Trabajo Practico de Laboratorio

Arquitectura de Computadoras 1 (2021)

Profesor Asignado: Martín Vázquez

Lujan, Nicolas

email: Nico.L.2014@hotmail.com

Martinefsky, Ezequiel

email: ezequielmartinefsky@hotmail.com

**Implementación de Registro y FF:**

Tanto los registros como los Flip-Flops utilizados para almacenar datos en las pipelines fueron modelados utilizando un mismo proceso sensible al clk y al reset.

En un inicio el modelado de dichos registros fue a partir de procesos individuales para cada uno, posteriormente y con el fin de un código más ordenado se optó por instanciar el registro.

Dado que era necesario agregar un registro de 5 bits como una implementación de FF, coincidimos en que la mejor opción finalmente era la de implementar un único proceso que englobe a todos los registros/FF.

**Modelado de las Instrucciones Inmediatas:**

Para que el procesador opere con las instrucciones inmediatas pedidas (addi, ori, lui y andi) se agrega un bit a ALUOp, (ALUOp2), existiendo así las señales ALUOp0, ALUOp1 y ALUOp2 para el control de la ALU.

Este nuevo bit ALUOp2, es igual a 1 si se trata de una de las señales mencionadas anteriormente y 0 en otro caso.

Con los 2 bits restantes (ALUOp1 y ALUOp0), teniendo 4 combinaciones posibles, se configuran para que cada combinación represente una de las instrucciones que se buscan agregar al procesador:

| (ALUOp2, ALUOp1, ALUOp0) | Instrucción |
| --- | --- |
| 100 | LUI |
| 101 | ADDI |
| 110 | ANDI |
| 111 | ORI |

*\*la tabla es implementada en los If’s anidados a partir de la línea 300 del archivo processor.vhd*

**Consideraciones:**

Para los archivos “programa”, y “datos” que contienen las memorias de instrucciones y datos respectivamente fue necesario dar la ruta completa de dichos archivos cambiando las líneas 108 y 122 de processor\_tb.vhd.

En caso de ejecutar el proyecto en un equipo diferente al autor, cambiar dichas líneas completando la ruta de los respectivos archivos.

**Entorno de simulación utilizado:**

En su gran mayoría, el entorno de simulación utilizado fue Vivado, frente a algunos inconvenientes iniciales con la plataforma, también se trabajó en EdaPlayground para las etapas iniciales (como la implementación de la ALU).

**Conclusión:**

La simulación de los archivos implementados en conjunto con los archivos presentados por la cátedra se comporta de la manera esperada, generando los resultados deseados frente al set de instrucciones introducido.

La elaboración del trabajo se desarrolló de manera acorde a lo presentado en cada clase práctica, sin grandes inconvenientes, quitando algunos problemas iniciales propios del lenguaje.

Las teorías presentadas junto las consultas tratadas y la bibliografía propuesta fueron material suficiente para una correcta elaboración del trabajo.