Leading the
Electric
Vehicle
Charge

The battery management system plays a critical role in today's electric vehicles: It monitors the state of the battery, manages its operation to optimize vehicle performance and battery life, and ensures the safety of occupants. Researchers at National Electric Vehicles Sweden (NEVS) who are developing the BMS for their next-generation electric vehicle are achieving 30 percent higher productivity by using ANSYS SCADE Suite instead of the industry's traditional model-based design toolset.

By **Christian Fleischer**,
Manager Software Architecture –
Advanced Battery Technology,
National Electric Vehicles Sweden,
Trollhättan, Sweden

> **"The SCADE model allows them to *detect errors* in their specification *early* in the *design process instead* of during integration testing."**

The battery is the most expensive component in an electric vehicle. It helps to determine the range of the vehicle, which is a crucial performance specification parameter. The battery management system (BMS) plays a key role in optimizing the performance of the battery by monitoring its condition and controlling its operation. As the brains behind delivering the power required to operate the vehicle, it conserves the charge to prolong the life of the battery, and detects and responds to unsafe operating conditions. A team of 10 researchers at NEVS, the successor to the famed Saab automotive brand, are developing a cutting-edge BMS for the company's next-generation electric vehicle.

NEVS researchers did not use the automotive industry's most widely used model-based development toolset because it includes a code generator that is not compliant with the safety standard governing the embedded software development process. This toolset produces code that requires extensive manual verification, validation and back-to-back testing. Instead, NEVS researchers selected the ANSYS SCADE end-to-end model-based development solution, which includes an ISO 26262–qualified code generator that eliminates the need for costly code reviews and low-level testing activities to verify that the code is functionally equivalent to the model. The team expects to achieve a 30 percent productivity gain compared to the usual toolset. Upfront simulation is vital in the electric vehicle industry where competition is fierce and time to market is paramount.

^ The BMS master controller embedded hardware (red) sits on top of the BMS slave controller (green), which in turn sits on top of the battery demonstrator. The slave controller measures cell voltages and temperature, and executes the cell balancing strategy. The high-voltage battery disconnect (black box in rear) is responsible for connecting and disconnecting the battery from the load.
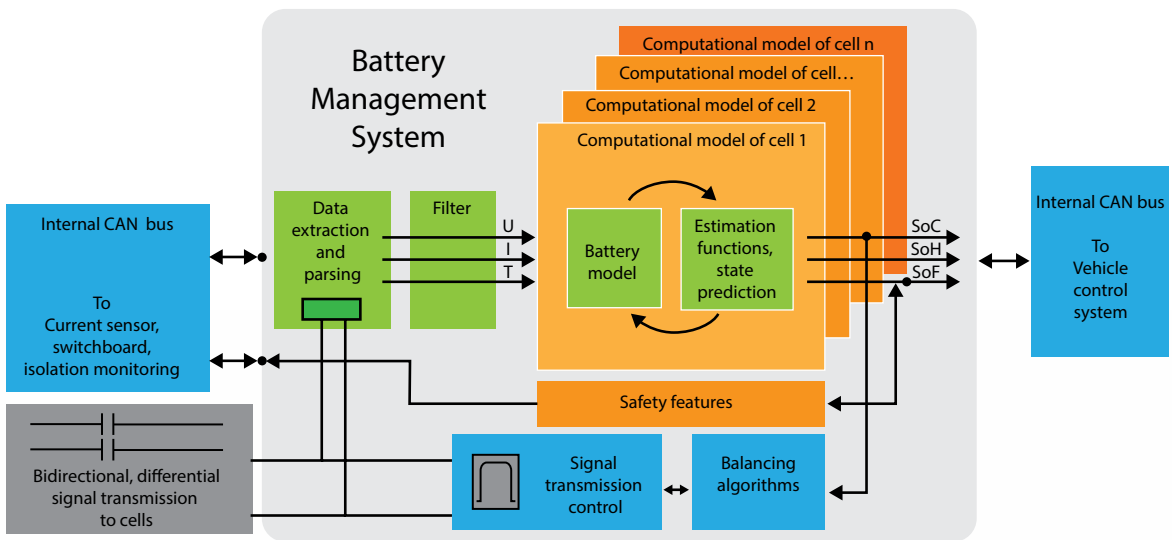
UNDERSTANDING BATTERY STATE

The battery of an electric vehicle is a highly nonlinear system with a continually changing state. Understanding changes in each cell of the battery is critical to optimizing the performance of an electric vehicle. The battery contains a wide range of sensors to monitor its operation, and the BMS interfaces with these sensors to measure voltage, current, temperature and other parameters. But it is not possible to incorporate enough sensors to provide a complete picture of the battery's condition, so the BMS uses complicated algorithms to estimate the battery's state of charge (SOC)

^ **Overview of battery management system functionality.** *Information courtesy Fraunhofer IISB.*

[the equivalent of a fuel gauge for the battery pack], its state of health (SOH) [the condition of the battery compared to its ideal conditions], its state of life (SOL) [the remaining useful life of the battery] and other important parameters.

Based on its estimation of the battery condition, the BMS controls the battery to protect against over-charge, over-discharge, over-current, short circuits, overheating, ground-faults and other potential problems to maintain the battery in a state in which it can fulfill its functional design requirements for as long as possible. The BMS also informs the application controller how to make best use of the battery at any moment to power the vehicle.

SELECTION OF MODEL-BASED DEVELOPMENT TOOLSET

When preparing for the BMS development process, NEVS researchers evaluated the leading model-based development environments. Model-based development has greatly improved the quality and time-to-market of safety-critical automotive systems such as BMSs by enabling a graphical model to replace software architectural and unit design during the development process. Engineers can simulate the behavior of the model and immediately view the results without the need for physical hardware. This makes it possible to gain critical insights early in the design process and rapidly improve the model's performance. Later, the model is used to automatically generate the embedded code, eliminating the need for manual coding and enabling the engineers to test their model instead of the code. This eliminates all code-based verification of the application.

> **"*The team* expects to achieve a *30 percent productivity gain* compared to the usual toolset."**

NEVS engineers recognized that they would need to automatically generate the embedded code many times during the development process as they improved the BMS. Using the traditional development toolset, each of these iterations would require back-to-back testing and functional testing phases to verify that the generated code implements the requirements correctly and matches the behavior of the model used in the design process. Some parts of this testing process can be automated, but others, such as demonstrating traceability of software requirements, must be done manually.
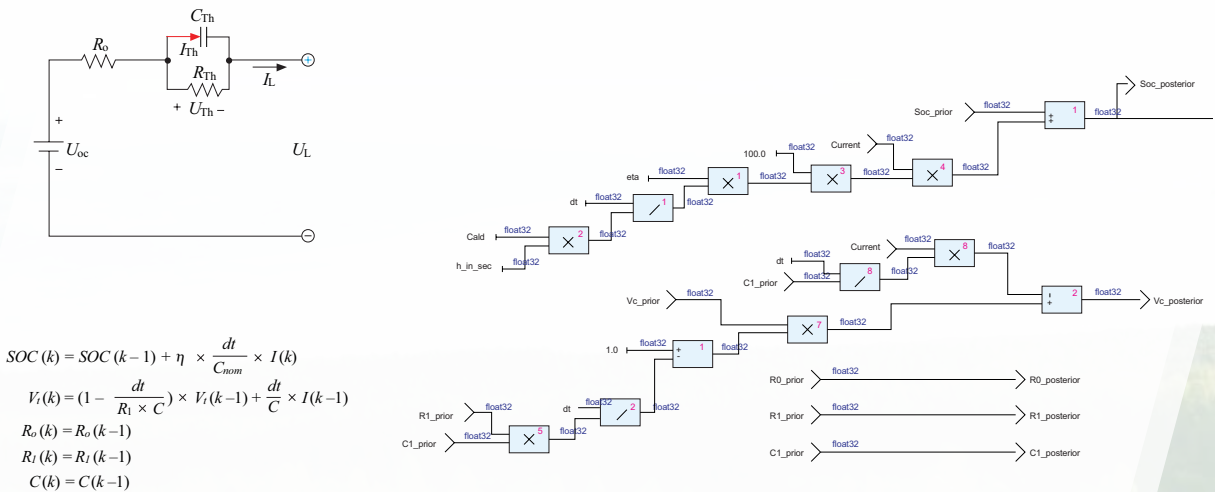
Developing ISO 26262–Compliant Automotive Safety-Related Applications
ansys.com/compliant

"NEVS researchers are *substantially reducing the time* required to develop the *battery* management system."

BMS DEVELOPMENT PROCESS

NEVS eliminated the need for this manual effort by using the ANSYS SCADE Suite complete end-to-end model-based development toolset to develop its BMS. The researchers use the SCADE Architect tool to describe the system and software architecture using SysML block diagrams. They write algorithms on a whiteboard to address the functional requirements of the BMS. This is followed by modeling the algorithms and other software components associated with the architecture, using complex state machines and data flows to model the logic and control laws. They then simulate the model to detect functional faults early in the design process. By creating, running and validating functional test cases in the SCADE Test environment for the SCADE model early in the design process the researchers can detect errors in their specification sooner, instead of during integration testing.

When researchers feel the model is ready, they use the SCADE Suite KCG code generator to generate C source code for the target environment. This code generator has been qualified for developing ISO 26262–compliant applications up to ASIL D, the highest safety requirement for automotive applications. They run the generated code on the target

$$SOC(k) = SOC(k-1) + \eta \times \frac{dt}{C_{nom}} \times I(k)$$

$$V_t(k) = (1 - \frac{dt}{R_1 \times C}) \times V_t(k-1) + \frac{dt}{C} \times I(k-1)$$

$$R_o(k) = R_o(k-1)$$

$$R_l(k) = R_l(k-1)$$

$$C(k) = C(k-1)$$

^ **View of battery management system model developed in ANSYS SCADE Suite**

hardware connected to a demonstrator battery to evaluate its performance. NEVS researchers have developed simulated driving cycles that they use on the demonstrator to evaluate the ability of the embedded software to accurately predict the state of the battery. The researchers tune the algorithms to better capture the physical and chemical behavior of the battery cells. For example, they compare the algorithm's voltage predictions with physical measurements on the battery cells.

By implementing the ANSYS SCADE tool chain, NEVS has achieved substantial advantages compared to automobile original equipment manufacturers using the conventional industry tool chain. Maximizing the performance of NEVS' next-generation vehicle will require many model-generation code-testing iterations. On each iteration, the SCADE KCG code generator eliminates the time normally required to verify that the generated code matches the model, to perform code reviews and to prepare documentation. As a result, NEVS researchers are substantially reducing the time required to develop the BMS. ⋀