Adam Greenblatt
1103 ʻĀinalako Road
Hilo, HI 96720-3722

adam.greenblatt@gmail.com
(808) 896-5265
linkedin.com/in/adamgreenblatt

Objective Fun, challenging software projects, either as a consultant or as a remote employee.

Skills Decades of programming experience in C/C++ under Linux, custom embedded OSes, and Windows. Expert at optimizing speed and size of critical algorithms. C/C++ compiler internals: new CPU support, language extensions, runtime library design. CPU simulator design and implementation. Conversant in LISP, SQL, bash, perl, various assembly dialects, git etc. Rapidly learn new languages on demand. Digital and analog IC design. Hardware troubleshooting. Pyrotechnics. Unicycling through bonfires.

Education

Bachelor of Science with Honors, Caltech 6/1988

Experience

President, Cycle Counters, Hilo, HI 1/2002 to present
Ran a husband and wife consulting firm that implemented a wide variety of software and firmware projects, including real-time image processing, custom compilers, device drivers, etc. Clients include Altera, Ixia, Motorola, Rapid Prototypes, Synaptics, and Zebra Imaging. See "Sample Project Details" below.

Principal Member Technical Staff, Oracle, Redwood Shores, CA 6/1996 to 2/1999
Redesigned Oracle RDBMS portable memory interface, wrote Solaris implementation of same. Added support for server-based Java memory management. Wrote widely used internal source code control tools.

Senior Member Technical Staff, Oracle, Redwood Shores, CA 12/1993 to 6/1996
Designed and co-implemented Blaze, a lightweight SQL database shipped as part of Oracle Power Objects. Designed and implemented Power Objects' database interface, permitting transparent access to Blaze, Oracle, and Sybase databases.

Design Engineer, Synaptics, San Jose, CA 5/1990 to 12/1993
Analog IC design, including simulation, layout, fabrication, and testing. Designed and implemented C language extensions for transparent multiprocessor communication, lisp-based automated circuit layout, hardware accelerated neural network simulations, and general purpose image processing.

Technical Staff, TRW, Redondo Beach, CA 2/1988 to 4/1990
Designed and simulated the FDF-3 text-processing ASIC. This device, capable of performing complex pattern matching at 25-100 Mbytes/sec, later became the basis for the company Paracel. Designed and implemented BSL, a domain-specific language for geneticists using the earlier FDF-2 ASIC.

Patents and Honors

Patent 5,243,554 – Writable analog reference voltage storage device	1992
Patent 5,166,562 – Writable analog reference voltage storage device	1991
TRW Award for Innovation	1989
1st Place, ACM International Scholastic Programming Contest	1988
Carnation Prize for Academic Excellence	1985

References Available upon request.

Adam Greenblatt
1103 ʻĀinalako Road
Hilo, HI 96720-3722

adam.greenblatt@gmail.com
(808) 896-5265
linkedin.com/in/adamgreenblatt

Sample Project Details

A C/C++ compiler and high-speed simulator for a custom CPU

The CPU in question has a number of unique features (for example, no hardware stack, multiple separate address spaces, etc.) that made my gcc port non-trivial. The compiler eventually grew to support several dozen variations on the CPU, with billions of units shipped in aggregate. The simulator runs faster-than-real-time, and includes modeling of multiple hardware hyperthreads and an associated vector coprocessor.

A real-time photometrics system for a custom 20 meter hemispherical display

This system used about 100 off-the-shelf projectors to produce a single seamless image across the entire irregularly shaped display. I used an off-the-shelf telescope mount and a pair of carefully calibrated digital SLR cameras combined with image analysis software to determine the precise 3D shape of the screen, as well as the precise 3D extent of each projector's throw, to millimeter accuracy. Careful lens calibration was required: no two SLR lenses are *sufficiently* identical to be usable uncalibrated. The final system produced seamless "retina-display" quality videos across the entire screen. This project involved a fair amount of higher mathematics, particularly in computing minimum worst-case error surfaces from noisy data.

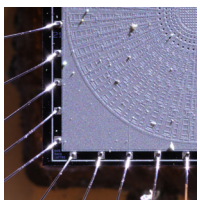
A synthetic hologram production system

This system used transparent LCDs to serve as an artificial source of wavefront data in lieu of physical objects to make large white-light viewable full-color holograms. My role was to generate roughly 200 megabyte/second streams of data to feed the LCDs in perfect synchronization with the stepper motors that were raster scanning the images onto the roughly 1 meter square sheets of holographic film. The hard part was avoiding any stuttering that would ruin one or more voxels in the final image. This was around 2005, when sustaining 200 megabytes/second was non-trivial.

An iOS application for the disabled

This was an alternate keyboard for people with limited dexterity to enter text into a phone or tablet. The app worked well: I could type easily with my big toe. It was an interesting exercise in working in the iOS ecosystem, but was not a commercial success.

A variety of novel image sensors



I designed, fabricated, and tested some novel image sensors. They used simple analog circuits in each pixel to perform local computation on the image, analogous to how mammalian visual cortex performs edge and motion detection. The resulting images were easier for the limited computers of the day to analyze. (See misc.cyclecounters.org/wruv.jpg)

I've written a lot of other software as part of these and other projects. I chose the examples above to highlight my versatility and the breadth of my experience.