

ANDY GOETZ

503.708.7958 • andy@andygoetz.org • andygoetz.org • github.com/apgoetz • linkedin.com/in/andygoetz

OBJECTIVE Obtain an entry level position as an embedded systems software and hardware engineer.

B.S. Computer Engineering, graduation Spring 2013, **GPA 3.85**
Portland State University

WORK EXPERIENCE **Research Assistant** Winter 2013–Present
Portland State University

- Obtained grant to study performance of heterogeneous 3d network-on-chip.
- Using ParadisEO framework to study on-chip network architectures constructed out of heterogeneous link types.

Computer Engineering Intern Spring 2012–Summer 2012
Electro-Scientific Industries, Portland, OR

- Developed control software for high speed ceramic capacitor test equipment.
- Used Computational Intelligence techniques to monitor air pressure in high-speed transport system.

Software Intern Summer 2010–Spring 2012
Intel Corporation, Hillsboro, OR

- Developed software in C#/C++ to automate chipset validation.

Software Developer Winter 2010–Spring 2010
Business Solutions Group, Corvallis, OR

- Used C# and ASP.net to develop dynamic web applications.

LEADERSHIP EXPERIENCE Secretary of the Oregon Beta Chapter of the Tau Beta Pi Honor Society.
ECE tutor for the engineering college at Portland State University.
Project leader for 4-person Engineering Practicum.
Project leader for 3-person Engineering Capstone.

SKILLS Experience with Windows and Linux environments
Programming in C/C++, Java/C#, ARM/z80/MIPS Assembly, Perl, Verilog.
Experience with version control systems, including TFS and git.

HONORS AND AWARDS Semiconductor Research Corporation grant to study 3d on-chip networks.
Honorable mention for the *T-16 Audio Synthesizer* in Practicum Competition.
“Spur the Competition Award” for predictor design in Computer Architecture class.
Member of HKN Honor Society *Limited to top 25% of Department*
Member of TBII Honor Society *Limited to top 12.5% of College*
Member of ΦKΦ Honor Society *Limited to top 7.5% of University*

Project Descriptions Continue on Next Page

MAJOR PROJECTS

Explored Heterogenous 3d On-Chip Network Design *(C++, Perl)*

I used evolutionary-programming techniques to evaluate alternative interconnect topologies for on-chip bus designs. This involved heavy use of symmetric multiprocessing to harness the full power of our simulation servers.

Developed a simple processor cache simulator *(Verilog)*

As the final project for a microprocessor system design course, I worked with several classmates to design a processor cache simulator. This simulator would take in a plain text file containing a sample trace of memory operations, and determine the hit ratio of the cache. I was mainly responsible for developing the trace file parser, as well as developing a testbench to verify the cache design.

github.com/ekrause/0xBEEFA55

Developed a Branch Target Predictor Simulator *(C++, Perl)*

For the final project of a Computer Architecture class, I worked with a classmate to develop a branch predictor simulator. This project explored several different predictor designs. At its peak, the parallel simulation kept PSU's engineering computer labs pegged at 100% usage. The project won several awards as a part of an in class contest, including the 'Spur the Competition Award', and 'Most Conspicuous Consumption of Computing Cycles'.

github.com/apgoetz/CBTB

Designed and implemented an ARM-based synthesizer *(C, KiCad)*

For my Industry Design Processes class at PSU, I worked with a team of 3 other students to design and build a microcontroller-based project. We developed a microcontroller based-audio synthesizer capable of generating up to 6 frequencies simultaneously. Gathered requirements, prototyped solutions, designed a PCB, and implemented all of the firmware in a single, 10-week semester.

github.com/killerfriend/womprats

Developed Linux kernel driver for data acquisition board *(C, Linux)*

As part of my Linux Device Drivers course at PSU, I developed a Linux kernel module for a USB data acquisition board. This driver was capable of supporting multiple data acquisition boards connected simultaneously.

github.com/apgoetz/linux_labjack

Developed novel algorithms for compressing debug traces *(C#, C++)*

Modern microprocessors are capable of generating an enormous amount of debug data. While at Intel, I developed software tools to extract debugging data from development platforms, and make it useable for validation engineers.

Developed firmware for pneumatics control board *(C, C#)*

While working at ESI, I developed firmware features for a pneumatics control board that used computational intelligence techniques to automatically determine if the board was functioning correctly.