European vehicle manufacturers are today presented with big challenges and opportunities. Past growth in sales has been achieved to a large extent by innovations in comfort, with functions like automatic climate control and navigation; safety, such as, for example, vehicle stability control, multiple air-bag systems, and seat-belt control; and environmental protection, including direct fuel injection and catalytic converter control.

These innovations have been made possible by the use of embedded electronics. Today, a car has several tens of computers, communicating through data buses in a complex, distributed, in-vehicle electronic architecture. Future functions will be even more complex, distributed, interconnected and necessarily interdependent. Their correct behaviour will not simply be a matter of functional correctness, that is, making sure that the results of the computations are correct, but they will depend on timing and reliability constraints. Building such systems in an efficient, predictable and reliable way in spite of the increased complexity of functions and architectures and managing the supply chain in a way that allows predictable integration of software components and platforms is the future major challenge of the automotive industrial sector.

Many systems today are time-critical or at least time-dependent. The effects of improper timing range from a loss of comfort to life-threatening situations. Vehicle stability control, involving differential braking of individual wheels is an example of a function in which safety depends on the timely delivery of the braking commands. Precise timing and prioritisation of functions are essential for both safety and comfort.

Today, timing is mostly taken into consideration late in the development process, during the implementation and integration phase. Timing behaviour is verified by means of measurements at testing time, rather than through formal and systematic analysis. The likely consequences are long and costly design iterations whenever problems are detected. For this reason, a considerable number of innovative functionalities cannot be implemented in a cost-effective manner, and may therefore not be realised. A predictable development process able to handle timing in all phases and capable of verifying and validating the timing behaviour of a real-time system early in the process is a key factor in bringing new innovative features to market and in handling their implementation complexity.
The European research project TIMMO (TIMing MOdel) is after a breakthrough in the area of automotive system timing management using a common, standardised approach for handling all timing-related information during the development process. The complexity—and the cost—of the development cycle is reduced significantly, while reliability is improved. TIMMO is about developing a Timing Augmented Description Language (TADL) and an accompanying methodology that provide

- a formal and standardised specification, analysis and verification of timing constraints across all development phases, avoiding over- or under-dimensioned systems and unnecessary iterations in the development process;
- a formal and standardised specification, analysis and verification of timing constraints at all levels of abstraction enabling, e.g., timing requirements to be traced across all abstraction levels;
- an improved and predictable development cycle enabling a common, standardised infrastructure for handling timing to shorten the development cycle and increase its predictability.

These are fundamental prerequisites to avoid costly delays in vehicle start-of-production dates and, in turn, to assure confidence in the dependability and quality of a given solution.

In developing the timing concepts, industrial and academic partners of the TIMMO project have drawn on their extensive experience with automotive systems. TIMMO concepts are indeed specific to the automotive domain, complementing the automotive standards AUTOSAR and EAST-ADL. The developed language and methodology leverage state-of-the-art results from academic research without sacrificing applicability to practical systems. The main beneficiaries of the TIMMO results are the vehicle manufactures and their suppliers that can use the new language and methodology for a well-defined exchange of timing information in automotive embedded-system development.
In summary, the TIMMO project team has provided highly relevant timing concepts fully aligned with the most influencing standard in the automotive domain – AUTOSAR. Accordingly, we believe the TIMMO concepts will soon become part of the industrial automotive state-of-the-practice development.

Prof. Alberto Sangiovanni Vincentelli
University of California at Berkeley

Prof. Marco Di Natale
Scuola Superiore Sant'Anna
01 TIMING MODEL – MASTERING IN-VEHICLE TIMING CONSTRAINTS

01.1 MOTIVATION

Electronics and software are playing an ever growing role in modern cars. They are a key driver for innovations in the automotive industry. Since so many functions in cars are implemented by means of electronics and software, timing synchronisation between different sub-systems and -functions has become crucial – especially for safety-relevant functions like braking or stability control. Therefore, it is becoming increasingly important to deal with the timing constraints of a system throughout the entire development process.

Today, timing is mostly dealt with rather late in the development process, during integration and testing. Designers first focus on the functional behaviour of the system under design, and then investigate the timing behaviour. This is mostly addressed by means of measuring and testing, rather than through formal and systematic analysis. The consequence is that errors related to timing are often detected too late, which leads to costly re-iterations in the development process.

01.2 GOALS

The main goal of TIMMO is to develop a formal, standardised approach to describing timing-related information in embedded-system design for the automotive industry. It shall provide means for formal specification, analysis and verification of timing constraints at different levels of abstraction as well as throughout the entire development process.

Such a common infrastructure for handling timing will enable an early analysis of whether a system can meet the desired timing requirements, and avoid over- or under-dimensionalised systems and unnecessary iterations in the development process. The result is a shortened development cycle with increased predictability.
01.3 EXPECTED RESULTS

TIMMO will have three major results: A formal language for modelling timing aspects, an accompanying methodology that describes how to apply the language in the development process, and a set of case studies serving as example applications and as validators for the language and methodology.

01.4 ORGANIZATION

Establishing a well-accepted, common language requires major industrial players in the field as well as leading European research institutes to participate. Therefore, the TIMMO consortium consists of vehicle manufactures, system suppliers, tool vendors, and research institutes from Austria, France, Germany, The Netherlands, and Sweden. It combines leading industrial partners in Europe both on the manufacturers' and suppliers' side as well as advanced research in the field.

The project started in April 2007 and ends in September 2009. The schedule and the distribution of work packages (see Figure 1) reflect the main desired results. After the definition of the project scope in WP 1, there are three main technical work packages which develop the language, methodology, and validators respectively.

Figure 1: The TIMMO time schedule.
01.5 BROCHURE OVERVIEW

The following Chapters 02 and 03 give a detailed presentation of the Timing Augmented Description Language (TADL) and the methodology developed in TIMMO. Chapter 04 describes the validation activities and the different case studies developed in order to demonstrate the applicability of the TIMMO results in practice. Chapter 05 gives an overview of related approaches and the context TIMMO is embedded in. Finally, in Chapter 06 a summary and conclusion is given.
02 TIMING AUGMENTED DESCRIPTION LANGUAGE

The definition of the Timing Augmented Description Language (TADL) is based on modelling concepts from EAST-ADL (http://www.atesst.org) and AUTOSAR (http://www.autosar.org), by which the structural definition of the considered system is modelled. The augmentation is done by adding information related to timing and events referring to structural elements. Further inputs for the definition are the requirements and the methodology as defined within the TIMMO project, and the parameters used by the analysis tools from the TIMMO partners. MARTE (Modelling and Analysis of Real-Time Embedded systems, http://www.omgmarte.org/) from OMG also serves as an input to TADL.

The semantics describes the meaning of the TADL concepts and a foundation is given for, e.g., black-box parameters, end-to-end delay constraints, and paths.

The TADL is formally defined as a meta-model, which references the AUTOSAR metamodel. This is supplemented with structural concepts from EAST-ADL, and all added concepts conform to the AUTOSAR meta-modelling rules. In this way, the exchange format of a timing model is defined and an AUTOSAR editor is extended to allow for EAST-ADL and TADL modelling.

02.1 EVENTS AND EVENT CHAINS

On the higher abstraction levels (vehicle, analysis, and design) structural modelling is performed as defined within EAST-ADL. On the implementation level AUTOSAR modelling is performed. The TADL constraints are defined separately from the structural modelling and refer to structural elements via Event Chains and Events, see Figure 2. The events are tailored to refer to specific elements in the structural model, i.e. different sets of events are available for the EAST-ADL model and the AUTOSAR model.
Figure 2: This is a simplified extract of the UML meta-model, which shows the relationships between the elements available for modelling of constraints, event chains, events and structural elements.

The constraints as defined by TADL have a constraint kind as specified by MARTE allowing for the distinction between whether a constraint is offered, required, or a contract. Additionally all TADL constraints carry a timing bound and a mode. The requirement support in EAST-ADL allows for tracing from solutions as modelled in the structural model to requirements, and from verification cases to requirements. The TADL constraints fit in the requirement support as refinements of the requirements. Further support in EAST-ADL includes the tracing between abstraction levels by realization for the structural elements and derivation between requirements.
Figure 3: A snapshot of the VSA editor where an extract of an EAST-ADL model is seen. A Functional Analysis Architecture with two functions builds up the structural model. A Reaction Constraint with bound attributes has been defined; this refers to an Event Chain built up by an in event (EventIn) and an out event (EventOut) referring to structural ports.

Figure 4: Conceptual diagram of the design modelled in the tree view seen in the previous figure.
02.2 AGE, REACTION, AND LATENCY TIMING CONSTRAINTS

Multirate systems, which are common in automotive applications, have over- and undersampling effects, which an end-to-end delay semantics has to consider. Very often register based communication is used where message loss (undersampling) or duplication (oversampling) occurs. This can be seen in the figure below. TIMMO has discovered that there are, in general, four semantics to distinguish for end-to-end delay constraints. Further details on the semantics can be obtained in [10].

![Figure 5: Illustration of the four end-to-end semantics; screenshot taken from the Scheduling Analysis tool SymTA/S.](image)

From these theoretical cases TIMMO has identified two end-to-end delay semantics, which are interesting for real world communication delay. These delays are the reaction delay (a, first to first) and age delay (d, last to last).
The reaction delay is utilized if only the first reaction to a signal is of importance. This is usually the case in body electronics, e.g. a light can only be switched on, and a second activation signal will not influence the status of the light any further.

The age of data is important in control engineering. Here responding several times on same data decreases control quality.

The delay from any of these semantics can be composed of segments throughout the complete system and might span multiple ECUs and buses. This is part of the time budgeting and decomposition as described in the TIMMO Methodology. To support this on a timing language basis TIMMO uses the black box concept. This can be seen as an interface for timing delay composition. A black box contains all the information required to bridge the gap at the edge between e.g. a bus and an ECU.

Given these semantics and the corresponding semantical TADL elements it is possible to describe and verify end-to-end delay constraints throughout the complete supplier and car manufacturer chain. Furthermore there is the possibility to explicitly define the kind of delay, which will reduce misunderstandings even before implementation or design stage.

### 02.3 INPUT AND OUTPUT SYNCHRONIZATION CONSTRAINTS

Constraining timing is not always about delays between stimuli and responses. An important class of constraints deals with the synchronization between either stimuli or responses, respectively. One example is a central lock function. The ReactionConstraint between button pressed (stimuli) and door locked (response) could typically have a span between fastest and slowest reaction of several hundreds of milliseconds. However, the tolerated difference between when the different doors are locked is perhaps just some tenths of milliseconds. This difference between a number of responses related to the same stimuli, is expressed in TADL by OutputSynchronizationConstraint. Figure 6 depicts a central lock system where the locks of the four doors are controlled by the remote control in the key. The locking of one door is defined as an event each. The OutputSynchronizationConstraint, constrains the maximum difference in time between any of these four events. Note that this value can be much less than the tolerated jitter in the ReactionConstraint between the stimulus event defined as pressing the key and any of these four response events defined as locking of a door.
In a similar way, there is also a constraint called InputSynchronizationConstraint in the TADL. This is to constrain the maximum difference between a number of stimuli events, e.g. sampling of sensors.

Figure 6: Central lock system.
03 TIMING MODEL METHODOLOGY

03.1 MODELING APPROACH FOR THE TIMMO METHODOLOGY

The TIMMO Methodology has to meet the challenge of finding acceptance in a world of already established or currently emerging automotive development models and standards. This applies to the actual development steps, which are the building blocks of the methodology, as also to the way the methodology is modelled and specified.

Due to the particularly close relationship between TIMMO and AUTOSAR, the Software Process Engineering Meta-model (SPEM), which is the underlying standard for the development of the AUTOSAR Methodology, was selected as the meta-model for the development of the TIMMO Methodology, too. SPEM has been introduced by the Object Management Group (OMG) for modelling arbitrary software and system development processes. It is based on the widely accepted UML (in the form of a UML profile) but is better suited for the domain of software and system process engineering than the pure, more generic, UML.

Reusability and adaptability of the methodology are inherent benefits of this approach. SPEM encourages a clear separation of the “method content” from the definitions of either reusable capability patterns or full delivery processes.

In a first step, the method content is described by basic elements like tasks, work products, roles, and their interrelations. The basic elements can be enriched by associations with tools suitable for performing tasks or with additional guidance like templates, guidelines, or checklists. In the second step, tasks, associated input and output work products, and assigned roles, are used to create more complex, reusable activity patterns or even full processes. The end user of the methodology may either reuse selected patterns to integrate them in already established or new development processes, or may use a predefined full process directly or after performing some specific adaptations.

The predefined process in the TIMMO Methodology is conceived as one possible example of a timing-oriented development process. This process example is based on EAST-ADL with regard to the interpretation of the higher abstraction levels (vehicle,
analysis and design level) and on AUTOSAR with regard to the implementation level including VFB, system and ECU views. Furthermore, the process is inspired by the V-Model development approach.

It is easily understood that a sophisticated underlying meta-model like SPEM cannot be applied thoroughly without tool support. For this purpose, TIMMO makes use of the Eclipse Process Framework (EPF) Composer, which is the Open Source version of the IBM Rational® Method Composer. The EPF Composer allows modelling the SPEM method content, capability patterns, and delivery processes as described above. The resulting model of the TIMMO Methodology is available as an XML export library that is complemented by a hypertext documentation accessible by common web browsers.

03.2 RELATION OF INPUT- AND OUTPUT WORK PRODUCTS

The relationship between input and output work products is modelled in the context of tasks. A task specifies both optional and mandatory input work products, as well as output work products (always mandatory). However, it is not possible to model optional outputs of a task, and it is not always obvious from the task description exactly which input work products are used as source for an individual output work product.

To overcome this deficit the TIMMO methodology adds a textual description in form of a table to the main description of each task. This table is used to specify the above mentioned missing information for each output work product which contains timing information.

Consider for example the task “Create design requirements” which is shown in the Figure 7. This task produces as output the work product “Design timing requirements”. The table in the main description of the task states that “Design timing requirements” always contains end-to-end delay requirements (mandatory) and that it may also contain synchronisation requirements (optional). The table also specifies which timing information in which input work products are sources for the output. In case of the end-to-end delay in the Design timing requirements, the end-to-end delay contained in the work product “Analysis timing requirements” is used as input.
Figure 7: Creation of design requirements.

Note that the main description of a task contains an individual table for each output work product. This table may of course be empty if the specific output contains no timing information.

### 3.3 INTEGRATION OF TOOLS

The TIMMO methodology guides deriving a concrete implementation from the requirements. While refining different aspects of the model the correct timing behaviour has to be validated. Depending on the modelling phase different possibilities to validate the model can be applied. These possibilities include a variety of techniques like testing, simulation, or formal verification. To support the engineer the TIMMO methodology provides recommendations on when a set of tools, like simulators, can be applied; what data are required to make use of the mentioned validation method and what the possible results are. At some points in the methodology a specific tool is proposed because it is difficult to establish input and output information in general to a complete set of similar tools.

An example is given below in order to convey the idea behind refining timing information and requirements respectively: An end-to-end delay constraint is broken down into "finer/smaller" time constraints across the different levels of abstraction.

On vehicle level an end-to-end delay constraint is introduced that has been derived from a given requirement. This requirement has to be valid for the surrounding physical
systems. The possibilities to derive such a constraint from physics or the environment are very complex and there is no general solution available. One has to use specific tools and methods fitting to the requirement at hand. Later at system level this constraint is split into sub-constraints and contract between companies. End-to-end delay sub-constraints will usually span a complete ECU, bus or gateway. This can be represented using the black box semantics provided by the TADL. Validation at this level can only be done using formal verification methods that work on these formal models. Only at later stages during system level integration these formal verifications can be supplemented using other validation methods like Hardware-in-the-Loop (HIL) tests. At implementation level one can even use simulation, testing and formal methods to validate that each task does not exceed its execution time constraint.

03.4 SEAMLESS METHODOLOGY

Today in automotive development processes conformance to timing requirements is difficult to ensure, which is mainly due to the increasing size and complexity of modern systems, and the concurrent design between car manufacturer and suppliers involving dozens of parallel activities that need to be coordinated. In addition timing validation is becoming more and more difficult by the fact that system performance is not easily composable, due to the increasing interdependencies between functions within the vehicle. By and large, the system integrator—typically the car manufacturer—cannot automatically conclude that the integrated system satisfies the given timing constraints from the fact that all supplied components comply with the requirements imposed on them individually.

The seamless TIMMO methodology describes a way of how timing can be systematically controlled and validated during all design stages and across all collaborating design teams. Thereby, the basic concept consists of iteratively exchanging timing information between car manufacturer and suppliers. Such timing information includes the temporal and functional behaviour of sub-systems and their components, as well as between sub-systems and components. This information exchange is the basic prerequisite to address timing issues in automotive systems, and enables to consecutively refine timing requirements evolving from the earlier development phases (vehicle and analysis phase) down to the more concrete phases concerned with the design and implementation of
actual components and subsystems (implementation phase). The TADL has been specifically developed for this purpose.

This basic principle enables to understand the timing behaviour of the developed system much better and is the basis for the second basic concept included in the TIMMO methodology: the breaking down of timing requirements (a.k.a. time budgets). This divide-and-conquer approach allows managing the complexity of timing validation and directly supports business processes which are playing an important role in the collaboration between car manufacturers and suppliers. It consists of breaking down timing requirements in a system into appropriate timing requirements for different involved suppliers.

03.5 METHODOLOGY AND TADL

As mentioned in the previous sections, one of the major results of TIMMO is the Timing Augmented Description Language TADL that is used to express timing information, like timing requirements and properties.

Another important result is the TIMMO methodology which describes how the TADL language is utilized in the context of collaboration: for example in the cooperation between vehicle manufacturers and suppliers, and concurrent engineering, for example at the same time several suppliers develop components for a specific sub-subsystem.

The methodology is based on the development processes introduced by the EAST-ADL and the AUTOSAR Methodology. Based on the abstraction levels of EAST-ADL—which are Vehicle, Analysis, Design and Implementation Level—and the different AUTOSAR views—which are VFB, System and ECU view—the methodology describes the tasks that are carried out in order to process the timing information on each level of abstraction and view respectively. One of the major ideas behind the methodology is to enable the time budgeting using the capability to specify Event, Event Chain and Event Chain Segments respectively. This capability of the TADL enables to subsequently refine time budgets across different level of abstractions and therefore in different phases of the development process.

For example, on the highest level of abstraction, the Vehicle Level, the timing requirement is given that specifies an end-to-end delay between the point in time when it is detected that the driver presses the brake pedal and the point in time when the brake
actuators at the four wheels of a passenger car are taking into effect. The value of this timing requirement is given as 550ms and the accepted tolerance is +3ms and -2ms. This means that the brake actuators could take into effect 2ms earlier or 3ms later than expected. In the following development step, the Analysis Phase, one concludes that three functions are required to realize the braking (decelerating) functionality: 1) the sensor that monitors the position of the brake pedal; 2) the brake controller that processes the information about the brake pedal's position along with other information, like vehicle state; and 3) the actuators that decelerate the vehicle by applying the brake force to the wheels. At this level the given end-to-end delay is subdivided into three time budgets for the sensor, the controller, and the actuators.

On the next level of abstraction, the Design Level, possibly three different suppliers concurrently are developing the three components during the Design Phase: the sensor, the controller, and the actuators. Each of the suppliers follow their own development process where the given time budgets are continuously broken into smaller time budgets appropriate for the particular design and structure. On this level, artefacts containing timing information must be exchanged between the involved parties in order to ensure a proper design of the entire sub-system, for example ensuring that the frequencies at which data being exchanged between the components are consistent. Both, the TIMMO methodology and the TADL making sure that this collaboration is supported.

On the Implementation Level, the functional design is transformed into software and hardware domain and the time budgets are further refined and assigned/mapped to software parts like Runnable Entities and hardware parts like Bus Latencies, etc. Eventually, each component is implemented and the binary code constituting the component’s function is executed on the target system. At this point in time the timing properties of each component are determined and could be verified against the given requirements. For example, the Runnable Entity dealing with the monitoring of the brake pedal's position responses within 12ms to a request to capture the brake pedal's position and the given timing requirement states that this Runnable Entity shall respond within 15ms to a request to capture the brake pedal’s position.
The main results of the TIMMO project are evaluated by five different validators in order to show a broad coverage of the TIMMO approach in the context of the AUTOSAR standard. The main objective of the validation is to prove the applicability of the TIMMO methodology as well as of the individual elements of the Timing Augmented Description Language (TADL).

Each validator implements a real-world application with real-time constraints and other functional and non-functional requirements and covers different aspects of the TIMMO methodology and the TADL. Since TIMMO partners with different roles (1-tier suppliers, car manufacturers, and tool vendors) jointly participate in the development and elaboration of the validators, the validators provide an adequate testbed for the validation of the TIMMO methodology. Different hardware topologies and software configuration are implemented to support the specification of different timing constraints by the TADL.

The five different validators with their contributors are:

Validator 1: Anti-Lock Braking System  
Contributor: Volvo Technology, Mentor Graphics

Validator 2: Steer-by-Wire and Active Damping  
Contributors: TTTech, University of Paderborn, Symtavision

Validator 3: Engine Control  
Contributors: Bosch, ETAS, Symtavision

Validator 4: Transmission Control  
Contributors: ZF Friedrichshafen AG, Symtavision

Validator 5: Cruise Control and Security System  
Contributors: Continental Automotive

In parallel, TIMMO concepts are additionally validated by Denso with a vehicle dynamics control application.
The next pages give a comprehensive overview of the five validators and introduce more details of the applied hardware and tools as well as the individual software configurations.

04.1 VALIDATOR 1: ANTI-LOCK BRAKING SYSTEM

The system is a distributed braking application with ABS, implemented with a single wheel, brake and sensors. It is designed from scratch, covering all phases of the TIMMO methodology and TADL, including all abstraction levels defined in EAST-ADL. Timing requirements, constraints and properties are given extra focus during the design.

§ External timing requirements
   (Ex: ergonomy and regulations)
§ Internal timing requirements
   (Ex: for control performance)

Modeling languages
   (TADL + EAST-ADL/AUTOSAR)
Development methodology
   (TIMMO methodology)

System engineering tools
   (Ex: Papyrus, VSA)
Control engineering tools
   (Ex: Simulink / Embedded Coder)
Verification tools
   (Ex: timing analysis tools)
AUTOSAR Tools
   (Ex: authoring tool / BSW SW)

ABS

ECUs
   (Ex: MPC5517G)
Communication technology
   (Ex: FlexRay)
Peripheral devices
   (Ex: sensor + actuators)

Figure 8: Components of validator 1.

Figure 8 gives an overview of the scope of Validator 1. Using TADL and the TIMMO methodology, focus is on showing a) how timing requirements on the system are decomposed and handled, b) how tools are used to support the development, and c) how design choices made during the development that affect the end results.

The focus of the validation is on timing, including external timing requirements, such as response times for brake commands (e.g., time from pressed brake pedal to brake actuator reaction), or internal timing requirements such as timing requirements imposed by the control algorithms used (e.g., periodicity and sequences of events, execution times of software) or constraints on the infrastructure (e.g., CPU load or schedulability).

The system is structurally designed using EAST-ADL and AUTOSAR. Timing requirements, properties and constraints are formally defined using TADL within the same model (VSA is used for TADL modelling). Models serve both, as documentation of the engineering process and as input for various engineering tasks (e.g., analysing...
response time and schedulability using SymTA/S). The model is finally transformed (AUTOSAR software generated from Simulink) into an AUTOSAR software architecture executing on a distributed prototype system, consisting of two ECUs connected to a common FlexRay bus. Peripheral devices are attached using local IO to the ECUs. Final verification is made on the prototype, i.e., ensuring that timing requirements are met.

04.2 VALIDATOR 2: STEER-BY-WIRE AND ACTIVE DAMPING

Validator 2 implements a speed-dependent steer-by-wire system including active damping. The validator hardware is composed of two main components: an active steering wheel from Stirling Dynamics and an in-house setup of a steering and active damping testbed. The system comes with four components: steering wheel, speed sensing, steering axle with a tire and active damping. In this configuration, the speed sensing is only realized as a virtual sensor. The other three components are currently linked via three individual CAN bus segments to their Electronic Control Units (ECUs). All four ECUs communicate via a FlexRay communication bus as shown in Figure 9. The combined FlexRay-CAN cluster is especially dedicated to the investigation of TADL event chains over different ECUs and different bus segments.

For ECUs, Universal Control Units from TTTech are applied which implement a high-performance, stacked-board expansion system. For the configuration of the FlexRay schedule and the ECUs, TTXPlan and TTXBuild are used in combination with a target
compiler and a Lauterbach debugger. In this context TTXBuild provides generated C-code compliant to the AUTOSAR communication stack.

As such the system realizes the validation of event chains and execution times specified with and exchanged by TADL and the AUTOSAR standard between different tool sets for simulation, timing verification, and the hardware configuration (see Figure 10). This basically applies to parts of the design, implementation and test phase of the TIMMO methodology.

![Figure 10: Exchange of timing constraints.](image)

For simulation, CAN and FlexRay SystemC libraries from the University of Paderborn are taken where the latter can be configured via FIBEX for time-based simulation. Timing verification is executed by SymTA/S from SyntaVision for budgeting, scheduling verification, and optimization of processors and networks. The combination and exchange of information between the three tool sets allows a detailed simulation, analysis, and verification of the event chains, end-to-end delays, and execution times. It further investigates the transformation of AUTOSAR and TADL timing information between tool sets which are applied during different steps of the development process.

### 04.3 VALIDATOR 3: ENGINE CONTROL

The Engine Control Application validator is realized as a single-ECU system with an AUTOSAR-compliant ECU software architecture. The Engine Control Application software is developed with ASCET from ETAS according to the AUTOSAR methodology and implemented on a microcontroller with an Infineon TriCore1796 processor. As platform software, the AUTOSAR-compliant real-time operating system (RTOS) RTA-OSEK and the AUTOSAR Runtime Environment from ETAS are used. The
microcontroller with the Engine Control Application is coupled to a simulation environment via a CAN bus.

The simulation environment executes a model which simulates the combustion processes in an actual engine. For this, the model processes the computed ignition times and injection times from the Engine Control Application and provides inputs such as the current accelerator pedal position set by the driver to the Engine Control Application.

The application contains several stringent timing requirements which must be maintained during runtime. Various control loops must be continuously processed in parallel (e.g., control loop for idle throttle control, control loop for lambda control, etc.) while computing the necessary fuel mass and ignition time for the combustion processes. RPM-dependent ignition and injection imposes another important reactive timing requirement on the implementation.
To provide the required real-time properties in the implementation, a RTOS which provides the required real-time system scheduling mechanisms is used. For validation and verification purposes, the RTOS is fully instrumented to obtain meaningful execution time values during the run of simulated drive cycles. The execution time values are used to perform system-level scheduling analysis with the SymTA/S tool suite from Symtavision to verify system-level timing and compute the maximum microcontroller utilization for the timing-critical corner cases.

04.4 VALIDATOR 4: TRANSMISSION CONTROL

Validator 4 is built up with an evaluation board—its processor is typically used in series productions of transmission control systems—for a simplified driving strategy and a corresponding testbed. The testbed consists of a PC, capable of computing all necessary model parts in real-time and equipped with a CAN connection card. Furthermore a connector box is used to adapt the PC connectors with that ones of the evaluation board, to display the current gear and the proposed one, to shift the gear manually and to influence the CPU or CAN load on the evaluation board using manually different potentiometer.

Regarding TIMMO methodology the validator has the focus on the relationship between system supplier and basic software supplier. Nowadays the basic software is more and
more delivered by third-party and must be regarded as black-box software because of intellectual property (IP) for example. Due to this situation it is important to get experience in handling of this kind of deliverables in all development steps knowing resource properties (e.g. timing).

Therefore the validator will investigate an end-to-end delay scenario which starts with a manual gear shift event (connector box) and ends by sending a CAN message including the proposed gear from the driving strategy to the transmission control. This considered signal path will go twice through different software layers delivered by third-party and will support the focus of this validator.

The validator shall also demonstrate the possibilities of TIMMO methodology to provide the development process for additional requirements to system resources (e.g. increasing CAN or CPU load) for an existing system.

Regarding TADL the validator will deliver time stamps of task and function calls via a power trace debugger. These time stamps will be analysed and used to create a system model in consideration above described end-to-end delay scenario using the scheduling analysis tool SymTA/S. This model with its timing information and constraints shall show the practical applicability of the TADL.

04.5 VALIDATOR 5: CRUISE CONTROL AND SECURITY SYSTEM

Due to the wide range of the domains of automotive software, the validator comprises two applications, a closed loop application (cruise control) from the powertrain domain, and a more user oriented state based application, realized by a security system of a car.

The applications are modelled and/or given in AUTOSAR and EAST-ADL (UML2-Profile). The goal is to use TADL in conjunction with these models and define timing constraints with TADL. The model shall give an example of how to apply TADL on the different abstraction levels of EAST-ADL and how and to which extend, it can be used for validation and verification of TADL Timing Constraints. The application exists as a model and as runtime components (Figure 13).

The cruise control regulates the vehicle speed according to the desired speed set by the driver. In the given scenario, various inputs, such as engine speed, vehicle speed, desired speed set by the driver, and driving conditions, like uphill and downhill, are considered and appropriate actions have to take place within a defined time frame. This
leads to timing requirements (end to end delays) between sensors (e.g., speed) and actuators (e.g., rpm).

The security system controls and triggers alarms in case of an unauthorized access of the car. It is possible to arm different zones of a car in defined situations. In the armed state, unauthorized access triggers an alarm. The security system is mainly state-based and has various timing requirements attached to its state transitions.

Figure 13: Model, components of cruise control/security system.
TIMMO was inspired by several other projects and efforts. High-level methodology findings were re-used from ATESST, modelling of abstract real-time properties was partially adopted from MARTE, and the lower implementation levels were aligned with AUTOSAR. The TIMMO project covers the entire picture of timing in the design of automotive systems: from high- to low-level, from abstract to detailed, and in line with industry-established standards.

Furthermore, a special link has been established between TIMMO and the EU-funded research project ALL-TIMES. ALL-TIMES aims at building curriculum in the use of timing analysis technology and in using combinations of different technologies. The following timing tool categories from ALL-TIMES have also been included into the analysis-related steps in the TIMMO methodology:

- WCET tools analyze the object code of a software function, consider the hardware platform, and provide worst-case execution times.
- Tracing and measurements help visualizing the system timing during tests.
- Scheduling analysis performs corner case analysis of system load, response times and end-to-end latencies.

In one or the other way, all these techniques have also been used in the TIMMO validators. WCET analysis, tracing and measurements were used for collecting detailed timing information. The scheduling analysis tool suite SymTA/S has been used in the validators for system optimizations in earlier design stages, and for system verification in later stages. This has shown that the major TIMMO results—TADL and Methodology—are directly usable in practice with and supported by existing analysis tools.

TIMMO also gives back, especially to ALL-TIMES and to the AUTOSAR Timing Subgroup. Both groups report that the experience of the TIMMO project provides valuable input, especially for the methodological aspects in both projects.
Innovations in the automotive sector require new development approaches in order to cope with the mental complexity of large and decentralised control systems. In modern vehicles both, existing and new functions must be integrated into a common electronic architecture such as comfort functions (e.g., park assist, climate control, navigation), safety functions (e.g., stability control, airbag, belt pretensioner), and functions for protecting the environment (e.g., direct fuel injection, catalytic optimisation).

Over the last years there have been several approaches to manage complexity challenges of new automotive electronic systems. AUTOSAR is probably the most well known initiative which had its roots in the EUREKA ITEA project EAST-EEA and which is constantly further developed by industry. AUTOSAR can be seen as the basis for a common in-vehicle software infrastructure and is already found in series production cars. Regarding comprehensive architecture description features, the EAST-ADL language which dates back to the EU FP6 project EAST-EEA and which has been enhanced in the project ATESST (and is currently further enhanced in ATESST2), offers viable means for the specification of in-vehicle electronic architectures.

As of today there is, however, lack of support for handling timing aspect throughout the whole development process of complex electronic automotive systems. Timing is often deemed relevant in late development phases only such as the implementation and integration phases. In addition, timing is often seen in context of measurements and tests and only infrequently used with formal and systematic analysis. As the effort for the development of innovative automotive functions steadily increases, new functions can often not be implemented in a cost-efficient way. Furthermore, the lack of systematic timing analysis potentially leads to defects in the field (along with expensive repairs and callbacks) which are related to timing problems.

TIMMO tackles existing challenges of end-to-end timing modelling of complex in-vehicle systems. The major achievements of TIMMO are on the one hand a formal, XML based language—the so-called Timing Augmented Description Language (TADL)—which is used for system wide timing specification of automotive functions, and on the other hand a systematic development methodology which guarantees practical exploitation of the TADL within (real-world) automotive development phases, i.e., which allows specification
of temporal requirements over different stages of development. Both, TADL and the according development methodology of TIMMO are strongly based on AUTOSAR and EAST-ADL with the ambition to enhance the existing concepts within these initiatives.

The strong positioning of the TIMMO project is guaranteed by a consortium of renowned European automotive companies, system suppliers, tool vendors, and research institutions, including: Audi Electronics Venture, CEA-LIST, Chalmers University of Technology, Continental Automotive, Denso Europe, ETAS, Mentor Graphics, Bosch, Siemens IT Solutions and Services, Symtavision, TTTech, Universität Paderborn, Volkswagen, Volvo Technology, and ZF Friedrichshafen.
07 SELECTED TIMMO PUBLICATIONS

07.1 MAGAZINE ARTICLES


07.2 CONFERENCE AND WORKSHOP CONTRIBUTIONS


