

LABORATORIO DE DISEÑO DIGITAL MODERNO

Diseño y construcción de dos sistemas trabajando al mismo tiempo.

Práctica 12

Profesora: M.I. Norma Elva Chávez Rodríguez.

OBJETIVO:

El alumno aprenderá a diseñar utilizando código VHDL, dos sistemas que hacen muy distintas tareas, trabajando al mismo tiempo, dentro de un mismo FPGA.

ESPECIFICACIONES:

Diseño y construcción de 2 sistemas trabajando al mismo tiempo.

El sistema 1 requiere el diseño y construcción de dos contadores, que trabajen al mismo tiempo, uno que cuente del cero al nueve y vuelva a empezar y el otro que cuente del nueve al cero y vuelva a empezar, utilizando dos displays de 7 segmentos conectados en forma paralela. (Práctica 8.).

El sistema 2 es un control de semáforos en dos avenidas. Las restricciones de los semáforos son las siguientes: la luz roja debe durar 60 segundos, la luz verde 45 segundos y la luz amarilla 15 segundos. La luz luces en los semáforos norte y sur se prenderán y apagarán al mismo tiempo, La luz luces en los semáforos norte y sur se prenderán y apagarán al mismo tiempo.

DIAGRAMA DE BLOQUES:

LABORATORIO DE DISEÑO DIGITAL MODERNO

Diseño y construcción de dos sistemas trabajando al mismo tiempo.

Práctica 12

Profesora: M.I. Norma Elva Chávez Rodríguez.

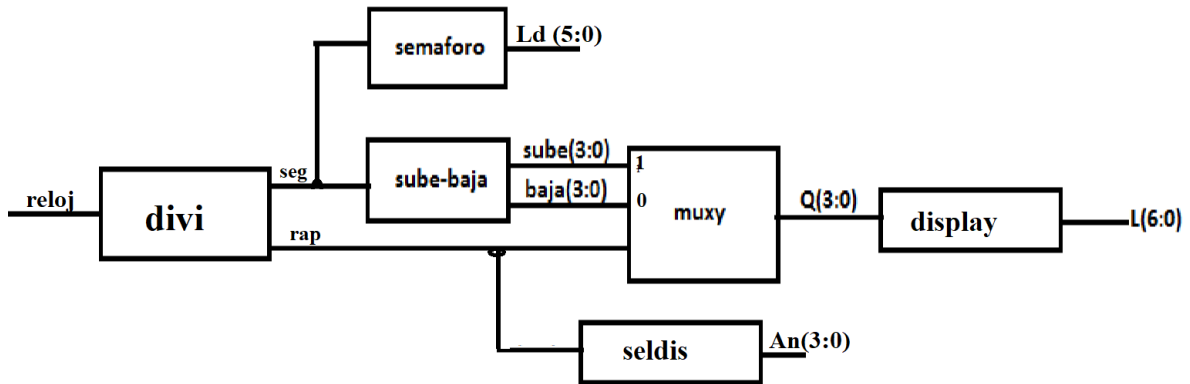
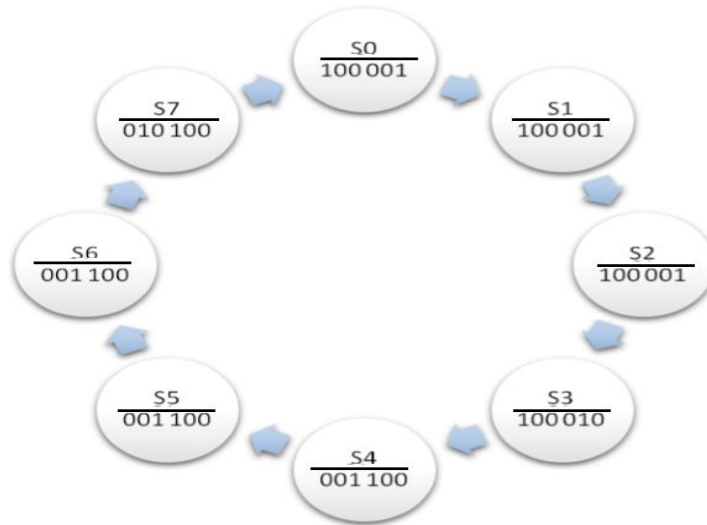


DIAGRAMA DE ESTADOS SISTEMA 2:



CÓDIGO EN LENGUAJE VHDL:

LABORATORIO DE DISEÑO DIGITAL MODERNO

Diseño y construcción de dos sistemas trabajando al mismo tiempo.

Práctica 12

Profesora: M.I. Norma Elva Chávez Rodríguez.

```
5 entity dosistemas is
6     Port (RELOJ : in STD_LOGIC;
7           An : out std_logic_vector (3 downto 0);
8           led : out std_logic_vector (5 downto 0);
9           L : out std_logic_vector (6 downto 0));
10 end dosistemas;
11 architecture Behavioral of dosistemas is
12     signal SEGUNDO : STD_LOGIC;
13     signal RAPIDO : STD_LOGIC;
14     signal SUBE: STD_LOGIC_VECTOR(3 downto 0);
15     signal BAJA: STD_LOGIC_VECTOR(3 downto 0);
16     signal Q: STD_LOGIC_VECTOR(3 downto 0);
17     type estados is (s0,s1,s2,s3,s4,s5,s6,s7);
18     signal epresente, esiguiente :ESTADOS;
19
20 begin
21
22     divisor : process (RELOJ)
23     variable CUENTA: STD_LOGIC_VECTOR(27 downto 0) := X"00000000";
24     begin
25         if rising_edge (RELOJ) then
26             if CUENTA=X"48009E0" then
27                 cuenta := X"00000000";
28             else
29                 cuenta := cuenta+1;
30             end if;
31         end if;
32     segundo <=CUENTA(24);
33     rapido <=CUENTA(10);
34     end process;
35
36     contador: process (segundo)
37     variable ARRIBA: STD_LOGIC_VECTOR(3 downto 0) := "0000";
```