

1)

A - ¿Qué es la jerarquía de memoria?

La jerarquía de memoria es una manera de organizar la memoria basada en distintos niveles que corresponden a un elemento de hardware distinto, cada uno con una capacidad, costo y velocidad de acceso diferente. El objetivo de la misma es tener una gran cantidad de memoria al costo de la memoria más barata y con la velocidad de la memoria más rápida. A medida que más nos alejamos del procesador, las unidades de memoria se vuelven más lentas, pero a su vez su costo por bit disminuye y su capacidad aumenta.

B - ¿Por qué funciona la Jerarquía de memoria?

La jerarquía funciona por dos principios:

- Inclusión: este principio establece que los datos presentes en un determinado nivel deben estar incluidos en los niveles subyacentes.
- Coherencia: este principio establece que los datos presentes en un determinado nivel deben poseer la misma información que las copias de estos en los niveles subyacentes.

2)

A - ¿Qué es la coherencia de datos en un sistema de memoria?

Este principio establece que los datos presentes en un determinado nivel deben poseer la misma información que las copias de estos en los niveles subyacentes.

B - Describa las políticas de escritura de una memoria caché.

Las políticas de escritura de una memoria caché pueden categorizarse dependiendo si al escribir se produce un acierto o un fallo.

Las políticas de escritura en acierto son:

- Escritura inmediata: se escribe el dato requerido en el nivel correspondiente de la memoria caché y en el nivel subyacente de la jerarquía de memoria.
- Post-Escritura: se escribe el dato requerido en el nivel correspondiente de la memoria caché, pero no se modifica el siguiente nivel de memoria. Se establece un bit de "sucio" que indica que el bloque ha sido modificado. Lo que nos permite a la hora de reemplazar ese bloque por otro en la memoria caché, modificarlo en el nivel correspondiente.

Las políticas de escritura en fallo son:

- Write-Allocate: se transfiere el dato desde el nivel de memoria en el que se encuentra hacia la memoria caché y la escritura se realiza en función de la política utilizada en caso de acierto. Utilizada habitualmente con Post-Escritura.
- No-Write Allocate: se modifica el dato directamente en el nivel de memoria en el que se encuentra. Utilizada habitualmente con Escritura inmediata.

3) ¿Qué elementos diferentes determinan un bus de interconexión?

Los elementos que determinan un bus de interconexión son:

- Ancho del bus.
- Latencia
- Arbitraje
- Temporización
- Tipo de bus (D/M)
- Tipo de transferencia (S/P)

4)

A - ¿Qué características tiene una implementación superescalar?

Las principales características que tiene una implementación superescalar son:

- Estrategias de captación simultánea de múltiples instrucciones.
- Lógica para determinar dependencias verdaderas entre valores de registros y mecanismos para comunicar esos valores.
- Mecanismos para iniciar o emitir múltiples instrucciones en paralelo.
- Recursos para la ejecución en paralelo de múltiples instrucciones.
- Mecanismos para entregar el estado del procesador en un orden correcto.

B - ¿Cuál política de emisión/retiro de instrucciones tiene en cuenta las "excepciones precisas"?

La política de emisión desordenada y finalización ordenada. La misma nos permite enviar a ejecución las instrucciones de manera desordenada lo cual maximiza el uso de las unidades de procesamiento y además almacenar los resultados

de manera ordenada (finalización ordenada), lo cual es necesario para administrar precisamente las excepciones.

5)

A - ¿Qué describe la Taxonomía Flynn?

La taxonomía Flynn es una manera de clasificar los distintos tipos de computadores según su nivel de paralelismo, en función de su/s flujo/s de instrucciones y de datos. Las categorías son:

- Single Instruction, Single Data (SISD): Son los típicos monoprocesadores, que procesan un único flujo de datos e instrucciones.

- Single Instruction, Multiple Data (SIMD): Son computadores que procesan un único flujo de instrucciones con múltiples flujos de datos. Un ejemplo de estas son las que utilizan procesadores vectoriales o matriciales.

- Multiple Instruction, Single Data (MISD): se transmite una secuencia de datos a un conjunto de procesadores cada uno de los cuales ejecuta una secuencia de instrucciones diferente. Nunca fue implementado ya que no tiene aplicaciones

prácticas.

- Multiple Instruction, Multiple Data (MIMD): un conjunto de procesadores ejecuta simultáneamente secuencias de instrucciones diferentes con conjuntos de datos diferentes. Un ejemplo de estos son los SMP o los clusters.

B - ¿Qué ventajas potenciales nos brindan los SMP?

Las ventajas potenciales que nos brindan los Multiprocesadores Simétricos son:

- Crecimiento incremental: se pueden agregar procesadores para aumentar las prestaciones.

- Disponibilidad: si un procesador falla, el sistema sigue funcionando correctamente debido a que cada procesador puede realizar las mismas tareas.

- Escalado: en función del número de procesadores que configuran el sistema, se pueden ofrecer una gama de productos con distintos precios y prestaciones.

- Prestaciones: si el trabajo es paralelizable, a mayor número de procesadores, el trabajo se realizará de manera más eficiente.