

# Organización y Arquitectura de Computadoras 2017-2

## Práctica 4: Lógica Secuencial

Profesor: José de Jesús Galaviz Casas  
Ayudante de laboratorio: Roberto Monroy Argumedo

Publicada: Marzo 6, 2017  
Límite de entrega: Marzo 11, 2017

### 1. Objetivos

#### Generales:

- El alumno aprenderá a analizar, diseñar y simular circuitos secuenciales.

#### Particulares:

Al finalizar la práctica, el alumno estará familiarizado con:

- El diseño de máquinas de estados finitos.
- El funcionamiento de un banco de registros y su interacción con la unidad aritmética lógica.

### 2. Requisitos

#### ■ Conocimientos previos:

- Minimización de funciones de conmutación por medio de manipulación algebraica y mapas de Karnaugh.
- Los componentes básicos del diseño de circuitos combinacionales: transistores y compuertas **AND**, **OR** y **NOT**.
- Conceptos de circuitos secuenciales: estado, memoria, reloj, etc.
- Funcionamiento de: biestables, latches y flip-flops.

Se pueden consultar los temas en [2] y [3].

- **Tiempo de realización sugerido:**  
5 horas.

- **Número de colaboradores:**  
Individual.
- **Software a utilizar:**
  - *Java Runtime Environment* versión 5 o superior, utilidad para ejecutar programas *Java*.
  - El paquete *Logisim* [1].

### 3. Planteamiento

Un **registro** es un circuito con la capacidad de almacenar datos, los cuales pueden ser leídos o escritos según sea necesario; son utilizados para guardar tanto operandos como resultados de las operaciones realizadas por el procesador, por lo que es común contar con un arreglo de registros junto con puertos dedicados para lectura y escritura, a esta unidad se le conoce como **banco de registros** o *register file*. Por la naturaleza de estos componentes, la implementación puede llevarse a cabo por medio de circuitos secuenciales. Nuestra tarea para esta práctica será simular un banco de registros, el cual interconectaremos con la ALU de 8 bits elaborada en la práctica anterior.

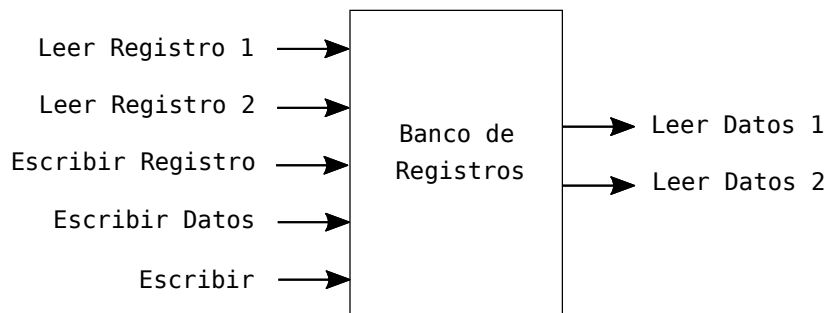


Figura 1: Banco de registros.

El banco de registros, ilustrado en la figura 1, contará con:

- Cuatro registros de 8 bits.
- Dos puertos de lectura. El banco de registros debe incluir dos entradas: **Leer registro 1** y **Leer registro 2** para indicar los registros que serán consultados y dos salidas: **Leer Datos 1** y **Leer Datos 2** en donde se colocarán los datos almacenados en los registros indicados.

- Un puerto de escritura. Para realizar la escritura de datos, se contará con tres entradas: **Escribir registro**, para indicar el registro en el que escribirán los datos; **Escribir Datos**, en donde se colocarán los datos a escribir; y **Escribir**, cuando esta señal de entrada cambie a 1, los datos serán escritos en el registro indicado.

## 4. Desarrollo

El diseño de circuitos secuenciales es similar al diseño de circuitos combinatoriales, lo único distinto es que debemos considerar el estado en el que se encuentra el circuito para calcular el siguiente estado. Recordando el proceso de diseño de las paráticas anteriores, el primer paso es hallar las variables de entrada para el circuito, como es necesario considerar el estado del circuito para calcular el siguiente, agregaremos las variables necesarias para representar el estado actual del circuito. A las funciones de salida le agregaremos las que representen el estado al que debe cambiar el circuito. El resto del proceso continúa intacto hasta el momento de la simulación. El diseño que hemos obtenido hasta ahora es el de un circuito combinatorial, agregando elementos de memoria para guardar el estado y un reloj para controlar la transición de estados, convertirá a nuestro circuito en uno secuencial, como se muestra en la figura 2. Los componentes que usaremos como memoria serán flip-flops.

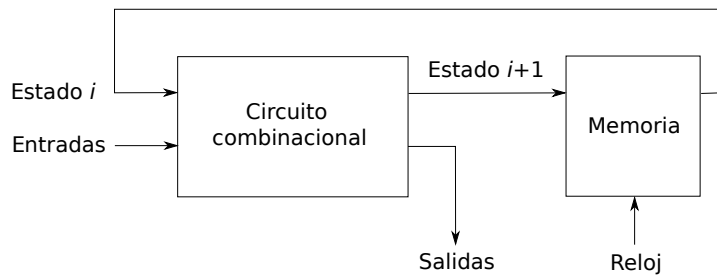


Figura 2: Estructura de un circuito secuencial.

## 5. Entrada

El circuito principal a desarrollarse en esta práctica, además de otros ejercicios, es la interconexión entre una ALU de 8 bits y el banco de registros, dicho circuito recibirá como entrada de 18 bits, la entrada representará una operación que deberá llevar a cabo el circuito. La decodificación de la operación será como se muestra en la tabla 1, comenzando desde el bit más significativo.

Los códigos de operación de la ALU son los mismos que la práctica anterior.

No. de bits	Uso
2	Código de operación
2	Número de registro para el primer operando
1	Selector de segundo operando: 0 para registro, 1 para inmediato
2	Número de registro para el segundo operando
8	Inmediato entero para el segundo operando
2	Número de registro en el que se almacenará el resultado

Tabla 1: Decodificación de la instrucción.

## 6. Salida

La salida del circuito será el resultado de la operación realizada.

## 7. Variables libres

No hay variables libres para el desarrollo de esta práctica.

## 8. Procedimiento

Deberás entregar un solo archivo de *Logisim* con las soluciones de los ejercicios y un documento con las respuesta a las preguntas planteadas. Solamente puedes hacer uso de compuertas lógicas **AND**, **OR** y **NOT**; flip-flops tipo D, JK y SR; multiplexores; separadores; y pines de entrada y salida. Recuerda etiquetar las entradas y salidas de cada uno de los subcircuitos.

## 9. Ejercicios

- Usando flip-flops tipo JK, diseña un circuito secuencial que realice un conteo de 0 a 9, el conteo deberá incrementar cuando el pulso del reloj cambie a 1. El estado posterior al estado 9 será el estado inicial 0, reiniciando así el conteo. La salida del circuito será la representación binaria del estado.
  - Conecta la salida del circuito a un display para visualizar el conteo.
  - Diseña un nuevo circuito en donde, utilizando el circuito anterior, el conteo sea de 0 a 999.
- Diseña un circuito secuencial que simule el autómata de la figura 3. Para guardar el estado, deberás usar flip-flops tipo D.
- Diseña un circuito para simular un registro, sólo puedes usar flip-flops tipo SR.
- Diseña un circuito para simular el banco de registros descrito en la sección 3. Utiliza el registro diseñado en el ejercicio anterior.

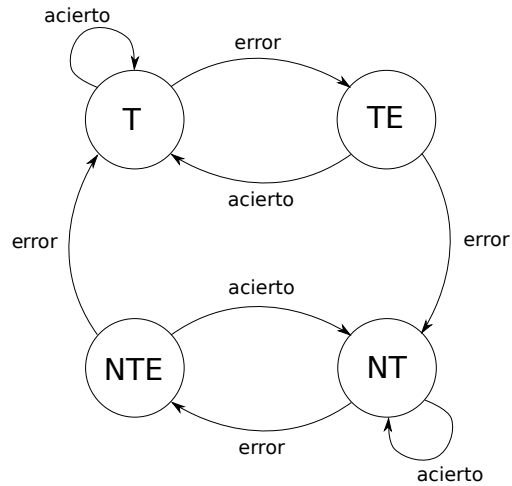


Figura 3: Autómata.

5. Interconecta la ALU de 8 bit elaborada en la práctica anterior con el banco de registros del ejercicio anterior como se muestra en la figura 4 y de acuerdo a las especificaciones de las secciones 5 y 6.

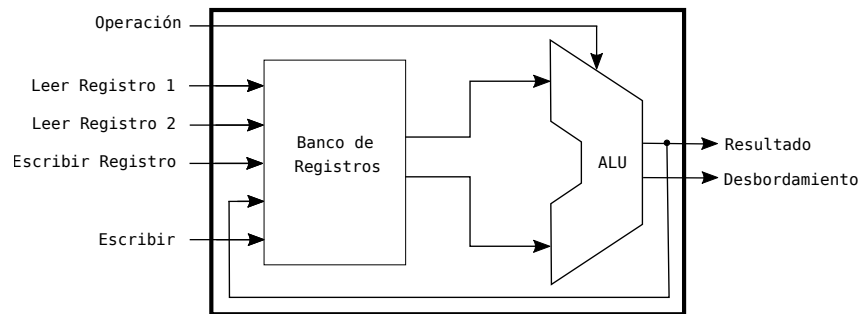


Figura 4: Interconexión de la ALU con el banco de registros.

## 10. Preguntas

1. ¿En qué difieren los distintos tipos de flip-flops? ¿Cómo se decide qué tipo se usará en el circuito?
2. Si el conteo del ejercicio 1 fuera de 0 a 15, ¿cómo harías el circuito?

3. Un **registro de desplazamiento** es un circuito secuencial que desplaza a la izquierda o a la derecha la información contenida en él. Considerando el desplazamiento de 1 bit a la izquierda, ¿cómo se implementa dicho circuito? ¿Cómo podríamos simular su funcionamiento con las operaciones que se tienen en la ALU de 8 bits?

## 11. Bibliografía

- [1] Carl Burch. *Logisim*. Consultado: 18 de enero, 2016. Publicado: 2014. URL: <http://www.cburch.com/logisim/>.
- [2] M. Morris Mano. *Digital Logic and Computer Design*. 1st. Prentice Hall PTR, 1979.
- [3] David A. Patterson y John L. Hennessy. *Computer Organization and Design, Fifth Edition: The Hardware/Software Interface*. 5th. Morgan Kaufmann Publishers Inc., 2013.