

# Ian Chang

[iac@berkeley.edu](mailto:iac@berkeley.edu) | 310 890-2413 | [github.com/iachang](https://github.com/iachang)

Goal: Develop my career in electrical engineering and computer science as an intern at a high-tech company

Education	<p><b>UNIVERSITY OF CALIFORNIA, BERKELEY</b>, 2018-2022.</p> <p>BS in Electrical Engineering and Computer Sciences (Emphasis in Computer Engineering) GPA: 3.74/4.0 Courses: Data Structures, Structure and Interpretation of Computer Programs (Programming Paradigms), Designing Information Devices &amp; Systems I (Linear Circuits, Linear Algebra), Designing Information Devices &amp; Systems II (Non-linear Circuits, Robotics &amp; Controls, Interpolation)</p> <p><b>SANTA MONICA COLLEGE</b>, 2015-2018.</p> <p>Entry Level Programmer Certificate (Dual Enrollment) GPA: 4.0/4.0 Courses: C Programming, Internet Programming, Visual Basic Programming, Intro to Computer Science</p>
Honors	<p>UC Berkeley Bowles' Hall Honor Roll Crawford Memorial Mathematics Scholarship USA Computing Olympiad Platinum USA Computing Olympiad Gold Perfect Scorer Santa Monica High School Salutatorian (Class of 700)</p>
Employment	<p><b>Bitcoin Presenter, Santa Monica Public Library.</b> April 2018</p> <ul style="list-style-type: none"><li>➤ Arranged and founded the "Introduction to Bitcoin" library program, teaching over 35 citizens on the technical workings of Bitcoin technology.</li><li>➤ Sold-out library event, averaging 4 out of 5 star reviews.</li></ul>
Research	<p><b>Research Assistant (under Prof. Puneet Gupta), NanoCAD Lab, UCLA.</b> September 2016-June 2018</p> <ul style="list-style-type: none"><li>➤ Developed Verilog parser using Python to re-arrange FPGA I/O chip hierarchies using tree data structures and depth first-search algorithms. Replaced manual rearrangement of I/O blocks with instantaneous, automated block rearrangement to expedite FPGA code design process.</li><li>➤ Architected Verilog block hierarchies and unit-tested results with Xilinx ISE Design Suite.</li><li>➤ Implemented a Mac OS X port for an open-source memory benchmarking tool (X-Mem) using C++, POSIX, and Mach thread libraries. Increased users of X-Mem in utilized courses by 30% (previously only Windows &amp; Linux supported)</li><li>➤ Integrated Clang compiler support and SCons compatibility for X-Mem Mac OS X developers using Python, expanding compiler compatibility to Mac OS X systems.</li><li>➤ Designed real-time data visualization website for X-Mem that uploads and parses X-Mem CSV results using MySQL, PHP, Plotly.js, and Bootstrap CSS. Created a system where users can share and compare their memory benchmarking results online, replacing the old-fashioned and inefficient method of physical file transfer and loading the CSV results in a spreadsheet program.</li></ul>
Teaching	<p><b>Teaching Assistant, Los Angeles Computing Circle, UCLA.</b> 40 hours. July 2016-August 2016</p> <ul style="list-style-type: none"><li>➤ Mentored and outreached to high school students in college-level material, including fast and efficient algorithms, graph theory, and mobile-development as part of a UCLA EE Department hosted program.</li></ul>
Volunteering	<p><b>Programming Volunteer, Santa Monica Public Library, Santa Monica, CA.</b> Sept 2017 – October 2017</p> <ul style="list-style-type: none"><li>➤ Collaboratively developed a video game using PyGame that was publicly featured and played by library visitors during International Failure Day.</li></ul>
Projects	<p><b>Coinlet — Bitcoin/Ethereum/Litecoin Price Tracker</b></p> <ul style="list-style-type: none"><li>➤ Developed RESTful Android application to track cryptocurrencies prices in real-time using Android Studio, Retrofit Library, and Coinbase API.</li></ul> <p><b>Robotic Motor Car</b></p> <ul style="list-style-type: none"><li>➤ Built a motorized car utilizing an MSP430 microcontroller, encoders, frequency-response filters, and open-loop model to drive multiple directions/angles.</li></ul> <p><b>Scheme Interpreter</b></p> <ul style="list-style-type: none"><li>➤ Built a Scheme interpreter that implemented Read-Evaluation-Print-Loop schematic and tail-recursion optimization using Python.</li></ul>
Languages	Java, Scheme (Functional Programming), Python, SQL, Verilog, C, Bash
Skills	Git, Vim, Tmux, IntelliJ IDE (Debugger), Unit Testing, Soldering, Circuit Design and Implementation