

Organización de Computadoras

1. Definición: La organización de computadoras se refiere a la estructura operativa y la implementación de sistemas informáticos, centrándose en cómo los componentes de hardware interactúan para ejecutar instrucciones.
2. Evolución Histórica: Rastrear el desarrollo desde las primeras computadoras mecánicas hasta los procesadores multicore modernos.
3. Arquitectura de Von Neumann: Un modelo fundamental en el que la CPU, la memoria y la E/S están interconectadas a través de un bus.
4. Arquitectura de Harvard: Separa las vías de almacenamiento y señalización para instrucciones y datos, mejorando el rendimiento.
5. Componentes de la CPU: Incluye la Unidad Lógica Aritmética (ALU), la Unidad de Control (CU) y los Registros.
6. Funciones de la ALU: Realiza operaciones aritméticas y lógicas como la adición, la sustracción, AND, OR.
7. Rol de la Unidad de Control: Dirige la operación del procesador decodificando instrucciones y generando señales de control.
8. Registros: Pequeñas ubicaciones de almacenamiento rápidas dentro de la CPU utilizadas para mantener datos e instrucciones temporalmente.
9. Memoria Caché: Memoria de alta velocidad ubicada cerca de la CPU para reducir el tiempo de acceso a los datos.
10. Jerarquía de Memoria: Organiza la memoria en niveles basados en velocidad y costo, incluyendo registros, caché, RAM y almacenamiento secundario.
11. RAM (Memoria de Acceso Aleatorio): Memoria volátil utilizada para almacenar datos y código de máquina actualmente en uso.
12. ROM (Memoria de Solo Lectura): Memoria no volátil utilizada para almacenar firmware y las instrucciones de arranque del sistema.
13. Estructura de Bus: Un sistema de comunicación que transfiere datos entre componentes dentro o fuera de una computadora.
14. Bus de Datos: Transporta los datos reales que se están procesando.
15. Bus de Direcciones: Transporta información sobre dónde deben enviarse o recuperarse los datos.
16. Bus de Control: Transporta señales de control desde la CPU a otros componentes.
17. Arquitectura del Conjunto de Instrucciones (ISA): Define el conjunto de instrucciones que una CPU puede ejecutar.

18. RISC (Computación de Conjunto de Instrucciones Reducido): Una filosofía de diseño de ISA que utiliza un conjunto pequeño y altamente optimizado de instrucciones.
19. CISC (Computación de Conjunto de Instrucciones Complejo): Una ISA con un gran conjunto de instrucciones, algunas de las cuales pueden ejecutar tareas complejas.
20. Pipelining: Una técnica en la que se superponen múltiples fases de instrucciones para mejorar el rendimiento de la CPU.
21. Etapas del Pipeline: Incluyen típicamente Fetch, Decode, Execute, Memory Access y Write Back.
22. Peligros en el Pipelining: Problemas como peligros de datos, peligros de control y peligros estructurales que pueden interrumpir el flujo del pipeline.
23. Predicción de Ramas: Un método para adivinar la dirección de las instrucciones de rama para mantener el pipeline lleno.
24. Arquitectura Superscalar: Permite que múltiples instrucciones se procesen simultáneamente en una sola etapa del pipeline.
25. Procesamiento Paralelo: Utiliza múltiples procesadores o núcleos para ejecutar instrucciones de manera concurrente.
26. Procesadores Multinúcleo: CPUs con múltiples núcleos de procesamiento integrados en un solo chip.
27. SIMD (Instrucción Única, Datos Múltiples): Una arquitectura de procesamiento paralelo donde una sola instrucción opera sobre múltiples puntos de datos simultáneamente.
28. MIMD (Instrucciones Múltiples, Datos Múltiples): Una arquitectura paralela donde múltiples procesadores ejecutan diferentes instrucciones en diferentes datos.
29. Gestión de Memoria: Técnicas para gestionar y asignar memoria de manera eficiente, incluyendo paginación y segmentación.
30. Memoria Virtual: Extiende la memoria física al almacenamiento en disco, permitiendo que los sistemas manejen cargas de trabajo más grandes.
31. Paginación: Divide la memoria en páginas de tamaño fijo para simplificar la gestión de memoria y reducir la fragmentación.
32. Segmentación: Divide la memoria en segmentos de tamaño variable basados en divisiones lógicas como funciones o estructuras de datos.
33. Técnicas de Mapeo de Caché: Incluyen cachés directamente mappeadas, completamente asociativas y set-asociativas.
34. Políticas de Reemplazo de Caché: Determina qué entrada de caché reemplazar, como la menos recientemente utilizada (LRU) o la primera en entrar, primera en salir (FIFO).

35. Coherencia de Caché: Asegura la consistencia de los datos almacenados en múltiples cachés en un sistema multiprocesador.
36. Modelos de Consistencia de Memoria: Define el orden en el que las operaciones parecen ejecutarse para mantener la consistencia del sistema.
37. Sistemas de Entrada/Salida: Gestiona la comunicación entre la computadora y los dispositivos externos.
38. Clasificación de Dispositivos de E/S: Incluye dispositivos de entrada, dispositivos de salida y dispositivos de almacenamiento.
39. Interfaces de E/S: Estándares como USB, SATA y PCIe que definen cómo los dispositivos se comunican con la placa madre.
40. Acceso Directo a Memoria (DMA): Permite que los dispositivos transfieran datos a/desde la memoria sin intervención de la CPU.
41. Interrupciones: Señales que notifican a la CPU de eventos que requieren atención inmediata, permitiendo el procesamiento asíncrono.
42. Manejo de Interrupciones: El proceso mediante el cual la CPU responde a interrupciones, incluyendo la conservación del estado y la ejecución de rutinas de servicio de interrupciones.
43. Controladores de DMA: Componentes de hardware que gestionan operaciones de DMA, liberando a la CPU de tareas de transferencia de datos.
44. Controladores de Dispositivos: Software que permite que el sistema operativo se comunique con dispositivos de hardware.
45. Interconexión de Componentes Periféricos (PCI): Un estándar para conectar periféricos a la placa madre.
46. Comunicación Serial vs. Paralela: La comunicación serial envía datos un bit a la vez, mientras que la comunicación paralela envía múltiples bits simultáneamente.
47. Puertos Seriales: Interfaces como RS-232 utilizadas para la comunicación serial con dispositivos.
48. Puertos Paralelos: Interfaces utilizadas para la comunicación paralela, a menudo con impresoras y otros periféricos.
49. Arbitraje de Bus: El proceso de gestionar el acceso al bus entre múltiples dispositivos para evitar conflictos.
50. Buses del Sistema vs. Buses Periféricos: Los buses del sistema conectan la CPU, la memoria y los componentes principales, mientras que los buses periféricos conectan dispositivos externos.
51. Tabla de Vectores de Interrupción: Una estructura de datos utilizada para almacenar las direcciones de las rutinas de servicio de interrupciones.

52. Controladores de Interrupción Programables: Hardware que gestiona múltiples solicitudes de interrupción y las prioriza.
53. Ancho de Bus: El número de bits que pueden transmitirse simultáneamente sobre un bus.
54. Velocidad de Reloj: La tasa a la que una CPU ejecuta instrucciones, medida en GHz.
55. Ciclo de Reloj: La unidad de tiempo básica en la que una CPU puede realizar una operación básica.
56. Desfase de Reloj: Diferencias en los tiempos de llegada de la señal de reloj en diferentes partes del circuito.
57. Distribución de Reloj: El método de entrega de la señal de reloj a todos los componentes en la CPU.
58. Disipación de Calor: El proceso de eliminar el exceso de calor de la CPU para evitar el sobrecalentamiento.
59. Soluciones de Enfriamiento: Incluyen disipadores de calor, ventiladores y sistemas de enfriamiento líquido utilizados para gestionar las temperaturas de la CPU.
60. Unidades de Alimentación (PSUs): Proporcionan la energía necesaria a todos los componentes de la computadora.
61. Reguladores de Voltaje: Aseguran que se entreguen niveles de voltaje estables a la CPU y otros componentes.
62. Arquitectura de la Placa Madre: La principal placa de circuito que alberga la CPU, la memoria y otros componentes críticos.
63. Chipsets: Grupos de circuitos integrados que gestionan el flujo de datos entre la CPU, la memoria y los periféricos.
64. Firmware: Software permanente programado en una memoria de solo lectura que controla las funciones de hardware.
65. BIOS/UEFI: Interfaces de firmware que inicializan el hardware durante el proceso de arranque y proporcionan servicios en tiempo de ejecución.
66. Proceso de Arranque: La secuencia de operaciones que inicializa el sistema cuando se enciende.
67. Etapas del Pipeline de Instrucciones: Incluyen típicamente Fetch, Decode, Execute, Memory Access y Write Back.
68. Profundidad del Pipeline: El número de etapas en un pipeline, afectando el rendimiento y la latencia de las instrucciones.
69. Equilibrio del Pipeline: Asegurar que cada etapa tenga un tiempo de ejecución aproximadamente igual para maximizar la eficiencia.
70. Peligros de Datos: Situaciones en las que las instrucciones dependen de los resultados de instrucciones anteriores en un pipeline.

71. Peligros de Control: Ocurren debido a instrucciones de rama que interrumpen el flujo del pipeline.
72. Peligros Estructurales: Suceden cuando los recursos de hardware son insuficientes para soportar todas las posibles ejecuciones de instrucciones simultáneamente.
73. Reenvío (Bypass de Datos): Una técnica para reducir los peligros de datos ruteando datos directamente entre etapas del pipeline.
74. Parada (Burbuja del Pipeline): Insertar ciclos de inactividad en el pipeline para resolver peligros.
75. Ejecución Fuera de Orden: Ejecutar instrucciones a medida que los recursos se vuelven disponibles en lugar de en el orden del programa original.
76. Ejecución Especulativa: Ejecutar instrucciones antes de saber si se necesitan, para mejorar el rendimiento.
77. Algoritmos de Predicción de Ramas: Técnicas como la predicción estática, la predicción dinámica y la predicción adaptativa de dos niveles utilizadas para adivinar direcciones de ramas.
78. Paralelismo a Nivel de Instrucción (ILP): La capacidad de ejecutar múltiples instrucciones simultáneamente dentro de un solo ciclo de CPU.
79. Desenrollamiento de Bucles: Una técnica de optimización que aumenta el cuerpo de los bucles para disminuir la sobrecarga del control del bucle.
80. Superpipelining: Aumentar el número de etapas del pipeline para permitir velocidades de reloj más altas.
81. VLIW (Palabra de Instrucción Muy Larga): Una arquitectura que permite que múltiples operaciones se codifiquen en una sola palabra de instrucción.
82. EPIC (Computación de Instrucciones Paralelas Explícitas): Una arquitectura que habilita la ejecución paralela de instrucciones a través de la asistencia del compilador.
83. Renombrado de Registros: Una técnica para eliminar dependencias de datos falsas asignando dinámicamente registros.
84. Hiper-Threading: La tecnología de Intel que permite que un solo núcleo de CPU ejecute múltiples hilos simultáneamente.
85. Niveles de Memoria Caché: L1 (más cercana a la CPU, más rápida), L2 y L3 caches con tamaño y latencia creciente.
86. Cachés de Escritura a Través vs. Cachés de Escritura por Retraso: La escritura a través actualiza tanto la caché como la memoria simultáneamente, mientras que la escritura por retraso actualiza solo la caché y difiere las actualizaciones de memoria.
87. Asociatividad en Cachés: Determina cómo las líneas de caché se mapan a conjuntos de caché, afectando las tasas de acierto y los tiempos de acceso.

88. Prefetching: Cargar datos en la caché antes de que se soliciten realmente para reducir la latencia de acceso.
89. Patrones de Acceso a Memoria: Acceso secuencial vs. aleatorio y su impacto en el rendimiento de la caché.
90. NUMA (Acceso a Memoria No Uniforme): Un diseño de memoria donde el tiempo de acceso a la memoria varía según la ubicación de la memoria relativa a un procesador.
91. SMP (Multiprocesamiento Simétrico): Un sistema donde múltiples procesadores comparten una sola memoria centralizada.
92. Sistemas de Memoria Distribuida: Sistemas donde cada procesador tiene su propia memoria privada, comunicándose a través de una red.
93. Redes de Interconexión: La topología y los protocolos utilizados para conectar múltiples procesadores y unidades de memoria.
94. Escalabilidad: La capacidad de un sistema informático para aumentar el rendimiento añadiendo más recursos.
95. Tolerancia a Fallos: La capacidad de un sistema para continuar operando correctamente en caso de fallo de algunos de sus componentes.
96. Redundancia: Incorporar componentes adicionales para aumentar la fiabilidad y disponibilidad.
97. Detección y Corrección de Errores: Técnicas como bits de paridad, sumas de verificación y ECC (Código de Corrección de Errores) para identificar y corregir errores de datos.
98. Eficiencia Energética: Diseñar sistemas para minimizar el consumo de energía mientras se mantiene el rendimiento.
99. Potencia Térmica de Diseño (TDP): La cantidad máxima de calor que se espera que una CPU o GPU genere bajo cargas de trabajo típicas.
100. Tendencias Futuras: Explorar avances como la computación cuántica, las arquitecturas neuromórficas y los procesadores fotónicos que están dando forma al futuro de la organización de computadoras.