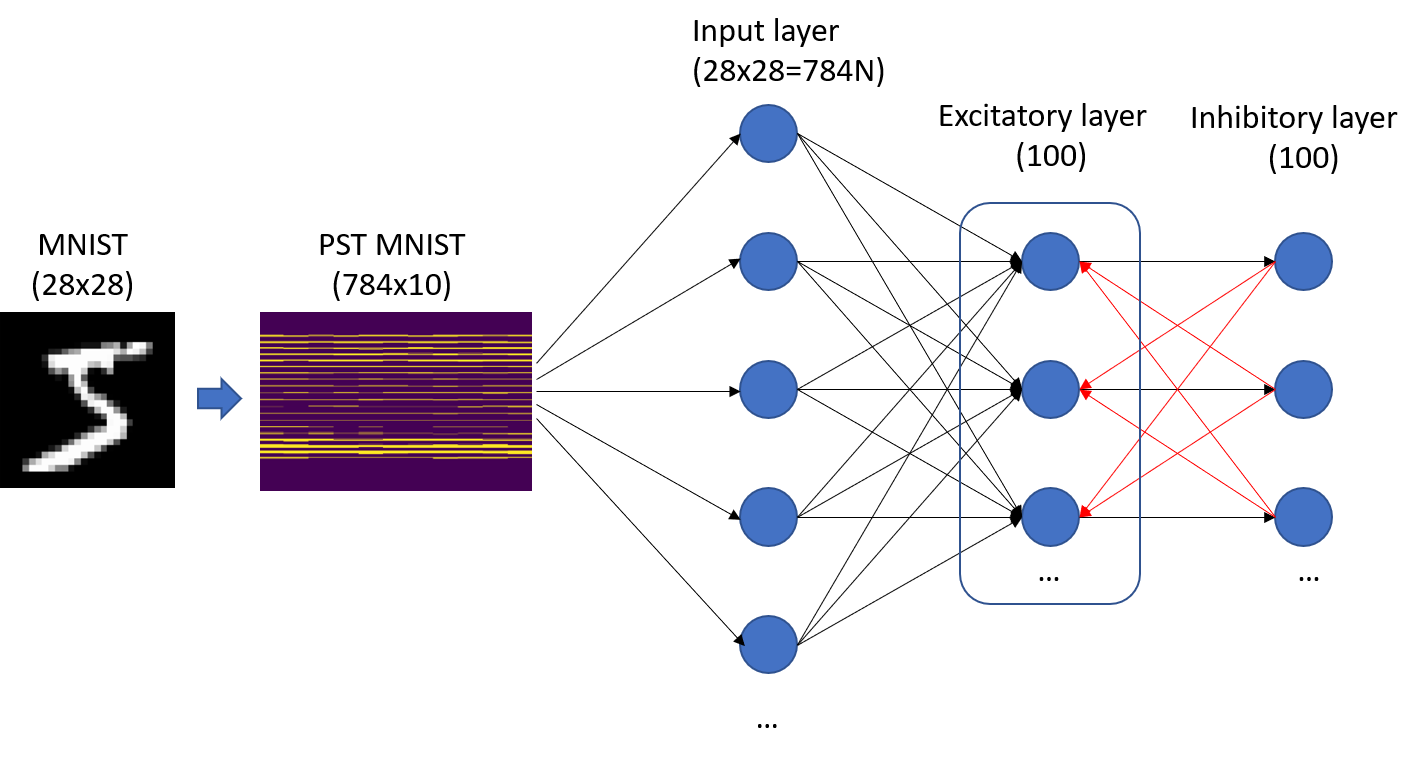
Project Report: Phase II

# Spiking Neural Network (SNN) Design

Our SNN consists of 3 layers of neurons as illustrated in fig (1): input layer, excitatory layers, and inhibitory layers. MNIST data set is used training and test data. Since MNIST data set contains images with resolution of pixels, there are neurons in the input layer, each representing information of a single pixel of the original image. The neurons in the input layer are encoded to actively fire spike randomly at specific rates determined by the intensities of the pixels represented by them. If a pixel is very bright, its corresponding neuron will fire very frequently and vice versa. The second layer is an excitatory layer consisting of 100 neurons that are responsible for classification. The third layer also consists of 100 neurons that form a recurrent network, which functions as lateral inhibition. The inhibitory neurons will try to suppress other neurons that are not firing. At end of each classification or training, total fire count of each excitatory neuron will be recorded and used to classify the input data.



**Fig 1.** Illustration of the SNN. MNIST data will be preprocessed to form Poisson spike train, which is understandable by the SNN. The input layer will then fire corresponding spike repeatedly and randomly at specific rates to excitatory neurons. The process will last for to in simulation depending on the accuracy requirement.

# Neuron Dynamics

The input neurons are encoded to fire randomly at pre-determined rates based on their corresponding pixels as shown in the following formula:

, where is the fire rate of input neuron, is the pixel value of pixel, 255 is the maximum possible pixel value, and is the maximum fire rate set by us. The maximum fire rate will be increased if spikes are not enough for form any meaningful results or decreased if there are too many spikes and the network is saturated. If a pixel has value of 100, and we set the maximum fire rate to , its corresponding neuron will have a fire rate of .

The reset neurons (excitatory and inhibitory) will have a leaky model that means their potential will exponentially decrease if there is no input. Some leaky models are very complicated and considers conductance of connected synapses. Here we only use a very simple one to simplify the system. The potential dynamics of any neuron is described by the following differential equation:

, where is the potential change rate, is reset potential, is the current potential of the neuron, and is the time constant of the neuron potential, which determines how fast the neuron potential will decay. In our network, we set the reset potential to 0 so that the potential of every neuron will eventually decay to zero without any spike input. With frequent spike inputs, the potential will reach threshold potential and the neuron will fire a spike through output synapse. The threshold is not static. Whenever a neurons fires a spike, its threshold will increase by certain amount so that the neuron will not dominate the network. This is called homeostasis, which describes the behavior of a biological neuron that would become harder to fire if it is stimulated too frequently. Therefore, the threshold dynamic is also described by following equations:

, where is the threshold potential, is the rest (minimum) threshold, is additional threshold added to neuron as homeostasis effect. will exponentially decay if no new spike is fired as described in the differential equation. is the change of the additional threshold that will happen when a neurons fires a spike. is set to for our network. So a neuron’s threshold will increase by whenever it fires.

# Synapse Dynamics

The synapses between input layer and excitatory layer will exhibit STDP property described by the following exponential equations:

, where is the weight update to synapses whenever a post-synaptic neuron fires, is the learning rate, is the firing time of the post-synaptic neuron, is the last time the pre-synaptic neuron fires. If post-synaptic neuron fires shortly after pre-synaptic neurons fires, the synaptic weight will increase, and it will decrease if time between two spike fires are too long. Such weigh update rule is also the unsupervised learning rule of our network.

# Learning Rule

Training data consist of 60000 MNIST images and is split into 6-fold for cross validation. Synaptic weights are randomized at the beginning of the training. Each image is converted to PST and feed to the network and run for . The network will wait for before next image is fed to the network to prevent residual neuron potential contamination. The network will automatically adjust its synaptic weights to adapt the training data. After all training data is learned by the network. Average fire count of each excitatory neuron for each image class is calculated and training data label is used to classify the neurons with the highest fire count to the corresponding label.

# Model Compression (Pruning)

Before the training, we set weight threshold, , for synapses. For each batch during the training process, synapses with weights lower than will be set to zero first, after all training batches are done, synapses that still have weights lower than will be deleted from the network, also known as pruning.

# Hardware Simulation

After the network is trained. The network details will be sent to hardware team for further hardware simulation

# Results

Currently the team is still working on building the network. We are still waiting for the results. We will update once we make new progress.