

Guía docente de la asignatura

## Sistemas Empotrados (Especialidad Ingeniería de Computadores) (296114I)



Fecha de aprobación: 27/06/2023

<b>Grado</b>	Grado en Ingeniería Informática	<b>Rama</b>	Ingeniería y Arquitectura				
<b>Módulo</b>	Formación de Especialidad 3: Ingeniería de Computadores	<b>Materia</b>	Sistemas de Cómputo para Aplicaciones Específicas				
<b>Curso</b>	4º	<b>Semestre</b>	1º	<b>Créditos</b>	6	<b>Tipo</b>	Obligatoria

### PRERREQUISITOS Y/O RECOMENDACIONES

Se recomienda la superación de los contenidos y adquisición de competencias de las materias de formación básica, de las asignaturas obligatorias de rama Estructura de Computadores, Arquitectura de Computadores y Sistemas Operativos, y de las asignaturas del módulo de Ingeniería de Computadores Desarrollo de Hardware Digital y Sistemas con Microprocesadores.

### BREVE DESCRIPCIÓN DE CONTENIDOS (Según memoria de verificación del Grado)

- Metodología y herramientas de trabajo para sistemas empotrados.
- Selección de la plataforma y el procesador.
- Software de sistema y configuración.
- Desarrollo de drivers.
- Sistemas seguros y críticos para aplicaciones empotradas.
- Sistemas multi-núcleo.

### COMPETENCIAS ASOCIADAS A MATERIA/ASIGNATURA

#### COMPETENCIAS GENERALES

- CG04 - Capacidad para definir, evaluar y seleccionar plataformas hardware y software para el desarrollo y la ejecución de sistemas, servicios y aplicaciones informáticas.
- CG11 - Capacidad para analizar y valorar el impacto social y medioambiental de las soluciones técnicas, comprendiendo la responsabilidad ética y profesional de la actividad del Ingeniero Técnico en Informática.

#### COMPETENCIAS TRANSVERSALES



- CT02 - Capacidad para tomar decisiones basadas en criterios objetivos (datos experimentales, científicos o de simulación disponibles) así como capacidad de argumentar y justificar lógicamente dichas decisiones, sabiendo aceptar otros puntos de vista.

## RESULTADOS DE APRENDIZAJE (Objetivos)

### Objetivos formativos particulares

- Analizar sistemas empotrados. Identificar las características que diferencian a los sistemas de cómputo empotrados de los de propósito general.
- Seleccionar la plataforma más apropiada en función de los requerimientos del sistema empotrado.
- Seleccionar, configurar y usar herramientas de desarrollo y depuración de un sistema empotrado.
- Desarrollo de firmware para aplicaciones empotradas.
- Desarrollar controladores para los periféricos del sistema empotrado con distintos niveles de abstracción.
- Optimizar el código empotrado para la arquitectura del sistema, haciendo énfasis en maximizar las prestaciones y minimizar el consumo de energía.
- Caracterizar un sistema operativo empotrado. Selección, configuración y uso.
- Entender y manejar niveles de fiabilidad y tolerancia a fallos de un sistema empotrado. Aplicar técnicas de diseño y herramientas de prevención de los mismos.
- Identificar las certificaciones existentes para sistemas empotrados seguros.
- Identificar y argumentar la necesidad de usar varios procesadores o núcleos en un sistema empotrado.

### Objetivos formativos de carácter general (Competencias según BOE de 4 de Agosto de 2009)

- Capacidad de desarrollar procesadores específicos y sistemas empotrados, así como desarrollar y optimizar el software de dichos sistemas.
- Capacidad de analizar, evaluar y seleccionar las plataformas hardware y software más adecuadas para el soporte de aplicaciones empotradas y de tiempo real.

## PROGRAMA DE CONTENIDOS TEÓRICOS Y PRÁCTICOS

### TEÓRICO

1. Introducción a los sistemas empotrados
2. Procesador y mapa de memoria
3. Cargador de arranque
4. Excepciones e interrupciones
5. Entrada/salida

### PRÁCTICO

#### Seminarios:

1. Perspectivas laborales
2. Configuración e instalación de las herramientas de desarrollo

#### Prácticas:



1. Conexión y gestión remota de la plataforma
2. Introducción al ensamblador
3. Introducción al enlazador y el resto de las binutils
4. Arranque e inicialización del entorno de ejecución
5. Gestión de excepciones e interrupciones
6. El controlador de interrupciones
7. Gestión de los pines de entrada/salida
8. Desarrollo de un driver L0 para las UART
9. Desarrollo de un driver L1 para las UART
10. Desarrollo de un driver L2 para las UART

## BIBLIOGRAFÍA

### BIBLIOGRAFÍA FUNDAMENTAL

1. L. Edwards. [Embedded Systems Design on a Shoestring: Achieving High Performance with a Limited Budget](#). Newnes, 2003.
2. M. Barr, A. Massa. [Programming Embedded Systems: With C and GNU Development Tools](#), 2nd ed. O'Reilly, 2006.
3. M. Barr. [Embedded C Coding Standard](#). Netrino Institute, 2013.
4. J. Ganssle. [The Firmware Handbook](#), Newnes, 2004.
5. C. Walls. [Embedded Software. The Works](#), 2nd ed. Newness, 2012.
6. Q. Li, G. Yao. [Real-Time Concepts for Embedded Systems](#), CMP Books, 2003.
7. W. Hohl. [ARM Assembly Language. Fundamentals and Techniques](#), 2nd ed. CRC Press, 2014.
8. A. N. Sloss, D. Symes y C. Wright. [ARM System Developer's Guide. Designing and Optimizing System Software](#). Morgan Kaufmann, 2004.

### BIBLIOGRAFÍA COMPLEMENTARIA

1. A. Berger. [Embedded Systems Design: An Introduction to Processes, Tools & Techniques](#). CRC Press, 2011.
2. S. Heath. [Embedded System Design](#). Newness, 2ª edición, 2003.
3. P. Marwedel. [Embedded System Design: Embedded Systems Foundations of Cyber-Physical Systems](#), 2nd ed. Springer, 2011.
4. W. Wolf. [Computers as Components: Principles of Embedded Computing System Design](#), 3rd ed. Morgan Kaufmann, 2012.
5. F. Vahid y T. Givargis. [Embedded System Design: A Unified Hardware/Software Introduction](#), 2nd ed. John Wiley & Sons, 2002.
6. J. J. Labrosse, et al. [Embedded Software: Know it All](#). Newnes, 2008.
7. J. Ganssle, et al. [Embedded Hardware: Know it All](#). Newnes, 2008.
8. J. Ganssle (ed.). [Embedded Systems: World Class Designs](#). Newnes, 2008.
9. I. Lee, J. Y-T. Leung, S. H. Son. [Handbook of Real-Time and Embedded Systems](#). Champman & Hall/CRC, 2008.
10. K. E. Curtis. [Embedded Multitasking](#), Newnes, 2006.

## ENLACES RECOMENDADOS

- [EDN](#)



- [Embedded](#)
- [ARM Processors](#)
- [GNU Binutils](#)
- [GCC, the GNU Compiler Collection](#)
- [GDB: The GNU Project Debugger](#)
- [Open On-Chip Debugger](#)
- [Executable and Linking Format \(ELF\) Specification](#)
- [Mapping memory](#)
- [The Newlib Homepage](#)
- [The Red Hat newlib C Library](#)
- [Embed with GNU. Porting the GNU tools to embedded systems](#)
- [Porting and Using Newlib in Embedded Systems](#)
- [Porting Newlib](#)

## METODOLOGÍA DOCENTE

- MD01 - Lección Magistral (Clases Teóricas-Expositivas)
- MD02 - Actividades Prácticas (Resolución de Problemas, Resolución de Casos Prácticos, Desarrollo de Proyectos, Prácticas en Laboratorio, Taller de Programación, Aula de Informática, Prácticas de Campo).
- MD03 - Seminarios (Debates, Demos, Exposición de Trabajos Tutelados, Conferencias, Visitas Guiadas, Monografías).
- MD04 - Actividades no presenciales Individuales.
- MD05 - Actividades no presenciales Grupales.
- MD06 - Tutorías Académicas.

## EVALUACIÓN (instrumentos de evaluación, criterios de evaluación y porcentaje sobre la calificación final)

### EVALUACIÓN ORDINARIA

La calificación final que aparecerá en el Acta será un número comprendido entre 0 y 10 con una precisión de un dígito decimal. La metodología de evaluación por defecto según la normativa de la Universidad de Granada es la evaluación continua.

Tanto la teoría como las prácticas se evaluarán mediante un breve cuestionario sobre los contenidos de cada unidad en la sesión siguiente a su finalización. La siguiente tabla muestra la contribución de cada una de las actividades a la nota final de la asignatura y la nota mínima exigida, en su caso, para cada una de ellas:

Porcentajes de evaluación

Actividad	Porcentaje	Mínimo
Teoría	30%	1,5
Prácticas	70%	3,5

Todo lo relativo a la evaluación se regirá por la normativa sobre planificación docente y organización de exámenes vigente en la Universidad de Granada. El sistema de calificaciones se expresará mediante calificación numérica de acuerdo con lo establecido en el art. 5 del R.D. 1125/2003, de 5 de septiembre, por el que se establece el sistema europeo de créditos y el sistema de calificaciones en las titulaciones universitarias de carácter oficial y validez en el territorio nacional.

### EVALUACIÓN EXTRAORDINARIA



La teoría y las prácticas se evaluarán con sendos cuestionarios. Los porcentajes con los que cada cuestionario contribuirá a la nota final, así como la nota mínima exigida en cada uno de ellos, coinciden con los establecidos para la evaluación ordinaria.

### EVALUACIÓN ÚNICA FINAL

Se usará la misma metodología que en la evaluación extraordinaria.

### INFORMACIÓN ADICIONAL

- Preferentemente se hará uso de [Swad](#) como plataforma LMS de la asignatura, si bien se tendrá en consideración el uso de otras plataformas alternativas como [PRADO](#) en caso de ser necesario.
- En caso de ser necesario hacer videoconferencias, se usará [Google Meet](#), con acceso desde las cuentas @go.ugr.es.

