Open SI ASIC Chip Design

Design Document

Sdmay23-28

Client and Advisor: Dr. Henry Duwe

Tyler Green, Katherine Gisi, Aaron Sledge, William Zogg, Fulai Zhu

https://sdmay23-28.sd.ece.iastate.edu

sdmay23-28@iastate.edu

Executive Summary

Development Standards & Practices Used

Engineering Standards

IEEE 1801-2018 - Design and Verification of Low-Power, Energy-Aware Electronic Systems is a standard defines a way to express the power intent of some electronic design. This standard would apply to our project when comparing our ASIC design's power consumption to that of other integrated circuits, either those in the efabless program or others which would be relevant. It can also help us determine which portions of our circuit are the most power intensive, and could be further optimized.

IEEE 1076.4-1995 - VITAL ASIC Modeling Specification is a standard that creates aa model for effective and efficient use of hardware design languages in ASICs. We will use this standard throughout our design for structure, clarity, and repeatability.

IEEE 1500-2022 - Testability Method for Embedded Core-based Integrated Circuits is a standard to guide designing for testability. We will be using this standard to navigate how we design our testing.

Summary of Requirements

Physical Requirements

1. Logic must fit within the space provided by efabless

Environment Requirements

1. The project must be posted on a publicly accessible git-compatible repo
2. Project is designed utilizing open-source tools (constraint)
3. Project should pass checks in caravel harness GitHub repository

Design and Functionality Requirements

1. Logic represents at least one neuron in a Spiking Neural Network
2. The project must be targeted on the currently supported SkyWater Open PDK for the 130nm process
3. Projects must use a common test harness and padframe based on the Caravel repo.

4. Design uses a digital configuration
5. Design has some reasonable amount of novelty
6. Design must successfully pass the Open MPW precheck tool
7. Design should implement and pass a simulation test bench
8. Functionality of fabricated design is software testable

Documentation Requirements

1. Repository should have a README file
2. All documentation should use inclusive language
3. Project mistakes and learning outcomes should be well documented

Applicable Courses from Iowa State University Curriculum

The courses that have applicable content to this project are as follows:

- CprE 281
- CprE 288
- CprE 381
- CprE 487
- EE 330
- EE 435
- EE 465

List of figures/tables/symbols/definitions

1 Team

1.1 Team Members

Tyler Green, Katherine Gisi, Fulai Zhu, William Zogg, Aaron Sledge

1.2 Required Skill Sets for Your Project

Required:

- Burning desire to see a working digital hardware ASIC
- Value teamwork and helping others succeed.

Preferred:

- Familiarity with HDL design and testing, synthesis processes, preferably also for ASIC designs
- Experience with analog design
- Experience with PCB design
- Experience with MCU programming
- Experience with version-control software

1.3 Skill Sets covered by the Team

Required:

- Whole team
- Whole team

Preferred:

- Katherine Gisi, Tyler Green, William Zogg,
- William Zogg
- Aaron Sledge, William Zogg
- Whole team
- Tyler Green

1.4 Project Management Style Adopted by the team

Our group will be using the agile project management style. One major component of our project is doing many iterations on several pieces of the project, with emphasis on both design and testing. It is likely that we will need to go back and rework components in order to meet the needs of the Caravel project (overall size, ability to harden, etc.) We will also be seeking input from our client over the course of the project because of our goal of developing a curriculum around chip design for new students in CprE.

1.5 Initial Project Management Roles

Katherine Gisi: Spokeswoman, SNN neuron design, Leader

William Zogg: SNN neuron Design

Tyler Green: Software tool research, Environment setup

Aaron Sledge: Software tool research & Documentation

Fulai Zhu: Documentation

2 Introduction

2.1 Problem Statement

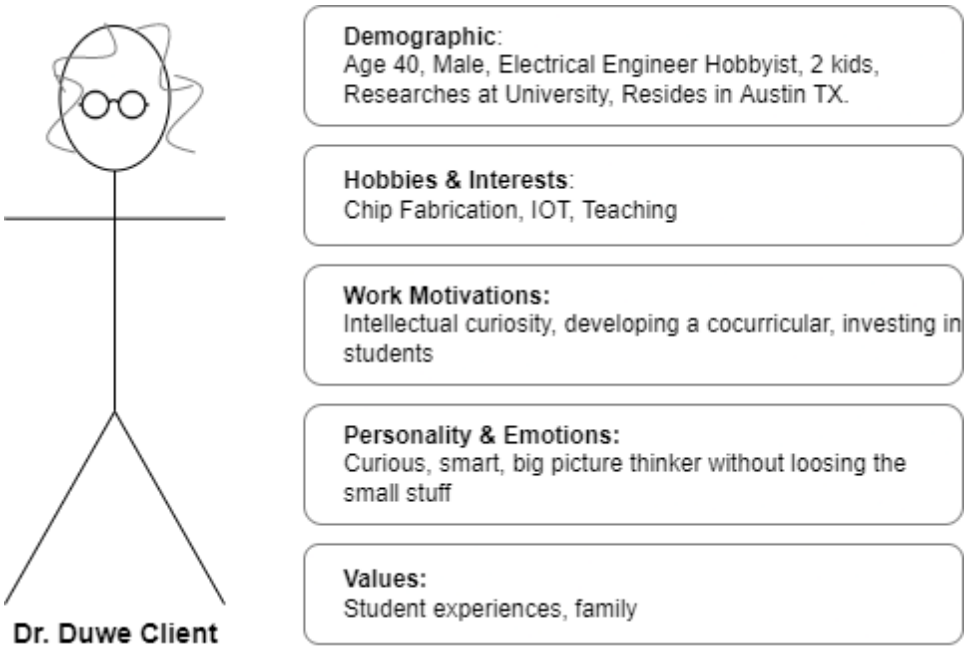
Iowa State University's ECpE department does not offer a chip design curriculum for those that are interested in the subject. This is an issue especially with the increasing demand for people in the workforce that can design complex chips. So, our group is trying to aid in laying down the groundwork for getting a chip design curriculum started at ISU by attempting and documenting our experiences designing an ASIC.

2.2 Intended Users and Uses

The general user groups of this project are threefold. The present users are those that will be continuing to build the infrastructure surrounding open-source ASIC bring-up at Iowa State University. These include our senior design team, future senior design teams, and our client who is a professor interested in building a cocurricular in line with this project. The future users will be those students of that potential cocurricular. Lastly, the broader users will be the open-source silicon community.

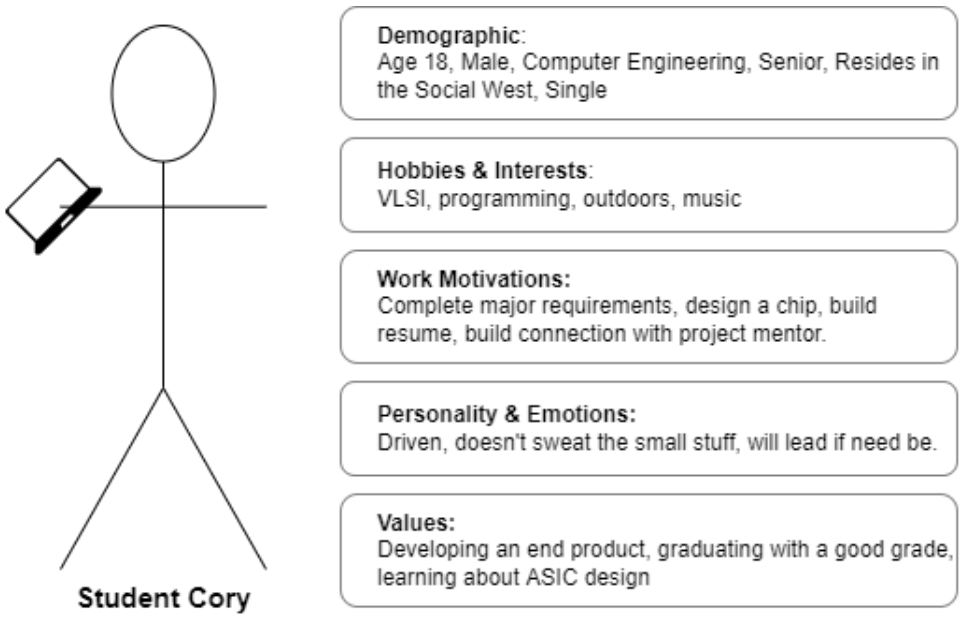
The present users who are partnering with our team to build infrastructure at ISU are described as outlined below.

- Key Characteristics: Interested in chip design, wants a working chip at the end of the day, looking to learn and share knowledge, have a desire to show their aptitude for chip design for future work-related opportunities.
- Needs: Documentation on processes involved, documentation on how to operate open-source design tools, solutions to potential problems, example projects to build that knowledge base.
- Benefits from this project: Knowledge gained from chip design process, knowledge of open-source design tools, ability to teach others, have a novel project to present to employers for future work-related opportunities.



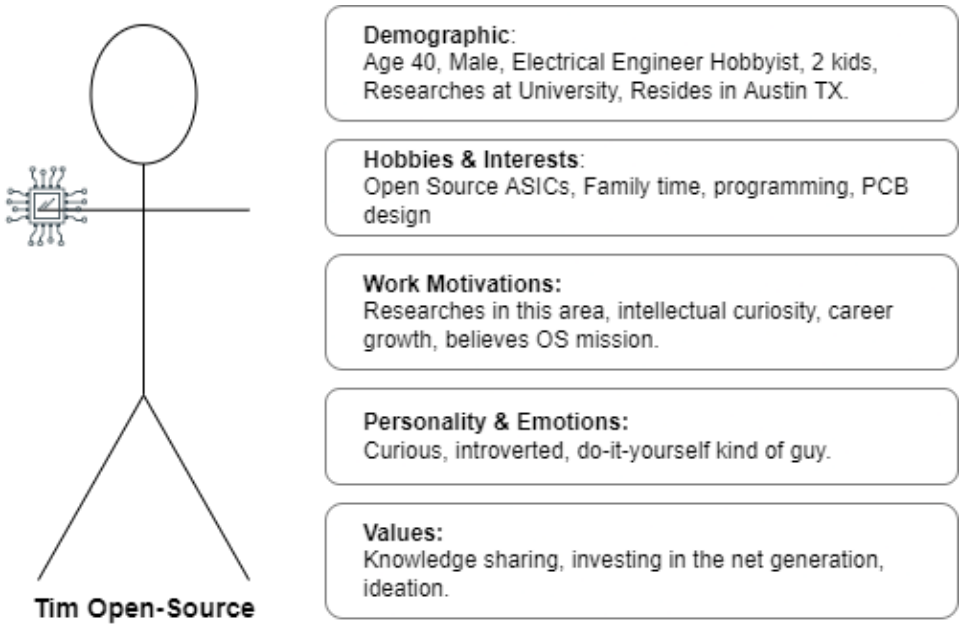
The future users who may be involved in a class based on the work done by senior design projects such as this one can be characterized similarly.

- Key Characteristics: ECpE students, enrolled in a chip design course, want an end product/interested in something to hold in their hand. Additionally, future students of CprE 487/587 want the best class experience and labs possible.
- Needs: A relatively smooth class experience, examples to draw from, modularity of process flow. For 487/587 students, a working, reliable spiking neural network ASIC that can be used for a class lab.
- Benefits from this project: Completed design example, documentation for course work, knowledge given to the professor who may teach this course. For 487/587 students, a new piece of hardware to further the lab experiences possible for the course, have a novel project to present to employers for future work-related opportunities.



The open-source community is a user by nature of this project. Open-source silicon is a relatively new field, and the program that will be participated in for the fabrication of this chip was created to build the OS ecosystem in general. The attributes of the community as listed below.

- Key Characteristics: Participating in similar projects, may or may not have previous chip design experience, possibly looking to improve upon early iterations of a design
- Needs: New projects released and developed using fully open-source tools and information, well-documented instructions on how to use the product
- Benefits from this project: The open-source community will gain a new resource for a spiking neural network ASIC, as well as documentation and code for continuing/modifying an existing design



2.3 Requirements & Constraints

Physical Requirements

- Our logic must fit within the space provided by efabless, specifically in a 10 square mm area (constraint)

Environment Requirements

- The project must be posted on a git-compatible repo and be publicly accessible and contain the following items:
    - The Caravel test harness with the SNN occupying the user space for the layout.
    - The GDSII in a compressed format
    - A makefile with targets to compress/uncompress, and clean the project
    - All source code required to generate the GDSII including any third-party components
    - The makefile should also include verify target to execute a test verification suite for the design
    - LICENSE files for the top-level project as well as each of the third-party components used
- Project is designed utilizing open-source tools (constraint)
- Project should pass checks provided in caravel harness GitHub repository before submitting to efabless (constraint)

Design and Functionality Requirements

- Logic represents at least one neuron in a Spiking Neural Network

- The project must be targeted on the currently supported SkyWater Open PDK for the 130nm process
- Projects must use a common test harness and padframe based on the Caravel repo.
- Design uses a digital configuration
- Design has some reasonable amount of novelty
- Design must successfully pass the Open MPW precheck tool, including LVS and DRC clean using the referenced versions of OpenLane flow
- Design should implement and pass a simulation test bench for their design integrated into Caravel
- Functionality of fabricated design is software testable

Documentation Requirements

- Repository should have a README file
- All documentation should use inclusive language
- Project mistakes and learning outcomes should be well documented for future use by senior design teams and possibly a future curriculum

2.4 Engineering Standards

IEEE 1076.4-1995 - VITAL ASIC Modeling Specification: Since our group is designing an ASIC according to a preset standard, we are therefore able to clearly communicate all aspects of our design to other engineers as necessary.

IEEE 1500-2022 - Testability Method for Embedded Core-based Integrated Circuits: Our group is creating an ASIC that will be tested, and those tests will follow a testing standard that will allow for others who come across our open-source design to easily understand what it is that we are doing. And validate the testing that we have done on our design.

IEEE 1801-2018 - Design and Verification of Low-Power, Energy-Aware Electronic Systems: This standard applies to our design because it will allow for others to analyze our chip timing and power consistently across a broad set of electric design automation.

3 Project Plan

3.1 Project Management/Tracking Procedures

Our group will be using the agile project management style. One major component of our project is doing many iterations on several pieces of the project, with emphasis on both design and testing. It is likely that we will need to go back and rework components in order to meet the needs of the Caravel project (overall size, ability to harden, etc.) We will also be seeking input from our client over the course of the project because of our goal of developing a curriculum around chip design for new students in CprE.

Tracking our progress will be done using a combination of Git and Teams. Using Git, we can keep our work organized and manage conflicting changes on the same pieces of code. We can also have backups of our work in case we need to find where something started to break/fail tests. In addition to Git, we can use Microsoft Teams to communicate with each other and set up tasks and dependencies that can be assigned to group members, just like Trello.

3.2 Task Decomposition

The bring up of our design will proceed according to the following task decomposition. Each task is categorized into 3 phases. This will allow us to follow the agile management plan within a phase but still give us a structure to move forward.

Task Decomposition	
Task	**Goal Date**
Phase 1 Backround Developement	**11/7/2022**
Sample Process	10/24/2022
Completer User_Adder Exmple	
Every Member Have Design Environment Set-Up	
Make updates and iterations to sample	10/31/2022
Do something outside of the sample with the adder	
Possibly do layout visualization	
Learn about Spiking Nueral Networks	10/24/2022
Deterime what our design will do	
Know theorehtical Input and Output	
Pen and Paper Logic SNN	11/7/2022
Create Initial Design Sketch	
Create Design Flowchart	
Phase 2 Design	**12/12/2022**
Determine High-Level Design	11/14/2022
Divide into Memory/Logic Parts	
Deterime ideas for how to test each piece	
Part by Part Block Design	11/28/2022
Create RTL for wishbone bus	
Create Verilog (x number of pieces)	
Test Blocks	
Finalize Tests	12/5/2022
Test and Reiterate	
(Bring-Up) Create Full RTL for design	12/12/2022
Generate Netlist	
Phase 3 Implementation	**2/20/2023**
Place in harness	1/23/2023
Test & Validate Entire Chip	1/30/2023
Harden Design	2/6/2023
Pass all Prechecks	2/13/2023
Submit to Fab (Hard Deadline: Date yet to release)	2/17/2023
Phase 4 Functional Finalization	**5/1/2023**
Application Software	3/6/2023
Documentation	5/1/2023

3.2.1 Task Detail

1. Phase 1 Background Development
    1. Sample process
        1. Have design environment set up & complete user adder example
            1. Be able to use the design environment to clone the user adder example and get it to harden within the design environment
    2. Make updates and iterations to user adder example
        1. Make changes to user adder example
            1. Try and add components and/or change the functionality of components within the user adder example to gain more experience and familiarity with the design environment
        2. Layout visualization
            1. Attempt to add software that will allow us to see how the design environment is placing the components and routing the design.
                1. It could be very useful later in our project to see if our design is being placed and routed correctly
    3. Learn about spiking neural networks
        1. Determining what our design will be and what inputs and outputs we will need

1. Basically, we will attempt to come to a consensus on what sort of specific application we will replicate that uses a neural network to operate

4. Pen & paper logic
    1. Create initial design sketch and design flow chart
        1. Essentially Create High Level Diagram of circuit
        2. Identify what components need to be present in each high-level section of the design
        3. Create K-Map for each component to identify gates that will be used
        4. Draw a pen and paper design of SNN to fully sketch out all the gates that make up each component and how each component will be connected to each other

2. Phase 2 Design
    1. Determine High-level Design
        1. Divide into memory & logic parts & determine how to test each component
            1. After splitting our components into the groups mentioned above, we will have to come up with a rough draft for code in a Verilog test bench. This is to ensure that depending on the inputs given that each component outputs what they are supposed to
        2. Part by part block design
            1. Create RTL for wishbone bus & create Verilog code for "x" number of pieces
                1. Write the Verilog code to describe all the necessary components within the circuit
        3. Test Blocks
            1. Finalize and use Tests, reiterate when necessary
                1. Finish our test code to ensure it tests the desired criteria
                2. If components don't behave as designed, then we will repeat the part-by-part block design then test the components again.
                    1. Continue to do this process until all components behave as desired
            2. Create full RTL for design
                1. Generate Netlist
                    1. Essentially create a description of all the components in our circuit and all the nodes that each component is connected to

3. Phase 3 Implementation
    1. Each step in this section shouldn't require further explanation as they are clear and self-explanatory

4. Phase 4 Functional Finalization
    1. Application Software
        1. Create code that will test the functionality of our chip design in the development board that will be sent to us by efabless.
            1. This is to ensure that it completes the desired functionality for the specific SNN that we decided to create
    2. Documentation
        1. We will write up the process that we went through to create our design not only for our senior design class but also for our client

1. Our client wishes to create a co-curricular at ISU for chip fabrication but needs a well-documented and thought-out plan of action of how to do so meaning:
    1. What applications to use
    2. How to gain access to each of those applications
    3. How to use those applications
    4. Etc.

3.3 Project Proposed Milestones, Metrics, and Evaluation Criteria

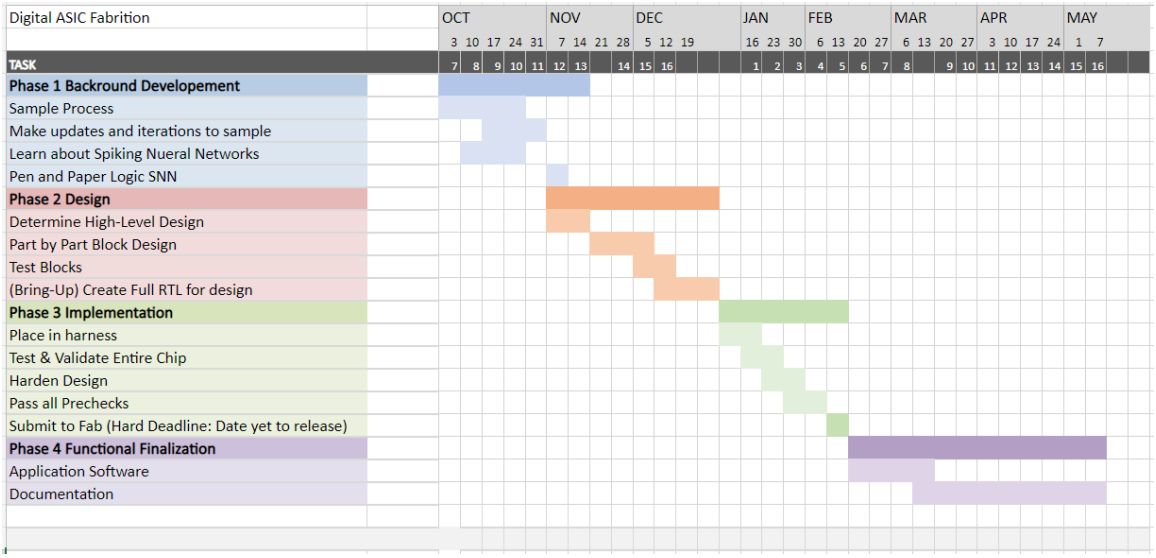
Our design should allow us to implement at least one neuron of a spiking neural network with accurate results.

7. High-level design with all components
8. Test plan for each component
9. Software interface work started
1. Ability to test sub-modules
1. Ability to test full design
1. Design fitting in bounds of harness
2. Ability to simulate full design

a. Submission to and acceptance by Efabless
b. Software interface finished and tested

3.4 Project Timeline/Schedule

The proposed project schedule is included in the figure below.



3.5 Risks And Risk Management/Mitigation

The risks for this project are relatively few as the primary goal is to gain knowledge. The secondary goal is to deliver a working product to the customer at the end of the process. That process itself, which goes through Efabless, has some inherent risks as laid out in the table below.

Risk description	*Probability*
Efabless program ends	5%
Design does not fit in user area	30%
Efabless does not select our project for program	10%

3.6 Personnel Effort Requirements

The initial estimate of the hours needed to complete this project are as proposed by the table below. These estimates are based off of a previous senior design team and experiences from previous class projects. There may be uncounted for time spent in debugging and other surprise tasks.

Task:	**Estimate of Hours Needed:**
Phase 1: Background Development	
Sample Process	8
Make Updates and Iterations to Sample	4
Learn about Spiking Neural Networks (SNN)	3
Pen and Paper Logic SNN	10
Phase 2: Design	
Determine High-Level Design	4
Part by Part Block Design	4
Test Blocks	16
(Bring Up) Create Full RTL for Design	16
Phase 3: Implementation	
Place in Harness	16
Test and Validate Entire Chip	16
Harden Design	8
Pass all Prechecks	4
Submit to Fab (Hard Deadline: Date yet to Release)	10
Phase 4: Functional Finalization	
Application Software	40
Documentation	20

3.7 Other Resource Requirements

The resources that this project will utilize are as listed.

3. Silicon wafer

4. Development board
    5. 1.8 Volt Regulator
    6. 3.3 Volt Regulator
    7. SPI Flash
    8. Serial to USB connector
- Open-source RTL & PDK
- Layout visualization software (maybe)
- Gitlab
- Github
- User adder example project
- Pen & paper

4 Design

4.1 Design Context

4.1.1 Broader Context

Spiking Neural Networks (SNN) is a third-generation network, it can be applied in many areas, such as image processing, voice recognition, and human-like computing (AI). Compared to other generation networks, SNN has advantages of higher energy efficiency, higher area efficiency, efficiency on chip learning and more fault tolerance. The disadvantages of SNN are training is difficult, the accuracy does not match the other generation networks, and programming frameworks are still in their infancy.

Our project is to make an open-source SNN in an application specific integrated circuit (ASIC), with the main point of our project being that we can provide others with our solution and document the process to help others complete their own ASIC. The communities that our design will be for will be for those looking to further the open-source design of digital chips. Specifically, those who are a part of the slack community looking to add to the amount of open-source chip designs that exist. This project will fulfill the needs of all who are looking to accelerate the growth and complexity of digital chip ideas and projects alike.

Area	**Description**	**Examples**
Public health, safety, and welfare	How does your project affect the general well-being of various stakeholder groups? These groups may be direct users or may be indirectly affected (e.g., solution is implemented in their communities) **Our project won't necessarily have a large effect on the public's health. But it does influence the public's welfare & quality of life. Because as digital chip designs grow more complex at a faster rate there will be more complex and useful applications in the electronics that are used by the public.**	Increasing/reducing exposure to pollutants and other harmful substances, increasing/reducing safety risks, increasing/reducing job opportunities **If our design is used for purposes related to health research/studies, the results of those may have a positive impact on public health, which would be indirectly caused by our design.**

Global, cultural, and social	**The open-source culture is one of openness, growth, and collaboration. Our project is part of an open-source initiative for silicon design. It will accordingly reflect the values of the community as we aim to create a foundation for future groups whether in out University or the open-source silicon community at large.**	**This project as a spiking neural network accelerator can be used as a component of an artificial intelligence system. AI comes with its own set of challenges and potential violations of community ethics.**
Environmental	What environmental impact might your project have? This can include indirect effects, such as deforestation or unsustainable practices related to materials manufacture or procurement. **The materials that are used to fabricate our chip are natural resources. The mining of those and the fabrication process itself is often unsustainable and has an environmental impact in the waste product that is disposed of.**	Increasing/decreasing energy usage from nonrenewable sources, increasing/decreasing usage/production of non-recyclable materials **Our project is to fabricate the chip, so the usage of nonrenewable sources and non-recyclable materials such as silicon, photoresist or other chemical products will increase.**
Economic	What economic impact might your project have? This can include the financial viability of your product within your team or company, cost to consumers, or broader economic effects on communities, markets, nations, and other groups. **In the long term, open-source chip design has the potential to disrupt the industry and pave the way for fast paced innovation and economic growth in that area. Cost of devices to consumers could be reduced. In the short term, a curriculum at Iowa State for chip design will create more learned future engineers and give a broader base for students to draw off and implement their unique, creative ideas.**	Product needs to remain affordable for target users, product creates or diminishes opportunities for economic advancement, high development cost creates risk for organization **Our final product, a fabricated SNN neuron chip design, will be free and publicly available for other's use. It will need to be documented well.**

4.1.2 Prior Work/Solutions

One of the literature pieces that we are using as a basis for our project is called "An Adaptive Memory Management Strategy Towards Energy Efficient Machine Inference in Event-Driven Neuromorphic Accelerators". Now we are primarily using this literature piece to get a general idea of how SNN will likely need to be laid out as well as some of the high-level component ideas that will be needed for our SNN. The document can be found below:

An Adaptive Memory Management Strategy Towards Energy Efficient Machine Inference in Event-Driven Neuromorphic Accelerators: https://par.nsf.gov/servlets/purl/10113424

Advantages:

Don't have to start designing SNN from scratch which is good since none of us have done so before

Gives an idea of some of the high-level components necessary to construct circuits

Gives an idea of general functionality of the high-level components that it does describe

Disadvantages:

The version of their SNN is more Bio based and therefore some of the components in there do not apply to our SNN

This project will be a continuation of an cocurricular development by the client. As such, a previous Iowa State senior design team performed a similar project and has provided resources to this team to continue developing a knowledge base. The work of the previous senior design team can be found as described in their website linked below.

Website: http://sddec22-17.sd.ece.iastate.edu/

This team will be following their personal advice, environment set-up documentation, and design flow. Our project will be different from theirs in functionality and structure. While their team completed the design of a bitcoin mining chip, our team will be creating a SNN accelerator.

FPGA implementation is a close relative to ASIC design. To guide our design process on SNN acceleration, our team will be gleaning information from related FPGA projects. Our project is fundamentally different as the functionality is built into the circuit before the chip is created, but much of the circuit creation is similar. To get a grasp of the inputs, outputs, and steps required in an SNN, the paper "FPGA implementation of Spiking Neural Networks" was referenced.

Rosado-Muñoz, A., Bataller-Mompeán, M., & Guerrero-Martínez, J. (2016, April 21). *FPGA implementation of spiking neural networks*. IFAC Proceedings Volumes. Retrieved October 21, 2022, from https://www.sciencedirect.com/science/article/pii/S1474667015404562?ref=pdf_download&fr=RR-2&rr=75dc73ab0b3c8114

In breaking down the design of our neural network, logic structures will need to be created. The work "Design of Various Logic Gates in Neural Networks" provided a starting place for our design.

Yellamraju, S. (2013, December 23). *Design of Various Logic Gates in Neural nNetworks*. ResearchGate. Retrieved October 21, 2022, from https://www.researchgate.net/publication/259990623_Design_of_Various_Logic_Gates_in_Neural_Networks

4.1.3 Technical Complexity

In our project design, we need to complete the hardware design, which needs the electrical engineering knowledge to finish the circuits design, and how to arrange the control unit, random access memory etc. We also need to encode on the device to make SNN able to calculate, so we will use the knowledge from computer science. Our project consists of the following engineering principles:

4. Hardware design/testing
5. Software design/testing
6. Documentation
7. Neural networks

The previous senior design group that worked in this project field has begun documentation of their process, but outside of that, the work we are doing is completely new to the ISU computer/electrical engineering curriculum. Our client is using the knowledge we gain to help develop a program for future students in digital chip fabrication, so there is currently no existing solution. The challenges in our project are how to improve accuracy by programming, how to increase complex learning capabilities.

4.2 Design Exploration

4.2.1 Design Decisions

The first decision we make is to make updates and iterations to user adder example and make changes to user adder example, by trying and adding components and change the functionality of components within the user adder example to gain more experience and familiarity with the design environment.

The second decision we make is to create an initial design sketch and design flow chart. This is important because we are laying the groundwork for creating high-level diagrams of circuits, identifying what components need to be present in each high-level section of the design, and creating K-Map for each component to identify gates that will be used.

The third decision we make is to add test block, the benefit of adding test block is it able to test our codes after finishing the test code and ensure it tests the desired criteria. If components don't behave as designed, then we will repeat the part-by-part block design then test the components again. The test block will save us a lot of time and improve fault tolerance.

4.2.2 Ideation

Each member individually investigated possible applications to use in our design. After brainstorming on our own, we compiled a list of possibilities that we could suggest to our client. The options we considered were

5. RSA encryption/decryption
6. Image reconstruction
7. Fingerprint reader accelerator
8. Digital encoder

9. Spiking neural network

To decide which application to use, we considered the requirements of our project, as well as where our own skills exist as a team. Our design requirements specify that the design must fit on the chip and should have some level of novelty. This eliminates fingerprint reading, RSA, image reconstruction, and digital encoder. With the remaining application of spiking neural network, we decided to select this since it would meet the requirements of the design and is something we believe we can implement.

4.2.3 Decision-Making and Trade-Off

Our group agreed to use spiking neural networks for the following reasons:

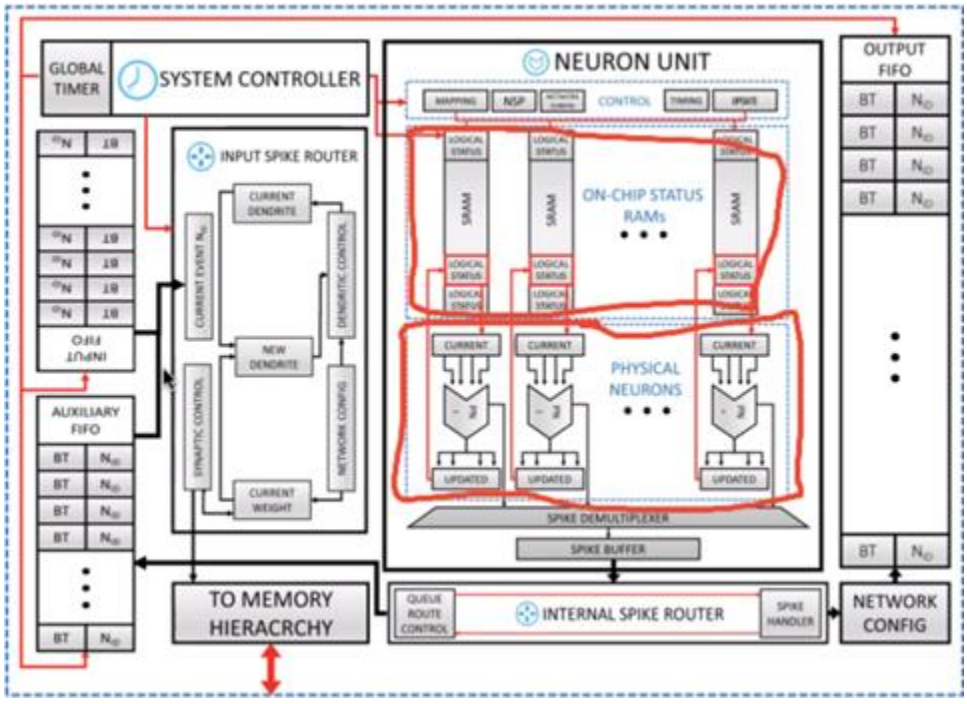
6. It is a design which we will most likely be able to fit in our user area of the chip
7. Our client has an interest in this area of technology
8. It has a level of novelty that will help in our selection for the efabless program
9. It has elements that would benefit from the acceleration of an application specific integrated circuit (ASIC)

After identifying these reasons, our group selected spiking neural networks as our application for design.

4.3 Proposed Design

4.3.1 Overview

The 3rd generation of neural networks, spiking neural networks, aims to bridge the gap between neuroscience and machine learning, using biologically realistic models of neurons to carry out computation.



4.3.2 Detailed Design and Visual(s)

SNN tries to more closely mimic a biological neural network. Therefore, instead of working with continuously changing in time values used in ANN, SNN operates with discrete events that occur at certain points of time. SNN receives a series of spikes as input and produces a series of spikes as the output (a series of spikes is usually referred to as spike trains).

The general design descriptions:

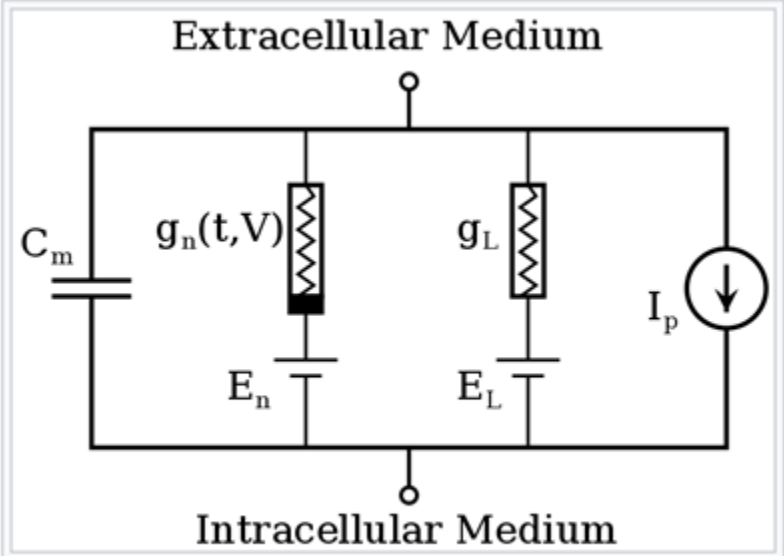
1. At every moment of time each neuron has some value that is analogous to the electrical potential of biological neurons;

2. The value in a neuron can change based on the mathematical model of a neuron, for example, if a neuron receives a spike from the upstream neuron, the value might increase or decrease;

3. If the value in a neuron exceeds some threshold, the neuron will send a single impulse to each downstream neuron connected to the initial one;

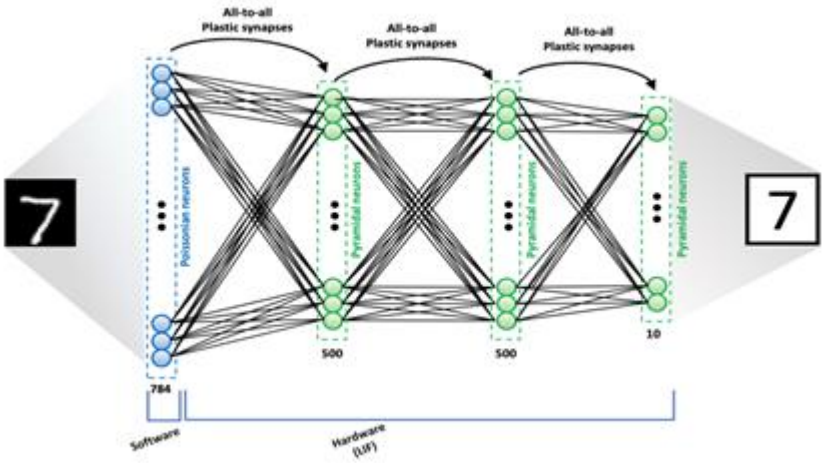
4. After this, the value of the neuron will instantly drop below its average. Thus, the neuron will experience the analog of a biological neuron's refractory period. Over time the value of the neuron will smoothly return to its average.

To build SNN, first we need SNN neuron models. SNN neurons are built on the mathematical descriptions of biological neurons. There are a lot of models that can be used. For example, Hodgkin–Huxley model, or conductance-based model, is a mathematical model that describes how action potentials in neurons are initiated and propagated. It is a set of nonlinear differential equations that approximate the electrical characteristics of excitable cells such as neurons and muscle cells. It is a continuous-time dynamical system.



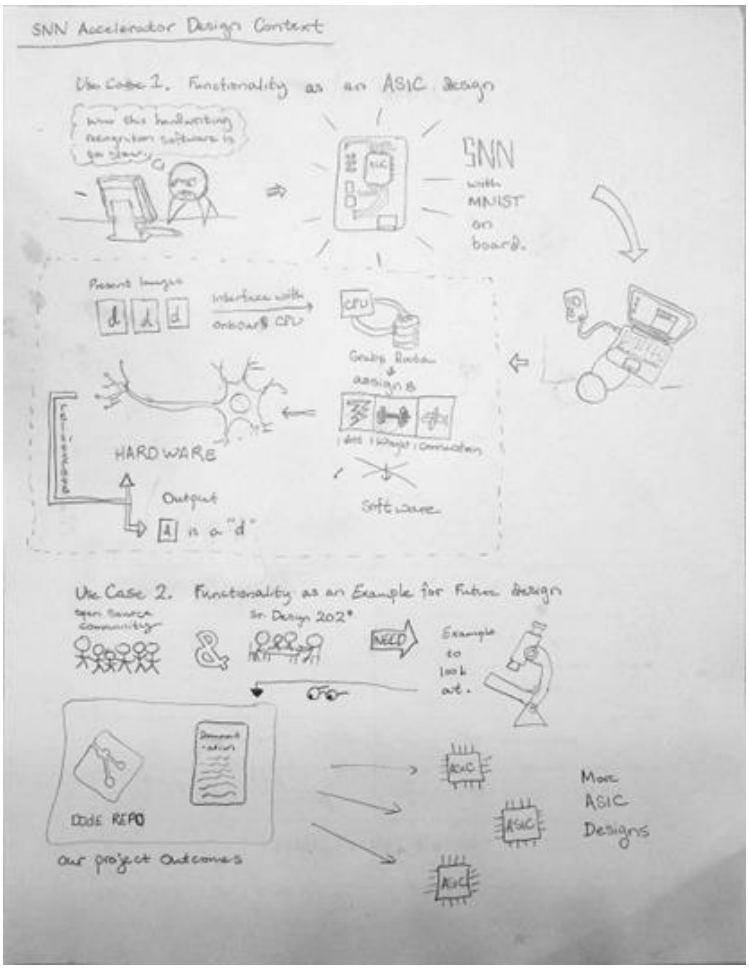
Basic components of Hodgkin–Huxley-type models which represent the biophysical characteristic of cell membranes.

Also, SNN architectures is an important part of the design, there are three types of architectures, and we use Feedforward Neural Network which is a classical NN architecture that is widely used across all industries. In such an architecture the data is transmitted strictly in one direction – from inputs to outputs, there are no cycles, and processing can take place over many hidden layers.



4.3.3 Functionality

Our design will provide customers with complete SNN design plans and drawings, as well as resources about programming. Users can get these resources for free when they need them and can use these resources to improve their own programs. For example, if a university needs to offer a course on SNN, then they can directly use our design to schedule the lab. Our design will provide each stage of SNN design including methods and processes. Our designs can save a lot of time and provide different ideas for those who need them.



4.3.4 Areas of Concern and Development

Our current design isn't the most important aspect of fulfilling our user needs. Considering that our client is looking for us to bring up a fully functional digital chip design our design really could have been any number of things. The most important part is that we get our design to function as intended. That way our client will have further proof that it is possible to implement a chip design co-curricular in Iowa state.

The first of our primary concerns for our current design is that our logic isn't sound and that unwanted outputs from our neurons may occur. Which in a simulation of 800 or more neurons can cascade and cause quite a few issues. The second is that we will struggle trying to find a way to fit as many of the general components of the SNN in the user area space. We may only be able to fit two neurons with synapses to go with them, as well as some additional SRAM as needed and if we have space left over. And lastly, we will not be able to get our design hardened within the caravel harness for the chip. Without being able to harden it we won't have a chance of getting our design fabricated.

We will be sure to go through the digital logic aspect of our design and carefully consider as many scenarios as possible. That way we can carefully design our hardware and software to operate in a

manner that will account for all those scenarios. The second one we can't truly deal with until we have completely solidified our design and tested its accuracy using the Verilog test environment. And likewise the third one cannot be dealt with or planned for until we have completed and dealt with the issues that come before hardening the design.

4.4 Technology Considerations

We have many different technologies when we build neuron models and architectures.

There are two types of neuron models we can choose. First, Conductance-based models describe how action potentials in neurons are initiated and propagated. Second, Threshold models generate an impulse at a certain threshold.

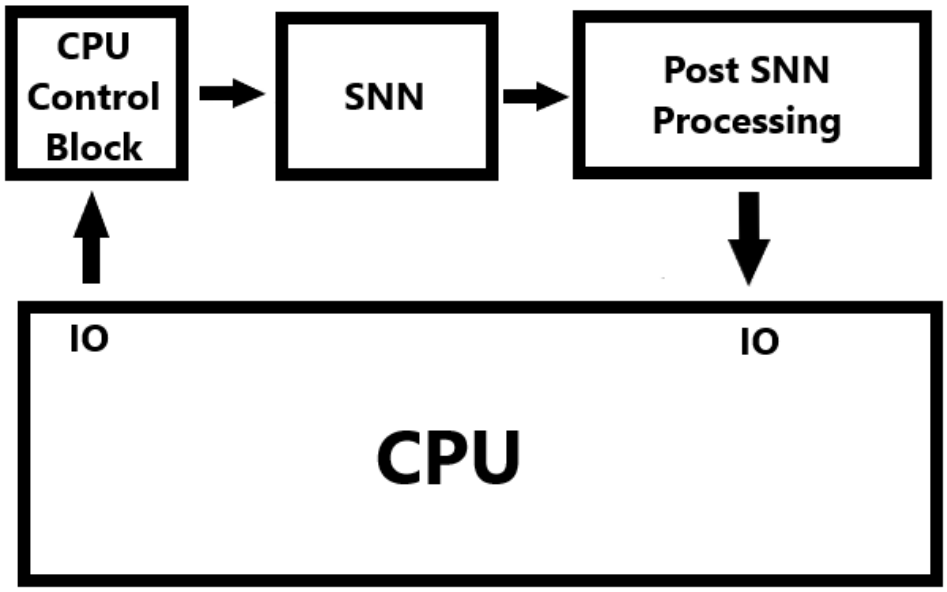
The benefits of the Conductance-based models are that models for excitable cells are developed to help understand underlying mechanisms that contribute to action potential generation, repetitive firing and bursting and so on.

In turn, as the number of currents included in conductance-based models expands, it becomes more difficult to understand and predict the resulting model dynamics due to the increasing number of differential equations.

The possible solution if we meet some model problem, we can just switch to another model, because there are many models can use, such as, Hodgkin-Huxley model, FitzHugh–Nagumo model, Morris–Lecar model, Hindmarsh–Rose model, Izhikevich model etc. And each model behaves slightly differently.

4.5 Design Analysis

We are now starting to draw logical circuits and SNN hardware design, I think our design is feasible because the technology used in each step is essential, and as each step is completed, we can fabricate the SNN chip successfully. We are working on the hardware of the SNN as expected part of the construction, our plan is to complete the hardware design first and then solve the programming problem, this is the most reasonable plan for our overall design, because the hardware design is the cornerstone of the SNN chip design, and then all the processes must be worked on this basis.



This is our proposed design and flow at a high level. The data from the CPU will be gotten from the SRAM through an I/O. It will arrive first at a control block which will 'distribute' it accordingly. We need this block to interface with the built-in features of the Caravel harness that all the MPW design submissions sit on. Additionally it will allow the data to flow more efficiently. Then the neuron/MAC hardware accelerator will be fed the data and complete the calculations according to the model we choose and is in line with the MNIST data set. Last the output of the neuron will be restructured and stored according to a predefined function block.

5 Testing

Testing is an extremely important component of most projects, whether it involves a circuit, a process, power system, or software.

The testing plan should connect the requirements and the design to the adopted test strategy and instruments. In this overarching introduction, given an overview of the testing strategy and your team's overall testing philosophy. Emphasize any unique challenges to testing for your system/design.

In the sections below, describe specific methods for testing. You may include additional types of testing, if applicable to your design. If a particular type of testing is not applicable to your project, you must justify why you are not including it.

When writing your testing planning consider a few guidelines:

Is our testing plan unique to our project? (It should be)

Are you testing related to all requirements? For requirements you're not testing (e.g., cost related requirements) can you justify their exclusion?

Is your testing plan comprehensive?

When should you be testing? (In most cases, it's early and often, not at the end of the project)

5.1 Unit Testing

The unit that we will have to test for our design is the Verilog code that we will create for our neuron model. We will be testing to make sure our design is working the way it is supposed to by constructing a test bench in Linux. It will be a combination of C code to set up the virtual hardware that was provided to us as well as Verilog code to simulate the inputs and outputs to our neuron model.

The tools that will be used will be Linux based software, it will have Icarus Verilog as well as docker set up in its environment. All of that along with a self-made test bench which, as mentioned before, will be a combination of C based code file and Verilog based code files.

5.2 Interface Testing

The digital design will include numerous interfaces, the two that will be most utilized are SPI and GPIO. The SPI pins are the interface that will allow a remote device to communicate with the ASIC. GPIO is used to allow the integrated RISC-V CPU to communicate with our fabricated area. We can test both of these interfaces using our logic analyzer. When creating our design, the logic analyzer allows us to be able to monitor the signal of specified pins. We then have the option to wire these pins to GPIO and SPI if we want to be able to monitor what each pin is currently at and use that information to debug as needed.

5.3 Integration Testing

The integration of our design will be tested as we follow the design flow. There will be three major areas for integration testing along the way, integrating the block level components to the top-level design, integrating the design into the harness, and integrating our gained knowledge into documentation.

Block to Top Testing:

To test that each block works along the way, we will analyze the output of each using simulation software. With the goal of using all open-source mediums, we will be using GTKWave for viewing waveforms and Verilator for simulating system Verilog. For each level, block, mid, and top, we will write test benches along the way to verify that our component and its parts are working as they should.

Design to Harness Integration Testing

The testing plan for checking that our design will work in the contexts of the caravel harness, will be running the provided checks that come with the environment. This is detailed in the Acceptance Testing section below, though mentioned here as it is an important part of the design integration.

Knowledge to Documentation

An important outcome of our project will be the building of a knowledge base for future projects and/or classes. We will test our documentation by peer reviewing our teammates' documents. As students we are in the primary audience demographic and aim to test our writing by considering what would make the process easier for us.

5.4 System Testing

Our system testing strategy is to test the accuracy of our Verilog code first, then test whether the overall product meets all its requirements, and finally test the product's performance and usability. Unit tests need to ensure that the Verilog codes are able to run and with high accuracy. The interfaces test needs to allow SPI and GPIO to successfully receive signals, and integration tests should be sufficient to satisfy that each block is working correctly. The tools we used to test our system is various commercial and open-source tools, such as Linux, Verilator etc.

5.5 Regression Testing

One way we will ensure new additions do not break old functionality is by performing merge checks in GitLab (tool). Every time a new change has been made, a job will be executed to verify (requirement) that all tests that have been made are still passing with the new changes before allowing changes to our main branch of source code. This will require us to write tests that extensively test all aspects of our project previously made. The quality of the regression testing will then be reliant on our existing tests. The critical features that should not break are those that will be used as the building blocks of other components. This will be made up of things like registers, arithmetic units, buses, etc. Since they will be the most common in our design, the larger functionality will be dependent on their reliability.

5.6 Acceptance Testing

The acceptability of our design will be majorly tested by the Efabless provided prechecks. These are the tests done before the submission is allowed onto the shuttle and to verify that all the components of the design are working in tandem. The way we will do these tests is by executing the scripts and commands included in the repository found by following this link. https://github.com/efabless/mpw_precheck. We plan to pass all the acceptability testing before we submit to the program

5.7 Results

Our test results prove that our product is usable and does its job correctly. Every part of our design needs to be tested and guaranteed to function properly and meet our requirements. Our design is from hardware to software, which is very useful and efficient because once we have completed the hardware design and testing, there will be a lot of room for trial and error in the code part.

6 Implementation

Our implementation plan in the following semester will consist of the following:

Continuing to implement individual components in HDL (neuron, neuron controller, etc.)

Continuing to write and conduct tests for individual components

Integrate components together under the Caravel wrapper

Write and conduct tests for the fully integrated Caravel wrapper

Run the required prechecks for our project created by the Efabless program

Submit to Efabless program

The majority of our time will likely be spent on the testing process, as our high-level design covers most of the actual component implementation. Once we begin these tests, it may be necessary for us to redesign certain aspects of our components to work how we expect them to.

7 Professional Responsibility

This discussion is with respect to the paper titled "Contextualizing Professionalism in Capstone Projects Using the IDEALS Professional Responsibility Assessment", International Journal of Engineering Education Vol. 28, No. 2, pp. 416–424, 2012

7.1 Areas of Responsibility

Table 1: McCormack's Seven Areas of Professional Responsibility

Area of Responsibility	Definition	NSPE Canon	Q1 IEEE-CS/ACM Code of Ethics	Q2: Importance of Responsibility	Q3 Current Level of Performance
Work Competence	Perform work of high-quality integrity timeliness, and professional competence	Perform services only in areas of their competence; Avoid deceptive acts	Software engineers shall ensure that their products and related modifications meet the highest professional standards possible	High	Medium
Financial Responsibility	Deliver products and services of realizable value and at reasonable costs.	Act for each employer or client as faithful agents or trustees.	Software engineers shall act in a manner that is in the best interests of their client and employer	Low	Low
Communication Honesty	Report works truthfully, without deception, and understandable to stakeholders	Issue public statements only in an objective and truthful manner; Avoid deceptive acts	Ensure adequate documentation, Cooperate in efforts to address documentation	Medium	High

Health, Safety, Well-Being,	Minimize risks to safety, health, and well-being of stakeholders	Hold paramount the safety, health, and welfare of the public.	8 Principles are intended to collectively apply	Low	Low
Property Ownership	Respect property, ideas, and information of clients and others.	Act for each employer or client as faithful agents or trustees.	Use only in authorized ways, do not violate intellectual property law	High	Medium
Sustainability	Protect environment and natural resources locally and globally.	N/A	Identify, define and address issues, including environmental issues related to work projects	Medium	High
Social Responsibility	Produce products and services that benefit society and communities	Conduct themselves honorably, responsibly, ethically, and lawfully to enhance the honor, reputation, and usefulness of the profession	Software engineers shall act consistently with the public interest Software engineers shall advance the integrity and reputation of the profession consistent with the public interest.	Medium	High

Note: Adapted from McCormack et al. (2012).

While the NSPE and IEEE code of ethics both address the seven areas of Professional Responsibility, there are differences in how they prioritize and describe these areas. For the NSPE, the descriptions are more general, because engineers as a category are general, while the IEEE features electronics-specific elements, such as the importance of good documentation. Because the IEEE statement was drafted alongside the Association for Computing Machinery, these patterns make sense.

7.2 Project Specific Professional Responsibility Areas

For this project, work competence has a high level of importance because all members of the project team share a responsibility with each other to produce the best work possible to produce high-quality products.

Communication Honesty is at medium level because as a group, our work competence is directly tied to communication already, so prioritizing work competence ensures our fair contribution.

Property ownership is of high importance because we should all know what our work is, and how responsibilities have been executed, so taking on someone else's work would be a serious problem. For both sustainability and social responsibility, the responsibility has been set at medium importance because the priorities of this project are part of the capstone requirements, and we do not have a high amount of flexibility to introduce concepts. Instead, we avoid harm to society.

Financial Responsibility and Health, Safety, Wellbeing are the lowest level of our projects because our project does not involve financial issues or require doing dangerous work. The most sources we used for this project are free.

7.3 Most Applicable Professional Responsibility Area

Most applicable professional responsibility areas are work competence and Communication Honesty.

8 Closing Material

8.1 Discussion

The main result of our project is the final product we fabricated can satisfy all the requirements and work with high accuracy and efficiency.

8.2 Conclusion

So far, we have completed the reading and understanding of the background knowledge, set up all the tools we needed, and completed the basic research and design. Our Goals is to fabricate the digital chip successfully and to achieve this goal we need to follow the schedule to complete the mission step by step.

8.3 References

1. Saha, Saunak, et al. "An Adaptive Memory Management Strategy towards Energy Efficient Machine Inference in Event-Driven Neuromorphic Accelerators." *2019 IEEE 30th International Conference on Application-Specific Systems, Architectures and Processors (ASAP)*, 2019, https://doi.org/10.1109/asap.2019.000-2.

2. "Digital Simulation of Spiking Neural Networks." *Pulsed Neural Networks*, 1998, https://doi.org/10.7551/mitpress/5704.003.0014.

3. "IEEE Standard VITAL Application-Specific Integrated Circuit (ASIC) Modeling Specification," in *IEEE Std 1076.4-1995* , vol., no., pp.1-96, 17 May 1996, doi: 10.1109/IEEESTD.1996.80811.

4. "IEEE Standard Testability Method for Embedded Core-based Integrated Circuits," in IEEE Std 1500-2022 (Revision of IEEE Std 1500-2005) , vol., no., pp.1-168, 12 Oct. 2022, doi: 10.1109/IEEESTD.2022.9916221.

5. "IEEE Standard for Design and Verification of Low-Power, Energy-Aware Electronic Systems," in IEEE Std 1801-2018 , vol., no., pp.1-548, 29 March 2019, doi: 10.1109/IEEESTD.2019.8686430.

8.4.1 Team Contract

Team Members:

1) Katherine Gisi 2) William Zogg

3) Tyler Green 4) Aaron Sledge

5) Fulai Zhu

Team Procedures

1. Day, time, and location (face-to-face or virtual) for regular team meetings:
    a. Monday, 3:00 -3:30 in Durham conference room.
    b. Monday, 10-11 or Wednesday 2:00-3:00. Face to face in the TLA as needed
2. Preferred method of communication updates, reminders, issues, and scheduling (e.g., e-mail, phone, app, face-to-face):
    a. Microsoft Teams
    b. App on phone
3. Decision-making policy (e.g., consensus, majority vote):
    a. Majority vote
4. Procedures for record keeping (i.e., who will keep meeting minutes, how will minutes be shared/archived):
    a. Keep track of what we need to talk about during meetings.
        i. Minutes- Each take notes at discretion and share in teams files

Participation Expectations

1. Expected individual attendance, punctuality, and participation at all team meetings:
    a. Participation and punctuality is expected at all team functions unless there is an actual conflict in which case team members will be notified. If it is a meeting with Duwe, he will be included in the notification.
2. Expected level of responsibility for fulfilling team assignments, timelines, and deadlines:
    a. You are expected to complete your task or let others know. Preferably as soon as possible.
    b. If a deadline needs to be extended, be in communication about it.
    c. Have levels of deadlines (soft, hard), but attempt to fulfill the set timeline as much as possible.
3. Expected level of communication with other team members:
    a. At least once a day check teams
    b. During more intense project times, have notifications enabled for the SD workspace.
4. Expected level of commitment to team decisions and tasks:
    a. Senior design is not a back burner class.
    b. If you commit to doing something, you are responsible for doing said task.

Leadership

Leadership roles for each team member (e.g., team organization, client interaction, individual component design, testing, etc.):
Roles will be determined as the project takes shape.

The goal in assigning roles is to distribute the work and position members according to their strengths.
Strategies for supporting and guiding the work of all team members:
Frequent updates on what team members have worked on and will work on.
Notifications on who is/ what task is holding back progress.
Explanations on what each member has learned as is relevant.
Helpful tips and acceptance of advice in the case of knowledge imbalance.
Strategies for recognizing the contributions of all team members:
Log of major and minor tasks.
Tell each other good job periodically.

Collaboration and Inclusion

1. Describe the skills, expertise, and unique perspectives each team member brings to the team.
    a. Tyler Green: Low level programming, effective communication, problem-solving, CI/CD and repository upkeep, CPU design, linux
    b. William Zogg: C, Verilog, VHDL, knowledge of communication interfaces (AXI / SPI), Graphics HW design, CPU HW design, PCB design, Analog circuit design, experience in Embedded programming / interfacing with digital ICs, Soldering, Data Science / Jupyter Notebook, debugging skills (ability to see ahead of time what could be causes of headache later)
    c. Katherine Gisi: 2 Internships in ASIC design, Enthusiasm, some familiarity with open source chip design, task oriented.
    d. Fulai Zhu: Circuit drawing, Experience in Embedded system, programming.
    e. Aaron Sledge: The expertise that I bring to the project in terms of technical ability is my knowledge of circuit components as well as my understanding of circuitry. I also have a fair amount of experience with embedded coding along with some experience with Verilog. As well as the skills I will acquire with PCB design through my EE 333 class fall of 2022. My non-technical skills that I bring to the project is my ability to communicate and work well with others regardless of who they are. As well as my intuitiveness, so when I don't know something, I am able to teach myself about it in a quick and efficient manner so as to not become a burden on my teammates. Lastly, I am a diligent and hard worker, so my work ethic is another thing I bring to the table. My teams don't have to worry about me pulling my weight because I will always be sure that I do so.

2. Strategies for encouraging and supporting contributions and ideas from all team members:
    a. Make a point for asking for an opinion.

3. Procedures for identifying and resolving collaboration or inclusion issues (e.g., how will a team member inform the team that the team environment is obstructing their opportunity or ability to contribute?)
    a. Share with the individual of the team if they are the root of the issue

Goal-Setting, Planning, and Execution

1. Team goals for this semester:

    a. Communicate well
    b. Steady progress throughout the semester when possible
2. Strategies for planning and assigning individual and team work:
    a. Log of tasks (major and minor as mentioned above)
    b. Use a specific task keeping application as needed
3. Strategies for keeping on task:
    a. Deadlines as described above to keep a consistent calendar

Consequences for Not Adhering to Team Contract

1. How will you handle infractions of any of the obligations of this team contract?
    a. Hear the person out to see what external factors may have contributed to said infraction.
    b. Collectively plan a solution to get back on track.
2. What will your team do if the infractions continue?
    a. Contact team mentor (Duwe).

****

a) *I participated in formulating the standards, roles, and procedures as stated in this contract.*

b) *I understand that I am obligated to abide by these terms and conditions.*

c) *I understand that if I do not abide by these terms and conditions, I will suffer the consequences as stated in this contract.*

1) William Zogg_____ DATE 9/14/22_____

2) Tyler Green_____DATE 9/14/22_____

3) Aaron Sledge_____ DATE 9/14/22_____

4) Katherine Gisi_____ DATE 9/14/22_____

5) Fulai Zhu_____ DATE 9/14/22_____