

Ficha de Unidade Curricular – (Versão A3ES 2018-2023)

1. Caracterização da Unidade Curricular.

- 1.1. **Designação da unidade curricular (1.000 carateres).**
Sistemas Embebidos / Embedded Systems
- 1.2. **Sigla da área científica em que se insere (100 carateres).**
IC
- 1.3. **Duração¹ (100 carateres).**
Semestral
- 1.4. **Horas de trabalho² (100 carateres).**
162 h
- 1.5. **Horas de contacto³ (100 carateres).**
Total – 67,5 h
T – 22,5 h
TP – 15 h
PL – 30 h
- 1.6. **ECTS (100 carateres).**
6
- 1.7. **Observações⁴ (1.000 carateres).**
Optativa. Comum com outros cursos.
- 1.7. **Remarks (1.000 carateres).**
Elective. Common with other courses.

2. Docente responsável e respetiva carga letiva na Unidade Curricular (preencher o nome completo) (1.000 carateres).

Pedro Miguel Fernandes Sampaio - 135 h

3. Outros docentes e respetivas cargas letivas na unidade curricular (1.000 carateres).

Não aplicável.

4. Objetivos de aprendizagem (conhecimentos, aptidões e competências a desenvolver pelos estudantes). (1.000 carateres).

Os estudantes que terminam com sucesso esta unidade curricular serão capazes de:

- (1) Desenhar estruturas de hardware baseadas em microcontroladores.
- (2) Desenvolver software de sistema de interface com o hardware.
- (3) Desenvolver software de aplicação para sistemas embebidos.
- (4) Depurar o hardware e o software realizado.

4. Intended learning outcomes (knowledge, skills and competences to be developed by the students). (1.000 characters).

Students who successfully complete this curricular unit will be able to:

- (1) Design hardware structures based on microcontrollers.
- (2) Develop system software interface hardware.
- (3) Develop application software for embedded systems.
- (4) Debug the hardware and software.

5. Conteúdos programáticos (1.000 carateres).

- I. Introdução aos sistemas embebidos: paradigma, arquiteturas típicas, aplicações e limitações;
- II. Estudo da arquitetura Cortex-M3: conjunto de instruções, modelo de programação e interrupções (vetorização);
- III. Desenvolvimento de software para sistemas embebidos: paradigma, ferramentas, ambiente de

desenvolvimento (incluindo módulo de arranque e ficheiros de configuração do linker) e métodos de validação e de teste.

IV. Protocolos série síncronos: SPI, I2C.

V. Organização de software para sistemas embebidos: modelo de camadas, modelo de máquina de estados e construção e utilização de bibliotecas estáticas.

VI. Desenvolvimento de aplicações para sistemas embebidos sem sistema operativo (bare metal): interação com dispositivos periféricos por pesquisa de estado e interrupção, produção automática da documentação.

5. Syllabus (1.000 characters).

I. Introduction to embedded systems: paradigm, standard architectures, applications and limitations;

II. Study of the Cortex-M3 architecture: instruction set architecture, programming model and interrupt mechanism (vectorization);

III. Development of software for embedded systems: paradigm, tools, development environment (including boot and configuration files of the linker) and methods for test and debug;

IV. Synchronous serial protocols: SPI, I2C;

V. Organization of software for embedded systems: model of layers, state machine model and development and utilization of static libraries;

VI. Development of applications for embedded systems without operating system (bare metal): interface with peripherals by pooling and interrupt, automatic delivering of documentation.

6. Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular (1.000 caracteres).

Quando iniciam esta UC, os alunos sabem programar em linguagem C, sabem eletrónica digital e conhecem o princípio de funcionamento de um processador.

O estudo de dispositivos e interfaces hardware (conteúdos I, II, III e IV) permite atingir o objetivo 1 conceber sistemas baseados em microcontroladores.

O estudo das interfaces programáticas e gestão de dispositivos periféricos (ponto IV) permite atingir o objetivo 2.

A estruturação do software de interação com dispositivos (ponto V) e o estudo do modelo de máquina de estados (ponto VI) permite atingir o objetivo 3.

O domínio das ferramentas de desenvolvimento e da representação dos programas nas várias etapas da sua produção permitem atingir o objetivo 4.

6. Evidence of the syllabus coherence with the curricular unit's intended learning outcomes (1.000 characters).

When they start this curricular unit, students know how to program in C language, digital electronics and the principle of a how processor works.

The study of hardware devices and interfaces (contents I, II, III and IV) achieves goal 1 to design systems based on microcontrollers.

The study of programmatic interfaces and management of peripheral devices (point IV) allows to reach objective 2.

The structuring of the device interaction software (point V) and the study of the state machine model (point VI) allows us to reach goal 3.

The mastery of the development tools and the representation of the programs in the various stages of their production allow us to reach goal 4.

7. Metodologias de ensino (avaliação incluída) (1.000 caracteres).

Nas aulas teóricas os alunos são incentivados a participar colocando questões ou resolvendo os problemas propostos durante as aulas teórico-práticas. Na parte prática são utilizados módulos que permitem a montagem de sistemas de prototipagem. As práticas são focadas no estudo de elementos isolados (periféricos, metodologias de programação, protocolos, subsistemas) que conduzem à execução do projeto apresentado no início da UC. Os objetivos 1, 2 e 3 são avaliados em teste escrito individual. O objetivo 4 é avaliado durante as aulas práticas. Todos os objetivos são avaliados em discussão final, com base nos relatórios das atividades prática, com atribuição de nota individual. A classificação final resulta de uma média aritmética ponderada das três componentes de avaliação, em que o teste escrito individual tem um peso de 40%, o trabalho de projeto tem um peso de 40% e a discussão final tem um peso de 20%. A classificação mínima exigida no teste escrito individual é de 10 valores.

7. Teaching methodologies (including assessment) (1.000 characters).

In the theoretical classes the students are encouraged to participate by asking questions or solving the problems proposed during the theoretical-practical classes. Practices are focused on the study of isolated elements (peripherals, programming methodologies, protocols, subsystems) that lead to the execution of the project presented at the beginning of the semester. Goal 1, 2 and 3 are evaluated in individual written test. Goal 4 is assessed during practice classes. All the objectives are evaluated in the final discussion, based on the reports of the practical activities, with individual assignment. The final classification results from a weighted arithmetic mean of the three evaluation components, in which the individual written test has a weight of 40%, the project work has a weight of 40% and the final discussion has a weight of 20%. The minimum grade required in the individual written test is 10.

8. Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular (3.000 caracteres).

As aulas teórica e teórico-práticas são compostas pela exposição e discussão dos conceitos teóricos, seguidas da aplicação desses conceitos na resolução de problemas concretos.

Um número significativo das soluções dos problemas resolvidos nas aulas teórico-práticas são concretizadas ou concretizáveis, sobre os dispositivos montados nos módulos eletrônicos utilizados em laboratório.

As atividades práticas consistem na montagem dos módulos e na realização de programas experimentais, baseados em software realizado nas aulas teórico-práticas e em software concebido pelos alunos. A concepção das montagens e a elaboração dos respectivos programas, contribuem para os objetivos 1, 2 e 3. A realização experimental das montagens e depuração dos respectivos programas permitem atingir o objetivo 4.

O projeto consiste na realização de um sistema completo e autônomo dada uma especificada em linguagem natural.

8. Evidence of the teaching methodologies coherence with the curricular unit's intended learning outcomes (3.000 characters).

The theoretical-practical classes are composed by the exposition and discussion of the theoretical concepts, followed by the application of these concepts in the resolution of concrete problems.

A significant number of the solutions of the problems solved in the theoretical-practical classes are tested on the devices mounted on the electronic modules used in the laboratory.

Practical activities consist of the assembly of the modules and the realization of experimental programs, based on software carried out in the theoretical-practical classes and in software designed by the students. The design of the assemblies and the elaboration of the respective programs contribute to objectives 1, 2 and 3. The experimental realization of the assemblies and purification of the respective programs allow to reach objective 4.

The project consists in the realization of a complete and autonomous system given one specified in natural language.

9. Bibliografia de consulta/existência obrigatória (1.000 caracteres).

D. Lacamera, *Embedded Systems Architecture: Explore architectural concepts, pragmatic design patterns, and best practices to produce robust systems*, Packt Publishing, 2018. ISBN 978-1788832502

M. Wolf, *Computers as Components*, 3rd edition, Morgan Kaufman, 2012. ISBN 9780123884367

M. Barr, A. Massa, *Programming Embedded Systems: With C and GNU Development Tools*, 2nd edition, O'Reilly 2006. ISBN 978-0596009830

J. Catsoulis, *Designing Embedded Hardware: Create New Computers and Devices*, 2nd edition, O'Reilly, 2005. ISBN 978-0596007553.

A. Sloss, D. Symes & C. Wright, *ARM System Developer's Guide Designing and Optimizing System Software*, Morgan Kaufmann, 2005. ISBN 1558608745

¹ Anual, semestral, trimestral, ...

² Número total de horas de trabalho.

³ Discriminadas por tipo de metodologia adotado (T - Ensino teórico; TP - Ensino teórico-prático; PL - Ensino prático e laboratorial; TC - Trabalho de campo; S - Seminário; E - Estágio; OT - Orientação tutorial; O - Outro).

⁴ Assinalar sempre que a unidade curricular seja optativa.