Model-Checking the Architectural Design of a Fail-Safe Communication System for Railway Interlocking Systems

Bettina Buth[1] and Mike Schrönen[2]

[1] BISS, Bremen Institute for Safe Systems,
bb@informatik.uni-bremen.de
[2] Department of Electrical Engineering, University of Cape Town,
mschronen@eleceng.uct.ac.za

The design and development of safety-critical systems requires particular care in order to ensure the highest level of confidence in the systems. A variety of life-cycle models and development standards have evolved in various areas. Formal methods are touted to be the best approach for the development on all levels. Up to now, the lack of adequate tools, the lack of knowlegde on the developers side, and the well-known problems of scalability have prevented a migration of these methods into industries.

Schrönen proposes a methodology for the development of microprocessor based safety-critical systems which takes into account the state-of-the-art methods and guidelines for this specific field. The use of formal methods for the verification of the overall design as well as the specification of tests is proposed and demonstrated, using the development of a fail-safe data transceiver (FSDT) for Transnet, South Africa, as a case study. Here we report on the validation of the system architecture based on CSP specification and refinement. The model-checker FDR2 was used as a tool for this task.

The validation was a joint effort of the two authors, the one being the developer of the transceiver, the other an experienced user of FDR2 and thus a prototypical setting for industrial projects: formal methods specialists take part in the system development by supporting the modelling and validation process. Experiences are positive in this respect, but also support the claim that it is not possible in general to perform this kind of validation without cooperation with the developer of a system.

On the technical side, experiences show that while it is possible to use FDR2 as a validation tool, there are some aspects of the case study which are not easily captured in the CSP model. This is true for timing properties both for the environment and internal actions as well as for the atomicity of internal actions. In both cases, semaphores and flags were used for modelling on the CSP side.

The overall result of the work so far is positive even with the problems mentioned above. We plan to continue the cooperation for further investigation of the FSDT, starting with investigations related to exceptional behaviour such as corrupted and lost data. In general it would be interesting to investigate how far other model-checking tools as for example the SPIN tool, allow an easier approach to modelling the system and formulating the desired properties.

J. Wing, J. Woodcock, J. Davies (Eds.): FM'99, Vol. II, LNCS 1709, pp. 1869–1869, 1999.