Bus-Based Multi-Processor System

Shared Memory Model:

CPU cores

Bridge

Memory Controller

Physical Memory (DRAMs)

CPU Bus

Shared Memory Implementation:
Cache Coherence Problem

(Assume writeback caches)

1. P₁ performs ld A
2. P₁ performs st A
   ⇒ A is dirty in P₁'s cache
3. P₂ performs ld A

⇒ P₂ loads stale copy of A from memory.

(Assume write through caches)

1. P₁ performs ld A
2. P₂ performs ld A
3. P₁ performs st A
4. Pₐ performs ld A

⇒ P₂ loads stale copy of A from cache.
Memory Consistency Model

⇒ when sequential consistency is broken.

Sequentially consistent order:

1. \( P_1 \) writes \( A = 1 \)
2. \( P_1 \) writes \( X = 1 \)
3. \( P_2 \) reads \( X = 1 \)
4. \( P_2 \) reads \( A = 1 \)
5. \( P_2 \) writes \( B = A = 1 \)

⇒ Caches reorder writes.

P. order: \( A = 1, X = 1 \)
P. order: \( X = 1, A = 1 \)

Assume write back caches:

1. \( P_1 \) writes \( A = 1 \)
2. \( P_1 \) writes \( X = 1 \)
3. \( X \) is written back to memory
4. \( P_2 \) reads \( X = 1 \)
5. \( P_2 \) reads \( A = 0 \)
6. \( P_3 \) writes \( B = A = 0 \)
Snoopy Cache Coherence Buses

Main Memory

Cache Bus

Snoopy Cache Coherence Hardware
Bus-Based Write Invalidate