IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 24, 2021, accepted April 13, 2021, date of publication April 20, 2021, date of current version May 3, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3074269

Design and Validation of a Portable AD5933–Based Impedance Analyzer for Smart Agriculture

PIETRO IBBA[1], (Student Member, IEEE), MARCO CREPALDI[2], (Member, IEEE), GIUSEPPE CANTARELLA[1], (Member, IEEE), GIORGIO ZINI[2], ALESSANDRO BARCELLONA[2], MARIA RIVOLA[1], MATTIA PETRELLI[1], (Student Member, IEEE), LUISA PETTI[1], (Member, IEEE), AND PAOLO LUGLI[1], (Fellow, IEEE)

[1]Faculty of Science and Technology, Free University of Bozen-Bolzano, 39100 Bolzano, Italy
[2]Electronic Design Laboratory, Istituto Italiano di Tecnologia, 16152 Genoa, Italy

Corresponding authors: Pietro Ibba (pibba@unibz.it) and Luisa Petti (luisa.petti@unibz.it)

This work was supported by the Open Access Publishing Fund of the Free University of Bozen-Bolzano.

ABSTRACT This paper presents a portable impedance analyzer based on the AD5933 integrated circuit, specifically designed for fruit quality monitoring throughout its entire supply chain. The custom designed system allows to flexibly tune the parameters (i.e. excitation voltage, feedback resistor, settling time) for each AD5933 frequency range from 10 Hz to 100 kHz, leading to a precise system calibration and consequently, measurement. *FruitMeter* runs MicroPython on an STM32 microcontroller, allowing ease of development, advanced low-power mode, and both on-demand and scheduled measurements. The system employs Low-Energy Bluetooth and USB communication with a micro SD data storage which, together with its small size ($7.5 \times 2 \times 4$ cm in volume) and the 300 mAh rechargeable battery with USB recharge, increases the system portability and connectivity, critical for smart agriculture on-field applications. The developed portable system is validated by comparing its performance with a benchtop impedance analyzer with both passive components and fruit samples (apple, avocado, banana, mango and tomato). The comparison resulted in an average Root Mean Square Error (RMSE) of 1.0 % and 4.3 % for the passive components and of 1.36 % and 6.8 % for the fruit samples, for impedance magnitude and phase, respectively. We finally validated the system for a real use-case regarding the evaluation of banana aging evolution.

INDEX TERMS AD5933, EIS, portable impedance analyzer, custom design, fruit quality, smart agriculture.

I. INTRODUCTION

Electrical Impedance Spectroscopy (EIS) is based on the study of the interaction of the passive electrical properties of a sample with an AC current (or voltage), applied over a range of frequencies [1], [2]. The popularity of this technique has rapidly increased, as this method is nowadays used to characterize the electrical properties of a broad range of biological [3]–[7] and non-biological materials [8]–[10] for a wide range of different applications.

Conventionally, most of the EIS studies are performed in a laboratory environment, employing bulky and expensive benchtop impedance analyzers. Nevertheless, the emerging of novel applications in which measurements need to be carried out directly on field, allowing a real-time decision making, calls for portable instruments. A possible way to carry out on-field impedance measurements is represented by the use of commercially available portable systems, such as the Metrohm DropSens, the WaveNano and the Palmsens, that allow a good measurement accuracy over a wide frequency range. However, such systems are expensive, limiting the access to a small number of end users and restricting thus the range of possible applications. To solve this issue, during the last decade many designs of compact and low-cost impedance analyzers have been proposed, implementing different system design approaches [11]–[13].

In this context, the most widely used low-cost option is represented by the AD5933 integrated circuit, a high precision impedance converter produced by Analog Devices [14]. Such chip can operate in the 10 Hz−100 kHz frequency range,

The associate editor coordinating the review of this manuscript and approving it for publication was Norbert Herencsar.

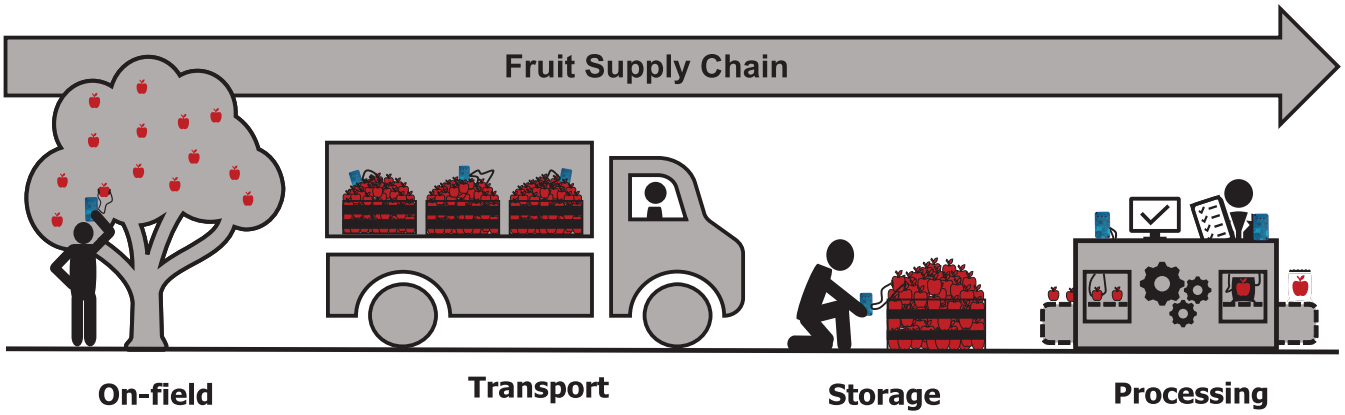


FIGURE 1. Fruit quality assessment applications covering the supply chain and made possible with the *FruitMeter*.

implementing the functionality of an impedance analyzer and offering also the possibility to be interfaced to a microcontroller through its I^2C bus, paving the way to hand-held, battery-operated devices [15]. Various designs of AD5933-based systems can be found in literature. Such systems were employed for biomedical [16], biological [17] and textile-embedded [18] applications, for the characterization of full body composition [19], saline solutions concentration [20], metal corrosion [21] and human skin bioimpedance [22]. Furthermore, some notable work proposed the extension of the use of such circuit for a four electrode bioimpedance measurement (1−100 kHz range) of bioimpedance in embedded applications [23]. Another interesting work shows how two AD5933 can be combined to monitor anti-corrosion coatings on infrastructures, using a 0.1 Hz−100 kHz frequency range, which allows measuring a wide span of impedance values from 10 Ω to 100 GΩ [24]. Regarding the application to natural food products, such as fruit samples, the AD5933 was implemented in various works [25]–[27], obtaining portable systems suitable for the detection of fruit quality. An exhaustive technical data review of some of the most representative above-mentioned portable impedance meters is presented in Sec. 6–Table 6.

Although well-established, such integrated circuit presents some drawbacks, mainly related to the calibration procedure, that need to be taken into account in the design of new systems, especially if accurate and reproducible measurements are required. In fact, the AD5933 calibration procedure is based on the use of a resistor having a load comparable to the sample-under-test impedance magnitude. However, in a wide frequency range, an unknown impedance may fall outside the single resistor calibration range, leading to the acquisition of noisy or, in the worst case scenario, incorrect impedance data. Possible workarounds are represented by either the reduction of the frequency spectrum bounds or by the division of the measurement, employing different calibrations to properly cover the impedance range. Furthermore, the integrated chip presents a fixed feedback resistance, that in relation to the excitation voltage, limits the lowest measurable impedance. Both issues need to be considered in the design of a new AD5933-based instrument, to fully exploit its potential and enhance its ease of use.

In this work, we present a handheld AD5933-based impedance analyzer and its applications for fruit quality monitoring, hereinafter called *FruitMeter* (FM). The proposed system features a highly configurable calibration procedure, allowing its application to the complete AD5933 frequency range (10 Hz – 100 KHz), that is divided into eight independent ranges. For each range, a wide flexibility and customization is possible by selecting the voltage, the feedback resistor, the Programmable Gain Amplifier (PGA) and the settling time (S_T), leading to the obtainment of a precise calibration, and thus a reliable measurement for a unique wide-band frequency sweep. The system also comprises three internal calibration resistors, Bluetooth and USB communication, microSD data storage and a 300 mAh battery, rechargeable through the USB battery charge module. Such system resulted to have a low error and a long battery life, all in a small volume ($7.5 \times 2 \times 4$ cm), extending the range of applicability and portability of the instrument. The MicroPython based firmware allows ease of development with intelligible code, advanced low-power mode, and both on-demand and scheduled measurements. Additionally, such firmware enables a smooth communication, via both Bluetooth and USB, with the custom-developed python-based application for smartphone and laptop, allowing a real time visualization of the collected bioimpedance spectrum.

FruitMeter is specifically thought for fruit quality monitoring applications throughout the whole supply chain (see Fig. 1), from the field to the processing, thanks to its high portability, long battery life and customizability. In this work, we present the design of the proposed system in Sec. II, from the design requirements to the system architecture, including the impedance loop and calibration. Sec. III describes FM firmware organization, including the main commands, the enabled functions and the battery consumption under different operating modes. Sec. IV presents system validation by comparing its performance to a state-of-the-art impedance

analyzer, with both passive components and fruit, and discussing possible weak points, such as measurement errors and battery life. Sec. V focuses on a real use-case of the system, employing the acquired bioimpedance data, fitted with the Cole equivalent circuit, for the evaluation of banana aging. Sec. VI provides a comprehensive comparison with reference to similar state-of-the-art AD5933-based impedance analyzers, also discussing possible future directions and further performance improvements. Finally, Sec. VII concludes the paper.

II. FruitMeter SYSTEM DESIGN

A. DESIGN REQUIREMENTS

Our main goals in the design of the system are twofold. First, the implementation of a high Technology Readiness Level (TRL) prototype that operates as a reliable laboratory instrument to enable repetitive qualitative measurements on fruit of different kind. Second, to pose the basis for a commercial tool which could be installed in the application domains specified in Fig. 1, after an aggressive miniaturization and a consequent re-design. To accomplish these goals, we have considered high level use-case requirements, which greatly depend on both the final application and on external aspects. As a general rule, when planning a fruit quality measurement campaign, it is important to consider the variability among sample batches, given primarily by environmental factors. In fact, growing conditions of fruit still attached to the plant can vary within meters and often within the same plant (depending on its size) due to different light exposure and soil nutrient status, while during transport and storage fruit are theoretically subject to uniform storage conditions [e.g. Temperature (T) or Relative Humidity (RH)], translating in the need for different measurement requirements [28]. The duration of fruit transport and storage depends on the type of fruit and on its perishability, which varies greatly between fruit. For example, apples are able to last up to 12 months at optimal storage conditions (4° C and 90-95 % RH), kiwifruits up to 5 months (0° C and 90-95 % RH) and bananas up to 14 weeks (14° C and 90-95 % RH) [29]. A crucial step in the fruit shelf-life, especially for fruit cultivated only in specific areas of the world, is represented by its transport. In fact, for fruit such as banana, transport is carried out by ship and last on average 14 days [30], during which the fruit needs to be closely monitored and maintained in its optimal storage condition to avoid wastes. Hence, in order to be effective in such diverse contexts, the portable instrumentation must be battery operated and be able to reliably and effectively sustain (i) hours of intensive on-field measurement, which commonly do not exceed the maximum 8 working hours, and (ii) days, or even months, of off-field punctual acquisition during transport and storage. For the latter cases, assuming constant storage conditions, the number of necessary measurements for fruit quality monitoring, depending on the case, can be reduced to a minimum of 1 per day, allowing to easily follow a long shipment (e.g. 14 days for banana) or a short storage period using a small sized battery. Nevertheless, a simple implementation of a larger battery can be easily considered in the future, to enable the long term monitoring of fruit such as apples and kiwifruits, which can be stored for months before commercialization.

Moreover, the acquired bioimpedance data need to be saved and logged correctly so that hundreds of measurement sessions can be easily identified for post-processing. Additionally, considering the necessity of running a calibration based on the type of sample under test, FM must enable the possibility to easily load different configurations, to reduce the time in between measurement requiring separate calibrations. Given the above applications, the device must provide high reliability and flexibility for the user, and in addition, ease of connectivity to both a laptop or smartphone to check ongoing measurements, hence reducing systematic errors. The form factor of the device shall be compact enough to enable easy movements in the field and the system shall provide enough non-volatile memory to save big quantities of measurements. Such memory, for ease of use, needs to be eventually removed from the device and connected to a personal computer to download measurement data.

In terms of connectivity to commercial personal computers or portable devices, the possible options are given by the transceivers normally present in commercial tablets or laptops, *i.e.* Bluetooth, Wi-Fi and LTE. While Wi-Fi and LTE provide high speed connectivity, Bluetooth provides lower data rates but with the advantage of improved power consumption compared to the former. Our application requires low power consumption, do not involve a very large data change and provides data that can be easily saved. For this reason, the use of transceiver would be focused on the transmission of commands or short data snippets to assess a measurement preview, therefore making Bluetooth radio an optimal candidate for our FM wireless communication. Indeed, the choice of the FM connectivity depends mainly on the adopted type of sequential measurement. With wireless connectivity the user can rapidly and frequently change the measurement parameters, at the expense of a larger memory use to sustain the code controlling of the Bluetooth radio. On the other hand, some repetitive measurement conditions require one-time calibration settings. For these reasons, the FM can be connected to a personal computer both using a standard wired port [e.g., an Universal Serial Bus (USB)] or based on a Bluetooth radio. While the former version can be intended to run more measurements with higher accuracy and explore the impendentiometric features of fruits, the latter is intended to be used as a final prototype to check the ongoing trends of the extracted features.

In terms of accuracy, the system shall be capable to run measurements with a minimum of 100 frequency points in a frequency space ranging 10 Hz – 100 kHz, with impedance magnitudes from kΩ to MΩ. Usually, in a battery powered portable device, it is not possible to use very high voltage headrooms for the measurements and to maintain ultra-low power at the same time. This is due to the fact that typical

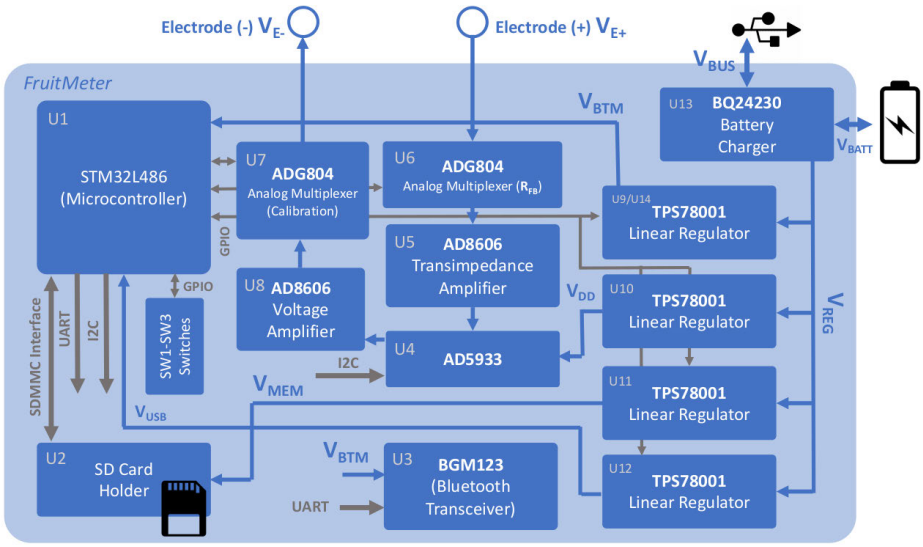


FIGURE 2. **MicroPython-based FM PCB block scheme with details of commercial components. Adapted from "FruitMeter: An AD5933-Based Portable Impedance Analyzer for Fruit Quality Characterization" by P. Ibba *et al.*, 2020 IEEE International Symposium on Circuits and Systems (ISCAS), Sevilla, 2020, pp. 1-5.**

commercial Li-poly cells provide a nominal voltage of 3.7 V. Because of this, a very high and a very low impedance cannot be measured using the same voltage excitation and amplification settings (e.g. the current obtained with 1 V across a 10 MΩ resistor is only 100 nA, while 1 kΩ results in a 1 mA current). Moreover, at low frequency biological tissues usually present a considerably high impedance magnitude, while at high frequency capacitive effects are predominant, making the impedance magnitude decrease. This implies that the device shall be capable of flexibly setting excitation and gains based on the current frequency range.

In terms of firmware development, we need to consider the following high-level requirements: i) FM must be reliable (*i.e* the system shall not stall in any case even in presence of a software bug, to avoid invalidating an on-field measurement session), ii) must be easy to use, and iii) software improvements and modifications must be fast and code shall be readable in an intelligible way to favor cooperation among people with different research background. Therefore, it is reasonable to use a rapid prototyping technique. Among notable ones, we identify Arduino™ and MicroPython™ as interesting alternatives to empower embedded systems. The former toolchain generates binary code only to be flashed in the device, while the software is typically written in a low-level C/C++ language. For this reason, we have chosen the latter thanks to its similarities to numerical computing tools such as MATLAB™. Moreover, Micropython interprets the code ran by a microcontroller and enables cross-compiling to improve performance. The impact of execution speed is not a particular issue for our application, as active measurements can last even few seconds. In our work we have favored flexibility and code intelligibility rather than performance.

B. ARCHITECTURE

Fig. 2 shows the *FruitMeter* architecture, which is based on a commercial AD5933 integrated impedance meter. The system provides complete AD5933 control, measurement and calibration file data save and Bluetooth connectivity based on MicroPython™ 1.11, which has been build from source on this particular Printed Circuit Board (PCB). FM provides three general purpose switches `SW1–SW3` that implement, respectively, device power-on, calibration and measurement without having to control the device remotely. `SW1` provides also a bootstrap to the main Micro-Controller Unit (MCU) voltage regulator `U9` to generate power supply V_{BTM} for both `U3` and the CPU. A total four voltage regulators, `U9–U12`, were used in the system to (i) isolate power supply for the AD5933 from both Bluetooth and Microcontroller thus providing a low noise supply for the impedance measurement IC, (ii) to enable selective power for the external SSD to limit power consumption when the system is not in use and (iii), to power up the Real Time Clock (RTC), hence enabling scheduled measurements. The latter, operating at 1.8 V, is not shown in Fig. 2.

FruitMeter is based on an STM32L486 Central Processing Unit (CPU) `U1`. A Bluetooth Low Energy (BLE) transceiver `U3` (BGM123) provides connectivity to a personal computer that uses a specific BLE key (*pyKey*) to enable remote control of the device. The BGM123 communicates to the MCU using a dedicated Universal Asynchronous Receiver and Transmitter (UART) port. To save measurement data, the system provides a microSD card slot `U2`, capable of handling GByte data using a FAT filesystem, to save subsequent measurements and calibration data, and it is connected to the MCU using a dedicated `SDMMC` interface, that can support GByte disk space. Considering that a single measurement

generates files of ≈6 kB, FM can store a very high amount of bioimpedance files before storage memory is full. To significantly reduce power consumption, the device includes several linear voltage regulators based on TPS78001 low-dropout chips that can be selectively activated. The real-time clock of the MCU is always powered on using a specific regulator U14 (in Fig. 2 collapsed in U9 for the sake of brevity). The system provides USB connectivity for both firmware update and battery charge (or text-based command interface for the USB version). Thanks to MicroPython, the former can be easily implemented based on a straightforward file copy to the internal MCU flash, once the device is connected to a Personal Computer (PC). The latter is implemented on the module using a BQ24230 battery charger (U13), which converts the 5 V USB voltage V_{BUS} (or the battery voltage V_{BATT}) to 4.6 V as supply voltage for the subsequent linear regulators and recharging of the battery at the same time. One of them, in particular U12, generates a 3.3 V V_{USB} stable supply for the USB-On-The-Go (OTG) sub-system provided by the STM32L486 MCU. The microSD card is powered using a dedicated V_{MEM} generated by a dedicated linear regulator U11. All the linear voltage regulators are controlled using General Purpose I/O (GPIO) of the microcontroller.

FruitMeter provides an impedance measurement loop comprising the AD5933 (U4, controlled through a dedicated I2C bus), two operational amplifiers in voltage/current amplifier configuration (U8 and U5), respectively, both based on an AD8606, and on two analog multiplexers U6 and U7, used to set either an internal calibration resistance (1% tolerance, directly soldered on the PCB) or the output external load on electrodes $V_{\mathrm{E+}}$ and $V_{\mathrm{E-}}$, and the feedback resistance of the transimpedance amplifier U5. The analog multiplexers, that provide 125 pF to-ground parasitic capacitance, are controlled using dedicated MCU GPIO. The system provides internal calibration resistors of 470 kΩ, 15 kΩ and 200 Ω. While it is connected to an external load the system provides a series resistance of 100 Ω to increase Electro-Static Discharge (ESD) performance, together with a TPD4E02B04, not shown for the sake of brevity. The selectable feedback resistors (1% tolerance) R_{FB} are 200 kΩ, 20 kΩ, 2 kΩ and 200 Ω. The Master Clock (MCLK as of the AD5933 datasheet) is directly synthesized by the MCU based on the measurement range. The supply voltage V_{DD} of the AD5933 has been set to 2.85 V to provide the largest headroom voltage to exploit battery power as much as possible, thus enabling extensive portability. The voltage amplification of U8 is set to $G_{\mathrm{V}} = 1.4$ V to recover the limited input range posed by the <3.3 V V_{DD} voltage supply.

C. IMPEDANCE LOOP

Fig. 3 shows a simplified schematic of the measurement loop implemented on the FM. This sub-system enables pre-amplification to adapt the voltage excitation across the sample and amplify the resulting current signal. Having a programmable amplification chain is important to match the diverse impedance magnitude ranges that, as stated above,

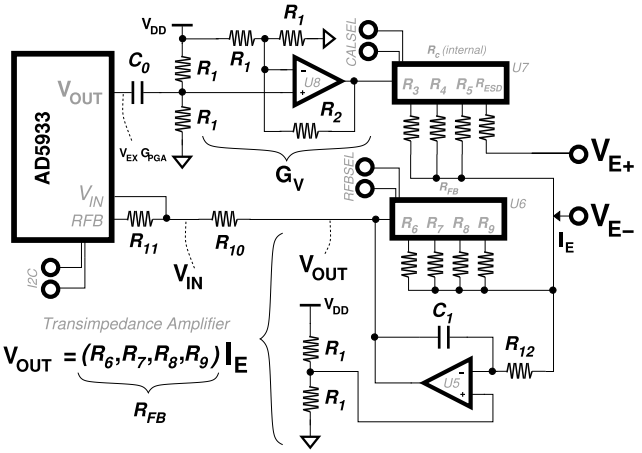


FIGURE 3. Simplified schematic of the impedance loop with analog multiplexers. Adapted from "FruitMeter: An AD5933-Based Portable Impedance Analyzer for Fruit Quality Characterization" by P. Ibba *et al.*, 2020 IEEE International Symposium on Circuits and Systems (ISCAS), Sevilla, 2020, pp. 1-5.

can range from kΩ to MΩ. After a DC block implemented using C_0, nominally 2.2 μF, the impedance loop provides voltage amplification through U8, with $R_1 = 110$ kΩ and $R_2 = 22$ kΩ. Specifically, the R_1 voltage divider operates in the system as virtual ground for the operational amplifiers, that is posing reference voltage at $V_{\mathrm{DD}}/2$. To enable ease of calibration we have included internal calibration resistors (surface mount technology) that can be selected at firmware level. This functionality enables a large degree of freedom in the fruit impedance exploration as the user can choose to select internal or external calibration based on the required measurement accuracy. The analog multiplexer U7 selects internal calibration resistors R_3–R_5 or toggles the signal to $V_{\mathrm{E+}}$ by including a series ESD resistor $R_{\mathrm{ESD}} = 100$ Ω (control signal CALSEL). The current resulting from the sample I_{E}, that returns from $V_{\mathrm{E-}}$, is amplified by the transimpedance amplifier. This is based on the OpAmp U5, with feedback resistors R_{FB} that can be selected using U6 (control signal RFBSEL) among the values R_6–R_9, i.e., 200 Ω–200 kΩ (the additional feedback is used to stabilize the amplifier for $R_{12} = 1$ kΩ and $C_1 = 6.8$ pF). Specifically, resistors R_6–R_9 are used as feedback resistors to the transmipedance amplifier U5 to implement a variable amplification. One end of the multiplexer U6 is connected to V_{out} while the other four terminals can be selected to be connected to V_{E-}. R_{11} and R_{10} have nominal value of 22 kΩ. The use of internal feedback resistors is fundamental to enable the variation of excitation parameters during the evaluation of a particular fruit. The AD5933, indeed, provides a Discrete Fourier Transform-based (DFT) impedance measurement method spanning a frequency range of 10 Hz–100 kHz, with the upper frequency point imposed by the AD5933 limitation. According to the datasheet [14], depending on the impedance to be identified, the system requires a specific calibration, which implies the careful control of the excitation voltage V_{EX} (1, 2, 0.4, and 0.2 V are

possible), the internal Programmable Gain Amplifier (PGA) amplification (G_{PGA}, 1 or 5), and the selection of a calibration resistance that needs to approximately match the unknown impedance (R_6–R_9 in the schematic are the internal calibration resistors). Due to internal implementation and to the concept of DFT, based on the desired frequency range, an appropriate `MCLK` frequency must be set. Calibration needs to be performed for each frequency range corresponding to a specific `MCLK` frequency. Based on the AD5933 evaluation board datasheet, it varies from 25 kHz to 16 MHz, in particular 25, 50, 100, 250, 500, 2000, 4000, 16000 kHz for frequency ranges 10–20, 20–30, 30–100, 100–200, 200–300, 300–1000, 1 k–5 k and 5 k–100 kHz. In all measurement conditions, the signal at the AD5933 input must match the 2.85 V power supply range and a linearity condition can be easily devised based on the above hypotheses. In all conditions to ensure linearity (during calibration as well) the following model must hold,

$$V_{IN} = G_{PGA} G_V V_{EX} \frac{R_{FB}}{R}, \tag{1}$$

where R is the unknown/calibration resistor. To ensure linearity, $V_{IN} < V_{DD}$, where $V_{DD} = 2.85$ V. The calibration resistance must be carefully selected based on the specific fruit to be measured. Examples on the fruit impedance ranges might be found here [31]–[33]. As a successful metric, we have implemented a selection of the parameters V_{EX} and R_{FB} (and keeping $G_{PGA} = 1$) for each frequency range so that $V_{EX} \sim 1$ V. Additional development will allow the compensation of the temperature, acquired with the AD5933, in the calibration algorithm. Observe that Eqn. (1) must be valid both during calibration and real measurements. In fact, during calibration the system calculates and saves a series of coefficients, defined as gain factor and phase factor, from a known calibration resistor load (R, either internal or external) which must have an impedance comparable to the one of the unknown sample. Consequently, if during real measurement the latter presents a highly different impedance value, Eqn. (1) is not valid, making the system operate out of its linear range, hence leading to a non accurate measurement.

D. CALIBRATION AND MEASUREMENT

Calibration is the process aimed at the determination of the system intrinsic Gain Factor (G_F) and Phase Factor (∇_F), assuming a known resistance load. The known load can be external to the *FruitMeter*, therefore connected directly to V_{E+} and V_{E-}, or based on an internal calibration resistance selected based on a `U7` setting. Differently from what suggested in the datasheet [14], to improve measurement accuracy, we implemented a complete multi-point calibration in which the complete frequency range 10 Hz–100 kHz is calibrated and data is saved on specific CSV files. Gain and phase factors are computed for each frequency point as $G_F = 1/(R_c\sqrt{r_c^2 + i_c^2})$ and $\nabla_F = \angle(r_c, i_c)$, respectively, where R_c is the external/internal calibration resistance, $\angle$ is complex numbers' phase, while r_c and i_c are the AD5933 register outputs (2's complement) referred to the calibration resistance. What needs to be taken into account is that calibration considers all the intrinsic non-idealities of the system, therefore the parasitic contribution of the measurement loop, in particular the analog multiplexers capacitance. Both terms are computed for each frequency point used afterwards for the measurement. In this way, calibration files can be segmented and multiple conditions can be collapsed into a single calibration file using different R_{FB}. This is done by means of a MATLAB script that loads the sample impedance, acquired with a benchtop system (or that calculates the theoretical value, in the case of passive component circuits), dividing it in 8 frequency ranges correspondent to the *FruitMeter* ones. Afterwards, it calculates the optimal combination of R_{FB} and V_{EX} for each range median magnitude to return a $V_{IN} \approx 1$, ideal to both respect the linearity conditions in Eqn. (1) and prevent a noisy output signal, given by a low V_{IN}. These files can be then uploaded in the system memory and easily recovered using a specific `LOADC` command through both a pyKey BLE connection (Bluetooth version) and using the USB emulated serial port (USB version). Measurements (that can be run only if calibration data has been recalled or newly computed) are determined as $|Z| = 1/(\sqrt{r^2 + i^2} G_F)$ and $\angle Z = \angle(r, i) - \nabla_F$, where r and i are the real and imaginary parts given by the AD5933 internal registers while measuring an unknown impedance.

III. FIRMWARE

The complete FM firmware has been developed using MicroPython and cross-compiled from source to enable memory save for storing a large number of measurements and calibration points (the USB version does not load the BLE driver and enables the storage of a larger number of points, overall 180 module and phase points). The Bluetooth version has a specific `bgm123.py` module that handles connection and implements an emulated wireless serial port based on two generic GATT characteristic. A parser module `upyParse.py` reads a text file in the μSD memory comprising a list of commands with their parameters. The use of parametric commands enables flexibility in the development and in the firmware modifications, as the list is maintained intelligible and readable when adding a command. After loading commands, the module detects the incoming USB/Bluetooth serial characters parses the incoming strings and executes commands. A custom `pyAD5933.py` module manages completely the AD5933, enabling the complete control of the device, including temperature measurements. This module allows also the introduction of a processing gain of the AD5933 (and therefore, noise reduction), through the possibility of running measurements and calibration assuming a determined (S_T) and multiplication factor. While setting a higher S_T improves noise performance, it increases measurement time (*i.e.* the scan rate on the full device frequency span). During each measurement session, the device sets on-the-fly a sequence of G_{PGA}, V_{EX}, S_T and R_{FB} values for each frequency range and adjusts the corresponding

MCLK frequency accordingly using an hardware timer. The pyAD5933.py module is also capable (through the os module) to write on the File Allocation Table (FAT) formatted filesystem of the SD card and save measurement data in a Comma Separated Value (CSV) format, also including all the measurement parameters previously set before issuing a measurement command.

Thanks to the Hardware Abstraction Level (HAL) drivers for this class of STM32 MCU ported to MicroPython, *Fruit-Meter* can enter into a scheduled measurement mode (later defined as SLEEPSCHEDULE), therefore running NT measurements every DEL minutes, while entering in an ultra-low power sleep mode and using the RTC for wake-up. As previously schematized in Fig. 2, the RTC dedicated power supply has been kept connected to the battery source, irrespective of the power state of the device. This way the microcontroller can schedule wake-up using this dedicated timer. Using this hardware-software technique, with the on-board 300 mAh Li-Po battery, we have successfully started and stopped the device after a ~ 54 h of scheduled measurements every 60 min, still with remaining charge. The use of MicroPython enables very high flexibility in the development and in the follow-up of the device, with the possibility to implement bug fixes and include other relevant updates by always maintaining an intelligible code and ease of firmware update. The system operation is found to be robust thanks to the use of a try–except speculative execution, allowing a straightforward handling of software bugs without leading to device crashes.

A. FIRMWARE ORGANIZATION

The firmware is organized in modules, coordinated by a main module which implements an infinite loop, and handles I/O push-buttons and physical transmission medium. The AD5933 comprises three separate modules, a driver that implements all the low level fabric, comprising data load and save from the flash FAT filesystem, a master clock MCLK frequency module that provides mapping of the measurement intervals, and a selector module to switch feedback resistors and internal calibration resistors. The external LEDs are managed using a separate module, which internally implements blinking and control using a dedicated hardware timer. The BGM123 is handled with a separate module which enables byte-level parsing through the UART port and is invoked periodically in the main loop to implement bus transactions (*i.e.* data read or command send), through a specific flush method. This allows to avoid using interrupt request (IRQ) call-backs or hardware timers, a beneficial feature for code-reuse, being aligned to the same coding strategy used by built-in Micropython USB Virtual COM Port (VCP) methods. By implementing an on-demand flushing method, the Bluetooth methods can be simply replaced by USB VCP commands, leaving *inter alia*, memory resources to the storage of impedance points without incurring in potential memory errors of callbacks due to code complexity.

The main module of the software comprises an infinite loop that checks periodically if a physical medium is present, and in that case checks if the VCP COM port or Bluetooth UART port (Bluetooth version) have a new string of characters ready to be read. If a new set of characters is detected, the system runs a dedicated and parametrizable parser (implemented as a separate module) to extract commands and parameters and to check their syntax. Based on the supplied command, decoded through an equivalent case statement, the corresponding operation is executed and a response through the same physical medium, wired or wireless is re-transmitted. All the commands return in an immediate or at least seconds-order completion time, including longer commands such as CALIBRATE and MEASURE operating within the complete frequency range, and set MCLK frequency (as specified next). The only command not immediately returning is the USB mode SLEEPSCHEDULE command, which enters ultra-low power mode and runs subsequent measurements in very long time spans. In this case the firmware enters a separate loop duty cycled by an RTC IRQ, lasting the number of scheduled measurements required. For the sake of brevity we do not report details on all modules, while we focus on the most important conceptual parts of the firmware regarding calibration and measurement using the AD5933, and the scheduled measurement command in ultra-low power mode.

B. CALIBRATION AND MEASUREMENT

Alg. 1 shows a simplified pseudocode for both calibration and measurement loop assuming an USB-only device. In case of the Bluetooth version, the pseudocode changes only in the settings of the excitation voltage and feedback resistance. To keep track of an effectively executed calibration, a dedicated flag calibrated is used (initialized as False). This is set when calibration is performed at least once and it is used to check whether gain and phase factors are available prior to running measurements, otherwise leading to the returning of an error message. Let us assume we are running calibration, therefore the flag calibrate is True. Calibration aims at determining the gain and phase factors G_F and $\angle_F$ for all the complete frequency range of the device assuming an unknown resistor R. The datasheet of the AD5933 forecasts two possible calibration loops, single point or two-point. In our implementation, in order to achieve the maximum possible accuracy and account for possible non-idealities, we consider G_F and $\angle_F$ for all the frequency points (multi-point calibration).

The AD5933 performs a Discrete Fourier Transform (DFT) on the acquired measurements and poses precise constraints in terms of MCLK frequency. The datasheet specifies, indeed, eight measurement intervals I_{SET} each with a specific clock frequency that, in our implementation, is synthesized by the MCU using a dedicated counter. Each master clock frequency ensures that within a particular frequency measurement range the DFT result is reliable. To enable increased flexibility, in our implementation each i-th frequency interval comprises a corresponding excitation voltage and feedback

Algorithm 1 Calibration and Measurement Loop Pseudocode

```
Result: G_F, Δ_F if calibrate is True – magnitude, phase if calibrate is False
if (calibrate is False) and (calibrated is False) then
    Return Error;
else if calibrate is True then
    Reset G_F, Δ_F;
else
    Reset magnitude, phase;
Reset M, ∠;
i = 0;
Send AD5933 RESET;
while i < I_SET do
    settings = get_interval_settings(i);
    set_clock(settings);
    set_excitation_voltage(settings);
    set_rfb(settings);
    data = run_impedance_measurement(settings);
    M, ∠ ← append(data);
    i++;
i = 0;
if calibrate is True then
    while i < length(M) do
        if Internal Resistor then
            G_F[i] = M[i]/R;
        else
            G_F[i] = M[i]/(R+R_ESD);
        Δ_F[i] = ∠[i];
        i++;
    calibrated = True;
else
    while i < length(M) do
        magnitude[i] = M[i]G_F[i];
        phase[i] = ∠[i] − Δ_F[i];
        i++;
```

resistance R_{FB}. Setting such parameters for all frequency intervals is essential to avoid incurring in saturation of the circuit, as the impedance magnitude of a fruit may significantly vary in the complete 10 Hz–100 kHz range, by up to several orders of magnitude.

When calibration is invoked, first G_F, $\angle_F$ and all magnitude and phase measurements from the AD5933 M and $\angle$, respectively, are reset. Next, a `AD5933 RESET` command is sent to the device so that to enforce a known state before starting its internal acquisition cycle. As previously specified, calibration must be run on a known impedance R that can be an on-PCB value (R_3–R_5) or externally connected to the device using an R_{ESD} protection resistor. Sequentially, for each i-th frequency set `settings` comprising maximum frequency, minimum frequency, the frequency increment, feedback resistance R_{FB} used and excitation voltage are acquired using a specified `get_interval_settings` command. The number of increments (*i.e.* the number of measurements within each interval) is fixed, but it can be changed by intervening at firmware-level. The frequency intervals (Hz unit) are 10–20, 20–30, 30–100, 100–200, 200–300, 300–1000, 1000–5000, 5000-100000, for a `MCLK` frequency of 25 kHz, 50 kHz, 100 kHz, 250 kHz, 1 MHz, 2 MHz, 4 MHz and 16 MHz, and for an increment of 2, 2, 5, 50, 50, 200, 250 and 5000 Hz, respectively. This results in a number of measurements points of 5, 5, 14, 2, 2, 3, 16 and 19, respectively. By default, the feedback resistor values are all 20 kΩ and the excitation voltage is 1 V, except for the highest frequency interval, in which it is 200 mV. These parameters can be changed before calibration by the user, using the dedicated commands for both Bluetooth and USB version. Next, `set_clock` and `set_excitation_voltage` sets the corresponding master clock and excitation voltage as a function of the previously acquired object `settings`. Afterwards, the feedback resistor R_{RB} is set through the function `set_rfb` according to the i-th interval. Within the limits defined by the object `settings`, a measurement cycle is run through `run_impedance_measurement`. The measurement output `data` is then appended to the M and $\angle$ vectors through a byte-to-`float` conversion, not reported here for the sake of brevity, but available on the datasheet.

When the above segmented measurement for each interval I_{SET} is completed, G_F and $\angle_F$ are computed for all the measurement points, according to the specifications of the datasheet. If R (the unknown reference resistor) is external, the gain factor is accounted for the additional R_{ESD} assuming no phase change, because the target impedance to be measured is in the worst case at least 1 order of magnitude larger compared to R_{ESD}.

After calibration, by assuming that the user does not change any of the operating conditions used to acquire calibration, a measurement session can be performed. In the conceptual pseudocode of Alg. 1 measurement is run if `calibrate` is `False` and at least calibration is run once (`calibrated` is `True`). The operation of the measurement loop is very similar compared to the calibration one, with the difference in the usage of the obtained values of M and $\angle$, corrected by G_F and Δ_F, respectively by multiplication and subtraction, to generate `magnitude` and `phase` vectors.

C. SCHEDULED MEASUREMENTS

Lst. 1 shows the Micropython listing that is run every time a new `SLEEPSCHEDULE` command is invoked. It is based on global variables `NT` and `DEL`, representing the number of measurements and the delay between each measurement, while the global variable `times` is used to track the

```
1  def sleep_schedule():
2     global times      # stores the current iterations
3     global NT     # SLEEPSCHEDULE <NT> <DEL>, <NT> parameter from the parser
4     global DEL      # SLEEPSCHEDULE <NT> <DEL>, <DEL> parameter from the parser
5     global gc     # garbage collector class
6     global cancelled    # global cancel variable
7     global extint   # physical key for interrupt
8     extint.enable()  # enables physical key interrupt
9     times = 0     # initializes the current measurements to zero
10    while True:
11       gc.collect()     # frees unused memory
12       times += 1     # keeps track of the number of measurements
13       if times <= NT:    # if not finished
14          rtc.wakeup(1000*60*DEL)  # ms units
15          cancelled = False
16          pyb.stop()             # cancelled is set on extint callback
17          if cancelled == True:
18             cancel_led_indication()      # specific LED indication to indicate canceling
19             extint.disable()       # disables interrupt
20             rtc.wakeup(None)       # sets no RTC callback
21             led1.blink()         # blinks LED1, waiting for commands
22             break              # returns to the main loop
23
24       else:
25           extint.disable()       # disables interrupt
26           rtc.wakeup(None)       # sets no RTC callback
27           led1.blink()         # blinks LED1, waiting for commands
28           break              # returns to the main loop
29       try:     # runs the measurement and saves impedance file
30           led1.stop()
31           led2.stop()
32           led3.stop()
33           dev.set_external()  # makes sure the mux is toggled to external
34           dev.measure()       # runs measurement
35           dev.save_imp_file()    # saves impedance file
36       except:    # error handling
37           extint.disable()     # disables interrupt
38           rtc.wakeup(None)     # sets no RTC callback
39           led1.blink()       # blinks LED1, waiting for commands
40           break           # returns to the main loop
```

List. 1. `SLEEPSCHEDULE` **python code.**

i-th measurement. The module references also other global classes and variables useful for its operation, in particular `gc` (Garbage Collector), `extint` the physical interrupt pin to cancel the execution of `SLEEPSCHEDULE`, and a global variable `cancelled` which carries information on the pressing of the physical key. Classes `led1–led3` are referred to physical LED, `rtc` to RTC, `dev` to the AD5933 device and `extint` to the physical pin used to cancel operation. The subsequent measurements are organized in a `while` loop that is interrupted based on various conditions. First, the `gc` is called to clear unused memory and avoid memory errors, given the requirement of saving the large number of floating point numbers acquired with measurements in the next steps. If `times <= NT`, the system is sent to ultra-low power sleep mode by (i) programming the real time clock (`rtc`) indicating the amount of time before the next wake up (millisecond units), (ii) setting the `cancelled` flag to `False` and (iii) by stopping the Python Board with `pyb.stop()`, which poses the module in ultra-low power mode. At this point, the MCU can be woken up by two different events, (i) a physical keypress that sets, through a dedicated routine the `cancelled` flag to `True` or (ii) by the Real-Time Clock (RTC). When (i) occurs the system performs a flash pattern on all LED to indicate interruption through calling `cancel_led_indication()` disabling the interrupt on the physical key (through `extint.disable()`) to restore the normal management of the key. Next, it sets no wakeup for the RTC, restores LEDs `led1` to normal blinking to indicate the possibility to accept commands, and exits the `while` loop with `break`. The same command sequence (except for a dedicated cancel indication on the LEDs) is performed when the desired number of iterations is reached (i.e., `times > NTIMES`). If none of the above conditions holds, once woken up, the system runs a single measurements using speculative execution guards `try–except`, to enable the handing of errors and not to stop program execution. First, on lines 30–32 it stops any running LED, it sets the multiplexer `U7` to external path (therefore with R_{ESD} in series, `set_external` method), runs a measurement with `measure()` and saves the obtained values in an impedance file on line 35. In case of internal errors, the device needs to return to the main loop enabling the intervention of the user, therefore on lines 37–40, the same termination commands used for the previous canceling or loop normal termination are used.

D. COMMANDS AND PC INTERFACE

FruitMeter accepts a series of commands through the Command Line Interface (CLI) via both Bluetooth and USB connection. Each command provides an identifier and some

TABLE 1. ***FruitMeter*** **main commands.**

Command	Description	Parameters
SETCF <Name>	Sets a file <Name> for calibration file.	A string of characters {A...Z,0...9,_,-}
CALIBRATE	Runs calibration with previously defined settings.	–
SAVECF	Saves the current calibration data on a calibration file with name specified using the command SETCF.	–
SETIF <Name>	Sets a file <Name> for impedance files.	A string of characters {A...Z,0...9,_,-}
MEASURE	Runs a single measurement with previously defined settings.	–
SAVEIF	Saves the current impedance data on an impedance file with name specified using SETIF.	–
INTCAL <RES>	Sets the internal calibration resistance <RES> and toggles the multiplexer to an internal load.	<RES>: 200, 15000 or 470000 (Ω)
LOADC <Filename>	Loads a calibration file with name <Filename> on the SD card.	–
EXTCAL <RES>	Sets the calibration resistance value <RES> and toggles the multiplexer to an external load.	<RES>: an integer number that identifies the external calibration resistance (Ω)
EXTERNAL	Toggles the multiplexer to an external load.	–
SETST <N> <F>	Sets the settling time <N> and multiplication factor <F>.	<N>: a number from 0 to 255. <F>: a number from 0 to 255.
SETPGA <P>	Sets the PGA gain factor <P>.	<P>: 1 or 5.
DUMP	Dumps all current measurement data in an ASCII format line-by-line. This command is useful to generate a real-time plot of the current measurement.	–
POWEROFF	Shuts the system down.	–
Bluetooth Version Only		
SETEXV <V>	Sets the excitation voltage <V> for all the frequency measurement intervals.	<V>: an integer number from 1 to 4 (1 = 1 V, 2 = 2 V, 3 = 0.4 V and 4 = 0.2 V)
SETEXVH <VH>	Sets the excitation voltage <VH> for the highest frequency measurement interval 5 kHz–100 kHz.	<VH>: an integer number from 1 to 4 (1 = 1 V, 2 = 2 V, 3 = 0.4 V and 4 = 0.2 V)
SETRFB <RES>	Sets the internal feedback resistance for the current to voltage amplifier.	<RES>: 200, 2000, 20000 or 200000 (Ω)
USB Version Only		
SETEXV <V1> ... <V8>	Sets the excitation voltage for each frequency measurement interval (<V1> ... <V8>).	<V1> ... <V8>: Excitation voltage, an integer number from 1 to 4 (1 = 1 V, 2 = 2 V, 3 = 0.4 V and 4 = 0.2 V)
SETRFB <R1> ... <R8>	Sets the internal feedback resistance for the current to voltage amplifier for each frequency interval (<R1> ... <R8>).	<R1> ... <R8>: 200, 2000, 20000 or 200000 (Ω)
SLEEPSCHEDULE <NT> <DEL>	Starts subsequent measurements <NT> with a certain period <DEL> using an RTC interrupt based timer.	<NT>: an integer number. <DEL>: an integer number identifying the delay in minutes between each consecutive measurement.

parameters, separated through a character, compactly,

$$\texttt{CMD} < \texttt{P1} >< \texttt{P2} >< \texttt{P3} > \ldots < \texttt{PN} >$$

where CMD is the command and <P1>–<PN> are the CMD specific parameters. The list of the main commands, together with their function and main parameters is presented in Tab. 1. The two versions, Bluetooth and USB, have mostly the same commands, differing only in the choice of the feedback resistors (R_{FB}) and excitation voltage (V_{EX}). In fact, compared to the Bluetooth version, the USB one does not load internal firmware on Bluetooth and thus the STM32L486 microprocessor has more RAM memory available. Consequently, the USB version frequency space table includes more measurement points compared to Bluetooth, and different feedback resistors and excitation voltages can be applied for each of the 8 frequency range using the SETEXV and SETRFB commands. For the Bluetooth version, the memory RAM shortage allowed the setting of only one R_{FB} and V_{EX} for the entire frequency range. This might limit the capability of the system to correctly evaluate an unknown impedance, particularly for samples with large magnitude differences between the high and low frequency points. To partially solve this issue, the Bluetooth version implements the SETEXVH command, which allow the user to set a different V_{EX} for the high frequency range interval (5 kHz — 100 kHz), where the bioimpedance tends to be low. Furthermore, the USB version implements a SLEEPSCHEDULE command, which starts subsequent measurements with a certain period using an RTC interrupt–based timer to enable ultra-low power consumption between measurements, extending the system battery life. This command was thought for both an employment in the monitoring of fruit during transport, using the battery, and for the monitoring of longer processes powering the system via USB, virtually until the saturation of the available memory.

In terms of PC connectivity we have implemented using Python a command line interface console (running under Windows, Linux and MacOS) with which the user can send the above command by using the USB serial port (USB version only) or a fully custom-designed Bluetooth dongle, that provides an emulated USB serial port. The dongle runs MicroPython as well and interfaces the same BGM123 Bluetooth chipset used in the *FruitMeter*. The PC console provides also a macro _PLOT command which is capable of plotting real-time the measurement data using matplotlib.

E. CONTINUOUS PLOT MODE

The console software provides a continuous plot mode for debugging and demonstration purposes. The _PLOT command provides a magnitude and phase plot, strictly following device calibration, by using the measure button on the device. Once plot mode is entered, the console cyclically waits for each subsequent measurement to displays the result

FIGURE 4. Typical impedance magnitude and phase plots obtained in a continuous plot mode measurement for a banana, for both the USB version (on the laptop) and the BLE version (on the smartphone).

TABLE 2. *FruitMeter* current consumption from the LiPo battery terminals for both BT and USB versions, under different operating conditions.

Bluetooth™ version	
Mode	**Current Consumption (mA)**
BT Connected/Accepting Commands	28.2
BT Connected/Measuring/Calibrating	40.3
USB version	
Mode	**Current Consumption (mA)**
USB Connected/Accepting Commands	26
USB Connected/Measuring/Calibrating	35.3
`SLEEPSCHEDULE` Measurement$^{\diamond}$	$I_{\text{off}} = 2.5$ $I_{\text{on}} = 35.3$ $t_{\text{m}} = 14.4$ s

$\diamond$ = Measurement run under the default set-up conditions.

in magnitude and phase form, as shown in Fig. 4. To demonstrate ease of connectivity with commercial portable devices, a similar continuous plot mechanism is implemented also in a iOS 12 demo application (not shown here for the sake of brevity) which connects to the *FruitMeter* using an iPhone 8 built-in BT 5.0 transceiver. This enables, *inter alia*, to verify the compatibility of the internal BT 4.2 with a most recent BT 5.0 transceiver.

F. POWER CONSUMPTION

Tab. 2 shows the current consumption of *FruitMeter* under different operating conditions, acquired using a Tektronix TCP0030 current probe combined with a Tektronix MSO4104 oscilloscope. While connected through Bluetooth, the device consumes 28.2 mA with established connection with the dedicated USB key. This translates in an autonomy of about 10 h using the installed $C_{\text{batt}} = 300$ mAh Li-Po battery. The power consumption increases while running measurements or calibration, still by maintaining the connection using the Bluetooth™ transceiver. By assuming a continuous use on the field, for instance by checking multiple trees, and assuming for instance 50 % of the time running measurements and 50 % waiting for commands, this translates in an autonomy in excess of 8 h, therefore allowing a complete working day of measurements on the field. The USB version provides similar figures, for a slightly increased maximum measurement in the field of 10 h. Observe that the Bluetooth transceiver impact is minimal because used only for sending commands, therefore its radio sub-system needs to maintain connection using keep-alive packets only. Such energy consumption and battery autonomy can be considered more than satisfactory for the on-field fruit quality monitoring requirements listed in Sec. II-A and Fig. 1. Furthermore, when a `sleepschedule` command is issued, the device runs scheduled measurements, by enforcing the CPU to a deep sleep mode and subsequent wake ups by using the internal RTC. When the device runs only using the RTC, therefore in ultra-low power mode, it consumes an I_{off} of 2.5 mA. When the device wakes up for measurements, it consumes a nominal measurement current I_{on} of 35.3 mA. The active measurement time under default conditions (*i.e.* $V_{\text{EX}} = 1$ V p-p and 4 cycles of S_T) is 14.4 s, therefore with standard settings of settling time and averaging. Battery autonomy can be calculated as $t_{\text{RUN}} = C_{\text{batt}} / \left[\frac{I_{\text{on}} N t_{\text{m}}}{T} + \frac{I_{\text{off}}(T - N t_{\text{m}})}{T} \right]$, with $T = 1$ h. This equation can be used to define the battery runtime t_{RUN}, *i.e.* the system autonomy, given a desired number of measurements N and the battery capacity C_{batt}. Herein, the 300 mAh battery (C_{batt}) of *FruitMeter* would easily allow to cover the maximum 8 h considered in Sec. II-A for on field fruit bioimpedance measurement. On the other hand, considering the constraints of fruit shipment and storage monitoring (*i.e.* 1 measurement per day for a minimum of 14 days), the FM equipped with such battery is able to last for a maximum of ≈ 120 h (≈ 5 days). This limitation can be simply solved by replacing the battery with another one having larger capacity, such as a common 2000 mAh smartphone battery, which would allow the monitoring of fruit condition for ≈ 800 h (≈ 33 days). Observe that the current consumption provided here refers to the complete PCB, therefore including quiescent power of voltage regulators and biasing circuits. Furthermore, the 2.5 mA consumed by *FruitMeter* in ultra-low power mode (I_{off}), which has the greater contribution on the system power consumption, can be further reduced to μA by reducing the leakage power of the system components.

IV. MEASUREMENT AND VALIDATION

The system prototype (see Fig. 5) was thoroughly tested during multiple experiments focusing on its performance characterization in different contexts, extensively discussed in the following section. First, in Sec. IV-A it was tested using passive components, to verify the correctness of the acquired data against the circuits theoretical values and to test the calibration performances. The system capabilities for the evaluation of fruit quality were tested comparing the result with a benchtop impedance analyzer in Sec. IV-B. This section, other than resulting useful for system validation purposes, posed the basis for the use of FM in a real use-case

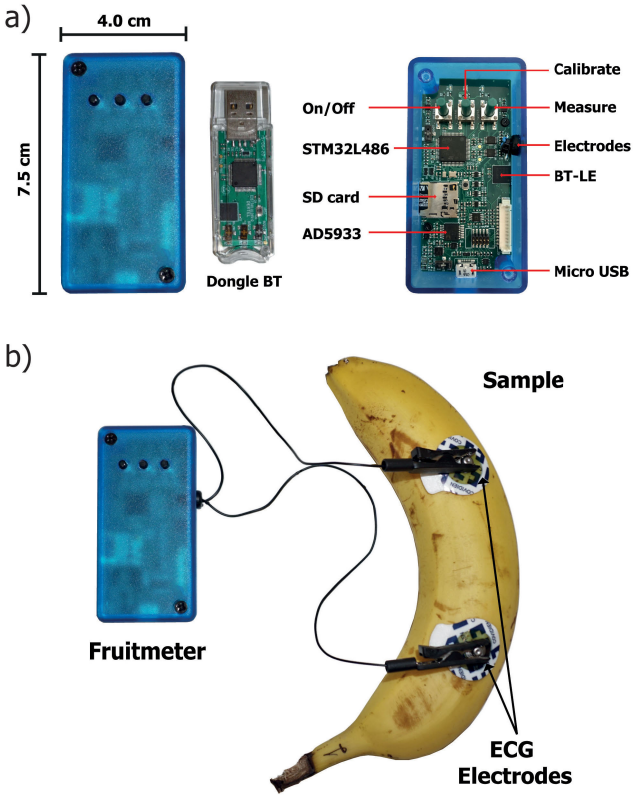


FIGURE 5. a) *FruitMeter* prototype (with size) and Bluetooth dongle (left) and the main PCB components. b) Example connection with fruit sample using commercially available ECG electrodes.

TABLE 3. *FruitMeter* RMSE and NRMSE on circuit measurements.

	Magnitude		Phase	
Circuit	RMSE (Ω)	NRMSE (%)	RMSE (°)	NRMSE (%)
A	314.9	0.91	3.80	4.72
B	241.4	1.20	0.97	2.46
C	433.5	0.78	2.82	5.75
Average	329.9 ± 79.2	0.96 ± 0.18	2.53 ± 1.17	4.31 ± 1.37

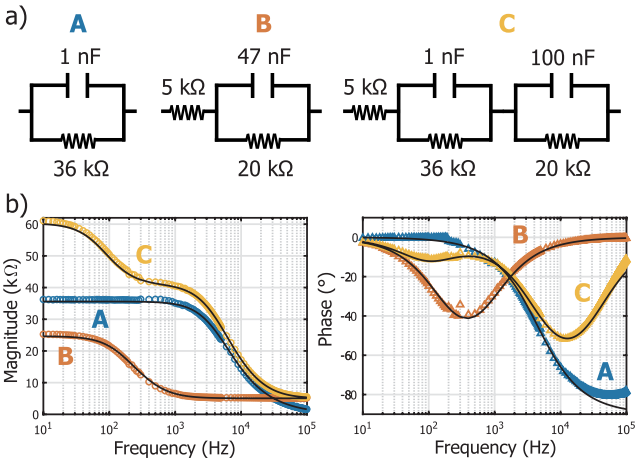


FIGURE 6. a) Test circuits used for *FruitMeter* validation. b) Measured impedance magnitude (left, in dotted points – ∘) and phase (right, in triangular points – △) points of the 3 considered circuits. The FM measurement [Circuit A (∘,△), Circuit B (∘,△) and Circuit C (∘,△)] is compared with the calculated theoretical value (–).

application, presented in Sec. V. Then, the battery life, crucial for its on-field use, was tested using the *FruitMeter* continuous measurement mode, acquiring the impedance data of a banana for several days (see Sec. IV-C). Finally, in Sec. IV-D the system two electrode measurement was compared to a benchtop impedance analyzer in a 2 and 4 electrode configuration measuring the same fruit, to further validate its performance and to test possible new feature developments in future releases. The above-mentioned tests were run interfacing the instrument with a Microsoft Windows™ laptop using USB communication. The USB firmware was chosen over the Bluetooth one due to the higher number of frequency points and the possibility to calibrate each of the 8 frequency ranges with different V_{EX} and R_{FB} parameters.

A. PASSIVE COMPONENTS ANALYSIS

System performance is first tested with a two-terminal measurement on three series-parallel circuits, assembled with different configurations using off-shelf components, to evaluate the capability of the system to precisely follow different impedance curve trends. Fig. 6a shows the test circuits used for the system validation and the value of each circuit component. Circuit A is constituted by a parallel of a 36 kΩ resistor and a 1 nF capacitor. Circuit B is composed by a 5 kΩ resistor in series with a parallel of a 20 kΩ resistor and a 47 nF capacitor. Finally, circuit C includes a 5 kΩ resistor in series with a parallel of a 36 kΩ resistor and a 1 nF capacitor and with another parallel of a 20 kΩ resistor and a 100 nF capacitor. Accuracy is evaluated in terms of Root Mean Square Error (RMSE) and Normalized Root Mean Squared Error (NMRSE) of the measured data compared to theoretical calculations, calculated along the entire frequency spectrum. NRMSE is computed normalizing the RMSE value to the maximum and minimum value of each magnitude and phase spectra, to obtain a comparable index to evaluate the system performance. The circuit magnitude and phase points, measured using the *FruitMeter*, are shown together with the relative calculated theoretical curve in Fig. 6b, left and right side, respectively. The circuit measurements resulted in an average RMSE of around 330 Ω (NMRSE – 0.96 %) and 2.53 ° (NMRSE – 4.31 %) for the impedance magnitude and phase, respectively. Tab. 3 shows the RMSE and NMRSE of the measured impedance magnitude and phase for the considered circuits. From both Fig. 6b and Tab. 3, we observe a difference in the accuracy of the measurement of the impedance magnitude compared to the phase. In fact, while the former has a low, almost negligible error, the latter presents a larger inaccuracy.

Fig. 7 shows the absolute phase difference of the measurement of the considered circuits compared to the corresponding theoretical values. The small phase peaks at 200 Hz and 1 kHz, correlated to the upper limits of two of the eight frequency ranges, are most probably due to

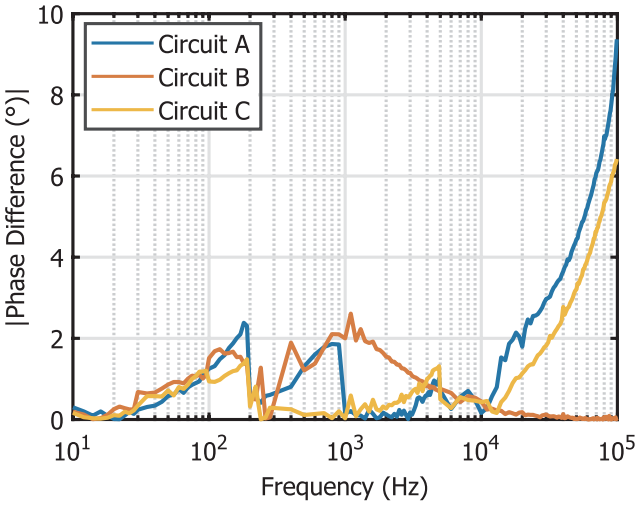


FIGURE 7. Absolute phase difference of the considered circuits with their corresponding theoretical value.

the MCLK frequency value. In fact, at these frequencies, the respective 1 MHz and 20 MHz clock DFT start degrading, leading to a deviation from the expected phase value. Furthermore, despite the good average RMSE of the phase points, it is important to highlight that the phase difference resulted to be larger when the reactive component of the tested impedance was prevailing, around 1 kHz for circuit 2 and at high frequency ($f \geq 10$ kHz) for circuit 1 and 3. This is most probably due to the presence of a parasitic capacitor (not considered in the calibration resistor model), or to a limited MCLK lower frequency. While the latter implies a straightforward firmware correction, the former, which has a stronger influence on the phase calculation (especially with a mostly reactive sample behavior), can be also corrected during the calibration process in further releases of the device. Once such problem is solved, the system precision will be on a comparable level with state-of-the-art impedance analyzers.

B. FRUIT ANALYSIS

The instrument was then tested by measuring the impedance of five different fruit (apple, avocado, banana, mango and tomato - 3 samples each) and subsequently comparing the results with a Zurich Instruments MFIA benchtop impedance analyzer (Zurich Instruments, Zurich, Switzerland), able to operate in the 1 mHz–5 MHz range with a 0.05 % basic accuracy, in the two electrodes configuration. The obtained impedance magnitude and phase points were compared in terms of RMSE and NRMSE (Tab. 4), over 20 coincident frequency points of the *FruitMeter* and MFIA spectra. The RMSE calculation over this limited number of frequency points is due to the limitation of the benchtop impedance analyzer, which allows the choice of a precise frequency range but not of the exact frequency points of the *FruitMeter*. The electrical contact with the fruit samples is established by means of two pre-gelled ECG (Ag/AgCl) electrodes

TABLE 4. Average RMSE and NRMSE on fruit measurements.

	Magnitude		Phase	
Fruit	RMSE (Ω)	NRMSE (%)	RMSE (°)	NRMSE (%)
Apple	834.66 ± 389.5	0.7 ± 0.37	5.55 ± 0.15	9.90 ± 0.3
Avocado	71.80 ± 23.1	2.1 ± 0.90	1.11 ± 0.12	3.32 ± 0.3
Banana	103.29 ± 13.90	0.6 ± 0.10	2.52 ± 0.30	6.49 ± 0.9
Mango	319.37 ± 151.7	2.7 ± 1.30	3.03 ± 0.40	6.47 ± 0.9
Tomato	221.90 ± 105.6	0.7 ± 0.21	1.75 ± 0.30	3.57 ± 0.5

(Fiab F3001ECG), placed with same spacing on all fruit triplicates, to reduce and standardize the measurement error. This type of electrodes is commonly considered to be the best starting choice for bioimpedance studies, including the present work, as they have been demonstrated to provide reliable and stable measurements up until 10 MHz [1], [34]. The spacing between electrodes, showed in Fig. 5b, was chosen to be representative not only of the fruit skin contribution on the measured bioimpedance, but also of the fruit pulp, as at longer distances correspond a deeper and more diverse penetration of the current flow pathways in the fruit [1]. Fig. 8A and Fig. 8B show the impedance magnitude and phase curves, respectively, for the above-mentioned fruit. The FM measurement data are acquired using a calibration dedicated to each fruit type, obtained as described in Sec. II-D.

The measured fruit impedance magnitude, showed in Fig. 8A, ranges from a maximum of around 280 kΩ for the apple to a minimum of around 4 kΩ for the avocado, while the phase, displayed in Fig. 8B, ranges from a maximum of around $-64\,°$ to a minimum of $-0.7\,°$. The average NRMSE, shown in Tab. 4, for the impedance magnitude ranged from a minimum of 0.6 % for the banana to a maximum of 2.7 % for the mango, while for phase from a minimum of 3.3 % to a maximum of 9.90 %, for apple and avocado, respectively. The above-mentioned results are coherent with the RMSE trends observed for the circuit impedance in Sec. IV-A, considering the non-ideality of the fruit samples compared to passive components and the comparison of the measurement to an instrument rather than a theoretical curve. Furthermore, the different number of points for the calculation of the RMSE might have further enhanced the relative differences in this metric observed in the circuit experiment compared to the fruit one.

From the presented results it also possible to appreciate that the 3 samples of mango and banana fruit (Fig. 8, magnitude A and phase B, diagrams c and d, respectively) were measured at a similar stage (having comparable magnitude values), while the other considered fruit, namely apple, avocado and tomato, significantly differed in their measured status (Fig. 8, magnitude A and phase B, diagrams a, b and e, respectively). Since the focus of this section was to use different fruit as a validation sample, investigating the ability of the system to measure widely different fruit types, the differences among fruit are not discussed. Nevertheless, the results presented in this section allowed to further validate the quality of the FM bioimpedance data acquisition, comparable in precision

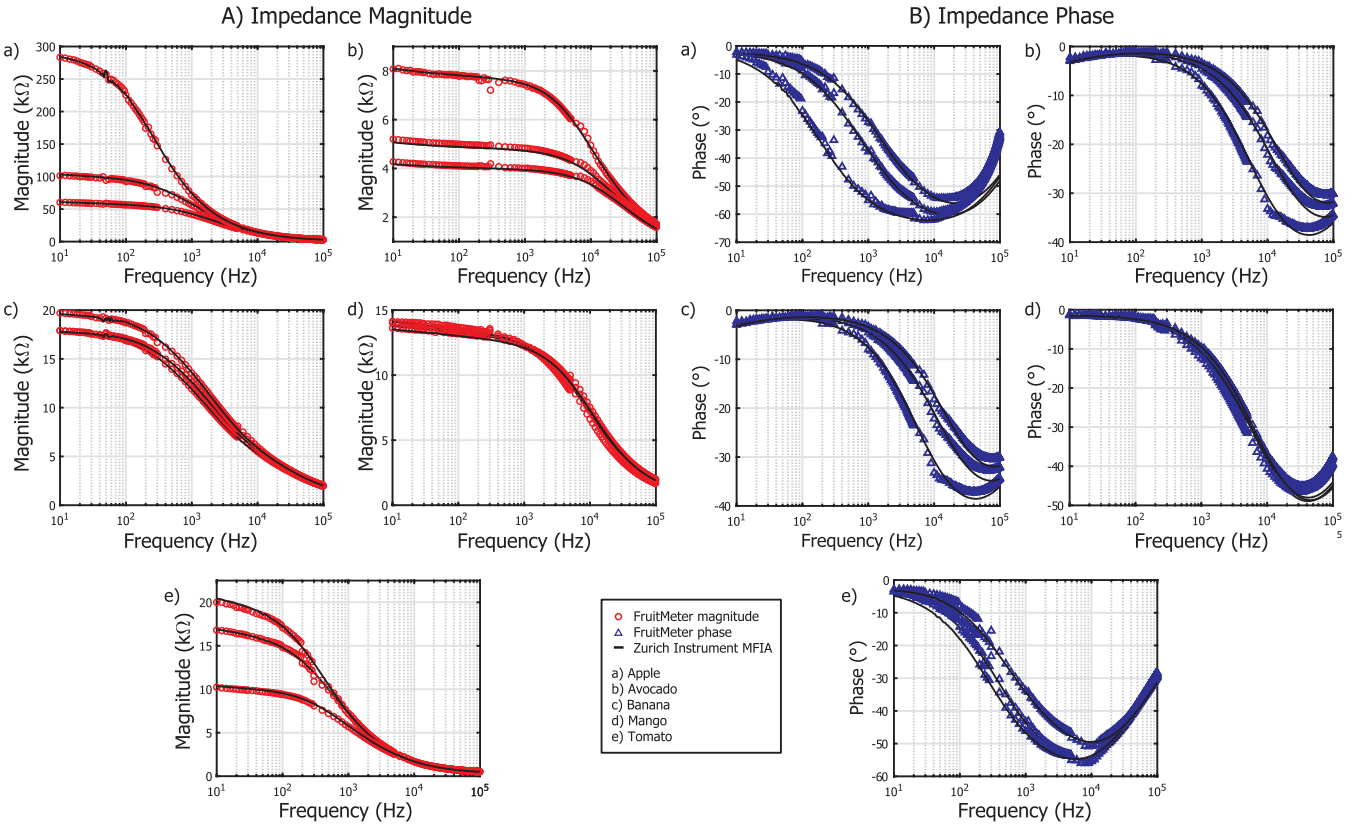


FIGURE 8. Measured impedance magnitude (A) and phase (B) spectra of 3 samples of apple (a), avocado (b), banana (c), mango (d), and tomato (e). The FM magnitude (○) and phase (△) measurements are compared with the Zurich Instrument MFIA impedance analyzer (–).

to the benchtop system, to be employed in the fruit quality investigation scenario, presented in Sec. V of this work.

C. CONTINUOUS MEASUREMENT

The system is then validated for its continuous use, to evaluate the battery life and validity of the calibration for a changing impedance, for a potential use in the unsupervised post-harvest monitoring of fruit quality. The two-electrode measurement, scheduled every 60 min, was carried out on a banana, up until the complete depletion of the 300 mAh battery. The electrical contact with the fruit samples is established as described previously in Sec. IV-B, using ECG electrodes. Such electrodes are proven to be stable during time, with a possible slight increase of their own impedance, caused by the drying of the contact gel at the sample-electrode interface [35]. To minimize this aspect and reduce the measurement uncertainty, the ECG electrodes are maintained on the fruit surface throughout the entirety of the experiment. Such method is used within studies showing both increasing [36] and decreasing [37] impedance magnitude trends during the days, proving negligible the effect of the electrode degradation on the continuous impedance measurement.

As highlighted in Fig. 9, showing the result of the above-mentioned measurement on the banana, the instrument is capable of easily and independently following the aging trend

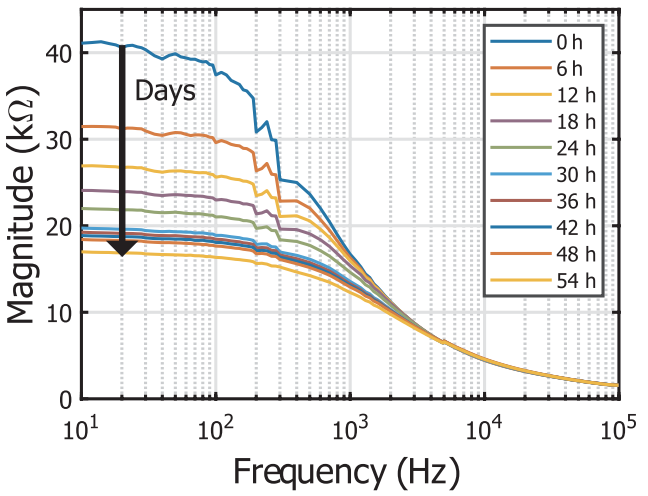


FIGURE 9. Continuous impedance magnitude measurement of a banana for 54 h every 60 minutes. The lines represent a measurement every 6 h.

of a fruit for several days or, as in this case, until the battery discharges. This limitation can be easily overcome by using a larger battery. Furthermore, the possibility of powering the system via USB allows such system's mode to be used for the collection of a large and automated quantity of bioimpedance data without the need of an operator. In fact, once the system

is calibrated and the continuous mode is activated, it results to be completely autonomous in the data collection process. The different powering possibilities, coupled with the continuous mode, allow the system to be used and integrated at full potential also in the post-harvest fruit quality monitoring, during its transport, storage and processing.

D. TWO ELECTRODES VS. FOUR ELECTRODES

To further verify the correctness of the system, additional tests concerning the comparison of the two-terminal measurement with a four-terminal one were done using a benchtop impedance analyzer. The contact with the fruit is established as described in Sec. IV-B, adding two additional electrodes in between the ones used for the two-electrodes configuration. The impedance spectra of the samples were first measured using the *FruitMeter*, then the fruit impedance was collected using the MFIA impedance analyzer in the two and four-electrode configurations, respectively.

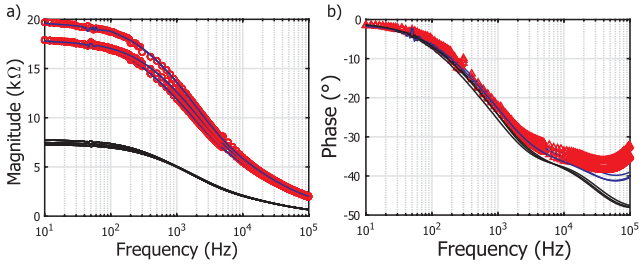


FIGURE 10. Measured impedance magnitude (a) and phase (b) points of the 3 considered banana. The *FruitMeter* measurements (in dotted and triangular points for impedance and phase – ∘ and △, respectively) are compared with the Zurich Instrument MFIA impedance analyzer in the 2 (–) and 4 (–) electrodes configuration.

Fig. 10 displays the comparison of the fruit impedance spectra acquired with the different electrode configurations. As expected, both impedance magnitude and phase, while presenting the same overall curve behavior, have a lower magnitude and a smoother curve between the two modes. This is due to the fact that the four-electrode measurement applies the voltage and measures the current between two different pairs of electrodes, thus excluding the electrode polarization at the electrode-sample interface from the final result.

In practice, using a four- rather than a two-electrode measurement yields a more precise result while complicating the system setup. Therefore, the choice of the optimal system architecture is usually made depending on the specific application constraints. In the case of *FruitMeter*, specifically designed to carry out a rapid and on-field impedance measurement, we considered the two-electrode setup to be the ideal one, thus greatly simplifying and accelerating the electrode positioning step before each single measurement.

V. REAL USE-CASE

After its extensive validation on both passive components and fruit samples presented in the above sections, FM was further validated for the evaluation of banana aging evolution, therefore in a real use-case scenario. A total of 15 green bananas were purchased from a local dealer and used as sample under test, keeping the samples at controlled T and RH (T = 18 °C; RH = 70 %) for 10 days and measuring the bioimpedance every 24 h. The electrical contact with the sample was established as depicted in Fig. 5b. Furthermore, image analysis was performed, as described in [37], in order to obtain a browning index of the fruit, which consists in the detection of the percentage of the dark areas over the total surface of the bananas, to correlate the aging stage with the impedance measurements.

The analysis of the resulting impedance from the Nyquist plot in Fig. 11a shows that there is a clear difference between days in terms of impedance and that, in addition, it is possible to identify two opposite trends. In a first phase, corresponding to the first six days of storage, the impedance clearly increases, while during the next days of analysis the impedance tends to constantly decrease, in accordance with similar trends observed in [7], [27], [37].

Afterwards, in order to reduce the dimensionality of the data and to obtain indexes useful to correlate results with the fruit physiological conditions, the banana bioimpedance data were fitted with an equivalent circuit model, able to represent the flow of current in the plant tissues [38]. Among the variety of available equivalent circuits [39], the Cole model [40] was chosen, due to its simplicity and widespread application in this context. The equivalent circuit model fitting resulted in an average % RMSE, obtained by averaging the values of the 15 bananas over the 10 days period, of 3.03 %, corresponding to an average error, in terms of impedance magnitude, of 505 Ω. Given the low error of the fitting results, it is possible to state that the Cole model is a good approximation of the impedance evolution for the banana aging. Hence, the resulting extracted parameter, namely the series resistor (R_s), parallel resistor (R_p), and the constant phase element CPE-T and CPE-P values, can be used to estimate the physiological state of the fruit, both individually and combined.

Subsequently, two-way analysis of variance (ANOVA) was performed on both the banana skin browning index evolution and among each extracted circuit parameter to evaluate statistically significant differences during the fruit aging. Tukey's HSD (honestly significant difference) test [41] was performed, and the differences were calculated for an appropriate level of interaction ($p \leq 0.05$). Results are reported as the mean, standard deviation (SD) of the mean and resulting grouping both in table 5 and graphically in Fig. 11b. Here, the SD are not shown for the sake of figure clarity. Regarding the circuit parameters, no statistically significant differences were found between the days of aging and the series resistance (R_s, $p > 0.05$), which is typically used to describe the flow of current in the intracellular fluids, while the other extracted parameters (R_p, CPE-T and CPE-P), showed statistically significant differences among the considered aging days ($p \ll 0.05$). Such parameters are respectively related to the extracellular resistance (R_p), to the cell membrane

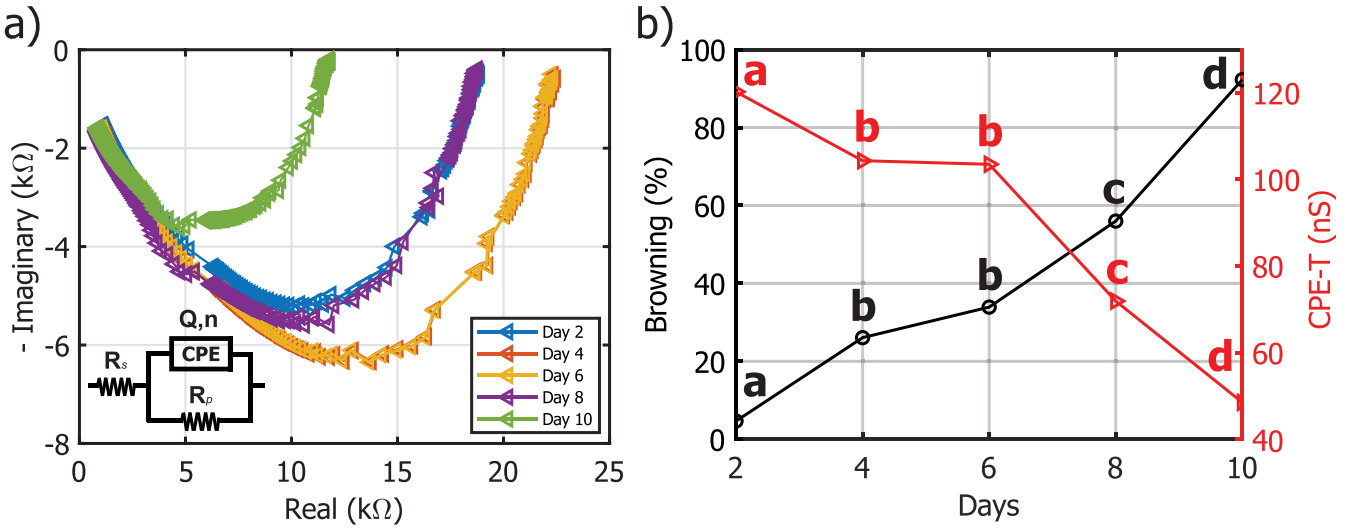


FIGURE 11. a) Nyquist plot of the average real and imaginary components of 15 bananas during a time span of 10 days every 2 days. The inset on the bottom-left side shows the circuit schematic of the Cole model used to fit the experimental data. b) Banana skin browning (left y-axis, –) and CPE-P Cole model parameter (right y-axis, –) evolution during the 10 days of fruit aging. Values represents the average of 15 fruits at each day of aging. Mean values with no common letters are statistically different according to HSD ($p \leq 0.05$).

TABLE 5. Effect of the aging of banana on the fruit skin browning evolution and on the electrical parameters of the fitted equivalent circuit (Cole model). Values represents the average ± SD of 15 fruits at each day of aging. Different letters indicate statistically significant difference among each group, mean values with no common letters are statistically different according to HSD ($p \leq 0.05$).

	Color		Cole circuit parameters							
Day	Browning (%)		Series resistance R_s (Ω)		Parallel resistance R_p (kΩ)		CPE–T Q (nS)		CPE–P n (a.u.)	
2	4.68 ± 1.7	a	24.26 ± 48.7	a	18.5 ± 1.6	a	120 ± 34	a	0.632 ± 0.02	a
4	26.08 ± 9.10	b	12.18 ± 34.3	a	21.7 ± 1.2	b	104 ± 21	b	0.641 ± 0.01	a
6	33.92 ± 12.6	b	7.845 ± 23.6	a	21.5 ± 1.2	b	103 ± 20	b	0.641 ± 0.01	a
8	55.98 ± 18.9	c	$0.405\,\mu \pm 1\,\mu$	a	18.4 ± 1.9	a	71.9 ± 24	c	0.675 ± 0.02	b
10	92.28 ± 4.50	d	$0.529\,\mu \pm 1.4\,\mu$	a	11.9 ± 1.9	c	48.5 ± 17	d	0.704 ± 0.02	c
p value	$\ll 0.05$		0.04		$\ll 0.05$		$\ll 0.05$		$\ll 0.05$	

pseudo-capacitance (CPE-T) and to the heterogeneity of size and shape of the fruit cells (CPE-P). It is particularly interesting to observe how R_p precisely follow the bioimpedance evolution trend depicted in Fig. 11(a), with an initial increase of the resistance value (day 2 to 4), followed by a steady phase, where such parameter magnitude remains relatively constant (day 4 and 6), and by a rapid decrease in day 8 and 10. Such results, also observed in [6] in the case of apple aging, allow to consider such parameter as a good fruit aging indicator.

Furthermore, the results yielding from the statistical analysis of the banana skin browning and of the Cole model CPE-T (Fig. 11b, black and red line, respectively) highlighted an equivalent but opposite trend in time of such parameters. In fact, both the color and the CPE-T present the same grouping pattern, with a steady phase at day 4 and 6, where the fruit skin remains at around 30 % browning and the cell membrane pseudo-capacitance value does not significantly change. The latter is followed by a rapid decrease in time, *i.e.* a symptom of a faster cell-wall degradation and consequent cell-death, also confirmed by the color change trend. Such result can be considered as a starting point to develop models able to discriminate a fruit acceptability threshold after which bananas are typically no longer accepted by consumer, typically associated to a high browning degree of banana peel with its spoilage.

VI. DISCUSSION

Tab. 6 compares the *FruitMeter* performance with similar state-of-the-art portable impedance analyzer. Comparing the available information on analogous instruments in literature, our system present comparable frequency range, number of channels and excitation voltage (V_{EX}) values. These features are mostly uniform, due to the use of the AD5933 chip-set,

TABLE 6. List of portable microcontroller-based impedance analyzers.

Reference	Application	Frequency range	Channels	Excitation voltage	Internal calibration resistors	Internal feedback resistors
[23]	Embedded applications	1 – 100 kHz	4	20 – 200 mV	No	No
[24]	Technical objects diagnostic	0.01 Hz – 100 kHz	2	1 mV – 1 V	No	10 Ω – 100 MΩ (in decades, 8 total)
[16]	In-vivo biomedical applications	10 Hz – 500 kHz	3	10 Vrms	No	No
[17]	Biological applications	1 – 100 kHz	2	200 mV, 400 mV, 1 V, 2 V	No	3 (not specified)
[25]	Fruit quality control	5 – 100 kHz	2	200 mV, 400 mV, 1 V, 2 V	No	33 Ω, 330 Ω, 3.3 kΩ, 33 kΩ – Manual switch
[20]	Saline solution concentration	10 Hz – 100 kHz	2	100 mV, 500 mV, 1 V	No	5 kΩ, 10 kΩ, 15 kΩ – Digital potentiometers
[21]	Metals corrosion monitoring	0.01 Hz – 100 kHz	2	10 mV – 2 V	No	No
[22]	Human skin bioimpedance	5 – 100 kHz	2	25 mV, 50 mV, 200 mV, 400 mV, 1 V, 2 V, 3 V	20 kΩ	15 kΩ
This work	Fruit quality control	10 Hz – 100 kHz	2	200 mV, 400 mV, 1 V, 2 V	200 Ω, 15 kΩ, 470 kΩ	200 Ω, 2 kΩ, 20 kΩ, 200 kΩ – Programmable multiplexer

Reference	Maximum error	SD Card	Data communication	Battery operated	Battery Size - Life	Size
[23]	2 % – Magnitude 1.5 ° – Phase	No	N/A	No	N/a	N/A
[24]	1.6 % – Magnitude 0.6 % – Phase	No	ZigBee, USB	Yes	N/A	65 × 120 mm
[16]	12.3 % – Magnitude 12.1 ° – Phase	No	SPI	No	N/a	N/A
[17]	3.5 % – Magnitude 2.8 ° – Phase	No	USB	No	N/a	N/A
[25]	N/A	Yes	BT	Yes	4400 mAh – 24 h	N/A
[20]	3.5 % – Not specified	No	USB	No	N/a	N/A
[21]	5 % – Magnitude 3 ° – Phase	No	USB	No	N/a	N/A
[22]	6 % – Magnitude 3 ° – Phase	No	BT, USB	Yes	N/A	N/A
This work	1.2 % – Magnitude 3.8 ° – Phase	Yes	BT, USB	Yes	300 mAh – 54 h*	75 × 40 mm

N/A = Not Available, N/a = Not applicable, * = Calculated in continuous mode with a 60 minutes measurement span.

which limits the upper frequency point to 100 kHz, while better performing at the pre-programmed excitation voltages (0.2, 0.4, 1 and 2 V). Futhrermore, the AD5933 is configured for a two channel use. The maximum error of the system, related to the passive component measurement to exclude other influences (e.g. electrode polarization and positioning, comparison with a benchtop impedance analyzer and biological samples), resulted to be in line with, or even improved as compared to the best results in literature. In fact, the maximum impedance magnitude error resulted to be 1.20 %, against 1.60 % in [24], while the maximum phase error was 3.8 °, comparable to most of the considered instruments. While the former appears to be already optimized to ensure good performances, the latter can be dramatically and easily improved, as also discussed in Sec. IV-A, by correcting the parasitic capacitance contribution appearing with the reactive behavior.

Given the above-mentioned characteristics to be in line with other similar systems, the strong points of our system are represented by the implementation of internal calibration and feedback resistors and by the extreme portability (7.5 × 2 × 4 cm in volume). While no other systems, except for [22] implementing a single 20 kΩ resistor, present this feature, our system implements 3 internal calibration resistors (200 Ω, 15 kΩ and 470 kΩ) allowing an easy and fast on–field calibration. Furthermore, our system includes 4 feedback resistors (R_{FB} – 200 Ω, 2 kΩ, 20 kΩ and 200 kΩ), covering a wide range of possible impedance magnitude and selectable by means of a programmable multiplexer. This, coupled with the division of the frequency range into 8 independent ranges, allows the choice of the optimal combination of R_{FB} and V_{EX} for each frequency range, to minimize the output signal chance of saturation and noise. This is a great advantage compared to other systems implementing internal R_{FB}, such as [15] and [20], which require the users to manually select the single resistor to calibrate the entire frequency range, greatly limiting the flexibility and range of application of the systems.

FruitMeter also includes the possibility to save the bioimpedance data on an internal SD card and both a Bluetooth and USB data communication, features implemented only by [25] and [22], respectively. Furthermore our system results to have a smaller size than [24] and a longer battery life compared to [25], with systems such as [24] and [22] only citing the possibility to be battery operated. The above-mentioned characteristics, coupled with the MicroPython-based firmware, allowed the development of a flexible, low cost and highly portable system, which can be used throughout the fruit supply chain, from the field to its transport and storage using both on-demand and scheduled measurements. Furthermore the possibility to be interfaced with the custom-made application for smartphone and laptop, via both Bluetooth and USB, for a real-time data visualization, paves the way for an easy integration of a user friendly interface

and a real time data analysis. Such wide features range, other than improving the state-of-the-art on similar AD5933-based portable system, allows FM to be more suitable for our desired application (see Fig. 1) compared to commercially available portable impedance analyzers. In fact, commercial systems such as the Metrohm DropSens^{TM} and the SinePhase^{TM}, while allowing a better measurement accuracy and having the possibility of being battery operated, have the major drawback of requiring a PC connected via USB to acquire and store bioimpedance data. On the contrary, FM allows to store an extremely high number of measurement in the internal memory storage, thus making the system completely stand-alone. Furthermore, additionally to the above-mentioned parasitic capacitance correction and to the development of a reliable and user-friendly app interface, future releases of the instrument will consider the use of different values of R_{CAL} and R_{FB}, to cover a more precise impedance range and the integration of a larger battery for the on-field data collection. Moreover, the results from the real use-case contributed to give both a better insight of the processes occurring during the fruit aging and confirmed the validity of using the bioimpedance data acquired with the FM to achieve such task, successfully discriminating different fruit physiological conditions.

VII. CONCLUSION

In this paper, *FruitMeter*, a portable impedance analyzer thought specifically for fruit quality characterization, is designed, tested and validated. The AD5933-based system operates in the 10 Hz−100 kHz frequency range, features internal R_{CAL} and R_{FB} selectable at firmware level, both BLE and USB communication, microSD data storage and a 300 mAh rechargeable battery, all comprised in a compact size. The system shows a good accuracy of the measurement compared to a benchtop impedance analyzer, for both passive electrical components and fruit samples. Its accuracy, portability, low cost and easiness of use, coupled with the high firmware customizability, allow the system to be utilized for fruit quality monitoring in different contexts of fruit supply chains, from the field to the transport, storage and processing, paving the way for smart agriculture applications.

ACKNOWLEDGMENT

The authors would like to thank Prof. Ørjan G. Martinsen, head of the bioimpedance and Medical Technology group at the University of Oslo, for the access to its laboratories and instruments for the impedance measurements. The authors also thank Davide Dellepiane, Electronic Design Laboratory (IIT), for support in the device assembly. This paper is an extension of the confernece paper FruitMeter: An AD5933-Based Portable Impedance Analyzer for Fruit Quality Characterization presented at IEEE ISCAS 2020. https://ieeexplore.ieee.org/document/9181287

REFERENCES

[1] S. Grimnes and O. G. Martinsen, *Bioimpedance and Bioelectricity Basics* (Biomedical Engineering), 3rd ed. London, U.K.: Academic, 2015.

[2] M. Grossi and B. Riccò, "Electrical impedance spectroscopy (EIS) for biological analysis and food characterization: A review," *J. Sensors Sensor Syst.*, vol. 6, no. 2, pp. 303–325, Aug. 2017.

[3] U. Pliquett, "Bioimpedance: A review for food processing," *Food Eng. Rev.*, vol. 2, no. 2, pp. 74–94, Jun. 2010.

[4] F. Clemente, M. Romano, P. Bifulco, and M. Cesarelli, "EIS measurements for characterization of muscular tissue by means of equivalent electrical parameters," *Measurement*, vol. 58, pp. 476–482, Dec. 2014.

[5] D. El Khaled, N. N. Castellano, J. A. Gazquez, R. M. G. Salvador, and F. Manzano-Agugliaro, "Cleaner quality control system using bioimpedance methods: A review for fruits and vegetables," *J. Cleaner Prod.*, vol. 140, pp. 1749–1762, Jan. 2017.

[6] P. Ibba, G. Cantarella, B. D. Abera, L. Petti, A. Falco, and P. Lugli, "Selection of cole model bio-impedance parameters for the estimation of the ageing evolution of apples," in *Proc. 17th Int. Conf. Electr. Bioimpedance*. Singapore: Springer, 2019, p. 191.

[7] P. Ibba, A. Falco, B. D. Abera, G. Cantarella, L. Petti, and P. Lugli, "Bio-impedance and circuit parameters: An analysis for tracking fruit ripening," *Postharvest Biol. Technol.*, vol. 159, Jan. 2020, Art. no. 110978.

[8] C. Andrade, V. M. Blanco, A. Collazo, M. Keddam, X. R. Nóvoa, and H. Takenouti, "Cement paste hardening process studied by impedance spectroscopy," *Electrochimica Acta*, vol. 44, no. 24, pp. 4313–4318, Jul. 1999.

[9] H. Zhu, H. Luo, D. Ai, and C. Wang, "Mechanical impedance-based technique for steel structural corrosion damage detection," *Measurement*, vol. 88, pp. 353–359, Jun. 2016.

[10] M. Z. Iqbal, "Preparation, characterization, electrical conductivity and dielectric studies of Na_2SO_4 and V_2O_5 composite solid electrolytes," *Measurement*, vol. 81, pp. 102–112, Mar. 2016.

[11] M. Carminati, A. Rottigni, D. Alagna, G. Ferrari, and M. Sampietro, "Compact FPGA-based elaboration platform for wide-bandwidth electrochemical measurements," in *Proc. IEEE Int. Instrum. Meas. Technol. Conf.*, May 2012, pp. 264–267.

[12] Z. Xu, J. Yao, Z. Wang, Y. Liu, H. Wang, B. Chen, and H. Wu, "Development of a portable electrical impedance tomography system for biomedical applications," *IEEE Sensors J.*, vol. 18, no. 19, pp. 8117–8124, Oct. 2018.

[13] I.-H. Park, Y. Hong, H.-S. Jun, E.-S. Cho, and S. Cho, "DAQ based impedance measurement system for low cost and portable electrical cell-substrate impedance sensing," *BioChip J.*, vol. 12, no. 1, pp. 18–24, Mar. 2018.

[14] A. Devices, "1 MSPS, 12 bit impedance converter network analyzer," Analog Devices, Norwood, MA, USA, Datasheet AD5933, 2010, p. 40.

[15] A. AboBakr, M. Mohsen, L. A. Said, A. H. Madian, A. S. Elwakil, and A. G. Radwan, "Toward portable bio-impedance devices," in *Proc. 4th Int. Conf. Adv. Comput. Tools Eng. Appl. (ACTEA)*, Jul. 2019, pp. 1–4.

[16] J. Punter-Villagrasa, B. Del Moral-Zamora, J. Colomer-Farrarons, P. Miribel-Catala, I. Rodriguez-Villarreal, J. Cid, and B. Prieto-Simon, "A portable point-of-use EIS device for *in-vivo* biomédical applications," in *Proc. 29th Conf. Design Circuits Integr. Syst. (DCIS)*, 2014, pp. 1–5.

[17] K. Chabowski, T. Piasecki, A. Dzierka, and K. Nitsch, "Simple wide frequency range impedance meter based on AD5933 integrated circuit," *Metrol. Meas. Syst.*, vol. 22, no. 1, pp. 13–24, Mar. 2015.

[18] J. Ferreira, F. Seoane, and K. Lindecrantz, "AD5933-based electrical bioimpedance spectrometer. Towards textile-enabled applications," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Aug. 2011, pp. 3282–3285.

[19] J. Ferreira, F. Seoane, A. Ansede, and R. Bragos, "AD5933-based spectrometer for electrical bioimpedance applications," *J. Phys., Conf. Ser.*, vol. 224, Apr. 2010, Art. no. 012011.

[20] M. Grossi, C. Parolin, B. Vitali, and B. Riccò, "Electrical impedance spectroscopy (EIS) characterization of saline solutions with a low-cost portable measurement system," *Eng. Sci. Technol., Int. J.*, vol. 22, no. 1, pp. 102–108, Feb. 2019.

[21] S. Grassini, S. Corbellini, M. Parvis, E. Angelini, and F. Zucchi, "A simple arduino-based EIS system for *in situ* corrosion monitoring of metallic works of art," *Measurement*, vol. 114, pp. 508–514, Jan. 2018.

[22] J. R. Harvey and Y. Mendelson, "A portable sensor for skin bioimpedance measurements," *Int. J. Sensors Sensor Netw.*, vol. 7, no. 1, p. 1, 2019.

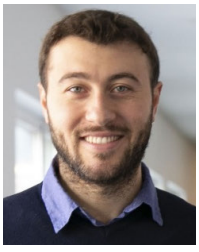
[23] C. Margo, J. Katrib, M. Nadi, and A. Rouane, "A four-electrode low frequency impedance spectroscopy measurement system using the AD5933 measurement chip," *Physiol. Meas.*, vol. 34, no. 4, pp. 391–405, Apr. 2013.

[24] J. Hoja and G. Lentka, "A family of new generation miniaturized impedance analyzers for technical object diagnostics," *Metrol. Meas. Syst.*, vol. 20, no. 1, pp. 43–52, Mar. 2013.

[25] A. Al-Ali, A. Elwakil, A. Ahmad, and B. Maundy, "Design of a portable low-cost impedance analyzer," in *Proc. 10th Int. Joint Conf. Biomed. Eng. Syst. Technol.* Porto, Portugal: Scitepress, Mar. 2017, pp. 104–109.
[26] A. A. Bakr, A. G. Radwan, A. H. Madian, and A. S. Elwakil, "Aging effect on apples bio-impedance using AD5933," in *Proc. 3rd Int. Conf. Adv. Comput. Tools Eng. Appl. (ACTEA)*. Beirut, Lebanon: IEEE, Jul. 2016, pp. 158–161.
[27] A. Chowdhury, T. K. Bera, D. Ghoshal, and B. Chakraborty, "Studying the electrical impedance variations in banana ripening using electrical impedance spectroscopy (EIS)," in *Proc. 3rd Int. Conf. Comput., Commun., Control Inf. Technol. (C IT)*, Hoogly, West Bengal, Feb. 2015, pp. 1–4.
[28] B. Nicolaï, T. Defraeye, B. De Ketelaere, E. Herremans, M. Hertog, W. Saeys, A. Torricelli, T. Vandendriessche, and P. Verboven, "Nondestructive measurement of fruit and vegetable quality," *Annu. Rev. food Sci. Technol.*, vol. 5, pp. 285–312, Jan. 2014.
[29] G. Paltrinieri, "Handling of fresh fruits, vegetables and root crops," FAO, Rome, Italy, Tech. Rep., 2014. [Online]. Available: http://www.fao.org/3/a-au186e.pdf
[30] R. Jedermann, U. Praeger, M. Geyer, and W. Lang, "Remote quality monitoring in the banana chain," *Phil. Trans. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 372, no. 2017, Jun. 2014, Art. no. 20130303.
[31] A. F. Neto, N. C. Olivier, E. R. Cordeiro, and H. P. de Oliveira, "Determination of mango ripening degree by electrical impedance spectroscopy," *Comput. Electron. Agricult.*, vol. 143, pp. 222–226, Dec. 2017.
[32] A. Chowdhury, S. Datta, T. K. Bera, D. Ghoshal, and B. Chakraborty, "Design and development of microcontroller based instrumentation for studying complex bioelectrical impedance of fruits using electrical impedance spectroscopy," *J. Food Process Eng.*, vol. 41, Jun. 2018, Art. no. e12640.
[33] A. R. Varlan and W. Sansen, "Nondestructive electrical impedance analysis in fruit: Normal ripening and injuries characterization," *Electro-Magnetobiol.*, vol. 15, no. 3, pp. 213–227, Jan. 1996.
[34] A. Bosnjak, A. Kennedy, P. L. Herrera, M. Borges, J. McLaughlin, and O. Escalona, "Performance assessment of dry electrodes for wearable long term cardiac rhythm monitoring: Skin-electrode impedance spectroscopy," in *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, Jul. 2017, pp. 1861–1864.
[35] P. Ask, P. Å. Öderg, S. Ödman, T. Tenland, and M. Skogh, "ECG electrodes: A study of electrical and mechanical long-term properties," *Acta Anaesthesiologica Scandinavica*, vol. 23, no. 2, pp. 189–206, Apr. 1979.
[36] A. Chowdhury, P. Singh, T. K. Bera, D. Ghoshal, and B. Chakraborty, "Electrical impedance spectroscopic study of mandarin orange during ripening," *J. Food Meas. Characterization*, vol. 11, no. 4, pp. 1654–1664, Dec. 2017.
[37] P. Ibba, A. Falco, A. Rivadeneyra, and P. Lugli, "Low-cost bio-impedance analysis system for the evaluation of fruit ripeness," in *Proc. IEEE SENSORS*, Oct. 2018, pp. 1–4.
[38] C. Tronstad and A. H. Pripp, "Statistical methods for bioimpedance analysis," *J. Electr. Bioimpedance*, vol. 5, no. 1, pp. 14–27, Apr. 2014.
[39] A. Prasad and M. Roy, "Bioimpedance analysis of vascular tissue and fluid flow in human and plant body: A review," *Biosyst. Eng.*, vol. 197, pp. 170–187, Sep. 2020.
[40] K. S. Cole, "Permeability and impermeability of cell membranes for ions," in *Proc. Cold Spring Harbor Symp. Quant. Biol.*, vol. 8, 1940, pp. 110–122.
[41] H. Abdi and L. J. Williams, "Tukey's honestly significant difference (HSD) test," *Encyclopedia Res. design*, vol. 3, no. 1, pp. 1–5, 2010.

PIETRO IBBA (Student Member, IEEE) received the bachelor's degree in industrial biotechnologies and the master's degree in food quality and safety from Tuscia University, Viterbo, Italy, in 2015 and 2017, respectively. He is currently pursuing the Ph.D. degree in food engineering and biotechnology with the Sensing Technologies Laboratory, Free University of Bozen-Bolzano. His current research interests include development of electrical impedance spectroscopy methods for the characterization of fruit quality.

MARCO CREPALDI (Member, IEEE) received the degree in engineering *(summa cum laude)* and the Ph.D. degree in electronic engineering from the Politecnico di Torino (Polito), Turin, Italy, in 2005 and 2009, respectively. He was a Visiting Scholar with the Electrical Engineering Department, Columbia University, New York City, in 2008. After the Ph.D., he held a postdoctoral position at the VLSI-Laboratory, Electrical Engineering Department, PoliTo. He held a postdoctoral position at the former Istituto Italiano di Tecnologia@PoliTo Center for Space Human Robotics (IIT-CSHR). He is currently the Coordinator of the Electronics Design Laboratory, IIT Center for Human Technology in Genova. He is the author or coauthor of more than 60 publications and two international patents. His research interests include analysis, simulation and development of integrated event-driven and all-digital impulse-radio ultra-wide band (IR-UWB) systems.



GIUSEPPE CANTARELLA (Member, IEEE) received the M.Sc. degree in micro and nanotechnology for ICTs from the Polytechnic of Turin, Italy, in 2013, and the Ph.D. degree in electrical engineering from ETH Zurich, Zurich, Switzerland, in 2018. He was Postdoctoral Fellow with ETH Zurich and the Free University of Bozen-Bolzano, Italy. He has been appointed as an Assistant Professor with the Free University of Bozen-Bolzano, in February 2020. His research interests include design, fabrication, and characterization of flexible and printable electronics.

GIORGIO ZINI received the High School Diploma from the Liceo Scientifico L. Respighi, Piacenza, in 1978. Since 1979, he has been developing a deep knowledge in electronic design and research and development management working as an employee in several companies, such as Computer Application Engineering and Silverstar Ltd., Milan, Ansaldo S.p.A., Esacontrol and Orsi Automazione s.r.l., Genova, and Rubbini s.r.l., Bologna). He is also a member of a Research and Development company, LUZ elettronica, Piacenza. He worked as a consultant for prestigious firms and organizations like Siemens S.p.A., DESY Accelerator Consortium, British Petroleum, Enel, Ferrovie Italiane, and Università di Genova. He is currently a Senior Electronic Designer with the Electronic Design Laboratory (EDL), Istituto Italiano di Tecnologia (IIT), Genova. His experience covers many fields in electronic engineering: microprocessor and embedded microcontroller systems, analog data acquisition systems, digital signal processors, FPGA and CPLD devices, high speed data transmission in local area and wide area networks, switched mode power supplies, PWM motor controllers, very-low power battery operated and solar-cells powered devices. During his 39 year career he successfully designed (or participated in the development of) hundreds of electronic systems ranging from small pocket devices, to large industrial plants.

ALESSANDRO BARCELLONA received the degree in engineering *(summa cum laude)* in automation engineering from the University of Palermo, in 2007. He received a research grant in biophysics field from the National Research Council (CNR) of Palermo for the hardware and software design of the instrumentation electronic control and command for atomic force microscopy (AFM) on living cells. He joined the Nanophysics Department, Istituto Italiano di Tecnologia (IIT). He is currently a Technician of the Electronic Design Laboratory Facility. His main activities regarded the hardware and software design and implementation of electronic devices for open source and automated microscopy imaging systems. His research interests include hardware design, short-term electrical and physical design, reworking, and assembly management.

MARIA RIVOLA received the bachelor's degree in food technology and the master's degree in food science and technology from the University of Bologna, Cesena, Italy, in 2016 and 2018, respectively. She is currently pursuing the Ph.D. degree in food engineering and biotechnology with the Free University of Bozen-Bolzano, Italy. Her M.Sc. thesis was carried at University of Granada, Spain. She carries out her research at the Sensing Technologies Laboratory. Her Ph.D. research project is focusing on quality and safety monitoring along food industry production using electrical impedance spectroscopy.

MATTIA PETRELLI (Student Member, IEEE) received the bachelor's degree and the master's degree in electronic engineering from the University of Salerno, in 2012 and 2017, respectively. He is currently pursuing the Ph.D. degree with the Sensing Technologies Laboratory, Free University of Bozen-Bolzano, Bolzano, Italy. He is enrolled in the Advanced-Systems Engineering Ph.D. program. His project is aimed at the design, fabrication and characterization of real-time wearable solutions to monitor the muscles' activity, to provide immediate feedback to the athlete about their physiological status.

LUISA PETTI (Member, IEEE) received the M.Sc. degree in electronic engineering from the Politecnico di Milano, Milan, Italy, in 2011, and the Ph.D. degree in electrical engineering and information technology from ETH Zürich, Zürich, Switzerland, in 2016. She is currently an Assistant Professor with the Free University of Bozen-Bolzano, Bolzano, Italy, where she researches on flexible and printed electronics for a wide range of applications, including food engineering and biotechnology.

PAOLO LUGLI (Fellow, IEEE) graduated in physics from the University of Modena and Reggio Emilia, Italy, in 1979. He received the M.Sc. and Ph.D. degrees in electrical engineering from Colorado State University, Fort Collins, CO, USA, in 1982 and 1985, respectively. He is currently the Rector of the Free University of Bozen-Bolzano. He has authored more than 350 scientific articles.