|  |
| --- |
| **SUMMARY OF QUALIFIFCATIONS** |
| **Embedded Platform Software Engineer** with more than 25 years of software development experience in various technical and cultural settings. Demonstrated ability to bring projects from inception to production within budget and schedule. Intimately familiar with all processes of software development and life cycles with emphasis on projects close to hardware specifications, driver designs for OS integration or pre and post Silicon validation and test automation.  Technical expertise includes:   * ARM, MIPS single and multi-cores based platforms * C, C++, C#, Objective C, Assembly, tcl/tk, apple script, Lua, python, Perl, m4, MySQL, Mono. * Yocto, Linux, Android. * Git, Gerrit * Ethernet, TCP/IP, I2C, PCI, SDIO, USB 2.0 OTG * PCI, PCIE * Linux, Windows, Mac OS X, VxWorks, Nucleus, “bare metal” platforms. * Open Source systems and applications * 3W Technologies LAMP WAMP XAMP |

|  |
| --- |
| **PROFILE** |
| **Objective:**  Desire to contribute as a contract basis to the successful completion of any projects involving embedded platforms and their challenges. Notably interested in any project where device drivers design and testing are major components. Board Support Packages for U-Boot, Linux, FreeBSD, FreeRTOS or Open Source projects are areas where I can make a positive contribution. |

|  |
| --- |
| **PROFESSIONAL EXPERIENCE** |
| |  |  | | --- | --- | | PCH International | **Firmware Consultant** | | **Consumer Electronics** | **June 2018 – January 2019** | | PCH International Company:  ***Project #1***: Automatic Insect Pheromone Spraying Device: The embedded device is designed to spray pheromones on a preprogramed schedule.  It is used in an agricultural setting to affect the reproductive mechanism of insects affecting the crop.  The device is developed around an MSP430 from Texas Instrument.  The application is coded in C.  The program could be considered smallish (~1000LOC).  The technical challenge consisted in achieving sub micro amp power consumption.  The MCU provides a collection of mechanism to put itself in ultra-power saving mode.  A timer wakes the MCU at regular intervals.  Battery live has to be tested to last 10 years in the field.  No Field updates of firmware is designed in.  Code is maintained on Git via GitHub.  Documentation is provided via Doxygen.  ***Project #2***: Cosmetic device for the precise application of makeup formula to cover skin blemishes.  The system existed in a Generation 1A version. The statement of work consisted in a study for cost reduction of the hardware component along with a major redesign of the firmware.  The system consists of two units. One called the Applicator the other one called the service station.  The Applicator consists of a high-resolution camera and a printing cartridge as the main components. These are controlled by an STM32F429 Cortex-M4 MCU.  The original code was a bare metal implementation written mainly in C++ for the application layers and C for the HAL layers. The major “surgical operation” was to adopt the popular FreeRTOS as the mechanism to segregate functionality in appropriately defined tasks.  Real Time requirements of sub micro seconds for the firing of the print cartridge necessitated very precise design of the GPIO driving the cartridge.  Most of the Application remained written in C++. The Hardware Abstraction Layer is partially generated by the STM33CubeMx utility. This utility is critical in properly configuring the hardware components and the code generation of HAL drivers.  A host application running on a Windows 10 platform is written in C#. The application was modified to accommodate new features in the firmware side. New widget on the UI were created or modified. Development of this portion of the work was done with Visual Studio 2017.  Additional peripherals used in this work are: UART, I2C, SPI, DMI DVP, DMA, a large selection of Timers used to orchestrate the proper sequencing of events in the system.  A host application using a proprietary protocol over CDC USB Serial port, permits the configuration and debugging of the device. The development environment is IAR and JTAG.  The resulting code base is in excess of 80K LOC. | |  |  |  | | --- | --- | | ANKI | **Consultant** | | **Robotics** | **February 2018 – May 2018** | | Robotic AI Toy Company   * Target Platform derived from Qualcomm Snapdragon MSM8909 Reference design * Work performed:   + Yocto build system for Android and Linux.  The team is migrating from an Android centric build to a Linux one. All the tools are still based on the Android toolkit such as adb and related utilities. * Personal intervention on the following subsystems   + High Speed Serial Driver. Port of this driver and fix several issues   + SELinux: Create and modify some of the Yocto (poky) recipes to allow smooth user management and daemon permissions.   + Ensure singleton execution of all daemons within the ANKI robot.   + Port of Industrial Input / Output Linux driver for the BMI160 Bosh IMU device with following functions:     - Magnetometer     - Gyroscope     - Accelerometer     - Temperature * V4L2   + Fixed issues with CSI clock tree (very tricky) | | | **PCH Lime Lab** | **Consultant** | | **Consumer Video/Audio Devices** | **April 2017 – July 2017** | | Our client is a consulting firm providing turnkey systems to their own clients. The particular project we were involved in is a consumer electronics device. The handheld camera provides all the expected features of a modern camera Viewfinder, connectivity to a host system and sound recording capabilities among a set of other features.  The system is developed around an IMX6 NXP platform running Linux. The software components are built using Yocto build-system. Our contribution to the project included the following:   * Video subsystem with HDMI and LCD mirroring requirement, the video subsystem is provided by the IMX6 Soc. * Audio Subsystem to handle audio input and output for Line IN and microphone jack.   Alsamixer utility used to fine tunes the various audio components of the subsystem Wolfson Stereo Codec WM8962 is the core of the audio subsystem.   * Various drivers were studied and modified to achieve the desired functionality in conjunction with device tree file (dts) modification   Project is developed in C, C++ and uses Qt5 in the application layers. | |  |  |  | | --- | --- | | Logitech Ultimate Ear | **Consultant** | | **Audio Consumer Device** | **November 2014 – December 2016** | | As a consultant to the Ultimate Ear (UE) division of Logitech,  The division who hired my services is designing and building Audio Devices for the general public.  The technology makes heavy use of the Bluetooth Standards and profiles both standard and Bluetooth Low Energy (BLE).  This for the audio streaming and wireless control of the device, field firmware update and music streaming.  USB technology is used intensively as well as a mean to debug, control and update the firmware of the device.  The general mission of my contribution was to design, implement and deploy a comprehensive automation system to enable the Engineering and Quality Assurance teams to design and run regression test suites.  The system that I designed, implemented and deployed is built around a collection of Python 3.4.3 modules and packages. They address the following areas of concerns:  USB Interface:  Bluetooth Interface  SPI utilities  Audio Fingerprinting  Audio Quality assessment.  Database of Test results  HTML report generator  Continuous Integration of Software (Jenkins, Hudson)  Management of GIT repositories  Issues Tracking with JIRA  As stated earlier the majority of the code created is in Python but also involved the development of shared libraries in C and C++. | |  |  |  | | --- | --- | | **Trimble Navigation AG Division** | **Consultant** | | **Embedded GPS Systems** | **January 2014 – November 2014** | | Trimble Navigation produces GPS receivers and specific software system for a large number of field applications.  This particular division produces systems for the agriculture industry and automated farming tools.  The project I was involved in is meant to supply a European OEM with all the tools and documentation required to integrate the Trimble receiver in a larger system onboard of tractors and farm implements. My role in this context was to provide the following:  An automated documentation system used to publish CONFLUENCE pages ([www.atlassian.com/](http://www.atlassian.com/)**confluence)**  The pages are automatically created from an xml description of the protocol packets required to configure a receiver. The tools used to interface with the confluence server are in a mix of java, python and bash scripting on Ubuntu virtual host.  A tool to send and receive protocol packet over a communication link (Serial, CAN, Ethernet) to the GPS receiver for configuration purpose. The tool is written in C and C++. This tool is ported as well to Win32 and ECOS platforms  A tool to update receiver firmware in the field. The update is achieved via serial link and specialized protocols to upload and flash the new image of **firmware** to the target.  The tool is developed in C and C++. It is organized in a set of libraries for easy integration and portability.  Added functionality to the receiver to implement new protocol packet handling (cmd/rsp). The code is written in C and C++.  The host platforms are Win32 and Linux (as VM under Virtual Box)  The target platform runs an RTOS (ECOS) for a PowerPC proprietary SOC. | |  |  |  | | --- | --- | | **PureWave Networks, Santa Clara CA** | **Consultant** | | **Linux embedded Software Engineer (Linux, ARM, BSP NXP LPC3250, U-Boot, Linux BSP, Ethernet, UBIFS)** | **August 2013 – October 2013** | | PureWave Networks develop a Small Cell Base Station for LTE networks. Their platform of choice is a TI reference design based on the TMS320TCI6614. An ARM Cortex 8 and 4 DSP cores are the main components of the SOC. I was called to this situation to debug issues in U-Boot and Linux for the Ethernet PHY (Vitesse 8572) and UBIFS on NAND flash. Worked initially on the AIF2 (Antenna Interface CPRI protocol). | |  |  |  | | --- | --- | | **Sentient Energy Inc, Burlingame CA** | **Consultant** | | **Linux embedded Software Engineer (Linux, ARM, BSP NXP LPC3250, U-Boot, Sensor Drivers, C, Python)** | **January 2011 – September 2013** | | Sentient-Energy provides distribution power lines monitoring devices. The project I was contributing to, is a complex monitoring unit based on NXP LPC3250 reference design, for the first version of the device. The second version of the product is developed around the ATMEL AT91SAM9260 SOC. Early in this project I designed and implement proprietary boot process, later replaced by U-Boot. Bring up of Linux 2.6.34 with initially a simple make process later improved with Open Embedded Build framework. Wrote several drivers for the sensor subsystem using i2c bus and on-chip 10 bits ADC. The unit communicates over a mesh network similar to the smart meter one and uses at this time the Silver Spring Network Radios. Our code lives in GIT repositories. Designed and implemented an automation system for regression verification and validation of the system.  Also developed Block Level Testing infrastructure (bare metal) to validate individual component of the system such as SDRAM initialization, memory tests, Data Acquisition subsystem and sensor calibration.  Application space includes current monitoring and fault reporting facilities. Temperature monitoring and Fault Current Indication applications. This application space leverages the DNP3 open-source provided by GEC.  Design and implement manufacturing test infrastructure in Python and C code for both platforms. | |  |  |  | | --- | --- | | **Interface Masters Technologies, San Jose CA** | **Consultant** | | **Firmware Engineer (Linux, MIPS, Cavium Networks, GB Ethernet MAC & PHY programming** | **September 2009 – December 2011** | | My contributions included developing U-Boot BSP and Hardware Bring Up on a Cavium Networks based platform - MIPS 64 multi core.  Developed interface drivers for Marvell PHY and MAC IP's. Worked on CPSS based Marvel switch re-architecture under Linux and FreeBSD.  Development of Transfer protocol over Serial Line on Freescale Mx37 platform running Linux, Ltib, C++, Boost +, Qt 4.6 for User Interface. | |  |  |  | | --- | --- | | **Broadcom Corporation, Sunnyvale CA** | **Consultant** | | **Firmware Engineer (Linux, ARM, USB, SDIO)** | **October 2006 – January 2009** | | Worked on the Post-Silicon verification and validation of the overall platforms.  My contribution to this group has been the validation and verification of the FS and HS USB OTG IP within all the ASIC’s I had opportunity to work with.  This IP is an ASIC integration of a core supplied initially by Synopsys.  During my contract within this group I acquired a complete understanding and intimate knowledge of this core and its integration and interaction with the remainder of the ASIC series where it exists.  The ASIC in based on a set of ARM cores ARM9 and ARM11 present as single or dual cores in the ASIC’s  The software environment is to be described as a three tiers approach to verification   * Block level Testing: This level of testing is not running under the control of an OS it accesses the registers and memory blocks of the chip directly and handles interrupts and events in a loop of code background and foreground. * System Level Testing: This level of testing is running under an embedded Linux OS. It allows the scheduling and execution of various tasks mostly related to the interaction of multimedia blocks within the ASIC. Most of this is implemented within a rich offering of devices drivers and specific user space applications. * Reference Phone Application: This level of testing is done in the context of a RTOS such as Nucleus and WinCE.   Synopsys supplied with their core a sample driver that I took as a base for my work. Beside basic verification of the USB core with this driver and the block level testing. I used the driver as a base to provide richer connectivity to the platform under test. Combined with an Ethernet driver I was able to run the TCP/IP stack over the USB link. This opened the platform to all TCP/IP and socket applications.  Ported and validated the Open SSH, DROPBEAR (embedded open SSH).  Host Linux box running Centos 5, JTAG RVI ARM ICE, JTAG LAUTERBACH ICE, various Broadcom proprietary modular validation platforms, mixed signal oscilloscopes, Logic Analyzers, GSM traffic analyzer and generators.  Code is a mix of C and ARM assembly code. | |  |  |  | | --- | --- | | **AMCC, Sunnyvale CA** | **Consultant** | | **Senior Software Engineer (tcl/tk, Instrumentation)** | **November 2005 – May 2006** | | Designed and Implemented a fully automated test platform for the post validation of network processor ASCIC’s. The system is tcl/tk based. | |  |  |  | | --- | --- | | **National Semiconductor, San Jose CA** | **Consultant** | | **Senior Software Engineer (Bluetooth, CP3000, IAR)** | **June 2003 – October 2005** | | Designed, developed and validated an “In System Programming” (ISP) Feature for National Semiconductor Bluetooth reference designs boards.  The reference design boards are developed around a CP3000 ASIC providing MCU and a variety of peripheral devices configured in various versions of the chip.  The code developed for NSC is meant to be fully configurable and modular allowing a single source base to provide functionality for the complete offering of configurations.  The MCU is a CR16C. The devices are Bluetooth base band, USB 2.0, USART, On and Off chip Flash memory.  Cross Development tools are IAR C compilers and linkers and NSC GNU based tool chain.  Debugging is through Win-IDEA IC3000 in circuit emulator. | |  |  |  | | --- | --- | | **Infineon Technologies, San Jose CA** | **Consultant** | | **Senior Software Engineer (VxWorks, SONET, tcl/tk)** | **June 2001 – May 2002** | | Designed and developed SONET/SDH telecom ASIC diagnostics software. Coded OC192 device driver for Agilent test equipment. Developed remote diagnostics modules for OC768 Framer chipset. Implemented alarm system via pager/email. Coded signal capturing and display modules.  System needs to work unchanged on both Linux and Windows.  Assembled and programmed a rack of IEEE-488 addressable instruments and a Win2K host. Developed host application and device drivers with tcl/tk and C/C++.  Environment: Windows 2000, Linux, TCL/TK, C/C++, VxWorks, and PowerPC. | |  |  |  | | --- | --- | | **Cypress Semiconductor, San Jose CA** | **Consultant** | | **Senior Software Engineer (Bluetooth, ARM, 8051)** | **October 2000 – May 2001** | | Ported, debugged and validated Bluetooth stack. Designed and developed a Host Controller Interface (HCI) Exerciser Tool. Established comprehensive Test plan and verified standard conformance of a Bluetooth Baseband and RF ASIC. Environment: C/C++, ARM7, 8051, USB, RS232, Ericsson, Digianswer. Win2k. | |  |  |  | | --- | --- | | **Sutmyn Storage Corporation, Sunnyvale CA** | **Consultant** | | **Senior Software Engineer** | **May 1996 – May 2001** | | Developed a C++ SCSI Device Driver Exerciser for the purpose of verifying the DLT8000 Simulation component of network wide back-up solution. Adapted LINUX 6.2 and generic SCSI device driver to function along with expect and tcl/tk to provide a comprehensive QA verification tool. Coded Tool in a combination of tcl/tk routines and C/C++ with specific Classes to encapsulate the core of SCSI commands. Created Graphic User Interface using tcl/tk.  Designed and implemented a proxy system administrator for the integration of IBM ADSM within a storage management solution. Architect and designed proxy with the Rose98 UML OOD modeling tool. Coded proxy system in C++ and ported to Solaris 2.X. The proxy monitors and controls the activity of the ADSM server and his associated processes. Wrote test plan, test suite Tests in tcl/tk.  Designed and implemented a TCP/IP back plane network driver between SBUS and I960 boards.  Designed and implemented an ESCON and Host Simulator for a Virtual Tape Device.  The device is an emulation of an IBM 3490 Tape Control Unit and provides a Virtual Tape Library on large RAID.  The Development environment is a heterogeneous network of Sun Sparc Stations and a set of proprietary embedded controllers.  The embedded controller is developed around the I960 chip running VxWorks with GNU Tool Chain. Applied strict Development methodologies to this very complex system.  Used Rational Rose UML Object Modeling for every aspects of design.  Used Atria Clearcase for Configuration Management. Used VxWorks GNU C++ compilers in all aspects of the project. The overall project is over 100 KLOC with about 30K LOC dedicated to the embedded side. Testing with VM and MVS, Muff, Friend, Dutch, JCL on an IBM mainframe  Created Documentation in Word6.0 and HTML3.2 User Guide on Sutmyn Intranet. | |  |  |  | | --- | --- | | **Minolta MIL, Sunnyvale CA** | **Consultant** | | **Senior Software Engineer (Print Engine, VxWorks RTOS)** | **April 1999 – September 2000** | | Designed and developed an embedded printer controller firmware. Designed and Developed C/C++ bring up and QA verification firmware for PM-24 ASIC. Designed and developed C/C++ VxWorks compliant device drivers for Flash Memory, SDRAM, DMA, ATA-2 IDE, IEEE1284, USB and TCP/IP. SNMP printer management through MIB’s  Developed Printer Driver on the Window 98 and Window 2000 Side.  Developed boot up code for SH2 and PowerPC Board Support Package (BSP).  Environment: VxWorks, C/C++, Hitashi SH2, PixelMagic PM24, FPGA Verilog HDL. | | |

|  |
| --- |
| **EDUCATION** |
| |  |  | | --- | --- | | **1972 - 1974** | **Computer Science Institute (September 1974) - *“Computer Science Degree”***  Institute of Technologies, Clermont-Ferrand France  Degree in Computer Sciences. Logic, Compiler Design, BASIC, FORTRAN, COBOL, ALGOL, Programming, Mathematics, Numeric Analysis. | | **1969 - 1972** | **Clermont-Ferrand University (1972) - *“Business Administration Degree”***  Degree in Business Administration, Law, Accounting and Humanities. | |