Expanded Final Report

Decomposing digital-system specifications into interacting sequential processes

Grant Number: MIP-9208745, 12/1/92–5/31/97[1]
Principal Investigator: Steven D. Johnson

Computer Science Department
Indiana University
June 4, 1997

Contents

[1] Period includes a one-year, no-cost extension.

1 Summary

1.1 Summary statement (NSF Form 98A, Part II)

In order to design at higher levels of abstraction, system designers require systematic techniques to decompose specifications into interacting subsystems. Since such decompositions often involve complex process interactions to achieve proper synchronization, they are very difficult to do correctly. The first objective of this project was to characterize this problem with sufficient mathematical rigor to support formal verification. Our formalization is based on an algebra for constructive correctness, as opposed to a logic for retrospective correctness proofs, but both aspects must be represented in any practical methodology. Hence, an important subtopic of this work is heterogeneous reasoning, that is, the integrated use of various formal systems in a common design context. This work is aimed at supporting a creative reasoning process applied at appropriate levels for human involvement in design. A reasoning support tool called DDD embodies the research. A number of case studies were done using DDD to demonstrate how this formal method interacts with real design processes and CAD tools. The immediate applications of this research are in high-assurance design, where the extreme cost of design errors justifies the greater expense of formalized design processes. Further improvement of the supporting tools will lead to broader deployment in engineering practice and ultimately to improved engineering methodology.

1.2 Outline

Further Information about this project is posted on the World Wide Web at URL `http://www.cs.indiana.edu/hmg/hmg.html`. The main results of the project were in the following areas:

- A formal characterization of sequential-system decomposition based on a first-order state-transition systems and an *interface specification language*.
- *Behavior tables* as a specification notation for system design and decomposition.
- Further enhancements to the *DDD* transformation system for interactive design derivation (but not a full implementation of the theoretical results).

- Disseminable case studies, including a formally designed, fabricated, and tested integrated circuit for fault-tolerant clock synchronization, and a computer dedicated to executing compiled Scheme.

As the research progressed, two critical issues intruded on the main line of work. One was the need for *heterogeneous reasoning*, that is, the integrated use of a variety of reasoning tools to accomplish a design goal efficiently. Second was the need for a better notation to serve as the "medium" for formalized design. Results from the latter half of this project reflect a broadening of the research to address these new issues, as well as continued refinement of the derivation algebra.

2 Results

2.1 Formal methods

2.1.1 Sequential decomposition

Rath's dissertation [20, 21] develops a basis for decomposition based on *interface specification*. Briefly, the main idea is to graft a *complement path implementation* into the specification behavior at its point(s) of interaction with a determined co-process. Such a decomposition is constructive, hence, in principle, requires no proof of correctness.[2] mong several remaining open problems are: heuristics for composing decompositions both serially and in parallel; provisions for human guidance; and integration with other aspects, specifically, type and behavioral hierarchies. Zhu's work (completed before the start date of this project), was based on a weaker form of interface specification but included some decidability results [26]. In experimental investigations we demonstrated how a restricted form of *continuations* are used for leverage in deriving decompositions from a sub-procedure relationship [24].

2.1.2 Heterogeneous reasoning

Performing derivations at higher levels of abstraction magnifies the need for an integrated system of reasoning support tools. This aspect is highlighted

[2] This is, admittedly, a simplistic statement. One verification problem lies in *mapping* the interface specification to a particular concrete component. It is also generally the case that such transformations may generate formal side-conditions, such as preservation of a surrounding synchronization protocol.

two case studies, both of which will be presented in detail in Miner's forthcomming PhD dissertation. The first is the verification of a fault-tolerant clock synchronization circuit, involving design derivation, boolean equivalence, and higher-order theorem [17, 19]. In a subsequent study, Miner used PVS to prove a *class* of SRT division algorithms, and then DDD to instantiate one instance of that class and reduce it to a hardware description . Observations about the challenges of integrated reasoning support, based on these case studies, are summarized in [15].

The studies led to more basic research into the foundations of heterogeneous reasoning. In addition to a comparative study of how different reasoning tools address a common design [16, 11]. We were also involved in collaborative research into the use of diagrams in formal reasoning [14, 10].

2.2 Automation

2.2.1 Digital Design Derivation system

Bose's dissertation updates the description of the DDD transformation system and gives several examples of its use [1]. Other, more sketchy, examples of the system in operation are found in [16, 4, 17].

Bose, Tuna, and another Indiana PhD student, Venkatesh Chopella, founded Derivation Systems, Inc., Carlsbad, California, to commercialize software developed under grants MIP-9208745 and MIP-8921842. They received SBIR Phase I and II funding through NASA to redevelop DDD as a system called DRS [5].

2.2.2 DDD user interface

In the course of our case studies, a tabular representation, *behavior tables*, emerged as a perspicuous representation for DDD derivations. We developed this notation formally and graphically [25, 23], and have proposed further refinements as a continuation of this research [13].

2.3 Experimental systems

2.3.1 FM9001 microprocessor

This case study (See Photo 2.3.1(a)) was completed prior to the start of MIP-9208745, but several publications about it after 1992 cite support from this grant . The prototype project, which performs direct comparisons between Hunt's FM9001 and Bose's DDD-FM9001, was made available on Internet in 1993 for small software experiments. The FM9001 prototype is used in a continuing project to develop an instructional vehicle for a first undergraduate course in computer organization. Two undergraduate students, Derek Kern and Lisa Hatchett, have done independent-study projects developing this system.

2.3.2 Scheme computer

A case study to derive major components (garbage collector, CPU, etc.) of a computer system for execution of compiled Scheme was completed [4] (See Photo 2.3.1(b)). A continuation of the study is proposed to verify system level interactions and connect to formal derivations of the compiler.

2.3.3 Clock synchronization circuit

Paul Miner used the DDD system in conjunction with other reasoning tools to design a fault tolerant clock synchronization circuit [17, 19]. A realization of the circuit was fabricated through MOSIS and successfully tested at NASA's Langley Research Center.

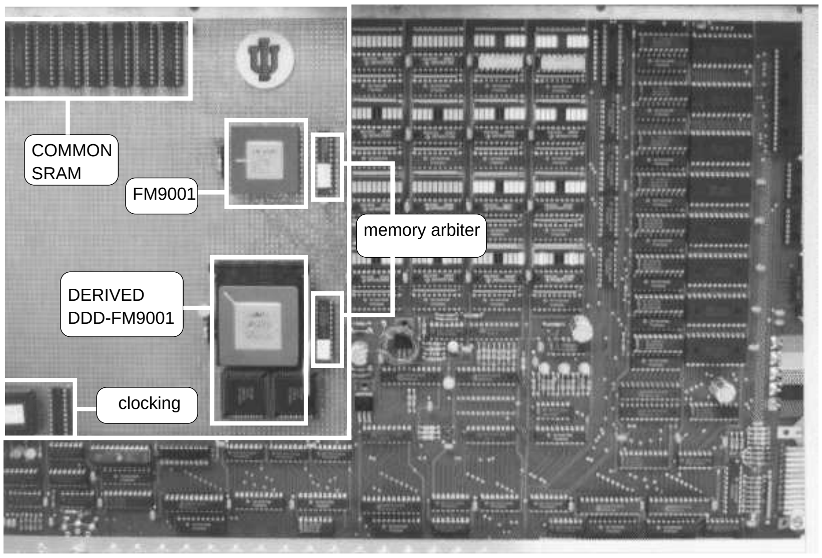
Research Publications

Those citing support from MIP92-08745 are indicated with a ⋆

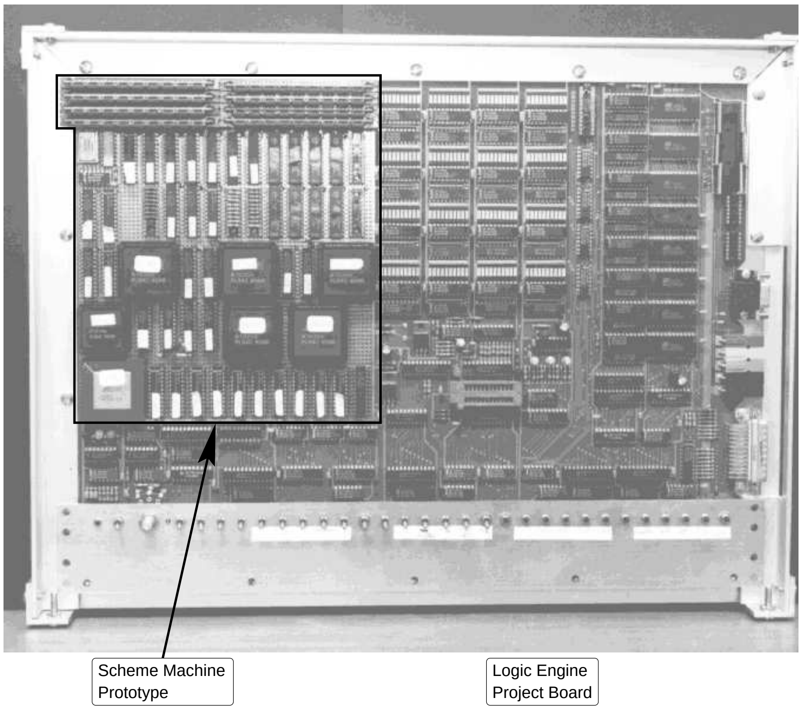
⋆[1] Bhaskar Bose. *DDD-FM9001: Derivation of a Verified Microprocessor.* PhD thesis, Computer Science Department, Indiana University, USA, 1994. Technical Report No. 456, 155 pages.

⋆[2] 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, *Proceedings of IFIP Conference on Correct Hardware Design and Verification Methods*, pages 191–202. Springer, LNCS 683, 1993. also published as Technical Report 380, Computer Science Department, Indiana University.

⋆[3] Bhaskar Bose, Steven D. Johnson, and Shyam Pullela. Integrating boolean verification with formal derivation. In D. Agnew, L. Claesen,



(a) Two verified versions of the FM9001.



(b) Derived Scheme Computer

and R. Camposano, editors, *Proceedings of IFIP Conference on Hardware Description Languages and their Applications*, pages 127–134. Elsevier, April 1993. Also published as Technical Report No. 372, Dept. of Computer Science, Indiana University.

⋆[4] Robert G. Burger. The scheme machine. Technical Report 413, Indiana University, Computer Science Department, August 1994. 59 pages.

[5] Derivation Systems, Inc., Carlsbad, California. *DRS: Derivational Reasoning System*, 1.2.1 edition, December 1995. Contact `drs@derivation.com`.

[6] Kathryn Fisler. A cononical form for circuit diagrams. Technical Report 432, Indiana University Computer Science Department, May 1995.

[7] Kathryn Fisler. Extending formal reasoning with support for hardware diagrams. In Ramayya Kumar and Thomas Kropf, editors, *Theorem Provers in Circuit Design*, pages 298–303. Springer, 1995. LNCS vol. 901, proceedings of the Second International Conference, TPCD'94.

[8] Kathryn Fisler. A logical formalization of hardware design diagrams. Technical Report 416, Indiana University Computer Science Department, September 1995.

[9] Kathryn Fisler. *A Unified Approach to Hardware Verification Through a Heterogenious Logic of Design Diagrams*. PhD thesis, Computer Science Department, Indiana University, USA, 1996.

⋆[10] 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. CHDL proceedings pp. 493-696 of the "ACV'95" held August 29 to September 1, 1995, Chiba, Japan.

⋆[11] Steven D. Johnson. `ftp://ftp.cs.indiana.edu/pub/singlepulserstudy`. A collection of sources and data for various studies in [16].

⋆[12] Steven D. Johnson. The scheme machine: a case study in progress of digital design derivation at system levels. In *Third NASA/Langley Formal*

Methods Workshop. NASA Conference Publication 10176, 1995. Visual materials presented at the workshop, May 10-12, Hampton Virginia.

⋆[13] Steven D. Johnson. Behavior tables. In *Fourth NASA Langley Formal Methods Workshop*, 1997. to appear.

⋆[14] 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. In press.

⋆[15] Steven D. Johnson and Paul S. Miner. integrated reasoning support in system design: design derivation and theorem proving. submitted.

⋆[16] 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*, pages 209–227. Springer, 1995. LNCS vol. 901, proceedings of the Second International Conference, TPCD'94.

⋆[17] Paul S. Miner and Steven D. Johnson. Verification of an optimized fault-tolerant clock synchronization circuit: A case study exploring the boundary between formal reasoning systems. In Satnam Singh, Mary Sheeran, and Geraint Jones, editors, *Third Workshop on Designing Correct Circuits*. Springer Electronic Workshops in Computing, 1996. `http://www.springer.co.uk/ewic/workshops/DCC96`.

[18] Paul S. Miner and James F. Leathrum. verification of IEEE compliant subtractive division algorithms. In M. Srivas and A. Camilleri, editors, *Formal Methods in Computer Aided Design*. Springer, 1996. LNCS 1166, 1st FMCAD Conference.

⋆[19] 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*, pages 128–137, 1994. Proceedings of SRDS 94 held at Dana Point, California, October 1994.

⋆[20] Kamlesh Rath. *Sequential System Decomposition*. PhD thesis, Computer Science Department, Indiana University, USA, 1995. Technical Report No. 457, 90 pages.

⋆[21] Kamlesh Rath, Venkatesh Choppella, and Steven D. Johnson. Decomposition of sequential behavior using interface specification and complementation. *VLSI Design Journal*, 3(3-4):347–358, 1995. In print, special issue on decomposition.

⋆[22] Kamlesh Rath, M. Esen Tuna, and Steven D. Johnson. Behavior tables: A basis for system representation and transformational system synthesis. In *Proceedings of the International Conference on Computer Aided Design (ICCAD)*, pages 736–740. IEEE, November 1993. Also published as Technical Report 89, Computer Science Department, Indiana University.

⋆[23] Kamlesh Rath, M. Esen Tuna, and Steven D. Johnson. An introduction to behavior tables. Technical Report 392, Indiana University Computer Science Department, December 1993. condensed version published in ICCAD95.

⋆[24] M. Esen Tuna, Steven D. Johnson, and Bob Burger. Continuations in hardware-software codesign. In *Proceedings of the International Conference on Computer Design (ICCD)*, pages 264–269. IEEE, October 1994. Also published as Tech Report # 409, Computer Science Department, Indiana University.

⋆[25] 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*, pages 86–89. IEEE, March 1995. Pre-published as IUCS-TR 398 (February 1994).

⋆[26] 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, October 1994.

3 Personnel

People supported by MIP-9208745		
person	*position*	*period(s)*
Steven D. Johnson	PI	3 summer months 1994–96
M. Esen Tuna	RA	1/94–5/94
Kamlesh Rath	RA	1/93–12/93
Jeanette Calvert-Coffrin	RA	8/94–5/95
Wei Li	RA	8/94–12/94
John Zuckerman	RA	6/96–5/97

3.1 Supported personnel

- *Steven D. Johnson*, Principal Investigator.

- *M. Esen Tuna*, Research Assistant. Tuna remains an active, non-resident PhD candidate. He is employed by Derivation Systems, Inc., Carlsbad, CA., a start-up company commercializing this research. His dissertation research topic, behavior tables, is a direct outgrowth of this project.

- *Kamlesh Rath*, Research Assistant. Rath *Kamlesh Rath* received his PhD in January 1995. His dissertation [20] addresses the central research topic in this project. He is currently employed at Phillips Research in New Jersey.

- *Jeanette Calvert-Coffrin*, Research Assistant. Calvert-Coffrin is an inactive PhD candidate. She passed the qualification exam in 1995, but then took an MS degree and relocated to Minnesota. She is working in the computer industry.

- *Wei Li*, Research Assistant. Mr. Li's left the graduate program after a semester of support.

- *John Zuckerman*, Research Assistant. Zuckerman is an active PhD candidate, working in compilers under Kent Dybvig. As an assistant to this project, he designed a foundation for user-interface development in design derivation, using a Scheme based library called *SWL* developed by the Scheme Educational Infrastructure group at Indiana University.

3.2 Affiliated personnel

- *Bhaskar Bose.* Bose completed his PhD in November 1994, under a NASA fellowship (1992–94). He is the principal architect of the DDD transformation system. His dissertation [1] describes the system and a major case study involving formal derivation of the FM9001 microprocessor. After graduation, Bose formed Derivation Systems, Inc. (Carlsbad, CA), in order to commercialize his research. Two other IU students, including Esen Tuna from this research group, have joined that company.

- *Paul S. Miner.* Miner attended IU from 1993 to 1995 on educational leave from NASA's Langley Research Center, where he is in the formal methods group. He is now finishing a dissertation on heterogeneous reasoning in formal system design. He performed case studies integrating DDD with the PVS verification system . He has also formalized the IEEE standard for floating point representations.

- *Kathryn Fisler.* Fisler received an AT&T fellowship (1993-1997) and held several internships there during her doctoral studies. Her research reflects a collaboration between this research group and Jon Barwise's Visual Inference Laboratory. She investigated the mathematical foundations of incorporating various kinds of circuit diagrams in formal/logical reasoning system [9, 8, 6, 7].

- *Shyamsundar Pullela.* After qualifying for PhD candidacy and participating in several case studies [19, 3], Pullela left graduate school to work as verification engineer at HP-Convex Computer Corp., Richardson Texas.

- *Bob Burger* Burger, an NSF Fellow, worked on the Scheme computer derivation [4]. He competed a dissertation in 1997 in the area of compilers.

3.3 Technical/administrative support

Budget lines for technical and/or clerical support were pooled with other soft-money sources and administered by the Computer Science Department. At the time this project was funded, there was not a recorded correlation between grant accounts and services. Funds for this project were assigned to clerical salaries. However, in addition to normal technical support and services, two members of the department's technical staff were substantively involved with the work:

- *Willie Hunt* made numerous engineering contributions to this project's case studies. An extremely gifted electrical engineer, Hunt contributed both design and, more importantly, methodological insight in our formal investigations.
- *Ingo Cyliax* provided computer engineering and CAD support, and also participated directly in case studies, particularly the Scheme-machine effort. He has also supported collaborations between this research group and the department's robotics group.

4 Travel

TRAVEL SUPPORTED (IN WHOLE OR PART) BY MIP-9208745			
dates	*person*	*location*	*purpose*
3/10–3/18/93	Johnson	Oakland, CA	UCB CAD seminar
4/24–4/28/93	Rath	Ottawa, Ca.	CHDL conference
4/25–4/29/93	Johnson	Ottawa, Ca.	CHDL conference
10/2–10/6/93	Rath	Boston, MA	ICCD conference
10/4–10/6/93	Johnson	Boston, MA	ICCD conference
11/6–11/11/93	Rath	San Jose, CA	ICCAD conference
11/6–11/10/93	Johnson	San Jose, CA	ICCAD conference
4/17–4/18/94	Johnson	Boston, MA	ICCD conference
5/18–5/20/94	Johnson	Waterloo, Ca.	HLSS symposium
9/21–9/29/94	Johnson	Frankfurt, Ger.	Codes/CACHE workshop
			EDAC conference
			TPCD conference
4/22–4/25/95	Johnson	Austin, TX	ICCD TPC meeting
			CHDL PC meeting
8/27–9/5/95	Johnson	Chiba, Japan	ACV'95 tri-conference

4.1 Presentations citing MIP92-08745 support

1. Steven D. Johnson. Design derivation with behavior tables. University of Cincinnati, CSECS Department, System Design Group, October 10,

1996.

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

3. Steven D. Johnson. Integrating design and verification environments through a logic supporting hardware diagrams. CHDL'95 [10].

4. Kamlesh Rath. Specification and synthesis of bounded indirection. GLSVLSI'95 [25].

5. Steven D. Johnson. The Scheme Machine: a case study in progress of digital design derivation at system levels. Third NASA/Langley Formal Methods Workshop [12].

6. Steven D. Johnson. Studies of the single-pulser in various reasoning systems. TPCD 94 [16]

7. M. Esen Tuna. Continuations in hardware-software codesign. ICCD'94 [24].

8. Steven D. Johnson. Capturing synchronization specifications for sequential compositions. ICCD'94 [26].

9. Paul Miner. Interaction of formal design systems in the development of a fault-tolerant clock synchronization circuit. RDS'94 [19].

10. Steven D. Johnson. 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.

11. Kamlesh Rath. Behavior tables: a basis for system representation and transformational system synthesis. ICCAD'93 [22]

12. Bhaskar Bose. System factorization in codesign: A case study of the use of formal techniques to achieve hardware-software decomposition. ICCD'93.

13. Kamlesh Rath. Derivation of a DRAM memory interface by sequential decomposition. ICCD'93.

14. Steven D. Johnson. Automatic synthesis of sequential Synchronizations. CHDL'93.

15. Kamlesh Rath. Toward a basis for protocol specification and process decomposition. CHDL'93.

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

5 Acquired Equipment

EQUIPMENT ITEMS PURCHASED UNDER MIP-92-08745			
Date	*Cost*	*Equipment*	*Purpose*
9/94	$3,620	SIMMs, 128Mbytes	laboratory workstation upgrades
10/94	$5,469	One Sparcstation 5 Model 85, 85MHZ Microsparc II Processor	PI's workstation upgrade
5/97	$5300	SIMMs, HD/CD drive, lg. monitor, SM41 CPU module	lab server upgrade