Simulation Environment For FPGA-based Sensor-Actuators For Active Vibration Control

François Philipp, Faizal Arya Samman and Manfred Glesner
LOEWE-Zentrum AdRIA, Forschungsgruppe Mikroelektronische Systeme,
TU Darmstadt, Deutschland, Email: francois.philipp@mes.tu-darmstadt.de

Introduction

In order to keep the computational latency within the domain of real-time constraints, a high performance signal processing unit is essential for a complex distributed active vibration control system. In a decentralized scenario, where multiple control units are spatially deployed on structures, customized computing units could furthermore compensate delays induced by the communication between the control units. FPGA devices are highly suitable for such scenarios since they allow a concurrent implementation of multiple computing units, leading to higher performance results compared to its counterpart, DSP devices. However, processing and communication delays of such systems are difficult to precisely estimate and complicate the optimization of the system. An accurate model is then required accordingly. In the rest of this paper, simulation environments for a mechanical system subjected to active vibration control employing distributed sensor-actuator nodes are presented. The first section shortly describes classical development and simulation flows. A second section introduces a development environment based on the Matlab/Simulink Framework and illustrated by a design realized on the HaLOEWEn platform [2]. Finally, a novel environment based on the VHDL-AMS language is presented.

Classical Flow

During the development of a control system with stringent time constraints, like it is usually the case for a distributed sensor-actuator network for active vibration control [4], not only the functionality of the control algorithm must be validated but also its implementation. In particular, the efficiency of the control system could be improved by reducing processing delays. In the case of FPGAs or integrated systems implemented on custom hardware in general, this process involves development cycles combining programming using Hardware Description Languages (HDL) with behavioral and post-synthesis simulations of the design. Specialized tools like ModelSim from Mentor Graphics enable cycle-accurate simulation for the digital part of such systems. At a higher level, the communication within a distributed sensor-actuator network must also be simulated and validated. As they often are the bottleneck of such systems, communication protocols must be accurately tested. Network simulators like COOJA for wireless networks based on the Contiki operating system or the Omnet++ framework are commonly used for such purposes. The Matlab-Simulink framework is then frequently adopted to develop, simulate and optimize the control algorithm. In the case of active vibration control applications, effects on the mechanical structure must be simulated and verified. The differential partial equations involved in such cases are then usually solved using FE-models and software like Ansys. Figure 1 illustrates a cyclic development process of a distributed sensor actuator network using the aforementioned simulation tools.

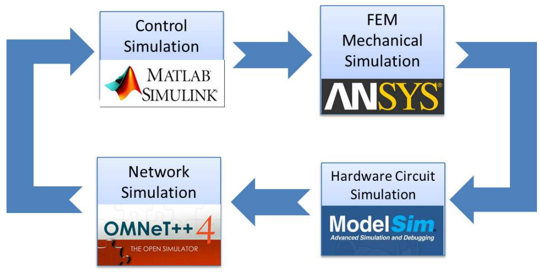

Figure 1: Classical Iterative Simulation Flow Using Multiple Specialized Simulators

A major issue with such process is to accurately specify the interface between the different tools, especially for simulation in the time domain. As they are all based on different simulation methods, porting results obtained from one tool to another inevitably require the establishment of error-prone abstractions. Multiple abstractions between different tools further create an avalanche effect which can lead to erroneous simulations. It is therefore very likely to integrate the development and the simulation of the system in the minimum amount of tools.

Matlab/Simulink Environment

Recent innovations in the domain of high-level FPGA CAD tools enabled the synthesis of digital hardware designs directly from a Matlab/Simulink model. Precompiled blocksets are provided by FPGA vendors for the fast elaboration of high performance signal processing cores. Simulink models used for the development of adaptive controllers could thus be compiled and uploaded to the target platform after only a short hardware design time.

When considering network communication, existing tools like TrueTime could be coupled to the Simulink model for accurate, concurrent simulation of multiple nodes. Simulation of the mechanical structure is under certain conditions supported as well by the environment. Without taking into consideration the recent FEM toolbox from Matlab, reduced order models obtained from FEM tools can be translated into Simulink blocks for fast and precise evaluation of the system [1]. This however only applies to systems that can be evaluated deterministically and modally decomposed. Similarly, models of sensors, actuators and the related electronic circuitry (ADCs, DACs , filters, etc. . .) can be integrated into a global Simulink model in order to take the non-ideal characteristics of the acquisition system into account. The obtained environment allows a simultaneous and fast evaluation of the complete system with sufficient accuracy for both mechanical and electrical elements.

When hardware prototypes become available, simulations are then iteratively replaced by real experiments enabling so called Hardware-In-The-Loop development. In such cases, reaction of the environment to the signals issued from the hardware are emulated by an external system connected to the target platform. If the available resources are sufficient and the complexity of the system stays low, high-level synthesis tools can be used to emulate the complete system, including sensors, actuators and the mechanical system on one single FPGA. This again aims to simplify and accelerate Hardware-In-The-Loop developments.

Following this approach, an active noise control application has been fully emulated on the HaLOEWEn sensor node based on Microsemi IGLOO FPGAs [2]. The simplified block diagram of the system is depicted in Figure 2.

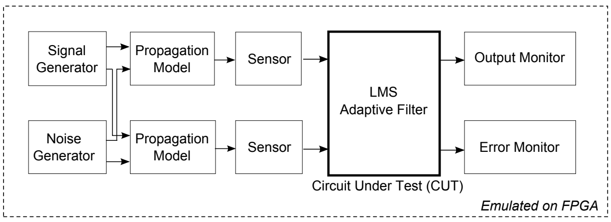

Figure 2: Hardware-In-The-Loop simulation on FPGA from Simulink models

Output signals of the adaptive controller under test were observed with a logic analyzer. The high number of I/Os of a FPGA allows the simultaneous monitoring of a large amount of signals. Using this method, the implementation of the filter has been optimized applying a so called algorithm-architecture co-design. The performance difference between a FIR and a IIR type of filter has notably been evaluated. The total area utilization of the FPGA did not exceed 70%, from which 90% was allocated to the circuit under test itself. Considering that the rest of the model occupies a limited space and that the remaining available area is still large, emulation of more complex designs is possible. Furthermore, the target FPGA belongs to a family of low-power devices with limited resources. Chips from other vendors could possibly implement larger designs.

However, a major problem using Simulink to FPGA synthesis is that the obtained hardware design is not optimal. High-level synthesis tools are not yet mature to achieve the performance and area utilization of designs developed in HDL. We therefore introduce another solution to face this issue.

VHDL-AMS Environment

VHDL-AMS is a description language for mixed-signal environments. Unlike VHDL or Verilog, which are limited to digital systems, VHDL-AMS allows developers to include models of analog circuits in their designs. Extensions of the language based on specific IEEE libraries even enable mixed-domain simulation. Models could then simultaneously include electrical, mechanical, fluidic, thermal or magnetic elements. A model of an electro-mechanical actuator for active vibration control in the automotive domain was for example described in [3]. As VHDL-AMS is based on the VHDL language, it becomes then straightforward to link it to the design of an optimized digital controller for an accurate simulation. Using the same approach as Simulink, complex mechanical structures can be simulated with reduced order models obtained from FEM tools. Mixed-signal modeling of ADCs and DACs further enable the performance analysis of anti-aliasing and reconstruction filters. A model for communication networks can be established as well, especially for RF interfaces which are widely modeled using VHDL-AMS. All in all, VHDL-AMS offers similar features as Simulink but allows a much higher level of refinement for the electronic hardware design. This feature is however at the cost of simulation time, leading to the traditional accuracy versus duration trade-off.

Acknowledgments

This work has been supported by the European project Maintenance on Demand (MoDe) Grant FP7-SST-2008-RTD and by Hessian Ministry of Science and Arts through project AdRIA (Adaptronik-Research, Innovation, Application) with Grant Number III L 4-518/14.004 (2008).

References

[1] Herold, S., Mayer, D. and Hanselka, H.: Transient Simulation of Adaptive Structures. J. of Intelligent Material, Systems and Structures, 2004, 15, 215 - 224

[2] Philipp, F., Samman, F. A. and Glesner, M.: Design of an Autonomous Platform for Distributed Sensing-Actuating Systems. In Proc. of the IEEE International Symposium on Rapid System Prototyping (RSP), 2011

[3] Wang, L. and Kazmierski, T. J.: VHDL-AMS Modeling of An Automotive Vibration Isolation Seating System. In Proc. of the International Conference on Circuits, Signals and Systems, 2005

[4] Orosz, G., Sujbert, L. and Péceli, G. "Real" Signal Processing with Wireless Sensor Networks Proceedings of Regional Conference on Embedded and Ambient Systems (RCEAS), 2007