Memory System Performance

Memory system performance is largely captured by three parameters,

- Latency, Bandwidth, Average memory access time (AMAT).

- Latency: The time it takes from the issue of a memory request to the time the data is available at the processor.
- Bandwidth: The rate at which data can be pumped to the processor by the memory system.
- AMAT: The average time it takes for the processor to get a data item it requests.

Average Memory Access Time (AMAT)

- The time it takes to get requested data to the processor can vary: due to the memory hierarchy.
- Performance of a cache is largely determined by:
  - Cache hit rate: number of cache hits divided by number of accesses.
  - Cache miss rate: number of cache misses divided by number of accesses.
  - Cache hit time: the time between sending address and data returning from cache.
  - Cache miss penalty: the extra processor stall cycles caused by access to the next-level cache.

AMAT can be expressed as:

$$\text{AMAT} = \text{Cache hit time} + \text{Miss rate} \times \text{Miss penalty}$$

Example-1

- Main memory access time (Tm)=100nsec. Cache access time (Tc) is 5 times lower than main memory access time. Hit rate is 95%. Find out AMAT?

- AMAT = $h * T_c + \{(I-h) * (T_c + T_m)\}$

Example-2

- Cache hit time=10nsec. Miss penalty=150nsec. Hit rate is 98%. Find out AMAT?

- AMAT= Cache hit time + Miss rate * Miss penalty

CPU Performance

- Basic Terms related to CPU performance
- Frequency (f) or clock rate.(in HZ)
- Cycle Time (CT) (=1/Frequency)
- Cycle count (CC): Total no. of cycles required to execute a set of instructions or programs
- Instruction Count (IC): Total number of instructions in a set of instructions
- Cycles per instruction (CPI) = CC/IC
- CC = IC*CPI
- Execution Time=CC * CT = IC * CPI * CT =IC * CPI /F

Example-1

There is three processors. The processor P1 has 3GHz clock rate and a CPI of 1.5. The processor P2 has 2.5GHz clock rate and a CPI of 1.0. The processor P3 has 4GHz clock rate and a CPI od 2.2.

Which processor has the highest performance

Solution

Performance =1/Execution time

Execution time (ET)= Total CPU clock cycles * Cycle time

= total CPU clock cycles * (1/clock rate or frequency)

P1: ET= (I*1.5)/3G, Performance=2G/I

P2: ET= (I*1)/2.5G, Performance= 2.5G/I

P3: ET= (I*2.2)/4G, Performance= 1.8G/I

P2>P1>P3

Impact of Cache Memory System on Processor Performance

- Basic Terms related to Cache memory performance

CPU Time=Total number of cycles * clock cycle time

Total no. of cycles= CPU clock cycles + Memory stall cycles

CPU clock cycles= IC * CPI

Memory stall cycles= No. of misses * Miss Penalty

= (IC * Misses/Instruction) * Miss Penalty

= IC*(Memory accesses * Miss rate)/Instruction * Miss Penalty

= IC *(Memory accesses/Instruction)*Miss rate * Miss Penalty

=IC * Memory Stall CPI

[Miss rate= No. of misses/ Memory accesses

$\Rightarrow$No. of misses=Memory accesses * Miss rate]

Memory Stall CPI= (Memory accesses/Instruction) *Miss rate * Miss Penalty

Total no. of cycles= CPU clock cycles + Memory stall cycles

= (IC * CPI) + {IC *(Memory accesses/Instruction)*Miss rate * Miss Penalty}

Example-1:

Assume we have a computer where the cycle per instruction (CPI) is 1.0 when all memory accesses hit in the cache. The only data accesses are loads and stores, and these total 50% of the instructions. If the miss penalty is 25 clock cycles and the miss rate is 2%, how much faster would the computer be if all instructions were cache hits?

Solution:

CPU execution time (in clock cycle terms)= (CPU Clock Cycles + Memory stall cycles) Clock cycle

Computation of the performance of computer that always gives cache hits:

CPU execution time=(IC*CPI +0) Clock cycle =IC*1.0 Clock cycle

Performance1= 1/(IC*1.0)

Computation of the performance of computer with the real cache:

Memory stall cycles= IC*(Memory access / Instruction)* Miss rate * Miss penalty

=IC*(1+.5)* 0.02*25= IC* 0.75 Clock cycles

CPU execution times= CPU cycles + Memory stall cycles

= (IC* CPI) + (IC *0.75)= (IC* 1.0) + (IC *0.75)

=1.75 *IC Clock cycle

Performance1=1/(IC*1.75)

Performance ratio=Performance1/ Performance2

= (IC*1.75) /(IC *1.0)=1.75 times faster

Example 2

- Suppose:
  - Clock Rate = 200 MHz (5 ns per cycle), Ideal (no misses) CPI = 1.1
  - 50% arith/logic, 30% load/store, 20% control
  - 10% of data memory operations get 50 cycles miss penalty
  - 1% of instruction memory operations also get 50 cycles miss penalty
- Compute CPI and AMAT ?

Solution

CPI= 1.1+ (0.30 * 0.10 * 50) + (1 * 0.01 * 50)

= 1.1 + 1.5 + .5 cycle/ins = 3.1

(memory per instruction for data is 30% (due to load and store) whereas for instruction is always 100%)

AMAT= Cache hit time + Miss rate * Miss penalty

Cache hit time=Ideal CPI

AMAT={%instruction*(Cache hit time + Miss rate * Miss penalty)}+ {%data*(Cache hit time + Miss rate * Miss penalty)}

AMAT=(1/1.3)*[1.1+0.01*50]+(0.3/1.3)*[1.1+0.1*50] = 2.63

Example-3

- Assume that:
- Instruction miss rate 2%
- Data miss rate 4%
- CPI is 2 (without any memory stalls)
- Miss penalty 40 cycles
- 36% of instructions are load/store
- Determine how much faster a machine would run with a perfect cache that never missed.

Solution

Total memory stall cycles=IC*[(1*0.02*40)+(0.36*0.04*40)]=1.38*IC

CPU execution time with stall=(IC*2)+(IC* 1.38) Clock cycle=3.38 *IC

CPU execution time with perfect cache (in clock cycle)=(IC*2) Clock cycle

CPU time with stall/CPU time with perfect cache=(3.38*IC)/(2*IC)

=1.69

Example 4

Assume 20% Load/Store instructions

Assume CPI without memory stalls is 1

Cache miss penalty = 100 cycles

Miss rate = 1%

What is: memory stall cycles per instruction?

CPI with and without cache?

Solution

Total memory accesses per instruction = 1+0.2 = 1.2

Memory Stall cycle per instruction=

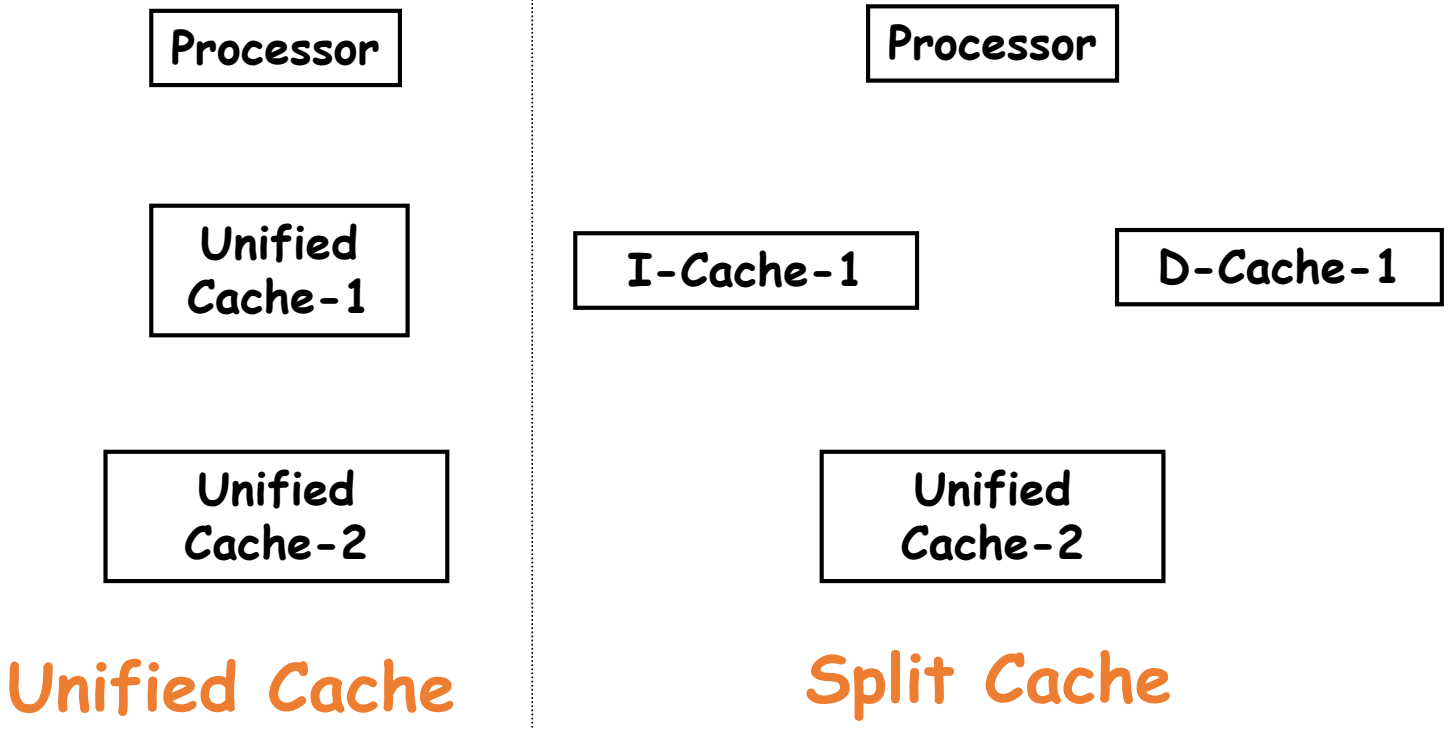
Total memory accesses per instruction * Miss rate * Miss Penalty
=(1+0.2)*0.01*100=1.2 cycles

CPI with cache = Ideal CPI + Memory Stall cycle per instruction = 1+1.2=2.2

CPI without cache= Ideal CPI+(Total memory access per instruction * Miss Penalty)= 1+(1.2*100)=121

Unified vs Split Caches

- Separate Instruction and Data caches:
  - Avoids structural hazard



Example-1 (Split vs Unified)

Total instruction=1000	Miss per 1000 instruction
Instruction cache size=16 KB	3.82
Data cache size=16 KB	40.9
Unified cache size=32 KB	43.3

36% of the instructions are data transfers

74% of the memory accesses are instruction reference

A hit takes 1 clock cycles and a miss penalty is 200 clock cycles.

A load/store hit takes extra 1 clock cycle on a unified cache

Which has lower miss rate? Obtains AMAT in each case.

Solution

Miss rate= Misses/memory access=(Misses/Instruction)/(Memory access/ Instruction)

Total instruction=1000	Miss per instruction
Instruction cache size=16 KB	3.82/1000
Data cache size=16 KB	40.9/1000
Unified cache size=32 KB	43.3/1000

Miss rate (of 16KB instruction)= (3.82/1000)/1.00=0.004

Miss rate (of 16 KB data) = (40.9/1000)/0.36=0.114 (36% of the instructions are data transfer instruction)

Overall miss rate for the split cache=(74% *0.004)+(26% * 0.114)=0.0326

- Miss rate (of the 32 KB unified Cache)= (43.3/1000)/(1.00+0.36)=0.0318
- 32KB of unified cache has slower miss rate than two 16 KB split cache.

Average memory access time=%instruction * (Hit time + Miss rate* Miss penalty)+ %data * (Hit time + Miss rate* Miss penalty)

AMAT (split)= 74%*(1+0.004*200)+ 26%*(1+0.114*200)=7.52

AMAT (unified)= 74%*(1+0.0318*200)+ 26%*(1+1+0.0318*200)=7.62

- Split cache have better memory access time than Unified cache.

Example-2 (Split vs Unified)

Instruction cache size=16 KB

Data cache size=16 KB

Unified cache size=32 KB

Miss rates are as follows:

16KB instruction cache has miss rate= 0.64%

16KB data cache has miss rate= 6.47%

32KB unified cache has miss rate= 1.99%

75% of the memory accesses are instruction reference

A hit takes 1 clock cycles and a miss penalty is 50 clock cycles.

A load/store hit takes extra 1 clock cycle on a unified cache

Which has lower miss rate? Obtains AMAT in each case.

Solution

Overall Miss rate of Split cache={0.75*(0.64/100) }+{0.25*(6.47/100)}

={(0.75*0.64)+(0.25*6.47)}/100

=(0.48+1.61)/100=2.09/100=2.09%

Miss rate of Unified Cache=1.99%

Unified cache has slower miss rate.

- AMAT (split)= 75%*{1+(0.64/100)*50}+ 25%*{1+(6.47/100)*50)

  =0.75(1+0.32)+0.25(1+3.235)=2.048
- AMAT (unified)= 75%*{1+(1.99/100)*50}+ 25%*{1+1+(1.99/100)*50)

  =2.246
- Split cache have better (lower) memory access time than Unified cache.

Example 3

- Which has a lower miss rate?
  - A split cache (16KB instruction cache +16KB Data cache) or a 32 KB unified cache?
- Compute the respective AMAT also.
- 40% Load/Store instructions
- Hit time = 1 cycle
- Miss penalty = 100 cycles
- 70% of memory accesses are instruction references
- Simulator showed:
  - 40 misses per thousand instructions for data cache
  - 4 misses per thousand instructions for instruction cache
  - 44 misses per thousand instructions for unified cache

Solution:

- Miss rate = (misses/instructions)/(mem accesses/instruction)
- Instruction cache miss rate= (4/1000)/1.00=0.004
- Data cache miss rate = (40/1000)/0.4 =0.1
- Unified cache miss rate = (44/1000)/1.4 =0.04
- Overall miss rate for split cache = 0.3*0.1+0.7*0.004=0.0303
- Split cache has slower miss rate than Unified cache.

- AMAT (split cache)= 0.7*(1+0.004*100)+0.3(1+0.1*100)=4.3
- AMAT (Unified)= 0.7(1+0.04*100)+0.3(1+1+0.04*100)=4.5
- Split cache has better memory access time than Unified cache.

Example 4

- Assume 16KB Instruction and Data Cache:
- Inst miss rate=0.64%, Data miss rate=6.47%
- 32KB unified cache: Aggregate miss rate=1.99%
- Assume 33% data ops (load/store)
- hit time=1, miss penalty=50
- Data hit has 1 additional stall for unified cache
- Which is better?

Solution

Assume 33% data ops ⇒ 75% accesses from instructions (1.0/1.33)

$AMAT_{Split}$ =75%x(1+0.64%x50)+25%x(1+6.47%x50)
= 2.05
$AMAT_{Unified}$=75%x(1+1.99%x50)+25%x(1+1+1.99%x50)
= 2.24

Split cache is better than Unified cache.

Example-5

- Cache miss penalty=200 clock cycles
- All instruction normally take 1.0 clock cycle (ignoring memory stall)
- Assume that the average miss rate is 2%, there is an average of 1.5 memory references per instruction, and the average number of cache misses per 1000 instruction is 30.
- What is the impact on performance when behavior of the cache is included?
- Calculate the impact using both misses per instruction and miss rate?

(Calculation of Performance using misses per instruction and Calculation of Performance using miss rate)

- Calculate CPI without cache?

CPU Time=IC*(Ideal CPI + Memory stall CPI)*Cycle time

Calculation of Performance using misses per instruction

Memory stall CPI= Misses/Instruction * Miss Penalty

CPU Time=IC*{1+[(30/1000)*200]}*Cycle time = IC* 7.00*Cycle time

Calculation of Performance using miss rate

Memory stall CPI= Memory access/Instruction * Miss rate* Miss Penalty

CPU Time=IC*{1+(1.5*0.02*200)}*Cycle time = IC* 7.00*Cycle time

CPI without cache=Ideal CPI+(Total memory access per Instruction * Miss Penalty)= 1+(1.5*200)=301

Direct Mapping Vs Set Associative Mapping

Direct Mapping:

K-way set associative mapping:

In set associative mapping, an extra multiplexer is added to select a cache block from a selected set.

Example-1

Assume that CPI with a perfect cache is 1.6 Clock cycle

Clock cycle time is 0.35 ns

There are 1.4 memory references per instruction.

Size of both cache is 128KB, and both have block size of 64bytes.

One cache is direct mapping and the other is two-way set associative mapping.

For the set associative caches, a multiplexer is added to select between the blocks in the set depending on the tag match. Since the speed of processor tied directly to the speed of a cache hit, assume the processor clock cycle time must be stretched 1.35 times to accommodate the selection multiplexor of the set associative cache.

Cache miss penalty is 65ns for either cache organization.

Assume hit time is 1 clock cycle

Miss rate of direct mapped 128 KB cache is 2.1%

Miss rate for a two-way set associative cache of the same size is 1.9%.

Calculate AMAT and processor performance for both cache?

Solution

AMAT=hit time+(Miss rate * Miss penalty)

Given clock cycle time=0.35ns

Hit time= 1 clock cycle= 1 * 0.35ns=0.35ns

AMAT (Direct Mapping)= 0.35+(2.1% * 65)=0.35+(0.021 * 65)=1.715 ns

AMAT(two-way set associative mapping)= (0.35*1.35)+(1.9% * 65)

=0.4725 + 1.235=1.707 ns

The two-way set associative cache is better than direct mapping

CPU time= IC * Ideal CPI * Cycle time

+ IC * (memory access/instruction) * miss rate * miss penalty* Cycle time

=IC * { Ideal CPI * Cycle time

+ (memory access/instruction) * miss rate * miss penalty* Cycle time }

Miss penalty= 65ns = (miss penalty * cycle time)

CPU time (Direct mapping)= IC*{(1.6*0.35) + (1.4 * 2.1%* 65)}

=IC*(0.56 + 1.911)=2.471*IC

CPU time(2-way set associative mapping)= IC*{(1.6*0.35 *1.35) + (1.4 * 1.9%* 65)}=IC*(0.756+1.729)=2.485* IC

Performance (Direct Mapping) / Performance(2-way Set associative mapping) = CPU time (2-way)/CPU time(1-way)=(2.485*IC)/(2.471*IC)=1.0056

Direct mapping processor performance is slightly better than 2-way set associative mapping processor performance.

Example 2

- What is the impact of 2 different cache organizations on the performance of CPU?
- Clock cycle time 1nsec
- Cache hit time=1 cycle
- Ideal CPI=2 cycle
- 50% load/store instructions
- Size of both caches 64KB:
  - Both caches have block size 64KB
  - one is direct mapped the other is 2-way sa.
- Cache miss penalty=75 ns for both caches
- Miss rate DM= 1.4% Miss rate SA=1%
- CPU cycle time must be stretched 25% to accommodate the multiplexor for the SA

- Solution

- AMAT of DM= 1+(0.014*75)=2.05nsec
- AMAT of SA=1*1.25+(0.01*75)=2ns

- CPU Time= IC*(CPI+(Misses/Instr)*Miss Penalty)* Clock cycle time
- CPU Time of DM= IC*{2*1.0+(1.5*0.014*75)}=3.58*IC
- CPU Time of SA= IC*{2*1.25+(1.5*0.01*75)}=3.63*IC