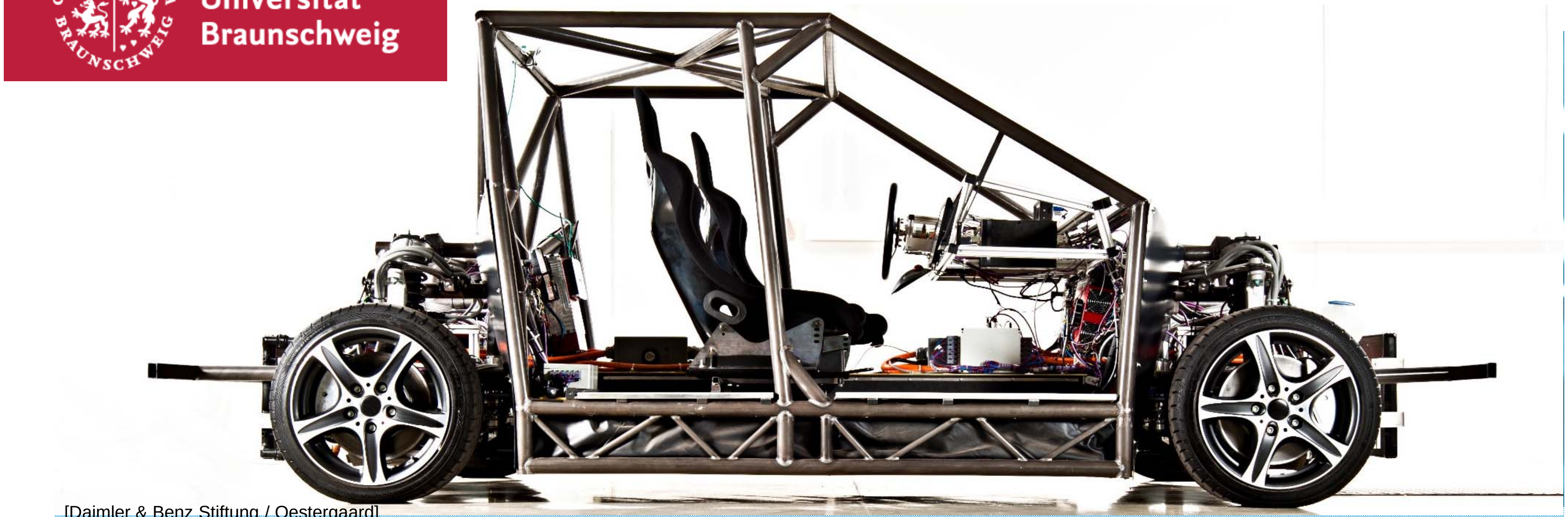
Technische Universität Braunschweig



[Daimler & Benz Stiftung / Oestergaard]

Beyond the deadline:
New interfaces between control and scheduling for the design and analysis of critical embedded systems – Tutorial introduction

Rolf Ernst, TU Braunschweig, 15.10.2017

Designing with deadlines

- deadlines are specifications for system execution
- many purposes
  - system timing - constrain function delays
  - scheduling control – e.g. EDF
  - data consistency - avoid data loss due to overwriting or early reading,
  - LET based design
  - memory usage – e.g. avoid stack overflow due to overlapping job executions
- deadlines guarantees are expensive
  - requires worst case design

Deadlines revisited

- move beyond worst case design by allowing designs with controlled deadline misses
  - “less-than-worst-case” design
- the effect of a deadline miss depends on the deadline purpose
  - function delay constraint → modified system function
  - data consistency → data loss or data corruption
  - memory bounds → exceptions or system crash
- these effects must be modeled in the application

Modeling deadline misses

- **probabilistic models**
  - **usually strong assumptions – random variables, independence**
  - **extensions for more general models (e.g. copulas)**
- **models for soft real-time systems**
  - **quantification of „soft“ deadline overrun → optimization problem**
- **weakly-hard real-time systems**
  - **bounds the tolerated number of deadline misses**
  - **defined as a window function**
    - ***at most m deadline misses out of k executions are tolerated***

- ***these models require transition from single worst case to worst case execution traces***

This tutorial – weakly-hard real-time systems

- straightforward extension of established deadline model
  - based on execution traces → compatible to industrial design practice
- formal methods available
  - compatible extension of existing Worst Case Response Time Analysis (e.g. pyCPA, SmTA/S)
- weakly-hard deadline miss model compatible to function models
  - this tutorial gives examples for automatic control and communication

Tutorial presentations

- **Tutorial Introduction**
  Rolf Ernst, TU Braunschweig, Germany

Function

- **Industrial Requirements & Solutions for a Suitable Interface between Function Development and Real-Time Systems Integration**
  Dirk Ziegenbein, Bosch, Germany
- ***Stability analysis and control design for weakly hard real-time systems***
  Steffen Linsenmayer, Univ. Stuttgart

10:00 Coffee break

- **Design and Verification of Real-Time Systems with m-k Requirements: Current Practices**
  Rafik Henia, Thales TRT
- ***Timing Design and Verification of Real-time Systems with m-k Requirements: the Use of Typical Worst-Case Analysis***
  Sophie Quinton, INRIA, Grenoble
- **Beyond the m-k model: restoring performance considerations in the time abstraction**
  Marco Di Natale, Scuola Superiore Sant'Anna, Pisa

Analysis

Tutorial panel

- panel with all presenters
- some initial questions to the panelists
  - The tutorial focused on the model of weakly hard real-time systems. There are other proposals to extend real-time behavior beyond hard deadlines, e.g. using probabilistic models. Is there room for several such models in the industrial design process? What will decide on the usage?
  - is the weakly-hard system model limited to few specific applications in control and communication or could it become a general model for system design?
  - will the use of (m, k) guarantees change the embedded system design process and how?

Please, add your own questions