Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Masterseminarausarbeitung WS2011-2012

Florian Bartols

Cluster Simulation of Real-time Ethernet based Electronic Control Units in Context of Distributed Automotive Applications

Fakultät Technik und Informatik
Department Informatik

Florian Bartols

Thema der Masterseminarausarbeitung
WS2011-2012
Restbussimulation von Echtzeit Ethernet-basierten Steuergeräten im Automobil

Stichworte
Echtzeit Ethernet, TTEthernet, Restbussimulation, Bussysteme, Automotive Anwendungen, Fibex

Kurzzusammenfassung
Die zunehmende Komplexität automobiler Netzwerke und deren hohe Bandbreitenanforderungen erfordert neue Konzepte in der Vernetzung. TTEthernet ist ein geeigneter Kandidat diese Problemstellung zu lösen. Um Frühzeitig im Entwicklungsprozess Steuergeräte zu testen hat sich die Restbussimulation als adäquates Mittel etabliert und wird im Projekt für das TTEthernet-Protokoll erarbeitet. Diese Arbeit gibt einen Ausblick auf meine Masterarbeit und stellt die dazugehörigen Arbeitspakete und die verbliebenen Risiken vor.

Title of the paper
Cluster Simulation of Real-time Ethernet based Electronic Control Units in Context of Automotive Applications

Keywords
Real-time Ethernet, TTEthernet, Cluster Simulation, Bussystems, Automotive Applications, Fibex

Abstract
The increasing complexity of automotive networks and their high bandwidth requirements will demand new concepts in networking. TTEthernet is a suitable candidate to solve this problem. In order to allow testing of control units in an early point in the development process, cluster simulation has been established as an adequate manner and therefore it is developed for the TTEthernet protocol in this project. This paper gives an outlook of my upcoming Master's thesis and presents the work packages and the remaining risks.

Contents

1 Introduction

Today's cars are complex, mixed critical [1] distributed systems with more than 70 single components called electronic control units (ECUs) [2]. These units are responsible for either information and entertainment systems e.g. radio navigation or safety relevant functions such as x-by-wire or adaptive cruise control systems and therefore the requirements for these systems differ in bandwidth, reliable data transmission and real-time behavior. Safety critical functions have high demands in real-time characteristics, whereas entertainment systems require high bandwidth capabilities. The different requirements resulted in an inhomogeneous infrastructure where for every kind of application class a single system for on-board networks has been utilized. Therefore, the required on-board network topology is getting more and more complex and is nowadays nearly impossible to handle [3]. Also next-generation driver assistance functions like camera or radar based driver assistance functions won't lead into a solution of this problem. On the contrary: These applications have both high real-time requirements and high demands in bandwidth. State-of-the-art network systems like CAN [4] and FlexRay [5] operate already on their bandwidth limits or do not meet the real-time requirements (Most). Thus, they cannot be applied in the field of these systems. A possible solution for the addressed problems is the application of Ethernet-based networks in vehicles [6, 7, 8]. TTEthernet [9] is a suitable candidate for future on-board networks to overcome the issues imposed by next-generation driver assistance. It provides a reliable data transmission with real-time characteristics and also high bandwidth. The complexity of networks can also be reduced, due to the ability of dedicated message transmission in different classes, which would result in one network system only.

The main key driver for new innovations in the automotive industry is the use of software controlled applications. The utilization has increased in a few years from 20 percent to 80 percent and forecasts claim that 90 percent of the automotive functions will be software in this decade [10]. This fact has a direct influence on the costs during the development process of a new automobile. To minimize these costs, the software has to be tested in an early phase. The functionality spread over several units makes this goal even more difficult, since the development of new cars is also more and more distributed [11]. The Original-Equipment-Manufacturer (OEM) designs the car and its features in specifications while external suppliers produce only parts. Therefore, the suppliers have to test their developed ECUs entirely on their own, prior to the first assembly of all components. Missing nodes that are responsible for a foreign but essential part of the function can make this testing impossible. For this purpose, a tool called cluster simulation has to be applied by the suppliers. A cluster simulator imitates the missing nodes and their behavior inside the network and therefore enables a solution for testing purposes. It has already proven to be utilized inside the development processes for state-of-the-art technologies, since many tools have been developed [12]. The tool CANoe [13] by Vector Informatik was successfully used for the development of an electric car, which shows that this kind of testing is also ready for future cars and applications.

Since TTEthernet is a new technology and research on application in in-car networks has just started, for the best of our knowledge there are no cluster simulators basing on this protocol implemented yet. Therefore, this topic is interesting and challenging as well. Principles that are already in use in state-of-the-art-simulators can be applied and have to be extended to meet the requirements for TTEthernet. Furthermore, to achieve a close to the standard configuration function, a common description language for in-car networks have to be adapted for TTEthernet.

This paper gives an outlook of my Master's thesis. It determines the focus of my upcoming thesis and also presents the main work packages and discusses some of the given problem statements.

This paper is outlined as followed: The next section 2 on the following page will state out and determine the goals and focus of my Master's thesis. The problem statements and the working packages are presented in section 3 on page 6. In section 4 on page 10 remaining risks are to be discussed and finally, section 5 on page 11 concludes and gives an outlook.

2 Thesis Goal and Fundamentals of Cluster Simulation

First of all, the goal of the presented Master's thesis have to be stated out. Therefore, this part describes in a short manner the main components, which are discussed in detail in the next chapter 3 on page 6. Thereafter, fundamentals and examples in application are declared.

2.1 Goal of the Thesis

The objective is to enable a cluster simulation for the real-time Ethernet protocol TTEthernet. The simulator will be implemented on a standard hard- and software platform. At the moment it is planned to use a Linux-based x86 machine to run the simulation. To achieve the required real-time characteristics, the RT-Kernel Patch by Ingo Molnar [14, 15, 16] will be used. In comparison with the standard Linux Kernel, it provides an advanced interrupt handling and enables priority driven scheduling even in Kernelspace. This allows tasks running in Userspace to preempt Kernelspace processes, which is in general not the case in standard Linux. Furthermore, when hardware interrupts any running process, the assigned interrupt service routine (ISR) handling is outsourced in a Kernel task. This task has a predefined priority and therefore is preemptible for other processes. As a consequence, the ISR just preempts all running processes to start a Kernel task, which is much faster than handling all hardware driver routines.

The advantage of using a standard hard- and software platform is the ability to utilize libraries and components, which are widely deployed. This reduces the costs and risks on development significantly.

It should also be possible to use standard configuration files, that are already in use to describe an in-car network topology. The Field Bus Data Exchange Format (Fibex) [17], that is specified by the Association of Standardisation of Automation and Measuring Systems (ASAM e.V. [18]), is the common description language for current state-of-the-art in-car networks. Therefore, the specification has to be extended for the TTEthernet protocol.

Since a cluster simulation of a complete network is very resource intensive, can influence real-time behavior [19] and maybe not needed for testing purposes of one ECU, a strategy for complexity reduction on relevant functions only has to be applied on the configuration.

2.2 Fundamentals of Cluster Simulation

This section gives a retrospective view on previous work [20, 12] and presents the essentials of a cluster simulation and gives some examples of testing approaches. Additionally in the beginning, it determines the technology from the close related hardware-in-the-loop simulation.

2.2.1 Determination of HiL- and Cluster Simulation

Cluster simulation is not the only opportunity to test ECUs during the developments process. Another approach is hardware-in-the-loop (HiL) simulation [21]. Since these types of testing tools are very close related, a clear determination is hard to achieve. In advanced development stage the lines between those two type are often blurry, because some features are used both in cluster and HiL simulation. In general, they are existing side bye side and therefore, this section explains the approaches.

Cluster Simulation A cluster simulator is connected to the device-under-test (DUT) via the communication interface and triggers the DUT only with regular data frames. The simulator is only responsible for generating the missing frames. As a consequence it is only possible to verify the reaction of the ECU on the logical level. A comparison of the desired and actual value has be achieved by analyzing the received data.

Hardware-in-the-Loop Simulation Shall the reaction of the ECU on the environment also be monitored, the testing tool has to be connected to the sensors and actuators of the ECU, too. The DUT is connected to the HiL simulator via the communication interface and an environment interface. Events of the environment are either simulated too, or represented by real circumstances. This enables also a verification of the actual technical behavior.

Since the network will evolve towards fault tolerance, a modified topology presented in [22, 23] has to be applied. Nodes, called peripheral interface nodes, which are wired to sensors and actuators and only perform read and write operation on the network, and functional nodes, which perform the functional task are introduced. Thus, a cluster simulations will be even more necessary and will be the goal of my Master's thesis.

2.2.2 Differences Between Bus and TTEthernet Based Cluster Simulation

As already stated out, cluster simulation has proven to be an adequate testing tool for bus based ECU development. TTEthernet bases on a switched network architecture so that the media access and thus the visibility of messages differs. Bus based systems are shared media with one shared collision domain, where every peer will receive each transmitted message. The receiving peer decides based on the message-id whether the data is for him to be processed or to be discarded. Through the shared collision domain, bus systems also allows only a sequential data transmission with only one frame allocating the whole network system.

In switched network topologies every single peer has its own congestion domain. The visibility of messages is bounded to this domain and thus the receiver only detects the messages that are explicitly routed to him. This allows a simultaneous data transmission, since one frame cannot allocate the whole network system.

This behavior has to be in mind when test cases are generated by recorded network traffic. To record traffic in switched networks it is not sufficient anymore to attach a recorder to the network and listen on the medium, like in bus systems. The traffic has to be collected in every collision domain and must be merged afterwards to get the correct causality of the traffic.

Another difference between time triggered bus and TTEther-Networks is the synchronization of the local clocks, which is also a result of the shared media access. Whereas automotive bus systems use a dedicated synchronization frame to synchronize the clocks, TTEthernet uses a network transparent synchronization approach. Standard Ethernet frames with customized payload are transmitted.

2.2.3 Scenarios for Utilization

The testing method when using cluster simulation comply to the black-box testing [24], where only the in- and output parameters matter. The tester doesn't need to know how the function on the ECU is implemented and thus, the test cases are based on specifications. Typical test design techniques are for example equivalence partitioning, boundary value analysis and state transition tables.

For ECU testing, it has to be distinguished between open-loop and closed-loop simulation. To test the static ECU behavior, like correct implementation of network protocol dependent features, tests basing on open-loop are sufficient. The generation of test data can be performed offline and are either issued from static attributes of the network description file or recorded real traffic. With these information, tests on general correct functionality can be performed with open-loop simulation, by applying equivalence partitioning and boundary value analysis.

On the other hand, the analysis of dynamic ECU behavior can be only achieved with a closed-loop simulation, that is also known as reactive testing. Hereby the test data is generated online and depends on the behavior model of the simulated nodes. Control loop algorithms can only be tested with reactive testing, since the reaction of the ECU depends on the current state, for instance.

3 Work Packages

This chapter presents the work packages in a more detailed way and will give background information about the Fibex file format and explains the current description format for TTEther-Networks. It will be cleared out, how a suitable TTEthernet extension for Fibex can be applied. The strategy for complexity reduction for relevant functions only will be stated out afterwards. Finally, an architecture overview will be given.

3.1 Description Formats for In-Vehicle Components

In-vehicle networks and their functions are specified and designed via tool utilization by the OEM producer. Since the development is distributed, these information must be available through all participants in the development process to implement the components and configure the used tools. They are used to describe the network topology and the actual behavior of the ECUs. Additionally, they must be readable with machines to enable an easy to use tool configuration, and thus, cost reduction can be achieved, because configuration work doesn't have to be done twice.

The representation of these information is either binary- or text-based. In general, binary configuration files are proprietary solutions, that permit changes with standard editors and therefore make a deployment as a standard difficult. Text-based files can be read and modified with standard text editors, but context is hard to parse with machines without standardized formatting. The Extensible Markup Language (XML) is such a file format, that uses a dedicated formatting for information and is defined by the W3C [25] as an open standard. This enables both machine- and human-readable files [26, 27]. Therefore, XML is also a suitable solution for configuration files inside the automotive industry.

3.1.1 The Field Bus Data Exchange Format

The Field Bus Data Exchange Format (Fibex), formally known as ASAM MCD-2 NET, is an XML-based file format for data exchange between tools and applications that deal with automotive bus communication systems [17]. It was defined as an open standard by the Association for Standardization of Automation and Measuring Systems (ASAM) to replace the proprietary DBC format [28], which has been used by Vector Informatik to describe distributed applications on CAN networks.

The object oriented data model provides dedicated data types for all attributes of in-car communication systems (see figure 3.1). There are entities that describe the system topology, such as cluster, channel or ECU objects. Frames, their timing, or the in-/output-port are entities, which characterize the communication level. Also functions and the transmitted signals are represented. The entities shown in figure 3.1 are general attributes, defined by Fibex. Due to the object oriented design, MOST, LIN, CAN and FlexRay are supported via extensions, that specify the presented standard entities with their characteristic system attributes. The recently published Fibex v. 4.0.0 supports standard Ethernet, which also shows that new technologies can be added, and thus makes it appropriate for future systems.

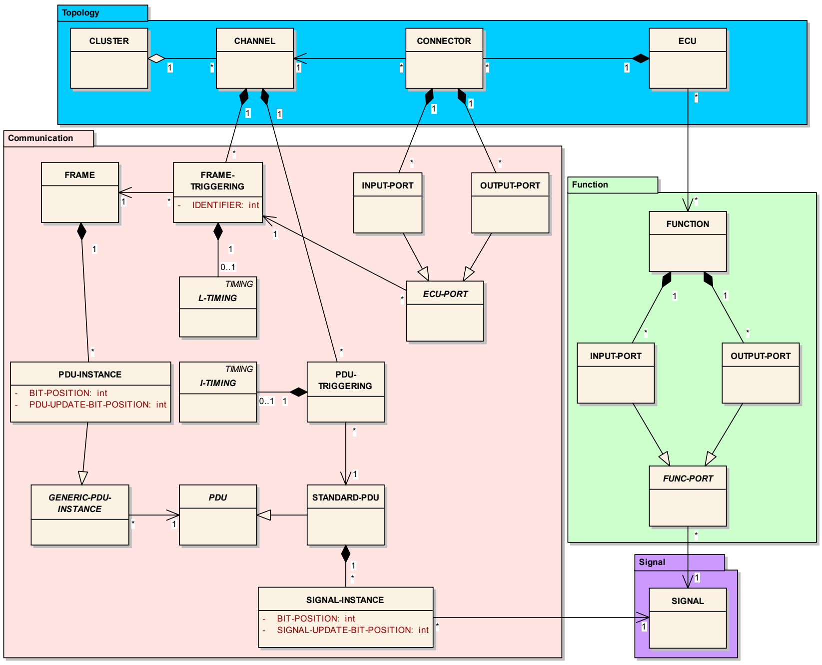

Figure 3.1: *Overview of the entities defined in Fibex and the relationships between them*

3.1.2 TTEthernet Network Description

The configuration of TTEthernet networks is also driven by tool utilization. Therefore, the elaborated network design has to be published to the used tool chain and is achieved by using XML based files, too. It describes topology dependent attributes like the static message routing and virtual links, which represent the logic connection between two nodes. Hence, in TTEthernetworks the routing of critical traffic is configured static to be predictable and achieve real-time behavior. Furthermore, device specific operation attributes like the scheduling of messages and the role in the synchronization process are depicted as well [9, 29].

3.1.3 Extending Fibex for TTEthernet

As declared in section 2 on page 3 the goal is to achieve a configuration of a TTEthernet cluster simulation with standard formats, especially the Fibex format. The differences between Fibex and TTEthernet network description are, semantic descriptions of messages and the actual function of the ECU are not included in the TTEthernet network description. Both are necessary to achieve a complete configuration of a cluster simulator.

This means the Fibex format has to be extended for the TTEthernet protocol. This can be achieved by using the standard Ethernet specification, introduced in version 4.0.0. The principles and data objects for time-triggered transmission of frames can be inherited by the FlexRay extension for Fibex, because of its time-triggered behavior. These have to adopted to meet the requirements of TTEthernet. Furthermore, the role during the synchronization process must be recognisable, too.

Additionally, a transformation function between the Fibex format and the network description would be useful as well. A TTEthernet Network designer tool is currently under development by TTTech, which will export the configuration only as network description format.

3.2 Strategy for Complexity Reduction

A possible solution for the complexity reduction is to work on the topology, virtual links and scheduling entities. The node for which the configuration is built is marked, and all virtual links plus the attached nodes, connected with this node are selected then. Participants, that have no relation to the to be tested node can be removed from the configuration, since they don't have any effects on the cluster simulation. The scheduling of messages have to be adopted afterwards, because the topology during the simulation changes. The relative distance between nodes has a direct connection to the scheduling of messages, since the propagation delay increases with the distance. The adoption can be applied by using the receive point in time of a message at the DUT. This is then the scheduled start of transmission at the cluster simulator without the physical transmission delay of the Ethernet technology.

3.3 Architecture Overview

The last section will depict the current design architecture of the cluster simulator, presented in figure 3.2 on the following page. The set-up is based on the standard for cluster simulation, presented in [13, 30] with a configurator/analyzer, the actual simulator and the device-under-test. The configurator is a standard workstation based component, which builds the executable for the simulator and gets the information via the preliminary discussed Fibex or TTEthernet network description. If only the network description is available, the transformation function has to be used to create a Fibex file. To model the behavior of simulated nodes, a possible solution is to use Matlab generated code. The simulator executes the simulation and logs the results. Afterwards, they are transferred to the configurator again and can be analyzed by the tester.

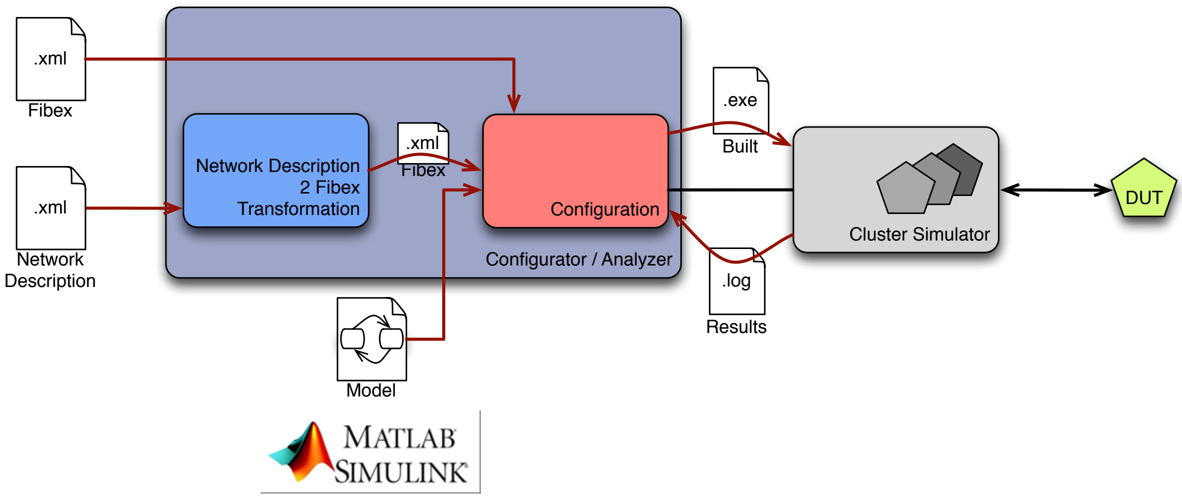

Figure 3.2: *Overview of the Cluster Simulator Architecture*

4 Assessment of Risks

This chapter presents the remaining risks which can arise during the development and implementation of the cluster simulator. Basically, the risks are took soundings from previous work, presented in [20] and [12], but there are still remaining risks, which are manageable.

A possible risk is, the TTEthernet extension for Fibex won't be accepted by the community of automotive industry, because the technology has minor relevance at the moment. Where this is in fact acceptable for my work and research, a bigger problem is the existence of real automotive ECU specification, since these are closed documents in general. The goal is to achieve a complete cluster simulation of a real automotive distributed system. Therefore, it exists a dependency on real specifications.

Another problem could be the performance of the x86, RT-Linux based simulator. In reliance on the topology, the cluster simulator has to imitate a lot of ECUs. Thus, the behavior calculation of the ECUs can influence the real-time characteristics. If the performance is not enough sufficient, deadlines in execution of code or transmission of frames can be missed and an entirely simulation cannot be achieved.

Moreover, there is still no tool available to create and design TTEthernet networks. The network description has to be created and edited manually at the moment, which is in fact very error-prone, due the complex XML scheme.

There are also some solution ideas to overcome the presented risks. To make sure, the Fibex extension will be accepted by all participants, the research group CoRE is currently in contact with TTTech and it is planned to get in touch with ASAM. For another project, the group is in contact with BWM Forschung und Technik (see [31]), where TTEthernet is compared with Ethernet AVB. This relation could be extended to get the required ECU specifications, too.

If the execution performance or the real-time capabilities of RT-Linux are not enough sufficient, the simulation can be adapted to run on an ARM based microcontroller with TTEthernet stack [32, 33], or the simulation will be executed distributed where just a part runs on each simulator. The architecture design allows a variable choice of the building toolchain for different platforms.

5 Conclusion and Outlook

Finally, this section concludes and gives an outlook on future work, especially the project and master thesis.

In this paper, an overview of the topic of my Master's thesis was given. Additionally background information and a determination of hardware-in-the-loop and cluster simulation were presented in the beginning. Both approaches are used in coexistence during the development and testing phase of an ECU. The difference between them is the stimulation and the observation of the reaction of the DUT. While this is only achieved via the communication interface during a cluster simulation, the stimulation via the sensors and actuators is done by using a hardware-in-the-loop simulation. The test scenarios are based on black-box-testing method, where the test cases are generated through specifications, in general.

The work packages were presented afterwards. The configuration of the simulator will be done with the Fibex format, which is a quasi standard. It is a XML-based file, that supports all automotive communication systems as extensions. Therefore, a dedicated extension for TTEthernet has to be developed. Furthermore, a transformation function between the TTEthernet network description, which is based on XML, too, and Fibex will simplify the configuration process. The complexity reduction of the cluster simulation will be achieved considering the connections and message scheduling between the DUT and the simulated nodes. The system architecture consists of a configurator / analyzer, the actual simulator and the DUT.

During project 1, first theoretical and analytical work will be done regarding the Fibex format. The implementation of the presented architecture will be aspired in project 2. A complete cluster simulation of a real distributed automotive application is then performed during the Master's thesis.

Bibliography

[1] W. Steiner and G. Bauer, "Mixed-criticality networks for adaptive systems," in *2010 IEEE/AIAA 29th Digital Avionics Systems Conference (DASC)*, Oct. 2010, pp. 5.A.3–1–5.A.3–10.

[2] B. Müller-Rathgeber, M. Eichhorn, and H.-U. Michel, "A unified Car-IT Communication-Architecture: Network switch design guidelines," in *IEEE International Conference on Vehicular Electronics and Safety*, Sep. 2008, pp. 16–21.

[3] J. Badstübner, "Kollaps im Bordnetz: Schluss mit Can, Lin und Flexray," *KFZ-Betrieb*, no. 17, pp. 68–70, 2008.

[4] Robert Bosch GmbH, "Controller area network." [Online]. Available: http://www.semiconductors.bosch.de/

[5] FlexRay Consortium, "Flexray," Stuttgart. [Online]. Available: http://flexray.com/

[6] T. Steinbach, F. Korf, and T. C. Schmidt, "Real-time Ethernet for Automotive Applications: A Solution for Future In-Car Networks," in *2011 IEEE International Conference on Consumer Electronics - Berlin (ICCE-Berlin)*. Piscataway, New Jersey: IEEE Press, Sep. 2011, pp. 216–220.

[7] B. Müller-Rathgeber, M. Eichhorn, and H.-U. Michel, "A unified car-it communication-architecture: Design guidelines and prototypical implementation," in *IEEE Intelligent Vehicles Symposium, 2008*, Jun. 2008, pp. 709–714.

[8] F. Reimann, A. Kern, C. Haubelt, T. Streichert, and J. Teich, "Echtzeitanalyse Ethernet-basierter E/E-Architekturen im Automobil," in *GMM-Fachbericht - Automotive meets Electronics, (Automotive meets Electronics (AmE'10), Dortmund, Germany, Apr. 15-16, 2010)*. Berlin: VDE Verlag, 2010, pp. 9–14.

[9] W. Steiner, "TTEthernet Specification," TTTech Computertechnik AG, Nov. 2008. [Online]. Available: http://www.tttech.com

[10] E. Bringmann and A. Krämer, "Model-based testing of automotive systems," in *2008 1st International Conference on Software Testing, Verification and Validation*, Apr. 2008, pp. 485–493.

[11] J. Schäuffele and T. Zurawka, *Automotive Software Engineering*. Wiesbaden: Vieweg und Teubner, 2010.

[12] F. Bartols, “Restbussimulation von Time-Triggered Ethernet in Automotive Anwendungen: Verwandte Arbeiten,” Aug. 2011, bericht.

[13] Vector Informatik, “CANoe - Restbussimulation,” Vektor Informatik. [Online]. Available: http://www.vector.com/vi_canoe_de.html

[14] K. Yaghmour, J. Masters, G. Ben-Yoseff, and P. Gerum, *Building Embedded Linux Systems - 2nd Edition*. O’Reilly Media, Inc., Aug. 2008.

[15] T. Ts’o, D. Hart, and J. Kacur, “RT-Kernel Wiki,” 2012. [Online]. Available: https://rt.wiki.kernel.org/index.php/Main_Page

[16] F. Bartols, T. Steinbach, F. Korf, and T. C. Schmidt, “Performance Analysis of Time-Triggered Ether-Networks Using Off-the-Shelf-Components,” in *14th IEEE International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing Workshops (ISORCW) 2011*. Piscataway, New Jersey: IEEE Press, Mar. 2011, pp. 49–56.

[17] ASAM - Association for Standardisation of Automation and Measuring Systems, “Data Model for ECU Network Systems (Field Bus Data Exchange Format),” ASAM e.V., Specification 4.0.0, Sep. 2011.

[18] Association for Standardisation of Automation and Measuring Systems, “ASAM.” [Online]. Available: http://www.asam.net

[19] T. M. Galla, “Cluster Simulation in Time-Triggered Real-Time Systems,” Ph.D. dissertation, TU Wien, Wien, Dec. 1999.

[20] F. Bartols, “Restbussimulation von Time-Triggered Ethernet in Automotive Anwendungen – AW1 Ausarbeitung,” Feb. 2011, bericht.

[21] M. Schlager, *Hardware-in-the-Loop Simulation*. Vdm Verlag Dr. Müller, Apr. 2008.

[22] K. Klobedanz, A. Koenig, and W. Müller, “A reconfiguration approach for fault-tolerant FlexRay networks,” in *Design, Automation Test in Europe Conference Exhibition (DATE), 2011*, Mar. 2011, pp. 1–6.

[23] K. Klobedanz, A. Koenig, W. Müller, and A. Rettberg, “Self-Reconfiguration for Fault-Tolerant FlexRay Networks,” in *2011 14th IEEE International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing Workshops (ISORCW)*, Mar. 2011, pp. 207–216.

[24] G. J. Meyers, C. Sandler, and T. Badgett, *The Art of Software Testing*. New Jersey: John Wiley & Sons, 2011.

[25] The World Wide Web Consortium, “About W3C,” W3C. [Online]. Available: http://www.w3.org/Consortium/

[26] E. R. Harold and W. S. Means, *XML in a Nutshell, Third Edition*, S. S. Laurens, Ed. Sebastopol, CA 95472: O’Reilly, Oct. 2004.

[27] W3C XML Working Group, “The xml 1.0 standard (5th edition),” 2008. [Online]. Available: http://www.network-theory.co.uk/w3c/xml/

[28] Vector Informatik, “DBC Communcation Database for CAN,” Vektor Informatik. [Online]. Available: http://www.vector.com/vi_candb_en.html

[29] SAE - AS-2D Time Triggered Systems and Architecture Committee, “Time-Triggered Ethernet (AS 6802),” 2009. [Online]. Available: http://www.sae.org

[30] M. Schwager, *Using Matlab with CANoe*, 1st ed., Vector Informatik GmbH, Apr. 2008.

[31] T. Steinbach, H.-T. Lim, F. Korf, T. C. Schmidt, D. Herrscher, and A. Wolisz, “Tomorrow’s In-Car Interconnect? A Competitive Evaluation of IEEE 802.1 AVB and Time-Triggered Ethernet (AS6802),” in *Proceedings of the 76th IEEE Vehicular Technology Conference (VTC2012-Fall)*, 2012, under submission.

[32] K. Müller, “Time-Triggered Ethernet für eingebettete Systeme: Design, Umsetzung und Validierung einer echtzeitfähigen Netzwerkstack-Architektur,” Aug. 2011, bachelorthesis.

[33] K. Müller, T. Steinbach, F. Korf, and T. C. Schmidt, “A Real-time Ethernet Prototype Platform for Automotive Applications,” in *2011 IEEE International Conference on Consumer Electronics - Berlin (ICCE-Berlin)*. Piscataway, New Jersey: IEEE Press, Sep. 2011, pp. 221–225.