



# Article Digital Impedance Emulator for Battery Measurement System Calibration

Francesco Santoni \*D, Alessio De Angelis D, Antonio Moschitta D and Paolo Carbone

Engineering Department, University of Perugia, 06125 Perugia, Italy; alessio.deangelis@unipg.it (A.D.A.); antonio.moschitta@unipg.it (A.M.); paolo.carbone@unipg.it (P.C.) \* Correspondence: francesco\_santoni@unipg.it

\* Correspondence: francesco.santoni@unipg.it

**Abstract:** Meaningful information on the internal state of a battery can be derived by measuring its impedance. Accordingly, battery management systems based on electrochemical impedance spectroscopy are now recognized as a feasible solutions for online battery control and diagnostic. Since the impedance of a battery is always changing along with its state of charge and aging effects, it is important to have a stable impedance reference in order to calibrate and test a battery management system. In this work we propose a programmable impedance emulator that in principle could be used for the calibration of any battery management system based on electrochemical impedance spectroscopy. A digital finite-impulse-response filter is implemented, whose frequency response is programmed so as to reproduce exactly the impedance of a real battery in the frequency domain. The whole design process of the filter is presented in detail. An analytical expression for the impedance of real battery in the frequency domain is derived from an equivalent circuit model. The model is validated both through numerical simulations and experimental tests. In particular, the filter is implemented on a low-cost microcontroller unit, and the emulated impedance is measured by means of a custom-made electrochemical impedance spectroscopy measuring system, and verified by using standard commercial bench instruments. Results on this prototype show the feasibility of using the proposed emulator as a fully controllable and low-cost reference for calibrating battery impedance measurement systems.

**Keywords:** battery management; impedance spectroscopy; impedance emulator; digital filter; fir filter; instrument calibration

# 1. Introduction

Measuring the impedance of a battery is being increasingly recognized as a fundamental step for its online diagnostic [1,2], i.e., when it is connected and operating in any battery-powered electric/electronic system. It is well known that Electrochemical Impedance Spectroscopy (EIS) data can provide meaningful information on the internal state of a battery [3,4]. Different portions of the impedance curve in the frequency domain are indeed related to different internal components and processes that can be modeled by means of equivalent circuits, and that are correlated with the battery State-of-Charge (SOC) and the State-of-Health. Yet as much as this topic has been widely investigated, most of the published experimental results are obtained by means of standard laboratory bench equipment that is not suited for online applications [5]. Online battery control and diagnostic systems, standardly referred to as Battery Management Systems (BMSs), are a key component in many applications, the automotive sector in particular being presently the subject of extensive research [6,7]. The integration of online EIS measurements into BMSs is an important development that is currently being investigated by several authors, with some promising solutions that have already been published [8–12].

Calibration and test under several working conditions are unavoidable stages in the developing process of a BMS [13,14]. The use of real batteries at this stage is not feasible,



**Citation:** Santoni, F.; De Angelis, A.; Moschitta, A.; Carbone, P. Digital Impedance Emulator for Battery Measurement System Calibration. *Sensors* **2021**, *21*, 7377. https:// doi.org/10.3390/s21217377

Academic Editor: Marco Carratù

Received: 20 October 2021 Accepted: 3 November 2021 Published: 6 November 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/). since the internal state of a battery and its response to external signals and solicitations change among different batteries (even of the same brand and model). Moreover, the characteristics of a single battery are not stable enough when repeated measurements are performed. A fully controllable reference is thus needed. Commercial bench instruments, such as the Keithley 2281S Series or ITECH IT6400 Series, are DC power supplies allowing for both current sourcing and sinking, which can simulate the voltage of a battery according to a predefined discharge curve [15]. These instruments are well suited to test charge and discharge cycles, but they do not emulate the impedance of a real battery and cannot be used to simulate the time transients expected in operating conditions.

A BMS testing approach frequently discussed in the literature is based on the so-called Hardware-In-the Loop paradigm (HIL) [16–18]. In general, a HIL simulator is a hardware that emulates all the input and outputs of the actual system under consideration. In this case, the HIL is designed to emulate a battery or a battery pack, its voltage, current, frequency response, state-of-charge, aging and failures. The BMS under test is interfaced to the HIL, and it can then be operated as if it were connected to an actual battery pack. The HIL is a piece of hardware fast enough to calculate and reproduce in real time the outputs of a real battery according with all the external inputs and solicitations. In general, a HIL simulator is a complex modular system that has to be designed and assembled for specific operations and requirements. Widely used commercial equipment is provided by dSPACE GmbH. In particular, the dSPACE EV1077 emulation board can emulate four cells and in principle can be programmed according to any mathematical model, such as equivalent circuit models including temperature and aging effects. Less complex and even low-cost solutions have also been presented in the literature [19–21].

In [22], a programmable setup implementing impedance emulation is proposed for the calibration of LCR meters. A power supply is used to sink the current supplied by the LCR meter, while a voltage generator and a current generator are used to emulate the voltage drop *V* generated by a current *I* across an arbitrary impedance *Z*. Both *V* and *I* are then measured by the LCR-meter in order to estimate  $Z = \frac{V}{I}$ . In this work, we propose an even simpler method to emulate the impedance of a battery that can be used as a reference to calibrate an EIS based BMS or other EIS equipment. The basic idea is to program a Finite Impulse Response filter (FIR), so that its frequency response would match exactly the impedance of the battery that has to be emulated. A general overview of the method is presented in Section 2. An experimental realization will be presented in Section 4.

The main difference of the proposed emulator with respect to the other reviewed solutions is that it does not source or sink any current. As regards the advantages, the emulator can be simply connected in place of the battery to the EIS equipment. It is programmable virtually with any impedance curve, so as to provide a reference for any state of a real battery. Finally, it is a low-cost solution requiring minimal equipment and components, that can be easily replicated to emulate multiple batteries.

## 2. General Overview of the Impedance Emulation Method

We refer to the BMS/EIS equipment presented in [11], since it is a very simple system possibly representing the basic scheme of a class of BMSs to be developed. The reference EIS equipment is illustrated in Figure 1. It essentially consists of a controllable current source and two differential Analog-to-Digital Converters (ADCs). A controlled current *I* is injected into the battery through a shunt resistor  $R_{shunt}$  of known value. The injected current is estimated by measuring with ADC1 the voltage difference  $V_{shunt}$  across the shunt. ADC2 measures the voltage difference  $V_{out}$  across the battery. Thus, the complex impedance can be obtained as

$$Z(\omega) = \frac{V_{out}(\omega)}{I(\omega)} = R_{shunt} \frac{V_{out}(\omega)}{V_{shunt}(\omega)},$$
(1)

where  $\omega$  is the angular frequency in rad/s.





**Figure 1.** (Left): The scheme of the actual EIS measurement system connected to a battery. (**Right**): the same EIS instrument, but connected to the battery emulator implemented by means of a micro-controller unit provided with a unipolar ADC and a DAC.

The impedance emulator, sketched on the right of Figure 1, is designed to be connected in place of the real battery, with the only difference that the current *I* is not injected into the emulator, but flows directly towards the ground through a load resistor  $R_{load}$ . By means of an ADC, the emulator acquires the voltage  $V_{in}$  across the load, and outputs a voltage  $V_{out}$  through a Digital-to-Analog Converter (DAC). A Microcontroller Unit (MCU) is programmed to generate  $V_{out}$  according with a predefined impedance model. A picture of the built prototype is shown in Figure 2.



**Figure 2.** A picture of the prototype EIS measurement instrument and the impedance emulator. The EIS instrument consists of a custom-made current source with its power supply, and a Data Acquisition board (DAQ) that provides two differential ADCs (cf. Figure 1) and is also used to control the current source. The impedance emulator is implemented on a Texas Instruments MCU development board. A simple breadboard is used to connect the shunt resistor and to interface the BMS to the impedance emulator.

The MCU acts as a digital FIR filter, acquiring  $V_{in}$  at a sample rate  $F_s = \frac{1}{T_s}$ , obtaining as input and output the discrete time sequences  $x[n] = V_{in}(nT_s)$  and  $y[n] = V_{out}(nT_s)$ , which are related as:

$$y[n] = \sum_{k=0}^{N-1} h[k]x[n-k],$$
(2)

where h[n] is the impulse response of the system in the time domain, and N is the total number of samples. It is well known from the theory of digital filters [23] that h[n] is related to the frequency response of the system  $Z(\omega)$  by the Discrete Fourier Transform (DFT) as:

$$Z(\omega_k) = \sum_{n=0}^{N-1} h[n] e^{-i\frac{2\pi kn}{N}},$$
(3)

where  $\omega_k = \frac{2\pi F_s}{N}k$ . Thus, the MCU is programmed so as to keep the ratio of  $V_{out}(\omega)$  over  $V_{in}(\omega)$  always equal (numerically) to the battery impedance  $Z(\omega)$  that one wants to emulate, independently of the value selected for  $R_{load}$ . When the emulated impedance is measured by means of the EIS equipment, the following relations hold:

$$Z(\omega) = \frac{V_{out}(\omega)}{V_{in}(\omega)} = \frac{V_{out}(\omega)}{R_{load}I(\omega)} = \frac{R_{shunt}}{R_{load}} \frac{V_{out}(\omega)}{V_{shunt}(\omega)}.$$
(4)

Thus, once programmed, the same emulator could be in principle adapted to different EIS instruments by just selecting a suitable  $R_{load}$ .

A reference impedance model  $Z(\omega)$  has to be chosen. It can be obtained from the measured impedance of a real battery as it will be illustrated below. Then, the coefficients h[n] are obtained by inverting the DFT and stored in the non-volatile memory of the MCU. Each time a sample of  $V_{in}$  is acquired by the ADC, the convolution sum (2) is calculated, and the resulting  $V_{out}$  is written on the DAC register. The length of the sequence N is upper bounded by the memory capacity, and also by the clock frequency of the MCU, since the convolution sum has to be computed in a time shorter than the sampling period  $T_s$ ; N is also lower bounded in accordance with the frequency resolution  $\Delta f = \frac{F_s}{N}$  required by any particular application. The full design process of the FIR filter is illustrated step-by-step in Section 3, while the experimental implementation of a test prototype is presented in Section 4.

#### 3. Design of the Fir Filter

The impedance emulating FIR filter has been designed by going through the following steps. A paragraph will be devoted to the details of each step.

- 1. *Modeling the impedance of a battery.* Since the aim of the filter is to emulate a real battery, it has been designed to reproduce an experimental impedance. An analytical model facilitates the design process, and allows for more control, thus, the measured impedance has been fitted to an equivalent circuit model.
- 2. *Choosing the number of samples and the sampling rate.* This has to be done according to the frequency range of interest for the impedance, and with the memory capacity and the clock frequency of the MCU.
- 3. *Definition of the impulse response*. It has been derived from the impedance curve in the frequency domain by an Inverse DFT.
- 4. *A numerical simulation of the filter response*. This has been performed on MATLAB in order to verify that the model is correct.

## 3.1. Step 1: Modeling the Impedance of the Battery

As reference, we used the battery model ICR18650-26J by Samsung, that had been already used to test our custom EIS equipment in [11]. The chosen excitation current signal was a multisine whose frequency components were logarithmically spaced: [0.1, 0.2, 0.4, 1, 2, 4, 10, 20, 40, 100, 200, 400] Hz. We measured the impedance  $Z(\omega)$  for SOC 100% and 20%.

The experimental results are shown in Figure 3. The choice of the two SOCs is motivated by the fact that they are associated with two impedance curves that are significantly different, practically being at the extremes of the range of variation of the battery impedance. Hence, we wanted to check the performance of the system at both the extremes (see Section 5.1). The battery has been modeled as the equivalent circuit shown in the same figure. It is a model commonly used in the literature, which we had previously verified to fit well the experimental impedance curves of the Samsung battery [24]; it implements some fractional order components, i.e., two Constant Phase Elements (CPEs) and a Warburg element. The complex impedance of the equivalent circuit is:

$$Z(s) = R_0 + sL + \frac{R_1}{1 + R_1 Q_1 s^{\alpha_1}} + \frac{R_2}{1 + R_2 Q_2 s^{\alpha_1}} + \frac{\sqrt{2}A_w}{s^{0.5}},$$
(5)

where  $s = i\omega$ . The model has thus nine parameters  $\theta = [R_0, L, R_1, Q_1, \alpha_1, R_2, Q_2, \alpha_2, A_w]$  that have to be fitted to experimental data. The solid curves plotted in Figure 3 have been calculated by using Formula (5), after  $\theta$  had been estimated through a non-linear least-squares fitting algorithm. The chosen frequency interval allows detection of the two main features of the impedance curve: the semicircle at higher frequencies and the straight line at 45° at lower frequencies.



**Figure 3.** Measured impedance curve of the Samsung battery at SOC 100% and 20%. The equivalent circuit model shown on the right has been fitted to experimental data. A picture of the actual battery is also shown.

# 3.2. Step 2: Choosing the Number of Samples and the Sampling Rate

The number of samples *N* and the sampling rate  $F_s$  cannot be selected independently, since they both define the frequency resolution as  $\Delta f = \frac{F_s}{N}$ . The frequency range of interest is 0.1–400 Hz, hence, in order to meet the Nyquist condition, the sampling rate should be  $F_s > 800$  Sa/s. We chose  $F_s = 1000$  Sa/s. Given the memory capacity of the MCU used to implement the prototype (see the experimental Section 4), we set N = 30,000 Sa. Thus, the frequency resolution of our impedance emulator is  $\Delta f = 33$  mHz, which is enough to even discriminate between the two lower frequency components (100 and 200 mHz). On the chosen MCU, the computation of (2) with N = 30,000 requires 827 µs, which is compatible with the chosen sampling period  $T_s = 1$  ms. In case of a different frequency range of interest or different MCU for other applications, the parameters should be reconfigured accordingly to the following analogous criteria. The parameters of the prototype configuration are summarized in Table 1 in the experimental Section 4.

Feature	Value	
ADC range	3 V, unipolar	
ADC resolution	12 bit	
ADC sampling rate $F_s$	1000 Sa/s	
ADC sample & hold time	125 µs	
DÂC range	0–3 V	
DAC resolution	12 bit	
Number of samples N	30,000	
CPU cores	2	
CPU clock frequency	200 MHz	

Table 1. Technical features of the impedance emulator prototype.

# 3.3. Step 3: Definition of the Impulse Response

By construction, the frequency response of the FIR filter (i.e., numerically, the impedance that we want to emulate) is the DFT of the impulse response h[n]. Thus, by means of (5), the impedance  $Z(f_k)$  can be calculated for the frequency values  $\{f\}_k = \frac{F_s}{N_s}[0, 1, \dots, k, \dots, \frac{N}{2}]$  (i.e., N equally spaced frequency values in the range  $0-\frac{F_s}{2}$ ), then the coefficients h[n] are calculated by performing the inverse DFT of  $Z(f_k)$  and stored in the non-volatile memory of the MCU.

A problem arises with the impedance of the Warburg element  $Z_w(s) = \frac{\sqrt{2}A_w}{s^{0.5}}$ , since it is not defined for  $f_0 = 0$ . Thus, the inverse DFT

$$h[n] = \frac{1}{N} \sum_{k=0}^{N-1} Z(f_k) e^{i\frac{2\pi kn}{N}n}$$
(6)

cannot be computed. The solution is to use the low-frequency approximation of  $s^{-0.5}$  given in [25]:

$$Z_i = \sqrt{2}A_w \frac{s^4 + 36s^3 + 126s^2 + 84s + 9}{9s^4 + 84s^3 + 126s^2 + 36s + 1},$$
(7)

i.e., the fractional order response is approximated with an integer order one. Let us define:

$$\tilde{Z}_w(f) = \begin{cases} Z_i(f) & f < 1\text{Hz} \\ Z_w(f) & f \ge 1\text{Hz} \end{cases}$$
(8)

Hence, by defining  $Z_r = Z - Z_w$ , the impedance can be approximated as  $Z \approx Z_r + \tilde{Z}_w$ . A very good approximation is thus obtained, as shown in Figure 4.



į

**Figure 4.** FIR frequency response obtained by using the approximation (7) compared with the experimental data. The components  $Z_r$ ,  $Z_w$  and  $Z_i$  are also reported as to show the goodness of the approximation  $Z_w \approx Z_i$  for f < 1 Hz.

#### 3.4. Step 4: A Numerical Simulation of the Filter Response

The emulator model has been simulated with MATLAB in order to verify that numerical and discretization errors, as well as the noise will not significantly affect the results. Since on the MCU all the computations will be performed using 32-bit single-precision representation of numbers, in MATLAB the 32-bit precision was also explicitly selected, the 64-bit precision being the default.

Both the signal acquisition with the EIS equipment, and the acquisition and processing of the emulator have to be simulated. The simulation can be divided into three stages:

- 1. A simulated signal  $V_{in}$  is acquired by the EIS equipment at a sampling rate  $F_{aq}$ , and by the emulator at sampling rate  $F_s$ . ADC signal quantization is simulated;
- 2. The signal *V*<sub>out</sub> is computed by summing (2), and the DAC output is synthesized as a Zero-Order-Hold signal (ZOH);
- 3. The acquisition of  $V_{out}$  by the EIS equipment at sampling rate  $F_{aq}$  is simulated. The impedance is estimated as the ratio of the DFTs of  $V_{out}$  and  $V_{in}$ .

#### 3.4.1. Simulation Stage 1

Sampling rates and ADC ranges are reported in Tables 1 and 2 on the experimental Section 4.  $F_s = 1$  kSa/s, and initially  $F_{aq} = 10$  kSa/s. The acquisition window is 30 s (number of samples  $N_{aq} = 300,000$ ). A multisine signal  $V_{in}$  was generated with frequency components [0.1, 0.2, 0.4, 1, 2, 4, 10, 20, 40, 50, 80, 100, 200, 400] Hz. The amplitude of each component is 50 mV. Signal quantization was simulated as:

$$V_{in}^{q} = \left\lfloor (V_{in} + \epsilon) \frac{\text{ADC range}}{\text{ADC levels}} \right\rfloor,\tag{9}$$

where  $\lfloor \rfloor$  indicates the *floor* function, while  $\epsilon$  is a zero-mean Gaussian noise with variance  $\sigma^2$ . Simulations were performed by setting  $\sigma = 0$  and 3 mV. The second value for the noise was used since it is comparable to the random noise observed on experimental measurements. The simulated input signal is shown in Figure 5 for  $\sigma = 3$  mV.

Table 2. Settings of the data acquisition system.

Feature	EIS Equipment/Keysight DAQ	Bench Oscilloscope	
ADC1 range	5 V, bipolar	1.2 V, bipolar	
ADC2 range	1.25 V, bipolar	0.12 V, bipolar	
ADC1/2 resolution	16 bit	8 bit	
ADC1/2 sampling rate $F_{aq}$	10 and 100 kSa/s	10 and 100 kSa/s	
Number of samples N <sub>aq</sub>	2 MSa	2 and 5 MSa	



**Figure 5.** (Left): The power spectrum of the generated  $V_{in}$ . (Right): A portion of  $V_{in}$  in the time domain. The sampled and quantized sequences are renormalized for ADC ranges and levels, in order to be shown together with the original signal.

## 3.4.2. Simulation Stage 2

The output  $V_{out}$  resulting from (2) is shown in Figure 6. The solid blue line is the ZOH signal generated by the DAC with period  $T_s$ , and  $\sigma = 3$  mV. The green line with dot markers is the simulation result of the DAC signal as acquired by the EIS measurement system with sampling period  $T_{aq} = 10T_s$ . Additionally, the ADC quantization for  $V_{out}$  was simulated as it was done for  $V_{in}$ . The analogous results for  $\sigma = 0$  are also reported in the plot. It can be noted that  $V_{out}$  as synthesized by the DAC is very stable when going from  $\sigma = 0$  to  $\sigma = 3$  mV; this is explained by the fact that  $V_{out}$  is approximately an order of magnitude lesser than  $V_{in}$ , hence the noise acquired with  $V_{in}$  is reduced accordingly, and it almost disappears given the finite resolution of the DAC. Then, the noise is added again as the DAC signal is re-acquired by the EIS system.



**Figure 6.**  $V_{out}$  resulting from (2) simulated as a zero-order-hold signal generated by the DAC and sampled with the ADC2 of the EIS instrument. See Section 3.4.2 for more details.

## 3.4.3. Simulation Stage 3

The frequency spectra of both input and output signals are obtained by applying a Fast Fourier Transform (FFT) to the sampled sequences of measured voltages:  $V_{in}(f_k) = FFT\left(V_{in}^q[n]\right)$  and  $V_{out}(f_k) = FFT\left(V_{out}^q[n]\right)$ , where  $\{f\}_k = \frac{F_{aq}}{N_{aq}}[0, 1, \dots, k, \dots, \frac{N_{aq}}{2}]$ . The spectrum of  $V_{out}$  has to be corrected for the distortion introduced by the ZOH as

$$V_{out}(f_k) \to V_{out}(f_k) \exp\left[i2\pi f_k(T_s - T_{aq})\right] \frac{\operatorname{sinc}(f_k T_{aq})}{\operatorname{sinc}(f_k T_s)}.$$
(10)

This correction is explained and formally derived in Appendix A. The ratio between the corrected  $V_{out}(f_k)$  (10) and  $V_{in}(f_k)$  yields the FIR frequency response.

#### 3.4.4. Simulation Results and Discussion

In Figure 7, the simulated response of the FIR filter is reported and compared with the analytical response function (5). For  $\sigma = 0$  the agreement is very good for both amplitude and phase. For  $\sigma = 3$  mV, the agreement is still very good for the amplitude, but some errors are introduced in the phase, in particular at higher frequencies f > 20 Hz. Regardless, in the worst case, the relative error affecting  $\Im(Z)$  is 9%, while the mean relative error is 3%. The relative error on  $\Re(Z)$  is negligible, its mean value being 0.2%, and 0.4% in the worst case.



**Figure 7.** (Left): Nyquist plot of the simulated FIR frequency response compared with the analytical response. (**Right**): Bode plots of amplitude and phase of the FIR frequency response.

The simulation was performed assuming that signal acquisitions on the EIS measurement system and on the emulator are perfectly synchronous. Although it would be possible, in principle, to implement the synchronization on the actual system, the emulator is intended to be a portable instrument applicable to different equipment, also when the synchronization is not possible. When the two systems are not synchronized, the sampling instant of the emulator can fluctuate with a flat distribution within the  $T_{aq}$  sampling period of the EIS equipment. This results in a random delay or anticipation between  $V_{in}$  and  $V_{out}$  affecting the phase of the impedance. Several simulations have been performed by including between  $V_{in}$  and  $V_{out}$  a random delay  $\Delta t$  in the interval  $\left(-\frac{T_{aq}}{2}, \frac{T_{aq}}{2}\right)$ . The Bode plots of the frequency response for a couple of simulations are shown in Figure 8. The response amplitude is not affected at all, but for  $F_{aq} = 10$  kSa/s, the phase measured at higher frequencies is not repeatable because of the random delay fluctuations. The only solution to this problem, if synchronization is not feasible, is to use a higher sampling frequency, such as  $F_{aq} = 100$  kSa/s, as shown on the right of Figure 8.



**Figure 8.** Effects of the non-synchronization of the EIS board with the emulator board. A random delay or anticipation has been inserted between  $V_{in}$  and  $V_{out}$  in the simulation. (Left): Simulation results for amplitude and phase of the FIR frequency response when the sampling period of the EIS board is set to  $T_{aq} = 100 \ \mu$ s. (Right): Simulation results when  $T_{aq} = 10 \ \mu$ s.

#### 4. Experimental Implementation

#### 4.1. Implementation of the Impedance Emulator on an MCU

The impedance emulation method described above could be in principle implemented on any MCU equipped with an ADC and a DAC, provided that its specifications such as clock frequency, ADC sampling rate, resolution, and so on, meet the requirements of the particular application it is intended to be used for. In order to test the devised method, we used the low-cost development board LAUNCHXL-F28379D by Texas Instruments, mounting the 32-bit, 200 MHz dual-core microcontroller TMS320F28379D. All the relevant specification and settings of this implementation are summarized in Table 1. Important configuration details have already been discussed on Section 3.2. Here we add that the duration of the ADC sample and hold window had to be set at least to 125  $\mu$ s as reported, otherwise, for shorter times, the acquired signal was to noisy, resulting in unacceptable measurement errors.

The computation of the convolution sum (2) was performed as follows. The ADC buffer consists of N 16-bit unsigned integer memory locations storing the sequence x[n], while the impulse response h[n] is stored in a constant buffer consisting of N 32-bit floating point memory locations. In order to store the continuously updating x[n], circular buffering was used, i.e., a pointer to an address of the ADC buffer is incremented by one at each acquisition, and when the end of the buffer is reached, the pointer returns to the beginning of the buffer; hence, after N acquisitions, older values start to be overwritten, since they are not needed anymore. In order to exploit the dual-core parallelism of the MCU, the convolution was split into two separate sums, one over the even terms, computed by core 1, and one over the odd terms, computed by core 2. The sequence of the iterations performed for both sums is illustrated in Figure 9, starting from the memory location labeled as t storing the last acquired sample.



**Figure 9.** The sequence of the iterations performed to compute the convolution (2), split into two separate sums over the even and odd terms, respectively. The first term of the summation is indicated in green, that is, the memory location storing the last acquired sample, i.e., the sample at current time *t*. The last term is indicated in red.

#### 4.2. The Acquisition System

The DAQ of our custom EIS equipment is the 16-bit U2351A data acquisition board from Keysight. The setting we used to test the impedance emulator by acquiring  $V_{in}$ and  $V_{out}$  are summarized in Table 2. The application of the FFT and the computation of the impedance, as illustrated on Section 3.4.3, are performed in post-processing by using MATLAB. No windowing has been applied, since the sampling rates and the number of samples were chosen as to always acquire an integer number of periods of each sinusoidal component [11]. The DAQ board also mounts a DAC that we used to pilot the voltage-controlled current pump in order to generate the multisine excitation signal. The amplitude of each current component was 50 mA. The shunt and load resistor values were  $R_{shunt} = 200 \text{ m}\Omega$ , and  $R_{load} = 5 \Omega$ .

In order to check the reproducibility of the results, we also tested the emulator by using a bench waveform generator to generate  $V_{in}$ , and a bench oscilloscope to acquire  $V_{in}$  and  $V_{out}$ . Resistors  $R_{shunt}$  and  $R_{load}$  were not used in this phase. The settings of the oscilloscope are reported in Table 2. The clocks of the two instruments were synchronized by wiring the oscilloscope external-clock input connector to the clock output connector of the waveform generator.

After some preliminary measurements, and by following the analysis presented in Section 3.4.4, we decided to measure the emulated impedance in two separate stages, one for the lower and one for the higher frequencies. Indeed, at lower frequencies, a longer acquisition time is needed in order to acquire several periods of the slower sinusoidal components, hence, the sampling rate  $F_{aq} = 10$  kSa/s was used in order to limit the total number of samples. At frequencies greater than 20 Hz, we instead used the sampling rate  $F_{aq} = 100$  kSa/s, in order to avoid phase fluctuations of the kind shown in Figure 8. The results were then combined into a unique impedance curve over the whole frequency interval 0.1–400 Hz.

As a last point, in simulations,  $V_{out}$  was treated as it were generated instantly as  $V_{in}$  was acquired by the ADC of the emulator. Of course, this is not the case in the real system, that has instead a latency due to the time required to compute (2), introducing a delay  $T_c = 827 \ \mu s$  between  $V_{in}$  and  $V_{out}$ . Since the Laplace transform of a delayed function is simply:

$$\mathcal{L}[f(t - T_c)](s) = \exp(-sT_c)\mathcal{L}[f(t)](s), \tag{11}$$

the effect of the MCU latency on the impedance  $Z(f_k)$  is corrected by just multiplying  $V_{out}(f_k)$  by the phase factor  $\exp(i2\pi f_k T_c)$ .

# 5. Results and Discussion

## 5.1. Results Obtained on the Custom EIS Measurement System

The emulated impedance for the SOC 100%, as measured by means of the custom EIS equipment, is reported in Figure 10 for seven repeated measurements. It can immediately be noticed that, although the fluctuations due to random noise are small, there are, how-ever, considerable systematic phase distortions at higher frequencies. Points measured at frequencies greater then 100 Hz looks like complete outliers, while at smaller frequencies, it seems that a linear phase distortion  $\exp(-i2\pi f_k T_d)$  is present. This can be verified by applying the following calibration procedure:



**Figure 10.** The emulated impedance as measured by the custom EIS system for seven repeated measurements.

1. *Phase calibration*. Let us consider the column vector of the measured phases  $\varphi_m = [\varphi(f_0), \varphi(f_1), \ldots]_m^T$ , where the index *m* is used to indicate each one of the repeated measurements, and the vector defined by arranging all the  $\varphi_m$  in a single column,  $\varphi = [\varphi_1, \varphi_2, \ldots]^T$ . Let us also consider the analogous vector  $\varphi_0$  of the expected phases computed analitically. Finally, the column vector **f** of all the frequencies  $\{f\}_k$  repeated many times on a column as the number of repeated measurements (i.e., **f**,  $\varphi$  and  $\varphi_0$  have the same number of elements). The difference between the measured and

the expected phase can then be written as the following linear system in the single unknown  $T_d$ :

$$2\pi \mathbf{f} T_d = \boldsymbol{\varphi}_0 - \boldsymbol{\varphi}. \tag{12}$$

The least-squares solution is  $T_d = -29.7 \,\mu$ s. Only the frequencies up to 100 Hz were included in the equation system.

2. *Amplitude calibration*. Since the ADCs of the acquisition system and of the emulator are different, and can produce different results on equal signals, in general, the amplitude has to be calibrated. As above, let us consider the column vector of the measured amplitudes  $\mathbf{A}_m = [\mathbf{A}(f_0), \mathbf{A}(f_1), \ldots]_m^T$ , then again the single column arrangement  $\mathbf{A} = [\mathbf{A}_1, \mathbf{A}_2, \ldots]^T$ , and finally the vector  $\mathbf{A}_0$  of the expected amplitudes computed analytically. The amplitude correction is given by a calibration constant  $k_A$  determined by the following linear system:

$$\mathbf{A}k_A = \mathbf{A}_0. \tag{13}$$

The least-squares solution is  $k_A = 0.9938$ , very close to 1. Indeed, in this case, as it can be seen in Figure 10, the amplitudes were already well matching the expected curve even without calibration.

By applying the calibration procedure, the corrected results shown in Figure 11 were obtained. The good agreement of the measured and expected curves up to 100 Hz proves that the hypothesis of a linear phase distortion was right. Nevertheless, the non-linear systematic phase distortion above 100 Hz remains. By keeping the values of  $T_d$  and  $k_A$  just computed, and measuring the emulated impedance for the SOC 20%, again a good matching is obtained, as shown in Figure 12. This suggests that the calibration is linked to the EIS system, and does not depend on the response programmed in the filter. In any case, the origin of such a strong phase distortion is not clear. In order to verify that it is due to the EIS system, and not to the emulator, we performed a set of measurements with a bench oscilloscope.



**Figure 11.** The emulated impedance as measured by the custom EIS measurement system after the calibration illustrated in Section 5.1.



**Figure 12.** The emulated impedance as measured by the custom EIS measurement system after the calibration illustrated in Section 5.1.

### Results Obtained on the Oscilloscope

The emulated SOC 100% was measured by means of the oscilloscope, and the same calibration procedure described in Section 5.1 was applied, obtaining  $T_d = -3 \mu s$ , and  $k_A = 1.02$ . The results are reported for two different numbers of samples in Figures 13 and 14, showing a good agreement between measured and expected curves. The slight phase distortion is within the time resolution set on the oscilloscope ( $T_{aq} = 10 \mu s$  at higher frequencies), and might be due to the fact that there is no clock synchronization between the oscilloscope and the impedance emulator. The mean and standard deviation over five repeated measurements of the amplitude and phase of the emulated impedance are reported in Figures 15 and 16. As expected, the error is lower for a greater number of samples. Systematic distortions are at any rate present also in the results obtained with the oscilloscope. However, the phase distortion above 100 Hz in this case is much smaller than that observed with the custom EIS system in Figures 13 and 14. This indicates that the distortion at high frequency is due to the EIS system, and not to the emulator.



**Figure 13.** The emulated impedance as measured by means of a bench oscilloscope after the calibration illustrated in Section 5.1.



**Figure 14.** The emulated impedance as measured by means of a bench oscilloscope after the calibration illustrated in Section 5.1.



Figure 15. Mean and standard deviation over five repeated measurement of the emulated impedance.



Figure 16. Mean and standard deviation over five repeated measurement of the emulated impedance.

## 6. Conclusions

We presented a method to emulate the impedance of a battery by means of a digital filter. All the details of the design are described. A low-cost prototype of the impedance emulator has been implemented and tested. A good agreement between the programmed and measured impedance was obtained when the emulator was tested by means of a bench oscilloscope. A relevant non-linear phase distortion was instead observed when the emulator was tested on a custom EIS measurement system. Such distortion is clearly linked with the EIS equipment and will need further investigation. Additionally, other excitation signals (e.g., binary sequences [10]) still need to be investigated on the emulator.

Overall, however, results show the feasibility of using the proposed emulator as a fully controllable and low-cost reference for calibrating battery impedance measurement systems. An immediate continuation of this work will be the implementation of the emulator on a better performing MCU or on a single-board computer, in order to increase both the sampling rate  $F_s$  and the number of samples  $N_s$ . This should allow emulation of the impedance at frequencies higher than 400 Hz, at the same time reducing the phase distortion.

**Author Contributions:** Conceptualization, F.S., A.D.A., A.M., P.C.; software and simulations, F.S.; development of the measuring system and experiments; A.D.A., F.S.; writing—original draft preparation, F.S.; writing—review and editing, F.S., A.D.A., A.M., P.C. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded in part by the University of Perugia grant "Identificazione e caratterizzazione accurata di sistemi e segnali, Ricerca di Base," 2019 and 2020.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

**Data Availability Statement:** The data presented in this study are available on request from the corresponding author. The data are not publicly available because they have not been ordered and stored in a clear and manageable form.

Conflicts of Interest: The authors declare no conflict of interest.

#### Appendix A. Sampling the DAC Signal

First, we briefly review some basic notions on ZOH signals. Let X(f) be the Fourier transform of any continuous time (c.t.) signal x(t), and  $X_{zoh}(f)$  the Fourier transform of its respective c.t. ZOH signal  $x_{zoh}(t)$ , sampled with period T. It is also assumed that the sampling rate F = 1/T and the bandwidth of the c.t. signal satisfy the Nyquist criterion. Hence, the following relation holds [23]:

$$X_{zoh}(f) = X(f) \exp(-i\pi fT) \operatorname{sinc}(fT), \tag{A1}$$

i.e., the ZOH signal has a delay of T/2 with respect to the original signal x(t), and also its amplitude is distorted by the sinc(fT). In general, given the ZOH spectrum  $X_{zoh}(f)$ , the spectrum of the original signal X(f) can be reconstructed by inverting (A1).

From now on, we always assume that the frequencies of all the spectral components, the sampling rates, and the duration of the acquisition window, are chosen in order to always acquire an integer number of cycles for each component, and that an integer number of samples is acquired at each cycle. Thus, the spectral components of the DFT with rectangular windowing, divided by the number of samples, are the same as the spectral components of the c.t. signals.

The DAC signal of Figure 6 is a c.t. ZOH with period  $T_s$ , generated from the sequence  $V_{out}(nT_s)$  as computed by the FIR filter. Let us indicate as  $V_{out}(nT_{aq})$  the sequence obtained by sampling the DAC signal with sampling rate  $F_{aq} = 1/T_{aq} = 10$  kHz (the EIS samples of Figure 6). To calculate the impedance, we need to calculate DFT[ $V_{out}(nT_s)$ ](f), where the sequence  $V_{out}(nT_s)$  could be simply obtained by decimating  $V_{out}(nT_{aq})$ . However, we want to keep all the samples, since, in general, this has an averaging effect on the noise,

thus producing better results. We then derive a different method. One might think that (A1) could be inverted as

$$\frac{1}{N_s} \text{DFT}[V_{out}(nT_s)](f) = \frac{1}{N_{aq}} \text{DFT}[V_{out}(nT_{aq})](f) \frac{\exp(i\pi fT_s)}{\operatorname{sinc}(fT_s)},$$
(A2)

where  $N_{aq}$  is the number of acquired samples, while  $N_s = N_{aq}T_{aq}/T_s$  is the decimated number of samples. Although it is correct to identify  $X(f) \equiv \text{DFT}[V_{out}(nT_s)](f)/N_s$ , this inversion does not yield correct results. Indeed,  $\text{DFT}[V_{out}(nT_{aq})](f)/N_{aq}$  cannot be identified with the spectrum  $X_{zoh}(f)$  of the c.t. ZOH signal of the DAC, since  $X_{zoh}(f)$  is not band-limited. The two spectra might be considered the same when  $T_{aq} \ll T_s$ , and (A2) would thus be correct to a very good approximation. In any case, as we will see in the following, the problem has an exact solution, and it is not necessary to use an approximation, whatever the values of  $T_{aa}$  and  $T_s$  might be.

Intuitively, the solution can be derived as follows. The DAC output can be viewed as a c.t. ZOH signal with period  $T_{aq}$  and constant for N periods, with  $N = T_s/T_{aq} = N_{aq}/N_s$ . Hence, in accordance with the general Formula (A1), its Fourier transform can be written as:

$$X_{zoh}(f) = \frac{1}{N_{aq}} \text{DFT}\left[V_{out}(nT_{aq})\right](f) \exp\left(-i\pi f T_{aq}\right) \text{sinc}\left(fT_{aq}\right).$$
(A3)

Since, by construction, the c.t. ZOH signal with period  $T_{aq}$  is the same as the c.t. ZOH signal with period  $T_s$ , their spectrum is the same, hence, we can invert (A1) as:

$$\frac{1}{N_s} \text{DFT}[V_{out}(nT_s)](f) = X_{zoh}(f) \frac{\exp(i\pi fT_s)}{\operatorname{sinc}(fT_s)} = = \frac{1}{N_{aq}} \text{DFT}[V_{out}(nT_{aq})](f) \exp[i\pi f(T_s - T_{aq})] \frac{\operatorname{sinc}(fT_{aq})}{\operatorname{sinc}(fT_s)}$$
(A4)

that is the same as the correction (10) that we applied in Section 3.4.3. Let us now derive the same result more formally.

For the sake of brevity, we define  $X_s(f) \equiv \text{DFT}[V_{out}(nT_s)](f)$ , and  $X_{aq}(f) \equiv \text{DFT}[V_{out}(nT_{aq})](f)$ . Let us also define the zero-padded sequence:

$$V_{out}^{zp}(mT_{aq}) = \begin{cases} V_{out}(nT_s) & m = nN \\ 0 & \text{otherwise} \end{cases}$$
(A5)

and let  $X_{aq}^{zp}(f)$  be its DFT. It is a known result in the theory of signal processing that  $X_{aq}^{zp}(f) = X_s(f)$  [23]. The sequence  $V_{out}(nT_{aq})$  can be rewritten as a sum of translated zero-padded sequences:

$$V_{out}(mT_{aq}) = \sum_{k=0}^{N-1} V_{out}^{zp} [(m-k)T_{aq}],$$
(A6)

and its DFT can thus be expressed as:

$$X_{aq}(f) = X_s(f) \sum_{k=0}^{N-1} \exp(-i2\pi f k T_{aq}).$$
 (A7)

Clearly, since  $X_s(f)$  is band-limited, so it is  $X_{aq}(f)$ . The last summation is also a known result, it being the DFT of the discrete-time rectangular function with *N* samples:

$$\sum_{k=0}^{N-1} \exp\left(-i2\pi f k T_{aq}\right) = \exp\left[-i\pi f (N-1)T_{aq}\right] \frac{\sin\left(N\pi f T_{aq}\right)}{\sin\left(\pi f T_{aq}\right)} = = \exp\left[-i\pi f \left(T_s - T_{aq}\right)\right] \frac{N_{aq}}{N_s} \frac{\operatorname{sinc}(f T_s)}{\operatorname{sinc}(f T_{aq})}$$
(A8)

Thus, finally:

$$\frac{1}{N_s}X_s(f) = \frac{1}{N_{aq}}X_{aq}(f)\exp\left[i\pi f\left(T_s - T_{aq}\right)\right]\frac{\operatorname{sinc}(fT_{aq})}{\operatorname{sinc}(fT_s)},\tag{A9}$$

i.e., the same result is obtained as that in (A4) and (10).

## References

- Tran, M.-K.; Fowler, M. A Review of Lithium-Ion Battery Fault Diagnostic Algorithms: Current Progress and Future Challenges. *Algorithms* 2020, 13, 62. [CrossRef]
- 2. Guha, A.; Patra, A. Online estimation of the electrochemical impedance spectrum and remaining useful life of lithium-ion batteries. *IEEE Trans. Instrum. Meas.* **2018**, 67, 1836–1849. [CrossRef]
- 3. Babaeiyazdi, I.; Rezaei-Zare, A.; Shokrzadeh, S. State of charge prediction of EV Li-ion batteries using EIS: A machine learning approach. *Energy* **2021**, 223, 120116. [CrossRef]
- 4. Galeotti, M.; Cinà, L.; Giammanco, C.; Cordiner, S.; Di Carlo, A. Performance analysis and SOH (state of health) evaluation of lithium polymer batteries through electrochemical impedance spectroscopy. *Energy* **2015**, *89*, 678–686. [CrossRef]
- 5. Koleti, U.R.; Dinh, T.Q.; Marco, J. A new on-line method for lithium plating detection in lithium-ion batteries. *J. Power Sources* **2020**, 451, 227798. [CrossRef]
- 6. Wang, X.;Wei, X.; Zhu, J.; Dai, H.; Zheng, Y.; Xu, X.; Chen, Q. A Review of Modeling, Acquisition, and Application of Lithium-ion Battery Impedance for Onboard Battery Management. *eTransportation* **2020**, *7*, 100093. [CrossRef]
- Komsiyska, L.; Buchberger, T.; Diehl, S.; Ehrensberger, M.; Hanzl, C.; Hartmann, C.; Hölzle, M.; Kleiner, J.; Lewerenz, M.; Liebhart, B.; et al. Critical Review of Intelligent Battery Systems: Challenges, Implementation, and Potential for Electric Vehicles. *Energies* 2021, 14, 5989. [CrossRef]
- Gong, Z.; Liu, Z.; Wang, Y.; Gupta, K.; da Silva, C.; Liu, T.; Zheng, Z.H.; Zhang, W.P.; van Lammeren, J.P.M.; Bergveld, H.J.; et al. IC for online EIS in automotive batteries and hybrid architecture for high-current perturbation in low-impedance cells. In Proceedings of the APEC 2018, San Antonio, TX, USA, 4–8 March 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1922–1929. [CrossRef]
- Carbonnier, H.; Barde, H.; Riga, L.; Carre, A. Electrochemical Impedance Spectroscopy for Online Satellite Battery Monitoring Using Square Wave Excitation. In Proceedings of the 2019 European Space Power Conference (ESPC), Côte d'Azur, France, 30 September–4 October 2019; pp. 1–6. [CrossRef]
- Crescentini, M.; De Angelis, A.; Ramilli, R.; De Angelis, G.; Tartagni, M.; Moschitta, A.; Traverso, P.A.; Carbone, P. Online EIS and Diagnostics on Lithium-Ion Batteries by Means of Low-Power Integrated Sensing and Parametric Modeling. *IEEE Trans. Instrum. Meas.* 2021, 70, 1–11. [CrossRef]
- De Angelis, A.; Buchicchio, E.; Santoni, F.; Moschitta, A.; Carbone, P. Practical Broadband Measurement of Battery EIS. In Proceedings of the 2021 IEEE International Workshop on Metrology for Automotive (MetroAutomotive), Bologna, Italy, 1–2 July 2021; pp. 25–29. [CrossRef]
- 12. Olarte, J.; Martínez de Ilarduya, J.; Zulueta, E.; Ferret, R.; Fernández-Gámiz, U.; Lopez-Guede, J.M. A Battery Management System with EIS Monitoring of Life Expectancy for Lead–Acid Batteries. *Electronics* **2021**, 10, 1228. [CrossRef]
- Allen, J. Simulation and Test Systems for Validation of Electric Drive and Battery Management Systems. In Proceedings of the SAE Technical Paper, SAE 2012 Aerospace Electronics and Avionics Systems Conference, Phoenix, AZ, USA, 30 October–1 November 2012. [CrossRef]
- 14. Fleischer, C.; Sauer, D.U.; Barreras, J.V.; Schaltz, E.; Christensen, A.E. Development of software and strategies for Battery Management System testing on HIL simulator. In Proceedings of the 2016 Eleventh International Conference on Ecological Vehicles and Renewable Energies (EVER), Monte Carlo, Monaco, 6–8 April 2016; pp. 1–12. [CrossRef]
- 15. Huesca-Nieto, E.A.; Villa-Villaseñor, N.; López-Hernández, J. Arbitrary Waveform Generation Based on Direct Digital Synthesis for Automotive Battery Voltage Transient Simulation. In Proceedings of the 2020 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC), Ixtapa, Mexico, 4–6 November 2020; pp. 1–6. [CrossRef]
- Bui, T.M.N.; Niri, M.F.; Worwood, D.; Dinh, T.Q.; Marco, J. An Advanced Hardware-in-the-Loop Battery Simulation Platform for the Experimental Testing of Battery Management System. In Proceedings of the 2019 23rd International Conference on Mechatronics Technology (ICMT), Salerno, Italy, 23–26 October 2019; pp. 1–6. [CrossRef]
- Barreras, J.V.; Swierczynski, M.; Schaltz, E.; Andreasen, S.J.; Fleisher, C.; Sauer, D.U.; Elkær, A. Functional analysis of Battery Management Systems using multi-cell HIL simulator. In Proceedings of the 2015 Tenth International Conference on Ecological Vehicles and Renewable Energies (EVER), 31 March–2 April 2015; pp. 1–10. [CrossRef]
- 18. Barreras, J.V.; Fleischer, C.; Christensen, A.E.; Swierczynski, M.; Schaltz, E.; Andreasen, S.J.; Sauer, D.U. An Advanced HIL Simulation Battery Model for Battery Management System Testing. *IEEE Trans. Ind. Appl.* 2016, 52, 5086–5099. [CrossRef]
- Buccolini, L.; Orcioni, S.; Longhi, S.; Conti, M. Cell Battery Emulator for Hardware-in-the-Loop BMS Test. In Proceedings of the 2018 IEEE International Conference on Environment and Electrical Engineering and 2018 IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe), Palermo, Italy, 12–15 June 2018; pp. 1–5. [CrossRef]

- 20. Vergori, E.; Mocera, F.; Somà, A. Battery Modelling and Simulation Using a Programmable Testing Equipment. *Computers* **2018**, 7, 20. [CrossRef]
- 21. Di Rienzo, R.; Roncella, R.; Morello, R.; Baronti, F.; Saletti, R. Low-cost modular battery emulator for battery management system testing. In Proceedings of the 2018 IEEE International Conference on Industrial Electronics for Sustainable Energy Systems (IESES), Hamilton, New Zealand, 31 January–2 February 2018. [CrossRef]
- 22. Overney, F.; Jeanneret, B. Calibration of an *LCR*-Meter at Arbitrary Phase Angles Using a Fully Automated Impedance Simulator. *Proc. IEEE Trans. Instrum. Meas.* **2017**, *66*, 1516–1523. doi: 10.1109/TIM.2017.2652500. [CrossRef]
- 23. Haykin, S.; Van Veen, B. Signals and Systems, 2nd ed.; John Wiley & Sons: Hoboken, NJ, USA, 2002.
- 24. Santoni, F.; De Angelis, A.; Moschitta, A.; Carbone, P. Analysis of the Uncertainty of EIS Battery Data Fitting to an Equivalent Circuit Model. In Proceedings of the 2021 IEEE 6th International forum on Research and Technology for Society and Industry (RTSI), Online Forum, 6–9 September 2021.
- 25. Vinagre, B.M.; Podlubny, I.; Hernández, A.; Feliu, V. Some approximations of fractional order operators used in control theory and applications. *Fract. Calc. Appl. Anal.* 2000, *3*, 231–248.