

Promotion Dossier

Mark L. Chang
mark.chang@olin.edu

September 2010

This document contains bookmarks and hyperlinks that are active when read with a PDF viewer.

Contents

1 CV	5
2 Research Statement	15
2.1 Research Philosophy	15
2.2 Reconfigurable Computing	15
2.3 Mobile, Social, and Ubiquitous Computing	15
2.3.1 Mobile and ubiquitous computing: localization	16
2.3.2 Social and human computing: multitouch interfaces	16
2.4 Engineering Education	17
2.4.1 Embedded systems in education	17
2.4.2 Smartphones in education	17
2.4.3 SCOPE	18
2.5 Research Impact	18
2.6 Consulting Activities	19
3 Teaching Statement	21
3.1 A Community of Creators	21
3.1.1 SCOPE	21
3.1.2 Mixed Analog-Digital VLSI (MADVLSI)	22
3.1.3 Mobile Application Development	22
3.1.4 Computer Security	23
3.2 Commitment to Teaching	23
4 Service Statement	25
4.1 Service to the College	25
4.1.1 Intercollegiate relations	25
4.1.2 Curriculum reform	25
4.1.3 Honor board	25
4.1.4 Resident scholar	26
4.1.5 List of Committees and Department Service	26
4.2 Service to the Profession	26
5 Other Contributions	27
5.1 Advanced Computing Laboratory	27
5.2 External Relations	27
5.2.1 Mobile Application Development	27
5.2.2 Computer Architecture	28
5.2.3 Seminar series	28
5.3 Faculty Pub Night	28
6 Record of Intellectual Vitality Achievements	29
6.1 Publications in Print	29
6.2 Other Intellectual Vitality Achievements	30
6.2.1 Conference posters	30
6.2.2 Panel and workshops	30
6.2.3 Invited talks	30
6.3 Grants Submitted	30
6.4 Reprints	31
6.4.1 Movement Detection for Power-Efficient Smartphone WLAN Localization	32
6.4.2 Interactionless Calendar-Based Training for 802.11 Localization	42

6.4.3	Work in Progress: synthesizing design, engineering, and entrepreneurship through a course in mobile application development	51
6.4.4	Work in Progress: Impact of early design instruction on capstone experiences	53
6.4.5	A Long-Duration Study of User-Trained 802.11 Localization	55
6.4.6	A Parameterized Stereo Vision Core for FPGAs	72
6.4.7	A Semi-Automatic Approach for Project Assignment in a Capstone Course	76
6.4.8	A Blank Slate: Creating a New Senior Engineering Capstone Experience	87
6.4.9	Device Architecture	100
6.4.10	Précis: A Design-Time Precision Analysis Tool	128
6.4.11	Automated Least-Significant Bit Datapath Optimization for FPGAs	141
6.4.12	Précis: A Design-Time Precision Analysis Tool	150
6.4.13	Adaptive Computing in NASA Multi-Spectral Image Processing	160
6.4.14	REU Site: Engineering Education Research: Understanding and Improving Student Experiences	170
6.4.15	Collaborative Research TUES-Type 1: Fostering Student Learning Continuity Employing a Personal Active Learning Platform	192
6.4.16	Olin College Innovation Fund Proposal: Network Hacking	214
6.4.17	Adaptive Rough Terrain Navigation on a Legged Robot Platform	217
6.4.18	Ubiquitous Computing for Carepartner Relief Through Patient Independence	224
7	Record of Teaching Materials	229
7.1	Computer Architecture	229
7.1.1	Syllabus	230
7.1.2	HW2	233
7.1.3	MP3	234
7.2	Digital VLSI	237
7.2.1	Syllabus	238
7.2.2	HW0	240
7.2.3	MP1	241
7.3	Mixed Analog-Digital VLSI	245
7.3.1	Syllabus	246
7.3.2	MP0	248
7.3.3	MP1	249
7.4	Embedded Systems Design	252
7.4.1	Syllabus	253
7.4.2	Overview	255
7.4.3	MP0	259
7.5	Mobile Application Development	264
7.5.1	Syllabus	265
7.5.2	HW1	267
7.5.3	HW6	270
7.5.4	Contest	271
8	Record of Service Achievements	273
8.1	Certificate in Engineering Studies	274
8.2	Exposure Analysis Preliminary Report	279
8.3	Proposal for 4-1 Program for Wellesley and Babson Students	284
8.4	The Task Force on the Sophomore and Junior Years Final Report	288
9	List of Potential Reviewers	301
9.1	Scott Hauck	301
9.2	Miriam Leeser	301
9.3	Seth Teller	301
9.4	Dr. Jeffrey Hightower	302

9.5 Lukas Kencl	302
9.6 Adele Wolfson	302

Mark L. Chang

CONTACT INFORMATION

Franklin W. Olin College of Engineering
1000 Olin Way
Needham, MA 02492
USA

Voice: 781.292.2559
Fax: 781.292.2508
Email: mark.chang@olin.edu
Web: <http://faculty.olin.edu/~mchang>

EDUCATION

1. Ph.D. in Electrical Engineering, University of Washington, Seattle, WA, 2004.
Thesis: *Variable Precision Analysis for FPGA Synthesis*
Adviser: Scott Hauck
2. M.S., Electrical and Computer Engineering, Northwestern University, Evanston, IL, 2000.
Thesis: *Adaptive Computing in NASA Multi-Spectral Image Processing*
Adviser: Scott Hauck
3. B.S. with University and Departmental Honors, Electrical and Computer Engineering, The Johns Hopkins University, Baltimore, MD, 1997.

RESEARCH INTERESTS

FPGA Architectures, Applications, and Tools; Reconfigurable Computing; Ubiquitous Computing; Computer Architecture; VLSI Design; Human Computer Interaction; Engineering Education.

AWARDS

1. Intel Corporation: 2002-2003 Intel Foundation Graduate Fellowship
2. University of Washington: Outstanding Graduate Research Assistant (2002), Nominated for the Yang Research Award (2002)
3. Northwestern University: Royal E. Cabell Fellowship (1997), ECE Department Best Teaching Assistant Honorable Mention (1998)
4. Johns Hopkins University: IEEE student chapter president (1996), Eta Kappa Nu chapter president (1996, 1997), Tau Beta Pi, Dean's List, Electrical and Computer Engineering Chair Award
5. National Merit Scholar, National Computer Systems Merit Scholarship

EMPLOYMENT

1. Resident Scholar Franklin W. Olin College of Engineering
Needham, MA 08/2005 - Present
Living on campus as an academic resource for students at Olin College. Responsible for academic advising and intellectually stimulating activities in the residence halls.
2. Assistant Professor Franklin W. Olin College of Engineering
Needham, MA 08/2004 - Present
Electrical and Computer Engineering faculty member.
3. Graduate Research Assistant University of Washington
Seattle, WA 07/2000 - 07/2004
Developed variable precision design tools for FPGAs.
4. Software Developer Quicksilver Technologies, Inc.
Seattle, WA 07/2001 - 10/2001
Assisted design and development of software development tools for Quicksilver's reconfigurable hardware.

5. Graduate Research Assistant Northwestern University
Evanston, IL 09/1997 - 06/2000
Developed FPGA implementations of NASA image processing applications.
6. Customer service operator National Computer Systems
Iowa City, IA 06/1997 - 08/1997
Phone operator for the Department of Education.
7. Undergraduate Research Assistant Johns Hopkins University
Baltimore, MD 10/1994 - 06/1997
Worked on a portable high performance linear algebra library. Investigated the IEEE-1394 draft standard in conjunction with the JHU Applied Physics Laboratory for a 1394-based spacecraft bus design.
8. Assistant System Administrator Johns Hopkins University
Baltimore, MD 03/1995 - 06/1997
Maintained a network of servers and workstations for the Center for Language and Speech Processing.
9. Webmaster Johns Hopkins University
Baltimore, MD 05/1995 - 01/1997
Designed and maintained a web site for the Maryland Space Grant Consortium.
10. Embedded Software Developer Anton-Paar, GmbH
Graz, Austria 06/1996 - 07/1996
Participated in a cooperative internship with the Technical University of Graz, Austria. Developed embedded software for use in concentration determination instruments.
11. Programmer and technician Products Unlimited, Corp.
Iowa City, IA Summer 1989 - 1994
Set up and maintained a network of PCs for a small engineering office. Developed computer-aided testing facilities using IEEE-488 instruments and hardware.

CONSULTING

1. Applications Technology, Inc. 06/2009-present
Research supporting machine language translation software and systems.
2. The MITRE Corporation 08/2007
Researcher investigating 3D virtual worlds for collaboration.
3. NetFrameworks, Inc. & Applied Minds, Inc. 07/2001 - 09/2001
Primary software developer for proprietary groupware system.
4. Hunter Benefits Consulting Group 09/2000
Lead software developer.
5. HumaniTree.com, LLC 12/1998 - 03/1999
Web developer and Java programmer.

TEACHING

1. Franklin W. Olin College of Engineering, Needham, MA

Semester	Course
Spring 2010	ENGR 3499A: Mobile Application Development ENGR 3427: Mixed Analog-Digital VLSI II ENGR 4190: SCOPE (Linden Lab, Microsoft FUSE, Apple) MythTV Co-Curricular
Fall 2009	ENGR 3410: Computer Architecture ENGR 3426: Mixed Analog-Digital VLSI I ENGR 4190: SCOPE (Linden Lab, Microsoft FUSE)
Spring 2009	ENGR 3499A: Principles of Intelligent Systems Engineering (No course #): Mobile Application Development ENGR 3427: Mixed Analog-Digital VLSI II ENGR 4190: SCOPE (MITRE) Social Justice Reading Group
Fall 2008	ENGR 3410: Computer Architecture ENGR 3426: Mixed Analog-Digital VLSI I ENGR 4190: SCOPE (MITRE) Physical Security Systems Co-Curricular Social Justice Reading Group
Spring 2008	ENGR 3427: Mixed Analog-Digital VLSI II ENGR 3499A: Advanced Digital Systems ENGR 4190: SCOPE (MITRE, Nortel Networks) Social Justice Reading Group
Fall 2007	ENGR 3410: Computer Architecture ENGR 3426: Mixed Analog-Digital VLSI I ENGR 4190: SCOPE (MITRE, Nortel Networks) Social Justice Reading Group
Spring 2007	ENGR 3430: Digital VLSI Design ENGR 3499A: Embedded Systems Design ENGR 4190: SCOPE (IBM Research) Social Justice Reading Group
Fall 2006	ENGR 3410: Computer Architecture ENGR 4190: SCOPE (IBM Research) Social Justice Reading Group
Spring 2006	ENGR 3430: Digital VLSI Design ENGR 4190: SCOPE (John Deere, Motorola Labs) Olin Works Co-Curricular Social Justice Reading Group
Fall 2005	ENGR 3410: Computer Architecture ENGR 4190: SCOPE (John Deere, Motorola Labs) Olin Works Co-Curricular Social Justice Reading Group
Spring 2005	ENGR 3430: Digital VLSI Design Green Engineering Co-Curricular
Fall 2004	ENGR 3410: Computer Architecture

2. Yonsei University, Seoul, Korea

Principles of Engineering, Yonsei University International Summer School, Summer 2008.

3. University of Washington, Seattle, WA

EE 471: Computer Design and Organization. Instructor, Winter 2003.

Overall class evaluation rating 4.13/5.0.

(<http://www.washington.edu/cec/e/EE471A4003.html>).

4. Northwestern University, Evanston, IL

B01: Introduction to Digital Logic Design. Instructor, Summer 1999.

Overall class evaluation rating 5.56/6.0.

C91: VLSI Systems Design. Teaching Assistant, Winter 1999.

C92: VLSI Systems Design Projects. Teaching Assistant, Spring 1998.

STUDENT ADVISING

Credit-bearing advising activities are grouped by academic year below. As of Spring 2010, I have advised 329 credit hours of research, independent study, Olin self study, and Passionate Pursuit activities.

Research Students

Academic Year	Project	Student
2009-2010	Cybersecurity and the hacker curriculum Tangible and large format interactable displays WiFi localization Twitter social behaviors GPU processing FPGA Applications Parking lot car detection	Noah Tye Jacob Getto Andrew Barry Noah Tye Ilari Shafer Greg Marra Ilari Shafer Ben Fisher James Getzendanner Andrew Barry
2008-2009	Low-cost, high-speed FPGA interfaces Stereo Vision on FPGA	John Morgan Christopher Nissman Stephen Longfield
2007-2008	Low-cost, high-speed FPGA interfaces Stereo Vision on FPGA Alzheimer's carepartner relief technologies	John Morgan Stephen Longfield Alex Davis
2006-2007	Multi-touch user interfaces Low-cost, high-speed FPGA interfaces Stereo Vision on FPGA Alternative input devices	Jonathan Tse Anthony Roldan Chris Stone Olek Lorenc John Morgan Stephen Longfield George Harris Evan Morikawa Benjamin Hayden Greg Marra Rebecca Scholl Jon Cass
2005-2006	Ubiquitous computing for seniors FPGA applications research FPGA acceleration of computational fluid dynamics FPGA-based neural networks	Daniel Lindquist Zachary Brock Brian Shih Eric Gallimore Nathaniel Smith Eric VanWyk
2004-2005	Evolvable hardware on FPGAs	Joy Poisel Christopher Murphy

Independent Study, OSS, and Passionate Pursuits

Academic Year	Type	Project	Student
2009-2010	OSS	Novice GUI toolkits Second Life programming Video game engine programming Software development for CORe Multi-cultural cook-book	Ben Fisher Logan Dethrow Avinash Uttamchandani Roland Crosby Nina Sawhney
2008-2009	OSS	Software systems Game engine development	Chujiao Ma Andrew Price Erik Kennedy Roberto Santana Nik Wittenstein Zachary Kratzer Jeff Stanton
	IS	Turret automation Video game artificial intelligence iPhone development	Kelly Butcher Daniel Bathgate Zachary Brass Varun Mani Xavier Ziembra Jennifer Cross James Switzer
2007-2008	OSS	FPGA Systems Chess Home-brew dynamometer Software using Django Video game design Video game design Home-brew engine management Jujitsu Multi-touch software frameworks 3D Rendering Automobile competition	Anthony Roldan Christopher Stone Gabriel Greely Benjamin Hayden Kent Munson Andrew Kalcic Eamon Doyle Hans Borchardt Samuel Freilich Jeffrey DeCew Joseph Funke
2006-2007	OSS	Traffic monitoring systems Embedded art project Analog and Digital VLSI Mobile user interface design Visualization and clustering FPGA hardware for blob detection	Jeffrey Glickman Nathaniel Smith Benjamin Hill Sean McBride Brian Shih Cody Wheeland Kelcy Adamec Leslie Velez
	PP	Bartending	
2005-2006	OSS	Mobile phone marketplace Recommender systems Software engineering Advanced digital embedded design Ubiquitous computing Machine vision Reconfigurable Computing History of Film and Technology	Michael Crayton Sean Munson Drew Harry Christopher Murphy Daniel Lindquist Sarah Leavitt Michael Foss Thomas Kochem Kevin Tostado
	PP	History of Film and Technology Acrobatic plane construction	Adam Bry
	Other	Advised student startup company	Matthew Colyer
2004-2005	PP	Mandarin Chinese Korean Video Game Design	Christopher Doyle Sutee Dee Katherine Kim Matthew Colyer Dean Dieker Brendan Doms Sean McBride Brian Shih

Student Outcomes

Providing training outside the classroom as researchers and engineers is something I am very proud of. The following is a select list of research students and students with whom I worked closely, and their current post-Olin endeavors.

Student	Postgraduate work
Andrew Barry	MIT PhD
Christopher Murphy	MIT PhD
Drew Harry	MIT PhD
Ilari Shafer	Carnegie Mellon PhD
Jennifer Cross	Carnegie Mellon PhD
Stephen Longfield	Cornell PhD
Jonathan Tse	Cornell PhD
Benjamin Hill	Cornell PhD
Sean Munson	University of Michigan PhD
Connor Skye Riley	University of California Berkeley MS
Benjamin Hayden	Google
Greg Marra	Google
Brian Shih	Google
Nathaniel Smith	Google
George Harris	Microsoft
Ben Fisher	Microsoft
Ellen Chisa	Microsoft
Daniel Lindquist	Yahoo, Kellogg School of Management MBA
Alex Davis	Yelp

PUBLICATIONS

Name Order Convention

For publications listed below, students are generally listed first, in descending order of contribution. Next, faculty members are listed in descending order of contribution. Any papers that do not follow this form are noted and the contribution clarified.

Key to author list

Bold Italic: Mark L. Chang

Bold: Student

No Marking: Non-student collaborator

Italics: Mark L. Chang's graduate adviser

Contribution

For any paper with collaborators in the author list, the approximate contribution of M. Chang is listed, broken down into four categories: Concept, Implementation/Data Gathering, Analysis, and Writing (including editing).

As Assistant Professor at Olin College

1. **Ilari Shafer, *Mark L. Chang***, "Movement Detection for Power-Efficient Smartphone WLAN Localization", *13th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, October 2010.
Concept: 50%, Implementation/Data Gathering: 0%, Analysis: 25%, Writing: 10%
2. **Andrew Barry, Noah Tye, *Mark L. Chang***, "Interactionless Calendar-Based Training for 802.11 Localization," *The 7th IEEE International Conference on Mobile Ad-hoc and Sensor Systems*, November 2010.
Concept: 50%, Implementation/Data Gathering: 0%, Analysis: 25%, Writing: 25%

3. **Mark L. Chang**, “Work in Progress: synthesizing design, engineering, and entrepreneurship through a course in mobile application development”, *Frontiers in Education Conference*, 2010. Concept: 100%, Implementation/Data Gathering: 100%, Analysis: 100%, Writing: 100%
4. Jessica Townsend, **Mark L. Chang** “Work in Progress: Impact of early design instruction on capstone experiences”, *Frontiers in Education Conference*, 2010. Concept: 50%, Implementation/Data Gathering: 50%, Analysis: 50%, Writing: 50%
5. Andrew Barry, Benjamin Fisher, **Mark L. Chang**, “A Long-Duration Study of User-Trained 802.11 Localization,” *Proceedings of the Second ACM International Workshop on Mobile Entity Localization and Tracking in GPS-less Environments*, September 2009. Awarded best paper and best presentation. Concept: 10%, Implementation/Data Gathering: 0%, Analysis: 15%, Writing: 30%
6. Stephen Longfield, Jr., **Mark L. Chang**, “A Parameterized Stereo Vision Core for FPGAs”, (**Short Paper**) *IEEE Symposium on Field-Programmable Custom Computing Machines*, April 2009. Concept: 75%, Implementation/Data Gathering: 35%, Analysis: 75%, Writing: 100%
7. **Mark L. Chang**, Allen Downey, “A Semi-Automatic Approach for Project Assignment in a Capstone Course”, *Proceedings of the American Society for Engineering Education Annual Conference*, June, 2008. Concept: 50%, Implementation/Data Gathering: 50%, Analysis: 50%, Writing: 50% Author ordering is alphabetical
8. **Mark L. Chang**, Jessica Townsend, “A Blank Slate: Creating a New Senior Engineering Capstone Experience”, *Proceedings of the American Society for Engineering Education Annual Conference*, June, 2008. Concept: 50%, Implementation/Data Gathering: 50%, Analysis: 50%, Writing: 50% Author ordering is alphabetical
9. **Mark L. Chang**, “Device Architecture”, in *Reconfigurable Computing: The Theory and Practice of FPGA-Based Computation*; Scott Hauck, Andre DeHon, Editors; Morgan Kaufmann/Elsevier, 2008, pp. 3-27. Concept: 100%, Implementation/Data Gathering: 100%, Analysis: 100%, Writing: 100%
10. **Mark L. Chang**, Scott Hauck, “Précis: A Design-Time Precision Analysis Tool”, *IEEE Design and Test of Computers*, Vol. 22, No. 4, pp. 349-361, July-August 2005. Concept: 100%, Implementation/Data Gathering: 100%, Analysis: 100%, Writing: 100%

Prior to Employment at Olin College

1. **Mark L. Chang**, *Variable Precision Analysis for FPGA Synthesis*, Ph.D. Dissertation, University of Washington, Department of Electrical Engineering, 2004.
2. **Mark L. Chang**, Scott Hauck, “Automated Least-Significant Bit Datapath Optimization for FPGAs”, *IEEE Symposium on Field-Programmable Custom Computing Machines*, April, 2004. Concept: 100%, Implementation/Data Gathering: 100%, Analysis: 100%, Writing: 100%
3. **Mark L. Chang**, Scott Hauck, “Variable Precision Analysis for FPGA Synthesis”, *Earth Science Technology Conference*, June, 2003. Concept: 100%, Implementation/Data Gathering: 100%, Analysis: 100%, Writing: 100%
4. **Mark L. Chang**, Scott Hauck, “Précis: A Design-Time Precision Analysis Tool”, *Earth Science Technology Conference*, June, 2002. Concept: 100%, Implementation/Data Gathering: 100%, Analysis: 100%, Writing: 100%
5. **Mark L. Chang**, Scott Hauck, “Précis: A Design-Time Precision Analysis Tool”, *IEEE Symposium on Field-Programmable Custom Computing Machines*, pp. 229–238, 2002. Concept: 100%, Implementation/Data Gathering: 100%, Analysis: 100%, Writing: 100%

6. **Mark L. Chang**, *Adaptive Computing in NASA Multi-Spectral Image Processing*, M.S. thesis, Northwestern University, Dept. of ECE, December, 1999.
Concept: 100%, Implementation/Data Gathering: 100%, Analysis: 100%, Writing: 100%
7. **Mark L. Chang**, Scott Hauck, “Adaptive Computing in NASA Multi-Spectral Image Processing”, *Military and Aerospace Applications of Programmable Devices and Technologies International Conference*, 1999.
Concept: 100%, Implementation/Data Gathering: 100%, Analysis: 100%, Writing: 100%
8. P. Banerjee, A. Choudhary, S. Hauck, N. Shenoy, C. Bachmann, **Mark L. Chang**, M. Haldar, P. Joisha, A. Jones, A. Kanhare, A. Nayak, S. Periyacheri, M. Walkden, “MATCH: A MATLAB Compiler for Adaptive Computing Systems”, *Northwestern University Department of Electrical and Computer Engineering Technical Report CPDC-TR-9908-013*, 1999.
Concept: 0%, Implementation/Data Gathering: 10%, Analysis: 10%, Writing: 10%

CONFERENCE POSTERS

1. Mihir Ravel, **Mark L. Chang**, Mark McDermott, Michael Morrow, Nikola Teslic, Mihajlo Katona, Jyotsna Bapat, “A Cross-Curriculum Open Design Platform Approach to Electronic and Computing Systems Education,” *IEEE International Conference on Microelectronic Systems Education*, July 2009.
Concept: 0%, Implementation/Data Gathering: 15%, Analysis: 15%, Writing: 15%
2. C. Murphy, D. Lindquist, A.M. Rynning, T. Cecil, S. Leavitt, **M.L. Chang**, “Low-Cost Stereo Vision on an FPGA”, IEEE Symposium on Field-Programmable Custom Computing Machines, 2007.
Concept: 50%, Implementation/Data Gathering: 0%, Analysis: 0%, Writing: 100%
3. **Mark L. Chang**, Scott Hauck, “Least-Significant Bit Optimization Techniques for FPGAs”, *ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*, February, 2004.
Concept: 100%, Implementation/Data Gathering: 100%, Analysis: 100%, Writing: 100%

PANELS AND WORKSHOPS

1. Hal Abelson, **Mark L. Chang**, Cyprien Lomas, David Wolber, “Google App Inventor for Android: Building mobile applications as a first computing experience” *Frontiers in Education Conference*, 2010. (*to be presented*)
2. Hal Abelson, **Mark L. Chang**, Eni Mustafaraj, Franklyn Turbak, “Mobile Phone Apps in CS0 Using App Inventor for Android”, *15th Annual Conference of the Northeast region of the Consortium for Computing Sciences in Colleges*, 2010.
3. Ellen Spertus, **Mark L. Chang**, Paul Gestwicki, David Wolber, “Novel Approaches to CS0 with App Inventor for Android”, *The 41st ACM Technical Symposium on Computer Science Education (SIGCSE)*, 2010.

INVITED TALKS

1. “Master of motivation: engaging students with smartphones and Google Android”, *Boston-area Advanced Technological Education Connections IT Futures Forum*, May 2010.
2. “Spinning the World Wide Web: How the Internet Really Works”, Needham Exchange Club presentation, October 2008.
3. “Olin College: Accrediting an Innovative Engineering Curriculum”, Yonsei University Engineering Seminar, August 2008.
4. “Olin College: Rethinking Engineering Education”, Microsoft Research, June 2008
5. “A Beginner’s Guide to Bad Engineering Presentations”, University of Hartford, November 2007.

6. “Spinning the World Wide Web: How the Internet Really Works”, Olin College Lecture Series, Needham Adult Education Program, October 2007.

GRANTS

Awarded or under review

1. *Under review:* Senior personnel on NSF proposal with co-PIs Debbie Chachra (Olin) and Lynn Stein (Olin), *REU Site: Engineering Education Research: Understanding and Improving Student Experiences.*
2. *Under review:* Co-PI on NSF TUES proposal with Gunar Schirner (Northeastern University), David Kaeli (Northeastern University), Mark Somerville (Olin), and Mihir Ravel (Olin), *Collaborative Research, TUES-Type 1: Fostering Student Learning Continuity Employing a Personal Active Learning Platform.*
3. Summer 2010: Olin Innovation Grant funding for “Network Hacking and Cyber Security” course development.
4. Summer 2010: Wellesley Tanner conference grant for work on extending the reach of the 10th anniversary Wellesley Tanner conference. Funding awarded to support equipment and Olin summer student Jacob Getto.

Not awarded

1. Spring 2010: MITRE Grant with Ozgur Eris (Olin) and Doug Phair (MITRE) for distributed design technologies and assessment methodologies (first round complete, not selected in final round).
2. Spring 2008: Alzheimer’s Foundation Early Technologies for Alzheimer’s Care proposal with Aaron Boxer (Olin) and Stephen Schiffman (Olin), *Ubiquitous Computing for Carepartner Relief Through Patient Independence.*
3. Spring 2008: DARPA proposal for BAA 07-46 with David Barrett (Olin) and Dr. Nahid Sidki (SAIC), *Portable Autonomous Communications Robotic Assistant: PACRAT.*
4. Spring 2005: HP Technology for Teaching grant

DONATIONS

1. Kevin and Marlene Getzendanner (P’10) funded a 5-year Olin Tuition Scholarship named in honor of Mark Chang as a result of Marks impact on the education of their son, James Getzendanner (’10)
2. Altera Corp., donation of FPGA hardware and software (2006-2008)
3. AndroidCentral.com, financial support for Mobile Application Development Course: \$3,000 (2009)
4. Applications Technology, Inc., financial support for Mobile Application Development Course: \$2,000 (2009)
5. CommonsWare, textbook for all students in Mobile Application Development Course (2009, 2010)
6. Hewlett-Packard, Inc., donation of workstations for VLSI teaching laboratory: \$23,824 (2005)
7. Microsoft, hardware and software for Mobile Application Development Course (2009)
8. Nokia Research Center, donation of handheld computing hardware (2008)
9. Palm, Inc., donation of textbook for all students in Mobile Application Development Course (2009)
10. Xilinx, Inc., donation of FPGA hardware and software: \$15,635 (2004), \$11,170 (per year, 2005-present)

PROFESSIONAL ACTIVITIES

Conference Steering Committee Member

1. Publicity Chair, IEEE Conference on Field-Programmable Custom Computing Machines, 2010
2. General co-chair, IEEE Workshop on Mobile Entity Localization and Tracking. Co-located with *The 7th IEEE International Conference on Mobile Ad-hoc and Sensor Systems*, November 2010

Program Committee Member

1. IEEE Microelectronic Systems Education Conference (2005, 2007, 2009, 2011)
2. IEEE International Conference on Field-Programmable Technology (2007, 2008, 2009, 2010)
3. IEEE International Conference on Field Programmable Logic and Applications (2005, 2006, 2007, 2008, 2009, 2010)
4. International Symposium on Applied Reconfigurable Computing (2008, 2009, 2010, 2011)
5. IEEE Conference on Field-Programmable Custom Computing Machines (2010)

Reviewer

1. ACM Symposium on User Interface Software and Technology (2010)
2. NSF ECCS Division BRIGE Program
3. IEE Proceedings of Computers & Digital Techniques
4. IEEE Transactions on Computers
5. IEEE Transactions on Education
6. IEEE Transactions on VLSI Systems
7. IEEE Transactions on Computer-Aided Design of Integrated Circuits & Systems
8. IEEE Transactions on Instrumentation & Measurement
9. ACM Transactions on Design Automation of Electronic Systems
10. IEEE International Symposium on Circuits and Systems
11. EURASIP Journal of Embedded Systems
12. ACM Transactions on Reconfigurable Technology and Systems
13. Journal of Real-Time Image Processing
14. International Journal of Reconfigurable Computing

COMMITTEES AND DEPARTMENT SERVICE

1. Ad hoc committee on curricular innovation, Fall 2009
2. SCOPE Director search, Fall 2009
3. Committee on Diversity and the Academic Experience, 2005 - 2007
4. Electrical and Computer Engineering Faculty Search committee, 2004, 2005, 2007
5. Electrical and Computer Engineering Program Group, 2004 - present
6. Faculty / IT committee, 2004 - 2007
7. Honor Board faculty representative, 2004 - 2009
8. Intercollegiate Relations Committee, 2007 - present (chair 2007 - present)
9. Task force on the 2nd and 3rd year curriculum, 2007 - 2008 (chair)
10. Wellesley Olin Working Group Committee, 2004 - 2007
11. Olin Certificate in Engineering Studies coordinator, 2007 - present

2 Research Statement

2.1 Research Philosophy

Over the past six years, I have developed a research approach at Olin that is rather different than if I were a faculty at a large research university. The mission of the college, its founding vision, and its core values put the undergraduate engineering student experience at the center of the work that we all do. Therefore, my research at Olin is a natural extension of my role as an educator, with the following objectives:

- excite Olin students about research
- train Olin students in the practice of research
- incorporate Olin students into all aspects of research practice
- provide experiences that augment the classroom in content and approach
- further the current state of the art

I am grateful to be at an institution whose values and foundational statements make *teaching* research just as important as *doing* research. Engaging our students in the process of discovery, experimentation, and scientific dissemination truly makes our educational continuum complete. This research philosophy has led to successful publication of student-authored peer-reviewed technical work as well as publication of my work as an educator in various engineering education venues.

To date, the research I have engaged in encompasses three computer-engineering areas: Reconfigurable Computing; Mobile, Social, and Ubiquitous Computing; and Engineering Education.

2.2 Reconfigurable Computing

My longest thread of research is one that dates back to my first days as a graduate student: utilizing field-programmable gate arrays (FPGAs) to both accelerate traditional computing systems and provide novel computing structures. The culmination of this work has been captured in a peer-reviewed journal article.¹ Following my Ph.D., I was invited to write the opening chapter to a book on reconfigurable computing.²

My first attempt to incorporate Olin students in my research endeavors came in the first year of SCOPE with our partner, John Deere. My experience with FPGAs helped me guide the Olin team on the development of a low-cost FPGA-based stereo vision platform for agricultural use. Their work was presented as a poster at the premiere FPGA-related conference, FCCM,³ and received very positive comments from my colleagues at the conference. All I spoke with were surprised that the work was completed by undergraduates. This work was extended by Stephen Longfield, Jr. and presented by him at FCCM in 2009.⁴

I have been recognized by the FPGA industry and community of researchers through donations of software and hardware for my research and courses, invitations to be a member of technical program committees of almost all major reconfigurable-computing-related conferences, and invitations to review for all reconfigurable-computing-related journals. I also served on the organizing committee of FCCM 2010 as publicity chair. This work has also led directly to the creation of several new courses in conjunction with Aaron Boxer, an industry expert in hardware design and reconfigurable computing in part-time residence at Olin.

2.3 Mobile, Social, and Ubiquitous Computing

I have found that with the design-centric curriculum at Olin, our students tend to find strong interest at the intersection of society and computing. With the increasing adoption of “smart” mobile devices such as the iPhone, the level of computational capacity and device availability is making computing truly more ubiquitous. Today, ubiquity plays an important role in the rapid adoption of many social media services such

¹Mark L. Chang, Scott Hauck, “Précis: A Design-Time Precision Analysis Tool”, *IEEE Design and Test of Computers*, Vol. 22, No. 4, pp. 349-361, July-August 2005. (section 6.4.10, p.128)

²Mark L. Chang, “Device Architecture”, in *Reconfigurable Computing: The Theory and Practice of FPGA-Based Computation*; Scott Hauck, Andre DeHon, Editors; Morgan Kaufmann/Elsevier, 2008, pp. 3-27. (section 6.4.9, p.100)

³C. Murphy, D. Lindquist, A.M. Rynning, T. Cecil, S. Leavitt, M.L. Chang, “Low-Cost Stereo Vision on an FPGA”, IEEE Symposium on Field-Programmable Custom Computing Machines, 2007.

⁴Stephen Longfield, Jr., Mark L. Chang, “A Parameterized Stereo Vision Core for FPGAs”, (Short Paper) *IEEE Symposium on Field-Programmable Custom Computing Machines*, April 2009. (section 6.4.6, p.72)

as Twitter and Facebook—services that are widely used by the current generation of “digital natives”, such as our students. I have a strong research interest in how mobile devices (and other novel human-computer interfaces) enable new forms of communication and computing. Working with mobile platforms has also led to interest in the educational impacts of incorporating modern devices into the classroom. More can be found in section 2.4.2.

2.3.1 Mobile and ubiquitous computing: localization

As individuals increasingly carry a sensor-laden, always connected, capable modern computer with them at all times in the form of a smartphone, researchers are increasingly asking, “how can location improve the user experience?” While outdoors, location is easily attained through the use of GPS. However, in GPS-denied environments, there are no ubiquitously-deployed technologies capable of determining your location other than cell tower triangulation. Therefore, researchers have been working on using wireless access points—commonplace in most buildings now—to triangulate location to the room level.

In the Spring of 2008, Benjamin Fisher ('10) and Andrew Barry ('10) approached me, as the faculty responsible for the Advanced Computing Lab (ACL), to host their personal project for indoor localization: Marauder's Map. After helping port their client application to the Macintosh platform, and about a year later, Andrew thought to publish his work. We started working in earnest in the Spring of 2009, and before the end of the semester, our paper had been accepted for publication.⁵

I think of this as a perfect example of undergraduate research—a student with the desire to learn about research and publication, and a true collaboration to achieve scholarly recognition. Andrew remarked many times that he was learning a lot about doing and publishing research working together. In advising Andrew, I wanted to recreate a graduate school atmosphere as much as possible. I expected Andrew to take the lead in technical implementation, while we would collaborate closely on the research direction and the writing of the paper. As Andrew's project did not start as a research project, much of my work was in helping Andrew frame a research question and design the experiments to answer that question. Only then did we have something that was publishable. In this way, Andrew was very much an apprentice to the practice of research.

Our participation in the Mobile Entity Localization and Tracking (MELT) workshop and the 2009 ACM Conference on Ubiquitous Computing (UbiComp) hatched several new directions for our research. This past year we recruited Ilari Shafer ('10) and Noah Tye ('13) to expand our research group, and successfully published two additional papers in selective peer-reviewed conferences.^{6,7} As with the initial work, I took much pleasure in guiding this group of students as they learned how to pose research questions, design experiments to answer those questions, and take and analyze their data to support the writing of a successful paper. All of these students will be traveling to present their work at their respective conferences.

Following from our successful work in the area of localization within ubiquitous computing, I was invited to be general co-chair the 2010 MELT workshop, to be held in conjunction with the 7th IEEE International Conference on Mobile Ad-hoc and Sensor Systems (IEEE MASS 2010) in San Francisco, CA. My work with Noah Tye in localization research continues.

2.3.2 Social and human computing: multitouch interfaces

In 2006, Jefferson Han⁸ demonstrated his novel multi-touch screen technology to the assembled audience at TED 2006.⁹ By 2008, he was recognized as one of Time Magazine's “Time 100”.¹⁰ This demonstration caught the imagination of our students, and many approached me—and continue to—to learn if we could

⁵ Andrew Barry, Benjamin Fisher, Mark L. Chang, “A Long-Duration Study of User-Trained 802.11 Localization,” *Proceedings of the Second ACM International Workshop on Mobile Entity Localization and Tracking in GPS-less Environments*, September 2009. Awarded best paper and best presentation. (section 6.4.5, p.55)

⁶ Andrew Barry, Noah Tye, Mark L. Chang, “Interactionless Calendar-Based Training for 802.11 Localization,” *The 7th IEEE International Conference on Mobile Ad-hoc and Sensor Systems*, November 2010. (section 6.4.2, p.42)

⁷ Ilari Shafer, Mark L. Chang, “Movement Detection for Power-Efficient Smartphone WLAN Localization”, *13th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, October 2010. (section 6.4.1, p.32)

⁸http://en.wikipedia.org/wiki/Jefferson_Han

⁹http://www.fastcompany.com/magazine/112/open_features-canttouchthis.html

¹⁰http://www.time.com/time/specials/2007/article/0,28804,1733748_1733754_1735325,00.html

build something like it at Olin. I sponsored a small group of students who would go on to build three versions of a multi-touch surface and wow their own audiences at Olin open houses, Candidate's Weekends, and several Olin Expo presentations.

For the tenth anniversary celebration of the Tanner Conference at Wellesley College, the Tanner organizing committee wanted to extend the reach and deepen the experience of Tanner for all attendees. With colleagues from the Wellesley Computer Science department, we put forth a proposal to build custom large-scale multi-touch surfaces to engage the audience in rich multimedia exhibits. The organizing committee and the Science Center at Wellesley College agreed to fund the build of the surface and the hiring of several summer students, including an Olin student, Jacob Getto, to work with us this summer. We are building the largest known table-oriented multi-touch surface, and will be working with Prof. Orit Shaer (Wellesley CS) on research in the fall to understand new and novel human-computer interaction modes made possible by such an expansive surface that can be used simultaneously by a large number of people.

2.4 Engineering Education

I believe as a consequence to so closely aligning my research to the experiences I provide in the classroom, much of my work has been recognized in both traditional discipline research publications as well as engineering education venues. I have also been invited to be on the technical program committee for the IEEE Microelectronic Systems Education Conference, and review for IEEE Transactions on Education.

2.4.1 Embedded systems in education

I have spent three semesters iterating on an embedded systems / digital systems course at Olin. I have worked with Aaron Boxer and Mihir Ravel, both visiting Olin, to deliver these courses. Mihir Ravel and I authored a poster about our experiences with several other faculty.¹¹ Following from this work, with colleagues at Olin and Northeastern University, we submitted an NSF education proposal that is currently under review.¹²

2.4.2 Smartphones in education

My work in utilizing the Google Android platform in my course, *Mobile Application Development*, has led to positive press for the course,^{13,14} very satisfied students, and an opportunity to discuss my experiences at the Frontiers in Education Conference this fall.¹⁵ With the widespread interest in the motivational power of mobile phones in the classroom, I was invited to speak about my educational experiences with high school and college students and teachers in the Boston area at the IT Futures Forum.¹⁶

As a result of working heavily with Android, I collaborated with Hal Abelson of MIT on an Android education-related research project aiming to explore the use of visual programming tools and Android to create a new first computing experience. I worked with Lynn Stein, Mark Sheldon, and several students (Gregory Marra ('10), Ilari Shafer ('10), and Michael Lintz ('11)) to pilot a study at Olin in the Fall of 2010. In addition to much recent public press coverage,^{17,18,19} I have had (and will have) the opportunity to speak

¹¹Mihir Ravel, Mark L. Chang, Mark McDermott, Michael Morrow, Nikola Teslic, Mihajlo Katona, Jyotsna Bapat, "A Cross-Curriculum Open Design Platform Approach to Electronic and Computing Systems Education," *IEEE International Conference on Microelectronic Systems Education*, July 2009

¹²NSF TUES proposal with Gunar Schirner (Northeastern University), David Kaeli (Northeastern University), Mark Somerville (Olin), and Mihir Ravel (Olin), *Collaborative Research, TUES-Type 1: Fostering Student Learning Continuity Employing a Personal Active Learning Platform*. (section 6.4.15, p.192)

¹³Lamere, Paul, "22 students + 10 days + Echo Nest == Awesome!" *Music Machinery Blog*, March 19, 2010. <http://musicmachinery.com/2010/03/19/22-students-10-days-echo-nest-awesome>

¹⁴"Here come the music apps", *the echo nest blog*, May 12, 2010. <http://blog.echonest.com/post/592373596/here-come-the-music-apps>

¹⁵Mark L. Chang, "Work in Progress: synthesizing design, engineering, and entrepreneurship through a course in mobile application development", *Frontiers in Education Conference*, 2010. (section 6.4.3, p.51)

¹⁶"Master of motivation: engaging students with smartphones and Google Android", *Boston-area Advanced Technological Education Connections IT Futures Forum*, May 2010.

¹⁷Lohr, Steve. "Google's Do-It-Yourself App Creation Software." *The New York Times*. July 12, 2010.

¹⁸Pogue, David. "D.I.Y. Tool for Apps Needs Work." *The New York Times*. August 11, 2010.

¹⁹Valentino-DeVries, Jennifer. "App Inventor Shows Another Difference Between Apple, Google." *The Wall Street Journal*. July 12, 2010.

about our research at many panels and workshops targeting educators.^{20,21,22}

2.4.3 SCOPE

Much of my time has been spent helping to create the SCOPE program. While heavily inspired by the Harvey Mudd Clinic program, I have spent a considerable amount of effort in refining the implementation for our community. Being involved since the beginning, I have learned a lot about running student teams with large budgets for a year long project. Not only has SCOPE been a direct inspiration for technical research publications, I have also had the opportunity to collaborate with some Olin faculty in disseminating our experiences at various education conferences.^{23,24,25} I really feel that our work in assessing and improving our capstone program is just beginning. My hope is to continue working on reforming SCOPE to make it a better experience for not just our students, but for any engineering capstone program.

2.5 Research Impact

I am proud to be able to list the following students as just some of those that I believe our outside-the-classroom work together helped them reach where they are currently.

Student	Postgraduate work
Andrew Barry	MIT PhD (in progress)
Christopher Murphy	MIT PhD (in progress)
Drew Harry	MIT PhD (in progress)
Ilari Shafer	Carnegie Mellon PhD (in progress)
Jennifer Cross	Carnegie Mellon PhD (in progress)
Stephen Longfield	Cornell PhD (in progress)
Jonathan Tse	Cornell PhD (in progress)
Benjamin Hill	Cornell PhD (in progress)
Sean Munson	University of Michigan PhD (in progress)
Connor Skye Riley	University of California Berkeley MS (complete)
Benjamin Hayden	Google
Greg Marra	Google
Brian Shih	Google
Nathaniel Smith	Google
George Harris	Microsoft
Ben Fisher	Microsoft
Ellen Chisa	Microsoft
Daniel Lindquist	Yahoo, Kellogg School of Management MBA (in progress)
Alex Davis	Yelp

My success as a research mentor has also been noticed by students. For example, Elena Oleynikova ('11) and Stanislaw Antol ('11) approached myself and Prof. David Barrett to be their research advisers for a proposal they were submitting.²⁶ Their proposal was funded by the Computing Research Association's

²⁰Hal Abelson, Mark L. Chang, Cyprien Lomas, David Wolber, "Google App Inventor for Android: Building mobile applications as a first computing experience" *Frontiers in Education Conference*, 2010. (to be presented)

²¹Hal Abelson, Mark L. Chang, Eni Mustafaraj, Franklyn Turbak, "Mobile Phone Apps in CS0 Using App Inventor for Android", *15th Annual Conference of the Northeast region of the Consortium for Computing Sciences in Colleges*, 2010.

²²Ellen Spertus, Mark L. Chang, Paul Gestwicki, David Wolber, "Novel Approaches to CS0 with App Inventor for Android", *The 41st ACM Technical Symposium on Computer Science Education (SIGCSE)*, 2010.

²³Jessica Townsend, Mark L. Chang "Work in Progress: Impact of early design instruction on capstone experiences", *Frontiers in Education Conference*, 2010. (section 6.4.4, p.53)

²⁴Mark L. Chang, Jessica Townsend, "A Blank Slate: Creating a New Senior Engineering Capstone Experience", *Proceedings of the American Society for Engineering Education Annual Conference*, June, 2008. (section 6.4.8, p.87)

²⁵Mark L. Chang, Allen Downey, "A Semi-Automatic Approach for Project Assignment in a Capstone Course", *Proceedings of the American Society for Engineering Education Annual Conference*, June, 2008. (section 6.4.7, p.76)

²⁶Adaptive Rough Terrain Navigation on a Legged Robot Platform, Collaborative Research Experience for Undergraduates proposal. (section 6.4.17, p.217)

Committee on the Status of Women in Computing Research in cooperation with the NSF, and we will be beginning this project in Fall 2010.

Perhaps most unexpected of all impacts, however, was the announcement that Kevin and Marlene Getzendanner (P'10) funded a 5-year Olin Tuition Scholarship named in my honor. The Getzendanner family made this gift to the college as a result of the impact my research with their son, James Getzendanner ('10), had on his education. I view this gift as validation that integrating research and teaching with the student experience as the focus is a valuable and impactful philosophy. I am humbled by this act of philanthropy.

Finally, as of August 2010, nine of my publications have received a total of 78 citations. According to *Publish or Perish*,²⁷ these papers have an estimated research impact h-index of 5 and g-index of 8. Of my total publications, 4 are yet to appear in their respective conferences, the 2 ASEE papers are not indexed by Google Scholar, and my book chapter is not individually cited (the book is cited 52 times).

2.6 Consulting Activities

A good part of my time outside the academic semesters has been consulting for various industry partners. In these endeavors, extending the reach of the classroom experience is equally important to me. For all my consulting activities to date, I have incorporated Olin students and alumni.

In 2007, MITRE hired me as a consultant in preparation for the 2007-2008 SCOPE program. Per my recommendation, they hired Matthew Donahoe ('08) as an intern for the summer, and together, we investigated platforms for 3D virtual worlds and their applicability for remote meeting collaboration. This led to the work the SCOPE team did starting in the fall, where Matt was a member of the MITRE SCOPE team.

In 2008, I was invited to be a visiting professor at Yonsei University, one of the top universities in South Korea. I taught a version of Principles of Engineering there to summer students, mostly from the United States.

In 2009, as a result of incorporating Applications Technology, Inc. (AppTek), into the midterm contest for my *Mobile Application Development* course that spring,²⁸ I was hired as a consultant to further develop mobile applications using their machine translation technology and perform research on scaling their back end translation applications into on-demand cloud services. At my suggestion, AppTek hired two Olin students, Jeffrey Moore ('10) and George Harris ('10) as remote interns at Olin to be managed by me. Our work is currently being adopted into AppTek's core product line.

In 2010, AppTek once again hired me as a consultant to focus on several U.S. Military contracts. I am senior personnel on a DARPA proposal to do battlefield translation on Android-powered mobile devices and worked on building an end-to-end (Iraqi Arabic to English to Iraqi Arabic) voice-to-voice translation application on a Google Nexus One Android Phone for deployment in Iraq. For this, I hired an Olin graduate, Ilari Shafer ('10) as a subcontractor.

²⁷Publish or Perish uses data from Google Scholar, and is available at <http://www.harzing.com/pop.htm>

²⁸"AppTek HMT Contest Results," *Multilingual News*, <http://www.multilingual.com/mlNewsArchiveDetail.php?id=2439#6761>

3 Teaching Statement

From its founding, the primary mission for the faculty of Olin College has been to provide the best undergraduate engineering experience possible. It is an inspiring foundational statement, and one that often makes us the envy of our colleagues at other institutions. At Olin, I have embraced the expectation for excellent teaching and have adapted to become a more creative and collaborative teacher. I have involved students in the inception and design of courses and have found the most inspiring moments to be learning *with* our students, rather than simply teaching them.

3.1 A Community of Creators

Over the years, my philosophy and focus on teaching has changed quite a bit. When I started, I had just completed graduate school under the mentorship of a great adviser who really cared about undergraduate education. For him, being prepared and clear was the essence of teaching at a large research university. I took those principles and applied them to the courses I taught in the first year (Computer Architecture and Digital VLSI). Before the classes had even begun, I had every lecture, homework, test, and example prepared. For Computer Architecture alone, this amounted to 300-400 powerpoint slides, two midterms, a final, weekly homework assignments, and five lab assignments—a tremendous amount of work that paid off for the novice educator that I was at the time. I received continuous positive feedback for the preparedness and quality of my teaching, and much positive comparison to courses from other Olin faculty that were teaching more experimental courses that were less well defined.

However, in hindsight, having such a rigid structure predefined left me little room to adjust the course to meet the interests and abilities of the students. If the course turned out to be too easy, I did not have a reasonable mechanism to increase the depth of material. If it was too difficult, I was so bound to my predefined materials that it was difficult to accomodate change. More importantly, I began to feel that the bright young minds in front of me were not being utilized to their fullest potential. Could they have more agency in their own learning? How could I find my way toward more *inspirational teaching*?

During the May 2007 President’s Council meeting, President Miller circulated a white paper asking, “What’s so Special About Olin College?” In the community discussions that followed, it became clear that to a large extent, our community is what makes Olin so special. The term “community of learners”, has been used quite often when referring to the Olin student body. In the spirit of engineering, this phrase may be better coined as “community of creators”.

Olin students often want more than to just be taught—they want to create. Many of our students have a genuine desire to take a larger part in defining their educational experiences. Thus, in refining courses and creating new courses over the years, I have tried to incorporate our students in as much of the process as possible. In the next sections, I discuss how I have modified existing courses and introduced new courses to encourage students *to become co-creators of their educational experiences*.

3.1.1 SCOPE

Teaching SCOPE, the Senior Capstone Program in Engineering, for the first time in 2005-2006 was without a doubt a catalyst for changing my focus on teaching. I credit David Barrett, then Director of the SCOPE program, with allowing the SCOPE faculty so much latitude with how they mentored and managed their student teams and corporate liaisons. Without this level of trust, I would not have acquired a taste for the unknown.

Working with my two talented teams for John Deere and Motorola Research, I learned so much about what motivates Olin students, what they are truly capable of, and how mature they can be. I learned to walk the fine line between team member and team mentor. Some of the key lessons I learned from SCOPE include:

- collaborative learning is exhilarating for faculty and motivating for students
- less structure is a great way to force students to come up with what works for them
- embracing failure is a must

3.1.2 Mixed Analog-Digital VLSI (MADVLSI)

Taking what I learned from SCOPE, Prof. Brad Minch and I merged our two separate integrated-circuits courses—Digital VLSI and Analog VLSI—into a single year-long sequence, Mixed Analog-Digital VLSI (I and II). Working with Brad was a learning experience in being agile and responding to student needs. Brad’s style is an easy blend of a thoroughly prepared technical underpinning coupled with a looser, more student-focused and project-focused in-classroom style. Working together, we sketched a loose plan of material for the fall semester and took strong guidance from students at many points as to what material to cover. The projects and labs remained consistent points to which we could anchor material. However, year to year, the exact topics we cover changes fairly fluidly.

I learned in my first co-teaching with Brad that his depth of understanding of integrated circuit design is what makes this engaging style of teaching possible. Without his preparation, the seamless transitions from topic to topic would be impossible. Amplifying the approach from the first semester, the second semester is completely student driven through a seminar-style approach. Students select their own topics, find appropriate papers, and lead discussions with their peers every class meeting. It is a wonderful experience, and one that has me learning new things every semester.

3.1.3 Mobile Application Development

Another successful and rewarding collaborative learning experiment has been introducing a *Mobile Application Development* course at Olin. During the summer of 2008, inspired by the success of Stanford’s iPhone development course, I sought to develop Olin’s own version of a mobile application development course. I proposed the basic idea of the course to the Olin student body in a survey of interest and received many enthusiastic responses. Of particular interest was a response from Gregory Marra ('10) and Zachary Coburn ('10) who noted that they had hatched a similar idea the previous semester and had written up a mock proposal. They really wanted to make this happen, and they were enthusiastic about co-creating the class. Once we got all the potential students on board, picked a spring semester time slot, secured a classroom from Linda Canvan, it was clear that we were going to be doing everything collaboratively. From even before the first class meeting, it became clear that the students:

- wanted to incorporate design and entrepreneurship
- wanted to build real software and release it to the public
- wanted to contribute back to the community of Android developers

Recognizing that a deep technical understanding would be necessary, I spent the winter break purchasing devices and writing fairly complex applications for all the major smartphone platforms (Nokia Maemo, Google Android, Apple iPhone, Windows Mobile). I also recruited Professors Lynn Stein and Stephen Schiffman to help co-teach the course to augment my own understanding of human-computer interaction and entrepreneurship, respectively.

From the outset, the class was one of the most collaborative learning environments I’ve ever been a part of. We started with enumerating our goals for the class into a common plan of study. Together, we designed all the assignments, cooperated in code reviews and assessment, worked on our industry-sponsored contest, and helped each other develop final project applications that were submitted to the Google Android Market during our launch party in May. I was able to bring in external guests from the mobile industry, including Jason Jacobs, CEO of FitnessKeeper; Larry Marturano, Business Manager and Project Manager, InContext Design Enterprises; Kate Imbach, Director of Marketing, and Joel Nelson, Software Engineering, Skyhook Wireless.

What we created was a unique synthesis course of design, entrepreneurship, and engineering, giving our students a chance to practice so many of the skills they learned in previous courses (Foundations of Business and Entrepreneurship, User-Oriented Collaborative Design, Human Factors in Interface Design, and Software Design). We also created a template of faculty/student collaboration that will guide my design of new and engaging courses in the future. Our efforts were rewarded with an invitation to Mobile Monday Boston in late April 2010, and given an opportunity to present our work, privately, to Governor Deval Patrick, one of my proudest moments as an educator.²⁹

²⁹Photos of the event at <http://bit.ly/c5yTWL> and <http://bit.ly/cMOJew> Michael Ducker ('09), Jacob Getto ('11), and Michael Lintz ('11), presenting to Governor Patrick.

We reprised the course in Spring of 2010 to 20+ students and were treated to a great experience and even more interesting speakers: Jason Jacobs, CEO of FitnessKeeper; Dan Katcher, CEO of Rocket Farm Studios; Paul Lamere, director of the application developer community for The Echo Nest; and Rich Miner, managing partner at Google Ventures and former CEO and founder of Android. The students collaborated again on a midterm contest for The Echo Nest, a music analytics company in Somerville.³⁰ Again, we received positive press for the class. One student project in particular, a music discovery application called Slice, was featured in many popular media outlets.^{31,32,33}

3.1.4 Computer Security

In my most recent co-creation effort, in the Summer of 2010, I am working with Noah Tye ('13) to create a new course in computer and network security with funding from the Olin Innovation grant.³⁴ This has been a collaborative summer, working with Noah on the technical underpinnings of the course, traveling to the premiere hacking conference—DefCon 18—and planning for a pilot course this fall. Dozens of students have already expressed an interest in a software security course, and again, as a collaborative exercise between myself, our domain expert (Noah Tye), and a few bold students, we will hope to create another new course experience.

3.2 Commitment to Teaching

Like many other early faculty at Olin, I chose to come to Olin over more traditional research institutions because of the College's commitment to excellence and innovation in teaching engineering to undergraduates. While we may not be a pure teaching college in the mold of many others, the opportunity to experiment with educating in new ways is why I am here. I have immersed myself in creating new class offerings and supporting independent student work. I have continuously taught at least four courses and advised at least one SCOPE team since the beginning of 2007. If SCOPE were counted as a regular four-credit class, I have been teaching at least six four-credit courses per academic year.

Above my individual in-class commitments, since starting at Olin in Fall 2004 through Spring 2010, I have advised 329 credit hours of research, independent study, Olin Self Study, and Passionate Pursuit activities, the full details of which can be found in my CV. I have also led co-curricular activities every semester except two over the past six years.

³⁰See <http://mobdev.olin.edu/2010/contest.html> for more information.

³¹"Visual Music Discovery with Slice for Android," *Revision3*, May 17, 2010. http://revision3.com/appjudgment/an_eileen_slice

³²Mims, Christopher, "Developers Reinvent the Music Store," *Technology Review*, May 26, 2010. <http://www.technologyreview.com/communications/25387/>

³³Dredge, Stuart, "Android to get Slice visual music exploration app," *Mobile Entertainment*, May 13, 2010. <http://www.mobile-ent.biz/news/37115/Android-to-get-Slice-visual-music-exploration-app>

³⁴Olin Innovation Grant funding for "Network Hacking and Cyber Security" course development. (section 6.4.16, p.214)

4 Service Statement

4.1 Service to the College

I am fortunate to work with colleagues that value service so highly. We all understand that the faculty responsibilities of running the college are not only spread among so few faculty, they also bear a significant weight when bringing a new institution into prominence. As a faculty, we have had to invent all of our policies and processes, which, at most times, has required an extraordinary amount of effort. I am proud to highlight my most significant service to the college in several areas:

- intercollegiate relations
- curriculum reform
- honor board
- resident scholar

4.1.1 Intercollegiate relations

I have spent my entire time at Olin building the relationships between Olin and our partner schools, Babson College, Brandeis University, and Wellesley College. First, with the Wellesley-Olin Working Group committee led by Prof. Helen Donis-Keller, and then as the chair of the Intercollegiate Relations Committee (IRC) since 2007. We have brought the three schools closer than they have ever been, and have forged many new collaborations in curriculum, student exchange, faculty research, and process and policies. Our work has culminated in the recent joint statement on collaboration issued by the three presidents of Babson, Olin, and Wellesley in August of 2009. I continue to work as chair of the IRC and was appointed the official faculty liaison to Babson and Wellesley College in February of 2010.

Some of the work I am most proud of and had a significant responsibility steering, include:

- with Mark Somerville, the creation of a preliminary proposal for a 4-1 dual degree program for Babson and Wellesley Colleges (section 8.3, p.284)
- with input from many Olin stakeholders, creation, implementation, and maintenance of the Olin Certificate Program in Engineering Studies (section 8.1, p.274)
- with help from the Academic Calendaring Committee, co-planning and synchronizing academic calendars between Babson, Olin, and Wellesley
- with help from the Office of Student Life at Olin and the Office of Transportation and Housing at Wellesley, the creation of the cross-campus shuttle service
- creation of a risk analysis report for the Olin Board of Trustees (section 8.2, p.279)
- being invited to speak to the Wellesley Board of Trustees about the Olin/Wellesley partnership
- promoting our course offerings to students at our partner institutions

4.1.2 Curriculum reform

I have a demonstrated interest in practicing curriculum reform in both my research and teaching activities. Naturally, being asked to serve my colleagues in this capacity has been a privilege. As a result of our curriculum review during a faculty retreat in 2007, the ARB asked me to chair a task force on the 2nd and 3rd year curriculum. We were charged with investigating the current status of the year 2/3 curriculum, and to suggest avenues for improvement. The result of our work can be found in the final report from our effort (section 8.4, p.288).

I was also asked to serve on an adhoc committee to investigate the state of curricular innovation. The culmination of our work was presented by Prof. Mark Somerville at a Spring 2010 faculty meeting.

4.1.3 Honor board

From 2004-2009, I also served as the faculty representative to the Honor Board. In my capacity as faculty representative, I went to weekly meetings of the Honor Board, advised the board on student-faculty relations, presented Honor Board data to the faculty, and most importantly, represented the faculty on academic-related honor board cases.

4.1.4 Resident scholar

Since the Fall of 2005, my wife, myself, and since November of 2006, my son Carter, have been living in East Hall. In my duties as Resident Scholar, I run academically-enriching activities every semester, often in the form of co-curriculars. Most prominently, my wife Caryn Park and I have run a Social Justice Reading Group that affords our students time to think about the world in which we live, social problems, how society is organized, and the workings of power. Our readings and discussions have been attended regularly by about a dozen or more students each semester, with most students having tenure in the group for multiple years. Focusing on social change and how it intersects with engineering, we have read texts that range from critical theorists such as Michel Foucault, to postmodernists writing about The Matrix, to cultural, postcolonial, and critical race/gender theorists. We also participate in orientation activities, especially around the topics of diversity and the honor code.

On a personal note, the privilege of living on campus has allowed me, as a father of a new family, to condense my regular daytime hours so I can spend time with my family, as I am able to hold office hours and student interactions in the late evenings when it is more convenient for both the students and my family. It has made being a new parent much easier.

4.1.5 List of Committees and Department Service

1. Ad hoc committee on curricular innovation, Fall 2009
2. SCOPE Director search, Fall 2009
3. Committee on Diversity and the Academic Experience, 2005 - 2007
4. Electrical and Computer Engineering Faculty Search committee, 2004, 2005, 2007
5. Electrical and Computer Engineering Program Group, 2004 - present
6. Faculty / IT committee, 2004 - 2007
7. Honor Board faculty representative, 2004 - 2009
8. Intercollegiate Relations Committee, 2007 - present (chair 2007 - present)
9. Task force on the 2nd and 3rd year curriculum, 2007 - 2008 (chair)
10. Wellesley Olin Working Group Committee, 2004 - 2007
11. Olin Certificate in Engineering Studies coordinator, 2007 - present
12. Adviser for all Korean exchange students

4.2 Service to the Profession

I am very active in maintaining visibility and good standing with many research communities. I continually review for conferences and journals across a broad range of research topics. I have been on the steering committee of FCCM 2010, one of the most prominent conferences in the field of reconfigurable computing. I am also general co-chair of a workshop on localization technologies that has been co-located with premier conferences in the field of networking and ubiquitous computing. Details can be found in my CV.

5 Other Contributions

For many faculty at Olin, our work might not always fall neatly into one of the bins of research, teaching, and service. Or oftentimes, significant work has been accomplished that blends more than one of these areas. I would like to highlight several activities that don't necessarily fit well into other statements.

5.1 Advanced Computing Laboratory

Since arriving in 2004, in collaboration with Prof. Gill Pratt, Prof. Allen Downey, and Prof. Lynn Stein, we have advocated for a model of a computing facility that was similar to the operation of the Olin machine shop. While each of our students has a laptop for their personal and school use, many computing-related investigations, projects, courses, and research, require alternate environments that cannot be well supported by a personal laptop. Therefore, in September 2004, we proposed the Advanced Computing Laboratory (ACL). Since then, I have maintained the lab and worked with students and faculty to support student work on campus.

Since our inception, students have continuously used the services and facilities for classes, projects, and research. It has lived up to our expectations in many ways and has become an instrumental part of everyday computing on campus. Some of the highlights include:

- code revision management and project management tools (subversion and trac), which almost all SCOPE teams and many PoE teams have used
- HALP, the advising tool
- Directory, the student contact directory database
- Cadence software for VLSI courses
- Olin Design Observatory research lab
- Olin Intelligent Vehicles Lab resources
- The OLPC student project resources
- Olins Mauraders map location service
- CORe
- Microsoft and VMWare licensed software distribution

5.2 External Relations

Interfacing with the technical community at large has been a passion of mine, especially in classes like Mobile Application Development. It provides so much context for the learning that happens in the classroom, and it generates buzz and positive feedback for Olin in general. Several efforts in particular deserve mention.

5.2.1 Mobile Application Development

For my class, I have routinely invited external guests to speak to the class. I have also engaged external industry partners to sponsor the class in general, and to sponsor and provide technical resources for midterm contests to help motivate students.

Because of the popularity of the class and the proximity of Olin to so much of the wireless technology sector in the Boston area, we have been able to attract an enviable cast of guests. These include:

- Jason Jacobs, CEO of FitnessKeeper
- Larry Marturano, Business Manager and Project Manager, InContext Design Enterprises
- Kate Imbach, Director of Marketing, and Joel Nelson, Software Engineering, Skyhook Wireless
- Dan Katcher, CEO of Rocket Farm Studios
- Paul Lamere, director of the application developer community for The Echo Nest
- Reed Sturtevant, Founder and Managing director of Project 11 Ventures, former Founding Director of Microsoft Startup Labs, CTO of Eons, Inc., Managing Director and Vice President of Technology for IdeaLab
- Rich Miner, managing partner at Google Ventures and former CEO and founder of Android

Our industry partners have contributed funds to purchase hardware, textbooks for the students to use, access to technologies for our contests, and prizes to motivate our students. Our partners have included:

- Applications Technology: donated \$2,000 in prize money and access to their proprietary machine translation software for a class contest
- The Echo nest: donated prizes of Apple iPod Touch devices and access to their music analytics engine for a class contest
- AndroidCentral.com: financial support to buy hardware and mobile data service for the class
- CommonsWare: donation of textbooks for all students
- Palm, Inc.: donation of textbooks for all students

5.2.2 Computer Architecture

Encouraging very open-ended, self-directed, computing-related projects has been both exciting and rewarding for myself and for the student teams. Their hard work on interesting projects has paid off in particular for two projects. First, the K'Nex computer, by Matt Donahoe ('08), Jeff DeCew ('08), and Olek Lorenc ('08), received mention on the popular Make Magazine Blog.³⁵. Second, a final project to build a cheap 3D scanner has received much attention, including a mention in Wired's Gadget Lab.³⁶.

I am very proud of these student accomplishments. No doubt they will encourage current and future students to follow in their innovative footsteps.

5.2.3 Seminar series

With Debbie Chachra, we have tried to take advantage of our proximity to a rich and vibrant technology sector by running an adhoc seminar series in 2009-2010. Some of our honored guests have included:

- Scott Kirsner, Technology writer for Variety, The Boston Globe, and many other publications
- Zoz Brooks, Host of Discovery Channel's *Prototype This* show
- danah boyd, Senior Social Media Researcher at Microsoft Research New England
- Scott Berkun, author of "Confessions of a Public Speaker"
- Kasson Crooker, Senior Producer, Harmonix Music Systems

5.3 Faculty Pub Night

Over the past few years, and in particular during the 2009-2010 academic year, I have made an effort to bring our faculty together in a more social atmosphere. I believe in a strong collegial workplace, and so often, our hectic and distributed work schedules do not allow for us to interact with one another. Recognizing that one of our greatest assets are our colleagues, Prof. John Geddes and I spearheaded this past year's Faculty Pub Night organization.³⁷. Each event has been well received and well attended. I am proud to be a part of bringing a sense of family back to our overworked faculty. I will be continuing this effort as long as possible.

³⁵Torrone, Phillip, "The KNEX Computer", *Make Magazine Blog*, June 28, 2007. http://blog.makezine.com/archive/2007/06/the_knex_computer.html

³⁶Rowe, Aaron, "Young Engineer Uses Webcam, Laser to Build Budget 3-D Scanner," *Wired: Gadget Lab*, August 8, 2010. <http://www.wired.com/gadgetlab/2010/08/budget-3-d-scanner/>

³⁷<http://facultypubnight.com>

6 Record of Intellectual Vitality Achievements

6.1 Publications in Print

The following encompasses the majority of publications in print. All are in peer-reviewed venues.

As Assistant Professor at Olin College

1. Ilari Shafer, Mark L. Chang, "Movement Detection for Power-Efficient Smartphone WLAN Localization", *13th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, October 2010. (section 6.4.1, p.32)
(132 submissions, 43 regular and 9 short papers accepted, rate: 33%)
2. Andrew Barry, Noah Tye, Mark L. Chang, "Interactionless Calendar-Based Training for 802.11 Localization," *The 7th IEEE International Conference on Mobile Ad-hoc and Sensor Systems*, November 2010. (section 6.4.2, p.42)
(185 submissions, 51 accepted, rate: 28%)
3. Mark L. Chang, "Work in Progress: synthesizing design, engineering, and entrepreneurship through a course in mobile application development", *Frontiers in Education Conference*, 2010. (section 6.4.3, p.51)
4. Jessica Townsend, Mark L. Chang "Work in Progress: Impact of early design instruction on capstone experiences", *Frontiers in Education Conference*, 2010. (section 6.4.4, p.53)
5. Andrew Barry, Benjamin Fisher, Mark L. Chang, "A Long-Duration Study of User-Trained 802.11 Localization," *Proceedings of the Second ACM International Workshop on Mobile Entity Localization and Tracking in GPS-less Environments*, September 2009. Awarded best paper and best presentation. (section 6.4.5, p.55)
6. Stephen Longfield, Jr., Mark L. Chang, "A Parameterized Stereo Vision Core for FPGAs", (Short Paper) *IEEE Symposium on Field-Programmable Custom Computing Machines*, April 2009. (section 6.4.6, p.72)
(95 submissions, 25 full and 24 short papers accepted, rate: 52%)
7. Mark L. Chang, Allen Downey, "A Semi-Automatic Approach for Project Assignment in a Capstone Course", *Proceedings of the American Society for Engineering Education Annual Conference*, June, 2008. (section 6.4.7, p.76)
8. Mark L. Chang, Jessica Townsend, "A Blank Slate: Creating a New Senior Engineering Capstone Experience", *Proceedings of the American Society for Engineering Education Annual Conference*, June, 2008. (section 6.4.8, p.87)
9. Mark L. Chang, "Device Architecture", in *Reconfigurable Computing: The Theory and Practice of FPGA-Based Computation*; Scott Hauck, Andre DeHon, Editors; Morgan Kaufmann/Elsevier, 2008, pp. 3-27. (section 6.4.9, p.100)
10. Mark L. Chang, Scott Hauck, "Précis: A Design-Time Precision Analysis Tool", *IEEE Design and Test of Computers*, Vol. 22, No. 4, pp. 349-361, July-August 2005. (section 6.4.10, p.128)

Prior to Employment at Olin College

1. Mark L. Chang, Scott Hauck, "Automated Least-Significant Bit Datapath Optimization for FPGAs", *IEEE Symposium on Field-Programmable Custom Computing Machines*, April, 2004. (section 6.4.11, p.141)
2. Mark L. Chang, Scott Hauck, "Précis: A Design-Time Precision Analysis Tool", *IEEE Symposium on Field-Programmable Custom Computing Machines*, pp. 229–238, 2002. (section 6.4.12, p.150)
3. Mark L. Chang, Scott Hauck, "Adaptive Computing in NASA Multi-Spectral Image Processing", *Military and Aerospace Applications of Programmable Devices and Technologies International Conference*, 1999. (section 6.4.13, p.160)

6.2 Other Intellectual Vitality Achievements

6.2.1 Conference posters

1. Mihir Ravel, Mark L. Chang, Mark McDermott, Michael Morrow, Nikola Teslic, Mihajlo Katona, Jyotsna Bapat, "A Cross-Curriculum Open Design Platform Approach to Electronic and Computing Systems Education," *IEEE International Conference on Microelectronic Systems Education*, July 2009.
2. C. Murphy, D. Lindquist, A.M. Rynning, T. Cecil, S. Leavitt, M.L. Chang, "Low-Cost Stereo Vision on an FPGA", IEEE Symposium on Field-Programmable Custom Computing Machines, 2007.
3. Mark L. Chang, Scott Hauck, "Least-Significant Bit Optimization Techniques for FPGAs", *ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*, February, 2004.

6.2.2 Panel and workshops

1. Hal Abelson, Mark L. Chang, Cyprien Lomas, David Wolber, "Google App Inventor for Android: Building mobile applications as a first computing experience" *Frontiers in Education Conference*, 2010. (*to be presented*)
2. Hal Abelson, Mark L. Chang, Eni Mustafaraj, Franklyn Turbak, "Mobile Phone Apps in CS0 Using App Inventor for Android", *15th Annual Conference of the Northeast region of the Consortium for Computing Sciences in Colleges*, 2010.
3. Ellen Spertus, Mark L. Chang, Paul Gestwicki, David Wolber, "Novel Approaches to CS0 with App Inventor for Android", *The 41st ACM Technical Symposium on Computer Science Education (SIGCSE)*, 2010.

6.2.3 Invited talks

1. "Master of motivation: engaging students with smartphones and Google Android", *Boston-area Advanced Technological Education Connections IT Futures Forum*, May 2010.
2. "Spinning the World Wide Web: How the Internet Really Works", *Needham Exchange Club presentation*, October 2008.
3. "Olin College: Accrediting an Innovative Engineering Curriculum", *Yonsei University Engineering Seminar*, August 2008.
4. "Olin College: Rethinking Engineering Education", Microsoft Research, June 2008
5. "A Beginner's Guide to Bad Engineering Presentations", *University of Hartford*, November 2007.
6. "Spinning the World Wide Web: How the Internet Really Works", *Olin College Lecture Series, Needham Adult Education Program*, October 2007.

6.3 Grants Submitted

The following are reprints of grants submitted.

1. *Under review:* NSF TUES proposal with Gunar Schirner (Northeastern University), David Kaeli (Northeastern University), Mark Somerville (Olin), and Mihir Ravel (Olin), *Collaborative Research, TUES-Type 1: Fostering Student Learning Continuity Employing a Personal Active Learning Platform*. (section 6.4.15, p.192)
2. *Under review:* NSF proposal as senior personnel with co-PIs Debbie Chachra (Olin) and Lynn Stein (Olin), *REU Site: Engineering Education Research: Understanding and Improving Student Experiences* (section 6.4.14, p.170)

3. *Awarded student proposal: Adaptive Rough Terrain Navigation on a Legged Robot Platform*, Collaborative Research Experience for Undergraduates proposal. (section 6.4.17, p.217)
4. *Awarded*: Olin Innovation Grant funding for “Network Hacking and Cyber Security” course development. (section 6.4.16, p.214)
5. *Not awarded*: Alzheimers Foundation Early Technologies for Alzheimers Care proposal with Aaron Boxer (Olin) and Stephen Schiffman (Olin), *Ubiquitous Computing for Carepartner Relief Through Patient Independence*. (section 6.4.18, p.224)

6.4 Reprints

Reprints of the work cited above can be found in the following pages.

Movement Detection for Power-Efficient Smartphone WLAN Localization

Ilari Shafer
ilari.shafer@alumni.olin.edu

Mark L. Chang
mark.chang@olin.edu

Olin College of Engineering
1000 Olin Way
Needham, MA 02492

ABSTRACT

Mobile phone services based on the location of a user have increased in popularity and importance, particularly with the proliferation of feature-rich smartphones. One major obstacle to the widespread use of location-based services is the limited battery life of these mobile devices and the high power costs of many existing approaches.

We demonstrate the effectiveness of a localization strategy that performs full localization only when it detects a user has finished moving. We characterize the power use of a smartphone, then verify our strategy using models of long-term walk behavior, recorded data, and device implementation. For the same sample period, our movement-informed strategy reduces power consumption compared to existing approaches by more than 80% with an impact on accuracy of less than 5%. This difference can help achieve the goal of near-continuous localization on mobile devices.

Categories and Subject Descriptors

C.2 [Computer-Communication Networks]: Miscellaneous; C.3 [Special-Purpose and Application-Based Systems]

General Terms

Design, Experimentation, Measurement

Keywords

Localization, mobility, power-efficiency, smartphone, experimental evaluation, accelerometer

1. INTRODUCTION

The continued spread of mobile and ubiquitous computing has brought with it a desire to determine the location of portable devices and their users. Using the contextual information provided by location offers the possibility of co-ordinating the behavior of devices or their environments.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MSWiM'10, October 17–21, 2010, Bodrum, Turkey.
Copyright 2010 ACM 978-1-4503-0274-6/10/10 ...\$10.00.

Location-based services, which include asset tracking and routing [8], person tracking [23], phone behavior modification [24], and advertising [16], constitute a large and growing market [19].

One particular focus of localization work is in indoor environments, where many traditional location sources like GPS and cellular networks have poor coverage or are unavailable [11]. To overcome these obstacles and provide high precision indoors, systems have emerged based on radio frequency ID (RFID) tags, infrared transceivers, custom radio systems, and 802.11 WLAN infrastructure [20]. Among these technologies, positioning based on 802.11 is an attractive solution and often offers the lowest setup cost: many facilities are already equipped with access points, 802.11 devices are an inexpensive commodity, and many network users already use the platform.

A number of obstacles lie in the path of effective WLAN positioning, including system accuracy, training or site surveys, calibration, coverage, and the observance of users' privacy [18]. One design goal that has not received much attention is the device power consumption of a localization system. Energy efficiency is key for mobile platforms, which typically have highly constrained battery resources and a need or desire to operate independently for extended periods. In particular, power conservation is an increasingly valuable aspect for recent feature-rich smartphones, [7] for which many localization applications are designed.

Here, we propose a method for WLAN localization that is supplemented with accelerometer data to localize only when a user moves, thereby reducing power consumption. Since this sensor requires much less power than an 802.11 radio and its associated computational work, the system provides the same quality of location data (provided a sufficiently accurate movement-detection metric) for a longer time.

2. RELATED WORK

Indoor localization is a well-studied problem in the literature. Early localization systems such as the Active Badge (Want et al [25]) used custom hardware to detect wearable devices, and many frameworks today use specialized tags [8, 23]. Bahl et al, in developing RADAR, were among the first to demonstrate the possibility of using 802.11 networks to provide localization based on triangulation using the received signal strength indicator (RSSI) [2], and today WLAN information is even used commercially by providers like Skyhook Wireless [22].

Existing research efforts have often observed that the bat-

terry life of mobile devices that localize regularly is poor; Hightower et al find that a battery life of 10 hours is “less than desirable” [12]. Work on consumer mobile devices has been more severely limited by power usage. For example, PlaceSense (while not focusing on battery life) drained the battery of a smartphone within 4-5 hours [15].

Noting the energy conservation needs of localization applications, a number of approaches have developed to produce lower power consumption. Many aim to predict the user’s mobility: the velocity history of a device can serve as a heuristic to inform sampling frequency. Xu et al simulate such a strategy, and observe a tradeoff between missing changes in location and energy conservation for any such scheme [27]. Chen et al observe the variety of efforts to reduce the power consumption of hardware on mobile phones, and instead focus on a software-based approach using access point clustering to reduce the computational and communication cost of localization [6].

From the sensor side of our approach, accelerometers have been used in localization applications to be the sole provider of data or augment the information provided to a localization service. Since many modern smartphones have a variety of built-in sensors, typically including an accelerometer, it is attractive to use the sensor either passively (i.e. to sense the environment) or actively (i.e. to determine the response to vibration). Woodman and Harle use accelerometers as part of an inertial measurement unit and exemplify the dead-reckoning approach to positioning [26], whereas SurroundSense [1] proposes using accelerometers in conjunction with light and sound information to create a location fingerprint.

Most similar to our work is the approach taken by You et al [28], which focuses on using accelerometer data to estimate the mobility of a user. Much like our approach, the authors use an accelerometer to help determine when localization should occur. However, the approach adapts its sample period to expected movement, which risks “nonconformance” due to missed samples. We aim to develop a strategy that only misses localization when movement misclassification occurs. Additionally, whereas You et al use 802.15.4-based MICAz sensor nodes and custom transmitters, we are interested in the impact on consumer WLAN devices such as commercially-available smartphones.

3. LOCALIZATION STRATEGY

Our goal in creating a localization strategy involves meeting multiple requirements. Our goals are fourfold:

3.1 Goals

1. *Reduces the power consumption of localization:* This is the primary goal of our work.
2. *Runs on consumer smartphones:* We aim to develop a strategy that can be applied to popular, functioning WLAN localization systems like those described in [22] and [4].
3. *Works with existing localization frameworks:* We would like to use an approach that augments existing WLAN services, so that increasing energy efficiency only involves modification of client-side software. For example, although moving location computation to the client reduces energy-consumptive communication [6],

it requires system-wide modification and does not scale well to large environments.

4. *Does not introduce “non-conformance” error:* In contrast to approaches that increase the sample period, we develop a technique that does not miss movement due to prediction error.

These requirements inform our design choices. In order to meet (2) above, we perform our analysis consistently using the HTC Dream (aka G1) [14], which is a prototypical smartphone based on the Android software platform. To satisfy (3), we consider the system described in [4], which provides an existing on-campus localization service. Although it is not currently trained for smartphone localization, it allows us to produce the precise set of events necessary to localize.

3.2 Strategy

The strategy we develop aims for simplicity: in essence, we localize when we detect a user has moved to a new location. To do so, we poll the accelerometer of a device at a fixed period to determine whether a user is walking or not. We only perform a full localization when a user was previously walking and is no longer walking. This behavior is summarized in Fig. 1.

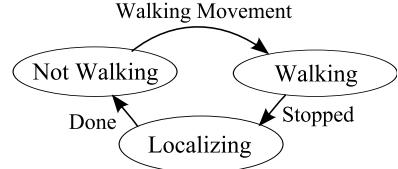


Figure 1: System State Diagram. Information about user movement from the accelerometer mediates transitions to and from the walking state.

A full localization event for the WLAN framework we use (represented by “Localizing”) requires information about the device’s location in signal space and a communication to map to physical space. In detail, a full localization comprises:

- Waking the device’s WLAN radio
- Initiating a scan for nearby access points and waiting for it to complete
- Assembling a list of the RSSI and identifier for each discovered access point (a “fingerprint”)
- Associating with an access point to establish a WLAN data connection
- Sending the fingerprint to a server running the localization service over the connection
- Receiving an identifier for a physical location from the server

We obtain information about user movement by polling the accelerometer periodically and analyzing the resulting data for movement events. See Sec. 6.1 for further explanation of the walk detection process, and Fig. 7 for a depiction of the alignment of localization with respect to movement.

Although our strategy uses a fixed sample rate to determine movement, we note that it does not preclude the addition of other approaches. For example, power savings from hardware-level WLAN optimization would decrease the cost of each full localization and the strategy as a whole.

4. POWER PROFILING

An analysis of the power behavior of the strategy outlined above depends upon an understanding of the energy consumption of localization, accelerometer polling, and movement detection. We describe our approach to studying these factors here.

4.1 Current Measurement

One way of determining the power profiles of WLAN access, accelerometer polling, and device sleep is through direct measurement of the current and voltage delivered by its battery. Since battery voltage over short timescales is relatively constant, instantaneous power is simply given by $P(t) = V \cdot I(t)$.

To measure voltage, we bring the device battery to the fully-charged state and measure a steady 4.15V across it. Then, we construct the current measurement arrangement shown in Fig. 2, in which a 0.1Ω Ohmite 12FR010 current sense resistor lies between battery and device. We use a Keithley 2400 single-channel source-measure unit (SMU) as the sensor labeled “V”. To calibrate for the additional resistance of our arrangement, we use the SMU to perform a sweep over the voltage range we measure and linearize the resistance we observe with a small current offset.

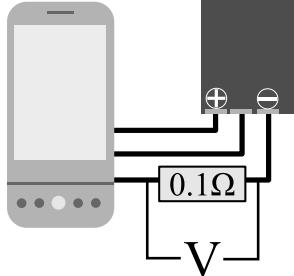


Figure 2: Power Measurement Arrangement. Potential difference across the small fixed resistance is measured by a source-measure unit with data logging capabilities.

To capture power consumption information for each of the activities we need for localization, we focus on measuring power for each of the following scenarios:

- *Completely Off*: Device has been manually shut down
- *Sleep*: When the device is not in use, it falls back to a sleep mode during which certain events (like calls or scheduled wakeups) can bring it back online.
- *Screen On*: The screen consumes a great deal of power, and since our test for CPU usage requires the screen to be on, this value serves as a baseline to subtract from “Heavy CPU”
- *Heavy CPU*: We produce full CPU usage by running a compute-intensive task on the device.

- *Heavy WLAN*: We attempt to determine WLAN radio power consumption by using the radio to continuously transmit and receive HTTP requests for email.
- *Accelerometer*: We write an application that performs the intermittent accelerometer polling required for movement detection and measure its consumption.

Table 1 summarizes our findings and compares them with rough estimates of their relative values. These estimates are obtained from a resource in the Android OS source code that provides values for screen, Bluetooth, WLAN, CPU, DSP, cell radio, and GPS power consumption for a generic device [10]. Many of our test conditions are best represented by a combination of these values; for example, the “Heavy CPU” task is performed at full screen brightness (screen.full) and places a full load on the CPU (cpu.full). Our measurement setup creates an additional load, which is reflected by a small measured power consumption even when the phone is completely off.

Activity	Measured (mW)	Scaled / Unscaled [type] (unspecified power units)
Completely Off	4.18	4.18 / 0 [none]
Sleep	8.63	8.43 / 1.6 [cpu.idle]
Screen On	317	311 / 116 [cpu.idle+screen.full]
Heavy CPU	799	677 / 254 [cpu.full+screen.full]
Heavy WLAN	754	799 / 300 [cpu.normal+wifi.active]
Accelerometer	321	269 / 100 [cpu.normal]

Table 1: Power Measurements. The measurements we obtain for localization-specific behaviors have similar relative values to rough estimates from power data for a generic device. For comparison, we produce the “Scaled” values by scaling the range of the unscaled Android power profile data to the power values we measure. These unscaled data are generated from combinations of CPU, WLAN, and screen power consumption.

The primary observation we make from these results is that device sleep is very inexpensive in terms of power use: its consumption is nearly two orders of magnitude smaller than heavy CPU usage. A device that remains in sleep would take more than 20 days to completely drain its battery. Additionally, polling for movement, which only requires CPU usage and the minimal power drain of the accelerometer, is less costly than performing wireless localization — particularly since it also takes less time (see Sec. 6.1).

Although these data confirm the general principle that polling for movement requires significantly less power than localization, a more specific analysis is necessary for the power consumption of WLAN scanning and access. Unlike accelerometer readings, which have a fairly constant power use, the WLAN radio has multiple levels of power usage, which depend upon the details of establishing a connection and transmitting data [3]. As shown in Fig. 3, we observe these variations in power use when obtaining a trace from

our test setup for a single WLAN data request. This nonuniformity motivates our desire to measure average-case WLAN energy consumption for localization with a rundown analysis.

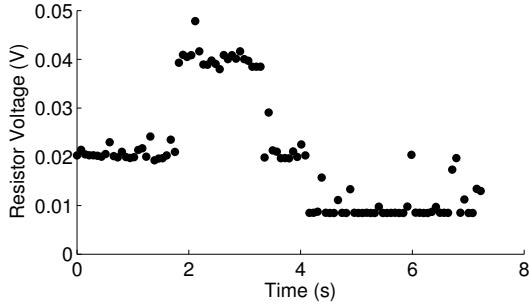


Figure 3: Variable WLAN Power Consumption. WLAN access and scanning has multiple power states, three of which are visible in this power trace of a single HTTP GET. Voltage is recorded every 0.073s.

We use the current measurements from Table 1 to confirm the measurements we make in the next section and to inform our choice of additional infrequent scanning to our strategy (see Sec. 7.2). Our measurements of device sleep power establish the value we use for inter-event power consumption.

4.2 Rundown Analysis

To obtain a more comprehensive portrait of power consumption, we collect information on the time required to drain the device’s battery for localization (WLAN scanning and data transmission) and movement sensing. Although this process takes much longer and is less precise than current measurement, it adds average-case information about localization that we could not otherwise obtain. Our primary goal in performing this alternate analysis is to determine the relative cost of accelerometer polling and WLAN localization, which complements the relationship to sleep and CPU cost obtained above.

A key requirement of rundown analysis is a battery sensor that is consistent and accurate, though not necessarily precise. Fortunately, the HTC Dream device we use exposes a hardware battery sensor to the Android API that can meet these requirements with minimal compensation. This API reports a percentage of battery charge to user code.

To test the power characteristics of movement sensing and localization, we develop an Android application that performed the same duties as the final strategy, but that only performs one or the other activity repeatedly. Based upon the measurements in Sec. 4.1, we focus on the *per-event* energy expenditure of accelerometer and WLAN activity, since device sleep is a constant contributor that only depends on inter-event time.

Fig. 4 illustrates traces for three rundown trials at the same event rate — two for WLAN scanning to demonstrate the correspondence between repeated measurements, and one for WLAN localization. The battery sensor itself has a nonlinear sensed-battery versus time characteristic for a constant power draw. To compensate for this nonlinearity, we average all of the rundown trials that use constant power to find an aggregate battery consumption curve. We

then compute the numerical inverse of this curve and use it as a lookup table for battery readings from the device. In essence, we find a discrete function that maps from average measured battery reading to a linear profile.

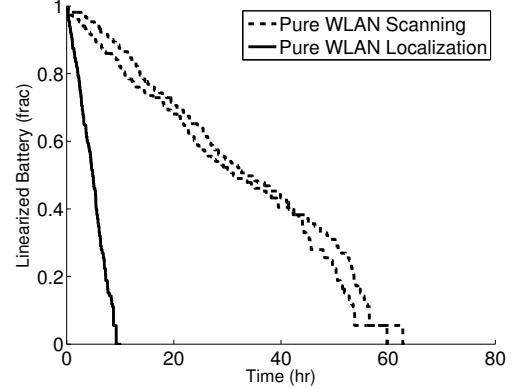


Figure 4: Sample Battery Rundown. Battery drain traces are shown for localization and WLAN scanning, both at 0.05Hz. Though inexact, there is a clear correspondence between traces from the same activity (as shown here, Pure WLAN Scanning). Additionally, state of charge vs time is sufficiently linear to be useful.

The rundown analysis indicates that the mapped battery measurements, while not perfectly linear, are sufficiently consistent for use in measuring state of charge. These measurements, in conjunction with the data from Table 1, form a basis for evaluating our localization strategy.

5. SIMULATION

To analyze the power and accuracy characteristics of our strategy under a variety of conditions, we simulate its long-term power consumption. The empirical data gathered from the current measurement and rundown profiles of the phone enables us to perform this simulation. Each time full WLAN localization, WLAN scanning, or movement sensing occurs, the simulation computes power cost as:

$$Cost = \frac{t - t_{last}}{TimeToSleepDrain} + EventCost$$

This model of battery consumption uses a *TimeToSleepDrain* that would be required to fully drain the battery according to the current measurements in Table 1 and an average *EventCost* taken from the additional data in rundown trials.

We verify that the simulation is an accurate model by running it for the same behaviors we use for rundown analysis. Fig. 5 depicts the concordance of the simulation results with experimental data. For this particular comparison, we run the experimental WLAN localization until the battery is depleted (which causes the device to shut down) and run accelerometer polling for 29 hours.

6. IMPLEMENTATION

To evaluate the real-world performance of our strategy, we write an Android application to evaluate the functionality, accuracy, and power consumption of our localization strategy. It must be able to detect user movement and exhibit consistent power consumption.

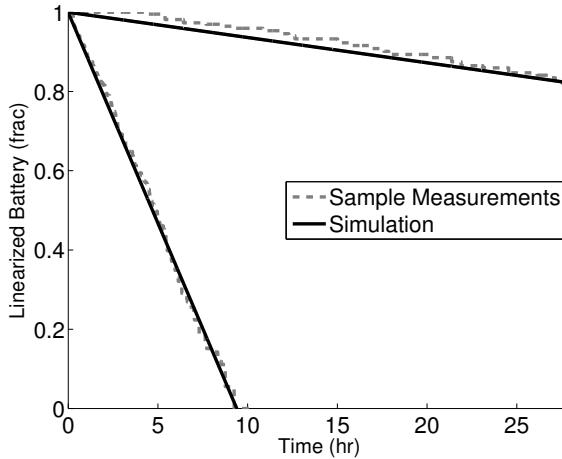


Figure 5: Simulation of Power Profile. We run a simulation of power consumption for two simple behaviors: use the WLAN radio to localize at a fixed rate, and poll for accelerometer data at the same rate. These simulated results match experimental results for the same behaviors.

6.1 User Movement

A key requirement of accelerometer-based movement detection is that it reliably captures periods when the user is moving. Fortunately, considerable work has been done in detecting gait from accelerometer measurements [9]. One finding we incorporate is the length of time we should sample to produce useful information about movement. We opt to sample for 350ms at the fastest update rate delivered by the software (approximately 35Hz), although the sorts of accelerometer data we expect vary based upon the location of the phone on a user’s body.

The details of gait capture are in fact not critical for the movement detection algorithm, since we do not actually need to detect gait — rather, we need a boolean output of whether the user is walking. To meet this requirement, and to reduce the computation associated with sensing movement, we use a metric for movement that is the sum of the unbiased variance of X, Y, and Z acceleration:

$$\text{Var}(m_1..m_N) = \frac{\sum_{i=1}^N m_i^2 - \frac{1}{N} \left(\sum_{i=1}^N m_i \right)^2}{N - 1}$$

$$\text{Metric} = \text{Var}(x_1..x_N) + \text{Var}(y_1..y_N) + \text{Var}(z_1..z_N) \quad (1)$$

We compute the sum of squares and the sum in the variance computation as data collection proceeds, which allows us to avoid storing measurements. Also notable is that since Eq. 1 does not depend upon the mean of the accelerations; the phone can be held in any orientation. To verify that this metric suffices for movement detection (and as part of the process of selecting it), we record accelerometer data for a number of user activities. We log data to a file while holding the phone, typing on its physical keyboard, interacting with the screen, resting with the phone in a pocket, and walking.

Fig. 6 contains empirical cumulative distribution functions for walking behaviors and non-walking behaviors. It is evident that a threshold on this metric can be a good predictor of walking movement; we choose one that includes

all walk events we obtained. We intentionally choose such a low threshold since excluding walking events means missing desired localization events — a situation we do not want to introduce. The impact of the classification accuracy is discussed further in Sec. 7.2.

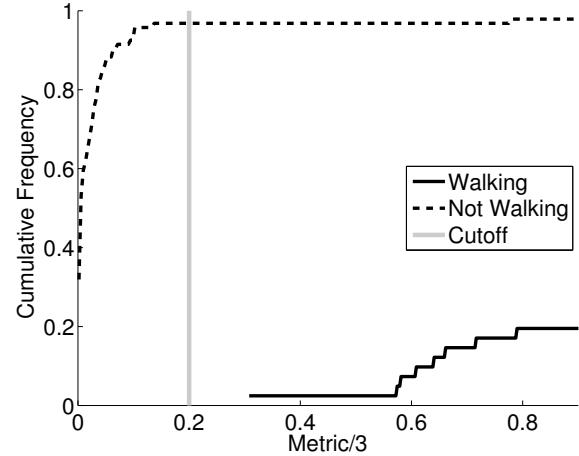


Figure 6: Movement Detection Threshold. We aggregate all data from non-walking behaviors and walking, then compute the metric in Eq. 1. A clear distinction between walking and non-walking events is discernible, and we choose a cutoff threshold accordingly.

6.2 Application

We design our application to be minimally dependent upon other variable power costs. To control the impact of other intermittent sources of power consumption, we stop tasks that may run based on other inputs from the environment.

To do so, we disconnect the device from the cellular network by removing its SIM card and remove the association with an email account that is typically carried by Android phones. Display power consumption is negligible, since we write the application as a background service that does not wake the screen. Also, we stop other background processes that are not part of the default operating system. These techniques mimic the use of smartphones in previous studies [15], and also ensure that our impact on the phone is limited to a standard user-level Android application.

7. RESULTS

We evaluate our movement-informed localization strategy on the basis of its long-term accuracy and power savings. These results are gathered from a simulation that is based on the data we describe in Sec. 4. To demonstrate that the strategy functions in practice, we first describe the accuracy and power savings of an implementation.

7.1 Functionality

While walking around two floors of an indoor space, we run our application on a phone that is carried in a user’s pocket. Concurrently on a second mobile device, the user records “ground truth” data of when he is actually walking. We sample for movement with a fixed period, and record

when the localization strategy determines that a full localization is necessary.

As shown in Fig. 7, the periods of time that the application identifies as “walk” intervals are the same as those recorded by the user, and differ in time by less than one sample period. Localization events, as expected, occur on the transition from walking to not walking. It is clear that the accelerometer-based strategy uses much less power than the battery consumption we find from localizing every sample period. We discuss power usage more thoroughly in Sec. 7.3 below.

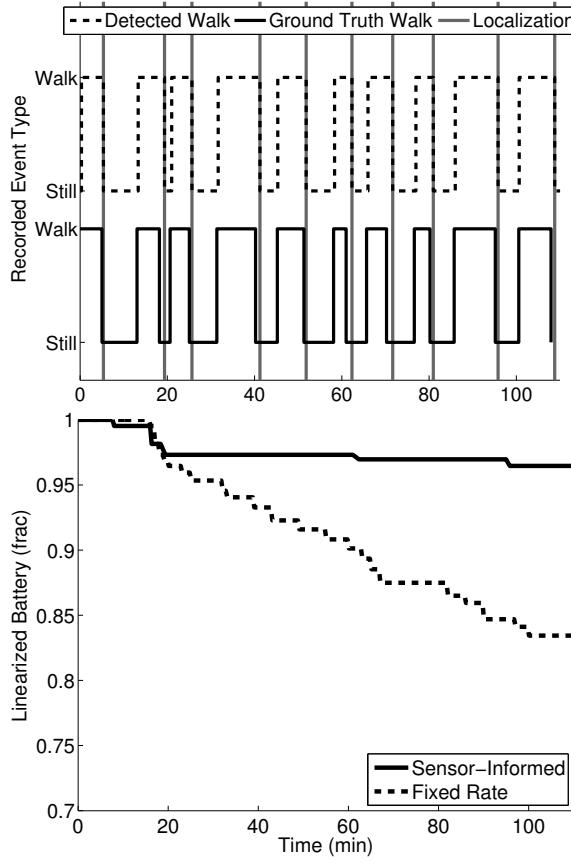


Figure 7: Implementation Logs. Periods of actual walking (“Ground Truth Walk”) are detected by our application (“Detected Walk”) and result in localization events on walk to still transitions (“Localization”). Power consumption for the accelerometer-informed strategy (“Sensor-Informed”) is much lower than naïve localization (“Fixed Rate”) with the same sample period (once every 20 seconds).

7.2 Accuracy

We define the accuracy of our localization strategy as the fraction of changes in location that are correctly identified. To study this metric of system performance, we simulate our strategy with walking profiles generated from observations of human walking patterns. Since the accuracy of the underlying WLAN localization system is not our focus, we remove it from our analysis: we do not track the actual location of the user.

Because a walking to not walking transition triggers a localization, there are two circumstances that can cause a missed localization. Either an entire walking period can be identified as non-walking (which may occur if no movement sample occurs within the walk), or an entire non-walking period can be considered a walk. Therefore, for our purposes of accuracy analysis, a basic long-term portrait of a human activity profile consists of the distribution of walk durations and the durations between walks (sedentary periods).

Fortunately, recent work by Chastin and Granat in quantification of human walking points indicates that the sedentary periods between walks are well modeled by a power law distribution [5]. The authors study walking patterns of individuals who are active, sedentary, and of limited mobility. We use their model for healthy active adults as a worst case for missing a span of non-walk periods, and we sample sedentary duration from a Pareto distribution with the parameters they observe. Notably, the nature of sedentary time is much less important than walking time, since sedentary periods are over an order of magnitude longer than walking times and error from missing a non-walk requires missing the entire interval.

Since accuracy is much more sensitive to the length and distribution of walking periods, we do not use a single profile but instead analyze accuracy for four diverse distributions of walk time: constant, uniform, Pareto, and normal. Each of these distributions has a minimum walk length of consideration $T_{\min} = 6\text{s}$ based on the accuracy of our underlying localization system. We select a duration for each that matches total walk time to the mean of a typical adult. In detail, we choose an expected value T for each such that

$$T \cdot E[\text{WalkingPeriodsPerDay}] = E[\text{WalkTimePerDay}]$$

where we obtain $\text{WalkingPeriodsPerDay}$ from the distribution for sedentary periods, and we select WalkTimePerDay based on a study by Bates et al of weekly walking time for 6626 U.S. adults [17].

Name	Sampling Function (sec)
Constant	$T_{\min} + T$
Uniform	$T_{\min} + U(0, 1) \cdot 2T$
Pareto	$T_{\min} + (U(0, 1))^{-2/3} \cdot T/3$
Normal	$T_{\min} + \max(N(T, 144))$

Table 2: Walk Time Distributions We use multiple models of walking time to assess strategy accuracy. $U(a, b)$ represents a uniform random number between a and b , and $N(\mu, \sigma^2)$ a normally-distributed random number with mean μ and variance σ^2 . The Pareto distribution used for sampling has $\alpha = 1.5$.

Using this definition of accuracy and the distributions in Table 2, we consider two primary sources of error: missing an event due to *not sampling*, and *misclassifying* events. We do not adapt the sampling period since our goal is to not miss localization due to prediction error. Therefore, any error due to not sampling will be produced by checking for movement with a period longer than the shortest walk time. This shortest period is $T_{\min} + T$ for the constant distribution and T_{\min} for the others shown in Table 2; periods below this threshold produce no error from a failure to sample for movement. For other sampling periods, Fig. 8 shows a simulation of the dependence of accuracy on sampling period

assuming perfect classification. The choice of a low sample period is necessary to maintain accuracy.

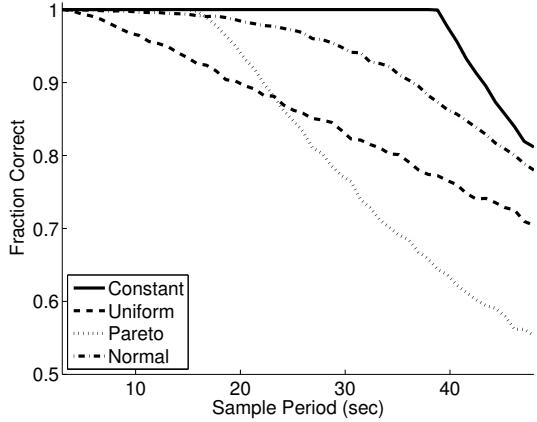


Figure 8: Impact of Sample Period on Accuracy. The impact of sample period on localization accuracy as defined in Sec. 7.2 is shown for the four synthetic walking patterns shown in Table 2. Though the dependence varies, only short sampling periods provide high accuracy.

Such short sampling periods are similar to those used in previous systems. Kim et al [15] sample once every 10 seconds, and Want et al intentionally save power by sampling once every 15 seconds [25], noting that the period is long but sufficient for office environments. You et al [28] use an adaptive sampling period with a non-conformance rate that is typically 10% to 17%, but still poll for movement as often as 2-4 seconds.

Given that a short sampling period is necessary for accuracy, we are most interested in the impact of misclassification. Misclassifying an entire walk is the primary error introduced by the movement-sensing system, and a significant impact of classification error on accuracy would run counter to our goal of not introducing additional error. Fortunately, two properties of movement sensing cause a small impact. First, as we observe in Sec. 6.1, a simple metric can be very accurate in classifying walking/non-walking intervals — the threshold we choose does not misclassify any walk events in our sample dataset. Even for greater misclassification rates, though, the strategy will often sample again within a walk. The probability of an incorrect classification is then the product of multiple incorrect classifications. As simulated and illustrated in Fig. 9, the accuracy of the strategy is robust to more incorrect classification than we observe in our tests.

One additional accuracy-related concern is the outcome if a localization is missed and a user does not move for a long period of time. In this case, the strategy will not relocalize until another walk event occurs. To avoid this situation, the system can check the user’s location in signal-space using a WLAN scan on an infrequent basis when no walking is detected. Although a single WLAN scan consumes approximately 2.2 times as much power as a movement detection, it consumes 10 times less than full localization. The combination of infrequent scanning and relatively low per-event cost produces a low energy impact, as discussed below.

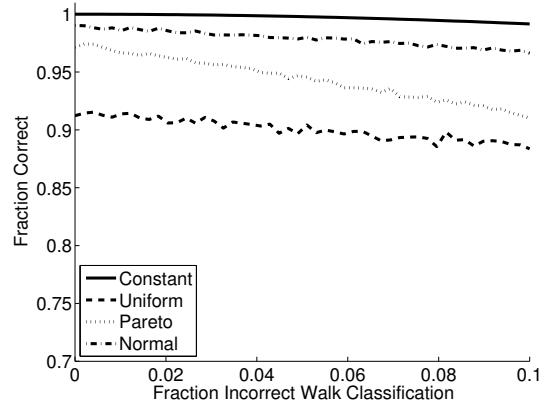


Figure 9: Impact of Misclassification on Accuracy. Localization accuracy is not highly dependent upon walk misclassification in the low range we experience, even for relatively long sampling periods. Here, a sampling period of 17.9s is used for each walk time distribution.

7.3 Power

We evaluate the power consumption of our strategy using the simulation techniques described in Sec. 5 for the same distributions of walk and non-walk times used for the accuracy analysis above. To obtain a worst-case scenario for power consumption, we use a 0% misclassification rate; this ensures that a maximum number of energy-expensive localization events are performed after performing movement detection. We also include a WLAN scan to localize the phone in signal space every five minutes to avoid the potential long periods of incorrect location information introduced by any missed localizations.

Since the strategy parameter that affects power consumption the most is sample period, we compute power usage as a function of sample period for each of the distributions in Table 2. Our baseline for comparison is full (naïve) localization with the same period. Fig. 10 presents the reduced power cost for the movement-sensing assisted strategy as compared to baseline energy expenditure. For the short sampling periods of interest, we observe a computed reduction in power consumption that exceeds 80%. The comparative savings are reduced for longer sample periods primarily due to the cost of inter-event phone sleep, which causes a tapering of device lifetime.

We also investigate the power behavior of our localization strategy for a few individual activity profiles to confirm that our aggregate model for walk times encompasses actual walks. A few data samples provided by PAL Technologies [21] serve this purpose well. These profiles are recorded using a custom device and an algorithm that isolates intervals of sitting, standing, and walking. For example, Fig. 11 shows a simulation of power consumption for the walk pattern of a 41-year-old executive. The power savings simulated for the HTC Dream when using a 20 second sample period (92.8%) and a 10 second sample period (94.3%) are in line with the savings predicted by the walk model and shown in Fig. 10.

Data from other smartphones indicate comparable power savings. We perform rundown profiles of the HTC Magic (aka MyTouch 3G), generate a battery sensor linearization,

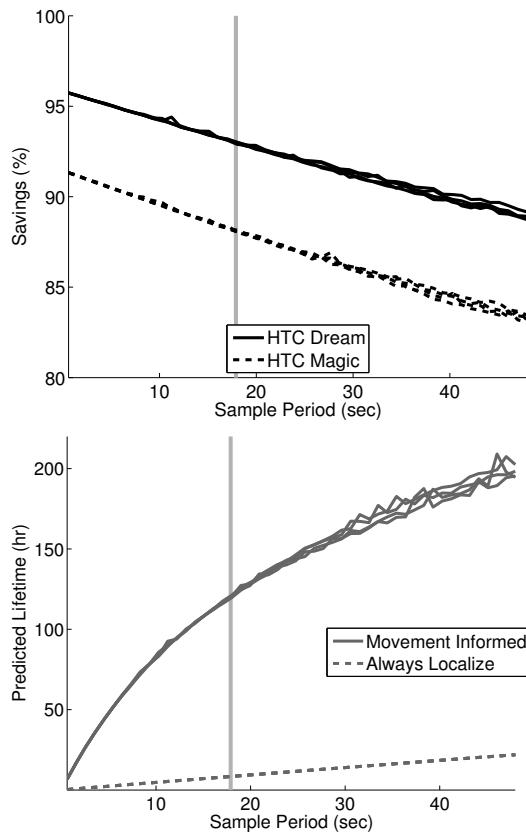


Figure 10: Power Savings. Power consumption for our localization strategy is much lower than that for localization for the same sample period. All distributions for walk time are overlaid for both our strategy (“Movement Informed”) and naïve localization (“Always Localize”). The light gray vertical bar indicates the sampling period of 17.9s that is used in Fig. 9.

and conservatively assume that its sleep lifetime is the same as the HTC Dream we discuss earlier (even though its actual sleep lifetime is longer) [13]. The simulated power savings for this device are illustrated in Fig. 10. Our initial profiling of another smartphone (the Motorola Droid) shows similar promise for power savings, although further work is needed to fully characterize the device, in part since its battery sensor provides a much coarser measurement.

For both the short duration implementation results shown in Fig. 7 (which show a power savings of 79%, possibly due to the association costs we discuss in Sec. 8.2), and in the simulated long-term behavior in Fig. 10 and Fig. 11, we observe a considerable power savings for our strategy. These results indicate great promise for movement-informed localization on consumer smartphones.

7.4 Comparison with Other Methods

Since we focus on improving the battery life of smartphones when localizing, the accuracy and power use of other localization systems on these devices ought to be considered. Unfortunately, a direct comparison is problematic, since the goals and requirements of the technologies differ. For ex-

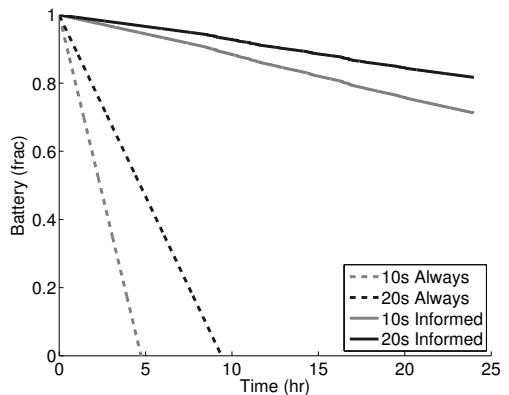


Figure 11: Example Walk Power Use. We use data on walk periods to compute the energy expenditure of always localizing (“Always”) and our strategy (“Informed”) for two different sampling periods. No events are missed by either sampling period.

ample, GPS-based localization is largely orthogonal to our goals: although its accuracy can be 1-5m outdoors [20], it is much less effective indoors, requires a longer time to associate than WLAN and consumes greater than 50% more power than WLAN while active [10].

A more comparable localization mechanism on consumer devices uses the identity of towers on the cellular network. Typical accuracy for these systems is very low (on the order of 50-200m). Prototype systems have achieved a resolution as fine as 2.5m in the best case, but require a dense set of neighboring cells [20]. Accuracy for WLAN-based systems is typically much greater: for example, the underlying framework we use is accurate to within 10m for 94.9% of localizations [4], and multiple systems with greater complexity and training are consistently accurate to within less than 3m [18-20]. The relative energy consumption is dependent upon the type of network used: for a single full localization requiring 1-2KB of transfer, power consumption based on the findings of [3] would be 7J for WLAN, 4J for GSM, and 13J for the 3G radios popular in smartphones.

Efforts most closely related the present work show considerable promise but are not directly comparable. For example, Chen et al reduce CPU power consumption required for offline localization, which is unlike our server-based localization framework [6]. The work of You et al mentioned earlier achieves up to a 68.92% reduction in power consumption as compared to periodic sampling as compared to the greater than 80% reduction we predict. However, these results are not directly comparable: the authors concurrently focus on improving accuracy through predictive sampling. Additionally, both studies use wireless sensor nodes, which have different power characteristics than the smartphones we study. In this spirit, since our emphasis is on consumer devices and their users, we focus our comparison on improvement for their situation in the next section.

7.5 Predicted Benefits

We intentionally study the power consumption of our strategy without other user applications running on the device to control our experiment. However, actual users of a localization service will likely perform other activities like making

calls. Power usage on smartphones in particular is heavily dependent on activity: as easily drawn from the component power consumption values in Table 1 and observed by users, device lifetime can range from less than 10 hours for continuous heavy network and media use to longer than 48 hours for intermittent general use.

To capture the expected impact of user activity, we predict the lifetime of the device by simulating additional power consumption alongside localization. Since battery life varies so widely based on activity, we choose a range of expected battery lifetimes (24–48 hours) for user activity only based on typical user behavior of charging every day or every other day. We then find the average power consumption that produces these lifetimes, and use it as the power use between localizations in a simulation with a 20s movement detection period. Clearly, movement detection and user activity will overlap at times with such a model; however, since movement detection takes only 350ms of every 20s period (1.75%), we find that the effect of overlap on lifetime is less than 4%.

Device Activity	Lifetime (hours)
Only Sleep	493
Movement-Informed Localization	130
Always Localize (only WLAN)	8.7
Typical User Activity	24–48
Typical User Activity + Movement-Informed Localization	20–37
Always Localize (only WLAN) + Movement-Informed Localization	6.5–7.5

Table 3: Predicted Lifetime with Other Activity. Using movement-informed localization in the presence of other power consumption remains advantageous. We use a sample time of 20s for both localization strategies.

Table 3 summarizes the impact of our movement-informed localization strategy on device lifetime. Using our strategy with the 20s sample period used in our studies of power produces a 130h lifetime in the absence of user activity. It would allow users to keep to the same charge schedule: for a one-day original lifetime, it only reduces device lifetime by 4h, and reduces it by 11h for a 48h original lifetime. This time window allows for much more usable localization applications than using existing WLAN-only localization approaches: a 6.5–7.5h lifetime demands multiple charges per day. Therefore, we find that users of our target devices (smartphones) stand to benefit from localization services built around movement-informed localization while feeling little impact on their usage patterns. Some optimizations and study, detailed below, could produce even further benefits.

8. LIMITATIONS AND FUTURE WORK

Although the power savings afforded by movement detection are considerable for minimal cost in accuracy, there are considerations that limit its use. Perhaps the most significant is the potential for a long span of incorrect location information due to a missed localization. Intermittent scanning, as described in Sec. 7.2, can ameliorate this concern, but hardware to perform continuous sensing would improve both power use and accuracy.

8.1 Continuous Sensing Hardware

One of the current limitations of performing movement sensing on smartphones like the one we use is the necessity of waking the CPU to obtain data from the accelerometer. This property causes power usage that, while not as large as localization, is still much larger than device sleep. Additionally, the need to poll the accelerometer makes it infeasible to perform continuous movement sensing. In contrast, many sensor nodes have the capability to wake only when a given sensor’s measurement exceeds a threshold value [8].

Small changes to smartphone hardware could allow for wake-on-accelerometer behavior, thereby permitting lower power use and higher accuracy through continuous sensing. For example, the smartphone we study contains a Qualcomm PM7500 power management IC that manages waking the CPU. It contains its own housekeeping ADC, which could be used in place of the ADCs on the CPU for low-power sensing and triggering the CPU to wake if necessary.

8.2 WLAN Association Cost

We observed a power savings for an actual implementation that was slightly lower than predicted from measurements and simulation. The likely source of this cost is the additional energy expenditure needed to associate with a new WLAN access point. When walking, we model actual behavior by walking between different locations, which had different access points that require authentication. Future work could reduce this reassociation cost — perhaps by maintaining an unauthenticated network limited to localization or performing improved caching of connection information at the session layer.

8.3 Integration with Outdoor Localization

Our focus in this work is on saving power for WLAN localization. Similar movement-sensing approaches are also applicable in outdoor environments, where WLAN localization is often not possible or not preferable. The power cost of common outdoor localization schemes based on GPS and cellular signals is also high, and could benefit from sensor-informed approaches like the one we develop here. Extending the present work to a comprehensive power study and localization system that includes multiple techniques could make it suitable for deployment among a much broader audience. Such a deployment would provide the opportunity to study implementation power behavior on phones that undergo typical use. This work would help address the need for deeper study of the accuracy and power use of the strategy under population workloads.

9. CONCLUSION

We describe an approach to WLAN localization that aims to reduce power consumption. By only localizing when a user has moved, our strategy leverages less energy-expensive sensors to achieve the same sample period as fixed rate localization with a much lower power cost. Our target platform is the recent collection of consumer smartphones, which provide WLAN radios and an accelerometer that we use for movement sensing.

Our study of the power consumption of movement sensing and localization on a smartphone allows us to simulate our strategy for a number of movement profiles. We show that for three cases — long-term models of human walking, recorded activity profiles, and actual implementation —

movement-informed localization reduces power consumption drastically with only a small impact on accuracy. We also show that our methodology can be successfully combined with normal phone user patterns with acceptable power impact and minimal impact on users' charge schedules. Small changes at the hardware level and expansion to other forms of localization could produce even further improvements for this strategy, and lead to power saving for location-based services and their users.

Acknowledgment

The authors would like to thank Andrew Barry, Noah Tye, and Brad Minch for their thoughtful input. Ilari Shafer is supported by an Olin Scholarship.

References

- [1] M. Azizyan and R. R. Choudhury. SurroundSense: mobile phone localization using ambient sound and light. In *ACM SIGMOBILE Mobile Computing and Communications Review*, volume 13, pages 69–72, 2009.
- [2] P. Bahl and V. N. Padmanabhan. RADAR: An in-building RF-based user location and tracking system. In *Proc. IEEE Infocom 2000*, pages 775–784, 2000.
- [3] N. Balasubramanian, A. Balasubramanian, and A. Venkataramani. Energy consumption in mobile phones: a measurement study and implications for network applications. In *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference*, pages 280–293, 2009.
- [4] A. Barry, B. Fisher, and M. L. Chang. A long-duration study of user-trained 802.11 localization. In *Mobile Entity Localization and Tracking in GPS-less Environments (MELT)*, pages 197–212, Sept. 2009.
- [5] S. Chastin and M. Granat. Methods for objective measure, quantification and analysis of sedentary behaviour and inactivity. *Gait & Posture*, 31(1):82–86, 2010.
- [6] Y. Chen, Q. Yang, J. Yin, and X. Chai. Power-efficient access-point selection for indoor location estimation. *IEEE Transactions on Knowledge and Data Engineering*, 18(7):877–888, Jul. 2006.
- [7] G. B. Creus and M. Kuulusa. *Mobile Phone Programming*, chapter 25: Optimizing Mobile Software with Built-in Power Profiling, pages 449–462. Springer Netherlands, 2007.
- [8] Ekahau, Inc. Ekahau Products. <http://www.ekahau.com/products.html>.
- [9] A. Godfrey, R. Conway, D. Meagher, and G. ÓLaighin. Direct measurement of human movement by accelerometry. *Medical Engineering & Physics*, 30:1364–1386, 2008.
- [10] Google Inc. Android source code. <http://source.android.com/download>.
- [11] J. Hightower and G. Borriello. Location systems for ubiquitous computing. *IEEE Computer*, 34:57–66, 2001.
- [12] J. Hightower, G. Borriello, and R. Want. SpotON: An indoor 3D location sensing technology based on RF signal strength. Technical Report 2000-02-02, Univ. Washington, Feb. 2000.
- [13] HTC Corp. HTC Magic. <http://www.htc.com/www/product/magic/specification.html>.
- [14] HTC Corp. T-Mobile G1. <http://www.htc.com/www/product/g1/specification.html>.
- [15] D. H. Kim, J. Hightower, R. Govindan, and D. Estrin. Discovering semantically meaningful places from pervasive RF-beacons. In *Proceedings of the 11th international conference on Ubiquitous computing*, pages 21–30, 2009.
- [16] B. Kölmel and S. Alexakis. Location based advertising. In *The First International Conference on Mobile Business*, 2002.
- [17] J. Kruger, S. A. Ham, D. Berrigan, and R. Ballard-Barbash. Prevalence of transportation and leisure walking among U.S. adults. *Preventive Medicine*, 47(3):329 – 334, 2008.
- [18] J. Letchner, D. Fox, and A. Lamarca. Large-scale localization from wireless signal strength. In *Proc. of the National Conference on Artificial Intelligence (AAAI)*, 2005.
- [19] H. Lim, L. Kung, J. C. Hou, and H. Luo. Zero-configuration, robust indoor localization: theory and experimentation. In *Proceedings of IEEE INFOCOM*, pages 123–125, 2006.
- [20] H. Liu, H. Darabi, P. Banerjee, and J. Liu. Survey of wireless indoor positioning techniques and systems. *IEEE Transactions on Systems, Man, and Cybernetics*, 37(6):1067–1080, Nov. 2007.
- [21] PAL Technologies LTD. Complete record for subject i session 1 day 1. <http://www.paltech.plus.com/examples/index.html>, Aug 2001.
- [22] Skyhook Wireless, Inc. <http://www.skyhookwireless.com/>.
- [23] Sonitor Technologies, Inc. <http://www.sonitor.com>.
- [24] two fourty four a.m. LLC. Locale. <http://www.twofortyfouram.com/>.
- [25] R. Want, A. Hopper, V. Falcão, and J. Gibbons. The active badge location system. *ACM Transactions on Information Systems*, 40(1):91–102, Jan. 1992.
- [26] O. Woodman and R. Harle. Pedestrian localisation for indoor environments. In *Proceedings of the 10th international conference on Ubiquitous computing*, volume 344, pages 114–123, 2008.
- [27] Y. Xu, J. Winter, and W. Lee. Prediction-based strategies for energy saving in object tracking sensor networks. In *Proceedings of the 2004 IEEE International Conference on Mobile Data Management*, 2004.
- [28] C.-W. You, P. Huang, H. hua Chu, Y.-C. Chen, J.-R. Chiang, and S.-Y. Lau. Impact of sensor-enhanced mobility prediction on the design of energy-efficient localization. *Ad Hoc Networks*, 8(8):1221–1237, Nov. 2008.

Interactionless Calendar-Based Training for 802.11 Localization

Andrew J. Barry, Noah L. Tye, Mark L. Chang

Franklin W. Olin College of Engineering

Needham, MA, USA

{andrew.barry, noah.tye, mark.chang}@olin.edu

Abstract—This paper presents our work in solving one of the weakest links in 802.11-based indoor-localization: the training of ground-truth received signal strength data. While crowdsourcing this information has been demonstrated to be a viable alternative to the time consuming and accuracy-limited process of manual training [2], one of the chief drawbacks is the rate at which a system can be trained. We demonstrate an approach that utilizes users’ calendar and appointment information to perform interactionless training of an 802.11-based indoor localization system. Our system automatically determines if a user attended a calendar event, resulting in accuracy comparable to our previously published large-scale crowdsourced deployment. We find that no other user interaction is necessary to train the system to that level of accuracy when calendar data are available. In ideal conditions, this technique can reduce training time by over a factor of six.

Keywords-Location measurement; calendar; crowdsourcing; localization; location representation; location-based services

I. INTRODUCTION

As computing becomes increasingly mobile, location systems, and the applications that leverage them, become more a part of everyday life. With the proliferation of laptops, sensor-laden smartphones, a blanket of wireless access points, and hybrid localization techniques, consumers are beginning to demand location-aware capabilities of their hardware and software. For most of these devices, localizing outdoors with GPS is accurate and provides good coverage. But once indoors, or in an otherwise GPS-denied environment, coverage and accuracy suffer dramatically. Commercial solutions, such as those from Skyhook Wireless [13], attempt to provide location by using a combination of GPS, cell-tower triangulation, and 802.11-based techniques. In GPS-denied environments, the localization relies primarily upon a database of access point signatures acquired by a fleet of 802.11-scanner-equipped vehicles that are limited to scanning from public roads. Therefore, while the quality of location data may be adequate in and around some public indoor spaces, private spaces—large corporate offices for instance—will not necessarily be well covered.

To localize accurately indoors, most proposed systems catalog the received signal strength (RSS) of 802.11 access-points throughout an indoor space, and use those signatures to compare against a user’s current signal scan results to determine location [5], [4], [1], [15]. Most systems require an extensive manual training phase in order to achieve usable

accuracy. We proposed a crowdsourced system [2], similar to [14], [3], that requires very little interaction and minimal dedicated training time, yet achieves very high (room-level) accuracy. One of the oft-cited drawbacks of crowdsourced data is the long effective training time required to achieve good results. In effect, it is the chicken and the egg problem: without good manual training data, users will not utilize (and therefore train) the system; but without users utilizing the system (and actively providing training data), the system will not improve its accuracy.

In this work, we propose a novel solution to this “cold start” problem. By overlaying calendar and appointment data that is tagged with location information (such as a meeting room or office), every report from a user’s device will be automatically bound to a location. Thus, instead of requiring a user to manually associate semantically-relevant room information with a signal scan (*fingerprint*), we do so automatically. Because users’ calendars are not perfectly accurate, we treat every calendar/RSS pair as a noisy sensor. Our use of a clustering algorithm allows us to determine when a user attended or skipped an event, enabling us to extract accurate location data.

II. RELATED RESEARCH

There has been a rich history of published work that attempts to solve the indoor localization problem. Beginning with the Active Badge [15] and Cricket [12] systems, researchers were able to demonstrate reasonable room-level localization. However, these systems required specialized hardware that had to be location-bound and installed by trained personnel.

The growing ubiquity and dense coverage of 802.11 access points saw researchers use them as fixed-location radio beacons. Microsoft’s RADAR [1], and later Haeberlen, et. al. [4], show successful room-level indoor localization using just 802.11 signal strength information. These approaches, while accurate, require the system to be trained to create a database of location and signal strength tuples. This necessitates a substantial up-front investment in time and effort, something that may be a barrier to adopting an indoor localization system. Many more infrastructure-focused approaches are described in [5].

The focus of research then shifts to alleviating the burden of up-front training and high infrastructure costs. Intel Re-

search demonstrated an algorithm that can estimate location through proximal sensing, and expands its known area with continued use [7]. While the initial cost of this system is minimal, the time required to achieve acceptable localization accuracy and wide coverage is high.

Closer to our own work, in approaches described by Teller et. al. [14] and Bolliger [3], training and correction data are collected using a crowdsourced approach rather than via trained personnel. Instead of triangulating estimated absolute positions, these approaches estimate position by returning symbolic representations of physical spaces. This is accomplished by comparing wireless beacon scans to stored *fingerprints* of signal strengths collected at user-annotated locations. The advantages of crowdsourced data sources are obviating the requirement for significant up-front training, and quality training of the system by users in actual physical places of interest.

Unfortunately, purely crowdsourced training data has one significant drawback: the acquisition of training data is completely dependent on user participation. In the next section, we describe our novel approach to solving this “cold-start” problem without requiring any separate intervention from the user.

III. INTERACTIONLESS TRAINING WITH CALENDARS

Almost every medium- to large-scale organization occupies some sort of dedicated physical space; from a floor of a larger building, an entire building, to even an entire campus spanning several buildings. It is not uncommon for these organizations to employ some sort of shared calendaring mechanism, such as Microsoft Exchange, Google Calendar, or Lotus Notes, to help their employees both coordinate work and find one another for interactions. In the latter, the calendar can almost serve as a sensor [9]—providing room-level localization information with some degree of certainty.

Instead of relying on users to manually associate a vector of signal-strengths in *signal space* with a semantically-relevant *physical space*, we can automate training by using the location field of shared appointments in users’ calendars as the physical space annotation for each signal scan. In the ideal case users’ mobile devices, such as laptops and smartphones, would be loaded with signal scanning software, and every user would share accurate and complete appointments in their calendars. With this approach, as people move around to different physical locations to attend scheduled meetings, the system would be trained with accurate data automatically.

There are several benefits to this approach. Most importantly, the “cold-start” issue is largely eliminated. With no infrastructure costs other than the installation of software, users will train the system with accurate data as they progress through their normal day without changing their behavior. With a wide deployment of scanning software, the system will be able to localize accurately in all visited spaces

almost immediately. Secondly, this approach no longer requires any user intervention other than allowing software to be run in the background of their mobile devices to scan and report to a central server. In a sense, it is a modified form of crowd-sourcing, requiring *no actual interaction* from the crowd. Finally, this approach will continually provide new localization data. With the right filtering or clustering mechanisms, the system can continually adapt to changes in the environment, such as the movement of access points, deployment of new access points, the reconfiguring of physical spaces, and the addition of new physical spaces.

Calendar-based training does have some limitations, however. The most obvious is the reliance on location-annotated, shared calendar appointments. If users do not utilize this field in their appointments, or if users do not share their appointment data, system accuracy will suffer. In particular, if a meeting of many people has no annotation of location, that space will not be trained. However, the lack of calendar data would have to be widespread among users to significantly affect the system. As long as there are some users with annotated and shared appointment data that utilize that room, the space will be successfully trained given enough time.

We note that relying on appointment event locations causes the localizer to succeed only in areas where users have specified these events. Locations without such events, such as warehouses, corridors, factories, residence halls, or apartment complexes will exhibit particularly poor performance. Thus, a calendar-trained localizer is most applicable in an office or other structured environment where users regularly schedule appointments and share their calendars.

Perhaps more serious than a lack of data, however, is the case when a user annotates an appointment, but does not take their mobile device to that location. If we treated each calendar/scan intersection as valid training data, this would become a problem. Fortunately, we demonstrate that we can successfully detect and mitigate this situation by employing a clustering algorithm to filter inbound data. So long as we have a plurality of users in the annotated location, the algorithm will ignore these false data.

What cannot be accounted for easily is if all attendees at a meeting with annotated calendars hold the meeting at a different location than noted in the appointment. Since there is no convenient mechanism to override the calendar data other than every attendee changing the appointment location information in their calendar, the system will incorrectly annotate the wrong location. This, however, can be mitigated using the same clustering technique. If there are enough correct localizations in the new location, the moved meeting will be ignored. And even if the new meeting location is incorrectly trained, with enough correct training data, the cluster will eventually move to be associated with the correct location.

IV. EXPERIMENTAL DESIGN

To demonstrate the viability of our system, we perform two experiments. In the first, *Playback Simulation*, we combine our extensive existing localization database (from [2]) with calendar data from our users. We simulate, over time, the training of our system with only calendar/fingerprint intersections—exactly as if we had deployed a calendar-based training system from the beginning—and compare the results with our previous crowdsourced results.

In the second experiment, *Ideal Deployment Simulation*, we motivate a more likely use case: that of wide-scale deployment within an organization that has more consistently location-annotated calendar data, such as a corporate office. We compare these results to all of our previous results.

A. Playback Simulation

We chose to implement this experiment using our currently running localization system’s recorded wireless access point data (tagged with a user ID) and publicly available calendar archives. Using logged data presents an issue, however, in that it is time invariant. Theoretically we could access all two years of data to perform correlations which are not possible when training the system in real time. To emulate a deployed system, we perform all tests over a period of simulated time. As the simulated time moves forward, more data becomes available to the localizer based on when users actually provided valid calendar/fingerprint intersections. In this way, we assess the localizer’s accuracy as it would have appeared throughout the past two years.

To determine appointment and wireless data intersections, we capture a user’s wireless scan data during the entirety of his or her appointments’ durations. We expand appointments that include repetition to consider every instance of the event. One might consider limiting the time range that appointments correlate to scan data to reduce error when users are late or leave early, but we find that the signal-space clustering algorithm is adequate for this task.

To test our system’s performance we captured signal strength data in every unlocked and accessible room in the academic building. In each room we recorded two wireless scans and the true room number. These data allow us to determine the true accuracy of our system and compare the performance of the calendar-trained system to the user-trained system.

It is important to note that users do not use a completely consistent naming schema for rooms, although many users label rooms in a form similar to “AC 204,” where “AC” is consistent and the room number varies. We chose to filter our data to include only rooms labeled in this way, although in the future work section we describe our thoughts on extending this parser to garner both denser and better labeled data.

B. Ideal Deployment Simulation

The *Playback Simulation* does not necessarily demonstrate the expected use-case, as the existing data was trained based on opt-in usage of a small subset of our community. In contrast, we anticipate the benefit of using our calendar-based approach is for *new* installations of localization systems into organizations that utilize a calendaring system throughout the entire organization. In order to demonstrate the potential effectiveness of our approach, we simulate an idealized environment.

One idealized environment might be a corporate setting, determined both by the personnel and their behavior, as well as the physical spaces. We envision employees of this organization to utilize company-issued mobile devices—both smartphones and laptops—have average work days that include some meetings with others, and have location-annotated appointments in their calendars. Deployment of the system then only consists of installing a scanning and reporting application on all devices. We argue that with managed IT services and an expectation of calendar usage, these assumptions are within reason for any organization that would consider deploying a localization system.

In this simulated environment, we will show that we can achieve very high rates of accuracy in an extremely short time span with nearly zero investment in training and infrastructure. In essence, we obtain all the benefits of crowdsourced data without the long time-lag often associated with the approach.

We choose to use our institution’s course schedule to simulate this type of environment. For one semester, we determined when and where every class was taught along with each classes’ enrollment information. We then ran four simulations, assuming that 10%, 25%, 50%, and 100% of students were using our system. In each case, we assumed that 35% of participating users did not attend lecture. In this way, we simulated the high data-density we expect to occur in a corporate environment while retaining an appropriate amount of erroneous reports.

To provide our simulated students with ground-truth data, we used the combination of our user and calendar generated data. Whenever a simulated student “gathered” data, we copied the appropriate room’s data from those sources. For the 35% of users localizing in incorrect locations, we selected a random point.

As in our existing system, each simulated user localized every 5 minutes during all events regardless of if he or she was attending. With these data providing a simulated ground truth over time, we performed the same operations as in the Playback Simulation, including signal-space clustering, localization, and accuracy testing.

V. SYSTEM ARCHITECTURE

As in [2], we use a client-server architecture to perform localizations. Clients capture data and calendar information

which are transmitted to a server that determines ground truth, trains the system, and estimates locations. We extract location data from the users' calendars which, when combined with relevant wireless signal strength data, is used to train the localizer. Thus, to be used as training data, we require an event to be ongoing while signal strength scans are recorded.

A. Deployment Site

While we hypothesize that this system would perform best in a corporate setting, we use a college campus to perform our study. Olin College is a small residential engineering college with approximately 300 students and a campus encompassing over 300,000 square feet. Classes take place almost exclusively in a single building, so we limit our test to that area where calendar events are dense. We collect calendar data from a shared Microsoft Exchange system and wireless network data from students who chose to participate in the previous study focused on active user-entered localization data. To participate, students run a small client program on their notebook computer that records wireless access point data along with a user ID and timestamp.

B. Localization Method

Like in [2], [1], we use a received signal strength (RSS) indicator to perform a simple nearest neighbor search in signal space for localization. Essentially, the localizer traverses every known data-point and computes the Euclidean distance in signal space to the point under consideration. The localizer returns the point corresponding to the minimum computed distance. We chose to continue using this localizer to show that the interactionless component of the system does not require a specialized localization algorithm, and could easily be extended to the more accurate and complex systems like those in [6] and [11].

C. Calendar Integration

To acquire calendar data, we accessed users' publicly shared Microsoft Exchange calendars, converted them into the standard iCal format, and imported the iCal files into the system. While our users run Microsoft Outlook, this conversion step generalizes the system, allowing it to be used with almost any modern calendaring application. Each appointment entry provides us with some or all of the following: a user ID, a time range, and a location string. Calendar appointments that do not include all of these elements lack a critical piece of data and are thus not useful for training.

1) *Calendar/802.11 Intersection:* In addition to a user ID, a time span, and a location string, an 802.11 scan must have occurred during the appointment for it to be useful as training data. As our localizer software running on both laptops and Android-powered mobile devices reports a wireless signal strength fingerprint once every five minutes,

we are able to capture, on average, twelve points per hour of appointments.

2) *Calendar Filtering via Signal-Space Clustering:* Users are not always located where their calendar indicates. Calendars are often double-booked, users do not attend all meetings, meetings move without being updated in the system, and users do not bring their wireless devices to all appointments. Our crowdsourced localization system reports that users are located where calendar events specify 68% of the time. Therefore, to extract useful training data, we must filter the calendar appointments.

As noted in [10], even without an existing localization mechanism, the collected wireless signal strength data provides insight. In general, a user either attends a meeting in the location specified, or does not. If not, the user's actual location has little correlation to where the appointment indicates. Thus, given a reasonable meeting attendance rate, we find that clusters of accurate data are easy to find in signal space. We are able to prune almost all events where users do not attend meetings, providing us with interactionless and accurate training data.

To filter the raw data, we use a clustering algorithm based on distance in signal space. We have 76 wireless access points on campus so we cluster in 76-dimensional space. For each location we compute a mean vector by taking the arithmetic mean of each component in the individual vectors. We then calculate the Euclidean distance between each vector and the mean vector. Unlike in [10], the inherent size of a calendar dataset allows us to use a majority voting system to identify outliers, implemented by rejecting entries with a distance greater than its associated location's median. This is a harsh filter but, like [8], we have found that relatively few erroneous points can significantly reduce accuracy. Listing 1 shows this algorithm in pseudo code form.

```

for location in All_Locations:
    meanVector = ComputeMeanVector(All_Events[location])

    for event in All_Events[ location ]:
        distances.append(EuclideanDistance(event, meanVector))

    medianDistance = ComputeMedian(distances)

    for event in All_Events[ location ]:
        if EuclideanDistance(event, meanVector) < medianDistance:
            Filtered_Events.append(event)

```

Listing 1. Simple Signal Space Clustering Algorithm.

To simulate a time-dependence, we ran this algorithm once for each day, retraining and retesting the system on each iteration.

VI. RESULTS: PLAYBACK SIMULATION

A. Time-Based Analysis

We measure the quality of a localization system with two statistics, *coverage* and *accuracy*. Coverage describes the

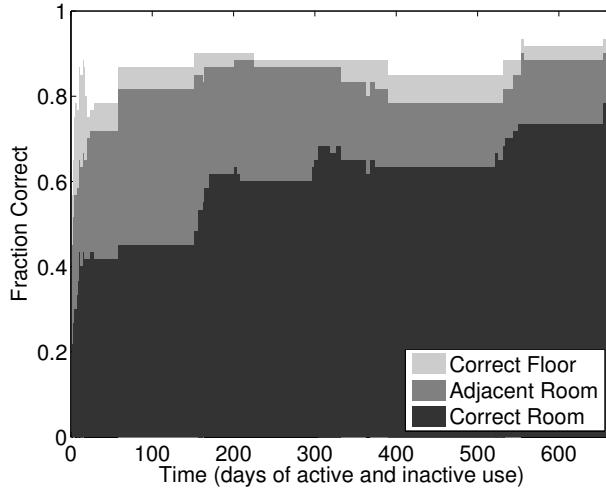


Figure 1. Accuracy of calendar-trained system over time. Dark gray indicates fraction of localizations resulting in a correct prediction. Middle gray indicates fraction of predictions in a correct or adjacent room, and light gray indicates fraction of localization predictions on the correct floor.

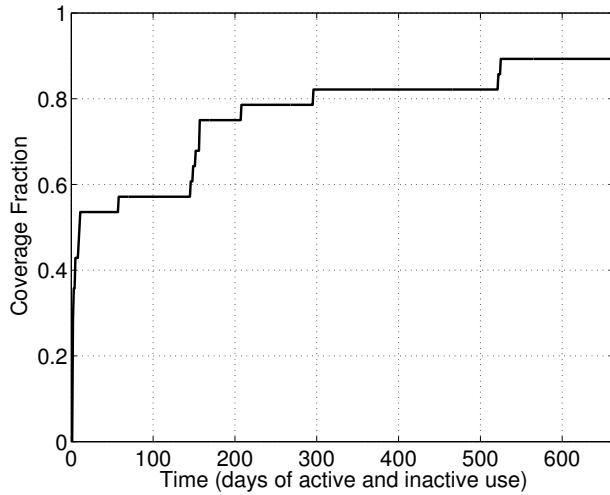


Figure 2. Fraction of rooms with training data over time. As users have meetings in new locations, the localization system’s coverage improves. The dramatic increase near Day 145 is a result of students returning to campus with a new calendar schedule. Note the strong correlation between coverage and accuracy (Figure 1).

fraction of locations with training data, while accuracy denotes how often the localizer correctly determines position.

We find that the system achieves 50% room-level accuracy (85% in adjacent rooms) after 148 days of active use (Figure 1). We define the system to be in active use when users are resident on campus, thus excluding intersession periods such as winter break. When the system is not in active use, its accuracy neither rises nor falls.

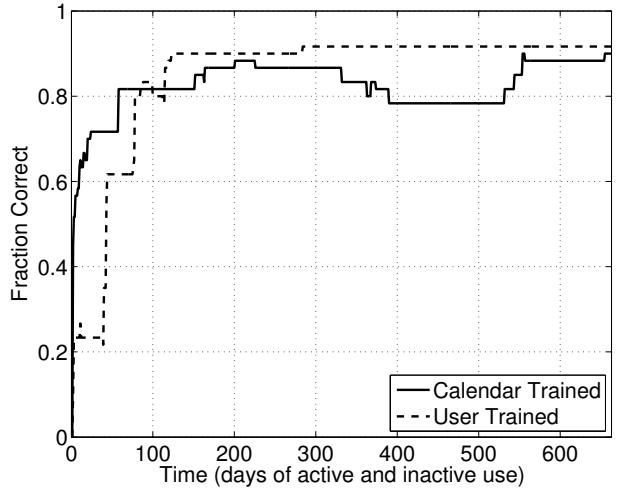


Figure 3. Comparison of correct or adjacent room accuracy over time between calendar-trained (solid) and user-trained (dashed) systems. Note that, unlike the user-trained system, the calendar based localizer is not monotonically increasing because its additional data is not always correct. The horizontal segments in the user-trained accuracy (before Day 100) are related to development periods when the system was not in active use.

Of the 28 rooms we localized, 89% had at least one signature bound to that location by the end of our experiment. Figure 2 shows how coverage in the calendar-trained localizer changes with time. After 372 days of active use, no signatures for new rooms were added to the database.

Three rooms remained without coverage: a biology lab, a professor’s research lab, and a mechanical project space. These rooms had no data because students rarely use their computers there as there are very few classes, labs, or meetings scheduled there.

B. Comparison to User-Trained Localization

The user-trained system took 18 days of active use to localize to the correct room in 50% of trials and 47 days of active use to localize to the correct room or an adjacent room in 85% of trials. Figure 3 compares the accuracy over time of this crowdsourced localizer with the calendar-trained localizer. Note that Figure 3 includes development periods (visible as plateaus near the beginning of the plot) where the user-trained system was unusable and therefore we do not regard them as periods of active use. The above statistics do not include these days.

The accuracy of the user-trained system stabilized after 55 days of active use, at 70% room-level accuracy and 90% including adjacent rooms. The calendar-based system proved slower than its user-trained counterpart, requiring 381 days of active use to achieve equivalent room-level accuracy. It took only 71 days of active use, however, to achieve adjacent room-level accuracy within 10% of the user-trained system.

The localizer’s accuracy decreases when data are added

that cause the clustering algorithm to fail. This occurs when a significant number of users are not present at their appointments but are still providing data (for instance, by leaving their laptop running at a different location), skewing the values of location cluster means.

One phenomenon observed in the crowdsourced system is that users seem to lose interest, and the number of new binds decreases. As a result, accuracy plateaus. In contrast, the calendar-trained localizer continually accumulates training data (Figure 4). Thus, the training set is in constant flux, ensuring that the system will adapt to variations in the environment such as new network access points, additional furniture, changing architecture, and even relabeled rooms.

We find that a disadvantage of the calendar-based system is that, while long-term accuracy continually improves, short-term accuracy fluctuates. The system's constant stream of new data implies that some incorrect data is also being added, potentially reducing accuracy until enough correct data overcomes the issue. Users rarely submit bad data to the user-trained localizer, explaining why user-trained accuracy rarely decreases.

Accuracy in the calendar-trained localizer does drop after Day 373. This could be for several reasons. As already mentioned, since the calendar-trained system's data is automatically harvested rather than explicitly entered by users, bad data is much more likely. Wireless access points are sometimes moved by network administrators to improve wireless performance. This change not only invalidates old data points but also makes the clustering filter incorrectly discard valid data points. The access points also have automatic and manual gain control, the use of which has a similar effect as physically moving the access points. Nevertheless, the localizer adapts and recovers after 37 days of active use, and after 11 more days of active use the localizer is more accurate than it was before the dip.

C. Clustering Analysis

Our clustering filter is a deliberately aggressive algorithm, discarding approximately one third to one half of all incoming data. Ideally we would evaluate the algorithm's performance based on ground-truth data, but we do not have a mechanism to determine when users attended their calendar events. As a substitute, we use our user-trained localization system as ground truth, but only compute statistics for floor-level accuracy to mitigate the impact of the user-trained system's localization error on our results.

On the final day of simulation, we find that the algorithm accepts 85% of the valid data and discards 76% of the incorrect data, discarding 33% of total data. Clearly, we discard a significant amount of correct, useful data. This state is acceptable, however, because incorrect training data is highly detrimental while collecting more intersections costs little.

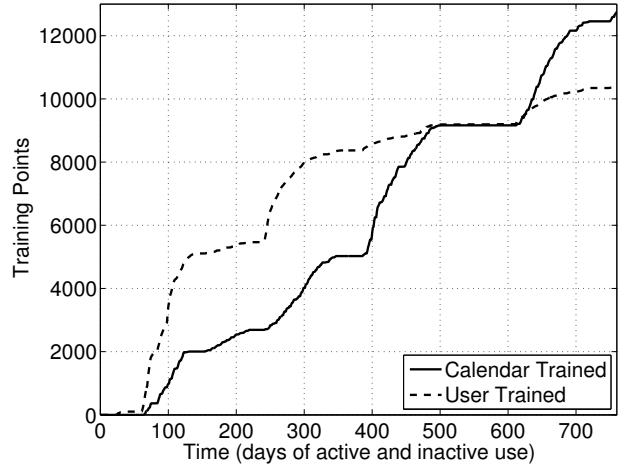


Figure 4. Comparison between calendar-based and user-based binds over time. Notice that as the system becomes more accurate, users bind less. Thus, we expect the user-trained system to become less reactive to environment changes as it ages, while the calendar-based system will continuously acquire new data. We are not surprised to see few binds for both systems in the summer and January terms when students are not on campus.

D. Experimental Details

We recorded over 1.8 million wireless scans over a period of more than two years from 278 users, of which 99 had published calendars. These shared calendars contained 47,955 events and 93,005 single event-occurrences (many events repeat). Users with public calendars accounted for 725,533 wireless scans with only 85,237 (11.7%) of these scans overlapping with a calendar event. We were able to parse location data on 28,732 (33.7%) of these. We expect that in a more favorable environment, such as in an institution-sponsored program, the amount of useful raw data available would be substantially higher.

Based on the localization data from our previously published system, we find that users are in the location that their calendar predicts in approximately 65% of cases. We evaluate our clustering algorithm based on that system's output to estimate how successful the algorithm is. In other words, we can use our previous system to determine how well our clustering algorithm is performing.

E. User Behavior

In our calendar data we see a less typical mass interaction curve than in our previous crowdsourced approach. One quarter of users provide most of the data, however this group is much larger than that in other mass interaction applications[16]. We find that 25% of users provide 52.3% of all calendar data while 25% contribute 56.3% of all wireless network data. These data are more level than those of our user-trained system and of MIT's crowdsourced localization system [14].

We find that recurring appointments are critical to the system's success, accounting for 84.8% of all appointment/wireless data intersections and 87.1% of intersections determined to be in the correct place by the user-trained system. Users' individual reliability varies greatly, giving an approximately uniform distribution when considering users with a significant number of intersections.

VII. RESULTS: IDEAL DEPLOYMENT SIMULATION

Accuracy, measured only for rooms where classes are scheduled, increases much more quickly in the data-rich environment. As the number of participants increases, the rate of accuracy increase improves in parallel (Figure 5). We note that our signal-space clustering successfully mitigates the increasing number of incorrect location updates that accompanies the additional users.

With 100% participation, we find that the system takes 8 days to achieve full accuracy at the room level, compared to the approximately 150 days for the non-ideal system. As expected, it takes longer with fewer users, taking 15, 50, and 64 days for 50%, 25%, and 10% participation, respectively. Notably, we find that to achieve stable accuracy at the floor level, all simulations require less than one week of training data.

Even in the case of only 10% user participation, we find that the data-rich environment achieves similar accuracy in less than one half of the time required for the user-trained system (15% of the time required for the demonstrated calendar-based system). With increased participation, this duration decreases further, to less than 15% and 5% of the user-trained and calendar-trained systems respectively.

While the simulation's time-to-accuracy exceeds that of the two other methods, its final-value accuracy does not. This result stems from the simulator's reliance on the combined data of the user and calendar trained systems. The final accuracy of the user and calendar trained systems is limited to all available data, a limitation that the simulator faces as well.

VIII. FUTURE WORK

A. Integration with Other Data Sources

There are a multitude of other data sources that could improve our system's localization performance. We are considering integration with a room-scheduling system and a student-course scheduling system, both of which provide user-identified location information at a specific time. We are also considering data sources that do not traditionally supply location information, such as a desk telephone. Answering the telephone indicates that a user is most likely at his or her desk, allowing us to bind all data close in signal space to the user's workspace. We are considering this same strategy for capturing data from desktop computer idle times, screensavers, instant messaging, and even printing applications.

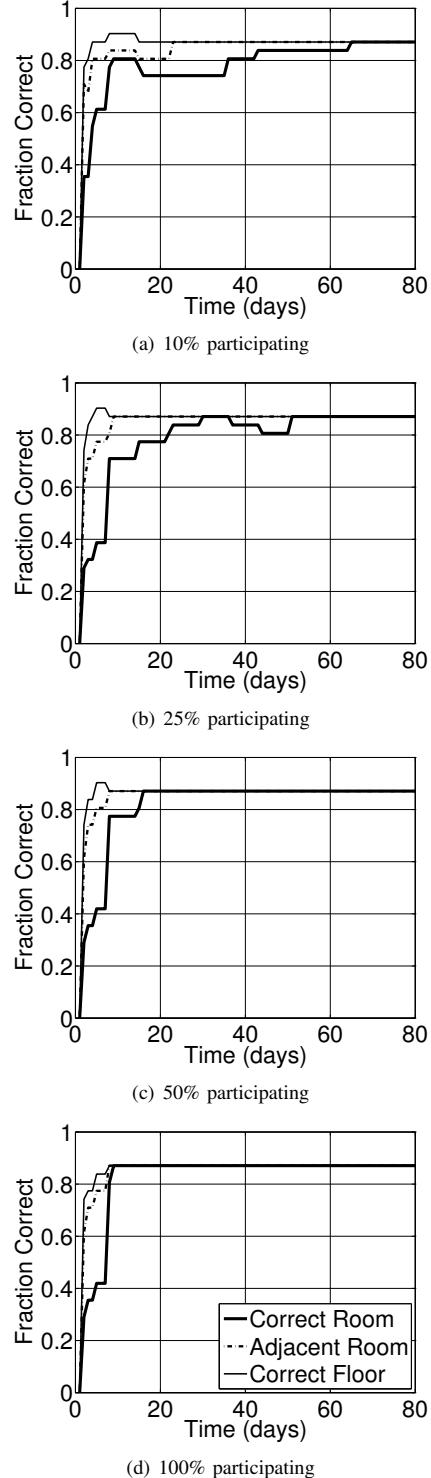


Figure 5. Simulation results for 10%, 25%, 50%, and 100% of users participating. Solid bold indicates correct room, dashed indicates adjacent rooms, and solid thin indicates correct floor. Notice that the system becomes accurate more quickly as the number of users increases.

It is important to note that this type of bind can be extremely helpful to the system. Users rarely schedule meetings at their own desk, but there is likely to be a large number of wireless scans in the area, since the user spends a significant amount of time there. Thus, extracting a bind from these non-traditional data sources allows us to correlate the past data taken at the user's desk, dramatically increasing the number of binds and thus accuracy in the area.

B. Calendar Correlation

A second method of improvement might stem from correlating calendars. We propose studying users' calendars in groups to determine which meetings users are likely to attend. Employees are very likely to attend meetings with their boss, but less likely to attend meetings where they are less important. Such correlation could be as complex as necessary or as simple as the difference between the user being in the "To" or "CC" field in an appointment notification.

C. User-Defined Location Labels

We might expand our coverage by removing the restriction that appointments must label locations in a defined format. However, user-defined names introduce the potential for duplicate location labels, uninformative names, etc. Signal-space analysis might allow us to identify duplicate labels, which could help mitigate the issue.

D. Identifying Unnamed Locations

A final technique we are considering allows the system to name locations that never appear in calendar data. These locations tend to be areas where one works alone, such as an office or dormitory. We propose correlating long stays in these areas as a way to determine their status. For example, at a workplace, a laptop might be left on overnight in a user's office. A college student might turn his or her laptop off late at night before going to sleep. If these cases occurred in a predictable and identifiable manner, we might name them "[Username's] Office" or "[Username's] room," depending on application. We believe that this type of inference, combined with new data sources and improved correlation systems, can significantly improve the performance of the localizer described here.

IX. CONCLUSION

We have proposed a novel method for training an indoor wireless localization system that utilizes location-annotated calendar data to perform interactionless training of the system. Our experiments have shown that this approach yields equal or better accuracy than crowdsourced training approaches, yet requires a fraction of the training time. We have performed experimental verification of our approach on a working database of location fingerprints, and have motivated and demonstrated the performance of more typically expected deployment scenarios.

We believe this approach is the beginning of a larger set of novel approaches that incorporate a wide range of inputs as location sensors to further improve localization accuracy, reduce training time, and provide even more robust localization in all locales.

ACKNOWLEDGEMENTS

We thank the community at Olin College for their ideas, testing, and feedback. Andrew Barry and Noah Tye are supported by F. W. Olin Scholarships.

REFERENCES

- [1] P. Bahl and V. Padmanabhan. RADAR: An in-building RF-based user location and tracking system. In *Proc. IEEE Infocom 2000*, pages 775–784. Proc. IEEE Infocom 2000, IEEE CS Press, 2000.
- [2] A. Barry, B. Fisher, and M. L. Chang. A long-duration study of user-trained 802.11 localization. In *Mobile Entity Localization and Tracking in GPS-less Environments (MELT)*, pages 197–212, Sept. 2009.
- [3] P. Bolliger. Redpin - adaptive, zero-configuration indoor localization through user collaboration. In *MELT '08: Proceedings of the first ACM international workshop on Mobile entity localization and tracking in GPS-less environments*, pages 55–60, New York, NY, USA, 2008. ACM.
- [4] J. Haeberlen, Flannery, et al. Practical robust localization over large-scale 802.11 wireless networks. In *Proceedings of the Tenth ACM International Conference on Mobile Computing and Networking (MOBICOM)*, Sept. 2002.
- [5] J. Hightower and G. Borriello. Location systems for ubiquitous computing. *Computer*, 34(8):57–66, Aug. 2001.
- [6] T. King, S. Kopf, T. Haenselmann, C. Lubberger, and W. Efeldsberg. Compass: A probabilistic indoor positioning system based on 802.11 and digital compasses. In *Proceedings of the First ACM International Workshop on Wireless Network Testbeds, Experimental evaluation and Characterization (WiNTECH)*, Los Angeles, CA, USA, September 2006.
- [7] A. LaMarca et al. Self-mapping in 802.11 location systems. In *UbiComp 2005: Ubiquitous Computing*, pages 87–104. LNCS 3660, Springer, 2005.
- [8] M. Lee, C. Yu, H. Yang, and D. Han. Crowdsourced radiomap for room-level place recognition in urban environment. In *IEEE PerCom Workshop on Smart Environments (SmartE)*, 2010.
- [9] E. Mynatt and J. Tullio. Inferring calendar event attendance. In *IUI '01: Proceedings of the 6th international conference on Intelligent user interfaces*, pages 121–128, New York, NY, USA, 2001. ACM.
- [10] J. Park, B. Charrow, D. Curtis, J. Battat, E. Minkov, J. Hicks, S. Teller, and J. Ledlie. Growing an Organic Indoor Location System. In *8th Annual International Conference on Mobile Systems, Applications and Services (MobiSys)*, June 2010.

- [11] I. Paschalidis, K. Li, and D. Guo. Model-Free Probabilistic Localization of Wireless Sensor Network Nodes in Indoor Environments. *Mobile Entity Localization and Tracking in GPS-less Environments (MELT)*, pages 66–78, Sept. 2009.
- [12] N. Priyantha, A. Chakraborty, and H. Balakrishnan. The cricket location-support system. In *6th Ann. Int'l Conf. Mobile Computing and Networking (Mobi-com 00)*, pages 32–43. ACM Press, 2000.
- [13] Skyhook Wireless. <http://www.skyhookwireless.com>.
- [14] S. Teller et al. Organic indoor location discovery. Technical Report MIT-CSAIL-TR-2008-075, MIT, Dec. 2008.
- [15] R. Want, A. Hopper, V. Falcao, and J. Gibbons. The active badge location system. *ACM Transactions on Information Systems*, 40(1):91–102, Jan. 1992.
- [16] S. Whittaker, L. G. Terveen, W. C. Hill, and L. Cherny. The dynamics of mass interaction. In *In Conference on Computer-Supported Cooperative Work (CSCW)*, Nov. 1998.

Work in Progress: Synthesizing Design, Engineering, and Entrepreneurship Through a Mobile Application Development Course

Mark L. Chang

Franklin W. Olin College of Engineering, mark.chang@olin.edu

Abstract – In this paper, we describe our experiences in designing and delivering a course that blends together design, engineering, and entrepreneurship through the use of mobile devices. The significance of this work is in advocating for and demonstrating the motivational and educational benefits of using a mobile platform, and describing how to utilize the mobile marketplace to provide an authentic, real-world experience across these three domains.

Index Terms – Android, design, entrepreneurship, mobile, software engineering.

DESIGN AND ENTREPRENEURSHIP CURRICULUM AT OLIN COLLEGE

The Olin College curriculum strives to integrate new modes of thinking into the traditional engineering foundation. We consider user-centered design principles and entrepreneurial thinking of equal importance to technically rigorous engineering and a broad set of experiences in arts, humanities, and the social sciences. Currently, a four-year design stream is embedded in the curriculum, and entrepreneurship courses are required of every student.

While specific coursework in these domains has strong value in developing foundational expertise, without cross cutting courses that synthesize these principles, we will miss unique educational opportunities. We believe that in the electrical and computer engineering and computer science disciplines, the modern mobile device has matured into a platform that provides rich experiences in these three areas.

MOBILE APPLICATION DEVELOPMENT

To leverage the growing student interest in smart phones such as Apple's iPhone and Google's Android, we created a course, *Mobile Application Development* [1], in the Spring of 2009. The objective of the course is to investigate the mobile landscape through the lenses of design, entrepreneurship, and engineering. We draw inspiration for this course from Hal Abelson's *Building Mobile Applications* at MIT [2], Stanford's *CS 193P: iPhone Application Programming* [3], and Maneesh Agrawala's *CS160: Introduction to Human Computer Interaction* at Berkeley [4].

The course leverages required coursework at Olin in software design, user-oriented design, and business and

entrepreneurship, and applies these concepts in a mobile context. Technically, students learn about all aspects of the Google Android platform and SDK. Through design, students engage in ideation, user study, and lightweight rapid prototyping. To experience entrepreneurship, students unpack the mobile market space to find points of opportunity. In the final project for the course, students work in teams to develop commercially viable Android applications and release them to the public through the Google Android Market.

A significant component of the course is interacting with representatives from all facets of the mobile industry. We are fortunate to be in an area where there is a lot of mobile business activity, and have been successful in attracting CEOs, expert designers, marketing professionals, and other mobile industry leaders. These guest lectures provide the students with detailed, real-world insight into the inner workings of devices, markets, and applications that their generation has found indispensable for work and recreation.

PRELIMINARY FINDINGS

I. The mobile market ecosystem provides the lowest barrier to a real, interdisciplinary world of design, engineering, and entrepreneurship

Advances in three key areas have made the current generation of "smartphones"—such as the Apple iPhone, Google's Android, and the Palm Pre—a uniquely-positioned platform for educators to adopt for interdisciplinary coursework. First, the hardware has matured to include multitasking operating systems with gigabytes of storage and ubiquitous high-speed connectivity to the Internet. Second, developer support in the form of first-class language support, powerful and mature development environments, and vendor support of developers independent of the cellular carrier, provides developers the tools to create compelling user experiences. And third, successful direct-to-consumer distribution channels, providing turnkey application publishing and sales to any developer, such as the Apple App Store, Google's Android Market, and others.

The barrier to entry into this unique intersection of disciplines can be as low as \$25USD, the cost to create a distribution account with Google's Android Market.

Session T1A

II. Successfully leveraging the mobile market requires students to cross discipline boundaries

The interdisciplinary educational impact from these three advances is profound. First, the hardware allows for novel, connected, computationally expensive, and beautifully rendered applications that allow good design principles to be expressed and realized. Second, the development environment is simple, with clean abstractions and a wealth of libraries to build upon—students can build applications quickly without straying too far from their foundational computer science knowledge. Third, the distribution channel allows student's work products to escape into a thriving marketplace effortlessly, providing an opportunity to put market analysis, advertising, and revenue model concepts to the test.

The Apple App Store model of the on-device application marketplace is, by all accounts, a runaway success. With over 140,000 applications available, however, success is not guaranteed. With so many resources from industry being devoted to these marketplaces, students must truly embrace all facets of developing for customers to be successful. This includes identifying **business** opportunities in the mobile space, providing novelty and utility through **design** and **entrepreneurship**, delivering functionality and stability through good **engineering**, and successfully promoting and marketing the product or service as an **entrepreneur**.

III. One cannot underestimate the allure of working with a modern smartphone

Both semesters that the course has been offered, student interest has been very strong. Our engineering students, by and large, are excited by the opportunity to work with cutting edge technologies. Advanced mobile phones such as the iPhone or an Android device represent the leading edge of alluring techno-gadgets. These devices present a unique intersection of desirable devices that can be programmed. Loaning students devices and giving them the power to control their devices has received overwhelmingly positive feedback from students. But perhaps the greatest attraction to the course is the opportunity to participate in the incredibly trendy and seemingly lucrative mobile application marketplace. Student desire to see their applications being downloaded onto devices is positively amazing.

Taking all of these motivators together, student commitment to the course is astounding. While the initial Android SDK learning curve may be a bit steep for many, students are remarkably eager to learn new features of the SDK and tackle increasingly complex assignments. Anecdotally, students are driven by a simple desire to produce something successful that might be used by millions of users. What is unique about a modern mobile platform is that they are acutely aware that it is within their grasp.

IV. Blending design, engineering, and entrepreneurship in an authentic way presents a staffing challenge, or an opportunity for team teaching

While we have argued that the modern mobile platform gives students the opportunity to blend these three disciplines together, it is not a simple task to be the instructor in this context. Rarely does one individual possess significant expertise in all three areas. While this may be viewed as a pitfall, it should be considered yet another opportunity, this time, for team teaching.

Alternatively, importing these skills through guest speakers from academia and industry provides an authentic voice in class. Finally, enforcing stronger prerequisites may be a necessary fallback to ensure adequately prepared students when staffing presents as a barrier.

V. The end goal of market release is a rare opportunity for closure and feedback

In our course, a requirement is to release the final application to the public through the Google Android Market. This presents several key educational opportunities:

- Student motivation is significantly higher when their work “goes public”. With millions of handsets sold, the “public” is very large and motivating.
- Students desire to learn good software engineering principles, such as revision management, testing methodologies, and bug tracking.
- There is a unique opportunity for feedback on both the engineering quality and the design decisions. Rarely do students get feedback of this magnitude and reality on the decisions they make in open-ended projects.

We have described our mobile application development course design and some preliminary results from two iterations. We are actively developing the course with industry partners and widening the internal staffing to bring a broader set of expertise to bear. We hope to expand the course to encompass more student-centered technologies. Our next iteration will likely mix mobile and web applications using the same framework.

REFERENCES

- [1] Chang, M., “Mobile Application Development”, *Franklin W. Olin College of Engineering*, <http://mobdev.olin.edu>
- [2] Abelson, H., “Building mobile applications with Android”, *MIT*, <http://people.csail.mit.edu/hal/mobile-apps-spring-08/>
- [3] “iPhone Application Programming”, *Stanford University*, <http://cs193p.stanford.edu>
- [4] Agrawala, M., “CS160 User Interface Design”, *University of California, Berkeley*, <http://vis.berkeley.edu/courses/cs160-sp10/wiki>

AUTHOR INFORMATION

Mark L. Chang, Assistant Professor of Electrical and Computer Engineering, Franklin W. Olin College of Engineering, mark.chang@olin.edu.

Work in Progress - Impact of Early Design Instruction on Capstone Experiences

Mark L. Chang and Jessica Townsend
Franklin W. Olin College of Engineering, Needham, MA
mark.chang@olin.edu, jessica.townsend@olin.edu

Abstract - In the Olin College curriculum, students have significant, early, and continuous exposure to user-oriented design principles. As a result, our students have a very user-centered approach to problem solving that has affected our yearlong, industry-sponsored capstone in several ways. We have reflected on five years of capstone engagements in order to learn how our program has changed because of the design emphasis in our curriculum. The significance of our work is to inform the many departments that are already undertaking design-centric curriculum reform on how they may modify their capstone experiences to best take advantage of new student understanding, and what to expect when using design principles to engage industry problems.

Index Terms – Capstone design, user-centered design, student consulting, sponsored programs.

DESIGN CURRICULUM AT OLIN COLLEGE

Traditionally, capstone design experiences have been focused on delivering technically rigorous and challenging culminating experiences specific to an engineering discipline. ABET requires engineering programs to include a major design experience and defines engineering design as “the process of devising a system, component, or process to meet desired needs” [1]. At Olin College, that definition has been expanded to include identifying and defining problems, applying technical and non-technical knowledge and skills, developing an understanding of design processes, exploring contextual factors that contribute to design decisions, and mustering the necessary resources to realize solutions [2].

The Olin design curriculum provides students with exposure and experience in these areas. Our first-year students take Design Nature, a course that introduces the traditional aspects of the design process (as described by ABET) through the development of a bio-inspired system, focusing on developing additional skills in teaming, planning, and fabrication. In the sophomore User-Oriented Collaborative Design course, student teams observe and engage potential product users to develop a deep understanding of their values and needs in order to seek holistic solutions integrating user and functional perspectives. In their third year, students choose a design depth course, with such offerings as Distributed Engineering

Design, Sustainable Design, Design for Manufacturing, and Human Factors and Interface Design.

In the final year, students participate in the Olin Senior Capstone Project in Engineering (SCOPE), a sponsored yearlong project. The kind of work our students engage in varies widely in terms of design content, from the very technical and specific—building an autonomous vehicle for the DARPA Grand Challenge—to the very user-focused and general—improving the quality of life for senior citizens. How our students, with their design- and engineering-heavy coursework, engage these problems differently than expected motivates our study. As our capstone industry partners do not always embrace user-centered design thinking, we have encountered many successes and pitfalls that bear mentioning. We discuss our preliminary findings based on examining the past five years of student and sponsor experiences surrounding the design component of projects.

USER-CENTERED DESIGN APPROACH TO CAPSTONE PROJECTS

The most notable effect of our students’ early design instruction on their approach to their SCOPE capstone projects is the inclusion of a user-centered design approach. Since private sector companies sponsor the majority of our senior capstone projects, the work is often product-focused, with true “real-world” complexity and context.

Some of our sponsors have a fairly clear technical goal that they would like achieved. They have already justified the need for the project through their own market or user research and prefer the student team to work only on the more traditional engineering components. An example of a recent project in this category is an effort to implement a vision-based road-following algorithm for an autonomous vehicle. In this case, the work is highly technical and the main challenges are centered around system and component design.

For other projects, it can be less clear. A good example has been when sponsors have a piece of working technology that they have acquired or invented, but have not found an effective monetization route. In these cases, the team must both determine the capabilities and limits of the technology and focus on identifying and studying the potential users and or market to provide the sponsor with a full picture of how to leverage the technology.

Session T1A

Olin students have also worked on capstone projects where they have been tasked with developing a product or service to better serve a particular group, such as Alzheimer's patients. This type of project typically begins with a user-oriented collaborative design approach and only moves into technology development in the second half of the year.

Almost as expected, our students view all of these problems through a user-centered design lens, and instinctively start by trying to find users to give them more insight into the solution space. This can work well, but can also prove to be a markedly deficient approach in two common ways: 1) the project has little real user-centered design exploration need, and 2) the sponsoring organization has differing (and sometimes conflicting) opinions about user-centered design as a problem-solving technique. Our preliminary findings regarding these two issues are summarized below.

PRELIMINARY FINDINGS

We need to help student teams communicate with the sponsor to determine whether the project could benefit from a user-centered approach. There is often a significant tension within sponsoring organizations around the validity of user-centered design in solving engineering problems. Student teams sometimes forge ahead without demonstrating the potential benefit of a user-centered design process. Negotiating with the sponsors can be difficult for students to navigate, particularly if this is their first industry project. Coaching students to convince the sponsor to allow them to tackle problems with the skills they know best is an approach many SCOPE faculty have had to adopt.

We need to clarify the strengths and skills of Olin College students regarding design approaches when recruiting potential sponsors. In recent years, we have been more careful, in our earliest negotiations with potential sponsors, to clarify our definition of user-centered design and contrast that with pure engineering system design. This allows us to craft the project description more carefully so our students can make more informed decisions when voting for a particular project. A major component of any industry-sponsored capstone experience is soliciting partnerships and scoping the work for student teams. Striking a balance between satisfying the engineering needs of potential sponsors and providing opportunities for our students to practice their design training with experts, is at best, tricky.

We should encourage students to reflect on prior design experiences and learn that there is not just one way to do

design. Students often fall back on the exact pedagogical processes defined in their design courses as the only way to "do design". If the user-centered design class was taught in one 14-week semester, students think that they should spend a semester on user-centered design for their capstone project, even if it is not warranted.

The senior capstone is the perfect proving ground for a user-centered design approach. Our students are trained to be able to work across the traditional boundaries of design, engineering, and entrepreneurship. The capstone experience, inclusive of all the problems we discussed above, provides a remarkable pre-professional experience in communicating across different disciplines. Students gain experience identifying and understanding the needs and values of their stakeholders, and learn how to respond to those needs appropriately.

An added benefit of working with traditional technology companies is that for many, user-centered design is not integral to their workflow. Thus, when the students execute well, SCOPE can be the ultimate external validation of their user-centered approach while delivering significant value to the sponsor in the form of new problem solving techniques.

REFERENCES

- [1] ABET, Criteria for Accrediting Engineering Programs, Engineering Accreditation Commission, 2010-2011, <http://www.abet.org/Linked%20Documents-UPDATE/Criteria%20and%20PP/E001%2009-10%20FAC%20Criteria%2012-01-08.pdf>, p. 3, Accessed March 2010.
- [2] Olin College, Course Catalog 2009 – 2010, http://www.olin.edu/academics/pdf/CourseCat2009-10_8-18_for%20website.pdf, p. 11, Accessed March 2010.

AUTHOR INFORMATION

Mark L. Chang, Assistant Professor of Electrical and Computer Engineering, Franklin W. Olin College of Engineering, mark.chang@olin.edu

Jessica Townsend, Assistant Professor of Mechanical Engineering, Franklin W. Olin College of Engineering, jessica.townsend@olin.edu

A Long-Duration Study of User-Trained 802.11 Localization

Andrew Barry, Benjamin Fisher, and Mark L. Chang

F. W. Olin College of Engineering, Needham, MA 02492
{andy, benjamin.fisher}@students.olin.edu,
mark.chang@olin.edu

Abstract. We present an indoor wireless localization system that is capable of room-level localization based solely on 802.11 network signal strengths and user-supplied training data. Our system naturally gathers dense data in places that users frequent while ignoring unvisited areas. By utilizing users, we create a comprehensive localization system that requires little off-line operation and no access to private locations to train. We have operated the system for over a year with more than 200 users working on a variety of laptops. To encourage use, we have implemented a live map that shows user locations in real-time, allowing for quick and easy friend-finding and lost-laptop recovery abilities. Through the system's life we have collected over 8,700 training points and performed over 1,000,000 localizations. We find that the system can localize to within 10 meters in 94% of cases.

1 Introduction

Computerized localization, the automatic determination of position, will augment existing applications and provide opportunities for new growth. One can easily imagine a phone, computer, or other device changing behavior based on location. A phone might disable its ringer when in a conference or classroom. Calendar reminders would only appear if a user was not already in the event's location. A laptop could automatically select the closest printer when printing. Finding a colleague would be as simple as looking up a phone number.

Localization abilities have spawned a number of companies including GPS navigation [5][16], asset tracking [4][17][20], and E911 systems [6]. The most common form, GPS, performs well in many instances, but it cannot achieve good accuracy indoors. To provide indoor localization, researchers have examined the use of dedicated hardware including ultrasound, IR, and RF beacons. Most of these platforms provide good resolution, but often have high installation, maintenance, and usage costs. With the advent of 802.11 wireless networking, researchers have turned to utilizing wireless access points as fixed RF beacons. This method mitigates the high hardware and installation costs of earlier systems, but often requires a substantial amount of off-line training, or the collection signal strength samples in many locations. Here we describe a large scale deployment of a system that uses 802.11 access points to localize, but transfers the training burden to the system's users, providing a cheap, fast, accurate, and low maintenance method for automated indoor localization.

We use the following terminology to describe our system: a wireless *fingerprint* denotes the signal strengths of surrounding access points at a given location. A *bind* is the act of associating a fingerprint with a location. An *update* is a location scan and localization calculation. Fingerprints are collected automatically, binds are performed by users, and updates are performed automatically or by user request.

2 Related Work

Location-aware computing is not new. Perhaps the best-known location-discovery platform is GPS, which uses U.S. government satellites to compute latitude and longitude [8]. While the high costs associated with GPS systems have disappeared, a receiver can only obtain a location with a clear sky-view. Moreover, GPS experiences substantial drift and is often not accurate enough to obtain room-level localization. Another set of commonly available localization systems serve the FCC's E911 initiatives [6]. These systems focus on approximately 100 meter accuracy and thus, like GPS, are of limited utility in indoor, room-level environments.

The first indoor location-aware systems, such as Active Badge and MIT's Cricket, succeed with specialized hardware [13][18]. Active Badge uses wearable transmitters and a network of sensors to gather location information and report it back to a server. The Cricket system uses a combination of RF and ultrasound to provide accurate and private location data. These systems avoid training, but instead require a substantial hardware installation phase. Both Active Badge and Cricket require location-bound hardware that necessitates prior access by trained personnel to each desired localization area. This installation and the associated time and hardware costs limit these systems' wide-scale use.

As 802.11 networks became common, researchers began utilizing existing hardware to compute location. Microsoft's RADAR and later Haeberlen *et al.* show success in using the signal strength of 802.11 nodes to determine fine-grained indoor location [1][7]. These and similar systems [11][14] require specialized training to create a database of location–signal strength tuples. Training demands a substantial upfront effort and physical access to all of the desired areas. Moreover, after some time, the training data needs to be refreshed to account for changes in the environment and access point locations.

To reduce the expense of training, Intel Research demonstrates an algorithm that can estimate location with only minimal data by expanding its known area with continued use [10]. While the self-mapping algorithm costs little to implement, it requires a significant period of time to gain acceptable localization accuracy and coverage. Moreover, multiple radio configurations complicate the implementation of a shared-training system. Wardriving can be used to seed the algorithm, but those data are often not dense enough for accurate indoor localization.

Both Bolliger and Teller *et al.* introduce crowdsourcing methods that allow users to train and correct the system [2][15]. Teller's work, conducted in parallel with our own, is similar to the system described here although on a smaller scale in time, space, and number of users. They studied 16 trained users limited to a single building with a specialized platform for only 20 days. Here we present a year-long deployment of a similar crowdsourcing method with over 200 untrained users spanning five buildings

operating on personal laptops. Bolliger’s system engineering is also similar to our own, but, again, he does not present results from a significant deployment.

In the commercial space, Ubisense has deployed accurate localization based on UWB signals in industrial environments [17]. Ekahau has been working on 802.11 localization for a number of years [4]. Skyhook Wireless has combined crowdsourced data with a substantial set of training data to improve their worldwide 802.11 localization system [14]. Navizon uses exclusively user-produced data, but like Skyhook, their system focuses on outdoor localization [12].

3 Architecture

3.1 Overview

Like [2] and [15], we use a client-server architecture to enable fast, accurate localization and provide a mechanism for feedback. To localize, clients perform an *update* in which they collect wireless 802.11 signal strength information (a *fingerprint*) to send to a server. The server computes the client’s location and sends the estimate back to the client for optional user review. The server also updates the friend-finding interface with the client’s new location in case another user wants to locate the first. When the client receives the location estimate, it offers the user an opportunity to confirm or correct it (Figure 1). If the user chooses to take this opportunity (*binding* a fingerprint), the client sends the new ground-truth data back to the server, which stores the record for use in all future localization computations.

Our system architecture is similar to MIT’s Organic Indoor Location system (OIL) [15] with a server-based localizer and without client-side caching. For brevity, we will examine only the novel aspects of our system in depth and direct the reader to the OIL implementation for other details. To compute locations, we use a Euclidean distance algorithm, comparable to RADAR’s Nearest Neighbor in Signal Space (NNSS) [1]. We have implemented the algorithm in SQL, interfacing PHP and wxPython clients that run on Windows, Linux, and Mac operating systems. We are planning to implement clients for smartphones and PDAs in the near future.

3.2 Deployment Site

A number of our design decisions were driven by the context in which our system was deployed. We developed and continue to run our localizer at Olin College, a small residential engineering school near Boston. Olin houses its entire 300-student population on campus with five buildings in total, encompassing more than 300,000 square feet. Our primary user base is students, with faculty and staff comprising less than 3% of users. Each student owns an institution-issued laptop although some use their own systems. Since each entering class has a slightly newer laptop model, we find a variety of similar but distinct radio/antenna combinations on campus.

In asset tracking, medical, or warehouse situations, one might choose to localize every few seconds, but students remain in the same place, be it a classroom, library, or residence hall, for extended periods of time. Thus, we chose to localize once every five

minutes, an unscientifically chosen but reasonable interval, in order to preserve system resources. Finally, we note that with the system providing a useful service, users have an incentive to provide accurate data and very little impetus to falsify locations.

3.3 Client Interface

The user interface is designed to be as non-invasive as possible. Since the system runs on personal laptops, it must have a small resource footprint minimizing CPU, memory, and power consumption. The client autostarts minimized in the system task tray and, to encourage use, never prompts the user without request, even when the location estimate is known to be poor. We found that we could collect enough training data without interrupting users and annoyed far fewer people in the process. If users want to train the system or access others' locations, double-clicking the task-tray icon brings up our deliberately simplistic interface (Figure 1). When an update is performed, the client displays its location estimation in question form, prompting a training response. The user can then accept the given location, choose from a list of likely locations, or create a new point. Figures 1–3 show typical GUI screens.



Fig. 1. Typical interface. The client has localized and asks the user to confirm its estimate. The upper left displays a user-name and the upper right contains a humorous checkbox.

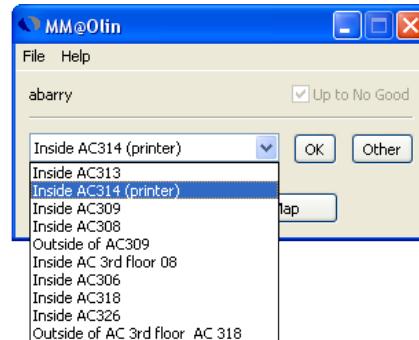


Fig. 2. Training interface. The user has clicked “No” in Figure 1 and is now prompted with nearby locations. Clicking “Other” allows the user to create a new location point.

Should the user determine that the localization is not correct, he or she can create a new location point. The client prompts for the building, floor, and a text name, suggesting a room number or descriptive phrase for the text entry (Figure 3). Once a user has entered those data, a labeled map appears allowing the user to select where the new location will appear visually. We found that making this process intuitive is key to ensuring the success of a crowdsourced data collection application.

New point creation is challenging for both the user and GUI designer. Most users are unable to correctly identify their location on an unlabeled blueprint, so the user inter-

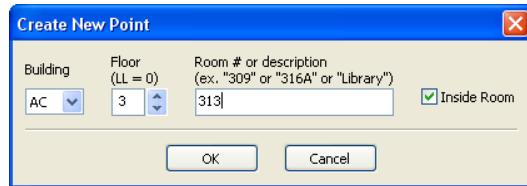


Fig. 3. New location creation interface. The user has clicked “Other” in Figure 2 and is now prompted to enter the details of his or her location. After clicking “OK,” the user will be prompted to select the location on a map of the local area.

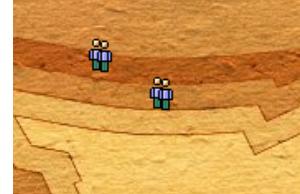


Fig. 4. Floor interface. Darker edges indicate a lower floor. In this case, two people are located on the ground floor and two more are located on the first floor.

face must be copiously labeled to prevent errors. We chose to allow users to customize location names, making their descriptions much more useful to others. For example, instead of calling a room “335,” users labeled it “3rd floor lounge.”

Allowing for free-form input, however, provided substantial possibilities for error. Although “lounge” is a natural input, it is not useful without context. To obtain both flexibility and accuracy, we constrain building and floor choices while allowing for a textual description. Thus, points have reliable context information and custom labels.

Finally, we note that custom naming lowers the entry barrier for new localization systems. Where we need only rough building diagrams, other systems require fully digitized blueprints or CAD models to generate an initial map.

3.4 Friend-Finding Service

To motivate users to train the system, we implemented a friend-finding service that publishes user locations on a map-like interface. The goals of the service are threefold: allow users to quickly and easily locate specific people, display all users’ locations on one screen, and act as an advertisement for the project.

To satisfy our goals, the interface must allow for both searching and browsing while maintaining an attractive look and feel (Figure 5). To encourage use and help explain the concept, we themed the project as the “Marauder’s Map at Olin,” a reference to a magic map that displays people’s locations in the popular Harry Potter series. While this theming might seem trivial, we found that it was critical to building and maintaining a user base. The map theme helped people understand why the service is useful and encouraged them to share with their friends.

Displaying hundreds of users on multiple floors proved to be a challenge. Early tests showed that users clustered near the edges of buildings, so we expanded our building representations to show the edges repeatedly, utilizing the new space to indicate vertical displacement and avoid icon overlap (Figure 4). This is not a perfect solution, as some buildings have popular locations in their centers, but we found these overlaps to be relatively minor.

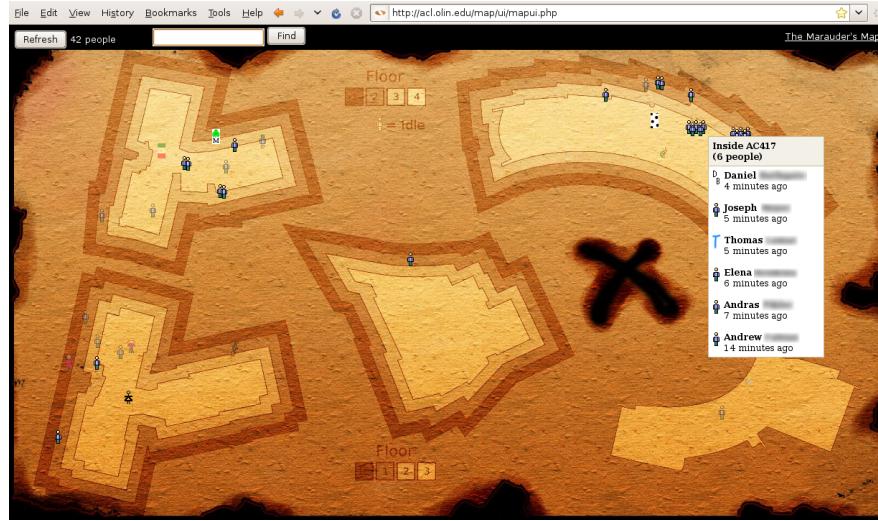


Fig. 5. Friend-finding webpage frontend. The interface is themed like an old magical map. The user has moved the mouse over a cluster of people, prompting a drop-down list of location, names, and update times.

3.5 Privacy

Privacy is a concern in any localization system. Given that the primary application of our implementation is a friend-finding service, we found that concerned users were aware of the implications and simply chose not to participate. Some users requested the ability to remove their location report at any time, a feature we implemented. All of the system's services reside only on internal servers to ensure that location information is not published outside of our institution.

4 Approach to Crowdsourcing

4.1 Motivation

The primary motivation for crowdsourced data is the reduction in time required to train the localizer. Moreover, with crowdsourcing, users provide most of the data while the system is already localizing, reducing the time before the localizer can be used. When training, researchers found that gaining access to private and semi-private spaces (offices, residence hall rooms, etc.) was difficult and awkward, a problem that user-trained systems avoid [7].

A pre-trained system requires retraining to account for changes in the environment, but a user-trained system is continuously updated with no overhead. In addition, crowd-sourced training naturally produces data that are dense in places that are commonly

visited. Users tend to bind in places they frequent, causing common locations to have dense data. Because our localizer treats each bind separately, it weights common locations more heavily, resulting in a natural location-frequency dependency.

Finally, traditionally trained systems suffer from a conflict between coverage, the number of distinct locations with data, and accuracy, the measure of how often the localizer chooses the correct location. If the system is aware of many often unoccupied locations, it will suffer from a decrease in accuracy. Crowdsourcing helps mitigate the problem by naturally ignoring rooms that users do not frequent. For example, there are small, narrow trash rooms in each wing of the residence halls that were never bound in our system (Figure 6). A traditionally trained system might incorrectly place a user in one of these rooms, even though the probability of a user being located there is very small. Our system will always avoid these areas because no users have bothered to train them.

4.2 Initial Training

While almost all of our data are provided by users, we found that a minimally trained system was important to convince users that the software was compelling. Typically, this type of initial training takes about 1-3 minutes per location [7]. At this rate, a system covering over 350 spaces would take approximately 16 person-hours to train manually, but we can train the system to a minimally usable state, in about 1.5 hours. With that training, we create a sparse map including hallways and common areas, allowing for reasonable (within 10-20 meters) estimates that support further training by users. This training is easy to perform because all fingerprints can be collected in public areas and with relative infrequency. To train our system we simply walked down most hallways and bound one or two points per hall.

5 Results

After a short beta test, we deployed our system campus-wide at Olin College in April 2008 and have continued operations for over a year. We announced the project with an email to a common list in which we explained the concept and encouraged users to download and run the client. To date, we have had more than 200 unique users, 8,700 binds (95% of users contributing), and over 1,000,000 location updates.

5.1 System Accuracy

A localization system's performance is determined by both *coverage* and *accuracy*. Coverage measures how much of the deployment area has associated fingerprints while accuracy measures of how often the localizer reports the correct location. We first discuss coverage and then proceed to examine our localizer's accuracy.

At the beginning of deployment, only our initial survey supplied data, so we started with poor coverage, especially in private rooms (Figure 6(a)). We found that within 30 days of launch, our coverage stabilized at a reasonably complete level, with over 75% of all known locations at the year's end having already been bound. Coverage progression

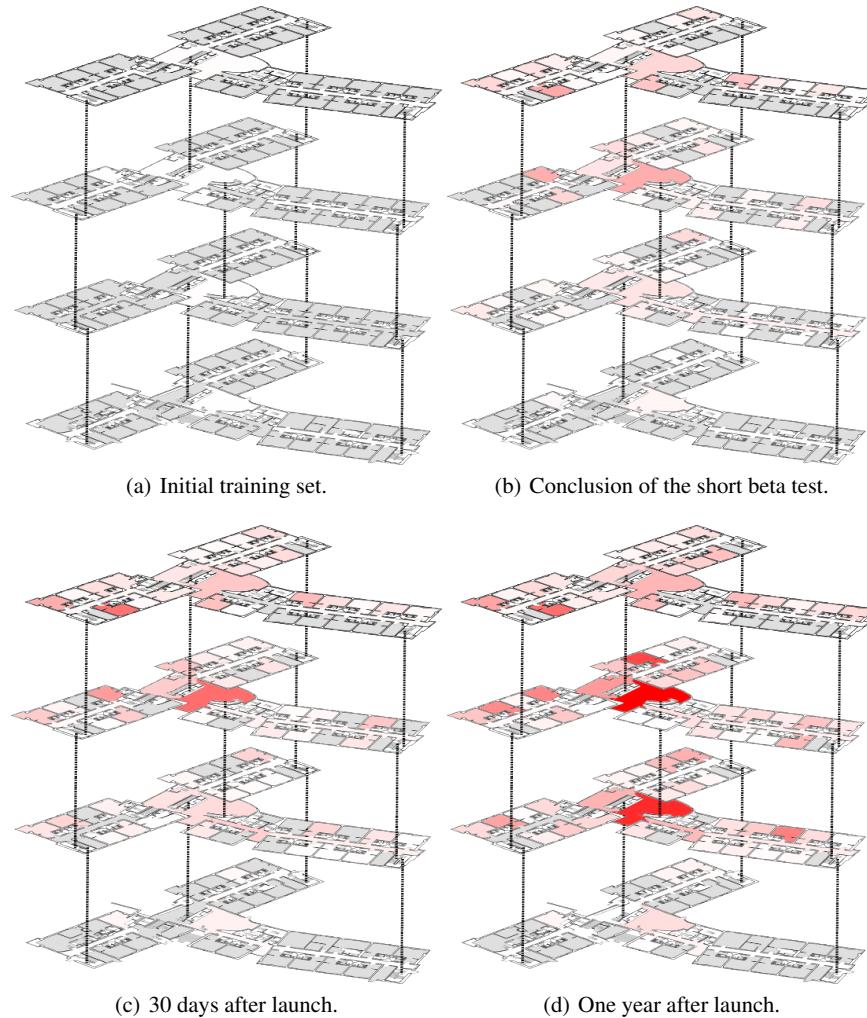


Fig. 6. Map of training density in one of the five deployment buildings. Gray indicates no training data. Light to dark red indicates progressively more fingerprints for that space. Note that common areas (center of the building) have far more data than individual's rooms.

for one building can be seen in Figure 6. Other buildings show similar patterns, although places students are unlikely to spend time, such as faculty offices, never achieve good coverage.

Our second metric is system accuracy. Accuracy starts poor and improves with the number of binds. To measure accuracy, we note that we should not simply test the system at random locations. Real accuracy is determined by how often the localizer correctly estimates *users'* locations. To use the aforementioned example, the localizer's performance in a small trash room is of little importance to true accuracy.

To test our system in this manner, we chose to survey our users during deployment. After the system had reached a steady state, we emailed our users asking them to report the localizer's accuracy at that moment. Thus, users opened the client, checked its current estimate against their real location, and reported performance in meter ranges (ie within 0-5 meters, 5-10 meters, etc.). We received 57 reports representing more than half of all online users within 8 hours. While these reports were self-selected based on which users chose to respond, we do not believe this bias has skewed our data measurably.

Figure 7 shows localization errors. We find that we localize to within 5 meters in 69.9% of attempts and to within 10 meters in 94.9% of cases. This accuracy is approximately equal to other published systems, although it does not achieve the performance of Haeberlen *et al.*'s calibrated or King *et al.*'s dense data techniques [7][9][15].

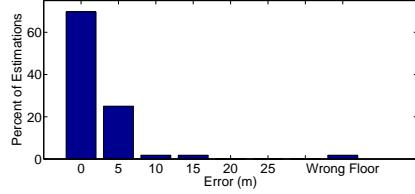


Fig. 7. Localization error. We determined error by asking users to perform spot checks in their current location. We find that we localize correctly in 69.9% of attempts and are within 10 meters in 94.9% of cases. We localize to the wrong floor only 1.8% of the time.

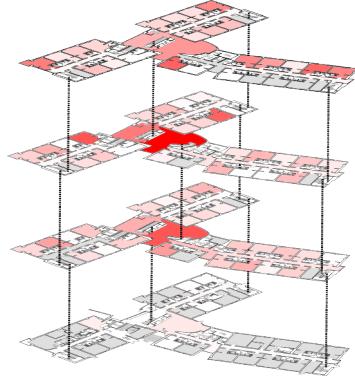


Fig. 8. Combined density of location reports for one year in one residence hall. Clearly, common areas are visited far more often than individuals' rooms. No students live on the first floor so we are not surprised to see few reports in that location.

5.2 System Vulnerabilities

We note a number of conditions that degrade our performance. First, during our deployment, a large fraction of the campus's access points received firmware modifications that resulted in a change in MAC address. The system does not recognize the new configurations and assumes that none of the old access points exist. While our architecture is designed to easily adapt to new access points with a single dictionary replacement, we found that losing these access points did not significantly degrade performance, so we allowed the system to operate without intervention.

In addition to changing MAC addresses, network administrators moved a small fraction of access points to improve wireless performance. While users have retrained the system, this movement continues to degrade our performance. To mitigate this issue, we are considering a weighting system that favors new fingerprints, marginalizing old and possibly outdated data. The design of the weighting system requires further study to determine if it should universally downgrade old data or only ignore old fingerprints when newer ones are available for a location. The first implementation would cause the localizer to favor newly bound points while ignoring old fingerprints, effectively reducing coverage over time. The second implementation retains coverage, but might not reflect newer user-movement patterns.

A third potential degradation of performance is the automatic and manual gain control on access points. During our deployment both automatic and manual power adjustments were made, but the system showed no noticeable decrease in performance. Without a clear indication that this was causing degradation of localizations, we have not spent the engineering resources to study this effect further. It is worth noting, however, that these power changes affect only the local area around the modified access points and simply shift the localizer's tendency closer or further from the respective wireless node.

A fourth potential degradation of performance is our support of a wide range of laptops, including Mac hardware and at least four models of the Dell Latitude D-series. We do not currently account for radio/antenna configurations, although the system has a natural correction based on user binding patterns, as described below.

In the user space, we note that user-generated data are not as reliable as professionally generated scans. It is difficult to determine how often localization errors are due to user error when training the system or from signal strength variations resulting from dynamic objects, antenna orientation, radio chipset, access point power fluctuations, or a host of other phenomenon. General accuracy statistics provide some insight for an upper bound on errors, but we have not yet developed a metric to study these errors explicitly. A time-based weighting on data, as examined above, would provide some mitigation for mistaken inputs, allowing them to be corrected as new, better data become available.

Finally, our system does not address the possibility of malicious users, beyond marking each bind with an identifier that allows the elimination of all binds by a particular user. While this is a concern, we are not aware of a single instance of such activity. Moreover, to be effective, malicious users would need to create a substantial number of false data points in many locations to overwhelm the existing fingerprint set. Many binds in one location would overwhelm that particular location, but the damage would

be confined to the local area. To be truly successful, a malicious user would need to bind incorrect data throughout the system's coverage, a far more difficult task.

In the event that malicious activity becomes an issue, we have discussed the implementation of weights based on how similar new data are to existing fingerprints. In this way, outliers are rendered harmless automatically. If a significant number of outliers were added to the system by different users, the weights would begin to skew towards those new fingerprints, accounting for dramatic environment changes such as access point relocation.

5.3 User Behavior

After release, we collected about 71 binds per day and within 2 months our data set had grown to 27 times the initial training, a collective effort of approximately 25 person-hours in ideal conditions. While training rates decreased as accuracy increased, users still train the system over one year later. Figure 9 shows these trends in database size over time.

As expected, we collected more fingerprints in common and public locations than in private rooms. The median number of binds per room is 7 and the maximum is 305, which occurs in a residence hall lounge (a common socializing space with couches and a TV.) 17% of known locations have only one bind and 53% have 10 or fewer. As the system's coverage grew, the number of new locations bound quickly decreased. Our initial survey bound 16% of total locations, beta testers bound 48%, and our general user base bound the remaining 36%. Thus, we find that users are far more likely to pick an existing location than they are to bind a new one.

In addition, we find that the user contribution profile is similar to other mass-interaction applications [15][19]. A few enthusiastic users bind an inordinate number of times. Interestingly, these users do not update their location as often as we might expect, with very little correlation between bind and update frequencies.

We hypothesized that our data collection method would result in dense data in frequented places, and our expectations are confirmed. We see this in the similarity between Figures 8 and 6(d) which show where locations are reported and bound, respectively. These data confirm our second hypothesis, that users bind in places that they, individually, frequent. 51.2% of all location updates occur in places that users had bound themselves. In other words, half the time a user is in a place where he or she has contributed data. Thus, we confirm that one of the primary advantages of crowdsourced data collection is that users are willing to train where they frequent and that they tend to reside places where their own data are best.

5.4 Application Use

While we have not yet performed a formal study, we feel that users are satisfied with the localization application. In one year we logged over 14,000 friend-finding page loads, averaging well above 50 hits per day during both fall and spring terms. We note that not only did users utilize the system at launch, they continue to use the service throughout its deployment. We also note that with 100 active users, we are localizing about 1/3 of the student population's systems.

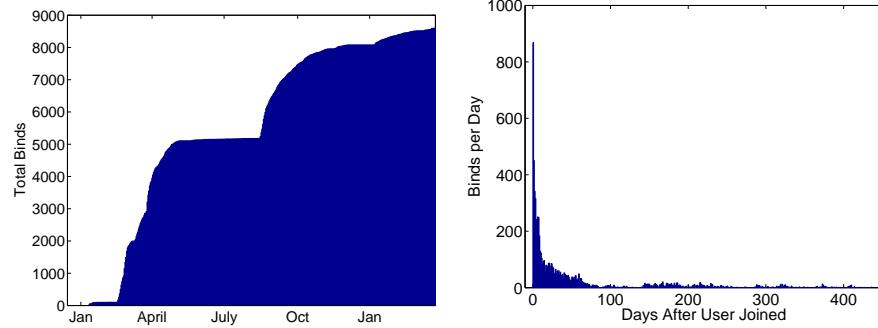


Fig. 9. Fingerprint database size over time. Training rates were steady after release and have reduced as the system became more accurate. We are not surprised to see little data added during the summer term when students are not on campus.

Fig. 10. All binds sorted by days since the binding user first localized. Clearly, users bind a significant amount in their first day and steadily less throughout their usage. We find that 11.5% of all binds occur on a user’s first day.

Our last method of evaluating user satisfaction is purely anecdotal. Users tell us that they enjoy using the system and have only rarely contacted us with complaints, despite our contact information being readily available. Perhaps our favorite example of users’ creativity is using the system in a scavenger hunt. An on-campus business group was running a promotion in which they planted clues advertising the whereabouts of free product samples. To create one of the clues, the group, unbeknown to us, managed to emulate a client and change the reported name and user icon to their name and logo, respectively. They then used the reported location to advertise where free samples could be found. Months later, when performing data analysis, a developer found the odd entry and finally traced it to the group who reported that it was the most popular clue in their entire game.

6 Detailed Analysis

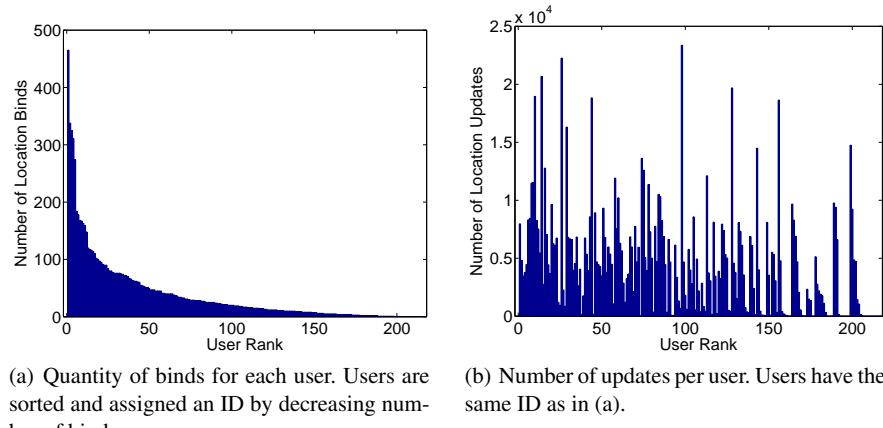
We now discuss the system’s usage in detail. We examine trends in both when and which users train the system and the profile of where users localize. We find that new users are the primary providers of ground-truth data and that, despite our high participation rate, a small set of users provide most of the system’s data. These often-training users, however, are neither more or less likely to localize than their non-binding peers.

6.1 Training Rates

Users tend to bind data in their first few days of use and rapidly stop providing ground-truth information thereafter. Throughout the system’s life, 43% of all binds occur within 10 days of the binding user’s first application use. We offer two explanations for this

phenomenon. First, users may train the system because of the novelty of providing data. Once that novelty fades, users become less interested and only localize. Second, student movement patterns do not change substantially from one day to the next. Once a user has trained his or her habitual places, the system may not require further training for accurate use and thus exclusively localizing is acceptable. Figure 10 shows bind occurrences sorted by days since adoption.

While new users train the system more often than longtime users, we find that some provide a disproportionate amount of data. Figure 11(a) shows a sorted profile of users based on number of binds. In our system, 20% of users provide 66% of the data. As mentioned above, we notice that there is little correlation between users who provide data and those who localize often. This is manifested in the difference between 11(a) and 11(b).



(a) Quantity of binds for each user. Users are sorted and assigned an ID by decreasing number of binds.

(b) Number of updates per user. Users have the same ID as in (a).

Fig. 11. Number of binds and updates for each user. Users are sorted by decreasing number of binds. In (a) we see that 20% of users bind 66% of the ground-truth data. In (b) we find that those often-binding users are not significantly more likely to update their location often, showing that the primary producers of data are not necessarily the primary consumers.

6.2 Application Usage Patterns

We find cyclic patterns in usage, both throughout each day and throughout the year. As expected, we find that users cluster in academic buildings during the day while returning to residence halls at night. Figure 12(a) shows localizations on an hourly basis in an academic building while 12(b) displays a similar plot for a residence hall.

We find that long-term training patterns are driven by new users. It appears that while localizing holds longevity as a useful service, the novelty of binding data fades,

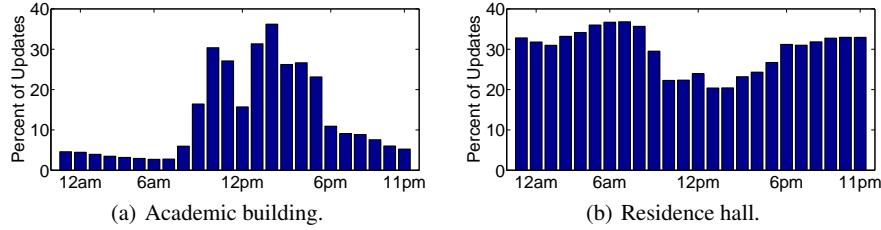


Fig. 12. Percentage of localizations per hour in both an academic building and a residence hall. As expected, we find that users occupy academic buildings during daylight hours (excluding lunch) and residence halls at night. We also note that while users arrive to classes in large groups, they tend to leave in a more gradual manner.

causing users to stop binding. This trend might also be influenced by an increase in accuracy in the places that users individually frequent, requiring less training as the system learns from the user. In addition to Figure 10, we see this pattern in the similarity between Figures 13 and 14, displaying when users join and when users bind, respectively.

7 Future Work

This paper presents an implementation of a user-trained localization service covering an entire college campus. To utilize this framework in other scenarios, we consider a number of additions including new applications and novel ways to collect training fingerprints.

To augment our user base, we are considering implementing additional localization-based services. For example, we might create a tagging system for files that records location information on creation and modification. With this information, a user could search for all files created in a particular room. Users tend to create different types of files in different places, such as minutes in a meeting room and source code in a lab, so searching based on location might be helpful. Other potential services include location-based printer selection and an API to support new developers and new deployments.

Porting our code to hand-held and smart-phone devices would provide interesting new data sources and challenges. Compared to laptops, these platforms are more ubiquitous and would provide more continuous and varied data with their constantly active radios. This more diverse set of hardware might require a platform identification mechanism and conversion functions between device fingerprints, like Haeberlen's implementation across different radio platforms [7].

To gather more binds, we are considering automatic calendar integration. Many calendar appointments are tagged with a location which we could extract and utilize to automatically train the system. When a user's calendar indicates that he or she is in a specific place, the system could automatically collect fingerprints and bind them to that

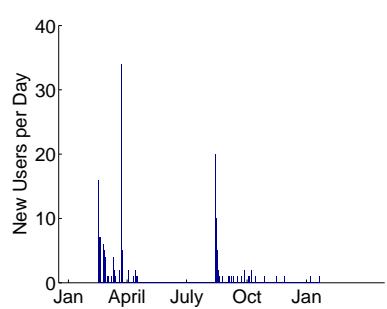


Fig. 13. Number of new users per day. Our campus-wide release occurred in April and a new academic year started in September.

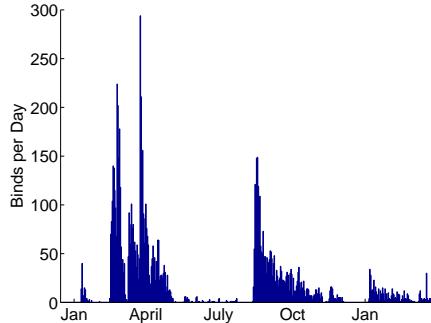


Fig. 14. Binds per day. Given that new users bind data often, we are not surprised to see a significant correlation to Figure 13. One interesting case is the system's second January, where we see no new users but a significant number of binds. In this period, students return to campus after intersession and appear again interested in binding data.

location. Obviously, users and/or their wireless devices are not always located where their calendar indicates, but with intelligent use of idle-times, a limited fingerprint set, and perhaps a movement detector like [3], this type of training could be made reliable. In more extreme cases, we might consider using calendar integration to train an entire system without any user interaction, although the accuracy of these fingerprints would require careful study.

Finally, we are continuing analysis of our data and are planning more formal user surveys to better characterize the system's strengths and weaknesses. Moreover, we are planning more expansive accuracy experiments that will inform the distinction between random-location accuracy and the common-area accuracy that a normal user experiences.

8 Conclusion

We have described a long-running test of a user-trained system that performs accurate indoor wireless localization in areas with existing 802.11 networks. The system can be deployed in any location that has a pervasive network and a group of users willing to train it. By utilizing personal laptops and existing access points, we do not need to build or buy any additional hardware. The system's interfaces are simple and intuitive, allowing users to localize, find others, and contribute training data with no instruction.

After more than one year, our system continues to operate and has accumulated more than 200 users, 8,700 binds, and over 1,000,000 location updates. Usage patterns

provide natural guidance for the localizer, improving accuracy by accumulating dense data in common areas. These methods result in successful localization to within ten meters in over 94% of cases, providing convincing evidence that crowdsourcing is a practical method for cheap, pervasive wireless localization.

9 Acknowledgements

We would like to thank the anonymous reviewers for their valuable comments and guidance. We also thank the community at Olin College for their ideas, testing, and feedback. Andrew Barry and Benjamin Fisher are supported by F. W. Olin Scholarships.

References

1. P. Bahl and V. Padmanabhan. RADAR: An in-building RF-based user location and tracking system. In *Proc. IEEE Infocom 2000*, pages 775–784. Proc. IEEE Infocom 2000, IEEE CS Press, 2000.
2. P. Bolliger. Redpin - adaptive, zero-configuration indoor localization through user collaboration. In *MELT '08: Proceedings of the first ACM international workshop on Mobile entity localization and tracking in GPS-less environments*, pages 55–60, New York, NY, USA, 2008. ACM.
3. P. Bolliger, K. Partridge, M. Chu, and M. Langheinrich. Improving location fingerprinting through motion detection and asynchronous interval labeling. In T. Choudhury, A. J. Quigley, T. Strang, and K. Suginuma, editors, *LoCA*, volume 5561 of *Lecture Notes in Computer Science*, pages 37–51. Springer, 2009.
4. Ekahau, Inc. <http://www.ekahau.com>.
5. Garmin Ltd. <http://www.garmin.com>.
6. D. Geer. The E911 dilemma. *Wireless Business and Technology*, Nov. 2001.
7. Haeberlen, Flannery, et al. Practical robust localization over large-scale 802.11 wireless networks. In *Proceedings of the Tenth ACM International Conference on Mobile Computing and Networking (MOBICOM)*, Sept. 2002.
8. J. Hightower and G. Borriello. Location systems for ubiquitous computing. *Computer*, 34(8):57–66, Aug. 2001.
9. T. King, S. Kopf, T. Haenselmann, C. Lubberger, and W. Effelsberg. Compass: A probabilistic indoor positioning system based on 802.11 and digital compasses. In *Proceedings of the First ACM International Workshop on Wireless Network Testbeds, Experimental evaluation and Characterization (WiNTECH)*, Los Angeles, CA, USA, September 2006.
10. A. LaMarca et al. Self-mapping in 802.11 location systems. In *UbiComp 2005: Ubiquitous Computing*, pages 87–104. LNCS 3660, Springer, 2005.
11. Letchner, Fox, and LaMarca. Large-scale localization from wireless signal strength. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, 2005.
12. Navizon: Peer-to-peer wireless positioning. <http://www.navizon.com>.
13. N. Priyantha, A. Chakraborty, and H. Balakrishnan. The cricket location-support system. In *6th Ann. Intl Conf. Mobile Computing and Networking (Mobi-com 00)*, pages 32–43. ACM Press, 2000.
14. Skyhook Wireless. <http://www.skyhookwireless.com>.
15. S. Teller et al. Organic indoor location discovery. Technical Report MIT-CSAIL-TR-2008-075, MIT, Dec. 2008.

16. TomTom NV. <http://www.tomtom.com>.
17. Ubisense, Ltd. <http://www.ubisense.net>.
18. R. Want, A. Hopper, V. Falcao, and J. Gibbons. The active badge location system. *ACM Transactions on Information Systems*, 40(1):91–102, Jan. 1992.
19. S. Whittaker, L. G. Terveen, W. C. Hill, and L. Cherny. The dynamics of mass interaction. In *In Conference on Computer-Supported Cooperative Work (CSCW)*, Nov. 1998.
20. Wisetrack, brand of TVL, Inc. <http://www.wisetrack.com>.

A Parameterized Stereo Vision Core for FPGAs

Stephen Longfield, Jr. and Mark L. Chang

Franklin W. Olin College of Engineering

Needham, MA 02492

Email: mark.chang@olin.edu

Abstract—We present a parameterized stereo vision core suitable for a wide range of FPGA targets and stereo vision applications. By enabling easy tuning of algorithm parameters, our system allows for rapid exploration of the design space and simpler implementation of high-performance stereo vision systems. This implementation utilizes the Census Transform algorithm [1], [2] to calculate depth information from a pair of images delivered from a simulated stereo camera pair. This work advances [3] through implementation improvements, a stereo camera pair simulation framework, and a scalable stereo vision core.

I. INTRODUCTION

In the pursuit of machine vision systems that can perform near the fidelity of biological vision systems, one area of intense study is the acquisition of an accurate three-dimensional model of an unstructured environment. While many machine vision algorithms and systems have sufficed with a two-dimensional understanding of the world around them, advanced robotics, navigation, and the machine intelligence that drives them, would greatly benefit from enhanced spatial perception.

Native three-dimensional perception systems such as LiDAR remain cost- and form-factor-prohibitive for many applications. Fortunately, CMOS camera technology is both inexpensive and favorably packaged for embedded, mobile vision systems. Just as the human vision system creates three-dimensional vision through a pair of eyes that each perceive only two dimensions, stereo vision attempts to infer depth from the disparities between two camera images.

In this paper, we present a significant improvement in the design-time use of stereo vision on FPGAs through a fully parameterized stereo vision core. As a natural extension of our previous work [3], this system enables fast exploration of the design space allowing for better tuning of the stereo vision core to application needs and target hardware resources. This section introduces stereo vision algorithms and related work in hardware acceleration of stereo vision. Section II discusses the hardware and software implementation of our scalable stereo core and presents simulation and synthesis results. Section III details our ongoing and future work.

A. Previous Work

This work extends our previous work [3] through algorithm implementation improvements, a generalized stereo camera pair simulation framework, and by exposing all tunable parameters of the stereo core to the end user for easy modification. As this implementation serves as only the core of larger stereo

vision platform, features such as real hardware interfacing with cameras, USB output to a host computer, and host software for post-processing stereo images is not included. However, as each platform's needs are unique, we do not see this as a severe limitation and only note it as a difference from our previous work.

B. Stereo Vision

Reconstructing depth information from a pair of two-dimensional images is an area of active research both in basic algorithms and in their acceleration with hardware. The basic stereo sensing mechanism mimics biological systems with a pair of cameras, separated by some distance, viewing approximately the same “scene”. Because of the separation of the cameras, an object will appear to have a lateral shift inversely proportional to its distance from the cameras. You can experience this phenomenon by holding an object close to your face and viewing it with only the left eye, then with only the right eye. The object will appear to have a larger lateral shift the closer it is to your eyes.

By finding the lateral displacement (or disparity, as it is more commonly referred) of an object between camera images, it is simple geometry to calculate the range of an object from the camera. Objects that are relatively far from the camera will occupy nearly the same location in both images, while objects that are relatively close will have a large disparity.

Finding an object in both camera images is called *correspondence*. As described in [4], most correspondence algorithms fall into two basic approaches: area-based or feature-based. Area-based algorithms take windows of pixel intensities in one camera image and attempt to find correspondence in a window of the same size in the other camera image. Feature-based algorithms abstract pixel intensities into features such as edges, corners, lines, contours, or patches, and attempt to find the corresponding features in the other camera image. The potential benefits of feature-based algorithms is that these features will have less variation than raw pixel intensities and will therefore be more robustly correlated across image pairs.

For any of these approaches to be effective, both cameras must deliver images in the same exact global coordinate system. This entails removing or compensating for image distortion, such as rotation, scaling, and vertical translation. This is often accomplished through careful mechanical alignment of the camera pair and a calibration phase to compensate for lens and imager differences.

C. Census Transform

A thorough investigation of a variety of algorithms from [1], [5]–[7] led to the selection of the Census Transform [1], [2] for implementation. The Census Transform is in the class of area-based algorithms and is well-suited for acceleration in FPGAs, as the majority of the operations performed consist of bit manipulations and simple arithmetic. The Census Transform also exhibits relative insensitivity to absolute differences in imaging devices, and operates well on images with high dynamic range.

The Census Transform begins by considering a window around the pixel under test from one (the primary) image. An example of a window with radius “1” is the following, centered on a pixel with intensity 130:

$$\begin{array}{ccc} 127 & 129 & 131 \\ 126 & 130 & 129 \\ 126 & 131 & 133 \end{array}$$

This window of intensities is then transformed into a bit string by computing whether each of the neighbor pixels in the window is greater or less than the center value. For this example, pixels with lower intensities are given a value of 1, and pixels with higher intensities are given a value of 0. Pixels with intensity values equal to the intensity value of the center pixel can be treated as either a 1 or 0, but must remain consistent. The transformed matrix would therefore be:

$$\begin{array}{ccc} 1 & 1 & 0 \\ 1 & X & 1 \\ 1 & 0 & 0 \end{array}$$

This matrix is then rearranged into a bit string to capture the entire window, resulting in the eight-symbol pattern: “11011100”. With this pattern that represents the window about the pixel under test in the primary image, we must find the closest matching pixel window in the secondary image by calculating the corresponding bit strings for several possible matching center pixels in the secondary image and comparing bit strings. For our implementation, we used the Hamming distance between the bit strings as a comparison measurement. The Hamming distance is simply the number of places in the bit string where bits in the same position differ. Pixel windows with similar intensities relative to the center pixel will have similar bit strings, which in turn will yield smaller Hamming distances.

The same pixel position in the secondary image yields a disparity of zero and an infinite range estimate. The algorithm continues to shift and compare until a predetermined maximum disparity is reached, at which point the search is stopped. Now, among the set of Hamming distances, the smallest corresponds to the pixel under test’s most likely disparity value.

D. Parameterization

The need for parameterization can be most acutely demonstrated during prototyping of robotic platforms. In deciding and evolving platform characteristics such as the physical

camera separation (baseline), the camera type (possibly changing pixel depth and resolution), the physical environment (indoor vs. outdoor), the expected speed of the robot, and the available FPGA resources, basic algorithmic parameters require constant manipulation to achieve the desired results. Making these changes requires non-trivial modification to the hardware design of the stereo vision core.

By being able to quickly generate the hardware stereo core, it becomes possible to perform experiments *in situ*, in hardware, with little delay. Additionally, by storing a large number of configurations, platforms can now determine and utilize, in the field, the best parameters for operating conditions.

E. Related Work

As many contemporary stereo vision algorithms have heavy computational requirements, a number of acceleration efforts have been documented to date. Of interest are FPGA-based implementations [8]–[10], and dedicated commercial ASIC solutions (Tyzx Deep Sea products), none of which offer the parameterization of our implementation.

Algorithmically, the most closely related work is [8]. The Census Transform was used here on the PARTS Reconfigurable Computer, consisting of 16 Xilinx 4025 FPGAs and 16MB of SRAM. Our implementation is a generalization of this basic idea to be able to target both low-cost, small FPGAs, and large, high-performance FPGAs, while obviating the need for external memories.

In [9], [10], the authors implement phase-based correspondence algorithms that differ significantly from the Census Transform in both design and arithmetic complexity. These implementations, however, are single instances and do not allow for easy parameter tuning.

Stereo vision remains an active area of research whose breadth and depth is beyond the scope of this paper. Interested readers are encouraged to explore [7] for an a broad survey of the field of study and a quantitative and qualitative comparison of approaches.

II. IMPLEMENTATION

The implementation of the stereo vision core was done in Verilog, simulated with ModelSim, and synthesized using the Xilinx toolchain. Parameterization was accomplished using a combination of basic Verilog generate constructs and Verilog code generation with software written in Python. The parameters that are available to the end-user to modify include:

- pixel depth
- image width
- image height
- search window size
- maximum disparity

The *pixel depth* parameter adjusts the maximum dynamic range. Typically, this would be 8 bits per pixel. As the implementation also generates a simulation model for typical OmniVision-brand CMOS imagers (as previously used in [3]), the *image width* and *image height* parameters are used for the simulation of row and frame timing information. The *search*

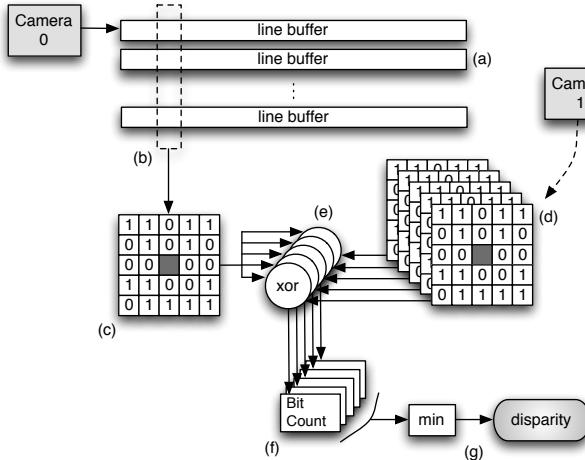


Fig. 1. Graphical representation of data flow in the Census Transform.

window size parameter specifies both the width and height of the square window of pixels surrounding the pixel under test. Finally, the *maximum disparity* parameter specifies the amount of lateral shift that will be searched to find correspondence in the secondary image.

Figure 1 depicts the flow of data from cameras to disparity calculation and closely mirrors the previous description. Both *Camera 0* and *Camera 1* are assumed to operate synchronously—that is, they deliver the same pixel coordinate at the same time, and have synchronized pixel clocks. This implementation is streaming in nature, and will deliver a disparity calculation at every pixel clock with some fixed latency. Starting from *Camera 0*, a pixel is generated with every pixel clock.

Pixels must be stored in *line buffers* (b) until there are enough pixels for a window to be formed around the pixel under consideration. A single line buffer width is the same as the image width, and we require *search window size* number of them. The Xilinx synthesizer is able to utilize BlockRAM to implement these line buffers efficiently.

Once enough pixels are available to begin filling a window around the pixel under consideration, a register bank of (*search window size*)² registers (c) is filled, a column at a time, from the line buffers. At every column fill, the binary relative intensity is calculated and stored in a register array the same size as (c) (not shown).

At the same time, *Camera 1* has been delivering pixels in an identical fashion to a different instance of the same structure of line buffers and pixel window registers (d) (not shown, and indicated by the dashed line). The only difference is that there are *maximum disparity* copies of the pixel window registers. As soon as there are enough pixel windows available from *Camera 1* to compare against the pixel window from *Camera 0*, the current pixel window from *Camera 0* is XOR’ed against all *maximum disparity* number of pixel windows from *Camera 1* (e).

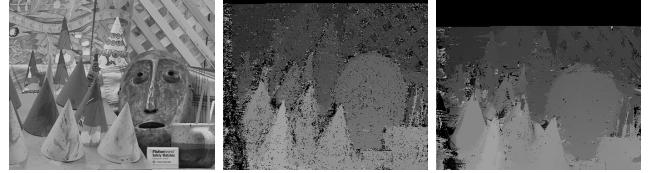


Fig. 2. “Cones” from [11]: (left) original image; (center) output, window size 10x10, maximum disparity 90; (right) output, window size 60x60, maximum disparity 90.

The number of differences resulting from these parallel XORs are calculated in (f), giving a scalar value that represents the level of mismatch between the pairs of images. The index of the minimum of these values is then taken (g) and delivered as the resultant disparity estimate at the position of the pixel under consideration from *Camera 0*.

A. Simulation and Verification

The design was simulated and verified using ModelSim SE 6.4c with a variety of parameter configurations using images from [11], [12]. Simulation output demonstrating the effects of parameter modification can be seen as raw output in Fig. 2. No pre- or post-processing that might typically accompany range estimation has been done except to scale disparity values to occupy the full gray scale.

Comparing the middle and right images of Fig. 2, one can easily see the smoothing impact of an increased window size. Less noise is present, while less detail is also found. The right image is vertically offset as the algorithm does not currently begin processing until there is a full window for evaluation. Therefore, going from 10 to 60 pixels in the window increases the “band” at the top of the image by 50 pixels.

B. Parameter Evaluation

The design was synthesized for a variety of parameters using the Xilinx ISE 9.2.04i. The target FPGA was a Xilinx Spartan-3 XC3S2000, chosen to match [3]. In Fig. 3, we can see some reasonable trends in logic utilization versus maximum disparity for a variety of pixel window sizes.

Fig. 3 utilizes an image size of 320x240, pixel depth of 8 bits, maximum disparities from 2 to 40 pixels, and window sizes from 7 to 19 pixels. As the maximum disparity increases, the number of registers required to store pixel windows increases. The number of XORs required to compute the mismatch between pixel windows increases with maximum disparity, and the size of each XOR scales with the size of the pixel window. Additionally, the number of bit counters required to calculate the scalar correspondence between pixel windows increases with maximum disparity, and the complexity scales with the size of the pixel window. Therefore, an increase in either maximum disparity or window size should demand more resources. For large window sizes and large maximum disparity, the number of data points is fewer than others as the resource requirements exceeded the capacity of the target FPGA. BlockRAM usage is not shown,

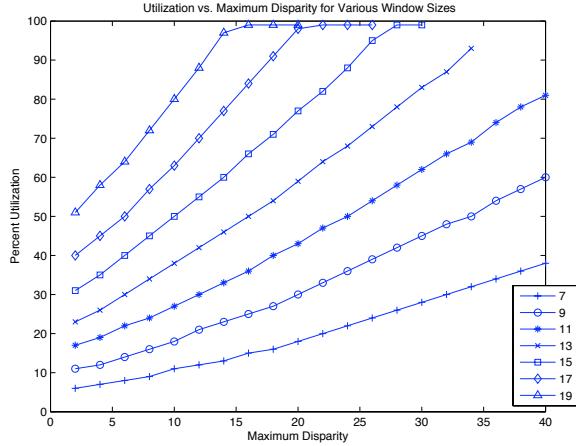


Fig. 3. FPGA logic utilization versus maximum disparity for various window sizes. Image: 320x240, pixel depth: 8 bits, disparities (2-40), window size (7-19).

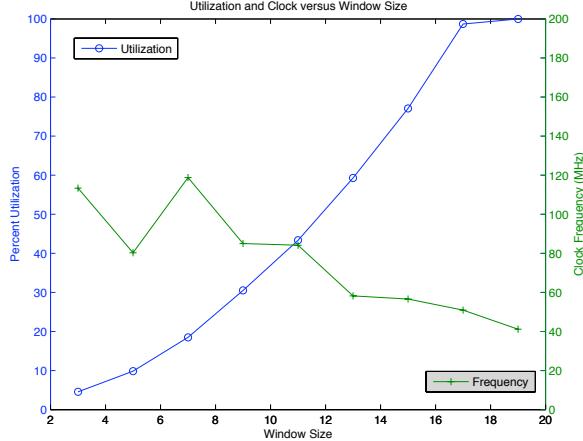


Fig. 4. FPGA utilization and clock frequency versus window size for maximum disparity of 20. Image: 320x240, pixel depth: 8 bits, window size (3-13).

however, it scales similarly to logic and is dependent primarily upon window size.

Finally, it is noteworthy to compare our results to our previous implementation [3]. With the same parameters, our current implementation uses nearly exactly the same logic resources (59% versus 57%), the same amount of BlockRAM (26 out of 40), and runs at 58MHz instead of 26MHz. The frequency discrepancy is due to the fact that our previous design included support logic to drive the USB host interface. At 58MHz, the current design can support over 300 320x240 grayscale frames per second.

To get a more detailed notion of trends, a single maximum disparity of 20 pixels is chosen for Fig. 4, which plots FPGA utilization and clock frequency versus window size between 3 and 13 pixels square. Again, the trends are as expected,

with utilization scaling with window size, and the clock frequency trending downward, although with significant noise attributable to the probabilistic nature of the place and route processes.

Finally, run times for each implementation with the Xilinx suite of tools on a 3.4GHz Intel Xeon workstation were between 5 and 30 minutes, depending on utilization. Code generation completes in under a second.

III. CONCLUSIONS AND FUTURE WORK

We have presented our work in developing a parameterized stereo vision core which allows for the rapid exploration of the design space of stereo vision systems. Our implementation functions as expected, and can dramatically decrease the time necessary to implement stereo vision systems.

We are currently developing area and performance models to facilitate accurate estimation of resource utilization so users may make more informed design decisions without needing to perform an implementation. We are also expanding our code generator to implement a suite of stereo vision algorithms that include optical flow, phase-based, and other approaches, allowing the user to explore an even wider range of solutions with ease.

REFERENCES

- [1] R. Zabih and J. Woodfill, "Non-parametric local transforms for computing visual correspondence," in *ECCV '94: Proceedings of the third European conference on Computer Vision (Vol. II)*, 1994, pp. 150–158.
- [2] R. Zabih, "Individuating unknown objects by combining motion and stereo," Ph.D. dissertation, Stanford University, 1994.
- [3] C. Murphy, D. Lindquist, A. M. Rynning, T. Cecil, S. Leavitt, and M. L. Chang, "Low-cost stereo vision on an FPGA," in *Proc. 15th Annual IEEE Symposium on Field-Programmable Custom Computing Machines FCCM 2007*, 23–25 April 2007, pp. 333–334.
- [4] R. D. Henkel, "Fast stereovision by coherence detection," in *CAIP '97: Proceedings of the 7th International Conference on Computer Analysis of Images and Patterns*. London, UK: Springer-Verlag, 1997, pp. 297–304.
- [5] M. Z. Brown, D. Burschka, and G. D. Hager, "Advances in computational stereo," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 8, pp. 993–1008, 2003.
- [6] K. Mühlmann, D. Maier, J. Hesser, and R. Männer, "Calculating dense disparity maps from color stereo images, an efficient implementation," *International Journal of Computer Vision*, vol. 47, no. 1-3, pp. 79–88, 2002.
- [7] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International Journal of Computer Vision*, vol. 47, no. 1-3, pp. 7–42, 2002.
- [8] J. Woodfill and B. V. Herzen, "Real-time stereo vision on the parts reconfigurable computer," in *IEEE Symposium on FPGAs for Custom Computing Machines*. IEEE, April 1997.
- [9] A. Darabiha, J. Rose, and J. W. Maclean, "Video-rate stereo depth measurement on programmable hardware," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, 18–20 June 2003, pp. I-203–I-210.
- [10] D. K. Masrani and W. J. MacLean, "Expanding disparity range in an FPGA stereo system while keeping resource utilization low," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 25–25 June 2005, pp. 132–132.
- [11] D. Scharstein and R. Szeliski, "High-accuracy stereo depth maps using structured light," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, 18–20 June 2003, pp. I-195–I-202.
- [12] C.-C. Wang. Vision and autonomous systems center's image database. Carnegie Mellon University. [Online]. Available: <http://vasc.ri.cmu.edu/idb/>

AC 2008-1961: A SEMI-AUTOMATIC APPROACH FOR PROJECT ASSIGNMENT IN A CAPSTONE COURSE

Mark Chang, Franklin W. Olin College of Engineering

Mark L. Chang is an Assistant Professor of Electrical and Computer Engineering at the Franklin W. Olin College of Engineering.

Allen Downey, Olin College of Engineering

Allen Downey is an Associate Professor of Computer Science at the Franklin W. Olin College of Engineering.

A Semi-Automatic Approach for Project Assignment in a Capstone Course

Abstract

This paper presents a semi-automatic approach to assigning students to project teams for a year-long, industry-sponsored senior capstone course. Successful assignment requires knowledge of at least individual project requirements, student skills, student personalities, and student project preferences. This mix of hard skills, soft skills, and interpersonal impressions requires human involvement to produce a high-quality assignment. The importance of faculty input often requires that the assignment process be labor- and time-intensive.

Our approach attempts to reduce the time required to perform this assignment by selectively automating parts of the task flow. An automated search uses a randomized greedy algorithm combined with local optimizations to explore a large space of solutions. Candidate “good” solutions are then presented to capstone faculty. Criteria such as skill set, student capability, and personality compatibility are applied by human evaluators to reduce the candidate solution set. These candidate solutions are then distributed to small groups of faculty to look for improvements using system-generated tables of options.

This approach leverages automation at appropriate stages while keeping the experts—the faculty—involved in the selection process. Our initial implementation has reduced the time needed to select an allocation by about a factor of three over previous manual approaches.

Introduction

As engineering programs at colleges and universities strive to make pedagogical reinventions, faculty are experimenting with active learning methods to bring authentic engineering experiences into the classroom. A prominent feature of many project-based learning approaches is the use of student teams to solve complex problems. One of the significant challenges is therefore the assignment of students to teams.

In an experience such as a final-year senior engineering capstone, the administrative burden of team formation is often exacerbated by the needs of a more complex, department- or college-wide capstone program. Issues may include larger teams, interdisciplinary needs of projects, satisfying external constituencies, budgeting, and more. These higher stakes make a high-quality team selection process even more important.

In this paper we present a semi-automatic approach for placing students onto project teams. The chief goals of using this approach are to save personnel time and increase the level of satisfaction for all users. The users for our system include students, faculty, and the capstone program.

The Team Formation Problem

The Franklin W. Olin College of Engineering requires all students to complete a two-semester,

senior-year, engineering capstone project course. This course is the culminating engineering design experience for our students, and strives to provide real-world engineering problems and experiences from paying industry sponsors. Each year, approximately 75 students participate in 13-15 projects, and it is the job of the capstone program to try and best assign these students to the projects.

Our capstone program began in the fall of 2005, and is heavily modeled upon the Harvey Mudd Clinic program.

Successful team formation requires knowledge of project requirements and student skills and preferences. Students are expected to work with external sponsors as well as faculty and each other, and they are required to manage a significant budget, so we need to consider “soft” skills like leadership, teamwork and communication along with more technical skills.

Many of the projects are multi-disciplinary. Most include mechanical, electrical and/or software components, and many involve areas such as industrial design, environmental studies, ethnographic studies, and business/entrepreneurship.

Effective team formation is important for the short- and long-term goals of the program. In the short term, if students feel that the allocation conforms to their preferences and interests, they are likely to have higher morale and motivation. In the long term, we expect a good match between projects and student skills to yield good project outcomes, which is important for sustaining external support for the program.

The data we use to form teams comes from several sources, including student transcripts and project descriptions from external sponsors. Information about student preferences comes from a survey we administer at the beginning of the academic year. Students are given a short description and presentation on each project. They have a few days to ask questions, investigate the projects and the sponsors, and to talk to each other. Then we ask them to complete an online survey.

The survey asks students to score each project on a scale from 1-5, where 1 indicates no interest, 5 indicates strong interest, and 3 indicates that the student would be willing, but not necessarily happy, to work on a project.

Students are also allowed to identify up to two other students they do not want to work with. To encourage students to use this option sparingly, we ask them to name another student only if they believe that being on the same team with that student would be detrimental to the success of the project.

We have used almost the same survey instrument for all three years of the program. The rationale for this simple preference survey is that students will self-select projects that they want to participate in, and have the capability to make an impact on.

From student transcripts, we have each student’s major, courses taken and grades, and grade point average (GPA).

Some issues arise in using this data set in an automated fashion. Some projects are more popular than others; projects that attract few students constrain the space of feasible allocations. Also, there are usually a few students who are identified as an anti-preference by a disproportionate number of classmates. Assigning these students to a project also constrains the space of solutions. Nevertheless, in the three years of the program, it has always been possible to staff each project with interested students without violating any anti-preferences.

An important source of soft data is the knowledge of the people participating in the allocation process. The people “in the room” for our process include the capstone faculty, the Dean of Students, the Dean of Faculty, and the capstone program director. From classes and other interactions, the faculty have personal knowledge about each student. The Dean of Students often knows about student life histories and interpersonal relationships. The Dean of Faculty provides oversight of the process from the perspective of educational outcomes. The Program Director provides oversight from a programmatic perspective.

The next section explains how we use data from the survey and other sources to allocate students to projects.

A Semi-Automated Approach

Motivation

In a fully-automated process, we would formulate allocation as an optimization problem and use a program to search for an optimal solution, where “optimal” means that the solution satisfies all constraints (like the number of students on each project) while minimizing a cost, like violations of student preferences, or maximizing a complementary benefit.

There are several problems with this approach:

- It requires all relevant information to be encoded in a way that can be manipulated by the program. This is easy with “hard” data, like the results of the student survey, but impossible with “soft” data, like our knowledge of students’ personalities.
- It requires all constraints and costs to be quantified. Again, this is easy for some of our goals, like assigning students to projects they are interested in, but either complicated or impossible for other goals, like assuring an appropriate mix of skills for each project.
- It requires all tradeoffs between conflicting goals to be quantified. There is no general procedure for making these kinds of tradeoffs; it is only possible to consider, and reach a consensus about, specific cases.
- It requires all participants to trust the system and believe that the outcome is optimal. This is only possible if the process gives participants a mental model of the space of possible solutions.

The faculty and students involved in the capstone have different and sometimes conflicting goals and values. There is no “optimal” allocation that will satisfy everyone. What is needed

is an allocation *process* that assures everyone involved that the outcome is an appropriate compromise that is as good as possible.

Our Approach

To achieve this goal, we propose a semi-automated approach that tries to combine the efficiency of an automated search with the user satisfaction of a human-centered approach.

Our approach iterates between two phases:

1. The first phase uses an automated search to find a pool of allocations that are good candidates according to a coarse cost function.
2. In the second phase the capstone faculty evaluate proposed allocations and either accept one of them or adjust the parameters of the automated search and generate a different set of candidates.

There are a number of advantages to this two-phase approach:

- It uses computers to do what computers do best, and humans to do what humans do best.
- It takes advantage of both the hard data, which can be encoded and used during the automated phase, and soft data, which may exist only in the heads of the users.
- It can deal with both hard constraints, which cannot be violated, and soft constraints, which can be violated at a “cost.”
- It gives the users opportunities to fine tune the allocation to balance the needs of students, faculty, and the program as a whole.
- It allows the users to discover and modify constraints and costs as the process goes along.
- It helps the people involved understand what the space of possible solutions looks like so that their expectations are calibrated appropriately.

Regarding this last point, we have observed that different people approach this kind of problem with different mindsets. One classification divides people into “maximizers” and “satisficers.” Satisficers generally try to find an *acceptable* solution; maximizers try to find the *optimal* solution. The approach we are proposing can help avoid conflicts between these mindsets.

For satisficers, the process takes significantly less time and produces a large number of high-quality candidate solutions to choose from. For maximizers, the use of a computer algorithm

to search a very large space of solutions can alleviate the anxiety of a fully-human approach by providing an overall view of the landscape of solutions. Also, starting with a small pool of candidate solutions, maximizers can look for local optimizations until they are convinced that no further improvements are possible.

Our implementation

The following is a more detailed description of the steps we followed.

- Automated search for feasible solutions: Using information from the survey of student preferences, we use a greedy algorithm with local optimization to generate a large number of good quality feasible solutions.

The automated search is guided by a cost function that assigns a score to each candidate allocation. Undesirable features are assigned a “cost”; the total score for an allocation is the sum of all costs.

Most constraints are enforced by assigning high costs for violations. For example, understaffing or overstaffing a project costs 10000 points; in practice this means that no solution with this feature will win, but the program can consider allocations that violate this constraint as it moves from one local optimum to another.

Assigning a student to a project with a preference of 5 (“high level of interest and strong match of skill”) is free; a preference of 4 costs 1 point; a preference of 3 costs 5 points; a preference of 2 costs 1000 points and a preference of 1 (“not interested, or no match of skills”) costs 10000. An allocation that assigns a student to a project with preference 2 or 1 is considered unacceptable, but can be used to move from one local optimum to another. The program tries to avoid assigning students to projects with preference 3, but in our experience so far, it has not been possible to avoid putting a small number of students in that position.

If two students who conflict are assigned to the same project, that costs 100 points. This weight reflects our desire to accommodate anti-preferences almost absolutely while still considering that violating an anti-preference might allow the program to explore a part of the solution space that yields a better global allocation.

The program that generates solutions works in three phases:

1. During the first phase, the program uses one of two probabilistic greedy algorithms to generate an initial allocation. One algorithm enumerates the students in random order and assigns each student to the available project with the highest preference. The other algorithm enumerates the projects in random order and chooses the student with the highest preference for the project, repeating until all students are allocated.

Because the algorithms enumerate the students and projects in random order, and break ties at random, they generate different initial allocations each time they run.

Both algorithms generate allocations with an acceptable number of students on each project, and they tend to assign students to projects with high preference, but they make no attempt to avoid violating anti-preferences.

2. In the second stage, the program considers all possible trades (swapping two students) and moves (moving a student from one project to another) in random order. Any trade or move that improves the overall score for the allocation is accepted.

Stage 2 is repeated until there are no moves or trades that improve the allocation.

3. In the third stage, the program identifies the students who are most unhappy, chooses one at random, and takes “desperate measures” to make the student happy, even at the cost of violating an anti-preference or making another student unhappy. The effect is to move the allocation out of a local optimum; the program then repeats Stage 2 to find the new local optimum.

Stage 3 is repeated as long as it continues to find improvements. If a desperate measure fails to find a new optimum, the program discards the allocation and starts again with Stage 1.

As the program runs, it records all solutions with a global cost below a certain threshold. The longer the program runs, the more low-cost solutions it finds. In our experience, the program tends to find a few good solutions in 10-20 minutes; after a few hours it seems unlikely to do any better. Of course, we are making no effort to find a true optimum, and we wouldn’t know if we found it.

Among the lowest-cost solutions, there tend to be repeating patterns: one project might be assigned the same team in many solutions, or a set of students might tend to be on the same team.

After the program has run for a while, we print about 20 of the best solutions and bring them to the next stage.

- Scoring candidate solutions: Capstone faculty evaluate possible allocations. If an acceptable solution emerges at this stage, the process can stop, but it is likely that additional constraints will be identified.

We distribute hardcopies of the top-20 allocations to the capstone faculty for evaluation. Faculty advisers tend to evaluate their own teams, taking into account factors like the major, skillset and GPA of the students as well as personal information based on prior experience. For each allocation, faculty assign a score to the proposed team on a 5-point scale: 5 indicates a very strong team, 4 is strong, 3 is acceptable but borderline, 2 is unacceptable but possibly reparable, and 1 is unacceptable.

During this process, faculty identify recurring patterns and problems. New constraints might be identified that eliminate some allocations from consideration.

This stage can be decentralized; that is, the faculty involved don’t have to meet to evaluate the candidate allocations. The faculty in charge of the allocation process (the authors of this paper) can aggregate responses from faculty advisers and other participants.

- Generation of more solutions: If additional constraints are identified, they can be enforced automatically by modifying the search code, or candidate solutions can be filtered by hand.

We run the search program again with new constraints and select a new set of candidate allocations.

- Selection of best candidates: In a second round of scoring, capstone faculty identify a small number of candidates allocations that are either acceptable or nearly acceptable.

During this stage, for the first time, faculty are all in the same room. By this time, they have a sense of what the search space looks like and an idea of how good an allocation is likely to be found. We have identified allocation features that are “showstoppers” and generated solutions that avoid obvious problems.

- Search for local optimizations: Capstone faculty are divided into teams; each team is given a candidate allocation to work with. For a given allocation there is often a particular problem they are asked to resolve. Their search is facilitated by automatically-generated tables of possible moves and trades. This stage ends when all teams decide that they cannot find additional improvements.

The teams present their final allocations to the group. The faculty discuss the features of the proposed allocations with the goal of choosing one by consensus.

Evaluation

Our capstone program started in the 2005-6 academic year so we are offering it for the third time this year.

In the first two years we used a computer program to collate and print the data from the survey, but the allocation was performed manually by capstone faculty. We implemented the approach described in this paper for the first time this year.

The following sections describe our experiences in the first two years.

Year 1

In the inaugural year, the allocation process was completely manual. Without significant forethought we implemented a greedy algorithm based on student preferences, and then searched for local optimizations.

With a single piece of paper representing the survey results for each student, we allocated each student to the project with the highest score. At the end of this round, many projects were understaffed and a few were overstaffed.

Next we started an ad-hoc parallel search for better solutions. This often resulted in faculty making local trades between projects to form teams of 4-5 students per project. Other times faculty would ask the entire room for a student that met certain criteria, such as ranking their

project highly and having a particular major. Conflict resolution was done by individual faculty.

This approach yielded a relatively good allocation of students: of approximately 66 students, all but one was placed on a team they ranked 5 or 4 (highest and second highest). The remaining student was placed on a team ranked 3. However, the allocation process took a long time—approximately 96 person hours (8 hours by 12 people). The process was also prone to human error; at one point we thought we were done, then discovered that an anti-preference was accidentally violated. It took another 90 minutes to find a set of trades and moves that solved this problem.

At the end of the allocation, the faculty overwhelmingly agreed that a better process was needed.

Year 2

In the second year of the program, in an effort to provide more global guidance and a structured method for allocating students, we implemented a process similar to professional sports draft.

In round-robin fashion, each faculty member chose the best student for their team. This process, while seemingly appropriate, was quickly found to be flawed.

After a few rounds we reached an incomplete allocation where many projects were understaffed, but the remaining students could not be assigned to the understaffed projects without violating student preferences.

The draft algorithm fails because it solves the easy part of the problem first and leaves the hard part for last. This violates a basic heuristic of allocation: “Pack the big rocks first.”

Students with good academic records, interest in several projects, and few anti-preferences are “small rocks;” in a draft algorithm, they are likely to be chosen first. Students with weaker records, interest in few projects, or more anti-preferences are “big rocks.” If they are left until the end, there will be nowhere for them to go.

To make things worse, this process puts the faculty in a competitive mindset, with each one trying to select the best teams for their projects. This makes trading more difficult.

After a few rounds, we were stuck in a locally-optimal solution that was far from globally optimal. There were few small, local changes that made things better; we needed to make big, non-local moves to get to another part of the solution space.

But in the competitive atmosphere of a draft algorithm, faculty are less likely to accept trades that involve giving up a “good” student (greedily selected early in the process) for a weaker student (left until the end). This problem is compounded because in the early rounds, faculty associated with a popular project are able to assemble a “dream team,” which raises

their expectations unreasonably.

The result was a qualitatively worse solution than the previous year. Eight students were placed on a project they ranked 3 (compared to 1 in the previous year). The process took as long or longer (about 8 hours) and seemed more frustrating.

We left the room with the feeling that there were better solutions we could not reach. This feeling was confirmed when we developed the algorithm proposed here; using the data from Year 2 for development and testing, we were quickly able to find solutions that allocated only 3 students to projects ranked 3. Faculty who evaluated these solutions agreed that they were qualitatively better than the allocation we generated manually.

From this experience, it was clear that we needed to refine the process and introduce some degree of automation.

Year 3

For the third year, the authors prepared the approach detailed in the previous section.

The process took much less time—fewer than 40 person hours. Capstone faculty had an initial one hour meeting to perform the parallel scoring, then met again after new candidate solutions were produced to determine the final allocation. The number of person-hours is still large, but the process was perceived as less stressful and more satisfying.

More importantly, the allocation we generated was met with high satisfaction from both faculty and students. Almost all students were assigned to projects they ranked 5 or 4; only one was assigned to a project ranked 3.

Exportability

The authors expect this approach to be useful in other contexts where assigning students to teams is necessary. In exporting this approach, we must take into account scalability. Our program is small—we assign about 75 students to about 15 teams—and our process depends on the detailed familiarity of the faculty with the students.

If the ratio of students to faculty were much larger, the first phase of manual optimization might be less valuable for improving solutions, because the faculty would have less “soft” data about students. But this phase might still be helpful for giving the faculty an overview of the space of possible solutions, which makes it easier to achieve consensus and the feeling that the chosen solution is about as good as possible.

If the number of projects were much larger, the second phase of manual optimization might be less effective because of the increased number of possible moves and trades. For example, in our process, 10 faculty are divided into 3 teams, each attempting to optimize a candidate allocation of 75 students onto 15 teams. If the number of projects, students and faculty were much larger, it would be difficult for the teams to find local improvements.

One possible solution would be to introduce hierarchy into the human optimization of candidate allocations. Instead of groups of faculty assessing all teams, one could assign mechanical engineering projects to mechanical engineering faculty to optimize. However, this would restrict the faculty to moving students between a subset of teams, and might not work as well for multidisciplinary students and projects. The resulting solutions may then be less globally optimal, as it would be difficult for any group of faculty to have a global view and understanding of the full allocation.

In either case, scaling the problem size up makes the use of a semi-automated system more attractive. The use of a faculty-driven parallel search process (as in Year 1) becomes more difficult as the size of the problem increases. The use of a draft model (as in Year 2) also becomes more difficult as it gets harder to find, and get faculty to accept, the big moves that are necessary, toward the end of the process, to move from a local optimum to a more globally optimal solution.

Conclusions

The authors present a semi-automated approach to allocating students onto project teams for a senior capstone experience. The allocation is a high-stakes endeavor for all constituents, as increased student agency and happiness can lead to strong motivation and a successful project. Our approach is novel as it selectively automates parts of the process while keeping the human operators in the optimization loop at appropriate times. Using this approach, we have demonstrated initial success not only in reducing the time required to perform the allocation, but also improving the quality of the allocation.

AC 2008-1987: A BLANK SLATE: CREATING A NEW SENIOR ENGINEERING CAPSTONE EXPERIENCE

Mark Chang, Franklin W. Olin College of Engineering

Mark L. Chang is an Assistant Professor of Electrical and Computer Engineering at the Franklin W. Olin College of Engineering.

Jessica Townsend, Franklin W. Olin College of Engineering

Jessica Townsend is an Assistant Professor of Mechanical Engineering at the Franklin W. Olin College of Engineering.

A Blank Slate: Creating a New Senior Engineering Capstone Experience

Abstract

This paper presents some of the challenges, successes, and experiences in designing a new senior engineering capstone program at the Franklin W. Olin College of Engineering. Senior capstone design programs in engineering colleges have evolved over many years and are often modified and reinvented to keep up with the needs of both students and external constituencies. Harvey Mudd College's Clinic program is one of the largest and longest-running capstone programs in the country that relies heavily on industry sponsors to provide real world problems and funding to execute the projects. For many reasons, and in no small way because of its track record of success, our own capstone course offering is modeled closely upon the Harvey Mudd Clinic program.

However, completely importing a well-established program into a different context would be haphazard at best, and would ignore a unique opportunity to retool the program to meet the specific needs of a different college. This paper presents our experience in developing SCOPE, the Senior Consulting Program for Engineering at Olin College, and applying lessons learned from the Clinic Program and other successful capstone programs. We discuss the difficulties such as recruiting industry sponsors for a new and unproven program, developing assessment methodologies, and developing the policies and procedures needed to keep the program running smoothly and in a sustainable fashion. Through this narrative, the authors endeavor to inform other programs that are in need of modification, and educators who find themselves with the opportunity to start a capstone program from the ground up.

Olin College Background

Franklin W. Olin College of Engineering is a new, four-year engineering school in Needham, Massachusetts. The college was started and funded by the New York-based Olin Foundation, which has awarded grants totaling more than \$300 million to construct and fully equip 72 buildings on 57 independent college campuses. Starting in the late 1980's, the National Science Foundation and engineering community at large started calling for reform in engineering education. In order to serve the needs of the growing global economy, it was clear that engineers needed to have business and entrepreneurship skills, creativity and an understanding of the social, political and economic contexts of engineering. The F.W. Olin Foundation decided the best way to maximize its impact was to help create a college to address these emerging needs. The Foundation's commitment in excess of \$400 million to Olin College remains one of the largest such commitments in the history of American higher education.

The college officially opened in Fall 2002 to its inaugural freshman class. During the prior year, thirty student "partners" worked with Olin's faculty to create and test an innovative curriculum that infused a rigorous engineering education with business and entrepreneurship as well as the arts, humanities and social sciences. They developed a hands-on, interdisciplinary approach that better reflects actual engineering practice. From the beginning, it was clear that a two-semester,

senior-year, engineering capstone project course would be part of the curriculum for all Olin students. Just prior to the first year of instruction at Olin, the Curricular Decision Making Board put together plans for the senior year, and noted that “by the time students are seniors, they’ll be doing the real engineering on their own, in a year-long capstone project that will look very much like professional practice.” Development work on this program, eventually named SCOPE, the Senior Consulting Program for Engineering, began in earnest in the fall of 2004, when the first SCOPE Director was hired, Dr. David Barrett.

The unique challenge, and, perhaps, the greatest advantage, in developing the SCOPE program was the absence of a pre-existing capstone program. The intent was to launch the SCOPE program during the first senior year offered at Olin College. Although an overall vision for the senior year had been developed, faculty and administration needed to create and implement a fully-functioning capstone program so that the very first Olin College class would receive as close to the same capstone experience as those students that followed. Olin College is certainly not the first institution to develop a new engineering capstone program, therefore the most logical course of action was to look at how other schools have run their capstone programs.

Due in part to its similar mission, scale, and approach to undergraduate engineering education, an obvious model for Olin’s capstone program is the Harvey Mudd Clinic Program. The Clinic Program is the longest running sponsored capstone program for undergraduates. For reasons detailed in later section, the Clinic Program became the blueprint from which the SCOPE program was designed.

Goals of this paper

In writing this paper, the authors intend to describe through a narrative, the history and evolution of the program over its first three years. The intention is to put the reader into the context of developing a capstone course from the ground up such that our experiences may inform the efforts of other faculty and administrators seeking to build, expand, or enhance their own capstone programs.

It is important for the reader to keep in mind that while this paper was written in consultation with faculty and administrators involved in all aspects of the capstone program, it represents the interpretation of the challenges, successes, and experiences of many people by the two primary authors. We have made an effort to synthesize our observations with those of our colleagues, and would like to acknowledge the hard work of all those involved in the design and execution of the program.

The Harvey Mudd Clinic Program – A Major and Direct Influence

Harvey Mudd College was founded in the mid-1950s. The original curriculum was strong in engineering science analysis, but engineering practice and professional training was lacking. Specifically, students were not getting experience in solving open-ended problems, or with project and team skills.¹ The Harvey Mudd Engineering Clinic Program was started by professors Jack Alford and Mack Gilkeson in 1963 as a way to address these issues by bringing

real-world engineering problems to the campus in a close approximation to professional engineering practice.²

The Clinic Program has been in operation for more than 40 years and has proven to be sustainable, both financially and logically. Funding for Clinic comes directly from the sponsoring companies. Students work in four- or five-member teams, along with a faculty advisor and a liaison from the sponsoring company on a project that is meaningful and useful to the sponsor. The students are responsible for developing the project statement, goals, and outcomes in concert with the sponsor liaison, and then are fully responsible for carrying out the project. The faculty advisor is a mentor, coach, and assessor. The student teams give a number of design presentations to the Harvey Mudd College audience, the sponsor, and to the general public as part of the end-of-year Projects Day.³

The Evolution of SCOPE

Year Zero: The First Steps in Developing the Olin College SCOPE Program

In creating a new engineering college, the founding faculty and administrators were presented with the unique challenge of wholesale *invention* of a four-year engineering curriculum. From the outset, every version of the Olin curriculum, “retained a two-semester senior capstone project course.”⁴ At the conclusion of Olin College’s first sophomore year, it was clear that the successful execution of a year-long, authentic, project-based experience would require significant planning effort. While there exist many models of capstone experiences, our then Dean of Faculty, Dr. Michael Moody, advocated the adoption of the Harvey Mudd Clinic Program as our model in his June 2004 memorandum on the upcoming senior capstone course. His experiences as Chair of the Mathematics department at Harvey Mudd, participating in the Math Clinic Program, and having significant exposure to the Engineering Clinic Program, greatly influenced this decision. As a limitation of his experiences, and because of the anecdotal successes of the Clinic Program, this was the only model considered. By leveraging his direct experiences and intimate knowledge of a successful program, much of the work of “creating” the program was lessened. Dr. Moody remains a key leader in the continued evolution of our program, and his experiences were and continue to be invaluable to our efforts.

In creating a sponsored program, two key decisions needed to be made which strongly influenced the rest of the program: the grant vehicle for sponsors to make monetary contributions to the program, and the dollar amount to charge for participation in the program. The fee for the program was set at \$50,000, compared to approximately \$41,000 for the Clinic Program in 2006/2007. The justification and rationale for this price tag was that given the nature of our institution, we should be able to support a program of comparable complexity and cost to the Clinic Program. Specifically, given the high caliber of our students, the residential nature of our college, proximity to high-tech companies in the greater Boston area, our qualified faculty and administration, and the two-semester capstone engagement structure, we believed to be in a position to offer an experience similar in scale to the Clinic Program. Our costs being higher than the Clinic Program come primarily from taking into account program startup costs.

With respect to the sponsorship vehicle, the program opted for a fairly complex contractual agreement between the College (signed by students and faculty advisers) and the corporate sponsor. This agreement addressed issues such as confidentiality, intellectual property rights, indemnification, and set baseline expectations for deliverables. It was felt that in the startup phase of the program, protecting the students and the College through a legally binding document would be preferred by both parties. This approach received considerable thought and revision in later years.

By April 2005, the program had been named SCOPE, and key personnel that would ultimately be responsible for the day-to-day execution of the capstone were chosen. This included the SCOPE faculty, the student representatives, an external advisory committee, and the program director. Initial sponsor solicitation began, and two signed contracts were in place with 30 potential candidates being actively courted. It was at this point that the faculty who were to participate in SCOPE starting in September of 2005 were notified. They were asked to target companies they would like to work with, help prioritize the current list of candidate sponsors, and help respond to new contacts and projects.

One unique aspect of the SCOPE program is the support of the Franklin W. Olin SCOPE Project. As Olin College itself is the result of a philanthropic organization, and each of our students the recipient of philanthropy, it was important to provide an outlet for student philanthropy through engineering experiences. The Franklin W. Olin SCOPE Project was conceived of as a student-generated, Olin-funded SCOPE project of the same scale and level of technical challenge as a paying sponsor project. The vision was to have students craft detailed project proposals, of which one or two per year (depending on budget flexibility) would be selected and funded by the College.

Reflections on Year Zero

Development of Pedagogy: During the development year prior to the launch of SCOPE, a great deal of administrative energy was spent on soliciting potential sponsors. This focus, however, left many pedagogical issues untouched until the summer preceding the launch of the program when many of the faculty responsible for advising teams and running the program finally joined the development effort. Important issues such as how to best utilize the class time within the weekly schedule, program-wide curricular milestones and objectives, common deliverables, grading and assessment rubrics and techniques, and team advising methods and techniques, for example, were only briefly discussed in meetings and drafted into the handbook before launch. After the program got underway, many of the details were either finalized just in time to be implemented, or the faculty members were left to decide their own course of action.

In retrospect, there was not enough development time for faculty before launch to permit a thorough investigation into what portions of the teaching tasks should be common between faculty, and which portions were best left up to individuals to decide. A common struggle was, and continues to be, finding the balance between treating SCOPE as teaching multiple sections of the same course and therefore requiring common practices; versus acknowledging that each project is unique and therefore requires specific decisions regarding policies such as advising/mentorship and grading. Providing more time for faculty development of the

pedagogical tools would significantly ease the anxiety of all parties, and would perhaps ensure a more even level of performance between teams.

An important lesson learned was that although the adoption of the Clinic Program gave our program a strong and successful framework to build upon, it does not adequately inform the day-to-day operations and procedures that one must have to actually carry out the program. More preparation time and faculty thought would have made the first year a much less anxiety-filled for all parties. This preparation would improve the chance for success in any program.

Capstone Advising Committee: While the program had as a model the Clinic Program, one thing that we did not import from Harvey Mudd was an advisory committee. Harvey Mudd has an approximately 20-person advisory committee that includes members of the Board of Trustees and members from industry. Their purpose is to help sustain and improve the Clinic Program by providing feedback to faculty, staff, and students, and through conducting a satisfaction survey of the industry partners. While the SCOPE program has a very small handful of external advisors, they serve more in a technical advisory role for individual projects rather than aiding the program as a whole. While recruiting an advisory committee can be challenging, their presence could provide some useful and directed industry and academic feedback to improve our program. Without them, we run the risk of being short-sighted and limited in our contextual understanding of the corporate partner experience. We will soon be starting development efforts to create such an advisory committee for SCOPE.

Contracts: The contractual agreement also received some criticism as the first summer of sponsor solicitations came to a close. While it was thought that a more industry-flavored contract would make it easier for sponsors to agree to fund a project, in many instances, it was a bottleneck in negotiations. If we had used a less formal agreement mechanism—a letter of understanding, for example—the legal teams for the sponsors may have been less inclined to get involved. With the complex legal document we drafted, however, as it was clearly a contract, it required the attention of the legal departments within potential corporate partners, slowing (and sometimes halting) negotiations.

The agreement as a contract also changes the psychology and expectations of the program as whole. With a firm legal document in place, sponsors might have felt the relationship more as a subcontractor rather than primarily an educational partnership. This significantly alters the expectations of the sponsor and might put inappropriate pressures on the student teams.

Successful Launch: In the positive, going into launch, the program was very successful in soliciting 13 paying external sponsors for a completely new and unproven program at an almost equally new and unproven institution. Much of the credit for this success is given to the tireless effort of Dr. David Barrett, the SCOPE program director. Coming directly from 25 years in research and industry, Dr. Barrett brought with him significant relationships with individuals and corporations in a position to sponsor a project. With a concrete capstone framework, Dr. Barrett could promote a well-structured and proven program that met corporate needs. Additionally, the college had received a large amount of media coverage during its startup years, and had already established itself as an engineering education innovator and a business-friendly

atmosphere. The general excitement surrounding Olin piqued interest among corporations looking for new ways to engage academia.

Year One: Off and Running

The launch of the SCOPE program in the first week of September 2005 coincided with the last of the 13 contracts being signed. In the summer months leading up to the launch, many of the mechanisms needed to successfully execute a project were put into place but never tested. The return of the students and the beginning of the SCOPE program would put not only our planning to the test, but also our ability, as a program, to be agile and adapt to the needs of the program. In this section we will reflect on the year's experiences in three sections: curriculum and student experience, facilities, and program & infrastructure. We will discuss some of the challenges and lessons learned in each of these areas.

Curriculum and Student Experience

As previously mentioned, the amount of pedagogical preparation that was done prior to starting the program was not as much as any faculty would have preferred. The negative impact of this is perhaps minimized at our institution due to the fact that significant portions of the curriculum already involved large-scale projects and student-directed learning. The students had experience working with the unknown, and faculty had experience leading and advising these efforts. However, it was still not decided when to treat the capstone as a single course taught by many faculty, and when to treat it as a collection of completely separate and autonomous projects.

While there is no correct answer, the vagueness was felt not only by the faculty, but also experienced by the students, as individual faculty led their teams in their own unique ways. In hindsight, it would have been useful to devote more time to discussing how the program could most effectively support these unique project experiences by enforcing uniformity in places, and encouraging autonomy in others.

These decisions directly influence much of the student experience. Program activities such as design reviews, assessment and grading, and faculty-student interactions are all areas that draw from either programmatic direction or from individual faculty preferences. Without significant preparation, these many activities become more ad-hoc in nature, and students' motivation can suffer from the lack of coherency.

It became more clear as the year progressed that maintaining student motivation was a key component to success. An area that could see improvement is in the scheduling of activities, especially those toward the end of the second semester. As seniors start finalizing their post-graduation plans, having accepted job offers or gained admission in graduate school, motivation can be problematic. One lesson learned in our inaugural year was the importance of scheduling end-of-year events. In particular, the culminating final presentations to sponsor personnel was scheduled after finals. Many students found themselves "burned out" from their capstone experience, the crunch of finals for their other classes, the stress of graduation and impending major life changes, and simply the wear of four years of college. Positioning such a high-stakes and high-visibility event such as final presentations at a time when many students feel they should be celebrating can spell disaster.

Finally, in contrast to the Clinic Program, our students have no direct training in project management. The program offered several hours of instruction and guidance, but only to the student project coordinators. By introducing a required course in project management earlier in the curriculum, or by introducing formal instruction in various required project courses leading up to the capstone, our students might find managing their peers and interfacing with liaisons easier.

Facilities

After just a few weeks, it became clear that the capstone program's impact was not isolated to the faculty and staff directly associated with the program. As projects moved forward, purchases, services, and space became more necessary. While the mechanisms for purchasing were already in place for the College as a whole, the disparate and frequent needs of the projects put a significant strain on purchasing personnel. Beyond purchasing, project needs found the students wanting physical space, computing resources, and IT resources that were not adequately planned for.

All teams were assigned their own team room, a small office outfitted with tables and chairs, a whiteboard and a lockable file cabinet, but many projects required additional physical space to store, fabricate, and test devices. Some needs were modest and could be accommodated in the team rooms or existing laboratories. However, the handbook had no guidelines for requesting rooms or other types of spaces, nor did the program have any significant predefined space allocated in advance. As is with most other colleges, excess space is rarely available. The process was therefore ad-hoc and required the involvement of many college administrators to help "find" space. This lack of immediate space influenced the student team's ability to develop an appropriate statement of work in partnership with the sponsor liaison as they could not make an assumptions about having dedicated facilities.

A lesson learned here is that if at all possible, the program and college should find dedicated work spaces for some fraction of the student teams with the assumption that some projects will have significant space requirements. Fortunately, the needs of the projects and the resources of the college came into alignment and physical space was temporarily granted when necessary.

While physical space only affected a few teams, computing resources were a problem for a wider selection of teams. The nature of team-based engineering often requires sharing files between students, sharing files off campus, and purchasing high-performance workstations. In the first year, we did not adequately prepare our IT department for the flood of requests from student teams for support and equipment. Complex issues such as maintaining confidentiality of digital information on a network, issues of trademark and liability when making information publicly available on web sites, and advanced development that required different access rules than typically allowed by the college network and computing infrastructure were not adequately addressed or anticipated. While we resolved the majority of these issues, it was a heavy additional burden on a separate department that could have been lessened with some planning.

Program & Infrastructure

As it was our first attempt at executing a set of large-scale engineering projects for corporate sponsors, no amount of planning could have prevented some of the challenges detailed in this section. However, a lesson learned was that perhaps solving some of these issues beforehand might make the road to a new capstone program a bit smoother.

Project confidentiality: In the first year of the program, we were not in a position to turn down many projects. Therefore, we found ourselves with several projects that had very restrictive confidentiality requirements. While the projects may have been successful and interesting, their closed nature prevented students from presenting much information regarding their work at design reviews. In addition to being logically problematic, the students missed opportunities to get valuable feedback on their work from anyone that had not signed the non-disclosure agreement.

Cross-registered students: Olin College has a very close relationship with Babson College and Wellesley College, allowing Babson and Wellesley students to freely cross-register in nearly any Olin course, including the capstone. From our experiences this first year, it became clear that given the high-stakes nature of working on sponsored projects, having a system to screen for appropriate students was necessary. Having an adequate match between potential students in both interest and skill set would give the project a better chance of success.

Human subjects policies and review: Due to the heavy design component of many of the projects, soliciting feedback from volunteers at various stages of product development would be a common practice. Protecting these users from harm is an ethical requirement and responsibility of any college. We did not have the sufficient infrastructure in place to perform human subjects review of the work related to the capstone. In many cases, it was suggested that students follow the human subjects practices and requirements of their sponsoring company. However, sometimes the internal corporate review committees did not move at the pace necessary to be useful for a student team with a short time budget, and sometimes corporations had no internal review boards to leverage. Having a more program-wide solution to this need would be both educational and practical.

Reflections on Year One

As the faculty looked back over the first year of SCOPE, it became clear that Olin students were utilizing the design skills learned earlier in the curriculum in their SCOPE projects. In many ways, design is at the center of the engineering curriculum at Olin. All Olin students take Design Nature during their first year where they receive instruction in design processes and methodologies. During their second year they take a class called *User Oriented Collaborative Design*, which focuses on including the user in the design process. Many teams found that these design processes and the collaborative approach were useful tools for their projects. Both faculty advisers and program sponsors noted the teams' strengths in this area.

A second area of strength noted was the communication skills of the students, particularly when interfacing with the sponsor companies and during design reviews and presentations. This was likely due to the emphasis on oral communication and presentations throughout the entire Olin curriculum and to the strong oral communication skills many students enter Olin with.

Finally, the students seemed able to handle many team dynamics issues on their own. The first senior class at Olin College had only 66 students, and most students were very aware of each other's work and communication styles going into the SCOPE program. This did not mean that team dynamics issues were non-existent, just that the teams had some skills to work through these kinds of issues already.

An area of weakness noted by many faculty advisers was the students' willingness to take on hard analytical problems. There was a sense that students were capable of this level of work, but had difficulty in setting up the problem, making assumptions and working out a first order model. This was attributed not to a lack of open-ended problems in the curriculum, but to a lack of open-ended problems that required engineering science analysis.

Although these issues that surfaced during the first year of SCOPE were regarded as important to address, it would be premature to make major changes to the program or to the Olin curriculum without more steady state data. The feedback regarding analysis was given to faculty responsible for teaching the engineering science classes, and some small changes were made programmatically. Most of the feedback we received from Year One was incorporated through faculty advising of teams. Faculty now had a year behind them and were able to bring those lessons learned, both on a team scale and a program-wide scale, to their advising. Having weekly SCOPE faculty meetings provided the best opportunity to compare notes and share experiences. This is something that Harvey Mudd faculty have noted as well, that learning to advise a Clinic Project is experiential learning for the themselves. The most valuable resource a new Clinic faculty member can have is a solid group of experienced Clinic faculty members to talk to. A cohort of dedicated faculty members is what will keep a successful capstone program running and sustainable.³

Years Two and Three: Small Changes, Gaining Experience

Much of the programmatic structure that was put into place during Year One was kept in subsequent years. One change made in Year Three was renaming the main student leadership position from "Project Coordinator" to "Project Manager." The original intent was that the Project Coordinator would handle many of the administrative and sponsor communication duties, but would not necessarily be "managing" the other team members. It was expected that students would take responsibility for keeping up with their work and all team members would ensure that work was distributed fairly. During the first two years it became evident that students would not have a problem with one student managing the project and the work, and some students preferred this altogether. In the end, the position was renamed, but more emphasis was put on letting students rotate through this position during the one-year project. After one semester, this leadership scheme seems to be working well for the teams.

There are several other major issues that the SCOPE Program has looked at and revamped over the first two and half years of the program. These are highlighted in the sections below.

Projects and Sponsors

During Year Zero, the SCOPE Director spent many months on the road recruiting sponsors for the first year of SCOPE. The pitch to sponsors included a description of the types of projects that would have the biggest return on investment for the sponsors, and would also provide the most meaningful learning experience for the students. As the SCOPE Director explains, “All companies have three classes of problems. Class 1: are the mission critical problems that your best people must focus on for survival. Class 2: are the strategic problems that have been sitting on your back burner for years, for lack of time and skilled labor to address them. Addressing these problems could significantly and positively affect your bottom line, provide a foundation for explosive growth or enable successful entry into a profitable new business area. Class 3: are low-level problems that have no significant impact on your corporation’s operations.” When talking to potential sponsors, we emphasize that Class 2 problems are ideal for SCOPE projects.

However, in the first few years of SCOPE, we did take on several projects that were not exactly Class 2 problems. One project was done for a small start-up and involved development of a technology that was on a mission critical path for the company. In this case the faculty adviser spent a lot of time managing the sponsor’s expectations of the outcome. Another project involved a technical analysis that was beyond the scope of the students’ abilities. The team spent the first semester attempting the problem and eventually went back to the sponsor with evidence that the project was better suited for a PhD dissertation, and asked the company if they had another project they could tackle. In this case, the company responded well, brought another project to the table, and in the end was pleased with the students’ efforts.

One way to help appropriately set sponsor expectations is in how the program is pitched. In our original efforts to recruit sponsors, we indicated that sponsoring a SCOPE team was very much akin to hiring five talented entry-level engineers who could make traction on a problem that that company did not have the resources to solve. After the first year, we adjusted our pitch to maintain more of a balance between a learning experience for the students and a benefit for the company. We also found that it became easier to help guide a sponsor towards an appropriately scoped project for our students.

Finally, we learned that projects that are highly successful have an involved and accessible sponsor liaison (the company representative that interfaces directly with the team).

Design reviews

All SCOPE teams are required to give regular design reviews to the Olin community throughout the year. The purpose and intent of the design reviews has changed over the three years of the program, with different faculty advisers setting different expectations for their teams. In general, the purpose is to present technical issues and challenges, show progress, describe and justify design decisions and receive feedback from the audience. The design reviews during Year One were not as interactive as we would have liked, and in Year Two we told students that they were expected to contribute whether they were on the presenting team or not. However, during the first two years of the program, teams were expected to give design reviews every two weeks. This was taking time away from the technical work that needed to get done, and in Year Three, teams have been presenting twice a semester instead.

Grading and Assessment

Prior to launching the program the SCOPE faculty did not explicitly discussed grading and assessment. While course outcomes, grading criteria and deliverables were described in the SCOPE Handbook, no common method for assessment was presented. Faculty were encouraged to provide students with a mid-semester grade, but by the end of the semester it was clear that the methods of assessment for the SCOPE projects varied quite widely.

The first major discussion of grading and assessment came at the end of the first semester of SCOPE where there was general agreement of what level of achievement each letter grade indicated. In subsequent semesters, faculty made progress in agreeing on a common set of deliverables and a common set of competencies and outcomes for SCOPE students. By the second year of SCOPE a set of peer assessment forms had been developed that were in use among most teams, and faculty advisers were more clearly communicating their grading policies to their teams. However, it is fair to say that assessment does still differ among faculty advisers, although more of an effort is made to streamline the assessment process, and faculty are communicating with each other more to maintain a similar set of expectations for their teams.

Finding the Balance

The creation of a capstone experience for our students was, and continues to be, a challenging effort that requires significant resources. In preparing this document, our goal was to discuss lessons learned along the way such that other programs seeking to incorporate new, or revamp existing, capstone programs would benefit from our experiences. One consistent theme in the struggle to make our program successful was finding the right balance between the investment of our sponsors and the expectations of our students. There are many factors that contribute to this “balance”.

Overall curriculum

Matching the goals of the capstone experience to the content and trajectory of the entire curriculum ensures a better chance of success for our students. Set the bar too low, and the experience is not authentic. Set the bar too high, and the students struggle to succeed. Recruit projects with the wrong balance of technical challenges, design, and humanities, and our students will struggle. Match the project to the curricular goals of the college, and the students will flourish.

Time in the academic schedule

Originally conceived, the program was 16 credit hours, essentially half the academic workload for a year. This was scaled back to 8 credit hours before launch. Balancing the time the students put into the project versus the expectations of the corporate partner is critical to success.

Team resources

Giving the students an appropriate budget and physical space to complete hard engineering work is critical to the students achieving success. While how much money and space a particular project needs varies significantly, ensuring that the program generates appropriate value for the sponsor is critical to sponsor happiness and sustainability of the program. It is important to recognize, however, that if simply measured against “consultant” efficiency, student teams will

not compare favorably. Providing value goes beyond the deliverables generated by the students to encompass philanthropy, recruiting opportunities, and investment in an academic mission.

Faculty advising

Faculty must be engaged in the process and the outcome of the program, and given schedule space to dedicate time to the capstone as any other course. Without a dedicated and passionate adviser, the experience is not as rewarding for the students, and the work suffers as a result.

Involved faculty can also play a large role in sustainability—with successful projects, sponsors may be more willing to partner with individual faculty long-term, easing the burden of finding sponsors.

The sum of these factors serves as a coarse indicator of how one might set the fee for corporate participation in the program. The more the college dedicates to the endeavor, the higher the valuation becomes. While it is not a simple task to balance this system, keeping it in balance can ensure a higher chance of sponsor satisfaction and program sustainability.

Future directions

In developing the SCOPE Program, the intent was to create an industry-sponsored capstone design program that would provide meaningful educational and professional experiences for our students, while providing enough value to the industry sponsors so that the program would be sustainable. In each of the first three years, we have brought in enough funded projects for all the fourth-year students, and our program can be considered a resounding success if observed day-to-day. Much of this success is due to the tireless dedication of the current capstone director.

Implementing a capstone course can give students a truly unique experience that can solidify their engineering education and propel them into the next stage of their careers. The costs to the college are as high as the rewards. Sustainability of the program is probably the biggest challenge we face going forward. We have started to recognize that while a dedicated individual can be primarily responsible for the success in recruiting sponsors, more needs to be done to set a positive track record that will help us continue to recruit sponsors in the future. We remain cautiously optimistic that the continued short-term successes of the program will make sponsor recruiting easier and more sustainable.

Bibliography

1. J. R. Phillips and M. M. Gilkeson, "Reflections on a Clinical Approach to Engineering Design," *Proceedings of The International ASME Conference on Design Theory and Methodology*, Miami, FL, September 1991.
2. A. Bright and J. R. Phillips, "The Harvey Mudd Engineering Clinic: Past, Present, and Future," *Journal of Engineering Education*, 88 (2), 189–194, April 1999
3. A. Bright, "Student, Faculty and Liaison Roles in the Engineering Clinic Program at Harvey Mudd College," *Proceedings of the 26th Fundamentals in Education Conference*, Salt Lake City, UT, November 1996.
4. M. Moody, "Capstone Project Issues", *internal memorandum*, June 6, 2004, revised June 15, 2004.

PART I

RECONFIGURABLE COMPUTING HARDWARE

At a fundamental level, reconfigurable computing is the process of best exploiting the potential of reconfigurable hardware. Although a complete system must include compilation software and high-performance applications, the best place to begin to understand reconfigurable computing is at the chip level, as it is the abilities and limitations of chips that crucially influence all of a system’s steps. However, the reverse is true as well—reconfigurable devices are designed primarily as a target for the applications that will be developed, and a chip that does not efficiently support important applications, or that cannot be effectively targeted by automatic design mapping flows, will not be successful.

Reconfigurable computing has been driven largely by the development of commodity field-programmable gate arrays (FPGAs). Standard FPGAs are somewhat of a mixed blessing for this field. On the one hand, they represent a source of commodity parts, offering cheap and fast programmable silicon on some of the most advanced fabrication processes available anywhere. On the other hand, they are not optimized for reconfigurable computing for the simple reason that the vast majority of FPGA customers use them as cheap, low-quality ASICs with rapid time to market. Thus, these devices are never quite what the reconfigurable computing user might want, but they are close enough. Chapter 1 covers commercial FPGA architectures in depth, providing an overview of the underlying technology for virtually all generally available reconfigurable computing systems.

Because FPGAs are not optimized toward reconfigurable computing, there have been many attempts to build better silicon devices for this community. Chapter 2 details many of them. The focus of the new architectures might be the inclusion of larger functional blocks to speed up important computations, tight connectivity to a host processor to set up a coprocessing model, fast reconfiguration features to reduce the time to change configurations, or other concepts. However, as of now no such system is commercially viable, largely because

- The demand for reconfigurable computing chips is much smaller than that for the FPGA community as a whole, reducing economies of scale.
- FPGA manufacturers have access to cutting-edge fabrication processes, while reconfigurable computing chips typically are one to two process generations behind.

For these reasons, a reconfigurable computing chip is at a significant cost, performance, and electrical power-consumption disadvantage compared to a commodity FPGA. Thus, the architectural advantages of a reconfigurable computing-specific device must be huge to make up for the problems of less economies of scale and fabrication process lag. It seems likely that eventually a company with a reconfigurable computing-specific chip will be successful; however, so far there appears to have been only failures.

Although programmable chips are important, most reconfigurable computing users need more. A real system generally requires large memories, input/output (I/O) ports to hook to various data streams, microprocessors or microprocessor interfaces to coordinate operation, and mechanisms for configuring and reconfiguring the device. Chapter 3 considers such complete systems, chronicling the development of reconfigurable computing boards.

Chapters 1 through 3 present a good overview of most reconfigurable systems hardware, but one topic requires special consideration: the reconfiguration subsystems within devices. In the first FPGAs, configuration data was loaded slowly and sequentially, configuring the entire chip for a given computation. For glue logic and application-specific integrated circuit (ASIC) replacement, this was sufficient because FPGAs needed to be configured only once, at power-up; however, in many situations the device may need to be reconfigured more often. In the extreme, a single computation might be broken into multiple configurations, with the FPGA loading new configurations during the normal execution of that circuit. In this case, the speed of reconfiguration is important. Chapter 4 focuses on the configuration memory subsystems within an FPGA, considering the challenges of fast reconfiguration and showing some ways to greatly improve reconfiguration speed.

CHAPTER 1

DEVICE ARCHITECTURE

Mark L. Chang
Franklin W. Olin College of Engineering

The best race car drivers understand how their cars work. The best architects know how carpenters, bricklayers, and electricians do their jobs. And the best programmers know how the hardware they are programming does computation. Knowing how your device works, “down to the metal,” is essential for efficient utilization of available resources.

In this chapter, we take a look inside the package to discover the basic hardware elements that make up a typical Field-Programmable Gate Array (FPGA). We’ll talk about how computation happens in an FPGA—from the blocks that do the computation to the interconnect that shuttles data from one place to another. We’ll talk about how these building blocks fit together in terms of FPGA architecture. And, of course, because programmability (as well as reprogrammability) is part of what makes an FPGA so useful, we’ll spend some time on that, too. Finally, we’ll take an in-depth look at the architectures of some commercially available FPGAs in Section 1.5, Case Studies.

We won’t be covering many of the research architectures from universities and industry—we’ll save that for later. We also won’t be talking much about how you successfully program these things to make them useful parts of a computational platform. That, too, is later in the book.

What you *will* learn is what’s “under the hood” of a typical commercial FPGA so that you will become more comfortable using it as a platform for solving problems and performing computations. The first step in our journey starts with how computation in an FPGA is done.

1.1 LOGIC—THE COMPUTATIONAL FABRIC

Think of your typical desktop computer. Inside the case, among other things, are storage and communication devices (hard drives and network cards), memory, and, of course, the central processing unit, or CPU, where most of the computation happens. The FPGA plays a similar role in a reconfigurable computing platform, but we’re going to break it down.

In very general terms, there are only two types of resources in an FPGA: *logic* and *interconnect*. Logic is where we do things like arithmetic, $1+1=2$, and logical functions, `if (ready) x=1 else x=0`. Interconnect is how we get data (like the

results of the previous computations) from one node of computation to another. Let's focus on logic first.

1.1.1 Logic Elements

From your digital logic and computer architecture background, you know that any computation can be represented as a Boolean equation (and in some cases as a Boolean equation where inputs are dependent on past results—don't worry, FPGAs can hold state, too). In turn, any Boolean equation can be expressed as a truth table. From these humble beginnings, we can build complex structures that can do arithmetic, such as adders and multipliers, as well as decision-making structures that can evaluate conditional statements, such as the classic if-then-else. Combining these, we can describe elaborate algorithms *simply by using truth tables*.

From this basic observation of digital logic, we see the truth table as the computational heart of the FPGA. More specifically, one hardware element that can easily implement a truth table is the lookup table, or LUT. From a circuit implementation perspective, a LUT can be formed simply from an $N: 1$ (N -to-one) multiplexer and an N -bit memory. From the perspective of our previous discussion, a LUT simply enumerates a truth table. Therefore, using LUTs gives an FPGA the generality to implement arbitrary digital logic. Figure 1.1 shows a typical N -input lookup table that we might find in today's FPGAs. In fact, almost all commercial FPGAs have settled on the LUT as their basic building block.

The LUT can compute any function of N inputs by simply programming the lookup table with the truth table of the function we want to implement. As shown in the figure, if we wanted to implement a 3-input exclusive-or (XOR) function with our 3-input LUT (often referred to as a 3-LUT), we would assign values to the lookup table memory such that the pattern of select bits chooses the correct row's "answer." Thus, every "row" would yield a result of 0 except in the four cases where the XOR of the three select lines yields 1.

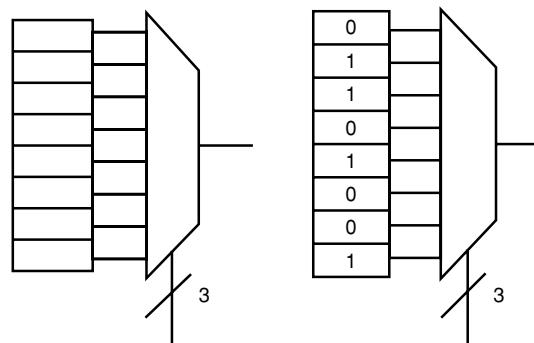


FIGURE 1.1 ■ A 3-LUT schematic (a) and the corresponding 3-LUT symbol and truth table (b) for a logical XOR.

Of course, more complicated functions, and functions of a larger number of inputs, can be implemented by aggregating several lookup tables together. For example, one can organize a single 3-LUT into an 8×1 ROM—and if the values of the lookup table are reprogrammable—an 8×1 RAM. But the basic building block, the lookup table, remains the same.

Although the LUT has more or less been chosen as the smallest computational unit in commercially available FPGAs, the size of the lookup table in each logic block has been widely investigated [1]. On the one hand, larger lookup tables would allow for more complex logic to be performed per logic block, thus reducing the wiring delay between blocks as fewer blocks would be needed. However, the penalty paid would be slower LUTs, because of the requirement of larger multiplexers, and an increased chance of waste if not all of the functionality of the larger LUTs were to be used. On the other hand, smaller lookup tables may require a design to consume a larger number of logic blocks, thus increasing wiring delay between blocks while reducing per-logic-block delay.

Current empirical studies have shown that the 4-LUT structure makes the best tradeoff between area and delay for a wide range of benchmark circuits. Of course, as FPGA computing evolves into wider arenas, this result may need to be revisited. In fact, as of this writing Xilinx has released the Virtex-5 SRAM-based FPGA with a 6-LUT architecture.

The question of the number of LUTs per logic block has also been investigated [2], with empirical evidence suggesting that grouping more than one 4-LUT into a single logic block may improve area and delay. Many current commercial FPGAs incorporate a number of 4-LUTs into each logic block to take advantage of this observation.

Investigations into both LUT size and number of LUTs per block begin to address the larger question of computational *granularity* in an FPGA. On one end of the spectrum, the rather simple structure of a small lookup table (e.g., 2-LUT) represents *fine-grained* computational capability. Toward the other end, *coarse-grained*, one can envision larger computational blocks, such as full 8-bit arithmetic logic units (ALUs), more typical of CPUs. As in the case of lookup table sizing, finer-grained blocks may be more adept at bit-level manipulations and arithmetic, but require combining several to implement larger pieces of logic. Contrast that with coarser-grained blocks, which may be more optimal for datapath-oriented computations that work with standard “word” sizes (8-/16-/32-bit) but are wasteful when implementing very simple logical operations. Current industry practice has been to strike a balance in granularity by using rather fine-grained 4-LUT architectures and augmenting them with coarser-grained heterogeneous elements, such as multipliers, as described in the Extended Logic Elements section later in this chapter.

Now that we have chosen the logic block, we must ask ourselves if this is sufficient to implement all of the functionality we want in our FPGA. Indeed, it is not. With just LUTs, there is no way for an FPGA to maintain any sense of state, and therefore we are prohibited from implementing any form of sequential, or state-holding, logic. To remedy this situation, we will add a simple single-bit storage element in our base logic block in the form of a D flip-flop.

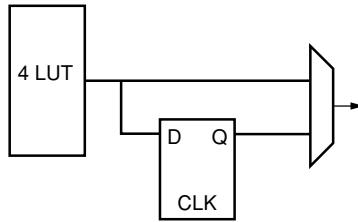


FIGURE 1.2 ■ A simple lookup table logic block.

Now our logic block looks something like Figure 1.2. The output multiplexer selects a result either from the function generated by the lookup table or from the stored bit in the D flip-flop. In reality, this logic block bears a very close resemblance to those in some commercial FPGAs.

1.1.2 Programmability

Looking at our logic block in Figure 1.2, it is a simple task to identify all the programmable points. These include the contents of the 4-LUT, the select signal for the output multiplexer, and the initial state of the D flip-flop. Most current commercial FPGAs use volatile static-RAM (SRAM) bits connected to configuration points to configure the FPGA. Thus, simply writing a value to each configuration bit sets the configuration of the entire FPGA.

In our logic block, the 4-LUT would be made up of 16 SRAM bits, one per output; the multiplexer would use a single SRAM bit, and the D flip-flop initialization value could also be held in a single SRAM bit. How these SRAM bits are initialized in the context of the rest of the FPGA will be the subject of later sections.

1.2 THE ARRAY AND INTERCONNECT

With the LUT and D flip-flop, we begin to define what is commonly known as the *logic block*, or *function block*, of an FPGA. Now that we have an understanding of how computation is performed in an FPGA at the single logic block level, we turn our focus to how these computation blocks can be tiled and connected together to form the fabric that is our FPGA.

Current popular FPGAs implement what is often called *island-style* architecture. As shown in Figure 1.3, this design has logic blocks tiled in a two-dimensional array and interconnected in some fashion. The logic blocks form the islands and “float” in a sea of interconnect.

With this array architecture, computations are performed spatially in the fabric of the FPGA. Large computations are broken into 4-LUT-sized pieces and mapped into physical logic blocks in the array. The interconnect is configured to route signals between logic blocks appropriately. With enough logic blocks, we can make our FPGAs perform any kind of computation we desire.

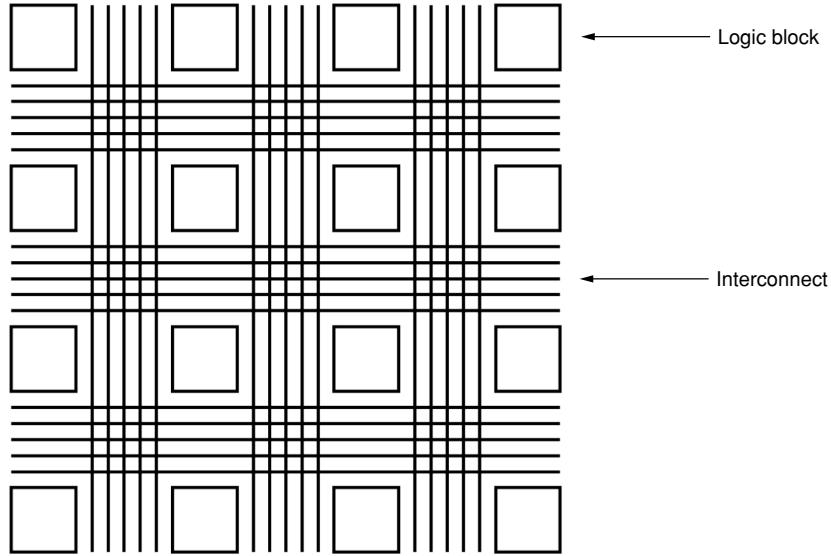


FIGURE 1.3 ■ The island-style FPGA architecture. The interconnect shown here is not representative of structures actually used.

1.2.1 Interconnect Structures

Figure 1.3 does not tell the whole story. The interconnect structure shown there is not representative of any structures used in actual FPGAs, but is more of a cartoon placeholder. This section introduces the interconnect structures present in many of today's FPGAs, first by considering a small area of interconnection and then expanding out to understand the need for different styles of interconnect. We start with the simplest case of nearest-neighbor communication.

Nearest neighbor

Nearest-neighbor communication is as simple as it sounds. Looking at a 2×2 array of logic blocks in Figure 1.4, one can see that the only needs in this neighborhood are input and output connections in each direction: north, south, east, and west. This allows each logic block to communicate directly with each of its immediate neighbors.

Figure 1.4 is an example of one of the simplest routing architectures possible. While it may seem nearly degenerate, it has been used in some (now obsolete) commercial FPGAs. Of course, although this is a simple solution, this structure suffers from severe delay and connectivity issues. Imagine, instead of a 2×2 array, a 1024×1024 array. With only nearest-neighbor connectivity, the delay scales linearly with distance because the signal must go through many cells (and many switches) to reach its final destination.

From a connectivity standpoint, without the ability to bypass logic blocks in the routing structure, all routes that are more than a single hop away require

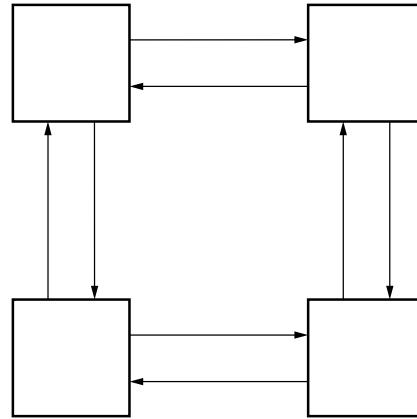


FIGURE 1.4 ■ Nearest-neighbor connectivity.

traversing a logic block. With only one bidirectional pair in each direction, this limits the number of logic block signals that may cross. Signals that are passing through must not overlap signals that are being actively consumed and produced.

Because of these limitations, the nearest-neighbor structure is rarely used *exclusively*, but it is almost always available in current FPGAs, often augmented with some of the techniques that follow.

Segmented

As we add complexity, we begin to move away from the pure logic block architecture that we've developed thus far. Most current FPGA architectures look less like Figure 1.3 and more like Figure 1.5.

In Figure 1.5 we introduce the connection block and the switch box. Here the routing structure is more generic and mesh-like. The logic block accesses nearby communication resources through the connection block, which connects logic block input and output terminals to routing resources through programmable switches, or multiplexers. The connection block (detailed in Figure 1.6) allows logic block inputs and outputs to be assigned to arbitrary horizontal and vertical tracks, increasing routing flexibility.

The switch block appears where horizontal and vertical routing tracks converge as shown in Figure 1.7. In the most general sense, it is simply a matrix of programmable switches that allow a signal on a track to connect to another track. Depending on the design of the switch block, this connection could be, for example, to turn the corner in either direction or to continue straight. The design of switch blocks is an entire area of research by itself and has produced many varied designs that exhibit varying degrees of connectivity and efficiency [3, 4, 5]. A detailed discussion of this research is beyond the scope of this book.

With this slightly modified architecture, the concept of a segmented interconnect becomes more clear. Nearest-neighbor routing can still be accomplished, albeit through a pair of connect blocks and a switch block. However, for

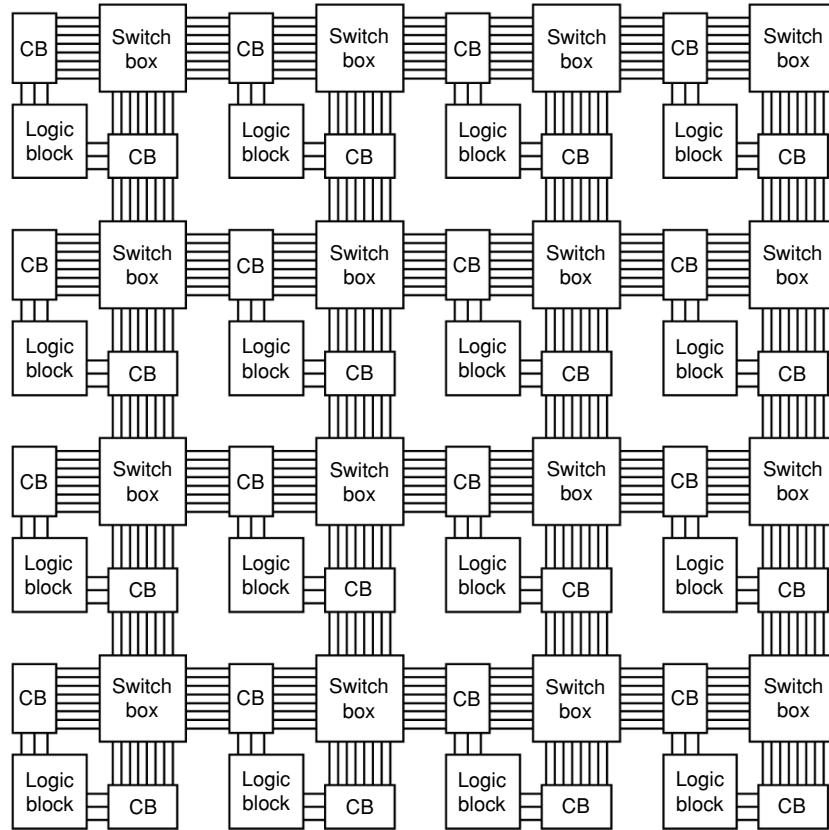


FIGURE 1.5 ■ An island-style architecture with connect blocks and switch boxes to support more complex routing structures. (The difference in relative sizes of the blocks is for visual differentiation.)

signals that need to travel longer distances, individual segments can be switched together in a switch block to connect distant logic blocks together. Think of it as a way as to emulate long signal paths that can span arbitrary distances. The result is a long wire that actually comprises shorter “segments.”

This interconnect architecture alone does not radically improve on the delay characteristics of the nearest-neighbor interconnect structure. However, the introduction of connection blocks and switch boxes separates the interconnect from the logic, allowing long-distance routing to be accomplished without consuming logic block resources.

To improve on our structure, we introduce longer-length wires. For instance, consider a wire that spans one logic block as being of length-1 (L1). In some segmented routing architectures, longer wires may be present to allow signals to travel greater distances more efficiently. These segments may be length-4

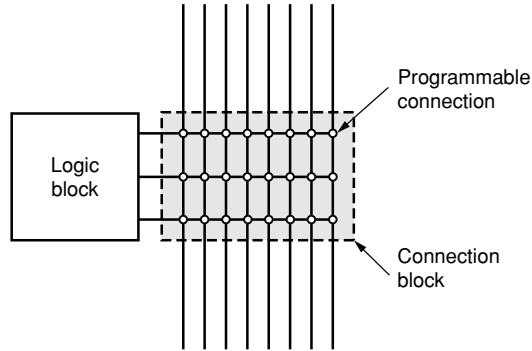


FIGURE 1.6 ■ Detail of a connection block.

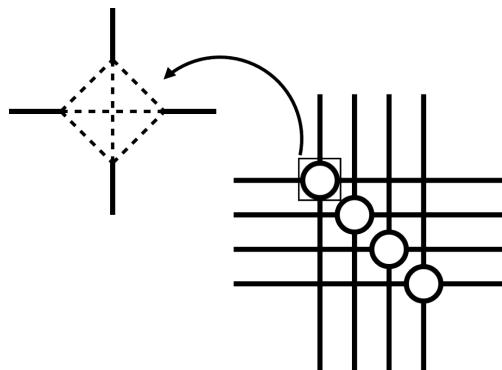


FIGURE 1.7 ■ An example of a common switch block architecture.

(L4), L8, and so on. The switch blocks (and perhaps more embedded switches) become points where signals can switch from shorter to longer segments. This feature allows signal delay to be less than $O(N)$ when covering a distance of N logic blocks by reducing the number of intermediate switches in the signal path.

Figure 1.8 illustrates augmenting the single-segment interconnect with two additional lengths: direct-connect between logic blocks and length-2 (L2) lines. The direct-connect lines leave general routing resources free for other uses, and L2 lines allow signals to travel longer distances for roughly the same amount of switch delay. This interconnect architecture closely matches that of the Xilinx XC4000-series of commercial FPGAs.

Hierarchical

A slightly different approach to reducing the delay of long wires uses a hierarchical approach. Consider the structure in Figure 1.9. At the lowest level of hierarchy, 2×2 arrays of logic blocks are grouped together as a single cluster.

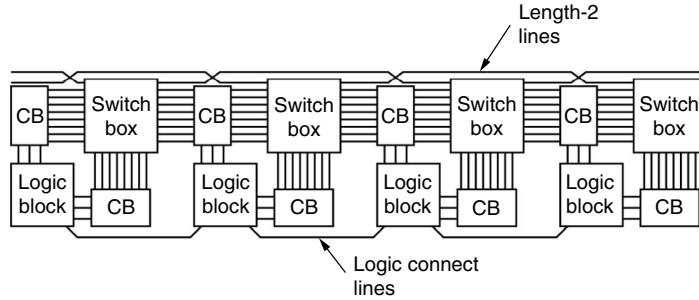


FIGURE 1.8 ■ Local (direct) connections and length-2 connections augmenting a switched interconnect.

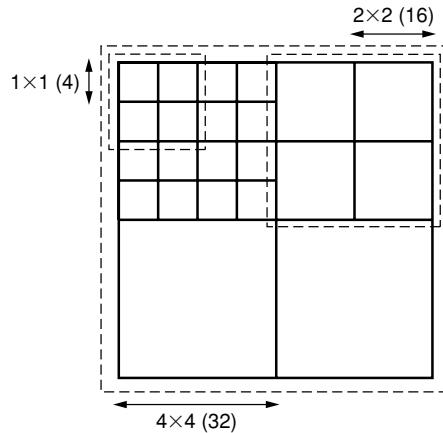


FIGURE 1.9 ■ Hierarchical routing used by long wires to connect clusters of logic blocks.

Within this block, local, nearest-neighbor routing is all that is available. In turn, a 2×2 cluster of these clusters is formed that encompasses 16 logic blocks. At this level of hierarchy, longer wires at the boundary of the smaller, 2×2 clusters, connect each cluster of four logic blocks to the other clusters in the higher-level grouping. This is repeated in higher levels of hierarchy, with larger clusters and longer wires.

The pattern of interconnect just described exploits the assumption that a well-designed (and well-placed) circuit has mostly local connections and only a limited number of connections that need to travel long distances. By providing fewer resources at the higher levels of hierarchy, this interconnect architecture remains area-efficient while preserving some long-length wires to minimize the delay of signals that need to cross large distances.

As in the segmented architecture, the connection points that connect one level of routing hierarchy to another can be anywhere in the interconnect structure. New points in the existing switch blocks may be created, or completely independent

switching sites elsewhere in the interconnect can be created specifically for the purpose of moving between hierarchy levels.

1.2.2 Programmability

As with the logic blocks in a typical commercial FPGA, each switch point in the interconnect structure is programmable. Within the connection block, programmable multiplexers select which routing track each logic block's input and output terminals map to; in the switch block, the junction between vertical and horizontal routing tracks is switched through a programmable switch; and, finally, switching between routing tracks of different segment lengths or hierarchy levels is accomplished, again through programmable switches.

For all of these programmable points, as in the logic block, modern FPGAs use SRAM bits to hold the user-defined configuration values. More discussion of these configuration bits comes later in this chapter.

1.2.3 Summary

Programmable routing resources are the natural counterpart to the logic resources in an FPGA. Where the logic performs the arithmetic and logical computations, the interconnection fabric takes the results output from logic blocks and routes them as inputs to other logic blocks. By tiling logic blocks together and connecting them through a series of programmable interconnects as described here, an FPGA can implement complex digital circuits. The true nature of *spatial computing* is realized by spreading the computation across the physical area of an FPGA.

Today's commercial FPGAs typically use bits of each of these interconnect architectures to provide a smooth and flexible set of routing resources. In actual implementation, segmentation and hierarchy may not always exhibit the logarithmic scaling seen in our examples. In modern FPGAs, the silicon area consumed by interconnect greatly dominates the area dedicated to logic. Anecdotally, 90 percent of the available silicon is interconnect whereas only 10 percent is logic. With this imbalance, it is clear that interconnect architecture is increasingly important, especially from a delay perspective.

1.3 EXTENDING LOGIC

With a logic block like the one shown in Figure 1.2, tiled in a two-dimensional array with a supporting interconnect structure, we can implement any combinational and sequential logic. Our only constraint is area in terms of the number of available logic blocks. While this is comprehensive, it is far from optimal. In this section, we investigate how FPGA architects have augmented this simple design to increase performance.

1.3.1 Extended Logic Elements

Modern FPGA interconnect architectures have matured to include much more than simple nearest-neighbor connectivity to give increased performance for

common applications. Likewise, the basic logic elements have been augmented to increase performance for common operations such as arithmetic functions and data storage.

Fast carry chain

One fundamental operation that the FPGA is likely to perform is an addition. From the basic logic block, it is apparent that we can implement a full-adder structure with two logic blocks given at least a 3-LUT. One logic block is configured to compute the sum, and one is configured to compute the carry. Cascading N pairs of logic blocks together will yield a simple N -bit full adder.

As you may already know from digital arithmetic, the critical path of this type of addition comes not from the computation of the sum bits but rather from the rippling of the carry signal from lower-order bits to higher-order bits (see Figure 1.10). This path starts with the low-order primary inputs, goes through the logic block, out into the interconnect, into the adjacent logic block, and so on. Delay is accumulated at every switch point along the way (including the interconnect structure, which we have not yet discussed).

One clever way to increase speed is to shortcut the carry chain between adjacent logic blocks. We can accomplish this by providing a dedicated, minimally switched path from the output of the logic block computing the carry signal to the adjacent higher-order logic block pair. This carry chain will not need to be routed on the general interconnect network. By adding a minimal amount of overhead (wires), we dramatically speed up the addition operation.

This feature does force some constraints on the spatial layout of a multibit addition. If, for instance, the dedicated fast carry chain only goes vertically, along columns of logic blocks, all additions must be oriented along the carry chain to take advantage of this dedicated resource. Additionally, to save switching area the dedicated carry chain may not be a bidirectional path, which further restricts the physical layout to be oriented vertically and dictates the order of the bits relative to one another. The fast carry chain of the Xilinx XC4000E is shown in Figure 1.11. Note that the bidirectional fast carry chain wires are arranged along the columns while the horizontal lines are unidirectional. This allows large adder structures to be placed in a zig-zag pattern in the array and still make use of the dedicated carry chain interconnect.

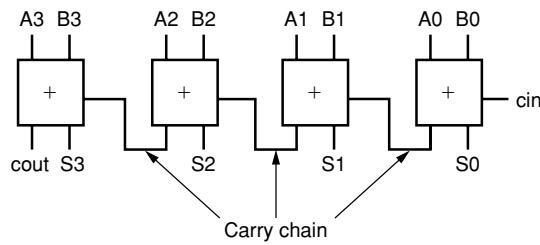


FIGURE 1.10 ■ A simple 4-bit full adder.

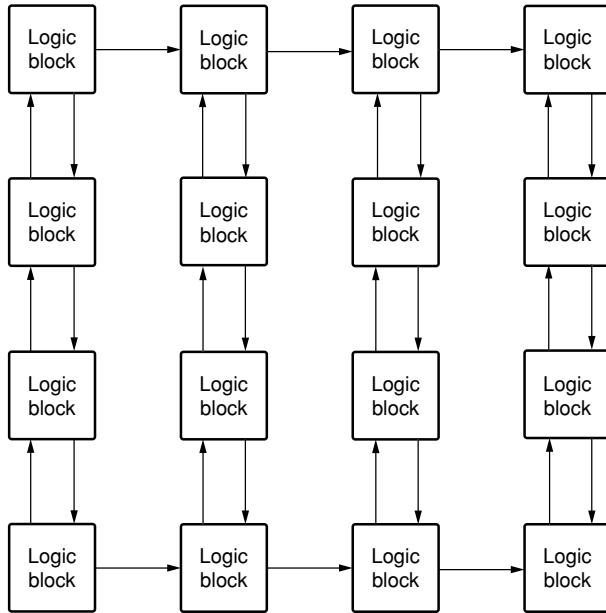


FIGURE 1.11 ■ The Xilinx XC4000E fast carry chain. (Source: Adapted from [6], Figure 11, p. 6-18.)

The fast carry chain logic is now commonplace in commercial FPGAs, with the physical design constraints at this point completely abstracted away by the tools provided by manufacturers. The success of this optimization relies on the toolset's ability to identify additions in the designer's circuit description and then use the dedicated logic. With today's tools, this kind of optimization is nearly transparent to the end user.

Multipliers

If addition is commonplace in algorithms, multiplication is certainly not rare. Several implementations are available if we wish to use general logic block resources to build our multipliers. From the area-efficient iterative shift-accumulate method to the area-consumptive array multiplier, we can use logic blocks to either compute additions or store intermediate values. While we can certainly implement a multiplication, we can do so only with a large delay penalty, or a large logic block footprint, depending on our implementation. In essence, our logic blocks *aren't very efficient* at performing a multiplication.

Instead of doing it with logic blocks, why not build *real* multipliers outside, but still connected to, the general FPGA fabric? Then, instead of inefficiently using simple LUTs to implement a multiply, we can route the values that need to be multiplied to actual multipliers implemented in silicon. How does this save space and time? Recall that FPGAs trade speed and power for configurability when compared to their ASIC (application-specific integrated circuit) counterparts. If you asked a VLSI designer to implement a fast multiplier out of transistors

any way she wanted, it would take up far less silicon area, be much faster, and consume less power than we could ever manage using LUTs.

The result is that, for a small price in silicon area, we can offload the otherwise area-prohibitive multiplication onto dedicated hardware that does it much better. Of course, just like fast carry chains, multipliers impose important design considerations and physical constraints, but we add one more option for computation to our palette of operations. It is now just a matter of good design and good tools to make an efficient design. Like fast carry chains, multipliers are commonplace in modern FPGAs.

RAM

Another area that has seen some customization beyond the general FPGA fabric is in the area of on-chip data storage. While logic blocks can individually provide a few bits of storage via the lookup table structure—and, in aggregate, many bits—they are far from an efficient use of FPGA resources. Like the fast carry chain and the “hard” multiplier, FPGA architectures have given their users generous amounts of on-chip RAM that can be accessed from the general FPGA fabric.

Static RAM cells are extremely small and, when physically distributed throughout the FPGA, can be very useful for many algorithms. By grouping many static RAM cells into banks of memory, designers can implement large ROMs for extremely fast lookup table computations and constant-coefficient operations, and large RAMs for buffering, queuing, and basic scratch use—all with the convenience of a simple clocking strategy and the speed gained by avoiding off-chip communication to an external memory. Today’s FPGAs provide anywhere from kilobits to megabits of dedicated RAM.

Processor blocks

Tying all these blocks together, most commercial FPGAs now offer entire dedicated processors in the FPGA, sometimes even more than one. In a general sense, FPGAs are extremely efficient at implementing raw computational pipelines, exploiting nonstandard bit widths, and providing data and functional parallelism. The inclusion of dedicated CPUs recognizes the fact that algorithm flows that are very procedural and contain a high degree of branching do not lend themselves readily to acceleration using FPGAs.

Entire CPU blocks can now be found in high-end FPGA devices. At the time of this writing, these CPUs are on the scale of 300MHz PowerPC devices, complete, without floating-point units. They are capable of running an entire embedded operating system, and some are even able to reprogram the FPGA fabric around them.

The CPU cores are not nearly as easily exploited as the carry chains, multipliers, and on-chip RAMs, but they represent a distinct shift toward making FPGAs more “platform”-oriented. With a traditional CPU on board (and perhaps up to four), a single FPGA can serve nearly as an entire “system-on-a-chip”—the holy grail of system integrators and embedded device manufacturers. With standard programming languages and toolchains available to developers, an entire project might indeed be implemented with a single-chip solution, dramatically reducing cost and time to market.

1.3.2 Summary

In the end, modern commercially available FPGAs provide a rich variety of basic, and not so basic, computational building blocks. With much more than simple lookup tables, the task for the FPGA architect is to decide in what proportion to provide these resources and how they should be connected. The task of the hardware designer is then to fully understand the capabilities of the target FPGAs to create designs that exploit their potential.

The common thread among these extended logical elements is that they provide critical functionality that cannot be implemented very efficiently in the general FPGA fabric. As much as the technology drives FPGA architectures, applications provide a much needed push. If multipliers were rare, it wouldn't make sense to waste silicon space on a "hard" multiplier. As FPGAs become more heterogeneous in nature, and become useful computational platforms in new application domains, we can expect to see even more varied blocks in the next generation of devices.

1.4 CONFIGURATION

One of the defining features of an FPGA is its ability to act as "blank hardware" for the end user. Providing more performance than pure software implementations on general-purpose processors, and more flexibility than a fixed-function ASIC solution, relies on the FPGA being a reconfigurable device. In this section, we will discuss the different approaches and technologies used to provide programmability in an FPGA.

Each configurable element in an FPGA requires 1 bit of storage to maintain a user-defined configuration. For a simple LUT-based FPGA, these programmable locations generally include the contents of the logic block and the connectivity of the routing fabric. Configuration of the FPGA is accomplished through programming the storage bits connected to these programmable locations according to user definitions. For the lookup tables, this translates into filling it with 1s and 0s. For the routing fabric, programming enables and disables switches along wiring paths.

The configuration can be thought of as a flat binary file whose contents map, bit for bit, to the programmable bits in the FPGA. This *bitstream* is generated by the vendor-specific tools after a hardware design is finalized. While its exact format is generally not publicly known, the larger the FPGA, the larger bitstream becomes.

Of course, there are many known methods for storing a single bit of binary information. We discuss the most popular methods used for FPGAs next.

1.4.1 SRAM

As discussed in previous sections, the most widely used method for storing configuration information in commercially available FPGAs is volatile static RAM, or SRAM. This method has been made popular because it provides fast and infinite reconfiguration in a well-known technology.

Drawbacks to SRAM come in the form of power consumption and data volatility. Compared to the other technologies described in this section, the SRAM cell is large (6–12 transistors) and dissipates significant static power because of leakage current. Another significant drawback is that SRAM does not maintain its contents without power, which means that at power-up the FPGA is not configured and must be programmed using off-chip logic and storage. This can be accomplished with a nonvolatile memory store to hold the configuration and a micro-controller to perform the programming procedure. While this may seem to be a trivial task, it adds to the component count and complexity of a design and prevents the SRAM-based FPGA from being a truly single-chip solution.

1.4.2 Flash Memory

Although less popular than SRAM, several families of devices use Flash memory to hold configuration information. Flash memory is different from SRAM in that it is nonvolatile and can only be written a finite number of times.

The nonvolatility of Flash memory means that the data written to it remains when power is removed. In contrast with SRAM-based FPGAs, the FPGA remains configured with user-defined logic even through power cycles and does not require extra storage or hardware to program at boot-up. In essence, a Flash-based FPGA can be ready immediately.

A Flash memory cell can also be made with fewer transistors compared to an SRAM cell. This design can yield lower static power consumption as there are fewer transistors to contribute to leakage current.

Drawbacks to using Flash memory to store FPGA configuration information stem from the techniques necessary to write to it. As mentioned, Flash memory has a limited write cycle lifetime and often has slower write speeds than SRAM. The number of write cycles varies by technology, but is typically hundreds of thousands to millions. Additionally, most Flash write techniques require higher voltages compared to normal circuits; they require additional off-chip circuitry or structures such as charge pumps on-chip to be able to perform a Flash write.

1.4.3 Antifuse

A third approach to achieving programmability is antifuse technology. Antifuse, as its name suggests, is a metal-based link that behaves the opposite of a fuse. The antifuse link is normally open, (i.e., unconnected). A programming procedure that involves either a high-current programmer or a laser melts the link to form an electrical connection across it—in essence, creating a wire or a short-circuit between the antifuse endpoints.

Antifuse has several advantages and one clear disadvantage, which is that it is not reprogrammable. Once a link is fused, it has undergone a physical transformation that cannot be reversed. FPGAs based on this technology are generally considered one-time programmable (OTP). This severely limits their flexibility in terms of reconfigurable computing and nearly eliminates this technology for use in prototyping environments.

However, there are some distinct advantages to using antifuse in an FPGA platform. First, the antifuse link can be made very small, compared to the large multi-transistor SRAM cell, and does not require any transistors. This results in very low propagation delays across links and zero static power consumption, as there is no longer any transistor leakage current. Antifuse links are also not susceptible to high-energy radiation particles that induce errors known as single-event upsets, making them more likely candidates for space and military applications.

1.4.4 Summary

There are several well-known methods for storing user-defined configuration data in an FPGA. We have reviewed the three most common in this section. Each has its strengths and weaknesses, and all can be found in current commercial FPGA products.

Regardless of the technology used to store or convey configuration data, the idea remains the same. From vendor-specific tools a device-specific programming bitstream is created and used either to program an SRAM or Flash memory, or to describe the pattern of antifuse links to be used. In the end, the user-defined configuration is reflected in the FPGA, bringing to reality part of the vision of reconfigurable computing.

1.5 CASE STUDIES

If you've read everything thus far, the FPGA should no longer seem like a magical computational black box. In fact, you should have a good grasp of the components that make up modern commercial FPGAs and how they are put together. In this section we'll take it one step further and solidify the abstractions by taking a look at two real commercial architectures—the Altera Stratix and the Xilinx Virtex-II Pro—and linking the ideas introduced earlier in this chapter with concrete industry implementations.

Although these devices represent near-current technologies, having been introduced in 2002, they are not the latest generation of devices from their respective manufacturers. The reason for choosing them over more cutting-edge examples is in part due to the level of documentation available at the time of this writing. As is often the case, detailed architecture information is not available as soon as a product is released and may never be available depending on the manufacturer.

Finally, the devices discussed here are much more complex than we have space to describe. The myriad ways modern devices can be used to perform computation and the countless hardware and software features that allow you to create powerful and efficient designs are all part of a larger, more advanced dialogue. So if something seems particularly interesting, we encourage you to grab a copy of the device handbook(s) and dig a little deeper.

1.5.1 Altera Stratix

We begin by taking a look at the Altera Stratix FPGA. Much of the information presented here is adapted from the July 2005 edition of the *Altera Stratix Device Handbook* (available online at <http://www.altera.com>).

The Stratix is an SRAM-based island-style FPGA containing many heterogeneous computational elements. The basic logical tile is the logic array block (LAB), which consists of 10 logic elements (LEs). The LABs are tiled across the device in rows and columns with a multi-level interconnect bringing together logic, memory, and other resources. Memory is provided through TriMatrix memory structures, which consist of three memory block sizes—M512, M4K, and M-RAM—each with its own unique properties. Additional computational resources are provided in DSP blocks, which can efficiently perform multiplication and accumulation. These resources are shown in a high-level block diagram in Figure 1.12.

Logic architecture

The smallest logical block in the array is the LE, shown in Figure 1.13. The general architecture of the LE is very similar to the structure that we introduced earlier—a single 4-LUT function generator and a programmable register as a state-holding element. In the Altera LE, you can see additional components to facilitate driving the interconnect (*right* side of Figure 1.12), setting and clearing the programmable register, choosing from several programmable clocks, and propagating the carry chain.

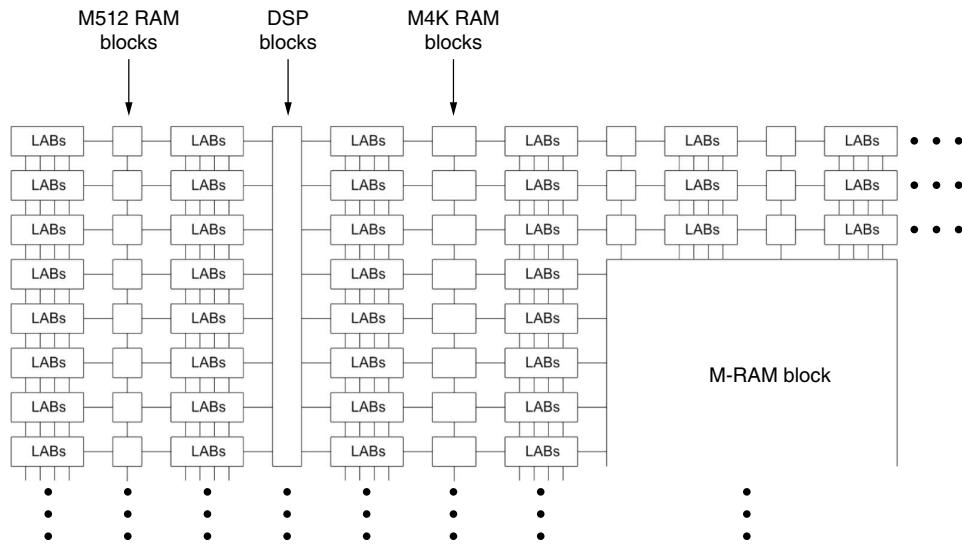


FIGURE 1.12 ■ Altera Stratix block diagram. (Source: Adapted from [7], Chapter 2, p. 2–2.)

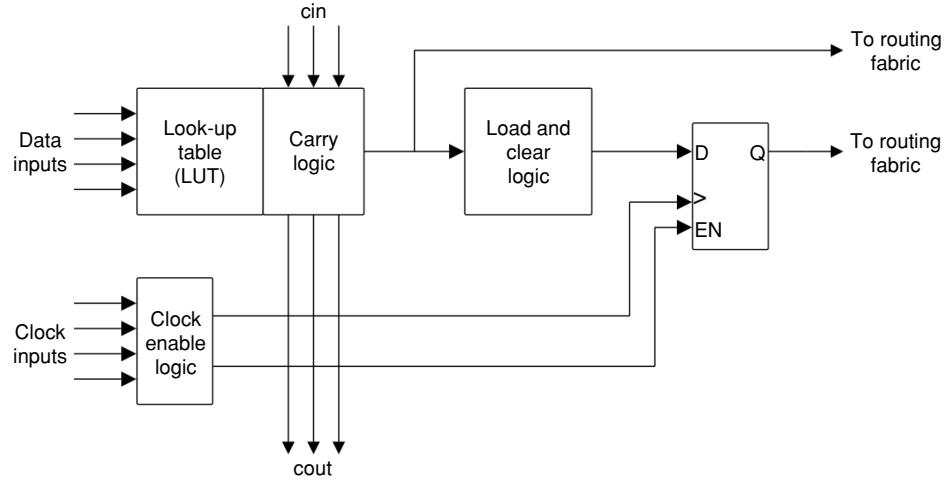


FIGURE 1.13 ■ Simplified Altera Stratix logic element. (Source: Adapted from [7], Chapter 2, p. 2–5.)

Because the LEs are simple structures that may appear tens of thousands of times in a single device, Altera groups them into LABs. The LAB is then the basic structure that is tiled into an array and connected via the routing structure. Each LAB consists of 10 LEs, all LE carry chains, LAB-wide control signals, and several local interconnection lines. In the largest device, the EP1S80, there are 101 LAB rows and 91 LAB columns, yielding a total of 79,040 LEs. This is fewer than would be expected given the number of rows and columns because of the presence of the TriMatrix memory structures and DSP blocks embedded in the array.

As shown in Figure 1.14, the LAB structure is dominated, at least conceptually, by interconnect. The local interconnect allows LEs in the same LAB to send signals to one another without using the general interconnect. Neighboring LABs, RAM blocks, and DSP blocks can also drive the local interconnect through direct links. Finally, the general interconnect (both horizontal and vertical channels) can drive the local interconnect. This high degree of connectivity is the lowest level of a rich, multi-level routing fabric.

The Stratix has three types of memory blocks—M512, M4K, and M-RAM—collectively dubbed TriMatrix memory. The largest distinction between these blocks is their size and number in a given device. Generally speaking, they can be configured in a number of ways, including single-port RAM, dual-port RAM, shift-register, FIFO, and ROM table. These memories can optionally include parity bits and have registered inputs and outputs.

The M512 RAM block is nominally organized as a 32×18 -bit memory; the M4K RAM, as a 128×36 -bit memory; and the M-RAM, as a $4K \times 144$ -bit memory. Additionally, each block can be configured for a variety of widths depending on the needs of the user. The different-sized memories throughout the array provide

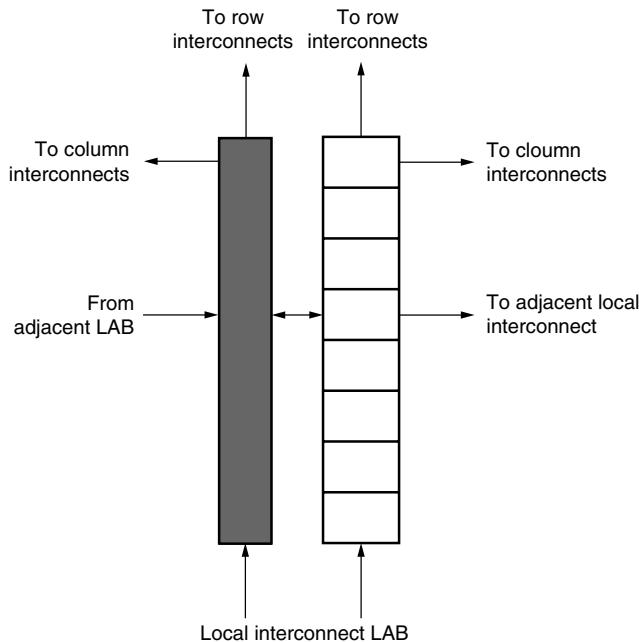


FIGURE 1.14 ■ Simplified Altera Stratix LAB structure. (Source: Adapted from [8], Chapter 2, p. 2–4.)

an efficient mapping of variable-sized memory designs to the device. In total, on the EP1S80 there are over 7 million memory bits available for use, divided into 767 M512 blocks, 364 M4K blocks, and 9 M-RAM blocks.

The final element of logic present in the Altera Stratix is the DSP block. Each device has two columns of DSP blocks that are designed to help implement DSP-type functions, such as finite-impulse response (FIR) and infinite-impulse response (IIR) filters and fast Fourier transforms (FFT), without using the general logic resources of the LEs. The common computational function required in these operations is often a multiplication and an accumulation. Each DSP block can be configured by the user to support a single 36×36 -bit multiplication, four 18×18 -bit multiplications, or eight 9×9 -bit multiplications, in addition to an optional accumulation phase. In the EP1S80, there are 22 total DSP blocks.

Routing architecture

The Altera Stratix provides a interconnect system dubbed MultiTrack that connects all the elements just discussed using routing lines of varying fixed lengths. Along the row (horizontal) dimension, the routing resources include direct connections left and right between blocks (LABs, RAMs, and DSP) and interconnects of lengths 4, 8, and 24 that traverse either 4, 8, or 24 blocks left and right, respectively. A detailed depiction of an R4 interconnect at a single

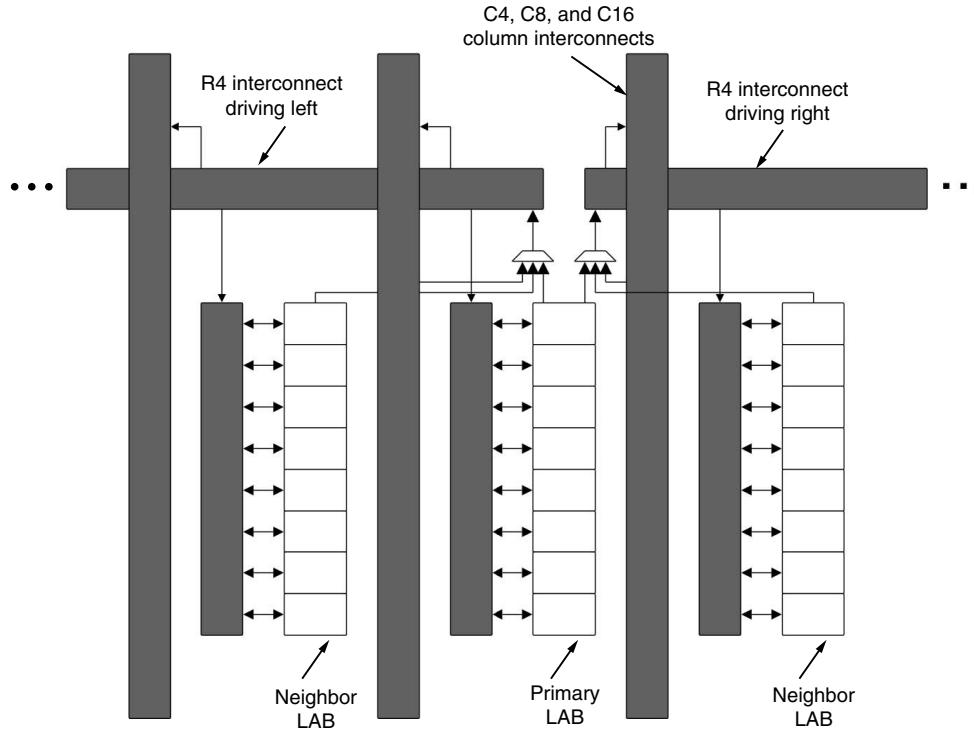


FIGURE 1.15 ■ Simplified Altera Stratix MultiTrack interconnect. (Source: Adapted from [7], Chapter 2, p. 2–14.)

LAB is shown in Figure 1.15. The R4 interconnect shown spans 4 blocks, left to right. The relative sizing of blocks in the Stratix allows the R4 interconnect to span four LABs; three LABs and one M512 RAM; two LABs and one M4K RAM; or two LABs and one DSP block, in either direction.

This structure is repeated for every LAB in the row (i.e., every LAB has its own set of dedicated R4 interconnects driving left and right). R4 interconnects can drive C4 and C16 interconnects to propagate signals vertically to different rows. They can also drive R24 interconnects to efficiently travel long distances.

The R8 interconnects are identical to the R4 interconnects except that they span eight blocks instead of four and only connect to R8 and C8 interconnects. By design, the R8 interconnect is faster than two R4 interconnects joined together. The R24 interconnect provides the fastest long-distance interconnection. It is similar to the R4 and R8 interconnects, but does not connect directly to the LAB local interconnects. Instead, it is connected to row and column interconnects at every fourth LAB and only communicates to LAB local interconnects through R4 and C4 routes. R24 interconnections connect with all interconnection routes except L8s.

In the column (vertical) dimension, the resources are very similar. They include LUT chain and register chain direct connections and interconnects of lengths 4, 8, and 16 that traverse 4, 8, or 16 blocks up and down, respectively. The LAB local interconnects found in row routing resources are mirrored through LUT chain and register chain interconnects. The LUT chain connects the combinatorial output of one LE to the fast input of the LE directly below it without consuming general routing resources. The register chain connects the register output of one LE to the register input of another LE to implement fast shift registers.

Finally, although this discussion was LAB-centric, all blocks connect to the MultiTrack row and column interconnect using a direct connection similar to the LAB local connection interfaces. These direct connection blocks also support fast direct communication to neighboring LABs.

1.5.2 Xilinx Virtex-II Pro

Launched and shipped right behind the Altera Stratix, the Xilinx Virtex-II Pro FPGA was the flagship product of Xilinx, Inc., for much of 2002 and 2003. A good deal of the information presented here is adapted from “Module 2 (Functional Description)” of the October 2005 edition of *Xilinx Virtex-II Pro™ and Virtex-II Pro X™ Platform FPGA Handbook* (available at <http://www.xilinx.com>).

The Virtex-II Pro is an SRAM-based island-style FPGA with several heterogeneous computational elements interconnected through a complex routing matrix. The basic logic tile is the configurable logic block (CLB), consisting of four *slices* and two 3-state buffers. These CLBs are tiled across the device in rows and columns with a segmented, hierarchical interconnect tying all the resources together. Dedicated memory blocks, SelectRAM+, are spread throughout the device. Additional computational resources are provided in dedicated 18×18 -bit multiplier blocks.

Logic architecture

The smallest piece of logic from the perspective of the interconnect structure is the CLB. Shown in Figure 1.16, it consists of four equivalent *slices* organized into two columns of two slices each with independent carry chains and a common shift chain. Each slice connects to the general routing fabric through a configurable switch matrix and to each other in the CLB through a fast local interconnect.

Each slice comprises primarily two 4-LUT function generators, two programmable registers for state holding, and fast carry logic. The slice also contains extra multiplexers (MUXFx and MUXF5) to allow a single slice to be configured for wide logic functions of up to eight inputs. A handful of other gates provide extra functionality in the slice, including an XOR gate to complete a 2-bit full adder in a single slice, an AND gate to improve multiplier implementations in the logic fabric, and an OR gate to facilitate implementation of sum-of-products chains.

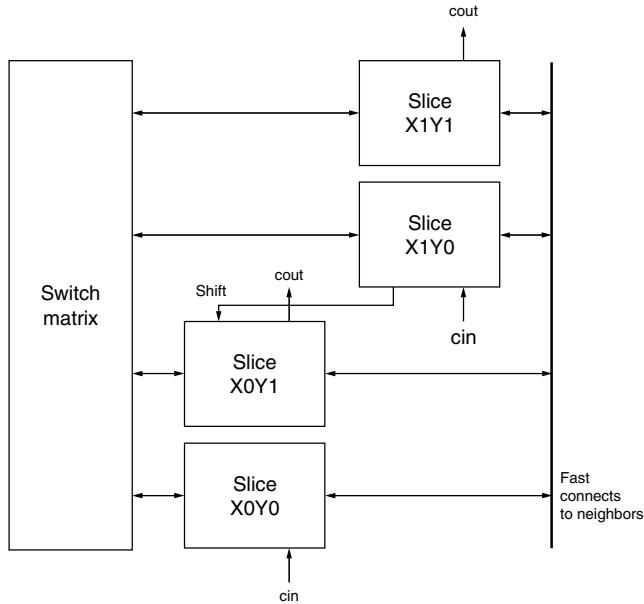


FIGURE 1.16 ■ Xilinx Virtex-II Pro configurable CLB. (Source: Adapted from [8], Figure 32, p. 35.)

In the largest Virtex-II Pro device, the XC2VP100, there are 120 rows and 94 columns of CLBs. This translates into 44,096 individual slices and 88,192 4-LUTs—comparable to the largest Stratix device. In addition to these general configurable logic resources, the Virtex-II Pro provides dedicated RAM in the form of block SelectRAM+. Organized into multiple columns throughout the device, each block SelectRAM+ provides 18Kb of independently clocked, true dual-port synchronous RAM. It supports a variety of configurations, including single- and dual-port access in various aspect ratios. In the largest device there are 444 blocks of block SelectRAM+ organized into 16 columns, yielding a total of 8,183,808 bits of memory.

Complementing the general logic resources are a number of 18×18 -bit 2's complement signed multiplier blocks. Like the DSP blocks in the Altera Stratix, these multiplier structures are designed for DSP-type operations, including FIR, IIR, FFT, and others, which often require multiply-accumulate structures. As shown in Figure 1.17, each 18×18 multiplier block is closely associated with an 18Kb block SelectRAM+. The use of the multiplier/block SelectRAM+ memory, with an accumulator implement in LUTs allows the implementation of efficient multiply-accumulate structures. Again, in the largest device, just as with block SelectRAM+, there are 16 columns yielding a total of 444 18×18 -bit multiplier blocks.

Finally, the Virtex-II Pro has one unique feature that has been carried into newer products and can also be found in competing Altera products. Embedded

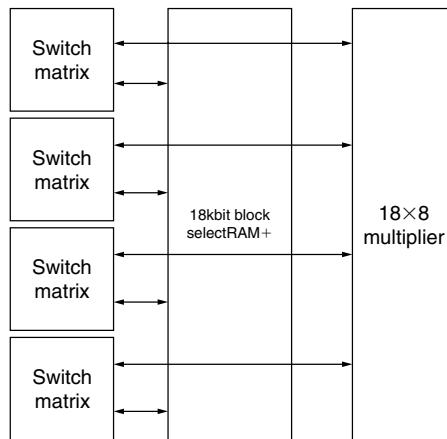


FIGURE 1.17 ■ Virtex-II Pro multiplier/block SelectRAM+ organization. (Source: Adapted from [8], Figure 53, p. 48.)

in the silicon of the FPGA, much like the multiplier and block SelectRAM+ structures, are up to four IBM PowerPC 405-D5 CPU cores. These cores can operate up to 300+ MHz and communicate with surrounding CLB fabric, block SelectRAM+, and general interconnect through dedicated interface logic. On-chip memory (OCM) controllers allow the PowerPC core to use block SelectRAM+ as small instruction and data memories if no off-chip memories are available.

The presence of a complete, standard microprocessor that has the ability to interface at a very low level with general FPGA resources allows unique, system-on-a-chip designs to be implemented with only a single FPGA device. For example, the CPU core can execute housekeeping tasks that are neither time-critical nor well suited to implementation in LUTs.

Routing architecture

The Xilinx Virtex-II Pro provides a segmented, hierarchical routing structure that connects to the heterogeneous fabric of elements through a switch matrix block. The routing resources (dubbed Active Interconnect) are physically located in horizontal and vertical routing channels between each switch matrix and look quite different from the Altera Stratix interconnect structures.

The routing resources available between any two adjacent switch matrix rows or columns are shown in Figure 1.18, with the switch matrix block shown in black. These resources include, from top to bottom, the following:

- 24 long lines that span the full height and width of the device
- 120 hex lines that route to every third or sixth block away in all four directions
- 40 double lines that route to every first or second block away in all four directions

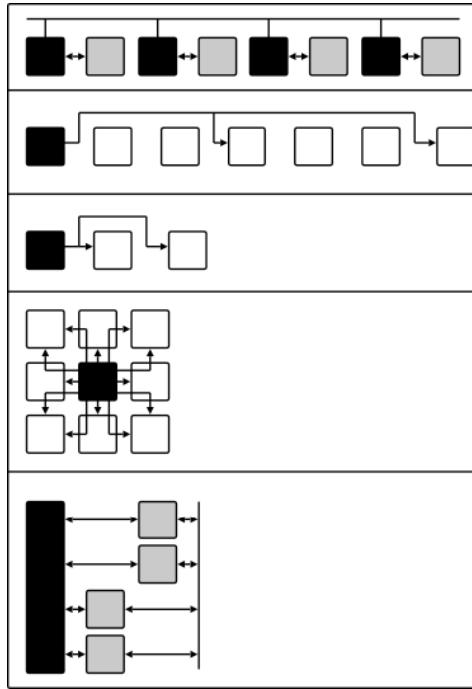


FIGURE 1.18 ■ Xilinx Virtex-II Pro routing resources. (Source: Adapted from [7], Figure 54, p. 45.)

- 16 direct connect routes that route to all immediate neighbors
- 8 fast connect lines in each CLB that connect LUT inputs and outputs

1.6 SUMMARY

This chapter presented the basic inner workings of FPGAs. We introduced the basic idea of lookup table computation, explained the need for dedicated computational blocks, and described common interconnection strategies. We learned how these devices maintain generality and programmability while providing performance through dedicated hardware blocks. We investigated a number of ways to program and maintain user-defined configuration information. Finally, we tied it all together with brief overviews of two popular commercial architectures, the Altera Stratix and the Xilinx Virtex-II Pro.

Now that we have introduced the basic technology that serves as the foundation of reconfigurable computing, we will begin to build on the FPGA to create reconfigurable devices and systems. The following chapters will discuss how to efficiently conceptualize computations spatially rather than procedurally, and the algorithms necessary to go from a user-specified design to configuration

data. Finally, we'll look into some application domains that have successfully exploited the power of reconfigurable computing.

References

- [1] J. Rose, A. E. Gamal, A Sangiovanni-Vincentelli. Architecture of field-programmable gate arrays. *Proceedings of the IEEE* 81(7), July 1993.
- [2] P. Chow, et al. The design of an SRAM-based field-programmable gate array—Part 1: Architecture. *IEEE Transactions on VLSI Systems* 7(2), June 1999.
- [3] H. Fan, J. Liu, Y. L. Wu, C. C. Cheung. On optimum switch box designs for 2-D FPGAs. *Proceedings of the 38th ACM/SIGDA Design Automation Conference (DAC)*, June 2001.
- [4] ———. On optimal hyperuniversal and rearrangeable switch box designs. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 22(12), December 2003.
- [5] H. Schmidt, V. Chandra. FPGA switch block layout and evaluation. *IEEE International Symposium on Field-Programmable Gate Arrays*, February 2002.
- [6] Xilinx, Inc. *Xilinx XC4000E and XC4000X Series Field Programmable Gate Arrays, Product Specification* (Version 1.6), May 1999.
- [7] Altera Corp. *Altera Stratix™ Device Handbook*, July 2005.
- [8] Xilinx, Inc. *Xilinx Virtex-II Pro™ and Virtex-II Pro™ Platform FPGA Handbook*, October 2005.

Précis: A Usercentric Word-Length Optimization Tool

Mark L. Chang

Franklin W. Olin College of Engineering

Scott Hauck

University of Washington

Translating an algorithm designed for a general-purpose processor into an algorithm optimized for custom logic requires extensive knowledge of the algorithm and the target hardware. Précis lets designers analyze the precision requirements of algorithms specified in Matlab. The design time tool combines simulation, user input, and program analysis to help designers focus their manual precision optimization efforts.

■ **ONE OF THE MOST DIFFICULT** tasks in implementing an algorithm in custom hardware is handling precision problems. Typical general-purpose processor concepts such as word size and data type are no longer valid in the world of custom logic, where data paths might be custom tailored to suit the algorithm's needs. Instead, designers must implement and use bit-precise data paths.

More specifically, for a general-purpose processor, algorithm designers can typically choose from a predefined set of variable types with a fixed word length. Examples are C data types such as *char*, *int*, *float*, and *double*. These data types correspond to variable-size data paths within the microprocessor. Most of the work of padding, word boundary alignment, and operation selection is hidden from the programmer by compilers and assemblers, thus making one data type as easy to use as another.

In contrast, custom and customizable hardware such as an ASIC or an FPGA do not have predefined data path widths that allow designers to tune data paths to any width desired, and choosing the appropriate data path size can be quite difficult. Having too many bits along a data path is wasteful, whereas having too few can result in erroneous output.

The difficulty is translating an initial algorithm into one that is precision optimized for hardware imple-

mentation. This task requires extensive knowledge of both the algorithm and the target hardware. Unfortunately, there are few tools that aid the hardware designer in this translation. We fill that gap by introducing Précis, a usercentric tool for design time analysis of the impact of precision on algorithm implementation.

Converting software algorithms to hardware

At the head of the development chain is the algorithm. Often, the algorithm under consideration has been implemented in a high-level language such as Matlab, C, or Java, targeted to run on a general-purpose processor such as a workstation or desktop PC. The most compelling reason to use a high-level language running on a workstation is that it provides considerable flexibility and a comfortable, rich environment in which to rapidly prototype algorithms. Of course, the reason for converting the algorithm to a hardware implementation is to gain considerable speed, size, and power advantages.

A typical tool flow requires the designer to first convert a software-prototyped algorithm into a hardware description. From this hardware description language (HDL) specification, various intermediate tools perform simulation and generate custom logic, either through standard-cell VLSI layout or reconfigurable-logic bitstreams.

A simple conversion without precision analysis would most likely yield an unreasonably large hardware implementation. For example, by emulating a general-purpose processor or DSP with a fixed 32-bit data path throughout the system, the designer might encounter wasted area. This occurs when the data on which

Related work

Most precision optimization techniques are simulation-based, analytical, or a hybrid of the two. We can also categorize them by the amount of user interaction required to perform analysis and the amount of feedback they provide the user.

Sung and Kum¹ introduced a method and tool for word-length optimization targeting custom VLSI implementations of DSP algorithms. Purely simulation-based, this method and tool used an internal, proprietary VHDL-based simulation environment.² Cadence released this software as the commercial tool Fixed-Point Optimizer,^{3,4} which required the user to design a performance evaluation block in the description language. The block would return a positive value when quantization effects on the output were within acceptable limits. Common blocks included signal-to-quantization-noise ratio (SQNR) computations. The system used basic hardware models from a commercial VLSI standard-cell library to estimate various implementations' hardware cost. Results were positive but required a lot of manual user intervention. Although not inherently a drawback, its lack of optimization suggestions for the designer and its reliance on a programmatically determined goodness function differentiates the Fixed-Point Optimizer from our work.

In a closely related effort, Kim, Kum, and Sung used operator overloading in C++ to perform range estimation of variables and fixed-point simulation.⁵ This method pro-

vides the ability to simulate and estimate the ranges of nonlinear and time-varying algorithms. However, it is still a completely manual optimization routine for the designer, with only a simulation-based analysis and no hardware models to aid in area estimation.

Willems et al. proposed a somewhat similar method.⁶ They also used standard general-purpose programming languages with custom libraries and data types to perform fixed-point simulation. They introduced the idea of interpolating ranges of intermediate variables without requiring the user to specify them explicitly. However, the steps toward efficient optimization are left for the user to deduce interactively with no suggestions provided by the system.

Constantinides, Cheung, and Luk focused on developing algorithms for almost fully automatic word-length optimization.⁷⁻⁹ These methods still require a user-supplied criterion—either a latency target⁷ or a goodness function evaluator.^{8,9} Although the process is very nearly automatic, the employed techniques limit its scope to linear time-invariant systems. Constantinides later extended the previous efforts to nonlinear components in a data path and investigated the effect of precision optimization on power reduction.¹⁰

Stephenson, Babb, and Amarasinghe introduced the Bitwise Precision-Analysis Engine and the DeepC Silicon Compiler.^{11,12} These tools operate on C source code and

algorithm operates does not require the full 32-bit data path. In that case, pruning much of the area occupied by the oversized data path is desirable. Reducing the area of a hardware implementation has several benefits: decreased power consumption, decreased critical-path delay, and increased parallelism resulting from freeing area on the device to perform other operations simultaneously. On the other hand, when the algorithm requires more precision for some data sets than the 32-bit data path provides, incorrect results can occur because of unchecked overflow or underflow conditions.

Therefore, it is important that the designer determine more accurate data path bounds for the HDL description. Typically, this involves running a software implementation of the algorithm with representative data sets and performing manual fixed-point analysis. At the very least, this method requires reengineering the software implementation to record the ranges of variables throughout the algorithm. From the results, the designer can infer candidate bit widths for the hardware implementation. Even so, such methods are tedious and often error prone.

Unfortunately, although there are well-developed tools to help automate difficult tasks in many other stages of hardware development, few tools can automate HDL generation from a processor-oriented higher-level language specification. Although higher-level design tools are available, such as the Synopsys System Studio that supports SystemC (<http://www.synopsys.com> and <http://www.systemc.org>) and the Celoxica Handel-C Compiler (<http://www.celoxica.com>), they don't offer designer aids that help with precision analysis of existing algorithms implemented in a high-level language.

Usercentric automation

Much existing research focuses on fully automated optimization techniques (see the "Related work" sidebar). Although these methods have achieved good results, we believe designers should remain close at hand during all design phases because they possess key information that an automatic optimization methodology cannot deduce or address.

An automatic precision optimization tool must be

provide a fully automatic static approach to precision analysis and bit-width reduction. The tools don't let the designer optimize bit widths further while tolerating an error impact on the output, nor do they give any suggestions as to which direction to take for iterative optimization.

References

1. W. Sung and K.-I. Kum, "Simulation-Based Word-Length Optimization Method for Fixed-Point Digital Signal Processing Systems," *IEEE Trans. Signal Processing*, vol. 43, no. 12, Dec. 1995, pp. 3087-3090.
2. K.-I. Kum and W. Sung, "VHDL Based Fixed-Point Digital Signal Processing Algorithm Development Software," *Proc. IEEE Int'l Conf. VLSI and CAD*, IEEE Press, 1993, pp. 257-260.
3. W. Sung and K.-I. Kum, "Word-Length Determination and Scaling Software for Signal Flow Block Diagram," *Proc. Int'l Conf. Acoustics, Speech, and Signal Processing* (ICASSP 94), ACM Press, 1994, pp. 457-460.
4. *Fixed-Point Optimizer User's Guide*, Alta Group of Cadence Design Systems, San Jose, Calif., 1994.
5. S. Kim, K.-I. Kum, and W. Sung, "Fixed-Point Optimization Utility for C and C++ Based Digital Signal Processing Programs," *Proc. Workshop VLSI Signal Processing VIII*, IEEE Press, 1995, pp. 197-206.
6. M. Willems et al., "System Level Fixed-Point Design Based on an Interpolative Approach," *Proc. 34th Design Automation Conf.* (DAC 97), ACM Press, 1997, pp. 293-298.
7. G. Constantinides, P. Cheung, and W. Luk, "Heuristic Datapath Allocation for Multiple Wordlength Systems," *Proc. Conf. Design, Automation and Test in Europe* (DATE 01), IEEE CS Press, 2001, pp. 791-797.
8. G.A. Constantinides, P.Y.K. Cheung, and W. Luk, "Optimum Wordlength Allocation," *Proc. 10th Ann. IEEE Symp. Field-Programmable Custom Computing Machines* (FCCM 02), IEEE Press, 2002, pp. 219-228.
9. G.A. Constantinides, P.Y.K. Cheung, and W. Luk, "The Multiple Wordlength Paradigm," *Proc. 9th Ann. IEEE Symp. Field-Programmable Custom Computing Machines* (FCCM 01), IEEE Press, 2001, pp. 51-60.
10. G.A. Constantinides, "Perturbation Analysis for Word-Length Optimization," *Proc. 11th Ann. IEEE Symp. Field-Programmable Custom Computing Machines* (FCCM 03), IEEE Press, 2003, pp. 81-90.
11. M. Stephenson, J. Babb, and S. Amarasinghe, "Bitwidth Analysis with Application to Silicon Compilation," *Proc. SIGPLAN Conf. Programming Language Design and Implementation* (PLDI 00), ACM Press, 2000, pp. 108-120.
12. M.W. Stephenson, *Bitwise: Optimizing Bitwidths Using Data-Range Propagation*, master's thesis, Dept. Electrical Eng. and Computer Science, Massachusetts Institute of Technology, May 2000.

guided by a *goodness function* (a function that determines the quality of a result based on measurable metrics) that evaluates the performance of each optimization step. In some cases, such as 2D-image processing, a simple signal-to-noise ratio (SNR) is an appropriate goodness function. In other cases, the goodness function must be significantly more complex and is therefore more difficult to develop. In either case, the designer must implement the goodness function within the framework of the automatic optimization tool.

By simulating a human designer's evaluation of an appropriate trade-off between result quality and hardware cost, the automatic optimization tool loses a critical resource: the knowledgeable designer's greater sense of context in evaluating the current solution. Moreover, for many classes of applications, a programmatically evaluated goodness function is difficult or even impossible to implement. In other words, for many applications, a knowledgeable designer is the best, and perhaps only, guide of precision optimization.

Therefore, in many instances, a fully automatic precision optimization tool should not or cannot be used.

In a departure from fully automatic methods, we approach this problem by providing a design time precision analysis tool that interacts with the user to guide optimization of the hardware data path.

The typical sequence of steps in manual data path optimization requires the designer to answer four questions about the algorithm and the implementation:

- What are my algorithm's provable precision requirements?
- What are the effects of fixed precision on my results?
- What are my data sets' actual precision requirements?
- Where along the data path should I optimize?

By repeatedly asking and answering these questions, hardware designers can perform effective word-length optimization. To answer these questions, they must manually analyze the trade-offs between area consumption and accumulated error within the computation—a time-

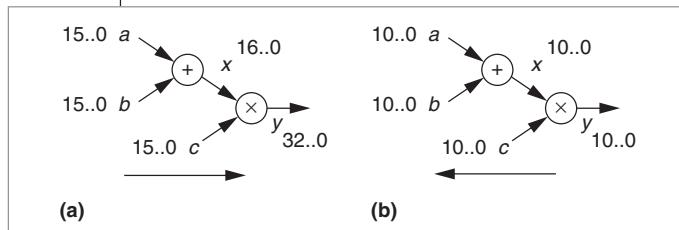


Figure 1. Simple propagation example: forward (a) and backward (b).

consuming and error-prone process. Our prototyping tool Précis aids in this process.

Précis toolset

Algorithms written in the Matlab language serve as input to Précis. Matlab is a very high-level programming language and prototyping environment that has become popular particularly in signal and image processing.^{1,2} More than just a language specification, Matlab is an interactive tool that lets designers manipulate algorithms and data sets to quickly see the impact of changes on an algorithm's output. The ease with which designers can explore an algorithm's design space in Matlab makes it a natural choice to pair with Précis for a design time precision analysis environment.

To automate many of the mundane and error-prone tasks necessary to answer the four precision analysis questions, Précis integrates several tools in a single application framework. These tools provide constraint propagation, simulation support, range-finding capabilities, and a slack analysis phase. Précis complements the existing tool flow at design time, coupling with the algorithm before it is translated into an HDL description and pushed through the vendor back-end bitstream generation tools. Thus, it provides a convenient way for the user to interact with the algorithm under consideration. Our goal is for knowledgeable users to have a much clearer idea of the precision requirements of the data paths in their algorithms.

Précis takes the parsed Matlab code output generated by the Match compiler^{3,4} (used primarily as a Matlab parser) and displays a GUI that formats the code into a treelike representation of statements and expressions. The user can click on any node and, depending on the node type, receive more information. An entry dialog lets the user specify fixed-point precision parameters, such as range and type of truncation. Using this graphical interface, the user can then perform the various tasks described in the following sections.

Propagation engine

A core component of Précis is a constraint propagation engine. Its purpose is to answer the first of the four precision analysis questions: What are the provable precision requirements of my algorithm? By learning how the algorithm's data path grows in a worst-case scenario, the designer can obtain a baseline for further optimization as well as easily pinpoint regions of interest—such as areas that explode in data path width.

The propagation engine, inspired in part by the work of Stephenson et al.,^{5,6} models the effects of using fixed-point numbers and fixed-point math in hardware. It does this by letting the user (optionally) constrain variables to a specific precision by specifying the bit positions of the most and least significant bits (MSB and LSB). Variables that are not manually constrained begin with a 64-bit default width, chosen because it is the width of a double-precision floating-point number, the base number format used in Matlab. Typically, the user can provide constraints easily for at least the circuit inputs and outputs.

The propagation engine traverses the expression tree and determines the resultant ranges of each operator expression from its child expressions. To do this, the propagation engine implements a set of rules governing the changes in resultant range that depend on the input operand range and the type of operation being performed. For example, consider the statement $a = b + c$. If the user constrains both b and c to an MSB position of 2^{15} and an LSB position of 2^0 , then output a requires 17 bits, because an addition conservatively requires one additional high-order bit for the result in the case of a carry-out from the highest-order bit. Similar rules apply for all supported operations.

The propagation engine works in this fashion across all the program's statements, recursively computing the precision of all expressions. This form of propagation is often called value-range propagation. Figure 1 depicts an example of forward and backward propagation. In this trivial example, assume the user sets all input values (a , b , and c) to use bits [15, 0]—that is, to have a range from $2^{16} - 1$ to 0. Forward propagation would result in x 's having a bit range of [16, 0] and y 's having a range of [31, 0]. If, after further manual analysis, the user finds that the output from these statements should be constrained to a range of [10, 0], backward propagation following forward propagation will constrain the multiplication's inputs (c and x) to [10, 0]. Propagating still further constrains input variables a and b to range [10, 0] as well.

The propagation engine gives a quick, macroscale

estimate of the growth rate of variables throughout the algorithm by constraining the precision of input variables and a few operators and by performing propagation. This lets the user see a conservative estimate of how the input bit width affects the size of operations downstream. Although the propagation engine provides important insight into the effects of fixed-point operations on the resultant data path, it forms a conservative estimate. For example, in an addition, the propagation engine assumes that the operation requires the carry-out bit to be set. It's appropriate to consider the data path widths determined by the propagation engine to be worst-case results, or in other words, an upper bound. This upper bound, as well as the propagation engine, becomes useful in later analysis phases of Précis.

Simulation support

To answer the second question during manual precision analysis (What are the effects of fixed precision on my results?), the designer must operate the algorithm in a fixed-point environment. Designers often do this by trial and error because there are few, if any, structured, high-level, fixed-point environments. To aid in fixed-point simulation, Précis easily produces annotated Matlab code. The user simply selects variables to constrain and requests that Matlab simulation code be generated. Figure 2 shows the code generation flow.

The code generated by the tool includes calls to Matlab helper functions that we developed to simulate a fixed-point environment, eliminating the need for the designer to construct custom fixed-point blocks. In particular, we developed a Matlab support routine, fixp, to simulate a fixed-point environment. Its declaration is

```
fixp(x, m, n, lmode, rmode)
```

where x denotes the signal to be truncated to $(m - n + 1)$ bits in width. Specifically, m denotes the MSB bit position, and n the LSB bit position, inclusively, with negative values representing positions to the right of the decimal point. The remaining two parameters, lmode and rmode, specify the method desired to deal with overflow at the variable's MSB and LSB portions, respectively. These modes correspond to different hardware implementation methods. Possible choices for lmode are *sat* (saturation to $2^{(MSB+1)} - 1$) and *trunc* (truncation of all bits above the MSB position).

For the variable's LSB side, there are four modes: *round*, *trunc*, *ceil*, and *floor*. *Round* rounds the result to the nearest integer, *trunc* truncates all bits below the LSB

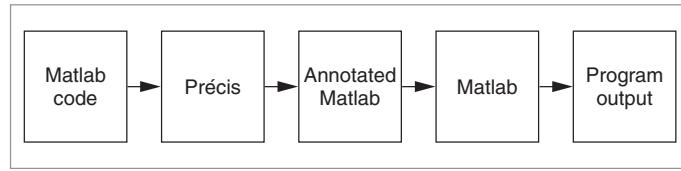


Figure 2. Code generation flow for simulation.

$a = 1;$ $b = 2;$ $c = 3;$ $d = (a + (b * c));$ (a)	$a = 1;$ $b = 2;$ $c = 3;$ $d = (\text{fixpp}(a, 12, 3, 'trunc', 'trunc') + (b * c));$ (b)
--	---

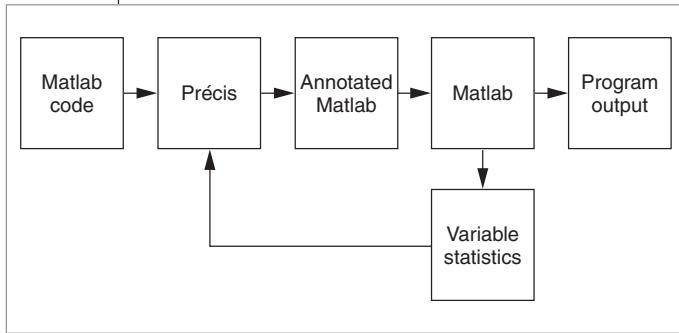
Figure 3. Sample output generated for simulation, with the range of variable *a* constrained: Matlab input (a); annotated Matlab (b).

position, *ceil* rounds up to the next integer level, and *floor* rounds down to the next-lower integer level. With the exception of *trunc*, these modes correspond exactly to Matlab functions and thus behave as documented by Mathworks. The routine performs *trunc* through the modulo operation. Figure 3 shows an example of output generated for simulation.

After the user has constrained the variables of interest and indicated the mechanism that will control overflow of bits beyond the constrained precision, Précis generates annotated Matlab code. The user can then run the generated code with real data sets. The purpose of these simulations is to determine the effects of constraining variables on the implementation's correctness. Not only might the eventual output be erroneous, but the precision constraints' effects might make the algorithm fail to operate entirely.

If the algorithm's output is acceptable, the user can consider constraining additional key variables, thereby further reducing the hardware circuit's eventual size. On the other hand, if the output generates unusable results, then the constraints were too aggressive, and the user should increase the width of the data paths used by some of the constrained variables.

During this manual phase of precision analysis, merely testing whether the fixed-precision results are identical to the unconstrained-precision results is typically not sufficient, because such testing is probably too restrictive. In applications such as image processing, lossy compression, and speech processing, users might be willing to trade some degree of result quality for a more efficient hardware implementation. As a designer assistance tool,

**Figure 4. Range-finding analysis development cycle.**

$a = 1;$ $b = 2;$ $c = 3;$ $d = (a + (b * c));$ (a)	$a = 1;$ $b = 2;$ $c = 3;$ $d = (a + (b * c));$ $\text{rangeFind}(d, 'rfv_d');$ (b)
--	--

Figure 5. Sample range-finding output: Matlab input (a); range-finding output (b).

Précis lets designers create their own goodness function and make this trade-off as they see fit. The Précis environment shortens this iterative development cycle by quickly generating and executing the fixed-point simulation code, thus letting the user view results and error effects without tediously editing algorithm source code.

Range finding

Although the simulation support just described is useful on its own for fixed-point simulation, it reaches its full usefulness only if users can accurately identify variables that they believe can be constrained. This leads to the third question that must be answered for effective data path optimization: What are my data sets' actual precision requirements? Précis answers this question by providing a range-finding capability that helps users deduce the data path requirements of intermediate nodes whose ranges are not obvious.

Figure 4 shows the range-finding development cycle. After the Matlab code is parsed, the user targets variables for range analysis and Précis generates annotated Matlab, much as it generated simulation code. Instead of fixed-point simulation, however, Précis annotates the code with another Matlab support routine that monitors the range of values attained by the variables under question.

This support routine, rangeFind, monitors the maximum and minimum values attained by the variables.

Précis runs the annotated Matlab with some sample data sets to gather range information on the variables. The user can then save these values in data files that can be fed back to Précis for further analysis phases. Figure 5 shows an example of a range-finding output.

The user then loads the range values discovered by rangeFind back into the Précis tool and (optionally) constrains the variables. The range-finding phase has given the user an accurate profile of the precision each variable requires for the data sets under test. The user can now perform propagation to conservatively estimate the effect of these data path widths on the rest of the system.

The propagation engine and the range-finding tools work together to give users a more comprehensive picture of the algorithm's precision requirements than either tool could provide alone. The propagation engine, with user knowledge of input and perhaps output variable constraints, achieves a first-order estimate of the algorithm's data path widths. The range-finding information significantly refines this estimate because the discovered variable statistics allow the implementation of narrower data path widths that more closely reflect the algorithm's true precision requirements.

In another useful step, users can constrain variables even further than suggested by the range-finding phase and then perform subsequent simulations to learn whether these further refinements introduce an acceptable amount of error into the result. Like the earlier ones, these simulations are easy to generate and execute in the Précis framework.

The range-finding method's results are data set dependent. Users must take care to use representative data sets; if the data sets are significantly different from one another in precision requirements, even on the same algorithm, the final hardware implementation can generate erroneous results. For the range-finding phase to gather meaningful and robust statistics, the data sets should represent the precision of the common case as well as boundary and extreme cases.

It is useful, therefore, to regard range-gathered precision information as a lower bound on the precision required by the algorithm. Because user-run data sets exercise a known amount of data path width, any further reduction in precision will likely incur error. Given that the precisions obtained from the propagation engine are conservative estimates, or an upper bound, manipulating the difference between these two bounds leads to a novel method of user-guided precision analysis, which we call slack analysis.

Slack analysis

One of the goals of Précis is to help designers know where to focus manual precision analysis and hardware-tuning efforts. This is the subject of the fourth precision analysis question: Where along the data path should I optimize? To help designers answer this question, Précis provides a list of tuning points in decreasing order of potential overall reduction of circuit size. With this information, the designer can start a hardware implementation using generic data path precisions, such as a standard 64- or 32-bit data path, and iteratively optimize code sections that yield the most benefit. Iteratively optimizing code sections or hardware is a commonly used technique for efficiently meeting constraints such as development time, cost, area, performance, or power. The tuning list gives designers effective starting points for each manual optimization iteration, putting them on the most direct path to meeting their constraints.

Recall that if we perform range-finding analysis and propagation analysis on the same set of variables, the tool obtains what amounts to a lower bound from range analysis and an upper bound from propagation. We consider range analysis a lower bound because it is the result of true data sets. Whereas other data sets might require even less precision, we know we need at least the ranges gathered from range analysis to maintain error-free output. Further testing with other data sets might show that some variables require more precision. Thus, if we implement the design with the precision found, we might encounter erroneous output, supporting the premise that range analysis finds a lower bound.

On the other hand, propagation analysis is very conservative. For example, in the statement $a = b + c$, where the user has constrained b and c to 16 bits wide, the resultant bit width of a can be 17 bits because of the addition. In reality, however, both b and c can be well within the 16-bit limit and an addition might never overflow into the 17th bit position. For example, if $c = \lambda - b$, then λ governs the range of values that a can ever attain. To someone investigating this code section, this seems very obvious when $\lambda - b$ is substituted for c in $a = b + c$. But these more macroscopic constraints in algorithms are difficult or impossible to find automatically outside of linear time-invariant systems.^{7,8} This is why we consider propagated range information to be an upper bound.

Given a lower and upper bound on a variable's bit width, we can treat the difference between them as slack. The actual precision requirement most likely lies between the two bounds. Manipulating the precision of nodes that have slack can achieve systemwide precision gain

because changes in any single node can affect many other nodes in the circuit. We define gain as a reduction in precision requirements and the resultant improvements in area, power, and performance. Through careful analysis of a node's slack, Précis calculates how much gain we can achieve by manipulating the precision between the two bounds. Additionally, by performing this analysis independently for each node with slack, the tool generates an ordered list of tuning points that the user should consider when performing optimization iteration.

To compute a node's gain with respect to area, power, and performance, we have developed basic hardware models to capture the effect of precision changes on these parameters. We used a simple area model as our main metric. For example, an adder has an area model of x , indicating that as the precision decreases by one bit, the area reduces linearly and the gain increases linearly. In contrast, a multiplier has an area model of x^2 , indicating that the area reduction and gain achieved are proportional to the square of the word size. Intuitively, these models will result in a higher overall gain value for a multiplier's bit reduction than an adder's, in line with implementations familiar to hardware designers. Using these parameters, our approach chooses the nodes with the most possible gain to suggest to the user.

The goal of slack analysis is to identify which nodes, when constrained, are likely to have the greatest effect on circuit area. Although it is unrealistic to expect users to constrain all variables, users should consider constraining a few controlling values. Précis helps users spend time efficiently by guiding them to the most important variables to consider. Précis can also provide users a stopping criterion: After measuring the maximum possible benefit from future constraints by constraining all variables to their lower bounds, users can decide to stop investigating when the difference between the current and lower bound area is no longer worth optimizing.

The slack analysis methodology in Précis is straightforward. For each node with slack, the tool sets the precision to the range-finding value—the lower bound. Then Précis propagates that change's impact over all nodes and calculates its overall gain in terms of systemwide area. After recording this value as the effective gain resulting from modifying that node, the user resets all nodes and repeats the procedure for the remaining nodes with slack. The tool sorts the resultant list of gain values in decreasing order and presents this information to the user in a dialog window. The GUI lets the user easily see how to modify the appropriate nodes to achieve the highest gain. It is then up to the user to determine

Perform Slack Analysis

```

1 constrain user-specified variables
2 perform propagation
3 baseArea←calculateArea()
4 load range data for some set of variables n
5 listOfGains←0
6 for each m in n
7   reset all variables to baseline precision
8   constrain range of m to the range analysis value
9   perform forward and reverse propagation
10  newArea←calculateArea()
11  if (newArea < baseArea) then
12    listOfGains←(m,baseArea – newArea)
13 sort listOfGains by decreasing gain

```

Figure 6. Slack analysis pseudocode.

lower bound of the area by implementing all suggestions simultaneously to determine how quickly the tool converged on the lower bound.

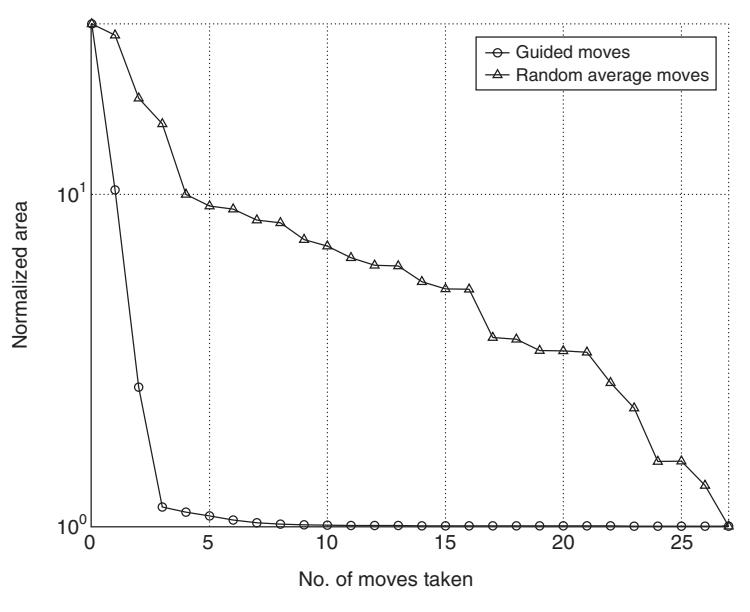
Wavelet transform

The first benchmark we optimized is the wavelet transform, a form of image processing applied primarily prior to application of a compression scheme such as set partitioning in hierarchical trees (SPIHT).^{9,10} A typical discrete wavelet transform runs high- and low-pass filters over the input image in one dimension. The results are then downsampled by a factor of two, and the process is repeated in the other dimension. Each pass results in a new image composed of a high- and low-pass subband, each half the size of the original input stream. These sub-bands can be used to reconstruct the original image.

Earlier, Fry mapped this algorithm to hardware, spending significant time converting the floating-point source algorithm into a fixed-point representation with methodologies similar to those we present here, albeit by hand.⁹ The result was an implementation running at 56 MHz, capable of compressing 8-bit images at a rate of 800 Mbits per second. This represents a speedup of nearly 450 times compared with a software implementation running on a Sun Sparcstation 5.

We subsequently implemented the wavelet transform in Matlab and optimized it in Précis. In total, we selected 27 variables to be constrained. We marked these variables for range-finding analysis and generated annotated Matlab code. We then ran this code in the Matlab interpreter with a sample image file (Lena) to obtain range values for the selected variables. We loaded these values into Précis to obtain a lower bound for use during slack analysis.

Figure 7 shows the slack-analysis results. We normalized these results to the lower bound obtained by setting all variables to their lower-bound constraints and computing the resulting area. The results suggested constraining the input image array, then the low- and high-pass filter coefficients, and finally the results of the additions in the filtering operation's multiply-accumulate structure. By iteratively performing the optimization moves suggested by the Précis slack analysis, we came within 15% of the lower-bound area in three moves. In about seven moves, the nor-

**Figure 7. Wavelet transform results: area versus number of optimization steps implemented.**

which nodes, if any, should actually be more tightly constrained than suggested by Précis. Figure 6 shows the pseudocode for the slack analysis procedure.

Benchmarks

To measure its effectiveness, we used Précis to optimize various image- and signal-processing benchmarks. To evaluate its suggestions, we constrained the variables in the order the tool suggested them and calculated the resulting area. We determined the area using the model discussed earlier, giving adders a linear area model and multipliers an area model proportional to the square of their input word size. We also determined an asymptotic

range values for the selected variables. We loaded these values into Précis to obtain a lower bound for use during slack analysis.

Figure 7 shows the slack-analysis results. We normalized these results to the lower bound obtained by setting all variables to their lower-bound constraints and computing the resulting area. The results suggested constraining the input image array, then the low- and high-pass filter coefficients, and finally the results of the additions in the filtering operation's multiply-accumulate structure. By iteratively performing the optimization moves suggested by the Précis slack analysis, we came within 15% of the lower-bound area in three moves. In about seven moves, the nor-

malized area was within 3% of the lower bound, and further improvements were negligible. At this point, a typical user would choose to stop optimizing the system.

To determine whether this methodology is sound, we compared the suggested optimization steps with the performance we would obtain if we optimized randomly. We performed four optimization runs in which the nodes selected for optimization were randomly chosen from the set of nodes with slack. We used the same values for the upper- and lower-precision bounds as the guided optimization scheme. Figure 7 plots the average area of these random passes against the guided slack-analysis approach. As shown, the guided optimization route suggested by Précis reaches very near the lower bound more quickly than the random method. The random method, though improving with each optimization step, does so far more slowly than the guided slack-analysis approach. From this, we conclude that the slack-analysis approach provides the user valuable feedback on which node selection and order generates the largest gains in the fewest steps. We performed the comparison with random moves for all the following benchmarks.

We must point out that Précis calculates area values by reducing the range of a number of variables to their range-found lower bounds. This yields what we can regard as the best-case solution only for the input data sets considered. In reality, using different input data with these range-found lower bounds might introduce errors into the system. Therefore, it is important to continue testing the solution with new data sets after optimization is complete. Automatic generation of annotated simulation code for use in Matlab makes this testing step easier.

Cordic

The next benchmark is the Cordic (coordinate rotation digital computer) algorithm.¹¹ The algorithm is novel in that it is an iterative solver for trigonometric functions that requires only a simple network of shifts and adds and produces approximately one additional bit of accuracy for each iteration. Andraka presents a more detailed discussion of the algorithm and a survey of its FPGA implementations.¹²

The Cordic algorithm operates in two modes: rota-

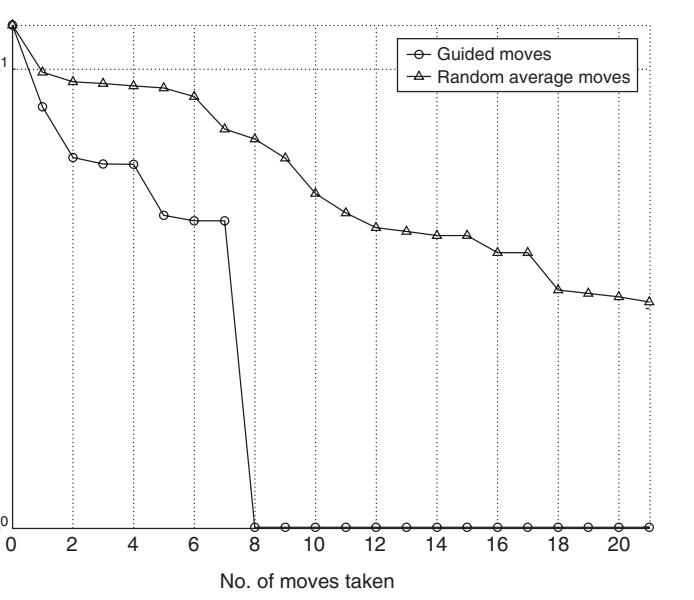


Figure 8. Cordic benchmark results: area versus number of optimization steps implemented.

tion and vectoring. For this benchmark, we used rotation mode, which rotates an input vector by a specified angle, simultaneously computing the input angle's sine and cosine. The difference equations for rotation mode (from Andraka¹²) are

$$\begin{aligned}x_{i+1} &= x_i - y_i d_i (2^{-i}) \\y_{i+1} &= y_i + x_i d_i (2^{-i}) \\z_{i+1} &= z_i - d_i \tan^{-1}(2^{-i})\end{aligned}$$

where

$$d_i = \begin{cases} -1 & \text{if } z_i < 0 \\ +1 & \text{otherwise} \end{cases}$$

We unrolled the Matlab implementation of Cordic in 12 stages. To obtain a variety of variable-range information during the range-finding phase, we developed a test harness that swept the input angle through all integer angles between 0 and 90 degrees. Then we passed the results to Précis and chose all 41 intermediate nodes for slack analysis. Figure 8 shows the results, truncated to the first 21 moves suggested by the tool. The results are consistent with those for the wavelet benchmark.

The suggested moves don't converge on the lower bound as quickly as for the wavelet benchmark, not reaching the lower-bound area until the eighth move. We attribute this to the slack-analysis algorithm's greedy

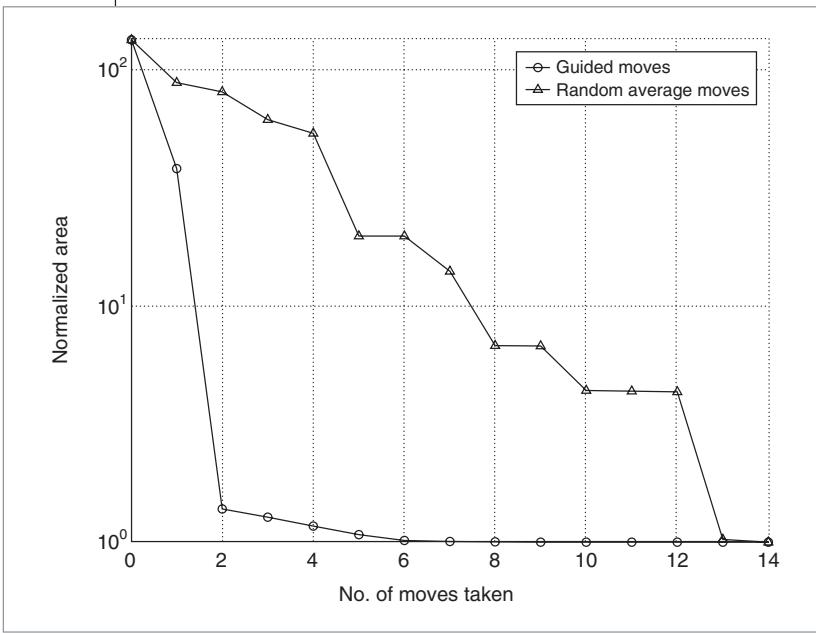


Figure 9. Gaussian blur results: area versus number of optimization steps implemented.

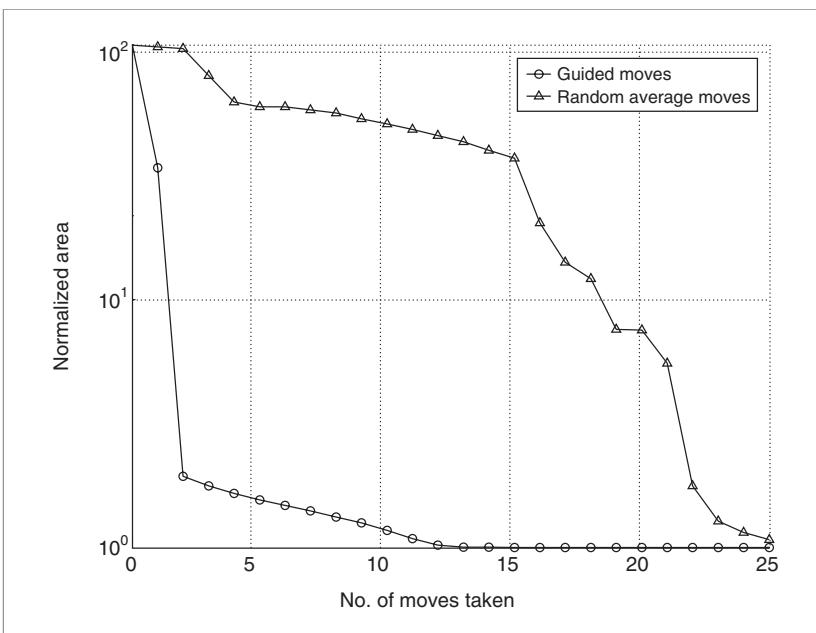


Figure 10. Eight-point 1D DCT results: area versus number of optimization steps implemented.

nature. The first few proposed moves all originated at the outputs. Only after these were constrained did the slack analysis suggest moving to the input variables. This behavior is partly due to the depth of the adder tree pre-

sent in the algorithm's 12-stage unrolling. The gain achieved by constraining the outputs was greater than the limited impact of constraining any one input, because the output nodes were significantly larger. Shortly after constraining the outputs, though, Précis constrained all the input variables, obtaining a large improvement in area after the seventh suggested move, at which point the Cordic algorithm's very linear data path collapsed to near the lower bound.

Gaussian blur

The third benchmark is a Gaussian blur implemented as a spatial convolution of a 3×3 Gaussian kernel with a 512×512 gray-scale input image. For simplicity, we ignored rescaling the blurred image. We input the algorithm into Précis and chose 14 intermediate nodes for slack analysis. Figure 9 shows the results. The slack analysis prompted us to constrain first the Gaussian kernel and then the input image. This led to the largest area improvement—within 28% of the lower bound in three moves, and within 8% in five moves. Again, the tool made good choices for optimization and achieved performance near the lower bound in far fewer optimization steps than with the random-move approach.

1D discrete cosine transform

The next benchmark is a 1D discrete cosine transform. The DCT is a frequency transform much like the discrete Fourier transform, except that it uses only real numbers.¹³ It is widely used in image and video compression. We based our implementation on Loeffler's work,¹⁴ as used by the Independent JPEG Group's JPEG software distribution.¹⁵ Our implementation requires only 12 multiplications and 32 additions.

Our Matlab implementation performed an eight-point 1D DCT on a 512×512 input image. Figure 10 shows the results for all 25 nodes chosen for slack analysis. To get within a factor of two of the lower bound, we constrained the input image and DCT

input vector. The suggested moves achieved within 50% of the lower bound in six moves, and within 2% in 12 moves.

Probabilistic neural network

The final benchmark we investigated was a multispectral image-processing algorithm designed for NASA satellite imagery. Performing a function similar to clustering analysis or image compression, the algorithm uses multiple spectral bands of instrument observation data to categorize each image pixel into one of several classes. For the NASA application, these classes define terrain types, such as urban, agricultural, rangeland, and barren. In other implementations, these classes could be any significant distinguishing attributes present in the underlying data set. This type of algorithm transforms multispectral images into a form more useful for human analysis.

One proposed scheme for performing this automatic classification is the probabilistic neural network (PNN) classifier.¹⁶ This implementation compares each multispectral image pixel vector with a set of training pixels or weights known to be representative of a particular class. The following formula gives the probability that the pixel under test, \vec{X} , belongs to the class under consideration, S_k . This comparison is made for all classes, and the class with the highest probability indicates the closest match:

$$f(\vec{X}|S_k) = \frac{1}{(2\pi)^{d/2} \sigma^d} \left(\frac{1}{P_k} \right) \left(\sum_{i=1}^{P_k} \exp \left[-\frac{(\vec{X} - \vec{W}_{ki})^T (\vec{X} - \vec{W}_{ki})}{2\sigma^2} \right] \right)$$

Here, \vec{W}_{ki} is the weight i of class k , d is the number of spectral bands, k is the class under consideration, σ is a data-dependent smoothing parameter, and P_k is the number of weights in class k .

An earlier study involved manually implementing this algorithm on an FPGA board.¹⁷ Like the wavelet transform described earlier, the algorithm required significant time and effort for variable-range analysis, with particular attention to the large multipliers and exponentiation. This manual implementation obtained

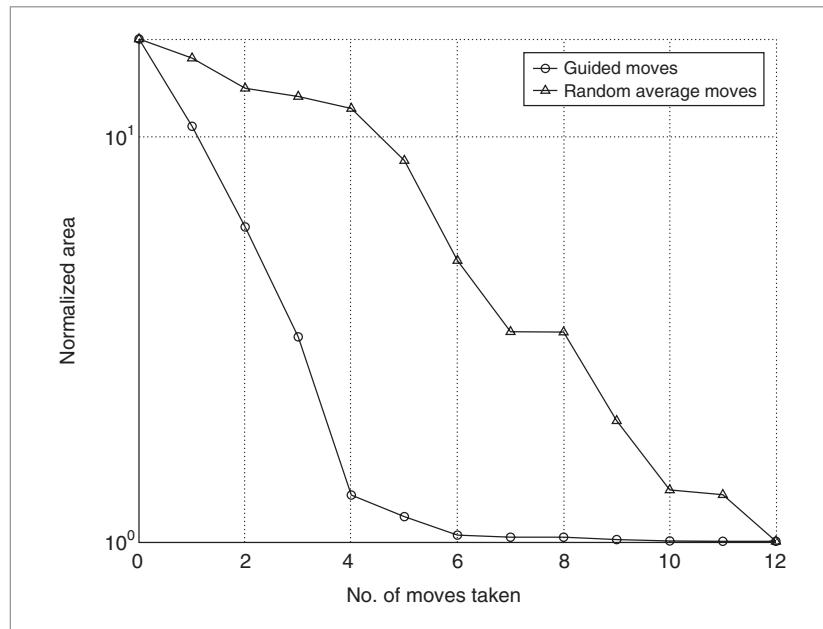


Figure 11. Probabilistic neural network classifier results: area versus number of optimization steps implemented using only range-analysis-discovered values.

speedups of 16 times versus a software implementation on an HP workstation.

We implemented the algorithm in Matlab and optimized it with Précis. We selected 12 variables and performed slack analysis as for the previous benchmarks. Again, we normalized all results to the lower-bound area. As Figure 11 shows, the tool behaved consistently with other benchmarks and came within 4% of the lower bound within six moves, after which additional moves made only minor improvements in area.

For a seasoned designer who has insight into the algorithm and already has an idea of how the algorithm will map to hardware, the range analysis sometimes returns suboptimal results. For example, in our experiments, the PNN algorithm's range analysis of a typical data set resulted in several variables' being constrained to ranges such as $[2^0, 2^{-25}]$, $[2^8, 2^{-135}]$, $[2^0, 2^{-208}]$, and so on. This simply means that the range-finding phase discovered extremely small values and thus recorded the range as requiring many fractional bits (bits to the right of the decimal point) to capture all precision information. The shortcoming of automated range analysis is that it cannot determine the precision at which values become too small to affect subsequent calculations and therefore might be considered unimportant. With this in mind, the designer typically restricts the variables to nar-

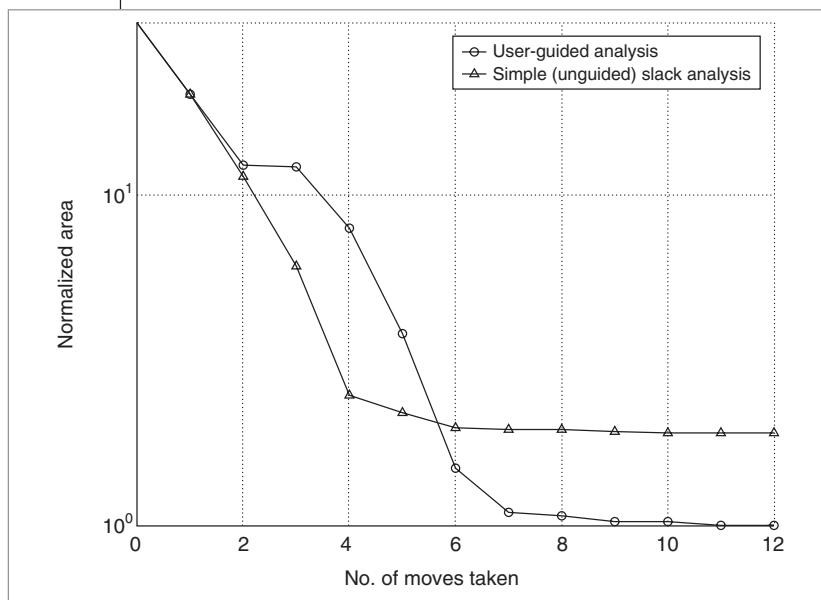


Figure 12. Probabilistic neural network classifier results: area versus number of optimization steps implemented using user-defined variable-precision ranges.

rower ranges that preserve the results' correctness while requiring fewer bits of precision.

Précis provides a functionality that lets users make these decisions during annotated Matlab code generation. In that case, the user chooses a narrower precision range and a method by which to constrain the variable to that range, consistent with how the operation will be implemented in hardware—truncation, saturation, rounding, or any of the other methods presented earlier. Then, the developer generates annotated Matlab code for simulation and reruns the algorithm in Matlab with typical data sets. This lets the user determine how narrow a precision range is tolerable, and subsequently to constrain the variables in Précis accordingly. The user can then continue the slack-analysis phase, optionally reconstraining variables through simulation as wider-than-expected precision ranges are encountered.

We performed this user-guided method by reconstraining the variables suggested by the slack-analysis phase to more reasonable ranges. For instance, the third variable suggested by slack analysis, *classTotal*, had a range-found precision of $[2^{10}, 2^{-60}]$, far too wide to implement in an area-efficient manner. We reconstrained this value to $[2^{37}, 2^0]$, which includes an implicit scaling factor. We performed this reconstraining in the order the variables were suggested by Précis. Figure 12 shows the results, normalized to the lowest bound between the

standard and user-guided approaches.

At first glance, the two methods appear to show similar trends, approaching the lower bound within five to seven moves. We expected this behavior, and it's consistent with the other benchmarks' results. However, the results also show that when we reconstrain variables to narrower ranges during slack analysis, the user-guided approach achieves a lower bound almost 50% lower than slack analysis without user guidance. As expected, the unguided approach makes no further improvement as the number of optimization steps increases.

In this case, Précis used the hardware designer's intuition to achieve a more area-efficient implementation than possible with unguided slack-analysis optimization. The ability to keep the user in the loop for optimization is crucial to obtaining good implementations, an ability that Précis clearly exploits.

PRÉCIS AIDS both new and experienced hardware designers in performing data path optimization at a very high level, before HDL is generated. At this time, small design changes almost always lead to large differences in the final implementation's performance. Thus, it is crucial that designers have data path optimization tools from the very beginning of the design cycle.

Our ongoing research focuses on developing techniques for accurate area and quantization error estimation for precision analysis. Specifically, we are developing methodologies for selecting the least-significant-bit position along a data path subject to area and error constraints. ■

Acknowledgments

This work was partially supported by grants from the National Science Foundation and NASA, and donations from Xilinx. Scott Hauck was partially supported by an NSF CAREER award and a Sloan Fellowship. Mark L. Chang was partially supported by an Intel Fellowship.

■ References

1. C.B. Moler, *MATLAB—An Interactive Matrix Laboratory*, tech. report 369, Univ. of New Mexico, Dept. of Computer Science, 1980.

2. C.B. Moler, *MATLAB User's Guide*, tech. report, University of New Mexico, Dept. of Computer Science, 1980.
3. P. Banerjee et al., *MATCH: A MATLAB Compiler for Configurable Computing Systems*, tech. report CPDC-TR-9908-013, Northwestern Univ., ECE Dept., 1999.
4. P. Banerjee et al., "A MATLAB Compiler for Distributed, Heterogeneous, Reconfigurable Computing Systems," *Proc. IEEE Symp. Field-Programmable Custom Computing Machines* (FCCM 00), IEEE CS Press, 2000, pp. 39-48.
5. M. Stephenson, J. Babb, and S. Amarasinghe, "Bitwidth Analysis with Application to Silicon Compilation," *Proc. SIGPLAN Conf. Programming Language Design and Implementation* (PLDI 00), ACM Press, 2000, pp. 108-120.
6. M.W. Stephenson, *Bitwise: Optimizing Bitwidths Using Data-Range Propagation*, master's thesis, Dept. Electrical Eng. and Computer Science, Massachusetts Institute of Technology, 2000.
7. G.A. Constantinides, P.Y.K. Cheung, and W. Luk, "Optimum Wordlength Allocation," *Proc. 10th Ann. IEEE Symp. Field-Programmable Custom Computing Machines* (FCCM 02), IEEE CS Press, 2002, pp. 219-228.
8. G.A. Constantinides, P.Y.K. Cheung, and W. Luk, "The Multiple Wordlength Paradigm," *Proc. 9th Ann. IEEE Symp. Field-Programmable Custom Computing Machines* (FCCM 01), IEEE Press, 2001, pp. 51-60.
9. T.W. Fry, *Hyperspectral Image Compression on Reconfigurable Platforms*, master's thesis, Dept. Electrical Eng., Univ. of Washington, 2001.
10. T.W. Fry and S. Hauck, "Hyperspectral Image Compression on Reconfigurable Platforms," *Proc. IEEE Symp. Field-Programmable Custom Computing Machines* (FCCM 02), IEEE CS Press, 2002, pp. 251-260.
11. J. Volder, "The CORDIC Trigonometric Computing Technique," *IRE Trans. Electronic Computing*, vol. 8, no. 3, Sept. 1959, pp. 330-334.
12. R. Andraka, "A Survey of CORDIC Algorithms for FPGAs," *Proc. ACM/SIGDA 6th Int'l Symp. Field Programmable Gate Arrays* (FPGA 98), ACM Press, 1998, pp. 191-200.
13. N. Ahmed, T. Natarajan, and K. Rao, "Discrete Cosine Transform," *IEEE Trans. Computers*, vol. 23, no. 1, Jan. 1974, pp. 90-93.
14. C. Loeffler, A. Ligtenberg, and G.S. Moschytz, "Practical Fast 1-D DCT Algorithms with 11 Multiplications," *Proc. Int'l Conf. Acoustics, Speech, and Signal Processing* (ICASSP 89), IEEE Press, 1989, pp. 988-991.
15. T. Lane et al., "The Independent JPEG Group's JPEG Software Library," June 2004, <http://www.ijg.org/files/jpegsrc.v6b.tar.gz>.
16. S.R. Chettri, R.F. Cromp, and M. Birmingham, "Design of Neural Networks for Classification of Remotely Sensed Imagery," *Telematics and Informatics*, vol. 9, nos. 3-4, 1992, pp. 145-156.
17. M.L. Chang, *Adaptive Computing in NASA Multi-Spectral Image Processing*, master's thesis, Northwestern Univ., Dept. Electrical and Computer Eng., 1999.



Mark L. Chang is an assistant professor of electrical and computer engineering at the Franklin W. Olin College of Engineering. He was with the University of Washington when he did this work. His research interests include engineering education and FPGA architectures, tools, and applications. Chang has a BS in electrical and computer engineering from Johns Hopkins University, an MS in electrical and computer engineering from Northwestern University, and a PhD in electrical engineering from the University of Washington.



Scott Hauck is an associate professor of electrical engineering at the University of Washington. His research interests include FPGA architectures, applications, and CAD tools. Hauck has a BS in computer science from the University of California, Berkeley, and an MS and a PhD, both in computer science, from the University of Washington.

■ Direct questions and comments about this article to Mark L. Chang, Franklin W. Olin College of Engineering, Olin Way, Needham, MA 02492; mark.chang@olin.edu.

For further information on this or any other computing topic, visit our Digital Library at <http://www.computer.org/publications/dlib>.

Automated Least-Significant Bit Datapath Optimization for FPGAs

Mark L. Chang and Scott Hauck

Department of Electrical Engineering

University of Washington

Seattle, Washington

Email: {mchang,hauck}@ee.washington.edu

Abstract—In this paper we present a method for FPGA datapath precision optimization subject to user-defined area and error constraints. This work builds upon our previous research [1] which presented a methodology for optimizing for dynamic range—the most significant bit position. In this work, we present an automated optimization technique for the least-significant bit position of circuit datapaths. We present results describing the effectiveness of our methods on typical signal and image processing kernels.

I. INTRODUCTION

With the widespread growth of reconfigurable computing platforms in education, research, and industry, more software developers are being exposed to hardware development. Many are seeking to achieve the enormous gains in performance demonstrated in the research community by implementing their software algorithms in a reconfigurable fabric. For the novice hardware designer, this effort usually begins and ends with futility and frustration as they struggle with unwieldy tools and new programming paradigms.

One of the more difficult paradigm shifts to grasp is the notion of bit-level operations. On a typical FPGA fabric, logical and arithmetic operators can work at the bit level instead of the word level. With careful optimization of the precision of the datapath, the overall size and relative speed of the resulting circuit can be dramatically improved.

In this paper we present a methodology that broadens the work presented in [1]. We begin with background on precision analysis and previous research efforts. We describe the problem of least-significant bit optimization and develop several optimization techniques that provide finer control of area-to-error tradeoffs than more traditional methods. We then present a simulated annealing-based approach to automatically apply these optimizations to a datapath. Finally, we present the results of using our techniques to optimize the datapath of image processing circuits and draw some conclusions.

II. BACKGROUND

General-purpose processors are designed to perform operations at the word level, typically 8, 16, or 32 bits. Supporting this paradigm, programming languages and compilers abstract these word sizes into storage classes, or data-types, such as `char`, `int`, and `float`. In contrast, most mainstream reconfigurable logic devices, such as FPGAs, operate at the

bit level. This allows the developer to tune datapaths to any word size desired. Unfortunately, choosing the appropriate size for datapaths is not trivial. Choosing a wide datapath, as in a general-purpose processor, usually results in an implementation that is larger than necessary. This consumes valuable resources and potentially reduces the performance of the design. On the other hand, if the hardware implementation uses too little precision, errors can be introduced at runtime through quantization effects, such as roundoff and truncation.

To alleviate the programmer’s burden of doing manual precision analysis, researchers have proposed many different solutions. Techniques range from semi-automatic to fully-automated methods that employ static and dynamic analysis of circuit datapaths. We will touch on some of these efforts in the following section.

A. The Least-Significant Bit Problem

In determining the fixed-point representation of a floating-point datapath, we must consider both the most-significant and least-significant ends. Reducing the relative bit position of the most-significant bit reduces the maximum value that the datapath may represent, sometimes referred to as the dynamic range. On the other end, increasing the relative bit position of the least-significant bit (toward the most-significant end) reduces the maximum precision that the datapath may attain. For example, if the most-significant bit is at the 2^7 position, and the least-significant bit is at the 2^{-3} position, the maximum value attainable by an unsigned number will be 255.875, while the precision will be quantized to multiples of $2^{-3} = 0.125$. Values smaller than 0.125 cannot be represented as the bits necessary to represent, for example, 0.0625, do not exist.

Having a fixed-point datapath means that results or operations may exhibit some quantity of error compared to their floating-point counterparts. This quantization error can be introduced in both the most-significant and least-significant sides of the datapath. If the value of an operation is larger than the maximum value that can be represented by the datapath, the quantization error is typically a result of truncation or saturation, depending on the implementation of the operation. Likewise, error is accumulated at the least-significant end of the datapath if the value requires greater precision than the

datapath can represent, resulting in truncation or round-off error.

Previous research includes [2], [3], which only performs the analysis on the most-significant bit position of the datapath. While this method achieves good results, it ignores the potential optimization of the least-significant bit position. Other research, including [4], [5] begin to touch on fixed-point integer representations of numbers with fractional portions. Finally, more recent research, [6], [7] begin to incorporate error analysis into the overall optimization of the fractional width of the datapath elements.

Most of the techniques introduced deal with either limited scope of problem, such as linear time-invariant (LTI) systems, and/or perform the analysis completely automatically, with minimal input from the developer. While again, these methods achieve good results, it is our belief that the developer should be kept close at hand during all design phases, as there are some things for which an automatic optimization method simply cannot handle.

Simply put, a “goodness” metric must be devised in order to guide an automatic precision optimization tool. This “goodness” function is then evaluated by the automated tool to guide its precision optimization. In some cases, such as image processing, a simple block signal-to-noise ratio (BSNR) may be appropriate. In many cases, though, this metric is difficult or impossible to evaluate programmatically. A human developer, therefore, has the benefit of having a much greater sense of context in evaluating what is an appropriate tradeoff between error in the output and performance of the implementation. We have used this idea as the guiding principle behind the design of our precision analysis tool Précis [1]. In this paper we provide the metrics and methodology for performing least-significant-bit optimization.

III. ERROR MODELS

The observation that the relative bit position of the least-significant bit introduces a quantifiable amount of error over a floating-point datapath is an important one. After performing the optimization for the most-significant bit position as described in [1], we must perform an area/error analysis phase to optimize the position of the least-significant bit. In order to quantify changes to the datapath, we introduce models for area and error estimation of a general island-style FPGA.

Consider an integer value that is M' bits in length. This value has an implicit binary point at the far right—to the right of the least-significant bit position. By truncating bits from the least-significant side of the word, we reduce the area impact of this word on downstream arithmetic and logic operations. It is common practice to simply truncate the bits from the least-significant side to reduce the number of bits required to store and operate on this word. We propose an alternate method—replace the bits that would normally be truncated with constants, in this case zeros. Therefore, for an M' -bit value, we will use the notation $A_m 0_p$. This denotes a word that has m correct bits and p zeros inserted to signify bits that

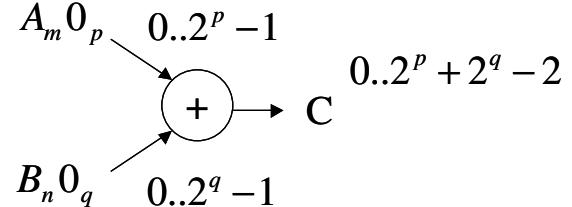


Fig. 1. Error model of an adder.

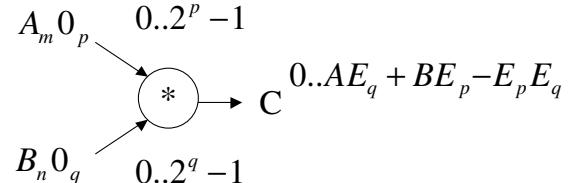


Fig. 2. Error model of a multiplier.

have been effectively truncated, resulting in an $M' = m+p$ -bit word.

Having performed a reduction in the precision that can be obtained by this datapath with a substitution of zeros, we have introduced a quantifiable amount of error into the datapath. For an $A_m 0_p$ value, substituting p zeros for the lower portion of the word, gives us a maximum error of $2^p - 1$. This maximum error occurs when the bits replaced were originally *ones*, making this result too low by the amount $2^p - 1$. If the bits replaced were originally *zeros*, we will have incurred no error. We will use the notation $[0..2^p - 1]$ to describe this resultant error range that our substitution method produces.

This error model can be used to estimate the effective error of combining quantized values in arithmetic operators. To investigate the impact, we will discuss an adder and multiplier in greater detail.

A. Adder Error Model

An adder error model is shown in Fig. 1. The addition of two quantized values, $A_m 0_p + B_n 0_q$, results in an output, C , which has a total of $\max(M', N') + 1$ bits. Of these bits, $\min(p, q)$ of them are substituted zeros at the least-significant end. In an adder structure, the range of error for the output, C , is the sum of the error ranges of the two inputs, A and B . This gives us an output error range of $[0..2^p + 2^q - 2]$.

B. Multiplier Error Model

Just as we can derive an error model for the adder, we do the same for a multiplier. Again we have two quantized input values, $A_m 0_p * B_n 0_q$. These are multiplied together to form the output, C , which has a total of $M' + N'$ bits. Here, $p + q$ of them are substituted zeros at the least-significant end. This structure is shown in Fig. 2.

The output error is more complex in the multiplier structure than the adder structure. The input error ranges are the same,

$[0..2^p-1]$ and $[0..2^q-1]$ for $A_m 0_p$ and $B_n 0_q$, respectively. Unlike the adder, multiplying these two inputs together requires us to multiply the error terms as well, as shown in (1).

$$\begin{aligned} C &= A * B \\ &= (A - (2^p - 1)) * (B - (2^q - 1)) \quad (1) \\ &= AB - B(2^p - 1) - A(2^q - 1) + (2^p - 1)(2^q - 1) \end{aligned}$$

The first line of (1) indicates the desired multiplication operation between the two input signals. Since we are introducing errors into each signal, line two shows the impact of the error range of $A_m 0_p$ by subtracting $2^p - 1$ from the error-free input A . The same occurs for input B .

Performing a substitution of $E_p = 2^p - 1$ and $E_q = 2^q - 1$ into (1) yields the simpler (2):

$$\begin{aligned} C &= AB - BE_p - AE_q + E_p E_q \quad (2) \\ &= AB - (AE_q + BE_p - E_p E_q) \end{aligned}$$

From (2) we can see that the range of error resulting on the output C will be $[0..AE_q + BE_p - E_p E_q]$. That is to say, the error that the multiplication will incur is governed by the actual correct values of A and B , multiplied by the error attained by each input. In terms of maximum error, this occurs when we consider the maximum attainable value of the inputs multiplied by the maximum possible error of the inputs.

IV. HARDWARE MODELS

In the previous section we derived error models for adder and multiplier structures. Error is only one metric upon which we will base optimization decisions. Another crucial piece of information is hardware cost in terms of area.

By performing substitution rather than immediate truncation, we introduce a critical difference in the way hardware will handle this datapath. Unlike the case of immediate truncation, we do not have to change the implementation of downstream operators to handle different bit-widths on the inputs. Likewise, we do not have to deal with alignment issues, as all inputs to operators will have the same location of the binary point.

For example, in an adder, as we reduce the number of bits on the inputs, the area requirement of the adder decreases. The same relationship holds true when we substitute zeros in place of variable bits on an input. This is true because we can simply use wires to represent static zeros or static ones, so the hardware cost in terms of area is essentially zero.

If the circuit is specified in a behavioral fashion using a hardware description language (HDL), this optimization is likely to fall under the jurisdiction of vendor tools such as the technology mapper and the logic synthesizer. Fortunately, this constant propagation optimization utilizing wires is implemented in most current vendor tools.

In the next sections we outline the area models used to perform area estimation of our datapath. We will assume a simple 2-LUT architecture for our target FPGA and validate this assumption through implementation on target hardware.

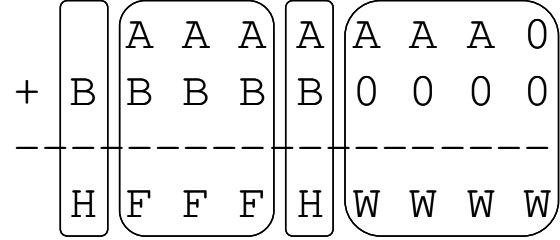


Fig. 3. Adder hardware requirements.

TABLE I
ADDER AREA

Number	Hardware
$\max(M' - N' , 0)$	half-adder
$\max(M', N') - \max(p, q) - M' - N' - 1$	full-adder
1	half-adder
$\max(p, q)$	wire

A. Adder Hardware Model

In a 2-LUT architecture, a half-adder can be implemented with a pair of 2-LUTs. Combining two half-adders together and an OR gate to complete a full-adder requires five 2-LUTs. To derive the hardware model for the adder structure as described in previous sections, we utilize the example shown in Fig. 3.

Starting at the least-significant side, all bit positions that overlap with zeros need only wires. The next most significant bit will only require a half-adder, as there can be no carry-in from any lower bit positions, as they are all wires. For the rest of the overlapping bit positions, we require a regular full-adder structure, complete with carry propagation. Finally, at the most-significant end, if there are any bits that do not overlap, we require half-adders to add together the non-overlapping bits with the possible carry-out from the highest overlapping full-adder bit.

The relationship described in the preceding paragraph is generalized into Table I, using the notation previously outlined. For the example in Fig. 3, we have the following formula to describe the addition.

$$\begin{aligned} &A_m 0_p + B_n 0_q \\ &m = 7, p = 1, n = 5, q = 4 \end{aligned}$$

This operation requires two half-adders, three full-adders, and four wires. In total, 19 2-LUTs.

B. Multiplier Hardware Model

We use the same approach to characterize the multiplier. A multiply consists of a multiplicand (top value) multiplied by a multiplier (bottom value). The hardware required for an array multiplier consists of AND gates, half-adders, full-adders, and wires. The AND gates form the partial products, which in turn are inputs to an adder array structure as shown in Fig. 5.

$$\begin{array}{r}
 \begin{array}{cccc} A & A & A & 0 \\ \times & B & B & 0 & 0 \\ \hline \end{array} \\
 \begin{array}{cccc} A0 & A0 & A0 & 00 \\ A0 & A0 & A0 & 00 \\ AB & AB & AB & 0B \\ + & AB & AB & 0B \\ \hline \end{array}
 \end{array}$$

Fig. 4. Multiplication example.

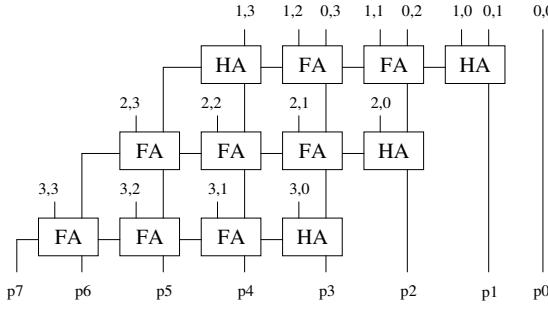


Fig. 5. Multiplication structure.

Referring to the example in Fig. 4, each bit of the input that has been substituted with a zero manipulates either a row or column in the partial product sum calculation. For each bit of the multiplicand that is zero, we effectively remove an inner column. For each bit of the multiplier that is zero, we remove an inner row. Thus:

$$\begin{aligned}
 & A_m 0_p * B_n 0_q \\
 & m = 3, p = 1, n = 2, q = 2
 \end{aligned}$$

is effectively a 3×2 multiply, instead of a 4×4 multiply. This requires two half-adders, one full-adder, and six AND gates, for a total of 15 2-LUTs. This behavior has been generalized into formulas shown in Table II.

C. Model Verification

To verify our hardware models against real-world implementations, we implemented both the adder and multiplier

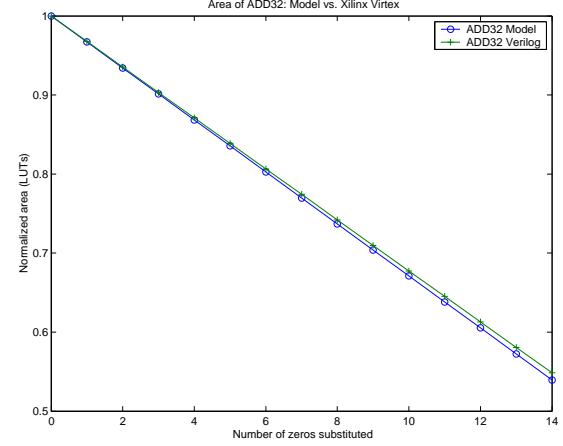


Fig. 6. Adder model verification.

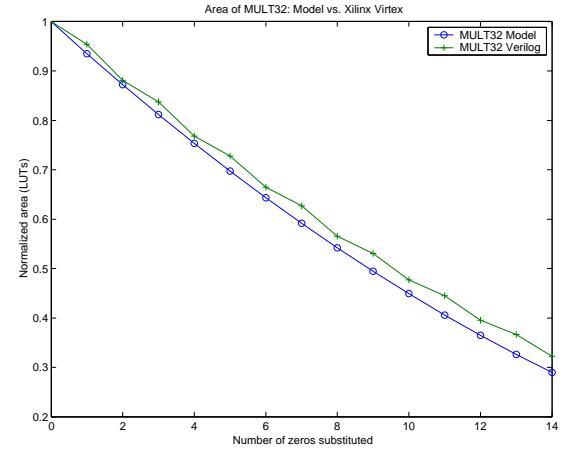


Fig. 7. Multiplier model verification.

structures in Verilog on the Xilinx Virtex FPGA using vendor-supplied place and route tools.

For the adder structure, we observe in Fig. 6 that our model closely follows the actual implementation area, being at worst within two percent of the actual Xilinx Virtex hardware implementation. The number of bits substituted was the same for each input at each data point.

The multiplier in Fig. 7 has a similar result to the adder, being at worst within 12 percent of the Xilinx Virtex implementation. These results support the use of our simple 2-LUT approximation of general island-style FPGAs to within a reasonable degree of accuracy.

V. OPTIMIZATION METHODS

Using the models described in the previous sections, we can now quantify the tradeoffs between area and error of various optimization methodologies.

A. The Nature of Error

Looking at the typical error introduced into a data path using the standard method of simple truncation, we see that the error

TABLE II
MULTIPLIER AREA

Number	Hardware
$\min(m, n)$	half-adder
$mn - m - n$	full-adder
mn	AND
$p + q$	wire

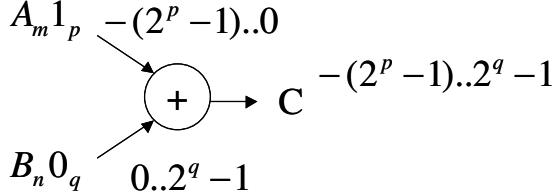


Fig. 8. Normalized error model of an adder.

is skewed, or biased, only in the positive direction. As we continue through datapath elements, if we maintain the same truncation policy to reduce the area requirement of our circuits, our lower-bound error will remain zero while our upper bound will continue to skew toward larger and larger positive values. This behavior also holds true for our own zero-substitution policy in Fig. 1 and Fig. 2.

This error profile does not coincide with our natural understanding of error. In most cases we consider the error of a result to be the *net distance from the correct value*, implying that the error term can be either positive or negative. Unfortunately, neither straight truncation nor our zero-substitution policy, as defined in previous sections, matches this notion of error. Fortunately, substituting constants for the least-significant bits allows us to manipulate their static values and capture this more intuitive behavior of error. We call this process “renormalization”.

B. Renormalization

It is possible for us to capture the more natural description of error with our method of zero-substitution because the least-significant bits are still present. We can use these bits to manipulate the resultant error range. An example of renormalization in an adder structure is shown in Fig. 8. We describe this method as “in-line renormalization” as the error range is biased during the calculation. It is accomplished by modifying one of the input operands with one-substitution instead of zero-substitution. This effectively flips the error range of that input around zero. The overall effect is to narrow the resultant error range, bringing the net distance closer to zero. Specifically, if the number of substituted zeros and ones are equal, we achieve an error range whose net distance from zero is half that if we were to use zero substitution only. If instead truncation were performed, no further shaping of the error range would be possible, leaving us with a positively skewed error range not consistent with our natural notion of error.

For example, in Fig. 1, a substitution of p, q zeros results in an error range of $[0..2^p + 2^q - 2]$. By using renormalization, this same net distance from the real value can be achieved with more bit substitutions, $p + 1, q + 1$, on the input. This will yield a smaller area requirement for the adder. Likewise, the substitution of p, q zeros with renormalization now incurs half the error on the output, $[-(2^p - 1)..2^q - 1]$, as shown in Fig. 8.

As with the adder structure, renormalization of the multiplier is possible by using different values for least-significant

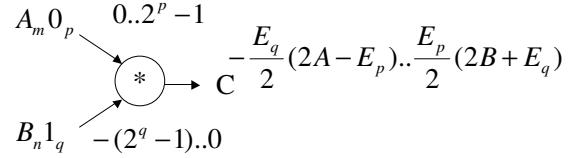


Fig. 9. Normalized error model of a multiplier.

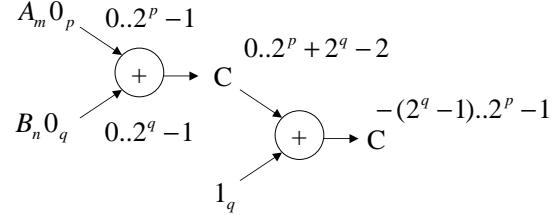


Fig. 10. Inserting a constant add performs an “active renormalization”.

bit substitution, yielding an error range that can be biased. Fig. 9 depicts a normalization centered on zero by substituting ones instead of zeros for input B . The derivation of the resultant error range is as follows in (3):

$$\begin{aligned}
 C &= (A - E_p)(B + E_q) \\
 &= AB + AE_q - BE_p - E_p E_q \\
 &= AB + AE_q - (BE_p + E_p E_q) \\
 &= AB + AE_q - \frac{E_p E_q}{2} - \left(BE_p + \frac{E_p E_q}{2} \right) \\
 &= AB + \frac{E_q}{2}(2A - E_p) - \frac{E_p}{2}(2B + E_q)
 \end{aligned} \tag{3}$$

Another method of renormalization can be accomplished after an operation, or operations, have been completed. By inserting a constant addition, we can accomplish a very similar biasing of error range, this time referred to as “active renormalization”. An example is shown in Fig. 10.

C. Renormalization Area Impact

The benefits of renormalization can come very cheaply in terms of area for the “in-line” method. Our adder structure example in Fig. 3 originally requires 19 2-LUTs and has an error range of $[0..16]$. We can achieve a completely negative bias of $[-16..0]$ without an area penalty by modifying the structure of the least-significant half-adder to have a constant carry-in of 1. At the 2^4 bit position, this effectively adds 16 to the addition without incurring an area penalty. This has the same effect as using the “active renormalization”, where an explicit addition is performed to change the error bias of the datapath. Alternatively, if we wanted to balance the error, we could achieve an error range of $[-8..8]$ by doing the same thing but at one bit position lower, 2^3 . Unfortunately, since there is no existing half-adder hardware to modify for this bit position, we must create a half-adder structure at the 2^3 bit position to add together the value from input A and a constant “1”. We also must change the existing half-adder at the 2^4 position into a full-adder to compensate for the possibility of

a carry-out from the newly added half-adder. Together, this increases the area requirement of this adder by 5 2-LUTs.

Finally, we can do a smaller renormalization by substituting a “1” for one of the least-significant bits on one of the inputs. This would yield an output error range of $[-1..15]$. While not particularly biased, it doesn’t incur any area penalty as the newly substituted “1” lines up with a zero from the other input, requiring no computational hardware.

Even when substituted ones and zeros on the inputs completely overlap, consideration must be made for downstream operations, as we now have ones in the least-significant bit positions which may need to be operated upon in subsequent operations. This may adversely impact the overall area of the circuit, at which point “active” renormalization should be considered as an alternative that can be implemented cheaply later in the datapath to “fix up” the error range using a constant bias.

The behavior of renormalization in multiplier structures is equally interesting. As can be seen in Fig. 4, zeros substituted at the least-significant end of either the multiplier or the multiplicand “fall” all the way through to the result. For the multiplication $A_m 0_p * B_n 1_q$, p zeros will be present at the least-significant end of the result. With this behavior, we can obtain a renormalized error result while still providing zero-substituted bit positions that will not have to be operated upon in downstream operations. This is important in providing opportunities for area savings throughout the datapath. As with the adder structure, we pay a penalty for this renormalization. For the multiplier, we must put back an inner row and column for each one-substitution present in the multiplier and multiplicand, respectively.

Finally, active renormalization has an area penalty. As it is simply an addition between an input value and a constant positive bias, the impact is simply the area requirement of the biasing adder.

D. Alternative Arithmetic Structures

As discussed in previous sections, our zero-substitution method for multipliers gives a reduced area footprint at the cost of increased error in the output over an exact arithmetic multiplication. An alternative to this method of area/error tradeoff is one described in [8]. This work, and the work of others ([9], [10]), focuses on removing a number of least-significant columns of the partial-product array.

As described in [9], by removing the n least-significant columns from an array-multiplier multiplication, we save (for $n \geq 2$) $\frac{n(n+1)}{n}$ AND gates, $\frac{(n-1)(n-2)}{2}$ full adders, and $(n-1)$ half adders. The column removal is depicted in Fig. 11. This method has a different area-to-error tradeoff profile, and is shown in Fig. 12 for a 32-bit multiplier.

While the truncated multipliers have a more favorable area-to-error profile, one drawback in their use is that they require the full precision of both operands to be present at the inputs of the multiplier. This has the effect of requiring higher precision on upstream computations, possibly negating the area gain at a particular instance of a multiplier by requiring larger

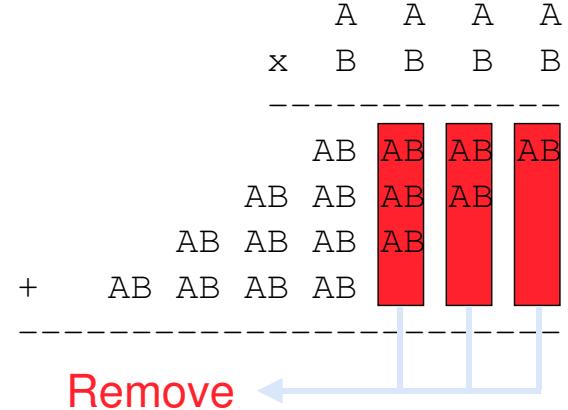


Fig. 11. A truncated multiplier removes least-significant columns from the partial product array.

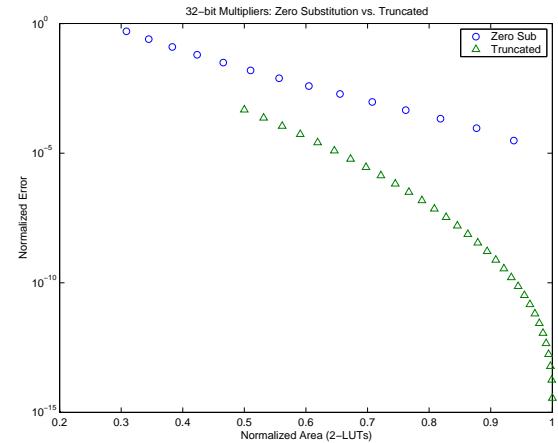


Fig. 12. Error to area profile of zero-substitution 32-bit multiplier and truncated 32-bit multiplier.

operations at upstream nodes. This makes it more valuable in multiplications closer to the inputs than those closer to the outputs.

VI. AUTOMATED OPTIMIZATION

We have presented in the previous section several optimization methods designed to allow more control of the area/error profile of our datapath. Unfortunately, due to the strongly interconnected nature of datapaths and dataflow graphs in general, it is hard to analytically quantify the impact of each method on the overall profile of the system. Making a small change, such as increasing the number of zero-substituted bits at a particular primary input, will impact the breadth of possible optimizations available at every node.

Fortunately, we have provided a model that can accurately estimate the area and error of each node within the datapath. With these measurements and optimization “moves”, we can utilize simulated annealing [11] to choose how to use our palette of optimizations to achieve an efficient implementation area under a user-specified error constraint. We have developed

an automated approach using simulated annealing principles similar to those found in [12] to area-optimize a dataflow graph. Simulated annealing has shown to produce good results on often intractable problems, and is a good candidate for our design challenge.

The possible moves in our system are the various optimization methods. At each temperature we choose randomly between altering the amount of zero-substitution at the inputs and changing multiplier structures. Our cost function for determining the quality of moves is determined by the area estimate of the entire datapath combined with a user-specified error constraint. This error constraint is identified as an error range at a particular node, dubbed the error node. Our cost function is defined in (4), where $error$ is the absolute value of the difference between the maximum error and the target error at the error node. We have determined through experimentation that $\beta = 0.25$ gives a good balance between an area efficient implementation and meeting the error constraint.

$$cost = \beta * area + (1 - \beta) * error \quad (4)$$

When modifying an input, we allow the annealer to randomly choose to increase or decrease the number of bits substituted with constants by one bit. Thus, an input A_{502} can move to A_{601} or A_{403} .

When modifying the structure of a multiplier, we randomly choose a multiplier and adjust its degree of truncation. As with the inputs, we allow the annealer to increase or decrease by one the number of columns truncated from the partial product array. This allows a smooth transition from the traditional array multiplier to a highly-truncated multiplier.

After the move has been completed, we perform a greedy renormalization. Recalling from previous sections, there are several instances where the effect of renormalization can be achieved without an area impact. For each adder that may be renormalized without area penalty, we perform renormalization and observe the impact on the error node of interest. The adder that exhibits the most reduction in maximum error at the error node through renormalization is renormalized. This process is repeated until either our list of candidate adders is exhausted, or there can be no error improvement through renormalization. After the annealer has finished, we optionally apply active renormalization at the error node if it yields a lower overall implementation cost.

VII. EXPERIMENTAL RESULTS

We have implemented our automated optimization techniques as a subset of our design-time tool presented in [1]. To test the effectiveness of our methodologies, we have used our technique to optimize several benchmark image processing kernels. These include a matrix multiply, wavelet transform, CORDIC, and a one-dimensional discrete cosine transform.

A typical use of our methods would begin with the user performing basic truncation. As mentioned before, while basic truncation does afford an area savings throughout the datapath, there is very little guidance as to which inputs to manipulate, and how changes might affect the overall performance of

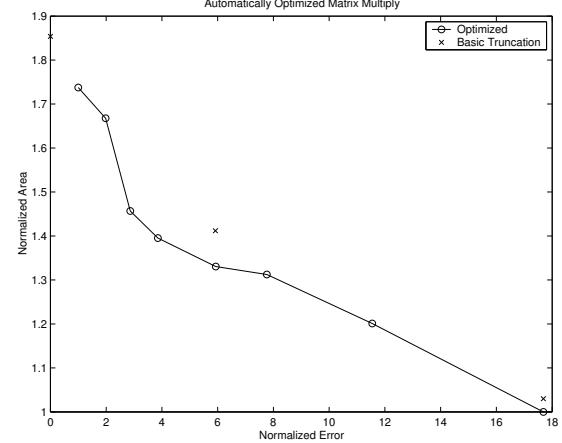


Fig. 13. Optimized results for matrix multiply.

the implementation. The starting points we have used in our experiments are truncating zero, one, and two bits from every input. These can be seen in Figs. (13-18) as the “Basic Truncation” points on the plots.

From these initial estimates of area and error, we performed the automated optimization using these points as guidelines for error constraints. The flexibility of our methods allows us to choose any error constraint, giving us far more area/error profiles to consider for implementation. As can be seen in the plots, the automated optimization method is able to obtain better area/error tradeoffs than the basic truncation method, except in a few cases in the wavelet transform and 1-D discrete cosine transform. We attribute this to the need for further tuning of some of the parameters in our simulated annealing algorithm. In particular, tuning the β parameter to adjust the weighting of meeting the error constraint vs. obtaining an area-efficient datapath. In the future, perhaps this parameter could be influenced by the user.

Careful observation will note a difference in performance between Figs. (15,16) and Figs. (17,18). In the experiments for the latter figures, we performed a slightly different experiment to determine whether or not our tool would be able to more aggressively optimize a single “precision critical path” in a circuit. In both the CORDIC and DCT, there were several output nodes to be considered. In our experiments for (17,18), we only constrained the error on one output node. From the plots it can be seen that the tool was able to maintain the desired precision at the output nodes of interest while finding more area efficient implementations. This type of optimization can be very useful when the developer is aware of varying degrees of precision required at the outputs.

VIII. CONCLUSIONS AND FUTURE WORK

We have described and motivated the need to investigate the optimization of the least-significant bit position. In order to do so, we have proposed models of area and error for an alternative area reduction technique to straight truncation—constant substitution. Using this method and models, we have

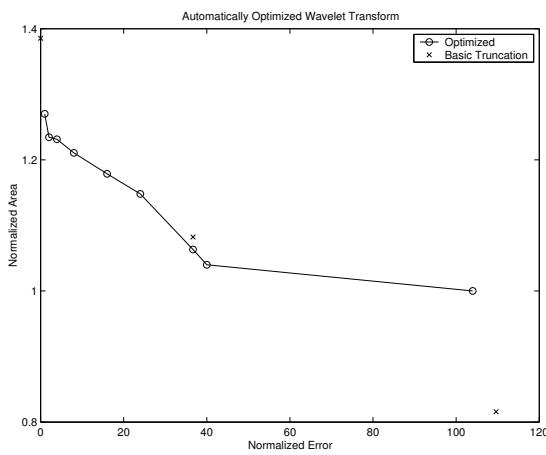


Fig. 14. Optimized results for wavelet transform.

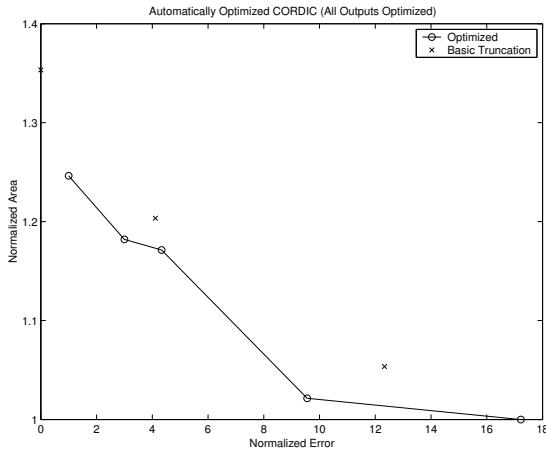


Fig. 15. Optimized results for CORDIC, all outputs optimized.

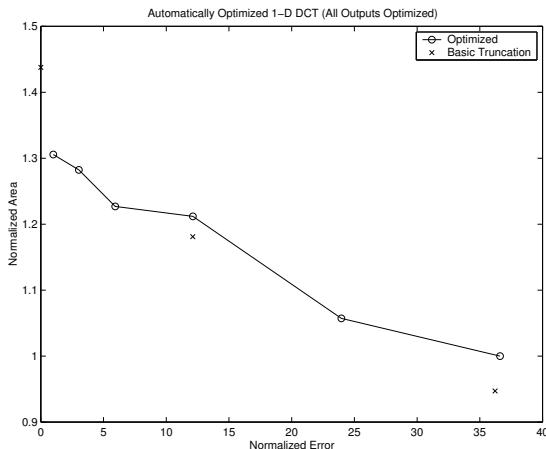


Fig. 16. Optimized results for 1-D discrete cosine transform, all outputs optimized.

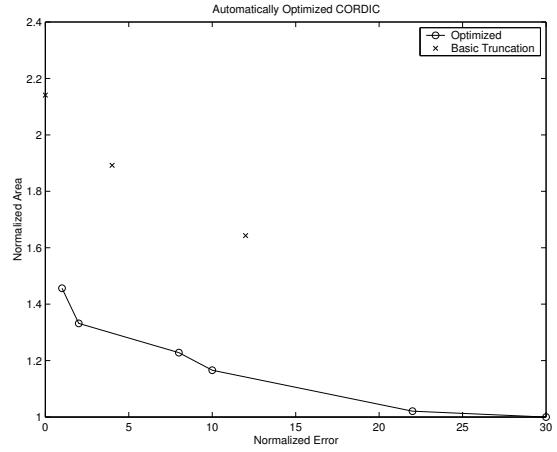


Fig. 17. Optimized results for CORDIC, single output selected for optimization.

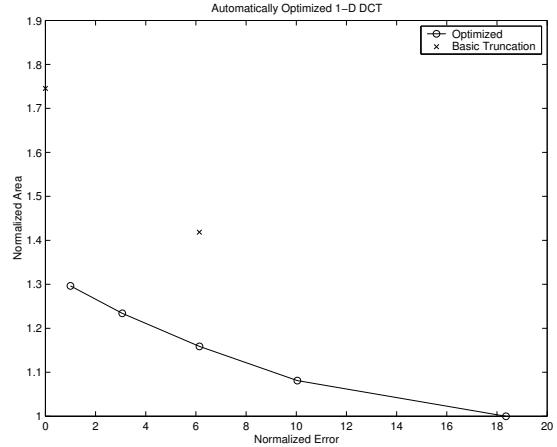


Fig. 18. Optimized results for 1-D discrete cosine transform, single output selected for optimization.

proposed several optimization techniques aimed at giving the developer more control over the area-to-error tradeoff during datapath precision optimization that would not be available if simple truncation were used. We have proposed techniques for area-efficient renormalization, allowing us to more effectively capture our intuitive notion of error. We have introduced the use of alternative arithmetic structures, such as the truncated multiplier, in datapath optimization. Finally, we have implemented our techniques in an automated tool that is able to optimize a datapath subject to a user-supplied error constraint. More importantly, our techniques and tools give the user a broader range of options to consider, as well as a mechanism to achieve specific area/error targets when performing implementations.

In future work, we will incorporate more optimizations to further expand the design space. We will implement more of the renormalization techniques presented here in our automated tool. This will require a more comprehensive renormalization routine that will attempt the transformations that may

increase the cost of a design. We hope to incorporate further alternative structures, such as floating-point and pseudo-floating-point to allow for high-precision (and high-area) portions of the datapath to be realized.

REFERENCES

- [1] M. L. Chang and S. Hauck, "Précis: A design-time precision analysis tool," in *IEEE Symposium on Field-Programmable Custom Computing Machines*, 2002, pp. 229–238.
- [2] M. Stephenson, J. Babb, and S. Amarasinghe, "Bitwidth analysis with application to silicon compilation," in *Proceedings of the SIGPLAN conference on Programming Language Design and Implementation*, June 2000.
- [3] M. W. Stephenson, "Bitwise: Optimizing bitwidths using data-range propagation," Master's thesis, Massachusetts Institute of Technology, May 2000.
- [4] W. Sung and K.-I. Kum, "Simulation-based word-length optimization method for fixed-point digital signal processing systems," *IEEE Transactions on Signal Processing*, vol. 43, no. 12, pp. 3087–3090, December 1995.
- [5] S. Kim, K.-I. Kum, and W. Sung, "Fixed-point optimization utility for C and C++ based digital signal processing programs," in *Workshop on VLSI and Signal Processing*, Osaka, 1995.
- [6] A. Nayak, M. Haldar, et al., "Precision and error analysis of MATLAB applications during automated hardware synthesis for FPGAs," in *Design Automation & Test*, March 2001.
- [7] G. A. Constantinides, P. Y. Cheung, and W. Luk, "The multiple wordlength paradigm," in *IEEE Symposium on Field-Programmable Custom Computing Machines*, 2001.
- [8] Y. Lim, "Single-precision multiplier with reduced circuit complexity for signal processing applications," *IEEE transactions on Computers*, vol. 41, no. 10, pp. 1333–1336, October 1992.
- [9] M. J. Schulte and J. Earl E. Swartzlander, "Truncated multiplication with correction constant," in *VLSI Signal Processing VI, IEEE Workshop on VLSI Signal Processing*, October 1993, pp. 388–396.
- [10] K. E. Wires, M. J. Schulte, and D. McCarley, "FPGA resource reduction through truncated multiplication," in *Proceedings of the 11th International Conference on Field Programmable Logic and Applications*, August 2001, pp. 574–583.
- [11] S. Kirkpatrick, J. C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, May 13 1983.
- [12] V. Betz and J. Rose, "VPR: A new packing, placement and routing tool for FPGA research," in *Proceedings of the Seventh International Workshop on Field-Programmable Logic and Applications*, 1997, pp. 213–222.

Précis: A Design-Time Precision Analysis Tool

Mark L. Chang and Scott Hauck

Department of Electrical Engineering

University of Washington, Seattle, WA

{mchang,hauck}@ee.washington.edu

Abstract

Currently, few tools exist to aid the FPGA developer in translating an algorithm designed for a general-purpose processor into one that is precision-optimized for FPGAs. This task requires extensive knowledge of both the algorithm and the target hardware. We present a design-time tool, Précis, which assists the developer in analyzing the precision requirements of algorithms specified in MATLAB. Through the combined use of simulation, user input, and program analysis, we demonstrate a methodology for precision analysis that can aid the developer in focusing their manual precision optimization efforts.

1. Introduction

One of the most difficult tasks in implementing an algorithm in an FPGA-like substrate is dealing with precision issues. Typical general-purpose processor concepts such as *word size* and *data type* are no longer valid in the FPGA world, which is dominated by finer-grained computational structures, such as look-up tables. Instead, the designer must use and implement bit-precise data paths.

More specifically, in a general-purpose processor, algorithm designers can typically choose from a predefined set of variable types that have a fixed word length. Examples of these predefined types are the C data types such as `char`, `int`, `float`, `double`. These data types correspond to specific memory storage sizes, and subsequently, into different ways of handling operations upon these memory locations within the microprocessor. Much of the work of padding, word-boundary alignment, and operation selection is hidden from the programmer by compilers and assemblers, which make the use of one data type equally easy as another.

In contrast, an FPGA does not have predefined data widths for its data path. Instead, designers must provide all the structures necessary to handle operations on different data widths and types. Therefore, it is paramount

that FPGA designers implement their algorithms such that they utilize resources efficiently and accurately. Too many bits allocated to a particular operation is wasteful, while too few can result in erroneous output.

The difficulty is in the translation of an initial algorithm into one that is precision-optimized for FPGAs. This task requires extensive knowledge of both the algorithm and the target hardware. Unfortunately, there are few tools that aid the would-be FPGA developer in this translation. In this paper, we discuss our work in filling that gap by introducing a developer-oriented tool for the design-time analysis of the impact of precision on algorithm implementation.

2. Background

Currently, the typical tool flow for development of an FPGA-targeted algorithm is as shown in Figure 1.

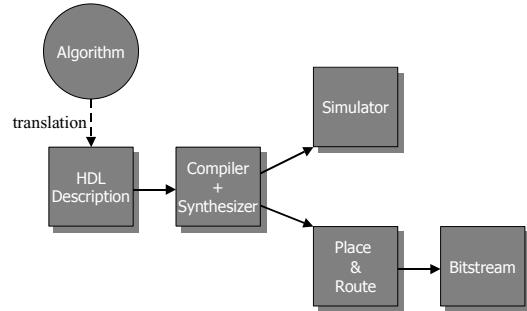


Figure 1. Typical tool flow for implementing a high level language specified algorithm on an FPGA.

At the head of the development chain is the algorithm. Often, the algorithm under consideration has been implemented in some high-level language, such as MATLAB, C, or Java, targeted to run on a general purpose processor, such as a workstation or desktop personal computer. The most compelling reason to utilize a high level language running on a workstation is that it provides infinite flexibility and a comfortable, rich environment in which to rapidly prototype algorithms. Of course, the reason one would convert this algorithm into a hardware implementation is to gain considerable

advantages in terms of speed, size, and power.

This tool flow requires the developer to first convert a software prototyped algorithm into a hardware description. From this hardware description language (HDL) specification, various stages and intermediate tools are used to perform simulation and generate target bitstreams, which are then executed on reconfigurable logic. As mentioned earlier, one of the more difficult steps in implementing the algorithm in hardware is highlighted in Figure 1 with a dashed arrow – the conversion from a high-level software language, such as C, Java, or MATLAB, into an HDL description.

A simple conversion without precision analysis would most likely yield an unreasonably large hardware implementation. For example, by blindly choosing a fixed 32-bit data path throughout the system, the developer may encounter two problems: wasted area and incorrect results. The former arises when the actual data the algorithm operates upon does not require the full 32-bit data path. In this case, much of the area occupied by the oversized data path could be pruned. There are several benefits to area reduction of a hardware implementation: reduced power consumption, reduced critical path delay, and the increased probability of parallelism by freeing up more room on the device to perform other operations simultaneously. On the other hand, the latter case occurs when the algorithm actually requires more precision for some data sets than the 32-bit data path provides. In this case, the results obtained from the algorithm could potentially be incorrect due to unchecked overflow or underflow conditions.

Therefore, within the HDL description, it is important that the developer determine more accurate bounds on the data path. Typically, this involves running a software implementation of the algorithm with representative data sets and performing manual fixed-point analysis. At the very least, this requires the re-engineering of the software implementation to record the ranges of variables throughout the algorithm. From these results, the developer could infer candidate bit-widths for their hardware implementation. Even so, these methods are tedious and often error-prone.

Unfortunately, while many of the other stages of hardware development have well-developed tools to help automate difficult tasks, few tools can automate HDL generation from a processor-oriented higher level language specification. And while there are C-to-Verilog[1] and C-to-VHDL[2] tools in existence, they do not offer such “designer aids” that would help with precision analysis of existing algorithms implemented in a high level language.

3. Précis

In order to fill this void in hardware development tools, we are developing *Précis*, a design-time precision analysis

tool. *Précis* utilizes MATLAB as an input specification for algorithms and is designed to interact with the developer in order to assist them in making the best choices regarding data path precision. Currently, *Précis* aids the developer by providing a constraint propagation engine, simulation support, range finding capabilities, and performing precision slack analysis.

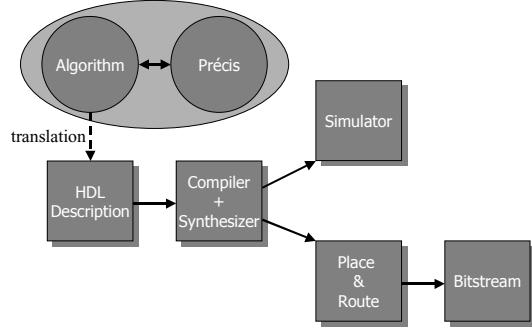


Figure 2. *Précis*' role in the tool chain.

Précis is designed to complement the existing tool flow in the manner shown in Figure 2. *Précis* is not meant to be an HDL generator, a MATLAB-to-HDL converter, or an optimizing compiler of any sort. Instead, it is meant to provide a convenient way for the user to interact with the algorithm under consideration. Our goal is for the knowledgeable user, after interacting with our tool, to have a much clearer idea of the precision requirements of their data path. It is our belief that the developer of the algorithm, with suitable software assistance, can perform much better precision analysis and optimization than a fully automated tool could ever achieve. In the following sections, we describe in more detail the constituent parts of *Précis*.

3.1. MATCH front-end

The front-end of *Précis* comes from Northwestern University in the form of a modified MATCH compiler[3,4]. The MATCH compiler understands a subset of the MATLAB language and can transform it into efficient implementations on FPGAs, DSPs, and embedded CPUs. It is used here primarily as a pre-processor to parse MATLAB codes. The MATCH compiler was chosen as the basis for the MATLAB code parsing because no official grammar is publicly available for MATLAB. We are not constrained to using the MATCH compiler, though, as our tool may be updated to accommodate an alternate MATLAB-aware parser.

MATLAB was chosen as the target high level language because the researchers involved in this work also contribute to the MATCH project at Northwestern University. From this work, it has become clear that MATLAB is a strong favorite for algorithm prototyping

and exploration, especially among scientists that might have little to no hardware design expertise. With the proliferation of reconfigurable co-processor boards capable of providing great speedups to many classes of algorithms, it would be advantageous to provide tools to help these same scientists target their MATLAB algorithms to FPGAs. Précis can be used both by developers prototyping in MATLAB before hand converting to an HDL, or to develop pragmas (designer hints) for MATCH's automatic compilation.

The MATCH compiler remains a work in progress and is currently being marketed by AccelChip[5]. For our purposes, we have modified the base MATCH compiler to generate a non-hierarchical (flattened) representation of parsed MATLAB code from its internal abstract syntax tree. This representation is then read into the main *Précis* tool for display and user interaction.

3.2. Précis application

The main Précis application is written in Java, in part, due to its relative platform independence and ease of graphical user interface creation. Précis takes the parsed MATLAB code output generated from the MATCH compiler and displays a GUI that formats the code into a tree-like representation of statements and expressions. An example of the GUI in operation is shown in Figure 3. The left half of the interface is the tree representation of the MATLAB code. The user may click on any node and, depending on the node type, receive more information in the right panel. The right panel displayed in the figure is an example of the entry dialog that allows the user to specify fixed-point precision parameters, such as range and type of truncation. With this graphical display the user can then perform the various tasks described in the following sections.

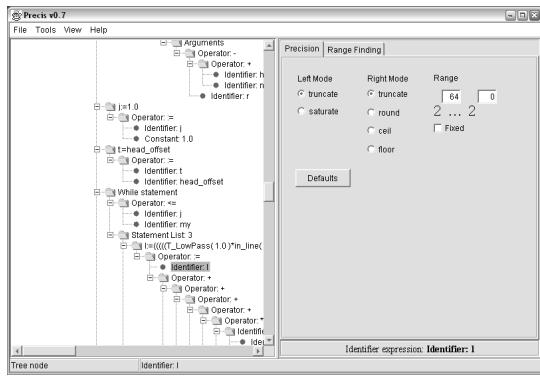


Figure 3. Screen capture of the Précis GUI.

3.3. Propagation engine

A core component of the Précis tool is a constraint

propagation engine. The propagation engine simulates the effects of using fixed-point numbers and fixed-point math in hardware. This is done by allowing the user to (optionally) constrain variables to a specific precision by specifying the bit positions of the most significant bit (MSB) and least significant bit (LSB). Variables that are not manually constrained begin with a default width of 64 bits. Typically, a user should be able to provide constraints easily for at least the circuit inputs and outputs.

The propagation engine traverses the expression tree and determines the resultant ranges of each operator expression from its child expressions. This is done by implementing a set of rules governing the change in resultant range that depend upon the input operand(s) range(s) and the type of operation being performed. For example, in the statement $a = b + c$, if b and c are both constrained by the user to a range of 2^{15} to 2^0 , 16 bits, the resulting output range of a would have a range of 2^{16} to 2^0 , 17 bits, as an addition conservatively requires one additional high order bit for the result in the case of a carry-out from the highest order bit. Similar rules apply for all supported operations.

The propagation engine works in this fashion across all statements of the program, recursively computing the precision for all expressions in the program. This form of propagation is often referred to as value-range propagation. One shortcoming of the currently implemented propagation engine is that it does not handle loop carried variables or conditional branches. This is to be rectified in later revisions of the tool. A more complete study of propagation and its effects upon hardware synthesis can be found in [6]. We plan to continue development of our own propagation tool to a similar extent in the near future.

An example of forward and backward propagation is depicted in Figure 4

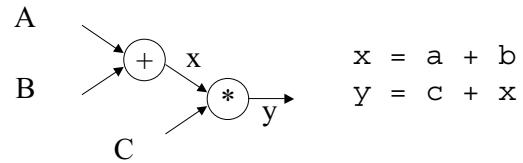


Figure 4. Simple propagation example.

In this trivial example, assume the user sets all input values (a , b , c) to utilize the bits $[15,0]$, i.e. have a range from $2^{16}-1$ to 0 . Forward propagation would result in x having a bit range of $[16, 0]$ and c having a range of $[31, 0]$. If, after further manual analysis, the user notes that the output from these statements should be constrained to a range of $[10, 0]$, backwards propagation following forward propagation will constrain the inputs (c and x) of the multiplication to $[10, 0]$ as well. Propagating yet further, this constrains the input variables a and b to

the range [10, 0] as well. Obviously, these are very conservative propagation values. Knowing strict values for the variables would increase our accuracy, as can be shown in [6].

The propagation engine can be used to get a quick estimate of the growth rate of variables through the algorithm. This is done by constraining the precision of input variables and a few operators and performing the propagation. This will allow the user to see a conservative estimate of how the input bit width affects the size of operations down stream.

While the propagation engine will provide some information as to the effects of fixed-point operations on the resultant data, it is at best a conservative estimate. It would be appropriate to consider the bit widths determined from the propagation engine to be worst-case results, or in other words, an upper bound. This upper bound will become useful in further analysis phases of Précis.

3.4. Simulation support

As previously mentioned, a typical step in precision analysis is the actual running of the algorithm in a fixed-point environment. Précis can automatically generate annotated MATLAB code to aid in fixed-point simulation of the user's algorithm. The user simply selects variables to constrain and requests that MATLAB simulation code be generated. The code generated by the tool includes calls to MATLAB helper functions that we developed to simulate a fixed-point environment. The simulation flow is shown in Figure 5.

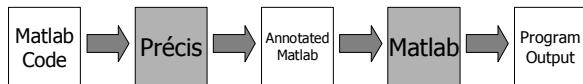


Figure 5. Code generation for simulation.

In particular, a MATLAB support routine, “`fixp`” was developed to simulate a fixed-point environment. Its declaration is `fixp(x,m,n,lmode,rmode)`, where ‘`x`’ denotes the signal to be truncated to ‘`(m-n+1)`’ bits in width. Specifically, ‘`m`’ denotes the MSB bit position and ‘`n`’ the LSB bit position, inclusively, with negative values representing positions to the right of the decimal point. The remaining two parameters, ‘`lmode`’ and ‘`rmode`’ specify the method desired to deal with overflow at the MSB and LSB portions of the variable, respectively. These modes correspond to different methods of hardware implementation. Possible choices for ‘`lmode`’ are `sat` and `trunc`—saturation to $2^{(MSB+1)-1}$ and truncation of all bits above the MSB position, respectively. For the LSB side of the variable, there are four modes, `round`, `trunc`, `ceil`, and `floor`. `Round` rounds the result to the nearest integer, `trunc` truncates

all bits below the LSB position, `ceil` rounds up to the next integer level, and `floor` rounds down to the next lower integer level. These modes correspond exactly to the MATLAB functions with the exception of `trunc`, and thus behave as documented by Mathworks. `Trunc` is accomplished through the modulo operation. An example of output generated for simulation is shown in Figure 6.

MATLAB Input	Annotated MATLAB
<code>a = 1;</code>	<code>a=1;</code>
<code>b = 2;</code>	<code>b=2;</code>
<code>c = 3;</code>	<code>c=3;</code>
<code>d = (a+(b*c));</code>	<code>d=(fixpp(a,12,3,'trunc','trunc') + (b*c));</code>

Figure 6. Sample output generated for simulation, with the range of a variable constrained.

After the user has constrained the variables of interest and indicated the mechanism by which to control overflow of bits beyond the constrained precision, Précis can generate annotated MATLAB. The user can then run the generated MATLAB code with real data sets. The purpose of these simulations is to determine the effects of constraining variables on the correctness of the implementation. Not only might the eventual output be erroneous, but the algorithm may also fail to operate entirely due to the effects of precision constraints.

If the user finds the algorithm’s output to be acceptable, they might consider constraining additional key variables, thereby further reducing the eventual size of the hardware circuit. On the other hand, if the output generates unusable results, the user knows then that their constraints were too aggressive and that they should increase the precision of some of the constrained variables. Note that it is typically not sufficient to merely test whether the fixed precision results are identical to the unconstrained precision results, since this is too restrictive. In situations such as image processing, lossy compression, and speech processing, users may be willing to trade some result quality for a more efficient hardware implementation. Précis, by being a designer assistance tool, allows the designer to create their own “goodness” function, and make this tradeoff as they see fit. With the Précis environment, this iterative development cycle is shortened, as the fixed-point simulation code can be quickly generated.

3.5. Range finding

While the simulation support described above is very useful on its own for fixed-point simulation, it is only truly useful if the user can accurately identify the variables that they feel can be constrained. If the user does not really have an idea of where to begin, one place to start is utilizing the Précis range finding capability. The development cycle utilizing range finding is shown in

Figure 7.

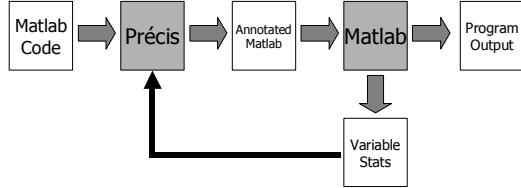


Figure 7. Development cycle for range finding analysis.

After the MATLAB code is parsed into the tool, the user can select variables they are interested in monitoring. Variables are targeted for range analysis and annotated MATLAB is generated, much like the simulation code is generated in the previous section. Instead of fixed-point simulation, though, Précis annotates the code with another MATLAB support routine that monitors the range of the values that the variables under question take on.

This support routine, ‘rangefind’, monitors the maximum and minimum values attained by the variables. The annotated MATLAB is run with some sample data sets to gather range information on the variables under consideration. The user can then save these values in data files that can be fed back into Précis with another routine, ‘saverangefind’. Example range finding output is shown in Figure 8.

MATLAB Input	Range Finding Output
<code>a = 1;</code>	<code>a=1;</code>
<code>b = 2;</code>	<code>b=2;</code>
<code>c = 3;</code>	<code>c=3;</code>
<code>d = (a+(b*c));</code>	<code>d=(a+(b*c));</code>
	<code>rangeFind(d, 'rfv_d');</code>

Figure 8. Sample range finding output.

The user then loads the resultant range values discovered by rangefind back into the Précis tool and (optionally) constrains the variables. The user now has an idea of what precision each variable requires for the sample data. Propagation can now be performed to determine the effect these precisions have on the rest of the system. Another useful step that the user can perform is to constrain the variables under question even further and perform a simulation to see how much error it introduces into the output. The results from this range finding method, however, are data set dependent. If the user is not careful to use representative data sets, the final hardware implementation could still generate erroneous results if the data sets were significantly different in precision requirements, even on the same algorithm.

For this reason we will consider range-gathered precision information to be somewhat of a lower bound. Given that the precisions obtained from propagation are

conservative estimates, or an upper bound, manipulating the difference between these two bounds leads us to another method of precision analysis—slack analysis.

4. Slack analysis

One of the goals of this tool is to provide the user with “hints” as to where the developer’s manual precision analysis and hardware tuning efforts should be focused. Ultimately, it would be extremely helpful for the developer to be given a list of “tuning points” in decreasing order of potential overall reduction of circuit size. This way, the developer could start a hardware implementation using more generic data path precision and iteratively optimize code sections that would give them the most benefit to meet constraints, such as time, cost, area, performance, or power. We believe this type of “tuning list” would give a developer a head start on precision analysis and put them on the right path of development faster than non-automated techniques.

As mentioned earlier, if the user performs range finding analysis and propagation analysis on the same set of variables, the tool would obtain what would amount to a lower bound from range analysis and an upper bound from propagation. We consider the range analysis a lower bound because it is the result of true data sets. While other data sets may require even lower amounts of precision, we know we need *at least* the ranges gathered from the range analysis. Further testing with other data sets may show that some variables would require more precision. Thus, if we implement the design with the precision found, we might encounter errors on output, thus the premise that this is a lower bound.

On the other hand, propagation analysis is very conservative. For example, in the statement $a=b+c$, where b and c have been constrained to be 16 bits wide by the user, the resultant bit width of a may be *up to* 17 bits due to the addition. But in reality, both b and c may be well within the limits of 16 bits and an addition might never overflow into the 17th bit position. For example, if $c=\lambda-b$, the range of values a could ever take on is governed by λ . To a person investigating section of code, this seems very obvious when c is substituted into $a=b+c$, but these types of more “macroscopic” constraints in algorithms can be difficult or impossible to find automatically. It is because of this that we can consider propagated range information to be an upper bound.

Given a lower and upper bound on the bit width of a variable, we can consider the difference between these two bounds to be the slack. The actual precision requirement is most likely to lie between these two bounds. Manipulating the precision of nodes with slack can net gains in precision system-wide, as changes in any single node may impact many other nodes within the

circuit. The reduction in precision requirements and the resultant improvements in area, power, and performance can be considered gain. Through careful analysis of the slack at a node, we can calculate how much gain can be achieved by manipulating the precision between these two bounds. Additionally, by performing this analysis independently for each node with slack, we can generate an ordered list of “tuning points” that the user should consider.

For this paper, we consider the reduction of the area requirement of a circuit to be gain. In order to compute the gain of a node with respect to area, power and performance, we need to develop basic hardware models to capture the effect of precision changes upon these parameters. One simple implementation that we have utilized is to provide simple weighting parameters for different operator types. Thus, for example, if an adder has an area model of x , it indicates that as the precision decreases by one bit, the area reduces linearly and the gain increases linearly. In contrast, a multiplier has an area model of x^2 , indicating that the area reduction and gain achieved are proportional to the square of the word size. Intuitively, this would give a higher overall gain value for bit reduction of a multiplier than of an adder. Using these parameters, our approach can more effectively choose the nodes with the most possible gain to suggest to the user. We detail our methodology in the next section.

4.1. Performing slack analysis

The goal of slack analysis is to identify which nodes, when constrained by the user, are likely to have the greatest impact upon the overall circuit area. While we do not believe it is realistic to expect users to constrain all variables, most users would be able to consider how to constrain a few “controlling” values in the circuit.

Our method seeks to efficiently use designer time by guiding them to the next important variables to consider for constraining. Précis can also provide a stopping criterion for the user: we can measure the maximum possible benefit from future constraints by constraining all variables to their lower bounds. The user can then decide to stop further investigation when the difference between the current and “lower bound” areas is no longer worth further optimization.

Our methodology is straightforward. For each node that has slack, we set the precision to the range-find value, the lower bound. Then, we propagate the impact of that change over all nodes and calculate the overall gain for the change, system-wide. We record this value as the effective gain as a result of modifying that node. We then reset all nodes and repeat for the remaining nodes that have slack. We then order the resultant list of gain values in decreasing order and present this information to the user in a dialog window. The user then can see which nodes to change to get the highest gain and in what order.

It is then up to the designer to consider these nodes and determine which, if any, should actually be more tightly constrained.

To further illustrate this analysis method, refer to the pseudo-code shown below.

Algorithm: Slack Analysis

```
User Step #1: Constrain known variables
User Step #2: Perform propagation
User Step #3: Load range data for some set of variables
'n'

set list_of_gains to empty list
for each variable 'm' in 'n'
    set aggregate_gain = 0
    constrain range of 'm' to the range analysis value
    perform forward and reverse propagation over all
variables
    for all variables
        if range of variable is narrower than range originally
propagated in 'User Step #2'
            set aggregate_gain += old_area - new_area
        end
    next
    add (variable 'm' and aggregate_gain) to list_of_gains
    for all variables
        reset range of variable to range originally propagated in
'User Step #2'
    next
next

sort(list_of_gains) by decreasing aggregate_gain
```

5. Benchmarks

In order to determine the effectiveness of our slack analysis methodology, we allowed the tool to perform slack analysis with propagated and range-found range values. To gauge how effective the suggestions were, we constrained the variables the tool suggested in the order they were suggested to us, and calculated the resulting area. The area was determined utilizing the same area model discussed in previous sections, i.e. giving adders a linear area model while multipliers are assigned an area model proportional to the square of their input word size. We also determined an asymptotic lower bound to the area by implementing all suggestions simultaneously to determine how quickly our tool would converge upon the lower bound.

5.1. Wavelet Transform

The first benchmark we present is the wavelet transform. The wavelet transform is a form of image processing, primarily serving as a transformation prior to applying a compression scheme, such as SPIHT[8]. A typical discrete wavelet transform runs a high-pass filter and low-pass filter over the input image in one dimension.

The results are down sampled by a factor of two, effectively spatially compressing the wavelet by a factor of two. The filtering is done in each dimension, vertically and horizontally for images. Each pass results in a new image composed of a high-pass and low-pass sub-band, each half the size of the original input stream. These sub-bands can be used to reconstruct the original image.

This algorithm was hand-mapped to hardware as part of work done by Thomas Fry[8]. The hardware utilized was a WildStar FPGA board from Annapolis Microsystems consisting of three Xilinx Virtex 2000E FPGAs and 48 MBytes of memory. Significant time was spent converting the floating-point source algorithm into a fixed-point representation by utilizing methodologies similar to those we present in this paper. The result was an implementation running at 56MHz, capable of compressing 8-bit images at a rate of 800Mbits/sec. This represents a speedup of nearly 450 times as compared to a software implementation running on a Sun SPARCStation 5.

The wavelet transform was implemented in MATLAB and passed into Précis. In total, 27 variables were selected to be constrained. These variables were then marked for range-finding analysis and annotated MATLAB code was generated. This code was then run in the MATLAB interpreter with a sample image file (Lena) to obtain range values for the selected variables. These values were then loaded into Précis to obtain a lower bound to be used during the slack analysis phase. The results of the slack analysis are shown in Figure 9.

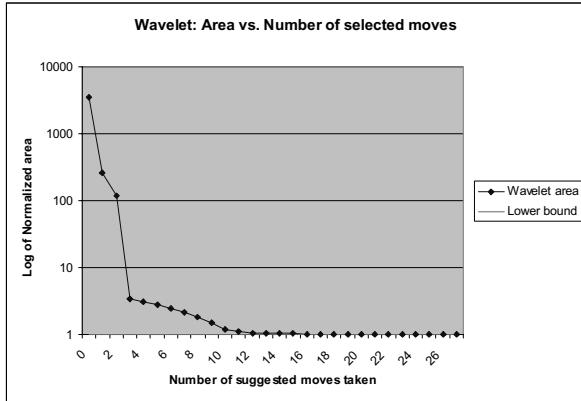


Figure 9. Wavelet area vs. number of suggestions implemented.

These results are normalized to the lower bound obtained by setting all variables to their lower bound constraints and computing the resulting area. This graph shows that between the upper bound and lower bound, there is a theoretical area difference of about three orders of magnitude. The slack analysis results suggested constraining the output image array, then the low and high pass filter coefficients, and then the results of the

additions in the multiply-accumulate structure of the filtering operation. By taking the suggested moves in order and recomputing the order at each step, we were able to reach with ten percent of the lower bound area of the system in eleven moves. Perhaps more importantly, the tool was able to suggest a pattern of moves that would allow us to reach within a factor of three from the lower bound in just four moves. Finally, by about thirteen moves, the normalized area was within less than three percent of the lower bound, and further improvements were negligible. At this point a typical user may choose to stop optimizing the system.

It is important to note that the area values obtained by Précis are simply calculated by reducing the range of a number of variables to their range-found lower bounds. This yields what could be considered the “best-case” solution when optimizing. In reality, though, one would add another step to the development cycle whereby upon choosing the variable for optimization as suggested by the tool, the developer would perform an intermediate simulation step to determine if, by lowering the precision requirements of that variable, any error would be introduced in the results. This step is made easier by the automatic generation of annotated simulation code for use in MATLAB. In many cases, there might be an intolerable amount of error introduced by utilizing the lower bound, in which case the user would choose an appropriate precision range, fix that value as a constraint upon that variable in Précis and continue utilizing the slack analysis phase to find the next variable for optimization.

5.2. Probabilistic Neural Network: PNN

Another benchmark we investigated was a multi-spectral image-processing algorithm designed for NASA satellite imagery that is similar to clustering analysis or image compression. More details can be found in [7]. Briefly, each multi-spectral image pixel vector is compared to a set of “training pixels” or “weights” that are known to be representative of a particular class. The probability that the pixel under test belongs to the class under consideration is given by the formula depicted in Equation 1.

$$f(\vec{X} | S_k) = \frac{1}{(2\pi)^{d/2} \sigma^d} \frac{1}{P_k} \sum_{i=1}^{P_k} \exp \left[-\frac{(\vec{X} - \vec{W}_{ki})^T (\vec{X} - \vec{W}_{ki})}{2\sigma^2} \right]$$

Equation 1. The core PNN formula.

Here, \vec{X} is the pixel vector under test, \vec{W}_{ki} is the weight i of class k , d is the number of spectral bands, k is the class under consideration, σ is a data-dependent “smoothing” parameter, and P_k is the number of weights in class k . This formula represents the probability that

pixel vector \bar{X} belongs to the class S_k . This comparison is then made for all classes and the class with the highest probability indicates the closest match.

This algorithm was manually implemented on a WildChild board and described in [7]. The WildChild board from Annapolis Microsystems consists of eight Xilinx 4010E FPGAs, a single Xilinx 4028EX FPGA, and 5MBytes of memory. Like the wavelet transform described earlier, significant time and effort was spent on variable range analysis, with particular attention being paid to the large multipliers and the exponentiation required by the algorithm. This implementation obtained speedups of 16 versus a software implementation on an HP workstation.

The algorithm was implemented in MATLAB and passed into Précis. From here, twelve variables were selected for range finding analysis, annotated MATLAB was generated, range-analysis was performed, and slack analysis was run utilizing the derived lower and upper bounds.

Again, all results were normalized to the lower bound area. As shown in Figure 10, the tool behaved similarly to the wavelet benchmark in that it was able to reach within five percent of the lower bound within six moves, where after additional moves serve to make only minor improvements in area. However, with the PNN algorithm, we are able to demonstrate even further refinement of the slack analysis approach.

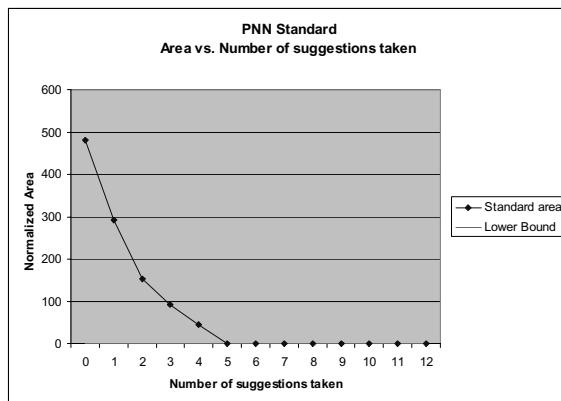


Figure 10. PNN area vs. number of suggestions implemented utilizing only range-analysis-discovered values.

For a seasoned developer that has a deeper insight into the algorithm, or for one that already has an idea of how the algorithm would map to hardware, the range-analysis phase sometimes returns results that are sub-optimal. For example, the range-analysis of the PNN algorithm upon a typical dataset resulted in several variables being constrained to ranges such as $[2^0, 2^{-25}]$, $[2^8, 2^{-135}]$, $[2^0, 2^{-208}]$, and so on. This simply means that the

range-finding phase discovered values that were extremely small and thus recorded the range as requiring many bits to the right of decimal point to capture all the precision information. The shortcoming of the automated range-analysis is that it has no means by which to determine at what precision values become too small to affect follow-on calculations, and therefore might be considered unimportant. With this in mind, the developer would typically restrict the variables to narrower ranges that preserve the correctness of the results while requiring fewer bits of precision.

Précis provides the functionality to allow the user to make these decisions in its annotated MATLAB code generation. In this case, the user would choose a narrower precision range and a method by which to constrain the variable to that range consistent with how they will be implementing the operation in hardware—truncation, saturation, rounding, or any of the other methods presented in previous sections. Then, the developer would generate annotated MATLAB code for simulation purposes, and re-run the algorithm in MATLAB with typical data sets. This would allow the user to determine how narrow of a precision range would be tolerable, and subsequently constrain the variables in Précis accordingly. The user can perform this determination either during slack analysis, or prior to beginning slack analysis.

There are two types of scenarios that may occur depend primarily on the experience level of the developer. With a developer that has not dealt with precision analysis and software to hardware mappings extensively, it may be that they wouldn't notice the unreasonable range information obtained by the range-finding analysis phase until the variable was suggested for optimization by the tool. For this case, the user would perform an appropriate simulation of the variable at that stage of the slack analysis and obtain tighter bounds. On the other hand, for a more experienced hardware designer that has encountered precision analysis before, they might look closely at the range-finding results prior to running the slack analysis. In this case, they would most likely run simulations and find more reasonable precision ranges for the variables in question, and constrain them before even beginning the slack analysis phase.

The results for these two scenarios are shown plotted together in Figure 11, normalized to the lowest bound among all three approaches. To differentiate the three methods, the first proposed method is shown as “simple”, and is the same method used to plot the results for the wavelet benchmark. The “user guided” method refers to fixing the variables during slack analysis. Finally, the “start constrained” method denotes fixing the variables in question prior to starting slack analysis.

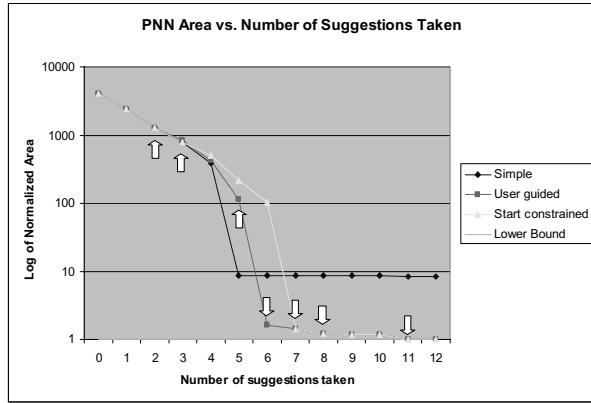


Figure 11. PNN area with user-defined variable precision ranges. Moves that had variables constrained to more reasonable ranges are highlighted with arrows.

At first glance, one can see that all three methods provide similar trends, approaching the lower bound within five to seven moves. This behavior is expected and is consistent with the results of the wavelet benchmark. However, one might expect that the start-constrained and user-guided methods would reach near the lower bound more quickly than the simple method. Instead, they take one or two additional moves to get near the lower bound compared to the simple method. This can be explained by understanding how the tool performs propagation across variables whose ranges are constrained by the user. By fixing the range of a variable, neither forward nor backward propagation will alter their precision ranges. In effect, we trust the user's decision when they fix a variable's precision range. The net effect is that any gains that might have been realized through back-propagation of smaller ranges will not be achieved if they must propagate through a variable whose range has been fixed. Finally, as the method used to compute the order of variables to constrain is greedy by nature, changing the order in which constraints are applied will alter the curve slightly.

6. Related work

While there have been several recent research efforts targeting precision analysis, none have approached it in such an interactive fashion. As mentioned in previous sections, Mark Stephenson and Jonathan Babb's work developing the Bitwise compiler at MIT [6] is an excellent foundation work regarding precision propagation techniques. They have applied their techniques in the SUIF compiler infrastructure and are targeting the C language for silicon compilation.

Anshuman Nayak's work at Northwestern University [9] is very relevant to our own research, as it is based upon the same MATCH compiler framework as our own. This work utilizes a similar propagation engine within the MATCH compiler as optimization phases and attempts to

perform all analysis, including error, automatically, generating RTL VHDL suitable for synthesis.

Two other research efforts, one at the University of Southern California and one at Imperial College in London, approach the precision matter in an entirely different way. Kiran Bondalapati's work on dynamic precision management of loop computations [10] concentrates on developing a formal methodology for analyzing the precision requirements of loop structures. Finally, George A. Constantinides, et. al. have developed a Synoptix-based system for the analysis and automated generation of DSP applications[11].

7. Conclusions

In this paper we have demonstrated the need for precision analysis tools in the development cycle of software to hardware mapping. To direct the developer's efforts in hand-optimizing the precision of algorithms mapped to hardware, we have developed and demonstrated a tool, Précis, which allows the user to automate many tasks necessary for effective precision analysis. We have demonstrated how our tool can aid the developer in simulation of fixed-point math with automatic annotated MATLAB code generation. We have also developed MATLAB scripts that support range analysis of a user's MATLAB code in order to deduce a theoretical lower bound to the precision of selected variables. We have also presented a framework for propagation of precision range information over a MATLAB program. Finally, we have described our methodology of slack analysis, and have shown how the suggestions provided by this methodology can be helpful in guiding the user in their manual precision optimization on real-world benchmarks.

8. Acknowledgements

This research was supported by contracts with NASA and DARPA, and a grant from NSF. Scott Hauck was supported in part by an NSF CAREER award and a Sloan Research Fellowship.

9. References

- [1] Synopsis CoCentric SystemC Compiler.
http://www.synopsys.com/products/cocentric_systemC/cocentric_systemC_ds.html
- [2] Celoxia Handel-C Compiler.
http://www.celoxica.com/products/technical_papers/datasheets/DATHNC002_0.pdf
- [3] P. Banerjee, N. Shenoy, A. Choudhary, S. Hauck, C. Bachmann, M. Chang, M. Haldar, P. Joisha, A. Jones, A. Kanhere, A. Nayak, S. Periyacheri, M. Walkden. "MATCH: A MATLAB Compiler for Configurable Computing Systems". Technical report CPDC-TR-

9908-013, submitted to IEEE Computer Magazine,
August 1999.

- [4] P. Banerjee, A. Choudhary, S. Hauck, N. Shenoy.
“The MATCH Project Homepage”.
<http://www.ece.nwu.edu/cpdc/Match/Match.html> (1 Sept. 1999).
- [5] AccelChip, info@accelchip.com,
<http://www.accelchip.com>.
- [6] Mark Stephenson. “Bitwise: Optimizing Bitwidths Using Data-Range Propagation”. Master's thesis. Massachusetts Institute of Technology. May 2000.
- [7] Mark L. Chang. “Adaptive Computing in NASA Multi-Spectral Image Processing”. Master's Thesis. Northwestern University, Evanston, IL. December 1999.
- [8] Thomas W. Fry. “Hyperspectral Image Compression on Reconfigurable Platforms”. Master's Thesis. University of Washington, Seattle, IL. May 2001.
- [9] A. Nayak, M. Haldar, A. Choudhary, P. Banerjee, “Precision And Error Analysis Of MATLAB Applications During Automated Hardware Synthesis for FPGAs”, Proc. Design Automation and Test in Europe (DATE 2001), Berlin, Germany. Mar. 2001.
- [10] Kiran Bondalapati and Viktor K. Prasanna, “Dynamic Precision Management for Loop Computations on Reconfigurable Architectures”, IEEE Symposium on Field-Programmable Custom Computing Machines, April 1999.
- [11] George A. Constantinides, Peter Y.K. Cheung, Wayne Luk, “The Multiple Wordlength Paradigm”, IEEE Symposium on Field-Programmable Custom Computing Machines, April 2001.

Adaptive Computing in NASA Multi-Spectral Image Processing

Mark L. Chang
Northwestern University
Evanston, IL 60208 USA
mchang@ece.nwu.edu

Scott A. Hauck
University of Washington
Seattle, WA 98195 USA
hauck@ee.washington.edu

Abstract

This paper presents our work accelerating a NASA image processing application by utilizing adaptive computing techniques. In our hand mapping of a typical algorithm, we were able to obtain well over an order of magnitude speedup over conventional software-only techniques with commercial off-the-shelf hardware. This work complements [6, 7] by widening the comparison between hardware and software approaches and updating the effort to include more current technologies. Additionally, we discuss the applicability of NASA applications to adaptive computing and present this work in the context of Northwestern's MATCH Project—a MATLAB to adaptive computing systems compiler.

1 Introduction

In 1991 NASA launched the *Earth Science Enterprise* initiative to study, from space, the Earth as a scientific system. The centerpiece of this enterprise is the Earth Observing System (EOS), which plans to launch its first satellite, *Terra*, before the turn of the century. With the launch of *Terra*, ground processing systems will have to process more data than ever before. In just six months of operation, *Terra* is expected to produce more data than NASA has collected since its inception [7].

As a complement to the *Terra* satellite, NASA is establishing the EOS Data and Information System (EOSDIS). Utilizing Distributed Active Archive Centers (DAACs), EOSDIS aims to provide a means to process, archive, and distribute science and engineering data with conventional high-performance parallel-processing systems.

Our work, combined with the efforts of others, strives to augment the ground-based processing centers with adaptive computing technologies such as workstations equipped with FPGA co-processors. These co-processors allow engineers to implement their algorithms in a hardware substrate with hardware-like speeds while maintaining the flexibility of software.

A trio of factors motivates the use of adaptive computing over custom ASICs: speed, adaptability, and cost. The current trend in satellites is to have more than one instrument downlinking data, which leads to instrument-dependent processing. These processing “streams” involve many different algorithms producing many different “data products”. Due to factors such as instrument mis-calibration, decay, damage, and incorrect pre-flight assumptions, these algorithms often change during the lifetime of the instrument. Thus, while a custom ASIC would most likely give better performance than an FPGA solution, it would be far more costly in terms of time and money due to an ASIC’s cost, relative inflexibility, and long development cycle.

This paper presents the results we have obtained accelerating a NASA multi-spectral image processing application with an FGPA co-processor. We discuss the application, our implementation and results, and conclude with the MATCH Project, our automated MATLAB-to-FPGA compiler.

2 Multi-Spectral Image Classification

In this work, we focus on accelerating a typical multi-spectral image classification algorithm. This algorithm uses multiple spectrums of instrument observation data to classify each satellite image pixel into one of many classes. This technique transforms the multi-spectral image into a form that is more useful for analysis by humans. It also represents a form of data compression or clustering analysis. In our

implementation these classes consist of terrain types such as urban, agricultural, rangeland, and barren. In other implementations these classes could be any significant distinguishing attributes present in the dataset. One proposed scheme to perform this automatic classification is the Probabilistic Neural Network classifier as described in [5]. In this scheme, each multi-spectral image pixel vector is compared to a set of “training pixels” or “weights” that are known to be representative of a particular class. The probability that the pixel under test belongs to the class under consideration is given by the following formula.

$$f(\vec{X} | S_k) = \frac{1}{(2\mathbf{p})^{d/2} \mathbf{s}^d} \frac{1}{P_k} \sum_{i=1}^{P_k} \exp \left[-\frac{(\vec{X} - \vec{W}_{ki})^T (\vec{X} - \vec{W}_{ki})}{2\mathbf{s}^2} \right]$$

Equation 1

Here, \vec{X} is the pixel vector under test, \vec{W}_{ki} is the weight i of class k , d is the number of bands, k is the class under consideration, \mathbf{s} is a data-dependent “smoothing” parameter, and P_k is the number of weights in class k . This formula represents the probability that pixel vector \vec{X} belongs to the class S_k . This comparison is then made for all classes and the class with the highest probability indicates the closest match.

3 Implementation

The PNN algorithm was implemented in multiple software languages as well as on a COTS adaptive computing engine. The software languages included Matlab, Java, and C, while the hardware version was written in VHDL.

3.1 Software approaches

3.1.1 MATLAB (iterative)

Because of the high-level nature of MATLAB, this is likely the simplest and slowest implementation of the PNN algorithm. The MATLAB source code snippet is shown in Figure 1. This code uses none of the optimized vector routines that MATLAB provides. Instead, it uses an iterative approach, which is known to be slow in interpreted MATLAB. The best use for this approach is to benchmark improvements made by other approaches.

MATLAB has many high-level language constructs and features that make it easier for scientists to prototype algorithms, especially when compared to more “traditional” languages such as C or Fortran. Features such as implicit vector and matrix operations, implicit variable type declarations, and automatic memory management make it a natural choice for scientists during the development stages of an algorithm. Unfortunately, MATLAB is also much slower than its C and Fortran counterparts, due to its overhead. In later sections we discuss how the MATCH compiler can compile MATLAB codes to reconfigurable systems (such as DSPs and FPGAs), thus maintaining the simplicity of MATLAB while achieving higher performance.

```

for p=1:rows*cols
    fprintf(1,'Pixel: %d\n',p);
    % load pixel to process
    pixel = data( (p-1)*bands+1:p*bands);
    class_total = zeros(classes,1);
    class_sum = zeros(classes,1);
    % class loop
    for c=1:classes
        class_total(c) = 0;
        class_sum(c) = 0;
        % weight loop
        for w=1:bands:pattern_size(c)*bands-bands
            weight = class(c,w:w+bands-1);
            class_sum(c) = exp( -(k2(c)*sum( (pixel-weight).^2 ))) + class_sum(c);
        end
        class_total(c) = class_sum(c) * k1(c);
    end
    results(p) = find( class_total == max( class_total ) )-1;
end

```

Figure 1: Matlab (iterative) code

3.1.2 MATLAB (vectorized)

To gain more performance, the iterative MATLAB code in Figure 1 was rewritten to take advantage of the vectorized math functions in MATLAB. The input data was reshaped to take advantage of vector addition in the core loop of the routine. The resultant code is shown in Figure 2.

```

% reshape data
weights = reshape(class',bands,pattern_size(1),classes);
for p=1:rows*cols
    % load pixel to process
    pixel = data( (p-1)*bands+1:p*bands);
    % reshape pixel
    pixels = reshape(pixel(:,ones(1,patterns)),bands,pattern_size(1),classes);
    % do calculation
    vec_res = k1(1).*sum(exp( -(k2(1).*sum((weights-pixels).^2)) ));
    vec_ans = find(vec_res==max(vec_res))-1;
    results(p) = vec_ans;
end

```

Figure 2: Matlab (vectorized) code

3.1.3 Java and C

A Java version of the algorithm was adapted from the source code used in [6]. Java's easy to use GUI development API (the Abstract Windowing Toolkit) made it the obvious choice to test the software approaches by making it simple to visualize the output. A C version was also developed based upon the Java version.

3.2 Hardware approaches

3.2.1 FPGA Co-Processor

The hardware used was the Annapolis Microsystems WildChild FPGA co-processor [2]. This board contains an array of Xilinx FPGAs—one “master” Xilinx 4028EX and eight Xilinx 4010E’s, referred to as Processing Elements (PEs)—interconnected via a 36-bit crossbar and a 36-bit systolic array. PE0 has 1MB of 32-bit SRAM while PEs 1-8 have 512K of 16-bit SRAM each. The layout is shown in Figure 3. The board is installed into a VME cage along with its host, a FORCE SPARC 5 VME card. For reference, the Xilinx 4028EX and 4010E have 1024 and 400 Configurable Logic Blocks (CLBs), respectively. This, according to [9] is roughly equivalent to 28,000 gates for the 4028EX and 950 gates for the 4010E. This system will be referred to as the “MATCH Testbed”.

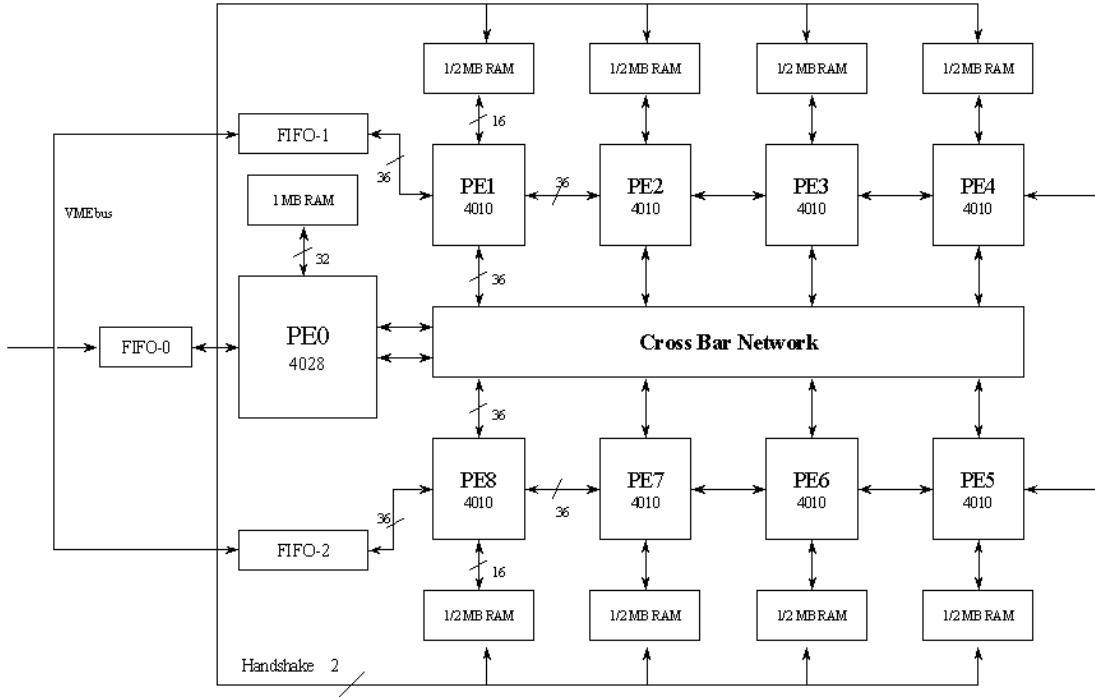


Figure 3: Annapolis Microsystems WildChild(tm) FPGA Co-processor

3.2.2 Initial mapping

The initial hand mapping of the algorithm is shown in Figure 4.

$$(\text{Note: } K_2 = \frac{1}{2s^2} \text{ and } K_1 = \frac{1}{(2p)^{d/2}s^d})$$

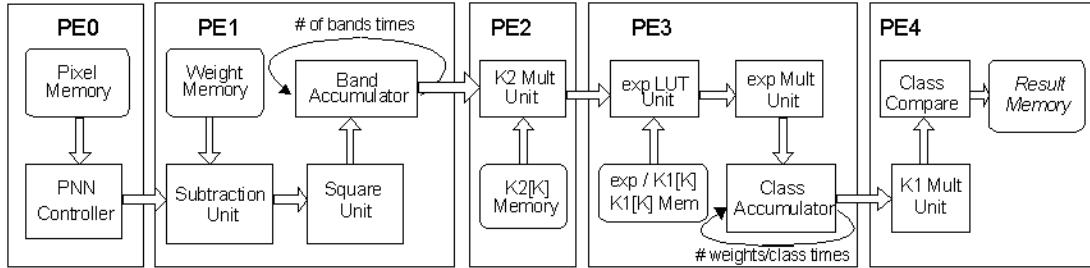


Figure 4: Initial FPGA mapping

PE	% Used
PE0	5%
PE1	67%
PE2	85%
PE3	82%
PE4	82%

Table 1: FPGA Utilization -- Initial mapping

The mappings were done completely by hand utilizing behavioral VHDL. The designs were simulated with Mentor Graphics' VHDL simulator using VHDL models supplied by Annapolis Microsystems. The

code was synthesized using Synplicity's Synplify synthesis tool. Placement and routing was accomplished with Xilinx Alliance tools. As shown in Figure 4, the computation for Equation 1 is spread across five FPGAs.

The architecture of the WildChild board suggests that PE0 be the master controller for any system mapped to the board. For example, PE0 is in control of the FIFOs, several global handshaking signals, and the crossbar. Thus, in our design, PE0 is utilized as the controller and "head" of the computation pipeline. PE0 is responsible for synchronizing the other processing elements, including beginning and ending the computation.

The pixels to be processed are loaded into the PE0 memory. When the system comes out of its Reset state, PE0 waits until it receives handshaking signals from all the slave processing elements. It then acknowledges the signal and sends a pixel to PE1. PE0 then waits until PE1 signals completion before it sends another pixel. This process repeats until all available pixels have been exhausted.

PE1 implements $(\vec{X} - \vec{W}_{ki})^T (\vec{X} - \vec{W}_{ki})$. As the input pixel data in our data set is 10 bits wide, we perform an 11-bit signed subtraction with each weight in each training class, and obtain a 20-bit result from the squaring operation. In the equation, \vec{X} and \vec{W}_{ki} represent vectors since the input data is multi-band. In our hardware, we accumulate across these bands, giving us a 22-bit result, which is then passed to PE2 along with the current class under test. When we have compared the input pixel with all weights from all classes, we signal PE0 to deliver a new pixel.

PE2 implements the K_2 multiplication, where $K_2 = \frac{1}{2s^2}$, and s is a class-derived smoothing

parameter given with the input data set. The K_2 values are determined by the host computer and loaded into PE2 memory. The processing element uses the class number passed to it by PE1 and looks up the 16-bit K_2 value in memory. The 16-bit K_2 values are stored in typical fixed-point integer format, in this case shifted left by 18 bits. The result of the multiplication is a 38-bit value where bits 0...17 represent the fractional part of the result, and bits 18...37 represent the decimal part. To save work in follow-on calculations we examine the multiplication result. The follow-on computation is to compute $e^{(\dots)}$. For values larger than about 32, $e^{-32} = 12.6 * 10^{-15}$. Given the precision of later computations, we can and should consider this result to be zero. Conversely, if the multiplication result is zero, then the exponentiation should result in a 1. We test the multiplication result for these conditions and set flags appropriately before sending the multiplication result and the flags to the next PE. Only bits 1...22 of the multiplication result are sent to the next PE. Bit 0 is of such low order that it makes no significant impact on follow-on calculations.

PE3 implements the exponentiation in the equation. As our memories in the slave PEs are only 16-bits wide, we cannot obtain enough precision to cover the wide range of exponentiation required by the algorithm by using only one lookup table. Since we can break up e^{-a} into $e^{-b} * e^{-c}$, by using two table lookups and one multiplication, we can achieve a higher precision result. Given that the low-order bits are mostly fractional, we devote less memory space to them than the higher order, decimal, portions. Thus, out of the 22 bits, the lower 5 bits look up e^{-b} while the upper 17 bits look up e^{-c} in memory. If the input flags do not indicate that the result should be zero or one, these two 16-bit values are multiplied together and accumulated. When all weights for a given class have been accumulated, we send the highest 32 bits of the result (the most significant) to PE4. Additionally, in the next clock cycle we send the appropriate K_1 constant read from local memory.

PE4 performs the K_1 multiplication and class comparison. The accumulated result from PE3 is multiplied by the K_1 constant. This is the final value, $f(\vec{X} | S_k)$, for a given class, S_k . As discussed, this represents the probability that pixel vector \vec{X} belongs to class S_k . PE4 compares each class result and keeps the class index of the highest value of $f(\vec{X} | S_k)$ for all classes. When we have compared all classes for the pixel under test, we write the maximum class index to memory. The final mapping utilization for each processing element is given in Table 1.

3.2.3 Host program

As with any co-processor, there must be a host processor that issues work to the co-processor. In this case it is the FORCE VME Board. The host program is written in C and is responsible for packaging the data for writing to the FPGA co-processor, configuring the processing elements, and collecting the results when the FPGA co-processor signals its completion. All communication between the host computer and the FPGA co-processor are via the VME bus.

3.2.4 Optimized mapping

To obtain higher performance we developed several optimizations to the initial mapping. The resultant optimized hand mapping is shown in Figure 5. Specifically, due to the accumulators in PE1 and PE3, there are different inter-arrival rates for each of the processing units. The processing elements following the accumulators thus have lower inter-arrival rates. In our data set there are four bands and five classes. Thus, PE2 and PE3 have inter-arrival rates of one-fourth that of PE1, while PE4 has an inter-arrival rate of one-twentieth that of PE1. Exploiting this we can use PE2 and PE4 to process more data in what would normally be their “stall”, or “idle” cycles. This is accomplished by replicating PE1, modifying PE2 through PE4 to accept input every clock cycle, and modifying PE0 to send more pixels through the crossbar in a staggered fashion.

PE	% Used
PE0	5%
PE1-4	75%
PE5	85%
PE6	61%
PE7	54%
PE8	97%

Table 2: FPGA Utilization -- Optimized mapping

As shown in Figure 5, we now utilize all nine FPGAs on the WildChild board. FPGA utilization figures are given in Table 2. PE0 retains its role as master controller and pixel memory handler (Pixel Reader). Now, the Pixel Reader unit sends four pixels through the crossbar, one to each of four separate Subtract/Square units (PE1-PE4). The Pixel Reader then rotates the crossbar configuration to direct the output of each Subtract/Square unit to the K2 Multiplier unit in sequence.

The Subtract/Square units in PE1-PE4 remain virtually unchanged from the initial mapping, except for minor handshaking changes with the Pixel Reader unit. Each of the Subtract/Square units now must fetch one of the four different pixels that appear on the crossbar. This is accomplished by assigning each Subtract/Square unit a unique ID tag that corresponds to which pixel to fetch (0 through 3). The other change is to the output stage. Two cycles before output, each Subtract/Square unit signals the Pixel Reader unit to switch crossbar configurations to allow its output to be fed to the K2 Multiplier unit through the crossbar.

The K2 Multiplier unit, now in PE5, is pipelined in order to accept input every clock cycle. The previous design was not pipelined in this fashion. Now, instead of being idle for three out of four cycles, the K2 Multiplier unit operates every clock cycle.

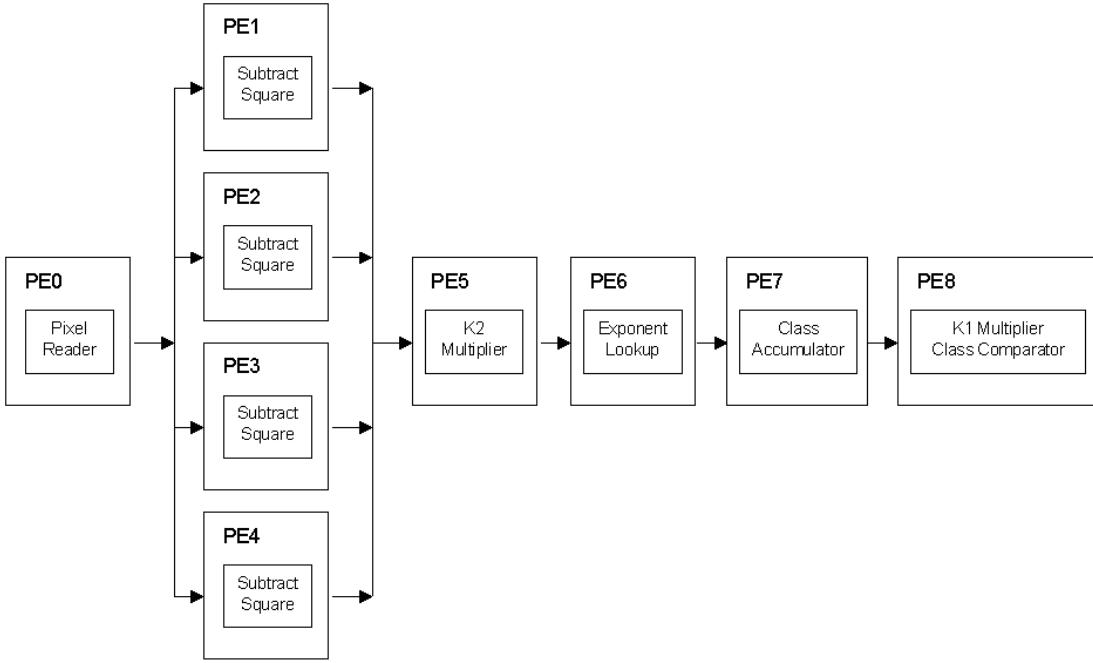


Figure 5: Optimized mapping

The Exponent Lookup unit changes significantly in this optimized mapping. Due to the need for a pipelined unit, we had to reduce the number of memory lookups within the PE. Having two lookups (one for e^{-b} and one for e^{-c}) would require stalling the pipeline and reducing overall performance. By wiring additional data and address lines through the crossbar to PE7's (the Class Accumulator unit) memory, we can perform two memory reads per clock cycle.

The class accumulator has been moved out of the exponentiation unit and into its own processing element. This is because the pipeline will be operating on four pixels at a time, thus requiring four separate accumulators. These accumulators are implemented in PE7, the Class Accumulator unit.

Finally, in PE8, we implement the K_1 multiplier and Class Comparison unit. Instead of receiving K_1 constants from the previous processing element, the constants are read from the local PE8 memory. This is possible because although all the PEs have the same number of memory locations, PE0 needs four times as many locations because each pixel consists of four different bands of data. In the result memory, we need only one location per pixel. PE8 accumulates class index maximums for each of the four pixels after the input data has been multiplied by the appropriate K_1 class constants. Like in the initial design, after all classes for all four pixels have been compared, the results are written to memory. For this design to fit within one Xilinx 4010E FPGA, we were forced to utilize a smaller 8x8 multiplier in multiple cycles to compute the final 38-bit multiplication result. This is possible because PE8 has a much slower inter-arrival rate than PE7, as it is after an accumulator.

4 Results

4.1 Test platform

The test platform for the software versions of the algorithm (MATLAB/Java/C) was an HP Visualize C-180 Workstation running HP-UX 10.20 with 128MB of RAM. The reference platform for the hardware version was the entire MATCH Testbed as a unit. This includes a 100MHz FORCE SPARC 5 VME card running

Solaris 2.5.1 (SunOS 5.5.1) with 64MB of RAM and the WildChild FPGA co-processor. Additional timings were performed on software versions running on the MATCH Testbed. Unfortunately, Matlab was not available for our MATCH Testbed at the time of this writing.

The Matlab programs were written as Matlab scripts (m-files) and executed using version 5.0.0.4064. The Java version was compiled to Java bytecode using HP-UX Java build B.01.13.04 and executed with the same version utilizing the Just-In-Time compiler. The C version was compiled to native HP-UX code using the GNU gcc compiler version 2.8.1.

4.2 Performance

Table 3 shows the results of our efforts in terms of number of pixels processed per second and lines of code required for the algorithm. The lines of code number includes input/output routines and excludes graphical display routines. Software version times are given for both the HP workstation and the MATCH Testbed where appropriate.

Platform	Method	Pixels/sec	Lines of code
HP	Matlab (iterative)	1.6	39
HP	Matlab (vectorized)	36.4	27
HP	Java	149.4	474
HP	C	364.1	371
MATCH	Java	14.8	474
MATCH	C	92.1	371
MATCH	Hardware+VHDL (initial)	1942.8	2205
MATCH	Hardware+VHDL (optimized)	5825.4	2480

Table 3: Performance results

When compared to more current workstations (1997) such as the HP, the optimized adaptive hardware implementation achieves a speedup of 3640 over the slowest implementation — the Matlab Iterative benchmark — 40 over the HP Java version, and 16 over the HP C version. An alternate comparison can be made to quantify the acceleration that the FPGA co-processor can provide over just the FORCE 5V Sparc system. This might be a better comparison for those seeking more performance from an existing reference platform by simply adding a co-processor such as the WildChild board. In this case, we compare similar technologies, dating from 1995. In this case, the hardware implementation is 390 times faster over the MATCH Testbed Java version, and 63 times faster than the MATCH Testbed executing the C version.

Of course, the cost in this approach is the volume of code and effort required. The number of lines metric is included in an attempt to gauge the relative difficulty of coding each method. This is a strong motivating factor for our work in the MATCH Project.

5 MATCH

The MATCH Project [3, 4] at Northwestern University encompasses much more than just the hand mapping of algorithms to an adaptive computing system. MATCH is a project whose goal is to develop a MATLAB compilation environment for distributed heterogeneous adaptive computing systems. The main objective of the MATCH project is to simplify tasks such as the one described here. Specifically, MATCH strives to make it easier for users to develop code that targets an adaptive computing system, such as our MATCH Testbed. The core effort is in the design of a MATLAB compiler that can generate code targeting COTS FPGA coprocessors, embedded microprocessors, and DSP coprocessors. We describe the MATCH compiler here briefly. For more detailed information, please refer to [3, 4].

As our work here shows, currently mappings to adaptive computing systems are time consuming, require personnel proficient in developing for the target hardware, and a low-level understanding of the algorithm architecture. To allow more users to take advantage of the performance potential of hardware acceleration through adaptive computing techniques, the MATCH compiler will compile MATLAB codes to a configurable computing system automatically. The most obvious benefit of this MATLAB compiler will

be in its translation of concise, easy to write, widely accepted MATLAB codes into fast hardware implementations with little or no user intervention.

More than just FPGAs, a complete configurable computing system would have other configurable components, such as DSPs and embedded processors, to share the workload. In most cases, purely FPGA-based systems are not ideal targets for complete algorithm implementation. Code that is infrequently used wastes valuable space on an FPGA, while operations such as floating point and complex arithmetic are better suited for a general purpose processor. In these cases, the MATCH compiler and an associated MATCH Testbed will be able to automatically optimize performance by partitioning work among a variety of processing elements. Here, the goal of the MATCH compiler is to optimize performance under resource constraints, and to obtain performance within a factor of 2-4 of the best manual approach.

The basic compiler approach begins with parsing MATLAB code into an intermediate representation, an Abstract Syntax Tree (AST). From this, we build a data and control dependence graph from which we can identify scopes for varying granularities of parallelism. This is accomplished by repeatedly partitioning the AST into one or more sub trees. Nodes in the resultant AST that map to predefined library functions map directly to their respective targets, and any remaining procedural code is considered user-defined procedures. A controlling thread is automatically generated for the system controller — in our case, the FORCE 5V processor. This controlling thread makes calls to the functions that have been mapped to any of the processing elements in the order defined by the data and control dependency graphs. Any remaining user defined procedures are automatically generated from the AST. The target for this generated code is determined by the scheduling system [8].

Currently, our compiler is capable of compiling MATLAB code to any of the three targets (DSP, FPGA, and embedded processor) as directed by user-level directives in the MATLAB code. The compiler currently exclusively uses library calls to cover the AST. We have developed library implementations of such common functions as matrix multiplication, FFT, and IIR/FIR filters for FPGA, DSP, and embedded targets. Work is ongoing in MATLAB type inferencing and scalarization to allow automatic generation of C+MPI codes and Register Transfer Level (RTL) VHDL which will be used to map user-defined procedures (non-library functions) to any of the resources in our MATCH Testbed. Work is also ongoing in integrating our scheduling system, Symphony [8], into the compiler framework to allow automatic scheduling and resource allocation without the need for user directives.

6 Conclusion

We have accelerated a typical NASA image-processing application by using an adaptive computing engine and a hand mapping of the algorithm. We have developed techniques and approaches for accelerating an existing hand mapping of an algorithm and have shown a more efficient implementation of our algorithm. Finally we have introduced the MATCH project which will enable scientists with little or no knowledge of VHDL and reconfigurable computing to obtain higher performance from their MATLAB code.

There are many motivating factors for developing a working relationship with NASA with respect to the MATCH project. The principle factor being that NASA applications are well suited to the MATCH project. Satellite ground-based applications, especially with the launch of Terra, will need to be accelerated in order to accomplish the scientific tasks that they were designed for in a timely and cost-effective manner. As we have shown, a typical image-processing application can be accelerated by several orders of magnitude over conventional software-only approaches by using adaptive computing techniques. But, to accomplish this requires someone that is knowledgeable in the use of FPGA co-processors and is comfortable in a hardware description language such as VHDL or Verilog. Very few scientists can be bothered to learn such languages and concepts. This too goes well with MATCH, as MATCH strives to enable users of a high-level language such as MATLAB to increase the performance of their MATLAB codes without intimate knowledge of the target hardware.

NASA has also demonstrated an interest in adaptive technologies, as can be witnessed by the existence of the Adaptive Scientific Data Processing (ASDP) group at NASA's Goddard Space Flight Center in Greenbelt, MD. The ASDP is a research and development project founded to investigate adaptive computing with respect to satellite telemetry data processing [1]. They have done work in accelerating

critical ground-station data processing tasks using reconfigurable hardware devices. Their work will only be complemented with the advancement of the MATCH compiler.

Finally, our work fits into the MATCH framework as a benchmark for the MATCH compiler. If the goal of the MATCH compiler is to get performance within a factor of 2-4 of the best hand-tuned mapping of an algorithm [4], then our work with multi-spectral image processing is not only a good benchmark for compiler performance, but it will also help identify functions and procedures necessary for real-world applications. Work is ongoing in the testing of the compiler and these codes.

7 Acknowledgments

The authors would like to thank Marco A. Figueiredo and Terry Graessle from the ASDP at NASA Goddard for their invaluable cooperation. We would also like to thank Prof. Clay Gloster of North Carolina State University for his initial development work on the PNN algorithm. This research was funded in part by DARPA contracts DABT63-97-C-0035 and F30602-98-0144, and NSF grants CDA-9703228 and MIP-9616572.

8 Bibliography

- [1] Adaptive Scientific Data Processing. “The ASDP Home Page”. <http://fpga.gsfc.nasa.gov> (22 Jul. 1999)
- [2] Annapolis Microsystems. *Wildfire Reference Manual*. Maryland: Annapolis Microsystems, 1998.
- [3] P. Banerjee, A. Choudhary, S. Hauck, N. Shenoy. “The MATCH Project Homepage”. <http://www.ece.nwu.edu/cpdc/Match/Match.html> (1 Sept. 1999)
- [4] P. Banerjee, N. Shenoy, A. Choudhary, S. Hauck, C. Bachmann, M. Chang, M. Haldar, P. Joisha, A. Jones, A. Kanhare, A. Nayak, S. Periyacheri, M. Walkden. *MATCH: A MATLAB Compiler for Configurable Computing Systems*. Technical report CPDC-TR-9908-013, submitted to IEEE Computer Magazine, August 1999.
- [5] S. R. Chettri, R. F. Cromp, M. Birhmingham. *Design of neural networks for classification of remotely sensed imagery*. Telematics and Informatics, Vol. 9, No. 3, pp. 145-156, 1992.
- [6] M. Figueiredo, C. Gloster. *Implementation of a Probabilistic Neural Network for Multi-Spectral Image Classification on a FPGA Custom Computing Machine*. 5th Brazilian Symposium on Neural Networks—Belo Horizonte, Brazil, December 9-11, 1998.
- [7] T. Graessle, M. Figueiredo. *Application of Adaptive Computing in Satellite Telemetry Processing*. 34th International Telemetering Conference—San Diego, California, October 26-29, 1998.
- [8] U.N. Shenoy, A. Choudhary, P. Banerjee. *Sympathy: A Tool for Automatic Synthesis of Parallel Heterogeneous Adaptive Systems*. Technical report CPDC-TR-9903-002, 1999.
- [9] Xilinx, Inc.. *The Programmable Logic Data Book-1998*. San Jose, California: Xilinx, Inc.

REU Site: Engineering Education Research: Understanding and Improving Student Experiences

Project Elements:

- *New REU Site*
- **REU Site: Engineering Education Research: Understanding and Improving Student Experiences**
- Principal Investigator: *Debbie Chachra*
- Submitting Organization: *Franklin W. Olin College of Engineering*
- Other Organizations: *N/A*
- Location: *Franklin W. Olin College of Engineering, Needham, Massachusetts*
- Main field and sub-field of the research: *Engineering education*
- Number of undergraduate participants per year: *10*
- *Summer* REU Site,
- *10* weeks per year
- Project includes an *RET component*
- Student applicant contact: *Sharon Breitbart, Assistant Director, Initiative for Innovation in Engineering Education, Franklin W. Olin College of Engineering, 1 Olin Way, Needham, MA 02492 Phone: 781 292 2526 Email: reu@olin.edu*
- URL: *TBD*

Project Summary:

Olin College is engaged in an ambitious research program to understand and improve the educational experience of engineering undergraduates. The undergraduate student researchers who have collaborated with Olin faculty over the years have found the experience transformative, giving them new perspectives on the field of engineering and on their own educational experiences. A number have gone on to graduate research and careers in engineering education. This project proposes to extend this opportunity beyond the Olin student body, to include 10 undergraduates and 2 teachers per year for each of the next 3 years. In recruiting participants, we will place special emphasis on those early in their undergraduate careers, members of historically under-represented groups, and those reconsidering a decision to major in engineering, for whom insights gained from an engineering education REU may be especially significant.

Intellectual Merit: To prepare graduates for a rapidly changing world defined by global challenges and opportunities, engineering education needs to undergo a transformation. Olin College advances engineering educational reform by providing an academic environment that supports the ongoing creation, implementation, and assessment of innovative curricula and pedagogies. Olin faculty share a deep commitment to educational reform: seven of the project personnel are currently engaged in NSF-funded educational research projects, and one-third of the total Olin faculty are participants in this proposal. The projects proposed here focus on three broad themes: the creation and assessment of *curricular and pedagogical innovation*; improvements to student development of *professional and interpersonal skills* (including teamwork and communication), and new insights into the relationships between educational outcomes and *student orientations towards learning*, including intrinsic motivation, self-direction, and identity.

Broader Impact: The primary goals of this REU/RET in engineering education are to attract, support and retain students from a range of backgrounds; to broaden students' and teachers' contextual awareness; and to foster participants' ability to engage in lifelong learning. This REU site will provide a research opportunity for students who are particularly at risk with respect to engineering identity, including those from historically underrepresented groups and early undergraduates, as well as for teachers who work with these same students in pre-college settings. Past experience suggests that this program will be transformative of participant relationship to engineering and research. Research findings will be disseminated broadly, with REU/RET participants taking significant roles in this process.

Project Description

Overview:

What is the experience of undergraduate engineering students? How is the student experience altered by pedagogy, instructor attitude or action? How does the experience vary based on student background or personal attributes? How do these factors interact to create different outcomes, retention rates, or professional identities for different students? These questions are at the heart of the research program that informs this REU Site proposal. In different ways, using different techniques, investigating different populations and diverse institutions and different aspects of the engineering undergraduate experience, the senior personnel on this proposal aim to understand and to improve the experience of undergraduate engineering students.

Innovation and research in engineering education are part of the mission of Olin College, and the senior personnel involved in this project are already advancing the field through work funded by the National Science Foundation and other organizations. With this REU site and RET supplement, we propose to extend the opportunity to explore these questions to ten undergraduate students and two teachers per year. Based on our prior experiences, we believe that this opportunity will be transformative for participants, enriching their experience as researchers and altering their perception of their own education and that of the students who surround them.

The research objectives of this proposal are:

- To understand factors in engineering education that affect learning outcomes, retention, and development of professional identity.
- Through this understanding, to improve learning outcomes, retention, and development of professional identity among engineering undergraduates.

The specific objectives of hosting an REU site also include the following:

- To provide participants with an intellectual toolkit for research, including qualitative and quantitative analysis tools; skills for creative and generative thinking; and improved communication skills.
- By engaging undergraduates in formal research projects on these topics, to affect student researchers' understanding of their own engineering education.

Full details of the (supplemental) RET program, including focus, structure and recruitment, are appended.

The REU program is structured as 10-week residential programs, to be held on the Olin College campus in suburban Boston, during the summers of 2011, 2012, and 2013. The program will be jointly overseen by PI Professor Debbie Chachra (academic/intellectual components) and co-PI Professor Lynn Andrea Stein (cohort/structural components). Olin's Initiative for Innovation in Engineering Education (I2E2) will also provide administrative coordination of the REU Site. Planned activities for participants include: active research in collaboration with and closely mentored by faculty; interactive sessions on various elements of educational research practice, facilitated by faculty from Olin and other academic institutions; and social events to help foster a sense of community among the participants.

Students will be recruited nationally, with an emphasis on underrepresented groups. We plan to enroll students from a range of backgrounds, including both engineering and education. We will make a special point of reaching out to students in their first years of an undergraduate program, students from community colleges, and students who may be questioning their relationship with engineering. Undergraduate students who have previously engaged in engineering education research with faculty at Olin College have reported that this work has changed their perception of and relationship to their own

engineering education. We anticipate that this REU experience would both appeal to and help foster retention of women and of groups that are historically underrepresented in engineering; this type of experience is not widely available on college campuses.

Olin College was founded specifically to develop, provide, research and disseminate innovations in engineering education. We are undergraduate-focused and student-centered. Faculty, staff, and students enjoy close and collegial relationships, with project teams – from the initial design of the college onwards – typically including membership across these groups. Beyond developing courses and curricula and teaching undergraduates, a significant fraction of Olin College faculty lead funded research programs in engineering education (twelve faculty, or more than a third of the total number, are listed as senior personnel on this proposal). Mentoring, advising, and performing research with undergraduate students is fundamental to the College. As well, the Initiative for Innovation in Engineering Education (I2E2) was founded to foster and disseminate research on engineering education and innovative pedagogical practices. With our emphasis on undergraduate mentoring and engineering education, Olin College is uniquely situated to host the proposed REU program.

Nature of Student Activities: Research, Professional Development, and Community Development

A transformative experience:

Since the founding of Olin College and the arrival of the first class in 2002, research and innovation in engineering education has integrally involved undergraduate students. For many of the students, the experience was transformative:

Janet Tsai, Class of 2006: Over the course of her time at Olin, Janet Tsai was involved in educational research with Professor Yevgeniya V. Zastavker (*Diversifying Engineering Through Gateway Courses: Assessment of Project-Based Learning in Undergraduate Physics, Mathematics and Engineering*) as well as enrolling in a co-curricular offering focusing on gender and engineering education. She wrote, “It’s amazing how much my perspective and worldview changed due to the papers we read and the discussions we had, and I can’t imagine it any other way.” On graduation, Janet joined iRobot Corporation as a mechanical engineer, where she worked for several years. Janet is currently enrolled in the PhD program in engineering education at the University of Colorado at Boulder, with a self-described goal of “chang[ing] how engineering is perceived and practiced to create a more inclusive community that finds strength in diversity instead of unease.” In addition to publishing her work with Prof. Zastavker [27], Janet contributed an essay, “An Engineering Approach to Feminism,” to the anthology *Click: When We Knew We Were Feminists* (Seal Press, 2010).

Jared Frey, Class of 2008: Jared began an independent study project with Professor Jonathan Stolk, initially on a subject unrelated to education. Over the course of their weekly discussions, he became intrigued by educational research, particularly intrinsic motivation and self-directed learning. Throughout the semester, Jared read widely in the field and developed research ideas, and he continued this work into the following summer. The research collaboration culminated in the submission of a research grant proposal to the Spencer Foundation in August 2008. Jared is currently completing a PhD in psychophysical studies of human vision, and is considering a return to engineering education for his postdoctoral work.

Casey Canfield, Class of 2010: Casey also worked with Professor Zastavker on a project in engineering education. Her research on “Mathematics and Physics Faculty Conceptions of Teaching in a First-Year Integrated Project-Based Engineering Curriculum,” presented at the annual conference of the American Society of Engineering Education in 2009, not only won the

prize for Best Student Presentation but also placed 3rd in the Educational Research and Methods Division. [25] Casey is now a marine engineering technician at the University of Washington.

Yoonkyung (Alison) Shin, Class of 2013: First-year engineering student Alison Shin approached Professor Debbie Chachra about a potential summer research project investigating first-year students in design courses. Professor Chachra gave Alison several research papers on the subject of gender and engineering education in order to provide Alison with information to help her decide if she was interested. Alison remarked on how much reading about the experiences of other women in engineering allowed her to understand and contextualize her own experiences during her first year of engineering school. She decided to work in education research during the summer of 2010, and plans to continue her involvement.

These students – and many peers with similar stories – have reflected on the ways that research into engineering education changed their perceptions of engineering, of learning, and of themselves. In some cases, conducting research into factors affecting the experiences of others provided important new perspectives on students' struggles with engineering or identity. While many Olin students have had similar experiences through research and para-curricular activities offered on our campus, these opportunities are less common at many other academic institutions. We are confident that an REU program at Olin College has the potential to lead to similarly transformative experiences for all its participants.

In order to provide the richest possible experience to our REU and RET participants, we have designed a program consisting of three closely-integrated components: research, professional development, and community/cohort development. Details of each of these three components follow.

Structure of research projects:

Most summer projects will begin with data that has already been collected by research mentors and their teams. REU and RET participants will learn about the dataset and background questions, develop rubrics for analyzing data, perform data analysis, and write up their findings. Towards the end of the summer, REU and RET participants will generally have the opportunity to design follow-on studies. While some individual experiences may deviate from this pattern, we expect most student experiences to follow this structure as it will give students the greatest opportunity to experience the majority of the research project cycle in the ten-week summer interval. The nature of most of the affiliated projects is that data-collection is time-intensive and often must take place during the academic year. By analyzing already-collected data, REU/RET participants will be able to probe the heart of the research questions addressed. Participants will also have the opportunity to write up their findings and, in many cases, to co-author research publications. By designing follow-on studies, participants will also “close the loop” and participate in experimental design. (Further information on RET-specific follow-up is included in the RET appendix.)

REU students who are interested in continuing their projects over the subsequent year will be encouraged to do so and supported to the extent possible. In some of these cases, Olin faculty members will be able to arrange collaborations with faculty members at students' home institutions; in others, students may collaborate with Olin faculty members directly at a distance. See the ‘Student Recruitment’ section, below, for further information on Olin’s relationship with some anticipated student home institutions. A further possibility is that students, on completing the REU program, be will be able to work on new projects at their home institution; a letter of commitment for an example of one such situation, from Dr. Michael Prince at Bucknell University, is appended.

Examples of research projects:

Olin College was founded in responses to calls for a new direction in engineering education, such as the National Academy of Engineering’s *The Engineer of 2020*. Like many other academic institutions, Olin aspires to educate well-rounded, professionally skilled, insightful and motivated engineers. Hands-on

projects, design and communication, active learning pedagogies, and work on diverse teams are all hallmarks of Olin's education. As these approaches become increasingly prevalent within engineering education, Olin faculty members have also undertaken rigorous research projects to understand the ways in which various personal and environmental factors interact to produce differing experiences and outcomes. Each of these aspects plays a role in this REU Site proposal: REU/RET participants will have the opportunity to study pedagogical and curricular innovation; professional and interpersonal skills development; and the engineering student experience. Representative projects are described below; currently funded projects are noted.

Pedagogical/curricular innovation: In response to calls by academic, government, and industrial leaders for engineering education reform, much work has been done in the area of pedagogical and curricular innovation. New technologies, new pedagogies, and the pressures of ever-expanding content compel us to develop new approaches to what happens in our classrooms. In addition, increasing recognition of diverse student needs demands that we alter our approaches to meet students where they are and not simply where we imagine they might be. A number of projects at Olin address innovation in curriculum or in pedagogy.

Dr. Amon Millner's work in engineering and computation at the K-12 level reaches out to inner city youth who do not imagine themselves as programmers or as people who use computational design to express themselves creatively. Dr. Millner's research explores the intersection of human computer interaction, tangible user interface design, community organizing, and the learning sciences fields. He has designed, deployed, and evaluated the Hook-ups System – a computational construction kit coupled with engaging activities that enabled young people to create their own novel tangible user interfaces – specifically for use in community centers. Together with REU/RET participants, Dr. Millner will continue his explorations of communication, teamwork and learning communities in this environment, as well as the development of new tools to support these activities. (*funded: NSF/ITR*)

Dr. Yevgeniya V. Zastavker has been engaged in a comparative study of innovative and traditional curricula. With a focus on project-based learning as an example of an innovative curriculum/pedagogical structure, Dr. Zastavker investigates how teaching methods and curricula may relate to gendered patterns of performance, interests, and participation in undergraduate science, technology, engineering, and mathematics introductory courses that are part of engineering programs. Several undergraduates have already participated in this project and co-authored publications; future REU/RET participants will have similar opportunities. (*funded: NSF/GSE*)

Other potential projects for REU/RET participants include: development of courses that integrate engineering and history content, skills and attitudes and a study of personal, institutional, and cultural barriers to successful implementation of the courses (R. Martello, J. Stolk); development and validation of an instrument that correlates performance in science courses with math skills and preparation (C. Morse), and the development of interdisciplinary engineering courses, including assessing the effects that novel pedagogies might have on learning outcomes (M. Somerville, J. Geddes, others).

Professional and interpersonal skills development: At one time, engineering education was understood primarily as the acquisition of technical knowledge: engineering science as the basis of professional competence. While technical prowess remains crucial to engineering success, so-called 'soft skills,' such as teamwork and communication, are increasingly seen as crucial aspects of the professional development of engineers. Olin faculty members investigate the ways in which these skills are acquired and understood as well as the interaction between these skills and other traits that students bring into the classroom.

Dr. Lynn Andrea Stein is currently investigating the development of verbal and visual communication skills, teamwork, and hands-on project experiences with a compressed core curriculum in computing. A recent REU supplement is supporting student participation in analysis of current data and should lead to publication within the next year. The broader project involves a dissemination grant and study of curricular innovation and change. Research questions include the importance of various factors in

facilitating adoption of curricular innovation, the extent to which professional and interpersonal skills are developed, and the impact of curricular change on student outcomes. Many additional REU/RET collaborations are possible as this project progresses (*funded: NSF/CPATH*).

Other potential REU/RET projects in this area include: analysis and self-reflection of student behavior in team settings using videorecordings (N. Tatar, Y.V. Zastavker, D. Chachra); investigation of the role of discourse in the transition of undergraduate engineering students from pre-professional to professional practice by analyzing the patterns of discourse during Olin College's senior design project, a long-term engineering project that involves external technical liaisons (M. Chang); and a study of the impact of similar 'soft skills' curricular augmentation on multiple campuses (L.A. Stein, M. Somerville, others).

The engineering student experience: A desire for increased diversity in the engineering talent pool, coupled with a recognition that students are not one-size-fits-all, requires a better understanding of students and their experiences. Recruitment, retention, and individual satisfaction can be modulated by environmental factors interacting with personal traits. The psychology of learning plays an important role in this research; understanding the student experience can help us to better support student development as engineers as well as the acquisition of lifelong learning skills. The specific research targets of Olin faculty members include self-efficacy, motivation, self-direction, agency, and identity development.

Dr. Debbie Chachra recently received an NSF CAREER award to investigate the relationship between engineering self-efficacy, activities, and learning outcomes in first-year engineering design courses at three institutions: Olin College, MIT, and the University of Washington. The underlying hypothesis is that students entering college with lower engineering self-efficacy (which may include those in traditionally underrepresented groups) are less likely to volunteer for engineering tasks. The resultant relative deficit in mastery experiences therefore leads to lower gains in engineering self-efficacy. As part of this study, interventions to improve engineering self-efficacy outcomes and retention will be developed, tested, and implemented. (*funded: NSF/CAREER*)

Another research project in this space examines how instructor choices affect a range of student outcomes related to their development as self-directed and lifelong learners. Dr. Jonathan Stolk is conducting a study to investigate students' motivational orientations, application of cognitive and behavioral skills, and perceptions of autonomy support in four different engineering courses at four colleges. This research builds on self-determination theory, drawing on existing research that suggests strong correlations between student autonomy support and outcomes related to SDL. (*funded: NSF/IECI*)

Other studies in this space include research into the role that projects and interdisciplinary approaches may play in student motivation and self-direction (J. Stolk, R. Martello); and the analysis of collaborative curriculum design in changing faculty conceptions of learning and student experiences (M. Somerville, J. Geddes). Complementing and supporting these research areas is Dr. Jon Adler's work on agency, the extent to which an individual describes her/his life as self-directed and empowered. As well as facilitating quality educational outcomes, fostering agency in engineering students is likely to produce positive mental health outcomes.

Other research areas: Engineering education is an active field of research at Olin College. Faculty regularly attend conferences such as the American Society for Engineering Education and Frontiers in Education, have collaborations across the US and in other countries, and initiate and seek funding for new projects. The descriptions provided above, therefore, are simply examples of ongoing and proposed research in these fields: it is likely that additional projects will be added over the course of an REU program. In addition, Olin hosts a continual stream of medium- and long-term visitors who bring interesting projects and potential collaborations to the campus.

Professional development of REU participants:

The REU and RET experiences are not simply opportunities to engage in research; they are also significant cohort activities designed to develop the broad professional skills of participants. At Olin, these aspects of the experience will come through the mentoring relationship as well as peer collaboration; through specific skills workshops; and through activities such as journal club, weekly research meetings, writing groups, and a culminating research conference.

Collaborative, mentored research environment: An important component to the professional development of participants is direct mentoring by senior personnel; Olin College does not have a graduate program. All REU/RET supervision will be provided directly by research mentors who are heavily involved in the day-to-day conduct of these projects. Direct mentoring and apprentice-style learning, as in graduate education, provides a singularly effective way to learn the professional practice of research.

All REU participants will be provided with workspaces in a shared studio space – the education foundry – and most REU mentors will also work in this space during the summer. Part of this studio space will be shared soft seating and brainstorming space, facilitating informal as well as structured interaction. Olin's culture is highly communal and interactive, with unusually strong student-faculty interactions. The shared studio space will also encourage peer collaboration and cohort development. To facilitate the concurrent community and professional development of participants, informal coffee breaks will be held frequently.

Research skills workshops: *[faculty-led]* Three day-long research skills workshops, with sessions led by experienced faculty from Olin College and from other institutions, will be held over the course of the summer in order for students to develop the skills and methodologies needed to perform research in engineering education. The first session will focus on human subjects research (as most of the participating students will be working on projects involving human subjects¹ and will need to be certified as trained in this type of research), research ethics, and basic research skills such as keeping a notebook. Later sessions will include topics such as quantitative methods, qualitative methods, interviewing, generative thinking, experiment design, and how to write a research paper. Past workshops in these areas have been led by Olin faculty members or by guests such as Chris Rogers, Director of the Center for Engineering Education and Outreach at Tufts University, and Boston University's Professor Robert Weiss. In keeping with Olin's pedagogical philosophy, these workshops will stress engagement and interaction of the research participants.

Research meetings: *[facilitated by PI Chachra]* As is typical for research labs, a weekly research meeting will be held. These will be lunchtime meetings, at which the REU/RET participants will discuss their progress in the preceding week, as well as their plans for the upcoming week. Attendance at the meetings will be mandatory for REU/RET participants and expected of research mentors. Regular research meetings allow students to be aware of the different projects and their progress, create a sense of shared purpose, and also give students a forum in which they can discuss and share successes and challenges.

Journal club: *[student-led]* Journal clubs are common in graduate schools and research groups; the typical format is that participants read and discuss papers, taking turns to choose the paper and lead the discussion. Olin's REU/RET program will include a weekly journal club held over coffee and tea in the shared studio. Participants will take turn leading the discussion. As well as increasing student exposure to

¹ Although individual students will require human subjects research training, IRB approvals are not necessary for this REU site/RET supplement as all projects will have received prior approval. No new IRB protocols are anticipated in this grant.

the relevant educational research, these discussions will allow students to develop their skills in critical analysis of the literature and their facility with academic discourse.

Seminar series and career planning sessions: *[facilitated by co-PI Stein]* Periodic formal and informal sessions will be scheduled during daily coffee and tea times. These will include seminars from Olin's frequent visitors and senior personnel, practice talks by REU/RET participants, and informal discussions of topics of interest. In addition, several career planning sessions will be held with senior personnel and other Olin college faculty and staff.

Work buddies and writing groups: At the beginning of the summer, participants will be paired with peers to function as "work buddies", to provide mutual support and 'reality checks' of day-to-day work. This program, based on the writings of Virginia Valian [1], has proved effective for research, writing, and other open-ended tasks, providing a structured means for participants to learn self-regulation. As students transition from data analysis to writing, existing work-buddy pairs or new small groups will be established to provide mutual support and assistance through the writing process. Work buddy pairings and writing groups will be supervised by co-PI Stein.

Research conference: At the end of the summer session, all participants will be required to present their projects and research findings at a day-long conference. In addition to the Olin community, faculty and researchers from other institutions will be invited to attend.

Community and cohort development:

The professional development activities described above are structured to concurrently foster community development. Olin College is a residential college, so REU students will live together in the Olin residence halls. Lunches are frequently communal, including a weekly barbecue hosted by one of the research groups on campus, and the relatively small on-campus community is highly collegial. A social program will also be part of the REU experience, including a welcome reception, trips to Boston, and a farewell dinner. At least once during the session, the participants will be invited to a dinner hosted by a member of the senior personnel at his or her home.

The Initiative for Innovation in Engineering Education (I2E2) also hosts an annual summer faculty institute on Designing for Student Engagement as well as other professional development events for faculty members from other institutions. REU/RET participants will be invited to engage with the faculty in attendance, both formally and at social events.

In addition to this program, there will be undergraduates working on mathematics research (participants in NSF EMSW21-MCTP: Long-term Undergraduate Research Experience; Prof Sarah Adams, PI.), students involved in engineering projects funded by NASA, and undergraduate researchers in a variety of other research laboratories and disciplines. These students also typically live in the Olin dormitories and will form a part of the residential summer community for REU participants. In addition, several programs for middle- or high-school students – technology enrichment and/or leadership development –typically take place on the Olin campus during the summer, and interactions (including mentoring by REU/RET participants) will be arranged where appropriate.

The Research Environment:

Facilities at Olin College:

Workspaces: Olin College has committed to provide REU participants with individual workspaces in a collaborative studio-style workspace. These spaces, normally used for design courses including User-Oriented Collaborative Design, are designed to facilitate collaborations and interactions among small groups, and to encourage the use of the environment to foster learning and experimentation (for example, students are encouraged to hang documents on the walls, pin them to dividers, etc.). The design studio

also includes shared soft seating space, full-wall whiteboards, and other communal resources to encourage conversation and collaboration.

In addition to the shared spaces, students also will have access to a range of spaces at Olin including classrooms, seating areas, conference rooms, and workrooms. This will allow participants to find workspaces that best support their specific needs.

As many students will be working on research governed by Human Subjects Research guidelines, secure storage areas for computers and documents will be provided as necessary.

Living spaces: Students will be provided with dormitory rooms in Olin College's modern residence halls. Normally, summer students at the college do not have a meal plan; rather, they take advantage of the shared kitchen spaces in the residence halls to prepare their own meals. However, a meal plan may be available for part of the summer sessions. (Financial arrangements have been made for either of these contingencies; see the Budget Justification for details.)

Computing facilities: All participants will be provided with a computer (either a laptop or a desktop system) for use during the program. Technical support will be provided by Olin's Information Technology department, which includes a walk-in help desk as well as online and telephone support. As well, students will have access to Olin College's computing infrastructure (e.g. servers to back up and share research data).

Other facilities: The College will provide students with other supplies and support required for their research projects, including photocopying, office supplies, and the like, as well as access to the college library, including online access and interlibrary loan.

Diversity at Olin College: Olin College has an institutional commitment to gender equity (approximately 44% of the student body is female), and a culture that is highly supportive of LGBT students, as evidenced by the College's ranking of #4 in the Princeton Review "LGBT-Friendly" list. Olin College was also ranked #3 for "Lots of Race/Class Interaction."

Local academic community: Olin College is located in the Greater Boston area, where there is a concentration of institutions for higher education, allowing us to extend the research community beyond the campus. Wellesley College has already committed to support the participants in the REU (see attached letter) and we intend to engage other local institutions as well.

Faculty mentors at Olin College:

A college-wide commitment: It is illustrative of the important role that engineering education research plays at Olin College that the twelve senior personnel listed on this proposal represent more than one-third of the *total* faculty at the college. In fact, the NSF REU Site limitation to twelve senior personnel prevented us from including several additional colleagues whose work is closely related. They and other visitors may participate in advising and community activities along with the present proposal team.

Training and mentoring of faculty: As faculty at an undergraduate engineering college, all mentors in this program are both committed to and have considerable experience with working closely with undergraduate students. Project-based learning is emphasized in the curriculum, which means that faculty frequently play a mentoring role for students engaged in self-directed, intrinsically-motivated activities. Senior personnel on this project also regularly attend training and professional development sessions at conferences and professional meetings. Within the College, ongoing professional development programs include What's Working Wednesdays, a weekly meeting in which faculty discuss pedagogy and other topics, multidisciplinary week-long faculty-led workshops in which faculty are exposed to new areas (past topics have ranged from digital photography to lab-based cell biology), as well as extensive informal colleague-to-colleague mentoring. Other professional development opportunities include I2E2 programs, such as the summer faculty institute on Designing for Student Engagement.

Diversity of the mentor pool: Of the twelve senior personnel, three are female, three are persons of color (including one African-American), and two who do not identify as heterosexual.

Monitoring of faculty: Informal monitoring of faculty will occur through faculty participation in the REU program: for example, participation or facilitation of interactive sessions, attendance at research meetings, and involvement in social events. Participating students will also be asked to complete a short survey regarding their experience at the midpoint of the program in order to provide timely and actionable feedback. Co-PI Stein will be responsible for the formal monitoring of the student experience and mentor relationships.

Biographies of faculty mentors:

Olin College faculty collaborate and publish extensively with undergraduate students, some of which are noted here and can be found in the References section (collaborators who were undergraduate students at the time of the research are marked with an asterisk).

(PI) Debbie Chachra, Associate Professor of Materials Science: Dr. Chachra's current research interests in engineering education include an investigation of self-efficacy in first-year, project-based design courses. She has also studied and published on other aspects of the student experience, including persistence and migration, as well as differences in the engineering experience between male and female students. Dr. Chachra has also led and co-facilitated workshops on a number of topics, including implementing curricular innovation, minorities in engineering, and incorporation of educational research findings into the classroom. Dr. Chachra recently received an NSF CAREER Award in support of her research on engineering education. She also has an ongoing research program with undergraduate research students in the area of biological materials [2, 3]. As an undergraduate engineering student, Dr. Chachra herself participated in summer research programs, and she also co-founded a science and engineering summer camp within the Faculty of Applied Science and Engineering at the University of Toronto for students from the 5th-8th grade.

(co-PI) Lynn Andrea Stein, Professor of Computer and Cognitive Science and Director of the Initiative for Innovation in Engineering Education: As a member of Olin's founding faculty, Dr. Stein helped to design and develop the curriculum, hands-on learning pedagogies, and early recruitment of Olin's gender-balanced student body. Dr. Stein's research, at Olin and over a prior decade on the MIT faculty (1990-2000), spans the fields of artificial intelligence, human-computer interaction, cognitive science, and engineering and computer science education: she is a co-author of the foundational documents of the semantic web and the "mother" of a humanoid robot and an intelligent room. Dr. Stein has received the National Science Foundation Young Investigator Award, and several teaching and educational awards; she has also served on the Executive Council of AAAI, on the Member Services Board of the ACM, and in various leadership and advocacy positions as a woman in computing. Dr. Stein has long been a pioneer in computing and engineering education, introducing the Robot-Building Lab at the National Conference on Artificial Intelligence (which led to adoption of small robots in many CS curricula), creating the Rethinking CS101 (intro CS) and Small Footprint (major) curricula, and leading the development of the NAE Grand Challenge Scholars Program. Dr. Stein's leadership in professional development for faculty members, along with Olin's broad outreach mission, led to the creation in 2009 of the Initiative for Innovation in Engineering Education, of which Dr. Stein is the founding director.

Jon Adler, Assistant Professor of Psychology: Dr. Adler is a psychologist whose program of research focuses on identity development and mental health. One of the characteristics at the heart of this program of research is the theme of agency – the extent to which an individual describes his or her life as self-directed and empowered – which he has shown to be associated with positive mental health using a range of approaches. Dr. Adler will bring a psychological focus to our REU, highlighting the developmental aspects of fostering the next generation of engineers.

Mark L. Chang, Assistant Professor of Electrical and Computer Engineering. As one of the first cohort of faculty at Olin, Dr. Chang is deeply committed to engineering education. He has directly developed numerous courses, including Computer Architecture, Embedded Systems, Mixed Analog-Digital VLSI, and Mobile Application Development. In these courses, Dr. Chang has made an effort to integrate design and entrepreneurship, external expert speakers, and active-learning approaches. Dr. Chang is also involved in leading several teams of students each year in Olin's year-long senior capstone program, which provides a culminating engineering design experience through industry-funded projects. Mark has published and presented numerous papers related to computing and undergraduate engineering education, many with undergraduate students at Olin College.[4, 5, 6, 7, 8]

John Geddes, Professor of Mathematics and Associate Dean for Faculty Affairs and Research: Dr. Geddes' work in engineering education has included development of project-based approaches for first year curricula, pedagogical approaches that support student self-direction, and alternative approaches to feedback and assessment. [9, 10]

Robert Martello, Associate Professor of the History of Technology. Dr. Martello has led Olin College's Arts, Humanities, and Social Science curricular development since 2001. Dr. Martello has co-published several papers, facilitated curriculum-design workshops, and delivered numerous presentations investigating the relationship between interdisciplinary integration, self-directed learning techniques, and student motivation. Dr. Martello's history of technology research centers on early American industrialization, most recently culminating in the forthcoming book *Midnight Ride, Industrial Dawn: Paul Revere and the Rise of American Enterprise* (October 2010).

Amon Millner, Computing Innovation Fellow: Dr. Millner conducts research at the intersection of human computer interaction, tangible user interface design, community organizing, and learning sciences. He is currently developing a framework (the Constellation of Connected Creators) that provides strategies for people to facilitate workshops for others to learn through creating projects, then facilitating workshops for others. These projects typically take advantage of technological tools that help people learn approaches to design and come to understand engineering ideas in a hands-on manner. Dr. Millner's research involves developing and refining technological toolkits that are approachable to youth from diverse backgrounds; for example, the Hook-ups System - a platform built upon the Scratch programming language and the Scratch Sensor Board. This system was introduced to youth according to the Constellation of Connected Creators Framework.

Christopher Morse, Lecturer in Chemistry: Dr. Morse's pedagogical research to date has focused in two different areas: investigating the factors that affect success in college science courses, and developing a textbook on the chemistry of art, to aid in exposing non-science students to science concepts. He has studied the effects of student math skills and experiences and how they correlate to success in later college science courses, including designing a math diagnostic tool for this purpose. The textbook on chemistry and art, scheduled for release in 2012, covers topics in art, art history and art preservation, and has been a vehicle for working with undergraduate students to design experiments and as research assistants.

Mark Somerville, Associate Professor of Electrical Engineering and Physics and Associate Dean of Academic Programs and Curricular Innovation: Dr. Somerville's work in engineering education has included development of project-based approaches for first year curricula, pedagogical approaches that support and student self-direction, and gender-related task division in project-based learning. He is particularly interested in change processes for engineering education, the use of participatory design techniques and user-centered approaches as a means of facilitating curricular change, and the application of design thinking to instruction in non-design topics (e.g., mathematics and science). [11]

Jonathan Stolk, Associate Professor of Mechanical Engineering and Materials Science: Dr. Stolk has been a leader in the development of student-directed, project-based undergraduate engineering, science,

and design courses at Olin College. He has published several papers on student self-direction and project-based learning, and he has contributed to the design and implementation of integrated, project-based courses in the Department of Materials Engineering at Cal Poly, San Luis Obispo. Dr. Stolk is currently engaged in a collaborative study of the role of faculty in supporting students' growth as life-long learners, and an investigation of effects of disciplinary integration and autonomy on students' motivation and competency development. He also consults with a range of academic institutions on the design of project-based engineering curricula. Dr. Stolk has published extensively with undergraduates, in both education [12, 13, 14, 15, 16] and materials science [17, 18, 19, 20].

Nick Tatar, Assistant Dean of Student Life and Instructor: Dr. Tatar's research in engineering education has focused on the transition between high school and college, diversity within engineering, and the development of teaming skills. He recently developed a seminar series for first-year engineering students to enable them to engage with different aspects of the college academic environment.

Yevgeniya V. Zastavker, Associate Professor of Physics: Dr. Zastavker's current research interests include (i) investigation of biological and synthetic self-assembling membranes; and (ii) science/engineering education with specific emphasis on the issues of gender at the intersection with innovative pedagogical and curricular practices (e.g., project-based learning). She has served on the Committee on the Status of Women in Physics of the American Physical Society and represented the U.S. at the three International Conferences on Women in Physics (co-leading the U.S. delegation to the last event.) Dr. Zastavker currently serves on the Board of Trustees of the Spirit of Knowledge Charter School, a middle and high school that targets diverse population of the Worcester, MA, area and concentrates on STEM education. She is also a founding and current member of the Executive Committee of the Global Technology and Engineering Consortium, which works with international students ranging from middle school through college. Dr. Zastavker has mentored, collaborated, and published extensively with undergraduate students. [21, 22, 23, 24, 25, 26, 27].

Current funding in undergraduate education or engineering education research:

A number of the senior personnel on this proposal presently hold funding in the area of engineering education; a partial list follows.

CAREER: Exploring the Relationship Between Self-Efficacy and Project-Based Learning Among Engineering Students. **PI: Debbie Chachra.** January 1, 2010-December 31, 2014. \$400,084.

CPATH-1: Spreading Small Footprints. **PI: Lynn Andrea Stein.** August 15, 2009-July 31, 2012. \$157,755.

Workshop: Developing a National Network of Grand Challenge Scholars Programs. **PI: Lynn Andrea Stein.** September 1, 2009-February 29, 2012. \$50,000

Collaborative Research: Role of Faculty in Support Lifelong Learning: An Investigation of Self-Directed Environments in Engineering Undergraduate Classrooms. **PI: Jonathan Stolk.** September 1, 2008-August 1, 2011. \$83,667.

Motivation, Self-Direction, and Competency Development: A New Toolkit for 21st Century Undergraduate Engineers. **PI: Jonathan Stolk. Co-PI: Robert Martello.** June 15, 2008-May 31, 2011. \$149,439

GSE/RES: Does Project-Based Learning Matter to Undergraduate Women in Engineering? A Study of Performance, Interests and Participation in Gateway Technical Courses. **PI: Yevgeniya V. Zastavker.** September 1, 2006-August 2011. \$354,287.

Amon Millner is a Computing Innovation Fellow, supported by a subaward on NSF #CNS-1019343. The Second CI Fellows Project, sponsored by the Computing Research Association.

John Geddes is a member of the Senior Personnel in a multi-site program offering long-term research experiences for undergraduates (DMS #0636528; EMSW21-MCTP: Long-Term Undergraduate Research Experience).

In addition to these funded proposal, senior personnel have several pending proposals and continue to seek continue to seek external funding for research projects. We expect that these will provide additional opportunities for REU participants.

Student Recruitment and Selection:

This REU site will specifically target early undergraduates (first- or second-year). We intend to recruit students who are majoring in or considering a major in engineering, as well as those from outside of engineering with an interest in STEM, including students in education and students attending community colleges or liberal arts institutions that may not offer engineering programs. Because one goal of this project is to affect participating students' attitudes towards engineering, and because the research at the heart of this REU site proposal does not require advanced technical training in engineering as a prerequisite to productive contribution, we are ideally situated to reach out to students who may not have significant access to or fit with other REU sites. This REU site grant is also specifically intended to draw students from outside of Olin College: since many Olin College students have the opportunity to participate in engineering research during the academic year, we intend for the REU to primarily consist of students from other institutions (7 or more of the 10 participants each year). In addition, we intend to support two teachers in each year through an RET Supplement.

Our recruitment strategies involve targeting the many academic institutions with which Olin and/or senior personnel already have existing relationships and collaborations. At these schools, we have the greatest chance of reaching the early-career and questioning students who might not otherwise engage in REU sites. We will work closely with undergraduate and first-year program administrators at these schools to help identify engineering and pre-engineering students – as well as those who may be questioning their relationship to engineering – for whom this REU opportunity would be a particularly good fit.

We have strong relationships with the undergraduate program leadership, who can help to identify and recruit students for this program, in the following institutions:

- The University of Illinois at Urbana-Champaign, through the Olin-Illinois Partnership and the iFoundry program, scaling to 300 students next year.
- Wellesley College (a women's college) and Babson College, where we have close and expanding collaborations involving numerous student activities and programs; neither has its own engineering program, but students can receive an Olin certificate.
- Smith College (a women's college), which launched its new engineering program at the same time that Olin was beginning, and with which we frequently collaborate.
- MIT, Harvard University, Rensselaer Polytechnic Institute, Georgia Tech, Cal Poly SLO, Bucknell University, Duke University, USC, Louisiana Tech, St. Louis University, Wichita State University, WPI, Union College, Tufts University, and Brandeis University, where we have engineering education research collaborations, joint studies, or programmatic collaborations. A representative letter of commitment from MIT is appended.
- The University of Massachusetts at Amherst and Creighton College, where we have run joint faculty/student workshops that will draw students to this REU.

In addition, we will make use of extensive existing networks to recruit and publicize as broadly as possible, with the understanding that this less-targeted marketing will have wider reach but perhaps shallower draw. These include:

- Union College and, through it, participants in the annual Symposium on Engineering and Liberal Education.
- Duke University, USC, and the several dozen engineering schools participating in the Grand Challenge Scholars Program (which Olin College co-leads at the national level).
- The ASEE national and regional networks, in which most of the senior personnel are active participants.
- ACM's Special Interest Group on Computer Science Education (SIGCSE) as well as the regional Consortia for Computing in Small Colleges (CCSCs).
- The Society for Women Engineers and its local and regional chapters, in which Olin has been heavily involved.
- WEPAN, the Anita Borg Institute, NSBE, SHPE, and other organizations that reach historically underrepresented engineering students.

We will also rely on relationships with existing K-12 school districts in Massachusetts – notably Needham and Wellesley, which are our local districts, and Framingham, Worcester and Roxbury, minority-heavy districts where we have historically collaborated – to draw RET participants as well as to recruit graduates who may be eligible REU participants.

To complement our existing networks, we intend to extend our recruiting efforts to include engineering students at historically black colleges and universities (such as Howard University), at colleges for Native American students (such as Salish Kootenai College), and at colleges with a high proportion of Hispanic and Latino students (such as the University of Texas at El Paso or New Mexico State University).

Historically, senior personnel on this grant have worked extensively with female students, students from underrepresented minority groups, and students with disabilities, and we expect to continue these patterns in this REU Site, while making a concerted effort to reach students from ethnicities which are historically underrepresented in engineering. As well, this REU has been designed to appeal to students whose interaction with and opportunity for engineering research has been limited, including early-career students, and students with STEM interests who may be questioning their relationship with engineering.

Selection of students will be based on academic history (including transcript), an application letter detailing their interest in the program and any related experience, and a letter of recommendation. We intend to weight the narrative material relatively heavily compared to the transcript: while grades in courses typically reflect time management skills, discipline and effort as much as raw ability, many of our applicants will be in the early semesters of an engineering program, and the courses in their curricula may not map closely to the types of research they will be involved in during this REU program.

Project Management, Evaluation and Reporting:

Personnel and Project Management:

The senior personnel on this project have a long history of collaborative work and have managed numerous projects – including the creation of Olin College, the design and revision of its curriculum, and the establishment of Olin's Initiative for Innovation in Engineering Education – jointly. Much of the impetus for this REU site at Olin comes from our existing collaborative efforts in engineering education research and our desire to share this environment with a broader set of students. We have chosen a distributed management model – in which three individuals take the lead in complementary aspects of this project – to reflect our existing practices and overall management style.

The project director for this REU Site will be the PI, Debbie Chachra, Associate Professor of Materials Science and Engineering, holder of an NSF CAREER Award in Engineering Education, and former

(visiting) Research Scientist at the University of Washington's Center for the Advancement of Engineering Education. Prof. Chachra will oversee the research agenda of this REU Site. In coordination with the mentor group, Prof. Chachra will determine which of the ongoing affiliated projects will receive (how many) REU and/or RET participants in each year – depending on the stage of the project and its suitability for an educational and productive summer participant experience – and will oversee the recruitment and selection process for Olin's REU Site and RET applicants each year. The mandatory weekly research and coordination meetings will be led by Prof. Chachra. Prof. Chachra will also have primary responsibility for the evaluation of the REU Site, including development/modification of the specific instruments used to track changes in participant attitude.

Implementation and ongoing monitoring of the mentoring program will be led by co-PI Lynn Andrea Stein, Professor of Computer and Cognitive Science and Director of the Initiative for Innovation in Engineering Education. This includes both mentor and participant training. Prior to each summer, Prof. Stein will work with that summer's mentors to ensure coordinated cross-coverage and appropriate scaffolding. During the summer, Prof. Stein will oversee participant development sessions on quantitative and qualitative methods and on generative thinking, as well as the Engineering Education summer seminar series, work buddy pairings, and writing groups. Prof. Stein will also facilitate career planning sessions for the REU participants.

All other aspects of the REU Site, including the logistics of application, housing, weekly and special sessions, the end-of-summer research conference, administration of pre-, post- and longitudinal data collection, and all financial aspects, will be the responsibility of Sharon Breitbart, REU Site Coordinator. As Assistant Director of the Initiative for Innovation in Engineering Education, Ms. Breitbart has administrative and fiscal responsibilities for programs ranging from short courses to faculty fellowships. She will serve as a single point of contact for REU and RET participants for all administrative aspects of their experience.

Project Evaluation and Reporting:

As required by NSF reporting requirements, we will collect the following data about REU participants:

- i) academic information (home institution, degree of schooling, year, etc.)
- ii) demographic information (gender, ethnicity, race, disability status, citizenship)
- iii) site data, such as number of applicants, level of stipends, other expenses
- iv) program data, including titles and mentors for projects, and descriptions of adjunct activities

We also plan to:

Track our recruitment efforts (recipients, timing, method, number of applicants), in order to improve our processes over the course of the project.

Meet with project mentors (senior personnel) at the end of each summer, to discuss their experiences with REU participants, their satisfaction with the program, and their feedback on how to improve the program.

Follow-up with students and senior personnel about any dissemination of findings (through conference presentations, journal papers, or other means).

We plan to collect additional information regarding the development of student attitudes, behaviors and identity of students, as well as their academic and career trajectories. Unless otherwise specified, the following data will be collected at the start of the study, at the end of the session, and annually thereafter for the duration of the study.

- i) Academic information: current major, intended major, post-baccalaureate plans or activities

- ii) Confidence: confidence in math and science skills, confidence in professional and interpersonal skills, and confidence in solving open-ended problems

For students who are in STEM fields, we will also ask the following:

- iii) Persistence in STEM: academic (intention to complete degree) and professional (intention to practice engineering after graduation)
- iv) Motivation to study STEM: parental influence, high school mentor, college mentor, to do social good, psychological (intrinsic)

The items (survey questions) for the persistence, confidence and motivation constructs have been validated and previously administered as part of a longitudinal study of engineering students. [28]

In addition to the quantitative data described above, REU participants will be asked to provide qualitative information in the form of a written narrative, in which they will be asked to reflect upon their experiences in the summer program. Prompts for the reflection may include: *What is the most valuable thing you learned this summer? What, if anything, will you do differently in your education? Is there anything you wished you learned but didn't? If you could participate in this program again, what would you differently?* The reflections will be used to inform program design for the following year, as well as to provide context and nuance for the quantitative data described above. Follow-up surveys will also include the opportunity for participants to provide additional information about the impact of the program.

The PI and co-PI plan to apply for a renewal or no-cost extension to allow continued data collection and longitudinal follow-up with project participants. In any case, follow-up data collection from Olin student participants will be fully covered by existing institutional assessment mechanisms, and every effort will be made to include non-Olin student participants.

Evaluation plans for the RET component of this project are described in the RET appendix, making use of the aspects of REU evaluation that are comparable and augmenting them with teacher-specific and classroom assessment.

Results from Prior NSF Support

Olin College has not previously held an REU Site proposal. Some of the senior personnel have had REU supplements to NSF grants and others have included undergraduates in their research teams through internal grants or funding from other agencies. In particular, Professor Stein has held REU supplements to three education-related NSF grants (Young Investigator Award 9357761: *Embodiment Informs Cognition*; EIA 9979859/CNS 0196404: *CISE Educational Innovation: Radically Rethinking CSI*; CISE 0939128: *CPATH 1: Spreading Small Footprints*) involving a total of a dozen undergraduates and resulting in half a dozen student papers so far. [29, 30, 31, 32, 33, 34] (Professor Stein has supervised another three dozen undergraduate projects in other areas as well, with many students going on to graduate school and/or research careers.) Professor John Geddes has participated in a multi-site program offering long-term research experiences for undergraduates (DMS #0636528; *EMSW21-MCTP: Long-Term Undergraduate Research Experience*) that has involved 100 students and 20 mentors at five colleges including an HBCU and MI. Professor Geddes has worked directly with a half dozen students on this project, leading to two publications to date. [9, 10]

Most of the senior personnel on this proposal have worked extensively with undergraduates in research projects. The results of these collaborations are documented elsewhere in this proposal including in the references section, where collaborators who were undergraduates at the time of performing research and/or publication of findings are called out with an asterisk.

References

- 1 Valian, V. (1985). Solving a work problem. In M.F. Fox (Ed.), *Scholarly writing and publishing: Issues, problems, and solutions* (pp. 99-110). Boulder, CO: Westview Press.
- 2 *Angmo D, *McMahon M, Morse C, Chachra D (2008) Materials characterization of nest cell linings of bees from the family *Colletidae* (*Colletes inaequalis*). Presented at the Materials Research Society Spring Meeting, March 24-28, 2008, San Francisco, CA.
- 3 Chachra D, *McCusker EM (2005) Investigating Cell-Substrate Mechanical Interaction: A Tissue Engineering Approach. Presented at the 3rd European Medical and Biological Engineering Conference, November 20th-25th, Prague, Czech Republic.
- 4 Ilari Shafer*, Mark L. Chang , "Movement Detection for Power-Efficient Smartphone WLAN Localization", 13th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, October 2010.
- 5 Andrew Barry*, Noah Tye*, Mark L. Chang, "Interactionless Calendar-Based Training for 802.11 Localization," The 7th IEEE International Conference on Mobile Ad-hoc and Sensor Systems, November 2010.
- 6 Andrew Barry*, Benjamin Fisher*, Mark L. Chang, "A Long-Duration Study of User-Trained 802.11 Localization," Proceedings of the Second ACM International Workshop on Mobile Entity Localization and Tracking in GPS-less Environments, September 2009. (awarded best paper and best presentation)
- 7 Stephen Longfield, Jr*,., Mark L. Chang , "A Parameterized Stereo Vision Core for FPGAs", (Short Paper) IEEE Symposium on Field-Programmable Custom Computing Machines, April 2009.
- 8 C. Murphy*, D. Lindquist*, A.M. Rynning*, T. Cecil*, S. Leavitt*, M.L. Chang, "Low-Cost Stereo Vision on an FPGA", IEEE Symposium on Field-Programmable Custom Computing Machines, 2007.
- 9 I. Shafer*, M. Boes*, R. Nancollas*, J. B. Geddes, and A. Sieminski. Geometry and Stability of Microvessels. Submitted to Mathematical Medicine and Biology (2009).
- 10 D. Gardner*, Y. Li*, B. Small*, J. B. Geddes, and R. T. Carr. Multiple Equilibrium States in a Microvascular Network. Submitted to Mathematical Biosciences (2009).
- 11 M. Somerville and A. Horton*, "Centerpiece Projects", presented at Active Learning in Engineering Education, Boston, MA, 2003.
- 12 J. Stolk, S. Krumholz*, and M. Somerville, "Non-Traditional Assessments for New Learning Approaches: Competency Evaluation in Project-Based Introductory Materials Science," *Journal of Materials Education* 28 (1) 127-133 (2006).
- 13 J. Stolk and A. Dillon*, "How much can (or should) we push self-direction in introductory materials science?" Document 2006-1851, *2006 American Society for Engineering Education Annual Conference & Exposition*, Chicago, IL (2006).
- 14 J. Stolk, R. Martello, and S. Krumholz*, "Student-Directed, Project-Based Learning in an Integrated Course Block," Document 2005-1850, *Proceedings of the 2005 American Society for Engineering Education Annual Conference & Exposition*, June 12-15, 2005, Portland, OR (2005).
- 15 J. Stolk, S. Krumholz*, and M. Somerville, "Non-Traditional Assessments for New Learning Approaches: Competency Evaluation in Project-Based Introductory Materials Science," *2005 MRS Fall Meeting, Symposium PP: Forum on Materials Science Education*, Boston, MA, Paper 0909-PP04-05 (2005).
- 16 M. Somerville, D. Chachra, J. Chambers*, E. Cooney, K. Dorsey*, J.B. Geddes, G. Pratt, K. Rivard*, A. Schaffner, L.A. Stein, J. Stolk, S. Westwood*, and Y. Zastavker, "Work in Progress - A Provisional Competency Assessment System," *35th ASEE/IEEE Frontiers in Education Conference*, October 19 – 22, 2005, Indianapolis, IN, SIC-1 – S1C-2 (2005).

-
- 17 B. Sterling*, J. Stolk, L. Hafford*, and M. Gross, "Sodium Borohydride Reduction of Aqueous Silver-Iron-Nickel Solutions: A Chemical Route to Synthesis of Low Thermal Expansion-High Conductivity Ag-Invar Alloys," *Metallurgical and Materials Transactions A* 40 (7) 1701-1709 (2009).
- 18 E. Sterling*, L. Hafford*, and J. Stolk, "Rethinking Heat Sinking: Synthesis of Nanoscale Ag-Fe-Ni for Thermal Management Applications," *2007 New England American Society for Engineering Education Conference*, University of Rhode Island, South Kingstown, RI (2007).
- 19 B. Hollen*, J. Itescu*, M. Siripong*, and J. Stolk, "Cost Effective Fabrication of Ionic Polymer-Metal Composites," *2007 New England American Society for Engineering Education Conference*, University of Rhode Island, South Kingstown, RI (2007).
- 20 M. Siripong*, S. Fredholm*, J. Itescu*, Q.A. Nguyen*, B. Shih*, and J. Stolk, "A Novel, Cost-Effective Fabrication Method for Ionic Polymer-Metal Composites," *2005 MRS Fall Meeting, Symposium W: Electroresponsive Polymers and Their Applications*, Boston, MA. Paper 0889-W04-03 (2005).
- 21 C. Canfield* and Y. V. Zastavker, "Faculty on Integrated Project-Based Learning," *Academic Exchange Quarterly*, 13(1): 100-107, 2009
- 22 E. E. Blair,* R. B. Miller, M. Ong, Y. V. Zastavker, "Faculty Constructions of Gender Difference in Undergraduate Science, Technology, Engineering, and Mathematics Courses," *American Educational Research Association (AERA) Annual Meeting*, San Diego, CA, April 2009.
- 23 J. Baca* and Y. V. Zastavker, "Effects of Students' Course Conceptions on Role Differentiation within Project- Based Group Work", 2009 New England American Society for Engineering Education (ASEE) Conference, Bridgeport, CT, April 2009.
- 24 J. Baca* and Y. V. Zastavker, "Effects of Students' Course Conceptions on Role Differentiation within Project- Based Group Work", Joint Annual Meeting (JAM) 2009 Conference for NSF grantees, Washington, DC, June 2009.
- 25 C. Canfield* and Y. Zastavker, "Mathematics and Physics Faculty Conceptions of Teaching in a First-Year Integrated Project-Based Engineering Curriculum," 2009 American Society for Engineering Education Annual Conference & Exposition, Austin, TX (2009).
- 26 B. S. Tilley, Y. V. Zastavker, C. Laughlin,* and A. Dorsk,* "Unsteady Slider Bearing Dynamics," *Society for Industrial and Applied Mathematics Annual Meeting*, Boston, MA, Jun. 2006
- 27 K. F. Cummings,* C. D. Laughlin,* A. E. Lee,* J. Y. Tsai,* Y. V. Zastavker, and M. Ong, "They Speak: One-on-One Interviews with Female Engineering Students on Project-based Learning," *ASEE Regional Conference*, Worcester, MA, Mar. 2006.
- 28 O. Eris, D. Chachra, H. Chen, S. Sheppard, L. Ludlow, C. Rosca, T. Bailey, and G. Toye. Outcomes of a longitudinal administration of the Persistence in Engineering Survey. *Journal of Engineering Education* (in press).
- 29 T. Adams,* L. D. Braida, A. Hunter, B. Johnson, M. Jones, N. Khan,* M. Pierce, L. A. Stein, L. Tucker-Kellogg, S. Yeh, H. Abelson. *Women Undergraduate Enrollment in Electrical Engineering and Computer Science at MIT: Final Report of the EECS Women Undergraduate Enrollment Committee*. January 3, 1995. <http://www-swiss.ai.mit.edu/~hal/women-enrollment-comm/final-report.html>.
- 30 C. Henderson* "Robot World: A Learning Laboratory for Prospective Computer Scientists," June 1999 (M.Eng.).
- 31 E. Olson* "Otto: A Low-Cost Robotics Platform for Research and Education," May 2001 (M.Eng., also used for Sc.B.).
- 32 M. Bajrachaya* "Design and Development of a High-Performance, Low-Cost Robotics Platform for Research and Education," May 2001 (M.Eng., also used for Sc.B.).

-
- 33 S. Syme* "The Gender Effects of Introductory Computer Science at the Massachusetts Institute of Technology", (2000). Thesis, Kennedy School of Government, Harvard University. Professor J. Fountain, Advisor. Professor D. Zinberg, PAC Leader. Submitted to: L. A. Stein.
 - 34 M. Bajracharya* and E. Olson* "Low-Cost, High-Performance Platform for Education and Research", (2001). In Proceedings of AAAI Spring Symposium, Robotics & Education, Palo Alto, CA.

Research Experiences for Teachers (RET) in Engineering Education

Overview of project:

This RET, proposed in conjunction with the REU site for engineering education, will afford teachers of students in grades K-12 the opportunity to engage with and immerse themselves in original research projects with engineering faculty and undergraduate students. The purpose of this program is threefold:

- To enable teachers to participate in original research of a nature and scope that is usually not possible at their home institutions due to limitations of time, facilities, and equipment, as well as lack of mentorship;
- To develop a community of learners in engineering education by bringing together engineering professionals, educators, and K-16 students, and providing networking opportunities for further professional development; and
- To provide teachers of students in elementary schools, middle schools, and high schools with a unique targeted opportunity for professional development, as well as a support structure for further life-long learning and development.

By working with college faculty and undergraduates in research at this level, teachers will:

- Gain a greater understanding of methods of research in engineering education;
- Attain knowledge and skills for basic research in STEM-related educational research;
- Develop a greater understanding of current pedagogical and curricular practices in STEM (Science, Technology, Engineering, and Mathematics) at the K-16 level;
- Build professional relationships in the engineering education community;
- Fulfill professional development requirements as mandated for re-certification by the MA State Department of Education;
- Be able to use the new knowledge and skills gained through the program in development and implementation of classes taught upon the return to their own classrooms;
- Be able to critically reflect upon and evaluate their new teaching practices and use the tools gained in the program for further improvement and enhancement; and
- Evaluate their experiences at the REU/RET site and inform the program about the ways in which their participation is a transformative experience in terms of their teaching practices.

The RET program is structured as a six- to eight-week non-residential experience for K-12 teachers, to be held at Olin College during the summers of 2011, 2012, and 2013. (Length will depend on the schedule of the particular teacher participants and will be individually negotiated.) Planned activities include: participation in research, with close mentorship by faculty and in close collaboration with undergraduate student researchers; interactive sessions on various elements of educational research practice, facilitated by faculty from Olin and other academic institutions; mentorship on the basics of STEM education at the college level; opportunities for speaking engagements about teachers' own experiences at their home institutions; and social events to help foster a sense of community among the participants. In becoming members of a community, K-12 teachers will make professional and social connections with a variety of constituencies, who may influence not only the teachers themselves – directly or indirectly – but also their students. In particular, students begin to make decisions about heading towards further education in STEM well before they enter high school. The teachers and mentors that they have in grades K-12 can have a large impact on their perception of further education and careers in STEM fields; exposing these teachers to research in engineering education therefore helps expand the influence of this learning community.

The RET Experience:

Teachers in this program will work on a current project with Olin faculty and undergraduate students. Sample research projects are listed in the body of this REU proposal. A typical experience will include quantitative and qualitative data analysis based on the analytical techniques used within a specific project on the existing data sets. For many teachers, this will require learning the tools of education research, including methods, practices and ethical issues involved. Training sessions in these areas will be included in the teachers' programs. RET participants will learn both to use research methods to answer existing research questions as well as to ask new research questions based on the themes emerging from the data analysis performed. In addition, teachers will participate in a full-day workshop on creativity and generative thinking skills, a cornerstone of engineering design curricula as well as an essential tool in experiment design. We believe that these latter skills will be particularly important for teachers as they grow into practitioner-researchers upon returning to their home institutions with their own research questions and ideas.

RET participants will also be encouraged to continue working with the Olin faculty during the school year, to complete work begun during the summer session or to initiate new projects. At a minimum, they will participate in the design and conduct of assessment of the effects of the RET experience on their own classrooms (See 'Evaluation and Followup', below). Teachers will also participate in dissemination efforts through manuscript writing and presentations at conferences.

Teachers who participate in the six- to eight-week RET summer program at Olin will be able to meet several professional development goals. In order to be eligible for recertification in the Commonwealth of Massachusetts, teachers must earn 150 Professional Development Points (PDPs) in their primary professional license area every five years. We will work with the Commonwealth and the teachers on establishing the appropriate rubrics for assessment and evaluation of the PDPs. Teachers will have the opportunity to use new ideas and teaching strategies in their classrooms. RET participants will also be able to share their new knowledge with colleagues who did not attend the summer program, whether informally or by presenting on the topic of engineering and pedagogy. It is the aim of the RET to empower the teachers to develop as thoughtful, reflective practitioners, using their summer learning experiences to inform instruction. The teachers will continually evaluate their methods and their impact on student achievement.

Recruitment:

Teachers will be recruited locally from eastern and central Massachusetts to allow for ease of commuting and family-related accommodations. The recruiting efforts will target middle and high schools serving predominantly underrepresented groups and emphasizing STEM education. As specified in the RET solicitation, participants must be currently teaching a STEM subject at their institution in order to participate in the program. Teachers involved in the program will be chosen to represent a range of backgrounds within this constraint, to allow for breadth and diversity of educational experiences. We also aim to further broaden participation in engineering by including educators who not only serve a diverse and traditionally underserved student population but also those who teach in complementary academic disciplines. Of note, Massachusetts was the first state in the US to include engineering in the K-12 curricular standards; however, Massachusetts does not currently offer K-12 certification in engineering. We therefore will help qualifying educators in a range of disciplines to gain a greater understanding of, and confidence with, engineering education. The intention is to broaden the participants' conception of engineering as a field, as well as exposing them to current trends in engineering education.

For the inaugural summer session (in 2011), we will invite participation from teachers on the faculty of a specific school that meets our criteria: Spirit of Knowledge Charter School (SOKCS). SOKCS is a new, tuition-free public school with admission by lottery and no entrance examination. Located in Worcester, the second-largest city in Massachusetts, SOKCS' vision is to empower students to achieve academic

success through a rigorous, innovative curriculum consisting of sequentially organized, multi-year core-subject courses. With a focus on helping students achieve proficiency in mathematics, the sciences, and technology, SOKCS embodies a unique school culture based on a value-creating philosophy and a positive character-building system to foster academic achievement and personal growth.

SOKCS will open its doors in September 2010, with an enrollment of 156 students in grades 7-9. When at full capacity, the school will educate 275 students in grades 7-12. SOKCS' mission is to offer a rigorous program of STEM, humanities, and physical education to a traditionally under-served population.

Currently, more than half the students are eligible for free or reduced-cost lunch. Nearly one-third of the students are from families whose home language is not English. Approximately 17% of students have an Individualized Education Plan (IEP) or Chapter 504 accommodations.

Moreover, in alignment with the goals for the REU/RET site recruitment efforts, approximately half the students enrolled are female. Furthermore, the racial/ethnic demographics of the student body (based on the information received from approximately 85% of families who returned surveys before the start of school) is 28% African-American, 24% Hispanic, 9% Asian, 1% American Indian, with the remaining percentage of students being white, non-Hispanic. SOKCS is committed to serving the youth of inner city Worcester, and nearly 95% of its currently enrolled students reside within the city limits.

The rationale for targeting SOKCS for the first year of the program is twofold: First, SOKCS is very similar to Olin College, in that it is ‘starting from scratch’ as Olin College did in 2002. A number of the senior personnel on this REU/RET proposal were involved in creating the curriculum at Olin, and this experience has informed their research. Further, as a new school, SOKCS is in an excellent position to incorporate insights and research plans developed as part of (or inspired by) the RET; we therefore see working with SOKCS teachers in the first year of this program as likely to have the greatest impact. A letter of commitment from SOKCS is appended.

In subsequent years, we will use local networks to identify and recruit (pairs of) teacher participants. Olin has pre-existing relationships with public school districts including Needham and Wellesley – our local districts – as well as Framingham and Roxbury – both minority-serving areas – and with Dana Hall, a local private school for girls. We also have collaborations with local organizations such as Tech Hub (a state-wide network addressing computing education in K-12), METCO (a minority-serving cross-district placement service), and Boston’s Museum of Science (which has a significant educational outreach program).

Evaluation and Followup:

The three components of the RET – K-12 teachers’ engagement in research, development of a community of learners in engineering education, and professional development – are designed to broaden participation in engineering by creating new learning opportunities, forging professional relationships, and building foundations for more meaningful classroom experiences and inquiry.

Participants in the RET program will complete evaluation rubrics similar to those given to the REU participants – concerning attitudes towards engineering, self-image and identity, etc. – as well as participating in a RET-specific evaluations. Most important among these will be pre-, post-, and retrospective RET interviews to gather data on the classroom experiences of the participating teachers. In the pre-RET interview, teachers and their mentors will lay out classroom-specific goals for the teachers during the subsequent school year, including construction of explicit metrics. These goals and metrics will be revisited in the post-RET interview. We anticipate that most metrics will be similar to those used by Olin faculty members in our current research projects, with RET participants selecting among these the ones that best reflect their own classroom goals. (For example, RET participants may wish to incorporate creativity/generative thinking skills instruction into their classrooms and might then use either an innovation battery or an intrinsic motivation rubric to assess impact.) Olin faculty members and the RET participants will then conduct appropriate measurements during the subsequent school year.

Collaborative Research, TUES-Type 1: Fostering Student Learning Continuity Employing a Personal Active Learning Platform

Gunar Schirner, David Kaeli, Mark L. Chang, Mark Somerville, Mihir Ravel

August 25, 2010

This is a cover page for identification purpose only. Neither the cover page nor the table of content will be submitted to NSF.

B. Collaborative Research, TUES-Type 1: Fostering Student Learning Continuity Employing a Personal Active Learning Platform

A significant amount of engineering education is spent presenting engineering theory in the classroom, followed by practice of the theory in a laboratory experience. There is little opportunity for students to immediately apply the theory to the solution of practical problems. This disconnect negatively impacts student retention of the engineering theory, and further distances theory from practice.

A second issue in engineering education is the time gap between courses and course topics; how do we keep students engaged in the discipline while they are away from the classroom either on internships or co-ops? These time gaps allow students to forget critical material that must be recovered in the next class - these time gaps provide opportunities for active learning by students on their own. Presently, little is done to insure students can hit the ground running when they begin a new class.

The key curricular innovation proposed in this project is to exploit an active learning educational platform to specifically impact the Electrical and Computer Engineering undergraduate experience at both Northeastern University and Olin College. The enabling technology is the introduction of a personal educational hardware/software platform that will allow students to practice theory in the classroom on their Personal Active Learning (PAL) system, as well as pursue continuous learning outside of the classroom as they pursue open-ended design experiments while not on campus. The PAL system has already been used in classes at both schools, and based on the feedback, students have been requesting their own personal system to continue their design explorations well beyond the end of final exams. The goals of this project include:

- The development of courses at each school (minmally 4 courses total) that will explore the benefits of introducing the PAL system to allow students to immediately connect theory to practice during lectures, as well as to experience the synthesis process in the classroom,
- The development of a set of open-ended active learning modules that students can access via the Internet and will allow them to both reflect on their past classroom experiences, as well as to prepare for their next semester's class, and
- The implementation of an instrumentation system that will allow us to assess how student utilized the PAL system while both in school, as well as when they are learning on their own.

The results of this project will include:

- The delivery of a new learning model for students to develop stronger design-based skillsets, as well as a deeper appreciation and increased retention of theory-based course content,
- An evaluation of the educational value of using PAL in an integrated classroom/active learning environment,
- The development of a rich set of open-ended design assignments that allow students to experiment on their own time, and
- An assessment of how student utilize the PAL platform, which will help the research team improve upon the current PAL hardware/software design.

Intellectual Merit: The proposed implementation and assessment of an active learning platform will improve student learning and theory retention both inside and outside of the classroom. We will also explore how best to instrument the Personal Active Learning platform to capture student experience profiles while they use the system in an unstructured learning environment.

Broader Impact: This project will both impact undergraduates at Northeastern University and Olin College, and is also slated for adoption at other area Boston-area schools such as Boston University, Tufts University and MIT. Further, the educational research related to both student learning and platform instrumentation will be reported in the literature. The PIs will provide tutorials at technical conferences on the PAL system and associated learning model.

C. Table of Contents

Contents

B. Collaborative Research, TUES-Type 1: Fostering Student Learning Continuity Employing a Personal Active Learning Platform	B-1
C. Table of Contents	C-1
D. Project Description	D-1
1 Introduction	D-1
2 Background	D-2
2.1 Approaches to Integration	D-2
2.2 Approaches to Life-long Learning	D-3
2.3 Integration of Theory and Practice, Educational Efficacy	D-3
3 Approach	D-4
3.1 Active Learning Continuum	D-4
3.2 Enabling the Active Learning Continuum through a Personal Active Learning Platform	D-6
4 Course Development	D-7
4.1 Computer Architecture	D-7
4.2 Microprocessor-based Design	D-8
4.3 Principles of Engineering	D-9
4.4 Digital Systems	D-10
5 Assessment	D-10
6 Preliminary Work	D-11
6.1 Initial Experiences with Personal Active Learning Platforms	D-11
6.2 Microprocessor-based Design at Northeastern	D-12
7 Plan of Work	D-13
7.1 Dissemination	D-13
8 Qualifications of the PIs	D-14
9 Prior NSF Support	D-15
10 Coordination between Institutions	D-15
E. References Cited	E-1

D. Project Description

1 Introduction

Over the last twenty years, the National Science Foundation, the National Academy of Engineering, and the engineering community in general have called for systemic changes in engineering education, including increasing student capacity for life-long learning, enhancing students' abilities to engage in system-level thinking, improving the efficacy of engineering education, and incorporating more engineering practice and design throughout the curriculum [34] [12] [36]. Despite these calls, change in engineering education remains a very slow process. In most programs, the engineering coursework is structured around a classroom theory course, and practice is confined to a concurrent laboratory. Such an approach gives little opportunity for students to immediately apply the theory to practical problems, and often leads to low student engagement, low retention of engineering theory, and little understanding of what engineers actually do.

A promising model for effecting change in engineering education is hinted at by Traylor et al.: one can start with a *technology* that facilitates change. Traylor proposes the concept of a “Platform for Learning” – a common unifying object that weaves the student experience together across multiple courses [53]. As Traylor notes, “using a common platform throughout a degree program [can enhance] the integration of knowledge. The platform provides the conceptual “glue” ... Interactions between topics becomes clear.”

This proposal suggests a model for changing electrical and computer engineering education starting with a platform: the Personal Active Learning (PAL) system, a heterogeneous mobile development platform. In particular, the proposal suggests that by introducing a highly-configurable and flexible experimental platform, we can develop consistency and integration across the engineering curriculum, connect theory to practice, enhance student engagement and self-direction, increase students' system-thinking abilities, and most importantly, *facilitate* – rather than dictate – pedagogical change.

The PAL approach is compelling for several reasons:

- First, it facilitates student-centered vertical integration. By utilizing the same platform in multiple classes, the PAL system helps students to see the coherence between subjects, and make connections between subjects, not just within a semester, but across the entire curriculum.
- Second, the system facilitates integration of theory and practice within and outside the classroom. Within the classroom, the system's small size and heterogeneity of hardware allow it to be used for “studio” exercises – even in constrained traditional lecture spaces. Outside the classroom, the system allows students to put theory into practice by realizing actual working prototypes.
- Third, the system enhances student ownership and autonomy. It is imagined that the PAL system will be owned by the student¹, and consequently will be available for learning 24/7 – even outside of the semester. Our experience to date suggests that students are excited about engaging with the system for self-directed experimentation and design. Such self-directed work is a key to developing life-long learning skills.
- Finally, the PAL is truly a platform for realizing real world designs. Its flexible, heterogeneous architecture allows students to design and realize products ranging from sensor systems to mobile video players. Utilizing such real-world examples is critical for encouraging student self-direction and engagement.

We propose to conduct this work at two complementary institutions that in close geographical proximity, cover a significant educational spectrum. We will use educational heterogeneity to guide us in developing transferable course modules and exercises. Northeastern University (NEU) is a research university with traditionally strong ties to undergraduate education. NEU is widely recognized as a leader in cooperative education, and has strong ties with industry. The emphasis at NEU is on providing a practically relevant education. The student body at NEU is quite diverse, and in particular includes a large population of non-traditional students. NEU provides a realistic test bed for implementing the system in a research university environment. The

¹ PAL systems can also be made available on long term loan by the university.

Franklin W. Olin College of Engineering is a relatively new undergraduate-only engineering college (students were first admitted in 2002). During its existence, Olin has established a reputation for project-based education and for approaches that increase students' creativity and entrepreneurial skills. Due to a flexible curricular structure, it is relatively easy to test new course and project ideas at Olin.

The PAL system has already been used in classes at both schools, and based on the feedback, students have been requesting their own personal system to continue their design explorations well beyond the end of final exams.

Specific activities that will be carried out at Northeastern University and Olin College include (1) development of courses at each school (minimally 4 courses total) that will explore the benefits of introducing the PAL system in the classroom; (2) development of a set of open-ended active learning modules that students can access outside of the classroom; and (3) implementation of an instrumentation system that will allow us to assess how student utilized the PAL system while both in school, as well as when they are learning on their own.

The impact of the educational technology and modules developed in this project will include (1) development of a new learning model for engineering students to integrate and appreciate design-based skill sets as well as theory-based course content; (2) evaluation of the educational value of using PAL in an integrated classroom/active learning environment; (3) development of a rich set of open-ended design assignments that allow students to experiment on their own time; and (4) assessment of how student utilize the PAL system, which will help the research team improve upon the current PAL hardware/software design.

2 Background

This proposal suggests the development, assessment, and dissemination of new courses and learning modules centered around a novel Personal Active Learning system, which will facilitate integration, life-long learning, and increased educational efficacy. This section reviews previous work in these areas.

2.1 Approaches to Integration

Within engineering education, the need to increase integration between subjects has been acknowledged for many years [6], and there have been numerous reforms aimed at increasing integration. The most prevalent strategy for enhancing integration has been what might be termed *horizontal integration*: in a given semester, students might take two or more subjects in an integrated fashion. [19] Examples of horizontal integration beyond the first year or two of engineering curricula are relatively rare, and in many cases, horizontal integration has been difficult to sustain, due to resource constraints, changes in personnel, and institutional politics [9].

Capstone design projects also present an integration opportunity, as students are typically challenged with design problems that require them to apply material from multiple previous courses. This sort of integration might be termed *summative integration*; unfortunately, many instructors report that students' ability to recall and apply material from previous courses is not as strong as might be desired.

The approach to integration suggested here might be termed *vertical integration*: rather than connecting one course to another within a given semester, the use of a common platform, accompanied by appropriate learning modules, can foster integration between semesters. In contrast with horizontal integration, in which connections between subjects are largely driven by the instructors, the vertical approach suggests that an enabling technology – the PAL system – combined with appropriately designed materials can foster continuity and connections between subjects[44].

Such an approach is relatively rare within ECE education. Perhaps the most comprehensive example of such vertical integration is the proposed “platform for learning” strategy outlined by Traylor et al at Oregon State University [53]. Traylor et al. define a platform for learning as a ‘common unifying object or experience that weaves together the various classes in a curriculum”, and they suggest that such a platform can serve as a physical “manipulative” to enhance learning. Using this framework, Oregon State has developed the TekBots

system, which is used in multiple courses throughout the ECE curriculum, from the first year introductory ECE course to several upper-division classes. The TekBots approach begins with a very simple motorized robot; over the course of their four years, students increase the capability to the robot by, for example, adding a digital controller board, a microcontroller board, an FPGA system, an IR communications board, and perhaps a microprocessor board. The TekBots approach has also been adapted by a few other schools[44].

Two alternative approaches include Virginia Tech’s move towards a unified platform in Computer Engineering using an FPGA board, and RPI ’s mobile studio. At Virginia Tech, sophomore, junior, and senior students are required to purchase a Digilent Spartan3E Starter Kit, which is being used in approximately five courses across the CpE curriculum there. Initial results of the program seem promising [3]. At RPI, the mobile studio approach gives students a pc board that, combined with a laptop, provides the functionality of a scope, a function generator, a multimeter, a power supply, as well as providing digital I/O channels. [33] Mobile studio has been adopted at Rensselaer and at Howard University, in approximately five different classes.

2.2 Approaches to Life-long Learning

STEM educators have long understood that engineering students’ development of life-long learning skills is vital for their success in today’s global and rapidly changing technological environment [34], and ABET requires engineering educators to demonstrate the development of students’ life-long learning skills through their curricula [10]. Achieving the long-term outcome of life-long learning requires that learners gain competence in self-direction [7]. The UNESCO Institute for Education noted that life-long education, as a means for promoting life-long learning, is dependent for its successful implementation on people’s increasing ability and motivation to engage in self-directed learning activities’ [13].

While there is no universal agreement on how best to foster self-direction, and hence life-long learning, research suggests that project-based and team-based learning can be more effective at developing self-directed learners who become life-long learners. [17], [62] STEM educators are making significant strides in this regard, through re-design of traditional programs to incorporate greater student autonomy and increased project-based education, and the results are promising. [1] [55] [56] [25] It is likely that the efficacy of project-based approaches is in part due to the extent to which these pedagogies give students freedom, choice, control, and ownership. Indeed, research suggests that the learner’s sense of autonomy is key to self-motivation, internalization of learning goals, self-efficacy, and task value [14] [40]. [21].

2.3 Integration of Theory and Practice, Educational Efficacy

Active learning pedagogies – from Peer Instruction to Project-Based Learning – yield demonstrably better learning outcomes than traditional “chalk and talk” strategies.[45] In physics education, the use of mini-lectures and collaborative “conceptests” has been shown to substantially increase students’ comprehension and retention of fundamental physics concepts [30]; this approach has also been adopted across a variety of engineering topics. [22] Similarly, the use of “studio instruction”, in which lecture is combined with hands-on practice has been shown to be effective in both physics and engineering [59]. Finally, as noted above, there is an increasing trend to the incorporation of project-based education into engineering; research indicates that project-based education leads to stronger learning outcomes than traditional strategies, especially with respect to the development of professional competencies, but also with respect to retention of key technical concepts.[26]

Such pedagogical innovations are particularly important as we consider the challenges facing engineering educators in the next decade. First, there the ever-present need to help students *bridge the gap between theory and practice* – a need that is only amplified as technological change makes both theory and practice more complex. Second, there is an increasing need to help students become systems thinkers [34] – engineers who not only understand the details of a given subsystem, but who understand and can make tradeoffs between subsystems.

3 Approach

Current undergraduate engineering education programs face several challenges - we attempt to address a number of them in this project: (a) minimizing the time gaps between the presentation of theory and the opportunity to practice it, (b) addressing the needs for a more integrated interdisciplinary curriculum that fosters system-level thinking versus that standard practice of presenting a collection of disconnected point-approaches, and (c) stimulating students to pursue life-long learning.

These pedagogical challenges are intertwined with practical and organizational challenges. First and foremost, the infrastructure overhead can be quite large when introducing project-based courses as well as studio-based learning. Current ECE curricula span a wide range of subjects, and given the rate of innovation in ECE-related research, the range of topics to be covered continues to grow and evolve. Given this increase in breadth, however, presents challenges to integrative learning. The amount of coordination between subjects increases, and students will be exposed to disconnected islands of subject matter. We need to rethink how best to enable integration across multiple dimensions.

In this proposal, we aim to enable new forms of both horizontal integration and vertical integration by utilizing a *common Personal Active Learning (PAL) platform* that is *student-owned* and lends itself to experimentation throughout the engineering curriculum. Using a common student-owned platform fosters student-centered subject integration and reduces the synchronization effort between facilitating faculty. In addition, a common student-owned platform opens the door to more widespread independent exploration throughout a course. We plan to develop a new *studio-based classroom experience* that alternates between small lecture modules, conveying the necessary theoretical background, combined with in-class experimentation, enabling the students to practice the concept almost immediately. Second, we propose to use the PAL for lab exercises off-line, such as at home, at the library, between classes, and between semesters.

The next sections describe our approach in detail. We will first describe the pedagogical aspects of the active learning continuum which we plan to establish. Second, we describe the actual PAL system that will be used in this project.

3.1 Active Learning Continuum

With the work in this proposal we aim to enable an active learning continuum (Fig. 1) and develop a number of integrated learning experiences leveraging a common student-owned PAL system. We envision the following enhancements and learning objectives:

Increase learning efficiency over time We envision that PAL will accompany students from their first year introductory courses through their final capstone projects.

Enable location independence (Section 3.1.2). With the 24/7 availability of the personal platform, we envision expanding past traditional lab-based learning locations and hours.

Integration over subject matter (Section 3.1.3). Finally, with the added flexibility of a heterogeneous PAL, we will integrate over subjects from introductory overview courses, core subjects, to specialized electives.

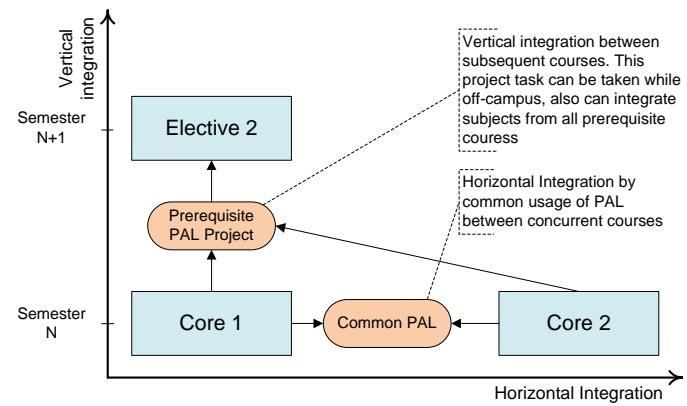


Figure 1: Active learning continuum.

3.1.1 Learning Integration over Time

Students will purchase a PAL for the price of 2-3 regular textbooks in the first year of the curriculum. As Section 3.2 will outline, PAL is a heterogeneous platform providing two computation fabrics, a Digital Signal Processor (DSP), a Field Programmable Gate Array (FPGA), and many sensors that are geared toward a wide range of classroom experiments. The platform is also extensible to accommodate special course requirements.

We plan to introduce PAL directly in the first year into an introductory course, such as the Olin College Principles of Engineering (see Section 4.3). As an outgrowth of this project, we have similar plans for change in the first-year required core at Northeastern. One challenge in introducing PAL early in the curriculum is that students have very little ECE discipline knowledge. Course materials will include mainly ready-made modules, such as a pre-implemented data acquisition module, which will dramatically simplify working with the platform. At this stage, the goal is to provide modules that enable students to compose the experiment's solution without the burden of low level implementation details. These introductory courses are meant for showcasing the subject areas of ECE, and are designed to spark the student's interest to explore more.

During the 2nd and 3rd year in the curriculum, students will begin to take ECE courses. Here PAL will serve as the main hands-on vehicle to practice the theory presented in the classroom. Example courses include Digital Systems Section 4.4 in which students discover combinational/sequential logic design and finite-state machine design, or Computer Architecture Section 4.1 which presents architecture as an abstraction of the physical hardware, and discusses the hardware/software interface. Given that students will already be familiar from using PAL from the first year, the amount of course startup time will be reduced.

In the later years students focus on elective courses oriented toward specialized subject matter such as Digital Signal Processing (DSP), Biomedical Engineering, or Control Systems. Here the core concepts of the previous courses will be required, which students have already experimented with them on their PAL and so we expect that more of the necessary preparatory material has been mastered. By vertically integrating and synthesizing the different learning experiences with their PAL, seniors can aggressively pursue more advanced capstone projects.

Due to the use of the common PAL, we anticipate a significant reduction in infrastructure overhead. Instead of creating a new lab or experimental environment for each course, common modules can be shared between courses (and faculty members). More importantly, from a student perspective, all PAL-based courses are offered with a consistent set of tools. Students can become proficient in the tool set and thus can focus their learning energy on the subject matter and concepts instead of learning a new environment for each course. Conversely, from an instructor's perspective, less time has to be devoted to lab introductions and tool lectures.

3.1.2 Encouraging Learning Location Independence

Introducing a mobile experiment platform, such as the PAL, can eliminate confining experiments only to designated laboratory classes, and thus broaden the impact of experiential education. The positive impact following a similar approach on learning outcomes in STEM education have already been reported for the Mobile Studio environment [33].

Studio-based learning has shown to be an efficient means to actively engage students in class settings and to provide the immediate connection between theory and practice. [24, 4], for example, show the successful adaptation to computer science courses through the use of notebooks in the class setting. We will introduce the studio-based learning approach in courses that traditionally have a larger gap between theory and practice due to the need of specialized laboratory equipment. With a student owned mobile platform, such as PAL, this equipment can now be made available with low infrastructure overhead right in the classroom setting.

Lab sections benefit from the location of the integrated learning. Before starting a new lab, students have already explored the basic practical principles in the classroom setting. Therefore, lab exercises can build upon this experience much more directly, and tackle more complex problems that require students to combine and synthesize different theoretical concepts. The further integration into offline experiments offers an additional degree of freedom. Homework assignments, which necessarily have focused on theoretical aspects,

can now integrate both practical and theoretical components to combine design, implementation and analysis. By expanding the experiments from the time constrained laboratory setting to the more individually controlled home environment, a new set of exercises becomes possible that foster project-based-learning [26]. Without the time pressure in the lab, students are more motivated to explore open ended projects.

Finally, the time integration of the learning continuum can also expand to even before the actual course start. This can be used for preparing students for the course start. This is of particular interest for Northeastern University with its strong commitment to co-operative education. Here students are off campus in an industrial setting for three 6-month periods. While this is very beneficial for ensuring practical relevance and professional preparation, it poses challenges of getting students back into *classroom learning mode*. In our model, consider the scenario that involves small encapsulated project building a gadget that stimulates student creativity even before starting the course. The project can be designed to refresh material covered in previous semesters (particularly math and programming concepts) and It should achieve a much higher retention and ensure that student "hit the ground running" when they return to campus.

3.1.3 Connecting Diverse Subject Matter

The third dimension of the active learning continuum addresses how best to connect a range of diverse engineering topics, enabling students to see the links between specialties and allows students to form a coherent view of the curriculum. By presenting course material in a common learning environment that leverages a common hands-on system (PAL) lends itself to connecting course subjects. Next, we provide a few examples.

A strong connection exists between the core subjects of Digital Systems, Computer Architecture and Microprocessor-based Design. These classes are target of our first round of implementation and will be described in Section Section 4. One course typically builds on content presented in the earlier course. This is one curricular aspect we intend to strengthen with the introduction of the PAL. We plan to measure this improved connectiveness in our assessment.

A second example is targeted for our follow-on work, where we add Digital Signal Processing to our PAL-enabled set of courses. By using the PAL in both Computer Architecture and DSP, now students can build heterogeneous systems that carry out both signal processing and general purpose computing. With the PAL system architecture (with processor and FPGA) this type of system synthesis becomes more natural.

3.2 Enabling the Active Learning Continuum through a Personal Active Learning Platform

A number of educational institutions such as OSU, RPI, and Virginia Tech are adopting platform approaches for bringing innovative learning into the curriculum while also achieving significant cost, efficiency, and space benefits [44, 33, 3]. Although the use of platforms is rather recent in academics, it is a well established practice in industry where platforms have been the basis of product development for several decades [32, 58, 43, 42, 8]. To support active learning across the ECE curriculum, and guided by feedback from global faculty using an initial version of PAL (see Section 6.1), we defined key properties for a platform aimed at student personal learning:



- Heterogeneous technologies covering a broad set of ECE domains
- Support system thinking exercises in partitioning and tradeoffs
- Sufficient performance and resources for real-world design projects
- Hardware supports both open source and commercial design tools
- Portable "look and feel" to engage students' imagination

Based on these needs, PAL (Fig. 2) has been developed in a collaboration between faculty (at Olin College, U-Texas, Northeastern), and several ECE-

centric industry partners (Analog Devices, Xilinx, The Learning Labs). This result is a second generation PAL system architecture composed of modular elements for the generic ECE functions of communications, computing, and physical world interfaces. The PAL system is a significant evolution over the initial PAL featuring a simplified architecture and portable form factor that can engage students' imaginations while still supporting ECE education needs.

Fig. 3 shows the PAL architecture with links to relevant course domains. The PAL system provides heterogeneous computing fabrics of a 600 MHz RISC-DSP processor and a Field Programmable Gate Array (FPGA). Together they span a range of processing from low-level logic programming using HDLs, to embedded programming in C/C++, and up to high level Python on the PAL's embedded Linux operating system. PAL offers a rich set of interfaces for projects and circuit/sensor exercises: 24bit/96KHz audio CODEC, accelerometer, power and temperature sensors. The networking interfaces (USB, Ethernet,Serial) allow

distributed systems to be prototyped along with mapping to communications concepts. An engaging human interface which includes a 320x240 color touch screen, and cap sense scroll wheel and buttons gives the feel of real-world consumer electronics. The 1.3MPixel camera and LCD display allows both student creativity and mapping onto vision/imaging domains. The architecture is designed for user expandability through high speed parallel IOs and standard serial interfaces (SPI, I2C). This architecture allows for courses to use end-to-end examples, such as image processing loops (video recording, processing, output), signal processing of audio data, and machine control. Naturally no platform can cover all ECE domains, so the PAL emphasis is to allow user customization tailored to the unique teaching needs - the platform approach is an enabler rather than a fixed solution. Since the system is based on global faculty feedback and developed collaboratively with educators rather than being a purely industry product, we expect the proposed PAL system has sufficient scalability in performance and peripheral expansion to support learning experiments from introductory engineering up to advanced elective topics.

4 Course Development

We propose to update four courses for using PAL. Computer Architecture Section 4.1 and Microprocessor-based Design Section 4.2, will be upgraded at Northeastern, analyzing vertical integration between core subjects in the curriculum. At Olin College, we will evaluate the interaction between a multi-disciplinary course – Principles of Engineering Section 4.3, and a core course Digital Systems Section 4.4. The selection of courses not only represent good technology/classroom targets, they also represent radically different pedagogical models, students, faculty approaches, and even course objectives and expectations – which allows us to reasonably broadly explore the impact of our proposed PAL approach. Fig. 4 outlines these courses in time with the NEU curriculum names (in bold). Courses for future adoption are shown in light blue.

4.1 Computer Architecture

A key required class in most Computer Engineering and Computer Science curriculums is Computer Architecture. It is in this class that students learn about the hardware/software interface. They are exposed to concepts such as an instruction set architecture, assembly code, pipelining, branching, synchronization, arithmetic precision, addressing and caching.

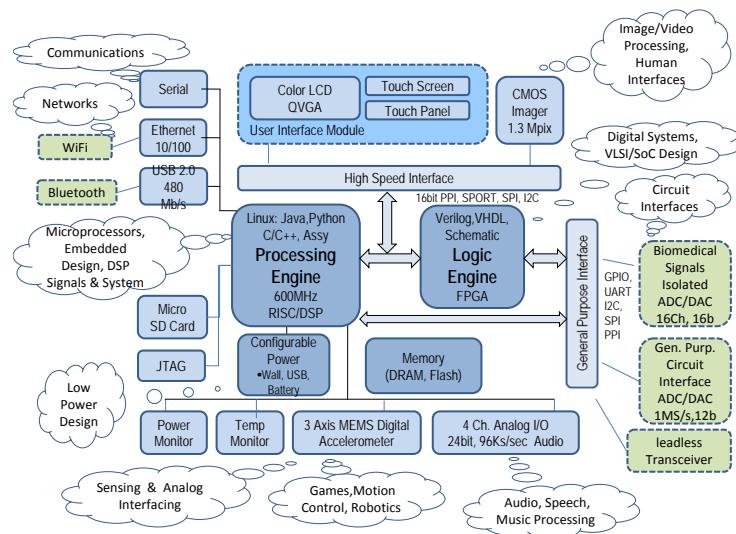


Figure 3: Personal active learning (PAL) system

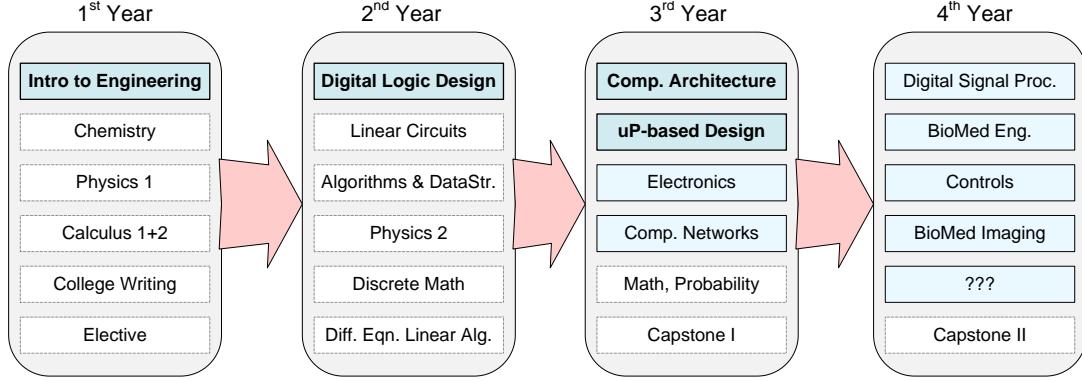


Figure 4: NEU computer engineering curriculum excerpt.

Teaching these topics presents a challenge to an instructor. Simulators and tools can be used in a lab or in the classroom to connect the theory to an actual implementation [60]. Some examples include the SPIM simulator [27] for the MIPS instruction set, and the Dinero [16] cache model. Many times a design lab can accompany the classroom course, providing students with experience implementing an architecture on a Field Programmable Gate Array (FPGA).

While both of these approaches can enhance the classroom experience, they each possess limitations that are addressed in this proposal. First, the software simulators used in the classroom have very limited capabilities, and do not allow students to interact with real hardware in the classroom. While students can immediately practice the theory presented, the fact that software simulator is still an abstracted version of the actual system limits its ability to impact student retention of the theory. Second, in the FPGA implementation, student wait until they are in lab to practice the theory presented in class. These time gaps are critical and further impact student retention of the classroom material.

By introducing the use of the PAL in the classroom, we will be able to implement theoretical concepts in the classroom to support the retention of any concept. For our pilot student we will develop two different hands-on experiences to support topics in computer architecture that students tend to have difficulty with. The two topics selected will be: pipelining and caching. Student will be able to implement a pipelined processor on the FPGA present on the PAL, measure performance (in terms of cycles) to demonstrate the value of pipeline, and finally will run real programs on their design to demonstrate correctness. The pipeline design will be built incrementally, using a standard 5-stage design as presented in many texts [23, 35]. For studying cache design, students will have to use the cache of the Blackfin Digital Signal Processor (DSP) that is on board. The memory system of the Blackfin is highly configurable [5, 41], and provides a perfect interface to explore the impact of different caching algorithms on performance. The students will also have the ability to add a simple cache to their pipeline design on the FPGA as part of an open-ended homework assignment. The key will be that once students start using the PAL in their early classes, the learning curve to utilize the FPGA and DSP hardware will disappear.

4.2 Microprocessor-based Design

This course provides an introduction to embedded systems, and embedded system design combining both software and hardware aspects. It integrates domain subjects such as embedded computer architecture, systematic design, and interfacing with real-world sensors and actuators. It has the objective to enable students undertaking a complete embedded product cycle: from analyzing the idea all the way to validating the HW/SW implementation. Given the strong project-based focus and wide topic diversity, the course naturally benefits from the PAL system. Initial experiences with the PAL are very promising Section 6.2.

In the upgrade, the student owned PAL will be fully utilized in various learning activities. A *pre-course exercise* will be developed to engage students before the course start. Students will be tasked to record and

display platform motion using ready-made libraries for sensor reading and graphical output. In addition to familiarizing students with the PAL, this gadget exercise builds confidence and we anticipate sparking student curiosity about on the inner workings of an embedded system.

A *studio-based learning* activity may center around the theory topic of interrupt handling. Interrupts are asynchronous events that are commonly used for I/O operations and operating system services. After following a lecture module, outlining interrupt concepts, students will transition to an in-class experiment that explores the sequence of events occurring during an interrupt. Students will collect actual measurements of the interrupt latency for a given application. Naturally, students will get a range of results, which sparks the discussion for the subsequent lecture module: interrupt latencies.

The *lab experiments* will build on the interrupt latency classroom experiment and focus on developing an interrupt handler for a media player. The in-lab experiment can further connect to a practical homework assignment that leverages the PAL to experiment with interrupt sensitivity (e.g., level/edge, pos/neg). The same concept of interrupt handling reappears in later labs as a modular building block, allowing students to experience the advantages of design reuse. One example is a periodic power analysis module for evaluating the power profile of an application. Based on our past experience with using PAL in this environment, students will develop power-efficient embedded systems considering interrupts. This is a good example of horizontal integration in a curriculum.

By systematically utilizing PAL, we anticipate that students will be able to better master the course principles, while also covering more material (e.g. due to more overlap with other classes). In past versions of this class using PAL, our students have constantly asked for longer lab times to accommodate their imagination. Thus, we expect the greatest benefit will be related to learning location independence, allowing students equipped with their personal platforms to explore and experiment autonomously.

4.3 Principles of Engineering

Principles of Engineering (PoE) at Olin College is a required "introductory engineering" second year course for all Olin students. Starting with studio exercises and ending with an open team-design project, students learn to integrate concept definition, analysis, design, prototyping and test to improve their ability to engineer real systems. Projects usually include both a electro-mechanical system design and an electronic-computing system design involving both hardware and software components from circuits to laptop.

Currently, for the first third of the semester, groups work with a Microchip PIC18F microcontroller in studio exercises aimed at building simple peripherals based on the PIC and the student's low-level C code and custom circuit. The PIC18F is easy to use as a USB device to transfer data between a host computer and the student-designed peripheral, but this is problematic for two reasons. First, some teams use this approach in order to escape having to develop code that runs adequately on the microcontroller, using it only as a "bridge" to the mechatronic system they built. Second, some teams have computational requirements that exceed that of the microcontroller and therefore must offload computation.

The PAL enables powerful local processing and provides a familiar programming environment as it can run Linux and high-level languages such as Python. This encourages software-centric students to safely start exploring hardware concepts. In an informal project-based seminar run at Olin in Spring '10, several "pure" software students became so engaged with their PAL project ideas (handheld ECG, balloon sensor package) that they spent the most time on learning circuit concepts. Without tethering to a laptop for added resources, the portable and compact PAL allows students to explore a broader range of technical complexity and realism for their projects.

In the first two iterations of PoE, offered in Fall 2010 and Spring 2011, a subset of student teams will use the PAL for the project portion of the course, labeled "Experimental PAL" in Table 1. This will allow assessing the student reaction to the platform as an input to development of course materials. We will then update the course with the PAL as a separate "track" to the PIC18F. The impact of this course modification is expected to be wide within the Olin community. Currently, the PIC18F is the preferred path for many student self-directed

projects. Increasing the student body expertise to more capable technologies could enable a whole new range of student projects.

4.4 Digital Systems

Digital Systems is the name of a new course being offered in conjunction with this proposal. The course will build from a base of three previous third year offerings in the same subject area, *Embedded Systems*, *Advanced Digital Systems*, and *Principles of Intelligent Systems Engineering*. The previous courses aimed to expose students in the third year to hardware/software co-design, rigorous digital design, and the gap between simulation and synthesis in FPGA design.

The new Digital Systems course will offer a natural follow-on learning experience from PoE and Computer Architecture, and will utilize the PAL platform familiar to students from earlier course use. This will initially provide an opportunity to study the effects of using the same platform longitudinally. Where the PAL will provide most benefit can be seen from the details of the previous trials of the course.

In the first iteration, Embedded Systems, hardware/software co-design was explored by having students combine a PIC18F, a Xilinx-based FPGA board, and several peripherals (OLED display, MP3 decoder, wireless links, SD card). Students spent much time building communication mechanisms—both HW and SW—between these devices, which was frustrating and unproductive as there was no common standard utilized among devices. Introducing a common platform will provide a much higher level of device integration. Students would not have to “reinvent the wheel” for each interface, and can concentrate on the hardware/software divide and optimizing for appropriate resource utilization.

In the second iteration, Advanced Digital Systems, we used a Xilinx FPGA board with a soft-core processor to provide the C-programmable resource. While the integration between CPU and reconfigurable fabrics was tighter and higher speed, the toolchain (the Xilinx Embedded Development Kit) had a very steep learning curve. Additionally, this hardware platform did not solve the peripheral problem of the previous iteration, and debugging, visibility, and interactivity were equally frustrating. Introducing the PAL would give students easy access to reliable peripherals while allowing exploration of soft-core CPU instances where the HW/SW tradeoffs are well exposed.

Finally, in the Principles of Intelligent Systems Design course, we utilized the first generation PAL in a “lab-based” use model. The well integrated and heterogeneous HW/SW architecture inspired students to tackle realistic design concepts (audio special effects, physics engines), but since access to the platforms was limited, students did not have as engaging an experience as we had hoped. With the more familiar mobile-design theme of the PAL platform for this project, our hypothesis is that student engagement will be improved with greater project scope and increased self-directed learning

5 Assessment

The central goal of this proposal is to enable an active learning continuum. To this end, we have described the following objectives: integration over time, integration over subject matter, and enabling location independence. Given these goals, we plan to conduct a number of evaluations using both validated instruments and instruments developed as part of the proposed work.

Integration over time and **integration over subject matter** can be measured both quantitatively and qualitatively. Quantitatively, we will use classical measures to assess learning, such as homework, labs, test scores, and final project evaluation. Qualitatively, as our approach is highly student-centric and experiential, we will use a measure of motivation to evaluate our approach. From [37], the intrinsic and extrinsic goal orientation and the task value subscales seem most appropriate. From [15], the interest/enjoyment, perceived competence, effort/importance, and value/usefulness subscales are of interest.

As course development will be staggered, we can conduct pre and post versions of these surveys for the existing courses and the modified courses to evaluate the differential impact of the new approach.

As there is a strong longitudinal component to both **integration over time** and **integration over subject matter**, we will develop a longitudinal study that tracks students from course to course to both correlate their motivational survey results as well as introduce new instruments that can evaluate student perception of the benefits of time and subject integration. These instruments will attempt to assess the longitudinal impact of the platform qualitatively. They will likely take the form of both online surveys, in-class discussions, and written reflections. It is important to not only assess the opinions of the students, but of the instructors as well. Development of these instruments will be ongoing for the duration of the proposed work.

Finally, an interesting aspect is to understand the impact of the platform approach in different contexts. By using the same instruments, pre and post course development, we will have the opportunity to evaluate the effect of the different institutions.

In order to gauge the utility of the platform in enabling **location independence**, we will instrument the platform hardware and the development software used by the students to provide anonymized usage statistics. This information will initially include platform runtime, individual hardware utilization, software modules used, and device drivers used.

6 Preliminary Work

Next, we summarize our previous experiences with PAL-based educational initiatives both at Northeastern and Olin, as well as with other institutions.

6.1 Initial Experiences with Personal Active Learning Platforms

In 2006 an exploratory engineering education project was initiated, by PI Ravel and Mark McDermott at the University of Texas-Austin, in collaboration with several high technology firms (Analog Devices, Xilinx,

Freescale Semiconductor, ARM Holdings, LeCroy), a local social enterprise (The Learning Labs), and with early input from a number of ECE educators from diverse institutions [39]. The project objective was to see if an open "ECE-systems" design platform could be developed with enough generality to support a broad range of experiential learning exercises for the various course domains in a typical ECE curriculum. Input needs and best practices were gathered from visits with leading ECE faculty in the US, Europe, and Asia to help define a system that would be globally relevant. The faculty input covered a cross section of the ECE curriculum that would be likely to benefit from a design-centric active learning pedagogy. A guiding premise was that the platform be usable for both studio and lab exercises along with enabling projects having relevancy to realistic ECE products and systems.

In observing faculty that had built their own custom platforms to cover one or several of their courses, a number of general attributes emerged for an ECE cross-curriculum platform:

- Availability of both open source and commercial engineering software tools
- Modular hardware design with user expandability
- Desktop size and ideally a single physical unit to allow redeployment across labs
- Heterogeneous technology elements to insure broad course coverage
- Industry standard technology to enable student exposure to real-life components and practices
- Cost less than or equal to a typical ECE lab bench setups

With these objectives as guides, an initial prototype was developed in 2007 that could be used to investigate the utility in a course or project lab setting [39]. The initial PAL system was a highly modular architecture consisting of multiple technology options for each of the basic system elements of communications, computing, and real-world interfaces.

Since 2007 the Gen1 system has been piloted at multiple engineering education institutions in the US, Europe and Asia. The organizations have spanned a range of sizes from large research universities (U. of Texas-Austin, U. of Wisconsin-Madison, IIT-Madras, IISc) to medium (U. of Novi Sad) to small (IIIT-Bangalore, Olin College of Engineering), and the pilots ranged from 2nd year undergraduate to 2nd year graduate level courses [38, 31].

The following is a summary of lessons learned from Gen1 pilots:

- It is possible to impact a broad range of courses
- Common SW tools inspired spontaneous team work sessions - mostly outside the lab
- A small set of elements provided the most value
- Heterogeneous technologies enable partitioning and tradeoff exercises fostering system thinking skills
- Student engagement is not as high in a lab-centric model - there were many requests for personal units

We used the lessons from this earlier experience to guide the architecture of our PAL system, the Gen2 System. We now have a good basis for judging what works and what does not work across a broad range of ECE courses and programs. The Gen1 experience targeted individual courses at the five institutions, so the focus was not pursuing horizontal and vertical integration. Further, there was little attention paid to studying how students learn outside of the classroom/laboratory. The proposed project is the next logical step in a much broader deployment of our Personal Active Learning approach

6.2 Microprocessor-based Design at Northeastern

The Microprocessor-based Design course was based for the past years on a commercial platform, BF561-EZKIT as introduced by PI Kaeli [5]. For Spring 2010, PI Schirner upgraded the course to use the PAL system and we have gathered valuable experience from this course.

In Spring 2010, 12 students (mostly juniors) participated in the course. The PAL system was used in the lab only and not yet available as a student-owned system. Two 100 minute lecture blocks were accompanied by 120 minutes of lab time per week. In five two week lab exercises students explored the principles of embedded software development and understanding of the embedded architecture. In addition, the lab contained an open-ended project of the students choosing allowing them to combine and utilize the course knowledge. Students specified and designed their projects in parallel to the regular lab sessions and the projects implementation phase was limited to the last 4 weeks of the course. It is the first time that the course actually facilitates an open project, and students have enthusiastically acted upon this opportunity. Project topics included a handheld signal analyzer and a compressed-audio wireless link (shown in Fig. 5).

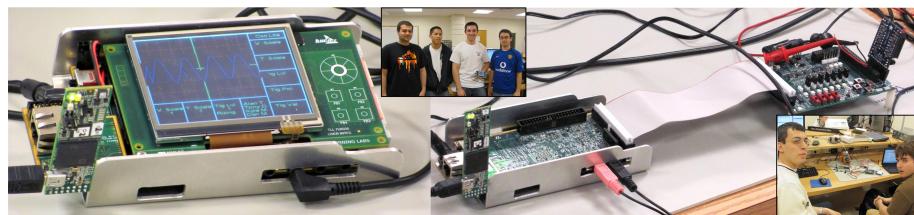


Figure 5: Signal analyzer and compressed-audio wireless projects in 2010.

A large portion of the course success can be attributed to using the PAL system. Students reacted very positively and motivated toward using the PAL system. They were intrigued by its power and versatility, which also triggered students' creativity. All groups chose a products for their project topic and naturally utilized the wide range of interfaces and sensors on board. For example, one team replicated the classical marble maze using the integrated accelerometer for detecting the board's position. It is interesting to note that all groups have expanded on their own beyond the course taught concepts: two groups used the LCD interface although graphics drivers were not discussed (or provided) in class; the other two groups have expanded PAL for wireless communication, and RFI interfacing, respectively. Given the self identified product motivation students naturally opted for self-learning to reach their goals. Students were quite ambitious with their project ideas, and highly motivated follow them through. This motivation sparked through a product like project design is significant potential we will capitalize in future classes.

The lab projects also had a valuable integrative component, combining knowledge from different domains. The heterogeneity and versatility of PAL allowed students to automatically link forward to new course topics to bridging different domains like: audio coding, networking, user interface design and circuits. While it is not feasible to cover these subjects all in one course, it helps for students to discover their domain specific interests.

Naturally this freedom leads to a student-centered integration of the different subject matters and makes them much more attentive for future specialized classes.

Even though the current lab exercises are strongly geared toward active learning, instead of using pre-canned examples, students get a coarse framework, have to actively research material to find and discover the solution on their own. While this is very desirable and promotes life-long learning, it also increases the students' time investment. Since the past class had the PAL system only available during lab time, many of the discoveries were delayed the scheduled lab time itself. A student-owned platform could eliminate this time pressure. When surveyed, 80% of the students have indicated that an own PAL would be very beneficial to their academic progress. They gave concrete examples of "developing at home", "some debugging ahead of time", "playing at home". Only one student was indifferent and cited time limitations as a restriction.

In addition, we plan to use the student-owned platform for more studio-based class settings in which we can offer a short cycle between theory and practice.

7 Plan of Work

Table 1 maps the planned activities that will be carried out over the 2-year period of this project. Preparatory activities start in Fall 2010 with developing the assessment material at Olin College for the surveys and the PAL instrumentation. Simultaneously at Northeastern, we will establish the web presence and repositories for sharing course material.

Table 1: Timeline of planned work.

Location	NEU			Olin		
	Course	Computer Architecture	Microprocessor-based Design		Principles of Engineering	Digital Systems
Fall 10	Develop web presence		Update	Develop assesment	Exp. PAL	Udate
Spring 11		Teach current	Teach PAL (a)		Exp. PAL	Exp. Pal
Summer 11		Update	Update		Update	Update
Fall 11		Teach PAL			Teach PAL	
Spring 12			Teach PAL (b)		Teach PAL	Teach PAL
Summer 12	Disseminate			Disseminate		

Northeastern will lead the upgrade of Microprocessor-based Design using the student-owned PAL in Spring'11. We will draw upon our successful trials during Spring'10. The data obtained through our assessment during the Spring'11 deployment will significantly drive the course development in Summer'11 across all courses (both at NEU and Olin). We will then roll out additional PAL-enabled courses using the student-owned PAL, while incorporating earlier feedback. Each of the four courses that are studied in this project will be run using PAL in at least one instance. We will then disseminate our findings in ASEE conferences and relevant journals, and prepare course materials for further dissemination to other universities.

7.1 Dissemination

The proposed project is performed jointly between Northeastern University and the Olin College. The institutions offer educational heterogeneity, which will help us in developing university transferable course material and lab modules. Since easy access to the developed material is paramount, we will create a community web site for dissemination and sharing that incorporates the following: (a) a Wiki for sharing instructions and resources, (b) a repository for distributed development of software and hardware modules, (c) a repository for sharing lab exercises and course modules, (d) a portfolio showcasing student projects, (e) several discussion forums capturing questions and solutions. With openly sharing our course material anticipate to stimulate an open dialog with other educators. We are looking forward to their input guiding us in the development of modules suitable for the larger community.

During the two years as outlined in this proposal, we will continue to work closely other institutions both for feedback and to positively stimulate other undergraduate curricula. PIs are in contact with other regionally local universities, which also expressed interest in the presented approach: Tufts University, Boston University, and Massachusetts Institute of Technology. The PIs will share the results (experiences and assessments) with the engineering education community through ASEE conferences, and contributions to relevant journals. After establishing initial results as per this proposal, PIs will extend the scope to include K-12 institutions.

8 Qualifications of the PIs

PI Gunar Schirner actively shapes the undergraduate curriculum by contributing to the Undergraduate Studies Committee. He is also the faculty co-advisor for the IEEE student chapter, organizing extra curricular learning opportunities. During his Ph.D, Gunar Schirner was awarded a Pedagogical Fellowship at UC Irvine honoring his education capabilities. Gunar Schirner's commitment to education also shows in an earlier engagement – before starting the Ph.D. he developed and taught over 30 courses in informatics related topics over a time period of 5 years. His research interest is the design of embedded systems with a focus on design automation. He has an active publication record and co-authored the book "Embedded System Design" [20] which is adopted for graduate education at different institutions. Gunar Schirner has 5 years industry experience designing distributed real-time software for telecommunication systems.

PI David Kaeli has a sustained record pursuing a number of different educational initiatives. He presently serves as the Chair of the ECE Undergraduate Study Committee at Northeastern, a position he has held for the past 5 years. In 1995, Kaeli created the first workshop on Computer Architecture Education, which has been held annually with major Computer Architecture conferences for the past 15 years. He has served as an NSF REU mentor for the past 7 years, and in 2010 received an REU site award from the EEC Division. He has been a PI/co-PI on 13 previous NSF awards, including an NSF CAREER Award. At Northeastern, he has received both teaching awards and student mentoring awards. He has 12 years of industrial experience at IBM and was recently awarded an IEEE Fellow.

PI Mark L. Chang is an Assistant Professor of Electrical and Computer Engineering at Olin College. As one of the first cohort of faculty at Olin, Mark is deeply committed to engineering education and has been heavily involved in developing the ECE curriculum. He has directly developed numerous courses, including Computer Architecture, Embedded Systems, Mixed Analog-Digital VLSI, and Mobile Application Development. In several of these project-based courses, Mark has led the effort to integrate reconfigurable computing into the student experience. Mark is also involved in leading several teams of students each year in Olin's year-long senior capstone program, which provides a culminating engineering design experience through industry-funded projects. Mark has published and presented numerous papers related to both reconfigurable computing and undergraduate engineering education.

Mark Somerville, Co-Pi, Associate Professor of Electrical Engineering and Physics and Associate Dean of Academic Programs and Curricular Innovation (Ph.D., MIT, 1998) has been heavily involved in curriculum design and development since joining the faculty of Olin College in 2001. He led the committee that developed the inaugural curriculum at Olin, and also led the college's first strategic planning effort. Dr. Somerville has extensive experience consulting with faculty at other institutions around the globe regarding curricular change; he recently spent a sabbatical year at Delft University, where he played a leading role in facilitating the complete redesign of Delft's undergraduate aerospace engineering program. Dr. Somerville has published and presented numerous papers and talks related to student-directed learning, curricular integration, project-based learning, and curricular change.

Mihir K. Ravel is a Distinguished Research Scientist in Residence at the F. W. Olin College of Engineering. He is a technology leader in the electronics design and automation industries with expertise in developing high performance system architectures and platforms for science/engineering tools. He was a Tektronix Fellow and head of Strategic Technologies, and later Vice-President of Technology at National Instruments. He has

led many collaborations with universities, and now focuses on applying his experience to fostering design-centric active learning. During 2004-2007 he met with STEM education leaders in emerging economies, and was an invited faculty in India at IIIT-Bangalore in 2007-8. He continues as an advisor to IIIT-B along with IIT-Madras, IIIT-D&M, and IISC. He has been an innovation advisor to global leaders including the U.S. State Dept., the U.S. Engineering Deans Institute, the National Learning Strategies Conference, the Indian National Research Development Corporation, and the Indo-US Collaboration on Engineering Education.

9 Prior NSF Support

Gunar Schirner has not received any prior funding from NSF.

David Kaeli is currently involved in a project titled: *Archer - Seeding a Community-based Computing Infrastructure for Computer Architecture Research and Education*, grant number CRI-CNS-0750868. This project builds a virtualized set of physically distributed computing resources for use by the Computer Architecture Research Community [18]. The project leverages virtualization and the Condor job submission system. This is an infrastructure project, though serves the research needs of many, including our own work in hardware/software vulnerability [50, 51, 52], and modeling of virtualized systems [2].

Kaeli has a newly starting project titled "A Biomedical Imaging Acceleration Testbed," EIC-0946463, which will developed a software-engineered set of GPU libraries for the biomedical imaging community. More information is available at <http://www.ece.neu.edu/groups/nucar/gpucomputing.html>.

Kaeli is the PI on a third project titled *Tuning Libraries to Effectively Exploit Hierarchical Memory Systems* grant number CCF-0342555 . This project that ended in 2008 and was a collaboration with co-PIs Misha Kilmer (from Tufts University) and Gene Cooperman (from Northeastern University). The project studied how sparse matrices impact I/O performance on clusters. More information on this project is available at the TunLib project website [54]. We have published the following papers [11, 29, 28, 57].

Mark Chang has not received any prior funding from NSF.

Mark Somerville lead as a PI NSF DUE #0231231: "Centerpiece Projects: Developing Large-scale, Interdisciplinary Design Experiences for Freshman Engineering" (\$74,921, 07/01/03 - 06/30/05). This grant supported the development of multiple large-scale "centerpiece" projects that integrate mathematics, physics, and engineering concepts for use in a freshman curriculum. The grant led, also to the development of an intellectual framework based in design learning research for self-directed projects early within engineering curricula. Publications include: [46],[47],[61],[48],[49].

Mihir Ravel has not received any prior funding from NSF.

10 Coordination between Institutions

The proposed project is jointly performed by two regionally close institutions. The PIs will hold a monthly meeting to coordinate course development and align the educational strategy. Keeping a close interaction between the institutions will ensure more readily transferable results when later expanding to other institutions. In addition, the PIs will use these monthly meetings to share results and experiences of ongoing classes.

Developed course material, and the supporting hardware and software modules will be stored in a centralized repository at Northeastern University. Therefore PIs can draw upon the earlier experience during new course design and give further feedback to foster a general approach. Once course material is releasable, it will enter the dissemination stage. Then, it will be made publicly available through the web community site already described in Section 7.1.

E. References Cited

- [1] S. Aleksandar and D. Maconachie. Strategic curriculum design: An engineering case study. *European Journal of Engineering Education*, pages 19–33, 1997.
- [2] Emmanuel Arzuaga and David Kaeli. Quantifying load imbalance on virtualized enterprise services. In *ACM Conference on Performance Engineering*, pages 235–242, Jan 2010.
- [3] Peter Athanas and Cameron Patterson. A holistic approach towards a unified cpe laboratory platform. In *International Conference on Microelectronic Systems Education (MSE)*, page 2. IEEE, 2007.
- [4] Miri Barak. Studio-based learning via wireless notebooks: A case of a java programming course. *International Journal of Mobile Learning and Organization*, 1, 2007.
- [5] Michael Benjamin, David Kaeli, and Richard Platcow. Experiences with the blackfin architecture in an embedded systems lab. In *WCAE '06: Proceedings of the 2006 Workshop on Computer Architecture Education*, page 2. ACM, 2006.
- [6] Joseph Borgogna, Eli Fromm, and Edward Ernst. Engineering education: Innovation through integration. *Journal of Engineering Education*, pages 3–8, January 1993.
- [7] P. Candy. Self-direction for lifelong learning: A comprehensive guide to theory and practice. *Jossey-Bass*, 1991.
- [8] L. Carloni, F. De Bernardinis, C. Pinello, A. L. Sangiovanni-Vincentelli, , and M. Sgroi. Platform-based design for embedded systems. In R. Zurawski and Boca Raton, editors, *The Embedded Systems Handbook*. CRC Press, 2005.
- [9] M. Clark, J. Froyd, P. Merton, and J. Richardson. The evolution of curricular change models within the foundation coalition. *Journal of Engineering Education*, pages 37–47, 2004.
- [10] ABET Engineering Accreditation Commission. Criteria for accrediting engineering programs, 2005.
- [11] G. Cooperman, X. Ma, and V.H. Nguyen. Static Performance Evaluation for Memory-Bound Computing: The MBRAM Model. In *Proceedings of the 2004 International Conference on Parallel and Distribute Processing Techniques and Applications*, 2004.
- [12] ASEE Deans Council and Corporate Roundtable. *The Green Report: Engineering Education for a Changing World*. ASEE, Washington, D.C, 1994.
- [13] A. J. Cropley. *Lifelong Education: A Stocktaking*. Pergamon Press/Unesco Institute for Education: Oxford/Hamburg, 1979.
- [14] E. L. Deci, R. J. Vallerand, L. G. Pelletier, and R. M. Ryan. Motivation and education: The self-determination perspective. *Educ. Psychol.*, 3-4:325–346, 1991.
- [15] Edward Deci and Richard Ryan. Intrinsic motivation inventory. <http://goo.gl/Vc5F>.
- [16] Jan Edler and Mark Hill. Dinero iv trace-driven uniprocessor cache simulator, 2000. <http://www.cs.wisc.edu/~markhill/DineroIV>.
- [17] D.H. Evensen and C.E. Hmelo, editors. *Problem? Based Learning: A Research Perspective on Learning Interactions*. Mahwah, NJ: Erlbaum, 2000.

- [18] Renato J. O. Figueiredo, P. Oscar Boykin, José A. B. Fortes, Tao Li, Jie-Kwon Peir, David Wolinsky, Lizy K. John, David R. Kaeli, David J. Lilja, Sally A. McKee, Gokhan Memik, Alain Roy, and Gary S. Tyson. Archer: A community distributed computing infrastructure for computer architecture research and education. In *CollaborateCom*, pages 70–84, 2008.
- [19] Jeffrey Froyd and Matthew Ohland. Integrated engineering curricula. *Journal of Engineering Education*, pages 147–164, January 2005.
- [20] Daniel D. Gajski, Samar Abdi, Andreas Gerstlauer, and Gunar Schirner. *Embedded System Design: Modeling, Synthesis and Verification*. Springer, 2009.
- [21] T. Garcia and P. R. Pintrich. The effects of autonomy on motivation and performance in the college classroom. *Contemp. Educ. Psychol.*, 4:477–486, 1996.
- [22] S.R. Hall, I. Waitz, D.R. Brodeur, D.H. Soderholm, and R. Nasr. Adoption of active learning in a lecture-based engineering class. *32nd Annual Frontiers in Education*, pages T2A9–15, 2002.
- [23] John Hennessey and David Patterson. *Computer Organization and Design*. Morgan Kaufmann, 2008.
- [24] Christopher D. Hundhausen, N. Hari Narayanan, and Martha E. Crosby. Exploring studio-based instructional models for computing education. In *Proceedings of ACM Technical Symposium on Computer Science Education*, March 2008.
- [25] S. Jiusto and D. DiBiasio. Experiential learning environments: Do they prepare our students to be self-directed, life-long learners? *Journal of Engineering Education*, pages 195–204, 2006.
- [26] Anette Kolmos. Premises for changing to pbl. *International Journal for the Scholarship of Teaching and Learning*, pages 1–7, January 2010.
- [27] James Larus. Spim s20: A mips r200 simulator, 1990. Technical Report TR966, University of Wisconsin - Madison.
- [28] X. Ma, J. Ansel, and G. Cooperman. Adaptive checkpointing for master-worker style parallelism. In *Proceedings of the 2005 Computer Society International Conference on Cluster Computing*, 2005.
- [29] X. Ma and G. Cooperman. Fast query processing by distributing an index over cpu caches. In *Proceedings of the 2005 Computer Society International Conference on Cluster Computing*, 2005.
- [30] Eric Mazur. Farewell, lecture? *Science*, pages 50–51, 2009.
- [31] Mark McDermott, Jacob Abraham, and Mihir Ravel. Balancing virtual and physical prototyping across a multi-course vlsi/embedded-systems/soc design curriculum. In *Proceedings of American Society for Engineering Education Annual Conference*, 2009.
- [32] M. H. Meyer and A. P. Lehnerd. *The Power of Product Platforms*. 1997, Simon & Schuster.
- [33] Don Lewis Millard, Mohamed Chouikha, and Frederick Berry. Improving student intuition via rensselaer's new mobile studio pedagogy. In *Proceedings of American Society for Engineering Education Annual Conference*, June 2007.
- [34] National Academy of Engineering. *Educating the Engineer of 2020: Adapting Engineering Education to the New Century*. Washington, D.C.: The National Academies Press, 2005.
- [35] Yale Patt and Sanjay Patel. *Introduction to Computing Systems: From Bit and Gates to C and Beyond*. McGraw Hill, 2003.

- [36] I. C. Peden, E.W. Ernst, and J.W. Prados. *Systemic Engineering Education Reform: An Action Agenda*. NSF, Washington, D.C., 1995.
- [37] Paul R. Pintrich. *A Manual for the Use of the Motivated Strategies for Learning Questionnaire (MSLQ)*. National Center for Research to Improve Postsecondary Teaching and Learning, 1991.
- [38] Mihir Ravel, Mark L. Chang, Mark McDermott, Michael Morrow, Nikola Teslic, Mihajlo Katona, and Jyotsna Bapat. A cross-curriculum open design platform approach to electronic and computing systems education. In *International Conference on Microelectronic Systems Education (MSE)*, pages 69–72, 2009.
- [39] Mihir Ravel and Mark McDermott. An electronic system design platform for systematic learning in ece and ict curriculum. In *International Conference on Microelectronic Systems Education (MSE)*, pages 145–146, 2007.
- [40] R. M. Ryan and E. L. Deci. Self-determination theory and the facilitation of intrinsic motivation. *Social Development and Well-Being, Am. Psychol.*, 1:68–78, 2000.
- [41] Kaushal Shanghai, David Kaeli, Alex Raikman, and Ken Butler. A code layout framework for embedded processors with a configurable memory hierarchy. In *Proceedings of the 5th Workshp on Optimizations for DSP and Embedded Systems*, 2007.
- [42] A. Sangiovanni-Vincentelli, L. Carloni, F. De Bernardinis, and M. Sgroi. Benefits and challenges for platform-based design. In *Proceedings of the Design Automation Conference (DAC)*, 2004.
- [43] A. L. Sangiovanni-Vincentelli. Defining platform-based. www.eedesign.com/story/OEG20020204S0062, 2002.
- [44] Matthew Shuman, Donald Heer, and Terri S. Fiez. A manipulative rich approach to first year electrical engineering education. In *Proceedings of the 2008 Frontiers in Education Conference*, 2008.
- [45] Karl Smith, Sheri Sheppard, David Johnson, and Roger Johnson. Pedagogies of engagement:classroom-based practices. *Journal of Engineering Education*, pages 87–101, January 2005.
- [46] M. Somerville and J. Geddes. Along the spectrum of inquiry: A project-based approach to the first year experience. *Proceedings of the 2005 Frontiers in Education*, page 116, 2005.
- [47] M. Somerville and J. Geddes. *Research and Practice of Active Learning In Engineering Education*, chapter Early Exploration: A Project Based Approach. Amsterdam University Press, 2005.
- [48] M. Somerville and A. Horton. Centerpiece projects. *Active Learning in Engineering Education*, 2004.
- [49] M. Somerville, S. Spence, J. Stolk, and Y. Zastavker. Olin college: Its alive! (invited special session). *Frontiers in Education*, 2005.
- [50] Vilas Sridharan and David R. Kaeli. Quantifying software vulnerability. In *Workshop on Radiation Effects and Fault Tolerance in Nanometer Technologies (WREFT-1)*, 2008.
- [51] Vilas Sridharan and David R. Kaeli. The effect of input data on program vulnerability. In *Workshop on System Effects of Logic Soft Errors (SELSE-5)*, 2009.
- [52] Vilas Sridharan and David R. Kaeli. Eliminating microarchitectural dependency from architectural vulnerability. In *International Symposium on High Performance Computer Architecture (HP CA-15)*, 2009.
- [53] Roger L. Traylor, Donald Heer, and Terri S. Fiez. Using an integrated platform for learning to reinvent engineering education. *IEEE Transactions on Education*, 46, November 2003.

- [54] TunLib Project. <http://www.ece.neu.edu/groups/nucar/research/TunLib/>.
- [55] L. Vanasupa, J. Stolk, and T. Harding. Application of self-determination and self-regulation theories to classroom design: Encouraging signs of lifelong learning competencies. *International Journal of Engineering Education*, in press.
- [56] M.R. Varanasi, O.N. Garcia, P. Guturu, Hai Deng, Xinrong Li, and Shengli Fu. Work in progress: An innovative electrical engineering program integrating project-oriented and lifelong learning pedagogies. *36th ASEE/IEEE Frontiers in Education Conference*, pages S1H-7, 2006.
- [57] Y. Wang and D. Kaeli. Load balancing using grid-based peer-to-peer parallel i/o. In *Proceedings of the 2005 IEEE Cluster Computing Conference*, 2005.
- [58] Steven C. Wheelwright and Kim B. Clark. *Revolutionizing product development: quantum leaps in speed, efficiency, and Quality*. 1992, Simon & Schuster.
- [59] J. Wilson. *Technology Enhanced Learning: Opportunities for Change*, chapter The Development of the Studio Classroom. Lawrence Erlbaum Associates, 2001.
- [60] W. Yurcik, G.S. Wolffe, and M.A. Holiday. A survey of simulators used in computer organization/architecture courses. In *Proceedings of the Summer Computer Simulation Conference*, 2001.
- [61] Y. Zastavker, J. Crisman, M. Jeunette, and B.S. Tilley. Kinetic sculptures: A centerpiece project integrated with mathematics and physics. *International Journal of Engineering Education*, pages 1031–1042, 2006.
- [62] Y. Zastavker, M. Ong, and L. Page. Women in engineering: Exploring the effects of project-based learning in a first-year undergraduate engineering program. In *Proceedings of the 36th ASEE/IEEE Frontiers in Education Conference*, 2006.

**Olin College Innovation Fund Proposal
Network Hacking, Mark L. Chang**

Summary

Over the course of the summer, I propose to develop a course addressing one of the Grand Challenge topics, cyber security. I will work with Franklin Turbak, Associate Professor of Computer Science at Wellesley College to develop a complete set of course materials. The course, entitled “Network Hacking”, aims to provide a unique opportunity to teach computer-science and computer-engineering principles through a failure-driven model. We will promote the use of a hacker mindset to strengthen student understanding of core computing abstractions. Instead of focusing on the way things *should* work, we will concentrate on how systems can be *made to fail*. In doing so, we will acquire the skills to question assumptions and models of trust in engineering systems [Bratbus09].

Motivation

As the computing landscape grows increasingly mobile, networked, and ubiquitous, sensitive information begins to distribute itself among devices, services, and storage mediums. From the perspective of the typical end user, the formerly disconnected and complex silo’ed world has recently become more accessible, convenient, and productive. Unfortunately, our growing reliance on this ubiquity of computing has made exposure of private and sensitive information more commonplace. Identity theft, credit card fraud, and other countless private bits are ending up in the wrong hands. The fact is that not enough of the engineering community creating these innovations are focused on security from the outset. In fact, one of the greatest technological revolutions, the Internet, was designed almost completely devoid of a security model.

The NAE has called out this need to “secure cyberspace” as one of the fourteen Grand Challenges for the 21st century. Unfortunately, there is frighteningly little curriculum in U.S. higher education dedicated to creating the next generation of security-minded engineers. Even as recently as March 24, 2010, Richard Marshall, the director of global cyber security management at the Department of Homeland Security remarked, “...we are going to fail if we don’t invest more money, time, attention and rewards to educate the workforce” [Corbin09]. In most CS curriculum, there is but one course, “Network Security”, that even begins to address some of these concerns.

The challenge at Olin is compounded by the lack of a CS major and the accompanying small-footprint CS curriculum. While it would be easy to modify and import MIT’s 6.857, Computer and Network Security, courses like MIT’s are often situated in a CS curriculum and are connected to a set of courses and expectations that Olin cannot mimic. Therefore, we need a dedicated solution.

Project Proposal

The proposed “Network Hacking” course will be derived from the existing CS342: Computer Security course taught by Franklin Turbak at Wellesley College, and the immersive two-week SISMAT (Secure Information Systems Mentoring and Training) program offered at Dartmouth College.

Collaborating with Prof.Turbak will lend significant expertise and a wealth of existing resources in the area of network security to the project, not to mention an existing curricular framework. I aim to contribute to Prof.Turbak through working to help improve his course by developing detailed hands-on lab experiences and an approach that focuses more on project-based learning.

The SISMAT program offers a smaller footprint starting point that focuses very much on highly motivating, hands-on lab experiences. I believe that deriving from a “boot camp” approach will be easier to integrate into the Olin curriculum. In SISMAT, students are encouraged to both perform exploits on real systems, as well as reconstruct the timeline and events of a successful attack. But perhaps more importantly, the program was designed explicitly to utilize the hacker mindset to encourage students to approach systems as an attacker would: questioning assumptions of trust models, examining across layered abstractions, probing corner cases, and moving away from “how it should work” to “how to make it fail”. These essential learning objectives fit extremely well with the Olin curricular models of scientific inquiry. They are also the spiritual complement to many of the entrepreneurial mantras our students value in the software realm, such as agile development, rapid iteration, customer-driven design, and a quick (or even prototype) to market. With Olin’s focus on entrepreneurship, I believe this alternative hacker mindset will fill a large hole in the technical approaches our students utilize when implementing new software and services.

A key part of the proposed work will be to involve at least one student either in the summer or the Fall of 2010. This student will be responsible for aiding the development of lab exercises. I anticipate this to be the single most challenging aspect of this project, and having additional personnel would be of great use. The budget needs during the summer would be much greater than during the Fall, when research credit could be offered in lieu of salary.

Important travel items include a trip to Dartmouth to observe the SISMAT course in late June 2010, and a trip to Defcon, the premier hacker conference, in late July 2010.

Outcomes and Evaluation

We will develop a complete set of course materials for the Network Hacking course. This includes at least identifying relevant texts and resources for students, developing exercises and ideas for labs, preparing lectures, prototyping a number of final project ideas, and creating a list of local security experts that would be able to serve as guest speakers. A large effort will be in setting up a secure computing environment, or “sandbox”, for the students to practice attack vectors and vetting our system with Olin’s IT department.

We will pilot the course as a half-semester course at Olin and as a revamped CS342 course at Wellesley. This will give us two opportunities for refinement before offering it as a Special Topics course to the Olin community. The evaluation process will be through student and co-instructor (if any) feedback, and hopefully feedback from SISMAT faculty at Dartmouth and George Mason as they come observe our classrooms and learn about our modified curriculum.

Timeline

June 2010: Resource acquisition and learning the field of network security.

June 25, 2010: Initial visit to Dartmouth to observe SISMAT students.

July 2010: Developing initial labs, sandbox, and lectures.

July 29, 2010: Visit Defcon 18.

August 2010: Prepare for initial courses at Wellesley and Olin.

Fall 2010: Work with student to refine labs (if no student during summer)

Fall 2010: CS342 is offered at Wellesley

Spring 2010: Offer half-course, Network Hacking

Grand Challenges and Interdisciplinary Elements

The topic of the new course is directly addressing a Grand Challenge. The project-based nature of the proposed course would fulfill the Grand Challenge Scholars Project component. The course is not interdisciplinary in the strictest sense. Rather, it is an intercollegiate collaboration, and one that hopefully crosses engineering and computer science boundaries.

Budget and Budget Justification

Summer salary for one month

This is to support Mark to work for the months of June and July, the estimated time necessary to complete the project. Estimates can be provided by Terri Dunphy.

\$600, Travel to Dartmouth College, Hanover, NH

To visit and observe the faculty and students during the middle of the SISMAT experience. June 25-27, 2010.

\$1000, Travel to Defcon 18 conference, Las Vegas, NV

To learn from the best in the business at the premier hacker conference. Many of the most notorious vulnerabilities have been announced at this conference, complete with real-time exploits. July 29, 2010 - August 1, 2010.

\$4000 Summer 2010 researcher student or

\$1400 Fall 2010 paid research student (12 hours/week for 12 weeks) or

\$0 Fall 2010 unpaid research students

The student will be responsible for aiding in the development of lab exercises.

References

[Bratus09] Bratus, S., Shubina, A., and Locasto, M. E. 2010. Teaching the principles of the hacker curriculum to undergraduates. In Proceedings of the 41st ACM Technical Symposium on Computer Science Education (Milwaukee, Wisconsin, USA, March 10 - 13, 2010). SIGCSE '10. ACM, New York, NY, 122-126.

[Corbin09] Corbin, K., "U.S. Faces Cyber Security Gap Without Training, Education", <http://bit.ly/dmtfc4>

[Locasto09] Locasto, M.E. and Sinclair, S., "An Experience Report on Cyber-Security Education and Outreach", In Proceedings of the Annual Conference on Education in Information Security, 2009.

ADAPTIVE ROUGH TERRAIN NAVIGATION ON A LEGGED ROBOT PLATFORM

ELENA OLEYNIKOVA, STANISLAW ANTOL
F. W. OLIN COLLEGE OF ENGINEERING, NEEDHAM, MA 02492

PROJECT DESCRIPTION

We would like to research methods of rough terrain navigation on a legged robot platform. As an end goal, we would like to provide a robust method of traversing terrain on a legged robot, including high-level path planning and low-level footstep planning.

GOALS

One goal of our research is to explore long-term path planning algorithms specifically suited for rough terrain. Many path planning algorithms divide the environment into two categories: passable and impassable. However, rough terrain poses a much more complicated problem, where parts of the terrain may be passable given a certain path (for example, climbing a path around a hill as opposed to going straight over), and impassable using others. We would like to build upon existing path planning algorithms to use awareness of the terrain and the robot's capabilities to plan an optimal path over terrain.

Another goal of our research is to learn more about footstep planning algorithms and implement our own as part of a larger rough terrain navigation system. Footstep planning is often explored in the context of biped robots in indoor environments. For example, Ozawa *et al.* use visual odometry for footstep planning on a biped robot [1]. Chestnutt *et al.* explore a more generic footstep planning algorithm that is tested on both a biped and a quadruped robot [2]. We would like to build on these methods and integrate them into a larger, long-term path planning algorithm to have a complete system for rough terrain navigation on a legged robot.

Our final goal is to do original research that is publishable at a conference or in a journal. This would help greatly with graduate school admissions and give us important experience in writing and publishing in the computer science and robotics fields.

METHODS

In order to develop our own rough-terrain navigation algorithm, we plan to do an extensive literature search on existing methods (some of which are cited in the References section) for path planning and footstep planning. We will then implement these methods on our platform to test their viability and find the advantages and disadvantages of using each method in the testing environment. We will then use our simulation (described below) and lab and field testing to make improvements to the algorithms and hopefully create something novel and robust.

Additionally, in order to make the robot be able to assess whether or not the robot is able to climb over uneven terrain, our group wants to write a simulation that will consist of the robot and the imported environment from the laser range finder data. The simulation will start off as simple as possible (e.g. having a simple robot model, gravity, and friction). Depending on its success, we can further develop it, as needed. After we have a working simulation environment, we can run different possible foot movements to navigate over the terrain. The simulation will guide the robot in deciding on a proper sequence of steps that are within our robot's capacity and will also keep the robot balanced, even in unusual positions. We can also try to define a metric, based on factors such as the terrain and how many footsteps the robot would need to take, which can be used to classify how difficult different areas would be to climb over. We will then incorporate this into our overall path planning algorithm.

RESEARCH PLATFORM

In the 2009-2010 academic year, a team of four (Elena Oleynikova, Stanislaw Antol, Ann Wu, and Alex Trazkovich) built a research platform and a testing environment, shown in Figure 1. The research platform itself is a hexapod robot, chosen for its superior mobility over rough terrain and lack of turning radius restrictions that many wheeled robots possess. The robot has on-board step sequencing and accepts wireless commands indicating the direction and size of the steps to be taken.

The testing setup is a sandbox with gravel, sand, and large rocks to simulate real rough terrain in an indoor environment. There is a stereo camera overhead (not shown) that overlooks the sandbox, that acts as both a GPS (detecting the hexapod in each frame) and as a 3D terrain scanner to provide an almost complete map of the robot's environment. Though this setup is ideal for developing algorithms that assume a known terrain, we would like to transition the sensing to be on board the hexapod to develop algorithms that would work in unknown terrain. For this purpose, we would like to purchase a Hukuyo Laser Range Finder, discussed in the budget section.



Figure 1. A photograph showing the existing setup, including a sandbox with gravel, rocks, and sand to use as an indoors rough terrain environment, and the hexapod robot. There is also an overhead stereo camera, not shown, that currently functions as both a GPS, by detecting where the hexapod is in a given image, and as a 3D terrain scanner.

BUDGET

Though most of the platform is built and functional, we would like to purchase a Hokuyo URG-04LX-UG01 Laser Range Finder (<http://www.acroname.com/robotics/parts/R325-URG-04LX-UG01.html>) for \$1200 to use as a primary sensor.

PROJECT TIMELINE

We have created a timeline with goals for each month. We will make a week by week timeline with more specific tasks at the beginning of each month.

Month	Goals
September	Purchase and mount Hokuyo LIDAR. Set up the testing environment - rocks, slopes, gravel, sand, etc. Mount other sensors (foot pressure sensors, other range finders). Implement an existing terrain navigation algorithm on-board. Analyze advantages/disadvantages of existing algorithms. Do literature search on existing terrain navigation methods.
October	Develop simulation. Come up with ideas to improve existing algorithms to better suit rough terrain.

	Begin implementing improvements in simulation. Continue literature search on terrain navigation.
November	Implement improved terrain navigation on physical platform. Continue developing algorithm.
December	Begin writing semester report. Continue developing algorithm.
January	Finish semester report. Continue testing the path-planning algorithms in lab environment.
February	Begin developing footstep-planning algorithms. Continue developing footstep-planning algorithms.
March	Do field-tests in unknown terrain (not lab terrain). Finish development.
April	Choose possible journals and conferences to try to publish in. Begin writing paper.
May	Continue writing paper. End writing paper. Write a hand-off document to future students working on this project.

DELIVERABLES

The first semester report will be a document produced at the end of first semester (ending in mid-December) documenting the platform, our literature search, and our development progress.

The second is a paper to be submitted to be published in a journal or conference, documenting our research and our results.

The last is a hand-off document that would allow future students to continue our research and use our research platform.

IMPACT ON CREU

The project is a study of computer engineering and computer algorithms as they apply to path planning in robotics. Our project will give the team members valuable research experience in the robotics and computer engineering fields, and hopefully yield a publication that will help significantly with graduate school admissions.

TEAM MEMBERS AND RESPONSIBILITIES

STUDENTS

Elena Oleynikova (Elena.Oleynikova@students.olin.edu), Class of 2011, is majoring in Engineering with a Concentration in Robotics. She has done two years of robotics and computing research at the Olin Intelligent Vehicles Lab, including vision algorithms for person detection, stereo vision for object separation, and most recently building and programming the hexapod platform. She also worked at the Olin Intelligent Vehicles Lab last summer on the Autonomous Surface Vehicle Medea, and will present the paper "Perimeter Patrol On Autonomous Surface Vehicles Using Marine Radar" at *IEEE Oceans '10* in Sydney, Australia in May 2010.

Elena's responsibilities will include maintaining the robotics platform, processing sensor data, and helping develop and document the algorithm.

Stanislaw Antol (Stanislaw.Antol@students.olin.edu), Class of 2011, is majoring in Electrical and Computer Engineering. He has only recently become interested in Robotics, thus does not have much experience directly in the field besides having taken Olin's Robotics class and starting to work on this project. His various interests have led him to have a very diverse background. He participated in a Math REU program at Texas A&M, where he worked on a program to recognize cursive letters using Wavelets. He also participated in a Non-linear Dynamics REU at the University of Maryland-College Park where he implemented a time-delayed non-linear feedback circuit with continuously tunable time delay. His last experience has been working on setting up the data processing system for a new Neurobiology lab at Harvard Medical School doing research on color vision in primates. Thus, he has gained a lot of experience with various kinds of programming and is eager to apply his knowledge to this robotics project.

Stan's responsibilities will include developing the simulation, and helping develop and document the algorithm.

FACULTY ADVISORS

The first faculty advisor is Dr. David Barrett (David.Barrett@olin.edu), Associate Professor of Mechanical Engineering at F. W. Olin College of Engineering, and Direction of the Intelligent Vehicles Lab. Dr. Barrett will be advising the robotics, hardware, and testing aspects of the project.

The second faculty advisor is Dr. Mark Chang, (Mark.Chang@olin.edu), Assistant Professor of Electrical and Computer Engineering at F. W. Olin College of Engineering. Dr. Chang will be advising us on the computing and algorithms aspect of the research.

The team will meet with each faculty advisor once per week to discuss progress, problems, and future directions. The faculty advisors will provide advice, support, and ideas for the research.

REFERENCES

- [1] R. Ozawa, Y. Takaoka, Y. Kida, K. Nishiwaki, J. Chestnutt, J. Kuffner, J. Kagami, H. Mizoguchi, and H. Inoue, "Using visual odometry to create 3d maps for online footstep planning," in 2005 IEEE International Conference on Systems, Man and Cybernetics, vol. 3, 2005.
- [2] J. Chestnutt, K. Nishiwaki, J. Kuffner, and S. Kagami, "An adaptive action model for legged navigation planning," in Proceedings of the IEEE-RAS/RSJ International Conference on Humanoid Robots, vol. 11, Citeseer, 2007.
- [3] J. Latombe, Robot motion planning. Springer Verlag, 1990.
- [4] H. Chen and M. Dong, "3D map building based on projection of virtual height line," in IEEE Asia Pacific Conference on Circuits and Systems, 2008. APCCAS 2008 , pp. 1822–1825, 2008.
- [5] P. Lamon and R. Siegwart, "3D-Odometry for rough terrain-Towards real 3D navigation," in IEEE INTERNATIONAL CONFERENCE ON ROBOTICS AND AUTOMATION, vol. 1, pp. 440–445, IEEE; 1999, 2003.
- [6] M. Pivtoraiko, "Adaptive Anytime Motion Planning For Robust Robot Navigation In Natural Environments,"
- [7] M. Asada, "Map building for a mobile robot from sensory data," IEEE Transactions on Systems, Man and Cybernetics, vol. 20, no. 6, pp. 1326–1336, 1990.
- [8] J. Rajput and K. Hasan, "Design and implementation of a hexapod with 2-degree-of-freedom legs and its fuzzy-controller for the obstacle avoidance," in Electrical Engineering, 2009. ICEE '09. Third International Conference on, pp. 1 –6, april 2009.
- [9] G. Parker, P. Nathan, C. Coll, and N. London, "Evolving sensor morphology on a legged robot in niche environments," in Automation Congress, 2006. WAC'06. World, pp. 1–8, 2006.
- [10] J. Yang and J. Kim, "A fault tolerant gait for a hexapod robot over uneven terrain," IEEE Transactions on Systems, Man, and Cybernetics, Part B, vol. 30, no. 1, pp. 172–180, 2000.
- [11] S. Kaliyamoorthy, R. Quinn, and S. Zill, "Force sensors in hexapod locomotion," The International Journal of Robotics Research, vol. 24, no. 7, p. 563, 2005.
- [12] O. Janrathitikarn and L. Long, "Gait Control of a Six-Legged Robot on Unlevel Terrain Using a Cognitive Architecture," in Proceedings of the IEEE Aerospace Conference, 2008.
- [13] W. Flannigan, G. Nelson, and R. Quinn, "Locomotion controller for a crab-like robot," in 1998 IEEE International Conference on Robotics and Automation, 1998. Proceedings, vol. 1, 1998.
- [14] C. Ye and J. Borenstein, "A method for mobile robot navigation on rough terrain," in 2004 IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04, vol. 4, 2004.
- [15] D. Kim and R. Nevatia, "Representation and computation of the spatial environment for indoor navigation," in IEEE COMPUTER SOCIETY CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION, pp. 475–475, INSTITUTE OF ELECTRICAL ENGINEERS INC (IEEE), 1994.
- [16] X. Zhao, Q. Luo, and B. Han, "Research on the real time obstacle avoidance control technology of biologically inspired hexapod robot," in Intelligent Control and Automation, 2008. WCICA 2008. 7th World Congress on, pp. 2306–2310, 2008.
- [17] A. Hait, T. Simeon, and M. Taix, "Robust motion planning for rough terrain navigation,"

- in 1999 IEEE/RSJ International Conference on Intelligent Robots and Systems, 1999. IROS'99. Proceedings, vol. 1, 1999.
- [18] W. Chung, S. Kim, M. Choi, J. Choi, H. Kim, C. Moon, and J. Song, "Safe Navigation of a Mobile Robot Considering Visibility of Environment," IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, vol. 56, no. 10, p. 3941, 2009.
- [19] A. Chilian and H. Hirschm "uller, "Stereo camera based navigation of mobile robots on rough terrain," in Proceedings of the 2009 IEEE/RSJ international conference on Intelligent robots and systems, pp. 4571–4576, IEEE Press, 2009.

KRS: APPLICATION FORM FOR INDIVIDUAL RESEARCH PROJECT[®]

Applicant Name (Last, First, Middle): Chang, Mark, L.

Project Title: Ubiquitous Computing for Carepartner Relief Through Patient Independence

PROBLEM STATEMENT (1 Page)

Individuals with Alzheimer's disease (AD) often require extensive carepartner assistance. As the disease progresses, carepartners can experience frustration and burnout from the ever increasing care needs of patients. Current assistive technology research spans themes including patient monitoring, direct patient assistance, and enhancing communication and connectedness. An area often overlooked is employing technology aimed at relieving the carepartner through enabling AD patient independence. The questions driving this study are a) how to develop a computing platform that can act as a smart surrogate for the carepartner by providing answers to common questions and daily reminders for the AD patient, and b) how effective can such a solution be in providing relief to the carepartner?

These questions are significant because providing emotional and psychological relief for carepartners will likely increase their effectiveness and longevity in caring for an AD patient. In the early stages of AD, these carepartners are often family members or part-time professionals. If and when this care network becomes insufficient—often due to burnout—patients are placed (perhaps prematurely) in assisted living facilities. Prolonging independence delays the high cost of assisted living and the loss of autonomy and quality of life that often accompanies the transition.

Research in this area has often focused on delivering location and behavioral information about the AD patient to carepartners, or providing technologies to make the AD patient more autonomous. Little investigation has been done into how carepartner relief can be achieved through relatively simple technology used by the patient and customized by the carepartner. Tracking and behavioral monitoring do not address the moment-to-moment needs of the patient (without the carepartner present, no one answers the patient's questions), while patient-assistive devices must be extremely robust to restore autonomy and can have catastrophic failure modes (patient gets lost while walking to the grocery store). Our approach investigates low-risk methods for providing effective carepartner relief without the need for extensive infrastructure investment by focusing on emulating some common carepartner roles: reminding and repetitive question answering. Simplifying the research questions puts the answers within reach of today's technology, hopefully allowing this research to positively impact the lives of AD patients and their carepartners immediately.

A pilot study conducted in a partnership with the Massachusetts Alzheimer's Association by Co-PIs Dr. Stephen Schiffman and Mr. Aaron Boxer in 2006-2007 concluded that a device targeting the underserved community of carepartners using this approach has great promise. The team developed a proof-of-concept device that explored necessary functionality, user interface, and form factor through collaborative design with AD patients, carepartners, and expert AD consultants. The proposed research will allow for more focused research into platform development and evaluation through extensive patient and carepartner trials.

Our objectives for this research are to develop a handheld computing platform for AD patients that can act as a carepartner surrogate for simple questions and reminders, and to assess the merit of the platform and the approach through patient and carepartner field-testing, evaluations, and interviews. The long-range goals of this research will be to inform the Alzheimer's technology research community of the importance of carepartner health—emotional, physical, and psychological—with respect to AD patient quality of life.

KRS: APPLICATION FORM FOR INDIVIDUAL RESEARCH PROJECT[®]

Applicant Name (Last, First, Middle): Chang, Mark, L.

Project Title: Ubiquitous Computing for Carepartner Relief Through Patient Independence

WORK PLAN (5 Pages)**Project Description**

Individuals with Alzheimer's disease (AD) often require extensive carepartner assistance. As the disease progresses, carepartners can experience frustration and burnout from the ever increasing care needs of patients. The goal of this project is to provide carepartner relief through patient independence by employing a handheld computer acting as a carepartner surrogate. Our novel approach is to not focus computing technology on extremely hard problems with low failure tolerances, such as patient tracking and monitoring, or direct patient assistance. Instead, we propose a relatively simple, low-cost device targeted at the AD patient, customized by the carepartner, to provide daily reminders and answers to repetitive, commonly asked questions. By providing answers in the voice of the carepartner, and by being omnipresent, the device may lead to increased AD patient autonomy and reduced AD patient anxiety. By reducing the cognitive and interactive load of caring for an individual with AD, we hope to increase the effectiveness and longevity of carepartners to delay the transition of the AD patient into an assisted living facility.

A pilot study conducted in a partnership with the Massachusetts Alzheimer's Association by Co-PIs Dr. Stephen Schiffman and Mr. Aaron Boxer in 2006-2007 concluded that a device targeting the underserved community of carepartners using this approach has great promise. Through interviews with medical experts, carepartners, AD patients, and expert AD consultants, the team gained deep, personal insight into the disease and its affect on the patient/carepartner relationship. In addition to developing empathy for the intended user group, some key design guidelines were discovered in these interviews, including:

- AD patients often have difficulty learning new technologies, thus any technology must have simple, intuitive interfaces that ideally mimic already known interfaces
- AD affects everyone differently, with no complete set of symptoms and a varying rate of degradation; thus any technology must be adaptable to an individual's needs, and may not be applicable to everyone with AD
- Medication distribution and repetitive questions were two areas that greatly affected carepartners

From these user studies and extensive brainstorming, the team developed the concept of a question-answering carepartner surrogate. Utilizing paper prototyping techniques and low-fidelity prototypes (blue foam mock-ups) further interviews with carepartners and AD patients were done to design the interaction with the proposed handheld device. The research culminated in the development of a proof-of-concept device that began exploration into necessary functionality, user interface, and form factor. The proposed research will allow for continued research into platform development and evaluation through extensive patient and carepartner trials.

Procedures and Methods

The proposed research will be completed in four phases: 1) prototype development, 2) small-scale field study, 3) prototype refinement, and 4) larger-scale field study.

Phase 1: Prototype Development, July 2008 – May 2009

The objective of this phase is to develop an initial prototype appropriate for the small-scale field study in Phase 2. To accomplish this objective we will:

- 1.1: Select candidate handheld/portable hardware platforms

To be used only by the Alzheimer's Association,
KR and Associates, and their agents that evaluate this application.
DO NOT DUPLICATE

KRS: APPLICATION FORM FOR INDIVIDUAL RESEARCH PROJECT[®]

Applicant Name (Last, First, Middle): Chang, Mark, L.

Project Title: Ubiquitous Computing for Carepartner Relief Through Patient Independence

WORK PLAN (5 Pages)

- 1.2: Evaluate candidate hardware platforms and select one for concentrated development
- 1.3: Obtain Human Subjects / IRB approval
- 1.4: Solicit patients and carepartners to co-design interaction models and user interfaces
- 1.5: Develop the first iteration of the application and user interfaces through co-design with patients and carepartners

(1.1) Candidate hardware will be selected from among the many small-form-factor and handheld computing devices available commercially. This research project will focus on software and user interface development rather than hardware development, so finding an appropriate computing platform is crucial to the success of the project. (1.2) The candidate hardware platforms will be evaluated using many criteria, including, but not limited to computing performance, cost, size, weight, durability, screen size and resolution, battery life, multimedia capabilities, expandability, patient and carepartner feedback, and ease of programming. One platform will be selected for initial deployment and purchased, while another may be selected in Phases 3 and 4 if necessary.

(1.3) In parallel with these procedures, we will obtain human subjects approval through New England IRB. (1.4) Upon human subjects approval, AD patients and carepartners will be solicited through our expert collaborators at the Massachusetts chapter of the Alzheimer's Foundation, Dr. Paul Raia, PhD, Director of Patient Care and Family Support, and Gerald Flaherty, Vice President, Medical and Scientific Programs. We will seek to recruit patients in early-stages of the disease and their immediate carepartners as co-design partners in this phase. We hope to obtain four English-speaking groups that exhibit or have experience with varying physical ability—visual and auditory acuity, fine motor control of hands and fingers, hand and arm strength—and of a broad age spectrum. This will allow us to incorporate design attributes taking into account different levels of physical ability and computer/technology familiarity without being overwhelmed with an unmanageably large number of subjects.

(1.5) Through co-design techniques using paper prototypes, low-fidelity mock-ups, and medium-fidelity computer interfaces presented to AD patients and carepartners in interviews, we will develop the first iteration of the carepartner surrogate software and the carepartner customization software for the handheld device. Our previous research efforts have indicated voice-recognition software in conjunction with an easy-to-read touch screen interface as likely input methods for AD patients.

Phase 2: Small-scale field study, June 2009 – August 2009

The objective of this phase is to gather user feedback on the initial prototype through interviews and surveys. The feedback on our high-fidelity prototype will be used in the refinement of the device in Phase 3. To accomplish this objective we will:

- 2.1: Solicit additional patients and carepartners for field study partners
- 2.2: Deploy two devices into the field for trial and evaluation
- 2.3: Gather feedback through interviews and surveys
- 2.4: Begin analysis of user feedback and begin creation of design modification guidelines

(2.1) Through our collaborators at the Massachusetts chapter of the Alzheimer's Foundation, we will solicit subjects for a small-scale field study. These subjects may include some from the previous phase, but should include new subjects so we may obtain feedback from persons that have not already been involved in testing

To be used only by the Alzheimer's Association,
KR and Associates, and their agents that evaluate this application.
DO NOT DUPLICATE

KRS: APPLICATION FORM FOR INDIVIDUAL RESEARCH PROJECT[©]

Applicant Name (Last, First, Middle): Chang, Mark, L.

Project Title: Ubiquitous Computing for Carepartner Relief Through Patient Independence

WORK PLAN (5 Pages)

and developing the device. We will seek to solicit about four groups of subjects with similar characteristics as in Phase 1, however, it is important that we include patients and their actual immediate carepartners. This criterion is crucial as only the direct carepartner will be able to adequately customize the device response to the patient's repetitive questions, and the recognizable voice of the carepartner is a critical design feature of our system.

(2.2) Two devices will be deployed in parallel into patient homes for two two-week trials. As there may be some remaining bugs, the short trial will allow us to make any necessary changes to the software and redeploy for another two-week trial. (2.3) Once during each trial, user feedback will be solicited from both the AD patient and the carepartner through interviews and surveys. We will employ similar interview and observation techniques as (1.5) to determine the usability of the device for both the patient and carepartner. We will also use the interview and surveys to determine carepartner relief effectiveness and patient autonomy. These instruments are preferred in this context because the feedback is primarily the perception of patient autonomy and carepartner relief. In cases of perceptions, attitudes, and beliefs, surveys and interviews can be quite effective instruments¹. These instruments will be designed in conjunction with the carepartners to leverage their unique context-specific knowledge.(2.4) At the conclusion of the field study, we will begin analyzing interview transcripts and survey responses to inform the next phase.

Phase 3: Prototype refinement, September 2009 – August 2010

The objective of this phase is to complete a revision of the software and hardware systems. For the field study in Phase 4, 10-12 prototype devices will be developed. To accomplish this objective we will:

- 3.1: Finalize analysis of user feedback
- 3.2: Implement design changes through continued co-design with patients and carepartners
- 3.3: Create 10-12 prototype devices for field study

The refinement based on user feedback in this phase of the project differs significantly from that of the design cycle in Phase 1. Whereas in Phase 1 the co-design process focuses on interface and interaction design, the analysis of the feedback from the small-scale user study in (2.4) and (3.1) will inform design changes that address the performance of the system with respect to carepartner relief and patient autonomy. The feedback may require further brainstorming to develop solutions to possible problems. From the concepts generated from (3.1), we will take proposed design changes back to users to obtain feedback.

We will implement our design changes (3.2) and create 10-12 identical prototypes (3.3) for deployment in Phase 4. At this phase (3.2) we will evaluate the need to select an alternative computing platform and/or change our technical approach to accommodate user feedback with respect to increasing carepartner relief and patient autonomy.

Phase 4: Larger-scale field study, September 2010 – June 2011

The objective of this phase is to conduct a larger-scale field study to obtain significant data regarding the effectiveness of our approach. To accomplish this objective we will:

- 4.1: Solicit a larger group of patients and carepartners for field study partners
- 4.2: Deploy 10-12 devices into the field for long-term (1-3 months) trial and evaluation

¹ Tuckman, Bruce W., *Conducting Educational Research, Fifth Edition*, Belmont:Wadsworth Group/Thomson Learning.

To be used only by the Alzheimer's Association,

KR and Associates, and their agents that evaluate this application.

DO NOT DUPLICATE

KRS: APPLICATION FORM FOR INDIVIDUAL RESEARCH PROJECT[®]

Applicant Name (Last, First, Middle): Chang, Mark, L.

Project Title: Ubiquitous Computing for Carepartner Relief Through Patient Independence

WORK PLAN (5 Pages)

4.3: Gather feedback through interviews and surveys

4.4: Evaluate effectiveness of device on carepartner relief and patient autonomy

(4.1) Through our collaborators at the Massachusetts chapter of the Alzheimer's Foundation, we will solicit subjects for a field-study to include 10-12 groups of AD patients and their immediate carepartners. As in (2.1), it is critical to obtain patients and their immediate, direct carepartners. We may also attempt to solicit a wider range of age groups and physical capabilities to explore the impact of our approach on a more varied population. We may also attempt to select for carepartner variation, including non-professional (e.g. family members) vs. professional, and part-time vs. full-time.

(4.2) We will deploy devices into homes for longer-term trials (1-3 months) to better assess the impact of our approach as well as the robustness of our device. (4.3) Approximately twice during the trial, we will gather feedback through interviews and surveys, with the primary objective being to determine the effectiveness of carepartner relief through the use of the device. (4.4) We will analyze the feedback results to determine the effectiveness of our approach. As in (2.3), evaluation of effectiveness through surveys and interviews of users is the most accurate way to understand and quantify the impact of our device. In the case of the carepartner, the self-reporting of relief is more reliable than observation of the carepartner in field. From this, it is a matter of scale to quantify the level of relief, and user narrative to assess the quality.

7 Record of Teaching Materials

For each course I've taught, I have included the following materials:

- course syllabus
- example assignments

7.1 Computer Architecture

WHO

Mark L. Chang (mark.chang at olin dot edu)

TA

Marc Sweetgall and Lorraine Weis

WHAT

Computer Architecture!

WHEN

MTh 2-3PM, T 1-3PM

WHERE

AC 304

WHY

Because ECEs have to. Because you couldn't think of a better class to take, otherwise.

Communications

 ca@lists.olin.edu

Office Hours

By appointment.

Text

Patterson, Hennessy, *Computer Organization and Design: The Hardware/Software Interface*, Fourth Edition, Morgan Kaufmann. 

[Amazon Link](http://www.amazon.com/Computer-Organization-Design-Fourth-Architecture/dp/0123744938/ref=sr_1_1?ie=UTF8&s=books&qid=1251853999&sr=8-1). Samir Palnitkar's

Verilog HDL: A Guide to Digital Design and Synthesis is also recommended but not required.

Both texts are on reserve.

Prerequisites

Engineering, Math, and Physics Foundation. Programming background strongly recommended.

Topics Covered

Introduction to computer architecture, algorithms, hardware design for various computer subsystems, CPU control unit design, memory organization, cache design, and virtual memory.

Assignments

The major goals of the class are to familiarize you with basic structure of microprocessors. As part of this, students will develop a Verilog implementation of a simple RISC microprocessor based upon the MIPS instruction set.

Exams

There will be one midterm.

Attendance

At your own risk :).

Laptop Use

Please feel free to bring your laptop, but leave the closed and off during class.

Outline

The class will have the following approximate schedule. Material may be added or dropped based on class timing and progress.

- Introduction to processor architecture. Performance measures.
- Assembly language programming.
- Computer Arithmetic.
- Processor Datapaths & Control.
- Pipelining.
- Memory hierarchy, caches, virtual memory.
- Advanced topics in computer architecture.

Objectives

By the end of the course, students should be able to:

- design, build, and simulate a working processor
- write programs in assembly and machine code
- describe complex hardware systems in Verilog
- analyze, comprehend, and critique commercial and research processors
- research and give an oral presentation on an advanced topic in the field of computer architecture
- analyze and calculate the tradeoffs of implementing optimizations

Collaboration

Groups of 2-3 for labs, individual for homework. For any other assignments, details will be given.

- For the labs the intention is to work primarily with your partners. If you run into problems, discussion with your classmates is fine. If you use classmates outside of your group, please note who they are on your submissions. I am also readily available to answer questions.
- For homework, the intention is to work primarily alone. If you are stuck, or need help, discussion with your classmates is fine. Again, please annotate who you collaborated with on a per-problem basis. I am happy to take any questions regarding homework in my office, or via email.
- The design of this policy requires good self-monitoring. If you are constantly relying on others to help you through the problems, there is something amiss. The collaboration policy is designed to foster discussion and group learning.

ENGR 3410: HW#2

Number Systems & MIPS Assembly

Due 11:59pm October 2, 2008

The purpose of this homework is to be a diagnostic of your number systems, MIPS assembly, and machine code skills. Please show all work and annotate the people you worked with explicitly.

2.1 Number systems

Textbook problems 2.2, 2.3

2.2 MIPS instruction set

Textbook problem 2.4

2.3 Assembly coding

Textbook problem 2.6

2.4 MIPS assembly

Textbook problems 2.29, 2.59 (off of your CD). For convenience, problem 2.59 is:

Show the minimal MIPS instruction sequence for a new instruction called **swap** that exchanges two registers. After the sequence completes, the Destination register has the original value of the Source register, and the Source register has the original value of the Destination register. Convert this instruction:

```
swap $s0,$s1
```

The hard part is that this sequence must use only these two registers! [Hint: It can be done in three instructions if you use the new logical instruction. What is the value of (A xor B xor A)?]

2.5 Machine code

Using the MIPS program in Exercise 2.34 (bugs and all), determine the instruction format for each instruction and the decimal (base 10) value of each instruction field.

ENGR 3410: MP#3

Full MIPS Single-Cycle CPU

Due in class (with demo), November 5, 2009

1 The Problem

The purpose of this machine problem is to complete your simple 32-bit MIPS single-cycle CPU. The CPU instructions to be implemented are:

LW, SW, J, JR, BNE, XORI, ADD, SUB, and SLT.

The book will give some examples of how architectures are put together, and will be useful as you design your own CPU. For this CPU, you will use your two previous machine problems (the register file and the ALU) so you will need to have these fully functional before proceeding to work on your CPU. *Contact me immediately if you do not have working code you can use!*

I provide some fake memory Verilog modules, an assembler, some assembly code source, and some compiled machine code to get you started. You need to test more thoroughly! You should very much be writing your own assembly code to test your CPU.

2 Implementation Details

The data memory and instruction memory modules are provided in the files `datamem.v` and `instrmem.v`, respectively. Please find these files on the wiki. I also provide a bunch of test programs on the wiki to help you test the functionality of your CPU. You can change the program loaded by editing the `instr.dat` string in `instrmem.v`. You are responsible for coming up with the top-level testbench for this assignment—use previous machine problems' testbenches as guidance.

You will be given an assembler that will allow you to assemble your own assembly-language test benches into machine language that can be loaded into your system for testing. Please find a Windows command-line executable on the wiki. If you have trouble executing the assembler, please let me know immediately. *Please note, this is not a commercial assembler. Be gentle.*

The control logic for your CPU can (and should) be done in behavioral Verilog.

This machine problem is significantly more complicated from a system integration perspective than the previous ones. Certainly, there are many more failure points in this problem than in the previous as there is much more integration of parts. Additionally, if your previous machine problems do not work correctly, you will need to make sure they are functional before you can test your CPU.

My estimate for completion of this machine problem is approximately 30 person-hours.

3 Requirements

There is no top-level test bench! You are designing your CPU, you must design a way to reliably test it. These tests will be used to prove to us that your CPU works. We have our own tests. These will take the form of **machine code** that gets loaded into memories.

As before, you should be writing test benches for **every** module you build. You are getting good at Verilog, I understand, however, this is a big machine problem, and you will get something wrong in putting it all together. Tests are your friend.

One good way to structure your test bench is to make a generic CPU test bench that instantiates your entire CPU (control and datapath) as well as both memories. These memories are simply Verilog files that fake memories and fill them with values from another file. **Look at the code for examples of using memories.** Then, you just change the contents of the memories and voila, your CPU executes a different piece of code. Observe the results as generated in the register file or in the input to the data memory write line. You have to instrument these “probes” for any useful data to come out.

Write some assembly code! Assemble it with the provided assembler and make sure your CPUs work!

4 Deliverables

4.1 Write-Up

I expect a semi-formal “lab write-up” of this machine problem. It does not need to be as rigorous as lab notebooks in other, more experimental classes.

Your write-up (Word, L^AT_EX, PDF) and supporting Verilog code, should be placed in a single, well-named archive file (ZIP or TAR). Please name this file after your team. So if you are *Team Smack*, your directory would be “teamsmack”, and you would ZIP that up into a file “teamsmack_mp2.zip”.

I will provide a drop.io place for you to turn in your first deliverable before the deadline.

Other notes:

- You may turn in one deliverable for all group members
- Please use the drop.io system I will send out via email to turn in your work
- We expect all group members to participate in every aspect of this machine problem
- Please check out the tutorials on the class wiki. They are actually useful, I promise.

4.2 Demos

DEMOS ARE REQUIRED, WHETHER YOUR CODE WORKS OR NOT

The Demo is when your team convinces us that your implementation does what it was supposed to do. This is accomplished by your team running our test bench file. I will distribute this file about 12-24 hours before the demo time so you have a chance to plug everything together and see if it works.

The Demo time is also a time for us to gauge the level of involvement of each of the group members. Demos will be done *during class time on the due date*, or before the demo time, offline, with results submitted.

If you do not demo your assignment, your team will automatically get a zero. Missing your demo slot without prior approval will impose a late penalty on the entire assignment. All team members should be present for the demonstration unless a prior arrangement has been made.

HONOR CODE INFORMATION HERE: I expect that your team will complete the assignment **before** you try my test bench. I expect that your team will **not change the code after this time** except to correct errors. If you have to change the code, you must let the grader know that you have made modifications to the code. During the demo you must demonstrate your broken code first, then any modifications you did to fix it.

5 Hints and Tips

- Did you notice that there were tons of demo assembly codes and no real CPU test harness? You have to write it. It literally instantiates the CPU parts and ticks a clock. Easy. Now build the CPU.

- You must write some assembly code and test it to make this complete!

7.2 Digital VLSI

WHO

Mark L. Chang (mark.chang at olin dot edu)

WHAT

Digital VLSI

WHEN

TF 10-10:50AM, W 1-2:50PM

WHERE

AC 304 (lecture), AC 313 (lab)

Communications

 vlsi@lists.olin.edu

Office Hours

By appointment.

Text

CMOS VLSI Design, a Circuits and Systems Perspective (3rd edition), Weste and Harris, is suggested. Several additional reserve texts are also available.

Facilities

We will be using several software tools in this class developed by Cadence. These tools run only on UNIX machines and will be installed on campus computers so you do not have to install them on your laptop. If you want to run these applications on your own machines, you need to install an X server, or you can attempt to install the software directly on your Linux machine. A suggested X server is provided with the Cygwin ( <http://cygwin.com>). I will not be explicitly supporting remote operation or local installation, but feel free to ask me questions.

Prerequisites

Engineering, Math, and Physics Foundation. Computer Architecture is very helpful knowledge. If you are enrolled in Introduction to Microelectronic Circuits, there might be some overlapping material.

Topics Covered

We will be covering the basics of Digital VLSI Design, including a basic physical model of a metal-oxide semiconductor transistor, fabrication techniques and processes, and hierarchical design of large digital systems using computer-aided design tools. Some of the topics include the design and simulation of digital gates, finite state machines, combinational and sequential logic, static and dynamic logic families, memories, arithmetic units, data paths, and processors.

Objectives

By the end of the course, students should be able to design, implement, and verify a complex hierarchical digital design in CMOS technology. Beyond machine work, students should be able to understand and qualitatively analyze physical and electrical properties of VLSI designs; area, power, and speed tradeoffs between different CMOS logic families; floorplanning and clocking strategies for large digital designs; and basic VLSI fabrication techniques.

Assignments

There will be several machine problems designed to familiarize you with the CAD tools employed in VLSI design. The second half of the semester will focus upon an open-ended project where you design and implement a VLSI chip of your own. These will be optionally fabricated and tested.

Exams

There will be at least a midterm exam.

Attendance

At your own risk :).

Laptop Use

No laptop use during lecture-style classes unless otherwise noted.

Collaboration

Homework is done alone. Machine problems done alone. For all assignments, details will be given.

- For homework, the intention is to work primarily alone. If you are stuck, or need help, discussion with your classmates is fine. Again, please annotate who you collaborated with on a per-problem basis. I am happy to take any questions regarding homework in my office, or via email.

Spring 2007/Course Syllabus (last edited 2009-03-18 23:30:55 by localhost)

ENGR 3430: HW#0

This homework is due before class, February 16th, 2007.

0.1

Sketch a transistor-level schematic for a compound CMOS logic gate for the logic functions below. Balance rise and fall times by showing the relative size of each transistor.

- $F = \overline{ABC + D}$
- $G = \overline{(AB + C) * D}$
- $H = \overline{AB + C(A + B)}$

0.2

Use a combination of CMOS gates (drawn as schematic symbols) to generate the following functions from A, B, and C. You do not need to draw the transistor diagrams, design this at the gate level.

- $Y = A\bar{B} + \bar{A}B$ (XOR)
- $Y = \overline{AB} + AB$ (XNOR)
- $Y = AB + BC + AC$ (majority gate)

0.3

Sketch a transistor-level-schematic of a CMOS 3-input XOR gate. You may assume you have both true and complementary versions of the inputs available. Balance rise and fall times and annotate the size of each transistor.

ENGR 3430: MP#1

Full Adder

End of class, February 21, 2007

1 Problem description

In this machine problem, you will design and implement a one-bit full adder. This full adder will likely consist of an XOR and a NAND gate. There are plenty of other ways to implement the full adder, so you may use any combination of gates that work.

This full adder will be used in the next machine problem which will be to create and verify an 8-bit full adder/subtractor. This machine problem calls for *hierarchical* design techniques, which means designing, testing, and laying out several basic cells and combining them to make the larger circuits.

Your success depends very much on this style of hierarchical design. Trying it any other way will likely result in utter and complete failure, complete with in-class mocking.

Another emphasis of this machine problem is good layout practices. You should design each cell such that it interfaces well with other cells. In class we talked about the utility of “pitch matching” cells so that they snap together nicely. For some basic design styles and tips, visit your textbook, Section 8.8. For a design of this size and complexity, a floorplan for the cell placement and interconnection between cells is critical to obtaining an efficient design. *Part of your grade will be “style” points.*

This machine problem is to be done individually. Consultation with other students is fine, but please turn in your own work, citing those that helped and in what capacity.

2 Deliverables

You must do your design such that there are two distinct deliverables. First, please complete your design in a single directory. Something like “MP1” would be fantastic. I will be taking a snapshot of these directories on the due date and will be perusing them within the Cadence environment to explore your layouts in finer detail.

Second is a *single PDF file* that contains, for all your individual circuits:

- schematics
- digital simulation – Verilog test harness and waveform results
- analog simulation – schematic and waveform results
- layouts
- text of the LVS

The simulation should include a bit of narrative that explains what you were trying to prove, and a textual description of how your test proves your circuit is functional.

You do not need to turn in anything for circuits that you did in the previous machine problem, unless they have changed significantly.

3 Design

There is no good tutorial on how to perform good layout. Chapter 8 in your book, specifically Section 8.8 attempts to give you a few pointers, however, nothing can beat learning by doing.

Try to make your basic cells as small as possible. Your cells will turn out much smaller if you keep all the n-diffusion near the ground line and the p-diffusion near the Vdd line, much like we've done in class and in the sticks diagrams. You should have a Vdd bus running across the top of each cell, and a corresponding Gnd bus running across the bottom. In general, the inputs should be on the left and top of the cell, and the outputs on the bottom and right. Make all the cells the same height (or "pitch") to facilitate easy snapping together of different cells, matching up Vdd and Gnd lines in one continuous piece of metal. Use layout somewhat like Fig. 1 below as a design template or guide.

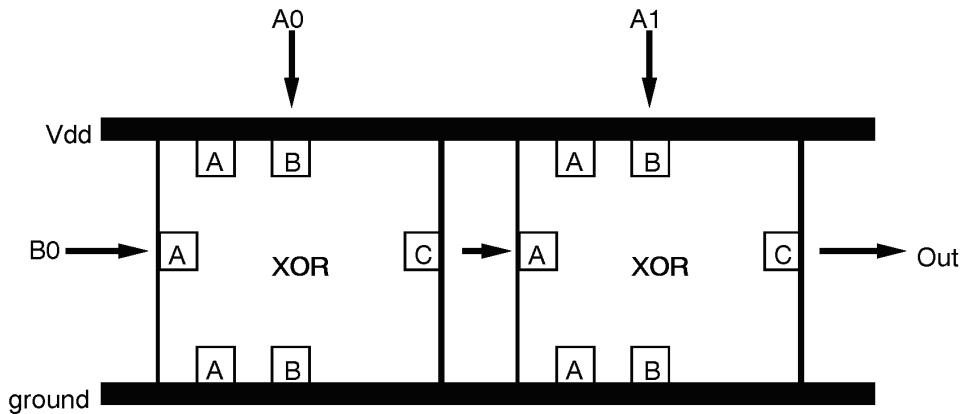


Figure 1: Out = B0 xor A0 xor A1. Note that inputs pass through the cells vertically and horizontally, and the inputs and outputs are aligned.

3.1 Design Challenge

The textbook, in Chapter 10—as well the digital logic book *Contemporary Logic Design* on reserve—discuss XOR designs using basic gates such as NORs and NANDs. Those work, but there is a 10-transistor implementation that is pretty neat. Avoid using the 6-transistor XOR design as shown in parts of Figure 10.61 in your textbook. These rely on pass-transistor logic that is difficult to simulate in digital simulators and may result in slower circuits. Don't bother using Google to find a solution. There is nothing out there.

You will get a 10% point bonus if you complete a 10-transistor XOR design.

You should put effort into making your layouts compact. For example, don't use your previous NOR gate layout in your XOR layout (that's a hint for design challenge participants). Customizing your design and placing all nMOS or pMOS transistors within the same diffusion area, for instance, can yield much smaller layouts. Remember, good and compact subcells are essential for good and compact overall designs.

Your layouts must also match exactly the transistor schematic. Verify this through LVS. It is a good practice to run the LVS on each cell as it is completed. Do not wait until you've put together the entire full adder to try and LVS your design. It won't work. I promise.

3.2 Adder cell design

Using the basic cells you've designed as subcells, build and test the following circuits: **HalfAdder**, **Ful1Adder**.

The HalfAdder should be created using an XOR and a NAND, the FullAdder using two HalfAdders and a NAND. Check the textbook and the Katz textbook in reserve if you do not see how to make these circuits. Those texts, or a quick truth table, will tell you all you need to know.

3.3 Half Adder Layout

The HalfAdder can be built from an XOR and a NAND. You should place an upside-down NAND (use the “rotate” function) below the XOR and move it close enough so that the XOR and NAND share the same ground line (see Fig. 2).

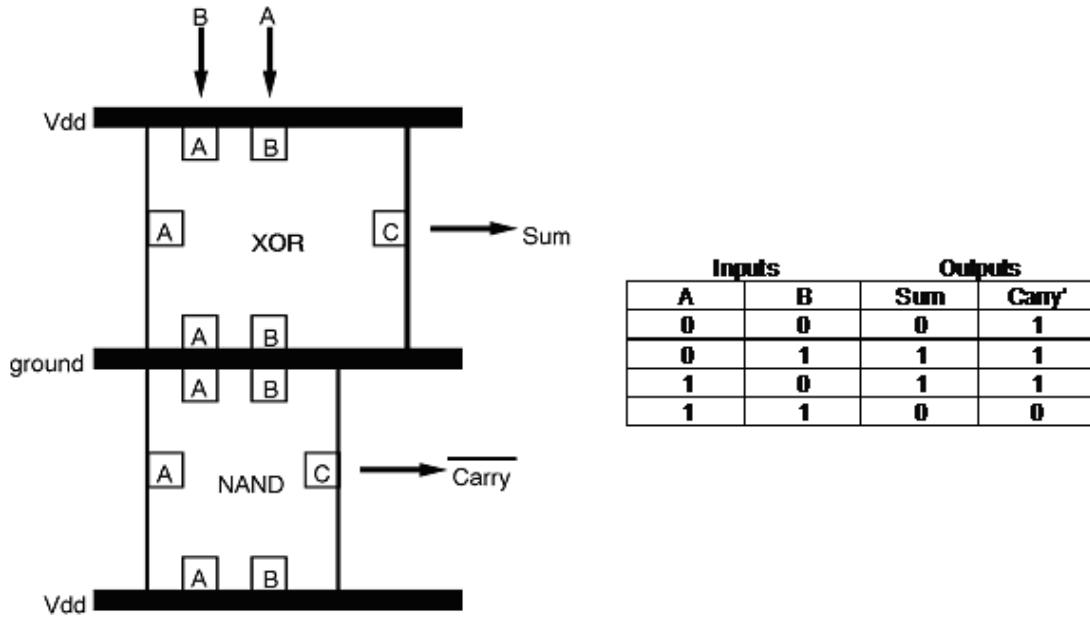


Figure 2: HalfAdder, floorplan and truth table.

3.4 Full Adder

The FullAdder is typically made up of two HalfAdders and a NAND gate. It should add A, B, and C_{n-1} to give Sum and C_n. The exact implementation can vary, so design it however it makes best sense to you. An example placement of cells is shown in Fig. 3. The Carry output from the first HalfAdder must be routed to the NAND through or around the second HalfAdder. Two possible ways of doing this are by using Metal 2 or by running a wire around the bottom of the cell.

Your finished FullAdder cell should have its inputs and outputs routed so that it will be pitch-matched with the other FullAdders in the 8BitAdder (next machine problem). A and B should enter the top of the cell and Sum should exit the bottom. The Carry signal should enter from the left and the CarryOut signal should exit to the right. This carry “chain” should cross the borders of the cell at the same relative position and be made of the same material so that they will be connected when cells are placed next to each other in the 8BitAdder.

4 Tips

- Completely cover adjacent contacts that are at the same potential

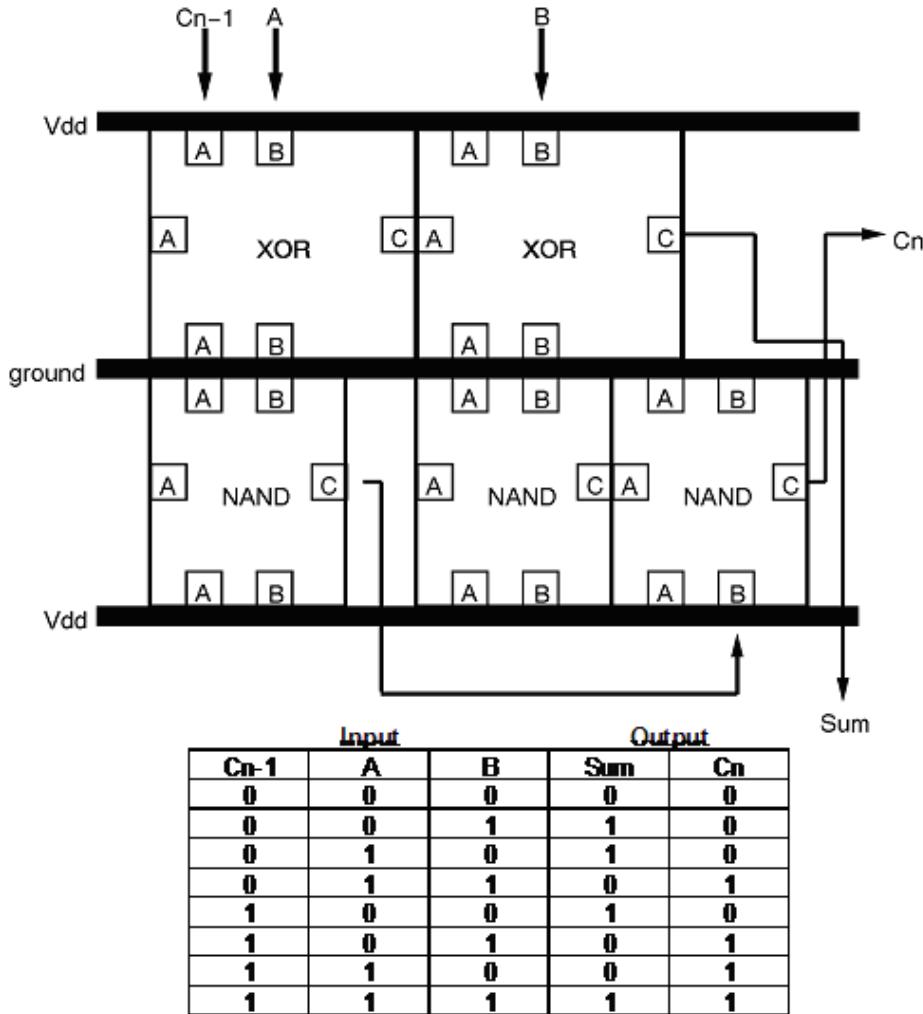


Figure 3: FullAdder, floorplan and truth table.

- Don't forget well contacts and substrate contacts. You should shoot for one per every 5 transistors, up to one per every 5 gates. *Every well must have at least one well contact.*
- Always consider the “big picture” of how each gate fits into the overall design and floorplan. Don't let the input get placed in weird places for no good reason.
- Sloppy lines don't really make for a poor-performing circuit, but they are often indicators of a not-well-thought-out floorplan. They can also negatively affect the use of the cell in a hierarchical design.
- Don't overpack your layout. Sometimes, inputs are jammed into the center of a very tight cell that will be almost impossible to get to when one wants to use the cell. Factor in the poly contact!
- Don't forget to connect Vdd and Gnd together so that the *entire circuit* has Vdd and Gnd at the top level of hierarchy. Your layout will probably LVS since they have the same label. It will *not* work, though.

7.3 Mixed Analog-Digital VLSI

WHO

Mark L. Chang (mark.chang at olin dot edu) and Bradley Minch (bradley.minch at olin dot edu)

WHAT

Mixed Analog-Digital VLSI (MADVLSI) I

WHEN

Tuesday/Friday 10-12 noon

WHERE

AC 304

Communications

 vlsi@lists.olin.edu

Office Hours

By appointment.

Suggested Text

CMOS VLSI Design, a Circuits and Systems Perspective (3rd edition), Weste and Harris, is suggested. Several additional reserve texts are also available.

Facilities

We will be using several software tools in this class developed by Cadence. These tools run only on UNIX-based operating systems, and thus, you will need to use Linux. We will make one machine remotely available for your use, and we will make a VMWare virtual machine downloadable so you can run Linux anywhere.

If you want to run Cadence from a remote machine, you need to install an X server. We suggest  <http://www.straightrunning.com/XmingNotes/>. I will not be explicitly supporting remote operation or local installation, but feel free to ask me questions.

Prerequisites

Circuits, Engineering, Math, and Physics Foundation. Computer Architecture is very helpful knowledge.

Topics Covered

We will be covering the basics of Digital and Analog VLSI Design, including a basic physical model of a metal-oxide semiconductor transistor, fabrication techniques and processes, and hierarchical design of large digital systems using computer-aided design tools. Some of the topics include the design and simulation of digital gates, finite state machines, combinational and sequential logic, static and dynamic logic families, memories, arithmetic units, data paths, and processors.

Objectives

By the end of the course, students should be able to design, implement, and verify a complex hierarchical digital design in CMOS technology. Beyond machine work, students should be able to understand and qualitatively analyze physical and electrical properties of VLSI designs; area, power, and speed tradeoffs between

different CMOS logic families; floorplanning and clocking strategies for large digital designs; and basic VLSI fabrication techniques.

Assignments

There will be several machine problems designed to familiarize you with the CAD tools employed in VLSI design. The second half of the semester will focus upon an open-ended project where you design and implement a VLSI chip of your own. These will be optionally fabricated and tested.

Exams

There will be at least a midterm exam.

Attendance

At your own risk :).

Laptop Use

No laptop use during lecture-style classes unless otherwise noted.

Collaboration

Homework is done alone. Machine problems done alone. For all assignments, details will be given.

- For homework, the intention is to work primarily alone. If you are stuck, or need help, discussion with your classmates is fine. Again, please annotate who you collaborated with on a per-problem basis. I am happy to take any questions regarding homework in my office, or via email.

2009-2010/Course Syllabus (last edited 2009-08-25 20:06:50 by MarkChang)

MADVLSI I: MP#0

Due beginning of class, September 11, 2008

1 Problem description

In this machine problem you will design the following four gates: **2-input NAND, 2-input AND, 2-input NOR, and 2-input OR**. You will likely use the inverter you designed during this past week. You will be responsible for doing the **transistor-level schematic, a symbol, digital simulation, analog simulation, layout, and LVS**.

This machine problem is to be done individually. Consultation with other students is fine, but please turn in your own work, citing those that helped and in what capacity.

2 Deliverables

You must do your design such there are two kinds of deliverables. First, please complete your design in a single directory. Something like “MP0” is appropriate. I will be taking a snapshot of these directories on the due date and will be perusing them within the Cadence tools to explore your layouts in finer detail.

Second is a *single PDF file* that contains, for all gates:

- schematic
- digital simulation — Verilog test harness and waveform results
- analog simulation — schematic and waveform results
- layouts
- complete text of the LVS

3 Hints and Tips

- Figure out how to transfer files from the VLSI machines to your own machine for editing the final “report”. SSH and SFTP are your friend, and they should be installed in the Windows partition of your machine if you are using the student image.
- Figure out how to compose a single PDF file early. Microsoft Word to PDF might be your friend. If you don’t have Adobe Acrobat installed in Windows, go down to IT and get it installed now.
- For all simulations—digital and analog—just showing raw waveforms and Verilog files will not be sufficient. You will be graded on your ability to craft test benches that adequately test your circuit and prove the proper functionality.

MADVLSI I: MP#1

Due beginning of class, October 2, 2009

1 Problem description

In this machine problem you will be doing a layout challenge. We give you a schematic, you do the layout. You try and make it as small as possible.

Your goal is to make your layout correct, as verified by LVS, and small as possible, in terms of square area. You will receive credit based on the area of the bounding box of your layout.

This machine problem is to be done individually. Consultation with other students is fine, but please turn in your own work, citing those that helped and in what capacity.

2 The Schematic

You can get the schematic by executing the following command from your shared `cadence` folder:

```
tar f xv /home/cadence/2009/mchang/mirror_adder.tar
```

This will extract the schematic into a directory called `mirror_adder`. Simply change into this directory and start Cadence and do the layout there.

The schematic is also attached as the second page to this PDF.

3 Deliverables

Please complete your design in the `mirror_adder` folder. I will be taking a snapshot of this directory on the due date and will be perusing them within the Cadence tools to explore your layouts in finer detail.

Secondly, simply report to me your bounding box dimensions (vertical and horizontal), in an email.

- The carry-generation circuit requires two inverting stages per bit. As mentioned above, minimizing the carry-path delay is the prime goal of the designer of high-speed adder circuits.
- The sum generation requires one extra logic stage, but this is not that important, since this factor appears only once in the propagation delay Eq. (7.4).

Although slow, the circuit includes some smart design tricks. Notice that in the first gate of the carry-generation circuit, the NMOS and PMOS transistors connected to C_i are placed as close as possible to the output of the gate. This is a direct application of a circuit-optimization technique discussed in Section 4.2—transistors on the critical path should be placed as close as possible to the output of the gate. For instance, in stage k of the adder, signals A_k and B_k are available and stable long before $C_{i,k}$ ($= C_{o,k-1}$) arrives after rippling through the previous stages. In this way the capacitances of the internal nodes in the transistor chain are precharged or discharged in advance. On arrival of $C_{i,k}$, only the capacitance of node X has to be (dis)charged. Putting the $C_{i,k}$ transistors closer to V_{DD} and GND would require not only the (dis)charging of the capacitance of node X but also of the internal capacitances.

The speed of this circuit can now be improved gradually by using some of the adder properties discussed in the previous section. First of all, the number of inverting stages in the carry path can be reduced by exploiting the inverting property—inverting all the inputs of a full adder cell also inverts all the outputs. This rule allows us to eliminate an inverting gate in a carry chain, as demonstrated in Figure 7.6. The only disadvantage is that this design needs different cells for the even and odd slices of the adder chain.

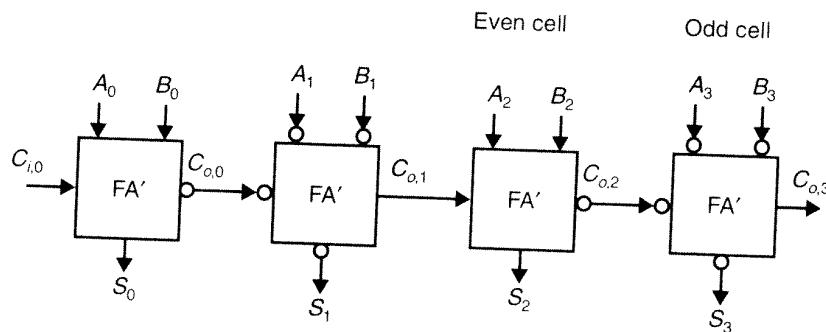


Figure 7.6 Inverter elimination in carry path. FA' stands for a full adder FA without the inverter in the carry path.

Improved Adder Design

An improved adder circuit, also called the *symmetrical, or mirror adder*, is shown in Figure 7.7 [Weste85]. Its operation is based on Eq. (7.3). The carry generation circuitry is worth analyzing. First, the carry-inverting gate is eliminated, as dictated by the previous section. Secondly, the PDN and PUN networks of the gate are not dual. Instead, they form a smart implementation of the propagate/generate/delete function: when either D or G is high, \bar{C}_o is set to V_{DD} or GND , respectively. When the conditions for a *Propagate* are valid

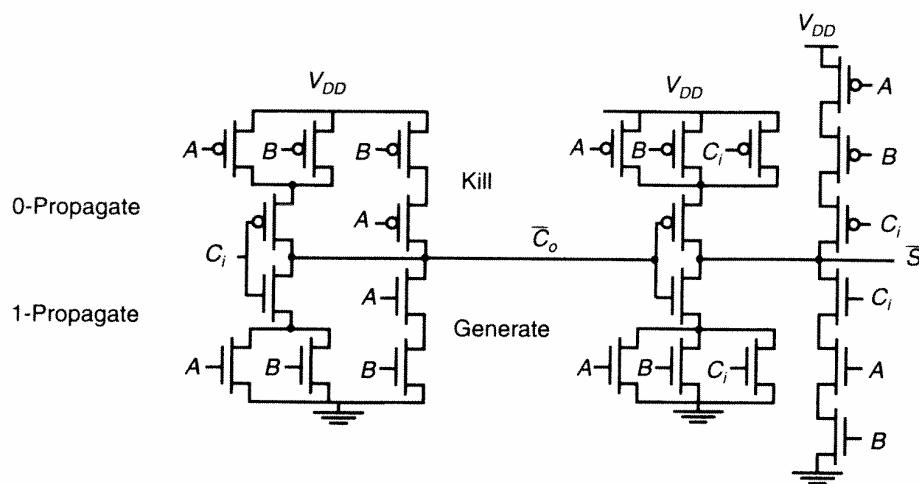


Figure 7.7 Mirror adder—circuit schematics.

(or P is 1), the incoming carry is propagated (in inverted format) to \bar{C}_o . This results in a considerable reduction in both area and speed. The analysis of the output circuitry is left to the reader. Some other observations are worth considering.

- This full adder cell requires only 24 transistors.
- The NMOS and PMOS chains are completely symmetrical. This guarantees identical rising and falling transitions if the NMOS and PMOS devices are properly sized. A maximum of two series transistors can be observed in the carry-generation circuitry.
- When laying out the cell, the most critical issue is the minimization of the capacitance at node \bar{C}_o . The reduction of the diffusion capacitances is particularly important.
- The capacitance at node \bar{C}_o is composed of four diffusion capacitances, two internal gate capacitances, and six gate capacitances in the connecting adder cell, or a total of ± 12 gate capacitances assuming that the diffusion capacitance is approximately equal to a gate capacitance (see Chapter 3). This is identical to the fully complementary implementation of Figure 7.6.
- The transistors connected to C_i are placed closest to the output of the gate.
- Only the transistors in the carry stage have to be optimized for speed. All transistors in the sum stage can be minimum-size.

Example 7.2 Static Adder Design

Consider a slight modification of the static ripple-carry cell of Figure 7.5. A differential approach is used; that is, every full adder cell generates both C_o and \bar{C}_o . The crucial gates of the cell are depicted in Figure 7.8. It is left as an exercise for the reader to fill in the rest of the cell (use transmission-gate EXORs as much as possible). The transistor sizes for our $1.2\mu\text{m}$ CMOS process are annotated on the schematics (in λ). Observe how a progressive sizing is used. Explain why the PMOS transistor connected to P is smaller than the one connected to C_{in} in the first gate.

7.4 Embedded Systems Design

This section is incomplete as many of the assignments were described in class and in videos posted online. I have included a few materials that were appropriate in paper format.

WHO

Mark L. Chang (mark.chang at olin dot edumacate) and Aaron Boxer (aaron.boxer @ you.know.where)

WHAT

Advanced Digital Systems

WHEN

T/F 1-3pm

WHERE

AC 304

Communications

 embedded@lists.olin.edu

Office Hours

By appointment.

Text

None.

Facilities

We will be using several software tools in this class provided by various vendors. You will be using your laptops as your primary development environment. Please make a lot of hard drive space available on your machines -- you will need it to install all the tools. Fortunately, we have site licenses for all the important tools we will be using in this course, so you can install the tools on your more powerful desktop machines if necessary.

Prerequisites

PoE and Computer Architecture are required.

Topics Covered

This course will explore the hardware/software boundary through a series of hands-on projects leading to the creation of "The Olin Gaming Console". Students will learn advanced digital design principles and techniques while designing four subsystems: audio, video, control, and network. These subsystems will be coupled with an on-chip processor to implement a gaming console system. The course will culminate with a multiplayer game demo.

Objectives

By the end of the course students should be able to design, implement, and verify a complex digital design using a hardware description language, soft-IP, and reconfigurable hardware. Students will be able to effectively partition computational tasks between software-based approaches and custom logic-based approaches, and fuse them together into a platform that meets real physical constraints.

Assignments

There will be several machine problems to get you to build the Olin Gaming

Console. Then, you get to write the game for the console.

Exams

There are no planned exams.

Attendance

At your own risk :).

Laptop Use

Always bring your laptops to class.

Collaboration

All work outside of the project will be done alone.

Spring 2008/Course Syllabus (last edited 2009-03-19 00:14:16 by localhost)

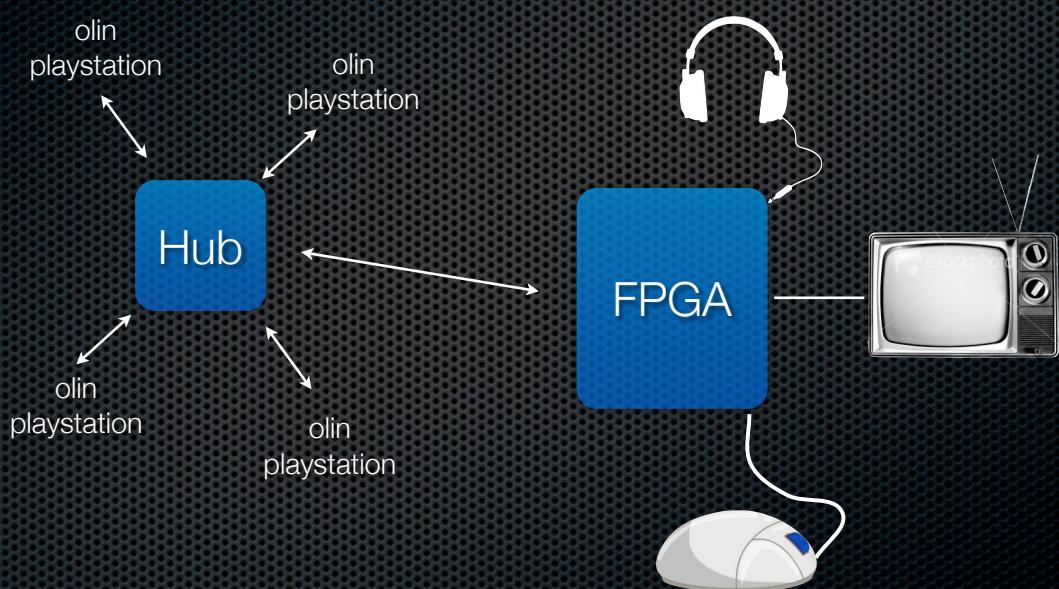
Advanced Digital Systems

Mark L. Chang and Aaron Boxer

Spring 2008

<http://embedded.olin.edu/>

the class project



~1h lecture on tuesdays
rest is lab time and discussion

block	date	topics
audio	jan 22	intro, admin issues, verilog basics, pulse width modulation and audio interface
mouse	feb 5	state machines, clock domains, bi-direction buses and PS/2 mouse interface
video	feb 19	on-chip and off-chip memories, resistor ladder DACs, CRTs and composite video interface
comm	mar 4	manchester encoding, clock synchronization and communication interface
	mar 18	spring break
game	mar 25 - may 12	game specification and design. all lab, discussion and status updates

due date	milestone
feb 5	audio
feb 19	mouse
mar 4	video
mar 25	communication
apr 1	game spec done
apr 15	tbd
apr 29	tbd
may 12	expo multiplayer demo

milestone grading policy

	quality (65%)	schedule (35%)
4	only minor bugs	on time
3	basic function demonstrated	< 1 week late
2	doesn't perform basic function	< 2 weeks late
1	doesn't compile	before end of semester
0	not submitted	not submitted

milestone grading policy

final grade == average of milestones

early submission incentive

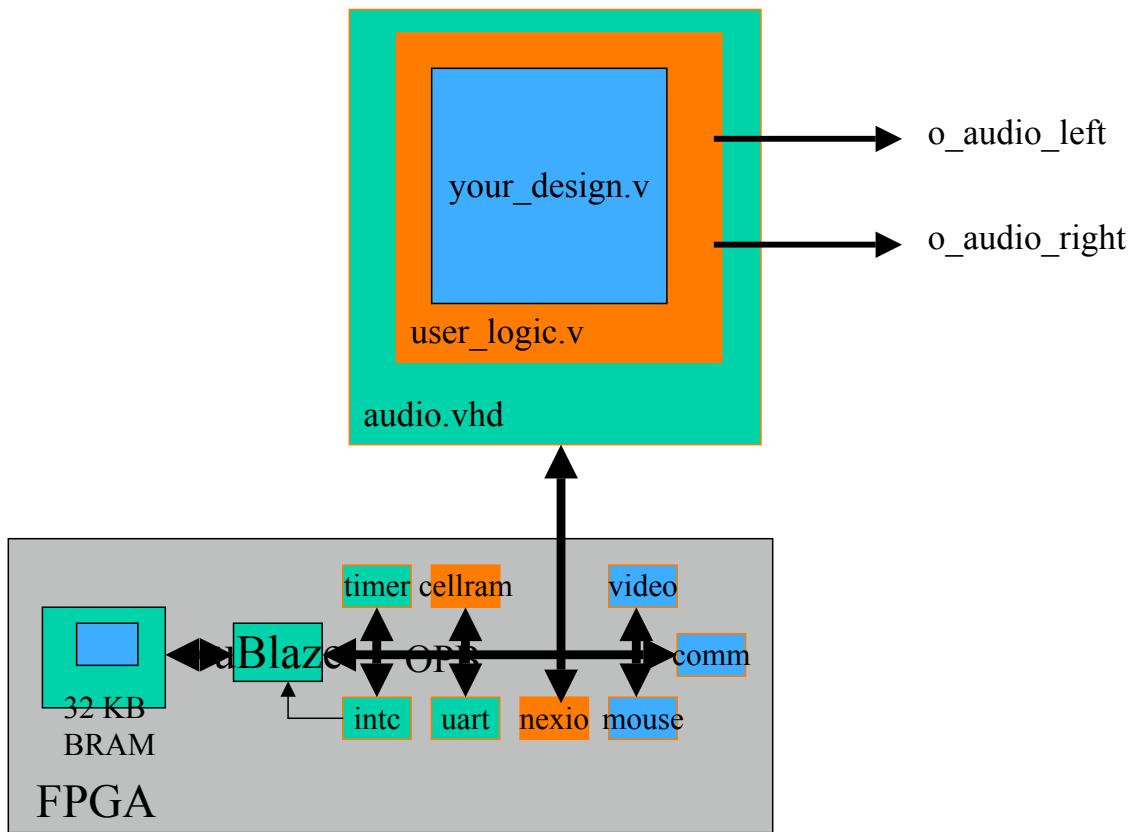
3 days early ...

feedback and grade given within 2 days
you may submit milestone again

honor code expectations

- work is to be done alone unless otherwise noted
- you may consult another person only after you have attempted the work in question
- others may only help/guide, may not swap code
- must annotate who and how you collaborated
- deadline is *the deadline*. to the second.
- *ask first, ask often.*

Audio Template



Design Resources

C:\EDK\hw\XilinxProcessorIPLib\pcores\opb_ipif_v2_00_h\doc\opb_ipif.pdf

Using the IP Slave Interface

Pages 58-60 figures 38-41 – register or memory model

Using the IP Master Interface

Pages 74-77 figures 50-52

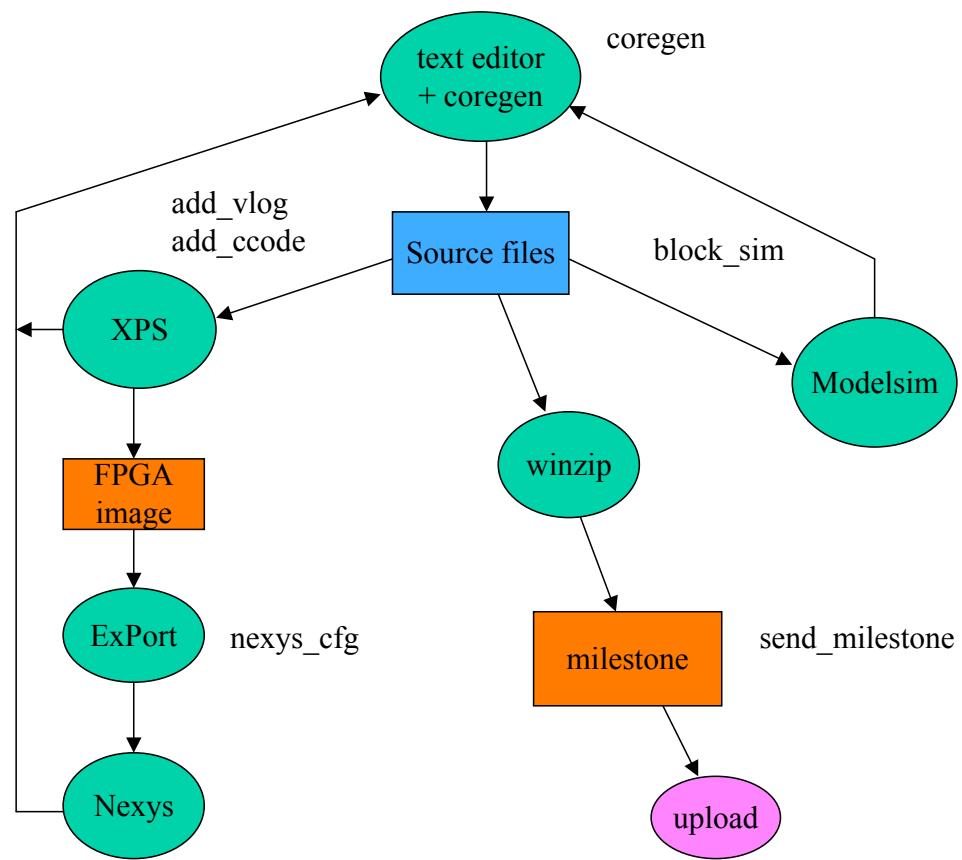
Online Documentation

Verilog Tutorial - <http://www.asic-world.com/verilog/index.html>

Verilog Reference - http://www.sutherland-hdl.com/on-line_ref_guide/vlog_ref_body.html

Pulse Width Modulation - http://en.wikipedia.org/wiki/Pulse-width_modulation

Design Flow Videos



Getting Started

- Install all the tools
- Download the oplay_console EDK template
- Work through the tools presentations – peripheral board not needed
 - Nexys_config - (oplay_console\tools\download.bit)
 - Add_sw – (oplay_console\game_sw\src\main.c)
 - Block_sim – (oplay_console\pcores\nexio_v1_00_a\sim)
 - Add_hw – (oplay_console\pcores\nexio_v1_00_a\hdl\verilog)

Audio Interface Spec

Show both channels operating independently

Show uBlaze controlling audio operation

The rest is up to you – just write it down!

7.5 Mobile Application Development



mobdev 2010
mobile
• design
• business
• technology

Mobile Application Development (mobdev) is an experimental course at Olin College offered by [Mark L. Chang](#). See our exploits from last year, <http://mobdev.olin.edu/2009>. Most everything for 2010 is located [in the blog](#).

The Course

The objective of the course is to investigate the mobile landscape through the lenses of design, entrepreneurship, and engineering. In the final project for the course, students will work in teams to develop commercially viable Android applications. We draw inspiration for this course from Hal Abelson's [Building Mobile Applications](#) at MIT, Stanford's [CS 193P iPhone Application Programming](#), and Maneesh Agrawala's [CS160 Introduction to Human Computer Interaction](#) at Berkeley.

The Idea

This course aims to offer an experience at the intersection between design, engineering, and entrepreneurship. Mobile Application Development leverages required coursework at Olin in software design, user-oriented design, and business and entrepreneurship, and applies these concepts in the mobile space. Technically, students will be learning about all aspects of the Google Android platform and SDK. Students, through Design, will engage in ideation, user study, and lightweight rapid prototyping. Using Entrepreneurship, students will unpack the mobile market space to find points of opportunity.

Due Tuesday 1/26/2010

Readings

- Application Fundamentals
- User Interface (all of it)
- UI Practices
- Developing on the Device

Readings
Check this out
The Assignment
Grading
Submission
Notes on Presentations
Rhan Kim
Ann Wu
Hyeontaek Oh
Andrew Pethan
Ben Fisher

Check this out

- On the internet: [Hello Views](#)
- Play with on the emulator:[API Demos](#)

The Assignment

1. Create a new Activity that includes at least 4 functional GUI elements that do something. For example, a radio list of states, a checkbox of colors, a slider for a value, and a button to "Go". Then output to a text element the state selected, the colors checked, and the value of the slider.
2. Use XML layout style, instead of programmatically.

Grading

- The following randomly-chosen people will be doing demos on Tuesday, 1/26/2010:
 - hyeontaek.oh@students.olin.edu
 - ann.wu@students.olin.edu
 - benjamin.fisher@students.olin.edu
 - andrew.pethan@students.olin.edu
 - rhan.kim@students.olin.edu
- If you cannot do this, notify me via email ASAP.
- I will check out the whole SVN tree, so please add a README.TXT or something that guides me. Probably not more than one sentence.
- Grading will be based upon your completing the assignment correctly and with minimal crashing. Real, working code is what I want here. Oversimplified GUI implementations will receive fewer points. If the GUI controls exist and **don't do anything**, that is not okay.

Submission

1. Check everything in to SVN
2. Copy the little XML snippet below, modifying **PROJECT_URL** with your SVN folder, and **USERNAME** with your acl username
 - you can find **user_folder** by right clicking on your project, selecting **Properties**, selecting **Subversion**, and copy paste the **URL** field.
3. Paste it between the triple squiggles at the bottom of this page

TEMPLATE

```
<project reference="0.9.3,PROJECT_URL,USERNAME" />
```

SUBMISSION LIST

```
<project reference="0.9.3,http://acl.olin.edu/svn/mobdev2010/tryan/Presenter">
<project reference="0.9.3,http://acl.olin.edu/svn/mobdev2010/ehwang/Metronom">
```

```
<project reference="0.9.3,http://acl.olin.edu svn/mobdev2010/jstanton/hw3_wi
<project reference="0.9.3,http://acl.olin.edu svn/mobdev2010/dbathgate/GuiEl
<project reference="0.9.3,http://acl.olin.edu svn/mobdev2010/awu/GUIElements
<project reference="0.9.3,http://acl.olin.edu svn/mobdev2010/echisa/Gui_home
<project reference="0.9.3,http://acl.olin.edu svn/mobdev2010/rkim/assignment
<project reference="0.9.3,http://acl.olin.edu svn/mobdev2010/mbejar/GuiTest/
<project reference="0.9.3,http://acl.olin.edu svn/mobdev2010/hoh/FirstWebBro
<project reference="0.9.3,http://acl.olin.edu svn/mobdev2010/ldethrow/fourwi
<project reference="0.9.3,http://acl.olin.edu svn/mobdev2010/keerthik/MD%20h
<project reference="0.9.3,http://acl.olin.edu svn/mobdev2010/psingal/SingalG
<project reference="0.9.3,http://acl.olin.edu svn/mobdev2010/adyer/GUIElemen
<project reference="0.9.3,http://acl.olin.edu svn/mobdev2010/bfisher/assignm
<project reference="0.9.3,http://acl.olin.edu svn/mobdev2010/jinman/gui_stuf
<project reference="0.9.3,http://acl.olin.edu svn/mobdev2010/jharley/P1%20-%
<project reference="0.9.3,http://acl.olin.edu svn/mobdev2010/cma/GUIElements
<project reference="0.9.3,http://acl.olin.edu svn/mobdev2010/jreed/Assignmen
<project reference="0.9.3,http://acl.olin.edu svn/mobdev2010/ntye/SimpleTime
<project reference="0.9.3,http://acl.olin.edu svn/mobdev2010/jmurray/jmurray
```

Notes on Presentations

Rhan Kim

- Date picker and time picker, gets time from computer and shows how much time till your event
- landscape view
- In switch from landscape to portrait, created another xml
 - Allows you to control the look in landscape mode so it looks good
 - When you're orientation switches, your application calls onCreate again
- If you don't put something in a scrollable it just rolls off the screen
- Android application need to think about lifecycle (onCreate, onPause, onDestroy, etc.) – need to address each of these, do I want to save state? There's a bundle where you can stash objects and then access

Ann Wu

- Draw squares or rectangles, you can choose the color
- Drop downs, slider, checkbox
- Event mechanism for slider, notices when slider changes (slider.setOnProgressChange), listener, read color value and change color
- Put in a scrollable view so that when view is switched content doesn't go off the screen
- Scale things with pixel density so scales on devices with difference resolutions

Hyeontaek Oh

- Web view with progress bar
- Menu items: go, refresh, forward
- Web view has get progress – used value to display progress in spinner

Andrew Pethan

- Color blindness tester
- State holding

Ben Fisher

- An application to help you file for unemployment checks
- Alert dialog
- Main Dialog class has a lot of subclasses that are easier to use dialogs
- Put space between text boxes using padding (in xml)

The Contest Guidelines

Ask questions if this is not clear.

Goal

To have a complete “pitch” to the judging panel by 3/5. This includes a presentation and demo that motivates your product vision. You need to convince the panel of the business merit, illuminate one or two key interactions/scenarios that your application implements, and demonstrate, live, on a phone, at least one scenario. All portions of the presentation need to be grounded in reality. For example, the business case should have some market numbers and competitive analysis. The design should illustrate the interactions and describe the need(s) that your application addresses. And the demonstration should be functional, with at least some real parts that interact with the echonest API, but can have many portions “canned” with the user using more of a script than a free run of the demo. Crashing is okay. Constant crashing to the point unusable is not.

Timeline

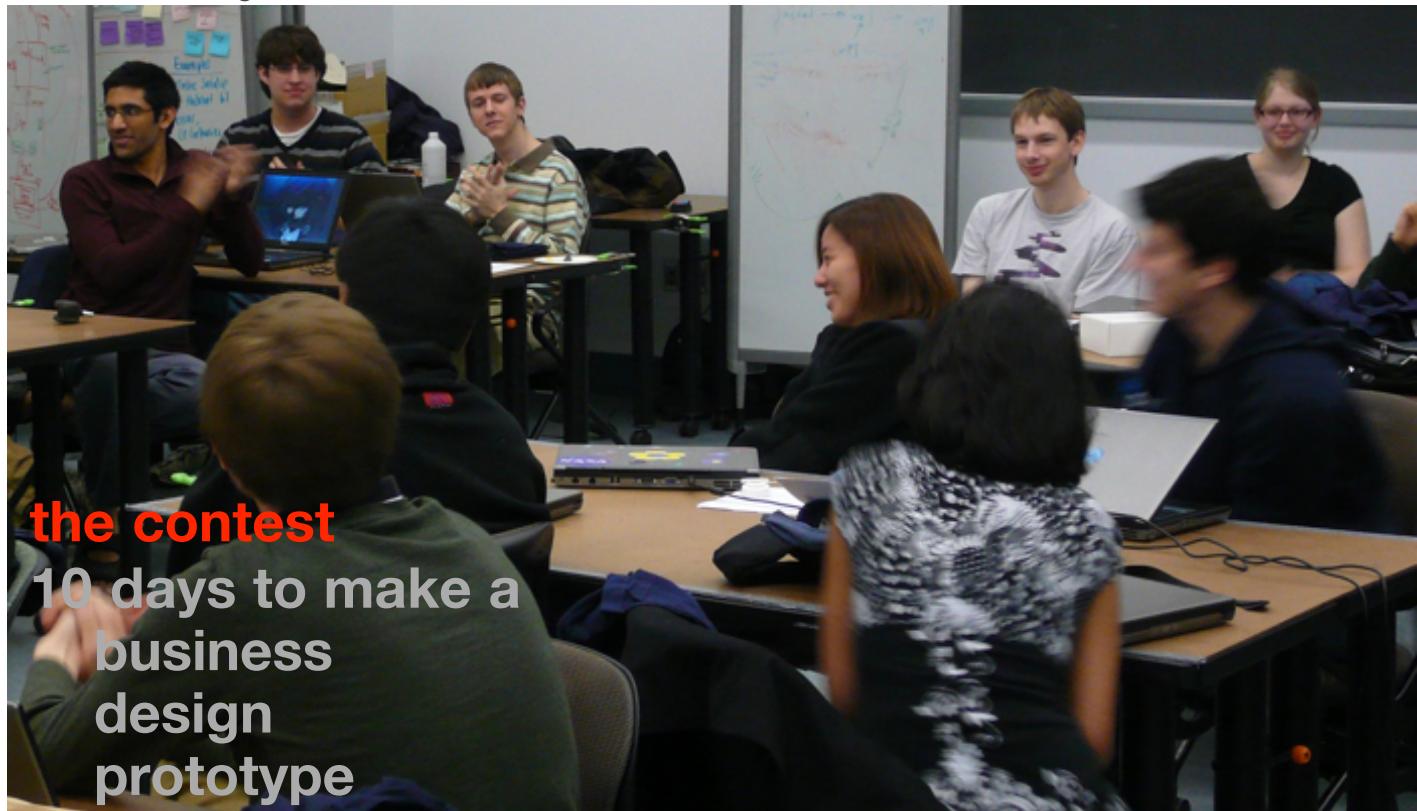
- 2/26: Shannon and Mark will walk around and help scope ideas. If you have enough of an idea already, paper prototypes would be a good starting point. By the end of the class period, you should have started on prototyping and have a few sentences/paragraphs about your idea so that we can send them to Paul for review.
- 3/2: You should have some working code at this point, and reasonable sense of the market analysis. Maybe a skeleton presentation. You will have all the time in class to work. Please come so we can help you. We will set up presentation run-through times to go over your presentations with Mark and Shannon outside of class. We want something polished for our guests!
- 3/5: Presentation to the panel. We will deliberate and announce the winner! Maybe have some cake.

The Teams

- Danny Bathgate, Andy Pethan, Tim Ryan, Ann Wu
- Noah, Benjamin, Rangel
- Poorva, Rhan, Hyeontaek, Miguel
- Jonathan Reed, Amy Dyer, Chujiao Ma
- Keerthik, John, Logan, John
- Ellen Chisa, Eric Hwang, Zach Kratzer, Jeff Stanton

Resources

- Decent overview of existing Android music apps



The Contest

The #mobdev contest this year was a 10-day sprint to create a compelling product prototype that exploited the [Echo Nest API](#). Teams had to pitch a business model, great design, and reasonable technical depth to our panel of five esteemed judges. Substance, style, and a convincing way to make money.

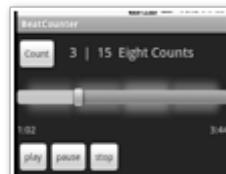
Every team did a fantastic job! The entire panel was very, very impressed at the amount of stuff our students were able to get done in such a brief amount of time. Cruise through the links to your right to find out more about the contest entries.

Note: since the presentations were done live, many of the presentation materials don't really support an "archival" mode. Thus, they are not as detailed as last year's contest entries. Oh well.

The Details



The Judges



Beat Counter



MusicTrails



DJMiXr - Winner



BeatBlocker



PacePlayer



Bandroid



Driving Beat

8 Record of Service Achievements

Reprints of work cited in section 4.

1. “Franklin W. Olin College of Engineering Certificate in Engineering Studies”, Mark L. Chang (and others), also available at <http://crossreg.olin.edu> (section 8.1, p.274)
2. “Exposure Analyais Preliminary Report”, Intercollegiate Relations Committee, February 10, 2008. (section 8.2, p.279)
3. “Proposal for 4-1 Program for Wellesley and Babson Students”, Mark L. Chang and Mark Somerville, 2010. (section 8.3, p.284)
4. “The Task Force on the Sophomore and Junior Years, Final Report”, July 17, 2007. (section 8.4, p.288)

Franklin W. Olin College of Engineering Certificate in Engineering Studies

Certificate in Engineering Studies: Olin College offers a Certificate in Engineering Studies for students at Brandeis University, Wellesley College, or Babson College. Certificate programs in a number of different disciplines are offered. The courses of study are designed to provide the student with a fundamental understanding of an engineering field, and typically consist of five courses ranging from first-year to upper-level.

Rationale for the Program: There are many paths to becoming an engineer. Not all of them begin with getting an undergraduate degree in engineering. For those students that want to explore the world of engineering studies—either out of curiosity, training to become professional engineers, or to prepare themselves for graduate study in engineering—Olin provides the Certificate in Engineering Studies. The Certificate provides a structured course of study in engineering that allows students to gain more exposure, more education, and more experience in the art and science of engineering. We believe that students completing the Certificate have the opportunity to expand their post-graduate options for careers or advanced study in technological fields by demonstrating a more significant degree of engineering expertise and a commitment to further engineering education.

Credit for Courses Taken at Home Institution: At most one course taken at a student's home institution covering equivalent material may substitute for an Olin course, whether or not it is used to also satisfy other requirements at the home institution. A list of already approved courses appears as a supplement to this document. Other courses will be considered upon request.

Prerequisites: The Certificate programs do not require any engineering prerequisites. However, most Olin engineering courses have general math and science prerequisites, such as calculus, physics, or biology. The Electrical and Computer Engineering program and Engineering Systems program may also require previous course experience in software design or discrete mathematics. These non-engineering prerequisite courses will normally not be taken at Olin. Courses taken at Olin must be chosen to avoid substantial overlap in subject matter with courses taken at the student's home institution. For each course in the Certificate program, generalized prerequisite information is provided as a supplement to this document.

Advising Structure: Managing challenging coursework from more than one institution is a complex task that should not go unsupervised. For students enrolling in the Program, the Certificate Program Coordinator will appoint an Olin faculty member to serve as their disciplinary advisor. The disciplinary advisor will coordinate the course of study with the student and the student's home institution academic advisor to ensure that the proposed program is coherent and appropriate. In addition to individual advising, the Certificate Program Coordinator will hold an information and orientation session each semester to help answer questions for interested students.

Enrolling in a Certificate Program: Students wishing to participate in the Program should be considering Olin courses as early as their sophomore year. In general, students will require strong backgrounds in math and science in order to meet prerequisite requirements. Prior to enrolling in the Olin Certificate in Engineering Studies courses, students are encouraged to discuss the program with their academic advisor (at the home institution) and to receive her or his approval. Prior to enrolling in their first course, students should contact the Certificate Program Coordinator at Olin College, who will direct them to their disciplinary advisor to craft a plan of study. Finally, students should contact their Olin course instructor(s) to discuss their preparation and receive her or his approval to enroll. The registration process is supported by the cross-registration procedures of the home institution. Information should be available through the student's Registrar's Office.

A note about Olin courses: Olin courses typically have significant project components and normally require considerable team-based work. Non-Olin students should be prepared to work closely with their Olin counterparts, both inside and outside class.

Programs of study: There are six programs of study proposed for the Certificate in Engineering Studies; details of the programs are appended. Note that not all courses are offered each semester, and the course offerings are updated frequently; students should discuss course selection with their Olin advisor. In general, students may take elective courses that are not specifically listed in the program description with the approval of their Olin advisor and the instructor. Full details regarding courses (hours, prerequisites, credits) can be found in the Olin Catalog.

Example Course Plans: A number of example course plans outlining a four-year experience that includes an Olin Certificate in Engineering Studies appear as a supplement to this document. Given the number of possible combinations of majors and certificate programs, these examples illustrate only a fraction of the options available to interested students.

For more information, please contact the Certificate Program Coordinator, Professor Mark L. Chang, mark.chang@olin.edu or 781 292-2559.

Engineering Design:

The Engineering Design Certificate prepares students to address important societal and environmental needs through design thinking. Students work individually and collaboratively to understand people and their needs, to manage creative processes to transform ideas into prototypes of new products and services for meeting those needs, and to shape those ideas to address perspectives such as usability, sustainability and manufacturability.

- ENGR1200 Design Nature
- ENGR2250 User-Oriented Collaborative Design

Three of the following courses:

- AHSE1500 Foundations of Business and Entrepreneurship
- ENGR3210 Sustainable Design
- ENGR3220 Human Factors and Interface Design
- ENGR3380 Design for Manufacturing
- Other approved elective course

Materials Engineering:

Materials engineering is a field that integrates the physics and solid-state chemistry of solids with engineering considerations, such as mechanical and electrical properties. In addition, emphasis is placed on processing and applications of materials.

- SCI1410 Materials Science and Solid State Chemistry

Four courses chosen from among the following:

- ENGR3450 Semiconductor Devices
- ENGR3810 Structural Biomaterials
- ENGR3820 Failure Analysis and Prevention
- ENGR3830 Phase Transformation in Ceramic and Metallic Systems
- ENGR38xx Engineering Polymers
- ENGR36xx Biomedical Materials
- SCI3120 Solid State Physics
- Other approved elective course

Bioengineering:

Bioengineering is a field that integrates engineering devices and systems with biological systems. The Olin program places a strong emphasis on the fundamentals of both biology and engineering.

- | | |
|----------|---|
| ENGR3600 | Topics in Bioengineering |
| SCI1410 | Materials Science and Solid State Chemistry |

Three courses chosen from among the following:

- | | |
|--------------------------------|---|
| ENGR3810 | Structural Biomaterials |
| ENGR36xx | Biomedical Materials |
| ENGR3699 | Special Topics in Bioengineering (<i>topics vary by year</i>) |
| SCI2110 | Biological Physics |
| SCI2120 | Biological Thermodynamics |
| SCI2210 | Immunology |
| SCI3210 | Human Molecular Genetics in the Age of Genomics |
| Other approved elective course | |

Electrical and Computer Engineering:

The Electrical and Computer Engineering (ECE) program introduces students to the fundamentals of electrical engineering, including the devices and structures of computing and communication systems.

- | | |
|----------|---------------------------|
| ENGR1200 | Design Nature |
| ENGR2210 | Principles of Engineering |

Two of the following courses:

- | | |
|----------|--|
| ENGR2410 | Signals and Systems |
| ENGR2420 | Introduction to Microelectronic Circuits |
| ENGR2510 | Software Design |
| ENGR3410 | Computer Architecture |
| ENGR3420 | Analog and Digital Communications |

One additional course chosen from among the following:

- An additional course from the list above
- | | |
|--------------------------------|-----------------------|
| ENGR3390 | Robotics |
| ENGR3425 | Analog VLSI |
| ENGR3430 | Digital VLSI |
| ENGR3440 | Modern Sensors |
| ENGR3450 | Semiconductor Devices |
| Other approved elective course | |

Mechanical Engineering:

The Mechanical Engineering (ME) program introduces students to the design of mechanical, thermal, and fluid systems.

ENGR1200 Design Nature

Two of the following courses:

- ENGR3310 Transport Phenomena
- ENGR3320 Mechanics of Solids and Structures
- ENGR3330 Mechanical Design
- ENGR3340 Dynamics
- ENGR3350 Thermodynamics

Two additional courses chosen from among the following:

- An additional course from the list above
- ENGR3360 Topics in Fluid Dynamics
- ENGR3370 Controls
- ENGR3380 Design for Manufacturing
- ENGR3390 Robotics
- ENGR3820 Failure Analysis and Prevention
- Other approved elective course

Engineering Systems:

The Engineering Systems program focuses on the design of products that integrate significant technology from multiple engineering disciplines, with a focus on products that merge mechanical and electrical systems.

ENGR1200 Design Nature
ENGR2210 Principles of Engineering

ECE requirement: one of the following courses:

- ENGR2410 Signals and Systems
- ENGR2420 Introduction to Microelectronic Circuits
- ENGR3420 Analog and Digital Communications

ME requirement:

ENGR3320 Mechanics of Solids and Structures

Integrative course:

ENGR3710 Systems

*Intercollegiate Relations Committee
Exposure Analysis Preliminary Report
Sunday, February 10, 2008*

Executive Summary

The purpose of this report is to begin to quantify the risk to the College associated with cross-registration at the three partner institutions: Babson, Brandeis, and Wellesley (referred to hereafter as the BBW schools). This report only presents the data that the committee has collected thus far, and suggests avenues of analysis that may be taken by the committee in the future.

Cross-Registration Activity

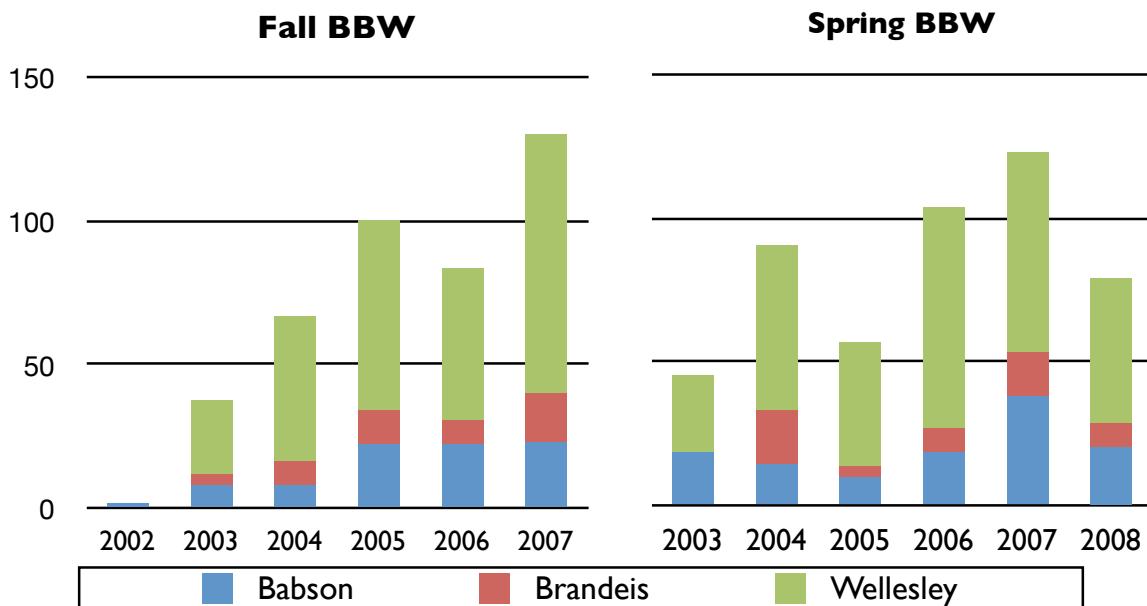
This information was provided by the Olin Registrar's office. Enrollment for Spring 2008 is preliminary, as the add/drop date for Wellesley cross-registration has not yet been reached. Thus, students from Olin are very likely to be adjusting their schedules with respect to taking courses at Wellesley.

Number of courses taken by Olin students at BBW schools

School	Fall							total
	2002	2003	2004	2005	2006	2007		
Babson	1	7	7	21	21	22		79
Brandeis	0	4	9	12	9	18		52
Wellesley	0	26	50	66	53	90		285

School	Spring							total
	2003	2004	2005	2006	2007	2008		
Babson	18	13	9	18	37	19		114
Brandeis	0	19	4	9	16	9		57
Wellesley	27	57	43	77	70	50*		324

* Preliminary data.



Number of courses taken by BBW students at Olin

School	Fall							total
	2002	2003	2004	2005	2006	2007		
Babson	0	2	0	0	1	4		7
Brandeis	0	0	0	0	1	0		1
Wellesley	0	0	0	1	17	3		21

School	Spring							total
	2003	2004	2005	2006	2007	2008		
Babson	0	7	2	0	4	2		15
Brandeis	0	0	0	0	0	0		0
Wellesley	0	0	10	14	11	19		54

Analysis of Olin Faculty Teaching Contributions

It is recognized that Olin faculty provide seats for BBW students outside of standard cross-registration. However, these seats are not accounted for in our statistics if the course is taught at partner institutions without an Olin course number. For example, Gill Pratt and Brian Storey have collaborated with Wellesley to teach “Introduction to Engineering Science.” This course has a Wellesley course number, therefore it does not appear in our cross-registration statistics.

These contributions are difficult to track. The IRC will make an effort to capture this data more completely in the next revision of this report. The committee has the following approximate data for Wellesley, and will be working to obtain more accurate data for all schools.

Professor	Course	Term
Sarah Spence Adams	Special Topics in Cryptology and Coding Theory	Fall 2005, Spring 2006
Caitrin Lynch	Culture, Knowledge, and Creativity	Spring 2005
Caitrin Lynch	Everyday Life in South Asia	Spring 2006
Burt Tilley	Introduction to Mathematical Modeling	Winter session 2006
Brian Storey & Gill Pratt	Introduction to Engineering Science	Winter session 2006
Helen Donis-Keller	Topics in Human Genetics with Laboratory	Winter session 2007
Joanne Pratt	Immunology	Fall 2007
Richard Miller	Leadership and Ethics	Various

Financial Risks

With such a significant number of Olin students taking advantage of cross-registration at the BBW schools, the College is potentially at a financial risk if these partner institutions cease to provide free access for our students. Specifically, since 2002, the net number of courses taken by Olin students at Babson, Brandeis, and Wellesley are, respectively, 171, 108, and 534.

An estimate of the historical financial “liability” of the College can be summarized in the following table:

	Babson*	Brandeis^	Wellesley*
Annual tuition (undergraduate)	\$ 34,112	\$ 34,556	\$ 34,770
Typical number of credits/year taken by a student	32		
typical number of credits per course	3.50		
average number of courses/year taken by a student	9.14		8
tuition per course	\$ 3,731	\$ 4,320	\$ 4,346

* 2007-8 tuition per course calculated by dividing annual tuition by average number of courses per year.
^ tuition per course for degree credit taken from Brandeis university bulletin (provisional 2008-9)

From this tuition per course estimate, we can obtain the following estimated pro-rated tuition costs for the net number of courses taken by Olin students at BBW schools.

	Fall							
School	2002	2003	2004	2005	2006	2007	total	
Babson	\$3,731	\$18,655	\$26,117	\$78,351	\$74,620	\$67,158	\$268,632	
Brandeis	\$0	\$17,280	\$38,880	\$51,840	\$34,560	\$77,760	\$220,320	
Wellesley	\$0	\$113,002.50	\$217,312.50	\$282,506.25	\$156,465	\$378,123.75	\$1,147,410	

	Spring							
School	2003	2004	2005	2006	2007	2008	total	
Babson	\$67,158	\$22,386	\$26,117	\$67,158	\$123,123	\$63,427	\$369,369	
Brandeis	\$0	\$82,080	\$17,280	\$38,880	\$69,120	\$38,880	\$246,240	
Wellesley	\$117,348.75	\$247,736.25	\$143,426.25	\$273,813.75	\$256,428.75	\$134,733.75	\$1,173,487.50	

Thus, to cover the costs of pro-rated tuition for Olin students to take courses at the BBW schools for the 2007-2008 academic year might have cost the College \$760,083. This, of course, might be considered a worst-case scenario if the College were forced to pay at the full pro-rated tuition rate. Note that the data for Babson does not include the early financial relationship we maintained for AHS Foundation courses taught at Babson.

Curricular Risk

Beyond simply financial implications, the potential loss of easy cross-registration with the BBW schools could have curricular and enrollment risks. Not having access to BBW courses would severely impact our ability to deliver breadth and depth in humanities and social sciences. From an enrollment perspective, students may opt not to apply or enroll at Olin if we cannot provide access to the curricular experiences they are so clearly obtaining at the BBW schools.

Without the ability to provide humanities courses, we potentially run the risk of negatively impacting our NEASC accreditation. Certainly, this poses a significant risk to the College and warrants more investigation.

Analysis and Recommendations

The IRC cautions that this report is preliminary. The committee has not begun to analyze the impact of this data, and emphasizes that the data can be interpreted in many ways. Without framing and context, this information could lead to haphazard actions that would not benefit the College and its partners. At this time, the committee offers possible avenues of analysis that we may undertake in the near future. The IRC also plans to partner with the AHS Committee, which is working on an analysis of and recommendations about many overlapping issues.

One analysis that might be performed is how the “trade imbalance” may be lessened through the hiring of more full-time faculty at Olin to teach courses that our students are seeking elsewhere. At first glance, many of these courses would likely fall into the category of AHS/E. The AHS Committee has already begun the task of analyzing the role of AHS in the Wellesley/Olin relationship and how additional Olin AHS faculty and the structure of the Olin AHS

curriculum could lessen the burden on Wellesley. However, this analysis is not yet finished. The IRC strongly suggests incorporating their findings into future reports.

Another axis of analysis that has yet to be done is discovering the specific types of courses that, if offered at Olin, would have the most positive impact in reducing the cross-registration traffic. For instance, it may be that providing foreign language instruction or adjusting the AHS requirement at Olin would significantly reduce the amount of cross-registration. Again, the committee stresses that this analysis has not been done.

Finally, it is important to keep in mind the pool of students from BBW schools from which we may potentially draw to Olin. For example, it is unreasonable to assume that every Wellesley student would have an interest in taking an Olin course. In reality, students from Physics and Computer Science may represent our best prospects for attracting cross-registration for our engineering courses. The Wellesley-Olin Working Group committee status report to Michael Moody from October of 2006 provides the following potential pool of Wellesley students:

Major	Average # of Graduates per year
Biology	25
Biological chemistry	15
Neuroscience	25
Physics	6
Computer Science	15
Mathematics	15
Chemistry	13
Total	114

Doing this analysis for each school would yield important information regarding our relative success in attracting students to our campus for engineering courses. A similar analysis would need to take place to determine the pool of students that may be interested in Olin humanities courses. This information would be invaluable in maintaining a healthy relationship with our partner institutions.

Conclusion

This initial report is meant to highlight the importance of our relationships with our partner schools and how one might begin to quantify the risks associated with our continued utilization of these curricular (and other) resources. The IRC will begin to conduct an analysis of these risks as they relate to the sustainability of the College and its mission.

Proposal for 4-1 Program for Wellesley and Babson Students

Summary

We propose a “4-1” program in which students from Wellesley and Babson would be able to complete two degrees in five years: one from their home institution, and an engineering degree from Olin. Such a program would provide significant benefit to both Wellesley and Babson, would help Olin address its “balance of trade” with Wellesley and Babson, and would provide Olin with a small number of transfer students.

In this document we summarize the mechanics of the proposed program, outline some of its unique features, and highlight some of the benefits and risks of the proposal.

Mechanics

Enrollment: We propose that during the first four years of study, the student would be enrolled at his or her home institution (and is paying tuition there). During these years the student:

1. completes all degree requirements for their degree at the home institution;
2. completes mathematics and science requirements for the engineering degree (typically at their home institution)
3. completes 4-6 engineering courses through cross-registration at Olin (e.g., by completing an Olin certificate program)

In the fifth year of study, the student would enrol as a full-time student at Olin, and completes the remaining engineering courses required for the Olin degree.

Admission: We imagine that students would have to go through a simple application process to be admitted as full-time students at Olin (e.g., in order to be admitted, students must have completed sufficient work towards the engineering degree, and have a minimum GPA in relevant courses).

Financial Aid: During years 1-4, we imagine the student would be at his or her home institution, and the home institution would deal with all financial aid and tuition issues. In year 5, we imagine the student would be a “normal” Olin student – i.e., he or she would receive normal Olin financial aid, including the 50% tuition scholarship.

Feasibility: We have done initial calculations using a number of degrees from both Wellesley and Babson. We assumed that a student completing this program would need to meet all the degree requirements of the home institution, would need to satisfy ABET requirements – in particular, a year of appropriate mathematics and science, and a year and a half of engineering topics – and that the required engineering courses would include both the general engineering requirements at Olin (e.g., UOCD, POE) and the requirements specific to the Olin degree program

(ECE/ME/E). Based on our calculations, it appears that a Wellesley science or mathematics major would **easily** be able to complete requirements for both a Wellesley degree and an Olin engineering degree in five years. For Babson students, and for many other Wellesley majors (e.g., English), completing two degrees in five years would be possible, but would require more careful planning.¹

Unique Aspects of the Program

- “Try before you buy” – smooth transition and low risk as students can “taste” engineering via regular cross-registration opportunities, then complete the certificate program if interest prevails
- “Graduate with your class” – Many 3-2 (and 3-1-1) programs require students to break up their program in significant ways
- “Culture match” – Compared to many 3-2 options, this option involves campuses that are already co-operating, and that know each other
- “It’s easy” – Much easier for the student to do than other 3-2 options

Benefits

Balance of Trade – Currently balance of trade is uneven. Although this proposal is unlikely to result in an even balance of trade, it is at least a gesture in that direction.

Benefits to Partner Institutions – Both Wellesley and Babson would benefit from being able to offer this option to their students. There is existing interest from a small number of students on both campuses in such an opportunity; furthermore, both colleges could potentially use this program as a recruitment tool (this could be

¹ A typical Wellesley student graduates having completed between 32 and 38 courses; of these courses, typically 26 are used to satisfy degree requirements – and at least 3 of the 26 must be in mathematics and science. Thus, a typical Wellesley student has between 9 and 15 courses “available” that could be used to make progress towards an engineering degree. Similarly, the Babson degree requirements (126 credits) includes 12 credits of science and mathematics that could be used to count towards an engineering degree, as well as 28 credits of free electives. These “available” courses, combined with courses from an additional year of study, would be enough to construct a degree that meets the appropriate Olin degree requirements. We have created some rough example programs to see what this looks like for different majors; we intend to create more refined examples if it is appropriate to do so.

of particular interest to Wellesley, given the existence of the Smith engineering program).

Integration of Three Campuses – This program would potentially increase student traffic between the campuses and lead to better integration. It also might be a first step towards other dual degree or shared degree options.

Benefits to Olin – Participation by students from Wellesley and Babson may increase student diversity on many axes, including race, ethnicity, socioeconomic status, gender, backgrounds, and expertise. Balance of trade with partners is also a key benefit to Olin.

Risks

Enrollment in Olin Courses – Olin courses could become over-subscribed. But this is unlikely. In the “Exposure Analysis Preliminary Report” delivered to the Board in February of 2008 by the Intercollegiate Relations Committee (IRC) outlines the potential risks, including financial and curricular risks that the College is exposed to by relying heavily on our partner institutions for cross-registration opportunities for our students. The report focuses on the idea of trade imbalance, and as mentioned previously, a 4-1 program would be a step in the direction of trade balance.

While the committee has not been charged with performing the in-depth analysis of those risks, the report notes that from a report from 2006 made to the Dean of Faculty, the potential pool of Wellesley students that would have serious interest in an engineering degree is limited to approximately 114 graduates per year. The table below summarizes the data from 2006 for Wellesley.

Major	Average # of Graduates per year
Biology	25
Biological chemistry	15
Neuroscience	25
Physics	6
Computer Science	15
Mathematics	15
Chemistry	13
Total	114

Clearly, assuming any reasonable fraction of these students participating in cross-registration activities, it would be hard to envision overwhelming our faculty with dual-degree students. It would also not likely to be an instantaneous change in cross-registration activity, as a program such as this requires significant planning and coordination which should provide ample lead time for Olin to make appropriate staffing and curricular interventions. Finally, activities tied to a 4-1 program would be naturally spread across all four years of the curriculum and not be concentrated in any one group of faculty. The enrollment in a 4-1 program may therefore improve faculty loading by requiring upper level coursework rather than the more popular cross-registration targets, such as Design Nature, UOCD, and PoE.

Finally, per an email conversation of with Adele J. Wolfson, Associate Dean of Wellesley, the current Wellesley student participation in existing 3-2 programs is extremely limited. Approximately 5-6 students/year participate in the MA in International Economics and Finance program at Brandeis. No students have participated in the MIT 3-2 program in the past five years. No students have participated in any other dual-degree program in recent history. Therefore, it would be optimistic and unlikely that Olin would see an overwhelming response, especially at the outset of the program.

“Turf wars”, a.k.a. historical administrative politics – There is a long-standing dual-degree relationship between Wellesley and MIT. To propose an alternative program, especially in the area of Engineering, may have unpredictable political fallout for all parties involved.

The Task Force on the Sophomore and Junior Years
Final Report, July 17, 2007

Members

Brian Bingham, Mark L. Chang (chair), Jose Oscar Mur Miranda, Mark Somerville
Lynn Andrea Stein, Raymond Yim, Simon Helmore ('07), Joan Liu ('09)

Charge

The task force on the sophomore and junior years has been formed to investigate the current state and possible avenues for improvement of the sophomore and junior year curricula. To this end, we will:

Identify

- ... our goals as an institution for this portion of the curriculum
- ... the current state of the curriculum as it affects all “users and customers, internal and external”, e.g. our students, potential employers, graduate schools
- ... the current state of the curriculum with respect to the student experience
- ... opportunities for change, improvement, and innovation
- ... problem areas and tradeoffs necessary to fix them
- ... our position compared to other institutions through benchmarking (as time allows)

Produce

- ... a philosophical statement addressing the “Olin Brand” for the second and third year experience
- ... concrete examples of *what* we do well, *what* we need to fix, and *proof of concepts* of improved curriculum
- ... recommendations for next steps
- ... detailed documentation that may be used for processes such as ABET

We will complete this work by May 1, 2007, and submit reports and recommendations to the Academic Recommendation Board.

Forward

“The junior year needs some love.”

Despite the lack of clarity around what exactly needs love, and perhaps what “love” in this sense is, this statement (in part) led to the creation of an ARB task force to investigate possible avenues for improvement of the “space between”—years two and three—of the Olin curriculum. While our task force worked for a semester to lend some insight into this statement, we believe that unpacking the current curriculum and improving it is a much larger task that needs further study and wider audience. To this end, this report focuses on setting the groundwork for further initiatives in curricular reform, and provides some thematic avenues for improvement.

State of the Curriculum

“ Identify the current state of the curriculum as it affects all “users and customers, internal and external”, e.g. our students, potential employers, graduate schools. Identify the current state of the curriculum with respect to the student experience. ”

Beginning with external stakeholders such as potential employers and graduate schools, we note that we have relatively little data about the effect of the curriculum from their perspectives. What data we do have, mostly from solicited feedback through the Office of Postgraduate Planning, is positive. Internship surveys indicate very high levels of happiness with Olin students as employees, and to date, Olin students have been successful in pursuing both jobs and graduate school positions. This data does not provide us with much detail in our pursuit of improvement, and often only compares Olin students (favorably) with students from other colleges and universities. Additionally, the number of Olin alumni is very limited, and their experiences are very different thanks to a quickly changing curriculum during “the startup years”. Therefore it is still too early to understand the impact of the curriculum on the external successes of our students.

When surveying faculty, we hear varied concerns about the effectiveness of the curriculum. Grouping these comments into broader terms, they include the negative:

- Discipline “silos”
- Lack of technical depth
- Lack of technical breadth
- Lack of professionalism
- Inability to *finish* a project
- “Impedance mismatch” to the fourth year
- Not ready for SCOPE
- Doesn’t “feel” like Olin

...and the positive:

- Communication
- Teamwork
- Maturity and enthusiasm
- Intuition
- Robustness to bad courses

Let us look at some of these observations in more detail.

Discipline “silos”: this observation, and the corresponding “we’re not interdisciplinary” observation, come as departures from the first and fourth year experiences. Courses such as the ICBs, PoE, Design Nature and UOCD, all make strong attempts to explore interdisciplinary boundaries and blend content. SCOPE in the fourth year often brings design, engineering, budget and project management, as well as cross-discipline experiences together in a real-world engineering setting. The year 2/3 experience stands in stark contrast (with the possible exception of the Stolk/Martello History of Technology + Materials Science course) with very discipline-specific courses that “look like” more traditional programs at other colleges. As we will discuss in the next section, we do not offer interdisciplinary experiences in the year 2/3 courses, nor do students sample across discipline boundaries. This leads to siloed experiences for the students, and little cross-discipline pedagogical and research engagements for our faculty.

Lack of technical depth/breadth: these observations are a commonly sighted problem area among both students and faculty. Looking closer, some of these feelings come from curricular comparisons to other colleges and universities. The evidence supports the claim of lack of technical depth and breadth—our degree programs often have (far) fewer degree course requirements when compared to peer institutions. In some respects, our degree requirements provide only the bare minimum of introductory material. Providing courses that are only introductory in nature allows our students to maintain a degree of flexibility in the courses they select. A side effect to the thin prerequisite structure that is currently in place is that faculty can rarely count on learned knowledge in a prior class as a foundation and often find themselves re-teaching material and being less aggressive with the depth of material covered. We also struggle to provide discipline breadth—we cannot provide significantly more courses without hiring more faculty in broader areas of engineering and accepting course enrollments in electives to drop to very low numbers. With our current curriculum (and approach to curriculum development) both observations have merit. The consequences of lacking technical breadth and depth also play a major role in the perceived preparedness of our students for their senior year activities, in particular, SCOPE.

“Impedance mismatch” to the fourth year: this statement is a generalization of many comments regarding the transition between being a 3rd year student and being a 4th year student at Olin. One oft-cited example is the difference in structure between the 3rd and 4th years. In the 3rd year, most students spend the majority of their time in the Academic Center taking classes that meet at regularly scheduled times. But in the 4th year, much of their time is spent self-scheduled and on large, open-ended projects. They have to contend with SCOPE, OSS, and AHS Capstone, all of which are significantly self-directed and self-scheduled. The “impedance mismatch” can come from the learning objectives set out and achieved in the 3rd year (qualitative analysis, quantitative analysis, discipline-specific knowledge) versus the needs of the 4th year (teamwork, communication, budgeting, long-term project management, time management, life-long learning).

Not ready for SCOPE: as a natural follow on from the previous observation about the “impedance mismatch” between the 3rd and 4th years, SCOPE faculty have commented that our students are often not prepared for the SCOPE experience, both from a technical perspective as well as a maturity perspective. From a technical perspective, our students might lack the aforementioned technical depth/breadth necessary to complete the kinds of technically challenging problems presented in SCOPE. The observation that students do not have many experiences *finishing* projects prior to SCOPE is another component to not being ready for a pre-professional engagement like SCOPE. Professionally speaking, our students get little formal training in project and budget management, and no training dealing with paying clients and management. In contrast, the Harvey Mudd Clinic program requires that students take a course in project management and participate on a clinic team in their junior year to gain exposure to the needs of a large scale project.

Doesn't "feel" like Olin: perhaps the most ambiguous observation, yet the one that captures the feeling of opportunities for improvement. Many faculty have commented that the year 2/3 experience just doesn't feel as innovative as many other parts of the curriculum. For instance, looking at the titles of the courses offered in the year 2/3 curriculum finds courses that you would find at any other institution. Contrast that with the first year program of integrated course blocks, and the senior year capstone experiences, and the year 2/3 courses look very staid. Speaking with faculty, we got the feeling that many of the year 2/3 courses are content driven in order to fill out specific discipline requirements rather than part of a larger thematic picture of the major as a whole. The selection of courses offered in the year 2/3 curriculum also comes, in part, as a result of ABET accreditation needs and the subsequent urge to put forth a low(er)-risk curriculum. Finally, the amount of time and energy invested into the year 2/3 curriculum might be less than the 1st and 4th years. While it is constantly improving, faculty generally agreed that there was opportunity for improvement to make it more unique and innovate—more “Olin”.

On the positive side, many faculty have shown great appreciation of our students' abilities to communicate effectively (especially in oral presentations and to external constituencies). Our students also exhibit high levels of maturity and enthusiasm for learning and teamwork, often citing the “culture of learning” at Olin as a motivating force for excellence. Intuition was also seen as a positive trait of our students—often times exceeding our own high expectations in pursuing a solution or design. Finally, many of our students have endured less than ideal courses. Courses being taught for the first time, using experimental methodologies, by adjunct faculty, with limited and unprepared resources, have all been endured by our students while maintaining a high level of constructive feedback and an understanding attitude toward the development of curricula.

Note some contradictions between positive and negative perspectives, as, in general, there is little consensus from faculty about the exact nature of the problem, or even if there is a problem. The most focused statement we can draw from our discussions with faculty is that there is a general feeling of opportunity for improvement (“we could do better”), but each individual has their own idea of what “needs fixing”. Add to this the fact that year 2/3 is generally seen as the time to build program-specific

knowledge, most faculty recognize that improvements would be hard to come by in light of degree requirement constraints. Coupled with each faculty member's own unavoidably narrow views of their portion of the curriculum, most problem areas and possible solutions identified by individual faculty may be myopic.

A final stakeholder in the curriculum is the student body. The task force interviewed a small number of students (about a dozen) representing all program groups and years to ascertain the student experience in year 2/3 courses. Their collective perspectives were slightly more focused than the faculty, and the following consistent themes emerged in our discussions:

- Foundation courses feel “disconnected” from discipline courses
- There is a culture of Math separate from Engineering
- Lack of meaningful projects (“tacked on”)
- Projects are done when I give up or run out of time
- Don’t know why we’re doing projects other than “we should”
- Not multidisciplinary
- Life-long-learning and self-scheduling are lacking entering the senior year
- Too many problem sets

These themes followed closely along discipline lines. Adding more detail (**bold**):

- Foundation courses feel “disconnected” from discipline courses (**ME & ECE**)
- There is a culture of Math separate from Engineering (**all**)
- Lack of meaningful projects (“tacked on”) (**ME**)
- Projects are done when I give up or run out of time (**all**)
- Don’t know why we’re doing projects other than “we should” (**ME**)
- Not multidisciplinary (**ME & ECE**)
- Life-long-learning and self-scheduling are lacking entering the senior year (**all**)
- Too many problem sets (**ME**)

Projects and workload: in interviewing regarding the year 2/3 experience, much of the conversation was spent discussing the nature of the coursework. It became more obvious that different majors had different experiences. ME students expressed a feeling of “too many problem sets” and stronger sense of project dissatisfaction as compared to ECE students. A widely held perspective among students could be summarized as, “ECE students do projects, ME students do problem sets”. In fact, some students use this observation in choosing between the two majors.

Many students also expressed a significant feeling of disconnect between the foundation courses and the year 2/3 courses. For example, for ECEs, courses such as Biology or Materials Science did not play any major role in their major. Students talked of “getting them out of the way” in order to move forward into more discipline-specific courses. Noteworthy are the E program students (in particular, E:Bio) who expressed a much stronger sense of discipline need for foundation courses. Math very clearly fell the disconnected category for most students. Students felt that the Math curriculum was not well connected to any discipline-specific courses, or even utilized effectively at all outside the math courses. For example,

rigorous analytical skills learned in math courses were often not used in favor of easier-to-use computer simulation and modeling techniques. Another example from SCOPE was the reluctance of students to perform rigorous analysis and often wanting instead to build and experiment to find solutions.

Discipline “silos”: the observations made by faculty were well reflected in the students. Very few students discussed having significant multidisciplinary experiences during year 2/3. However, not all students expressed negative feelings toward the “silos”. Rather, they embraced their discipline-specific courses in a positive way. Noteworthy exceptions are E-program students who did not feel the discipline “silos” as acutely as they are often naturally multidisciplinary.

Discontinuities entering SCOPE and the senior year: like the faculty observations, students also spoke of hurdles when starting their senior year. Like the faculty, they observed a steep learning curve in terms of self-scheduling and time management of their workloads. With almost entirely self-scheduled endeavors, students often found themselves working for the next deliverable. Those activities with less structure, then, received less attention. Students cited OSS as often falling victim to other, more pressing deadlines. Some students, however, felt a positive attitude toward having such extensive control of their time. Students also noted lack of training in long-term projects, project management, and budgeting as barriers to success in SCOPE.

Offering More Courses

From these observations we can begin to piece together avenues for improvement. One obvious avenue for improvement would be to offer more courses. Offering more courses might tackle problems such as discipline “silos” (by offering more multidisciplinary classes), technical depth and breadth (by offering more advanced and/or a broader variety of discipline-specific courses), and the “impedance mismatch” to the 4th year (by offering courses in project management, requiring juniors to participate in SCOPE, requiring additional AHS/E! credit in preparation for the humanities capstone). But can we offer more “specialization” courses?

Available Faculty-Credit Hours

One clear resource limit is the number of faculty we have. The Dean has previously calculated that we have the equivalent of approximately 28-30 “real” FTEs. This number is lower than the official number of faculty (which is on the order of 35), but if anything, is likely optimistic when taking developmental leaves into account. Given 30 FTEs, and an assumed average faculty load of 14.5 credit hours per year (which is consistent with current average loads), **Olin is able to offer approximately 440 faculty-credit hours per year, or about 110 unique sections per year.**

Section Size Limitations

From a student:faculty ratio perspective, this number seems reasonable. For example, if we assume 75 students in each entering class, taking an average of 28

credits on the Olin campus per year, Olin must deliver a total of 8400 student-credit hours each year, or 2100 student-sections per year. Given this number of students, the number of faculty, and the assumed faculty teaching load, **Olin should have an average section size of about 20 students.**

Constraints Imposed By Degree Requirements

Assuming that we maintain a section size on the order of 20 (i.e., we do not want to create large first year classes in order to free resources for junior year classes), we are limited by resources and the structure of degree requirements with respect to how many courses we can offer each year within a major. For example, the current degree requirements include 1.5 years of engineering, 1 year of math and science, and about 1 year of AHS/E!. These requirements, combined with reasonable assumptions about off-campus enrollment, imply that, unless we are going to devote resources in an inequitable manner, that **about 50% of our offered courses should be in ENGR, 30% in MTH/SCI, and 20% in AHS/E!** To date this has been borne out – for example, in the 2006-2007 academic year, Olin offered a total of about 55 sections of ENGR, 32 of MTH/SCI, and 21 of AHS/E!. Figure 1 depicts the current distribution of class offerings.

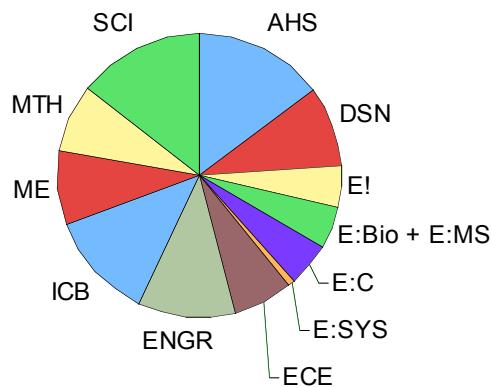


Figure 1: Approximate distribution, by # of sections offered, of classes for 2006-2007 academic year.

A Limited Number of Major-Specific Courses

Within the ENGR courses, somewhat less than half are general degree requirements, as opposed to program-specific courses. General degree requirements impose 6 sections of MC/ECS/EDS, 3 sections of POE, 3 sections of UOCD, about 4-5 sections of design depth, and about 8-9 sections of SCOPE. **Accordingly, less than 30 sections per year can be devoted to program specific courses.**

Currently, these program specific courses are divided fairly evenly between the three programs. For example, in 2006-7, Olin offered approximately 10 sections of courses that were specifically appropriate for E concentrations (E:Bio, E:MS, E:C, E:Sys), 7.5 sections of ECE-specific courses (note that E:C courses also serve ECE majors), and

9 sections of ME-specific courses. Some of these courses are sufficiently popular that multiple sections are required – e.g., Software Design is typically offered every semester, as is Mechanical Design. **As a consequence, each major at Olin can afford to offer about 8 unique courses a year.** This conclusion is supported by previous offerings. Note that of these 8 unique courses, five are “core” within ECE and ME. Thus, **within the ME and ECE programs, it is possible to offer 1-2 electives each semester. Within E, virtually all courses are “core” to one of the concentrations.** Finally, within E, many courses are already taught on a staggered schedule in order to maximize resources.

Specialization Silos

Finally, it is worth noting that currently there is relatively little cross-major “mingling”, with the exception of E:Sys students, who, by their degree requirements, must take courses from all three programs. As Figure 2 indicates:

- The typical ME student takes no ECE courses, and one E course;
- The typical ECE student takes no ME courses, and one E course;
- The typical E student (excepting E:Sys) takes no ECE courses, and perhaps $\frac{1}{2}$ of an ME course.

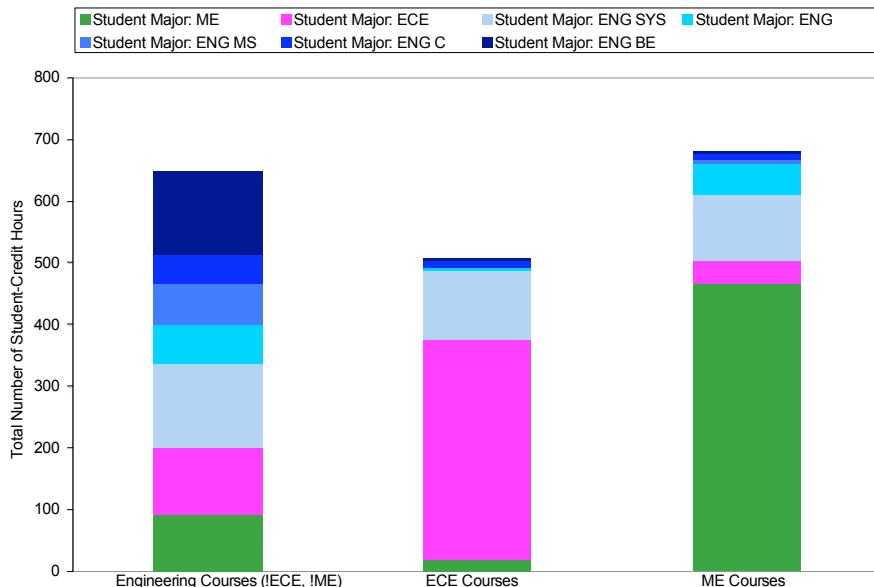


Figure 2: Enrollment, by major, in major-specific courses.

Our Challenge

Currently we are able to offer a very limited palette of courses within each major. Within ME and ECE, we can offer 2-3 discipline specific electives each year; within E, we are able to offer “just enough” to allow students to satisfy concentration requirements. Furthermore, students are not currently taking specialization classes outside of their majors. As a consequence,

- Students have very little flexibility within their majors – “electives” are often in effect required.

- We currently do not have the necessary resources to offer additional electives, unless we choose to reallocate resources from elsewhere in the curriculum.
- While individual courses may emphasize an interdisciplinary approach, students are choosing to be highly disciplinary in their enrollment patterns.

In effect, without obtaining more resources, the following figure illustrates our current state—each major can offer approximately 8 unique courses per year. The challenge is then to improve the curriculum without increasing resources.

ECE	E	ME

Curricular Goals for Year 2/3

“ Identify our goals as an institution for this portion of the curriculum. ”

In our discussions it became clear that as a faculty we lack clarity regarding our goals for this portion of the curriculum, and indeed for most portions of the curriculum. Perhaps it is a result of the “just in time” nature of our curriculum roll-out. Perhaps it is a positive thing. Perhaps it is a negative. Our task force chose to look back at the College’s Mission Statement for guidance (emphasis added).

Olin College prepares future **leaders** through an **innovative** engineering education that **bridges science and technology, enterprise, and society**. Skilled in **independent learning** and the art of **design**, our graduates will seek **opportunities** and take initiative to make a positive difference in the world.

Given the mission, we propose that the curriculum as whole should...

- Provide students with leadership and entrepreneurial opportunities
- Be pedagogically innovative
- Provided students with opportunities to learn engineering in an interdisciplinary mode, bridging science and technology, enterprise, and society
- Provide students with opportunities to develop their ability to learn independently
- Provide students with opportunities to practice the “art of design”

We believe that these guidelines for the curriculum as a whole should apply in approximately equal measure to all four years; however, we recognize that different

portions of the curriculum have different goals in addition to these curriculum-wide goals:

- The first 1.5 years of the curriculum serves to provide students with a range of common foundational experiences. Because we value student flexibility, we have explicitly designed a curriculum in which students do not commit to a degree program until relatively late. Thus, the first 1.5 years also serves to expose students to multiple perspectives before they choose a degree program.
- Given that the first 1.5 years is nominally common to all students, years 2-4 must provide students with the opportunity to learn the technical material necessary to prepare themselves for whatever their “next step” will be—graduate school, work in industry, etc.
- The final year of the curriculum also has a capstone goal—SCOPE (and potentially the AHS and E! capstones) are billed as culminating experiences, in which students take a project from conception to fruition, and in which students work with professional orientation. It is not clear whether we intend SCOPE to be a culminating learning experience (in which case we might define particular learning objectives for SCOPE—e.g., project management), or a first professional experience (in which case we would hope that students will have already acquired the skills they need to be successful in SCOPE).

With these guiding principles we can begin to craft avenues for improvement.

Themes for Improvement

“ Identify problem areas and tradeoffs necessary to fix them.
Identify opportunities for change, improvement, and innovation.”

From our task force discussions, and from discussions with faculty and students, we have created four “themes” for improvement:

- | | |
|--------------------------------------|-------------------------------------|
| • Projects and Authentic Experiences | • Flexibility / Away Program |
| • Interdisciplinary Experiences | • Analysis / Math & Technical Depth |

Recommendation: Each discipline should develop a philosophical framework (core objectives) for year 2/3 and build courses to serve the framework.

Themes addressed: Projects and Authentic Experiences, Interdisciplinary Experiences, Analysis / Math & Technical Depth

Cost: Time and effort of faculty. Possible large scale restructuring of course content.
Accreditation risk.

Reasoning: One common complaint about the year 2/3 experience is that it doesn’t “feel like Olin”. One component to that is that as a faculty, we have not had the time and energy to examine the courses that occupy the year 2/3 space. A second is the fact that ABET accreditation did not allow us to develop a more innovative (and perhaps riskier) curriculum. The first year curriculum does not suffer as much from these problems as it has had the

longest time under development. It also does not have the same programmatic ABET dependencies as it is much less discipline specific.

By taking time to develop a philosophical framework we can focus on what it is, at the core, each of the disciplines *is* at Olin. This helps us avoid the “Topic X” syndrome—where we teach a topic because it either traditionally appears in a particular discipline, or is something that is a core competency of a faculty member. It makes for more coherent disciplines by forcing faculty to (perhaps) agree on what, for instance, being an ECE major at Olin means. It may make more intelligent use of course space—a scarce resource—by allowing each program group to pare away unneeded material and build curriculum that has fewer overlaps in material. Finally, with more clarity about the core objectives of a discipline, it may make it easier to identify opportunities for multidisciplinary courses.

In effect, we recommend allowing faculty the time to *design* the year 2/3 experience.

Recommendation: Improve clarity in terms of project goals through the creation and use of project experience axes.

Themes addressed: Projects and Authentic Experiences

Cost: Time and effort of faculty.

Reasoning: Many of the observations from both students and faculty centered upon the work in the class. Much of this work at Olin is project based. In light of student concerns about project appropriateness and design, impedance matching to SCOPE, and discontinuities in learning objectives between the 3rd and 4th years, we propose faculty be more explicit in describing the learning objectives for projects. This may help students understand the larger pedagogical goals for the project, and may help faculty focus projects better rather than doing projects because “we have to”. Additionally, coupled with the aforementioned discipline framework, program groups can examine these project learning axes to determine if an appropriate level of coverage is given to all axes as students transition into their senior year.

An example of possible project “learning axes” is shown below.

- Learning project management skills
 - Self-directed (life-long) learning
 - Professional experience (executing, finishing or deciding when done)
 - Communication (presenting results)
 - Content Synthesis (connections between more than one topic)
 - Knowledge content (deliver, motivate, reinforce or expand on faculty-delivered content)
 - Skill content (delivery, motivation or reinforce pertinent skills)
 - Identify tools, techniques or process (problem solving attitude)
 - Fun (Motivation and intrinsic motivation)
-

Recommendation: Introduce a 3rd year required project course.

Themes addressed: Projects and Authentic Experiences, Analysis / Math & Technical Depth

Cost: Resources: faculty time, money, administrative overhead. May be difficult to find many “authentic” project experiences.

Reasoning: As students enter their senior year, there are many hurdles to success in SCOPE. Scheduling, problem solving, project management, time management, and large-scale project completion are just some of the obstacles. How one *approaches* a large problem is also something that gets little exposure in our project courses. Requiring all students to participate in a 3rd year project course would give students and faculty an opportunity to explore these principles in a low-risk environment. This type of course is required of all Harvey Mudd students to prepare them for their senior capstone. There, they are paired with actual capstone projects—serving a limited role—but getting valuable exposure and experience along similar axes.

Recommendation: Introduce a “plan-based” course of study option for each major, similar to the E:Self program.

Themes addressed: Interdisciplinary Experiences, Flexibility / Away Program

Cost: Advising time and effort, ABET risks, lack of courses to choose from

Reasoning: In an effort to break down the “silo” walls between disciplines, we propose a plan-based course of study for each major. Each student would still obtain a major degree (ECE/ME/E), but will have the option of creating an approved plan to obtain their degree that may or may not include all of the traditional (and existing) core requirements. For instance, an ECE student may opt not to take Communications, and instead elects to take Controls. For the right set of courses, this may be an acceptable tradeoff that still permits the student to have sufficient ECE content, while gaining multidisciplinary exposure. This recommendation improves student flexibility, facilitates multidisciplinary experiences, and encourages student ownership of their degree. The Away Program also becomes a more viable option for students that elect to take a more self-directed approach to their degree, as it may free up slots in that vital junior year.

Recommendation: Require a semester away, leading to an expected 9-semester degree.

Themes addressed: Flexibility / Away Program

Cost: Cost, possible disruption to the student’s learning path, time and effort to increase destination options

Reasoning: The College's Mission statement is clear in its support of students that are capable of engaging engineering for societal change. Exposing students to the world through encouraging the Away experience is a vital part of creating the "Renaissance Engineer". By requiring that students participate in the Away program, we will be giving our students an opportunity that they might not select for themselves.

Next Steps

The task of overhauling two years of curriculum is one that cannot be accomplished in a semester's time. The goal of this task force was to determine state of the curriculum as it applies to all parties, and start exploring practical avenues for improvement. The general feeling of opportunity is one that we encountered in almost all people we spoke to, and we hope that the recommendations we've made here will be taken forward, refined, and acted upon.

Perhaps the biggest constraint is our inability, without further resources, to expand the number of classes offered within each major. This constraint has led to, we believe, inefficient use of these spaces, and unnecessary division along discipline lines. With each recommendation, it is our intention that the ARB find appropriate groups and seed them with appropriate recommendations for further study. Hopefully, examining the true goals of each discipline and coming to an agreed framework will soften the divisions and allow the faculty to create a more coherent picture of the Olin year 2/3 experience.

9 List of Potential Reviewers

9.1 Scott Hauck

Scott Hauck is a Professor of Electrical Engineering at the University of Washington. Prof. Hauck was my PhD and Master's adviser. I include him here as he has a strong interest in undergraduate education, understands Olin, and is an expert in the field of reconfigurable computing.

Biosketch

He is a Professor at the University of Washington's Department of Electrical Engineering, in the Embedded Systems and VLSI group, and an Adjunct in the Department of Computer Science and Engineering. His work is focused around FPGAs, chips that can be programmed and reprogrammed to implement complex digital logic. His interests are the application of FPGA technology to situations that can make use of their novel features: high-performance computing, reconfigurable subsystems for system-on-a-chip, computer architecture education, hyperspectral image compression, and other areas. His work encompasses VLSI, CAD, and applications of these devices. He is editor (with Andre' DeHon) of a book on reconfigurable computing: Scott Hauck, Andre' DeHon (editors), "Reconfigurable Computing: The Theory and Practice of FPGA-Based Computation", Morgan Kaufmann/Elsevier, 2008.

9.2 Miriam Leeser

Miriam Leeser is Professor of Electrical and Computer Engineering at Northeastern University. Prof. Leeser is an expert in reconfigurable computing and would be able to evaluate my research in that area. I have known Prof. Leeser since I was a graduate student, and we have worked, in the past, in similar research areas. We have not collaborated on work, however, I invited her to give a seminar talk at Olin in 2005.

Biosketch

Miriam Leeser joined the faculty at Northeastern University in January 1996. Her specialty is Computer Engineering. She received a BS in Electrical Engineering from Cornell University, and Diploma and PhD degrees in Computer Science from Cambridge University in England. She joined the faculty of Cornell University's Department of Electrical Engineering in 1988. She received a National Science Foundation Young Investigator Award in 1992.

Her research interests are in the areas of design and design tools for field programmable logic, and high level design tools for digital systems.

9.3 Seth Teller

Seth Teller is a Professor Electrical Engineering and Computer Science, MIT. Prof. Teller is an expert in robotics and has published papers on localization using similar methods to those that I have published. I have no working relationship with Prof. Teller, and know him only by his work.

Biosketch

Seth Teller obtained a Ph.D. from U.C. Berkeley in 1992, focusing on accelerated rendering of complex architectural environments. After post-doctoral fellowships at the Computer Science Institute of the Hebrew University of Jerusalem Institute of Computer Science, and Princeton University's Computer Science Department, Teller joined MIT's Department of Electrical Engineering and Computer Science, Lab for Computer Science, and Artificial Intelligence Lab in 1994. (In 2004, the two labs merged into CSAIL, MIT's Computer Science and Artificial Intelligence Laboratory.)

At CSAIL, Teller heads the Robotics, Vision, and Sensor Networks group (RVSN), where his research focuses on enabling machines to become aware of their surroundings and interact naturally with people. Some of his lab's recent projects include: hand-held and body-worn devices that provide navigation assistance indoors; a voice-commandable robotic wheelchair; a self-driving LandRover; and an unmanned, outdoor forklift commanded through speech and gestures.

9.4 Dr. Jeffrey Hightower

Dr. Hightower is a Senior Scientist and Engineering Manager, Intel Labs Seattle. Dr. Hightower is an expert in localization and would be able to comment on my work in this area. I have spoken with Dr. Hightower once, when visiting Intel Labs in Seattle during graduate school. Otherwise, I have no working relationship with Dr. Hightower.

Biosketch

Jeffrey is a Senior Research Scientist and the Engineering Manager at Intel Labs Seattle, part of the exploratory research arm of Intel. His research interests include sensors, machine learning, location technology, and ubiquitous computing to allow computing to fade calmly into the background of everyday life. He is currently part of the Everyday Sensing and Perception project seeking to enable accurate computer understanding of everyday situations. He previously co-led the Place Lab project. He has a BS in Computer Science from the University of Colorado and MS and PhD degrees in Computer Science and Engineering from the University of Washington. When not doing research, Jeffrey is a professional ski instructor, whitewater rafter, swiftwater rescue technician, and amateur woodworker.

9.5 Lukas Kencl

Dr. Kencl is the Director of the Research and Development Centre for Mobile Applications (RDC), a joint research center between Vodafone, IBM, Ericsson, and the Czech Technical University in Prague. Dr. Kencl is an expert in wireless and wireless localization technologies and would be able to evaluate my work in this area. I am working with Dr. Kencl as co-chair of a workshop in indoor localization, Mobile Entity Localization and Tracking (MELT) 2010.

Biosketch

Lukas Kencl obtained a Ph.D. degree in Communication Networks from Ecole Polytechnique Federale de Lausanne (EPFL), Switzerland, in 2003, and a MSc. degree in Computer Science from Charles University, Prague, in 1995. Since May 2007 he is RDC Director. Previously, he was a Senior Researcher at Intel Research, Cambridge, UK (2003-6) and a Pre-Doc at the IBM Zurich Research Laboratory (1999-2003). His research focuses on novel interfaces, services and applications in mobile wireless networks and on architecture and performance optimization of networking systems. He regularly publishes at established networking conferences and journals, is a co-inventor of multiple network technology patents and holder of several industrial grants (Vodafone, IBM).

9.6 Adele Wolfson

Adele Wolfson is the Nan Walsh Schow '54 and Howard B. Schow Professor in the Physical and Natural Sciences and Professor of Chemistry at Wellesley College. I have worked with Prof. Wolfson over the years on many intercollegiate relations issues. She would be able to comment on my service to Olin College and the building of intercollegiate relations with Wellesley.

Biosketch

Adele J. Wolfson is associate dean of the college at Wellesley College, where she holds the Nan Walsh Schow '54 and Howard B. Schow Professorship in the Natural and Physical Sciences, and is professor of chemistry. In her role as associate dean, she has partnered with the dean of students to create programs that support the academic success of all students and to raise awareness of stereotype threat. Wolfson chairs the colleges curriculum committee, which, under her leadership, has raised academic standards and encouraged departments self-evaluation.

At Wellesley, Wolfson was faculty director of the Science Center, where she brought faculty from all departments and programs together to promote the sciences, and served as program director for the colleges award from the Howard Hughes Medical Institute Undergraduate Science Program. She has also been faculty

director of the Pforzheimer Learning and Teaching Center, where she implemented programs for new and continuing faculty.

Wolfson received an A.B. in chemistry from Brandeis University, studied at Hebrew University in Jerusalem and received a Ph.D. in biochemistry from Columbia University. She has held post-doctoral positions in Paris and Melbourne.

She is particularly interested in science education, innovative pedagogy and the participation of women and underrepresented minorities in science. Along with a Wellesley colleague, she created an integrated cluster for introductory chemistry and biology. She has served as a consultant to Project Kaleidoscope, on the committee of examiners for the GRE, the editorial board of Biochemistry and Molecular Biology Education and the advisory board to the NSF-ADVANCE project at the University of Maryland Baltimore County. Wolfson is principal investigator on a grant from the Teagle Foundation to the American Society for Biochemistry and Molecular Biology aimed at assessing the relationship between the goals of the undergraduate major and a liberal education.