

Expresión general

Desafío

Preguntas

Arquitectura de Von Neumann

En una **máquina de Von Neumann**, entonces, aparecen dos componentes básicos fundamentales que son la CPU y la memoria,

Arquitectura de Von Neumann

la primera conteniendo una **Unidad de Control, o UC**, para realizar el **ciclo de instrucción**,

Arquitectura de Von Neumann

y una **Unidad Lógico-Aritmética, o ALU**, para el cómputo.

Arquitectura de Von Neumann

En la máquina existen diferentes clases de buses para interconectar los componentes: **buses internos** de la CPU para comunicar la UC y la ALU,

Arquitectura de Von Neumann

buses de sistema que relacionan la CPU y la memoria,

Arquitectura de Von Neumann

y otros **buses de Entrada/Salida** para comunicar todo el sistema con los dispositivos de entrada o de salida.

¿En qué momento se utilizará cada clase de bus? Cuando la UC disponga que se debe ejecutar una instrucción, tal como una suma, enviará los datos de partida, y la instrucción, a la ALU a través de un bus interno. Si la ALU necesita más datos, los obtendrá de la memoria a través de un bus de sistema. Si la CPU encuentra instrucciones que ordenan presentar el resultado del cómputo al usuario, usará un bus de Entrada/Salida para emitir ese resultado por pantalla o por impresora.

Traza de ejecución

Este programa en particular presenta la traza que veremos a continuación.

Queda como ejercicio seguir la traza e interpretar qué está ocurriendo en cada momento con cada uno de los registros y las posiciones de memoria.

Ayuda

- 000 No operación
- 001 Parada
- 010 Mem \rightarrow Ac
- 011 Ac \rightarrow Mem
- 100 Sumar al Ac
- 101 Restar al Ac
- 110 Salto
- 111 Salto cond

Intel I7

El microprocesador I7 es actualmente el procesador más avanzado para computadoras personales de la firma Intel. Es retrocompatible con toda la línea de procesadores de la **arquitectura x86** desarrollada por esa firma. La cronología siguiente muestra algunos significativos cambios de microarquitectura.

- 1979: 8086, 8088, primeros microprocesadores de arquitectura x86
- 1980: 8087, **coprocesador numérico**
- 1982: 80286, multitarea, **modo protegido**
- 1985: 80386, procesador **de 32 bits**
- 1989: 80486, coprocesador numérico en el mismo circuito integrado
- 1993: Pentium, procesador **superescalar**
- 1995: Pentium Pro, procesador con **ejecución fuera de orden y ejecución especulativa**
- 1997: Pentium II, procesador que incorpora **instrucciones vectoriales MMX**
- 1999: Pentium III, incorpora **instrucciones vectoriales SSE**
- 2000: Pentium IV, mejoras en las instrucciones SSE
- 2006: Core 2, nueva microarquitectura, reducción del consumo, múltiples núcleos, **virtualización** en hardware, menores velocidades de reloj
- 2010: Core i3, i5, i7, procesadores con varias microarquitecturas en evolución; presentan desde 2 hasta 12 núcleos, velocidad de reloj variable

Intel I7

El i7 pertenece a una generación de procesadores donde, para enfrentar los problemas derivados de la microminiaturización, los diseñadores optaron por **replicar**, es decir, incorporar múltiples instancias de, las unidades de cómputo o **núcleos**.

- Cada uno de los núcleos, a su vez, puede ejecutar dos secuencias de programa independientes (dos **threads** o **hilos**).
- Cada núcleo tiene su memoria cache privada, dividida en cache de datos y de instrucciones, y además existe un segundo nivel de cache privada para datos e instrucciones a la vez.

Intel I7

- Además existe un tercer nivel de memoria cache compartida, donde se ubican datos que pueden ser necesitados por cualquiera de los núcleos.

Intel I7

- El procesador integra unidades para controlar la consistencia de la memoria interna, y para regular las diferentes actividades dentro del microchip a fin de mantener controlados el uso de energía y la generación de calor.

Intel I7

- En la misma “pastilla” o unidad física del microprocesador se encuentra una unidad procesadora de gráficos o GPU. Esta GPU es un procesador con una arquitectura especial, dedicado a la generación de gráficos avanzados, pero que además puede utilizarse para cálculos paralelos de propósito general.

Swapping o intercambio

Supongamos un sistema donde existen dos procesos activos, con algunas páginas en memoria principal, y una zona de intercambio en disco.

El proceso P1 tiene asignadas cuatro páginas (de las cuales sólo la página 2 está presente en memoria principal), y P2, dos páginas (ambas presentes). Hay tres marcos libres (M4, M6 y M7) y la zona de intercambio está vacía.

Swapping o intercambio

P1 recibe la CPU y en algún momento ejecuta una instrucción que hace una referencia a una posición dentro de su página 3 (que no está en memoria).

Swapping o intercambio

Ocorre una falta de página que trae del almacenamiento la página 3 de P1 a un marco libre. La página 3 se marca como presente en la tabla de páginas de P1.

Swapping o intercambio

Enseguida ingresa P3 al sistema y comienza haciendo una referencia a su página 2.

Swapping o intercambio

Como antes, ocurre una falta de página, se trae la página 2 de P3 del disco, y se copia en un marco libre. Se marca la página 2 como presente y P3 continúa su ejecución haciendo una referencia a una dirección que queda dentro de su página 3.

Swapping o intercambio

Se resuelve como siempre la falta de página para la página 3 y P3 hace una nueva referencia a memoria, ahora a la página 4.

Swapping o intercambio

Pero ahora la memoria principal ya no tiene marcos libres. Es el momento de elegir una página víctima para desalojarla de la memoria. Si la página menos recientemente usada es la página 2 de P1, es una buena candidata. En caso de que se encuentre modificada desde que fue cargada en memoria, se la copia en la zona de intercambio para no perder esas modificaciones, y se declara libre el marco M2 que ocupaba.

Swapping o intercambio

Se marca como no presente la página que acaba de salir de la memoria principal.

Swapping o intercambio

Se copia la página que solicitó P3 en el nuevo marco libre, se la marca como presente en la tabla de páginas de P3, y el sistema continúa su operación normalmente.

Estados de los procesos

Estados de los procesos

Estados de los procesos

Estados de los procesos

Procesos concurrentes

Procesos concurrentes

Procesos concurrentes

Procesos concurrentes

Procesos concurrentes

Procesos concurrentes

Procesos concurrentes

Procesos concurrentes

Procesos paralelos

Procesos

Procesos

Comando top

Memorias

Composición de color

Parámetros de onda

Cálculo de punto flotante

Tabla de códigos ASCII

Datos de la imagen

Datos de la imagen

Routers

Clementina

universidad argentina decidió hacer una importante inversión, que fue la compra de una computadora de primera generación, bautizada aquí **Clementina**. El video adjunto cuenta interesantes detalles técnicos de la computadora, muestra cómo eran las personalidades involucradas por ese entonces en el proyecto científico y tecnológico argentino, y explica el contexto histórico en el que fue iniciado (y, lamentablemente, truncado) ese proyecto.

Traductores

Una ventaja comparativa de la compilación respecto de la interpretación es la mayor velocidad de ejecución. Al separar las fases de traducción y ejecución, un compilador alcanza la máxima velocidad de ejecución posible en un procesador dado. Por el contrario, un intérprete alterna las fases de traducción y ejecución, por lo cual la ejecución completa del mismo programa tardará algo más de tiempo.

Inversamente, el código interpretado presenta la ventaja de ser directamente portable. Dos plataformas diferentes podrán ejecutar el mismo programa interpretable, siempre que cuenten con intérpretes para el mismo lenguaje. Por el contrario, un programa compilado está en código máquina para alguna arquitectura específica, así que no será compatible con otras.

Referencias

Este emulador de PC construido en Javascript nos permite practicar los comandos del shell dentro de una *máquina virtual* Linux y desde el navegador, sin necesidad de una instalación completa en nuestro equipo.

- Para retroceder en el terminal
- Teclas Ctrl-Up, Ctrl-Down, Ctrl-PageUp y Ctrl-PageDown.
- Para copiar datos a la máquina virtual
- Copiar el texto a la caja o **clipboard** a la derecha.
- En el shell de la máquina virtual escribir: `cat < /dev/clipboard > /tmp/archivo`
- Ahora tenemos ese mismo texto en el archivo `/tmp/archivo`.
- Para extraer datos de la máquina virtual
- Invertir el procedimiento anterior: `cat mi_archivo > /dev/clipboard`
- Seleccionar el texto en la caja **clipboard** y copiarlo a alguna otra aplicación en nuestro equipo, tal como un editor.

Diapositiva 559

Generación de PDF