Chp 2 - Exercise 4 - Data Coherence

•You have 4 processors (nodes) each with their own cache memory, but all tied to the same shared main memory.

•Assume all four are making modifications to an int variable.

•Using the Invalidate protocol show how the state of that int variable changes as each processor makes an update to it.

•Do the same, but this time using the snoopy approach.

•Given it does take time for information to move from the cache to memory what would it look like if two processes tried to update the int variable at the same time and how might it get handled?

Grading Rubric:
_____ (3 Points) For changes by any two of the processors show how the Invalidate protocol handles this (can be using a diagram or just clearly stated in words)
_____ (2 Points) For changes by any two of the processors show how the Snoopy approach handles this (can be using a diagram or just clearly stated in words)
_____ (2 Points) For simultaneous changes by two processors show what happens with the invalidate protocol.

Answer :

1. Invalidate Protocol (Handling changes between two processors):
When a processor writes to the shared int variable, it broadcasts an invalidate message to the other processors, marking their caches as invalid. For example:

- If P1 modifies the int variable, an invalidate message is sent to P2, P3, and P4, forcing them to mark their cache as invalid.
- If P2 later modifies the variable, it will fetch the updated value from memory, modify it, and then broadcast an invalidate message to the other processors.

2. Snoopy Approach (Handling changes between two processors):
In the snoopy approach, when a processor modifies the int variable, it broadcasts the change on a shared bus. Other processors "snoop" on the bus, detecting the modification, and automatically invalidate or update their caches. For example:

- If P1 modifies the int variable, it broadcasts a message, and P2, P3, and P4 will snoop on the bus and invalidate or update their caches accordingly.
- If P2 modifies the variable, a similar broadcast occurs, ensuring coherence

without explicit cache invalidation.

3. Simultaneous Changes (Invalidate Protocol):

If two processors try to update the int variable simultaneously, the invalidate protocol handles this by serializing the writes. For example:

- If P1 and P2 both try to modify the variable, whichever processor broadcasts the invalidate message first (say P1) will succeed, and P2 will have to invalidate its cache and fetch the latest value from memory before applying its own update.