Synchronous Techniques for Software and Hardware Embedded Systems

Gérard Berry
Esterel Technologies, Villeneuve-Loubet, France

The lectures present the synchronous programming approach to embedded systems design, based on specific languages and programming and verification environments. Born in research in the early 80's and first industrialized in the 90's, the synchronous approach is now widely used for the design and verification of embedded systems in two main application areas: on one hand, safety-critical embedded software in avionics, railways, heavy industry, and automotive applications; on the other hand, high-level synthesis of Systems-on-Chips components. Although the involved industries are vastly different in their development flows and application constraints, they deal with the same basic objects: control-theory or signal-processing data-flow networks, and finite state machines. Synchronous languages are dedicated to these two major design components and to their integration in a common hierarchical and concurrent framework.

We first describe the synchronous cycle-based approach to embedded systems design, based on the Esterel, Lustre, and Scade 6 languages. The key is the use of synchronous parallelism, which scales up to large applications while remaining deterministic by construction, unlike classical asynchronous parallelism. We describe the mathematical semantics of the languages, which is central to their internal correctness and to the safety of their compilation and formal verification. We discuss the compiling technology, with software translation for Lustre / Scade and hardware synthesis for Esterel. We present the formal verification technology for synchronous programs, based on BDD and SAT techniques.

We then discuss integration in industrial flows and actual applications. For software applications, the Scade 5 compiler has been shown certifiable at level A of the DO-178B avionics norm, which greatly simplifies embedded software certification: the generated code can be embedded without being unit-tested. The new Scade 6 compiler is being certified in the same way. We discuss applications to a number of major avionics programs, including the Airbus A380, and applications to railways and automotive software. For hardware application, Esterel v7 supports extensive formal verification, generation of synthesizable VHDL / Verilog code, and generation of C / SystemC models. We discuss applications to networking and handset chips design.

The standard synchronous approach is dedicated to compact non-distributed embedded systems. We discuss ongoing extensions towards distributed software systems and large System-on-Chips, in conjunction with asynchronous technologies.

References

1. D. Potop-Butucaru, S. Edwards, G. Berry. *Compiling Esterel*. In Circuits and Systems Collection"; Springer; 2007.
2. G. Berry. *SCADE : Synchronous Design and Validation of Embedded Control Software*. In "Next Generation Design and Verification Methodologies for Distributed Embedded Control Systems"; Proc of the GM R&D Bangalore Workshop; S. Ramesh and P. Sampath (eds.); pp 19–33; Springer; 2007.

3. J.-L. Colaço, B. Pagano, and M. Pouzet. *A Conservative Extension of Synchronous Data-flow with State Machines*. In Proc of the ACM International Conference on Embedded Software (EMSOFT'05); New Jersey, USA; 2005.

4. A. Benveniste, P. Caspi, S.A. Edwards, N. Halbwachs, P. Le Guernic, and R. de Simone. *The Synchronous Languages 12 Years Later*. In Proc of the IEEE; Vol. 91, Nr. 1; 2003.

5. G. Berry. *The Semantics of Pure Esterel*. Web book; 2002. `http://www.esterel-technologies.com/company/management-team/gerard-berry.html`

6. G. Berry, A. Bouali, R. de Simone, X. Fornari, E. Ledinot, and E. Nassor. *Esterel: a Formal Method Applied to Avionic Software Development*. Science of Computer Programming 36; pp 5–25; 2000.

7. G. Berry. *The Foundations of Esterel*. In "Proof, Language and Interaction: Essays in Honour of Robin Milner"; G. Plotkin, C. Stirling and M. Tofte (eds.); MIT Press; 2000.

8. N. Halbwachs, F. Lagnier and C. Ratel. *Programming and Verifying Critical Systems by Means of the Synchronous Data-flow Language Lustre*. In IEEE Transactions on Software Engineering, Special Issue on the Specification and Analysis of Real-Time Systems; 1992.