



Universidad  
de Alcalá

# GUÍA DOCENTE

## Sistemas en Tiempo Real

Grado en Ingeniería de Computadores

**Universidad de Alcalá**

---

**Curso académico 2019/2020**  
**3º Curso – 2º Cuatrimestre**

## GUÍA DOCENTE

Nombre de la asignatura:	<b>Sistemas en Tiempo Real</b>
Código:	<b>590008</b>
Titulación en la que se imparte:	<b>Grado en Ingeniería de Computadores</b>
Departamento y Área de Conocimiento:	<b>Dpto.: Automática / Ingeniería de Sistemas y Automática</b>
Carácter:	<b>Obligatoria de tecnología específica</b>
Créditos ECTS:	<b>6</b>
Curso y cuatrimestre:	<b>Tercer Curso, Segundo cuatrimestre</b>
Profesorado:	<b>Ignacio Parra Alonso</b>
Horario de Tutoría:	<b>M 15-17h J 10-12 y 15-17h</b>
Idioma en el que se imparte:	<b>Español/English Friendly</b>

### 1a. PRESENTACIÓN

El incremento en fiabilidad y velocidad de proceso de los ordenadores, junto con una disminución en su tamaño y precio, ha hecho que los encontremos en todos los ámbitos de nuestra vida, desde un teléfono móvil hasta el control del tráfico aéreo. Una de las áreas de expansión más rápida de la explotación de ordenadores es la relacionada con los sistemas empotrados y de tiempo real. Estas áreas necesitan procesar información con el objetivo de controlar procesos. Se ha estimado que más del 90 por ciento de la producción mundial de microprocesadores se utiliza en este tipo de sistemas. Estas aplicaciones plantean requisitos específicos para los lenguajes de programación necesarios para programarlos, ya que tienen características diferentes de las aplicaciones de procesamiento de información tradicionales.

Esta asignatura estudia los sistemas de tiempo real y su implementación. Plantea la forma de modelar sistemas de tiempo real que tengan restricciones estrictas de temporización y fiabilidad, así como las posibilidades que nos ofrecen los lenguajes de programación para implementar este tipo de sistemas.

## 1b. PRESENTATION

Increase in speed and reliability of personal computers, along with size and price reduction, has put them on almost every area of our daylife, from mobile phones to air traffic control. One of the areas with the highest expansion is embedded and real time systems. These kind of systems process information to control processes. Real time systems account for about 90% of global microprocessors production. The specific characteristics of these kind of systems require the use of programming languages specifically designed to this end.

This course studies real time systems and their implementation aspects. It poses the problem of modeling this kind of systems with strong limitations on timing and reliability. In addition, this course studies the possibilities of different languages oriented to these kind of systems.

## 2. COMPETENCIAS

### Competencias generales:

CG3 Capacidad para diseñar, desarrollar, evaluar y asegurar la accesibilidad, ergonomía, usabilidad y seguridad de los sistemas, servicios y aplicaciones informáticas, así como de la información que gestionan.

CG4 Capacidad para definir, evaluar y seleccionar plataformas hardware y software para el desarrollo y la ejecución de sistemas, servicios y aplicaciones informáticas, de acuerdo con los conocimientos adquiridos según lo establecido en el apartado 5 de la resolución BOE-A-2009-12977.

CG5 Capacidad para concebir, desarrollar y mantener sistemas, servicios y aplicaciones informáticas empleando los métodos de la ingeniería del software como instrumento para el aseguramiento de su calidad, de acuerdo con los conocimientos adquiridos según lo establecido en el apartado 5 de la resolución BOE-A-2009-12977.

CG9 Capacidad para resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad. Capacidad para saber comunicar y transmitir los conocimientos, habilidades y destrezas de la profesión de Ingeniero Técnico en Informática.

### Competencias específicas:

CIC2 Capacidad de desarrollar procesadores específicos y sistemas empujados, así como desarrollar y optimizar el software de dichos sistemas.

CIC5 Capacidad de analizar, evaluar y seleccionar las plataformas hardware y software más adecuadas para el soporte de aplicaciones empujadas y de tiempo real.

### Resultados de aprendizaje:

- RA1. Comprender cómo se programa un sistema embebido. Comprender la importancia del bajo consumo. Identificar las fuentes de fallo de un sistema y aplicar estrategias para minimizar problemas. CE-ESY4, CE-ESY5.
- RA2. Distinguir sistemas operativos en tiempo real de los sistemas operativos de sobremesa y de servidores. Comprender cómo emplear herramientas para soportar el diseño de sistemas en tiempo real. CE-ESY3, CE-ESY7.
- RA3. Capacidad de utilizar métodos de planificación de STR, tanto básicos como avanzados, para asegurar el cumplimiento de restricciones temporales
- RA4. Capacidad de diseñar e implementar código de un sistema de tiempo real para el control de procesos con recursos compartidos.

### 3. CONTENIDOS

Bloques de contenido (se pueden especificar los temas si se considera necesario)	Total de clases, créditos u horas
Modelado de sistemas de tiempo real	• 10 T+8 P horas
Implementación concurrente de sistemas de tiempo real	• 10 T+10 P horas
Fiabilidad en los sistemas de tiempo real	• 6 T+6 P horas
Metodologías de diseño de sistemas de tiempo real	• 4 T+4 P horas

### Cronograma (Optativo)

Semana / Sesión	Contenido
01 <sup>a</sup>	• Sistemas de control y de tiempo real. Definiciones. Entorno o contexto de un sistema de control. Modelos de sistemas de control. Características de los sistemas de tiempo real. Desarrollo de sistema de tiempo real.
02 <sup>a</sup>	• Introducción a la programación. Elementos de lenguaje. Bloques. Definiciones de tipos y declaración de objetos. Expresiones. Estructuras de control. Subprogramas. Tipos avanzados.
03 <sup>a</sup>	• Programación a gran escala. Modularidad, abstracción y reutilización de código.
04 <sup>a</sup>	• Control de sistemas dinámicos. Planteamiento del problema. Realización de un controlador PID. Implementación de un

	controlador PID genérico.
05 <sup>a</sup>	<ul style="list-style-type: none"> <li>• Media de tiempos y retardos. Dispositivos de entrada/salida. Implementación de un controlador PID concreto.</li> </ul>
06 <sup>a</sup>	<ul style="list-style-type: none"> <li>• Modelado de sistema concurrentes. Redes de Petri. Comportamiento dinámico de una red de Petri. Redes de Petri interpretadas. Subclases de redes de Petri. Traducción de Redes de Petri a programas.</li> </ul>
07 <sup>a</sup>	<ul style="list-style-type: none"> <li>• Sistemas dirigidos por tiempo. Modelado temporal de tareas. Diseño e implementación de un ejecutivo cíclico. Estimación del tiempo de cómputo.</li> </ul>
08 <sup>a</sup>	<ul style="list-style-type: none"> <li>• Sistemas multiprogramados. Modelos de concurrencia. Tipos tarea. Tareas dinámicas. Estados de una tarea. Planificación de tareas independientes.</li> </ul>
09 <sup>a</sup>	<ul style="list-style-type: none"> <li>• Comunicación y sincronización de tareas. Memoria compartida. Semáforos. Objetos protegidos. Planificación de tareas esporádicas. Planificación de tareas dependientes.</li> </ul>
10 <sup>a</sup>	<ul style="list-style-type: none"> <li>• Comunicación mediante mensajes. Semántica del paso de mensajes. El concepto de cita extendida. Llamada condicional. Llamada temporizada.</li> </ul>
11 <sup>a</sup>	<ul style="list-style-type: none"> <li>• Aspectos generales de fiabilidad. Prevención de errores. Tolerancia de fallos. Redundancia. Programación con N versiones. Redundancia dinámica.</li> </ul>
12 <sup>a</sup>	<ul style="list-style-type: none"> <li>• Aspectos concretos de fiabilidad. Fallos y excepciones. Manejadores de excepciones. Propagación y postmanejo de excepciones. Gestión de excepciones. Transferencia asíncrona del control.</li> </ul>
13 <sup>a</sup>	<ul style="list-style-type: none"> <li>• Programación de bajo nivel. Representación de los datos. Gestión de interrupciones. Planificación de manejadores de dispositivos.</li> </ul>
14 <sup>a</sup>	<ul style="list-style-type: none"> <li>• Metodologías de diseño de sistemas de tiempo real. Análisis y especificación de requisitos. Diseño. Implementación. Integración y pruebas.</li> </ul>

## 4. METODOLOGÍAS DE ENSEÑANZA-APRENDIZAJE.-ACTIVIDADES FORMATIVAS

### 4.1. Distribución de créditos (especificar en horas)

Número de horas presenciales:

28 horas teoría + 28 horas laboratorio + 4 horas de examen de evaluación

Número de horas del trabajo propio del estudiante:	90 horas
Total horas	150 horas = 6 ECTS

#### 4.2. Estrategias metodológicas, materiales y recursos didácticos

	Exposición teórica y demostraciones con el ordenador.
	Resolución de problemas en grupo y resolución de prácticas individuales y trabajos por cada alumno.
	Exposiciones de los alumnos.

### 5. EVALUACIÓN: Procedimientos, criterios de evaluación y de calificación

#### *Procedimientos de Evaluación*

##### - Convocatoria ordinaria

+ Evaluación Continua: en la convocatoria ordinaria, todos los alumnos serán evaluados en la modalidad de evaluación continua, que constará de dos pruebas parciales y la evaluación de las prácticas de laboratorio. Los estudiantes que hayan seguido la evaluación continua y no la hayan superado, no podrán acogerse a la evaluación final de la convocatoria ordinaria. Los alumnos que no se presenten a ninguna de las pruebas parciales o que no entreguen dos o más de las prácticas propuestas, serán considerados como No Presentados.

+ Evaluación Final: aquellos alumnos que presenten solicitud por escrito al Director de la Escuela y tengan una causa justificada, podrán ser evaluados mediante evaluación final. Esta evaluación consta de un examen final que incluirá pruebas teóricas y prácticas. El plazo límite de solicitud será de dos semanas desde el comienzo de las clases o desde la matriculación en la asignatura en el caso de que sea posterior.

- Convocatoria Extraordinaria: en la convocatoria extraordinaria, los alumnos que no hayan superado la convocatoria ordinaria realizarán una prueba que incluirá cuestiones teóricas y prácticas.

#### *Criterios de Evaluación*

Los Criterios de Evaluación deben atender al grado de adquisición de las competencias por parte del estudiante.

CE1: El alumno muestra capacidad e iniciativa a la hora de resolver problemas prácticos asociados al diseño de sistemas de tiempo real.

CE2: El alumno puede realizar un diseño de un sistema de tiempo real asegurando el cumplimiento de los requisitos temporales.

CE3: El alumno es capaz de implementar un sistemas de tiempo real, analizando los fallos y asegurando el cumplimiento de las restricciones temporales.

CE4: El alumno ha adquirido los diferentes conceptos relativos al diseño, análisis e implementación de sistemas en tiempo real.

#### *Instrumentos de Evaluación*

- *Prueba de evaluación Intermedia (PEI)*: Consistente en la resolución de problemas prácticos de diseño e implementación de sistemas de tiempo real, así como la demostración del conocimiento de las características de los mismos.
- *Prácticas de laboratorio (PL)*: Consistentes en la resolución de problemas prácticos con herramientas informáticas mediante la programación y modelado de sistemas de tiempo real.
- *Prueba de evaluación Final (PEF)*: Consistente en la resolución de problemas prácticos de diseño e implementación de sistemas de tiempo real, así como la demostración del conocimiento de las características de los mismos.

#### *Criterios de Calificación*

En la evaluación continua de la convocatoria ordinaria la relación entre los criterios, instrumentos y calificación es la siguiente:

Competencias	Resultado de Aprendizaje	Criterio de Evaluación	Instrumento de Evaluación	Peso en la calificación
CG3-5, CG9, CIC2, CIC5	RA2,RA3	CE1,CE2,CE4	PEI1	30%
CG3-5, CG9, CIC2, CIC5	RA1, RA4	CE1,CE3	PL	30%
CG3-5, CG9, CIC2, CIC5	RA2,RA3	CE1,CE2,CE4	PEF	40%

En la evaluación final de la convocatoria ordinaria la relación entre los criterios, instrumentos y calificación es la siguiente:

Competencias	Resultado de Aprendizaje	Criterio de Evaluación	Instrumento de Evaluación	Peso en la calificación
CG3-5, CG9,	RA1-RA4	CE2-CE4	PEF	100%

CIC2, CIC5				
------------	--	--	--	--

En la evaluación final de la convocatoria extraordinaria la relación entre los criterios, instrumentos y calificación es la siguiente:

Competencias	Resultado de Aprendizaje	Criterio de Evaluación	Instrumento de Evaluación	Peso en la calificación
CG3-5, CG9, CIC2, CIC5	RA1-RA4	CE2-CE4	PEF	100%

## 6. BIBLIOGRAFÍA

### Bibliografía Básica:

- A. Burns y A. Wellings. *Sistemas de Tiempo Real y Lenguajes de Programación*. 3ª edición. Pearson Educación. 2003
- ISO/IEC 862:1995(E) - RM95;6.0 *Ada Reference Manual*.

### Bibliografía complementaria:

- T. Murata. *Petri Nets: Properties, Analysis and Applications*. Proceeding of the IEEE **77**(4). 1989.





Universidad  
de Alcalá

# TEACHING GUIDE

## Real Time Systems

**Bachelor's Degree in Computer  
Engineering**

**University of Alcalá**

**Academic year 2019/2020**

**3<sup>rd</sup> Year – 2<sup>nd</sup> Semester**

## TEACHING GUIDE

Subject:	<b>Real Time Systems</b>
Code:	<b>590008</b>
Degree:	<b>Degree in Computer Engineering</b>
Department and knowledge area:	<b>Computer Engineering Department / Automation and Systems Engineering Area</b>
Character:	<b>Obligatory of specific technology</b>
ECTS Credits:	<b>6</b>
Curso y cuatrimestre:	<b>3rd Year – 2nd Semester</b>
Teachers:	<b>Ignacio Parra Alonso</b>
Tutorial sessions:	<b>M 15-17h Th 10-12 y 15-17h</b>
Teaching language:	<b>Spanish/English Friendly</b>

### 1. INTRODUCTION

Increase in speed and reliability of personal computers, along with size and prize reduction, has put them on almost every area of our daylife, from mobile phones to air traffic control. One of the areas with the highest expansion is embeeded and real time systems. These kind of systems process information to control processes. Real time systems account for about 90% of global microprocessors production. The specific characteristics of these kind of systems require the use of programming languages specifically designed to this end.

This course studies real time systems and their implementation aspects. It poses the problem of modeling this kind of systems with strong limitations on timing and reliability. In addition, this course studies the possibilities of different languages oriented to these kind of systems.

## 2. SKILLS

### General skills:

CG3 Ability to design, develop, evaluate and ensure the accessibility, ergonomics, usability and safety systems, services and applications, as well as the information they manage.

CG4 Ability to define, evaluate and select hardware and software platforms for the development and implementation of systems, services and applications, according to the knowledge acquired as provided in paragraph 5 of resolution BOE-A-2009-12977.

CG5 Ability to conceive, develop and maintain systems, services and applications using the methods of software engineering as a tool for quality assurance, according to the knowledge acquired as provided in paragraph 5 of resolution BOE-A- 2009-12977.

CG9 Ability to solve problems with initiative, decision making, autonomy and creativity. Ability to communicate and transmit knowledge and skills of the profession of Technical Engineer.

### Specific skills:

CIC2 Ability to develop processors specific and embedded systems and software development and optimization of such systems.

CIC5 Ability to analyze, evaluate and select the most appropriate hardware platforms and support for embedded real-time applications and software.

### Learning outcomes:

RA1. Understand how to program an embedded system. Understand the importance of low power consumption. Identify failures and applying the strategies to minimize problems. CE-ESY4, CE-ESY5.

RA2. Being able to differentiate between real time operative systems and desktop and server systems. Understand how to use tools to design real time systems. CE-ESY3, CE-ESY7.

RA3. Capacity to use planning methods for real time systems, both basic and advanced to ensure the temporal restrictions.

RA4. Capacity to design and implement code for a real time system applied to a control system with shared resources.

### 3. CONTENTS

Block Content	Hours
Real time systems modelling	• 10 T+8 P h
Concurrent Implementation of real time systems	• 10 T+10 P h
Reliability on real time systems	• 6 T+6 P h
Design methodologies for real time systems	• 4 T+4 P h

### Chronogram

Sesion	content
01 <sup>a</sup>	<ul style="list-style-type: none"> <li>Control and real time systems Definitions. Context of a control system. Models of control systems. Characteristics of real time systems. Development of real time systems.</li> </ul>
02 <sup>a</sup>	<ul style="list-style-type: none"> <li>Introduction to programming. Language elements. Blocks. Type definition and objects declaration. Control estructuras. Subprograms. Advanced types.</li> </ul>
03 <sup>a</sup>	<ul style="list-style-type: none"> <li>Modularity, abstraction and code reuse.</li> </ul>
04 <sup>a</sup>	<ul style="list-style-type: none"> <li>Dynamic systems control. PID controllers.</li> </ul>
05 <sup>a</sup>	<ul style="list-style-type: none"> <li>Time and delay measurement. Input/Output devices.</li> </ul>
06 <sup>a</sup>	<ul style="list-style-type: none"> <li>Concurrent systems. Petri nets. Dynamical behaviour of a Petri net. Interpreted Petri nets. Petri nets subclasses. Petri nets translation to programs.</li> </ul>
07 <sup>a</sup>	<ul style="list-style-type: none"> <li>Temporal modlling. Computation time estimation.</li> </ul>
08 <sup>a</sup>	<ul style="list-style-type: none"> <li>Tasks. Dynamic Tasks.Planning of independent tasks.</li> </ul>
09 <sup>a</sup>	<ul style="list-style-type: none"> <li>Tasks communication and synchronization. Shared memory. Protected objects. Panning of dependent tasks.</li> </ul>
10 <sup>a</sup>	<ul style="list-style-type: none"> <li>Communication through messages. Message pass. Conditional and temporal date.</li> </ul>
11 <sup>a</sup>	<ul style="list-style-type: none"> <li>Reliability. Error prevention. Fails tolerance. Redundancy. Dynamic redundancy.</li> </ul>
12 <sup>a</sup>	<ul style="list-style-type: none"> <li>Fails and exceptions. Exceptions handlers. Propagation and</li> </ul>

	management of exceptions. Asynchronous control transfer.
13 <sup>a</sup>	• Low level programming. Device handlers.
14 <sup>a</sup>	• Methodologies for the design of real time systems.

## 4. TEACHING-LEARNING METHODOLOGIES. FORMATIVE ACTIVITIES

### 4.1. Credit distribution

Number of attendance hours:	Theory 28 hours + Lab 28 hours + 4 hours of assessment
Student work hours:	90 hours
Total hours	150 hours = 6 ECTS

### 4.2. Methodological strategies, teaching materials and resources

Lectures	Master classes with theory and practical examples.
Practical exercises resolution	Master classes and team work on practical exercises resolution. Small groups problem discussion. Oral and written exposition.
Laboratory	Work in up to 2 people groups with the laboratory equipment. Exposition and discussion of practical examples. Common resolution/approach to the problem. Group implementation and defence in a presentation.

## 5. ASSESMENT

### Evaluation procedure

- Ordinary call

+ Continuous evaluation: In the ordinary call, all the students will be evaluated in continuous evaluation. This will consist of two partial tests and the evaluation of the laboratory. Students following continuous evaluation that fail, can not attend to the final evaluation of the ordinary call. Students not attending to any of the partial tests or not delivering two or more of the proposed laboratory practices will be considered as no show.

+ Final evaluation: In the first two weeks from the start of the course or from registration (whichever is last), those students with a justified cause can present a request to be evaluated through final evaluation to the Dean of the School. Upon approval of this request, these students will be evaluated in a final exam that will include theoretical and practical exercises.

- Extraordinary call: Those students that failed on the ordinary call, will be evaluated in a final exam that will include theoretical and practical exercises.

### *Evaluation criteria*

Evaluation criteria should address the acquisition of skills by the student.

CE1: The student shows capacity and initiative to resolve practical problems in the area of control systems design for real time systems.

CE2: The student is able to implement a real time system ensuring the time constraints.

CE3: The student is able to implement a real time system analyzing errors.

CE4: The student has acquired the different concepts about design, analysis and implementation of real time systems.

### *Assessment tools*

- *Intermediate Evaluation Test (IET): Practical exercises resolution about control systems implementation.*
- *Laboratory works (LW): Practical problems resolution in the laboratory.*
- *Final Evaluation Test (FET): Practical exercises resolution about control systems implementation.*

### *Final Marks Criteria*

In the ordinary call the relationship between criteria, instruments and marks will be as follows:

Skills	Learning Outcome	Evaluation Criteria	Assessment tool	Weight in mark
CG3-5, CG9, CIC2, CIC5	RA2,RA3	CE1,CE2,CE4	IET	30%
CG3-5, CG9, CIC2, CIC5	RA1, RA4	CE1,CE3	LW	30%
CG3-5, CG9, CIC2, CIC5	RA2,RA3	CE1,CE2,CE4	FET	40%

In the final evaluation the relationship between criteria, instruments and marks will be as follows:

Skills	Learning Outcome	Evaluation Criteria	Assessment tool	Weight in mark
CG3-5, CG9, CIC2, CIC5	RA1-RA4	CE2-CE4	FET	100%

In the extraordinary call the relationship between criteria, instruments and marks will be as follows:

Skills	Learning Outcome	Evaluation Criteria	Assessment tool	Weight in mark
CG3-5, CG9, CIC2, CIC5	RA1-RA4	CE2-CE4	FET	100%

## 6. BIBLIOGRAFÍA

### Basic references:

- A. Burns y A. Wellings. *Sistemas de Tiempo Real y Lenguajes de Programación*. 3ª edición. Pearson Educación. 2003
- ISO/IEC 862:1995(E) - RM95;6.0 *Ada Reference Manual*.

### Extended References:

- T. Murata. *Petri Nets: Properties, Analysis and Applications*. Proceeding of the IEEE **77**(4). 1989.