



Universidad Nacional  
Autónoma de México  
Facultad de Ingeniería



División de Ingeniería Eléctrica

Alumno: Fernando Iván Silva Jiménez

Profesora: M.I. Norma Elva ChávezRodríguez

Laboratorio de Diseño de Sistemas Digitales

Practica 10

(Diseño y construcción de una carta ASM)

Semestre 2014-1

Fecha de realización: 16/octubre/2013

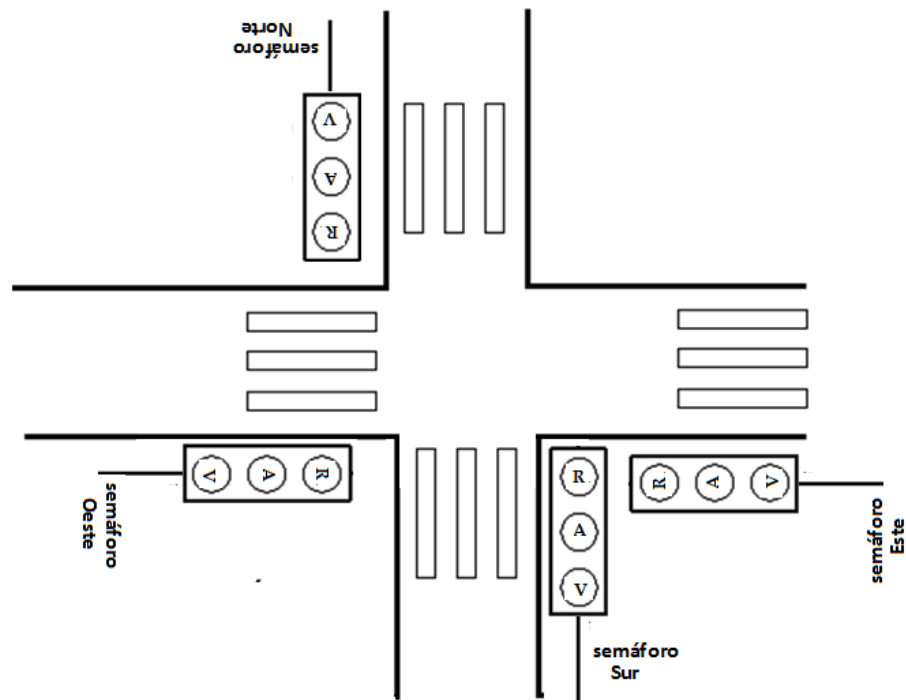
Fecha de entrega: 20/octubre/2013

**OBJETIVO:**

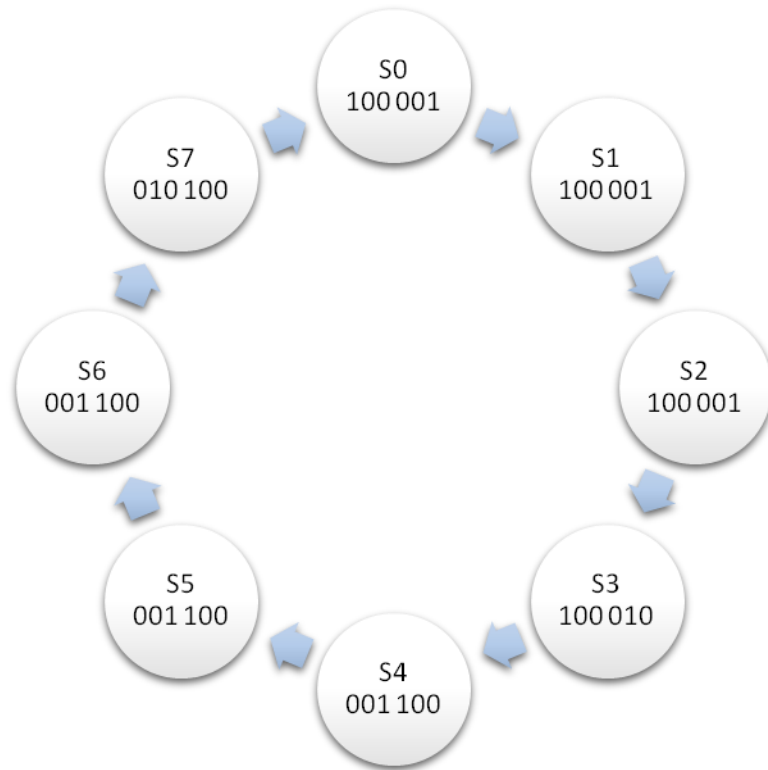
El alumno aprenderá a diseñar utilizando código VHDL, una carta ASM, además de aprender a controlar 2 sistemas que hacen muy distintas tareas al mismo tiempo.

**ESPECIFICACIONES:**

Diseño y construcción de 2 sistemas trabajando al mismo tiempo: el sistema 1 será un contador ascendente y el otro descendente (realizado en la practica 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 verde en los semáforos norte y sur se prenderán y apagarán al mismo tiempo, La luz roja en los semáforos norte y sur se prenderán y apagarán al mismo tiempo.



## DIAGRAMA DE ESTADOS DE LOS SEMÁFOROS



### Explicación del diagrama de estados

- S0 es el primer estado en el cual el semáforo 1 la luz verde está encendida y en el semáforo 2 la luz roja está encendida.
- S1 es el segundo estado en el cual el semáforo 1 la luz verde sigue encendida, porque solo han pasado 15 segundos y debe durar 45 segundos, y en el semáforo 2 la luz roja permanece encendida pues debe de durar 60 segundos.
- S2 es el tercer estado en el cual el semáforo 1 la luz verde sigue encendida ya que han pasado 30 segundos de los 45 que debe de durar, y en el semáforo 2 la luz roja está encendida.
- S3 es el cuarto estado en el cual el semáforo 1 la luz verde se ha apagado y la luz amarilla enciende ya que han pasado 45 segundos, y en el semáforo 2 la roja aun sigue encendida.
- S4 es el quinto estado en el cual el semáforo 1 la luz roja se enciende y en el semáforo 2 la luz verde también está encendida.

- S5 es el sexto estado en el cual el semáforo 1 la luz roja sigue encendida, porque solo han pasado 15 segundos y debe durar 60 segundos, y en el semáforo 2 la luz verde permanece encendida pues debe de durar 45 segundos.
- S6 es el séptimo estado en el cual el semáforo 1 la luz roja sigue encendida ya que han pasado 30 segundos de los 60 segundos que debe de durar, y en el semáforo 2 la luz verde está encendida.
- S7 es el octavo estado en el cual el semáforo 1 la luz roja sigue encendida, y en el semáforo 2 la luz verde se ha apagado y la luz amarilla enciende ya que han pasado 45 segundos. Y el estado siguiente es el S0 ya que se vuelve a repetir el ciclo.

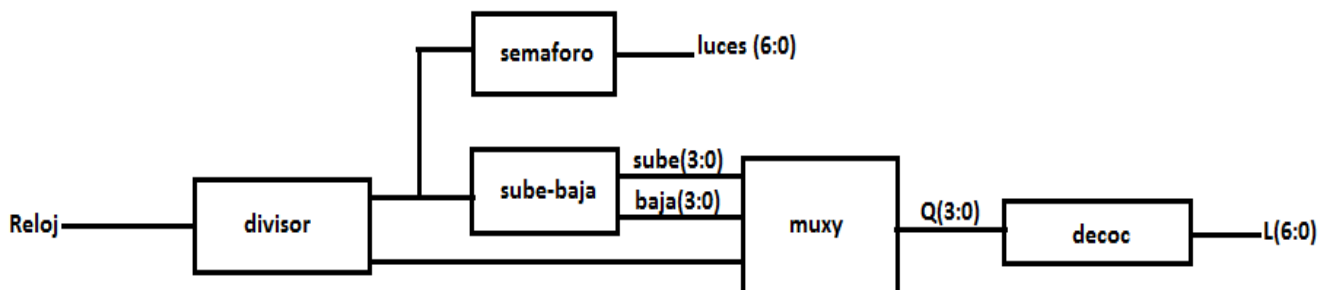
### DIAGRAMA DE BLOQUES:

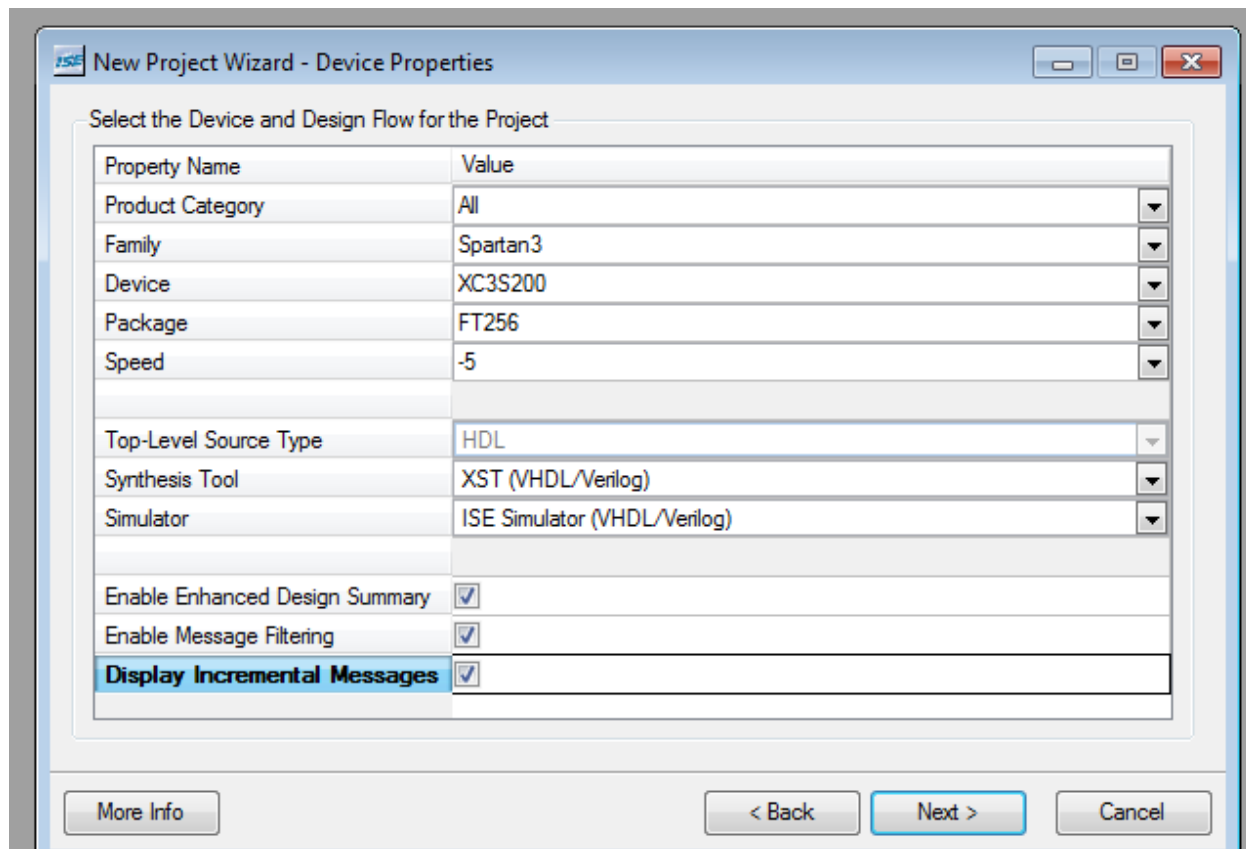
### PROCEDIMIENTO

Para el diseño vamos a hacer uso del de la plataforma ISE de Xilinx y la tarjeta SPARTAN 3. Vamos a crear una carpeta llamada “semáforo”, en la cual se estará trabajando para este proyecto.

Abrimos la plataforma ISE y nos dirigimos a crear un nuevo proyecto. Lo guardaremos con el nombre “semáforo” en la misma carpeta que se creó previamente.

Usamos la siguiente configuración para el correcto funcionamiento:





Para diseñar el semáforo vamos a crear una nueva fuente con el nombre de "semaforo" y elegimos "VHDL Module" después creamos una entrada con el nombre de "divi" y una salida llamada "luces" la cual será un vector de 6 elementos.

```
4 library IEEE;
5 use IEEE.STD_LOGIC_1164.ALL;
6 use IEEE.STD_LOGIC_ARITH.ALL;
7 use IEEE.STD_LOGIC_UNSIGNED.ALL;
8 |
9 entity semaforo is
10     Port ( divi : in  STD_LOGIC;
11           luces : out STD_LOGIC_VECTOR (5 downto 0));
12 end semaforo;
13
```

```
14 architecture Behavioral of semaforo is
15
16 type estados is (s0,s1,s2,s3,s4,s5,s6,s7);--definicion de los estados a usar
17 signal epresente, esiguiente: estados;--definicion de las señales para interconectar procesos
18 begin
19 process (divi)--lista de sensibilidad dada por epresente
20   begin
21     if rising_edge(divi) then
22       epresente <=esiguiente;
23     end if;
24   end process;
25   process(epresente)
26     begin
27       case epresente is
28         when s0=>
29           luces<="100001";
30           esiguiente<=s1;
31         when s1=>
32           luces<="100001";
33           esiguiente<=s2;
34         when s2=>
35           luces<="100001";
36           esiguiente<=s3;
37         when s3=>
38           luces<="100010";
39           esiguiente<=s4;
40         when s4=>
41           luces<="001100";
42           esiguiente<=s5;
43         when s5=>
44           luces<="001100";
45           esiguiente<=s6;
46         when s6=>
47           luces<="001100";
48           esiguiente<=s7;
49         when s7=>
50           luces<="010100";
51           esiguiente<=s0;
52         when others=>
53           luces<="010100";
54         end case;
55     end process;
56 end Behavioral;
57
58
```

El código para que el semáforo funcione es el siguiente:

Después agregamos una nueva fuente en donde vamos a agregar nuestros códigos que ya teníamos previamente hechos (decodi.vhd, MUX.vhd, divisor.vhd, sube\_baja.vhd).

Ya que se hayan agregado los archivos creamos el esquemático de cada uno para poder crear nuestro diseño.

Creamos una nueva fuente la cual va a ser "schematic" el cual vamos a llamar "TOP", después con ayuda del filtro de nombre de los símbolos agregamos cada elemento del cual creamos nuestros esquemáticos.

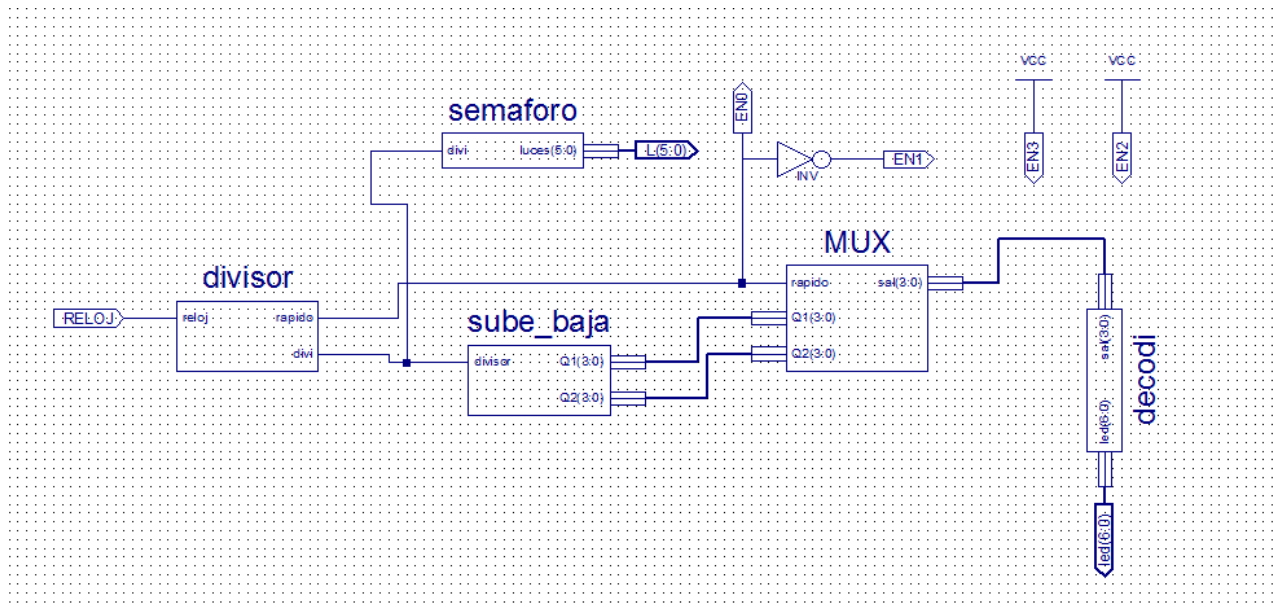


DIAGRAMA ESQUEMATICO.

### CONCLUSION

Esta fue una práctica en la cual pude hacer dos controladores los cuales hicimos funcionar al mismo tiempo, y a parte aprendí a diseñar un semáforo con máquinas de estado. Creo el objetivo se cumplió y reafirme mis conocimientos obtenidos en la clase de teoría.