

**Objective** Solve interdisciplinary problems for automation, efficiency, reusability.

---

## Education

### Princeton University, Princeton, N.J.

Bachelor of Science in Engineering, in Electrical Engineering Sept. 2012 – May 2016  
Certificates (minors): Applications of Computing, Robotics and Intelligent Systems

---

## Experience

### Google LLC / Alphabet Inc.

Software Engineer — Nest, Devices & Services, Mountain View / Palo Alto, Calif. May 2017 – present

- **Led migration to protobuf-based cloud communications** for thermostats (RTOS-focused)
  - Directed and delegated the contributions of 5½ engineers for this project
  - Architected a **multi-OS data model library / framework**, and built initial implementations for a system for handling 100+ protobuf messages / 500+ fields
  - As lead / domain expert, **optimized protobuf message** definitions and parsing, for the needs and constraints of low-memory RTOS environments
  - Wrote whitepapers for service engineers, for developing **memory-efficient / RTOS-tractable implementations of REST APIs**, while considering service costs and scalability
  - Designed and implemented features for Google's open-source [Pigweed library](#), to add functionality that Nest needed for advanced protobuf features (reflection, RPC, etc.)
- **Led the remote comfort sensing (RCS) feature** for integrating [user-placeable external sensors](#) with Nest thermostats
  - Planned, designed, and implemented a dynamic runtime **migration of RCS cloud communications to a new Weave-based API**; led creation of [Weave schema data models](#) for RCS
  - Designed and built the first version of the **RCS multi-room sensing feature**, including all logic and data modelling, the APIs for mobile apps and services, and all UI changes
  - Advocated for and implemented various API improvements for handling sensor liveness, and for arbitrating between multiple data sources
  - Worked with teammates to augment thermostats' Bluetooth driver stacks for RCS-specific application needs, some of which the drivers did not originally support
- Identified and implemented missing features for the **build and test infrastructure** for 6–10 separate thermostat products, both **RTOS-based and Linux-based**
  - Upgraded the toolchain and build for several legacy products from GCC 5.1 to GCC 9.3 to facilitate C++17 usage; fixed numerous compatibility issues
  - Advocated for and implemented **fuzz test builds and tooling**, and led collaboration with the [OSS-Fuzz](#) and Nest DevInfra teams to **set up a continuous fuzzing pipeline**
  - Built novel tooling to automatically surface third-party dependencies in build infrastructure, to automate license compliance needs
  - As team expert, identified and fixed tricky Makefile and Blaze configuration issues

- Redesigned + optimized control algorithm libraries, for [launch of the latest Nest Thermostat](#)
  - Adapted thermostat algorithms and data structures for a low-memory environment
  - Spearheaded a [~20k LoC library of data structures and utilities](#), to support expressive OO design within tight memory constraints, including thorough unit tests for the library
- Established and advocated [C++ and testing best practices](#), primarily for team code, and also for volunteer company-wide readability reviews
- Improved Google-wide tooling for C++ analysis (AST-based dead code identification)

**Software Engineer** — *Access, Mountain View, Calif.*

Aug. 2016 – Feb. 2017

- Specified and implemented an [OpenConfig telemetry client](#) for MDU switch firmware
- Integrated IEEE 802.15.4g low-rate WPAN [wireless personal area network] and IEEE 802.15.5 [mesh networking](#) support into a tracking system for emergency first responders

**Software Engineering Intern** — *Corp Eng, New York, N.Y.*

June – Aug. 2015

- Designed and implemented an system for [generating, storing, and serving insights](#) (information about user behavioural patterns) in a productivity-tracking tool for Googlers
  - Formulated / implemented a pipeline to compute a user's most productive time for coding

**Engineering Practicum Intern** — *Geo, Mountain View, Calif.*

June – Sept. 2014

- Refactored >13k LoC of Java to improve + rework UI components in an internal GIS tool
- Rearchitected the UI model for validated data entry: from a blocking modal dialog, to one where errors are visually highlighted, and invalid data silently discarded if necessary

**Princeton University, Princeton, N.J.**

**Student Researcher** — *Dept. of Electrical Engineering*

Sept. 2014 – Aug. 2016

- Improved the optical performance of a [noninvasive mid-IR laser blood glucose sensor](#)
- Implemented [regression and ML methods](#) to improve accuracy of glucose level predictions
- Developed and characterized strategies to recognize and reject off-screen movements in a [far-field capacitive user-sensing system](#), as a pre-filter for a user gesture system
- Implemented data collection, ML, analysis, and visualization code in Python + Mathematica

**Qualcomm Inc.**

**Engineering Intern** — *Qualcomm Technologies Inc., San Diego, Calif.*

May – Aug. 2013

- Re-engineered and implemented a camera test framework for Windows RT
  - [Designed a language and wrote a parser](#) for driving the tablet camera testing framework
  - Wrote a file system-based IPC mechanism to work around Windows 8 RT limitations

**University of Saskatchewan, Saskatoon, Sask.**

**Student Researcher** — *Dept. of Computer Science*

Sept. 2009 – Feb. 2013

- Pioneered a novel [scalable mathematical framework](#) to identify unknown infectious persons and high-risk persons within a given contact network, using [probabilistic Bayesian inference](#)
- Formulated [mathematical methodologies](#) to incorporate contact pattern data and health history information into the process of inferring likely infection times for individuals

## Major skills

---

- C++ (C++17, C++20, [template metaprogramming](#), [GoogleTest](#), Abseil, Boost), C (C11), assembly ([ARMv7](#), ARMv8, x86/x64, MIPS)
- Recursive Make, [Blaze/Bazel](#), embedded toolchains; clang-tidy / Clang AST
- Linux, FreeRTOS, Poky, Docker; Bash, Fish shell; [Git](#), Mercurial, Perforce
- [Unit & integration testing](#), hardware test automation, [fuzzing](#), stress testing, TDD
- [Java](#), JVM bytecode, JUnit, reflection; Guava, Android, Swing; Kotlin, Scala; Jenkins CI
- Python (Pytest, Django); scientific computing: [Mathematica](#), MATLAB, NumPy, SciPy
- SQL (BigQuery), [protocol buffers](#) ([proto2](#), [proto3](#), nanopb), JSON
- HTML, CSS, XML, Markdown, Doxygen, LaTeX; TypeScript/JavaScript, PHP
- C#/.NET, Windows RT APIs, Windows UWP apps, VBA
- Analog / digital circuit design, layout, assembly, and debugging (Cadence, Verilog)
- Programming for microprocessors (Cypress PSoC, Arduino, PICAXE), FPGAs (Xilinx)
- Typography, low-level libraries for 2D graphics rendering and animation, UI design + testing
- [Design docs](#), [design scoping](#), agile / scrum; [Jira](#), Confluence, Gerrit, Bitbucket
- Fluent in non-technical Chinese, conversational intermediate French

## Honours & awards

---

- Tech impact award, Devices & Services, Google (for rewrite of thermostat algo code)
- Fuzzys award, Google, 2021 (for building the fuzz test framework for thermostat code)
- Phi Beta Kappa, Princeton Univ., 2016 (scholastic standing, top decile of graduates)
- Sigma Xi, Princeton Univ., 2016 (research achievement & potential)
- Tau Beta Pi, Princeton Univ., 2015 (engineering students in top eighth of class)

## Selected publications

---

A. Werth, S. Liakat, **A. Dong**, C. M. Woods, and C. F. Gmachl. "Implementation of an integrating sphere for the enhancement of noninvasive glucose detection using quantum cascade laser spectroscopy". In *Applied Physics B*, 2018.

**A. Dong** and N. Osgood. "The limits of modeling continuous individual dynamics with discrete aggregate measures: A cautionary tale from immuno-epidemiological dynamics". In *Proceedings of the 29th International Conference of the System Dynamics Society*, 2011.