

Departamento de Arquitectura Instituto de Computación
Universidad de la República
Montevideo - Uruguay

Notas de Teórico

Contadores

Arquitectura de Computadoras
(Versión 4.3 - 2016)

8 CONTADORES

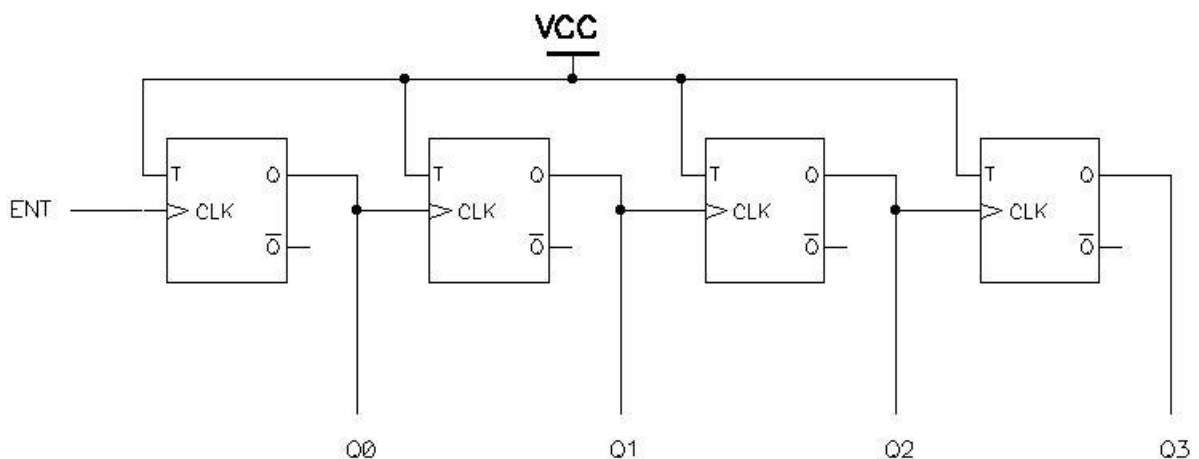
8.1 Contadores

Ahora nos concentraremos en un caso particular de circuito secuencial: el *contador*. El contador es un circuito cuyas salidas representan, en un código binario dado, la cuenta de la cantidad de cambios en su entrada, módulo su capacidad.

Usualmente se utilizan contadores binarios (que usan el código binario para representar números naturales) aunque se pueden diseñar contadores que utilicen el código Gray, el entrelazado 2 de 5, el decimal empaquetado o cualquier otro que se pueda considerar. Normalmente cuentan las transiciones de la entrada de 0 a 1 (flancos ascendentes). No tienen que empezar en 0, ni su módulo tiene porqué ser una potencia de 2.

8.1.1 Contador en base a Flip-Flops T

La manera más simple de construir un contador es a partir de flip-flops T, de la siguiente manera:



Para analizar el comportamiento del circuito supongamos que inicialmente todas las salidas de los FF están en 0.

Al pasar la entrada ENT de 0 a 1 generará un flanco ascendente en la entrada de reloj del primer flip-flop por lo que su salida Q0 pasará a 1. Esto a su vez generará un flanco ascendente en la entrada de reloj del segundo flip-flop, llevando su salida Q1 a 1, lo que genera otro flanco en el tercer flip-flop y algo análogo ocurre con el cuarto flip-flop. Esto quiere decir que la situación luego del primer flanco es:

	Q3	Q2	Q1	Q0
Inicial	0	0	0	0
Primer flanco	1	1	1	1

Al ocurrir el segundo flanco el primer flip-flop vuelve a cambiar, pero no genera un flanco ascendente en la entrada de reloj del segundo, por lo que éste queda incambiado, lo mismo que el resto de la cascada.

Hasta entonces la secuencia es:

	Q3	Q2	Q1	Q0
Inicial	0	0	0	0
Primer flanco	1	1	1	1
Segundo flanco	1	1	1	0

Al llegar el tercer flanco en la entrada ENT el primer flip-flop vuelve a cambiar su salida, la que pasa a 1. Esto sí genera un nuevo flanco ascendente en el segundo flip-flop que lleva a cambiar su salida Q1 de 1 a 0. Como esto no genera un flanco ascendente, el resto de los flip-flops permanece igual:

	Q3	Q2	Q1	Q0
Inicial	0	0	0	0
Primer flanco	1	1	1	1
Segundo flanco	1	1	1	0
Tercer flanco	1	1	0	1

En el siguiente flanco ascendente (cuarto) de la entrada ENT vuelve a cambiar Q0, pero como pasa de 1 a 0 el resto sigue igual. Pero al próximo (quinto) Q0 pasa de 0 a 1, generando un flanco a la entrada del segundo flip-flop, lo que mueve a Q1 de 0 a 1 y esto representa un flanco ascendente en la entrada del tercer flip-flop que, entonces, cambia de 1 a 0. Notar que el cuarto flip-flop no se modifica:

	Q3	Q2	Q1	Q0
Inicial	0	0	0	0
Primer flanco	1	1	1	1
Segundo flanco	1	1	1	0
Tercer flanco	1	1	0	1
Cuarto flanco	1	1	0	0
Quinto flanco	1	0	1	1

El análisis continúa de la misma manera hasta llegar al dieciseisavo flanco quedando:

	Q3	Q2	Q1	Q0
Inicial	0	0	0	0
Primer flanco	1	1	1	1
Segundo flanco	1	1	1	0
Tercer flanco	1	1	0	1
Cuarto flanco	1	1	0	0
Quinto flanco	1	0	1	1
Sexto flanco	1	0	1	0
Séptimo flanco	1	0	0	1
Octavo flanco	1	0	0	0
Noveno flanco	0	1	1	1
Décimo flanco	0	1	1	0
Onceavo flanco	0	1	0	1
Doceavo flanco	0	1	0	0
Treceavo flanco	0	0	1	1
Catorceavo flanco	0	0	1	0
Quinceavo flanco	0	0	0	1
Dieciseisavo flanco	0	0	0	0

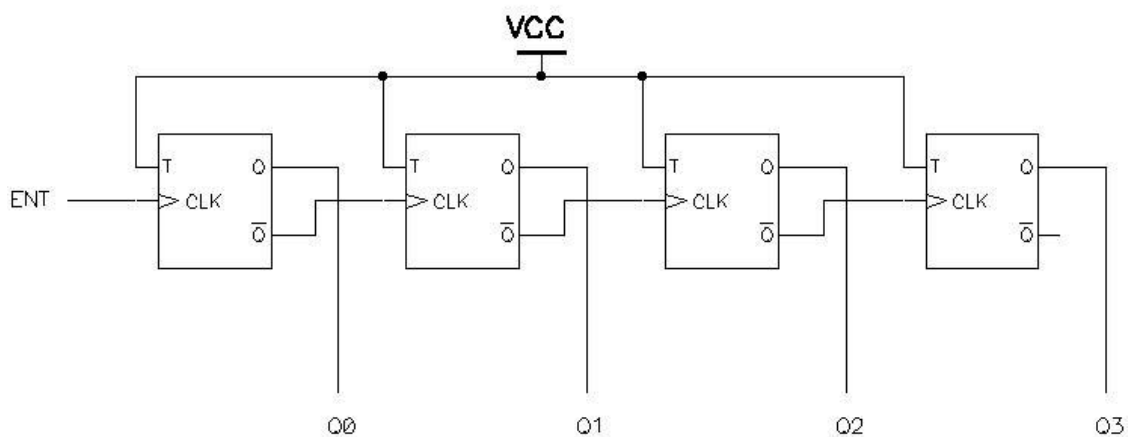
Notemos que al cabo de 16 flancos llegamos a la misma condición inicial, por lo que a partir de ese momento la secuencia se repetirá.

Si observamos la tabla que quedó notamos enseguida que las salidas Q3, Q2, Q1, Q0 forman un número en binario de 4 bits que va disminuyendo desde 1111 (15) hasta 0000 (0) en forma cíclica (llegado a 0 luego vuelve a 15 y comienza de nuevo a disminuir)

Por tanto el circuito formado por los flip-flops T conectados de esa manera se comporta como un *contador binario descendente* de 4 bits.

Para tener un número mayor de bits (y por tanto mayor capacidad de cuenta) basta con agregar más flip-flops T conectados de manera análoga. Por cada flip-flop que se agrega se duplica la capacidad de cuenta.

Si quisiéramos que la cuenta fuera ascendente deberíamos usar el siguiente esquema de conexión de los flip-flops:



8.1.2 Contador en base a Flip-Flops D

Para analizar como diseñar un contador en base a flip-flops tipo D vamos a proponernos construir un contador ascendente que cuente en código Gray entre 1 y 10. Si bien es un ejemplo particular la metodología que desarrollaremos se puede aplicar al diseño de cualquier otro contador. Como usaremos flip-flops tipo D sincrónicos por flanco, el contador será un contador de flancos (transiciones de 0 a 1) de la entrada.

Dada la especificación lo primero a determinar es la cantidad de flip-flops necesarios. Para ello veamos como se va construyendo un código Gray:

```

0
1
----
11
10
----
110
111
101
100
----
1100
1101
1111

```

```

1110
1010
1011
1001
1000

```

Notar que el código se genera espejando lo ya hecho hasta los lugares potencia de dos (indicados por las líneas punteadas) agregándole un 1 delante.

Si tomamos, arbitrariamente, que el primer código representa el 0, nos queda la siguiente tabla de codificación:

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

Para el caso que nos ocupa tomamos los códigos binarios desde el 1 al 10. Obviamente, y como era de esperar, precisamos 4 dígitos para representar los códigos Gray de estos números, por lo que nuestro contador requiere de 4 flip-flops.

El siguiente paso es escribir una tabla que nos indique en función del valor actual de las salidas Q3, Q2, Q1, Q0, cuál deben ser los nuevos valores de forma que el circuito se comporte como deseamos (notar que por haber elegido un número de cuenta que no es potencia de 2, el código no queda "perfectamente" cíclico en Gray):

	Actuales	Nuevos
	Q3Q2Q1Q0	Q3Q2Q1Q0
1	0 0 0 1	0 0 1 1
2	0 0 1 1	0 0 1 0
3	0 0 1 0	0 1 1 0
4	0 1 1 0	0 1 1 1
5	0 1 1 1	0 1 0 1
6	0 1 0 1	0 1 0 0
7	0 1 0 0	1 1 0 0
8	1 1 0 0	1 1 0 1
9	1 1 0 1	1 1 1 1
10	1 1 1 1	0 0 0 1

Recordando que la ecuación característica de los flip-flops tipo D es:

$$Q_{n+1} = D_n$$

esto es que el nuevo valor de la salida del flip-flop corresponde al valor actual de la entrada, para lograr los nuevos valores indicados en la tabla para los Q_j , basta con colocar en su entrada actual esos valores.

De esta manera nos queda una tabla de verdad de un circuito combinatorio, cuyas entradas son los valores actuales de los Q_j y sus salidas son los valores actuales de D_j necesarios para lograr la transición deseada.

Tenemos entonces la siguiente tabla de verdad del circuito combinatorio:

Q3	Q2	Q1	Q0	D3	D2	D1	D0
0	0	0	1	0	0	1	1
0	0	1	1	0	0	1	0
0	0	1	0	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	0	1	0	1
0	1	0	1	0	1	0	0
0	1	0	0	1	1	0	0
1	1	0	0	1	1	0	1
1	1	0	1	1	1	1	1
1	1	1	1	0	0	0	1

El siguiente paso consiste en minimizar las expresiones de las ecuaciones de los D_j utilizando Karnaugh. Usaremos comodines ("don't care") en las cuentas que no son alcanzadas (0, y 11 a 15).

D3

Q ₃ Q ₂ / Q ₁ Q ₀	00	01	11	10
00	X	1	1	X
01			1	X
11				X
10			X	X

D2

Q ₃ Q ₂ / Q ₁ Q ₀	00	01	11	10
00	X	1	1	X
01		1	1	X
11		1		X
10	1	1	X	X

D1

Q ₃ Q ₂ / Q ₁ Q ₀	00	01	11	10
00	X			X
01	1		1	X
11	1			X
10	1	1	X	X

D0

Q_3Q_2 / Q_1Q_0	00	01	11	10
00	X		1	X
01	1		1	X
11		1	1	X
10		1	X	X

Las ecuaciones resultantes son:

$$D_3 = Q_1'Q_0' + Q_3Q_1' \quad D_2 = Q_0' + Q_3'Q_2 + Q_2Q_1'$$

$$D_1 = Q_2' + Q_1Q_0' + Q_3Q_1'Q_0 \quad D_0 = Q_3 + Q_2'Q_1' + Q_2Q_1$$

El circuito completo queda como se muestra en la próxima página:

