

Laboratorio de Electrónica Digital II

Práctica No. 1 - Warm Up

Generador de Secuencia de Números

Profesores

Luis Fernando Castaño L. (luis.castanol@udea.edu.co)

Luis Germán García M. (german.garcia@udea.edu.co)

Agosto 22, 2023



Fecha de entrega: agosto 29 a septiembre 1 de 2023

Medio de entrega: <https://virtualingenieriaudea.co/>

Sustentación: Horario de Laboratorio

Valor Práctica: 6% del curso

1 Introducción

En esta práctica de laboratorio, el grupo de estudiantes diseñará un sistema electrónico digital basado en FPGA, para la generación de una secuencia de números a visualizarse en displays de 7-segmentos. El núcleo del sistema será una máquina de estados finitos (FSM) que implementará determinada secuencia de números. La descripción de la FSM junto con la lógica requerida será realizada empleando el lenguaje de descripción de hardware SystemVerilog. Se deberá emplear la herramienta de simulación ModelSIM para llevar a cabo una serie de pruebas que permitan verificar

el correcto funcionamiento del secuenciador. Finalmente, el grupo de trabajo implementará el diseño en el sistema de desarrollo DE10-Lite, donde se deberá verificar el correcto funcionamiento de todo el sistema.

2 Objetivo de la Práctica

Diseñar e implementar una máquina de estados finitos junto con la lógica adicional requerida, para la generación de una secuencia de números, empleando el lenguaje de descripción de hardware SystemVerilog y herramientas de desarrollo para FPGAs.

3 Procedimiento

Para el correcto diseño e implementación de la práctica, es necesario leer completamente esta guía. Se sugiere seguir el procedimiento indicado a continuación:

- a. Analizar la secuencia de números que debe generar. El profesor le asignará en la sesión de laboratorio, una de las secuencias que se indican en la siguiente sección de esta guía.
- b. Elaborar un diagrama de bloques de alto nivel (*diagrama de bloques principal*) para representar el problema principal junto con las entradas y las salidas del sistema. Este diagrama incluye el bloque que contiene la FSM junto con la lógica adicional requerida (codificador a 7 segmentos, divisores de frecuencia, etc.).
- c. Elaborar un diagrama de estados para la FSM a implementar, teniendo en cuenta los valores de las entradas que generan las transiciones y los valores de las salidas que se establecen en cada estado.
- d. Describir el diseño de la FSM y la lógica adicional usando SystemVerilog y emplear la herramienta de simulación ModelSIM para verificar el correcto funcionamiento del mismo. Llevar a cabo las correcciones pertinentes, si fuese el caso.
- e. Implementar el diseño en la FPGA disponible en el sistema de desarrollo DE10-Lite (disponible en el laboratorio LED) y verificar el correcto funcionamiento del mismo. Llevar a cabo las correcciones pertinentes, si fuese el caso.
- f. Escribir un corto reporte sobre el diseño e implementación de la práctica y subirlo a la plataforma de Ingeni@ junto con los archivos principales de su diseño en Quartus Prime.
- g. Sustentar el desarrollo en el horario de laboratorio correspondiente.

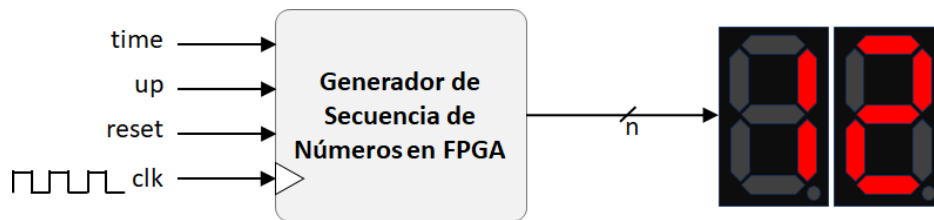


Fig. 1: Diagrama de bloques para el secuenciador

4 Especificaciones

A continuación se indican las especificaciones que el grupo de trabajo deberá considerar para el diseño e implementación del generador de la secuencia de números asignada, representado en la Fig. 1:

- a. La secuencia consiste de 8 números de 4-bits a generar a determinada frecuencia. El profesor le indicará cual de las siguientes secuencias deberá generar:
 - (a) **Seq. 1:** 1, 4, 7, 10, 13, 0, 3, 6.
 - (b) **Seq. 2:** 2, 6, 10, 14, 1, 5, 9, 13.
 - (c) **Seq. 3:** 3, 6, 9, 12, 15, 2, 5, 7.
 - (d) **Seq. 4:** 4, 8, 12, 0, 3, 7, 11, 15.
 - (e) **Seq. 5:** 5, 10, 15, 4, 9, 14, 3, 8.
 - (f) **Seq. 6:** 6, 8, 10, 12, 14, 3, 5, 7.
 - (g) **Seq. 7:** 7, 14, 5, 12, 3, 10, 1, 8.
 - (h) **Seq. 8:** 8, 11, 14, 1, 4, 7, 10, 13.
 - (i) **Seq. 9:** 9, 14, 3, 8, 13, 1, 6, 11.
 - (j) **Seq. 10:** 10, 13, 0, 3, 6, 9, 12, 15.
- b. La generación de la secuencia se realizará ascendentemente sí la entrada **up** está en '1' **lógico**; en caso contrario, la generación será realizada descendentemente. Ejemplo de generación ascendente: 1, 3, 5, 7, 9, 11, 13, 15, repetir continuamente. Ejemplo de generación descendente: 15, 13, 11, 9, 7, 5, 3, 1, repetir continuamente.
- c. El tiempo para la generación del siguiente valor de la secuencia será controlado por la entrada **time**. Cuando **time** es igual a '0' **lógico**, el tiempo entre valores de la secuencia será de un (1) segundo; en caso contrario, será de 0.5 segundos.
- d. La secuencia se deberá visualizar en **dos displays de 7 segmentos**, con el propósito de mostrar valores en decimal (tanto decenas como unidades). No se permite visualización en hexadecimal.
- e. Debe existir una entrada de reinicio (**reset**) que permita reiniciar la secuencia en cualquier momento.

5 Entrega

El grupo de trabajo deberá escribir un breve reporte en formato IEEE que contenga los siguientes elementos (ver guía para reportes en la página del curso):

- a. **Abstract:** resumen del diseño e implementación de la práctica
- b. **Diseño y Simulación:** diagrama de bloques del sistema, diagrama de estados de la FSM y una o varias imágenes que muestren el resultado de la simulación realizada. Tenga en cuenta que la simulación debe mostrar la secuencia tanto de manera ascendente como descendente.
- c. **Conclusiones:** dos o tres conclusiones sobre el trabajo realizado por el grupo de trabajo. Indicar el tiempo que les tomó realizar la práctica en las conclusiones.

Crear un archivo comprimido que incluya el reporte y los archivos importantes de su proyecto en Quartus Prime como se describe a continuación:

- a. **Reporte:** archivo con extensión .pdf
- b. **Archivos proyecto Quartus:** archivos con extensión .sv, .qpf y .qsf.

El nombre del archivo comprimido deberá tener el siguiente formato:

p1_primerapellidointegrante1_primerapellidointegrante2_horariolaboratorio.zip.

Ejemplo: si el primer apellido de ambos integrantes es **Castano** y **Garcia**, respectivamente, y el laboratorio es el Martes 9-12, entonces el archivo debe ser nombrado: *p1_castano_garcia_m9-12.zip*.

6 Evaluación

La evaluación de la práctica se divide en tres partes: funcionamiento (50%), sustentación (40%) y reporte (10%). La nota del funcionamiento se asigna por igual a todos los integrantes del grupo de trabajo (máximo dos personas por equipo), mientras que la nota de sustentación es individual. En caso un estudiante obtenga una nota inferior a 3.0 en la sustentación, la nota final de la práctica para el estudiante en mención será la que obtuvo en la sustentación, es decir, no se tendrá en cuenta el funcionamiento en el cálculo.

Cada grupo de trabajo deberá sustentar la práctica en un tiempo de 15 minutos, 8 minutos para revisar la implementación y 7 minutos para preguntas. Es importante tener abierto el proyecto y programada la FPGA para cuando el profesor llegue a su puesto de trabajo. No habrá tiempo para hacer correcciones de último momento.

7 Referencias

- a. Quartus Prime Lite Edition
<https://www.intel.com/content/www/us/en/products/details/fpga/development-tools/quartus-prime.html>

- b. Sistema de desarrollo DE10-Lite
<http://de10-lite.terasic.com/>
- c. Tutorial de SystemVerilog
<https://verilogguide.readthedocs.io/en/latest/>