



Ministerio de Cultura y Educación  
Universidad Nacional de San Luis  
Facultad de Ciencias Físico Matemáticas y Naturales  
Departamento: Electrónica  
Area: Electrónica

(Programa del año 2022)

### I - Oferta Académica

Materia	Carrera	Plan	Año	Período
PROCESADORES II	PROF.TECN.ELECT	005/09	2022	1° cuatrimestre
PROCESADORES II	TEC.UNIV.ELECT.	15/13-CD	2022	1° cuatrimestre

### II - Equipo Docente

Docente	Función	Cargo	Dedicación
MURDOCCA, ROBERTO MARTIN	Prof. Responsable	P.Adj Exc	40 Hs
NUÑEZ MANQUEZ, ALEJANDRO ENRIQUE	Responsable de Práctico	JTP Exc	40 Hs
CABALLERO, CLAUDIO NICOLAS	Auxiliar de Práctico	A.1ra Simp	10 Hs

### III - Características del Curso

Credito Horario Semanal				
Teórico/Práctico	Teóricas	Prácticas de Aula	Práct. de lab/ camp/ Resid/ PIP, etc.	Total
Hs	3 Hs	Hs	4 Hs	7 Hs

Tipificación	Periodo
B - Teoria con prácticas de aula y laboratorio	1° Cuatrimestre

Duración			
Desde	Hasta	Cantidad de Semanas	Cantidad de Horas
21/03/2022	24/06/2022	14	90

### IV - Fundamentación

Actualmente, en la industria, automóviles, sistemas de comunicaciones, electrónica de consumo, laboratorios y en general en la mayoría de los equipos electrónicos se tiene un microcontrolador embebido encargado de controlar el funcionamiento de dicho dispositivo o instrumento.

Por ello es importante que el estudiante conozca las arquitecturas actuales para implementación de sistemas embebidos como así también las herramientas de software para la implementación de los mismos.

### V - Objetivos / Resultados de Aprendizaje

Estudiar la arquitectura y programación de microprocesadores y microcontroladores actuales de 8, 16 y 32 bits. Brindar los conceptos necesarios para que el estudiante pueda enfrentar la programación de Sistemas Embebidos. Introducir a los estudiantes en los conceptos de programación para resolver problemas a través de lenguajes de programación estándar de alto nivel.

Además, se pretende desarrollar el pensamiento lógico y crítico en un contexto de trabajo colaborativo.

Mejorar la habilidad para escribir documentos técnicos.

## VI - Contenidos

### **Unidad 01: Tecnologías y Arquitecturas.**

Tecnologías de Sistemas Digitales. Lógica cableada, programable, hardware reconfigurable. Evolución de los sistemas de cómputo. Lógica programable: Microprocesadores y Microcontroladores. Clasificación por aplicación o propósito: GPP, Microcontrolador, DSP, ASIC, ASIP, GPU. Clasificación por arquitectura: RISC, CISC, Superescalar, VLIW, Reconfigurable, System on a Chip (SoC), Network on a Chip (NoC). Clasificación de Flynn: SISD, SIMD, MISD, MIMD. Clasificación por conjunto de instrucciones (ISA).

### **Unidad 02: Introducción a los Microcontroladores y Sistemas Embebidos.**

Microcontroladores de 8, 16 y 32 bits. Mercado actual de Microcontroladores. Características de los Microcontroladores de propósito general. Aplicaciones. Periféricos internos de los microcontroladores. Sistemas Embebidos. Definición. Áreas de aplicación de los Sistemas Embebidos. Ejemplos de sistemas embebidos. Software Embebido. Firmware. Lenguajes de programación utilizados. Modelos de software. Herramientas de desarrollo integrado. Programas de diseño y simulación electrónica. Descripción de placas de desarrollo para microcontroladores de gama baja, media y alta.

### **Unidad 03: Microcontroladores ARM Cortex.**

Arquitecturas ARM. Breve historia de ARM. Características generales. Mercado actual de Microcontroladores ARM. Diferencias entre las variantes de ARM Cortex A, R y M. Introducción al Cortex-M3/M4. Arquitectura. Registros. La Unidad Aritmética Lógica (ALU). Buses. Modos de funcionamiento. Mapa de Memoria. Secuencia de Reset. Pipelining. Modos de operación. Excepciones y controlador de interrupciones. Características del Timer SysTick. Mapa de memoria. Arquitectura de buses. Pipeline. Set de instrucciones.

### **Unidad 04: Herramientas de desarrollo.**

Conceptos básicos sobre compilado en alto nivel. Compilador C. Pre-procesador. Librerías. El Linker. Elementos del lenguaje C. Estructura de un programa en C. Reglas de programación. Cross compilador C para microcontroladores. Particularidades. Entorno de desarrollo integrado (IDE). C Embebido. Técnicas de depurado (Debugging). Herramientas para programación / Debugging. Software para la plataforma CIAA y EDU-CIAA. Herramientas para control de versiones y repositorios.

### **Unidad 05: Características del Microcontrolador LPC4337.**

El LPC4337 de NXP. Características generales. Conceptos básicos de GPIO. Configuración. Reloj. Fuentes y configuración del reloj, el concepto de PLL. Configuración del PLL. Interrupciones en Cortex-M. El NVIC. Configuración. Tabla de vectores. Manejo de interrupciones. Sincronización de E/S. Interrupciones de periféricos. Temporizadores y contadores, configuración y sus interrupciones. Temporizador SysTick. Modulación de ancho de pulso (PWM). Comunicación serie. Comunicación síncrona y asíncrona. Comunicación serie en el Cortex-M4. Interface a la PC usando terminal serie. Plataforma embebida CIAA y EDU-CIAA. Características. Aplicaciones. Placa de expansión (Poncho). Interface RS232 y USB.

### **Unidad 06: Interfaces con Dispositivos Externos.**

Conexión de dispositivos externos con el microcontrolador. Características eléctricas de los puertos de microcontroladores. Interfaces con dispositivos de entrada. Conexión de Interruptores y pulsadores. Interface con teclados: independientes y matriciales. Adaptación de señales de entrada. Opto aislación. Interfaces con dispositivos de salida. Interface con LEDs. Uso de Displays. Conexión a Display 7 segmentos. Display 7 segmentos multiplexado. Display LCD inteligente. Displays gráficos. Interfaces de potencia. Interface con circuitos de corriente alterna. Interfaces con reles. Interface con otros periféricos externos. Interface SPI síncrono. Aplicaciones. Interface I2C. Características. Configuración. Dispositivos I2C.

### **Unidad 07: Desarrollo de Sistemas Embebidos.**

Técnicas de diseño programación de sistemas embebidos. Etapas de desarrollo de un proyecto. Programación Bare Metal. El patrón Súper Lazo. Programación mediante interrupciones (background / foreground). Modularización del software. Bibliotecas estáticas. Modelo de capas de abstracción de hardware (HAL). Librería CMSIS. Librería LPCOpen. Uso de la HAL sAPI. Uso de modelos de software. Máquinas de estados finitos (FSM). Tablas y diagramas de estados. Implementación de FSM en C. Procesos. Threads. Procesos secuenciales. Procesos concurrentes. Diagrama de Estado (Statecharts). Herramientas para Modelado, Simulación y Validación de Modelos.

### **Unidad 08: Sistemas Operativos en Tiempo Real.**

Introducción a los Sistemas de Tiempo Real. Hard Real Time System. Soft Real Time System. Clasificación de Sistemas; reactivos y transformacionales. Procesamiento en tiempo real. Sistemas Operativos en Tiempo Real (RTOS). Componentes básicos de un RTOS. Multitarea cooperativa y preventiva. Caso de estudio: FreeRTOS. Aspectos generales. Tipos de tareas y sus implementaciones. Estados de las tareas. Sincronización entre tareas: uso de semáforos. Intercambio de datos entre tareas: uso de colas. Mecanismos de exclusión mutua en el acceso a recursos compartidos (Mutex). Problemas asociados: inversión de prioridades, deadlocks. Uso de interrupciones. Manejo dinámico de memoria.

### **Unidad 09: Otras plataformas de desarrollo.**

Breve descripción de plataformas actuales embebidas. Arduino. Plataformas compatibles con Arduino. ChipKit. STM32. STM32 Núcleo. Plataformas para IoT. El ESP8266. Características. El ESP-32. Características. Aplicaciones. Plataformas computacionales de alto desempeño: Raspberry Pi. Distintos modelos. Descripción y características.

## **VII - Plan de Trabajos Prácticos**

Laboratorio 01 - Herramientas de control de versiones y repositorios.

Laboratorio 02 - Herramientas de desarrollo para plataforma EDU-CIAA.

Laboratorio 03 - Programación en C Embebido. Uso de HAL.

Laboratorio 04 - Temporizadores e Interrupciones.

Laboratorio 05 - Modelos de programación (FSM).

Laboratorio 06 - Comunicación Serie.

Laboratorio 07 - Introducción a la Plataforma ESP-32.

Laboratorio 08 - Sistemas Operativos en Tiempo Real I.

Laboratorio 09 - Sistemas Operativos en Tiempo Real II.

## **VIII - Regimen de Aprobación**

Para obtener la regularidad en la materia y rendir el examen final como alumno regular será necesario:

- Haber aprobado la totalidad de los exámenes parciales. Cada examen parcial posee dos recuperaciones.
- Haber aprobado el 100% de las Prácticas de Laboratorio.
- Se podrán recuperar solo el 30% de las prácticas de laboratorio, no aprobadas durante el cuatrimestre.

No se aceptan alumnos que no estén en condiciones regulares.

La materia no podrá rendirse en forma libre.

## **IX - Bibliografía Básica**

[1] Joseph Yiu (2013). The Definitive Guide to ARM® Cortex®-M3 and Cortex®-M4 Processors , 3rd Ed. Elsevier.

[2] Product data sheet – NXP. UM10503 LPC43xx/LPC43Sxx ARM Cortex-M4/M0 multi-core microcontroller.

[3] Galeano Gustavo (2009). Programación de Sistemas Embebidos en C, México, Alfaomega Grupo Editor.

[4] G. Booch, J. Rumbaugh, I. Jacobson (2006). El Lenguaje Unificado de Modelado, 2ª Ed, Addison-Wesley.

[5] Qing L, Yao C. (2003). Real-Time Concepts for Embedded Systems. CMP Books.

[6] Nicolas Melot (2010). Study of an operating system: FreeRTOS.

[7] Richard Barry (2010). Using the FreeRTOS Real Time Kernel. Standard Edition.

## **X - Bibliografía Complementaria**

[1] Ganssle Jack, The Firmware Handbook – EEUU, ELSEVIER, 2004, 365p.

[2] Zurawski Richard (2006). Embedded Systems HandBook, Taylor & Francis Group.

[3] Keith E. Curtis (2006). Embedded multitasking with Small Microcontrollers. ELSEVIER.

[4] Ken Arnold (2000). Embedded Controller Hardware Design. LLH Technology Publishing.

[5] Tammy Noergaard (2005). Embedded Systems Architecture A Comprehensive Guide for Engineers and Programmers. ELSEVIER.

## **XI - Resumen de Objetivos**

Estudiar la arquitectura y programación de microprocesadores y microcontroladores de 8, 16 y 32 bits actuales.  
Utilizar herramientas de desarrollo para realizar la programación de una familia de microcontroladores.  
Dotar al alumno de los conocimientos necesarios para desarrollar un sistema embebido.

## **XII - Resumen del Programa**

Unidad 01: Tecnologías y Arquitecturas.  
Unidad 02: Introducción a los Microcontroladores y Sistemas Embebidos.  
Unidad 03: Microcontroladores Cortex de ARM.  
Unidad 04: Herramientas de desarrollo.  
Unidad 05: Características del Microcontrolador LPC4337  
Unidad 06: Interfaces con Dispositivos Externos.  
Unidad 07: Desarrollo de Sistemas Embebidos  
Unidad 08: Sistemas Operativos en Tiempo Real.  
Unidad 09: Otras plataformas de desarrollo.

## **XIII - Imprevistos**

El presente programa puede estar sujeto a cambios dada la situación epidemiológica por COVID-19.

## **XIV - Otros**