

Verificación funcional y diseño de la implementación física de un procesador PicoSoC

Bitácora

Estudiante:

José Mario Navarro Bejarano
B75398

Contenido

Semana #1	3
Semana #2	4
Semana #3	5
Semana #4	6

Semana #1

Ω Sesión: miércoles 14 de agosto (2 horas):

- El miércoles se dedicaron dos horas al asistir a la clase introductoria del curso en donde se discutieron temas básicos necesarios para el curso.

Ω Sesión viernes 16 de agosto (7 horas):

- Ya el proyecto se encontraba seleccionado antes de comenzar el semestre, por lo que estas horas de la primera semana fueron dedicadas a realizar un estudio sobre el repositorio que contenía el RTL a utilizar, leer su documentación y en general obtener un mejor entendimiento del documento con el que se iba a trabajar.

Semana #2

Ω Sesión: lunes 19 de agosto (1 hora)*:

- Se mantuvo una reunión con el profesor Gerardo Castro, en la cual se discutieron los términos generales del proyecto.
- Se discutió el cronograma con el que se va a trabajar, incluyendo el documento requerido para entregar la próxima semana (test plan).

Ω Sesión: miércoles 21 de agosto (2 horas)

- Se asistió a clase presencial en donde se discutieron algunos temas del curso como el formato para algunos entregables y demás.

Ω Sesión: viernes 23 de agosto: (6 horas)

- La mitad de las horas fueron dedicadas a estudiar el DUT, el cual es un núcleo RISCv, por lo que fue necesario estudiar la especificación y entender la mayor cantidad de detalles posibles.
- La otra mitad del tiempo fue dedicado a realizar el documento test plan, con base en lo solicitado por el profesor y la especificación del DUT.

*Nota: En las sesiones con el profesor guía, se anotan varias mini-tareas que deben cumplirse antes de la próxima sesión. Sirven de recordatorio para las cosas que se deben realizar.

- Montar el testplan.
- Se decidió el cronograma, es necesario confeccionarlo formalmente.
- Se podría realizar un repositorio en Github, pero no es urgente.

Semana #3

Ω Sesión: lunes 27 de agosto (2 horas)*:

- Se tuvo una reunión con el profesor en donde se realizó la entrega de la primera versión del test plan.
 - Se discutieron detalles menores sobre la ejecución del proyecto.
 - Se acordó los archivos en cuales se debe seguir avanzando para entregar la próxima semana.

Ω Sesión: domingo 8 de agosto (7 horas):

- Se dedicó 1 hora a completar los archivos necesarios de la semana, como entregar el anteproyecto y crear y actualizar la bitácora con la información de las semanas anteriores.

*Nota: En las sesiones con el profesor guía, se anotan varias mini-tareas que deben cumplirse antes de la próxima sesión. Sirven de recordatorio para las cosas que se deben realizar.

- Reemplazar “co-pro” por BFM en entorno de verificación.
- Remover pruebas para C, E, Fence y System.
- Investigar funciones do_compare y do_copy (para posible uso en scoreboard)
- Investigar más sobre IRQ.
- Explicar más cómo obtener datos del DUT en el diagrama de bloques.
- Investigar test para comprobar compliance con RISC-V.
- Investigar sobre CSR.
- Buscar y usar un coding style.
- Recordar escribir el header en cada documento y en inglés.
- Se puede usar código en C para comprobar saltos de Branch.
- Aclarar test para PCPI, la interfaz en específico.

Semana #4

Ω Sesión: lunes 2 de septiembre (1 hora):*

- Se tuvo una reunión con el profesor en donde se habló sobre los avances del testplan y sobre los resultados de la información que había que investigar.

Ω Sesión: domingo 8 de setiembre (8 horas):

- Se estuvo trabajando principalmente en agregar el header con la licencia utilizada y en crear un primer test sencillo que funcionara, para comprobar la utilidad del uso de BFM en lugar de una memoria simulada.

*Nota: En las sesiones con el profesor guía, se anotan varias mini-tareas que deben cumplirse antes de la próxima sesión. Sirven de recordatorio para las cosas que se deben realizar.

- Investigar sobre alguna licencia para agregar al proyecto.
- Investigar cómo acceder a los valores de los registros sin tener que sacar las señales como output (talvez mediante operador punto)
- Recordar agregar “Fix Me” cuando se hace un cambio temporal en el código.
- Investigar e implementar UVC para reset, memoria y PCPI. Es decir, tener 3 o 4 driver, scoreboard, monitor...
- Estudiar virtual sequencer.
- Investigar sobre “manejo de registro” , “mapa de registros”, “acceso a registros”. UVM_Reg.
- Investigar las capas de UVM (como UVM_Reg)
- Reemplazar BFM en lugar de sim_mem.
- Ya que se va a usar BFM para la memoria, revisar en UVM_driver cuando el DUT responder (¿handshake?)
- Cambiar el diagrama de capas para hacerlo mas específico y menos genérico (por ejemplo, agregar los 3 agents).
- Agregar la sección de compliance en el testplan.
- Tener prueba de concepto del BFM para la próxima reunión.
- Revisar cómo funciona y cómo se hace la verificación formal del DUT.