Assignment 1
Course: Hardware Software Co-Design
Marks 20

Design and implement a simple feedforward neural network using Verilog HDL on an FPGA. The network will have one hidden layer and will perform a basic classification task.

Specifications:

1. Network Architecture:

- Input Layer: 4 inputs (each input is 8-bit wide)
- Hidden Layer: 3 neurons, each with a ReLU activation function.
- Output Layer: 2 outputs (each output is 8-bit wide), representing two classes.

2. Weights and Biases:

- For simplicity, use 8-bit signed integers for weights and biases.

3. Activation Function:

- Implement the ReLU activation function in Verilog: ReLU(x) = max(0, x).

4. Task:

- Implement the feedforward pass of the network, Calculate the outputs of the network for a given input set.

5. FPGA Implementation:

- Synthesize and implement the design on a specific FPGA platform, Use simulation tools to verify the correctness of the implementation.

Submission Requirements:

1. Verilog Code:

- Submit the Verilog code for the neural network, including the input layer, hidden layer, output layer, and activation function.

2. Testbench:

- Create a testbench to simulate the neural network with at least three different input sets and show the corresponding outputs.

3. Simulation Results:

- Provide simulation waveforms that demonstrate the correct operation of the neural network.

4. Report:

- Write a brief report explaining the design choices, including how the neural network was implemented in Verilog and how the testbench verifies the network's functionality. Discuss any challenges encountered during the implementation.