

Laboratorio de Electrónica Digital II

Práctica 3: Circuitos Aritméticos para Números Decimales

Multiplicador de Números de Punto Flotante empleando Formato de Precisión Simple del Estándar IEEE-754

Profesores

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

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

Septiembre 5, 2023



Fecha de entrega: Del 19 al 22 de Septiembre de 2023
Medio de entrega: <https://virtualingenieriaudea.co/>
Sustentación: Horario de Laboratorio
Valor Práctica: 10% del curso

1 Introducción

En esta práctica de laboratorio, el grupo de estudiantes implementará un sistema electrónico digital para la multiplicación de números decimales empleando el formato de precisión simple (32-bits) del estándar IEEE-754. Para esta práctica, el hardware del circuito se describirá mediante dos bloques principales, control y *datapath*. El *datapath* será la unidad encargada de manipular los datos de entrada, realizar la operación de multiplicación y llevar el resultado a displays de 7-segmentos de

manera apropiada. Por otro lado, la unidad de control coordinará el funcionamiento del *datapath*. Cada una de las unidades mencionadas podrá a su vez tener otras sub-unidades encargadas de funciones específicas. La descripción del multiplicador junto con la lógica requerida para ingresar los datos y mostrar los resultados 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 multiplicador. 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 un sistema electrónico digital para la multiplicación de números de punto flotante empleando el formato de precisión simple del estándar IEEE-754, haciendo uso del 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. Estudiar la sección 5.3.2 Floating-Point Number Systems, del texto guía *Digital Design And Computer Architecture - ARMEdition*.
- 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 el multiplicador junto con la lógica adicional requerida para el ingreso de datos y visualización de resultados. Se sugiere analizar la plantilla en SystemVerilog disponible en la página del curso que se ha dejado como base para la descripción del multiplicador de número de punto flotante IEEE-754.
- c. Describir el diseño del multiplicador y la lógica adicional usando SystemVerilog y emplear la herramienta de simulación ModelSIM para verificar el correcto funcionamiento del mismo de los componentes internos del circuito. Llevar a cabo las correcciones pertinentes, si fuese el caso.
- d. 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.
- e. 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.
- f. Sustentar el desarrollo en el horario de laboratorio correspondiente.

4 Especificaciones

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

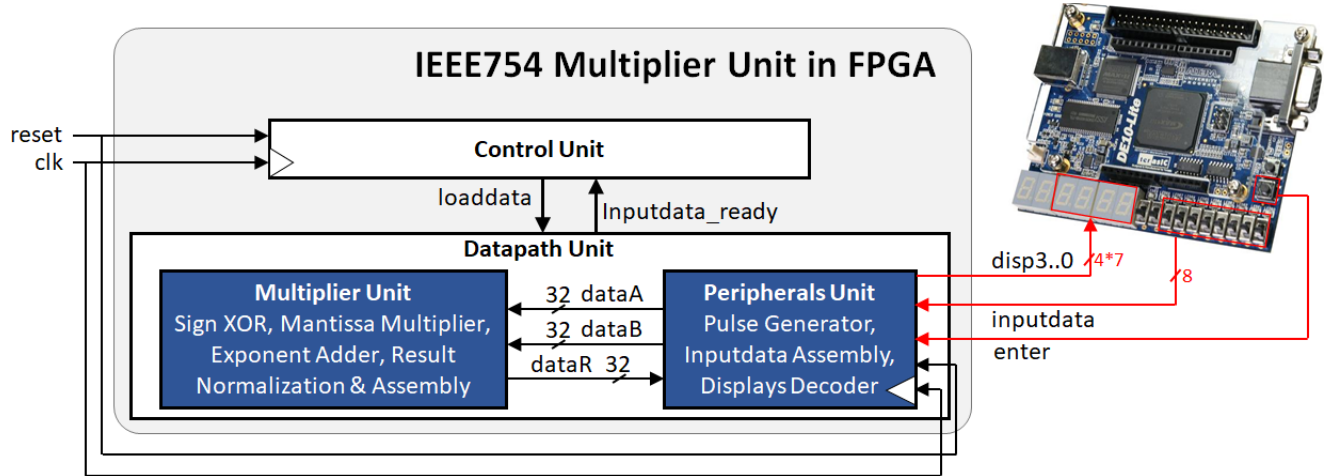


Fig. 1: Diagrama de bloques para el sistema

- El diseño del circuito digital debe incluir los dos bloques que se muestran en la Fig. 1: Control y *Datapath*.
- La unidad de Control será una simple máquina de estados encargada de coordinar el funcionamiento del *Datapath*. Después de un **Reset**, la unidad de Control le indicará al *Datapath* (**loaddata** = 1) que es el momento de cargar los valores A y B para realizar la operación. Una vez la unidad de *Datapath* haya recibido los valores de A y B, ésta le indicará a la unidad de Control (**inputdata_ready** = 1), para que retire la indicación de carga de datos (**loaddata** = 0) y se proceda a realizar la multiplicación.
- La unidad de *Datapath* se encargará de tomar los datos A y B desde los suiches del sistema de desarrollo DE10-Lite, realizar la multiplicación de punto flotante y, finalmente, mostrar el resultado en los displays de 7-segmentos del mismo sistema (DE10-Lite). Para estas funcionalidades, se han definido dos bloques principales: *Multiplier* y *Peripherals Units*. Cada bloque podrá contener uno o más sub-bloques encargados de realizar determinadas tareas.
- La unidad de multiplicación (*Multiplier Unit*) es un módulo combinacional encargado de realizar la multiplicación entre A (**dataA**) y B (**dataB**) y generar el resultado R (**dataR**) empleando el formato de precisión simple del estándar IEEE-754. La unidad de multiplicación debe estar en capacidad de resolver los siguientes casos especiales:

(a) $0 \times \pm finito = 0$

- (b) $0 \times \pm infinito = NaN$
- (c) $0 \times NaN = NaN$
- (d) $\pm finito \times \pm infinito = \pm infinito$
- (e) $\pm finito \times NaN = NaN$
- (f) $\pm infinito \times \pm infinito = \pm infinito$
- (g) $\pm infinito \times NaN = \pm NaN$
- (h) $NaN \times NaN = NaN$

e. La unidad de periféricos (*Peripherals Unit*) se encargará de dos funciones esenciales:

- (a) **Carga de operandos:** los operandos A y B son valores de 32-bits que serán cargados en 4 grupos de 8-bits cada uno, empleando para ello los suiches de la DE10-Lite (**inputdata**) y uno de los pulsadores denominado **enter**. Cada que se presiona el pulsador **enter**, la unidad de periféricos almacenará el valor dado por señal de entrada **inputdata** dentro del correspondiente registro. En total se requiere cargar un total de 8 bytes, cuatro para A y cuatro para B.
- (b) **Visualización de datos:** tanto los operandos A y B, como el resultado R, se deben poder visualizar en los displays de 7-segmentos de la DE10-Lite, en formato hexadecimal. Se dedicarán cuatro displays para la visualización de operandos y resultado. Los dos displays más significativos indicarán el operando y el byte que se está visualizando: A0 (LSB), A1, A2, A3 (MSB), B0 (LSB), B1, B2, B3 (MSB), R0 (LSB), R1, R2, R3 (MSB). En cambio, los dos displays menos significativos indicarán el valor del byte seleccionado. Por ejemplo, para la multiplicación entre $A = -18(0xC1900000)$ y $B = 9.5(0x41180000)$, la cual tiene como resultado $R = -171(0xC32B0000)$, los operandos A y B se visualizarán durante el proceso de carga de datos como se muestra en las Figs. 2 y 3. Observe que se ingresa primero el byte menos significativo de los operados. De manera similar, el resultado de la multiplicación se mostrará como se indica en la Fig. 4 .

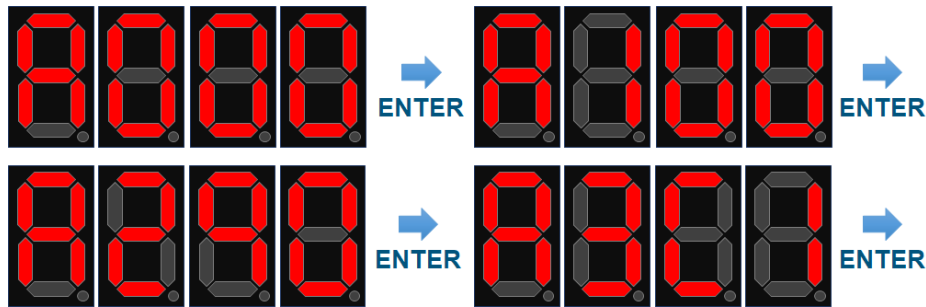


Fig. 2: Ingresando el operando A

- f. Finalmente, las entradas **clk** y **reset** son requeridas por los bloques que requieren circuitos secuenciales o memoria.

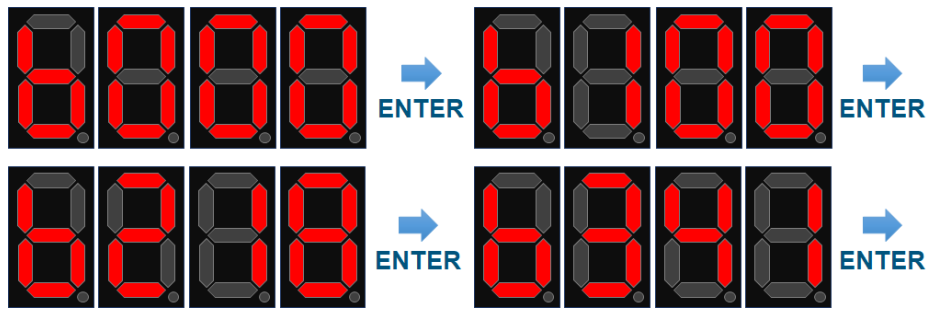


Fig. 3: Ingresando el operando B

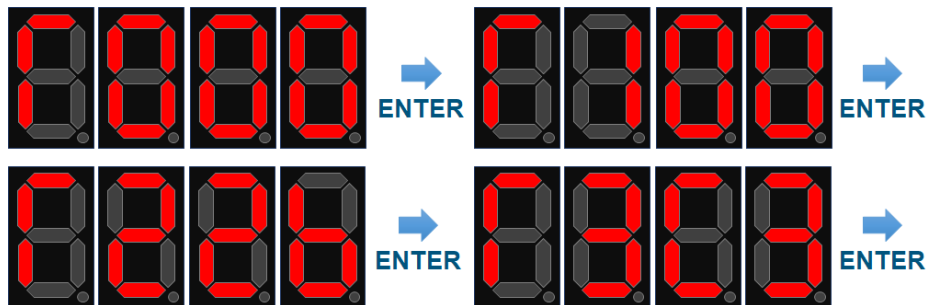


Fig. 4: Mostrando el resultado R

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):

- Abstract:** resumen del diseño e implementación de la práctica
- Diseño y Simulación:** diagrama de bloques del sistema y varias imágenes que muestren el resultado de la simulación realizada para las unidades *Multiplier*, *Datapath* y *Control*.
- 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:

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

El nombre del archivo comprimido deberá tener el siguiente formato:

p3_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: *p3_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/>