



Arquitectura de Computadoras y Ensambladores 1

<b>CÓDIGO:</b> 0778	<b>CRÉDITOS:</b> 5
<b>ESCUELA:</b> Ciencias y Sistemas	<b>ÁREA:</b> Ciencias de la Computación
<b>PREREQUISITO:</b> 0796 - 0964	<b>POSTREQUISITO:</b> 0281 - 0779 - 0970
<b>CATEGORÍA:</b> Obligatorio	<b>SECCIÓN:</b> A
<b>HORAS SEMANALES DEL CURSO:</b> 4	<b>HORAS SEMANALES DEL LABORATORIO:</b> 2
<b>DÍAS QUE SE IMPARTE EL CURSO:</b> Martes y Jueves	<b>DÍAS QUE SE IMPARTE EL LABORATORIO:</b> Sábado
<b>HORARIO DEL CURSO:</b> 10:40 - 12:20	<b>HORARIO DEL LABORATORIO:</b> 14:00 - 15:40

**DESCRIPCIÓN DEL LABORATORIO**

El laboratorio del curso de Arquitectura de Computadoras y Ensambladores 1, trata sobre la parte práctica del curso. Se encarga de la aplicación de electrónica digital haciendo uso de Microcontroladores y plataformas de desarrollo. Refuerza los conocimientos de electrónica digital y secuencial. Además de entender el funcionamiento de programas a bajo nivel, y manipular el uso de la memoria en los programas informáticos.

**OBJETIVOS**

- Evaluar los campos de aplicación que tiene ARM y como este tiene alcance en la electrónica a través de sistemas IoT, así como de procesadores para Inteligencia Artificial y el aumento de la gama de los dispositivos móviles.
- Desarrollar la capacidad analítica y lógica de resolver problemas utilizando el ensamblador ARM, siendo capaces de optimizar y depurar código ensamblador para lograr mejores resultados.
- A través de la estructura, funcionamiento y características comprender la arquitectura ARM y sus modos de operación y la utilidad de esta arquitectura en la actualidad.

**METODOLOGÍA**

1. Clases magistrales para guiar y asesorar al estudiante, fortaleciendo el conocimiento de los diferentes microcontroladores y plataformas de desarrollo, como Raspberry Pi
2. Autoaprendizaje y lectura acerca de las herramientas a utilizar.
3. Exámenes cortos, tareas, prácticas y proyecto.
4. Se realizarán proyectos para poder evaluar los conceptos adquiridos en clase sobre la arquitectura de computadoras.

**HABILIDADES**

- Conocimiento en programación de los lenguajes utilizados por los microcontroladores y plataformas de desarrollo.
- Comprensión de la importancia y aplicación del código intermedio en la construcción de software.
- Comprensión de los requerimientos que se les planteen en los enunciados a lo largo del curso.

**COMPETENCIAS**

- Interpretar parámetros utilizados en robótica.
- Comprender el uso de motores

## EVALUACION DEL RENDIMIENTO ACADEMICO

El laboratorio tiene una ponderación de 26 puntos, distribuidos de la siguiente manera:

Actividad	Ponderación	Inicio – Fin
Proyecto 1	20 pts.	22 de feb – 21 mar
Proyecto 2	20 pts.	22 de mar – 11 abr
Proyecto 3	25 pts.	12 de abr – 3 de may
Cuestionarios Semanales	10 pts.	25 de ene – 3 de may
Ejercicios en clase	15 pts.	25 de ene – 3 de may
Examen final	10 pts.	3 de may – 3 de may
<b>Total</b>	<b>100 pts.</b>	

Todas estas fechas corresponden a el año 2025.

Para aprobar el laboratorio se debe tener una nota igual o mayor al 61%

### OBSERVACIONES

- La calificación de los proyectos de laboratorio es personal acoplándose al día y horario que se indique previamente.
- Las copias parciales o totales de los proyectos o prácticas tendrán una nota de 0 puntos y los responsables serán reportados a la Escuela de Ingeniería en Ciencias y Sistemas.
- Se debe de mandar los archivos entregables en fechas establecidas para tener derecho a calificación.

### CONTENIDO DEL CURSO

- Semana 1 - Arquitectura Y Tecnologías
  - Arquitectura Von Neuman
  - Arquitectura Harvard
  - Línea Tecnológica RISC
  - Línea Tecnológica CISC
- Semana 2 – Evolución De Los Procesadores Hasta La Actualidad
  - Primeros Procesadores x86
  - Evolución De Los Procesadores x86
  - Primeros Procesadores ARM
  - Evolución De Los Procesadores ARM
  - ARM vs X86 (En La Actualidad)
- Semana 3 – Electrónica Digital
  - Conceptos De Electrónica Digital
  - Componentes Electrónicos
  - Controladores De Dispositivos
  - Interfaces De Comunicación
  - Seguridad De Hardware
  - Simulación De Sistemas Electrónicos

- Semana 4 – Sistemas IoT En El Mundo
  - IoT en la actualidad
  - Implementación De IoT En La Industria
  - IoT Como Emprendimiento
  - Ventajas Y Desventajas Del IoT
- Semana 5 – Introducción Al IoT
  - ¿Qué es IoT?
  - Características principales de un sistema IoT
  - Primeros pasos de un proyecto IoT
  - IoT (Hardware)
  - IoT (Software)
- Semana 6 – Creando Un Sistema IoT
  - Smart Connected Design Framework
  - Aplicaciones Inteligentes
  - Analítica
  - Conectividad
  - Sensores y Actuadores
  - Infraestructura De Producto
- Semana 7 – Sistema IoT (Hardware)
  - Raspberry Pi vs Arduino
  - Python vs C
  - Control Del Hardware de Raspberry Con Python
  - Control De Sensores Y Actuadores Con Python
  - Construyendo Un Sistema IoT (Hardware)
- Semana 8 – Sistema IoT (Software)
  - Raspberry Y Cloud
  - Principales Servicios Para Sistemas IoT
  - Desarrollando Un Sistemas De Software Para Un Sistema IoT
  - Utilización De Servicios En La Nube
- Semana 9 – introducción ARM
  - Que Es Ensamblador
  - Vocabulario Básico y Tipos De Datos
  - Registros Básicos Del Procesador
  - Instalación De Herramientas En Linux
  - Instrucciones Aritméticas

- Instrucciones Lógicas
  - Instrucciones De Carga y Almacenamiento
- Semana 10 – Primeros Pasos Con ARM
  - Pre y Post Indexing
  - Ciclos Y Branching
  - Desplazamientos Y Rotaciones
  - Banderas
  - Selección Condicional
  - Niveles Y Tipos De Excepciones
  - Syscalls
- Semana 11 – Explorando ARM Y Sus Herramientas
  - Directivas De Ensamblador
  - Usando El Ensamblador De ARM (GDB)
  - Usando El Depurador De ARM (GDB)
  - Manejo De Registros Flotantes
  - Creacion De Funciones
- Semana 12 – STACK de ARM
  - El Stack Del Procesador
  - Manejo Del Stack A Nivel De Código
  - Casos De Uso Del Stack
- Semana 13 – Compatibilidad de ARM Con C
  - Desarrollando Bloques En ARM
  - Integración De Bloques ARM En C
  - Herramientas Para Analizar Rendimiento
  - Casos De Uso
  - Ventajas y Desventajas
- Semana 14 – Conclusiones del curso y lo que viene después en los siguientes cursos.

## **BIBLIOGRAFIA**

- ARM 64-Bit Assembly Language, Larry D. Pyeatt with William Ughetta, Editorial Newnes; Primera Edicion