Eric Crosson
706 W. 34th #202 Austin, TX 78705
(512) 222 - 9052 / esc@ericcrosson.com / www.ericcrosson.com

Bachelor of Science in Computer Engineering at University of Texas 2016
Recent classes include embedded & real time operating systems, embedded systems design, computer architecture
Interested in automation via software abstraction, GNU/Linux, free software, LISP, formal verification

---

Extra Curricular Activities

- Self-study of functional programming with a strong emphasis on LISP, formal verification (ACL2), DevOps
- Founded FIRST Robotics Competition (FRC) team 3529; mentor FTC, FLL, Jr. FLL teams

Hobbyist programmer, GNU/Linux aficionado, Emacs appreciator

- Expertise in LISP, git, Ruby, C/C++, Python, Java, Perl, Bash, asm, LaTeX, documentation, CMake
- Familiar with Octave, Haskell, JavaScript, Tcl, VHDL, S. Verilog, CI, Docker, Packer, Vagrant, Selenium, agile

---

Work Experience

ShoreTel- Software Engineer (2012 - 2013)

- Created IP Phone VNC client and unit test architecture/DSL for firmware regression

Intel- Post Silicon Validation (2013 - 2014)

- Managed tests to determine quality of hardware
- Wrote, executed tests to stress hardware components

Intel- Pre Silicon Validation (2014 - 2015)

- Created analysis engine for internal signals of 3rd party RTL
- Integrated DHCP model tests against project RTL

Centaur Technology- Design Verification (2013)

- Integrated processor simulation with virtual computer to allow for cycle-accurate simulations of presilicon hardware on FPGAs

Centaur Technology- Design Verification (2014)

- Created multicore PSE36/PAE paging test generators
- Designed, implemented true LRU in System Verilog

IBM- Cloud Infrastructure Services (2015)

- Community work with OpenStack
- DevOps management of public cloud
- Created API to manage production accounts

Self-motivated projects

- http://www.github.com/EricCrosson
- Programming contests in Java, C++ and z80 asm
- Wavelength to RGB conversion with ADC feedback
- Founded competitive UIL computer science team (taught C++, Java, Perl)
- Static image background extraction
- Eye-gaze projection software
- Cell-tracking image processing
- And much more

Autodidactic

- Seeking knowledge from OpenCourseware, Coursera, edx, and other manuscripts
- Classes taken: machine learning, neural networks, big data, algorithms II, cryptography II, hardware security