

Proyecto Final: Bichito Saltarín

Jorge Agüero Zamora, Arturo Chinchilla Sánchez, Luis Murillo Rios
georgeaz56@gmail.com mchinchilla11@gmail.com luismuelino@gmail.com

Área Académica de Ingeniería en Computadores
Instituto Tecnológico de Costa Rica

Resumen—En el presente proyecto se desarrolla una micro-arquitectura uniciclo que implementa un grupo de instrucciones sencillas de la arquitectura ARMv4. Esto con el fin de poder correr un juego llamado “Bichito Saltarín” que interactúa con un monitor VGA para mostrar las imágenes y con un teclado PS/2. Se utiliza SystemVerilog como HDL para describir los módulos de hardware y se implementa la solución en un FPGA Cyclone V sobre el dispositivo DE1-SoC de Terasic.

Palabras clave—ARMv4, PS/2 keyboard, Memory Mapped I/O, Sprite, VGA

I. INTRODUCCIÓN

El presente proyecto consta de la implementación de una micro arquitectura para el set de instrucciones de la arquitectura ARMv4. Se utilizan las instrucciones de esta arquitectura para desarrollar un juego simple llamado “Bichito Saltarín”.

El juego de “Bichito Saltarín” consiste en un personaje (el Bichito) que se mueve horizontalmente sobre una superficie plana, en la cual se presentan obstáculos de manera semi aleatoria, que se moverán de derecha a izquierda de la pantalla, para que el bichito deba saltar/agacharse para esquivarlos. El juego continúa hasta que el bichito colisione con algún obstáculo, el usuario presione el botón de reset, o el bichito haya caminado cierta distancia. Se presentará un conteo de la distancia recorrida por el bichito hasta el momento de la colisión y se volverá a dar inicio al juego.

Para interactuar con el juego se debió implementar un controlador PS/2 para teclado, de manera que este se pudiera usar para mover al bichito. De igual manera se implementó un controlador VGA para mostrar las imágenes del juego en un monitor.

Este documento está estructurado de la siguiente manera, sección de resultados, sección de análisis de los resultados obtenidos, conclusiones y las referencias utilizadas durante este proyecto.

II. RESULTADOS

II-A. Análisis de potencia

Utilizando la herramienta de Power Analyzer de Quartus se encontró que al utilizar las características típicas de operación de la herramienta, las cuales son, una temperatura de operación de 25 grados centígrados, temperatura de unión de

25 grados centígrados y un voltaje de operación (VCC) de 1.10 V, el consumo total de corriente del dispositivo sería 49.22 mA. El análisis de la disipación de poder del circuito mostró que la cantidad de energía disipada por el calor es de 419.72 mW.

II-B. Análisis de tiempo

Utilizando la herramienta Time Analyzer de Quartus se hizo una medida de la frecuencia máxima de operación del circuito entero, tomando en cuenta las memorias de datos e instrucciones, el controlador PS/2, los sprites y el controlador VGA. La frecuencia encontrada en una operación lenta a 1100mV a 85grados centígrados es de 375.52 MHz.

III. ANÁLISIS DE RESULTADOS

Con respecto a los módulos que conforman el sistema, la arquitectura del procesador ejecuta las instrucciones más básicas del set ARMv4 que se necesitan para la ejecución del juego, las cuales son LDR, STR, ADD, SUBTRACT, AND, OR y BRANCH.

Las instrucciones se cargaron de manera que se repitiera el ciclo de la creación semi-aleatoria de los obstáculos y cuando el usuario presionase una tecla (arriba o abajo) del teclado se genere una interrupción en el procesador, de manera que se ejecute otra sección de instrucciones para hacer que el “bichito” salte o se agache dependiendo de la tecla presionada.

Para cada Sprite (bichito, obstáculo y background) se utilizó una memoria por separado, de manera que cada una guardara la información de cada Sprite. Con ayuda de un MUX se seleccionaba las señales de RGB del Sprite que se iba a pinar.

Para los Sprites del Bichito y del Obstáculo se utilizaron dos registros, el primer registro para almacenar la posición en el eje Y del bichito, con esto se facilita la modificación de este parámetro cuando el bichito deba saltar, teniendo en cuenta que su posición en el eje X no cambia. El segundo registro almacena la posición en el eje X del obstáculo, que se debe ir moviendo a una velocidad uniforme desde la derecha hacia la izquierda, facilitando también la operación del procesador sobre esta

parámetro. Para el background no se utilizó ningún registro debido a que sus dos posiciones son fijas.

El funcionamiento del programa fue probado mediante la implementación de archivos de *testbench*, en los que se simulan las posibles entradas de la aplicación y el comportamiento de cada uno de los componentes que componen el sistema (procesador ARMv4, memorias de datos e instrucciones, controlador VGA, registros, etc). En la figura 1 se pueden observar las principales señales generadas por el sistema.

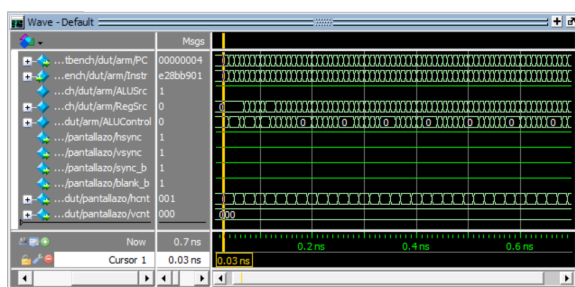


Figura 1. Señales generadas en el testbench aplicado al sistema del juego Bichito Saltarín

En la figura 2 se puede observar el juego del Bichito Saltarín mostrado utilizando un monitor VGA conectado directamente a la FPGA.

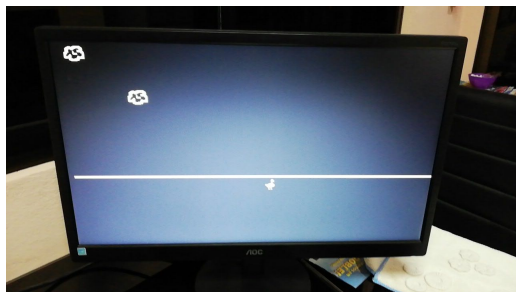


Figura 2. Juego del Bichito Saltarín presentado en un monitor VGA

IV. IMPACTOS SOCIALES Y MEDIOAMBIENTALES

El uso de dispositivos electrónicos es tan común hoy en día que las cantidades de desechos electrónicos que se producen se estiman a los 40 millones de toneladas métricas anuales en todo el planeta según datos de la UNEP (United Nations Environment Programme)[2]. Estos desechos terminan en vertederos de basura, donde no solo añaden a la contaminación por desecho sino que algunos de los elementos en los que se descomponen son tóxicos para el ambiente y para los humanos que viven cerca de estas zonas.

De igual manera la recolección de los metales utilizados para la producción de los dispositivos electrónicos proviene en gran cantidad de la minería, la cual también es una actividad usualmente

de gran impacto medioambiental, y usualmente se utilizan sustancias tóxicas para la extracción de estos, lo cual contamina el medio ambiente y pone en riesgo a las personas que trabajan en estas minas. La cantidad de metales que se pueden minar de la tierra es finito y eventualmente estas reservas se van a acabar, lo cual hace que esta actividad no sea sostenible en el tiempo.

Esto ha hecho necesario el reciclaje de los materiales electrónicos para poder asegurar continuidad del mercado de electrónicos. De igual manera se ha hecho más prevalente el tomar en cuenta los impactos socio-ambientales en el diseño de productos para intentar reducir los impactos negativos, técnicas como el Diseño para el Ambiente [3] (DeF, por su siglas en inglés) intentan definir procesos para poder lograr estos objetivos. Estas iniciativas logran reducir la cantidad de impactos negativos que el mercado de electrónicos tiene en el ambiente.

El presente proyecto implementará únicamente un dispositivo con la solución. El dispositivo utilizado para la implementación es una FPGA, la cual permite poder borrar la solución y reutilizar la misma tarjeta para otros tipos de proyectos, esto hace que el tiempo de uso del dispositivo donde se implementará la solución tenga una larga vida y no deba ser desechado al terminar el proyecto. El dispositivo será utilizado por otros estudiantes hasta que cumpla su vida útil, momento en el cual el Tecnológico se encargará de desecharlo responsablemente.

V. CONCLUSIONES

Al utilizar lógica secuencial para realizar el circuito se debe mantener el control del clock, para los distintos módulos ya que esto puede crear errores a la hora de la comunicación entre los mismos.

Los contadores proveen un mecanismo sencillo para poder dividir la frecuencia de un *clock* externo para así obtener las frecuencias deseadas.

Utilizar registros para almacenar datos que cambian con el tiempo facilita el proceso de operación de los mismos.

El funcionamiento de la FPGA es con lógica negativa, por lo que es recomendable realizar la implementación en SystemVerilog de la misma forma. Eso facilita el proceso de montaje, y evita errores.

REFERENCIAS

- [1] S. Harris and D. Harris, *Digital design and computer architecture*. Amsterdam [i 11 pozostałych]; Elsevier / Morgan Kaufmann Publishers, 2016.
- [2] United Nations Environmental Program (UNEP), *Recycling—From E-Waste to Resources* (New York: UNEP, 2009), accessed at www.unep.org, on Oct. 16, 2018.
- [3] Qian, X., & Zhang, H.C. (2009). Design for Environment: An Environmentally Conscious Analysis Model for Modular Design. *IEEE Transactions on Electronics Packaging Manufacturing*, 32(3), 164–175. DOI: 10.1109/TEPM.2009.2022544