MichaelBartling

Graduate Applications Engineer, Arm

contact

4500 Steiner Ranch Blvd
Apt 3503
Austin, TX 78732
USA

+1 (214) 707-2808

michael.bartling15
@gmail.com
minionhut.com
github.com/mbartling

programming

Python, C++, C++11
Julia, Matlab, Verilog
C, Embedded C, R,
SystemC

OS

Debian, RHEL6-7,
Windows, mbed OS,

Interests

professional: C++11 development, embedded software, optimization methods, theoretical security, machine learning, data visualization
personal: cooking, 3D CG art (Blender), auto restoration, animation

Experience

Full Time and Internships

Arm — **Graduate Applications Engineer: Developer Experience (DevX)** — 2017–Now
Austin, Texas

- **uTensor**. Designed lightweight machine inference capabilities for Cortex M devices. `https://github.com/utensor`.
- **Iterative model learning on edge devices**(*In Progress*).
- **SpamBlaster**. Designed domain language specific spam classifier for Mbed ecosystem which significantly outperformed previous system.
- **TheFAQ**. Semi-automatic ontology generator for better search within Arm.
- **Mbed Greenlight** Front end accessibility testing for `os.mbed.com`
- **Device Health project**.

University of Texas — **Graduate Research Assistant** — 2014–2017
Austin, Texas

- **Dynamic analysis of Windows malware on networks**. Designed large scale malware analysis engine and virtual machine management system using AWS and MongoDB. Wrote low-overhead system call interceptor for Windows platforms. Developed robust anomaly detection pipeline for Windows malware. This software is basis for **one of the largest dynamic malware analysis ever conducted in academia**, collecting approximately 3400 hours of malicious system call traces.
- **Dynamic analysis of mobile malware on networks**. Built state-of-the-art user trace record and replay system for Android applications, injected key malware categories into common applications, designed intelligent anomaly detectors for Android system calls.
- **Context aware sensing**. Automatic classification of user motion into activities based on smart phone accelerometers. Dynamically *learned* privacy preserving user motion models. Automatic fall prediction and detection, which is the leading cause of death due to injury of the elderly. Inferring information across untrusted contextual boundaries.

Texas Instruments — ***Software Development Intern*** — Summer 2014
Dallas, Texas

- RFSDK Software development
- Designed end-to-end experiment manager for software-hardware interfacing.
- Designed intelligent LTE frame modeling and generation scripts significantly reducing software/hardware testing times while allowing for dynamic end-user capacity simulations.
- Digital pre-distortion design

Texas Instruments — ***Software Development Intern*** — Winter and Summer 2013
Dallas, Texas
Wireless Backhaul Project

- **Ported Contiki OS** to TI FRAM line microcontrollers. Completely redesigned build system allowing for faster incremental builds. Third party required $45k and 3 months to port code, I finished porting the code for free in just two weeks in my spare time. Enabled TI to conduct IoT R&D with minimal effort.
- Designed and optimized Line of Sight channel estimation drivers.
- Designed and optimized Line of Sight 2x2 and 4x4 MIMO channel equalizer drivers. Conducted precision study on fixed point versus floating point implementations.

Texas Instruments — ***Software Development Intern*** — Winter 2013
Dallas, Texas
Helped formulate non line-of-sight transmitter chain on C6614 EVM

Texas Instruments — ***Software Development Intern*** — Summer 2012
Dallas, Texas
Designed and optimized Reed Solomon processing chain for TI C6614 EVM

Noteworthy side projects

2018 — **uTensor Handwriting recognition demo** — Arm
uTensor minimum viable product demo based on simple 2-layer fully connected NN and MNIST. Code: `https://github.com/uTensor/utensor-mnist-demo`, Video: `https://www.facebook.com/neil.tan/videos/10159971829870385/`

Spring 2016 — **Spatially Hashed Photon Map** — UT Austin
Computer Graphics Final Project. High performance ray tracer with photon mapping support written in C++11. Key idea is that can encode photon aggregation into a data structure at build time rather than render time. Furthermore, can leverage O(1) lookup time during rendering. `http://mbartling.github.io/photonMapper/`

Spring 2016 — **QtLC3 and pyLC3** — UT Austin
Rewrote Yale Patt's LC3 architecture simulator for use in classrooms. Simulator includes full python integration for easy unit testing and grading, and the GUI is written in the Qt5 framework. `http://minionhut.com/blog/post/lc3-simulator-overview`

Education

2014–
Dec. 2016 **M.S.** Computer Engineering — The University of Texas at Austin
Advisor: Mohit Tiwari
Context-aware sensing, Dynamic malware analysis, Machine Learning.
GPA: 3.8

2011–2014 **Bachelor of Science** , Summa Cum Laude — Texas A & M University, College Station
Electrical Engineering
Specialized in Computer Engineering
Sub-specialized in Signal Processing and Image Processing.
GPA: 3.9

2009–2011 **Advanced High School Diploma** — Texas Academy of Mathematics and Science
UNT, Denton, Texas
Graduated high school 2 years early to attend accelerated TAMS program.
GPA: 3.89

Awards

2015 **Dell Innovation Award: Hack TX** — Austin TX
Distinguishing style and content in images: The ability to create any Instagram filter.

2015 **2nd Place MDP Hackathon** — Athena Health, Austin TX
Accurate fall prediction and motion state regression using cellphone accelerometer information.

2014-Present **Departmental Fellowship** Computer Architecture and Embedded Processing, The University of Texas

2014 **Summa Cum Laude** — Texas A& M University, Electrical and Computer Engineering

2011-2014 **President's Endowed Scholar** — Texas A& M University, Electrical and Computer Engineering

2011-2014 **Boltzman Scholar** — Texas A& M University, Electrical and Computer Engineering

2008 **Eagle Scout** — Boy Scouts of America

Courses

- Convex Optimization
- Large Scale Machine Learning
- Multicore programming
- Parallelism and locality
- Real Time Operating Systems
- Security: Hardware Software Interfaces
- Engineering Programming Languages
- Computer Graphics
- Computer Architecture
- Digital Signal Processing

- Image Processing
- Microprocessor Design
- Advanced Logic Design
- Ultrasound Imaging
- VLSI I