FP Bonus: NoC Topology

1. **Mesh:**

   Routers are arranged in a grid-like structure. Each router connects to its immediate neighbors (north, south, east, west). Implemented using a grid of routers, each connected to local processing elements (PEs). I use XY routing algorithms for simplicity.

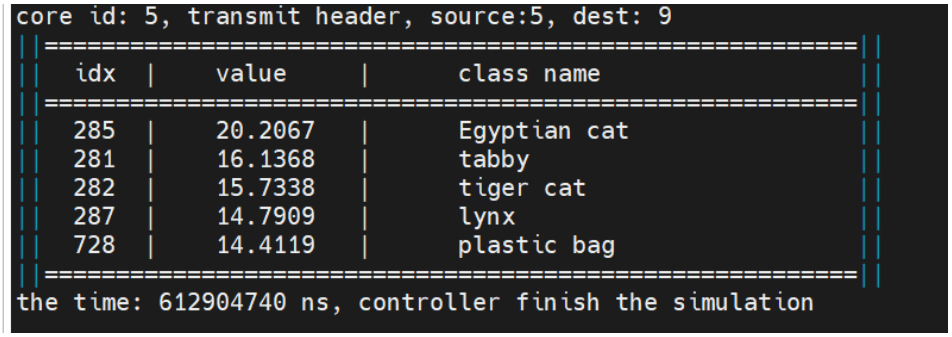
   The latency of mesh topology is generally moderate. As the number of nodes increases, the average distance (hop count) between nodes grows linearly. As the other side of throughput, mesh topology has good scalability; each node has multiple paths for data transmission.

   Pros:

   - Scalability: Easy to expand by adding rows and columns.
   - Redundancy: Multiple paths between nodes provide fault tolerance.
   - Implementation: Relatively simple to design and implement.

   Cons:

   - Power Consumption: Can be high due to the large number of links and routers.
   - Area: Larger chip area required compared to simpler topologies.
   - Complexity: Routing algorithms can become complex in large meshes.



2. **Ring:**

   Routers are arranged in a circular fashion. Each router is connected to two neighbors (one clockwise and one counterclockwise). I Implemented ring topology using a series of routers connected in a loop. I used simple token ring protocols or round-robin arbitration for routing.

   The latency of ring topology can be high for larger rings as data may have to traverse many nodes. As the other side of throughput, ring topology limited by the number of nodes and the single-path nature of the ring.

   Pros:

   - Simplicity: Simple design and easy to implement.
   - Area: Requires less chip area and fewer links than mesh or torus.
   - Power Consumption: Lower power consumption due to fewer links and routers.

   Cons:

   - Scalability: Poor scalability as adding more nodes increases latency.
   - Fault Tolerance: Single point of failure can disrupt the entire network.
   - Performance: Lower throughput and higher latency for large networks.

```
core id: 3, transmit header, source:3, dest: 7
||=========================================================||
||   idx   |    value     |      class name                 ||
||=========================================================||
||   285   |    20.2067   |      Egyptian cat               ||
||   281   |    16.1368   |      tabby                      ||
||   282   |    15.7338   |      tiger cat                  ||
||   287   |    14.7909   |      lynx                       ||
||   728   |    14.4119   |      plastic bag                ||
||=========================================================||
the time: 612904870 ns, controller of ring topology finish the simulation

Info: /OSCI/SystemC: Simulation stopped by user.
1:21 mlchip004@ee21[~/final]$
```

3. **Torus**

Since 2*2 type is the same as mesh, I make router0, 1, 2 as dimension1, and router4 be dimension2 of torous topology. Similar to a mesh but with wrap-around connections (i.e., the edges connect to form a doughnut shape). I Implemented torous topology by adding wrap-around connections to a mesh structure. Routing can be done using enhanced XY routing algorithms or adaptive routing to utilize wrap-around links effectively.

The latency of torous is better than mesh for large networks due to shorter average path lengths. And the throughput of torous is higher than mesh due to additional wrap-around links providing more paths.

Pros:

- Performance: Improved latency and throughput compared to mesh.
- Fault Tolerance: Increased due to additional wrap-around links.
- Scalability: Better scalability compared to mesh and ring.

Cons:

- Complexity: More complex to design and implement due to wrap-around links.
- Power Consumption: Higher power consumption than mesh due to extra links.
- Area: Requires more chip area than mesh.

```
core id: 4, transmit header, source:4, dest: 8
||=========================================================||
||   idx   |    value     |      class name                 ||
||=========================================================||
||   285   |    20.2067   |      Egyptian cat               ||
||   281   |    16.1368   |      tabby                      ||
||   282   |    15.7338   |      tiger cat                  ||
||   287   |    14.7909   |      lynx                       ||
||   728   |    14.4119   |      plastic bag                ||
||=========================================================||
the time: 612904730 ns, controller of torous topology finish the simulation
```

Performance Analysis and Simulation Time:

- **Mesh Topology: Offers a balanced trade-off between complexity and latency. Suitable for a range of applications but may face latency issues as the network size grows.**
- **Ring Topology: Simplest and least complex, but latency increases significantly with network size, making it less suitable for large networks.**
- **Torus Topology: Provides the best latency performance for large networks due to shorter average path lengths but at the cost of increased complexity and power consumption.**

	Mesh	Ring	Torous
Simulation Time	612904740	612904870	612904730