Steven D. Johnson

Professor of Computer Science

Indiana University School of Informatics

812-855-2567
sjohnson@cs.indiana.edu
http://www.cs.indiana.edu/~sjohnson
Lindley Hall Rm 215, 150 S Woodlawn Ave., Bloomington IN 47405-7104

Biographical Sketch

Steven D. Johnson is a Professor of Computer Science in the Indiana University School of Informatics and Computing and Chairs Division B in the School. He received the B.A. in mathematics and Russian from Depauw University in 1970, the M.A. in mathematics from Indiana University in 1977, and the Ph.D. in computer science from Indiana University in 1983. His Ph.D. dissertation, *Synthesis of Digital Designs from Recursion Equations*, was an ACM Distinguished Dissertation in 1984. His research centers on formal methods applied to system design where he is currently working on formal synthesis of embedded systems. In addition to his primary research, he is an affiliated faculty member of Indiana University's Logic, Cognitive Science Robotics, and Security Informatics groups. Prof. Johnson has served on numerous conference program committees, including the IEEE International Conference on Computer Design (ICCD), the IFIP Conferences on Computer Hardware Description Languages and their Applications (CHDL, which he chaired in 1995), the Advanced Research Working Conference on Correct Hardware Design and Verification Methods (CHARME), the International Conference on Formal Methods in Computer Aided Design (FMCAD, which he co-chaired in 2000), among many others. Prof. Johnson was born October 11, 1948. He is married with two children

Research interests

Formal methods. System synthesis and verification. Embedded, real-time and safety-critical systems. Vehicular robotics. Parallel symbolic processing.

Education

Ph.D	1983	Indiana University (Computer Science)
M.S.	1976	Indiana University (Computer Science)
M.A.	1972	Indiana University (Mathematics)
B.A.	1970	DePauw University (Mathematics and Russian)

Experience

2010– now	Chair, Division B, Indiana University School of Informatics & Computing
2001– now	Professor of Computer Science, Indiana University
2004	Staff, Midwest Proton Therapy Institute, Bloomington, IN, May 12–Aug 30
2003	Visiting Scientist, National Aeronautics Institute, Hampton VA, July 1–21
1996	Visiting Scholar, ECE/CS Department, University of Cincinnati
1993–1995	Chair, Computer Science Department, Indiana University
1990–2001	Associate Professor of Computer Science, Indiana University
1984–1990	Assistant Professor of Computer Science, Indiana University
1983–1984	Visiting Assistant Professor of Computer Science, Indiana University
1982–1983	Visiting Lecturer, Department of Computer Science, Indiana University
1977–1979	Member of Technical Staff, Bell Telephone Laboratories, Holmdel, NJ

Refereed and Invited Publications

46. Steven D. Johnson, Bryce Himebaugh, and Scott A. Dial. Homogeneous resource configuration and access for an autonomous robotic vehicle. *SAE Int. J. Commer. Veh*, 1(1):534–543, 2008. Invited paper.

45. Venkatesh Choppella, Arijit Sengupta, Edward L. Robertson, and Steven D. Johnson. Preliminary explorations in specifying and validating entity-relationship models in PVS. In John Rushby and Natarajan Shankar, editors, *AFM '07: Proceedings of the second workshop on Automated formal methods*, New York, NY, USA, 2007. ACM. http://portal.acm.org/citation.cfm?id=1345169#

44. Alex Tsow and Steven D. Johnson. Data refinement for synchronous system specification and construction. In Dominique Borrione and Wolfgang Paul, editors, *Correct Hardware Design and Verification Methods (CHARME 2005)*, volume 3725 of *Lecture Notes in Computer Science*, pages 398–401, Berlin, October 2005. Springer-Verlag. 13th IFIP WG 10.5 Advanced Research Working Conference.

43. Lee Pike and Steven D. Johnson. The formal verification of a reintegration protocol. In *EMSOFT '05: Proceedings of the 5th ACM international conference on Embedded software*, pages 286–289, New York, NY, USA, 2005. ACM Press. http://www.cs.indiana.edu/~lepike/pub_pages/emsoft.html

42. Gary B. Parker, Matt Parker, and Steven D. Johnson. evolving autonomous agent control in the Xpilot environment. In *2005 IEEE Congress on Evolutionary Computation (CEC 2005)*, pages 2416–2421. IEEE Press, September 2005.

41. Steven D. Johnson. Formal methods in embedded design. *Computer*, 36:104–106, November 2003. http://www2.computer.org/portal/web/csdl/doi/10.1109/MC.2003.1244539

40. Danko D. Antolovic, Bryce Himebaugh, and Steven D. Johnson. Skeyeball: Real-time vision system for an autonomous model airplane. In *Proceedings of the 22nd Digital Avionics Systems Conference (DASC'03)*, October 2003. Indianapolis, Indiana. http://www.cs.indiana.edu/~sjohnson/papers/dasc03/DASC03.pdf

39. Danko Antolovic, Bryce Himebaugh, and Steven D. Johnson. The Sk^{e}y^{e}ball tracking project. 16th Florida Conference on Recent Advances in Robotics (FCRAR'2003), http://www.eng.fau.edu/conf/fcrar2003/i, May 2003. 8–9 May 2003, Florida Atlantic University Seatech campus, Dania Beach, Florida. http://www.cs.indiana.edu/~sjohnson/pub/fcrar03/fcrar03.pdf

38. Steven D. Johnson. View from the fringe of the fringe. In Tiziana Margaria and Thomas Melham, editors, *11th Advanced Research Working Conference on Correct Hardware Design and Verification Methods, CHARME 2001, Livingston, Scotland, Proceedings*, volume 2144 of *Lecture Notes in Computer Science*, pages 1–12. Springer-Verlag, 2001. Invited paper. http://link.springer.de/search.htm

37. Steven D. Johnson, Yanhoung A. Liu, and Yuchen Zhang. A systematic incrementalization technique and its application to hardware design. *International Journal on Software Tools for Technology Transfer*, 4:211–223, February 2003. http://link.springer.de/search.htm

36. Alex Tsow and Steven D. Johnson. Visualizing system factorizations with behavior tables. In Warren A. Hunt, Jr. and Steven D. Johnson, editors, *Formal Methods in Computer-Aided Design, Third International Conference, FMCAD 2000, Austin, TX, USA, November 1-3, 2000, Proceedings*, volume 1954 of *Lecture Notes in Computer Science*, pages 523–541, Heidelberg Berlin, 2000. Springer-Verlag. http://link.springer.de/link/service/journals/10009/first/tfirst.htm

35. Steven D. Johnson and Alex Tsow. Algebra of behavior tables. In C. M. Holloway, editor, *Lfm2000: Fifth NASA Langley Formal Methods Workshop*, pages 23–34, 2000. NASA Conference Publication NASA/CP-2000-210100. Proceedings of the 5th NASA Langley Formal Methods Workshop, Williamsburg, Virginia, 13-15 June, 2000. http://shemesh.larc.nasa.gov/fm/Lfm2000/

34. Steven D. Johnson, Yanhong A. Liu, and Yuchen Zhang. A systematic incrementalization technique and its application to hardware design. In L. Pierre and T. Kropf, editors, *Correct Hardware Design and Verification Methods (CHARME'99)*, volume 1703 of *Lecture Notes in Computer Science*, pages 334–337,

Berlin, 1999. Springer. 10th IFIP WG 10.5 Advanced Research Working Conference, CHARME '99, Bad Herrenalb, Germany, September 1999, Proceedings. http://link.springer.de/search.htm

33. Steven D. Johnson. A workshop on formal methods education: an aggregation of opinions. *International Journal on Software Tools for Technology Transfer*, 2(3):203–207, November 1999. http://link.springer.de/search.htm

32. Steven D. Johnson, Warren P. Alexander, Shiu-Kai Chin, and Ganesh Gopalakrishnan. Report on the 21st century engineering consortium workshop, March 1999. Report of the meeting held March 1998 at Melbourne, Florida. http://www.cs.indiana.edu/formal-methods-education/xxiec/report.html

31. Steven D. Johnson and Paul S. Miner. Integrated reasoning support in system design: design derivation and theorem proving. In Hon F. Li and David K. Probst, editors, *Advances in Hardware Design and Verification (CHARME'97)*, pages 255–272. Chapman-Hall, 1997. IFIP TC10 WG 10.5 International Conference on Correct Hardware and Verification Methods, 16-18 October 1997, Montreal, Canada.

30. Steven D. Johnson. A tabular language for system design. In C. Michael Holloway and Kelly J. Hayhurst, editors, *Lfm97: Fourth NASA Langley Formal Methods Workshop*, sep 1997. NASA Conference Publication 3356, Proceedings of the 4th NASA Langley Formal Methods Workshop, Hampton, Virginia 10-12 September, 1997. http://archive.larc.nasa.gov/shemesh/Lfm97/

29. Paul S. Miner and Steven D. Johnson. Verification of an optimized fault-tolerant clock synchronization circuit. In Mary Sheeran and Satnam Singh, editors, *Designing Correct Circuits*, Electronic Workshops in Computing,. Springer, September 1996. I cannot locate a link to this Springer series. http://www.cs.indiana.edu/~sjohnson/papers/Miner-Johnson-96.pdf

28. Steven D. Johnson, Gerard Allwein, and K. Jon Barwise. Toward the rigorous use of diagrams in reasoning about hardware. In Gerard Allwein and Jon Barwise, editors, *Logical Reasoning with Diagrams*. Oxford University Press, 1996.

27. Kathryn Fisler and Steven D. Johnson. Integrating design and verification envrionments through a logic supprting hardware diagrams. In *Procedings of the 1995 IFIP International Conference on Computer Hardware Description Languages and Their Applications*, pages 669–674. IEEE Cat. No. 95TH8102, September 1995. http://www.cs.indiana.edu/~sjohnson/papers/Fisler-Johnson-95.pdf

26. M. Esen Tuna, Kamlesh Rath, and Steven D. Johnson. Specification and synthesis of bounded indirection. In *Proceedings of the Fifth Great Lakes Symposium on VLSI (GLSVLS I'95)*, pages 86–89. IEEE, March 1995. http://www.cs.indiana.edu/Research/techreports/

25. Steven D. Johnson, Paul S. Miner, and Albert Camilleri. Studies of the single pulser in various reasoning systems. In Ramayya Kumar and Thomas Kropf, editors, *Theorem Provers in Circuit Design, Theory Practice and Experience (TPCD'94)*, volume 901 of *Lecture Notes in Computer Science*, pages 209–227. Springer, 1995. Second International Conference, TPCD '94, Bad Herrenalb, Germany, September 26-28, 1994, Proceedings. http://www.cs.indiana.edu/~sjohnson/papers/Johnson-Miner-Camilleri-95.pdf

24. Paul S. Miner, Shyamsundar Pullela, and Steven D. Johnson. Interaction of formal design systems in the development of a fault-tolerant clock synchronization circuit. In *13th Symposium on Reliable Distributed Systems (RDS'94)*, pages 128–137, 1994. http://www.cs.indiana.edu/~sjohnson/papers/Miner-Pullela-Johnson-94.pdf

23. M. Esen Tuna, Steven D. Johnson, and Bob Burger. Continuations in hardware-software codesign. In *Proceedings of the 1994 IEEE International Conference on Computer Design (ICCD'94)*, pages 264–269. IEEE Cat. No. 94CH35712, October 1994. http://www.cs.indiana.edu/Research/techreports/

22. Zheng Zhu and Steven D. Johnson. Capturing synchronization specifications for sequential compositions. In *Proceedings of the 1994 IEEE International Conference on Computer Design (ICCD'94)*, pages 117–121. IEEE Cat. No. 94CH35712, October 1994. http://www.cs.indiana.edu/~sjohnson/papers/Zhu-Johnson-94.pdf

21. Kamlesh Rath, Venkatesh Choppella, and Steven D. Johnson. Decomposition of sequential behavior using interface specification and complementation. *VLSI Design*, 3(3-4):347–358, 1995.

20. Kamlesh Rath, M. Esen Tuna, and Steven D. Johnson. Behavior tables: A basis for system representation and transformational system synthesis. In *Proceedings of the IEEE/ACM International Conference on Computer Aided Design (ICCAD'93)*, pages 736–740. IEEE Cat. No. 93CH3344-9, November 1993. http://www.cs.indiana.edu/Research/techreports/

19. Kamlesh Rath, Bhaskar Bose, and Steven D. Johnson. Derivation of a DRAM memory interface by sequential decomposition. In *Proceedings of the International Conference on Computer Design (ICCD'93)*, pages 438–441. IEEE Cat. No. 93CH3335-7, October 1993. http://www.cs.indiana.edu/Research/techreports/

18. Bhaskar Bose, M. Esen Tuna, and Steven D. Johnson. System factorization in codesign: A case study of the use of formal techniques to achieve hardware-software decomposition. In *Proceedings of the International Conference on Computer Design (ICCD'93)*, pages 458–461. IEEE Cat. No. 93CH3335-7, October 1993.

17. Bhaskar Bose, Steven D. Johnson, and Shyam Pullela. Integrating boolean verification with formal derivation. In D. Agnew, L. Claesen, and R. Camposano, editors, *Computer Hardware Description Languages and their Applications (CHDL'93)*, volume A-32 of *IFIP Transactions*, pages 127–134. North-Holland, 1993. Proceedings of the 11th IFIP WG 10.2 International Conference on Computer Hardware Description Languages and their Applications - CHDL'93, sponsored by IFIP WG 10.2 and incooperation with IEEE COMPSOC, Ottawa, Ontario, Canada, 26-28 April, 1993. http://www.cs.indiana.edu/Research/techreports/

16. Bhaskar Bose and Steven D. Johnson. DDD-FM9001: Derivation of a verified microprocessor. an exercise in integrating verification with formal derivation. In G. Milne and L. Pierre, editors, *Correct Hardware Design and Verification Methods (CHARME'93)*, volume 683 of *Lecture Notes in Computer Science*, pages 191–202. Springer, 1993. IFIP WG 10.2 Advanced Research Working Conference, CHARME'93, Arles, France, May 24-26, 1993, Proceedings. http://www.cs.indiana.edu/Research/techreports/

15. Zheng Zhu and Steven D. Johnson. Automatic synthesis of sequential synchronizations. In D. Agnew, L. Claesen, and R. Camposano, editors, *Computer Hardware Description Languages and their Applications (CHDL'93)*, volume A-32 of *IFIP Transactions*, pages 285–301. North-Holland, 1993. Proceedings of the 11th IFIP WG 10.2 International Conference on Computer Hardware Description Languages and their Applications - CHDL'93, sponsored by IFIP WG 10.2 and incooperation with IEEE COMPSOC, Ottawa, Ontario, Canada, 26-28 April, 1993. http://www.cs.indiana.edu/Research/techreports/

14. Kamlesh Rath and Steven D. Johnson. Toward a basis for protocol specification and process decomposition. In D. Agnew, L. Claesen, and R. Camposano, editors, *Computer Hardware Description Languages and their Applications (CHDL'93)*, volume A-32 of *IFIP Transactions*, pages 169–186. North-Holland, 1993. Proceedings of the 11th IFIP WG 10.2 International Conference on Computer Hardware Description Languages and their Applications - CHDL'93, sponsored by IFIP WG 10.2 and incooperation with IEEE COMPSOC, Ottawa, Ontario, Canada, 26-28 April, 1993.

13. Steven D. Johnson and Bhaskar Bose. A system for mechanized digital design derivation. In *IFIP and ACM/SIGDA International Workshop on Formal Methods in VLSI Design*, 1991. IFIP WG 10.2 conference series. Available as Indiana University Computer Science Department Technical Report No. 323 (rev. 1997). http://www.cs.indiana.edu/Research/techreports/

12. Zheng Zhu and Steven D. Johnson. An example of interactive hardware transformation. In P. A. Subramanyam, editor, *IFIP and ACM/SIGDA International Workshop on Formal Methods in VLSI Design*, January 1991. IFIP WG 10.2 conference series. http://www.cs.indiana.edu/Research/techreports/

11. Zheng Zhu and Steven D. Johnson. An algebraic framework for data abstraction in hardware description. In Garaint Jones and Mary Sheeran, editors, *Designing Correct Circuits (DCC'90)*, Workshops in Computing. Springer, 1990. Workshop jointly organised by the Universities of Oxford and Glasgow, 26-28 September 1990, Oxford.

10. John Franco, Daniel P. Friedman, and Steven D. Johnson. Multi-way streams in scheme. *Computer Languages*, 15(2):109–125, 1990.

9. Steven D. Johnson, R.M. Wehrmeister, and B. Bose. On the interplay of synthesis and verification: Experiments with the FM8501 processor description. In Claesen, editor, *Formal VLSI Specification and Synthesis, VLSI Design Methods - I*, pages 385–404. North-Holland, 1990. Proceedings of the IFIP WG 10.2/WG 10.5 International Worksho p on Applied Formal Methods for Correct VlSI Design, Sponsored by IMEC, Houtha len, Belgium, 13-16 November, 1989.
8. Zheng Zhu and Steven D. Johnson. An algebraic characterization of structural synthesis for hardware. In L.J.M. Claesen, editor, *Formal VLSI Specification and Synthesis, VLSI Design Methods - I*. North-Holland, 1990. Proceedings of the IFIP WG 10.2/WG 10.5 International Workshop on Applied Formal Methods for Correct VlSI Design, Sponsored by IMEC, Houthalen, Belgium, 13-16 November, 1989.
7. C.D. Boyer and Steven D. Johnson. Using the digital design derivation system: case study of a VLSI garbage collector. In Darringer and Ramming, editors, *Ninth International Symposium on Computer Hardware Description Languages (CHDL'89)*, Amsterdam, 1989. IFIP WG 10.2, Elsevier.
6. Steven D. Johnson. Manipulating logical organization with system factorizations. In M. Leeser and G. Brown, editors, *Hardware Specification, Verification and Synthesis: Mathematical Aspects*, volume 408 of *Lecture Notes in Computer Science*, pages 260–281. Springer, July 1989.
   `http://www.cs.indiana.edu/~sjohnson/papers/Johnson-98.pdf`
5. Steven D. Johnson and C.D. Boyer. Modelling transistors applicatively. In G.J. Milne, editor, *The Fusion of Hardware Design and Verification*, pages 397–420. North-Holland, 1988. Proceedings of the IFIP WG 10.2 Working Conference on The Fusion of Hardware Design and Vierification, Glasgow, Scotland, 4-6 July, 1988.
4. Steven D. Johnson, B. Bose, and C.D. Boyer. A tactical framework for digital design. In Birtwistle and Subramanyam, editors, *VLSI Specification, Verification and Synthesis*, pages 349–383. Kluwer, Boston, 1988.
3. Steven D. Johnson. Digital design in a functional calculus. In G. Milne and P.A. Subramanyam, editors, *Formal Aspects of VLSI Design*, pages 153–178. North-Holland, Amsterdam, 1986. Proceedings of tthe 1985 Edingurgh Workshop on VLSI, Edinburgh, Scotl and, U.K.
2. Steven D. Johnson. Applicative programming and digital design. In *POPL '84: Proceedings of the 11th ACM SIGACT-SIGPLAN symposium on Principles of programming languages*, pages 218–227, New York, NY, USA, 1984. ACM.
1. Steven D. Johnson. Circuits and systems: Implementing communication with streams. *IMACS Transactions on Scientific Computation*, II:311–319, 1983.

Books and Collections

3. Warren A. Hunt, Jr. and Steven D. Johnson, editors. *Formal Methods in Computer-Aided Design, Third International Conference, FMCAD 2000, Austin, TX, USA, November 1-3, 2000, Proceedings*, volume 1954 of *Lecture Notes in Computer Science*, Heidelberg Berlin, 2000. Springer-Verlag.
   `http://www.springerlink.com/content/978-3-540-41219-9/`
2. Steven D. Johnson, editor. *Computer Hardware Description Languages and Their Applications (CHD L'95)*. IFIP WG 10.2/5, IEEE Cat. No. 95TH8102, August 1995. Proceedings of the IFIP International Conference held in conjunction with ASP-DAC'95 and VLSI'95, August 29-Sept 1, Makuhari Messe, Chiba, Japan.
1. Steven D. Johnson. *Synthesis of Digital Designs from Recursion Equations*. MIT Press, Cambridge, 1984.

Other Publications and Reports

22. Steven D. Johnson, Bryce Himebaugh, and Caleb Hess. ERTS: a robotic platform for collaborative experimentation, June 2009. Exhibit, poster, and presentation at the 2009 International Robots, Vision and Motion Control Show and Conference, Chicago.

21. Venkatesh Choppella, Arijit Sengupta, Edward L. Robertson, and Steven D. Johnson. Constructing and validating entity-relationship data models in the PVS specification language: A case study using a textbook example. Technical Report 632, Indiana University Computer Science Department, Bloomington, Indiana, April 2005. http://www.cs.indiana.edu/Research/techreports/

20. Danko Antolovic, Alex Leykin, and Steven D. Johnson. Vanishing point:a visual road-detection program for a darpa grand challenge vehicle. Technical Report 622, Indiana University Computer Science Department, Bloomington, Indiana, December 2005. http://www.cs.indiana.edu/Research/techreports/

19. Steven D. Johnson, 2002. Photo credit in *Contexts,* **2**(2):70.

18. Steven D. Johnson and Eric Jeschke. Modeling with streams in Daisy and the SchemEngine project. In M. Sheeran and T. Melham, editors, *Designing Correct Circuits, (DCC'02)*. ETAPS 2002, 2002. Presentation at the Workshop on Designing Correct Circuits, held on 6–7 April 2002 in Grenoble, France. http://www.math.chalmers.se/~ms/DCC02/Proceedings.html

17. Steven D. Johnson. Formal derivation of a scheme computer. Technical Report 544, Indiana University Computer Science Dept., Bloomington, Indiana, September 2000.
http://www.cs.indiana.edu/Research/techreports

16. Caleb Hess, Steven D. Johnson, Robert W. Wehrmeister, and Ingo Cyliax. Logic Engine User Manual. Electronically published for various Indiana University Computer Science Department courses. Revised yearly through 2004.

15. Steven D. Johnson, Caleb Hess, Bryce Himebaugh, Franklin Prosser, David Winkel, and David Wilson. B441/541 laboratory experiments. Electronically published for Indiana University Computer Science Department courses. Revised yearly.

14. Steven D. Johnson. Franklin Prosser. In *In Honor of Retiring Faculty: April 14, 1999*. Indiana University, 1999.

13. Ingo Cyliax, Steven D. Johnson, and Bhaskar Bose. Arriving at FPGA based hardware unix-encription using iterated codesign methods. Technical Report 496, Indiana University Computer Science Dept., Bloomington, Indiana, October 1997. http://www.cs.indiana.edu/Research/techreports/

12. Steven D. Johnson, Gary B. Parker, Ingo Cyliax, and David Braun. Using cyclic genetic algorithms to reconfigure hardware controlers for robots. Technical Report 494, Indiana University Computer Science Department, Bloomington, Indiana, US, December 1993. http://www.cs.indiana.edu/Research/techreports/

11. Kamlesh Rath, M. Esen Tuna, and Steven D. Johnson. An introduction to behavior tables. Technical Report 392, Indiana University Computer Science Department, Bloomington, Indiana, US, December 1993. http://www.cs.indiana.edu/Research/techreports/

10. Steven D. Johnson *et. al.* Computer science department handbook. Indiana University Computer Science Department. 1993–1994 edition revised with Leslie Ortquist and Dedaimia Whitney; 1992–1993 edition revised with Dedaimia Whitney; 1991–1992 edition cowritten with Pam Larson, Pam Milam and Dedaimia Whitney.

9. K. Rath, I. Celis, and R. M. Wehrmeister. RTBA: A generic bit-sliced bus architecture for datapath synthesis. Technical Report 321, Indiana University Computer Science Department, December 1990.
http://www.cs.indiana.edu/Research/techreports/

8. Steven D. Johnson and Bhaskar Bose. A system for digital design derivation. Technical Report 289, Indiana University Computer Science Department, Bloomington, Indiana, US, August 1989.
http://www.cs.indiana.edu/Research/techreports/

7. Steven D. Johnson. Daisy, DSI and LiMP: Issues in architecture for suspending construction. Technical Report 288, Indiana University Computer Science Department, Bloomington, Indiana, US, August 1989.
http://www.cs.indiana.edu/Research/techreports/

6. Steven D. Johnson. How Daisy is lazy: Suspending construction at target levels. Technical Report 286, Indiana University Computer Science Department, Bloomington, Indiana, US, August 1989.
http://www.cs.indiana.edu/Research/techreports/

5. Steven D. Johnson. Daisy programming manual, 2004. http://www.cs.indiana.edu/~sjohnson/dsi/

4. Steven D. Johnson. Storage allocation for list multiprocessing. Technical Report 168, Indiana University Computer Science Department, Bloomington, Indiana, US, March 1985. http://www.cs.indiana.edu/Research/techreports/

3. Steven D. Johnson. Synthesis of digital designs from recursion equations. Technical Report 141, Indiana University Computer Science Department, Bloomington, Indiana, US, May 1983. Revision published by MIT Press in 1984. http://www.cs.indiana.edu/Research/techreports/

2. A. T. Kohlstaedt and Steven D. Johnson. DSI program description. Technical Report 120, Indiana University Computer Science Department, Bloomington, Indiana, US, November 1981. http://www.cs.indiana.edu/Research/techreports/

1. Steven D. Johnson. An interpretive model for a language based on suspended construction. Technical Report Technical Report, Indiana University Computer Science Department, Bloomington, Indiana, US, aug 1977. MS thesis. http://www.cs.indiana.edu/Research/techreports/

Systems

6. The *ERTS* autonomous vehicle. With Bryce Himebaugh, Caleb Hess, and others. An platform for experimental research in embedded system design and cognitive robotics. https://www.cs.indiana.edu/classes/p545/Public/

5. The *Starfish* formal synthesis system. With Alex Tsow. A tool for design derivation based on a tabular design representation. http://www.cs.indiana.edu/cgi-bin/techreports/TRNNN.cgi?trnum=TR650

4. The *Digital Design Derivation* System. A transformation system for deriving hardware system descriptions from functional expressions. Implemented in Scheme, Emacs, and Tcl/TK.

3. The *Daisy/DSI Programming System*. A concurrent functional list-processing language. Implemented in C under Unix/Linux.

2. *Research prototypes and case studies.* Artifacts b–f were synthesized using the Digital Design Derivation system to demonstrate formal methods research. More information at http://www.cs.indiana.edu/hmg/index.html.

   g. *Sk$^{e}_{y}$^{e}ball* airborne vision system (2002) http://www.cs.indiana.edu/hmg/skeyeball/.

   f. *Schemachine computer* (1994), CPU, hardware garbage collector, and custom memory architecture, realized in FPGAs and PLDs; functioned as a stand-alone computer, and tested against its own executable specification.

   e. *FM9001 microprocessor* (1993), realized in Actel FPGAs; tested against the LSI gate-array version developed by Warren Hunt.

   d. *VLSI garbage collector* (1989), realized in PLAs; partially tested on a testbench.

   e. *SECD computer* (1988), realized in MSI and PLD, with CPU and hardware garbage collector; functioned as a stand-alone computer.

   b. *Hardware garbage collector* (1988), realized in MSI and PLD, and tested against a production implementation of Scheme.

   a. *Parallel DSI/Daisy* (1987), implemented in C on a 32-node BBN Butterfly multiprocessor, including a parallelized heap allocator, garbage collector, and Daisy language implementation. The purpose of the system was performance measurement of a truly parallel computational model. See Eric Jeschke's dissertation.

1. *Instructional laboratories and platforms:*

e. Robotic golf car for instruction in embedded systems. See also *ERTS*, above. `http://www.cs.indiana.edu/classes/p545/`.

d. Visual tracking laboratory for a course entitled *Real-Time and Embedded Systems*, based on an *XESS XSV Virtex 800* development board and a PC operating under *RTLinux*.

c. *Servobot hexapod controller* (1997) realized in reconfigurable FPGAs. A robotics exhibit introducing K-12 students to robotics and control. `http://www.cs.indiana.edu/~sjohnson/projects/servobot/`

b. *TrCAS* instructional lab (2000), hardware infrastructure for a symmetric collision avoidance system, mounted on HO-guage model railroad engines; realized with PIC microprocessor, Xilinx FPGA, and infra-red range-finding sensor. `http://www.cs.indiana.edu/hmg/TrCAS/`

a. The *Logic Engine*. Originally developed by Franklin P. Prosser and David E. Winkel, version LEv4 is an FPGA based digital prototyping environment. `http://www.cs.indiana.edu/hmg/le/LEv4/`

Invited Talks and Competitive Presentations

· ERTS: a robotic platform for collaborative experimentation. Emerging Technologies Session at the International Robotics, Vision & Motion Control Show and Conference, Rosemont, IL, June 11, 2009. See report [22].

· Homogeneous resource configuration and access for an autonomous robotic vehicle. SAE 2008 Commercial Vehicle Engineering Congress & Exhibition. Technical Session CV415 on Autonomous/Robotics Vehicles, paper 2008-01-2719. See publication [46].

· Preliminary explorations in specifying and validating entity-relationship models in PVS. Second Workshop on Automated Formal Methods (AFM'07), November 6, 2007, Atlanta, Georgia. See publication [45].

· Current Topics in Interactive System Design Derivation. Intel Workshop on Applications of Functional Programming to Hardware Description. Hillboro, Oregon, March 31–April 1.

· Modeling with Streams in Daisy and The SchemEngine Project Workshop on Designing Correct Circuits, held on 6–7 April 2002 in Grenoble, France. See item [19] of Other Publications.

· View from the fringe of the fringe. Invited presentation to a joint session of CHARME 2001 and TPHOLs 2001 conferences, Royal Museum of Scotland, Edinburgh, Scotland, September 5, 2001. See publicat ion [39].

· A systematic incrementalization technique and its application to hardware design. 10th IFIP WG 10.5 Conference on Correct Hardware Design and Verification Methods (CHARME'99) See publication [35].

· Evolving Cyclic Behaviors in Hexapod Robots. DePauw University, Apri l3, 1998.

· Using cyclic genetic algorithms to reconfigure hardware controllers for robots. Refereed poster presentation at IEEE Conference on Field Programmable Gate Arrays (FPGA'97), Santa Monica, California, February 1997.

· Integrated reasoning support in system design: Design derivation and theorem proving. Advanced Research Working Conference on Correct Hardware Design and Verification Methods (*CHARME'97*), Montreal, October 18, 1997. See publication [32].

· A tabular language for system design. Fourth NASA Langley Formal Methods Workshop, September 10, 1997. See publication [31].

· Integrating design and verification environments through a logic supporting hardware diagrams. International Conference on Hardware Description Languages and their Applications (CHDL'95), August 30, 1995. See publication [28].

· The Scheme Machine: a case study in progress of digital design derivation at system levels. Third NASA Formal Methods Workshop, Langley Research Center, Hampton Virginia, May 10–12, 1995. Proceedings (visual materials) published as NASA Conference Publication 10176.

· Studies of the single-pulser in various reasoning systems. Second IFIP International Conference on Theorem Provers in Circuit Design (TPCD 94), Bad Harrenalb, Germany, September 1994. See publication [26].

· Capturing synchronization specifications for sequential compositions. International Conference on Computer Design (ICCD 94), Cambridge MA, October 1994. See publication [23].

· A taxonomy of hardware verification methods. NSA Science Advisory Board study group on formal methods in hardware. February 9–10, 1993, Ft. George G. Meade, MD.

· Derivational techniques for hardware verification. Second NASA Formal Methods Workshop, Langley Research Center, August 11–13, 1992.

· Derivational reasoning for digital-system verification. NASA Langley Research Center, Hampton VA, February 22, 1992.

· Heterogeneous reasoning in digital system design. University of Cincinnati Dept. of Electrical Engineering, February 5, 1992.

· Daisy/DSI – an environment for exploring near-functional parallelism. University of Utah Computer Science Department, October 25, 1991.

· Deductive and derivational reasoning in digital system design. University of Utah Computer Science Department, October 24, 1991.

· On the interplay of synthesis and verification: Experiments with the FM8501 microprocessor description. IFIP WG 10.2/10.5 International Workshop on Applied Formal Methods for Correct VLSI Design, Houthlaten, Belgium, November 1989. See publication [10].

· A system for digital design derivation. Refereed abstract, the 1989 IEEE High Level Synthesis Workshop, Kennebunkport, MA, October 1989 (See report [9]).

· Manipulating logical organization with system factorizations, Cornell Mathematical Sciences Institute Workshop on Specification, Verification, and Synthesis, July 1989 (See publication [7]).

· Hardware synthesis in the functional algebra. IFIP WG 2.8 (functional programming) meeting, Mystic, Connecticut, May 1989.

· Algebra for digital design synthesis and its relation to formal verification. AT&T Bell Laboratories, Murray Hill, December 14, 1988.

· The DSI model of concurrent graph processing. 1988 Apsenas Workshop on the Implementation of Lazy Functional Languages, University of Chalmers, Goteberg, September 1988.

· Modeling transistors applicatively. IFIP WG10.2 Working Conference on Hardware, Glasgow, July 1988 (See publication [6]).

· Research in digital design. 1988 IEEE VLSI Workshop, Clearwater Beach, March 1988.

· Digital design derivation. Digital Equipment Corp., Maynard, July 1987.

· A tactical framework for digital design. 1987 Calgary Hardware Verification Workshop, January 1987 (See publication [5]).

· Research at Indiana University in hardware. 1985 IEEE Workshop on Computer Aided Design, Santa Barbara, 1985.

· Digital design based on a functional calculus. Carnegie Mellon University, March 1985.

· Digital design in a functional calculus. IFIP WG 10.2 Workshop on VLSI, Edinburgh, 1985 (See publication [4]).

· Applicative programming and digital design. Eleventh Annual ACM SIGACT-SICPLAN Symposium on Principles of Programming Languages, Salt Lake City, January 1984 (See publication [3]).

Colloquia

· Can Reasoning be Taught? IUCS Programming Languages Seminar. November 18, 2011.

· Aspects of Parallelism in DSI (1977–1995). IUCS Programming Languages Seminar. February 25, 2011.

· The Year of ERTS. Indiana University Computer Science Honors Seminar. October 26, 2009.

· ERTS: a robotic platform for education and research in embedded systems. IUCS Honors Seminar, September 10, 2007.

· Public field trials of GPS navigation in the ERTS vehicle, April 27, 2007.

· System Design Derivation. IUCS Programming Languages Seminar. April 17, 2007.

· Daisy and DSI. IUCS Programming Languages Seminar. April 5, 2007.

· Formal design & verification methods. IU/SOI/CS *Bring IT On* workshop, October 21, 2006.

· A platform for instruction and research in embedded & real-time systems. IUCS System Seminar, April 11, 2006.

· Current Topics in Interactive System Design Derivation. IUCS Honors Seminar, April 25, 2005.

· Ultra-reliable infrastructure for safety-critical applications. IUCS System Seminar, October 30, 2003.

· Skeyeball: vision based guidance for a UAV. IUCS Systems Seminar, October 2, 2003.

· Research survey: Steven D. Johnson. IUCS research survey, March 2003.

· Some background on modeling with streams. The IUCS Programming Languages Seminar. November 20, 1998.

· A basis for design derivation. The IUCS Applied Logic Lunch. March 9, 1998.

· A tool for design derivation. The IUCS Programming Languages Seminar. March 13, 1998.

· Introduction to design derivation I–IV. University of Cincinnati, ECE/CS Department, System Design Group, May 20, June 3, July 8, July 22, 1997.

· Design derivation with behavior tables. University of Cincinnati, ECE/CS Department, System Design Group, October 10, 1996.

· Digital design derivation: an illustrated introduction, University of Cincinnati Department of Electrical and Computer Engineering, System Design Group, November 29, 1995.

· Review of CHDL'95. Applied Logic Seminar, Indiana University Computer Science Department, October 23, 1995.

· An overview of hardware verification issues. Indiana University Computer Science Department Applied Logic Lunch, October 18, 1993.

· Digital design derivation. Berkeley CAD Seminar (ECS 298-11), University of California at Berkeley Department of Electrical Engineering, March 17, 1993.

· Verifications of a microprocessor: two methods of proving hardware correct. Indiana University Logic Group, October 1992.

· Digital design derivation. Indiana University Logic Group, February 1991.

· Digital design derivation. Indiana University Computer Science Department, November 1988.

· Modeling transistors applicatively. Indiana University Computer Science Department, June 1988.

· Functional programming and its relation to design. Indiana University Research Expo, April 1985.

· Programming in Daisy, I & II. Indiana University Computer Science Department, summer 1984.

· DSI/Daisy: boxes, arrows and clouds in 1983 (!?). Indiana University Computer Science Dept., July 1983.

· Connection networks for output driven list multiprocessing. ACM National Conference, St. Louis, 1981.

· An intuitive look at frons. Indiana University Computer Science Dept., 1980.

Other Meetings Attended

· FMCAD 2011, Austin, Texas, October 30 – November 2, 2011.

· FMCAD Steering Committee meeting, San Jose, CS, November 15, 2006.

· Indy Robot Racing Team meetings, September–December, 2004.

· FMCAD Steering Committee meeting, Austin, Texas, November 6, 2004.

· SIG-CHARME meeting, Portland, Oregon, November 8, 2002.

· SIG-CHARME meeting, Livingston, Scotland, September 6, 2001.

· Formal Methods World Congress, Toulouse, September 20–24, 1999.

· Design Automation Conference, New Orleans, June 21–25, 1999.

· IFIP Working Group 10.5 meeting, New Orleans, June 22, 1999.

· IFIP Working Group 10.5 meeting, Palo Alto, California, November 7, 1998.

· International Conference on Formal Methods in Computer-Aided Design. Palo Alto, California, November 4–6, 1998.

· NASA/LRC Formal Methods Group PVS School, Langley, Virginia, August 3-6, 1998.

· Xilinx University Program training workshop, Philadelphia, Pennsylvania, June 24, 1998.

· 21st Century Engineering Consortium Workshop, Melbourne Florida, March 16-18, 1998.

· Second ACM/SIGSOFT Workshop on Formal Methods in Software Practice Clearwater Beach, Florida, March 3-5, 1998

· Planning meetings for the 21st Century Engineering Consortium. Syracuse, New York, March 24, 1997; New York City, May 9, 1997; Cincinnati, Ohio, July 28, 1997, August 11, 1998.

· International Conference on Formal Methods in Computer-Aided Design. Palo Alto, California, November 6–8, 1996.

· Combined Asia and South Pacific Design Automation Conference, IFIP International Conference on Computer Hardware Description Languages and their Applications, IFIP International Conference on Very Large Scale Integration, Makuhari Messe, Chiba, Japan, 29 August to 1 September 1995.

· IFIP WG 10.5 meeting, San Jose, California, November 11, 1994.

· IFIP WG 10.2/10.5 meeting, Grenoble, France, September 1994.

· 1994 IEEE International Conference on Codesign, held jointly with the 1994 IFIP International Conference on Computer-aided Hardware/Software Engineering (CODES/CASHE 94), Grenoble, September 1994.

· 1994 CRA Computer Science Chairs Meeting, Snowbird, Utah, July 1994.

· 1994 High Level Synthesis Symposium, Niagra-on-the-lake, Canada.

· 1993 IEEE International Conference on Computer Design (ICCD 93), Boston, October 1993

· 1993 IEEE International Conference on Computer Aided Design (ICCAD 93), Santa Clara, November 1993.

· IFIP WG 10.2 meeting, Ottawa, Canada, 1993.

· IFIP WG 10.2 Eleventh International Symposium on Computer Hardware Description Languages and their Applications, Ottawa, Canada, 1993.

· Workshop on Computer-aided Verification of Digital Circuits, Motorola Corp., Austin Texas, October 30, 1992.

· Computational Logic Inc. Research Presentation, Austin Texas, April 21–23, 1992.

· 1992 IEEE International Conference on Computer Design Boston, MA, October 11–13, 1991. Session chairman, *Environments for High-level CAD.*

· Second NASA Formal Methods Workshop, Langley Research Center, Hampton, VA, August 11–13, 1992. Panel member: *Issues in Hardware Verification.*

· IFIP Working Group 10.2 committee meeting, Boston, September 1991.

· 1991 IEEE International Conference on Computer Design, Boston, MA, September 1991.

· 1991 Microelectronic System Education Conference, San Jose, CA, July 1991.

· IFIP WG10.2 meeting, Charlottesville, VA, November 1990.

· Oxford Workshop on Designing Correct Circuits (DCC 90), Oxford, U.K., September 1990.

· 1990 ACM/IEEE Design Automation Conference, Orlando FL, June 1990.

· IFIP WG10.2 meeting, Orlando FL, June 1990. Orlando, Florida, June 1990.

· Trusted Systems Development, Computational Logic, Inc., Austin TX, October 1988.

· Second Calgary Workshop on VLSI Verification, Banff, Canada, June 11–17, 1988.

· IEEE/ACM-SIGACT Symposium on Logic in Computer Science, Cambridge, MA, June 16–18, 1986.

· The 1981 Marktoberdorf Summer School, Marktoberdorf, Germany, July–August, 1981.

Public Presentations

· "Skeyeball: Vision for a robotic airplane," with Danko Antolovic, Caleb Hess and Bryce Himebaugh. South Central Indiana Regional Science and Engineering Fair, Ivy Tech State College, Bloomington Campus, March 6, 2004.

· Robot demonstrations, Monroe County Public Library Bloomington and Ellettesville branches, with Bryce Himebau and Caleb Hess, July 15, 2002.

· "Robots!," with Bryce Himebaugh and Caleb Hess, Wonderlab Science Center, Bloomington, Indiana, March 2, 2002.

· "Robots!," with Bryce Himebaugh and Caleb Hess, Wonderlab Science Center, Bloomington, Indiana, August 23, 1998.

· Binford Elementary School Intersession, Bloomington, Indiana, February 1998.

· Pleasant Run Elementary School Writers Seminar, Cincinnati, Ohio, November 18, 1996.

· Physics Department Open House, Bloomington, Indiana September 1995.

Grants and Donations

21. *Hand It To Me.* With Matthias Scheutz and Randall Beer. Indiana University Faculty Research Support Program. $75,118.

20. IU School of Informatics and Computer Science Department, December 2005, $10,000 for E&RTS lab development.

19. Indy Robotics, 2004, components valued at $3,000.

18. Xilinx Corporation Educational Program donation, 2002. Components valued at $225,000.

17. *A design tool based on behavior tables* (a NASA/GSRP fellowship), NASA NGT-1-01009, 2002–2004. Faculty supervisor. $70,000.

16. *Daisy/Patmos Connection Machine Pilot Project* Patmos International Corporation, Principal Investigator, $37,000.

15. *Hardware System Design with Behavior Tables.* NSF MIP96-10358, 1997–2000, Principal Investigator. $282,000.

13. *Assistantship for a Pilot Project in Electronic Application to Graduate School.* Indiana University Office of Research and the University Graduate School, 1993–1995. Project Director. $22,000.

12. *Graduate Research Traineeship Initiative in Robotics and Intelligent Control.* NSF GER93-54898, 1993–1998. Project Director with co-director Jonathan W. Mills. $557,500 plus $41,250 cost sharing.

11. *An Infrastructure for Conceptualization and Visualization of Computation* (PI: David S. Wise). NSF CDA 93-03189, 1993–1998. Faculty participant, (PI: D. S. Wise) departmental CISE research infrastructure project, $1.7 million.

10. *Decomposing digital-system specifications into interacting sequential processes.* NSF MIP92-08745, 1992–1995. Principal Investigator. $142,000.

9. *The Derivation of a Verified Microprocessor FM8502* (a NASA fellowship), NASA NGT-50861, 1991–1994. Faculty supervisor $66,000.

8. SIGDA library grant, 1990. $1,000.

7. *Algebra for digital design derivation.* NSF MIP89-21842, 1990–1992. Principal Investigator. $203,000.

6. *Digital design derivation.* NSF MIP87-07067, 1987–1989. co-Principal Investigator with David E. Winkel. $206,000.

5. *A conduit from theory to practice.* NSF DCR85-21497, 1985–1990, departmental CER project, 1985–1991, Participating faculty member (PI: E. R. Robertson), proposal coordinator and editor. $2,853,313 plus cost sharing.

4. *Methods and architectures for applicative programming.* NSF DCR84-05241, 1984–1987, co-Principal Investigator with David S. Wise. Approximately $175,000.

3. *Applicative Programming for Indeterminate Systems.* MCS82-03978, 1982–1984. Research Assistant (co-PIs Daniel P. Friedman and David S. Wise),

2. *Applicative Programming for Systems.* MCS77-22325 1978–1982. Research Assistant (co-PIs Daniel P. Friedman and David S. Wise),

1. *Structured Recursion: Its Properties, Translation, and Implementation.* MCS75-08145, 1975–1978. Research Assistant (co-PIs Daniel P. Friedman and David S. Wise),

Students Directed

· Danko Antolovic (MS, 2001). *Development of a Real-Time Vision System for an Autonomous Model Airplane.*

· Bhaskar Bose (PhD, 1994), *DDD/FM9001: Derivation of a Verified Microprocessor.* NASA Fellow.

· Scott Dial. Topic: Synchronous Process Communication via the File System

· Kathryn Fisler (PhD, 1996). *A Unified Approach to Hardware Verification Through a Heterogeneous Logic of Design Diagrams.* AT&T Bell Labs Fellow. Co-supervised with Jon Barwise.

· Eric Jeschke (PhD, 1995). *An Architecture for Parallel Symbolic Computation based on Suspending Construction.*

Lee Pike (PhD, 2006). *The Formal Verification of Time-Triggered Systems*

· Paul Miner (PhD, 1998). *Hardware Verification using Coinductive Assertions*

· Kamlesh Rath (PhD, 1995). *Sequential-system Factorization.*

· Alex Tsow (PhD, 2007). *Starfish: A Table-Centric Tool for Design Derivation.*

· Zheng Zhu, (PhD, 1992). *Structured Hardware Design Transformations.*

· *Research Committees:* Gerard Allwein (PhD 1992, M Dunn), Venkatesh Choppella (PhD 2002, C Haynes), Ignacio Celis (PhD 1996, J Mills), Cordelia Hall (PhD 1987, D Wise), Alex Leykin (CS, M Tucerian), Byron Long (CS, D Leivant), Margaret Montenyohl (PhD 1986, M Wand), Richard Mong (PhD 1987, P Purdom), Ahmet Mustacoglu (CS, G Fox), Robert Montante (PhD 1997, J Mills), Gary Parker (PhD 1999, J Mills), Bhanu Pisupati (CS, G Brown), Lei Qian (PhD 2005, L Moss), Lavanya Ramakrishnan (CS, D. Gannon), Jeremy Siek (PhD 2005, A Lumsdane), John Springer (CS, E Robertson), Michael Wollowski (PhD 1998, J Barwise).

· *External Examiner:* Ricky Lap Ki Chan (PhD 1991, G Hellestrand), K Y Cheung (PhD 1998, G Hellestrand).

Teaching

Courses: Computer Organization. Digital Design I–II. Discrete Structures for Computer Science. Embedded & Real-Time Systems. Introduction to Verification. Logic and Program Verification. Mathematical Foundations of Computer Science. Microelectronic Computer-Aided Design. Operating Systems I–III. Theory of Operating Systems.

Seminars: Formal methods in System Design (1998); *Hardware derivation* (1992, 1993); *Logic Synthesis*, with Jonathan W. Mills (1991); *Laboratory in lisp: Experiments in functional multiprocessing*, with David S. Wise and Eric Jeschke (1990); *Hardware derivation* (1989); *Daisy, DSI and LiMP: Topics in functional multiprogramming* (1988); *Applicative system description* (1987); *Daisy and DSI* (1985).

Professional Activities, Affiliations and Honors

· Member of the *Indy Robotics Racing Team*, 2004–2005.

· *Teaching Excellence Recognition Award*, Indiana University College of Arts and Sciences, Spring 1999.

· Director, *System Design Methods Laboratory*, Indiana University Computer Science Department.

· *Editorial Board:*, *Formal Methods in System Design*, 1990–2005.

· *Program Chair*: *1995 IFIP Working Group 10.2 International Conference on Hardware Description Languages and their Applications (CHDL'95). 2000 International Conference on Formal Methods and Computer Aided Design (FMCAD00), with Warren A. Hunt, Jr.*

· *Planning group*, 21st Century Engineering Consortium.

· *Reviewer for: Acta Informatica*, *ACM Trans. on Programming Languages and Systems (ToPLAS)*, *ACM Computing Surveys*, *Formal Methods in System Design*, *IEEE Transactions on Computer Aided Design*, *IEEE Transactions on Computers*, *IEEE Transactions on Parallel and Distributed Computing*, *IEEE Transactions on Software Engineering*, *IEEE Transactions on Design Automation for Electronic Systems*, *IEEE Transactions on VLSI*, *International Journ. of Parallel Programming*, *Integration–The VLSI Journal*, *Journal of Automated Reasoning*, *Journal of Functional Programming*, *Journal of Systems Architecture*, *Software—Practice and Experience*, *Software Tools for Technology Transfer*, The Oxford University Press, National Science Foundation, U.S. Civilian Research and Development Foundation, IFIP World Congress, Morgan-Kaufmann Publishing Co. Springer Publishing Co.

· *Program Committees:* IEEE International Conference on Computer Design (ICCD92–05); IFIP International Conference on Computer Hardware Description Languages and their Applications (CHDL 1993, 1995, 1997, 1999); IFIP International Conference on Correct Hardware Methods (CHARME 1997,

1999, 2001, 2003, 2005); Conference on Design Automation and Test in Europe (DATE 1999, 2000); Designing Correct Circuits (DCC 1990, 2002, 2004); Workshop on Hardware Design and Functional Languages (HFL 2007,2008) ACM Conference on Embedded Software (EMSOFT 2005); International Conference on Formal Methods and Computer Aided Design (FMCAD 1996, 1998, 2000, 2002, 2004, 2006); NASA-Langley Formal Methods Workshop (Lfm 2000); Workshop on Qualitative Aspects of Programming Languages (QAPL 2004–06); International Conference on Information and Communication Technology (ICICT 2005); International Conference on Theorem Proving and Higher Order Logics (TPHOLs 2000); VHDL International User Forum (VIUF 1995).

- *Member:* SIG-CHARME Special Interest Group for Formal Design and Verification Methods for Hardware-Like Systems (chair 2001–2005), FMCAD Steering Committee, IFIP WG10.5, (system engineering and design tools), Association for Computing Machinery, IEEE Computer Society, IFIP, Sigma Xi; observer IFIP WG2.8 (Functional Programming, 1989).

- *Department/School Service Appointments:* Computer Science Department Director of Graduate Studies (1984–1991); Computer Science Department Chair (1993–1994); Chair of the School of Informatics Promotion & Tenure Committee (2006–2007). Committee memberships have included Undergraduate Education, Graduate Education, and Computing Facilities.

- ACM Distinguished Dissertation, 1984. Esther M. Kinsley Dissertation Award, 1984.

Research Statement

My main research area is *formal methods for systems* of digital components—both software and hardware. My approach stems from earlier investigations in applicative programming languages and methods. Although it involves formalism, this research is applied in character. I study design methods with the two-fold purpose of gaining insight into conceptual aspects of "design" and of implementing better reasoning tools. I am particularly interested in interactive aspects of reasoning, that is, the interplay between (human) conceptualization and automated analysis. My methodology centers on algebraic manipulation of functional modeling expressions and its interplay with logical reasoning. In 2005 I began adapting and re-focusing my work from the digital hardware domain to that of embedded software.

A thesis of my work is that *design* involves orthogonal aspects whose synthesis must be reflected in a reasoning formalism. My metaphor is a tetrahedron whose four facets are labeled by the principal aspects of system design: architecture, behavior, concurrency and data. One can view any facet straight on, but a balanced view of any two or three facets distorts each of them, and seeing all at the same time is impossible. Neither is it possible to subsume all these aspects in one conceptual framework, nor, consequently, a single design notation. I do not think it will prove fruitful to blend all four aspects into a monolithic formalism. Instead, a foundation is needed which allows different aspects to be brought out at different times, as one might turn the tetrahedron to expose the facets.

Although my work addresses all the principle aspects, it is centered on the formal synthesis of architecture. I have come to believe that architecture is poorly reflected in most treatments of verification and synthesis, even as it becomes a dominant feature in modern programming.

I want my work to contribute to and be informed by practice, so I devote a substantial portion of my research effort to implementing the formalisms I study, integrating them with lower level tools, and gaining experience with them through case studies and tutorial demonstrations.

Teaching Philosophy

My two guiding principles as a computer science educator are juxtaposing conceptual abstraction with physical realization, and enabling individual discovery. The core problem in computing is one of translating the specification of a computation to an implementation based in a class of target models. Programming (or design) languages provide medium for performing this translation. Students are presented with a variety

of languages, some concept oriented and some machine oriented. To make sense of this variety, they need exposure to the relevant mathematics, on the one hand, and structured design experiences on the other.

One cannot judge an abstraction without knowing what it abstracts *from*. Neither the mathematics nor the "physics" of computing should take precedence. They should be taught progressively, hand in hand. In my courses I strive to create an environment and circumstances in which my students can make discoveries and draw their own conclusions from their experience. I develop laboratories in which the participants are exposed to real devices, and experience the snowballing complexities of ever larger design challenges.

My approach to pedagogy is dialectic: in the classroom, I present an orderly progression of topics and techniques, even while knowing the laboratory experience requires a holistic understanding from the outset. It is vital for students to individually experience and resolve these two perspectives.