Section A

Institution: University of St Andrews

Unit of Assessment: UoA 11: Computer Science and Informatics

Title of case study: Use of Precise Architectural Memory Consistency Models and Validation in Processor Companies

Period when the underpinning research was undertaken: 2013 - 31 December 2020

Details of staff conducting the underpinning research from the submitting unit:

Name(s):	**Role(s) (e.g. job title):**	**Period(s) employed by submitting HEI:**
Susmit Sarkar	Reader	01 February 2013 - present

Period when the claimed impact occurred: 2016 - 31 December 2020

Is this case study continued from a case study submitted in 2014? N

Section B

1. Summary of the impact (indicative maximum 100 words)

Modern day computers, mobile devices and servers are all multicore, which means that they rely on multiple computing hardware units working seamlessly together to deliver high-performance computing power. Programming such combinations of hardware units, and reasoning about programs running on them, is very challenging, particularly due to a phenomenon of multicore hardware called “memory consistency”.

Hardware specifications are subtle and complex, and their compliance is hard to determine. Hardware not conforming to their official specifications, especially with regards to memory consistency, has led to high-profile correctness and performance problems over the years. Usually, these situations arise due to one of the following two reasons. First, the official specifications are imprecise and vague, making them difficult to use. Second, there are no straightforward ways for hardware vendors to test specification conformance, leading to the possible release to market of nonconforming hardware.

Precise specifications, usable by programmers and compiler and code analysis tool designers and incorporating St Andrews’ research on hardware validation and memory consistency models, have been adopted by hardware manufacturers such as ARM and RISC-V as part of their official specifications. A library of “litmus tests” (short test programs) developed by St Andrews to test conformance to those official models is now routinely used within large hardware companies, such as IBM, to test processors before public release, and has identified subtle problems before shipping, saving significant costs. Taken together, they strengthen the assurance of a substantial proportion of computer hardware and the reliability of programs running on them. The importance of this work is far reaching if we recall that ARM chips are used in 90% of smartphones today, with over 160,000,000,000 chips having been shipped to date and an average of 22,000,000,000 over the past three years.

2. Underpinning research (indicative maximum 500 words)

A longstanding research goal in the field of verification has been the creation of precise and formal specifications of computer hardware, and testing conformance of hardware to those specifications. This has been particularly difficult in multicore hardware (as essentially all computers today are): these give rise to poorly understood, and difficult to reason about, problems, such as memory consistency. This is when different threads of execution have different inconsistent views on shared memory resources. This arises from the interaction of hardware implementation features, and even hardware manufacturers, let alone programmers on such hardware, have had gaps in their understanding of memory consistency.

In 2016, we showed how the novel features of the ARM architecture relevant to memory consistency could be made amenable to precise specifications [R1]. This showed how techniques from a different field of computer science (programming languages) could be used to create precise mathematical specifications of hardware, especially of complex memory consistency issues in multicore ARM and similar processors. Furthermore, these specifications could be explained to programmers with the development of prose counterparts of the mathematical specifications. These developed prose specifications are expressed in natural language but express the same concepts as the simultaneously developed mathematical specifications. After that, work was done on mixed-size concurrency [R2], for ARM as well as for POWER, and even x86. This revealed substantial complications not resolved by any earlier work. It showed the techniques developed in R1 could be scaled up to both reveal those previously hidden complications, as well as to suggest a solution to those complications.

The research culminated in a paper [R3] published in 2018, incorporating solutions to the complexities of R1 and R2 above. This work radically simplified and made more accessible the ARM memory consistency models, and was done in collaboration with engineers from ARM. The accessibility of the prose counterparts of the mathematical specifications was found sufficient for the working hardware engineers.

A key innovation of this research was the use of short test programs (called "litmus tests"), many of them developed by Sarkar. These tests, and their intended outcomes, serve as guides to explain the consequences of the formal specification models to programmers and implementors. These were crucial to get insights from the hardware implementors, who could say concretely what the hardware was expected to do on the test cases much better than they could appreciate the mathematical nuances of the specifications. The tests were created by the researchers from the details of the mathematical specifications. The guidance by the hardware implementors in turn would be fed back into nuances of the mathematical specifications. The development of mathematical specifications, the development of prose specification counterparts, and development of the tests exhibiting the specification nuances were all closely intertwined. Papers R1-R3 described above, thus, developed a suite of test programs. These tests could then be used to test conformance with the developed memory consistency models. As explained in the impact section, such tests can be easily adapted to be used within processor companies for testing conformance to the newly created specifications.

3. References to the research (indicative maximum of six references)

All the outputs have been published in international peer-reviewed publications.

[R1] Shaked Flur, Kathryn E. Gray, Christopher Pulte, Susmit Sarkar, Ali Sezgin, Luc Maranget, Will Deacon, and Peter Sewell. "Modelling the ARMv8 Architecture, Operationally: Concurrency and ISA". In: *Proceedings of the 43rd Annual SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'16)*. Ed. by Rastislav Bodík and Rupak Majumdar. New York, NY, USA: ACM (2016), pp. 608–621. ISBN: 978-1-4503-3549-2. DOI:10.1145/2837614.2837615.

[R2] Shaked Flur, Susmit Sarkar, Christopher Pulte, Kyndylan Nienhuis, Luc Maranget, Kathryn E. Gray, Alan Sezgin, Mark Batty, and Peter Sewell. "Mixed-size Concurrency: ARM, POWER, C/C++11, and SC". In: *Proceedings of the 44th ACM SIGPLAN Symposium on Principles of Programming Languages, (POPL 2017).* Ed. by Giuseppe Castagna and Andrew D. Gordon. New York, NY, USA: ACM (2017), pp. 429–442. ISBN: 978-1-4503-4660-3. DOI:10.1145/3009837.3009839.

[R3] Christopher Pulte, Shaked Flur, Will Deacon, Jon French, Susmit Sarkar, and Peter Sewell. "Simplifying ARM Concurrency: Multicopy-atomic Axiomatic and Operational Models for ARMv8". In: *Proceedings of the ACM on Programming Languages (PACMPL)* 2.POPL (2018), 19:1–19:29. ISSN: 2475-1421. DOI: 10.1145/3158107.

4. Details of the impact (indicative maximum 750 words).

Precise memory consistency models based on St Andrews Research have been incorporated into the official architecture specifications: ARM Architecture Reference Manual (2016 and subsequent versions) [S1, section B2.3, pp. B2-115-132]; RISC-V ISA specification [S2, Chapter 8, pp. 47-52 and Chapter 14, pp. 81-92, and Appendix A&B, pp. 161-217]; and (in-process) IBM POWER Architecture Reference. For the last, insights from the research [R1-R3] above is planned to be incorporated into the manual following a similar path to ARM; this work was delayed by the Covid-19 pandemic but is still planned to be done shortly [S3, 3rd para]. Furthermore, "litmus tests" developed at St Andrews are now routinely used within hardware companies to test processors' conformance to the specifications before shipping; this has caught several problems before shipping resulting in what IBM describes as "*saving significant costs*" [S3, 2nd para]. This increases the reliability and assurance of a large proportion of currently sold processors: ARM processors have 90% market share in mobile devices [S4, slide 12] and sold over 6,400,000,000 processors in the 4th quarter of 2019 [S5, p. 1]; POWER-based server systems reported revenues of just over USD4,000,000,000 in the 4th quarter of 2019 [S6, p. 2].

Context: the problems

The problem of reducing and removing "memory consistency" bugs is a major concern with "multicore" machines, where several computing units work together to deliver high-performance computing power. All current processors in mobile devices and cloud computers (since about 2010) are indeed multicore. Memory consistency problems have in the past led to high-profile performance problems. For example, in 2008 AMD disclosed correctness problems ("TLB bug") that necessitated significant downgrades in performance, and in 2013 Samsung processors had a significant performance problem ("CCI 400 bug") that led to a line of hardware chips once talked of as Samsung's "flagship" never becoming widely used.

Specifications incorporating research insights adopted by ARM (95% market share of mobile devices), RISC-V, and soon IBM (15% market share in cloud servers)

Research results [R1-R3] above have been incorporated into the ARM Architecture Reference Manual, the official specification of the ARM Architecture since 2016 (v8.3). Section B2.3 of the current manual [S1] has a description directly derived from work done in [R3], and ARM maintains the formal memory consistency model first done in that work [R3] as the official formal specification [S7]. This work was done in collaboration with ARM, and ARM senior engineer Will Deacon was a co-author of [R3].

The RISC-V architecture specification [S2] has a memory model based on the work of [R1-R3] but modified significantly for RISC-V. This version was ratified in 2019 and acknowledges Sarkar by name (on p. 2). Sarkar was invited to be the member of the experts working group on RISC-V memory consistency model, and contributed his expertise based on his research [R1-R3]. RISC-V is an international effort started in 2013 to create a new open-source line of processors. Key members of the RISC-V consortium are UC Berkeley, MIT CSAIL, Google, Samsung, Qualcomm, NVIDIA, BAE Systems, ETH Zurich, IBM, Huawei, Seagate. The RISC-V Memory Consistency Model Experts Group did its work between late 2016 and 2018 and wrote Sections 8 and 14 of the specification [S2].

IBM is considering a significant simplification of its memory consistency model and incorporating formal models into the POWER Architecture specification, similar to the work ARM has undertaken [S3, 3rd para]. This work was supposed to happen this year but has been delayed by the COVID-19 travel bans [S3, 3rd para].

The availability of such precise formal models as official specifications is important to all programmers of these hardware. To take an example, programming language and compiler designers can now specify what concurrent programs in, for example C++, can do on ARM or POWER [S8, see discussion on p1 and forwards]. As another example, operating systems developers such as the Linux kernel development team can now reason about and fix "infamous" bugs in Linux [S3, 5th para].

Our suite of "litmus tests" is used to test processors by IBM

Once precise specifications are available, conformance of hardware to those specifications can be tested by running tests. Papers [R1-R3] had associated with them a library of such tests, called “litmus tests”. These tests, and their intended outcomes, serve as guides to explain the consequences of the formal models to programmers and implementors. They can also be used to test the conformance of the hardware to the intended consistency model guarantees. This is of huge benefit to hardware designers and implementors, since modern hardware is very complex and very hard to get right. The “litmus” toolchain developed primarily at INRIA with expert input from Sarkar is still actively developed (http://diy.inria.fr/doc/litmus.html).

This is used to run tests devised as part of the research [R1-R3] above, and has been in regular use within IBM, in the IBM POWER division. This has been used on several generations of POWER hardware. While the details are commercially sensitive, several problems were caught and, therefore, fixed before public release in different unreleased hardware [S3, 2nd para]. This has resulted in what IBM calls “*saving significant costs*” [S3, 2nd para]. The details are naturally commercially sensitive, but we can get an idea of the potential savings by considering the analogous Samsung example when a line of processors described in the press as “[*Samsung’s*] *next flagship line of processors*” to never achieve widespread use. Recall that IBM Power represents reported 15% market share in cloud servers, with reported revenues of just over USD4,000,000,000 in Quarter 4 of 2019 [S6, p. 2].

Collaborations:

The underpinning research was done in a close collaboration of three research groups: Sarkar at St Andrews, Sewell at Cambridge, and Maranget at INRIA. Sarkar and Sewell worked co-equally on development of the precise models and development of litmus tests. This leveraged Sarkar’s prior expertise in memory consistency models on other architectures such as X86 (Intel and IBM Power). Maranget worked on the development of litmus tests and running them on hardware.

Conclusion:

In summary, we quote a Senior Engineer at IBM: “*the industrial impact of Susmit’s work in a variety of areas in formal modeling of modern processor multiprocessor concurrency is outstanding, […] I’ve never seen more significant industrial impact*”. [S3, 6th para]

5. Sources to corroborate the impact (indicative maximum of ten references)

[S1] ARM Architecture Reference Manual, v8.5E (available at https://static.docs.arm.com/ddi0487/ea/DDI0487E_a_armv8_arm.pdf) Section B2.3 Definition of the ARMv8 Memory Model, pp. B2-115

[S2] The RISC-V Instruction Set Manual Volume I: Unprivileged ISA (available at https://riscv.org/specifications/)

[S3] Senior Engineer, IBM Austin POWER division

[S4] “Worldwide Server Market Revenue Grew 7.5% Year Over Year in the Fourth Quarter of 2019, According to IDC :https://www.businesswire.com/news/home/20200312005798/en/Worldwide-Server-Market-Revenue-Grew-7.5-Year-Over-Year-in-the-Fourth-Quarter-of-2019-According-to-IDC

[S5] The ARMv8 Application Level Memory Model: https://github.com/herd/herdtools7/blob/master/herd/libdir/aarch64.cat

[S6] ARM Limited roadshow slides 2019 https://www.arm.com/-/media/global/company/investors/PDFs/Arm_SBG_Q1_2019_Roadshow_Slides_FINAL.pdf

[S7] Record shipments of Arm-based chips in previous quarter: https://www.arm.com/company/news/2020/02/record-shipments-of-arm-based-chips-in-previous-quarter

[S8] Revising the C++ memory model (Working paper WG21/P0668R5 of ISO WG21: Programming Language C++) http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2018/p0668r5.html