

# Electrónica Digital

## Guía de Trabajos Prácticos N° 4

### *Circuitos aritméticos y lógicos*

**Nota:** Para la realización de esta guía, tendremos en cuenta las siguientes definiciones:

- **Diseño** Son todos los pasos para llegar a un circuito final optimizado en papel.
- **Simulación** Se refiere a describir el circuito diseñado, mediante Verilog; para su análisis temporal en el programa GTKWave.
- **Síntesis** Constituye la transformación del código diseñado en el hardware de la FPGA.
- **Implementación** refiere a impactar el modelo sintetizado en la placa prototipo provista por la cátedra.

1. Diseñe los siguientes circuitos combinacionales:
  - a. Sumador completo de un bit.
  - b. Restador completo de un bit.
  - c. Integre los circuitos anteriores con las operaciones lógicas para construir una Unidad Aritmético Lógica (ALU) de un bit. La ALU tendrá como entrada los dos operandos y como salida el resultado y la banderas de acarreo para suma (C) y resta (B), cero (Z) y paridad (P) y podrá realizar las operaciones listadas en la siguiente tabla de:

**Tabla 1:** Operaciones realizadas por la ALU y sus códigos

Código	Operación
001	Suma
010	Resta
100	Suma lógica
101	Producto lógico
110	Complemento
***	No operación

Analice y compruebe su correcto funcionamiento mediante simulación de un testbench. Impacte los circuitos obtenidos en la FPGA.

2. Diseñe un sumador y restador de 4 bits utilizando solo sumadores. Analice su funcionamiento y compruebe su correcto funcionamiento mediante simulación de un testbench.
3. Diseñe, simule e implemente un circuito que compara dos números **A** y **B** de 4 bits y provea como salida tres señales:
  - a. Una que indique si **A > B**;
  - b. Una que indique si **A = B**;
  - c. Una que indique si **A < B**.

Compare la solución obtenida con la hoja de datos del circuito integrado **SN5485**. Analice su funcionamiento y compruebe su correcto funcionamiento mediante simulación de un

testbench. Impacte los circuitos obtenidos en la FPGA.

4. Diseñe e implemente un sumador de 4 bits con acarreo anticipado. Compare la solución obtenida para el generador del acarreo anticipado con aquella provista por la hoja de datos del circuito integrado **CD40182**. Analice y compruebe su correcto funcionamiento mediante simulación de un testbench. Impacte los circuitos obtenidos en la FPGA.
5. Diseñe e implemente un barrel shifter de 4 bits que implemente desplazamientos lógicos y aritméticos, así como rotaciones en ambas direcciones (derecha o izquierda).

**Tabla 1:** Operaciones y sus códigos

Código	Operación
000	No operación
001	Desplazamiento lógico a derecha
010	Desplazamiento lógico a izquierda
101	Desplazamiento aritmético a derecha
110	Desplazamiento aritmético a izquierda
011	Rotación a derecha
111	Rotación a izquierda

Analice y compruebe su correcto funcionamiento mediante simulación de un testbench. Impacte los circuitos obtenidos en la FPGA.

6. Diseñe e implemente una Unidad Aritmética Lógica (ALU) que pueda conectarse en paralelo con otras para expandir el ancho de palabra (bit slicing), capaz de realizar las operaciones listadas en la tabla 2 con operandos de 4 bits.

**Tabla 2:** Operaciones realizadas por la ALU y sus códigos

Instrucción	Función
0001	AND (A & B)
0010	OR (A   B)
0011	XOR (A⊕B)
0101	Suma (A+B)
0110	Resta (A-B)
1001	Comparación <i>menor que</i> (A<B)
1010	Comparación <i>igual que</i> (A=B)
1101	Desplazar un bit a la derecha
1110	Desplazar un bit a la izquierda