

VNIVERSITAT  VALÈNCIA



UNIVERSITAT DE VALÈNCIA

ARQUITECTURA E INGENIERÍA DE LOS
COMPUTADORES
(Segunda Parte)

INGENIERÍA INFORMÁTICA

Profesor: Fernando Pardo Carpio

Valencia, 27 de septiembre de 2004

Prefacio

El temario recogido en estos apuntes se corresponde con la segunda parte de la asignatura de *Arquitectura e Ingeniería de los Computadores* de la Ingeniería Informática de la Universidad de Valencia. Esta asignatura completa consta de 6 créditos teóricos y 3 prácticos, y se cursa en cuarto.

El contenido principal de la segunda parte de la asignatura es básicamente el estudio de los sistemas multiprocesadores y paralelos de altas prestaciones.

Estos apuntes son un subconjunto de los apuntes de la asignatura de plan viejo *Arquitecturas Avanzadas*. Los contenidos de esta asignatura de plan viejo se han dividido entre dos asignaturas, por un lado la de *Ampliación de Arquitectura de Computadores*, y por el otro en la segunda parte de la asignatura de *Arquitectura e Ingeniería de los Computadores*. Por esta razón existen tres versiones de estos apuntes:

- *Arquitecturas Avanzadas*. Esta asignatura ya no se imparte pero son los apuntes más completos, ya que recogen todos los temas preparados hasta el momento.
- *Ampliación de Arquitectura de Computadores*. Se recogen los temas de vectoriales, matriciales, multicomputadores y otras arquitecturas avanzadas. Se incluye también una introducción al paralelismo como apéndice.
- *Arquitectura e Ingeniería de los Computadores* (segundo cuatrimestre). En esta asignatura se recogen el resto de temas de *Arquitecturas Avanzadas* como son el rendimiento de los sistemas paralelos, redes de interconexión y multiprocesadores.

Estos apuntes se empezaron a preparar en 1998. Posteriormente, a principios del 2000, el profesor Antonio Flores Gil del Departamento de Ingeniería y Tecnología de Computadores de la Facultad de Informática de la Universidad de Murcia, aportó las fuentes de sus mucho más completos apuntes de *Arquitectura de Computadores* con el fin de que se pudiesen adaptar a las necesidades del temario. Con los apuntes que ya se tenían y con los del profesor Antonio Flores, se ha realizado esta nueva versión que todavía está en construcción.

Se incluyen las referencias bibliográficas utilizadas en la elaboración de los apuntes, incluyéndose un pequeño apéndice donde se comentan los diferentes libros utilizados. Con esto se ha pretendido dar una visión global a las arquitecturas avanzadas, ofreciéndole al alumno un abanico amplio de posibilidades para ampliar información.

Fernando Pardo en Valencia a 30 de enero de 2002

Índice

4. Introducción a las arquitecturas paralelas	1
4.1. Clasificación de los sistemas paralelos	2
4.1.1. Clasificación de Flynn	2
4.1.2. Otras clasificaciones	3
4.2. Fuentes del paralelismo	10
4.2.1. El paralelismo de control	11
4.2.2. El paralelismo de datos	11
4.2.3. El paralelismo de flujo	13
5. Multiprocesadores	15
5.1. Coherencia de las cachés	15
5.1.1. El problema de la coherencia de las cachés	15
5.1.2. Direcciones físicas y virtuales, problema del <i>aliasing</i>	18
5.1.3. Soluciones a los problemas de coherencia	19
5.1.4. Esquemas de coherencia de las cachés	21
5.2. Protocolos de sondeo o <i>snoopy</i> (medio compartido)	22
5.2.1. Protocolo de invalidación de 3 estados (MSI)	25
5.2.2. Protocolo de invalidación de 4 estados (MESI)	29
5.2.3. Write once	31
5.2.4. Berkeley	33
5.2.5. Protocolo de actualización de 4 estados (Dragon)	34
5.2.6. Firefly	36
5.2.7. Rendimiento de los protocolos de sondeo	38
5.2.8. El problema de la falsa compartición	39
5.3. Esquemas de coherencia basados en directorio	39
5.3.1. Protocolos basados en directorio	39
5.3.2. Protocolo de mapeado completo	40
5.3.3. Protocolo de directorio limitado	41
5.3.4. Protocolo de directorio encadenado	42
5.3.5. Rendimiento de los protocolos basados en directorio	44
5.4. Modelos de consistencia de memoria	48
5.4.1. Consistencia secuencial	50
5.4.2. Otros modelos de consistencia	54
5.4.3. Implementación de los modelos relajados	58
5.4.4. Rendimiento de los modelos relajados	60
5.5. Sincronización	61
5.5.1. Cerrojos (exclusión mutua)	64
5.5.2. Eventos de sincronización punto a punto (banderas)	73
5.5.3. Eventos globales de sincronización (barreras)	74
5.6. Generalidades sobre las redes de interconexión	78
5.6.1. Definiciones básicas y parámetros que caracterizan las redes de interconexión	79
5.6.2. Topología, control de flujo y encaminamiento	82
5.6.3. Clasificación de las redes de interconexión según su topología	85

5.6.4.	Redes de medio compartido	85
5.6.5.	Redes Directas	88
5.6.6.	Redes Indirectas	89
5.6.7.	Redes Híbridas	89
5.7.	Redes de interconexión para multiprocesadores	94
5.7.1.	Redes de medio compartido. Buses	94
5.7.2.	Redes indirectas	95
5.7.3.	Red de barra cruzada	96
5.7.4.	Redes de interconexión multietapa (MIN)	98
5.7.5.	Tipos de etapas de permutación para MIN	100
5.7.6.	Clasificación de las redes MIN	103
5.7.7.	Encaminamiento en redes MIN	109
5.7.8.	Resumen de las redes indirectas y equivalencias	116
5.8.	Conclusiones	117
6.	El rendimiento de los sistemas paralelos	119
6.1.	Magnitudes y medidas del rendimiento	119
6.1.1.	Eficiencia, redundancia, utilización y calidad	119
6.1.2.	Perfil del paralelismo en programas	120
6.1.3.	Rendimiento medio armónico. Ley de Amdahl	123
6.2.	Modelos del rendimiento del <i>speed-up</i>	126
6.2.1.	Ley de Amdahl, limitación por carga de trabajo fija	127
6.2.2.	Ley de Gustafson, limitación por tiempo fijo	129
6.2.3.	Modelo del <i>speed-up</i> limitado por la memoria fija	131
6.3.	Modelos del rendimiento según la granularidad	134
6.3.1.	Modelo básico: 2 procesadores y comunicaciones no solapadas	135
6.3.2.	Extensión a N procesadores	136
6.3.3.	Otras suposiciones para las comunicaciones	140
A.	Comentarios sobre la bibliografía	147
	Bibliografía	149
	Índice alfabético	151

Índice de figuras

4.1. Clasificación de Flynn de las arquitecturas de computadores.	3
4.2. Clasificación de las arquitecturas paralelas.	4
4.3. El modelo UMA de multiprocesador.	5
4.4. El modelo NUMA de multiprocesador.	6
4.5. El modelo COMA de multiprocesador.	6
4.6. Diagrama de bloques de una máquina de flujo de datos.	8
4.7. Paralelismo de control.	11
4.8. Ejemplo de paralelismo de control.	12
4.9. Paralelismo de datos.	12
4.10. Ejemplo de la aplicación del paralelismo de datos a un bucle.	13
4.11. Paralelismo de flujo.	13
5.1. El problema de la coherencia de cachés con dos procesadores.	16
5.2. Ejemplo del problema de coherencia de las cachés.	17
5.3. Direcciones físicas y virtuales en la caché.	18
5.4. Protocolo básico de invalidación con tres estados.	26
5.5. Protocolo de invalidación de 3 estados en acción para las transacciones mostradas en la figura 5.2.	27
5.6. Orden parcial en las operaciones de memoria para una ejecución con el protocolo MSI.	28
5.7. Diagrama de transición de estados para el protocolo Illinois MESI.	30
5.8. Diagrama de transición del protocolo Write once.	32
5.9. Diagramas de transición del protocolo Berkeley.	34
5.10. Diagrama de transición de estados del protocolo de actualización Dragon.	36
5.11. Diagrama de transición de estados del protocolo de actualización Firefly.	37
5.12. Directorio en el protocolo de mapeado completo.	40
5.13. Protocolo de directorio encadenado en el estándar SCI.	43
5.14. Tasa de fallo en función del número de procesadores.	45
5.15. Tasa de fallo en función del tamaño de la caché.	46
5.16. Tasa de fallo en función del tamaño del bloque suponiendo una caché de 128 KB y 64 procesadores.	46
5.17. Número de bytes por referencia en función del tamaño del bloque.	47
5.18. Características de la máquina basada en directorio del ejemplo.	48
5.19. La latencia efectiva de las referencias a memoria en una máquina DSM depende de la frecuencia relativa de los fallos de caché y de la localización de la memoria desde donde se producen los accesos.	49
5.20. Requisitos de un evento de sincronización mediante flags.	50
5.21. Abstracción desde el punto de vista del programador del subsistema de memoria bajo el modelo de consistencia secuencial.	52
5.22. Órdenes entre los accesos sin sincronización.	52
5.23. Ejemplo que ilustra la importancia de la atomicidad de las escrituras para la consistencia secuencial.	53
5.24. Modelos de consistencia de memoria.	55

5.25. Ordenaciones impuestas por varios modelos de consistencia. Se muestran tanto los accesos ordinarios como los accesos a variables de sincronización.	57
5.26. Ejemplos de los cinco modelos de consistencia vistos en esta sección que muestran la reducción en el número de órdenes impuestos conforme los modelos se hacen más relajados.	58
5.27. Rendimiento de los modelos de consistencia relajados sobre una variedad de mecanismos hardware.	60
5.28. Implementación de (a) un intercambio atómico y (b) una operación de lectura e incremento (<i>fetch-and-increment</i>) utilizando las operaciones <i>load linked</i> y <i>store conditional</i> .	62
5.29. Pasos en la coherencia de la caché y tráfico en el bus para tres procesadores, P0, P1 y P2 que intentan adquirir un cerrojo en protocolo de coherencia por invalidación.	66
5.30. Tiempo para adquirir y liberar un cerrojo cuando 20 procesadores están compitiendo por el mismo.	67
5.31. Un cerrojo con <i>exponential back-off</i> .	68
5.32. Estados de la lista para un cerrojo cuando los procesos intentan adquirirlo y cuando lo liberan.	70
5.33. Rendimiento de los cerrojos en el SGI Challenge para tres posibles escenarios.	71
5.34. Rendimiento de los cerrojos son la máquina SGI Origin2000, para tres escenarios diferentes.	73
5.35. Código de una barrera centralizada.	74
5.36. Código para una barrera sense-reversing.	75
5.37. Rendimiento de algunas barreras en el SGI Challenge.	78
5.38. Ejemplos del cálculo del ancho de la bisección: toro 2-D. (b) toro 3-D (no se muestran los enlaces de cierre).	80
5.39. Ejemplo de empaquetamiento de un multicomputador.	83
5.40. Clasificación de las redes de interconexión. (1-D = unidimensional; 2-D = bidimensional; 3-D = tridimensional; CMU = Carnegie Mellon University; DASH = Directory Architecture for Shared-Memory; DEC = Digital Equipment Corp.; FDDI = Fiber Distributed Data Interface; HP = Hewlett-Packard; KSR = Kendall Square Research; MIN = Multistage Interconnection Network; MIT = Massachusetts Institute of Technology; SGI = Silicon Graphics Inc.; TMC = Thinking Machines Corp.)	86
5.41. Una red con bus único. (M = memoria; P = procesador.)	87
5.42. Arquitectura de un nodo genérico.	89
5.43. Algunas topologías propuestas para redes directas.	90
5.44. Un red conmutada con topología irregular.	90
5.45. Redes Híbridas. (a) Una red multibus. (b) Una jerarquía de dos niveles de buses.	91
5.46. Malla bidimensional basada en clusters.	92
5.47. Una hipermalla bidimensional.	93
5.48. Una hipermalla unidimensional con conmutador crossbar distribuido.	94
5.49. La red de conmutación en barra cruzada.	96
5.50. Estados de un punto de conmutación en una red de barras cruzadas.	98
5.51. Estructura generalizada de una interconexión multietapa (MIN).	99
5.52. Visión detallada de una etapa G_i .	99
5.53. Barajado perfecto, barajado perfecto inverso, y bit reversal para $N = 8$.	101
5.54. La conexión mariposa para $N = 8$.	102

5.55. La conexión cubo para $N = 8$	102
5.56. La conexión en línea base para $N = 8$	103
5.57. Una red Beneš 8×8	104
5.58. Cuatro posibles estados de un conmutador 2×2	104
5.59. Cuatro redes de interconexión unidireccionales multietapa de 16×16	105
5.60. Red de línea base 8×8	106
5.61. Redes basadas en el barajado perfecto.	107
5.62. Conexiones en un conmutador bidireccional.	108
5.63. Una MIN mariposa bidireccional de ocho nodos.	108
5.64. Caminos alternativos para una MIN mariposa bidireccional de ocho nodos.	109
5.65. Árbol grueso y BMIN mariposa.	110
5.66. Vista funcional de la estructura de una red Omega multietapa.	112
5.67. Selección del camino por el algoritmo de encaminamiento basado en etiquetas en una MIN mariposa de 16 nodos.	114
5.68. Caminos disponibles en una BMIN mariposa de ocho nodos.	115
6.1. Perfil del paralelismo de un algoritmo del tipo divide y vencerás.	121
6.2. Media armónica del <i>speed-up</i> con respecto a tres distribuciones de probabilidad: π_1 para la distribución uniforme, π_2 en favor de usar más procesadores y π_3 en favor de usar menos procesadores.	125
6.3. Mejora del rendimiento para diferentes valores de α , donde α es la fracción del cuello de botella secuencial.	126
6.4. Modelos de rendimiento del <i>speed-up</i>	127
6.5. Modelo del <i>speed-up</i> de carga fija y la ley de Amdahl.	130
6.6. Modelo de <i>speed-up</i> de tiempo fijo y la ley de Gustafson.	132
6.7. Modelo de <i>speed-up</i> de memoria fija.	133
6.8. Tiempo de ejecución para dos factores R/C diferentes.	137

