Demo Abstract: An Inverted Pendulum Demonstrator for Timed Model-Based Design of Embedded Systems

Kai Huang, Gang Chen, Nadine Keddis, Michael Geisinger, Christian Buckl
fortiss GmbH, Guerickestr. 25, 80805 Munich, Germany
Email: {firstname.lastname}@fortiss.org

I. Introduction

Model-based approaches for designing embedded systems are increasingly popular in order to cope with the ever-scaling complexity of the integrated software [1]. Such approaches can reduce the design complexity by hierarchically modeling a system at different levels of abstraction and using code generators to convert system abstraction from one level to another. Meanwhile, applying formal models is the prerequisite for providing guarantees on extra-functional properties.

An important extra-functional property, for instance for the classes of automotive and autonomous embedded systems, is timing, which relates to the predictability of the system-level timing behavior, e.g., end-to-end latencies and jitter. Being aware that priority-based scheduling can lead to discontinuous behavior in time and timing anomalies [2], timing isolation in software execution is required to protect safety-critical embedded systems from timing faults.

Within this context, we are developing a modular software framework named CHROMOSOME [3], serving as a middleware and execution platform for model-based software design of safety-critical embedded systems. Specifically in this work, we enable temporal isolation within the middleware and use an inverted pendulum case study to demonstrate our framework on an FPGA platform.

II. Background

CHROMOSOME Middleware We are prototyping a software framework for model-based design of embedded systems. The core of the framework is the CHROMOSOME middleware which employs the concept of data-centric design: components within the middleware communicate with each other by means of a publish-subscribe mechanism, forming a self-organized dynamic system. The advantage of the framework is the modularity that is applied for both the design of the middleware itself and the application functionality running on top of it. One major goal of CHROMOSOME is to take into account extra-functional requirements such as timing or fault-tolerance when executing an application. This demo shows a first approach towards this goal.

Inverted Pendulum The inverted pendulum [4] is a classical problem in dynamics and control theory and is therefore often used to test time critical control software. During runtime, a feedback control is used to stabilize an open-loop unstable system. In our case the pendulum is on a cart. The control software swings up the pendulum that is initially in a suspended position (i.e., down) and balances it upright while tracking a desired cart position. The controller uses a proportional-velocity algorithm during the swing-up state and uses a linear quadratic regulator to maintain the balanced state. Figure 1 shows such an inverted pendulum.

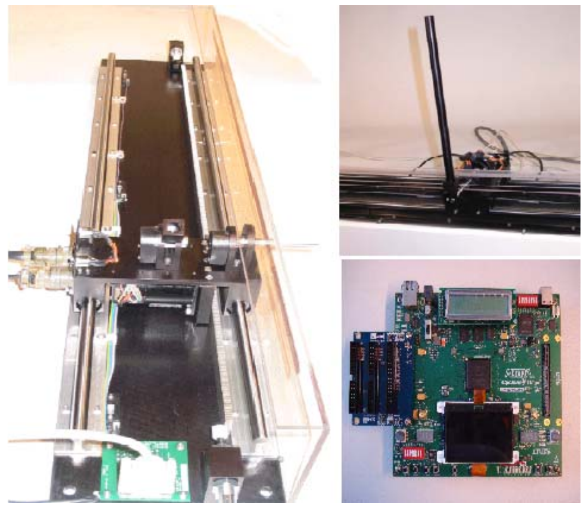

Fig. 1. The inverted pendulum demonstrator and the FPGA platform.

III. Demonstration

In this demonstration, we show following contributions:

- We developed an FPGA implementation for our middleware based on the Altera® Nios II soft core. Applications developed on top of the middleware can be seamlessly run on the Altera CYCLONE III FPGA without being aware of the underlying hardware configuration.
- We prototyped a time division multiple access (TDMA) scheduling policy on top of the FreeRTOS real-time operating system to provide temporal isolation for the middleware. Currently, the context switching overhead for our scheduling is within 50 microseconds at an FPGA clock frequency of 85 MHz. We expect a smaller overhead with further investigation.
- We implemented the inverted pendulum using our CHROMOSOME middleware. The inverted pendulum runs in a temporally isolated manner, without interference from the workload injected from other parts of the system.

References

[1] R. Passerone et al., "Metamodels in europe: Languages, tools, and applications," *IEEE Design & Test of Comp.*, vol. 26, pp. 38–53, 2009.
[2] A. Sangiovanni-Vincentelli et al., "Embedded system design for automotive applications," *IEEE Computer*, vol. 40, pp. 42–51, Oct. 2007.
[3] fortiss: CHROMOSOME, "http://chromosome.fortiss.org/."
[4] Quanser Inc., "Self-erecting single inverted pendulum control. Document number: 532; revision: 02."

978-0-7695-4695-7/12 $26.00 © 2012 IEEE
DOI 10.1109/ICCPS.2012.45