

Sistemas empotrados y ubicuos

1er semestre, curso 2018-2019

Juan Zamorano

Profesores

- ◆ **José Luis Pedraza**
 - pedraza@fi.upm.es
- ◆ **Fernando Pérez**
 - fperez@fi.upm.es
- ◆ **Francisco Rosales**
 - frosal@fi.upm.es
- ◆ **Juan Zamorano**
 - jzamora@fi.upm.es

Objetivos de la asignatura

- ◆ Comprender los problemas específicos de los sistemas empuotrados y ubicuos, los conceptos básicos asociados, así como las características que los diferencian de otros sistemas informáticos.
- ◆ Conocer algunos lenguajes de programación y sistemas operativos adecuados para la realización de sistemas empuotrados.
- ◆ Conocer los métodos más importantes que se utilizan para desarrollar sistemas empuotrados con un grado de fiabilidad elevado, y comprender sus principios y formas de aplicación.
- ◆ Evaluar la idoneidad de una plataforma, hardware y software de base, para una determinada aplicación empuotrada

Enfoque

- ◆ Conceptos teóricos sobre sistemas empotrados: lenguajes de programación, sistemas operativos, hardware y métodos de diseño.
- ◆ Conceptos prácticos sobre sistemas operativos para sistemas empotrados.
 - Desarrollo cruzado
 - Desarrollo de manejadores de dispositivos
 - Desarrollo de módulos para el *kernel*
- ◆ Conceptos teóricos y estudio de casos de sistemas ubicuos.

Carga de trabajo

- ◆ 4.5 ECTS equivalentes a 121.5 horas (aproximadamente 7 horas semanales).
- ◆ Presencialidad de 2 horas semanales:
 - 28 horas de contenido teórico.
 - 11 horas de contenido práctico (6 en la franja de los lunes).
- ◆ Trabajo y estudio autónomo 5 horas semanales.

Programa: teoría

- ◆ Introducción a los sistemas empotrados.
- ◆ Diseño de sistemas empotrados.
- ◆ Desarrollo de manejadores de dispositivos.
- ◆ Sistemas operativos para sistemas empotrados.
- ◆ Hardware para sistemas empotrados.
- ◆ Sistemas de tiempo real.
- ◆ Sistemas ubicuos.

Programa: prácticas

- ◆ Desarrollo cruzado sobre máquina desnuda.
 - Demostración en laboratorio.
- ◆ Desarrollo de un módulo para el kernel de Linux.
 - Demostración en laboratorio y trabajo práctico.
- ◆ Práctica de Linux empotrado.
 - Demostración en laboratorio y trabajo práctico.
- ◆ Estudio de artículos sobre computación ubicua.
 - Lectura, análisis y presentación en clase.

Programa: proyecto

Desarrollo de un manejador para altavoz y construcción de un sistema Linux adaptado a esta funcionalidad.

- ◆ Desarrollo del proyecto con computadores compatibles PC con GNU/Linux para PC.
 - Se pueden usar portátiles si poseen el altavoz compatible PC.
 - Se dispone de computadores compatibles en el laboratorio.
- ◆ Desarrollo del proyecto con computadores RASPBERRY PI con GNU/Linux para ARM.
 - Se dispone de computadores RASPBERRY PI para los alumnos que elijan esta opción.

Evaluación “sólo examen final”

- ◆ **Plazo límite de solicitud: 30 de septiembre**
- ◆ **Examen**
 - Preguntas cortas y problemas.
- ◆ **Proyecto**
 - Memoria del proyecto.
 - Demostración y entrevista con el profesor encargado.

Evaluación “continua”

◆ Exámenes

- Primer parcial el 5 de noviembre (lunes).
- Segundo parcial el 14 de enero.
 - » Recuperación del primer parcial prevaleciendo esta última nota.
- Preguntas cortas y problemas.

◆ Proyecto

- Memoria del proyecto.
- Demostración y entrevista con el profesor encargado.

◆ Prácticas

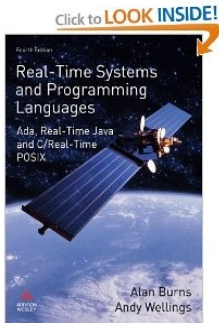
- Documentación de las prácticas.

Cálculo de la nota final

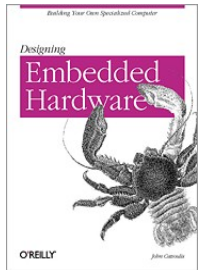
$$\text{nota final} = 0,5 * \text{nota de teoría} + 0,5 * \text{nota de prácticas}$$

- ◆ La nota de teoría es la media de las notas de ambos parciales o, en su caso, del examen final.
- ◆ La nota de prácticas es en un 80% la nota del proyecto y en un 20% la media de las prácticas de laboratorio.
 - 6.7% la práctica de sistemas empotrados, 6.7% la de programación de dispositivos y 6.7% la de computación ubicua.
 - La práctica de Linux empotrado es una introducción al proyecto.
- ◆ **La nota de teoría y nota de prácticas debe ser mayor o igual a 4.**

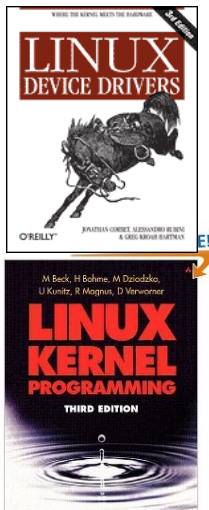
Bibliografía



Alan Burns and Andy Wellings
Real-Time Systems and Programming Languages
4th ed. Addison-Wesley, 2009



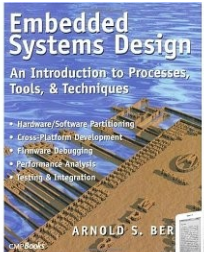
John Catsoulis
Designing Embedded Hardware
O'Reilly Media, 2002



Jonathan Corbet, Alessandro Rubini, and Greg Kroah-Hartman
LINUX Device Drivers
3rd ed. O'Reilly Media, 2005

Michael Beck, Harald Bohme , Mirko Dziadzka, Ulrich Kunitz, Robert Magnus and Dirk Verworner
Linux Kernel Programming
3rd ed. Addison-Wesley, 2002

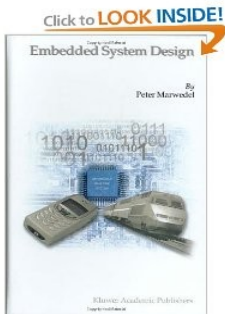
Bibliografía



Arnold S. Berger

Embedded Systems Design. An Introduction to Processes, Tools, and Techniques

CMP Books, 2002



Peter Marwedel

Embedded System Design

Springer, 2006

Ficheros fuentes y documentación de sistemas operativos abiertos:

Linux: <http://www.linux.org>

ORK: <http://polaris.dit.upm.es/~ork>

Xtratum: <http://www.xtratum.org>

MaRTEOS: <http://marte.unican.es>

RTEMS: <http://www.rtems.com>

GNAT for Mindstorms: <http://libre.adacore.com/libre/tools/mindstorms>

Otros recursos

- ◆ Web de la asignatura
 - <http://www.datsi.fi.upm.es/docencia/SEUM>
- ◆ Laboratorio de sistemas empotrados y tiempo real
 - Sala Alpera. Bloque 4 planta baja.