

## Formulario de aprobación de curso de posgrado/educación permanente

**Asignatura:** Sistemas embebidos para tiempo real

<b>Modalidad:</b> (posgrado, educación permanente o ambas)	<b>Posgrado</b> <input checked="" type="checkbox"/>
	<b>Educación permanente</b> <input type="checkbox"/>

**Profesor de la asignatura:** Dr. Ing. Julián Oreggioni, Profesor Adjunto, IIE

**Profesor Responsable Local:** No corresponde

**Otros docentes de la Facultad:**

Dr. Ing. Leonardo Steinfeld, Profesor Adjunto, IIE  
Dr. Ing. Leonardo Barboni, Profesor Adjunto, IIE  
Ing. Javier Schandy, Asistente, IIE  
Ing. Mauricio González, Ayudante, IIE  
Dr. Ing. Conrado Rossi, Profesor Agregado, IIE

**Docentes fuera de Facultad:** No corresponde

**Programa(s) de posgrado:** Maestría en Ingeniería Eléctrica, Doctorado en Ingeniería Eléctrica.

**Instituto o unidad:** IIE

**Departamento o área:** Departamento de Electrónica

---

**Horas Presenciales:** 50

**Nº de Créditos:** 10

**Público objetivo:** Ingenieros o técnicos con interés en el desarrollo de sistemas electrónicos embebidos.

**Cupos:** 6 estudiantes (máximo). Sin cupo mínimo.

---

**Objetivos:**

Al finalizar el curso el estudiante será capaz de:

- Definir sistemas embebidos y sistemas embebido de tiempo real, e identificar sus características.
- Describir el proceso de compilación de una aplicación, la inicialización de datos, el arranque de programa y el cargado de programa.
- Programar utilizando control de versiones.
- Describir el hardware asociado a un sistema embebido. Describir los conceptos y bloques básicos de microcontroladores y periféricos más relevantes.
- Utilizar e interpretar documentación técnica compleja (manuales de usuario, hojas de datos, etc.).
- Describir el flujo de ejecución de un microcontrolador: contexto principal y de interrupciones. Describir la secuencia de ejecución de interrupciones. Reconocer el problema de datos compartidos. Identificar los tiempos involucrados en las interrupciones.

- Describir las principales arquitecturas de software e identificar sus características. Comparar las ventajas y desventajas de cada una. Seleccionar la arquitectura adecuada para una aplicación particular.
- Conocer los conceptos básicos de los sistemas operativos para tiempo real. En particular el concepto de tareas, estados de una tarea, el rol del planificador, y la vinculación entre tareas y datos.
- Analizar y aplicar técnicas de reducción del consumo a nivel circuito y a nivel sistema, y de programación para bajo consumo.
- Definir los principales conceptos de test. Identificar las diferencias entre test en PC y test en sistemas embebidos. Describir los métodos de test embebido: en host y con simuladores.
- Diseñar el hardware y desarrollar el software embebido para implementar un sistema basado en microcontrolador. En particular, balanceando adecuadamente el procesamiento que se realiza dentro de las interrupciones y en la tarea, separando el código dependiente del independiente de hardware, entre otros.
- Enfrentarse a un problema real e implementar una solución al mismo utilizando un sistema embebido para tiempo real.

**Conocimientos previos exigidos:** Programación en lenguaje de alto nivel (preferentemente en lenguaje C) con énfasis en tipos de datos avanzados (punteros, arreglos, estructuras y uniones), modularización, y encapsulamiento. Conocimientos básicos de microprocesadores, incluyendo modos de direccionamiento, lenguaje de máquina, lenguaje de ensamblador, set de instrucciones, interrupciones, uso de stack y subrutinas. Electrónica digital, incluyendo el transistor MOS como llave, y circuitos digitales. Niveles lógicos y márgenes de ruido; tiempos de propagación, tiempos de subida, y bajada; consumo estático y dinámico. Estructura básica de compuertas lógicas CMOS, compuertas con etapas de salida de colector/drain abierto y tercer estado. Niveles lógicos TTL y su interfaz con lógica CMOS. Manejo de hojas de datos.

**Conocimientos previos recomendados:** Conversores analógico-digitales, Interfaces de comunicación (RS232, UART o similares), y redes de datos.

---

#### Metodología de enseñanza:

Descripción de la metodología:

El curso comprende clases teóricas, laboratorios y la realización de un proyecto final. Se propone una metodología que promueva la participación activa de los estudiantes, en todas las actividades del curso. En las clases de teórico se propondrán actividades de trabajo grupales e individuales. El trabajo de laboratorio se realizará en forma grupal mientras que se tenderá a que el proyecto final se realice de forma individual.

Los laboratorios son de asistencia obligatoria, y tienen una evaluación individual y grupal. Se entregarán a los estudiantes un kit de desarrollo que permite, conectándolo a una computadora, la preparación y realización de la práctica.

El proyecto final también se desarrollará sobre la base de un kit de desarrollo provisto por los docentes. La complejidad del proyecto será acorde a un curso de posgrado, donde se espera aprovechar al máximo la madurez y los conocimientos previos del estudiante, para explorar temas avanzados de sistemas embebidos. Por ejemplo, implementación de técnicas del estado del arte para reducción del consumo, problemas avanzados sobre procesamiento de datos, diseño basado en modelos, proyectos con sistemas operativos de tiempo real, aplicación de técnicas para confiabilidad en sistemas críticos, entre otros.

A lo largo del curso los estudiantes deberán realizar al menos dos presentaciones, y se deberá entregar una documentación.

Detalle de horas:

- Horas de clase (teórico): 22
- Horas de clase (práctico): 0
- Horas de clase (laboratorio): 16
- Horas de consulta: 10
- Horas de evaluación: 2
  - Subtotal de horas presenciales: 50
- Horas de estudio: 4
- Horas de resolución de ejercicios/prácticos: 16
- Horas proyecto final/monografía: 80
  - Total de horas de dedicación del estudiante: 150

---

**Forma de evaluación:**

La asistencia a las clases de laboratorio e instancias presenciales de proyecto es obligatoria. El curso no tiene examen y para su aprobación se deberá aprobar cada una de las partes del curso (laboratorio y proyecto). La calificación final será ponderada de la siguiente manera:

Laboratorio: 40 % considerando cuestionarios individuales, y el desempeño individual y grupal durante la práctica.

Proyecto: 60 % considerando el desempeño durante la ejecución del proyecto, los resultados obtenidos, las presentaciones y la documentación.

---

**Temario:**

1. Introducción y conceptos básicos: introducción a los sistemas embebidos y sistemas embebidos de tiempo real, hardware en sistemas embebidos, metodologías de desarrollo de software embebido.
2. Microcontroladores e interrupciones: arquitectura de microprocesadores y microcontroladores, periféricos, fundamentos de las interrupciones, latencia, el problema de datos compartidos.
3. Arquitecturas de software embebido: Round-Robin, Round-Robin con interrupciones, planificación por encolado de funciones, y sistema operativo para tiempo real. Análisis y selección de la arquitectura.
4. Introducción a los sistemas operativos para tiempo real: tareas y el estado de una tarea, el planificador, vinculación entre tareas y datos, semáforos y datos compartidos. Cola de mensajes, buzones y pipes, funciones de temporizado, eventos, manejo de memoria, manejo de rutinas de atención a interrupciones.
5. Sistemas embebidos de bajo consumo: técnicas de reducción del consumo a nivel circuito y a nivel sistema. Programación para bajo consumo.
6. Criterios básicos de diseño de un sistema embebido: criterios generales y específicos de sistemas operativos para tiempo real.

7. Otros temas de sistemas embebidos: diseño basado en modelos, test y técnicas para confiabilidad en sistemas críticos, otros tópicos de sistemas operativos para tiempo real: etc.

---

**Bibliografía:**

- "An Embedded Software Primer", David E. Simon, Addison-Wesley Professional, ISBN: 020161569X, 1999.
- "Real-Time Embedded Systems. Design principles and engineering practices", Xiaocong Fan, ISBN 978-0-12-801507-0, 2015.
- "Fundamentals of Embedded Software: Where C and Assembly Meet", Daniel W. Lewis, Prentice Hall, ISBN: 0130615897, 2001.
- "MicroC OS II: The Real Time Kernel", Jean J. Labrosse , CMP Books, ISBN: 1578201039, 2002.
- "Power aware design methodologies", M. Pedram, and J. M. Rabaey, eds. Springer Science & Business Media, ISBN 978-0-306-48139-0, 2002.
-

**Datos del curso**

---

**Fecha de inicio y finalización:** Marzo a julio de 2026

**Horario y Salón:**

**Arancel:** 4000 UI

[Si la modalidad no corresponde indique "no corresponde". Si el curso contempla otorgar becas, indíquelo]

**Arancel para estudiantes inscriptos en la modalidad posgrado:**

**Arancel para estudiantes inscriptos en la modalidad educación permanente:**

**Actualizado por expediente n.º:** 060180-000191-25

---