

# Practica 4: Implementación del microprocesador ICAI-RISC-16

## 1. Objetivos

El objetivo de esta práctica es la implantación del microprocesador ICAI-RISC-16 descrito en el documento: “Arquitectura del microprocesador ICAI-RISC-16”.

## 2. Tiempo de laboratorio

3 sesiones.

## 3. Memoria

Se entregarán los esquemas y el código VHDL de todos los componentes del diseño.

## 4. Introducción

En esta práctica se va a implementar un microprocesador sencillo de 16 bits en la FPGA del laboratorio.

Como en todo diseño complejo, es necesario dividir el circuito en bloques más sencillos para que el diseño sea abordable. La primera división consiste en separar la ruta de datos del circuito de control.

La ruta de datos se muestra en la figura 1 y como se puede apreciar está formada por una ALU, un banco de registros, los registros PC, IR y salida de la ALU, las memorias de datos y de programa, y una serie de multiplexores que controlan el flujo de datos entre estas unidades.

Por otro lado, el circuito de control será una máquina de estados cuyo diagrama se muestra en la figura 2.

## 5. Diseño de la ruta de datos

El diseño de la ruta de datos se abordará bloque a bloque. La ALU ha sido ya diseñada en la práctica anterior, por lo que no tiene más que copiarla. El resto de bloques se diseñarán en VHDL, salvo la ROM, que es generada automáticamente por el programa ensamblador. La memoria RAM se creará de 64 palabras, aunque es conveniente crearla de forma genérica para que sea fácilmente modificable. Para diseñar estos bloques se seguirán las ideas expuestas en la clase de teoría.

Por último, destacar que cada uno de los bloques diseñados ha de simularse por separado.

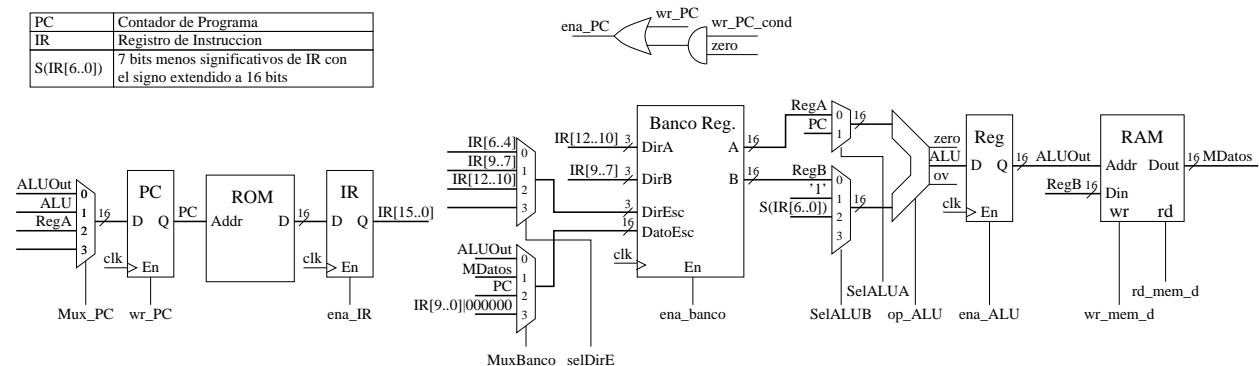


Figura 1: Diagrama de bloques de la ruta de datos del ICAI-RISC-16

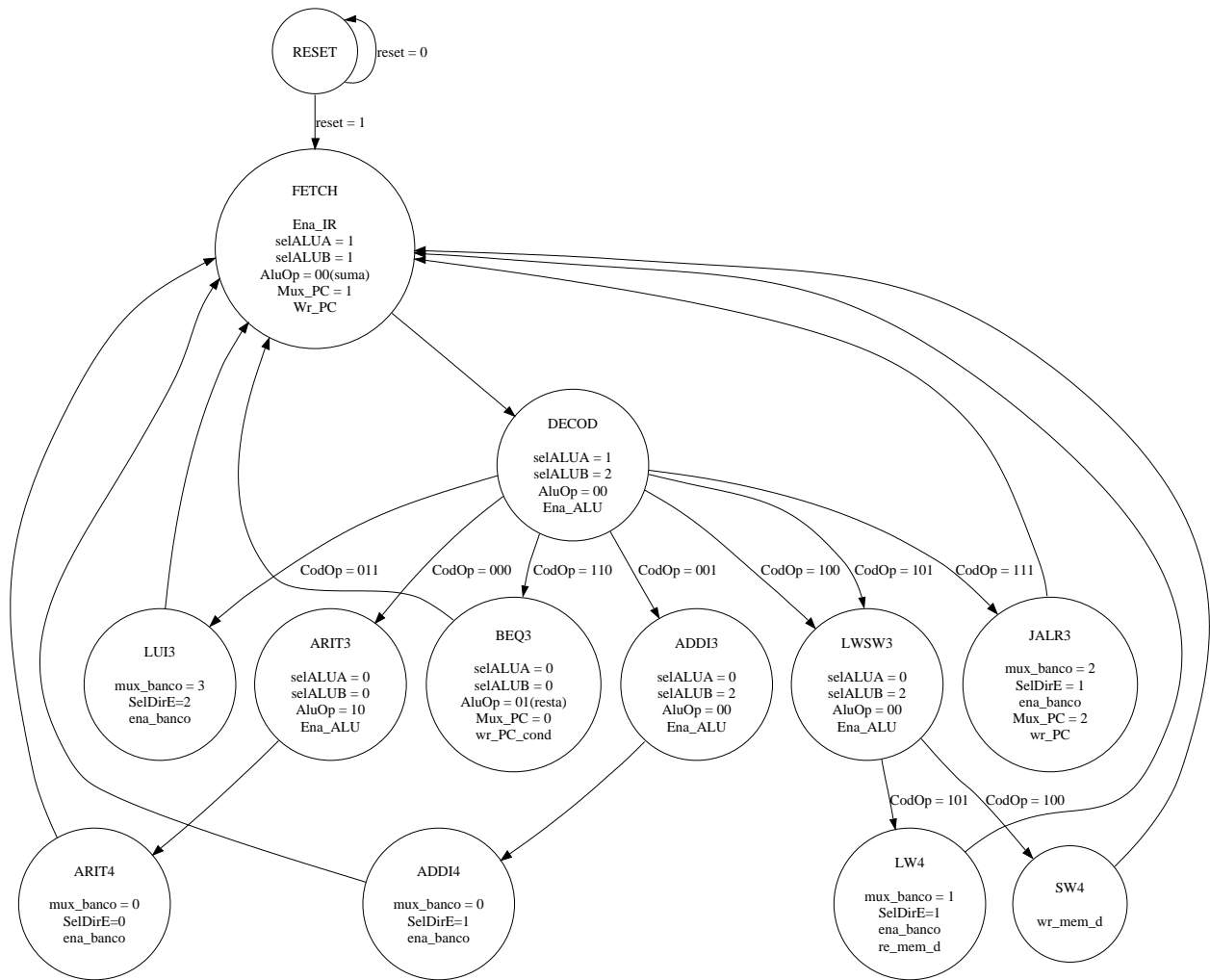


Figura 2: Diagrama de estados del circuito de control de la CPU.

## 6. Diseño del circuito de control

Como se ha dicho en la introducción, el circuito de control será una máquina de estados, cuyo diagrama de estados se ha mostrado en la figura 2.

## 7. Entrada/Salida

Para realizar una prueba en la tarjeta, es conveniente añadir un periférico de entrada/salida a la arquitectura. Para ello, puesto que sólo se han usado las 64 palabras más bajas de la memoria, puede dividirse el mapa de memoria en dos mitades, usándose las posiciones bajas para acceder a la RAM y las altas (0x8000 a 0xFFFF) para acceder a los periféricos de entrada/salida. De esta forma si el bit 15 del bus de direcciones vale 0 se accederá a la RAM y si vale 1 a los periféricos de entrada/salida.

Para no complicar mucho el sistema, se usará un solo puerto de entrada/salida. Como entradas se usarán los 10 interruptores y como salidas los 10 LEDs rojos (en ambos casos se conectarán los 10 bits menos significativos del puerto). Cualquier lectura a una posición de memoria comprendida entre 0x8000 y 0xFFFF tomará el dato de los interruptores y cualquier escritura comprendida entre esas direcciones guardará el dato en un registro cuya salida estará conectada a los LEDs.

El programa de prueba consistirá simplemente en leer el puerto de entrada y mostrar su valor en los LEDs.