

Car-to-X Simulation Environment for Comprehensive Design Space Exploration Verification and Test

2010-01-0451

Published
04/12/2010

Christoph Roth, Oliver Sander, Michael Hübner and Jürgen Becker
Karlsruhe Institute of Technology (Information Processing Technology Lab (ITIV))

Copyright © 2010 SAE International

ABSTRACT

A future car-to-x communication system has to fulfil a lot of different requirements concerning high performance and functionality that are given by the field of application. To be able to optimize the system architecture regarding these constraints an intensive architecture evaluation and investigation is necessary.

Within this paper a simulative approach for comprehensive design space exploration, verification, and test of a car-to-x communication unit is presented. The proposed simulation environment allows for a flexible adaption to the test case by being able to interconnect an arbitrary number of simulators of different type and different granularity. As a novelty complete embedded car-to-x systems can be investigated by integrating several SystemC based architecture models into an environmental simulation and observing their behavior and interaction. We introduce the concept of scaling in combination with a hybrid simulation approach which allows eliminating unnecessary details of simulation. So an aimed investigation with maximum possible simulation performance can be done which is the basis for an extensive architecture optimization. The simulation can optionally be executed in real-time. This enables a connection of real car-to-x hardware for advanced verification.

INTRODUCTION

Car-to-x communication is the next step towards increased traffic safety and efficiency. Future vehicles will be equipped with a wireless car-to-x communication unit, establishing a so called Vehicular-Ad-hoc-NETwork (VANET). Information exchange is done via different types of messages which are transmitted periodically or event-triggered. Inside the receivers information is aggregated, processed, and presented

to the driver. That way the range of the internal sensor/actor systems of each vehicle is extended across the line of sight since information is received not only from the local area around the vehicle.

The proposed simulation environment is intended for the evaluation and test of embedded car-to-x communication systems. The concepts are introduced using the example of an FPGA based car-to-x communication unit that has already been developed. By using an FPGA instead of a conventional microcontroller based system a lot of new realization alternatives open up, especially if dynamic partial hardware reconfiguration is applied. Nevertheless a generalization to a simulation environment for other car-to-x systems consisting of complex hw/sw architectures is possible.

The environmental scenarios a car-to-x system is exposed to, range from simple ones in which only a few number of vehicles take part to complex wide range scenarios with several thousands of vehicles. Beside that a great amount of different test cases is imaginable that focus on different functional parts of the system. For that reason the proposed simulation environment is highly adaptable to the particular test cases. It comprises three domains ([Fig. 1](#)), namely Network Domain, Traffic Domain and System Domain which are established by the flexible interconnection of three types of simulators: traffic simulators, network simulators and system simulators. Each connected system simulator can be stimulated. In turn each system can influence all remaining simulators. Since several system simulators can be connected and executed in parallel, their interaction behavior is observable in detail. A positive implication of the modular structure is the opportunity to set up most diverse application scenarios, varying in complexity which is the basis for a comprehensive investigation.

The environment provides a virtual representation of the real world with the required degree of abstraction. All simulators work together on different levels of accuracy by using different granularity. Since accuracy of each simulator is adaptable and scalable, an adjustment of the overall test system to the performance and accuracy demanded by the particular test case is feasible. For example system models are developed using the executable specification language SystemC which enables a top down development approach. Different sub systems can be modelled on different levels of abstraction. The beneficial result is the opportunity for an aimed investigation of a car-to-x units' sub systems under as realistic conditions as necessary.

The opportunity to either execute the environment in full simulation mode or real-time mode allows either connecting system simulators or real hardware. Connecting real hardware is the last step in a top down development process as it is supported and facilitated by SystemC and required for a last intensive verification of the system before it is tested and applied in the real world.

The paper is structured as follows: After a presentation of the car-to-x system and a motivation for the proposed simulation environment, a description of related work is given. In the subsequent section the principles of the novel simulation environment are described and the wide-ranging applicability and the benefits of the introduced concept are shortly illustrated by means of two examples. Afterwards computational challenges, a methodology to cope with these challenges and aspects for implementation are discussed. Finally the last section concludes the proposed approach and gives an outlook to tasks and issues that are lying ahead.

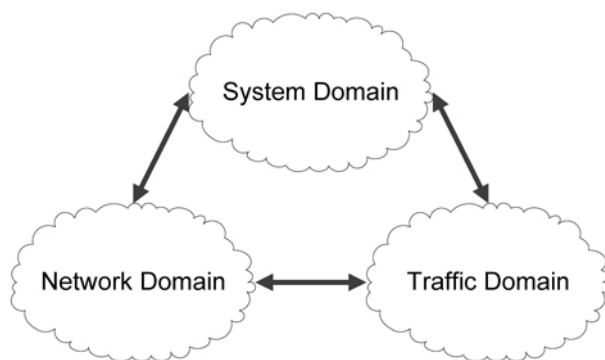


Fig. 1. Simulator Domains

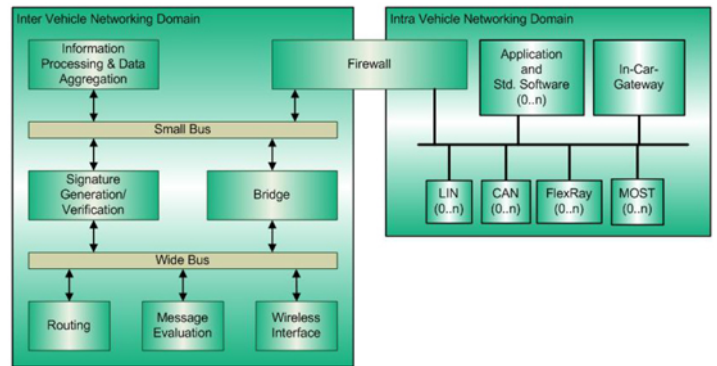


Fig. 2. Car-to-X System

CAR-TO-X SYSTEM

The car-to-x system (Fig. 2) was previously presented in [1], [2], [3] and [4]. It is based on the last North American and European standardization efforts ([5], [6], [7]) provides a flexible and modular hardware framework for various car-to-x applications and routing algorithms and is implemented on an FPGA.

The concept combines a car-to-x unit and a central gateway on one single chip. It forms the link between the car-to-x network and the internal vehicular bus network and is divided into an Inter-Vehicle-Networking-Domain (InterVND), providing an interface to the external car-to-x network, and an Intra-Vehicle-Networking-Domain (IntraVND), providing an interface to the internal vehicular bus-systems.

The system is able to fulfil accruing tasks by taking advantage of a highly specialized system architecture. Different modules, each fulfilling a certain accruing task (for example security and data preprocessing, routing or evaluation and filtering of single incoming messages), work in parallel and communicate via a packet based communication system. The modules consist of a complex structure of different hardware and software components. Since environmental conditions are often unforeseeable, selfadaptivity significantly contributes to respond to the different demands of urban and rural traffic. Because of that self-X characteristics have been embedded into the system wherever possible for example by utilizing partial dynamic reconfiguration.

MOTIVATION

When designing an embedded system, different realization alternatives need to be compared regarding certain optimization criteria like performance, functionality, energy consumption, flexibility or data throughput. To enable a comparison of different hw/sw architectures, simulation is an approach often used. The system architecture then is described using executable specifications like SystemC which

allows for the integration of metrics into the code for evaluation. Beside that an executable specification like SystemC allows for a functional verification of the system architecture already in early development stages since the specification can be executed even if different parts of the system are modelled with different accuracy.

In the special case of the design of a car-to-x communication system the possibility for the execution of one single system specification is not enough for a comprehensive investigation since the system is characteristically in high interaction with the environment. Several applications (security applications, traffic information applications ...) are running in parallel and exchange information with the environment. For that reason the simulation environment must provide a representation of the real environment that is able to generate stimulation data on application level. Furthermore test and evaluation of the correct interaction between functions running on a certain number of distributed nodes is crucial. Mutual impact of one function running on different nodes can directly be observed on all participating nodes and errors can be detected easier.

In the area of car-to-x self-adaptability is an important feature which has to be evaluated intensively. If using an FPGA as basis, adaptable software components as well as dynamic reconfigurable hardware components may exist parallel on one single chip. Self-adaptability highly depends on interaction with the environment. Simple testbenches generating ordinary input signals and monitoring the output signals are not sufficient. The car-to-x system is rather embedded into an impact loop, that means output information (messages on application level in case of the car-to-x unit) generated on the basis of some input data should in turn influence the behaviour of the environment.

Being able to exchange the system models with real hardware units (realtime mode) would allow an evaluation under more realistic conditions, for example the real energy consumption of one single node could be measured by using real hardware. Beside that it could be verified, that the system models are correct and show the same behavior as real hardware nodes.

RELATED WORK

Regarding car-to-x communication a lot of research is done worldwide. Main focus is the development of applications and protocols. Application or protocol tests are usually done with the help of simulators. The hardware on which applications finally will be executed actually plays a minor part. To the best of our knowledge also the test of the interaction of a number of complete On Board Units (OBUs) with each other is not considered. In general simulation is inevitable since testing of applications or complete systems in real traffic would be too dangerous. One often used approach for the creation of a simulation environment for application

testing is the interconnection of an event-based network simulator like NS-2 [22] or JiST/SWANS [23] with an event-based traffic simulator like SUMO [24] or VISIM [25] (e.g. [8], [9], [10]). The open-source tool TraNS [11] combines the traffic simulator SUMO with the network simulator NS-2. A feedback loop is established via a TCP connection between both simulators that enables direct interaction. The protocol techniques are described in [12]. In [13] a car-to-x simulator environment called VSimRTI is presented which is based on the High Level Architecture (HLA) [14]. It allows for a flexible combination of time discrete simulators and the integration of real car-to-x applications. The tool presented in [15] allows for application development for provided On Board Units. It is not clear whether custom hardware OBUs even less complete embedded system models can be added to the environment.

A first approach in which also the execution of hardware comes into play is presented in [10]. Application models as well as real hardware are coupled to an open-source simulator environment for application testing. However, it does not allow for the integration of realistic hw/sw system models of complete embedded systems. The hardware units on which the applications are executed are no real OBUs but small model cars equipped with an embedded computer.

First steps towards testing of a complete FPGA based OBU already have been done in [16]. Here a simple traffic simulator serving as message generator is coupled to a car-to-x unit for stimulation. A simple functional verification of a single hardware unit is possible but no advanced testing. Beside that only real hardware can be integrated. A design space exploration with complete system models is not feasible.

None of the presented work allows for the evaluation of a complete system architecture of a complex sensor/actor node that is part of a huge dynamic sensor/actuator network and none of them allows for the connection of several system models specifying hardware and software components. The problems regarding simulation performance that come along with such comprehensive simulations still have not been touched. Also the scaling technique that allows a tradeoff between accuracy and performance is a novelty.

SIMULATION ENVIRONMENT

VIEWS

When regarding the simulation environment as a parallel simulation of different domains this is called the Simulated Domain View. Three domains can be distinguished (Fig. 3): Network Domain, Traffic Domain and System Domain. Each domain is established by different kinds of simulators: Network simulators, traffic simulators and system simulators. The simulators are coupled together and exchange relevant