Pau Añó Martínez

Balancing Techniques for Battery modules

Master Thesis

Directed by Dr. Enric Vidal Idiarte

Master in Electric Vehicle Technologies

UNIVERSITAT ROVIRA I VIRGILI

Tarragona
2023

Index

1 Introduction

1.1 Objective

The objective of the thesis is to implement and test different Balancing techniques into a simulated Battery Module to establish a comparison between methods. All the simulations, and the implementation of the different balancing techniques are performed into MATLAB environment using Simulink.

To perform the test of the different techniques, a practical MATLAB project environment has been created, allowing the engineer to perform tests with different parameters with ease.

The project will be focused on balancing implementation into a module level with a distributed BMS system.

After performing the different simulations, a series of conclusions will be made comparing the different methods.

1.2 Motivation

Currently, Battery Systems is one of the most growing topics in the automotive industry. OEMs, as per the restrictions that EU has imposed to ICE vehicles, are switching to Electric Vehicle (EV) technologies to reduce the carbon footprint on their manufactured cars.

One of the problematics that make consumers afraid of buying EVs, is range. Battery engineers have different ways to increase the range of an EV. In an **extremely** simplified way, it can be resumed with its pros and drawbacks:

- Improving Battery Cells. New materials, and new processes are currently being developed to get the maximum energy capacity with the farthest EOL[1] possible. This requires complex R&D. New technologies are kept as trade secrets by the cell's manufacturers.
- Increase Battery Pack size. This could be seen as a direct increase on the range. However, as the battery gets bigger, the weight of the car increases, so the energy consumption of the vehicle increases in concordance. Furthermore, there is a price increase due to the addition of more cells into the vehicle.
- SOC[2] estimation and increasing battery range. This is one of the fields that manufacturers are more focused on. Battery range can be increased with an improved SOC estimation process and by increasing the Voltage range that is considered operational. This is an action that some OEMs perform by updating the vehicle's software. Usually, it is performed after the vehicle has been on the roads for some years, as all the collected data from those vehicles can be analysed and changes on the Software can be derived from these data.
- Efficiency. The ability to get the maximum effective energy in a battery pack with the minimum losses permits to increase the range. Cell balancing as will be seen on further sections of the project, is a technology that does so.

As an Electronic engineer I have knowledge about Electronic Hardware, Low Level Software and Circuit Theory. Among all the topics related with Battery Systems I feel that Battery Balancing is one in which I could apport with my current knowledge.

[1] End of Life – Time or kilometres that the battery could support without having a considerable age phenomenon.
[2] State Of Charge – Proportion of the Battery capacity remaining in respect with its maximum value. Can be expressed in (0% - 100%) or in a (0-1) range.

2 EV Battery Basics

2.1 Cell

A Cell is the most basic element of a battery. It relies into an Oxidation-Reduction reaction to store energy inside of it. Depending on the chemistry and its construction, the voltage range of the cell can vary.

Manufacturers usually provide the following parameters on the datasheets of the cells:

- Nominal Voltage of the cell.
- Maximum and minimum operational Voltages of the cell.
- Nominal Capacity in Ah (definition in section 2.1.3.2).
- Internal resistance at 25 ºC. A value in $m\Omega$ of the maximum value of internal resistance that a cell can have after the end of line test at a temperature of 25 ºC.
- Operational temperature range. The range of temperatures in which the cell can operate safely.
- VOC-SOC table. This table expresses a relationship between the remaining charge of the cell and the voltage measured on its terminals. This table may be not provided to public as it's a more technical parameter.

In following sections, an introduction of the basics of battery cells will be presented. Also, some of the parameters seen on these points are further explained.

2.1.1 Lithium-Ion

Cells based into Lithium-Ion electrolyte are the most common used on the market as per their excellent characteristics. Following are some of the characteristics that make Lithium-Ion cells the most common used:

- Energy density. Parameter expressed in kWh/kg. Gives an indication of the capability of the cell to store energy per mass unit.
- Volumetric Energy density. Parameter expressed in kWh/m^3. Gives an indication of the capability of the cell to store energy per volume unit.
- Coulombic efficiency close to the unity. Defined as the efficiency that the cell has in transferring its charge to current (when is giving away charge) and when its recombining electrons (when is being charged).
- Low Hysteresis Phenomena. Defined as the voltage difference seen between charge and discharge on a same SOC value.
- End of life. The number of operational cycles (charge - discharge) that the cell can operate keeping a remaining 80 % of capacity.

One of the characteristics that is relevant for this project and with electronics working with batteries, is the characteristic *Open Circuit Voltage - SOC* curve of the Cell. Each chemistry present different type of curve. Lithium ion presents a flat curve in the middle part of SOC.

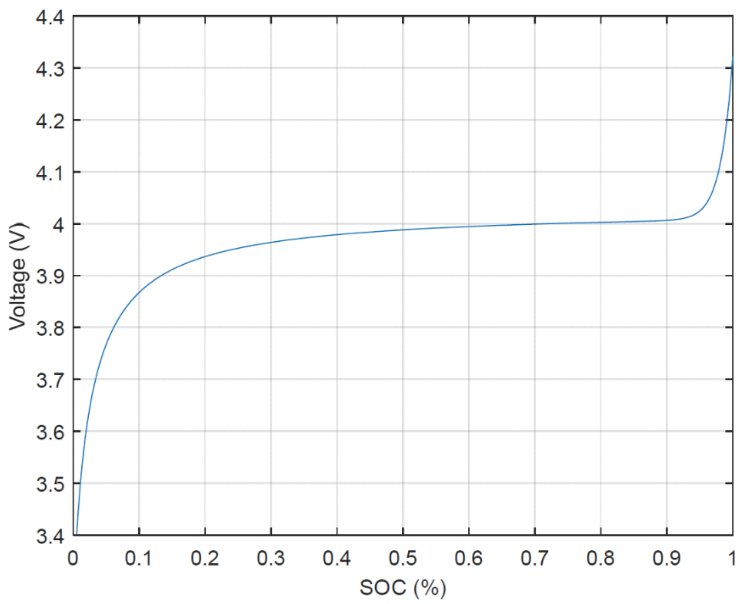


Figure 1. Characteristic Open Circuit Voltage[3]-SOC of lithium Ion Cells.

As can be seen on Figure 1, the voltage curve remains flat for most part of the area between 25% and 75% SOC. Lower and Higher to that boundary, there's an exponential behaviour. On the flat part, is important to notice how a difference of approximately 50 mV on the cell can be the difference between a SOC of 25% and a SOC of 90%.

The previous curve would be created in a Laboratory performing a charge cycle in a controlled environment. However, battery behaviour cannot be characterized with only one curve. The behaviour of the OCV-SOC curve of the cells varies due to current rate, temperature, aging (among other parameters). That's why cell manufacturers usually provide different curves for the same cell but with different ambient and operation conditions.

The voltage range of the cell varies due to manufacturer specifications and EOL required. Typically, the voltage range that a Lithium-Ion Cell has is from 2.8 V to 4.2 V.

2.1.2 Housing

In the battery market, cells can be assembled in three different housings:

- Pouch cells. These cells are the ones that are commonly used on mobile phones. Out of the three types, is the one that presents fewer mechanical properties as it does not have a metallic case. This makes them lightweight but, in an EV application it forces the creation of a well-engineered module. Also, the process of placing and joining the different cells together is challenging for the manufacturer. The module has to offer protection against mechanical agents as well as hold the different cells soldered.
- Cylindrical cells. These cells have a casing that has the same form as the one used on the AA batteries. There where the first cells used on EV's thanks to the fact that the production processes were easier with this topology of cells. They present a lower capacity (Ah) compared to the other two types. To achieve high current capability some cells need to be connected in parallel.

[3] Open Circuit Voltage (OCV): Voltage reading on battery cell terminal when there's no current flowing though the cell.

- Prismatic cells. Out of the three types are the ones that are bulkier as they present a rigid housing. On a vehicle application it would be the only type of cell that would be feasible to be replaced.

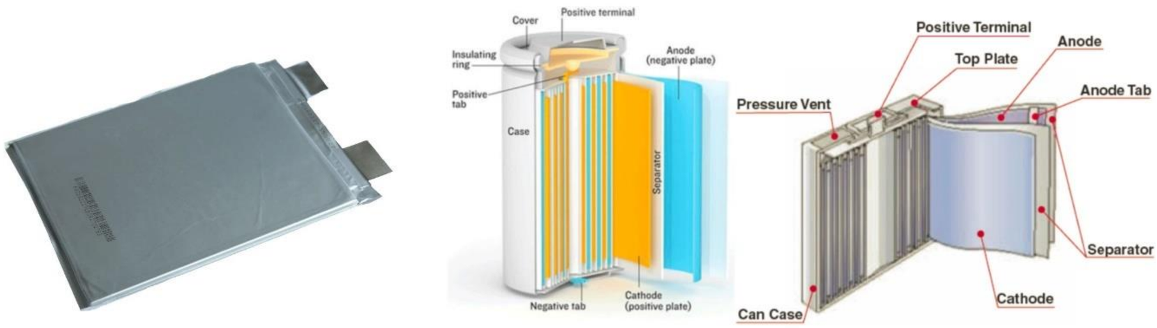


Figure 2. Different Lithium-Ion cell housings. Left: Pouch Cell. Centre: Cylindrical Cell. Right: Prismatic cell.

The housing could seem like a less relevant parameter; however, it affects different aspects on an EV application that are not performance-wise. The different types of housing impose different processes for building EV batteries. At the start, companies used cylindrical cells because the industrial process were easier on that type. Recently, other manufacturers have developed systems to mass produce with the other types of cells. Each of these housings imply different construction constraints and different construction processes that impact on the cost of the battery assembly.

Comparing the three housings in terms of energy density (kWh/kg), volumetric energy density (kWh/m3) or EOL is difficult. Cell manufacturers are constantly advancing and improving the cells and only provide the information to their clients. Making a comparison on those parameters today would be based on references that may be no longer at the forefront of technology.

Furthermore, talking of energy density or volume considering only the cell would not be a good comparison. Each of the housing types impose a different module and battery pack construction. For example, with a pouch cell you need a module that offers enough rigidity as the cells do not have mechanical properties. In a prismatic cell you would not need a module so rigid or with much material as the cell itself offers mechanical properties. Then, other parameters like cooling plates or battery pack chassis would need to be added to compute a real energy density and volumetric energy density values to make a true comparison.

Briefly, it could be said that out of the 3 cells, the ones that could be considered the "best" are prismatic and pouch. EV vehicles started mass production with voltage ranges up to 400 V with cylindrical cells. Manufacturers developed efficient industrial processes to build those batteries. Then, the wide gap semiconductors (SiC) appeared giving the chance to the manufacturers to raise the voltage of the battery pack, as the power electronics could handle it. With a higher Voltage, there are less losses in operation as the current is lower. Nowadays, the EV that have batteries with voltages of up to 800 V present Pouch cells, and EVs with 600 V have been seen using Prismatic cells. However, the manufacturers that keep using cylindrical cells still are on the 400 V range. This is no coincidence, and shows that cylindrical cells are behind (in a vehicle application).

2.1.3 Electrical Modelling and definitions

Battery cells present many no linearities and complex phenomena. Furthermore, battery behaviour varies due to electrical parameters (current drawn, actual voltage, SOC, aging), and is dependent on temperature. It must be added, that all these parameters are not independent to the others. In fact, they may be dependant in some factor with each other giving complexity to the modelling task.

Automotive environments are harsh, with vehicles having to be technically prepared and homologated to be functional on temperature ranges varying from -20 ºC to 65 ºC.

Furthermore, a cell embedded in a Battery pack must deliver a minimum range (defined by project requirements), certain charging and discharging current rates and a desired EOL range.

That is why battery testing is among other tests performed by the manufacturers to develop a new car platform. With a correct previous modelling of a battery pack, test costs can be drastically reduced, as the need for expensive Pre-Series batteries is reduced as technical issues can be found into the simulation phase.

To model a battery pack, with all its variables, first is needed to develop a battery cell model.

2.1.3.1 Circuit models

Among all the models, there is one that exceeds its acceptance by the engineers on the battery field. It is called the second order model.

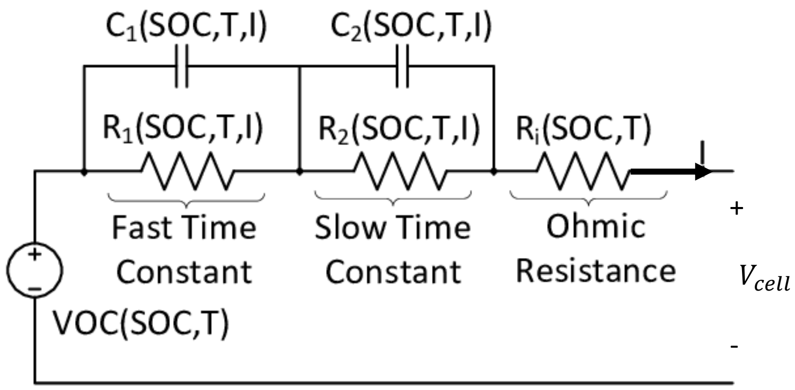


Figure 3. Second order model of a battery.

- $V_{oc}(SOC,T)$: Open circuit voltage. It is the voltage value that the cell offers with no current. This value is dependant to SOC and temperature.
- R_{int}: Internal resistance. It could be defined as the step on the voltage due to a step in current. It is directly linked to the connection elements of the cell and its internal behaviour.
- R_1 and C_1: These are describing the fast dynamics on the cell. When there's a step on current there's a voltage drop due to the R_{int}. Then this drop is increasing a little bit due to R_1 and C_1, as C_1 increases its value until steady state.
- R_2 and C_2: These are describing the slow dynamics on the cell. It's like R_1 and C_1 but with a slower time constant. This phenomenon can easily be confused with the raise of the V_{oc} due to charge increase.

The equations that are derived from the model are the following:

$$V_{cell} = V_{oc} - R_{int} * I_{cell} - V_{o1} - V_{o2} \quad (1)$$

$$V_{o1}(t) = (R_1 * I_{cell})\left(1 - e^{-\frac{t}{R_1 C_1}}\right) = (R_1 * I_{cell})\left(1 - e^{-\frac{t}{\tau_1}}\right) \quad (2)$$

$$V_{o2}(t) = (R_2 * I_{cell})\left(1 - e^{-\frac{t}{R_2 C_2}}\right) = (R_2 * I_{cell})\left(1 - e^{-\frac{t}{\tau_2}}\right) \quad (3)$$

As per standard on the field and literature, when a battery is being charged, the current is considered negative. The voltage seen on the terminals is higher than the VOC (1). When is being discharged the current is considered positive. The voltage then, is lower than VOC.

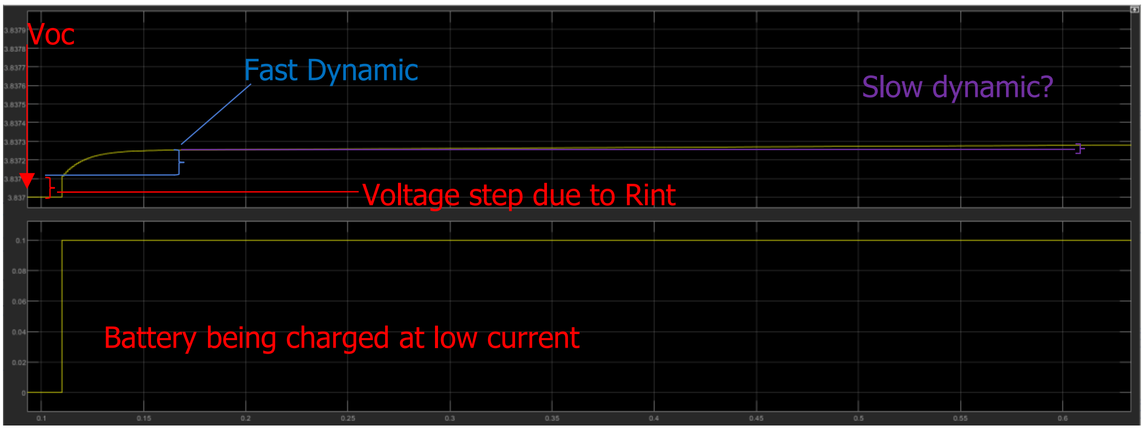


Figure 4. Top: Voltage of the Cell. Bottom: Current Curve.

Figure 4 can help to explain the different parts of the second order model. With no current, the voltage seen on the battery terminals is directly V_{oc}. Then there's a step on the current to 0.1 A. This causes a 0.1 mV increase on the voltage seen on terminals. This is directly caused by R_{int}. Then, there's the region characterised by the fast dynamic R_1 and C_1 and then there's the region characterized by slow dynamics. The slow dynamic section cannot always be determined as the increase of voltage could be associated with the increase of V_{oc} due to the charge increase.

The V_{oc} value depends on the SOC of the battery and the temperature of operation. On section 2.1.3.3 this topic is taken into further analysis.

The other parameters of the model ($R_{int}, R_1, R_2, C_1, C_2$) often aren't constant. Their value varies with the operation point, temperature, SOC, and aging of the cell. The change that may experience however, is expected to be minimum in comparison on its "original" or nominal value. Without real data, is difficult so stablish values or relationships to describe the change of these phenomena. For the modelling task and simulations $R_{int}, R_1, R_2, C_1, C_2$ are considered constant.

2.1.3.2 Capacity and C rate

The cell capacity is defined as the quantity of current that can be drawn constantly in an hour from a 100% charged cell that brings the cell to 0% charge. Is a parameter delivered by the cell manufacturer and it's a direct indicator of the cell's ability to store energy. It is defined in Ampers per hour (Ah).

If we consider two different cells, one with a capacity of 10 Ah and one with 1 Ah. With a nominal voltage of 4 V (parameter also provided by manufacturer) we can extract the following cell energy.

$$Cell\ Energy\ (Wh) = Nominal\ Voltage\ (V) * Cell\ Capacity\ (Ah) \quad (4)$$

$$E_{cell\ 1} = 4 * 10 = 40\ Wh \quad (5)$$

$$E_{cell\ 2} = 4 * 1 = 4\ Wh \quad (6)$$

It is observed how the first cell stores more energy than the second one.

The C rate is a parameter that indicates the maximum allowed current that can flow safely through a cell. It is a unitless value that is used together with the cell's capacity. For example, a 10 Ah cell that has a charge C rate of 2, and a discharge C rate of 3, would have a maximum charge current of 20 A and a maximum discharge current of 30 A.

These values are the maximum current in which the manufacturer ensures a well behaviour of the cell. However, cells could handle a current transient higher than that value as there's no limitation to the current that can deliver or absorb. If a short circuit is caused to a cell, the current that would flow through that cell would be high until the cell would start it's degradation by probably burning.

To obtain the real capacity of a cell, at a given time, the procedure described in the following points must be performed. This procedure is useful to analyse the capacity fade[4] of the cell. It must be done in a temperature-controlled environment as cells parameters are temperature sensitive. Usually, this procedure is performed at 25 ºC. To extract an accurate value, precise equipment must be used, especially on current measurement.

- Perform CC-CV[5] charge until maximum terminal voltage. The charge process performs a Constant Current (CC) charge until the maximum cell voltage is reached. At that point, the charger enters in Voltage Control (CV) Mode. The cell would absorb energy from the charger and dictates the drawn current. The current tends to reduce its value until it's infinitesimal. At that moment, the cell is in its maximum charge capacity, but as the cell initial state was unknown, the Cell capacity in Ah cannot be determined.

- Perform CC-CV discharge until minimum terminal voltage with a determined discharge rate. It is like the process seen before but with the aim to reach minimum charge. As we departed from a known point, with this operation the cell capacity can be extracted. This value is known as Q_{dcha}. This value defines the capacity that the cell has, to a given discharging rate, to provide energy.

- Perform again CC-CV charge until maximum terminal voltage with a determined charge rate. As we departed from a known point, with this operation, the Cell capacity can be extracted. In this case, the value is known as Q_{cha}. This value defines the capacity that the cell has, to a given charging rate, to store energy.

On the previous points, we could see two different nomenclatures that show two capacity values that can be slightly different from each other. If the current in which the CC process has been performed is low, $Q_{dcha} \cong Q_{cha}$. However, if the current rate is high, there can be differences. In case that the discharge rate value is high, the internal resistance makes Q_{dcha} lower. In charge, the effect of a high impedance would increase the Q_{cha} value, as more charge would be required to drive the cell to its maximum point.

With a cell that has some aging effect, both capacities would reduce overtime due to capacity fade and the CC-CV processes would last less time (the cell would charge and discharge faster). If there's a significant increase on R_{int} also, we would see that the times on these processes

[4] Capacity Fade: Known as the decrease of Capacity on a cell due to Aging effects.
[5] CC-CV: Is the way to charge the battery. Further implementation and behaviour described in 4.1.3.

would vary. The charge process would last longer, but thanks to the CC-CV process, the maximum capacity would be achieved. In the case of discharge, it would last less time.

To sum up when there would be aging effect Q_{dcha} and Q_{cha} would decrease. However, if the aging effect is high on the internal resistance of the cell, Q_{dcha} would decrease in a faster than Q_{cha}.

2.1.3.3 SOC

The state of charge (SOC) of a cell is the proportion of charge that is remaining in it in respect to its maximum. Is an indicator of the energy available to be drawn from the cell. SOC can be seen in a 0%-100% range, where 100% indicates that the cell is at maximum charge. Also, it can be presented in a 0-1 range where 1 indicates maximum charge.

In an ICE powered car, we can measure the remaining energy stored in the fuel tank (element that is equivalent to a battery) by a direct measurement via a fuel gauge. However, in a Battery there is no way to measure SOC directly so, it must be **estimated**.

This estimation must be performed with the variables that are accessible to the control electronics during operation. These are cell voltage at terminals, current and battery temperature.

On the following points, 3 estimation methods are presented. These are the ones more commonly deployed to battery management systems.

Lookup tables

A way of estimate the SOC of a cell is through a Look Up table. This table is created from the SOC-VOC curve (Figure 1). In this case however, the table is inverted so in the y axis there's the Open Circuit Voltage. This method is Voltage-Based and is dependent on the Voltage seen on the terminals of the cell.

$$V_{cell} = VOC(SOC) - \sum_i R_i i_{Ri} - R_{int} I_{cell} \tag{7}$$

From (7) the VOC can be extracted, even under operation. Then, with this voltage the SOC can be achieved using the table. This method relies on the current sense, voltage measurement and the accuracy of the Lookup table. An error on those parameters could led to a systematic error on the estimation.

This method of estimating SOC could be highly problematic. Lithium-Ion cells present a flat curve in voltage for a high SOC difference. If there's an error on the previous equation on the input parameters, it could lead to an error on the VOC voltage that would lead to a poor SOC estimation. In case the cell chemistry is more linear, this effect would not be so critical.

This method needs a set of different Lookup tables and must integrate a separate method to consider the cells aging behaviour. The different lookup tables would be necessary to estimate the SOC in different temperatures. Also, different tables could be used to characterize the aging behaviour. This would force to introduce to the electronics of the battery different tables as well as incorporating a series of conditions to switch between them.

Kalman filters

Kalman filter is one of the best ways to estimate SOC and it's largely adopted by manufacturers. It is a model-based estimation method that, with different algorithms, estimates the state of charge of the cell.

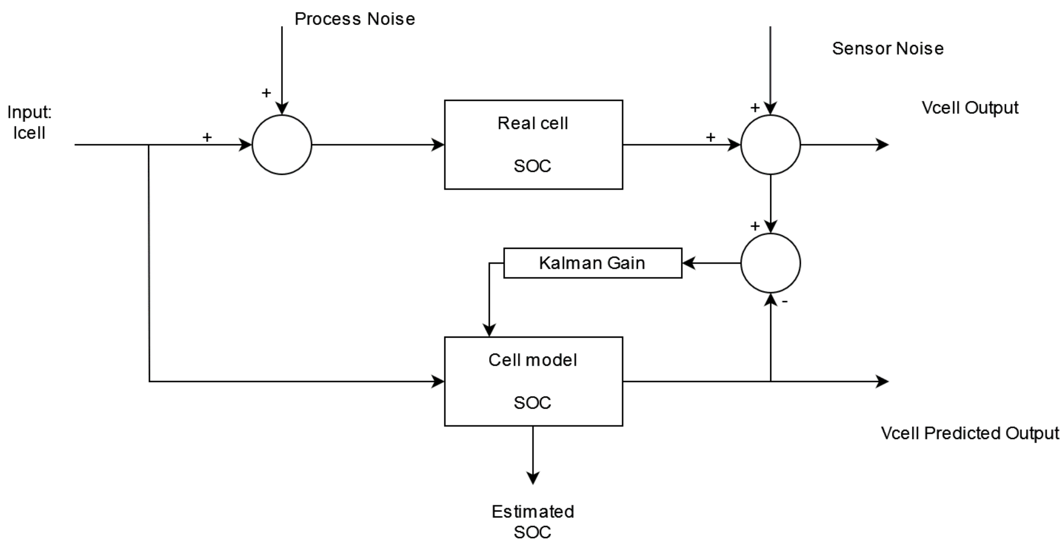


Figure 5. Kalman Filter Algorithm in block diagram.

The code of the Kalman filter is applied on the cell model part. The Kalman filter is based in a Generic Gaussian sequential probabilistic recursion. Its precision is gained through multiple iterations of the process.

The filter, from the current input and with its initial condition, performs an estimation of the current SOC. This state of charge is translated to an estimated cell voltage. Then this voltage is compared with the measured one. This gives an error value that after it passes through a gain stage, is feedbacked to the Kalman filter. The algorithm needs some time to achieve a good initial SOC estimation. This time can be reduced with an accurate initial condition given to the Kalman filter.

The Kalman filter algorithm has two stages that are explained on the following points. On each of the points, the equivalent equation using a second order battery model is written.

A simple filter has a state variable which is SOC, an input which is I_{bat} and V_c $(cell\ voltage)$ as an output.

- Prediction. In the prediction stage, the model generates an estimated V_c value. In this stage, the SOC estimation is obtained. The prediction section has 3 parts:

    1. P1: Prediction of the SOC with prior information.

$$SOC(k) = A_{k-1} * \widehat{SOC}(k-1) - \frac{1}{C_{cell}} i_{bat}(k-1) \quad (8)$$

$$SOC(k) = A_{k-1} * \widehat{SOC}(k-1) - B_{k-1} * \ i_{bat}(k-1) \quad (9)$$

    2. P2: Updating the error covariance matrix. Proprietary and necessary for the algorithm as it is a probabilistic recursion.

$$\sum\nolimits_{\widetilde{SOC},k} = A_{k-1} * \sum\nolimits_{\widetilde{SOC},k-1} * A_{k-1}^{T} + E_w \quad (10)$$

Written with a more understandable nomenclature:

$$Cov(k) = A_{k-1} * Cov(k-1) * A_{k-1}^T + Cov_{process} \quad (11)$$

Process covariance is a constant value.

3. P3: Prediction of the system output. Here is where with the SOC estimation, the battery model, and the VOC curve, a predicted V_c is obtained. The following equation is based on the second order model of a cell.

$$V_{cell}(k) = VOC\left(\widehat{SOC}(k)\right) - \sum_i R_i i_{Ri} - R_{int} i_{bat}(k) \quad (12)$$

In this case we are trying to explain the equations of a **linear Kalman filter**. The VOC (Soc(k)) parameter is non-linear. A fast linearization would be the following.

$$VOC(SOC(k)) = V_{cmin} + (V_{cmax} - V_{cmin})SOC(k) \quad (13)$$

Where V_{cmin} is the minimum cell voltage that the manufacturer indicates and V_{cmax} is the maximum one.

With (13) and simplifying the second order to a model with only an internal resistance:

$$V_c(k) = V_c + (V_{cmax} - V_{cmin})SOC(k) - R_{int} i_{bat}(k) \quad (14)$$

$$V_c(k) = V_{cmin} + C_k \widehat{SOC}(k) - D_k i_{bat}(k) \quad (15)$$

- Correction:

1. C1: Update the feedback gain from the error value.

$$L_k = \sum\nolimits_{\widetilde{Soc},k} C_k^T \left[C_k \sum\nolimits_{\widetilde{SOC},k} C_k^T + \sum \tilde{v} \right]^{-1} \quad (16)$$

Written with understandable nomenclature:

$$L_k = Cov(k) * C_k^T \left[C_k Cov(k) C_k^T + Cov_{sens} \right]^{-1} \quad (17)$$

The covariance of the sensor is a constant given to the algorithm.

2. C2: Get the error and update $\widehat{SOC}\,(k)$. In the next iteration this value will be $\widehat{SOC}\,(k-1)$.

$$\widehat{SOC}(k) = \widehat{SOC}(k) + L_k\left(V_{cellmeas} - V_{cellpred}\right) \quad (18)$$

3. C3: Update the state error measurement for next iteration. This value will be $Cov(k-1)$ in the next iteration.

$$\sum\nolimits_{\widetilde{SOC},k} = \sum\nolimits_{\widetilde{SOC},k} - L_k \sum\nolimits_{\widetilde{SOC},k} L_k^T \quad (19)$$

Written with more understandable nomenclature:

$$Cov(k) = Cov(k) - L_k * Cov(Estimated\ V_c) * L_k^T \quad (20)$$

$Cov(Estimated\ Vbatt)$ is defined as the covariance calculation among all the different estimated Vc.

The topic about Kalman filters is extensive and cannot be covered under this project as it is out of the Scope. Reference [10] on section 6.2 drives deep into this topic and provides the basis of Kalman Filters.

Coulomb counting

Coulomb counting is one of the simplest methods to estimate the state of charge. It departs from a known state, with the cells fully charged or discharged. This method constantly measures the current of the battery and then, with the sample time of the measurement, the net capacity extracted or given to the battery is calculated.

This net capacity (Ah) is added/removed to the initial state capacity. If the battery was fully charged, the initial state capacity $C(0) = C_{cell}$. If it was fully discharged, $C(0) = 0$.

Then, by comparing the result of the adding operation with the capacity of the cell, the state of charge can be obtained.

$$SOC(t) = \frac{1}{C_{cell}} * \left[C(0) + \int_0^t I_{cell} * dt\right] \quad (21)$$

$$SOC(t) = SOC(0) + \frac{1}{C_{cell}} * \int_0^t I_{cell} * dt \quad (22)$$

If we extrapolate this method to a hydraulic system, a cell would be a water tank. If it departs form the maximum level of 10 L (100%), and we start extracting water at a 0.1 L/s ratio, after 10 seconds, 1 L would have been removed. So, the tank would be at 90% capacity. This calculation is the same that is being done in *Coloumb Counting*.

The current sense is crucial to the well behaviour of *Coloumb counting.* A systematic error or hysteresis introduced on the current sense measurement would affect the SOC(t) calculation. This, added to the fact that it needs to depart from a situation of a battery cell being in full charge or discharge, could led to a significative error on the SOC estimation. This error increases until the battery reaches full charge or discharge. Then the initial state of SOC (SOC (0)) can be re-calculated.

These reasons make this way of estimating SOC not usable into an HEV non-plug-in vehicle. In a HEV non-plug-in vehicle, the battery usually is between the 30% and 70% SOC. The only way to get to a 0% or 100% is with the vehicle being driven for a long and with the ICE providing energy to the battery. This would involve different electro-mechanical devices on the vehicle.

Furthermore, *Coulomb Counting* does not consider the effect of the temperature on its computing.

2.2 Battery Pack

For EV applications, multiple cells are required to give the vehicle the autonomy required by the users. At large, it could be said that a Battery Pack is only a grouping of cells. However, there are more elements needed that will be presented in the following sections.

2.2.1 Structure and elements

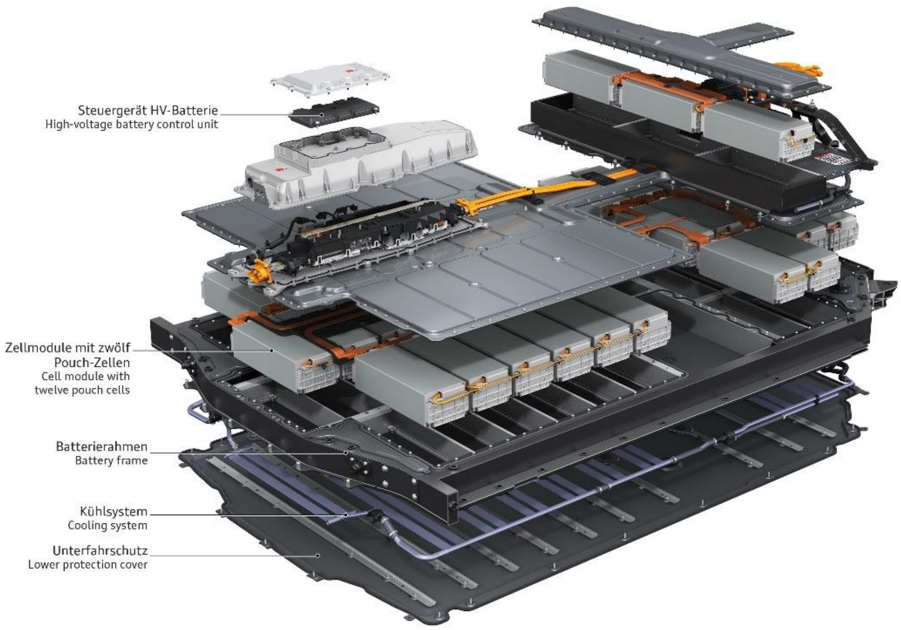


Figure 6. 3D cut of a battery pack.

The first that can be observed in Figure 6 is how the cells are grouped and connected. In this battery pack cells are contained on a larger unit called module. By putting different modules in series, the desired battery capacity and voltage is obtained. The elements that connect the different modules together are called busbars. These are the large orange connectors on the figure.

After connecting the different modules in series, two terminals with the battery voltage are obtained. These terminals, for safety reason, cannot go directly to the vehicle's DC bus. In this bus other elements such as the inverter or the On-Board Charger (OBC) are connected. The element that stands in between battery terminals and the "outside world" of the battery is what in this battery pack is called High-Voltage Battery control unit. In other battery packs this element can be named as Battery Junction Box or Battery Distribution Unit (BDU). The difference is only on the name.

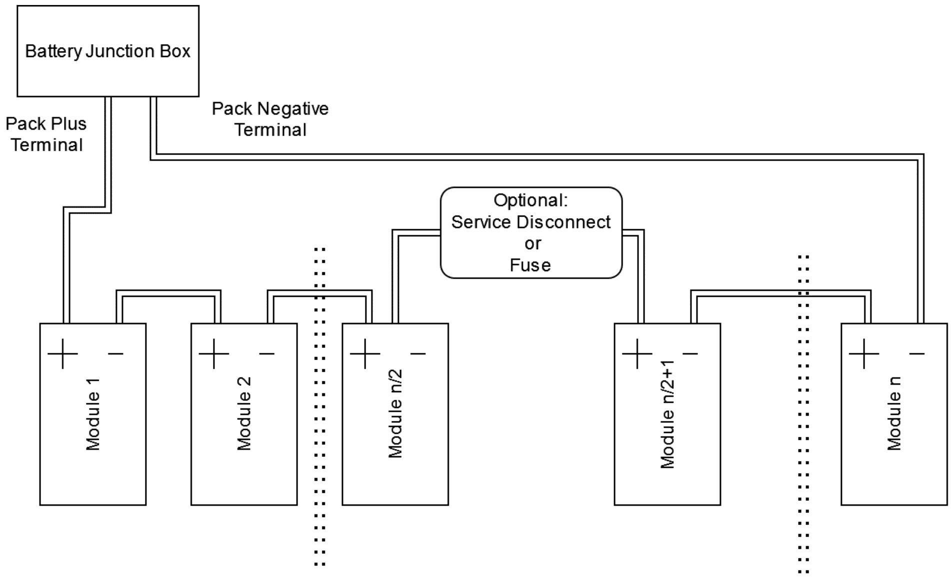


Figure 7. Electrical structure of a battery pack.

A Battery Junction Box contains several protections and part of the control of the battery pack. Regarding protections, minimum, it contains two relays that permit the connection of the battery pack to the other car elements. Furthermore, there can be fuses that in case of a short circuit would protect the battery pack from overcurrent. This element also has some electronics that have the control of these relays and fuses. Those electronics decide whether the relays can stay closed or shall open. Furthermore, there are certain measurements that are made in the Junction box. Mainly battery pack voltage, battery current and temperature.

In certain battery packs, normally at the middle (electrical wise speaking), a fuse or a service disconnect (or both) can be found. A service disconnect is a mechanical element that permits to cut in half electrically the battery pack. As its name states, it is intended for the service of the battery.

In the following sections the elements commented before are introduced and explained in further detail.

2.2.1.1 Modules

A Battery module is a grouping of cells connected in a certain way. Cell topology and number of cells needed dictate its positioning and morphology.

By creating a battery by grouping modules, manufacturers have found a way of optimizing the manufacturing process, increasing safety and recyclability.

Regarding safety, module construction permits to isolate mechanically the modules. In some cases, a metallic wall being part of the battery frame is placed between modules. In case that there's a mechanical impact, these walls can protect modules from impacting into others.

Also, a battery pack made in modules is easier to service. If a problem is found on a cell, the module containing that cell can be extracted and replaced, enlarging the life of the rest of the battery pack.

Finally, having a module construction permits to have a better recyclability. By removing the modules, all the cells can be removed from the pack. The frame then can be recycled as regular metal scrape. Then, modules and cells can be recycled using the most convenient way that the manufacturer consider.

Figure 8. BMW I3 module, top plastic removed.

Figure 8 shows a module with prismatic cells. In this case it is a module extracted form a BMW I3 (one of the first mass produced EV's). There are a total number of 12 cells being connected in series using plates that are welded in the manufacturing process. The blue arrows indicate the connection of the different cells in series.

Small cables can be observed emerging from the plates of the connections. These cables contain the voltage measurement of cells. These connections can be used to perform balancing on the cells. The section of these cables is generally small and the maximum current that can pass through them is limited in concordance.

In some cases, the small cables are replaced by a flexible PCB. This PCB may be cheaper than the small cables but presents a capacity to drawn current from the cell lower. This PCB permits to reduce the volume of the module.

NTC sensors measuring cell's temperature may be present on the module. However, are not visible on the figure.

The orange cable and the aluminium plate are the terminals of the module. To these terminals other modules connect to create a battery pack.

2.2.1.2 Protection elements

To ensure safety for the vehicle and to the battery itself, battery packs have different elements to provide safety. Usually, these elements are contained in the Battery Junction box.

Relays

Figure 9. Hv relay example.

HV relays are placed between the battery pack terminal and the different elements that are connected to the DC bus of the vehicle. On the battery pack terminal there's always voltage as batteries are an element that cannot be switched off. The HV relays are the ones that enable that voltage to reach the different elements on the HV bus.

When the vehicle needs traction the HV relays close. Before that, and to avoid overcurrent on the battery, a precharge is made on the HV bus. This precacharge is necessary to slowly charge all the capacitances present on the DC bus. This precharge circuit would need an additional switch between the battery terminals and the HV bus.

When the precharge is finished, the relays close. The different elements on the DC bus are then allowed to give or consume energy from the battery.

In case that the electronics of the vehicle detect an anomaly on the system, the relays are opened. With this the HV bus is isolated from the battery. The anomaly can be an overcurrent, a crash of the vehicle or a malfunction. These relays must have the cutting power to open under operation (current flowing).

Fuse

Figure 10. Fuse Example.

On the battery junction box, a set of fuses are placed. Those fuses have a cut curve in which if there's an overcurrent for a determined time, they cut the connection permanently. So, the battery is no longer usable. Its main intention is to protect the battery pack against overcurrent with no software intervention.

Let's imagine a short circuit, on the HV bus. The current then can be up to 1000 or 1500 Amperes. In that case, the protection that should be triggered is the fuse. The reason behind using the fuse and not the relays is that relays depend on SW to open. That SW would take time to analyse the data from the current sensor, verify that the input is correct, and then act to the HV relays. With the fuses however, if the current is above their triggering curve, they cut the current flow permanently. As it is a fuse, if it has acted then it must be replaced.

Service disconnect

Figure 11. Service disconnect Renhotec EV.

Figure 11 shows a service disconnect. Is an element that is placed on the battery pack housing and must be accessible for the end user and service personnel. As all the connectors that are related with the HV bus of the battery, it has an orange colour. This connector has a Poka-Yoke construction, in which is not possible to do a misconnection. This element also has an interlock. The electronics of the battery, with that interlock, detect if this element is plugged or unplugged.

When this device is removed it cuts in half the battery. As it is cut in half, considering that there's no loss of isolation on the battery, it makes the works on the Battery safer. For example, if the service disconnect is unplugged, and a metallic tool that the technician drops touches both plus and negative terminals of the battery on the Junction Box, no short circuit would happen.

2.2.1.3 Current Sense

Current measurement is one of the important variables at a battery pack level. Measuring current not only can serve as a method to protect the battery pack but it also provides a way to estimate certain parameters of it.

A battery pack has a maximum current limit in which can operate. The BMS monitors the current drawn or given to the battery and if the current is overpassed it may open the relays and isolate the battery. It is considered that if there's an overcurrent it could be because there's some element in the circuit malfunctioning.

Furthermore, the current measurement provides a way of on-line estimating cells internal resistance and SOC via coulomb counting. These two characteristics require the current measurement to be precise, especially in the case of coulomb counting.

There are two main sensor types that manufacturers use for sensing Battery pack current.

Shunt Sensor

Shunt sensors are a known resistance in which the voltage drop that happens across it is monitored. In a battery, are placed in a way that all the current that flows through the pack passes through this sensor. By using ohm's law, the current that passes through it is extracted, so the Battery Pack current value is achieved.

Benefits:

- There's no Voltage offset error. If no current is passing through the shunt resistance the voltage drop is 0 V.

Drawbacks:

- Shunt resistance dissipates energy. This decreases the overall efficiency of the pack and creates a point that could overheated.
- Resistance precision. If the resistance has a different ohmic value than the one used on the ohm's law calculation, there will be an offset difference on all measurements.

  Resistances are a passive element that are temperature dependant. Shunt resistances can be affected by the change of the ambient temperature and by the temperature of the shunt itself. To mitigate this drawback, BMS manufacturers establish a series of tables depending on the operating temperature of the shunt resistance. BMS software choses the correct ohmic value for each of the current calculations.

Hall Sensor

A hall effect sensor uses the magnetic field produced by a conductor that carries current. The current flowing on that conductor creates a voltage on the sensor proportional to the current value. These sensors are placed like a ring over the conductor that it needs to be measured.

Benefits:

- The accuracy of the measurement does not depend on the temperature.
- The sensor is isolated from live parts.

Drawbacks:

- Hall sensors suffer from offset. It can be corrected with calibration in zero current scenarios (like in case of having the HV relays opened).

2.2.2 BMS and topologies

The Battery management system or BMS consists of electronics that monitor, protect, and control a battery pack. The following table shows the principal variables that a BMS monitors.

Table 1. BMS principal monitored variables.

Battery Variable	*Use*
Cell Voltage	SOC, Protection against Over Voltage or Under Voltage on the cell
Cell Temperature	Protection against Over Temperature or Under temperature
Module Voltage	Redundancy
Pack Voltage	Redundancy
Pack current	Operational state of the battery and SOC

Voltage measurement is performed on each one of the cells of the battery pack. This measurement is essential to ensure that the battery operates in a safe state and protects the cell grouping against Under-Voltage or Over-Voltage.

The overall module or pack voltage could be determined with the sum of each one of the cells. However, having independent measurements of the module or the pack provides robustness to the system. For example: without a global measurement of the voltage of the battery pack, the BMS would not be able to determine if there has been a disconnection of a busbar between modules.

Temperature measurement is performed usually by NTC sensors placed near the cells inside the module casing. Temperature is a variable that has a slow dynamic. That's why NTC sensors are placed. The sampling frequency of the temperature is slightly lower than the one used for voltage measurements. The need for a temperature measurement is due to battery operational temperature range. On lower temperatures, the capacity is significantly lower than the nominal one. Furthermore, if the battery is operated in such low temperature the aging of the cells can be accelerated. On higher temperatures, battery cells can offer more current and have a higher electrical performance. However, that statement comes to a cost as the aging is also accelerated.

To protect the battery, the BMS can reduce the maximum current allowed (on charge and discharge) to ensure that the battery does not operate outside temperature limits.

BMS can be deployed in a series of different structures and configurations, on the following points the most common structures are presented.

Figure 12 show how the different topologies would be deployed on a Battery Pack application.

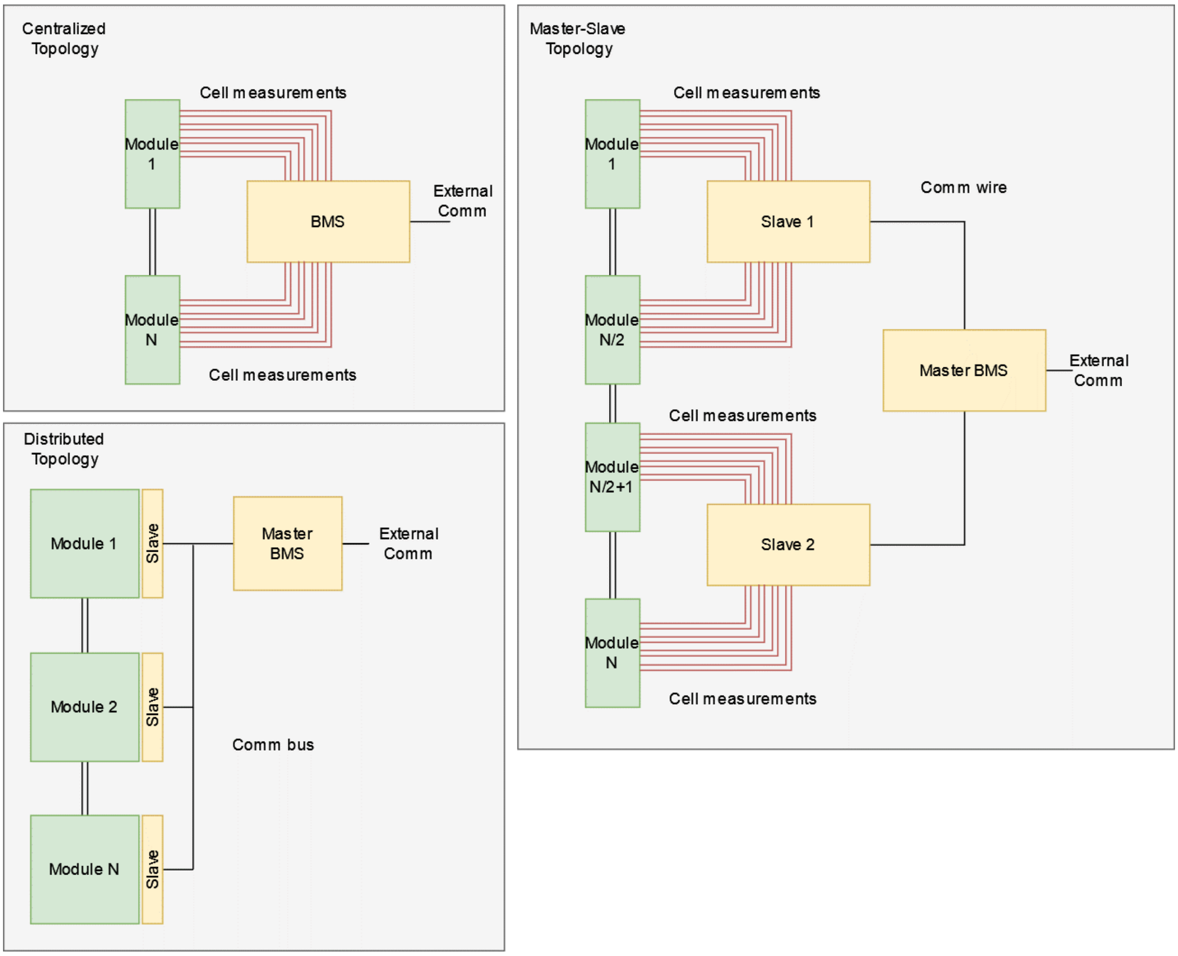


Figure 12. Deployment of different BMS topologies.

Centralized Topology

In a centralized BMS there's only one control board present on the battery pack. All the cell's measurements go to the same point. It does not matter how the cells are organized mechanically; all the measurements come together at the same point. The balancing is performed on that board.

The board controls the BDU and provides communication and diagnostic services with the vehicle.

Despite it might seem that this is a simple way to structure a BMS, it presents a series of drawbacks:

- All the balancing action is performed at one spot. The thermal management of the energy flowing though the dissipative resistors must be considered to prevent damage to the BMS board. The balancing resistor value chosen has to be taken into account together with the wire resistance from the cell to the BMS.
- Several wires with measurements must be routed between the BMS and the measurement points. As many wires are required, the cost and the weight of the battery pack is increased.
- Having so many wires through the battery pack has an impact on safety. In an event of crash or service, the isolation of the conductors can be compromised. If two live wires contact each other, they can create a path in which current can flow causing a short circuit.
- The long distance between the cells and the BMS can introduce electrical noise into the measurement. However, the BMS board could be placed away from the electrical noise caused by battery operation.
- The mounting of the wiring harness is a process that may take time and may increase the time needed to produce the complete battery pack.

Master Slave(s) Topology

In this configuration there's a master BMS that carries all the part of computing, communications and diagnostics. With this master there are several slaves. The slave boards receive the wiring connections of the cells and perform the balancing action. The slave can have some logic implemented like SOC estimation, internal resistance or can have the ability to trigger the balancing action. They also perform the temperature calculation with the voltage readout from the NTC sensor.

The slave reports periodically to the master all the information in regard of the status of the cells. For each slave there would be a direct cable to the master. With the configuration of master-slave, the slave could be placed close to the cells to reduce wiring. This would increase the number of housings needed but would reduce the wiring though the battery assembly (only the communication wiring would be necessary).

Distributed Topology

This configuration is similar to a master slave configuration but presents a difference on the way that the master is connected to the slaves. In this case each one of the slave boards are interconnected between them. The different boards are connected in a way that is known as daisy chain. In the end there's a bus creation. In this bus, each of the slaves share its state. The master BMS then gives commands to each one of the slaves.

With this configuration the wiring is reduced, as only a communication wire must be connected to each of the elements. Furthermore, the slaves are placed close to the module eliminating the number of wires carrying voltage measurements on the battery. In each of the slaves there's an isolation barrier between the measurement part and the communications layer, increasing even more the electrical safety. If the communication cable suffers a fault, it would not cause any major electrical risk as is isolated from the cell's measurement's part.

This configuration also makes the service of the battery pack easier. If there's a fault on the communication harness between the slaves, it could be replaced with apparently no risk as the pack would use isolated connectors. If there's a fault on the slave, it could also be replaced with no risk as the connectors from the modules are also isolated.

The configuration has the property that is fully modular, and it can be applied to different battery pack sizes. For example, if a manufacturer sells a same vehicle with two battery types, the difference on the BMS side would be to add more slaves to the communication bus.

This topology however, presents a couple of drawbacks:

- Cost increases due to the increase of PCB assemblies required. This cost increase has to be compared with the decrease of cost due to the reduction of cabling, as well as with the time reduction in production. What could seem an increase due to the cost of the PCBs, it couldn't be causing an over-cost in a global view.
- Electrical noise immunity. The PCBs are close to the cells, its components would be subject to noise introduction due to the cell's operation. The communication cables would also be subject to this electrical noise. To do not suffer loss of performance, extra measures would be required.

After reviewing the 3 topologies, the modelling part of the project will be made thinking of a **distributed BMS** application. I consider it to be the best out of the three thanks to its characteristics (mainly due to modularity).

On the next points of the project, especially in section 4, some parameters will be chosen according to this decision.

3 Balancing

3.1 Why is needed

Under operation, the cells have a limit voltage range. In an EV application there are multiple cells in series. In case only one of those cells reaches its operational limit the performance of the battery pack is limited.

To avoid this restriction caused by only one of the cell's, in EV applications there is always a balancing system.

3.1.1 Safety

Recently it has been seen how different electric scooters get ignited during charge processes. This is probably due to having the cells unbalanced together with a poor charger.

That charger may deliver a CC-CV charge profile in which its target voltage is the maximum voltage per cell multiplied by the number of cells in series. In an ideal scenario this would not cause any issue. All cells would evolve and charge at the same ratio and would depart from the same state (are in balance with each other).

Figure 13 shows how it would develop the charge process of a balanced and unbalanced battery pack. Both packs start having the cells at a 3.6 V, except one cell that has 3.75 V on the right one. The charge process consists of a constant current stage until the final voltage is reached. For this example, the final voltage is 12.9 V (maximum voltage of the cells is 4.3 V). The battery charger starts its CV part when the voltage seen on the terminals of the battery assembly is 12.9 V. In the left case is not a problem as all the cells have raised their voltage in a balanced manner until they reach the final voltage. However, in the second battery assembly when there are the 12.9 V on the terminals of the battery, the cell that started on 3.75 V may be on 4.4 V. This value is 0.2 V higher than the maximum value. In this situation we've considered that the charge would be linear, and that the same voltage difference would be kept after the charging process. However, it shows that if there's no BMS and the charger is not smart enough a situation with an overcharged cell can occur. In this scenario the cell could ignite by itself.

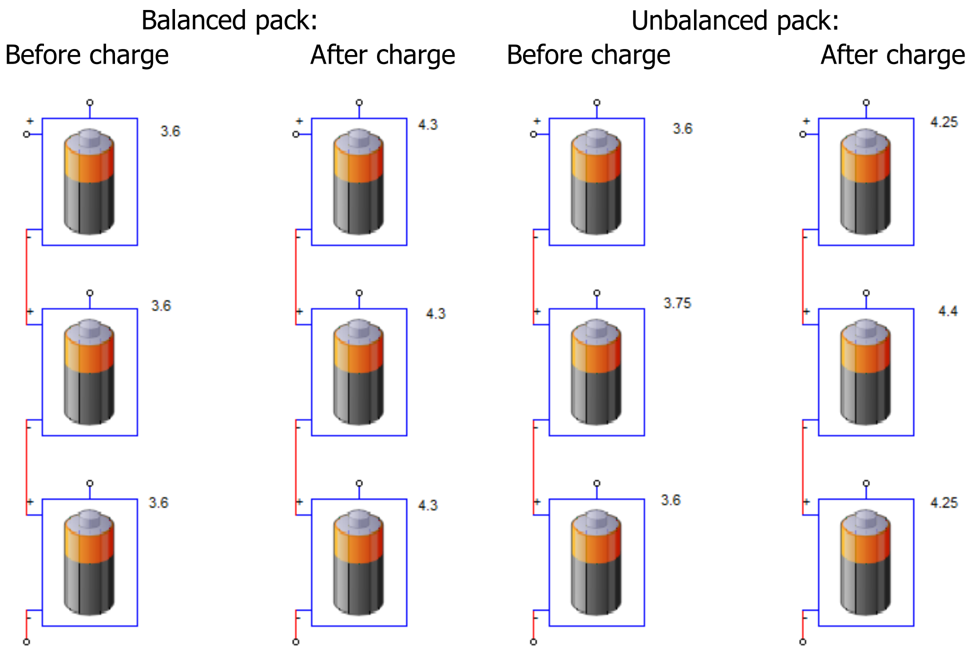


Figure 13. Left: Balanced pack behaviour. Right: Unbalanced pack behaviour.

That same example can be applied to the discharge process. It could lead to a scenario in which the voltage on terminal is the lower boundary, but one of the cells has been over discharged.

This example is simple, with 3 cells in series. Yet it shows how a minor difference on one single cell could cause a failure of the battery. In an electric vehicle that has an architecture of 800 V with up to 187 cells, the situation of having an over charged or over discharged cell cannot happen. In case that there's a cell that is above the limits, the performance of the car will be limited to avoid major issues.

3.1.2 Energy considerations

In the previous point it has been seen that an electric vehicle can limit its performance based on the maximum and minimum cell voltage values. If only a cell is outside this limit, all the battery pack will be limited by that single cell.

For making a comparison between the different cells being at different voltages, first the voltage-capacity approach is made. This approach assumes that a cell with ΔCapacity in respect of other cells has a ΔV.

On Figure 14 there's a graphic that serve as example of two scenarios. As per initial conditions the cells on the left case are perfectly balanced, they have the same SOC. On the right side however, there's a difference of 33% between the most charged one and the less charged one.

When there's a charge process, the cells on the left reach its maximum SOC (so maximum cell voltage). On the right side, the charge process stops when the two cells that were in a higher state, reach a 100% SOC. Leaving some cells with less energy stored into them. The energy gain per cell assembly on both cases is in green outside the graphic, and the "lost" energy is on red. If the charge process has kept the difference between the SOC of the cells, there's still a 33% difference.

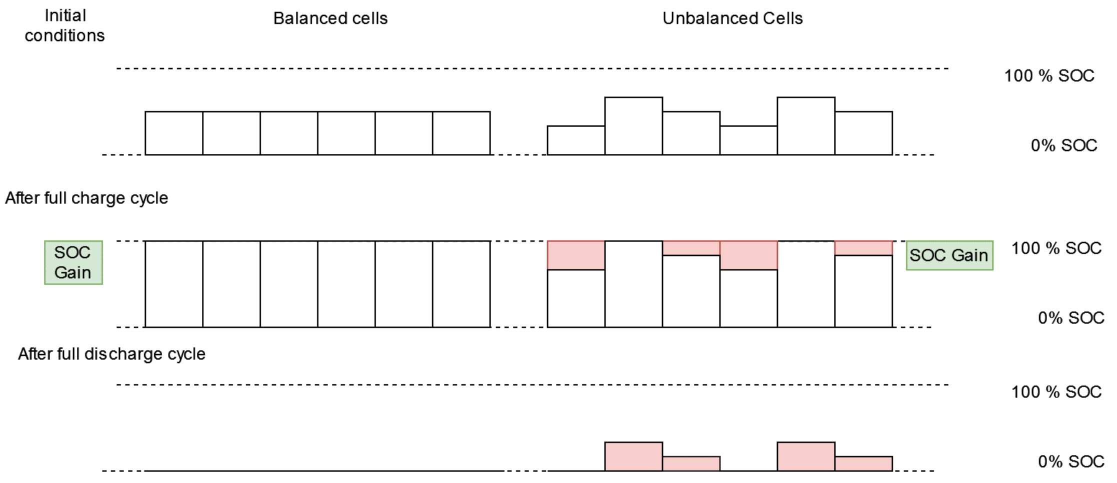


Figure 14. 6 Cell in series assembly scenario. Left: Balanced Cells Assembly. Right: Unbalanced Cells Assembly.

Right after the charge process, a discharge process starts. The less charged cells now mark the limit of discharge. When those cells reach the 0% SOC, there are cells that still have some capacity to offer.

In this example the operating range of the battery assembly has been reduced. On the left one the 100% of the SOC has been usable. In the right battery assembly, there's been a reduction of 33 % of the SOC available due to cell unbalance. So, there has been a 33 % of capacity that the battery pack could offer that has been lost.

$$SOC_{av} = 100\% - \Delta SOC_{c\,max} \tag{23}$$

$$C_{av} = C_{nom} * SOC_{av} \tag{24}$$

We consider a 60 kWh EV battery that after full charge has a 5% difference in SOC in their cells. An energy consumption of 20 kWh/100 km of the vehicle is considered.

Battery in perfect condition:

$$SOC_{av} = 100\% - 0\% = 100\% \tag{25}$$

$$C_{av} = C_{nom} = 60\ kWh \tag{26}$$

$$Range = \frac{C_{av}}{E_{cons}} = \frac{60\ kWh}{20\ kWh} * 100\ km = 300\ km \tag{27}$$

Battery with 5% ΔSOC:

$$SOC_{av} = 100\% - 5\% = 95\% \tag{28}$$

$$C_{av} = C_{nom} * 0.95 = 57\ kWh \tag{29}$$

$$Range = \frac{C_{av}}{E_{cons}} = \frac{57\ kWh}{20\ kWh} * 100\ km = 285\ km \tag{30}$$

There's been a reduction of vehicle's range only due to cell unbalance.

It could be argued that the over discharged or over charged cell(s) could be bypassed to keep the battery pack operating in the top or bottom point. That would require different switching elements in the interconnection of battery cells and modules. In a simulation environment or even on some applications this system could work. However, in an EV application is not feasible due to the over cost of the switching elements and the weight and volume increase that would imply.

3.2 Factors that led to cell unbalance

The factors that led to a cell unbalance are numerous. These factors can be attributed to electrical or mechanical phenomena. By sheer definition, battery cells have a series of phenomena that are not easy to understand. Furthermore, it's application on a harsh environment like the automotive one, increases the number of factors. The reason of that statement is that automobiles are subject to high temperatures, low temperatures, vibration and in some cases, extreme G-forces.

Battery packs, after they pass the end of line test on production, are checked to ensure that they leave the factory balanced. The following factors described may create unbalance after the battery leaves the production facility.

Differences in manufacturing

The processes on battery cell manufacturing are complex. They require technology and a lot of precision in the materials used in order to create them. Also, the fabrication processes require clean spaces to avoid undesired particles enter into the cells. Parameters like internal resistance, self-discharge (or leak) resistance, SOC-VOC curve and Capacity can change from one cell to another due to differences on those processes or due to imperfections on the materials.

Then, there's a phase called formation. This phase creates an initial voltage on the cells and performs a series of charging and discharging cycles. Compared to an ICE, it would be the equivalent to *running* the engine. In this phase as, it is close to the end of line, the desired specs of the pack can be checked.

If in the construction phase or in the formation phase there is any kind of major difference between cells, it could lead to unbalance after the battery pack leaves the factory.

Temperature

Automotive components operate in high temperatures and in freezing ones. Cells that are subject to temperature changes may suffer changes in its properties. It's SOC-VOC curve can change as well as its capacity. Also, in a vehicle battery pack the temperature change may be uneven so some cells could be affected more than others.

Operating in such conditions, in which the cell properties may change from one cell to another and keeping the same load for each one of them, could lead to a scenario in which after operation the cells are unbalanced.

Aging

Aging is a process that cells suffer to its performance due to operation. This process affects the cells capacity as well as other parameters (R_{int}, Self-Discharge). It can be caused by the number of cycles that that cell has suffered, and it's also function of the Deep of Discharge (DOD) of those cycles. Performing cycles outside the temperature range recommended by manufacturer accelerate the aging phenomena.

Aging is caused by the accumulation of impurities on the anode and cathodes due to the reversible reactions that the cell experiment during operation.

In automotive, is said that a cell that has reached the 80% of its original capacity has reached end of life. In that moment the cell is no longer usable for automotive applications.

3.3 Techniques for balancing a battery pack

In the following sections different balancing techniques are presented. These ones are simulated in the section four of the Thesis.

3.3.1 Dissipative Techniques

Dissipative Techniques are well known by the battery industry and are the ones that are currently used in EV.

The dissipative techniques are based on a HW that permits to put in parallel a cell with a resistor. Through the resistor, the excess energy that a cell may contain is dissipated achieving the desired balance between the different cells. Figure 15 shows the HW necessary.

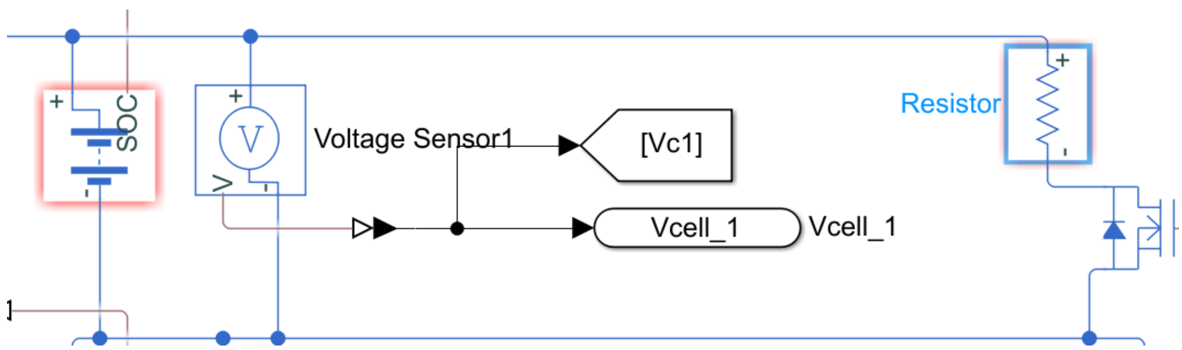


Figure 15. Battery Cell at the left, voltage measurement and balancing resistor with switch at the right.

The resistance value of the balancing resistor is a factor that, through Ohm's law, is directly linked with the balancing current.

$$I_{bal} = \frac{V_{cell}}{R_{br}} \rightarrow I_{bal_{max}} = \frac{V_{cell_{MAX}}}{R_{br}} \quad (31)$$

As the voltage of the cell decreases as the balancing action is applied, the current is also reduced. It is important to remark that, from a design perspective, the maximum balancing current is going to happen when the cell is in its maximum voltage.

In the HW presented in Figure 15, there will be switches and resistances as many cells are.

$$N_{switch} = N_{Cell} \quad (32)$$

Switched resistor

Some literature in the field presents the possibility of having an only balancing resistor for a grouping of cells. This resistor would be connected with the cell to be balanced though a switching network that would act like a multiplexer.

This HW proposal despite it permits to lower the number of resistors needed, it adds the need for a multiplexor. Furthermore, for a given fixed balancing current this method would be slower than the one presented in the Figure 15. In cases that there would be the need to balance multiple cells, the switched resistor would need to be shared with the different cells that are over charged. However, in the circuit in Figure 15 all cell's balancing action could operate at the same time synchronously.

The following techniques that are presented in this section present different SW and control strategies that can be applied to dissipative balancing circuits.

3.3.1.1 ΔVOC

ΔVOC approach is based on the consideration that the cells with the same open circuit voltage (VOC) have the same SOC.

$$V_{c1} = 3.7\,V;\ V_{c2} = 3.8\,V \rightarrow V_{c2}\ has\ a\ greater\ SOC \rightarrow It\ needs\ to\ balanced \quad (33)$$

Following are the input parameters of the algorithm:

- Cell voltages
- Cell current (ΔVOC – R drop)

This algorithm despite is one of the simplest ones present a series of challenges:

- Li-Ion flat curve: As seen in Figure 1, there's a region in which the difference in voltage is minimum compared with the difference on SOC. If the precision of the voltage measurement presents an error or offset, or the curve of a given cell changes from the other cells in the same pack, the algorithm could not detect a SOC difference on the cells.
- Resistance drop: The algorithm relies on the voltage measurement on the cell. If the battery is under operation (charging or discharging) or by the sheer balancing action, the voltage measured on the cell terminals is different from VOC (that indicates the SOC difference).

From the second challenge presented, it can be determined that ΔVOC algorithm cannot operate under load condition as the voltage values vary. Furthermore, as during balancing action there's a voltage drop, the algorithm would have to stop after a certain balance time to evaluate if more balancing is needed or not.

ΔVOC - R_{int}

Departing from the approach seen on Δ*VOC*, there's an equation that permits to estimate the *VOC* of the cell under operation and balancing.

$$V_{oc} = V_{cell} - R_{int}\left(I_{cell} - M * \frac{V_{cell}}{R_{bal}}\right) \tag{34}$$

M is 1 if the switch for balancing the cell is triggered. The current that flows though the cell is decremented in case there is balancing action. This equation permits the application of balancing under operation and can continuously evaluate the cell ΔSOC (though the ΔVOC relationship).

The equation is reliant on the R_{int} value of the cell. This value can be measured with special equipment in the production process. This value it's not constant and will change due to aging and temperature. Given the parameters available on the battery pack, the following equation permits to feed the previous one with the resistance value.

$$R_{int} = abs\left(\frac{V_{cell} - V_{cell\ old}}{I_{cell} - I_{cell\ old}}\right) \tag{35}$$

This equation must be called when there's a step on I_{cell}. Not having a continuous calculation of the R_{int} value does not affect the algorithm. It is expected that, on a normal operation on a vehicle, there are multiple steps on battery current. Furthermore, the dynamic in which the R_{int} evolves is slow.

Theorical time to balance

The time to balance that would be needed for a given cell is the following:

$$time\ to\ balance\ (h) = \frac{C_{to\ balance}\ (Ah)}{I_{bal}(A)} \tag{36}$$

Where:

$$C_{to\ balance} = C_{nom} * (SOC_{to\ balance}) \tag{37}$$

The SOC to balance could be determined with the difference in open circuit voltage together with a good Voltage-SOC table. However, it cannot be extracted directly with the algorithm inputs. A workaround would be to establish a linear relationship with the voltage:

Example with $V_{Max} = 4.3\ V$ and $V_{Min} = 2.8\ V$

$$Slope = \frac{V_{Max} - V_{Min}}{SOC(100-0)} = \frac{4.3 - 2.8}{1} = 15\frac{mV}{1\%\ SOC} \tag{38}$$

To calculate the difference of SOC between two cells:

Example with $V_{Cell\ high} = 3.95\ V$ and $V_{Cell\ Low} = 3.85\ V$

$$\Delta SOC\ \% = \frac{V_{Cell\ high} - V_{Cell\ Low}}{Slope} = \frac{3.95 - 3.85}{15 * 10^{-3}} = 6.6\ \% \tag{39}$$

This workaround could work but linearizing voltage to SOC it's not valid with a Li-Ion chemistry. It could be applied if the chemistry presents a fully linear voltage to SOC curve.

3.3.1.2 Final Voltage

The previous algorithm could be applied in a car even in operation. Final Voltage algorithms on the other hand, are only applied when the battery is being charged and it is in the final part of the charge process. These algorithms are based in the same HW presented in Figure 15.

In a vehicle application, it is needed that the owner of the vehicle keeps the vehicle connected to the Power Supply. This cannot always happen, so the balancing action could not happen for a long time. Not being able to perform balancing action for a long operational lime could led to a decay in vehicle's range.

These algorithms rely on the cell's voltage differences to establish a relationship between its SOC and choose which cells need balancing. As stated on the previous section, with voltage-based algorithms, the time to balance cannot be obtained only directly with the algorithm.

ΔVOC Final voltage

This algorithm is an application of the *ΔVOC*-R_{int} algorithm seen on the previous section. The difference with it is that *ΔVOC Final voltage* algorithm only works when the battery has finalized the charge process and the power supply is still connected to the battery.

Input parameter of the algorithm:

- Cell voltages
- Cell current

The algorithm conducts the R_{int} calculation for each cell and stores its value. When the charge process ends (and the supply is maintained to the battery pack) it compares the cell voltages. In case there is a cell that needs to be discharged, its balancing circuit is triggered, and charge is resumed with an small current value.

The current for this charge process is low, and its function of the highest cell voltage value and the balancing resistance. The objective is to make the net current that flows through the most charged cells equal to 0 A. With this, those High-Charged cells won't gain energy while the less charged will increase its SOC.

$$I_{cha} = \frac{Vcell_{max}}{R_{bal}} \quad (40)$$

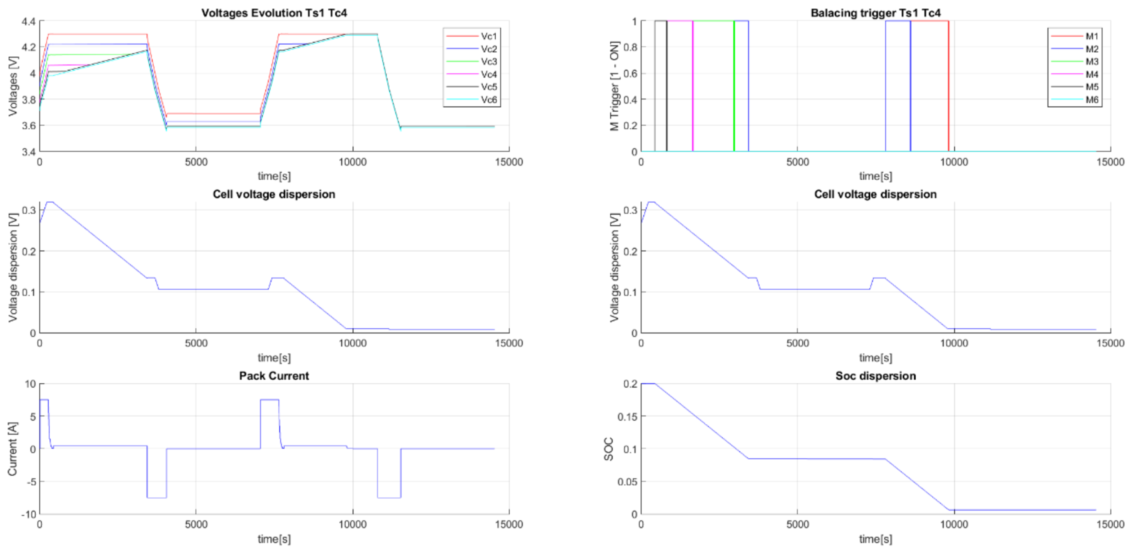


Figure 16. Various battery curves. ΔVOC Final voltage algorithm applied. Initial SOC:

Cell 1: 80 %; Cell 2: 75%; Cell 3: 70%; Cell 4: 65%; Cell 5: 62%; Cell 6: 60%.

To choose if a cell needs to be balanced, the difference in voltage is used. This is where the relationship with *ΔVOC* - R_{int} appears. The algorithm calculates the ΔVOC of the cell and, with the comparison between them, establishes which one's balancing circuit needs to be triggered.

Figure 16 shows the application of the *ΔVOC - Final voltage* on a 6-cell group. It has been forced with a large SOC Difference between cells to check its behaviour. First, notice that the battery current is not 0 when relaxation time is applied after the charge cycle.

On the voltage plot the cell's voltage ends up matching, as the lowest cell is being charged and no energy is flowing into the others. On the right figure there's the triggering of the different mosfets. Notice that the first triggering could be considered as only M5 is triggered. However, all mosfets except M6 are engaged (M5 plot shadows the other ones). Voltage curves help to confirm the last statement as all the voltages are kept almost flat.

Cells 2 to 5 while their mosfet is triggered are being charged a little bit. This is because their voltage is different from the highest cell one, so the current that flows into the resistor branch is lower than the one applied on the battery (41). So, there is net current flowing into the cell. That net current entering the cell would cause a voltage step on the cell's terminals, affecting the voltage measurement. This scenario shows why it is needed to rely on the *ΔVOC* - R_{int} calculation over a calculation with only the *ΔVOC.* As with only with the Voltage seen on terminals, the algorithm would base its triggering management with a variable that it is not true to the cell actual state.

$$I_{cha} = \frac{Vcell_{max}}{R_{bal}} > \frac{Vcell_{2\ to\ 5}}{R_{bal}} \tag{41}$$

(41) show that the current that would flow into the resistor branch of cells 2 to 5 would be lower than the current applied to the battery as is derived from the current calculated with $Vcell_{max}$.

Pulsed Charge

Input parameter of the algorithm:

- Cell voltages

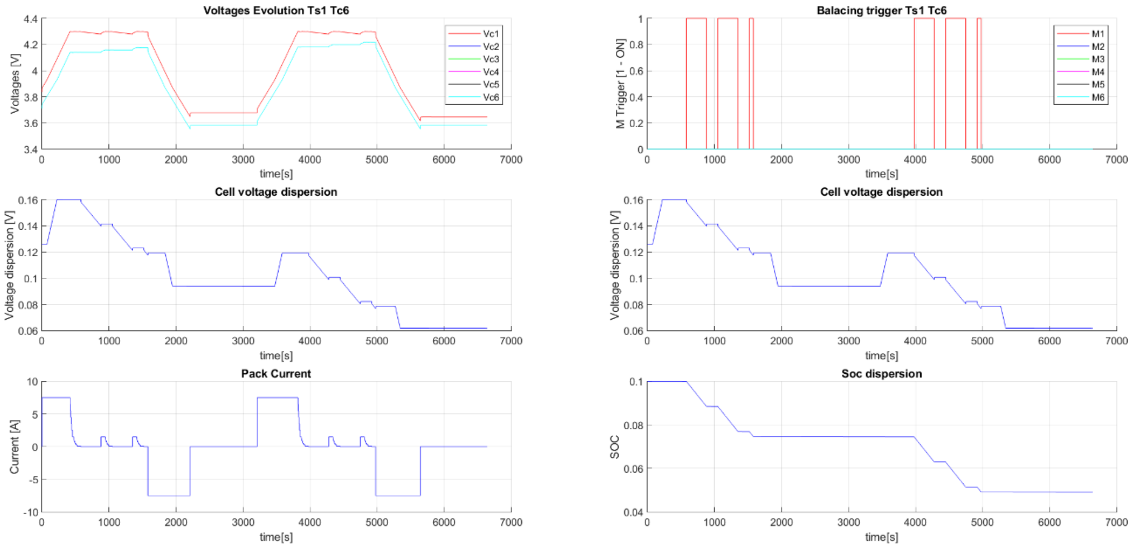


Figure 17. Various battery curves. Pulsed Charge Final voltage algorithm applied.

When a battery reaches the final part of the charge process and finishes the CV part it's when the balancing action starts. The algorithm evaluates the voltage of the cells and starts discharging the ones that present a higher voltage. After a certain amount of time (configurable parameter of the algorithm) the discharging action stops. Then, as the cells which were in a higher SOC have been discharged a little bit, the charge can be resumed (at a lower rate). The charge is expected to be short as the difference of SOC has been minimum. After the charge is stopped the process is resumed repeatedly until the overall battery pack or module has been balanced.

We could apply more balancing action time or establish a calculation of how many time the balancing action shall be applied to each of the cells to reduce the number of charging processes. However, it could subtract too much energy out of the battery pack, reducing excessively the range in case the owner of the vehicle decides to remove the supply to the vehicle.

One design parameter of this algorithm is the minimum power that the charger may be able to deliver. By charging at low current, the OBC or the external charger may enter in a region in which it decides to stop and not deliver more current to the battery pack because of stability and efficiency. The balancing action time may be needed to be to be established in concordance to the charger characteristics.

Figure 17 show the application of this algorithm into a 6-cell grouping. On the left, it can be seen the different voltages of the cells, the difference between the voltages of the cells, and the battery current. On the right it can be seen the mosfets that enable the balancing action, the maximum voltage difference, and the maximum SOC difference between cells.

As per initial conditions, cell 1 is in a 10% SOC higher capacity in respect to the other cells. The test illustrates a total of 2 charge and discharge cycles with 1000 s relaxation time when the battery achieves full charge or full discharge. It can be seen how the overcharged cell marks the end of the charge process; it reaches the maximum cell voltage. Then on discharge, the other cells limit the discharge process.

When the charge ends, the balancing action is starts, M1 (mosfet that triggers the balancing on cell 1) is triggered. After a configured time, it stops balancing and the charge is resumed (see the pack current little spikes in groups of 2). Despite there is balancing action ongoing, the discharge process starts, so the balancing is stopped. Cell voltage dispersion and SOC dispersion show that the balancing is effective. In the long term the cell would achieve the same SOC value as the other 5 cells.

3.3.1.3 SOC History

SOC history algorithm is the first algorithm seen that, instead of relying on voltage as a mean of compare SOC, it relies entirely on SOC estimation. To achieve balance it has a big dependence on a good estimation for each of the cells. This algorithm is deployed on the same resistive balancing circuit seen on Figure 15.

Input parameters of the algorithm:

- Each cell SOC estimation
- Capacity of the cell

The algorithm tries to equalize the cells to the same SOC value. SOC can be directed linked with the capacity of the cell and permits to estimate the time to balance for each of the cells.

The algorithm first calculates the SOC that must be extracted for each one of the cells. This is performed by comparing the SOC of the cell with the lowest SOC of the cell grouping. In case that there's a difference above a determined threshold balancing is enabled, and the following calculations are performed. From the SOC difference, and with the capacity of the cell, the capacity that needs to be extracted from the cell can be determined (Ah). From there, and together with the balancing current, the time to balance can be determined.

$$SOC_{to\ balance} = SOC_{cell} - SOC_{min} \tag{42}$$

$$C_{to\ balance}(Ah) = C_{real} * (SOC_{to\ balance}) \tag{43}$$

$$time\ to\ balance\ (h) = \frac{C_{to\ balance}\ (Ah)}{I_{bal_{max}}(A)} \tag{44}$$

$$I_{bal_{max}}(A) = \frac{V_{cmax}}{R_{bal}} \tag{45}$$

The balancing current used for calculation is the worst case (cell with maximum voltage). This is considered the worst case because with a higher balancing current considered the time to balance decreases. It's preferable to have to recalculate the time to balance rather than over-discharge the high charge cell.

In Figure 18 the SOC History algorithm has been applied. Just on the start it calculates the time required to balance for each of the cells (*Time to balance* from Figure 18). From then, the balancing is applied despite being the battery in operation. On 5000 s, the last cell associated switch is stopped and then is engaged again. It can be seen how a new time to balance is calculated.

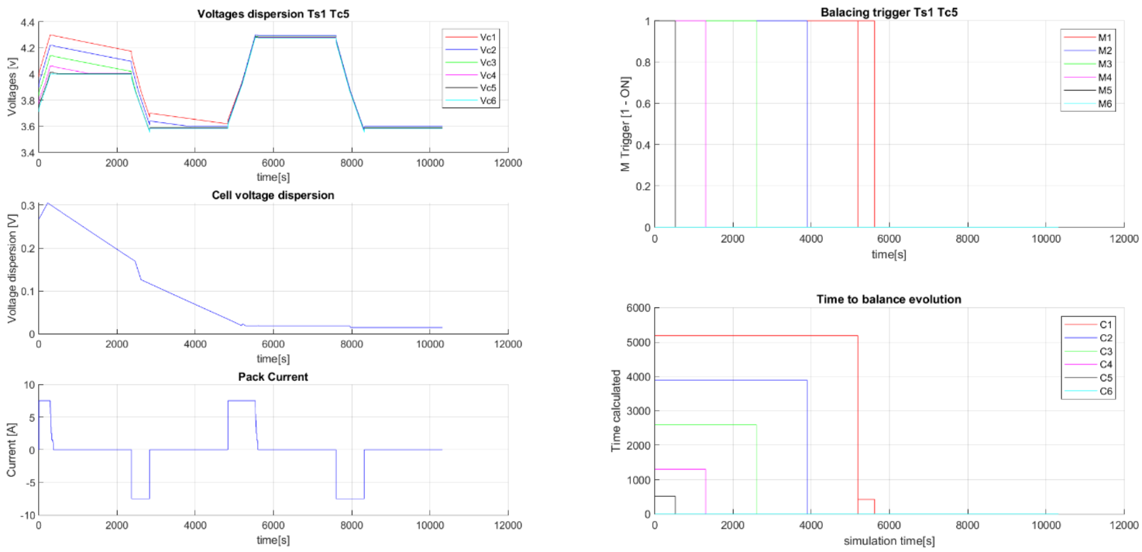


Figure 18. Various battery curves. Soc History algorithm applied.

The new need for calculation shows that the time estimation has been under calculated the first time. This need for recalculation could be linked with the fact that on the formula the I_{bal} considered is the maximum.

Figure 19 show the same algorithm with the same battery conditions. There's been no need for a recalculation. The difference with the previous algorithm it's been the time where the time calculation is performed. In this case is calculated after 1000 s of being the cell relaxed on the top charge setpoint. In a real application this delay on the time calculation could be useful to let the SOC estimation algorithm work and to let the cell relax to get measurements when the cells are stable.

On a real field application, multiple approaches could be taken regarding the moment to calculate the balancing times.

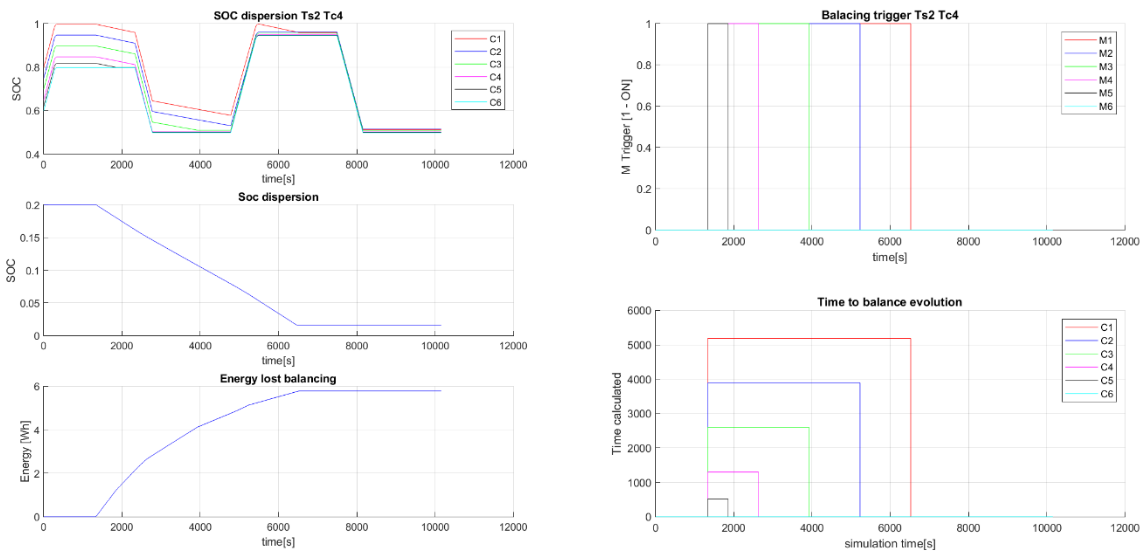


Figure 19. Various battery curves. Soc History algorithm applied start after 1000 s of relaxation.

3.3.2 Capacitor Based Active Methods

On the following section of the thesis one capacitor Based Active Balancing method is reviewed and different topologies are presented.

In the previous section, the way to remove the excess energy present on a cell was by dissipating its energy. Active techniques have intention to perform energy transfer from the cells that have more energy to the ones that have less charge.

Briefly, capacitor based active techniques rely on the use of a capacitor to store and give energy.

Different topologies

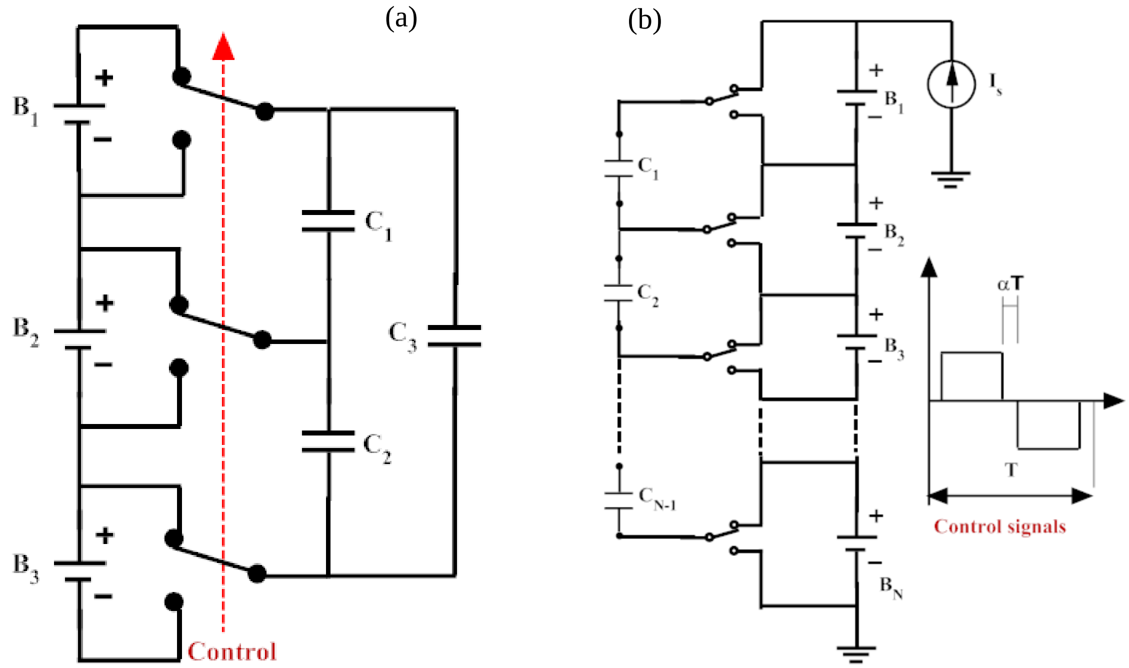


Figure 20. (a) Double-Tiered switched capacitor. (b) Switched capacitor technique.

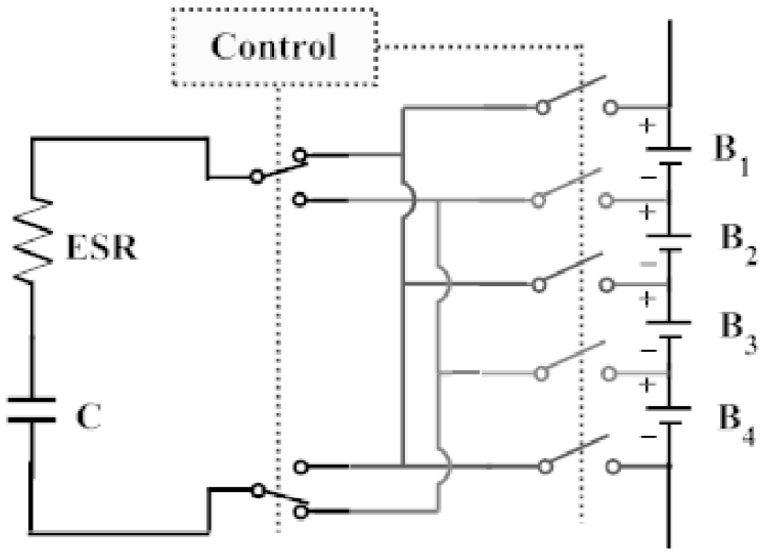


Figure 21. SSC circuit.

There are different topologies seen on literature. Following is a list of each method with the number of switches and other components needed. Each one of the values is kept in function of the number of cells.

Table 2. Number of components for the different Switched Capacitor techniques.

Balancing Method	N of switches	N of condensers	N of resistors
Switched Capacitor Figure 20 (a)	$2n_c$	$n_c - 1$	0
Double-Tiered SW Capacitor Figure 20 (b)	$2n_c$	n_c	0
SSC (Figure 21)	$n_c + 5$	1	0
Dissipative balancing (To provide reference)	n_c	0	n_c

The first two methods require many switching elements and capacitors. The number of capacitors by themselves is not an issue (increase on cost neglected). However, having so many switching elements, so many paths for energy transfer, make the two systems not feasible on an EV application.

The last statement is made thinking on safety concerns. The number of paths and possible operational combinations combined with the fact that vehicle applications are harsh environments, could lead to an unsafe situation like a loss of isolation or a short circuit between paths.

In resume, the benefits that may present these two techniques are not enough to compensate its drawbacks. Its implementation is not covered under this thesis.

3.3.2.1 Single Switch capacitor

In contrast of other capacitor-based techniques, SSC (Single Switch Capacitor) does not present many switching elements. From my point of view is the only technique that is currently feasible to be deployable into battery packs in certain conditions.

In Figure 21 the circuit that SSC requires is presented. For this technique n + 5 switches are required. As can be seen, there two different "rails". One for the plus connection of the even cells and one for the plus connection of the odd cells. Then, there are a group of switches that choose whether the plus part of the condenser shall be connected to the first or second rail.

If we go to the basics and we check the behaviour of an R-C circuit being charged:

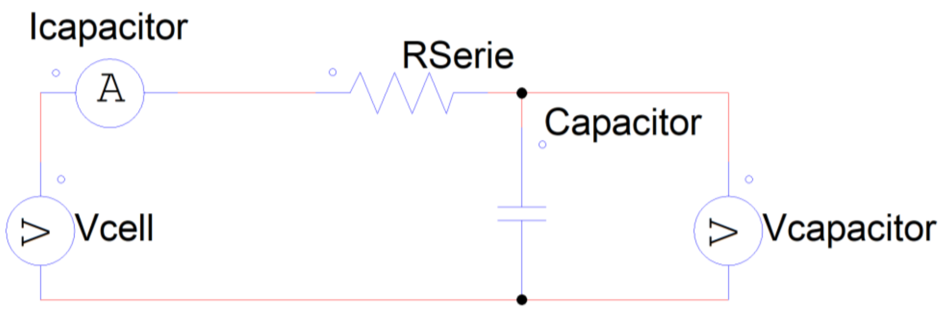


Figure 22. R-C Circuit.

The equation that regulates the voltage of the condenser is the following:

$$V_c(t) = V_c(0) + \left(V_{cell} - V_c(0)\right) * \left(1 - e^{\frac{-t}{\tau}}\right) = V_c(0) + \left(V_{diff}(0)\right) * \left(1 - e^{\frac{-t}{\tau}}\right) \quad (46)$$

$$\tau = R * C \tag{47}$$

As:

$$i_c = C * \frac{dV_c}{dt} \tag{48}$$

Then:

$$V_c(t) = V_c(0) + \left(V_{diff}(0)\right) * \left(1 - e^{\frac{-t}{\tau}}\right) \rightarrow \frac{dV_c}{dt} = \frac{V_{diff}(0)}{\tau} e^{\frac{-t}{\tau}} \tag{49}$$

$$i_c = C\frac{V_{diff}}{RC} * e^{\frac{-t}{\tau}} = \frac{V_{diff}}{R} * e^{\frac{-t}{\tau}} \tag{50}$$

With the expressions of the voltage and current of the capacitor, the energy can be extracted:

$$\begin{aligned} E_{cap} &= \int_0^{DT} i_c * v_c * dt = \int_0^{DT} \frac{V_{diff}}{R} e^{\frac{-t}{\tau}} * \left(V_c(0) + V_{diff} * \left(1 - e^{\frac{-t}{\tau}}\right)\right) dt = \\ &= \frac{V_{diff}}{R} \int_0^{DT} \left(V_c(0) e^{\frac{-t}{\tau}} + V_{diff} * e^{\frac{-t}{\tau}} \left(1 - e^{\frac{-t}{\tau}}\right)\right) dt = \\ &= \frac{V_{diff}}{R} \int_0^{DT} V_c(0) e^{\frac{-t}{\tau}} dt + \frac{V_{diff}}{R} \int_0^{DT} V_{diff} * e^{\frac{-t}{\tau}} * \left(1 - e^{\frac{-t}{\tau}}\right) dt \end{aligned} \tag{51}$$

If the integral calculation is made separately:

$$\begin{aligned} A &= \int_0^{DT} V_c(0) e^{\frac{-t}{\tau}} dt = \left[-\tau * V_c(0) e^{-\frac{t}{\tau}}\right]_0^{DT} = \tau * V_c(0) - \tau * V_c(0) e^{-\frac{DT}{\tau}} \\ B &= \int_0^{DT} V_{diff} * e^{\frac{-t}{\tau}} * \left(1 - e^{\frac{-t}{\tau}}\right) dt = V_{diff} \int_0^{DT} \left(e^{\frac{-t}{\tau}} - e^{\frac{-2t}{\tau}}\right) dt = \\ &= V_{diff} \left[-\tau e^{\frac{-t}{\tau}} + \frac{\tau}{2} e^{\frac{-2t}{\tau}}\right]_0^{DT} = V_{diff} \left(\frac{\tau}{2} e^{\frac{-2DT}{\tau}} - \tau e^{\frac{-DT}{\tau}} - \frac{\tau}{2} + \tau\right) \end{aligned} \tag{52}$$

The expressions are combined:

$$\begin{aligned} E_{cap} &= \frac{V_{diff}}{R}(A + B) = \\ &= \frac{V_{diff}}{R}\left(\tau * V_c(0) - \tau * V_c(0) e^{-\frac{DT}{\tau}} + V_{diff} \left(\frac{\tau}{2} e^{\frac{-2DT}{\tau}} - \tau e^{\frac{-DT}{\tau}} - \frac{\tau}{2} + \tau\right)\right) = \\ &= C * V_{diff}\left(V_c(0) - V_c(0) e^{-\frac{DT}{\tau}} + V_{diff}\left(\frac{e^{\frac{-2DT}{\tau}}}{2} - e^{\frac{-DT}{\tau}} + \frac{1}{2}\right)\right) = \\ &= C * V_{diff}\left(V_c(0) - V_c(0) e^{-\frac{DT}{\tau}} + V_{diff}\left(\frac{e^{\frac{-2DT}{\tau}}}{2} - e^{\frac{-DT}{\tau}} + \frac{1}{2}\right)\right) = \\ &= C * V_{diff}\left(\frac{V_{diff}}{2} e^{\frac{-2DT}{\tau}} - V_{in} e^{-\frac{DT}{\tau}} + V_c(0) + \frac{Vdiff}{2}\right) \rightarrow \\ E_{cap} &= C * V_{diff}\left(\frac{V_{diff}}{2} e^{\frac{-2DT}{\tau}} - V_{in} e^{-\frac{DT}{\tau}} + \frac{Vdiff}{2} + V_c(0)\right) \end{aligned} \tag{53}$$

This would be the energy that the capacitor would accumulate during one pulse. With the multiplication of the frequency the value after one hour could be extracted.

$$E_{cap} = C * V_{diff}\left(\frac{V_{diff}}{2}\, e^{\frac{-2DT}{\tau}} - V_{in}e^{-\frac{DT}{\tau}} + \frac{Vdiff}{2} + V_c(0)\right)\left(\frac{Ws}{pulse}\right) \qquad (54)$$

$$E_{cap}\left(\frac{Wh}{h}\right) = f * C * V_{diff}\left(\frac{V_{diff}}{2}\, e^{\frac{-2DT}{\tau}} - V_{in}e^{-\frac{DT}{\tau}} + \frac{Vdiff}{2} + V_c(0)\right) \qquad (55)$$

Theoretically, it would be interesting to get the expression of the energy extracted from the highest voltage cell, and the energy delivered to the lowest value one. As only there's a resistor between the condenser and the cell, the energy extracted from the cell could be described as:

$$E_{extracted=}E_{resistor} + E_{cap} \qquad (56)$$

$$\begin{aligned} E_{resistor} &= \int_0^{DT} i_c^2 * R * dt = \int_0^{DT}\left(\frac{V_{diff}}{R} e^{\frac{-t}{\tau}}\right)^2 R * dt = \\ &= \not{R} * \frac{V_{diff}^2}{R^{\not{2}}}\int_0^{DT} e^{\frac{-2t}{\tau}} dt = \frac{V_{diff}^2}{R}\left[-\frac{\tau}{2} e^{\frac{-2t}{\tau}}\right]_0^{DT} = \\ &= \frac{V_{diff}^2}{R}\left(-\frac{\tau}{2} e^{\frac{-2DT}{\tau}} + \frac{\tau}{2}\right)\left(\frac{Ws}{pulse}\right) \end{aligned} \qquad (57)$$

For the energy that the low voltage cell would receive:

$$E_{rec=}E_{cap-}E_{res} \qquad (58)$$

It must be taken into account that in this case, the V_{diff} would be defined as:

$$V_{diff} = V_{in} - V_c(0) \rightarrow V_c(0) > V_{in} \qquad (59)$$

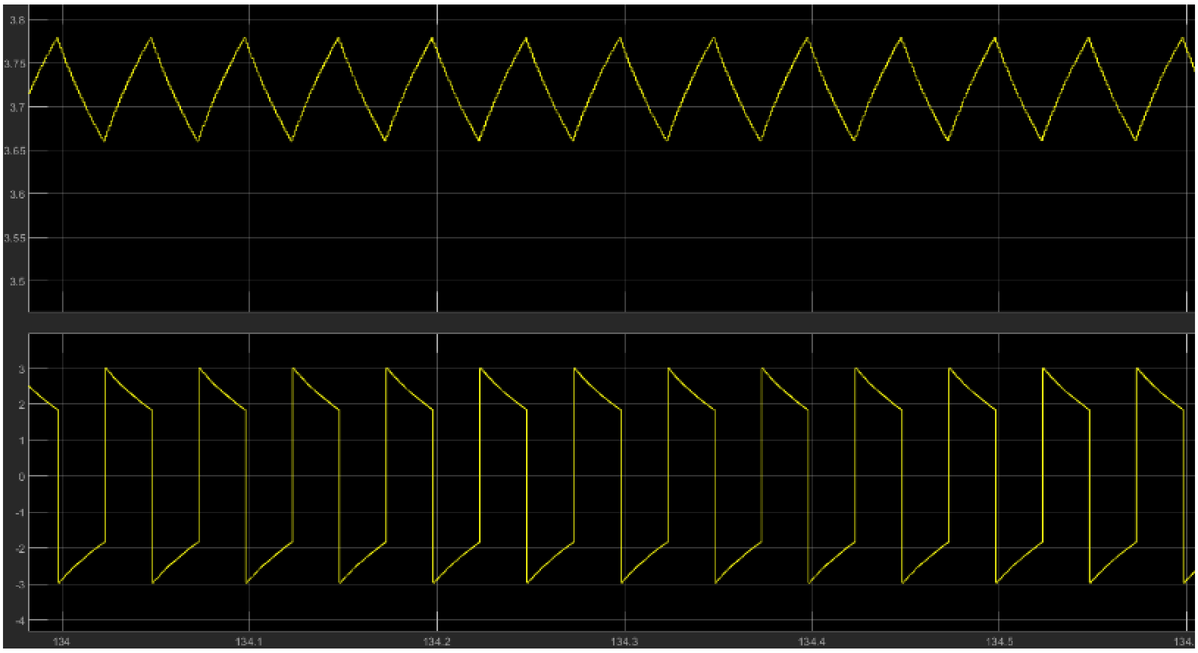


Figure 23. SSC Circuit behaviour. Top: Condenser Voltage. Bottom: Condenser Current.

The implementation of the SSC algorithm is shown in section 4.2.4.1, Figure 23 and Figure 24 illustrate **qualitatively** the behaviour of the SSC circuit in two modes of operation to provide an example for this section.

In Figure 23, DT is lower than the constant τ. It can be observed how the current does not decrease to 0 before the switching happens. The voltage is also not stabilized to the top cell voltage nor the lowest one. The duty cycle chosen has been 0.5. This way of operating, on DTs lower than τ increases the energy transfer.

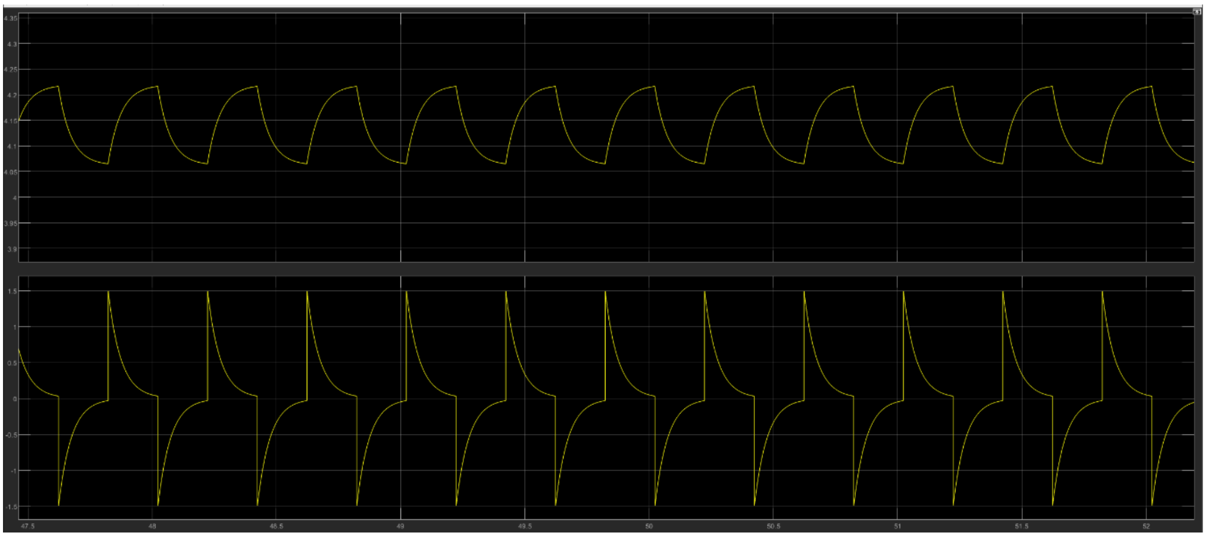


Figure 24. SSC Circuit behaviour. Top: Condenser Voltage. Bottom: Condenser Current.

In Figure 24 DT is above the constant τ. It can be seen how the current almost stabilizes to 0 A. The voltage shows the typical behaviour of an R-C circuit being charged and discharged. The selection of DT has been 25 ms for Figure 23, and 200 ms in Figure 24. Both cases are with a R of 0.1 Ω and a C of 0.5 F (τ=50 ms).

3.3.3 Aging topic

Balancing action needs to extract current from the cell at a given current rate. In the case of SSC, under operation, cells receive and give energy. With this, we are introducing an important number of discharge and/or charge cycles into the cells.

At lower current rates and with low deep of discharge (DOD) it is known that the aging effect does not affect so much the cells. In general, we could say that the benefits of those balancing techniques compensate the possible acceleration of aging due to balancing.

However, in the case of SSC, in which there's a periodic switching phenomenon, aging would need to be studied with real physical data.

4 Simulation

4.1 Model Basics

All the simulations and the implementation of the balancing algorithms have been done in MATLAB environment. It's one of the best software in its kind and offers a lot of options and add-ons to simulate.

4.1.1 General Description

The different models implemented can be structured in multiple layers. The modelling action has been performed with MATLAB and Simulink, and inside Simulink add-ons have been used.

MATLAB Layer

The MATLAB layer is the start and the end of the model execution. In this layer the preconditions of the model are created, and all the parameters needed for simulation are stablished.

When the simulation ends, the execution path returns to MATLAB layer to perform the report of the results. Then, if certain conditions are fulfilled, another simulation is performed. In sections 4.1.4 and 4.1.5 the two MATLAB layer important parts are further explained.

Simulink Layer

This is where the model is created, and the elements are created and interconnected. The constants and variables created on the MATLAB layer are used to initialize the values of certain parameters as well as to establish some constants on the model.

The parameters that interest for reporting are exported to the MATLAB environment to make the necessary plots for the different methods.

Some Simulink tools have been used on the modelling part. Apart from the usual blocks as constants, *Goto*, *From* and Scopes, also *MATLAB Function Blocks* have been used.

On the MATLAB Function Blocks is where in some models the algorithm is coded. On each of these functions a series of input parameters and output parameters from the Simulink environment are connected. The coding of the functions is like C language with differences in the syntaxis.

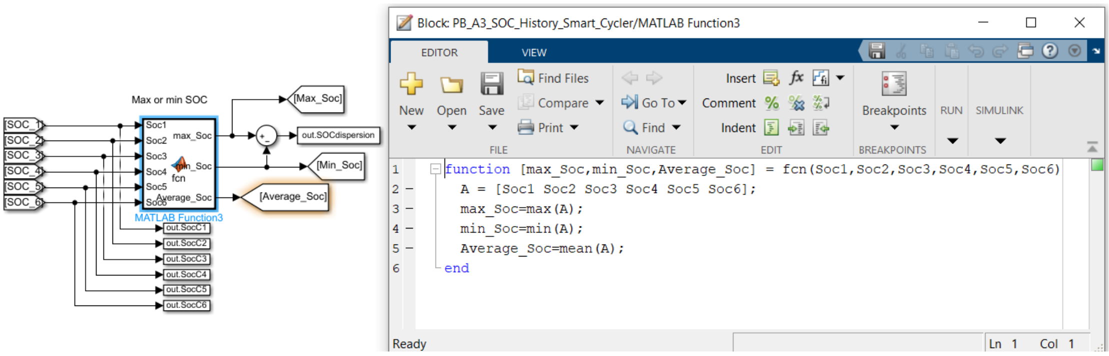


Figure 25. MATLAB Function Block and it's code example.

In Figure 25, it can be seen the inputs to the function on the left (a *From* block has been used). Then this input is entered into the function and after the coded calculations are made, a result is present on the output. The figure permits to see the out.x block. This block creates a copy of the simulation content to the MATLAB environment.

Simscape

This addon contains a series of electrical elements such as resistances, capacitors, battery cells and switches to ease the simulation task. It has been used in all the models to create the battery model as well as the balancing circuit.

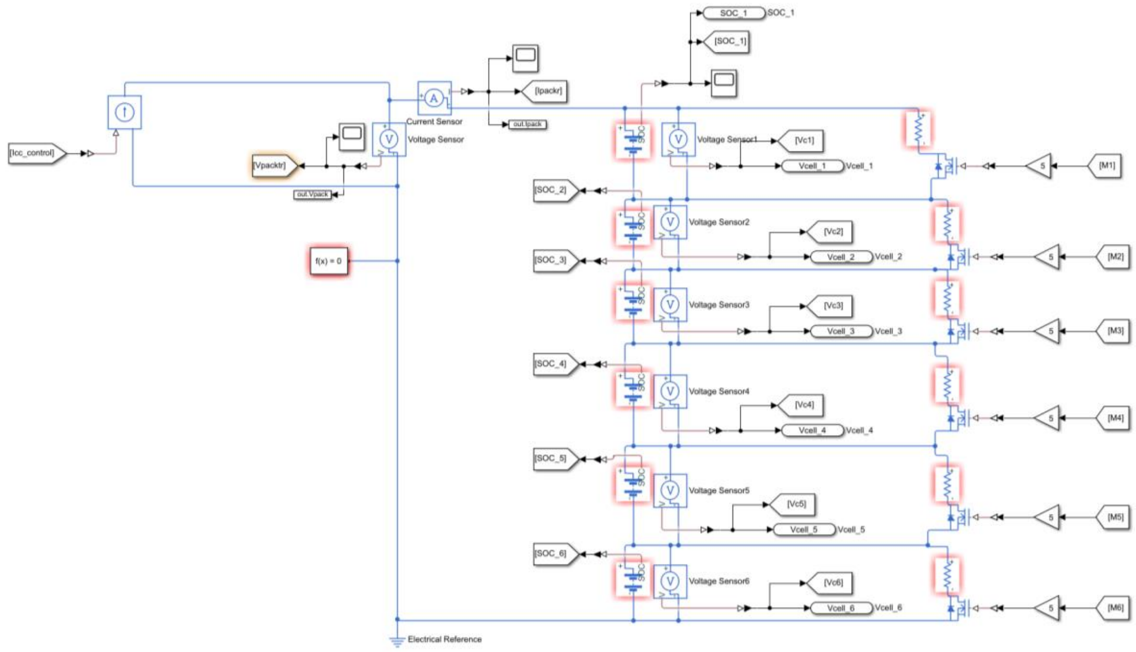


Figure 26. Implementation of the Battery Module with Simscape.

State Flow

This addon has only been used in certain models that require high complexity. It permits to create Finite State Machines (FSM) in a graphical way. Without this method, those FSM would need to be created with C code adding unnecessary complexity and making more difficult the code debugging and comprehension task.

These FSM’s take some inputs from the model. These inputs are evaluated on each of the states to check if a transition is needed. On each of the states, the output values to the model are updated.

In Figure 27 part of a state flow chart is shown. In charts there is hierarchy. For example, the state called Evaluate contains substates. When the Evaluate state is selected, it enters to the High state, as it’s defined as the initial one. Entry and during, or **en, du** describe actions that are performed when the state **en**ters in active mode and **du**ring it.

The transition between states is performed with an arrow. The conditions to transition are under []. The transition can be caused by a timer ([after (x,msec)]), by an internal variable (**like count**) or due to input values to the chart.

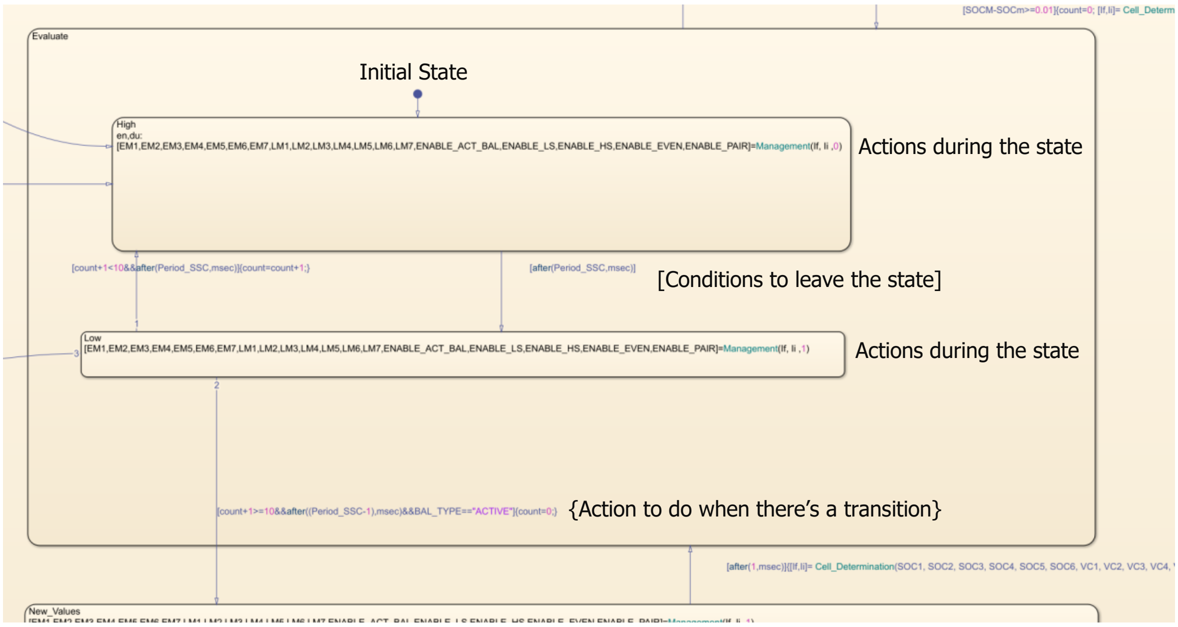


Figure 27. Part of an Stateflow Chart present on SSCM2 model.

When a transition it is enabled, an action can be performed. It is described under {}.

Resolution Times

The different models work in preconfigured discrete step times. The times have been selected to fasten simulation on some cases. The functions and state-flow charts inside the model have its execution time also. In the following table the times selected are presented.

Table 3. Resolution times on the different models.

Model name	*Battery model time*	*Fastest function*
PB_A1_Voltage_Based	1 s	10 ms (some functions)
PB_A2_AVoc	1 s	10 ms (some functions)
PB_A2_Pulsed_Cha_STF	1 s	10 ms (some functions)
PB_A3_SOC_History	1 s	10 ms (some functions)
PB_A4_SSC	1 ms	1 ms (control)
PB_A4_SSC2M	1 ms	1 ms (control)
PB_A1_Voltage_Based_Profile	0.1 s	10 ms (some functions)
PB_A3_SOC_History_Profile	0.1 s	10 ms (some functions)
PB_A4_SSC_Profile	1 ms	1 ms (control)

The battery model time is the step time that takes the solver of the Simscape to perform the simulation of the battery. In some models there are functions that are faster than the Simscape function. This is intended, as per to have the new iteration variables before the new iteration of the solver.

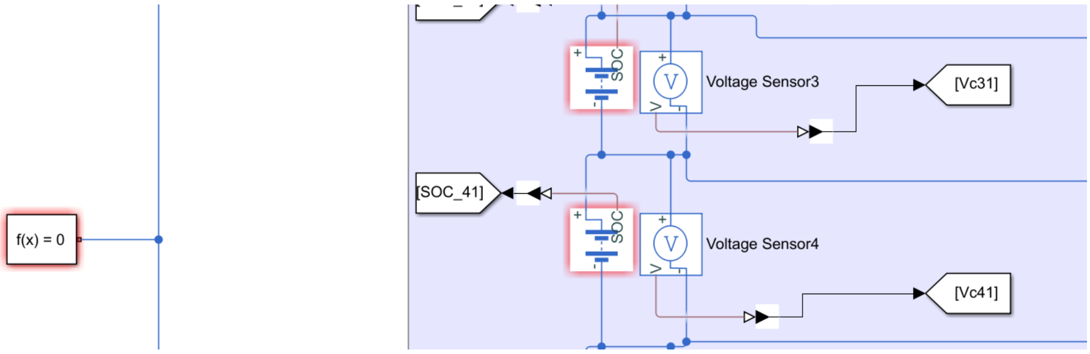


Figure 28. Solver detail on the Simscape elements.

Almost all the models have a solver time of 1s. This optimizes the simulation speed, but the electrical behaviour of the cells has a greater step. In the case of SSC and due to the resources of the project, the simulation is at 1 ms.

This value of 1 ms imposes a design restriction on the SSC values to be simulated. The maximum switching frequency that would translate onto a change on the electrical model is 1 kHz.

4.1.2 Cell Modelling

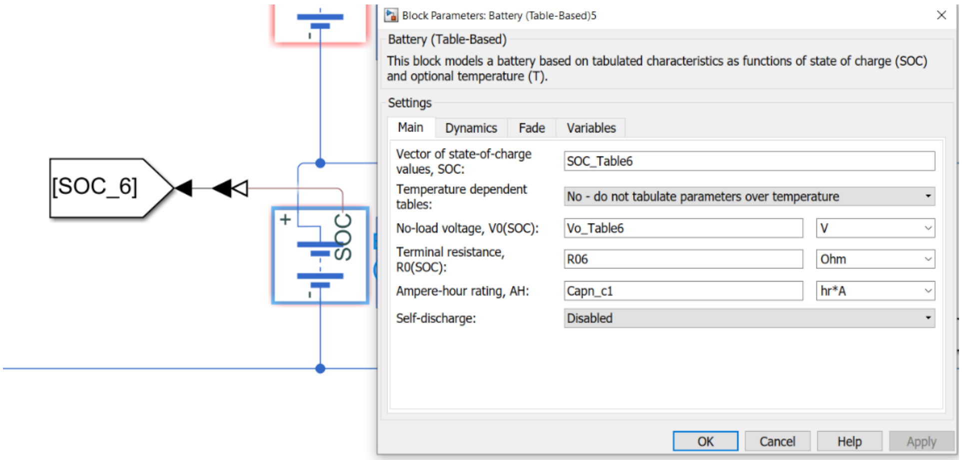


Figure 29. Cell Model with its parameters.

The block that has been used to model cells in Simulink is the *Battery (Table-Based)* from the Simscape add on. The VOC of this model is calculated from a VOC-SOC table that is an input parameter in the block. This input parameter table is generated in the MATLAB Layer.

Then different parameters are entered, such as capacity, internal resistance, dynamics, as well as the start values. For each of the tests the Initial SOC of the cells is established in the MATLAB Code.

This block in the end implements a second order model of the battery. Despite is a simple way to model the battery, it will ease simulation and provides SOC calculation without additional effort. Furthermore, the VOC-SOC table as well as the other parameters provide model reusability in case that different types of cells are needed to be tested.

4.1.3 Cycle and Profile Generation

The models implemented have two options to test the different algorithms. One it's via a current Profile and the other via a Cycle operation.

Cycle operation

Cycle tests consist in driving a cell from a SOC point to another multiple times, via charging and discharging. The system charges and discharges using the CC-CV technique.

The CC-CV technique is the most used for charging a battery. Its implementation using a state flow chart is shown in Figure 30. The charging method consists firstly of a Constant Current stage. The value of this constant current is specified into the input parameters of the simulation.

When a single cell reaches its maximum allowed voltage the CV area starts. In this area the control loop ensures that the voltage of that single cell does not surpass the top specified voltage. The reduction of the voltage on the cell terminal is done by reducing progressively the current that flows though the cells. The charge is considered finished when the current is close to 0 A.

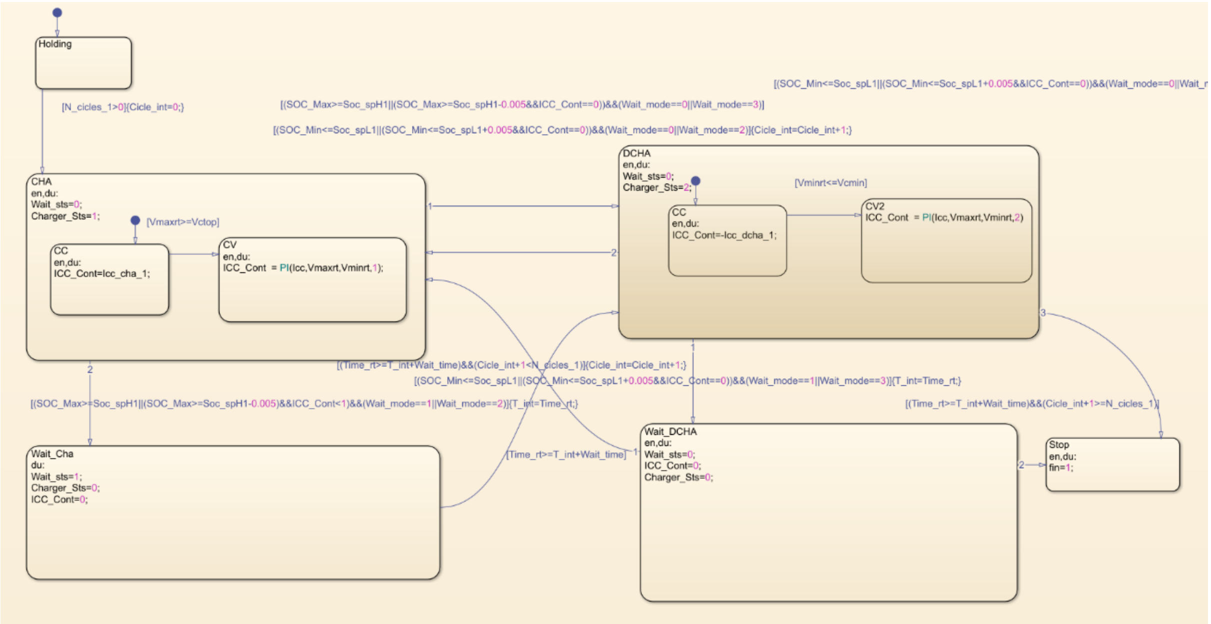


Figure 30. State Flow diagram of the CC-CV charge process.

With this way of charging an overvoltage on a cell is avoided. When discharging, the behaviour is the same but with the lower limit of the cell.

On the cycler, the SOC boundaries for the end of the charge/discharge processes can be chosen. When a cell reaches the SOC boundary the charge/discharge process is finished.

The cycle operation consists of the following states executed in order:

1- Charge until desired SOC (Soc_spH1) is reached with the CC-CV technique.

2- Wait on upper SOC. This option is chosen by an input parameter of the simulation. If is not chosen, after the state 1, the state 3 is executed. In this state, there's a rest time (input parameter) in which no current is flowing into the battery. When the rest time is surpassed the state 3 starts.

3- Discharge until desired SOC (Soc_spL1) is reached with the CC-CV technique.

4- Wait on lower SOC. This option is chosen on the input parameters of the simulation. In this state, there's a rest time (input parameter) in which no current is flowing into the battery.

When the state 3 finishes and state 4 is not enabled, or when state 4 finishes the rest time, an evaluation of the number of cycles executed is made. The number of cycles already executed is compared with the input parameter N_cicles_1. In case that the number of cycles performed is equal to the desired value, the simulation is ended.

Profile operation

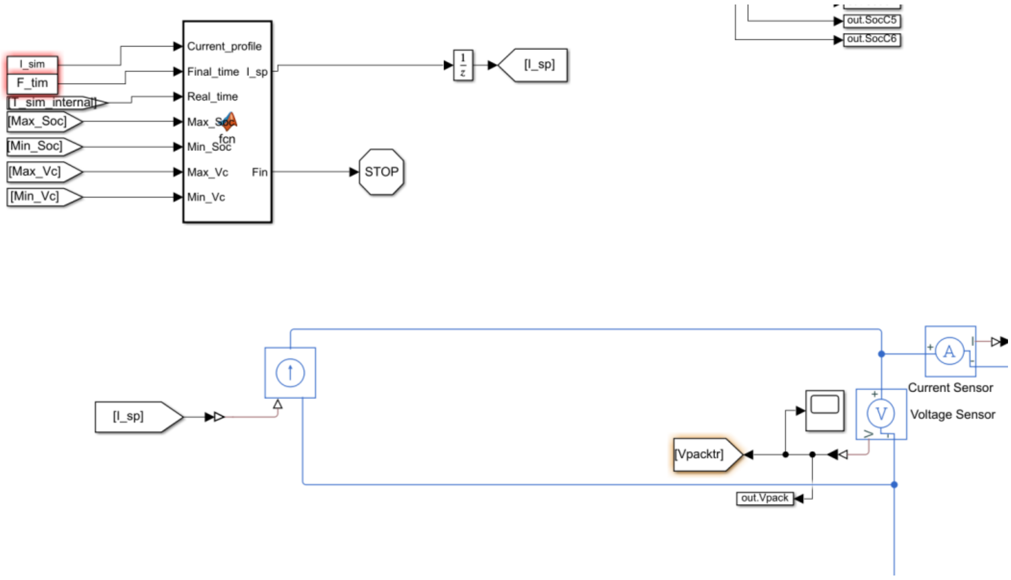


Figure 31. Detail of the Power supply area in the case of profile models.

In the profile testing the battery pack is subjected to a real driving profile extracted from measurement data of a BMW I3. This data, obtained via IEEE is at a sampling frequency of 10 Hz.

One of the variables on the measured data is battery current. This value is the one that is introduced into the battery pack simulation together with its corresponding time value.

For this kind of testing the charger used in the cycle tests is removed. The battery pack will be directly subject the current profile. A protection has been added to avoid over discharge. In the case that a cell reaches minimum value, no discharge current will be allowed, only charging current will be allowed.

Via *Test_cases_1* excel it can be chosen which driving cycle to perform from a repository of more than 30 different profiles. Also, by the N_cicles_1 variable, the same profile can be introduced multiple times.

Table 4. Example of parameter entry for a profile simulation.

Tc	*N_cicles_1*	*Config*	*Soc_spL1*	*Soc_spH1*	*PB_WT*	*Profile*	*NA*	*T_relax*	*Mode_T*
1	*4*	*41*	*50*	*100*	*0*	*1*	*0*	*1000*	*1*

In the case shown in Table 4, if the test case (Tc) 1 is selected, 4 cycles of the profile number 1 would be performed. Between each profile, 1000 s would pass with no current flow.

4.1.4 MATLAB Interface

To perform all the model simulations, there was the need to create an "smart" MATLAB environment to maximize efficiency to perform the different tests.

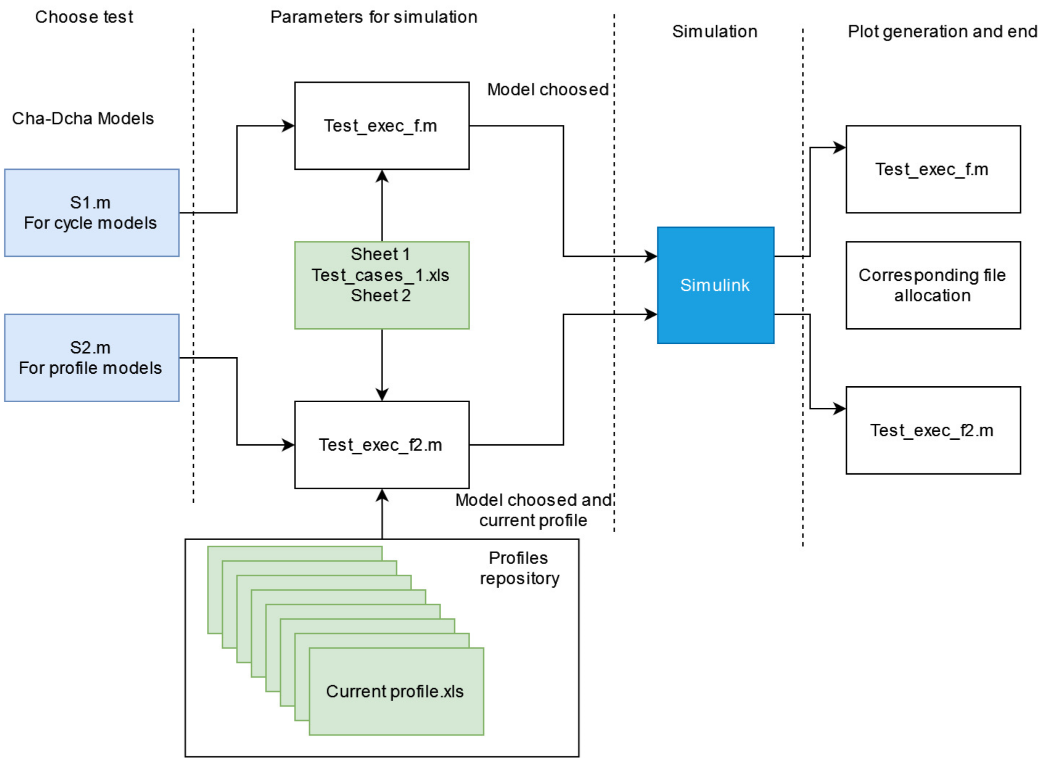


Figure 32. Flow chart of the MATLAB Simulations.

The scripts that generate the simulations are S1.m and S2.m. S1 is for the simulations of cycling and S2.m is for the current profile simulations.

The model parameters are then extracted from the Test_cases_1.xls excel file, as well as initial cell SOC's among other parameters. The excel file also contains which simulation is needed to be performed and with which operation mode (some modes have different configurable behaviours). Then the correspondent model is simulated.

When the simulation ends, the significant parameters are then plotted and stored in the correct path for reviewing the results.

4.1.5 Reporting

Simulations are automatically stopped once the model has reached the end. In that moment, execution path returns to the MATLAB script to perform different plots of the data.

The most significant variables are plotted and stored in order. The order is established by test's index that is contained in SX.m. Cycle results and profile results are stored in independent folders.

4.2 Techniques implemented

On the following points, each of the techniques that are implemented on the modelling part are reviewed in a practical way. First the resulting model circuitry is presented. Then, some remarkable technical concepts regarding SW implementation or HW topics are commented on the technic basis. Finally, the parameters that are used for simulation are chosen.

Battery Values

There are some values that need to be chosen for the cell model:

- Capacity: 10 Ah. This value is approximately 10 times smaller than the values seen on EVs. However, it shortens simulations, and the balancing action is stronger in relation with the cell capacity. So, the simulations will be more visual, and more conclusions can be extracted from them.
- Balancing capability: This will be the maximum allowed current to pass through the tracks of the cells that are connected with the BMS board that is implementing the balancing function.
    - SSC: **1 A.**
    - Passive balance circuit: **0. 1 A.**

  SSC value is higher, as if there's the added cost of implementing this technology, it would be worth to upgrade the cabling to allow more balancing current.

  Just to give reference, for the passive balancing circuit 1000 h would be needed to discharge the full cell. This is considering a linear current flow though the resistor.
- Maximum Voltage: 4.3 V. This is the upper limit of the cell.
- Minimum Voltage: 2.8 V. This is the lowest limit of the cell.
- VOC - SOC curve: This table is the one that is implemented on the battery blocks on the different models.

Table 5. VOC-SOC table implemented.

VOC	2.8 V	3.27 V	3.9 V	4.3 V
SOC	0	25 %	75 %	100 %

- Charging-Discharging rate for cycling tests: Up to 1.5 C (15 A).
- Internal resistance: 1 mΩ.
- First order dynamic resistance: 1.5 mΩ.
- First order time constant: 0.01 s.
- Second order dynamic resistance: 1.5 mΩ.
- Second order time constant: 3 s.

4.2.1 ΔVOC

ΔVOC technique is based on the theoretical concepts explained in the section 3.3.1.1.

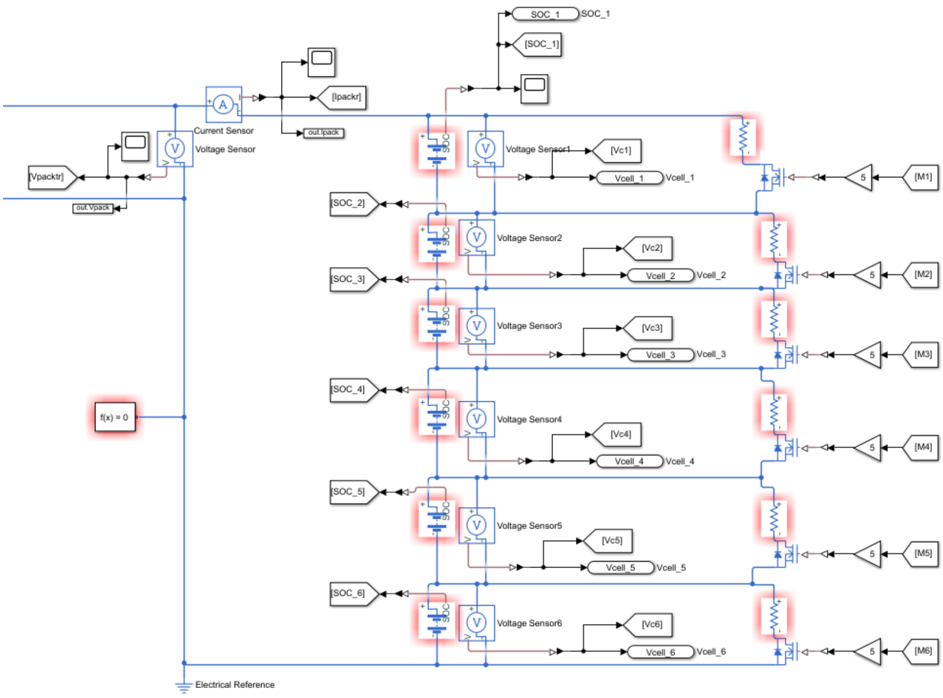


Figure 33. Electrical part of the Model.

4.2.1.1 Implementation

Following are some comments on the implementation of control part of the ΔVOC model. All the control part has been performed with MATLAB function blocks.

R_{int} Calculation

ΔVOC technique assumes that two cells that have a different open circuit voltage are in a different SOC. In this case, the technique is implemented to operate in a continuous way. For this reason equation (60) must be implemented.

$$V_{oc} = V_{cell} - R_{int}\left(I_{cell} - M_{cell} * \frac{V_{cell}}{R_{bal}}\right) \qquad (60)$$

This equation is implemented for each of the cells. It's only performed in case that there is an already calculated value of R_{int}. Following is the code extract for the R_{int} calculation implemented on the code. The calculation is based on equation (35).

Code Extract 1. Example of R_{int} calculation.

```
if (diff([Ipacktr Old_Ipack])>0.5)&&(Ipacktr~=0)

     R1i=abs((VC1-VC1o)/Ipacktr);
     R2i=abs((VC2-VC2o)/Ipacktr);
     R3i=abs((VC3-VC3o)/Ipacktr);
     R4i=abs((VC4-VC4o)/Ipacktr);
     R5i=abs((VC5-VC5o)/Ipacktr);
     R6i=abs((VC6-VC6o)/Ipacktr);
end
```

With the if condition, only big steps on current are considered. With bigger steps there are bigger voltage drops that permit calculate the internal resistance with more ease.

This may seem an easy way to get the internal resistance, though the voltage drop on the resistance for two samples. It could be argued that it could be checked for more samples. However, this method gives good results with low complexity. In a car application current demand is not constant, as the driver may change its inputs to the throttle or brakes. Also, there could be dynamics on the power electronics of the vehicle that could generate an error on this calculation. In this case I've preferred to keep the equation on its simplest form.

Criteria for triggering of the balancing mosfets

To trigger the mosfets of each of the cells, firstly the ΔVOC must be calculated for each of the cells.

Code Extract 2. Example of ΔVOC Calculation.

```
A=[Voc1 Voc2 Voc3 Voc4 Voc5 Voc6];

    MinPack=min(A);
    AVoc1=Voc1-MinPack;
    AVoc2=Voc2-MinPack;
    AVoc3=Voc3-MinPack;
    AVoc4=Voc4-MinPack;
    AVoc5=Voc5-MinPack;
    AVoc6=Voc6-MinPack;
```

And then from this value each of the ΔVOC is compared to a threshold, if a value is higher than the threshold the balancing action is started.

The code of this section is simple but functional. It has been observed on the different simulations that its capable to get correct values of internal resistance as well as it performs balancing until the threshold condition is fulfilled.

4.2.1.2 Parameters chosen for simulation

For the simulation side, the following parameters have been chosen:

- Balancing resistance: $R_{bal} = \mathbf{43\ \Omega}$. With this value the maximum current that would pass through the balancing resistor would be 0.1 A for the maximum voltage of the cell (4.3 V).
- Threshold: 25 mV.

4.2.2 Final Voltage

A total of two models have been implemented for the Final Voltage algorithms. These models are based on the hardware seen in Figure 33. The control part of these algorithms has been made with MATLAB functions in the case of Final Voltage - ΔVOC. In the case of Final Voltage - Pulsed charge, it has been made with a state flow chart.

- ΔVOC: In this model when the charger finishes the charge process, the BMS evaluates the situation. Firstly, the BMS compares the cell states. In case that balancing action is required, the balancing action starts. While the necessary mosfets are triggered a small current is delivered by the charger. That small current is the maximum cell voltage divided by the passive balancing resistor. The strategy of this technique is to do not put additional charge on the highly charged cells while charging the less charged ones.

- Pulsed Charge: In this model when the charge process is finished, the BMS control starts the balancing action if required. During the balancing action, there's no current flow into the battery. So, without balancing action the voltage read on the cell terminals is directly the open circuit voltage.

  After a defined balancing time, the balance action stops. Then there's a new charge process at a low current. When the charge process is finished, the balancing action is evaluated again.

  This process is performed until the cells are balanced or power supply is no longer available for the Battery.

In the case of Pulsed charge, there's an interaction between two Stateflow charts. Figure 34 shows the implementation of the Pulsed Charge charger and balancing control blocks.

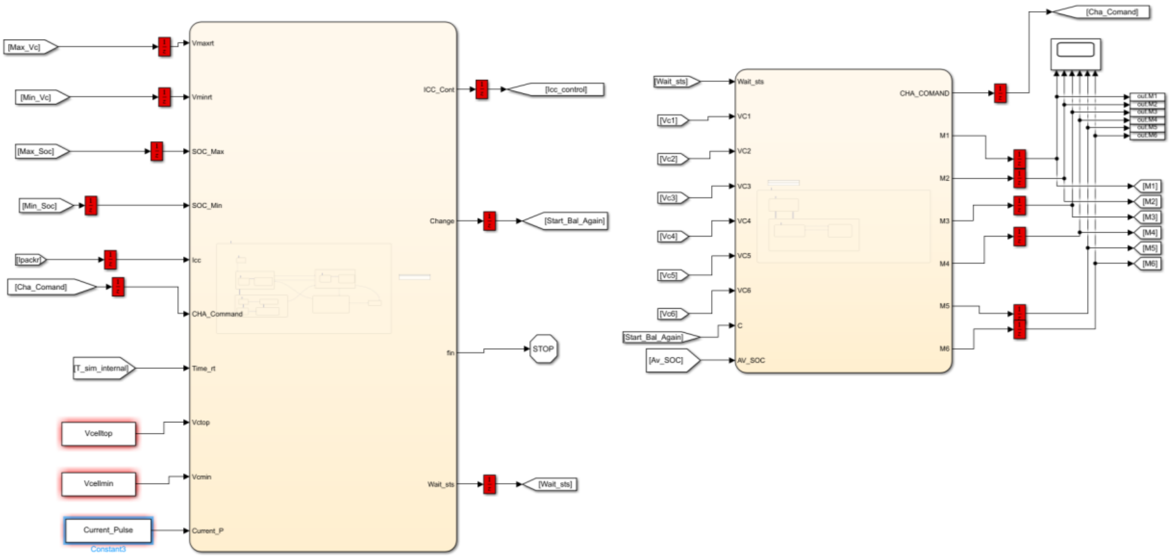


Figure 34. Charger and balancing block interactions.

4.2.2.1 Implementation

ΔVOC

When the charger finishes its charge process it provides a signal to the balancing control system. The system is based by a small charge process so if balancing is needed there will be some current flowing though the battery (determined by the balancing system). The voltage measurement of the cells then, won't reflect the VOC value of the cells. The balancing control system must monitor the cells during all operation to get its internal resistance value. This value is then used to perform the calculation of equation (60).

After the open circuit voltage calculation, a comparison between the cells is made. Together with a threshold, the cells that are needed to be balanced are stablished and triggered. If balancing action is needed, this system then calculates the indicated current to perform the charge process.

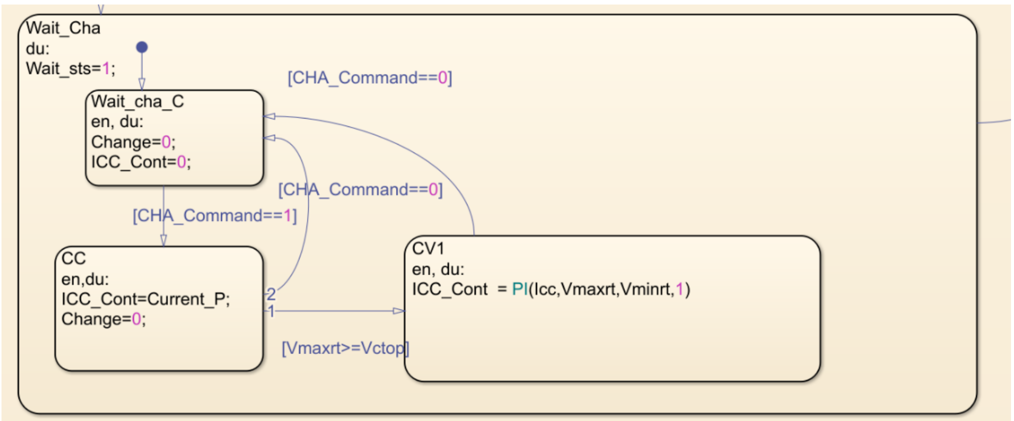


Figure 35. State of the charger when has finalized charge but it still has supply.

The current flow is stopped when the balancing process has finished or the supply is removed from the battery.

Pulsed charge

The pulsed charge implementation also requires a good communication between the charger and the balancing control action.

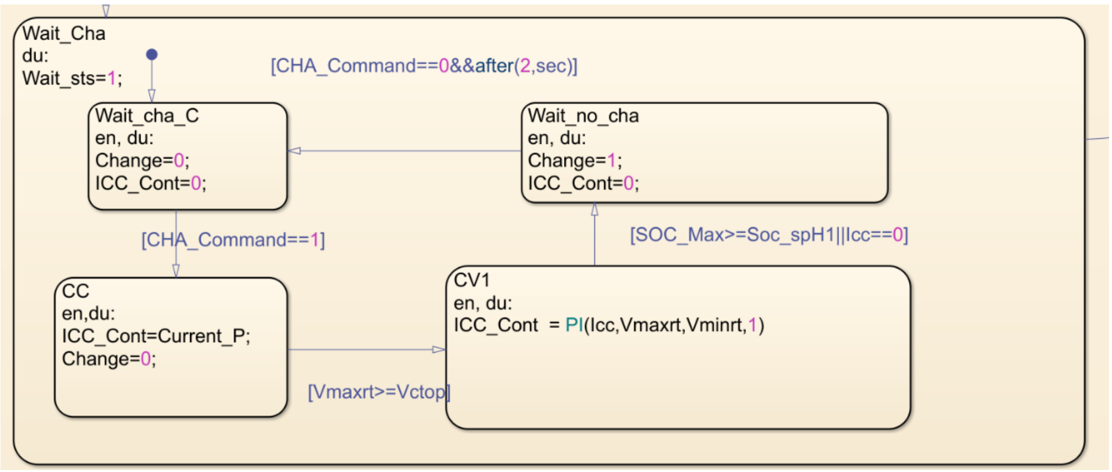


Figure 36. Control of the Pulsed Charge. Charge Area.

When the battery is fully charged, the charger enters on the *Wait_Cha* state. There it waits until the control of the balancing gives command to perform the charge. Then when the command is received a Current pulse is performed. The value is chosen on the *Current_P* parameter. This parameter is a design criteria.

As can be observed, in case that the maximum voltage on a single cell surpasses the maximum allowed, the charger enters on the CV part of the charge process. The current is then slowly decreased. If it reaches to a point in which the current is zero, the charge pulse has finished.

The balancing action starts if the battery is on *Wait_Cha* and the difference between cell voltages is above threshold. In case both conditions are fulfilled, the balancing action starts. The balancing action is performed for 5 minutes (design criteria). After that time, the pulse is started. When the charger has finished the pulse, it gives feedback to the balancing algorithm, and it decides to resume balance or not.

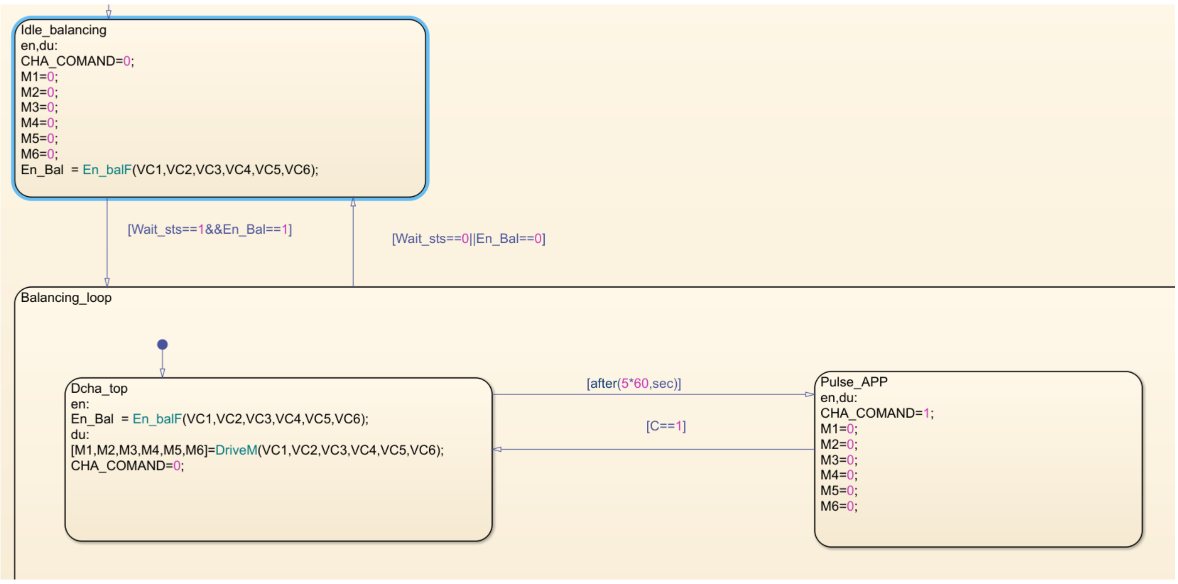


Figure 37. Pulsed charge control. Balancing Control Area.

4.2.2.2 Parameters chosen for simulation

ΔVOC

- Balancing resistance: 43 Ω.
- Voltage threshold: 25 mV.

Pulsed charge

- Balancing resistance: 43 Ω.
- Pulse current: $I_p = 5\ A$. Is equal to C/2.
- Balancing action time: 5 minutes.
- Voltage threshold: 25 mV.

4.2.3 SOC History

SOC history method relies on SOC estimation for each of the cells to choose whether to perform balancing. It is based on the same HW seen in Figure 33.

In the simulation side it has been chosen to rely on the SOC calculation of the battery block on Simulink.

4.2.3.1 Implementation

This algorithm compares the SOC of the cells at a specific moment. This is intended as it may be interesting to wait some time to let the SOC estimation SW to perform multiple iterations to get an accurate value or to let cells relax. This moment is a design criterion that can be changed for simulations.

A total of 10 options have been implemented on the simulation (Table 6). These options are chosen on the input parameters of the simulation. On all methods the delay time in which the time calculation is performed can be configured as an entry parameter on the simulation. This delay time is applied after the charge or discharge process is finished. After this delay, the calculations are performed, and the balancing action is started if necessary.

Table 6. Different Options for SOC history algorithm.

Option	*Behaviour of the SOC history algorithm*
0	No operation. The system never enables balancing action nor calculates the values for operation.
1	On charge start. On the start of every charge process the calculations to get the time to balance are performed. The balancing is then performed the defined time no matter the operational behaviour of the cells.
2	On discharge start. On the start of every discharge process the calculations to get the time to balance are performed. The balancing is then performed the defined time no matter the operational behaviour of the cells.
3	Wait on upper charge. When the cell has finished the charge process and there's some relaxation time, the time to balance is then established from each of the cells. The balancing is then performed the defined time no matter the operational behaviour of the cells.
4	Wait on final discharge. When the cell has finished the discharge process and there's some relaxation time, the time to balance is then established from each of the cells. The balancing is then performed the defined time no matter the operational behaviour of the cells.
5	Charge restricted. The time to balance is calculated every time that a charge process is started. The balancing action is only performed under charge process.
6	Discharge restricted. The time to balance is calculated every time that a discharge process is started. The balancing action is only performed under discharge process.
7	Wait on upper charge restricted. When the cell has finished the charge process and there's some relaxation time, the time to balance is then established from each of the cells. The balancing is then performed only when the cell has finished the charge process.
8	Wait on final discharge restricted. When the cell has finished the discharge process and there's some relaxation time, the time to balance is then established from each of the cells. The balancing is then performed only when the cell has finished the discharge process.
9	Always. Time calculation and evaluation are performed when there are no times to balance already calculated.

When the Balancing system compares the SOC of the cells, it establishes for each of the cells the ΔSOC in respect of the cell that has less SOC. With this ΔSOC a time to balance is established. This is the amount of time that the balancing action must be performed. The calculation formulas are seen in (42), (43), (44) and (45).

Out of all the algorithms implemented, this is the one that would require more non-volatile memory on a real environment. All the times to balance for each of the cells would be needed to be established and saved.

4.2.3.2 Parameters chosen for simulation

For the simulation the following parameters have been chosen:

- Balancing resistance: $R_{bal} = 43\ \Omega$
- SOC threshold: 0.5%
- Only method 9 has been used.

4.2.4 SSC

Single Switch capacitor technique is the active balancing method implemented in this thesis. As seen in the section 3.3.2.1, it relies on the charge and discharge of a capacitor to draw energy from a highly charged cell to a one with less charge. In this section the implementation on the model will be discussed together with certain topics and problematics associated with the methodology.

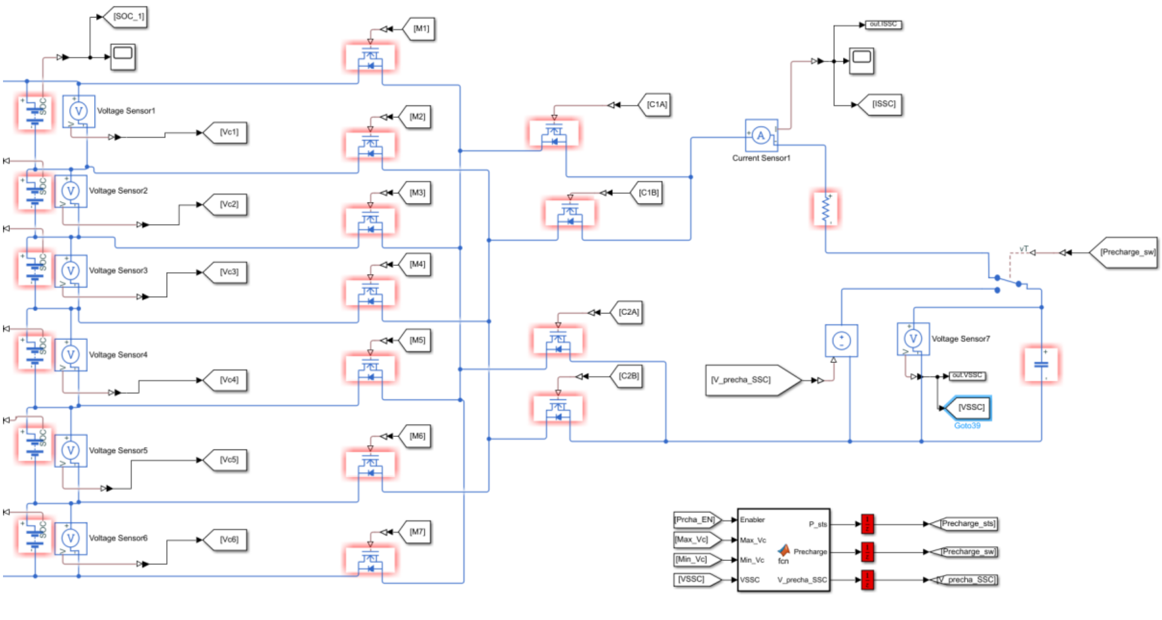


Figure 38. SSC Model implemented (control excluded).

Figure 38 shows the implementation of the balancing method. On the right side the capacitor can be seen. Next to it, there's a precharge circuit that will be commented on the next point. On the middle section of the model there are two sets of two switches. C1A and C2A are switches that permit the connection of the capacitor with the positive terminal of the cells that are even. C1B and C2B permit the connection of the capacitor with the positive terminal of the cells that are odd. It can be said that there's a *bus* of the even cells and a bus on the odd cells.

Then, on the left side of these *buses*, there are the different mosfets that give access to the *bus* (M1, M2…M7). With these switches and together with a strong logic implementation, all the needed combinations can be performed. Each cell can be charged and discharged to the capacitor.

On the previous sections the design criteria values were stated on the last part of the section. However, in this case the selection of the different elements is made in the technical basis part as its impact is higher on this method. On the last part of the section, the design criteria values are summarized.

4.2.4.1 Technical basis

Capacitor Inrush current

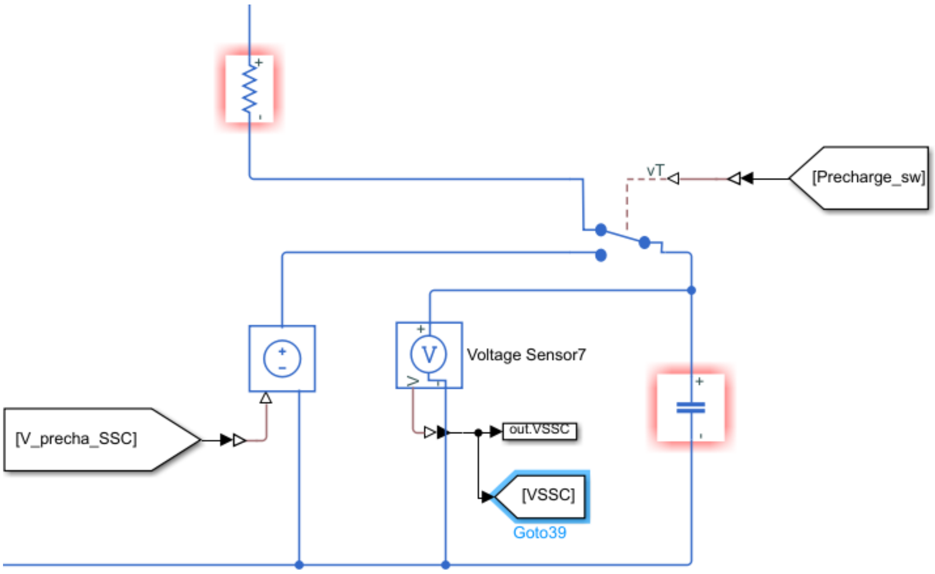


Figure 39. Detail of the precharge circuit, Capacitor, and series resistance from Figure 38.

In Table 2 it was stated that, theoretically, the SSC method requires $n_c + 5$ switches. However, on Figure 38, the proposed circuit on the model has a 6th switch with extra control that can be seen in detail in Figure 39. This is the precharge circuit. This is intentionally made to avoid the inrush current on the wires caused by the voltage difference between the highly charged cell and the capacitor at operation start.

If there would be the case of the capacitor being at 0 V and a cell being at 4 V, the equivalent circuit would be the one seen on Figure 40.

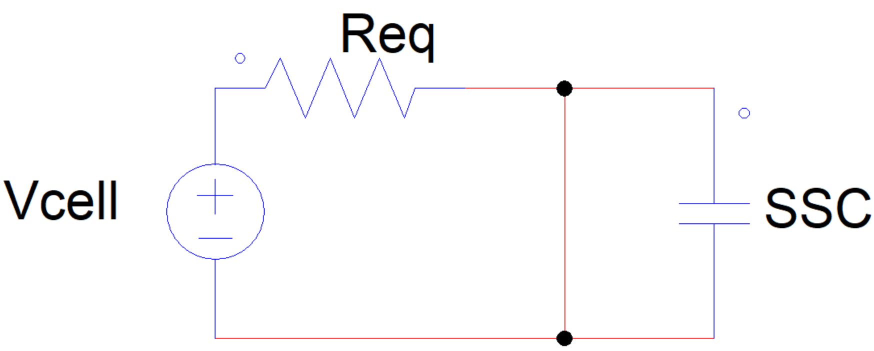


Figure 40. Equivalent circuit on t=0 s and Vssc=0 V.

As can be seen, the capacitor could be considered as a short circuit. The equivalent resistance of the circuit would dictate the inrush current. As the cell series resistance is in the order of mΩ, the inrush current would be large if additional measures are not taken. If this would happen on a real module, it could burn the cables used to connect the cells to the BMS.

The additional circuit of the 6th switch charges the capacitor to a voltage value that is between the cell that needs to be discharged and the cell that needs to be charged.

This event on the inrush current not only occurs on the start of the charge of the process, it also occurs in a lower significance every time switching occurs. In Figure 23 the operation of the SSC can be observed. There are spikes on the current, these spikes are directly caused by the difference on voltage between the Condenser Voltage and Cell's voltage.

Example: We have the upper cell at a 4.2 V value, and it must discharge energy to a 4.1 V one. Considering that the capacitor has the sufficient time to fully charge and discharge (T_{sw} >4τ), $I_c(0)$ would be:

$$I_C(0) = \frac{V_{cellh} - V_{SSC}}{R_{eqc}} = \frac{4.2 - 4.1}{R_{eq}} \rightarrow \frac{0.1\,V}{10\,m\Omega} = 10\,A \tag{61}$$

If R_{eq} is in the order $10\,m\Omega$, which could be the sum of the equivalent resistance of the cell plus the wire and R_{ds} resistance, we would have and instant current that would be of 10 A. This as commented before cannot happen, a resistor must be added to the circuit.

This imposes a design criteria value. A maximum current of a 1 A for a 0.1 V difference is considered. So, this resistor has a fixed value of **0.1 Ω.**

Furthermore, it also imposes a restriction to this method, the voltage difference between the high cell to capacitor has a maximum value of 0.1 V. The same happens with the low cell to capacitor. This restricts this method to be used in high voltage differences scenarios. This is expected to do not affect the performance of the balancing system. High voltage differences should not occur under normal conditions and the system would depart from a fully balanced battery module.

The addition of this resistor reduces proportionality the energy transfer as will be seen on the following section. Also, it adds losses.

Performance and Operational point

Departing from the considerations of the previous section and from the equations of the section 3.3.2.1 a series of plots are made to evaluate the performance of this balancing method and to choose the values of the parameters used on the final simulations.

$$E(Wh) = f * C * V_{diff} * \left\{ \left[\frac{V_{diff}}{2} * e^{-\frac{2D}{\tau f}} - V_f * e^{-\frac{D}{\tau f}} - \frac{V_{diff}}{2} + V_f \right] \right\} \tag{62}$$

The different plots will be based on equation (62) changing one parameter value while keeping the others at a selected value to see the dependencies on each case. The values selected for all the cases when the variable is not tested are the following:

```
Req=0.1;   (Design value of the Previous point)
C=0.1;     (τRC is 10 ms, -> frc = 100 Hz)
D=0.5;
Vf=4;
Vdiff=0.1;
f=10;
```

Several observations can be made from the equation and its graphical values:

- Regarding Capacitance (Figure 41): It starts to have an asymptotic behaviour after 3 F.
- Regarding duty cycle (Figure 42): Value D=0.5 maximizes the energy transfer. Value of D higher than 0.5 are not possible. Equation (62) gives the value of the

energy transfer from the cell to the capacitance. Then on the 1-D time the energy follows the same formula, but the energy transfer is from the capacitor to the cell.

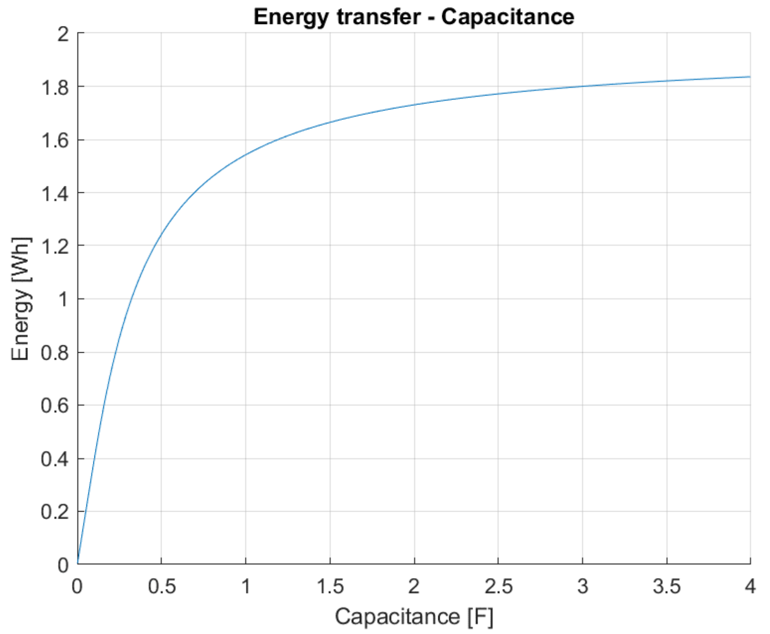


Figure 41. Energy transfer in respect on the capacitance.

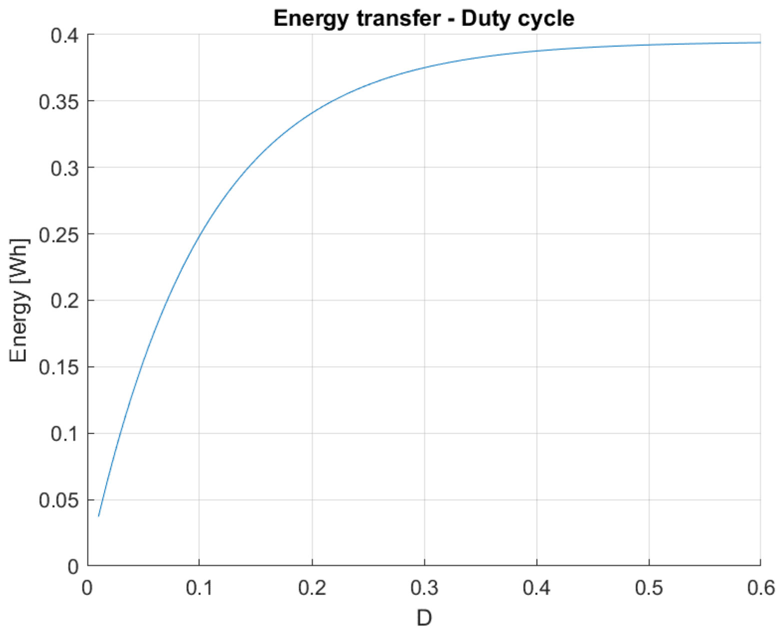


Figure 42. Energy transfer in respect on the Duty Cycle.

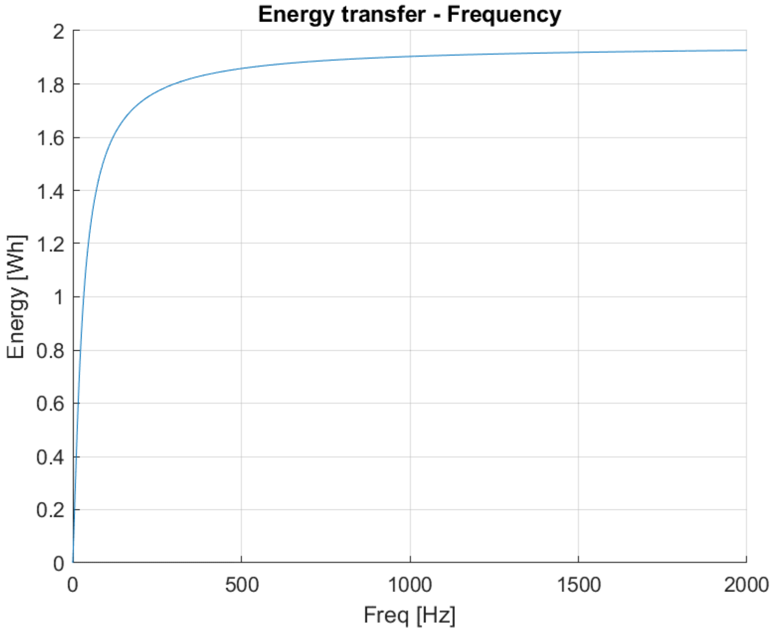


Figure 43. Energy transfer in respect on the switching frequency.

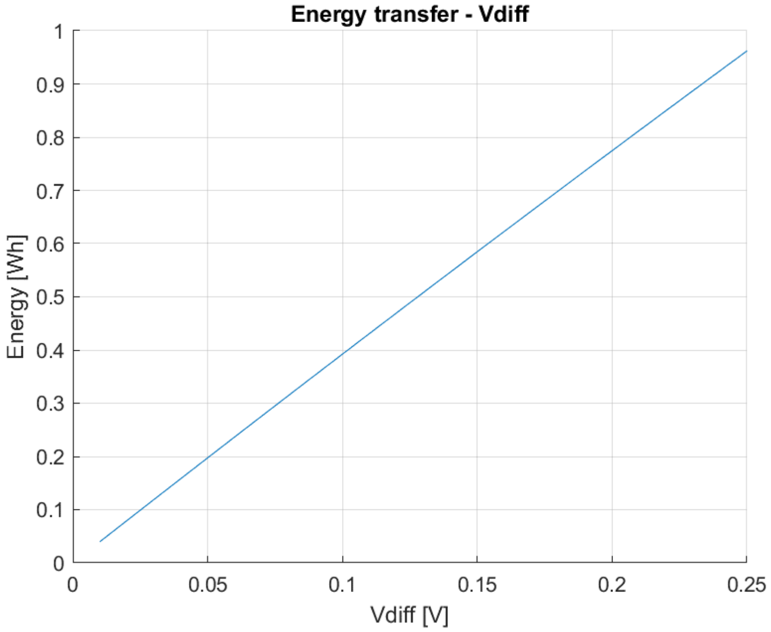


Figure 44. Energy transfer in respect on the Voltage difference.

- Regarding frequency (Figure 43): It gets an asymptotic behaviour after approximately 500 Hz.
- Regarding Voltage Difference (Figure 44): It gets an asymptotic behaviour after approximately 1 kHz.
- Regarding Req (Figure 45): It has a flat part on the lower value resistances. Then after 0.1 Ω (approximately) there is a drop.

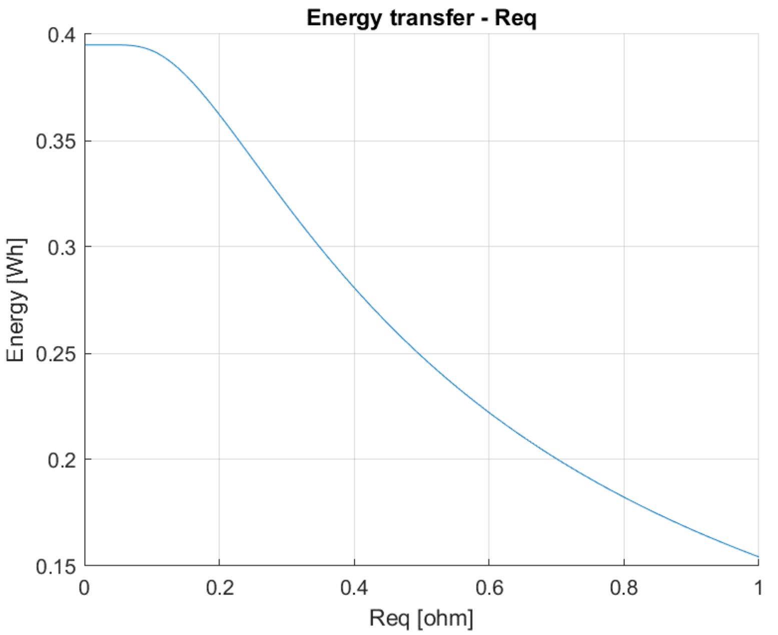


Figure 45. Energy transfer in regard of the Req.

The energy transfer is closely related with the time constant τ, the frequency of operation and duty cycle. Ideally, we would like to charge/discharge the condenser at 1 A rate, as it is the design value that we have considered as a maximum. The energy is the integral of power, so energy is the area that is formed by the multiplication of the capacitor voltage and the capacitor current.

However, capacitor current is the derivative of the voltage. If the voltage stabilizes then the current is reduced. When the capacitor voltage is close to the final value, the current is reduced. For the values used for Figure 43:

$$\tau_{RC} = C * R_{eq} = 0.1 * 0.1 = 0.01\, s; f_{sw} = 100\, Hz \quad (63)$$

For lower values of frequency than 100 Hz, there's a decay on the energy transfer value. However, for values higher than 100 Hz the curve starts to flatten.

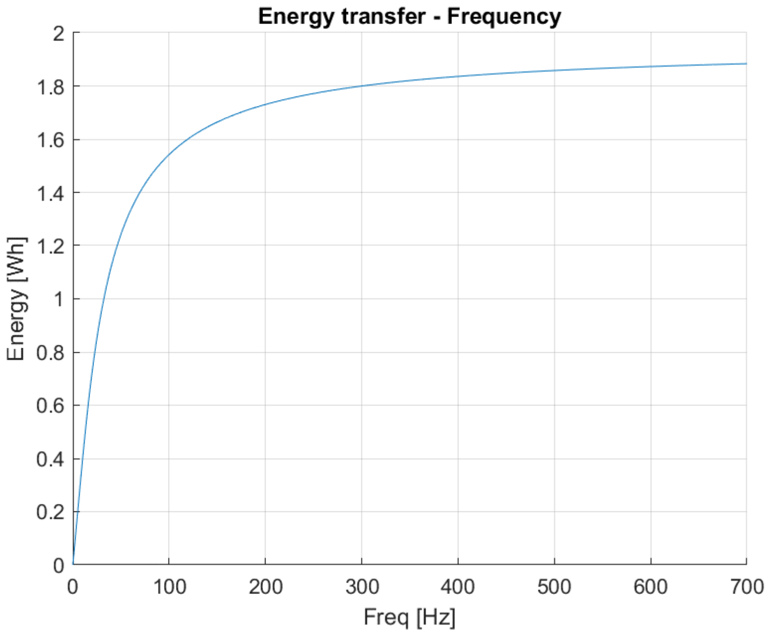


Figure 46. "Zoom" on lower frequencies of Figure 43.

If we look at the higher frequency values, we can see that there's an asymptotic behaviour after 500 Hz. From a design perspective the frequency could be fixed with (64) relation.

$$f_{sw} = \frac{k}{\tau_{RC}} \rightarrow k \geq 5 \quad (64)$$

The individual plots have provided some information regarding the behaviour of the circuit. To provide a better understanding, a series of 3D plots have been performed to compare the energy transfer with different combinations of inputs.

3D plots have been performed checking the dependence of each parameter in function of the frequency. This was performed in this way to check if it would be necessary to implement a variable switching frequency.

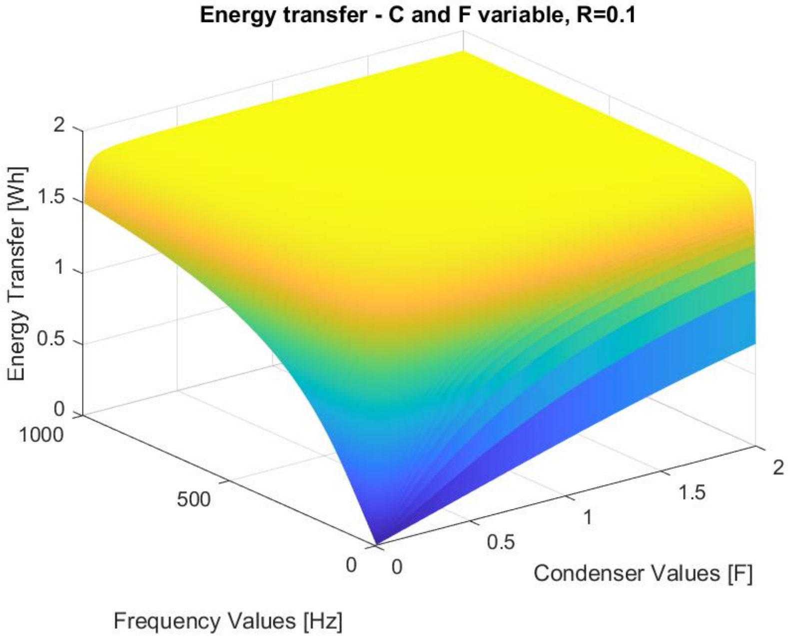


Figure 47. 3-D Graph, Energy transfer result of Condenser value and frequency matrix.

Figure 47 and Figure 48 have been done with the following parameters:

Figure 47:

```
D=0.5;
Req=0.1045;
Vdiff=0.1;
Vf=4;
[f,C]=meshgrid(1:1:1000,0.01:0.001:2);
```

Figure 48:

```
D=0.5;
Req=0.1045;
Vf=4;
C=0.5;
[f,Vdiff]=meshgrid(1:1:1000,0.01:0.01:0.3);
```

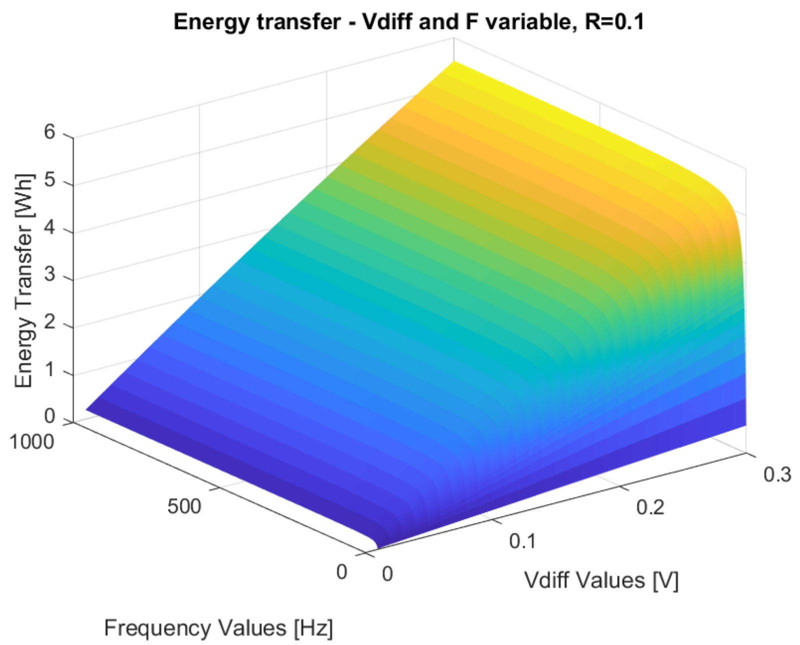


Figure 48. 3-D Graph, Energy transfer result of Vdiff and frequency matrix.

On Figure 47, is observed that despite increasing the frequency value and the capacitor value the energy transfer gets locked on a stable value. As long as the circuit is operating in a period smaller than $T_{sw} = \frac{\tau_{RC}}{k}; k \geq 5$, the energy transfer is kept at maximum value. Furthermore, there is no needed to enlarge the capacitance value of the capacitor in values higher than 0.5 F with a correct frequency selection.

Figure 48 shows that increasing the switching frequency for a given voltage difference does not imply an increase on energy transfer. A variable switching frequency is then discarded, as it does not apport more energy transfer. The difference between the energy transfer in lower frequencies in respect of the high frequencies is due to τ_{RC}.

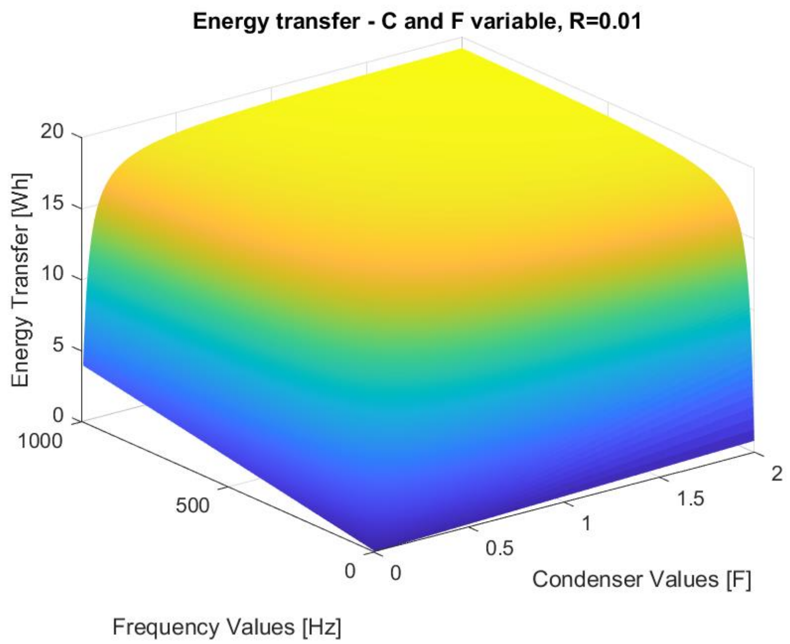


Figure 49. 3-D Graph, Energy transfer result of C and f matrix.

Figure 49 shows the results of reducing the resistance to 0.01 Ω. The resistance has been divided by a factor of 10 and the energy transfer has increased in a 10 factor in respect of Figure 47. This shows that the selection of the value of the resistance has a proportional impact on the energy transfer as the current values in the capacitor are higher.

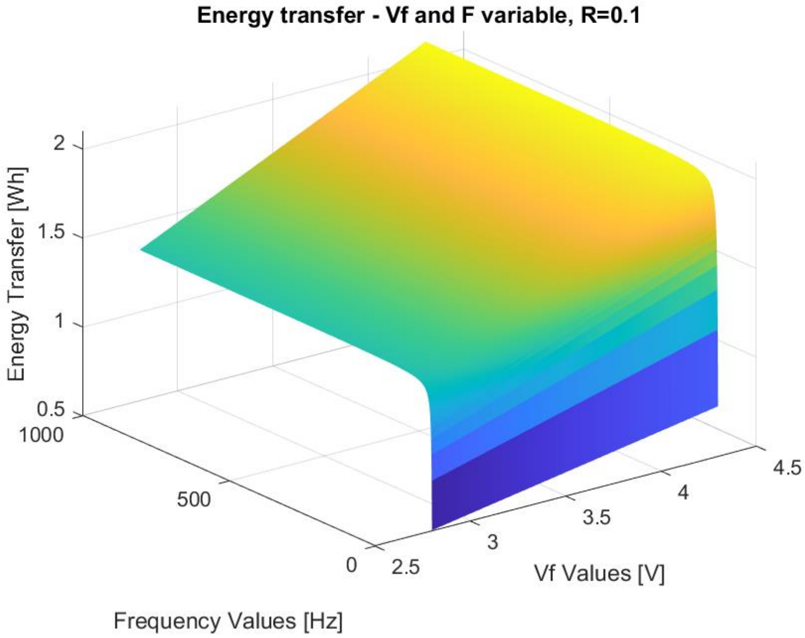


Figure 50. 3-D Graph, Energy transfer result of C and f matrix.

Figure 50 has been done with the following parameters:

```
D=0.5;
Req=0.1;
Vdiff=0.1;
C=2;
[f,Vf]=meshgrid(1:1:1000,2.8:0.001:4.3);
```

On Figure 50 it can be observed that a variable frequency would not lead to an increase on energy transfer.

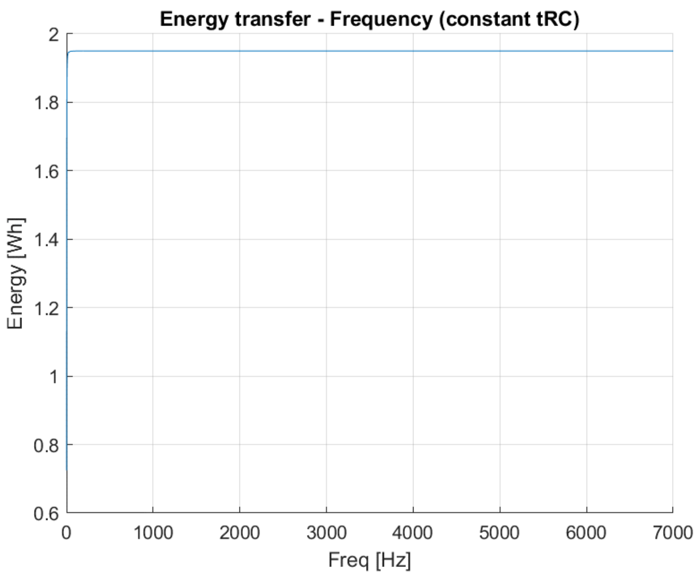


Figure 51. Energy transfer-frequency with a fsw > 5 frc.

After stating that the maximum energy transfer would happen when $f_{sw} \geq 5 * f_{RC}$ it was needed to check what would happen with a variable frequency but maintaining the relationship between f_{sw} and f_{RC}. The result is seen in Figure 51 and Figure 52 and there's not an increase on the energy transfer value.

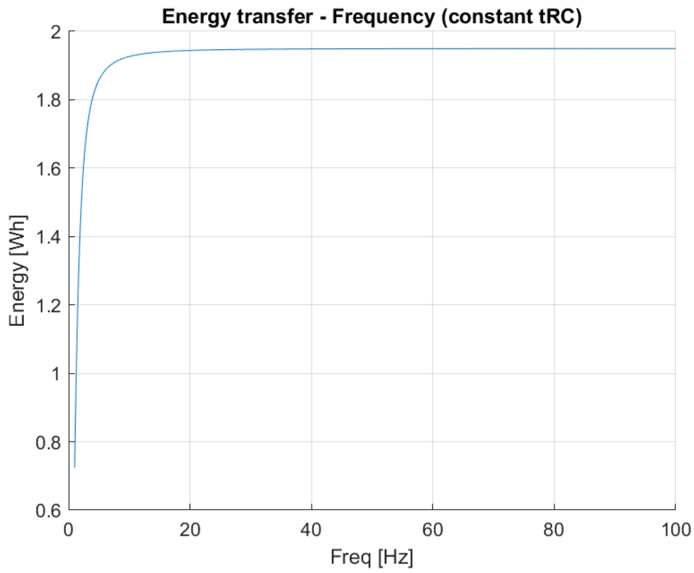


Figure 52. Energy transfer-frequency with a fsw >5 fRC. Detail on lower frequencies.

Simulation - Detail of the control

To provide a little bit of insight on the control side, in Figure 53 part of the control required for the SSC algorithm is showed. This control is not from a pure SSC implementation as it is for the multi module mode (4.2.5). However, it has the regular SSC logic implementation.

Firstly, it can be seen the precharge logic. Precharge is only needed if the voltage of the capacitor has a value that is not in between the highest cell and the lowest one with a certain boundary. When the precharge has finished, or in case that is not necessary, the SSC enters into operation.

First, the cell that must be charged and the cell that has to be discharged are selected. Then the capacitor is connected to the highly charged cell. After the defined time (DT) it connects to the low charge cell. After 10 iterations of this process, a revaluation of the status is performed. In case is needed, the cells that are interacting change.

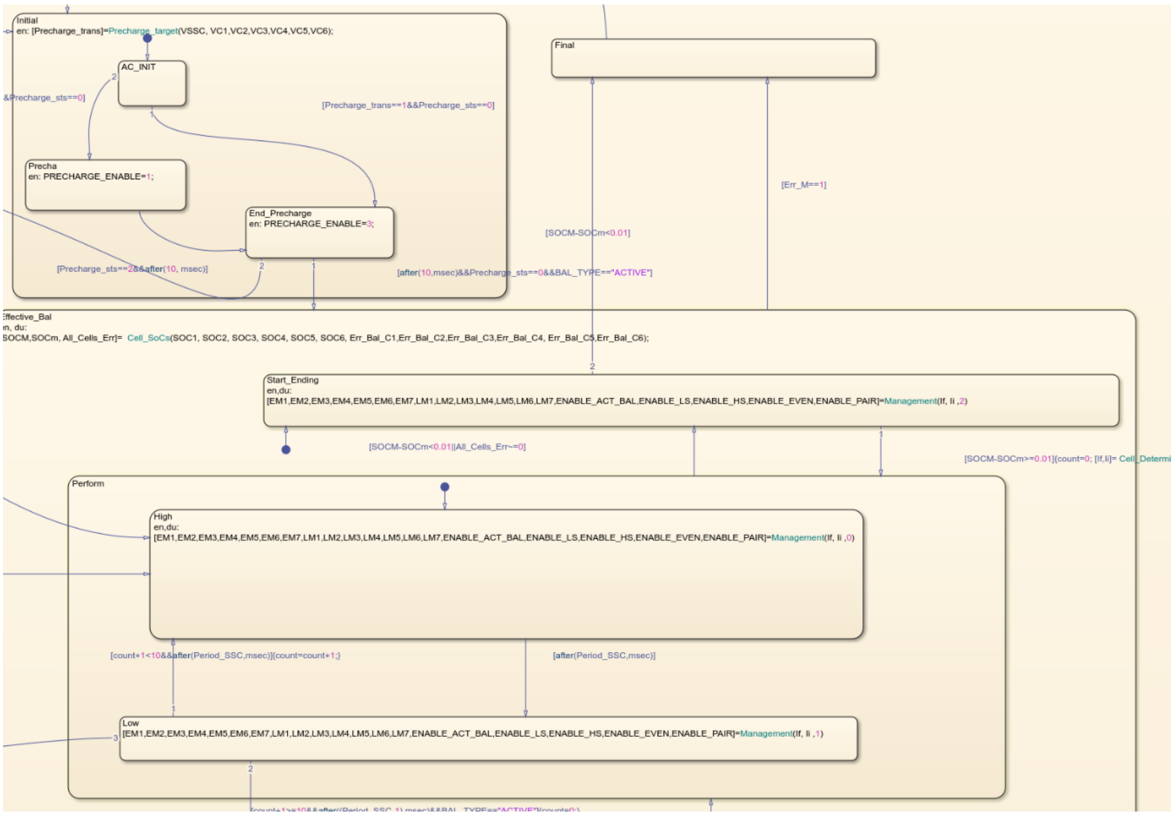


Figure 53. Detail of the Control part of SSC.

4.2.4.2 Parameters chosen for simulation

In this section the set of parameters that are chosen for simulation of SSC capacitor technique are shown. The SSC balancing system will have a 1 A restriction on the maximum allowed current on the wires between the cells and the balancing system. This maximum current value would apply for a 0.1 V difference between the cell and the capacitor.

The system will work in an $f_{sw} \geq 5 * f_{RC}$. The condenser value will be high enough to do not impose a high frequency system. With high frequency switching the switching losses would increase as the operation is hard switching.

Furthermore, at lower frequencies, the tasks that would implement the control on a real environment would not need to be triggered in a high cyclicity.

$$R = 0.1\,\Omega;\; C = 0.5\,F \rightarrow \tau_{RC} = 0.05s \rightarrow f_{RSSC} = 20\,Hz$$

$$T_{sw=}\,0.01 \rightarrow f_{sw} = 100\,Hz$$

$$D = 0.5$$

On the selected values the f_{sw} is five times above the f_{RC}.

4.2.5 Multi Module Control Approach SSC and Advanced SSC

This section is a little bit off the central topic of the master thesis. However, it is intended to provide a further look on the SSC implementation thinking of a real application.

The algorithms that have been implemented, are implemented taking in mind a module of a battery with 6 cells. The scalability of all the different models to a full battery pack is simple in all methods, except on SSC case.

On ΔVOC, Final voltage and SOC History algorithms, the impact of having more modules is low. The only change that there would be is that the minimum VOC, or SOC variable that is needed to perform the different calculations is the minimum out of **all** the battery pack. The balancing action of the different battery modules would be performed with the intention of getting all the cells to the minimum charge one.

In the SSC however, if there’s a difference on the mean value of the cells SOC of the different modules it cannot be compensated only with the SSC technique.

Table 7. Hypothetical behaviour of the SOC of the cells before and after balancing action.

Module 1	SOC Before	SOC After	Module 2	SOC Before	SOC After
Cell 1	90%	80%	Cell 1	70%	65%
Cell 2	80%	80%	Cell 2	65%	65%
Cell 3	80%	80%	Cell 3	65%	65%
Cell 4	80%	80%	Cell 4	65%	65%
Cell 5	80%	80%	Cell 5	65%	65%
Cell 6	70%	80%	Cell 6	60%	65%
Mean M1	80%	~80%	Mean M2	65%	~65%
Mean SOC of the Pack		72.5 %			

Table 7 shows what would happen after some SSC balancing action on a battery pack with two modules. These two modules would be placed in series to create a battery pack. Between modules, as in a vehicle application, there would be only the power connection and the communication between the slaves of the BMS.

The first module has a mean value of 80% SOC, with a maximum difference of 10% between the high charged cell and the lowest charged cell. On the other hand, the second one has a mean value of 65%. Considering that the battery pack has been performing SSC balancing with no operation, after some time the balancing action would have finished. After this balancing action, the mean value of the SOC of the two modules would have almost the same value (losses in the process ignored). In the end both modules would be independently well balanced, but the overall battery pack wouldn’t be in balance.

That’s why a multi module strategy is needed to be implemented. This strategy will permit to balance the modules independently and then achieve balance globally. The way to achieve balance globally is by using dissipative balancing when needed.

HW side

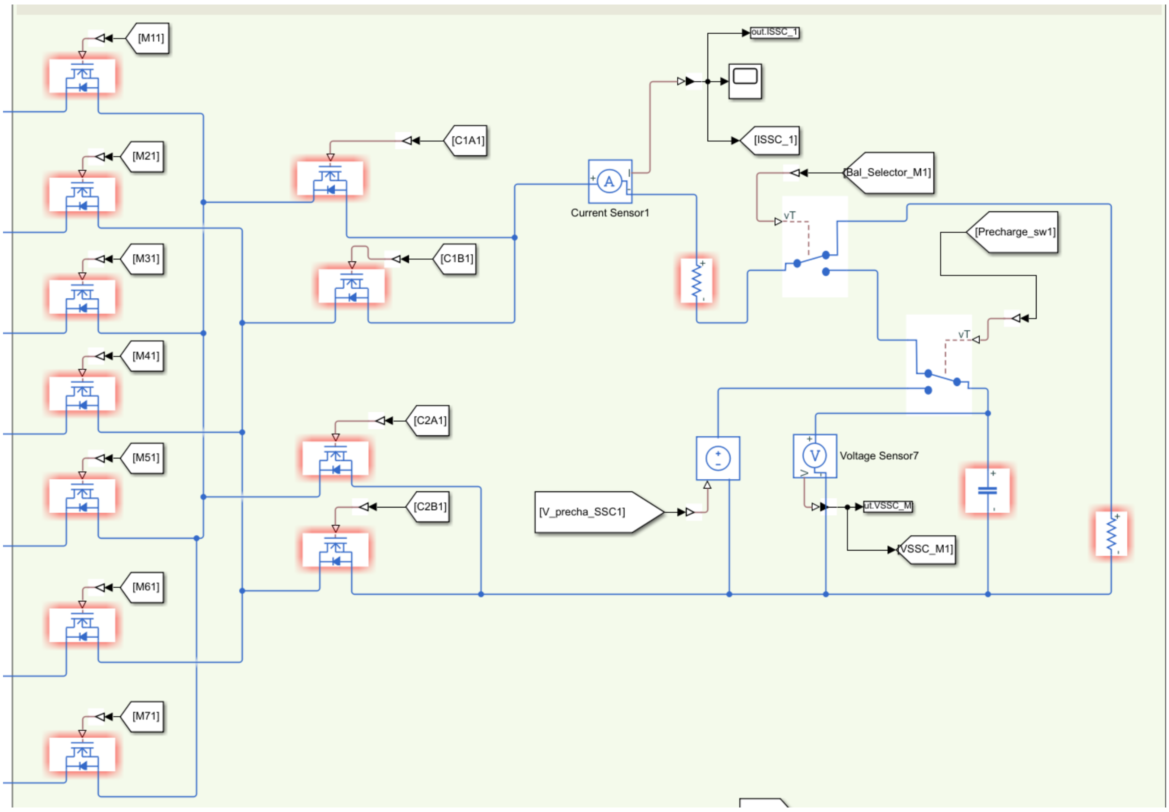


Figure 54. Update on the HW of the SSC.

To be able to perform some dissipative action one resistor has been added to the HW layout, as well as it's correspondent switching network. This balancing resistor will be shared with the different cells that need to lose energy.

Control

The control side of this implementation is complex. Is it implemented using *State Flow* and it has 4 main areas.

The first one is the dispatcher. The dispatcher chooses what action the system shall do in function of the input variables from the module. This section is shown in Figure 55.

The *Initial* state is the default state. If the SOC of the module is **lower** than the average SOC of the battery pack and there are differences on cells values, the module will enter on regular active balance operation. This area has the same functional implementation as SSC.

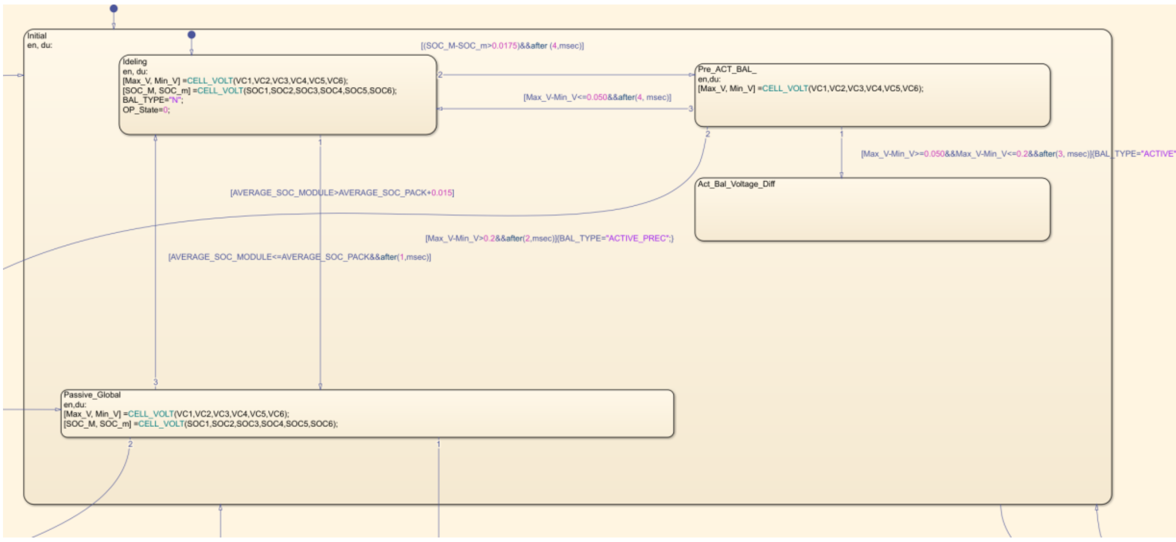


Figure 55. "Dispatcher" of the multi–Module SSC control.

In case that the SOC of the module is **above** the average SOC of the pack, the control state *Passive_Global* will be entered. In this case, there are two paths of execution on the control. The battery module can enter into active balance or perform a regular dissipative balance. The execution path will try to balance actively in case the following condition is fulfilled.

- There's one cell on the module that has a SOC below the average value of the lower charge module (target SOC).
- There's one cell on the module that has a SOC above the mean value of the lowest charged module. This condition as per scenario characteristics is always fulfilled.
- The difference on voltage between the cells is below the operational limit of 0.2 V.

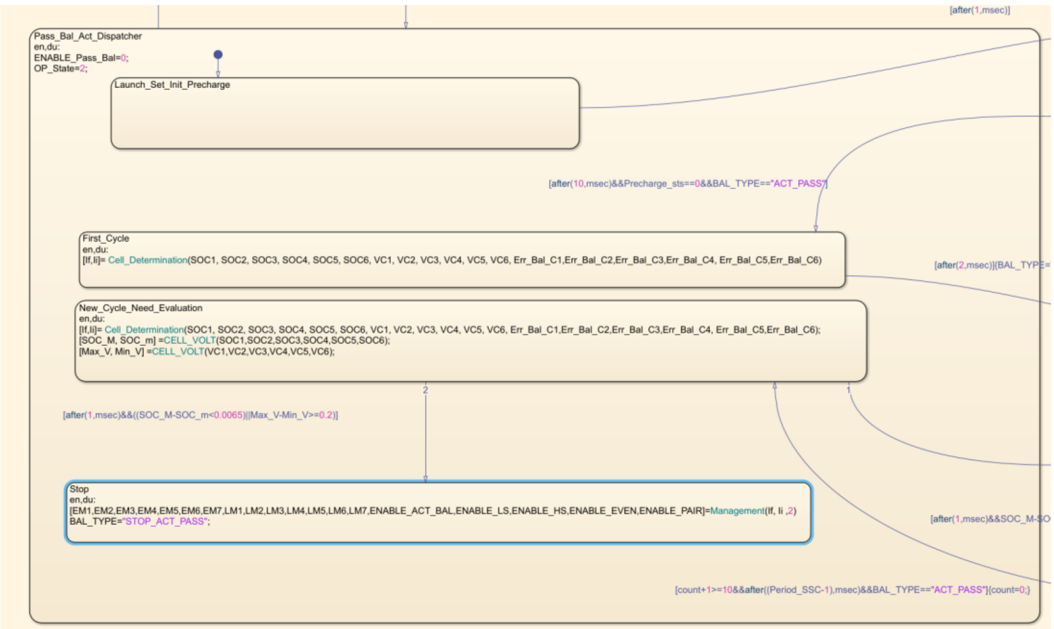


Figure 56. Passive-Active area.

Using active balance is needed for the scenario in which the module that needs to be discharged has a cell that is below the mean value of the SOC of the less charged module. In this case, that cell must be charged until its SOC is equal to the mean value of the less charged module. Without this implementation all the battery pack would have to reach this SOC. This scenario lead to have to draw a lot of energy from the other cells.

In the case that there's a noticeable difference on the SOC there may be a difference on the voltage of the cells. SSC active balance technique can transfer more energy in higher voltage differences. Is better to perform active balance operation and bring up the SOC first, and then balance the rest of the module to the desired final SOC.

The control area that implements this feature is known as Passive-Active area. This control section performs a selective call to the states that are present on the active balance area (regular SSC implementation).

Passive-Active area is finished after the cell that was below the mean value of the SOC of the less charged module, has a value close to it. After this, the remaining cells of the module can be balanced using the dissipative resistor.

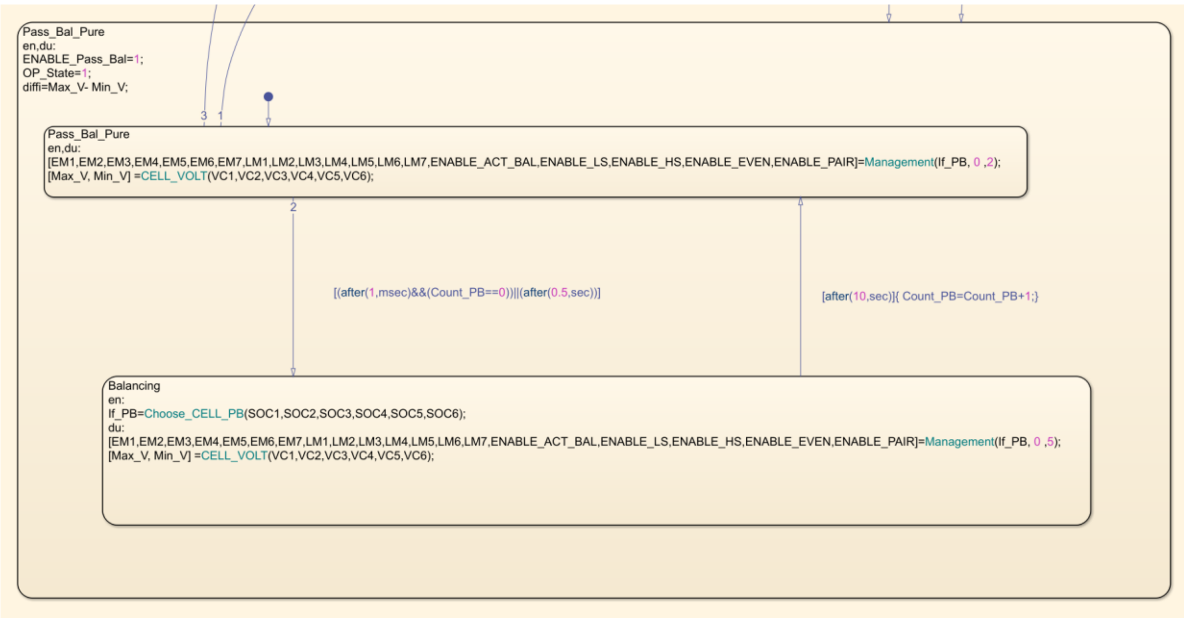


Figure 57. Pure Passive Balancing Approach.

The pure passive balancing chart is basically a selector. The cells that have a higher SOC than the average of the less charged module are discharged one at a time by the resistor.

HW wise, the system could be capable of balancing all the cells that are required at the same time. However, there are a series of drawbacks to this proposal:

- The number of cells that are needed to be balanced at the same time can vary from one to the number of cells on the module.
- There could be the scenario in which the cells would not be connected one right behind the other. So different options would have to be considered. This would increase the control logic. Furthermore, it would also be needed to balance blocks of cells at a time.
- The resistor value chosen must be selected with the maximum current allowed to be drawn from the cell tracks. This value would be chosen with the worst-case scenario. The worst-case scenario would be in the case that there's only one cell that needs to be balanced.

These drawbacks, and the fact that trying to balance all the cells simultaneously would not add any increase in energy dissipation justify the decision to balance the cells one at a time.

Control: Advanced SSC

In this multi module control approach, together with the capabilities of Stateflow, some additional control features were implemented.

The first feature on the control is applied in the case that the ΔV value of the cells is over the specification of the SSC balancing. In case that the voltage is above the specified value, the SSC balancing action cannot happen. In case that the voltage is outside the specification value, the current peaks can be over the current limit. (61) show the relation between the voltage difference and the inrush current.

In the case that the voltage difference is over the specification, first the high charged cell will be discharged until the difference between cells is lower than 0.2 V. Then regular SSC action will be performed.

The second feature added to the control is to check the SSC behaviour in front of errors. In case that there's an error on one of the cells, this is one discarded for the SSC balancing action.

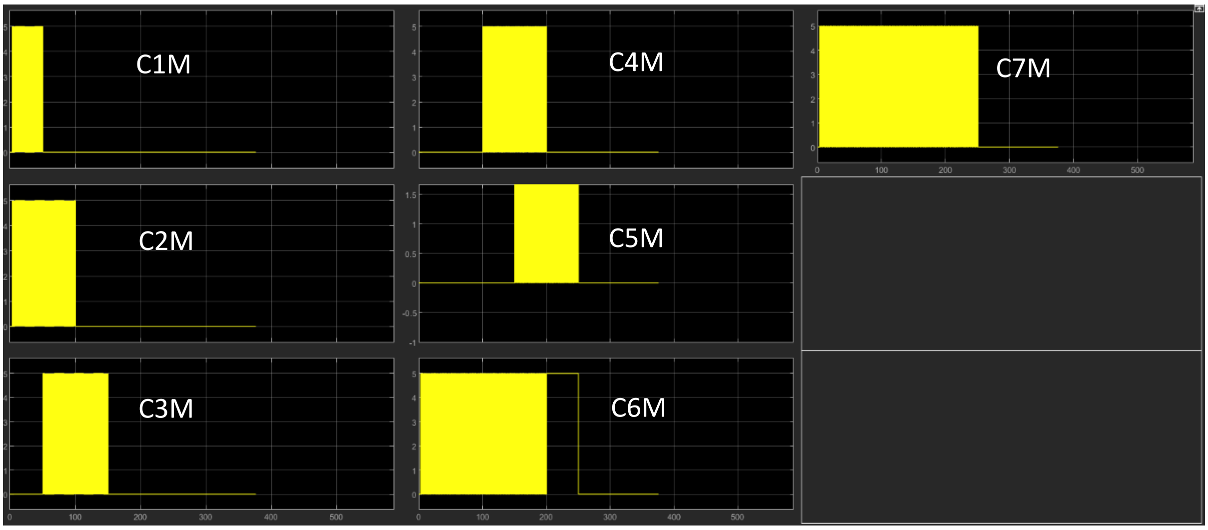


Figure 58. Behaviour of the Switches for SSC with error introduction.

On Figure 58, the cells are discarded progressively for balancing when the error is triggered. For example, an error is introduced on the first cell at 50s. This cell was used for SSC and C1M and C2M were working to charge the capacitor. Cell 6 is intentionally used to discharge the capacitor (C6M and C7M working). After the error introduction, the balancing action is performed with cell 2 (C2M and C3M engaged). When the system gets to 250 s an error is introduced on cell 5. With no remaining cells available, the balancing action is stopped.

Table 8. Initial SOC conditions table. SSC advanced control.

	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6
SOC	78 %	76 %	74 %	72 %	70 %	68 %

Table 9. Error introduction sequence and timing.

	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6
Error Introduction	50 s	100 s	150 s	200 s	250 s	300 s

This last part is not implemented to its full potential due to timing. However, it was wanted to explore the possibility to incorporate an error tolerance feature into the control. The result, together with the ease of implementation, remark the fact that State Flow add-on is an indispensable tool for this kind of SW development.

In Figure 59 a simple state flow chart is made for error generation on a single cell. This chart, from the input parameters, determines whether if the cell is on fault due to balancing, due to overtemperature or isolation fault. In a real application this chart would be applied to each of the cells.

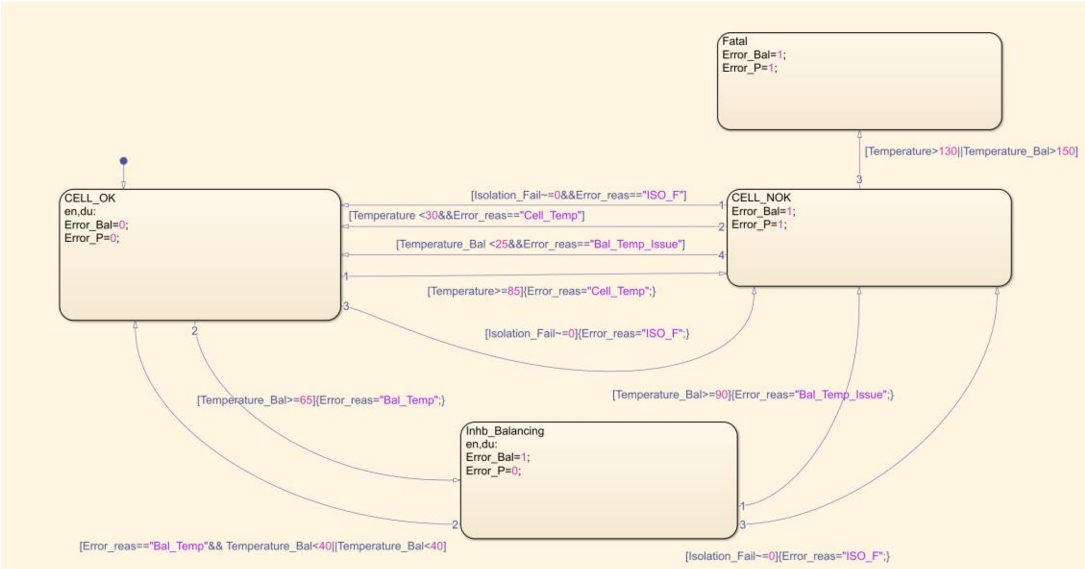


Figure 59. Basic error determination chart for one cell.

4.3 Test Scenarios

To compare the different balancing algorithms a series of tests are performed. In the following points the characteristics and objectives for each test are described.

Tests will be performed using cycle operation and introducing current profiles to the models. For the profile tests, trip A01 is selected.

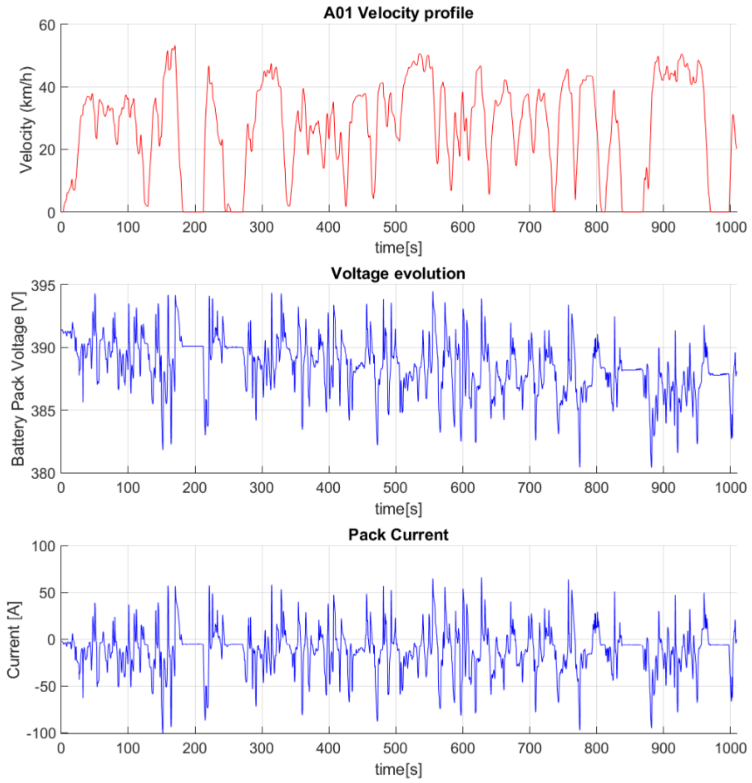


Figure 60. A01 Profile values.

Figure 60 show the most important curves of the A01 profile. The vehicle's current, speed and voltage are measured. The plot containing current is the one that will be introduced as an entry parameter on the models for simulations.

Functional tests

These tests have the intention to prove that the algorithms are functional and act properly in different scenarios. Following is the description of the four scenarios in which the models are tested.

2 Cycle test 5% ΔSOC unique cell

Algorithms are tested performing a 2 charge-discharge cycle at 10 A. All cells will start at 0 % SOC except of one that will start at 5% SOC. Between charging cycles there will be a wait time of 1000 s. The cycles will be performed between 100% SOC and 0 % SOC (100% DOD).

2 Cycle test ΔSOC multiple cells

Algorithms are tested performing a 2 charge-discharge cycle at 10 A. Cell's SOC at start will be: 0% 3% 4% 5% 6% 1% (SOC per cell in order, expressed in 0% - 100% range). Between charging cycles there will be a wait time of 1000 s. The cycles will be performed between 100% SOC and 0% SOC (100% DOD).

1 Profile Cycle test 10% ΔSOC unique cell

ΔVOC, Soc History and SSC are tested with one single A01 profile. The Final Voltage algorithms are ignored as they would not enter operation. All cells will start at 70% SOC except of one that will start at 80% SOC.

1 Profile Cycle test 15% ΔSOC multiple cells

ΔVOC, SOC History and SSC are tested with one single A01 profile. The Final Voltage algorithms are ignored as they would not enter operation. Cell's SOC at start will be: 65%, 65%, 60%, 55%, 55%, 70% (SOC expressed in 0% - 100% range).

Table 10. Resume of Functional tests.

Functional tests		
Number of tests per model:	2 per algorithm	2 per algorithm
Model ↓	Cycle Testing: 2 cycles	Profile test: 1 trip
ΔVOC	Yes	Yes
FV: ΔVOC	Yes	No
FV: Pulsed Cha	Yes	No
SOC H	Yes	Yes
SSC	Yes	Yes
Advanced SSC 2M	No	No

Aging tests

These tests have the intention to prove the behaviour of the algorithms in case that there is a battery with aging. The aging phenomena will be introduced into one cell and will be introduced on different variables of the cell.

Only cycle tests are performed. The initial conditions will depart from 0% SOC (on most cases). The cycles will be performed between 25% and 100 % SOC. No relaxation time will happen between tests. The current used for charge and discharge will be 15 A.

Rs fade

A degeneration of the internal resistance on a single cell will be introduced as an initial condition. The degeneration will be of a ±50 % in respect of the values of the other cells. The cell that is affected with the aging phenomena will depart from a +5 % SOC. As the interesting part is long term impact of this aging phenomena, 5 cycle tests will be performed.

Capacity fade

A degeneration on the capacity of the cell will be introduced. Two scenarios will be considered. In one, a single cell will have a 10% reduction in respect of the nominal capacity. In the other scenario, that single cell will have a 10% increase in respect of the nominal capacity. This will make the cell achieve full charge and full discharge sooner or later than the other cells.

The tests will be performed with no initial SOC difference between cells. As the interesting part is long term impact of this aging phenomena, 5 cycle tests will be performed.

VOC-SOC aging

A degeneration on the VOC-SOC table is introduced for these tests. The tests will be performed with no initial SOC difference between cells. As the interesting part is long term impact of this aging phenomena, 5 cycle tests will be performed.

Original VOC-SOC table:

```
Vo_Table1= [2.8, 3.27, 3.9, 4.3];
SOC_Table1= [0, .25,.75, 1];
```

Degenerated VOC-SOC table:

```
Vo_Table1= [2.8, 3.27, 4.1, 4.3];
SOC_Table1= [0, .15,.75, 1];
```

Table 11. Resume of Aging tests.

Aging tests			
Number of tests	*2*	*2*	*1*
Model ↓	*Rs*	*Capacity fade*	*VOC-SOC*
ΔVOC	$\pm 50\%$, 5 cy	$\pm 10\%$, 5 cy	5 cy
FV: ΔVOC	$\pm 50\%$, 5 cy	Not tested	Not tested
FV: Pulsed Charge	Not tested	Not tested	Not tested
SOC H	$\pm 50\%$, 5 cy	$\pm 10\%$, 5 cy	5 cy
SSC	$\pm 50\%$, 5 cy	$\pm 10\%$, 5 cy	5 cy
Advanced SSC	No	No	No

Final voltage tests are discarded as no relaxation time will be applied between cycles in this section. On the report only the FV ΔVOC is presented with the aged resistance test. In that case the objective is to provide evidence to compare the different models with a case in which no balancing is applied.

Error tolerance

These tests have the intention to prove the behaviour of the algorithms in front of systematic errors on the measurements of certain variables of the cells. The initial conditions of the cells will be the same in all cases. In one of the cells, an error will be introduced in one of the variables. All the tests will be performed using the current profile and with initial condition of 90% SOC.

Cell Voltage

An offset error of $\pm 50\ mV$ on the voltage measurement of the cell will be introduced. As it is interesting to see the long-term behaviour of the test, 2 profile sets will be performed.

SOC Estimation

An offset error of $\pm$2% on the SOC estimation on one single cell will be introduced. As it is interesting to see the long-term behaviour of the test, 2 profile sets will be performed. The cells will depart from physically the same SOC.

Current

An offset error of $\pm 250\ mA$ on the current measurement of the cell will be introduced. 2 profile sets will be performed. The cells won't have an initial SOC difference.

Table 12. Resume of the Error introduction tests.

Error introduction tests			
Model ↓	V_{cell}	*SOC estimation*	*Current*
Total tests	2	2	2
ΔVOC	Yes	Yes	Yes
FV: ΔVOC	No	No	No
FV: Pulsed Charge	No	No	No
SOC History	Yes	Yes	Yes
SSC	Yes	Yes	Yes
Advanced SSC	No	No	No

Special case: Advanced SSC

Advanced SSC implementation has two modules connected with two control blocks. This implementation requires a lot of simulation resources. That's why in this case, the testing is different from the other models. The objective of the tests is to check the functionality at high level.

To fasten simulation only 1 cycle tests will be performed. To make the balancing action stronger in front of the cell's capacity, the cell capacity value will be reduced to 1 Ah. With this last measure, more scenarios can be covered in less simulation time.

Test 1:

Table 13. Multi Module SSC test 1. Initial SOC conditions.

	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Mean SOC
Module 1	7%	23%	23%	23%	23%	30%	21.5%
Module 2	15%	15%	20%	20%	16%	19%	17.5%

Module 1 will have a mean SOC value over the second one. So, it will have to lose energy. Module 2 is expected to enter pure SSC to balance its cells. Module 1 will have to first reduce the voltage difference between cells (<0.2V). Then enter SSC for passive balancing and finally pure passive balancing.

Test 2:

Table 14. Multi Module SSC Test 2. Initial SOC conditions.

	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Mean SOC
Module 1	15%	15%	20%	20%	16%	19%	17.5%
Module 2	7%	23%	23%	23%	23%	30%	21.5%

Test 2 is the same as test 1 but with the inverted SOC's. This proves if there's any error on the implementation or if there's any condition that creates a special "preference" into one of the modules.

Test 3:

Table 15. Multi Module SSC Test 3. Initial SOC conditions.

	Cell 1	Cell 2	Cell 3	Cell 4	Cell 5	Cell 6	Mean SOC
Module 1	10%	10%	10%	10%	10%	5%	9.16%
Module 2	7%	7%	7%	7%	7%	7%	7%

Test 3 is created after the execution of test 1 and 2. Its intention is to show SSC being applied before entering passive balancing on the first module. The first module must be discharged. However, it has a cell which SOC is below the mean value of the other module. So first the SOC of this cell must be elevated. Then, the module must enter into pure passive balancing operation.

4.4 Results

In the following sections the different scenarios presented are tested into the different models. The captures provide the information of the important variables for each of the tests.

4.4.1 ΔVOC

4.4.1.1 ΔVOC: Functional tests

ΔVOC: Cycle test, Initial SOC Difference of 5%

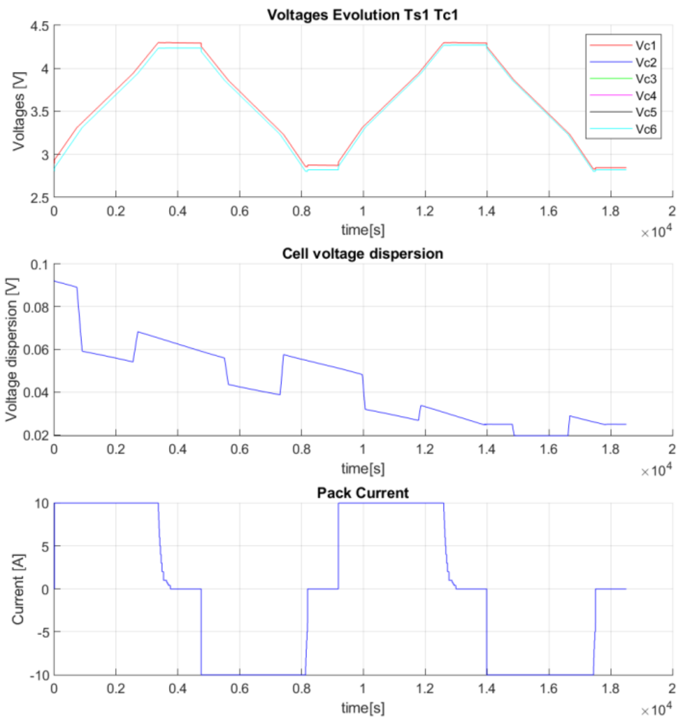


Figure 62. Functional test. Voltage figure.

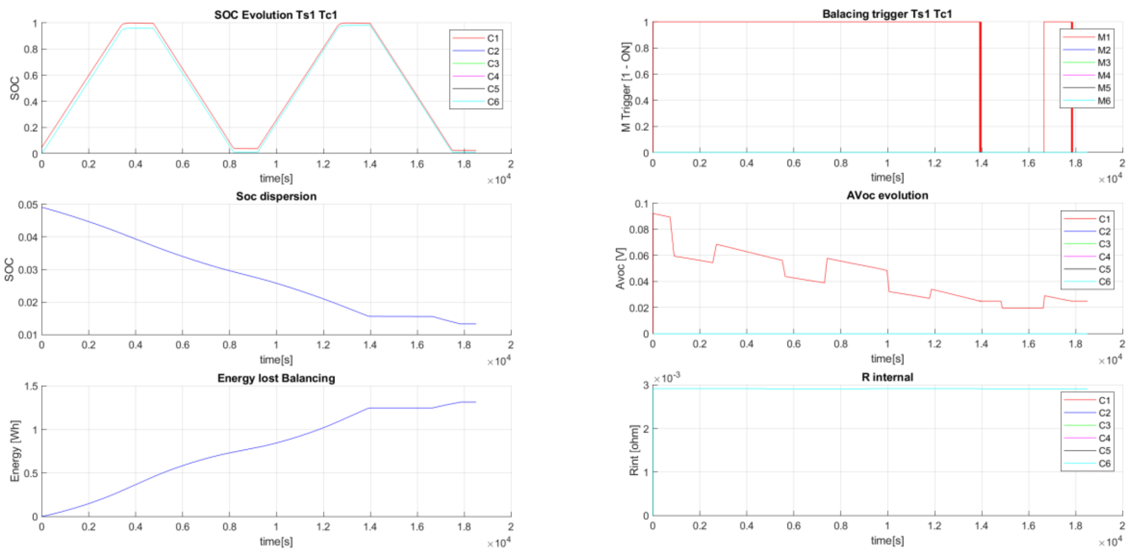


Figure 61. Functional test. Left: SOC evolution. Right: Balancing action.

ΔVOC: Cycle test, Initial SOC - 0% 3% 4% 5% 6% 1%

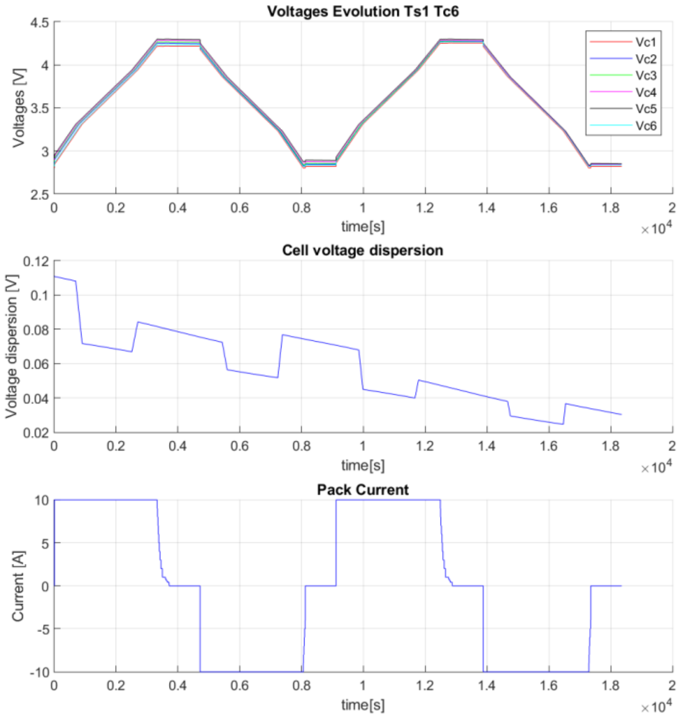


Figure 63. Functional test. Voltage figure.

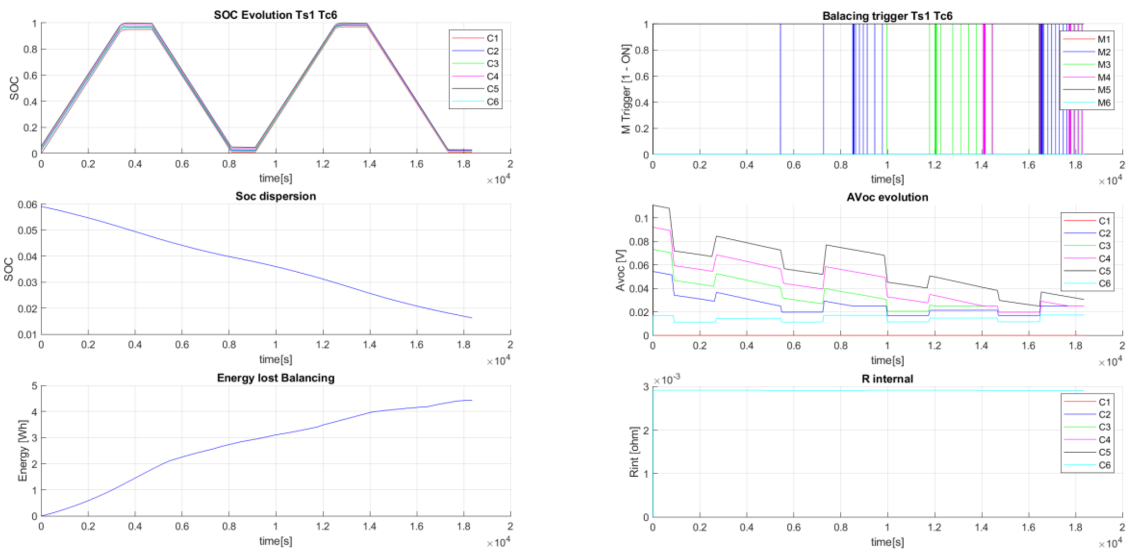


Figure 64. Functional test. Left: SOC evolution. Right: Balancing action.

ΔVOC: Profile test, Initial SOC - 80% 70% 70% 70% 70% 70%

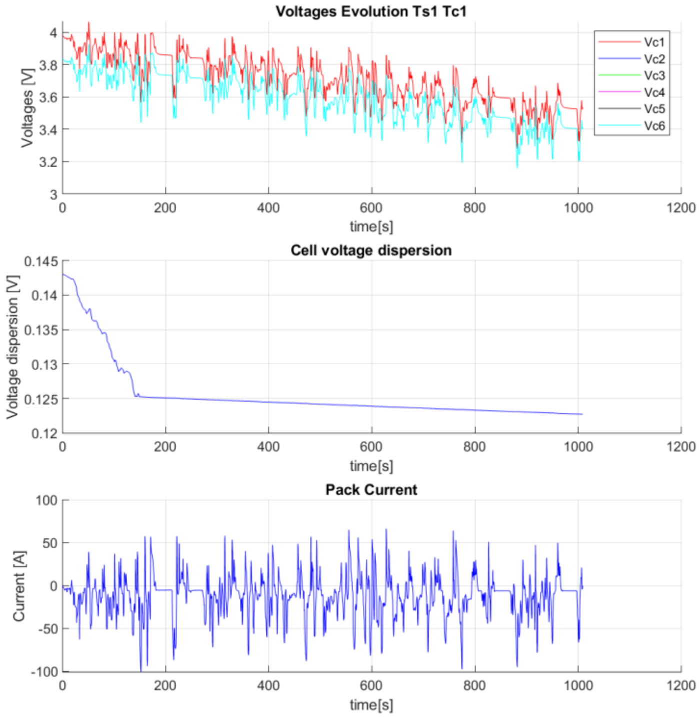


Figure 65. Functional test. Voltage figure.

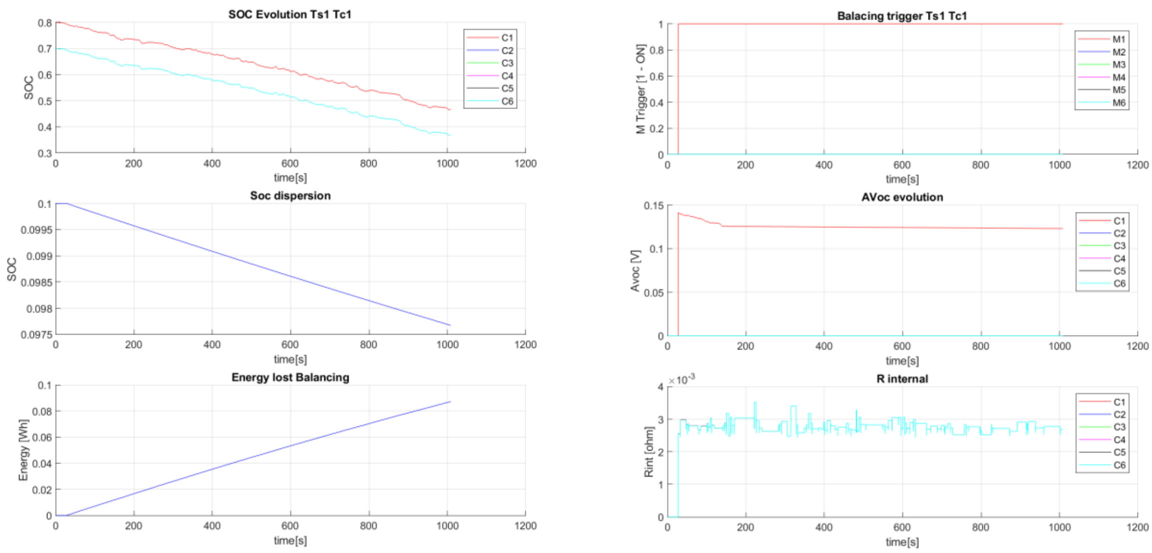


Figure 66. Functional test. Left: SOC evolution. Right: Balancing action.

ΔVOC: Profile test, Initial SOC - 65% 65% 55% 55% 70% 70%

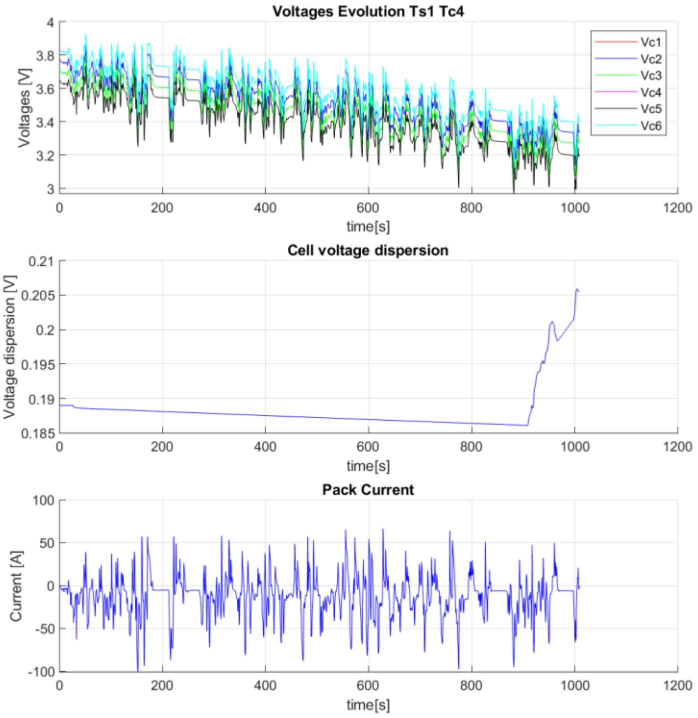


Figure 67. Functional test. Voltage figure.

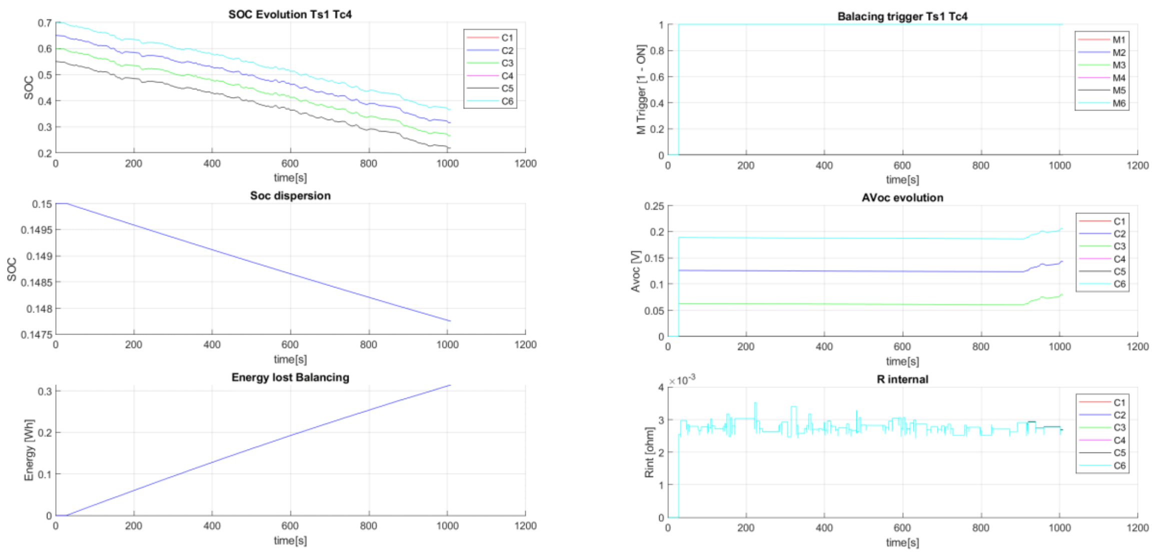


Figure 68. Functional test. Left: SOC evolution. Right: Balancing action.

4.4.1.2 ΔVOC: Aging

ΔVOC: Internal resistance +50%, Initial SOC difference of 5%

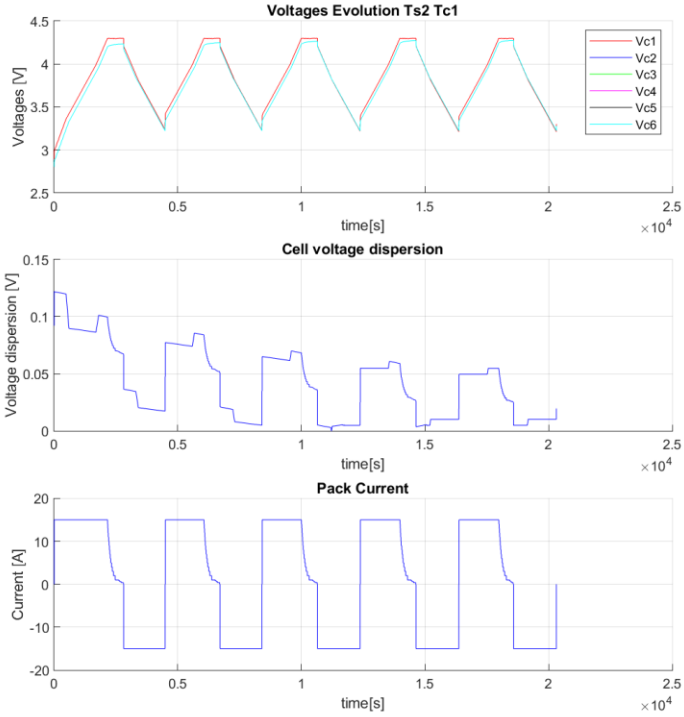


Figure 70. Internal resistance aging test. Voltage figure.

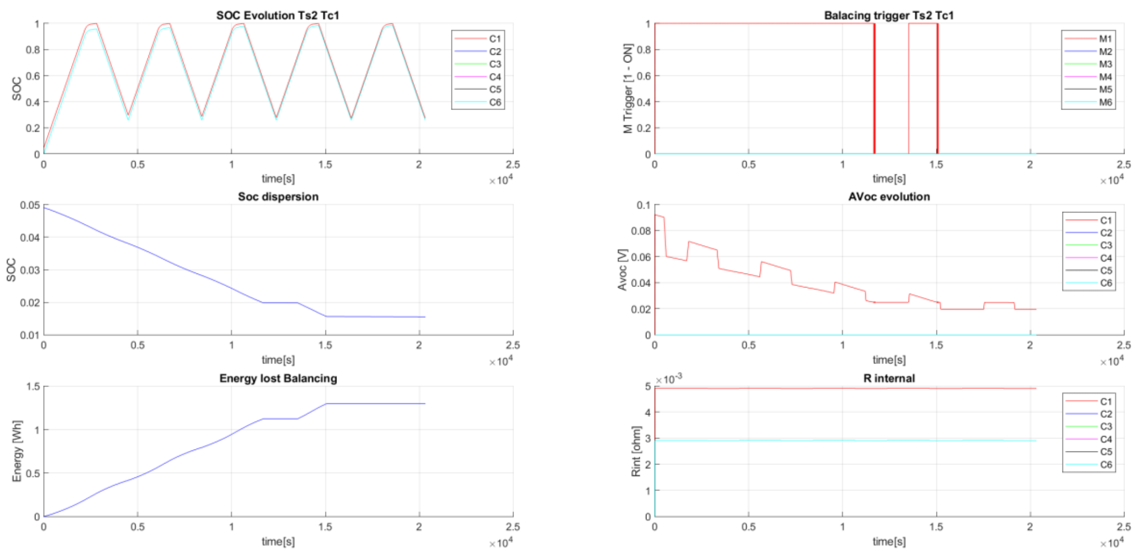


Figure 69. Internal resistance aging test. Left: SOC evolution. Right: Balancing action.

ΔVOC: Internal resistance -50%, Initial SOC difference of 5%

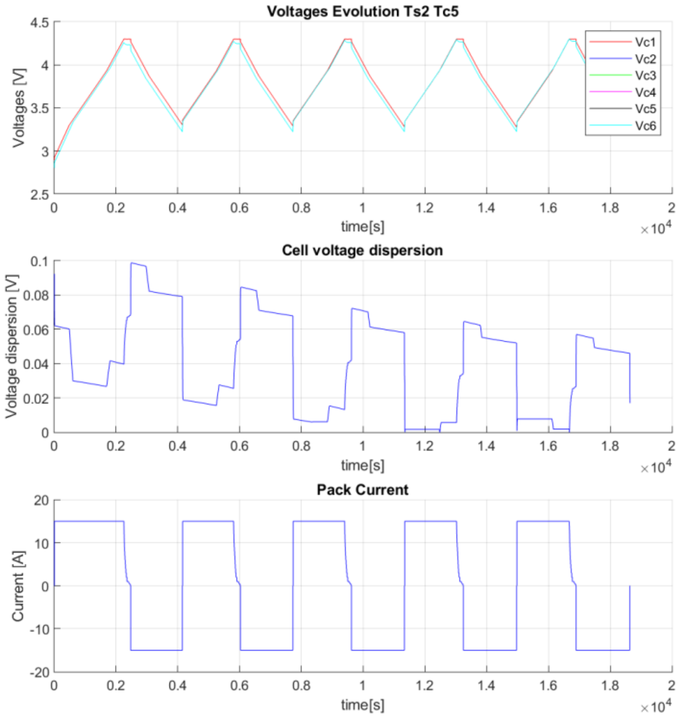


Figure 71. Internal resistance aging test. Voltage figure.

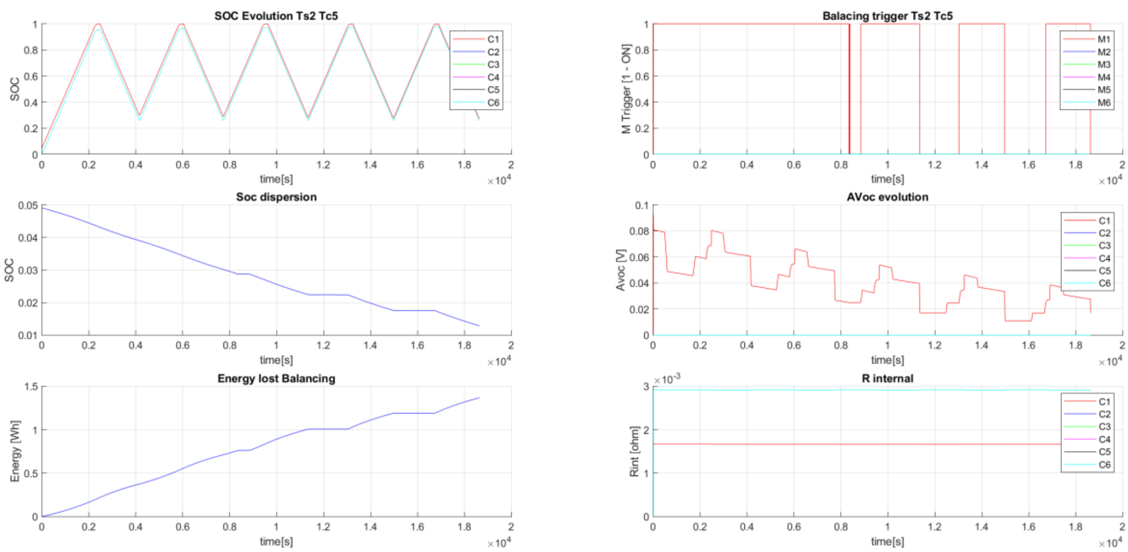


Figure 72. Internal resistance aging test Left: SOC evolution. Right: Balancing action.

ΔVOC: Capacity +10%

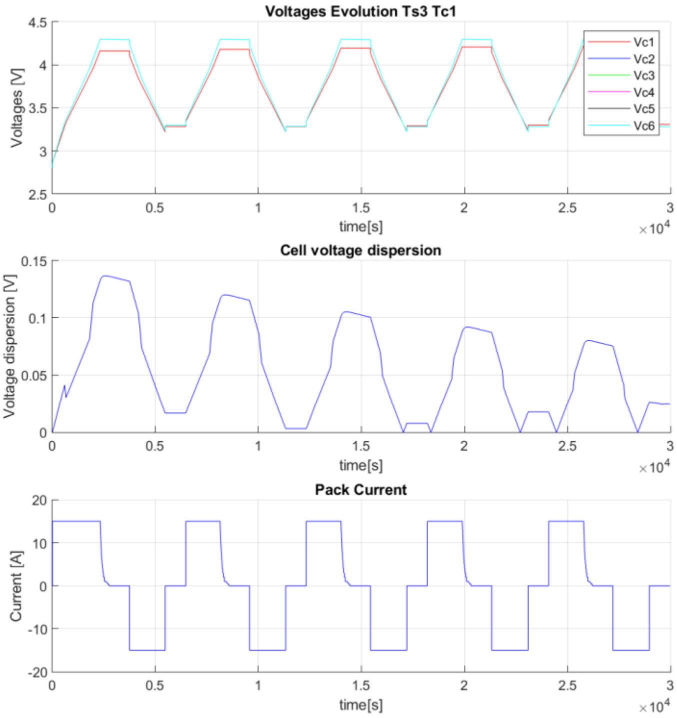


Figure 73. Capacity aging test. Voltage figure.

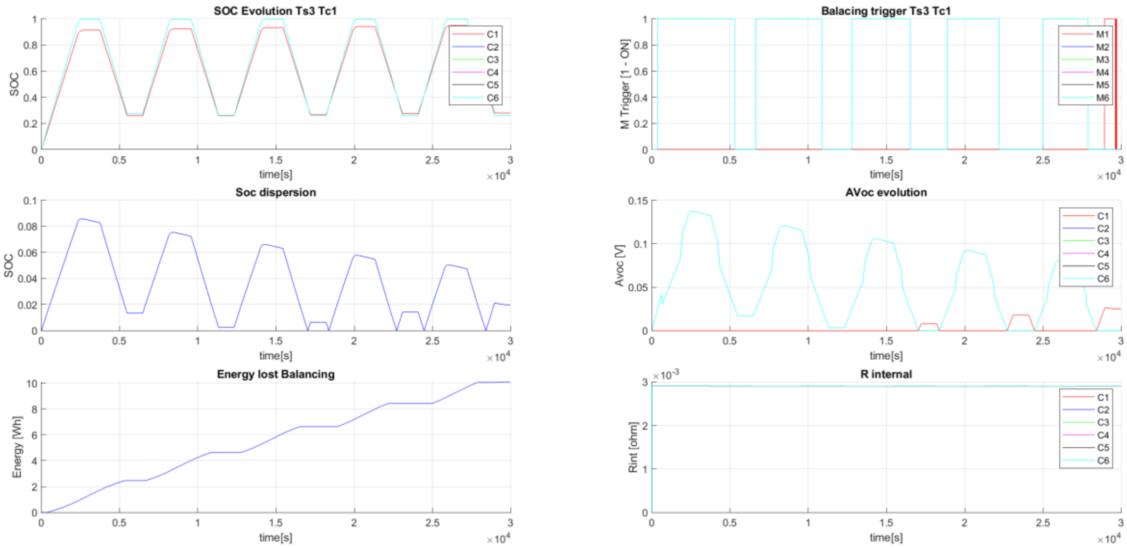


Figure 74. Capacity aging test. Left: SOC evolution. Right: Balancing action.

ΔVOC: Capacity -10%

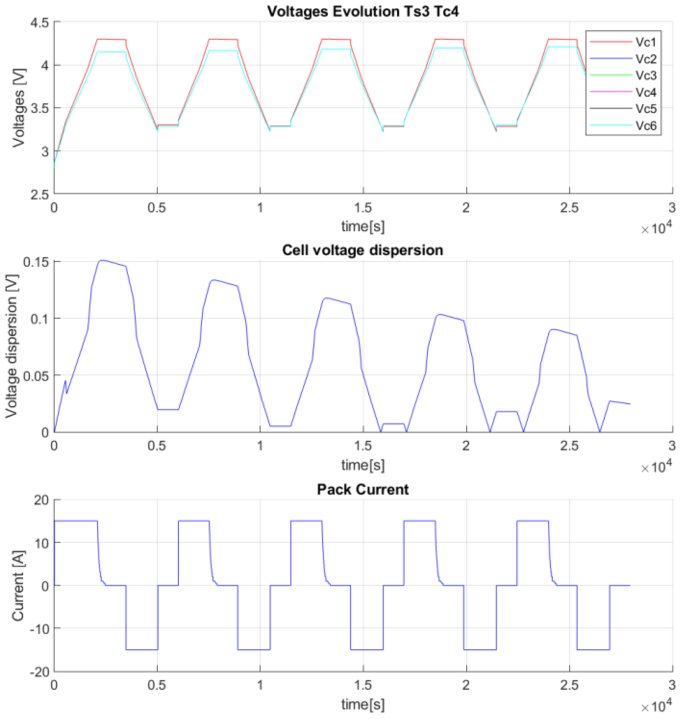


Figure 75. Capacity aging test. Voltage figure.

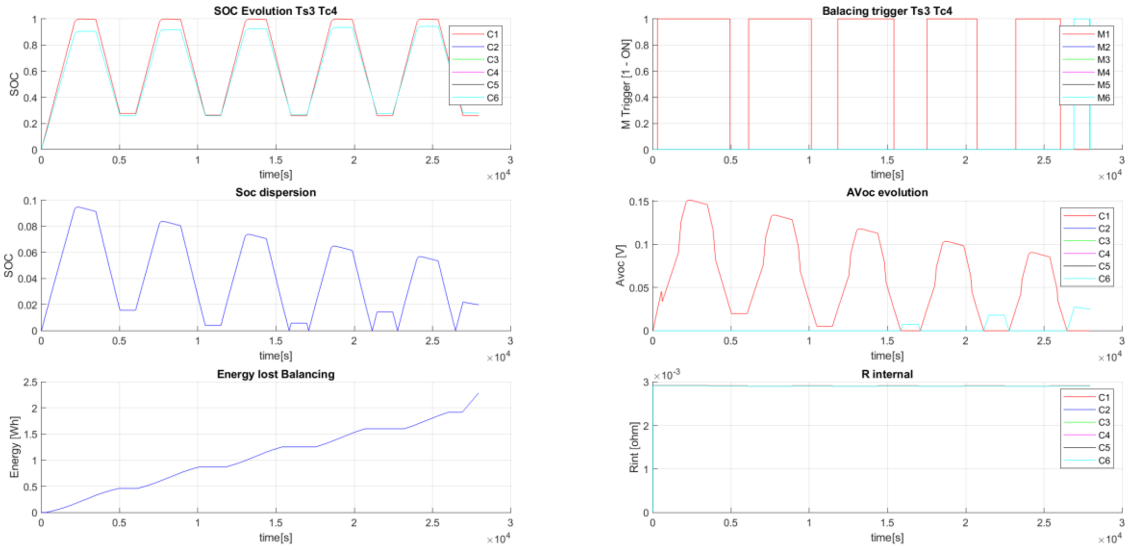


Figure 76. Capacity aging test. Left: SOC evolution. Right: Balancing action.

ΔVOC: VOC-SOC Table Change

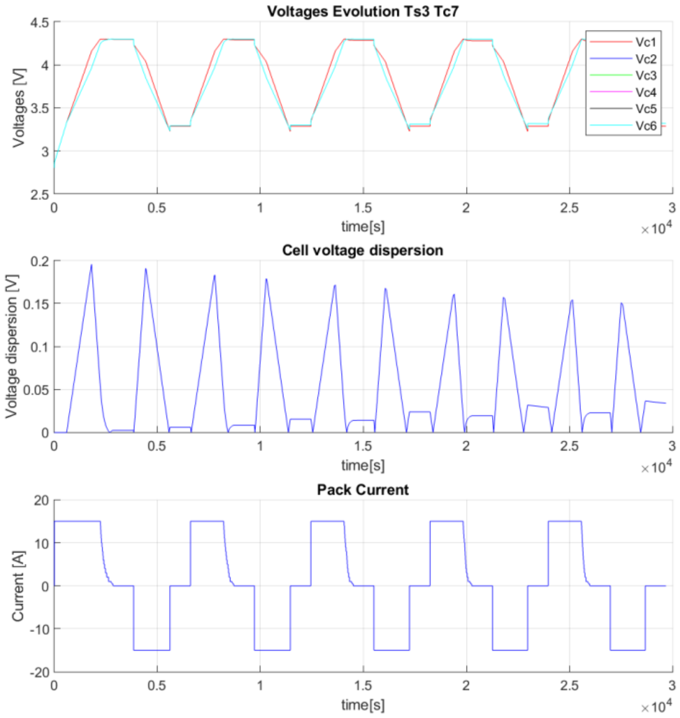


Figure 77. VOC-SOC table change test. Voltage figure.

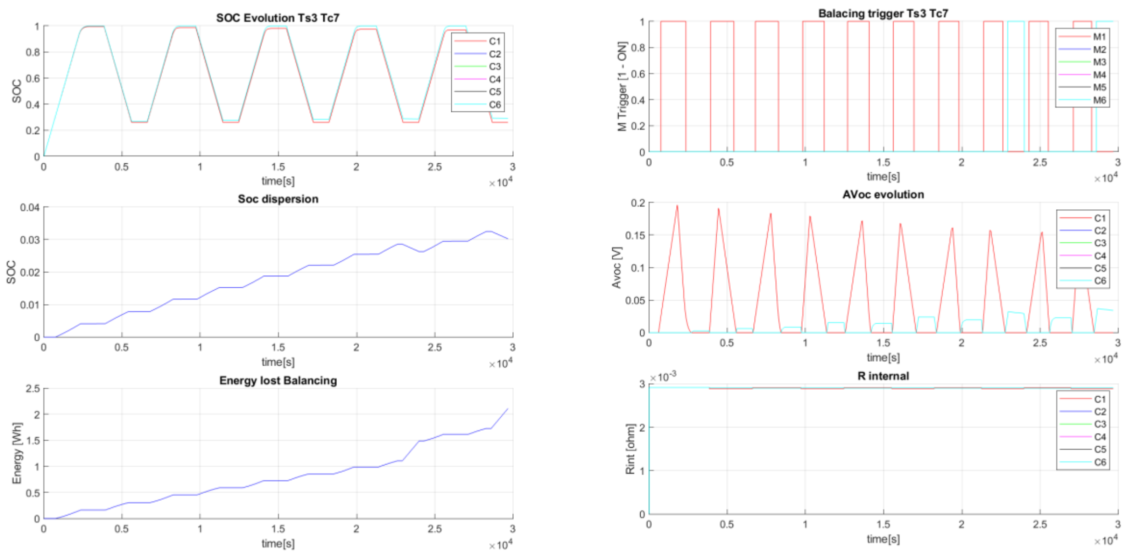


Figure 78. VOC-SOC table change test. Left: SOC evolution. Right: Balancing action

4.4.1.3 ΔVOC: Error tolerance

ΔVOC: Cell Voltage measurement offset, +50 mV

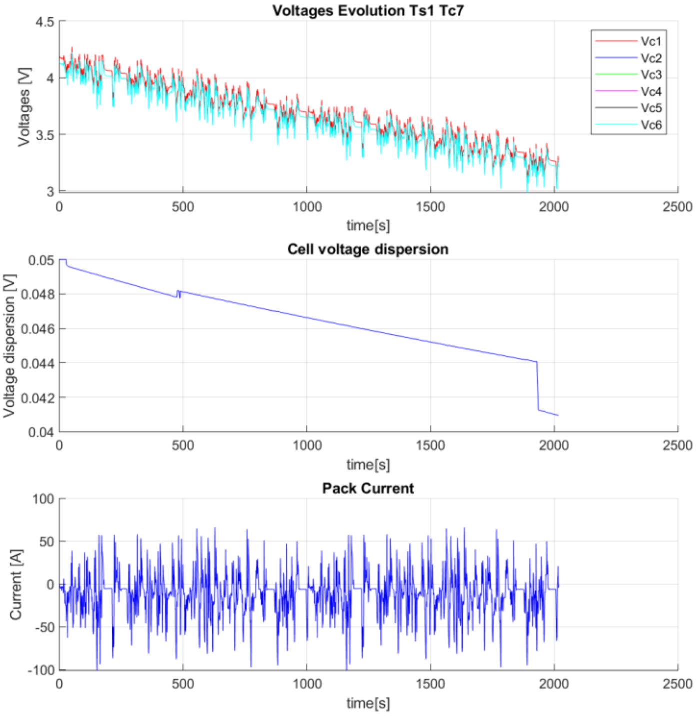


Figure 79. Cell voltage error introduction test. Voltage figure.

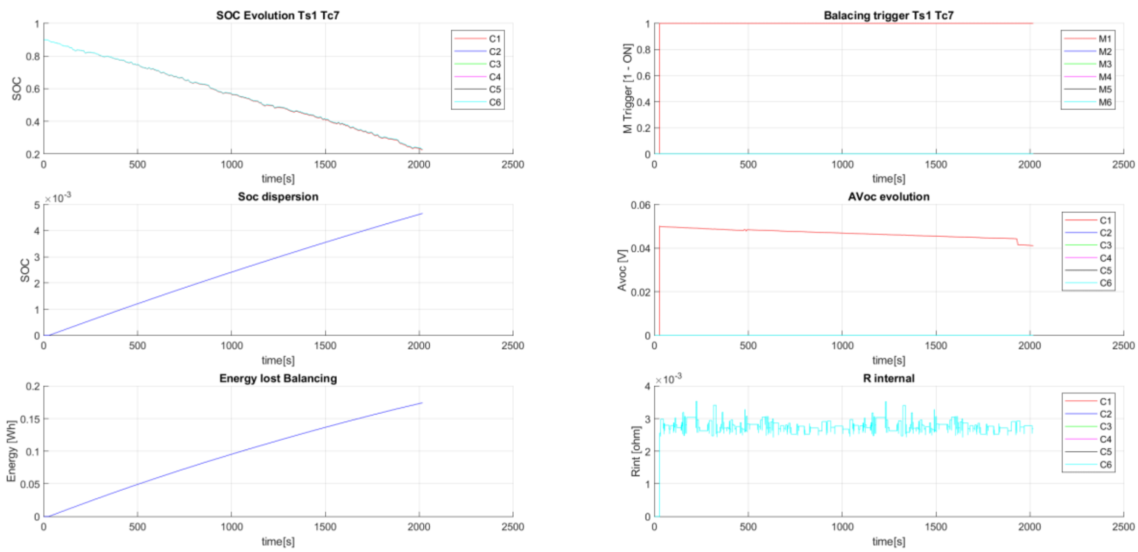


Figure 80. Cell voltage error introduction test. Left: SOC evolution. Right: Balancing action..

ΔVOC: Cell Voltage measurement offset, -50 mV

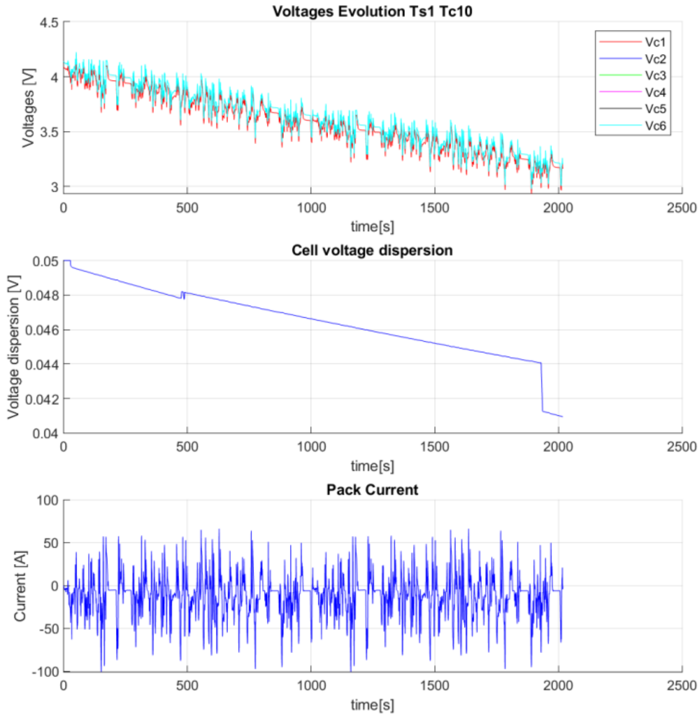


Figure 81. Cell voltage error introduction test. Voltage figure.

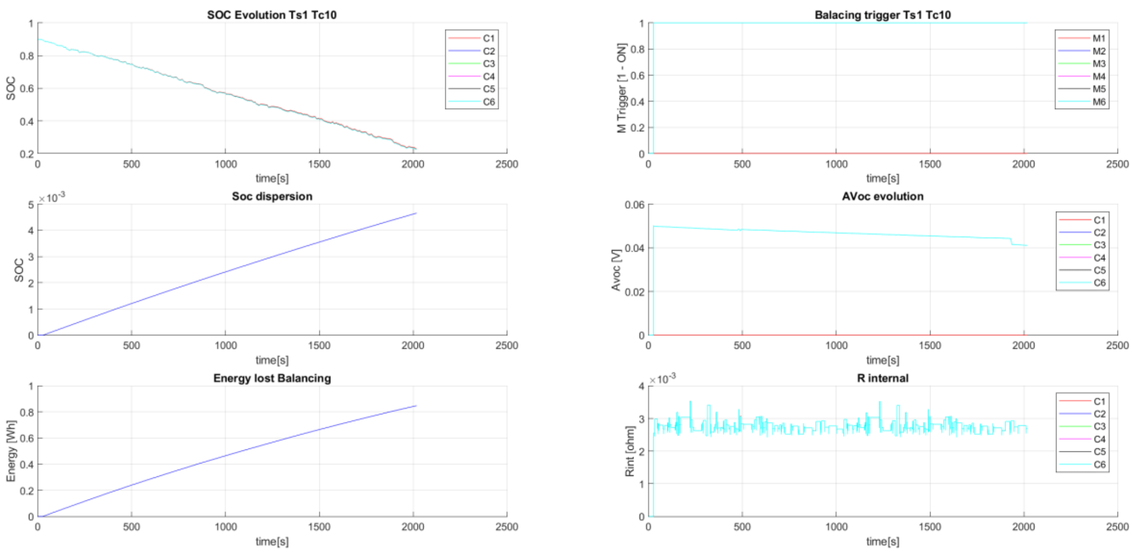


Figure 82. Cell voltage error introduction test. Left: SOC evolution. Right: Balancing action.

ΔVOC: SOC estimation offset, +2%

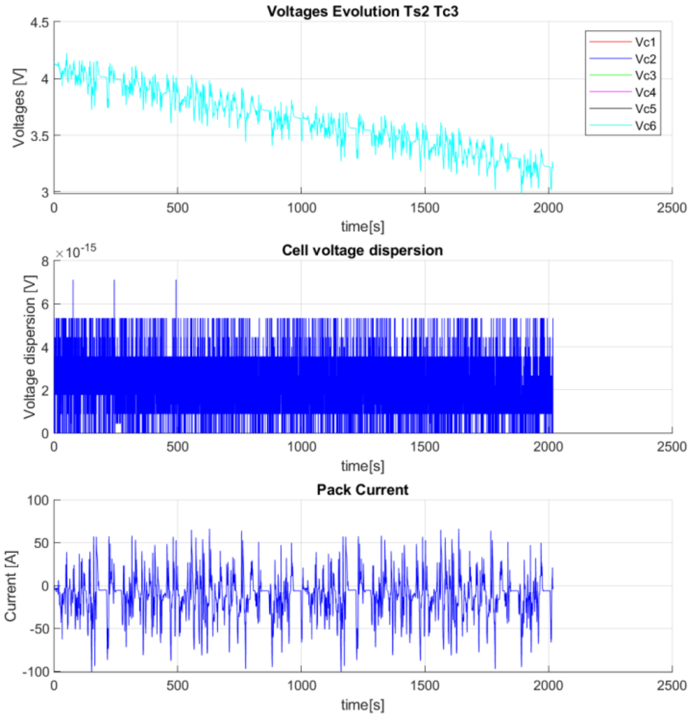


Figure 83. SOC estimation error introduction test. Voltage figure.

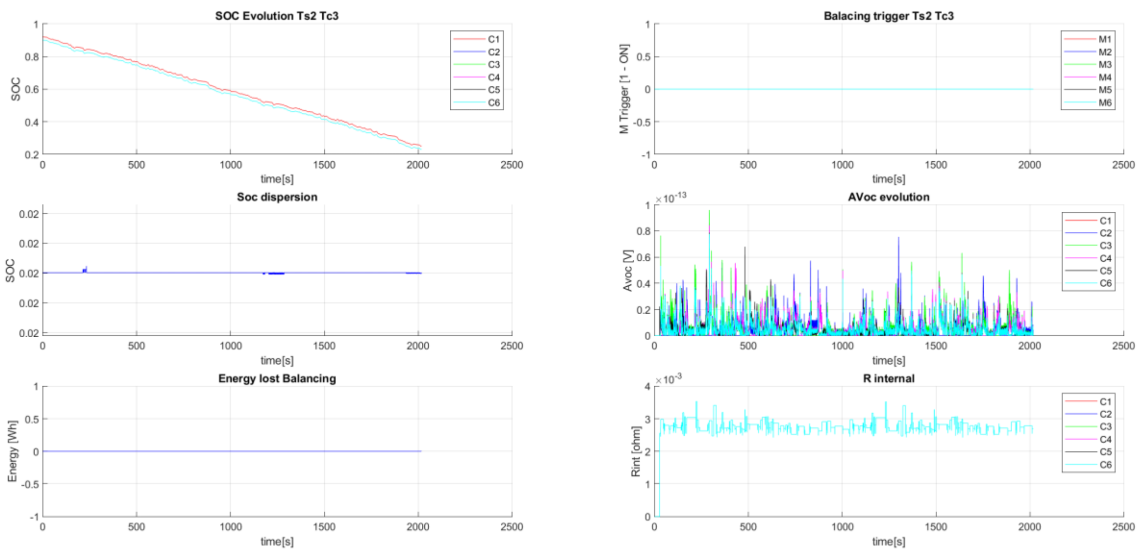


Figure 84. SOC estimation error introduction test. Left: SOC evolution. Right: Balancing action.

ΔVOC: SOC estimation offset, -2%

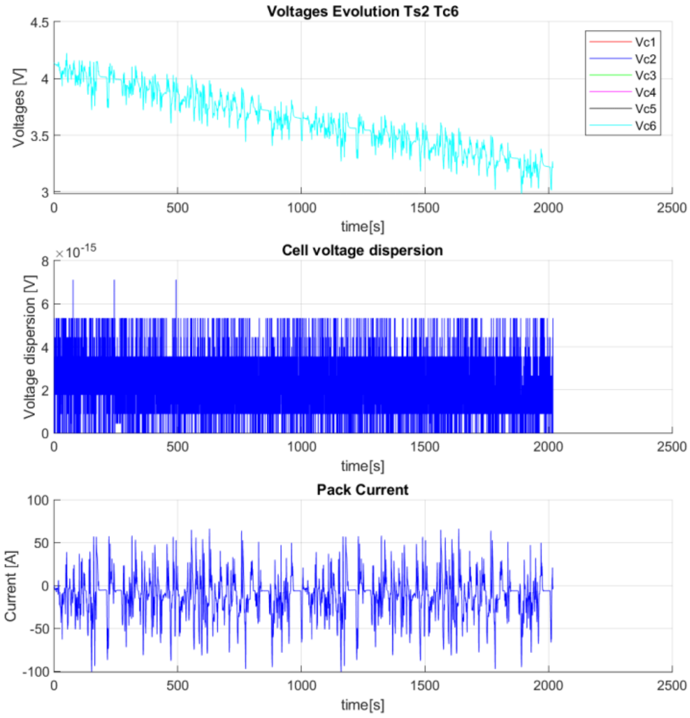


Figure 85. SOC estimation error introduction test. Voltage figure.

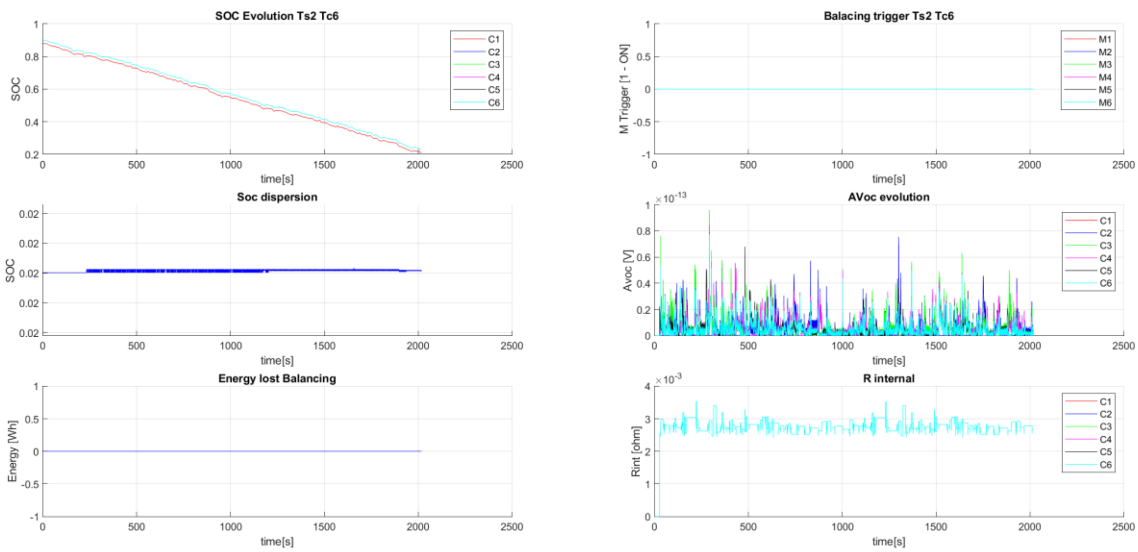


Figure 86. SOC estimation error introduction test. Left: SOC evolution. Right: Balancing action.

ΔVOC: Current sense offset, +250 mA

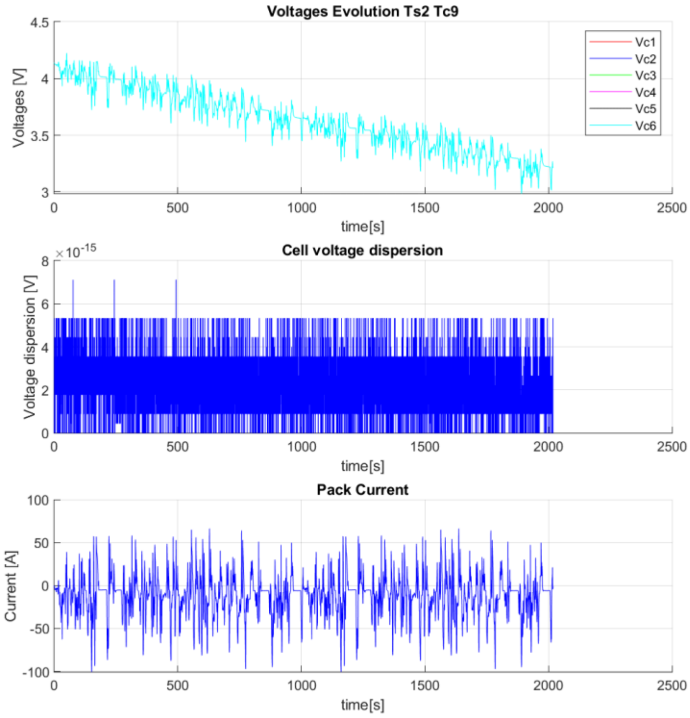


Figure 87. Current sense error introduction test. Voltage figure.

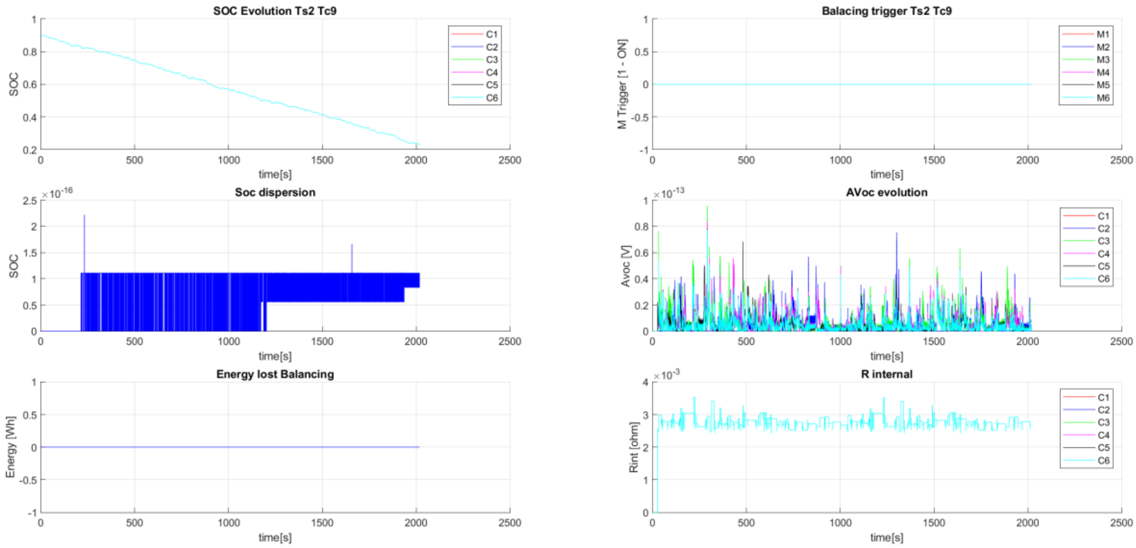


Figure 88. Current sense error introduction test. Left: SOC evolution. Right: Balancing action.

ΔVOC: Current sense offset, - 250 mA

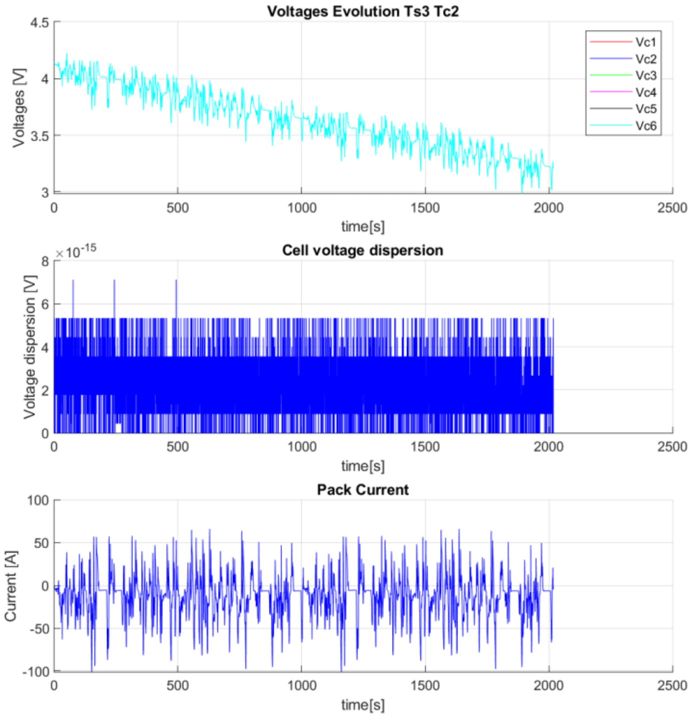


Figure 89. Current sense error introduction test. Voltage figure.

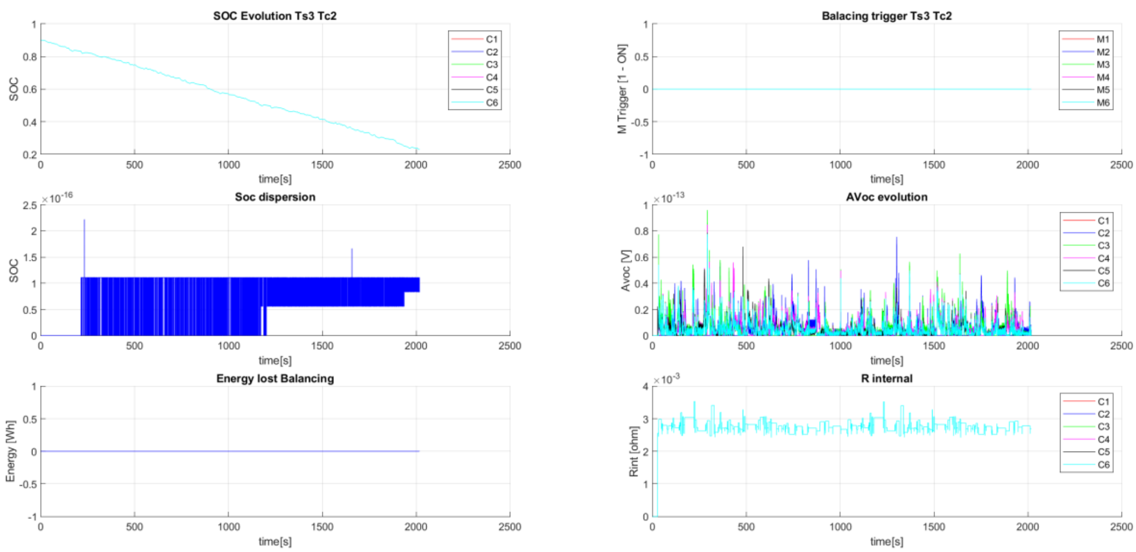


Figure 90. Current sense error introduction test. Left: SOC evolution. Right: Balancing action.

4.4.2 Final Voltage

4.4.2.1 Final Voltage: Functional tests

Final Voltage ΔVOC: Cycle test, Initial SOC Difference of 5%

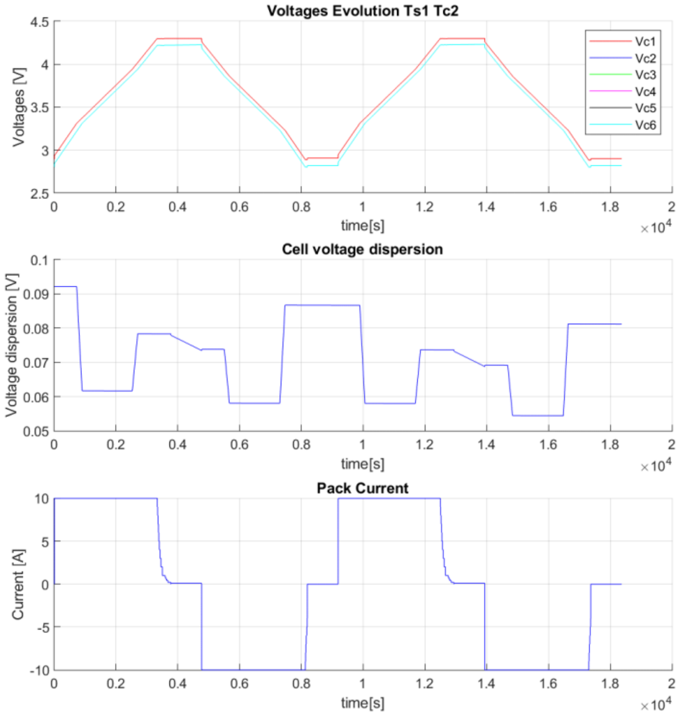


Figure 91. Functional test. Voltage figure.

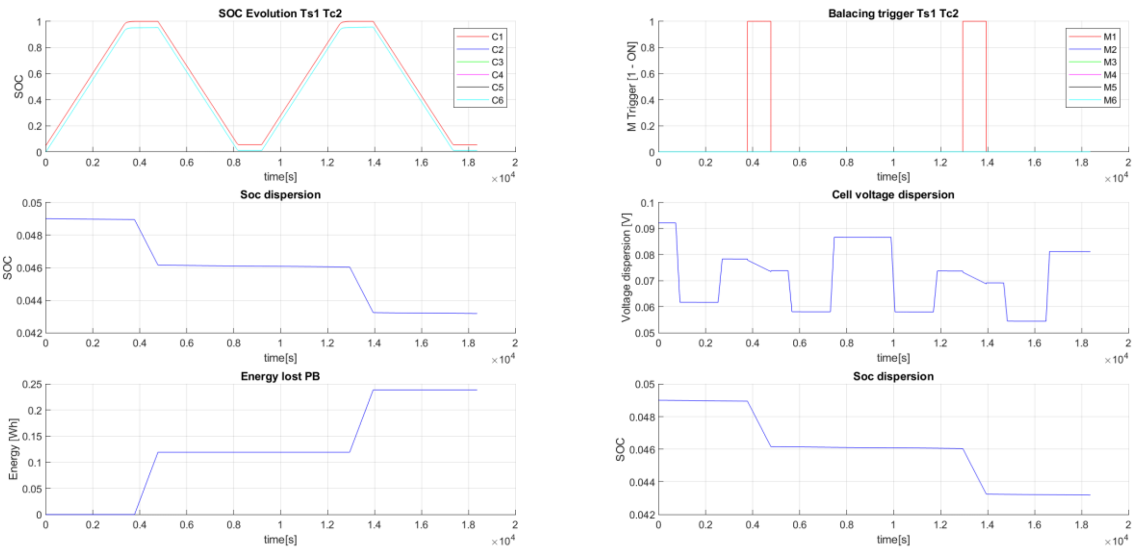


Figure 92. Functional test. Left: SOC evolution. Right: Balancing action.

Final Voltage ΔVOC: Cycle test, Initial SOC - 0% 3% 4% 5% 6% 1%

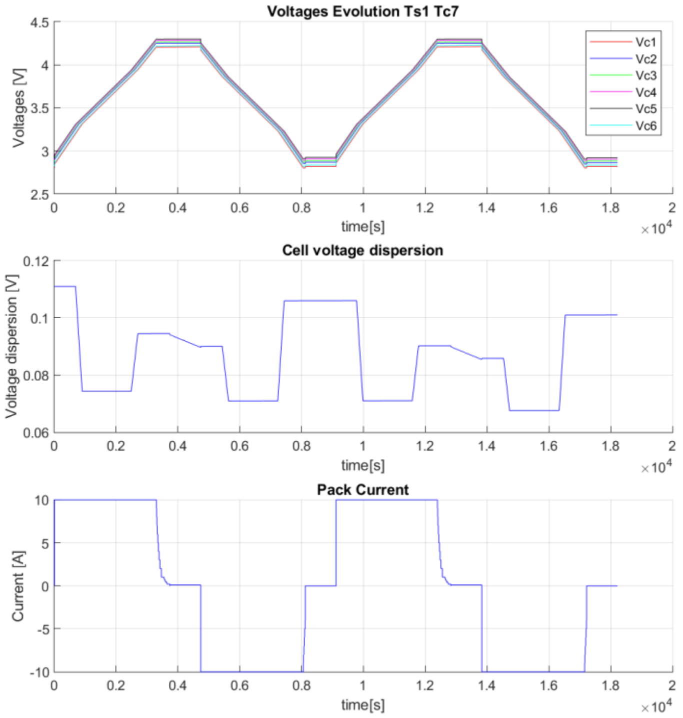


Figure 93. Functional test. Voltage figure.

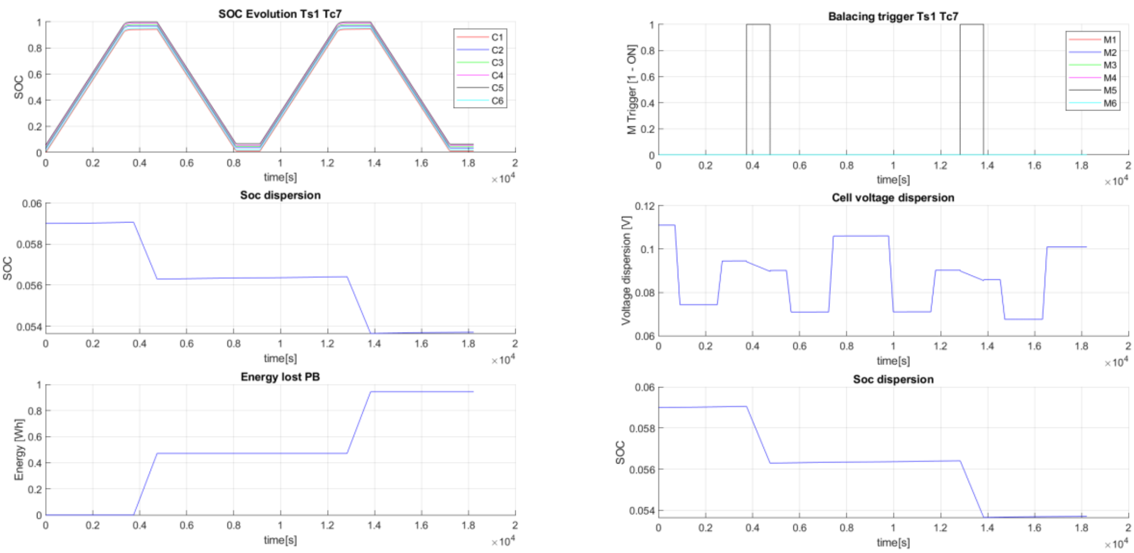


Figure 94. Functional test. Left: SOC evolution. Right: Balancing action.

Final Voltage Pulsed Cha: Cycle test, Initial SOC Difference of 5%

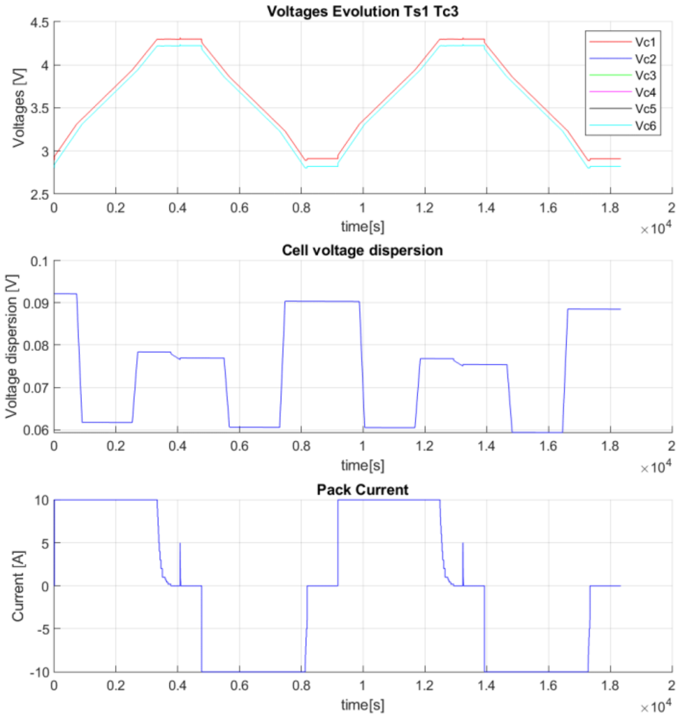


Figure 95. Functional test. Voltage figure.

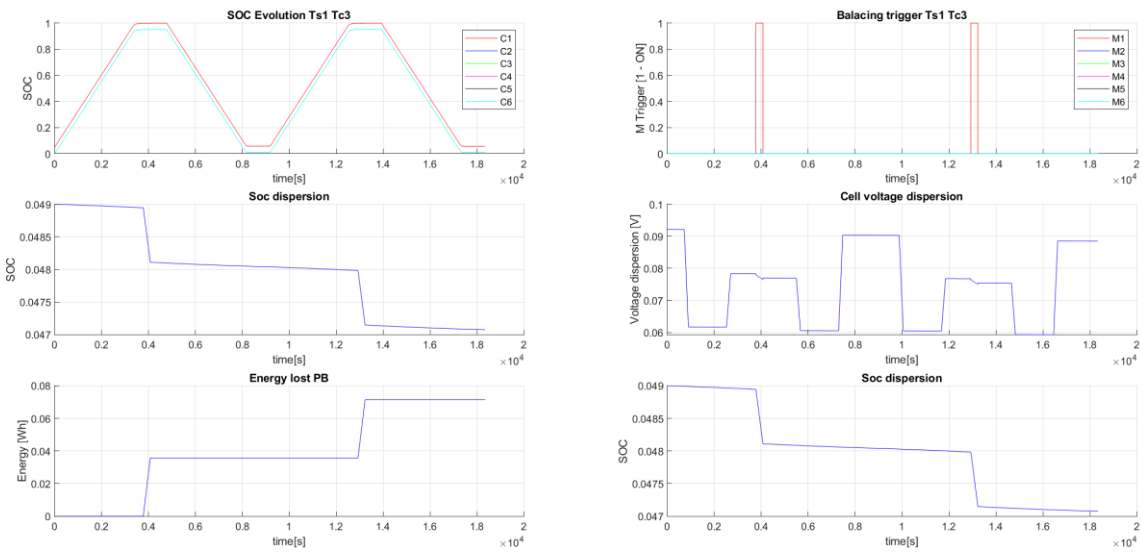


Figure 96. Functional test. Left SOC: Evolution. Right: Balancing action.

Final Voltage ΔVOC: Cycle test, Initial SOC - 0% 3% 4% 5% 6% 1%

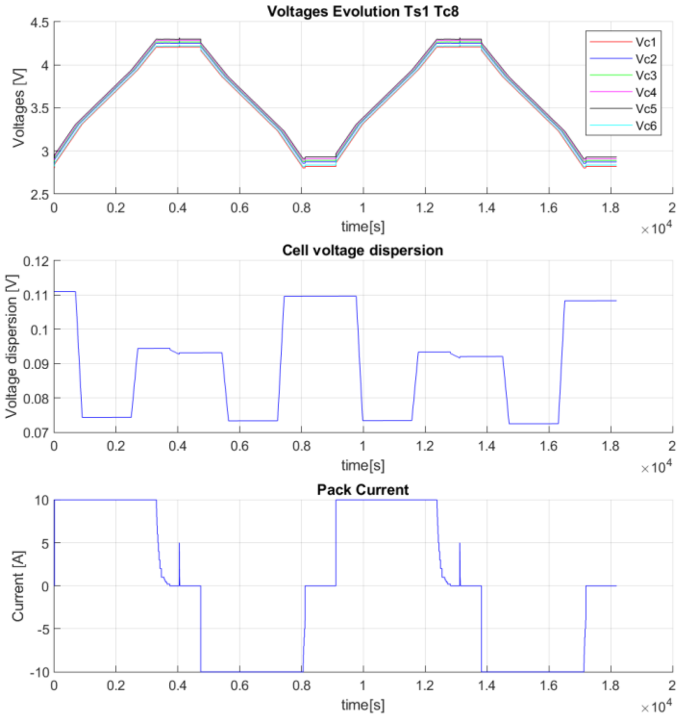


Figure 97. Functional test. Voltage figure.

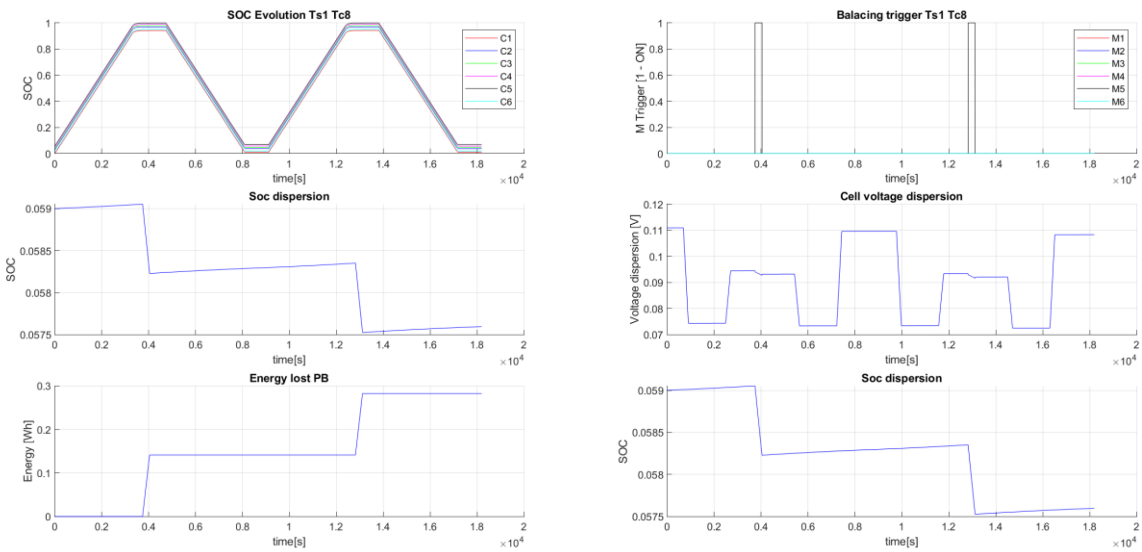


Figure 98. Functional test. Left: SOC evolution. Right: Balancing action.

4.4.2.2 Final voltage: Aging

Final Voltage: Internal resistance +50%, Initial SOC difference of 5%

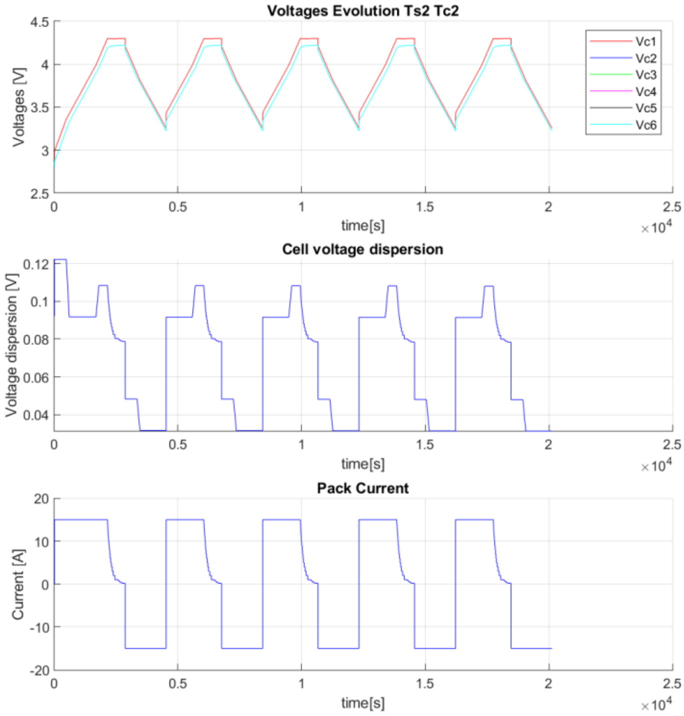


Figure 99. Internal resistance aging test. Voltage figure.

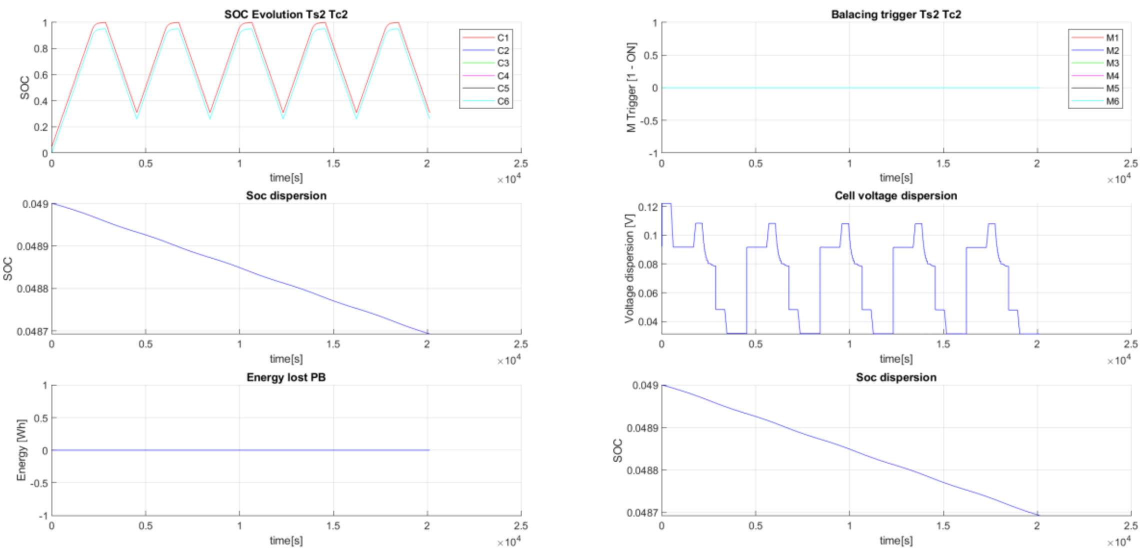


Figure 100. Internal resistance aging test. Left: SOC evolution. Right: Balancing action.

Final Voltage: Internal resistance -50%, Initial SOC difference of 5%

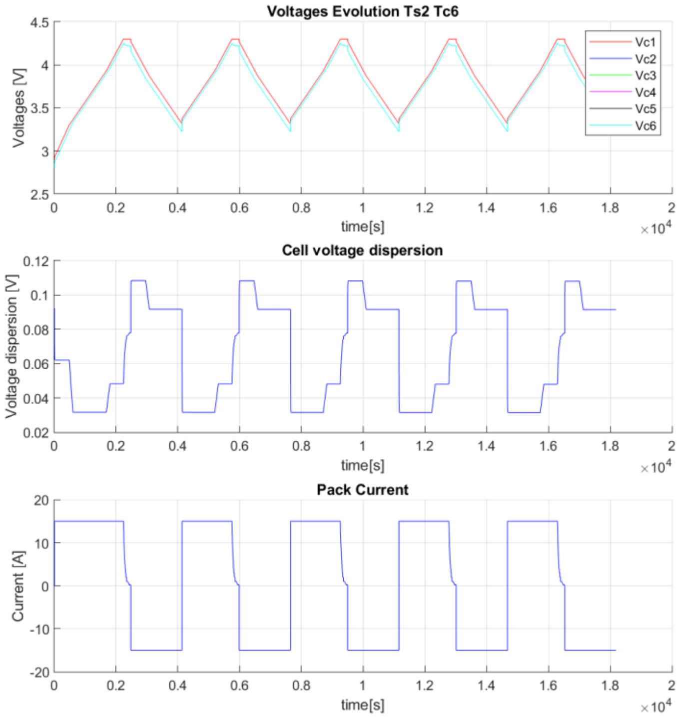


Figure 101. Internal resistance aging test. Voltage figure.

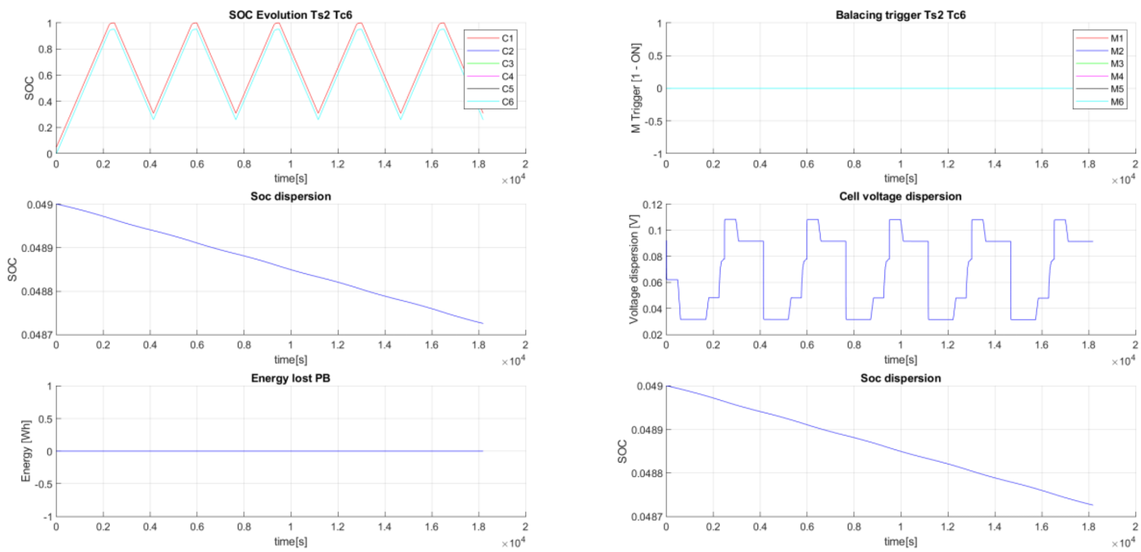


Figure 102. Internal resistance aging test. Left: SOC evolution. Right: Balancing action.

4.4.3 SOC History

4.4.3.1 SOC History: Functional tests

SOC History: Cycle test, Initial SOC Difference of 5%

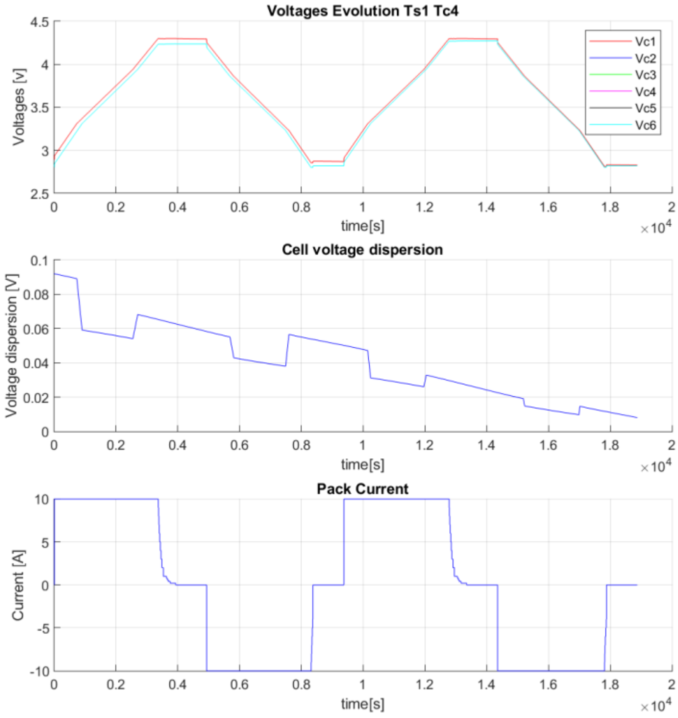


Figure 103. Functional test. Voltage figure.

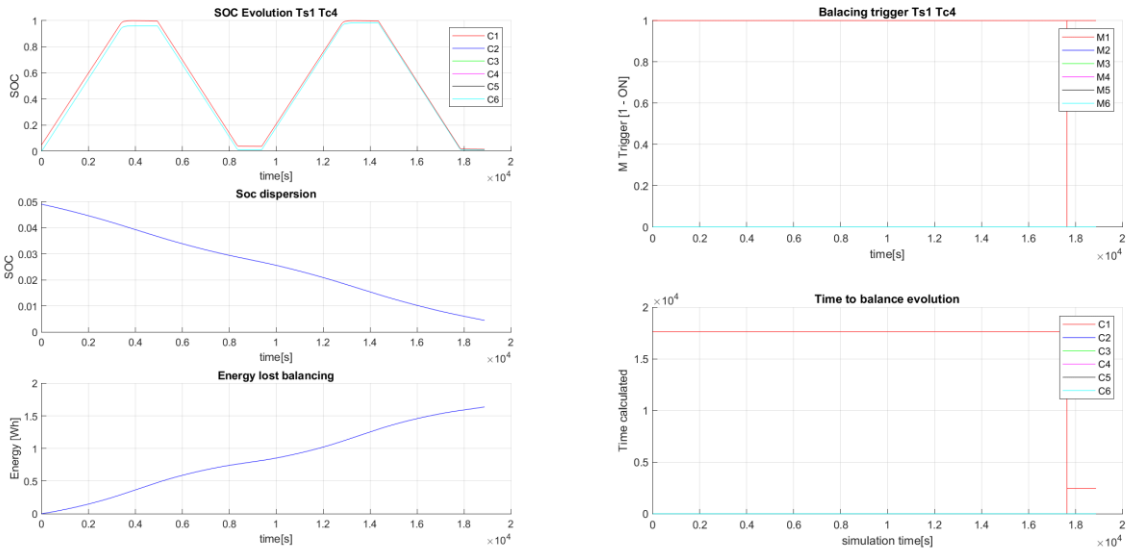


Figure 104. Functional test. Left: SOC evolution. Right: Balancing action.

SOC History: Cycle test, Initial SOC - 0% 3% 4% 5% 6% 1%

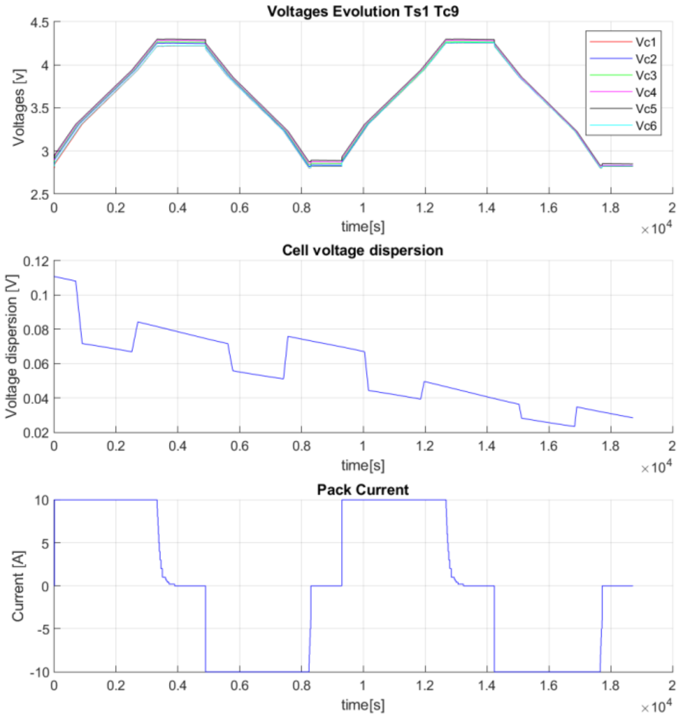


Figure 105. Functional test. Voltage figure.

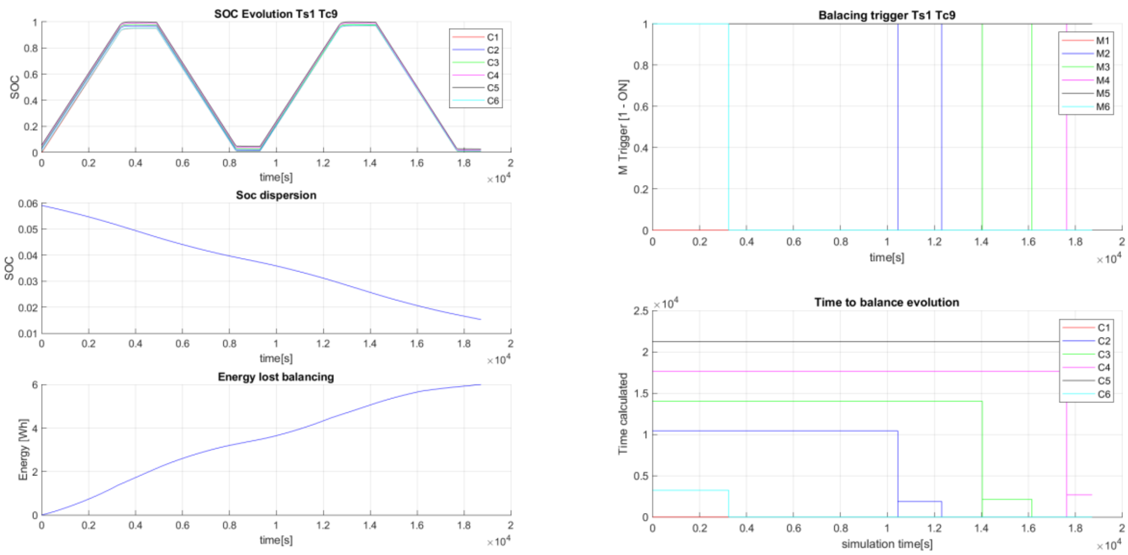


Figure 106. Functional test. Left: SOC evolution. Right: Balancing action.

SOC History: Profile test, Initial SOC - 80% 70% 70% 70% 70% 70%

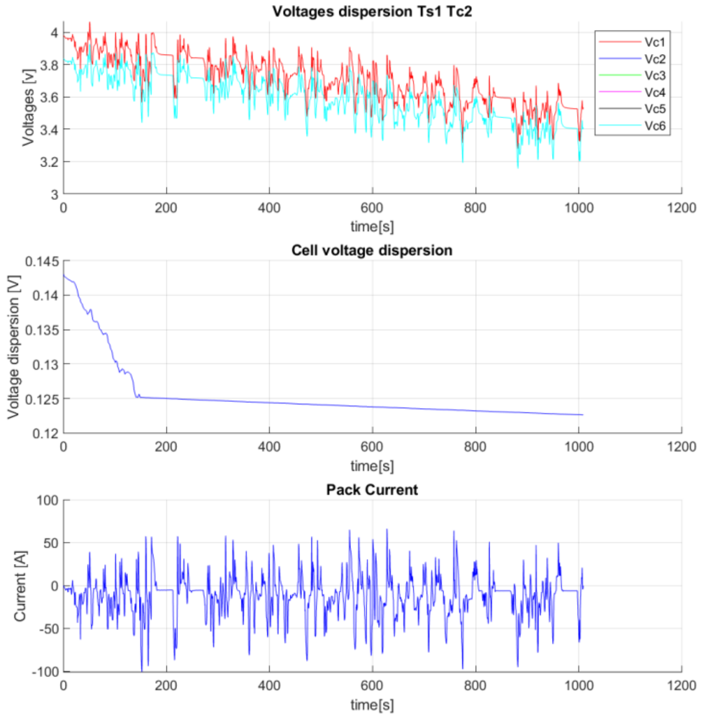


Figure 107. Functional test. Voltage figure.

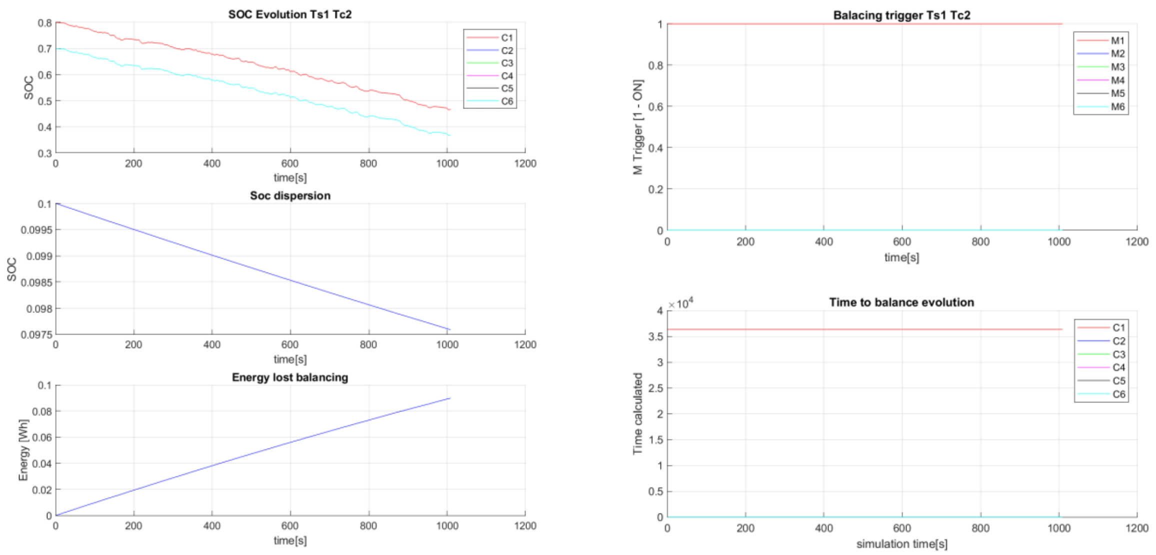


Figure 108. Functional test. Left: SOC evolution. Right: Balancing action.

SOC History: Profile test, Initial SOC - 65% 65% 55% 55% 70% 70%

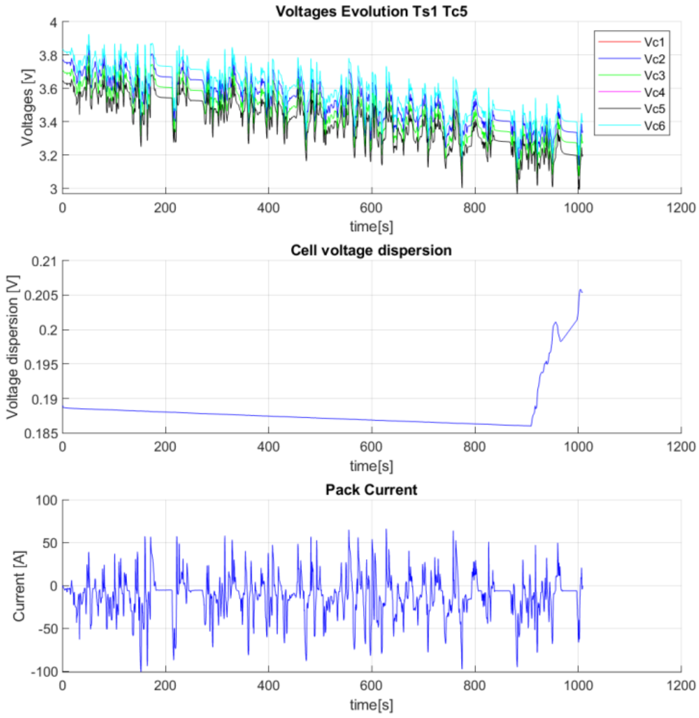


Figure 109. Functional test. Voltage figure.

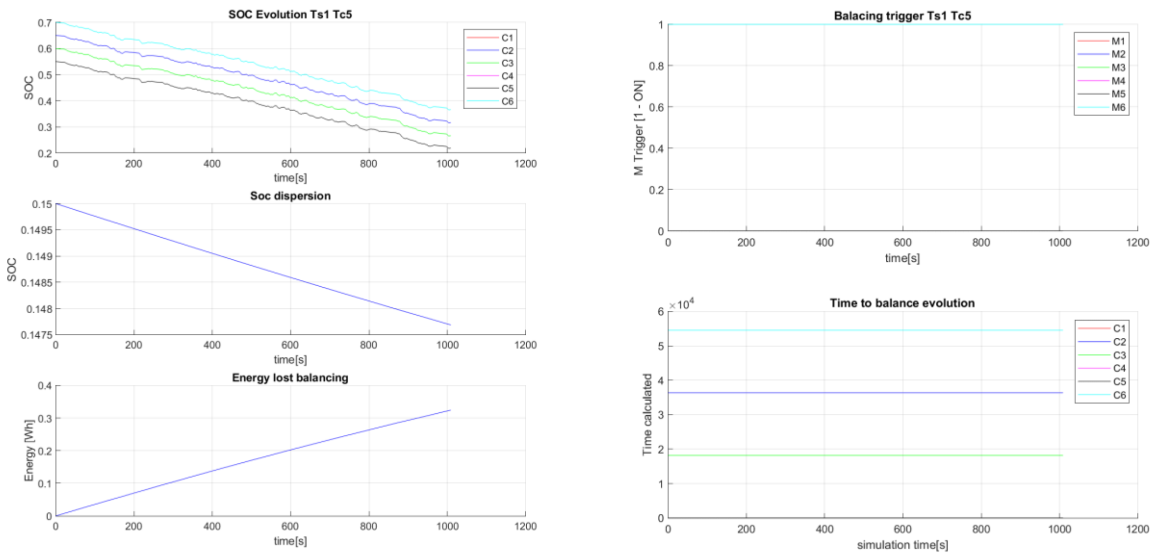


Figure 110. Functional test. Left: SOC evolution. Right: Balancing action.

4.4.3.2 SOC History: Aging

SOC History: Internal resistance +50%, Initial SOC difference of +5%

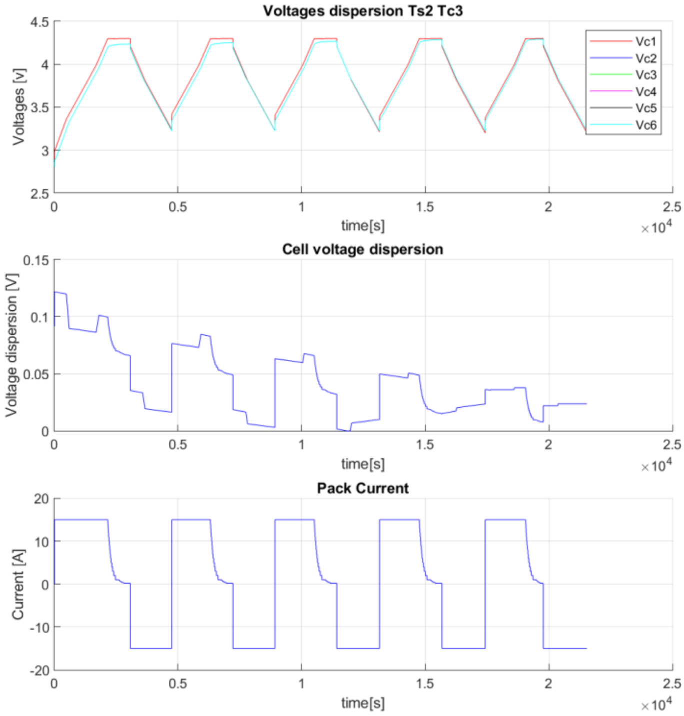


Figure 111. Internal resistance aging test. Voltage figure.

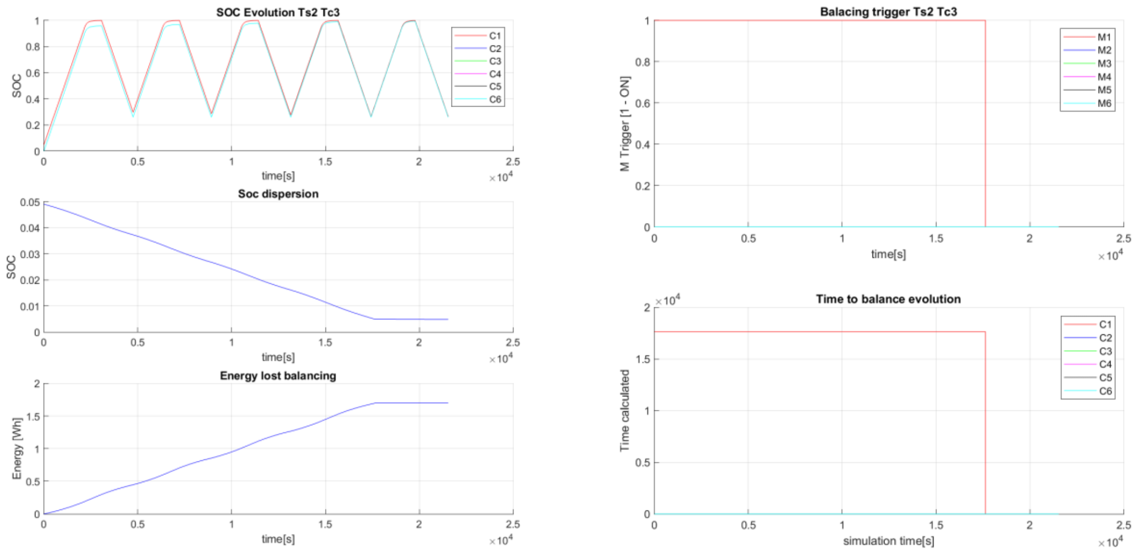


Figure 112. Internal resistance aging test. Left: SOC evolution. Right: Balancing action.

SOC History: Internal resistance -50%, Initial SOC difference of +5%

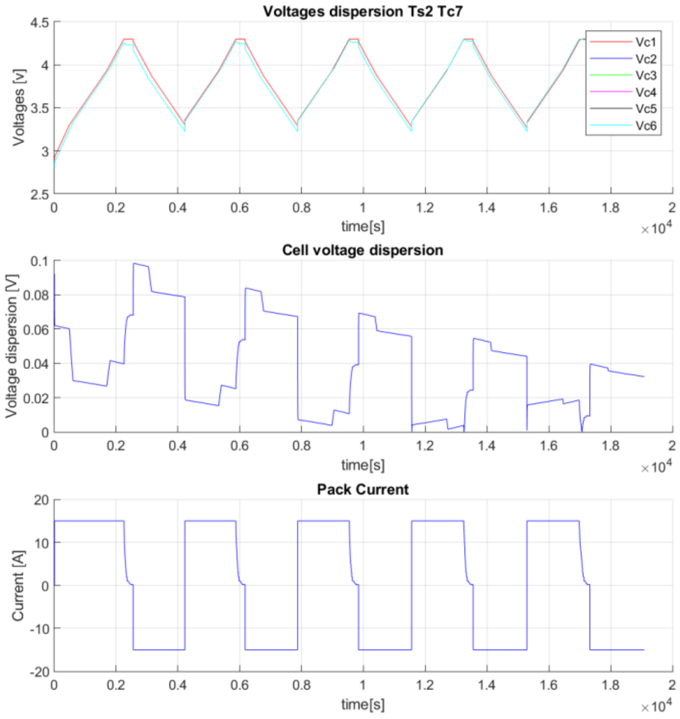


Figure 113. Internal resistance aging test. Voltage figure.

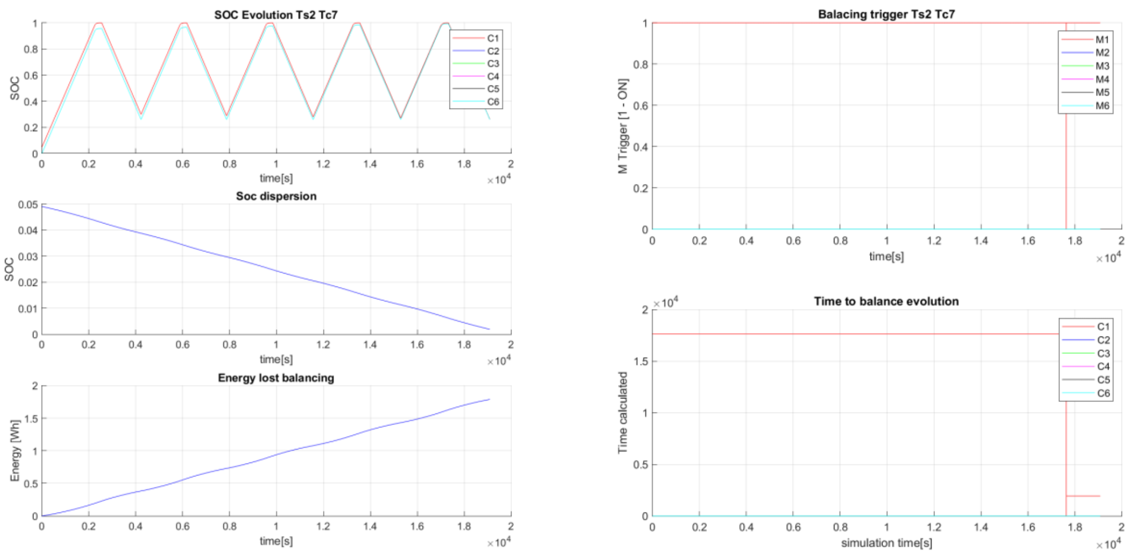


Figure 114. Internal resistance aging test. Left: SOC evolution. Right: Balancing action.

SOC History: Capacity +10%

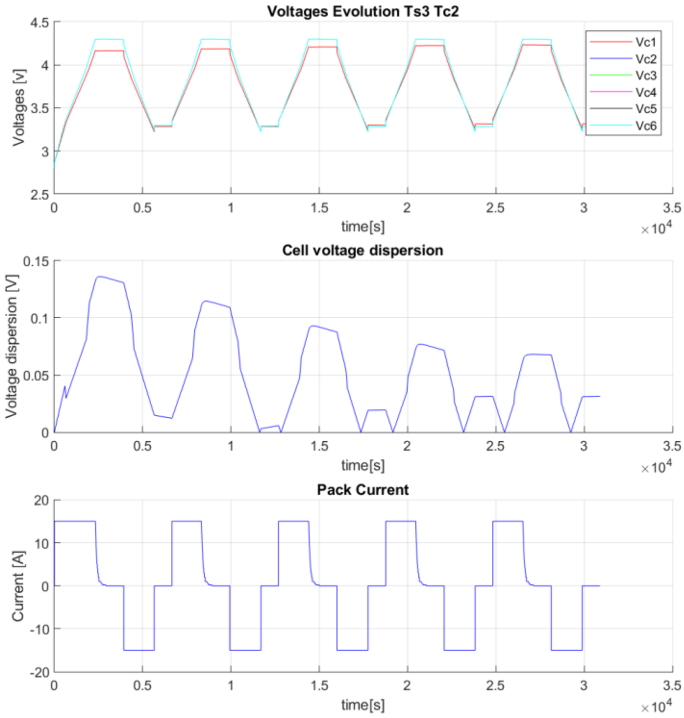


Figure 115. Capacity aging test. Voltage figure.

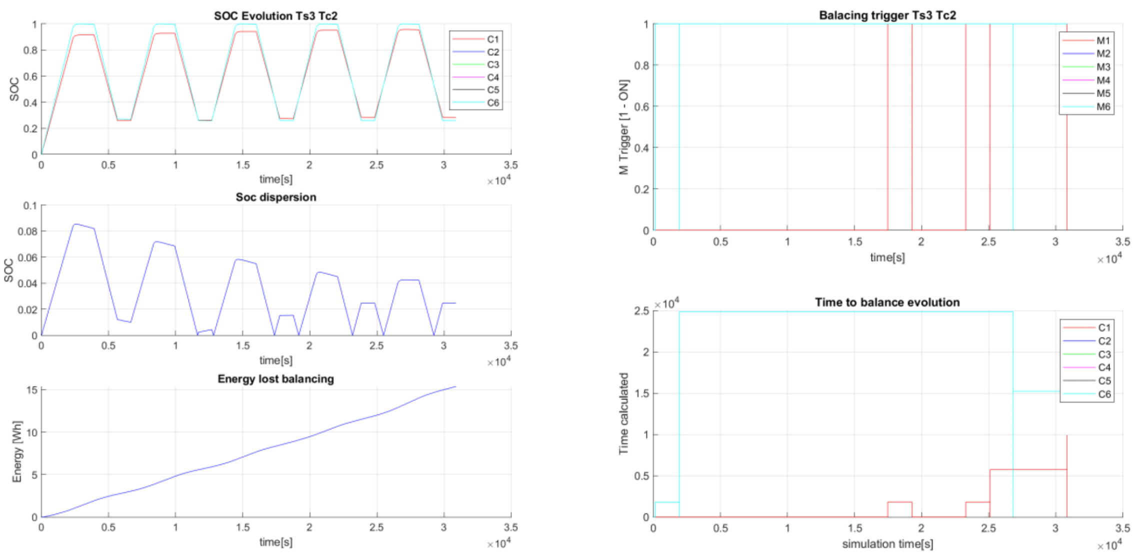


Figure 116. Capacity aging test. Left: SOC evolution. Right: Balancing action.

SOC History: Capacity -10%

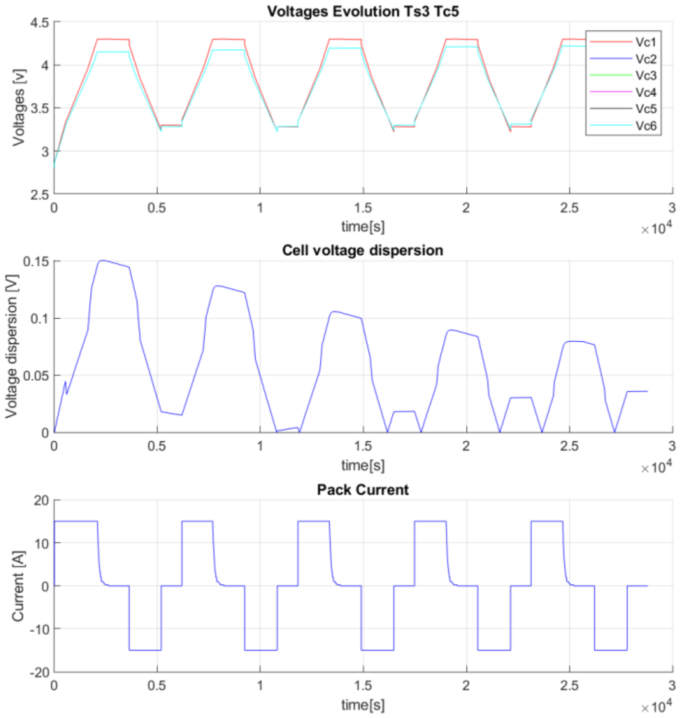


Figure 117. Capacity aging test. Voltage figure.

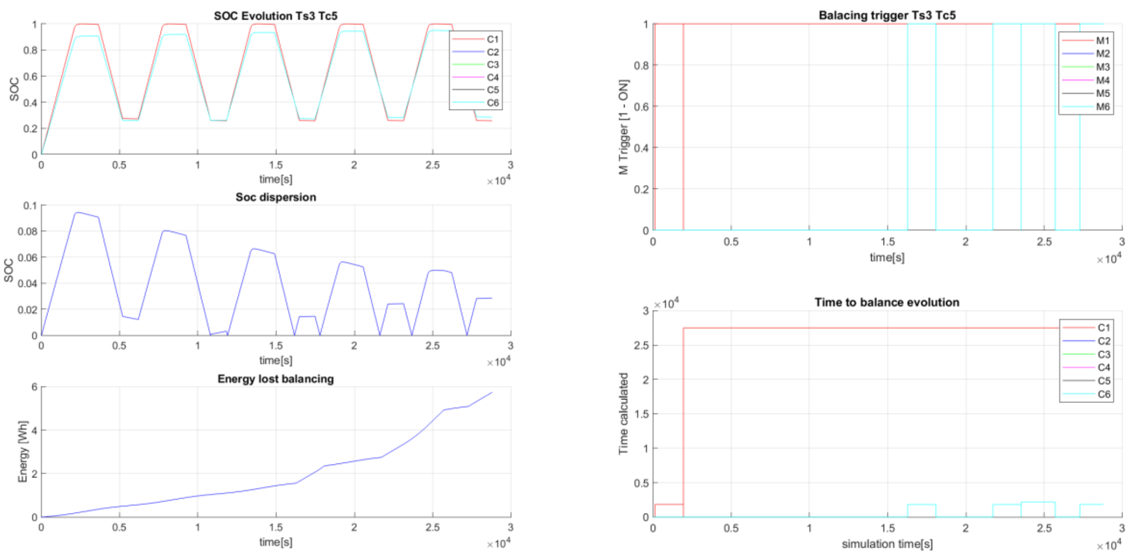


Figure 118. Capacity aging test. Left: SOC evolution. Right: Balancing action.

SOC History: VOC-SOC Table Change

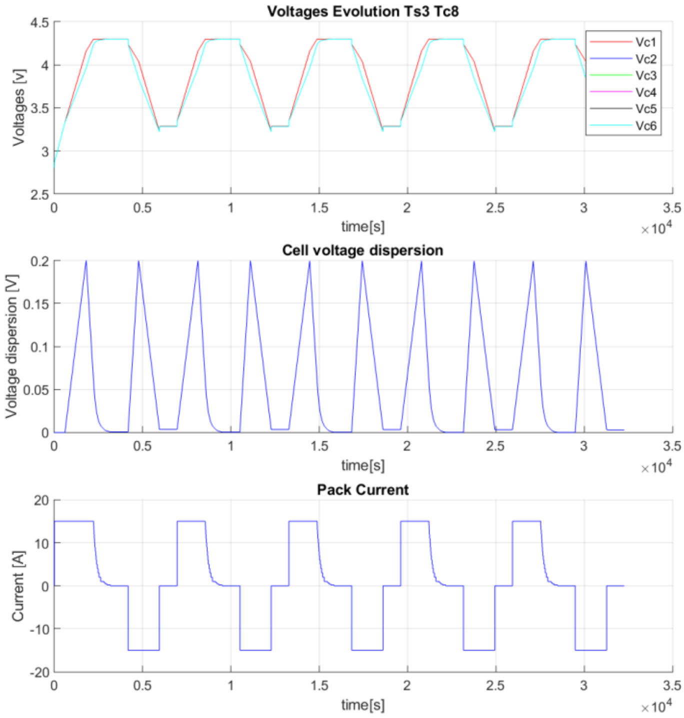


Figure 119. Change on VOC-SOC table test. Voltage figure.

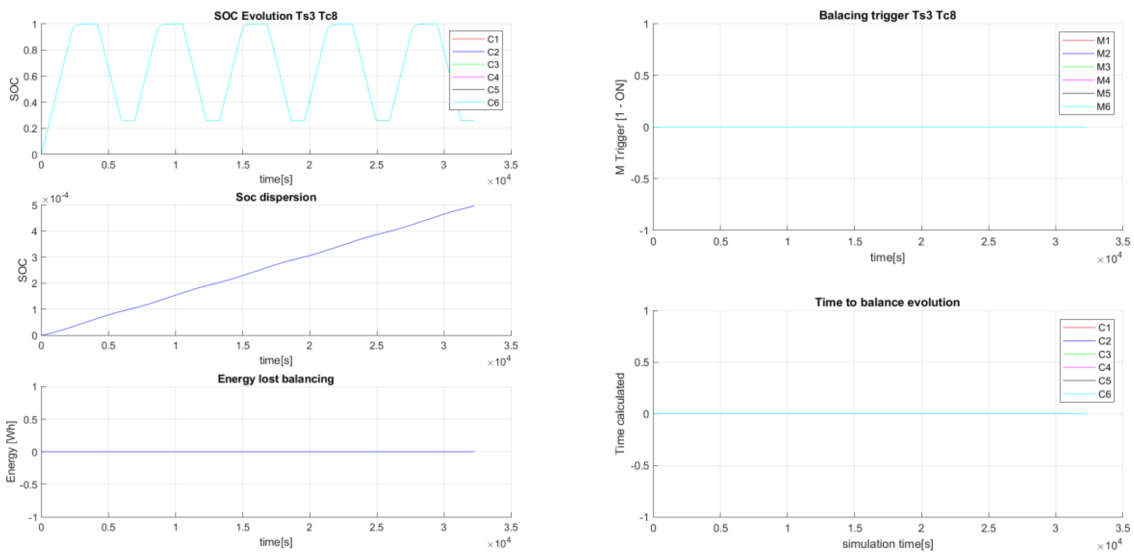


Figure 120. Change on VOC-SOC table test. Left: SOC evolution. Right: Balancing action.

4.4.3.3 SOC History: Error introduction

SOC History: Cell Voltage measurement offset, +50 mV

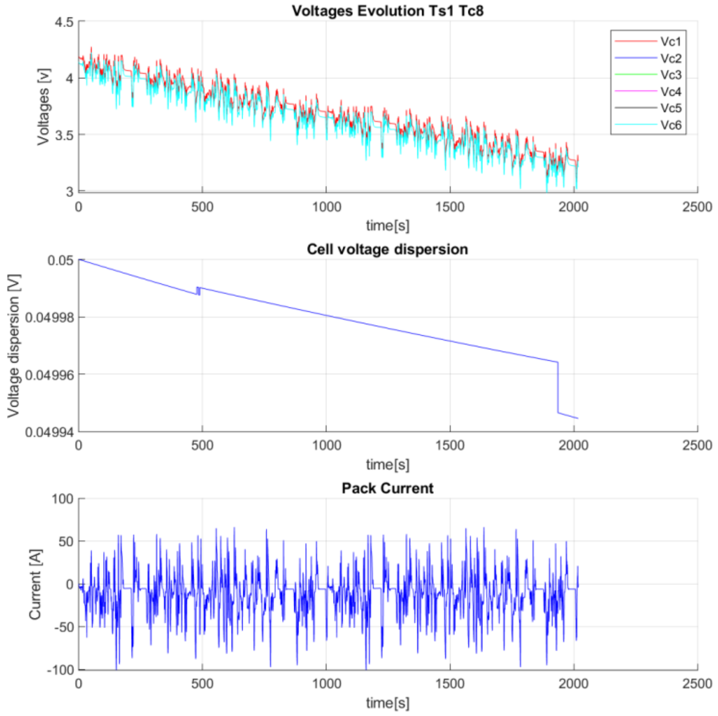


Figure 121. Cell voltage error introduction test. Voltage figure.

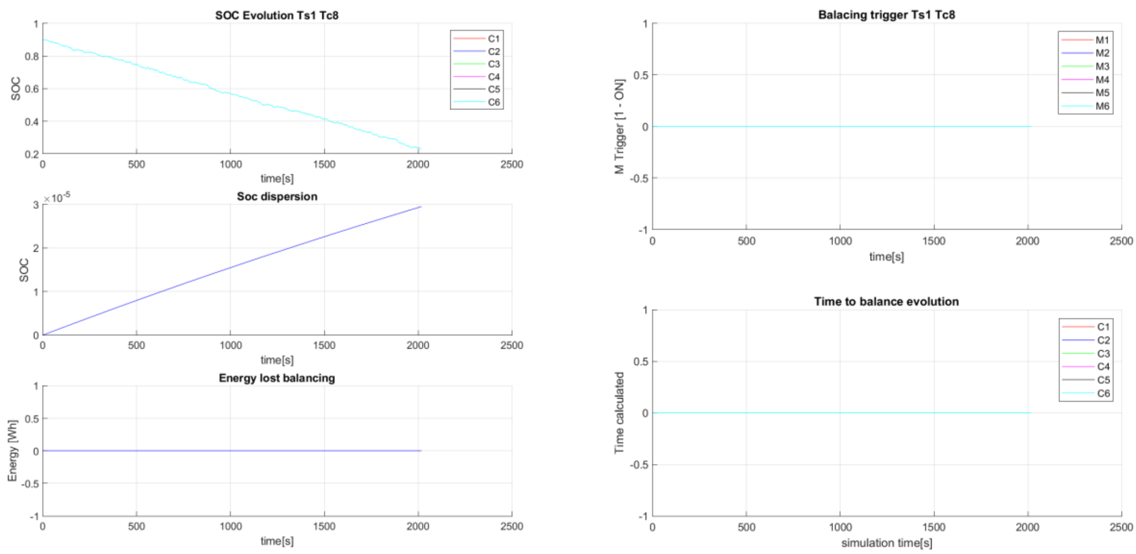


Figure 122. Cell voltage error introduction test. Left: SOC evolution. Right: Balancing action.

SOC History: Cell Voltage measurement offset, -50 mV

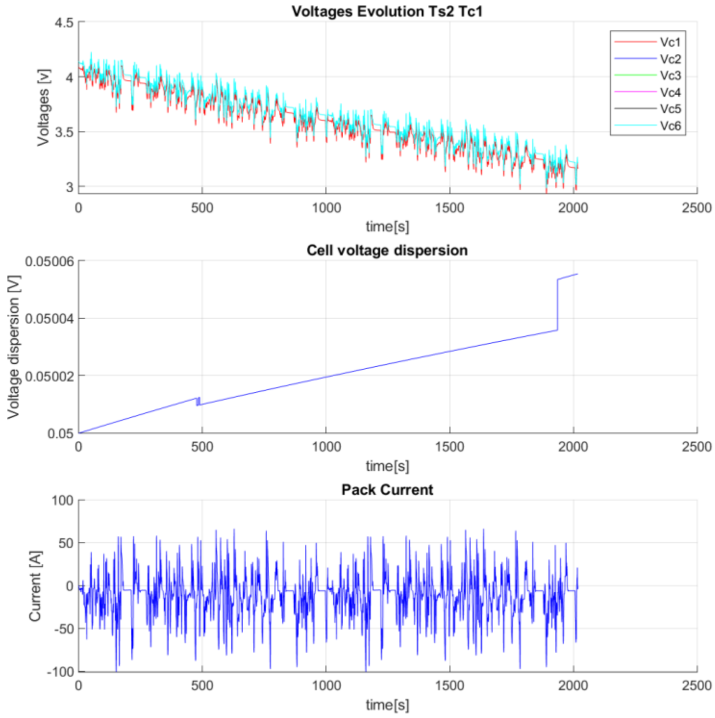


Figure 123. Cell voltage error introduction test. Voltage figure.

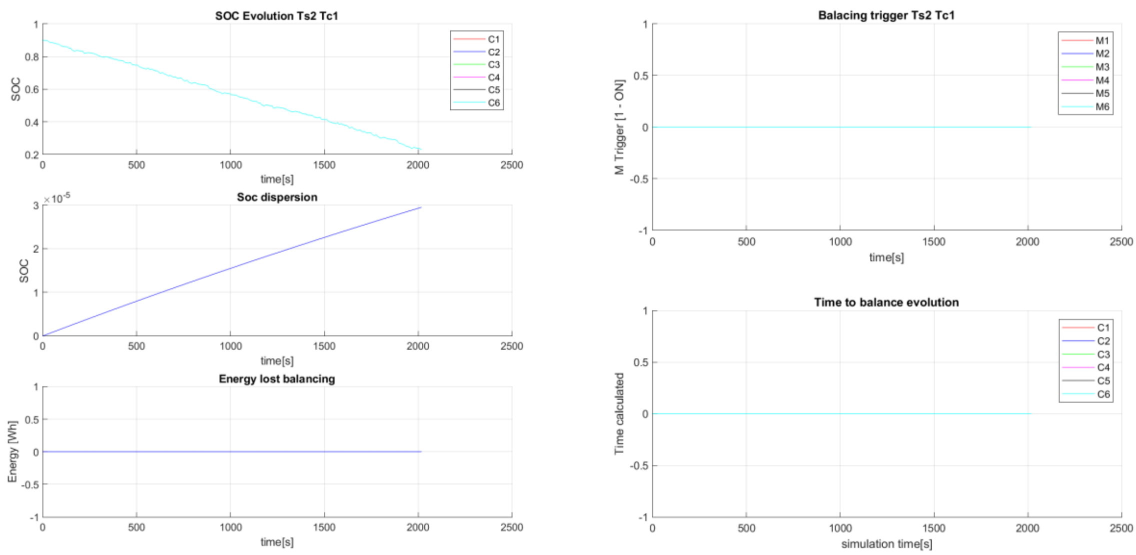


Figure 124. Cell voltage error introduction test. Left: SOC evolution. Right: Balancing action.

SOC History: SOC estimation offset, +2%

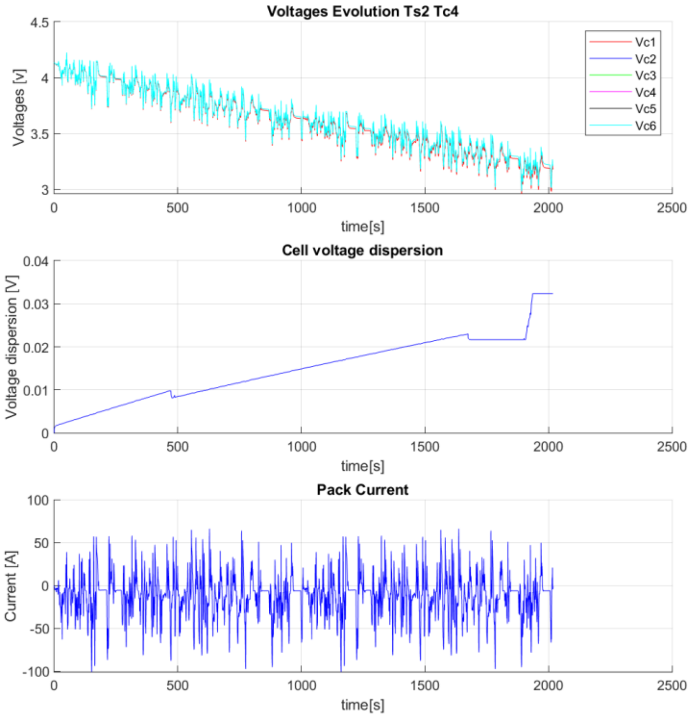


Figure 125. SOC estimation error introduction test. Voltage figure.

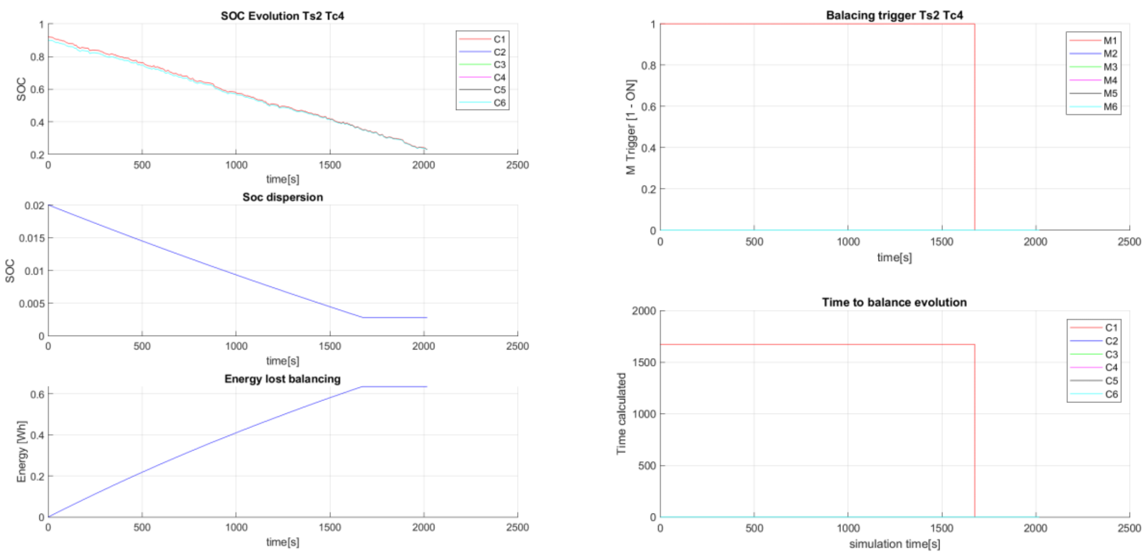


Figure 126. SOC estimation error introduction test. Left: SOC evolution. Right: Balancing action.

SOC History: SOC estimation offset, - 2%

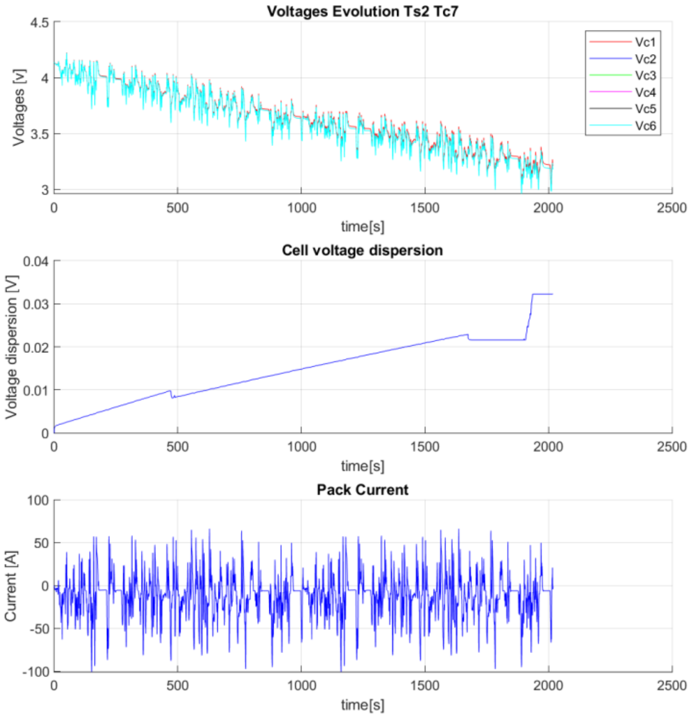


Figure 127. SOC estimation error introduction test. Voltage figure.

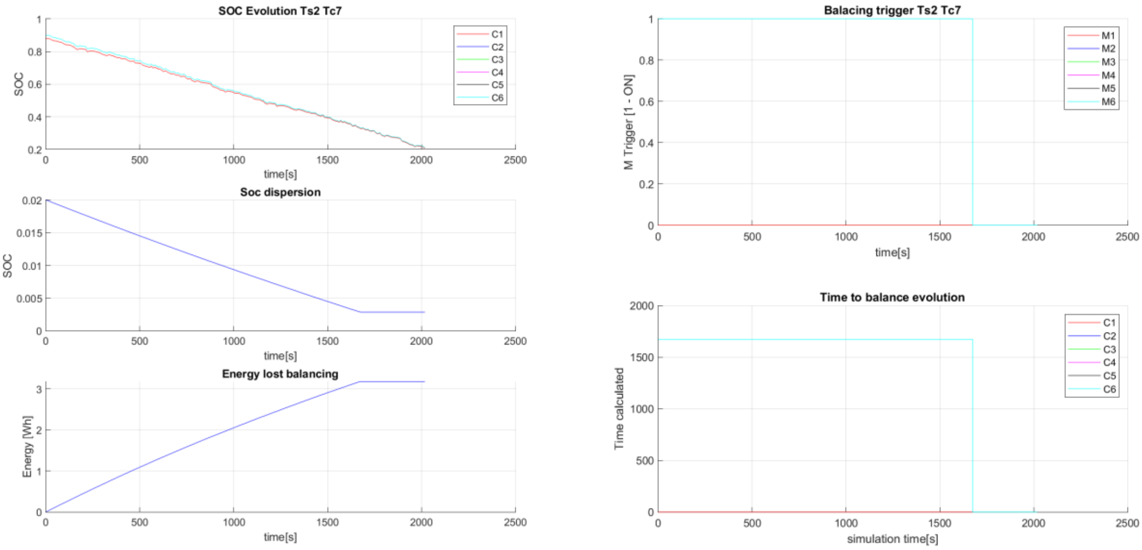


Figure 128. SOC estimation error introduction test. Left: SOC evolution. Right: Balancing action.

SOC History: Current sense offset, +250 mA

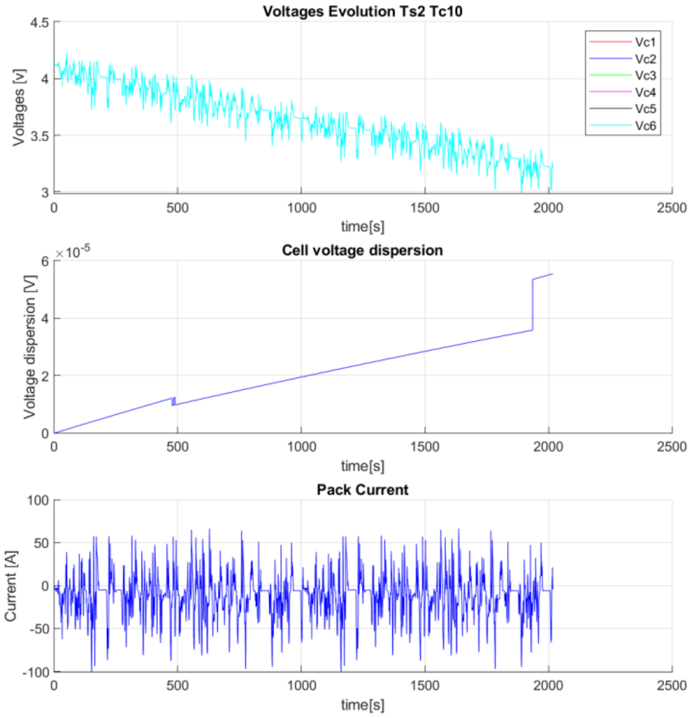


Figure 129. Current sense error introduction test. Voltage figure.

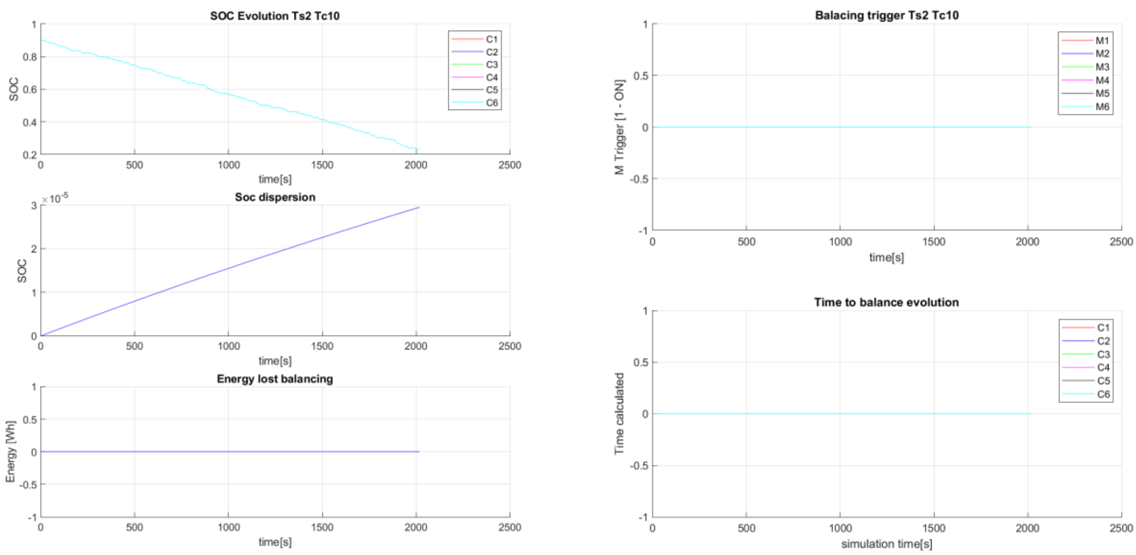


Figure 130. Current sense error introduction test. Left: SOC evolution. Right: Balancing action.

SOC History: Current sense offset, - 250 mA

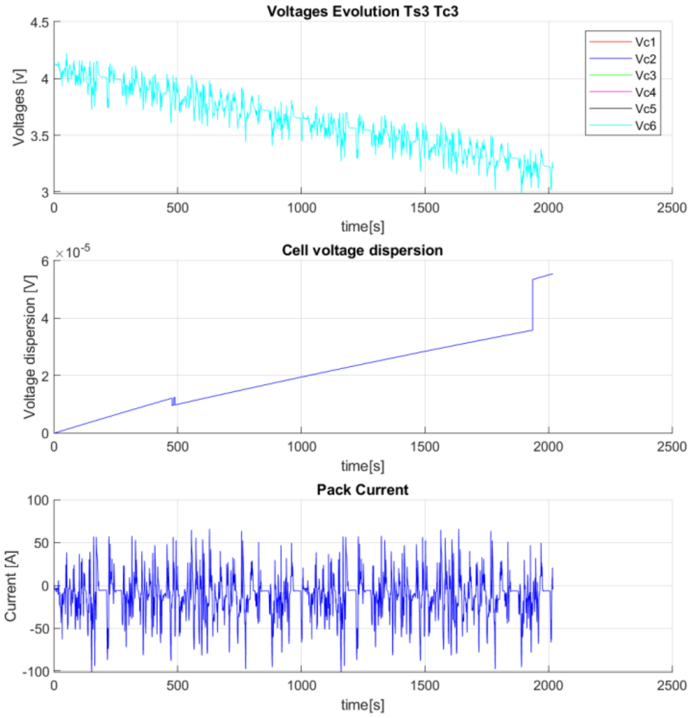


Figure 131. Current sense error introduction test. Voltage figure.

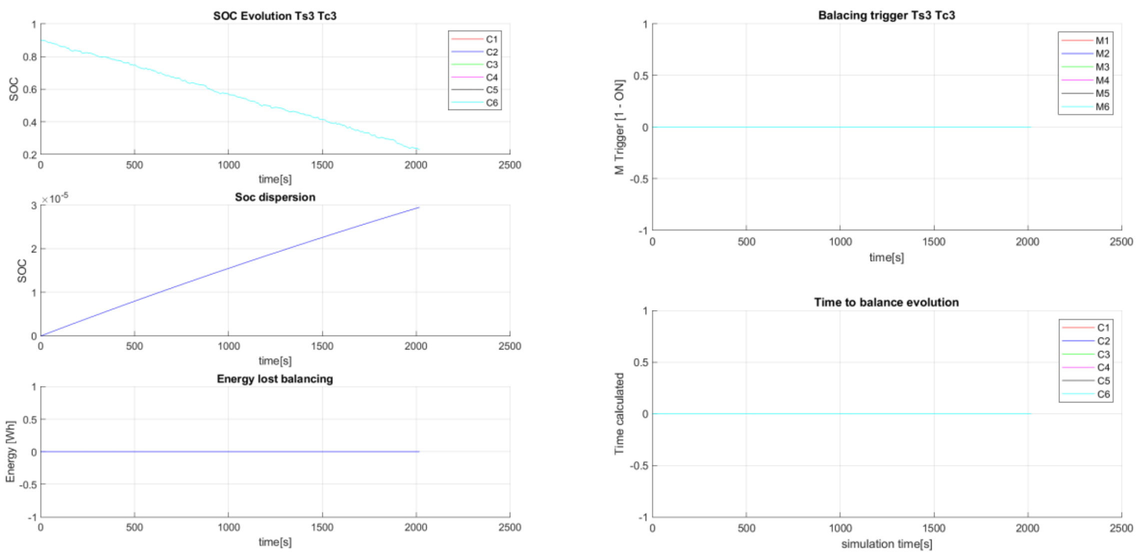


Figure 132. Current sense error introduction test. Left: SOC evolution. Right: Balancing action.

4.4.4 Comparison Between Dissipative Techniques

After the different sets of tests, a series of conclusions can be made. These conclusions will compare the different dissipative techniques used on the different test scenarios (functional, aging and Error introduction).

Functional wise:

- The three algorithms (ΔVOC, FV: ΔVOC, FV: Pulsed Charge, SOC History) have been proven to be fully functional and able to balance the battery cells. ΔVOC and SOC History have the advantage to work under operation.

  On the other hand, FV: ΔVOC, and FV: Pulsed Charge only work in certain conditions. Specially, the behaviour in the case of FV: Pulsed Charge is dependent to the battery condition. On Figure 95, only one pulsed charge spike is performed on the simulation cycle. The SOC difference is reduced but it's not highly improved.

Functional – Balancing Capability:

- For evaluating the balancing capability of the different systems, the plot of energy lost on balancing will be checked for the different algorithms. This plot will be checked on the functional tests with 5% SOC difference.

  These algorithms have the same hardware implementation with the same balancing resistor. However, the control part is what differentiates them one from the other.

  Table 16. Comparison of energy balanced per model.

	Energy Extracted	Figure (Left plot)
ΔVOC	1.3 Wh	Figure 61
FV: ΔVOC	0.24 Wh	Figure 92
FV: Pulsed Cha	0.07 Wh	Figure 96
SOC History	1.6 Wh	Figure 104

  As can be seen, FV algorithms provide poor performance. On the other hand, ΔVOC and SOC History provide more balancing capability as they operate in all conditions.

  The difference between ΔVOC and SOC History is due to the thresholds applied (0.5% on SOC, 0.025 V on ΔVOC).

Aging*:*

The algorithm behaviours must be analysed for each of the balancing methods.

- ΔVOC:
  - Internal resistance: The algorithm calculates correctly the new internal resistance value and balances the different cells. The algorithm performs correctly in this scenario.
  - Capacity: The introduction of a capacity difference between cells causes unbalance on the cell assembly after just the first cycle (Figure 75). The algorithm from then, as there's a difference in voltage, starts actuating. It is interesting to notice how it does no actuate when the battery is in its lower SOC boundary. In that case the voltage difference is small. Overall, the behaviour is correct.
  - VOC-SOC Change: In this case it's important to remark that the change on the VOC-SOC table does not mean a change on the capacity of the different cells. As the cell voltage is different due to the change on the table, the algorithm starts actuating

(Figure 78). This actuation causes that the cell that is discharged, in this case cell 1, loses SOC. As can be seen, the balancing action is stopped on the first cycle as the voltage is the same for both cells at the end of the discharge process. From then, there's a SOC difference that causes a voltage difference between cells. On the last cycles, it can be observed that the other cells start balancing on the ending of the discharge. The behaviour of the algorithm is incorrect.

  - Final Voltage: Only the test with internal resistance change was performed. On the tests that present a change on capacity or on the VOC-SOC table, is easy to derivate what would be the long-term impact of that effect on the cells without the balancing system. However, regarding internal resistance, it was not obvious.

    On Figure 100, the SOC difference over the different cycles with the change on the internal resistance is kept on the same value. So, there would be no need to balance apart from the initial SOC difference introduced on purpose.

- SOC History:
  - Internal resistance: As the algorithm is based on SOC estimation, the change on resistance does not introduce any miss behaviour (Figure 112). The balancing action is performed until the threshold is fulfilled. The behaviour of the algorithm is correct.
  - Capacity: The difference on capacity value makes the cell that has the aging phenomena charge faster (Figure 118) or slower (Figure 116). This causes a SOC difference that makes the algorithm actuate to perform balancing. In this case the balancing capability is not "powerful" enough to extract the ΔCharge on the cells. The behaviour of the algorithm is correct.
  - VOC-SOC Change: The difference on the cell output voltage does not have a SOC change implication (Figure 120). No balancing action is performed accordingly. The behaviour of the algorithm is correct.

Error introduction:

As in the previous section, the algorithm behaviours must be analysed for each of the balancing methods.

- ΔVOC:
  - Cell measurement: As the value introduced on the measurement side is above the balancing threshold, balancing is started on the cell that has the voltage offset (Figure 79). The introduction of balancing with no need will lead to a SOC difference between cells. The algorithm is **susceptible** to errors on voltage measurement.
  - SOC estimation: As the algorithm does not rely on SOC estimation for any of its calculation parameters, no balancing action is miss performed (Figure 83). The algorithm is **immune** to errors on SOC estimation.
  - Current offset: The algorithm relies on current measurement to determine the internal resistance for the ΔVOC calculation. However, as the balancing is performed by comparing the different cells, and the current measured affects all cells calculations, no balancing action is miss performed (Figure 88). The algorithm is **immune** to errors on current measurement.
- SOC History:

- Cell measurement: The algorithm does not rely on cell voltage measurement to perform balancing. No miss operation on the balancing side is performed (Figure 122). The algorithm is **immune** to errors on the voltage measurement.
- SOC estimation: The algorithm relies on SOC estimation to perform balancing. As the offset introduced is above the threshold, balancing action is performed (Figure 126). The cells that are physically not unbalanced that the algorithm choses to discharge, will start to have voltage differences with the unbalanced ones. This voltage differences will cause a loss of performance on charging and discharging. The algorithm is **susceptible** to errors on SOC estimation.
- Current offset: The algorithm does not rely on current measurement to perform balancing (Figure 130). No miss operation on the balancing side is performed. The algorithm is **immune** to errors on voltage measurement.

Final conclusions:

Each of these balancing methods have its advantages and drawbacks seen on previous points and plots.

Final Voltage algorithms require less effort in terms of development. They could be used in a cheap application that is known that the supply can be connected for a long time. For example, for stationary battery banks that are always connected to the grid. In a vehicle application I consider that these algorithms are discarded. There could be the scenario in which the battery never reaches the top SOC, or even that when it reaches it, the supply is immediately removed.

ΔVOC algorithm requires more development effort than final voltage algorithms. It permits continuous operation. It has a theoretical basis that requires a calculation parameter of the cell (R_{int}). On this calculation the error on the voltage measurement is introduced to the algorithm. The main weak point of the ΔVOC relies on the OCV as an indicator of the status of the cell. This, if the cell due to operation does not maintain stable its VOC-SOC curve (phenomena that can happen on a real application), can introduce unnecessary balancing action. I consider that the application of this algorithm could be adopted into a real vehicle application.

SOC History algorithm is the one that would require more development effort as SOC estimation is a complex algorithm. However, once the SOC estimation algorithm is precise, it makes the algorithm immune to errors on the measurement of physical variables. It is the one that would provide more robustness in terms of balancing in a real EV application.

4.4.5 SSC

4.4.5.1 SSC: Functional tests

SSC: Cycle test, Initial SOC difference of 5%

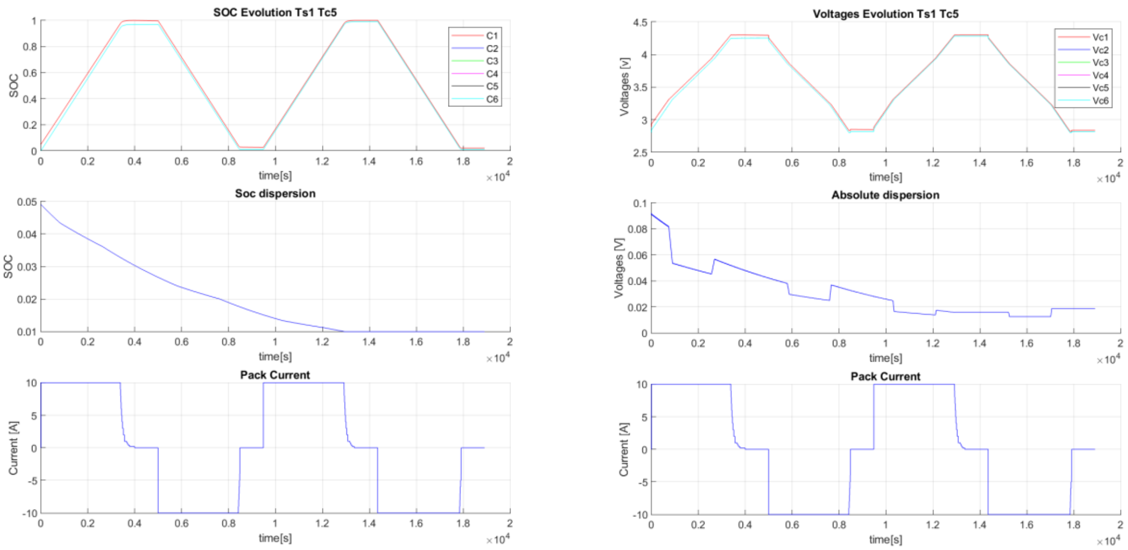


Figure 133. Functional test. Left: SOC evolution. Right: Voltage evolution.

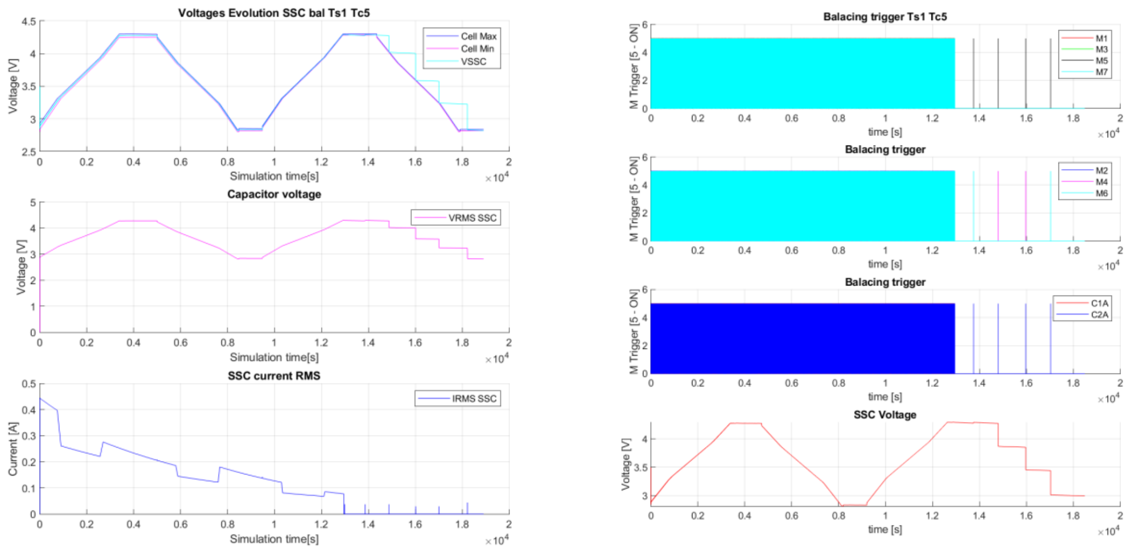


Figure 134. Functional test. Left: SSC current and voltage evolution. Right: Switches action.

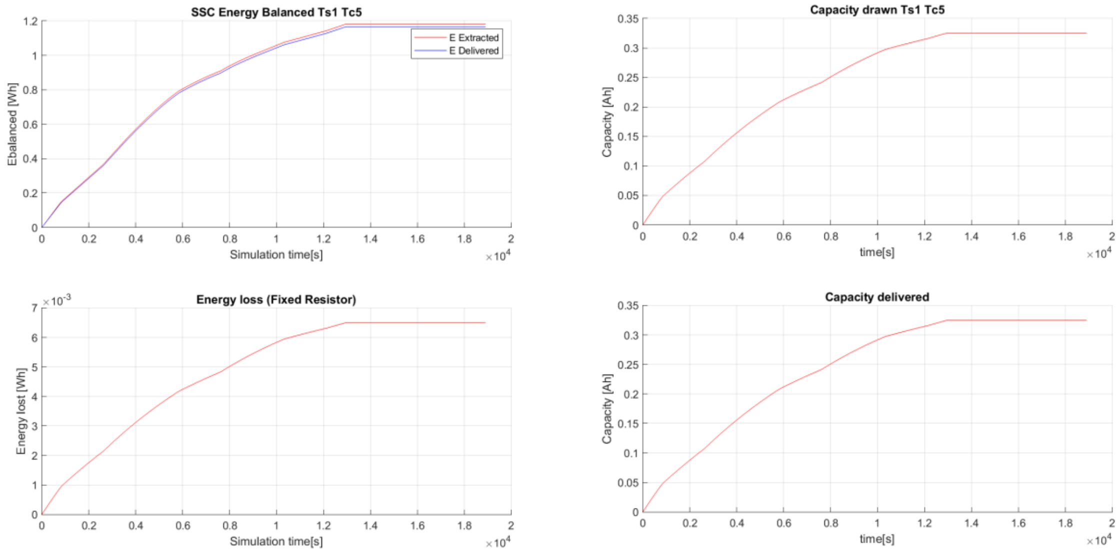


Figure 135. Functional test. Left: Energy parameters. Right: Capacity parameters.

SSC: Cycle test, Initial SOC - 0% 3% 4% 5% 6% 1%

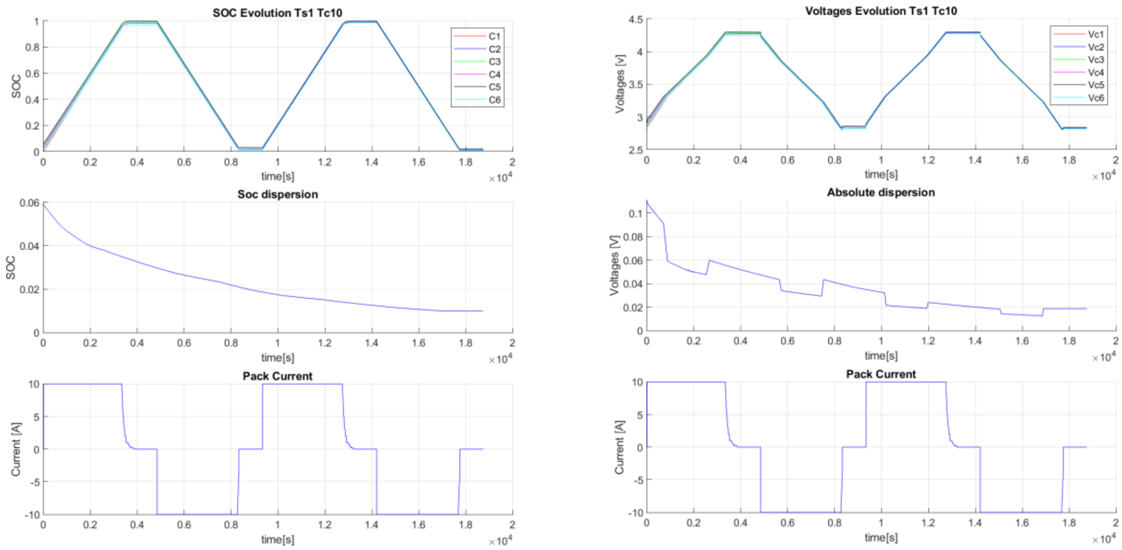


Figure 136. Functional test. Left: SOC evolution. Right: Voltage evolution.

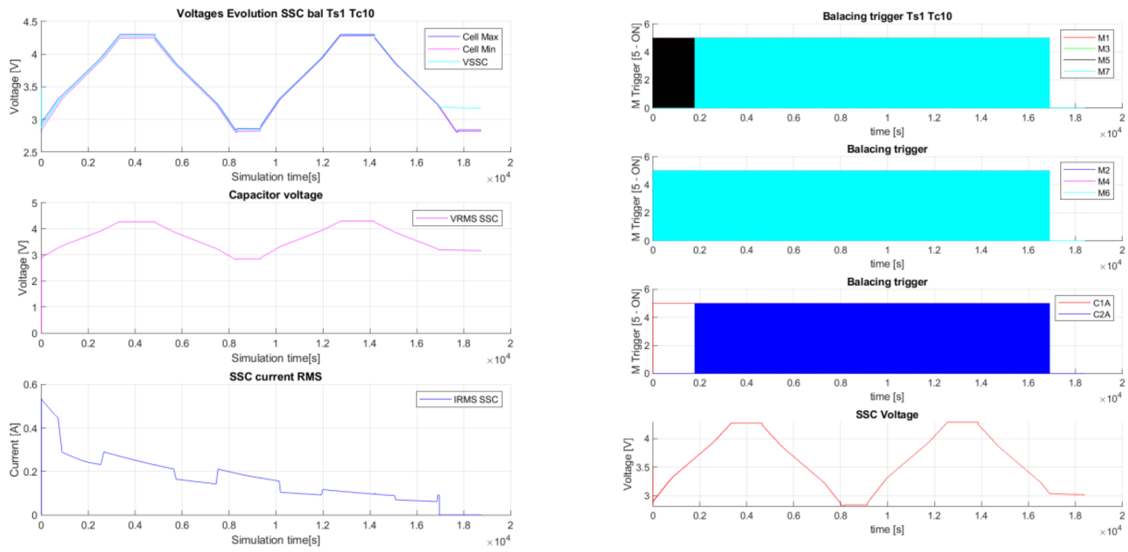


Figure 137. Functional test. Left: SSC current and voltage evolution. Right: Switches action.

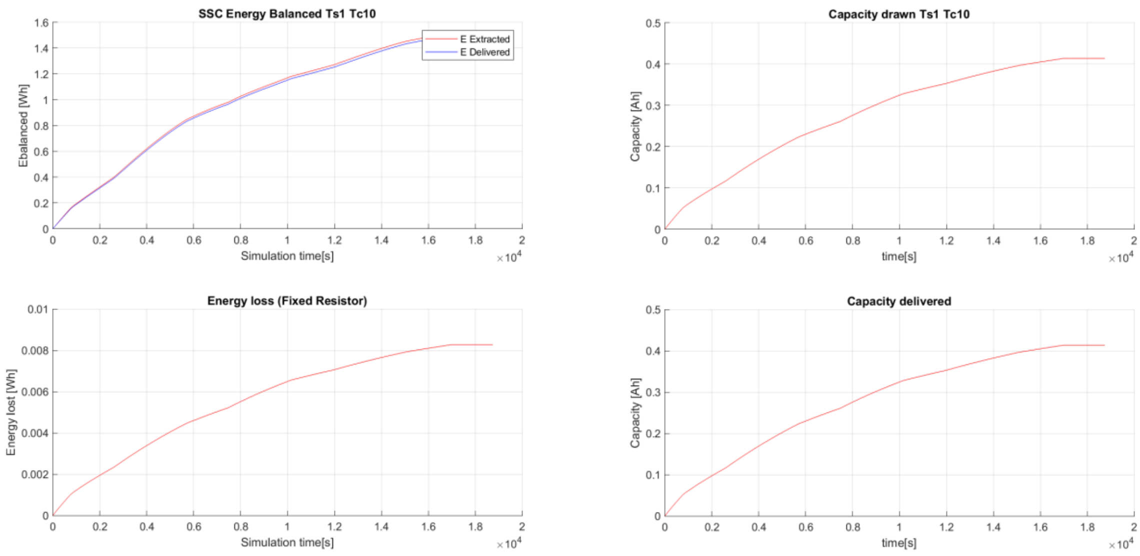


Figure 138. Functional test. Left: Energy parameters. Right: Capacity parameters.

SSC: Profile test, Initial SOC difference of 10%

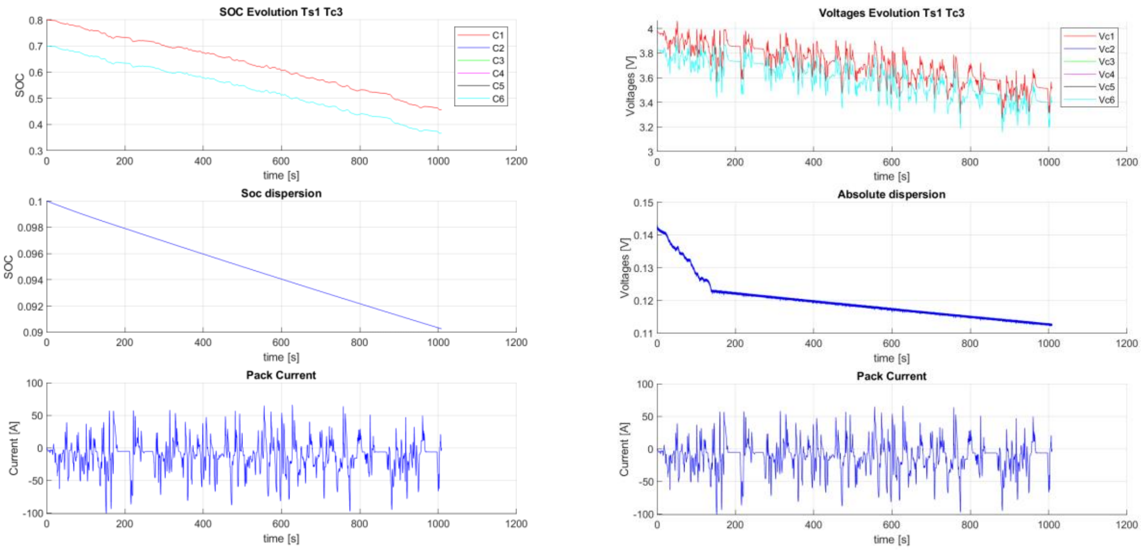


Figure 139. Functional test. Left: SOC evolution. Right: Voltage evolution.

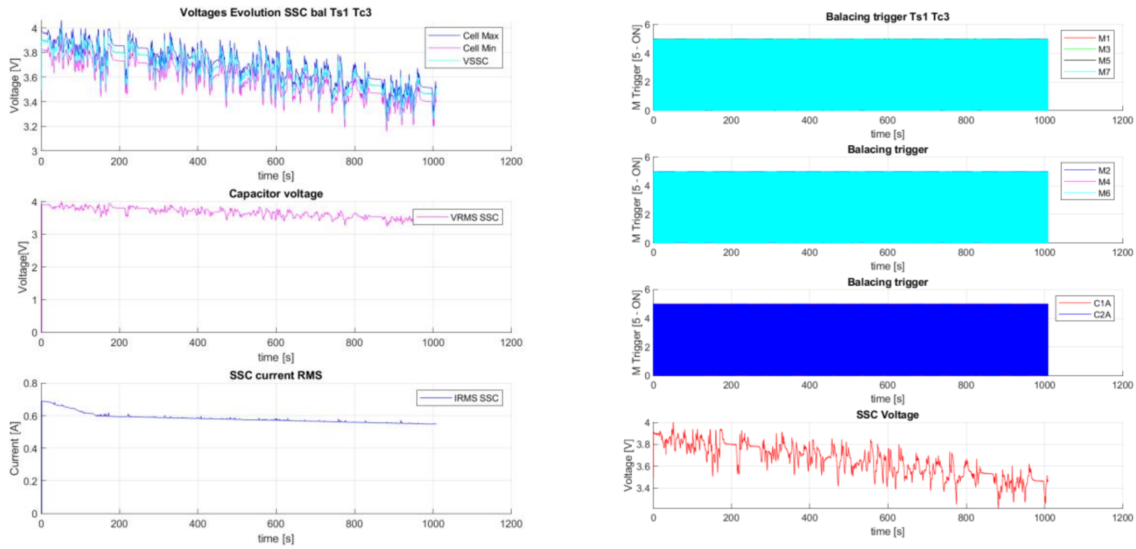


Figure 140. Functional test. Left: SSC current and voltage evolution. Right: Switches action.

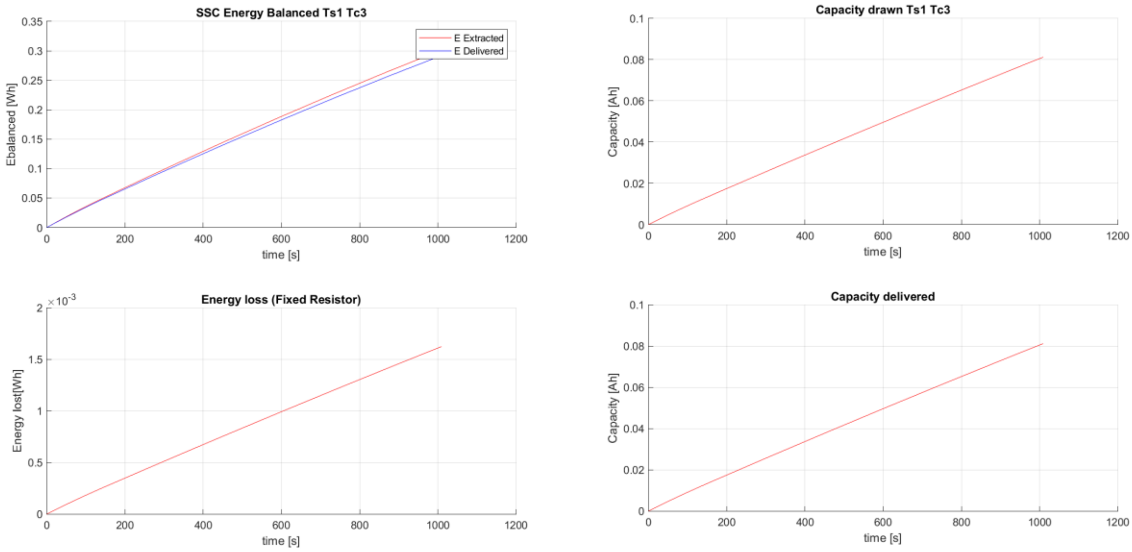


Figure 141. Functional test. Left: Energy parameters. Right: Capacity parameters.

SSC: Profile test, Initial SOC - 0% 3% 4% 5% 6% 1%

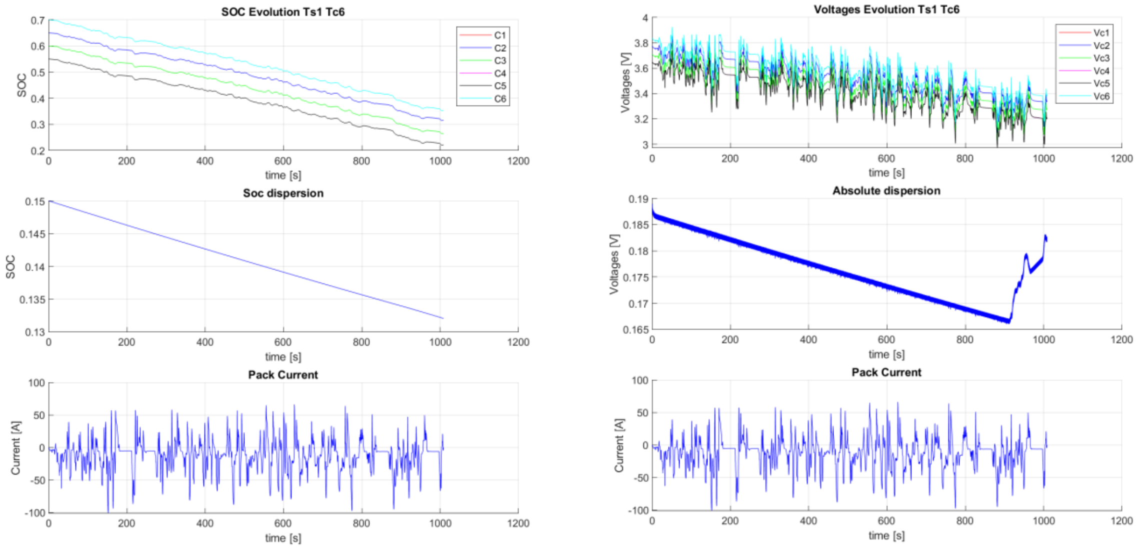


Figure 142. Functional test. Left: SOC evolution. Right: Voltage evolution.

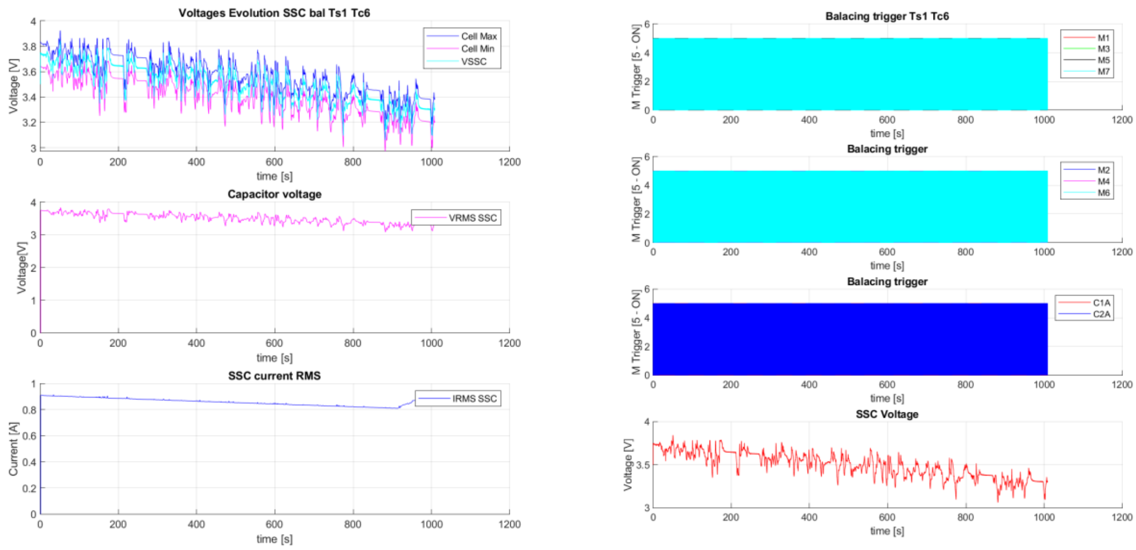


Figure 143. Functional test. Left: SSC current and voltage evolution. Right: Switches action.

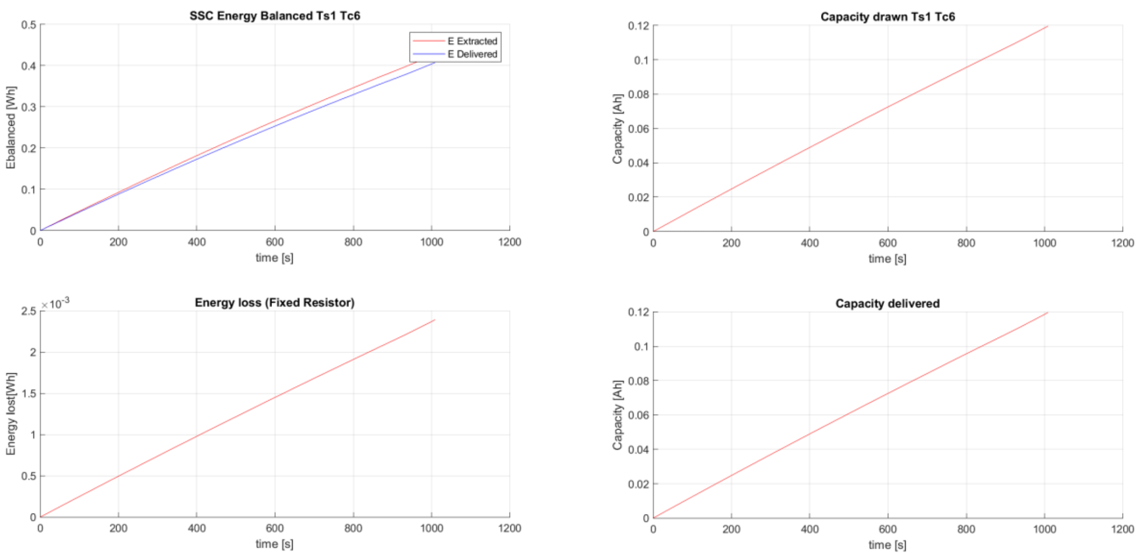


Figure 144. Functional test. Left: Energy parameters. Right: Capacity parameters.

4.4.5.2 SSC: Aging

SSC: Internal resistance +50 %, Initial SOC difference of 5%

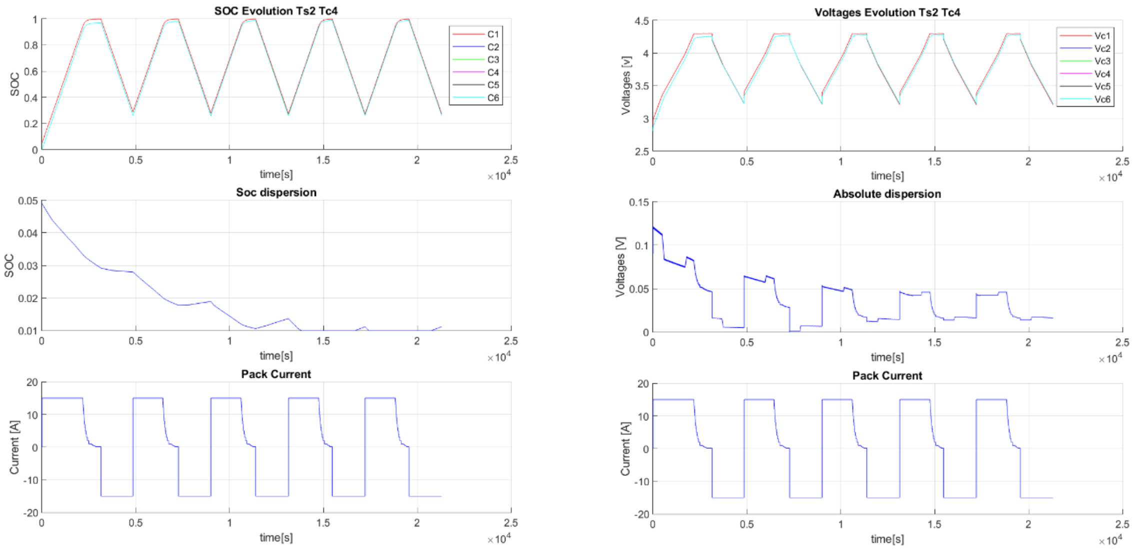


Figure 145. Internal resistance aging test. Left: SOC evolution. Right: Voltage evolution.

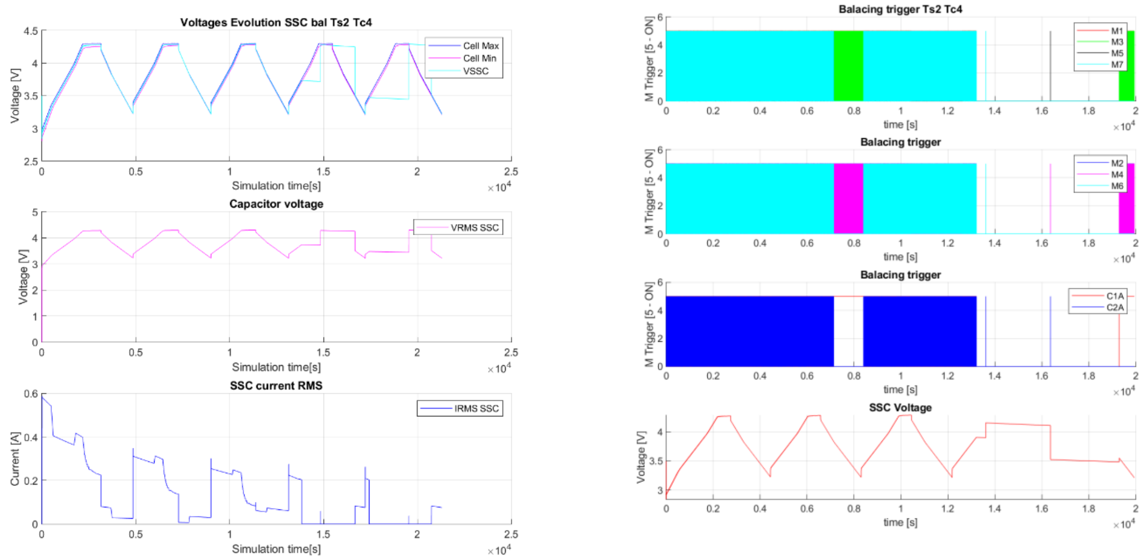


Figure 146. Internal resistance aging test. Left: SSC current and voltage evolution. Right: Switches action.

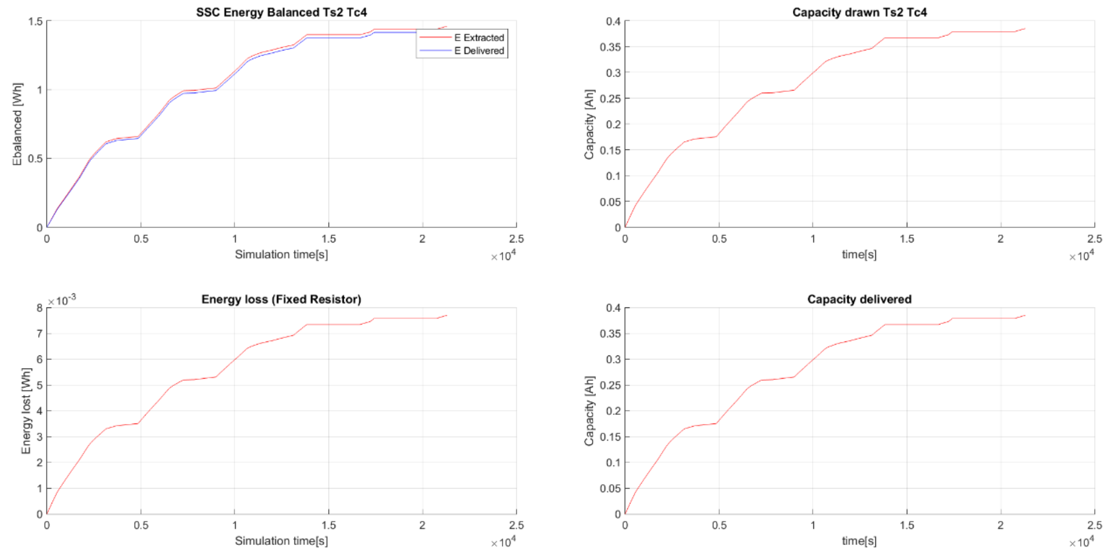


Figure 147. Internal resistance aging test. Left: Energy parameters. Right: Capacity parameters.

SSC: Internal resistance -50 %, Initial SOC difference of 5%

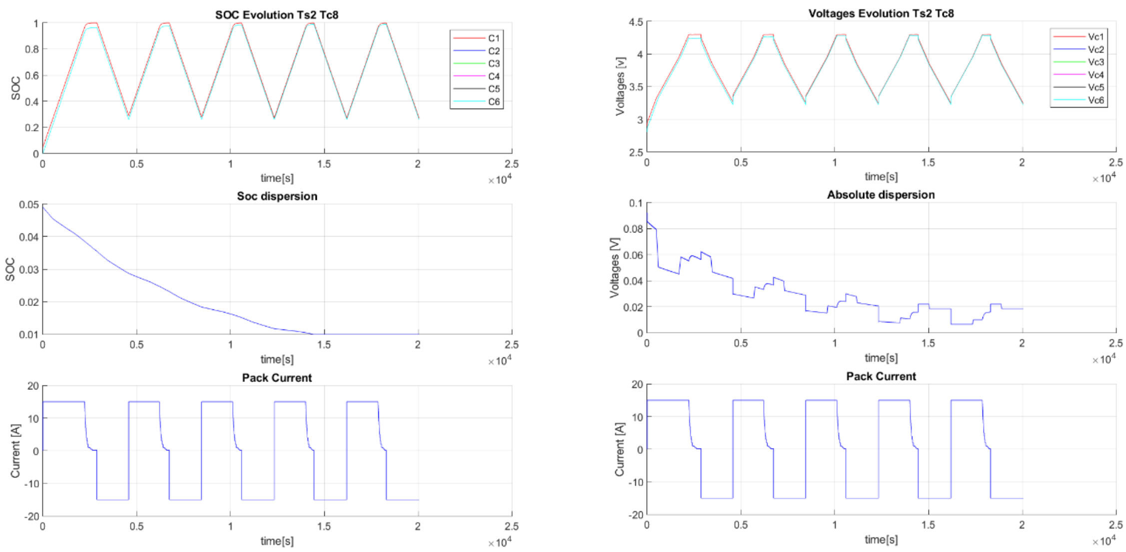


Figure 148. Internal resistance aging test. Left: SOC evolution. Right: Voltage evolution.

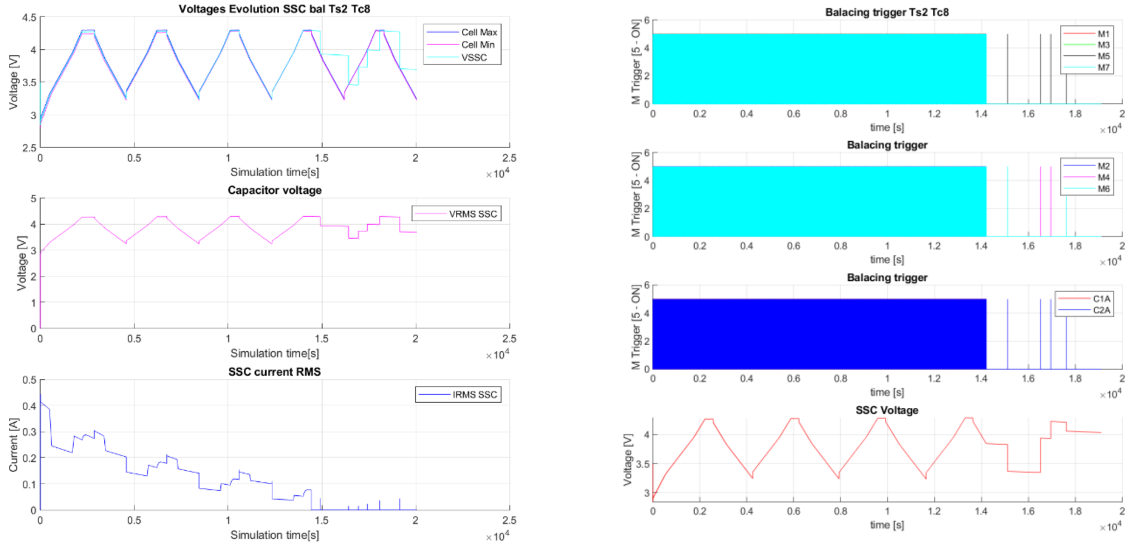


Figure 150. Left: SSC current and voltage evolution. Right: Switches action.

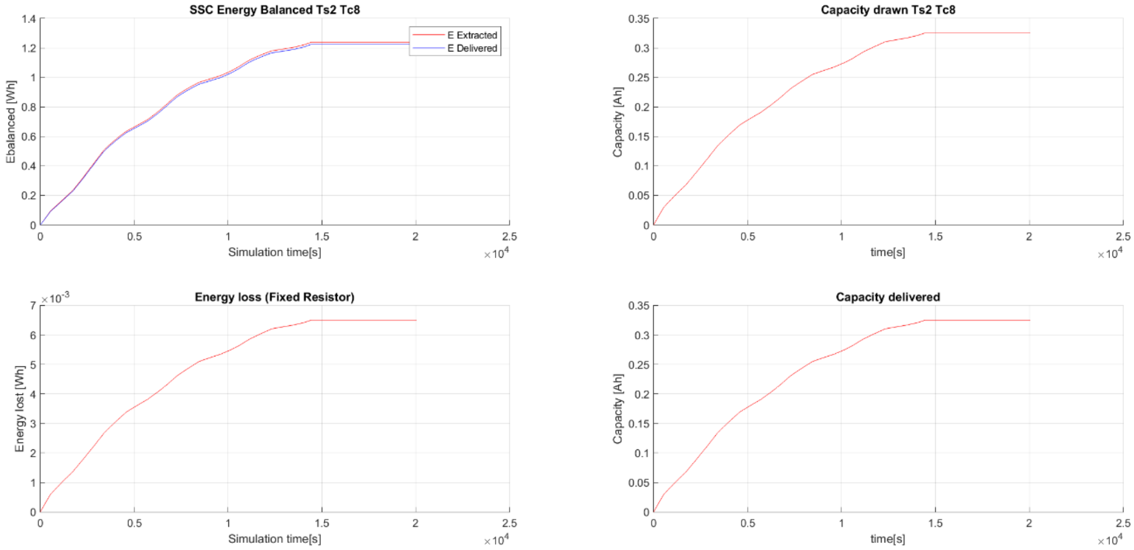


Figure 149. Internal resistance aging test. Left: Energy parameters. Right: Capacity parameters.

SSC: Capacity +10%

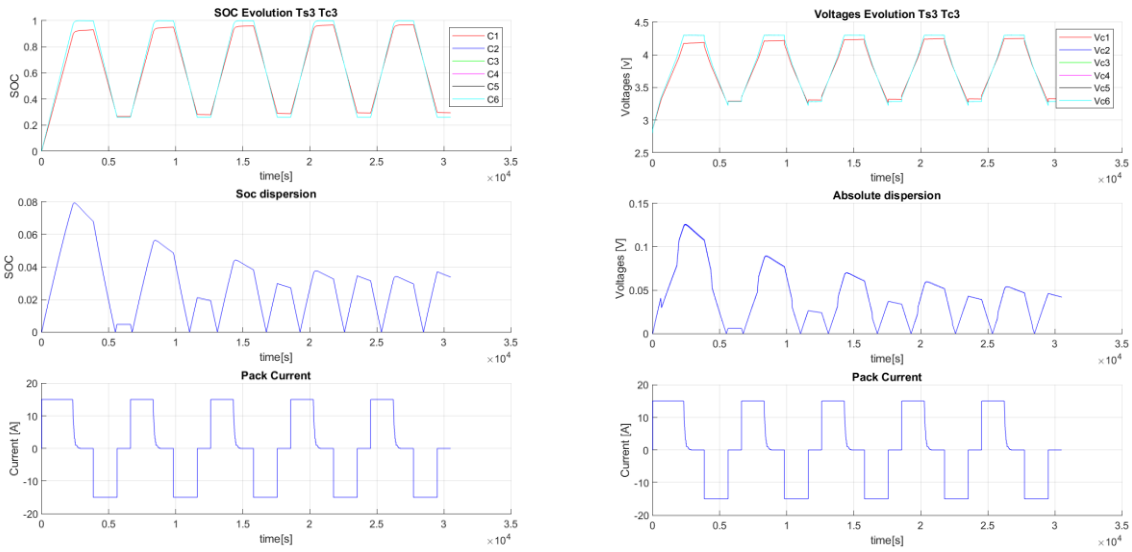


Figure 151. Capacity aging test. Left: SOC evolution. Right: Voltage evolution.

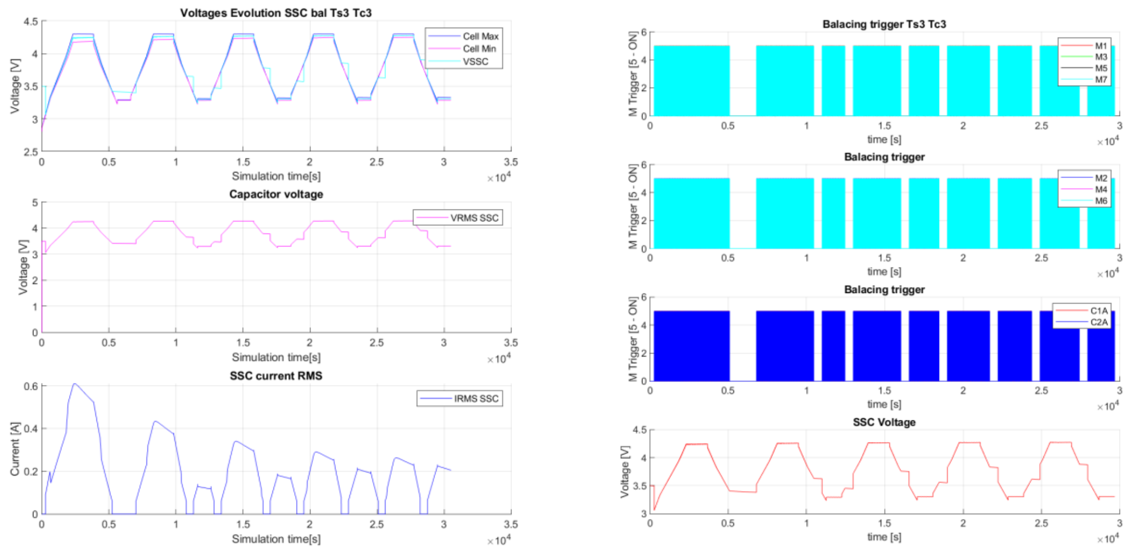


Figure 152. Capacity aging test. Left: SSC current and voltage evolution. Right: Switches action.

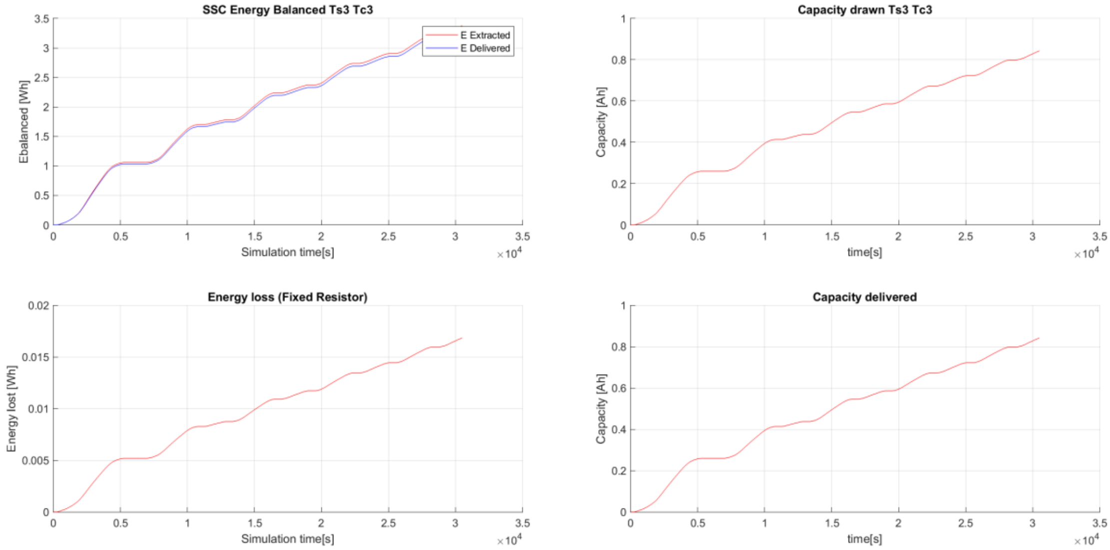


Figure 153. Capacity aging test. Left: Energy parameters. Right: Capacity parameters.

SSC: Capacity -10%

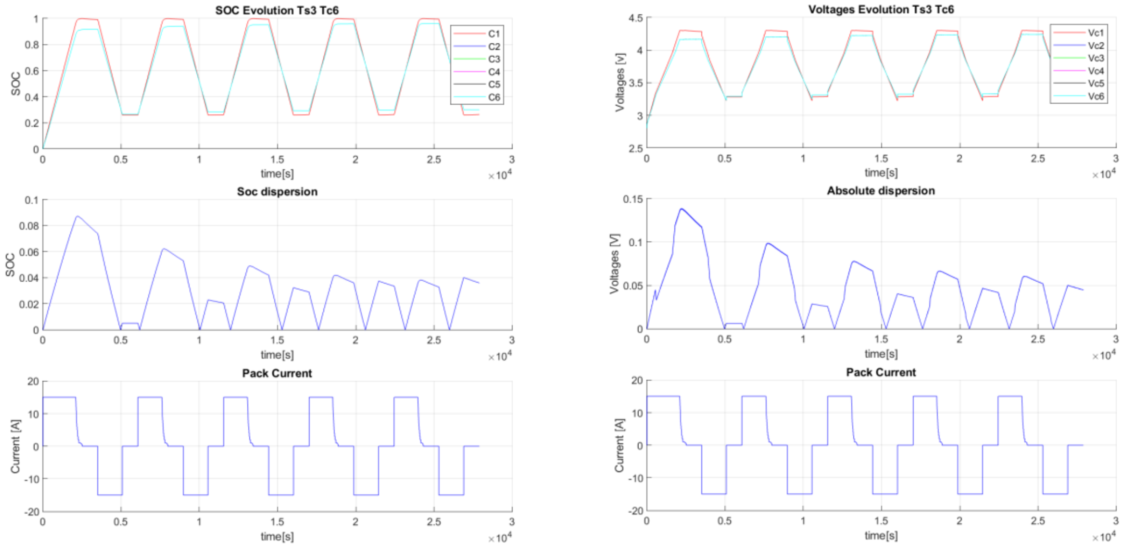


Figure 154. Capacity aging test. Left: SOC evolution. Right: Voltage evolution.

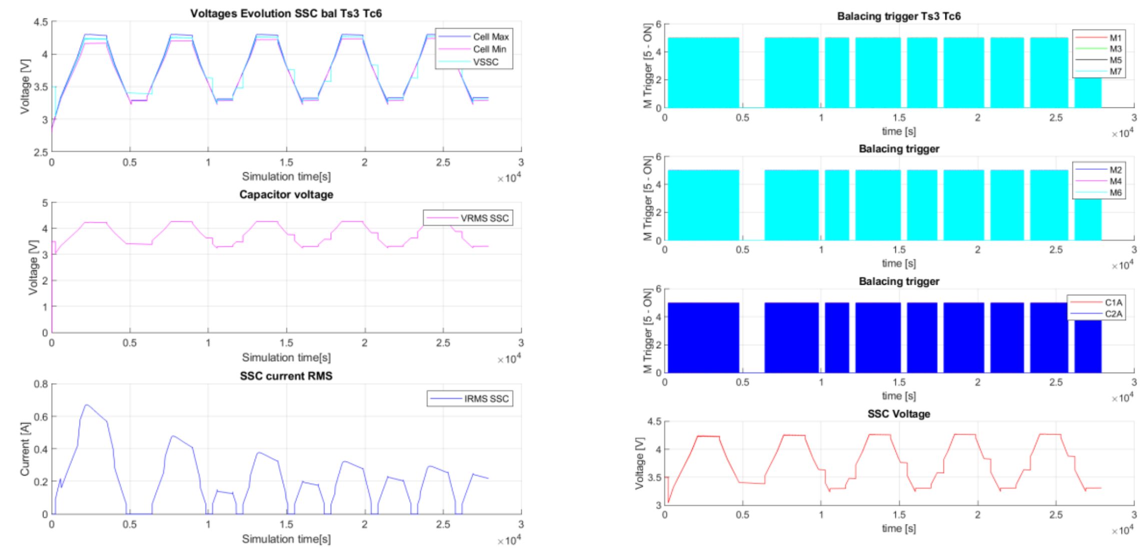


Figure 155. Capacity aging test. Left: SSC current and voltage evolution. Right: Switches action.

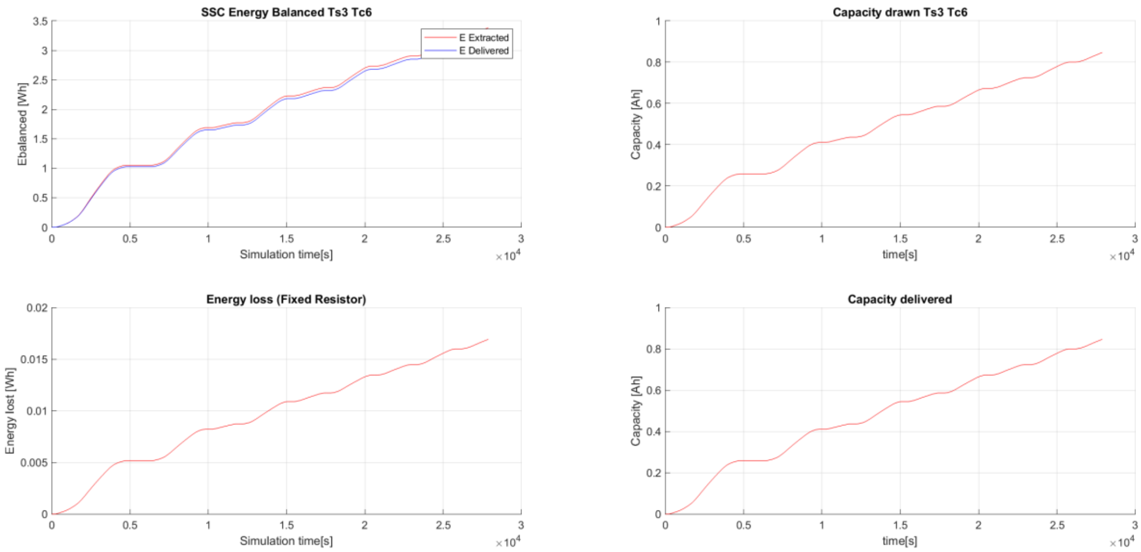


Figure 156. Capacity aging test. Left: Energy parameters. Right: Capacity parameters.

SSC: VOC – SOC Table Change

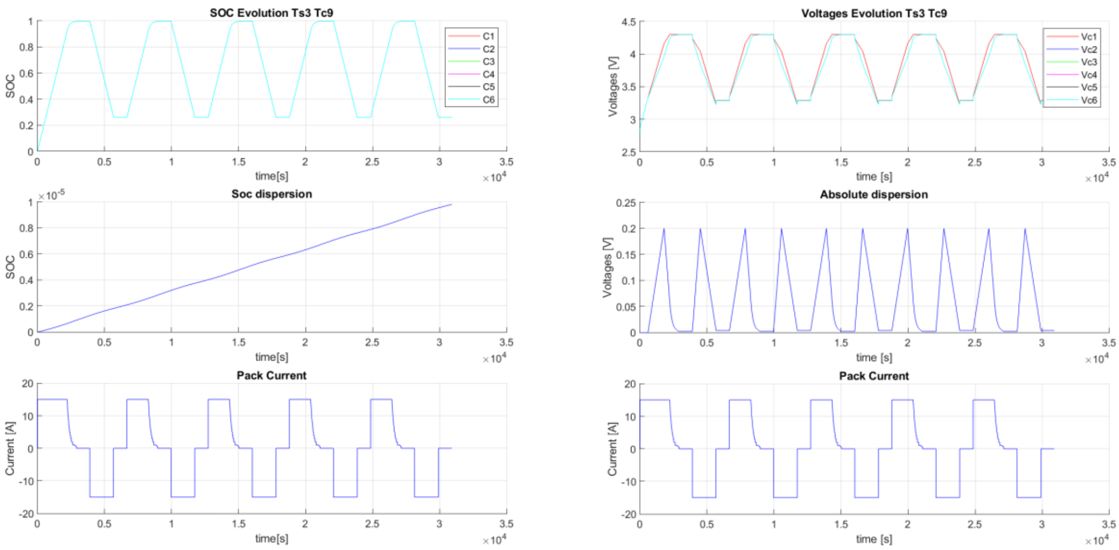


Figure 157. VOC-SOC table change test. Left: SOC evolution. Right: Voltage evolution.

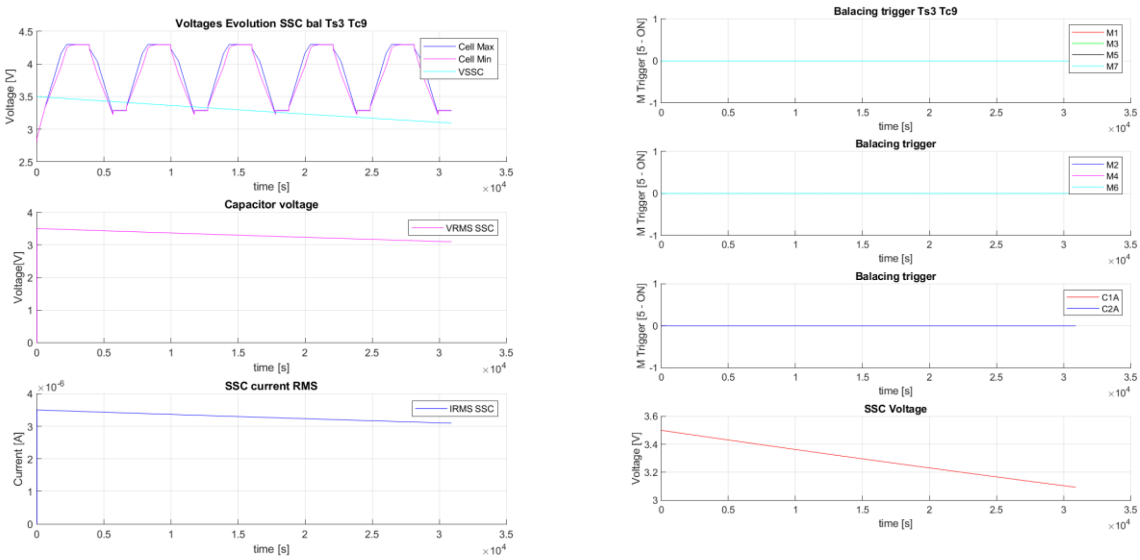


Figure 158. VOC-SOC table change test. Left: SSC current and voltage evolution. Right: Switches action.

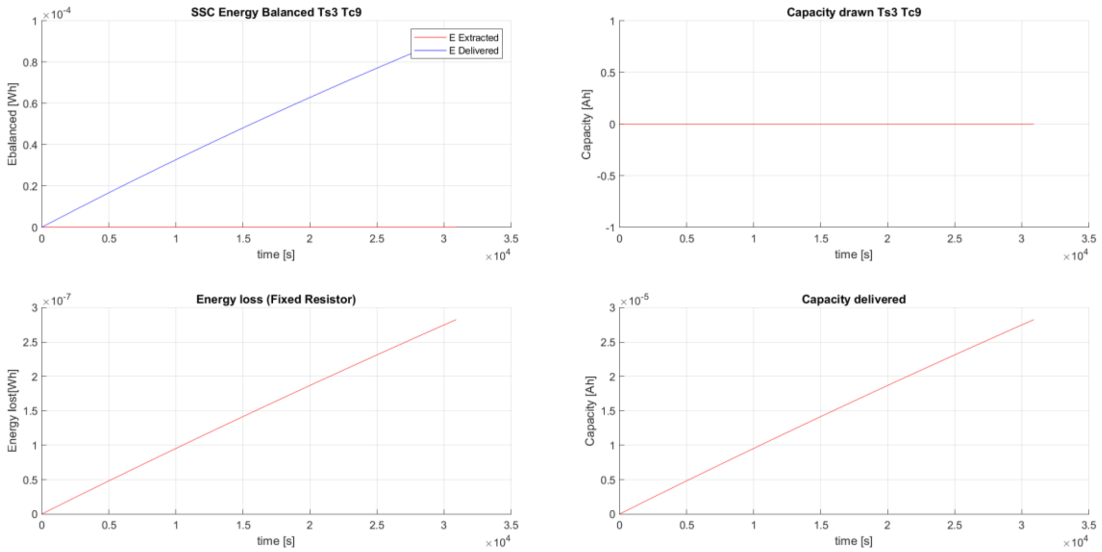


Figure 159. VOC-SOC table change test. Left: Energy parameters. Right: Capacity parameters.

4.4.5.3 SSC: Error introduction

For this specific section and to do not enlarge excessively the thesis report, the total number of images per SSC test has been reduced.

SSC: Cell Voltage measurement offset, +50 mV

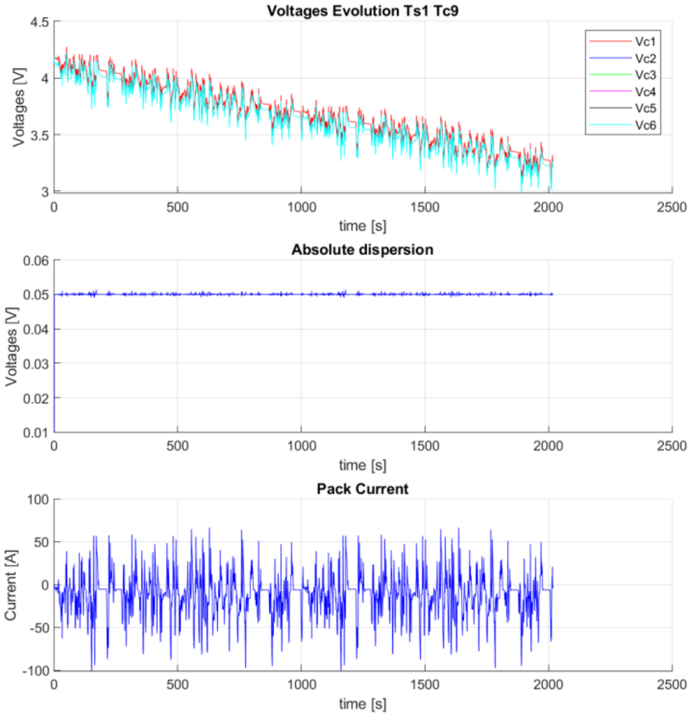


Figure 160. Cell voltage error introduction test. Voltage figure.

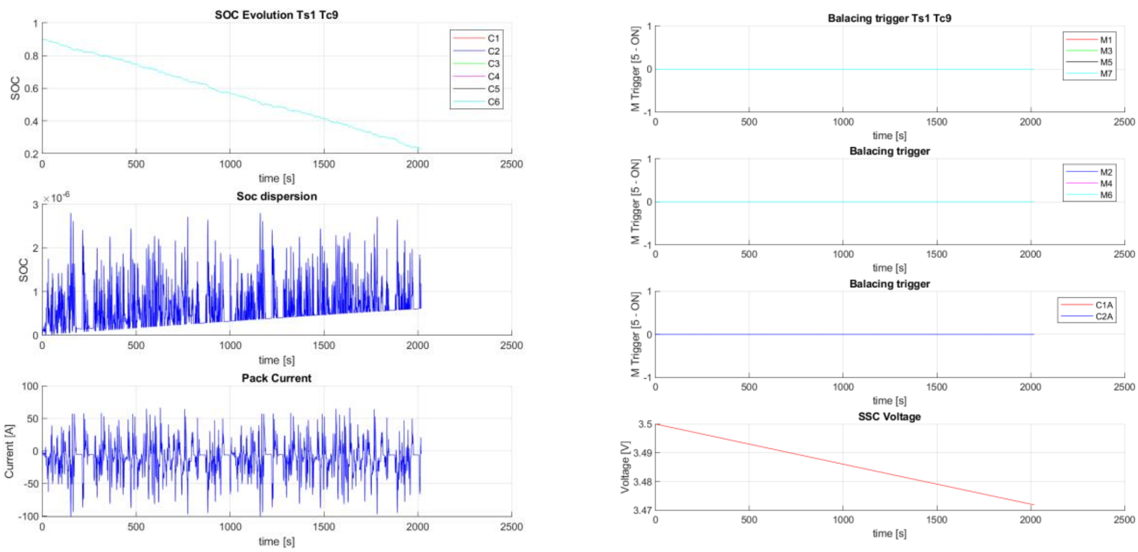


Figure 161. Cell voltage error introduction test. Left: SOC evolution. Right: Switches action.

SSC: Cell Voltage measurement offset, -50 mV

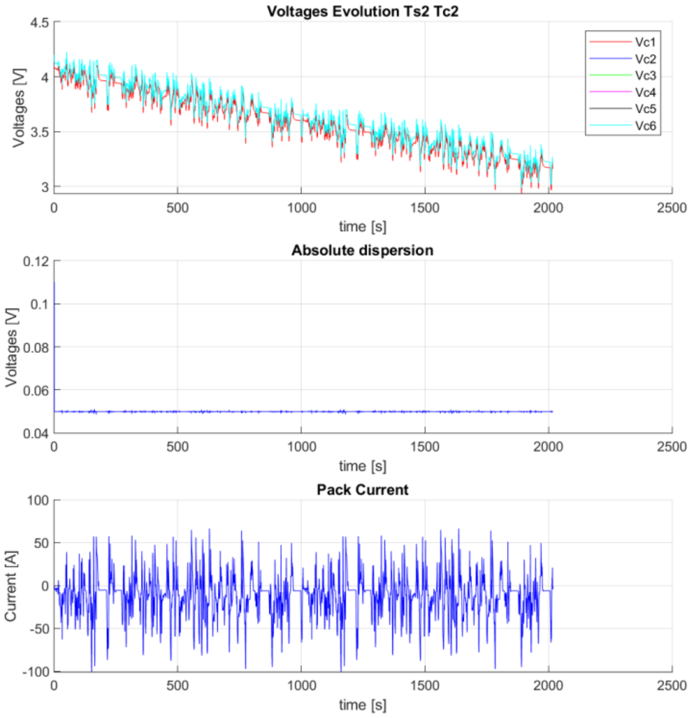


Figure 162. Cell voltage error introduction test. Voltage figure.

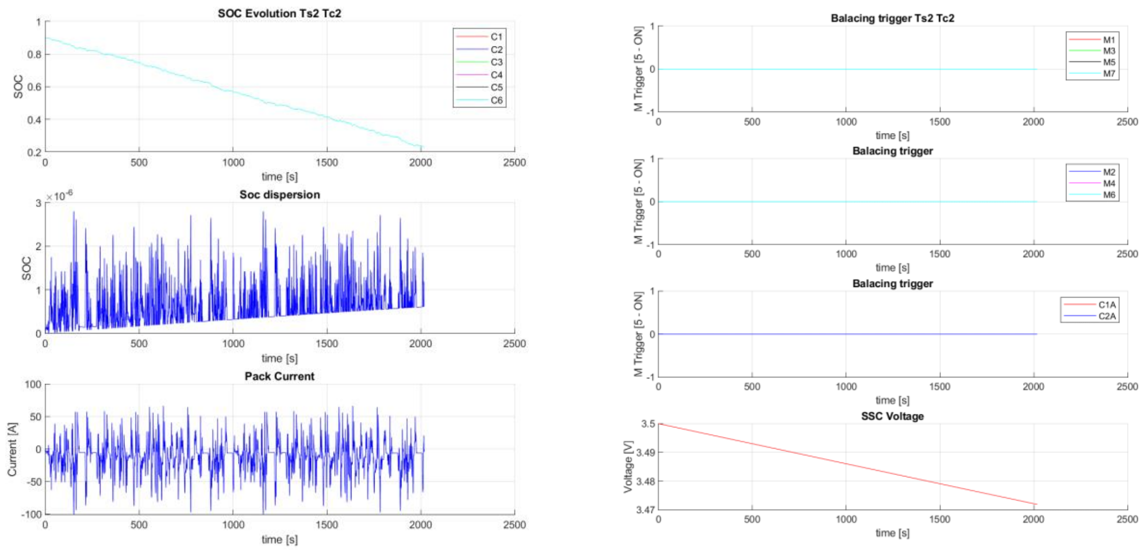


Figure 163. Cell voltage error introduction test. Left: SOC evolution. Right: Switches action.

SSC: SOC estimation offset, +2%

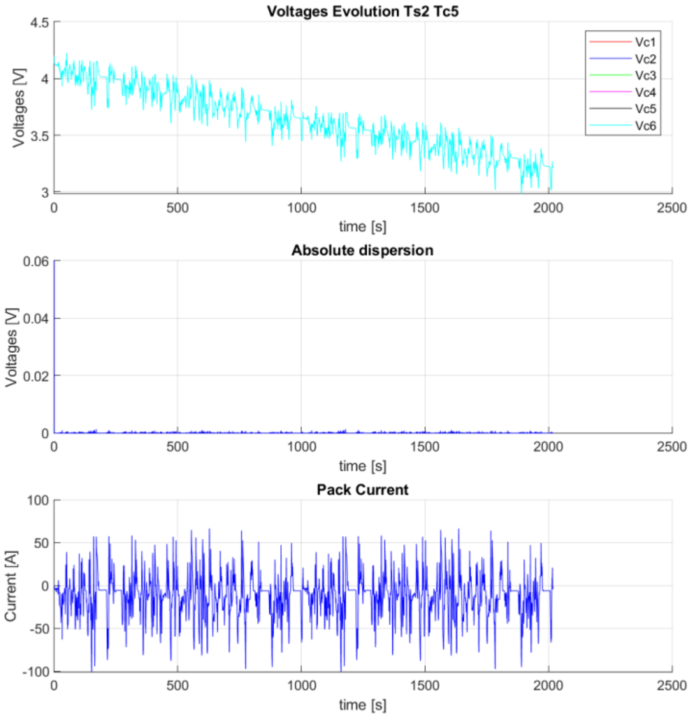


Figure 164. SOC estimation error introduction test. Voltage figure.

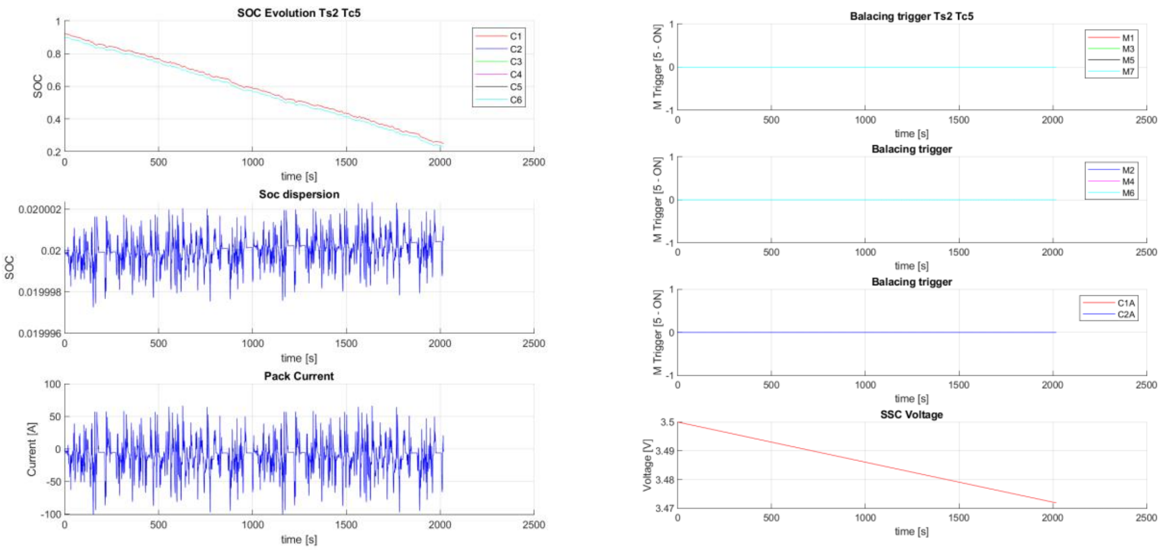


Figure 165. SOC estimation error introduction test. Left: SOC evolution. Right: Switches action.

SSC: SOC estimation offset, - 2%

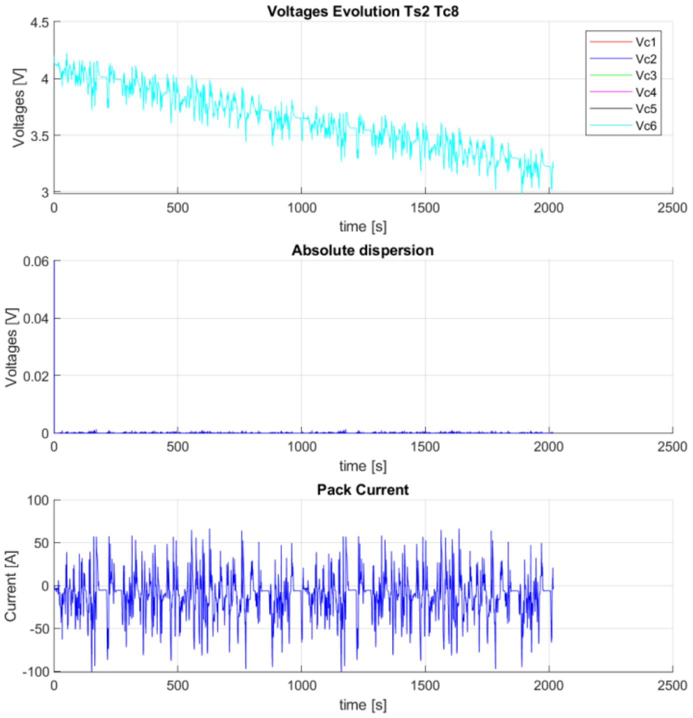
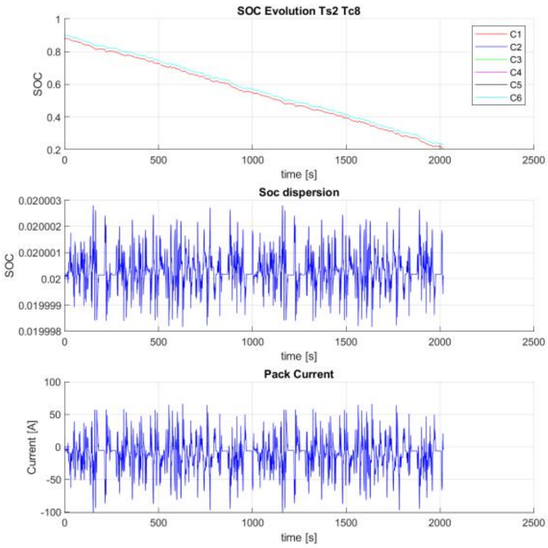


Figure 166. SOC estimation error introduction. Voltage figure.



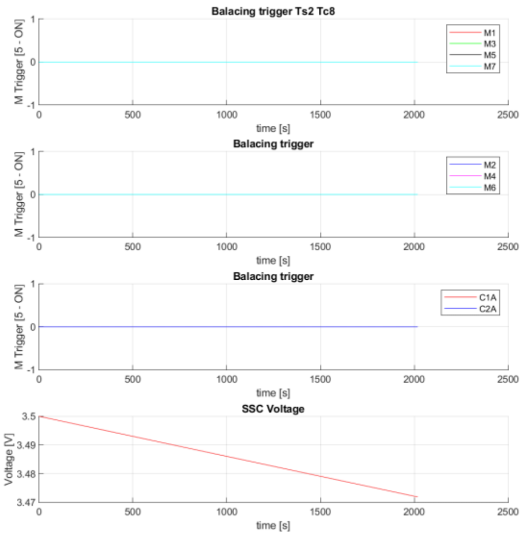


Figure 167. SOC estimation error introduction. Left: SOC evolution. Right: Switches action.

SSC: Current sense offset +250 mA

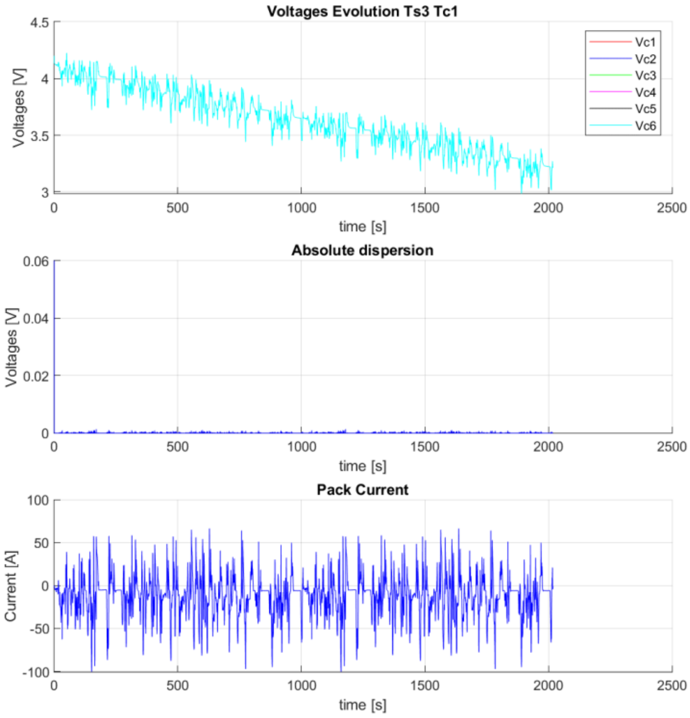


Figure 168. Current measurement error introduction test. Voltage figure.

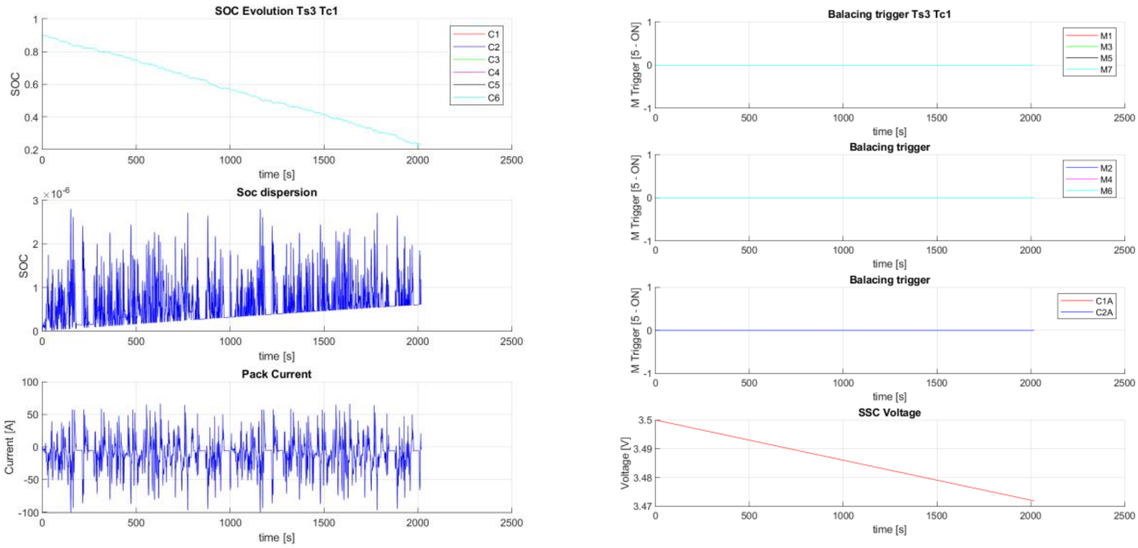


Figure 169. Current measurement error introduction test. Left: SOC evolution. Right: Switches action.

SSC: Current sense offset, - 250 mA

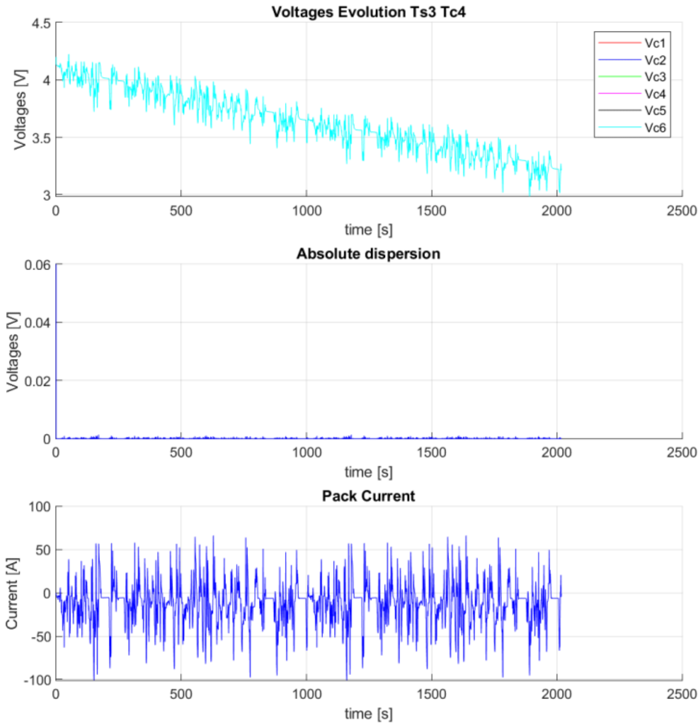


Figure 170. Current measurement error introduction test. Voltage figure.

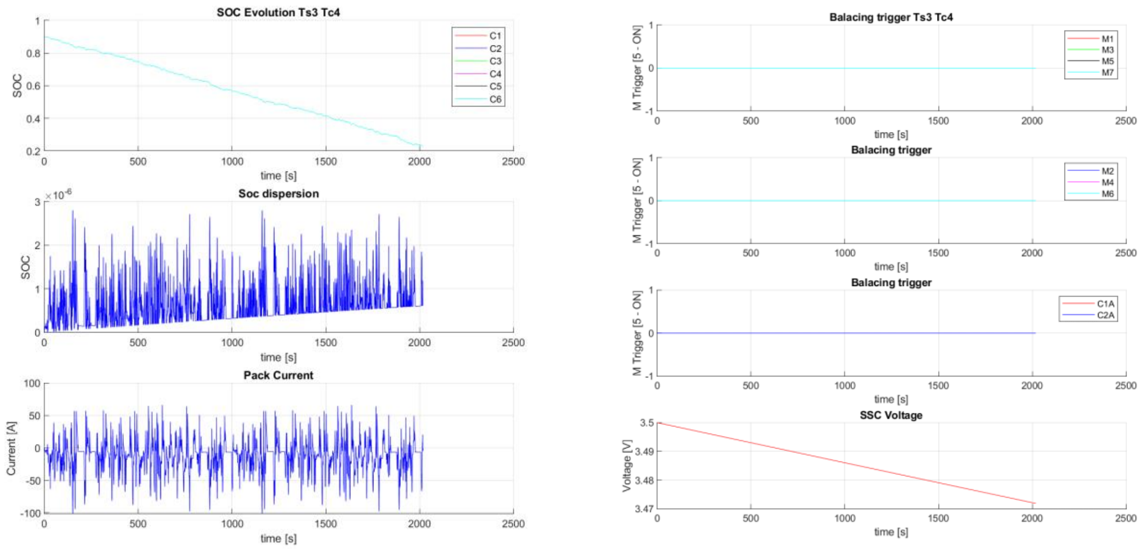


Figure 171. Current measurement error introduction test. Left: SOC evolution. Right: Switches action.

4.4.5.4 SSC conclusion

After the different sets of tests, a series of conclusions can be made regarding SSC. These conclusions will evaluate the technique on different the test scenarios (functional, aging and Error introduction).

Functional wise:

- The algorithm is capable to balance all the cells to finally achieve a full balance scenario. In Figure 136 and Figure 137 it can be observed that from a 6% SOC difference, thanks to the balancing action, a final estate of balance is achieved.
- The algorithm is functional in both cycle tests and profile tests (Figure 139 and Figure 142).

Functional – Balancing Capability:

- The maximum allowed current if there's a voltage difference of 0.2 V between two cells is 1A. The current values seen on the test cases is below 1 A in all scenarios.
- In the test of Figure 135, there's an initial SOC difference of 5 %. This means that a total of 0.5 Ah need to be extracted from the first cell and must be delivered to the other cells. In the top right figure (Capacity drawn) a total of 0.33 Ah have been removed from the most charged cell. This value is correct as this 0.33 Ah are distributed between the 5 less charged cells (each cell receives 0.066 Ah). As the top charge cell has 0.33 Ah less and the other cells are charged 0.066 Ah, the SOC difference between cells is now (65). As the cells have a 10 Ah capacity, this capacity value means a 1% SOC. This value is the threshold to balance. The performance of the circuit is correct.

$$Final\ Cap\ Difference = 0.5 - 0.33 - 0.066 \cong 0.104\ Ah \rightarrow 1\%\ SOC \qquad (65)$$

Aging:

- Internal resistance: The algorithm performs correctly having a cell with a different internal resistance value. On the internal resistance aging tests (Figure 145), it has been seen the well behaviour of the algorithm over 5 cycles.
- Capacity: The difference in capacity creates a difference in SOC over the threshold during the charge process, so balancing action is started (Figure 151 and Figure 152).

  During discharge, the difference on SOC is on the other way round. The balancing system starts working in "reverse". The overall performance of the system can be considered correct as it is not worsening the effect. Instead, is trying to get the cells to the same SOC.
- VOC-SOC Change: Figure 157 and Figure 158 show that despite the change on the voltage seen on the cell's terminals, no balancing action is performed. The selection is based on SOC, so the voltage is not a variable that affects the control part.

Error introduction:

- Cell measurement: The SSC balancing algorithm do not perform any unnecessary balancing when there's an error on the cell measurement (Figure 160 and Figure 161). The algorithm is immune to errors on voltage measurement.
- SOC estimation: The algorithm does rely on SOC estimation for its calculation parameters. The first thought would be that it would affect the SSC algorithm, and unnecessary balancing would be performed. However, the figures of the test report

show that no balancing action is performed (Figure 165 and Figure 167). This is because after determining the cells that need to be balanced, there's a voltage comparison on the algorithm. As the value is the same of both cells, no balancing is performed. The algorithm is immune to errors on SOC estimation if no cell measurement error is also added.

- Current offset: On Figure 170 no balancing action is performed. The algorithm does not rely on current measurement on any parameter calculation. The algorithm is immune to errors on current measurement.

Final conclusions:

Over the different passive balancing methods, SSC presents a larger complexity in the control side. However, it has been proven effective on balancing the different cells with no need of dissipating energy.

The operation of this circuit is in a low frequency switching. In this case, the switching frequency is 100 Hz. Compared to a power converter, is a low frequency.

The frequency needs to be chosen accordingly with the equivalent resistance of the circuit and the value of the capacitor used. If it's needed to have a smaller value capacitor, there just has to be an increase on the switching frequency to keep the same energy transfer.

Looking at its performance in terms of aging on the cells and error introduction, it presents the best behaviour out of **all** the algorithms tested. Even the SOC estimation error does not lead to an error on the operation of the system.

4.4.6 SSC Multi Module

SSC Multi Module: Test 1

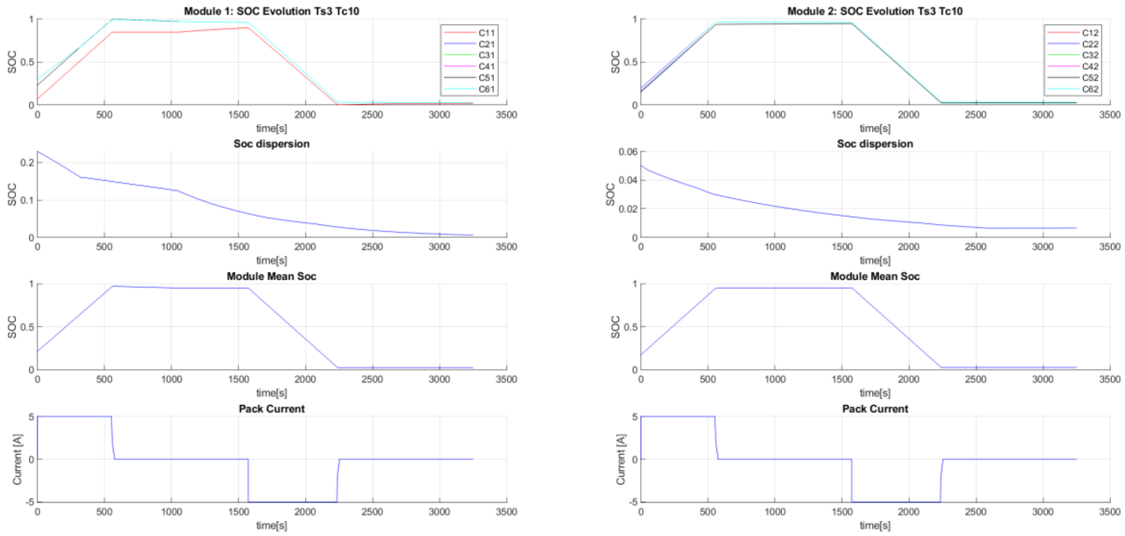


Figure 172. SSC Multi Module test. Left: SOC evolution M1. Right: SOC evolution M2.

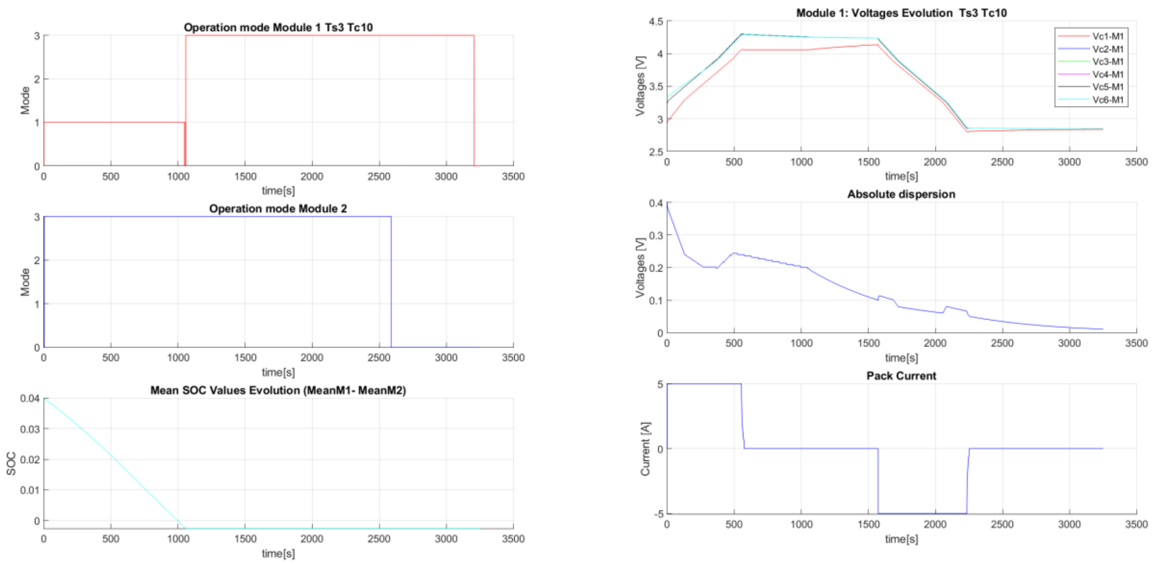


Figure 173. SSC Multi Module test. Left: Operation mode. Right: Voltage evolution M1.

Operation mode legend: 0 No operation, 1 Pure passive balancing, 2 Active-Passive, 3 Pure SSC.

SSC Multi Module: Test 2

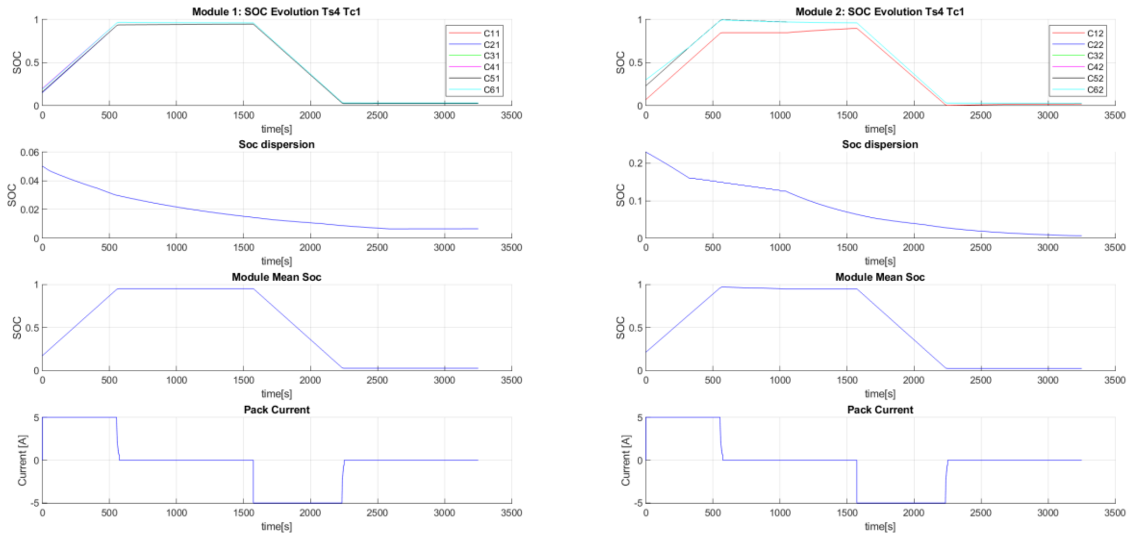


Figure 174. SSC Multi Module test. Left: SOC evolution M1. Right: SOC evolution M2.

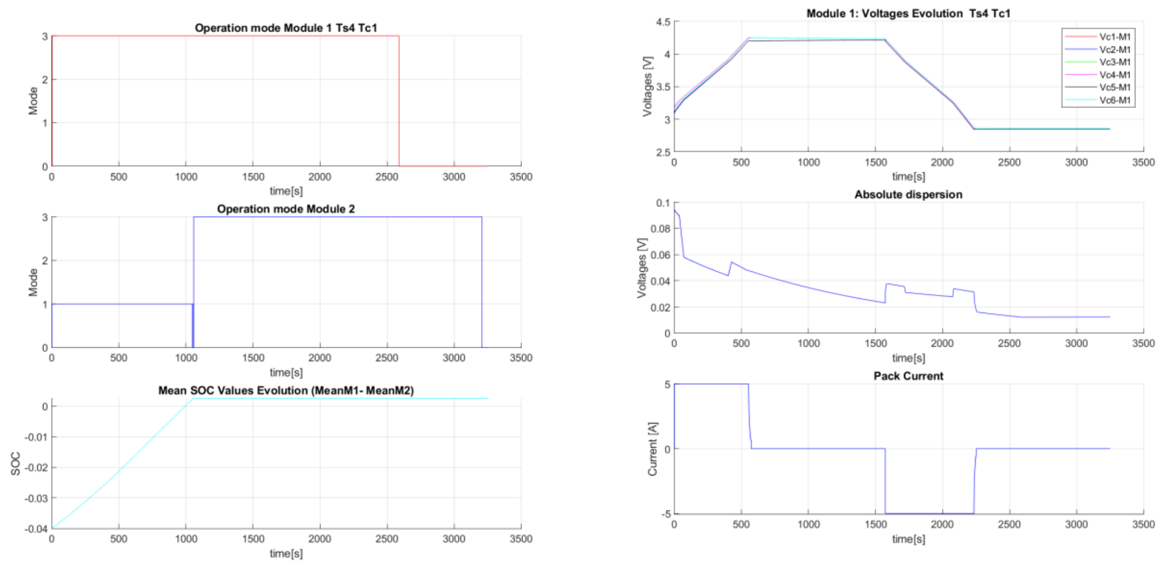


Figure 175. SSC Multi Module test. Left: Operation mode. Right: SOC evolution M1.

Operation mode legend: 0 No operation, 1 Pure passive balancing, 2 Active-Passive, 3 Pure SSC.

SSC Multi Module: Test 3

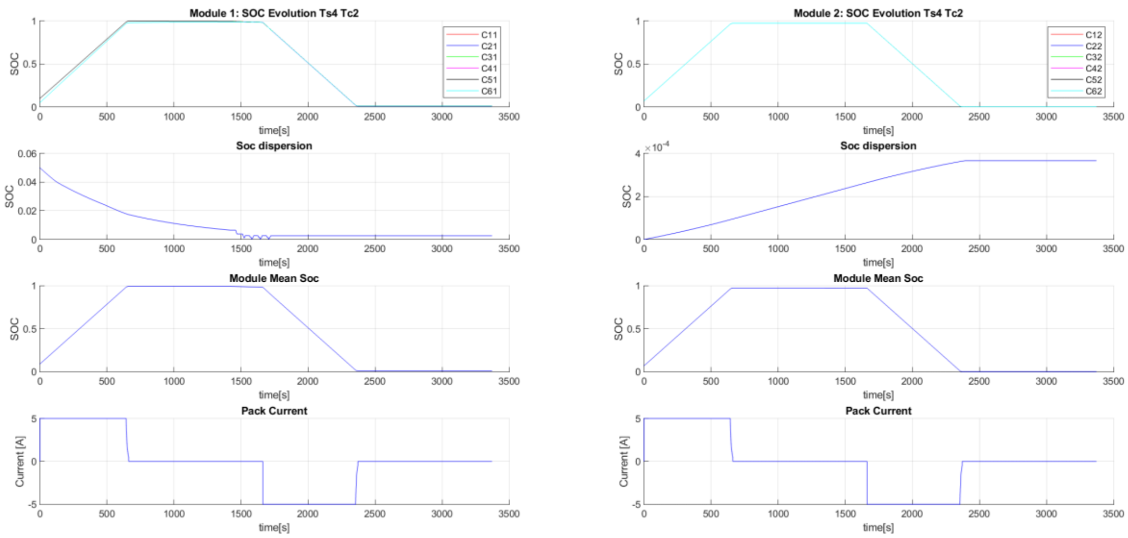


Figure 176. SSC Multi Module test. Left: SOC evolution M1. Right: SOC evolution M2.

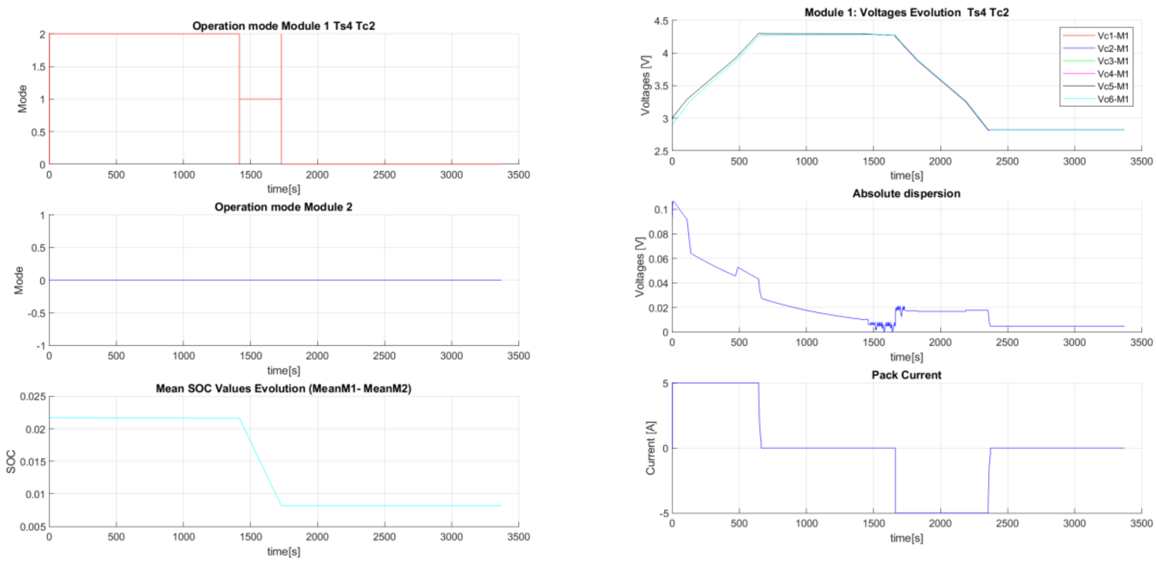


Figure 177. SSC Multi Module test. Left: Operation mode. Right: SOC evolution M1.

Operation mode legend: 0 No operation, 1 Pure passive balancing, 2 Active-Passive, 3 Pure SSC.

The previous captures permit to verify the correct behaviour of the Multi-Module SSC technique. The first two tests are the same but with the initial SOC's inverted. The behaviour is the same in both cases but inverted. If we look at test 1:

- Module 2 has a mean SOC lower than module 1 and has some cells that are not in balance. The module starts pure SSC balancing to balance the cells. After 2500 s the balancing operation is ended as the module is balanced.
- Module 1 has a mean SOC higher than module 2. However, it has a cell that has a SOC below the mean value of SOC of module. Theoretically, it should enter active passive area to bring up that cell to a value close to the mean value of the second module. However, it enters passive balancing as the voltage difference is higher than 0.2 V, so SSC operation is not permitted.

  Module 1 starts SSC operation to balance its cells when the SOC mean values of the modules are equalized, and the voltage differences on the module are below 0.2 V.
- After 1000 seconds both modules have a mean SOC close to each other in less than 1% (Figure 173 and Figure 175 on *Mean SOC Values Evolution*).

Regarding test 3, it was intentionally created to show the behaviour of operation mode 2. Module 2 is perfectly balanced and has a mean SOC value lower than the module 1. Module 1 has a mean SOC value above the one of module 2. However, it presents a cell in a lower value than the mean SOC value of module 2.

The system first brings up the SOC of that cell (operation mode 2). Then it performs dissipative balance to make the mean values equal (final 0.075% difference). The system can enter operation mode 2 (SSC before passive balancing) because the voltage difference on the cells is approximately 0.1 V (below 0.2 V).

Conclusion

The Multi-Module SSC strategy implemented has been functional to achieve a full balance in a multi module battery pack. The addition of the balancing resistor permits to combine passive and active balance to make the modules fully balanced.

The simulations are performed with 1 Ah and the SOC differences used were high. However, this choosing of values permits to see that the designed system is functional.

In a real vehicle application this would be the SSC algorithm and strategy that shall be used. It makes the system more robust and reliable to different behaviours on different cell groupings, and it adapts to the current battery constructions.

5 Conclusions

5.1 Balancing topic

After simulating the different algorithms and extracting some conclusions for SSC and the dissipative based methods, some overall conclusions of the balancing topic must be made.

In future with larger EV batteries and together with aging, the balancing topic could become key for different manufacturers to extend the operational range of their vehicles.

SSC in my opinion, it's a technique that could be easily deployed currently on the market. It does not require much more switching HW compared with pure dissipative balancing. In the SW side it would not need much computing power as its operational values (frequency, duty cycle) are a constant parameter that do not change under the different scenarios that the system can face. The active balancing has less losses than the dissipative one. This traduces on less charge being wasted and heat generation.

SSC implementation proposal for a multimodule approach can be fully functional and provide a solution that is robust for the different scenarios that a battery pack can face. This proposal is the one that, from my point of view, should be done in a real application.

SSC for me has an only weak point, it is the maximum current allowed. This value is directly linked to the capacity to transfer energy, for different reasons it may be needed to be low. In these cases, the addition of SSC would be costly for the benefits that would generate into the battery pack.

On the other hand, there are the dissipative techniques. These techniques are reliable, simple and have been implemented on the market for a long time. However, the heat generation that they cause on the BMS PCBs together with the loss of useful energy make these systems not efficient for future EV batteries.

In the end, is a balance between the costs of SSC (extra HW, extra SW costs and the increase on section on the cell conductors) and the benefits that is expected to generate in respect of the pure dissipative ones. The expected benefits could be quantified after the analysis on the behaviour of a real battery.

5.2 Project Conclusions

On this master thesis the balancing topic for batteries has been addressed. First an introduction to the key concepts and current state of the art was presented. The necessity of balancing and its importance was also shown.

Six algorithms have been successfully implemented and tested on Simulink. Different Simulink resources and add-ons have been part on those implementations. Each one of this add-on's required a learning curve to achieve its full potential on the application into the different simulations.

It's important to highlight *Stateflow* in this section. It has permitted to implement complex techniques and new functionalities to the control software. It's clear why this is one of the most preferred ways for application software development. The implementation that is derived from the charts it's clear and makes the programmer aware of how it's going to evolve.

The SW development of the control side has been performed in *"the right way"*. On the implementation of the different algorithms, I committed different errors. In Simulink you can use different debugging tools to check the well behaviour of the implemented algorithms. In *Stateflow* charts you are provided with all the data that is relevant for the chart, as well as the current state. After this development of algorithms, it's implementation on a microcontroller could be done. There could be the way of even autogenerating code from MATLAB into the microcontroller. Surely, in the microcontroller implementation there would appear different issues. However, as most of the logic has been already debugged and tested, significant logic errors would be avoided.

This way of development is *"the right way"* as errors and issues are detected before the HW implementation. Changing code in the implemented version would be more difficult to debug due to the different elements as well as more costly.

A total of 3 HW model schematics have been created as well as the different control logic to trigger the different switching elements. System architecture and structuration has been needed to create an understandable project as well as to make the implementation task easier.

I would like to remark the SSC implementation. There are different references that address the topic of SSC. However, none of them provide as detailed information on the implementation side and its problematics, as the different sections of this thesis. Inrush current, selecting the correct operational range and operational values, and the reason behind each selection are covered in the different sections of these project. The disclosure of the selection criteria of the algorithm implementation on the papers is not described.

Furthermore, the implementation of a multi module approach has not been found in any written reference. This multi module approach has been performed thinking with the current state of the art in battery packs and with the intention to, with the current available variables, implement a strategy that permits a full balance situation with the SSC technique.

The implementation of the different algorithms has given me key knowledge on different points of SW implementation on different aspects related with battery management and charging. Furthermore, the constant use of MATLAB and Simulink has permitted me to achieve a lot of confidence on those systems for modelling and implementing functions.

To conclude the section and the project, I would like to add that has been a project in which I enjoyed working on. On every model and implementation I was exploring different options to fulfill the different challenges that could appear in a battery environment.

6 References

6.1 Online references

[1] IEEE. Battery and Heating Data in Real Driving Cycles. IEEE 2020. [22/04/2023] Available in :<https://ieee-dataport.org/open-access/battery-and-heating-data-real-driving-cycles>

[2] Texas instruments. Cell balancing buys extra run time and battery life. Texas instruments 2020. [22/04/2023]. Available in: <https://www.ti.com/lit/an/slyt322/slyt322.pdf?ts=1682157163335&ref_url=https%253A%252F%252Fwww.google.com%252F >

[3] MDPI. A Review on Battery Modelling Techniques. MDPI 2020. [22/04/2023]. Available in: < https://doi.org/10.3390/su131810042 >

[4] AABC. SIMULATIVE COMPARISON OF BALANCING ALGORITHMS FOR ACTIVE AND PASSIVE CELL BALANCING. AABC 2013. [22/04/2023]. Available in:<https://www.researchgate.net/profile/Dirk-Uwe-Sauer/publication/328175824_SIMULATIVE_COMPARISON_OF_BALANCING_ALGORITHMS_FOR_ACTIVE_AND_PASSIVE_CELL_BALANCING_SYSTEMS_FOR_LITHIUM-ION_BATTERIES/links/5bbd156f299bf1049b786378/SIMULATIVE-COMPARISON-OF-BALANCING-ALGORITHMS-FOR-ACTIVE-AND-PASSIVE-CELL-BALANCING-SYSTEMS-FOR-LITHIUM-ION-BATTERIES.pdf >

[5] WEVA. Capacitor Based Battery Balancing System. WEVA 2012. [22/04/2023]. Available in: < https://www.researchgate.net/publication/325209363_Capacitor_Based_Battery_Balancing_System>

[6] MDPI. Single Switched Capacitor Battery Balancing System Enhancements. MDPI 2013. [22/04/2023]. Available in: < https://www.researchgate.net/publication/236208664_Single_Switched_Capacitor_Battery_Balancing_System_Enhancements>

[7] MDPI. Single Switched Capacitor Battery Balancing System Enhancements. MDPI 2013. [22/04/2023]. Available in: < https://www.researchgate.net/publication/236208664_Single_Switched_Capacitor_Battery_Balancing_System_Enhancements>

6.2 Written references

[8] Davide Andrea. Battery Management Systems for Large Lithium-ion Battery Packs. ARTECH HOUSE. ISBN: 978-1-63081-482-3

[9] Robert W. Erickson, Dragan Maksimovic. Fundamentals of Power Electronics. Springer 2001. ISBN: 978-1-4757-0559-1.

[10] Gregory L. Plett. Battery Management Systems Volume II: Equivalent-Circuit Methods. ARTECH HOUSE. ISBN: 978-1-63081-027-6.

6.3 Image references

Figure	Origin
Figure 1	2021-2022 ENERGY STORAGE AND CONVERSION Slides. Carlos Olalla. Link: https://campusvirtual.urv.cat/pluginfile.php/4022316/mod_resource/content/6/ECE_L3_Battery_Modeling_I_OUT.pdf
Figure 2	Self-made merge of images. Right: Exploring Overland. Jonathan Hanson. Link: https://www.exploringoverland.com/overland-tech-travel/2021/3/29/lifepo4-battery-cylindrical-cells-or-prism-cells Left: Alibaba Pouch size NMC. Link: https://www.alibaba.com/product-detail/Pouch-size-NMC-20AH-30AH-40AH_62028924308.html
Figure 3	2021-2022 ENERGY STORAGE AND CONVERSION Slides. Carlos Olalla. Link: https://campusvirtual.urv.cat/pluginfile.php/4022316/mod_resource/content/6/ECE_L3_Battery_Modeling_I_OUT.pdf
Figure 4	Proprietary scope capture of a MATLAB simulation
Figure 5	Proprietary diagram
Figure 6	Electric has gone AUDI. Link: https://electrichasgoneaudi.net/models/e-tron-gt/drivetrain/battery/
Figure 7	Proprietary diagram
Figure 8	BMW I3 module offer on second life batteries. Link: https://www.secondlife-evbatteries.com/products/bmw-i3-120ah-module
Figure 9	TE Connectivity KILOVAC LEV 200. RS Components. Link: https://es.rs-online.com/web/p/contactores/7827082?cm_mmc=ES-PLA-DS3A-_-google-_-CSS_ES_ES_Automatizacion_y_Control_de_Procesos_Whoop-_-(ES:Whoop!)+Contactores+(2)-_-7827082&matchtype=&pla-338872034086&gclid=Cj0KCQjwwtWgBhDhARIsAEMcxeDE2zJ9PmCGYHWZSdGaYCTFKinxuOXHOuS4A_aqnETuTn76GbiP1JcaAnzMEALw_wcB&gclsrc=aw.ds
Figure 10	Eaton Fuses. Link: https://www.eaton.com/us/en-us/catalog/emobility/ev-fuses.html
Figure 11	Renhotec. Link: https://www.renhotecev.com/product/manual-service-disconnect-ev-plug-2-pins-550a?lang=ru
Figure 12	Proprietary diagram
Figure 13	Proprietary diagram
Figure 14	Proprietary diagram
Figure 15	Proprietary MATLAB schematic
Figure 16	Proprietary MATLAB simulation
Figure 17	Proprietary MATLAB simulation
Figure 18	Proprietary MATLAB simulation
Figure 19	Proprietary MATLAB simulation
Figure 20	Online reference [5]
Figure 21	Online reference [5]
Figure 22	Proprietary schematic
Figure 23	Proprietary MATLAB scope capture extracted from simulation
Figure 24	Proprietary MATLAB scope capture extracted from simulation
Figure 25	Proprietary MATLAB function block
Figure 26	Proprietary MATLAB schematic
Figure 27	Proprietary MATLAB stateflow chart
Figure 28	Proprietary MATLAB schematic configuration
Figure 29	Proprietary MATLAB stateflow chart
Figure 30	Proprietary MATLAB schematic
Figure 31	Proprietary MATLAB schematic
Figure 32	Proprietary MATLAB schematic

Figure 33	Proprietary MATLAB stateflow blocks
Figure 34	Proprietary MATLAB stateflow chart
Figure 35	Proprietary MATLAB stateflow chart
Figure 36	Proprietary MATLAB stateflow chart
Figure 37	Proprietary MATLAB schematic
Figure 38	Proprietary MATLAB schematic
Figure 39	Proprietary PSIM shcematic
Figure 40	Proprietary MATLAB plot
Figure 41	Proprietary MATLAB plot
Figure 42	Proprietary MATLAB plot
Figure 43	Proprietary MATLAB plot
Figure 44	Proprietary MATLAB plot
Figure 45	Proprietary MATLAB plot
Figure 46	Proprietary MATLAB plot
Figure 47	Proprietary MATLAB plot
Figure 48	Proprietary MATLAB plot
Figure 49	Proprietary MATLAB plot
Figure 50	Proprietary MATLAB plot
Figure 51	Proprietary MATLAB stateflow chart
Figure 52	Proprietary MATLAB circuit schematic
Figure 53	Proprietary MATLAB stateflow chart
Figure 54	Proprietary MATLAB stateflow chart
Figure 55	Proprietary MATLAB stateflow chart
Figure 56	Proprietary MATLAB Scope capture of a simulation
Figure 57	Proprietary MATLAB stateflow chart
Figure 58	Proprietary MATLAB simulation of an IEEE profile. Online reference [1]
Figure 59	Proprietary MATLAB simulation
Figure 60	Proprietary MATLAB simulation
Figure 61	Proprietary MATLAB simulation
Figure 62	Proprietary MATLAB simulation
Figure 63	Proprietary MATLAB simulation
Figure 64	Proprietary MATLAB simulation
Figure 65	Proprietary MATLAB simulation
Figure 66	Proprietary MATLAB simulation
Figure 67	Proprietary MATLAB simulation
Figure 68	Proprietary MATLAB simulation
Figure 69	Proprietary MATLAB simulation
Figure 70	Proprietary MATLAB simulation
Figure 71	Proprietary MATLAB simulation
Figure 72	Proprietary MATLAB simulation
Figure 73	Proprietary MATLAB simulation
Figure 74	Proprietary MATLAB simulation
Figure 75	Proprietary MATLAB simulation
Figure 76	Proprietary MATLAB simulation
Figure 77	Proprietary MATLAB simulation
Figure 78	Proprietary MATLAB simulation
Figure 79	Proprietary MATLAB simulation
Figure 80	Proprietary MATLAB simulation

Figure 81	Proprietary MATLAB simulation
Figure 82	Proprietary MATLAB simulation
Figure 83	Proprietary MATLAB simulation
Figure 84	Proprietary MATLAB simulation
Figure 85	Proprietary MATLAB simulation
Figure 86	Proprietary MATLAB simulation
Figure 87	Proprietary MATLAB simulation
Figure 88	Proprietary MATLAB simulation
Figure 89	Proprietary MATLAB simulation
Figure 90	Proprietary MATLAB simulation
Figure 91	Proprietary MATLAB simulation
Figure 92	Proprietary MATLAB simulation
Figure 93	Proprietary MATLAB simulation
Figure 94	Proprietary MATLAB simulation
Figure 95	Proprietary MATLAB simulation
Figure 96	Proprietary MATLAB simulation
Figure 97	Proprietary MATLAB simulation
Figure 98	Proprietary MATLAB simulation
Figure 99	Proprietary MATLAB simulation
Figure 100	Proprietary MATLAB simulation
Figure 101	Proprietary MATLAB simulation
Figure 102	Proprietary MATLAB simulation
Figure 103	Proprietary MATLAB simulation
Figure 104	Proprietary MATLAB simulation
Figure 105	Proprietary MATLAB simulation
Figure 106	Proprietary MATLAB simulation
Figure 107	Proprietary MATLAB simulation
Figure 108	Proprietary MATLAB simulation
Figure 109	Proprietary MATLAB simulation
Figure 110	Proprietary MATLAB simulation
Figure 111	Proprietary MATLAB simulation
Figure 112	Proprietary MATLAB simulation
Figure 113	Proprietary MATLAB simulation
Figure 114	Proprietary MATLAB simulation
Figure 115	Proprietary MATLAB simulation
Figure 116	Proprietary MATLAB simulation
Figure 117	Proprietary MATLAB simulation
Figure 118	Proprietary MATLAB simulation
Figure 119	Proprietary MATLAB simulation
Figure 120	Proprietary MATLAB simulation
Figure 121	Proprietary MATLAB simulation
Figure 122	Proprietary MATLAB simulation
Figure 123	Proprietary MATLAB simulation
Figure 124	Proprietary MATLAB simulation
Figure 125	Proprietary MATLAB simulation
Figure 126	Proprietary MATLAB simulation
Figure 127	Proprietary MATLAB simulation
Figure 128	Proprietary MATLAB simulation

Figure 129	Proprietary MATLAB simulation
Figure 130	Proprietary MATLAB simulation
Figure 131	Proprietary MATLAB simulation
Figure 132	Proprietary MATLAB simulation
Figure 133	Proprietary MATLAB simulation
Figure 134	Proprietary MATLAB simulation
Figure 135	Proprietary MATLAB simulation
Figure 136	Proprietary MATLAB simulation
Figure 137	Proprietary MATLAB simulation
Figure 138	Proprietary MATLAB simulation
Figure 139	Proprietary MATLAB simulation
Figure 140	Proprietary MATLAB simulation
Figure 141	Proprietary MATLAB simulation
Figure 142	Proprietary MATLAB simulation
Figure 143	Proprietary MATLAB simulation
Figure 144	Proprietary MATLAB simulation
Figure 145	Proprietary MATLAB simulation
Figure 146	Proprietary MATLAB simulation
Figure 147	Proprietary MATLAB simulation
Figure 148	Proprietary MATLAB simulation
Figure 149	Proprietary MATLAB simulation
Figure 150	Proprietary MATLAB simulation
Figure 151	Proprietary MATLAB simulation
Figure 152	Proprietary MATLAB simulation
Figure 153	Proprietary MATLAB simulation
Figure 154	Proprietary MATLAB simulation
Figure 155	Proprietary MATLAB simulation
Figure 156	Proprietary MATLAB simulation
Figure 157	Proprietary MATLAB simulation
Figure 158	Proprietary MATLAB simulation
Figure 159	Proprietary MATLAB simulation
Figure 160	Proprietary MATLAB simulation
Figure 161	Proprietary MATLAB simulation
Figure 162	Proprietary MATLAB simulation
Figure 163	Proprietary MATLAB simulation
Figure 164	Proprietary MATLAB simulation
Figure 165	Proprietary MATLAB simulation
Figure 166	Proprietary MATLAB simulation
Figure 167	Proprietary MATLAB simulation
Figure 168	Proprietary MATLAB simulation
Figure 169	Proprietary MATLAB simulation
Figure 170	Proprietary MATLAB simulation
Figure 171	Proprietary MATLAB simulation
Figure 172	Proprietary MATLAB simulation
Figure 173	Proprietary MATLAB simulation
Figure 174	Proprietary MATLAB simulation
Figure 175	Proprietary MATLAB simulation
Figure 176	Proprietary MATLAB simulation

Figure 177	Proprietary MATLAB simulation