Final Project: Midpoint Check-In

Bryan Werth, Kaitlyn Keil, Prava Dhulipalla

Description of Project

We are trying to learn more about interconnected networks by implementing one in Verilog, and attempting to upload it to the FPGA and potentially writing simulation code to simulate the interconnected network we implement. See our Project Proposal for more description.

Research

Although some members of our team were very familar with the idea of an interconnected network simulator, others weren't. Over Thanksgiving break, we all researched interconnection networks in hopes to better understand them. For the purposes of this check-in document, we were only go in depth with the type of network that we are implementing/planning to implement, which is a torus network.

Our research primarily stemmed from the textbook *Principles and Practices of Interconnection Networks* by William James Dally and Brian Towles.

An interconnection network of a programmable system used to transport data between terminals. For instance, if we have six terminals in our interconnection networks, and we want terminal T1 to communicate with terminal T4, a message is sent into the network with the data. It is programmable because this can happen between any two terminals.

Interconnection networks are an important concept. They serve as a limiting performance factor in many computer systems. Two important ideas in a computer system - memory latency and memory bandwidth, are affected by the performance of an interconnection network.

When we say we want to create a torus network, this is referring to a topology - which refers to the arrangement of nodes and channels that, as a set, compose an interconnection network. The channels carry data packets from one router node to another in order to send the data to the correct node. There are many different types of topology - each with their own strengths and pitfalls in terms of performance and implementation.

Torus Topology

A torus network that is

n dimensional,
radix-*k*

k-ary *n* cube,

has

$N = k^n$ nodes

N-dimensional cube

k nodes in each dimension.

These networks are optimal because their physical arrangement allows for use of short wires. They are also bidirectional in the way they send data, and so for their communication patterns, throughput is higher but latency is lower “than for random traffic.”

What we are trying to create for our MVP is a functional 4-ary 1-cube torus network.

Progress

We are in the process of creating a top-level module to string together the modules to construct the nodes we have made. After this, we have successfully created a 4-ary 1-cube torus network (if it works).

Originally, we were planning on modelling a torus network through booksim, the Interconnection Network simulation. However, after an honest effort to download it, none of our team members could figure it out. We decided to put this on hold for the sake of research and implementation (but are planning on asking for help to download it). However, we have pivoted our MVP away from using booksim - it seems very interesting, but we feel we would learn a lot more otherwise.

Our next step is to expand this torus network into a 2-d network. This is a trivial step forward from what we already have - if that works, all it involves is expanding the number of inputs and outputs for nodes, as well as the number of nodes and channels.

If that implementation works, we hope to upload to the FPGA and somehow demonstrate the functionality of our network through a hardware implementation. In case if this is still not enough, we hope to write our own simulation software so we can test out our implementation.

Concerns

We have been having some difficulty in meeting with each other - as a result, it seems that our work is stalled. It also seems like our scoping is very off, so we are a bit concerned in overscoping our project, or if our current project is underscoped. It feels that way because implementation so far has been relatively simple, so the majority of our time has been wrapped up in understanding Interconnection Networks in terms of other topologies or general applications - and all of this knowledge is very enlightening, but it also doesn’t feel like enough. We hope that the forced group meeting during in-class time will propel us forward in implementation and from there, scoping appropriately will be simple.