



# Planificaciones

6666 - Seminario de Electrónica II

Docente responsable: CRUZ JUAN MANUEL

## OBJETIVOS

1. Conocer y evaluar tecnologías actuales de implementación de sistemas embebidos (SE). En particular, microcontroladores de 32 bits.
2. Aplicar mejores prácticas de la Ingeniería de Software, útiles en el desarrollo de SE, que sirvan para organizar el ciclo de vida de un proyecto y mejorar la eficiencia del trabajo en equipo.
3. Explicar el rol de modelado en el desarrollo de SE. Modelar soluciones utilizando los diagramas y las notaciones de uso más frecuente en SE.
4. Desarrollar aplicaciones embebidas en lenguaje C y utilizando, cuando se justifique, un sistema operativo de tiempo real (RTOS), empleando técnicas de programación específicas para lograr eficiencia, confiabilidad y reusabilidad, ante limitaciones que frecuentemente se presentan en proyectos reales.
5. Analizar y sintetizar circuitos básicos de apoyo e interfaces.
6. Mejorar la habilidad para escribir documentos relacionados con un desarrollo de ingeniería.
7. Adquirir una base general sobre SE tal que permita continuar el aprendizaje relativo al área, tanto en otras materias de FIUBA como por cuenta propia.

## CONTENIDOS MÍNIMOS

-

### PROGRAMA SINTÉTICO

1. Tecnologías y arquitecturas de sistemas embebidos y microcontroladores.
2. Plataforma de desarrollo.
3. Elementos de la Ingeniería del Software.
4. Modelado de sistemas embebidos.
5. Programación de microcontroladores en lenguaje C.
6. Sistemas operativos de tiempo real (RTOS).
7. Temas complementarios.

### PROGRAMA ANALÍTICO

1. Tecnologías y arquitecturas de sistemas embebidos y microcontroladores.
  - 1.1. Áreas de aplicación de sistemas embebidos.
  - 1.2. Tecnologías de implementación.
  - 1.3. Comparaciones de performance.
  - 1.4. Elementos de la arquitectura de computadoras.
  - 1.5. Disponibilidad actual de microcontroladores.
  - 1.6. Descripción de arquitecturas ARM.
  - 1.7. Análisis de documentación de fabricantes: hojas de datos, notas de aplicación, diseños de referencia.
2. Plataforma de desarrollo.
  - 2.1. Introducción a un entorno de desarrollo para microcontroladores de 32 bits.
  - 2.2. Análisis del hardware empleado en el curso.

### 3.Elementos de la Ing. del Software.

3.1.Ciclo de vida de un proyecto. Modelos "cascada", "en V" e "iterativo".

3.2.Modularidad. Diseño top-down y bottom-up.

3.3.Verificación. Análisis estático.

3.4.Tecnologías y técnicas de depuración.

3.5.Documentación.

3.6.Metodologías ágiles.

3.7.Control de versiones.

### 4.Modelado de sistemas embebidos.

4.1.Modelado de software y de sistemas digitales.

4.2.Diseño basado en modelos.

4.3.Modelado de sistemas reactivos mediante Diagramas de Estado.

4.4.Diagramas de actividad y de secuencia del UML.

4.5.Revisión de herramientas de software para modelado.

### 5.Programación de microcontroladores en lenguaje C.

5.1.Análisis en bajo nivel de la asignación de memoria.

5.2.Estructura de programas reactivos sin RTOS: ciclo round-robin e interrupciones.

5.3.Recomendaciones de codificación C para sistemas embebidos.

### 6.Sistemas operativos de tiempo real (RTOS).

6.1.Componentes básicos de un RTOS.

6.2.Multitarea cooperativa y preemptiva.

6.3.Sincronización y comunicación entre tareas.

6.4.Aplicaciones.

### 7.Temas complementarios

7.1.ADC/DAC alta velocidad.

7.2.ADC/DAC alta resolución.

7.3.Interfaces de uso en sistemas embebidos: USB, CAN, I2C.

7.4.Circuitos de Apoyo (Watch Dog Timer, VDD, PWM).

7.5.Técnicas de diseño de sistemas embebidos de bajo consumo y/o a baterías – Convertidores DC/DC.

## **BIBLIOGRAFÍA**

Apuntes preparados por los docentes y capítulos seleccionados de libros, revistas y publicaciones de fabricantes.

**Bibliografía adicional:**

P. Marwedel; Embedded System Design; Springer; 2006.

J. Ganssle; Embedded Systems (World Class Designs); Elsevier; 2008.

S. Furber; ARM System-on-Chip Architecture; Pearson Education; 2nd ed., 2000.

S.C. McConnell; Code Complete: A Practical Handbook of Software Construction; Microsoft Press, 2nd ed., 2004.

D.E. Simon; An Embedded Software Primer; Addison-Wesley; 1999.

Q. Li, C. Yao; Real-Time Concepts for Embedded Systems; CMP Books; 2003.

J. Labrosse et. al.; Embedded Software: Know It All; Newnes; 2008.

T. Noergaard; Embedded Systems Architecture: A Comprehensive Guide for Engineers and Programmers; Newnes; 2005.

P. Laplante; Real-Time Systems Design and Analysis; Wiley-IEEE Press; 3rd ed., 2004.

**RÉGIMEN DE CURSADA****Metodología de enseñanza**

a) Semana 1 a 10 correspondientes a los capítulos 1 al 5 inclusive del programa. Clases teórico prácticas, conformadas por explicaciones conceptuales con participación de los alumnos y/o resolución de ejercicios en papel o sobre un entorno de desarrollo.

b) Semana 11 a 16, correspondientes a los capítulos 6 y 7 del programa. Clases teórico-prácticas alternadas con clases de consulta y seguimiento del progreso del trabajo práctico final.

**Modalidad de Evaluación Parcial**

Evaluación escrita con desarrollo de temas teóricos y resolución de problemas.

Para la firma de TP es obligatoria la presentación de un trabajo practico final, junto con un informe completo sobre el mismo.

**CALENDARIO DE CLASES**

Semana	Temas de teoría	Resolución de problemas	Laboratorio	Otro tipo	Fecha entrega Informe TP	Bibliografía básica
<1> 09/03 al 14/03	Introducción al curso - Arquitecturas y Tecnologías					
<2> 16/03 al 21/03	Arquitecturas y Tecnologías - Arquitecturas y Tecnologías					
<3> 23/03 al 28/03	Programación de Sistemas Embebidos en C - Programación de Sistemas Embebidos en C					
<4> 30/03 al 04/04	Programación de Sistemas Embebidos en C - Programación de Sistemas Embebidos en C		TP1			
<5> 06/04 al 11/04	Modelado de Sistemas Embebidos - Modelado de Sistemas Embebidos		TP1			
<6> 13/04 al 18/04	Modelado de Sistemas Embebidos		TP2			
<7> 20/04 al 25/04	Embebidos - Modelado de Sistemas Embebidos		TP2			
<8> 27/04 al 02/05	Arquitecturas y Tecnologías - RTOS					
<9> 04/05 al 09/05	Arquitecturas y Tecnologías - RTOS		TP3			
<10> 11/05 al 16/05	Interfaces - RTOS		TP3			
<11> 18/05 al 23/05	Repaso - Parcial					
<12> 25/05 al 30/05	RTOS - RTOS		TP4			
<13> 01/06 al 06/06	RTOS - RTOS		TP4			
<14> 08/06 al 13/06	Diseño PCBs - Proyecto Final		TP5 (final)			
<15> 15/06 al 20/06	Ingeniería de software - Proyecto Final		TP5 (final)			
<16> 22/06 al 27/06	Convertidores ADC/DAC - Proyecto Final		TP5 (final)			

## CALENDARIO DE EVALUACIONES

### Evaluación Parcial

Oportunidad	Semana	Fecha	Hora	Aula
1º	11	20/05	16:00	L15
2º	13	03/06	16:00	L15
3º	16	24/06	16:00	L15
4º				