

Real-Zeit-Design für eine Plattform zur Fahrzeugumfeldsensierung – ein Erfahrungsbericht

Jürgen Hötzel, Günther Kirchhof-Falter, Dagmar Münch

Robert Bosch GmbH – Abt. FV/SLD

Eschborner Landstraße 130-132

D-60489 Frankfurt

E-Mail {juergen.hoetzel, guenther.kirchhof-falter}@de.bosch.com

Zusammenfassung: Die Ableitung innovativer Produkte aus einer kostenminimierten Gesamtarchitektur des Systems „Fahrzeugumfeldsensierung“, die unter Zuhilfenahme des Produktlinienansatzes (PLA) entwickelt wurde, steht im Fokus dieses Berichtes. Durch die Verwendung einer nach den PLA-Regeln entwickelten Gesamtarchitektur des Systems ist es einerseits möglich, neue Funktionen mit minimalem Aufwand in ein bestehendes System einzufügen, beziehungsweise Funktionen zu ersetzen und die Komponentenzahl zu minimieren. Durch die gemeinsame Nutzung der HW-Ressourcen und der damit verbundenen Minimierung der Komponentenanzahl sind Einzelfunktionen in einem Funktionsverbund kostengünstiger anzubieten, wodurch sich die Kundenakzeptanz für Sicherheits- und Komfortfunktionen erhöht. Andererseits zwingt das steigende Verlangen des Endkunden nach Diensten, die Funktionen für mehr Komfort und Sicherheit zusammenfassen, zu einer Revision der gegenwärtigen Elektronikanordnung im Fahrzeug: Die Einführung einer vernetzten Client/Server-Struktur, mit deren Hilfe ein Teil der bisherigen Einzelfunktionen als Software-Lösung auf einem oder mehreren Servern dargestellt werden kann, führt nicht nur zu erheblichen Einspareffekten, sondern es entsteht die Möglichkeit zur Realisierung aktuell verlangter funktionsübergreifender Dienste in erheblich kürzerer Zeit als bisher. Durch die Verwendung einer mit dem PLA abgeleiteten System-Architektur, werden zudem bereits im ersten Systementwurf die qualitativen Anforderungen des Produktes, die oft erst während der Produktlaufzeit relevant werden, berücksichtigt. Dieser systematische Ableitungsprozess führt letztendlich zu einer bestimmten Komponentenarchitektur in Hard- und Software. Es wird eine wiederverwendbare Basis geschaffen; so wird der Entwicklungsaufwand neuer Produkte minimiert [We00].

1 Motivation

In der Automobilzuliefererindustrie werden derzeit Einzelfunktionen entwickelt, die auf individuelle Fahrzeugumfeldsensierung zurückgreifen, wie beispielsweise ACC Stop&Go, PreCrash-Detektion, Einparkhilfe und Tote-Winkel-Detektion. Nachteilig ist, dass jede einzelne Funktion dazu eigene Hard- und Software, also eigene Sensoren und Steuergeräte, benötigt.

Sollen mehrere Funktionen im Fahrzeug so realisiert werden, erhöht sich bisher die Anzahl der Sensoren und Steuergeräte mit der Anzahl der Funktionen entsprechend. Die Nachteile liegen auf der Hand:

- gegenseitige Beeinflussung der Sensoren,
- hoher Platzbedarf für die Hardware-Komponenten,
- für jedes Einzelsystem entsprechende Kosten,
- und geringe Kundenakzeptanz durch hohen Gesamtpreis bei Vollausrüstung.

Vorteilhaft ist aber, dass Umfeldsensorik eine weitere Verbesserung der passiven Sicherheit des Fahrzeuges ermöglicht: Mittels einer PreCrash-Funktion ist die bisherige Auslöseperformance sowie die Robustheit und die Sicherheit vor unbeabsichtigter Auslösung der Rückhaltesysteme zu verbessern. Darüber hinaus ermöglicht die auf der Umfeldsensorik basierende PreCrash-Funktion die Ansteuerung reversibler Rückhaltesysteme vor Crashbeginn zum optimierten Schutz im Crashfall. Diese Funktion ist insbesondere für die Überwachung von Fahrzeugfront und -seiten unter dem Aspekt des Insassenschutzes von Interesse. Nahbereichsensorik kann auch zu einer Erweiterung einer Adaptive Cruise Control (ACC) um eine Stop&Go-Funktionalität (Staufolgefahrt) genutzt werden. Zur weiteren Verbesserung der Nachtsicht werden Fern- und Nacht-Infrarotsicht-Systeme verfolgt.

Mittel- bis langfristig realisierbare Funktionen auf Basis von Umfeldsensoren sind beispielsweise (siehe auch Abbildung 1):

- Funktionen, die die Abdeckung des Ultranahbereichs und des Nahbereichs erfordern:
Einparkhilfe [HS96], PreCrash-Erkennung, ACC Stop&Go, PreCrash-Erkennung Seite, Parklückenvermessung, Tote-Winkel-Detektion
- Funktionen, die die Abdeckung des Fernbereichs erfordern:
Adaptive Cruise Control, Fahrwerksteuerung, Lichtsteuerung
- Funktionen, die die Abdeckung des mittleren Bereichs mittels Videosensorik erfordern:
Adaptive Cruise Control, Fahrzeugquerführung durch Video-Auswertung, Verkehrszeichenassistentz

Erst die Kombination der oben genannten Funktionen zu einem System führt zu einem erheblichen Mehrwert bei Komfort und Sicherheit. Gerade im Frontbereich des Fahrzeuges werden die Vorteile der Verknüpfung besonders deutlich: Die 77 GHz-Sensoren decken den Fernbereich von 3 m bis 120 m ab und sind somit Datenlieferant für die Funktionalität der heutigen ACC. Allerdings erkennen diese Sensoren nur Entfernung und relative Geschwindigkeit zum Objekt; es sind keinerlei Aussagen über dessen Art, Form oder Größe möglich. Diese Aufgabe könnte die Videosensorik übernehmen. Spurwechsel und Abbiegen des Vorfahrenden sind so frühzeitig zu erkennen, das eigene Fahrzeug selbständig auf der Spur zu halten [SSK00].

Die Short-Range-Radar-Sensoren (24 GHz) detektieren den Bereich von 0,3 m bis 14 m vor dem Fahrzeug und unterstützen somit die ACC-Funktion im Nahbereich. Die Umschaltung zwischen den Sensortechnologien erfolgt vom Fahrer unbemerkt. Damit ist es möglich, ACC um die Funktion „Stop&Go“ zu erweitern, die eine weitgehend automati-

sierte Staufahrt ermöglicht. Die Geschwindigkeit des Fahrzeuges wird zu jeder Zeit an die Anforderungen der Umgebung angepasst; der Fahrer wird bei sicherem, vorausschauendem und umweltgerechtem Fahren aktiv unterstützt.

Etwa ein Drittel aller Unfälle geschieht durch unbeabsichtigtes Abkommen von der Spur oder unvorsichtiges Spurwechseln. Auch hier kann die Kombination von Radar- und Videosensorik helfen: Die Videosensoren erkennen anhand der Fahrbahnmarkierungen, ob sich das Fahrzeug korrekt auf der Fahrspur befindet, die Radar-Sensorik detektiert Verkehrsteilnehmer, die sich im toten Winkel des eigenen Fahrzeugs befinden. Da im Detektionsbereich der Videosensorik sowohl Verkehrszeichen als auch der weitere Fahrweg liegen, können unvorsichtige oder verbotene Überholmanöver über Fahrhinweise vermieden werden [HK01].

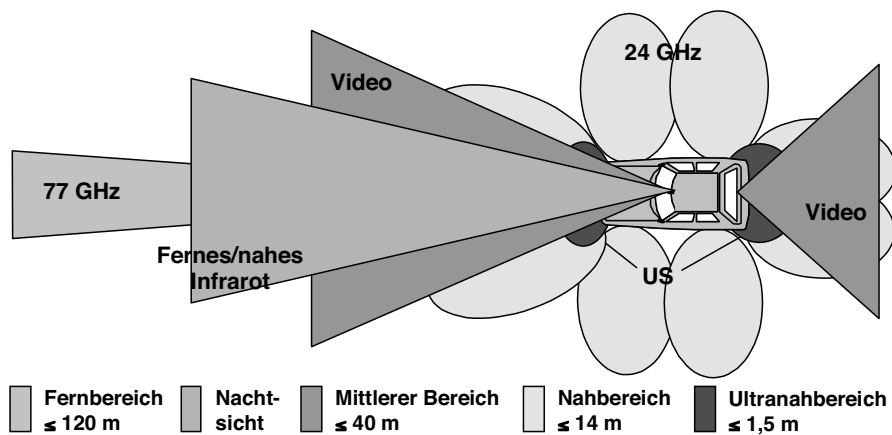


Abbildung 1: Übersicht Sensortechnologien

Die Verknüpfung der Messdaten von Video- und Radar-Sensoren ermöglicht zusätzlich die rechtzeitige Erkennung von Fußgängern. Die zum Schutz dieser Verkehrsteilnehmer im Fahrzeug z. B. ebenfalls zukünftig vorhandene Auslöseaktorik zum kurzzeitigen Anheben der Motorhaube, kann so rechtzeitig vor dem Zusammenstoß aktiviert werden [KWK01].

Die in diesem Bericht beschriebene Modellierungsmethodik basiert auf dem Produktlinienansatz, beschrieben in [BCK98].

2 Entwicklung in der Domäne Umfeldsensorik nach dem Produktlinienansatz

In der derzeitigen Softwareentwicklung wird Software überwiegend nach funktionalen Aspekten erstellt. Dies bedingt eine fortwährende Nachentwicklung bei Erweiterungen oder Veränderungen der Anforderungen der Produkte im Feld.

Bei einer System-Entwicklung nach dem Produktlinienansatz werden von der ersten Konzeption an bereits die qualitativen Eigenschaften des Produktes mitberücksichtigt.

Schon im ersten Entwurf der zu entwickelnden Produktlinie betrachtet man auch die Anforderungen, die erst später während der Produktlaufzeit relevant werden und die bisherigen Nachentwicklungen bedingen. Eine wichtige qualitative Anforderung im Bereich der Umfeldsensorik stellt die Konfigurabilität in der Domäne dar. Die Domäne Umfeldsensorik soll so entworfen werden, dass eine Unabhängigkeit von den heutigen Sensorarten gegeben ist, damit auch spätere Sensortechnologien problemlos hinzugefügt werden können. Somit können im dann entstehenden neuen Produkte wesentliche Teile der bestehenden Architektur wiederverwendet werden, das neue Produkt kann ohne großen Aufwand und in kürzester Zeit auf den Markt kommen .

3 Domänen-Engineering

Im Domänen-Engineering wird eine wiederverwendbare Wissensbasis des Systems geschaffen. Die funktionalen Anforderungen der zu entwickelnden Produktlinie werden in der Funktionsarchitektur abgebildet. Zu den Funktionen wird eine Tabelle von nicht-funktionalen Anforderungen mitgeführt (Ausführungszeit, Ausfallwahrscheinlichkeit, ...), die von der Systemarchitektur befriedigt werden müssen. Der systematische Ableitungsprozess führt zu einer Komponentenarchitektur in Hard- und Software. Das Domänen-Engineering lässt sich in folgende Schritte aufteilen:

- Auswählen der Domäne (Scoping)
- Funktionale Domänenanalyse aus Nutzer-Sicht
- Qualitative Domänenanalyse
- Domänenendesign.

3.1 Auswählen der Domäne

Der Prozess der Fahrzeugnutzung lässt sich aus Fahrersicht unterteilen in den folgenden Ablauf: Einsteigen und Justieren, Fahren von A nach B und Ankommen sowie Aussteigen und Sichern.

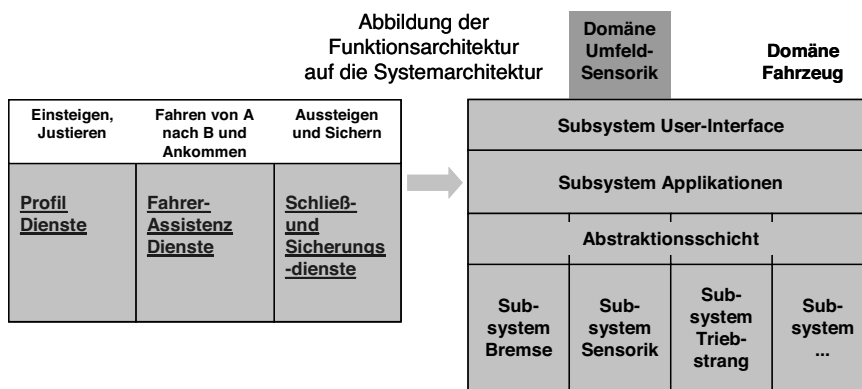


Abbildung 2: Dienste- und Systemstruktur

Auf Basis dieses Prozesses lassen sich folgende Funktionen beziehungsweise Dienste unter Verwendung von Umfoldsensoren identifizieren:

Einsteigen und Justieren: Justieren der Fahrzeugeigenschaften (Anhänger, Heckgepäckträger)

Fahren und Ankommen: oben beschriebene Dienste unter Verwendung der Umfoldsensierung

Aussteigen und Sichern: Passive Entry, Diebstahl- und Vandalismusschutz

Die Abbildung 2 verdeutlicht den Zusammenhang zwischen der oben beschriebenen Dienststruktur und einer möglichen Systemstruktur.

Im Folgenden liegt nur der hervorgehobene Bereich der Abbildung, also der Teil der Gesamtdomäne Fahrzeug, der vom Subsystem Sensorik mit Informationen bedient wird, im Fokus der Domänenbeschreibung [Fe00]. Es können langfristig aber auch Dienste bzw. Funktionen mit Daten bedient werden, die Einfluss auf andere Subsysteme haben wie beispielsweise die Subsysteme Triebstrang oder Bremse.

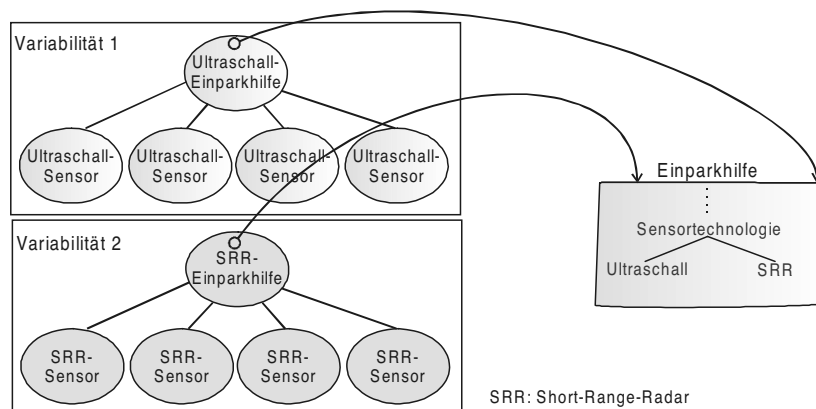


Abbildung 3: Variabilitäten einer Einparkhilfe

3.2 Funktionale Domänenanalyse aus Nutzer-Sicht

In diesem Arbeitsschritt werden die bekannten sowie zu erwartenden Funktionalitäten der Domäne unter Zuhilfenahme aller Kundenanforderungen spezifiziert, also konkrete Anwendungsfälle (Use Cases), beschrieben. Durch Zusammenfassen gleicher Instanzen aus verschiedenen Use Cases entsteht die Funktionsarchitektur des Systems. Diese Funktionsarchitektur kann man sich als eine dreidimensionale Strukturierung *aller* Kundenanforderungen vorstellen: Hinter jeder Instanz der Funktionsarchitektur liegt ein Feature-Baum, der alle Anforderungen dieser Instanz geordnet aufzeigt und somit die Gemeinsamkeiten, Variabilitäten und Ausschlussbedingungen zwischen den einzelnen Kundenwünschen visualisiert.

Die Abbildung 3 zeigt zwei sehr stark vereinfachte Variabilitäten der Funktion Einparkhilfe. Die Einparkhilfe kann entweder mit Ultraschall-Sensoren (bisherige Realisierung) oder mit Short-Range-Radar-(SRR)-Sensoren umgesetzt werden. Die eingesetzte Sensortechnologie variiert, stellt also Variabilitäten zwischen einzelnen Kundenwünschen der Funktion Einparkhilfe dar.

3.3 Qualitative Domänenanalyse

Hier werden alle nicht-funktionalen Anforderungen, die Funktionen nicht direkt zuzuordnen sind, gesammelt, wie Fehlererkennung, Kostenminimierung, Konfigurabilität usw. Die Gesamtheit der Anforderungen bestimmt die Systemarchitektur. Ein Teil davon stellt dominante Architekturtreiber dar.

3.4 Domänen-Design für die Systemarchitektur

Die Ergebnisse der Funktionsarchitektur und der qualitativen Analyse werden in der Systemarchitektur zusammengeführt. Im Beispiel der Domäne Umfeldsensorik sind nicht nur die oben bereits genannten Anforderungen entscheidend, sondern auch – als weiteres Qualitätsattribut – die Unterstützung unterschiedlichster Sensortechnologien. In der Abbildung 3 erkennt man die Notwendigkeit einer definierten Schnittstelle zwischen den Sensoren und der Verarbeitung der Signale für die Funktion Einparkhilfe: Am kostengünstigsten und flexibelsten ist sicherlich die Entwicklung *genau einer* Signalverarbeitung für die Einparkhilfe, die unabhängig von der eingesetzten Sensortechnologie ist. Dies erreicht man, indem alle Sensortechnologien Beiträge in eine standardisierte Umgebungsbeschreibung liefern.

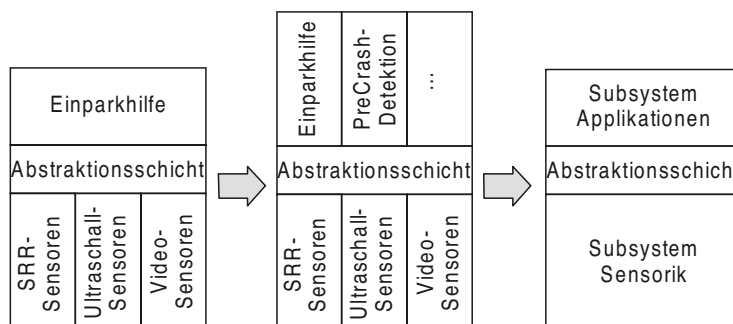


Abbildung 4: Von der Einparkhilfe zur Subsystemstruktur

Die davon abgeleitete Abstraktionsebene erlaubt Technologieunabhängigkeit in der Weiterverarbeitung für weitere, auf Sensordaten zugreifende Funktionen, wie beispielsweise die PreCrash-Erkennung oder die Tote-Winkel-Detektion. Diese so gewonnene Strukturierung der Domäne Umfeldsensorik in konzeptionelle Systemkomponenten (Abbildung 4) entspricht der in Abbildung 2 bereits gezeigten Systemstruktur der Domäne Fahrzeug.

Gemäß Abbildung 4 sind die Ergebnisse der vorhergehenden Analyse in konzeptionelle Subsysteme umgesetzt worden. In einem darauffolgenden, verfeinernden Schritt sind aus den Subsystemen konzeptionelle Hard- und Software-Komponenten abzuleiten. Wobei sich konzeptionelle Hard- und Software-Komponenten lediglich in ihren Beschreibungsparametern unterscheiden, sonst aber noch gleich zu behandeln sind.

Alle Funktionen, d. h. sicherheitsrelevante (beispielsweise PreCrash) wie auch Komfort-Funktionen (Einparkhilfe), greifen auf die gleichen Sensordaten zu und werten diese aus. Die Anforderungen der einzelnen Funktionen an die Sensordaten werden einerseits durch die Produkt-/Prototypentwicklungen andererseits aber auch durch Marktstudien, Geschäftsstrategien des eigenen Unternehmens, Wettbewerbsanalysen und Endkundenbefragungen ermittelt. Beispielhaft wird dies im Weiteren an den folgenden Funktionen Einparkhilfe, Tote-Winkel-Detektion und PreCrash-Detektion gezeigt. Abgebildet in Use Cases ergeben sich die in Abbildung 5 dargestellten Diagramme.

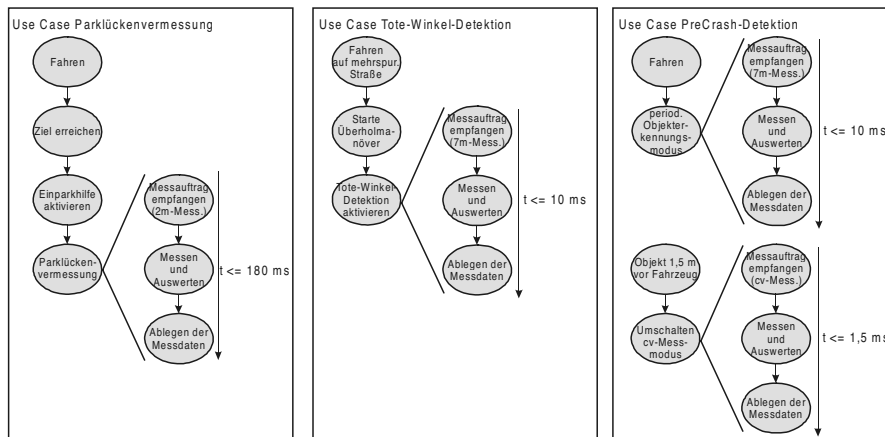


Abbildung 5: Use Cases der Funktionen Parklückenvermessung, Tote-Winkel-Detektion und PreCrash-Detektion

Betrachtet man die Anforderungen an die Messaufgaben aus Abbildung 5, erkennt man die unterschiedlichen Ansprüche der einzelnen Funktionen an das Subsystem: zum einen eine periodische Messung des Abstandes mit unterschiedlicher Messentfernung, zum anderen eine ereignisgesteuerte Messaufgabe zur Messung der Relativgeschwindigkeit (cv-Messung). Diese Aufgaben sind zunächst einmal unabhängig von der eingesetzten Sensortechnologie (Tabelle 1).

Funktion	Anforderungen an das Subsystem Sensorik
Parklückenvermessung	periodische 2 m-Messung alle 10 ms
Tote Winkel Detektion	periodische 7 m-Messung alle 10 ms
PreCrash-Detektion	durch die Relativgeschwindigkeit zwischen eigenem Fahrzeug und detektiertem Objekt bestimmte Messung der Relativgeschwindigkeit, des Abstandes, des Versatzes und des Aufprallwinkels zwischen beiden (cv-Messung)

Tabelle 1: Funktionen und ihre funktionalen Anforderungen an das Subsystem Sensorik

Bildet man die Messbeauftragung und deren Abarbeitung aus den Use Cases aus der Abbildung 5 in eine Funktionsarchitektur des Subsystem Sensorik ab, indem man gleiche Instanzen aus verschiedenen Use Cases zu einer Instanz, die mit den verschiedenen Features hinterlegt ist, zusammenfasst, ergibt sich Folgendes (Abbildung 6):

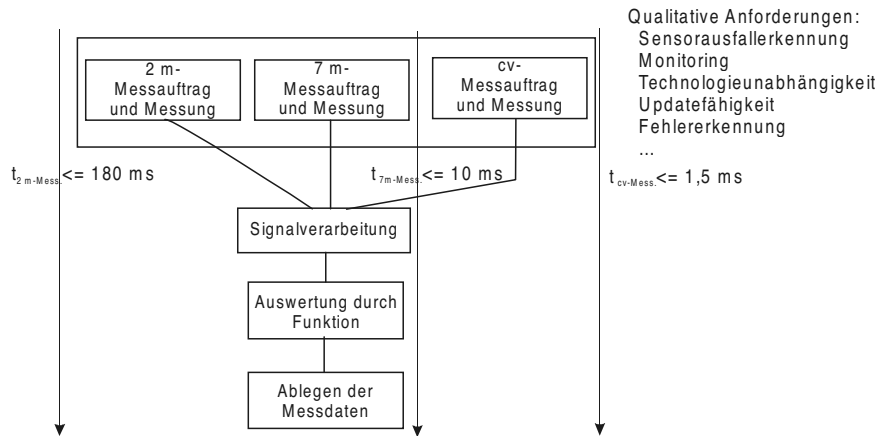


Abbildung 6: Funktionsarchitektur

4 Ableitung konzeptioneller Hard- und Software-Komponenten

Wie bereits erwähnt, werden die Ergebnisse der Funktionsarchitektur und der qualitativen Analyse zur Systemarchitektur zusammengeführt. Die einzelnen funktionalen und nicht-funktionalen Anforderungen müssen direkt auf die Systemkomponenten abbildbar sein. Weitere Abbildungsfunktionen sind durch technische Randbedingungen wie Buslast, gleichmäßige Verteilung von Rechenaufgaben, ... gegeben. Da die qualitativen Anforderungen, die u. a. stark durch Geschäftsinteressen geprägt werden, einen großen Einfluss nehmen, ergibt sich eine sehr spezifische Systemarchitektur (Abbildung 7).

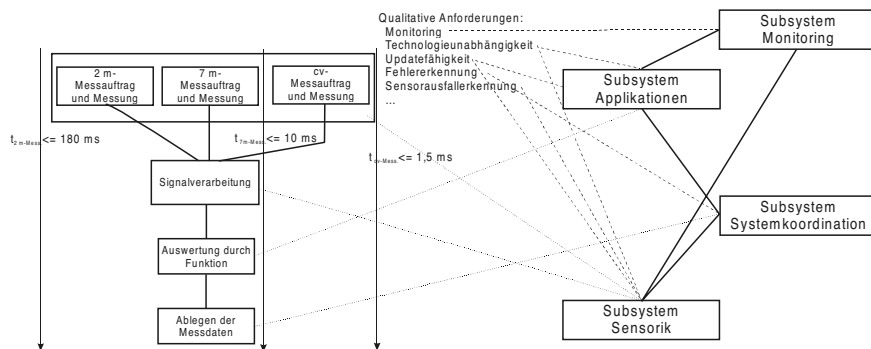


Abbildung 7: Abbildung von der Funktionsarchitektur auf eine Systemarchitektur

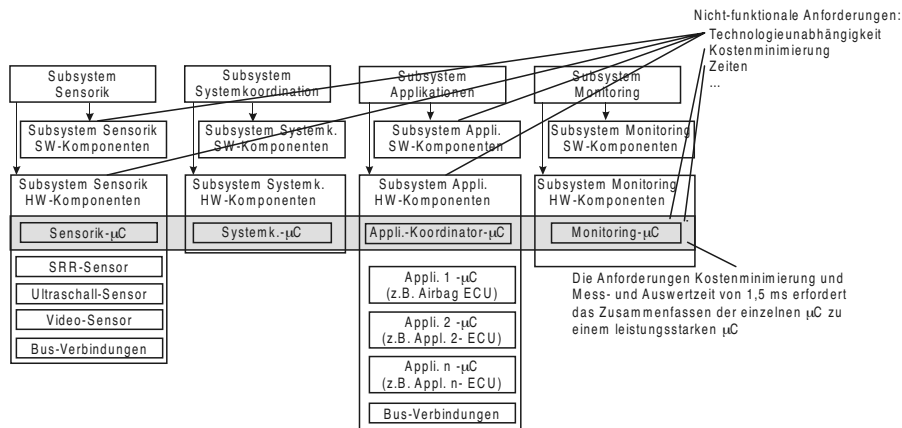


Abbildung 8: HW-/SW-Komponenten der Systemarchitektur

Aus dieser Systemarchitektur lassen sich nun die Hard- und Softwarekomponenten der konzeptionellen Subsysteme durch Berücksichtigung qualitativer Anforderungen, die teilweise bereits in der Systemarchitektur enthalten sind, ableiten (Abbildung 8).

Man sieht, dass die qualitativen Anforderungen einen erheblichen Einfluss auch auf die Hardware-Strukturierung des Systems Umfeldsensorik haben. Die Systementwicklung nach dem PLA führt zu einer optimierten Hardwarestruktur; die Systemkosten werden durch die zeitgleiche Entwicklung der Hard- und Softwarestruktur minimiert. Der Systemaufbau beinhaltet sowohl die funktionalen wie auch die nicht-funktionalen Anforderungen [Th01].

5 Beschreibung der einzelnen Subsysteme

5.1 Subsystem Sensorik

Es wird in den Anforderungen der einzelnen Funktionen nicht festgelegt (Abbildung 5, Tabelle 1), wie diese Aufgaben durchgeführt werden sollen. Die verwendete Sensortechnologie wird durch die funktionalen und nicht-funktionalen Anforderungen des Gesamtsystems bestimmt. Es können jedoch auch spezielle Kundenwünsche umgesetzt werden. So wird derzeit eine Einparkhilfe mit Ultraschallsensoren realisiert; aber auch eine Realisierung mit Short-Range-Radarsensoren (SRR-Sensoren) ist möglich. Die hinzukommende Funktion PreCrash benötigt SRR-Sensoren, da derzeit nur diese Sensoren in den Relativgeschwindigkeitsmessmodus (cv-Modus) umschaltbar sind, die jedoch wesentlich teurer sind als Ultraschallsensoren. (Die Umsetzung der Funktion PreCrash mit allen ihren Möglichkeiten, wie der situations- und ortsgenauen Ansteuerung einzelner Airbags, erfordert 4 Short-Range-Radarsensoren.) Jeder Kunde kann die konkrete Umsetzung bestimmen, auf die Subsystem-Struktur nimmt die Entscheidung keinen Einfluss.

Die Ultraschallsensoren können lediglich bis zu einer Entfernung von 2 m, die SRR-Sensoren bis zu 14 m messen; in den cv-Messmodus sind nur die SRR-Sensoren um-

schaltbar. Die Sensoren bekommen sensorspezifische Messaufträge, die durch die Funktionen vorgegeben sind. Dies und die qualitativen Anforderungen, wie beispielsweise die Sensorausfallerkennung, machen die Einführung eines Subsystemmanagers unabdingbar: Dieser dient als Schnittstelle zwischen dem Subsystem Sensorik und den anderen Subsystemen (Abbildung 9). Er empfängt die Messanforderungen der Funktionen und verteilt die Messaufgaben nicht nur nach geeigneter Sensortechnologie sondern auch nach örtlicher Präsenz am Fahrzeug.

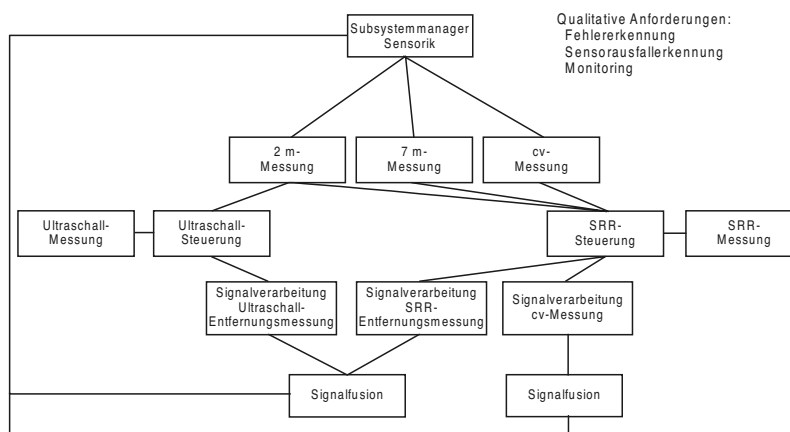


Abbildung 9: Funktionsarchitektur des Subsystems Sensorik mit Subsystemmanager

Gleichzeitig stellt er eine Liste mit Objekten und deren Attribute, die aus der Fusion der Messdaten der einzelnen Sensoren hervorgegangen ist, den anderen Subsystemen zur Verfügung. Der Subsystemmanager gibt Messaufträge an die Sensoren. Sensor-ID und -Bauart – wie auch Kalibrierung und Funktionstest – werden während der Initialisierung des Systems (also bei jedem Start des Fahrzeugs) durch den Subsystemmanager abgerufen. Der Aufbau dieses Subsystem erfolgt nach dem Master/Slave-Strukturmuster: Sensoren sind Slaves, Subsystemmanagement ist Master.

Die vom Sensor ermittelten Rohdaten werden zunächst einzeln ausgewertet. Die Abbildung auf die Hardware ist hier noch nicht erfolgt; das Modell ist so flexibel, dass diese erste Signalverarbeitung, also die Erzeugung einer Liste mit Objekten und deren Attribute aus den Daten *eines* Sensors, entweder auf einem intelligenten Sensor (Slave) selbst oder auf dem Master erfolgen kann. Die Fusion der Messdaten mehrerer Sensoren (Triangulation) muss hingegen zentral auf dem Master erfolgen, um eine möglichst genaue Ortsauflösung zu erhalten. Die Zeit, die dem System für diese Verarbeitung maximal zur Verfügung steht, ist abhängig von den eingesetzten Sensoren.

Das Subsystem Sensorik beinhaltet daher die folgenden Funktionalitäten:

- Der Aufbau des Subsystem erfolgt nach dem Master/Slave-Strukturmuster.
- Dem Subsystem sind alle Sensoren bekannt.
- Alle Sensoren werden verwaltet und überwacht.
- Alle Sensoren müssen im Echtzeitmodus angesteuert werden.
- Messaufträge werden sensorspezifisch erteilt.

- Refresh-Zyklen der Sensordaten müssen funktionsgerecht erfolgen.
- Subsystem oder Subsystemteile sind updatefähig.

5.2 Abstraktionsebene

Die Objektliste bildet die Grundlage für die weitere Auswertung durch Funktionen und Dienste, denen diese über die Abstraktionsebene zur Verfügung gestellt wird. In Abbildung 10 wird dies deutlich: Der SystemCoordinator bildet die Abstraktionsebene. Er führt eine jederzeit aktuelle Liste der angemeldeten, zu bedienenden Funktionen mit Informationen über die von ihnen benötigten Daten und kennt andererseits das Subsystem Sensorik mit den angeschlossenen Sensortechnologien und mit den sich bietenden Messmöglichkeiten und -bereichen.

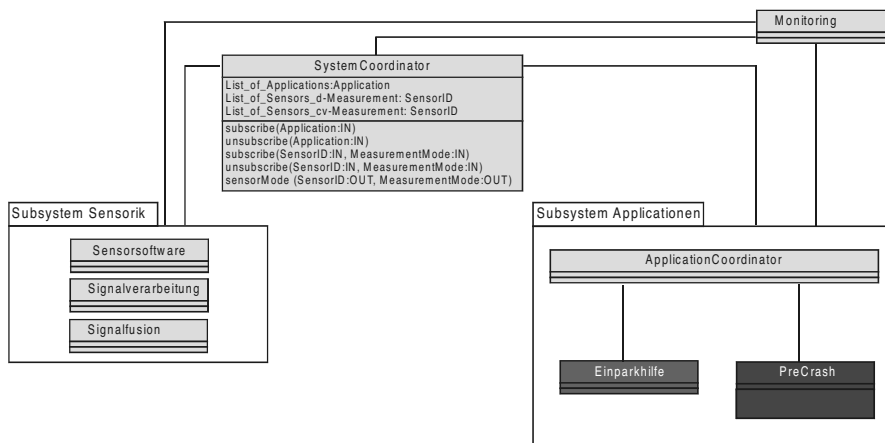


Abbildung 10: Software-Komponenten

5.3 Subsystem Applikationen

Dieses Subsystem ist nach dem Client/Server-Prinzip aufgebaut. Die einzelnen (bereits oben beschriebenen) Funktionen bilden dabei die Clients. Das Subsystem ist so modelliert, dass die Clients entweder auf derselben Hardware wie der Server (Subsystemmanager) oder auf einer separaten abgebildet werden können. Die im Subsystem Applikationen vorhandenen, freigeschalteten Funktionen (Clients) melden sich mit den von ihnen gewünschten Informationen bei ihrem Koordinator (Server) an. Beispielsweise fordert eine Einparkhilfe Messdaten an, wenn die Fahrzeuggeschwindigkeit unter 20 km/h liegt oder der Rückwärtsgang eingelegt wurde. Dieser Koordinator bedient die Funktion Einparkhilfe im System durch Übergabe der gemessenen Werte nur, wenn genau diese Voraussetzungen erfüllt sind.

6 Resümees

Durch Verwendung einer System-Architektur, die mit dem Produktlinienansatz (PLA) abgeleitet ist, werden bereits im ersten Systementwurf die qualitativen Anforderungen des Produktes, die oft erst während der Produktlaufzeit relevant werden, berücksichtigt. Dieser systematische Ableitungsprozess führt letztendlich zu einer bestimmten Komponentenarchitektur in Hard- und Software. Es wird eine wiederverwendbare Basis geschaffen; zwangsläufig wird so der Entwicklungsaufwand neuer Produkte minimiert.

Die Strukturierung erfordert spezifizierte Schnittstellen zwischen den Funktionen und dem Subsystemmanager Applikationen (Client/Server-Struktur) einerseits und dem Messbereich, also den Sensoren, und dem Subsystemmanager Sensorik (Master/Slave-Struktur) andererseits. Nur so wird sichergestellt, dass eine von der Sensortechnologie unabhängige Systemarchitektur entsteht. Auch bei technologischen Innovationen der Sensoren bleibt die jetzige Struktur erhalten.

Ein weiteres Argument für diese Subsystemlösung ist die Zusammenfassung der verschiedenen Sensorsystemdaten, die zukünftig an weitere Subsysteme, wie Motor- und Getriebesteuerung, Bremsen, Airbag-Steuergerät, oder an den Fahrer weitergereicht werden.

Literatur

- [HS96] Hötzel, J., Stahl, W.: Parktronic-System (PTS), aktueller Stand und Ausblick. Elektronik im Kraftfahrzeug 1996, Baden-Baden
- [BCK98] Bass, L., Clements, P., Kazman R.: Software Architecture in Practice, Addison Wesley Longman, Inc., 1998
- [Fe00] Ferber, S.: Eine Fallstudie für den Produktlinienansatz bei Bosch: Fahrzeugumfeldsensorik, 1. Deutscher Produktlinienworkshop, FhG IESE, Kaiserslautern, 09.11.2000
- [We00] Weber, J.: Notwendigkeit von Referenzarchitekturen für IT-Strukturen im Fahrzeug, Elektronik im Kfz, Baden-Baden, 05.-06.10.2000
- [SSK00] Seger, U., Stiller, Ch., Knoll, P. M.: Sensor Vision and Collision Warning Systems. Invited Paper für die Convergence (2000)
- [HK01] Hötzel, J.; Knoll, P. M.: Mehr Fahrerunterstützung mit Umfeldsensorik, Fachtagung Sensoren zum Messen mechanischer Größen im Kfz, Essen, 15./16.02.2001
- [Th01] Thiel, S., Ferber, S., Fischer, T., Hein, A. Schlick, M.: A Case Study in Applying a Product Line Approach for Car Periphery Supervision Systems, in: Proceedings of In-Vehicle Software 2001, SAE World Congress 2001, Detroit, USA, 05.-08.03.2001
- [KWK01] Knoll, P. M., Winner, H. Kallenbach, R.: Surround Sensing – Collision Warning Systems – Vehicle Guidance, ATA EL 2001, Lago Maggiore Juni 2001