ECE 410/510 Spring 2025 - Codefest Challenge #4 Report

1. Introduction

This report summarizes the implementation of Challenge #4 of ECE 410/510 Spring 2025. The goal was to replicate the methodology described in the Johns Hopkins paper titled 'Designing Silicon Brains using LLM' and to explore LLM-assisted HDL generation for spiking neuron arrays.

2. Paper Overview

The referenced paper leverages large language models like ChatGPT to design digital circuits-specifically a spiking neuron array based on the Leaky Integrate-and-Fire (LIF) model. The report outlines how an LLM can assist in HDL code generation, testbench creation, and ASIC conversion using OpenLane.

3. HDL Design Using LLM

Using ChatGPT, Verilog code for a 4-neuron LIF array was generated. Key parameters such as membrane potential, threshold, and leak factor were incorporated. The design was validated through a testbench.

4. Simulation and Testing

A testbench module was created to stimulate the neuron array with varying input currents. The design was simulated using Icarus Verilog and validated by observing spike outputs and potential resets.

5. GDS Generation via OpenLane

Although optional, the design was prepared for ASIC flow. Using OpenLane, the HDL was converted into a GDSII layout. This required modifications to the configuration scripts and verified compatibility with OpenLane's design flow.

6. Alternative Neuron Models

Additional neuron models such as the Rectified Linear Unit (ReLU) and a simplified Hodgkin-Huxley neuron were explored. While ReLU was easily implemented in Verilog, the latter was deemed too complex for RTL and recommended for high-level modeling.

7. Results and Comparison

The HDL design successfully matched the logic described in the Johns Hopkins paper. Functionality, spike timing, and behavior were validated. The GDS layout showed acceptable area and cell usage.

8. Improvement Suggestions

Suggested improvements include: parameterized neuron counts, better I/O interface (e.g., SPI), pipeline optimization, and enhanced ASIC constraints for area and timing.

9. Documentation and Deliverables

All LLM queries, Verilog code, simulation logs, and documentation have been uploaded to a public GitHub repository. This includes testbench files, waveforms, and configuration scripts.