MLCHIP Bonus Report

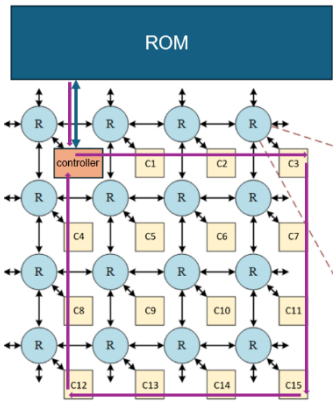
HW4 Implementation

NoC Router

Architecture	4x4 Torus NoC architecture
Routing algorithm	XY routing (deadlock free)
Switching	Wormhole switching
Buffer size	one flit (34 bits), depth=1
Virtual channel	No implementation
Flow control	ack/nack protocol (buffered flow control)

AlexNet

ID	AlexNet Layer	# FP multiplier
1	CONV_RELU_1	70,276,800
2	MAX_POOLING_1	-
3	CONV_RELU_2	223,948,800
7	MAX_POOLING_2	-
11	CONV_RELU_3	112,140,288
15	CONV_RELU_4	149,520,384
14	CONV_RELU_5	99,680,256
13	MAX_POOLING_3	-
12	LINEAR_RELU_1	37,748,736
8	LINEAR_RELU_2	16,777,216
4	LINEAR_3	4,096,000



$$\text{Convolution: multipliers} = C_{\text{in}} \times K \times K \times H_{\text{out}} \times W_{\text{out}} \times C_{\text{out}}$$

$$\text{Linear: multipliers} = C_{\text{in}} \times C_{\text{out}}$$

Performance

Torus simulation time: 1236679610 ns

Mesh simulation time: 1236679850 ns

Comparing Torus and Mesh NoC topology, we can find that Torus has smaller simulation time. This is because there are more possible ways to go in the Torus topology. For example, if a packet wants to go from C4 to C11, in the

Mesh topology, the XY routing path is R4→R5→R6→R7→R11; meanwhile, since Torus connects the left most cores with the right most cores and top cores with bottom cores, the minimal routing path is R4→R7→R11, which is shorter than the path in Mesh.

However, we can observe that the difference of simulation times is not very big. The bandwidth in HW4 is only one flit, so the controller can only send one flit out in a cycle. Therefore, the traffic on the NoC architecture is not very heavy. If there are many packets transferring on the topology, the benefits gained from Torus will be obvious. Note that the Torus topology contains long wires, which may induce crosstalk and signal degradation problems.

Routing rules

Mesh

```
if(x_dest[i] != x_coor){
    if(x_dest[i] > x_coor){
        out_dir[i] = EAST;
    }
    else{
        out_dir[i] = WEST;
    }
}
else if(y_dest[i] != y_coor){
    if(y_dest[i] > y_coor){
        out_dir[i] = SOUTH;
    }
    else{
        out_dir[i] = NORTH;
    }
}
else{
    out_dir[i] = LOCAL;
```

Torus

```
if(x_dest[i] != x_coor){
    if((3+x_dest[i]-x_coor) % 3 <= 1){
        out_dir[i] = EAST;
    }
    else{
        out_dir[i] = WEST;
    }
}
else if(y_dest[i] != y_coor){
    if((3+y_dest[i]-y_coor) % 3 <= 1){
        out_dir[i] = SOUTH;
    }
    else{
        out_dir[i] = NORTH;
    }
}
else{
    out_dir[i] = LOCAL;
}
```

Possible Optimization

- **Increasing the buffer size.** Allowing more flits buffered in the router may decrease the latency because more flits are waiting in the nearby location.
- **Implementing virtual channels.** Increasing the flexibility for the packet. Release the traffic if there are many packets going to the same direction.
- **Replace the AHB protocol with AXI protocol.** Pull up the ready signal as soon as the buffer is not full and handshake when the transmitting node is ready. Do not have to wait ack signal from the receiving node after req.