

# Vorlesung Automotive Software Engineering Teil 8-2 Beispiele aus der Praxis

TU Dresden, Fakultät Informatik

Sommersemester 2012

Prof. Dr. rer. nat. Bernhard Hohlfeld

[bernhard.hohlfeld@daad-alumni.de](mailto:bernhard.hohlfeld@daad-alumni.de)

# Entwicklung einer Antriebssteuerung für ein Hybridfahrzeug in einer Rapid Prototyping-Umgebung

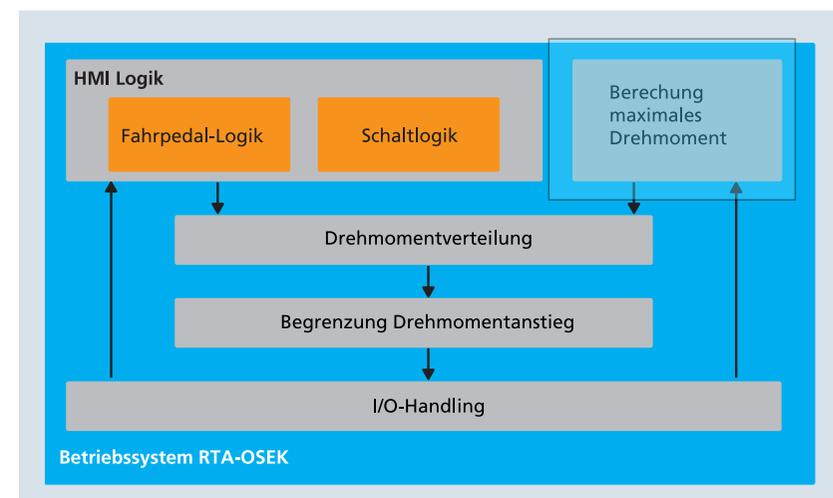
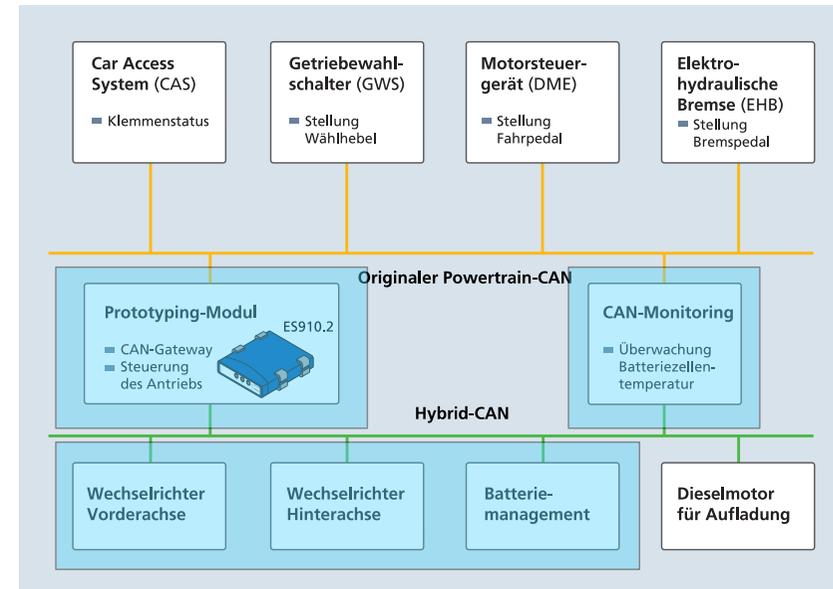


1. Einleitung
2. Das Fahrzeug
3. Die Bordnetz-Architektur
- 4. Funktionsentwicklung mit ASCET**
  1. Fahrpedal-Logik
  - 2. Berechnung des maximalen Drehmoments**
  3. Drehmomentverteilung
  4. Drehmomentanstiegsbegrenzung
  5. Funktionsmodellierung
5. Die Erprobung
  1. Offline-Simulation
  2. Rapid Prototyping
6. Höhere Antriebsfunktion
7. Fazit
8. Abkürzungsverzeichnis

# Berechnung des maximalen Drehmoments



- Die wassergekühlten Wechselrichter und Motoren sowie die Batterien werden temperaturüberwacht. Um ein Überhitzen der Komponenten zu vermeiden, werden in Abhängigkeit von den Temperaturen, die maximalen Drehmomentwerte für beide Achsen ermittelt. Hierzu werden, für jede Achse getrennt, entsprechende Kennlinien verwendet. Ein weiterer Einflussfaktor ist der Ladezustand der Batterie. Durch die Begrenzung des Drehmomentwertes wird ein Überladen bzw. Tiefentladen der Batterie verhindert.



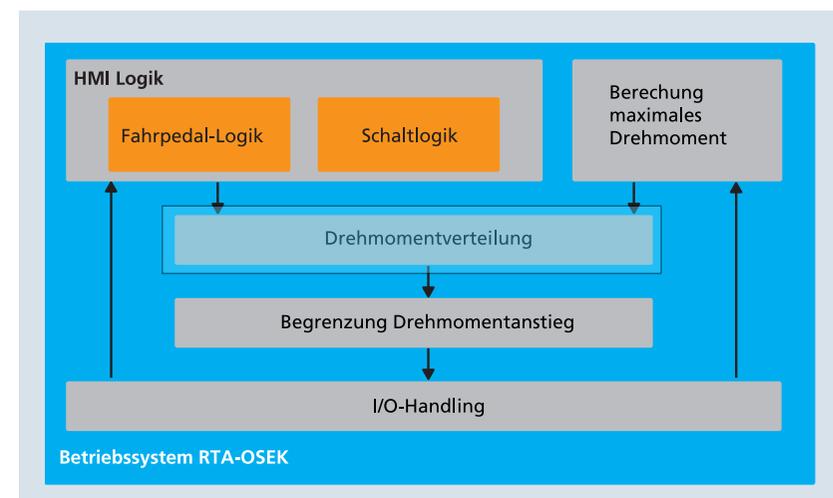
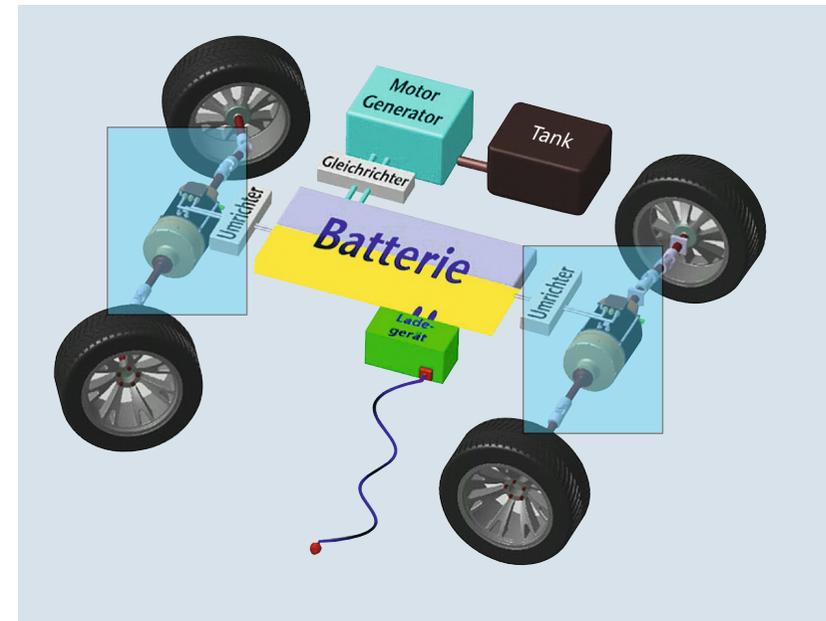
# Entwicklung einer Antriebssteuerung für ein Hybridfahrzeug in einer Rapid Prototyping-Umgebung



1. Einleitung
2. Das Fahrzeug
3. Die Bordnetz-Architektur
- 4. Funktionsentwicklung mit ASCET**
  1. Fahrpedal-Logik
  2. Berechnung des maximalen Drehmoments
  - 3. Drehmomentverteilung**
  4. Drehmomentanstiegsbegrenzung
  5. Funktionsmodellierung
5. Die Erprobung
  1. Offline-Simulation
  2. Rapid Prototyping
6. Höhere Antriebsfunktion
7. Fazit
8. Abkürzungsverzeichnis

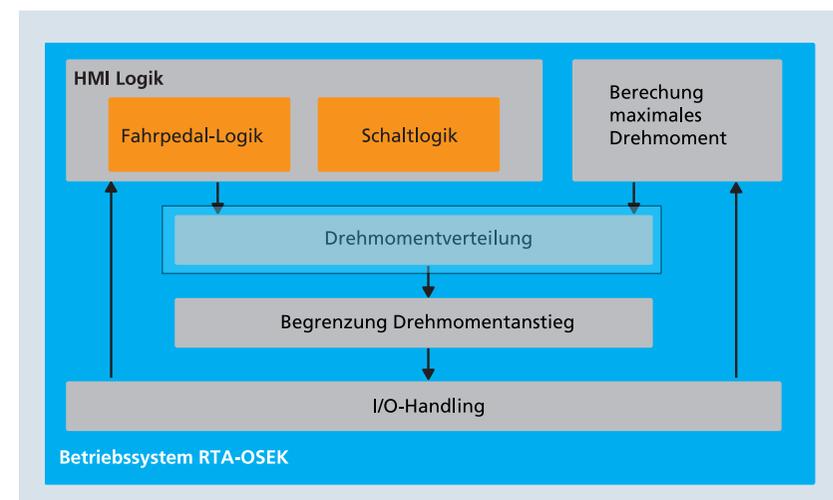
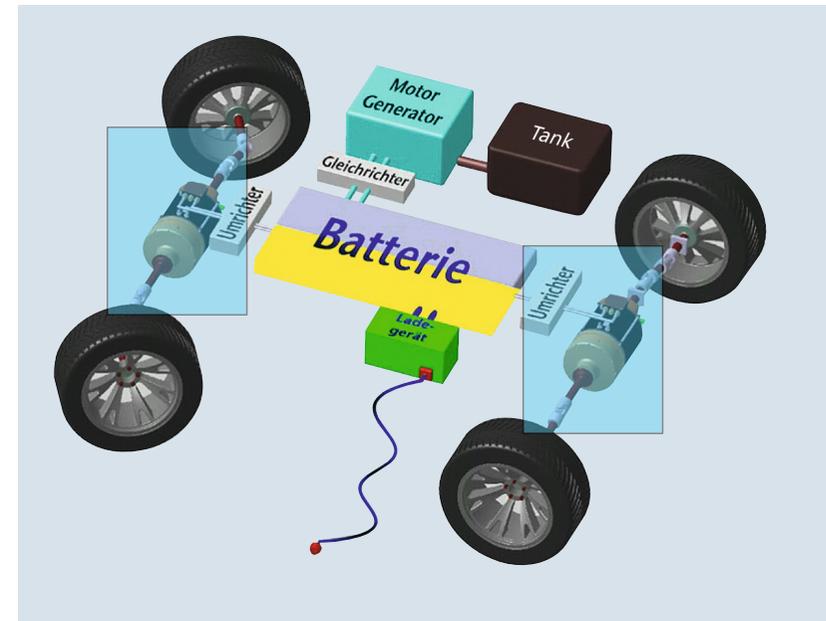
## Drehmomentverteilung (1)

- Nachdem das Soll-Drehmoment ermittelt worden ist, muss dieses auf die beiden Achsen verteilt werden. Die Verteilung erfolgt dabei variabel nach folgender Strategie:
- Zuerst wird das Drehmoment in Abhängigkeit von der Geschwindigkeit **auf die beiden Achsen verteilt**. Dies geschieht unter Verwendung einer Kennlinie mit mehreren Stützstellen und linearer Interpolation.



## Drehmomentverteilung (1)

- Die Soll-Drehmomentvorgaben werden anschließend für die Vorder- und die Hinterachse mit den ermittelten Grenzwerten verglichen. **Strategie ist es, das vom Fahrer angeforderte Gesamtdrehmoment soweit wie möglich aufzubauen.** Sollte eine Drehmomentvorgabe an einer Achse den Grenzwert überschreiten, so wird das Drehmoment an dieser Achse begrenzt. Es wird jedoch versucht dies auszugleichen, indem das Drehmoment an der anderen Achse soweit wie möglich (im Rahmen der Grenze) heraufgesetzt wird, um dem Fahrerwunsch nachzukommen. **Die ideale Achsverteilung wird also zu Gunsten des Drehmomentwunsches aufgegeben.**



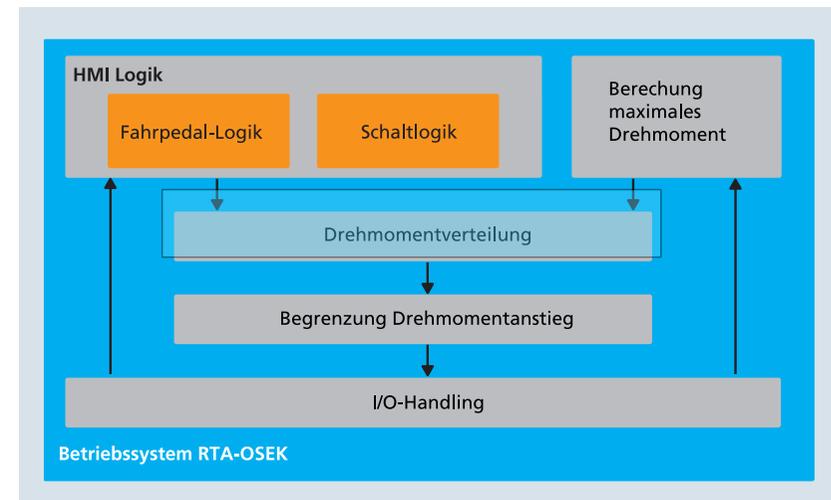
# Entwicklung einer Antriebssteuerung für ein Hybridfahrzeug in einer Rapid Prototyping-Umgebung



1. Einleitung
2. Das Fahrzeug
3. Die Bordnetz-Architektur
- 4. Funktionsentwicklung mit ASCET**
  1. Fahrpedal-Logik
  2. Berechnung des maximalen Drehmoments
  3. Drehmomentverteilung
  - 4. Drehmomentanstiegsbegrenzung**
  5. Funktionsmodellierung
5. Die Erprobung
  1. Offline-Simulation
  2. Rapid Prototyping
6. Höhere Antriebsfunktion
7. Fazit
8. Abkürzungsverzeichnis

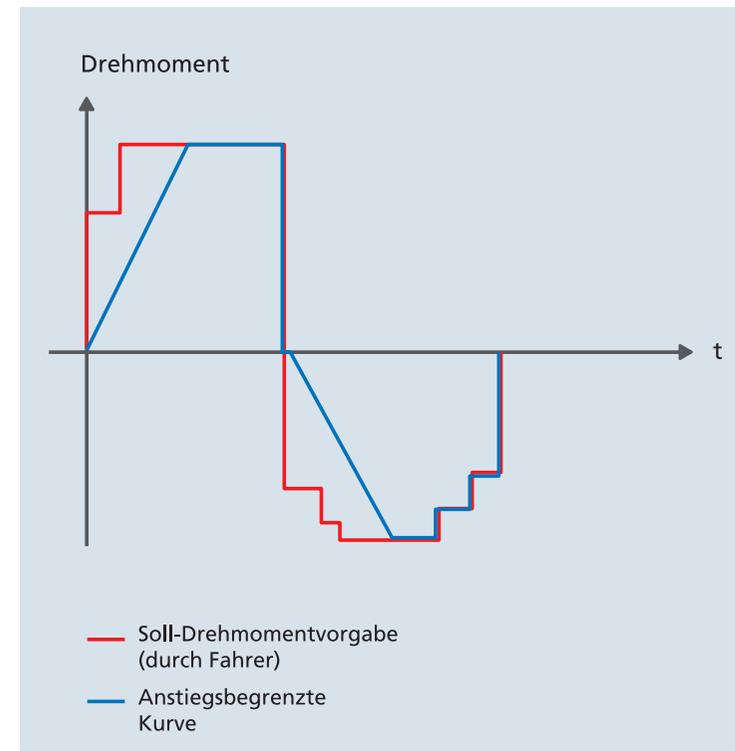
## Drehmomentanstiegsbegrenzung (1)

- Elektromotoren können innerhalb sehr kurzer Zeit ein sehr hohes Drehmoment aufbringen. Um eine Beschädigung des Antriebsstrangs zu vermeiden, wird die auf die Antriebswellen wirkende Drehmomentänderung (1. Ableitung des Drehmoments nach der Zeit) betragsmäßig auf  $MAX\_TRQ\_CH$  [Nm/s] begrenzt. Dazu wird das durch die Elektromotoren zu erzeugende Drehmoment mit einer Rampenfunktion aufgebaut. Ein sofortiges Abfallen des Drehmoments auf einen betragsmäßig niedrigeren Wert mit gleichem Vorzeichen bzw. auf 0 Nm kann zu keiner Beschädigung führen und ist deshalb möglich.



## Drehmomentanstiegsbegrenzung (2)

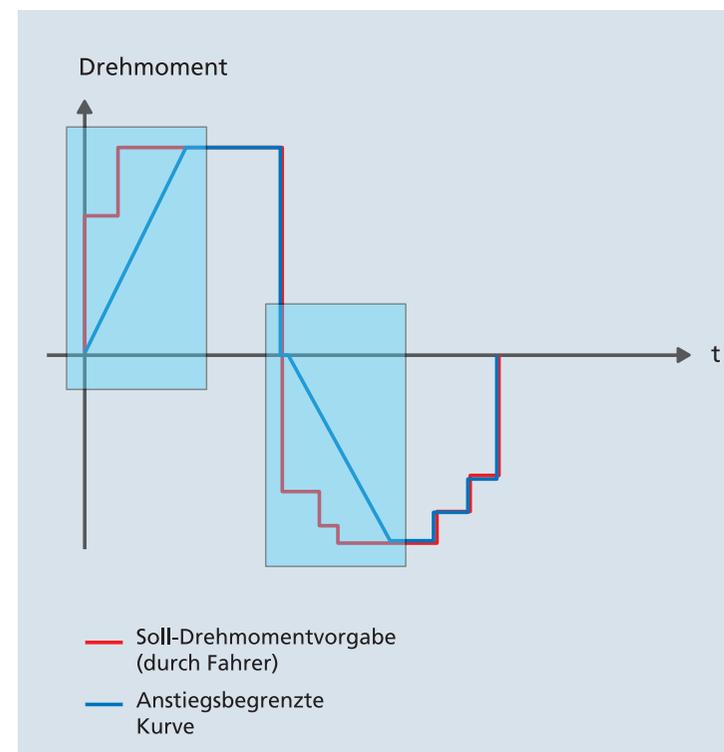
- Elektromotoren können innerhalb sehr kurzer Zeit ein sehr hohes Drehmoment aufbringen. Um eine Beschädigung des Antriebsstrangs zu vermeiden, wird die auf die Antriebswellen wirkende Drehmomentänderung (1. Ableitung des Drehmoments nach der Zeit) betragsmäßig auf  $MAX\_TRQ\_CH$  [Nm/s] begrenzt. Dazu wird das durch die Elektromotoren zu erzeugende Drehmoment mit einer Rampenfunktion aufgebaut. Ein sofortiges Abfallen des Drehmoments auf einen betragsmäßig niedrigeren Wert mit gleichem Vorzeichen bzw. auf 0 Nm kann zu keiner Beschädigung führen und ist deshalb möglich.



## Drehmomentanstiegsbegrenzung

## Drehmomentanstiegsbegrenzung (2)

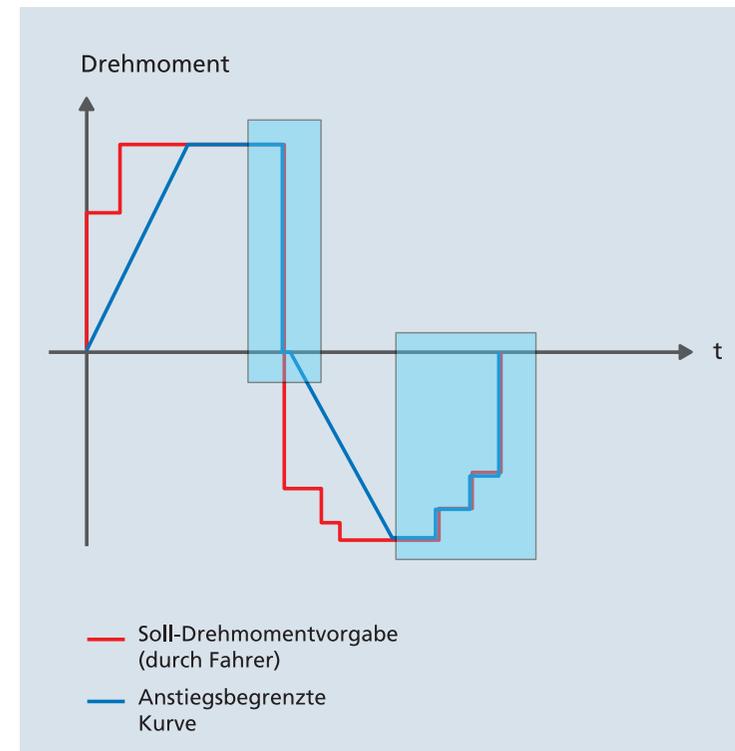
- Elektromotoren können innerhalb sehr kurzer Zeit ein sehr hohes Drehmoment aufbringen. Um eine Beschädigung des Antriebsstrangs zu vermeiden, wird die auf die Antriebswellen wirkende Drehmomentänderung (1. Ableitung des Drehmoments nach der Zeit) betragsmäßig auf  $\text{MAX\_TRQ\_CH}$  [Nm/s] begrenzt. Dazu wird das durch die Elektromotoren zu erzeugende Drehmoment mit einer **Rampenfunktion aufgebaut**. Ein sofortiges Abfallen des Drehmoments auf einen betragsmäßig niedrigeren Wert mit gleichem Vorzeichen bzw. auf 0 Nm kann zu keiner Beschädigung führen und ist deshalb möglich.



## Drehmomentanstiegsbegrenzung

## Drehmomentanstiegsbegrenzung (2)

- Elektromotoren können innerhalb sehr kurzer Zeit ein sehr hohes Drehmoment aufbringen. Um eine Beschädigung des Antriebsstrangs zu vermeiden, wird die auf die Antriebswellen wirkende Drehmomentänderung (1. Ableitung des Drehmoments nach der Zeit) betragsmäßig auf  $MAX\_TRQ\_CH$  [Nm/s] begrenzt. Dazu wird das durch die Elektromotoren zu erzeugende Drehmoment mit einer Rampenfunktion aufgebaut. Ein **sofortiges Abfallen des Drehmoments auf einen betragsmäßig niedrigeren Wert mit gleichem Vorzeichen bzw. auf 0 Nm** kann zu keiner Beschädigung führen und ist deshalb möglich.



## Drehmomentanstiegsbegrenzung

# Entwicklung einer Antriebssteuerung für ein Hybridfahrzeug in einer Rapid Prototyping-Umgebung



1. Einleitung
2. Das Fahrzeug
3. Die Bordnetz-Architektur
- 4. Funktionsentwicklung mit ASCET**
  1. Fahrpedal-Logik
  2. Berechnung des maximalen Drehmoments
  3. Drehmomentverteilung
  4. Drehmomentanstiegsbegrenzung
- 5. Funktionsmodellierung**
5. Die Erprobung
  1. Offline-Simulation
  2. Rapid Prototyping
6. Höhere Antriebsfunktion
7. Fazit
8. Abkürzungsverzeichnis

- Alle Antriebsfunktionen wurden mit dem Werkzeug ASCET entwickelt. Hierbei kamen Blockdiagramme, Zustandsautomaten und auch bedingte Tabellen zum Einsatz.
- [http://www.etas.com/de/products/ascet\\_software\\_products.php](http://www.etas.com/de/products/ascet_software_products.php)
- Die ASCET Produktfamilie ermöglicht die modellbasierte Entwicklung von Anwendungssoftware (engl. Application SW) und die automatische Codegenerierung aus diesen Modellen. ASCET wurde speziell entwickelt, um den spezifischen Automotive-Anforderungen an komplexe Embedded Software mit Echtzeit-, Effizienz- und Sicherheitsrestriktionen gerecht zu werden.
- Vom ersten Komponentenentwurf mittels Blockdiagrammen und Zustandsautomaten bis zur automatischen Generierung des Codes für das Mikrocontrollersystem fügen sich die ASCET-Komponenten nahtlos in die Prozesse ein.
- Seit 1997 wird ASCET erfolgreich eingesetzt, insbesondere auch für die Entwicklung sicherheitskritischer Software rund um Brems- und Lenksysteme sowie bei Motormanagementsystemen. Mit ASCET entwickelte Software ist seit dem in über 68 Millionen Steuergeräten zu finden. Diese langjährige Erfahrung garantiert eine effiziente und sichere Softwareentwicklung mit gleichbleibend hoher Qualität und ermöglicht es, Kosten-, Qualitäts- und Time-to-Market-Ziele sicher zu erreichen.

- Alle Antriebsfunktionen wurden mit dem Werkzeug ASCET entwickelt. Hierbei kamen **Blockdiagramme**, **Zustandsautomaten** und auch bedingte Tabellen zum Einsatz.
- [http://www.etas.com/de/products/ascet\\_software\\_products.php](http://www.etas.com/de/products/ascet_software_products.php)
- Die ASCET Produktfamilie ermöglicht die modellbasierte Entwicklung von Anwendungssoftware (engl. Application SW) und die automatische Codegenerierung aus diesen Modellen. ASCET wurde speziell entwickelt, um den spezifischen Automotive-Anforderungen an komplexe Embedded Software mit Echtzeit-, Effizienz- und Sicherheitsrestriktionen gerecht zu werden.
- Vom ersten Komponentenentwurf mittels **Blockdiagrammen** und **Zustandsautomaten** bis zur automatischen Generierung des Codes für das Mikrocontrollersystem fügen sich die ASCET-Komponenten nahtlos in die Prozesse ein.
- Seit 1997 wird ASCET erfolgreich eingesetzt, insbesondere auch für die Entwicklung sicherheitskritischer Software rund um **Brems- und Lenksysteme** sowie bei **Motormanagementsystemen**. Mit ASCET entwickelte Software ist seit dem in über 68 Millionen Steuergeräten zu finden. Diese langjährige Erfahrung garantiert eine effiziente und sichere Softwareentwicklung mit gleichbleibend hoher Qualität und ermöglicht es, Kosten-, Qualitäts- und Time-to-Market-Ziele sicher zu erreichen.

## Funktionsmodellierung (2)



### Funktionen auf einen Blick

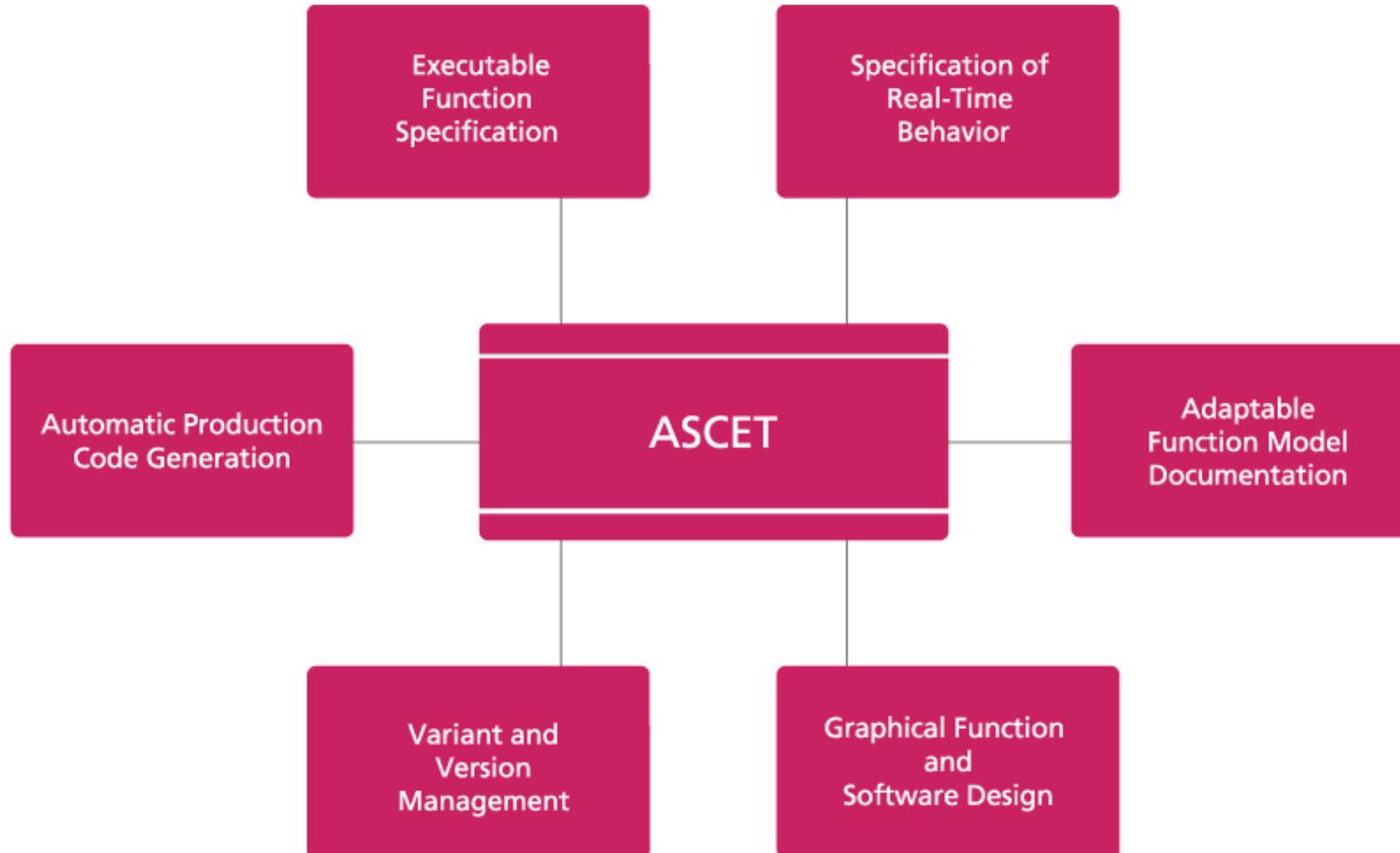
- Modellbasierte Entwicklung von Automotive Software mit Echtzeit-Anforderungen
- Automatische und hoch effiziente Generierung des Steuergeräte-Seriencodes
- Migration und Entwicklung von AUTOSAR Softwarekomponenten
- Einfache Integration in bestehende Entwicklungsumgebungen
- Einfache Integration von bestehendem C-Code und Library-Funktionen
- MISRA-konformer und zertifizierbarer Codegenerator (IEC61508 bzw. ISO26262)
- Flexible, anpassbare Dokumentation der Funktions- und Softwaremodelle
- Import von Simulink®- und UML-Modellen
- Einfache Anbindung von Software-Konfigurationsmanagement-Systemen

## Funktionsmodellierung (2)



### Funktionen auf einen Blick

- **Modellbasierte Entwicklung** von Automotive Software mit **Echtzeit-Anforderungen**
- Automatische und hoch effiziente **Generierung des Steuergeräte-Seriencodes**
- Migration und Entwicklung von **AUTOSAR** Softwarekomponenten
- Einfache Integration in bestehende Entwicklungsumgebungen
- Einfache Integration von bestehendem C-Code und Library-Funktionen
- MISRA-konformer und **zertifizierbarer Codegenerator (IEC61508 bzw. ISO26262)**
- Flexible, anpassbare Dokumentation der Funktions- und Softwaremodelle
- Import von **Simulink®**- und **UML-Modellen**
- Einfache Anbindung von Software-**Konfigurationsmanagement**-Systemen





# Entwicklung einer Antriebssteuerung für ein Hybridfahrzeug in einer Rapid Prototyping-Umgebung



1. Einleitung
2. Das Fahrzeug
3. Die Bordnetz-Architektur
4. Funktionsentwicklung mit ASCET
  1. Fahrpedal-Logik
  2. Berechnung des maximalen Drehmoments
  3. Drehmomentverteilung
  4. Drehmomentanstiegsbegrenzung
  5. Funktionsmodellierung
- 5. Die Erprobung**
  1. Offline-Simulation
  2. Rapid Prototyping
6. Höhere Antriebsfunktion
7. Fazit
8. Abkürzungsverzeichnis

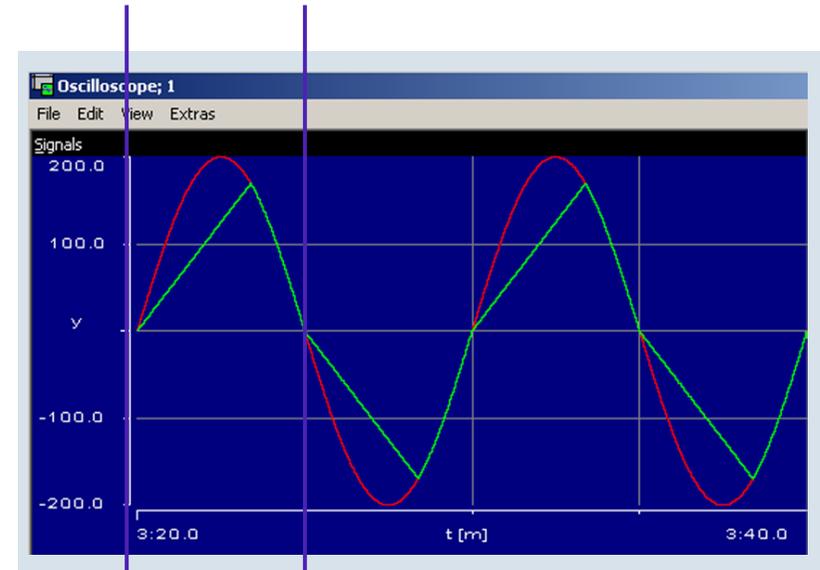
# Entwicklung einer Antriebssteuerung für ein Hybridfahrzeug in einer Rapid Prototyping-Umgebung



1. Einleitung
2. Das Fahrzeug
3. Die Bordnetz-Architektur
4. Funktionsentwicklung mit ASCET
  1. Fahrpedal-Logik
  2. Berechnung des maximalen Drehmoments
  3. Drehmomentverteilung
  4. Drehmomentanstiegsbegrenzung
  5. Funktionsmodellierung
- 5. Die Erprobung**
  - 1. Offline-Simulation**
  2. Rapid Prototyping
6. Höhere Antriebsfunktion
7. Fazit
8. Abkürzungsverzeichnis

## Offline-Simulation (1)

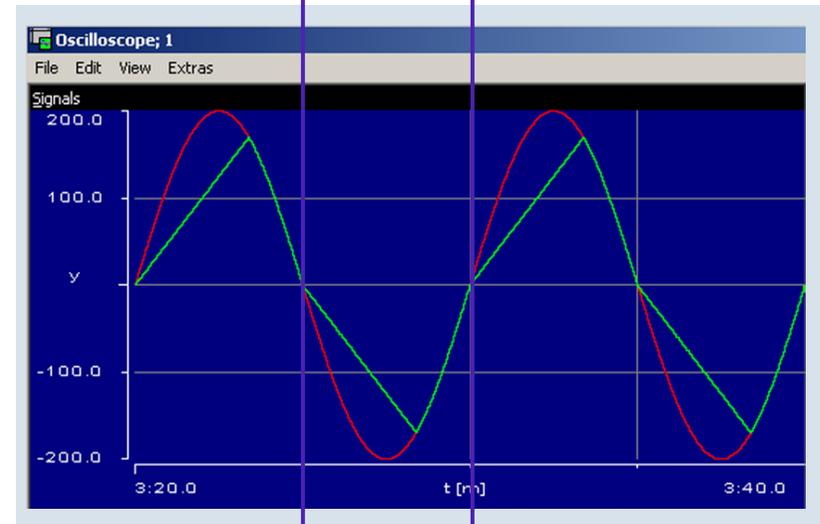
- Im ersten Schritt wurde das Verhalten der Softwarekomponenten durch Offline-Simulation erprobt. Die Abbildung zeigt ein Beispiel eines solchen Experiments, in dem die Drehmomentanstiegsbegrenzung getestet wird. In diesem sind der zeitliche Verlauf der Eingangsgröße (rote Kurve) und der Ausgangsgröße (grüne Kurve) dargestellt. Die Eingangsgröße wird hierbei durch eine Sinus-Funktion stimuliert. Wie man erkennen kann, baut sich das Drehmoment am Ausgang nur mit einer gewissen Steigung auf. In dem Moment, wo sich die grüne und die rote Kurve schneiden, nehmen beide Kurven (und damit die Eingangs- und die Ausgangsgröße) gemeinsam betragsmäßig ab.



Offline-Simulation der Drehmomentanstiegsbegrenzung

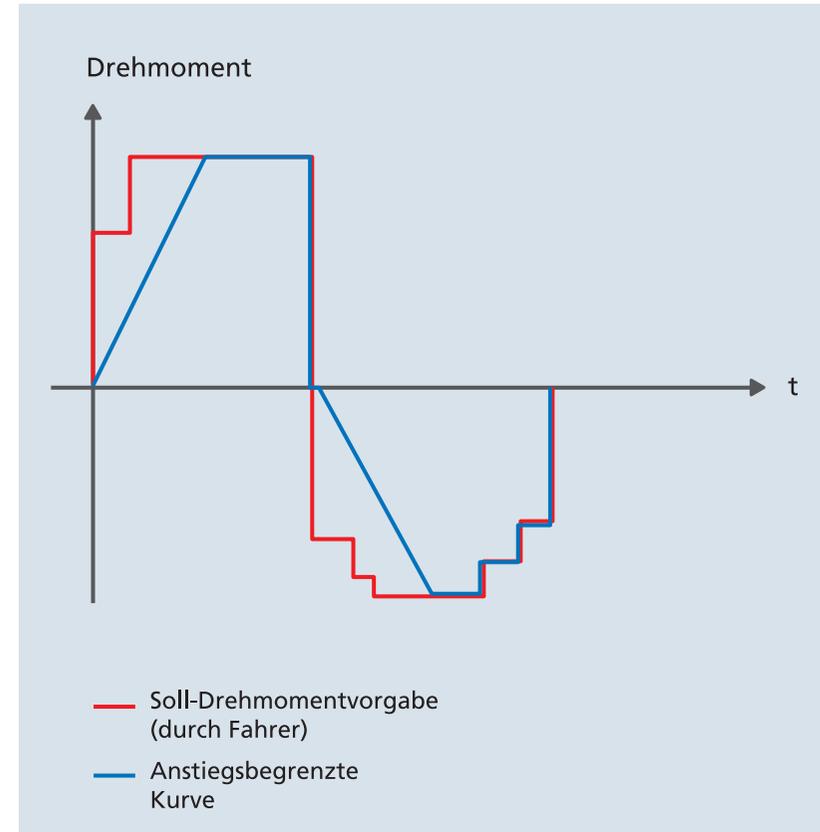
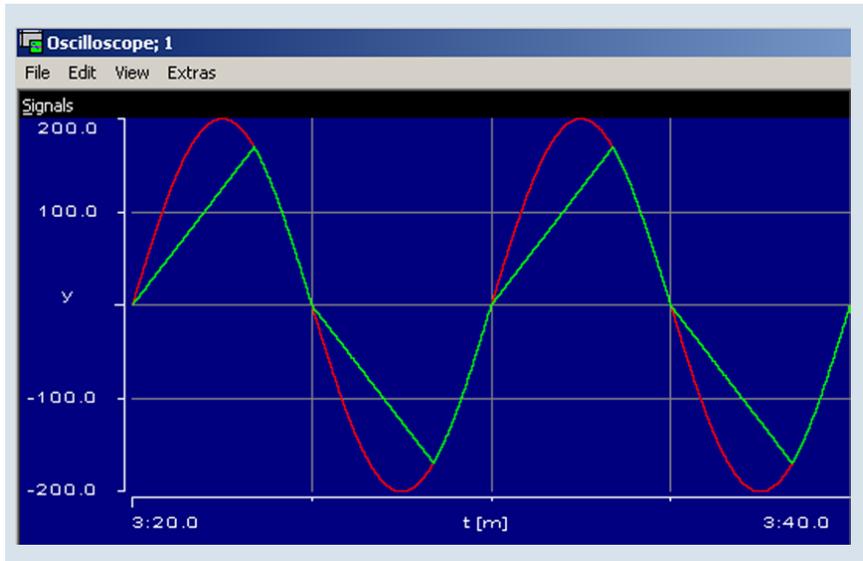
## Offline-Simulation (2)

- Ist der Nullpunkt durchbrochen, baut sich das Drehmoment am Ausgang langsam mit der vorgegebenen Steigung in die andere Richtung auf. Somit kann man sich vom korrekten Verhalten der Klasse überzeugen.

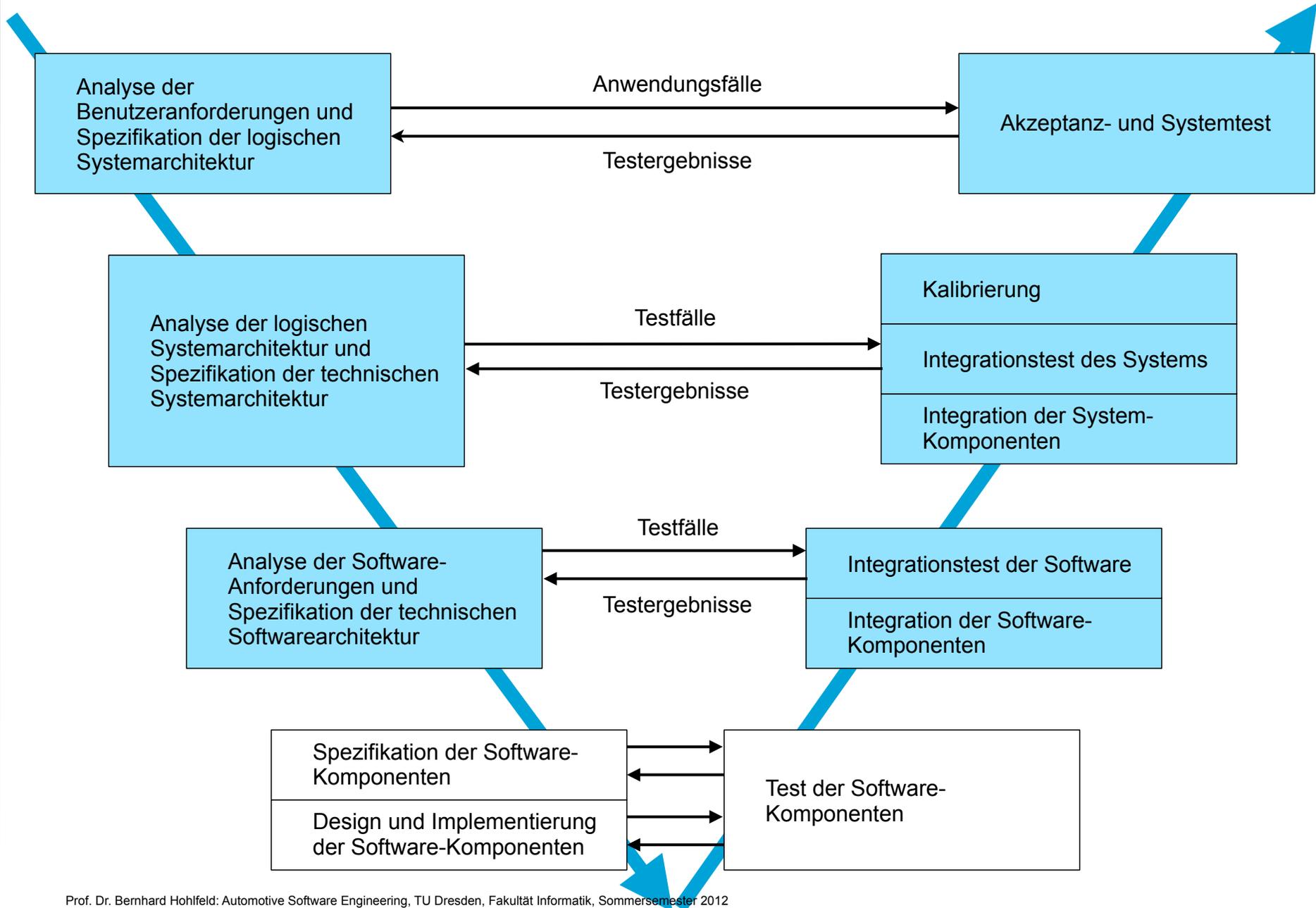


Offline-Simulation der Drehmomentanstiegsbegrenzung

## Offline-Simulation (3)



# Kernprozess zur Entwicklung von elektronischen Systemen und Software (Nach Schäuuffele, Zurawka) siehe Teil 6 der Vorlesung



# Begriffsdefinitionen



- MIL Model in the Loop, SIL Software in the Loop
  - Modell / SG-Software läuft auf simuliertem SG, das von simulierter Fahrzeugumgebung (rechnergenerierten Sensor- und Bussignalen) gespeist wird.
- HIL Hardware in the Loop
  - Reale SG-Hardware wird von simulierter Fahrzeugumgebung gespeist
- Prototyp
  - Simuliertes Steuergerät im Fahrzeug ("PC im Kofferraum")
- Prüfstand, Fahrversuch
  - Steuergerät im Fahrzeug unter Laborbedingungen bzw. Auf der Strasse (Testgelände, öffentliches Strassennetz)



	Steuergerät	
Umgebung, Fahrzeug	simuliert	real
simuliert	MIL, SIL	HIL
real	Prototyp	Prüfstand, Fahrversuch

# Begriffsdefinitionen



- MIL Model in the Loop, SIL Software in the Loop
  - Modell / SG-Software läuft auf simuliertem SG, das von simulierter Fahrzeugumgebung (rechnergenerierten Sensor- und Bussignalen) gespeist wird.
- HIL Hardware in the Loop
  - Reale SG-Hardware wird von simulierter Fahrzeugumgebung gespeist
- Prototyp
  - Simuliertes Steuergerät im Fahrzeug ("PC im Kofferraum")
- Prüfstand, Fahrversuch
  - Steuergerät im Fahrzeug unter Laborbedingungen bzw. Auf der Strasse (Testgelände, öffentliches Strassennetz)



	Steuergerät	
Umgebung, Fahrzeug	simuliert	real
simuliert	MIL, SIL	HIL
real	Prototyp	Prüfstand, Fahrversuch

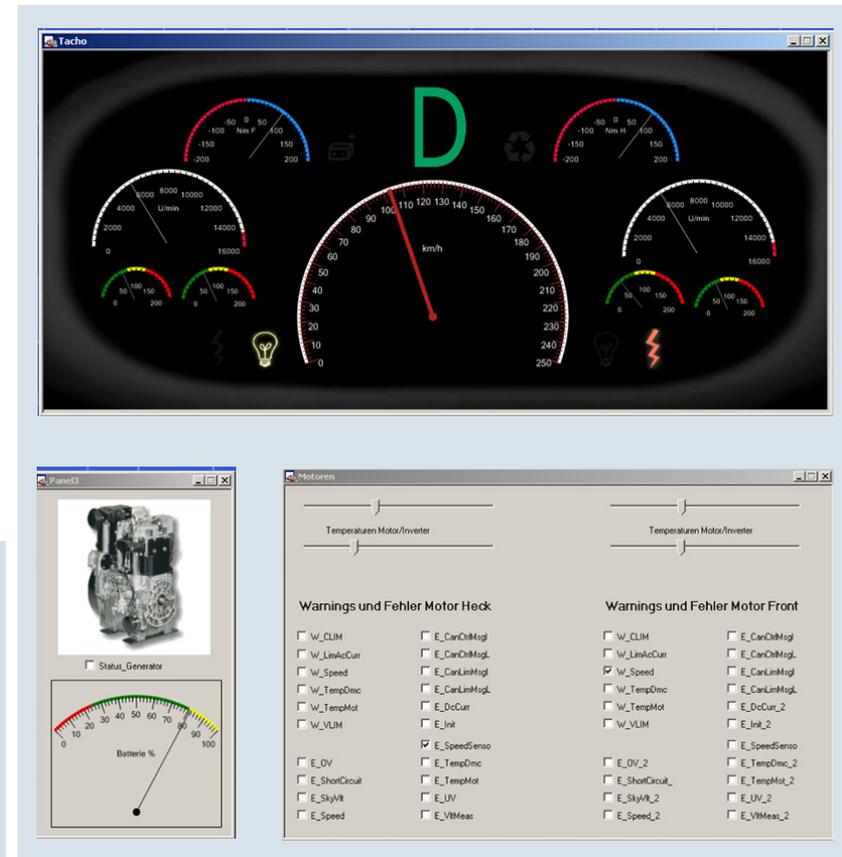
# Entwicklung einer Antriebssteuerung für ein Hybridfahrzeug in einer Rapid Prototyping-Umgebung



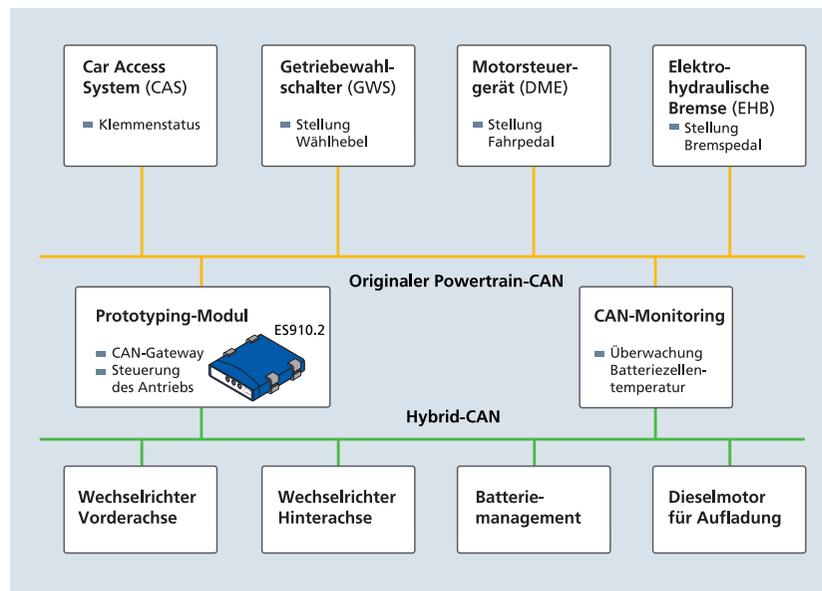
1. Einleitung
2. Das Fahrzeug
3. Die Bordnetz-Architektur
4. Funktionsentwicklung mit ASCET
  1. Fahrpedal-Logik
  2. Berechnung des maximalen Drehmoments
  3. Drehmomentverteilung
  4. Drehmomentanstiegsbegrenzung
  5. Funktionsmodellierung
- 5. Die Erprobung**
  1. Offline-Simulation
  - 2. Rapid Prototyping**
6. Höhere Antriebsfunktion
7. Fazit
8. Abkürzungsverzeichnis

# Rapid Prototyping (1)

- Im nächsten Schritt wurde das Funktionsmodell mit Hilfe des Werkzeugs INTECRIO auf das Rapid Prototyping-Modul ES910 gebracht. Bevor das Modul im realen Fahrzeug getestet wurde, erfolgte ein weiterer Absicherungsschritt der Gesamtfunktionalität gegenüber einer Restbussimulation (siehe Abbildung).



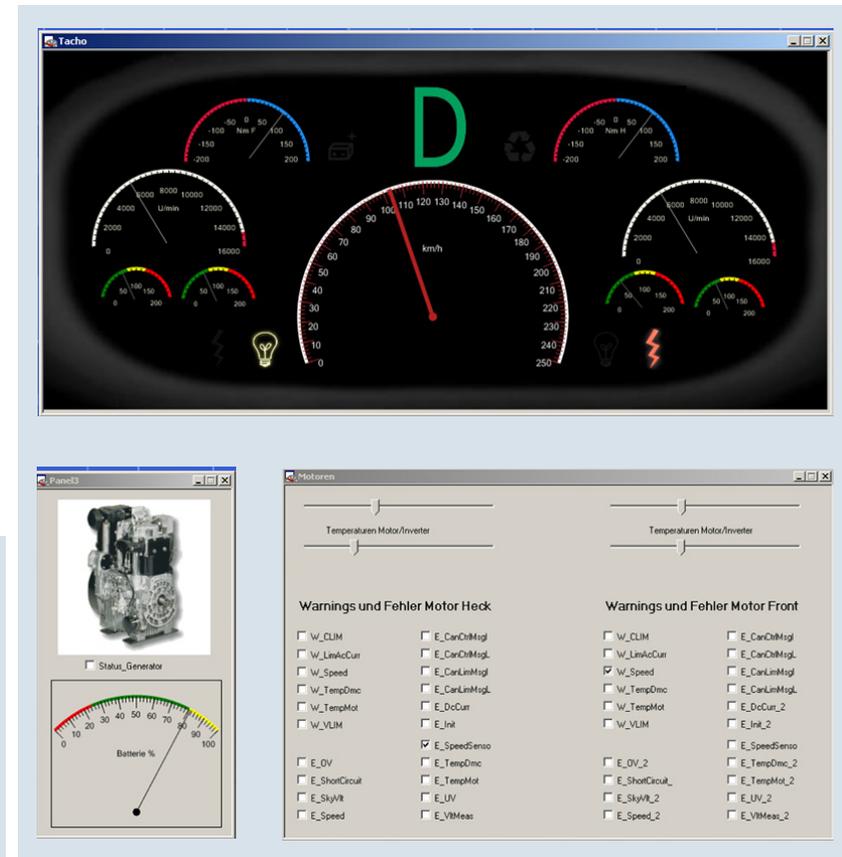
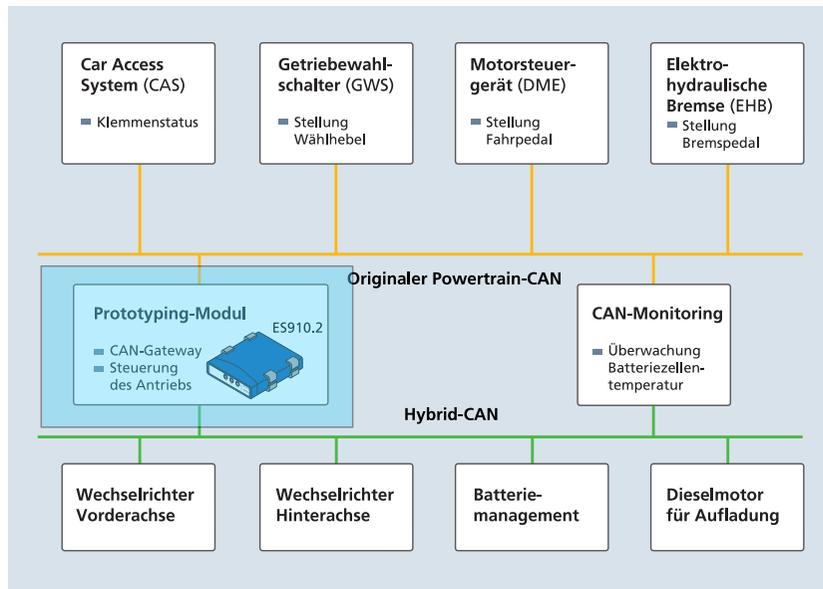
Panels der Restbussimulation



# Rapid Prototyping (1)



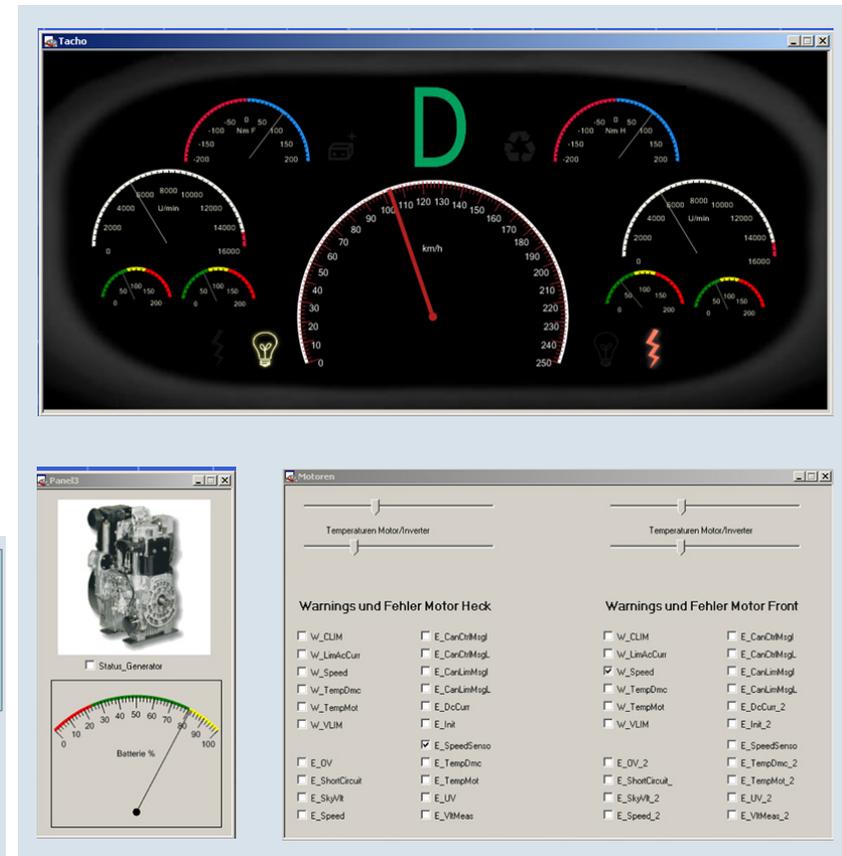
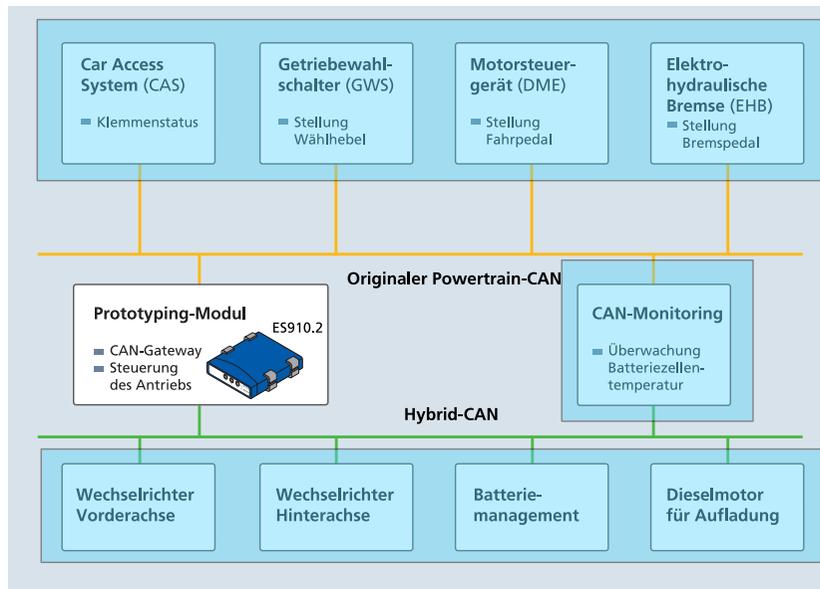
- Im nächsten Schritt wurde das Funktionsmodell mit Hilfe des Werkzeugs INTECRIO auf das **Rapid Prototyping-Modul ES910** gebracht. Bevor das Modul im realen Fahrzeug getestet wurde, erfolgte ein weiterer Absicherungsschritt der **Gesamtfunktionalität** gegenüber einer Restbussimulation (siehe Abbildung).



Panels der Restbussimulation

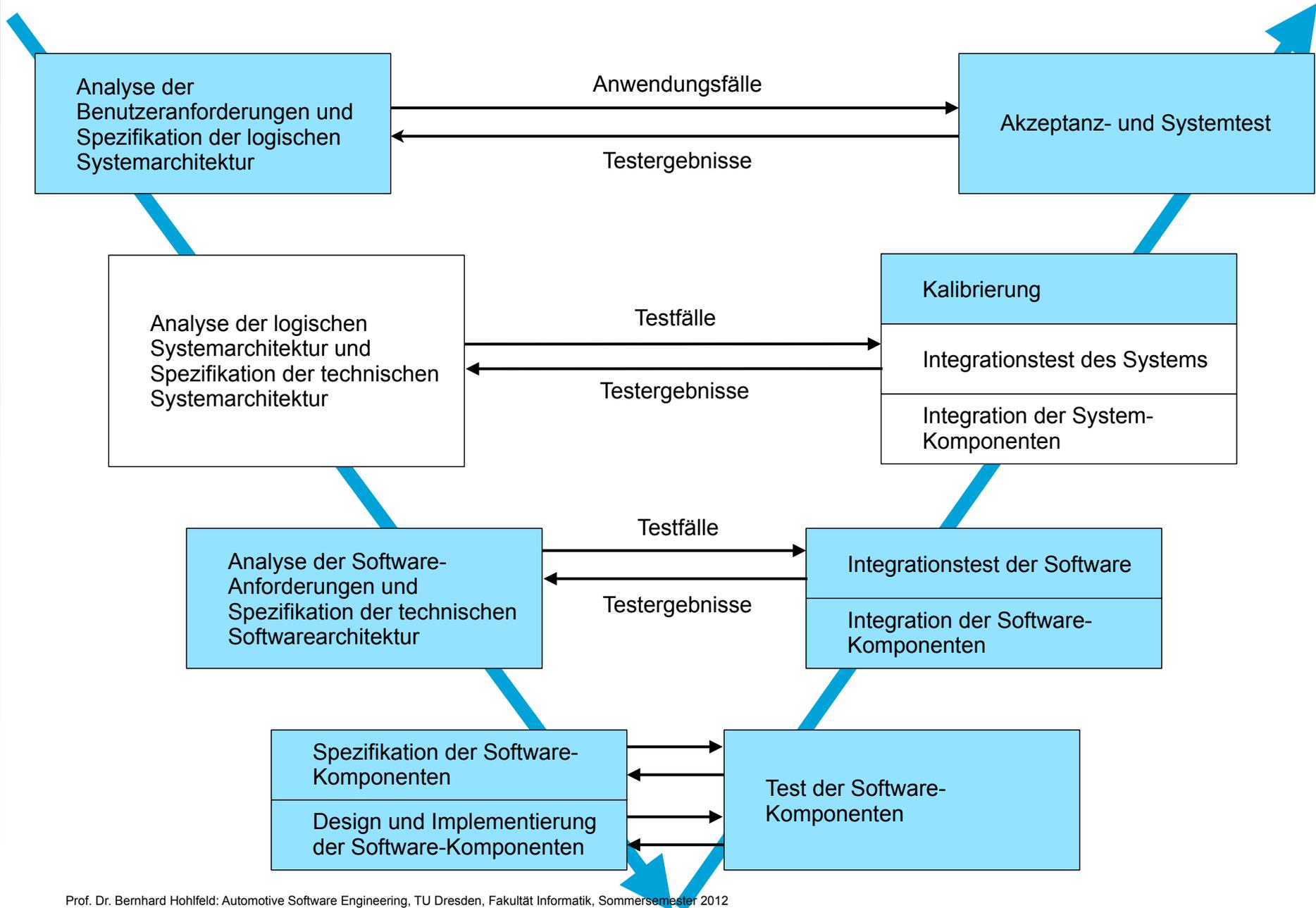
# Rapid Prototyping (1)

- Im nächsten Schritt wurde das Funktionsmodell mit Hilfe des Werkzeugs INTECRIO auf das Rapid Prototyping-Modul ES910 gebracht. Bevor das Modul im realen Fahrzeug getestet wurde, erfolgte ein weiterer Absicherungsschritt der Gesamtfunktionalität gegenüber einer **Restbussimulation** (siehe Abbildung).



Panels der Restbussimulation

# Kernprozess zur Entwicklung von elektronischen Systemen und Software (Nach Schäuuffele, Zurawka) siehe Teil 6 der Vorlesung



# Begriffsdefinitionen



- MIL Model in the Loop, SIL Software in the Loop
  - Modell / SG-Software läuft auf simuliertem SG, das von simulierter Fahrzeugumgebung (rechnergenerierten Sensor- und Bussignalen) gespeist wird.
- HIL Hardware in the Loop
  - Reale SG-Hardware wird von simulierter Fahrzeugumgebung gespeist
- Prototyp
  - Simuliertes Steuergerät im Fahrzeug ("PC im Kofferraum")
- Prüfstand, Fahrversuch
  - Steuergerät im Fahrzeug unter Laborbedingungen bzw. Auf der Strasse (Testgelände, öffentliches Strassennetz)



	Steuergerät	
Umgebung, Fahrzeug	simuliert	real
simuliert	MIL, SIL	HIL
real	Prototyp	Prüfstand, Fahrversuch

# Begriffsdefinitionen



- MIL Model in the Loop, SIL Software in the Loop
  - Modell / SG-Software läuft auf simuliertem SG, das von simulierter Fahrzeugumgebung (rechnergenerierten Sensor- und Bussignalen) gespeist wird.
- HIL Hardware in the Loop
  - Reale SG-Hardware wird von simulierter Fahrzeugumgebung gespeist
- Prototyp
  - Simuliertes Steuergerät im Fahrzeug ("PC im Kofferraum")
- Prüfstand, Fahrversuch
  - Steuergerät im Fahrzeug unter Laborbedingungen bzw. Auf der Strasse (Testgelände, öffentliches Strassennetz)



	Steuergerät	
Umgebung, Fahrzeug	simuliert	real
simuliert	MIL, SIL	<b>HIL</b>
real	Prototyp	Prüfstand, Fahrversuch

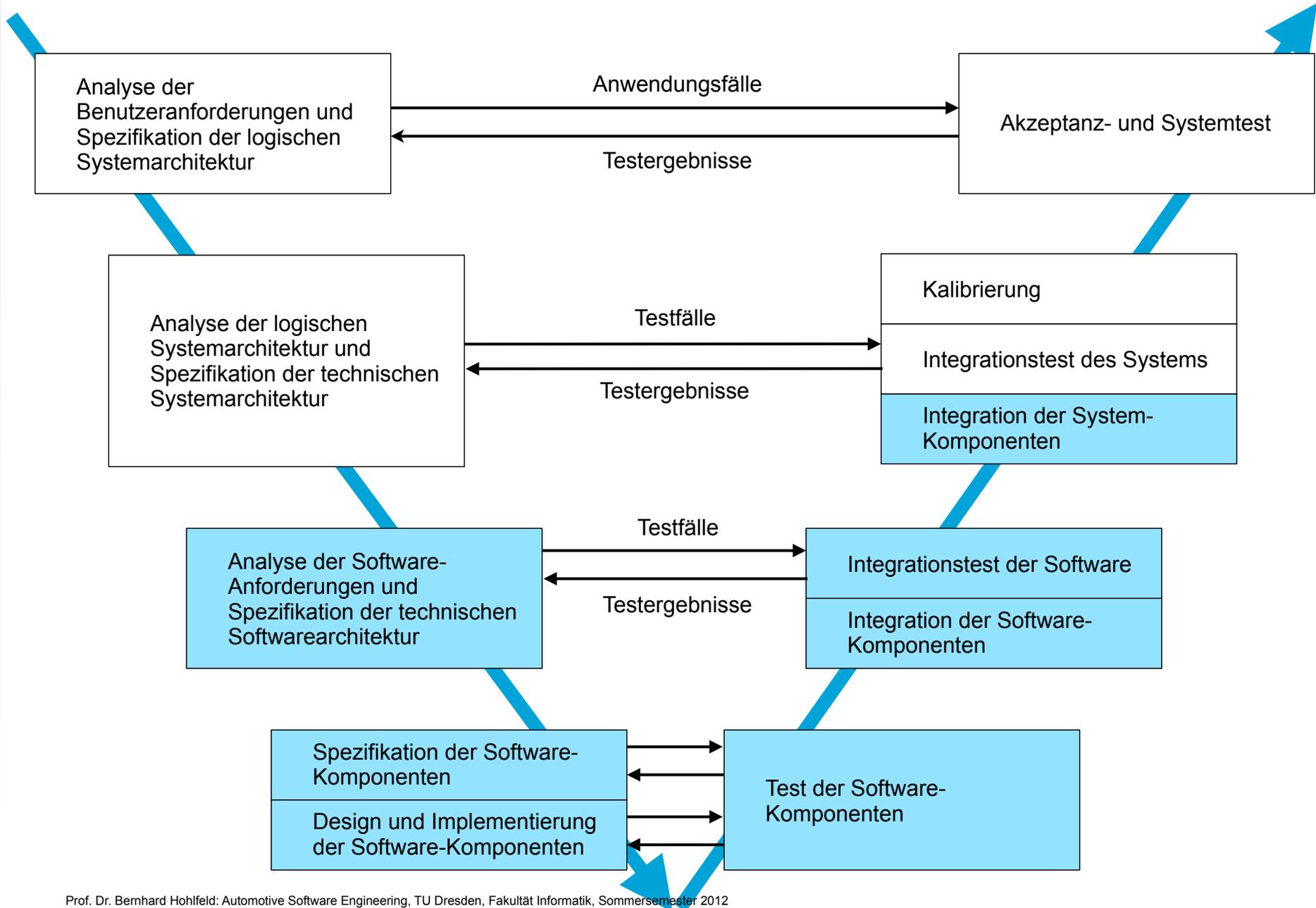
## Rapid Prototyping (2)



- Derzeit erfolgt die Erprobung des neuen Antriebssystems sowohl auf einem **Rollenprüfstand** als auch auf der **Straße**. **Ziel der Erprobung ist es, durch Veränderung vielfältiger Parametersätze das Fahrverhalten zu optimieren.** Im Zentrum steht dabei die Drehmomentverteilung auf die beiden Achsen des Fahrzeugs unter Berücksichtigung der vorhandenen Energie, der Temperatur aller Komponenten und der Fahrzeuggeschwindigkeit. Ferner wird mit unterschiedlichen Fahrpedalkennlinien experimentiert. Bei den ersten Versuchen hat sich gezeigt, dass man sich bei entsprechender Kalibrierung sehr schnell an das kombinierte Fahr-/Bremspedal gewöhnen kann.



# Kernprozess zur Entwicklung von elektronischen Systemen und Software (Nach Schäuuffele, Zurawka) siehe Teil 6 der Vorlesung



# Begriffsdefinitionen



- MIL Model in the Loop, SIL Software in the Loop
  - Modell / SG-Software läuft auf simuliertem SG, das von simulierter Fahrzeugumgebung (rechnergenerierten Sensor- und Bussignalen) gespeist wird.
- HIL Hardware in the Loop
  - Reale SG-Hardware wird von simulierter Fahrzeugumgebung gespeist
- Prototyp
  - Simuliertes Steuergerät im Fahrzeug ("PC im Kofferraum")
- Prüfstand, Fahrversuch
  - Steuergerät im Fahrzeug unter Laborbedingungen bzw. Auf der Strasse (Testgelände, öffentliches Strassennetz)



	Steuergerät	
Umgebung, Fahrzeug	simuliert	real
simuliert	MIL, SIL	HIL
real	Prototyp	Prüfstand, Fahrversuch

# Begriffsdefinitionen



- MIL Model in the Loop, SIL Software in the Loop
  - Modell / SG-Software läuft auf simuliertem SG, das von simulierter Fahrzeugumgebung (rechnergenerierten Sensor- und Bussignalen) gespeist wird.
- HIL Hardware in the Loop
  - Reale SG-Hardware wird von simulierter Fahrzeugumgebung gespeist
- Prototyp
  - Simuliertes Steuergerät im Fahrzeug ("PC im Kofferraum")
- Prüfstand, Fahrversuch
  - Steuergerät im Fahrzeug unter Laborbedingungen bzw. Auf der Strasse (Testgelände, öffentliches Strassennetz)



	Steuergerät	
Umgebung, Fahrzeug	simuliert	real
simuliert	MIL, SIL	HIL
real	Prototyp	<b>Prüfstand, Fahrversuch</b>

## Rapid Prototyping (3)

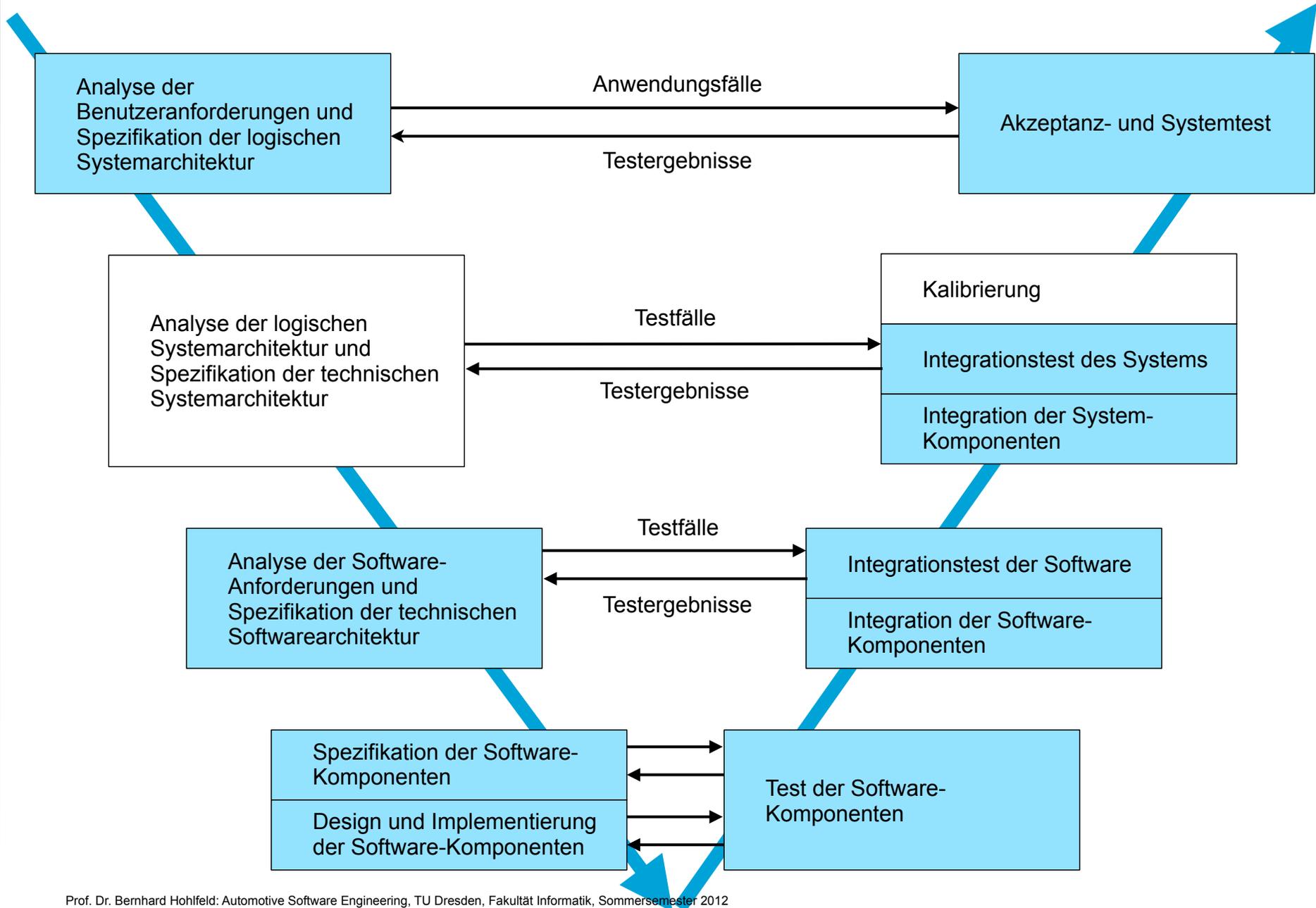


- Derzeit werden die Erprobungen noch direkt über das Werkzeug INTECRIO gemacht (siehe Abbildung). Maschinenbau- und Informatikstudenten arbeiten dabei Hand in Hand. Als nächster Schritt wird nun eine INCA-Oberfläche entwickelt, mit der die Maschinenbauer die Fahrfunktionen selbst kalibrieren können.



INTECRIO-Oberfläche zur Überwachung des Systems

# Kernprozess zur Entwicklung von elektronischen Systemen und Software (Nach Schäuuffele, Zurawka) siehe Teil 6 der Vorlesung



## Rapid Prototyping (4)



<http://www.etas.com/de/products/intecrio.php>

INTECRIO ist ein ETAS-Softwareprodukt, welches das Prototyping elektronischer Systeme im Fahrzeug auf der Basis von MATLAB®/Simulink®, ASCET-Modellen, C-Code-Modulen und AUTOSAR-Softwarekomponenten ermöglicht. Es dient der Funktionsentwicklung als Integrations- und Experimentierplattform.

Für die Verifikation und Validation im Fahrzeug unterstützt INTECRIO die modulare ES1000 VME Hardware-Serie von ETAS, die kompakten Rapid Prototyping Module ES900, die Mikromessmodule ES400 und ETK.

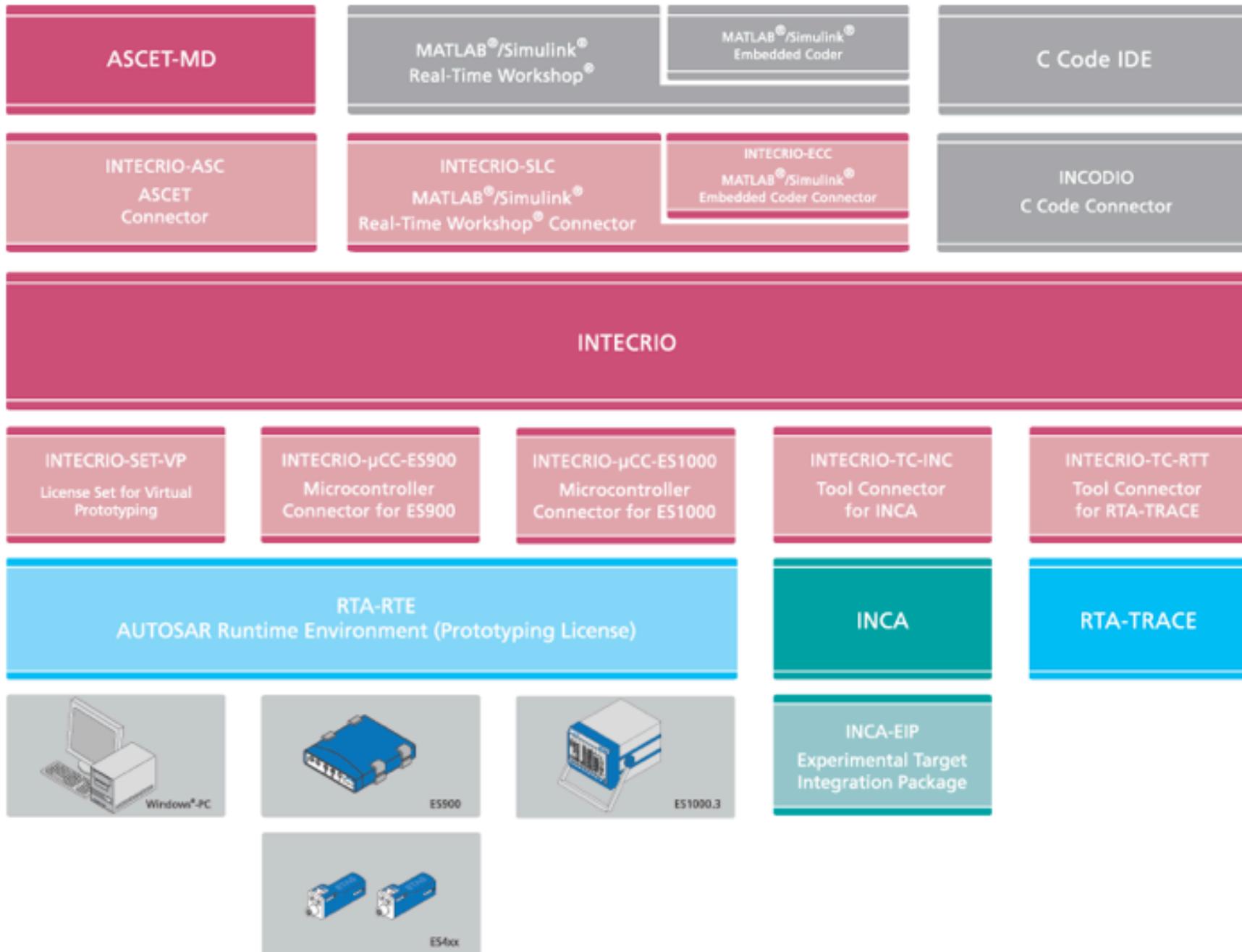
Zur Funktionsvalidation bietet INTECRIO die Möglichkeit zum virtuellen Prototyping auf Windows® PCs direkt am Schreibtisch des Funktionsentwicklers und ermöglicht so die bereichsübergreifende Zusammenarbeit zwischen den Funktionsentwicklern auf der einen Seite und den Systementwicklern und Simulationsexperten auf der anderen Seite. Auf diese Weise bietet das Produkt neue Möglichkeiten für die frühen Fahrzeugentwicklungsphasen, wie zum Beispiel die Vorkalibrierung im Labor.

## Rapid Prototyping (4)



### Funktionen auf einen Blick

- Offene Umgebung für das Rapid Prototyping von elektronischen Systemen im Fahrzeug
- Integration von MATLAB®/Simulink® und ASCET-Modellen und C-Code-Modulen
- Integration von AUTOSAR-Softwarekomponenten durch die Verwendung von AUTOSAR RTE
- Test neuer Funktionen unter virtuellen Bedingungen am PC (virtuelles Prototyping) oder unter realen Bedingungen (Rapid Prototyping)
- Unterstützung verschiedenster Steuergeräte-, Bus-, Sensor- und Aktuatorschnittstellen einschließlich ETK (Universelle Steuergeräte-Schnittstellen von ETAS, Abkürzung nicht erklärt)
- Funktionstest im Fahrzeug mit ETK-Bypass, dem modularen ES1000 Hardwaresystem oder dem **ES910 Rapid Prototyping-Kompaktmodul** in Verbindung mit den ES400 Mikromessmodulen
- Offene Schnittstellen für die Automatisierung und kundenspezifischen Anpassungen
- Intuitive grafische Benutzeroberfläche
- Nahtloses Zusammenspiel mit dem Mess- und Kalibrierwerkzeug INCA und dem Software Logic Analyzer RTA-TRACE



# Entwicklung einer Antriebssteuerung für ein Hybridfahrzeug in einer Rapid Prototyping-Umgebung



1. Einleitung
2. Das Fahrzeug
3. Die Bordnetz-Architektur
4. Funktionsentwicklung mit ASCET
  1. Fahrpedal-Logik
  2. Berechnung des maximalen Drehmoments
  3. Drehmomentverteilung
  4. Drehmomentanstiegsbegrenzung
  5. Funktionsmodellierung
5. Die Erprobung
  1. Offline-Simulation
  2. Rapid Prototyping
- 6. Höhere Antriebsfunktion**
7. Fazit
8. Abkürzungsverzeichnis

## Höhere Antriebsfunktion (1)



- Basierend auf den derzeitigen **Basisfunktionen**, entwickeln jetzt mehrere Studenten höhere Antriebsfunktionen. Dabei werden nicht nur **Komfortfunktionen**, wie z. B. Tempomat oder Halten am Berg entwickelt, sondern auch **Sicherheitsfunktionen**, wie z. B. eine Schlupfregelung für den elektrischen Antrieb. Auch diese Funktionen werden mit ASCET spezifiziert. Ziel ist es, einzelne gekapselte Softwarekomponenten nach **AUTOSAR** zu entwickeln. Im Vordergrund steht auch hier, wie beim gesamten Projekt, die Studenten praxisnah an modernen Entwicklungsmethoden auszubilden.



Überwachung der internen Betriebszustände mit INTECRIO

## Höhere Antriebsfunktion (2)

- Es ist geplant, das Fahrzeug auch in den kommenden Semestern als Entwicklungs- und Ausbildungsplattform zu nutzen. Das Spektrum reicht dabei vom 4-Motorenantrieb über Hochvoltsicherungen bis hin zu Batteriemanagementsystemen sowie Anzeige- und Bedienkonzepten.



Serieller Plug-in-Hybrid auf Basis eines BMW X5

# Entwicklung einer Antriebssteuerung für ein Hybridfahrzeug in einer Rapid Prototyping-Umgebung



1. Einleitung
2. Das Fahrzeug
3. Die Bordnetz-Architektur
4. Funktionsentwicklung mit ASCET
  1. Fahrpedal-Logik
  2. Berechnung des maximalen Drehmoments
  3. Drehmomentverteilung
  4. Drehmomentanstiegsbegrenzung
  5. Funktionsmodellierung
5. Die Erprobung
  1. Offline-Simulation
  2. Rapid Prototyping
6. Höhere Antriebsfunktion
- 7. Fazit**
8. Abkürzungsverzeichnis

Die Hochschule Landshut hat mit dem Projekt X5 Hybrid bewiesen, dass es möglich ist, seriennahe Hybridfahrzeuge zu bauen, ohne auf den gewohnten Komfort eines konventionellen Fahrzeugs verzichten zu müssen.

Für die Entwicklung der neuen Hybridfunktionen wurden die ETAS-Werkzeuge ASCET, INTECRIO, ES910 und INCA entlang des klassischen V-Modells eingesetzt. Damit konnten die Funktionen spezifiziert, abgesichert und letztendlich mit Rapid Prototyping-Technologie im Fahrzeug ausgeführt werden. So war es den Informatikern möglich, in relativ kurzer Zeit dem Fahrzeug der Maschinenbauer Leben einzuhauchen.

Nähere Informationen zum Projekt finden Sie unter [www.ex-drive.de](http://www.ex-drive.de).

Der Beitrag ist erschienen bei: Sirch, Ottmar (Hrsg.): Elektrik/Elektronik in Hybrid- und Elektrofahrzeugen II „Entwicklung einer Antriebssteuerung für ein Hybridfahrzeug in einer Rapid-Prototyping-Umgebung“: Klaus Beck, Thomas Escherle, Dieter Nazareth, expert verlag, 2010.

# Entwicklung einer Antriebssteuerung für ein Hybridfahrzeug in einer Rapid Prototyping-Umgebung



1. Einleitung
2. Das Fahrzeug
3. Die Bordnetz-Architektur
4. Funktionsentwicklung mit ASCET
  1. Fahrpedal-Logik
  2. Berechnung des maximalen Drehmoments
  3. Drehmomentverteilung
  4. Drehmomentanstiegsbegrenzung
  5. Funktionsmodellierung
5. Die Erprobung
  1. Offline-Simulation
  2. Rapid Prototyping
6. Höhere Antriebsfunktion
7. Fazit
- 8. Abkürzungsverzeichnis**

- CAN  
Controller Area Network  
Datenbus im Kraftfahrzeug und in der Automatisierungstechnik.
- CAS  
Car Access System  
Steuerung des Fahrzeugzugriffs, die u. a. den Status der Klemme 15 regelt.
- DMC  
Digital Motion Control  
Versorgt die Elektroantriebe mit Wechselspannung.
- DME  
Digitale Motorelektrik Konventionelles Motorsteuergerät des BMW X5.
- EHB  
Elektrohydraulische Bremse  
Bremsanlage des BMW X5 mit Bremskraftverstärker, ABS und ESP.
- GWS  
Getriebewahlschalter  
Wählhebel des Automatikgetriebes.
- DSC  
Dynamische Stabilitäts Control (sic!)
- PDC  
Park Distance Control (BMW)  
Parktronic (Mercedes)
- ACC  
Adaptive Cruise Control (BMW)  
Distronic, Abstandsregeltempomat (Mercedes)

## Wussten Sie

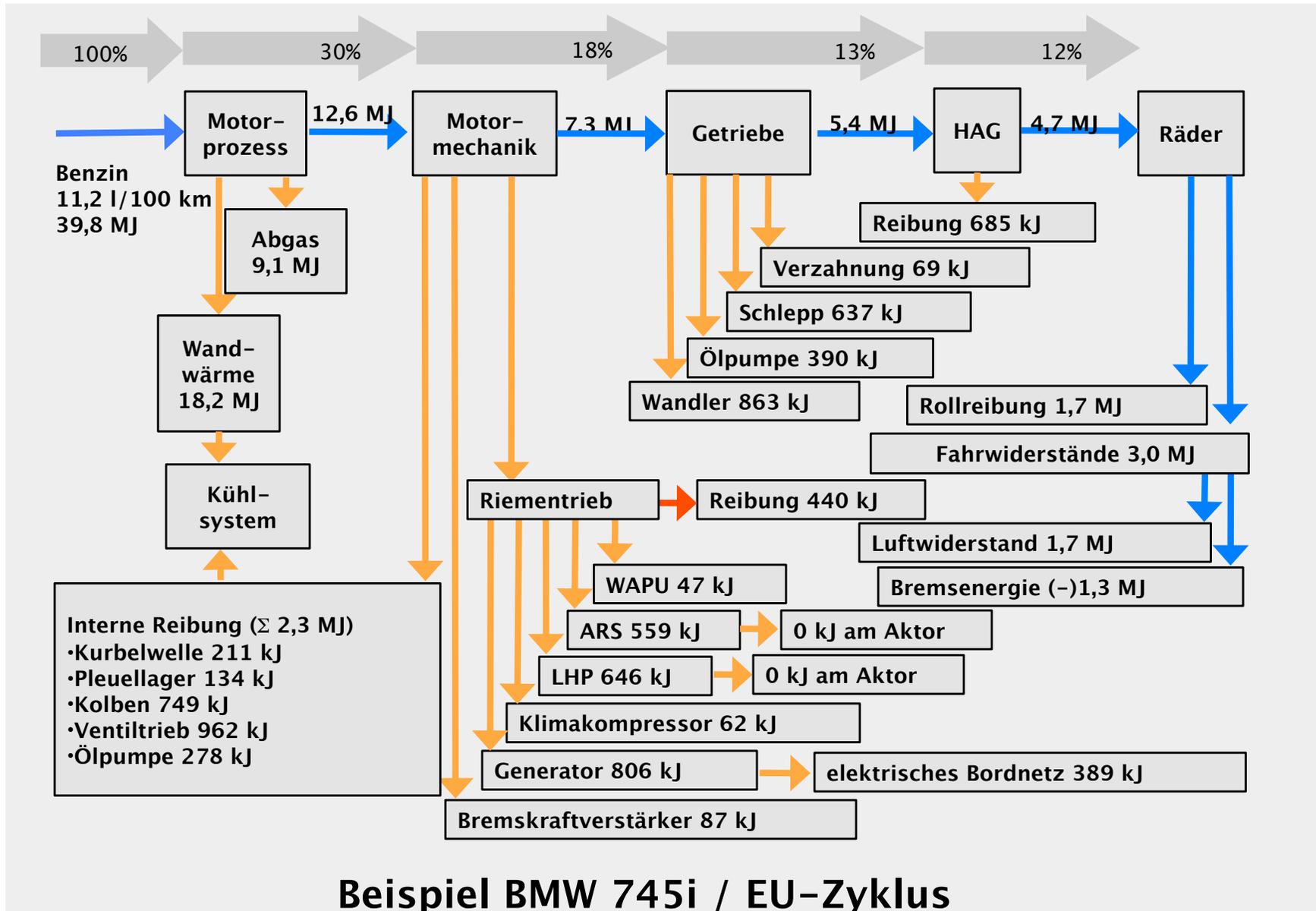


- dass bei einem PKW mit konventionellem Verbrennungsmotor nur ? % der Kraftstoffenergie bei den Reifen ankommt?  
Quelle: Technische Universität München / HdT-Tagung “Elektrik/Elektronik in Elektro- und Hybridfahrzeugen”, März 2010
- dass man mit einer vollen, 125 kg schweren Batterie eine Reichweite von 160 km erreicht, aber mit 125 kg Diesel die Reichweite 1.500 km beträgt?  
Quelle: EVONIC 2009 / ADAC Zukunftstechnologien, München 2009
- dass die Jahresproduktion von 90.000 t Lithiumcarbonat reichen, um 12 Millionen Hybridfahrzeuge auszustatten?  
Quelle: <http://www.lithiumaktien.com/> EN/566/1590

## Wussten Sie

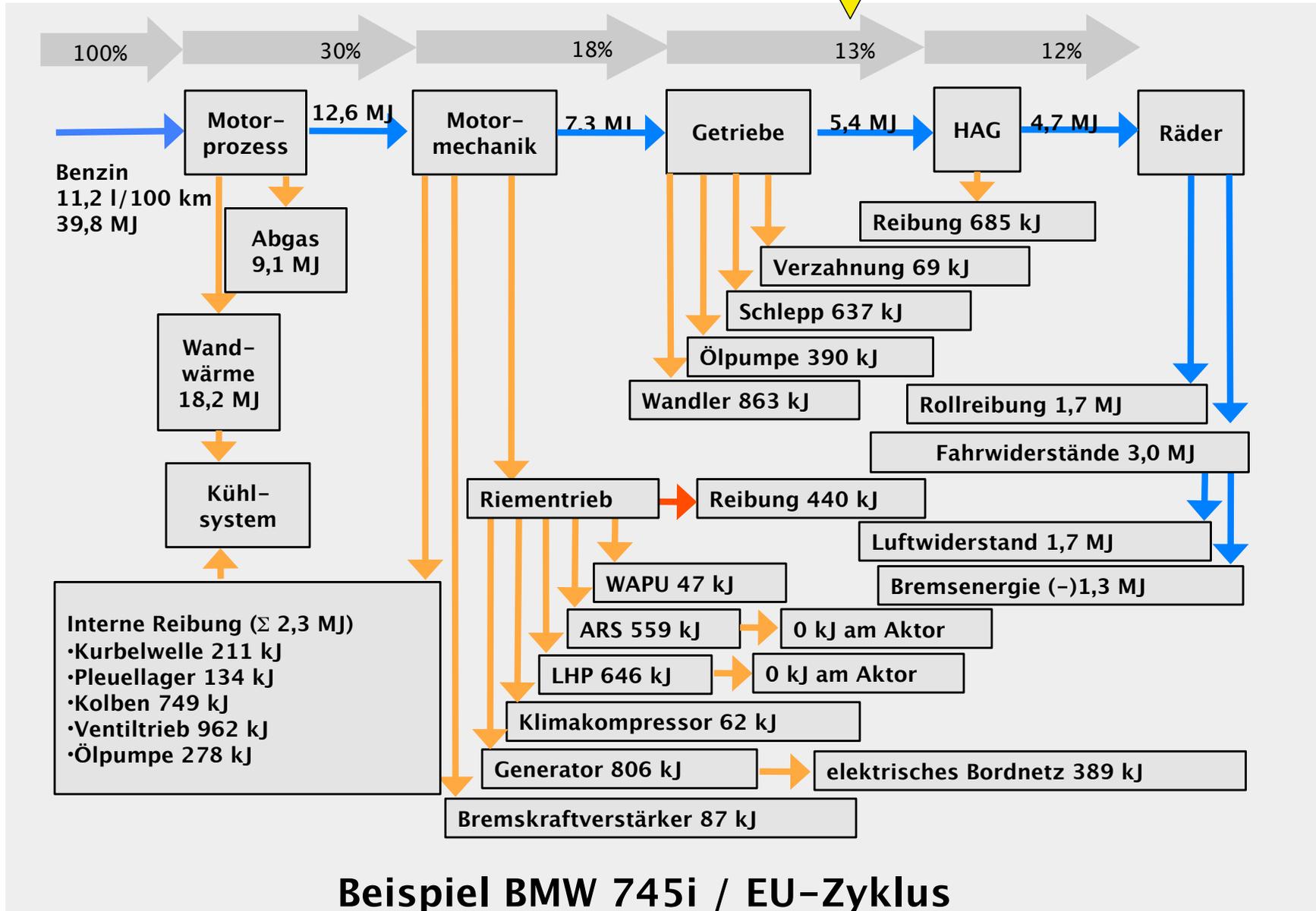


- dass bei einem PKW mit konventionellem Verbrennungsmotor nur 7 % der Kraftstoffenergie bei den Reifen ankommt?  
Quelle: Technische Universität München / HdT-Tagung “Elektrik/Elektronik in Elektro- und Hybridfahrzeugen”, März 2010
  - Brutto oder netto? (Rollwiderstand, Luftwiderstand)
- BMW EfficientDynamics – wir haben die Segel richtig gesetzt  
Dr. Johannes Liebl, BMW AG  
Vortrag auf der  
30. Tagung „Elektronik im Kraftfahrzeug“, Dresden, 16. - 17. Juni 2010
- Vortrag Neugebauer:  
Vom Saulus zum Paulus –  
Abwärme muss nichts Schlechtes sein.  
Dr. Stephan Neugebauer, BMW Group  
Leiter Wärmemanagement  
Vortrag auf der 20. Internationalen AVL Tagung „Motor & Umwelt“  
Graz, 11. – 12. September 2008

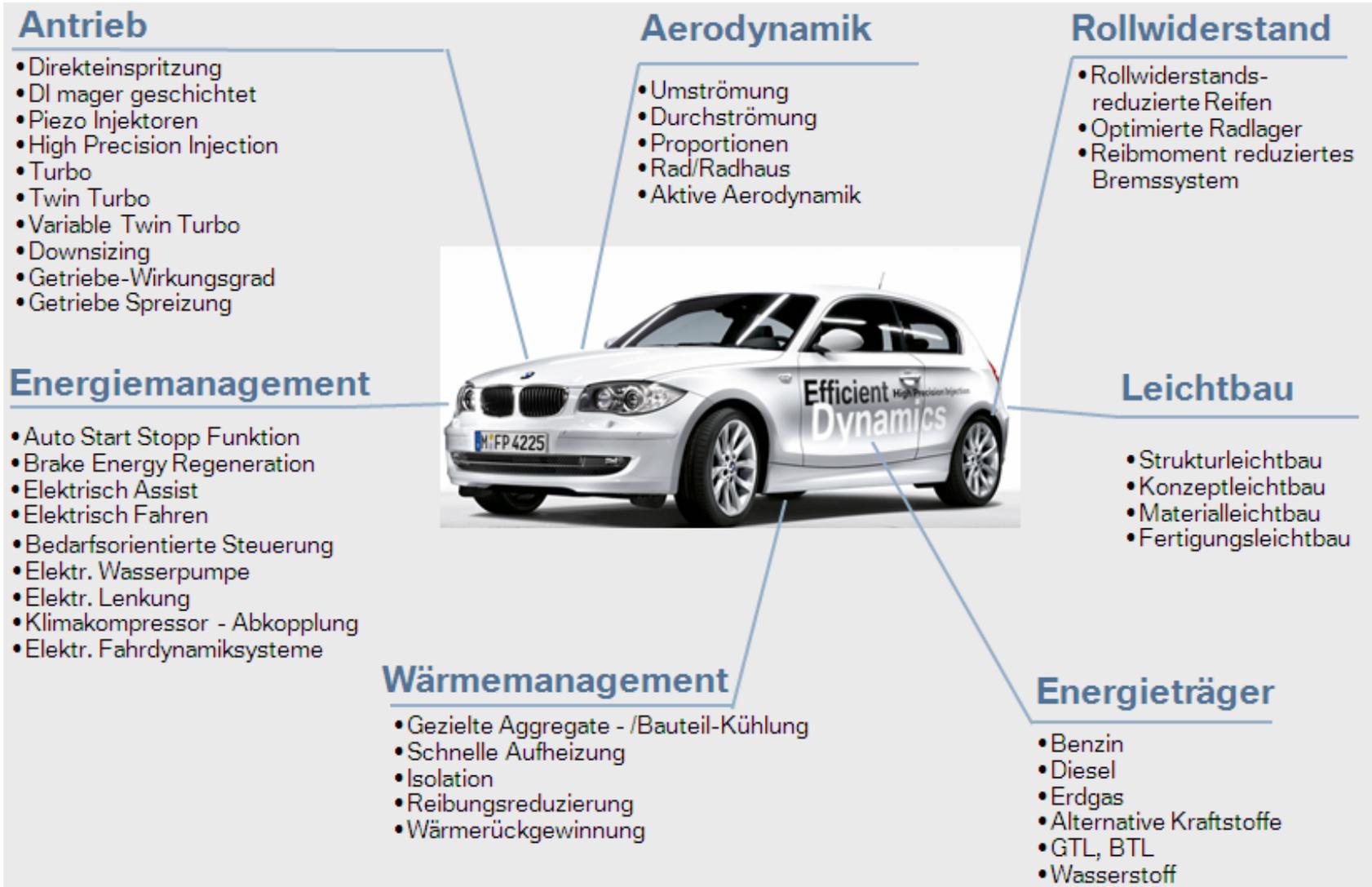


**Beispiel BMW 745i / EU-Zyklus**

## Hinterachsgetriebe



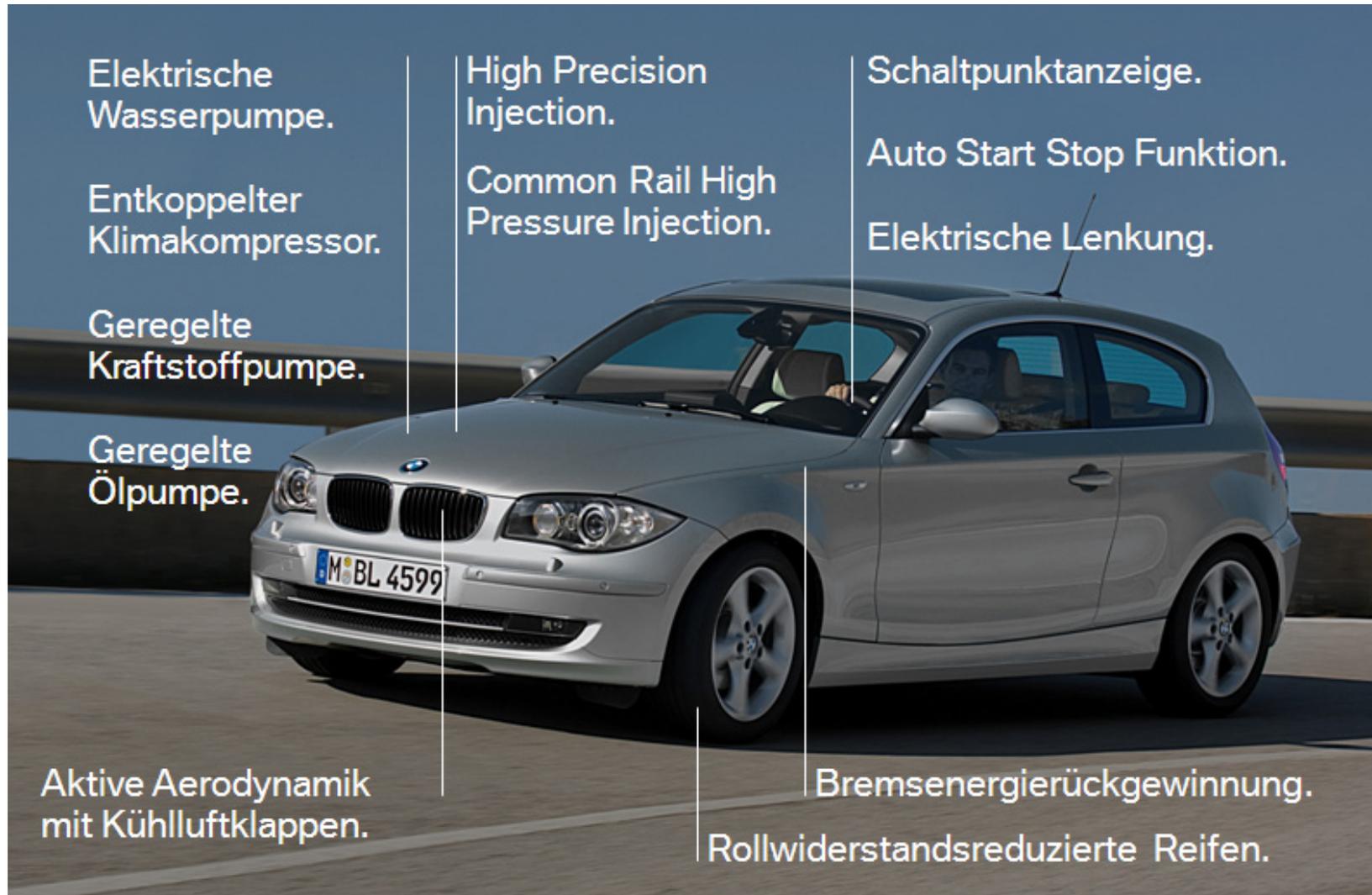
Beispiel BMW 745i / EU-Zyklus





- Systemtheorie = Vorgehensweisen zum Umgang mit Komplexität
- Ansatz: Divide et impera / Teile und Beherrsche
- Annahmen
  - Die Aufteilung des Systems in Komponenten verfälscht nicht das betrachtete Problem.
  - Die Komponenten für sich betrachtet sind in wesentlichen Teilen identisch mit den Komponenten des Systems.
  - Die Prinzipien für den Zusammenbau der Komponenten zum System sind einfach, stabil und bekannt.
- Diese Annahmen sind bei der Entwicklung von elektronischen Systemen und Software für Fahrzeuge erfüllt bzw. erfüllbar.
- Die Eigenschaften des Systems ergeben sich aus dem Zusammenspiel der Komponenten
- Komplexität des Systems führt zu komplexen und aufwändigen Analysen der Komponenten und ihres Zusammenspiels = Kern der Systemtheorie
- Komponenten
  - Technische Bauteile
  - Menschen / Benutzer
  - Umwelt

- Systemtheorie = Vorgehensweisen zum Umgang mit Komplexität
- Ansatz: Divide et impera / Teile und Beherrsche
- Annahmen
  - Die Aufteilung des Systems in Komponenten verfälscht nicht das betrachtete Problem.
  - Die Komponenten für sich betrachtet sind in wesentlichen Teilen identisch mit den Komponenten des Systems.
  - Die Prinzipien für den Zusammenbau der Komponenten zum System sind einfach, stabil und bekannt.
- Diese Annahmen sind bei der Entwicklung von elektronischen Systemen und Software für Fahrzeuge erfüllt bzw. erfüllbar.
- Die Eigenschaften des Systems ergeben sich aus dem Zusammenspiel der Komponenten
- Komplexität des Systems führt zu komplexen und aufwändigen Analysen der Komponenten und ihres Zusammenspiels = Kern der Systemtheorie
- Komponenten
  - Technische Bauteile
  - Menschen / Benutzer
  - Umwelt





## Wussten Sie



- dass bei einem PKW mit konventionellem Verbrennungsmotor nur 7 % der Kraftstoffenergie bei den Reifen ankommt?  
Quelle: Technische Universität München / HdT-Tagung “Elektrik/Elektronik in Elektro- und Hybridfahrzeugen”, März 2010
  - Brutto oder netto? (Rollwiderstand, Luftwiderstand)
- BMW EfficientDynamics – wir haben die Segel richtig gesetzt  
Dr. Johannes Liebl, BMW AG  
Vortrag auf der  
30. Tagung „Elektronik im Kraftfahrzeug“, Dresden, 16. - 17. Juni 2010
- Vortrag Neugebauer:  
Vom Saulus zum Paulus –  
Abwärme muss nichts Schlechtes sein.  
Dr. Stephan Neugebauer, BMW Group  
Leiter Wärmemanagement  
Vortrag auf der 20. Internationalen AVL Tagung „Motor & Umwelt“  
Graz, 11. – 12. September 2008

## Wussten Sie



- dass man mit einer vollen, 125 kg schweren Batterie eine Reichweite von 160 km erreicht, aber mit 125 kg Diesel die Reichweite 1.500 km beträgt?

Quelle: EVONIC 2009 / ADAC Zukunftstechnologien, München 2009

- Dichte Diesel 0,82 - 0,85, d.h. 125 kg Diesel sind ca. 150 l, 10 l / 100 km
- Gesamtenergiebilanz: Energie, Rohstoff, Verschmutzung, Arbeitskosten
  - Herstellung
  - Verteilung
  - Verbrauch / Gebrauch
  - Entsorgung
- Gewicht Batterie konstant, Gewicht Diesel nimmt ab
- Lebensdauer Batterie