

30240 - Sistemas empotrados 2

Información del Plan Docente

Año académico: 2019/20

Asignatura: 30240 - Sistemas empotrados 2

Centro académico: 110 - Escuela de Ingeniería y Arquitectura

Titulación: 439 - Graduado en Ingeniería Informática

Créditos: 6.0

Curso: 4

Periodo de impartición: Primer semestre

Clase de asignatura: ---

Materia: ---

1. Información Básica

1.1. Objetivos de la asignatura

La asignatura y sus resultados previstos responden a los siguientes planteamientos y objetivos:

El objetivo de la asignatura es formar al estudiante en el diseño y programación de sistemas empotrados dotados de un sistema operativo. No solo se estudiarán los fundamentos, sino que se pretende conseguir capacidad de análisis y diseño de este tipo de sistemas informáticos. El estudiante deberá ser capaz de: (1) Identificar la necesidad de un sistema operativo de un tipo determinado cuando convenga al diseño de un sistema empotrado, (2) Establecer procedimientos de migración del sistema operativo a una plataforma determinada, configurándolo o modificándolo convenientemente, y (3) Utilizar correctamente los recursos de planificación y sincronización proporcionados por el sistema operativo.

1.2. Contexto y sentido de la asignatura en la titulación

Se trata de una asignatura de Tecnología Específica (TE) de Ingeniería de Computadores perteneciente a la materia Sistemas Empotrados, obligatoria en esta TE. Desarrolla aspectos introducidos en Sistemas Empotrados I relacionados con sistemas operativos, y junto a esta asignatura, sirve de base a la asignatura Laboratorio de Empotrados. Se apoya por tanto en Sistemas Empotrados I, además de en Introducción a los computadores, Arquitectura y organización de computadores I y II, Programación I y II, Sistemas Operativos, Programación de sistemas concurrentes y distribuidos y Proyecto Hardware.

1.3. Recomendaciones para cursar la asignatura

Para cursar esta asignatura se recomienda haber cursado anteriormente la asignatura Sistemas Empotrados I y la asignaturas previas que la soportan (Introducción a los Computadores, Arquitectura y Organización de Computadores 1 y 2, Laboratorio Hardware), además de la asignatura de Sistemas Operativos.

El estudio y trabajo continuado, desde el primer día del curso, son fundamentales para superar con el máximo aprovechamiento la asignatura.

Es importante resolver cuanto antes las dudas que puedan surgir, para lo cual el estudiante cuenta con la asesoría del profesor, tanto durante las clases como en las horas de tutoría destinadas a ello. Pueden realizarse consultas puntuales a través de correo electrónico.

2. Competencias y resultados de aprendizaje

2.1. Competencias

Al superar la asignatura, el estudiante será más competente para...

Concebir, diseñar y desarrollar proyectos de Ingeniería (CT1).

Combinar los conocimientos generalistas y los especializados de Ingeniería para generar propuestas innovadoras y competitivas en la actividad profesional (CT3).

Resolver problemas y tomar decisiones con iniciativa, creatividad y razonamiento crítico (CT4).

Diseñar y construir sistemas digitales, incluyendo computadores, sistemas basados en microprocesador y sistemas de comunicaciones (CEIC1).

Desarrollar sistemas empotrados con sistema operativo, así como desarrollar y optimizar el software de dichos sistemas (CEIC2).

Diseñar e implementar software de sistema y de comunicaciones (CEIC4).

Analizar, evaluar y seleccionar las plataformas hardware y software más adecuadas para el soporte de aplicaciones empotradas y de tiempo real (CEIC5).

Comprender, aplicar y gestionar la garantía y seguridad de los sistemas informáticos (CEIC6).

2.2.Resultados de aprendizaje

El estudiante, para superar esta asignatura, deberá demostrar los siguientes resultados...

El estudiante, para superar esta asignatura, deberá demostrar los siguientes resultados:

1. Tiene una visión amplia de los sistemas operativos más extendidos en sistemas empotrados y de tiempo real.
2. Conoce y sabe manejar entornos de desarrollo para sistemas operativos de sistemas empotrados y de tiempo real.
3. Sabe diseñar y construir sistemas empotrados y de tiempo real basados en microprocesadores o en otras plataformas, de poca complejidad, atendiendo a criterios de seguridad, fiabilidad, tolerancia a fallos y consumo de energía.

2.3.Importancia de los resultados de aprendizaje

Gran parte de los sistemas empotrados incorporan actualmente un sistema operativo que facilita el desarrollo y la gestión del sistema en su conjunto. Routers de comunicación, teléfonos IP, sistemas de navegación aérea y espacial, sistemas de control en automóviles, Set-Top-Boxes, controladores de domótica y de un amplio abanico de sistemas industriales, utilizan hoy día un sistema operativo. A esta variedad de sistemas le corresponde una gran variedad de sistemas operativos, algunos muy específicos, para un número reducido de procesos, planificación fija y escaso consumo de recursos, otros de propósito general, que precisan ser adaptados a necesidades particulares.

Existe una notable demanda en el mercado de profesionales capaces de seleccionar el sistema operativo adecuado, de adaptarlo en el menor tiempo posible siguiendo objetivos específicos, y de participar en el desarrollo del producto como experto en ese sistema operativo.

3.Evaluación

3.1.Tipo de pruebas y su valor sobre la nota final y criterios de evaluación para cada prueba

El estudiante deberá demostrar que ha alcanzado los resultados de aprendizaje previstos mediante las siguientes actividades de evaluación

1. Los criterios de evaluación que se aplicarán son los siguientes:

1) Prácticas de Laboratorio (30 %)

Se calificarán en la propia sesión de laboratorio. Se valorará la preparación previa, el desarrollo de la sesión de laboratorio y la capacidad de puesta en marcha de este tipo de sistemas.

El estudiante que no supere las prácticas en el período docente, deberá realizar un examen de prácticas en el marco de las Pruebas Globales correspondientes a las Convocatorias Oficiales.

2) Trabajos y Actividades Evaluables (25%)

Con el fin de incentivar el trabajo continuado, se realizarán actividades evaluables distribuidas a lo largo del semestre. Dichas actividades se irán programando cada curso, consistiendo en trabajos en grupo, ejercicios individuales entregables, etc. Las actividades concretas a realizar se comunicarán en clase y en <http://moodle.unizar.es/>

Calificación global de 0 a 10 puntos, suponiendo un 25% de la calificación global.

El estudiante que no presente los entregables en las fechas que se establezcan durante el período docente, deberá superar la materia correspondiente en el marco de las Pruebas Globales a realizar en las Convocatorias Oficiales.

3) Examen Final (45 %)

Compuesto por cuestiones teórico-prácticas y problemas.

Calificación de 0 a 10 puntos; supondrá el 45 % de la calificación global del estudiante. Para superar la asignatura se debe obtener una calificación mínima de 4 puntos. Se valorará la corrección de las respuestas, desarrollos, diseños y resultados.

2. Prueba global (convocatorias oficiales; 100%)

En las convocatorias oficiales se llevará a cabo la evaluación global del estudiante. Quien haya superado las Prácticas y los Trabajos-Actividades Evaluables en el período docente, tan solo está obligado a realizar el Examen Final.

1) **Examen Final (45 %)**. Se realizará en el marco de las convocatorias oficiales. Para superar la asignatura se debe obtener una calificación mínima de 4 puntos sobre 10.

2) **Prueba sobre Trabajos y Actividades Evaluables (25 %)**, destinado a los estudiantes que no han superado esta parte en el período docente. Podrá realizarse solo si se ha obtenido más de 4 puntos en el Examen Final. La configuración de esta prueba se indicará en la convocatoria oficial de examen, pudiendo consistir en realizar un trabajo individual en el laboratorio, un examen escrito o cualquier otro formato que se indique. Para superar la asignatura se debe obtener una calificación mínima de 4.

3) **Examen de Laboratorio (30 %)**, destinado a los estudiantes que no han superado las prácticas en el período docente, se desarrollará en el laboratorio de prácticas. Podrá realizarse solo si se ha obtenido más de 4 puntos en el Examen Final. Para superar la asignatura se debe obtener una calificación mínima de 4 puntos.

La calificación final máxima de quienes no alcancen la calificación mínima exigida en alguna de las tres pruebas será de 4 puntos sobre 10.

4. Metodología, actividades de aprendizaje, programa y recursos

4.1. Presentación metodológica general

El proceso de aprendizaje que se ha diseñado para esta asignatura se basa en lo siguiente:

El proceso de enseñanza se desarrollará en los siguientes niveles: clases de teoría, problemas, trabajos y laboratorio, con creciente nivel de participación del estudiante.

- En las clases de teoría se expondrán las bases teóricas, ilustrándose con ejemplos.
- En las clases de problemas se desarrollarán problemas y casos tipo con la participación de los estudiantes.
- Se desarrollarán prácticas de laboratorio en grupos reducidos, donde el estudiante montará, programará y comprobará el funcionamiento de los sistemas.
- Asimismo, para incentivar el trabajo continuo y autónomo del estudiante, se llevarán a cabo actividades de aprendizaje adicionales a realizar a lo largo del semestre.

4.2. Actividades de aprendizaje

El programa que se ofrece al estudiante para ayudarle a lograr los resultados previstos comprende las siguientes actividades

1. Trabajo presencial: 2,4 ects (60 horas)

1) Clase presencial (tipo T1) (30 horas).

Sesiones expositivas de contenidos teóricos y prácticos. Se presentarán los conceptos y fundamentos de los sistemas operativos utilizados en sistemas empujados, ilustrándolos con ejemplos reales. Se fomentará la participación del estudiante.

2) Clases de problemas y resolución de casos (tipo T2) (12 horas).

Se desarrollarán problemas y casos con la participación de los estudiantes, coordinados con los contenidos teóricos. Se fomenta que el estudiante trabaje previamente los problemas. Parte de estas horas podrán dedicarse a las **actividades de aprendizaje evaluables** que se especifiquen en cada curso.

3) Prácticas de laboratorio (tipo T3) (18 horas).

El estudiante instalará, configurará y realizará modificaciones y medidas sobre dos sistemas operativos de características diferentes en laboratorio. Dispondrá de un guión de la práctica, que tendrá previamente que preparar. Cada práctica será calificada en el propio laboratorio.

2. Trabajo no presencial: 3,6 ects (90 horas)

1) Trabajos docentes (tipo T6) (30 horas).

Actividades que el estudiante realizará solo o en grupo y que el profesor irá proponiendo a lo largo del período docente. En esta asignatura cada estudiante realizará las actividades y trabajos que se propondrán durante el curso.

2) Estudio (tipo T7) (54 horas).

Estudio personal del estudiante de la parte teórica y realización de problemas. Se fomentará el trabajo continuo del estudiante mediante la distribución homogénea a lo largo del curso de las diversas actividades de aprendizaje. Se incluyen aquí las **tutorías**, como atención directa al estudiante, identificación de problemas de aprendizaje, orientación en la asignatura, atención a ejercicios y trabajos...

3) Pruebas de evaluación (tipo T8) (6 horas).

Además de la función calificadora, la evaluación también es una herramienta de aprendizaje con la que el alumno comprueba el grado de comprensión y asimilación alcanzado.

4.3. Programa

1. SOs Tiempo Real
 1. Introducción. Panorama actual.
 2. Revisión de métodos de planificación expulsiva con prioridades en sistemas TR
 3. Sincronización, inversión / herencia de prioridad
 4. Casos: TI SYS/BIOS y POSIX (pthreads)
3. Arquitectura de sistema
 1. Caso de estudio: ARM
 2. Gestión de interrupciones
 3. Factores que afectan a la latencia de interrupción
5. Linux/ARM como solución en empujados

1. Estructura y dinámica de un kernel Linux
 2. Modelos de expulsión en el kernel y sus consecuencias en empujados
 3. Sincronización en el kernel
 4. Excepciones. Latencia de interrupción.
 5. Callouts. Caso en Linux: tasklets, softirqs...
 6. Planificación / planificación tiempo real
 7. Gestión de Memoria en el kernel
 8. Gestión de ficheros y E/S - principales subsistemas
 9. Drivers (módulos) y acceso al hw - Linux Device model
- Prácticas de Laboratorio
 - Programación tiempo real sobre SYS/BIOS y POSIX
 - Diseño de una distribución Linux para un sistema empujado
 - Configuración y modificación del kernel
 - Medición de la latencia de interrupción
 - Proyectos de ampliación

4.4. Planificación de las actividades de aprendizaje y calendario de fechas clave

Calendario de sesiones presenciales y presentación de trabajos

Las clases magistrales y de problemas y las sesiones de prácticas en el laboratorio se imparten según horario establecido por el centro (horarios disponibles en su página web).

Cada profesor informará de su horario de atención de tutoría.

El resto de actividades se planificará en función del número de alumnos y se dará a conocer con la suficiente antelación. Podrá consultarse en <http://moodle.unizar.es>

El calendario detallado de las diversas actividades a desarrollar se establecerá una vez que la Universidad y el Centro hayan aprobado el calendario académico (consultar Web de la EINA).

La relación y fecha de las diversas actividades, junto con todo tipo de información y documentación sobre la asignatura, se publicará en <http://moodle.unizar.es/> (**Nota**). Para acceder a esta web el estudiante requiere estar matriculado).

A título orientativo:

- Cada semana hay programadas 3h de clases en aula.
- Se realizarán seis prácticas de laboratorio de 3h cada una.
- Las actividades adicionales que se programen (trabajos y otros) se anunciarán con suficiente antelación, tanto en clase como en <http://moodle.unizar.es/>.
- Las fechas de los exámenes y pruebas de convocatoria oficial las fijará la Dirección del Centro.

4.5. Bibliografía y recursos recomendados

<http://psfunizar7.unizar.es/br13/egAsignaturas.php?codigo=30240&Identificador=14705>

[BB: Bibliografía básica / BC: Bibliografía complementaria]

- [BC] Bovet, Daniel Pierre. Understanding the Linux Kernel / Daniel P. Bovet and Marco Cesati . - 3rd ed. Sebastopol [California] : O'Reilly, 2005
- [BC] Burns, Alan. Real-time systems and their programming languages / Alan Burns, Andy Wellings . - 2nd ed., repr. Harlow : Addison-Wesley, 1997
- [BC] Corbet, Jonathan. Linux device drivers / Jonathan Corbet, Alessandro Rubini, and Greg Kroah-Hartman . - 3rd ed. Beijing ; Sebastopol, CA : O'Reilly, 2005
- [BC] Love, Robert . Linux Kernel Development / Robert Love. 3rd Ed. Addison Wesley, 2010
- [BC] Pettazoni, Thomas. Device Tree for Dummies / Thomas Pettazoni
- [BC] Sloss, Andrew. ARM system developer's guide : designing and optimizing system software / Andrew Sloss, Dominic Symes, Chris Wright ; with a contribution by John Rayfield San Francisco, CA : Elsevier/ Morgan Kaufman, cop. 2004
- [BC] Venkateswaran, Sreekrishnan . Essential Linux Device Drivers / S. Venkateswaran. - 1st ed. Prentice Hall, 2008
- [BC] Yaghmour, Karim. Building embedded Linux systems / Karim, Yaghmour Beijing : O'Reilly, 2003

Listado de URL

- ARM Architecture Reference Manuals [<http://www.arm.com/>]
- Linux Kernel [<http://www.kernel.org>]
- Pettazoni ?Device Tree for Dummies?. Vídeo [https://youtu.be/m_NyYEBxfn8]
- TI SYS/BIOS kernel [<http://www.ti.com/tool/sysbios>]
- Transparencias (apuntes) de la asignatura. Hojas de problemas y Guiones de prácticas. [<http://moodle.unizar.es>]