

# GUÍA ACADÉMICA DE LA ASGINATURA DE SISTEMAS EMBEBIDOS

PAUL D. ROSERO-MONTALVO

## 1. Introducción

La presente asignatura es la integración de la todos los conocimientos adquiridos por el estudiante en los anteriores semestres relacionados a electrónica, matemáticas y programación. Se basa en el eficiente desarrollo de sistemas embebidos que recopilan datos del entorno y generan un tipo de respuesta en relación a un análisis de datos. Se usan diferentes tipos de simuladores para realizar las pruebas de desarrollo correspondientes al software y hardware para su correcto funcionamiento.

## 2. Modelo de enseñanza

Se planteo un modelo presencial/virtual (blended learning) ya utiliza gran cantidad recursos virtuales de apoyo para el proceso de aprendizaje del estudiante. Para ello, se establecen los recursos de la asignatura:

- Diapositivas
- Clases magistrales
- Lecturas de apoyo
- Tutoriales
- Plataforma virtual

Esta asignatura cuenta con una plataforma de educación virtual y una página web como soporte al sistema académico de la universidad. En donde se encuentra disponible todo el material didáctico y las correspondientes actividades. Por esta razón, el estudiante tiene la flexibilidad de presentar los trabajos en cualquier momento del periodo académico divididos en las dos parciales de primer y segundo bimestre. Sin embargo, las reuniones virtuales y las evaluaciones parciales y finales se encuentran en fechas específicas para evitar el plagio de respuestas.

## 3. Modelo de aprendizaje

El modelo empleado es el aprendizaje basado en proyectos donde el alumno sea protagonista de su propio aprendizaje. Permite a los alumnos adquirir los conocimientos y competencias clave en el siglo XXI mediante la elaboración de proyectos que dan respuesta a problemas de la vida real. Los alumnos se convierten en protagonistas de su propio aprendizaje y desarrollan su autonomía y responsabilidad, ya que son ellos los encargados de planificar, estructurar el trabajo y elaborar el producto para resolver la cuestión planteada. La labor del docente es guiarlos y apoyarlos a lo largo del proceso.

## 4. Calificación

La presente forma de calificación se encuentra presente dentro del sílabo. No obstante, se explica en este apartado a detalle cada componente.

### 4.1. Lecciones (20 %)

Se tiene planificado realizar dos lecciones parciales sobre 10 puntos en cada bimestre. Estas lecciones son de programación y se las realizan en computador. Para ello, cuenta con 1h45 de duración.

## 4.2. Tareas (20 %)

Las tareas es el refuerzo de cada ítem del sílabo a aprender por el estudiante. Existen deberes semanales donde su programa debe ser subido al repositorio GitHub <https://github.com/>. Para ello, debe el estudiante crear una cuenta. En la plataforma de la universidad solo sube su enlace de acceso al repositorio para ser revisado por el docente.

## 4.3. Reportes de laboratorio (20 %)

Los reportes de laboratorio es el componente práctico de la asignatura. En donde se realizará de forma individual simulada con un exposición virtual hacia le técnico docente donde valorará mediante. El formato de informes se encuentra en la página web.

Cuadro 1: Rúbrica de evaluación de laboratorios

| LABORATORIOS         | Deficiente  | Necesita mejorar                    | Regular                                                    | Bueno                                                 | Muy bueno                                                 | Excelente                                                 |
|----------------------|-------------|-------------------------------------|------------------------------------------------------------|-------------------------------------------------------|-----------------------------------------------------------|-----------------------------------------------------------|
|                      | 0 puntos    | 2 puntos                            | 4 puntos                                                   | 6 puntos                                              | 8 puntos                                                  | 10 puntos                                                 |
| <b>Funcionalidad</b> | No presenta | No funciona                         | Funcionamiento regular                                     | Buen funcionamiento                                   | Funcionamiento muy bueno                                  | Excelente funcionamiento                                  |
| <b>Presentacion</b>  | No presenta | Presenta sin el formato establecido | Deficiente presentación de cálculos, diseño y conclusiones | Buena presentación de cálculos, diseño y conclusiones | Muy buena presentación de cálculos, diseño y conclusiones | Excelente presentación de cálculos, diseño y conclusiones |

## 4.4. Proyecto bimestral (20 %)

El proyecto bimestral consiste en un aplicación en condiciones reales que solucionen un problema planteado. El formato de presentación de proyecto se encuentra en la página web.

## 4.5. Exámenes (20 %)

El estudiante rendirá su examen resolviendo un caso de programación que abarque los conocimientos de la asignatura. Para ello, cuenta con 1h45 de duración.

# 5. Materiales

Los materiales están enfocados a los requerimientos físicos y lógicos necesario por cada estudiante y grupo de trabajo.

## 5.1. Software

- IDE de Arduino <https://www.arduino.cc/en/Main/Software>
- Latex y Texmaker <http://www.xmlmath.net/texmaker/download.html> , <https://miktex.org/download>
- Fritzing <http://fritzing.org/home/>
- SourceTree <https://www.sourcetreeapp.com/>
- Processing <https://processing.org/download/>
- R y R-studio <https://www.rstudio.com/>, <https://www.r-project.org/>
- Proteus
- Excel

## 5.2. Hardware

- ESP8266
- Protoboard
- Sensores (ver lista de proyectos)
- PC personal o por grupo

## 6. Contenidos

Una de las características principales de un sistema embebido debe ser la adaptación. Es decir, que sean capaces de emular algunas habilidades del procesado que realiza el cerebro humano. Por tal motivo, debe ser un sistema inteligente. El mismo que implica de alguna manera, la capacidad de tomar decisiones, de aprender de estímulos externos, adaptarse a los cambios o la posibilidad de ejecutar algoritmos matemáticos inteligentes. Implícitamente, se basa en un paradigma computacional que recibe o procesa datos para lograr una tarea encomendada. Bajo este concepto, se espera que los sensores puedan brindar la mejor información posible. Una de las formas de tomar decisiones a base de parámetros establecidos, son los algoritmos de aprendizaje supervisado. Los cuales necesitan, por un lado, un conjunto de datos de entrenamiento del modelo y por el otro, un conjunto de validación. Como se comentó en anterioridad, los datos recolectados por los sensores pueden tener muchos errores de lectura, no es recomendable almacenarlos todos.

### 6.1. Introducción

El primer capítulo se enfoca en los conceptos teóricos que establecen la historia, avances y su descripción específica del funcionamiento de un microcontrolador y su paso a ser un sistema micro-procesado.

### 6.2. Puertos digitales

El presente capítulo se basa en el manejo de los periféricos internos digitales de un microcontrolador de hardware y software libre.

- Puertos Entrada Salida
- Visuaizadores
- LABORATORIO DE UNIDAD

### 6.3. Módulos de Comunicación

En este capítulo se trabaja con los módulos de comunicación de un microcontrolador.

- Comunicación Serial
- Comunicación I2C
- LABORATORIO DE UNIDAD

### 6.4. Periféricos Especiales

Estos periféricos buscan dar una mejor funcionalidad al microcontrolador al hacerlo trabajar por etapas y con una conexión a sensores.

- Convesor Análogo-Digital Digital-Análogo
- Interrupciones
- Timers
- LABORATORIO DE UNIDAD:

### 6.5. Funciones Específicas

Estos módulos permiten tener un mejor rendimiento del microcontrolador en condiciones reales.

- WDT
- Ahorros de energía
- Comunicación WiFi
- Tareas en Tiempo Real (RTC)
- LABORATORIO DE UNIDAD:

## 6.6. Desarrollo de Sistemas Embebidos

Conceptos teórico prácticos para el desarrollo de sistemas embebidos (SE)

- Tipos de SE
- Tipos de Sensores
- Acoplamiento de Hardware y Software
- LABORATORIO DE UNIDAD:

## 6.7. Aprendizaje automático

Este capítulo final, presenta la implementación de algoritmos que puedan proporcionar la capacidad de tomar decisiones del sistema electrónico

- Introducción a Sistemas Inteligentes
- Algoritmo del Vecino más Cercano
- Regresión lineal simple
- Teorema de Bayes
- LABORATORIO DE UNIDAD

## 7. Compromisos

Los compromisos presentes por parte de los estudiantes es:

- Llegar a clases hasta 15 minutos máximo de cada periodo académico.
- Presentar puntualmente los deberes y laboratorios presentes en el aula virtual.
- Reconocer la autoría de sus trabajos presentados.
- Presentarse a las evaluaciones con el repaso de la asignatura.
- Reconocer su falta en caso de ser encontrado copiando.

Los compromisos presentes por parte de los docente es:

- Llegar a clases hasta 15 minutos máximo de cada periodo académico.
- Enviar con anticipación los deberes y laboratorios presentes en el aula virtual.
- Proporcionar un espacio de tutorías para retroalimentación de los estudiantes.

## 8. Contacto

**Fono:** 0969432370

**correo:** pdrosero@utn.edu.ec

**Página web:** <http://www.paulrosero-montalvo.com>