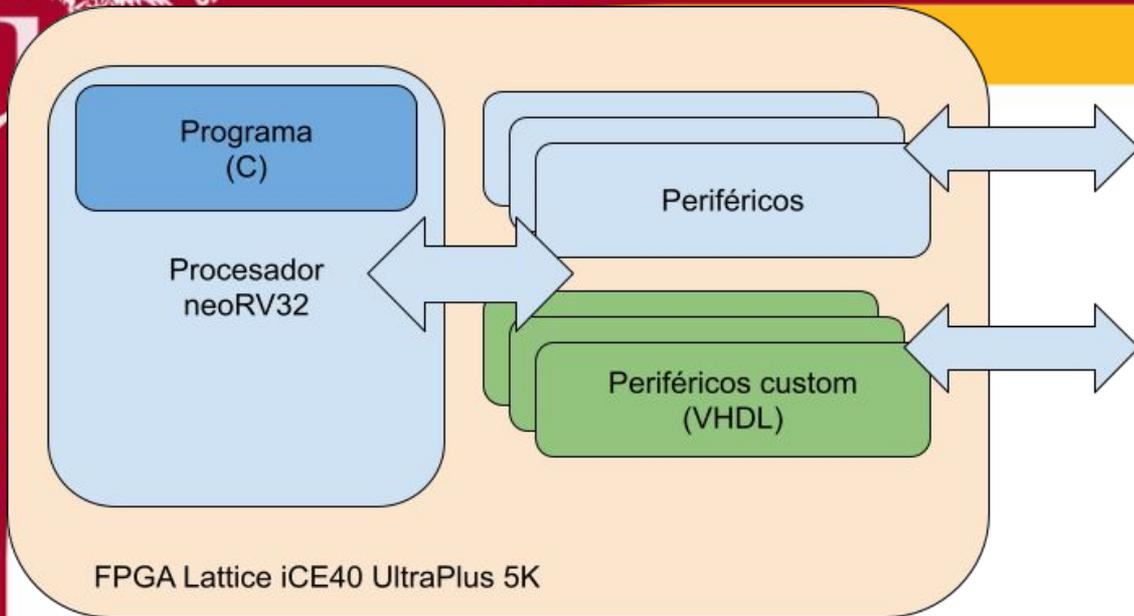


Sistemas Electrónicos para Automatización

Contenidos Proyecto Docente
Parte FPGAs

Hipólito Guzmán Miranda
hguzman@us.es

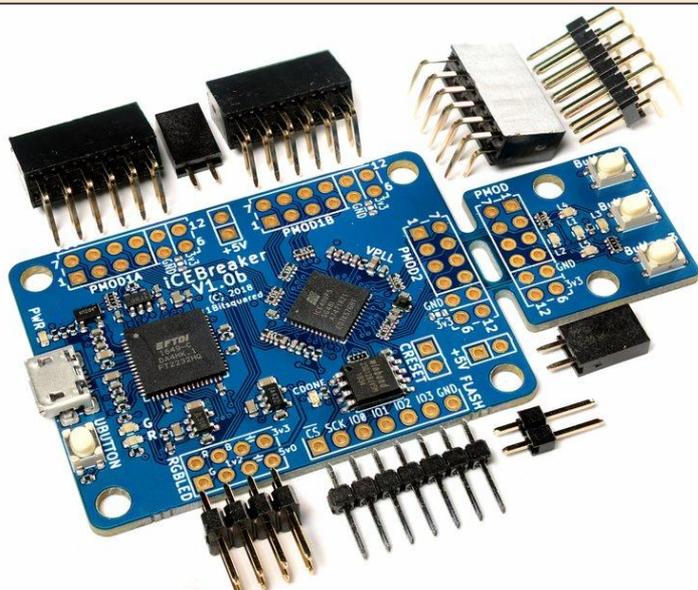
Objetivo: System-on-Chip



Software Prácticas:

- Herramientas libres diseño FPGA:
 - yosys
 - nextpnr
 - icestorm
- Toolchain RISC-V

Instrucciones de instalación en www.gitlab.com/sepagiern/sepa-practicas



Tarjeta iCEBreaker FPGA

¿Otro toolchain?

2º	Electrónica Digital	VHDL (modificando código existente)	Toolchain de Lattice
3º	Sistemas Electrónicos	VHDL (escribiendo el código desde cero)	Toolchain de Xilinx
4º	Sistemas Electrónicos para Automatización	C (software), VHDL (hardware custom)	Herramientas libres

- **ED, SE:** Diseño puramente en VHDL
- **SEPA:** Diseño de un sistema-en-chip completo: microprocesador y periféricos
 - Software que ejecuta el microprocesador: C
 - Periféricos a medida: VHDL

En la industria existen diferentes herramientas pero por dentro todas hacen los mismos pasos (síntesis, implementación, generación del bitstream, configuración). Es bueno conocer distintas herramientas para evitar el 'vendor lockdown'.

Tema 10 - Arquitectura de microprocesadores actuales

- Revisión de arquitecturas clásicas
- Concepto de Pipeline
- Memorias Caché

(Repaso arquitectura microprocesador)

Tema 11 - FPGAs para automatización

- FPGAs como System-on-Chip
- El 'Design Gap'
- Soft processors y diseño con IP cores
- Buses para microprocesadores empotrados

(Por qué es importante lo que vamos a ver)

Tema 12 - Arquitectura de un soft processor

- Arquitectura
- Buses
- Configuración
- Software para diseño
- Periféricos personalizados por el usuario

(Veremos una de las posibles alternativas de diseño, el procesador **neoRV32**, que es el que usaremos en esta parte de SEPA)

Tema 13 - Arquitectura de Microprocesadores Avanzados

- Sistemas multiprocesador
- Procesadores superescalares
- Super-pipeline
- Procesadores VLIW

(Profundizando en las arquitecturas avanzadas)

Tema 14 - Alternativas en soft processors

- Otros microprocesadores
- Alternativas libres en diseño FPGA
- Microcontroladores soft-core

(Visión de conjunto del resto de alternativas de diseño)

Prácticas

5 sesiones prácticas de 2h:

1. Manejo de las herramientas y toma de contacto con el procesador
2. Control de un teclado numérico utilizando GPIO
3. Creación de periféricos personalizados en VHDL
4. Trabajo libre y ampliaciones
5. Presentaciones de los system-on-chip desarrollados

Enunciados disponibles en acceso abierto en <https://gitlab.com/sepagierm/sepa-practicas>

Criterios de evaluación

Parte FPGAs = 55% Teoría + 45% Prácticas

Mínimo de 3.5 sobre 10 en Teoría y Prácticas

- Teoría: se evalúa mediante examen
- Prácticas: 5 sesiones.
 - Por parejas (1 tarjeta por pareja)
 - En cada sesión el trabajo se realiza implementando mejoras sobre un único diseño
 - Se entrega 1 única memoria final
 - Se realizará una defensa de unos 5 min. por pareja
 - Se evalúan: funcionalidades implementadas (las mínimas que se piden sirven para llegar al 5, se sube nota realizando ampliaciones), calidad de la memoria entregada, defensa, calidad y limpieza de las implementaciones y código.

Créditos ECTS

1 ECTS = 25 horas de trabajo. En GIERM se dividen en:

- 10 presenciales +
- 15 de estudio autónomo

SEPA tiene 4.5 ECTS = 67.5 h de estudio autónomo

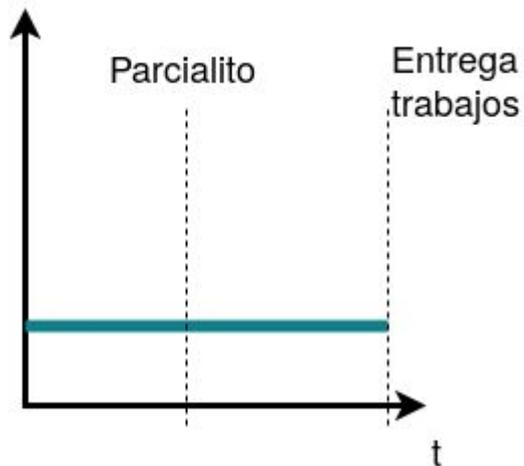
- Unas 4.5/4.8 h por semana de trabajo autónomo (depende de si hay 15/14 semanas en el C1)

1 práctica de 2h -> 3h trabajo autónomo

1 clase de teoría de 1h40 -> 2h30 de estudio autónomo

Horas no presenciales

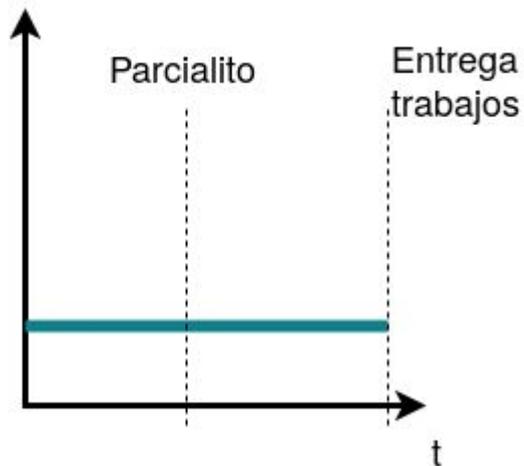
Dedicación no presencial



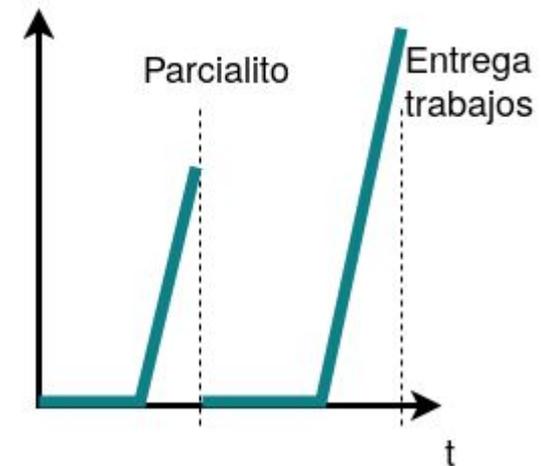
Cómo deberían ser

Horas no presenciales

Dedicación no presencial



Dedicación no presencial



Cómo deberían ser

Cómo son en realidad

Planificación temporal *sugerida*

(parte FPGAs)

Semana	Presencial	Autónomo
8	T10 + préstamo de tarjetas (1h40)	Estudiar T10, leer README.md de https://gitlab.com/sepagiern/sepa-practicas , preparar máquina virtual, instalar software y comprobar que está instalado
9	T11 (1h40)	Estudiar T11, repasar tema anterior, leer enunciado P1 (junto con su “documentación importante”), cacharrear con la tarjeta
10	T12 (1h40), P1 (2h)	Estudiar T12, repasar temas anteriores, terminar P1 (incluyendo la sección “preparación para la siguiente práctica”), leer P2 (con su “doc. importante”)
11	T12.2 (1h40), P2 (2h)	Estudiar T12.2, repasar temas anteriores, terminar P2 (incluyendo “preparación para la siguiente práctica”: ojo que toca repasar VHDL), leer P3 (incluyendo “doc. importante”), escribir los avances en la memoria de prácticas
12	T13 (1h40), P3 (2h)	Estudiar T13, repasar temas anteriores, terminar P3, escribir los avances en la memoria de prácticas, leer P4
13	T14 (1h40), P4 (2h)	Estudiar T14, repasar temas anteriores, terminar P4, terminar la memoria de prácticas, realizar la entrega de prácticas
14	Parcialito (1h40), P5: defensa prácticas (2h)	Último esfuerzo con la teoría