The Formal Verification of an ATM Network[1]

Paul Curzon
University of Cambridge
Computer Laboratory,
United Kingdom

Communication networks are rapidly becoming all pervasive. As this occurs, the consequences of errors in the design or implementation of network components becomes increasingly important. This is especially so if, as is increasingly probable, networks are used in safety-critical applications where communication problems could cause loss of life. Asynchronous Transfer Mode (ATM) is a relatively new technology that is being adopted by both the computer and telecommunication industries. It is likely to be the most important transfer mode of the foreseeable future. It is being touted as a technology that can be used "everywhere": in wide-area, metropolitan area, local area and even desk area networks [4]. ATM systems could become high-volume products for which high dependability is paramount. It is an important application for formal verification research.

The ATM Verification Project at Cambridge is investigating the use of formal methods, and in particular the HOL system [2], to validate an implementation of an ATM Network. The network under consideration is Fairisle [3]. It is a working network, carrying real user data. It was designed and implemented with no thought for formal verification. It provides a realistic case study for the investigation of the formal verification of an ATM network.

Initially, we are verifying the switch hardware using conventional machine-checked formal hardware verification techniques. We have formally verified a gate-level implementation of the Fairisle 4 by 4 switching fabric [1]. It forms the heart of the switch. It does the actual switching of cells from inputs to outputs and arbitrates cell clashes using a combination of priority filtering and round-robin arbitration. Routeing and arbitration decisions are based on information in a header tagged on to each cell before it is injected into the fabric.

Despite the need for a significant amount of reverse engineering, the formal specification and verification of the fabric was completed in a time-scale roughly similar to that originally spent designing, implementing and testing it. No errors were found in the fabricated design. However, an undocumented assumption about its environment was discovered during the formal verification. Errors were also found in the formal specifications which had been reverse-engineered from the implementation.

We are currently verifying the Fairisle 16 by 16 delta fabric. It consists of switching elements that are variations on the 4 by 4 fabric design. The 4 by 4 fabric took several months to verify. The variations were verified in a matter of days. This illustrates that formal proof can quickly track design changes of this kind. Effort invested in verifying a design need not be lost if the design is altered.

After verifying the switch hardware, we intend to verify the signalling protocol used by Fairisle. We will formally link this proof with the proofs of the hardware components. Our long-term aim is to perform a hierarchical machine-checked formal verification of a complete network.

[1] In Proceedings of the 1994 ACM Symposium on Principles of Distributed Computing

REFERENCES

[1] P. Curzon. The formal verification of the Fairisle ATM switching element: An overview. Technical report, University of Cambridge Computer Laboratory, 1994.

[2] M. J. C. Gordon and T. F. Melham. *Introduction to HOL: A Theorem Proving Environment for Higher-order Logic*. Cambridge University Press, 1993.

[3] I. M. Leslie and D. R. McAuley. Fairisle: An ATM network for the local area. *ACM Communication Review*, 19(4), September 1991.

[4] I. M. Leslie, D. R. McAuley, and D. L. Tennenhouse. ATM everywhere? *IEEE Network*, March 1993.