Istanbul Technical University
Department of Computer Engineering

BLG 322E – Computer Architecture
Recitation 4

Note: If you have a question about the recitation, you may contact the research assistants of the course (ozcelikfu@itu.edu.tr).

QUESTION 1:

In a symmetric multiprocessor (SMP) system using a shared bus:

- There are two CPUs (CPU1 and CPU2) that have local cache memories.
- The system does not have a shared L2 cache.
- Main memory size is **64 K words**.
- Each local cache memory can store **4 K words** of data.
- Data transfer between main memory and cache memories is in blocks of **16 words**.
- The cache control unit uses **direct mapping**.
- Cache memory is used only for data, not for instructions.
- For write operations, the **Write Back** (WB) method is used.
- To provide cache coherence, the snoopy **MESI** protocol is used.
- There is a shared array A in the system.
- The starting address of array A in main memory is $000C, and its size is 10 words.

a) Assume that the valid copy of array A is residing in main memory and in cache memory of CPU2. The cache memory of CPU1 is empty.
A program running in CPU1 clears the array by assigning zero to all elements using a loop.
   i) Which control messages are sent by the MESI cache controllers during the run of the loop? Write the messages in the order they are sent.
   ii) What are the MESI states of the corresponding frames storing array A in the caches of the CPUs? Is the value in main memory valid after this clear operation? (Write below.)
   CPU1: **CPU2:** **Main Memory:**

b) After the clear/write operation of CPU1, CPU2 uses a loop to read all elements of array A. Which control messages are sent by the MESI cache controllers during the run of the loop? Write the messages in the order they are sent.

SOLUTION 1:

a)

i)

Array A (A[0], … A[9]), address range = $000C - $001F

Array A occupies 2 frames (Frame 0 and Frame 1) in each cache memory.

A[0] ← 0; write miss ;

1. Control message from CPU1: "*read-with-intent-to-modify*"

A[1] ← 0, A[2] ← 0, A[3] ← 0; write hit ; No message

A[4] ← 0; write miss; 2nd Frame

2. Control message from CPU1: "*read-with-intent-to-modify*"

A[5] - A[9] ← 0; write hit; No message

Note: The problem statement asks for the control messages, not for the state transitions of the frames.

ii)

CPU1:	**CPU2:**	**Main Memory:**
Modified	Invalid	Invalid

b)

Read A[0] ; read miss;

1. Control message from CPU2: "*read*"

CPU1 blocks the read operation, writes Frame 0 back to main memory.

2. Control message from CPU1: "*Shared*"

Read A[1], A[2], A[3]; read hit; No message

Read A[4] ; read miss; 2nd Frame

3. Control message from CPU2: "*read*"

CPU1 blocks read operation, writes Frame 1 back to main memory.

4. Control message from CPU1: "*Shared*"

Read A[5] - A[9] ; read hit; No message

Note: The problem statement asks for the control messages, not for the state transitions of the frames.