

Diseño de un testbench UVM integrando IP de verificación Mentor Graphics y un IP propio, configurable y con soporte de cobertura

Autor: Daniel Baute Trujillo

Tutores: Dr. Valentín de Armas Sosa, Dr. Félix B. Tobajas Guerrero

GITT (Sistemas Electrónicos)

Julio de 2022

INTRODUCCIÓN

La verificación de productos de sistemas hardware digitales, está adquiriendo gran importancia, siendo en alguna medida la fase la que más tiempo se le dedica dentro del proceso de diseño. Es por lo que cada vez se impone la necesidad de buscar estándares y métodos de verificación que simplifiquen estas funciones, como es la metodología UVM [1].

OBJETIVO

El objetivo principal de este TFG es diseñar un *testbench* haciendo uso de los IP de verificación de *Mentor Graphics* para el protocolo AXI4, integrando un IP de verificación propio con protocolo *Wishbone*.

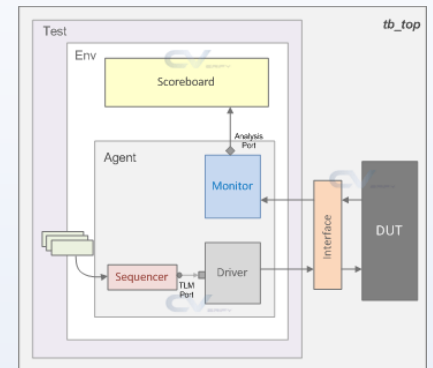


Figura I: Modelo de entorno UVM

METODOLOGÍA

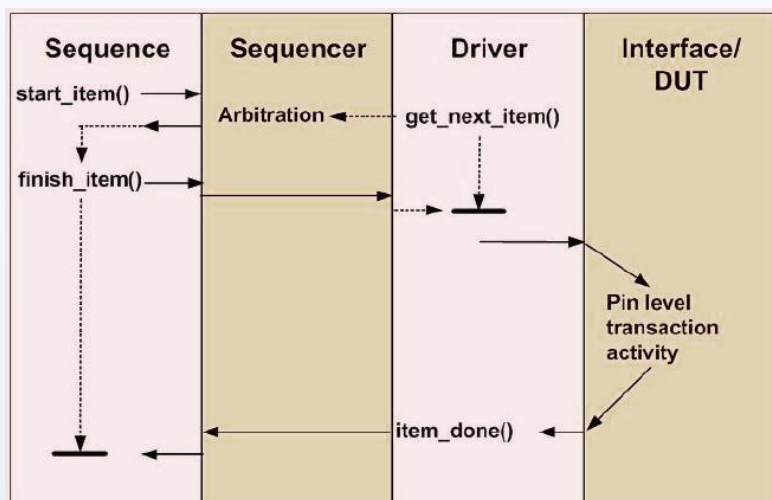


Figura II: Protocolo de *handshake* secuencia-driver

Este plan de verificación se basó en la estimulación de las entradas del DUV [2] con las secuencias creadas y en el exhaustivo análisis de las respuestas de la salida para verificar las características en el DUV seleccionado. Para ello, se desarrolló un entorno de verificación UVM y junto al IP de verificación propio correspondiente a un componente UVM *Agent* actuando como agente esclavo *Wishbone*. Las secuencias de este IP se desarrollaron con comportamiento reactivo, por lo que la función de los componentes correspondientes es reaccionar a las solicitudes del DUV. Para realizar esta función, es importante el protocolo de *handshake* utilizado para mandar la secuencia UVM al dispositivo a través del componente UVM *Sequencer* y el componente UVM *Driver*.

RESULTADOS

Se ha desarrollado un entorno de verificación UVM, además del diseño de las secuencias y la ejecución de las mismas, mediante la implementación de dos test diferentes para comprobar el funcionamiento del DUV, mediante el envío de una serie de transacciones de escritura y lectura. La diferencia entre ambos test consiste en el funcionamiento del agente esclavo reactivo. En primer lugar, este agente está en todo momento disponible para establecer la comunicación. En el segundo caso, el agente es capaz de pausar la comunicación manifestando la no disponibilidad del mismo. Se ha obtenido la respuesta esperada, por lo que el DUV se ha verificado con eficacia.

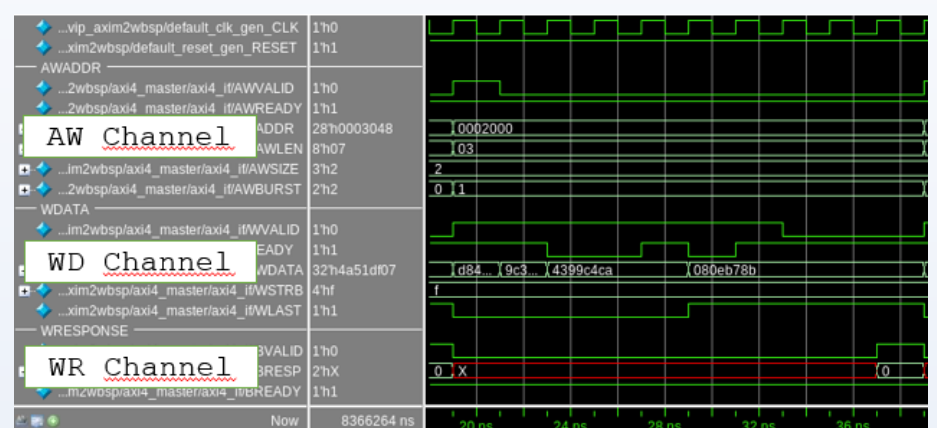


Figura III: Ejemplo de simulación de una escritura desde el punto de vista del protocolo AXI4

CONCLUSIONES

A la vista de los resultados obtenidos, todos los objetivos propuestos han sido cumplidos con éxito. El desarrollo de un entorno de verificación UVM junto a la capacidad de reutilización del código de los componentes programados, resulta un método eficaz, donde cada día se impone más la necesidad de buscar estándares y métodos de verificación que simplifiquen y faciliten las funciones de verificación.

REFERENCIAS

- [1] "Universal Verification Methodology (UVM) 1.2 User's Guide," 2015, Accessed: Jun. 01, 2022. [Online]. Available: <http://www.apache.org/licenses/>.
- [2] "wb2axip/axim2wbsp.v at master · Z ipC P U/wb2axip · GitHub." <https://github.com/Z ipC P U/wb2axip/blob/master/rtl/axim2wbsp.v> (accessed May 23, 2022).