VNIVERSITAT DE VALÈNCIA

Course Guide
34658 Computer Organization

COURSE DATA

Data Subject	
Code	34658
Name	Computer Organization
Cycle	Grade
ECTS Credits	6.0
Academic year	2020 - 2021

Study (s)

Degree	Center	Acad. year	Period
1400 - Degree in Computer Engineering	School of Engineering	2	Second term

Subject-matter

Degree	Subject-matter	Character
1400 - Degree in Computer Engineering	6 - Computer engineering	Obligatory

Coordination

Name	Department
ARNAU LLOMBART, VICENTE	240 - Computer Science
ORTIGOSA ARAQUE, NURIA	240 - Computer Science

SUMMARY

The course "Computer Organization" is a compulsory subject in the second degree course in Computer Engineering. It is part of the subject "Computer Engineering" and is assigned a dedication of 6 ECTS taught in the second semester of the second course.

This course aims to introduce students the basics architecture of current processors and multiprocessor systems from the point of view of process design and system memory, as the programmer. Similarly, the student must be able to evaluate the performance of a monoprocessor or multiprocessor system and propose improvements in both, system architecture and in the code executed. For this simultaneous level description of register transfer level description with the memory and processor-level description of the operating system.

The course starts from the unicycle, multicycle and pipelined processor revised previous courses and introduces the segmentation from a generic point of view as a first approximation to take advantage of instruction level parallelism of the code (ILP). Subsequently, the MIPS processor is particularized. Constraints arise together with segmentation, in which cases it raises the detention of the pipeline (risks). The types of risk are classified and strategies to combat them are suggested. Finally, the interruptions

problem is afforded from the segmentation point of view.

Afterwards, we introduce the superscalar processors as a way to overcome the limitation of one instruction per cycle. We propose a generic architecture that shows the advantages of this approach to maximize the ILP. Similarly are the problems that come with such architectures (new types of risk), proposing strategies to resolve them: Tomasulo algorithm + buffer renaming, reordering buffer and branch predictors. Finally it is introduced the support hardware required to support execution of multiple threads in a processor, evaluating their strengths, weaknesses and strategies for peak performance.

VLIW processors are introduced as a paradigm of how obtain performance with a low-cost approach. This topic introduces software planning techniques basic block (unrolling loops and software segmentation) along with other extended block techniques planning. Finally predicated instructions are introduced as an alternative way to reduce the conditional jump instructions.

Analyze the performance of a parallel system is not as easy as on a monoprocessor system, where execution speed of programs is the main measure. We will have to introduce concepts such as productivity measurement. And we also see that not all systems can be evaluated in the same way, the final goal of a parallel application is not always finish early. The accuracy of the result and the ability to analyze complex problems can be other factors to take into account the performance of these systems.

The main part of a multiprocessor is designing the memory system. We must ensure that access to each of the memory modules are made in a coherent and consistent way. All processors should see the memory so that when a certain direction is read, always it is obtained the latest written value.
We will discuss the protocols used in current multiprocessor memory system design, both based on surveys such as those based directory.

Another key part of getting on a multiprocessor system a parallel application, consisting of multiple processes running in parallel; to work properly are the synchronization events. These allow processes to communicate with each other to exchange values and also can be synchronized to properly execute parallel algorithms that execute on multiprocessors.

The laboratory classes focus on maximizing the performance code execution in both monoprocessor and multiprocessor systems. This should have a thorough knowledge of computer system architecture to maximize its features.

PREVIOUS KNOWLEDGE

Relationship to other subjects of the same degree

There are no specified enrollment restrictions with other subjects of the curriculum.

Other requirements

For this second year subject it is recommended to have passed the subject of Computer Structure and to know concepts introduced in the field of Informatics. It is also advisable to have certain ability in programming, both in high-level languages and assembler.

OUTCOMES

1400 - Degree in Computer Engineering

- G1 - Ability to design, write, organise, plan, develop and sign projects in the field of computer engineering aimed at the design, development or exploitation of computer systems, services and applications.
- G4 - Ability to define, evaluate and select hardware and software platforms for the development and implementation of computer systems, services and applications, in accordance with both the knowledge and the specific skills acquired in the degree.
- G6 - Ability to design and develop computer systems and centralised or distributed computer architectures which integrate hardware, software and networks, in accordance with both the knowledge and the specific skills acquired in the degree.
- G8 - Knowledge of basic subject areas and technologies that serve as a basis for learning and developing new methods and technologies, and of those which provide versatility to adapt to new situations.
- G10 - Knowledge to perform measurements, calculations, assessments, appraisals, surveys, studies, reports, scheduling and other similar work in the field of computer engineering, in accordance with both the knowledge and the specific skills acquired in the degree.
- R1 - Ability to design, develop, select and evaluate computer applications and systems while ensuring their reliability, safety and quality, according to ethical principles and current legislation and regulations.
- R6 - Knowledge and application of basic algorithmic procedures of computer technology to design solutions to problems, by analysing the suitability and complexity of the algorithms proposed.
- R7 - Knowledge, design and efficient use of the types and structures of data most suitable for solving a problem.
- R9 - Ability to know, understand and evaluate the structure and architecture of computers, and also the basic components that comprise them.
- R14 - Knowledge and application of the fundamental principles and basic techniques of parallel, concurrent, distributed and real-time programming.
- IC2 - Ability to analyse, evaluate and select the most appropriate hardware and software platforms to support embedded and real-time applications.

LEARNING OUTCOMES

1 To assess the different instruction formats depending on the number of addresses and length, fixed or variable, their formats.
5 To have a certain ability to take into account the memory hierarchy to reduce the effective memory latency.
6 To assess the role of cache memory and virtual memory system.
7 To use appropriate interrupts and I/O.
12 To understand the basics of instruction level parallelism (ILP) using segmentation and risks that may

arise.
13 To design simple programs those take advantage of processors segmentation.
14 To characterize the costs and benefits of different options for improvements in the processor.
15 To evaluate the performance of a computer in terms of its components (processor, memory, I / S, etc.)..
16 To describe superscalar architectures and their advantages.
17 To evaluate the concept of branch prediction and its utility.
18 To rate speculative execution and identify conditions those justify it.
19 To assess the benefits of multithreading and the factors that limit its performance.
20 Capacity to calculate the relevance of scalability benefits of a system.
21 To assess the benefits of parallel processing, establishing metrics for comparison.
22 To assess the impact on the performance of the interconnection network of a parallel system according to their different designs.
23 Effective use of multiprocessors in response to the memory organization.
24 To assess the advantages and disadvantages of different parallel architectures.
25 To design simple programs on a system with multiple processing elements.
26 To work in a team to make designs and configurations necessary, distributing the workload to deal with complex problems.

To complement the above results, this subject also to acquire the following skills and social skills:

• Understand the operation of segmentation in general and in particular segmented processors and analyze their performance.
• Design a simple segmented pipeline.
• Optimize code that will be executed in a pipelined processor.
• Design the control unit in nonlinear segmented systems.
• Understand the performance of superscalar processors and analyze its performance
• Analyze the performance of various branch predictors.
• Understand VLIW architectures.
• Use the techniques of planning software in VLIW processors and in general in segmented machines.
• Analyze the performance of a given parallel application which is the objectives of the calculation: Get the fastest performance, or achieve greater accuracy in the solution obtained in a given time, or get a problem addressed with greater complexity.
• Understand the problem of sharing the memory of a multiprocessor system and the issues involved concerning the coherence of caches.
• Understand the operation of survey protocols and analyze the functioning of the most used in current multiprocessors.
• Analyze the differences in directory-based protocols based on the survey and study its operation and use.
• Understand the various models of memory consistency.
• Analyze how events are designed for synchronization on multiprocessor systems.
• Study the implementation of locks and barriers as key elements in the synchronization process.

In addition to the specific objectives mentioned above, the course will encourage the development of several generic skills, among which include:

• Capacity for analysis and synthesis.
• Ability to argue from rational and logical criteria.
• Ability to communicate properly and in a organized way.

• Ability to personal work.
• Ability to work in groups.
• Positive evaluation of self-help culture as a means of acquiring knowledge.
• Understand that knowledge acquisition is to ensure professional competence for the future.

DESCRIPTION OF CONTENTS

1. Pipelining

Instruction level parallelism (ILP) and pipelining. Concept.
Ideal and real performance.
DLX pipeline.
Data, structural and control risks.
Pipelined processors interruptions.
Programming pipelined processors.

Theory: 6. Problems: 2. Laboratory 5. Non-contact hours: 8 +5 (Theory + Lab).

2. Superscalar Processors

Superscalar architecture.
Buffer renaming.
The Tomasulo algorithm.
Maintaining consistency: ROB. Interruptions.
Branch prediction.
Multithreading.

Theory: 6. Problems: 3. Laboratory 5. Non-contact hours: 7 +5.

3. VLIW processors

VLIW architectures.
Software segmentation, loop unrolling.
Traces planning.
Predicated instructions.

Theory: 3. Problems: 2. Lab: 6. Non-contact hours: 6 +6.

4. Parallel systems performance

Measuring and reporting performance.
Performance models.

Theory: 5. Problems: 1. Laboratory 1.5. Non-contact hours: 4 +1.5.

5. Coherence and consistency in multiprocessors

Coherence caches.
Protocols based on survey and directory.
Models of consistency and synchronization.

Theory: 6. Problems: 2. Lab 7.5. Non-contact hours: 9 +7.5.

WORKLOAD

ACTIVITY	Hours	% To be attended
Theory classes	30,00	100
Laboratory practices	20,00	100
Classroom practices	10,00	100
Development of individual work	12,00	0
Study and independent work	20,00	0
Preparation of evaluation activities	24,00	0
Preparing lectures	7,00	0
Preparation of practical classes and problem	27,00	0
TOTAL	**150,00**	

TEACHING METHODOLOGY

Theoretical activities.
Description: The lectures will develop the items by providing a global and inclusive vision, analyzing in detail the key issues and more complex, encouraging at all times, participation of students.
Workload for students on the total load of matter: 19%

Practical activities.
Description: Complementing theoretical activities in order to apply the basics and expand the knowledge and experience to be acquired in the course of the proposed work. They include the following types of classroom activities:
-classes of problems and issues in the classroom
-discussion sessions and problem-solving exercises and the students have previously worked
-Labs
- oral presentations
-tutorials scheduled (individualized or group) Conducting individual evaluation questionnaires in the classroom with the presence of teachers.

Workload for students on the total charge of the matter: 21%

Personal work.
Description: Realization (outside the classroom) of monographs, literature search directed issues and problems as well as the preparation of classes and exams (study). This is done individually and tries to promote self-employment.
Workload for students on the total charge of the matter: 45%

Working in small groups.
Description: Realization, by small groups of students (2-4) of work, issues, problems outside the classroom. This work complements the work and encourages individual ability to integrate into working groups.
Workload for students on the total charge of the matter: 15%

It will be used the platform of e-learning (*Aula virtual*) of the University of Valencia in support of communication with students. Through it you will have access to course materials used in class as well as solved problems and exercises.

EVALUATION

Continuous assessment based on participation and the degree of involvement in the teaching-learning process, taking into account regular attendance provided onsite, activities and resolution of issues and problems presented in class. Part of the problem assessment or work done in the tutorial hours individually, making it compulsory to attend tutorials to be evaluated. This part will count 50% of the final grade of the first call.

Objective individual exam, consisting of 1 test of 2 hours, conducted in class time, consisting of both theoretical and practical issues as problems. This part will count 30% of the final grade.

Assessment of practical activities based on the achievement of objectives in the laboratory sessions. Each laboratory session it will be given a questions bulletin that have a dual purpose, one hand guiding the work to be done and secondly to assess how the objectives proposed have been achieved in each laboratory session. This part will count 20% of the final grade.

In the second round, students must take the exam the second call. The exam will consist of a series of theoretical questions in which students must demonstrate knowledge of concepts and relationships, as another set of questions that will assess the practical part of the subject in which the student must prove his fitness and ability to relate the knowledge acquired in the course to the analysis of problems and case studies in the matter field.

In any case, the evaluation of this subject will be done in compliance with the University Regulations in this regard, approved by the Governing Council on 30th May 2017 (ACGUV 108/2017)

REFERENCES

Basic

- Ortega J.; Anguita M.; Prieto A, Arquitectura de Computadores. Ed. Thomson, 2005.
- Hennessy J. L.; Patterson D. A., Computer Architecture a Quantitative Approach. 4ª Edition. Morgan Kaufmann Publishers, 2012. http://links.uv.es/IFDrw2x

Additional

- David E. Culler; Jaswinder Pal Singh., Parallel Computer Architecture: a Hardware/Software Approach. Ed. Morgan Kaufmann Publishers. 1999.
- Sima, D.; Fountain, T.; Kacsur, P, Advanced Computer Architecture. Addison-Wesley, 1998
- K. Hwang. Advanced Computer Architecture. Parallelism, Scalability, Programmability, McGraw Hill, 1993.

ADDENDUM COVID-19

This addendum will only be activated if the health situation requires so and with the prior agreement of the Governing Council

The teaching methodology for this subject will follow the model approved by the Academic Committee of the GII / GIM degrees (https://links.uv.es/catinfmult/modeloDocent). If the facilities are closed because of COVID-19 pandemics, the scheduled lectures will be replaced by synchronous online sessions within the assigned time slots of the course, using the tools provided by the university.

If the facilities need to be closed due to the pandemics causing any of the evaluation exercises to be held at ETSE-UV, these exercises will be substituted by equivalent exercieses held online using the tools provided by the university. The weights for each activity will remain the same as specified in the teaching guide.