Nicholas (Nic) McDonald

n.mcdonald83@gmail.com
www.nicm.dev

Expertise:

I am a computer architecture research scientist and software/hardware engineer. I've designed systems of all sizes ranging from small embedded systems to exascale supercomputers. My Ph.D. research was enabling data centers with supercomputer networks through a novel hardware design and software programming model. I'm currently a hardware/software engineer at Google working on designing next generation high-performance network systems. Previously I worked at Hewlett Packard Labs where I designed high-performance network ASICs, network protocols, adaptive routing algorithms, simulation infrastructures, distributed system programming models, and submitted over 20 patent applications as lead inventor.

Education:

Stanford University

- *Ph.D. Electrical Engineering, 2016, GPA 3.90*
    - *Advisor: William J. Dally*

University of Utah

- *M.S. Electrical and Computer Engineering, May 2012, GPA 3.85*
    - *Advisor: Al Davis*
- *B.S. Computer Engineering, December 2010, GPA 3.81*

Employment:

Google - Senior Software Engineer 2019-present

- Developing next generation data center network topologies, routing algorithms, protocols, and switching architectures.
- Guiding the architecture of future network hardware and software systems.

Hewlett Packard Labs - Research Scientist 2016-2018

- Developed architectures for next-generation HPC systems based on emerging technologies such as photonics, non-volatile memories, memory-semantic network protocols, etc.
- Designed the routing and congestion management architecture of the Gen-Z protocol to support HPC systems with efficiency at extreme scales and high performance.
- Designed a hierarchical high-radix router architecture for use in next-generation HPC systems.
- Designed new routing algorithms to support fine-grained incremental adaptive routing.

Google - Network Research Intern 2014

- Designed, prototyped, and benchmarked next generation network topologies, congestion control algorithms, and load balancing mechanisms.
- Developed a Python-based software package for interacting in parallel with numerous hosts with focus on error tolerability and scripting productivity.

HP Labs - Research Associate 2013-2014, 2015

- Designed multiple microarchitectures for high-radix switches enabled by photonic I/Os with emphasis on scalability of both monolithic and hierarchical switch designs.
- Developed a cycle-accurate discrete event network simulator specializing in large network simulations, flexible endpoint modeling, and alternative microarchitecture modeling.

L-3 Communications - Digital Hardware Design Engineer 2010-2012

- Developed digital processing architectures for encryption, networking, waveform data processing, and network synchronization.
- Designed systems with strict requirements for high-performance, low-power, and minimal area.
- All designs were meticulously scrutinized for high-reliability and signal integrity.

L-3 Communications - Co-op Systems Engineer 2008-2010

- Worked in systems engineering to support the Global Hawk UAV program.
- Designed digital test circuitry to aid product interfacing and testing.
- Developed acceptance test procedures for L-3 products.
- Performed many debugging investigations and tests for Global Hawk systems.

Publications:

"Practical and Efficient Incremental Adaptive Routing for HyperX Networks", The International Conference for High Performance Computing, Networking, Storage, and Analysis (SC) 2019.

"SuperSim: Extensible Flit-Level Simulation of Large-Scale Interconnection Networks", The International Symposium on Performance Analysis of Systems and Software (ISPASS) 2018.

"High-Performance Service-Oriented Computing", Ph.D. dissertation, Stanford University, 2016.

Patents:

"Arbitrating Data Packets", 2017, US Patent 9,847,949
"Calculating Times to Live for Transaction Requests", 2019, US Patent 10,355,978
20+ patents pending, filed 2016-2019

Open-Source Projects:

SuperSim - Extensible flit-level network simulation, www.github.com/nicmcd/supersim
C++, 40k lines, 350 files
libdes - Parallel discrete event simulation framework, www.github.com/nicmcd/libdes
C++, 5k lines, 42 files
TaskRun - Parallel task management and execution, www.github.com/nicmcd/taskrun
Python, 6k lines, 44 files
ParaMgmt - Scriptable parallel server interaction, www.github.com/google/paramgmt
Python, 1k lines, 3 files

Academic Research:

A Distributed System Architecture for Secure High Performance Computing (Ph.D. thesis)

- *Developed a high performance distributed system architecture for modern large-scale applications that provides inherent network level isolation and security.*
- *Designed a network interface controller that supports this architecture by enforcing its policies efficiently in hardware and removing the need for complex software based solutions.*
- *Developed a custom logic simulator for the network interface controller and used it to show that it only increases message latency by 35-65 ns.*

Accelerator Multi-Threading (Multi-Core Architecture Class Project)

- *Developed a novel accelerator interface architecture for fine-grained system multi-threading.*
- *Implemented the design on a Xilinx Zynq platform and showed the dramatic benefits of hardware-based accelerator multi-threading versus software-based solutions.*

Arbitration Structures for High-Radix Routers (M.S. Research Project)

- *Implementation and investigation of the energy and delay scalability of parallel prefix arbitration structures for high-radix routers.*
- *Developed radix independent HDL and tested across multiple radices and technology libraries.*

Attack on SSL and the Public Key Infrastructure (Network Security Class Project)

- *Developed and implemented a system for covertly installing an evil certificate-authority (C.A.) in a victim's web browser.*
- *Developed software to reroute web traffic and use the evil C.A. to decrypt all transferred web data.*

Precise Time Synchronization in 804.15.4 Wireless Networks (Communications Class Project)

- *Developed digital hardware to precisely control time synchronization between nodes in an 802.15.4 wireless network.*

Fly-by-wire System for an Autonomous Unmanned Helicopter (B.S. Senior Project)

- *Developed a multi-processor system for sensor data acquisition, sensor fusion, and autonomous flight control using PID feedback.*
- *Created software for viewing and controlling the flight.*

Tiny 32-bit RISC Architecture (Processor Design Class Project)

- *Designed a programmer/compiler friendly instruction set architecture (I.S.A.).*
- *Created a fully featured assembler for machine code generation.*
- *Implemented and tested the processor design in an FPGA.*
- *Processor design includes a priority encoded interrupt system, variable sized instructions, SDRAM interface, and simple programming interface for precise hardware control.*