**1.**

**ANS**

**a)**

Average access time = (1-miss rate) x hit time + miss rate x miss time

= 0.97x1 + 0.03x110

= 4.27 cycles

**b)**

Probability of hit = 64KB/1GB

= 0.00006104

Average access time = 0.00006104x1 + 0.99993896x110

= 109.99334664 cycles

**2.**

**ANS**

CPU: 1.1 GHz (0.909 ns),

CPI = 1.35(memory accesses excluded)

Instruction Mix: 20% loads, 10% Stores, 70% non-memory access instructions

Caches: Split L1 with no bit penalty

L1 I-cache: 2% miss rate, 32 bytes blocks, miss penalty 15ns + 2 cycles

L2 D-cache: 5% miss rate, 16 bytes blocks, write through, miss penalty 15ns + 1 cycle

L1/L2 Bus: 128 bits bus, 266MHz bus between L1 and L2 caches

L2 unified cache: 512KB, write back, 80% hit rate, 50% of replaced blocks are dirty, 64-byte blocks, miss

penalty 60+7.52 ns = 67.52 ns.

Memory: 128 bits(16 bytes wide),

first access- 60 ns , subsequent accesses-1 cycle on 133MHz, 128-bit bus

**a)**

L1 miss time in L2 = 15ns(access time) + 2 L2 cycles

= 15+32 x 3.75/16

= 2.5ns

L2 miss time in memory = 60ns + 4 memory cycles

= 60 + 64/16 x7.5

= 90ns

Average memory access time = average AT in L2 cache + average AT in memory + access AT for l2 write back

= (0.02 x 22.5) + .(02 x.2 x 90) + (0.02 x 0.5x 90)

= 0.99ns

**b)**

L1 read miss time in L2 = 15 + 3.75

= 18.75ns

L2 miss time in memory = 90ns

Average memory AT for read = .(02 x 18.75) + (0.02 x.2 x 90) + (0.02 x 0.2 x.5 x 90)

= 0.92ns

**c)**

L1 write time to L2 = 15 + 3.75

= 18.75NS

L2 miss time in memory = 90ns

Average memory AT for write = (0.05 x 18.75) + (0.05 x. 2x 90) + (0.05 x 0.2 x.5 x 90)

= 2.29ns

**d)**

Overall CPI = 1.35 + 1.09 + 0.2 + 1.01 + .(1 x 2.52)

= 3.902 CPI

**3.**

**ANS**

**a)**

|  |  |  |
| --- | --- | --- |
| **Instruction** | **Memory- Hit/Miss** | **Cache Content** |
| Fld f0, 0(x0) | Load M[0], Miss | 4,1,-,-,-,-,-,- |
| Fld f2, 0(x2) | Load M[16], Miss | 4,1,7,9,-,-,-,- |
| Fsd f2, 0(x2) | Store M[16], Hit | 4,1,28,9,-,-,-,- |
| Fld f2, 0(x2) | Load M[24], Hit | 4,1,28,9,-,-,-,- |
| Fsd f2, 0(x2) | Store M[24], Hit | 4,1,28,36,-,-,-,- |
| Fld f2, 0(x2) | Load M[32], Miss | 4,1,28,36,5,3,-,- |
| Fsd f2, 0(x2) | Store M[32], Hit | 4,1,28,36,20,3,-,- |
| Fld f2, 0(x2) | Load M[40], Hit | 4,1,28,36,20,3,-,- |
| Fsd f2, 0(x2) | Store M[40], Hit | 4,1,28,36,20,12,-,- |
| Fld f2, 0(x2) | Load M[48], Miss | 4,1,28,36,20,12,1,2 |
| Fsd f2, 0(x2) | Store M[48], Hit | 4,1,28,36,20,12,4,2 |
| Fld f2, 0(x2) | Load M[56], Hit | 4,1,28,36,20,12,4,2 |
| Fsd f2, 0(x2) | Store M[56], Hit | 4,1,28,36,20,12,4,8 |
| Fld f2, 0(x2) | Load M[64], Miss | 6,8,28,36,20,12,4,8 |
| Fsd f2, 0(x2) | Store M[64], Hit | 24,8,28,36,20,12,4,8 |
| Fld f2, 0(x2) | Load M[72], Hit | 24,8,28,36,20,12,4,8 |
| Fsd f2, 0(x2) | Store M[72], Hit | 24,32,28,36,20,12,4,8 |
| Fld f2, 0(x2) | Load M[80], Miss | 24,32,7,3,20,12,4,8 |
| Fsd f2, 0(x2) | Store M[80], Hit | 24,32,28,3,20,12,4,8 |

**b)**

|  |  |  |
| --- | --- | --- |
| **Instruction** | **Memory- Hit/Miss** | **Cache Content** |
| Fld f0, 0(x0) | Load M[0], Miss | 4,1,-,-,-,-,-,- |
| Fld f2, 0(x2) | Load M[16], Miss | 4,1,-,-,7,9,-,- |
| Fsd f2, 0(x2) | Store M[16], Hit | 4,1,-,-,28,9,-,- |
| Fld f2, 0(x2) | Load M[24], Hit | 4,1,-,-,28,9,-,- |
| Fsd f2, 0(x2) | Store M[24], Hit | 4,1,-,-,28,36,-,- |
| Fld f2, 0(x2) | Load M[32], Miss | 4,1,-,-,28,36,-,- |
| Fsd f2, 0(x2) | Store M[32], Hit | 4,1,20,3,28,36, -,- |
| Fld f2, 0(x2) | Load M[40], Hit | 4,1,20,3,28,36, -,- |
| Fsd f2, 0(x2) | Store M[40], Hit | 4,1,20,12,28,36, -,- |
| Fld f2, 0(x2) | Load M[48], Miss | 4,1,20,12,28,36,1,2 |
| Fsd f2, 0(x2) | Store M[48], Hit | 4,1,20,12,28,36,4,2 |
| Fld f2, 0(x2) | Load M[56], Hit | 4,1,20,12,28,36,4,2 |
| Fsd f2, 0(x2) | Store M[56], Hit | 4,1,20,12,28,36,4,8 |
| Fld f2, 0(x2) | Load M[64], Miss | 6,8,20,12,28,36,4,8 |
| Fsd f2, 0(x2) | Store M[64], Hit | 24,8,20,12,28,36,4,8 |
| Fld f2, 0(x2) | Load M[72], Hit | 24,8,20,12,28,36,4,8 |
| Fsd f2, 0(x2) | Store M[72], Hit | 24,32,20,12,28,36,4,8 |
| Fld f2, 0(x2) | Load M[80], Miss | 24,32,20,12,7,3,4,8 |
| Fsd f2, 0(x2) | Store M[80], Hit | 24,32,20,12,28,3,4,8 |

**4.**

**ANS**

Average Memory Access Time = Hit Time + Miss Rate x Miss Penalty

Average Memory Access Time for Data = 1 + 0.06 \* 16 = 1.96

Average Memory Access Time for Instruction = 1 + 0.05 \* 16 = 1.8

Average = (1.96 + 1.8) / 2

= 1.88 cycles

**5.**

**ANS**

**a)**

|  |  |  |
| --- | --- | --- |
| **Cache Block** | **Set** | **Memory Blocks that can reside in cache** |
| 0 | 0 | M0, M8, M16, M24 |
| 1 | 1 | M1, M9, M17, M25 |
| 2 | 2 | M2, M10, M18, M26 |
| 3 | 3 | M2, M10, M18, M26 |
| 4 | 4 | M4, M12, M20, M28 |
| 5 | 5 | M5, M13, M21, M29 |
| 6 | 6 | M6, M14, M22, M30 |
| 7 | 7 | M7, M15, M23, M31 |

**b)**

|  |  |  |
| --- | --- | --- |
| **Cache Block** | **Set** | **Memory Blocks that can reside in cache** |
| 0 | 0 | M0, M2, M4, M6, … M30 |
| 1 | 1 | M1, M3, M5, M7, … M31 |
| 2 | 0 | M0, M2, M4, M6, … M30 |
| 3 | 1 | M1, M3, M5, M7, … M31 |
| 4 | 0 | M0, M2, M4, M6, … M30 |
| 5 | 1 | M1, M3, M5, M7, … M31 |
| 6 | 0 | M0, M2, M4, M6, … M30 |
| 7 | 1 | M1, M3, M5, M7, … M31 |

**6**

**ANS**

CPI with cache misses = 1 + (0.2 x 0.02) x 25

= 1.1

Speed up if all cache hits = 1.1/1

= 1.1.

The computer will be 10% faster if every instructions cache hits.

**7**

**ANS**

With critical word first it takes 10 cycles to transfer the critical word.

Without Critical word first it takes 10 + (32-1) = 41 cycles

So, L2 cache with Critical Word first will be 41/10 = 4.1 times faster.