Erik Nyquist

Personal Details

Website – https://www.ekn.io
Github – eriknyquist
Email – eknyquist@gmail.com

Areas of Expertise

Programming Languages
C
C++
C#
Python
UNIX shell scripting (bash, sh)

Tools & Environments
Git
GCC/Clang
Makefiles
GNU ld (linker) scripting
Google Protocol Buffers
GDB
OpenOCD
Valgrind
LATEX
Eclipse
Jenkins
JIRA
Unity/CMock

Personal Skills

- Test-driven development
- Fault finding and debugging on embedded systems
- C and C++ for memory constrained embedded systems
- Rapid prototyping/testing with Python or UNIX shell scripting

Interests

- Compiler design and implementation
- Programming language design and implementation
- Playing music (drums, piano)

An enthusiastic and skillful software engineer, with a comprehensive knowledge of development and validation practices for embedded software systems. Accustomed to delivering and enforcing high quality code and documentation in a high-pressure environment.

Experience

Firmware Engineer, NOVO Engineering
Vista, CA

Aug. 2017 - present

Designing and developing low-level firmware and user-level software for embedded systems (Linux, FreeRTOS)

- Developed firmware for multiple medical device products using nRF SoCs (closed loop insulin delivery system, "smart" insulin pen cap)

- Developed firmware for multiple pieces of test equipment for medical device products using nRF SoCs

- Developed automated unit tests and integration tests (Python, C) for multiple medical device products

- Participated in creation of product verification documentation for multiple medical device products

. . . .

Software Engineer, Intel
San Diego, CA

Aug. 2016 - Aug. 2017

Developed low-level hardware drivers and firmware for Intel's low-power SoC products with a small team, including Intel's Galileo, Joule and Curie modules (Linux, RTOS and bare-metal).

- Developed drivers and firmware (C, C++) for the Intel Arduino 101 board, using the open-source Zephyr project (co-operative scheduling real-time kernel for small embedded systems. www.github.com/01org/zephyr).

- Developed Arduino user libraries and APIs (C, C++) for the Intel Arduino 101 board, reviewed/accepted patches from public users regularly (www.github.com/01org/corelibs-arduino101).

- Developed drivers and developer tutorials (C++) for the Pattern Matching Accelerator on the Quark SE C1000 microcontroller (on-die network of 128 hardware neurons capable of autonomous learning/training, and pattern recognition. www.github.com/01org/Intel-Pattern-Matching-Technology).

- Participated in design and development of the Curie Open Developer Kit (C, C++, bash/shell, OpenOCD, GDB), a lightweight framework for developing applications (bare-metal, or using the Zephyr kernel) that can fully utilize the two asynchronous CPU cores on an Intel Curie device (www.intel.com/curieodk).

. . . .

SoC Software Engineer, Intel
San Diego, CA
<div align="right">

Sep. 2014 - Apr. 2016
</div>

Developed and tested low-level hardware drivers and firmware for Quark SoCs (Linux and bare- metal).

- Developed Linux-based test suites (C, Python), for various embedded protocols and systems (UART, CAN, I2C, SPI), on FPGA-emulated systems and during new device bring-up/power-on.

- Developed test suites (C, Python) for customer SW tools (tools for writing BIOS configuration data to an EEPROM/flash device on a customer board, tools for generating a bootable OS image with custom BIOS, kernel and root filesystem parts)

- Functional testing of customer SW tools & platforms (microcontroller I/O libraries, toolchain, debugging tools).

- Low-level debugging and fault-finding (source and assembly level debug with OpenOCD and GDB, analyzing linker scripts, linker map files, and looking at scope traces / logic analyzer traces, to debug I/O device drivers or C/C++ run-time init. code running on a target microcontroller device)

. . . .

Software Design Engineer, Intel
Ireland
<div align="right">

Aug. 2012 - Sept. 2014
</div>

Developed and tested Linux-based software & drivers for the Intel Quark X1000 SoC. Brought Intel Galileo board (first x86-based Arduino board) from design to market.

- Prototyped software solutions for supporting Arduino-compatible code on an embedded linux- based platform

- Usability and functional testing of customer SW tools (Code editor/IDE, setup tools and scripts)

- Provided technical support to users via online forum www.maker.intel.com

Education

Master of Science, Computer Science
University College Dublin, Belfield, Ireland

Bachelor of Engineering, Audio Visual Media Technology
Dun Laoghaire Institute of Art, Design and Technology, Dublin, Ireland