

INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE COMPUTO

DISEÑO DE SISTEMAS DIGITALES

Práctica 3. CONTADOR JOHNSON Y DE ANILLO .

**PROFESOR:**

Testa Nava Alexis

**Alumno:**

Ramírez Cotonieto Luis Fernando

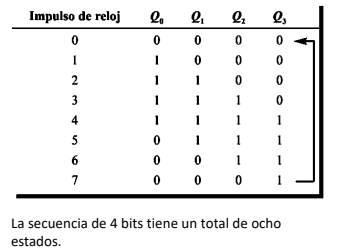
**GRUPO:**

2CV18

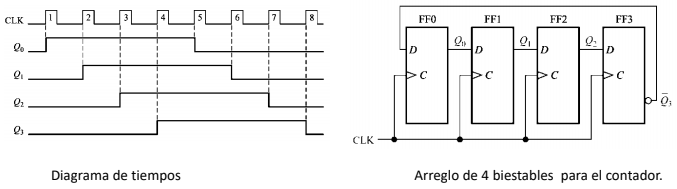
**FECHA:** 07/Mayo/2021

PRÁCTICA 3. CONTADOR JOHNSON Y DE ANILLO.

* Diseñar e implementar con HDL un Contador de Anillo y Contador Johnson de 8 bits.
* Incluir un selector de 1 bit. Cuando sea 0 mostrar contador Johnson cuando sea 1 contador de anillo. Incluir una entrada asíncrona de borrado para reiniciar el contador.
* En un contador Johnson, el complemento de la salida del último flip-flop se conecta a la entrada D del primero (también se puede implementar con otros tipos de flip-flop). Esta realimentación permite generar una secuencia de estados característica.

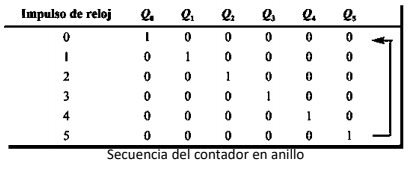


**Contador Johnson de cuatro bits**



**Contador en anillo**

El funcionamiento del contador de anillo es similar al del contador Johnson pero en este caso no se necesita lógica combinacional a la salida del ultimo flip-flop, se realimenta directamente como se muestra en la figura de tal forma que los datos se desplazan en forma de anillo.



• Utiliza un flip-flop para cada estado de su secuencia.

• Tiene la ventaja de que no se requieren compuertas de decodificación.

• En el caso de un contador en anillo de 6 bits, hay una única salida para cada dígito decimal.

Código

library ieee;

use ieee.std\_logic\_1164.all;

entity Practica3 is port(

clk : in STD\_LOGIC;

clear : in STD\_LOGIC;

sel : in STD\_LOGIC;

Q : out STD\_LOGIC\_VECTOR (7 downto 0)

);

end entity;

architecture A\_Practica3 of Practica3 is

signal TMP:STD\_LOGIC:='0';

signal QK,QT1,QT2,QD,T:STD\_LOGIC:='0';

signal q\_tmp: std\_logic\_vector(7 downto 0):= "00000000";

begin

TMP<='0' when(clear='1')else

'1' when(clk'event and clk='1');

process(sel,TMP,clear,q\_tmp)

begin

if(clear='1')then

q\_tmp<="00000000";

elsif(TMP='1')then

if(sel='0')then

--CONTADOR JOHNSON

q\_tmp(1) <= q\_tmp(0);

q\_tmp(2) <= q\_tmp(1);

q\_tmp(3) <= q\_tmp(2);

q\_tmp(4) <= q\_tmp(3);

q\_tmp(5) <= q\_tmp(4);

q\_tmp(6) <= q\_tmp(5);

q\_tmp(7) <= q\_tmp(6);

q\_tmp(0) <= not(q\_tmp(7));

elsif(sel='1')then

--CONTADOR ANILLO

q\_tmp(1) <= q\_tmp(0);

q\_tmp(2) <= q\_tmp(1);

q\_tmp(3) <= q\_tmp(2);

q\_tmp(4) <= q\_tmp(3);

q\_tmp(5) <= q\_tmp(4);

q\_tmp(6) <= q\_tmp(5);

q\_tmp(7) <= q\_tmp(6);

q\_tmp(0) <= q\_tmp(7);

end if;

end if;

end process;

q<=q\_tmp;

end A\_Practica3;

La combinación de las entradas nos permite que al tener “1” en el “Sel”, podamos seleccionar el contador de anillo, y al tener “0”, el contador Johnson.

Simulación

