Victor Sperry

444 210th Ave NE • Sammamish, WA • 98074
CELL (925) 354-5330 • **E-MAIL** vicsperry@gmail.com

EXPERIENCED SOFTWARE/FIRMWARE ENGINEER

Embedded • C • C++ • Python • Linux • git • Bare Metal • VxWorks • Automotive • Microcontrollers • Networks

Hi, I'm Vic. While studying engineering in college, I discovered that I really wanted to write software. I changed my major to Computer Science and never looked back. Over the next 34 years, I developed successful products in the defense, automotive and networking industries. In 2018, I was immersed in the world of drones and autopilots, learning all I could about the software and devices enabling these amazing new types of vehicles. Currently, I am working as a consultant for Facebook Reality Labs, writing firmware and learning about what goes on inside virtual reality headsets.

PROFESSIONAL EXPERIENCE

R&D Embedded Firmware Engineer • Facebook Reality Labs (Oculus) • Redmond WA **Jul 2019 • Present**

- Working for FRL via vendor Fresh Consulting writing embedded firmware for a variety of R&D projects using a mix of processors, languages, and tools. Wrote C code for MicroBlaze soft-CPU inside Xilinx FPGA using Xilinx SDK (Eclipse-based C environment). Development on Fedora Linux. Added functionality to existing Python tool. Ported Linux code to several RISC-V implementations, the SiFive High Five Unleashed (runs Linux) and the E34, runs inside a Xilinx FPGA on freedom-e-sdk (bare metal). JTAG debugging using Ozone debugger.
- Wrote drivers for hardware accelerators running in an FPGA. Visual Inertial Odometry C++ algorithm running on Linux under Fedora. Drivers run in user space and access FPGA registers over PCIe via /dev/mem and mmap(). Demoed the use of DMA to speed up user space drivers.
- Investigated porting parts of VIO code to ARM running on MPS2+ board. Cortex-M33 Soft-CPU running inside FPGA. Zephyr operating system on M33.

Embedded Software Engineer • Terrafugia • Petaluma CA **Apr 2018 • Mar 2019**

- R&D software for dem/val of TF-2 automated full-sized eVTOL aircraft. Leverage ArduPilot open source autopilot (C++ running on ARM in PixHawk cube) to control custom sub-scale drones. Modified supporting Python code, e.g. MAVProxy. Work with controls engineers to implement and refine algorithms. Develop Terrafugia's own code base in C++.

Principal Software Engineer • Calix • Petaluma CA **Dec 2006 • Feb 2018**

- Development for Calix' AXOS platform (multiple boards running Linux on custom hardware) using a variety of web-based tools such as Jama Contour for requirements, Jira for work tickets (bugs, Agile stories), Fisheye/Crucible for code reviews, git for version control. Build/development machines ran Ubuntu.
- Added support for a variety of new fiber and copper modules (QSFP, CDFP, NGPON2) to multiple new Calix cards, keeping the code consistent across multiple git branches.
- Brought up LED support on a Broadcom DNX switch for a Remote Optical Line Terminal Card. This involved understanding how a tiny embedded LED processor worked inside the DNX switch and writing assembly for the LED processor to make it work with our FPGA.
- Member of a small team that brought up the physical layer of Calix' first G.fast (DSL) line card. Abstracted the physical layer (based on ITU G.997.2) to ensure future support of new G.fast chips; helped code the driver to the Sckipio G.fast chipset. Code written in a mixture of C and C++.
- Designed and developed all aspects of the physical layer software for Calix' E7 VDSL2 line card, which uses the Broadcom bcm65300 VDSL2 chip and supports board-level vectoring.

- Developed a Java wizard for inclusion in the Calix Management System (iMS/CMS). Developed on a PC using Eclipse with Clearcase for version control.
- Designed and developed all aspects of the physical layer C software for the Calix C7 VDSL2 line cards, including: a driver for the Broadcom bcm6511 VDSL2 chip; selection and definition of managed objects for VDSL2 in the Calix C7 access platform; prototyping screens (HTML) to communicate the managed objects to other team members; implementing the middleware for the line unit necessary to present the managed objects to the rest of the C7; support for board and chip bring-up; debug; consultation on VDSL2 issues; documentation of design.

Senior Software Engineer • Zhone Technologies • Oakland CA **Aug 2005 • Nov 2006**

- VxWorks Driver development (SENS END) for Zhone's Malc product. I developed G.SHDSL drivers for several line cards and SELT/DELT (Single/Dual-Ended Line Test) for ADSL/2/+ line cards. Processor is MPC860.

Member Technical Staff • Paradyne Corporation • Largo FL **Jan 1996 • Aug 2005**

- Acted as lead software engineer for various releases of the 8820 chassis software. This is a chassis with an ATM backplane holding up to 20 cards working together as an ATM switch over various technologies such as xDSL, T1/E1, T3/E3, and Sonet. As a lead, my responsibilities include ensuring that all software tasks are captured on the schedule, estimating the task duration, and assigning people to tasks. This usually involved from 5 to 20 people, depending on the release. I was also responsible for writing release scripts, building official releases, and ensuring that official releases contained the correct versions of software modules. Version control was done with Clearcase. The lead also documents procedures that are important to other developers such as how to build code, debugging tips, etc.
- I was also a developer for the 8820. Software written in C for VxWorks, running on a variety of processors depending on the card. For example, IBM405 PowerPC, MPC860, and Wintegra (MIPS core). My specialty as an 8820 developer was board support packages for VxWorks, custom VxWorks network drivers for getting data into the management stack (i.e. SENS ENDs), design of the management plane IP topology, and any other infrastructure needed to make IP management plane applications work (for example telnet, NFS, FTP). Troubleshooter for VxWorks applications, especially running on PowerPC.

Software Engineer • Beckwith Electric • Largo FL **Aug 1995 • Nov 1995**

- Real-time embedded software for 68HC11 targeted for electric utility industry.

Contract Software Engineer • Sypris Solutions • Tampa FL **Mar 1995 • Aug 1995**

- Sypris was Group Technologies when I was there. Designed Ada communications software for military-hardened 486 PCs running UNIX. Software controls Tactical Communications Interface Module (TCIM), a military SCSI card.

Project Engineer • Delphi Corporation • Kokomo IN **Nov 1990 • Feb 1995**

- Delphi was Delco Electronics when I was there. Design real-time automotive software targeting 68HCxx microcontrollers. Responsible for all phases of software: requirements analysis, design, code, test, integration and maintenance. Designed complete software systems including executive, interrupts, I/O and control logic. Technical software lead for Delco's load control multiplex products team. Development environment was UNIX on SUN workstations. Code written in assembly. Used Motorola in-circuit emulators with HP logic analyzers for test and debug.

Software Engineer • Lockheed Martin • Orlando FL **Dec 1986 • Oct 1990**

- Lockheed Martin was Martin Marietta Electronic Systems when I was there. Design real-time systems using CASE tools including data flow analysis, object-oriented and structured design. VAX Ada and VAX/VMS experience. Ada code for VAX computers involved real-time data collection and playback, simulation, and user interfaces. Using C and assembly, programmed real-time embedded signal processor composed of 68020 single-board computers.

EDUCATION

University of Central Florida • Orlando, FL • Bachelor of Science in Computer Science (BSCS) • 1986
20 hours coursework toward MSCS from 1988-1990.

TRAINING

- LFD435: Developing Embedded Linux Device Drivers, The Linux Foundation, Nov 2020
  https://www.youracclaim.com/badges/87a97da6-91ff-4ba5-bbfd-ab6d90e9127c?source=linked_in_profile
- LFD420: Linux Kernel Internals and Development, The Linux Foundation, Nov 2019
  https://www.youracclaim.com/badges/7dab392a-53d0-42f3-bff4-992f7a789fd1?source=linked_in_profile
- Flying Car nanodegree, Udacity, Dec 2018
- Intro to Self-Driving Car nanodegree, Udacity, Sep 2018