Universidad Zaragoza
1542

Year : 2018/19

30240 - Embedded Systems II

Syllabus Information

Academic Year:	2018/19
Subject:	30240 - Embedded Systems II
Faculty / School:	110 -
Degree:	439 - Bachelor's Degree in Informatics Engineering
ECTS:	6.0
Year:	4
Semester:	Indeterminate
Subject Type:	
Module:	---

General information

Aims of the course

Context and importance of this course in the degree

Recommendations to take this course

Learning goals

Competences

Learning goals

Importance of learning goals

Assessment (1st and 2nd call)

Assessment tasks (description of tasks, marking system and assessment criteria)

Methodology, learning tasks, syllabus and resources

Methodological overview

The learning process designed for this course relies on lectures, problems, assignments and computer lab sessions.

Subject overview and theoretical fundations are introduced in lectures, closely referred to case studies.

Problems sessions foster the participation of the students.

Lab sessions are organized in reduced groups, where the student sets up, program and check the systems.

In order to foster continuous and autonomous learning, there can be additional learning activities throughout the course's period.

Learning tasks

The course's program encompasses the following activities

1. On-site activities: 2.4 ECTS (60 hours)

a) Regular classes and lectures (30 h)

Presentation of theoretical and practical contents, illustrated with case studies and fostering the participation of the students.

b) Problem-solving sessions (12 h)

The students will get involved in problem solving related to the theoretical contents. The students will usually have to prepare the problems in advance. Part of the time could be devoted to learning activities that can be graded, as specified each course.

c) Lab sessions (18 h)

The student will install, set up and make modifications and measurements on two different operating systems, a classic RTOS and a general-purpose OS leveraged for the purpose of an embedded system. The student will have a laboratory guide and will have to prepare the activity in advance. Each practical will be graded on-site.

2. Off-site activities: 3.6 ECTS (90 h)

a) Assigments (30 h)

Activities the student will do individually or working in group and that the instructor will propose throughout the course's time. In this course every student will have to complete the proposed assigments.

b) Study (54 h)

Student's personal time for studying the subject and for problem solving. This encompases office hours to assist the student to identify learning problems, and provide advice on the course and its activities.

c) Assesment tasks (6 h)

In addition to the grading role, evaluation constitutes a learning tool what the student checks how much she/he has learned and understood.

Syllabus

Embedded Systems II

1. Real-Time Operating Systems
    1. Overview
    2. Preemptive real-time scheduling on monoprocessors
    3. Synchronization, priority inversion and inheritance
    4. Case Study: TI SYS/BIOS and POSIX (pthreads)
2. System Architecture
    1. Case study: ARM
    2. Exception management
    3. Factors involved in the interruption latency
3. Linux/ARM for embedded systems
    1. Linux Kernel basics
    2. Preemption models and their consequences in embedded systems
    3. Kernel synchronization
    4. Exceptions. Interruption latency in Linux
    5. Callouts. Case study: tasklets, softirqs...
    6. Task scheduling and RT scheduling in Linux
    7. Kernel memory management
    8. Files and I/O subsystems
    9. Linux Drivers. The Linux Device model

- Lab sessions
  - RT programming on SYS/BIOS and POSIX
  - Design of a Linux distribution for an embedded system
  - Kernel programming and configuration
  - Interruption Latency measurement
- Course projects

Course planning and calendar

Classes, problems and laboratory sessions are scheduled according to EINA's calendar (check EINA's website at http://eina.unizar.es)

Instructor's office hours are available at the EINA's website and Moodle (http://moodle.unizar.es).

The remaining activities will be scheduled according to the number of students, will be announced well in advance, and will appear on the course site at Moodle.

Bibliography and recommended resources

[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 : Oâ??Reilly, 2005
- [BC] Love, Robert . Linux Kernel Development / Robert Love. 3rd Ed. Addison Wesley, 2010
- [BC] Pettazzoni, Thomas. Device Tree for Dummies / Thomas Pettazzoni
- [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]
- Pettazzoni ?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]