Kunal **Chandan**

B.A.Sc Honours Electrical Engineering

chandan.one | 814-807-7652
github/kunalchandan | kchandan@uwaterloo.ca | linkedin/kunal-chandan

---

Software: KiCAD, LTSpice, Cadence Virtuoso, LayoutEditor, Quartus Prime

Languages: Python, C++, SQL, Verilog, VHDL, MATLAB, Go, RISC-V

Libraries: Numpy, Pandas, PyTorch, Boost, FastAPI, Flask, CUDA

Education

UNIVERSITY OF WATERLOO – B.A.SC ELECTRICAL ENGINEERING

- Key Courses: Electronic devices, semiconductor physics, analog/digital integrated circuits, analog/digital/multivariable control systems
- Select Awards and Certifications: Baylis Medical Capstone Design Award, QNFCF and G2N Cleanroom Certifications

Experience

ELECTRICAL ENGINEER - COMPLIANCE

Enphase Energy
Aug 2023 - Present | Fremont/Petaluma, CA, USA

- Designed and implemented automated compliance testing for PV inverters to IEEE and UL standards
- Created business analytics and equipment management application, improving test equipment utilization by **20%** (**Ignition Perspective**, **Python**)
- Developed **MySQL** asset management database and interlock system ensuring regulatory compliance and saving $50K yearly

ELECTRICAL ENGINEERING RESEARCH ASSISTANT - DISPLAY SEMICONDUCTORS

University of Waterloo
Sept 2022 - Apr 2023 | Waterloo, ON, CA

- Designed custom PCBs in **KiCAD** for driving small μLED active/passive matrix displays using **STM32** microcontroller and accompanying circuitry
- Developed research plan for packaging of μLEDs onto TFT packplanes using indium electroplating
- Characterized results using **SEM** and **X-Ray Spectroscopy**,
- Designed characterization setups for μLEDs in **Fusion360** and **Arduino** interfaced with **Python**
- Validated flip-chip diebonding results with thermal and electrical simulations in **MATLAB**
- Designed and validated new μLED layouts to improve mechanical and electrical performance

SOFTWARE ENGINEER - FIRMWARE

Groq Inc.
Jan 2022 - Apr 2022 | Mountain View, CA, USA

- Defined algorithm for resource allocation over memory and processing units of tensors on Groq's TPU
- Developed **Python** and **C++** firmware API to improve streaming of instructions and data
- Used **PyBind11** for interoperability between C++ and Python firmware during codebase migration

SOFTWARE ENGINEER - DIGITAL COMPRESSION

Huawei Technologies
May 2020 - Aug 2020 | Waterloo, ON, CA

- Designed and analyzed non-cryptographic hash (NCHF) with linear algebra, SAT and self-designed $GF(2)$ matrix solver to verify properties
- Benchmarked the optimized SIMD hashing function against existing NCHFs (**Rust**, **C++**)
- Implemented novel border detection algorithm in **Go** using **probabilistic data structures** to maximize performance with Go-routines

SOFTWARE ENGINEER - MACHINE LEARNING

MappedIn
Sept 2019 - Dec 2019 | Waterloo, ON, CA

- Designed pipelines for data cleaning and analysis; integrated new **SQL** data warehouse
- Increased prediction accuracy from **40%** to **80%** on existing **LSTM** models with feature engineering, hyperparameter optimization, and automated data cleaning (**Python**, **SQL**)
- Created **Embeddings** + **SVM** + **Random Forest** ensemble models to replace existing LSTM models, reducing inference costs **2x** while maintaining prediction accuracy

Projects

BEAMFORMING HEARING AID SYSTEM

- Designed 4-channel microphone array PCB with active analog bandpass filtering, differential amplification, and multichannel **ADC** over **SPI** to Raspberry Pi (**KiCAD**)
- Created **Flask** server on R-Pi to compress and transfer audio data to **Pytorch** neural network for further digital filtering and beamforming
- Adapted and trained Pytorch quantized voice isolation model to minimize latency while maintaining desired audio quality
- Used **multiprocessing**, **asyncio**, and **websockets** to maximize system throughput, providing continuous audio output

PIPELINED RISC-V CORE

- Designed 5-stage pipelined **RISC-V** 32-bit core in **Verilog** using only synthesizable constructs
- Core synthesized on FPGA and successfully ran branching and recursive algorithms. Testbenches used to ensure cycle accuracy