

 Institución Universitaria Centro de Laboratorios	GUÍA DE TRABAJO PRÁCTICO - EXPERIMENTAL Talleres y Laboratorios de Docencia ITM	Código	FGL 029
		Versión	01
		Fecha	2014-08-20

1. IDENTIFICACIÓN DE LA GUÍA

Nombre de la guía:	Entrega No.3
Código de la guía (No.):	DDW103-1
Taller(es) o Laboratorio(s) aplicable(s):	Diseño Digital
Tiempo de trabajo práctico estimado:	4 horas TP – 8 horas TI
Asignatura(s) aplicable(s):	DDW103 Diseño Digital
Programa(s) Académico(s) / Facultad(es):	Tecnología en Electrónica – Tecnología en Telecomunicaciones

COMPETENCIAS	CONTENIDO TEMÁTICO	INDICADOR DE LOGRO
El estudiante está en capacidad de diseñar e implementar circuitos digitales sobre arquitecturas heterogéneas basadas en SoC-FPGA usando High-Level Synthesis (HLS).	Arquitecturas heterogéneas basadas en SoC-FPGA y High-Level Synthesis (HLS).	Describe una arquitectura sobre FPGA usando HLS para acelerar el cómputo de un algoritmo y programa sobre el ARM para enviar y recibir los datos del algoritmo.


2. FUNDAMENTO TEÓRICO

El fundamento teórico para esta guía fue impartido en clase basado en el material que se encuentra en las dos secciones anteriores en la plataforma CVIRTUAL:

- **Introducción a Paralelización y Vivado HLS y en la sección:**
 - Introduction to parallel computing
 - Vivado HLS o Vitis HLS

3. OBJETIVO(S)

- Diseñar e implementar diseños sobre arquitecturas heterogéneas basadas en SoC-FPGA.
- Implementar paralelizaciones sobre el diseño planteado para mejorar el desempeño del algoritmo.

 Institución Universitaria Centro de Laboratorios	GUÍA DE TRABAJO PRÁCTICO - EXPERIMENTAL Talleres y Laboratorios de Docencia ITM	Código	FGL 029
		Versión	01
		Fecha	2014-08-20

4. RECURSOS REQUERIDOS

- Estación de trabajo.
- Vivado Design Suite.
- Vivado HLS o Vitis HLS (depende de la versión)

5. PROCEDIMIENTO O METODOLOGÍA PARA EL DESARROLLO

La actividad consiste en implementar en Vivado HLS o Vitis HLS el algoritmo propuesto por el grupo. Los estudiantes pueden encontrar un proyecto HLS que sirve de base para la implementación de cualquiera de los algoritmos en la plataforma CVIRTUAL del curso DDW103. Dentro se encuentra un archivo comprimido llamado **rtos_master_axi.zip**, este archivo contiene los proyectos para Vivado y Vivado HLS para el desarrollo completo sobre el ZedBoard (es el mismo proyecto con el que se realizó el ejemplo en clase).

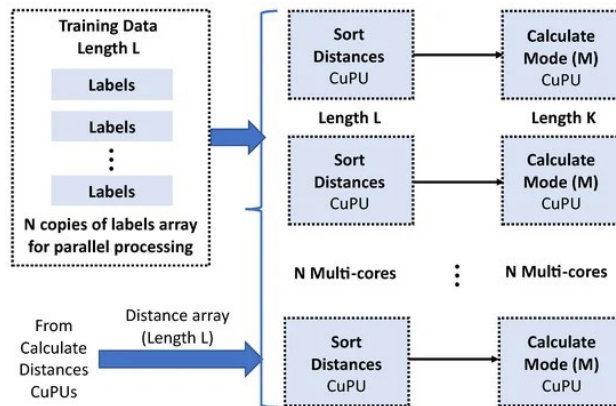
Para la **Entrega 3** solo es necesario abrir en Vivado HLS el proyecto `example_hls` que se encuentra dentro de la carpeta `rtos_master_axi`, como se realizó en clase. No es necesario trabajar con los otros archivos.

Los estudiantes deben realizar una exploración del espacio de diseño del algoritmo buscando acelerar la implementación por medio de directivas de optimización. **Toda la exploración realizada debe ser reportada en el informe de la entrega.** Para facilitar una exploración **MÍNIMA** del diseño, deben implementar los siguientes pasos sobre el algoritmo propuesto:

1. El algoritmo debe ser implementado en Vivado HLS (Vitis HLS) de manera secuencial sin directivas de paralelización para que sirva como línea base para comparar contra las futuras implementaciones.
2. El algoritmo debe ser implementado en Vivado HLS (Vitis HLS) y debe aplicarse en alguno de los ciclos la directiva Pipeline, adicional a las que se encuentran por defecto en el proyecto y analizar si hubo algún cambio en consumo de recursos (LUT, FF, BRAM, DSP) y latencia. Ver Pipeline y Unroll en : <https://docs.xilinx.com/r/en-US/ug1399-vitis-hls/HLS-Pragmas>
3. El algoritmo debe ser implementado en Vivado HLS (Vitis HLS) y debe aplicarse en alguno de los ciclos la directiva Unroll y analizar si hubo algún cambio en consumo de recursos (LUT, FF, BRAM, DSP) y latencia.
4. El algoritmo debe ser implementado en Vivado HLS y debe aplicarse en alguno de los ciclos la directiva Unroll y alguna directiva para particionamiento del arreglo, por ejemplo Array_Partition. Analizar si hubo algún cambio en consumo de recursos (LUT, FF, BRAM, DSP) y latencia. Ver mas directivas en el siguiente link: <https://docs.xilinx.com/r/en-US/ug1399-vitis-hls/HLS-Pragmas>
5. Usar la herramienta de análisis (Schedule Viewer como se vió en clase) y **proponer al menos tres (3) arquitectura** que produzca mejores resultados que la

línea base, mezclando las diferentes directivas de paralelización. Realice **al menos 1 cambio de tamaño del arreglo**, por un valor superior a 1024 y que sea potencia de 2 (2048, 4096, ...). Sino se obtiene un mejor resultado, de todas formas, reporte el intento con la respectiva explicación de lo que se quería obtener con esa configuración.

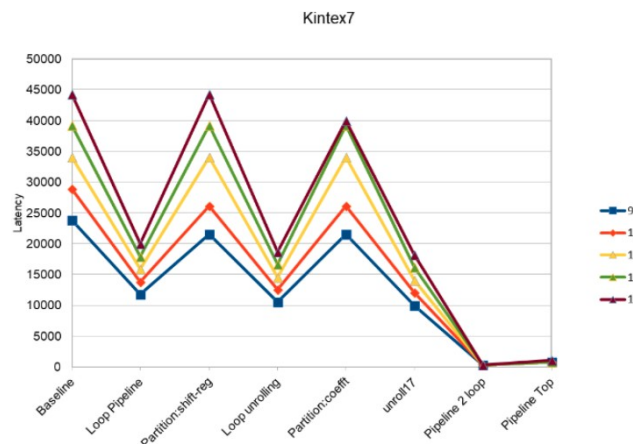
6. Realice diagramas de flujo y diagramas de bloques para explicar cada una de sus arquitecturas como se aprecia en la siguiente imagen. (Mas inspiración para reportar los resultados las puede encontrar en: <https://www.mdpi.com/2079-9292/10/5/627> , recomiendo bajar versión PDF para mejor visualización)

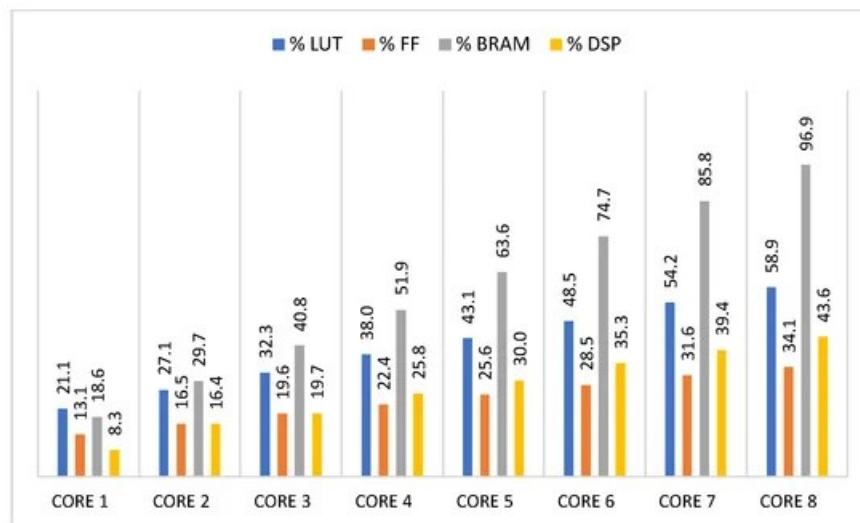
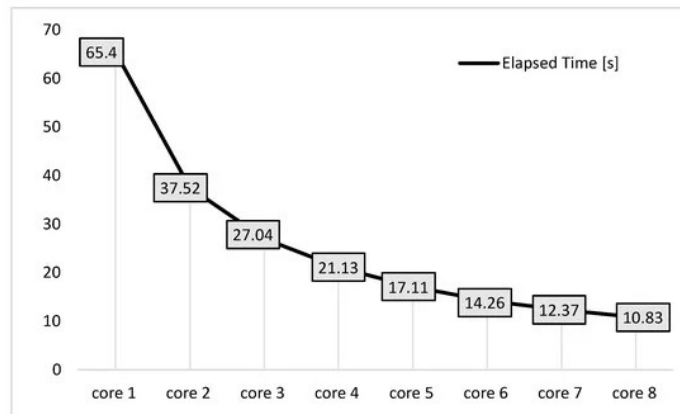


7. También, deben obtener los datos de consumo de recursos de sus implementaciones. (Esto debe aparecer en su informe). incluir tablas como la siguiente:

	Latency	BRAM %	FF %	LUT %	DSP %
Baseline	12704645	20.83	2.41	6.59	4.72
Parallel_224_2	254260	20.37	37.42	55.97	70.56
Parallel_197_2	221110	20.37	38.28	56.74	72.22
Parallel_147_2	170867	20.37	33.51	53.88	77.22

También, deben usar MS Excel o cualquier software parecido para realizar gráficas para observar las directivas de paralelización contra la latencia y los recursos, como las siguientes (**son solo un ejemplo, puede realizar unas diferentes**):





6. PARÁMETROS PARA ELABORACIÓN DEL INFORME

Al inicio del semestre se entrega una plantilla para la elaboración del informe.

7. CALIFICACIÓN

100% Exploración de espacio de diseño: **Proyecto en HLS e Informe Técnico** detallado de la exploración del espacio de diseño.

¿Que deben subir a la plataforma?

Crear una carpeta marcada de la siguiente forma **Grupo_X_P3_2023_2**, donde X es el número del grupo. Comprimir esa carpeta en un **ZIP** con los siguientes archivos:

- El proyecto en Vivado HLS donde se encuentran todas las soluciones implementadas en la exploración.
- El informe técnico en PDF

 Institución Universitaria Centro de Laboratorios	GUÍA DE TRABAJO PRÁCTICO - EXPERIMENTAL Talleres y Laboratorios de Docencia ITM	Código	FGL 029
		Versión	01
		Fecha	2014-08-20

8. BIBLIOGRAFÍA

- Brown, S., & Vranesic, Z. (2007). Fundamentos de lógica digital con diseño VHDL. México, 2a.
- Floyd, T. L., Caño, M. J. G., de Turiso, E. B. L., & Herrero, L. P. (1997). Fundamentos de sistemas digitales (Vol. 7). Prentice Hall.
- Mandado, E., Acevedo, J., & Fernández, C. (2009). Autómatas programables y sistemas de automatización. Marcombo.
- Mano, M. M., & Sánchez, G. D. (2003). Diseño digital. Pearson Educación.
- Tocci, R. J., & Widmer, N. S. (2003). Sistemas digitales: principios y aplicaciones. Pearson Educación.
- Wakerly, J. F. (2001). Diseño digital: principios y prácticas. Pearson educación.
- David G. Martínez, VHDL: El Arte de Programar Sistemas Digitales, primera edición, Ed. CECOSA, 2002, págs. 153-162.

Elaborado por:	<i>David Márquez</i>
Revisado por:	
Versión:	<i>3.0</i>
Fecha:	<i>Noviembre de 2023</i>