



Universidad
Zaragoza

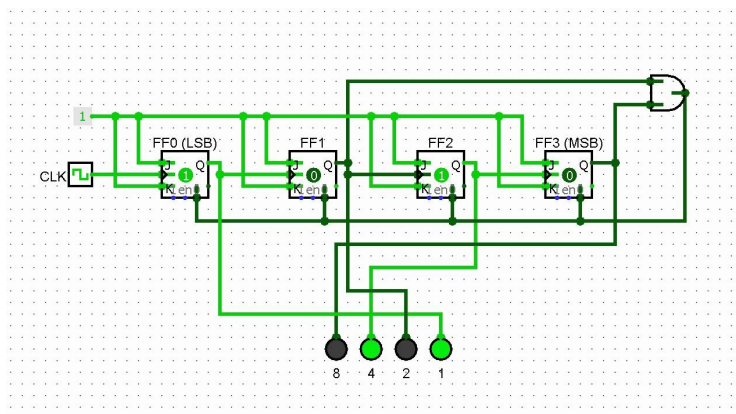
ARQUITECTURA Y ORGANIZACIÓN DE COMPUTADORES 2

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

UNIVERSIDAD DE ZARAGOZA

PRÁCTICA 2

DISEÑO DE UN BANCO DE REGISTROS



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



CONTENIDO

1	Introducción.....	1
2	Objetivos.....	2
3	Diseño de un banco de registros 32x32	2
3.1	Trabajo previo	3
3.2	Trabajo en el laboratorio: diseño con Logisim	3
3.3	Análisis temporal: Resultados	4
3.4	Material a entregar	5

1 INTRODUCCIÓN

En esta práctica se va a diseñar uno de los elementos principales de la mayoría de procesadores: el banco de registros (BR). En este caso en concreto, vamos a diseñar un banco de registros para un procesador MIPS. Seguiremos un procedimiento de diseño modular, utilizando bloques combinacionales básicos: registros, multiplexores y decodificadores. Finalmente, una vez diseñado, comprobaremos que funciona correctamente y realizaremos un análisis temporal, calculando además la frecuencia de reloj máxima (o el tiempo de ciclo mínimo) que permita su correcto funcionamiento.

Duración: 2 sesiones de laboratorio (4 horas)

2 OBJETIVOS

- Aprender a realizar diseños modulares. Utilizando módulos estándar, vamos a diseñar sistemas complejos de forma eficiente.
- Saber caracterizar el comportamiento temporal de un módulo síncrono complejo, determinando el tiempo de setup, el retardo y la frecuencia máxima.

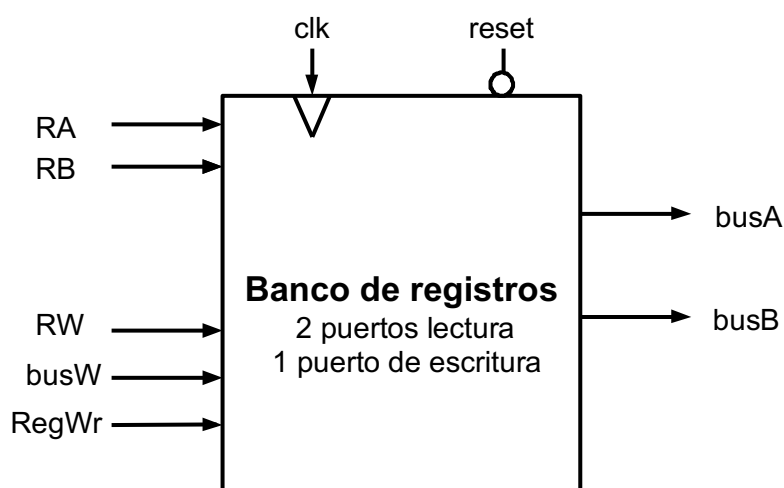
3 DISEÑO DE UN BANCO DE REGISTROS 32X32

En esta parte de la práctica, hay que diseñar un banco con 32 registros de 32 bits (BR32), con dos puertos de lectura y uno de escritura. Cada puerto de lectura tiene una entrada de dirección (**RA** y **RB**) y una salida de datos (**busA** y **busB**). El puerto de escritura tiene una entrada de dirección (**RW**), una entrada de datos (**busW**), y un permiso síncrono de escritura (**RegWr**, flanco ascendente en **clk**). Formalmente:

$\text{busA} = \text{BR}(\text{RA})$ después de T_{delayBR} ;

$\text{busB} = \text{BR}(\text{RB})$ después de T_{delayBR} ;






$\text{BR}(\text{RW})[\text{ciclo } i] \leftarrow \text{busW}[\text{ciclo } i-1]$ si busW respeta T_{setupBR} ;





3.1 TRABAJO PREVIO

1. Escribid en una tabla el número de bits de los que consta cada señal de entrada y de salida del banco. Debéis entender y comentar de qué depende el número de bits de cada señal.
2. Diseñad sobre el papel un banco de 4 registros de 32 bits (**BR4**), utilizando cuatro registros de 32 bits, dos multiplexores MUX 4:1x32 y un decodificador DECOD 2:4. El banco de registros debe permitir dos lecturas y una escritura en cada ciclo. En las entradas **RA** y **RB** se indica qué registros se quieren leer, obteniéndose los datos en **busA** y **busB**. Si **RegWr=1**, el dato en **busW** se escribirá en el registro **RW** en el flanco ascendente de final de ciclo.
3. Diseñad un banco de 32 registros de 32 bits (BR32) utilizando 8 BR4, dos MUX 8:1x32 y un DECOD 3:8. Recordad que el banco de registros debe permitir dos lecturas y una escritura en cada ciclo. Utilizad los mismos nombres en las entradas y salidas que en el BR4.

3.2 TRABAJO EN EL LABORATORIO: DISEÑO CON LOGISIM

- a) Colocad 4 registros (carpeta de componentes *Memoria*) en una hoja de Logisim. Configuradlos como registros de 32 bits (*propiedades/bits de datos*). Colocad un DECOD 2:4 (carpeta *Plexores/Decodificador* y *propiedades/seleccionar bits 2*) cuyas salidas controlen los permisos de escritura de los registros (**en**). Interconectad las señales de **reset**, **reloj** y de entrada de **datos**.
- b) Colocad dos MUX 4:1x32 (*Plexores/Multiplexor*, *propiedades/seleccionar bits 2*, *propiedades/Bits de Datos 32* y *propiedades/Include Enable? No*) que seleccionen el dato de alguno de los registros.
- c) Colocad entradas  para las señales **reset**, **reloj** y **busW**, **RA**, **RB**, **RW** y **RegWr**. Configuradlas con el número de bits correcto (*propiedades/Bits De Datos*) y nombradlas (*propiedades/etiqueta*). Conectadlas donde corresponda. Conectad salidas  a los MUX, configuradlas de 32 bits y nombradlas (**busA** y **busB**).
- d) Ya tenemos un banco de 4 registros. Comprobad que funciona correctamente realizando varias escrituras (**RW**, **busW**, **RegWr**), lecturas (**RA**, **RB**) y borrados (**reset**).
- e) Renombrad el circuito como "BR4" (Clic en  **main** /*propiedades/Nombre Del Circuito*). Para que quede más claro, debéis cambiar la apariencia del encapsulado y la posición de los pines pulsando . Pulsad  para volver al diseño del circuito.

- f) Añadir un circuito nuevo (en el menú de componentes *clic* derecho en carpeta raíz y selecciona *Añadir circuito*). Nombradlo como “BR32”. Marcadlo como circuito principal (*Clic* derecho en  **BR32**/Seleccionar Como Circuito Principal).
- g) Colocad 8 BR4. Añadir un DECOD 3:8 para que la escritura se efectúe correctamente en uno de los 32 registros. Añadir MUX 8:1x32 para que las dos lecturas se efectúen correctamente.
- h) Colocad entradas y salidas para todas las señales del banco de registros de 32 bits (también para la señal de reloj). Configuradlas con el número de bits adecuado. Utilizad separadores para separar los bits necesarios (*Wiring/Separador*) y conectadlos donde correspondan. Utilizad túneles (*Wiring/Tunnel*) para conectar señales a través de nombres. Cambiad la apariencia del encapsulado para facilitar su uso posterior.
- i) **Comprobad que todo funciona correctamente realizando varias lecturas y escrituras.** Para ello, debéis cambiar a modo simulación  y realizar varias escrituras y lecturas en distintos registros. Para visualizar el valor de cada señal podéis añadir componentes Ver (*Wiring/Ver*) o directamente hacer clic en la señal. También podéis hacer doble clic en un **BR4** para entrar en él y comprobar el contenido de los registros.

3.3 ANÁLISIS TEMPORAL: RESULTADOS

1. Realizad un análisis temporal de los bloques combinacionales (MUX 4:1, MUX 8:1, DECOD 2:4, DECOD 3:8), calculando el retardo de los caminos entre todas las entradas y salidas de cada bloque. Asumir una implementación siguiendo la expresión $Z = X_0 * \overline{S_1} * \overline{S_0} + X_1 * \overline{S_1} * S_0 + \dots$ para los MUX y $Z_0 = en * \overline{S_1} * \overline{S_0}$; $Z_1 = en * \overline{S_1} * S_0$; ... para los DECOD. Los retardos de las puertas utilizadas son: $d_{NOT} = 8ps$; $d_{OR2-4} = 25ps$; $d_{AND2-4} = 25ps$; $d_{REG} = 60ps$; $t_{setup_{REG}} = 35ps$. Tened en cuenta que disponemos de puertas OR de un máximo de 4 entradas.
2. Identificad todos los caminos combinacionales que intervienen en la **lectura** del **BR32** ($RA \rightarrow busA$ y $REG \rightarrow busA$) y calculad sus retardos ($d_{RA \rightarrow busA}$, $d_{REG \rightarrow busA}$). Debéis tener en cuenta que los bits de RA tienen distintos caminos, y por tanto, deberéis obtener el retardo máximo de todos ellos.
3. Identificad los caminos combinacionales que intervienen en la **escritura** del **BR32** ($RW \rightarrow REG$, $busW \rightarrow REG$, $RegWr \rightarrow REG$) y calculad sus retardos. Teniéndolos en cuenta, ¿con qué antelación al flanco de reloj tiene que establecerse el valor de cada entrada ($t_{setup_{RW}}$, $t_{setup_{busW}}$, $t_{setup_{RegWr}}$) para que se escriba correctamente? ¿Cuál podríamos decir que sería el tiempo de setup del BR32?

3.4 MATERIAL A ENTREGAR

Deberéis entregar todo lo realizado en los apartados 3.1, 3.2 y 3.3 (incluyendo los circuitos implementados y un documento/memoria de la práctica), dentro del plazo establecido. La fecha límite de entrega se indicará próximamente en el apartado Avisos de Moodle. Para que os podáis planificar, la entrega se deberá realizar aproximadamente dentro de 3 semanas.

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 un fichero comprimido **en formato ZIP** con los siguientes documentos:

1. Memoria en formato PDF
2. Ficheros con los circuitos implementados en los diferentes apartados

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

P2_NIP-Apellidos_Estudiente1_NIP-Apellidos_Estudiente2.zip

Por ejemplo: P2_345456-Gracia_Esteban_45632-Arribas_Murillo.zip