**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**

**SECTION 3: Unit level interfaces**

**SECTION 4: Sub partitioning the interfaces, Test harness structure**

**SECTION 5: Microarchitecture desig**

**SECTION 6: Verification Strategy**

**SECTION 7: Performance estimation**

**SECTION 8: Area estimate**

**SECTION 9: Bugs, Coverage**

**SECTION 10: Document revision history**

**Section 2:**

**Design Overview:**

Bus based interconnection systems are not scalable and power efficient for future many core System-on-chips or processors. Network on Chip provides a scalable and efficient solution to this problem. In this project, we will design a *synthesizable* 4x4 Network on Chip of mesh topology using SystemVerilog for design and validation.

A NoC consists of multiple communicating cores, intermediate router nodes and point-to-point links. Intermediate router nodes are responsible for routing data to its appropriate destination, and thus establish a notion of point-to-point communication between a sender and receiver. Data is represented by *messages*.A *message* can consist of multiple packets, where 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. All of the following flits (*body flits*) contain actual messages.

The placement of routers and cores is realized with a two-dimensional grid, where each node is identified with a unique (X, Y) co-ordinate. The routing algorithm used for this project is *deterministic*. Data will always be routed in a predefined order – first along the Y-axis and then along the X-axis. This is referred to as *dimension ordered routing*. After receiving the first flit of each packet, a router node will locate the co-ordinate of the destination node and route the flit through *Y-X dimension-ordered* routing scheme mentioned above. The NoC handles the messages in a *flit-by-flit* basis. This means that the 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 follow the same routing path established for the header flit. This is referred to as *wormhole routing*.

To prevent resource starvation and congestion inside the network, the NoC will implement a *distributed* *flow-control* mechanism. Specifically, we use *credit-based flow-control,* which ensures that a router will only transmit data when there are 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 *credits*. The number of credits indicates the number of free slots in the input data buffer of a router. By implementing *credit communication* among adjacent routers, we can ensure that the routers are keeping track of the available resources required to make routing decisions for each flit.

Each router node consists of 5 input and 5 output ports - one pair of input-output ports for data in the north, south, east, and west directions, with one extra pair for the local data. Each input port consists of a queue, which can store up to 5 incoming flits per direction. There is a crossbar switch, which routes one direction’s input to a different direction’s output, and 5 outputs buses, which are capable of sending one flit per cycle. In addition to this routing hardware, there exist credit counters that keep track of how many free slots exist in the neighboring router nodes’ input buffers.

The overall design will follow a top down methodology. The first goal is to implement and validate the routing of a single header flit. Afterwards the body flits will be incorporated with the header flit and it will be ensured that they follow wormhole routing scheme as mentioned before. One important checkpoint will be to design and validate one single router with all the ports and required functionality. Once, the router is ready, the whole NoC will be instantiated and the routers located on the perimeter will be optimized (i.e. eliminate unnecessary ports).