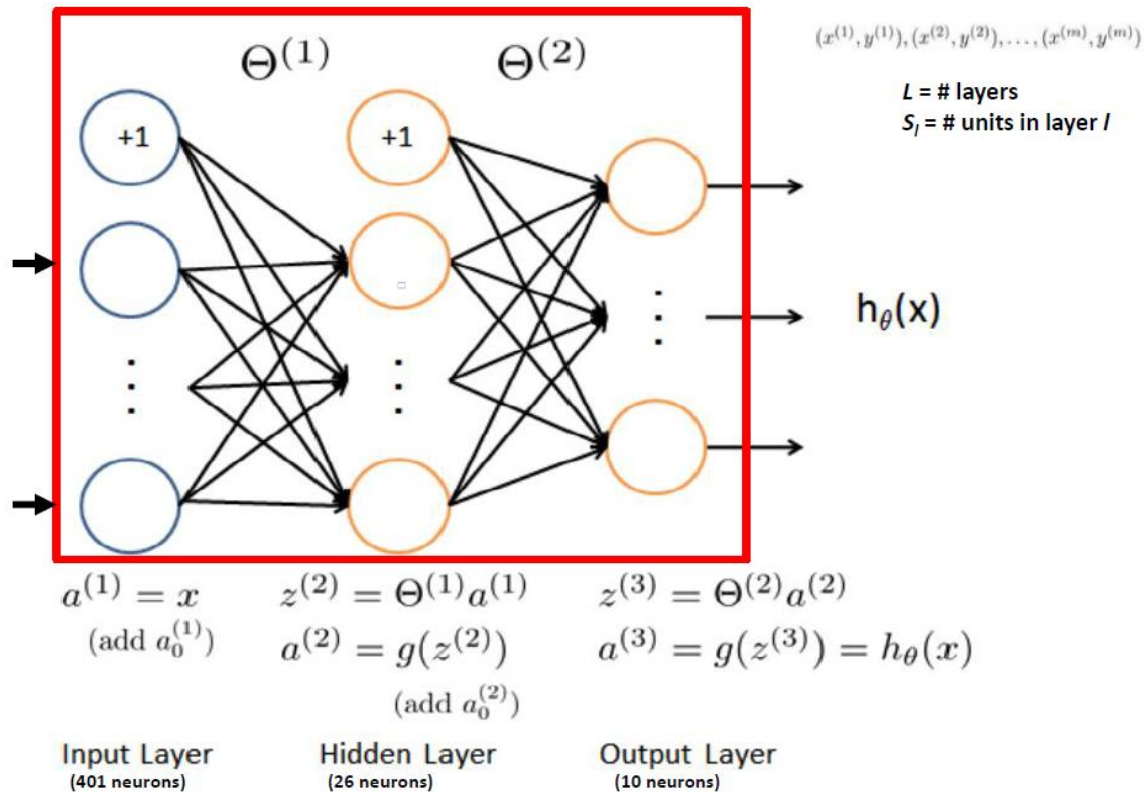## San Jose State University
Department of Electrical Engineering

Mini-Project III (Due Nov. 9 midnight)

Bit-Accurate Matlab (or other SDL) Model for Inference Engine for Digit Recognition

Using each of the three different bit-accurate MACs developed in the previous miniproject, implement the following "complete" forward data path of neuron nodes. Trained weight coefficients should be used from the Matlab run, Theta1 and Theta2. Input activations should be from the Matlab run, X. Also use sigmoid function, and ReLU functions for comparison purpose. For the fixed-point MAC, change the bit widths from 2 to 16, including input activations and weights, discuss how the precision, i.e., error rate of the NN varies, in comparison with μ-law and F16.

Write an insightful report, 10-20 pages, including (lots of) graphs and your discussion.

$$a^{(1)} = x$$
$$(\text{add } a_0^{(1)})$$

$$z^{(2)} = \Theta^{(1)} a^{(1)}$$
$$a^{(2)} = g(z^{(2)})$$
$$(\text{add } a_0^{(2)})$$

$$z^{(3)} = \Theta^{(2)} a^{(2)}$$
$$a^{(3)} = g(z^{(3)}) = h_\theta(x)$$

Input Layer (401 neurons)   **Hidden Layer** (26 neurons)   **Output Layer** (10 neurons)

In addition, your report should include the source code listing, compilation report, and screen shots showing the "bit-accurate" results of the test cases, along with detailed and in-depth discussion of the design process.

Test vectors for a_i's and θ_i's should be obtained from the Matlab code after the final training.

NOTE: To show the authenticity of your work, attach the last two digits of your student ID to every signal and variable name used in your code. Each of the screen captures of your code and simulation results should bear these signal names.