

CONTADORES ESPECIALES

Existen algunos contadores que por sus características se utilizan en aplicaciones especiales. Esos contadores son:

- Contador BCD
- Contador de Anillo.
- Contador Johnson

Contador BCD.

Es un contador que muestra el código BCD a su salida. El autómata de contador BCD se muestra en la ilustración 1.

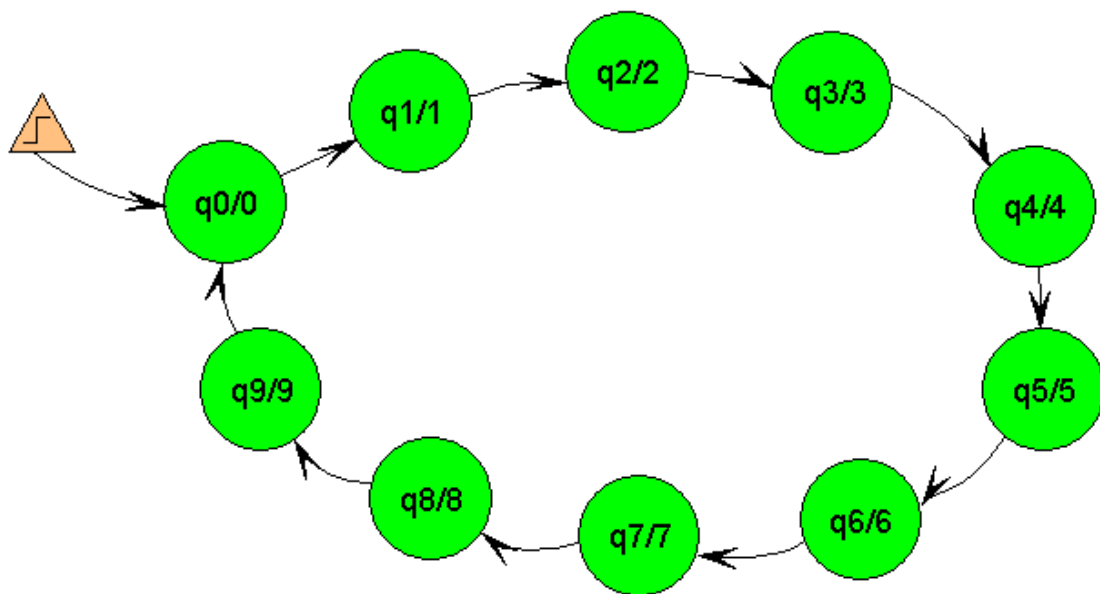


Ilustración 1 Autómata del contador BCD.

Descripción del contador como máquina de Moore

$M = (Q, \Sigma, \Delta, \delta, \lambda, q_0)$	$\delta(q_0, \varepsilon) = q_1,$	$\lambda(q_0) = 0$
$Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8, q_9\}$	$\delta(q_1, \varepsilon) = q_2,$	$\lambda(q_1) = 1$
$\Sigma = \{0, 1\}$	$\delta(q_2, \varepsilon) = q_3,$	$\lambda(q_2) = 2$
$\Delta = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$	$\delta(q_3, \varepsilon) = q_4,$	$\lambda(q_3) = 3$
	$\delta(q_4, \varepsilon) = q_5,$	$\lambda(q_4) = 4$
	$\delta(q_5, \varepsilon) = q_6,$	$\lambda(q_5) = 5$
	$\delta(q_6, \varepsilon) = q_7,$	$\lambda(q_6) = 6$
	$\delta(q_7, \varepsilon) = q_8,$	$\lambda(q_7) = 7$
	$\delta(q_8, \varepsilon) = q_9,$	$\lambda(q_8) = 8$
	$\delta(q_9, \varepsilon) = q_0,$	$\lambda(q_9) = 9$

Tabla de transiciones

Estado actual $Q(t)$	Entrada CLK	Estado siguiente $Q(t + 1)$	Salida
q_0	↑	q_1	0
q_1	↑	q_2	1
q_2	↑	q_3	2
q_3	↑	q_4	3
q_4	↑	q_5	4
q_5	↑	q_6	5
q_6	↑	q_7	6
q_7	↑	q_8	7
q_8	↑	q_9	8
q_9	↑	q_0	9

Estos contadores son usados para generar el código decimal o numeración en base 10.

Por ejemplo cuando se quiere mostrar el conteo en display de 7 segmentos se usan estos contadores para llevar el conteo en decimal. Posteriormente con un convertidor de código BCD a 7 segmentos se obtienen los códigos para mandar a los displays.

Otro ejemplo de aplicación de estos contadores es cuando se diseña un reloj de tiempo real. Los segundos, minutos y horas se manejan usando estos contadores.

Contador de Anillo.

Es un registro de corrimiento circular en el que un solo Flip-Flop esta activo en cualquier instante. El FF puede ser activo en alto, en este caso es un contador de anillo con un uno viajero. También el FF puede ser activo en bajo, en este caso es un contador de anillo con un cero viajero. En este documento se analizará el contador de anillo con un uno viajero.

Para un contador de anillo de “n” bits se tienen “n” flip-flops y “n” estados.

El autómata de contador de anillo de 4 bits se muestra en la ilustración 2.

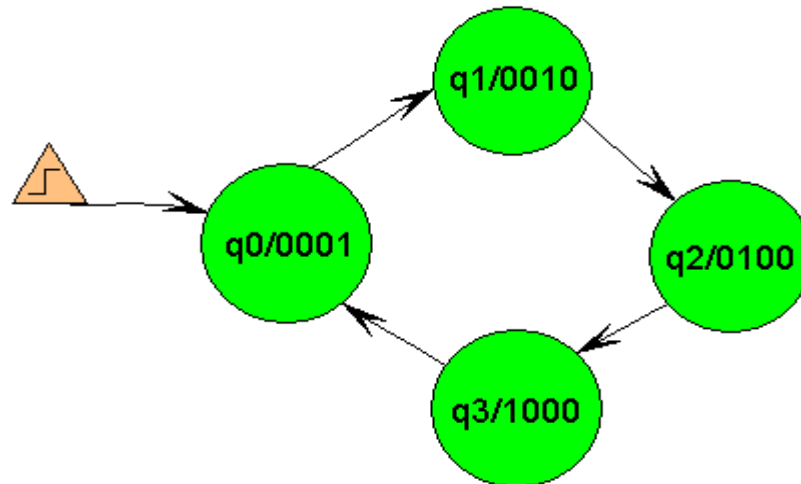


Ilustración 2 Autómata del contador de anillo

Descripción del contador como máquina de Moore

$$\begin{array}{lll}
 M = (Q, \Sigma, \Delta, \delta, \lambda, q_0) & \delta(q_0, \varepsilon) = q_1, & \lambda(q_0) = 0001 \\
 Q = \{q_0, q_1, q_2, q_3\} & \delta(q_1, \varepsilon) = q_2, & \lambda(q_1) = 0010 \\
 \Sigma = \{0, 1\} & \delta(q_2, \varepsilon) = q_3, & \lambda(q_2) = 0100 \\
 \Delta = \{0001, 0010, 0100, 1000\} & \delta(q_3, \varepsilon) = q_0 & \lambda(q_3) = 1000
 \end{array}$$

Tabla de transiciones

Estado actual $Q(t)$	Entrada CLK	Estado siguiente $Q(t+1)$	Salida
q_0	↑	q_1	0001
q_1	↑	q_2	0010
q_2	↑	q_3	0100
q_3	↑	q_0	1000

El circuito resultante del diseño usando la tabla de transiciones anterior se muestra en la ilustración 3.

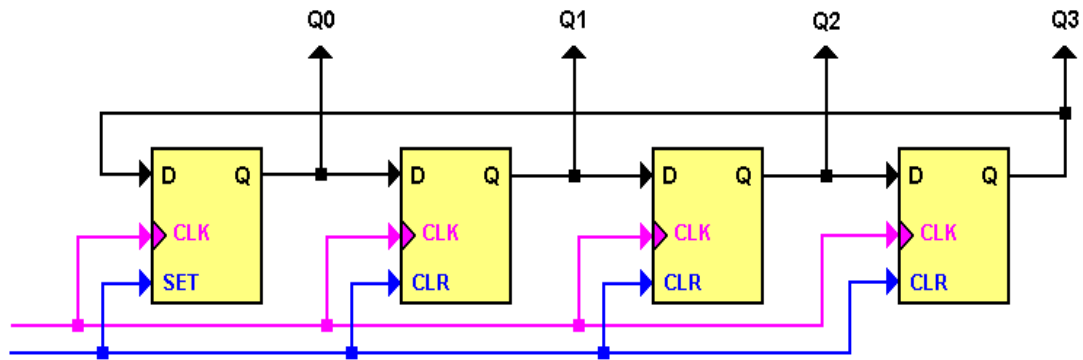


Ilustración 3 Contador de anillo de 4 bits.

En la ilustración 3 se puede notar que la señal de CLR es común para todos los FF's excepto para el primer FF (Q_0), la cual se conecta en la terminal SET. Esto provoca que al momento de activar la señal de CLR solo el primer FF se coloca en '1' y el resto en '0'. El funcionamiento del contador de anillo se muestra en el diagrama de tiempos de la ilustración 4.

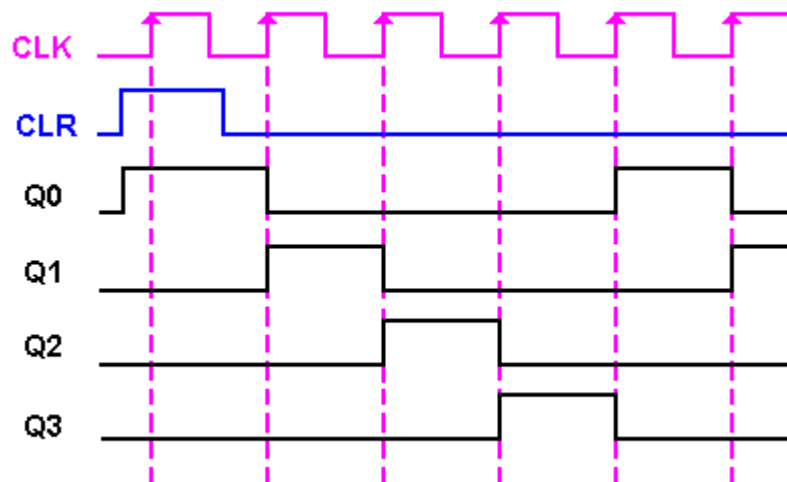


Ilustración 4 Diagrama de tiempo del contador de anillo

Para programar este contador usando VHDL se pueden usar muchas estrategias, una de ellas consiste en usar el operador ROL como se muestra en la ilustración 5.

```

1  LIBRARY IEEE;
2  USE IEEE.STD_LOGIC_1164.all;
3
4  ENTITY ANILLO IS
5      PORT(
6          CLR, CLK: IN STD_LOGIC;
7          Q : INOUT STD_LOGIC_VECTOR(3 DOWNTO 0)
8      );
9  END ANILLO;
10
11 ARCHITECTURE CONTADOR OF ANILLO IS
12 BEGIN
13     PANILLO : PROCESS (CLK, CLR)
14     BEGIN
15         IF ( CLR = '1' ) THEN
16             Q <= "0001";
17         ELSIF ( CLK'EVENT AND CLK = '1' ) THEN
18             Q <= TO_STDLOGICVECTOR( TO_BITVECTOR( Q ) ROL 1 );
19         END IF;
20     END PROCESS PANILLO;
21 END CONTADOR;

```

Ilustración 5 Programa en VHDL del contador de anillo de 4 bits

Contador Johnson.

Es un registro de corrimiento circular en el que la salida complementada del último Flip-Flop (Q_{MSB}) se conecta a la entrada del primer FF (Q_0).

Para un contador Johnson de “n” bits se tienen “n” flip-flops y “2n” estados.

El autómata de contador Johnson de 4 bits se muestra en la ilustración 6.

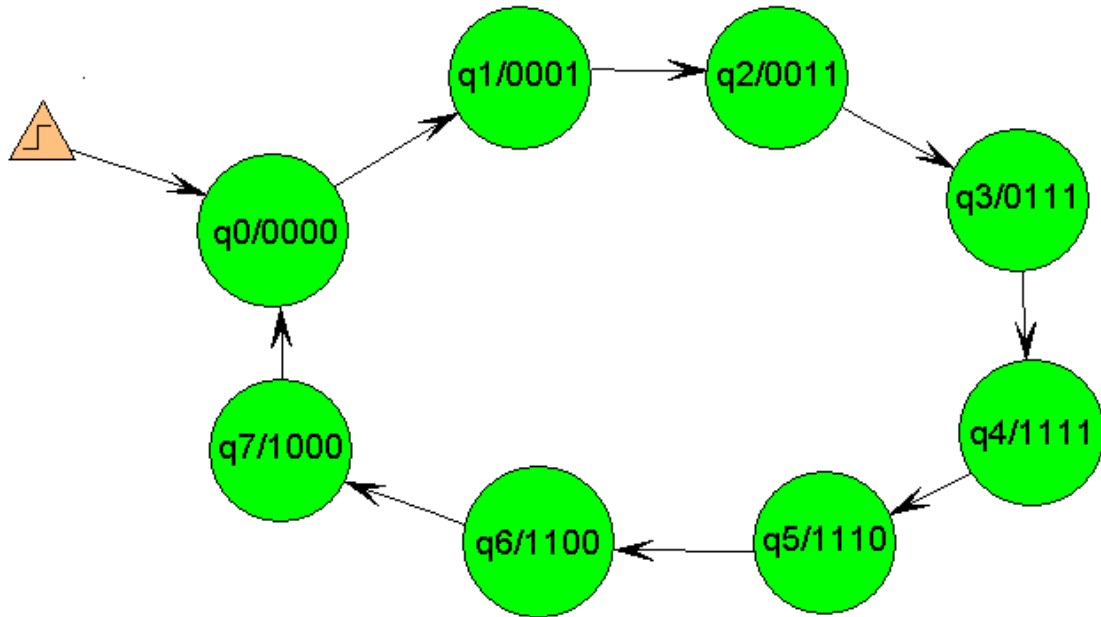


Ilustración 6 Autómata del contador Johnson

Descripción del contador como máquina de Moore

$M = (Q, \Sigma, \Delta, \delta, \lambda, q_0)$	$\delta(q_0, \varepsilon) = q_1,$	$\lambda(q_0) = 0000$
$Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7\}$	$\delta(q_1, \varepsilon) = q_2,$	$\lambda(q_1) = 0001$
$\Sigma = \{0,1\}$	$\delta(q_2, \varepsilon) = q_3,$	$\lambda(q_2) = 0011$
$\Delta = \left\{ \begin{array}{l} 0000, 0001, 0011, 0111, \\ 1111, 1110, 1100, 1000 \end{array} \right\}$	$\delta(q_3, \varepsilon) = q_4,$	$\lambda(q_3) = 0111$
	$\delta(q_4, \varepsilon) = q_5,$	$\lambda(q_4) = 1111$
	$\delta(q_5, \varepsilon) = q_6,$	$\lambda(q_5) = 1110$
	$\delta(q_6, \varepsilon) = q_7,$	$\lambda(q_6) = 1100$
	$\delta(q_7, \varepsilon) = q_0$	$\lambda(q_7) = 1000$

Tabla de transiciones

Estado actual $Q(t)$	Entrada CLK	Estado siguiente $Q(t + 1)$	Salida
q_0	↑	q_1	0000
q_1	↑	q_2	0001
q_2	↑	q_3	0011
q_3	↑	q_4	0111
q_4	↑	q_5	1111
q_5	↑	q_6	1110
q_6	↑	q_7	1100
q_7	↑	q_0	1000

El circuito resultante del diseño usando la tabla de transiciones anterior se muestra en la ilustración 7.

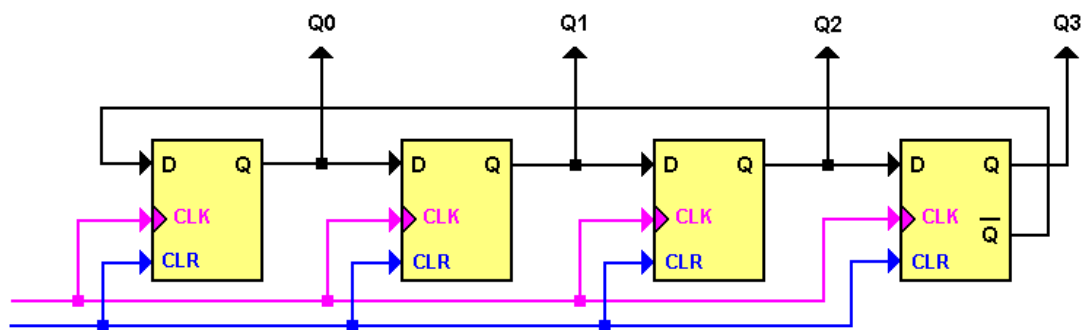


Ilustración 7 Contador Johnson de 4 bits

En la ilustración 7 se puede notar que el registro desplaza su contenido una vez a la izquierda con cada flanco de reloj y al mismo tiempo el complemento de Q_3 se transfiere a Q_0 . El contador comienza en cero, en cada flanco de reloj se realiza un corrimiento a la izquierda insertando 1's en Q_0 hasta que el registro queda lleno de 1's. A continuación se insertan 0's en Q_0 hasta que el registro queda lleno de 0's. El funcionamiento del contador de anillo se muestra en el diagrama de tiempos de la ilustración 8.

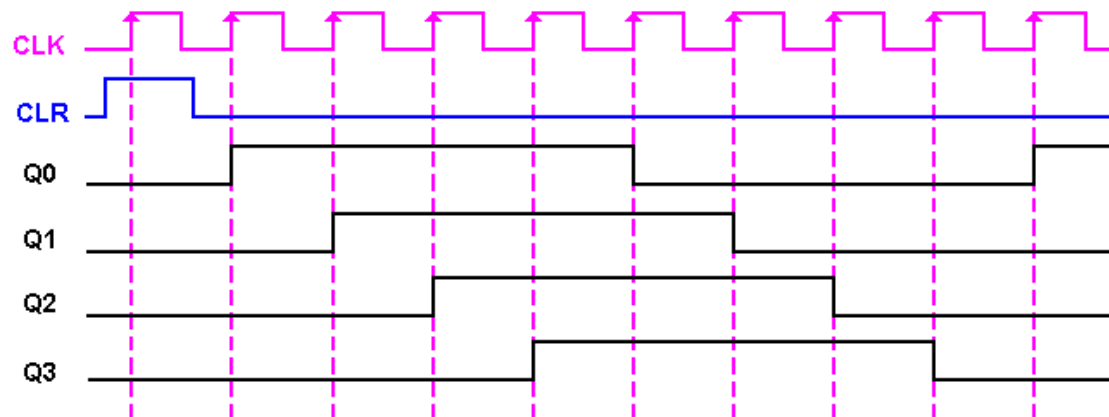


Ilustración 8 Diagrama de tiempo del contador Johnson

Para programar este contador usando VHDL se pueden usar muchas estrategias una de ellas consiste en usar el operador SLL como se muestra en la ilustración 9.

```

1  LIBRARY IEEE;
2  USE IEEE.STD_LOGIC_1164.all;
3
4  ENTITY JOHNSON IS
5      PORT (
6          CLR, CLK: IN STD_LOGIC;
7          Q : INOUT STD_LOGIC_VECTOR(3 DOWNTO 0)
8      );
9  END JOHNSON;
10
11 ARCHITECTURE CONTADOR OF JOHNSON IS
12 BEGIN
13     PJOHNSON : PROCESS(CLK, CLR)
14     BEGIN
15         IF( CLR = '1' ) THEN
16             Q <= (OTHERS => '0');
17         ELSIF( CLK'EVENT AND CLK = '1' ) THEN
18             Q(0) <= NOT(Q(3));
19             Q <= TO_STDLOGICVECTOR( TO_BITVECTOR( Q ) SLL 1 );
20         END IF;
21     END PROCESS PJOHNSON;
22 END CONTADOR;

```

Ilustración 9 Programa en VHDL del contador Johnson de 4 bits

Decodificación de un contador.

Los contadores se utilizan frecuentemente en aplicaciones donde debe determinarse o exhibirse de alguna manera el conteo representado por los estados de los FF's. Una forma de hacerlo es decodificando.

Analicemos los contadores de 4 bits de la tabla 1.

Estados	Contador secuencial MOD - 16				Contador Johnson MOD - 8				Contador de Anillo MOD - 4			
	Q_3	Q_2	Q_1	Q_0	Q_3	Q_2	Q_1	Q_0	Q_3	Q_2	Q_1	Q_0
q_0	0	0	0	0	0	0	0	0	0	0	0	1
q_1	0	0	0	1	0	0	0	1	0	0	1	0
q_2	0	0	1	0	0	0	1	1	0	1	0	0
q_3	0	0	1	1	0	1	1	1	1	0	0	0
q_4	0	1	0	0	1	1	1	1				
q_5	0	1	0	1	1	1	1	0				
q_6	0	1	1	0	1	1	0	0				
q_7	0	1	1	1	1	0	0	0				
q_8	1	0	0	0								
q_9	1	0	0	1								
q_{10}	1	0	1	0								
q_{11}	1	0	1	1								
q_{12}	1	1	0	0								
q_{13}	1	1	0	1								
q_{14}	1	1	1	0								
q_{15}	1	1	1	1								

Tabla 1 Contadores

Si se tiene un contador $MOD - X$, este tiene X diferentes estados. En los contadores de tabla 1 el contador secuencial es $MOD - 16$, el contador Johnson es $MOD - 8$ y el contador de anillo es $MOD - 4$. Para el caso de que los contadores sean de "n" bits el número de estados que se pueden generar se muestran en la tabla 2.

	Contador secuencial	Contador Johnson	Contador de Anillo
Estados	$MOD - 2^n$	$MOD - 2n$	$MOD - n$
Entradas en compuerta AND	n	2	0 (no se requiere lógica para decodificar)

Tabla 2 Estados de los contadores

Para poder identificar cada estado del contador se debe colocar un circuito decodificador que tenga X diferentes salidas, cada una detecta la presencia de un estado del contador. La decodificación de cada estado en el circuito decodificador se logra con una compuerta AND.

Con el contador secuencial se pueden manejar hasta 16 diferentes estados, sin embargo la lógica para decodificar necesita una compuerta AND de 4 entradas por estado. Para un contador de n bits se necesitan compuertas AND con " n " entradas. Con este contador se obtiene la mayor cantidad de estados, pero necesitan mucha lógica para la decodificación.

Con el contador Johnson se pueden manejar hasta 8 diferentes estados, sin embargo la lógica para decodificar necesita una compuerta AND de 2 entradas por estado. Para un contador de n bits se necesitan compuertas AND con solo 2 entradas. Estos contadores representan un punto intermedio entre los contadores de anillo y los contadores secuenciales. Un contador Johnson requiere menos FF's que un contador de anillo pero más que un contador secuencial. Tiene más lógica de decodificación que un contador de anillo, pero menos que un contador secuencial.

Con el contador de anillo se pueden manejar hasta 4 diferentes estados, pero en este contador no se requiere ninguna lógica para decodificar. En este contador a pesar de que se necesitan más FF's que en un contador secuencial para el mismo número MOD, este contador puede ser decodificado sin necesidad de usar lógica adicional.