

# Circuitos Lógicos: SDI-11322

## Práctica 5. Registros

Departamento Académico de Sistemas Digitales  
Instituto Tecnológico Autónomo de México

Otoño 2018

### 1. Objetivos

- Realizar la implementación de un *Register File* utilizando VHDL
- Utilizar procesos en VHDL para implementar circuitos secuenciales
- Utilizar una palabra de control para manipulación de registros.

### 2. Problema

Implementar el circuito mostrado en la Fig. 1. Consiste de la unidad funcional implementada en prácticas anteriores y un *Register File*. La palabra de control se muestra en la parte inferior de la imagen. En la palabra de control se muestran la cantidad de bits que tiene cada señal. ¿Cuántos registros tiene el *Register File*?

### 3. Esbozo de solución

Tres de las entradas del *Register File* son direcciones de memoria. *A address* determina el valor que se va leer en *A data* y de manera similar, *B address* determina el valor de *B data*. La señal *D address* determina la dirección donde se va a escribir el valor de *D data* si el bit *RW* es uno. Todas estas operaciones ocurren en el mismo ciclo de reloj. Adicionalmente, el *Register File* cuenta con una señal de *Reset* que al activarse, establece todos los valores de los registros a  $0x0000$ . El componente *Hexadecimal Display* es el circuito de la práctica 2.

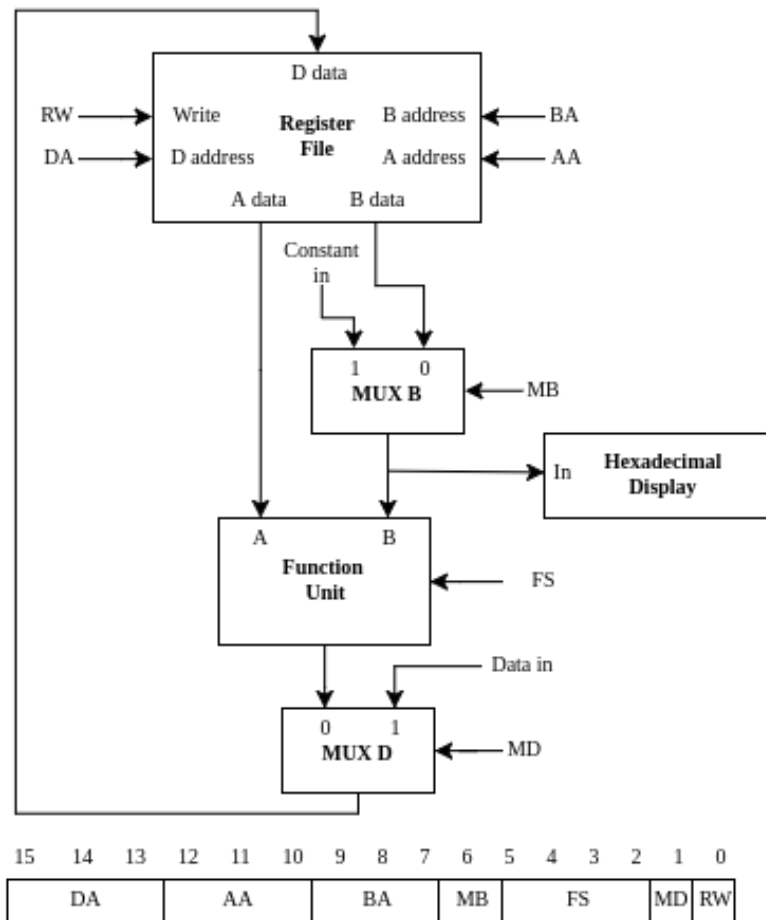


Figura 1: Esquema del circuito a implementar

## 4. Aspectos técnicos de implementación

### 4.1. Circuitos Secuenciales

Para implementar circuitos secuenciales en VHDL, se utilizan *procesos*. A continuación se muestra la sintaxis básica para utilizar procesos. La lista que sigue a la declaración del proceso define las señales que causan que el proceso se vuelva a evaluar.

```

1 process (b, c)
2 begin
3   a<=b xor c;
4 end process;
```

En el siguiente ejemplo se muestra cómo utilizar un proceso utilizando la señal de reloj.

```
1 PROCESS (CLK)
2 begin
3     if (CLK'event AND CLK = '1') then
4         (...)
5     else
6         (...)
7     end if;
8 end process;
```

## 4.2. Top Level Entity

- Conviene asignar la palabra de control de la siguiente manera:
  - El bit menos significativos a un boton de la tarjeta (recordar que están en lógica negada).
  - El resto del byte menos significativo a un dip switch.
  - El byte más significativo a un dip switch.
  - Asignar la señal de *Reset* a un botón de la tarjeta.
- Para la constante que entra en el multiplexor B, utilizar 0xABCD.
- Para la señal *data in* que entrada al multiplexor D, utilizar 0x1123.

## 5. Validación

Es conveniente simular el funcionamiento del *Register File* previo a realizar la implementación completa del circuito.

Una vez implementado todo el circuito y haber programado la FPGA, determinar y probar los valores de la palabra de control necesarios para realizar la siguiente secuencia:

```
R0 ← constant_in
R1 ← data_in
R2 ← R0 + R1
R3 ← R0 ⊕ R1
R4 ← R2 | R3
R5 ← slR3
R6 ←  $\overline{R3}$ 
R7 ← R3 ⊕ R1
```