**CACHES**

Offset bits = bits

Set size =

Number of sets in cache = sets

Index bits = = bits

Tag size = 32 – (10 + 6) = 16 bits

1. For a computer that always hits, CPI = 1

Computer with misses has a miss rate of 2% and miss penalty of 25 cycles

Memory access per instruction = 1 instruction access + 0.5 data access = 1.5 accesses per instruction

Stall cycles per instruction = memory access per instruction \* miss rate \* miss penalty

= 1.5 \* 0.02 \* 25 = 0.75% therefore CPI is 1.75

Hence the computer with no cache misses is 1.75 times faster

1. Miss rate = 0.05

Block size = 2 words (8 bytes)

Frequency of memory operations from processor =

Frequency of writes from processor = 0.25 \*

Bus can only transfer one word at a time to/from processor/memory

On average 30% of blocks in the cache have been modified (must be written back in the case of the write back cache)

Cache is write allocate So: Fraction of read hits = 0.75 \* 0.95 = 0.7125, Fraction of read misses = 0.75 \* 0.05 = 0.0375, Fraction of write hits = 0.25 \* 0.95 = 0.2375 and Fraction of write misses = 0.25 \* 0.05 = 0.0125

1. Cache is write through

* On a read hit there is no memory access and on a read miss memory must send two words to the cache
* On a write hit the cache must send a word to memory and on a write miss memory must send two words to the cache and then the cache must send a word to memory
* Average words transferred = (0.7125 \* 0) + (0.0375 \* 2) + (0.2375 \* 1) + (0.0125 \* 3) = 0.35
* Average bandwidth used = 0.35 \*
* Fraction of bandwidth used =

1. Cache is write back

* On a read hit there is no memory access and on a read miss:
* If replaced line is modified then cache must send two words to memory and then memory must send two words to the cache
* If replaced line is clean then memory must send two words to the cache
* On a write hit there is no memory access and on a write miss:
* If replaced line is modified then cache must send two words to memory and then memory must send two words to the cache
* If replaced line is clean then memory must send two words to the cache
* Average words transferred = (0.7125 \* 0) + [0.0375 \* (0.7 \* 2 + 0.34 \* 4)] +(0.2375 \* 0) + [0.0125 \* (0.7 \* 2 + 0.3 \* 4)] = 0.13
* Average bandwidth used = 0.13 \*
* Fraction of bandwidth used =
* Comparing (a) and (b) the write through cache uses more than twice the cache memory bandwidth of the write back cache

1. CPU performance: CPUTime = IC \* CPI \* ClockTime

CPI = + Stall cycles per instruction

Instruction miss penalty is 50 cycles

Data read hit takes 1 cycle

Data write hit takes 2 cycles

Data miss penalty is 50 cycles for write through cache

Data miss penalty is 50 cycles or 100 cycles for write back cache

Miss rate is 1% for data cache (MRD) and 0.5% for instruction cache (MRI)

MRD is 0.01 and MRI is 0.005

50% of cache blocks are dirty in the write back cache

26% of all instructions are loads

9% of all instructions are store

* Therefore = (0.26 \* 1) + (0.09 \* 2) + (0.65 \* 1) = 1.09

1. Write through

Stall cycles per instruction = (0.005 \* 50) + [0.01 \* (0.26 \* 50 + 0.09 \* 50)] = 0.425

CPI = 1.09 + 0.425 = 1.515

1. Write back

Stall cycles per instruction = (0.005 \* 50) + [0.01 \* {0.26 \* (0.5 \* 50 + 0.5 \* 100)}] + [0.09 \* (0.5 \* 50 + 0.5 \* 100)] = 0.5125

CPI = 1.09 + 0.5125 = 1.6025

* Comparing (i) and (ii) the system with the write back cache is 6% slower than the system with the write through cache