



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.

Impulso de reloj	Q_0	Q_1	Q_2	Q_3
0	0	0	0	0
1	1	0	0	0
2	1	1	0	0
3	1	1	1	0
4	1	1	1	1
5	0	1	1	1
6	0	0	1	1
7	0	0	0	1

La secuencia de 4 bits tiene un total de ocho estados.

Contador Johnson de cuatro bits

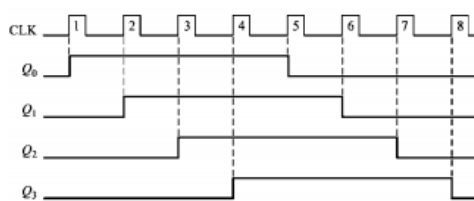
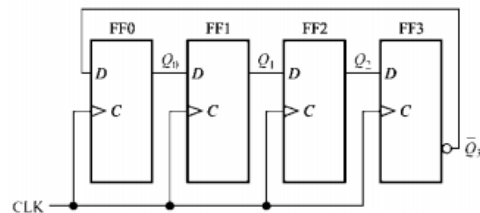


Diagrama de tiempos



Arreglo de 4 biestables para el contador.

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.

Impulso de reloj	Q_0	Q_1	Q_2	Q_3	Q_4	Q_5
0	1	0	0	0	0	0
1	0	1	0	0	0	0
2	0	0	1	0	0	0
3	0	0	0	1	0	0
4	0	0	0	0	1	0
5	0	0	0	0	0	1

Secuencia del contador en 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

