SAE International™

Domain Control Units - the Solution for Future E/E Architectures?

2010-01-0686
Published
04/12/2010

Wolfgang Stolz, Robert Kornhaas and Ralph Krause
Robert Bosch GmbH

Tino Sommer
Bosch Engineering GmbH

ABSTRACT

In order to master the increasing complexity of electrical/ electronic (E/E) systems in vehicles, E/E architecture design has become an established discipline. The task of the E/E architecture design is to come up with solutions to challenging and often contradictory requirements such as reduced cost and increased flexibility / scalability. One way to optimize the E/E architecture in terms of cost (electronics & wiring harness) is to integrate functions. This can be done by either combining functions from multiple ECUs into a single ECU or by introducing Domain Control Units. Domain Control Units provide the main software functionality for a vehicle domain, while relegating the basic functions of actuator control to connected intelligent actuators. Depending on the different market segments (low price, volume and premium) and the different vehicle domains, the actual usage of Domain Control Units can be quite different and sometimes questionable.

In this paper, the potential use of Domain Control Units is evaluated for the different vehicle domains with respect to the main drivers, including technical as well as functional trends. This evaluation is based on generic architecture patterns for integration. In addition, the future introduction of Domain Control Units in the three different market segments is evaluated.

INTRODUCTION

Driven by innovations in the areas of driver information, driver assistance, passenger safety and propulsion the number of electrical and electronic components is continuously increasing in modern cars.

The availability of new sensors for the vehicle surrounding (e.g. camera systems, radar sensors, ultrasonic sensors) offers the opportunity to develop new functions for driver information and driver assistance by combining the available sensor data. Using these additional data increases the interconnectedness between existing functions in the vehicle and it also increases the processing power requirements.

Another trend that influences the E/E architecture is the development of new propulsion systems: mild and full hybrid, range extender and electrical engines. This increases the demand for addressing system variance in the E/E architecture.

This leads to an increasing challenge for the layout of the E/E architecture in vehicles. In order to master the complexity in designing E/E systems, the E/E architecture design has introduced a hierarchical approach for the allocation and realization of functions in the system. The main strength of the E/E architecture design approach is the differentiation between the logical level and the implementation level. The logical level defines the functional chains which are necessary to realize the features that must be implemented (e.g. active front steering). The implementation level defines the technical realization of functions and their allocation to components in the vehicle.

One approach to handle the increasing interconnectedness as well as the increasing variance is the introduction of “central managers” (e.g. safety manager or motion manager) via Domain Control Units.

This paper investigates whether Domain Control Units are the answer to these challenges in E/E architecture design. To this purpose, the relevant aspects in defining domains, the use of implementation design patterns and the requirements in the different market segments are evaluated. As a conclusion, the probability of the introduction of DCUs in the different domains is stated.

E/E ARCHITECTURE DESIGN

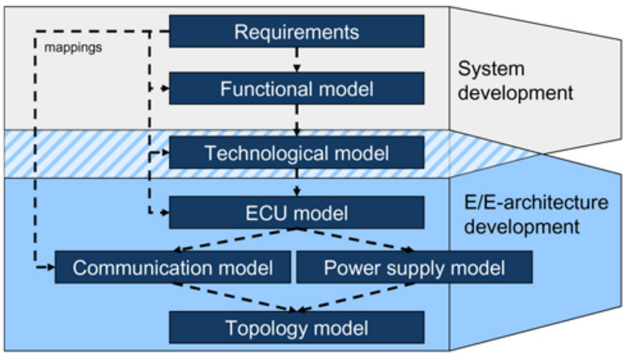

Figure 1. Models of E/E architecture

The E/E architecture design process is a systematic approach for the layout of vehicle communication and power supply networks. It covers the functional and non-functional requirements for the implementation. Figure 1 shows all layers needed in an E/E architecture design [1]. The main parts in this layer model are system development and E/E architecture development. The models for system development comprise the requirements definition and the required building blocks to describe the functional and technological model. These building blocks describe the functional chains which are necessary to realize the required features. The models of E/E architecture reflect the results of different integration stages of electronic systems in the vehicle.

The main benefit of applying the E/E architecture design process is the separation of the functional architecture (logical level) from the actual implementation in the vehicle components. With this separation it is possible to set up a functional architecture for complex functions in the design phase without the restraints of implementation.

As in many cases new vehicle platforms are derived from existing architectures, two approaches for architecture design are in use. The bottom-up approach is based on an existing E/E architecture whereas the top-down approach follows all the model steps starting with the complete set of functional and non-functional requirements. In the bottom-up approach the E/E architecture is derived by making use of existing models and extending them by adding the new functional and network aspects. In the top-down approach the functional complexity is in the main focus and is typically chosen in developing an E/E architecture for new vehicle platforms.

VEHICLE DOMAINS

Whereas in the mid seventies in some cars the only electrical components were the fan of the heating system and the car radio, today's vehicles comprise up to 70 electrical / electronic control units (ECU). These ECUs together with their associated sensors and electro-mechanical actuators will be referred to as components in this paper. The components fulfill tasks related to the control of the combustion engine, the braking and steering as well as driver comfort, driver information and driver assistance. In many cases, the development of functions requires an interaction between different components to realize the targeted functionality. Therefore it is reasonable to define domains in which the different functions can be grouped and worked on as a quasi independent sub system. The benefit in working in these domains arises in the synergies (e.g. development effort, integration effort) which can be gained by establishing a homogeneous field of work, i.e. appropriate grouping of sub systems.

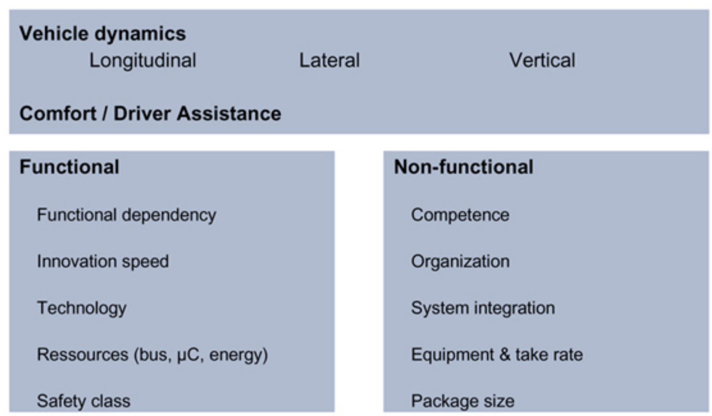

Figure 2. Classification criteria for definition of vehicle domains

As shown in Figure 2, classifying the domains only in terms of physics (vehicle dynamics) and other functions (comfort and driver assistance) would lead to only two very large inhomogeneous domains in terms of required technology and competence. Therefore according functional as well as non-functional criteria should be applied in order to define domains which are as large as possible without loosing possible synergies.

Most important functional criteria are:

• Functional dependency: clustering of all networked functions to limit the number of interfaces / dependencies between domains

• Innovation speed: the change rate / speed of functionality might lead to a temporary separation of a sub domain in order to handle these functions as an independent sub system

• Technology: clustering of functions which require specific technology in the implementation

Most important non-functional requirements are:

• Competence: clustering of the required technical engineering competence

• Organization: organization of the company

Due to the fact that companies are often organized based on the defined domains, non-functional requirements such as the existing, historically grown organization (OEM and / or supplier) strongly influence the possibility to adapt actual domain definitions. However, the domains on the logical level can be looked at as independent of the organizational setup. The following main domains find wide application in the E/E architecture:

• Body & Cabin (driver comfort and lighting)

• Infotainment (displays, driver entertainment and information systems)

• Vehicle Motion & Safety (chassis, active / passive safety and driver assistance functions)

• Powertrain (propulsion and exhaust gas treatment)

IMPLEMENTATION DESIGN PATTERNS

In the defined domains different design patterns for the implementation of functions can be found. In the E/E-architecture design process the functional building blocks are assigned to components in the node model after deciding about the technology in the technological model. The combination of ECU and communication model defines the basic implementation design patterns in the E/E architecture. Figure 3 shows the existing design pattern representing different integration levels:

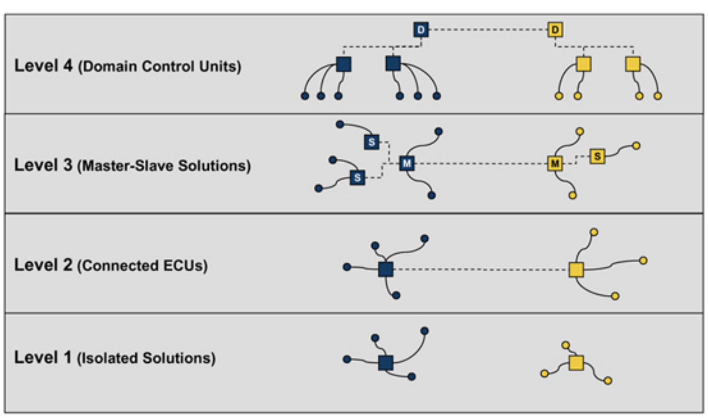

Figure 3. Basic design patterns in E/E architecture

• Level 1: Isolated electronic control units

• Level 2: Connected electronic control units - bus systems connect the ECUs for data exchange

• Level 3: Master/Slave networks - one ECU acts as master and synchronizes the so called slaves in executing their dedicated tasks

• Level 4: Domain Control Units (DCU) - a load free DCU absorbs the software functionality

Due to the fact that E/E architecture design is heavily cost driven, the introduction of DCU as defined in Figure 3 is often questioned regarding its benefit in terms of system costs. In existing E/E architectures the term DCU is sometimes used for different integrated ECU solutions. To distinguish between a purely cost driven integration of ECUs and a DCU concept it is important to reflect the different design patterns with respect to the logical level in the E/E architecture design. Figure 4 shows the basic building blocks necessary to realize a function in a vehicle. The functional software calculates the set point command (e.g. torque demand). The control software generates the according control values (e.g. current per second).

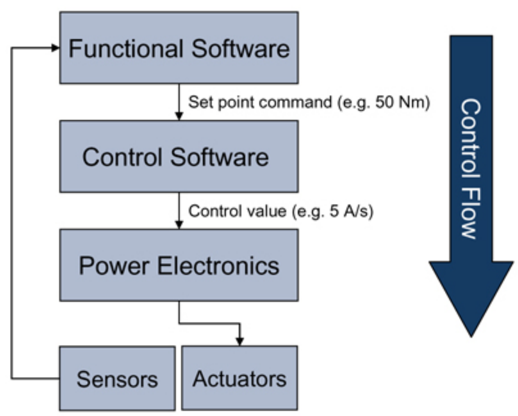

Figure 4. Basic building blocks in the E/E architecture design

The dependencies between the vehicle functions which have to be implemented determine the choice of the appropriate implementation design pattern. As Figure 5 indicates, Level 1 is preferable for isolated functions. Level 2 is used if the different ECUs exchange information (e.g. sensor signals) or if the software functions interact to close control loops in the functional software. With increasing functional dependency, i.e. the need to exchange data to close control loops, the application of Level 2 is limited by the complexity in the data handling between the functional software parts allocated to different ECUs. Therefore, with increasing data exchange and number of involved functions, an integration of functionality on Level 3 or 4 is reasonable in terms of performance and

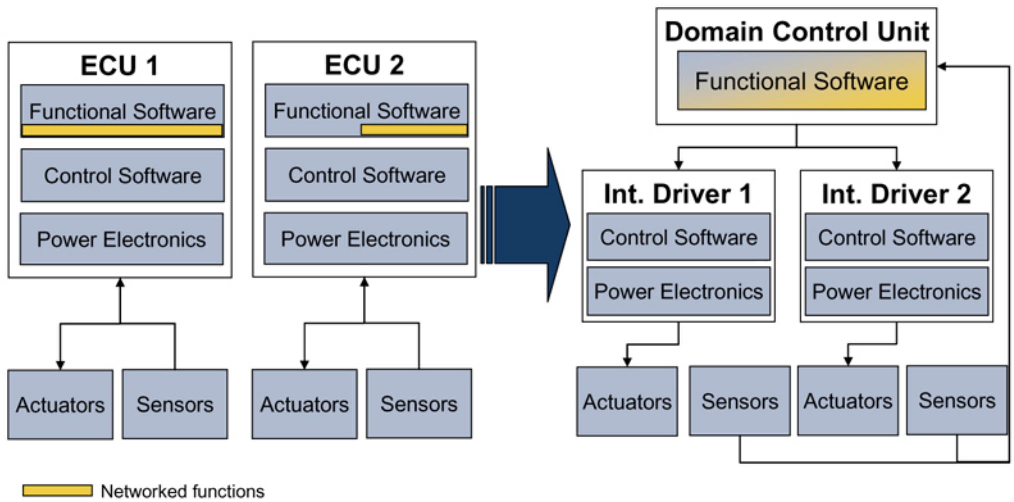

Figure 6. Domain Control Unit as integration platform for Level 2 ECUs

development effort (e.g. synchronization of development, system integration and validation).

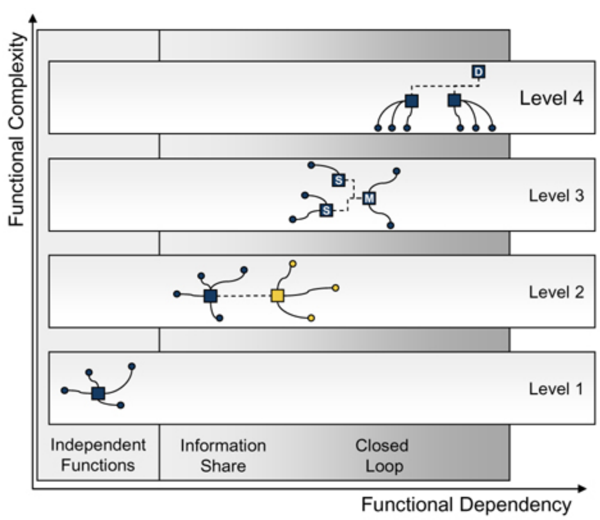

Figure 5. Application of design patterns

The main difference between Level 3 and 4 is the clear separation between the functional software, the control software and the hardware drivers. In the master/slave concept, only the ECU independent software functions are integrated in the master ECU, leaving part of the functional software allocated to the slave ECUs. In the Domain Control Unit the main functional software is integrated, while relegating the basic functions of actuator control to the connected intelligent drivers as shown in Figure 6.

<figure 6 here>

The benefit of introducing DCUs shows in the variant management of components used in different market segments (low price, volume and premium) and additional freedom in developing advanced networked functions. In this case the DCU is the focal point of changes. If the functional range of a component can be covered by the functional software w/o changes in the hardware, the variance can be handled completely in the DCU. In these cases the intelligent drivers remain invariant and can be used independent of the changes in the functional software, i.e. DCU. This leads to a cost benefit by re-using the same components for different vehicle types with different functional range. Furthermore the integration of the functional software in a DCU reduces the effort in developing advanced networked functions (e.g. synchronization of development, system integration and validation effort).

In order to decide about the introduction of a DCU and its software scope, the following guiding questions related to the decision box depicted in Figure 7 can be used:

• Functional Dependency: Are networked functions for e.g. vehicle motion management allocated to different ECUs?

• Functional Range: Depends the functional range for different market segments or functional options only on software, or is different hardware required?

• Innovation Driver: Is the functional innovation driven by software (e.g. driver assistance functions)?

• Equipment Rate: Will the functionality be implemented as standard or optional equipment?

• Coupling SW - HW: Is the functional software strongly coupled to the control software / hardware? Does the system response time prohibit a separation of functional and control software (e.g. due to latency times)?

• Restraints: Are there other functional or non-functional restraints to separate the functional and control software?

Functional Dependency	High	Low
Functional Range	SW only	SW & HW
Innovation Driver	SW	SW & HW
Equipment Rate	High	Low
Coupling SW - HW	De-coupled	Strongly Coupled
Restraints (e.g. Safety Requirements)	Low	High

⇩

DCU

Figure 7. Decision box for introduction of DCUs

Depending on the respective domain, the decision about implementing a DCU based on these guiding questions can be different.

On the implementation level the introduction of DCUs is enabled by actual trends in hardware and software technology:

- Software architecture: standardization of software / interfaces in AUTOSAR supports the decoupling of software and hardware
- µC performance: increasing µC performance at reduced cost results in decreasing importance of resource optimized software
- Bus systems: increasing bandwidth of e.g. TTCAN, FlexRay and Ethernet enables the decoupling of sensors / actuators from location of function

The development of a DCU does not automatically lead to an additional, separate control unit. The DCU functionality can also be implemented in an already existing ECU. An additional non-functional requirement for the implementation of DCUs might arise from ISO 26262: the definition of the ASIL for the different vehicle functions can lead to ASIL or multi-ASIL requirements. The decoupling of the functional software in a DCU from the (embedded) control software and the possibility to use multi core-µC supports the realization of these requirements.

DOMAINS - APPLICATION OF IMPLEMENTATION DESIGN PATTERNS

Due to the different technological development, requirements and technological maturity in the vehicle domains a variety of design patterns can be found even in the same vehicle domain.

DESIGN PATTERNS IN ACTUAL VEHICLES

- Body and Cabin: Level 3 design pattern is applied in most vehicles. A central body computer module is used in almost all vehicles driven by the high number of comfort and support functions.
- Infotainment: Level 2 for basic vehicles with low functional range; Level 3 is widely used in vehicles for the medium segment; Level 4 was introduced in some vehicles via the so called Head Unit.
- Vehicle Motion and Safety: Level 2 for basic vehicles with low functional range; Level 3 for vehicles with high functional range.
- Powertrain: Level 3 in most cases, connecting the engine control unit with e.g. the electronic control unit for exhaust gas treatment in the master/slave principle.

FUNCTIONAL TRENDS IN THE DOMAINS

- Body and Cabin: Except the innovation of LED technique for lighting the functionality is very stable. Therefore there is no trend to change from Level 3 to Level 4.
- Infotainment: For vehicles with extended functional range Level 4 already is introduced using an integrated Head Unit. Innovations in the area of Human Machine Interface (HMI) as well as enhanced functionality like internet or incorporation of portable consumer electronics support the spreading of Level 4 applications.
- Vehicle Motion & Safety: The advanced driver assistance functions are a broad and rapid growing field of innovation. Especially the possibility of sensor data fusion, i.e. combination of available data in the vehicle (e.g. gear rate), the vehicle surrounding (e.g. radar, cameras) and external data (navigation, C2X), allows the development of additional functions and simplification of existing functions. The development of behavior models for the vehicle enables the introduction of a so called safety manager (advanced driver assistance) or a motion manager (longitudinal, lateral and vertical dynamics). These managers improve the vehicle system response to the actual driving situation. With these new functions the introduction of Level 4 is reasonable.
- Powertrain: The development of alternative propulsion systems as hybrid, range extender and full electrical vehicle lead to an increasing variance in vehicle configurations. The introduction of a DCU to integrate the common functions would be a consequent step to support the variant management and to reduce the complexity. If the developments in the domain Vehicle Motion & Safety result in the introduction of a motion manager for the vehicle, the

integration of the Powertrain DCU into this manager is favorable.

MARKET SEGMENTS - INFLUENCE ON E/E ARCHITECTURE

The equipment rate and the available options in vehicles designed for the three different market segments (low price, volume and premium) vary significantly. The optimization of the E/E architecture in terms of cost therefore leads to different solutions in the three market segments. Figure 8 shows an architecture decision box addressing the 4 main topics which have to be decided in setting up the E/E architecture.

Vehicle Layer Functions	Centralized	Decentralized
Signal Routing	Central Gateway	Peripheral Gateways (Backbone)
Scalability	Integrated Boxes	Modular Boxes
ECU Partitioning	Function Oriented	Zone Oriented

Figure 8. Architecture decision box

- Vehicle Layer Functions: Decision to integrate the functional software in a DCU vs. allocation to separate ECUs
- Signal Routing: Decision to directly connect the ECUs for data exchange (backbone) vs. usage of a central gateway to connect the domains
- Scalability: Decision about usage of higher level integration vs. usage of isolated ECUs (Level 1)
- ECU partitioning: Decision about functional oriented integration in ECUs vs. zone orientated integration (e.g. front module and rear module)

The architecture decisions regarding scalability and vehicle layer functions are strongly influenced by the functional range which has to be covered and the equipment rate of the functions and the related components in the vehicles. The functional range for different components can vary significantly for the three market segments. E.g. the basic Airbag functionality only requires one sensor, whereas high end Airbag applications require up to 5 sensors and the according software functionality. The equipment rate of the vehicles in the three market segments also depends on the OEM strategy regarding standard and optional equipment. E.g. an antilock braking system is provided as standard equipment in many low price vehicles, whereas electronic stability control is offered as option or only for higher market segment vehicles. Therefore, typically modular boxes are used for optional equipment.

The different requirements in the low price market segment compared to the volume and premium market segment led to the development of different E/E architectures accordingly.

SUMMARY/CONCLUSIONS

The innovation in the different vehicle domains has led to an increase of functional software and networked functions. Due to this development, the application of the E/E architecture design process gains more and more importance. By differentiating between the logical and the implementation level it is possible to set up a functional architecture in the design phase without the restraints of implementation.

The introduction of Domain Control Units (i.e. the integration of the functional software) is a clear trend in the domains and also the market segments (excluding the low price segment). Based on the lead questions (Figure 7), the equipment rate of vehicles and the trends in functional development the following probability for introducing DCUs in the three different market segments [2] can be deduced.

Domain	Market Segment		
	Low Price	Volume	Premium
Body and Cabin	-	-	-
Infotainment	-	high	high
Vehicle Motion & Safety	-	medium	high
Powertrain	-	medium	high

Figure 9. Probability of introduction of DCUs in the three market segments

The introduction of DCUs in the volume segment for the domains Vehicle Motion & Safety and Powertrain depends on the equipment rate of driver assistance functions and the spread of alternative propulsion systems.

The trend for introducing DCUs is supported by:

- The standardization of software architecture (AUTOSAR): separation of the software and electronic hardware is enabled by standardizing the basis software and the software interfaces.
- Microcontroller performance: sufficient performance at reasonable cost will be available with the new microcontroller generations in development.
- SW based innovation: enhanced networked functions e.g. driver information and driver assistance are mainly based on software rather than on hardware. These new freely allocable functions require an integration platform.
- The trend towards standardized vehicle components: standardization of the basic functionality and the related

components in the vehicles leads to the need for an integration platform for additional functionality.

• The trend towards central vehicle management: central management of vehicle functions (e.g. safety manager and motion manager) leads to complexity reduction in the E/E architecture and development.

The main cost benefit of introducing DCUs in vehicles shows in the variant management for different market segments (i.e. re-use of intelligent drivers as defined in Figure 6) and reduced development effort. Integrating the functional software into the DCU furthermore enables an easier development of advanced networked functions. The DCU can either be implemented as a separate control unit or be integrated in an appropriate existing ECU.

The introduction of DCUs therefore is one key enabler to master the increasing complexity in electrical / electronic systems in vehicles at reasonable cost in the future.

REFERENCES

1. Powolny, S., Friedrich, T. Mischo, St., Kornhaas R. et al., “Model Based Top Down Process for Automotive E/E-Architecture Development,” SAE Technical Paper 2008-01-0284, 2008.

2. SAE Presentation: K. Williams: Vehicle Networks at the Crossroads - How to Cope with Worldwide Requirements?; April 2009

CONTACT INFORMATION

Wolfgang Stolz
Robert Bosch GmbH
wolfgang.stolz@de.bosch.com
Phone: +49 (0) 7062 - 911 - 6057

DEFINITIONS/ABBREVIATIONS

ASIL
Automotive Safety Integrity Level

DCU
Domain Control Unit

E/E
Electric/Electronic

ECU
Electronic Control Unit

HW
Hardware

OEM
Original Equipment Manufacturer

SW
Software

The Engineering Meetings Board has approved this paper for publication. It has successfully completed SAE's peer review process under the supervision of the session organizer. This process requires a minimum of three (3) reviews by industry experts.

ISSN 0148-7191

doi:10.4271/2010-01-0686

SAE Customer Service:
Tel: 877-606-7323 (inside USA and Canada)
Tel: 724-776-4970 (outside USA)
Fax: 724-776-0790
Email: CustomerService@sae.org
SAE Web Address: http://www.sae.org
Printed in USA

SAE International™