

Asignatura: **Sistemas Embebidos**

Código:10-09811

RTF

5

Semestre: 7

Carga Horaria

80

Bloque: Tecnologías Aplicadas

Horas de Práctica

40

Departamento: Computación

## Correlativas:

- Electrónica Digital 3

## Contenido Sintético:

- Visión histórica de los sistemas embebidos y de tiempo real
- Aplicar estándares, herramientas relevantes y criterios de ingeniería.
- Interfaz de entrada / salida y comunicación
- Técnicas para el funcionamiento de baja potencia
- Sistemas integrados móviles y en red
- Subsistemas periféricos
- Estrategias de Implementación para Sistemas Embebidos Complejos
- Problemas avanzados de entrada / salidas
- Integración, pruebas y validación de sistemas embebidos
- Mantenimiento, sostenibilidad, manufacturabilidad de sistemas embebidos

## Competencias Genéricas:

- CG1: Identificar, formular y resolver problemas de ingeniería.
- CG2: Concebir, diseñar y desarrollar proyectos de ingeniería (sistemas, componentes, productos o procesos).
- CG6: Desempeñarse de manera efectiva en equipos de trabajo.

Aprobado por HCD: 1042-HCD-2023

RES: Fecha:27/11/2023



Competencias Específicas:

CE4.3: Analizar, diseñar, programar, implementar, probar, depurar y evaluar hardware y software para sistemas de computación de propósitos específicos. (M)

CE4.4: Analizar, diseñar, programar, implementar, probar, depurar y evaluar hardware y software para sistemas de computación de propósitos generales

CE4.5 Analizar, diseñar, implementar y probar sistemas embebidos y su software asociado. (A)

CE6.3 Proyectar, desarrollar, dirigir, controlar, construir, operar y mantener sistemas operativos embebidos, para dispositivos móviles y de tiempo real. (A)

CE7.2 Proyectar, desarrollar, dirigir, controlar, construir, operar y mantener Sistemas Embebidos, sus periféricos y software de soporte. (A)

CE7.2.6 Diseñar circuitos impresos que integren componentes electrónicos y periféricos en un sistema embebido. (A)

CE7.2.7 Comprender, configurar y utilizar los protocolos de comunicación utilizados en sistemas embebidos. (A)

## Presentación

Esta asignatura se centra en los sistemas embebidos, los elementos que se utilizan para su construcción y los conocimientos necesarios para comprender su funcionamiento, afrontar su diseño y tomar decisiones sobre su implementación.

La asignatura comenzará con una introducción dónde se presentan las características fundamentales de los sistemas embebidos, explicando los elementos principales que los forman, las métricas más importantes que afectan a su desarrollo, así como las distintas posibilidades de abordar la implementación de su hardware de cómputo. Tras la introducción, se expondrán los distintos niveles de diseño de un sistema embebido, así como las técnicas empleadas, tanto de diseño como de validación, que permiten mejorar la productividad y la calidad de los sistemas desarrollados. Entre las técnicas explicadas, se describe, mediante un caso de uso, la técnica de co-diseño hardware-software, con el fin de que el alumno pueda conocer los retos y beneficios que supone la utilización de la misma en el desarrollo de sistemas embebidos.

A continuación se definirá el concepto de arquitectura de un sistema embebido, y se describirán los elementos más característicos que se emplean en las estructuras que forman parte de la arquitectura, así como la funcionalidad que proporcionan, justificando su idoneidad para formar parte de este tipo de sistemas.

Finalmente, se estudiará el proceso de desarrollo de los sistemas embebidos, definiendo el ciclo de vida que permite su especificación y validación, así como la posterior implementación y despliegue del sistema. Se hará, además, una breve introducción sobre algunas herramientas que facilitan el proceso. Paralelamente a la exposición teórica, el alumno llevará a cabo prácticas que refuercen los contenidos teóricos. En este sentido tendrá que ser capaz de desarrollar, desplegar y depurar software sobre el hardware propio de los sistemas embebidos, así como familiarizarse con la simulación un sistema embebido que implique aspectos hardware y software. Para las prácticas se utilizará primeramente el hardware básico de un sistema embebido, implementando drivers sencillos que permitan una comunicación serie, la gestión de las interrupciones y los servicios básicos de temporización.

Posteriormente se utilizarán servicios más avanzados, que faciliten la programación de funciones en el tiempo sobre un ejecutivo cíclico. Finalmente, el alumno realizará prácticas sobre sistemas multitarea, familiarizándose con la utilización de servicios de temporización, mecanismos de protección de acceso a recursos compartidos, y mecanismos de comunicación, tanto síncrona como asíncrona, entre esas tareas.

## Contenidos

### **Unidad 1. Introducción a los sistemas embebidos**

Visión histórica. Sistemas embebidos y de tiempo real. Fundamentos de los sistemas embebidos. Estándares, herramientas relevantes y criterios de ingeniería. Interfaz de entrada/salida. Comunicación serie entre una PC y un sistema embebido. Gestión del tiempo y señales analógicas. Técnicas para el

funcionamiento de baja potencia.

## **Unidad 2. Técnicas de diseño y validación de sistemas embebidos**

Estrategias de Implementación para Sistemas Embebidos Complejos. Máquinas de estado finito y Real-Time Clock. Modularización aplicada a los sistemas embebidos. procesos de diseño e implementación de un sistema embebido. Elicitación de requerimientos del proyecto y casos de uso. Diseño del hardware. Diseño del software. Implementación de la interfaz de usuario. Implementación de lectura de sensores. Control de actuadores. Implementación del comportamiento del sistema. Verificación del comportamiento del sistema. Documentación del Sistema

## **Unidad 3. Arquitectura de un sistema embebido**

Sistemas integrados móviles y en red. Subsistemas periféricos. Estrategias de Implementación para Sistemas Embebidos Complejos. Problemas avanzados de entrada/salidas. Comunicación con otros sistemas embebidos, comunicación con sistemas más complejos. Comunicaciones interplacas, comunicaciones de corta distancia y larga distancia.

**Unidad 4. Sistemas operativos en sistemas embebidos y seguridad** Cargadores de arranque, Sistemas operativos genéricos, sistemas de tiempo real, planificación de tareas y comunicación entre tareas. Gestión de memoria y drivers de dispositivos seguridad en sistemas embebidos.

## **Unidad 5. Proceso y entornos de desarrollo de los sistemas embebidos**

Integración continua y devops, pruebas y validación de sistemas embebidos. Mantenimiento, sostenibilidad, manufacturabilidad de sistemas embebidos. Actualizaciones de manera remota.

## **Metodología**

El desarrollo general de la materia se sustenta en clases teórico-prácticas. Por ello las estrategias de enseñanza que hemos seleccionado para llevar adelante nuestra propuesta son la exposición dialogada, el estudio de casos y la resolución de problemas. Cada unidad se desarrolla a partir de un material bibliográfico obligatorio. A su vez se ofrecerán trabajos prácticos que orientan el proceso de lectura y análisis del contenido como forma de evaluación y acreditación de cada unidad. Los trabajos prácticos estarán guiados por los casos de estudio presentados en cada unidad y se orientan a resolver problemas concretos aportando los contenidos teóricos necesarios para su resolución.

En ambientes de laboratorio el estudiante podrá realizar casos de estudio, verificar su implementación con el cumplimiento de los requerimientos funcionales a nivel software y componente de hardware.

## **Evaluación**

La evaluación está inspirada en los criterios de evaluación continua. Se valora el desarrollo de las competencias durante todo el proceso de aprendizaje de la asignatura mediante una serie de pruebas de carácter sumativo distribuidas a lo largo del curso, que permiten al estudiante abordar la asignatura de forma progresiva. Se garantiza la retroalimentación temprana en el proceso de aprendizaje del alumno y permite a los profesores hacer un seguimiento global,

con la posibilidad de actuar en caso de que lo aconsejen indicadores o situaciones determinadas. La evaluación de la parte relacionada con las prácticas se realizará al finalizar la actividad de laboratorio correspondiente.

Para la evaluación de la asignatura se utilizará la rúbrica, compuesta por los siguientes criterios, relacionados con los resultados del aprendizaje:

RA1. Enumerar las características específicas de los sistemas embebidos y describir cómo éstas afectan al su proceso de desarrollo.

RA2. Enumerar las distintas métricas de diseño de los sistemas embebidos.

RA3. Describir el papel de la unidad de procesamiento en el contexto de un sistema completo con E/S y memoria.

RA4. Describir cómo es la comunicación de la unidad de procesamiento con el mundo exterior por medio de los periféricos.

RA5. Identificar las alternativas de implementación de un sistema embebido y ser capaz de seleccionar la más adecuada acorde a las métricas de diseño marcadas como objetivo.

RA6. Identificar los distintos niveles de diseño de un sistema embebido y describir las técnicas utilizadas para desarrollarlos.

RA7. Definir el concepto de arquitectura de los sistemas embebidos y describir el proceso seguido para su especificación y validación.

RA8. Ser capaz de explicar el diseño de un sistema embebido.

RA9. Ser capaz de programar un sistema embebido y describir cómo el lenguaje de alto nivel se convierte en código ejecutable.

RA10. Ser capaz de emplear compiladores y entornos de desarrollo propios de los sistemas embebidos.

### **Instrumentos de evaluación**

El rendimiento de los alumnos será valorado mediante la toma de tres exámenes parciales con un recuperatorio y evaluación formativa. Los parciales buscan evidenciar los conocimientos y competencias adquiridas a través de presentación de casos prácticos (análisis, resolución de problemas, etc.) y/o respuestas a preguntas conceptuales. Constituyen en sí mismos una instancia de evaluación formativa ya que luego los estudiantes reciben una realimentación de los errores cometidos, además de resolver los temas del parcial en la clase siguiente.

Cada Actividad Práctica o de Laboratorio grupal propuesta (ver más adelante) demanda la presentación de prototipos funcionales e informes, donde los estudiantes explican individualmente las diferentes partes que componen el prototipo (criterios de diseño, fuentes consultadas, etc.), y particularmente sus detalles constructivos. Es el momento donde el estudiante pone en juego su participación en el equipo, el rol y peso de sus decisiones, su capacidad de comunicar detalles de diseño, y su correcto manejo y comprensión de las decisiones tecnológicas presentes en el prototipo. Para la entrega que demande

cada prototipo el equipo de cátedra genera un repositorio virtual con control de versiones.

## Condiciones de aprobación

### Condiciones de regularización

- Asistir al 80% de las clases.
- Aprobar todas las actividades prácticas de laboratorio con al menos el 60% o más de los criterios de evaluación expresados en la sección anterior.
- Aprobar los resultados de aprendizaje, con el 60% o más.
- Aprobar al menos dos parciales, incluido un recuperatorio, con el 60% o más de los contenidos evaluados.

### Condiciones de aprobación por promoción (no requiere examen final)

- Cumplir con todas las condiciones de regularización.
- Aprobar cada uno de los tres parciales propuestos, o un recuperatorio, con el 60% o más de los contenidos evaluados.

### Condiciones de aprobación por examen final

- Todas las condiciones de regularización expuestas anteriormente.
- Aprobación de un examen final con el 60% o más de los contenidos evaluados.

Para la nota final se promedian las notas obtenidas en cada una de las actividades prácticas y de laboratorio, y el resultado se promedia con las notas de los parciales.

## Actividades prácticas y de laboratorio

Se propone la realización de 5 (cinco) actividades donde se lleven a la práctica los conocimientos adquiridos en la asignatura y se desarrollen las competencias esperadas. Estos trabajos se realizan en grupos lo que permitirá que desarrollen competencias de trabajo en equipo y coordinación de tareas.

### APL1 – Cross Compilation, toolchain y testing en sistemas embebidos

En este trabajo los estudiantes aplican los conceptos de introducción a los sistemas embebidos en hardware real y hardware virtualizado. Especificación de requerimientos, pruebas unitarias, de integración y de sistema utilizando mocks y stubs.

APL2 – Sistemas operativos de tiempo real: En este trabajo los estudiantes aplican los conceptos básicos de conmutación de tareas y organización de memoria. Continúan mejorando sus herramientas de depuración y profiling para medir tiempos de latencia de una aplicación simple en un microcontrolador y lo comparan con un microprocesador con un sistema operativo de propósito general.

APL3 – Programación de drivers para sistemas embebidos: Todo componente electrónico que se conecte a una computadora necesita de un driver para



interactuar directamente con el sistema operativo. En este trabajo práctico se compilara un bootloader, un kernel y luego se desarrolla un driver sencillo para un

sistema operativo de propósito general que adquiera datos de un sensor y comunique los datos por algún canal como bluetooth o zigbee.

APL4 – API Rest y su integración con lenguajes de bajo nivel en hardware virtualizado: En este trabajo los estudiantes desarrollan actividades que les permitirán aplicar los conocimientos de redes e integrar el resto de los conceptos aprendidos en los trabajos anteriores. Desarrollan un servidor simple en un microcontrolador virtualizado y otro en un microprocesador de propósitos generales virtualizado.

APL5 – Implementación de un sistema de computación complejo: En este trabajo práctico de laboratorio los estudiantes diseñan e implementan un sistema de computación complejo que incluye la adquisición, almacenamiento, transmisión y visualización de datos distribuidos. Para ejercitar e integrar los conocimientos adquiridos, se realizan ejercicios con diferentes herramientas para cada una de las funcionalidades, se acompaña y guía a los estudiantes en el proceso de diseño e implementación. Finalmente deben presentar una prueba de concepto funcional junto a la documentación del proyecto y los casos de prueba diseñados, implementados y ejecutados.

## Resultados de aprendizaje

Al terminar con éxito esta asignatura/enseñanza, los estudiantes serán capaces de:

RA1. Enumerar las características específicas de los sistemas embebidos y describir cómo éstas afectan al su proceso de desarrollo.

RA2. Enumerar las distintas métricas de diseño de los sistemas embebidos.

RA3. Describir el papel de la unidad de procesamiento en el contexto de un sistema completo con E/S y memoria.

RA4. Describir cómo es la comunicación de la unidad de procesamiento con el mundo exterior por medio de los periféricos.

RA5. Identificar las alternativas de implementación de un sistema embebido y ser capaz de seleccionar la más adecuada acorde a las métricas de diseño marcadas como objetivo.

RA6. Identificar los distintos niveles de diseño de un sistema embebido y describir las técnicas utilizadas para desarrollarlos.

RA7. Definir el concepto de arquitectura de los sistemas embebidos y describir el proceso seguido para su especificación y validación.

RA8. Ser capaz de explicar el diseño de un sistema embebido.

RA9. Ser capaz de programar un sistema embebido y describir cómo el lenguaje de alto nivel se convierte en código ejecutable.

RA10. Ser capaz de emplear compiladores y entornos de desarrollo propios de los sistemas embebidos.

RA11. Interactuar en equipos variados, con objetivos claros, utilizando herramientas acordadas para la modalidad de trabajo acordado, logrando comunicación efectiva y concreción de resultados

Desagregación de Competencias Específicas	Resultados de Aprendizaje
CE4.5.A Analizar sistemas embebidos y su software asociado.	RA1,RA2
CE4.5.B Diseñar sistemas embebidos y su software asociado.	RA1, RA3,RA4,RA 5
CE4.5.C Implementar sistemas embebidos y su software asociado.	RA1, RA2, RA9
CE4.5.D Probar sistemas embebidos y su software asociado.	RA2
CE7.2.A Proyectar sistemas embebidos, sus periféricos y software de soporte.	RA3, RA4, RA5, RA6
CE7.2.B Desarrollar sistemas embebidos, sus periféricos y software de soporte.	RA3, RA4
CE7.2.C Dirigir proyectos de diseño de sistemas embebidos, sus periféricos y software de soporte.	RA4,RA7
CE7.2.D Controlar sistemas embebidos, sus periféricos y software de soporte.	RA4
CE7.2.E Construir sistemas embebidos, sus periféricos y software de soporte.	RA4, RA8, RA9
CE7.2.F Operar sistemas embebidos, sus periféricos y software de soporte.	RA7
CE7.2.G Mantener sistemas embebidos, sus periféricos y software de soporte.	RA7,RA8
CG6: Desempeñarse de manera efectiva en equipos de trabajo.	RA11

Tabla 1: Desagregación de competencias y resultados de aprendizaje

## Bibliografía

- Edward A. Lee and Sanjit A. Seshia, Introduction to Embedded Systems, A Cyber-Physical Systems Approach, Second Edition, MIT Press, ISBN 978-0- 262-53381-2, 2017.
- R. Barry, Mastering the FreeRTOS™ Real Time Kernel, 2016
- M. VanderVoord, Embedded Testing with Unity and CMock
- Benigno Jacob.; Reusable Firmware Development, A Practical Approach to APIs, HALs and Drivers; editorial apress; ISBN-13 (electronic): 978-1-4842- 3297-2; 2017.-