



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

(Programa del año 2020)  
 (Programa en trámite de aprobación)  
 (Presentado el 27/05/2020 12:19:15)

### I - Oferta Académica

Materia	Carrera	Plan	Año	Período
PROCESADORES II	ING.ELECT.O.S.D	010/05	2020	1° cuatrimestre
PROCESADORES II	ING.ELECT.O.S.D	13/08	2020	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
RETA, FACUNDO ESTANISLAO	Auxiliar de Práctico	A.2da Simp	10 Hs
ANDRADA TIVANI, ASTRI EDITH	Auxiliar de Laboratorio	A.1ra Semi	20 Hs

### III - Características del Curso

Credito Horario Semanal				
Teórico/Práctico	Teóricas	Prácticas de Aula	Práct. de lab/ camp/ Resid/ PIP, etc.	Total
Hs	2 Hs	Hs	3 Hs	5 Hs

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

Duración			
Desde	Hasta	Cantidad de Semanas	Cantidad de Horas
09/03/2020	19/06/2020	15	75

### IV - Fundamentación

En la industria, automóviles, sistemas de comunicaciones, electrónica de entretenimiento, laboratorios y en general en todos los equipos electrónicos actuales se tiene un microcontrolador embebido encargado de controlar el funcionamiento de dicho aparato o instrumento.

Este curso ofrece a los estudiantes la oportunidad de estudiar las características de un microprocesador / microcontrolador y aprender a desarrollar un sistema electrónico para resolver problemas del mundo real.

### V - Objetivos / Resultados de Aprendizaje

Estudiar la arquitectura y programación de microprocesadores y microcontroladores actuales de 8, 16 y 32 bits.  
 Utilizar herramientas de desarrollo para realizar el depurado y programación de los mismos en lenguajes de alto nivel.  
 Desarrollar los conocimientos y habilidades necesarios para diseñar un sistema embebido dedicado a una aplicación específica descrita mediante requisitos de diseño, tanto a nivel de conexionado eléctrico (Hardware) como de su programación (Software).  
 Mejorar la habilidad para escribir documentos relacionados con un desarrollo de ingeniería.

### 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**

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 Aritmético 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. Control de periféricos e interrupciones en lenguaje C. Entorno de desarrollo integrado. Técnicas de depurado (Debugging). Drivers y aplicaciones. Librerías de abstracción de Hardware. CMSIS. Otras librerías. Repositorios. GitHub. GitLab. Descripción.

## **Unidad 05: Periféricos internos del Cortex-M3**

Conceptos básicos de GPIO. Configuración. Reloj. Fuentes y configuración del reloj, el concepto de PLL. Configuración del PLL. Interrupciones en la arquitectura ARM 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).

## **Unidad 06: Interface 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 matriciales. Adaptación de señales de entrada. Opto aislación. Interfaces con dispositivos de salida. Interface con LEDs. 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.

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

Técnicas de diseño/programación de sistemas embebidos. Etapas de desarrollo de un proyecto. Sistemas en tiempo real. Hard Real Time System. Soft Real Time System. Clasificación de Sistemas; reactivos y transformacionales. Programación Bare Metal. El patrón Súper Lazo. Programación mediante interrupciones (background / foreground). Máquinas de estados finitos (FSM). Tablas y diagramas de estados. Implementación de FSM en C. Diagrama de Estado (Statecharts). Herramientas para Modelado, Simulación y Validación de Modelos. Uso de Drivers y librerías de abstracción de hardware.

## **Unidad 08: Comunicación serie**

Comunicación serie. Comunicación síncrona y asíncrona. Comunicación serie en el Cortex-M3. Las interfaces serie en la BaseBoard. Interface RS232, USB y UART 3.3V. Configuración. Interface a la PC usando terminal serie. Puertos serie virtuales. Interface SPI síncrono. Aplicaciones. I2C. Interfaces. Configuración. Dispositivos I2C. Transmisión y recepción por interrupciones.

## **Unidad 09: Sistemas Operativos en Tiempo Real**

Introducción a los Sistemas de Tiempo Real. 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 10: Otras plataformas de desarrollo**

Breve descripción de plataformas actuales embebidas. Intel Galileo. Arduino. ChipKit. STM32. STM32 Núcleo. Otras. Plataformas computacionales de alto desempeño: Raspberry Pi. Distintos modelos. Descripción y características. Software.

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

### **PRÁCTICAS DE LABORATORIO**

Laboratorio 01 - Introduccion a la IDE MCUXpresso.

Laboratorio 02 - Repaso de programacion en C.

Laboratorio 03 - Temporizadores, Interrupciones y libreria LPCOpen.

Laboratorio 04 - Modelos de Programacion.

Laboratorio 05 - Introduccion la plataforma Raspberry Pi.

## **VIII - Regimen de Aprobación**

La materia se aprueba con un exámen final o la realizacion de un proyecto integrador.

Para obtener la regularidad de la materia y poder rendir el examen final o realizar el proyecto integrador será necesario:

- Haber aprobado el 100% de los trabajos prácticos no presenciales.
- Haber aprobado el 100% de los trabajos prácticos de laboratorios que se realizarán de manera no presencial.
- Haber aprobado los dos parciales no presenciales.
- Para la aprobación de los trabajos prácticos no presenciales será necesario enviar los mismos, a través de la plataforma Classroom de la materia.

-Los alumnos tendrán derecho a dos recuperaciones de todos los exámenes parciales.

Los exámenes parciales y el exámen final se aprueban con al menos el 70% de las respuestas correctas.

La materia no podrá rendirse en forma libre.

## **IX - Bibliografía Básica**

- [1] The Definitive Guide to the ARM Cortex-M3 - Joseph Yiu, 2ª Edición, Newnes Elsevier Inc, 2010.
- [2] Cortex™-M3, Revision r2p0, Technical Reference Manual – ARM
- [3] ARM@v7-M Architecture, Reference Manual – ARM
- [4] UM10360 LPC176x/5x User Manual & LPC1769/68/67/66/65/64/63 Product data sheet – NXP
- [5] Galeano Gustavo, Programación de Sistemas Embebidos en C, México, Alfaomega Grupo Editor, 2009, 544p.
- [6] El Lenguaje Unificado de Modelado, G. Booch, J. Rumbaugh, I. Jacobson, 2ª Edición, Addison-Wesley, 2006
- [7] Barry R. Using the FreeRTOS Real Time Kernel. NXP LPC17xx Edition.
- [8] Qing L, Yao C. Real-Time Concepts for Embedded Systems. CMP Books 2003.
- [9] Study of an operating system: FreeRTOS - Nicolas Melot.

## X - Bibliografía Complementaria

- [1] Ganssle Jack, The Firmware Handbook – EEUU, ELSEVIER, 2004, 365p.  
[2] Zurawski Richard, Embedded Systems HandBook, Taylor & Francis Group, 2006, 1089p.  
[3] Keith E. Curtis, Embedded multitasking with Small Microcontrollers, ELSEVIER, 2006, 417p.  
[4] Ken Arnold, Embedded Controller Hardware Design, , LLH Technology Publishing, 2000, 244p.  
[5] Tammy Noergaard, Embedded Systems Architecture A Comprehensive Guide for Engineers and Programmers, , ELSEVIER, 2005, 657p.

## 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.  
Unidad 03: Microcontroladores ARM.  
Unidad 04: Herramientas de desarrollo.  
Unidad 05: Periféricos internos del Cortex-M3.  
Unidad 06: Interface con Dispositivos Externos.  
Unidad 07: Desarrollo de Sistemas Embebidos  
Unidad 08: Comunicación serie.  
Unidad 09: Sistemas Operativos en Tiempo Real.  
Unidad 10: Otras plataformas de desarrollo.

## XIII - Imprevistos

Cambio tecnológico: actualización por parte de fabricantes de microcontroladores.  
Incompatibilidades y bugs de nuevas versiones de software.  
Disponibilidad en el mercado local de partes y componentes necesarios para las prácticas.

## XIV - Otros

--

### ELEVACIÓN y APROBACIÓN DE ESTE PROGRAMA

ELEVACIÓN y APROBACIÓN DE ESTE PROGRAMA	
Profesor Responsable	
Firma:	
Aclaración:	
Fecha:	