**Hardware and Software Engineer**

Electrical engineer with significant experience in VLSI digital and analog circuit design, extensive skill in using Perl and Tcl/Tk to build design infrastructures and to automate design tasks. Strong contributor to Intel server processors across many CMOS process generations: IA-64 server processors (McKinley, Tukwila, Poulson), and Xeon server processors (Haswell Server, Broadwell Server, Denverton). Known for a wide breadth of skill, and a strong ability to learn new skills as proven by my professional experience across many different disciplines including hardware, software, firmware, and verification. Passionate about solving cool technical problems, and educating others in hardware and software engineering.

**Technology and Platform Expertise**

**VLSI:** Signal Integrity concepts (transmission lines, S-parameters), Power, Static Timing Analysis, Electrical Rules Checking, Random Variation, Reliability, ESD

**Languages:** Perl, Tcl/Tk, Python

**Circuit design tools:** Cadence Virtuoso (schematic capture), Cadence ADE XL (SPICE), Synopsys Discovery AMS (VCS-XA), Keysight Technologies ADS (3D EM field solver)

**Software engineering:** Linux, software configuration management, object oriented design, design patterns, SQL

**Communication Tools:** Microsoft Word, Excel, PowerPoint, Visio

**Career Summary and Achievements**

**Intel Corporation**, Fort Collins, CO 2005 - 2016

**Staff level Analog Engineer**

* **Designed 3-terminal inductors (T-coils) for high speed serial I/O receivers and transmitters for use in 20.625 Gb/s Thunderbolt3 and 16 Gb/s PCIe Gen4**
  + Optimized Rx and Tx T-coil inductor values through detailed simulations of insertion loss, return loss, and ESD. Used Keysight Technologies ADS to create S-parameter models of inductors from a full 3D electromagnetic extract.
  + Simulated inductive coupling between Rx and Tx pads, and also between pads and PLL inductor.
* **Led an innovation team to define an analog design environment that addresses inefficiencies in creating simulations and communicating results**
* **Organized and led an analog VLSI self-learning class**
* **Created the infrastructure for mixed signal validation of Broadwell Server’s PCIe Gen3 I/O receiver adaptation loops using Synopsys Discovery AMS**
  + Enabled design team to perform volume simulations of key analog circuitry (Continuous Time Linear Equalizer and Decision Feedback Equalizer) with real digital Verilog control (Clock Data Recovery) to verify the analog circuitry could receive data without bit errors. *CTLE and DFE circuit designers improved their circuits based on these simulation results which avoided post-silicon electrical issues.*
  + Wrote Perl scripts to automatically launch over Netbatch the running and data collection of the CTLE amplification gain, the 4-tap DFE coefficients, the clock phase of the CDR, eye height, and eye width, across variations in process, voltage, temperature, channel lengths, and PCIe clock frequencies.
  + *The team working on the Ethernet I/O leveraged my work to perform similar analysis for their design.*
* **Designed the delay locked loop (DLL) for Haswell Server’s 9.6 Gb/s QuickPath Interconnect**
* **Designed the clock duty cycle correction circuit and switched capacitor comparator circuits for Haswell Server**
* **Designed a low power, low offset, folded-cascode topology opamp for use across the Forwarded Clock Amplifier, DLL, CTLE, and bias generation blocks**.
  + *Consolidated on this one design to avoid multiple different opamp topologies. Low offset behavior also eliminated the need for an external calibration circuit and an extra calibration loop in the RTL.*
* **Created and maintained a macro API layer in Tcl for efficiently creating analog circuit simulation testbenches for Haswell Server**
  + Presented this work at the Intel Design Test and Technology Conference (DTTC) in 2012.
  + *Improved the efficiency of several team members in creating their circuit simulations. Rather than writing low-level Tcl commands, they could use a high-level macro. For example, creating the waveforms for short term and long term duty cycle amplification, and measuring and reporting the results.*
* **Built digital clock frequency divider capable of a divide by 2 through divide by 15. Used throughout Denverton microserver product**

**Staff level Design Automation Engineer**

* **Development team member for a Register File memory array automation for the Poulson IA-64 Server**
  + Team defined the methodology and tools to centrally create and deliver over 30 tape-out quality register files.
  + Wrote a launch table GUI in Tcl/Tk to manage all the arrays.
  + Pioneered the use of a wire routing database, using SQLite, to handle general routing constraints, but also to handle corner cases. *This was a key enabling feature for our central team to be able to deliver arrays to customers without their need for further cleanup.*
  + Our group won the Intel Convergence Award at DTTC 2008 for our effort to standardize and propagate this methodology.
* **Created and maintained multisite design data synchronization Perl scripts for Tukwila IA-64 Server.**
  + *Improved upon original manual process. Enabled the Intel Fort Collins CO site and the Intel Hudson MA site to collaborate on Tukwila using the Hewlett-Packard Itanium design team’s custom VLSI CAD tools.*
* **Designed and built the design data configuration management infrastructure in Perl for Tukwila IA-64 Server**
  + Introduced and championed configuration management practices to revision control the physical design data. Wrote Perl scripts to check in data, check out groups of data that were tagged to belong together, ECO capabilities for post silicon design edits. ***We went from not protecting our design data at all to this engineering best practice because of my effort.***
  + *This effort, along with my work on standard cell validation scripts and my role as the design tools release manager, earned me a promotion from senior engineer to staff engineer.*

**Hewlett-Packard Company**, Fort Collins, CO 1993 - 2005

**Senior Design Automation Engineer**

* **Built the Perl infrastructure to validate and release the VLSI standard cell library for Tukwila processor**
  + Solved numerous issues with correctly validating a large standard cell library in a timely manner by creating a validation algorithm based on the depth first search of the cell hierarchies. This solution included a custom parallel launch algorithm over many compute machines.
  + *This solution completely eliminated library releases that contained invalid cells. Design engineers were no longer stalled waiting for embarrassing re-releases.*
* **Managed validation and releases to the design team of custom VLSI physical design tools**

**VLSI digital design and other relevant engineering**

* Provided IA-64 PAL and SAL firmware support for post-silicon validation of a processor module consisting of dual Madison IA-64 processors plus an external third level cache
* Created schematics and layout for Itanium2 hardware page walker. This was a standard cell based digital logic design.
* Helped verify a workstation I/O chipset by debugging Merced bus failures sourced from random code generators
* Wrote Verilog for a finite state machine controller interface for Synchronous Graphics RAM for a low end workstation graphics accelerator.
* Designed PC board layout for a dual headed graphics display card for HP low end workstations

# Publications

**Register File Automation for Modern Microprocessor Design.** Poster presented at Intel DTTC 2008.

**Poulson RF Automation: Strategy for Delivering Tape-Out Quality Designs.** Paper published in DTTC 2010 proceedings.

**Analog Circuit Simulation and Data Standardization.** Selected for presentation for DTTC 2012. I was first author and presented.

# Education

**Master of Science Electrical Engineering**, Stanford University, 1993

**Bachelor of Science Electrical Engineering**, Cornell University, 1992

Relevant completed Coursera courses: Introduction to Power Electronics (University of Colorado), Machine Learning (Stanford), Digital Signal Processing (EPFL)