



**Universidad Nacional
Autónoma de México
Facultad de Ingeniería
División de Ingeniería Eléctrica**

Alumnos:

Chávez Ordoñez César Alejandro

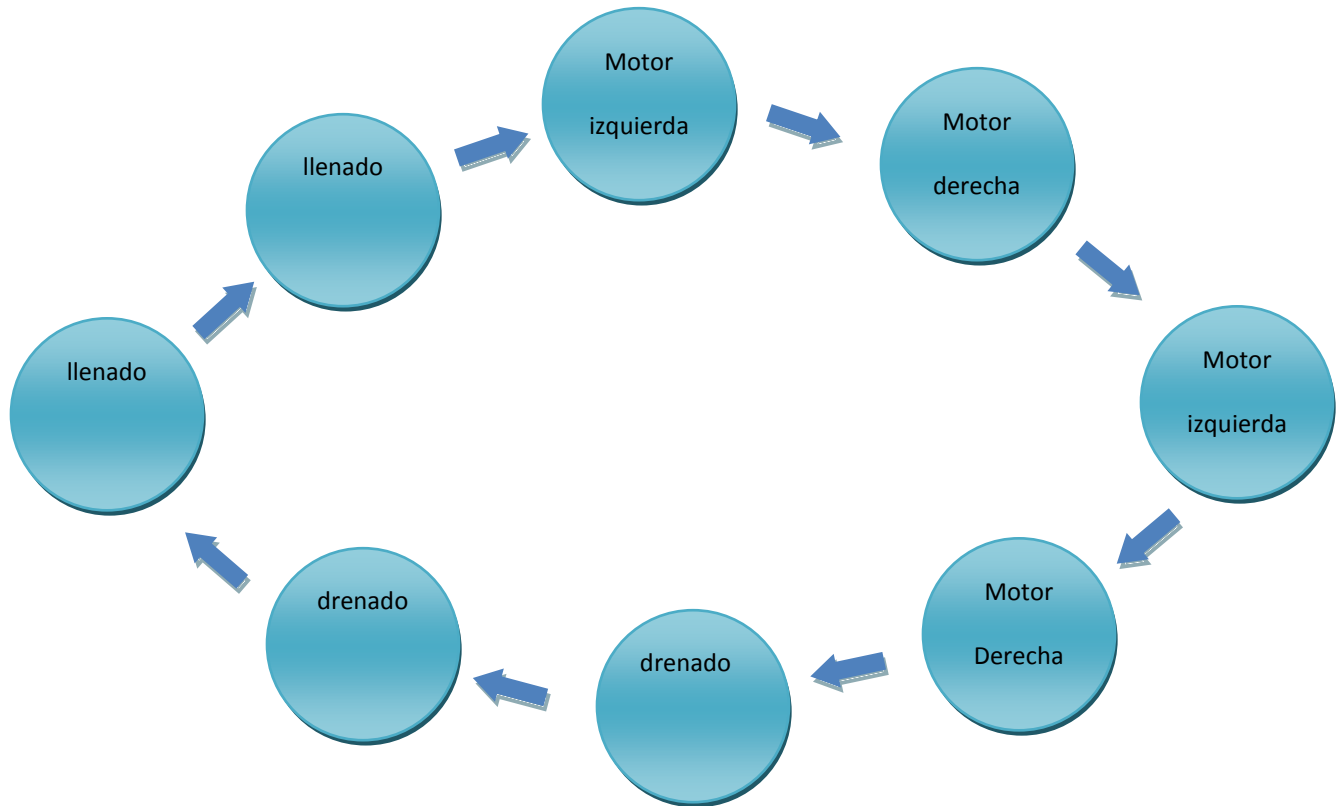
Guarduño Aguilar Natanael Jesua

**Profesora: M.I. Norma Elva Chávez
Rodríguez**

**Laboratorio de Diseño de Sistemas
Digitales**

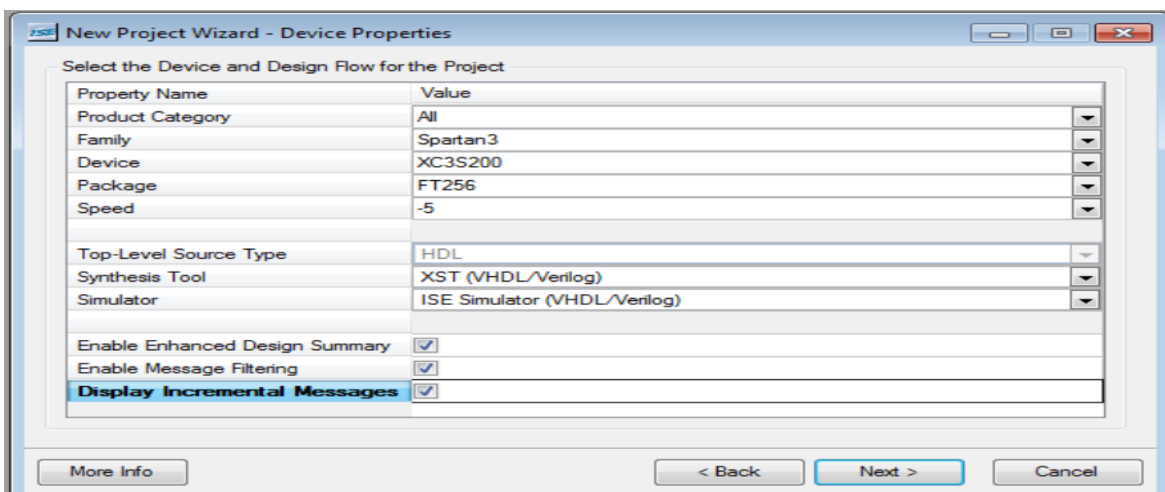
Diagrama de secuencias:

En esta práctica se han establecido 8 estados los cuales serán los que hay que implementar y son de la siguiente forma:

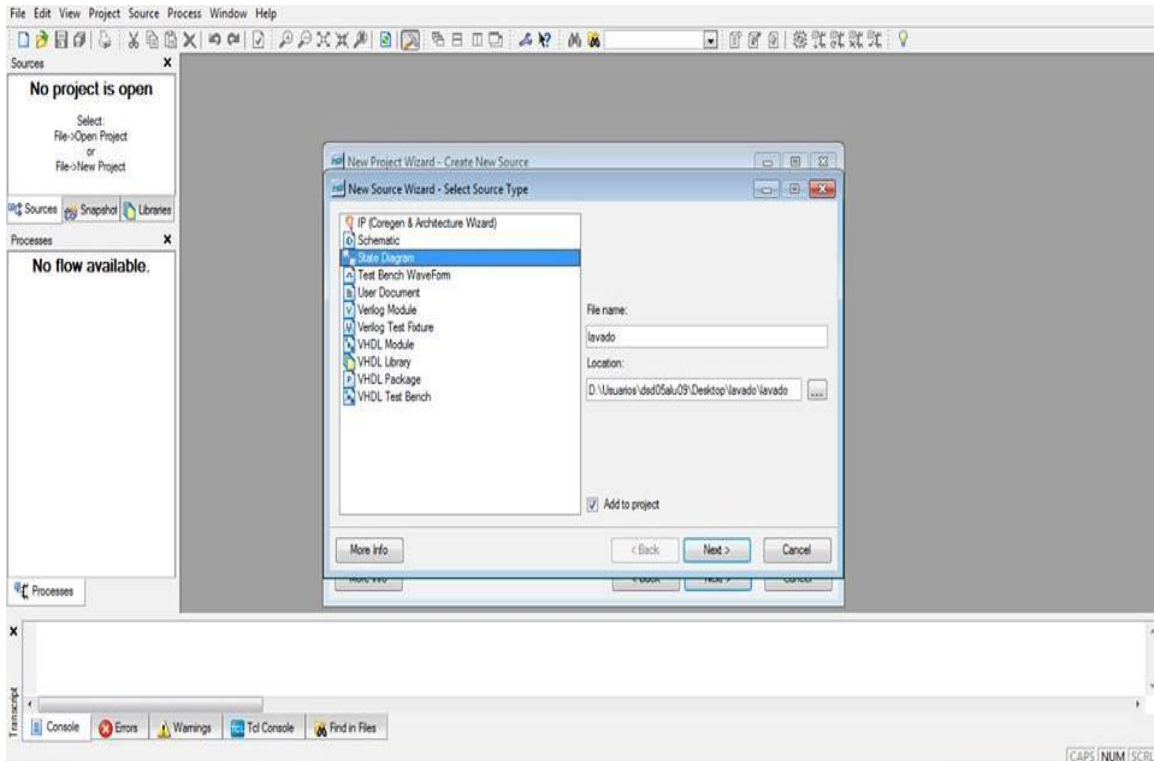


Procedimiento:

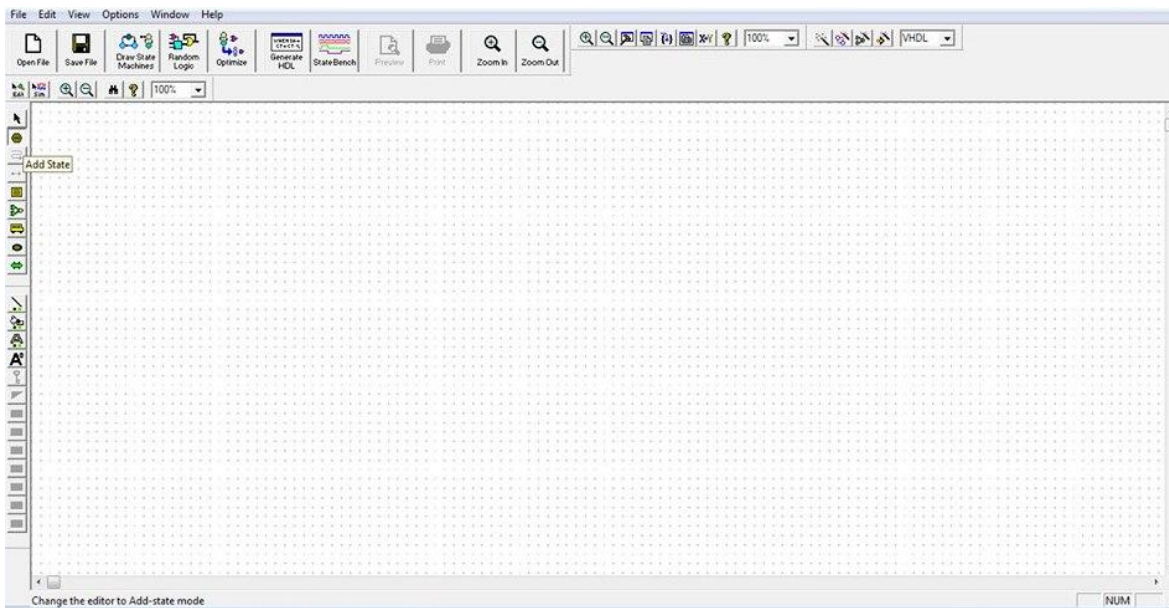
Iniciamos creando un nuevo proyecto y colocando los datos de la tarjeta:



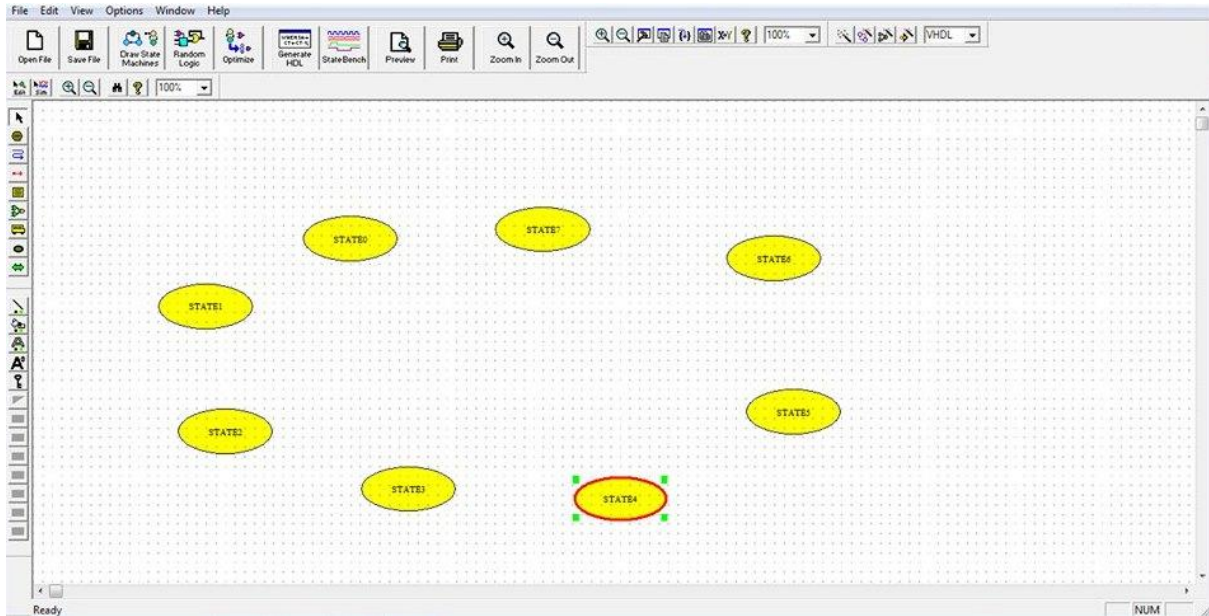
Se crea un nuevo proyecto con el tipo de “diagrama de estado” donde se desplegará una ventana para insertar los estados que se requieran:



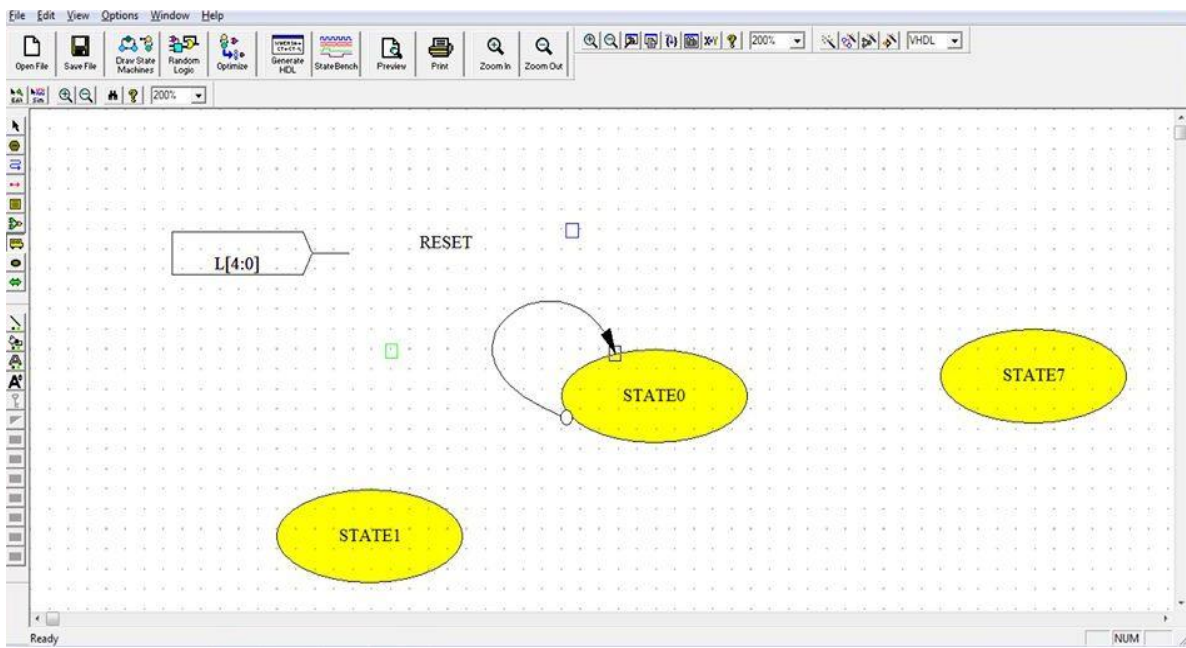
Esta ventana es la ventana principal de la herramienta y en esta se trabajará con todo lo que tenga que ver con el diagrama de estados:



Ahora se ponen los estados que se necesitan (segundo botón del menú de la izquierda) en este caso se anotan 8 estados:

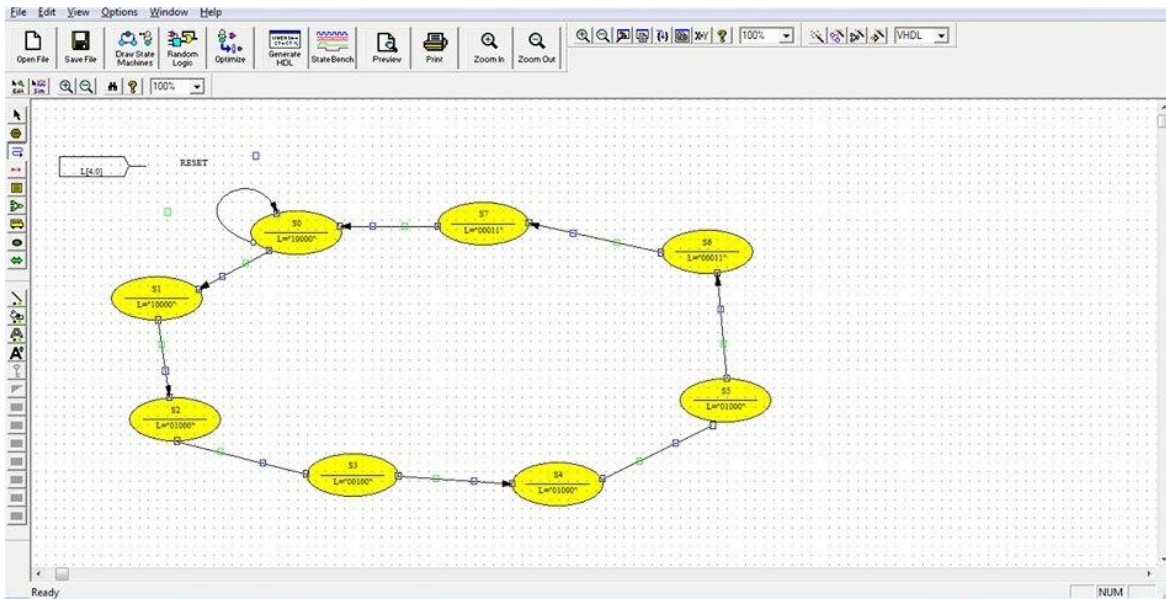


Se crea un “reset” en el estado 0 (con la cuarta herramienta de arriba abajo de la cinta de la izquierda) y también un bus de salida (con la séptima herramienta de arriba a abajo de la cinta de la izquierda) para que este atienda a los estados y cada una de sus salidas:

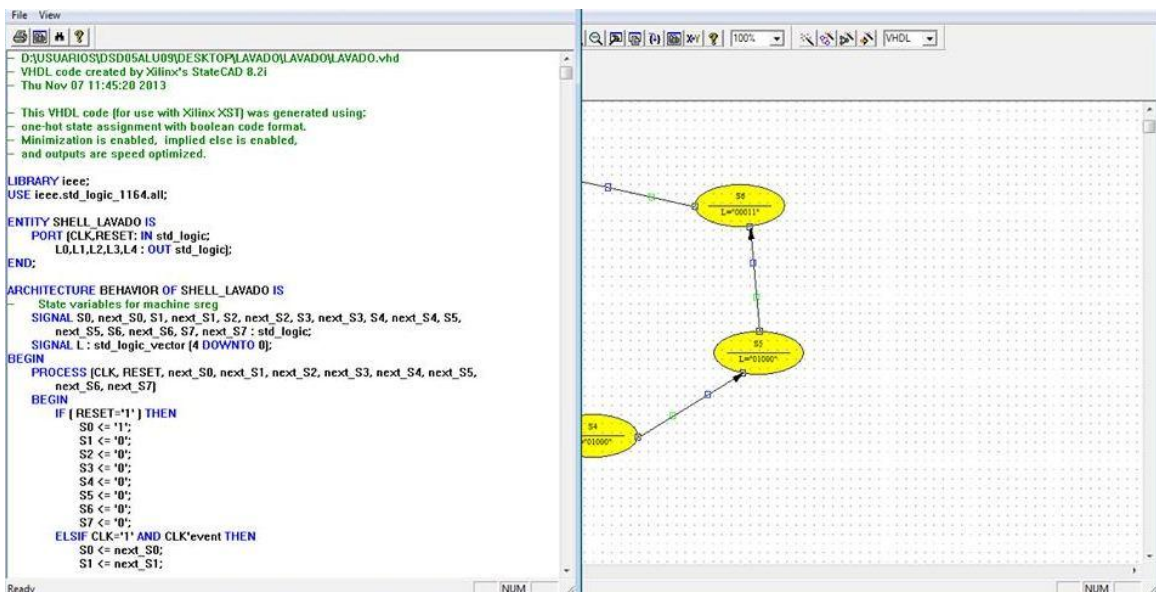


Dando doble clic a cada estado se accede a sus propiedades estas la modificaremos pero solo modificaremos el nombre de cada estado y la salida de cada uno de estos así como se verá en las imágenes siguientes.

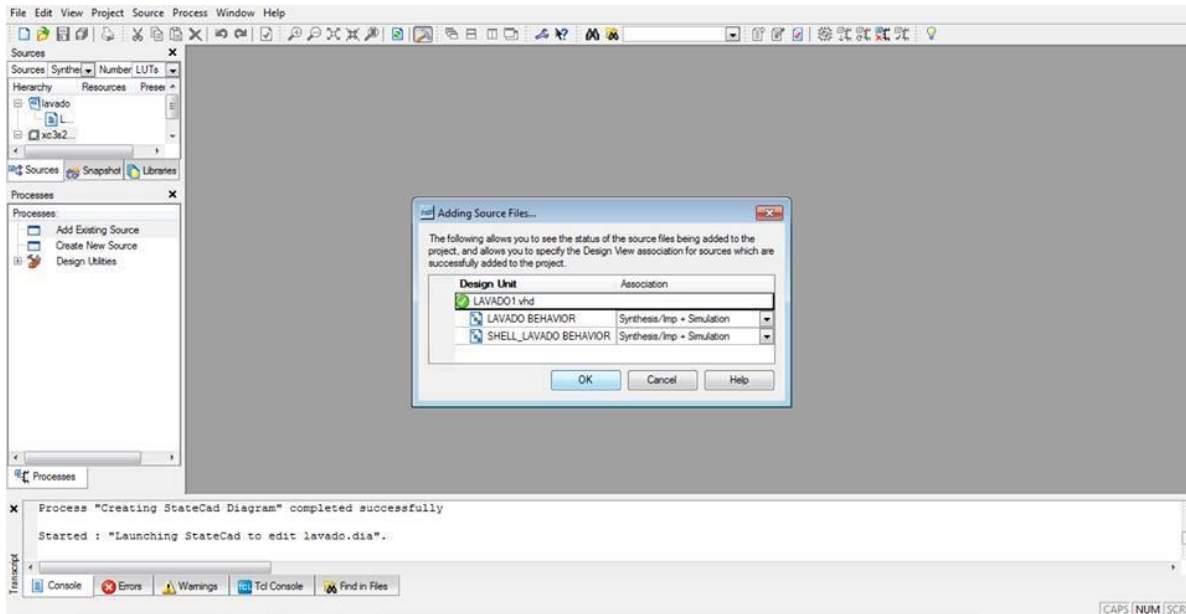
Se unen todos los estados con la tercer herramienta de la cinta de la izquierda (flecha azul) respetando la dirección de ellos para crear una secuencia de cambios de estados:



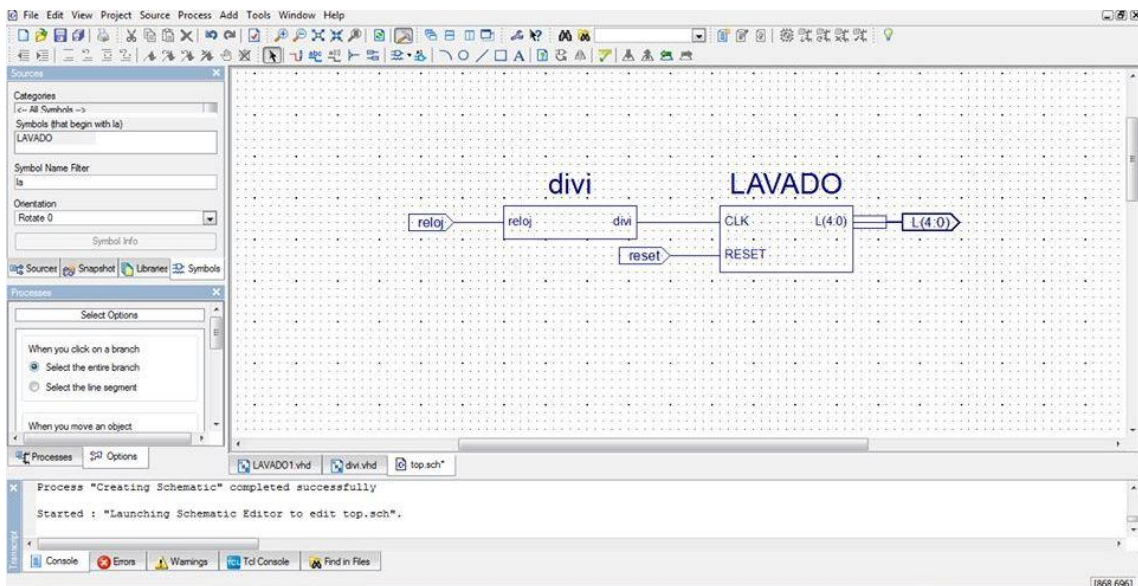
Dentro de Xilinx se tiene la opción de poder convertir el diagrama de estados a lenguaje VHDL, la cual se encuentra en el menú superior de iconos grandes de la ventana (quinto botón de izquierda a derecha) y al hacer clic se despliega la siguiente ventana:



Se añaden fuentes existentes que es lo que se creó como “lavadora” (este es el archivo generado por Xilinx ISE) y “divisor” (esta se obtiene del divisor de frecuencias hecho en clase), se checa sintaxis y se crea el símbolo de ambas.



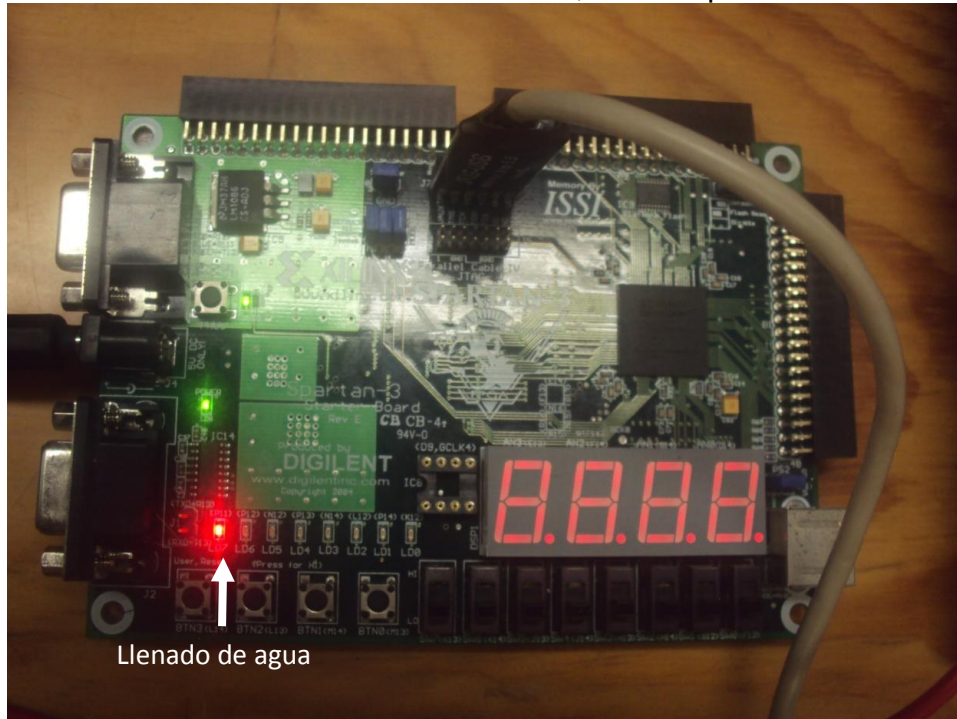
Se crea el esquemático llamado top (el nombre es indistinto), se insertan los símbolos, se definen las entradas y salidas.



Por último se implementa el archivo tipo esquemático llamado “top” y sus respectivas salidas en la tarjeta (datos de localización de leds y push botón de la tarjeta) para generar el archivo .bit para que sea programado en esta misma con la herramienta Adept que se debe encontrar instalada en nuestra PC.

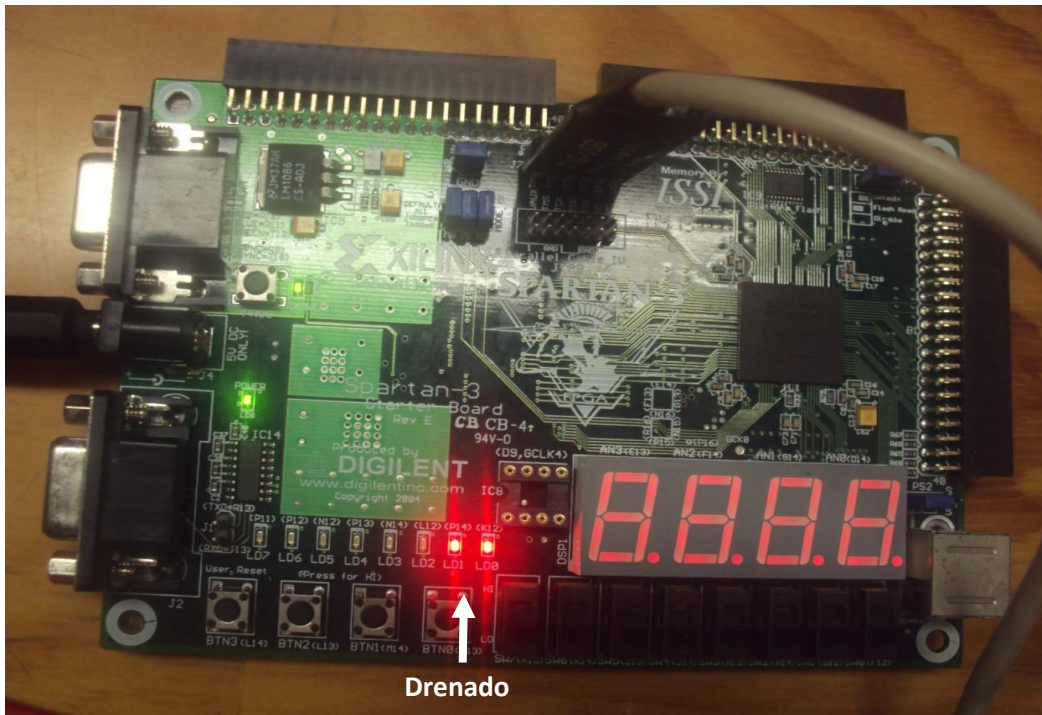
Resultados:

Al final de la implementación se obtiene el siguiente resultado en la tarjeta y se comprueba que nuestro diagrama de estados fue implementado con éxito. He aquí fotos del funcionamiento de la lavadora, viendo que si cambia de estado:





Aspa derecha



Drenado

Conclusiones:

Esta práctica de verdad mostró la forma de manejar bien las herramientas de Xilinx ISE ya que esta herramienta nos ayuda si aún no dominamos bien algún lenguaje HDL, a pasar un diagrama de estados sencillo a código vhdl por lo tanto es de suma importancia para comprender y obtener una ventaja de esta herramienta de desarrollo de sistemas digitales. Se debe hacer notar que **el mismo diagrama de estados en lenguaje VHDL ocupa menos espacio en el FPGA** que el código que utiliza esta herramienta.

Observaciones:

Se modificó el código de divisor para esta práctica puesto que se colocó una velocidad menor a las veces anteriores a esta y por lo tanto se forzó a que cada estado durará más y también se comentó en este código la salida rápida puesto que esa salida no era de importancia ya que no utilizaríamos los displays.

