Universidad Zaragoza
1542

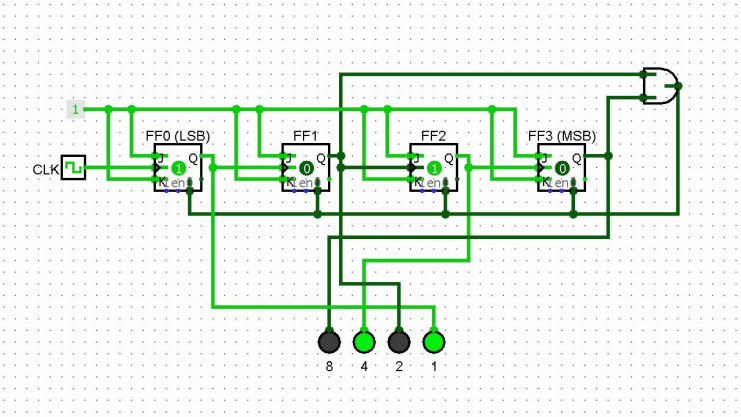
ARQUITECTURA Y ORGANIZACIÓN DE COMPUTADORES 2

2º DE GRADO EN INGENIERÍA INFORMÁTICA, CURSO 2020/2021

UNIVERSIDAD DE ZARAGOZA

PRÁCTICA 3

DISEÑO DE UN PROCESADOR CON BANCO DE REGISTROS



Luis M. Ramos
José Luis Briz
Javier Resano
Francisco J. Martínez

Logisim
una herramienta de diseño y simulación de circuitos lógicos

Universidad Zaragoza
1542

CONTENIDO

1 DISEÑO DE UN PROCESADOR

En esta práctica se va a diseñar *un procesador muy sencillo que trabaje con el banco de registros y un sumador. Lo programaremos para que realice una serie de escrituras y sumas. Una vez diseñado, realizaremos un análisis temporal para averiguar la frecuencia máxima a la que puede funcionar dicho procesador.*

Para poder realizar la práctica, se recomienda que antes de ***comenzar a usar el Logisim****, realicéis el* ***Trabajo Previo (Apartado 2.1).*** *Además, debéis disponer de vuestro* ***banco de registros*** *diseñado en la Práctica 2,* ***finalizado y funcionando correctamente****.*

Duración: 2 sesiones de laboratorio (4 + 4 horas)

Universidad Zaragoza
1542

2 DISEÑO DE UN PROCESADOR CON BANCO DE REGISTROS

2.1 TRABAJO PREVIO: DISEÑO EN PAPEL

1. Dibujad sobre papel los siguientes componentes: un banco de registros BR32, una ROM, un contador, un sumador de 32 bits, un extensor de signo y un multiplexor. Interconectad los componentes teniendo en cuenta que:

    a. En cada dirección de la ROM se escribirá una instrucción.
    b. El contador se utilizará para direccionar la ROM.
    c. Con el sumador se calculará la suma de los dos registros leídos del BR32.
    d. El multiplexor seleccionará el dato adecuado a escribir en el BR32.
    e. El procesador deberá soportar las siguientes instrucciones:

```
mov rd,inmed  ;BR(rd) ← SignExt(inmed) inmed. const. de 16 bits
add rd,rs,rt  ;BR(rd) ← BR(rs) + BR(rt)
```

2. Definid el formato de instrucción y la codificación de las instrucciones.

3. Diseñad la Unidad de Control del procesador, para que calcule el valor de las señales de control, a partir del código de operación de la instrucción.

4. Utilizando las instrucciones anteriores, escribid un programa (NIA.asm) que almacene los 6 dígitos de vuestro NIA en los registros r0 – r5, calcule la suma de todos los dígitos y la guarde en r6.

2.2 TRABAJO EN EL LABORATORIO: DISEÑO CON LOGISIM

2.2.1 DISEÑO MONOCICLO

a) Abrid en Logisim el circuito de la práctica anterior. Aseguraos de que no usáis ningún componente Reloj (la señal de reloj debe estar conectada a una entrada normal). Añadir un circuito nuevo. Nombradlo "PROCESADOR" y marcadlo como circuito principal. Guardadlo en un fichero .circ nuevo (NIA-monociclo.circ).

b) Colocad los componentes del apartado 2.1 e interconectadlos según vuestro diseño. Añadir un componente Reloj (Wiring/Reloj) y conectadlo al contador y al BR32.

c) Traducid el programa que habéis realizado en el último punto del apartado anterior a hexadecimal, y programad la ROM para que lo incluya.

d) **Ejecutad ciclo a ciclo hasta que funcione correctamente**.

2.2.2 DISEÑO MULTICICLO

a) Guardad el diseño anterior en un fichero nuevo (NIA-multiciclo.circ).

b) Modificad el diseño del procesador, conectando dos registros a la entrada del sumador (regA y regB) y otro a la salida (regSumOut).

c) Dibujad el diagrama de estados de la unidad de control para el nuevo procesador multiciclo. Implementadlo usando un registro (estado) y una única ROM de control (que incluya, por tanto, la función de transición y la función de salida).

d) **Ejecutad ciclo a ciclo hasta que funcione correctamente**. Debéis entregar también el autómata de la unidad de control NIA-multiciclo.fsm (dibujado con la herramienta Qfsm que tenéis disponible en moodle).

2.3 ANÁLISIS TEMPORAL: RESULTADOS

a) Considerad los siguientes retardos: d_{NOT}=5ps; $d_{SignExt}$=10ps; $d_{OR2\text{-}4}$=30ps; $d_{AND2\text{-}4}$=30ps; d_{REG}=70ps; $tsetup_{REG}$=30ps; d_{ROM}=95ps; d_{CONT}=75ps; d_{SUM}=180ps.

b) Identificad el camino crítico del procesador monociclo y calculad su retardo ($dmax_{monociclo}$). ¿Cuál será la frecuencia máxima de funcionamiento ($fmax_{monociclo}$)?

c) Identificad el camino crítico del procesador multiciclo y calculad su retardo ($dmax_{multiciclo}$). ¿Cuál será la frecuencia máxima de funcionamiento ($fmax_{multiciclo}$)?

d) ¿Cuánto tiempo tardará el procesador monociclo en ejecutar vuestro programa ($tex_{monociclo}$)?

e) ¿Cuánto tardará el procesador multiciclo ($tex_{multiciclo}$) en ejecutar el mismo programa?

f) ¿Cuál de los diseños es más rápido? ¿Cómo podrías cuantificarlo? Indica el porcentaje de mejora de uno con respecto a otro.

2.4 MATERIAL A ENTREGAR

Deberéis entregar todo lo realizado en los apartados 2.1, 2.2 y 2.3 (incluyendo el diseño en papel, los circuitos implementados, el autómata de la unidad de control y un documento/memoria de la práctica), dentro del plazo establecido, que os comunicaré por moodle. Para que os podáis planificar, la entrega se deberá realizar después de volver de Semana Santa.

Se recomienda la entrega a través de la página web de la asignatura (https://moodle2.unizar.es/add/course/view.php?id=36949).

Debéis enviar los siguientes documentos:

1. Memoria en formato PDF
2. Ficheros con las implementaciones realizadas en los diferentes apartados

Se pueden mandar los documentos por separado, o mandar un único fichero .zip por grupo. El fichero se nombrará de la siguiente manera:

P3_NIA-Apellidos_Estudiante1_NIA-Apellidos_Estudiante2.zip
Por ejemplo: P3_345456-Gracia_Esteban_45632-Arribas_Murillo.zip