UNIVERSITY OF CENTRAL FLORIDA

DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING

Initial Project and Group Identification Document

Luke Minks, EE
Deven Morone, EE
German Romero Castro, EE
Justin Sapp, CpE

October 4th, 2019

Group Number: 31

1.0 Introduction

Senior Design Group 31 is composed of four intelligent and hard-working individuals. The group members are Justin Sapp, German Romero Castro, Deven Morone and Luke Minks. Justin is pursuing a major in Computer Engineering, while Deven, German and Luke are pursuing majors in Electrical Engineering. A working title for the project is: A Neuromorphic Circuit for Hardware-Based Machine Learning. This document includes key details on what the project will be able to do, how it will function, and how it will be executed.

2.0 Project Narrative Description

Over the past decade, major advances have been made in machine learning technology, with the latest machine learning-based techniques showing impressive results in applications ranging from voice recognition to the generation of abstract artwork. However, the most common implementations of machine learning algorithms are currently software-based systems that perform a large number of mathematical operations digitally, and due to the computational expense of these operations, tasks such as voice-to-text transcription cannot be performed on inexpensive mobile devices but must instead be sent out to central servers. Hardware neural network implementations, capable of massively parallel analog computations, may soon offer an inexpensive way to accelerate these operations on mobile platforms and other resource-constrained devices.

For our Senior Design project, we plan to create a hardware-based implementation of a neural network that can be used for machine learning and classification/recognition tasks. While the emerging devices required to build a very large-scale neural network may not be ready for production for several years, we believe that we can gain valuable experience in neuromorphic hardware design by building a small-scale, yet highly capable, neural network circuit using commercially available parts.

We plan to design the neural network as a Multi-Layer Perceptron (MLP), which uses several layers of artificial "neurons" each with multiple "synapse" inputs. A scalar numeric weight value is assigned to each synaptic input, and the neuron's output is determined by multiplying each synapse's input value by its weight value, taking the sum of those products, and applying that sum value to some non-linear activation function. Common activation functions include the sigmoid function, hyperbolic tangent, and rectified linear unit.

A fully-connected network of these neurons and synapses can be "trained" using a set of input data samples that are paired with labels indicating the network's expected output. Inputs are applied to the synapses of the first neuron layer, and outputs are sampled from the neurons in the final layer. We plan to use an algorithm known as gradient descent optimization to implement training by backpropagation, where each layer's outputs, starting with the final layer and moving backward, are compared to the expected output values, and that layer's synaptic weights can then be adjusted based on the activation function's derivative in an attempt to minimize the output error percentage. This process is repeated for the entire network over multiple "training epochs" until some minimum level of error is attained.

In most neuromorphic circuits, synaptic weights are represented by variable resistances. Emerging non-volatile memory technologies such as memristors/resistive RAM, phase change memory, and magnetic tunnel junctions may soon offer nanometer-scale, low-cost devices that can store these synaptic weights, but most current research works on neuromorphic hardware are based on custom-fabricated VLSI devices that are not available to the general public.

Instead of using novel devices, we plan to use common digital potentiometers (with onboard memory) to implement non-volatile synaptic weight values in our circuit. While these devices may be too large and expensive to use in a circuit with millions of synapses, they should be useable in a circuit on the scale of several hundred synapses. We plan to implement the neurons' summing and non-linear activation functions using operational amplifiers.

3.0 Requirements/Specifications

Weight	<10kg
Footprint area	<1m x 1m
Inputs (5x5 pixel image)	25 inputs
Functionality	Ability to recognize and differentiate between two handwritten characters
Output latency	<500ms
Supply Voltage	<18Volts rail-to-rail
Accuracy	>50%
Power Consumption	<50Watts

Table 1: Specifications Table

		Neuron count	Size	Cost	Power	Training time
		+	-	-	-	-
Speed	+	↓	↑			↑↑
Accuracy	+	↑↑	↓↓	↓		↑
Size	-	↓↓		↑↑	↑	
Cost	-	↓↓	↑↑			↓
Flexibility	+	↑				↑↑
Requirement targets		30+	<1mx1m	<$900	<50 W	<1 hour

Figure 1: House of Quality

		Neuron count	Size	Cost	Power	Training time
		+	-	-	-	-
Neuron count	+		↓↓	↓↓	↓↓	↑
Size	-	↓↓		↑	↑	
Cost	-	↓↓	↑			↓
Power	-	↓↓	↑			↓
Training time	-	↑		↓	↓	

Figure 2: Roof of Quality

4.0 Project Block Diagrams

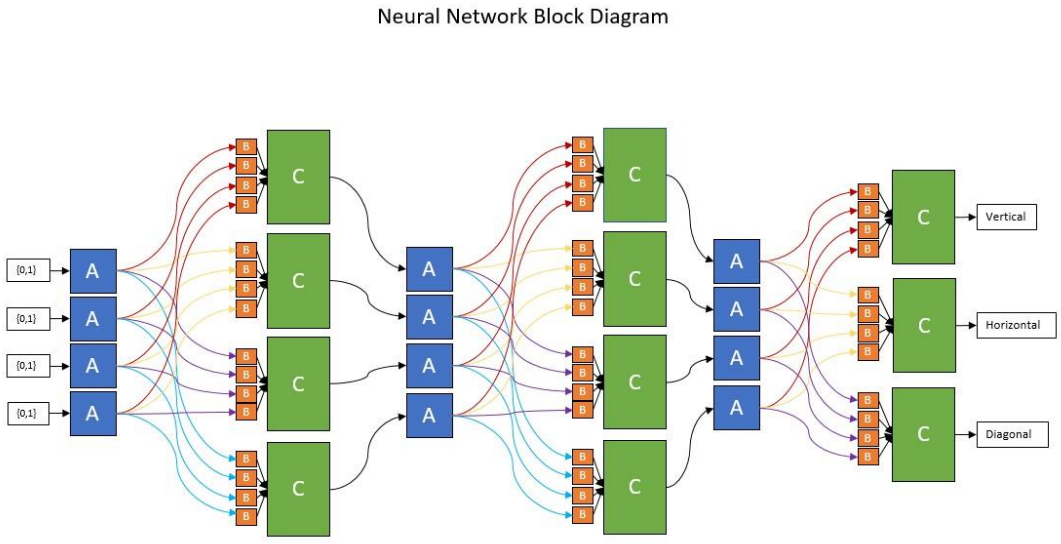

Figure 3: Neural Network Block Diagram

The Neural Network Block Diagram represents the main hardware components of our project. The 3 blocks it consists of (A, B, and C) perform the 3 main functions our Neural Network must achieve. It must be able to multiply, accumulate, and introduce non-linearity inside the network to be trained properly using machine learning algorithms. The blocks of this network diagram will be made up of analog circuitry that will execute these functions while being trained to correctly recognize user input handwritten characters.

The first block, A, is made up of a line driver circuit. This circuit will provide buffered and inverted source voltages to the next block in the sequence. The voltages that this circuit provides will be fed to block B, which is the circuit that will supply the synaptic weights that the input source voltages will be multiplied by. These synaptic weights will be adjusted, at the system level, by our MCU.

Finally, block C of the diagram contains the accumulation and activation function circuits of our Neural Network. The first component of this block must add all the outputs of the B blocks from the previous stage of the diagram, thus completing the multiplication and accumulation requirement of the network. This will be achieved using an inverting summing amplifier circuit. Afterwards, these accumulated and weighted inputs will be non-linearized by using another analog circuit to implement the activation function of our network. This non-linear activation function will be implemented using another operational amplifier circuit with an external rectifier circuit.

Note that this Neural Network layout consists of a 2x2, 4-pixel pattern recognition circuit. Our goal for this smaller scale network is to train it to distinguish the difference between horizontal, diagonal, or vertical lines in 4-pixel user inputs. Once this network is fully implemented and tested, we will look to expand the size of the network to a 5x5, 25-pixel network that will be able to recognize the difference between handwritten user input characters. This should be achievable as the only difference would be a larger number of components and PCB’s necessary to construct the network.

Role	**Person(s) Responsible**
Commercial Power Supply	Deven, German
Network	Deven, German
MCU	Justin, Luke
Touchscreen Input	Justin, Luke
Status LED Array	Deven, German, Luke
Training Data Set	Justin

Table 2: Subsystem Responsibilities

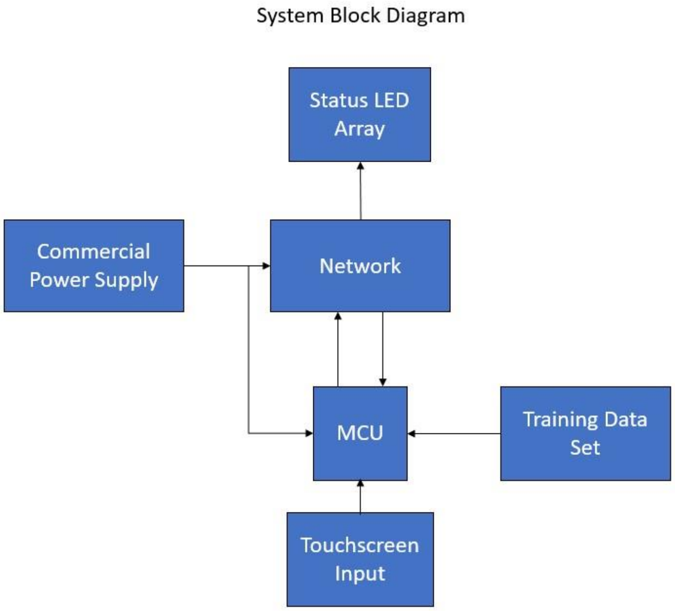

Figure 4: System Block Diagram

Our finalized project will be comprised of six main components that can be seen in the block diagram above. We will be utilizing a commercial power supply to provide power to all electrical components. This should be appropriate since we will meet the project’s hardware expectations with what has been explained in the paragraph above.

The MCU we choose will be responsible for the training and adjustment of our network’s circuit. Data sets will be provided to the MCU such that the network can be trained to recognize the touchscreen user input. User input will be provided by a touchscreen pad in which the user can write an alphabetical character or draw a vertical, horizontal, or diagonal line. This input will be processed and transmitted to our neural network as digital pixel data through the MCU.

Once this pattern recognition or handwritten character recognition (depending on the scale of the network) has been done, the result will be displayed in an LED status array in which the user can see either the orientation of the line they have drawn or the correct character they have entered.

5.0 Estimated Project Budget and Financing

Since costs scale exponentially with the number of neurons in the circuit, the cost estimates here assume a 2x2 test network and a 5x5 final network with all components purchased at full retail price. Depending on how the intermediate circuits are constructed and which component and PCB designs are used, it may be possible to recycle a significant portion of the investment between network iterations. While the individual components are generally inexpensive, the large number of op amps and potentiometers needed for a 5x5 or larger network would be relatively expensive, as it is approximately $1 for a 4-potentiometer EEPROM chip and $1 for a 4-op amp chip. Each synapse requires either 1 or 2 potentiometers dependent upon our chosen approach; additionally, in a worst-case scenario dependent upon the number of synapses that can be driven individually, 2 op amps would be required for a complementary buffer pair to avoid unintended voltage drops, which would corrupt the final result. On top of the hardware requirements for each synapse, each neuron requires 2 op amps to properly simulate the sigmoid activation function, in addition to a number of resistors and diodes. Because the addition of a neuron also necessarily means the addition of another synapse to and from each neuron in the adjacent layers, it becomes progressively more expensive to add neurons to the network.

Budget item	Estimated cost	Comments
Prototyping dev boards	$50	Already acquired
PCBs	$200	Assuming 2 PCB iterations
SMT processors for PCBs	$50	
Potentiometers	$200	Assuming 5x5 network
Amplifiers	$200	Assuming 5x5 network
Miscellaneous components	$150	Diodes, resistors, LEDs
Input touchscreen/camera	$50	Dependent upon method
Total Budget	$900	

Table 3: Estimated Cost

6.0 Initial Project Milestones

The milestones listed here are general estimates based upon both our current progress and general rate of progress to date. These goals are somewhat pessimistic; they will most likely – and ideally – be met early. As a significant amount of research has already been performed prior to the beginning of the semester, we have begun prototyping the constituent portions of the network. Because we have already successfully tested a single neuron’s sigmoid activation function, the next immediate step is to construct a basic 4-input network and attempt to train it to perform basic pattern identification. Beyond this, our goals are to streamline the training algorithm and correct any chronic identification errors, potentially build a larger intermediate network to scale up, and design the final network’s PCB and peripheral hardware. Our current goal is to ultimately produce a 5x5 network able to recognize a few select letters, though the number of inputs and outputs in the network may increase or decrease as we assess the difficulty and complexity of the project.

Date	Goal	Comments
September 30	4-pixel test network built on breadboards	
October 15	Complete initial training algorithm for 4-pixel test network; integrate microcontroller into network to prepare for training	2 weeks to modify training algorithm to be compatible with physical network
October 31	4-pixel network trained and evaluated; 60-page draft completed	2 weeks for troubleshooting and adjusting algorithm
November 15	Finished refining algorithm; potentially construct intermediate network if necessary	Additional 2 weeks for algorithm streamlining or ordering parts if necessary
November 21	Intermediate network trained and evaluated	May not be performed depending on 4-pixel network performance
December 2	Finalize part list that will be used for first iteration of final design; final documentation completed and submitted	Parts list dependent upon performance of prototype networks

Table 4: Senior Design I Milestone Dates

Date	Goal	Comments
January 15	Network hardware component of PCB design complete (neurons, synapses)	6 weeks including winter break to collect footprints and design PCB network layout
January 21	Power, microcontroller, communication components of PCB finished	Additional week to finish layout of peripherals and power
February 7	First iteration PCB ordered and populated	1-2 weeks to obtain and build
February 14	First iteration training completed	1 week due to streamlining from earlier networks
February 21	PCB updated and re-ordered if necessary	Additional week to troubleshoot and update design
March 7	Corrected PCB obtained, built, and retrained	2 weeks to obtain PCB/parts and retrain/troubleshoot
March 21	Finalized training and hardware	
March 31	Complete integration of sample reader	Dependent upon overall progress and chosen method of reading new samples
April 15	Network organized and completed	2 weeks to neaten up setup and smooth any remaining issues

Table 5: Senior Design II Milestone Dates