EE 580 Lab4

Summer 2017 Nazarian Score:_/100

Student ID:_____ Name: _____

Assigned: Tuesday, May 25

Due: Friday, June 2 at 11:59pm.

Late submissions will be accepted two days after the deadline with a maximum penalty of 15% per day: For each day, submissions between 12 and 1am: 2%, 1 and 2am: 4%, 2 and 3am: 8% and after 3am: 15%.

- **All assignments including this lab are based on individual work. No collaborations (including no discussions) are allowed.**
- **We may pick some students in random to demonstrate their design and simulations. Please watch the first lecture of this course regarding the academic integrity policies and also refer to the syllabus for a summary of AI policies (including the penalties for any violation).**
- **If you have any concerns or doubts about what is or is not allowed or prohibited in this course, please contact the instructor.**

Digital Neuron Activation

Following our lectures,

Part A (Sum of Product) (70 points)

1) Given sample input x and weights Θ, you need to calculate the sum of product. The inputs are in the form of 32 bit IEEE format floating point number.
2) Your design should be scalable to accept "n" pair of inputs (i.e. x and Θ).
3) For the sum of product, you need to use a 32 bit IEEE format floating point clocked adder and multiplier.
4) You are required to use only one adder and multiplier despite n multiplication and n -1 additions. First, you need to understand the functioning of the 32 bit IEEE format floating point clocked adder and multiplier.
5) To make the neuron work faster you need to pipeline your architecture. You are free to use buffer and FIFOs. Use **clock period of 4 ns**.

Part B (Activation Function) **(30 points)**

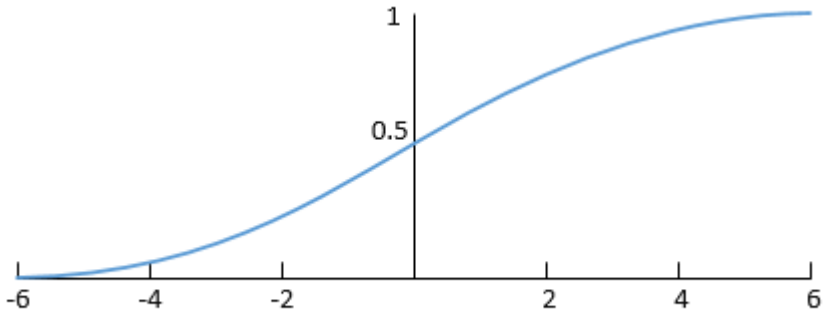
1) Generally, the activation function is Sigmoid or logistic function.
2) A modified form of sigmoid implementable in Verilog is:

$$F = \frac{X}{1 + abs(X)}$$ ... where **X is Sum of product calculated in part A**

$$\text{Sigmoid Function} = \frac{F+1}{2}$$

Note: abs mean absolute value

3) You are free to search on Internet and implement a different version that you find easier to code in Verilog but make sure it satisfies the result of logistic function shown below.



4) You don't need to optimize or pipeline this part.

Submission

1. Zip all the files you need to submit into a zip file named: "firstname_lastname_lab3_part1".zip.
2. Files given, 32 bit IEEE format adder, multiplier and divider.
3. Your zip file should include Readme.pdf, Verilog files for **module SOP, module Sigmoid, module adder, module multiplier, module divider.** You can create a single file named Activation.v under **module Activation. Make sure it works.**
4. In your Readme.pdf,
   a. **Create the image of the architecture you designed for the pipelined model in part A.**
   b. **Specify if you used a different sigmoid in part b. Please mention the formula for it.**
   c. Include any information that you think the course staff, especially the grader should know while grading your assignment: references, any non-working part, any concerns, etc.
   d. The citations should be done carefully and clearly, e.g.: *"to write my code, lines 27 to 65, I used the Diijkstra's shortest path algorithm c++ code from the following website: www.SampleWebsite.com/..."*
   e. The Readme file content of labs and PAs can be hand-written or typed. In case you decide to hand-write, then please scan and include in your Readme.pdf.
   NOTE: this policy is different from that of the HW. For HW assignments, the solutions have to be handwritten.

Use the provided BB submission link to submit your zip file for this assignment