

<b>Carrera INGENIERIA EN INFORMATICA</b>		
<b>Asignatura 3638 - Arquitectura de computadoras</b>		
<b>Trayecto Hardware</b>		
<b>Año académico 2023</b>		
<b>Responsable / Jefe de cátedra Lic. Carlos Maidana</b>		
<b>Carga horaria semanal 4hs</b>	<b>Carga horaria total 64hs</b>	<b>Créditos</b>
<b>Modalidad: Presencial</b>		
<b>Correlativas anteriores: 3631</b>		<b>Fundamentos de sistemas embebidos</b>
<b>Conocimientos necesarios:</b> Programación básica de lenguaje C. Sistemas de numeración. Lógica combinacional. Lógica secuencial.		

<b>Equipo docente</b>		
<b>Nombre</b>	<b>Cargo</b>	<b>Titulo</b>
Carlos E. Maidana	Jefe de Cátedra. Prof. Asoc.	Lic. en Gestión Educativa
Carlos Alberto Rodriguez	Profesor Adjunto	Ing. Electrónico Naval
Edgardo Alberto Gho	Jefe de Trabajos Prácticos	Ing. en Informática
Jair Ezequiel Hnatiuk	Jefe de Trabajos Prácticos	Ing. en Informática
Jorge Antonio Fiter	Jefe de Trabajos Prácticos	Lic. en Sist. de Información
Alberto Ezequiel Calaz	Ayudante de Primera	Ing. en Informática
Martín Ferreyra Biron	Ayudante de Primera	Ing. en Informática
Agostina Micaela Mottura	Ayudante Alumno	Estudiante Ing. en Informática
Alejo Burnowicz	Ayudante Alumno	Estudiante Ing. en Informática

### **Descripción de la asignatura**

La asignatura Arquitectura de Computadoras, está planteada como una materia básica en lo que hace al análisis y estudio del hardware de computadoras, en la que se estudia, desde los bloques funcionales que configuran su estructura hasta el nivel básico de los elementos que a su vez conforman dichos bloques.

En este sentido, el alumno cursante de la materia Arquitectura de Computadoras deberá poseer, al momento de su ingreso a la materia, los conocimientos básicos referidos a los elementos fundamentales en la configuración de computadoras como lo son la representación de información dentro de las mismas y el manejo de información binaria en sus distintas formas, así como los conceptos básicos del álgebra booleana y los circuitos lógicos.

Estos conocimientos, adquiridos en las materias correlativas previas, permitirán el avance hacia las aplicaciones directas de los mismos, lo que, a su vez, permitirá que el alumno tenga, una vez completado el cursado de la misma, un panorama general sobre las estructuras de hardware.

### **Metodología de enseñanza**

En cada unidad temática se introducen los conceptos fundamentales, realizando analogías

con ejemplos reales, que permite relacionar los contenidos de la materia con las herramientas habituales de trabajo.

Los contenidos de la asignatura se presentan de forma iterativa e incremental que le permitan al alumno, construir sus propios procedimientos para resolver una situación problemática.

Se motiva a los estudiantes en el uso de los foros de la plataforma MleL, para la resolución de dudas tanto de conceptos teóricos como prácticos.

#### **Objetivos de aprendizaje**

A través de esta asignatura, el alumno habrá adquirido los conocimientos necesarios para:

- Comprender los conceptos de la arquitectura de programación de una CPU.
- Entender los diferentes modos de direccionamiento y la aplicación de los mismos en la ejecución de instrucciones.
- Determinar y calcular la eficiencia de una CPU.
- Diferenciar los diferentes bloques funcionales de una computadora.
- Interpretar el funcionamiento de una máquina secuencial y de sus dispositivos periféricos.
- Entender el funcionamiento de procesadores de alta eficiencia (RISC - Harvard)

#### **Contenidos mínimos**

Unidad Central de Procesos. Arquitecturas. Microprogramación: Unidad de control. Microinstrucciones. Ciclos de máquina. Conceptos de Pipeline. Lenguaje Assembler. Memoria. Memoria caché. Administración de procesos de Entrada y Salida. Interrupciones y DMA.

#### **Competencias a desarrollar**

A lo largo del período de cursada al menos se desarrollan en cierto grado de profundidad las siguientes competencias:

##### **Genéricas**

- Desempeño en equipos de trabajo.
- Comunicación efectiva.
- Actuación profesional ética y responsable.
- Aprendizaje continuo.
- Desarrollo de una actitud profesional emprendedora.

##### **Específicas**

- Especificación, proyecto y desarrollo de software.
- Establecimiento de métricas y normas de calidad de software.
- Procedimientos y certificaciones del funcionamiento, condición de uso o estado de sistemas de información, sistemas de comunicación de datos, software, seguridad

informática y calidad de software.

- Identificación, formulación y resolución de problemas de ingeniería en sistemas de información/informática.
- Concepción, diseño y desarrollo de proyectos de ingeniería en sistemas de información / informática.
- Gestión, planificación, ejecución y control de proyectos de ingeniería en sistemas de información / informática.
- Utilización de técnicas y herramientas de aplicación en la ingeniería en sistemas de información / informática.
- Generación de desarrollos tecnológicos y/o innovaciones tecnológicas.

Programa analítico	
Unidad 0	<p><b>Computadoras.</b></p> <p>0.0 - Definiciones. Bloques funcionales. Buses. Unidad central de procesos. Formato de instrucciones.</p> <p>0.1 – Arquitecturas. Estructura vs. Arquitectura. Arquitecturas orientadas a la pila (Stack) , arquitecturas orientadas al acumulador, arquitecturas registro – memoria, registro – registro. Arquitectura Von Neumann vs Arquitectura Harvard. Unidad de control, microinstrucciones, ciclos de máquina. Computadoras de 3 buses y computadoras de 4 buses. Computadoras CISC vs RISC. Byte alignment &amp; Ordering. Historia de MIPS y RISC-V.</p> <p>0.2 – La arquitectura RISC-V. El Procesador RISC-V. Instrucciones de operaciones entre registros ( R ). Instrucciones con constantes ( I, U ). Instrucciones para acceder a memoria ( S, L ). Instrucciones de control de flujo ( B, J ). Ensamblado y pseudo instrucciones.</p> <p>0.3 – Programación assembler. Entorno de programación RIPES. Interpretación de estructuras básicas en lenguaje “C” y su traducción a ASMemblar de RISC-V. Subrutinas.</p> <p>0.4 – La Microarquitectura RISC-V. El procesador RISC-V2p.</p> <p>0.5 – Pipelining. Eficiencia. CPI (ciclos de reloj por instrucción). Optimización y mejora de los CPI. Latencia. Riesgos.</p>
Unidad 1	<p><b>Memoria.</b></p> <p>1.0 - Memoria principal. Clasificación de memorias. Memoria volátil estática, fundamentos tecnológicos. Memoria volátil dinámica, fundamentos tecnológicos. Refresco. Evolución y mejoras. Memoria no volátil. Estructura y bancos.</p> <p>1.1 – Memoria cache. Principios de localidad espacial y temporal. Esquemas multinivel, unificado y dividido. Concepto de línea de memoria</p>

	cache, etiquetas y bits de señalización Organizaciones Asociativa, directa y asociativa por conjuntos. Implementaciones. Políticas de carga y escritura. Tiempo de acceso. Algoritmos de reemplazo. Causas de fallas.
Unidad 2	<p><b>Entrada – Salida</b></p> <p>2.0 – Depuración. GNU CC. (compilador, ensamblador, linker, objdump). Entorno ESP-IDF.</p> <p>Un procesador ESP32 con RISC-V: ESP32-C3. Especificaciones del procesador, capacidad de memoria. Ejemplos de programación en lenguaje C. GDB (GNU Debugger) sobre USB. Ejemplos de depuración de programas escritos en lenguajes C.</p> <p>2.1 – Dispositivos de E/S. Periféricos. Interfaces.</p> <p>2.2 – Interrupciones y traps. Identificación de dispositivos. Timers.</p> <p>2.3 – Comunicaciones serie asincrónica y serie sincrónica.</p> <p>2.4 – Acceso Directo a Memoria. Función del DMAC. Robo de ciclos y detención de CPU. Controlador DMA. Ejemplos.</p>

Planificación de actividades					
Semana	Clase	Actividad	Tipo	Duración estimada	Unidad
Semana 1	1	<p>Presentación de la materia. Definiciones. Bloques funcionales. Buses. Unidad central de procesos. Formato de instrucciones. Arquitecturas. Estructura vs. Arquitectura. Arquitecturas orientadas a la pila (Stack) , arquitecturas orientadas al acumulador, arquitecturas registro – memoria, registro – registro. Arquitectura Von Neumann vs Arquitectura Harvard. Unidad de control, microinstrucciones, ciclos de máquina. Computadoras de 3 buses y computadoras de 4 buses. Computadoras CISC vs RISC. Byte alignment &amp; Ordering. Historia de MIPS y RISC-V. La arquitectura RISC-V. El</p>	Teórica	4 horas	Unidad 0.0 – 0.1 - 0.2

		Procesador RISC-V. Instrucciones de operaciones entre registros ( R ). Instrucciones con constantes ( I, U ). Instrucciones para acceder a memoria ( S, L ). Instrucciones de control de flujo ( B, J ). Ensamblado y pseudo instrucciones.			
Semana 2	2	Programación assembler. Entorno de programación RIPES. Interpretación de estructuras básicas en lenguaje “C” y su traducción a Assembler de RISC-V. Subrutinas.	Teórica	4 horas	Unidad 0.3
Semana 3	3	Programación utilizando el entorno de programación RIPES. Resolución de problemas	Práctica de laboratorio o clase virtual sincrónica	4 horas	Unidad 0.3
Semana 4	4	La Microarquitectura RISC-V. El procesador RISC-V2p. Pipelining. Eficiencia. CPI (ciclos de reloj por instrucción). Optimización y mejora de los CPI. Latencia. Riesgos.	Teórica	4 horas	Unidad 0.4 – 0.5
Semana 5	5	Repaso y de Eficiencia y Pipelining. Resolución de problemas. Memoria principal. Clasificación de memorias. Memoria volátil estática, fundamentos tecnológicos. Memoria volátil dinámica, fundamentos tecnológicos. Refresco. Evolución y mejoras. Memoria no volátil. Estructura y bancos.	Teórica y Práctica.	2 horas práctica. 2 horas teoría	Unidad 0.4 – 0.5. Unidad 1.0
Semana 6	6	Consultas pre-parcial. Repaso Memorias.	Consulta y Práctica. Teórica	4 horas	Unidad 0 y Unidad 1.0
Semana 7	7	Primer parcial. Memoria cache. Principios de localidad espacial y temporal. Esquemas multinivel, unificado y dividido. Concepto de línea	Evaluación. Teórica	2 horas evaluación – 2 horas teoría	Evaluación Unidad 0. Teoría Unidad 1.1

		de memoria cache, etiquetas y bits de señalización Organizaciones Asociativa, directa y asociativa por conjuntos. Implementaciones. Políticas de carga y escritura. Tiempo de acceso. Algoritmos de reemplazo. Causas de fallas.			
Semana 8	8	Práctica de memoria cache en entorno RIPES, con ejemplos de ejecución de programas.	Teoría - Práctica	4 horas	Unidad 1.1
Semana 9	9	Práctica de memoria cache con práctica de código y calculo de tasas de fallas configurando de distintas formas la memoria cache	Práctica de laboratorio o clase virtual sincrónica	4 horas	Unidad 1
Semana 10	10	Depuración. GNU CC. (compilador, ensamblador, linker, objdump). Entorno ESP-IDF. Un procesador ESP32 con RISC-V: ESP32-C3. Especificaciones del procesador, capacidad de memoria. Ejemplos de programación en lenguaje C. GDB (GNU Debugger) sobre USB. Ejemplos de depuración de programas escritos en lenguajes C.	Práctica de laboratorio	4 horas	Unidad 2.0
Semana 11	11	Dispositivos de E/S. Periféricos. Interfaces. Acceso Directo a Memoria. Función del DMAC. Robo de ciclos y detención de CPU. Controlador DMA. Ejemplos.	Teórica	4 horas	Unidad 2.1 – 2.2 – 2.4
Semana 12	12	Interrupciones y traps. Identificación de dispositivos. Timers. Práctica de interrupciones sobre ESP32-C3.	Práctica de laboratorio. Teórica	4 horas	Unidad 2.2
Semana 13	13	Comunicaciones serie asincrónica y serie sincrónica.	Práctica de laboratorio. Teórica	4 horas	Unidad 2.3

		Práctica de comunicación serie sobre ESP32-C3.			
Semana 14	14	Segundo Parcial.	Evaluación - Teórica	4 horas	Unidades 1 y 2
Semana 15	15	Resolución segundo Parcial. Ataques por canal secundario.	Práctica - Teórica	4 horas	Unidades 1 y 2
Semana 16	16	Examen recuperatorio. Cierre de notas.	Evaluación	4 horas	

**Evaluación**

Los exámenes parciales o recuperatorios, se llevarán a cabo mediante preguntas teóricas para el desarrollo, preguntas del tipo opción múltiple y un porcentaje no menor al 40% de ejercicios prácticos en los que el alumno debe completar datos en una grilla prearmada.

Debido a que es una materia con régimen de cursada cuatrimestral, la resolución 54/2011 establece que se deben tomar dos evaluaciones parciales presenciales, un solo recuperatorio no integrador, también presencial.

<b>Primera evaluación</b>	Semana 7 - Clase 7	Examen escrito	2 Horas. TM: 08:00 - 10:00 Hs TT: 14:00 - 16:00 Hs TN: 19:00 - 21:00 Hs
<b>Segunda evaluación</b>	Semana 14 - Clase 14	Examen escrito	2 Horas. TM: 08:00 - 10:00 Hs TT: 14:00 - 16:00 Hs TN: 19:00 - 21:00 Hs
<b>Recuperatorio</b>	Semana 16 - Clase 16	Examen escrito	2 Horas. TM: 08:00 - 10:00 Hs TT: 14:00 - 16:00 Hs TN: 19:00 - 21:00 Hs

**Bibliografía obligatoria**

Titulo	Autor	Editorial	Edición	Año
Fundamentos de los sistemas digitales	Thomas Floyd	Pearson	11	2016
Principios de arquitectura de computadoras	Murdocca - Heuring	PEARSON EDUCACION	1	2002
Computer Organization and Architecture	William Stallings	Pearson	10	2016
Computer Architecture	Hennessy - Paterson	Morgan Kaufmann	6	2017

Bibliografía complementaria recomendada				
Título	Autor	Editorial	Edición	Año
Digital Design and Computer Architecture	David Harris	Morgan Kaufmann	2	2012
Computer Organization and Design RISC-V edition	Patterson - Hennessy	Morgan Kaufmann	2	2020
The RISC-V Reader	Patterson - Waterman	Strawberry Canyon	1	2017

Otros recursos obligatorios (videos, enlaces, otros) Incluir una fila por cada recurso	
Nombre	Enlace
Sistemas de numeración	<a href="https://www.youtube.com/watch?v=xqulaQMYikM">https://www.youtube.com/watch?v=xqulaQMYikM</a>
Códigos	<a href="https://www.youtube.com/watch?v=1S0XN1hUWjI">https://www.youtube.com/watch?v=1S0XN1hUWjI</a>
Lógica Combinatoria	<a href="https://www.youtube.com/watch?v=Dw2bIGU_nfU">https://www.youtube.com/watch?v=Dw2bIGU_nfU</a>
Lógica secuencial	<a href="https://www.youtube.com/watch?v=4yc-1rnAU2w">https://www.youtube.com/watch?v=4yc-1rnAU2w</a>

Otros recursos complementarios (videos, enlaces, otros) Incluir una fila por cada recurso	
Nombre	Enlace
Repositorio de clases	<a href="http://tiny.cc/acunlam">http://tiny.cc/acunlam</a>