ECE428/ECE528 Project Instruction

Haibo Wang

In this project, you are going to design and implement an artificial neural network (ANN) that can recognize handwritten digits. The hand written digits are provided as images with 8×8 pixels. Each pixel has a value ranging from -0.5 to 0.5. It is represented by 8-bit fixed point 2's complementary numbers with 7 fractional bits. As shown in Figure 1, the ANN to be implemented has three layers: input, hidden, and output layers. The input layer has 64 nodes, each corresponding to a pixel of the digit image. The output layer has 10 nodes, each corresponding to a digit value (from 0 to 9). There are 12 nodes are in the hidden layer.

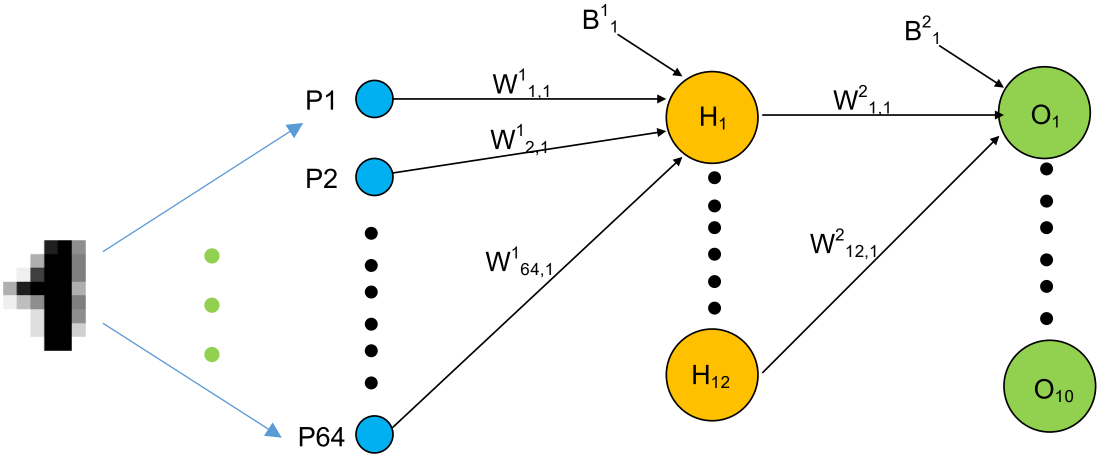


Figure 1: ANN structure

Computing the output value of the hidden layer and output layer nodes involve:

1) Perform weighted sum of previous layer node outputs: $y = \sum_{i=1}^{N} w_i \cdot x_i + b_i$
2) The weighted sum will be fed to a activation function: $f(y) = \frac{1}{1+e^{-y}}$

The weight values w_i are also given as 8-bit fixe point 2's complementary numbers with 5 fractional bits. The circuit that implements the activation function will be provided as an IP block, which can be instantiated in your design.

The project is partitioned into a number of milestones, which are discussed as follows.

1. Milestone 1: Multiplier and Accumulator (MAC) Module Design (40)

In this design phase, you are going to design an MAC module that perform computation $y = \sum_{i=1}^{16} w_i \cdot x_i$. Since both $w_i$and x_i are 8-bit fixed-point numbers, the final output should have 20 bits. The block diagram of the circuit is shown in Figure 2.

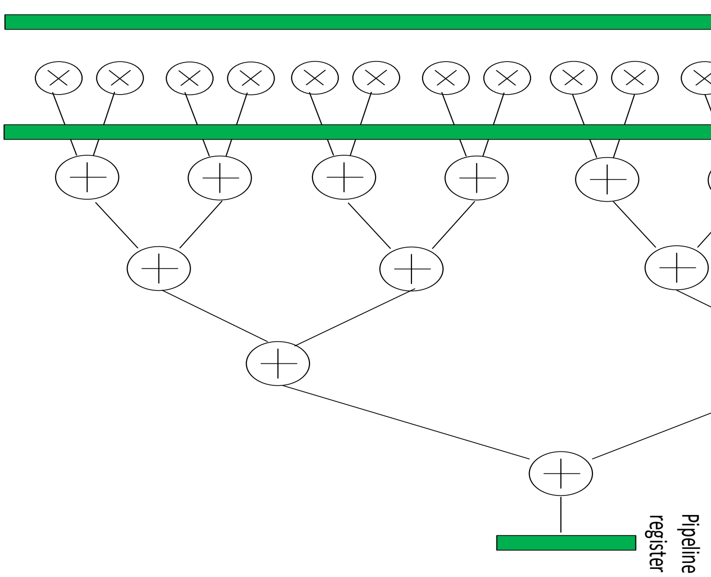


Figure 2: MAC block diagram

To verify your RTL design and synthesized netlist, you are required to write a test bench that reads $w_i$and x_i values from text files (provided to you) and writes simulation results into another text file. Then, you compare your simulated results with expected results, which is given to you in a text file. To get full score, you should show TA the process of simulating the synthesized netlist and comparing your simulation results with expected results. The following text files can be used in your simulation and verification:

- **Digits_hex.txt**: text file containing digit pixel data
- **Weights_hex.txt**: text file containing weight values
- **Result_mac_hex**.txt: expected output when using the above data as input

2. Milestone 2: Accumulator (ACC) Module Design (35)

Computing the output of a hidden node requires adding 64 multiplication results. However, the MAC module only compute 16 multiplication and add them together. Thus, we plan to use the same MAC module four times and accumulate the results as shown in Figure 3. Since MAC output is 20 bits and four MAC results are to be accumulated, the ACC will take four clock cycles to produce a result and the result has 22 bits.

To verify your RTL design and synthesized netlist of the combined MAC and ACC modules, you are required to write a test bench that reads $w_i$and x_i values from text files: weights_hex.txt and digits_hex.txt, and writes simulation results into another text file. Then, you compare your simulated results with expected results, which is provided in text file: **result_MacAcc_hex.txt**. To get full score, you should show TA the process of simulating the synthesized netlist and comparing your simulation results with the expected results.

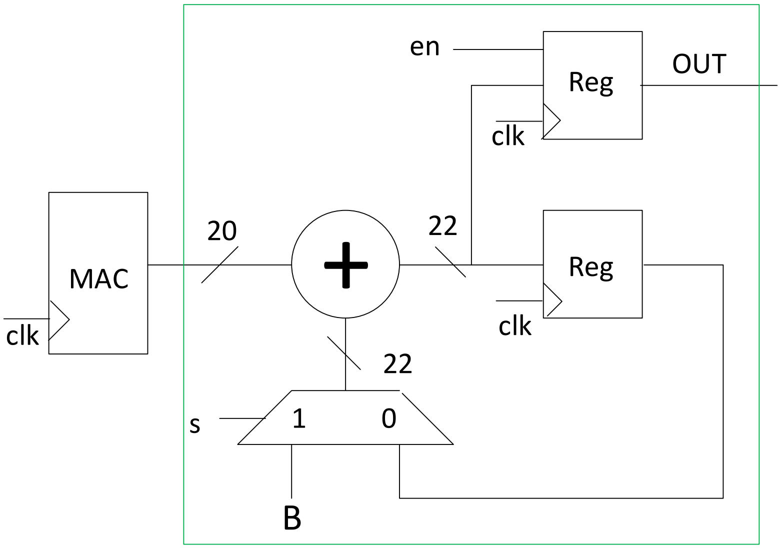
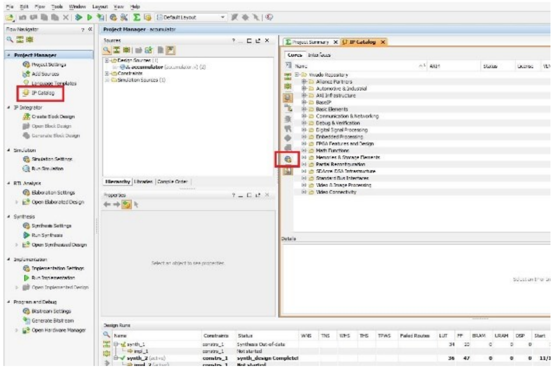


Figure 3: ACC block diagram

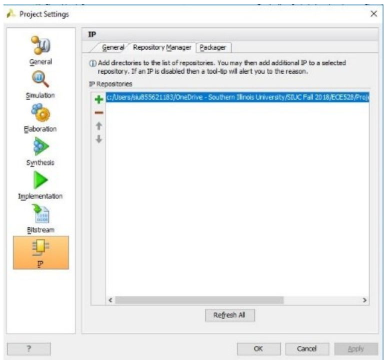
3. Milestone 3: Integrating with sigmoid IP block (20)

The circuit that implements the activation function is provided as a sigmoid IP block. In this design phase, you integrates the IP block with your existing design following the steps below:

- Download the IP file siu.edu_user_sigmoid_1.0.zip and extract the file into a folder, e.g. sigmoid_IP.
- Click **IP Catalog** located on the left navigation panel to bring up the IP Catalog panel. Then, click the IP setting icon in the panel to bring up **IP Setting** panel



Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Failed Routes	LUT	FF	BRAM	URAM	DSP	Start
synth_1	constrs_1	Synthesis Out-of-date							34	22	0	0	0	
impl_1	constrs_1	Not started												
synth_2 (active)	constrs_1	synth_design Complete!							36	47	0	0	0	11/1
impl_2 (active)	constrs_1	Not started												



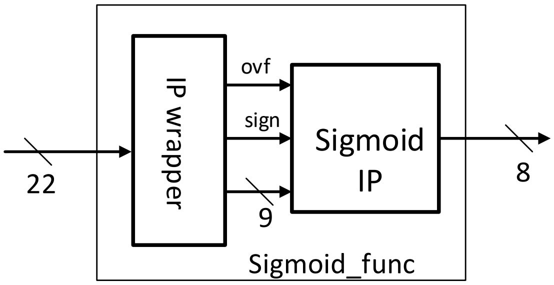
IP

Add directories to the list of repositories. You may then add additional IP to a selected repository. If an IP is disabled then a tool-tip will alert you to the reason.

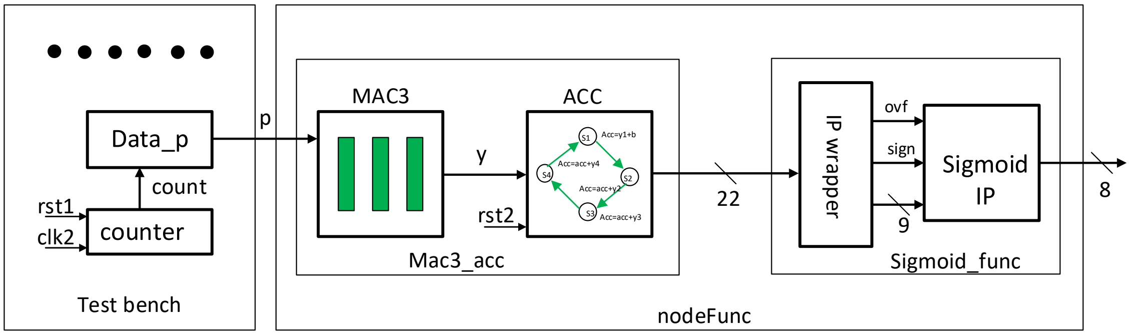
c:/Users/siu855621183/OneDrive - Southern Illinois University/SIUC Fall 2018/ECE528/Proj

- Go to **Repository Manager** and add the folder that contains the extracted Sigmoid IP module files to IP Repositories

- After importing, the sigmoid IP will be listed under **User IP** group. Double click it and select **Out of Context synthesis option** to add the sigmoid module to your project.



Design an interface circuit (IP wrapper circuit) as discussed in lecture. The signals from the IP wrapper circuit to the IP block are shown in the above figure. Use the test bench file (test_sigmoid.v) to simulate your design. Compare the simulation result with the expected output file: **result_sigmoid_hex.txt.**



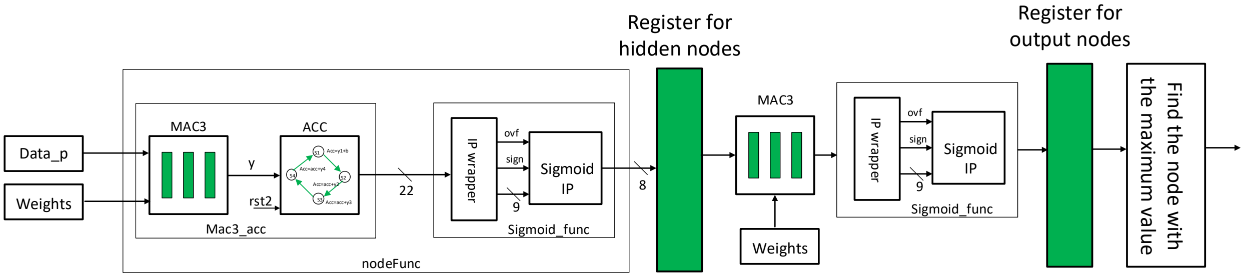
Finally, add the sigmoid_func module to the macc and acc module as shown above. This is the circuit that takes the 64 pixels as input to compute the output of a network node. Write a test bench to verify the synthesize netlist with inputs from weights_hex.txt and digits_hex.txt. The expected output is given in file: **result_nodeFunc_hex.txt**.

4. Design with considerations (5)

Because products designed by engineers become ubiquitous in modern societies and are affecting virtually all aspects of life, there is an increasing emphasis on considering public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors in engineering design process. We will discuss such consideration in lecture. In the design report, you should also discuss how these factors are considered in your design process. You can assume the design is a component of a large system and discuss the impact from the large system perspective. You can also assume the design is being carried out in an industrial setting in your discussion.

5. Bonus points: Complete the neural network circuit (up to 100)

As bonus tasks, you may slightly modify the modules developed in the previous milestone tasks and put them together to complete the entire neural network circuit, whose block diagram is given below. The bonus points earned will depends on the progress of your design.


Data_p
Weights
MAC3
y
ACC
S1
Acc=y1+b
Acc=acc+y4
S4
S2
Acc=acc+y2
S3
Acc=acc+y3
rst2
Mac3_acc
22
IP wrapper
ovf
sign
9
Sigmoid IP
Sigmoid_func
nodeFunc
8
Register for hidden nodes
MAC3
Weights
IP wrapper
ovf
sign
9
Sigmoid IP
Sigmoid_func
Register for output nodes
Find the node with the maximum value