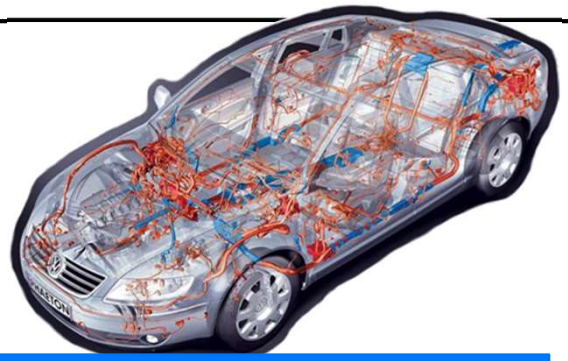
HOCHSCHULE
HAMM-LIPPSTADT

Stefan Henkler E-Mail: stefan.henkler@hshl.de

Internship / semester abroad

- Questions!

Motivation – Systems Engineering



What do these systems have in common?

Electronic engineering and computer science are main drivers for innovation!

Pervasive interaction with the environment (humans as well as other systems)

Enhanced functionality is realized by Software **embedded** into a **technical system** that interacts with the real physical world

-> outcome: consider more than one engineering discipline -> systems engineering is required

https://www.springerprofessional.de/

Outcomes

- Students will be able to use SysML for technical applications in various project phases
- … apply the methods and techniques of systems engineering in order to design complex systems…

Outline

HOCHSCHULE HAMM-LIPPSTADT

I. Introduction
II. Requirements
III. Analysis & Design
IV. The Development Live Cycle

Orga

- Today, this week plus next week during prototyping labs and lecture session of SE
- Lab
  - The overall evaluation of the module is given by the lab work!
    - Attending to the lab is a prerequisite for the evaluation
    - Team work

Prototyping and SE	Monday	Wednesday
from to	Prototyping A	Prototyping B
21.3 to 25.3	SE (2SWS) 2SWS (Team bui	SE (2 SWS) 2 SWS (Team building)
28.3 to 1.4	SE (2SWS) 2SWS Lab	SE (2SWS) 2SWS Lab
4.4 to 8.4	4 SWS Lab	4 SWS Lab
11.4 to 15.4	4 SWS Lab (presentation of	4 SWS Lab (presentation of 1st prototyp)
18.4 to 22.4		
25.4 to 29.4	Gido	Gido
2.5 to 6.5	Gido	Gido
9.5 to 13.5	Gido	Gido
16.5 to 20.5	Kris	Kris
23.5 to 27.5	Kris	Kris
30.5 to 3.6	Kris	Kris
6.6 to 10.6		Free feedback all
13.6 to 17.6	Free feedback all (KRSH)	Free feedback all (KRSH)
20.6 to 24.6	final presentation	final presentation
27.6 to 15.7	final paper	final presentation

HOCHSCHULE HAMM-LIPPSTADT

1. Definitions & Terminology
2. Examples

8 Overview

- https://youtu.be/EUAg3xJyQ9M

Definition software-intensive system

- What is a software-intensive system?

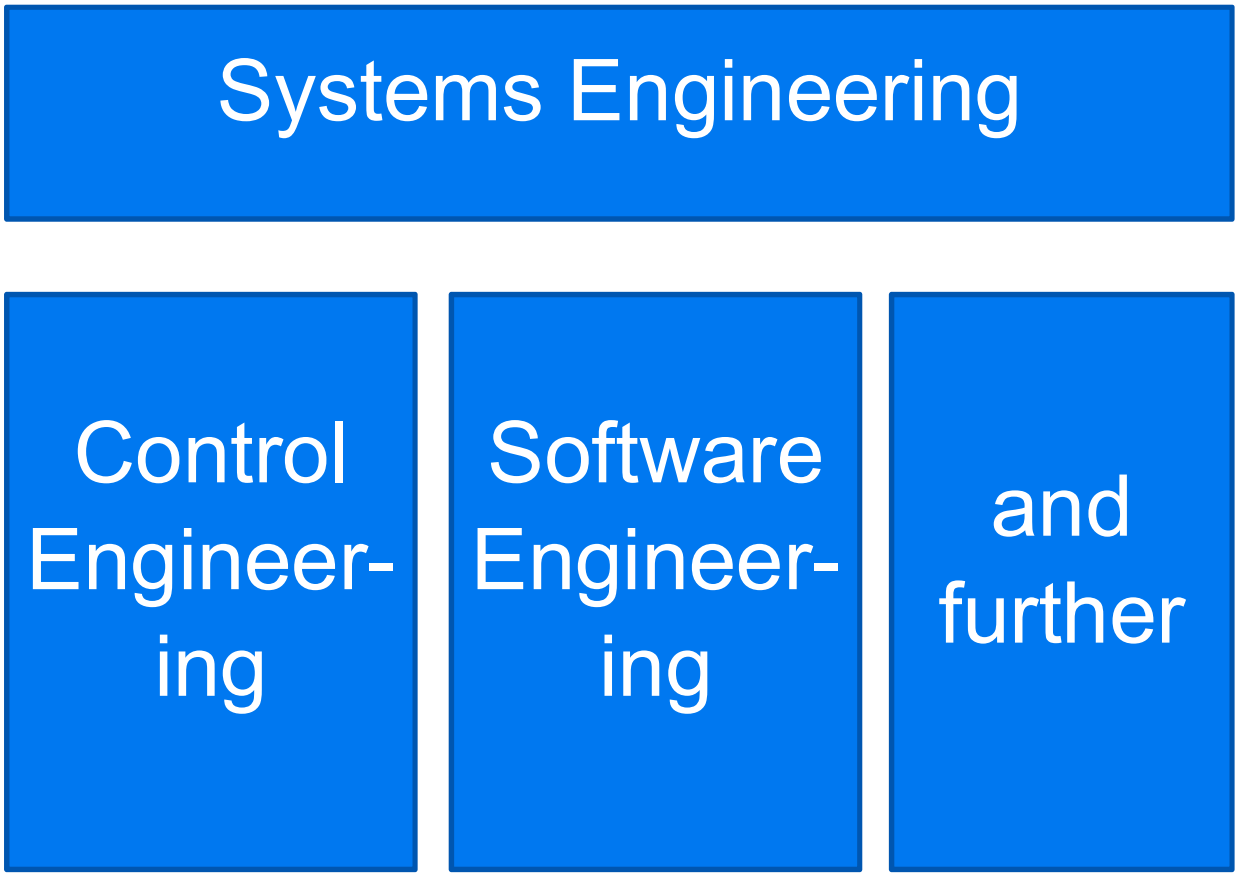
> A software-intensive system is any system where software contributes essential influences to the design, construction, deployment, and evolution of the system as a whole.
> [IEEE-Std-1471-2000]

- Observation [EU-NSF-SIS2004]:

"Software has become a key feature of a rapidly growing range of products and services from all sectors of economic activity. Software-intensive systems include"

- large-scale heterogeneous systems,
- embedded systems for automotive applications,
- telecommunications,
- wireless ad hoc systems,
- business applications with an emphasis on web services etc.

Our daily lives depend on complex software-intensive systems, from banking to communications to transportation to medicine.



Challenge / Requirement: the integration of the disciplines

Software Engineering

- *The Establishment and use of sound **engineering principles** in order to obtain **economically** software that is **reliable** and works **efficiently** on real machines.*

  F.L. Bauer *in* [Buxton& Randell1969]

- software engineering. (1) The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software. (2) The study of approaches as in (1).

  [IEEE-Std-610.12-1990]

- Looking Beyond Software?
  - For computer scientists, being told that creating software-intensive systems is difficult is not new. Even attendees at the 1968 NATO Software Engineering Conference in Garmisch, Germany, recognized the need to build, for example, a foundation for the systematic creation of software-intensive systems [3].

    (See [Freeman&Hart2004])

System Engineering

- Systems Engineering is an **interdisciplinary** approach and means to **enable** the **realization** of **successful systems**. It focuses on defining customer needs and required functionality early in the development cycle, documenting requirements, then proceeding with design synthesis and system validation while considering the complete problem:
  - Operations,
  - Cost & Schedule,
  - Performance,
  - Training & Support,
  - Test,
  - Disposal,
  - Manufacturing

Relation to Software Engineering:

- Systems Engineering **integrates** all the **disciplines** and specialty groups into a team effort forming a structured development process that proceeds from concept to production and operation. **Systems engineering** considers both the **business** and the **technical needs** of all customers with the goal of providing a **quality product** that **meets the user need.**
  - **Integrates and uses Software Engineering**

Control Engineering

- Control engineering is the engineering discipline that focuses on the **mathematical modeling** systems of a diverse nature, analyzing their dynamic behavior, and using **control theory** to make a controller that will cause the systems to behave in a desired manner.
- Control engineering has **diversified applications** that include **electrical** engineering, **mechanical** engineering, process control, science, finance management, and even human behavior.

Relation to Software Engineering:

- Control is challenging since it takes strong foundations in engineering and mathematics, **uses extensively computer software and hardware** and requires the ability to address and solve new problems in a variety of disciplines, ranging from aeronautical to electrical and chemical engineering, to chemistry, biology and economics.

Other relevant disciplines

- **Mechatronics** is the synergistic combination of
  - mechanical engineering ("mecha" for mechanisms),
  - electronic engineering ("tronics" for electronics), and
  - software engineering.
    - Purpose:
      - study of automata from an engineering perspective
      - controlling advanced hybrid-systems
      - …
- **Telematic** is the [integrated] use of telecommunications and informatics.
  - It is the science of **sending**, **receiving** and **storing information** via **telecommunication** devices.
  - Often used for Global Positioning System technology integrated with computers and mobile communications technology or for such systems within road vehicles (vehicle telematics)

The need for integration

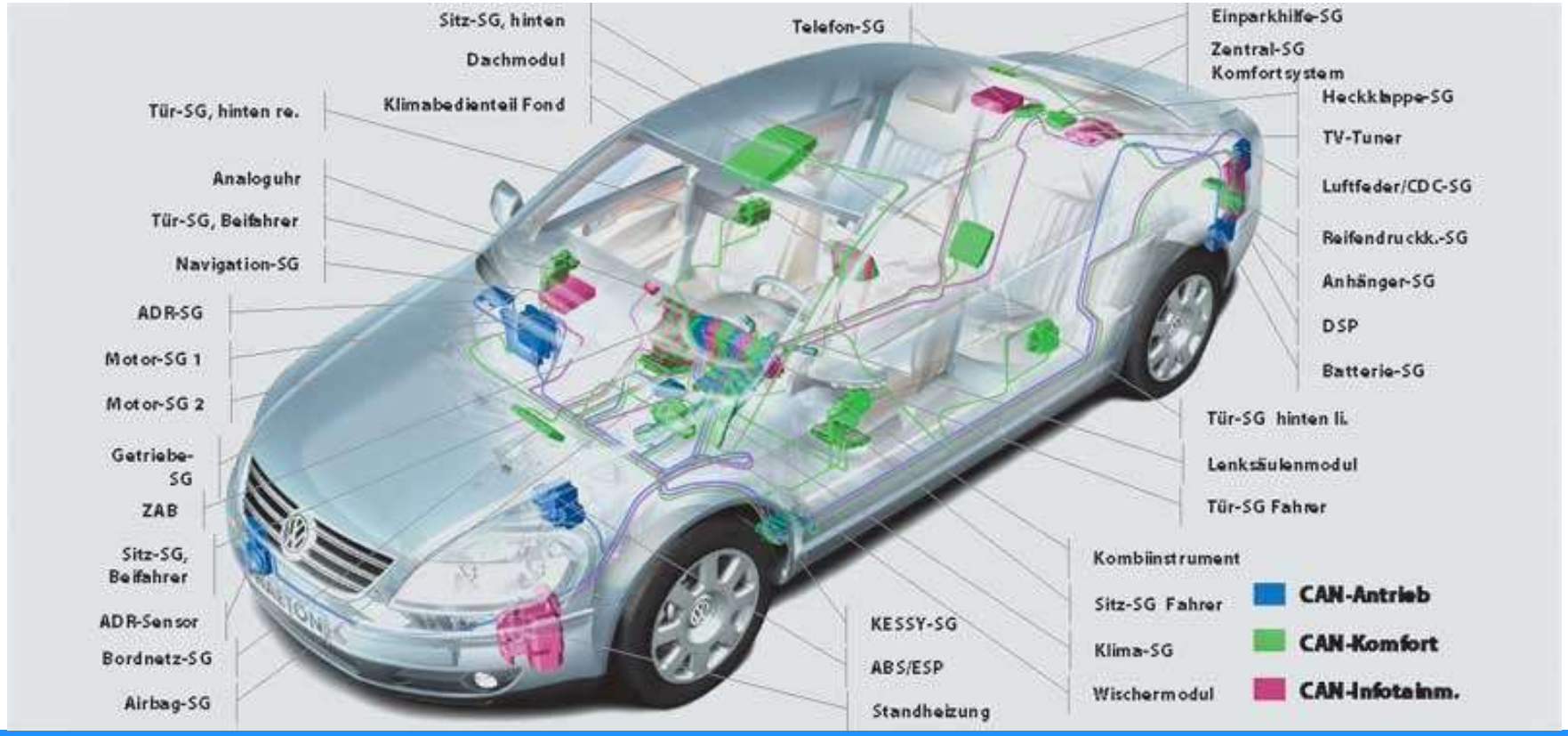
HOCHSCHULE HAMM-LIPPSTADT

- **The past (in most large projects):**
  - **System Engineering:** A set of interdisciplinary activities leading to a system that works and will meet stakeholder needs
  - **Software Engineering:** A set of processes that turn **software requirements** into **working code**.
  - **Control Engineering** simply **use extensively computer software and hardware**.
  - Lack in support of integration the different disciplines
    - Show stopper for large systems with complex functions
- **The needed future:**
  - Software Engineering participate in establishing **system requirements,** in exploring **design alternatives,** and in **integration** and system **test.**
  - Software Engineering must be **integrated** with Control Engineering in order to control future generation of systems of systems.
  - Systems -, Control -, and Software Engineering must seek to **integrate**, not **differentiate**

1. Definitions & Terminology
2. Examples (recap - µC)

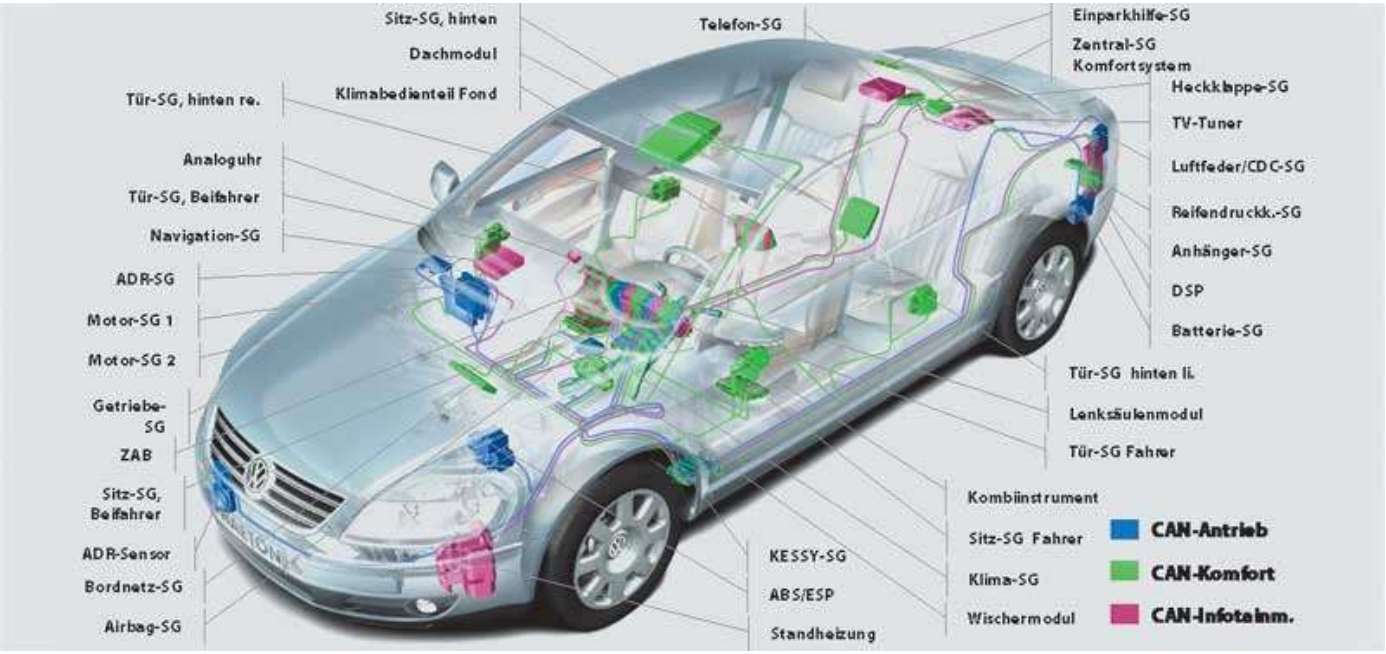
I.2.1 Example
Automotive

- Characteristics of the VW Phaeton:
  - 61 electronic control units (ECUs); 45 different ones
  - three coupled bus systems including one optical bus (3860 m cables; 64 kg)
  - about 2500 signals in 250
  - CAN-messages http://www.heise.de/ct/03/14/170/
  - 50 MB memory
  - Functions realized with software



Automotive
challenges

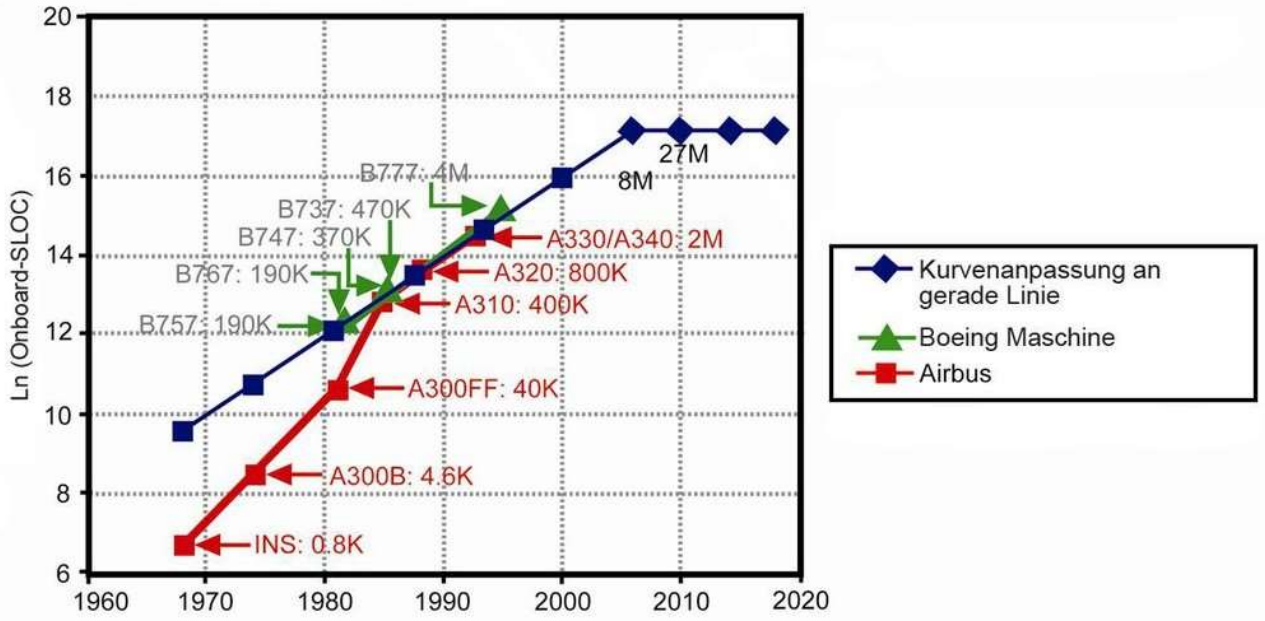
- Dramatic increase in complexity
- High cost pressure
  - E.g deploy new system functionality on existing architecture
    - Design Space Exploration
- Supply chains with vendors and suppliers
- Heterogeneity in hardware and software



I.2.2 Aircraft

- Over 1000 networked controller
- Current systems: over 27 M LOC

Voraussichtliches Wachstum der Onboard-SLOC



Airbus Datenquelle: J.P. Potocki De Montalk, "Computer Software in Civil Aircraft," Sechste Jahreskonferenz zur Softwaresicherung (Compass '91), Gaithersburg, MD, 24-27 Juni, 1991
Boeing Datenquelle: J.J. Chilenski, 2009

Aircraft

- Characteristics
  - Hard real time
  - Limited ressources
  - Highly networked
  - High requirements for safety
- Challenges
  - Dramtic increase of complexety
    - Communication
  - Heterogenity of hardware and software
  - Supply chain of integrator and supplier

Main references

- T. Weilkiens, TotalBoox, and TBX, *Systems Engineering with SysML/UML.* Elsevier Science, 2011.