
DEGREE CURRICULUM

# COMPUTER ARCHITECTURE

Coordination: ROIG MATEU, CONCEPCION

Academic year 2022-23

# COMPUTER ARCHITECTURE 2022-23

## Subject's general information

<table>
<tr><td>Subject name</td><td>COMPUTER ARCHITECTURE</td></tr>
<tr><td>Code</td><td>102014</td></tr>
<tr><td>Semester</td><td>2nd Q(SEMESTER) CONTINUED EVALUATION</td></tr>
<tr><td>Typology</td><td>
<table>
<tr><th>Degree</th><th>Course</th><th>Character</th><th>Modality</th></tr>
<tr><td>Bachelor's Degree in Computer Engineering</td><td>2</td><td>COMPULSORY</td><td>Attendance-based</td></tr>
<tr><td>Double bachelor's degree: Degree in Computer Engineering and Degree in Business Administration and Management</td><td>2</td><td>COMPULSORY</td><td>Attendance-based</td></tr>
<tr><td>Master's Degree in Informatics Engineering</td><td></td><td>COMPLEMENTARY TRAINING</td><td>Attendance-based</td></tr>
</table>
</td></tr>
<tr><td>Course number of credits (ECTS)</td><td>6</td></tr>
<tr><td>Type of activity, credits, and groups</td><td>
<table>
<tr><th>Activity type</th><th>PRALAB</th><th>TEORIA</th></tr>
<tr><td>Number of credits</td><td>3</td><td>3</td></tr>
<tr><td>Number of groups</td><td>3</td><td>2</td></tr>
</table>
</td></tr>
<tr><td>Coordination</td><td>ROIG MATEU, CONCEPCION</td></tr>
<tr><td>Department</td><td>COMPUTER SCIENCE AND INDUSTRIAL ENGINEERING</td></tr>
<tr><td>Teaching load distribution between lectures and independent student work</td><td>Globally, the subject has 60 hours of presential classes and 120 hours of independent student work.</td></tr>
<tr><td>Important information on data processing</td><td>Consult this link for more information.</td></tr>
<tr><td>Language</td><td>Catalan</td></tr>
</table>

# COMPUTER ARCHITECTURE 2022-23

| Teaching staff | E-mail addresses | Credits taught by teacher | Office and hour of attention |
|---|---|---|---|
| BIGORDA SOLDEVILA, TOMAS | tomas.bigorda@udl.cat | 6 | |
| ROIG MATEU, CONCEPCION | concepcio.roig@udl.cat | 6 | |
| TOMAS CUÑAT, ROSA ANA | rosana.tomas@udl.cat | 3 | |

## Subject's extra information

This subject is held during the second semester of the second course of the degree.

This is a compulsary subject.

To follow up the subject it is required to have the knowledge of functional units composing the computer system, that are studied in the previous subjects of Computer Organization I and II.

## Learning objectives

- Studying the global operation and the levels of the memory hierarchy in the computer.
- Learning the organization of the information in the memory system in order to have efficiency in the acces.
- Learning the pipeline mechanism to execute instructions inside the processor and evaluate its performance.
- Study the processes and the algorithms that are needed to carry out basic and complex arithmetic operations inside the arithmetic unit.
- Analyzing different solutions in terms of efficiency and cost. Being able to find which design solutions provide the best tradeoff between cost and performance.

## Competences

Degree-specific competences

- GII-FB5: Knowledge of the structure, organization, workings and inter-connexion of computer systems, the basis of their programming, and their applications in the resolution of engineering problems.

- GII-CRI9: Ability to know, understand and evaluate computer structures and architecture, as well as the basic components which constitute them.

Degree-transversal competences

- EPS5: Ability for abstraction and critical, logical and logical reasoning.

## Subject contents

**A. THEORETICAL CONTENTS**

# COMPUTER ARCHITECTURE 2022-23

**1. Memory hierarchy**

1.1. Introduction

- General concepts
- Principle of locality

1.2. *Cache memory*

- Cache memory configurations
- Mapping and identification of blocs
- Bloc replacing algorithms
- Writing policies
- Data consistency
- Cache performance

1.3. Main memory

- Organization for improving performace
- Alternative technologies

1.4. Virtual memory

- Elements of virtual memory
- Page table
- TLB (Transaction Look-aside Buffer).

**2. Pipeline processing**

2.1. Basic concepts

2.2. Hazard management

- Estructural hazards
- Data hazards
- Control hazards

2.3. Influency of instruction set

2.4. Superescalar execution

**3. Arithmetic processing**

3.1. Adder circuits.

- Half-adder, full-adder, parallel adder.
- Carry-look-ahead.
- adder/substractor circuit.

3.2. Binary multiplication algorithms.

3.3. Binary division algorithms

3.4. Floating point arithmetic

Floating point format

Approximate representation: rank and precision

Add and substract operations

Multiplication and division operations

**B. PRACTICAL CONTENTS**

Simulation of the functional units of the system.

- Practice of memory hierarchy with SMPcaché simulator. (Theme 1)

- Practice of pipeline processing with WinMIPS64 simulator. (Theme 2)

## Methodology

Classes are taught in face-to-face mode and are divided in different groups: theoretical group (Teo group) and problems/practices group (PraLab group).

The contents of the different kind of groups are divided in the following way:

- Theoretical group: They are expositive lessons where they are shown the main contents on the subject.

- PraLab group: they are lessons to solve exercices and practices, related to the contents exposed in the Teo group classes, in a participative and interactive way.

## Development plan

| Week | description | Activity Grup Teoria | Activity Grup PraLab |
|---|---|---|---|
| 1 | Memory hierarchy | Presentation of the subject.<br>General concepts. Principle of locality. | Exercises of memory hierarchy |
| 2 | Memory hierarchy | Cache memory configurations.<br>Mapping and identification of blocs | Exercises of memory hierarchy |
| 3 | Memory hierarchy | Bloc replacing algorithms. Writing policies | Exercises of memory hierarchy |
| 4 | Memory hierarchy | Data consistency | Exercises of memory hierarchy |
| 5 | Memory hierarchy | Cache performance | Practices of memory hierarchy |
| 6 | Memory hierarchy | Main memory. Virtual memory. | Practices of memory hierarchy |
| 7 | Pipeline processing | Basic concepts | Exercises of pipeline processing |
| 8 | Pipeline processing | Hazard management | Exercises of pipeline processing |
| 9 | Partial evaluation activities. | Realization of first partial exam | |
| 10 | Pipeline processing | Hazard management | Practices of pipeline processing |
| 11 | Pipeline processing | Influence of instruction set.<br>Superscalar execution | Practices of pipeline processing |
| 12 | Arithmetic processing | Adder circuits | Exercises of arithmetic processing. |
| 13 | Arithmetic processing | Binary multiplication algorithms | Exercises of arithmetic processing. |

| | | | |
|---|---|---|---|
| 14 | Arithmetic processing | Binary division algorithms | Exercises of arithmetic processing. |
| 15 | Arithmetic processing | Floating point arithmetic | Exercises of arithmetic processing. |
| 16 i 17 | Partial evaluation activities | Realization of second partial exam | |
| 18 | Seminars | | |
| 19 | Recuperation evaluation activities | Realization of the recuperation exam, if needed. | |

## Evaluation

| Acr. | Evaluation activity | Weighing | Minimum mark | In group | Compulsory | Recoverable |
|---|---|---|---|---|---|---|
| *P1* | *Exam 1$^{er}$ Partial* | 30% | NO | NO | NO | YES |
| *P2* | *Exam 2$^{on}$ Partial* | 50% | NO | NO | NO | YES |
| *PRA* | *Practices* | 20% | NO | YES (<=2) | NO | NO |
| **FINAL_MARK**=30% P1+50% P2+20% PRA<br>To pass the subject, it is necessary that FINAL_MARK is greater than or equal to 5. | | | | | | |
| In the case of not passing the subject there is the option of recuperating it separately for each of the two parts. In this case, the mark is calculated as following:<br>N_rec_P1: recuperation mark of first partial exam (or P1 if not presented to the recuperation exam)<br>N_rec_P2: recuperation mark of second partial exam (or P2 if not presented to the recuperation exam)<br>**FINAL_MARK** = 30% N_rec_P1 + 50% N_rec_P2 + 20% PRA | | | | | | |

## Bibliography

Stallings W., *Organización y arquitectura de computadores*. (7 edición) Prentice-Hall.

Hamacher C., Vranesic Z., Zaky S. *Organización de computadores* (5ªedición). McGraw-Hill.

Ortega J., Anguita M., Prieto A. *Arquitectura de computadores.* Thomson.

Hennessy J. L., Patterson D. A. *Computer Architecture. A Quantitative Approach*. Morgan Kaufmann.