

Arquitectura de computadores y periféricos

Código: 102685

Créditos ECTS: 7.5

Titulación		Tipo	Curso	Semestre
2500895 Ingeniería Electrónica de Telecomunicación		OB	2	2
2500898 Ingeniería de Sistemas de Telecomunicación		OB	2	2

La metodología docente y la evaluación propuestas en la guía pueden experimentar alguna modificación en función de las restricciones a la presencialidad que impongan las autoridades sanitarias.

Contacto

Nombre: Ana Ripoll Aracil

Correo electrónico: Ana.Ripoll@uab.cat

Uso de idiomas

Lengua vehicular mayoritaria: español (spa)

Algún grupo íntegramente en inglés: No

Algún grupo íntegramente en catalán: No

Algún grupo íntegramente en español: No

Equipo docente

Dolores Isabel Rexachs del Rosario

Vicente José Ivars Camáñez

Prerequisitos

Aunque no hay prerequisitos formalmente establecidos, es indispensable un buen conocimiento del funcionamiento básico de un computador y un cierto conocimiento de sus unidades funcionales y de los sistemas digitales. (Fundamentos de Informática , Sistemas digitales y lenguajes de descripción de hardware).

Objetivos y contextualización

Los objetivos básicos de la asignatura son:

- Definir el modo de funcionamiento básico de un sistema empotrado (embedded) basado en un microcontrolador.
- La adquisición de las habilidades fundamentales de la programación en lenguaje ensamblador.
- Dar una visión general de los periféricos más usuales, como funcionan, como controlarlos, y como conectarlos a un equipo informático basado en un microcontrolador (así como su programación en lenguajes de alto nivel, como C y ensamblador).

Competencias

Ingeniería Electrónica de Telecomunicación

- Aprender nuevos métodos y tecnologías en base a sus conocimientos básicos y tecnológicos, con gran versatilidad de adaptación a nuevas situaciones.
- Hábitos de pensamiento
- Hábitos de trabajo personal
- Trabajo en equipo

Ingeniería de Sistemas de Telecomunicación

- Aprender nuevos métodos y tecnologías en base a sus conocimientos básicos y tecnológicos, con gran versatilidad de adaptación a nuevas situaciones.
- Hábitos de pensamiento
- Hábitos de trabajo personal
- Trabajo en equipo

Resultados de aprendizaje

1. Aplicar los fundamentos de lenguajes de descripción de dispositivos de hardware.
2. Desarrollar aplicaciones en tiempo real.
3. Desarrollar el pensamiento sistémico.
4. Desarrollar estrategias de aprendizaje autónomo.
5. Desarrollar la capacidad de análisis y de síntesis.
6. Desarrollar un pensamiento y un razonamiento crítico.
7. Gestionar el tiempo y los recursos disponibles
8. Gestionar el tiempo y los recursos disponibles. Trabajar de forma organizada.
9. Realizar programación en tiempo real, concurrente, distribuida y basada en eventos, así como el diseño de interfaces persona-computador.
10. Trabajar cooperativamente.
11. Utilizar los fundamentos de diseño, verificación y validación de software en la descripción de sistemas hardware basados en lenguajes de descripción del hardware a alto nivel.

Contenido

INTRODUCCIÓN. ARQUITECTURA DE COMPUTADORES

- Conceptos básicos: microprocesadores y microcontroladores. Arquitecturas típicas.
- Evolución histórica de los microprocesadores: una clasificación generacional con ejemplos representativos.
- Repertorio de instrucciones. Programación en ensamblador.

PROGRAMACIÓN DE LOS MICROCONTROLADORES

- Microcontroladores: organización interna, interfaz externa. Mapa de memoria.
- Repertorio de instrucciones: movimiento de información, aritméticas, lógicas, saltos condicionales e incondicionales, subrutinas y misceláneas.
- Modos de direccionamiento: inmediato, directo y extendido, indexado, inherente y relativo.
- Lenguaje ensamblador: formato, directivas, macros, opciones, errores.

CONCEPTOS DE ENTRADA / SALIDA

- El espacio de Entrada/Salida (E/S) en los computadores.
- Clasificación de los dispositivos de E/S. Módulos de E/S.
- Programación de los dispositivos de E/S.
- Sincronización con el procesador: Encuesta programada, interrupciones y DMA.
- Ejemplos de dispositivos de E/S.

MEMORIA

- Tipos de memorias,

- Organización de la memoria

- Jerarquía de la memoria.

DISEÑO DE SISTEMAS BASADOS EN MICROCONTROLADORES

- Componentes de apoyo a la CPU.

- Herramientas para el diseño de sistemas basados en microcontroladores: Placas de evaluación, simuladores y monitores.

- Depuradores: conceptos básicos, técnicas hardware y software, conceptos avanzados (Background Debug Mode).

Metodología

- Clases magistrales: Los conocimientos propios de la asignatura se expondrán en forma de clases magistrales. En ellas se mostrarán al alumno los conceptos básicos expuestos en el temario de la asignatura e indicaciones de cómo completar y profundizar estos contenidos. Las clases magistrales son las actividades en las que se exige menos interactividad al estudiante: están concebidas como un método fundamentalmente unidireccional de transmisión de conocimientos del profesor al alumno. Las clases magistrales, a pesar de ser principalmente una explicación por parte del profesor, se promoverá la participación del alumno, planteando preguntas y problemas para comprobar la buena comprensión de la materia expuesta.
- Seminarios: La misión de los seminarios es doble. Por un lado, se trabajarán los conocimientos científico-técnicos expuestos en las clases magistrales para completar su comprensión y profundizarlos. Para ello, se desarrollarán actividades diversas, desde la típica resolución de problemas hasta la discusión de casos prácticos. Se implementarán metodologías de aprendizaje y resolución de problemas cooperativo. Por otra parte, los seminarios serán el foro natural en el que discutir en común el desarrollo del trabajo práctico, aportando los conocimientos que le faltan al estudiante para llevarlo adelante, o indicándole donde puede adquirirlos. La misión de los seminarios es hacer de puente entre las clases magistrales y el trabajo práctico, que promoverá la capacidad de análisis y síntesis, el razonamiento crítico, y que entrenará al estudiante en la resolución de problemas.
- Practicum: El alumno recibirá un dossier con el trabajo práctico que deberá desarrollar durante el curso. Este trabajo práctico se basa en el diseño y programación de programas en ensamblador y en C, que permitan comprender el funcionamiento de un computador y aprender los mecanismos del subsistema de Entrada / Salida. Las prácticas se desarrollarán en grupos de alumnos. El practicum incluye sesiones en el laboratorio, de 2 horas de duración, donde realizará la implementación y depuración de los programas.

Este planteamiento del trabajo está orientado a promover un aprendizaje activo y desarrollar las competencias de capacidad de organización, planificación, comunicación oral y escrita, trabajo en equipo y razonamiento crítico. Se valorará la calidad del proyecto realizado, su presentación y su funcionamiento.

Actividades

Título	Horas	ECTS	Resultados de aprendizaje
Tipo: Dirigidas			
Laboratorio	10	0,4	3, 4, 5, 8, 9
Problemas	10	0,4	3, 4, 5, 6, 8, 9, 10
Teoría	29	1,16	2, 5, 6, 9

Tipo: Autónomas

Estudio	125,5	5,02	3, 4, 5, 6, 8
---------	-------	------	---------------

Evaluación

a) Proceso y actividades de evaluación programadas

La asignatura consta de las actividades de evaluación:

- Pruebas parciales de las clases magistrales y seminarios: A lo largo del curso se realizan unas pruebas parciales (entre 2 o 3) donde el alumno verifica que ha adquirido los conocimientos y habilidades definidos en los objetivos de la asignatura, así como sus competencias (trabajadas en las clases magistrales y seminarios). El % de la nota media de las notas de estas pruebas, respecto al total de la asignatura, es del 60%. Se debe considerar que, individualmente, la nota de cada prueba parcial no supera el 50% de la calificación final del curso.

Para poder aprobar la asignatura, mediante la evaluación continua, hay que obtener una nota mínima de 4 en cada una de estas pruebas parciales y además, que la nota media de las notas de estas pruebas parciales sea mayor o igual que 5.

- Sesiones de prácticas de laboratorio: En estas sesiones de prácticas los alumnos ponen en práctica, sobre una placa de desarrollo, los conocimientos adquiridos en las clases magistrales y seminarios. El % de la nota final de estas sesiones de prácticas, con respecto a la nota total de la asignatura, es del 30%.

Para poder aprobar la asignatura mediante la evaluación continua, la nota final de estas sesiones de prácticas debe ser mayor o igual que 5. Es obligatoria la asistencia a todas las sesiones de prácticas. Se tolera una ausencia a una sesión de prácticas por motivos justificados oficialmente, pero en cualquier caso, antes de ausentarse, se debe consultar previamente al profesor de la sesión de prácticas. Hay que tener en cuenta que las sesiones de prácticas son actividades no recuperables, por tanto, suspenderlas con una nota inferior a 5, supone no poder aprobar la asignatura.

- Prueba de prácticas: En esta prueba, el alumno verifica que ha alcanzado los conocimientos adquiridos en las sesiones de prácticas. El % de la nota de esta prueba de prácticas, respecto a la nota total de la asignatura, es del 10%. Para poder aprobar la asignatura mediante la evaluación continua, la nota de esta prueba de prácticas debe ser mayor o igual que 4 y la nota media de la nota total de las sesiones prácticas más la de la prueba de prácticas, ha ser mayor o igual que 5.

b) Programación de actividades de evaluación

La Planificación temporal de las actividades de evaluación se dará el primer día de la asignatura. Se hará pública a través del Campus Virtual y en la web de la Escuela de Ingeniería, en el apartado de exámenes.

c) Proceso de recuperación

Todo alumno que haya suspendido la asignatura por el método de evaluación continua (y teniendo aprobada la nota final de las sesiones de prácticas ya que estas son no recuperables), podrá presentarse a una prueba de recuperación. En esta prueba, el alumno deberá examinarse de las pruebas parciales de las clases magistrales y / o de la prueba de prácticas, que ha realizado mediante la evaluación continua y de las que ha obtenido una nota inferior a 5. por lo tanto, el alumno podrá mantener, para la prueba de recuperación, las notas de estas pruebas de la evaluación continua que sean mayores o iguales que 5.

Para poder aprobar la asignatura, por el proceso de recuperación, las notas de las pruebas parciales de las clases magistrales y seminarios (las mantenidas del proceso de evaluación continua y las obtenidas en la prueba de recuperación) deben ser mayores o iguales que 5 y la nota media de estas notas de las pruebas parciales debe ser mayor o igual que 5.

Igualmente, la nota de la prueba de prácticas obtenida en la prueba de recuperación debe ser mayor o igual que 5 y la nota media de esta nota de la prueba de prácticas con la nota total de las sesiones prácticas, debe ser mayor o igual que 5.

Importante, según la normativa académica de la UAB, el estudiante puede presentarse a la recuperación siempre que se haya presentado a un conjunto de actividades de evaluación que representen, al menos, dos

terceras partes de la nota total de la asignatura . De estos, se podrán presentar en la recuperación, aquellos estudiantes que tengan como nota final de todas las actividades evaluables de la asignatura, una calificación superior a 3,5.

d) Procedimiento de revisión de las calificaciones

Para cada actividad de evaluación, se indicará un lugar, fecha y hora de revisión en la que el estudiante podrá revisar la actividad con el profesor. En este contexto, se podrán hacer reclamaciones sobre la nota de la actividad, que serán evaluadas por el profesorado responsable de la asignatura. Si el estudiante no se presenta en esta revisión, no se le revisará posteriormente esta actividad.

e) Calificaciones

Matrículas de honor. Otorgar una calificación de matrícula de honor es decisión del profesorado responsable de la asignatura. La normativa de la UAB indica que las MH sólo se podrán conceder a estudiantes que hayan obtenido una calificación final igual o superior a 9.00. Se puede otorgar hasta un 5% de MH del total de estudiantes matriculados.

Un estudiante se considerará no evaluable (NA) si no se ha presentado a ninguna de las actividades evaluables de la asignatura.

f) Irregularidades por parte del estudiante, copia y plagio

Sin perjuicio de otras medidas disciplinarias que se estimen oportunas y de acuerdo con la normativa académica vigente, se calificarán con un cero las irregularidades cometidas por el estudiante que puedan conducir a una variación de la calificación de un acto de evaluación. Por lo tanto, copiar, dejar copiar, plagiar, engañar, etc en cualquiera de las actividades de evaluación, implicará suspender con un cero, y si es necesario superarla para aprobar toda la asignatura, quedará suspendida. No serán recuperables las actividades de evaluación calificadas de esta forma y por este procedimiento y por lo tanto la asignatura será suspendida directamente sin oportunidad de recuperarla en el mismo curso académico.

Actividades de evaluación

Título	Peso	Horas	ECTS	Resultados de aprendizaje
Informe del laboratorio	0.1	2	0,08	1, 3, 4, 5, 6, 7, 8, 10, 11
Prueba Individual: Procesador y Repertorio de Instrucciones	0.24	2	0,08	1, 3, 5, 6, 7, 8, 9, 11
Prueba individual de prácticas	0.2	1	0,04	1, 2, 3, 5, 6, 7, 9, 11
Prueba individual: Conceptos de Entrada/Salida	0.24	2	0,08	1, 2, 3, 4, 5, 6, 7, 8, 9, 11
Prueba individual: Sistema de memoria	0.12	2	0,08	1, 3, 4, 5, 6, 7, 8, 11
Trabajo previo y durante el laboratorio	0.1	4	0,16	1, 3, 4, 5, 6, 7, 8, 9, 10, 11

Bibliografía

- "Organización y Arquitectura de Computadores. Principios de estructura y funcionamiento" William Stallings. Ed. .Megabyte. Noriega Editores
- Felix García Carballera "Problemas resueltos de estructura de computadores". Paraninfo.
- "Computer Organization & Design. The hardware/software interface" David Patterson/John L. Hennessy. Ed. Morgan Kaufmann Publishers.
- "Computer Systems Design and Architecture" Vicent P. Heuring / Harry F. Jordan. Ed. Addison-Wesley
- Designing Embedded Hardware, 2nd Edition By [John Catsoulis](#) Publisher: O'Reilly Media Released: May 2005
- The Art of Readable Code Simple and Practical Techniques for Writing Better Code By Dustin Boswell, Trevor Foucher Publisher: O'Reilly Media Released: November 2011

- Designing Mobile Interfaces Patterns for Interaction Design By Steven Hoober, Eric Berkman Publisher: O'Reilly Media Released: November 2011
- Making Embedded Systems Design Patterns for Great Software By Elecia White Publisher: O'Reilly Medi Released: October 2011