

Plan 95 Adecuado

ASIGNATURA:	DISEÑO DIGITAL CON LOGICA PROGRAMABLE	CODIGO:	95-0407
DEPARTAMENTO:	ELECTRÓNICA	CLASE:	ELECTIVA DE ESPECIALIDAD
ÁREA:	TÉCNICAS DIGITALES	HORAS SEM.:	4 HS.
		HORAS / AÑO:	64 HS.

Fundamentación:

La lógica digital ha evolucionado de manera continua y sostenida a lo largo de las ultimas 5 décadas, el salto que existe desde el transistor a los dispositivos conocidos como FPGA es enorme, pero ha seguido un continuo que resumiendo podría abarcar: lógica a transistor, compuertas, MSI, PAL/PLA, GAL, CPLD, FPGA y finalmente SOCs (Siglas del inglés con que se conocen los “Sistemas en un Chip”). Esta evolución tecnológica fue acompañado por su correspondiente evolución de las herramientas y técnicas de diseño.

En la actualidad el diseño digital con lógica programable en FPGA tiene un amplio campo de aplicaciones. Por citar algunos ejemplos significativos podríamos nombrar: implementación de sistemas de alta velocidad y gran capacidad de cálculo (características alcanzadas en gran medida debido al profundo grado de paralelismo inherente a dicha tecnología); Diseño de periféricos de uso específico; Dispositivos con posibilidades de reconfiguración ya sea para actualizaciones como para reutilización de hardware; Implementación para validación de ASICs (siglas del inglés que refieren a Circuitos Integrados de Aplicación Específica).

Desde hace tiempo los fabricantes de FPGA han incorporado microprocesadores en hardware y en forma de softcores, en la actualidad esta característica se ha intensificado. La mayoría de los fabricantes de FPGA ofrecen chips con procesadores ARM hardcore. Esto da la posibilidad de diseñar periféricos dedicados y conectarlos mediante un BUS interno de alta performance (típicamente AXI) a un procesador, tarea que hasta hace poco tiempo quedaba restringida a los fabricantes. Esto ha dado lugar al diseño sistemas de alta complejidad con interfaces de alto nivel usando sistemas operativos (Linux típicamente). Dicha circunstancia mejora considerablemente los diseños ya que las tareas de alto nivel (interfaz de usuario, comunicación, etc) se desarrollan mediante lenguajes de programación tradicionales en microprocesadores y la parte de alta performance se implementa en lógica programable. Esto da lugar a lo que se llama Co-Diseño software-hardware.

En la actualidad los fabricantes de FPGA, empujados por las estrictas leyes del mercado (ganar nuevos mercados, lo que significa en este caso, entre otras cosas tener más usuarios capacitados para el uso de sus tecnologías), han incrementado sus esfuerzos en la evolución de las técnicas y

herramientas de Síntesis de Alto Nivel (HLS). Esto es la descripción de un sistema digital mediante un lenguaje de alto nivel (C/C++) y su posterior traducción a un HDL, para ser finalmente implementado en una FPGA. Las herramientas para este tipo de diseño han acelerado su evolución (maduración) en los últimos años de manera significativa. Es claro que dicha evolución es soportado por el uso de mayor cantidad de recursos de hardware en los chips lo que redundo en diseños más costosos desde ese punto de vista, sin embargo, el tiempo de desarrollo puede ser disminuido considerablemente achicando el tiempo de puesta en mercado.

Las técnicas de HLS en conjunto con el Co-Diseño dejan ver que estamos frente a un cambio de paradigma en el Diseño Digital Programable el cual debe ser atendido a nivel académico. Esta materia se plantea, en el medio de dicho cambio. Por lo tanto si bien la asignatura está pensada para el diseño "tradicional" se le dará un sesgo hacia las nuevas modalidades y se prevé su actualización a lo largo del tiempo para acompañar dicha evolución.

Objetivos:

Los objetivos generales de la asignatura se centran en la formación de profesionales con claras competencias, en: capacidad de Análisis, especificación, implementación y validación de sistemas digitales basados en tecnología de lógica programable.

Al final del curso, el estudiante debe haber desarrollado las siguientes competencias:

Conocer el campo de aplicación de los dispositivos lógicos programables.
Implementar el ciclo completo de un desarrollo de mediana/alta complejidad
Dominar un lenguaje de descripción de hardware
Manejar una metodología de diseño con vistas a su verificación
Manejar técnicas de simulación y validación así como las herramientas de software para tal fin
Resolver problemas de implementación de origen temporal
Describir algoritmos matemáticos y poder dimensionar el ambiente aritmético para resolver problemas de aritmética finita
Tener un manejo básico de técnicas y herramientas utilizadas en co-diseño de hardware-software
Dominar las distintas tecnologías de lógica programable
Poder identificar los componentes de hardware necesarios para realizar un diseño basado en FPGA

Programa sintético:

- Unidad Temática 1- Arquitectura, Tecnologías y Herramientas.
- Unidad Temática 2- Descripción de hardware para síntesis.
- Unidad Temática 3- Simulación y Validación.
- Unidad Temática 4- Aspectos temporales.
- Unidad Temática 5- Implementación de algoritmos matemáticos y aritmética finita.
- Unidad Temática 6- Introducción a Co-Diseño y Síntesis de Alto Nivel.
- Unidad Temática 7- Aplicaciones y proyecto integrador.

Programa analítico:

Unidad 1: Arquitectura, Tecnologías y Herramientas

Arquitectura de CPLD, FPGA y SOC (System on Chip). Tecnologías de programación (FLASH,FUSE,SRAM). Configuración. Celdas de entrada salida, estándares de entrada-salida, LVCMOS, LVDS, PCI, DDR. SERDES. Restricciones de IO. BlockRAMs. Recursos de Reloj. Recursos para DSP. Primitivas. Sintetizadores, Técnicas denominadas "Place and Route". Simuladores. Análisis de los componentes necesarios para el desarrollo de una placa basada en FPGA.

Unidad 2: Descripción de hardware para síntesis

Repaso de VHDL para síntesis. Estándar VHDL 2008. Librerías estándar. Descripción de circuitos típicos y análisis de recursos y performance. Inferencia de memorias RAM/ROM simple y doble puerto. Controladores de memorias e interfaces con dispositivos externos (ADC, DAC). Portabilidad, uso de IP-Cores e instanciación de primitivas.

Unidad 3: Simulación y Validación

VHDL para simulación. Bancos de prueba. Manejo de archivos. Control de tiempo. Asserts. Generación de patrones de entrada (determinísticos y pseudoaleatorios). Validación. Cobertura de código. Introducción al Lenguaje de Especificación de propiedades (PSL).

Unidad 4: Aspectos temporales

Metaestabilidad, MTBF. Camino crítico y frecuencia máxima. Efectos de Skew, Jitter. Dominios de reloj. Sincronización. División de reloj. Sincronización de buses de datos. Restricciones de tiempo. Análisis de reportes de herramientas. Consumo. Retardo de entrada-salida

Unidad 5: Implementación de algoritmos matemáticos y aritmética finita

Recursos aritméticos en hardware: multiplicadores, sumadores, bloques DSP. Implementación de algoritmos matemáticos típicos: calculo de radicales, logaritmos, filtros Digitales, FFT, CORDIC. Implementación de funciones mediante tablas y/o ecuaciones. Dimensionamiento del ambiente aritmético. Librerías de Punto Fijo. Validación de algoritmos bit a bit.

Unidad 6: Introducción a Co-Diseño y Síntesis de Alto Nivel

Hardcores y Softcores. Mecanismos y buses de interconexión PSPL (siglas del inglés por Sistema Programable - Lógica Programable), Diseño de periféricos y su interfaz con el procesador. Introducción a Síntesis de Alto Nivel (HLS)

Unidad 7: Aplicaciones y proyecto integrador

Machine Learning. Deep Neural Networks xDNN. Sistemas de adquisición de señales de alta velocidad. Síntesis de señales arbitrarias. Síntesis de señales estocásticas. Generación de números aleatorios con distribuciones arbitrarias. Instrumentación Nuclear. Procesamiento digital de pulsos nucleares. Analizadores Multicanal para espectroscopia gamma. Photon Counting. LIDAR. TDC (time to digital conversion).

Estrategias Metodológicas

- Modalidades de enseñanza empleadas según tipo de actividad (teórica-práctica)

La asignatura Diseño Digital con Lógica Programable se ajusta a un cronograma cuatrimestral. Cada clase consta aproximadamente de un 50% de dictado teórico y un 50% práctico dándole gran dinamismo y un fuerte carácter práctico.

Cada unidad temática tiene ejemplos de aplicación y prácticas tipo hands-on. La asignatura posee tres trabajos prácticos a desarrollar por los estudiantes, que son entregados y evaluados de manera individual. Cada trabajo practico abarca entre dos y tres unidades temáticas. Los mismos se desarrollan de manera incremental dando como resultado un sistema de mediana complejidad a ser implementado en Kits de desarrollo. Cada unidad temática tiene un desarrollo aproximado de dos semanas.

La asignatura está diseñada de tal manera que los temas tratados se refuerzan entre sí y que conforme avanza el dictado de la asignatura se utilicen los conceptos y criterios aprendidos para los nuevos temas encarados. Cada unidad temática cuenta con su respectiva bibliografía y con notas de aplicación o de divulgación científicas, siendo esto un complemento necesario y obligatorio para poder llevar adelante la asignatura.

Esta modalidad permite combinar el aprendizaje de conceptos con su inmediata aplicación práctica, de modo de ir fomentando la creatividad y el ingenio por parte del estudiante, condiciones necesarias para el perfil de todo graduado en ciencias de la ingeniería.

El equipamiento necesario para el desarrollo de la asignatura consta de: kits de desarrollo de FPGA/CPLD, osciloscopios digitales, generadores de señales y generadores de señales arbitrarios, Analizadores lógicos.

Cronograma de clases y distribución de carga horaria:
--

Clase nro.	Tipo de actividad	Tema a tratar	Lugar de desarrollo	Elementos didácticos necesarios
1	Teoría (100)	Arquitectura, Tecnologías y Herramientas	Aula	Pizarrón Cañón
2	Teoría / Practica (75 / 25)	Descripción de hardware para síntesis	Laboratorio	Pizarrón PCs
3	Teoría / Practica (75 / 25)	Simulación y Validación	Laboratorio	Pizarrón PCs
4	Teoría / Practica (75 / 25)	Aspectos temporales	Laboratorio	Pizarrón PCs
5	Trabajo Practico (100)	Trabajo Practico I: Síntesis, Simulación y Validación	Laboratorio	Pizarrón PCs Kits de FPGA
6	Teoría / Practica (75 / 25)	Implementación de algoritmos matemáticos y aritmética finita	Laboratorio	Pizarrón PCs

7	Trabajo Practico (100)	Trabajo Practico II: implementación de algoritmos aplicados al tratamiento de señales	Laboratorio	Pizarrón PCs Kits de FPGA Osciloscopio Generador de señales
8	Trabajo Practico (100)	Trabajo Practico II: implementación de algoritmos aplicados al tratamiento de señales continuación	Laboratorio	Pizarrón PCs Kits de FPGA Osciloscopio Generador de señales
9	Teoría / Practica (75 / 25)	Síntesis de Alto Nivel	Laboratorio	Pizarrón Cañón PCs
10	Teoría / Practica (75 / 25)	Co-Diseño Hard/Soft	Laboratorio	Pizarrón Cañón PCs
11	Trabajo Practico (100)	Trabajo Practico III: Control de periféricos desde Hardcores	Laboratorio	Pizarrón Cañón PCs Osciloscopio Kits de FPGA
12	Trabajo Practico (100)	Trabajo Practico III: Control de periféricos desde Hardcores. Continuación	Laboratorio	Pizarrón Cañón PCs Osciloscopio Kits de FPGA
13	Teoría / Practica (50 / 50)	Aplicaciones y proyecto integrador	Laboratorio	Pizarrón PCs Kits de FPGA Osciloscopio Generador de señales
14	Teoría / Practica (50 / 50)	Aplicaciones y proyecto integrador	Laboratorio	Pizarrón PCs Kits de FPGA Osciloscopio Generador de señales
15	Teoría / Practica (50 / 50)	Aplicaciones y proyecto integrador	Laboratorio	Pizarrón PCs Kits de FPGA Osciloscopio Generador de

				señales
16	Parcial	Parcial integrador	Aula	Pizarrón

Evaluación:

Modalidad (tipo, cantidad, instrumentos):

Trabajos Prácticos:

Se realizarán tres trabajos prácticos los cuales están articulados en torno a un proyecto de mediana complejidad. Los componentes desarrollados para estos trabajos prácticos serán de utilidad al alumno para el proyecto integrador.

Examen parcial:

Evaluación sobre aspectos teóricos-prácticos de los principales temas de la asignatura Proyecto integrador:

A lo largo del cuatrimestre el alumno deberá desarrollar (en base a especificaciones) un proyecto de mediana complejidad el cual será acompañado de un informe. El tema del proyecto podrá ser propuesto por el alumno o por el docente

Requisitos de regularidad

- Cumplir con el régimen de asistencia.
- Aprobar los trabajos prácticos desarrollados a lo largo del cuatrimestre.
- Aprobar el examen parcial integrador con una nota igual o mayor a 6 (seis).
- Aprobar el proyecto integrador con una nota igual o mayor a 6 (seis).
- El examen parcial constará con dos instancias de recuperación.

Condiciones de promoción

- Cumplir condiciones de regularidad
- Aprobar el examen parcial integrador con una nota igual o mayor a 8 (ocho).
- Aprobar el proyecto integrador con una nota igual o mayor a 8 (ocho).
- El examen parcial para promoción constará con una instancia de recuperación.

Articulación Horizontal y Vertical con otras materias:

La asignatura se encuentra enmarcada dentro del Área de Técnicas Digitales. La articulación vertical se realiza con las asignaturas Técnicas Digitales I e informática II y horizontal con Técnicas Digitales II (a nivel de proyecto integrador). Se realizarán reuniones periódicas del área de Sistemas Digitales.

Bibliografía:

BIBLIOGRAFÍA OBLIGATORIA

"RTL Hardware Design Using VHDL: Coding for Efficiency, Portability, and Scalability", Pong P. Chu, Wiley-IEEE Press, ISBN-13: 978-0471720928, 2006.

Libro de principios fundamentales y de referencia internacional. Cubre la mayoría de los aspectos fundamentales de la materia con profundidad.

"Digital Signal Processing with Field Programmable Gate Arrays", U. Meyer-Baese, Springer, ISBN-13: 978-3642453083, 2014.

Libro de referencia orientado al desarrollo de algoritmos de procesamiento digital de señales con un profundo manejo de aritmética finita.

"The Zynq Book: Embedded Processing with the Arm Cortex-A9 on the Xilinx Zynq-7000 All Programmable Soc", de Louise H. Crockett, Ross a. Elliot, Martin a. Enderwitz, Strathclyde Academic Media. ISBN-13: 978-0992978709. 2014.

Si bien es un libro basado en un producto comercial (familia Zynq de Xilinx), el libro cubre el enfoque moderno orientado al desarrollo basado en SoC (por las siglas en Ingles de System on Chip).

BIBLIOGRAFÍA COMPLEMENTARIA

"Circuit Design and Simulation with VHDL", V. A. Pedroni, The MIT Press, ISBN-13: 978-0262014335, 2010.

"Finite State Machines in Hardware: Theory and Design (with VHDL and SystemVerilog)", Volnei A. Pedroni, ASIN: B00HRYBIQW, The MIT Press, 2013.

"The Designer's Guide to VHDL", P. Ashenden, Morgan Kaufman, ISBN-13: 978-01208878592008, 2008.

"VHDL 2008: Just the New Stuff", P. J. Ashenden, J. Lewis, Morgan Kaufmann, ISBN-13: 000-0123742498, 2007.

"Application-Specific Integrated Circuits", Michael J. S. Smith, Addison Wesley, ISBN-13: 978-0201500226, 1997.

"VHDL Analysis and Modeling of Digital Systems", Zainalabedin Navabi. Mc Graw-Hill, ISBN-13: 978-0071127325, 1993.

"Designing with Xilinx FPGAs: Using Vivado", Springer, ISBN-13: 978-3319424378

"The Zynq Book Tutorials for Zybo and ZedBoard", de Louise H. Crockett, Ross a. Elliot, Martin a. Enderwitz, Strathclyde Academic Media. ISBN-13: 978-0992978730

"A Practical Introduction to PSL (Integrated Circuits and Systems)", Cindy Eisner, Dana Fisman, Springer, ISBN-978-0-387-36123-9

"Assertion-Based Design", Harry D. Foster, Adam C. Krolnik, David J. Lacey, Kluwer Academic, ISBN-13: 978-1402080272

Correlativas:

Para cursar:

Cursada: Técnicas Digitales I
Informática II

Para rendir:

Aprobada: Técnicas Digitales I
Informática II