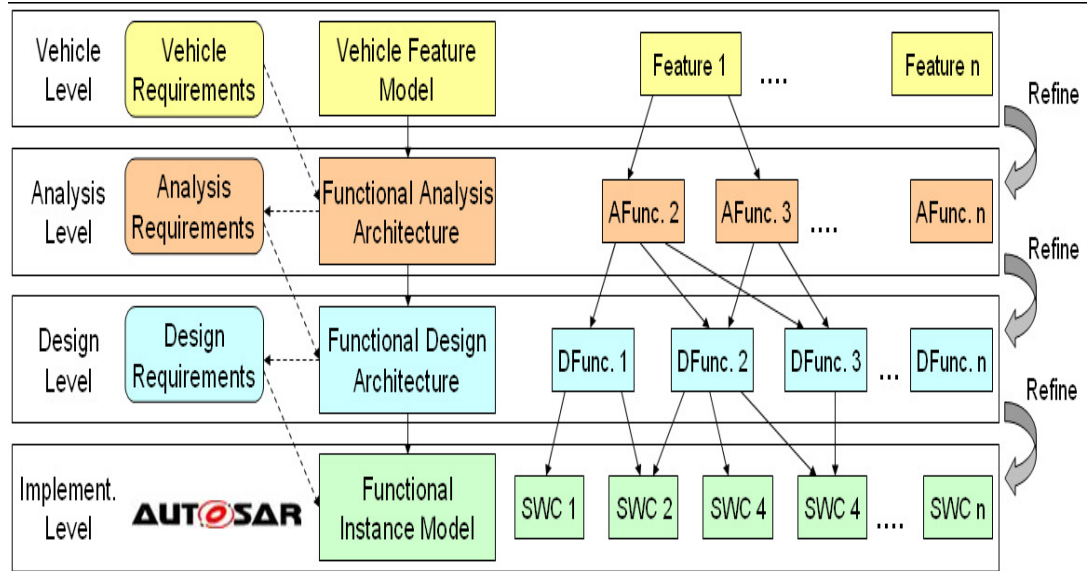


Top-Down Design of Distributed Embedded Systems in Light of Timing Considerations



Michael Seibt
Mentor Graphics Corporation

Proper safeguarding of safety-critical systems in an automotive environment cannot be ensured sufficiently without taking timing into consideration. The failure to observe timing constraints can lead to malfunctions and, in a worst-case scenario, can cause vehicle damage and personal injury. AUTOSAR 4.0 now supports timing constraints, but the standard, although very powerful, still is not able to address all aspects and requirements for electric/electronic (E/E) architectural design. However, alternative standards, such as EastADL2 and the Timing Extension (TIMMO) standard, have tackled this issue. By combining AUTOSAR with EastADL2 and the TIMMO timing language (TADL), it is possible to support a consistent, top-down design approach at both the functional and timing levels.

Architectural Modeling with East-ADL2 and AUTOSAR

The Electronic Architecture and Software Tools, Architecture Description Language (East-ADL) was originally developed within the East-EEA European Project. As part of the ATTEST project, an additional East-ADL2 release was prepared, which includes recommendations for integrating AUTOSAR.

East-ADL2 defines different levels of obstruction that build upon one another.

1. The vehicle (feature) level. At the uppermost level, the focus is primarily on handling the different variants and on modeling customer functions (features). This should make it possible to identify which different functionalities must be in place for a specific variant of the system. In this way, using this artifact offers support for a product line.

The setup of the vehicle view that is illustrated in Figure 1 was selected to allow the possibility of integrating characteristics taken over from another variant. Examples of features include seat heating, adaptive cruise control (ACC), or windshield wipers. Variants would then be, for instance, windshield wipers with or without a rain sensor.

2. The analysis level. The functionalities defined at the feature level and their dependencies are described in detail on the analysis level. This results in an “m-to-n” relationship between the vehicle view and the analysis architecture. Using this description, structural analysis is performed on the resulting functional networks. Duplicate functions can be identified and optimized. Furthermore, it is possible to depict functional behavior, which can be verified through simulation.
3. The design level. The design architecture is a refinement of the purely functional analysis architecture. In the end, it describes the hardware and software partitioning on the basis of the hardware architecture.

4. The implementation level (implementation architecture). The implementation level is derived from the design architecture. This level represents the “flat” software architecture which can be mapped onto the AUTOSAR abstraction level. Depending on the level of detail included in the implementation architecture, the mapping can be performed on the AUTOSAR software component level or on the runnable level.

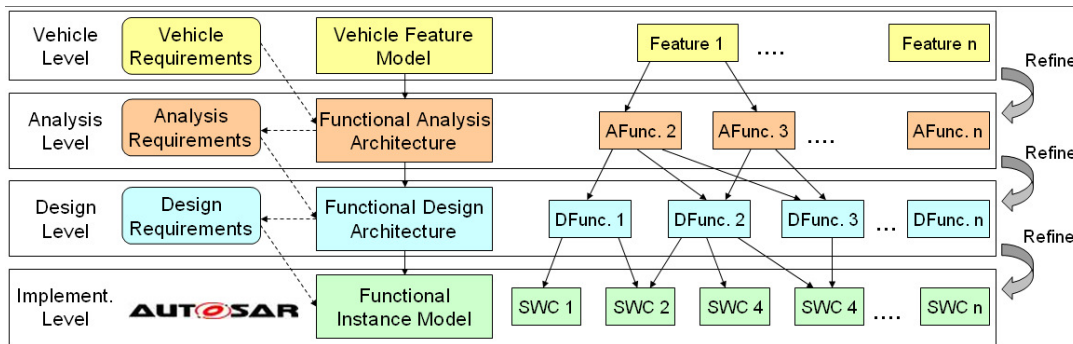


Fig. 1. Abstraction levels in East-ADL2

Requirements (including both functional and non-functional requirements, such as timing constraints) are defined at all levels. These requirements are then also described in detail from level to level.

Description and Representation of Timing Aspects

Within the framework of the ITEA project “TIMMO” (TIMing MOdel), East-ADL2 was expanded to include a timing model that is integrated into the relevant abstraction levels. The different functional levels defined in East-ADL2 use the ADLFunctionTypes or ADLFunctionPrototypes, which use ADLFlowPorts to exchange information with other functions. For input ports, events can be defined with different trigger policies.

Using the functional network descriptions from the different abstraction levels defined in East-ADL2 results in signal chains (event chains) with different degrees of granularity. On the basis of these signal chains, the Timing Augment Description Language (TADL) defined in TIMMO can be used to define timing constraints for any segment.

The following timing constraint concepts are supported:

1. End-to-end delay (age). A “DelayRequirement” defines a maximum permissible delay time for data between a transmitter and a receiver. For example, the availability of an up-to-date brake signal is indispensable following a minimum period for an object acquired via the radar of an ACC system.

2. Synchronization constraints. In certain application scenarios, signals have to be synchronous with one another. For example, in order to be able to compare the speed signals acquired from the four wheels in a meaningful way, the signals must have the same date/time at the brake regulator input. In the same way, the brake actuator signals must work in synchronization in order to ensure “correct” braking.
3. Event constraints. Depending on the application, events must be of a periodic, sporadic nature or follow a predefined pattern. The repetition rate (for a periodic event) defines the time of receipt for data at a port or the triggering period of ADLFunctions. What is the minimum frequency, for example, at which a sensor signal must be scanned in order for the signal to be acquired reliably?

Here, we will use adaptive cruise control (ACC) as an example in order to offer a more detailed explanation of how timing aspects are described in TIMMO.

In principle, ACC basically functions as follows:

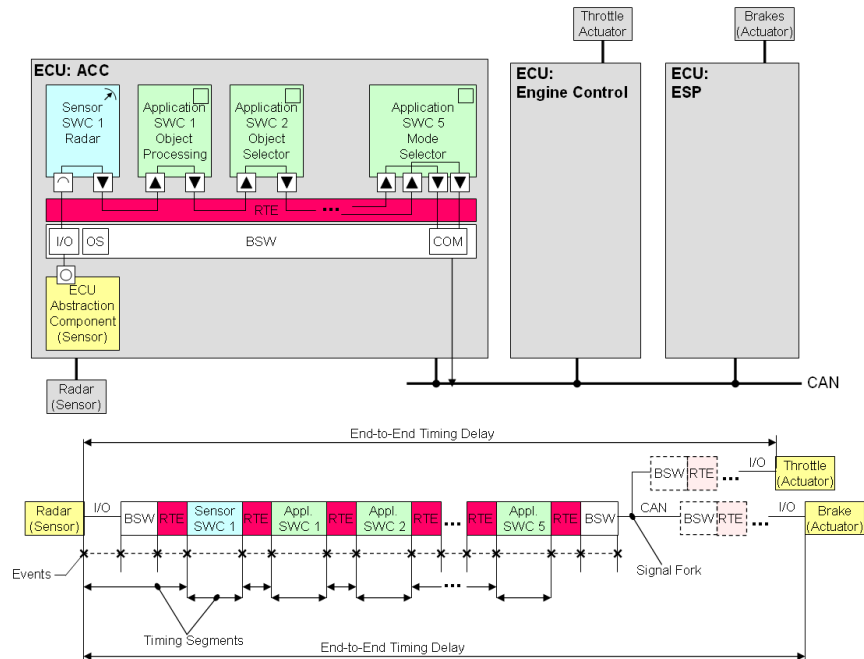
When ACC is activated, the vehicle’s speed is regulated in such a way that it remains at a pre-defined level. In the event that an obstacle arises, e.g. in the form of another vehicle driving ahead of the vehicle that is being controlled, the obstacle is detected. When the distance drops below a certain minimum interval, the vehicle’s speed is reduced. The speed can be reduced through systematic braking or through throttle control. Once the obstacle is no longer in the way, the vehicle automatically accelerates to the pre-defined speed.

Speed and distance are determined using tire sensors and radar. The unit controlling the speed (cruise controller) and the unit controlling the distance (follower controller) use these parameters to calculate a delay or acceleration as a variable. This variable then serves as the input variable for the throttle control and the brake system.

The illustration in Fig. 2 shows a section of an ACC system at the AUTOSAR RTE level, which is mapped onto the EAST-ADL2 implementation level. The signal generated by the radar sensor goes through the different layers of hardware and software. Each transition to a new layer generates an event. Two neighboring events then form a timing segment. Because the signal paths branch off, tree-like event chains arise. Relevant time constraints can then be assigned to these event chains.

Relevant timing constraints for this ACC example could be the end-to-end timing delay between object acquisition at the radar sensor and the brake actuator, or the synchronicity of the speed signals acquired at the wheels.

Fig. 2.
Section from
an ACC system



“Frontloaded” Design made Possible by Timing Model

It is the availability of a parameterized timing model that makes it possible to determine timing requirements at an early stage of development as well as analyze and optimize the timing behavior of distributed embedded systems. These principles have been implemented, for example, in the Mentor Graphics vehicle network architect (VNA) design tool, which supports two operational modes:

1. Analysis mode. On the basis of an existing communications matrix, deadline monotonic analysis (DMA) can be used to calculate worst-case signal times and bus loads for CAN and/or LIN networks.
2. Synthesis mode. In this mode, the input is specified as timing requirements in the form of maximum permissible signal latencies and the maximum permissible bus load. Using this information, an optimization algorithm is used to generate the signal-to-PDU and PDU-to-frame packing automatically. Scheduling tables for gateways are also generated automatically. This leads to a valid network configuration in accordance with the defined requirements.

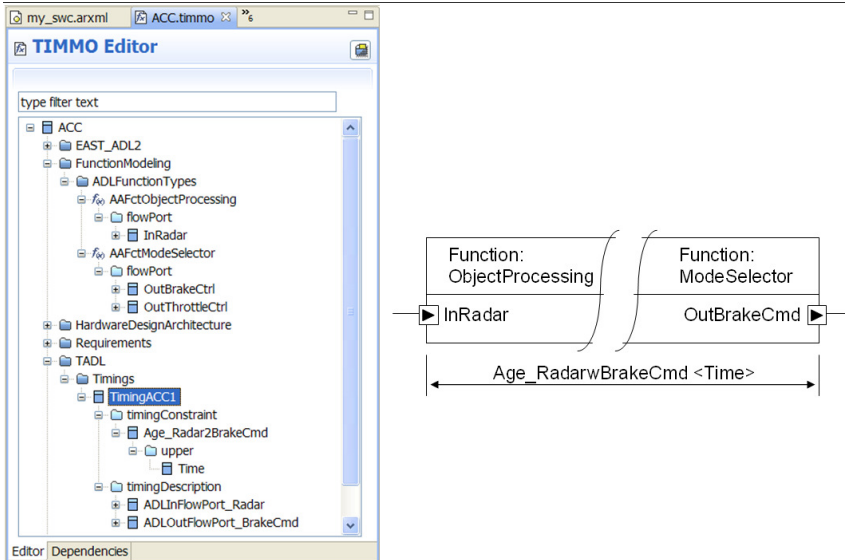


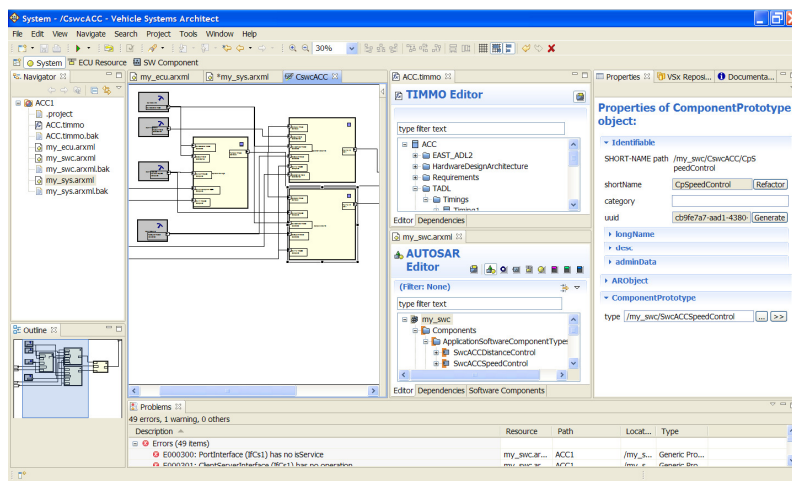
Fig. 3. Definition of an end-to-end timing constraint in the VSA TIMMO editor.

Tool Support with Volcano VSA

The Volcano Vehicle Systems Architect (VSA) architecture tool supports both AUTOSAR and the TIMMO/EastADL2 metamodel. Consequently, it is possible to seamlessly describe a logical vehicle architecture including the timing aspects on all of the required abstraction levels. Supported by “built-in” or user-defined constraints, the design can be verified for accuracy and completeness at every stage of development. Existing domain-specific graphic editors also support “occasional” users at the various stages of architectural development and in the AUTOSAR software configuration.

VSA is built upon Eclipse technology and has been on the market since 2009.

Fig. 4. Volcano Vehicle Systems Architect (VSA).



Top-Down Design of Distributed Embedded Systems in Light of Timing Considerations

By calculating metrics such as performance, resource utilization and cost, in the future it should be possible to evaluate architectural variants. The existing interface for wiring-harness development tools, such the Mentor Graphics CHS suite, builds a bridge to the physical architectural design.

Conclusion

On the basis of the TIMMO metamodel, E/E architectures can be described at the different layers of abstraction all the way up to the implementation level with regard to their function and timing. As a result, there is also a seamless transition to the AUTOSAR methodology, and thus to the software configuration. In this way, both top-down and bottom-up design are supported.

Ultimately, complete verification of system/subsystem functions can only be accomplished if timing aspects are taken into consideration. These aspects often make the difference between a system functioning or malfunctioning. For this reason, using a front-loaded design process can save time and money in two ways: It makes it possible to reduce the need for expensive hardware tests, and it makes it possible to increase the quality and reliability of the final product, the “vehicle.”

After all, time is money.

For more information, call us or visit www.mentor.com/systemvision

Copyright © 2010 Mentor Graphics Corporation. This document contains information that is proprietary to Mentor Graphics Corporation and may be duplicated in whole or in part by the original recipient for internal business purposed only, provided that this entire notice appears in all copies. In accepting this document, the recipient agrees to make every reasonable effort to prevent the unauthorized use of this information.

Corporate Headquarters Mentor Graphics Corporation

8005 S.W. Boeckman Road
Wilsonville, Oregon 97070 USA
Phone: 503-685-7000

North American Support Center

Phone: 800-547-4303
Fax: 800-684-1795

Silicon Valley Mentor Graphics Corporation

1001 Ridder Park Drive
San Jose, California 95131 USA
Phone: 408-436-1500
Fax: 408-436-1501

Europe Mentor Graphics Deutschland GmbH

Arnulfstrasse 201
80634 Munich
Germany
Phone: +49.89.57096.0
Fax: +49.89.57096.400

Japan Mentor Graphics Japan Co., Ltd.

Gotenyama Hills
7-35, Kita-Shinagawa 4-chome
Shinagawa-Ku, Tokyo 140
Japan
Phone: 81-3-5488-3030
Fax: 81-3-5488-3031

Pacific Rim Mentor Graphics Taiwan

Room 1603, 16F,
International Trade Building
No. 333, Section 1, Keelung Road
Taipei, Taiwan, ROC
Phone: 886-2-27576020
Fax: 886-2-27576027