L9 Cache II Exercises

Acknowledgement: some slides taken from UC Berkeley CS61C

Cache Replacement Policies

- Consider 12-bit memory address. Consider two cache configurations: a DM cache with total size 128 Bytes, 16 Bytes/block (8 blocks); and a 4-way SA cache of the same size. For the SA cache, we consider two replacement policies – Least Recently Used (LRU) and First-In-First-Out (FIFO).
- Consider the following sequence of memory addresses in hex, starting with an empty cache. **Complete the following tables** for the DM cache and both types of 4-way SA caches showing the progression of cache contents as accesses occur (in the tables, 'inv' = invalid, and the column of a particular cache block contains the tag of that block). *You only need to fill in elements in the table when a value changes.* The first few rows have been filled in for you.
- Note that the table format is different from that in "L8 Cache I Exercises", since we need to add a time dimension vertically.
  - We use L0 to denote Cache Line 0, which means the same as Cache Block 0, to avoid confusion with B0 in "L8 Cache I Exercises", which stands for Byte address 0 within a cache block.
  - Each table entry contains Tag of that cache block, instead of cache content in "L8 Cache I Exercises". For brevity, the hex prefix "0x" is omitted from the Tag.

Tag:Set Index:Offset

- For DM cache
  - # Bytes/block=16 → Offset is 4b
  - # Sets=#blocks=128/16=8 → SI is 3b
  - Tag size=12-4-3=5
  - Tag:Set Index:Offset bits: 5:3:4
- For 4-way SA cache
  - # Bytes/block=16 → Offset is 4b
  - # Sets=#blocks/#ways=(128/16)/4=2 → SI is 1b
  - Tag size=12-4-1=7
  - Tag:Set Index:Offset bits: 7:1:4

Q: DM Cache

Address	DM Cache								
	Cache Block (Tag in Hex)								hit?
	L0	L1	L2	L3	L4	L5	L6	L7	
0x110	inv	2	inv	inv	inv	inv	inv	inv	N
0x136				2					N
0x202	4								N
0x1A3									
0x102									
0x361									
0x204									
0x114									
0x1A4									
0x177									
0x301									
0x206									
0x135									

Time

- Tag:Set Index:Offset bits: 5:3:4.
- Memory address 0x110=000100010000(bin). Set Index=001(bin), hence it is mapped to L1, with Tag=10(bin)=0x2. Cache miss.
- Memory address 0x136=000100110110(bin). Set Index=011(bin), hence it is mapped to L3, with Tag=10(bin)=0x2. Cache miss.
- Memory address 0x202=001000000010(bin). Set Index=000(bin), hence it is mapped to L0, with Tag=100(bin)=0x4. Cache miss.

A: DM Cache

Address	DM Cache								
	Cache Block (Tag in Hex)								hit?
	L0	L1	L2	L3	L4	L5	L6	L7	
0x110	inv	2	inv	inv	inv	inv	inv	inv	N
0x136				2					N
0x202	4								N
0x1A3			3						N
0x102	2								N
0x361							6		N
0x204	4								N
0x114		2							Y
0x1A4			3						Y
0x177								2	N
0x301	6								N
0x206	4								N
0x135				2					Y

3 cache hits, 10 misses

- Memory address 0x110=000100010000(bin). Set Index=001(bin), hence it is mapped to L1, with Tag=10(bin)=0x2. Cache miss.
- Memory address 0x136=000100110110(bin). Set Index=011(bin), hence it is mapped to L3, with Tag=10(bin)=0x2. Cache miss.
- Memory address 0x202=001000000010(bin). Set Index=000(bin), hence it is mapped to L0, with Tag=100(bin)=0x4. Cache miss.
- Memory address 0x114=000100010100(bin). Set Index=001(bin), hence it is mapped to L1, with Tag=10 (bin)=0x2. Cache hit!
  - Memory addresses 0x110 and 0x114 only differ in their offsets, hence they are in the same cache block (with size 16 Bytes). Access to 0x110 was a miss, and brought in the cache block, so access to 0x114 is a hit.
- Memory address 0x1A4=000110100100(bin). Set Index=010(bin), hence it is mapped to L2, with Tag=11 (bin)=0x3. Cache hit!
- Memory address 0x135=000100110101(bin). Set Index=011(bin), hence it is mapped to L3, with Tag=10 (bin)=0x2. Cache hit!

Q: 4-Way SA Cache w/ LRU

Address	4-Way SA Cache								
	Cache Block (Tag in Hex)								hit?
	Set 0				Set 1				
	Way0	Way1	Way2	Way3	Way0	Way1	Way2	Way3	
0x110	Inv	Inv	Inv	Inv	08	Inv	Inv	inv	N
0x136						09			N
0x202	10								N
0x1A3									
0x102									
0x361									
0x204									
0x114									
0x1A4									
0x177									
0x301									
0x206									
0x135									

- Tag:Set Index:Offset bits: 7:1:4.
- Memory address 0x110=000100010000(bin). Set Index=1(bin), hence it is mapped to Set 1, with Tag=1000(bin)=0x8. It can be placed anywhere in the 4 ways of Set 1, but let's assume it is placed in Way 0. Cache miss.
- Memory address 0x136=000100110110(bin). Set Index=1(bin), hence it is mapped to Set 1, with Tag=1001(bin)=0x9. It can be placed anywhere in the remaining 3 ways of Set 1, but let's assume it is placed in Way 1. Cache miss.
- Memory address 0x202=001000000010(bin). Set Index=0(bin), hence it is mapped to Set 0, with Tag=10000(bin)=0x10. It can be placed anywhere in the 4 ways of Set 0, but let's assume it is placed in Way 0. Cache miss.

A: 4-Way SA Cache w/ LRU

4-way **Address**	LRU Cache								
	Cache Block (Tag in Hex)								hit?
	Set 0				Set 1				
	Way0	**Way1**	**Way2**	**Way3**	**Way0**	**Way1**	**Way2**	**Way3**	
0x110	Inv	Inv	Inv	Inv	08	Inv	Inv	inv	N
0x136						09			N
0x202	10								N
0x1A3		0D							N
0x102			08						N
0x361				1B					N
0x204	10								Y
0x114					08				Y
0x1A4		0D							Y
0x177							0B		N
0x301			18						N
0x206	10								Y
0x135						09			Y

5 cache hits, 8 misses

- 4-Way SA cache with 8 blocks, has 2 sets, and distribution of Tag:Set Index:Offset bits as 7:1:4.
- Memory address 0x110=000100010000(bin). Set Index=1(bin), hence it is mapped to Set 1, with Tag=1000(bin)=0x08. No tag match, cache miss. It can be placed anywhere in the 4 ways of Set 1, but let's assume it is placed in Way 0.
- Memory address 0x136=000100110110(bin). Set Index=1(bin), hence it is mapped to Set 1, with Tag=1001(bin)=0x09. No tag match, cache miss. It can be placed anywhere in the remaining 3 ways of Set 1, but let's assume it is placed in Way 1.
- Memory address 0x202=001000000010(bin). Set Index=0(bin), hence it is mapped to Set 0, with Tag=10000(bin)=0x10. No tag match, cache miss. It can be placed anywhere in the 4 ways of Set 0, but let's assume it is placed in Way 0.
- Memory address 0x204=001000000100(bin). Set Index=0(bin), hence it is mapped to Set 0, with Tag=10000(bin)=0x10. Tag match with block in Way 0, cache hit!
- Memory address 0x177=000101110111(bin). Set Index=1(bin), hence it is mapped to Set 1, with Tag=1011(bin)=0x0B. No tag match, cache miss. It can be placed anywhere in the remaining 2 ways of Set 1, but let's assume it is placed in Way 2.
- Memory address 0x301=001100000001(bin). Set Index=0(bin), hence it is mapped to Set 0, with Tag=11000(bin)=0x18. No tag match, cache miss. It replaces the LRU block in Way 2.

Q: 4-Way SA Cache w/ FIFO

4-way **Address**	FIFO Cache								
	Cache Block (Tag in Hex)								hit?
	Set 0				Set 1				
	Way0	**Way1**	**Way2**	**Way3**	**Way0**	**Way1**	**Way2**	**Way3**	
0x110	Inv	Inv	Inv	Inv	08	Inv	Inv	inv	N
0x136						09			N
0x202	10								N
0x1A3									
0x102									
0x361									
0x204									
0x114									
0x1A4									
0x177									
0x301									
0x206									
0x135									

- 4-Way SA cache with 8 blocks, has 2 sets, and distribution of Tag:Set Index:Offset bits as 7:1:4.
- Memory address 0x110=000100010000(bin). Set Index=1(bin), hence it is mapped to Set 1, with Tag=1000(bin)=0x8. It can be placed anywhere in the 4 ways of Set 1, but let's assume it is placed in Way 0. Cache miss.
- Memory address 0x136=000100110110(bin). Set Index=1(bin), hence it is mapped to Set 1, with Tag=1001(bin)=0x9. It can be placed anywhere in the remaining 3 ways of Set 1, but let's assume it is placed in Way 1. Cache miss.
- Memory address 0x202=001000000010(bin). Set Index=0(bin), hence it is mapped to Set 0, with Tag=10000(bin)=0x10. It can be placed anywhere in the 4 ways of Set 0, but let's assume it is placed in Way 0. Cache miss.

A: 4-Way SA Cache w/ FIFO

4-way Address	FIFO Cache								
	Cache Block (Tag in Hex)								hit?
	Set 0				Set 1				
	Way0	**Way1**	**Way2**	**Way3**	**Way0**	**Way1**	**Way2**	**Way3**	
0x110	Inv	Inv	Inv	Inv	08	Inv	Inv	inv	N
0x136						09			N
0x202	10								N
0x1A3		0D							N
0x102			08						N
0x361				1B					N
0x204	10								Y
0x114					08				Y
0x1A4		0D							Y
0x177							0B		N
0x301	18								N
0x206		10							N
0x135						09			Y

4 cache hits, 9 misses

- 4-Way SA cache with 8 blocks, has 2 sets, and distribution of Tag:Set Index:Offset bits as 7:1:4.
- Memory address 0x301=001100000001(bin). Set Index=0(bin), hence it is mapped to Set 0, with Tag=11000(bin)=0x18. No tag match, cache miss. It replaces the block in Way 2, which entered the cache the earliest (by access to 0x202).
- Memory address 0x206=001000000110(bin). Set Index=0(bin), hence it is mapped to Set 0, with Tag=10000(bin)=0x10. No tag match, cache miss. It replaces the block in Way 1, which entered the cache the earliest (by access to 0x1A3).

Q: AMAT

- Assume that the results of the above analysis can represent the average miss-rates. What would be the average memory access latency in CPU cycles for each type of cache? Assuming:
  - Cache hit time is 2 cycles for DM cache, 3 cycles for 4-way SA cache
  - Cache miss penalty is 20 cycles for both

A: AMAT

- The miss rate for the DM cache is 10/13. The miss rate for the LRU 4-way SA cache is 8/13. The miss rate for the FIFO 4-way SA cache is 9/13.
  - For DM cache, AMAT = 2+(10/13)*20 = 17.38 ≈ 18 cycles.
  - For LRU 4-way SA cache, AMAT = 3 + (8/13)*20 = 15.31 ≈ 16 cycles.
  - For FIFO 4-way SA cache, AMAT = 3 + (9/13)*20 = 16.85 ≈ 17 cycles.
- LRU 4-way SA cache has the best performance in terms of AMAT

LRU vs. FIFO

- Q: Does LRU always outperform FIFO?
- A: No. Neither is optimal.
- Consider an FA cache with 2 blocks, and an access sequence for memory addresses in blocks 1, 2, 1, 3, 2.
- With LRU: 1 (M), 2(M), 1(H), 3(M, replaces block 2), 2(M)
- With FIFO: 1 (M), 2(M), 1(H), 3(M, replaces block 1), 2(H)
- LRU→1 hit; FIFO→2 hits
- Consider an FA cache with 2 blocks, and an access sequence for memory addresses in blocks 1, 2, 1, 3, 1.
- With LRU: 1 (M), 2(M), 1(H), 3(M, replaces block 2), 1(H)
- With FIFO: 1 (M), 2(M), 1(H), 3(M, replaces block 1), 1(M)
- LRU→2 hits; FIFO→1 hit

Review

Average Memory Access Time (AMAT)

- Average Memory Access Time (AMAT) is the average time to access memory considering both hits and misses in the cache

AMAT = Hit rate * Hit time + Miss rate * Miss time

= (1 – Miss rate)*Hit time + Miss rate * (Hit time + Miss penalty)

= Hit time + Miss rate * Miss penalty

Review

Local vs. Global Miss Rates

- *Local miss rate* – the fraction of references to one level of a cache that miss
  - L2 Local Miss Rate = L2 Misses / L1 Misses
- *Global miss rate* – the fraction of references that miss in all levels of caches and must go to memory
  - Global Miss rate = L2 Misses / Total Accesses
  - = (L2 Misses / L1 Misses) × (L1 Misses / Total Accesses)
  - = L2 Local Miss Rate × L1 Local Miss Rate
- L1 Miss Penalty = L2 AMAT; L2 Miss Penalty = Memory access time
- L1 cache only: AMAT = Hit Time + Miss rate × Miss penalty
- L1+L2 caches: AMAT = L1 Hit Time + L1 Local Miss rate × (L2 Hit Time + L2 Local Miss rate × L2 Miss penalty)

Question: AMAT

- Compute AMAT for 1-level cache system, given:
  - L1 Hit Time: 1 cycle, L1 Miss Rate: 2%
  - Main Memory access time: 51 cycles
  - CPU clock cycle time: 200 ps/cycle

- Which of the following results in largest decrease in AMAT?

A. **Faster CPU with 190 ps cycle time**

B. **Reduce miss penalty to 40 clock cycles**

C. **Reduce miss rate to 0.015 misses/instruction**

Answer: AMAT

- Compute AMAT for 1-level cache system, given:
  - L1 Hit Time: 1 cycle, L1 Miss Rate: 2%
  - Main Memory access time: 51 cycles
  - CPU clock cycle time: 200 ps/cycle
- A: Miss penalty = 50 cycles; AMAT = 1 +.02*50 = 2 cycles = 400 ps
- Which of the following results in largest decrease in AMAT?

A. Faster CPU with 190 ps cycle time

AMAT = 1 +.02*50 = 2 cycles = 380 ps

B. Reduce miss penalty to 40 clock cycles

AMAT = 1 +.02*40 = 1.8 cycles = 360 ps

C. Reduce miss rate to 0.015 misses/instruction

AMAT = 1 +.015*50 = 1.75 cycles = 350 ps

Question: AMAT

- For 2-level cache system, given:
  - For every 1000 instructions, on average
    - 40 misses in L1, 20 misses in L2
  - L1 Hit Time: 1 cycle
  - L2 Hit Time: 10 cycles
  - Main memory access time: 100 cycles
- Compute local miss rate, AMAT, stall cycles per instruction

Answer: AMAT

- For 2-level cache system, given:
  - For every 1000 instructions, on average
    - 40 misses in L1, 20 misses in L2
  - L1 Hit Time: 1 cycle
  - L2 Hit Time: 10 cycles
  - Main memory access time: 100 cycles
- Compute:
  - 1. L1 local miss rate, L2 local miss rate, global miss rate; AMAT
  - 2. Repeat for the case without L2 cache
- A: 1. With L2 cache: L1 local miss rate=40/1000=0.04; L2 local miss rate = 20/40 = 0.5; global miss rate = 20/1000 = 0.02; AMAT=1+0.04*(10+0.5*100)=3.4
- 2. If we remove the L2 cache: L1 local miss rate=0.04; AMAT=1+0.04*100=5

Question: AMAT

- Compute AMAT for 2-level cache system, given:
  - L1 Hit Time: 1 cycle, L1 Local Miss Rate: 3%
  - L2 Hit Time: 6 cycle, L2 Local Miss Rate: 10%.
  - Main Memory access time: 120 cycles

Answer: AMAT

- Compute AMAT for 2-level cache system, given:
  - L1 Hit Time: 1 cycle, L1 Local Miss Rate: 3%
  - L2 Hit Time: 6 cycle, L2 Local Miss Rate: 10%.
  - Main Memory access time: 120 cycles

- AMAT = 1 + .03*(6 + .10*120) = 1.54

Question: AMAT

- Assuming main memory access time is 100 cycles. Compute AMAT for
- 1. 16KB L1 cache only with hit time 2 cycles, and hit rate 90%
- 2. 128KB L1 cache only with hit time 10 cycles, and hit rate 97.5%
- 3. 16KB L1 cache + 128KB L2 cache
  - L1 Hit Time: 2 cycles, Local Hit Rate: 90%
  - L2 Hit Time: 12 cycles, Local Hit Rate: 75%

Answer: AMAT

- Assuming main memory access time is 100 cycles. Compute AMAT for
- 1. 16KB L1 cache only with hit time 2 cycles, and hit rate 90%
  - A: 2+0.1*100=12 cycles
- 2. 128KB L1 cache only with hit time 10 cycles, and hit rate 97.5%
  - A: 10+0.025*100=12.5 cycles
- 3. 16KB L1 cache + 128KB L2 cache
  - L1 Hit Time: 2 cycles, Local Hit Rate: 90%
  - L2 Hit Time: 12 cycles, Local Hit Rate: 75%
  - A: 2+0.1*(10+0.25*100)=5.5 cycles