Zexin Fu

zexin.foo@gmail.com

Homepage: https://zexinfu.com (+86) 159-5959-2886

No.24 South Section 1, Yihuan Road, Chengdu, China, 610065

Education

Sichuan University, B.S. Microelectronics Science and Engineering 2019.9 - 2023.6

- **GPA 83.75/100**; Expected to graduate in June 2023.
- **English Proficiency:** Passed CET-4 and CET-6; TOEIC(scores of 840); Duolingo English Test(scores of 115).

Research Experience

An Analysis Framework for Instruction Fusion Targeting RISC-V ISA, Director 2023.02-2023.05

*Advised by **Prof. Peinan Li**.*

- Proposed an analysis framework for RISC-V instruction fusion scheme design.
- Listed 8 types of potential instruction pair implementations.
- Implemented a set of programs matched with the NEMU emulator for evaluating and analyzing fusion pairs.
- Analyzed Coremark based on the proposed framework: after adding our proposed instruction fusion pairs, the number of dynamic instructions decreased by 7.60% compared to the non-fusion situation.

A Heart Sound Recognition System Based on Nuclei E203 SoC, Director 2022.03-2022.07

*Co-advised by **Prof. Wanang Xiao** and **Prof. Bo Gao**. Awarded 2rd prize in CICIEC (Southeast).*

- Designed HSNet_Light, a lightweight CNN to classify MFCC-extracted features; Achieved 99.32% accuracy in the test set of Physionet; Reduced by 92.31% in parameters and 96% in calculations compared with the baseline.
- Adopted 16bit fix-point quantization method (Q11.4 for activations and parameters ultimatelyv after fix-point searches), gained a minimum loss of 6.38% in numerical error and 0.10% in classification error.
- Designed and realized a CNN accelerator with following characteristics:
  1) Support for most CNNs;
  2) Multi-way memory modules supporting 8-port R/W;
  3) A computing parallelism reaching 64;
  4) Using stride-conv instead of pool to reduce hardware resources;
  5) Support for 16-bit fixed-point operations.
- Attached the accelerator to the Nuclei E203 RISC-V SoC by adding custom extension instructions.
- Authored design documents and tutorials for instruction (available in the blog); Currently integrating the results and getting ready to submit a paper.

A Speaker Recognition System Based on FPGA, Core Member 2021.08-2021.11

*Advised by **Prof. Bo Gao**. Awarded 3rd prize in FPGA Competition.*

- Trained MobileNet1D neural network for the speaker recognition task based on TIMIT dataset, which reached the accuracy of 99.49% in the test set.
- Constructed a speaker recognition system on Xilinx ZYNQ board:
  1) ZYNQ receives sound through the audio module and delivers data to the PC with the UART interface;
  2) The PC recognizes voiceprints using the MobileNet1D neural network and displays the results on the screen.
- exploring the design space of neural network accelerators; drafted an accelerator of first version, which has been enhanced and applied in the next project of heart sound recognition.

FPGA Acceleration for Deep Learning Algorithm, Core Member 2021.08-2022.07

*Advised by **Prof. Wanang Xiao**. Supported by Innovation Training Plan of Institute of Semiconductors, CAS.*

- Mastered the characteristics and structures of commonly used convolution neural networks; Programmed in Pytorch for neural networks training and inference.
- Mastered the fixed-point quantization method of neural network in Python.
- Learned from various dataflow architectures for accelerators; Constucted and realized an FPGA-based CNN accelerator in Verilog for heart sound recognition application.
- Explored hardware algorithm co-design including loop tiling, compact networks and low-bit representation.

A Badminton Motion Correction System Based on Pose Estimation, Director 2020.12-2021.11

*Advised by **Prof. Bo Gao**. Supported by Innovation Training Plan of College of Physics, Sichuan University.*

- Deployed and test OpenPose algorithm, which generated skeleton keypoints of badminton images we collected.
- Proposed a method based on the angle between human joints to evaluate and correct the high clear movement divided into three parts for analysis.
- Programmed in Python to put this evaluation method into a practicable system with friendly user interface.

Awards / Honors

- **Second Prize, Southwest Area** (**Ranked 1st** in the undergraduate group of the RISC-V track), National College Integrated Circuit Innovation and Entrepreneurship Competition (CICIEC), July 2022.
- **Third Prize, National**, National Undergraduate FPGA Innovation Design Competition, November 2021.
- Intel China FPGA Engineer (Entry Level) Certificate, May 2021.
- Served as the vice president of the Algorithmic Robotics Association(SCU) (with a membership of 100+); Responsible for the study and competition training of digital design.

Knowledge / Skills

- **Digital Design**:
  - Studied courses in class including Digital Logic and System Design (scored 88/100), Advancing Fronts in Integrated Circuit Design (scored 92/100).
  - Familiar with digital design methods and FPGA development process; Practiced in Verilog programming and the use of Xilinx Vivado, Altera Quartus, Modelsim.
  - Completed several projects on Xilinx's Artix-7 series, ZYNQ-7000 series, Altera's Cyclone IV series boards.
- **Computer System and Architecture**:
  - Studied course online including Computer Organization (PKU, Coursera), Fundamentals of Computer System(NJU, iCourse), Operating System(HIT, iCourse).
  - Practiced in the use of NEMU (an instruction set simulator similar to Spike, but faster), including M/S/U mode(bare mental or booting Linux) and Simpoint Checkpoint.
- **Deep Learning and Its Hardware Implementation**:
  - Studied course online including Machine Learning(Hongyi Li, Homepage)(learned parts involving CNN), Intelligent Computing System(Yunji Chen, Homepage).
  - Familiar with the characteristics and structure of commonly used Convolutional Neural Networks;
  - Practiced in Pytorch, and capable of designing CNN structures, writing training and inference code.
- **High-level Language Programming**:
  - Practiced in C programming;
  - Practiced in Python programming and the use of libraries including Numpy and Pandas.
- **Other**: Practiced in the use of Makefile, Git, Markdown, Drawio, Wavedrom.