College Specification and analysis of Embedded Systems, 2R770, Spring 2003

Jan Friso Groote (J.F.Groote@win.tue.nl, www.win.tue.nl/~jfg, 040-247 5003/5010)

Goal

This course provides a number of techniques to model and analyse interacting, parallel systems, finding their roots in process algebra and logic. These techniques can be used to master the more complex problems of embedded system design: to understand the problem for which an embedded system is to be made and to understand how cooperation of individual systems does actually yield a solution for this problem. This is a field heavily under research at the moment. The course does address the research that is currently going on, while in the meantime students are working on the specification of an embedded system application that must meet certain quality standards.

Course material

There will be handouts during the course, including the following.

- Modelling Concurrent Systems: Protocol Verification in μCRL. W.J. Fokkink, J.F. Groote and M.A. Reniers. (draft).
- A.G. Wouters. Manual for the μCRL tool set (version 2.8.2). Technical report SEN-R0130, December 2001.

Prerequisites

Process algebra 2M920. Preferably: Automated Reasoning (2R880).

Examination

Students are asked to design tray transport system comparable to such a system designed and built by Nyquist. In cooperation with the teacher groups of students must develop a plan to model and analyse this embedded system. The groups can have up to four participants. An important aspect in this modelling phase is to select the right level of modelling. If the model is too detailed it might be too complex to gain insight. If the model is too simple, not all adequate aspects of the real system and its requirements can be studied. The result must be properly documented in the form of a technical report, showing that the design meets the quality standards, which have to be set a priori.

The requirements for the design will be formulated by Nyquist during a visit for which a date must still be set (preferably during the week of April 14). A sketch of the design must be handed in before the week starting at April 28, including a full set of requirements that the design must satisfy. At May 19 a fully formalized design must be finished. On June 23 a technical report must be handed in, including a full analysis that the design meets its requirements. At the end of June or at the beginning of July a final discussion about the work will take place.

Weekly content (indicative)

March 31 Introduction into μCRL. Syntax, axioms and CL-RSP. Data types. Rewrite rules.

April 7 No course

April 14 Weak and branching bisimulation. Elementary calculations and specifications.

April 28 Linear Process Operator. Greibach Normal Form. Linearising μCRL. Optimisation techniques for LPOs.

May 19 Toolset of μCRL. State space generation. Visualisation of large data spaces. Optimisations of linear process operators (constelm, parelm, structelm).

May 26 Confluence; EQ-BDDs. Local confluence.

June 2 Modal logic. Specification of properties and analysis of algorithms.

June 16 Cones and Foci theorem. Firewire configuration protocol. Hef system. Bounded Retransmission Protocol (RC6), P1394. Euris.