

## DADES GENERALS

## Curs acadèmic

<b>Tipus de curs</b>	Màster de Formació Permanent
<b>Nombre de crèdits</b>	60,00 Crèdits ECTS
<b>Matrícula</b>	0 euros (importe preu públic pendent d'aprovació pel Consell Social Universitat de València.)
<b>Requisits d'accés</b>	<p>Els perfils d'ingrés recomanat es correspondran als perfils formatius dels següents plans d'estudi a nivell de grau: Enginyeria Electrònica de Telecomunicació, Enginyeria en Tecnologies i Serveis de Telecomunicació, Enginyeria de Tecnologies de Telecomunicació, Enginyeria Electrònica Industrial, Enginyeria Electrònica i Automàtica Industrial, Enginyeria Electrònica Industrial i Automàtica, Informàtica Industrial i Robòtica, Enginyeria Informàtica, o graus, nacionals o estrangers, amb una alta afinitat als ací enumerats.</p> <p>També podrien considerar graduats en: Enginyeria de l'Energia, Enginyeria Aeroespacial, Enginyeria Telemàtica, Enginyeria Robòtica, Enginyeria Física, o títols afins.</p> <p>En menor mesura, es podrien considerar perfils científics tradicionals com a Grau en Física, Grau en Matemàtiques o Grau en Ciència de Dades.</p> <p>Es permetrà l'accés a l'estudiantat que li falte menys d'un 10% dels crèdits per a acabar els estudis de grau, de forma condicionada al fet que s'aproven durant el mateix curs acadèmic.</p> <p>Respecte al perfil personal de l'estudiant que millor s'adapta, correspon a persones que vulguen aprofundir amb rigor en els coneixements i les habilitats que es requereixen per a especialitzar-se en les àrees descrites per a l'orientació professional. Així haurien de tindre una afinitat amb la microelectrònica com a motor en amplis sectors socioeconòmics, i especialment, com a contribució essencial en sectors estratègics industrials.</p>

<b>Modalitat</b>	Presencial
<b>Lloc d'impartició</b>	ETSE
<b>Horari</b>	Vesprades i dissabte

## Direcció

<b>Organitzador</b>	Escola Tècnica Superior d'Enginyeria (ETSE-UV)
<b>Direcció</b>	Abilio Candido Reig Escriba Profesor/a Titular de Universidad. Departament d'Enginyeria Electrònica. Universitat de València María Teresa Bacete Castelló Site Director. Maxlinear

## Terminis

<b>Preinscripció al curs</b>	Fins a 12/07/2024
<b>Data inici</b>	Setembre 2024
<b>Data fi</b>	Juny 2025

## Més informació

<b>Telèfon</b>	961 603 000
<b>E-mail</b>	<a href="mailto:informacio@adeituv.es">informacio@adeituv.es</a>

## PROGRAMA

## TEORIA (36 horas)

Tema 1. Fundamentos de Electrónica y Física de Semiconductores (6 horas).

- 1.1. Teoría de bandas y estructura cristalina.
- 1.2. Materiales semiconductores: propiedades ópticas y eléctricas.
- 1.3. La unión PN.

Tema 2. Dispositivos electrónicos (12 horas).

- 2.1. Diodos.
- 2.2. Transistor BJT.
- 2.3. Transistor MOS.

Tema 3. Dispositivos fotónicos (6 horas).

- 4.1. Fotodetectores.
- 4.2. Diodo LED y láser.
- 4.3. Dispositivos ópticos integrados.

Tema 4. Tecnología microelectrónica (12 horas).

- 4.1. Crecimiento cristalino y depósito de capas delgadas.
- 4.2. Recubrimiento de obleas: oxidación, implantación iónica y difusión.
- 4.3. Litografía óptica.
- 4.4. Definición de patrones por ataque químico y plasma.

LABORATORIO (24 horas)

1. Simulación y caracterización del diodo (3 horas).
2. Simulación y caracterización transistor BJT (3 horas).
3. Simulación y caracterización del transistor MOSFET (3 horas).
4. Implementación de un amplificador/conmutador (3 horas).
5. Caracterización dispositivos optoelectrónicos (6 horas).
6. Práctica/demo 1 sala gris (3 horas).
7. Práctica/demo 2 sala gris (3 horas).

---

TEORIA (30h)

1. Introducción al diseño microelectrónico analógico
2. Diseño CMOS
3. Amplificadores CMOS monoetapa
4. Amplificadores CMOS diferenciales
5. Bloques básicos de diseño y circuitos de acondicionamiento
6. Amplificador operacional

LABORATORIO (30h)

1. Introducción. Caracterización de transistores MOS.
2. Fuentes y espejos de corriente
3. Amplificadores monoetapa
4. Amplificadores diferenciales
5. Amplificador diferencial. Análisis preliminar.
6. Análisis paramétricos
7. Optimización y casos límite ("corners")
8. "Layout". Posicionado e interconexión.
9. Verificación del layout. DRC. LVS.
10. Extracción de parámetros. QRC. Xstream.

---

Los contenidos de la teoría se han distribuido en 8 temas y un total de 3 ECTS.

TEORIA (30h)

Tema 1: El transistor MOS en conmutación.

En este tema se estudian las características eléctricas básicas del transistor trabajando en conmutación. Se estudia la puerta básica inversora y Schmitt-Trigger.

Tema 2: Puertas básicas CMOS y bloques combinacionales.

Puertas NAND, NOR, XOR. Niveles de ruido. Decodificadores, codificadores, multiplexores, comparadores.

Tema 3: Circuitos secuenciales CMOS.

Latch y flip-flop. Registros de desplazamiento, contadores y otros bloques combinacionales. Máquinas de estados.

Tema 4: Temporización.

Estudio de los tiempos que intervienen en el camino de datos y en el reloj.

Tema 5: Planificación de la alimentación.

Dominios de alimentación. Circuitos mixtos analógicos y digitales.

Tema 6: Planificación del Layout.

Áreas en el layout. Diseño de I/O.

Tema 7: Introducción a la síntesis lógica.

Descripción del diseño, esquemas y lenguajes, VHDL y Verilog.

Tema 8: Simulación y bancos de pruebas.

Preparación del circuito para simulación. Diseño y especificación de los bancos de pruebas.

LABORATORIO (30h)

El laboratorio supone un total de 3 ECTS distribuidos en 10 prácticas de 0,3 ECTS cada una.

Práctica 1: Puerta inversora CMOS, análisis estático y dinámico.

En esta sesión se introduce el uso de las herramientas de CAD para el diseño de circuitos digitales CMOS. Se diseña una puerta inversora CMOS y se estudian sus características mediante la simulación eléctrica Spice. Se estudian las características estáticas y dinámicas.

Práctica 2: Bloques combinacionales, análisis del camino de datos.

En esta sesión se diseñan algunas puertas con las que se crean circuitos básicos para estudiar los efectos en las características dinámicas de las cargas, así como los retrasos introducidos por el número de etapas y la carga.

Práctica 3: Bloques secuenciales, simulación de la frecuencia máxima.

En esta sesión se diseña un circuito secuencial simple y se estudia la frecuencia máxima de funcionamiento según diferentes circunstancias.

Práctica 4: Introducción al layout full-custom.

En esta sesión se estudia el funcionamiento de la herramienta de layout mediante la creación de un circuito combinacional básico. Se introducen los chequeos DRC y LVS.

Práctica 5: Diseño mixto analógico/digital.

En esta sesión se diseña y analiza un circuito mixto analógico/digital, prestando especial atención a los dominios de alimentación.

Práctica 6: Extracción de propiedades y simulación post-layout.

En esta sesión se implementa un circuito digital, se extraen sus propiedades a partir del layout y se simula, comparando los resultados pre y post layout.

Práctica 7: Anillo I/O.

En esta sesión se diseña el anillo de entrada/salida del chip con parte analógica y digital y separación de los dominios de alimentación.

Práctica 8: Análisis de los elementos básicos de una biblioteca digital.

En esta sesión se analiza un circuito digital construido a partir de elementos de una biblioteca digital.

Práctica 9: Simulación digital

En esta sesión se crea un circuito a partir de los elementos digitales de una biblioteca y se simula con un simulador digital.

Práctica 10: Introducción a la síntesis lógica.

En esta sesión se diseña un circuito sencillo utilizando un lenguaje de descripción hardware, se sintetiza generando el layout, se comprueba que coinciden layout y descripción y se simula.

---

## Bloque I Test (15h)

Teoría (6 h)

Introducción

Tipos de errores

Tipos de tests

Hardware

Instrumentación

Análisis de resultados

Prácticas (9h)

1. Latch up en transistores / Caracterización térmica
2. Automatización de medidas
3. Análisis de datos (R/Matlab)

## Bloque II Verificación digital (45h)

Introducción a Verificación Digital (6 h)

Verificación Digital (DV): contexto dentro del flujo de diseño de un chip.

Objetivos. Métricas de verificación. Coverage.

Metodología de trabajo: vPlan, regresiones, bug reporting y bug tracking. Tests dirigidos, tests randomizados, inyección de errores, stress.

Breve introducción a otras disciplinas en DV: verificación formal, emulación, DMS y AMS.

Testeando el test.

SystemVerilog para DV (9 h)

Verilog y SystemVerilog.

Interfaces. Conectando DUT y testbench. Glue logic.

Tipos de datos. Clases y relación entre las clases, patrones de software.

Tareas y funciones.

Randomización y constraints. Estrategias de generación de estímulos.

Hilos y comunicación entre procesos. Eventos, semáforos, mailboxes.

Assertions.

Definición de covergroups y coverpoints. Cross coverage. Sampling.

Breve introducción a DPI.

Ejemplos basados en casos reales.

Introducción a UVM (15 h)

Conceptos básicos de UVM (Universal Verification Methodology). Qué es una librería, un framework y por qué usarlo.

Modularización y reusabilidad.

Tipos de componentes: monitor, driver, agente, environment, scoreboard, testcase.

Arquitectura del testbench. Interacciones entre los componentes: llamadas y agregación. Transacciones, secuencias y uso de puertos.

Representación de registros. RAL.

Prácticas (15 h)

Práctica 1: vPlan.

Verificación de un DUT con SystemVerilog y clases.

Familiarización con las principales herramientas: compilador, simulador, visor de gráficas.

Práctica 2 (\*2 sesiones): verificación de un DUT con UVM.

Generación de un testbench con todos los componentes.

Generación de testcases dirigidos.

Práctica 3: randomización y assertions con UVM.

Generación de testcases randomizados. Constraints.Regresiones.

Práctica 4: Coverage.

Creación de covergroups y sampleo.

Análisis de métricas.

Familiarización con las herramientas de recolección de coverage y análisis de métricas.

Consecución de 100% de funcional coverage y code coverage.

---

## TEORIA (30h)

Tema 1: Introducción al diseño de sistemas embebidos.  
Tema 2: Arquitectura de los sistemas embebidos.  
Tema 3: Comunicación procesador con lógica programable.  
Tema 4: Creación periféricos usuario.  
Tema 5: Entorno de desarrollo software.  
Tema 6: Desarrollo y depuración software.  
Tema 7: Revisión del diseño de sistemas embebidos en tiempo real.  
Tema 8: Arquitectura avanzada de un sistema embebido.  
Tema 9: Depuración del sistema embebido (HW/SW) utilizando Logic Analyzer.  
Tema 10: Interfaces de memoria en un sistema embebido.  
Tema 11: Manejo de interrupciones en sistemas en tiempo real.  
Tema 12: Estudio de baja latencia y alto ancho de banda.  
Tema 13: Configuración del procesador y creación de un Bootloader del sistema.  
Tema 14: Estudio del profiling y de la optimización de rendimiento de un sistema embebido.

#### LABORATORIO (30h)

Laboratorio 1: Diseño hardware de un sistema embebido básico.  
Laboratorio 2: Añadiendo IPs en la lógica programable.  
Laboratorio 3: Creando y añadiendo periféricos propios.  
Laboratorio 4: Escribiendo aplicaciones software básicas.  
Laboratorio 5: Depuración de software utilizando SDK.  
Laboratorio 6: Creación de un sistema embebido completo.  
Laboratorio 7: Depuración software/hardware usando Logic Analyzer.  
Laboratorio 8: Extendiendo el espacio de memoria con BRAM.  
Laboratorio 9: Acceso directo a memoria utilizando CDMA.  
Laboratorio 10: Creación de un Bootloader del sistema embebido.  
Laboratorio 11: Profiling y optimización de rendimiento en sistemas embebidos.

---

Los contenidos de la materia serán diferentes dependiendo de la práctica concreta que se deba llevar a cabo. A continuación se relacionan de modo genérico las posibles actividades que pueden realizarse durante las prácticas externas:

- Diseño microelectrónico analógico o mixto
- Diseño microelectrónico digital
- Test y verificación
- Sistemas embebidos. Software y hardware
- Diseño de layout
- Diseño de sistemas electrónicos
- Procesado digital de señal en sistemas VLSI
- ...

---

Los contenidos del Trabajo Fin de Master serán diferentes dependiendo de los objetivos concretos del proyecto a realizar. Pueden ser objeto de tema de Trabajo Fin de Master todos aquellos que sean propios de los estudios del Master. En particular, se podrán proyectar toda clase de sistemas y dispositivos microelectrónicos por cuantos procedimientos permita realizar la ingeniería actual. También podrá ser objeto del Trabajo Fin de Master los trabajos de investigación y desarrollo, y el modelado teórico o numérico de los dispositivos, circuitos o sistemas microelectrónicos. Se podrán considerar asimismo como temas de Trabajo Fin de Master los estudios relacionados con los contenidos de la Titulación y relativos a equipos, fábricas, instalaciones, servicios o su planificación, gestión o explotación. Por tanto los contenidos de la materia serán diferentes dependiendo del trabajo fin de máster concreto que se haya seleccionado por el alumno.

---

Aunque la lista de seminarios será dinámica, se proponen, en esta primera edición, los siguientes títulos:

Fully Integrated Frequency Synthesizers: PLLs for Modern Wireless Communications Systems  
WiFi7 Physical layer transceiver design. An overview  
Clock Distribution for Modern RF ICs: an overview  
SW/HW codesign: FW architectures and development process during System On Chip design  
Digital verification: practical use case  
Business aspects in IC design  
Verification  
Medical applications  
Industrial applications  
Measurements

---

#### TEORIA (12h)

1. Revisión de estructuras básicas de circuitos
2. Técnicas de control aplicadas en diseño microelectrónico analógico
3. Ruido
4. Técnicas de layout
5. Ejemplos prácticos de proyectos: referencias de voltaje
6. Ejemplos prácticos de proyectos: reguladores de voltaje

#### LABORATORIO (18h)

1. Proyecto practico I: Diseño y layout de una referencia de voltaje (parte 1/3)
2. Proyecto practico I: Diseño y layout de una referencia de voltaje (parte 2/3)
3. Proyecto practico I: Diseño y layout de una referencia de voltaje (parte 3/3)
4. Proyecto practico II: Diseño y layout de un regulador de voltaje (parte 1/3)
5. Proyecto practico II: Diseño y layout de un regulador de voltaje (parte 2/3)
6. Proyecto practico II: Diseño y layout de un regulador de voltaje (parte 3/3)

---

#### TEORIA (20h)

1. Introducción a sistemas de comunicación.
2. Componentes para desarrollo de bloques de radio frecuencia.
  - Extensión modelos RF
  - Mecanismos de degradación y reliability
  - Líneas de transmisión integradas
  - encapsulado
  - Bobinas integradas y extensión de RF de componentes pasivos
3. Diseño amplificadores RF.
  - Introducción parámetros S
  - Criterios estabilidad
  - Topologías y amplificadores ganancia programable.
4. Amplificadores de bajo ruido
  - topologías
  - técnicas cancelación ruido
5. Amplificadores de Potencia
  - topologías
  - mecanismos de degradación, SOA y electro migración
6. Mezcladores:
  - activos vs pasivos
  - técnicas mejora linealidad
7. Osciladores
  - osciladores de anillo.
  - osciladores LC.
  - Osciladores Colpits.
  - Osciladores controlados digitalmente.

#### LABORATORIO (10h)

- P1. Diseño y simulación líneas de transmisión en inductancias
- P3. Selección punto de polarización de un transistor y Layout
- P4. Diseño Amplificador bajo ruido
- P5. Diseño y simulación de un Mezclador.
- P5. Diseño y Simulación de un VCO

---

#### TEORIA

- Tema 1: Circuitos de capacidades conmutadas (4h)  
Tema 2: Técnicas de compensación de offset, ruido de baja frecuencia y desapareamiento (4h)  
Tema 3: Convertidores de Nyquist (4h)  
Tema 4: Convertidores de sobremuestreo (4h)  
Tema 5: Sensores de temperatura integrados. (4h)  
Tema 6: Frontends para sensores capacitivos, resistivos o inductivos (4h)

#### LABORATORIO

- Práctica 1: técnicas de simulación de circuitos en tiempo discreto (1.5h)  
Práctica 2: modelado de comportamiento de moduladores sigma-delta (1.5h)  
Práctica 3: diseño y simulación de circuitos SC para convertidores Sigma-Delta (1.5h)  
Práctica 4: diseño y simulación de un sensor de temperatura (1.5h)

---

#### TEORIA (15h)

- Diseño Digital con System Verilog (8h)  
Codificación de diseños para síntesis. (2h)  
Codificación de Máquinas de Estado Finitas. (1h)  
Generación, procesado y distribución de relojes y resets (1.5h)  
Sincronización de datos entre dominios de reloj y dominios de reset (1.5h)  
Diseño de Bajo Consumo con UPF. Clock Gating, Power Gating, DVFS. (2h)  
Implementación Digital VLSI (7h)  
Introducción al Flujo de Implementación Digital (0.5h)  
Biblioteca de Celdas Digitales en Procesos de Fabricación Avanzados (0.5)  
Definición de Restricciones Temporales (1h)  
Síntesis (1h)  
Diseño para Test (1h)  
Emplazamiento y Rutado (1h)

Análisis Temporal Estático (1h)  
Análisis de Consumo (1h)

#### LABORATORIO (15h)

Proyecto practico 1: Diseño RTL (Verilog) (7h)  
Codificación de diseños para síntesis  
Codificación de Máquinas de Estado Finitas  
Generación, procesado y distribución de relojes y resets  
Sincronización de datos entre dominios de reloj y dominios de reset  
Proyecto practico 2: implementación (6h)  
Síntesis  
Diseño para Test  
Emplazamiento y Rutado  
Proyecto practico 3: análisis (2h)  
Análisis Temporal Estático  
Análisis de Consumo

---

#### TEORIA (18h)

Tema 1: Introducción a funciones principales de un sistema de comunicaciones  
Tema 2: Adquisición de datos  
Tema 3: Cuantificación  
Tema 4: Filtros, interpoladores/diezmadadores  
Tema 5: Modulación/Demodulación  
Tema 6: Optimización de Velocidad, Área y Consumo

#### LABORATORIO (12h)

Laboratorio 1: Arquitectura de un filtro FIR  
Laboratorio 2: Implementación de un filtro FIR  
Laboratorio 3: Simulación de un filtro FIR

---

#### TEORÍA (18h)

Tema 1: Introducción (2h)  
- Diferencias entre uC/Cpu/Core  
- Principales fabricantes de CPU del mercado  
- Profundizar en la introducción del ARM M4-Cortex como Core de Referencia para el curso  
Tema 2: Cortex-M4 core (4h)  
- Características del core  
- Modelo de memoria  
- Registros de proposito general  
- Stacks  
- Niveles de acceso y modos de programación  
- Excepciones  
- Vector table  
- Fault handling  
- Instrumentation Trace Macrocell (ITM)  
- AHB Access Port (AHB-AP)  
- Bus Matrix  
Tema 3: Perifericos del ARM M4-Cortex (4h)  
- Nested Vectored Interrupt Controller (NVIC)  
- System Controll Block  
- System timer  
- Memory Protection Unit (MPU)  
- Floating-point unit  
Tema 4: Herramientas para programar un ARM M4-Cortex (3h)  
- Proceso de compilacion  
- Toolchain  
- Makefile  
- Startup file  
- Linker script  
Tema 5: Integración de una CPU en diferentes microcontroladores (2h)  
- ARM M4 por stm32f4 y texas  
- ARM M0 por raspberry pico y stm32m0  
- Otros ejemplos...  
Tema 6: Interaccion del ARM M4-Cortex con los masters y esclavos del Stm32F4 (3h)  
- Arquitectura del sistem de un Stm32F4  
- Organizacion de la memoria  
- Mapa de memoria

#### LABORATORIO (12h)

- LAB1: Interacción con los registros de proposito general y de las configuraciones basicas
  - LAB2: Cambios de contexto para Irq y Excepciones Vs Interacción funciones caller/callee
  - LAB3: Creación de un Scheduler
  - LAB4: Creación Startup file + linker script
  - LAB5: Migrar todo lo realizado hasta ahora a la toolchain creando un makefile
  - LAB6: Analisis de consumo de memoria
- 

#### Tema 1: Introducción a los sistemas embebidos (1h)

- Conceptos básicos de los sistemas embebidos. Características diferenciales
- Tipos de sistemas embebidos
- Aplicaciones de los sistemas embebidos
- Seguridad de los sistemas embebidos

#### Tema 2: Lenguajes de programación para sistemas embebidos (1h)

- Lenguajes de bajo nivel para sistemas embebidos. Ensamblador
- Lenguajes de alto nivel para sistemas embebidos. C y herramientas de compilación.
- Lenguajes de scripting útiles
- Interfaz entre diferentes lenguajes de programación

#### Tema 3: Arquitecturas SW/HW para sistemas embebidos (1h)

- CPUs
- Memorias
- Hosted/Hostless
- Flash/Flashless
- SDK y API de clientes
- VKs

#### Tema 4: Desarrollo de software para sistemas embebidos (1h)

- Ciclo de vida del desarrollo de software para sistemas embebidos
- Metodologías agiles de desarrollo de software para sistemas embebidos
- Herramientas de desarrollo de software para sistemas embebidos

#### Tema 5: Descripción de características deseables del flujo de desarrollo (2h)

- Reduccion del Time-to-market
- Definición de requisitos
- Funcionales
- Temporales
- Coste
- Etc.
- Arquitectura de Sistema
- Co-diseño HW/SW
- Plataformas de pruebas HW/SW
- Simulaciones
- Emuladores
- FPGAs
- Definición de interfaces HW/SW
- HW drivers
- Proceso de bringup
- Test Driven Development (TDD)
- Control de versiones (SCM)
- Gestión de tareas y errores en proyectos (Agile + Jira)
- Tests de Sistema
- Sistemas de Integración Continua
- Test benches
- Documentación

#### Tema 6: Diseño de software para sistemas embebidos (3h)

- SW product line: HW and SW configurations
- Arquitectura SW de capas, componentes e interfaces
- Diseño para el re-uso
- Capas de abstracción, HAL, OSAL.
- Código independiente de aplicación
- Código dependiente de aplicación
- Componentes de terceros y cuestiones legales
- Diseño para compatibilidad hacia atrás
- Diseño escalable
- Uso extensivo de tecnicas de programacion defensiva (assert)
- Chequeo de hard/soft deadlines
- Chequeo de problemas con la memoria (overwrites, stack overflows, etc)
- Sistemas operativos de tiempo real (RTOS)
- Configuración
- Threads y prioridades
- Interrupciones
- Timers
- Stacks
- Primitivas de comunicacion
- Aplicaciones multiprocesador
- Utilidades de debug y analisis del rendimiento

- Problemas recurrentes:
- Thread preemption
- Tiempo de respuesta a interrupciones
- Inversión de prioridades
- Tipos de Componentes
- HOST SW: drivers, apps, libs
- Firmware: dev, prod, BIST, loader
- Herramientas y scripts
- Interface públicos / privados

#### Tema 7: Funcionalidades usuales en sistemas embebidos (2h)

- Configuración específica del producto
- Producción
- Remota
- SW upgrades
- SDK para extender/cambiar funcionalidad
- Flash FS
- Interfaces
- JTAG
- UART
- SPI
- Consola de debug/operación
- Watchdog
- Herramientas de debug
- Memoria dinámica

#### Tema 8: Optimización de SW en Sistemas Embebidos (2h)

- Memoria vs Rendimiento
- Coste vs Facilidad en el desarrollo
- Requisitos HW
- Optimización de Memoria
- Compactación de estructuras de datos
- Datos en memorias compartidas
- Asignación de memoria (linker script)
- Reutilización de memoria
- Implementación de subsets de librerías (matemáticas, libc)
- Optimización de rendimiento
- Aritmética de punto fijo
- Uso de ensamblador.
- Re-escritura de código para uso de HW específico (DSP)
- DMA

#### Tema 9: Análisis de fallos en Sistemas Embebidos (2h)

- Requisitos HW
- JTAG
- Unit Tests
- Logs (serial, ethernet, files).
- Debug Buffers
- CPU Trace buffer
- Memory dumps
- Análisis de problemas en tiempo. Profiling
- Parseo y visualización de datos.
- Debug en sistemas multiprocesador

#### LABORATORIO:

La duración de cada laboratorio será de 1,5h:

##### Laboratorio 1: Diseño de la arquitectura de un sistema embebido

- Requisitos
- Arquitectura y codiseño HW/SW
- Ventajas y desventajas de las distintas opciones en cuanto a coste, facilidad de desarrollo, rendimiento, etc.

##### Laboratorio 2: Máquina virtualizada para desarrollo

- Virtualización de la plataforma de diseño (Docker)
- Herramientas de desarrollo (GNU)
- Sistema de control de versiones (GIT)
- Unit tests (Google Test)
- Análisis Estático de Código (CppCheck)
- Cobertura de testeo (Gcov)
- Análisis dinámico de Código (Valgrind/Electric Fence)

##### Laboratorio 3: Implementación de plataforma SW independiente de la aplicación sobre un simulador de RTOS

##### Laboratorio 4: Implementación de utilidades para debug (I)

##### Laboratorio 5: Implementación de utilidades para debug (I)

##### Laboratorio 6: Implementación de Mock para simular un HW específico

##### Laboratorio 7: Implementación de una aplicación real time ejemplo sobre la plataforma anterior (I)

Laboratorio 8: Implementación de una aplicación real time ejemplo sobre la plataforma anterior (II)

Laboratorio 9: Debug funcional de la aplicación

Laboratorio 10: Análisis de rendimiento y uso de memoria

---

## TEORIA (15h)

Tema 0: Conceptos de validación y aseguramiento de la calidad (QA)

- Modelo en V
- Requisitos
- Pruebas
- Depuración y defectos
- Conceptos de CI y de CD (Continuous Integration & Continuous Delivery)

Tema 1: Definición de requisitos

- Requisitos hardware
- Requisitos software
- Requisitos de sistema
- Herramientas de gestión de requisitos

Tema 2: Definición de prueba a partir de requisitos

- Herramientas existentes vs propias
- Tipos de pruebas (caja negra, caja blanca, funcionales, no funcionales...)
- Definición de indicadores clave de rendimiento (KPI) a partir de requisitos
- Definición de criterios de aceptación ¿pass/fail¿

Tema 3: Introducción a la validación de sistemas embebidos

- Validación de sistemas embebidos vs sistemas software:
- Disponibilidad de recursos
- Tiempos de ejecución
- Posibilidad de automatización
- Definición de sistema bajo prueba
- Interacción con el sistema bajo prueba

Tema 4: Ejecución y automatización de pruebas

- Definición de bancos de prueba
- Pruebas automáticas vs manuales: Por qué automatizar
- Automatización:
- Codificación: Control de cambios
- Instrumentación
- Ejecución
- Informes
- Herramientas:
- Lenguajes de programación para pruebas automáticos
- Herramientas para registrar progreso de pruebas
- Frameworks de prueba
- Recolección organizada de datos para informe de defectos

Tema 5: Defectos

- Qué es un defecto y cómo identificarlos
- Cómo reportar un defecto correctamente
- Herramientas de informe y trazabilidad de defectos

Tema 6: CI para productos embebidos

- Qué ofrece una herramienta de CI en la validación de productos embebidos:
- Definición de trabajos fácilmente repetibles
- Distribución de los trabajos entre hardware disponible
- Ventajas de tener un CI automatizado:
- Optimización de tiempo maquina
- Distribución de recursos
- Informes de resultados automáticos
- Gestión de bancos de prueba:
- Coexistencia entre automatización y uso manual de recursos
- Interacción de los componentes de un entorno de CI- Herramientas habituales de CI:
- Jenkins, Teamcity, Jira Workflow

Tema 7: Obtención y análisis de indicadores clave de rendimiento (KPI)

- KPI relacionados con el producto:
- Rendimiento
- Estabilidad
- Repetibilidad
- KPI relacionados con el entorno de CI:
- Cobertura de requisitos
- Tiempo entre detección de defecto y el arreglo
- Uso de recursos

## LABORATORIO (15h)

Laboratorio 1: Definir requisitos a partir de una descripción breve de un producto.

Laboratorio 2: Definición de pruebas a partir de requisitos. Plan de prueba.

Laboratorio 3: Ejecución manual de un plan de prueba. Pruebas exploratorias.

Laboratorio 4: Automatización de las pruebas definidas.

Laboratorio 5: Identificación y reporte de defectos a partir de los resultados de las pruebas.

Laboratorio 6: Creación de un entorno CI completo.

Laboratorio 7: Definición de KPI a partir de requisitos y resultados de las pruebas.

---

## TEORIA (10h)

Conceptos generales de arquitectura de computadores

- La arquitectura (ISA: Instruction Set Architecture)

Diferencias entre CISC y RISC

- La CPU (Unidad Central de Proceso)

Unidad de control

ALU (Unidad Aritmético-Lógica)

Buses

Cache de instrucciones

Pipelining

Registros

- Cores

- Memoria (ROM, RAM)

Memoria caché

- Periféricos de entrada/salida

Interrupciones

- Microcontroladores

Conceptos generales de software

- Lenguajes de programación

Relación entre juego de instrucciones, código objeto y lenguaje ensamblador

- Tipos de archivo ejecutable

- Compiladores

- El linker o enlazador

Linker scripts y scatter files

- Memoria estática y dinámica

El stack y el heap

- El cargador de arranque o bootloader

Tiempo real y conceptos de RTOS

- RTOS vs GPOS

Latency

- RTOS vs ¿bare-metal¿ (¿super loop¿)

- Kernel / Scheduler

preemptive time-slicing

cooperative time-slicing

Tick, Idle task, Ticless idle

- Interrupciones hardware y software

- Tareas e hilos

thread stack

prioridades

o Paralelismo y concurrencia

Sección Crítica

Semáforos, mutex y operaciones atómicas

Colas de mensajes o otros mecanismos

Condition variables

Problemas clásicos

productor/consumidor

condición de carrera

inversión de prioridades

- Procesadores multi-core

SMP (Symetric Multi-Processing)

AMP (Asymmetric Multi-Processing)

- Comunicación entre tareas

Análisis de los RTOS más comúnmente usados

- FreeRTOS

- MicroC/OS-II (uCOS)

- ThreadX

- RTEMS

- Zephyr

- VxWorks

RTOS en FPGAs

- Procesadores softcore. Ejemplos

Xilinx Microblaze

Tensilica Xtensa

Implementaciones RISC-V (Mi-V RV32, NEORV32, FEMTORV32)

## LABORATORIO (20h)

Proyecto de un pequeño sistema controlado por RTOS:

Control de un ventilador accionado por motor DC (PWM) de forma automática mediante sensor de temperatura y controlable mediante comandos por Ethernet.

Los contenidos del "Proyecto Industrial en Microelectrónica" serán diferentes dependiendo de los objetivos concretos del proyecto a realizar. Pueden ser objeto de tema de aquellos que sean propios de los estudios del título. En particular, se podrán proyectar toda clase de sistemas y dispositivos microelectrónicos por cuantos procedimientos permita realizar la ingeniería actual. También podrá ser objeto del Proyecto Industrial en Microelectrónica los trabajos de investigación y desarrollo, y el modelado teórico o numérico de los dispositivos, circuitos o sistemas microelectrónicos. Se podrán considerar asimismo los estudios relacionados con los contenidos del título relativos a equipos, fábricas, instalaciones, servicios o su planificación, gestión o explotación.

## PROFESSORAT

### **Alejandro Acuña Muñoz**

Ingeniero de Diseño Analógico. Maxlinear

### **María Teresa Bacete Castelló**

Site Director. Maxlinear

### **José Antonio Boluda Grau**

Profesor/a Titular de Universidad. Departament d'Informàtica. Universitat de València

### **Javier Calpe Maravilla**

Profesor/a Titular de Universidad. Departament d'Enginyeria Electrònica. Universitat de València

### **Miguel Chanca Martín**

IC Lead. Robert Bosch

### **Fausto Codina Ferrús**

Senior Manager Layout. Analog Devices

### **Enrique Company Bosch**

Analog Design Manager. Analog Devices

### **Pablo Cruz Dato**

Digital Architect. Bosch.

### **Francisco Escuder Roberto**

Responsable de software. MaxLinear Hispania, S.L.

### **Luis Alfonso Espinosa Ortega**

Ingeniero de Firmware. Analog Devices, S.L.U

### **José Manuel García González**

Senior Principal design engineer. Ams-OSRAM

### **José Luis García Navas**

Ingeniero de calidad de software. MaxLinear Hispania, S.L.

### **Raimundo García Olcina**

Profesor/a Titular de Universidad. Departament d'Enginyeria Electrònica. Universitat de València

### **Marcos Hervás García**

Ingeniero de Diseño Digital. MaxLinear Hispania, S.L.

### **Francisco Javier Jiménez Marquina**

Director de Ingeniería. MaxLinear

### **Jose Rafael Lajara Vizcaino**

Profesor/a Asociado de Universidad. Departament d'Enginyeria Electrònica. Universitat de València

### **Enrique Llorens Bufort**

Ingeniero de Diseño Digital. MaxLinear Hispania, S.L.

### **José Francisco Martí Martín**

Software/Firmware Design Engineer. Ams-OSRAM

### **Fernando Pardo Carpio**

Catedrático/a de Universidad. Departament d'Informàtica. Universitat de València

---

**Joaquin Perez Soler**

Ayudante/a Doctor/a. Departament d'Enginyeria Electrònica. Universitat de València

---

**Sebastien Poirier**

Principal engineer. Ams OSRAM

---

**Ricardo Pureza Coimbra**

Principal Analog Design Engineer. Analog Devices

---

**Abilio Candido Reig Escriva**

Profesor/a Titular de Universidad. Departament d'Enginyeria Electrònica. Universitat de València

---

**Samuel Rodríguez Rodríguez**

Ingeniero de Verificación. MaxLinear

---

**Antonio Jesús Rubio Salcedo**

Ingeniero de Verificación de Diseño. Analog Devices, S.L.U

---

**Raul Sahuquillo Valle**

Ingeniero de Verificación. MaxLinear

---

**Rubén Salvador Edo**

Diseñador Digital de Circuitos. Analog Devices, S.L.U

---

**Rafael Serrano-Gotarredona**

Director General. ams-OSRAM

---

**Jesús Soret Medel**

Profesor/a Titular de Universidad. Departament d'Enginyeria Electrònica. Universitat de València

---

**Issac Suarez Alvarez**

Profesor/a Titular de Universidad. Departament d'Enginyeria Electrònica. Universitat de València

---

**Riccardo Tonietto**

Analog IC Designer. Bosch

---

**José Gabriel Torres País**

Profesor/a Titular de Universidad. Departament d'Enginyeria Electrònica. Universitat de València

---

**Ramón Tortosa Navas**

Principal Engineer. Analog Devices, S.L.U

---

**Lucas Valentin García**

Algorithms and Machine Learning Engineer. Analog Devices, S.L.U

---

## OBJECTIUS

Les sortides professionals que té el curs són:

Les eixides professionals previstes estan estretament vinculades als perfils més demandats en aquest àmbit, entre els quals es podrien destacar:

Dissenyadors de sistema (arquitectura del xip, partició Hw/Sw, algorítmica, DSP...); desenvolupadors de flux de disseny digital; dissenyadors de processadors i memòries embegudes; dissenyadors analògics; dissenyadors RF i MMIC; enginyers de disseny físic (P&R); dissenyadors de dispositius (layout); enginyers d'estàndards; enginyers de microprogramari; enginyers de programari; enginyers de test; enginyers de qualitat i fiabilitat; enginyers d'embalatge (chiplet...); dissenyadors de sensors i MEMS; tecnòlegs; dissenyadors de dispositius optoelectrònics; dieñadores de PCBs; ...

En els últims temps, les empreses del sector de la microelectrònica i els semiconductors han vist com les seues ofertes de treball quedaven sovint vacants per falta de persones amb la formació demandada per a llocs tan específics. Per tant, l'objectiu fonamental d'aquest Màster és oferir a l'ecosistema VaSiC perfils professionals amb les competències necessàries per a ser directament incorporats en les seues plantilles.

## METODOLOGÍA

El centre responsable del Màster de Formació Permanent Multinacional en Microelectrònica és la Escola Tècnica Superior

d'Enginyeria (ETSE), i s'impartirà de manera presencial, en castellà, en horari de vesprades i dissabte. La matrícula serà gratuïta. Aquest Màster propi es desenvolupa en 60 ECTS, entre els quals s'inclouen 6 de pràctiques en empresa i 9 de TFM, tots ells a desenvolupar en projectes en empreses de VaSiC. La primera meitat del màster consistirà en cinc assignatures fonamentals, de 6 ECTS cadascuna, on es presentaran continguts de Dispositius electrònics i fotònics, Disseny CMOS analògic, Disseny CMOS digital, Test i verificació i Sistemes embeguts. Aquests continguts seran impartits principalment per professorat de la ETSE. En la segona part, juntament amb les PdE i el TFM, l'alumnat podrà triar entre tres itineraris: Disseny microelectrónico avançat analògic i mixt, Disseny digital avançat i Sistemes-en-Xip (SoC). Per a això, es podran triar cinc assignatures optatives de 3 ECTS cadascuna entre nou possibles, i una de Seminaris. Aquestes assignatures seran impartides en la seua pràctica totalitat per professorat especialista de les empreses de VaSiC. Totes les assignatures, tant les fonamentals com les optatives, inclouran continguts teòrics i continguts pràctics. L'alumnat disposarà d'ordinadors adequats amb les eines utilitzades en l'estàndard industrial per al desenvolupament i anàlisi dels sistemes que es proposen: Cadence, Synopsys, desenvolupament ARM, Matlab... També disposaran de l'instrumental necessari per al test i caracterització dels dispositius que s'estudien. Es preveuen pràctiques en sala blanca.