

Low-cost Modular Battery Emulator for Battery Management System Testing

R. Di Rienzo, R. Roncella, R. Morello, F. Baronti, and R. Saletti
Dip. Ingegneria dell'Informazione, Università di Pisa, Via Caruso 16, Pisa, Italy

Abstract—This paper discusses the implementation of a custom battery emulator, specifically designed for functional testing of battery management systems at the end of the production line. Particular care has been paid to make the design of the battery emulator modular and low cost. These characteristics are sought in relatively low-volume medium-power battery applications, where the adoption of conventional hardware-in-the-loop solutions is not viable. A prototype of battery emulator has been implemented, validated, and successfully used to test a battery management system for 12 series-connected cells.

I. INTRODUCTION

Lithium-ion battery technology is presently the best choice to store electric energy in low-power devices, such as smartphones, tablets, and personal computers. It is also becoming the preferred choice in high power applications, especially in hybrid and electric vehicles, because of the outstanding features regarding energy and power density. In the last years, the use of lithium-ion batteries is also growing in the medium power/energy applications, such as off-road vehicles, gardening and power tools.

In medium power/energy systems, the battery is one of the most important and expensive parts, so that a good design and an appropriate test procedure are fundamental to increase the reliability of these systems, to avoid product recalls and to enhance customer satisfaction. An effective approach to increase the system reliability is to test each subsystem before assembling the complete system [1].

A lithium-ion battery consists of one or more elemental cells, properly connected in series or parallel to reach the desired voltage and capacity levels. The battery is controlled by an electronic circuit called Battery Management System (BMS). The BMS is in charge of monitoring the battery behavior and of applying safety policies in order to avoid catastrophic failures of the battery. The battery cells and the BMS should be tested separately, before assembling the battery. Usually, the cells are formed and tested by the cell manufacturer before delivery, so our work is focused on the BMS test, particularly on the functional test carried out in the final phase of the BMS production line [2].

One of the most used approach to test a BMS is based on the hardware-in-the-loop (HiL) paradigm, which consists in the real time emulation of the environment in which the BMS will operate [3], [4]. In the literature, HiL architectures are divided into two families: signal HiL architectures, which only emulate the signals from/to the BMS [4], and power HiL ones, which also emulate the power fluxes between the cells,

the BMS, and the load/charger [5]. Every function of the BMS hardware can be tested with realistic power fluxes, when power HiL architectures are used, even without a real battery. Instead, the signal HiL architectures provide limited testing capability, as no power exchanges occur between the BMS and the test infrastructure, but they are less expensive and safer.

Usually, the HiL solutions proposed in the literature or commercially available are rather expensive, because they are mostly used in the BMS design phase, when the battery behavior needs to be reproduced accurately, especially for the assessment of the complex algorithms used in the BMS to estimate the internal battery states [6]. For this reason, the real time battery simulators are complex and expensive, but they reach very high precision in simulating the cells and the load/charger behaviors. Furthermore, many medium power applications are characterized by rather low production volumes. Thus, the investments in the design of the system should be limited and classic solutions for HiL testing might not be affordable.

These considerations motivate the development of a custom BMS testing solution in these applications. The aim of this work is to develop a low-cost simulation platform based on the power HiL approach, for the functional testing of a BMS at the end of the production line.

The paper is organized as follows. The description of the typical architecture of a BMS for medium power/energy batteries is presented in Section II. This Section also describes the tests needed to verify the functionality of the BMS, before connecting it to the cells. Section III shows the developed testing platform that is based on the low-cost battery emulator described in detail in Section IV. The results of the validation experiments are described in Section V and some conclusions are drawn in Section VI.

II. BACKGROUND

The development of an effective battery emulation platform for the BMS functional testing starts from the identification of the general functions that a BMS for medium power applications should provide and the relevant possible sources of fault. The BMS for this type of applications usually manages from 6 to 16 series-connected groups of cells, where each group is composed of one or more cells connected in parallel. The number of parallel connected cells determines the capacity of the battery and, consequently, the runtime of the application. From the BMS point of view, the behavior of a group of paralleled cells is the same of a single cell with an equivalent

TABLE I
FUNCTIONS OF THE BMS AND RELATIVE POSSIBLE FAULT SOURCES.

Function	Fault sources
Acquisition of the cell voltage	Manufacturing defects in the cells' voltage measurement circuit The stack monitor does not work correctly There are issues in the communication between the microcontroller and the stack monitor There are malfunctions in the control algorithm of the stack monitor
Acquisition of the cell temperature	Manufacturing defects in the cells' temperature measurement circuit The stack monitor does not work correctly There are issues in the communication between the microcontroller and the stack monitor There are malfunctions in the control algorithm of the stack monitor
Acquisition of the battery current	Manufacturing defects in the current sensor circuit The microcontroller does not acquire correctly the current sensor output signal
Activation of the balancing circuit	Manufacturing defects in the balancing circuit The microcontroller cannot correctly control the activation of the balancing circuit There are malfunctions in the balancing algorithm
Activation of the protection switch	Manufacturing defects in the protection switch circuit The switch is not able to sustain the battery current The microcontroller cannot correctly control the switch There are malfunctions in the cell protection algorithm
Communication with the charger/load	Manufacturing defects in the communication interface There are issues on the communication bus There are malfunctions in the communication task

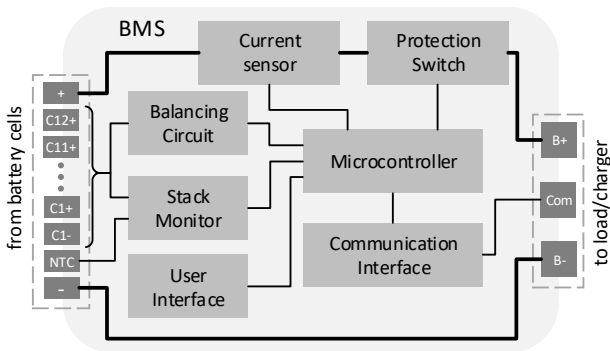


Fig. 1. Block diagram of a typical battery management system.

capacity equal to the sum of the individual cell capacities. Thus, the BMS design is independent of the number of cells in a group.

The typical block diagram of a BMS for a 12-cell battery for medium-power application is shown in Fig. 1. The main BMS scope is to control the battery cells, in order to avoid harmful situations that could lead to catastrophic failures or could reduce the battery lifetime [7].

The BMS is typically based on a microcontroller that acquires the cell voltage, current, and temperature values and manages the battery accordingly. To protect the battery, the BMS is equipped with a bidirectional protection switch, with which the microcontroller can interrupt the current flowing from/to the battery. This avoids unsafe situations during the charge and discharge phases. The most common unsafe situations are cell under/over voltages, a battery current that exceeds the maximum permitted value and under/over temperature.

The microcontroller usually manages a dedicated integrated circuit (i.e., a battery stack monitor), which measures the voltage of the cells and the temperature in some points inside the battery. Sometimes, in very low-cost batteries, the stack monitor IC is replaced with discrete components to reduce the BMS cost, at the detriment of the measurement accuracy, occupied area and reliability. Anyway, the battery emulator should be able to work with both kinds of BMS.

A current sensor is connected in series to the protection switch to measure the battery current. It is usually realized with a shunt resistor. The voltage across its terminals is amplified and sampled by the Analog to Digital Converter (ADC) peripheral of the microcontroller. A digital peripheral is instead used when the current sensor directly provides a digital output. The battery should be able to communicate with the load and the charger, in order to optimize its usage by adjusting their power profiles. The communication protocol depends on the complexity of the system in which the battery is used. In very simple systems, the communication is missing or is limited to just an analog signal which encodes very limited information on the battery, e.g., its temperature. In more complex systems, a digital communication link, such as a serial communication or a CAN-bus, is adopted. Usually, the BMS is provided with a minimal user interface that allows the user to check the battery status, particularly the residual battery charge. In most cases, the user interface is very simple and consists of a push button to enable the visualization of the battery state of charge (SOC) and some LEDs, with which this information is encoded.

Another important block shown in Fig. 1 is the balancing circuit that allows the BMS to restore the possible imbalance in the charge of each cell due to the cell mismatches. An unbalanced condition reduces the total charge that can be

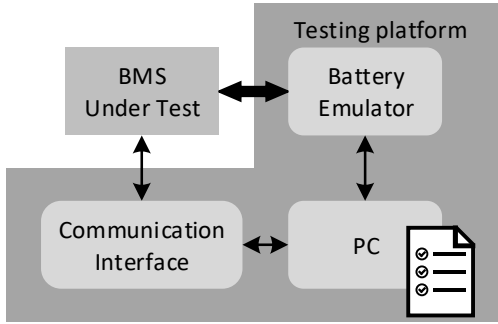


Fig. 2. Block diagram of the testing platform.

stored and delivered to the load. The reasons of the cell charge imbalance are widely studied in the literature [8]. The main cause is the difference in the self-discharge current of the cells, which may increase with aging. The main balancing techniques are divided in active and passive methods. In the first one, the battery is equalized by moving energy from the most charged cells to the less charged ones. In passive balancing, the extra energy stored in the cells with higher SOC is dissipated on a bleeding resistance. The power dissipated on the BMS board limits the balancing current to values in the order of 100 mA [9]. The balancing time is therefore rather large. In a BMS for medium-power applications, the balancing circuit is either omitted or the passive approach is adopted for the sake of cost reduction.

The BMS functions described above are summarized in Table I, where the corresponding most probable causes of fault have also been reported. An end-of-line testing platform should be able to check at least these functionalities, to guarantee the BMS quality before its connection to the cells in the battery assembly.

III. TESTING PLATFORM

The platform must be able to emulate the environment in which the BMS will be used, to perform the BMS functional testing in the end-of-line production phase. It applies the appropriate signals at the BMS external connectors, without the need of real batteries, loads and chargers. Fig. 2 shows the block diagram of the testing platform, where the battery emulator block reproduces the electrical behavior of the cells, of the load and of the charger. This block is controlled by a user interface developed in LabVIEW, which allows us to configure the battery emulator and to control the communication with the BMS through the communication interface block. Thus, the control interface automatically stimulates the BMS according to a user defined test procedure and checks the functionalities analyzing the BMS responses. The most important signals to be generated by the emulator are the cell voltages, with which the platform checks the cell voltage monitor functionality that also determines the activation of the protection switch. To check this safety function, the platform

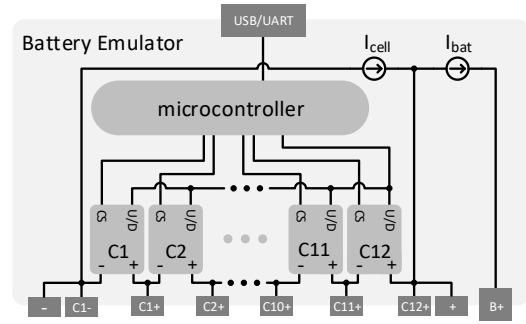


Fig. 3. Block diagram of the battery emulator.

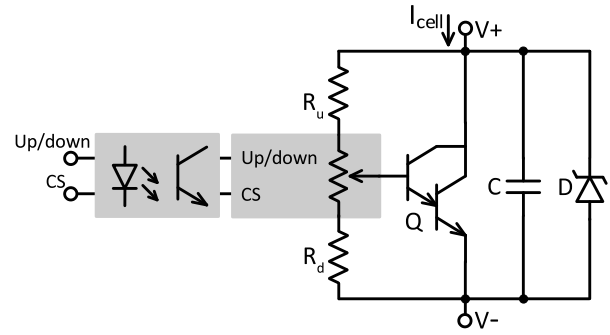


Fig. 4. Schematic of the stackable cell module.

must be able to generate an unsafe situation, for example when one or more cell voltages are outside the safe operating range of the lithium-ion cell chemistry adopted in that particular battery. The most common lithium-ion technologies have a cell voltage which varies in the range from 2.5 V to 4.2 V [10]. So, the platform must be able to vary the cell voltages in a rather large range, for example from 2 V to 4.5 V, to emulate any possible variation of the battery chemistry. For the same reason, the platform must be able to emulate the cell temperatures in a range larger than -10°C to 60°C , which is the typical operational range. The protection switch should also be activated in case of battery overcurrents, in both the charge and discharge phases. The overcurrent values depend on the cell capacity (up to 15 A h for medium power application) and thus it is up to 30 A/150 A, respectively, considering a maximum of 2C/10C charge/discharge rate.

To complete the BMS functional testing, the platform must also be able to verify the balancing circuit activation and functionality. Therefore, the emulated cells must source or drain the expected balancing current and provide, at the same time, a visible effect on the voltage values that validates the balancing circuit activation policy.

IV. BATTERY EMULATOR

The developed battery emulator is based on a modular approach, in which identical elemental cell emulators are stacked in series. Batteries characterized by a various number

of series connected cells and then by various terminal voltages can be emulated in this way. The block diagram of a 12-cell battery emulator is shown in Fig. 3, in which 12 stacked cell modules, an 8 bit microcontroller and two current generators are visible. The microcontroller manages the other blocks and communicates with the external LabVIEW interface through the USB port, which implements a virtual serial port.

The current generator I_{bat} is connected between the positive battery terminal (B+) and the positive power input of the BMS (+), so that the battery current measurement circuit and the functionality of the protection switch can be tested. In fact, the current value read by the BMS should be equal to the value set by the generator, if the protection switch is close. Otherwise, this value drops to 0.

The other current generator I_{cell} supplies the cell emulator modules, the circuit of which is shown in Fig. 4. It is based on a bipolar transistor V_{BE} multiplier circuit, in which the multiply factor, and thus the cell voltage, is selected by a digital potentiometer controlled by the microcontroller. The circuit is also provided with a capacity C to filter off the high frequency components of I_{cell} and with a Zener diode D , which is normally not active and acts as protection. In fact, the diode limits the cell voltage and protects the BMS monitor input in case of a hardware fault in the V_{BE} multiplier circuit. The cell voltage V_{cell} is calculated with the following equation:

$$V_{cell} = \frac{V_{BE}(R_u + R_{pot} + R_d)}{R_d + xR_{pot}} \quad (1)$$

where V_{BE} is the base-emitter voltage of the Darlington transistor Q , R_u and R_d are the resistances shown in the figure, R_{pot} is the nominal resistance of the digital potentiometer and x is the fraction of R_{pot} which is connected between the base of Q and R_d .

A Microchip Technology MCP4011 digital potentiometer is used. Its total resistance $R_{pot} = 2.1 \text{ k}\Omega$ is split in 64 steps that can be selected by means of two signals: the Chip Select (CS) and the Up/Down (U/D). These signals are driven by a 2-channel opto-isolator that allows us to stack the cell modules and to control them directly by the microcontroller general purpose input output (GPIOs) pins. The Up/Down signals are connected together and driven by one GPIO pin, while one or more cell modules are individually selected by the corresponding CS signal. The MCP4011 potentiometer can be directly supplied by the emulated cell voltage, avoiding an additional source. This feature strongly simplifies the battery emulator architecture and improves the solution used in [11], in which the cell modules need an isolated DC/DC converter to make them stackable. In [11], the cell is emulated by a Digital to Analog Converter (DAC) controlled through an isolated Serial Peripheral Interface (SPI) and powered by an isolated DC/DC converter. This solution is able to reproduce the cell voltage very accurately, but it is considerably more complex and expensive than our solution, so it seems less suitable for the considered applications.

The measurement of the battery temperature is another important task of the BMS that the platform should check. A

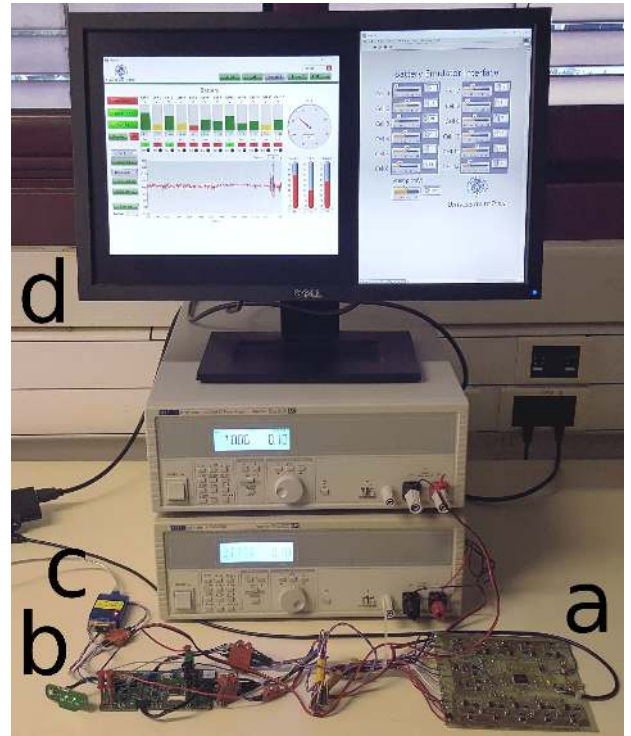


Fig. 5. A photo of the experimental setup, where the letter a indicates the battery emulator, b the BMS under test, c the communication interface, and d the control interface.

TABLE II
EXPERIMENTAL BATTERY EMULATOR COMPONENTS.

Component	Value	
Microcontroller	Atmel ATmega256A3BU	
I_{cell} generator	TTI QPX1200SP	
I_{bat} generator	TTI QPX1200SP	
Cell modules	R_u	1.8 k Ω
	R_{pot}	MCP4011
	R_d	1.5 k Ω
	Q	BCV49
	D	SMAZ5V1
	C	0.1 μ F

voltage that emulates the output of a voltage divider including a negative temperature coefficient (NTC) resistor is generated by the microcontroller DAC.

V. PLATFORM IMPLEMENTATION AND EXPERIMENTAL RESULTS

A 12-cell battery emulator and the relative control interface have been implemented and used to perform the functional test of the BMS used in a 48 V battery for gardening tools. Fig. 5 shows a photograph of the experimental setup.

A. Verification of the battery emulator

The battery emulator has been realized using a PCB fast prototyping service and the components reported in Table II. The comparison between the measured and the calculated voltage of a cell module is shown in Fig. 6 as a function of

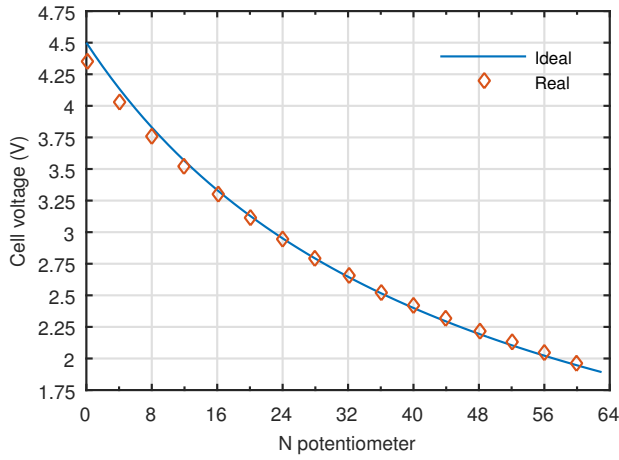


Fig. 6. Comparison between the cell voltage obtained by equation (1) and the measured one.

the potentiometer digital input signal. The calculated voltage is obtained by using (1) and the component values reported in Table II. The V_{BE} of the Darlington transistor Q is taken as 1.25 V. This value is obtained from the component datasheet with the collector current and temperature values measured when the cell module voltage was acquired, i.e., 100 mA and 50 °C. We can note from the figure an issue of the presented solution: the difference between the two voltages increases with the emulated voltage. In fact, the output voltage of the V_{BE} multiplier depends on the transistor temperature that changes with the power dissipated on the transistor and thus with the cell voltage itself. The consequent voltage inaccuracy does not impair the effectiveness of the platform because functional testing does not require a high precision in the voltage cell values, because its aim is rather checking the functionality of the BMS than the accuracy of the acquisition system. Moreover, the cell voltage is between 1.9 V and 4.5 V, a range large enough to exceed the limits that lead the BMS to activate the protection switch.

Another important characteristic of the cell module is the current that the module output is able to source/sink from the BMS. It is linked to the I_{cell} . The limit is set by the maximum collector current of Q. The chosen Darlington transistor withstands a maximum collector current of 500 mA, so the transistor and the PCB board must be able to dissipate a power up to 2.25 W. Therefore, the cell module can sink a current equal to the difference between the maximum collector current and I_{cell} .

A TTI QPX1200SP generator is used to provide the battery current I_{bat} . This generator does not completely cover the battery current range mentioned in Section III, but it allows us to generate a current up to 50 A. Should this value be not sufficient for the BMS to test, it can be replaced by a more powerful generator without changing the platform architecture and functionality.

B. Validation of a BMS for 12 cells

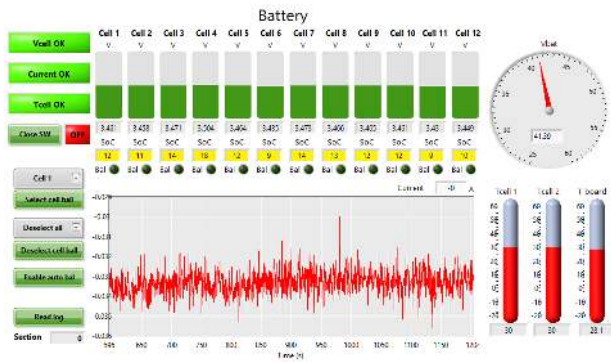
The experimental setup described above has been used to carry out all the functional tests described in Table I on a BMS developed for a 12-cell battery. The real battery cells are substituted by our battery emulator. Fig. 7 shows some results obtained during the test of the cell voltage acquisition function. It shows the interface which is used to communicate with the BMS and, at the same time, to control the battery emulator and set the cell voltage values. The interface shows the values and represents with graphical bars the measured quantities such as the voltages of the cells, the battery current and temperatures in three points, as generated by the battery emulator. The BMS communicates to the interface through the CAN-bus link, so the functionality of the communication port is also verified. Fig. 7(a) reports the case in which the BMS connected to the platform works well with an emulated battery that represents the normal behaviors of the cells. In fact, all the cell voltages read by the BMS show values compatible with those set by the battery emulator. Starting from this condition, another BMS unit, in which a short-circuit on the cell 4 input channel is created, is connected to the platform. The test result is shown in Fig. 7(b), where an incorrect reading of the fourth cell voltage occurs and the fault on the measuring channel is detected. The interface highlights the malfunction with a red indication on the problematic channel. It should be noted that connecting a real battery to this faulty BMS would have short circuited one of the cells with possible dangerous effect. Instead, the battery emulator safely reveals the malfunction.

The functionality of the BMS balancing circuit can also be tested by the platform. Let us assume a BMS balancing resistance R_{bal} of 33 Ω , I_{cell} of 100 mA and a cell voltage value higher than $R_{bal}I_{cell}$ (e.g., 4.1 V). The cell voltage becomes lower than 3.3 V when the BMS enables the balancing circuit. In fact, the BMS connects the bleeding resistance R_{bal} in parallel to the cell module. Thus, the biasing current I_{cell} , which flowed in the Darlington transistor Q before, is now split between the transistor and R_{bal} . This effect is shown in Fig. 8, where we see the voltage drop in the cell module after the balancing circuit is enabled. The experiment shows that the BMS is applying passive balancing in a correct way.

Let us now show the test of the battery current measurement function and the protection switch activity. I_{bat} is set to 1 A and the protection switch state is changed by sending a command through the BMS interface. The measured battery current shown in Fig. 9 demonstrates that the two functions work correctly. In fact, when the switch is closed the read value is equal to 1 A, otherwise the battery current goes to 0.

VI. CONCLUSIONS

This paper has presented the design of a battery emulator targeting the functional testing of BMSs for medium-power battery applications. The emulator is based on a custom board, combined with off-the-shelf power supplies. The design of the custom board is flexible and modular, as it can be configured to emulate batteries with a different number of series-connected cells, which is typically in the range between 6 and 16 for



(a)



(b)

Fig. 7. Comparison of the control interface in the cases of BMS without faults (a) and with the short-circuit on the cell 4 (b).

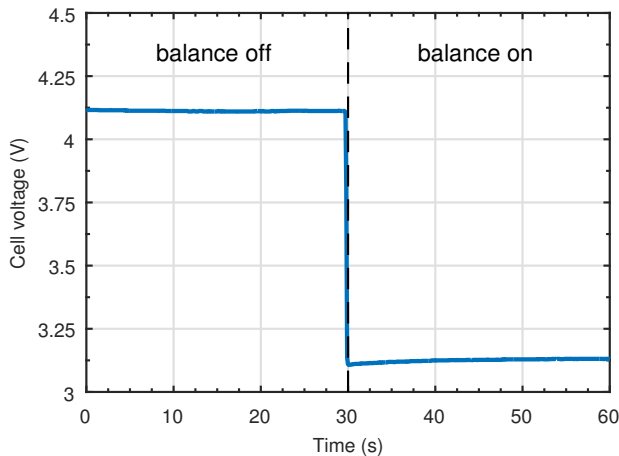


Fig. 8. Voltage of the cell module during the activation of the balancing circuit.

medium-power applications. A prototype of the custom board emulating a battery with 12 series-connected cells has been implemented and validated. We have shown that the designed emulator can effectively be used to test the typical functions of a BMS. The cost of the implemented custom board is below 50 €. The goal of designing a low-cost battery emulator for functional testing of a BMS at the end of the production line before assembling it to the battery is achieved.

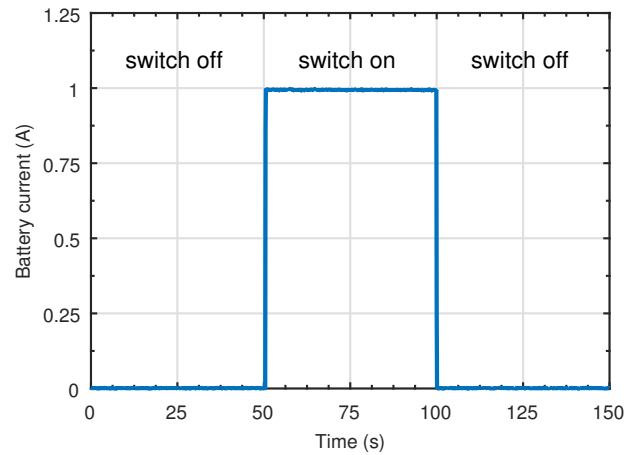


Fig. 9. Battery current during the functionality test of the protection switch.

REFERENCES

- [1] T. Letrouve, J. Pouget, and F. Chauvet, "Hardware-In-the-Loop Simulation: Hybrid Locomotive Energy Storage System Behavior Tests," in *2015 IEEE Vehicle Power and Propulsion Conference, VPPC 2015 - Proceedings*. IEEE, Oct 2015, pp. 1–6.
- [2] S. Abele and M. Weyrich, "A combined fault diagnosis and test case selection assistant for automotive end-of-line test systems," in *IEEE International Conference on Industrial Informatics (INDIN)*. IEEE, Jul 2017, pp. 1072–1077.
- [3] C. Fleischer, D. U. Sauer, J. V. Barreras, E. Schaltz, and A. E. Christensen, "Development of software and strategies for Battery Management System testing on HIL simulator," in *2016 11th International Conference on Ecological Vehicles and Renewable Energies, EVER 2016*. IEEE, Apr 2016, pp. 1–12.
- [4] Shuaijun Wu, Yijun Zou, Xin Peng, and Hongbiao Li, "Hardware-in-loop verification of battery management system with RT-LAB," in *2014 IEEE Conf. Expo Transp. Electr. Asia-Pacific (ITEC Asia-Pacific)*. IEEE, Aug. 2014, pp. 1–4.
- [5] J. V. Barreras, C. Fleischer, A. E. Christensen, M. Swierczynski, E. Schaltz, S. J. Andreasen, and D. U. Sauer, "An Advanced HIL Simulation Battery Model for Battery Management System Testing," *IEEE Trans. Ind. Appl.*, vol. 52, no. 6, pp. 5086–5099, Nov. 2016.
- [6] R. Morello, F. Baronti, X. Tian, T. Chau, R. Di Rienzo, R. Roncella, B. Jeppesen, W. H. Lin, T. Ikushima, and R. Saletti, "Hardware-in-the-loop simulation of FPGA-based state estimators for electric vehicle batteries," in *IEEE International Symposium on Industrial Electronics*, vol. 2016-Novem. IEEE, Jun 2016, pp. 280–285.
- [7] H. Rahimi-Eichi, U. Ojha, F. Baronti, and M.-Y. Chow, "Battery Management System: An Overview of Its Application in the Smart Grid and Electric Vehicles," *IEEE Ind. Electron. Mag.*, vol. 7, no. 2, pp. 4–16, 2013.
- [8] L. Rui, W. Lizhi, H. Xueli, D. Qiang, and Z. Jie, "A review of equalization topologies for lithium-ion battery packs," in *2015 34th Chinese Control Conference (CCC)*. IEEE, Jul 2015, pp. 7922–7927.
- [9] K. Vitols, "Design of an embedded battery management system with passive balancing," in *2014 6th European Embedded Design in Education and Research Conference (EDERC)*. IEEE, Sep 2014, pp. 142–146.
- [10] J.-M. Tarascon and M. Armand, "Issues and challenges facing rechargeable lithium batteries," *Nature*, vol. 414, no. 6861, pp. 359–367, Nov 2001.
- [11] H. Dai, X. Zhang, X. Wei, Z. Sun, J. Wang, and F. Hu, "Cell-BMS validation with a hardware-in-the-loop simulation of lithium-ion battery cells for electric vehicles," *Int. J. Electr. Power Energy Syst.*, vol. 52, pp. 174–184, Nov. 2013.
- [12] F. Baronti, G. Fantechi, R. Roncella, R. Saletti, G. Pede, and F. Vellucci, "Design of the battery management system of LiFePO4 batteries for electric off-road vehicles," in *2013 IEEE Int. Symp. Ind. Electron.* IEEE, 2013, pp. 1–6.