Paralelismo em Hardware

Paralelismo em nível de instrução

- **Pipeline dentro da instrução:** dividir a instrução em etapas, fazendo com que mais de uma instrução possa ser executada em um mesmo instante de tempo.

  Exemplo: arquitetura MIPS uma instrução é dividia em 5 etapas (busca, decodificação, execução, escrita em memória e escrita nos registradores), assim pode ser utilizada a pipeline para efetuar paralelismo.

  Implicações e consequências: Pipelines podem diminuir a performance por impedir que tarefas subsequentes sejam executadas no seu tempo correto, devido a interdependência entre as instruções e os dados utilizados, como por exemplo, duas instruções sendo executadas paralelamente precisam utilizar o mesmo recurso de hardware.

- **Very Long Instruction Word:** é uma arquitetura na qual um compilador divide as instruções do programa em operações básicas que podem ser executadas pelo processador de forma paralela, essas operações são colocadas em uma instrução muito longa, em que o processador pode desmontá-la e entregar cada operação a uma unidade funcional apropriada.

  Exemplo: Considere a instrução y= x1z1+x2z2
  *Num processador sequencial:
  Ciclo 1: carrega x1.
  Ciclo 2: carrega z1.
  Ciclo 3: carrega x2.
  Ciclo 4: carrega z2.
  Ciclo 5: multiplica x1 e z1 e armazena em w1.
  Ciclo 6: multiplica x2 e z2 e armazena em w2.
  Ciclo 7: Soma w1 e w2 e armazena em y.

  *Num processador que utiliza Very Long Instruction Word (com duas unidades de armazenamento, uma de multiplicação e uma de soma).

  Ciclo 1: carrega x1, carrega z1.
  Ciclo 2: carrega x2, carrega z2, multiplica x1 e z1 e armazena em w1.
  Ciclo 3: multiplica x2 e z2 e armazena em w2, soma w2 e w1 e armazena em y.

  Implicações e consequências: esta arquitetura possui uma grande dependência, sendo difícil existir compatibilidade entre máquinas que utilizam Very Long Instruction Word. Caso a taxa de operações por instrução for baixa, ocorrerá um mal uso da memória.

Paralelismo de dados

- **SIMD:** é uma arquitetura em que todas as unidades paralelas compartilham a mesma instrução, mas a realizam em diferentes elementos de dados.

  Exemplo: Considere os arrays A= [1, 2, 3 ,4] e B= [4, 3, 2, 1].
  Pode-se somar os dois para obter C=[5, 5, 5, 5] e, para isso deve haver quatro unidades aritméticas em trabalho e compartilhando a mesma instrução (neste caso, seria a soma).

  Implicações e consequências: é uma arquitetura muito eficaz para grande conjunto de dados como vetores e matrizes, pois divide essas estruturas em subconjuntos para realizar as operações e depois combinar os resultados.

Paralelismo de processador

- **Multiprocessadores:** arquitetura com um conjunto de processadores independentes que possuem acesso ao mesmo espaço de endereços na memória e efetuam múltiplas instruções sobre múltiplos dados.

  Exemplo: considerando múltiplas instruções de um programa, neste tipo de arquitetura, poderiam ser efetuadas múltiplas instruções em paralelo ou até mesmo serem efetuas múltiplas instruções de múltiplos programas, já que os processadores são independentes.

  Implicações e consequências: grande quantidade de processadores pode se tornar inviável devido ao superaquecimento e limitação de clock.

Paralelismo de memória

Memória compartilhada: compartilhamento da memória entre os elementos de processamento num único espaço de endereçamento.

Exemplo: considere um programa em que variáveis que possuem o escopo local, ou seja, são executadas dentro de uma função, precisam ser acessadas num escopo global, pode ser efetuado o compartilhamento de memória para que seja feito o acesso dessas variáveis fora da função.

Implicações e consequências: é necessário que o programador saiba utilizar os recursos de memória compartilhada e programação paralela para que este método seja eficaz.

Paralelismo de comunicação

- **Crossbar SMP:** replica o bus de memória para cada processador e controlador de I/O, ou seja, cada processador possui um caminho direto para a memória.

  Exemplo: já que possui não possui links compartilhados, terá uma largura de banda que escala linearmente.

  Implicações e consequências: pode possuir problemas de coerência de cache e possui um alto custo incremental, ou seja, não é viável para muitos processadores.