FIT3143 2024 - Applied Session Week 6

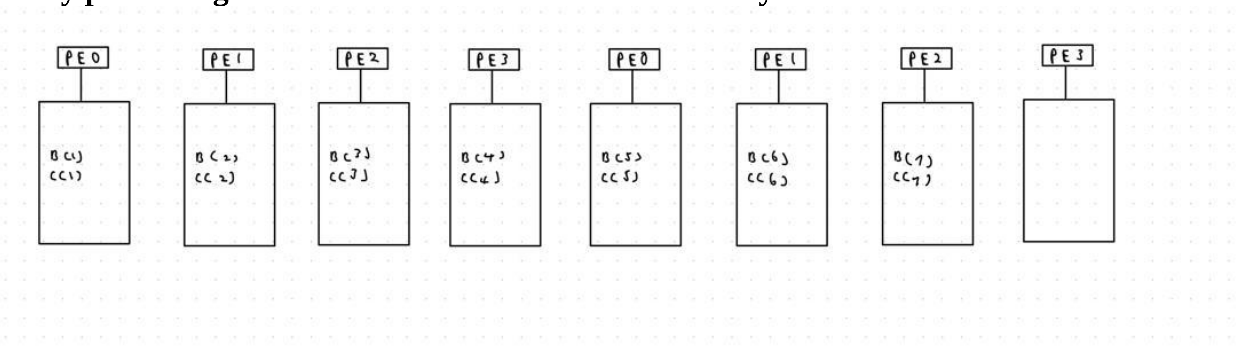
Student ID	Student Name	Student Email
32844700	Teh Jia Xuan	Jteh0015@student.monash.edu

Question 1 – Part A

Given a SIMD architecture machine with 4 processing elements. For the following piece of pseudocode performing array addition:

```
DO 10 I = 1,N
10 A(I) = B(I) + C(I)
```

A) If N = 7, draw a diagram (similar to W4 Lecture slides - page 16) to explain how the data is arranged in the memory for each processing element and how many loops are needed. Discuss if any processing elements will need to be disabled at any time.



The data will be arranged as the diagram above, each processing unit take 1 element which is B(N), C(N) at a time where N is the element that is processing. There will be 2 loops, but during the second loop, processing element 3 needs to be disable. To ensure only the required processing elements are active to save processing power.

Question 1 – Part B

For the following piece of pseudocode utilizing instruction for vector operations:

```
ADD N, A, B, C
```

B) Explain how the overhead of opcode fetches, decodes, and operand loads are reduced. Explain also how chaining is used in vector processing.

As it is using vector operations a single instruction like add can operate on entire arrays rather than on individual values. Thus, it only needs to fetch and decode once. For instance, Add N, A, B, C saying that single add operation is performed across all N elements in the vectors A, B and C. Hence, this reduces the overhead significantly.

Additionally, vector operations can load multiple data elements into a single memory access. So it no need to load each operand individually instead it loaded a block of data which reduces the number of memory access required. Therefore, reduce overhead.

Chaining is used in vector processing, when the result of one operation for a particular element can be immediately used as the input for the next operation even while other elements are still being processed. Thus, it does not need to wait the entire operation to complete.

Question 2 – Part A

Compare and contrast Distributed Memory MIMD and Shared Memory MIMD. Provide at least two advantages & two disadvantages of Distributed Memory MIMD, and two advantages & two disadvantages of Shared Memory MIMD.

Distributed Memory MIMD

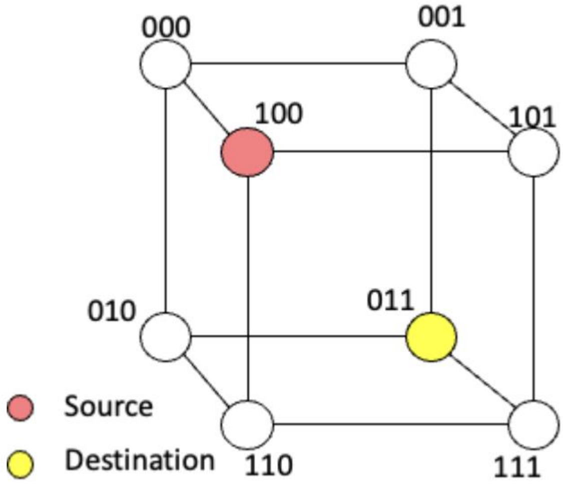
Advantage	**Disadvantage**
More scalable, each processor has its own memory so no need to make changes to the memory architecture.	More cost as more hardware is required when each processor have its own memory
Faster, no need to wait for other processor to finish their task before using it.	Complexity of communication between processors. Since each processor has its own memory. Data only can be passed using message passing protocols (MPI).

Shared Memory MIMD

Advantage	**Disadvantage**
Communication between processor is faster in shared memory system as they can directly access the same memory location and they don't need MPI to communicate.	Multiple processors trying to access the same memory location and it needs to stall. Thus, effecting the overall system performance
Low cost, as all the processor is sharing the same memory, less hardware is required.	Less scalable as if more processors are added. All processors are sharing the same memory and the shared memory bus can become a bottleneck.

Question 2 – Part B

B) Given the following (hyper)cube interconnection topology, which organizes nodes with identifiers in binary numbers.



Explain how we can find the number of hops between nodes 100 and 011. Will the number of hops be decreased if we rearrange into a balanced binary tree topology?

Suppose one more node needs to be inserted, explain whether it is possible to keep using only 3 bits for identification. Will node insertion be easier for a binary tree topology?

The number of hops between two nodes is determined by the Hamming distance, which is the number of bit positions at which the corresponding bits are different. For example, node 100 and 011. The first bit is 1 vs 0 and 0 vs 1 and 0 vs 1, So all three bits are different compared to the others. So the hops required to travel from node 100 to 011 is 3.
3

If we are using the balanced binary tree topology the two node is depends on their position within the tree. If the nodes are closer then the hop will be less. But if the nodes are far apart then the hops will be greater.

For this case the hops will be decreased as it only required 2 hops in a balanced binary tree topology.

It is not possible to add another node, if only 3 bits are using for identification as the maximum number of node is 2^3 = 8. The node insertion in binary tree topology is easier as we need to add a node as a leaf. While hypercube required us to increase the dimensionality which required restructuring connections.

Question 3

The issue in the code is the process 0 tries to receive from process 1 before process 1 sends the message and process 1 tries to receive from process 0 before process 0 sends the message. So process 0 waiting for process 1 to send the message while process 1 waiting for process 0 sends the message. This forms a deadlock as they are waiting for each other.