

## 30205 - Arquitectura y organización de computadores 1

### Información del Plan Docente

<b>Año académico</b>	2018/19
<b>Asignatura</b>	30205 - Arquitectura y organización de computadores 1
<b>Centro académico</b>	110 - Escuela de Ingeniería y Arquitectura 326 - Escuela Universitaria Politécnica de Teruel
<b>Titulación</b>	439 - Graduado en Ingeniería Informática 443 - Graduado en Ingeniería Informática
<b>Créditos</b>	6.0
<b>Curso</b>	1
<b>Periodo de impartición</b>	Segundo Semestre
<b>Clase de asignatura</b>	Formación básica
<b>Módulo</b>	Materia básica de grado

### **1. Información Básica**

#### **1.1. Objetivos de la asignatura**

La asignatura y sus resultados previstos responden a los siguientes planteamientos y objetivos:

Esta asignatura de introducción busca que cada estudiante sea capaz de comprender una arquitectura de lenguaje máquina y diseñar programas en lenguaje ensamblador capaces de comunicarse con periféricos.

#### **1.2. Contexto y sentido de la asignatura en la titulación**

Esta asignatura forma parte de la materia básica de Arquitectura de Computadores en el Grado de Ingeniería Informática. La asignatura enlaza con Introducción a los Computadores y es requisito para cursar Arquitectura y Organización 2.

#### **1.3. Recomendaciones para cursar la asignatura**

Para cursar esta asignatura es prerequisite haber cursado la asignatura Introducción a los Computadores.

### **2. Competencias y resultados de aprendizaje**

#### **2.1. Competencias**

Al superar la asignatura, el estudiante será más competente para...

Resolver problemas y tomar decisiones con iniciativa, creatividad y razonamiento.

Aplicar las tecnologías de la información y las comunicaciones en ingeniería.

Utilizar conocimientos básicos sobre el uso y programación de los ordenadores, bases de datos y programas informáticos con aplicación en ingeniería.

## 30205 - Arquitectura y organización de computadores 1

Aplicar conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería.

Estas dos últimas competencias se desarrollan de forma específica para reforzar las dos primeras, persiguiendo los resultados de aprendizaje de la asignatura (léase apartado Introducción, resultados de aprendizaje de esta guía).

### 2.2.Resultados de aprendizaje

El estudiante, para superar esta asignatura, deberá demostrar los siguientes resultados...

Conoce a nivel básico los parámetros que definen la arquitectura de lenguaje máquina (repertorio, formato y codificación de instrucciones, almacenes, tipos de datos, modos de direccionamiento, control del secuenciamiento y transferencias de control, gestión de excepciones).

Conoce y puede manejar la arquitectura de lenguaje máquina de un procesador de referencia.

Distingue los conceptos de lenguaje máquina y ensamblador.

Conoce los métodos de representación y codificación de la información y sus operaciones básicas. Es capaz de traducir estructuras de datos y control de lenguajes de alto nivel a ensamblador. Utiliza llamadas a procedimiento.

Entiende el modelo genérico de registros de un controlador de dispositivo periférico y los métodos básicos de sincronización y transferencia. Puede programar cualquier dispositivo de E/S y sabe cómo tratar las excepciones.

Sabe integrar código ensamblador y rutinas de librería en programas escritos en lenguajes de alto nivel.

### 2.3.Importancia de los resultados de aprendizaje

Esta asignatura fundamenta el diseño, programación y uso eficiente del computador, ya sea este de propósito general o específico (sistema empotrado, supercomputación, etc.).

## 3.Evaluación

### 3.1.Tipo de pruebas y su valor sobre la nota final y criterios de evaluación para cada prueba

El estudiante deberá demostrar que ha alcanzado los resultados de aprendizaje previstos mediante las siguientes actividades de evaluación

#### En la Escuela de Ingeniería y Arquitectura del Campus Rio Ebro:

La evaluación de la asignatura se realiza en base a dos pruebas:

**P1.** Prueba escrita (examen) en la que responder cuestiones y resolver ejercicios y problemas. Se requiere una nota mínima de 5.0 puntos en esta prueba para aprobar la asignatura.

Si se supera esta prueba, entonces la prueba pondera un 90% en la nota de la asignatura y, si no se alcanza este mínimo, entonces la calificación en la asignatura es la de esta prueba.

## 30205 - Arquitectura y organización de computadores 1

**P2.** Trabajos y pruebas de laboratorio. Cada estudiante deberá entregar los trabajos que se indiquen en las prácticas de la asignatura. Se requiere una nota mínima de 5.0 puntos en esta prueba para aprobar la asignatura.

Si se supera esta prueba, entonces la prueba pondera un 10% en la nota de la asignatura y, si no se alcanza este mínimo, entonces la calificación en la asignatura es la de esta prueba.

Si se suspenden ambas pruebas la calificación corresponde a realizar promedio como en el caso de aprobado.

**Pruebas con carácter voluntario.** A lo largo del cuatrimestre se podrán plantear una o varias pruebas voluntarias consistentes en la resolución de casos prácticos, ejercicios o problemas. El peso total será un extra de un 10% de la nota, únicamente en el caso de obtener una nota mínima de 5.0 sobre 10.

**Segunda convocatoria.** La evaluación de la asignatura se realiza en base a dos pruebas análogas a las de la primera convocatoria, con las mismas ponderaciones y exigencia de notas mínimas.

### Escuela Universitaria Politécnica de Teruel

Examen escrito dividido en dos partes, una parte de teoría y cuestiones y la otra de programación en ensamblador

## 4. Metodología, actividades de aprendizaje, programa y recursos

### 4.1. Presentación metodológica general

El proceso de aprendizaje que se ha diseñado para esta asignatura se basa en lo siguiente:

La asignatura se asienta sobre clases presenciales al principio ya que el alumnado no dispone de conocimientos previos. Esta actividad irá acompañada del estudio teórico por parte de cada estudiante.

En grupos reducidos se establecerán sesiones de aprendizaje basado en problemas y/o aplicaciones de teoría y de sesiones prácticas en laboratorio. Será preciso trabajo autónomo de preparación de las actividades y entregas en las sesiones prácticas.

Complementario a la sesión de prácticas se realizará un trabajo tutorizado de forma personalizada y de carácter eminentemente práctico, hacia el final de cuatrimestre. Este trabajo será evaluado de forma independiente.

Así mismo se realizarán tutorías no académicas y podrían complementarse con otro tipo de actividades formativas voluntarias (seminarios, por ejemplo).

### 4.2. Actividades de aprendizaje

Las actividades de aprendizaje que ayudarán al alumno a lograr los objetivos previstos de la signatura son las siguientes:

- Desarrollo del temario de la asignatura en las clases de teoría.
- En las clases de problemas se trabajan con detalle los conceptos tratados en las clases de teoría.
- Las clases de prácticas son sesiones de trabajo de programación en laboratorio, tuteladas por un profesor, en las que participan los alumnos en grupos reducidos.
- Los alumnos también pueden asistir a tutorías personalizadas.

## 30205 - Arquitectura y organización de computadores 1

### 4.3. Programa

El programa a desarrollar en esta asignatura es:

- Arquitectura del Procesador: Interpretación y traducción, lenguaje máquina y ensamblador, entorno de desarrollo, representación y codificación de la información, operaciones básicas, almacenes, modos de direccionamiento, repertorio de instrucciones, traducción de estructuras de datos y control de lenguajes de alto nivel.
- Subrutinas: Llamadas a procedimiento. Bloque de activación. Caso práctico. Integración de código de alto nivel con código ensamblador y rutinas de biblioteca.
- Subsistema de E/S: Modelo genérico de registros de controlador de dispositivo. Métodos básicos de sincronización y transferencia. Excepciones. Integración de periféricos en microcontroladores.

### 4.4. Planificación de las actividades de aprendizaje y calendario de fechas clave

Calendario de sesiones presenciales y presentación de trabajos

La organización docente de la asignatura es la siguiente.

- Clases teóricas (2 horas semanales)
- Clases de problemas (1 hora semanal)
- Clases prácticas de laboratorio
  - o Escuela de Ingeniería y Arquitectura de Zaragoza: 2 horas cada dos semanas
  - o Escuela Universitaria Politécnica de Teruel: 1 hora a la semana
- Tutorías y actividades de evaluación

Los horarios de todas las clases y las fechas de las sesiones de prácticas se anunciarán con suficiente antelación a través de las webs del centro y de la asignatura.

### Trabajo del estudiante

La dedicación del estudiante para alcanzar los resultados de aprendizaje en esta asignatura se estima en 150 horas distribuidas del siguiente modo:

- 56 horas, aproximadamente, de actividades presenciales (clases teóricas, de problemas y prácticas en laboratorio)
- 51 horas de estudio personal efectivo (estudio de apuntes y textos, resolución de problemas, preparación clases y prácticas, desarrollo de programas)
- 40 horas de trabajo de programación en equipo
- 3 horas de examen final escrito

El calendario de exámenes y las fechas de entrega de trabajos de evaluación se anunciará con suficiente antelación.

### 4.5. Bibliografía y recursos recomendados

Teruel:

BB	Hohl, William. ARM Assembly language : fundamentals and techniques / William Hohl Boca Raton (Florida) : CRC Press, 2009
BC	Furber, Stephen B. [q(Stephen Bo)]. ARM system-on-chip architecture / Steve Furber . - 2nd ed. Harlow : Addison-Wesley, 2000
BC	Seal, David. ARM Architecture Reference Manual / David Seal (2nd ed.) [s.l.] : Addison-Wesley. 2000
BC	Sloss, Andrew. ARM system developer's guide : designing and optimizing system software / Andrew Sloss, Dominic Syme

## 30205 - Arquitectura y organización de computadores 1

**BC**

Chris Wright ; with a contribution by John Rayfield San Francisco, CA : Elsevier/ Morgan Kaufman, cop. 2004  
Stallings, William. Organización y arquitectura de computadores. 7ª ed. / William Stallings ; traducción Antonio Cañas Vargas ... [et al.]. - coordinado y revisado técnicamente por Alberto Prieto Espinosa. - Madrid : Pearson Educación, D. L. 2006

Zaragoza:

- 1.ARM Assembly Language: Fundamentals and Techniques, William Hohl. CRC press. 2009
- 2.ARM System-on-Chip Architecture (2nd ed.), Steve Furber. Addison-Wesley. 2000
- 3.ARM Architecture Reference Manual (2nd ed.), David Seal. Addison-Wesley. 2000
- 4.ARM System Developer's Guide: Designing and Optimizing System Software, Andrew N. Sloss, Dominic Symes, Chris Wright. Elsevier. 2004
- 5.Organización y Arquitectura de computadores(7ª ed.). W. Stallings. Prentice Hall. 2006