Course guides
230116 - DSX - Digital Systems Using Embedded Linux

Unit in charge: Barcelona School of Telecommunications Engineering
Teaching unit: 710 - EEL - Department of Electronic Engineering.
Degree: BACHELOR’S DEGREE IN ELECTRONIC SYSTEMS ENGINEERING (Syllabus 2009). (Optional subject).
BACHELOR’S DEGREE IN TELECOMMUNICATIONS SCIENCE AND TECHNOLOGY (Syllabus 2010). (Optional subject).
BACHELOR’S DEGREE IN TELECOMMUNICATIONS TECHNOLOGIES AND SERVICES ENGINEERING (Syllabus 2015). (Optional subject).
BACHELOR’S DEGREE IN TELECOMMUNICATIONS SYSTEMS ENGINEERING (Syllabus 2010). (Optional subject).
BACHELOR’S DEGREE IN AUDIOVISUAL SYSTEMS ENGINEERING (Syllabus 2009). (Optional subject).
BACHELOR’S DEGREE IN NETWORK ENGINEERING (Syllabus 2010). (Optional subject).

Academic year: 2019  ECTS Credits: 6.0  Languages: Catalan, English, Spanish

LECTURER
Coordinating lecturer: MANUEL DOMÍNGUEZ
Others: JUAN A. CHÁVEZ, VICENTE JIMÉNEZ, JOAN PONS

PRIOR SKILLS
Knowledge of the programming language C.

TEACHING METHODOLOGY
- Lectures.
- Laboratory classes.
- Group work (distance learning).
- Oral presentations.
- Short exams.

LEARNING OBJECTIVES OF THE SUBJECT
- To know the main characteristics of embedded systems.
- To know how to schedule and manage Linux-based embedded systems, including the design of low-level software to control hardware components (drivers).
- To program and develop an interface between a Linux-based embedded system and a FPGA-based digital system.
- To know design criteria for timing clock signals management and frequency synthesis applied to programmable devices.
- To know signal processing techniques with FPGAs.
- To learn techniques of control and communication with various peripherals: ADC, Memory, etc.

STUDY LOAD

<table>
<thead>
<tr>
<th>Type</th>
<th>Hours</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hours large group</td>
<td>13,0</td>
<td>8.67</td>
</tr>
<tr>
<td>Hours small group</td>
<td>39,0</td>
<td>26.00</td>
</tr>
<tr>
<td>Self study</td>
<td>98,0</td>
<td>65.33</td>
</tr>
</tbody>
</table>
Total learning time: 150 h

## CONTENTS

### 1. Introduction to Linux embedded systems

**Description:**
- Introduction to embedded systems: concept, system kernel, boards and development tools, process design, market and major manufacturers.
- Introduction to GNU Linux system. File system, / proc file system.
- Shell Linux commands and scripts.
- Compiling, running and debugging programs on Linux embedded systems.
- Concurrency basics: scheduler, preemptive kernels.

**Full-or-part-time:** 20h  
Theory classes: 3h  
Laboratory classes: 4h  
Self study: 13h

### 2. Linux programming tools

**Description:**
- Concurrent programming: threads, forks.
- Process synchronization: semaphores POSIX, Mutexes.
- Pipes.
- Semaphores System V.

**Full-or-part-time:** 38h  
Theory classes: 2h  
Laboratory classes: 11h  
Self study: 25h

### 3. Driver development

**Description:**
Using drivers:
- Control of GPIO pins with virtual files, UARTs, etc.
- Communication with standard buses (SPI, I2C, etc.)
- Module-based utilities.

Programming drivers:
- System calls: open, read, write, close, ioctl.
- Memory management.
- Managing interruptions. Managing system hardware.
- Driver control of hardware programmed in FPGAs.

**Full-or-part-time:** 23h  
Theory classes: 2h  
Laboratory classes: 6h  
Self study: 15h
4. System configuration

Description:
- Kernel compilation,
- System management, selection of applications, boot scripts.
- Source code control: GIT.
- Installation and U-Boot configuration.

Full-or-part-time: 23h
Theory classes: 2h
Laboratory classes: 6h
Self study : 15h

5. Design technics for FPGAs

Description:
- System timing, clock management and frequency synthesis.
- Connectivity (buses).
- Signal processing with FPGAs.
- Peripherals: ADC, memories.

Full-or-part-time: 46h
Theory classes: 4h
Laboratory classes: 12h
Self study : 30h

ACTIVITIES

LABORATORY

Description:
Programming a Linux-based Beaglebone board and a DE-2 FPGA board, first separately and then connecting them. There will be an initial-guided lab work where programming techniques presented in class will be used. The second phase of the lab work consist either in improving the previous work or in programming an original-open project.

Full-or-part-time: 39h
Laboratory classes: 39h

(ENG) PRESENTACIONS ORALS

Description:
Oral presentation of the project done in the laboratory.

Full-or-part-time: 0h 30m
Theory classes: 0h 30m
GRADING SYSTEM

The final grade for the course comes from the course assessment (exams, work done during the course and laboratory work) and the final exam, applying the following scale:
Final exam: 15%
Controls: 10%
Projects and lab work: 75%

BIBLIOGRAPHY

Basic: