An end-to-end approach to Design and Verify Battery Management Systems: from Requirements to Virtual Field Testing

Problem Statement

Modern battery management systems (BMS) ensure cell packs' safe and efficient operation in various solutions within electric vehicles, power supplies, grid energy storage systems and most battery-driven equipment. Due to the wide range of domains where systems are battery powered, BMS algorithms development aim to protect the battery under numerous environmental conditions, to prevent discharge/overcharge and increase its lifespan.

To design efficient BMS algorithms, engineers need to get reasonable estimations of batteries operating modes, and they need to build an accurate battery model, which is often hard to characterize. Also, such systems imply long development iteration cycles due to overreliance on hardware-based testing, since most of the BMS algorithms are still tested on actual battery cells.

Therefore, the process of building such complex BMS applications, with a high degree of reliability, raises the challenges of thorough analysis, design, implementation, software integration, and testing stages. This entails a real need for efficiency in design and verification, which represents a major factor in companies keeping up with the high pace of innovation and competition.

State of the Art

Modelling, simulation, and virtual field testing are well-established approaches that allow to address algorithm design and verification challenges. When coupled with embedded code generation and other design automation capabilities, they can be used to set up model-based design workflows, which are proven to improve development efficiency and are nowadays adopted as a de facto standard in many domains, including BMS design.

However, the seamless transition from requirements, to models, test cases, generated code, digital twins, and hardware prototypes is very hard to achieve, since gaps still exist in most of the currently implemented model-based design workflows, preventing full artifact reuse and end-to-end design automation (for instance within continuous integration platforms).

Proposed Solution

In this tutorial, we present a fully connected model-based engineering workflow to develop and test BMS solutions that allow to overcome the previously described limitations.

We rely on a pragmatic digital approach, that represents the BMS and its environment as a single digital thread, that is a model reused across all development stages, from early specification and design to the final implementation and validation on the intended target hardware.

We start with an architecture-level model and intuitively sketch a system architecture that captures system requirements. The proposed workflow ensures a seamless transition into the design stage, where we incorporate both the BMS controller and battery pack behavioral models to perform design explorations and automated verification of operational and faulty BMS scenarios, using desktop simulation.

Using the same system model, we leverage specialized coders to automatically generate highly efficient and portable code from the models of BMS controller and plant components (such as battery packs). We then show how to reuse existing test artefacts to run non-real-time closed-loop verification using Software-in-the-Loop (SiL) and Processor-in-the-Loop (PiL) techniques. After that, we present how to validate the BMS embedded hardware through real-time virtual field testing, using specialized Hardware-in-the-Loop (HiL) solutions. Finally, we outline how to expedite and further automate testing of BMS solutions with the use of continuous integration methods that automate the build, test, and package of BMS software.

TUTORIAL SUBMISSION INFORMATION

Title

An end-to-end approach to Design and Verify Battery Management Systems: from Requirements to Virtual Field Testing

Topic

MODEL-BASED & MODEL SUPPORTED SOFTWARE DESIGN – Model based software design

Type

Education/Workshop

Presenters and Contributors

Irina Costachescu, Model Based Design Software Engineer, NXP Semiconductors

Email: irina.costachescu@nxp.com

Phone: +40 746513537

Marius-Lucian Andrei, Model Based Design Software Engineer, NXP Semiconductors

Email: mariuslucian.andrei@nxp.com

Phone: +40 723736284

Manuel Fedou, Senior Application Engineer – Electrification, Speedgoat

Email: manuel.fedou@speedgoat.ch

Phone: +41 31-552-1645

Conrado Ramirez Garcia, Application Engineer - Code Generation, MathWorks

Email: cramirez@mathworks.com

Phone: +49 894-5235-6783

Contributors

Mauro Fusco, Application Engineer - Design Automation and Code Generation, MathWorks

Email: mfusco@mathworks.com

Phone: +31627461207

Constantin Razvan Chivu, Model Based Design Manager, NXP Semiconductors

Email: constantinrazvan.chivu@nxp.com

Phone: +40 729970173

Cristian Macario, Industry Manager – Communications, Electronics, Semiconductors, MathWorks

Email: cmacario@mathworks.com

Phone: +49 89-45235-6817

Daniel Fonte, Head of Application Engineering, Speedgoat

Email: daniel.fonte@speedgoat.ch

Phone: +41 31-552 -0535