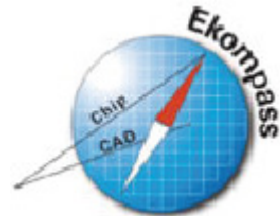




Gefördert durch



Projektbüro:



Förderkennzeichen:

01 M 3178

AutoSUN – Entwurf von elektronischen Automobil-Systemen aus toleranzbehafteten Baugruppen

Vortragende

Karsten Einwich
Fraunhofer IIS-EAS

Hannover, edaworkshop
04./05.05.2010



Gliederung

1. Motivation
2. Projektziele
3. Einordnung in die Hightech-Strategie und IKT 2020
4. Einordnung in die EDA-Matrix
5. Projektstruktur
6. Projektergebnisse



Motivation

- Autoverkehr nimmt trotz steigender Energiekosten zu
- Hohe Fahrzeugdichte
- Sehr heterogene Verkehrsverhältnisse (LKW, PKW, Motorroller, Fahrrad, Fußgänger)
- Hohe Unfallgefahr
- Ausweg: Elektronische Systeme zur Unfallvermeidung
- Hoher Ausrüstungsgrad wünschenswert → kostengünstig, für alle erschwinglich
- Elektronische Systeme müssen extrem zuverlässig funktionieren
- Industrie fit machen zur Lösung der Probleme und Verbesserung ihrer Wettbewerbsfähigkeit

→ Projekt leistet dazu Beitrag mit folgenden Zielen:



Projektziele

Optimierung des Entwurfsprozesses:

- Überprüfung von Entwürfen komplexer Systeme auf das Vorhandensein spezifizierter Eigenschaften bereits in frühen Entwicklungsstadien
- Systeme zuverlässig gestalten
- Verkürzung der Entwicklungszeiten
- Senkung der Entwicklungskosten
- Senkung der Produktkosten



Projektziele

- Schnelle Simulation (drei Größenordnungen besser)
- Komplexe Szenarien simulieren
- Gemeinsame Simulation Software – Digitale und **analoge** Hardware – **Sensoren – Mechanik zur Systemvalidierung**
- Mehr Szenarien testbar gegenüber Prototyperprobung (höhere Funktionssicherheit) → vollständige Beobachtbarkeit, alle Parameter variierbar, Worst-Case-Untersuchungen möglich
- Berücksichtigung von Bauteiltoleranzen (trotz Schwankungen sichere Komponenten)
- Verbesserung der Zusammenarbeit OEM-Tier1-Tier2 (durchgängige Qualität) mit IP-Schutz

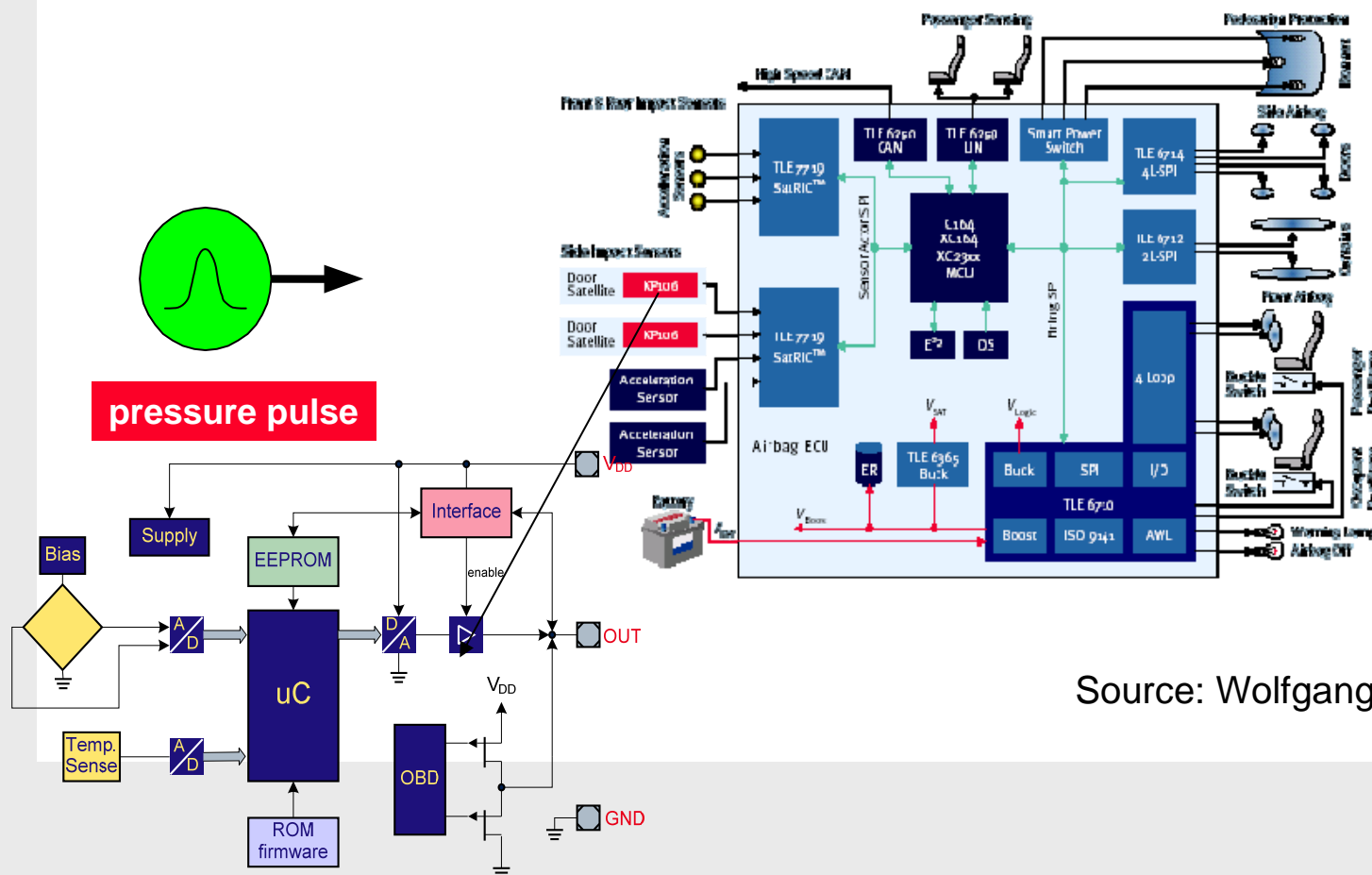


Automotive Value Chain

TIER2

TIER1

OEM



Source: Wolfgang Scherr Infineon AIM



Einordnung in das Forschungsprogramm IKT 2020

Anwendungsgebiet / Branchen: Automobil, Mobilität

Basistechnologie: Elektronik und Mikrosysteme (Chipentwurf - EDA)

→ Effektiver Systementwurf von sicherheitskritischen Systemen

→ Verifikation von Elektroniksystemen der Zukunft



Einordnung in die IKT 2020

Leitinnovation:

Initiative Automobilelektronik:

- Erschwingliche elektronische Systeme für Alle durch kosteneffektive Entwurfsmethoden
- Funktionssichere Systeme für die Automobilelektronik

Qualitätsziele:

- Sicherheit: sichere Fortbewegung auch für schwache Verkehrsteilnehmer
- Wirtschaftlichkeit: intelligentes energiesparende Systeme im Auto
 - **Entwurf innovativer Systeme für die Zukunft**
 - **ganzheitlicher Systemansatz über die gesamte Wertschöpfungskette**

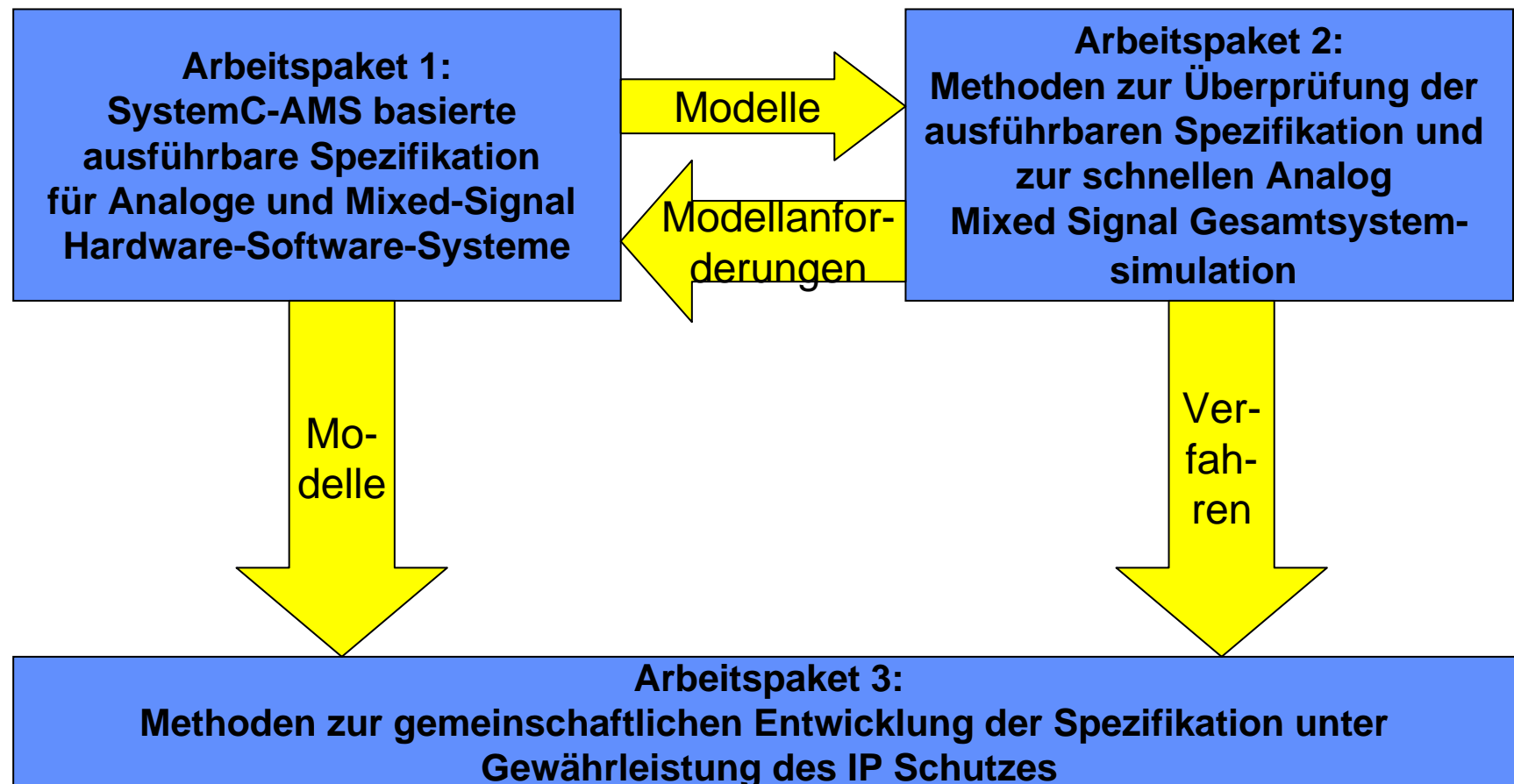


Einordnung in die edaMatrix

	System Level	Architecture Level	Dataflow Level	Electrical Level	Device and Technology Level
Specification					
Implementation					
Verification					
Manufacturing and Test					



Projektstruktur





Projektergebnisse

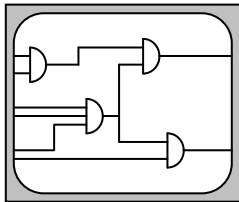
Fast Simulation and Abstract Modeling



Abstract Modeling

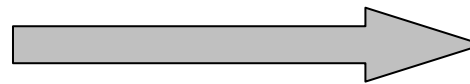
Testing Complex Systems

Hardware Design Models

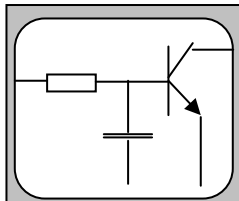


- Digital: RTL/gate level description
- Analogue: transistor level net lists

- Structural information



+ Simulation speed



- Simulation much to slow for system level verification

- Not available in early design phases

Structural representation

```
if (en) {  
    y = (a*c)+ d  
}
```

```
double t=sc_time_stamp().to_seco  
if (en.event()) {  
    t0=t;  
}  
double rel_t=t-t0;  
if (en) {  
    o=factor1 * (a-e1)*exp(-rel_t/  
} else {  
    o=factor1 * (a-e2)*(1-exp(-rel
```

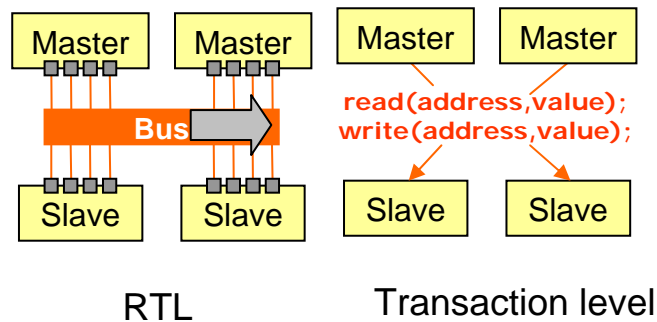
Abstract behavioral
representation



Abstract Modeling of Communication

Bus protocols may be quite complicated (and time consuming to simulate in a cycle accurate fashion on the bit level)

Basically they implement one type of operation: transport some data from sender to recipient(s)

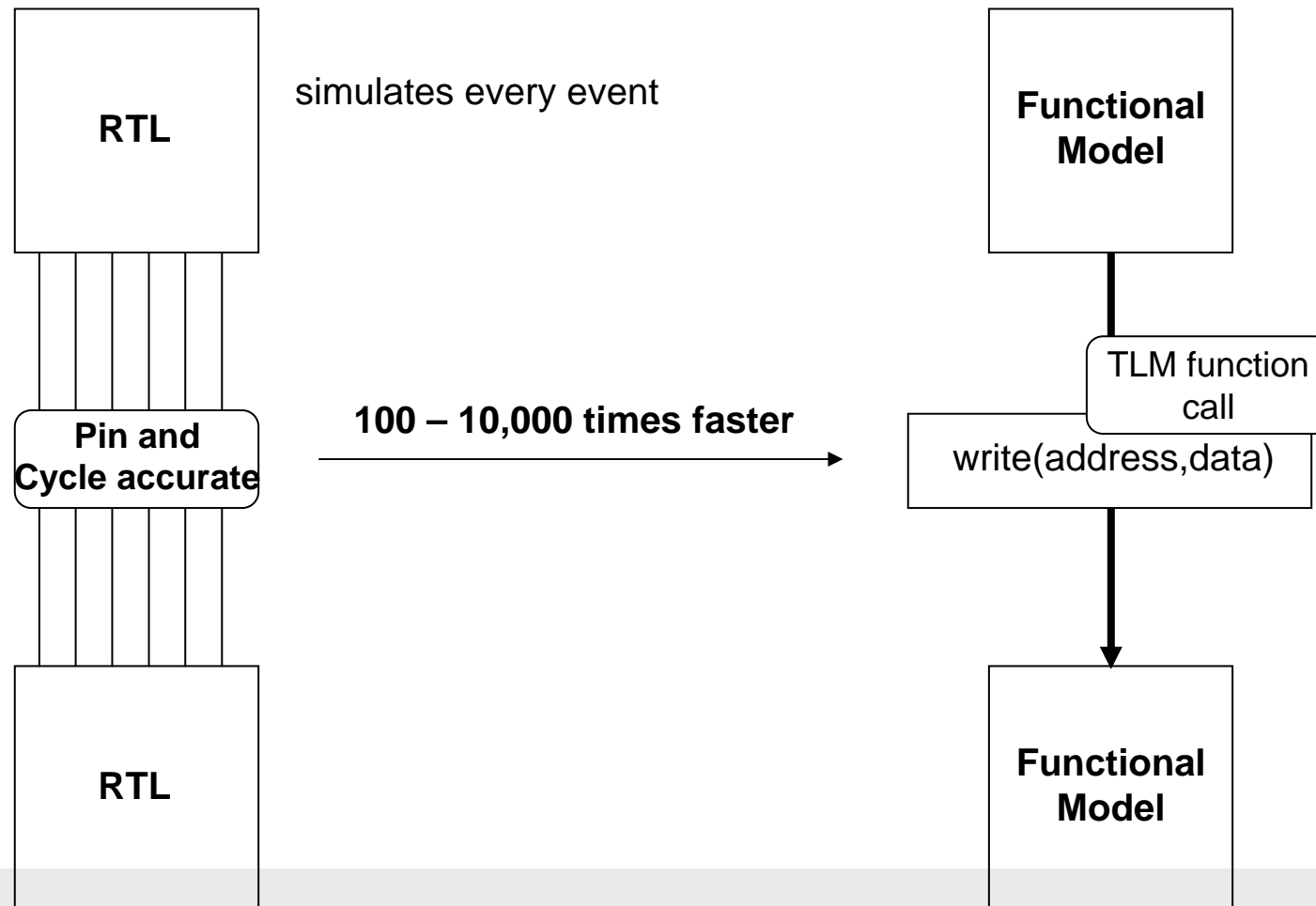


Implement transactions by simple function calls:

Transaction Level Modeling (TLM)



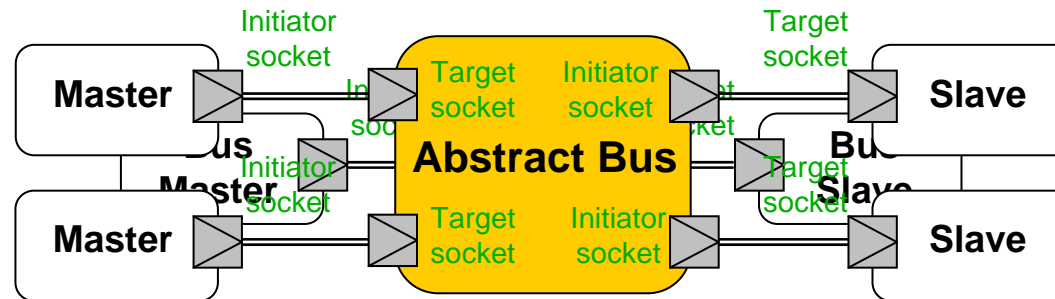
Was ist TLM?





TLM interfaces

Ensuring interoperability: Use of standardized interfaces ⇒ Sockets



Technically: C++ base classes offering virtual functions prototypes

Bus masters: *initiator sockets*

Bus slaves: *target sockets*

Initiator sockets may be bound to target sockets (similar to signal to port binding)

Bus is modeled by a SystemC module offering initiator sockets and target sockets

OSCI Standard for SystemC TLM communication: TLM2.0 (June 2008)



TLM for Behavioral AMS Models

Analogue Behavioral Modeling

Avoid time consuming numerical computations as far as possible

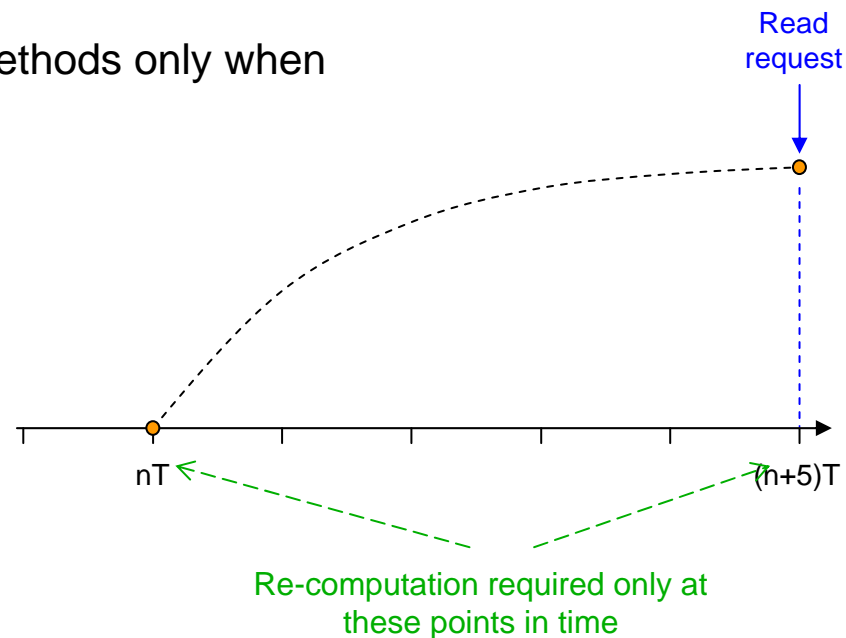
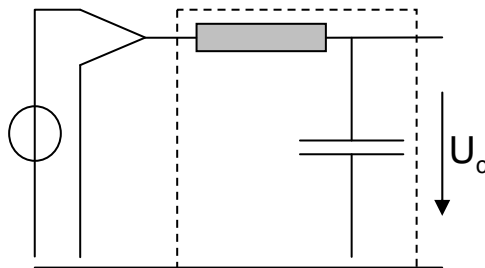
Compute result values by using pre-solved (partial) solutions

Keep time step as large as possible

Best case: invoke modules computations methods only when

- Input value changes
- Output value are requested

Example:

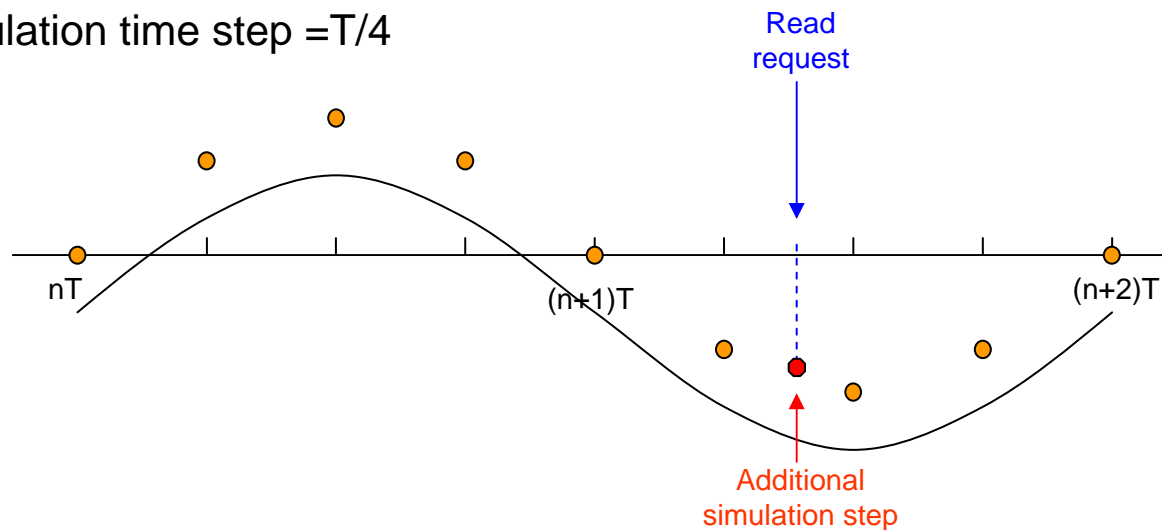




Auxiliary Computation Steps

Example:

- Reading value of an analog sinusoidal signal from AMS slave module
- Simulation time step $= T/4$



Additional simulation step is created by notifying an auxiliary event being added to event list of simulation kernel

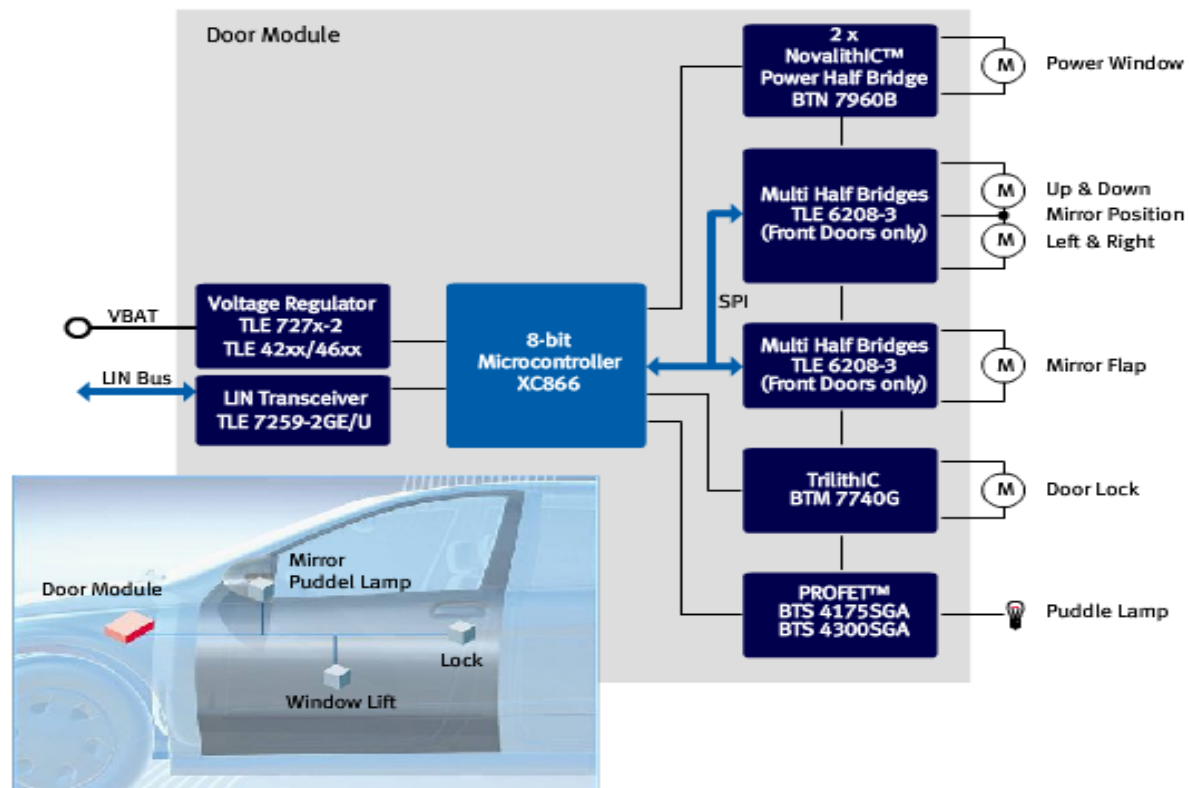


Projektergebnisse

Statistical Overall System Level Simulation



Window Lifter – Substantial Subset of Door ECU





MCU different modeling approaches

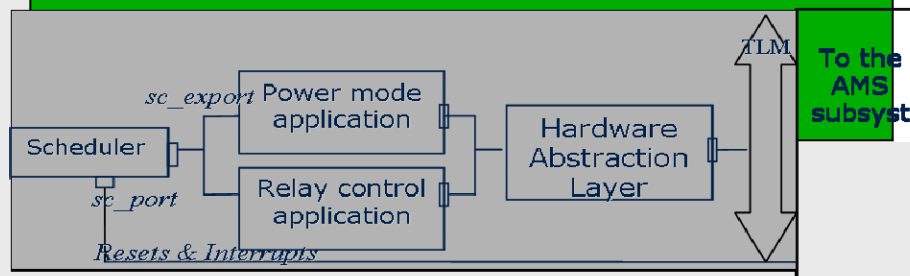
Microcontroller subsystem:

Algorithmic software model

- Application
- Scheduler
- Hardware abstraction layer

Cycle-accurate MCU model

- Instruction set + peripherals
- Embedded software in RAM-model
- Application + firmware
- Run real code (C or assembler)



```
018 ACTIVE:
019 ; enable and configure AMS parts
020 MOV ADDRESS_LIN,#007H
021 ;... and the rest
022 ;*****
023 BACK:
024 ; monitor Hall sensor, switch off if limiters reached
025 ACALL READ_HALL
026 ;*****
027 ; monitor LIN transceiver
028 ; switch on there are switch commands
029 ; switch off if Measurement interface signals it
030 ACALL TEST_LIN
031 ;*****
032 JMP BACK
033
```



To the
AMS
subsynt

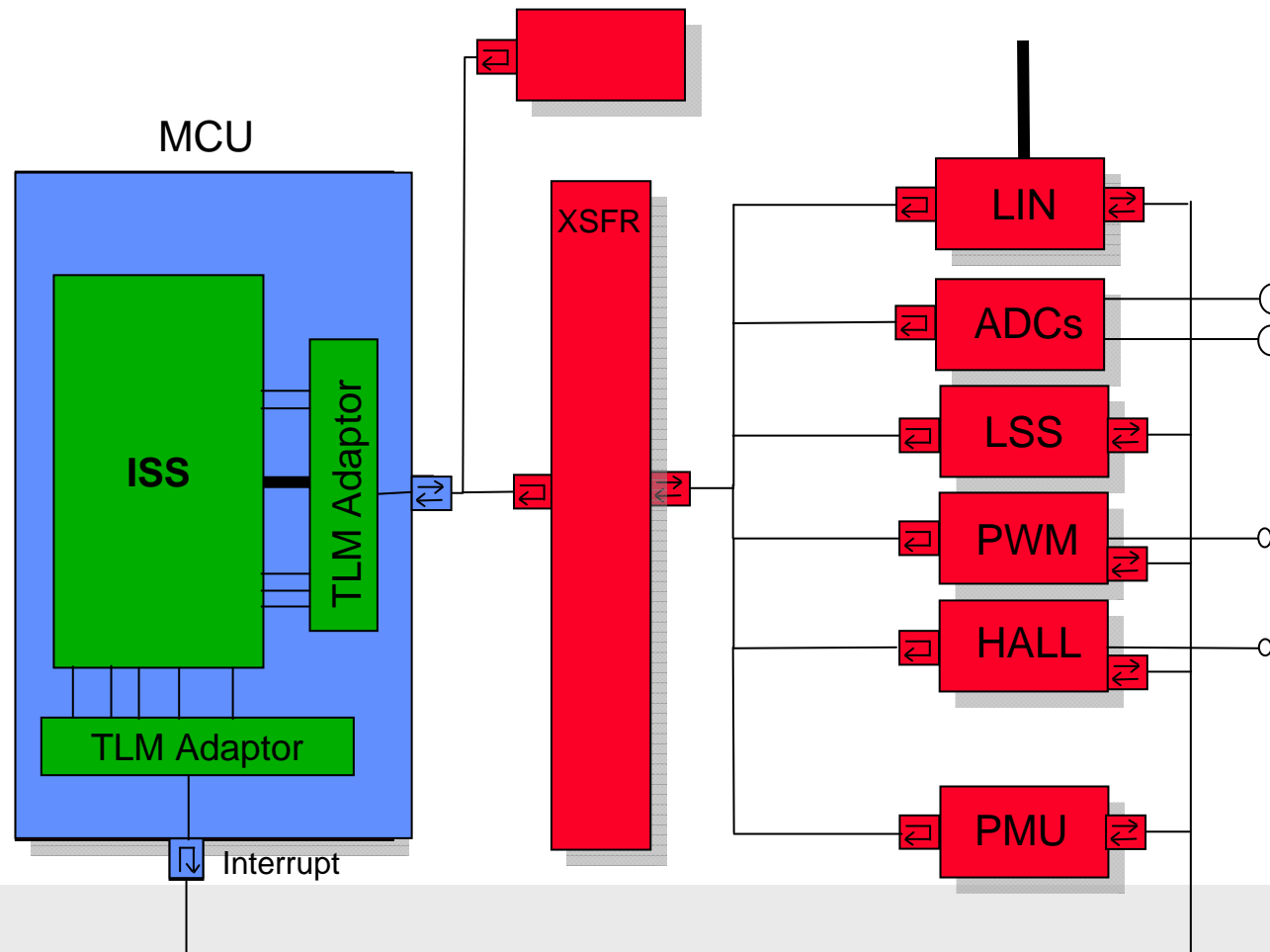
```
232 //sw_on:
233 if (sw_up_cmd & (limit_flag != 0x02) & !lss1_on_flag & !last_meas_result)
234 //sw lss1
235 {
236     lss1_on_flag = 1;
237     ADDRESS_LSS1 = 0x03;
238     //sw hss
239     ADDRESS_HSSw = 0x07;
240     //sw hall supply
241     ADDRESS_HSS = 0x01;
242 }
243 if (sw_down_cmd & (limit_flag != 0x03) & !lss2_on_flag & !last_meas_result)
244 //sw lss2
245 {
246     lss2_on_flag = 1;
247     ADDRESS_LSS2 = 0x03;
248     //sw hss
249     ADDRESS_HSSw = 0x07;
250     //sw hall supply
251     ADDRESS_HSS = 0x01;
252 }
```



To the
AMS
subsynt

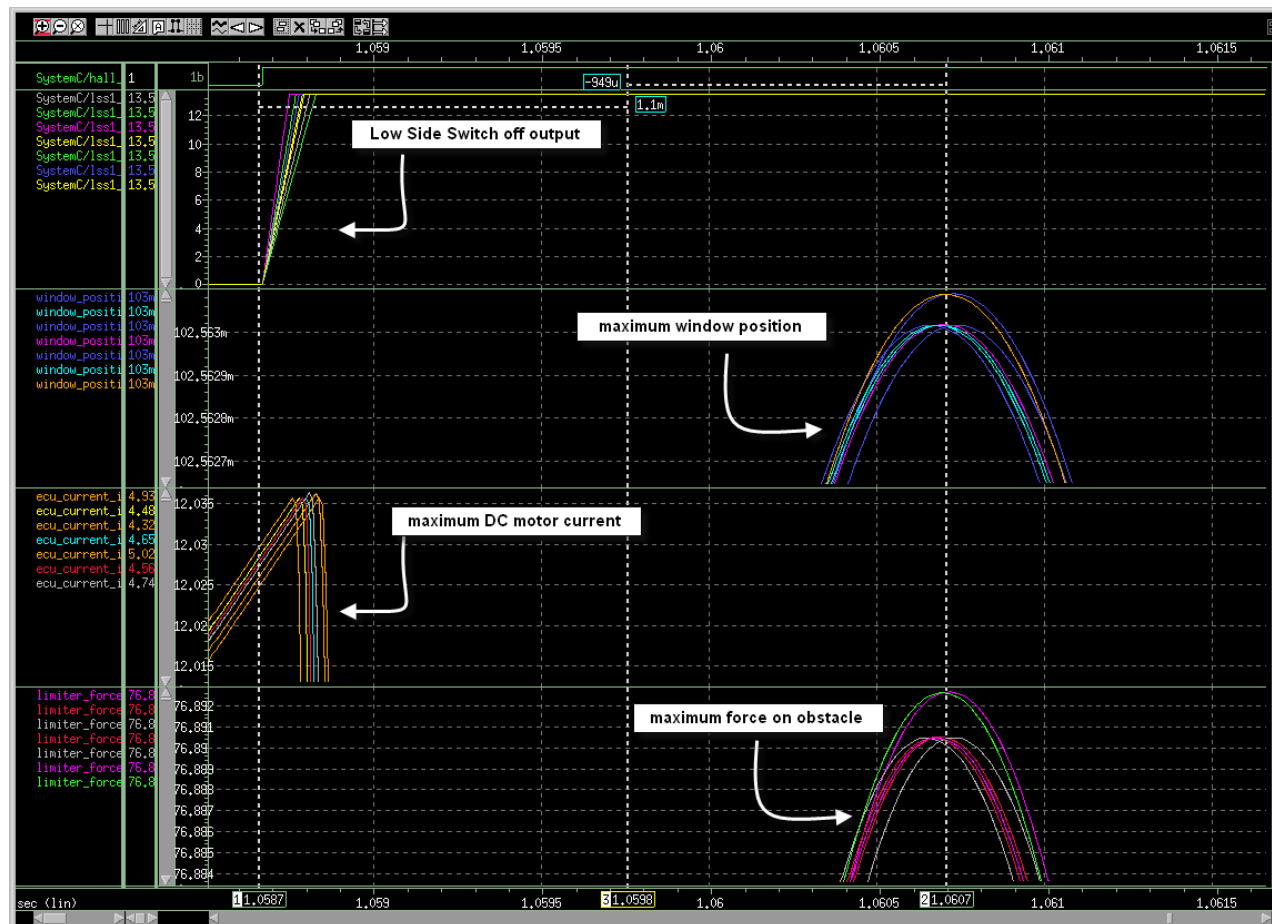


Communication / Architectural Modeling





Tons of Simulations with MC-Varied Parameters



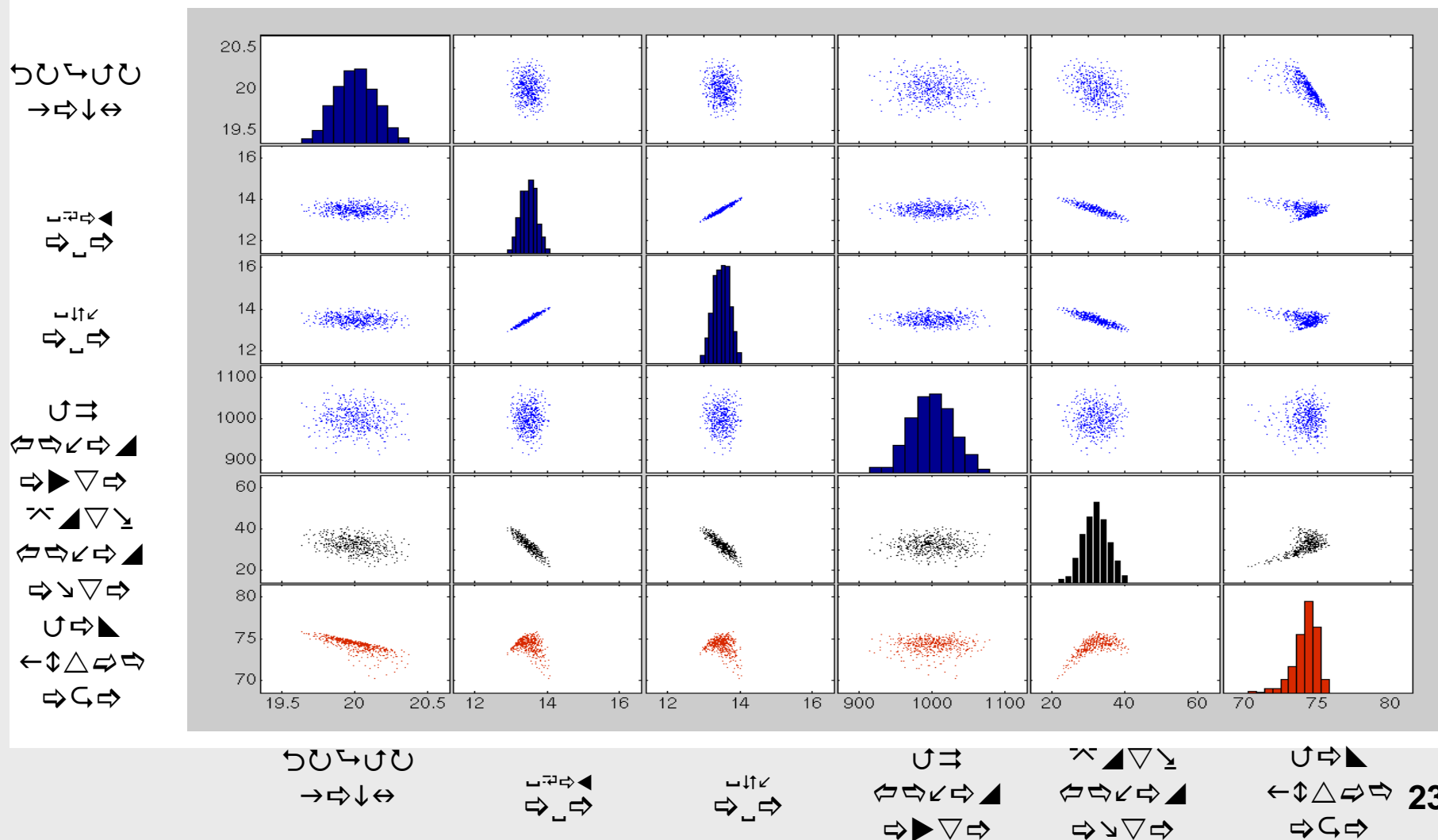
Here's one way to illustrate the results including the parameter and results correlations ...

Edaworkshop Hannover, 04.05./05.05.2010

K. Einwich, Fraunhofer IIS/EAS



System Monte-Carlo Simulation for





How to measure simulation performance:

- Evaluate factor F:

$$F = \frac{\text{simulation time}}{\text{simulated time}}$$

SystemC-AMS

Software on algorithm level,
i.e. directly bound-in

- Factor $F \approx 5 - 100$

SystemC-AMS

Cycle-accurate MCU model

- Factor $F \approx 1.000 - 5.000$

VHDL-AMS / VHDL

- Factor $F \square 100.000 - 1.000.00$

FastMOS (Ultrsim)

- Factor $F \square 1.000.000.000 - 10.000.000.00$ (start-up only)

Spice-alike (Spectre)

- No way!



Gesamtsimulation und Hardware in the Loop

Edaworkshop Hannover, 04.05./05.05.2010

K. Einwich, Fraunhofer IIS/EAS



Gesamtvernetzung BR 212.0 67x ECU's , 8x CAN Bus. 1x MOST



Innenraum - CAN

- ① Kombiinstrument (KI)
- ② Türsteuergerät vorn links (TSG VL)
- ③ Türsteuergerät vorn rechts (TSG VR)
- ④ Türsteuergerät hinten links (TSG HL)
- ⑤ Türsteuergerät hinten rechts (TSG HR)
- ⑥ Sitzverstellung Fahrer (SSG-F)
- ⑦ Sitzverstellung Beifahrer (SSG-BF)
- ⑧ Steuergerät Lenkrahmentheizung
- ⑨ SAM/SRB vorn
- ⑩ SAM/SRB hinten
- ⑪ Panorama Schiebedach
- ⑫ Dachbedieneinheit (DBE)
- ⑬ Steuergerät Anhängererkennung
- ⑭ Steuergerät Keyless Go (KG)
- ⑮ Klimatisierungsautomatik (KLA)
- ⑯ Elektronisches Zündschloss (EVS)
- ⑰ Pneumatikpumpe (FDS-VL, FDS-VR)
- ⑱ COMAND (nicht dargestellt: Audio20, Audio50)
- ⑲ Weight Sensing System (WSS)
- ⑳ Steuergerät Rückfahrkamera
- ㉑ Fahrdynamischer Multikontursitz vorn links
- ㉒ Fahrdynamischer Multikontursitz vorn rechts
- ㉓ Standheizung
- ㉔ Steuergerät Sitzheizung Fond
- ㉕ Multifunktionssteuergerät (MSS)
- ㉖ Steuergerät Kofferraumdeckel Steuerung

Fahrwerk - CAN

- ① Kombiinstrument (KI)
- ② SAM/SRB vorn
- ③ Elektronisches Zündschloss (EVS)
- ④ Steuergerät Radarsensor
- ⑤ Motorsteuergerät ME (M271)
- ⑥ Motorsteuergerät ME (M272, M273)
- ⑦ Motorsteuergerät CDI (OM651)
- ⑧ Motorsteuergerät CDI (OM642)
- ⑨ Motorsteuergerät CDI (OM642)
- ⑩ Motorsteuergerät CDI (OM642)
- ⑪ Motorsteuergerät CDI (OM642)
- ⑫ Motorsteuergerät CDI (OM642)
- ⑬ Motorsteuergerät CDI (OM642)
- ⑭ Motorsteuergerät CDI (OM642)
- ⑮ Motorsteuergerät CDI (OM642)
- ⑯ Motorsteuergerät CDI (OM642)
- ⑰ Motorsteuergerät CDI (OM642)
- ⑱ Motorsteuergerät CDI (OM642)
- ⑲ Motorsteuergerät CDI (OM642)
- ⑳ Motorsteuergerät CDI (OM642)
- ㉑ Motorsteuergerät CDI (OM642)
- ㉒ Motorsteuergerät CDI (OM642)
- ㉓ Motorsteuergerät CDI (OM642)
- ㉔ Motorsteuergerät CDI (OM642)
- ㉕ Motorsteuergerät CDI (OM642)
- ㉖ Motorsteuergerät CDI (OM642)
- ㉗ Motorsteuergerät CDI (OM642)
- ㉘ Motorsteuergerät CDI (OM642)
- ㉙ Motorsteuergerät CDI (OM642)
- ㉚ Motorsteuergerät CDI (OM642)
- ㉛ Motorsteuergerät CDI (OM642)
- ㉜ Motorsteuergerät CDI (OM642)
- ㉝ Motorsteuergerät CDI (OM642)
- ㉞ Motorsteuergerät CDI (OM642)
- ㉟ Motorsteuergerät CDI (OM642)
- ㊱ Motorsteuergerät CDI (OM642)
- ㊲ Motorsteuergerät CDI (OM642)
- ㊳ Motorsteuergerät CDI (OM642)
- ㊴ Motorsteuergerät CDI (OM642)
- ㊵ Motorsteuergerät CDI (OM642)
- ㊶ Motorsteuergerät CDI (OM642)
- ㊷ Motorsteuergerät CDI (OM642)
- ㊸ Motorsteuergerät CDI (OM642)
- ㊹ Motorsteuergerät CDI (OM642)
- ㊺ Motorsteuergerät CDI (OM642)
- ㊻ Motorsteuergerät CDI (OM642)
- ㊼ Motorsteuergerät CDI (OM642)
- ㊽ Motorsteuergerät CDI (OM642)
- ㊾ Motorsteuergerät CDI (OM642)
- ㊿ Motorsteuergerät CDI (OM642)

Frontbereich - CAN

- ① SAM/SRB vorn
- ② Steuergerät Xenon Scheinwerfer links
- ③ Steuergerät Xenon Scheinwerfer rechts

Antriebs - CAN

- ① Motorsteuergerät ME (M271)
- ② Motorsteuergerät ME (M272, M273)
- ③ Motorsteuergerät CDI (OM651)
- ④ Motorsteuergerät CDI (OM642)
- ⑤ Motorsteuergerät CDI (OM642)
- ⑥ Motorsteuergerät CDI (OM642)
- ⑦ Motorsteuergerät CDI (OM642)
- ⑧ Motorsteuergerät CDI (OM642)
- ⑨ Motorsteuergerät CDI (OM642)
- ⑩ Motorsteuergerät CDI (OM642)
- ⑪ Motorsteuergerät CDI (OM642)
- ⑫ Motorsteuergerät CDI (OM642)
- ⑬ Motorsteuergerät CDI (OM642)
- ⑭ Motorsteuergerät CDI (OM642)
- ⑮ Motorsteuergerät CDI (OM642)
- ⑯ Motorsteuergerät CDI (OM642)
- ⑰ Motorsteuergerät CDI (OM642)
- ⑱ Motorsteuergerät CDI (OM642)
- ⑲ Motorsteuergerät CDI (OM642)
- ⑳ Motorsteuergerät CDI (OM642)
- ㉑ Motorsteuergerät CDI (OM642)
- ㉒ Motorsteuergerät CDI (OM642)
- ㉓ Motorsteuergerät CDI (OM642)
- ㉔ Motorsteuergerät CDI (OM642)
- ㉕ Motorsteuergerät CDI (OM642)
- ㉖ Motorsteuergerät CDI (OM642)
- ㉗ Motorsteuergerät CDI (OM642)
- ㉘ Motorsteuergerät CDI (OM642)
- ㉙ Motorsteuergerät CDI (OM642)
- ㉚ Motorsteuergerät CDI (OM642)
- ㉛ Motorsteuergerät CDI (OM642)
- ㉜ Motorsteuergerät CDI (OM642)
- ㉝ Motorsteuergerät CDI (OM642)
- ㉞ Motorsteuergerät CDI (OM642)
- ㉟ Motorsteuergerät CDI (OM642)
- ㊱ Motorsteuergerät CDI (OM642)
- ㊲ Motorsteuergerät CDI (OM642)
- ㊳ Motorsteuergerät CDI (OM642)
- ㊴ Motorsteuergerät CDI (OM642)
- ㊵ Motorsteuergerät CDI (OM642)
- ㊶ Motorsteuergerät CDI (OM642)
- ㊷ Motorsteuergerät CDI (OM642)
- ㊸ Motorsteuergerät CDI (OM642)
- ㊹ Motorsteuergerät CDI (OM642)
- ㊺ Motorsteuergerät CDI (OM642)
- ㊻ Motorsteuergerät CDI (OM642)
- ㊼ Motorsteuergerät CDI (OM642)
- ㊽ Motorsteuergerät CDI (OM642)
- ㊾ Motorsteuergerät CDI (OM642)
- ㊿ Motorsteuergerät CDI (OM642)

Antriebssensor - CAN

- ① Motorsteuergerät ME (M272)
- ② Motorsteuergerät CDI (OM642)
- ③ Motorsteuergerät CDI (OM642)
- ④ Motorsteuergerät CDI (OM642)
- ⑤ Motorsteuergerät CDI (OM642)
- ⑥ Motorsteuergerät CDI (OM642)
- ⑦ Motorsteuergerät CDI (OM642)
- ⑧ Motorsteuergerät CDI (OM642)
- ⑨ Motorsteuergerät CDI (OM642)
- ⑩ Motorsteuergerät CDI (OM642)
- ⑪ Motorsteuergerät CDI (OM642)
- ⑫ Motorsteuergerät CDI (OM642)
- ⑬ Motorsteuergerät CDI (OM642)
- ⑭ Motorsteuergerät CDI (OM642)
- ⑮ Motorsteuergerät CDI (OM642)
- ⑯ Motorsteuergerät CDI (OM642)
- ⑰ Motorsteuergerät CDI (OM642)
- ⑱ Motorsteuergerät CDI (OM642)
- ⑲ Motorsteuergerät CDI (OM642)
- ⑳ Motorsteuergerät CDI (OM642)
- ㉑ Motorsteuergerät CDI (OM642)
- ㉒ Motorsteuergerät CDI (OM642)
- ㉓ Motorsteuergerät CDI (OM642)
- ㉔ Motorsteuergerät CDI (OM642)
- ㉕ Motorsteuergerät CDI (OM642)
- ㉖ Motorsteuergerät CDI (OM642)
- ㉗ Motorsteuergerät CDI (OM642)
- ㉘ Motorsteuergerät CDI (OM642)
- ㉙ Motorsteuergerät CDI (OM642)
- ㉚ Motorsteuergerät CDI (OM642)
- ㉛ Motorsteuergerät CDI (OM642)
- ㉜ Motorsteuergerät CDI (OM642)
- ㉝ Motorsteuergerät CDI (OM642)
- ㉞ Motorsteuergerät CDI (OM642)
- ㉟ Motorsteuergerät CDI (OM642)
- ㊱ Motorsteuergerät CDI (OM642)
- ㊲ Motorsteuergerät CDI (OM642)
- ㊳ Motorsteuergerät CDI (OM642)
- ㊴ Motorsteuergerät CDI (OM642)
- ㊵ Motorsteuergerät CDI (OM642)
- ㊶ Motorsteuergerät CDI (OM642)
- ㊷ Motorsteuergerät CDI (OM642)
- ㊸ Motorsteuergerät CDI (OM642)
- ㊹ Motorsteuergerät CDI (OM642)
- ㊺ Motorsteuergerät CDI (OM642)
- ㊻ Motorsteuergerät CDI (OM642)
- ㊼ Motorsteuergerät CDI (OM642)
- ㊽ Motorsteuergerät CDI (OM642)
- ㊾ Motorsteuergerät CDI (OM642)
- ㊿ Motorsteuergerät CDI (OM642)

Fahrdynamik - CAN

- ① Steuergerät Radarsensor
- ② Elektronisches Stabilitätsprogramm (ESP)
- ③ Sensorcluster
- ④ Steuereinheit DISTRONIC

Diagnose - CAN

- ① SAM/SRB vorn
- ② Steuergerät Notrufsystem

Telematik - CAN

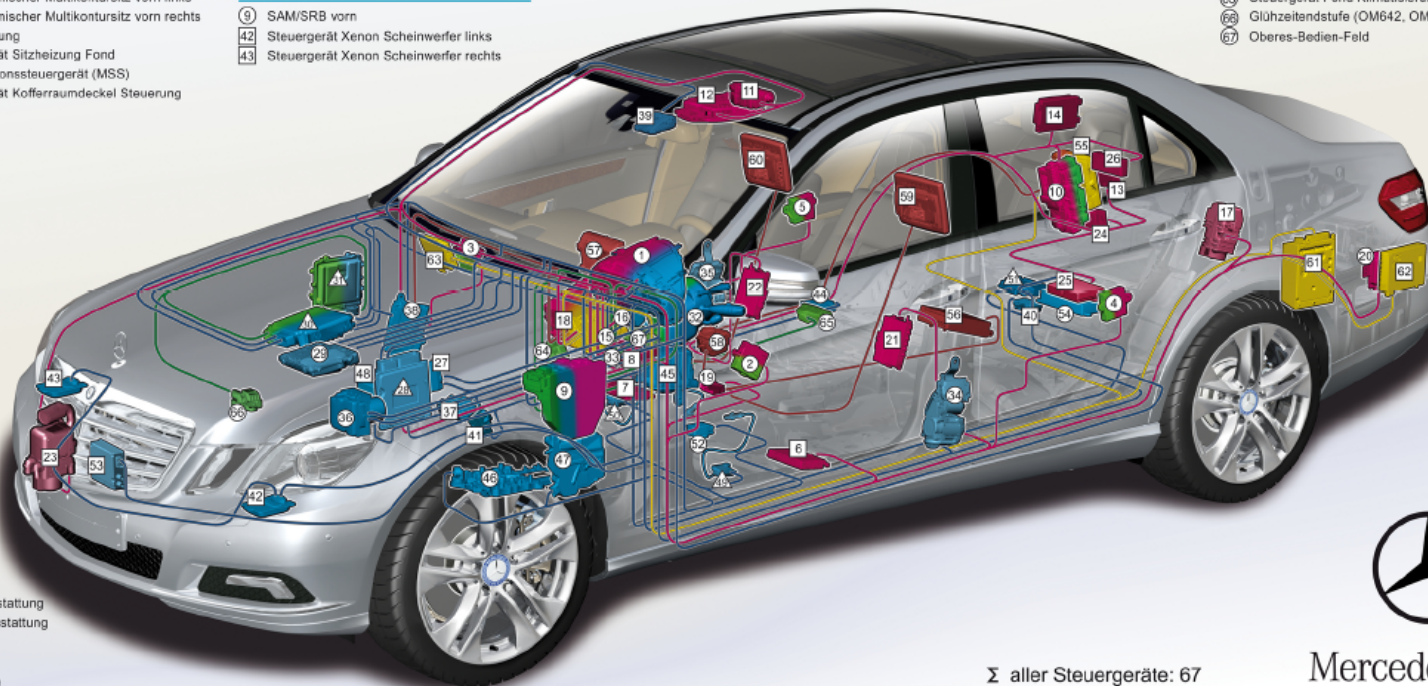
- ① COMAND (nicht dargestellt: Audio20, Audio50)
- ② TV-Tuner
- ③ DVD-Player
- ④ Zentralsdisplay
- ⑤ Audio/COMAND Bedieneinheit
- ⑥ Fonddisplay links
- ⑦ Fonddisplay rechts

MOST - Ring

- ① COMAND (nicht dargestellt: Audio20, Audio50)
- ② TV-Tuner
- ③ Soundverstärker
- ④ SDAR/HD-Tuner oder Digital Radio (DAB)
- ⑤ Media Interface

Private-Bus

- ① Türsteuergerät vorn links (TSG VL)
- ② Türsteuergerät vorn rechts (TSG VR)
- ③ Türsteuergerät hinten links (TSG HL)
- ④ Türsteuergerät hinten rechts (TSG HR)
- ⑤ SAM/SRB vorn
- ⑥ SAM/SRB hinten
- ⑦ Klimatisierungsautomatik (KLA)
- ⑧ Elektronisches Zündschloss (EVS)
- ⑨ Motorsteuergerät CDI (OM651)
- ⑩ Motorsteuergerät CDI (OM642)
- ⑪ Motorsteuergerät CDI (OM642)
- ⑫ Motorsteuergerät CDI (OM642)
- ⑬ Motorsteuergerät CDI (OM642)
- ⑭ Motorsteuergerät CDI (OM642)
- ⑮ Motorsteuergerät CDI (OM642)
- ⑯ Motorsteuergerät CDI (OM642)
- ⑰ Motorsteuergerät CDI (OM642)
- ⑱ Motorsteuergerät CDI (OM642)
- ⑲ Motorsteuergerät CDI (OM642)
- ⑳ Motorsteuergerät CDI (OM642)
- ㉑ Motorsteuergerät CDI (OM642)
- ㉒ Motorsteuergerät CDI (OM642)
- ㉓ Motorsteuergerät CDI (OM642)
- ㉔ Motorsteuergerät CDI (OM642)
- ㉕ Motorsteuergerät CDI (OM642)
- ㉖ Motorsteuergerät CDI (OM642)
- ㉗ Motorsteuergerät CDI (OM642)
- ㉘ Motorsteuergerät CDI (OM642)
- ㉙ Motorsteuergerät CDI (OM642)
- ㉚ Motorsteuergerät CDI (OM642)
- ㉛ Motorsteuergerät CDI (OM642)
- ㉜ Motorsteuergerät CDI (OM642)
- ㉝ Motorsteuergerät CDI (OM642)
- ㉞ Motorsteuergerät CDI (OM642)
- ㉟ Motorsteuergerät CDI (OM642)
- ㊱ Motorsteuergerät CDI (OM642)
- ㊲ Motorsteuergerät CDI (OM642)
- ㊳ Motorsteuergerät CDI (OM642)
- ㊴ Motorsteuergerät CDI (OM642)
- ㊵ Motorsteuergerät CDI (OM642)
- ㊶ Motorsteuergerät CDI (OM642)
- ㊷ Motorsteuergerät CDI (OM642)
- ㊸ Motorsteuergerät CDI (OM642)
- ㊹ Motorsteuergerät CDI (OM642)
- ㊺ Motorsteuergerät CDI (OM642)
- ㊻ Motorsteuergerät CDI (OM642)
- ㊼ Motorsteuergerät CDI (OM642)
- ㊽ Motorsteuergerät CDI (OM642)
- ㊾ Motorsteuergerät CDI (OM642)
- ㊿ Motorsteuergerät CDI (OM642)



- Serienausstattung
- Sonderausstattung
- △ Variante

Stand: Februar 2009

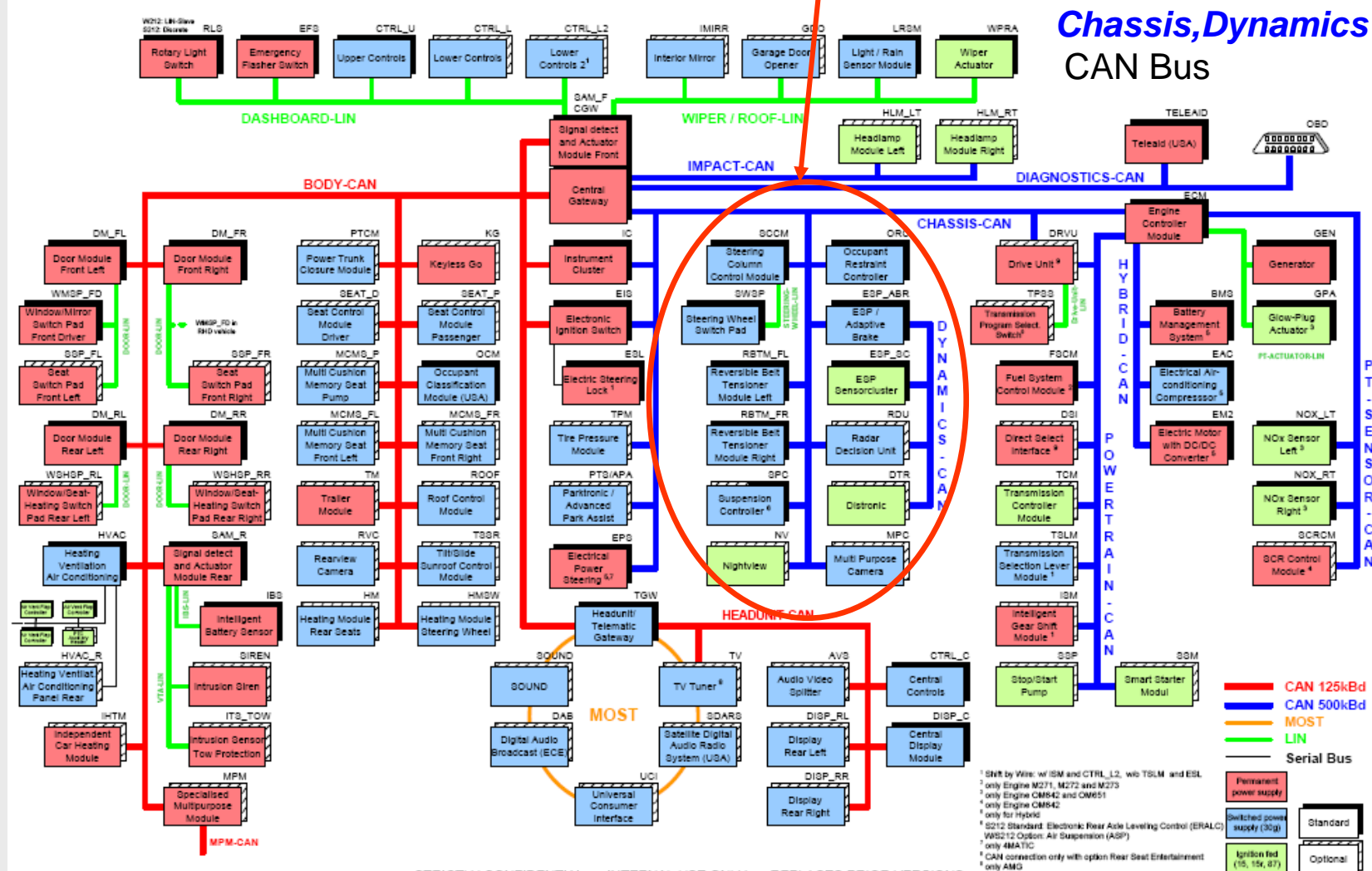
Σ aller Steuergeräte: 67



Edaworkshop Hannover, 04.05./05.05.2010

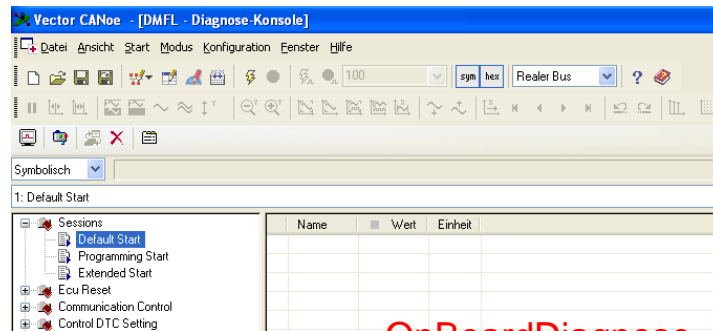
K. Einwich, Fraunhofer IIS/EAS
CAN Busarchitektur BR 221

Fahrerassistenzsysteme (DISTRONIC, MPC, NightView)

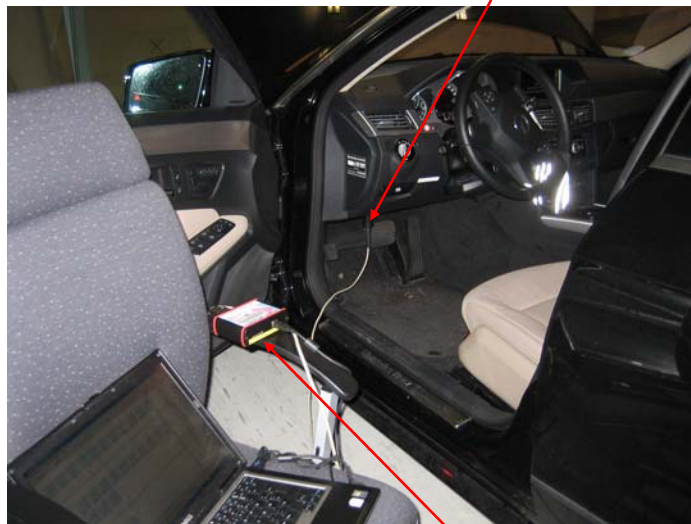




Hardware zur Diagnose am Fahrzeug : CANoe mit CAN Case/ Card



OnBoardDiagnose



CAN Case

CAN Case



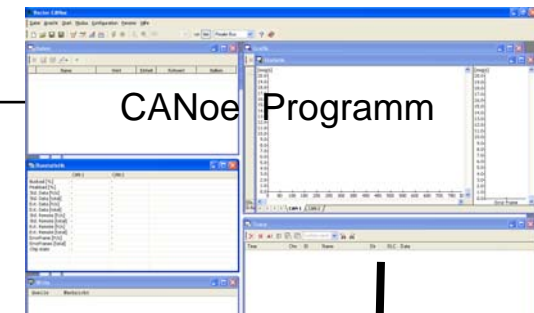
Alternative Diagnose hardware



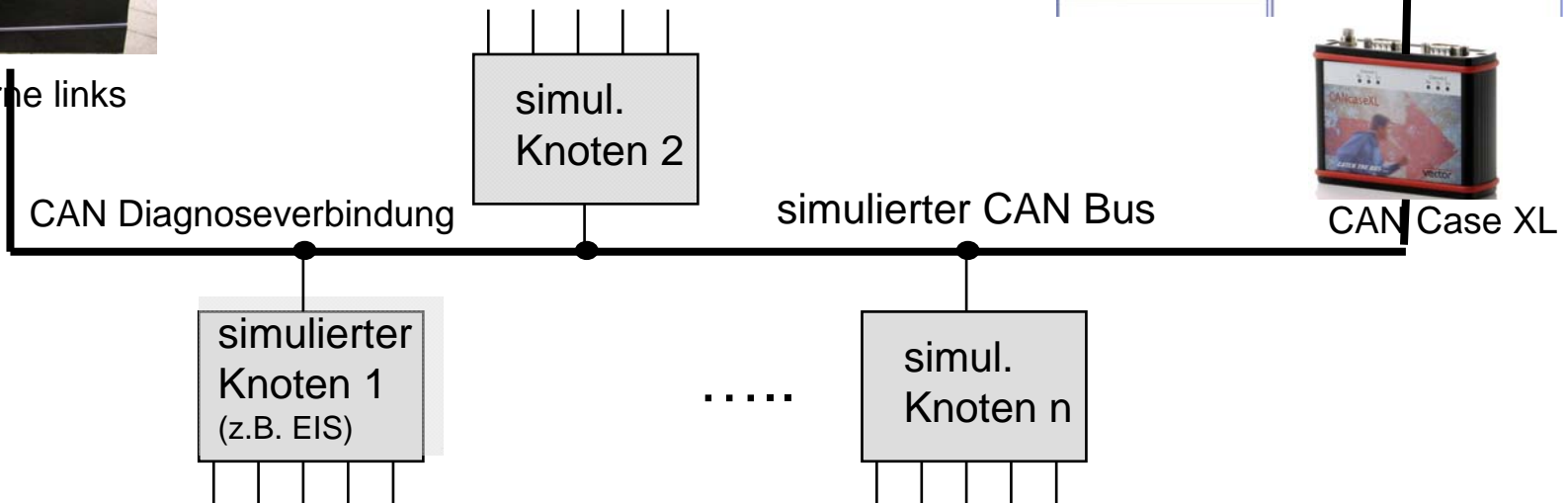
Diagnose eines Steuergeräts durch CAN Bus Modellierung



TSG vorne links



CAN Case XL

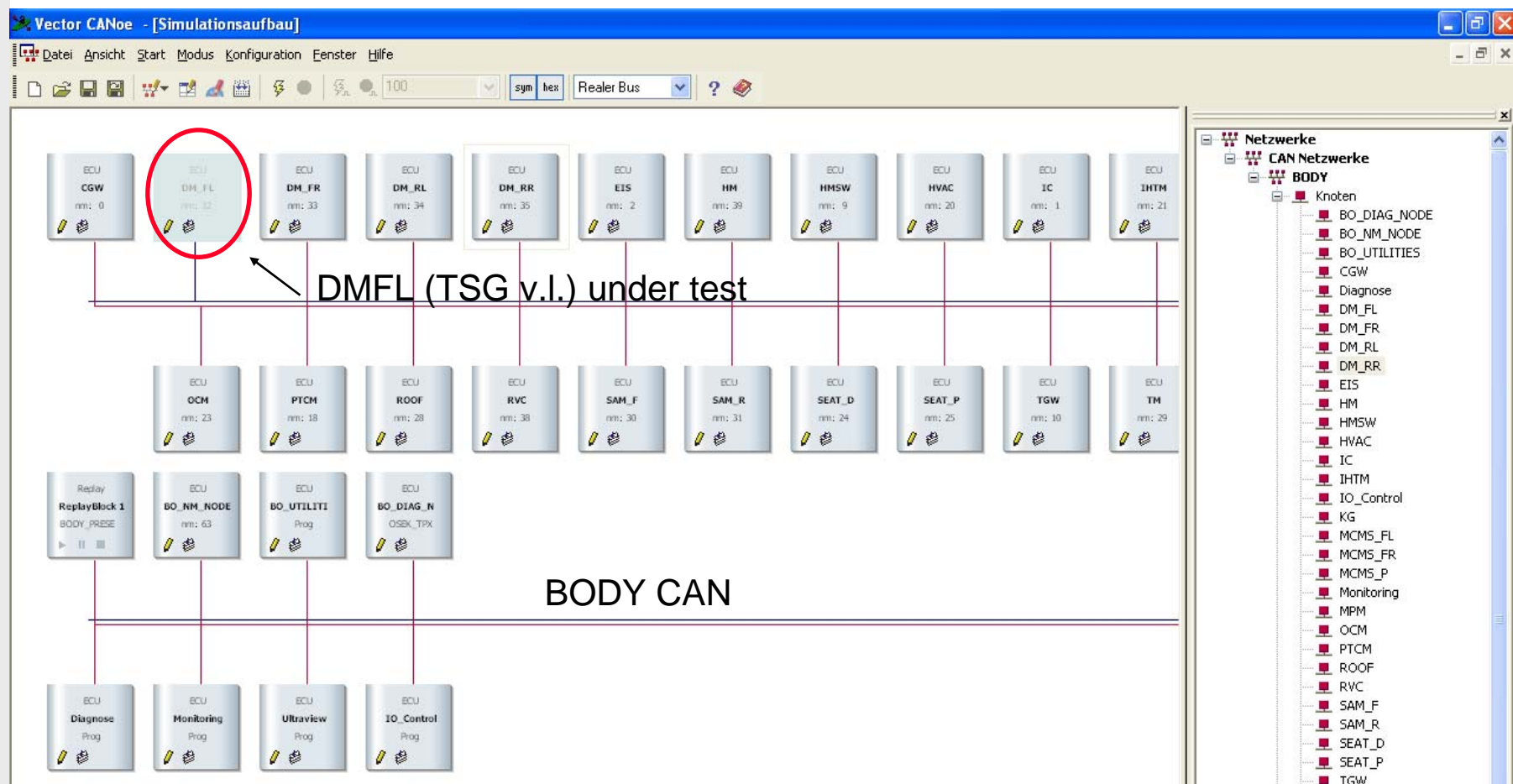


K.15

Virtuelles Gesamtsystem : **CAN- Simulation der nicht angeschlossenen Steuergeräte**

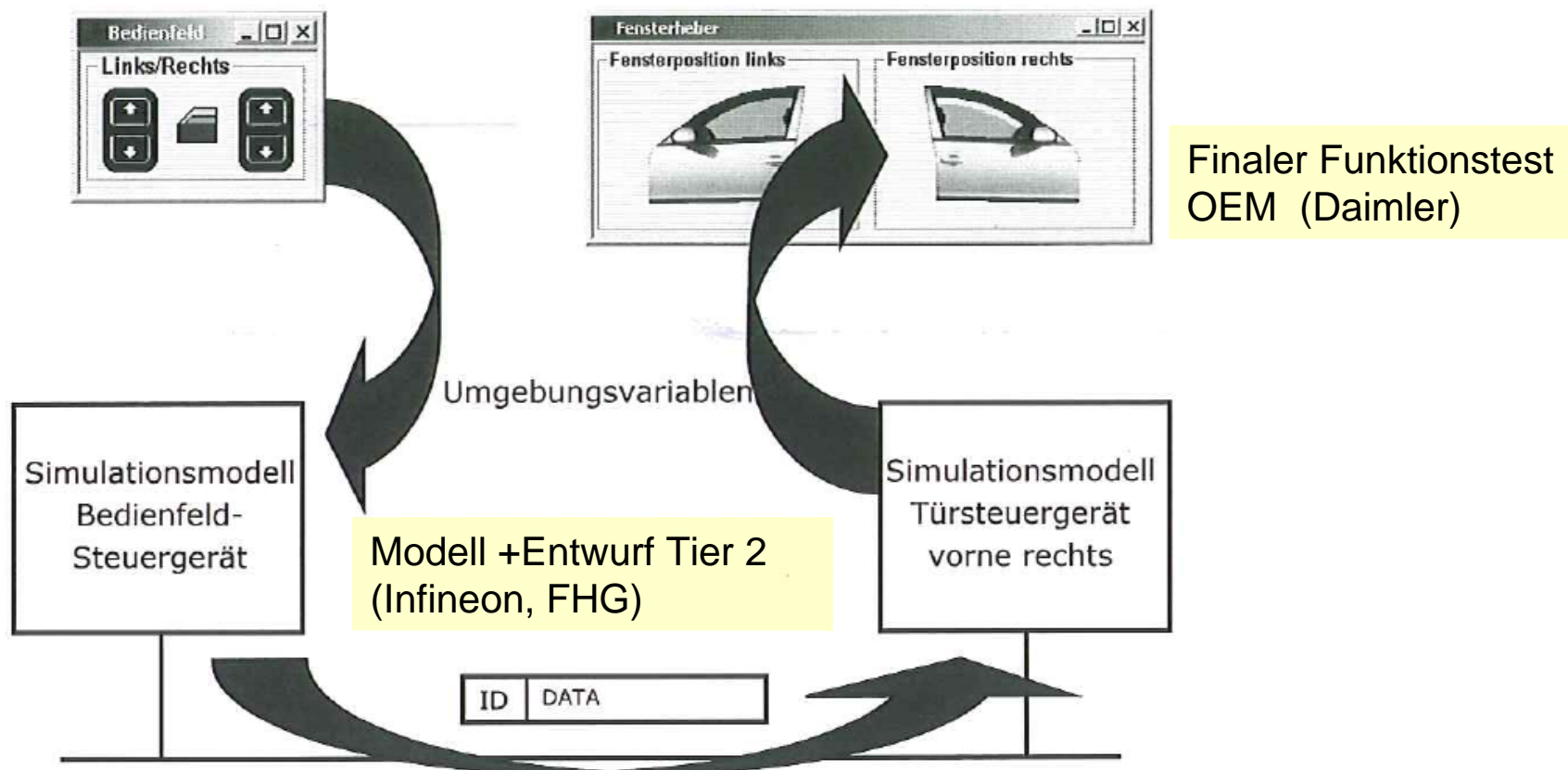


CAN Bus Simulation, BODY CAN BR212



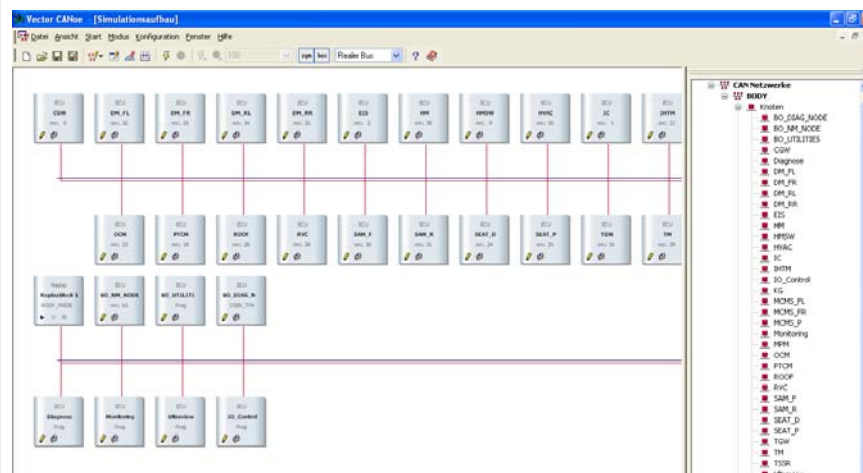


Simulieren und Testen mit CANoe : komplette Simulation

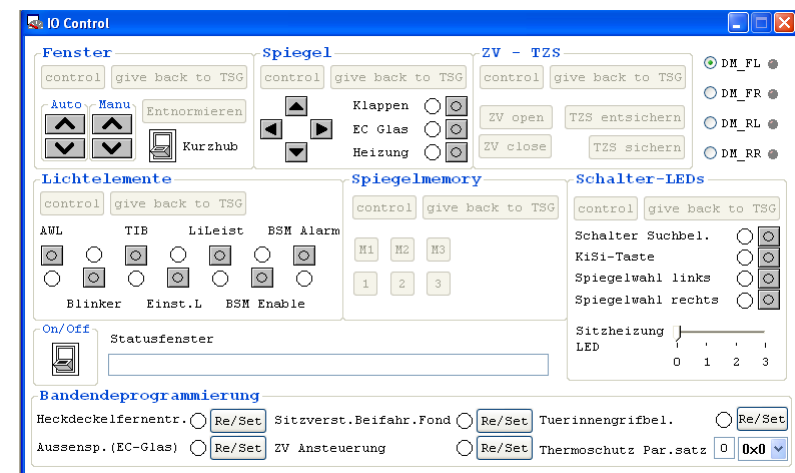




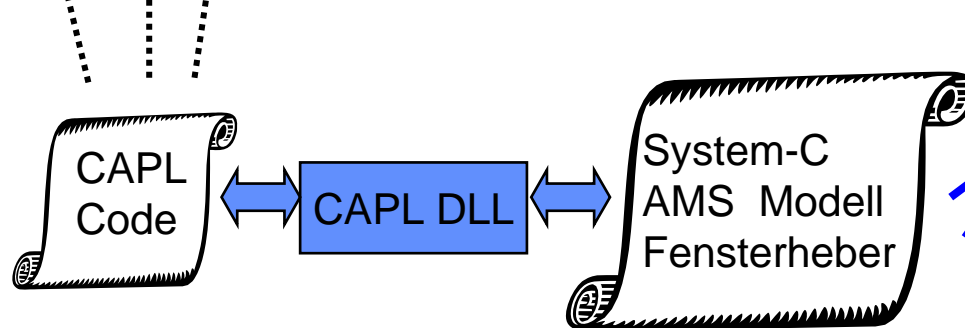
b) Schnittstelle CANoe und System-C AMS Modell via CAPL DLL



Simulationsblöcke



Steuerelement



Schnittstelle

Simulations Modell
Fensterheber





SystemC AMS – CANoe Integration

The screenshot displays the Vector CANoe software interface, which is used for CAN bus simulation. The main window is titled "Window Lifter" and contains several panels:

- Controller:** Includes a "Window Control" button, a "Time" display showing "Current CAPL Time:" and "Current SystemC AMS Time:", and a "Simulation Results" section showing "Pane Position [m]".
- Actors:** Displays a 3D model of a red car.
- Simulation Results:** Shows a "Pane Position [m]" display and a "Insert Obstacle at 38cm (ON/OFF)" button.

Overlaid on the bottom right is a smaller window titled "Vector CANoe" showing a "Simulationsaufbau" (Simulation Setup) diagram. This diagram illustrates the connection between various components, including a "CAN-Netzwerke" (CAN Network) and a "BODY" (Body) component. A red circle highlights a specific component in the diagram, and a red arrow points to a text box that reads:

SystemC AMS virtual Door Module

The diagram also shows a list of components on the right side, including "IO_Control", "IO", "MCHS_FL", "MCHS_FR", "MCHS_P", "Monitoring", "MCHS", "OCM", "PTCM", "ROOF", "SVC", "SAM_P", "SAM_R", "SEAT_D", "SEAT_P", "TGW", "TM", and "TSGR".



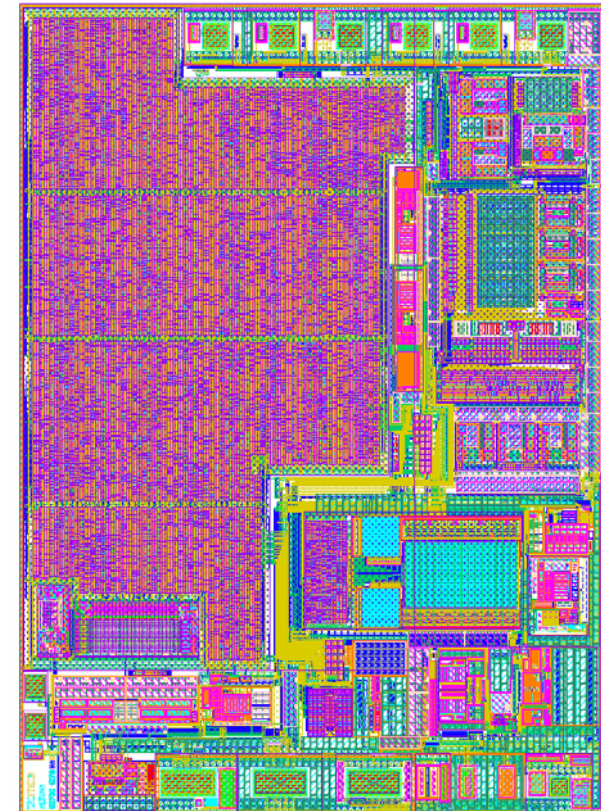
Verifikation



Introduction

ZMD's objective: serve customer with functional correct prototypes with silicon from 1st tape out

This requires thorough verification, but:
The verification problem of analog and mixed-signal systems is not solved as good as of digital systems





Outline

Development of a Verification Platform

- Tool for
 - Planning of verification
 - Execution of verifications
 - Evaluation and Summary of verification results
- Usable together with any verification tool (simulator as well as other tools)
- Usable for all designers: analog, digital and mixed-signal
- Focus on functional features
but
- Characterization of analog blocks shall be supported as well



Tool Supported Verification Planning & Tracking

For verification planning all designers, analog, digital and mixed-signal need work closely together

Analog designers work with the GUI

- Verification of performance parameters is carried out by analog designer

Digital designers / Verification engineers more or less rely on command line

- Test bench outputs are analysed with scripts
- Verification Engineer may carry out mixed-signal verification of functional features

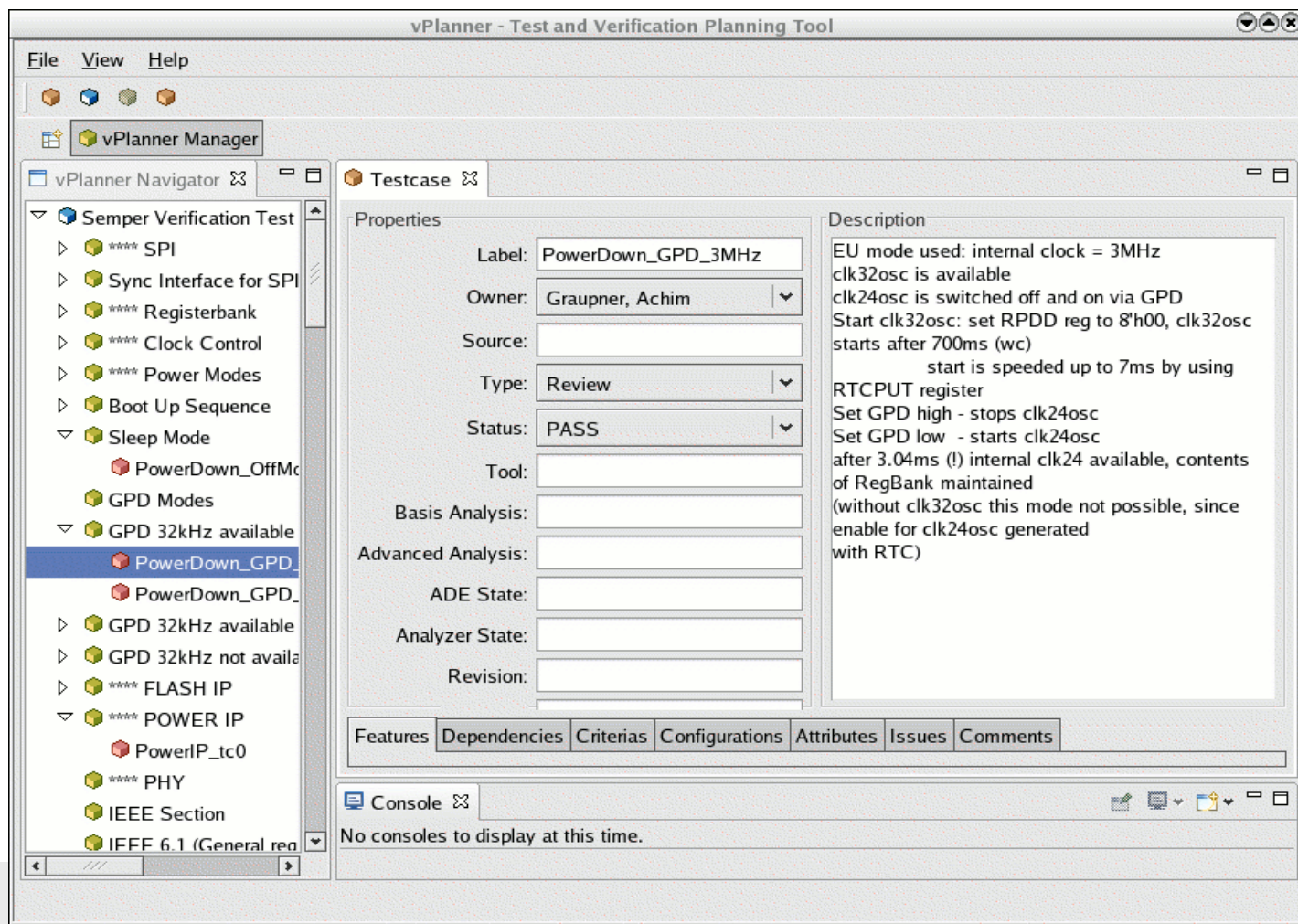
Verification status is either tracked manually (Excel-Sheets etc) or one domain is partly automated

Commercial tools are

- Dedicated for rather large systems
- Optimized for one of the domains
- tool development for Verification Planning, Tracking, Results Analysis and automated starting of verification runs

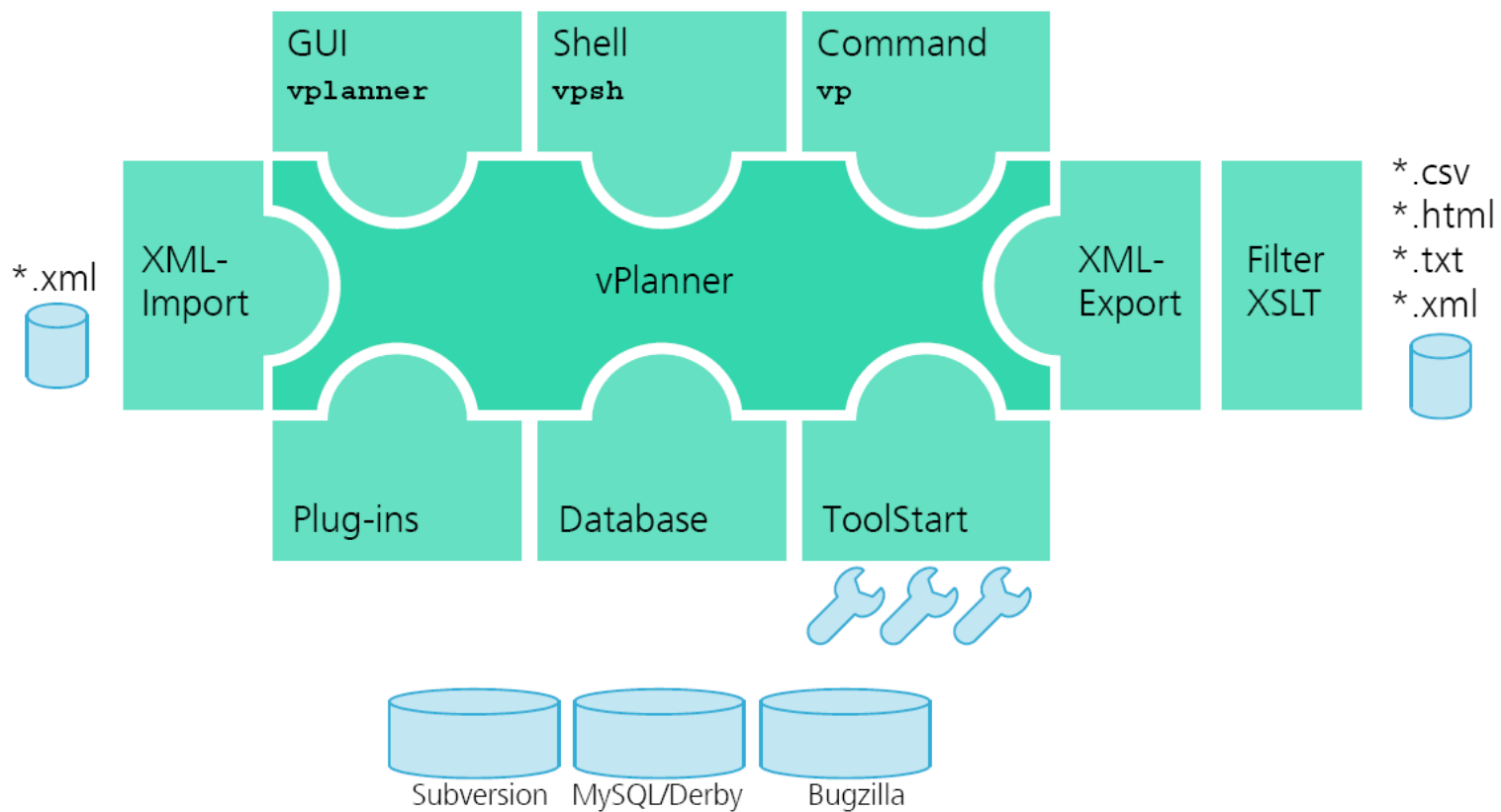


VPlanner (FhG) GUI





VPlanner (FhG) Software Architecture





vPlanner – Status & Outlook

1st working prototype, limited functionality

- GUI & database access
- Planning possible: Will be used soon in a study project

Further development

- Start of any verification task
- Inclusion of report functions
- Interface for scripting

Edaworkshop Hannover, 04.05./05.05.2010

K. Einwich, Fraunhofer IIS/EAS



Characterization: zmdAnalyser

In-house tool for verification of performance parameters

Automated evaluation of parameters incl. Limits

Result summary for storage into design database

OK Cancel Apply Help

Measurements Details ☒ Min/Max Value, Description

Add Delete Move

No	Name	Description	Min	Max	Calculator Expression	Open Calculator
1	fosc	resonance frequency	15.9e6	16.1e6	$\text{xmax}(\text{mag}(\text{VF}("/\text{oac}")))$	<<
2	Gain	maximum of the ampl.	1.8	100	$\text{value}(\text{mag}((\text{VF}("/\text{oac}') / \text{VF}("/\text{ioc}')))) \text{ xmax}(\text{mag}(\text{VF}("/\text{oac}'))))$	<<
3	iocs	current consumption	0	3e-3	$\text{value}(\text{mag}(\text{IF}("/\text{V1}/\text{MINUS}")) \text{ xmax}(\text{mag}(\text{VF}("/\text{oac}'))))$	<<

```
3
4 Measurement
5 Min @ Run
6 Max @ Run
7 Mean
8 Std deviation 1sigma
9 Std deviation 1sigma %
10 Out of Min Spec
11 Out of Max Spec
12 Error
13 Individual pass samples %
14 Total pass samples %
15 CPK
16
17
18
```

I_10uA_DELTA_overTemp	@ 1631	I_10uA_DELTA_overTemp_inPROZ	@ 1631
1.3071648947e-07		1.3071648947e+00	
3.9058181887e-07	@ 2384	3.9058181887e+00	@ 2384
1.9608181912e-07		1.9608181912e+00	
4.0208694676e-08		4.0208694676e-01	
2.0506079991e+01		2.0506079991e+01	
0		0	
0		0	
0		0	
1.0000000000e+02		1.0000000000e+02	
0.0000000000e+00		0.0000000000e+00	
3.3144068416e+03		1.6255341512e+00	

Edaworkshop Hannover, 04.05./05.05.2010

K. Einwich, Fraunhofer IIS/EAS



Austausch von Ergebnissen zwischen RapidMPSoC und AutoSUN

