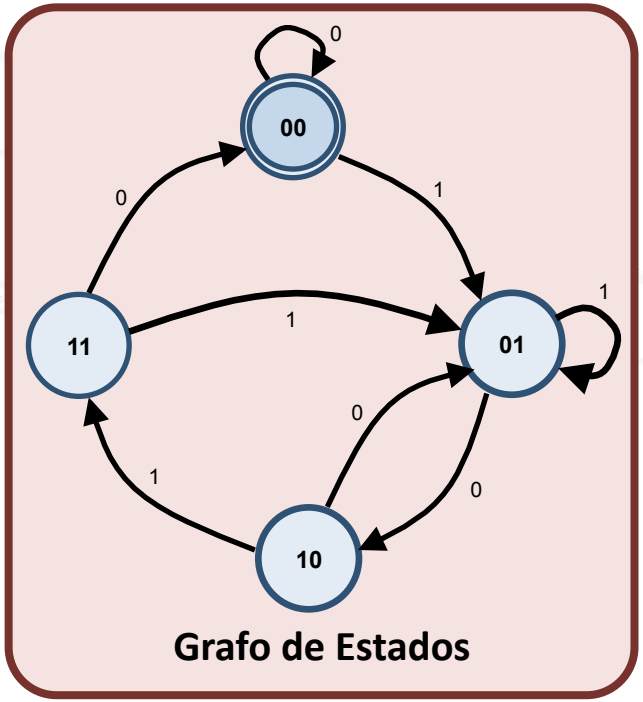
- **Ejemplo de Especificación de un CLS:**
  - X = (x_0), W = (w_1,w_0), Q = (q_1,q_0), Estado inicial (0,0).

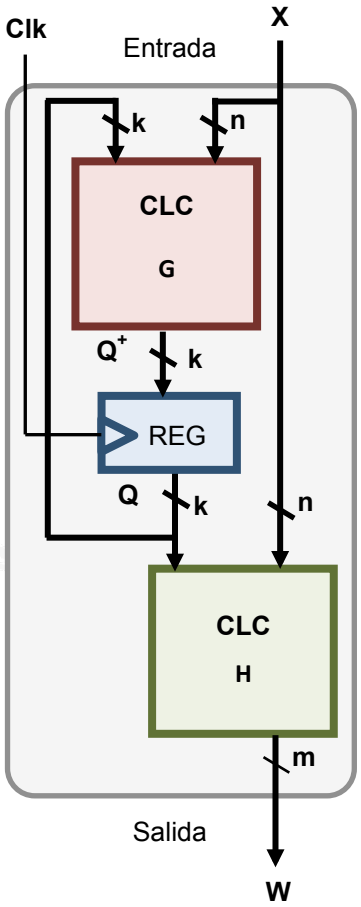
- Tabla de verdad de W

q_1	q_0	x_0	w_1	w_0
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	1	1
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	0

Tabla de Salida

- Tabla de verdad de Q^+





Máquinas de Estados

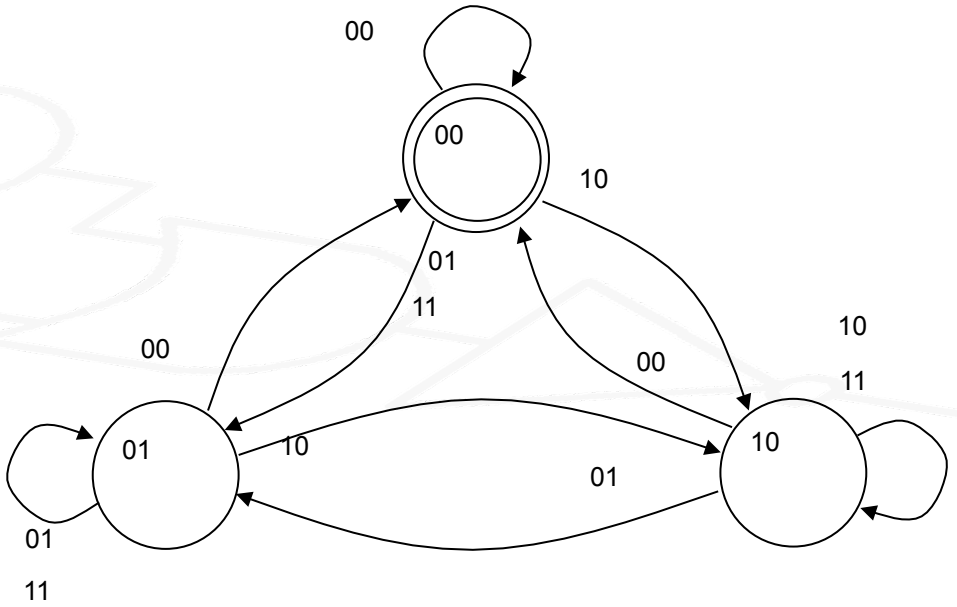
- **Ejercicio:**
  - Dada la siguiente Tabla de Transición de un CLS, dibuja su grafo de estados.

q_1	q_0	x_1	x_0	q_1^+	q_0^+
0	0	0	0	0	0
0	0	0	1	0	1
0	0	1	0	1	0
0	0	1	1	0	1
0	1	0	0	0	0
0	1	0	1	0	1
0	1	1	0	1	0
0	1	1	1	0	1
1	0	0	0	0	0
1	0	0	1	0	1
1	0	1	0	1	0
1	0	1	1	1	0
1	1	0	0	0	0
1	1	0	1	0	0
1	1	1	0	0	0
1	1	1	1	0	0

Máquinas de Estados

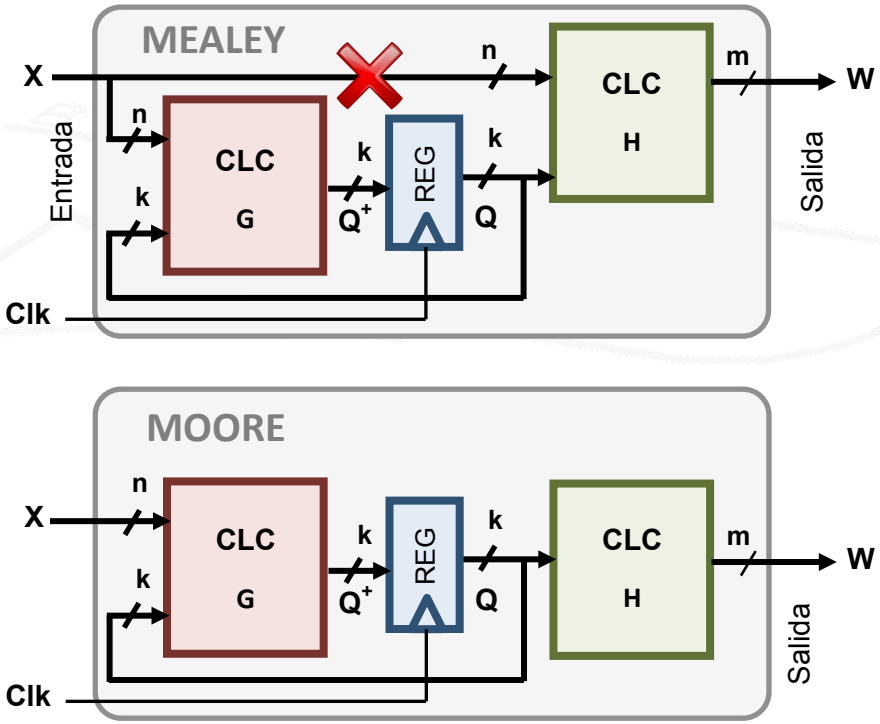
- **Ejercicio:**
  - Dada la siguiente Tabla de Transición de un CLS, dibuja su grafo de estados.

q_1	q_0	x_1	x_0	q_1^+	q_0^+
0	0	0	0	0	0
0	0	0	1	0	1
0	0	1	0	1	0
0	0	1	1	0	1
0	1	0	0	0	0
0	1	0	1	0	1
0	1	1	0	1	0
0	1	1	1	0	1
1	0	0	0	0	0
1	0	0	1	0	1
1	0	1	0	1	0
1	0	1	1	1	0
1	1	0	0	0	0
1	1	0	1	0	0
1	1	1	0	0	0
1	1	1	1	0	0



Máquinas de Estados

- **Máquinas de Estados, Modelo de Moore:**
  - CLS en el que los valores de salida únicamente dependen del estado, no de la señal de entrada.



Modelo de Mealy:

$W = H(X,Q)$
Ec. de salida

$Q^+ = G(X,Q)$
Ec. estado siguiente

Modelo de Moore:

$W = H(Q)$
Ec. de salida

$Q^+ = G(X,Q)$
Ec. estado siguiente

Máquinas de Estados

- **Especificación de un CLS tipo <u>MOORE</u>:**
  - X = (x_0), W = (w_1,w_0), Q = (q_1,q_0), Estado inicial (0,0).

- Tabla de verdad de Q^+

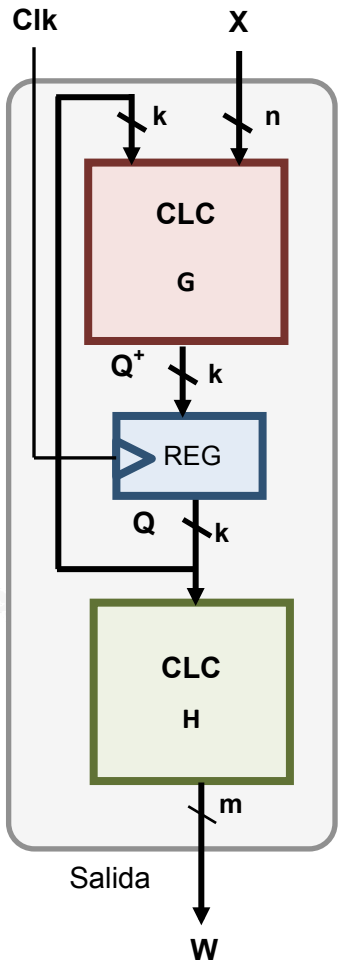
q_1	q_0	x_0	q_1^+	q_0^+
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	0	1
1	0	0	0	1
1	0	1	1	1
1	1	0	0	0
1	1	1	0	1

Tabla de Transición (estado siguiente)

- Tabla de verdad de W

q_1	q_0	w_1	w_0
0	0	0	0
0	1	0	1
1	0	0	0
1	1	1	0

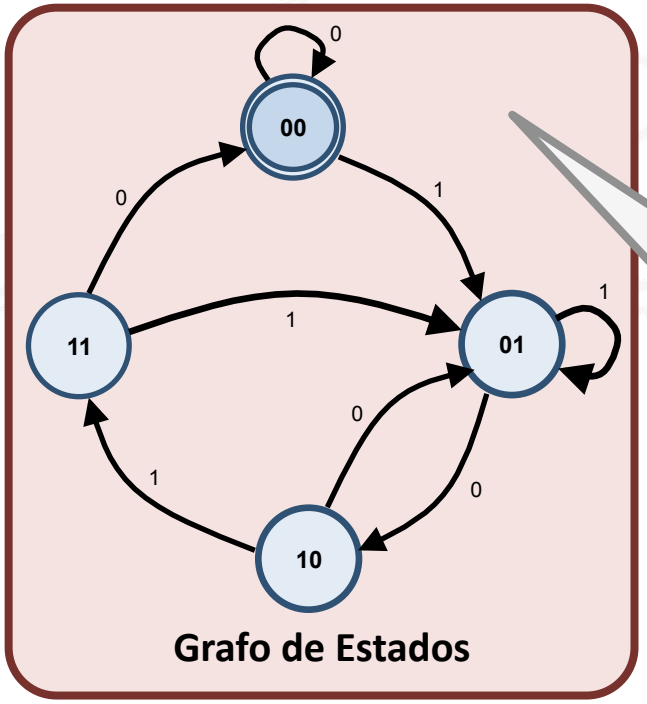
Tabla de Salida



Máquinas de Estados

- **Especificación de un CLS tipo <u>MOORE</u>:**
  - $X = (x_0)$, $W = (w_1,w_0)$, $Q = (q_1,q_0)$, Estado inicial (0,0).

- Tabla de verdad de Q^+

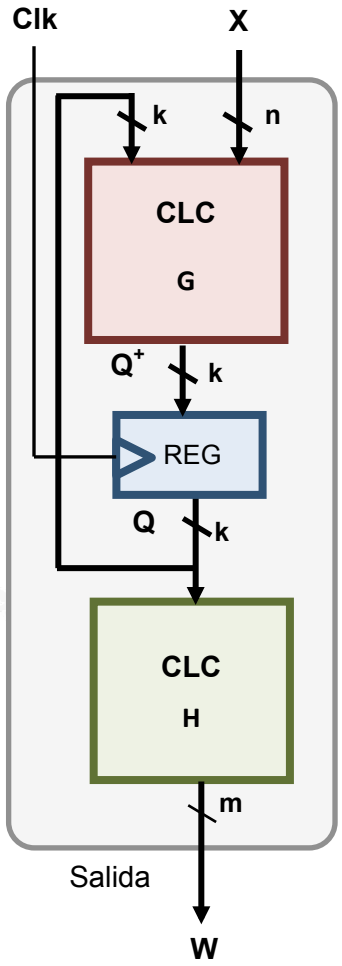


- Tabla de verdad de W

q_1	q_0	w_1	w_0
0	0	0	0
0	1	0	1
1	0	0	0
1	1	1	0

Tabla de Salida

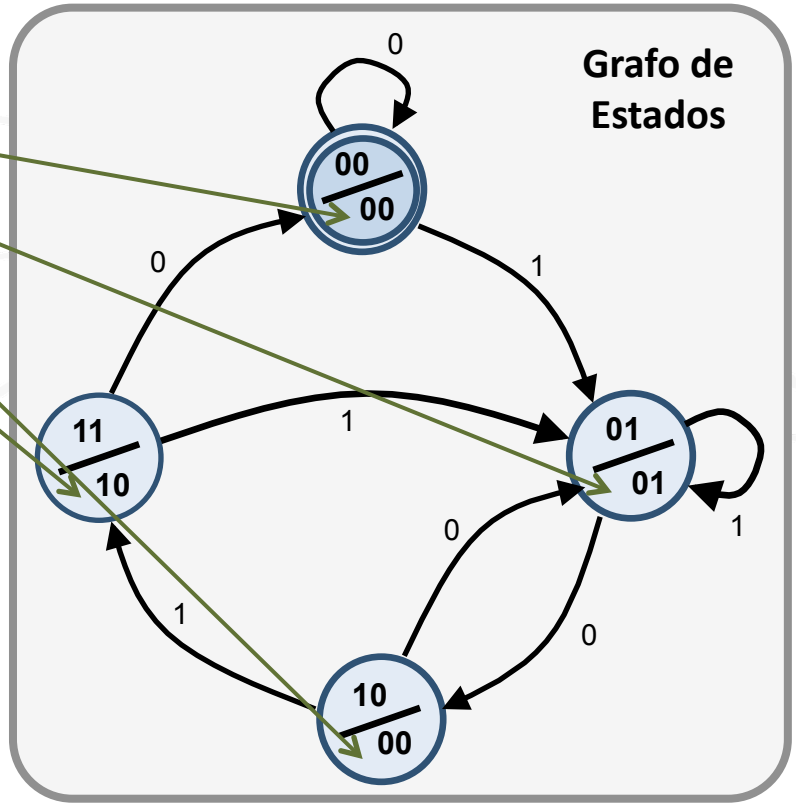
Igual que en Mealy, la Tabla de Transiciones se puede sustituir por el Grafo de Estados.



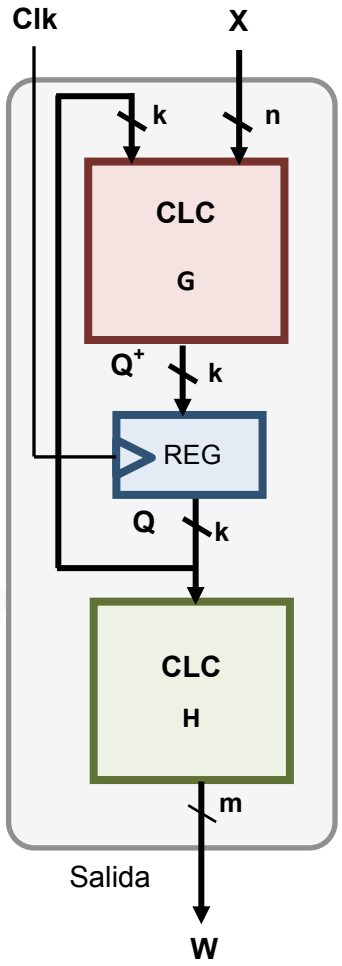
Máquinas de Estados

- **Especificación de un CLS tipo MOORE:**
  - X = (x_0), W = (w_1,w_0), Q = (q_1,q_0), Estado inicial (0,0).

q_1	q_0	w_1	w_0
0	0	0	0
0	1	0	1
1	0	0	0
1	1	1	0



Como los valores de salida solo dependen del estado, se pueden incluir en el Grafo de Estados, que sustituye ambas T.V. en la especificación.



Índice

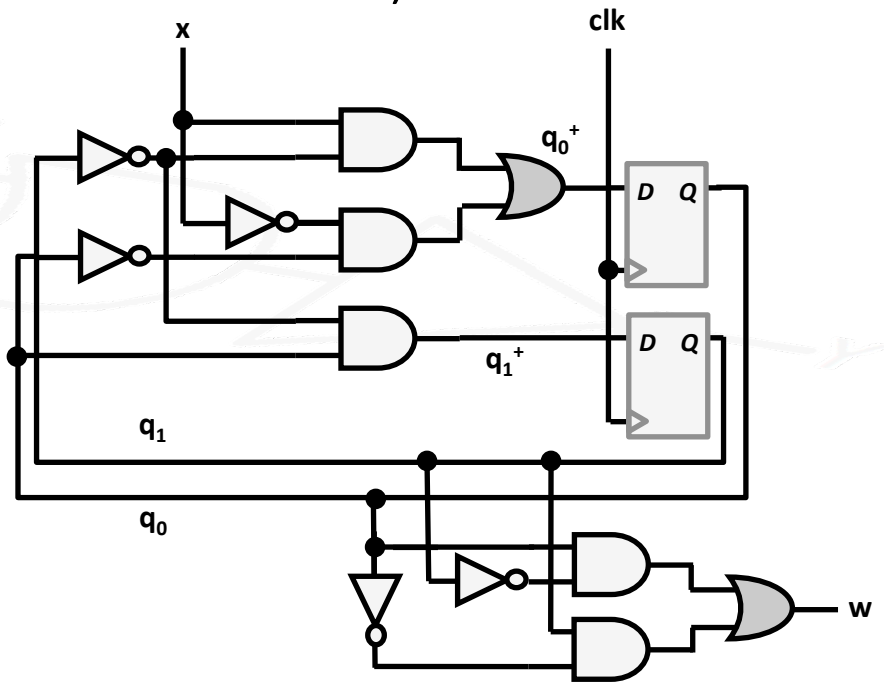
- **Introducción:**
  - Definición de CLS.
  - Necesidad de Memoria y Sincronización.
- **Latches y Flip Flops.**
- **Máquinas de Estados:**
  - Mealy.
  - Moore.
- **Análisis.**
- **Síntesis:**
  - Mínimo número de biestables.
  - 1 biestable por estado.
- **Timing.**

Análisis

- **Análisis Lógico: Dado el esquema lógico de un CLS obtener el grafo de estados:**
  - **PASO 1:** obtener las Tablas de Verdad (transición y salida).
  - **PASO 2:** obtener el Grafo (a partir de las tablas).

q_1	q_0	x_0	q_1^+	q_0^+
0	0	0	0	1
0	0	1	0	1
0	1	0	1	0
0	1	1	1	1
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	0	0

Tabla de Transición

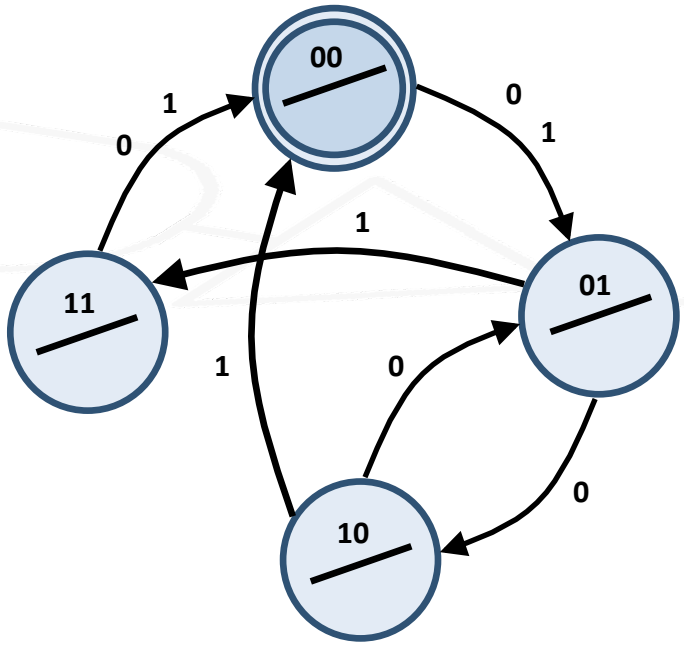


Análisis

- **Análisis Lógico: Dado el esquema lógico de un CLS obtener el grafo de estados:**
  - **PASO 1:** obtener las Tablas de Verdad (transición y salida).
  - **PASO 2:** obtener el Grafo (a partir de las tablas).

q_1	q_0	x_0	q_1^+	q_0^+
0	0	0	0	1
0	0	1	0	1
0	1	0	1	0
0	1	1	1	1
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	0	0

Tabla de Transición

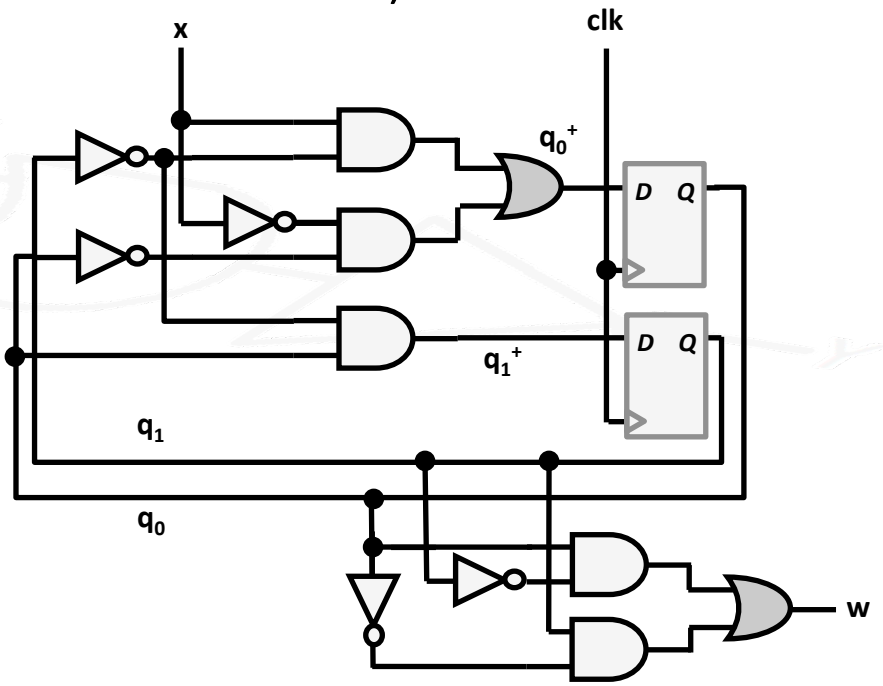


Análisis

- **Análisis Lógico: Dado el esquema lógico de un CLS obtener el grafo de estados:**
  - **PASO 1:** obtener las Tablas de Verdad (transición y salida).
  - **PASO 2:** obtener el Grafo (a partir de las tablas).

q_1	q_0	w
0	0	0
0	1	1
1	0	1
1	1	0

Tabla de Salida

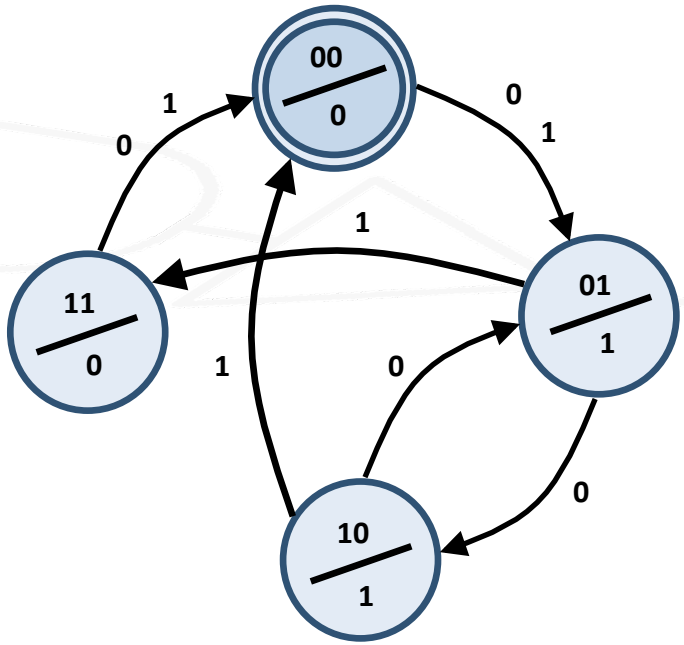


Análisis

- **Análisis Lógico: Dado el esquema lógico de un CLS obtener el grafo de estados:**
  - **PASO 1:** obtener las Tablas de Verdad (transición y salida).
  - **PASO 2:** obtener el Grafo (a partir de las tablas).

q_1	q_0	w
0	0	0
0	1	1
1	0	1
1	1	0

Tabla de Salida

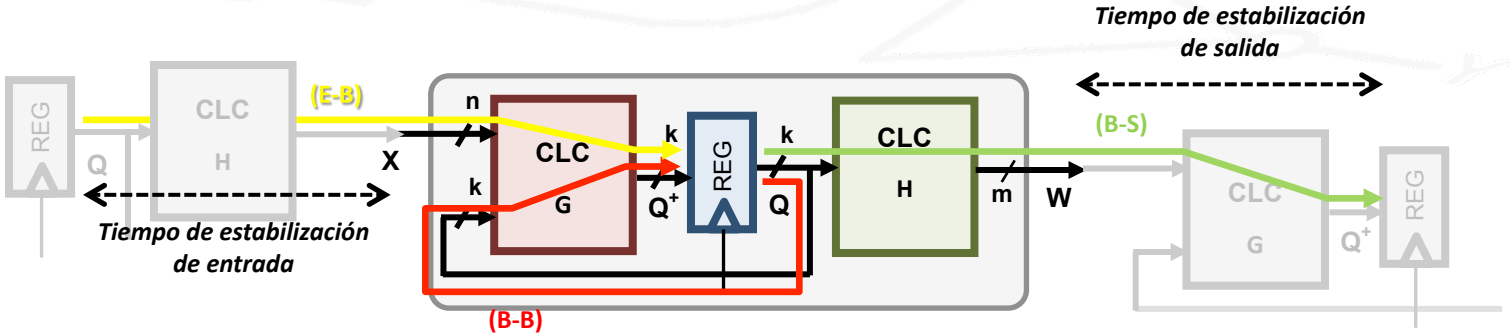


Índice

- **Introducción:**
  - Definición de CLS.
  - Necesidad de Memoria y Sincronización.
- **Latches y Flip Flops.**
- **Máquinas de Estados:**
  - Mealy.
  - Moore.
- **Análisis.**
- **Síntesis:**
  - Mínimo número de biestables.
  - 1 biestable por estado.
- **Timing.**

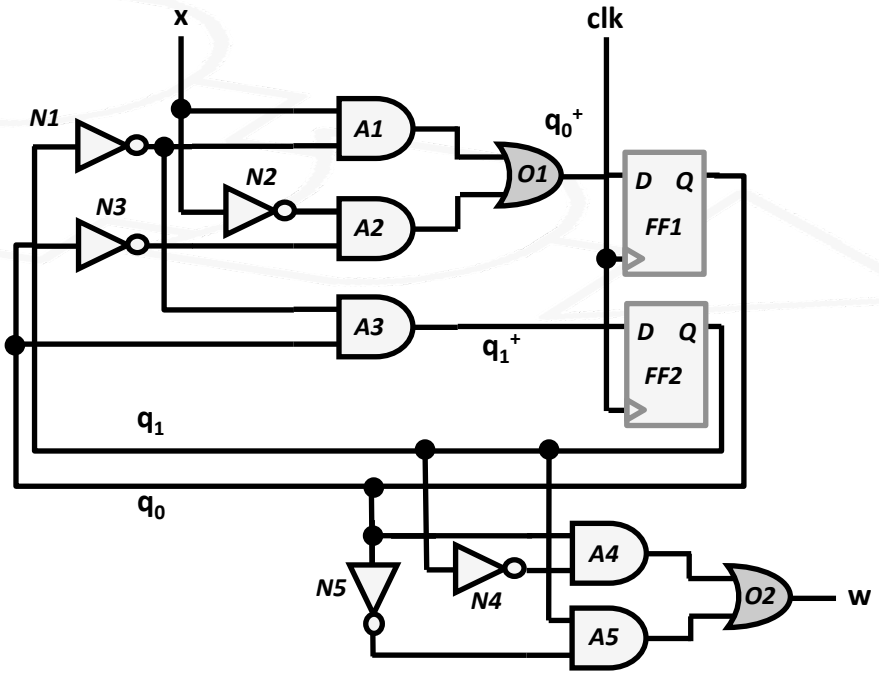
Timing

- **Análisis Temporal:** Dado el esquema lógico de un CLS, los tiempos de propagación de los dispositivos y los tiempos de estabilización de las entradas y las salidas, obtener el tiempo de ciclo mínimo.
- **Camino crítico:** En un CLS Síncrono, los caminos a evaluar son diferentes, y corresponden a los siguientes tipos:
  - Camino de Biestable a Biestable (B-B).
  - Camino de Entrada a Biestable (E-B).
  - Camino de Biestable a Salida (B-S).



Timing

- **Ejemplo: Obtener el T.C. mínimo del circuito. Tiempos:**
  - **Puertas:** OR: 30 u.t. AND: 30 u.t. NOT: 20 u.t.
  - **Biestables:** FF1: 100 u.t. FF2: 150 u.t.
  - **Estabilización:** Entrada: 110 u.t. Salida: 20 u.t.

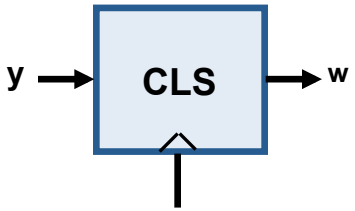


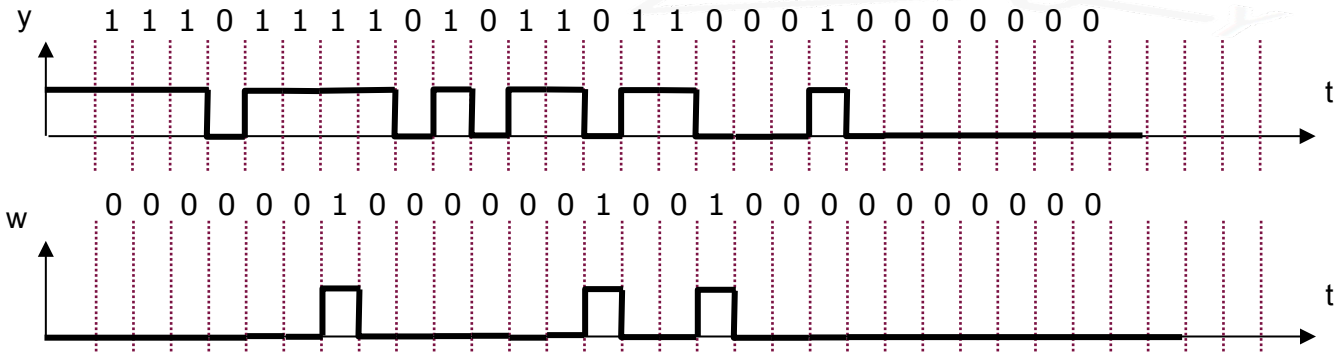
Índice

- **Introducción:**
  - Definición de CLS.
  - Necesidad de Memoria y Sincronización.
- **Latches y Flip Flops.**
- **Máquinas de Estados:**
  - Mealy.
  - Moore.
- **Análisis.**
- **Síntesis:**
  - Mínimo número de biestables.
  - 1 biestable por estado.
- **Timing.**

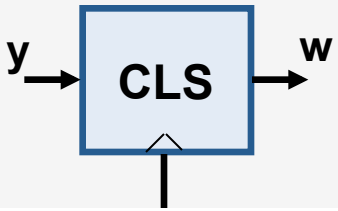
Síntesis

- **A partir de una descripción textual, obtener el Circuito Secuencial:**
  - El primer paso es obtener el Grafo de Estados.
  - Una vez completado el grafo, procedemos a la implementación física.
- **Ejemplo: Circuito reconocedor de secuencias:**
  - Obtener la implementación de un circuito de una entrada y una salida que reconoce la secuencia 011 de valores en la señal de entrada. La señal de salida toma valor 1 cada vez que la secuencia a detectar aparece.

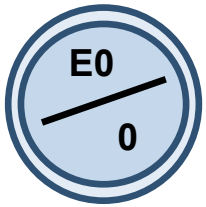




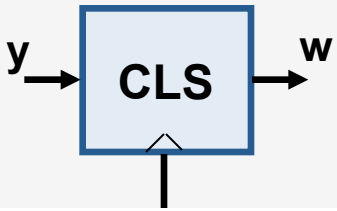
Síntesis



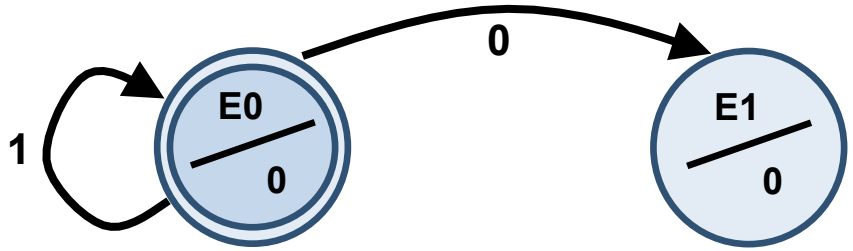
- **Grafo de Estados: Circuito reconocedor de secuencias (011).**
- ***Paso 1:*** obtener el estado inicial:
  - E0 (estado inicial): no ha llegado ningún bit de la secuencia a reconocer.
  - Salida (para este estado) w: 0.



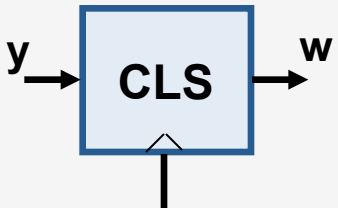
Síntesis



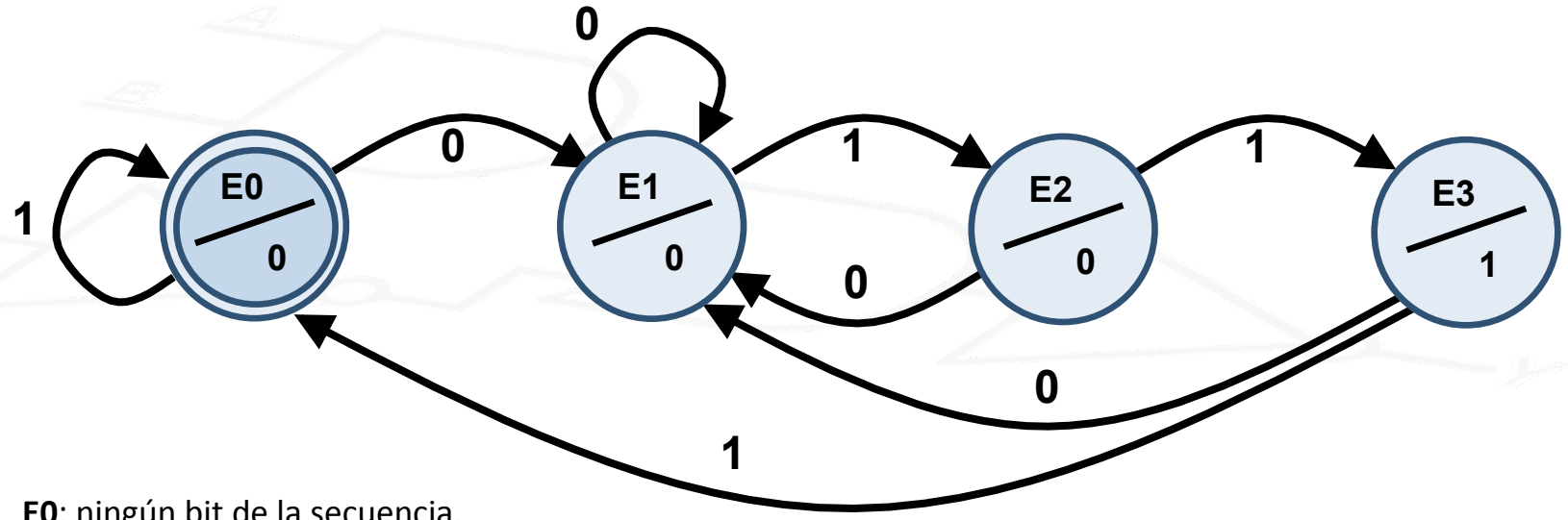
- **Grafo de Estados: Circuito reconocedor de secuencias (011).**
- ***Paso 1:*** obtener el estado inicial:
  - E0 (estado inicial): no ha llegado ningún bit de la secuencia a reconocer.
  - Salida (para este estado) w: 0.
- ***Paso 2:*** comienzo a construir el grafo:
  - Si me encuentro en el estado inicial, ¿qué ocurre para cada posible valor de entrada del CLS?:
    - Llega un 1: sigue sin llegar un bit de la secuencia (me quedo en E0).
    - Llega un 0: llega el 1er bit de la secuencia (nuevo estado E1).



Síntesis



- **Grafo de Estados: Circuito reconocedor de secuencias (011).**
- ***Paso 3:*** repetimos el procedimiento del paso anterior:
  - Lo hacemos así hasta tener completo el grafo.



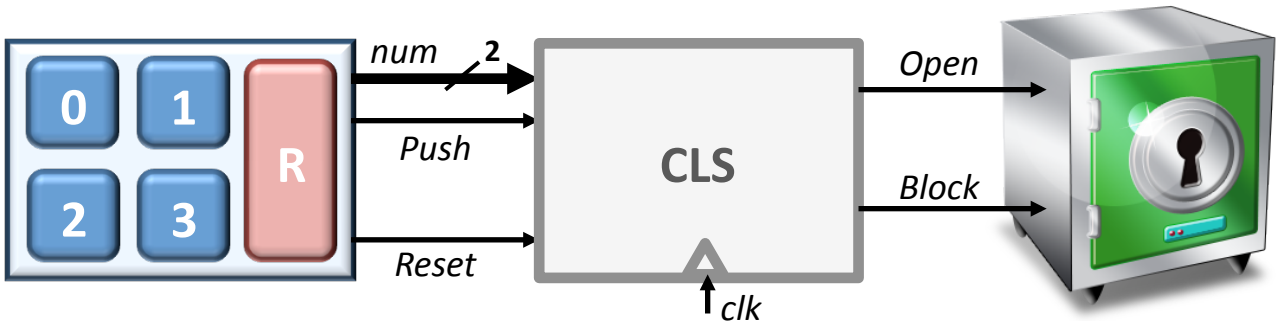
E0: ningún bit de la secuencia.
E1: ha llegado 1 bit de la secuencia (0).
E2: han llegado 2 bits de la secuencia (01).
E3: han llegado los 3 bits de la secuencia (011) -> Salida = 1.

Síntesis

- **Ejercicio 1:**
  - Obtener el grafo de estados de un contador up/down módulo 7. El contador tiene dos entradas: count Enable (E) y count Direction (D). Cuando E = 1 el contador cuenta en la dirección especificada por D, y termina cuando E = 0. El contador cuenta hacia arriba si D = 0 y hacia abajo si D = 1. El contador tiene una salida Y que toma valor 1 cuando el contador alcanza el valor 7 o el valor 0.

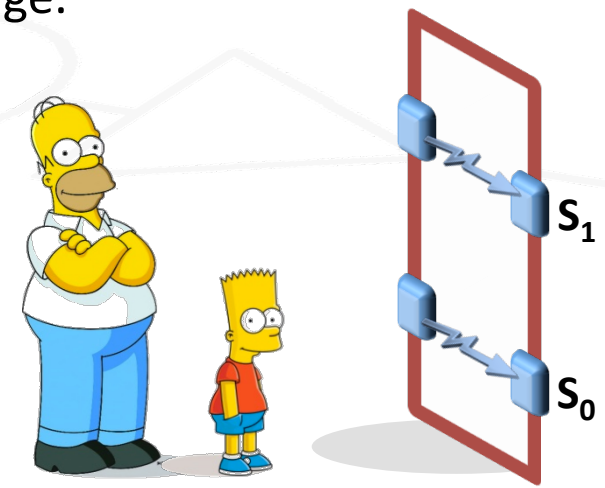
Síntesis

- **Ejercicio 2:**
  - Obtener el grafo de estados de una puerta con cerradura de apertura por secuencia de dígitos. El display tiene los botones mostrados en la figura inferior. Cada vez que pulsamos una tecla la señal *Push* se activa, y las señales num_i indican la tecla que se pulsó (valor en binario). La señal de *Reset* sirve para reiniciar la introducción del código de apertura.
  - El circuito tiene dos señales de Salida: *Open* y *Block*. Para que la señal *Open* se active y la puerta se abra, debemos introducir: 0-3-2. Si introducimos una secuencia de tres dígitos incorrecta, la puerta se bloquea (activamos señal *Block).*



Síntesis

- **Ejercicio 3:**
  - Marge quiere instalar un sistema de alarma que la indique las personas que se encuentran en la cocina en todo momento. Para ello, se instalan dos sensores en el marco de la puerta tal y como muestra la figura. Estos sensores toman valor 1 si su flujo es interrumpido, siendo 0 el resto del tiempo. Sabiendo que Bart y Homer no caben por la puerta si intentan entrar a la vez, Determina el grafo de estados del circuito que avisará a Marge.

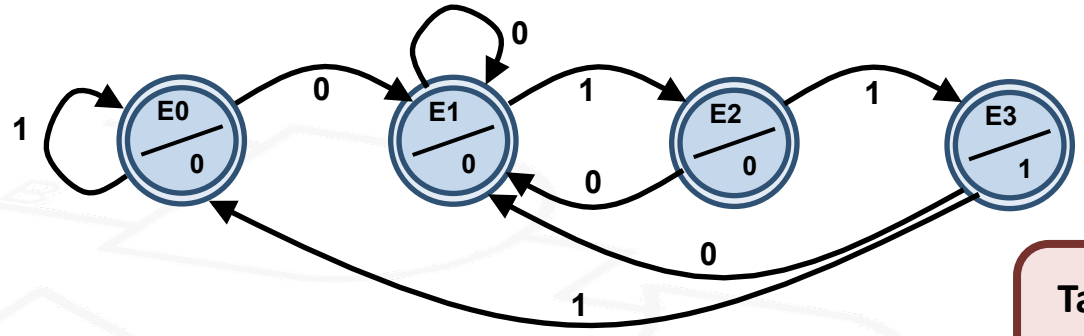


Síntesis

- **Del Grafo de Estados a la Implementación del Circuito:**
  - El Grafo de Estados nos proporciona las tablas de verdad.
  - Codificaremos los estados (binario) con el número mínimo de biestables:
    - Suponiendo n estados: número de biestables= int($\log_2 n$).
  - Conocidas las tablas, la síntesis utiliza las mismas técnicas vistas para CLCs.
- **Implementación de los CLCs de estado siguiente y salida:**
  - Con Not, And y Or en suma de minterms.
  - Con decodificadores y puertas Or.
  - Con ROMs:
    - Dos ROMs.
    - Una ROM.
    - Una ROM y un Multiplexor de buses.

Síntesis

- **Determinación de las Tablas de Verdad:**
  - Grafo de Estados: Circuito reconocedor de secuencias (011).



Codif. de Estado

Estado	Codif.
E0	00
E1	01
E2	10
E3	11

Tabla de Salida

q_1	q_0	w_1
0	0	0
0	1	0
1	0	0
1	1	1

Tabla de Transición

q_1	q_0	x_0	q_1^+	q_0^+
0	0	0	0	1
0	0	1	0	0
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	1
1	1	0	0	1
1	1	1	0	0

Síntesis

- **Implementación con Dos ROMs:**
  - Grafo de Estados: Circuito reconocedor de secuencias (011).

Codif. Estado

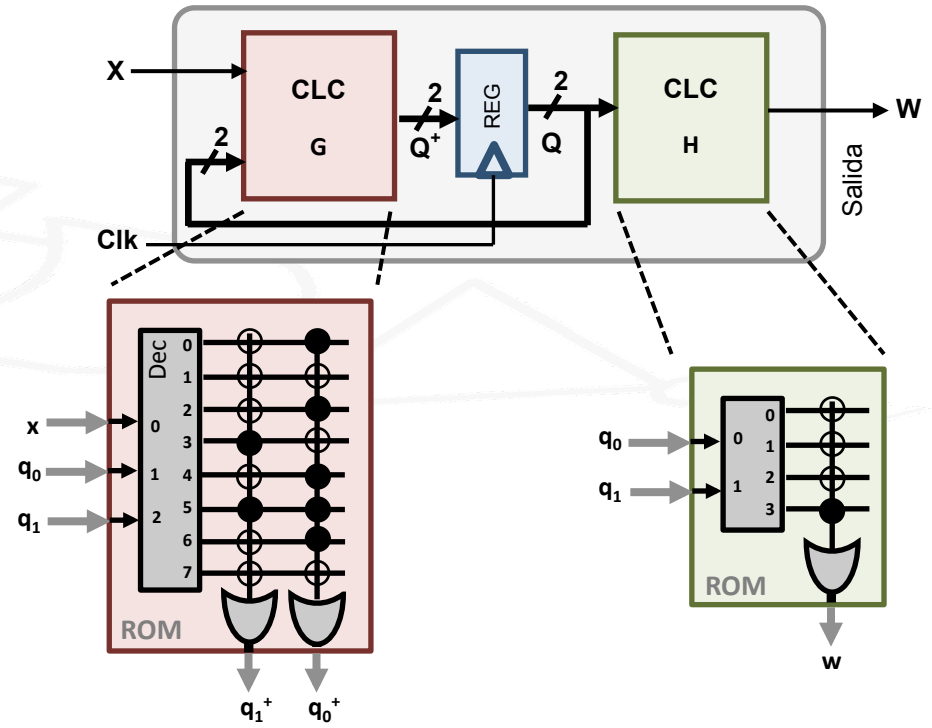
Est.	Cod.
E0	00
E1	01
E2	10
E3	11

Tabla de Salida

q_1	q_0	w
0	0	0
0	1	0
1	0	0
1	1	1

Tabla de Transición

q_1	q_0	x	q_1^+	q_0^+
0	0	0	0	1
0	0	1	0	0
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	1
1	1	0	0	1
1	1	1	0	0



Síntesis

- **Implementación con Una ROM (Mealy):**
  - Grafo de Estados: Circuito reconocedor de secuencias (011).

Codif. Estado

Est.	Cod.
E0	00
E1	01
E2	10
E3	11

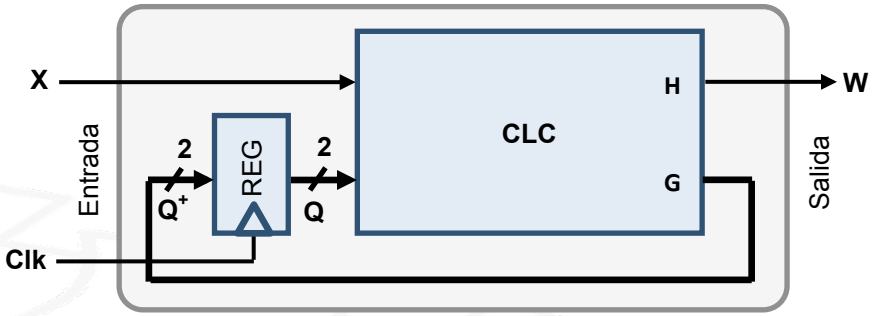
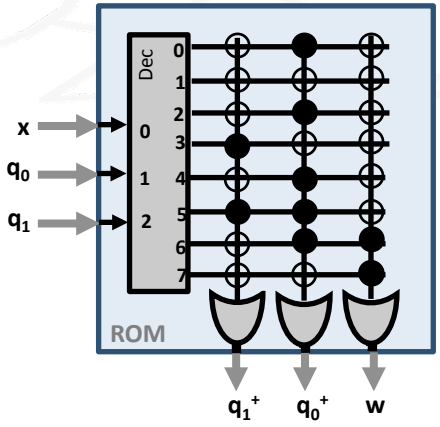


Tabla de Transición/Salida

q_1	q_0	x	q_1^+	q_0^+	w
0	0	0	0	1	0
0	0	1	0	0	0
0	1	0	0	1	0
0	1	1	1	0	0
1	0	0	0	1	0
1	0	1	1	1	0
1	1	0	0	1	1
1	1	1	0	0	1



Síntesis

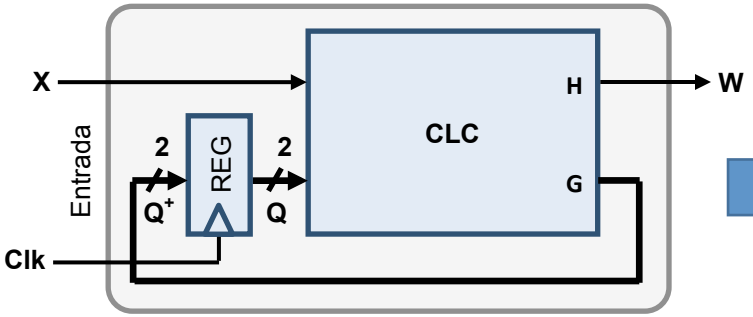
- **Implementación con Una ROM y Multiplexor.**

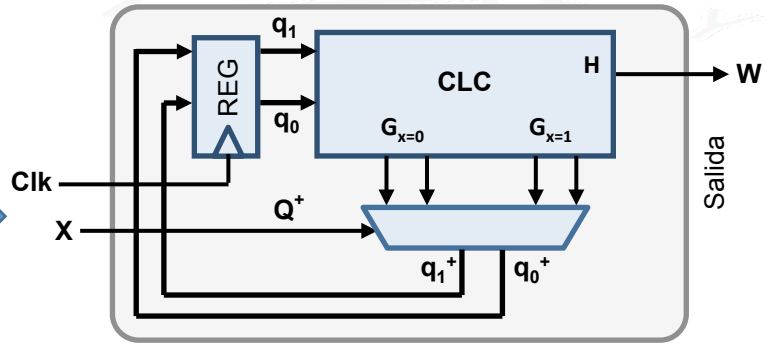
Tabla de Transición/Salida

q_1	q_0	x	q_1^+	q_0^+	w
0	0	0	0	1	0
0	0	1	0	0	0
0	1	0	0	1	0
0	1	1	1	0	0
1	0	0	0	1	0
1	0	1	1	1	0
1	1	0	0	1	1
1	1	1	0	0	1

Tabla de Transición/ Salida

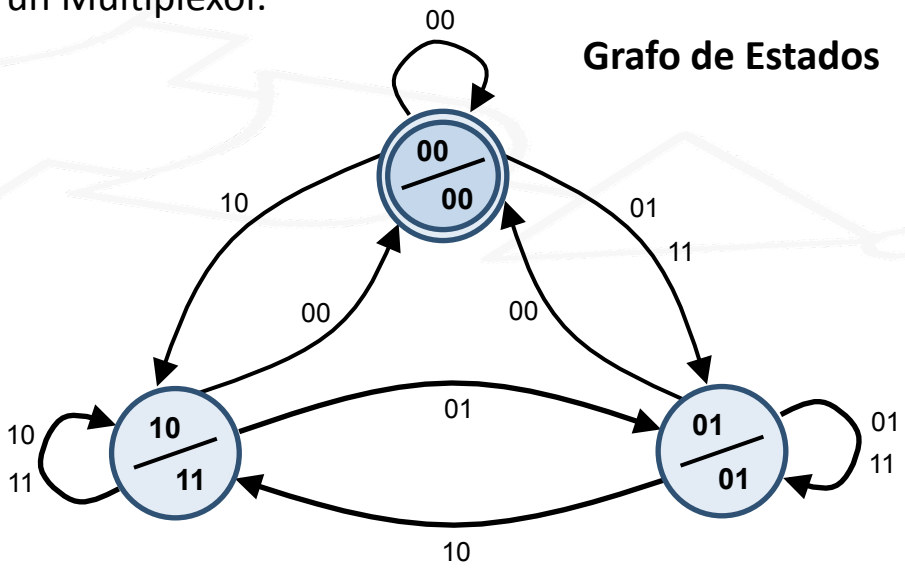
		X=1		X=0		
q_1	q_0	q_1^+	q_0^+	q_1^+	q_0^+	w
0	0	0	0	0	1	0
0	1	1	0	0	1	0
1	0	1	1	0	1	0
1	1	0	0	0	1	1





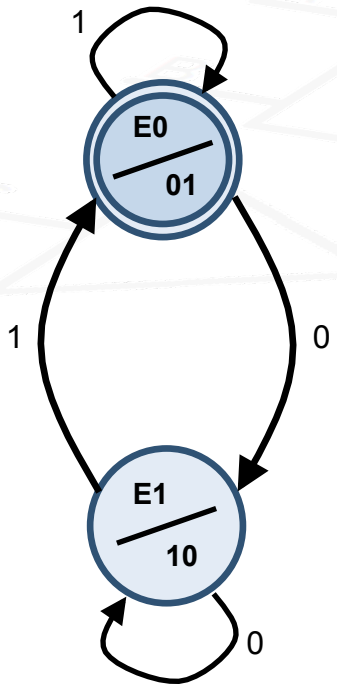
Síntesis

- **Ejercicio:**
  - Dado el grafo de estados inferior, implementar su circuito correspondiente utilizando el número mínimo de biestables y:
    - Dos ROM del tamaño que consideres necesario.
    - Una Sola ROM.
    - Una Sola ROM y un Multiplexor.

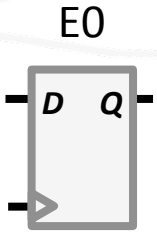


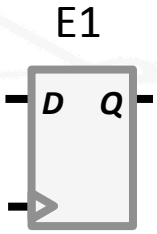
Síntesis

- **Implementación con un Biestable por estado, Demultiplexores y puertas OR:**
  - A partir del Grafo de Estados del circuito.



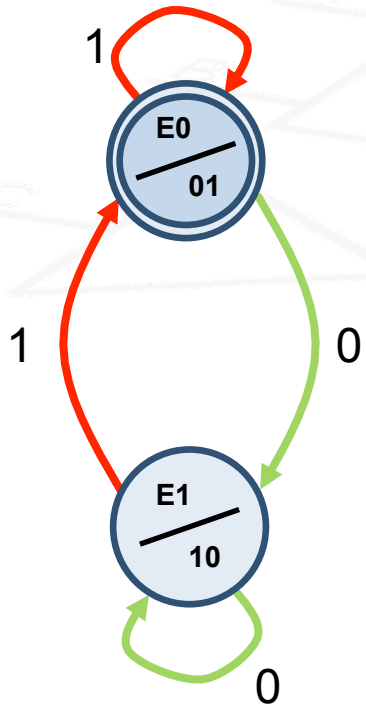
PASO 1: asignamos un biestable a cada estado.



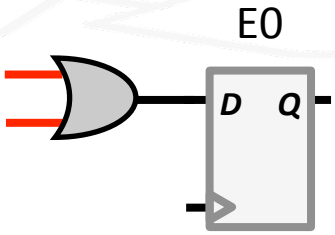


Síntesis

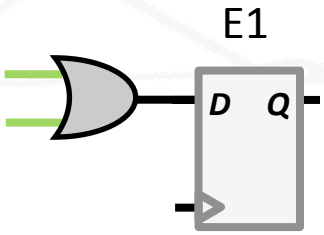
- **Implementación con un Biestable por estado, Demultiplexores y puertas OR:**
  - A partir del Grafo de Estados del circuito.



PASO 2: se conecta la salida de una puerta OR a la entrada D de cada biestable. El número de entradas de la OR será igual al número de arcos que llegan al Nodo (biestable).



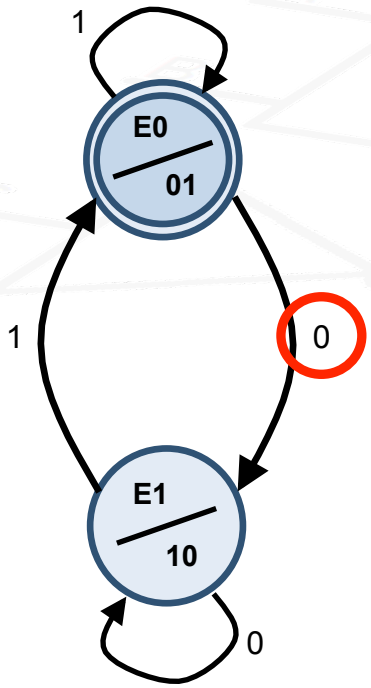
2 arcos -> 2 entradas



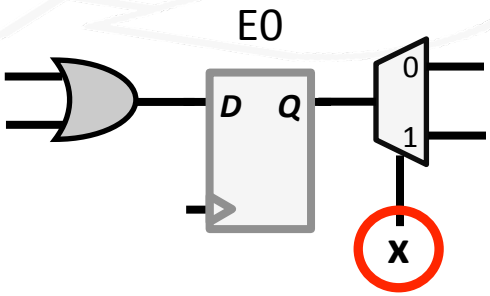
2 arcos -> 2 entradas

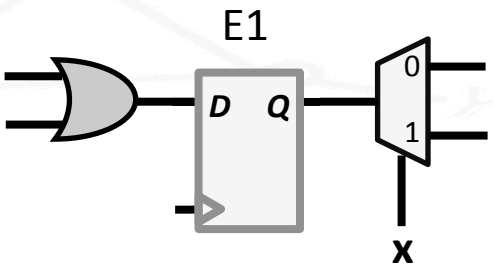
Síntesis

- **Implementación con un Biestable por estado, Demultiplexores y puertas OR:**
  - A partir del Grafo de Estados del circuito.



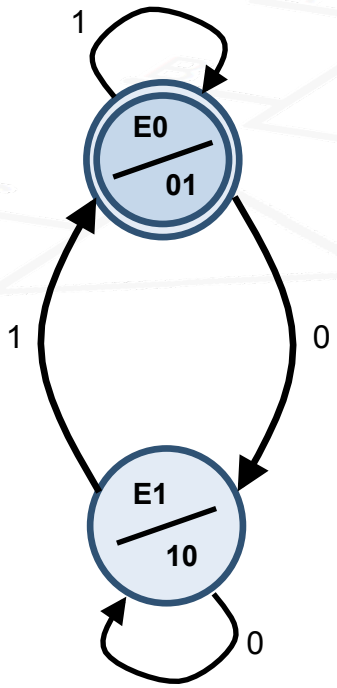
PASO 3: en la salida de cada Biestable se conecta un demultiplexor. Tendrá tantas señales de selección como señales de entrada tiene el circuito. Se conectan las entradas del circuito a las señales de selección de cada multiplexor.



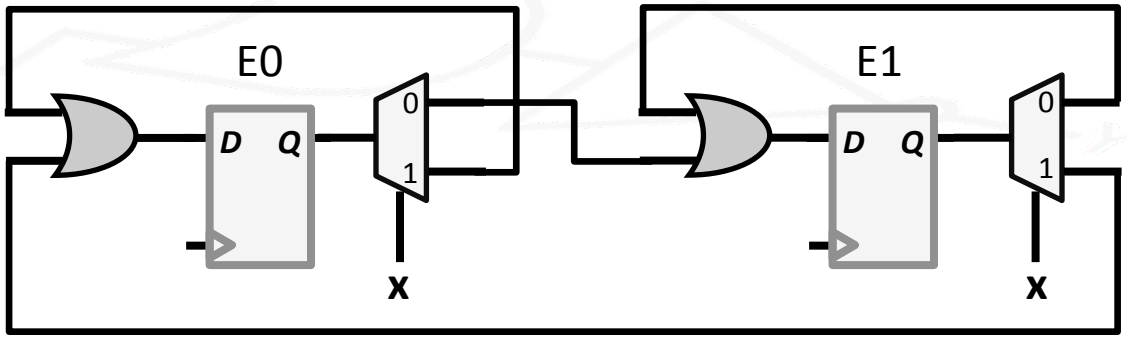


Síntesis

- **Implementación con un Biestable por estado, Demultiplexores y puertas OR:**
  - A partir del Grafo de Estados del circuito.

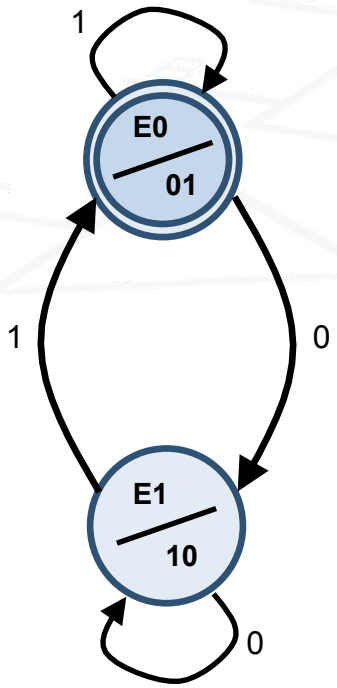


PASO 4: se conectan las salidas de los demultiplexores con las entradas de las OR, de acuerdo con el grafo.

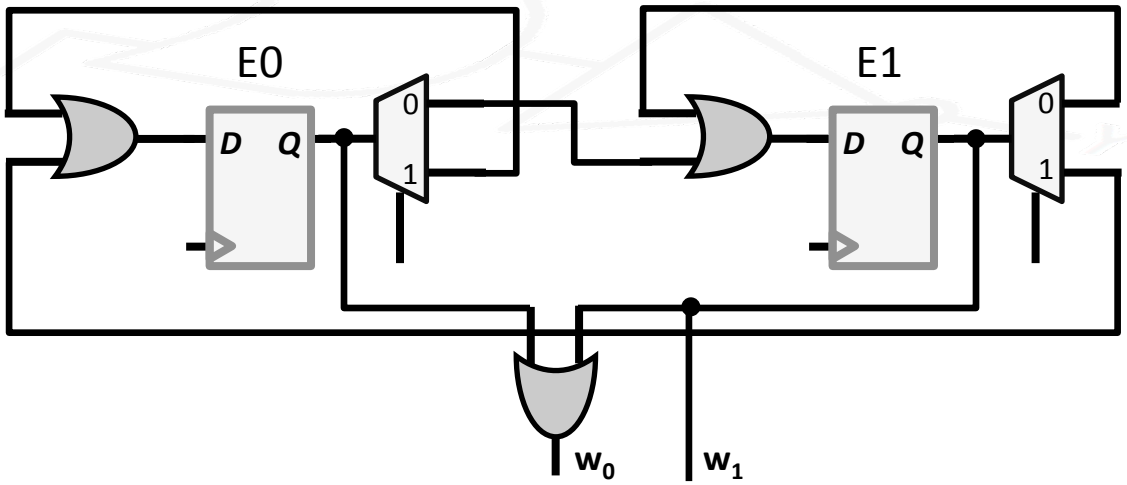


Síntesis

- **Implementación con un Biestable por estado, Demultiplexores y puertas OR:**
  - A partir del Grafo de Estados del circuito.



PASO 5: cada señal de salida se forma con la OR de las salidas de los biestables de los estados en que vale 1 esa salida.



Síntesis

- **Ejercicio:**
  - Dado el grafo de estados inferior, implementar su circuito correspondiente utilizando un biestable por estado.