

INFORME DE LABORATORIO

Autores: Emmanuel Garay Rivera, Kevin Jimenez Rincón

Laboratorio de Electrónica Digital 2 Departamento de Ingeniería Electrónica y de Telecomunicaciones Universidad de Antioquia

Resumen

Esta práctica se enfoca en el diseño e implementación de una Máquina de Estados Finitos (FSM) utilizando el lenguaje de descripción de hardware SystemVerilog y su herramienta de simulación Modelsim Altera. La FSM tiene como objetivo generar una secuencia de números y mostrar su estado actual en dos displays de 7 segmentos. Se tiene como consideración que el sistema debe poder manejar la secuencia de manera ascendente o descendente según se quiera.

Palabras clave: Display, FSM, Señal.

Introducción

En el mundo de los dispositivos electrónicos, estos se comunicación mediante pulsos eléctricos, los cuales solo se pueden representar en dos estados (1 y 0). Nosotros como seres humanos no dominamos este lenguaje, por lo que intentamos traducirlo, en el caso de esta práctica mediante un display.

En esta práctica se lleva a cabo el diseño y la implementación de una máquina de estados finitos, para la generación de una secuencia de números establecida, teniendo presente algunas consideraciones adicionales, como el desplazamiento de los estados de forma ascendente y descendente, entre otros. empleando el lenguaje de descripción de hardware SystemVerilog y su herramienta de simulación.

Marco teórico

Las máquinas de estado y los display de siete segmentos son herramientas muy utilizadas en la electrónica digital. Estos elementos desempeñan un papel crucial en la implementación y visualización de información en sistemas digitales complejos.

Procedimiento experimental y resultados

Esta FSM que se usa para controlar la secuencia dada (3, 6, 9, 12, 15, 2, 5, 7) y mostrar su estado actual en dos displays de 7 segmentos. Posee dos divisores de frecuencia que generan señales de reloj, una cada 1 segundo y otra cada 0.5 segundos. La señal "Time" permite seleccionar cuál de estos divisores se utilizará como entrada para la FSM. Además, la FSM recibe la señal "Up", que determina si la secuencia debe avanzar de manera ascendente o descendente. La FSM utiliza estas señales para cambiar de estado en cada flanco de subida de la señal de reloj. Finalmente, la FSM envía el estado actual a los dos displays de 7 segmentos, lo que permite visualizar en qué punto de la secuencia nos encontramos, el diagrama de bloques de este diseño se evidencia en la figura 1.

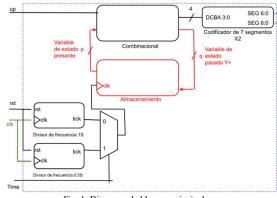


Fig. 1. Diagrama de bloques principal

Para verificar el correcto funcionamiento de la máquina de estados finitos primero se hizo un bosquejo de su funcionamiento (Fig. 2) y luego, usando la herramienta ModelSim Altera del programa Quartus se visualizó una simulación del diseño en la que se puede evidenciar la secuencia dada es correcta y a su vez que con la señal "Time" en 1 se reproduce de forma ascendente y en 0 de forma descendente.

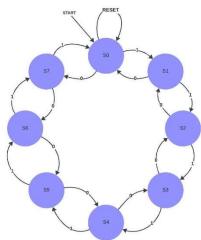


Fig. 1. Funcionamiento esperado de la FSM

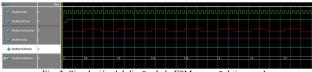


Fig. 3. Simulación del diseño de la FSM con señal time en 1



Fig. 4. Simulación del diseño de la FSM con señal time en 0

Conclusiones

Al diseñar una Máquina de Estados Finitos (FSM), hay que pensar en esta como si se estuviera diseñando una serie de pasos específicos para que una máquina o sistema los siga de manera ordenada y controlada. Es como crear un guion o un conjunto de reglas que le dice a la máquina qué hacer en cada situación.

Este diseño cuidadoso de estados y transiciones en una FSM es esencial para garantizar que la máquina o sistema siga el comportamiento deseado, sin importar lo complejo que sea el proceso. Ayuda a que todo funcione de manera ordenada y predecible, permitiendo que la máquina "sepa" en qué etapa se encuentra y qué acciones tomar en consecuencia logrando así un buen funcionamiento tal como se demuestra al lograr la secuencia dada.

Si bien el resultado es el esperado y la maquina funciona correctamente quizá podría ser mas optima al usarse un solo divisor de frecuencia y un array que almacene todos los datos de la secuencia, esto traería beneficios tales como la eficiencia misma y la facilidad de cambiar a distingas secuencias sin cambiar de manera brusca o significativa el hardware y a su vez permitiría agregar más estados de ser necesario, pero se ve expuesto a que su complejidad sea mayor y según el tamaño del array debería considerarse si el consumo de memoria valdría la pena, por lo cual la decisión de usar uno u otro modo viene dada por los requerimientos y gustos del usuario