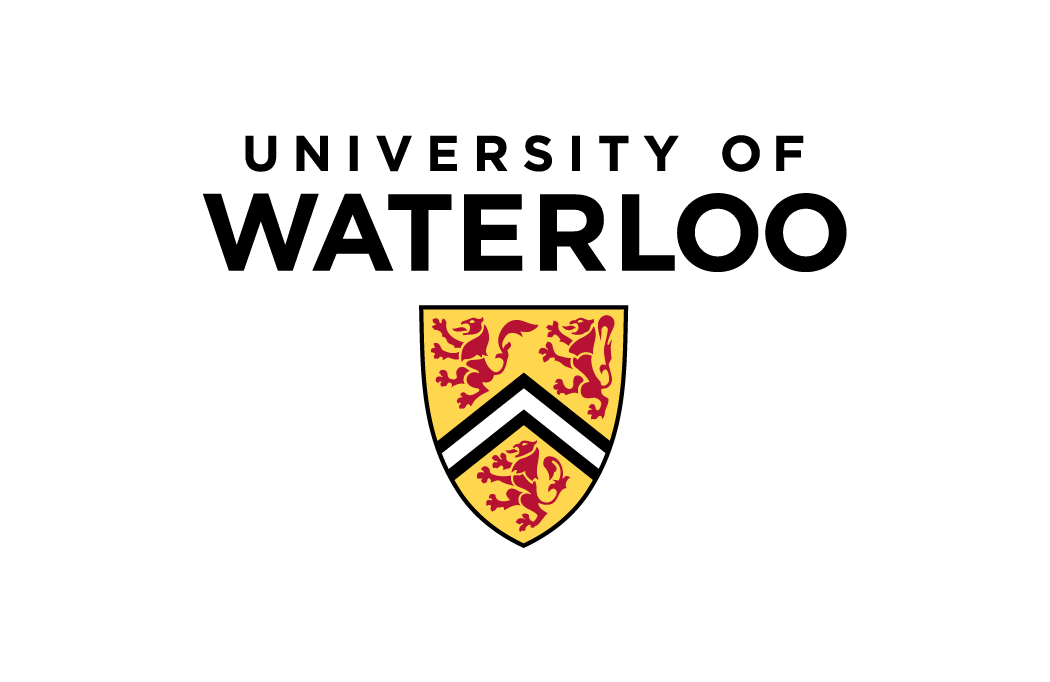
****

Faculty of Engineering

Department of Electrical and Computer Engineering

**Machine Learning on Verilog**

Group 2025.013

Prepared by:

**Kevin Kim, 20900507**

**Zefei Ou, 20909775**

**Fangrui Zhang, 20877763**

**Xilin Bai, 20842734**

Consultant: **Rodolfo Pellizzoni**

Submitted: July 26, 2024

**Table of Contents**

[**1. Overview of Project 3**](#_b0zhlk63b7bz)

[1.1 Revised Project Abstract 3](#_toefeqel2xo3)

[1.2 Original Project Timeline 3](#_w5y410ug0kab)

[**2. Current Status of Project 4**](#_o46bsrl9591j)

[2.1 Prototype Completion 4](#_rvhdjd7g7aw5)

[2.2 Student Hours 4](#_tc2r2sccirc)

[**3. Discussions 5**](#_88lcqrlylmrk)

[Changes Moving Forward 5](#_6wpll42twcml)

[Level of Confidence 5](#_p7yzi4jr4g1g)

[Level of Challenge 6](#_66qgyv9vxq6s)

[**Appendix A: Student Logs 7**](#_ontdz0m0am4a)

[**Appendix B: Initial Prototype Demonstration Feedback Sheet 17**](#_qosireen6rrc)

# 

# **1. Overview of Project**

## 1.1 Revised Project Abstract

With the rapid emergence of new designs and techniques in AI models, a hardware-aware approach is crucial when designing AI architectures to maximize performance metrics including power efficiency, latency, and throughput. The objective of this project is to provide a framework for machine learning models to be designed, deployed, and adjusted on hardware to efficiently validate model performance. This approach, though not as mature and refined as development cycles that target GPUs and TPUs, opens up possibilities for new designs and architecture that will truly exploit the potential of hardware platforms including FPGA and ASIC. With model-circuit mapping being the crucial part in the project, the framework is built upon existing compiler technologies, foremost higher-level synthesis. It also leverages open-source drivers and industry-standard serial protocols in order to abstract away the overhead of hardware knowledge and offer a plug-and-play experience for users to streamline the development process. By first utilizing FPGA platforms, this solution serves advantages over major alternatives by reducing cost, power consumption, and enabling flexible hardware-mapping without the architectural constraints posed by existing hardware technologies.

## 1.2 Original Project Timeline

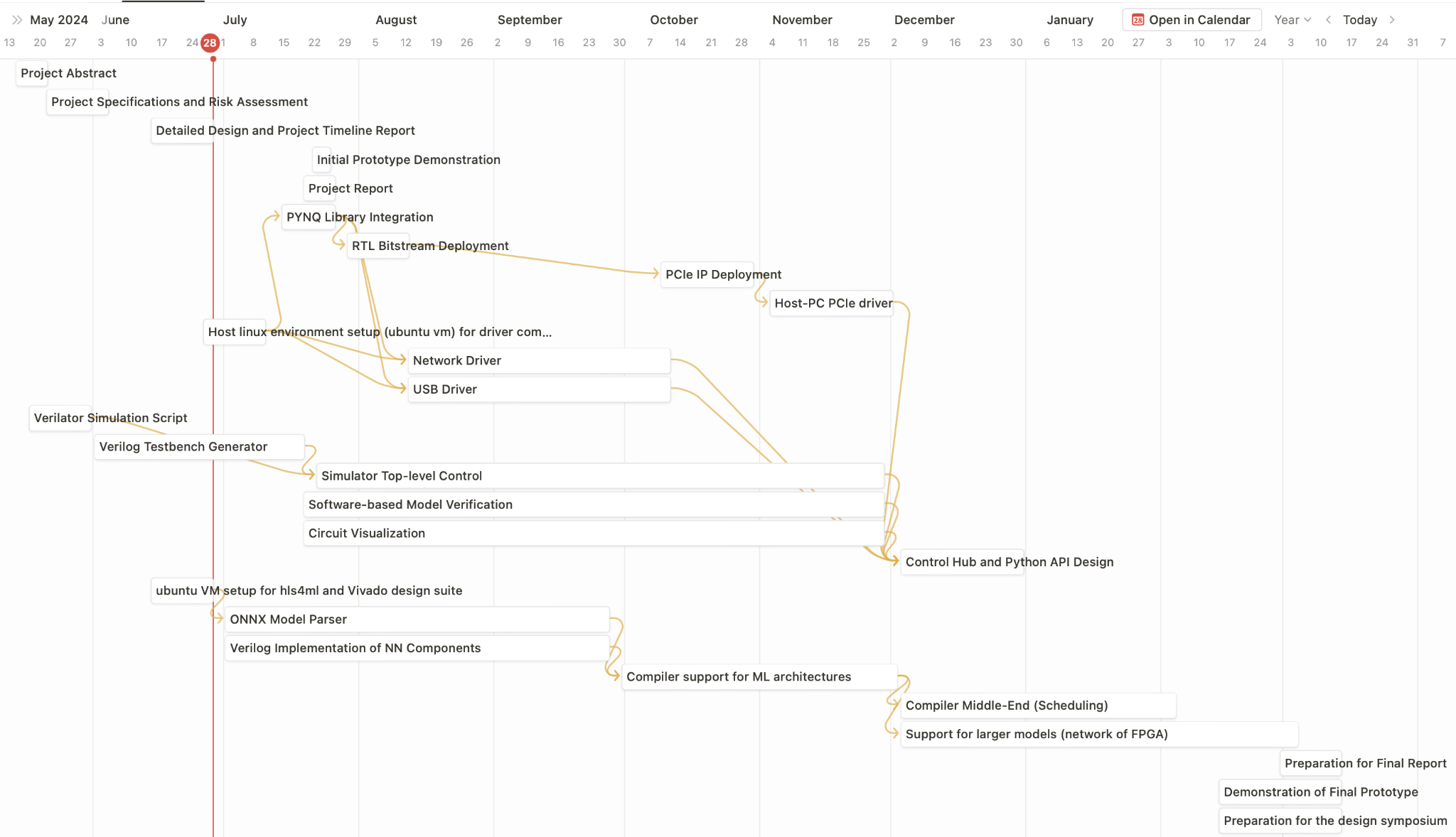


Figure 1. Project Timeline in Notion

# **2. Current Status of Project**

## 2.1 Prototype Completion

Our estimate of how complete our prototype construction is, is 76%. This is in alignment with our consultant’s estimate, as specified in Appendix B. In terms of the amount of work in the final version of the framework, we estimated that there is 20% in simulator, 40% in driver, and 40% in compiler. The progresses in each of these submodules are 90%, 70%, and 75%, respectively, which in weighted sum, add up to 76%.

The compiler relies on an open-source framework for training, compiling, and inference of deep neural networks on FPGAs. The FINN framework is well-structured and documented. Since the neural network we prepared for the demo has a typical structure and is fairly small, we are able to apply a predefined sequence of transformations provided by FINN on our model and it successfully generates a functional bitstream for us. The remaining work for the compiler is to understand how to construct a custom sequence of transformations. As FINN provides a comprehensive framework and all sets of transformations, all we need is more research into the implementation of FINN and compiler transformations targeting neural network models. Thus, we estimate the compiler is 75% done.

The current implementation of the simulator is able to parse the compiler-generated Verilog source files and creates a testbench according to the hardware interface. The only remaining task is to implement a program to feed the testbench with input samples provided by users and collect output for validation. Hence we estimate that the simulator is about 90% done.

The complexity in the driver primarily comes from the control flow of the program's status. For our prototype, we successfully demonstrated an ethernet-based driver that handles both pc-side logic and board-side deployment. The remaining work is to extend this pattern to the other protocols, namely USB and PCIe. There is minimal effort in USB as there are existing software solutions to abstract away the system-level details, but PCIe support does require extra hardware IP to be integrated into the deployed bitstream and a system-level driver to manipulate the registers. With all protocols combined, the driver submodule is roughly 70%.

## 2.2 Student Hours

Each student has put in over the expected 120 hours towards this project with small variation across each group member. Kevin Kim has put in a total of 125 hours, Fangrui Zhang has put in a total of 133 hours, Zefei Ou has put in a total of 126 hours, and Xilin Bai has put in a total of 135.5 hours. The total number of hours the group has put in is 519.5 hours, which is consistent with our student logs (Appendix A) and consultant’s impression (Appendix B).

# **3. Discussions**

### Changes Moving Forward

Both our consultant and group members agreed that our prototype was greater than 75% complete. However, each of our group members still aim to continue development during the fall term to explore further design improvements while located in different countries. We agreed that this entails some of the members to purchase their own SoC/FPGA development board on top of the allocated project budget (viable boards have already been researched). In addition, occasional remote meetings and discussions with our consultant will continue to occur for us to stay ahead of the schedule.

During our initial prototype demonstration, our consultant recommended we reduce our dependence on the compiler and possibly write code to perform our own model optimizations at the high-level (PyTorch). This would allow us to get a better idea on how we can split a given model architecture across hardware, instead of depending on the compiler’s optimized output, which is less of our control (as well as consideration for compiler errors).

Given the open-source compiler called FINN, which we found from Xilinx that met our needs for the initial prototype, a slight shift in prioritization in the project from compiler design to load distribution on a “network” of FPGA has been incurred. This is because model compilation into HLS/RTL was a necessary functional component in our design before proceeding with any implementations for performance enhancement. Additionally, a shift in focus to include support for PyTorch models has been incurred. Nonetheless, the project specifications specified in the previous report are still accurate representations moving forward.

### Level of Confidence

Our group is extremely confident that the prototype construction will be 100% complete by the March symposium, meeting all essential design specifications. Throughout this term, we have demonstrated strong teamwork, collective motivations, and individual strengths in domains across compilers, software engineering, embedded systems, and computer architecture. Each group member has a strong network of peers with strong engineering backgrounds whom we can seek technical help from. In addition, we have a project roadmap laid out with which we frequently discuss and complement with lots of research conducted online. Finally, after having several meetings with our consultant, we believe that his professional insight has been very helpful in staying on track and narrowing down on design decisions.

### Level of Challenge

During our demonstration, our consultant also stressed the importance of getting a crystal-clear idea on the system control flow, inputs, and outputs (masters and slaves). Only then, would we be able to profile our system effectively, analyzing hardware utilization and (communication) network congestion, which we foresee as the determining factor in achieving advantageous performance over existing solutions. Given that our project intersects many fields across machine learning, compilers, embedded software, and digital hardware, our consultant indicated that the project was significantly challenging.

We aim to maintain this level of challenge throughout ECE498B by furnishing implementations and making expansive improvements to our solution. To note some, scaling the hardware platform to multiple, interconnected FPGA boards necessitates workload distribution and congestion control on top of a network layout of compute nodes—all of which are topics covered in upper-level engineering. Additionally, with a data-streaming model between the host and target, optimal scheduling and pipelining techniques are necessary—again, upper-year knowledge of embedded systems and computer architecture. The front-end layer and interface for PyTorch and ONNX models imposes additional challenges in understanding upper-year engineering topics not limited to neural networks, quantization, and ML compiler flows.

# 

# 

# **Appendix A:** Student Logs

| **ECE498A: Student Log** | | | | | |
| --- | --- | --- | --- | --- | --- |
| **Name: Fangrui Zhang Group: 2025.013 Signature: FRZhang .**    **By signing above, I am stating that this is an accurate account of the tasks, dates, and times that I worked on my capstone design project.** | | | | | |
| **Task** | **Date** | **Start time** | **Finish time** | **Hours** | **Running total of hours** |
| Group meeting, project planning, task split-up, decide on consultant | 2024.05.15 | 12pm | 3pm | 3 | 3 |
| Group meeting, share progress, discuss findings, discuss simulator options | 2024.05.22 | 12pm | 4pm | 4 | 7 |
| Group meeting, share progress, create outline on project specifications | 2024.05.29 | 12pm | 3pm | 3 | 10 |
| Group meeting, share progress, decision on simulator | 2024.06.05 | 12pm | 3pm | 3 | 13 |
| Consultant and group discussion | 2024.06.06 | 6pm | 7pm | 1 | 14 |
| Group meeting, share progress, discuss formal project roadmap and timeline | 2024.06.12 | 12pm | 3pm | 3 | 17 |
| Simulation First Draft, Verilog modules, hardcoded testbenches | 2024.06.15 | 11am | 7pm | 8 | 25 |
| Simulation Scripts for automating build procedure | 2024.06.16 | 11am | 7pm | 8 | 33 |
| Group meeting, share progress, split sections of detailed design report | 2024.06.19 | 12pm | 3pm | 3 | 36 |
| Consultant and group discussion, feedback on simulator | 2024.06.20 | 6pm | 7pm | 1 | 37 |
| Learning FINN, MLP, I/O ports, resources needed for FYDP | 2024.06.25 | 1pm | 6pm | 5 | 42 |
| Group meeting, share progress updates | 2024.06.26 | 12pm | 3pm | 3 | 45 |
| Debugging Verilator Environment, resolved version issue | 2024.06.27 | 9am | 12pm | 15 | 60 |
| Group meeting, share progress on simulation environment | 2024.07.03 | 12pm | 3pm | 3 | 63 |
| Debugging dependencies errors for simulation, gtkwave debugging | 2024.07.04 | 11am | 1am | 14 | 77 |
| Consultant and group discussion | 2024.07.04 | 6pm | 7pm | 1 | 78 |
| Group meeting, shared progress on simulation | 2024.07.10 | 12pm | 3pm | 3 | 81 |
| Researching commands on Verilator, got the project building with hard-coded test bench | 2024.07.11 | 1pm | 12pm | 11 | 92 |
| Group meeting, shared progress | 2024.07.17 | 12pm | 3pm | 3 | 95 |
| Consultant and group discussion | 2024.07.18 | 6pm | 7pm | 1 | 96 |
| FINN research into axis interface, debugging output signal not visible error | 2024.07.20 | 2pm | 8pm | 6 | 102 |
| Improving hard-coded Verilog test bench, continued testing with FINN I/O ports and signals | 2024.07.23 | 3pm | 11pm | 8 | 110 |
| Group meeting, shared progress | 2024.07.24 | 12pm | 3pm | 3 | 113 |
| Test bench generator implemented, wrote script for automating whole process | 2024.07.24 | 3pm | 12am | 9 | 122 |
| Researching I/O ports, Verilog code extraction code to integrate simulation with compiler for demo | 2024.07.25 | 2pm | 12am | 10 | 132 |
| Prototype demonstration with consultant | 2024.07.26 | 3pm | 4pm | 1 | 133 |

Table 1. Fangrui Zhang - Time Log



| **ECE498A: Student Log** | | | | | |
| --- | --- | --- | --- | --- | --- |
| **Name: Kevin Kim \_ Group: 2025.013 Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**    **By signing above, I am stating that this is an accurate account of the tasks, dates, and times that I worked on my capstone design project.** | | | | | |
| **Task** | **Date** | **Start time** | **Finish time** | **Hours** | **Running total of hours** |
| Group meeting, project planning, task split-up, decide on consultant | 2024.05.15 | 12pm | 3pm | 3 | 3 |
| Abstract write-up, research on artificial intelligence statistics and similar, existing solutions | 2024.05.16 | 1:30pm | 10pm | 8.5 | 11.5 |
| Group meeting, share progress, discuss findings, discuss hardware platform options | 2024.05.22 | 12pm | 4pm | 4 | 15.5 |
| Research cloud compute platform | 2024.05.22 | 9pm | 11pm | 2 | 17.5 |
| Group meeting, share progress, create outline on project specifications | 2024.05.29 | 12pm | 3pm | 3 | 20.5 |
| Research available and optimal serial communication protocols for data transfer | 2024.06.02 | 3pm | 9pm | 6 | 26.5 |
| Project specifications write-up for drivers and programmable compute nodes | 2024.06.03 | 11am | 8pm | 9 | 35.5 |
| Group meeting, share progress, discuss standalone programmable compute nodes versus system-on-chip development boards | 2024.06.05 | 12pm | 3pm | 3 | 38.5 |
| Consultant and group discussion, ask about available development boards | 2024.06.06 | 6pm | 7pm | 1 | 39.5 |
| Group meeting, share progress, discuss formal project roadmap and timeline | 2024.06.12 | 12pm | 3pm | 3 | 42.5 |
| Group meeting, share progress, split sections of detailed design report | 2024.06.19 | 12pm | 3pm | 3 | 45.5 |
| Consultant and group discussion: open-source compilers for circuit generation | 2024.06.20 | 6pm | 7pm | 1 | 46.5 |
| Group meeting, share progress updates | 2024.06.26 | 12pm | 3pm | 3 | 49.5 |
| Research Xilinx design environment, toolchains, and frameworks | 2024.06.26 | 7pm | 12am | 5 | 54.5 |
| Research of on-board resources, dataflow feasibility, protocol specifications | 2024.06.27 | 1pm | 12am | 11 | 65.5 |
| Design report write-up | 2024.06.28 | 12pm | 11pm | 10 | 75.5 |
| Group meeting, share progress, discuss opens-source compiler testing and end-to-end flow | 2024.07.03 | 12pm | 3pm | 3 | 78.5 |
| Consultant and group discussion: input model size, memory constraints, software packages for communication interfaces | 2024.07.04 | 6pm | 7pm | 1 | 79.5 |
| Group meeting, share progress, discuss network of compute nodes as project solution | 2024.07.10 | 12pm | 3pm | 3 | 82.5 |
| Boot and test connections of development boards | 2024.07.11 | 10am | 6pm | 8 | 90.5 |
| Group meeting, share progress, brainstorm what to bring-up for initial prototype | 2024.07.17 | 12pm | 3pm | 3 | 93.5 |
| Inform consultant about prototype demo expectations/logistics | 2024.07.18 | 6pm | 7pm | 1 | 94.5 |
| Software Toolchain, Virtual Machine, Environment Setup | 2024.07.23 | 11am | 5pm | 6 | 100.5 |
| Group meeting, share progress update on prototype bring-up | 2024.07.24 | 12pm | 3pm | 3 | 103.5 |
| Correct environment compatibility and versioning errors between software and hardware stack | 2024.07.25 | 6am | 11am | 5 | 108.5 |
| Build custom operating system image | 2024.07.25 | 1pm | 11:30pm | 10.5 | 119 |
| Progress report write-up for initial prototype status | 2024.07.26 | 1am | 6am | 5 | 124 |
| Prototype demonstration with consultant | 2024.07.26 | 3pm | 4pm | 1 | 125 |

Table 2. Kevin Kim - Time Log

| **ECE498A: Student Log** | | | | | |
| --- | --- | --- | --- | --- | --- |
| **Name: Zefei Ou Group: 2025.013 Signature: Z.O. \_ .**    **By signing above, I am stating that this is an accurate account of the tasks, dates, and times that I worked on my capstone design project.** | | | | | |
| **Task** | **Date** | **Start time** | **Finish time** | **Hours** | **Running total of hours** |
| Research on initial ideas and designs | 2024.05.06 | 1pm | 2pm | 2 | 2 |
| Research on project feasibility | 2024.05.08 | 3pm | 5pm | 2 | 4 |
| Group meeting, share progress, discussion on initial planning of the project | 2024.05.15 | 12pm | 3pm | 3 | 7 |
| Case study about the available simulator solutions on the market | 2024.05.17 | 11pm | 2pm | 3 | 10 |
| Investigation on the usage of Verilator as the primary simulator | 2024.05.19 | 2pm | 6pm | 4 | 14 |
| Group meeting, share progress, final choice of simulator | 2024.05.22 | 12pm | 4pm | 4 | 18 |
| Create sample rtl circuit to be use with Verilator | 2024.05.24 | 7pm | 11pm | 4 | 22 |
| Worked on Verilator testbench | 2024.05.26 | 10am | 2pm | 4 | 26 |
| Group meeting, share progress, reports on Verilator integration, discussions on project specification | 2024.05.29 | 12pm | 3pm | 3 | 29 |
| Drawing project block diagram | 2024.05.31 | 6pm | 8pm | 2 | 31 |
| Writing project specification | 2024.06.01 | 4pm | 8pm | 4 | 35 |
| Developed python script to combine testbench and Verilog source files | 2024.06.03 | 7pm | 11pm | 4 | 39 |
| Group meeting, share progress, post-project-spec discussions | 2024.06.05 | 12pm | 3pm | 3 | 42 |
| Consultant and group discussion | 2024.06.06 | 6pm | 7pm | 1 | 43 |
| Determining interface for interface of ML model at Verilog level | 2024.06.08 | 11pm | 2am | 3 | 46 |
| Started implementing fully parallel matrix multiplication unit | 2024.06.10 | 10am | 2pm | 4 | 50 |
| Group meeting, shared updates on matrix multiplication module | 2024.06.12 | 12pm | 3pm | 3 | 53 |
| Wrapped up fully parallel matrix multiplication unit | 2024.06.14 | 10am | 2pm | 4 | 57 |
| Started implementing systolic array based matrix multiplication unit | 2024.06.16 | 10am | 2pm | 4 | 61 |
| Group meeting, shared updates on systolic array | 2024.06.19 | 12pm | 3pm | 3 | 64 |
| Consultant and group discussion | 2024.06.20 | 6pm | 7pm | 1 | 65 |
| Debugging systolic array module | 2024.06.22 | 10am | 2pm | 3 | 68 |
| Wrapped up implementation of systolic array module | 2024.06.24 | 10am | 1pm | 3 | 71 |
| Group meeting, shared updates on simulating custom matrix multiplication unit | 2024.06.26 | 12pm | 3pm | 3 | 74 |
| Developed a small multilayer perceptron using custom matrix multiplication unit | 2024.06.28 | 10am | 3pm | 5 | 79 |
| Group meeting, shared updates on custom Verilog ML model | 2024.07.03 | 12pm | 3pm | 3 | 82 |
| Consultant and group discussion | 2024.07.04 | 6pm | 7pm | 1 |  |
| Debugged and simulated the model on Verilator with a new testbench | 2024.07.06 | 10am | 3pm | 5 | 87 |
| Case study on options for implementation of ethernet-based driver | 2024.07.08 | 10am | 1pm | 3 | 90 |
| Group meeting, shared updates on ethernet-based driver | 2024.07.10 | 12pm | 3pm | 3 | 93 |
| Research on usage of PYNQ library on board-side driver | 2024.07.12 | 10am | 2pm | 4 | 97 |
| Implemented socket-based solution for ethernet driver | 2024.07.14 | 10am | 2pm | 4 | 101 |
| Group meeting, shared updates on status of driver | 2024.07.17 | 12pm | 3pm | 3 | 104 |
| Consultant and group discussion | 2024.07.18 | 6pm | 7pm | 1 | 105 |
| Experimented with socket-based driver on board and made attempts to resolve network issues | 2024.07.20 | 10am | 3pm | 5 | 110 |
| Implemented custom overlay class for deploying bitstreams and for supporting DMA IO with FPGA | 2024.07.22 | 10am | 4pm | 6 | 116 |
| Group meeting | 2024.07.24 | 12pm | 3pm | 3 | 119 |
| Implemented SSH-based ethernet driver to send deployment files and sample data to the dev board | 2024.07.25 | 10am | 4pm | 6 | 125 |
| Prototype demonstration with consultant | 2024.07.26 | 3pm | 4pm | 1 | 126 |

Table 3. Zefei Ou - Time Log

| **ECE498A: Student Log** | | | | | |
| --- | --- | --- | --- | --- | --- |
| **Name: Xilin Bai Group: 2025.013 Signature: X.B. .**    **By signing above, I am stating that this is an accurate account of the tasks, dates, and times that I worked on my capstone design project.** | | | | | |
| **Task** | **Date** | **Start time** | **Finish time** | **Hours** | **Running total of hours** |
| Brief on Project Overview | 2024.05.06 | 2pm | 4pm | 2 | 2 |
| Paperwork and communication to join FYDP as CS student | 2024.05.08 | 10am | 1pm | 3 | 5 |
| Group meeting, project planning, task split-up, decide on consultant | 2024.05.15 | 12pm | 3pm | 3 | 8 |
| Research compilers that target Verilog | 2024.05.17 | 4pm | 8pm | 4 | 12 |
| Group meeting, share progress, discuss possible compiler options | 2024.05.22 | 12pm | 4pm | 4 | 16 |
| Research Xilinx toolchain | 2024.05.27 | 12pm | 3pm | 3 | 19 |
| Group meeting, share progress, create outline on project specifications | 2024.05.29 | 12pm | 3pm | 3 | 22 |
| Learn High-Level Synthesis used by compilers | 2024.06.01 | 9am | 10:30am | 1.5 | 23.5 |
| Learn basics of Neural Network (NN) | 2024.06.02 | 9:30am | 11:30am | 2 | 25.5 |
| Learn NN in matrix form | 2024.06.02 | 12:30pm | 3pm | 2.5 | 28 |
| Group meeting, share progress, discuss possible on-board implementation of NN | 2024.06.05 | 12pm | 3pm | 3 | 31 |
| Consultant and group discussion, discuss functionalities of the compiler | 2024.06.06 | 6pm | 7pm | 1 | 32 |
| Research matrix multiplication (MatMul) methods in hardware | 2024.06.07 | 3pm | 7pm | 4 | 36 |
| Implementation of brute-force MatMul in Verilog | 2024.06.08 | 10am | 12pm | 2 | 38 |
| Implementation of Strassen MatMul in C++ | 2024.06.08 | 1pm | 2pm | 1 | 39 |
| Attempting to synthesize Strassen implementation | 2024.06.08 | 2pm | 5pm | 3 | 42 |
| Learn systolic array MatMul method | 2024.06.11 | 1pm | 5pm | 4 | 46 |
| Group meeting, share progress, discuss designs of compiler | 2024.06.12 | 12pm | 3pm | 3 | 49 |
| Learn HLS4ML Verilog generation method | 2024.06.14 | 12pm | 2pm | 2 | 51 |
| Install Xilinx Vivado 2024.1 | 2024.06.14 | 3pm | 8pm | 5 | 56 |
| Resolve environment errors for HLS4ML | 2024.06.14 | 8pm | 10pm | 2 | 58 |
| Setup Ubuntu 18.04 Virtual Machine | 2024.06.15 | 10am | 12pm | 2 | 60 |
| Reverting Vivado version to 2018.1 to satisfy HLS4ML requirement | 2024.06.15 | 2pm | 5pm | 3 | 63 |
| Modifying Ubuntu system file to satisfy Vivado 2018.1 requirement | 2024.06.15 | 7pm | 10pm | 3 | 66 |
| Setup Docker in virtual machine for HLS4ML environment | 2024.06.15 | 10pm | 12pm | 2 | 68 |
| Run example NN using HLS4ML | 2024.06.17 | 1pm | 4pm | 3 | 71 |
| Investigate hardware resource violation in result | 2024.06.17 | 7pm | 8pm | 1 | 72 |
| Group meeting, share progress on HLS4ML, split sections of detailed design report | 2024.06.19 | 12pm | 3pm | 3 | 75 |
| Consultant and group discussion, discuss other open-source compilers | 2024.06.20 | 6pm | 7pm | 1 | 76 |
| Install MLIR and CIRCT for Verilog generation | 2024.06.23 | 1pm | 5pm | 4 | 80 |
| Investigate build errors and support of CIRCT | 2024.06.23 | 8pm | 10pm | 6 | 82 |
| Group meeting, share progress on CIRCT | 2024.06.26 | 12pm | 3pm | 3 | 85 |
| Design report write-up | 2024.06.28 | 10am | 22pm | 12 | 97 |
| Group meeting, discuss FINN compiler feasibility | 2024.07.03 | 12pm | 3pm | 3 | 100 |
| Inform consultant about prototype demo expectations/logistics | 2024.07.04 | 6pm | 7pm | 1 | 101 |
| Install FINN compiler and read FINN papers | 2024.07.09 | 12pm | 4pm | 4 | 105 |
| Group meeting, share progress on FINN | 2024.07.10 | 12pm | 3pm | 3 | 108 |
| Install Vivado 2022.2 for FINN compiler requirement | 2024.07.10 | 3pm | 4pm | 1 | 109 |
| Follow FINN tutorial on an NN example to generate intermediate Verilog files | 2024.07.11 | 4pm | 6pm | 2 | 111 |
| Follow FINN tutorial to generate bitstream | 2024.07.12 | 2pm | 3pm | 1 | 112 |
| Prepare PYNQ-Z2 image | 2024.07.13 | 10am | 11am | 1 | 113 |
| Group meeting, share progress on FINN | 2024.07.17 | 12pm | 3pm | 3 | 116 |
| Consultant and group discussion, discuss prototype expectation and split model across boards | 2024.07.18 | 6pm | 7pm | 1 | 117 |
| PYNQ-Z2 board setup | 2024.07.23 | 2pm | 6pm | 4 | 121 |
| Run inference on board | 2024.07.23 | 5pm | 6pm | 1 | 122 |
| Group meeting, share progress on on-board inference | 2024.07.24 | 12pm | 3pm | 3 | 125 |
| Debug Verilator and gtkwave installation | 2024.07.24 | 9pm | 11pm | 2 | 127 |
| Prepare compilation demo program | 2024.07.25 | 8pm | 11pm | 3 | 130 |
| Final integration with Verilator | 2024.07.25 | 12pm | 3pm | 3 | 133 |
| Prototype demonstration with consultant | 2024.07.26 | 3pm | 4pm | 1 | 134 |
| Progress report write-up | 2024.07.26 | 6pm | 7:30pm | 1.5 | 135.5 |

Table 4. Xilin Bai - Time Log

# **Appendix B:** Initial Prototype Demonstration Feedback Sheet

