

## Tutorial 4

### System Software for Embedded Applications

**Krithi Ramamritham**

*Kanwal Rekhi School of Information  
Technology,  
Indian Institute of Technology, Bombay  
India*

*Tel: +91-22-25767900/01*

*Fax: +91-22-25720022*

*Email: krithi@it.iitb.ac.in*

**Kavi Arya**

*Kanwal Rekhi School of Information  
Technology,  
Indian Institute of Technology, Bombay  
India*

*Tel: +91-22-25767909/01*

*Fax: +91-22-25720022*

*Email: kavi@it.iitb.ac.in*

**Co-author (but will not be presenting)**

**Gerhard Föhler**

*Malardalen University,  
Dept. of Computer Science, Vdsteres,  
Sweden*

*Tel: +46-21 10 31 58*

*Fax: +46-21 10 31 10*

*Email: gerhard.fohler@mdh.se*

**Contact author:**

**Krithi Ramamritham**

#### Abstract

This tutorial focuses on the development of systems software for embedded applications. We begin with a discussion of the software structure for embedded systems from processor specific instructions, operating system, middleware and application layers, and then delve into issues of real-time embedded applications.

The rest of the tutorial will focus on two specific topics: language issues and resource management issues. We overview three high-level (new) languages for designing embedded systems: Esterel which is good for control applications, Handel-C which is esp. good for data-flow (e.g., DSP) applications, Lava which allows also us to verify systems within the same framework.

We then discuss resource Management and scheduling paradigms based on static priorities, static schedules, dynamic scheduling, and best effort approaches. To make the concepts concrete, examples of OSs for embedded systems are presented, with a critical evaluation of Real-Time Linux as an embedded operating system. The possibility of developing real-time embedded

systems without using an OS e.g., by "synthesizing" the OS capabilities into the s/w code will also be probed. A digital camera case study is used to illustrate the concepts.

### **Detailed outline:**

1. Introduction to Embedded Systems, hardware/software co-design, issues in deciding where to split the problem, sensors and interfacing techniques.

As a way to motivate some of the issues underlying software design, we illustrated how embedded applications are characterized by a number of cross cutting issues: control, sequencing, and signal processing and resource management. This sets the stage for identifying the required special real-time services/capabilities and non real-time (e.g. functional) requirements.

2. Detailed examples of embedded systems: digital camera, injection moulding process, flight simulator.

3. Programming Languages for Embedded Systems: - tools for building re-configurable embedded systems - with case studies. Handel-C, for casting algorithms into re-configurable hardware, Embedded Software Development Methodology.

Handel-C is a language based on ANSI-C, extended with concepts for timing, concurrency, flexible-width variables and resource allocation to let software engineers and hardware designers quickly implement complex algorithms efficiently in hardware.

4. Resource Management and scheduling paradigms based on static priorities, static schedules, dynamic scheduling, and best effort approaches. Current best practice in scheduling (e.g., Rate Monotonic vs. static schedules) and communication alternatives are covered and challenges posed by real-world issues such as blocking, unpredictability, interrupts, and caching are introduced and extant solutions compared. To make the concepts concrete, examples of OSs for embedded systems are presented, with a critical evaluation of Real-Time Linux as an embedded operating system.

5. The possibility of developing real-time embedded systems without using an OS e.g., by "synthesizing" the OS capabilities into the s/w code will also be probed.

### **Biography**

Prof. Kavi Arya received a Ph.D. in Computer Science from the University of Oxford and then worked with the Animation Workstation Group at IBM TJ Watson Research Centre (NY/USA). After a stint with the Tata Research Development and Design Centre (Pune, India) and a period in industry, he joined Kanwal Rekhi School of Information Technology in Jan.2000. His interests include formal methods, functional programming languages, graphics and embedded systems. Since '99 he has been particularly interested in exploring new software paradigms for embedded system design and particularly the new breed of languages such as Handel-C, Esterel and Lava.

He has been responsible for setting up a Configurable Systems Lab in KReSIT for exploring the use of these technologies.

Prof. Ramamritham received the Ph.D. in Computer Science from the University of Utah and then joined the University of Massachusetts. He is currently at the Indian Institute of Technology Bombay as the Vijay and Sita Vashee Chair Professor in the Department of Computer Science and Engineering. He was a Science and Engineering Research Council (U.K.) visiting fellow at the University of Newcastle upon Tyne, U.K. and has held visiting positions at the Technical University of Vienna, Austria and at the Indian Institute of Technology Madras. Ramamritham's interests span the areas of real-time systems, transaction processing in database systems, and real-time databases systems. He is applying concepts from these areas to solve problems in embedded systems, mobile computing, e-commerce, intelligent internet, and the Web. Prof. Ramamritham is a Fellow of the IEEE and a Fellow of the ACM. His conference chairing duties include the Real-Time Systems Symposium — as Program Chair in 1994 and as General Chair in 1995, the Conference on Data Engineering — as a Vice-Chair in 1995 and 2001 and as a Program Chair in 2003, and the Conference on Management of Data — as Program Chair in 2000. He has also served on numerous program committees of conferences and workshops. His editorial board contributions include IEEE Transactions on Knowledge and Data Engineering, IEEE Transactions on Parallel and Distributed Systems, IEEE Internet Computing, the Real-Time Systems Journal, the WWW Journal, the VLDB Journal, and ACM SIGMOD's Digital Review. He has co-authored two IEEE tutorial texts on real-time systems, a text on advances in database transaction processing, and a text on scheduling in real-time systems.

Gerhard Fohler is Professor and leader of the predictably flexible real-time systems group at SDL. He received his Ph.D. from Vienna University of Technology in 1994 for research towards flexibility for offline scheduling in the MARS system. He then worked at the University of Massachusetts at Amherst as postdoctoral researcher within the SPRING project. During 1996-97, he was a researcher at Humboldt University Berlin, investigating issues of adaptive reliability and real-time. Gerhard Fohler is currently chairman of the Technical Committee on Real-Time Systems of EUROMICRO.