



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

(Programa del año 2026)
(Programa en trámite de aprobación)
(Presentado el 19/03/2026 16:19:01)

I - Oferta Académica

Materia	Carrera	Plan	Año	Período
ELECTRÓNICA PROGRAMABLE II	ING.ELECT.O.S.D	0000 0000 0000 006	2026	1° cuatrimestre

II - Equipo Docente

Docente	Función	Cargo	Dedicación
HERNANDEZ VELAZQUEZ, SERGIO FE	Prof. Responsable	JTP Exc	40 Hs

III - Características del Curso

Credito Horario Semanal				
Teórico/Práctico	Teóricas	Prácticas de Aula	Práct. de lab/ camp/ Resid/ PIP, etc.	Total
6 Hs	Hs	Hs	Hs	6 Hs

Tipificación	Periodo
C - Teoría con prácticas de aula	1° Cuatrimestre

Duración			
Desde	Hasta	Cantidad de Semanas	Cantidad de Horas
11/03/2026	26/06/2026	15	90

IV - Fundamentación

El curso Electrónica Programable II tiene como objetivo profundizar las competencias en programación aplicada a sistemas embebidos, fundamentales en la electrónica moderna. Aborda programación en lenguaje C, sistemas operativos en tiempo real y lenguajes interpretados. De esta manera, el estudiante se inicia en el uso de software aplicado al funcionamiento de sistemas electrónicos.

V - Objetivos / Resultados de Aprendizaje

Competencias Generales:

- Desarrollar habilidades básicas de análisis y resolución de problemas relacionados con sistemas electrónicos y programación.
- Iniciar la práctica de razonamiento lógico y estructuración de algoritmos aplicables a sistemas embebidos.
- Fomentar la capacidad de trabajo autónomo y colaborativo durante las actividades de aulas.
- Manejar herramientas básicas de programación y depuración en entornos de desarrollo electrónico.

Competencias Específicas:

- Comprender y utilizar conceptos iniciales de programación en C: punteros, estructuras, gestión de memoria/archivos y

entradas/salidas.

- Conocer y aplicar principios básicos de depuración en circuito.
- Identificar y usar periféricos e interrupciones de manera elemental.
- Introducirse en conceptos de multitarea y comunicación entre tareas en sistemas embebidos.
- Familiarizarse con métodos computacionales simples y programación interpretada para el control de sistemas electrónicos.

Resultados de Aprendizaje:

- Capaz de programar aplicaciones simples en C que interactúan con hardware básico.
- Demuestra habilidad para probar y depurar programas simples.
- Aplica de forma inicial periféricos e interrupciones para controlar dispositivos electrónicos.
- Reconoce los conceptos de sistemas multitarea y comunicación entre tareas, aplicándolos en ejercicios básicos.
- Utiliza herramientas de cálculo numérico y programación interpretada para resolver problemas en sistemas electrónicos.

VI - Contenidos

Unidad 1

Introducción de lenguaje C en sistemas embebidos. Aritmética de punteros, punteros a estructuras y paso de parámetros por referencia para optimizar el uso de la pila. Declaración de structs y unions. Alineación de memoria y empaquetado de datos para protocolos de comunicación. Flujos estándar y funciones de formato avanzado para el procesamiento de tramas de datos.

Unidad 2

Reserva y liberación dinámica de memoria. Estrategias para evitar fugas de memoria y fragmentación en sistemas embebidos. Diferencias entre archivos de texto y binarios. Acceso secuencial y aleatorio mediante punteros de archivo.

Unidad 3

Microcontroladores y sus periféricos. Operaciones a nivel de bit. Concepto de interrupción de hardware y software. Manejo de periféricos básicos de sistemas embebidos.

Unidad 4

Concepto de multitarea en sistemas embebidos pequeños y el valor de usar un Kernel de Tiempo Real. Diferencia entre tareas y funciones convencionales: El bucle infinito y el prototipo de función de tarea. Creación de tareas y parámetros asociados. Estados de una tarea. Prioridades de tareas y el funcionamiento del planificador (Scheduler). Medición del tiempo y el Tick del sistema. Implementación de demoras no bloqueantes.

Unidad 5

Características de una cola: almacenamiento de datos de tamaño fijo y acceso por múltiples tareas. Funcionamiento FIFO (First-In-First-Out). Creación y configuración de colas. Envío de datos. Recepción y extracción de datos. Concepto de "Bloqueo por lectura" y "Bloqueo por escritura" cuando una cola está vacía o llena. Uso de colas para enviar estructuras de datos complejas entre hilos.

Unidad 6

Diferencias entre lenguajes compilados (C) e interpretados (Python). Sintaxis Básica: Variables, tipos de datos dinámicos y operadores. Estructuras de Control.

Unidad 7

Uso y diferencias entre listas y tuplas. Definición de funciones, paso de parámetros y retorno de valores. Importación de bibliotecas y organización de scripts.

Unidad 8

Uso del módulo math para funciones estándar. Representación de matrices mediante listas anidadas y fundamentos de bibliotecas de cálculo numérico. Implementación de algoritmos básicos para resolución de problemas de ingeniería y tratamiento de datos.

VII - Plan de Trabajos Prácticos

Los trabajos prácticos estarán distribuidos de la siguiente manera:

Trabajo Práctico N°01 – Punteros

Trabajo Práctico N°02 – Estructuras

Trabajo Práctico N°03 – Gestión de memoria

Trabajo Práctico N°04 – Periféricos e interrupciones

Trabajo Práctico N°05 – Manejo de archivos

Trabajo Práctico N°06 – Introducción a Sistemas Operativos en Tiempo Real

Trabajo Práctico N°07 – Hilos múltiples de ejecución y comunicación

Trabajo Práctico N°08 – Introducción a Python

Trabajo Práctico N°09 – Estructuras de datos y modularización en Python

Trabajo Práctico N°10 – Métodos numéricos con Python

VIII - Regimen de Aprobación

Para obtener la calificación de regular los estudiantes deberán:

- Contar con al menos el 90% de asistencia a clases.
- Aprobar el 100% de los Trabajos Prácticos con una calificación igual o superior al 70%.
- Aprobar dos evaluaciones con una calificación igual o superior al 70%.

La materia no es promocional y no se puede rendir en condición de libre.

IX - Bibliografía Básica

[1] Aprender a programar en C: de 0 a 99 en un solo libro, A. M. Vozmediano.

[2] Practical C Programming” - Steve Oualline, Third Edition, Ed. O’Reilly & Associates, 1997, ISBN: 1-56592-306-5.

[3] C/C++ Curso de programación, Fco. Javier Ceballos Sierra, 5ta. ed., Madrid, España: Ra-Ma Editorial, 2019. ISBN: 9788499648125.

[4] Introducción a la programación con Python, Andres Marzal, Isabel Gracia, Universitat Jaume I, 2014, Col·lecció Sapientia, 23. ISBN: 978-84-692-5869-9.

[5] Mastering the FreeRTOS™ Real Time Kernel A Hands-On Tutorial Guide Richard Barry -and- The FreeRTOS Team Release Version - 1.1.0.

X - Bibliografía Complementaria

XI - Resumen de Objetivos

- Introducir al estudiante en la programación básica y avanzada en C aplicada a sistemas embebidos.
- Desarrollar habilidades para probar y depurar programas.
- Familiarizar al estudiante con periféricos, interrupciones y entradas/salidas en sistemas embebidos.
- Presentar conceptos iniciales de multitarea y comunicación entre tareas en sistemas embebidos.
- Iniciar al estudiante en el uso de métodos computacionales y programación interpretada para resolver problemas elementales en electrónica.

XII - Resumen del Programa

XIII - Imprevistos

XIV - Otros

--

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