

Implementación de la instrucción jal en el procesador MIPS

Se suministra un modelo del procesador en SystemC con precisión de ciclo de reloj. En la implementación faltan algunas unidades funcionales y partes del control, pero es funcional para un buen número de instrucciones básicas.

Para esta parte del proyecto se requiere implementar la instrucción jal (jump-and-link), que no está soportada actualmente.

La tarea consiste en crear tanto el código que realice el salto y la modificación de \$ra, como completar el control. Tenéis libertad para realizar cambios en cualquier parte del HW, pero debéis justificarlos.

El código de operación de jal es 0x03. Aparte de esto, el formato es idéntico al de j (jump).

Para comprobar que todo funciona correctamente, se ejecutará un programa ensamblador que se proporciona ("programaJal.txt") que se proporciona. También hay un fichero de datos que no se usa en este caso ("datos.txt"). Es necesario utilizar ambos nombres de fichero al instanciar el procesador. Al final de la ejecución, el volcado del banco de registros debe arrojar los valores listados al final de este documento. De lo contrario, se habrá producido un error que deberá ser localizado y resuelto.

Como prueba del trabajo, es necesario presentar el código final, una captura de la salida del programa, y una explicación por escrito de los cambios y adiciones que se han realizado. No existe productor ni consumidor, el contenido de los registros es la única prueba de que ha funcionado correctamente. Finalmente, se explicará por videoconferencia el trabajo realizado al resto de la clase.

\$0 : 00000000	\$t0: 00000004	\$s0: 00000010	\$t8: 00000018
\$at: 00000001	\$t1: 0000000c	\$s1: 00000011	\$t9: 00000019
\$v0: 00000002	\$t2: 00000014	\$s2: 00000012	\$k0: 0000001a
\$v1: 00000003	\$t3: 0000001c	\$s3: 00000013	\$k1: 0000001b
\$a0: 00000004	\$t4: 00000024	\$s4: 00000014	\$gp: 0000001c
\$a1: 00000005	\$t5: 0000002c	\$s5: 00000015	\$sp: 0000001d
\$a2: 00000006	\$t6: 00000034	\$s6: 00000016	\$fp: 0000001e
\$a3: 00000007	\$t7: 0000003c	\$s7: 00000017	\$ra: 00000048