DSD FCHE 2011
Circuitos secuenciales
Diagramas de Estado

Diseño

ANALISIS DE LOS CIRCUITOS SECUENCIALES TEMPORIZADOS

El comportamiento de los circuitos secuenciales se determina de las entradas, las salidas y los estados de los flip-flops. Ambas entradas y el siguiente estado son una función de las entradas y el presente estado. El análisis de los circuitos secuenciales consiste en obtener una tabla o un diagrama de la secuencia de tiempo de las entradas, salidas y estados internos.

Diagramas de Estado

Autómata Finito Determinístico

Definiciones:

a) Autómata: Modelo matemático de una entidad que transita de un estado a otro en función de una entrada y del estado actual. (Autómata finito determinístico)
b) Matemáticamente: Se requiere de una función que se exprese usualmente como una tabla de transiciones o de diagrama de estado.
c) Encabezado: f (Estado actual, entrada) = > (Estado siguiente, salida)

Un autómata en general tiene 6 coordenadas

$$\mathbf{AFD = (Q, \Sigma, \Gamma, \delta, e\,I, F)}$$

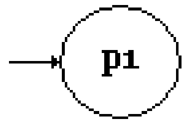
Q - Estado
Σ - Alfabeto de entrada
Γ - Alfabeto salida
δ - Función de transición
e I - Estado inicial
F - Estado de aceptación

Reglas

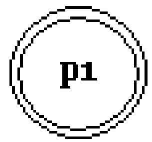
Diagramas de transiciones (diagramas de estado)

- Se dibuja un nodo de la gráfica por cada estado y se etiqueta con el nombre de ese estado.
- Se resalta el inicial con una flecha Pi

  Ejemplo:



- El estado final o de aceptación se enmarca con doble círculo.



- Transición entre estados.

Tabla de transiciones

Estado	Alfabeto de entrada	Estado siguiente	Alfabeto de salida
Qk	Σ	Qk +1	Γ
q1q0	b0b1	q1q0	γ0γ1

Ejemplo: Checador de Paridad

Diseñe un autómata que funcione, como checador de paridad, es decir que acepte todas aquellas sucesiones con un número par de unos.

$$AFD = (Q, \Sigma, \Gamma, \delta, \{e\}, F) \quad \Rightarrow \textbf{Gramática}$$

Entrada $\Sigma = \{0,1\}$

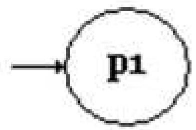
Salida $\Gamma = \{0,1\}$ => 0 es no paridad y 1 es paridad

Q => Se da durante el diseño la transición se pone en notación de conjuntos.

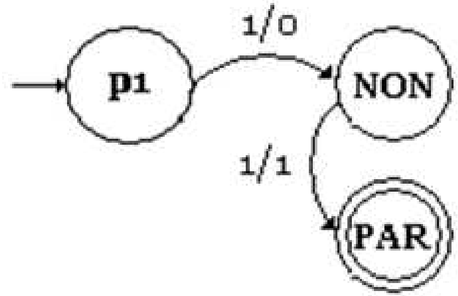
$$\boldsymbol{\delta: Q \times \Sigma \Rightarrow Q \times \Gamma}$$

Diseño

1) Para la cadena ideal = 11, comienzo:

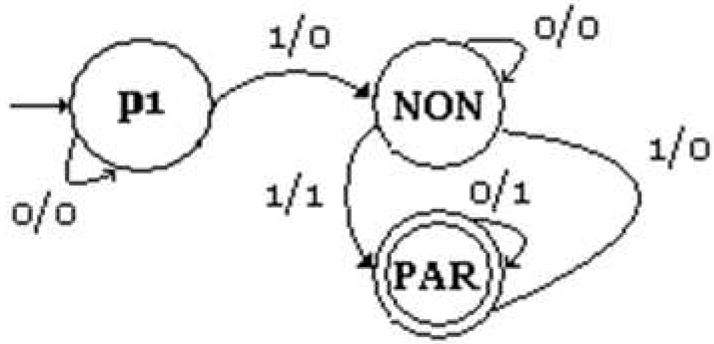


Si recibo un"uno" me voy al NON y si recibo otro voy a PAR:



Pasos…

2) Ahora hay que complementarlo (se hizo para "uno", se hace para "cero"= > por alfabeto).



Nota: Como no estoy contando ceros no me muevo, solo cuento unos.

3) Realizar tabla de transición para pasarlo al circuito.

n = **numero de bits que representan el estado**

Numero máximo de estados = 2^n

Tabla de transición

Se necesita un código para llenar la tabla:

Estados
e1 = 00
non = 01
par = 10

Símbolos			
Qk	Σ	Qk +1	Γ
e1	0	e1	0
e1	1	non	0
non	0	non	0
non	1	par	1
par	0	par	1
par	1	non	0

Codificación Binaria			
Qk q1q0	Σ b0	Qk +1 q1q0	Γ γ0
00	0	00	0
00	1	01	0
01	0	01	0
01	1	10	1
10	0	10	1
10	1	01	0

11 No hay estados para esta
11 asignacion

Ejemplo 2

- Dado código 4 4 1 -2, se requiere un circuito secuencial síncrono que detecte este código. Esta información se recibe y transfiere en serie. (El primer bit en transmitirse es el bit menos significativo)

dec	4 4 1 -2
0	0 0 0 0
2	0 0 1 0
5	0 1 0 1
7	0 1 1 1
4	0 1 0 0
10	1 0 1 0
13	1 1 0 1
15	1 1 1 1
12	1 1 0 0
14	1 1 1 0

0,2,4,5,7,10,12,13,14,15

no aparecen

1,3,6,8,9,11

dec	bin
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111

Resolviendo ejemplo2

- ¿Que números binarios no aparecen en este código? Estos serán la respuesta de un código que no es.

dec	4	4	1	-2
0	0	0	0	0
2	0	0	1	0
5	0	1	0	1
7	0	1	1	1
4	0	1	0	0
10	1	0	1	0
13	1	1	0	1
15	1	1	1	1
12	1	1	0	0
14	1	1	1	0

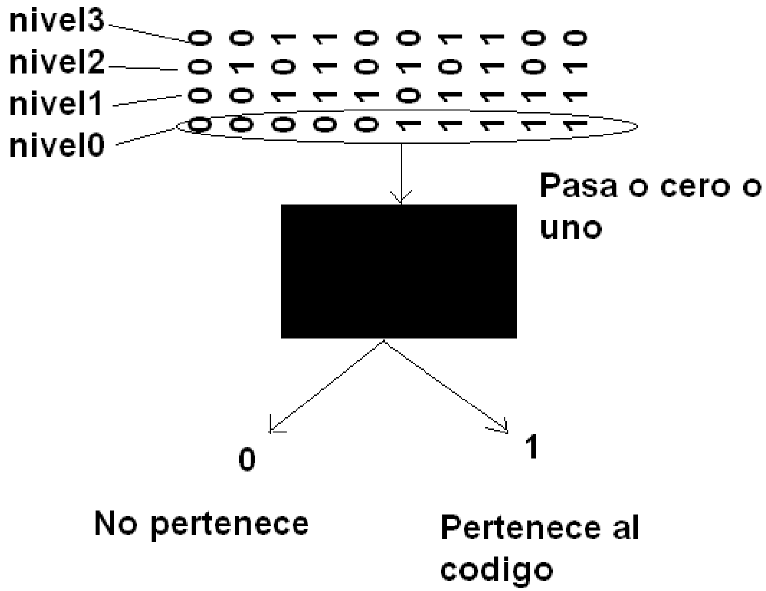
0,2,4,5,7,10,12,13,14,15

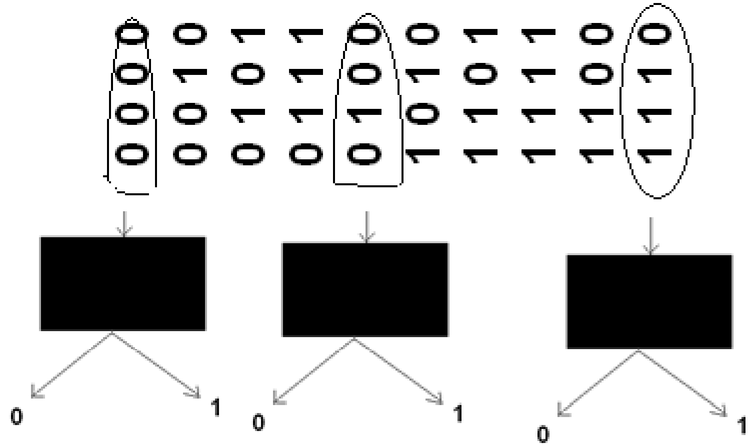
no aparecen

1,3,6,8,9,11

dec	bin
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111

Nos imaginamos… Lluvia de ideas





Resolviendo ejemplo2

Después de comprender el problema, determinamos entradas, salidas y estados.

Tabla de transiciones

Estado	Alfabeto de entrada	Estado siguiente	Alfabeto de salida
Qk	Σ	Qk +1	Γ
q1q0	b0b1	q1q0	γ0γ1

dec	4	4	1	-2
0	0	0	0	0
2	0	0	1	0
5	0	1	0	1
7	0	1	1	1
4	0	1	0	0
10	1	0	1	0
13	1	1	0	1
15	1	1	1	1
12	1	1	0	0
14	1	1	1	0

0,2,4,5,7,10,12,13,14,15

no aparecen

1,3,6,8,9,11

dec	bin
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111

Respuestas:

Estados: el dato exacto se da durante el diseño.

Entradas: transmiten en serie, pasaran los datos uno a uno, por lo tanto es b0 {0,1}

Salidas: Para cada estado determinamos nosotros que :

1 si el bit transferido NO corresponde al código.

0 si el bit transferido corresponde al código.

Por lo tanto la salida es un bit, y0 {0,1}

- Entra el menos significativo, puede ser 0 o 1

Resolviendo ejemplo2. Diagrama de Estado

dec	4 4 1 -2
0	0 0 0 0
2	0 0 1 0
5	0 1 0 1
7	0 1 1 1
4	0 1 0 0
10	1 0 1 0
13	1 1 0 1
15	1 1 1 1
12	1 1 0 0
14	1 1 1 0

0,2,4,5,7,10,12,13,14,15

no aparecen

1,3,6,8,9,11

dec	bin
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111

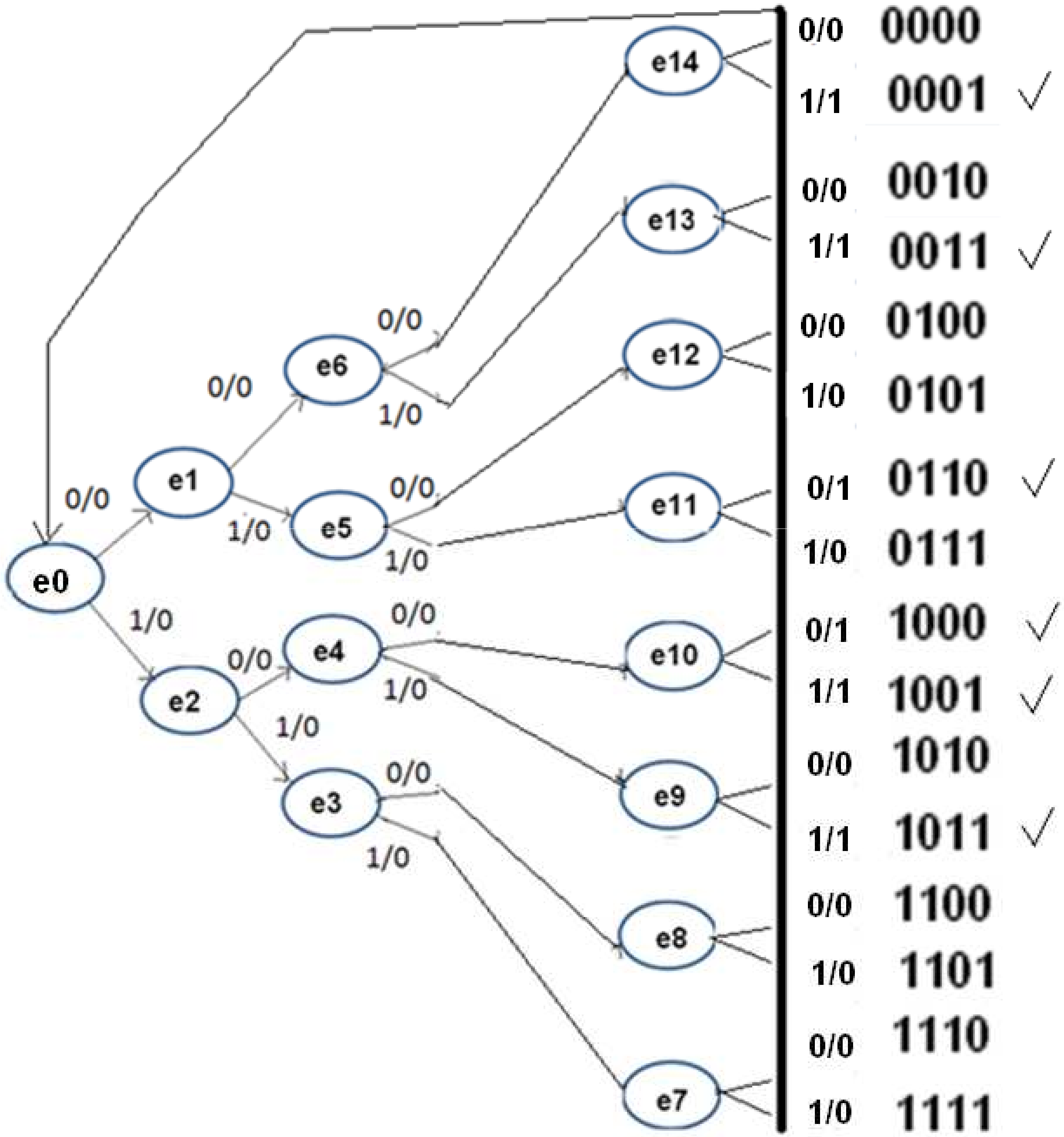


Tabla de Transicion

Q	ent	Q+1	sal
e0	0	e1	0
e0	1	e2	0
e1	0	e6	0
e1	1	e5	0
e2	0	e4	0
e2	1	e3	0
e3	0	e8	0
e3	1	e7	0
e4	0	e10	0
e4	1	e9	0
e5	0	e12	0
e5	1	e11	0
e6	0	e14	0
e6	1	e13	0

Q	ent	Q+1	sal	antecente
e7	0	e0	0	1110
e7	1	e0	0	1111
e8	0	e0	0	1100
e8	1	e0	0	1101
e9	0	e0	0	1010
e9	1	e0	1	1011
e10	0	e0	1	1000
e10	1	e0	1	1001
e11	0	e0	1	0110
e11	1	e0	0	0111
e12	0	e0	0	0100
e12	1	e0	0	0101
e13	0	e0	0	0010
e13	1	e0	1	0011
e14	0	e0	0	0000
e14	1	e0	1	0001

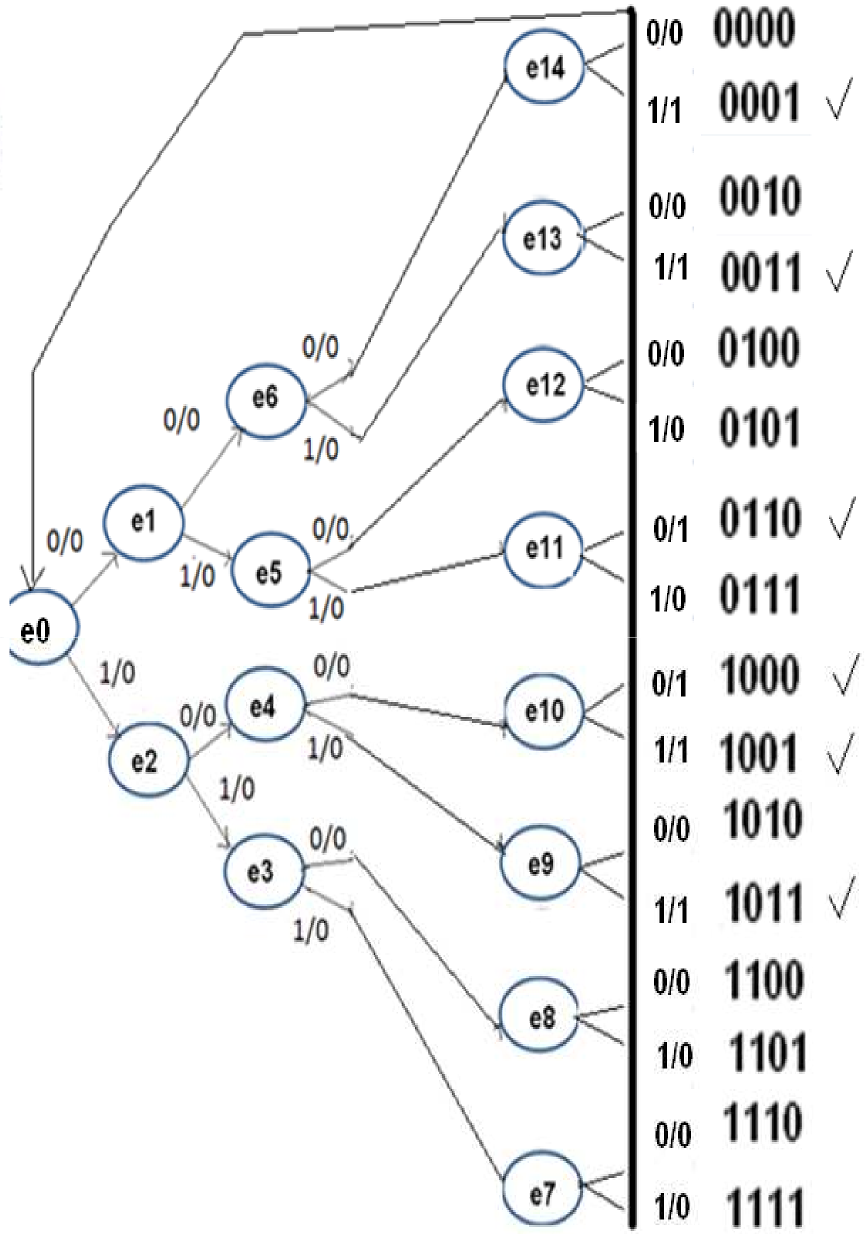


Tabla de Transicion

Q	ent	Q+1	sal
e0	0	e1	0
e0	1	e2	0
e1	0	e6	0
e1	1	e5	0
e2	0	e4	0
e2	1	e3	0
e3	0	e8	0
e3	1	e7	0
e4	0	e10	0
e4	1	e9	0
e5	0	e12	0
e5	1	e11	0
e6	0	e14	0
e6	1	e13	0

Q	ent	Q+1	sal	antecente
e7	0	e0	0	1110
e7	1	e0	0	1111
e8	0	e0	0	1100
e8	1	e0	0	1101
e9	0	e0	0	1010
e9	1	e0	1	1011
e10	0	e0	1	1000
e10	1	e0	1	1001
e11	0	e0	1	0110
e11	1	e0	0	0111
e12	0	e0	0	0100
e12	1	e0	0	0101
e13	0	e0	0	0010
e13	1	e0	1	0011
e14	0	e0	0	0000
e14	1	e0	1	0001

dec	4	4	1	-2
0	0	0	0	0
2	0	0	1	0
5	0	1	0	1
7	0	1	1	1
4	0	1	0	0
10	1	0	1	0
13	1	1	0	1
15	1	1	1	1
12	1	1	0	0
14	1	1	1	0

0,2,4,5,7,10,12,13,14,15

no aparecen

1,3,6,8,9,11

dec	bin
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111

Asignacion de estados

	dcba
e0	0000
e1	0001
e2	0010
e3	0011
e4	0100
e5	0101
e6	0110
e7	0111
e8	1000
e9	1001
e10	1010
e11	1011
e12	1100
e13	1101
e14	1110
~~X~~	1111

Tabla de Transicion

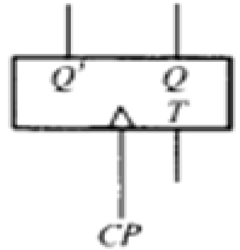
Q	ent	Q+1	sal
e0	0	e1	0
e0	1	e2	0
e1	0	e6	0
e1	1	e5	0
e2	0	e4	0
e2	1	e3	0
e3	0	e8	0
e3	1	e7	0
e4	0	e10	0
e4	1	e9	0
e5	0	e12	0
e5	1	e11	0
e6	0	e14	0
e6	1	e13	0

Q	ent	Q+1	sal	antecente
e7	0	e0	0	1110
e7	1	e0	0	1111
e8	0	e0	0	1100
e8	1	e0	0	1101
e9	0	e0	0	1010
e9	1	e0	1	1011
e10	0	e0	1	1000
e10	1	e0	1	1001
e11	0	e0	1	0110
e11	1	e0	0	0111
e12	0	e0	0	0100
e12	1	e0	0	0101
e13	0	e0	0	0010
e13	1	e0	1	0011
e14	0	e0	0	0000
e14	1	e0	1	0001

Pensemos implementarlo con FF T=>

Q	T	Q(t + 1)
0	0	0
0	1	1
1	0	1
1	1	0

Q	Q(t + 1)	T
0	0	0
0	1	1
1	1	0
1	0	1



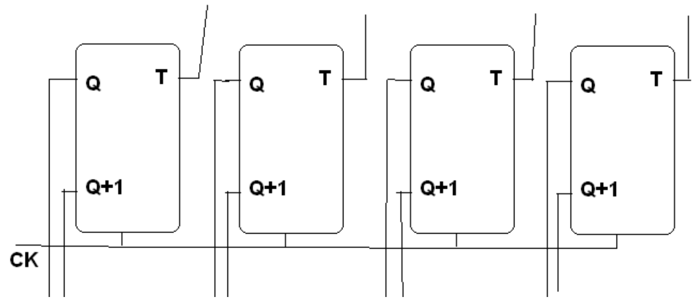
Separamos entradas ceros y unos:

Entrada=0

Q y3y2y1y0	Q+1	FF_s T t3 t2 t1 t0	sal
0000	0001	0 0 0 1	0
0001	0110	0 0 1 1	0
0010	0100	0 1 1 0	0
0011	1000	1 0 1 1	0
0100	1010	1 1 1 0	0
0101	1100	1 0 0 1	0
0110	1110	1 0 0 0	0
0111	0000	0 1 1 1	0
1000	0000	1 0 0 0	0
1001	0000	1 0 0 1	0
1010	0000	1 0 1 0	1
1011	0000	1 0 1 1	1
1100	0000	1 1 0 0	0
1101	0000	1 1 0 1	0
1110	0000	1 1 1 0	0

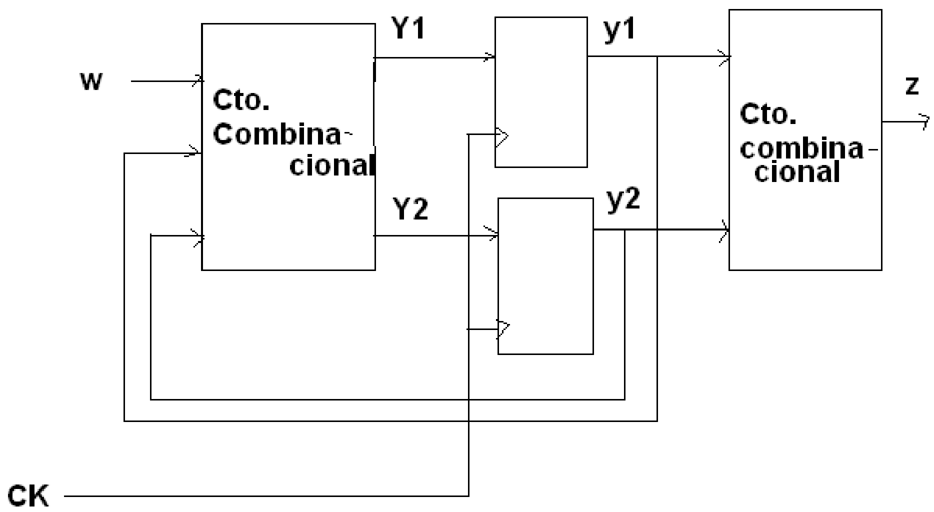
Entrada =1

Q y3y2y1y0	Q+1	FF_s T t3 t2 t1 t0	sal
0000	0010	0 0 1 0	0
0001	0101	0 1 0 0	0
0010	0011	0 0 0 1	0
0011	0111	0 1 0 0	0
0100	1001	1 1 0 1	0
0101	1011	1 1 1 0	0
0110	1101	1 1 1 1	0
0111	0000	0 1 1 1	0
1000	0000	1 0 0 0	0
1001	0000	1 0 0 1	1
1010	0000	1 0 1 0	1
1011	0000	1 0 1 1	0
1100	0000	1 1 0 0	0
1101	0000	1 1 0 1	1
1110	0000	1 1 1 0	1



T3= (3,4,5,6,8,9,10,11,12,13,14)

Ejercicio para el alumno.
Obtén los ts y el circuito correspondiente.
Pero, primero comprendamos como se construye un circuito…



Entrada=0

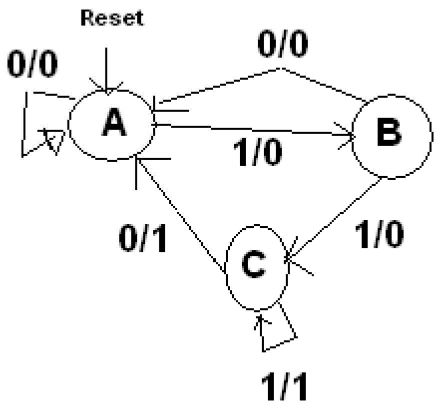
Q (y3y2y1y0)	Q+1	FFs T (t3 t2 t1 t0)	sal
0000	0001	0 0 0 1	0
0001	0110	0 0 1 1	0
0010	0100	0 1 1 0	0
0011	1000	1 0 1 1	0
0100	1010	1 1 1 0	0
0101	1100	1 0 0 1	0
0110	1110	1 0 0 0	0
0111	0000	0 1 1 1	0
1000	0000	1 0 0 0	0
1001	0000	1 0 0 1	0
1010	0000	1 0 1 0	1
1011	0000	1 0 1 1	1
1100	0000	1 1 0 0	0
1101	0000	1 1 0 1	0
1110	0000	1 1 1 0	0

Entrada =1

Q (y3y2y1y0)	Q+1	FFs T (t3 t2 t1 t0)	sal
0000	0010	0 0 1 0	0
0001	0101	0 1 0 0	0
0010	0011	0 0 0 1	0
0011	0111	0 1 0 0	0
0100	1001	1 1 0 1	0
0101	1011	1 1 1 0	0
0110	1101	1 1 1 1	0
0111	0000	0 1 1 1	0
1000	0000	1 0 0 0	0
1001	0000	1 0 0 1	1
1010	0000	1 0 1 0	1
1011	0000	1 0 1 1	0
1100	0000	1 1 0 0	0
1101	0000	1 1 0 1	1
1110	0000	1 1 1 0	1

Ejemplo 3. Construye un circuito.

Se escoge el tipo de FF
W es una entrada (0,1)



Q	Q+1		Sal
	w=0	w=1	
y2y1	Y2Y1	Y2Y1	z
A	A	B	0
B	A	C	0
C	A	C	1

y2y1

w \ y2y1	00	01	11	10
0	0	0	x	0
1	1	0	x	0

y2y1

w \ y2y1	00	01	11	10
0	0	0	x	0
1	0	1	x	1

Q	Q+1		Sal
	w=0	w=1	
y2y1	Y2Y1	Y2Y1	z
00	00	01	0
01	00	10	0
10	00	10	1
11	xx	xx	x

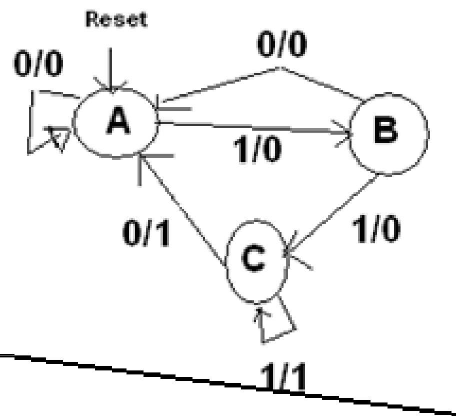
$Y1=\overline{w}y1y2$
$Y2=w(y1+y2)$
$z=y2$

y2 \ y1	0	1
0	0	0
1	1	x

$Y1=\overline{w}y1y2$
$Y2=w(y1+y2)$
$z=y2$

Ejemplo 3. Construye un circuito.

Se escoge el tipo de FF
W es una entrada (0,1)



Q	Q+1		Sal
	w=0	w=1	
y2y1	Y2Y1	Y2Y1	z
A	A	B	0
B	A	C	0
C	A	C	1

Q	Q+1		Sal
	w=0	w=1	
y2y1	Y2Y1	Y2Y1	z
00	00	01	0
01	00	10	0
10	00	10	1
11	xx	xx	x

$Y1=\overline{w}y1y2$

w \ y2y1	00	01	11	10
0	0	0	x	0
1	1	0	x	0

$Y2=w(y1 \oplus y2)$

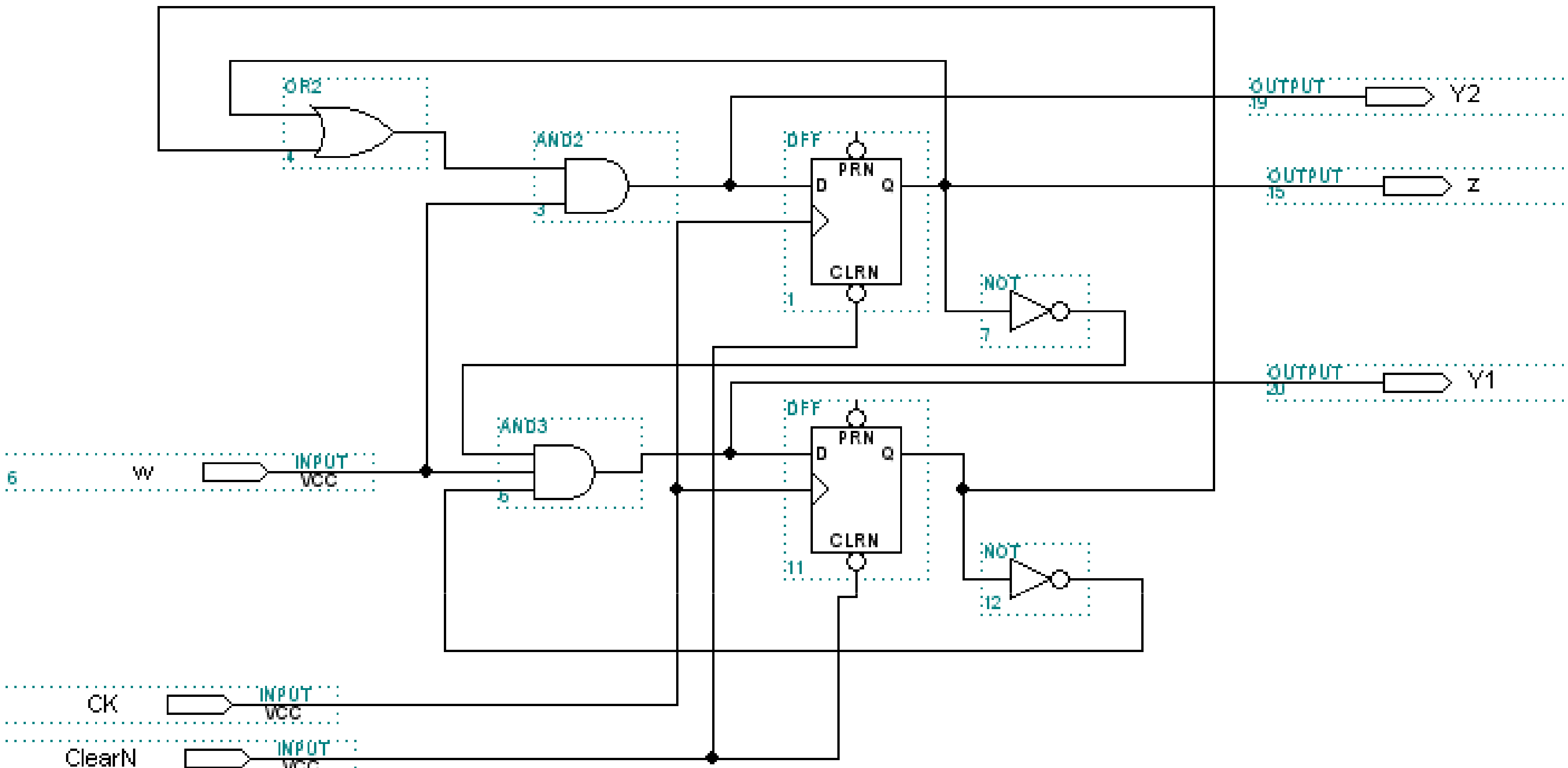
w \ y2y1	00	01	11	10
0	0	0	x	0
1	0	1	x	1

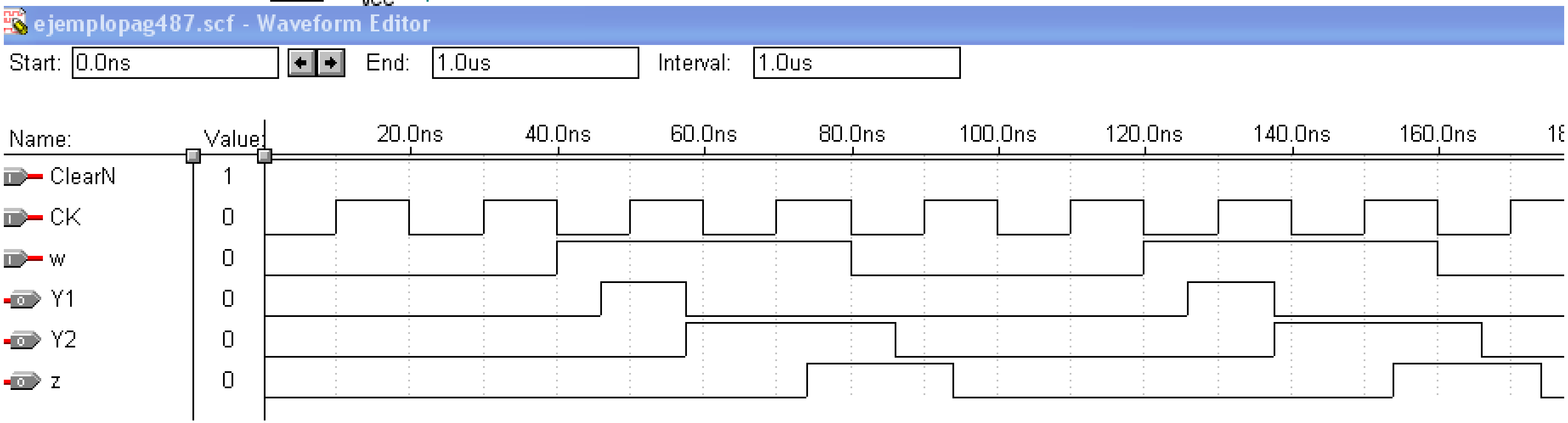
$Y1=wy2'y1'$
$Y2=wy2'y1 + wy2y1'$
$z=y2y1'$

z=y2

y2 \ y1	0	1
0	0	0
1	1	x

$Y1=\overline{w}y1y2$
$Y2=w(y1+y2)$
$z=y2$


OR2
AND2
DFF
PRN
D
Q
CLRN
NOT
OUTPUT
Y2
OUTPUT
Z
OUTPUT
Y1
DFF
PRN
D
Q
CLRN
NOT
AND3
w
INPUT
VCC
CK
INPUT
VCC
ClearN
INPUT
VCC


ejemplopag487.scf - Waveform Editor
Start: 0.0ns
End: 1.0us
Interval: 1.0us
Name:
Value:
20.0ns
40.0ns
60.0ns
80.0ns
100.0ns
120.0ns
140.0ns
160.0ns
ClearN 1
CK 0
w 0
Y1 0
Y2 0
z 0

Diseño de Contadores

Tabla de excitación para un contador binario de tres bits

Secuencia de cuenta			Entradas del flip-flop		
A_2	A_1	A_0	TA_2	TA_1	TA_0
0	0	0	0	0	1
0	0	1	0	1	1
0	1	0	0	0	1
0	1	1	1	1	1
1	0	0	0	0	1
1	0	1	0	1	1
1	1	0	0	0	1
1	1	1	1	1	1

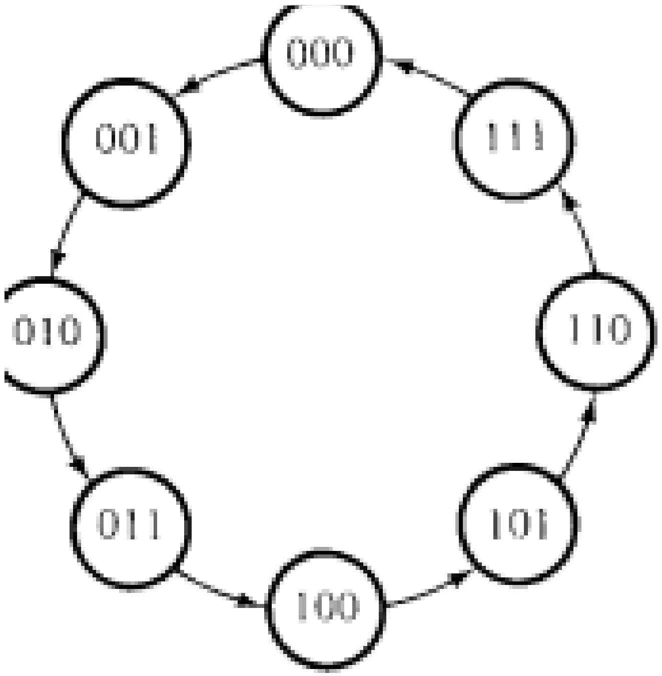
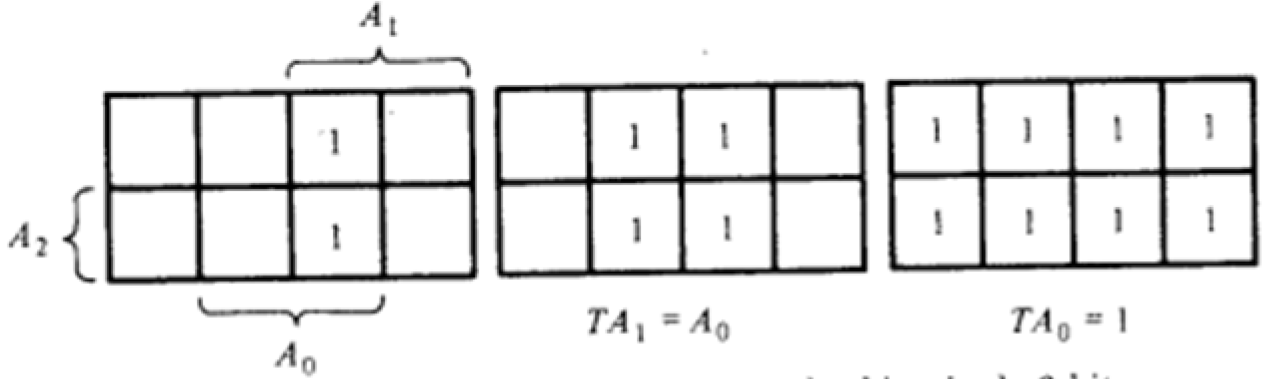


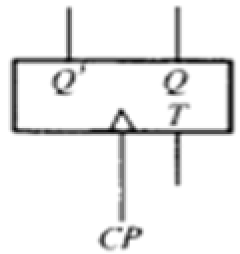
Diagrama de estado de un contador binario de 3 bits



$TA_2 = A_1 A_0$ $TA_1 = A_0$ $TA_0 = 1$

Mapas para un contador binario de 3 bits

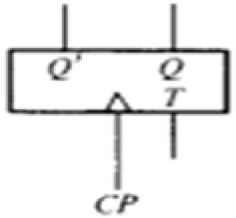
Q	$Q(t+1)$	T
0	0	0
0	1	1
1	1	0
1	0	1

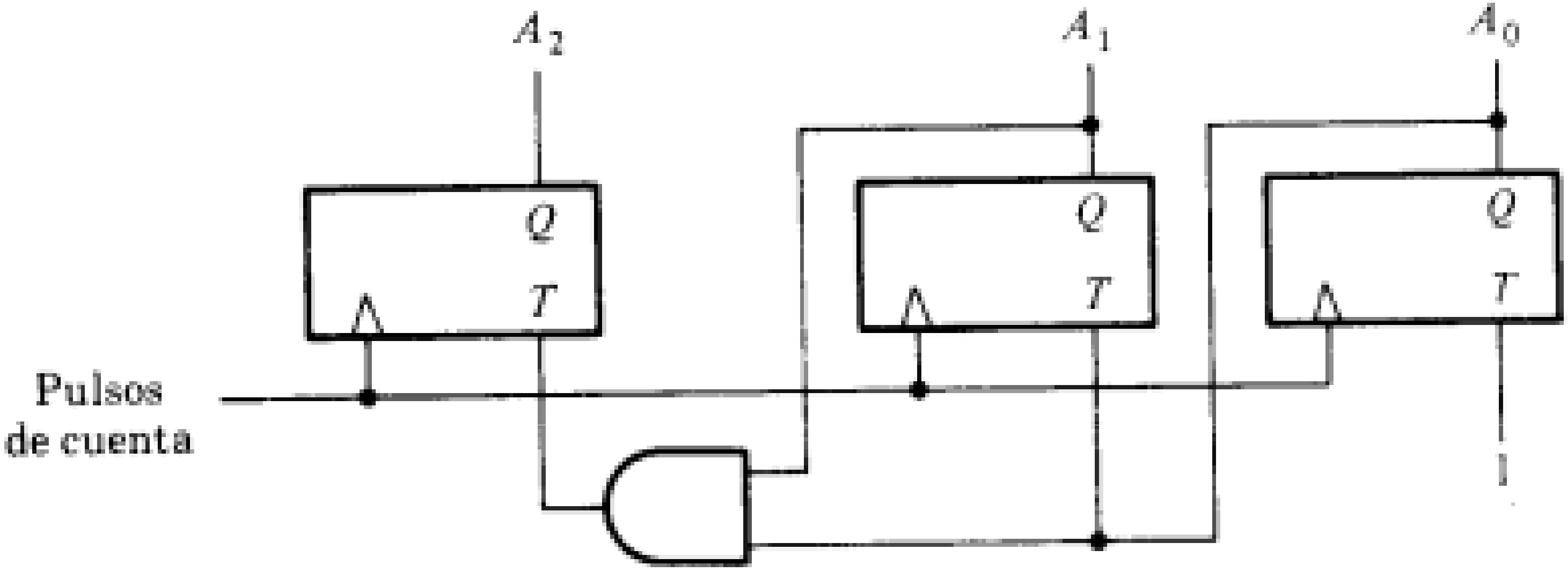


Q	Q+1	T
0000	0001	0001
0001	0010	0011
0010	0011	0001
0011	0100	0111
...		

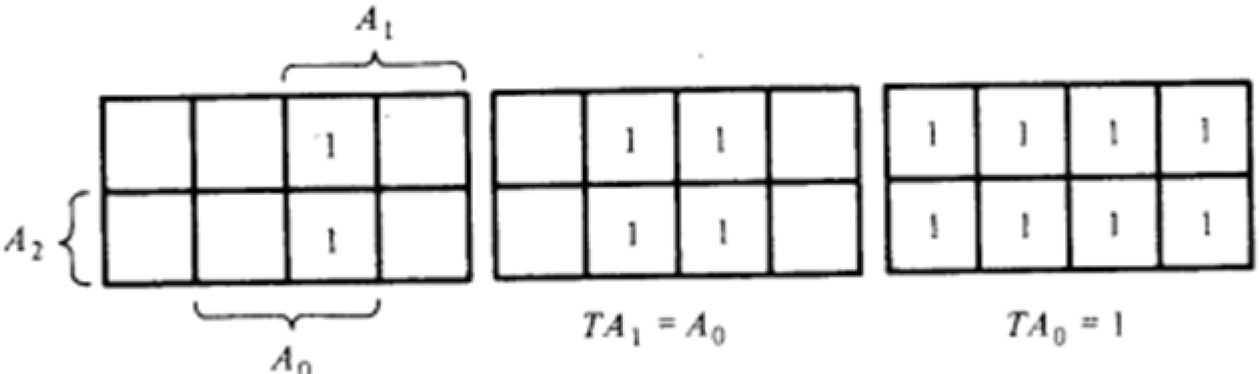
Q	T	$Q(t+1)$
0	0	0
0	1	1
1	0	1
1	1	0

Q	$Q(t+1)$	T
0	0	0
0	1	1
1	1	0
1	0	1





binario de 3 bits



Mapas para un contador binario de 3 bits

EJEMPLO Diséñese un contador que tenga una secuencia repetida de seis estados como la listada en la Tabla 6-13.

En esta secuencia, los flip-flops B y C repiten la cuenta binaria 00, 01, 10 mientras que el flip-flop A alterna entre los estados 0 y 1 cada tres cuentas. La secuencia de cuenta para A, B, C no es binaria directa y los dos estados 011 y 111 no se usan. La escogencia de los flip-flops JK resulta en una tabla de excitación de la Tabla 6-13.

Q	J	K	$Q(t+1)$
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

Tabla 6-13 Tabla de excitación para el Ejemplo 6-3

Secuencia de cuenta			Entradas del flip-flop					
A	B	C	JA	KA	JB	KB	JC	KC
0	0	0	0	X	0	X	1	X
0	0	1	0	X	1	X	X	1
0	1	0	1	X	X	1	0	X
1	0	0	X	0	0	X	1	X
1	0	1	X	0	1	X	X	1
1	1	0	X	1	X	1	0	X

Las entradas KB y KC tienen solamente 1 y X en sus columnas, de tal manera que esas entradas sean siempre 1.

$JA = B \qquad KA = B$

$JB = C \qquad KB = I$

$JC = B' \qquad KC = I$

Entendiendo…

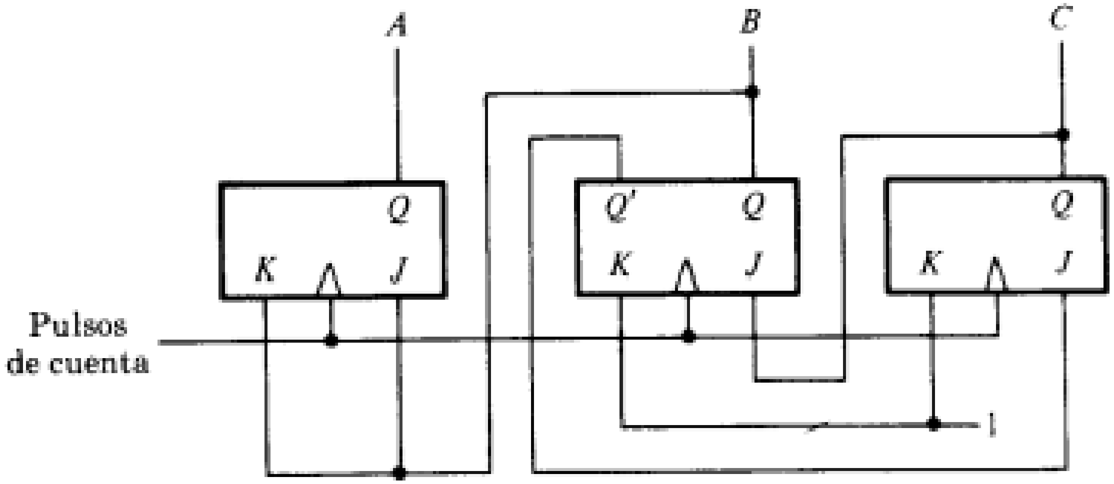
Q	Q+1	jajbjc	kakbkc
000	001	0 0 1	x x x
001	010	0 1 x	x x 1
010	100	1 x 0	x 1 x
100	101	x 0 1	0 x x
101	110	x 1 x	0 x 1

Q	Q+1	j	k
0	0	0	x
0	1	1	x
1	0	x	1
1	1	x	0

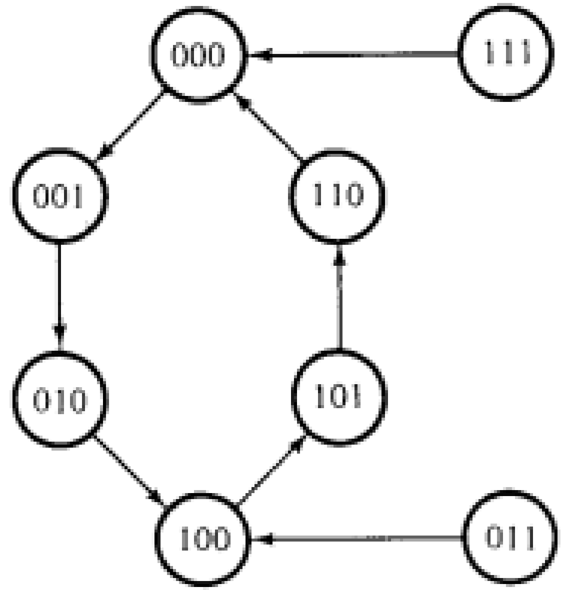
	(t + 1)		
Q	Q	J	K
0	0	0	0
0	0	0	1
0	1	1	0
0	1	1	1
1	0	1	1
1	0	0	1
1	1	1	0
1	1	0	0

Secuencia de cuenta			Entradas del flip-flop					
A	*B*	*C*	*JA*	*KA*	*JB*	*KB*	*JC*	*KC*
0	0	0	0	*X*	0	*X*	1	*X*
0	0	1	0	*X*	1	*X*	*X*	1
0	1	0	1	*X*	*X*	1	0	*X*
1	0	0	*X*	0	0	*X*	1	*X*
1	0	1	*X*	0	1	*X*	*X*	1
1	1	0	*X*	1	*X*	1	0	*X*

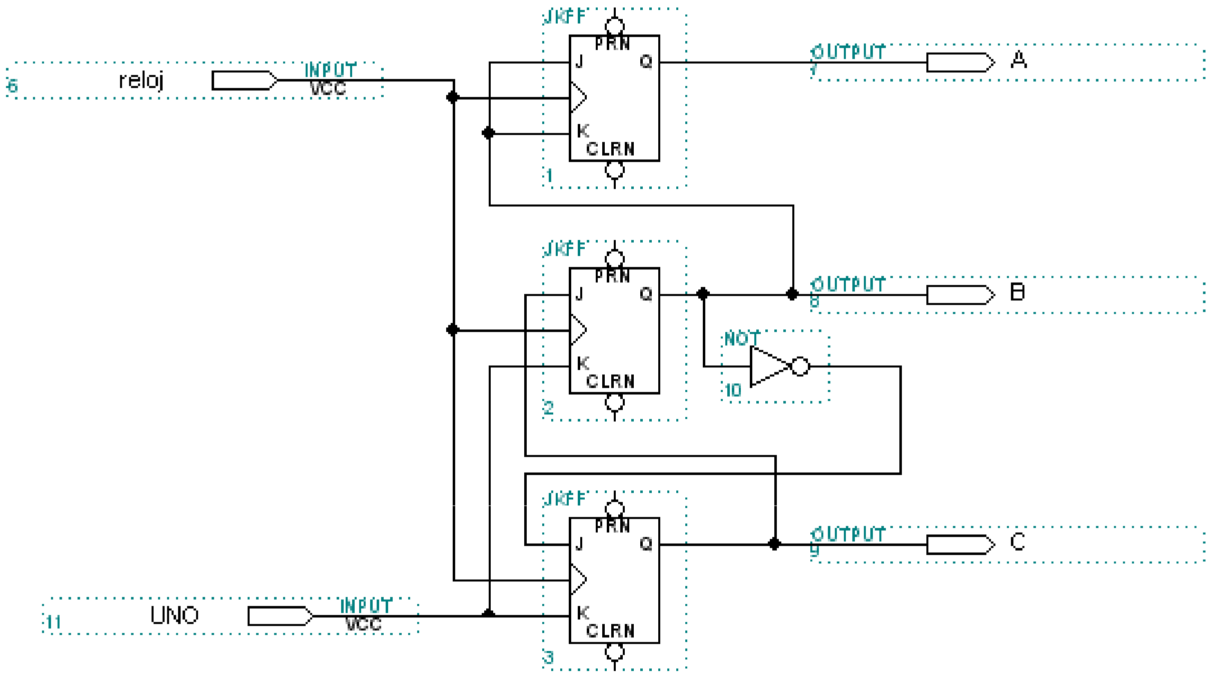
Solución



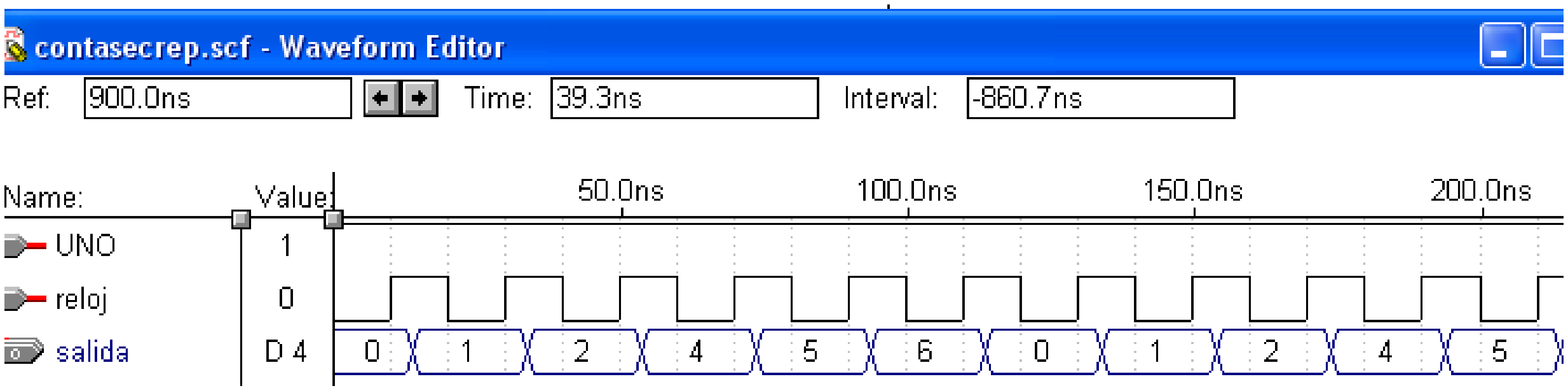
(a) Diagrama lógico del contador



(b) Diagrama de estado del contador



reloj
INPUT
VCC
UNO
INPUT
VCC
JKFF
PRN
J
Q
K
CLRN
OUTPUT
A
OUTPUT
B
OUTPUT
C
NOT



contasecrep.scf - Waveform Editor
Ref: 900.0ns
Time: 39.3ns
Interval: -860.7ns
Name:
Value:
50.0ns
100.0ns
150.0ns
200.0ns
UNO 1
reloj 0
salida D 4
0 1 2 4 5 6 0 1 2 4 5