UNIVERSITY OF TEHRAN
Electrical and Computer Engineering Department

Digital System Design with Hardware Description Language
ECE 846 – Spring 01-02
Homework 3: RTL Design – Neural Networks
Due Date: Ordibehesht 8th

Multi-Layer Perceptron:

A multi-layer perceptron (MLP) is a class of feedforward artificial neural networks (ANN). In a feedforward network, the neurons in each layer feed their output forward to the next layer until we get the final output from the neural network. An MLP consists of at least three layers of nodes: an input layer, a hidden layer, and, an output layer as shown in Figure 1.

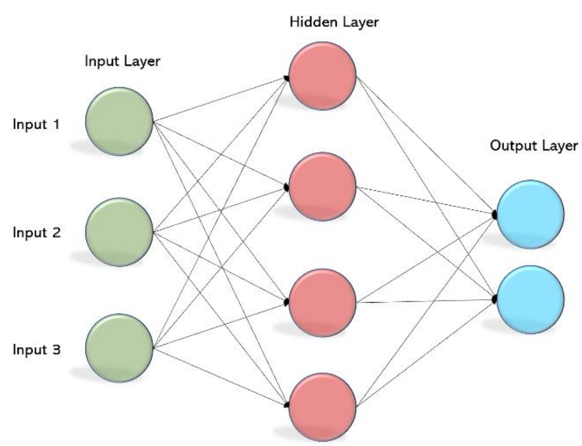

Figure 1: An example of MLP with one hidden layer

Except for the input nodes, each node is a neuron that uses a nonlinear activation function. In recent developments of deep learning the Rectifier Linear Unit (ReLU) is more frequently used as an activation function. Since MLPs are fully connected, each node in one layer connects with a certain weight to every node in the following layer. Each neuron in the hidden layer often has its own bias constant. This bias matrix is added to the weighted input matrix before the hidden layer applies ReLU. Figures 2 and 3 show the MAC operation in a neuron and MLP specification respectively.

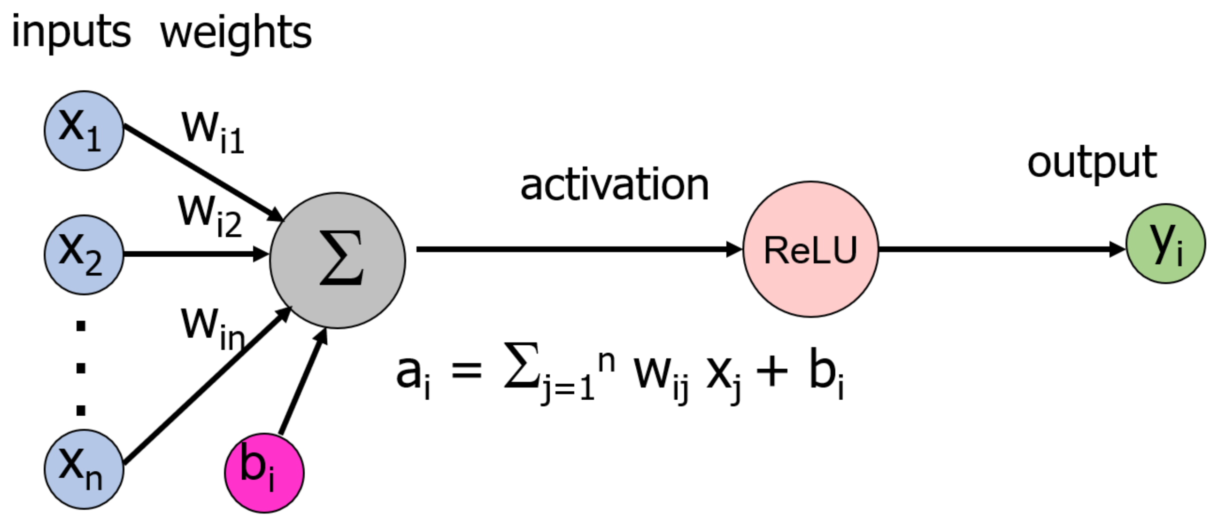

Figure 2: MAC operation in one neuron

Some Properties of MLP architecture:

- No connections within a layer
- No direct connections between the input and output layers
- Fully connected between layers
- The number of output units need not equal the number of input units
- The number of hidden units per layer can be more or less than the input or output units

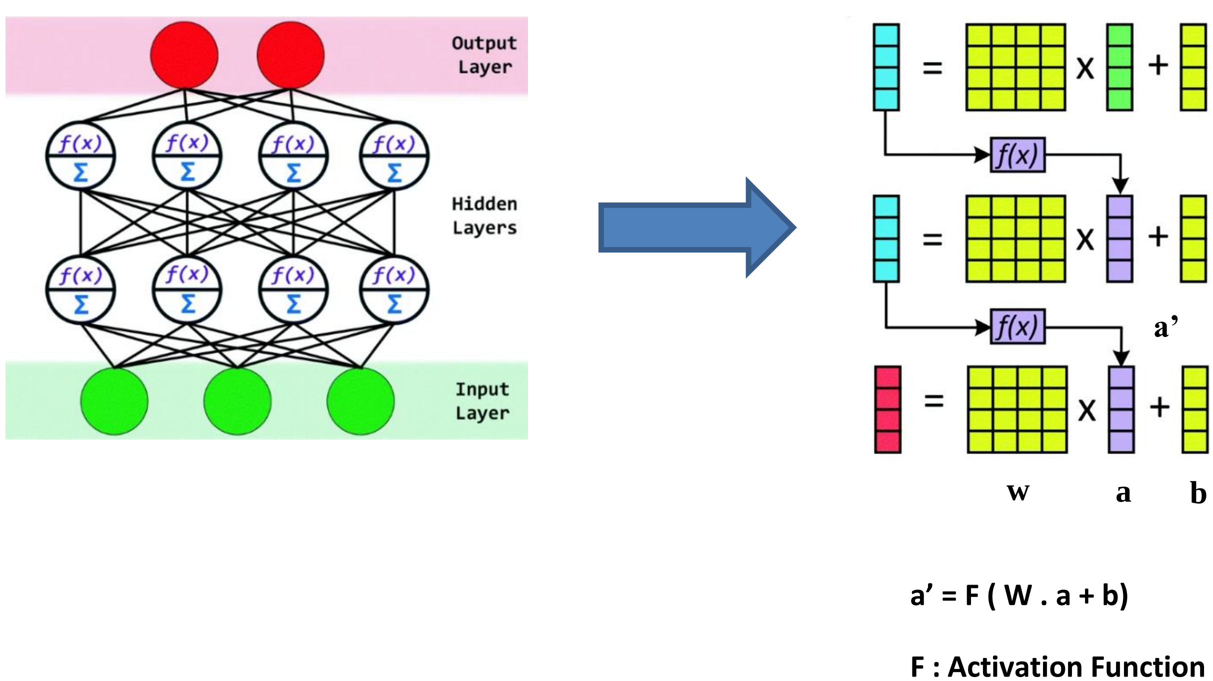

Figure 3: Multi-Layer Perceptron Specification

Problem Description:

The Iris flower data set or Fisher's Iris data set is a multivariate data set introduced by the British statistician, eugenicist, and biologist Ronald Fisher. The data set consists of 50 samples from each of three species of Iris (Iris Setosa, Iris Virginica, and Iris Versicolor). Figure 4 shows these species. Four features, i.e., the length and width of the sepals and those of the petals, were measured in centimeters from each sample. Based on the combination of these four features, Fisher developed a linear discriminant model to distinguish the species from each other.

The perceptron is a simple algorithm intended to perform classification, i.e., it predicts whether the input belongs to a certain category of interest. In this assignment, you should design an MLP to classify the Iris flower species. It receives the features of a flower as its four inputs and recognizes the type of flower as output.

See: https://en.wikipedia.org/wiki/Iris_(plant) to learn more about Iris.

(a) (b) (c)

Figure 4: Iris flower species: a. Iris Setosa, b. Iris Versicolor, c. Iris Virginica

Design Description:

In this assignment, you are to design a circuit to implement an MLP classifier. The dataset, *iris.data* file, contains a set of 150 records under four attributes - sepal length, sepal width, petal length, and petal width. The MLP has four inputs, two hidden layers, and three outputs. Use the weights based on data in the *w-iris.txt* file of the pre-trained network with 10 neurons for the first hidden layer and 8 neurons for the second one, as shown in figure 5.

First of all, you should design a neuron in which the multiplication and addition operations should be done in temporal sequences (sequentially). Then you should use this neuron for generating a layer. Furthermore, instantiations of neurons should be placed in GENERATE statements. The layer and the neuron should be described using GENERICs so that a single description can be adapted to multiple layers. Finally, the discussed MLP classifier can be generated by spatial instantiations of layers using GENERATE statements.

In the MLP design, each layer includes a datapath and a controller. The layer i starts its computations for a set of inputs when a positive edge appears on the input signal *start_i* of that layer. After finishing the required computations for layer i, the output signal *done_i* will be issued. In addition, the MLP architecture has a pipeline structure and you should consider appropriate handshaking between the layers using proper handshaking signals.

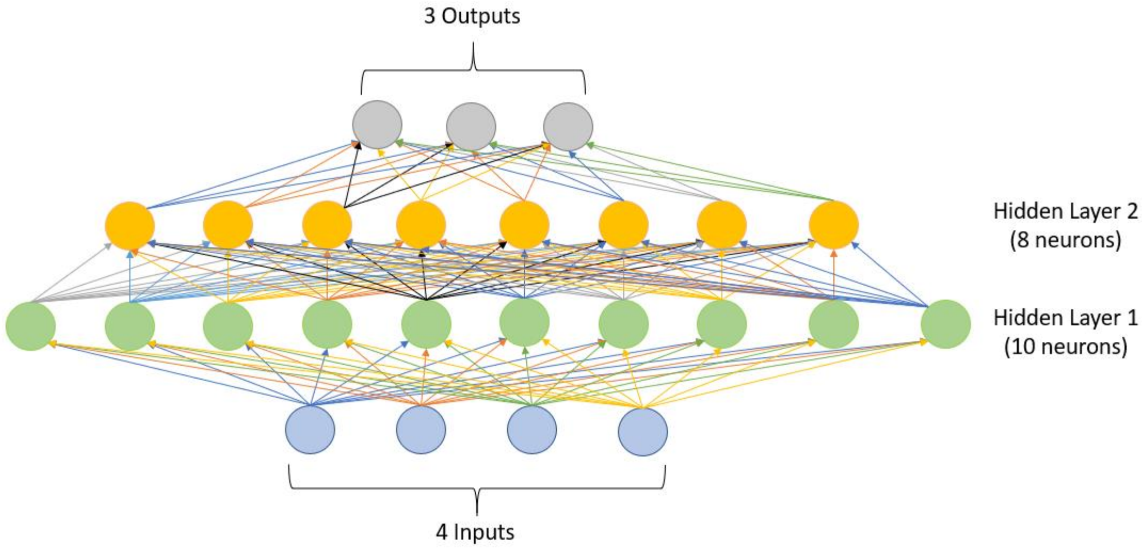

Figure 5: MLP with 2 hidden layers, 10 neurons for the first hidden layer and 8 neurons for the second one

Weights can be defined as 8-bit data. You can also assume that the required weights for each layer are defined as a constant number in each layer. To use features as inputs, you can multiply them by 10 and consider them as 8-bit inputs. Given that there are three types of flowers in the database, we only need 2-bit vectors for the three outputs. The table below shows the output values for the three Iris types.

Table 1: Type of flowers based on the output

Output	Type of Flower
00	Iris-Setosa
01	Iris-Versicolor
10	Iris-Virginica

Hint: Use ReLU as an activation function for two hidden layers but there is no need to use an activation function for the output layer.

** You have the option to train the network based on the Iris dataset and determine the number of hidden layer neurons and the weights for a network accuracy of better than 90%. You will use a different set of weight matrices. You can use 80% of the data set as training data and 20% as test data.

Design Phase:

A. For each level of design, show the complete schematic of your design including the datapath and controller. Use proper variable names as you use in the GENERATE statements.
B. Implement your RTL design in VHDL.
   - Those students who use a memory or buffer mechanism for reading weight numbers, receive a bonus.
C. You are to provide an effective testbench for the design developed, and then test your design with at least 4 sets of data. A *Python* code is provided to model the functionality of the **MLP circuit**. For justification of your result, you must use this *Python* code and compare your result with the *Python* result.

Deliverables:

A. All VHDL codes, with appropriate names.

B. A complete report containing answers to all parts of each question. Your report should include enough design illustration, description, actual data, and output justification. Note that your reports should be well-organized.

Attention:

A. Your design must be synthesizable.

B. Your report should include adequate design illustration, description, actual data, and output justification. Note that reports should be written in English, in IEEE format, and well-organized.

C. For drawing schematics, use Visio or other graphic tools.

D. Make sure you do an independent work and what you submit as your final report includes none but your own work.

E. Compress all files and documents mentioned in the Deliverables section into a zip file and upload on elearn. The name of the zip file must be in this format "YourFirstName-YourLastNme-HW#3".