

INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE COMPUTO

DISEÑO DE SISTEMAS DIGITALES

Práctica 5. contadores y sus

Aplicaciones

**PROFESOR:**

Testa Nava Alexis

**Alumno:**

Ramírez Cotonieto Luis Fernando

**GRUPO:**

2CV18

**FECHA:**

12/Mayo/2021

Práctica 5. contadores y sus aplicaciones

Aplicar la metodología para diseño de circuitos secuenciales y obtener las ecuaciones usando FF-D y FF-JK que permitan implementar el diseño de un contador que debe mostrar el nombre del alumno. Este nombre debe tener más de 8 letras y menos de 16 letras.

Nombre:

**F E R N A N D O**

**1 2 3 4 5 6 7 8**

Flujo de estados

Tablas de verdad de los Flip-Flop a utilizar

|  |  |  |  |
| --- | --- | --- | --- |
| Q | Q+1 | J | K |
| 0 | 0 | 0 | X |
| 0 | 1 | 1 | X |
| 1 | 0 | X | 1 |
| 1 | 1 | X | 0 |

|  |  |  |
| --- | --- | --- |
| Q | Q+1 | D |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

Código de asignaciones

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Estado | Salida | Secuencial | Gray | One-hot | Definido por el usuario |
| Q0 | F | 0000 | 0000 | 0000000000000001 | 0000001 |
| Q1 | E | 0001 | 0001 | 0000000000000010 | 1001000 |
| Q2 | R | 0010 | 0011 | 0000000000000100 | 0001000 |
| Q3 | N | 0011 | 0010 | 0000000000001000 | 1111010 |
| Q4 | A | 0100 | 0110 | 0000000000010000 | 0111000 |
| Q5 | N | 0101 | 0111 | 0000000000100000 | 1001111" |
| Q6 | D | 0110 | 0101 | 0000000001000000 | 0110000 |
| Q7 | O | 0111 | 0100 | 0000000010000000 | 1000111 |

Código VHDL

library ieee;

use ieee.std\_logic\_1164.all;

entity Practica5\_NombreFernando is

port( clock, clear, enable: in std\_logic;

display: out std\_logic\_vector(6 downto 0) );

attribute pin\_numbers of Practica5\_NombreFernando: entity is

"clock:1 clear:13 enable:2"

& " display(0):15 display(1):16 display(2):17 display(3):18 display(4):19 display(5):20 display(6):21";

--& " display(0):21 display(1):20 display(2):19 display(3):18 display(4):17 display(5):16 display(6):15";

--display(0):g display(1):f display(2):e display(3):d display(4):c display(5):b display(6):a

end Practica5\_NombreFernando;

architecture A\_Practica5\_NombreFernando of Practica5\_NombreFernando is

--Etiquetas

constant Etiqueta0 : std\_logic\_vector(1 downto 0) := "00";

constant Etiqueta1 : std\_logic\_vector(1 downto 0) := "01";

constant Etiqueta2 : std\_logic\_vector(1 downto 0) := "10";

--Simbolos (letras)

constant Letra\_F : std\_logic\_vector(6 downto 0) := "0000001";

constant Letra\_E : std\_logic\_vector(6 downto 0) := "1001000";

constant Letra\_R : std\_logic\_vector(6 downto 0) := "0001000";

constant Letra\_N : std\_logic\_vector(6 downto 0) := "1111010";

constant Letra\_A : std\_logic\_vector(6 downto 0) := "0111000";

constant Letra\_N : std\_logic\_vector(6 downto 0) := "1001111";

constant Letra\_D : std\_logic\_vector(6 downto 0) := "0110000";

constant Letra\_O : std\_logic\_vector(6 downto 0) := "1000111";

--Estados

--Usando un código definido por el usuario el cual es el numero en cod de display 7 seg y es diferente para cada uno por lo que lo puedo usar asi.

constant Estado0 : std\_logic\_vector(8 downto 0) := Etiqueta0 & Letra\_F; --"000000001";

constant Estado1 : std\_logic\_vector(8 downto 0) := Etiqueta0 & Letra\_E; --"001001000";

constant Estado2 : std\_logic\_vector(8 downto 0) := Etiqueta0 & Letra\_R; --"000001000";

constant Estado3 : std\_logic\_vector(8 downto 0) := Etiqueta0 & Letra\_N; --"001111010";

constant Estado4 : std\_logic\_vector(8 downto 0) := Etiqueta0 & Letra\_A; --"000111000";

constant Estado5 : std\_logic\_vector(8 downto 0) := Etiqueta0 & Letra\_N; --"001001111";

constant Estado6 : std\_logic\_vector(8 downto 0) := Etiqueta1 & Letra\_D; --"010000001";

constant Estado7 : std\_logic\_vector(8 downto 0) := Etiqueta1 & Letra\_O; --"011111000";

--señal

--"salida" hace referencia al cambio de estado y la salida que tiene el estado

signal salida: std\_logic\_vector(8 downto 0);

begin

process(clock,clear)

begin

if(clear = '1') then

salida <= Estado0;

elsif(clock'event and clock = '1') then

if(enable = '1') then

case (salida) is

when Estado0 => salida <= Estado1;

when Estado1 => salida <= Estado2;

when Estado2 => salida <= Estado3;

when Estado3 => salida <= Estado4;

when Estado4 => salida <= Estado5;

when Estado5 => salida <= Estado6;

when Estado6 => salida <= Estado7;

when Estado7 => salida <= Estado0;

when others => salida <= Estado0;

end case;

end if;

end if;

end process;

display <= salida(6 downto 0);

end A\_Practica5\_NombreFernando;