previous

Pat Pannuto <ppannuto@umich.edu>

The themes of my previous work are the development of new, readily available, user-friendly *platforms* [2, 1] and exploring *non-traditional domains* [3].

M-ulator.[1] The M-ulator is a *platform* originally developed as part of the Michigan Micro Mote (M3) initiative. The **intellectual merit** of the platform lies in its novel "subscription"-based pseudo-hardware architecture. The CPU is populated by instruction-handling functions that register a bitmask of instructions they know how to decode and execute. Similarly, the memory map is populated by registered callbacks who claim addresses a given peripheral wishes to associate with. This design enables such extensibility that M-ulator is both a si*mulator* and an e*mulator*. With the connection of an external hardware device, peripheral handling functions of the virtual processor can set I/O lines and interface with real physical hardware.

This innovation drives M-ulator's first **broader impact**. As modern engineering transistions to the challenging, fast-turn world of concurrent hardware-software co-design, it is extremely difficult to develop for a non-existant system and to debug the subsequent system as it comes into being. M-ulator allows for dynamic switching between real hardware and software emulation of hardware for any piece of a new system. This creates a new development paradigm, where software can be developed for a system in parallel with the creation of the system hardware. System integrators start from a "known-good" set of emulated hardware and slowly phase in potentially unreliable hardware components. Used currently with the M3 project, M-ulator has exposed both buggy hardware and buggy software while allowing for concurrent work in both arenas, rapidly accelerating the development time of cutting edge technology. As a resonably performant simulator with a replay engine for both synchronous and asychnronous events, M-ulator can be useful for debugging a diverse array of applications for the whole software community.

The M-ulator has proven its **broader impact** as a platform beyond the research community as well. As an instructor for the embedded systems course, I built a project around the M-ulator that student course evalutions declared the "most enjoyable assignment I've done so far at UofM ... I learned a ton going through these exercises. I highly recommend that this is kept in the future." Unlike traditional course projects, which are generally individual or in small teams of two to four, the students were faced with a single class-wide project. Presented with a bare M-ulator skeleton, students had to build functions to implement all of the CPU instructions (ADD, MUL, LDR, STR, etc), develop a C library to bootstrap the processor and access its peripherals, and write blink and echo programs that ran correctly. As a class, the students learned firsthand the challenges faced in real-world companies where time to market is critical and parallel software/hardware bringup is the norm. The M-ulator platform, external hardware, and course materials are all open source and available. Portions of the course material have already migrated to the University of Utah.

Michigan Micro Mote (M3). The M3 is ongoing work seeking to build a general-purpose, millimeter-scale computing *platform*. A collaboartive effort between four faculty and over

[1] Available: `https://github.com/ppannuto/M-ulator`

a dozen students, this ambitious project pushes the boundaries of science and encompasses a diverse array of **intellectual merit**. My personal contributions to the M3 project, in addition to M-ulator, have focused on the M3's novel low-power I2C variant [4]. No existing bus protocol was well-suited to the unique challenges presented by millimeter-scale computing. The developed protocol solved the ultra-low power and wire count requirements of M3, while appearing nearly identical on-wire to the original I2C protocol, easing adoption by those experienced with the standard.

The M3 project seeks to realize the Smart Dust vision, achieving revolutionary degrees of **broader impact**. Much in the way the general purpose i386 processor opened a world of computing, M3 is poised to create a world of micro-scale computing, featuring *in vivo* sensors, micro-scale robotics, and a world beyond what we can imagine today. I feel priviledged to a part of the team enabling this vision. Dissemination is a critical component of the M3 initiative, and in my IP+SN '12 presentation I introduced the sensor network community to the new platform, gathering contact information and project ideas from groups interested in the first round of hardware distribution.

Powerline Communications. The **intellectual merit** of this work was the exploration of a *non-traditional domain.* The inspiration came from a discussion of the design of AC-plug load meters. It seemed counterintuitive that a series of devices physically wired together would eschew that and instead form an ad-hoc wireless mesh network to relay data to a plugged-in base station. While powerline communications (PLC) is a known and effective technology in single electric domain of the average home, I sought to answer the question of whether and how a cohesive powerline network could be deployed in an electrically complicated commercial building. The work, presented at IP+SN '11, showed very good connectivity behind a single transformerer, but identified difficulties with whole-building connectivity [3].

The immediate **broader impact** of this work was an extended discussion with the IP+SN '11 workshop attendees exploring potential future research avenues around this new communications medium. Outside of the community, not all old buildings have Ethernet, and those that do may not have comprehensive coverage. Leveraging a building-wide PLC network, whole-building connectivity for older buildings can be realized for dramatically reduced cost. A particularly ambitious dream could eliminate the need for a separate Ethernet network entirely.

[1] Y.-S. Kuo, **P. Pannuto**, T. Schmid, and P. Dutta. Reconfiguring the software radio to improve power, price, and portability. In *Proceedings of the 10th ACM Conference on Embedded Networked Sensor Systems*, SenSys '12, New York, NY, USA, 2012. ACM.

[2] Y. Lee, S. Bang, I. Lee, Y. Kim, G. Kim, M. H. Ghaed, **P. Pannuto**, P. Dutta, D. Sylvester, and D. Blaauw. A modular $1mm^3$ die-stacked sensing platform with low power i2c inter-die communication and multi-modal energy harvesting. In *IEEE Journal of Solid-State Circuits*. To Appear, 2012.

[3] **P. Pannuto** and P. Dutta. Exploring powerline networking for the smart building. In *Extending the Internet to Low power and Lossy Networks*, IP+SN '11, April 2011.

[4] **P. Pannuto**, Y. Lee, B. Kempke, D. Sylvester, D. Blaauw, and P. Dutta. Ultra-constrained sensor platform interfacing. In *Proceedings of the 11th international conference on Information Processing in Sensor Networks*, IPSN '12, pages 147–148, New York, NY, USA, 2012. ACM.