

Paralelismo en Sistemas Mono Procesador

Temas: Conceptos, Buses, Bus de Dirección, Bus de Datos, Bus de Control

Paralelismo en Sistemas Monoprocesador

El paralelismo en sistemas monoprocesador es la capacidad de un procesador con una sola unidad de control para ejecutar múltiples operaciones de forma simultánea o solapada, gracias a la organización interna del hardware y a técnicas de optimización de ejecución.

Tipos de paralelismo interno:

- **Paralelismo a nivel de instrucción (ILP – Instruction Level Parallelism):**
Se ejecutan varias instrucciones al mismo tiempo usando técnicas como *pipeline* y ejecución fuera de orden.
- **Paralelismo de datos:**
Operar sobre múltiples elementos de datos a la vez (instrucciones vectoriales).
- **Paralelismo funcional:**
Uso simultáneo de varias unidades funcionales de la CPU (ALU, FPU, etc.).

Ventajas:

- Mayor rendimiento sin aumentar el número de procesadores.
- Mejor aprovechamiento de recursos internos.
- Reducción del tiempo de ejecución de programas.

Ejemplo:

Un procesador con *pipeline* puede estar decodificando una instrucción mientras ejecuta otra y accede a memoria para la siguiente.

Concepto de Bus

Un bus es el sistema de interconexión que permite transferir información entre las diferentes partes de una computadora. Está compuesto por múltiples líneas o conductores, cada una diseñada para transportar señales específicas.

Características generales:

- **Velocidad de transferencia:** Depende de la frecuencia de reloj y del ancho del bus.
- **Capacidad:** Determinada por el número de líneas (bits) que lo conforman.
- **Sincronización:** Puede ser síncrono (controlado por una señal de reloj común) o asíncrono.

Clasificación de buses según su función:

1. **Bus de Dirección** – Indica dónde se encuentran los datos o instrucciones.
2. **Bus de Datos** – Transporta la información.
3. **Bus de Control** – Coordina las operaciones.

Bus de Dirección

- **Definición:** Es un conjunto de líneas utilizado por la CPU para indicar la dirección de memoria o de un puerto de E/S con el que desea comunicarse.
- **Unidireccional:** La información fluye desde la CPU hacia la memoria o dispositivo de E/S.
- **Capacidad de direccionamiento:** Determinada por el número de líneas.
Ejemplo:
 - Un bus de dirección de 32 bits $\rightarrow 2^{32}$ direcciones posibles \rightarrow 4 GB de memoria direccionable.
 - Un bus de 64 bits $\rightarrow 2^{64}$ direcciones posibles \rightarrow 16 exabytes.
- **Importancia:** Cuantas más líneas, mayor será la memoria accesible.

3.4 Bus de Datos

- **Definición:** Canal por el que se transfieren datos entre CPU, memoria y periféricos.
- **Bidireccional:** Puede enviar y recibir datos.
- **Ancho del bus:** Determina cuántos bits se pueden transferir en un solo ciclo de reloj.
 - Un bus de 8 bits transfiere 1 byte por ciclo.
 - Un bus de 64 bits transfiere 8 bytes por ciclo.
- **Impacto en el rendimiento:** Un mayor ancho de bus permite transferir más datos simultáneamente.

3.5 Bus de Control

- **Definición:** Conjunto de señales usadas para coordinar y administrar el uso del bus de datos y de dirección.
- **Funciones:**
 - Indicar operaciones (lectura, escritura).
 - Gestionar interrupciones.
 - Sincronizar transferencias.
- **Señales comunes:**
 - RD (Read) → Leer datos.
 - WR (Write) → Escribir datos.
 - CLK (Clock) → Señal de sincronización.
 - RESET → Reinicio del sistema.
 - IRQ (Interrupt Request) → Solicitud de interrupción.

Interacción entre los tres buses

Ejemplo de **lectura desde memoria**:

1. CPU coloca la dirección en el **bus de dirección**.
2. CPU envía señal de lectura en el **bus de control**.
3. Memoria coloca el dato solicitado en el **bus de datos**.
4. CPU recibe el dato y lo almacena en un registro.

En resumen, se considera un bus como un conjunto de conductores eléctricos, pistas en una tarjeta de circuito impreso o canales ópticos que permiten la transferencia de información entre los componentes del sistema: CPU, memoria y dispositivos de entrada/salida (E/S).

La eficiencia de un sistema informático depende en gran parte del diseño y velocidad de sus buses, así como de la capacidad del procesador para aprovechar técnicas de paralelismo interno.

Referencias

- Quiroga, P. (2010). *Arquitectura de Computadoras*. Alfaomega.
- Moreno Pérez, J. C., & Serrano Pérez, J. (2014). *Fundamentos del hardware*. Ra-Ma.