Ethernet-Based Arithmetic Logic Unit and Beyond

FPGA Implementation Track

Eliah Reeves - *Computer Engineering B.S.*
Eric Chuang - *Computer Engineering B.S.*

Context

Ethernet has several many significant advantages over traditional serial communication protocols. Ethernet communication o□ers higher bandwidth and more flexible network topologies through its packet-based approach. While serial communication transmits data sequentially over a single channel with direct point-to-point connections, Ethernet enables multiple devices to communicate over a shared medium using CSMA/CD (Carrier Sense Multiple Access with Collision Detection) or switched networks. Modern Ethernet standards support data rates from 100 Mbps to 400 Gbps, far exceeding typical serial protocols like UART which typically operate at rates in the kbps. This increased performance comes with a trade-o□ in complexity, as Ethernet requires more sophisticated hardware and protocols to manage data transmission.

Objectives

The primary objective of our final project is to implement a basic Arithmetic Logic Unit (ALU) that can be controlled over an Ethernet connection. This will be very similar to the ALU we implemented in the first project but with the added complexity of Ethernet communication instead of UART. The ALU will be capable of performing basic arithmetic operations such as addition, signed multiplication, and signed division. To demonstrate functionality we will also create a python library which will allow a user to control the ALU over Ethernet.

Additional Goals

Depending on time remaining after completing our primary objective we will examine the feasibility of implementing the protocols necessary to communicate with the ALU over the internet requiring a TCP/IP stack. This would allow the ALU to be controlled from anywhere in the world. Although this would be quite di□cult, it would be a very interesting challenge to tackle, and our design could be used as the foundation for a high performance IoT device.

Intellectual Property Cores

Seeing as the core of this project is creating an Ethernet controlled device, we will create the Ethernet modules from scratch. In order to understand the protocols we will be implementing, we will be using resources available online. A comprehensive resource we may exime is Alex Forencich's Ethernet Implementation.

For functionality adjacent to Ethernet communication, we may use existing intellectual property (IP) cores. IP cores we will include basejump_stl which contains a variety of useful modules for our project including a multiplier and divider.

Required Hardware

For this project we will use the Nexys A7 FPGA board. This board contains a built-in Ethernet port, adequate system resources and an AMD MicroBlaze processor. All of these factors make it an ideal choice for a networking project.

Project Timeline

Week 1 - Week 2

Make significant progress implementing the Ethernet communication protocols. Ideally some testing will have been done and basic Ethernet communication will be close to functional.

Week 3 - Week 5

Complete the Ethernet communication protocols and connect to the ALU. Create python interface and verify Implementation. Evaluate feasibility of implementing TCP/IP stack, and potentially begin implementation.

Week 6 - Week 7

Wrap up project and prepare for presentation. Create demonstration of functionality and ensure all code is documented adequately. If TCP/IP stack was implemented, test functionality.

Team Responsibilities

Since we are a small team we will not have a strict division of lab.9626 TvP626r(w)28(e)-238iallycols.