

Abstract

Orlic, B., "*SystemCSP: A graphical language for designing concurrent component-based embedded control systems*", p. xi+237, ISBN 978-90-365-2573-2, 2007 (Supervised by van Amerongen, J. and Broenink, J.F.)

Realization of embedded control systems is a complex task. Increasing part of this complexity is nowadays located in the design and implementation of software that runs them. A major source of difficulties is the limitation of the average software developer to understand and design complex behavioral scenarios. A big part of this complexity comes from the interaction of concurrently existing entities.

Similar to the way energy-flow based modeling is used in bond-graph theory as a common view in multi-domain physical system modeling, in our approach concurrency is viewed as a glue layer that relates different domains and views. Concurrency used in a structured way should lead to systems that are simple to design and understand, easy to distribute, reconfigure and reuse.

The aim of this thesis is to develop a graphical design specification language that can offer a structured way of handling concurrency. A structured way of using concurrency is expected to raise the abstraction level away from concurrency problems. In that way, the introduced design specification language should become a vehicle for reducing complexity in inherently concurrent systems (e.g. complex control systems).

Our approach is not to invent the wheel all over again. Instead, we want to build upon the existing body of knowledge provided by relevant formal methods.

SystemCSP is based on the principles of both component-based design and CSP process algebra. It is applicable for specifying, documenting, visualizing and formal verification of component-based designs. It provides a way to visualize architecture, behavioral patterns of components, intra- and inter-component interactions and execution relations among components.

First, the elements of the language are introduced. Then, this core set of language elements is extended with elements dedicated to the specification of time properties. The thesis also offers some insights and some possible approaches for analysis and implementation of real-time systems. The core of the notation is further extended by introducing a set of higher-level primitives in the form of reusable design patterns (see Figure 1.6). These patterns can be used in designs as basic building blocks, extending in that way the vocabulary and raising the level of abstraction of design processes.

The usability of the notation for specifying interactions in embedded control systems is tested by designing software for a complex control setup consisting of several cooperating devices. This test case is a step towards industrial-strength setups.

SystemCSP seems to be a convenient way to specify interactions in concurrent component-based embedded control systems.