16 Bits Carry Skip Adder

Contreras R Orlando[1,2*] and Garcia B Iker[1,2*†]

[3]Departamento de Sistemas Electrónicos, UAA, Av. Universidad 940, Aguascalientes, 20131, Aguascalientes, México.

*Corresponding author(s). E-mail(s): {al348390,al307630}@edu.uaa.mx;
†Ambos autores contribuyeron de forma equitativa a este trabajo.

Abstract

El objetivo de la siguiente práctica es optimizar los tiempos en los sumadores, con el sumador de 16 bits no se logra percibir una diferencia, pero a grandes escalas los tiempos de retardo realmente importan.

Keywords: VHDL, Adder, bits, Tiempos, Sumador, FPGA

1 Introducción

En el diseño de circuitos digitales, la optimización juega un papel fundamental a la hora de ahorrar recursos, energía y tiempo. Esto genera que a la hora de diseñar circuitos se tome en cuenta el uso del circuito y el contexto de uso ya que todo cambia dependiendo a las necesidades del usuario.

2 Metodología

2.0.1 Diseño de los bloques de 4 bits

A la hora de hacer el sumador de 16 bits con carry skip se tomo la decisión generar bloques de 4 bits para facilitar las operaciones y el tamaño del código.

Esto hace mas modular el sumador permitiendo que a la hora de expandir sea mas fácil, practico y eficiente.

2.1 Optimización de tiempos

Se busca con el carry skip es el hecho optimizar el tiempo de respuesta, sin implementarlo el calculo de carry se tiene que hacer completo aun cuando la suma no se necesita haciendo que se gaste mas tiempo en completar el resultado. A la hora de implementarlo permite el ahorrar el tiempo de calculo del carry cuando no lo necesita permitiendo que los resultados tarden menos.

3 Resultados

En este caso, la salida termina siendo la misma, la diferencia radicaría en la optimización de tiempos.

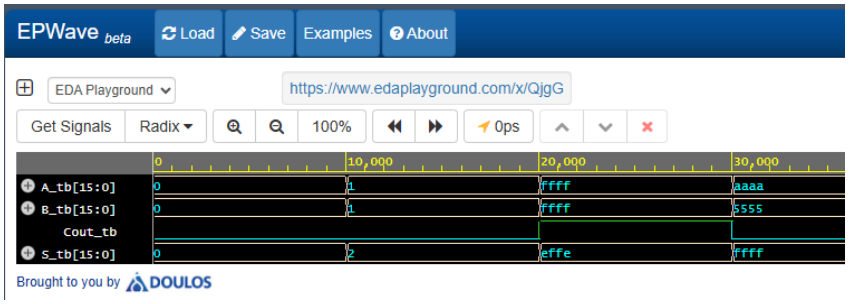

Fig. 1 Testbench realizado por EDA Playground

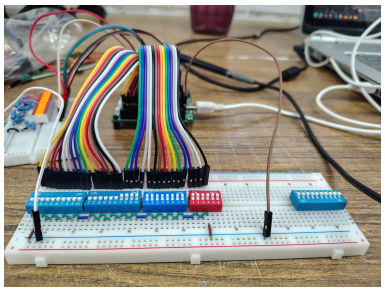

Fig. 2 Circuito Armado

4 Conclusiones

Esta práctica nos pareció muy sencilla debido a que directamente fue el mismo circuito anterior, nos hubiera gustado ver realmente la diferencia de tiempos para cuantificarlo, o verlo con alguna ecuación. Esperamos que con el software Xilinx se pueda hacer una simulación en tiempo real.