



Ministerio de Cultura y Educación  
 Universidad Nacional de San Luis  
 Facultad de Ciencias Físico Matemáticas y Naturales  
 Departamento: Informatica  
 Area: Area II: Sistemas de Computacion

(Programa del año 2010)  
 (Programa en trámite de aprobación)  
 (Presentado el 07/09/2010 10:29:42)

### I - Oferta Académica

Materia	Carrera	Plan	Año	Período
ARQUITECTURA DEL PROCESADOR II	LIC.EN CS.DE LA COMPUTACION	006/0 5	2010	2° cuatrimestre

### II - Equipo Docente

Docente	Función	Cargo	Dedicación
AGUIRRE, GUILLERMO CARLOS	Prof. Responsable	P.Adj Exc	40 Hs
GROSSO, ALEJANDRO LEONARDO	Prof. Colaborador	P.Adj Exc	40 Hs
ARROYUELO BILLIARDI, JORGE A.	Responsable de Práctico	JTP Exc	40 Hs
ARROYUELO, MONICA DEL VALLE	Auxiliar de Práctico	A.1ra Exc	40 Hs

### III - Características del Curso

Credito Horario Semanal				
Teórico/Práctico	Teóricas	Prácticas de Aula	Práct. de lab/ camp/ Resid/ PIP, etc.	Total
0 Hs	3 Hs	3 Hs	Hs	6 Hs

Tipificación	Periodo
C - Teoria con prácticas de aula	2° Cuatrimestre

Duración			
Desde	Hasta	Cantidad de Semanas	Cantidad de Horas
09/08/2010	19/11/2010	15	90

### IV - Fundamentación

La creciente demanda de mayor poder computacional de los procesadores ha dado origen a arquitecturas complejas con unidades funcionales especializadas, memorias de alta velocidad, multiprocesadores, división de tareas en etapas concurrentes, ejecución de intrucciones a medida que están disponibles sus operandos, etc. Todas estas particularidades hacen necesaria la aplicación de técnicas y estrategias más complejas que la involucradas en los procesadores simples. Las arquitecturas avanzadas están concebidas para procesar problemas específicos que deben ser identificados y resueltos mediante prácticas de programación especiales.

### V - Objetivos / Resultados de Aprendizaje

Al finalizar el curso los alumnos deberán poder:

- \*Identificar los principales parámetros que miden el desempeño de los procesadores.
- \*Reconocer las similitudes y diferencias entre los distintos esquemas de memorias cache.
- \*Desarrollar el análisis cuantitativo del desempeño de procesadores provistos con memoria cache.

\*Conocer las características de los procesadores pipeline y las particularidades consideraras al medir el desempeño de los mismos.

\*Dominar los detalles involucrados en el funcionamiento de los procesadores pipeline con etapas multicyclo y las estrategias empleadas para despachar instrucciones fuera de orden.

\*Percibir la complejidad requerida para computar soluciones en sistemas de multiprocesadores y las redes de interconexión necesarias en estos sistemas.

\*Determinar como se clasifican los diferentes procesadores según el nivel de paralelismo que presentan.

\*Desarrollar pequeños sistemas en VHDL.

## VI - Contenidos

### Unidad 1: Performance en procesadores RISC.

Arquitecturas según el conjunto de instrucciones. Procesadores RISC. Arquitectura del set de instrucciones RISC. Frecuencia y ciclo de reloj. Unidades pequeñas de tiempo. Ecuación de tiempo de CPU. Ciclos por instrucción. Relación entre el ciclo y el tiempo de CPU. Desempeño y tiempo de ejecución. Arquitectura multicyclo. Señales de control y camino de datos. Set de instrucciones y desempeño. Aceleración (speed-up). Ley de Amdahl.

### Unidad 2: Pipelining.

¿Qué es pipelining?. Comportamiento básico del pipeline. Performance ideal. Problemas en los pipe: los hazards. Distintos tipos de hazards: estructurales, de datos y de control. Dependencia de datos. Hazards de datos: RAW, WAW, WAR. Los atascos (stalls). Adelantamiento. Interlock. Impacto de los hazards en el análisis de la performance. Penalidades por branch. Salto demorado. Programación en DLXView.

### Unidad 3: Paralelismo a nivel de instrucción y Pipelines avanzados.

Ejecución multi-ciclo. Finalización fuera de orden. Latencia e intervalo de iniciación. Scheduling dinámico de instrucciones. Scoreboarding. Algoritmo de Tomasulo, aplicación en una unidad de punto flotante. Arquitecturas de IMB 360/91 y CDC 6600. Programación con planificación dinámica. Especulación

### Unidad 4: Memorias.

Jerarquía de memorias. Memorias cache. Hit y Miss. Consideraciones generales del empleo de cache. Organizaciones de cache: directas, asociativas y conjunto asociativo. Performance de memorias cache. Tipos de cache: write-back y write-through. Comportamiento de la jerarquía de memorias. Penalidades por miss. Desempeño considerando el uso de la jerarquía de memoria. Ejemplo Alpha AXP 21264.

### Unidad 5: Multiprocesadores y redes de interconexión.

Nivel de paralelismo en los programas. Clasificación de Flynn. Arquitecturas de memoria centralizada (UMA) y distribuida (NUMA). Redes de interconexión: consideraciones generales. Interconexiones dinámicas. Esquemas de única etapa y de múltiples etapas. Sistemas de buses. Redes crossbar. Redes Omega. Redes bloqueantes y no bloqueantes. Cache en sistemas de multiprocesadores. Coherencia y consistencia de cache. Protocolos snoopy y directorio. modelo de consistencia. Sincronización.

### Unidad 6: Diseño en VHDL.

Elementos básicos de VHDL. Paquetes. Entidad. Arquitectura. Sentencias concurrentes y secuenciales.

## VII - Plan de Trabajos Prácticos

Prácticos Nro. 1 y 2: Programación assembly y Medidas de Performance.

Programación en MIPS64. Uso de la Ley de Amdahl para el cálculo de la performance ganada por una mejora (Speedup). Uso de la ecuación de la CPU. Camino de datos

Prácticos Nro. 3 : Pipelines y riesgos.

Implementación del pipelining del MIPS64. Impacto de problemas en el pipe: hazard de estructurales y hazard de datos. Los hazard de control: técnicas utilizadas para las instrucciones de salto. Planificación estática de instrucciones para reducir los hazards.

Prácticos Nro. 4 : Planificación Dinámica

Algoritmo de scoreboarding. Manejo de los distintos tipos de hazards. Algoritmo de Tomasulo. Scoreboarding vs. Tomasulo

Práctico Nro. 5: Distintas organizaciones de memoria cache. Técnicas de ubicaciones de bloques: mapeo directo, conjuntos asociativos, memoria completamente asociativa. Pasos en la lectura y escritura de la memoria cache. Reduciendo los miss de cache: Comparación entre diferentes arquitecturas con memoria cache de diferentes tamaños.

Práctico Nro. 6: Programación en VHDL.

## VIII - Regimen de Aprobación

### \* Regularización

Para regularizar la materia el alumno deberá cumplir con los siguientes requisitos:

Aprobar un examen parcial, o su correspondiente recuperación sobre temas que abarcan las cuatro primeras unidades.

Aprobar un trabajo práctico que cubre la última unidad de la materia. Este trabajo puede ser defendido y presentado en grupos de hasta 3 alumnos.

Aquellos alumnos que estén en condiciones tendrán derecho a una recuperación adicional por trabajo.

Mostrar compromiso con la materia a través de asistencia regular a clase y realización de los prácticos de aula y máquina.

### \* Examen Final

Los alumnos regulares deberán rendir un examen final (que podrá ser oral o escrito) que consistirá en preguntas sobre los temas desarrollados durante el dictado de la materia.

### \* Alumnos libres

Los alumnos que desean rendir libre la materia se deberán poner en contacto con la cátedra con 5 días de anticipación a los efectos de realizar un práctico, el cual contendrá ejercicios similares a los desarrollados en los prácticos durante el dictado de la materia. Aprobando éste trabajo práctico el alumno tendrá derecho a rendir un examen oral con iguales características que el de los alumnos regulares.

## IX - Bibliografía Básica

[1] Computer architecture a quantitative approach. John Hennessy y David Patterson. 4th Edition. Editorial Morgan Kaufmann. (2006)

[2] Computer Organization & Design. David Patterson and Jhon L. Hennessy. 3rd Edition. Editorial Morgan Kaufmann. (2007)

[3] Parallel Computer Architecture: A Hardware/Software Approach. David Culler, J.P. Singh, Anoop Gupta

## X - Bibliografía Complementaria

[1] Computer Architecture design and performance. Barry Wilkinson. Editorial Prentice-Hall (1996).

[2] Zemian & otros, "The PH processor: a soft embedded core for use in university research and teaching"

[3] P.M. Sailer & D.R. Kaeli "The DLX Instruction Set Architecture Handbook" Morgan Kaufmann, 1996.

[4] M.Flynn, "Computer engineering 30 years after the IBM Model 91"

[5] D.A. Patterson, D.R. Ditzel "The Case for the Reduced Instruction Set Computer" Commun. ACM, year 1985, num.1, vol.28, pag.8--21

## XI - Resumen de Objetivos

Al finalizar el curso los alumnos deberan poder:

\*Identificar los principales parámetros que miden el desempeño de los procesadores.

\*Desarrollar diferentes analisis cuantitativos del desempeño de procesadores provistos con memoria cache.

\*Conocer las características de los procesadores pipeline y las particularidades consideraras al medir el desempeño de los mismos.

\*Percibir la complejidad requerida para computar soluciones en sistemas de multiprocesadores y las redes interconexión necesarias en estos sistemas.

\*Desarrollar programas simples en VHDL.

## **XII - Resumen del Programa**

Medidas de performance. Jerarquía de memorias. Memorias cache. Procesadores pipeline. Tipos de hazards. Performance en sistemas pipelining. Impacto de los branches en el pipe. Pipeline multiciclo. Latencia e intervalo de iniciación. Planificación dinámica de instrucciones. Multiprocesadores. Arquitecturas UMA y NUMA. Coherencia de cache. Nivel de paralelismo de los programas. Sistemas de interconexión. Programación en VHDL.

## **XIII - Imprevistos**

Comunicarse con la cátedra.

Arquitectura del Procesador II.

gaguirre@unsl.edu.ar

Departamento de Informática.

Facultad de Cs. Físico, Matemáticas y Naturales.

Universidad Nacional de San Luis.

Ejército de los Andes 950. CP 5700. Bloque II, 1º piso, Box 7

Tel 2652-420822 Interno 231

## **XIV - Otros**

--

### **ELEVACIÓN y APROBACIÓN DE ESTE PROGRAMA**

#### **Profesor Responsable**

Firma:

Aclaración:

Fecha: