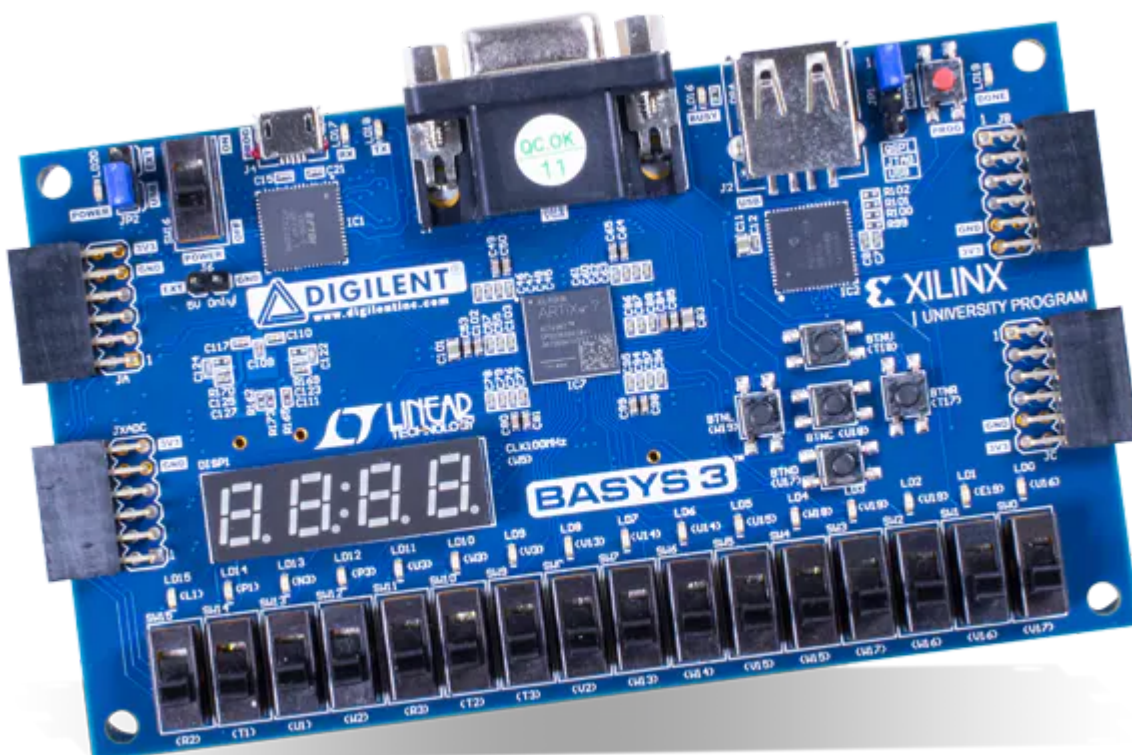


# Arquitectura de Computadoras

## Trabajo Práctico 1

### Unidad Aritmética Lógica



#### Alumno:

- Mirada Rodrigo: 41106889
- Worley Agustina: 39071465

# Índice

	Pág.
<b>Objetivos</b>	<b>3</b>
<b>Desarrollo</b>	<b>4</b>
Descripción	4
Interfaces	4
Entradas	4
Salidas	4
ALU	5
Entradas	5
Salidas	5
Top	5
Entradas	5
Salidas	5
Conectores	5
Esquema	5
Test Aleatorios	6
Análisis	6
Utilización	6
Consumo	6

## Objetivos

- Implementar en FPGA una ALU.
- La ALU debe ser parametrizable (bus de datos) para poder ser utilizada posteriormente en el trabajo final.
- Validar el desarrollo por medio de Test Bench.
  - El testbench debe incluir generación de entradas aleatorias y código de chequeo automático.
- Simular el diseño usando las herramientas de simulación de vivado incluyendo análisis de tiempo.

Operación	Código
ADD	100000
SUB	100010
OR	100100
XOR	100101
SRA	100110
SRL	000010
NOR	100111

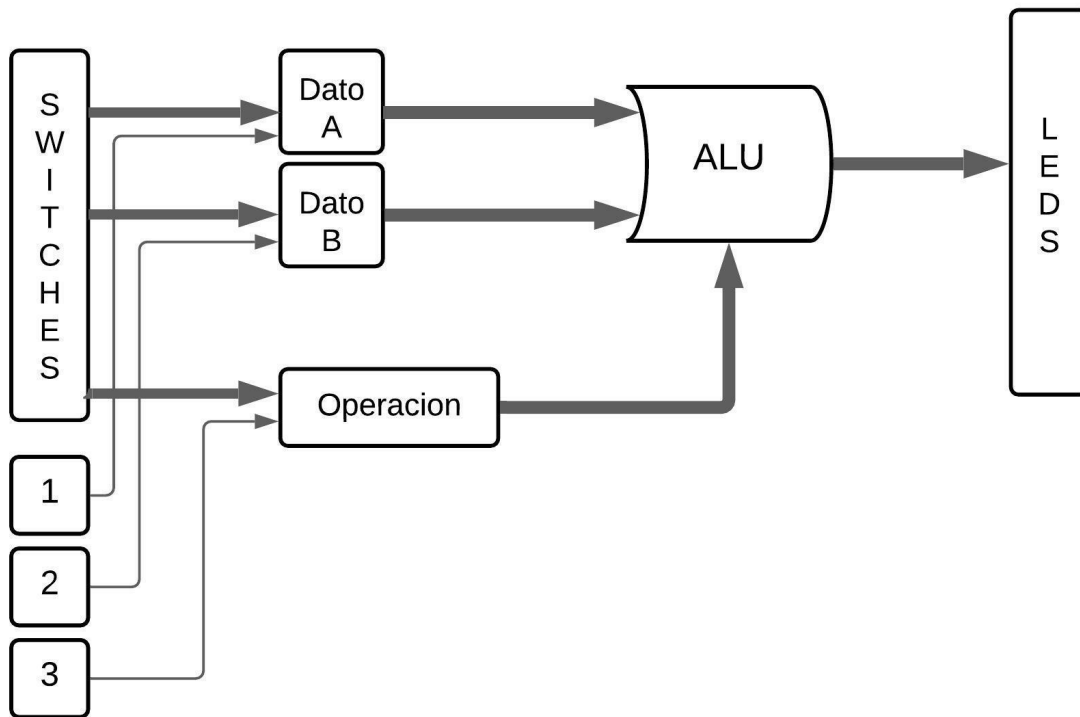
## Desarrollo

### Descripción

Tenemos un conjunto de N switches y un set de tres botones que en los casos:

- Se presiona el botón 1: El valor de los switches se inserta en el dato A.
- Se presiona el botón 2: El valor de los switches se inserta en el dato B.
- Se presiona el botón 3: El valor de los switches se inserta en el dato Operación.

Estos datos van ingresan en la ALU para que dependiendo del dato Operación se obtenga la salida LEDS.



### Interfaces

#### Entradas

Definimos cuatro entradas:

- input\_switches: encargada de tomar el dato de los switches.
- input\_buttons: encargada de tomar el dato de los botones.
- input\_clock.
- input\_reset.

#### Salidas

Definimos tres salidas

- output\_a: contiene la información de los switches seleccionados para el dato A.
- output\_b: contiene la información de los switches seleccionados para el dato B.
- output\_op: contiene la información de los switches seleccionados para la operación a realizar.

## ALU

### Entradas

Definimos tres entradas:

- date\_a: dato de ingreso a la ALU referido a A.
- date\_b: dato de ingreso a la ALU referido a B.
- op: dato de ingreso a la ALU referido a la operación a realizar.

### Salidas

Definimos una salida:

- result: dato de salida tras la operación dentro de la ALU.

## Top

### Entradas

Definimos cuatro entradas:

- i\_switches: información a mandar a las interfaces como switches.
- i\_buttons: información a mandar a las interfaces como botones.
- i\_clock: información a mandar a las interfaces como señal de clock.
- i\_reset: información a mandar a las interfaces como botón de reset.

### Salidas

Definimos una salida:

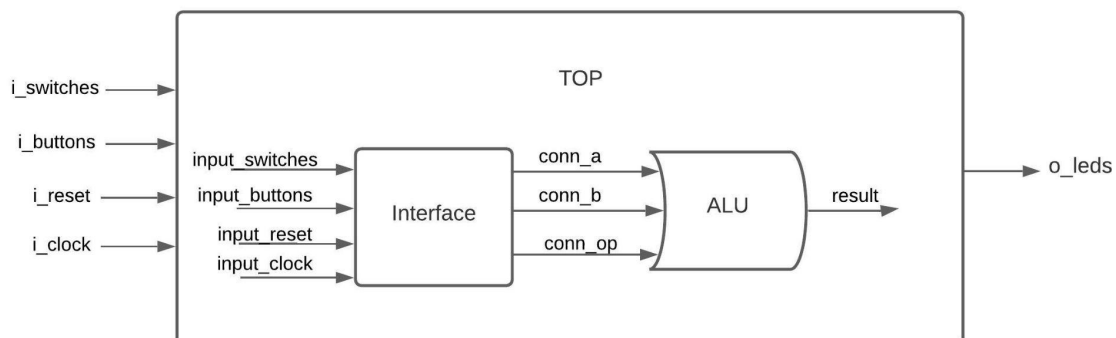
- o\_leds: información de salida del conjunto interfaces + ALU.

## Conectores

Definimos tres conectores:

- conn\_a: bus de cables de conexión entre la salida output\_a de la interfaz y la entrada date\_a de la ALU.
- conn\_b: bus de cables de conexión entre la salida output\_b de la interfaz y la entrada date\_b de la ALU.
- conn\_op: bus de cables de conexión entre la salida output\_op de la interfaz y la entrada op de la ALU.

## Esquema



## Test Aleatorios

Para los tests aleatorios se van a realizar una cantidad “test\_amout” por cada operación de manera que podamos escoger la cantidad de pruebas por operación.

En caso de ser correcto el resultado en la consola no se notificará nada, pero en caso de ser incorrecto saldrá el mensaje “Error en la operación XXX” y a posterior la operación con los valores que fallaron.

## Análisis

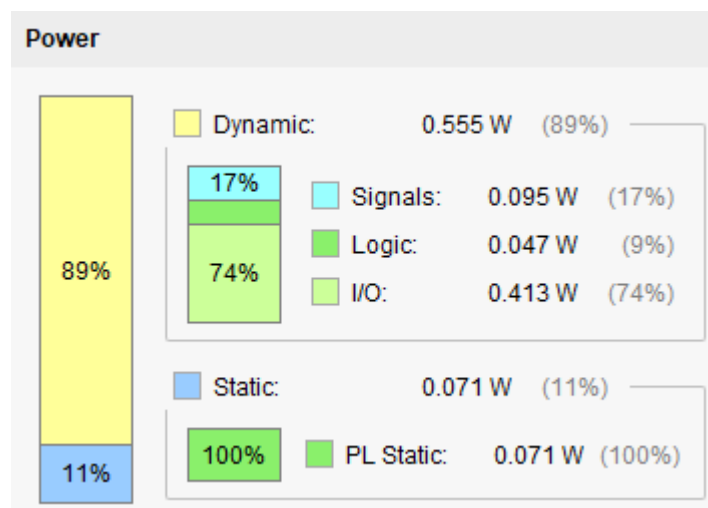
### Utilización

Luego del proceso de implementación quedamos con los siguientes datos:

Utilization			
		Post-Synthesis	Post-Implementation
		Graph   Table	
Resource	Utilization	Available	Utilization %
LUT	63	20800	0.30
FF	22	41600	0.05
IO	20	106	18.87
BUFG	1	32	3.13

### Consumo

Análisis de consumo:



Como se puede observar en el gráfico el consumo total es:

- Entradas y salidas: 66% del total.
- Señales: 15% del total.
- Lógica: 8% del total.
- Consumo básico: 11% del total.