




# MARCOS AUGUSTO FERREIRA

 (312) 818-0709

 mferreira103301@gmail.com

 [linkedin.com/in/marc103](https://www.linkedin.com/in/marc103)

 [marc103.github.io](https://github.com/marc103)

## EDUCATION

### Illinois Institute of Technology

Bachelor of Science, Computer Science (2020 - 2024)

Graduated: May 2024

Major GPA: 3.96

## TECHNICAL SKILLS

**Languages:** Verilog, System Verilog, TCL, Python, C, C++, Java, Scala (Chisel)

**Hardware Design:** FPGA Prototyping, RTL Design, Synthesis, Timing Closure, CDC (Clock Domain Crossing), FEV (Formal Equivalence Verification) on testbenches, BRAM, DSP, UART, I2C, USB

**Frameworks & Tools:** Vivado, Lattice Diamond, Yosys, ModelSim/Questa, Git, Linux

**Specialities:** DSP, Real-time Image Processing, Computer Vision Acceleration, Floating-point & Fixed-point Arithmetic, Pipelining, Parallel Processing

## PROFESSIONAL EXPERIENCE

### BiV Lab - Northwestern University

August 2024 - Present

FPGA Developer as a full-time funded research assistant

Chicago, IL

- RTL designed and verified using FEV to implement novel real-time computer vision algorithms
- Image stream processing, for efficient usage of BRAM, involves convolution filters, bilinear interpolators and various arithmetic modules, implemented on DSPs, to compose a hybrid fixed point to floating point system to minimize resource utilization whilst maximizing accuracy
- Co-first authored a submission to CVPR, for details & live demo, please contact me

### Qualcomm

May 2023 - August 2023

Software Engineering Intern

San Diego, CA

- Analyzed the behavior of the Neural Signal Processor under synthetic / real loads
- Developed Python scripts to automate chip/device configuration, benchmarking, and capturing and visualizing performance metrics

## PROJECTS

### Floating Point Utilities

[github.com/Marc103/Floating-Point-Image-Processing-SV-RTL](https://github.com/Marc103/Floating-Point-Image-Processing-SV-RTL)

- Developed pipelined floating point adder, multiplier and divider units and then window fetchers and floating point MAC units that assemble these pieces to perform convolutions in floating point
- Wrote a system verilog generator in python to optimize MAC operations for kernel matrices that are sparse and/or contain powers of two values

### Image Processing RTL Modules

[github.com/Marc103/Image-Processing-SV-RTL](https://github.com/Marc103/Image-Processing-SV-RTL)

- See a Convolution Filter and Bilinear Interpolator written and tested in System Verilog, using adjustable Fixed Point Arithmetic

### Debayering System

[github.com/Marc103/OV7670-with-FPGA-and-Demosaicing](https://github.com/Marc103/OV7670-with-FPGA-and-Demosaicing)

- Worked on embedded digital image processing with FPGAs to perform demosaicing bayer pattern
- Built an I2C driver to communicate with the camera being used (OV7670)

### LC-3 ISA CPU

[github.com/Marc103/LC-3-Verilog-Implementation](https://github.com/Marc103/LC-3-Verilog-Implementation)

- Implemented a CPU that follows the Little Computer 3 (LC-3) instruction set architecture in Verilog
- Performed simulation tests in Vivado Design Suite for QA and programmed Xilinx FPGA
- Leveraged UART as the communication protocol to send and receive data to a terminal emulator