

Proyecto Arquitectura

Implementación de una microarquitectura para el ISA RISC-V

I. Objetivo General.

Familiarizar al estudiante en el desarrollo de una microarquitectura; aplicando herramientas de implementación digital, métodos de diseño y conceptos de microprocesadores para implementar el set de enteros de la arquitectura de código abierto RISC-V.

II. Descripción del trabajo.

En este proyecto de diseño en ingeniería el grupo deberá enfrentarse a una labor de complejidad media para estudiantes de ingeniería de séptimo semestre, por ello este trabajo requiere (al igual que todo proyecto) de una adecuada planificación de las tareas a realizar por parte de los estudiantes. La evaluación se compone de una serie de avances de carácter formativo con el objetivo de guiar y retroalimentar a los estudiantes en el proceso de diseño, y dos evaluaciones sumativas que se detallan en secciones posteriores.

En este proyecto se requiere implementar una microarquitectura que soporte una parte reducida del set de enteros del ISA RISC-V: RISC-V32I. La implementación debe cumplir los siguientes requerimientos:

- Soportar varias de las instrucciones indicadas por el ISA RISC-V secciones 2.1 a la 2.6. Solo se deben soportar las instrucciones necesarias para la ejecución de las pruebas funcionales del proyecto.
- Usar el lenguaje de descripción de hardware Verilog para su implementación.
- Contemplar una ruta de datos y una unidad de control debidamente identificables.
- Ser desarrollada con el software gratuito ISE de Xilinx para implementación de hardware, en su defecto los estudiantes pueden proponer otro software de implementación de hardware que tengan a su disposición.

Propiamente el diseño y desarrollo del proyecto debe contemplar las siguientes secciones:

Diseño de la Microarquitectura:

Cada grupo en sus labores de diseño deben presentar circuitos que muestre gráficamente la solución a implementar. Para esto debe de considerar:

- Descripción de componentes o módulos y la funcionalidad de cada uno.
- Descripción de las señales de control de la Unidad de Control.
- Defensa de cada componente de la solución, haciendo alusión a alguna instrucción de RISC-V32I, un estilo de decodificación, entre otros.
- No se aceptan diagramas hechos a mano.

Los estudiantes tienen la libertad de presentar uno o más circuitos si esto facilita la explicación de su solución; no obstante, es obligatorio como mínimo presentar el circuito de la solución final completa.

Desarrollo en ISE:

La implementación debe ser realizada en un software de descripción de hardware mediante lenguaje Verilog. Para este desarrollo se espera lo siguiente:

- Cada módulo debe estar debidamente documentado y comentado. Se espera documentación sobre la funcionalidad del módulo, sus entradas, salidas y la razón de porque se introdujo el módulo.
- Cada módulo implementado debe poseer su respectivo archivo de prueba. El cual documenta como mínimo:
 - Objetivo de la prueba
 - Estímulos: El orden o secuencia en que se varían las señales de entrada del módulo
 - Descripción de resultados esperados
- El módulo principal solo puede tener dos entradas: Clock y Reset.
- Durante la evaluación, los estudiantes solo pueden modificar un componente: La memoria de instrucciones. Esto para cargar los distintos archivos compilados para su ejecución. Si los estudiantes requieren de cambios a su arquitectura “en media defensa”, solo podrán elegir una de tantas configuraciones para defender.

Pruebas Funcionales:

Se les facilitará a los estudiantes 1 código en lenguaje C. La microarquitectura debe de ser capaz de correr completamente las instrucciones de ensamblador necesarias para ejecutar dicho código. Las siguientes consideraciones aplican:

- Es responsabilidad de los estudiantes compilar el código y extraer el binario/hexadecimal para la arquitectura RISCV32. Este proceso será verificado durante el primer avance.
- Solo se deben probar las instrucciones afines a las líneas de código C. Llamados a sistema operativo, overheads y similares pueden ser ignorados. También se pueden ignorar las instrucciones posteriores al “}” de cierre de la función main.
- Los estudiantes deben hacer uso del software ISE/Vivado para poder hacer un dump de la memoria de dato, esto para facilitar la revisión. El dump de memoria debe de realizarse a algún archivo y presentar la información de una forma similar a la siguiente:

Dirección de Memoria	Valor en hexadecimal
0x0	0xABCD0000
0x4	0xEF9876543

- En caso de no soportar los códigos C completamente los estudiantes pueden demostrar las funciones que su microarquitectura puede soportar. Queda a responsabilidad de los estudiantes proveer los binarios para estas demostraciones.

III. Consideraciones generales:

1. El proyecto debe realizarse en grupos de 5 personas.
2. **Todos** los miembros del grupo deberán estar en la presentación final de lo contrario se le asignará un 0 al estudiante que falte y no presente la justificación respectiva.
3. Se deberá entregar un informe final con todo el diseño detallado. Descripción de cómo se implementó cada tipo de decodificación del RISC32I, justificación de módulos (con la referencia oportuna a una instrucción o una decodificación particular) y un análisis de los resultados. **Máximo 6 páginas** formato IEEE doble columna, diagrama de los circuitos parciales pueden ser agregados como anexos sin repercusión en el tamaño del documento. Todos los grupos de trabajo deben llevar una bitácora donde cuenten sus vivencias, toma de decisiones, anotaciones de reuniones de los grupos, cálculos, bosquejos necesarios y todo aquello que consideren pertinente para un adecuado seguimiento del proceso de diseño.

IV. Entregables:

Avance 1: Se deberá presentar en Semana 10 al final de la clase una breve explicación de cuales instrucciones del set RISC32I debe de soportar la arquitectura del proyecto (esto se puede obtener compilando los códigos C para RISC32I). Deben también presentar un circuito propuesto para soportar las instrucciones tipo S e I; puede ser en conjunto o por aparte.

Avance 2: Se deberá presentar en Semana 13. El equipo debe presentar todos los circuitos necesarios para soportar todas las decodificaciones, así como presentar la implementación que se tiene hasta el momento. La fecha y hora de la cita se definirá en conjunto entre el profesor y los estudiantes.

La entrega final del proyecto se realizará en Semana 17, mientras que el informe final deberá subirse en formato pdf al TecDigital antes de las 11:45 p.m. del mismo día, no se admiten documentos fuera de plazo o que no cumplan con la forma de entrega. La revisión se realiza mediante la plataforma Zoom o de forma presencial según sea el caso, uno de los estudiantes deberá compartir escritorio y presentar la solución al profesor de curso; el proceso de agendar citas de defensa, así como el día exacto de la defensa se definirá a lo largo del semestre.

V. Evaluación

Rubro	Puntaje	Fecha de entrega
Avance 1: Informe de análisis de requisitos y circuitos iniciales.	Formativo	Semana 10 Día de Clases
Avance 2: Revisión de circuitos completos e implementación parcial	Formativo	Semana 13
Entrega Final	70 pts	Semana 17 Día de Clases
Informe Final	30 pts	Semana 17 Día de Clases
Nota Final	100	

Evaluación de la Entrega final del Proyecto

Criterio	Detalle	Valor
Soporte de Código C propuesto	La capacidad de soportar el código C entregado por el profesor. Esto en la simulación usando el software ISE.	Excelente (15pts): Abarca todas las instrucciones necesarias correctamente Medio(6pts): Debe eliminar las operaciones tipo Load o Store para poder correr el código. Malo (0 pts): El código no puede ser ejecutado del todo.
Total de decodificaciones soportadas.	La microarquitectura propuesta es capaz de soportar las decodificaciones necesarias. Esto evaluado en el simulador ISE. NOTA: Soportar una decodificación se traduce a soportar todas las instrucciones necesarias para esa decodificación	Se otorga 3 puntos por cada decodificación. (Max=12pts)
Diseño de solución y circuitos.	Circuitos y diseño correcto capaz de soportar las distintas decodificaciones. Este rubro no depende de la implementación final en ISE. Nota: Se recomienda presentar un diagrama por codificación.	Se otorga 3 puntos por cada decodificación y la UC. (Max=15pts)
Implementación de Componentes Clave	Implementa correctamente los siguientes bloques en Verilog: <ul style="list-style-type: none"> - ALU - Memoria de datos/instrucciones - Unidad de Signo - UC 	Se otorga 4 puntos por cada componente. (Max=16pts)
Dump de memoria	Capaz de hacer un dump de memoria de las RAMs. Como mínimo debe de ser capaz de realizarlo en un test del módulo de memoria de datos.	Criterio binario (5pts): Funciona o no funciona.
Testbench	Posee un archivo de pruebas para cada una de los módulos implementados en ISE.	Excelente (7pts): Todos los módulos implementados poseen un testbench debidamente documentado Bueno (4pts): Todos los módulos implementados poseen un

		testbench, los cuales no están debidamente documentados Mínimo (2pts): No todos los módulos poseen testbenches, indiferente de la calidad de documentación. Mal (0pts): Solo existe un testbench, el de la solución completa
Respetar las únicas dos entradas del proyecto	El módulo final posee solo dos entradas: Clock y reset. Esto no cuenta el cambio al código cargado en las RAMs para la revisión de las distintas partes del proyecto.	Criterio binario, funciona o no funciona (5pt)
Bitácora	Todos los grupos de trabajo deben llevar una bitácora donde cuentan sus vivencias, toma de decisiones, anotaciones de reuniones de los grupos, cálculos, bosquejos necesarios y todo aquello que considere pertinente para un adecuado seguimiento del proceso de diseño.	Criterio binario (4pts): Posee o no posee bitácora legible y completa según requerimientos.

Nota 1: No olviden acatar todas las instrucciones que se encuentran en el programa del curso.

Nota 2: La bitácora deberá ser virtual.

Nota 3: Se asume que los estudiantes poseen conocimiento de programación Verilog, o bien están en capacidad de aprender lo necesario para el desarrollo del proyecto.

Nota 4: En caso de que durante la evaluación se observe un error que no calce exactamente con la rúbrica el profesor tiene la capacidad de asignar una penalización adecuada a su criterio. Es deber del profesor documentar esta nueva penalización para lograr una justa evaluación de todos los equipos.

Nota 5: La rúbrica de evaluación del informe se encuentra disponible en el TecDigital.