The AAMP5/AAMP-FV Project

Steven P. Miller
Collins Commercial Avionics
Rockwell International
Cedar Rapids, IA 52498 USA
spmiller@pobox.cca.rockwell.com

Mandayam Srivas
Computer Science Laboratory
SRI International
Menlo Park, CA 94025 USA
srivas@csl.sri.com

Software and digital hardware are increasingly being used in situations where failure could be life threatening, such as aircraft, nuclear power plants, weapon systems, and medical instrumentation. Several authors have demonstrated the infeasibility of showing that such systems meet ultra-high reliability requirements through testing alone [1,2]. Formal methods are a promising approach for increasing our confidence in digital systems, but many questions remain on how it can be used effectively in an industrial setting.

This presentation describes a project, formal verification of the microcode in the AAMP5 microprocessor, conducted to explore how formal techniques for specification and verification could be introduced into an industrial process. Sponsored by the Systems Validation Branch of NASA Langley and by Collins Commercial Avionics, a division of Rockwell International, it was conducted by Collins and the SRI International Computer Science Laboratory. The project consisted of specifying in the PVS language developed by SRI [3] a portion of a Rockwell proprietary microprocessor, the AAMP5, at both the instruction set and register-transfer levels and using the PVS theorem prover to prove the microcode correct for a representative subset of instructions.

While this presentation includes a brief technical overview (see [4,5] for a detailed technical discussion), its emphasis is on the lessons learned in using PVS for an example of this size and the implications for using formal methods in an industrial setting. The central result of this project was to demonstrate the feasibility of formally specifying a commercial microprocessor and the use of mechanical proofs of correctness to verify microcode. This is particularly significant since the AAMP5 was not designed for formal verification, but to provide a more than three fold performance improvement, by pipelining instruction execution, while remaining object code compatible with the earlier AAMP2. As a consequence, the AAMP5 is one of the most complex microprocessors to which formal methods have been applied.

Another key result was the discovery of both actual and seeded errors. Two actual microcode errors were discovered and corrected during development of the formal specification, illustrating the value of simply creating a precise specification. Two seeded errors were systematically uncovered while doing correctness proofs. One of these was an actual error that had been discovered after first fabrication but left in the microcode provided to SRI. The other error was designed to be unlikely to be detected by walkthroughs, testing, or simulation.

Several other results emerged during the project, including the ease with which practicing engineers became comfortable with PVS, the need for libraries of general purpose theories, the usefulness of formal specification in revealing errors, the natural fit between formal specification and inspections, the difficulty of selecting the best style of specification for a new problem domain, the high level of assurance provided by proofs of correctness, and the need to engineer proof strategies for reuse.

Many of the costs of the AAMP5 project can be attributed to the overhead of applying an experimental method for the first time. To determine how much these costs can be reduced through reuse of the AAMP5 expertise, Collins, SRI, and NASA are conducting a follow-on project to verify the microcode in the AAMP-FV, a smaller microprocessor design similar to those actually used in autoland systems. A report on the status of this project is also presented.

[1] Butler, R. and G. Finelli, The Infeasibility of Experimental Quantification of Life-Critical Software Reliability, *Software Engineering Notes*, Vol. 16, No.5, pg. 66–76, December 1991.

[2] Littlewood, B. and L. Strigini, Validation of Ultra-High Dependability for Software-based Systems, *Communications of the ACM*, Vol. 36, No. 11, pg. 69–80, November 1993.

[3] Owre, S., J. Rushby, and N. Shankar, PVS: A Prototype Verification System, In Deepak Kapur, Editor, *11th International Conference on Automated Deduction, (CADE)*, pg. 748–752, Saratoga, NY, June 1992, Vol. 607 of Lecture Notes in Artificial Intelligence, Springer-Verlag.

[4] Srivas, M. and S. Miller, Formal Verification of the AAMP5: A Case Study in the Verification of a Commercial Microprocessor, to appear in *Applications of Formal Methods*, Michael G. Hinchey and Jonathan P. Bowen, Editors, Prentice-Hall International Series in Computer Science.

[5] Srivas, M. and S. Miller, *Formal Verification of an Avionics Microprocessor*, to be submitted as a NASA Contractor Report.

Rockwell Avionics

Collins

The AAMP5/AAMP–FV Project

Steven P. Miller

Collins Commercial Avionics
Rockwell International
400 Collins Road NE
Cedar Rapids, Iowa 52498 USA
(319) 337–5149

spmiller@pobox.cca.cr.rockwell.com

Mandayam Srivas

Computer Science Laboratory
SRI International
333 Ravenswood Avenue
Menlo Park, CA 94025 USA
(414) 859–6136

srivas@csl.sri.com

Formal Verification of the AAMP5 Microprocessor

Introduction

- **Assess the Feasibility of Formal Verification for Industrial Use**
  - Participated in the MCC Formal Methods Transition Study (1990–91)
  - Pilots using RAISE for Formal Specification (1992–93)
- **Collaborative Effort**
  - Funded by NASA Langley and Collins
  - Performed by SRI International and Collins (with Assistance from NASA)
- **Formal Verification of the AAMP5 Microcode**
  - Specified Instruction Set (macro) Architecture in PVS (108 of 209 Instructions)
  - Specified Register Transfer (micro) Architecture in PVS
  - Proved Microcode for 11 Instructions Correct using the PVS Theorem Prover
- **Shadow Project**
  - Independent of Traditional Development and Verification Process of the AAMP5

Formal Verification of the AAMP5 Microprocessor

Overview of Presentation

- **Background**
- **Project History**
- **Formal Specification of the AAMP Macroarchitecture**
- **Formal Specification of the AAMP5 Microarchitecture**
- **Proofs of Microcode Correctness**
- **Conclusions**

Background

The AAMP Family of Microprocessors

CAPS–4	1974	Global Positioning System, General Development Model (GPS GDM)
CAPS–6	1977	Boeing 757, 767 Autopilot Flight Director System (AFDS), Lockheed L–1011 Active Control System (ACS), Lockheed L–1011 Digital Flight Control System (DFCS), NASA Fault Tolerant Multiprocessor (FTMP)
CAPS–8	1979	Boeing 757, 767 Electronic Flight Instrumentation System (EFIS), Boeing 757, 767 Engine Instrumentation/Crew Alerting System (EICAS)
CAPS–7	1979	Navstar Global Positioning System (GPS), Boeing 747–400 Integrated Display System (IDS), Boeing 747–400 Central Maintenance Computer (CMC), Boeing 737–300 Electronic Flight Instrumentation System (EFIS), Boeing 737–300 Engine Instrumentation/Crew Alerting System (EICAS), Air Transport Traffic Collision Avoidance System (TCAS), Air Transport TCAS Vertical Speed Indicator (TVI), Boeing 777 Flight Control Backdrive, Commercial GPS: Navcore I, Navcore II, Navcore V
CAPS–10	1979	
AAMP1	1981	
AAMP2	1987	
AAMP3	1992	Boeing 777 Standby Instruments
AAMP5	1993	Global Positioning Systems, Upgrade for AAMP2

Background

AAMP Family of Microprocessors

- Based on Stack Architecture
- Designed for Use in Embedded Systems with High-Level Languages (Ada)
- Multiple Addressing Modes
- Large, CISC-like Instruction Set
- Double Code Density of Most Microprocessors
- Provides Direct Hardware Support for Much of
  - Run Time Environment
  - Real-Time Executive

Rockwell Avionics
Collins

Background

AAMP5

- Object Code Compatible with Earlier AAMP2
- Uses Pipelining to Achieve 3x Performance of the AAMP2
- ~500,000 Transistors
  - ~20,000 Transistors in Other Formal Microprocessor Verification Efforts
  - ~3.1 Million Transistors in an Intel Pentium
- Performance Between Intel 386 and 486
- Intended for Use in Avionics Displays and Global Positioning Systems

Rockwell Avionics
Collins

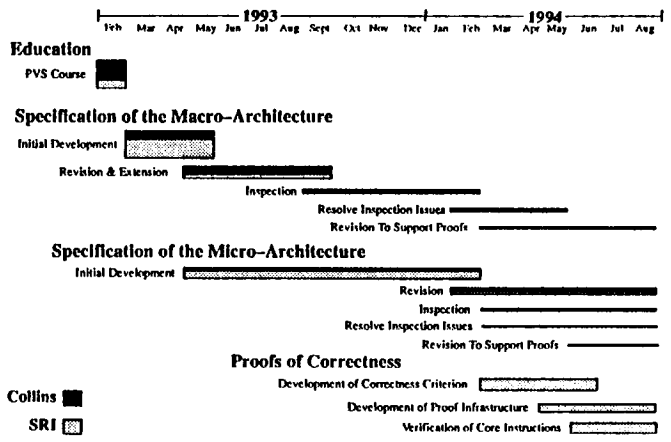
Background

PVS

- Environment for Formal Specification and Verification
- Developed by SRI International
- Expressive Specification Language
  - Simple and Easy to Learn
  - Notation Similar to Conventional Programming Languages
  - Sophisticated Type System
  - Higher Order Logic
- Typechecker
- Interactive Theorem Prover
  - Automates Many of the Low Level Proof Steps

Rockwell Avionics
Collins

Project History



Rockwell Avionics
Collins

Formal Specification of the AAMP Macroarchitecture

- **Initial Development Done by SRI**
  - 1,595 Lines of PVS and Comments Organized into 25 Theories
- **Bit Vectors Library Taken Over by NASA Langley**
  - Evolved into 2,030 Lines of PVS and Comments in 31 Theories
- **Revised and Extended by Collins**
  - Specified most of the Executive Service Routines
  - Completed 108 of the AAMP's 209 Instructions
  - 2,550 Lines of PVS and Comments Organized into 48 Theories
- **Performed 11 Inspections of the PVS Specifications**
  - Practicing Electrical Engineers
  - Inspectors had Little Trouble Reading PVS
  - Average Preparation Rate of ~150 Lines of PVS and Comments/Hour
  - Found 53 Minor (Style) Errors, 28 Major (Correctness) Errors

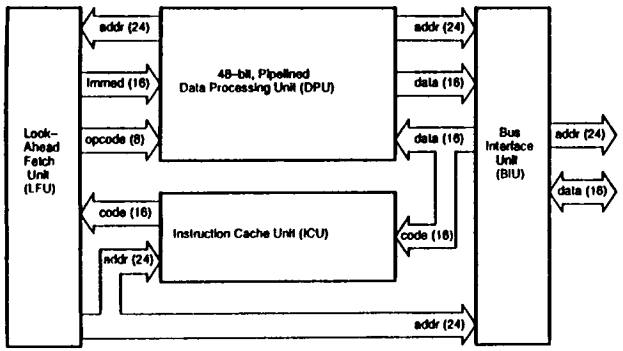
Formal Specification of the AAMP Macroarchitecture

Microcode Errors Found in This Phase

- **Discovered Two Microcode Errors**
  - Both Specific to the AAMP5
  - Both Corrected Before First Fabrication
  - Had Not Completed Traditional Verification Effort
- **Found While Specifying Behavior of the AAMP During Unusual Cases**
- **One was Due to a Missing Requirement**
  - Would Have Been Found During Ada Validation Testing
- **Other Was a Coding Error**
  - Required Specific Stack Configuration, an Improperly Sized Stack, and a Specific Instruction Sequence
- **Illustrates the Value of Simply Creating a Precise Specification**

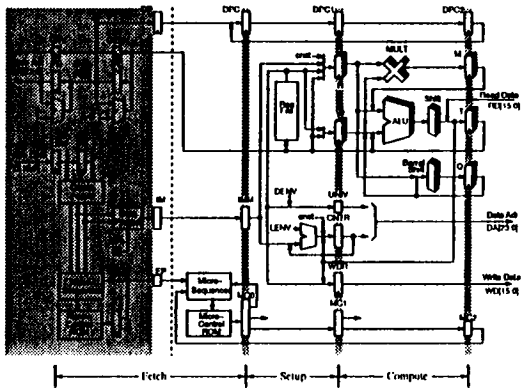
Formal Specification of the AAMP5 Microarchitecture

AAMP5 Block Diagram



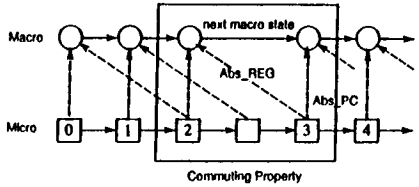
Formal Specification of the AAMP5 Microarchitecture

DPU Microarchitecture



Proofs of Microcode Correctness

Approach and Issues



- Pipelining
- Autonomous Code and Data Access
- Enhancements to Support Efficient Automation
  - Boolean Decision Diagram (BDD) Decision Procedures
  - Optimizations to the PVS Rewriting Engine
- Decomposition of Verification Approach to Facilitate Technology Transfer

Proofs of Microcode Correctness

Microcode Errors Found in This Phase

- Seeded Two Errors in Microcode Provided to SRI
- First was Designed to be Difficult to Detect
- Second was an Actual Error
  - Escaped Detection by Simulations and Walkthroughs
  - Discovered by Collins While Running Application Code on an Early Prototype of the AAMP5
- Both Errors were Found by SRI While Doing the Proofs
  - Discovery was Very Systematic
  - SRI Explained How to Correct Both Errors

Summary

- Specified 108 of the AAMP's 209 Instructions
- Specified the AAMP5 at the Register Transfer Level
- Developed a Large, Reuseable Library of Bit Vector Properties
- Discovered Two Errors in AAMP5 Microcode During Specification
- Conducted Inspections of the PVS Specifications With Design Engineers
- Developed General Approach to Decomposing the Proofs of Correctness
- Automated these Strategies in the PVS Prover
- Formally Verified the Microcode in 11 Instructions
- Discovered Two Seeded Errors during Formal Verification

Conclusions

- Demonstrated the Technical Feasibility of
  - Formally Specifying the AAMP5 at Instruction Set and Register Transfer Levels
  - Formally Verifying the Microcode in the AAMP5
- Benefits of Formal Specification
  - Encourages Clean Abstractions and Interfaces
  - Encourages "Looking in Corners"
- PVS Specifications Successfully Used by Practicing Engineers
  - Synergy between Specifications and Inspections was Key
  - Acceptance of PVS by Engineers Varies Widely
  - Difficult to Enforce the Discipline Needed to Ensure Quality Specifications
- Could Achieve *Dramatic* Gains in Acceptance Through
  - Notations that Fit a Specific Problem Domain

Conclusions

- **Formal Verification, Done Correctly, Provides Very High Levels of Assurance**
  - Does not Eliminate Good Process, Peer Reviews, Testing, Simulation, ...
  - May Facilitate or Lessen the Need for Some Traditional Practices
- **Expect Costs to be High the First Time In a New Problem Domain**
  - Expertise
  - Reusable Theories and Proofs
- **How Much Will Costs Drop on Subsequent Projects?**

AAMP–FV

- **Collaborative Effort**
  - Funded by NASA and Collins
  - Conducted by Collins and SRI
  - Initiated in January, 1995
- **Smaller Microprocessor**
  - Paper and Pencil Design
  - Similar to What We Would Use in an Autoland System
  - ~100,000 Transistors
- **Repeat AAMP5 Experiment**
  - Reuse Expertise and Theories
  - Demonstrate Cost Effectiveness
- **Current Status**
  - Specified Microarchitecture (50 Hours)
  - Nearly Completed the Proof of the First Instruction