**On-Chip Interconnection Network**

**CSEE 4340:**

**Computer Hardware Design**

**Fall 2012**

**SECTION 1:**

**Project Title: On-Chip Interconnection Network**

**Team Name:** Flit Busters

**Project Members:** Adil Sadik (ams2378)

Ashwin Ramachandran (ar2648)

Dechhin Lama (ddl2126)

Ayushi Rajeev (ar3110)

**Project Member Duties:**

**Design Team:**

* Adil Sadik (Master)
* Dechhin Lama

**Verification Team:**

* Ashwin Ramachandran (Master)
* Ayushi Rajeev

**Section 2:**

**Design Overview:**

In this project, we will be designing and testing a 4x4 on-chip interconnection network using SystemVerilog HDL.

As the number of IP-modules on System-On-Chip’s increases, bus based interconnections can no longer meet the necessary bandwidth, latency and power consumption requirements. Network on Chips provide a more efficient and scalable solution to this problem. NoC’s consist of routers that are connected to each IP-module. Routers are then connected in various topologies such that they can communicate to neighboring routers (nodes in a network).

\*Adil\*

In this project, we will design a Network on Chip of mesh (4x4) topology. NoC is consists of communicating cores, intermediate router nodes and point to point links. Intermediate routers are responsible to route the data to appropriate destination, and thus it establishes a notion of point-to-point communication between sender and receiver. Data is represented with *packets.* AMessage can consist of multiple packets and each packet is composed of multiple *flits*. The very first flit of each packet (*header flit*) contains the information required to locate the destination node and all the following flits (*body flits*) contain actual messages.

The placement of routers and cores is realized with a two-dimensional grid and each node is assigned with a co-ordinate (X, Y). The routing algorithm used for this project is *deterministic*. Data will always be router in a predefined order- first toward Y-axis and then X-axis- and it is referred as *dimension ordered routing*. After receiving the first flit of each packet, router will locate the information (i.e. co-ordinate) about the destination node and route the flit through *Y-X dimension-ordered* routing scheme as mentioned above. The NoC handles the messages in a *flit-by-flit* basis. Router doesn’t wait until it receives all the flits associate with a packet; instead it transmits each flit as soon as it knows that the next router has available space in its input buffer. All the body flits associated with a header flit follows the same routing path established for the header flit. It is referred as *wormhole routing*.

To prevent resource starvation and congestion inside network, the NoC will implement *distributed* *flow-control* mechanism. Particularly, we use *credit-based flow-control* which ensures that a router will only transmit data when there is available slots in the next router to accommodate the received flit. Hence, each router keeps tracks of the available free slots in the input buffers of its adjacent routers through the notion of *available* *credits*. By implementing *credit communication* among adjacent routers it will be ensured that the routers are keeping track of the available resources- required to make routing decision of each flits.

For our project, we will be designing a 4x4 mesh network where packets (a unit of message comprising of one or more flits) are routed to various nodes of the network. Routers will examine the header flit and route the following flits to the desired destination. Since we are designing a 4x4 mesh network, each node will be assigned an X-Y co-ordinate in a two-dimensional grid. Messages addressed to a particular co-ordinate are then passed along the nodes of the network, first in the Y-axis followed by the X-axis. Each router (node) will have four input-output pairs connecting to neighboring routers and one input-output pair to the local module.

In order to prevent cases where a transmitting router sends data when the receiving end is not yet ready to accept, we will be using a Credit-based flow control. Here, each router will have a buffer queue for each input where packets are stored temporarily. The transmitting router will maintain a count on the number of free slots available on the receiving end and transmit data only when there are “credits” available.

We will begin our design such that only single header flits are routed. Once this is successfully implemented and tested, we will then extend our design such that the header flit is followed by 4 body flits that follow through with the header flit to the designated destination (i.e. Wormhole routing). Furthermore, we will optimize our design by modifying the routers on the edge of the network such that unnecessary ports are omitted.