

Circuitos Lógicos: SDI-11322

Práctica 4. Circuito Lógico y Unidad Funcional

Departamento Académico de Sistemas Digitales
Instituto Tecnológico Autónomo de México

Otoño 2018

1. Objetivos

- Reforzar el uso de VHDL
- Realizar la implementación de un circuito lógico y un shifter
- Implementar la unidad funcional de la computadora

2. Problema

Se requiere implementar la unidad funcional de la computadora, la cual esta conformada de los siguientes componentes:

1. Un circuito aritmético (práctica 3)
2. Un circuito lógico
3. Un circuito *shifter*

3. Esbozo de solución

3.1. Circuito Lógico

Se requiere un circuito que implemente la tabla 1.
Las entradas del circuito son:

- S - Vector de selección de la operación
- A - Número de 16 bits
- B - Número de 16 bits

El circuito entrega una salida G de 16 bits

S_1	S_0	Output	Operation
0	0	$G=A\&B$	AND
0	1	$G=A B$	OR
1	0	$G=A\oplus B$	XOR
1	1	$G=\overline{A}$	NOT

Tabla 1: Tabla de un Circuito Lógico

S_1	S_0	Output	Operation
0	0	$G=B$	
0	1	$G=srB$	Shift Right
1	0	$G=slB$	Shift Left

Tabla 2: Tabla del Circuito Shifter

3.2. Shifter

Se requiere un circuito que implemente la tabla 2.
Las entradas del circuito son:

- S - Vector de selección de la operación
- B - Número de 16 bits

Se obtiene una salida G de 16 bits.

3.3. Unidad Funcional

El diagrama de la unidad funcional se muestra en la Fig. 1. La tabla de las operaciones implementadas se muestra en la tabla 3.

Entradas:

- FS - vector de selección.
- A - Número de 16 bits.
- B - Número de 16 bits.

Salidas:

- F - Vector de salida.
- C - bit de *carry out*.
- N - bit que indica si el resultado es un número negativo (MSB de F = 1)
- Z - bit que indica si el resultado es 0x0000.

3.3.1. Zero Detect

Se requiere determinar si todos los bits del resultado F son cero. Implementar la lógica necesaria para esto (no tiene que ser en otro archivo necesariamente).

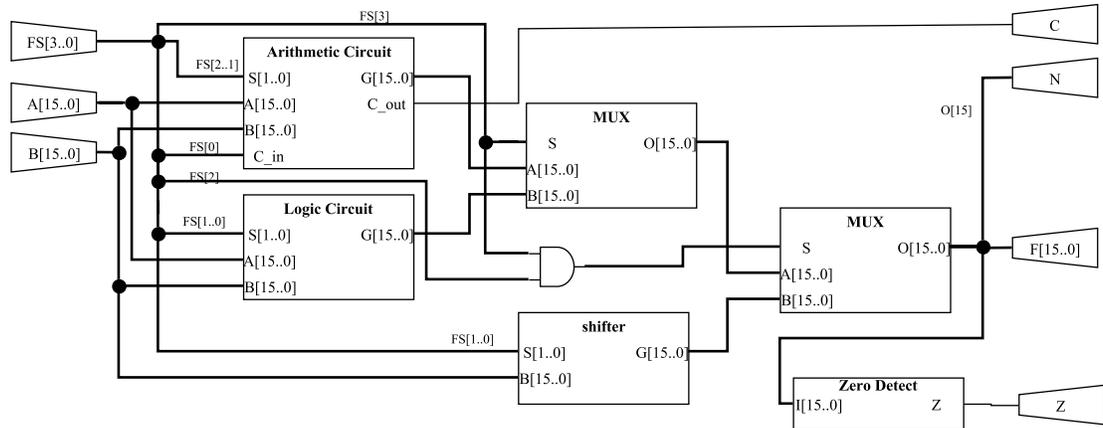


Figura 1: Esquema del circuito a implementar

3.3.2. Multiplexores

Anteriormente se habían utilizado multiplexores 4 a 1, ahora se requieren dos multiplexores 2 a 1. Implementar la lógica necesaria para realizar ésta operación (no tiene que implementarse en otro archivo necesariamente).

3.4. Diagrama general

Éste es el archivo de mayor jerarquía de ésta práctica (Top Level Entity). Se requiere conectar la unidad funcional con el circuito implementado en la práctica 2 y el circuito *binary_to_bcd* para poder visualizar el resultado de las operaciones en los displays de siete segmentos. La salida F de la unidad funcional debe ser la entrada del circuito *binary_to_bcd* y la salida de ésta ir a las entradas pertinentes del circuito de la práctica 2.

Entradas:

- clk - reloj de la FPGA (PIN_R8)
- FS - vector de selección. Asignarlo al dip switch de la tarjeta
- A - Número de 16 bits. Asignar el LSB (less significant byte) a un dip switch, el MSB (most significant byte) siempre valdrá 0x00.
- B - Número de 16 bits. Asignar el LSB a un dip switch, el MSB siempre valdrá 0x00.

Salidas:

- display - vector de 7 bits asociado a los displays de siete segmentos.
- D - vector de 4 bits para activar cada display.

FS[3..0]	Micro-operación
0000	$F=A$
0001	$F=A + 1$
0010	$F=A + B$
0011	$F=A + B + 1$
0100	$F=A + \overline{B}$
0101	$F=A + \overline{B} + 1$
0110	$F=A - 1$
0111	$F=A$
1000	$F=A \& B$
1001	$F=A B$
1010	$F=A \oplus B$
1011	$F=\overline{A}$
1100	$F=B$
1101	$F=srB$
1110	$F=slB$

Tabla 3: Micro-operaciones de la Unidad Funcional

- C - bit de carry, asignarlo a un led de la tarjeta.
- N - bit que indica si el número de salida F es negativo, asignarlo a un led de la tarjeta.
- Z - bit que indica si el número de salida F es igual a cero, asignarlo a un led de la tarjeta.

4. Validación

Es conveniente realizar simulaciones del circuito lógico y el shifter previo a realizar la implementación de la unidad funcional. Igualmente, tras implementar la unidad funcional es conveniente realizar una simulación de su funcionamiento.

Mostrar el circuito funcionando con los displays de 7 segmentos y utilizando los dip switches.