ITEA 2

INFORMATION TECHNOLOGY FOR EUROPEAN ADVANCEMENT

openETCS

Towards model-based design for ETCS speed and distance monitoring

joint work of
Alexander Nitsch, Benjamin Beichler, Frank Golatowski, Christian Haubelt

supported by:

Federal Ministry of Education and Research

Liberté • Égalité • Fraternité
RÉPUBLIQUE FRANÇAISE
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE

INVESTISSEMENTS D'AVENIR

Région de Bruxelles-Capitale

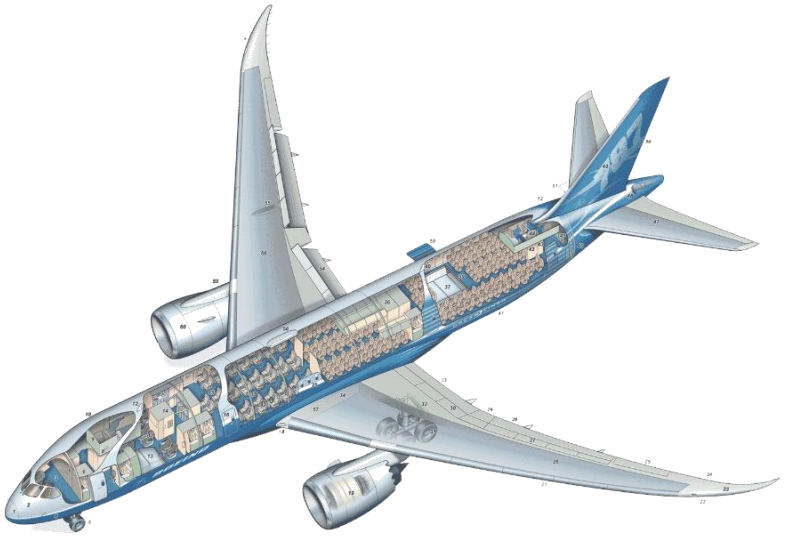
GOBIERNO DE ESPAÑA
MINISTERIO DE CIENCIA E INNOVACIÓN

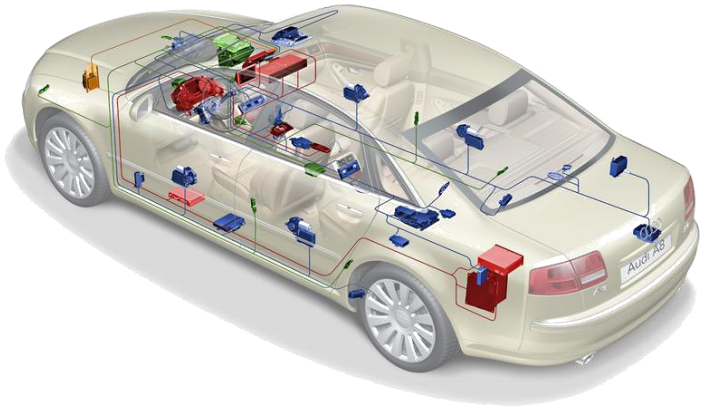
openETCS@ITEA2 Project

Alexander Nitsch

Berlin, 24.09.2014

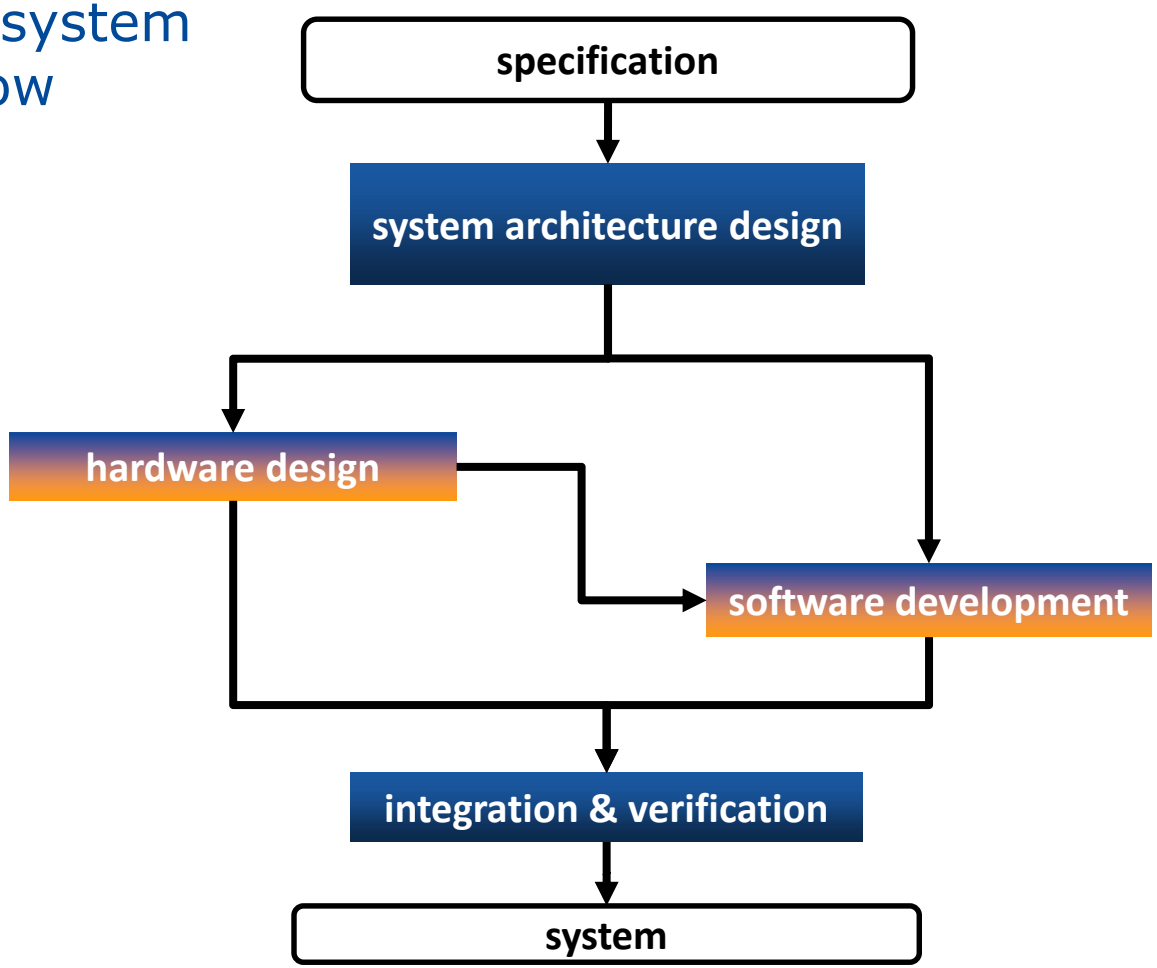

PHILIPS



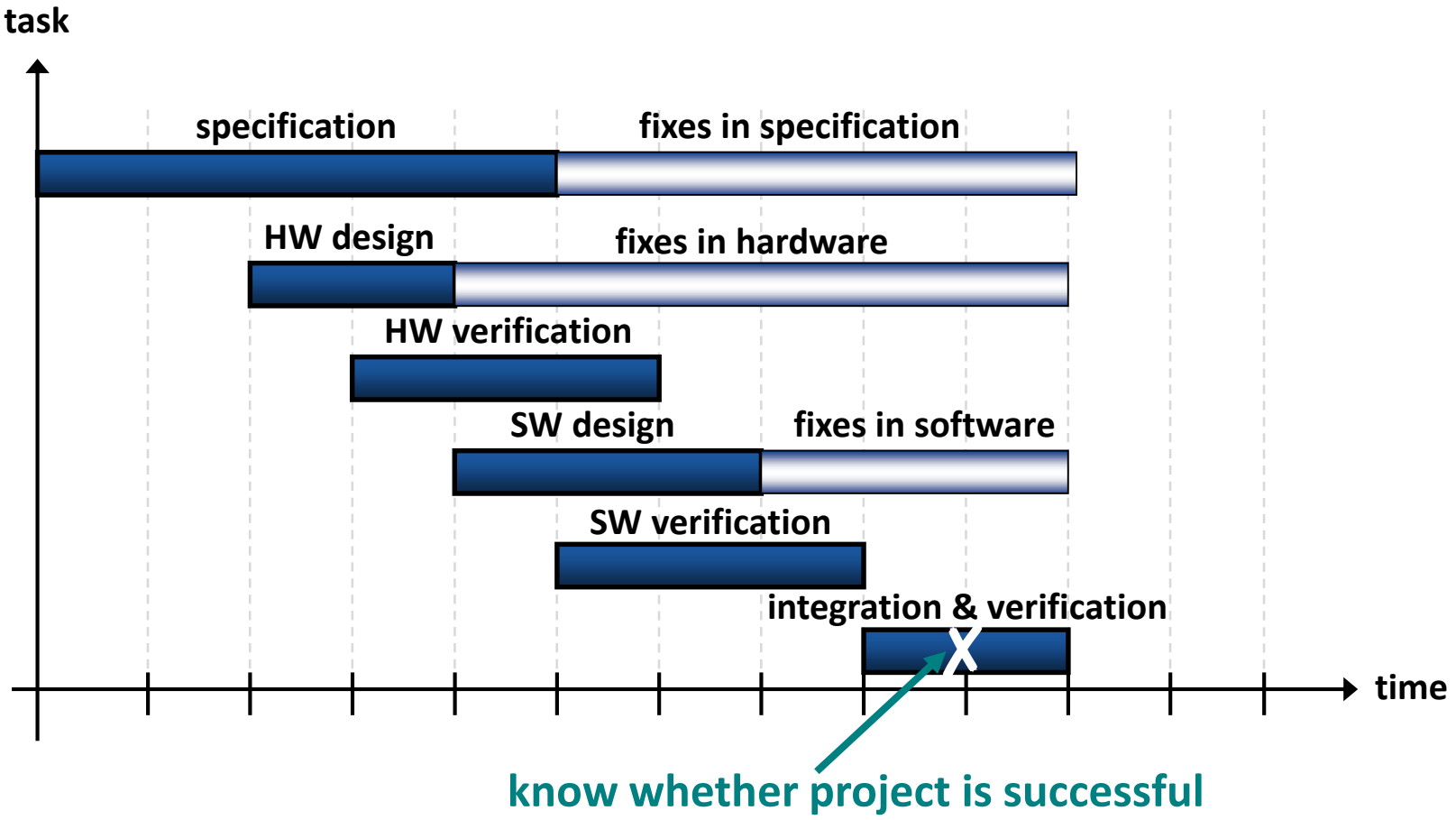

Audi A8


BD

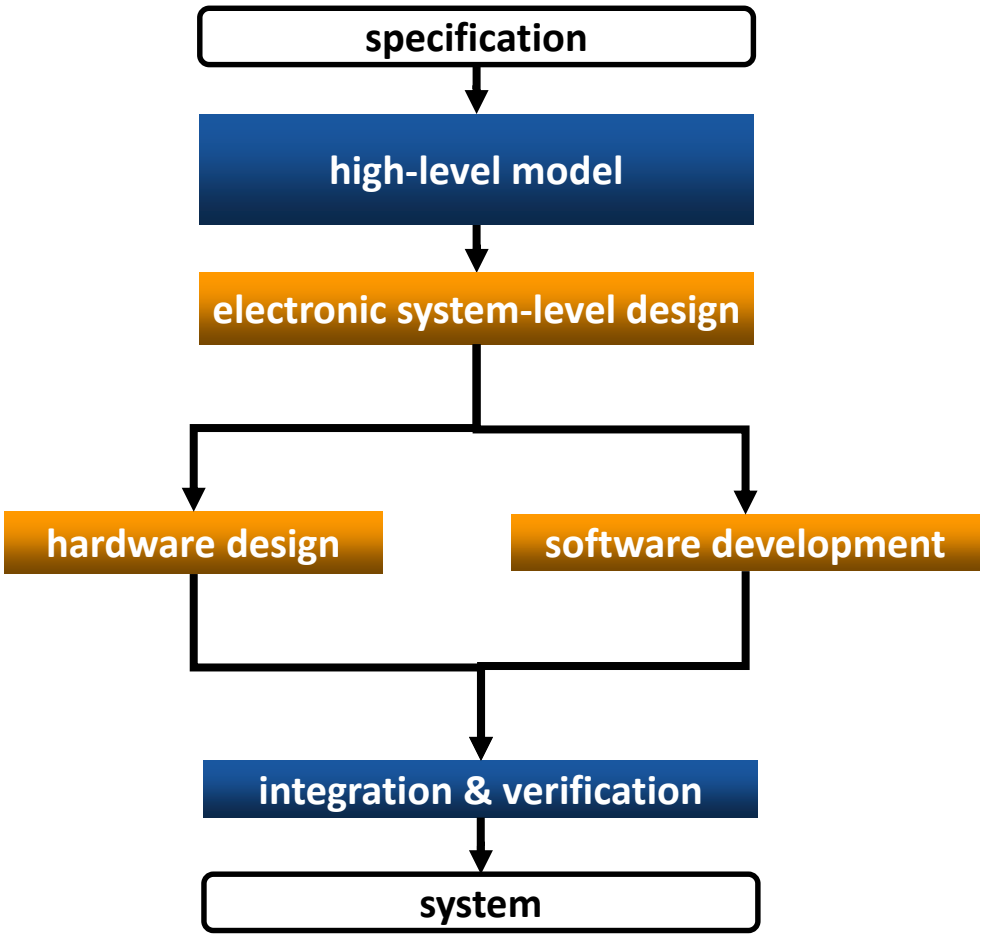
Classical system design flow



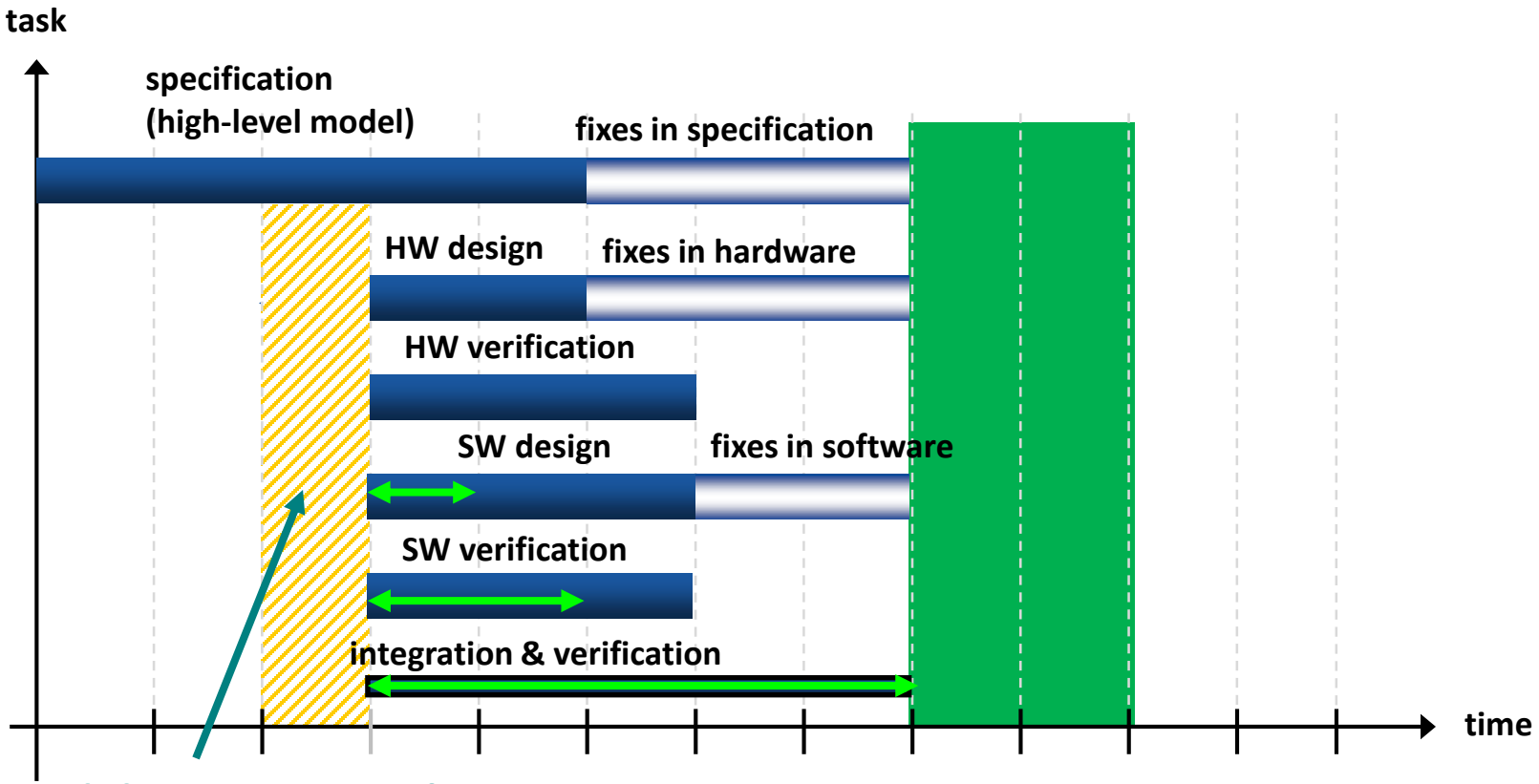
Design cycle of the classical design flow



Model-based design flow

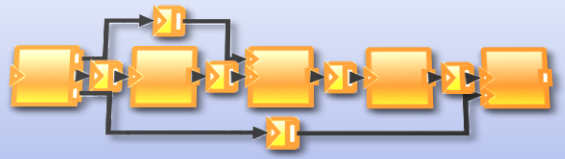


New Design Cycle



find good design options here

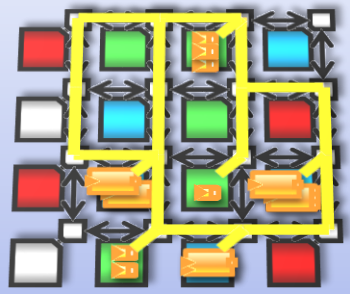
Research field University of Rostock



application model

hardware model

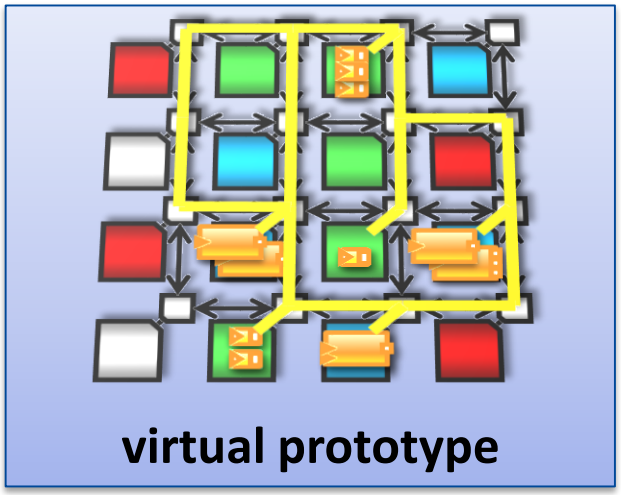
system synthesis: allocation, binding, scheduling



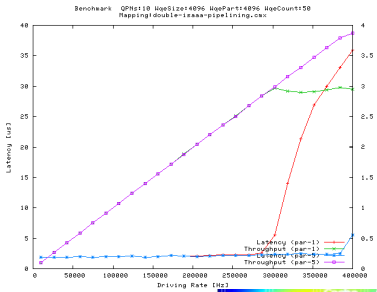
high level virtual prototype

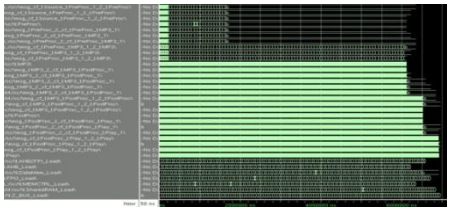
SYSTEMC™

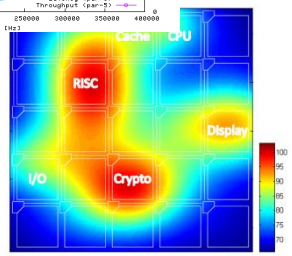
Research field University of Rostock

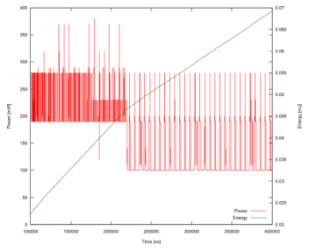


SYSTEMC™



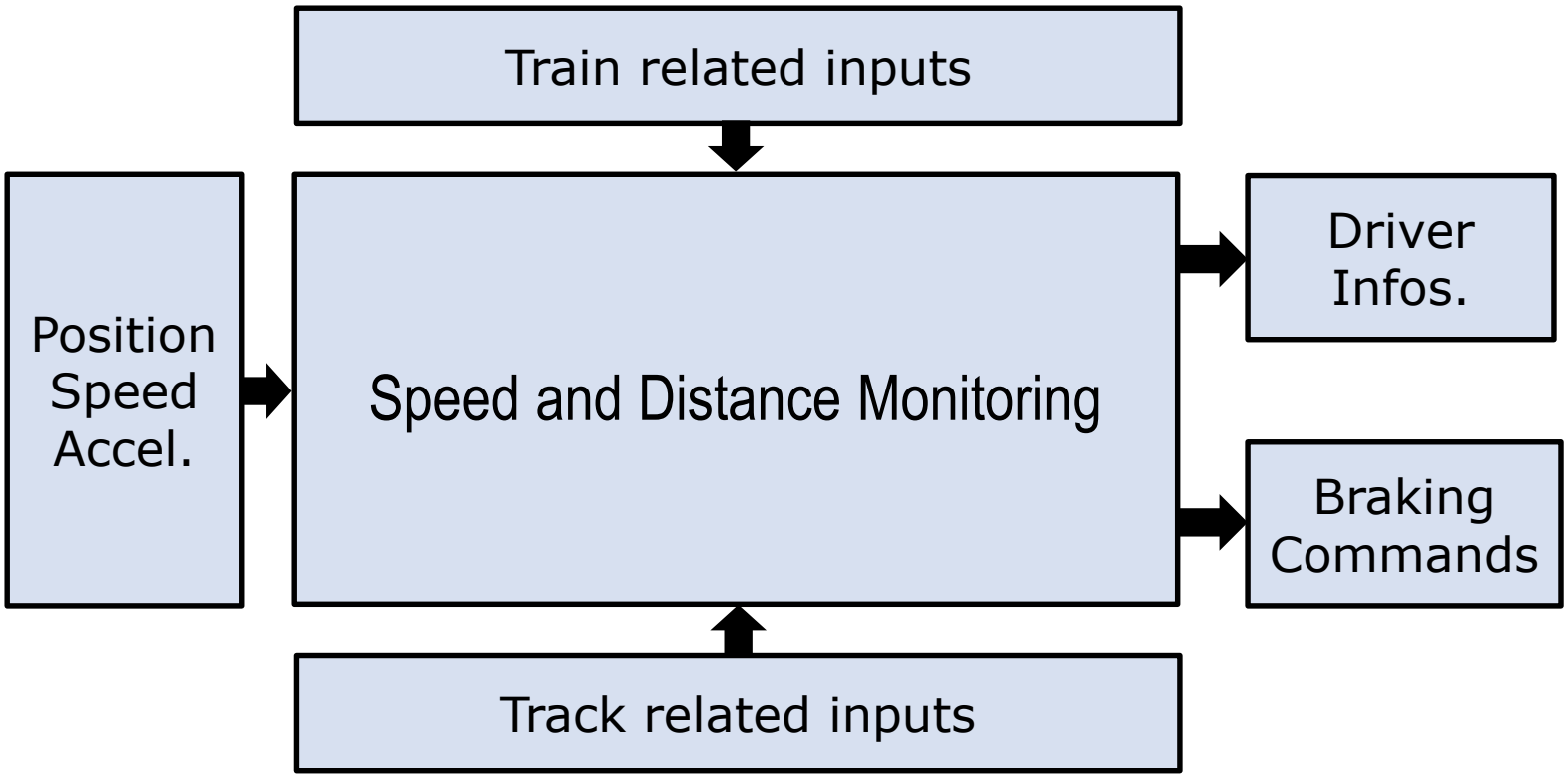






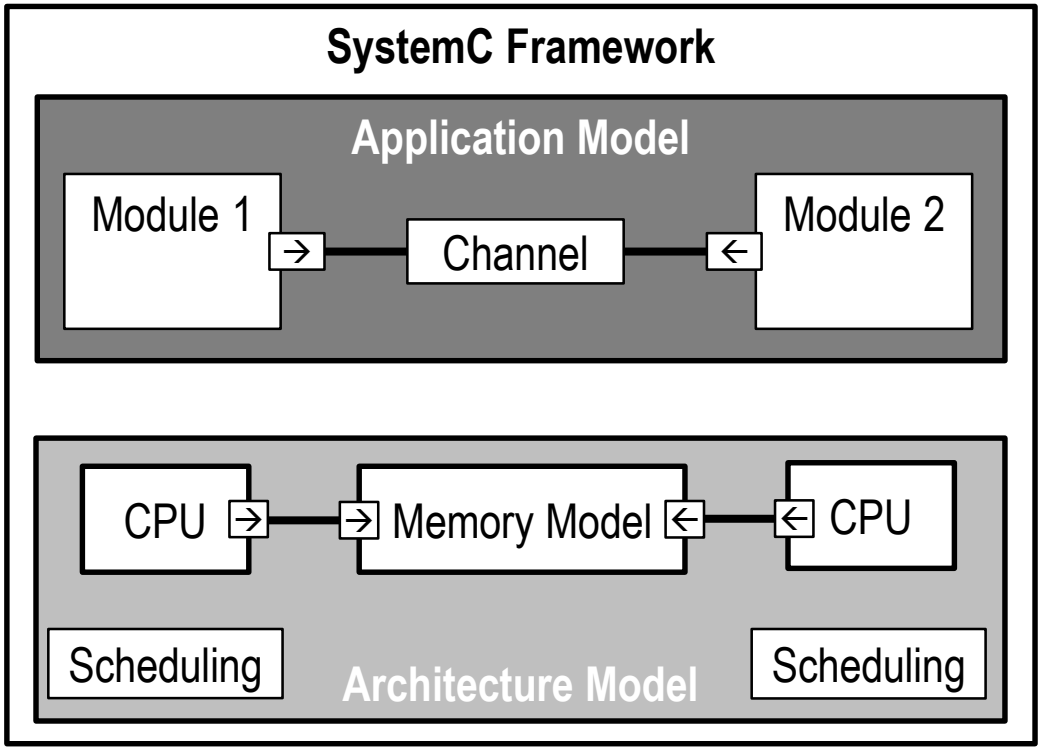
Framework for verification of nonfunctional properties:

- **Simulative evaluation of abstract architecture models**
- **Performance estimation of a hardware system running a specific application**
- **Application model: ETCS Speed and Distance Monitoring**



Framework for verification of nonfunctional properties:

- **Simulative evaluation of abstract architecture models**
- **Performance estimation of a hardware system running a specific application**
- **Application model: ETCS Speed and Distance Monitoring**

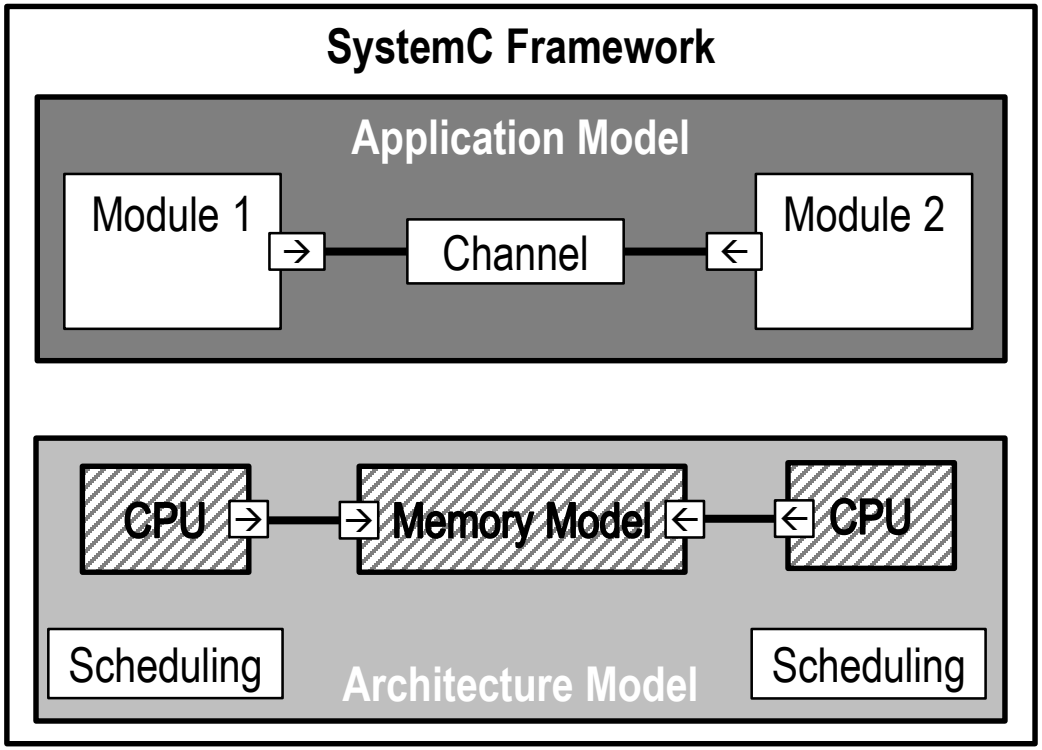


Framework for verification of nonfunctional properties:

- **Simulative evaluation of abstract architecture models**
- **Performance estimation of a hardware system running a specific application**
- **Application model: ETCS Speed and Distance Monitoring**

Steps

1. **Allocation**

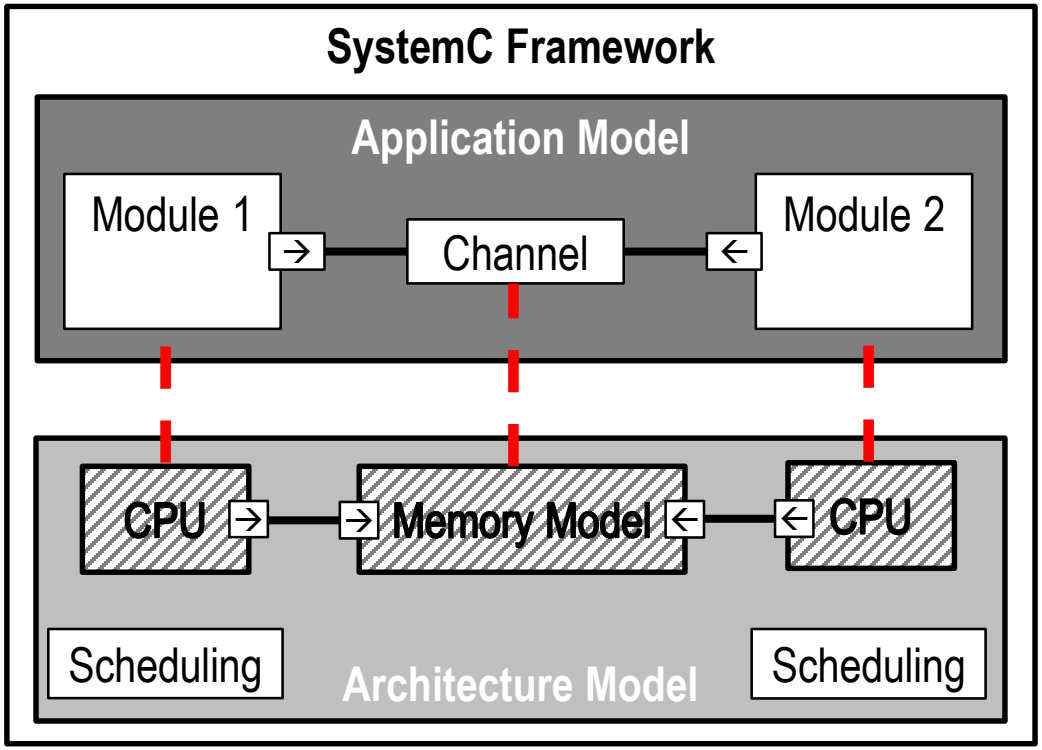


Framework for verification of nonfunctional properties:

- **Simulative evaluation of abstract architecture models**
- **Performance estimation of a hardware system running a specific application**
- **Application model: ETCS Speed and Distance Monitoring**

Steps

1. **Allocation**
2. **Binding**

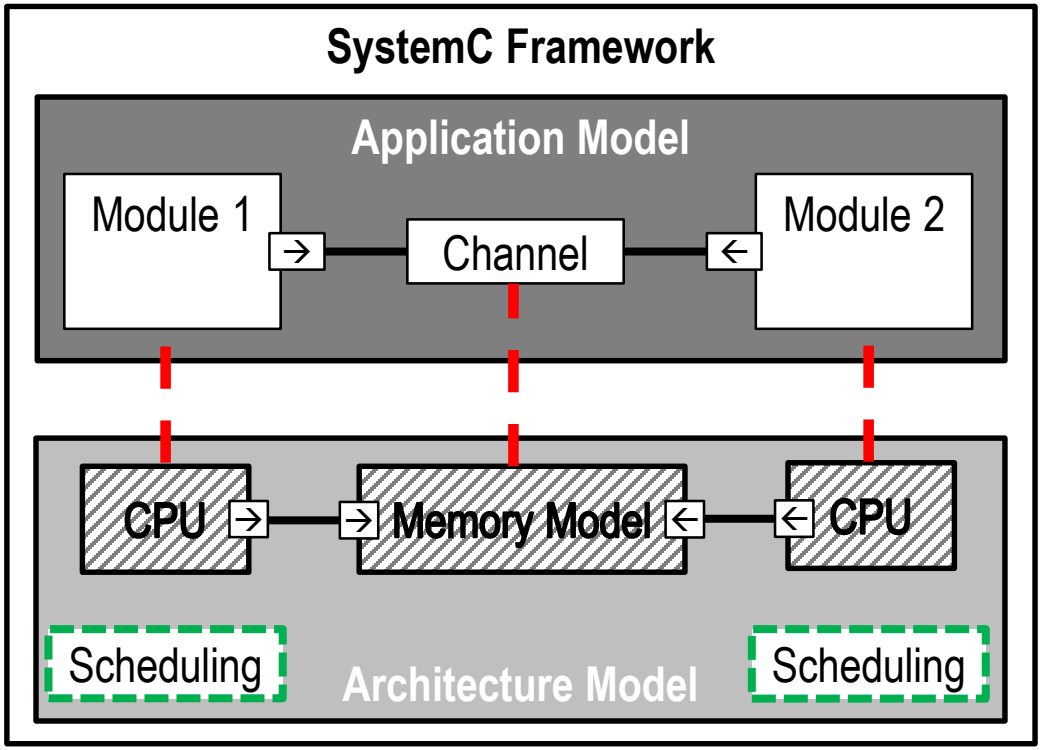


Framework for verification of nonfunctional properties:

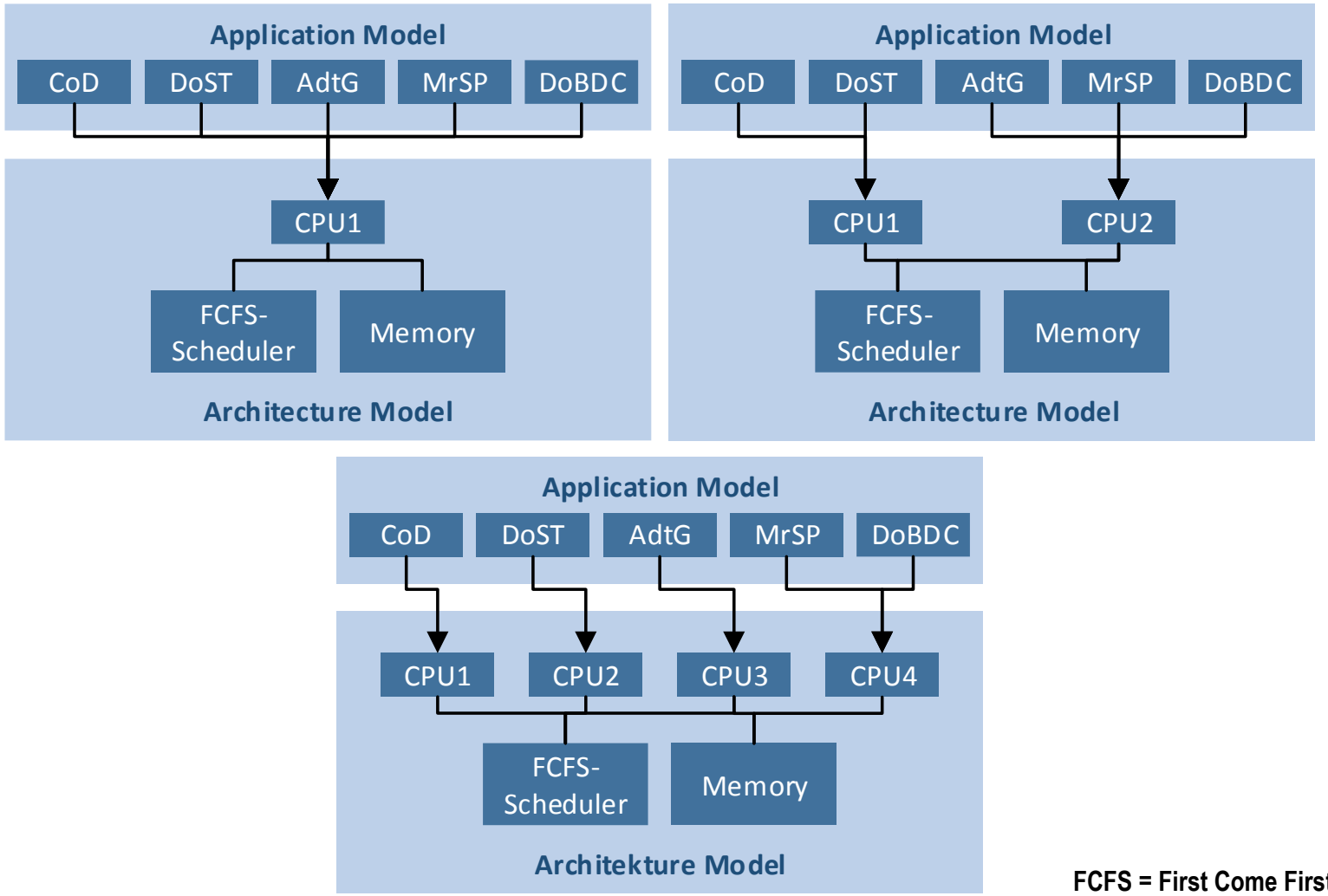
- **Simulative evaluation of abstract architecture models**
- **Performance estimation of a hardware system running a specific application**
- **Application model: ETCS Speed and Distance Monitoring**

Steps

1. **Allocation**
2. **Binding**
3. **Scheduling**



Different architecture models

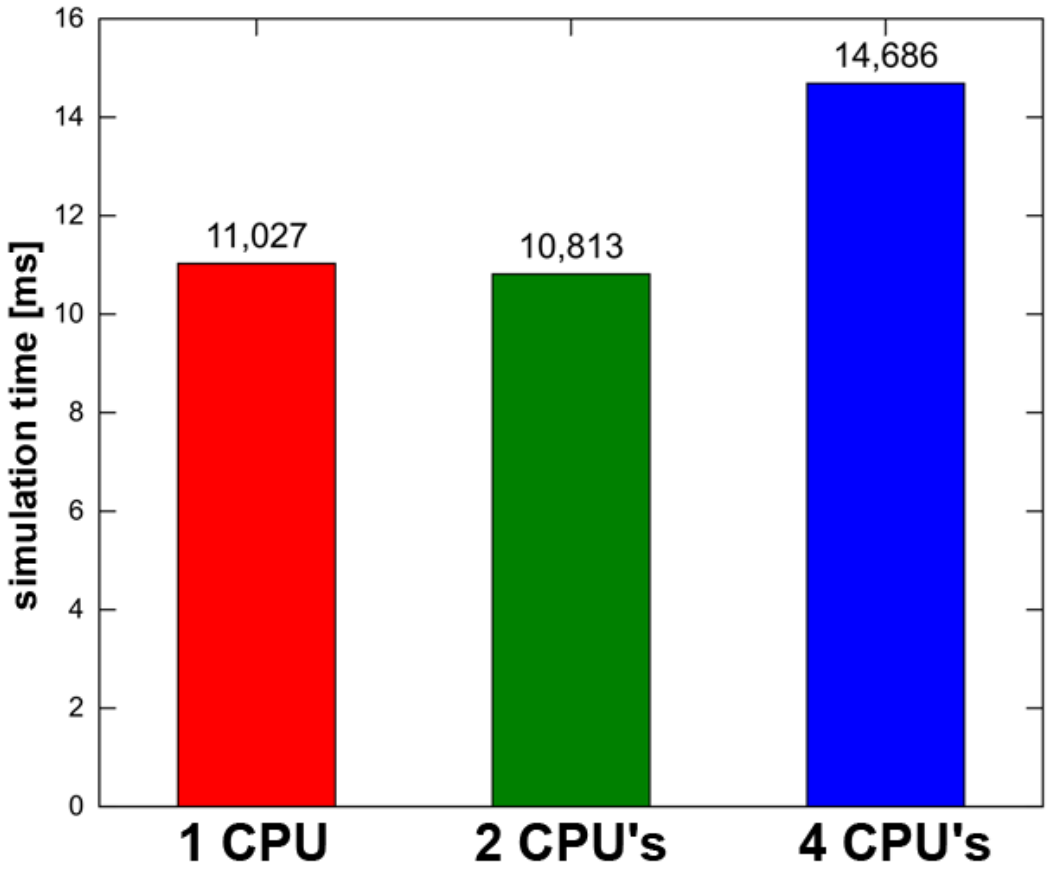


FCFS = First Come First Serve

Time of braking curve calculation

- FCFS-Scheduler
- SDRAM-Model
  Latency 20ns
  + 2ns/byte
- Scheduling do not consider data dependencies

→ Multi core design (4 CPU's) has highest execution time



Conclusion

- ***Model-Based Engineering/Design*** **enables fast and effective development of complex systems**
- **Benefit:**
  - **parallel development of hardware and software**
  - **reuse of existing models by easy adaptation to new (sub-)systems**

- **Formal specification in terms of a formal model allows:**
  - **Automated consistency checks**
  - **Automated testing**
  - **Automated documentation**
  - **Source code generation**
  - **Simulation of functional and extra functional aspects/properties**

- **Performance estimation framework of the University of Rostock eases design decisions for computation resources & communication resources**

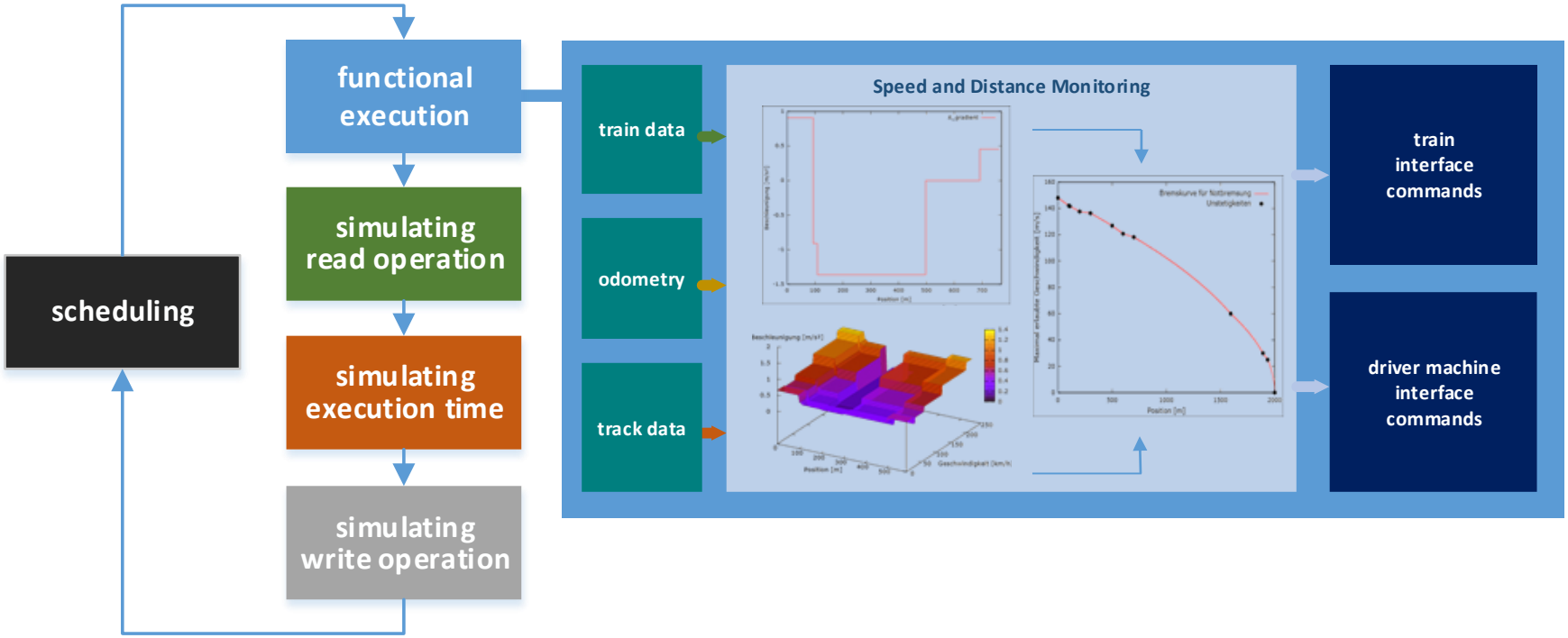
Thank you

Backup

Implementation of the specification with SystemC
Why using SystemC?

- **SystemC = modelling language initially developed to build hardware**
- **developed as a language to model the behavior of a system independent of implementation**
- **at an very early stage of system design: possible by means of this language to evaluate the system**
  - **Specification / Timing / Scheduling / Performance**
  - **Estimation of hardware resources (single core, multi core, quad core)**
  - **Software structure (modules)**
  - **Type of onboard unit (OBU)**

Implementation of the specification with SystemC Progress of work:

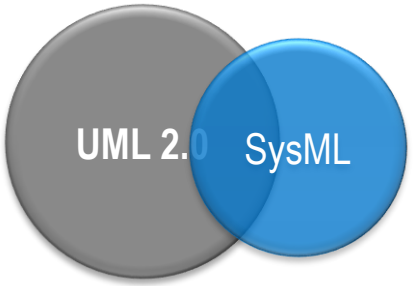


Implementation of the specification with SystemC Solutions:

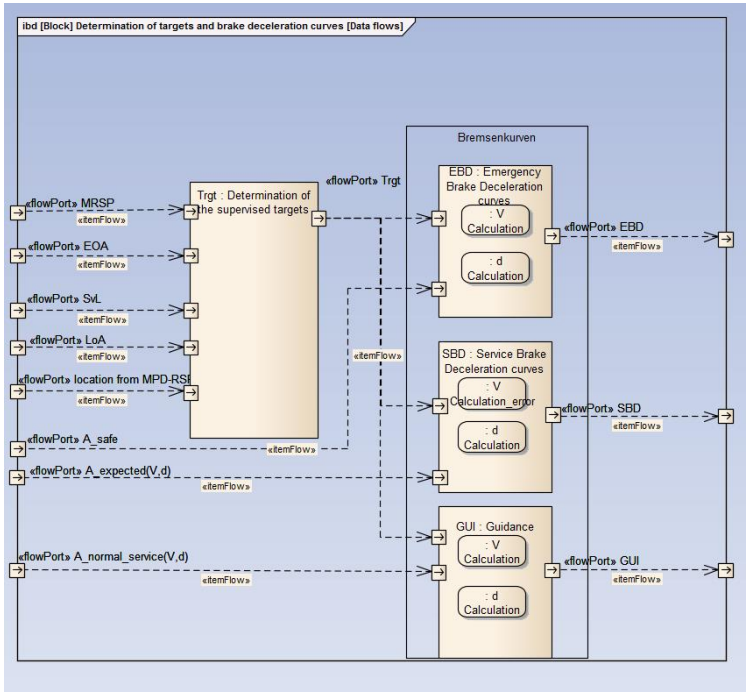
- **formalized specification and executable model (calculating braking curves with real input data)**
- **found and reported specification issues**
- **successful graduation of students**
- **first submitted publication**

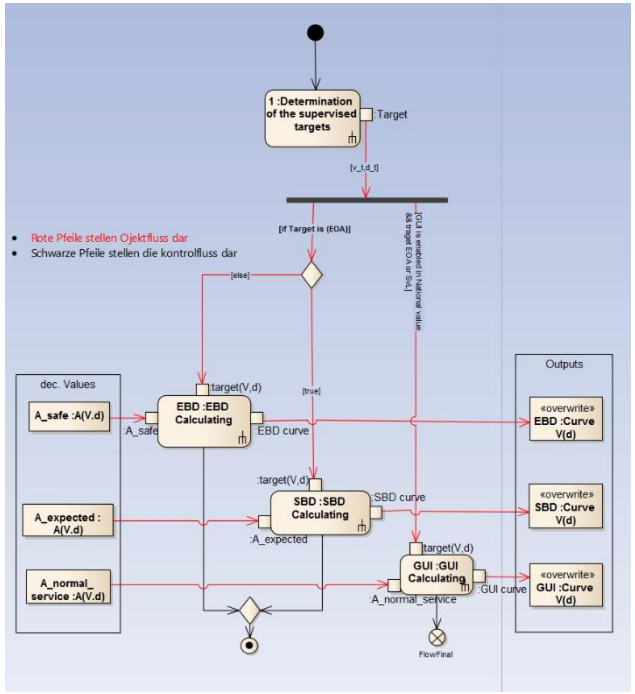
Implementation of the specification with SysML
What is SysML and why using?

- graphical modeling language suitable to needs of system engineering
- helps modeling system within a broad range of system variety that may include hardware, software, data, personnel and facilities
- supports the specification, analysis design, verification and validation of complex systems
- SysML supports system modeling in a (semi-) formal way:
  - architecture (statics)
  - behavior (dynamics)
  - requirements
- formal representation allows the automatic transformation into:
  - Code, Test cases



Implementation of the specification with SysML Progress of Work:





New (optimal) system design flow