



| Teaching Guide | | | | |
|---------------------|---|--------|-----------------------|---------|
| Identifying Data | | | | 2017/18 |
| Subject (*) | Hardware/Software Co-Design | Code | 614G01031 | |
| Study programme | Grao en Enxeñaría Informática | | | |
| Descriptors | | | | |
| Cycle | Period | Year | Type | Credits |
| Graduate | 2nd four-month period | Third | Optativa | 6 |
| Language | SpanishGalicianEnglish | | | |
| Teaching method | Face-to-face | | | |
| Prerequisites | | | | |
| Department | Enxeñaría de Computadores | | | |
| Coordinador | Rodríguez Osorio, Roberto | E-mail | roberto.osorio@udc.es | |
| Lecturers | Rodríguez Osorio, Roberto | E-mail | roberto.osorio@udc.es | |
| Web | | | | |
| General description | Currently, a large majority of computing systems are embedded, where hardware and software design go together. In these systems, the whole is larger than the sum of the parts. Therefore, design and testing procedures are not restricted to the hardware and software components, but they also include the interface between them. This subject addresses the world of codesign by focusing on several aspects such as: reconfigurable computing; system modeling; and application-specific processors. | | | |

| Study programme competences / results | |
|---------------------------------------|--|
| Code | Study programme competences / results |
| A31 | Capacidade de deseñar e construír sistemas dixitais, incluíndo computadores, sistemas baseados en microprocesador e sistemas de comunicacións. |
| A32 | Capacidade de desenvolver procesadores específicos e sistemas embarcados, así como desenvolver e optimizar o software dos ditos sistemas. |
| B1 | Capacidade de resolución de problemas |
| B3 | Capacidade de análise e síntese |
| C7 | Asumir como profesional e cidadán a importancia da aprendizaxe ao longo da vida. |

| Learning outcomes | | | |
|---|---------------------------------------|----------|----|
| Learning outcomes | Study programme competences / results | | |
| To understand the principles, methods and tools essential to hardware-software codesign | | B3 | C7 |
| To know the main techniques for designing reconfigurable hardware, understanding their advantages and limitations | A31 | | C7 |
| To learn to decide which methods and algorithms should be implemented in software, and which ones on hardware. To know to realize the interface between both. | A32 | B1 B3 | |
| To get to know which design scenarios would benefit of a solution based on reconfigurable hardware | | B1 B3 | |

| Contents | |
|---|---|
| Topic | Sub-topic |
| Fundamentals and Platforms for hardware/software codesign | Definition of codesign Application-specific hardware and reconfigurable hardware |
| Hardware/Software Codesign | Transaction and data flow level modeling Time-accurate modeling |
| Data-flow and control-flow modelling | Data -flow modeling and implementation Analysis of Control Flow and Data Flow |



| | |
|---|--|
| Application-specific instruction-set processors | Accelerators and coprocessors Systems on a chip (SoC) |
|---|--|

| Planning | | | | |
|--------------------------------|------------------------|--------------------------------------|-------------------------------|-------------|
| Methodologies / tests | Competencies / Results | Teaching hours (in-person & virtual) | Student's personal work hours | Total hours |
| Laboratory practice | A31 A32 B1 | 14 | 34 | 48 |
| Supervised projects | A31 B1 B3 C7 | 7 | 25 | 32 |
| Objective test | B1 B3 | 3 | 0 | 3 |
| Guest lecture / keynote speech | A31 A32 C7 | 21 | 42 | 63 |
| Personalized attention | | 4 | 0 | 4 |

(*)The information in the planning table is for guidance only and does not take into account the heterogeneity of the students.

| Methodologies | |
|--------------------------------|---|
| Methodologies | Description |
| Laboratory practice | Labs: A set of guided lab tasks will be assigned to the students. The aim is practicing the basic procedures of the subject and reflecting on them. The topic of the labs is linked to the guided projects. |
| Supervised projects | Guided projects: Students must work in small groups to complete hardware/software codesign projects. During the seminars, project coordination will be carried out, where the progress of each project will be assessed. However, most of the work must be done by the students in an autonomous way. |
| Objective test | Final test: A written test, lasting up to 3 hours, must be passed by the end of the course. |
| Guest lecture / keynote speech | Lectures: They will be focused on the different topics of the subject. The progress of the lectures will define the scheduling of the labs and seminars. |

| Personalized attention | |
|--|---|
| Methodologies | Description |
| Laboratory practice Supervised projects | Personalized attention is crucial for guiding the students when doing exercises, performing the labs, and working on projects. Moreover, it will also serve to validate and grade their work. |

| Assessment | | | |
|---------------------|------------------------|---|---------------|
| Methodologies | Competencies / Results | Description | Qualification |
| Laboratory practice | A31 A32 B1 | Labs: Grading will take into account both attending the sessions and fulfilling the tasks. It must be remarked that the labs are fundamental for accomplishing the objectives of the guided projects. | 30 |
| Supervised projects | A31 B1 B3 C7 | Guided projects: The quality of the obtained results will chiefly define the mark. However, participating in the discussions about the different projects will be also assessed. | 20 |
| Objective test | B1 B3 | Test: At the end of the course, a written test will be evaluated the level of knowledge on the contents of the subject. | 50 |

| Assessment comments |
|---|
| Those part time students that are exempt of attending lectures, must still produce the results of the labs in one week after the session in which the lab was proposed. These students must also find time and ways of collaborating with their team partners in order to develop the supervised project. |

| Sources of information |
|------------------------|
| |



| | |
|----------------------|--|
| Basic | <ul style="list-style-type: none">- Patrick R. Schaumont (2010). A Practical Introduction to Hardware/Software Codesign. Springer- David C. Black e Jack Donovan (2004). SystemC: From the ground up . Kluwer Academic Publishers- Peter J. Ashenden e Jim Lewis (2008). The Designer's Guide to VHDL, Third Edition (Systems on Silicon). Morgan Kaufmann |
| Complementary | <ul style="list-style-type: none">- Jayaram Bhasker (1999). A VHDL Primer . Prentice Hall- Wayne Wolf (). Computers as Components, 2nd edition. Principles of Embedded Computing System Design. Morgan Kaufmann |

Recommendations

Subjects that it is recommended to have taken before

Fundamentals of Computers/614G01007

Computer Structure/614G01012

Concurrency and Parallelism/614G01018

Subjects that are recommended to be taken simultaneously

Hardware Devices and Interfaces/614G01032

Subjects that continue the syllabus

Embedded Systems/614G01060

Other comments

(*)The teaching guide is the document in which the URV publishes the information about all its courses. It is a public document and cannot be modified. Only in exceptional cases can it be revised by the competent agent or duly revised so that it is in line with current legislation.