

ENERO 2013

1.Explica la utilidad del buffer de renombrado y enumera los tipos que hay según su direccionamiento

El buffer de renombrado es una estructura que da soporte para poder realizar el renombrado de registros. Esta técnica sirve para poder utilizar técnicas de desordenación que permitan aprovechar mejor la paralización sin riesgos por dependencias WAR y WAW. Cuando se está ejecutando alguna instrucción que tenga un registro como destino de escritura, dicha escritura no se realizara directamente sobre el registro correspondiente, sino que se almacenara el resultado en un buffer auxiliar donde podrá ser consultado por otras instrucciones de lectura.

Existen dos tipos según su direccionamiento:

Acceso asociativo:

- Permite varias escrituras pendientes a un mismo registro
- Utiliza un bit para especificar cual es la ultima escritura

Acceso indexado:

- Solo permite una escritura pendiente a un mismo registro
- Solo se conserva el resultado correspondiente a la escritura mas actual

2.Explica la diferencia entre predicción dinámica explícita y predicción dinámica implícita.

La predicción dinámica significa que la predicción que se va a realizar sobre una instrucción de salto puede variar en cada ejecución.

La diferencia entre la predicción dinámica explícita y la implícita es que en la explícita se utilizan bits de historia que almacenan información sobre las ejecuciones pasadas para las futuras predicciones, mientras que en la implícita, únicamente se almacena la dirección de la instrucción a la que se accedió después de la última ejecución de la instrucción de salto.

3.Justifica la diferencia que existe entre multicomputadores y multiprocesadores en términos de latencia y escalabilidad.

Los multicomputadores tienen una mayor escalabilidad que los multiprocesadores.

Esto se debe a que los multiprocesadores hacen un uso compartido de los recursos del sistema, por lo que cada procesador debe competir por el uso de estos, aparte de que la

comunicación entre ellos se realiza mediante un único bus. Por esta razón, cuantos más procesadores participen en el multiprocesador, mayor latencia existirá.

Los multiprocesadores no tienen este problema ya que cada nodo dispone de sus propios recursos.

4. Cuál es la característica distintiva en las redes de interconexión dinámicas frente a otros tipos de redes de interconexión?

La principal diferencia es que en las redes dinámicas, los enlaces entre nodos (topología) pueden ir variando según la necesidad de cada instante, mientras que en las redes estáticas la topología queda establecida de forma fija cuando se instala el sistema.

JULIO 2013

1. Explica la utilidad de la ventana de instrucciones y explica brevemente cómo puede ser el orden de emisión y el alineamiento de la ventana

La ventana de instrucciones se utiliza para controlar la ejecución de instrucciones de forma paralela. Esta almacena las instrucciones pendientes ya decodificadas y se utiliza un bit para indicar si la instrucción está disponible. Una instrucción estará disponible cuando tenga todos sus operandos disponibles y la unidad funcional donde se procesará.

Alineamiento:

- **Alineada:** No pueden introducirse nuevas instrucciones en la ventana hasta que esta no se vacíe.
- **No alineada:** Se pueden introducir nuevas instrucciones en la ventana en el espacio que vaya quedando libre.

Orden:

- **Ordenada:** Se respeta el orden de las instrucciones y si una instrucción no puede ser emitida por algún motivo, las posteriores no podrán emitirse aunque estén disponibles hasta que la primera se realice.
- **Desordenada:** Si una instrucción no está disponible en un momento determinado, podrá emitirse las siguientes sin que se produzca ningún bloqueo, siempre que estas estén disponibles.

2. Explica brevemente las estructuras que permiten la gestión de predicción de los saltos dinámica explícita.

- **Bits acoplados**

- **Branch Target Buffer (BTB)** es una pequeña memoria asociativa que guarda las direcciones de los últimos saltos ejecutados así como su destino. A su vez guarda información que permite predecir si el salto será tomado o no.

Solo pueden predecirse los saltos que están en la BTB.

- **Bits desacoplados**

- **Branch History Table (BHT)** almacena los bits de predicción de todas las instrucciones de salto condicional.
- **Branch Target Address Cache (BTAC)** almacena las direcciones destino de los últimos saltos ejecutados.

- **l-caché**

- almacena los bits de predicción de la instrucción de salto en la caché. También se añade en la cache un índice sucesor que indica mediante una tabla BTB independiente la dirección destino del salto.

3.Explica brevemente qué tipos de paralelismo existen y en qué niveles puede aplicarse

- **Paralelismo de datos:** una misma función, instrucción, etc. Se ejecuta en paralelo sobre un conjunto de datos distinto en cada una de las ejecuciones.
- **Paralelismo funcional:** Varias funciones, tareas, instrucciones, etc. Iguales o diferentes se ejecutan en paralelo.
 - **Nivel de instrucción (ILP):** las instrucciones de un programa se ejecutan en paralelo. Granulidad fina.
 - **Nivel de bucle o hebra (Thread):** Se ejecutan en paralelo distintas iteraciones de un bucle. Granulidad fina/media.
 - **Nivel de procedimiento (Proceso):** Distintos procedimientos de un programa se ejecutan en paralelo. Granulidad media.
 - **Nivel de programa:** Se ejecutan en paralelo diferentes programas que pueden pertenecer o no a la misma aplicación. Granulidad gruesa.

ENERO 2014

1.Explica brevemente cuáles son los tipos de paralelismo que podemos encontrar en un sistema informático

Paralelismo funcional

Se obtiene a través de la reorganización lógica de una aplicación. Varias funciones, tareas, instrucciones (iguales o distintas) se ejecutan en paralelo. Existen diferentes niveles de paralelismo funcional según las estructuras en que se reorganicen:

- Nivel de instrucción (ILP). Granulidad fina. Se paralelizan instrucciones
- Nivel de bucle o hebra (Thread). Granulidad fina/media. Se paralelizan iteraciones de un bucle o secuencias de instrucciones
- Nivel de procedimiento (Proceso). Granulidad media. Se paralelizan los procedimientos que constituyen el programa
- Nivel de programa. Granulidad gruesa. Se paralelizan distintos programas que pueden pertenecer (o no) a la aplicación

Paralelismo de datos

La misma función, instrucción se ejecuta en paralelo, pero en cada una de esas ejecuciones se aplica sobre un conjunto de datos distinto. Está relacionado con operaciones realizadas sobre grandes volúmenes de datos independientes entre sí. Implícito en operaciones de datos de tipo vector o matriz.

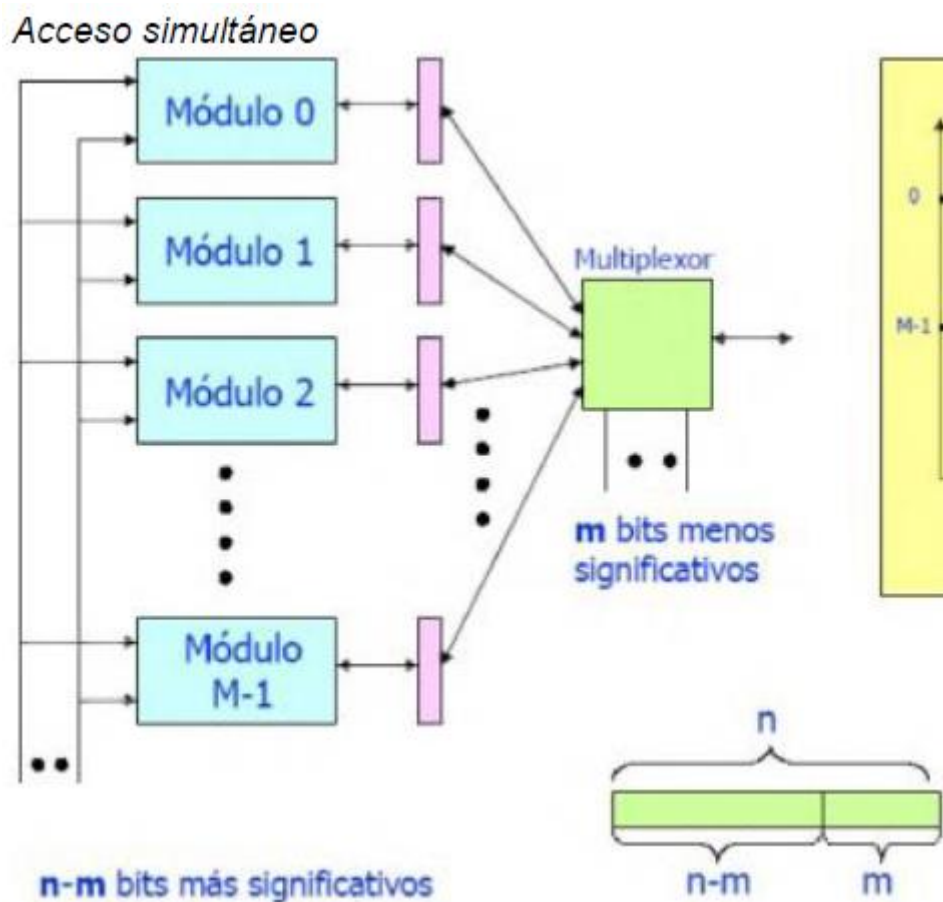
Paralelismo implícito

Paralelismo presente (aunque puede no estar ejecutándose de forma paralela) debido a la propia estructura de los datos (vectores) o de la aplicación (bucles).

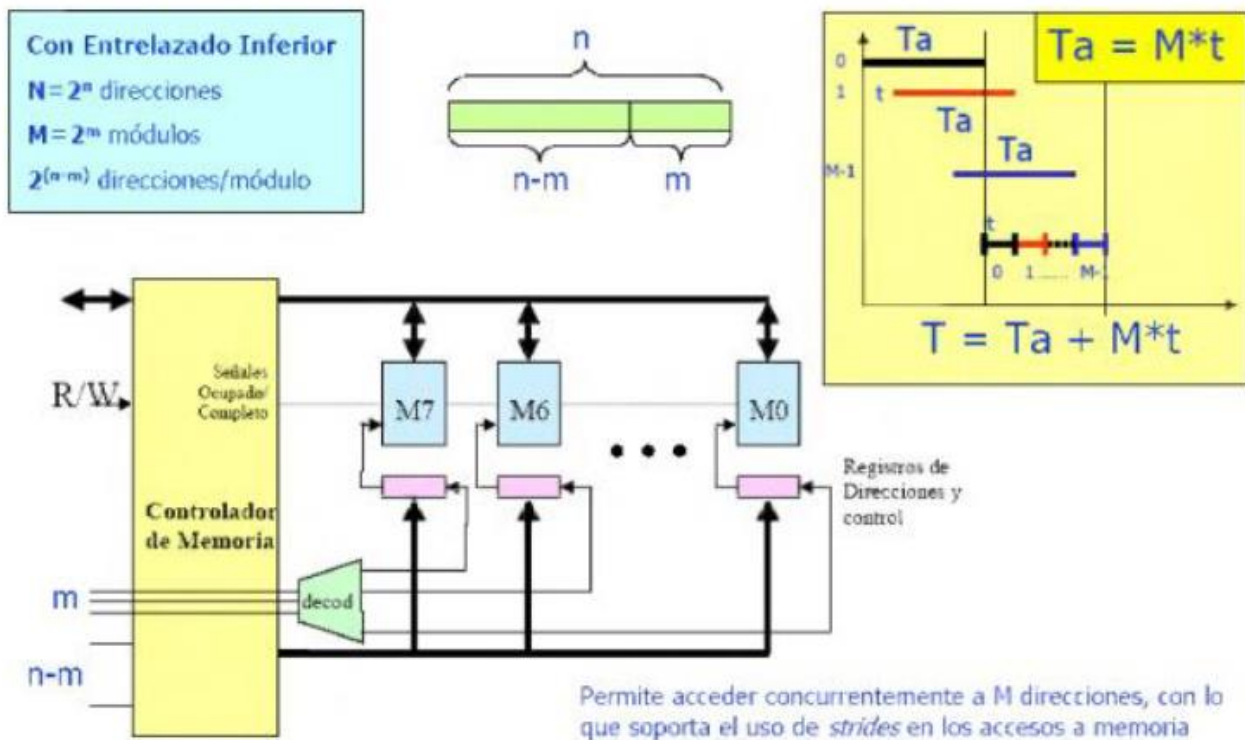
Paralelismo explícito

Paralelismo no presente de forma inherente en las estructuras de programación y que se debe indicar expresamente.

2, ¿Existe alguna diferencia entre el acceso a memoria concurrente y el acceso simultáneo en una máquina vectorial? Si es así, explícala muy brevemente.



Acceso
concurrente



3.¿Cuáles son las similitudes y diferencias entre el buffer de renombrado y el buffer de reorden?

Similitudes

En los procesadores con finalización desordenada, se pueden producir riesgos de dependencias WAR o WAW (Write After Read o Write After Write), y para minimizar el impacto que provocarían en la ejecución del programa haremos uso del renombrado de registros y reorden de los mismos, y dentro de ellos existen dos tipos:

- Implementación Dinámica: Se realizan durante la ejecución y requieren circuitería adicional
- Implementación Estática: Se realiza durante la compilación

Diferencias

Son dos buffers totalmente diferentes y a pesar de que estan hechos para el mismo fin realizan ese fin de forma deferente el buffer de renombrado es como si pusiera mas registros a disposición y el buffer de reorden lo que hace es reordenar las instrucciones para que no se produsca el riezgo war o waw.

4.Explica en qué consiste el procesamiento especulativo de los saltos

Una manera de gestionar los saltos condicionales no resueltos (Una condición de salto no se puede comprobar hasta que se ha evaluado) es mediante el procesamiento especulativo. Mediante esta técnica la ejecución prosigue por el camino más probable (se especula sobre las instrucciones que se ejecutarán), Si te ha errado en la ejecución hay que recuperar el camino correcto. Esta gestión de los saltos condicionales es típica en los procesadores superescalares actuales.

5.

a) Defina y enumere las diferencias entre proceso y una hebra

En la asignación de tareas, normalmente se asignan iteraciones de un ciclo a hebras y funciones a procesos. Los procesos son independientes, tienen un espacio de direcciones virtuales propio, llevan bastante información de estados, e interactúan sólo a través de mecanismos de comunicación dados por el sistema.

Los hilos generalmente comparten otros direcciones virtuales de forma directa.

Se tarda mucho menos tiempo en crear un hilo nuevo en un proceso existente que en crear un proceso.

Se tarda mucho menos en terminar un hilo que un proceso, ya que cuando se elimina un proceso se debe eliminar el BCP1 del mismo, mientras que un hilo se elimina su contexto y pila.

Se tarda mucho menos tiempo en cambiar entre dos hilos de un mismo proceso.

Los hilos aumentan la eficiencia de la comunicación entre programas en ejecución.

b) Hablando de paralelismo ¿Qué tipo de computador paralelo está asociado a cada uno de estos 2 conceptos? ¿Por qué?

(Ni puta idea porque tengo que saber esto ¿?)

c) ¿Puede paralelizar una aplicación usando al mismo tiempo distintos procesos y distintas hebras?

Indique, si es el caso, un ejemplo real de arquitectura que dé soporte a estos niveles de paralelismo

(Enserio en un examen te preguntas nombres de computadores wtf ¿?)

ENERO 2015

1. Explica brevemente que es el paralelismo en computación.

El paralelismo es una forma de computación en la cual varios cálculos pueden realizarse simultáneamente, basado en el principio de dividir los problemas grandes para obtener varios problemas pequeños, que son posteriormente solucionados en paralelo.

¿Siempre mejora el rendimiento?

El paralelismo en computación no siempre mejora el rendimiento, cuando hay cargas computacionales de bajo coste, realizar la paralelización ralentiza el tiempo respecto a la ejecución del programa en secuencial.

2. ¿Para que sirve el buffer de renombrado?

para evitar el efecto de las dependencias WAR, o Antidependencias(en la emisión desordenada) y WAW, o Dependencias de Salida(en la ejecución desordenada)

¿En que etapa o etapas se usa?

Se usa en la etapa de decodificación para leer los registros fuente que esten renombrados y para renombrar el registro destino. Al acabar la ejecución se escriben los resultados en el buffer de renombrado, y al acabar se escriben los resultados en los registros correspondientes

Pon un ejemplo descriptivo sencillo de su funcionamiento.

3. ¿En que consiste la predicción dinámica implícita?

La predicción dinámica quiere decir que cada vez que se predice una misma instrucción de salto la predicción puede ser diferente según la historia previa (si se han tomado o no las anteriores ejecuciones).

Ser implícita quiere decir que no contiene información en forma de bits de historia sino que almacena la dirección de salto de la instrucción que se ejecutó después del salto.

Explica como funciona poniendo un ejemplo.

4. ¿Cual es la principal diferencia entre multicomputadores y multiprocesadores?

La diferencia esencial reside en la memoria, el multiprocesador posee varios procesadores que pueden tener su memoria caché propia pero que al fin y al cabo comparten una misma memoria principal. El multicomputador esta compuesto por varios computadores que pueden o no ser multiprocesadores y que cada uno tiene su memoria principal independiente.

5. Explica brevemente los diferentes tipos de paralelismo que se pueden encontrar

Paralelismo funcional

Se obtiene a través de la reorganización lógica de una aplicación. Varias funciones, tareas, instrucciones (iguales o distintas) se ejecutan en paralelo. Existen diferentes niveles de paralelismo funcional según las estructuras en que se reorganicen:

- Nivel de instrucción (ILP). Granulidad fina. Se paralelizan instrucciones
- Nivel de bucle o hebra (Thread). Granulidad fina/media. Se paralelizan iteraciones de un bucle o secuencias de instrucciones
- Nivel de procedimiento (Proceso). Granulidad media. Se paralelizan los procedimientos que constituyen el programa
- Nivel de programa. Granulidad gruesa. Se paralelizan distintos programas que pueden pertenecer (o no) a la aplicación

Paralelismo de datos

La misma función, instrucción se ejecuta en paralelo, pero en cada una de esas ejecuciones se aplica sobre un conjunto de datos distinto. Está relacionado con operaciones realizadas sobre grandes volúmenes de datos independientes entre sí. Implícito en operaciones de datos de tipo vector o matriz.

Paralelismo implícito

Paralelismo presente (aunque puede no estar ejecutándose de forma paralela) debido a la propia estructura de los datos (vectores) o de la aplicación (bucles).

Paralelismo explícito

Paralelismo no presente de forma inherente en las estructuras de programación y que se debe indicar

expresamente.

6. Explica mediante un ejemplo detallado como funciona el algoritmo de encaminamiento de un hipercubo

7. Explica como se distribuyen los bits necesarios para mantener la coherencia utilizando un protocolo basado en directorios. ¿Para que sirve cada bit?

En el directorio: cada entrada de bloque tiene unos bits de presencia por cada caché y un bit de inconsistencia única

- Bits de presencia: especifica la presencia en las cachés de copias del bloque de memoria
- Bit de inconsistencia única: cuando este bit está activado, sólo uno de los bits presencia está a uno, es decir, sólo existe una caché con la copia de ese bloque o línea, con lo que sólo esa caché tiene permiso para actualizar la línea

Por cada caché:

- Bit de validación (v): indica si la copia es válida o no
- Bit de privacidad (p): indica si la copia tiene permiso de escritura, es decir, cuando este bit es uno entonces es la única copia que existe de esta línea en las cachés y por tanto tiene permiso para escribir

Enero 2016

1. Contesta las siguientes preguntas:

a) Dibuja una red mariposa de 8 entradas y 8 salidas con conmutadores 2x2

(Ni idea quien sepa dibujarla que la dibuje)

b) En que consiste la predicción dinámica implícita. Pon un ejemplo.

La predicción dinámica significa que la predicción que se va a realizar sobre una instrucción de salto puede variar en cada ejecución. En la implícita, únicamente se almacena la dirección de la instrucción a la que se accedió después de la última ejecución de la instrucción de salto.

5. Contesta las siguientes preguntas:

a) Explica en que consiste el encaminamiento de las operaciones en las maquinas vectoriales.

(Ni idea hemos dado esto no encuentro por ningun lado encaminamiento de maquinas vectoriales :()

b) ¿Para que se utilizan las estructuras buffer de renombrado y buffer de reorden de los procesadores superescalares?

para evitar el efecto de las dependencias WAR, o Antidependencias(en la emisión desordenada) y WAW, o Dependencias de Salida(en la ejecución desordenada)