



INTERNATIONAL
CAMPUS OF
EXCELLENCE

COORDINATION PROCESS OF
LEARNING ACTIVITIES
PR/CL/001



E.T.S. de Ingeniería y Sistemas
de Telecomunicación

ANX-PR/CL/001-01

LEARNING GUIDE

SUBJECT

595310047 - Embedded Systems Design With Raspberry Pi

DEGREE PROGRAMME

59ET - Doble Grado En Ing.Electronica De Comunicaciones Y En Ing.Telematica

ACADEMIC YEAR & SEMESTER

2021/22 - Semester 2

Index

Learning guide

1. Description.....	1
2. Faculty.....	1
3. Prior knowledge recommended to take the subject.....	2
4. Skills and learning outcomes	2
5. Brief description of the subject and syllabus.....	4
6. Schedule.....	5
7. Activities and assessment criteria.....	8
8. Teaching resources.....	11

1. Description

1.1. Subject details

Name of the subject	595310047 - Embedded Systems Design With Raspberry Pi
No of credits	4.5 ECTS
Type	Optional
Academic year of the programme	Fifth year
Semester of tuition	Semester 10
Tuition period	February-June
Tuition languages	English
Degree programme	59ET - Doble Grado en Ing.electronica de Comunicaciones y en Ing.telematica
Centre	59 - Escuela Tecnica Superior De Ingenieria Y Sistemas De Telecomunicacion
Academic year	2021-22

2. Faculty

2.1. Faculty members with subject teaching role

Name and surname	Office/Room	Email	Tutoring hours *
Mariano Ruiz Gonzalez	A4206	mariano.ruiz@upm.es	Sin horario.
Sergio Esquembri Martinez (Subject coordinator)	A4206	s.esquembri@upm.es	Sin horario.

* The tutoring schedule is indicative and subject to possible changes. Please check tutoring times with the faculty member in charge.

3. Prior knowledge recommended to take the subject

3.1. Recommended (passed) subjects

- Microprocesadores
- Programacion li
- Sistemas Basados En Microprocesador
- Sistemas Operativos
- Programacion I

3.2. Other recommended learning outcomes

- Basic computer networks knowledge

4. Skills and learning outcomes *

4.1. Skills to be learned

CE EC01 - Capacidad de construir, explotar y gestionar sistemas de captación, transporte, representación, procesado, almacenamiento, gestión y presentación de información multimedia, desde el punto de vista de los sistemas electrónicos.

CE EC03 - Capacidad de realizar la especificación, implementación, documentación y puesta a punto de equipos y sistemas, electrónicos, de instrumentación y de control, considerando tanto los aspectos técnicos como las normativas reguladoras correspondientes.

CE EC04 - Capacidad para aplicar la electrónica como tecnología de soporte en otros campos y actividades, y no sólo en el ámbito de las Tecnologías de la Información y las Comunicaciones.

CE EC05 - Capacidad de diseñar circuitos de electrónica analógica y digital, de conversión analógico-digital y digital-analógica, de radiofrecuencia, de alimentación y conversión de energía eléctrica para aplicaciones de telecomunicación y computación.

CE EC07 - Capacidad para diseñar dispositivos de interfaz, captura de datos y almacenamiento, y terminales para

servicios y sistemas de telecomunicación.

CE EC08 - Capacidad para especificar y utilizar instrumentación electrónica y sistemas de medida.

CG 02 - Capacidad de búsqueda y selección de información, de razonamiento crítico y de elaboración y defensa de argumentos dentro del área.

CG 03 - Capacidad para expresarse correctamente de forma oral y escrita y transmitir información mediante documentos y exposiciones en público.

CG 04 - Capacidad de abstracción, de análisis y de síntesis y de resolución de problemas.

4.2. Learning outcomes

RA492 - Conectar un circuito electrónico básico en uno de los interfaces digitales de la RaspBerry PI.

RA487 - Identificar la funcionalidad de cada uno de los interfaces digitales y analógicos que incluye la RaspBerry-PI.

RA489 - Conocer los elementos integrantes de una distribución de Linux para un sistema empotrado.

RA495 - Redactar documentos técnicos presentando los pasos seguidos y las conclusiones obtenidas en la realización de una aplicación.

RA493 - Desarrollar una aplicación software básica utilizando los interfaces de la RaspBerry PI.

RA494 - Presentar y defender en público propuestas técnicas para resolver problemas.

RA491 - Documentar el desarrollo de una aplicación con RaspBerry-PI y presentarla en público.

RA488 - Instalar un sistema operativo Linux y aplicaciones software en la RaspBerry Pi.

RA490 - Configurar y construir una distribución del sistema operativo Linux utilizando la herramienta Buildroot para la plataforma RaspBerry-PI.

RA486 - Conocer las características hardware básicas de un sistema electrónico embebido como la RaspBerry PI basado en un System On Chip

* The Learning Guides should reflect the Skills and Learning Outcomes in the same way as indicated in the Degree Verification Memory. For this reason, they have not been translated into English and appear in Spanish.

5. Brief description of the subject and syllabus

5.1. Brief description of the subject

Embedded Systems Design with Raspberry Pi is a tenth-semester elective course for the Telematics and Communications Electronics Engineering Double B. Eng. program. This course introduces the students to the use of embedded electronic systems for practical application development. During the course, students will learn to develop and embed their own Linux-based operating system in a very low-cost embedded system like the Raspberry Pi. The course is organized as laboratory assignments introduced by theory lessons. The laboratory assignments are designed to emphasize application development and integration skills, as well as the presentation of results in an oral and written manner.

5.2. Syllabus

1. Introduction to Raspberry Pi: Architecture and hardware resources
2. Raspberry Pi Linux Application development
 - 2.1. Raspberry Pi Operating System installation
 - 2.2. Linux basic use tutorial
 - 2.3. (Optional) Install and configure Ubuntu Virtual Machine
 - 2.4. Connecting sensors and actuators to Raspberry Pi
 - 2.5. C application development for Raspberry Pi
3. Generation of custom embedded Linux systems
 - 3.1. Embedded Linux System Fundamentals
 - 3.2. Embedded Linux System generation tools: Buildroot
4. Embedded Linux Application development
 - 4.1. Environment configuration for remote connection, development, deployment, testing, and debug
5. IoT Application development using Raspberry Pi and Embedded Linux
 - 5.1. IoT fundamentals: Protocols and use cases
 - 5.2. Raspberry Pi for IoT

6. Schedule

6.1. Subject schedule*

Week	Face-to-face classroom activities	Face-to-face laboratory activities	Distant / On-line	Assessment activities
1	<p>Course presentation Duration: 01:00</p> <p>Introduction to Raspberry Pi: Architecture and hardware resources Raspberry Pi Operating System Installation Duration: 02:00</p>			
2	<p>Linux basic use tutorial Duration: 00:30</p> <p>Connecting Sensors and actuators to Raspberry Pi Duration: 00:30</p>	<p>Raspberry Pi Operating System Installation Duration: 00:30</p> <p>C application development for Raspberry Pi Duration: 01:30</p>		
3		<p>C application development for Raspberry Pi Duration: 03:00</p>		
4				<p>C application development for Raspberry Pi: Questionary</p> <p>Continuous assessment Presential Duration: 00:10</p> <p>C application development for Raspberry Pi: Alumni presentations and questions by instructors</p> <p>Continuous assessment Presential Duration: 02:50</p>
5	<p>Embedded Linux System Fundamentals Duration: 03:00</p>			
6	<p>Embedded Linux System Generation Tools: Buildroot Duration: 01:30</p>	<p>Tutorial: Build custom embedded Linux system for Raspberry Pi Duration: 01:30</p>		

7		Tutorial: Build custom embedded Linux system for Raspberry Pi Duration: 03:00		
8	Environment configuration for remote connection, development, deployment and testing Duration: 01:00	Tutorial: Environment configuration for remote connection, development, deployment and testing Duration: 02:00		
9	IoT Application Development using Raspberry Pi and Embedded Linux Duration: 02:00	IoT application development Duration: 01:00		
10				<p>Generation of Custom Embedded Linux_ Questionary</p> <p>Continuous assessment Presential Duration: 00:05</p> <p>Generation of Custom Embedded Linux Systems: Alumni presentations and questions by instructors</p> <p>Continuous assessment Presential Duration: 02:55</p>
11		IoT application development Duration: 03:00		
12		IoT application development Duration: 03:00		
13				<p>IoT Application Development: Questionary</p> <p>Continuous assessment Presential Duration: 00:05</p> <p>IoT Application development: Alumni presentations and questions by instructors</p> <p>Continuous assessment Presential Duration: 02:55</p>
14				
15				
16				
17				<p>Final Assesment</p> <p>Continuous assessment Presential Duration: 01:00</p> <p>Application desing and development during days previous to the exam. Oral and writing exam. The exam will be conditioned to the necessary to the completion of the application proposed.</p>

				Final examination Presential Duration: 03:00
--	--	--	--	--

Depending on the programme study plan, total values will be calculated according to the ECTS credit unit as 26/27 hours of student face-to-face contact and independent study time.

* The schedule is based on an a priori planning of the subject; it might be modified during the academic year, especially considering the COVID19 evolution.

7. Activities and assessment criteria

7.1. Assessment activities

7.1.1. Continuous assessment

Week	Description	Modality	Type	Duration	Weight	Minimum grade	Evaluated skills
4	C application development for Raspberry Pi: Questionary		Face-to-face	00:10	5%	/ 10	CG 04 CG 02 CG 03
4	C application development for Raspberry Pi: Alumni presentations and questions by instructors		Face-to-face	02:50	10%	5 / 10	CG 02 CG 03 CG 04 CE EC01 CE EC03 CE EC04 CE EC05 CE EC07 CE EC08
10	Generation of Custom Embedded Linux_ Questionary		Face-to-face	00:05	5%	/ 10	CG 02 CG 03 CG 04
10	Generation of Custom Embedded Linux Systems: Alumni presentations and questions by instructors		Face-to-face	02:55	20%	5 / 10	CG 04 CE EC01 CE EC03 CG 02 CG 03 CE EC05
13	IoT Application Development: Questionary		Face-to-face	00:05	5%	/ 10	CG 02 CG 03 CG 04
13	IoT Application development: Alumni presentations and questions by instructors		Face-to-face	02:55	40%	5 / 10	CG 02 CG 03 CG 04 CE EC03 CE EC04 CE EC05 CE EC07 CE EC08
17	Final Assesment		Face-to-face	01:00	15%	/ 10	CG 03 CG 04 CE EC01 CE EC04 CE EC07

7.1.2. Final examination

Week	Description	Modality	Type	Duration	Weight	Minimum grade	Evaluated skills
17	Application desing and development during days previous to the exam. Oral and writing exam. The exam will be conditioned to the necessary to the completion of the application proposed.		Face-to-face	03:00	100%	5 / 10	CG 04 CE EC01 CE EC03 CE EC04 CE EC05 CE EC07 CE EC08 CG 02 CG 03

7.1.3. Referred (re-sit) examination

Description	Modality	Type	Duration	Weight	Minimum grade	Evaluated skills
Application desing and development during days previous to the exam. Oral and writing exam. The exam will be conditioned to the necessary to the completion of the application proposed.		Face-to-face	03:00	100%	5 / 10	CG 02 CG 03 CG 04 CE EC01 CE EC03 CE EC04 CE EC05 CE EC07 CE EC08

7.2. Assessment criteria

The following lists present the evaluation indicators that will be used for both the continuous assessment. The criteria used for the final examination and the referred (re-sit) examination will also be the same. For the last cases, the students will be asked to generate a custom Linux operating system and develop an application for Raspberry Pi during the 3 (three) hours of the exam, as well as completing a short writing exam.

Evaluation indicators:

- The student identifies the main hardware components of a Raspberry Pi.
- The student identifies the main characteristics of a BCM2835 chip or equivalent.
- The student identifies the functionalities of the different hardware interfaces available in the Raspberry Pi
- (Optional) The student is capable of installing and configuring a virtual machine with all the components required software packages.
- The student is capable of searching, downloading, installing, and booting a commercial operating system for Raspberry Pi.
- The student identifies the main functional blocks necessary for building an embedded Linux system.
- The student identifies the tools necessary for using Buildroot.
- The student identifies the main configuration parameters for building an embedded Linux system using Buildroot.
- The student knows the purpose and functionality of a bootloader.
- The student knows the purpose and functionality of the Linux kernel
- The student knows the purpose and functionality of a filesystem.
- The student knows the purpose and functionality of Busybox inside an embedded Linux system.
- The student is capable of configuring the host system for building an embedded Linux system using Buildroot
- The student is capable of booting a custom embedded Linux operating system with the Raspberry Pi
- The student is capable of changing the Buildroot configuration parameters to adapt to their specific requirements.
- The student is capable of identifying and fixing errors derived from the start-up of a custom embedded Linux system.
- The student identifies and understands the characteristics of electronic sensors and actuators with a digital interface
- The student is capable of connecting sensors and actuators to the proper Raspberry Pi interfaces.
- The student is capable of configuring the host system for remote connection, development, deployment, testing, and debugging of applications for the Raspberry Pi.
- The student is capable of design, develop and debug a C application for Raspberry Pi using sensors and

actuators.

- The student is capable of design, develop and debug an advanced application based on Raspberry Pi.
- The student is capable of presenting the Raspberry Pi characteristics and set-up to his peers.
- The student is capable of explaining how basic applications can be developed and executed in the Raspberry Pi.
- The student is capable of presenting the configuration, development, deployment and start-up cycle of a custom embedded Linux system for Raspberry Pi.
- The student is capable of preparing formal documentation detailing the steps followed and the decisions taken to develop an embedded Linux System for Raspberry Pi.
- The student is capable of preparing detailed documentation about the development and deployment of an advanced application using Raspberry Pi
- The student is capable of team working for joint project development.

8. Teaching resources

8.1. Teaching resources for the subject

Name	Type	Notes
Raspberry Pi Board	Equipment	Raspberry development board
Host Computer	Equipment	Personal computer with Windows operating System. It will require the installation of an Ubuntu Virtual Machine
Moodle	Web resource	UPM Course Moodle with the course support materials
Raspberry Pi Foundation	Web resource	http://www.raspberrypi.org
VMWare	Web resource	http://www.wmware.com
Unix Tutorial	Web resource	http://www.ee.surrey.ac.uk/Teaching/Unix
Buidroot	Web resource	http://buildroot.uclibc.org/