

## Diseño Digital Avanzado

### Unidad 5 - Pipelining, Retiming y Unfolding

**Dr. Ariel L. Pola**

[apola@fundacionfulgor.org.ar](mailto:apola@fundacionfulgor.org.ar)

November 15, 2021

# Tabla de Contenidos

1. Pipeling y Retiming

2. Paralelismo y Unfolding

# Pipelining y Retiming



# Pipelining y Retiming

## Objetivo

- El objetivo es evaluar la complejidad y el *timing* del sistema implementando las técnicas de reducción de complejidad Pipelining y Retiming.
- En todos los casos obtener el reporte de Celdas y el peor camino crítico.
- Justificar los resultados en un informe.

## Pasos 1

- Generar un nuevo proyecto instanciando los archivos entregados por la cátedra.
- Correr la implementación (**Run Implementation**).

# Pipelining y Retiming

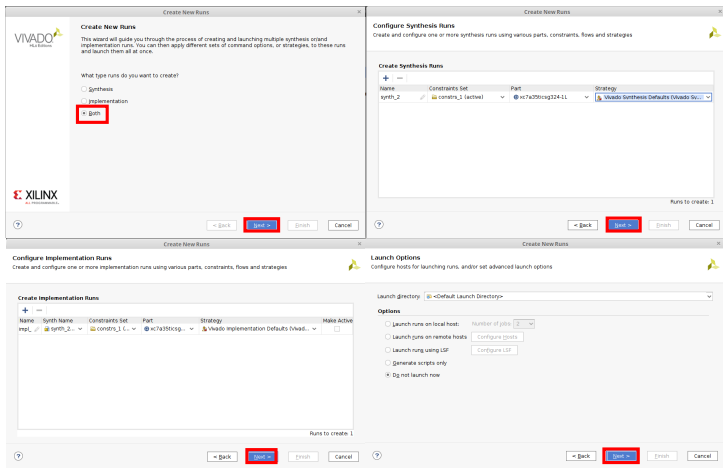
## Pasos 2

- En el archivo **fir.v** comentar la definición CASE1.
- Crear un nuevo entorno de trabajo, marcarlo como activo y correr de nuevo la implementación. (Ver figuras)

Tcl Console	Messages	Log	Reports	Design Runs	Power	DRC	Methodology	Timing	? _ □ □		
Q   Z   S   I   L   R   +   %											
Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	
✓ synth_1 (active)	constrs_1	synth_design Complete!								1007	
✓ impl_1 (active)	constrs_1	route_design Complete!	1.170	0.000	0.147	0.000	0.000	0.066	0	943	
✓ synth_2	constrs_1	Synthesis Out-of-date								838	
✓ impl_2	constrs_1	Implementation Out-of-date	0.389	0.000	0.154	0.000	0.000	0.065	0	758	

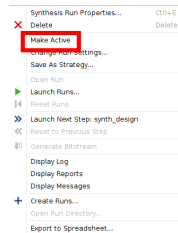
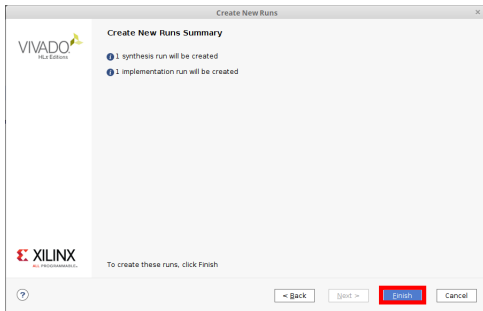
Paso 2 - a

# Pipelining y Retiming



Paso 2 - b

# Pipelining y Retiming



Click Derecho sobre el nuevo entorno de Síntesis

Paso 2 - c

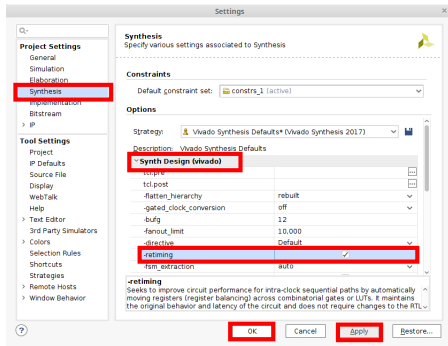
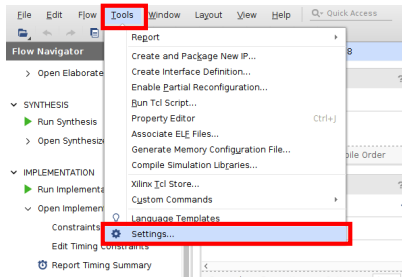
# Pipelining y Retiming

## Pasos 3

- Volver al código original del archivo **fir.v** descomentando la definición CASE1.
- Crear un nuevo entorno de trabajo
- Marcarlo como activo
- Asignar el método **Retiming** en la configuración de la síntesis. **Tools - Settings - Synthesis - Retiming** (Ver figuras)
- Correr de nuevo la implementación.



# Pipelining y Retiming



Paso 3

# Pipelining y Retiming

## Pasos 4

- Por último, crear un nuevo entorno de síntesis.
- Verificar que no esté asignado el parámetro de Retiming en el método de síntesis.
- Comentar la definición CASE2.
- Correr de nuevo la implementación.

# Paralelismo y Unfolding



# Paralelismo y Unfolding

## Filtro FIR Paralelo

- El objetivo es comparar la complejidad y el timing entre un filtro FIR Directo con paralelismo 4 y un filtro transpuesto Unfolding de orden 4.
- Pasos:
  - Generar un proyecto nuevo.
  - Diseñar e Implementar un filtro FIR paralelo.
  - Diseñar e Implementar un filtro FIR transpuesto utilizando la técnica unfolding de orden 4.
  - En ambos casos proponer un TestBench para verificar la conexiones.
  - En base a los ejercicios anteriores armar un constraint que seleccione algunos pines de la FPGA y configurar el pin de clock.
  - Sintetizar los diseños a 100MHz.
  - Obtener los reportes de celdas y camino crítico para ambos casos.
  - En caso de encontrar problemas de timing aplicar alguna de las técnicas vistas anteriormente para resolverlo (pipeline o retiming).

# Paralelismo y Unfolding

## Filtro FIR - Parámetros de diseño

- Paralelismo de entrada 4.
- Paralelismo de salida 4.
- Resolución:  $X$  S(8,7),  $h$  S(8,7) e  $Y$  determinada por el diseñador.
- El filtro FIR paralelo debe incluir un **Regressor** para sincronizar las muestra de entrada.
- En el caso del filtro FIR directo, las muestras entran a los filtros en forma paralela (no se necesita el bloque shift register).
- Todos los filtros utilizan los mismos coeficientes.
- $\text{coeff}[0] = 8'd0$ ;  $\text{coeff}[1] = 8'd229$ ;  $\text{coeff}[2] = 8'd0$ ;  $\text{coeff}[3] = 8'd81$ ;  
 $\text{coeff}[4] = 8'd127$ ;  $\text{coeff}[5] = 8'd81$ ;  $\text{coeff}[6] = 8'd0$ ;  $\text{coeff}[7] = 8'd229$ ;

# Paralelismo - Forma Directa

