

## [GIEE04] ARQUITECTURA DE COMPUTADORES

### DATOS GENERALES

<b>Titulación</b> GRADO EN INGENIERÍA INFORMÁTICA		<b>Materia</b> INGENIERÍA DE COMPUTADORAS
<b>Semestre</b> 1	<b>Curso</b> 4	<b>Mención / EMPRESA</b>
<b>Carácter</b> INTENSIFICACIÓN		<b>Especialidad</b>
<b>Plan</b> 2008		<b>Idioma</b> ENGLISH
<b>Créditos</b> 6	<b>H./sem.</b> 2,5	<b>Horas totales</b> 45 h. lectivas + 105 h. no lectivas = <b>150 h. totales</b>

### PROFESORES

(No hay profesores asignados a la asignatura)

### CONOCIMIENTOS PREVIOS REQUERIDOS

<b>Asignaturas</b>	<b>Conocimientos</b>
(No se requiere haber cursado asignaturas previas específicas)	(No se requieren conocimientos previos)

### RESULTADOS DE APRENDIZAJE

#### **RG1410** Conocer la estructura de un microcontrolador

COMPETENCIAS	HT	ACTIVIDADES FORMATIVAS	HL	HNL	HT
Diseñar y desarrollar una plataforma hardware que responda a una aplicación dada optimizando los recursos utilizados	45 h.	Estudio y trabajo individual, pruebas y exámenes	3 h.	7 h.	10 h.
Diseñar y desarrollar un sistema digital embebido que responda a una aplicación dada cumpliendo los estándares de calidad vigentes	45 h.	Realización de ejercicios individualmente y en equipo	5 h.	12 h.	17 h.
		Realización de prácticas en talleres y/o laboratorios	4 h.	14 h.	18 h.
SISTEMAS DE EVALUACIÓN	P	MECANISMOS DE RECUPERACIÓN			
Punto de control	50%	Punto de control			
Ejercicios	30%	<b>Observaciones:</b>			
Prácticas	20%				
<b>Observaciones:</b>					

**HL - H. lectivas:** 12 h.  
**HNL - H. no lectivas:** 33 h.  
**HT - Total horas:** 45 h.

#### **RG1411** Ser capaz de generar código compatible y portable a diferentes arquitecturas

COMPETENCIAS	HT	ACTIVIDADES FORMATIVAS	HL	HNL	HT
Diseñar y desarrollar un sistema digital embebido que responda a una aplicación dada cumpliendo los estándares de calidad vigentes	35 h.	Desarrollo, redacción y presentación de proyectos e informes, realizados individualmente o en equipos	4 h.	6 h.	10 h.
Diseñar y Desarrollar el Sw de un sistema embebido que responda a una aplicación dada cumpliendo los estándares de calidad vigentes	35 h.	Estudio y trabajo individual, pruebas y exámenes	3 h.	7 h.	10 h.
		Realización de prácticas en talleres y/o laboratorios	5 h.	10 h.	15 h.
SISTEMAS DE EVALUACIÓN	P	MECANISMOS DE RECUPERACIÓN			
Ejercicios	30%	Punto de control			
Prácticas	50%	<b>Observaciones:</b>			
Punto de control	20%				
<b>Observaciones:</b>					

**HL - H. lectivas:** 12 h.  
**HNL - H. no lectivas:** 23 h.  
**HT - Total horas:** 35 h.

#### **RG1412** Ser capaz de comprender y desarrollar mapeos y lecturas de memoria y periféricos

COMPETENCIAS	HT	ACTIVIDADES FORMATIVAS	HL	HNL	HT
--------------	----	------------------------	----	-----	----

Diseñar y desarrollar una plataforma hardware que responda a una aplicación dada optimizando los recursos utilizados	14 h.	Estudio y trabajo individual, pruebas y exámenes	4 h.	10 h.	14 h.
Diseñar y Desarrollar el Sw de un sistema embebido que responda a una aplicación dada cumpliendo los estándares de calidad vigentes	14 h.				

SISTEMAS DE EVALUACIÓN	P	MECANISMOS DE RECUPERACIÓN
[!] Ariketak	50%	[!] Kontrol puntuak
[!] Kontrol puntuak	50%	<b>Observaciones:</b>

**HL - H. lectivas:** 4 h.  
**HNL - H. no lectivas:** 10 h.  
**HT - Total horas:** 14 h.

**RG1413** Ser capaz de comprender y hacer uso de la estructura de microcontroladores actuales

COMPETENCIAS	HT	ACTIVIDADES FORMATIVAS	HL	HNL	HT
Diseñar y desarrollar una plataforma hardware que responda a una aplicación dada optimizando los recursos utilizados	56 h.	Desarrollo, redacción y presentación de proyectos e informes, realizados individualmente o en equipos	4 h.	8 h.	12 h.
Diseñar y desarrollar un sistema digital embebido que responda a una aplicación dada cumpliendo los estándares de calidad vigentes	56 h.	Estudio y trabajo individual, pruebas y exámenes	6 h.	15 h.	21 h.
Diseñar y Desarrollar el Sw de un sistema embebido que responda a una aplicación dada cumpliendo los estándares de calidad vigentes	56 h.	Realización de prácticas en talleres y/o laboratorios	7 h.	16 h.	23 h.

SISTEMAS DE EVALUACIÓN	P	MECANISMOS DE RECUPERACIÓN
Ejercicios	30%	Punto de control
Prácticas	50%	<b>Observaciones:</b>
Punto de control	20%	

**HL - H. lectivas:** 17 h.  
**HNL - H. no lectivas:** 39 h.  
**HT - Total horas:** 56 h.

## CONTENIDOS

### 1. - Estructura de un microcontrolador

- 1.1 Definiciones: microcontroladores y sistemas embebidos
- 1.2 Tiempos de ejecución y conceptos básicos
- 1.3 La ALU
  - 1.3.1. - Coma flotante
  - 1.3.2. - DSPs
- 1.4 Registros
- 1.5 Estructura de la memoria
  - 1.5.1 Buses de datos y control
  - 1.5.2 Harvard vs Von Neumann
  - 1.5.3 La memoria cache
  - 1.5.4 Memoria Virtual
  - 1.5.5 Endianness
- 1.6 Contador de Programa (PC)
- 1.7 El pipeline y decodificación de instrucciones

1.8 El vector de reset

1.9 Buses de I/O

1.10 DMA

1.11 Procesadores paralelos

1.12 Ejemplos de procesadores

## **2. - El Lenguaje ensamblador**

2.1 Definiciones

2.2 Tipos de instrucciones

2.3 Decodificación de instrucciones

2.4 Traducción de código

2.5 La unidad de control

2.5.1 Microcódigo

2.6 Ejemplos

2.6.1 Instrucciones en los PIC

2.6.2 Instrucciones en los ARM

2.6.2.1 Aritméticas, booleanas, lógicas...

2.6.2.2 Saltos

2.6.2.3 Interrupciones

2.6.2.4 Manejo del stack

2.6.2.5 THUMB y THUMB2

## **3. - Los saltos y el puntero de pila**

3.1 Saltos

3.1.1- Ejemplo: Saltos en un PIC

3.1.2- Ejemplo: Saltos en un ARM7

3.2 Saltos anidados y el stack pointer

3.3 Funciones y parámetros

3.4 Cambios de contexto

3.5 Ejemplo: ARM7

## **4 Sistemas de interrupción**

4.1 Sistemas de interrupción en un  $\mu$ C

4.2 Niveles de prioridad de interrupciones

4.3 Cambios de contexto en interrupciones

4.4 Interrupciones anidadas

4.5 Ejemplos de sistemas de interrupción en diferentes arquitecturas de  $\mu$ C

## **5.- Periféricos**

5.1 Buses de periféricos y ubicación de la memoria

5.2 Periféricos especiales

5.2.1 Módulos COP/BIST

5.2.3 Interfaces JTAG

## 6 Desarrollo de SW y estructuración de código

6.1 El bootloader y la aplicación

6.2 Segmentos de memoria

6.3 BSP y estructuración de código

### COMPETENCIAS DEL CURSO

Aplicar diferentes conceptos, métodos, patrones, y arquitecturas software para el análisis, diseño y desarrollo de aplicaciones informáticas y comunicaciones cumpliendo modelos y estándares de calidad vigentes

Analizar, seleccionar e implementar algoritmos que den respuesta a problemas de ingeniería

Identificar, definir y gestionar los procesos relacionados con el desarrollo de software de calidad

Diseñar y desarrollar una plataforma hardware que responda a una aplicación dada optimizando los recursos utilizados

Diseñar y desarrollar un sistema digital embebido que responda a una aplicación dada cumpliendo los estándares de calidad vigentes

Diseñar y Desarrollar el Sw de un sistema embebido que responda a una aplicación dada cumpliendo los estándares de calidad vigentes

Desarrollar sistemas de recogida, almacenamiento y tratamiento de datos

Implicar y orientar a las personas hacia un objetivo común con una visión global del trabajo a desarrollar y de las características que el mismo requiere equilibrando los intereses individuales y colectivos.

En un entorno real o simulado, tomar decisiones estratégicas y/o resolver problemas de la organización aplicando las teorías, métodos, técnicas y herramientas más adecuadas y proponiendo acciones de mejora desde el ámbito de la gestión.

Identificando las oportunidades de mercado, proponer y gestionar nuevos proyectos con una visión global de negocio sostenible para dar respuesta a los requerimientos del clientes y grupos de interés.

### RECURSOS DIDÁCTICOS Y BIBLIOGRAFÍA

Recursos didácticos	Bibliografía
<i>(No hay recursos)</i>	Organización y diseño de computadores - La interfaz hardware/software - David A. Patterson - John L. Hennessy. Ed. Mc Graw Hill ISBN: 84-481-1829-4