

Quantitative Aspects of **Embedded Systems**

Dagstuhl seminar 07101, March 5-9, 2007

Organizers:

Boudewijn Haverkort (UT), Lothar Thiele (ETHZ)
Joost-Pieter Katoen (RWTH)

Presenter:

Holger Hermanns (UdS)

Overview

- Some statistics
- Seminar topic
- Set-up of seminar
- Results of seminar

Some Statistics

- 30 half-hour talks
- 3 tutorials (full hour each)
- working group sessions
- 2 WG report sessions
- unfortunately, quite a few last-minute cancellations (from France, USA and Sweden)
- about 38 participants from:
 - Netherlands (8)
 - Germany (15)
 - Belgium (1)
 - Sweden (2)
 - Denmark (3)
 - Great-Britain (2)
 - Korea (2)
 - USA (2)
 - Austria (1)
 - Switzerland (1)
 - Russia (1)

Embedded Systems?

- Analysing or controlling computer systems that are not at hand
 - because they are embedded in a physical environment
 - because they are not existing and not so easy to play with
- Usually systems that are safety-critical, expensive, and/or numerous.

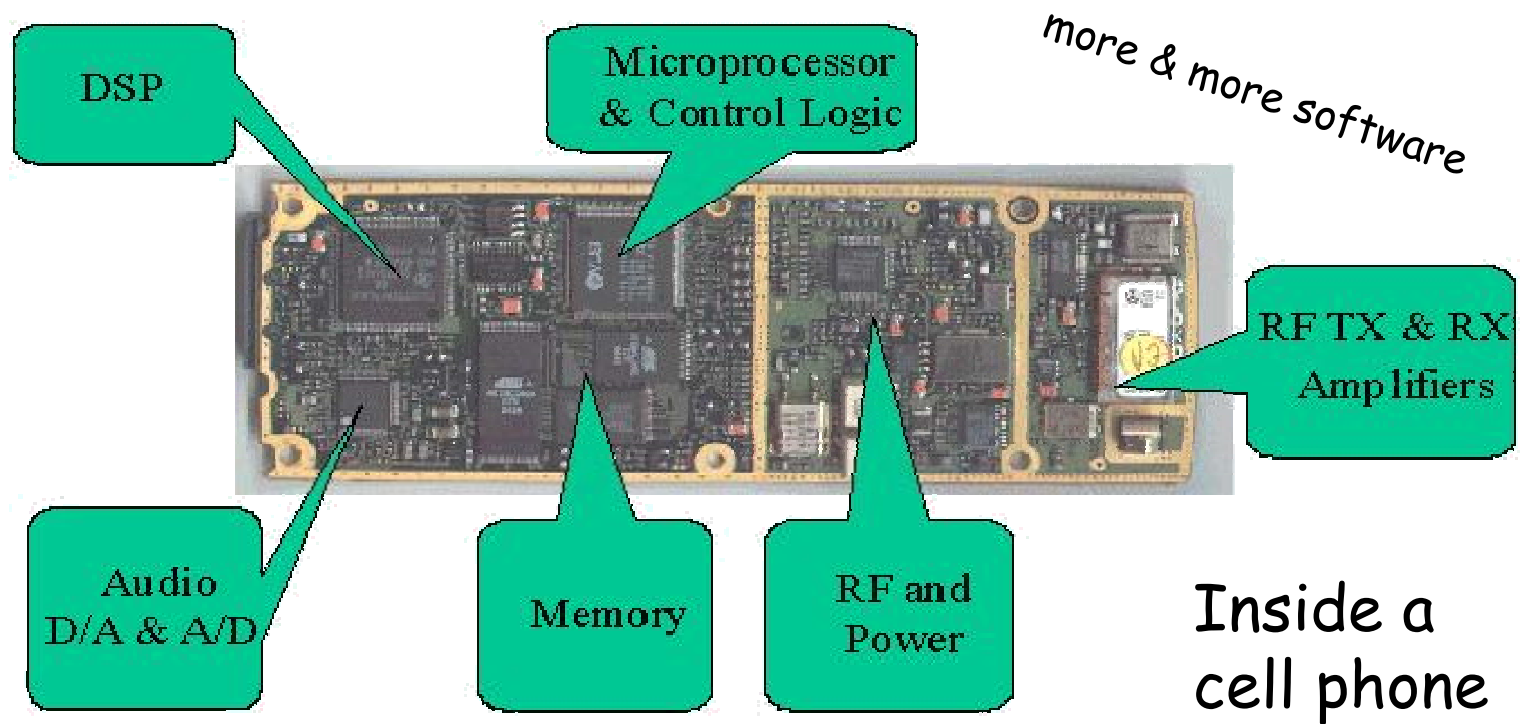




What is the Seminar About? (I)

- Systems that form an integrated part of technological systems and determine the well-operation and quality of such systems
(cars, airplanes, factory control)
- They have to react correctly and timely upon impulses from their environment
- Much more than “classical ICT,” embedded systems’ correctness also implies “timing-correctness”
- Ressource efficiency is a key design goal
- Requires HW/SW co-design, in which verification, performance, dependability and power-constraints need to be taken into account

Embedded Systems



Concurrency:

- component-based design
- system interacts with environment

Heterogeneity:

- digital and analog components
- discrete-time and real-time interaction

The World as We Know It

We do not consider how fast a processor runs when we evaluate whether it is “correct.”



Salvador Dalí, *The Persistence of Memory*, 1931.
(detail)

This is Sometimes Useful For

- Programming languages
- Virtual memory
- Caches
- Dynamic dispatch
- Speculative execution
- Power management (voltage scaling)
- Memory management (garbage collection)
- Just-in-time (JIT) compilation
- Multitasking (threads and processes)
- Component technologies (OO design)
- Networking (TCP)

But Time Sometimes Matters



Kevin Harvick winning the Daytona 500 by 20 ms, February 2007.

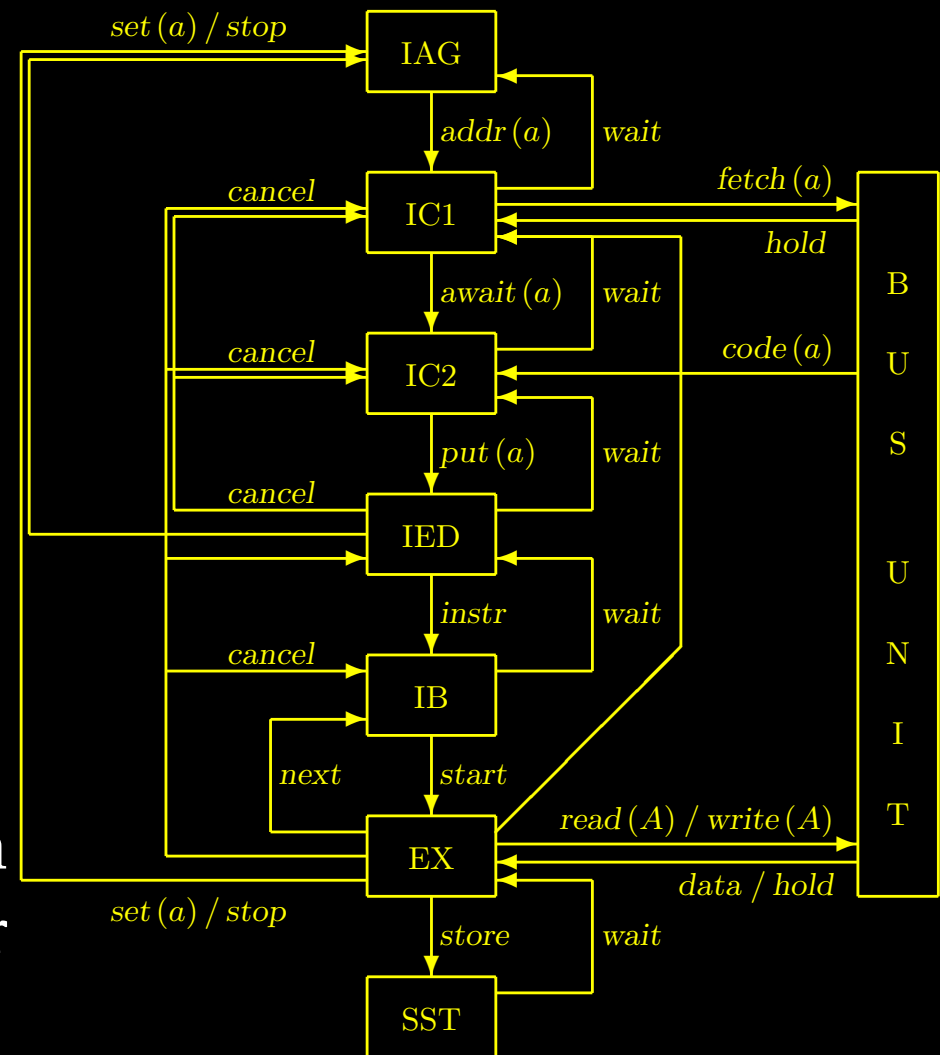
Certification

- Is rather expensive
- Software is *not* certified
- Entire system is certified
- Slight change, e.g., in the microprocessor, requires recertification
- Solution: stockpile parts; trust nobody



WCET on a Motorola ColdFire

- Two coupled pipelines (7-stage)
- Shared instruction/data cache
- Artificial example supplied by Airbus
- Twelve independent tasks
- Simple control structures
- Cache/Pipeline interaction lead to large integer linear programming problem



C. Ferdinand et al., "Reliable and precise WCET determination for a real-life processor," EMSOFT 2001.

The Problem

Digital hardware provides extremely precise timing



and architectural complexity discards it.

What is this Seminar About? (II)

- *Intrinsically multi-disciplinary*: computer architecture, software/programming, verification, compilers, networking, performance evaluation, real-time systems, power-control, control theory, fault-tolerance, hybrid systems.
- Bringing together experts from these areas
- Thereby focusing on the integration of quantitative methods in the design process, so as to assure correctness in the broad sense

Set-Up of Seminar

- homework (preparation): paper reading
- regular seminar with talk by participants
- mixed with working groups (and reports)
- session on teaching embedded systems

Homework papers

All participants were asked to study the following papers (for which we planned presentations):

- T. Henzinger, J. Sifakis: The embedded system design challenge, *Formal Methods, 2006*
- J.A. Stankovic, I. Lee, A. Mok, R. Rajkumar: Opportunities and obligations for physical computing systems, *IEEE Computer, 2005*
- R. Wilhelm, L. Thiele: Design for timing predictability, *Real-Time Systems, 2004*

Working Groups (I)

- Monday, four thematic WG were formed in the morning, upon suggestions from participants
- Before dinner: quick WG reports
- Reports showed, however, too mono-disciplinary approach (“in breed”)
- Organizers decided to form 4 new WG:
 - Use of interfaces in/for embedded system design
 - Worst-case vs. stochastic models
 - Formal methods in/for model-based design
 - Performance measures other than time
- Explicit WG time and reporting time scheduled

Working Groups (II)

- Worst-case vs. stochastic models
 - elaborate discussions about determinism, stochastics and non-determinism
 - different use of terminology in different communities!
 - table with comparisons/links between models
- Performance measures other than time
 - new classification of measures: physical vs. social measures, orthogonal to rate vs. cumulative measures
 - what do you want to know: mean, min, max, quantile, distribution, optimization
 - how to achieve/calculate these measures
 - how to control them

Working Groups (III)

- Formal methods for model-based design
 - cannot replace human creativity
 - can help to avoid “bloopers”
 - no one-size-fits-all model class
 - complexity often unfavorable
 - analysis of the design steps using FM, as well as the per-step designs
- Interfaces for embedded system design
 - did not report
 - some key participants had to leave early

Teaching Embedded Systems

- G Smit (UT) reports on the ES MSc program at 3TU in NL
- R Wilhelm (UdS) reports on the EU/Artist curriculum for ES
- L Thiele (ETHZ) reports on the Lugano ES MSc program
- Although there are a few ES curricula around, there is no overall consensus about what ES is (and what should be in an MSc curriculum)
- There is agreement on the use of quantitative methods, but not on which ones (everyone his/her hobbyhorse)
- Important for all ES curricula: interdisciplinary character should be stressed (e.g., project work between EE & CS)
- And it is different from Mechatronics (which is also about the design of the mechanical system itself; ES is not)

Overall Results (I)

- Better awareness about problems and challenges for ES design
- Better awareness about what other areas have to offer in that design process
- Mutual learning of design-supporting techniques
- Start with building explicit links between areas and approaches (e.g., RT calculus & queuing theory, model transformations & semantics, model checking & performance evaluation)
- Work in WGs helped in “getting closer” to colleagues (with other background)
- Wider angle towards teaching ES

Overall Results (II)

“The only distinguishing aspect of embedded systems is the quantitative one” (Thiele)

- In line with Henzinger & Sifakis: “ES ...involve computations that are subject to physical constraints...”
- This requires an explicit handling of the “extra-functional” (read: quantitative) properties, next to the traditional functional (correctness) properties
- This, in turn, requires true ES curricula to be more than just EE (or CS) curricula with some small extras