Arquitectura de Computadores

Ano lectivo 2009/2010 – Aula prática nº 9

Objectivos da sessão

Pretende-se complementar a informação fornecida nas aulas teóricas sobre a memória cache.

1 - O simulador de cache Camera

O simulador Camera

O simulador Camera é um programa escrito em Java. O código fonte está disponível e pode ser executado em diferentes plataformas

Para instalar, crie uma directoria para este efeito, chamada por exemplo *Camera*. Faça *cd* para esse directoria e copie para lá o ficheiro camera.zip. A seguir faça:

```
unzip camera.zip
```

Para compilar faça:

```
javac *.java
```

Para executar o simulador faça:

```
java Camera
```

O simulador está descrito neste manual (PDF). A configuração hardware simulada é a seguinte:

- A memória central tem 256 palavras organizadas em blocos ou linhas de 8 palavras. O endereço emitido pelo CPU tem, assim, 8 bits.
- A cache tem 16 blocos (ou linhas) cada um com 8 palavras, ou seja no total 128 palavras.

Mantendo a capacidade constante, o simulador pode organizar a cache das formas seguintes:

- Cache de mapa directo: 16 blocos ou linhas. Teremos assim:
  - os bits 0 a 2 para seleccionar a palavra dentro da linha
  - os bits 3 a 6 para seleccionar a linha
  - o bit 7 é a chave (tag)
- Cache associativa pura: com 16 blocos ou linhas.
  - os bits 0 a 2 para seleccionar a palavra dentro da linha
  - os bits 3 a 7 formam a chave (tag)
- Cache associativa por grupos: existe a opção de termos 2 linhas ou quatro linhas por grupo. Teremos assim:
  - 8 grupos com duas linhas cada (2-way): bits 0 a 2 para seleccionar a palavra na linha, bits 3 a 5 para seleccionar o grupo e bits 6 e 7 para a chave (tag)
  - 4 grupos com quatro linhas cada (4-way) : bits 0 a 2 para seleccionar a palavra na linha, bits 3 e 4 para seleccionar o grupo e bits 5 a 7 para a chave (tag)

Trabalho

O simulador permite:

- gerar uma sequência de 10 endereços aleatoriamente;
- introduzir manualmente uma sequência de endereços;
- guardar e ler o conjunto de endereços num ficheiro.

1) Verifique para cada tipo de cache de que forma o endereço é dividido, como são usados os vários campos do endereço e de que forma vai sendo usada a cache.

2) Construa e guarde num ficheiro uma sequência de 10 endereços à sua escolha. Para cada um dos tipos de cache descritos na aula teórica (mapa directo, associativa pura, associativa por grupos) verifique quantos *hits* e quantos *misses* ocorrem.

2 - Código amigo da cache

Verifique a diferença no tempo de execução, entre os programas que têm código “amigo” da cache e os que não o têm.
Para isso vamos comparar os tempos de execução de duas funções que somam todos os valores de uma matriz; uma percorre a matriz por linhas, outra por colunas.

Pode utilizar as função **sumarrayrows** e **sumarraycols** a seguir apresentadas

```
#include <stdio.h>

#define M (1<<12)
#define N M

int a[M][N];

int sumarraycols()
{
    int i, j, sum = 0;

    for (j = 0; j < N; j++)
      for (i = 0; i < M; i++)
          sum += a[i][j];
    return sum;
}

int main()
{
    int i, j;

    for (i = 0; i < M; i++)
      for (j = 0; j < N; j++)
          a[i][j] = ((j%2 == 0) ? 1 : -1);

    /* should sum to zero */
    printf("sum=%d\n", sumarraycols(a));
    exit(0);
}
```

```
#include <stdio.h>

#define M (1<<12)
#define N M

int a[M][N];

int sumarrayrows()
{
    int i, j, sum = 0;

    for (i = 0; i < M; i++)
      for (j = 0; j < N; j++)
          sum += a[i][j];
    return sum;
}


int main()
{
    int i, j;

    for (i = 0; i < M; i++)
      for (j = 0; j < N; j++)
          a[i][j] = ((j%2 == 0) ? 1 : -1);

    /* should sum to zero */
    printf("sum=%d\n", sumarrayrows(a));
    exit(0);
}
```

Tenha em conta que a matriz deve ter tamanho suficiente para a execução levar um tempo significativo; pode também experimentar executar a função um certo número de vezes para ter uma medição mais precisa.

Para medir o tempo de CPU pode usar a função da biblioteca standard do C **clock** que dá o número de ciclos de CPU gastos pelo programa até ai. A constante **CLOCKS_PER_SEC** dá o número de ciclos por segundo. Consulte o manual (**man 3 clock**).