

## ORGANIZACIÓN DE SISTEMAS DE CÓMPUTO

C.1 Arquitectura de procesadores  
(Kain Y. Richard)

Por: M. en C. Sergio Sandoval Reyes

### RESUMEN

Este libro describe los conceptos y métodos seguidos en el diseño de diversas arquitecturas de computadoras. El autor plantea desde el principio, una estructura a partir de la cual se clarifican las similitudes y diferencias de varias arquitecturas, mostrando no sólo las técnicas básicas, sino también las relaciones entre los niveles de software y hardware, en la implementación de dichos sistemas y su operación.

### DISEÑOS GENERALES Y ESPECÍFICOS

El autor incluye ejemplos de varios lenguajes de programación, sistemas operativos y diseño de procesadores, tanto de reciente aparición como tradicionales. La finalidad de ello es enfatizar las similitudes de los problemas en el diseño de sistemas de cómputo, y las alternativas de solución existentes. Para ello aplica una metodología *top-down* (principios generales, métodos de diseño, lenguajes, procesadores e implementación), empleando ejemplos actuales de lenguajes de programación, sistemas operativos y procesadores. De esta forma se ilustran los problemas y los diseños que los resuelven.

### ANTECEDENTES

El autor presupone que el lector está familiarizado con al menos un procesador, un sistema operativo, un lenguaje de programación (preferentemente C++), y el diseño de procesadores encausados RISC, al nivel de Patterson y Hennessy.

### ESTRUCTURA DEL LIBRO

El libro está formado por nueve capítulos y 12 anexos, pero dividido esencialmente en tres partes. La primera parte tiene una estructura basada en el modelo *Von Neuman*, de un sistema uniprocador (diseño de: conjunto de instruccio-

nes, memoria, y control). La segunda parte se enfoca a aspectos que se presentan cuando múltiples hilos de control (*threads*), están activos simultáneamente dentro de todo el sistema. La tercera parte, trata de las restricciones adicionales que se imponen en un sistema, cuando se consideran aspectos de protección y seguridad. Una breve descripción de los capítulos y anexos, se presenta a continuación.

**CAPÍTULO 1.** *Ilusiones:* Es una introducción a las suposiciones básicas o ilusiones que los programadores hacen al enfrentarse con la complejidad de sistemas de cómputo. El énfasis de este capítulo es señalar la medida en que las alternativas de diseño pueden cumplir con dichas suposiciones.

**CAPÍTULO 2.** *Diseño del conjunto de instrucciones:* Aquí se hace una breve revisión de las ideas básicas de la estructura de un procesador, lo cual sirve de base en los ejemplos de los subsecuentes capítulos y apéndices (en particular los apéndices A a D, en donde se detallan arquitecturas específicas de procesadores).

**CAPÍTULO 3.** *Organización de la memoria:* Aquí se ilustra el proceso de que partiendo del nombre de un objeto en memoria, se accese su localización en memoria física. Para ello se discuten aspectos como: mapeo de nombres (de tablas de símbolos a tablas *Hash* y jerarquía de tablas); encadenado; segmentación; paginado; memorias caché; y

estructuras de interconexión de la memoria. Los apéndices E y F proporcionan material de referencia para la discusión de este capítulo. El apéndice G presenta las funciones de las memorias asociativas y su implementación.

**CAPÍTULO 4. Control de flujo simple:** Se discuten estructuras básicas para especificar el flujo de control de un programa procesado secuencialmente. Se describen subsecuentemente las arquitecturas *Harvard* y *Von Neuman*; la representación de instrucciones; y la microprogramación. Este material es complementado por el apéndice H, en el que se describen estructuras de control no secuenciales en lenguaje *prolog*.

**CAPÍTULO 5. Procesamiento orientado a objetos:** En este capítulo se tratan tipos de objetos y clases; modularización; y soporte a procesadores y sistemas a modularizar, con base en clases. Este material es complementado por el apéndice I, en el que se cubre el lenguaje *LISP* y el diseño de una máquina que soporta este lenguaje.

**CAPÍTULO 6. Paralelismo en flujo de instrucciones Simple:** Aquí se discute la importancia del paralelismo; las formas de expresar dicho paralelismo; métodos para encontrarlo y agregarlo a los programas; y procesadores de palabra de instrucción muy larga (VLIW). Se discute además, la detección dinámica del paralelismo; y arquitecturas superescalares. Se analiza también, diversas formas de reestructurar programas para ampliar la posibilidad de su ejecución en paralelo. Por otro lado, el apéndice J cubre el tópico de arreglos sistólicos, los cuales pueden procesar algoritmos especiales empleando paralelismo estático.

**CAPÍTULO 7. Paralelismo por paso de mensajes:** Presenta los modelos de paso de mensajes; y sus modelos de programación. Se analiza además, la comunicación entre procesos y entre módulos, así como también las interacciones de entrada/salida. El apéndice K complementa este material en el que se describen sistemas de flujo de datos y su implementación. Asimismo, en el apéndice L se presentan argumentos que prueban la validez de estos modelos.

**CAPÍTULO 8. Sistemas con Recursos Compartidos:** En esta parte se tratan los sistemas de memoria compartida; y los métodos para garantizar sincronización y coherencia en programas y cachés.

**CAPÍTULO 9. Protección y seguridad:** En este último capítulo se presentan diversos esquemas de protección que emplean estructuras de página y segmentos. Se analizan los requerimientos del modelo de seguridad de *Bell* y *Lapadua*, así como el modelo de integridad de *Biba*. Se consideran además, diversas técnicas orientadas al diseño de sistemas seguros.

Cada capítulo está adecuadamente indexado y bien ilustrado, y ofrece un conjunto de problemas de creciente grado

de dificultad. Unos tienen una solución directa, otros problemas sin embargo, presentan objetivos de diseño que requiere que asumamos un papel de consultor de diseño. Es decir, se plantea una propuesta de diseño, y se nos pide que lo critiquemos y que examinemos sus implicaciones en el diseño global.

Los 12 apéndices ocupan una tercera parte del libro y complementan ampliamente lo tratado en capítulos individuales. El tema que trata cada apéndice es el siguiente:

- Apéndice A. Resumen del procesador SPARC.
- Apéndice B. Procesador Alpha AXP.
- Apéndice C. Procesadores MC680x0.
- Apéndice D. Sistemas orientados a pilas: El B 5700 y la HP 3000.
- Apéndice E. Nombrado de objetos en memoria.
- Apéndice F. Asignación de memoria.
- Apéndice G. Memorias asociativas.
- Apéndice H. Prolog.
- Apéndice I. Procesamiento de listas.
- Apéndice J. Arreglos sistólicos.
- Apéndice K. Sistemas de flujo de datos.
- Apéndice L. Razonamiento y validez.

En general el autor logra tratar con éxito, un conjunto de temas complejos con claridad y precisión. Con algunas excepciones (por ejemplo, un tratamiento un tanto débil de los procesadores *Intel*, y casi ninguna discusión sobre lenguajes de programación paralela), este libro contribuye a la disciplina de la computación al presentar dentro de una estructura lógica, los problemas a que se enfrentan los diseñadores de computadoras, y las alternativas de diseño que resuelven dichos problemas.

Para quienes se dediquen a la enseñanza de la computación, vale la pena contar con este texto que no es ni muy simple, ni tan complejo que uno se pierda en el detalle. Su tratamiento de la arquitectura de computadoras, es realmente efectiva.

## REFERENCIAS.

Kain Y. Richard, **Advanced Computer Architecture: a systems design approach**, Prentice Hall, Inc. Englewood Clieffs, New Jersey, 1996, 907 pp.,

