Framework to Test DC-DC Converters Developed for a Decentralized Battery Management System

Andrea Reindl*†, Thomas Singer*, Hans Meier*, Michael Niemetz*, Sangyoung Park†
*Faculty of Electrical Engineering and Information Technology, OTH Regensburg
andreareindl@ieee.org, thomas.singer@st.oth.regensburg.de, {hans.meier, michael.niemetz}@oth-regensburg.de
†Smart Mobility Systems, Technical University of Berlin
sangyoung.park@tu-berlin.de

2021 International Conference on Applied Electronics (AE) | 978-80-261-0973-0/21/$31.00 ©2021 | DOI: 10.23919/AE51540.2021.9542882

***Abstract*—DC-DC converters control the power flow and thus the power distribution between the components on different voltage levels. They are essential for (dis)charging batteries and influence the safety and stability of the entire battery management system (BMS). Therefore, testing the functionality and the reliability of DC-DC converters is crucial. This is especially true for decentralized battery management systems (DBMS), where multiple nodes communicate to collectively control the system. The used DC-DC converters are modified to parameterize them during operation via microcontroller interfaces. Integrating the communication into the control loop requires an analysis of the control behavior due to additional delays. Therefore, this paper proposes a framework to test DC-DC converters considering the control and communication perspectives. The response time, the control accuracy and stability of these DC-DC converters, e.g., under continuous and abrupt load changes, are measured in automated tests. The dedicated software framework simulates the DBMS and stimulates the hardware components (e.g. electronic loads, data acquisition) via respective interfaces (CAN, RS232). This allows the test of various DC-DC converters with flexibly adaptable load and power generation profiles. An initial application validates the test framework by verifying the aforementioned aspects and thus the applicability of a DC-DC converter within the DBMS.**

***Index Terms*—Battery management system, decentralized control, DC-DC converter, system testing, Python, hardware in the loop testing, object oriented programming**

I. Introduction

Battery management systems (BMS) administer system control and management with regard to energy storage and transmission [1]. Decentralized BMS (DBMS) are characterized by autonomous and locally operating control units attached to each battery. They offer significant advantages including scalability, easier maintenance and integration. By placing the sensing and control units closer to the battery and avoiding the dependence on a central master controller, potential single points of failures are minimized. An own approach of a DBMS was proposed [2], which also involves electrical sources and loads in addition to the batteries, in contrast to existing approaches [3, 4, 5]. All components are connected by a power line and at least one global communication line (Fig. 1).

For distributed control, each component is equipped with its own Local Control Unit (LCU), which includes:

- Current, voltage and temperature sensors,
- communication interfaces for local and global data exchange,
- a microcontroller for calculation and data management,
- a DC-DC converter with adjustable current and voltage limits by a microcontroller interface and
- a relay which is opened in case of failures or for maintenance.

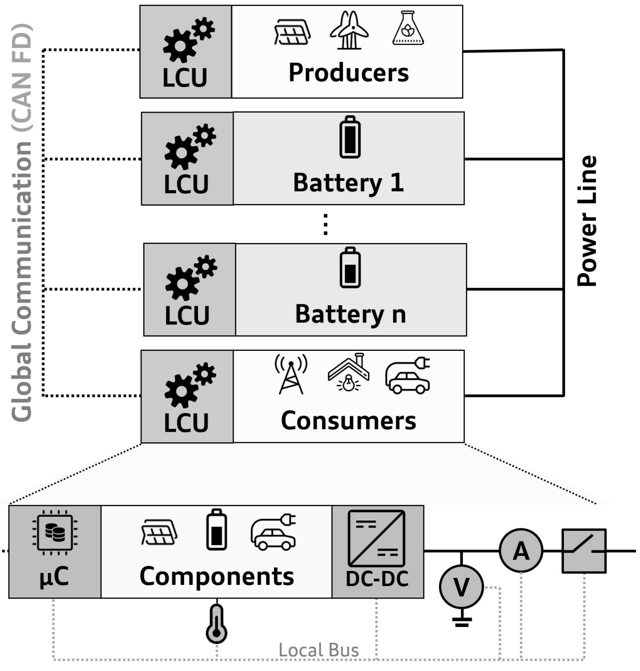

Fig. 1. Overview of the decentralized battery management system (top) and detailed presentation of a component with a local control unit (LCU) (bottom).

One of the battery nodes coordinates the global control. It specifies the required (dis)charge power of the remaining battery nodes, which manage their local control loops.

The DC-DC converters work as actuators and provide the power distribution between the components at different voltage levels [7]. They are essential for the (dis)charge of the batteries and influence the control stability of the overall system [8]. The DC-DC converters operating in this DBMS need to be validated under a range of conditions to ensure safety and reliability [8, 9]. They have to achieve a stable output voltage and a precise load current distribution. The response time is crucial for the global control and has to be tested. It is composed of

- the transmission of the setpoints via CAN (FD) messages,
- the adjustment of the control variables of the DC-DC converter by the microcontroller and
- the propagation delay of the DC-DC converter.

Comparison to Related Work

While there are numerous DC-DC converter test frameworks, it is difficult to find one that encompasses the aspects of networked control. Delays in the global control messages and possible data dropouts have a significant

ISBN 978-80-261-0973-0, © University of West Bohemia, 2021

impact on the functionality and reliability. Consequently, the cooperatively control has to be considered. Application and hardware-specific additional interfaces are required to connect the hardware [17, 18]. Existing test approaches often need dedicated hardware-in-the-loop platforms such as dSPACE [12, 13, 14], where the underlying models are programmed in MATLAB/Simulink [15] or LabVIEW [16]. In contrast, Python, an established object-oriented open-source language, integrates the capabilities of simulation and testing of hardware components in one system [19].

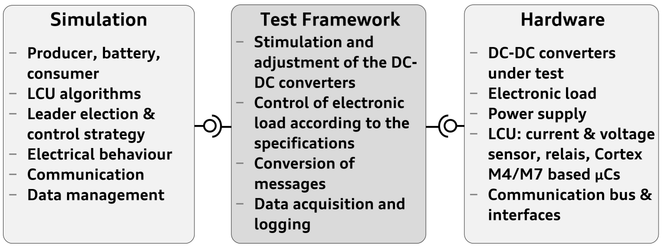
Simulation
– Producer, battery, consumer
– LCU algorithms
– Leader election & control strategy
– Electrical behaviour
– Communication
– Data management
Test Framework
– Stimulation and adjustment of the DC-DC converters
– Control of electronic load according to the specifications
– Conversion of messages
– Data acquisition and logging
Hardware
– DC-DC converters under test
– Electronic load
– Power supply
– LCU: current & voltage sensor, relais, Cortex M4/M7 based µCs
– Communication bus & interfaces

Fig. 2. Functional description and correlation of the simulation [11], the test framework and the hardware.

Contributions

In order to address the challenges, a test framework fulfilling the following requirements is needed. It considers the coupled control and communication perspectives to determine

- the response time,
- the control accuracy and
- the stability

of the DC-DC converters in various test scenarios including continuous and abrupt load changes realized by an electronic load. The support of automated tests is required to realize the comparison between different DC-DC converters and to evaluate their usability within the DBMS. Due to the digital control via the microcontroller and the customizable test framework, it is possible to apply different and multiple DC-DC converters simultaneously to investigate the interaction and load distribution among them.

Within a previously developed simulation [11], the DBMS can be effectively modified corresponding to various applications. The test framework has to provide the stimulation of the test case simulation hardware, including the electronic load and the DC-DC converters under test (DUT), according to the simulation data (Fig. 2). The proposed Python-based test framework combines the flexibility of a software simulation [11] with the improved results by using real hardware components to test networked and digitally controlled DC-DC converters within a DBMS.

Organization

The requirements of the test framework are analysed in Section II. In Section III, the architecture including the used hardware components, the developed software design and the applied test algorithm are described. To validate the proposed test framework, initial test results are shown and discussed in Section IV. The control stability and accuracy under load changes and the reaction speed of a DC-DC converter are thereby investigated. Finally a conclusion is drawn and possible future extensions and further developments are discussed.

II. Requirements Analysis for the Test Framework

A. Hardware Control According to the Simulation Data

For validation and verification of the DC-DC converters, a previously developed DBMS simulation [11] is used: The electrical behavior, the data communication and the LCUs of the DBMS are simulated and provide different test scenarios. A connection between the BMS simulation software and the hardware under test is required, driving the actuators and recording measured physical values (Fig. 2).

B. Scalability and Flexibility of the Test Framework

For different use cases and test scenarios, application-specific load profiles are required and the load sharing between DC-DC converters of the same or of different types has to be supported (Figs. 3 and 4). Comparing response time, control accuracy and stability enables the evaluation of the DUT in terms of its usability within a DBMS.

C. Various Test Scenarios for Improved Testability with Adjustable Profiles

In order to validate and verify the function of the DC-DC converters for a wide range of applications and operating modes, variable test scenarios are required. For an initial design, the response time and the control stability of the DC-DC converter are tested under continuous and abrupt, transient load changes (Fig. 8). The number and magnitude of the load changes depends on the properties of the DUT. Therefore, the test framework needs to apply adjustable load profiles while managing, processing and analyzing the measured physical values.

III. System Architecture of the Test Framework

The proposed test framework consists of hardware and software components (Fig. 4).

A. Hardware Components

1) Local Control Unit and Communication Interface

The LCU provides current and voltage sensors, a relay, a communication interface and a microcontroller (Fig. 1). The microcontroller receives the current and voltage target values via the communication line, controls the DC-DC converter according to the specifications and records the actual status by reading out the sensors. To verify the accuracy and stability of the current and voltage control of the DUT, corresponding sensors are located at both sides (Fig. 4). For the logging and evaluation of the results, the microcontroller sends the raw data with a time stamp back via the global communication interface.

The proposed framework implements testing of several DC-DC converters communicating over the network and controlling the system in a collaborative manner. Communication-based load sharing is a non-trivial task which requires a holistic testbed to ensure the safe and reliable operation of DC-DC

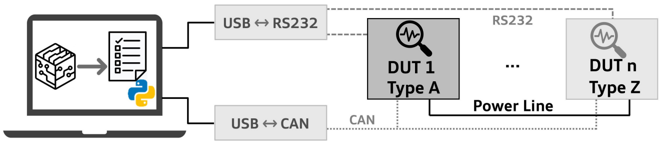

Fig. 3. Test architecture for several DUTs, possibly of mixed types.

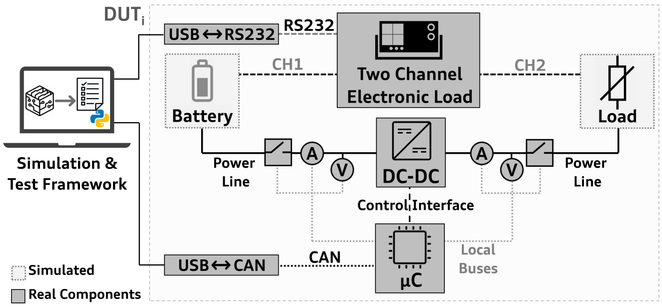

Fig. 4. Connection of a hardware DUT$_i$ to the software simulation.

converters within the DBMS. To realize the communication between test software (PC) and the microcontroller with a CAN bus a commercial USB-to-CAN interface [20] is used (Figs. 3 and 4). CAN (FD) messages enable the interaction between the LCUs (Fig. 1) and between the test software and the DUTs (Fig. 7) [6]. This enables the investigation of the condition-dependent, communication-based load sharing among the DUTs. The microcrontroller provides a wide range of control interfaces, including SPI, UART, I^{2}C and PWM signals, enabling effective adaptation to different DUTs and thus test scenarios for various applications (Fig. 4).

2) Programmable, Electronic Load

A programmable electronic load is used to test DC-DC converters under real conditions (Fig. 4) by working as power drain or source, depending on the test case. Each of the two channels provides a power (producer) and a load (consumer) mode. The operating modes *load* and *power* are adjustable via software commands and can be changed during operation. Associated Standard Commands for Programmable Instruments (SCPIs) (Fig. 6), a specific language for the control of programmable instruments standardized by IEEE 488.2 [22], are sent via the serial interface RS232 to the electronic load.

The dual power flow direction allows the testing of bidirectional DC-DC converters. Limiting DC-DC parameters (max. voltage/current) according to the DC-DC specifications are observed. The electronic load offers sequence functions to create individual voltage or power supply profiles. Pre-programmed templates can be used, whereby parameters for maximum and minimum limits, starting point and duration can be adjusted. Additionally individual points of the sequence can be set and arbitrary curves can be composed. For the effective generation of load and charge profiles an own special library (eLoadLib) is developed (Fig. 7). Every DUT requires such an electronic load, but there is no restriction in attaching additional DUTs.

B. Software Design

The software of the test framework bridges the simulation and the hardware components (Fig. 7). Fig. 5 shows an overview of the implemented and used classes.

1) eLoadLib: Library for Controlling the Programmable, Electronic Load

In order to effectively control the electronic load, the library eLoadLib was implemented, which converts the specifications from the simulation or a measurement series into SCPI (Fig. 6). For example, the command :MEAS2:VOLT? is used for the output voltage measurement on channel 2. For the efficient use of the electronic load, the library eLoadLib provides a number of functions including set, get, add and measure functions and in addition to that error handling mechanism. According to the call of the function and the optional parameters, the corresponding SCPI instructions are generated and sent. Addressing a COM port is thereby implemented with the Python library pySerial [23]. Error handling ensures valid parameter setting and correct command reception. Functions of the library eLoadLib allow the setting and querying of parameters of the electronic load.

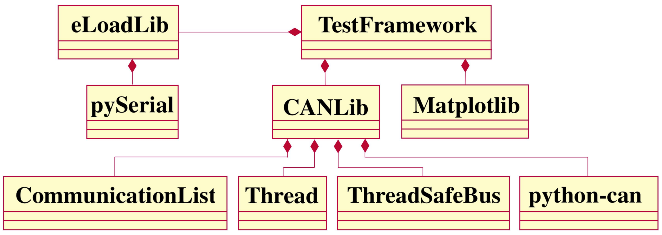

Fig. 5. Software design of the test framework.

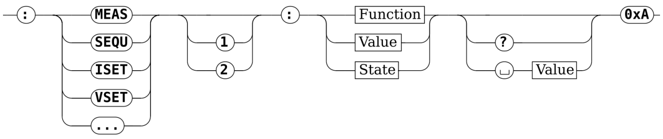

Fig. 6. Extract and syntax example of the SCPI to configure the electronic load

2) CAN Library for the Implementation of a Communication Interface to the Local Control Unit

CAN (FD) is used for communication with the LCU in the initial tests, whereby also additional communication interfaces are provided. The developed library CANLib establishes a connection between the test environment and the USB-to-CAN interface and organizes the sending and receiving of CAN messages. It provides the communication of the current and voltage target values to the LCUs and the actual current and voltage values at the input and output side of the DUTs to the test framework.
It contains two objects of the class CommunicationList to extend the receive and transmit FIFOs (Rx-, Tx-FIFOs) within the CAN controller to avoid data overflow. Separate thread objects send, receive and process messages. This ensures that the Rx- and Tx-FIFOs are regularly and independently updated.

The library python-can allows several threads to access an object and manages multiple accesses to global variables [24]. The use of threads optimally coordinates the timing and the exchange of messages. Additionally, the CANLib offers an error tracking function to check if the message was received correctly, taking into account the expected value, parameters and processing.

3) Use of Real or Simulated Data

Besides data from the simulation of the DBMS also load data of real consumers and batteries can be imported to test the DUT. Files in a .txt format are used for an efficient data exchange e.g., the actual load value or the (dis)charge power of the components. An own transmission protocol is defined (Tab. 1). It contains the functions of the electronic load and the setting options of the DUTs in a compact manner.

Tab. 1 Instructions and examples of the transmission protocol

Instruction	Description
PWR	Current (C) and voltage (V) setting with optional over voltage (OVP) and over current (OCP) protection for each channel (CH) Example: CH V C OVP OCP
LOAD	Load setting with optional OVP and OCP Example: CH MODE VAL OVP OCP
SEQ POINT	Setting of individual points in the sequence Example: CH NR V C TIME
SEQ TEMP	Parameterization of templates Example: OBJ CH START POINTS TIME MIN MAX TYPE INV/RATE SYM WIDTH
DCDC	Setting of output voltage and current of the DC-DC converter Example: ID V C OUT

For example, the instruction `DCDC 1 24 1.5 True` activates the DC-DC converter with the ID 0x01, sets the output voltage to 24 V and the output current to 1.5 A. Based on the custom defined syntax, corresponding functions are called. The respective SCPI commands are generated and executed. The class `TestFramework` contains objects of the class `eLoadLib` and `CANLib` (Fig. 5). It provides the interaction between the simulation and the hardware components (Fig. 7).

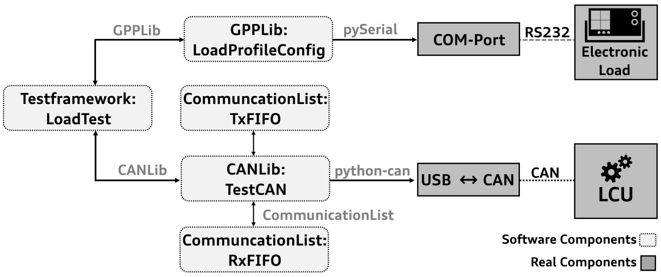

Fig. 7. Overview of the software and hardware components of the test framework

IV. Test Scenarios for an Initial Validation and Verification of the DC-DC Converters

A. Test algorithm and Processing of the Results

In this section, the framework is verified and validated with test scenarios. In order to achieve this, both independent measuring units of the electronic load and of the LCU record the actual sensor data using the corresponding `measure` functions. The LCU periodically polls the current and voltage sensors by default and reports all results with time stamps via CAN messages. Selected parameters to be additionally measured can be individually adapted and extended. After completion of a test run, the data are summarized in graphs to display the results and saved in a .csv file [25].

B. Implemented Test Scenarios

In order to evaluate DC-DC converters regarding their usability in a DBMS three different test scenarios were developed initially. In order to fully validate DC-DC converters within the DBMS, comparative tests have to be performed for all operating modes, including system startup and safe shutdown, and for various load profiles. However, since the scope of this paper is to introduce the setup itself capable of testing multiple DC-DC converters cooperating over a communication network, complete testing results will be provided in a future work.

1) Response Time Test

The reaction speed of the DC-DC converter directly influences the control within the DBMS and is therefore a relevant evaluation criterion. It is tested how fast the DC-DC converter reacts to a command transmitted via the CAN line and how long it takes to reach the desired output power. The generation, transmission and processing of CAN messages were measured in a previous work [6].

2) Load Test

The stability and accuracy of the power control by the DC-DC converter is a prerequisite for a fair, condition-based power distribution. Neither abrupt nor continuous load changes should influence the control of the DC-DC converter. Therefore, the control behavior is tested with abrupt and continuous load changes.

3) Supply Voltage Tests

Drops in the input voltage may also affect the stability and accuracy of the control and the output voltage. Therefore, the control of the DC-DC converter is investigated in case of supply voltage changes. Also in this case continuous as well as abrupt changes are considered.

V. Validation of the Test Framework

Initial tests to validate the framework have already been performed. The overall setup of the first test consists of the DC-DC converter to be tested, in this case a unidirectional, combined buck and boost converter [19], the LCU designed as a plug-on board, the electronic load, power lines, a CAN line with CAN-to-USB interface and a PC (Fig. 15). The DC-DC converter under test was adapted according to the requirements of the networked, communication-based control and is steered by PWM signals.

Figs. 8 - 13 show the test results for a load test with continuous load change. Fixed setpoints are specified via the CAN line: 24 V output voltage and a current limitation of 1 A. It is tested whether the DC-DC converter regulates to the specified target values and whether the control remains stable even with load changes. The control of this DC-DC converter remains stable, with output voltage values fluctuating in a range between 24.06 V and 24.05 V (Fig.12). Current and voltage measurements include both the electronic load measurements (solid line) and the LCU measurements (marked by crosses). Fig. 9 compares the input and output power, each calculated from the current and voltage values measured by the LCU, to determine the efficiency of the DC-DC converter.

Fig. 14 shows the measurement of the response time of the voltage control considering the networked control. The DC-DC converter is supplied with a constant voltage of 12 V and a current of 2 A during the entire test. The load is constantly set to 150 Ω. Initially, the target voltage of the DC-DC converter is set to 10 V and the current limit to 0.5 A. To measure the response time, a new target voltage of 20 V is specified via the CAN line. Fig. 14 shows that the DC-DC converter has regulated to the new target voltage approx. 0.12 s after the start of the transmission of the respective command via the

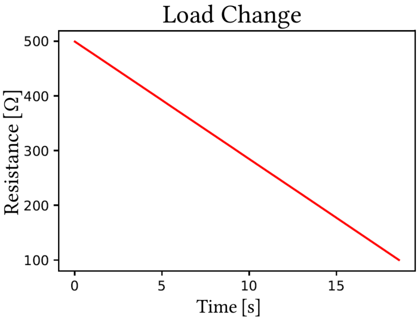

Fig. 8. Load test with continuous load change

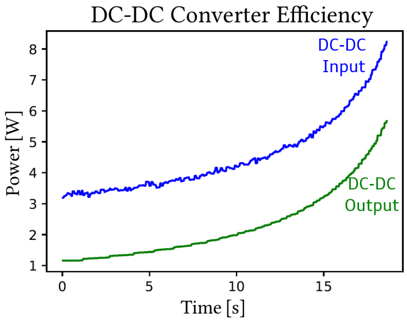

Fig. 9. Efficiency of the DC-DC converter [26]

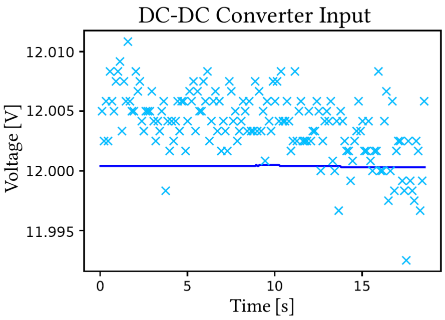

Fig. 10. Voltage at the input of the DUT [26]

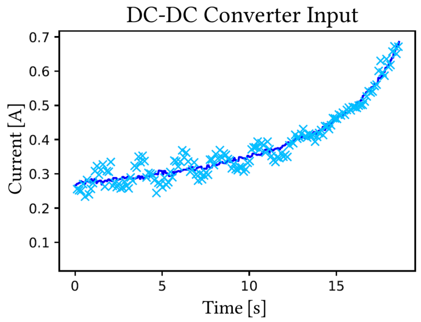

Fig. 11. Current at the input of the DUT [26]

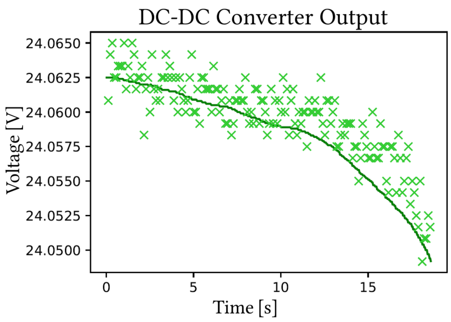

Fig. 12. Voltage at the output of the DUT [26]

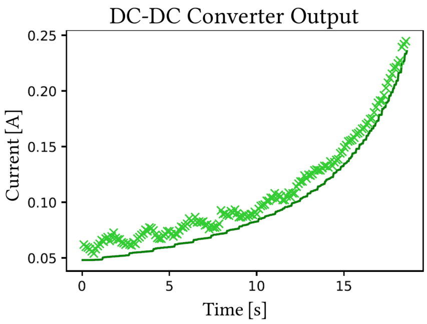

Fig. 13. Current at the output of the DUT [26]

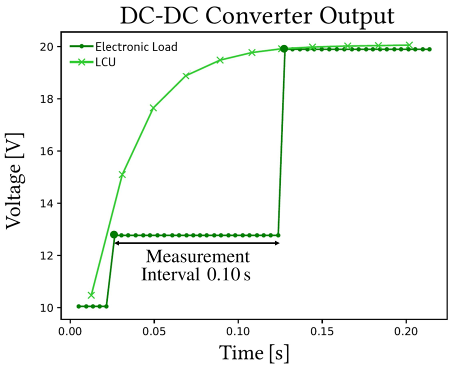

Fig. 14. Measurement of the response time considering the networked control of the DC-DC converter[26] .

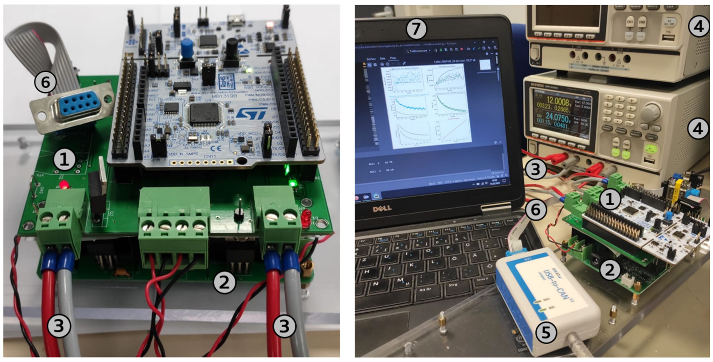

Fig. 15. The test setup consists of the LCU using a Cortex-M4 microcontroller ①, the DC-DC converter to be tested [26] ②, which is connected to the electronic load ④ via power lines ③. A CAN-to-USB interface ⑤ and a CAN line ⑥ are used for data transfer between the PC ⑦ and the DUT ①–④.

CAN line. The solid line shows the measurements of the electronic load, and the crosses show the ones of the LCUs. It is clearly shown that the LCUs measure more frequently and more precisely and also detect short-term changes, enhancing functional safety. The measuring units of the electronic load scan the values less frequently and send previous values, which leads to partly considerable deviations between the measured values of the LCUs and the electronic load.

Furthermore, another DC-DC converter [27], a unidirectional buck converter, was tested (Fig. 18) under an abrupt load jump from 150 to 200 Ω (Fig. 16). The DC-DC converter was also adapted to the communication-based control, with output current and voltage adjustable via UART (Fig. 18). The input voltage of the DC-DC converter during the test was 20 V and the input current 1.5 A. The output voltage was set to 10 V and the output current to 1 A. After the load jump occurred, the voltage increased slightly with a maximum change of 0.01 V (Fig. 17). This voltage change barely influences the control and therefore this DC-DC converter passed the load jump test. Furthermore, only the measurement results of the LCU were used in this test due to the high measurement interval of the electronic load and the resulting limited significance of these measurements.

The results show that the framework is capable of validating all essential functionalities of different DUTs including the control accuracy, the control stability under changing load and the response time considering the networked control. The consideration of the mentioned criteria enables an evaluation of the respective DC-DC converter regarding the reliability and the usability within the DBMS. Furthermore, the results confirm the communication, the adjustment of the LCUs' target values, the stimulation of the hardware components including the DUTs and the electronic load, and the acquisition and evaluation of the actual current and voltage values. This forms the basis for automated, comparative testing of several different DC-DC converters and for testing the communication-based power distribution among several DC-DC converters.

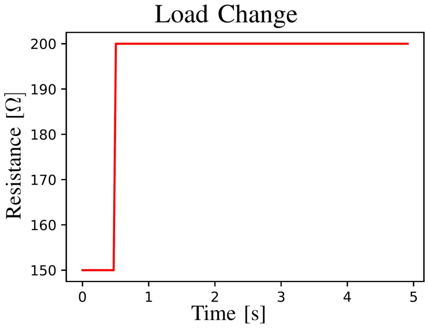

Fig. 16. Load test with abrupt load jump

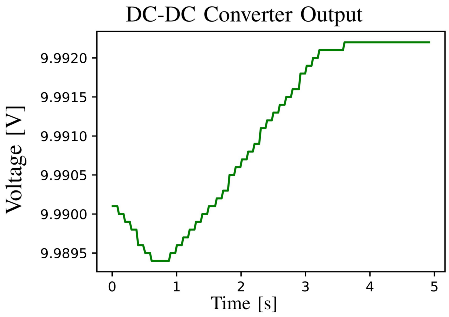

Fig. 17. Voltage at the output of the DUT [27]

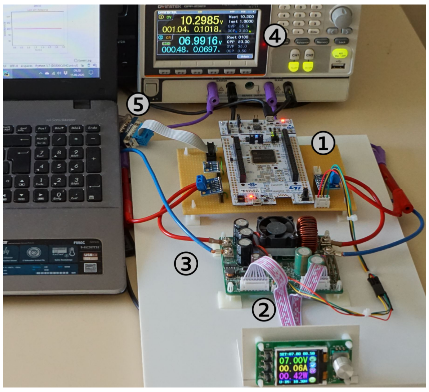

Fig. 18. The Cortex-M7 based LCU ① is attached to the DC-DC converter [27] under test ②.

VI. Conclusion and Outlook

DC-DC converters form the basis for the decentralized control of the DBMS by controlling the flow and distribution of power. Therefore, a customized framework to test DC-DC converter for the DBMS was proposed, which takes into account the aspects of the communication based, digital control among the DUTs. The test framework combines software simulation and real hardware components, which are stimulated according to the simulation data. The software simulation enables the adaption to various applications by supporting a variable number of batteries, different battery technologies, adjustable electronic loads and sources. The test framework evaluates the response time, the control accuracy and the stability of the DUTs and allows to assess the reliability and usability of the DUTs within the DBMS. Initial test scenarios validated and verified the test framework, successfully proving the communication, the interaction between software and hardware as well as the acquisition and analysis of the actual data. In further tests the simultaneous testing of several and different DC-DC converters will be validated. As a future work, the test framework is extended for testing a DC-DC converter with serial interface [28] and a bidirectional DC-DC converter [29].

Acknowledgment

The authors thank M. Farmbauer and C. Schimpfle for helpful discussions.

References

[1] M. A. Hannan et al., ”State-of-the-Art and Energy Management System of Lithium-Ion Batteries in Electric Vehicle Applications: Issues and Recommendations”, IEEE Access, vol. 6, pp. 19362-19378, 2018.

[2] A. Reindl, H. Meier, M. Niemetz, ”Scalable, Decentralized Battery Management System Based on Self-Organizing Nodes”, ARCS 2020: Int. Conference on Architecture of Computing Systems, Springer, 2020.

[3] D. F. Frost, D. A. Howey, “Completely Decentralized Active Balancing Battery Management System”, IEEE Trans. on Power Electronics, pp. 729-738, 2018.

[4] S. M. Chowdhury et al.,“A novel battery management system using a duality of the adaptive droop control theory”, IEEE Energy Conversion Congress, pp. 5164-5169, 2017.

[5] S. Steinhorst et al., “Smart Cells for Embedded Battery Management”, IEEE Int. Conf. on Cyber-Physical Systems, pp. 59–64, 2014.

[6] A. Reindl, D. Wetzel, N. Balbierer, H. Meier, M. Niemetz, S. Park, “Comparative Analysis of CAN, CAN FD and Ethernet for Networked Control Systems”, embedded world conference digital, 2021.

[7] J. Timmermans et al., ”Test platform for hybrid electric power systems: Development of a HIL test platform”, European Conf. on Power Electronics and Applications, pp. 1-7, 2007.

[8] L. Mihet-Popa, V. Groza, ”Battery management system test platform developed for electric vehicle applications”, 2015 IEEE 9th Int. Symposium on Intelligent Signal Processing, pp. 1-6, 2015.

[9] Y. Li, Z. Sun, J. Wang, ”Design for battery management system hardware-in-loop test platform”, 2009 9th Int. Conf. on Electronic Measurement & Instruments, Beijing, pp. 3-399-3-402, 2009.

[10] M. Winter, ”Zur Optimierung und Validierung von Managementsystemen für elektrische Energiebordnetze im Kraftfahrzeug”, Dissertation, Technische Universität München, 2019.

[11] A. Reindl, H. Meier, M. Niemetz, ”Software Framework for the Simulation of a Decentralized Battery Management System Consisting of Intelligent Battery Cells”, 2019 IEEE Student Conf. on Research and Development, pp. 75-80, 2019.

[12] R. Subramanian, P. Venhovens, B. P. Keane, ”Accelerated design and optimization of battery management systems using HIL simulation and Rapid Control Prototyping”, IEEE Int. Electric Vehicle Conf., pp. 1-5, 2012.

[13] S. Chakraborty et al., ”Scalable Modeling Approach and Robust Hardware-in-the-Loop Testing of an Optimized Interleaved Bidirectional HV DC/DC Converter for Electric Vehicle Drivetrains”, IEEE Access, pp. 115515-115536, 2020.

[14] J. V. Barreras et al., ”Functional analysis of Battery Management Systems using multi-cell HIL simulator”, 10th Int. Conf. on Ecological Vehicles and Renewable Energies, pp. 1-10, 2015.

[15] B. Tabbache et al., ”A simple and effective hardware-in-the-loop simulation platform for urban electric vehicles”, 1st Int. Conf. on Renewable Energies and Vehicular Technology, pp. 251-255, 2012.

[16] W. C. Lee, D. Drury, ”Development of a Hardware-in-the-Loop Simulation System for Testing Cell Balancing Circuits”, IEEE Trans. on Power Electronics, pp. 5949-5959, 2013.

[17] J. Stefan et al., ”A model-based interfacing concept for accurate power hardware-in-the-loop systems”, Mathematical and Computer Modelling of Dynamical Systems, pp. 1-20, 2016.

[18] S. Maxwell et al., ”Capability, compatibility, and usability evaluation of hardware-in-the-loop platforms for DC-DC converter”, IEEE Energy Conversion Congress and Exposition, pp. 1-6, 2016.

[19] T. Singer, ”Entwicklung einer Testumgebung für einen CAN-basierten DC-DC Wandler zur Validierung und Optimierung verschiedener Last- und Ladeprofile eines Batteriemanagementsystems”, bachelor thesis, Labor für industrielle Elektronik, OTH Regensburg, 2020.

[20] CAN-USB Adapter https://www.ixxat.com

[21] Multi-Output Programmable DC Power Supply GPP-2323 datasheet. https://www.calplus.de/gw-instek-gpp-2323.html

[22] Standard Digital Interface for Programmable Instrumentation - IEEE Std 488.2-1992

[23] pySerial 3.0, https://pythonhosted.org/pyserial/pyserial.html

[24] python-can 3.3.2, https://python-can.readthedocs.io/en/master/

[25] matplotlib 3.3.0, https://matplotlib.org/Matplotlib.pdf

[26] MingHe - D3806: Buck and Boost Modul

[27] Joy-it, ”DPS 5015 - Programmable Power Supply, Step Down Converter”, datasheet, Sep. 2019.

[28] S. Träger, ”Testsytem zur mikrocontrollerbasierten Ansteuerung mehrerer DC/DC-Wandler eines Batteriemanagementsystems mittels CAN, PWM und RS232”, master thesis, Labor für industrielle Elektronik, OTH Regensburg, 2020.

[29] M. Jupke, A. Reindl, H. Meier, M. Niemetz, ”Bidirectional DC-DC Converter with Digital Droop Parametrization”, 26th International Conference on Applied Electronics (AE), 2021.