

# Ross Wolin

Everett, WA

rowol@mysticind.com

## OBJECTIVE

Firmware/hardware/software design challenges.

Provides fully equipped lab and office: 100% offsite projects only.

## EDUCATION AND HONORS

BS in Electrical Engineering, 1987, Rensselaer Polytechnic Institute, Troy, NY, GPA 3.6 (obtained in 3.5 years)

Eta Kappa Nu, Electrical Engineering Honor Society

Dean's List and RPI scholarship recipient, all semesters

Valedictorian, Norwich High School

## EXPERIENCE SUMMARY

Firmware: ARM Cortex M0/M3/M4/F4/H7/R5/A8/A9/A53, ARM 11/9/7, Nordic nRF52840, STM32F/H, MX6/MX53/MX51/MX31, OMAP, 88AP510, Coldfire, 68332, PPC, DSP, 80x86, PIC, AVR, 8051, 68HC11/12, Z80, EZ-USB, VHDL, GCC/GNU, MPLAB, SmartSnippets, CodeWarrior, SDS, Microtec, HiTech, Cosmic

Hardware: VHDL, Xilinx, Altera, FPGA/CPLD/ASIC, OrCAD, bringup/debug, rework, PCB layout

Software: Linux, Unix, Zephyr, FreeRTOS, Bare Metal, Android, C/C++, C#, Assembler, Python, Perl, Bash Shell Scripting, device drivers, OpenGL, DSP, Qt, Git, Mercurial, SVN

Misc: USB, BLE, 1394/Firewire, RS232, SPI, SCI, I<sup>2</sup>C, A/D, LCD, TCP/IP, IrDA, CAN, ICE, ICD, BDM, JTAG, MMC

## PROJECT EXPERIENCE

### MAPIR, Firmware Engineer, San Diego, CA, 3/24-11/24

MAPIR makes a multi-camera ethernet connected array with Ardupilot compatible controller, used in drones

- Yocto/Linux/BSP + U-Boot firmware development on NXP iMX6 ARM Cortex A, using C/C++
- Created/modified/fixed Docker based Yocto build environment
- Modified/deugged/fixed various problems with U-Boot, Linux kernel, device tree, drivers, startup, application, gstreamer pipeline, etc
- Developed custom STM32H755 ARM Cortex M firmware using build, peripheral drivers (IMU, barometer, GPS, uSD card) and scheduler from ChibiOS/Ardupilot environment
- Prototyped debug system using Nucleo-H755 dev board and discrete peripheral boards
- Hardware bring-up/debug
- Schematic/circuit review
- **Tools:** C/C++, Python, Yocto/Bitbake, GCC ARM, Segger JLink/Ozone, GDB, Visual Code, EMACS/Global, Git, Bash, Beyond Compare, Docker, Github

### Beadedstream, Firmware Engineer, Anchorage, AK, 11/23-2/24

Beadedstream develops environmental data loggers and sensors

Worked on Nordic nRF52840 (ARM Cortex M4) based data logger running Zephyr OS

- Added new firmware features
- Debugged/fixed various problems with build, application firmware, Zephyr drivers/internals, Nordic HAL layer, hardware accessses
- Debugged hardware problems (datasheets/schematics, oscilloscope, rework)
- **Tools:** C/C++, GCC ARM, Git, Visual Code, EMACS/Global, Segger Jlink, Segger Ozone, GDB, Beyond Compare, Discord, Github

### **FreeFly Systems, Firmware Engineer, Woodinville, WA, 8/22-5/23**

FreeFly develops high quality digital video cameras and drones.

Worked on Xilinx Zynq Ultrascale+ MPSoC FPGA and NXP STM32 based 800FPS slow-motion camera

- Set up Petalinux (i.e. Yocto/Bitbake) build for Xilinx Zynq Ultrascale+ MPSoC FPGA containing four A53 ARM cores and two R5 ARM cores. Developed/verified unsupervised system for running Petalinux on A53 core(s) and bare-metal firmware on an R5, without using a hypervisor. This involved Linux kernel+device tree mods plus honor-system hardware partitioning. (Also evaluated OpenAMP, Xen hypervisor, and various other solutions)
- Developed shared memory communication system between Linux and bare-metal firmware, using 3rd party DMA buffer driver. Wrote device tree modifications and access API
- Wrote/modified/built: u-boot, Linux kernel, device drivers, rootfs, flashloader, 1st stage bootloader, application function libraries/APIs, and bare-metal firmware (A53, R5, and STM32), using modern C/C++.
- Debugged/fixed various problems with build, application software, U-Boot, library code, firmware, etc
- Modified FPGA's HDL using VHDL and Xilinx Vivado.
- **Tools:** C/C++, Yocto/Bitbake/Petalinux, Vitis, GCC ARM, Git, VHDL, Vivado, Visual Code, EMACS/Global, GDB, Beyond Compare, Slack, Bitbucket

### **Toyota, Firmware Engineer, Columbus, IN, 5/22-7/22**

Worked on an embedded Linux telematic device for Toyota forklift division

- Yocto/embedded Linux + U-Boot firmware development on NXP ARM iMX8 Cortex-A53, using modern C++, Docker container based build environment, and Azure DevOps (ADO) server
- Wrote OTA firmware update for Linux, which also updated firmware on two CortexMs and several CAN connected sensors. Interprocess communication used RedisPlusPlus messaging system.
- Debugged/fixed various problems with build, U-Boot, library code, etc
- **Tools:** C/C++, Yocto/Bitbake, GCC ARM, Git, Visual Code, EMACS/Global, UUU, GDB, Beyond Compare, MS Teams, ADO

### **MAPIR, Firmware Engineer, San Diego, CA, 3/21-4/22**

MAPIR makes a multi-camera ethernet connected array, used in drones

- Yocto/Linux/BSP + U-Boot firmware development on NXP ARM iMX6, using C/C++ and Python
- Worked with STM32F427 ARM bootloader and Ardupilot port, including initial debug
- Created/modified Docker based build environment
- Debugged/fixed various problems with build, drivers, Linux startup, U-Boot, etc
- Added commands+functionality to u-boot
- Implemented initial firmware programming and firmware update software
- Hardware bring-up/debug
- Schematic/circuit review
- **Tools:** C/C++, Python, Yocto/Bitbake, GCC ARM, Git, Bash, Wireshark, EMACS/Global, Beyond Compare, Slack

### **Gideon Health, Firmware Engineer, Austin, TX, 12/20-2/21**

- Wrote FreeRTOS based C firmware for a Dialog 14683 ARM Cortex M device
- Set up an automated/CI command line makefile to run on a Docker container on CircleCI. Build was makefile based using the GNU GCC ARM Embedded toolchain. Original build ran on Smart Snippets/Eclipse
- Developed command line tools for image generation/memory layout
- Wrote Python scripts to extract build information and configuration from logs
- **Tools:** C/C++, Python, GCC ARM, GDB, Git, Bash, Smart Snippets, Eclipse CDT, Beyond Compare, Slack

### **Apricorn, Firmware Engineer, Poway, CA, 6/20-8/20**

- Developed bare metal firmware for Maxim MAX32558 (ARM Cortex M) based secure/encrypted hard drive using Maxim's SDK. Code written in C using GCC ARM, debugged with GDB via JLink, etc Also reviewed schematics and modified hardware.
- Wrote Python scripts to generate flash patterns for flash erase/programming verification
- **Tools:** C/C++, Python, GCC ARM, GDB, Git, Eclipse CDT, Beyond Compare

**Sarcos Robotics, Firmware Engineer, Salt Lake City, UT, 7/19-1/20**

- Wrote bare metal ethernet bootloader for an NXP ARM Cortex M7 using lwIP TCP/IP stack and MCUXpresso SDK. Bootloader able to update application firmware in flash as well as the bootloader itself, over a TCP connection using custom memory mapping between OCRAM, DTCM, ITCM (RAM) and XIP flash. Bootloader also able to store/launch multiple applications, verify application integrity via SHA-256 or CRC-32, dump flash over ethernet, etc..
- Wrote application library to interface with bootloader, enforce flash map, read information blocks, etc...
- Wrote Python server scripts to download firmware and control bootloader via a remote TCP connection, control application via UDP. generate BIN from ELF file, etc.. Scripts used to verify operation, then later as a base for technician's server program.
- Development done on Linux using GCC ARM toolchain and CMake.
- Wrote documentation for bootloader and server scripts, as well as design proposals, communicated with NXP support engineers, etc...
- Debugged other various firmware, compiler, and hardware issues.
- **Tools:** C/C++, Python, Visual Studio Code, Beyond Compare, Git, Eclipse, GitLab

**Ventec Life Systems, Sr Firmware Engineer, Bothell, WA, 6/19-7/19**

- Debugged USB bus reset issue on QNX based embedded system, using a Beagle USB 480 protocol analyzer. Wrote shell scripts to aid in characterizing the problem and made recommendations for mitigation.
- Evaluated a multitude of C/C++ JSON libraries for use in porting project.. Library required to not use dynamic memory allocation or throw exceptions

**Radio Thermostat Company of America, Firmware Engineer, Modesto, CA, 2/19-3/19**

- Recover/recreate build of shipping Z-Wave thermostat source code from incomplete source
- Update Z-Wave firmware for ZM5202 to newest 6.81 Silicon Labs SDK and update security from S0 to S2

**EZLYNK, Software/Hardware Engineer, Cayman Islands, 2/18-1/19**

EZLYNK makes an automotive ECM/ECU programming/flashing device which stores/retrieves multiple engine profiles from cloud storage

- Reverse engineered compression and decompression algorithms and implemented in C#
- Reverse engineered decryption algorithms and implemented in C#
- Reverse engineered checksum and CRC algorithms using a variety of techniques including disassembly, interactive debugging, data analysis, etc and implemented in C#
- Reverse engineered/analyzed ECU firmware (SH2 and PPC assembler)
- Reverse engineered/analyzed PC/Windows applications using IDA, Reflector, OllyDbg
- Reverse engineered SPC564 based JTAG on ECU's PCB and prototyped a debugger controlled system
- Wrote firmware analysis tools using C# and C++
- **Tools:** C#, C++, MSFT Visual Studio, IDA, Reflector, OllyDbg, Exeinfo PE, DIE, dnSpy, ILSpy, ProcMon, Dumpbin, Beyond Compare, Git, EFILive, EcuFlash, RomRaider, RaceCon

**Voalte, Software Engineer, Sarasota, FL, 9/17-1/18**

Researched and developed an Alexa based Android/Linux hardware and software proof-of-concept for a medical device

- Researched and prototyped/tested various far-field microphone arrays and DSP engines on Android, Raspian/Raspberry Pi), and Linux
- Researched/built/modified various Java/Android libraries to login (LWA) and access Alexa Voice Services (Amazon's AVS interface/SDK is written in C++ and intended for stand-alone devices like the RPi)
- Research and assembled NDK module + Android library for custom wake-word recognition based on Snowboy...
- Wrote Android proof-of-concept app in Java to login to Amazon (LWA), recognize custom wake-word, interact with Alexa Voice Services (AVS), and map Amazon intents/actions returned from custom skills to Android intents, utilizing the two previously mentioned libraries
- **Tools:** Java, C++/C, Android Studio, NDK, GCC, Git

**Accomplo/Purigen Biosystems, Firmware Engineer, San Francisco, CA, 12/16-1/18**

Developed embedded Linux/Raspbian firmware for DNA processing machine

- Wrote/debugged C++ Qt5 Widget OpenGLS based touchscreen UI for Raspian/Raspberry Pi Model 3, dual build for RPi and Ubuntu/PC
- Wrote various custom widgets+libraries, including screen navigation stack, USB hot plug detection, button list, table display, CSV import, versioning, Python interface
- Wrote build, machine and test scripts in Python
- **Tools:** C++, Python, Qt Creator/Designer, GCC, GDB, Raspbian, Git, Emacs/GNU Global, Beyond Compare

**Apricorn, Firmware/Hardware Engineer, Poway, CA, 6/17**

Examined/reverse engineered SPI protocol for controlling secure/encrypted SATA bridge, using oscilloscope and logic analyzer

**Mystic Industries, Software Engineer, Everett, WA, 4/17-5/17**

- Prototyped an idea organizer/3D whiteboard cross-platform tool using PyQt + Python on Linux
- **Tools:** Python, Qt/pyQt, Qt Designer, Git

**Mystic Engineering, Firmware/Hardware Engineer, Everett, WA, 11/16-12/16**

Developed firmware and hardware for a programmable low cost solid-state relay and dimmer for Raspberry Pi (RPI), Arduino, and PC hobbyists

- Wrote Microchip PIC16F1454 and PIC16F876A firmware (originally prototyped using PIC16F876A) for low cost AC switch/dimmer with I2C and USB interfaces. Uses timer, edge detection, and USB/I2C interrupts
- Designed AC switching circuit hardware driven by PIC16F1454 (originally prototyped using PIC16F876A) Part selection, PCB layout, prototype construction, hardware bringup/debug
- Tested I2C functionality using RPi/Raspbian and Python
- Wrote libraries and sample code for RPi and Arduino
- **Tools:** C/PIC assembler, MPLAB X, XC8. OrCAD Capture/Layout, Emacs/GNU Global, Git

**Apricorn, Firmware/Hardware Engineer, Poway, CA, 9/15-6/16**

Various secure/encrypted USB3 portable hard drives + SSD drives, based on ARM Cortex M, PIC18 and PIC24

- Developed device driver and application firmware for Maxim MAX32550 (ARM Cortex M3) based next generation portable secure/encrypted USB3 hard drive with touchscreen, RFID two factor authentication via MAX66300, haptic feedback, BLE, and serial debug monitor. Code written in C using GCC ARM, debugged with GDB via JLink, etc Reviewed schematics, modified hardware and brought up/debugged/reworked hardware.
- Evaluated 10+ commercial and free embedded GUI libraries, selected uGFX for this project based on usability, performance, source code quality, and licensing terms. Wrote skeleton dialog stack mechanism (similar to Android, iOS, etc) for navigating up/down a stack of dialogs/screens and implemented various demos and feature tests
- Wrote Python scripts to generate C code arrays and/or files for various palettes, validate random number generator (via serial debug monitor) and test commands on the 66300 eval kit (via USB serial port)
- Modified firmware for several Microchip PIC18 based secure hard drives. Modifications included generalizing code to abstracting signal locations (makes it possible to run same code on different hardware platforms by editing a single header file), as well as adding new features. Code debugged via PICKIT3. Modified/reworked hardware.
- Reviewed/modified design for a next generation PIC24 based secure hard drive with LCD character display and RFID reader.
- **Tools:** C/C++, Python, Git, Emacs/GNU Global, Beyond Compare, GCC ARM, MPLABX, C18 compiler

**Mystic Industries, Software Engineer, Everett, WA, 7/16-9/16**

Wrote a variety of Python web automation/scraping apps using Qt, Mechanize, BeautifulSoup, Facebook-sdk on Linux

- Wrote a website creation tool with gathers crowd sourced links, identifies new links, and allows content creation with addition of article summary, modified title, optional humorous comment, and photo. Accepts links from my viewer, or via drag/drop from browser, etc
- Wrote a tool to automate recurring Facebook posts with mechanize (non-Graph, non-robot, etc), launched via cron job with random start delay (also wrote delay, see wolinslabs.com/blog/random.delay.html)
- Wrote tool to automate Facebook poke returns
- **Tools:** Python, Qt/pyQt, Qt Designer, BeautifulSoup, Mechanize, Emacs/GNU Global, Git

**Cortland Line, Firmware/Hardware Engineer, Cortland, NY, 5/15-7/15**

Industrial controller prototype for tapering fly fishing line, based on an Arduino Due (ARM Cortex M0)

- Designed interface circuitry for stepper motor, analog transducer, line position monitor, LCD, SD card, and pushbuttons
- Developed drivers + libraries in C++ for above peripherals as well as control algorithms, CSV import routines, and test/verification programs, using the Arduino framework where possible. Used interrupt-driven and timer driven methodology rather than blocking delays, wherever feasible. Worked around bugs in Arduino libraries.
- Wrote firmware for second Due used as a simulator/test fixture
- **Tools:** Arduino, C/C++, Git, Emacs/GNU Global, Beyond Compare

**FiftyThree, Firmware Engineer, Seattle, WA, 5/15-6/15**

Bluetooth (BLE) stylus for Apple iPad, based on TI CC2541 (8051 SoC)

- Debugged problem downloading firmware from internet server via Apple Mac, using code analysis and JTAG debugging
- **Tools:** Keil 8051 Embedded Workbench, C/C++, Git, Emacs/GNU Global, Beyond Compare

**Mystic Engineering, Hardware Engineer, Everett, WA, 2/15-4/15, 7/15-9/15**

Developed debugging tool containing USB serial port, RS232 voltage translation, two banks of 3.3V to 5V up/down selectable level shifters, debug LEDs and clocks.

- PCB design, layout, assembly and debug, through various prototype revisions
- **Tools:** OrCAD Capture/Layout, Git

**Mystic Engineering, Hardware/Kernel Engineer, Everett, WA, 12/14-2/15**

Developing a BCM2835 (Broadcom ARM), Linux based open source audio digital effect system and player/recorder.

- First prototype hardware based around Raspberry Pi model B+ with addition of codec eval board, guitar preamp circuit, and touchscreen
- Designed Maxim codec daughterboard which contains audio ADC/DAC, encoder/DSP, guitar preamp, input/output routing, etc. Assembled/debugged prototypes
- Wrote ALSA audio driver which outputs audio to GPIO pin based FM transmitter (based on PiFM project)
- Wrote Qt launcher/frontend to drive command line programs
- Evaluated custom netinst Raspbian rootfs vs Yocto/Poky based distro
- Evaluated multiple codecs and touchscreens, based on datasheet/specs and software support
- **Tools:** C/C++, GCC, Raspbian, Yocto, Bitbake, Qt, Git, Emacs/GNU Global, Beyond Compare, OrCAD Capture/Layout

**Adeneo Embedded, Linux Kernel Engineer/Consultant, Bellevue, WA, 9/14-12/14**

Worked with Freescale MX6 quad core ARM Cortex based custom automotive development board similar to Freescale Sabreauto. Yocto build environment used for bootloader, kernel, and rootfs development

- Embedded Linux BSP kernel + device driver development and 3rd party integration
- Custom distro/machine/rootfs development, 3rd party application integration
- Hardware bring-up/debug including reading schematics and rework
- Sysfs driver for TI430 RFID tag
- Third party driver integration
- **Tools:** C/C++, GCC, Yocto, Bitbake, Git, Emacs/GNU Global, Beyond Compare

**Mystic Engineering, Firmware/Hardware Engineer, Everett, WA, 11/11-8/12, 3/13-7/13, 10/13-2/14, 7/14-9/14**

Developed an STM32 ARM Cortex M4 based digital Theremin driving a software FM synthesizer+DSP

- Wrote/debugged ARM Cortex firmware using C/C++, which includes DSP, FM synthesis, digital effects, and curve fitting
- Designed ARM Cortex M4 based hardware to read performer hand position via capacitance to frequency conversion, with USB interface for configuration and debug, and virtual MIDI interface for musical instrument control.
- Prototyped/evaluated several different oscillator and mixer circuits for antenna proximity detection, as well as different firmware algorithms.
- Wrote Qt GUI test fixture to prototype various configurations of hardware controls (i.e. knobs, sliders) to calibrate instrument and test FM synth module. (Also evaluated Mono/C# and GTK+)
- Wrote Java/Swing POC prototype FM synthesizer modeled after Yamaha DX7/TX7. Ported back end of prototype to C++ library. Library can be compiled into ARM firmware or tested on PC via NDK
- Wrote Java/Swing POC using Eclipse to prototype various MIDI code sequence options for implementing continuous pitch variations
- Part selection, prototype construction, hardware bringup/debug, PCB layout
- **Tools:** C/C++, Bash, GCC, Git, Emacs/GNU Global, Source Insight, Beyond Compare, OrCAD Capture/Layout, Qt

**Protean Payment, Firmware Engineer - Consultant, Ann Arbor, MI, 6/14-7/14**

- Wrote SPI bootloader for Microchip PIC18 which allows reprogramming application image. Modified memory layout and low level startup. Bootloader also verifies image and makes a launch/no launch decision.
- Wrote STM32F4 ARM Cortex M4/F4 based downloader/test fixture to debug PIC18 bootloader, using C/C++ and GNU Tools for ARM Embedded Processors toolchain. Debugged with GDB/OpenOCD
- Wrote Linux/C++ command line program to read INHX8M/INHX32 formatted flash image, calculate a CRC32 and insert the CRC into the image along with various other parameters.
- **Tools:** C/C++, Perl, Bash, XC8, GCC, GDB, Git+Github, MPLABX, Emacs/GNU Global, Beyond Compare

### **Metronics Systems, Hardware Engineer - Consultant, Snoqualmie, WA, 3/14-5/14, 7/15**

Metronics Systems develops 3D laser scanners

- Wrote VHDL for a Xilinx FPGA using ISE, which controls a 3D laser scanner, reads HIRES quadrature encoder, coprocesses for a DSP, generates timing signals and safety interlocks, and is DSP programmable via serial link. New design uses encoder for more accurate position measurement.
- Simulated/verified VHDL design using ModelSim simulator
- Designed hardware modifications for existing PCB to accommodate new HIRES encoder and encoder based positioning, and updated schematics
- Tools: VHDL, ISE, OrCAD, Git+Bitbucket, Emacs, Beyond Compare

### **Qualnetics, Firmware/Linux Engineer - Consultant, Bellingham, WA, 8/13-9/13**

Worked with prototype automotive hardware based on ARM Cortex A9/Freescale iMX6

- Set up Linux builds for rootfs via LTIB, u-boot (standalone), and kernel (standalone)
- Modified 3.0.35 kernel and drivers, U-Boot bootloader, and Freescale Linux rootfs, start up inits, etc
- Evaluated MX6 Q7 iWave and BD-SL-iMX6 Boundary Devices boards
- Tools: C/C++, Bash, GCC, SVN, Emacs/GNU Global, Source Insight, Beyond Compare

### **Mystic Engineering, Firmware/Hardware Engineer, Everett, WA, 3/13-4/13**

Developed an ARM Cortex M3 based dual USB serial port PCB designed to aid in debugging other boards

- Designed PCB using STMicroelectronics STM32F102 (ARM Cortex M3) with multiple serial ports, A/D, I2C, and pulse outputs, designed to assist in debugging other projects.
- Performed low-level/hands-on bring-up using scope, logic analyzer, etc..
- Wrote firmware for STM32F4 + STM32F102 board on Linux workstation, using GNU Tools for ARM Embedded Processors (GNU/GCC/ARM toolchain) in C/C++, debugged using GDB
- Wrote article describing how to use open source tools to program/debug the STM32F4, online at [www.wolinlabs.com/blog](http://www.wolinlabs.com/blog)
- Wrote article open sourcing single USB virtual COM port, online at [www.wolinlabs.com/blog](http://www.wolinlabs.com/blog)
- Tools: C/C++, GCC, GDB, Git+Github, Emacs/GNU Global, Beyond Compare, OrCAD Capture/Layout, gEDA

### **Nebula, Software Engineer - Consultant, Redmond, WA, 8/12-3/13**

Adapted Linux BSP for Freescale MX6 based ARM Cortex A8 board which serves as a secure controller within a 2U secure data center server.

- Modified 3.0.15+3.0.35 kernel and drivers, U-Boot bootloader, and Ubuntu Core 12.04 and Ubuntu Server 12.04 rootfs, upstart start up inits, libraries, etc to work with ARM Cortex A9 based board
- Evaluated iMX6 MXM and Q7 boards, and MX53 boards from iWave, including debugging software and hardware problems using oscilloscope, reworking PCBs with microscope, etc.
- Wrote library to access GPIO input and output signals via sysfs interface, blocking a thread waiting for input rising/falling edge, etc. Chose GPIO signals by reading schematics and datasheets.
- Modified serial port/tty kernel driver to add additional serial ports
- Debugged/modified SATA kernel driver
- Modified U-boot SATA driver to work around a hardware problem
- Wrote WDT driver for U-Boot, getting technical info from MX6 reference and kernel source
- Wrote USB bootloader for PIC24 and USB downloader for ARM/Linux which allows ARM to reprogram PIC24. Modified memory layout and low level startup. Bootloader also verifies image and makes a launch/no launch decision.
- Wrote serial bootloader for PIC18 and serial downloader for ARM/Linux which allows ARM to reprogram PIC18. Modified memory layout and low level startup. Bootloader also verifies image and makes a launch/no launch decision.
- Wrote Linux/C++ program to read INHX32 formatted flash image, calculate a CRC and insert the CRC into the image along with various other parameters.
- Built debugging applications in C and Bash shell scripts to run on Ubuntu Server + Ubuntu Core
- Debugged custom ARM Cortex A9 based hardware designed by Nebula, using schematics and hand-on techniques involving oscilloscopes. Designed/built debounced switch for debugging/testing.
- Tools: C/C++, Bash, GCC, GDB, Git+Github, Emacs/GNU Global, Source Insight, Beyond Compare

**Mobile Integration Workgroup, Software Engineer - Consultant, Bellevue, WA, 12/11-7/12**

Adapted Linux BSP for MIW's 16 ARM (ARMBlock16) rack mounted Linux server, which utilizes the Marvell Armada 88AP510. Motherboard design loosely based on the Marvell Dove.

- Debugged/modified 2.6.32 and 3.0.11 kernel and drivers
- Debugged/modified U-Boot bootloader
- Debugged/modified Ubuntu Core 12.04+11.10, Ubuntu Server 12.04+11.10 and Debian 6.0.4 rootfs start ups, libraries, etc to work with ARMBlock
- Added PXE boot functionality to U-Boot. Set up PXE server on Debian for testing.
- Extracted, modified, and rebuilt initrd to program SPI flash from multiple image choices on a PXE server.
- Ported Hadoop Ant build to ARM, using a QEMU/ARM/Ubuntu Core 12.04/user mode virtual machine I built.
- Set up source control on BitBucket using Mercurial

**Silicon Mechanics, Systems Engineer - Consultant, Bothell, WA, 3/12-4/12**

Solved various emergency problems for Linux/RHEL/CentOS server installs on Microsystems/x86-64 multicore Xeon servers, which generally involved low level debug, modifying/patching drivers and kernel, tweaking BIOS settings, updating drivers, etc.

**Yankee Environmental Systems, Software Engineer - Consultant, Turners Falls, MA, 11/11-12/11**

Wrote a VHDL implementation of a Viterbi communications processor for a dropsconde receiver/monitor, using a Xilinx Spartan 3AN. Simulated/verified model.

**Centri Technology, Software Engineer - Consultant, Seattle, WA 8/11-10/11**

Wrote multithreaded asynchronous socket server for iPhone in Objective-C/C++ which acts as an HTTP proxy, modifying HTTP requests and responses (chunked and non-chunked encoding), and providing a compressed data path between iPhone and Centri's server/router over 3G, to conserve cellular bandwidth. iPhone server ran in background on iOS, using Grand Central Dispatch (GCD) for high performance, and redirected all inbound and outbound HTTP traffic through multiple paired socket connections. Server designed to demonstrate Centri's compression algorithm to prospective clients. Demo exhibited at 4G World 2011 show by Symantec. Used netcat for a variety of debugging tasks.

**Discovery Bay Games, Software/Firmware/Hardware Engineer - Consultant, Seattle, WA 5/11-1/12**

Discovery Bay Games develops iPad games and associated game controller hardware. Project was a communications bridge between a game controller and an iPad which used a piezo electric element to modulate a sound carrier in the superaudible range, to transmit data from the controller to the iPad

- Wrote iPad software in Objective-C/C++ which captured continuous audio from iPad's microphone and did realtime digital signal processing to extract digital bitstream from audio. Designed digital resonator (bandpass) filters to eliminate background noise.
- Packed iPad software into a library that can be used by third party developers to write games for DBG's hardware
- Wrote firmware in C for the Atmel AVR based communication module. Code implemented a serial over audio protocol using ASK/OOK modulation in the inaudible range.

Second generation of the product utilized a direct connection into the iPad mic input, to achieve a higher data rate

- Designed analog hardware interface from IO to iPad mic input
- Modified iPad software in Objective-C/C++ which captured continuous audio from iPad's microphone to process a pseudo higher speed digital waveform
- Packed iPad software into a library that can be used by third party developers to write games for DBG's hardware
- Wrote firmware in C for the Atmel AVR based communication module implementing a pseudo digital waveform
- Wrote Java software to generate a digital signals in a WAV file

## Mystic Industries, Software/Firmware/Hardware Engineer, Everett, WA 10/09-11/11

**Wrote a series of Android applications** (listed below) using Java, Eclipse, and Ubuntu Linux. For each application, wrote all code, organized/ran beta test, wrote help, generated website/graphics/screenshots, and handled Android Market submission. Kept all applications updated, fixing reported bugs and adding features, dealt directly with users email.

- Wrote Cricket Scorepad based on existing iPhone product. Generates/stores/recalls scorepads for the darts game of Cricket. Added capability to score 3 player (cutthroat) and 4 player games. Also wrote free ad supported version
- Wrote Super Scorepad based on existing iPhone product, with enhanced features. Generates/stores/recalls scorepads for any number of players for any game where scoring involves adding/subtracting columns of numbers. Larger scorepads scroll both horizontally and vertically using drag gesture (custom.) Pinch gesture shrinks/enlarges font (custom.) Also wrote free ad supported version.

**Wrote a series of iPhone+iPad applications** (listed below) in Objective-C, C++, Cocoa and OpenGL. For each application, wrote all code, organized/ran beta test, wrote help, generated website/graphics/screenshots, shot/edited promotional video and handled Apple provisioning/submission. Kept all applications updated, fixing reported bugs and adding features, dealt directly with users email.

- Wrote Pentomino Smackdown iPad only using OpenGL+lighting. 3D touchable tabletop two player spatial relations challenge based on the classic pentominoes puzzle. Designed to be played head to head with iPad between two players on a tabletop, like an actual board game. (Can also be played in a side to side mode, like 'normal' games.) [www.wolinlabs.com/pentosmack](http://www.wolinlabs.com/pentosmack)
- Wrote Mystic Puzzle Stick iPhone+iPad (universal binary) using OpenGL. 3D rotatable, touch manipulable 4 cube logic puzzle. [www.wolinlabs.com/puzzlestick](http://www.wolinlabs.com/puzzlestick)
- Wrote Black to White iPhone+iPad (universal binary) using OpenGL+lighting. 3D rotatable, touch manipulable puzzle made to render like a tabletop game. [www.wolinlabs.com/black2white](http://www.wolinlabs.com/black2white)
- Wrote Cricket Scorepad iPhone+iPad (universal binary) Generates/stores/recalls scorepads for the darts game of Cricket. [www.wolinlabs.com/cricketpad](http://www.wolinlabs.com/cricketpad)
- Wrote Super Scorepad iPhone+iPad (universal binary) Generates/stores/recalls scorepads for any number of players for any game where scoring involves adding/subtracting columns of numbers. [www.wolinlabs.com/scorepad](http://www.wolinlabs.com/scorepad) Also wrote free, iAd supported trial version of the same program.
- Wrote Space Castle 3D iPhone+iPad (separate products) using OpenGL based 3D Space Shooter arcade game, inspired by the 80s classic, Star Castle. Two control modes for ship, emulating a DPAD. [www.wolinlabs.com/spacecastle](http://www.wolinlabs.com/spacecastle)
- Wrote Mystic Photo Cube iPhone+iPad (universal binary) using OpenGL. Allows user to create a "virtual plastic photo cube" with the own photos, then manipulate the cube using intuitive hand gestures. [www.wolinlabs.com/photocube](http://www.wolinlabs.com/photocube)
- Wrote 3D Slider Puzzle iPhone+iPad (separate products) using OpenGL. 3D solvable version of the classic slider puzzle, manipulated using natural/intuitive gestures. [www.wolinlabs.com/3dslider](http://www.wolinlabs.com/3dslider)

## **Windows/OSX**

- Wrote application for musician's ear training, which plays a random sequence of notes in a given key and scale mode which musician tries to mimic and also plays multiple octave up/down scales in any key/scale mode/range. Written using C++/wxWidgets, for Win32 and Mac OSX. Also wrote a 30 day trial version licensing scheme and key generator, wrote help files, and set up a website to interface w/ PayPal for purchases - [www.mysticind.com/products/riffmaster](http://www.mysticind.com/products/riffmaster)
- Ported consultant timer multiple stopwatch C++ program from MFC/Win32 to wxWidgets, then added new features.

## **Hardware Development**

- Circuit design, schematic capture, and PCB layout for adapter board which converts a PC/ATX power supply into a bench supply. Specified/ordered components, assembled prototypes, designed website. [www.mysticengineering.com/atx.adapter](http://www.mysticengineering.com/atx.adapter)
- Working on circuit design for Analog Devices ADuC7024 (ARM7) with Spartan 3E FPGA + MIDI + USB. Will eventually write C/C++ firmware and VHDL for several different projects.

## **IT**

- Set up FreeBSD 7.2 NFS + SAMBA file server, IPP print server, SVN source control server, which also runs DNS + DHCP, and a software geom RAID.



**Clausius Technologies, Firmware Engineer - Consultant, Reno, NV 4/11, 6/11-7/11**

Wrote firmware in C for Freescale Coldfire based green energy device. Set up the build using CodeWarrior/Eclipse, generated build targets, memory layouts, linker files, and debug configurations for builds to run/debug from external flash and external RAM.

**Alpha Technologies, Firmware Engineer - Consultant, Bellingham, WA 8/10-1/11**

Wrote firmware for a Freescale Coldfire based high current, high availability power supply in C/assembler.

- Wrote low level hardware configuration and device driver firmware for XM3 board, including clocks, timers, PWMs, RS232, IO using Processor Expert.
- Wrote LCD/pushbutton based menu system for XM3 as an extensible framework which simplifies future modification and additions
- Wrote bootloader for XM3 inverter board which does low level system h/w configuration and memory layout, and allows reprogramming the flash based application code via RS232 or via image loaded into upper half of Coldfire flash. Performs CRC verification of application code at startup, prevents unit from running corrupt code.
- Wrote bootloader for Apps board which does low level system h/w configuration and memory layout, allows reprogramming from external SPI flash. Also performs CRC verification.
- Wrote C++/Win32 PC application which modifies an S-record image by adding a 32 bit CRC to the file (used with bootloaders). Incorporated this program into the Coldfire/CodeWarrior build process, so it happens automatically on each build.
- Wrote special PWM delayed pulse driver which could be used to add short interval space to the FET turn off/turn on.
- Hands on low level bringup/debug using schematics, oscilloscope and logic analyzer.
- Performed microscope SMD rework in conjunction with debug.

**BSquare, Sr Software Engineer - Consultant, Bellevue, WA 6/08-9/09**

**Auto entertainment/navigation system** - based on Freescale MX51 and WinCE/MSAuto Bristol (Ford Motor Company)

- Wrote MSAuto (Talladega+Bristol) NAND flash driver, based on three different drivers and datasheets
- Wrote MSAuto Bristol 1394/Firewire driver for Fujitsu MB88388A supporting IP over 1394, audio, and video streaming
- Wrote 1394 speed benchmark app, EOL tests for audio over 1394, and IP over 1394.
- Designed MMS hardware mod which reroutes 1394 audio channels on test fixture board, as part of the EOL fixture.
- Wrote parts of bootloader
- Wrote various diagnostics and 1394 EOL tests,
- Hands on hardware debug, hardware design modification, and microscope rework.

Ethernet audio router, based on Freescale MX31 (Microsoft)

- Wrote LED driver
- Modified KITL code to support both active (interrupt driven) as well as passive (polled) modes
- Modified KITL code to support USB KITL as well as ethernet KITL

**Mystic Industries, Firmware/Electrical Engineer, Everett, WA 3/09-11/09 (off+on, part time)**

- Designed hardware for a Microchip PIC 16F877A based altitude display for a MiG-29. Displays altitude in 10 ft increments based on encoder standard pressure RS232 altitude stream and corrects for barometric pressure based on rotary encoder input.
- Wrote/debugged firmware for altitude display using C/HiTech.
- PCB layout
- Assembled and tested prototypes
- Updated hardware and firmware design for 16F887 PIC

**Microsoft, Software Engineer - Consultant, Redmond, WA 3/08-5/08**

- Fixed bugs and added features for Mobile Outlook in WinCE/WinMobile 7.0 release using C++.
- Built whole CE platform images, including entire operating system
- Wrote Perl tools for monitoring WinCE platform build

**YES Inc, Firmware/Hardware Engineer - Consultant, Turners Falls, MA 4/04-12/07**

**CEILOMETER/LIDAR** - a LIDAR is a weather instrument that measures wind speed & direction at various altitude profiles using a laser, while a ceilometer measures cloud ceilings using a laser. The digital electronics for both instruments are very similar.

- Designed Atmel ARM7 based system with dual channel 12bit A/D, 16M SRAM, Altera CPLD, USB, RS232, and pushbutton/LCD user interface, performed PCB layout, some assembly and debug.
- Wrote co-operative tasking operating system and device drivers in C/Rowley Crossworks for ARM7 to sample to SRAM and communicate w/ PC via USB.
- Wrote C++ Windows application to communicate with and drive the ARM7 system.
- Designed Xilinx FPGA daughter card with 32M SDRAM and high speed A/D interface, PC/104 interface, pushbuttons and LCD interface, and general purpose I/O for a COTS ARM9 SBC, performed PCB layout, some assembly and debug.
- Wrote FPGA VHDL for dual SDRAM controller, 85Mhz dual channel A/D interface, input FIFO, PC/104 register-based communication interface, and DCM based clock generator.
- Wrote C++/GCC Linux drivers for FPGA based PC/104 devices
- Wrote C++/GCC Linux based system monitor for manipulating system for verification and testing.

**SODAR** - a miniature, low cost weather instrument which measures wind speed & direction at various altitude profiles using acoustic energy

- Designed high gain multistage microphone preamp circuit with 4th order op-amp bandpass filter, performed PCB layout, assembled, and debugged board.
- Designed piezo speaker array around PIC16F876 microcontroller based row and column phased beam-steering, performed PCB layout, assembled, and debugged board.
- Designed A/D and DSP based processing board
- Wrote firmware in C/assembler for PIC16F876 to accomplish beam steering by phasing 8 channels.
- Developed a proof-of-concept prototype around an office class PC running Debian Sarge (Linux) with a 2.4.24 custom kernel. PC used 8 channel pro audio sampler and a combination of open source and my custom written application software to experiment with various sensor configurations and geometries. Prototype performed filtering and FFT frequency determination using various DSP techniques with Linux software written in Perl and C++

**Microsoft , Firmware/Software Engineer - Consultant, Redmond, WA 2/04-5/04**

- Wrote firmware in C for a laptop auxiliary LCD display and input device based on the Cirrus EP7312 (ARM7) using IAR tools.
- Wrote C++/Win32 library to interface with USB bridge "network cable" for Longhorn migration

**Tex-Cal, Hardware/Firmware Engineer - Consultant, Beaumont, TX 1/04-4/04**

- Designed a portable inventory control computer prototype based around the Cirrus EP9315 (ARM9) processor.
- Built custom Linux kernel for Cirrus EP9301 using the GCC ARM9 port.
- Evaluated several other microprocessors for this project, including a Cirrus EP7312, NeoMagic NMS7210 and VG240 (x86)

**Keynote Systems, Electrical/Firmware/Software Engineer - Consultant, Seattle, WA 11/03-2/04**

Keynote's project involved reverse engineering the LCD controller and keyboard protocols of a Blackberry cell phone/PDA, then building an FPGA circuit to allow a PC to control the phone's keyboard and read the display back into the PC.

- Using a logic analyzer and oscilloscope, reverse engineering the LCD controller connections, control signals and communications protocol of RIM Blackberry Cell Phone/PDA.
- Wrote VHDL on Xilinx Spartan FPGA to capture LCD controller command and pixel data, compress data, and send data to PC via USB. VHDL also receives USB commands from PC and translates them to Blackberry keyboard presses.
- Perl programs to 1) dump Agilent 54622D Analyzer waveform data, 2) filter the data into LCD command and pixel writes, 3) render the command and pixel writes into a Windows bitmap representing the PDA screen
- Wrote Java program on Blackberry to perform graphic writes on keyboard command, such as pixel at x,y location, rectangle of width, height at x,y location, etc.

**Metron Systems, Electrical/Software Engineer - Consultant, Snoqualmie, WA 8/03-10/03**

- Designed/simulated/debugged Xilinx FPGA VHDL to perform image processing on camera data stream and glue logic in 3D laser scanner real-time embedded system
- Wrote C++ Win32/MFC/OpenGL based multi-threaded data acquisition/viewer application. Data acquisition receives control commands via TCP and data packets via UDP from single or multi-headed 3D laser scanner system. Viewer displays these points as acquired, or recalled from a saved file. Viewer rotates/zooms/pans 3D model using OpenGL, using a "virtual trackball."
- Wrote Perl programs to simulate data from a single or multi-headed 3D laser scanner, and to analyze saved binary file's structure.

**Hudson Valley Music, Electrical/Firmware Engineer - Consultant, Highland, NY 6/03-8/03**

- Designed a digital MIDI metronome based on PIC16F628, using OrCAD Capture. MIDI (musical instrument digital interface) metronome keeps time using a series of LEDs.
- Wrote/debugged PIC16F628 firmware in C/ assembler
- Design PCB layout for MIDI metronome using OrCAD Layout

**SunStar Timers, Electrical Engineer - Consultant, Woodinville, WA 5/03-8/03**

- Designed h/w for wireless and sound modification to base timer circuit
- Designed h/w for wireless and sound capable remote switch
- Reverse engineered schematics from Gerber board layout files for an 8096 based suntan bed timer system and created new schematics with OrCAD capture
- Modified reset and power supply circuitry to increase reliability
- Wrote Perl scripts on Linux to manipulate/sort/condense netlists and other schematic information.

**YES Inc, Software Engineer - Consultant, Turners Falls, MA 1/02-4/03**

Wrote software for a real-time embedded Linux based Intel PC LIDAR project. A LIDAR is a weather instrument that measures air turbulence at various altitudes using a laser. All software written on Linux using GCC C/C++.

- Wrote Linux application software for data acquisition, display and analysis.
- Wrote Linux device driver for a high-speed data acquisition card with onboard DSP. Device driver downloaded DSP firmware as well as read processed data packets.

**Myotronics, Software Engineer - Consultant, Tukwila, WA 12/02**

- Wrote Win32 test program to connect and performance test the USB/HID interface of an 8051 based dental device.
- Consulted with client on various other options to increase performance, including writing a custom WDM USB buffering device driver.

**Sunfire, Firmware/Hardware Engineer - Consultant, Snohomish, WA 2/01-6/02, 12/02-2/03**

- Wrote/debugged 68332 based firmware for a high end home theater audio/video processor. System included a front panel with knobs, buttons and VFD display, STV5730 OSD (On Screen Display), AM/FM tuner, Motorola 56366 DSP, multiple IrDA remote controls, volume controls, flash, Xilinx PLD, and nvm serial eeprom. All devices accessed via SPI using external decoding and bank selection. System capable of firmware download via serial port or data encoded within a CD audio sample. Wrote code in SDS C/assembler
- Wrote/debugged 68332 based operating system using a cooperative time based threading/tasking system
- Wrote/debugged a text based, table driven menuing system
- Wrote/debugged a dialog based, object oriented UI system, similar in function to Windows dialog manager
- Brought up and debugged system hardware using 68332 BDM and logic analyzer/scope.
- Debugged and corrected hardware problems.
- Performed surface mount prototype rework using Metcal and microscope to test hardware fixes.

**YES Inc, Software Engineer - Consultant, Turners Falls, MA 6/02-10/02**

- Wrote Linux application software and device drivers for an 80486 PC104 based wireless packet sniffer.
- Created a custom embedded Linux distribution starting from small disk space, small memory footprint GPL'd distribution
- Created custom kernel for small memory footprint.

**E\*swing!, Firmware/Hardware Consultant/Mentor, Seattle, WA 2/02-4/02**

Consulted with developer of a golf swing analyzer which uses infrared and ultrasound sensors to measure club angle, velocity, and track, to predict drive distance.

- Debugged PIC16F877 assembler and Microsoft VB application
- Debugged and corrected digital and analog hardware problems, include infrared and ultrasound sensor circuits.
- Performed prototype rework.

**WD Machines, Firmware/Hardware Consultant, Kirkland, WA 11/01-12/01**

- Wrote/debugged PIC 16F876 firmware for handheld wireless device used to measure relative or absolute angular displacement in a patient. Developed a simple, robust wireless protocol based on CPCA modulation.
- Wrote/debugged PIC 16F873 firmware for wireless receiver based on the Linx receiver which received messages from the receiver and passed them on to a host PC. The receiver performed error checking on the received messages and was also able to respond to a variety of configuration commands from the host PC.
- Designed, brought up and debugged receiver prototype hardware using PIC ICD and logic analyzer/scope.
- Laid out receiver PCB using OrCAD layout.
- Debugged and corrected hardware problems.
- All firmware written in C and assembler using HiTech ANSI C compiler

**MicroEncoder/Mitutoyo, Hardware Consultant, Kirkland, WA 5/01**

Design review for 100Mhz DSP/PIC 16F877 and Altera FPGA based 2D subnanometer precision measuring system.

**Lucks Co, Firmware/Hardware Consultant, Seattle, WA 2/01**

- Wrote/debugged firmware for a PIC 16F877 based oven controller using HiTech C. System monitored multiple thermocouples using onboard A/D and external analog mux arrangement. System communicated with other modules via CAN controller.
- Brought up and debugged hardware, using MPLAB, Microchip ICD setup, and scope. Made rework modifications to prototype.

**MicroEncoder/Mitutoyo, Firmware/Hardware Consultant, Kirkland, WA 8/00-12/00**

- Designed two Altera 7K series FPGAs and redesigned OrCAD board schematics for a 100Mhz DSP/PIC dual processor measurement system using a CMOS camera and dual ported SRAMs. Application is a real-time embedded subnanometer precision measuring system which samples images from a CMOS camera array/pulsed laser combination in real time at 30fps and generates precision measurements through image correlation.
- Wrote/debugged DSP firmware in TMS320C5402 assembler and PIC 16F877 firmware in HiTech C/MPLAB for above system.
- Performed board bringup/debug, made design modifications, and did rework to prototype using microscopic soldering techniques. Solved cable ringing problems for laser readhead assembly, reset and other issues..
- Redesigned measurement system for smaller PCB layout. Performed initial board bringup/debug.
- Wrote Win32 Console and MFC serial programs for testing.

**DECA Design, Software Consultant, Manchester, NH 6/00 - 8/00**

Wrote software for an embedded Linux, StrongARM based touch-screen tablet device with wireless comm used by major package delivery service. All code written on Linux in C/C++ and StrongARM assembler.

- Adapted touch screen device driver to specific hardware and compensated for hardware shortcomings.
- Wrote device driver for proprietary wireless comm device.

**DBC Avionics, Firmware/Hardware Consultant, Everett, WA 7/00-8/00**

- Designed low cost hardware using OrCAD and constructed prototype for an embedded altitude encoder tester/display. Prototype read gray code from aircraft altitude encoder and displayed altitude on a 4 digit seven segment display.
- Wrote firmware in PIC16C62A assembler

**Microsoft, Firmware/Hardware Consultant, Redmond, WA 2/00-5/00**

- Member of Xbox peripheral development team
- Designed/built prototype 8M flash memory card with high speed USB interface around Anchor EZ-USB development board.
- Designed/wrote firmware using Keil 8051 C compiler. Firmware able to dynamically adapt to 8M-32M flashes. Firmware responds to SCSI command subset (RBC) to making flash card a USB Mass Storage class Bulk-Only device. Prototype functions as a removable drive with Win2000 built-in USB/SCSI/filesystem drivers, able to be formatted, read, written, etc...
- Evaluated various USB micros for memory card, including Anchor EZ-USB, ScanLogic SL11R, and Atmel AVR. Wrote various speed/architecture/USB bandwidth use comparison white papers.
- Wrote Win32 console program to drive PVG120604 LCD controller with PC parallel port. Program displayed monochrome bitmaps on LCD.

## SeaMED, Firmware/Hardware Consultant, Redmond, WA 2/98-1/00

SeaMED is a contract designer and manufacturer of medical embedded system devices, practicing ISO9000 compliant design processes.

### **Heart Pump Monitor**

- Using Cosmic C and 68HC11 assembler, designed and wrote object oriented coprocessor code for a 68HC11E9 in a heart pump monitor. 68HC11 code was heavily interrupt service routine based and used the PA, SCI, SPI, and RTI interrupts. At times C code was debugged in 68HC11 assembler using a 68HC11 EVB with the BUFFALO monitor. 68HC11 displayed blood flow rate and motor RPM on a 7 segment display. Flow is obtained via a serial link to a Transonic flow board, while motor RPM was measured using the pulse accumulator. In addition, the SPI was used as a com link to the 68332 processor, to enable the independent processors to cross-check each other.
- Wrote Software Design Description document
- Prototyped several small circuits around a 68HC11 EVB to be able to start firmware before hardware was available.
- Initial bringup and debug of target hardware.
- Using Win32 and MFC, designed and wrote a flow board simulator. Simulator uses a PC serial port and impersonates a Transonic flow board. Simulator also allow setting flow values as well as injecting faults in the responses, for testing.

### **Home Patient Monitoring Station**

- Designed and wrote display and DSP device drivers for a WindowsCE embedded PC home patient monitoring station based on the MediaGX chipset using C++. Some prototype work for these drivers was done in a test environment developed with the DJGPP port of GCC.
- Wrote Software Design Description documents
- Using C++, designed and wrote background packet based sound record/play system for WinNT based on top of the Direct Sound COM interface and encapsulated in a DLL. System compressed sound for transmission over telephone lines, using ADPCM. Sound was also conditioned with a digital filter and interpolated in some cases.

### **Heart Muscle Drill**

- Wrote motion safety lockout firmware in PIC16C62A assembler using MPASM for a medical device used for drilling holes in heart muscle.
- Wrote Software Design Description document
- Prototyped a test circuit and did some hardware debugging.

### **Coronary Rotary Oblation Remote Display**

- Wrote calibration firmware for 68332 embedded system in a coronary rotary ablation machine remote display.

### **Prostate Oblation Machine**

- Wrote video driver firmware for 68332 embedded system in prostate ablation machine. Video driver initialized and controlled an SMOS SED1353 graphics controller and custom LCD panel, implementing bitblts to display text and graphics at the lowest level. Routines were originally written and optimized in C, then additional optimizations were made in 68332 assembler. Code was written with conditional compilation to allow compiling and testing on a PC (with an SED1353 evaluation board and external LCD panel) or compiling directly into the 68332 embedded system.
- Wrote a host of Win32 MFC and console applications to aid in development, including: application to rasterize True Type fonts to any resolution and save as Windows BMP, program to convert Windows BMPs to custom firmware BMP format, application to simulate LCD panel display (allowing another engineer to lay out user interface screens before real hardware was available), and application to test video driver on Win95 with a SED1353 evaluation board.
- Assisted in making circuit modifications and debug to add SED1353 and LCD panel into original prostate ablation embedded system.

### **Misc**

- Wrote realtime firmware and debugged hardware for a PIC16C71 embedded system handheld accelerometer (G meter.) Also debugged circuit design and reworked prototype hardware. Firmware written in PIC assembler
- Designed and prototyped hardware and wrote firmware for a PIC16C62 based digital cuckoo clock. Firmware written in PIC assembler
- Wrote Win32 application to intercept FKeys and insert macros into the keyboard buffer. Familiarity with Microtek C compiler, Cosmic C compiler, MPASM, MPLAB, TEKScope, digital oscilloscope

**Allied Signal/Avionics, Software Consultant, Redmond, WA 2/97-10/97**

Wrote a diagnostic and maintenance program for the Enhanced Ground Proximity Warning System (EGPWS) avionics embedded system. This Win95/WinNT program was written using C++/MFC. The application communicates with the EGPWS over RS232 using a totally asynchronous protocol and has both a terminal and continuous blast data display mode. Custom controls were written for these modes.

**Traveling Software, Software Consultant, Bothell, WA 7/96-1/97**

- Using C++ and OOP techniques, wrote the communications engine for a high performance, multithreaded, multiuser, socket based email acceleration server running on Windows NT for Microsoft Exchange enhancement. Server used Winsock sockets in conjunction with I/O completion ports, mutexes, and event thread synchronization objects to achieve a totally asynchronous communications server with a reasonable thread count. Server communicated with clients via a propriety protocol of my design and talked to Exchange via MAPI.

**Emulex, Software Consultant, Bellevue, WA 2/96-6/96**

- Wrote a Win95 VxD implementation of DEC's LAT protocol over Novell's IPX, providing a file I/O interface to work as a network transport for Reflections, Kea!, SmartTerm, and Kermit 95 terminal emulators. File I/O interface emulated Meridian's SuperLAT specification.
- Wrote a spy debugging VxD (LATSPY) which intercepts and displays commands intended for a VxD's device control procedure, before and after processing. VxD communicated with a small console application which displayed the LAT command calls in a window and optionally wrote them to a file.
- Wrote a small status display panel for LAT VxD parameter display and configuration, using MFC and VC++.
- VxDs were written in C using Vireo's VtoolsD and debugged with NuMega's SoftIce.

**Microsoft, Software Consultant, Redmond, WA 5/95-10/95**

- Developed parts of multithreaded fax gateway for Exchange Server (newest Email/Messaging system.) including a general interprocess communication object, preprocessor, and modem provider DLL. Product compilable on Win95, Win NT, and Win16. Experience with Extended MAPI, OLE2, Exchange, Unicode, SLM and localization techniques.
- Wrote C++ extractor/import DLL to decrypt and import Lotus cc:Mail post-offices into Microsoft Exchange Server (without a spec for Lotus' file format.)
- Designed UUCP gateway for Microsoft Exchange

**Starwave, Software Consultant, Bellevue, WA 3/95-5/95**

- Using Windows NT, VC++/32, Winsock, and TCP/IP, designed a distributed, dynamically scalable, networked server cluster for a multiplayer interactive network children's game which was a next generation graphical MUD. The option to port to Solaris was a design goal. Used object oriented design.
- Wrote asynchronous socket SMTP and POP3 mail client objects which will be integrated into the game client's mail system.

**Sun Microsystems/PC Networking Applications, Software Consultant, Chelmsford, MA 7/94-1/95**

Wrote a Win32s news reader client which used NNTP protocol (RFC 977) with XOVER extensions over Winsock/TCP/IP to retrieve USENET news. The application was written in a cockpit (client) and engine (server) architecture. The engine handled all news acquisition and processing while the cockpit presented the GUI. The engine was a DLL supporting my NewsReader API. The engine contained a CNewsServer object wrapped with a C exported interface which could be reused with other "cockpits" written in C++, C, or Visual Basic. Code written in C++/MFC (using VC++ for Windows NT) and developed on Daytona and WinNT 3.1. Built and set up an INN Server on FreeBSD1.1 to aid development and testing.

**Sun Microsystems/PC Networking Systems, Software Consultant, Chelmsford, MA 10/93-6/94**

(Note: All windows programs were written with Visual C++ and MFC. Wherever possible, development was done on Windows NT, using the Win16 subsystem. I designed my objects/classes with reusability in mind, and all programs shared many common classes. All programs used the Winsock interface, with the exception of Netstat)

- Wrote a Win 3.1/WFW version of a Unix style LPD (Line Printer Daemon) Print Server. Printer Server resides between TCP/IP port and Print Manager, spooling jobs (LPR), deleting jobs (LPRM), and providing queue status (LPQ). Because Printman has no interface to obtain queue contents/status, developed a DLL to notify LPD of queue modifications. If Printman crashes, LPD is able to requeue any remaining jobs at reboot.
- Wrote Windows Syslogd server which receives system error via a UDP socket and routes them to a display window, a file, and/or a udp socket on another host. Mimics a unix syslogd.
- Wrote Windows Netstat client similar to Unix netstat. Program displays statistics about network interfaces, memory usage, protocols, routing, sockets, and arp table and entries. Developed custom text display object and generic system/method for converting DOS apps to real Windows apps (nicer than QuickWin.)
- Wrote Windows RSH/REXEC client which executes commands on a remote server. Uses asynchronous sockets and allows additional input for commands that require it. (i.e. csh, mail, etc)
- Wrote Windows Finger client which retrieves information using the Finger User Information Protocol (RFC1288) and is able to penetrate a firewall. Program is generic enough to retrieve user information or service responses from "vending machines."
- Wrote Windows combination "DOS to Unix" AND "Unix to DOS" file conversion program. GUI allows file selection through modified COMMDLG file open box or through click and drag.
- Added "Slip on Demand" and dial/connect functionality to a DOS Slip device driver written in assembly.
- Wrote Windows GUI to program and control the slip device driver. Used DPMI functions to communicate with this real mode driver from a Windows protected mode program.
- Wrote Windows GUI for DHCP (Dynamic Host Configuration Protocol) to display and modify lease status. GUI had a custom (subclass) listbox control which always has one item selected and keeps this selection when items are added.

**Mystic Software, Software Consultant, Binghamton, NY, 5/94-7/94**

Using Visual C++ for Windows NT, wrote a graphical Unix File Transfer Program (FTP) for Windows NT over Winsock/TCP/IP, supporting drag and drop interface from other drag originators to File Manager (custom.) Developed many reusable classes, including a drag/drop listbox class.

**Mystic Software, Software Consultant, Binghamton, NY, 11/93-1/94**

Wrote Windows Screen Saver using Visual C++. Screen saver uses a maze object which bounces as it draws itself, then solves itself. Screen Saver setup allows user to alter maze construction in a variety of ways and allows choice between BitBlt() and ScrollWindow() animation. Because Visual C++ debugging requires medium model, yet screen savers must be small model, architecture was a small model front end which calls a medium model DLL.

**XLI Corporation, Software Consultant, Woburn, MA 6/93-8/93**

Using Microsoft Visual C++ and MFC, wrote a Windows for Workgroups Client/Server GUI to control a print queue manager. GUI client communicates with server via DDE and NDDE (network DDE). GUI contains custom click & drag interface, toolbar, and various custom controls. GUI capable of locked mode (no task switching allowed). GUI can lock separate Winexec'd application through my window subclassing DLL. Implemented pseudo server to test client GUI. Project contains numerous C++ base and derived classes, including several DDE objects.

**Digital Equipment Corporation, Software Consultant, Littleton, MA 4/93-5/93**

Designed and wrote C/Windows NT multi-threaded TCP/IP client/server application for controlling various workstations participating in a performance demo for the Comdex trade show. Master module communicated with Excel via DDE and with slave modules via Winsock/TCP/IP. Developed C++ communications modules for network DDE (NDDE). Jointly developed communications modules for Berkeley (Unix) sockets and DECMessageQ. All development done in Windows NT. Slave code was portable to Intel (Microsoft), Alpha, and MIPS compilers.

**Galactic Industries Corporation, Software Consultant, Salem, NH 10/92-2/93, 4/93-5/93, 10/93, 1/94**

- Wrote, modified and enhanced C/MS Windows programs.
- Wrote 80386 assembly language drivers for a Scientific Software company.
- Wrote C++ and assembly file conversion programs.

**Dun & Bradstreet Software, Software Consultant, Framingham, MA, 3/93 - 4/93**

Designed IBM 3270 mainframe application screens for MS Windows GUI. Involved capturing 3270 screens with a PC/mainframe connection, turning screens into GUI panels, and writing sequence code to drive the mainframe transactions. Wrote Windows 3.1 DLL in C++ to extend Flashpoint functionality. Used Flashpoint development tool and Soft-Ice for Windows debugger.

**Optical Technology Devices, Software Consultant, Elmsford, NY, 2/93 - 3/93**

Developed a C++/MS Windows program with GUI to control and collect data from a fluorometer in real time. Instrument communication was over RS232 serial port, with background data collection possible. Incoming data was displayed using a DLL custom, intelligent graphing control.

**Mystic Software, Software Consultant, Binghamton, NY, 11/92-1/93**

Developed a C++/MS Windows graphical application to render and rotate/animate a wire-frame model in 3 dimensions. ROTATE contains a DLL custom drawing control for smooth animation. Developed a C++/MS Windows data graphing application. Program reads points in ASCII X,Y format and figures out how to best display them. GRAPH also had a DLL custom control to scale and display the X,Y points.

**Microdyne, Senior Software Engineer/Project Leader, Westford, MA 4/92 - 10/92**

Using Microsoft C/C++, wrote a graphical Unix File Transfer Program (FTP) for Windows 3.1 over Sockets/TCP/IP, supporting custom (but MS standard) click and drag interface and coexisting/communicating with the Windows File Manager. Program consisted of a GUI layer and a FTP Engine, which ran over a Winsock API/DLL. Assisted with Winsock DLL/Network driver development and debugging. Attended Win32/Windows NT conference to determine impact on future software projects. Helped junior software engineers learn C/Windows programming, Windows DLL writing and debugging techniques.

**Galactic Industries Corporation, Senior Software Engineer, Salem, NH 8/90 - 4/92**

- Wrote MS/Windows 3.1 Program Editor, using C, with assembly language string searches, scrolling, etc for increased performance.
- Wrote MS/Windows 3.1 Install/Setup program for a multi-disk, script driven product installation, including custom controls and icon/bitmap creation.
- Developed and documented real-time 80386 assembly language programs and device drivers in the IBM PC/AT environment for a scientific software company.
- Wrote interrupt-driven device drivers to communicate with/control spectrophotometers and chromatographs through RS232 and proprietary interfaces.
- Wrote and updated other programs in 80386 assembler, Borland/Microsoft C/C++ and Array Basic. Extensive experience with Microsoft & Borland debuggers and dual monitor debug system.

**Yankee Environmental Systems, Software Consultant, Groton, MA 11/91 - 3/92**

- Wrote IBM PC/AT 80386 assembly language, memory resident program (TSR) to process serial port interrupts and log sensor data to disk. Operation is in background and totally transparent to the user.
- Wrote Windows application (in C) to configure and test the sensor.

**Ultra Systems, Firmware/Hardware Consultant, Westford, MA 5/91 - 7/91**

Modified firmware and microprocessor circuitry of a Z80 based currency counter to convert from counting American dollars to Indian rupees. Main circuit board contained Z80, Zilog support chips, and analog sensors. Performed various mechanical design tasks.

**Wang Laboratories, Hardware Engineer, Lowell, MA 6/87 - 8/90**

- Member of VS8000 Mainframe (IBM 370 Architecture) 4 person design team which produced an LMA10K CMOS ASIC version of 3 board VS7000 Mainframe CPU set. Designs included the Address Generation Unit, high speed 3 port adder, and decimal add/subtract correction circuitry.
- Designer of an LMA9K CMOS ASIC, responsible for design, functional & fault simulation, and layout/placement. Chip performed ATU (address translation unit) function.
- Member of next generation computer design team. Designed and simulated a memory cache controller and microsequencer. Wrote assembler for cache microsequencer in UNIX/AWK pattern scanning language.
- Wrote C programs (in a Unix and Windows/Dos environment), Unix scripts, and VS microcode programs.

**Grotech, Firmware Consultant, Groton, MA 4/90 - 6/90**

Wrote embedded, real-time code for 68HC11 microcontroller based weather monitoring system. System contained master and slave controllers connected by a high speed RS422 async and RS485 sync serial links. Remote sensor used pulse accumulator and A/D inputs on 6811. Evaluated 8051.

**Middlesex General Industries, Software Consultant, Woburn, MA 10/88-11/88**

Wrote real-time 80286 machine language interrupt driven VT240 emulator for an IBM AT. Program initiated C calls and called C subroutines. Used to display status from a 68000 oil bath controller.



**EG&G Electro-Optics**, Design Consultant, Salem, MA 5/88 - 8/88

Prototyped, tested, and debugged hardware model of an ASIC; tested actual ASIC by interfacing to a computer; tested ASIC with analog I/O devices and external circuitry which make up a kidney dialysis substitute. Experience with Xilinx gate array.

**Paperless Accounting**, Hardware/Firmware Consultant, Troy, NY 1/87 - 5/87

Designed Z80 based computer and Z80 assembly code to communicate with a personal, miniature memory element and user. Device similar to an automated teller machine.

**IBM, Plotter Engineering Co-op, Kingston, NY** 1/85 - 8/85

Wrote embedded microcode for an 8088 based IBM PC card; repaired plotters; modified interface test programs for RS232 and IEEE488; designed microprocessor circuits.

## **ARTICLES**

"Cron Job with Random Start Delay" - Blog

"Git Submodule with Local Changes Example" - Blog

"Copying a Single File with a Yocto Recipe" - Blog

"Semihosting Debug on STM32F4" - Blog

"STM32 Discovery Development on Linux" - Blog

"STM32F4 USB Virtual COM Port (VCP)" - Blog

"Realtime Audio DSP with the STM32F4" - Blog

"Tuning the Moog Etherwave Theremin" - Blog

"Code Browsing with Emacs, GLOBAL, and Speedbar" - Blog

"Android Activity Bar Framework" - Blog

"Generating an Audio Sine Wave with Java" - Blog

"Android Simple Yes/No Message Box" - Blog

"Creating a Bootloader Environment (Freescale Coldfire Example)" - Blog

"Build a Variable Voltage Supply for Cheap" - Seattle Robotics Society Encoder, 11-12/01

"Add RAM to your 68HC11 Development Board" - Seattle Robotics Society Encoder, 11/00

"RS232 Debugging Cheatsheet" - Seattle Robotics Society Encoder

## **HOBBIES**

Backcountry Skiing, Mountain Biking, Kayaking, Snowshoeing, Snowboarding, Tinkering