Universitat Autònoma de Barcelona

2022/2023

Microprocessors and Peripherals

Code: 102793
ECTS Credits: 6

Degree	Type	Year	Semester
2502441 Computer Engineering	OB	3	2
2502441 Computer Engineering	OT	4	2

Contact

Name: Dolores Isabel Rexachs del Rosario
Email: dolores.rexachs@uab.cat

Use of Languages

Principal working language: spanish (spa)
Some groups entirely in English: No
Some groups entirely in Catalan: No
Some groups entirely in Spanish: No

Other comments on languages

Se utiliza material en inglés

Teachers

Vicente José Ivars Camañez

External teachers

Eduardo Cabrera

Prerequisites

There are no prerequisites. It is also recommended for students to have notions of Computer Structure and Computer Architecture.

Objectives and Contextualisation

This course, of the Computer Engineering Mention, is in the third course, the second semester of the degree, within the subject "Design of computer systems oriented to applications".

"Microprocessors and Peripherals" is related to the subjects of Computer Fundamentals, Operating Systems, Computer Structure, and Computer Architecture.

The aim of the course is for students to understand the fundamental components used in the design of systems based on digital processors (microprocessors and peripheral devices) and how they are interconnected, analyzing the different interfaces.

Apply knowledge of computer architecture and system design to select the features of the microcontroller, peripherals and peripheral controllers best suited to the needs of the application.

Select the most suitable platform for the design of a system for a specific application and design and develop the solution based on the selected microcontroller.

Competences

Computer Engineering

- Have the capacity to define, evaluate and select hardware and software platforms for the development and execution of computer systems, services and applications.
- Have the capacity to design and build digital systems, including computers, microprocessor based systems and communication systems.
- Have the capacity to develop specific processors and embedded systems and to develop and optimise the software of said systems.
- Have the right personal attitude.

Learning Outcomes

1. Apply knowledge of computer architecture and system design in order to select the processor or embedded system characteristics that best adapt to the needs of the application.
2. Classify the different types of digital system.
3. Develop curiosity and creativity.
4. Identify the possible digital system based architectures for the design of microprocessor based computing systems.
5. Select the most suitable platform for a specific application and design and develop the solution based on the corresponding microprocessor.

Content

Block 1. Design of computer systems based on microprocessors and microcontrollers.

- General purpose digital processors for the design of systems based on applications: Microcontrollers, DSP
- Design Methods
- Selection criteria

Block 2. Peripheral devices. Storage systems.

- Input Peripherals
- Output Peripherals
- Storage Devices

Block 3. Interconnection of peripheral devices

- Interconnection of components: Buses
- Interfaces: Standards, usability.

Block 4. Performance evaluation.

- Specification and selection of metrics to evaluate performance, reliability, availability, sustainability, and energy efficiency.
- Impact of performance, reliability, and energy consumption on design and implementation.
- Methods and models to evaluate performance.

Methodology

The teaching methodology followed in the subject is based on a series of training activities that require the presence of the student in the classroom or laboratory, and a series of individual activities that require personal work on the part of the student.

Training activities

- Theoretical lectures
- Exercise-based classes
- Mandatory laboratory classes: They will be carried out in a specific laboratory of the subject
- Supervised activities: Tutoring and consultations
- Autonomous: Preparing exercises and practical assignments. Other independent studies
- There is no differentiated treatment for repeating students.

Annotation: Within the schedule set by the centre or degree programme, 15 minutes of one class will be reserved for students to evaluate their lecturers and their courses or modules through questionnaires.

Activities

Title	Hours	ECTS	Learning Outcomes
Type: Directed			
Exercises classes	12	0.48	1, 2, 5
Laboratory classes and Final test	6	0.24	1, 2, 4, 5
Mandatory laboratory classes	12	0.48	3, 5
Theory classes	22	0.88	1, 2, 4, 5
Tutoring and consults	2	0.08	3
Type: Supervised			
Exercises and practices preparation	6	0.24	1, 3
Exercises, reports, and practical assignements	2	0.08	
Type: Autonomous			
Personal work	70	2.8	1, 2, 4, 5
Preparation of works and reports of laboratory classes	10	0.4	1, 3, 5

Assessment

The evaluation of the degree of acquisition of the competencies on the part of the students is carried out on:

- Two individual tests. As part of continuous assessment, the first test will take place during lectures; the second (Final exam) will take place on the date specified by coordination.
- Exercise resolutions. As part of continuous assessment, activities must be carried out or exercises must be solved (via online quizzes). The non-online activities or exercises will take place during the exercise sessions.
- The activity carried out in the laboratory.

An overall grade of 5 or higher is required to pass the subject. The laboratory practices cannot be approved without a minimum attendance of 85% at the practice sessions and the reports delivered and approved. In

order to pass the course, it will be necessary to obtain a minimum grade of 5 in the activities. Keep in mind that the activity "Laboratory practices" is non-recoverable, therefore suspending it with a grade lower than 5, means not being able to pass the course.

Continuous-assessment dates will be published on Campus Virtual and on the presentation slides, specific programming may change when necessary. Any such modification will always be communicated to students through Campus Virtual, which is the usual communication platform between lecturers and students. To participate in the recovery process it is necessary to have obtained a minimum grade of 2.5 in the average of the subject.

In order to pass the course with honours, the final grade must be a 9.0 or higher. Because the number of students with this distinction cannot exceed 5% of the number of students enrolled in the course, this distinction will be awarded to whoever has the highest final grade.

A "non-assessable" grade cannot be assigned to students who have participated in any of the individual partial tests or the final exam.

No special treatment will be given to students who have completed the course in previous academic years

Notwithstanding other disciplinary measures deemed appropriate, and in accordance with the academic regulations in force, assessment activities will receive a zero whenever a student commits academic irregularities that may alter such assessment. Assessment activities graded in this way and by this procedure will not be re-assessable. If passing the assessment activity or activities in question is required to pass the subject, the awarding of a zero for disciplinary measures will also entail a direct fail for the subject, with no opportunity to re-assess this in the same academic year.

Assessment Activities

Title	Weighting	Hours	ECTS	Learning Outcomes
Exercises resolution	20%	2	0.08	1, 2, 3, 4
Individual tests	50%	6	0.24	1, 2, 4, 5
Mandatory laboratory	30%	0	0	1, 3, 5

Bibliography

Marilyn Wolf (2012) Computers as Components: Principles of Embedded Computing System Design. 3rd Edition. Morgan Kaufmann.

Sarah Harris, David Harris. (2015) Diseño Digital y Arquitectura de Computadores: ARM Edición. Morgan Kaufmann. Elsevier Science & Technology.

Antonio Díaz Estrella. TEORIA Y DISEÑO CON MICROCONTROLADORES DE FREESCALE. (2008) MCGRAW-HILL. ISBN 9788448170882

Muhammad Ali Mazidi; Shujen Chen; Sarmad Naimi; Sepehr Naimi. Freescale ARM Cortex-M Embedded. Programming Using C Language. (2014) Kindle Edition. Published October 31st 2014 by Mazidi & Naimi

Elecia White. (2011). Making Embedded Systems: Design Patterns for Great Software. O'Reilly Media, Inc.

Christopher Kormanyos. (2015). Real-Time C++: Efficient Object-Oriented and Template Microcontroller Programming. Springer

Joseph Yiu. (2011). The Definitive Guide to the ARM Cortex-M0 Elsevier. / Yiu, Joseph, Llibre en línia.

Software

Code Warrior (FRDM-KL25Z)

C Compiler (gcc)

Assembler (ARM)