

PROGRAMA DE CURSO

Código	Nombre			
EL4102	ARQUITECTURA DE COMPUTADORES			
Nombre en Inglés				
COMPUTER ARCHITECTURE				
SCT	Unidades Docentes	Horas de Cátedra	Horas Docencia Auxiliar	Horas de Trabajo Personal
	10	3	1,5	5,5
Requisitos			Carácter del Curso	
EL 4002 SISTEMAS DIGITALES			Electivo de línea de especialización Electivo de posgrado	
Resultados de Aprendizaje				
<p>Los Procesadores son sistemas digitales que se han desarrollado de una forma vertiginosa desde los años 80 y que han tomado su camino propio. Esto ha significado que, del punto de vista académico, este tema sea tratado en forma separada y no como parte de un curso de Sistemas Digitales. En este curso se analizan las diferentes arquitecturas de diseño de un Procesador que se han desarrollado en el tiempo y a lo que se ha llegado en la actualidad. Además, permite a los alumnos comprender la terminología y conceptos que se utilizan en la descripción y especificaciones de estos dispositivos.</p> <p>Al final del curso se espera que el estudiante:</p> <ol style="list-style-type: none"> 1. Comprenda los principios organizativos de los sistemas computacionales, así como las relaciones y funciones de sus distintos componentes de software y hardware. 2. Analice y diseñe software para programación de bajo nivel de sistemas computacionales. 3. Analice y evalúe el funcionamiento y el rendimiento de sistemas computacionales. 4. Conozca las tecnologías utilizadas en los sistemas computacionales actuales. 				

Metodología Docente	Evaluación General
De acuerdo a lo entregado en las clases expositivas, el alumno desarrollará proyectos de programación en lenguaje <i>assembly</i> de procesadores MIPS y RISC-V (5 a 6 Tareas en el Semestre) que irán de menor a mayor en cuanto a su complejidad, utilizando diferentes herramientas de simulación como MARS 4.5 (simulador para MIPS) y RARS 1.1 (simulador para RISC-V), disponibles en la Web.	Se realizarán Ejercicios, Tareas (Proyectos de Programación en lenguaje <i>Assembly</i>), Controles y Examen. Además, para el último proyecto del Semestre, los alumnos trabajarán en grupos de 2 a 3 personas y harán una presentación de los resultados.

Unidades Temáticas

Número	Nombre de la Unidad	Duración en Semanas
1	Niveles de Abstracción en los Computadores y Tecnologías	1 semana
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
1. La Revolución del Computador 1.1. Clases de Computadores (desde 1960) 1.2. El significado de "Performance" 1.3. Los ocho grandes conceptos en Arquitectura de Computadores 1.4. Anatomía de un Computador 1.5. Niveles de Abstracción 1.6. Tendencias de la Tecnología 1.7. El porqué del paso de Uniprosesadores a Multiprosesadores 1.8. "Benchmarks" para los procesadores 1.9. Errores de conceptos y falacias 1.10. Conclusiones	Se obtienen los conceptos básicos relacionados con la arquitectura de los computadores entre los cuales destacan: <i>performance</i> , los niveles de abstracción, evolución tecnológica y el concepto de Multiprosesadores <i>Multicore</i> .	[1] Patterson y Hennessy: Capítulo 1 [3] Patterson y Hennessy: Capítulo 1

Número	Nombre de la Unidad	Duración en Semanas
2	Instrucciones: el Lenguaje del Computador	2 semanas
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
2. El Lenguaje del Computador 2.1. Operaciones del Hardware del Computador 2.2. Operandos del Hardware del Computador 2.3. Números Con y Sin Signo 2.4. Representación de Instrucciones en el Computador 2.5. Operaciones Lógicas 2.6. Instrucciones para Tomar	Para comandar el hardware de un computador, se debe <i>hablar</i> su lenguaje. Las palabras de un lenguaje de un computador son las llamadas <i>instrucciones</i> y su vocabulario es el <i>conjunto de instrucciones</i> . En esta unidad, aprenderemos los conjuntos de instrucciones de computadores reales (MIPS y RISC-V), tanto en la forma escrita por una	[1] Patterson y Hennessy: Capítulo 2 [3] Patterson y Hennessy: Capítulo 2 [4] Patterson y Hennessy: Capítulo 2

<p>Decisiones</p> <p>2.7. Procedimientos en el Hardware del Computador</p> <p>2.8. Comunicación con las Personas</p> <p>2.9. Direccionamiento para Constantes y Direcciones Inmediatas en MIPS y RISC-V</p> <p>2.10. Paralelismo e Instrucciones: Sincronización</p> <p>2.11. Traducción y Ejecución de un Programa</p> <p>2.12. Ejemplo de un <i>Sort</i> en MIPS y RISC-V</p> <p>2.13. Arreglos versus Punteros</p> <p>2.14. Casos Reales: Instrucciones del x86</p> <p>2.15. Errores de conceptos y falacias</p> <p>2.16. Conclusiones</p>	<p>persona como en la forma leída por el computador.</p>	
---	--	--

Número	Nombre de la Unidad	Duración en Semanas
3	Aritmética para Computadores	2 semanas
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
<p>3. Aritmética para Computadores</p> <p>3.1. Sumas y Restas</p> <p>3.2. Multiplicaciones</p> <p>3.3. Divisiones</p> <p>3.4. Punto Flotante (Estándar IEEE 754)</p> <p>3.5. Paralelismo y Aritmética de Computadores: Asociatividad y Paralelismo a Nivel de Datos</p> <p>3.6. Errores de conceptos y falacias</p> <p>3.7. Conclusiones</p>	<p>Las palabras del computador están compuestas de bits; por lo tanto, las palabras pueden ser representadas como números binarios. Esta unidad muestra que los enteros pueden ser representados en forma decimal o binaria. Pero, ¿qué pasa con los otros números comúnmente utilizados? Por ejemplo:</p> <ul style="list-style-type: none"> • ¿Qué hay de las fracciones y otros números reales? • ¿Qué pasa si una operación genera un número mayor que el que se pueda representar? 	<p>[1] Patterson y Hennessy: Capítulo 3</p> <p>[3] Patterson y Hennessy: Capítulo 3</p> <p>[4] Patterson y Hennessy: Capítulo 3</p>

	<ul style="list-style-type: none"> • ¿Cómo hace el hardware realmente para multiplicar y dividir números? 	
--	--	--

Número	Nombre de la Unidad	Duración en Semanas
4	El Procesador	4 semanas
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
4. El Procesador 4.1. Convenciones del Diseño Lógico 4.2. Diseño e Implementación del Sistema Controlado ("Datapath") 4.3. La Arquitectura "Single-Cycle" 4.4. La Arquitectura "Multi-Cycle" 4.5. Introducción a la Arquitectura "Pipeline" 4.6. "Datapath" y Control de la Arquitectura "Pipeline" 4.7. "Hazards" de Datos: "Forwarding" versus "Stalling" 4.8. "Hazards" de Control 4.9. Excepciones e Interrupciones 4.10. Paralelismo y Paralelismo Avanzado a Nivel de Instrucciones 4.11. Ejemplos de Casos Reales 4.12. Errores de conceptos y falacias 4.13. Conclusiones	Esta unidad entrega una explicación de los principios y técnicas utilizadas en el diseño e implementación de un Procesador, comenzando con un enfoque abstracto y simplificado para luego continuar con la implementación de un "datapath" y, con ello, construir versiones simples de los procesadores MIPS y RISC-V con la suficiente capacidad para implementar un conjunto de instrucciones básicas en cada caso.	[1] Patterson y Hennessy: Capítulo 4 [3] Patterson y Hennessy: Capítulo 4

Número	Nombre de la Unidad	Duración en Semanas
5	La Memoria y su Esquema Jerárquico	4 semanas
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
<p>5. La Memoria</p> <p>5.1. Introducción a la Memoria Cache</p> <p>5.2. Medidas y Mejoras de la <i>Performance</i> de la Memoria Cache</p> <p>5.3. Memoria Virtual</p> <p>5.4. Una Infraestructura Común para las Jerarquías de Memoria</p> <p>5.5. Máquinas Virtuales</p> <p>5.6. El Uso de una Máquina de Estado Finito (FSM) para implementar el Control de una Memoria Cache Básica</p> <p>5.7. Paralelismo y Jerarquías de Memoria: Coherencia de la Memoria Cache</p> <p>5.8. Ejemplo del Diseño e Implementación de un Controlador para una Memoria Cache</p> <p>5.9. Ejemplos de Casos Reales: Jerarquías de Memorias en ARM e Intel i7</p> <p>5.10. Errores de conceptos y falacias</p> <p>5.11. Conclusiones</p>	<p>Esta unidad entrega una explicación de los principios y técnicas utilizadas en el diseño e implementación de la Memoria Cache. Se describe la Memoria y su esquema jerárquico. Se introducen los conceptos de Memoria Virtual y Máquina Virtual.</p>	<p>[1] Patterson y Hennessy: Capítulo 5</p> <p>[3] Patterson y Hennessy: Capítulo 5</p>

Número	Nombre de la Unidad	Duración en Semanas
6	Almacenamientos y Dispositivos de Entrada/Salida (E/S)	1 semanas
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
<p>6. Almacenamientos y Dispositivos de E/S</p> <p>6.1. Fiabilidad, Confiabilidad y Disponibilidad de los Dispositivos de E/S</p> <p>6.2. Almacenamiento en Discos</p> <p>6.3. Almacenamiento en Memorias <i>Flash</i></p> <p>6.4. Interconexión entre Procesadores, Memorias y Dispositivos de E/S</p> <p>6.5. Interfaces entre Dispositivos de E/S, Procesadores, Memoria y el Sistema Operativo</p> <p>6.6. Medidas de Performance de los Dispositivos de E/S: Ejemplos de Discos y Sistemas de Archivos</p> <p>6.7. Diseño de un Sistema de E/S</p> <p>6.8. Paralelismo y E/S: Estructuras de Discos RAID</p> <p>6.9. Conceptos de Redes de Computadores</p> <p>6.10. Errores de conceptos y falacias</p> <p>6.11. Conclusiones</p>	<p>Esta unidad entrega los principios y técnicas utilizadas actualmente en los dispositivos de almacenamiento y, en general, en los dispositivos de E/S.</p>	<p>[1] Patterson y Hennessy: Capítulo 6</p>

Número	Nombre de la Unidad	Duración en Semanas
7	<i>Multicores, Multiprocesadores y Clusters</i>	1 semana
Contenidos	Resultados de Aprendizajes de la Unidad	Referencias a la Bibliografía
<p>7. Procesadores Paralelos: Desde el Cliente hasta la “Nube”</p> <p>7.1. La dificultad de crear Programas con Procesamiento Paralelo</p> <p>7.2. Multiprocesadores con Memoria Compartida</p> <p>7.3. <i>Clusters</i> y otros Multiprocesadores “Pasa-Mensajes”</p> <p>7.4. Implementación en Hardware del “Multithreading”</p> <p>7.5. Arquitecturas SISD, MIMD, SIMD, SPMD y Vectoriales</p> <p>7.6. Introducción a las Unidades de Procesamiento Gráfico (GPU)</p> <p>7.7. Introducción a las Topologías de Redes de Multiprocesadores</p> <p>7.8. <i>Benchmarks</i> para Multiprocesadores</p> <p>7.9. “Roofline”: un Modelo Simple de la Medida de la Performance de Multiprocesadores</p> <p>7.10. Errores de conceptos y falacias</p> <p>7.11. Conclusiones</p>	<p>Esta unidad plantea la problemática que si el reemplazar grandes e ineficientes procesadores por muchos, pequeños y eficientes unidades de proceso, se puede lograr una mejor <i>performance</i> por <i>watt</i> o por <i>joule</i>. Si, además, el software es capaz de usarlos eficientemente. Por lo tanto, una mejora en la eficiencia del consumo de energía, tiene que ir aparejada con una mejora escalable de la <i>performance</i> en el caso de los multiprocesadores.</p>	<p>[1] Patterson y Hennessy: Capítulo 7</p> <p>[3] Patterson y Hennessy: Capítulo 6</p>

Bibliografía General

- [1] David A. Patterson and John L. Hennessy, "Computer Organization and Design: The Hardware /Software Interface", Fourth Edition Revised, Morgan Kaufmann Publishers, 2012
- [2] David A. Patterson and John L. Hennessy, "Computer Organization and Design: The Hardware /Software Interface", Fifth Edition, Morgan Kaufmann Publishers, 2014
- [3] David A. Patterson and John L. Hennessy, "Computer Organization and Design: The Hardware /Software Interface", RISC-V Edition, Morgan Kaufmann Publishers, 2018
- [4] David A. Patterson and John L. Hennessy, "Computer Organization and Design: The Hardware /Software Interface", ARM Edition, Morgan Kaufmann Publishers, 2017

Vigencia desde:	31 Enero 2019
Elaborado por:	Francisco Rivera
Revisado por:	