INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Impedance Measuring System based on a dsPIC

José Carlos Sequeira Martins dos Santos

Dissertation submitted for obtaining the degree of
Master in Electronic Engineering

Jury

President : Prof. Moisés Simões Piedade
Supervisor: Prof. Pedro Miguel Pinto Ramos
Members: Prof. Rui Manuel Rodrigues Rocha

November 2009

Acknowledgements

Above all, I would like to thank my parents, my sister and the rest of my family for the patience and support demonstrated throughout these five long years of the course. Without them I could not have made it so far.

I want to thank my coworkers and friends for always being there to help me in the difficult moments and for providing an excellent work environment. They are certain one of the best group of people I ever had the pleasure of meeting.

I thank Professor Pedro Ramos for the support and encouragement given throughout the work, and for giving me the opportunity to develop it.

Finally I would like to thank Professor Moisés Piedade for is undying efforts for creating the necessary conditions for all to develop their work.

Thank you all.

Abstract

Impedance measurement is currently one of the research and development areas under the work developed by the Instrumentation and Measurement Group of the Institute of Telecommunications. The existing measurement systems are either too costly or don't have enough accuracy and make measurements in a reduced frequency range.

This work is related to a project that is underway: *IMPANA – Development and implementation of an impedance analyzer based on analog to digital conversion and powerful signal processing algorithms* (Financed by the Foundation for Science and Technology, PTDC/EEA-ELC/72875/2006 from 01-Sep-2006 to 31-Aug-2010).

With the execution of this work one seeks to obtain a device capable of making impedance measurements at multiple frequencies. The device is based on a dsPIC (Digital Signal Peripheral Interface Controller) from Microchip as a central processing unit. A DDS (Direct Digital Synthesizer) is used to generate the stimulus signal of the measurement circuit, which has the reference impedance and the impedance under measurement. The voltage across the two impedances is amplified by programmable gain instrumentation amplifiers, whose outputs will then be digitized by analog to digital converters. The signal processing for the determination of the unknown impedance is performed using a sine-fitting algorithm. The dsPIC is connected through RS-232 to a personal computer (PC), where the user can monitor the resulting samples of the measurement and view the measurement results.

Keywords: Impedance measurement, dsPIC, ADC, *Sine-fitting*

Resumo

A medição de impedâncias é actualmente uma das áreas de investigação e desenvolvimento, no âmbito dos trabalhos desenvolvidos pelo Grupo de Instrumentação e Medidas no Instituto de Telecomunicações. Os sistemas de medição existentes ou são demasiado dispendiosos ou não têm exactidão suficiente e efectuam medições numa gama de frequências reduzida.

Este trabalho enquadra-se num projecto que se encontra a decorrer: *IMPANA – Development and implementation of an impedance analyzer based on analog to digital conversion and powerful signal processing algorithms* (Financiado pela Fundação para a Ciência e Tecnologia, PTDC/EEA-ELC/72875/2006 de 01-Set-2006 a 31-Ago-2010).

Com a execução deste trabalho pretende-se obter um dispositivo capaz de efectuar medições de impedâncias a diversas frequências. O dispositivo é baseado num dsPIC (*Digital Signal Peripheral Interface Controller*) da *Microchip* como unidade central de processamento. É utilizado um DDS (*Direct Digital Synthesizer*), sendo responsável por gerar o sinal de estímulo do circuito de medição, que tem a impedância de referência e a impedância a medir. As tensões aos terminais das duas impedâncias são amplificadas por amplificadores de instrumentação de ganho programável, cujas saídas são digitalizadas por conversores analógico-digitais. O processamento de sinal necessário para a determinação da impedância desconhecida é efectuado recorrendo ao algoritmo de adaptação de sinusóides. O dsPIC é ligado através de RS-232 a um computador pessoal (PC), onde o utilizador pode monitorizar as amostras resultantes da medição e visualizar os resultados da medição.

Palavras – Chave: Medição de impedâncias, dsPIC, ADC, *Sine-fitting*

Table of Contents

List of Figures

List of Tables

Acronyms

Acronym	Meaning
ADC	Analog to Digital Converter
ASCII	American Standard Code for Information Interchange
DAC	Digital to Analog Converter
DC	Direct Current
DDS	Direct Digital Synthesizer
DSC	Digital Signal Controller
DSP	Digital Signal Processor
dsPIC	Digital Signal Peripheral Interface Controller
FFT	Fast Fourier Transform
IDE	Integrated Development Environment
IpDFT	Interpolated Discrete Fourier Transform
MCU	Microcontroller
MIPS	Million Instruction Per Second
OP-AMP	Operational Amplifier
PC	Personal Computer
PCB	Printed Circuit Board
PGIA	Programmable Gain Instrumentation Amplifier
RAM	Random Access Memory
ROM	Read Only Memory
SPI	Serial Peripheral Interface
THD	Total Harmonic Distortion

1 – Introduction

1.1 – Impedance Concepts Overview

Impedance is an important parameter used to characterize electronic circuits, components, and the materials used to make components [1]. Impedance (Z) is generally defined as the total opposition a device or circuit offers to the flow of an alternating current (AC) at a given frequency, and is represented as a complex quantity which can be graphically represented on a vector plane – Fig. 1.1. An impedance vector consists of a real part (resistance, R) and an imaginary part (reactance, X).

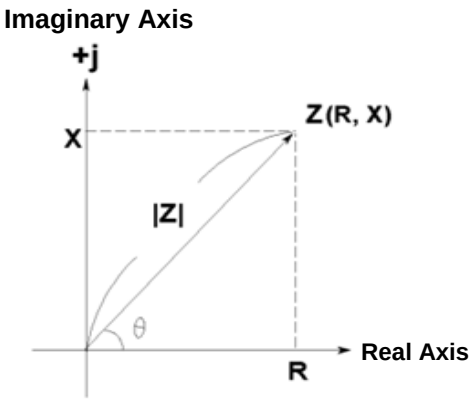


Fig. 1.1 - Impedance vector plane (taken from [1]).

Impedance can be expressed using the rectangular form,

$$Z = R + jX \tag{1.1}$$

or the polar form as a magnitude and phase angle,

$$Z = |Z| \angle \theta \tag{1.2}$$

Equations (1.3) to (1.6) show the mathematical relationships between R, X, |Z| and θ. The unit of impedance is the ohm (Ω).

$$R = |Z| \cos\theta \tag{1.3}$$

$$X = |Z| \sin\theta \tag{1.4}$$

$$|Z| = \sqrt{R^2 + X^2} \tag{1.5}$$

$$\theta = \tan^{-1}\left(\frac{X}{R}\right) \tag{1.6}$$

Reactance can take two forms: inductive (X_L) and capacitive (X_C). Fig. 1.2 represents the two possible forms of reactance and the correspondent representation in the impedance vector plane for a given frequency.

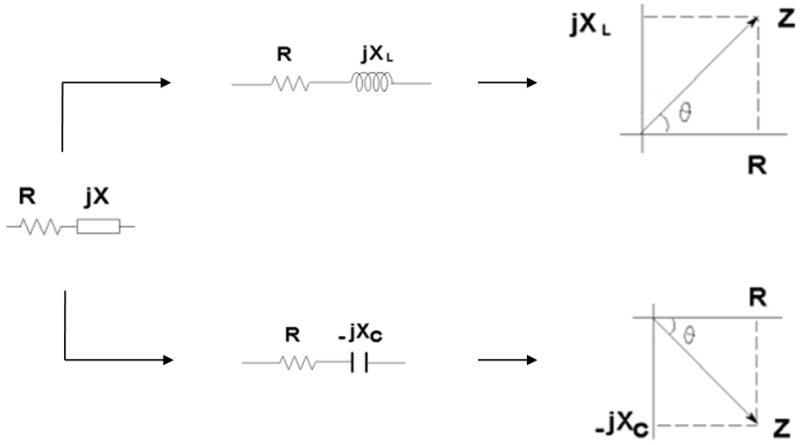


Fig. 1.2 - Representation of inductive and capacitive reactance in the impedance vector plane.

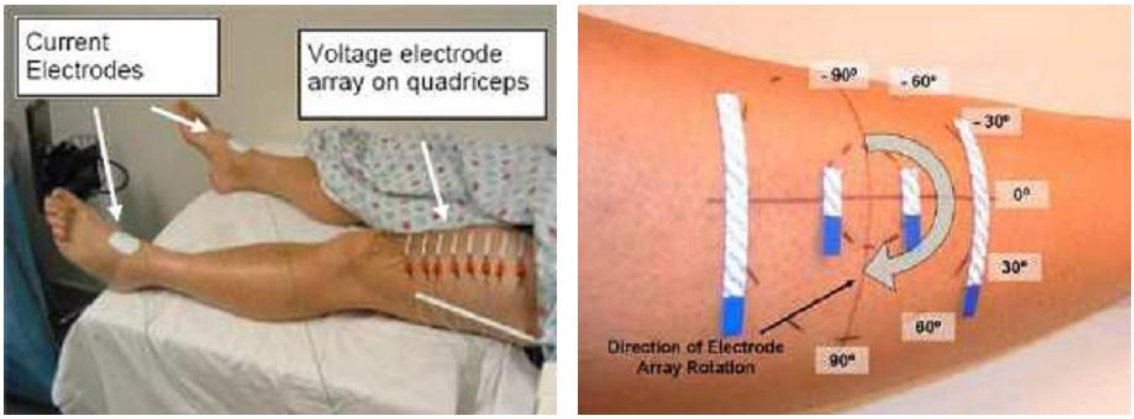
1.2 – Applications of Impedance Measurements

As the world of technology evolves, so does its need for improved measurement techniques. One of the major areas of investigation that has been growing for the past few years is impedance measurement. With a wide spectrum of applications, from medicine to electronics, impedance measurement is surely an area worth researching.

In the field of medicine, impedance measurement can be used in electrical impedance myography (EIM), which is a non-invasive technique for neuromuscular assessment [2], [3]. It enables the detection of degenerative neuromuscular diseases. Measurements can be taken painlessly and non-invasively using electrodes that are placed on top of the skin. They can also be taken from different parts of the body at the bedside. In this technique, a low-intensity alternating current is applied to a muscle and the consequent voltage difference is evaluated. From these measurements one can calculate the impedance of the underlying muscle and make inference about its structure.

There are two main methods used in this technique: linear EIM and rotational EIM. Linear EIM involves placing multiple pairs of voltage electrodes in a line directly above the muscle or muscle group of interest – Fig. 1.3(a). The current electrodes are placed so that current must flow in parallel to this line. The voltages at each point along the muscle are compared during analysis. Both single frequency (typically 50 kHz) and multi-frequency measurements can be taken.

Rotational EIM measures impedance at multiple angles relative to the direction of muscle fibers – Fig. 1.3(b). Both the current and voltage electrodes need to be rotated as they must be kept in line with each other at all angles to control the direction of current flow. For this method multi-frequency measurements are usually taken and then the average phase across all frequencies is compared at each different angle.



(a) (b)

Fig. 1.3 - An example of linear (a) and rotational (b) EIM measurements being performed on a patient (taken from [3]).

Impedance measurement can also play an important part in civil engineering. It can be used to measure the level of corrosion in some materials such as reinforced steel in concrete [4]. A concrete Surface-based Measurement Method (SMM) is used as nondestructive technique to determine the deterioration of structures and study the corrosion process. The method is able to determine the corrosion state of the reinforcing bars, as well as the resistivity of the concrete itself, from the concrete surface and with no need of connection to the reinforcement. It uses an array of four electrodes, equally spaced, and placed on the surface of the concrete to indirectly measure the frequency-dependent impedance between the reinforcing steel bars and the concrete – Fig. 1.4. Alternating current, varied over a wide range of frequencies, is injected between the two outer electrodes, and the voltage difference produced by the current flow is measured across the two inner electrodes. This voltage presents an angle phase shift with respect to the current. As the current is injected into the concrete, the steel bar surface becomes polarized creating a net dipole which is added to the voltage read by the inner electrodes – Fig. 1.5.

It is important to notice that the measured impedance on the concrete surface does not reflect directly the interfacial impedance, because it is influenced by the concrete resistivity. However, a change in the corrosion state of a steel bar, will be reflected in the measurement on the concrete surface, and given the concrete resistivity and measurement geometry, the impedance response is unique to each interfacial impedance.

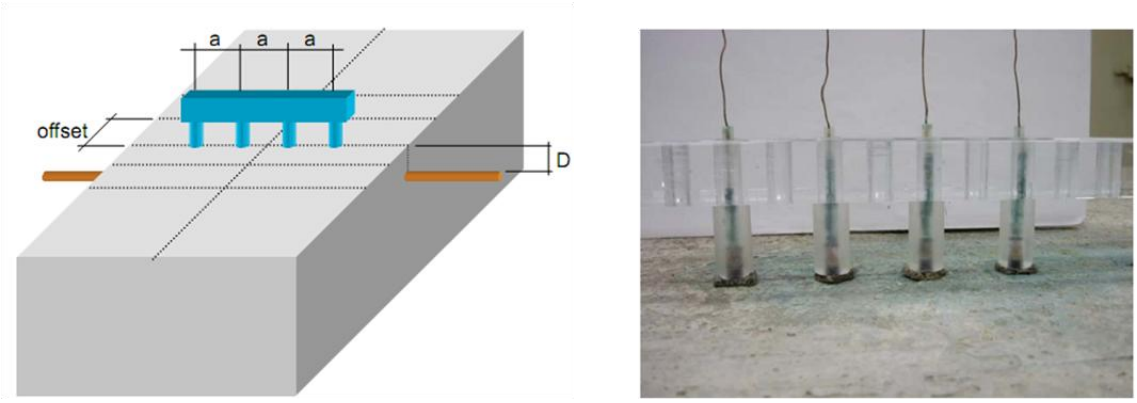


Fig. 1.4 - Four-electrode array placed on the concrete surface (taken from [4]).

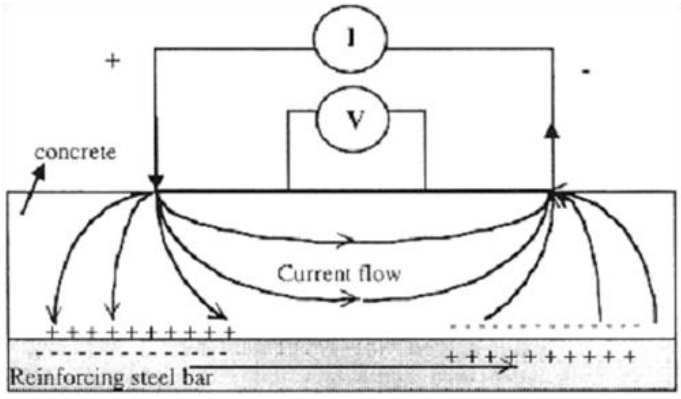


Fig. 1.5 - Net dipole created by the four-electrode array (taken from [4]).

1.3 – Motivation

The technological advances of the past few years have improved the performance of data acquisition systems to a point where it is possible to implement cheaper ways to make the same measurements as dedicated expensive equipment, without relevant degradation of the system accuracy.

The continuous improvement in performance of new devices, such as Analog to Digital Converters (ADCs), combined with more powerful signal processing techniques, are some of the factors that have contributed to this fact.

Typical measuring techniques/instruments are either too costly, or don't have enough accuracy and have a reduced frequency range. Agilent Technologies is one of the leading companies for measurement systems. Its simplest impedance measuring system [5] covers frequencies from 40 Hz up to 110 MHz with a basic accuracy of 0.08 %. However, its price of nearly 30.000€ makes the system well beyond the budget capabilities of many companies and research institutes.

Therefore there has been an increasing demand for low cost impedance measuring systems capable of performing measurements at a broad range of frequencies but still with comparable accuracy to that of sophisticated impedance measurement equipment.

1.4 – Work Goals

This work seeks to develop, implement and characterize a low-cost device capable of measuring impedances at a wide range of frequencies by using ADCs and a dsPIC as a central processing unit to both control all the necessary hardware of the measuring circuit and run the sine fitting algorithm necessary to the proper measurement of the impedance. A personal computer is also used, which acts as interface between the user and the device, allowing the monitoring of the acquired samples.

The device has to be able to measure impedances in the frequency range from 500 Hz to 200 kHz. The amplitude of the impedances to be measured is intended to be between 100 Ω and 10 kΩ.

The dsPIC is responsible for: i) controlling the stimulus module of the measurement circuit through a DDS which is responsible for generating the sine signal with the measurement frequency; ii) controlling the programmable gain instrumentation amplifiers (PGIAs) that amplify the voltage at the terminals of the two impedances (reference impedance and impedance under measurement); iii) controlling and communicating with the ADCs responsible for the digitalization of the output voltage from the PGIAs; iv) applying the algorithm to determine the amplitudes and phases of the sinusoidal signals across each impedance (sine-fitting); v) applying the algorithms to correct systematic errors; vi) transmitting the results to a PC through a RS-232 connection.

In the end the user will be able to control the device in order to decide at which frequency he wants to perform the impedance measurement.

1.5 - Thesis Organization

This thesis is organized as follows:

In Section 2, the State Of The Art for impedance measurement systems is presented. It is given an overview of the existing impedance measurement methods. A few points regarding sine-fitting algorithms are discussed and finally it is presented a new impedance measurement method that is currently under development.

In Section 3 it is presented the system's architecture, where the measurement circuit and method is explained, along with all the hardware used in the system.

Section 4 describes in detail all the algorithms implemented, for proper measurement of the unknown impedance.

In Section 5 the implemented control software running on a PC is described.

In Section 6 the experimental results obtained by the device are compared to the results obtained with a commercially available device.

The conclusions of the work are presented in Section 7.

Finally, Section 8 describes the proposed future work.

2 – State Of The Art

Impedance measurement systems are used to test electronic components, circuits and materials. They can also be used to measure sensor's outputs that translate physical quantities into electrical impedance.

2.1 – Impedance Measurement Methods

Several impedance measurement methods exist. Traditional methods include: bridge, resonant, I-V, RF I-V, network analysis and auto balancing bridge [1].

The bridge method applies the setup presented in Fig. 2.1.

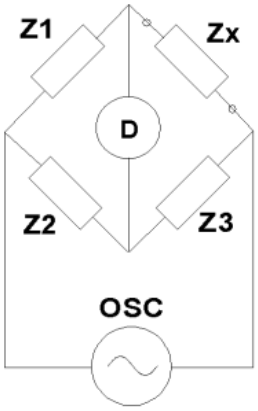


Fig. 2.1 - Bridge method setup (taken from [1]).

When no current flows through the detector (D), the value of the unknown impedance Zx can be obtained by the relationship of the other bridge elements by

$$Zx = \frac{Z1}{Z2} Z3\,. \tag{2.1}$$

Various types of bridge circuits, employing combinations of L, C, and R components as the bridge elements, are used for various applications.

This is a low cost method with a wide frequency coverage (DC to 300 MHz) by using different types of bridges. However, it needs to be manually balanced by experienced technicians.

In the resonant method, the setup from Fig. 2.2 is used.

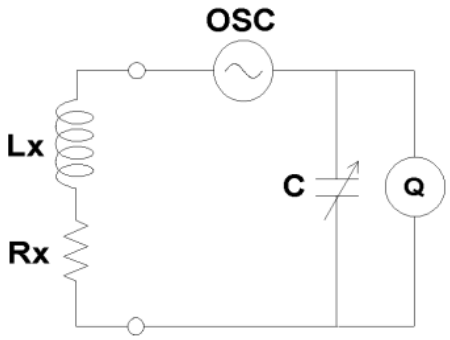


Fig. 2.2 - Resonant method setup (taken from [1]).

When the circuit is adjusted to resonance by adjusting the tuning capacitor C, the unknown impedance Lx and Rx values are obtained from the test frequency, the C value, and the Q value. Q represents the quality factor of the inductance. Q is measured directly using a voltmeter placed across the tuning capacitor. Because the loss of the measurement circuit is very low, Q values as high as 1000 can be measured.

It presents a good Q accuracy up to high Q, but the need for tuning to resonance and its low impedance measurement accuracy are clear disadvantages of this method. It has an applicable frequency range from 10 kHz to 70 MHz.

The I-V or volt-ampere method has the measurement setup presented in Fig. 2.3.

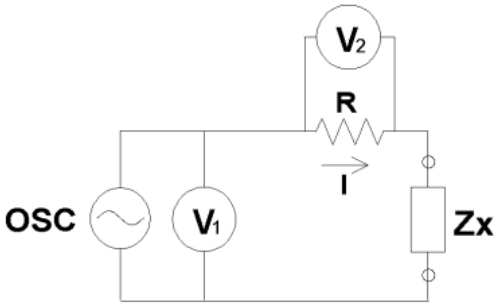


Fig. 2.3 - I-V method setup (taken from [1]).

The unknown impedance Zx can be calculated by using the measured voltages V_1 and V_2 – equation (2.2).

$$Zx = \frac{V_1}{I} = \frac{V_1}{V_2} R \tag{2.2}$$

The current value I is calculated using the voltage measurement (V_2) across an accurately known low value resistor, R. In practice a low-loss transformer is used in place of R to prevent the effects caused by placing a low value resistor in the circuit. The transformer, however, limits the low end of the applicable frequency range (10 kHz to 100 MHz).

This method has the advantage of being able to perform a grounded device measurement and is suitable for probe type test needs.

The measurement setup for the RF I-V method is presented in Fig. 2.4.

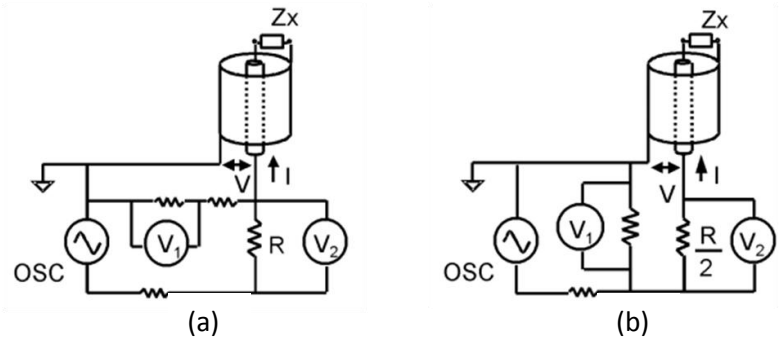


Fig. 2.4 - RF I-V method. (a) Low impedance measurement; (b) High impedance measurement (taken from [1]).

The RF I-V measurement method is based on the same principle as the I-V method, but it is configured in a different way by using an impedance matched measurement circuit (50 Ω) and a precision coaxial test port for operation at higher frequencies. There are two types of configurations which are suited for low impedance – Fig. 2.4(a), and high impedance – Fig. 2.4(b), measurements. The impedance Zx is derived from the measured voltage values V_1 and V_2, as illustrated. Equation (2.3) refers to the low impedance measurement arrangement and (2.4) refers to the high impedance measurement arrangement.

$$Zx = \frac{V}{I} = \frac{2R}{\frac{V_2}{V_1} - 1} \tag{2.3}$$

$$Zx = \frac{V}{I} = \frac{R}{2}\left(\frac{V_1}{V_2} - 1\right) \tag{2.4}$$

Similarly to the I-V method, the current that flows through Zx is calculated from the voltage measurement across a known low value resistor, R. Just like before, in practice, a low loss transformer is used in place of the low value resistor, R. The low end of the frequency range of this method (1 MHz to 3 GHz) is limited by this transformer.

This method presents a high accuracy (typically 1%) and a wide impedance range at high frequencies.

The network analysis method has the measurement setup presented in Fig. 2.5.

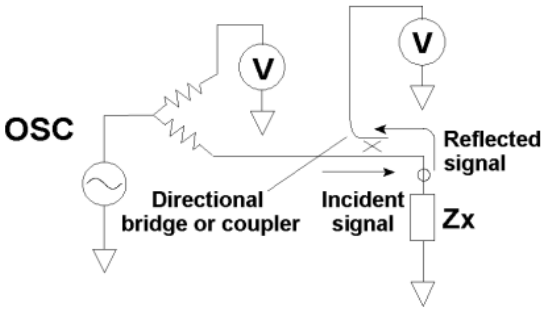


Fig. 2.5 - Network analysis method setup (taken from [1]).

In this method the reflection coefficient is obtained by measuring the ratio between the incident signal and the reflected signal. A directional coupler or bridge is used to detect the reflected signal and a network analyzer is used to supply and measure the signals. Since this method measures reflection at the impedance Zx, it is usable in a higher frequency range (300 kHz and above).

This method presents a good accuracy when the unknown impedance is close to the characteristic impedance of the network, but it has the disadvantage of requiring a recalibration procedure when the measurement frequency changes. It also has a narrow impedance measurement range.

The auto balancing bridge method setup is shown in Fig. 2.6.

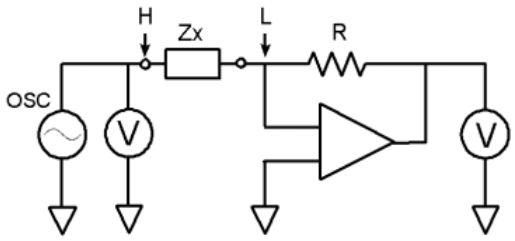


Fig. 2.6 - Auto balancing bridge setup (taken from [1]).

The current, flowing through the impedance Zx, also flows through the resistor R. The potential at the “L” point is maintained at zero volts (thus called a “virtual ground”), because the current through R balances with the Zx current by operation of the I-V converter amplifier. The Zx impedance is calculated by using the voltage measured at the “H” point and the voltage across R.

In practice, the configuration of the auto balancing bridge differs for each type of instrument. Generally LCR meters, in a low frequency range typically below 100 kHz, employ a simple operational amplifier for its I-V converter. This type of instrument has a disadvantage in accuracy, at high frequencies, because of the amplifier’s limits in terms of performance. Wideband LCR meters and impedance analyzers employ an I-V converter consisting of sophisticated null detector, phase detector, integrator (loop filter) and vector modulator to ensure a high accuracy for a broad frequency range over 1 MHz. This type of instrument can

attain a maximum measurement frequency of 110 MHz. These instruments have a minimum measurement frequency of about 20 Hz.

This method has a high accuracy over a wide impedance measurement range and is capable of performing a grounded device measurement.

Agilent Technologies, one of the leading companies of the instrumentation market, has several commercially available products that apply some of the methods described above. The 4287A RF LCR Meter [6] uses an RF I-V method, the 4395A Network/Spectrum/Impedance Analyzer [7] uses a network analysis method and the 4294A Precision Impedance Analyzer [5] uses an auto balancing bridge method.

2.2 – Sine-Fitting Algorithms

The objective of sine-fitting algorithms is to obtain a set of parameters corresponding to the analytical expression of a sine signal while minimizing the sum of the squared errors between the estimated parameters and the acquired samples.

The three-parameter and the four-parameter sine-fitting algorithms were standardized in [8]. The first is a non-iterative algorithm that with the knowledge of the frequency estimates the amplitude, the phase and the DC component of the signal. The second also estimates the frequency of the signal, but requires an iterative process to obtain the best estimation of the parameters. Because it uses an iterative process, the initial estimates of the parameters are critical for the convergence of the algorithm. In applications that use two channels with a common frequency, which is the case of this work, all the information from both records should be used to obtain a better estimative of the frequency. This can be done using a seven-parameter sine-fitting algorithm [9], which is an iterative algorithm similar to the four-parameter algorithm. The difference is that the seven-parameter algorithm estimates the amplitude, phase and DC component of both signals, plus the common frequency. The effectiveness of this algorithm was demonstrated in [10], when compared with the application of two four-parameter algorithms.

The seven-parameter sine-fitting algorithm was the selected algorithm for the purposes of this work and it will be described in more detail later on.

2.3 – New Impedance Measurement Method

In many applications it is necessary to accurately measure an unknown impedance corresponding to a sensor output. In [11] a new method for impedance measurements using DSP-based ellipse-fitting algorithms is described. This method uses a sine wave generator to supply current to the serial circuit composed by the reference impedance and the impedance to be measured. It also uses two instrumentation amplifiers that sense the differential voltages at the reference impedance and the impedance to be measured, a two channel ADC that digitizes the signals coming from the instrumentation amplifiers and a DSP (Digital Signal Processor) as

a central processing unit. A personal computer is also used to communicate with the DSP through an RS-232 connection. Fig. 2.7 shows the measurement setup.

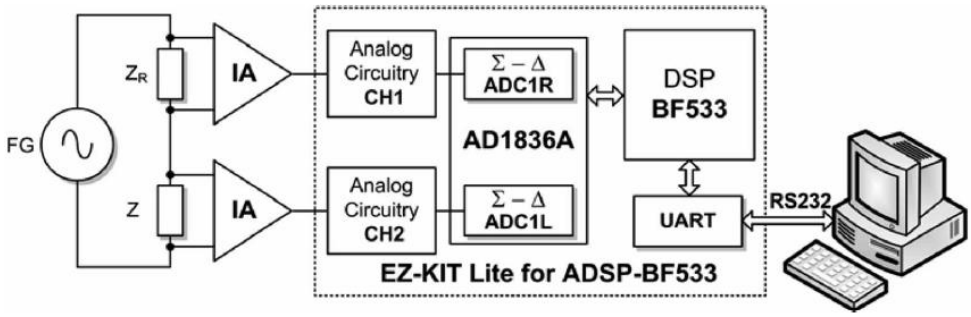


Fig. 2.7 - Measurement setup (taken from [11]).

In [11] the algorithm used to perform the signal processing is the ellipse-fitting algorithm. The ellipse-fitting algorithm is not iterative since it does not estimate the frequency of the signals. Because the frequency is imposed by the generator, there is no need to estimate it. This makes the ellipse-fitting algorithm very efficient, and since it can be modified from its original form [12] to have very low memory requirements, it turns out to be a very fast signal processing algorithm allowing the system to perform high speed measurements.

This methodology, given the nature of the devices used in the system, is a low-cost alternative to dedicated impedance measurement equipment, but with comparable accuracy.

This thesis applies a measurement method similar to that described above, except for the algorithm responsible for the signal processing. In the case of this thesis the selected algorithm is the seven-parameter sine-fitting algorithm. The measurement setup used in this thesis will be explained in detail later on.

3 – System's Architecture

3.1 – Measurement Circuit and Method

The impedance measurement circuit used to ascertain the value of the unknown impedance is presented in Fig. 3.1.

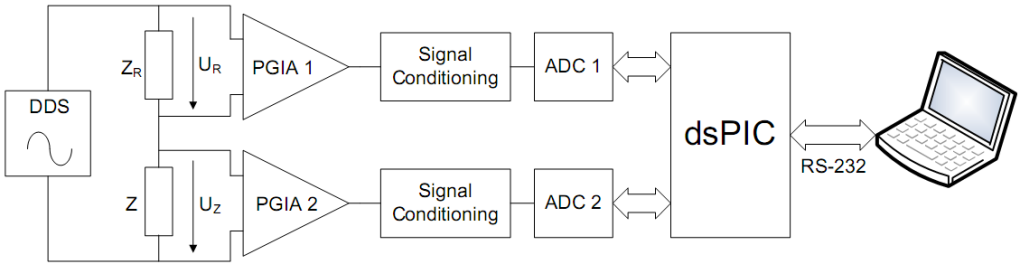


Fig. 3.1 - Measurement circuit. Z is the impedance under measurement and Z_R is the reference impedance.

The measurement method is based in the volt-ampere method [1], in which the module of the unknown impedance can be determined by knowing the module of the reference impedance and the ratio between the amplitude of the sine signal across the impedance under measurement and the amplitude of the sine signal across the reference impedance. The phase angle of the unknown impedance is calculated based in the phase angle of the reference impedance and the phase difference between the sine signal across the impedance under measurement and the sine signal across the reference impedance [13]. The method used is similar to that in [11], but in this case a dsPIC is used as a processing unit and a DDS is used to inject current in the circuit.

The measurement procedure is done according to the following steps:

(i) The dsPIC controls the DDS in order to generate a sine wave with the user-defined measurement frequency.

(ii) The dsPIC selects the gains of the amplifiers according to the magnitudes of the signals across the reference impedance and the impedance under measurement.

(iii) The dsPIC controls the ADCs to acquire simultaneously the samples (in this case 1024 per channel) of the output voltage of the PGIAs, which amplify the signals across the reference impedance (PGIA 1) and the unknown impedance (PGIA 2). The gains of both amplifiers are set by the dsPIC to maximize the amplitudes of the signals, but with care to avoid saturation of the digitizing channels.

(iv) A seven parameter sine-fitting algorithm is applied in order to determine the amplitudes $|U_x|$ and phases ϕ_{U_x} of the sine waves that best fit the acquired samples of both channels ($x = Z, R$).

(v) The modulus and phase angle of the impedance under measurement are calculated by

$$|Z| = |Z_R| \times \frac{|U_Z|}{|U_R|} \tag{3.1}$$

$$\phi_Z = \phi_{Z_R} + \left(\phi_{U_Z} - \phi_{U_R}\right) \tag{3.2}$$

Once all the calculations are done, the experimental results of the measurement are transmitted to the PC.

The entire measurement circuit is represented in the schematic from Fig. C.1 in appendix C.

3.2 – Processing Unit

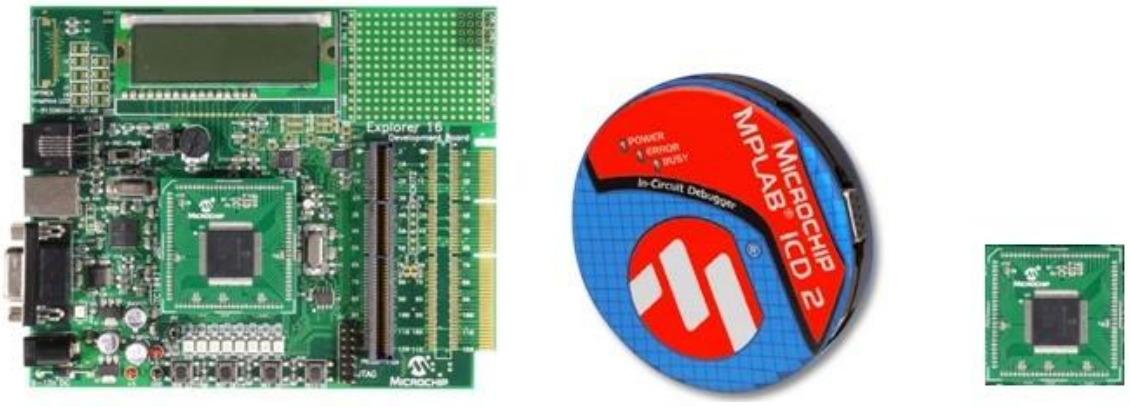
The processing unit used in the device is a dsPIC which is a Digital Signal Controller (DSC). A DSC is a single-chip, embedded controller that integrates the control attributes of a Microcontroller (MCU) with the computation and throughput capabilities of a Digital Signal Processor (DSP) in a single core. In terms of performance and price, a dsPIC is somewhere in the middle of the PIC MCU (in the low end) and the DSP (in the high end), and so it achieves the best of both worlds making it perfect for this embedded control application.

The dsPIC used in this work is a 16-bit dsPIC33FJ256GP710 microcontroller from Microchip - Fig. 3.2(c). It has a throughput up to 40 MIPS, 256 kB of flash program memory and 30 kB of RAM (Random Access Memory). It also includes two 12-bit ADCs capable of conversion speeds up to 500 kS/s.

The Explorer 16 Development Board from Microchip - Fig. 3.2(a), in which the dsPIC is connected to, is used as a prototyping tool to help implement the device. The board allows easy connection to an MPLAB ICD 2 In-Circuit Debugger - Fig. 3.2(b). This is a development tool allowing the programming of the dsPIC, and is capable of debugging ASM and C source code. The application code for the device is written in C language on the MPLAB Integrated Development Environment (IDE), and then it is loaded to the dsPIC using the MPLAB ICD 2.

The total program memory required for the execution of the developed code for this work is 16578 bytes, and the total data memory is 20850 bytes. These memory requirements are well within the memory capabilities of the dsPIC. The data memory required is so high given the nature of some of the algorithms used in this work. Each algorithm and their memory requirements will be explained in more detail in Section 4.

The hardware associated with the processing unit is represented in square 1 in the schematic from Fig. D.1 in appendix D.



(a) (b) (c)

Fig. 3.2 – Development tools for the work. (a) Explorer 16 Development Board; (b) MPLAB ICD 2 In-Circuit Debugger; (c) 16-bit dsPIC33FJ256GP710.

3.3 – Stimulus Module

The module used to stimulate the measurement circuit is a DDS. A DDS is an electronic programmable waveform generator, which produces a signal from a fixed-frequency clock source. The waveform is digitally generated by using sampling techniques. Typically it is used to generate a sine wave, although it can be used to synthesize any other periodic waveforms such as triangular and square signals.

A DDS is basically composed by a phase accumulator, a phase to amplitude converter and a digital to analog converter (DAC). A low-pass filter may also be included to eliminate unwanted high frequency spectral components from the analog output. Fig. 3.3 shows the block diagram of a general DDS.

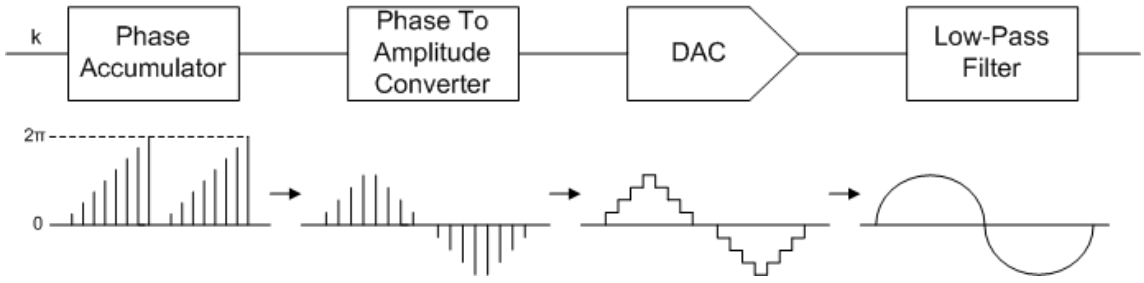


Fig. 3.3 - Block diagram of a general DDS.

The DDS operation consists of the following: the phase accumulator is incremented by using a clocked binary adder, and each value corresponds to a specific phase value between 0 and 2π. The phase to amplitude converter translates these values into a digital representation of the desired waveform by using, for example, a look-up table implemented in a ROM (Read Only Memory). The digital output of the phase to amplitude converter is then converted to an

analog output by the DAC. The phase information is used because unlike the magnitude, which is nonlinear and not easy to generate, the angular information of a sine wave is linear in nature. That is, the phase angle rotates through a fixed angle for each unit of time.

A DDS module can be very useful in various types of sensing, actuation, and modulation applications where it may be necessary to produce a locally controlled signal.

In this work the AD9833 from Analog Devices is used to produce the sine signal with the desired measurement frequency in order to inject current to the measurement circuit. It is capable of producing sine, triangle and square waveforms with an output frequency range from 0 MHz to 12.5 MHz, and the amplitude of the output signal is typically 0.6 Vpp. The module is also compatible with SPI interface, allowing its control by a microcontroller, in this case the dsPIC.

The DDS output sine signal with a frequency of 1 kHz and the correspondent FFT is presented in Fig. 3.4.

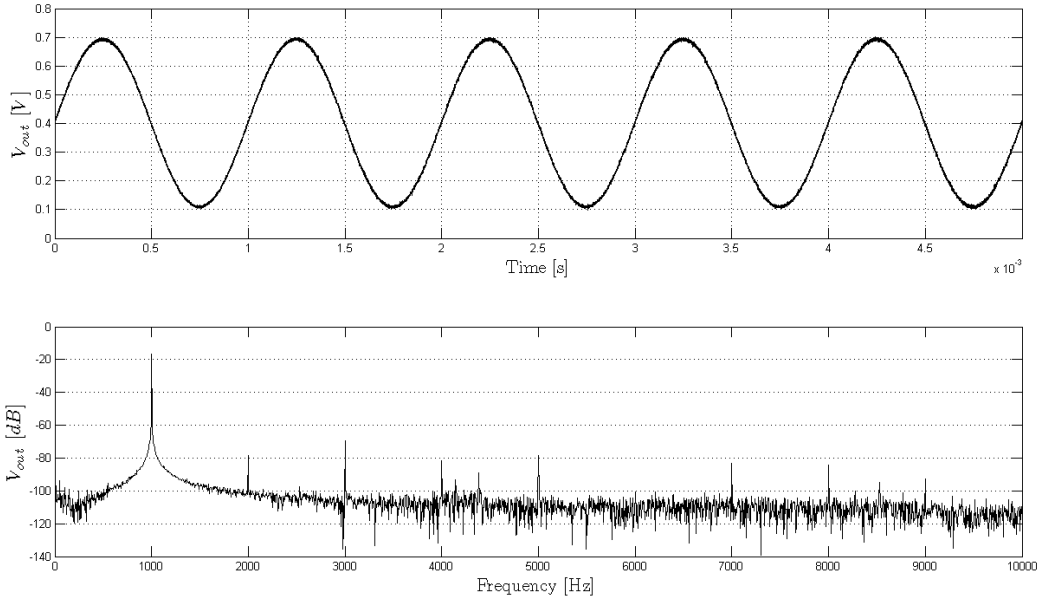


Fig. 3.4 - Output signal and correspondent FFT from the DDS at 1 kHz.

The signal was measured through an Agilent 54622D Mixed Signal Oscilloscope from Agilent Technologies. Two million samples of the signal were acquired and transferred to a computer and saved in a text file. In order to obtain the FFT of the signal, the samples were processed in the MATLAB environment where it was also possible to estimate the Total Harmonic Distortion (THD) of the sine signal. The estimated THD value is 51.175365 dB. The graphics represented in Fig. 3.4 were also plotted in MATLAB.

The AD9833 DDS does not produce a bipolar signal output, which means it has a DC component in the output signal. For the purposes of this work it was necessary to apply a bipolar signal to the measurement circuit, so it was necessary to implement a high-pass filter with a low cutoff frequency to eliminate the DC component. The implemented filter is presented in Fig. 3.5.

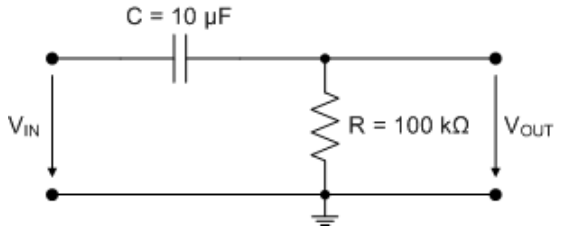


Fig. 3.5 - High-pass filter used to eliminate the DC component from the output signal of the DDS.

The cutoff frequency of the filter is given by

$$f_{cutoff} = \frac{1}{2\pi RC} = 0.159 \text{ Hz} \tag{3.3}$$

which is well beneath the range of frequencies intended for this work, 500 Hz to 200 kHz.

The delay of the implemented filter is not relevant for the range of frequencies considered in this work.

Because the magnitude of the output signal is so low (0.6 Vpp), it was decided to add an inverting amplifier circuit to the output of the filter to amplify the signal from the DDS. The implemented circuit is shown in Fig. 3.6.

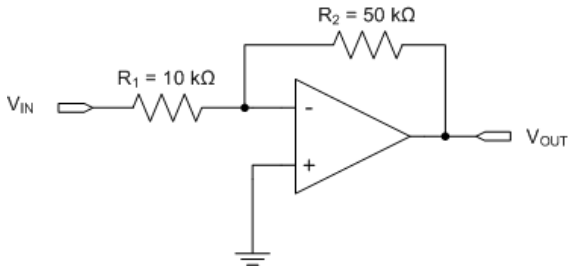


Fig. 3.6 - Inverting amplifier circuit used to add gain to the output signal of the DDS.

The gain of the inverting amplifier circuit is given by equation (3.4), which results in a signal magnitude of 3 Vpp.

$$A_V = \frac{V_{OUT}}{V_{IN}} = -\frac{R_2}{R_1} = -5 \tag{3.4}$$

The operational amplifier used was the OP484 from Analog Devices.

The closed-loop bandwidth of the amplifier is about 1 MHz, which is suitable for the frequency range used in this work.

The stimulus module of the measurement circuit is represented in square 1 in the schematic from Fig. C.1 in appendix C.

3.4 – Programmable Gain Instrumentation Amplifiers

An instrumentation amplifier is a differential-input amplifier [14], used for the accurate amplification of the difference between two voltages. It presents a high Common Mode Rejection Ratio (CMRR), allowing its use in circuits with high common-mode voltages and small differential voltages. It has a high input impedance, which allows its insertion in a measurement circuit with negligible influence over the circuit. Because of this, the instrumentation amplifier is perfect for data acquisition and measurement applications. It differs from an op-amp, which ideally has infinite open-loop gain and must be used in conjunction with external elements to define the closed-loop transfer function. Traditionally an instrumentation amplifier is composed by three OP-AMPs and its circuit is presented in Fig. 3.7.

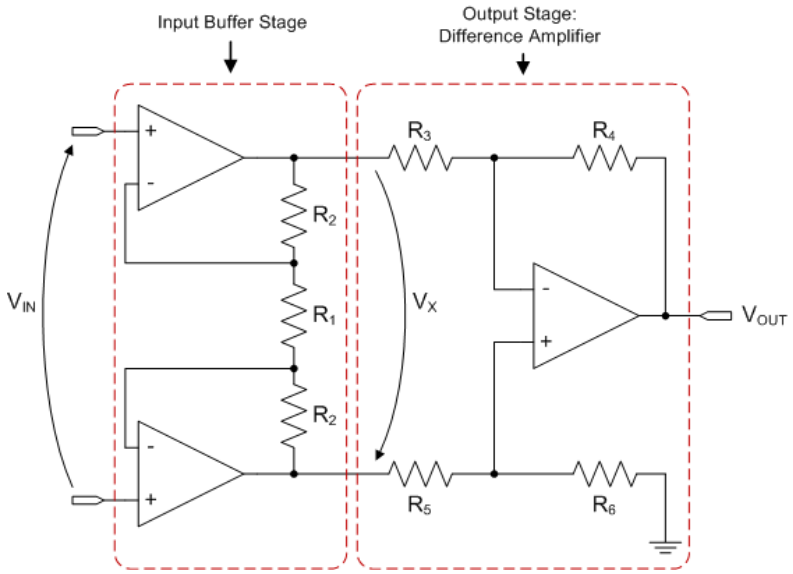


Fig. 3.7 - Typical instrumentation amplifier circuit.

If $\frac{R_3}{R_4} = \frac{R_5}{R_6} = k$, then

$$V_{OUT} = -k \times V_X \tag{3.5}$$

$$V_X = -\frac{2R_2 + R_1}{R_1} \times V_{IN} \tag{3.6}$$

$$V_{OUT} = k \times \frac{2R_2 + R_1}{R_1} \times V_{IN} \tag{3.7}$$

So, the overall gain of the instrumentation amplifier can be adjusted through the variation of a single resistor (R_1).

In this work an AD8250 programmable gain instrumentation amplifier from Analog Devices was used. It allows four programmable gains: 1, 2, 5 and 10. The gain is digitally set by

the processing unit in order to maximize the voltages at the ADCs inputs, while avoiding saturation of the digitizing channels. This is done in order to obtain a better resolution of the acquired signals.

Gain control in the AD8250 is achieved by switching resistors in an internal, precision resistor array. The $\overline{WR}$ (write enable), A0 (LSB gain selection pin) and A1 (MSB gain selection pin) digital inputs of the device control the desired gain.

There are two methods through which the gain of the AD8250 can be set: the transparent gain mode and the latched gain mode. In the transparent gain mode $\overline{WR}$ is tied to the negative supply voltage ($-V_S$) and any change in the voltage applied to A0 and A1 from logic low to logic high, or vice versa, immediately results in a gain change. Table 3.1 is the truth table for transparent gain mode.

Table 3.1 - Truth table for transparent gain mode.

$\overline{WR}$	A1	A0	Gain
$-V_S$	Low	Low	1
$-V_S$	Low	High	2
$-V_S$	High	Low	5
$-V_S$	High	High	10

In the latched gain mode the gain is set using $\overline{WR}$ as a latch. The voltages on A0 and A1 are read on the downward edge of the $\overline{WR}$ signal as it transitions from logic high to logic low. This latches in the logic levels on A0 and A1, resulting in a gain change. The advantage of the latched gain mode is that multiple devices can share A0 and A1 as single data bus and use the $\overline{WR}$ signal as a latch, which reduces the amount of pins necessary in a microcontroller.

The truth table for the latched gain mode is presented in Table 3.2. The timing diagram for the latched gain mode is presented in Fig. 3.8 and the digital timing specifications are listed in Table 3.3.

Table 3.2 - Truth table for latched gain mode.

$\overline{WR}$	A1	A0	Gain
High to Low	Low	Low	Change to 1
High to Low	Low	High	Change to 2
High to Low	High	Low	Change to 5
High to Low	High	High	Change to 10
Low to Low	X	X	No change
Low to High	X	X	No change
High to High	X	X	No change

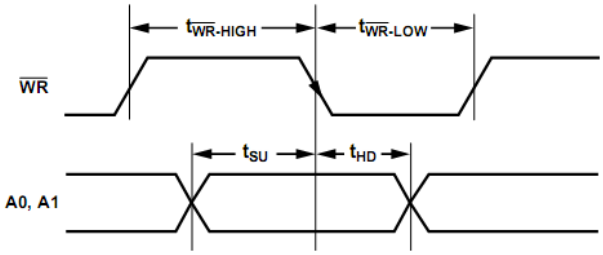


Fig. 3.8 - Timing diagram for latched gain mode (taken from [15]).

Table 3.3 - Digital timing specifications for latched gain mode.

Parameter	Description	Value (ns)
t_{SU}	Minimum setup time before the downward edge of $\overline{WR}$	20
t_{HD}	Minimum hold time after the downward edge of $\overline{WR}$	10
$t_{\overline{WR}-LOW}$	Minimum duration that $\overline{WR}$ can be held low	20
$t_{\overline{WR}-HIGH}$	Minimum duration that $\overline{WR}$ can be held high	40

For this work the chosen gain selection mode was the latched gain mode, because the transparent gain mode has the problem that any noise present in the digital inputs of the amplifier (A0 and A1) could cause an undesired gain change.

The PGIAs are represented in square 2 in the schematic from Fig. C.1 in appendix C.

3.5 – Analog to Digital Converters

An ADC is a device capable of converting an analog input voltage in a digital word, composed by a sequence of 0s and 1s. The maximum length of the digital output word corresponds to the number of bits of the converter.

At first, the ADCs used for the development of the work were the ones included in the dsPIC, which have a 12-bit resolution and are capable of conversion speeds up to 500 kS/s. This was done in order to gain experience with the acquisition of signals and to develop an algorithm capable of transmitting the acquired samples from the dsPIC to the PC for viewing purposes, so it would be possible to debug any possible errors with the signal processing.

After this step was complete, the external ADCs could be used. These external ADCs are the AD7980 from Analog Devices. The AD7980 is a successive approximation converter with 16-bit resolution and conversion speeds up to 1 MS/s.

The AD7980 is compatible with Serial Peripheral Interface (SPI) and also features the ability to daisy-chain several ADCs on a single data bus.

In this work, the two ADCs used were daisy-chained as represented in Fig. 3.9. The ADCs operate in what is referred as the chain mode without busy indicator [16].

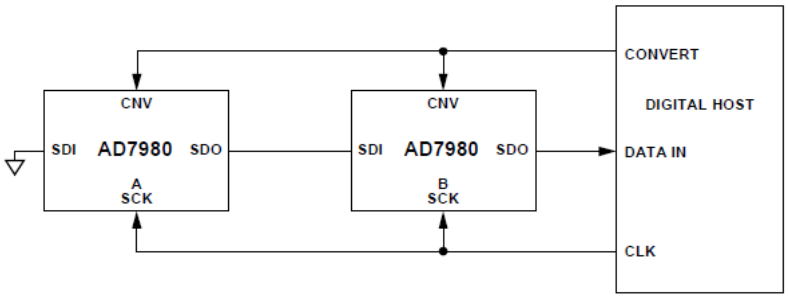


Fig. 3.9 - Simplified connection diagram of the two ADCs (taken from [16]).

In this mode, when SDI and CNV are low, SDO is driven low. With SCK low, a rising edge on CNV initiates a conversion, selects the chain mode, and disables the busy indicator. In this mode, CNV is held high during the conversion phase and the subsequent data read back. When the conversion is complete, the ADC outputs the MSB onto SDO and the AD7980 enters the acquisition phase and powers down. The remaining data bits stored in the internal shift register are clocked by subsequent SCK falling edges. Each ADC in the chain outputs its data MSB first, and 16 × 2 clocks are required to read back the 2 ADCs. The timing diagram for this mode of operation can be seen in the ADC's data sheet [16].

The sampling frequency of the two external ADCs is controlled by the dsPIC through the SPI interface. In order to be possible to use the ADCs maximum conversion speed of 1 MS/s and read the samples of both ADCs (32 bits), it would be necessary an SPI module with at least an operating frequency of 32 MHz, however at the dsPIC's maximum throughput, 40 MIPS, the SPI module can only operate at a maximum frequency of 10 MHz, and so the maximum sampling frequency possible for this work was greatly reduced. It can attain the maximum value of 178 kS/s.

For the purposes of this thesis it is necessary to acquire bipolar sinusoidal signals, which means it would be necessary a negative and a positive voltage reference for the ADC, but the AD7980 only accepts positive voltage references (as well as ground), and so it is necessary to do some signal conditioning in order to properly acquire the required signals. It was decided to use a circuit which adds a DC offset to the signal to acquire, enabling it to be between the voltage references of the ADC, thus removing the need for a negative voltage reference for the ADC. A differential amplifier circuit which is represented in Fig. 3.10 was used. This circuit is connected between the programmable gain instrumentation amplifier, which amplifies the signal to acquire, and the ADC input.

The operational amplifier used in the circuit from Fig. 3.10 was the OP484 from Analog Devices, which was the same kind used in the circuit from Fig. 3.6. As stated in Section 3.3, the closed-loop bandwidth of the amplifier is about 1 MHz, which is suitable for the frequency range used in this work.

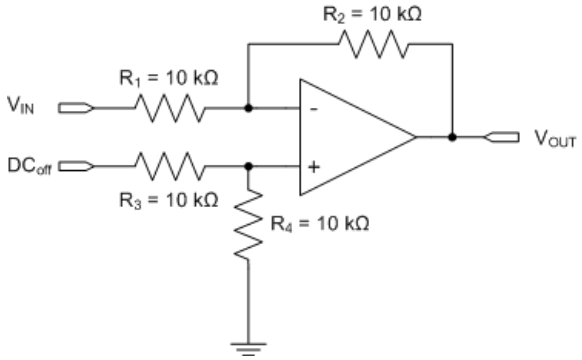


Fig. 3.10 - Circuit of a differential amplifier used to add a DC offset to the signal to acquire.

The output voltage of the circuit is given by

$$V_{OUT} = \frac{R_1 + R_2}{R_1}\frac{R_4}{R_3 + R_4} \times DC_{off} - \frac{R_2}{R_1} \times V_{IN} \tag{3.8}$$

If $\frac{R_1}{R_2} = \frac{R_3}{R_4}$, then

$$V_{OUT} = \frac{R_2}{R_1}\left(DC_{off} - V_{IN}\right) \tag{3.9}$$

Since all the resistors are of equal value, the output voltage is given by

$$V_{OUT} = DC_{off} - V_{IN} \tag{3.10}$$

From equation (3.10) it is evident that the signal to acquire (V_{IN}) will be inverted because of the circuit from Fig. 3.10, however since the signals from both PGIAs will be inverted and we are only interested in the phase difference between the two signals, there is no need to correct the inverting effect of the circuit from Fig. 3.10.

The ADCs are represented in square 3 in the schematic from Fig. C.1 in appendix C.

3.6 – Hardware for Selecting the Reference Impedances

In order to ensure that the amplitude of the acquired signals is as close as possible to the ADC input range, it is necessary to use different reference impedances according to the impedance to measure. This way it is possible to maximize the use of the ADC dynamic range. The proper reference impedance will be selected among a fixed set of impedances by using relays. Fig. 3.11 shows the setup used for the selection of the reference impedance.

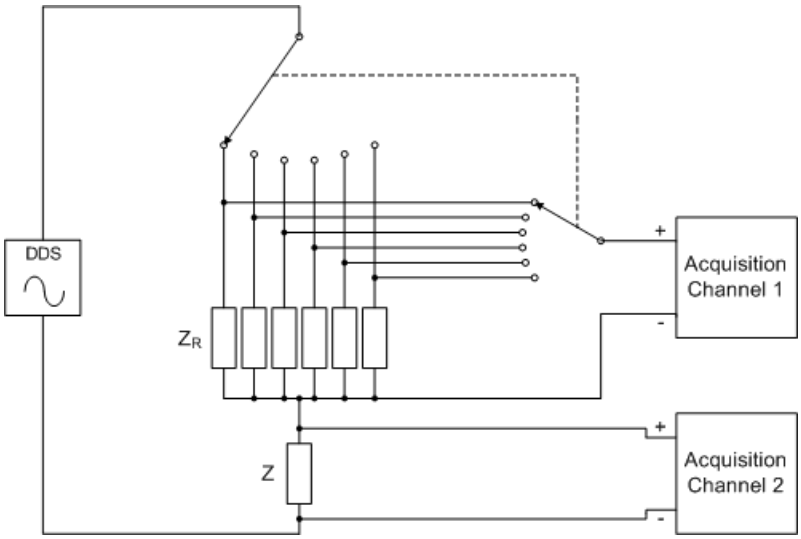


Fig. 3.11 - Setup for selecting the reference impedance.

In the setup from Fig. 3.11 each position of the switches corresponds to a relay. So, six relays are used to select six reference impedances.

The selection of the reference impedance is controlled by an algorithm running in the dsPIC.

The relays used in this work are the A5W-K from TAKAMISAWA. These relays have two poles, a nominal voltage of 5 V and their configuration is presented in Fig. 3.12.

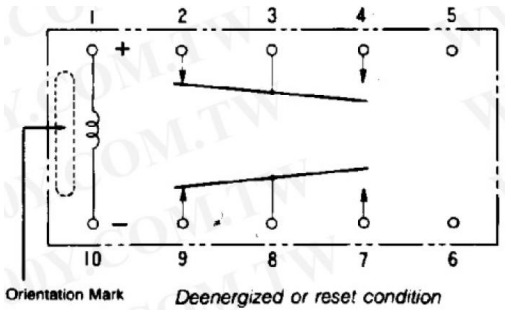


Fig. 3.12 - Configuration of the A5W-K relay.

In order to switch the position of the relay it is necessary a current of about 60 mA in its coil, however the maximum current available at any pin of the dsPIC is 4 mA. So, it was necessary to implement the circuit from Fig. 3.13 in order to control the position of the relay from the dsPIC.

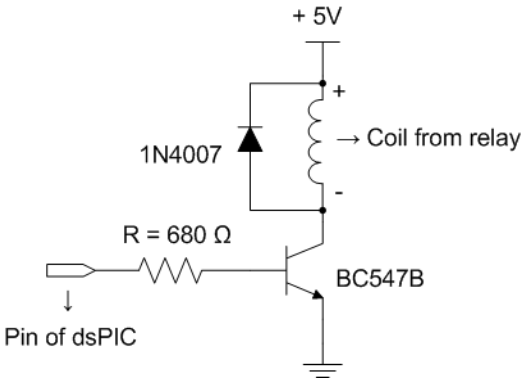


Fig. 3.13 - Circuit used to control the relays from the dsPIC.

The hardware necessary for the selection of the reference impedance is represented in square 4 in the schematic from Fig. C.1 in appendix C.

3.7 – Communication Protocols

The communication protocols used in this thesis are basically three: RS-232 (Recommended Standard 232) and SPI (Serial Peripheral Interface) and USB (Universal Synchronous Bus).

RS-232 is a standard for serial communication between devices. It supports full-duplex operating mode and it uses two data lines: Transmit (TxD) and Receive (RxD). The binary data uses ASCII (American Standard Code for Information Interchange) code with a word length of 7 or 8 data bits. It is capable of transmission rates up to 20000 b/s for long distances, and 921600 b/s for short distances. The transmission cable has a maximum length of approximately 15 m. Fig. 3.14 presents an example of typical RS-232 cable.

Fig. 3.14 - Example of a typical male-female RS-232 cable.

The RS-232 protocol is used to establish the communication between the dsPIC and the PC, making it possible to send commands from the PC to the dsPIC and to send the measurement results from the dsPIC to the PC. The transmission rate used in this work is 115200 b/s.

The hardware for obtaining the RS-232 interface is represented in square 2 in the schematic from Fig. D.1 in appendix D.

The SPI protocol is used to communicate between the dsPIC and the ADCs, both to control them and to transfer the acquired data from them. It will also be used to program the DDS to generate the signal with the proper frequency to perform the measurement.

SPI is a synchronous serial interface useful for communicating with other peripheral or microcontroller devices. It operates in master/slave mode where the master device initiates the transmission. In this case the dsPIC will operate as the master, and the ADCs and the DDS as the slaves. The serial interface consists of 4 pins: SDI (serial data input), SDO (serial data output), SCK (serial clock), and SS (active low slave select). The slave uses the master's clock, removing the slave's need for a precision oscillator. The SPI interface can operate using 2, 3 or 4 pins. In the 3-pin mode, SS is not used. In the 2-pin mode, neither SDO nor SS is used.

4 – Algorithms

This section describes the main algorithms used to effectively perform the measurement of the unknown impedance.

The program follows the measurement procedure presented in the flowchart shown in Fig. 4.1

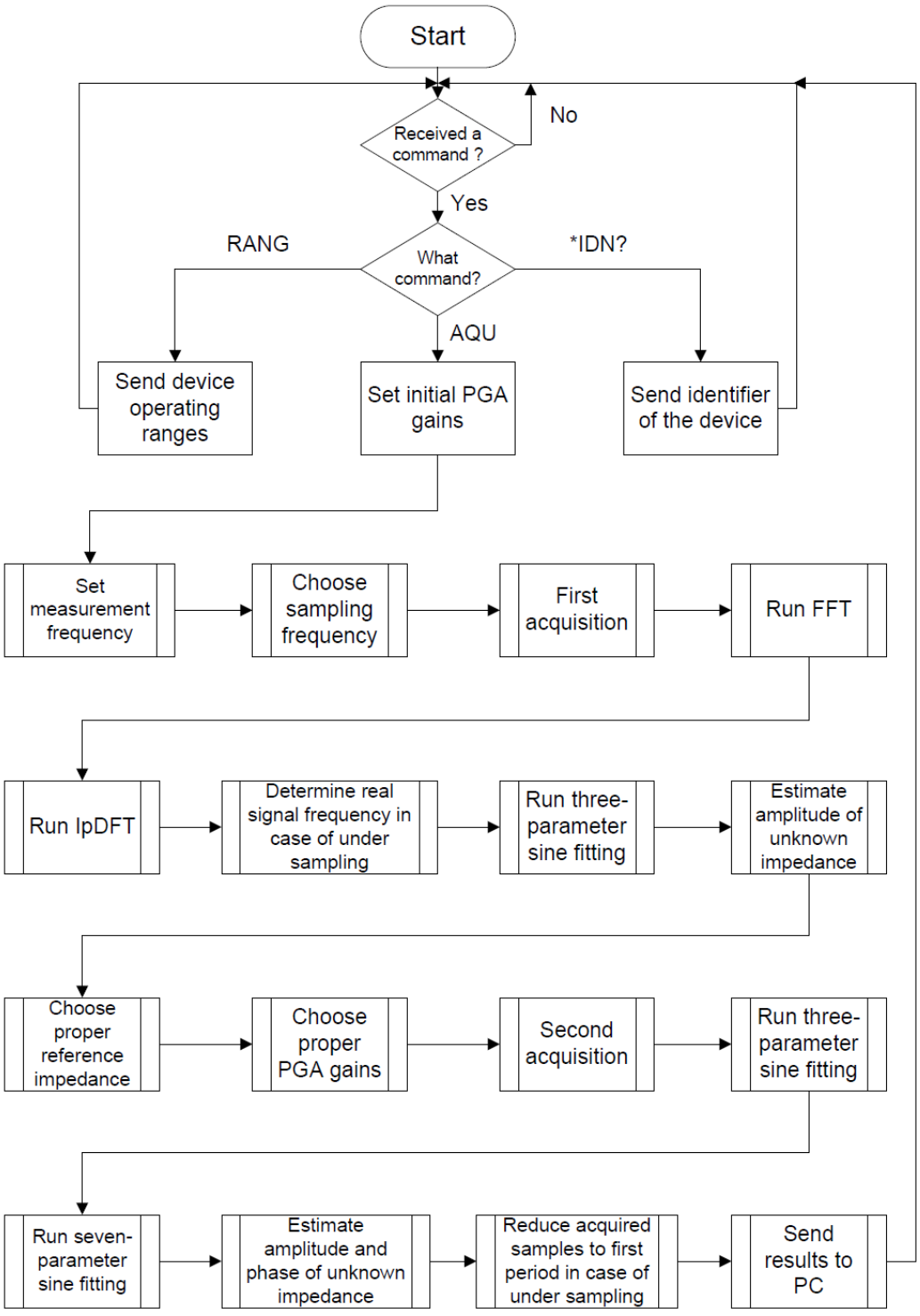


Fig. 4.1 - Flowchart representing the measurement procedure of the program.

The execution time of the acquisition will depend on the sampling frequency used. In the case of the maximum sampling frequency (178 kS/s), it is required 11.4688 ms to acquire all of the 2048 samples (1024 samples per channel).

4.1 – Algorithm for Selecting the Sampling Frequency

Given the measurement frequency provided by the user, it is necessary to select the appropriate sampling frequency to acquire the signals across the reference impedance and the impedance under measurement.

The range of frequencies used in this work goes from 500 Hz to 200 kHz. Since the maximum sampling frequency (F_{Smax}) is 178 kS/s, two cases must be considered for the acquisition of the signals: the case with no under sampling, for signals with frequencies under $F_{Smax}/2$, and the case with under sampling for signals with frequencies above $F_{Smax}/2$.

According to the Nyquist Sampling Theorem a signal must be acquired at a rate of at least twice the signal's maximum frequency (no under sampling), otherwise aliasing will occur and the acquired signal will present an apparent frequency (under sampling). In other words, the signal's frequency must not exceed half of the sampling frequency ($F_S/2$). Fig. 4.2 shows what happens to the acquired signal's spectrum in both cases.

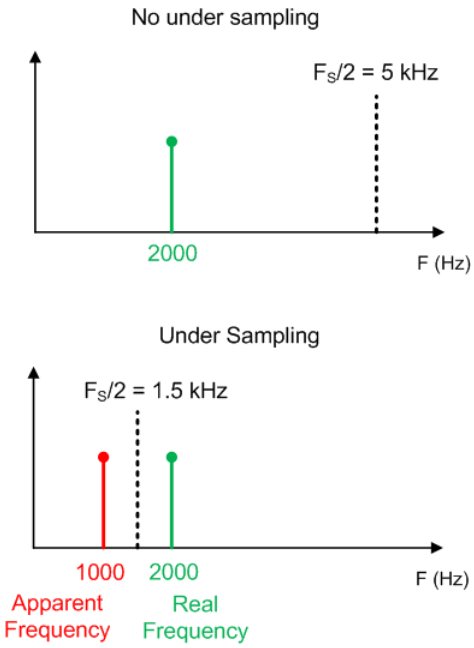


Fig. 4.2 - Behavior of an acquired signal spectrum in the case of no under sampling and in the case of under sampling.

In the case of under sampling, the real frequency of the signal can be retrieved from the apparent frequency. If the real frequency is known with an uncertainty of $F_S/4$, which is the case of this work since the measurement frequency is selected by the user, we can determine the number of folds of the spectrum by

$$m = \left\lfloor \frac{F_{est}}{F_S/2} \right\rfloor \tag{4.1}$$

where F_{est} is the estimative of the real frequency.

If m is odd, then

$$F_{real} = (m+1)\frac{F_S}{2} - F_{apparent} \tag{4.2}$$

If m is pair, then

$$F_{real} = m\frac{F_S}{2} + F_{apparent} \tag{4.3}$$

where F_{real} is the real frequency and $F_{apparent}$ is the apparent frequency of the acquired signal.

The flowchart from Fig. 4.3 describes the procedure to choose the sampling frequency. F is the measurement frequency, F_S is the selected sampling frequency, NP is the number of periods to acquire, N is the number of samples of each acquired signal, m_{real} is the number of full folds of the signal spectrum and m_{ideal} is the desired number of folds of the signal spectrum. The m_{ideal} value is chosen in a way that, in the case of under sampling, the acquired signal's frequency (apparent) will be at the middle of the spectrum, which corresponds to the best case.

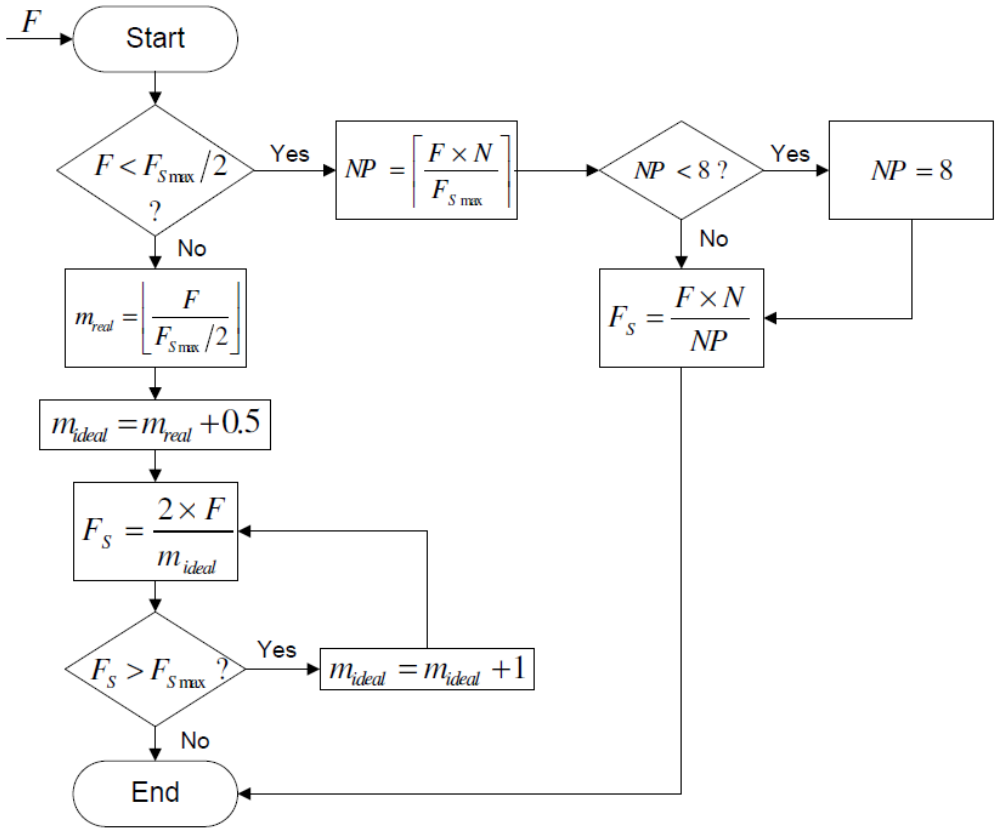


Fig. 4.3 - Flowchart of the algorithm used to choose the sampling frequency.

4.2 – FFT and IpDFT Algorithms

In order to use the sine fitting algorithms it is necessary to know the frequency of the acquired signal. To obtain the best estimative of the parameters from the sine fitting algorithms and to reduce the number of iterations of the seven parameter sine fitting algorithm, it is essential to have the best estimative possible of the signal's frequency. So, it was decided to use the FFT (Fast Fourier Transform) and IpDFT (Interpolated Discrete Fourier Transform) algorithm [17].

The FFT algorithm is used to translate the time domain samples of the signal into the frequency domain, thus obtaining the spectrum of the signal. Each sample in the frequency domain is a complex number composed by a real and imaginary part.

Given the spectrum of the signal, it is possible to use the IpDFT algorithm to accurately determine the frequency of the signal. The algorithm searches for the maximum element in the spectrum, X(L), and its largest neighbor, X(L+1), where L is the index of the maximum element in the array containing the frequency domain samples and L+1 the index of its largest neighbor. 0 < L < N, where N is the number of samples in the array. Given that,

$$X(L) = U_L + jV_L \tag{4.4}$$

$$X(L+1) = U_{L+1} + jV_{L+1} \tag{4.5}$$

then

$$\lambda = \frac{arccos\left(\frac{Z_2\cos\big(n(L+1)\big) - Z_1\cos(nL)}{Z_2 - Z_1}\right)}{n} \tag{4.6}$$

with

$$Z_1 = V_L\left(\frac{K_{opt} - \cos(nL)}{\sin(nL)}\right) + U_L \tag{4.7}$$

$$Z_2 = V_{L+1}\left(\frac{K_{opt} - \cos\big(n(L+1)\big)}{\sin\big(n(L+1)\big)}\right) + U_{L+1} \tag{4.8}$$

$$K_{opt} = \frac{\big((\sin(nL))(V_{L+1} - V_L) + (\cos(nL))(U_L + U_{L+1})\big)}{U_{L+1} - U_L} \tag{4.9}$$

where

$$n = \frac{2\pi}{N} \tag{4.10}$$

The frequency of the signal is given by

$$f = \lambda \Delta f \tag{4.11}$$

where Δf is the frequency resolution of the spectrum.

The FFT algorithm used consumes a great amount of memory (16434 bytes), which is necessary to hold the frequency domain samples, and takes a considerable amount of time to execute, 10.3662 ms. For this reason the execution time of the program is greatly affected, slowing down the measurement procedure.

The IpDFT algorithm only requires 64 bytes of memory and executes in 566.4 µs.

4.3 – Three-Parameter Sine-Fitting Algorithm

As described in Section 2.2, the three-parameter sine-fitting algorithm is a non-iterative algorithm that, with the knowledge of the frequency, estimates the amplitude, the phase and the DC component of the acquired sine signal.

The acquired sine signals can be represented by

$$u(t) = D\cos(2\pi f t + \phi) + C \tag{4.12}$$

or by

$$u(t) = A\cos(2\pi f t) + B\sin(2\pi f t) + C \tag{4.13}$$

where D is the amplitude, ϕ is the phase and C is the DC component of the signal.

The parameters in (4.11) are related to those in (4.12) by

$$D = \sqrt{A^2 + B^2} \tag{4.14}$$

and

$$\phi = atan2(B, A) \tag{4.15}$$

The $atan2$ function is a variation of the $arctan$ function. It takes into account the signs of both vector components (B and A), and places the angle in the correct quadrant considering all four quadrants. The function produces results in the range $[-\pi, \pi]$, which can be mapped to $[0, 2\pi]$ by adding 2π to negative results. The $atan2$ function does not have the problem that the $arctan$ function has for angles of $\frac{\pi}{2}$ and $-\frac{\pi}{2}$, because it would be necessary an infinite value for the $arctan$ function to produce such angles.

The estimated parameter vector is

$$\hat{x} = [\ A \quad B \quad C\ ]^T \tag{4.16}$$

which is given by

$$\hat{x} = M^{\dagger} y \tag{4.17}$$

where y is the sample vector

$$y = [u_1 \quad u_2 \quad \dots \quad u_N]^T \tag{4.18}$$

and

$$M = \begin{bmatrix} \cos(2\pi f t_1) & \sin(2\pi f t_1) & 1 \\ \vdots & \vdots & \vdots \\ \cos(2\pi f t_N) & \sin(2\pi f t_N) & 1 \end{bmatrix} \tag{4.19}$$

N is the number of samples acquired, and $M^{\dagger}$ is the pseudo inverse matrix of M.

In this work, the number of samples acquired for each channel was 1024. The algorithm takes 5.0652 ms to run and it doesn't require much data memory to be executed (160 bytes).

The algorithm was used to make an initial estimative of the impedance under measurement, with an initial acquisition, in order to choose the reference impedance that best matches the magnitude of the unknown impedance. This algorithm is also used to obtain the initial estimative for the parameters of the seven-parameter sine fitting algorithm, by using the samples from a second and final acquisition.

The flowchart presented in Fig. 4.4 describes the execution of the algorithm.

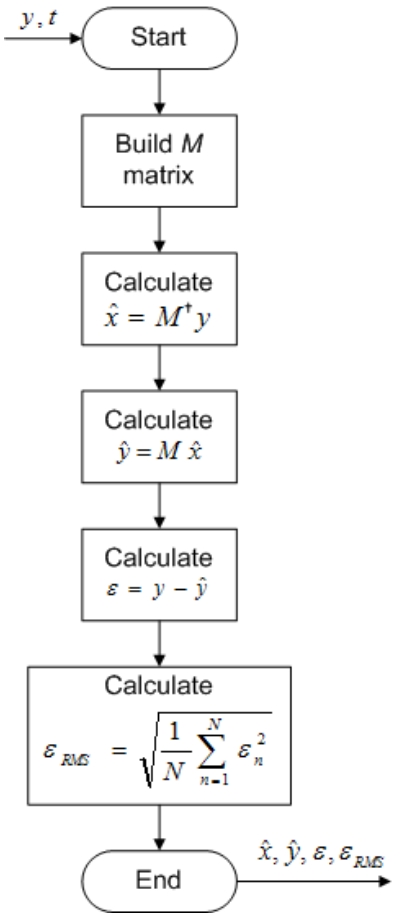


Fig. 4.4 - Flowchart of the three-parameter sine-fiting algorithm.

In the flowchart of Fig. 4.4 $\hat{y}$ represents the estimative of the samples, calculated based on the estimated parameters $\hat{x}$, ε are the errors between the acquired samples and the estimated samples and ε_{RMS} is the total root mean square error.

4.4 – Selection of the Reference Impedance and the PGIAs Gains

One function of this algorithm is to select the appropriate reference impedance to be used in the measurement. The algorithm will select the reference impedance that best matches the magnitude of the impedance under measurement. This is done in order to ensure that the amplitude of the acquired signals is as close as possible to the ADC input range. This way it is possible to maximize the use of the ADC dynamic range.

In order to choose the proper reference impedance, it is necessary to perform an initial acquisition and estimate the amplitude of the unknown impedance by resorting to the three-parameter sine fitting algorithm. This algorithm doesn't produce the best estimative for the parameters of the acquired signals, however since it is only intended to choose the reference impedance, the accuracy of these parameters will be sufficient.

Another function of this algorithm is to digitally control the gains of the PGIAs to have the maximum voltage possible at each ADC input channel in order to obtain a better resolution

of the acquired signals across the reference impedance and the impedance under measurement.

Given the magnitude of the signals, obtained with the three-parameter sine fitting algorithm, the algorithm determines whether it is possible or not to increase the gains of each amplifier and if so how much. The gains are set according to the values admitted by the amplifiers and in a way that does not exceed the input range of the ADCs.

Once the correct reference impedance has been chosen and the gains of the amplifiers are set, the actual measurement of the unknown impedance can take place.

The algorithm is divided into three steps: the selection of the reference impedance, the estimation of the signals across the impedance under measurement and the reference impedance after the correct reference impedance has been chosen, and the selection of the gains of the PGIAs.

The flowchart from Fig. 4.5 describes the steps taken to choose the reference impedance.

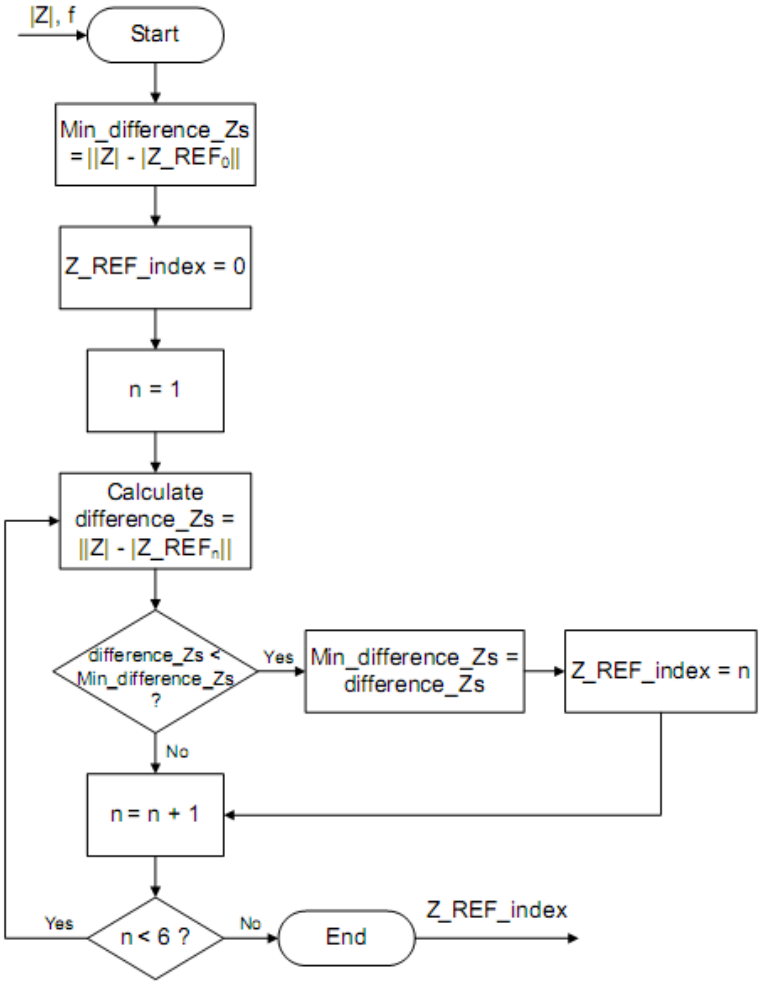


Fig. 4.5 - Flowchart for the selection of the reference impedance.

In Fig. 4.5, Z_REF_index refers to the index of the array that contains the values of the reference impedances.

The reference impedances used in this work were resistors of nominal value equal to: 120 Ω, 220 Ω, 510 Ω, 1200 Ω, 2200 Ω and 5100 Ω. These resistors have a tolerance of ±5%.

To obtain the best measurement results possible, it was necessary to characterize the reference impedances in order to determine their real values with precision, instead of just using their nominal values. To do so, it was used a 3522-50 LCR HiTESTER from HIOKI, to measure the reference impedances in the frequency range from 100 Hz to 100 kHz with 100 Hz intervals. Although the range of frequencies intended for this work goes from 500 Hz to 200 kHz, it was only measured values until 100 kHz because that is the HIOKI's device maximum measurement frequency. The measured values of the modulus of the reference impedances are presented in the graphics from Fig. A.1 to Fig. A.6 in appendix A.

After obtaining these values it was necessary to estimate the resistance value (R) that best adapts to the measured values. For this purpose it was used the fminsearch function in the MATLAB environment. The fminsearch function minimizes the errors between the measured values and the estimated R value, thus obtaining the optimum value for R. The estimated R values for each reference impedance are presented in Table 4.1.

Table 4.1 - Estimated R values.

Reference Impedance (Ω)	Estimated R Values (Ω)	Error
120	118.255608	1.564026×10^{-6}
220	216.015079	7.630637×10^{-5}
510	502.384931	2.214254×10^{-6}
1200	1182.838493	4.618403×10^{-5}
2200	2161.503298	3.286041×10^{-6}
5100	5040.373694	3.221815×10^{-5}

The value used for the phase of the reference impedances is zero.

After the reference impedance has been chosen, it is necessary to consider the circuit from Fig. 4.6 in order to estimate the magnitudes of the signals across the reference impedance and the impedance under measurement.

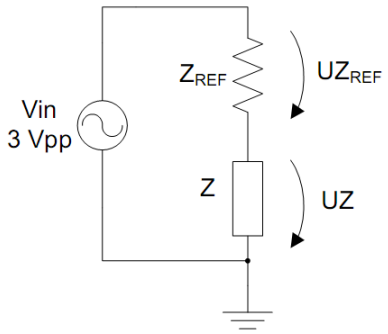


Fig. 4.6 – Simplification of the measurement circuit.

With the knowledge of the magnitude of the signal applied to the measurement circuit (|Vin|), the magnitude of UZ_{REF} is determined by

$$|UZ_{REF}| \cong \frac{|Z_{REF}|}{|Z_{REF}| + |Z|} \times |Vin| \tag{4.20}$$

and the magnitude of UZ is determined by

$$|UZ| \cong \frac{|Z|}{|Z_{REF}| + |Z|} \times |Vin| \tag{4.21}$$

Once the magnitudes of the signals across the impedances have been estimated, it is possible to determine the appropriate gains of the amplifiers. Note that (4.20) and (4.21) are an approximation since it does not account for the wiring that connects the impedances. It is however suitable for the PGIAs gain selection.

The flowchart from Fig. 4.7 describes the process to choose the gains of the amplifiers.

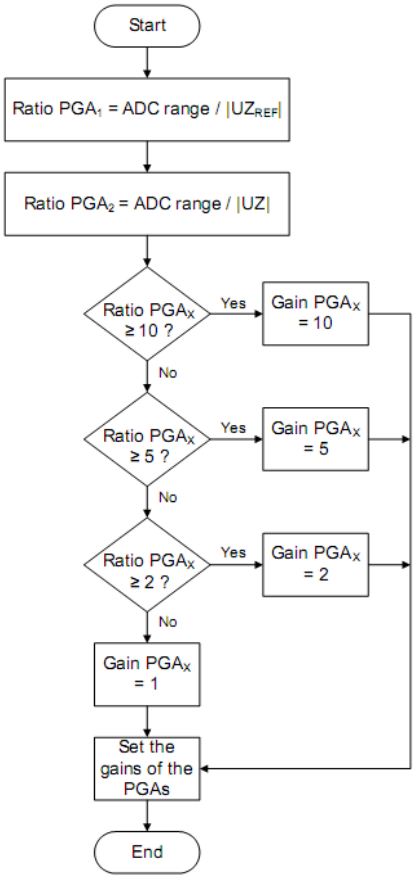


Fig. 4.7 - Flowchart describing how the gains of the amplifiers are chosen.

In the flowchart from Fig. 4.7, ADC range is a variable containing the range of the ADCs used in the work, the index X is 1 in the case of PGA_1, which is connected to the reference impedance, and 2 in the case of PGA_2, which is connected to the impedance under measurement.

4.5 – Seven-Parameter Sine-Fitting Algorithm

The seven-parameter sine-fitting algorithm was described in Section 2.2 as an iterative algorithm which uses the acquired data from two channels with the same frequency to estimate the amplitude, phase and DC component of both signals, plus their common frequency.

Just like in the case of the three-parameter algorithm, the sine signals of each channel can be represented by (4.11) or by (4.12) considering the relations in (4.13) and (4.14).

Since the initial estimates of the parameters are crucial for the algorithm to converge, these values were obtained through the three-parameter sine-fitting.

For each iteration i of the seven-parameter algorithm, the estimated parameter vector is

$$x^{(i)} = \begin{bmatrix} A_1^{(i)} & B_1^{(i)} & C_1^{(i)} & \Delta f^{(\mathrm{i})} & A_2^{(i)} & B_2^{(i)} & C_2^{(i)} \end{bmatrix}^T \tag{4.22}$$

A_k, B_k and C_k are the estimated parameters for the signals in channel k, with k = 1, 2.

$\Delta f^{(\mathrm{i})}$ is the frequency correction that updates the estimated common frequency. The iterative process ends when the relative frequency adjustment $\Delta f^{(\mathrm{i})}/\mathrm{f}^{(\mathrm{i})}$ is below a preset threshold, which is set to 5×10^{-7} in the case of this work. This value results from a compromise between obtaining a frequency result with good accuracy and using the best precision in the dsPIC.

The estimated parameter vector is given by

$$x^{(i)} = \left[\left[M^{(i)} \right]^T \quad M^{(i)} \right]^{-1} \left[\left[M^{(i)} \right]^T \quad y \right] \tag{4.23}$$

where y is the concatenated vector with samples from both channels

$$y = \begin{bmatrix} u_{1,1} & u_{1,2} & \dots & u_{1,N} & u_{2,1} & u_{2,2} & \dots & u_{2,N} \end{bmatrix}^T \tag{4.24}$$

and

$$M^{(i)} = \left[\begin{array}{c|c|c} Q_1^{(i)} & r_1^{(i)} & 0 \\ \hline 0 & r_2^{(i)} & Q_2^{(i)} \end{array}\right] \tag{4.25}$$

with ($\omega = 2\pi f^{(i)}$)

$$Q_k^{(i)} = \begin{bmatrix} \cos(\omega t_{k,1}) & \sin(\omega t_{k,1}) & 1 \\ \cos(\omega t_{k,2}) & \sin(\omega t_{k,2}) & 1 \\ \vdots & \vdots & \vdots \\ \cos(\omega t_{k,N}) & \sin(\omega t_{k,N}) & 1 \end{bmatrix} \tag{4.26}$$

and

$$r_k^{(i)} = \begin{bmatrix} \alpha_{k,1} \\ \alpha_{k,2} \\ \vdots \\ \alpha_{k,N} \end{bmatrix} \text{ with } \alpha_{k,n} = -2\pi A_k^{(i)} t_{k,n} \sin(\omega t_{k,n}) + 2\pi B_k^{(i)} t_{k,n} \cos(\omega t_{k,n}) \tag{4.27}$$

Considering the optimizations done to the algorithm in [18] in terms of program memory needed, it is possible to write

$$\left[\left[M^{(i)} \right]^T \quad M^{(i)} \right] = \begin{bmatrix} E_{1,1} & E_{1,2} & E_{1,3} & E_{1,4} & 0 & 0 & 0 \\ E_{1,2} & E_{2,2} & E_{2,3} & E_{2,4} & 0 & 0 & 0 \\ E_{1,3} & E_{2,3} & N & E_{3,4} & 0 & 0 & 0 \\ E_{1,4} & E_{2,4} & E_{3,4} & E_{4,4} & E_{4,5} & E_{4,6} & E_{4,7} \\ 0 & 0 & 0 & E_{4,5} & E_{1,1} & E_{1,2} & E_{1,3} \\ 0 & 0 & 0 & E_{4,6} & E_{1,2} & E_{2,2} & E_{2,3} \\ 0 & 0 & 0 & E_{4,7} & E_{1,3} & E_{2,3} & N \end{bmatrix} \tag{4.28}$$

with

$$E_{1,1} = \sum_{n=1}^{N} \cos^2(\omega t_n), E_{1,2} = \sum_{n=1}^{N} \cos(\omega t_n) \sin(\omega t_n), E_{1,3} = \sum_{n=1}^{N} \cos(\omega t_n), E_{1,4} = \sum_{n=1}^{N} \cos(\omega t_n)\, \alpha_{1,n},$$

$$E_{2,2} = \sum_{n=1}^{N} \sin^2(\omega t_n), E_{2,3} = \sum_{n=1}^{N} \sin(\omega t_n), E_{2,4} = \sum_{n=1}^{N} \sin(\omega t_n)\, \alpha_{1,n}, E_{3,4} = \sum_{n=1}^{N} \alpha_{1,n}, E_{4,4} = \sum_{n=1}^{N} \left({\alpha_{1,n}}^2 + {\alpha_{2,n}}^2 \right),$$

$$E_{4,5} = \sum_{n=1}^{N} \cos(\omega t_n)\, \alpha_{2,n}, E_{4,6} = \sum_{n=1}^{N} \sin(\omega t_n)\, \alpha_{2,n}, E_{4,7} = \sum_{n=1}^{N} \alpha_{2,n}.$$

Finally $\left[\left[M^{(i)}\right]^T \quad y\right]$ is given by

$$\left[\left[M^{(i)}\right]^T \quad y\right] = \begin{bmatrix} \sum_{n=1}^{N} \cos(\omega t_n)\, u_{1,n} \\ \sum_{n=1}^{N} \sin(\omega t_n)\, u_{1,n} \\ \sum_{n=1}^{N} u_{1,n} \\ \sum_{n=1}^{N} u_{1,n}\alpha_{1,n} + u_{2,n}\alpha_{2,n} \\ \sum_{n=1}^{N} \cos(\omega t_n)\, u_{2,n} \\ \sum_{n=1}^{N} \sin(\omega t_n)\, u_{2,n} \\ \sum_{n=1}^{N} u_{2,n} \end{bmatrix} \tag{4.29}$$

The flowchart from Fig. 4.8 describes the operations executed by the seven-parameter sine-fitting algorithm.

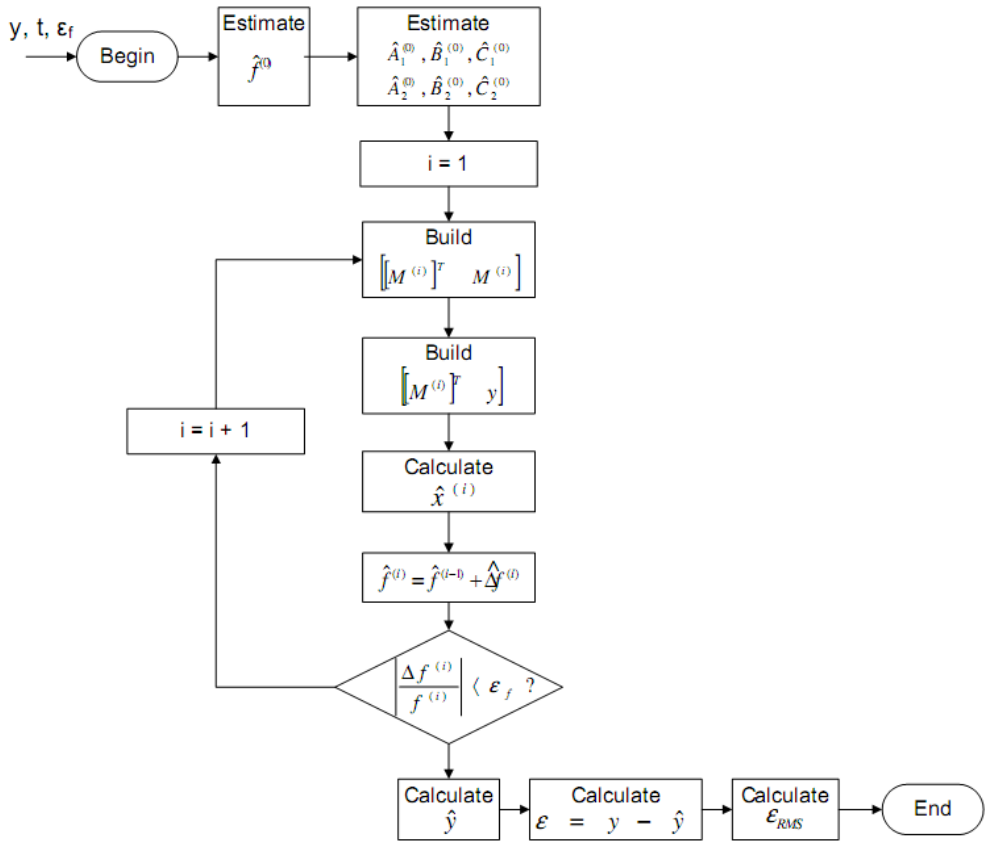


Fig. 4.8 - Seven-parameter sine-fitting algorithm flowchart.

In the flowchart from Fig. 4.8, ε_f is the preset threshold for the relative frequency adjustment, $\hat{y}$ represents the estimative of the samples, calculated based on the estimated parameters $\hat{x}^{(i)}$, ε are the errors between the acquired samples and the estimated samples and ε_{RMS} is the total root mean square error.

In the seven-parameter algorithm, with the increase of noise in the acquired sine signals, the total ε_{RMS} error will increase which results in greater deviation of the estimated parameters.

The implemented algorithm requires 582 bytes of memory and takes 8.7336 ms to run.

4.6 – Reduction of Acquired Samples to the First Period

In the case of under sampling it was decided to reduce the acquired samples to the first period of the signals. This was done to allow the user to view the acquired signals with the correct frequency and amplitude in order to facilitate the detection of any eventual error. Otherwise it would be extremely difficult to interpret the acquired samples.

To reduce the acquired samples to the first period one has to build an array with the new time stamps correspondent to the first period. To do so we must consider

$$t_{S\_new} = t_S - \left\lfloor \frac{t_S}{T_{Signal}} \right\rfloor T_{Signal} \tag{4.30}$$

where t_S are the original time stamps, T_{Signal} is the period of the signal and t_{S_new} is the array with the new time stamps reduced to the first period.

After obtaining the array with the new time stamps it is simply a manner of ordering the time array, along with the samples correspondent to each time stamp.

The flowchart from Fig. 4.9 describes the execution of the algorithm.

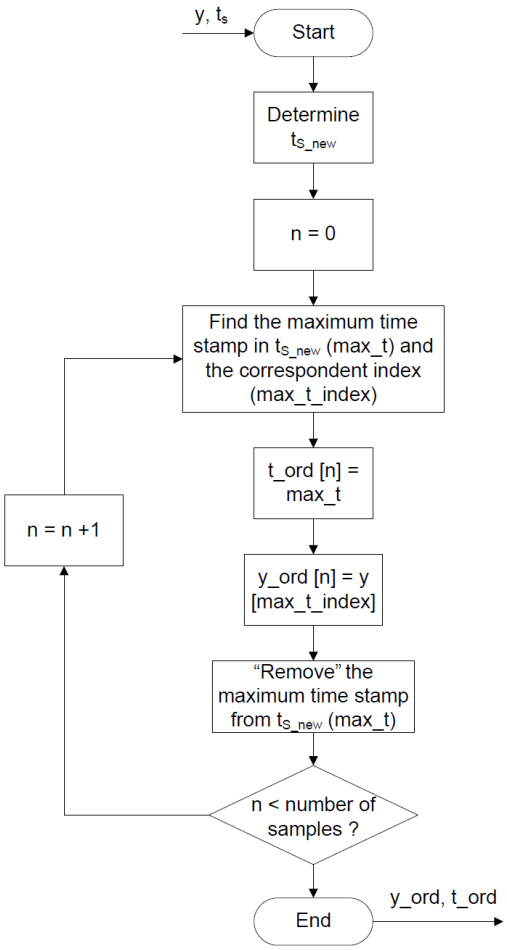


Fig. 4.9 - Flowchart of the algorithm to reduce the samples to the first period.

5 – Experimental Results

In this Section the experimental results obtained with the implemented system are presented.

Ten impedances with different amplitudes in the range from 100 Ω to 10 kΩ, and phases in the range from -90º to 90º were tested at 1 kHz and the results are compared to those obtained with the 3522-50 LCR HiTESTER from HIOKI.

The measurement results are presented in Table 5.1.

Table 5.1 - Measurement results obtained with the HIOKI and the implemented system.

	HIOKI		**Implemented System**	
Impedance	**Modulus (Ω)**	**Phase (º)**	**Modulus (Ω)**	**Phase (º)**
C = 32 nF	4979,4	-89,74	4989,23	-89,72
L = 15 mH	105,9	69,18	105,39	68,95
Serial RC R = 2 kΩ ; C = 47 nF	3899,1	-58,7	3932,89	-59,02
Serial RC R = 5,6 kΩ ; C = 47 nF	6527,7	-31,08	6546,40	-31,13
Serial RL R = 120 Ω; L = 15 mH	184,61	32,07	183,95	31,83
Serial RL R = 36 Ω; L = 15 mH	122,85	53,45	122,3	53,30
R = 1,8 kΩ	1782,6	0	1790,8	0,0012
R = 7,5 kΩ	7408,4	0	7454,59	-0,041
R = 9,1 kΩ	8957,1	0	9005,64	-0,056
R = 10 kΩ	9866,8	0	9923,33	-0,027

As shown in Table 5.1, the experimental results obtained with the implemented system are quite close to those obtained with the HIOKI, demonstrating that it is possible to implement a low-cost impedance measurement device with comparable precision of that of sophisticated measurement equipment.

The signals acquired by the implemented system correspondent to the impedances measured are presented in Fig. 5.1 to Fig. 5.10.

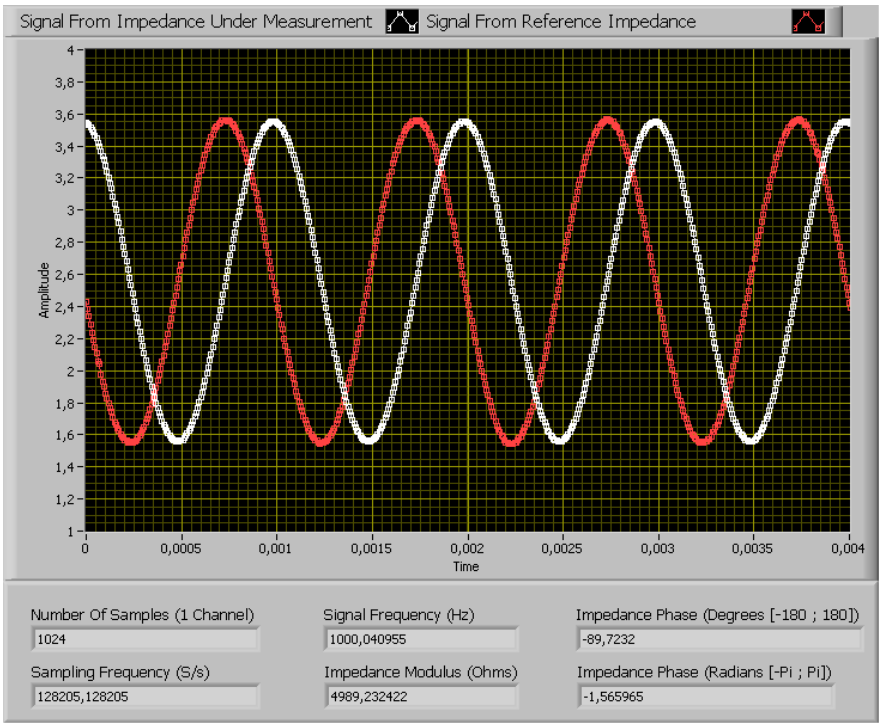


Fig. 5.1- Acquired signals and measurement results for C = 32 nF.

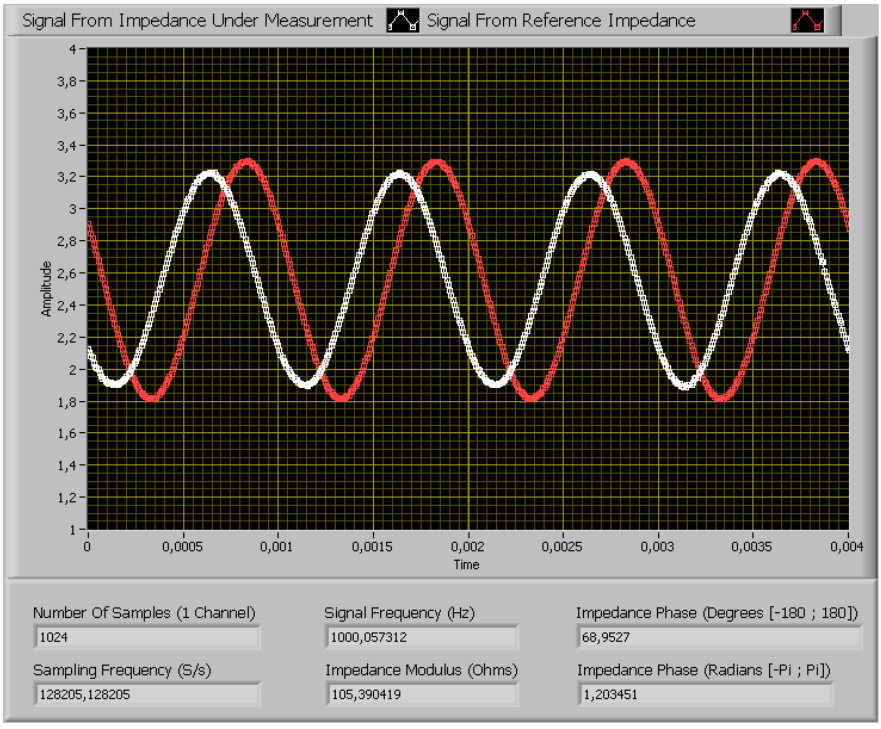


Fig. 5.2 - Acquired signals and measurement results for L = 15 mH.

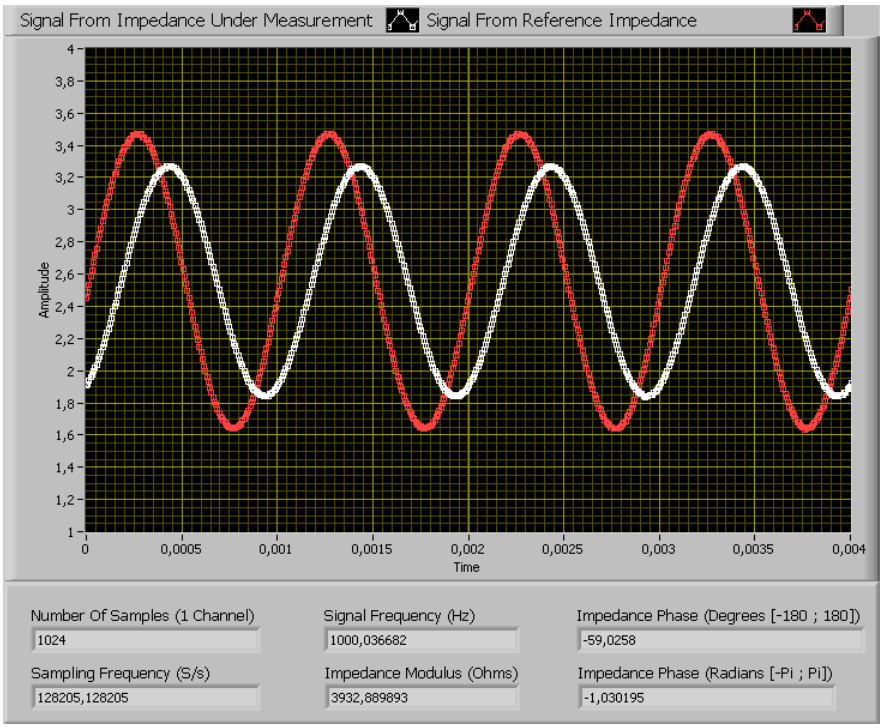


Fig. 5.3 - Acquired signals and measurement results for R = 2 kΩ, C = 47 nF.

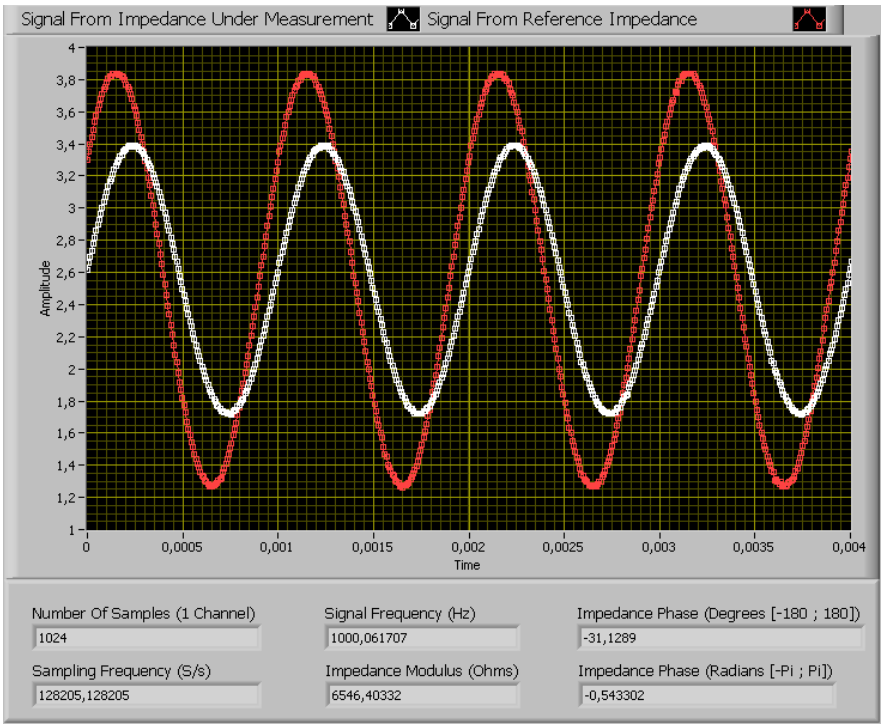


Fig. 5.4 - Acquired signals and measurement results for R = 5.6 kΩ, C = 47 nF.

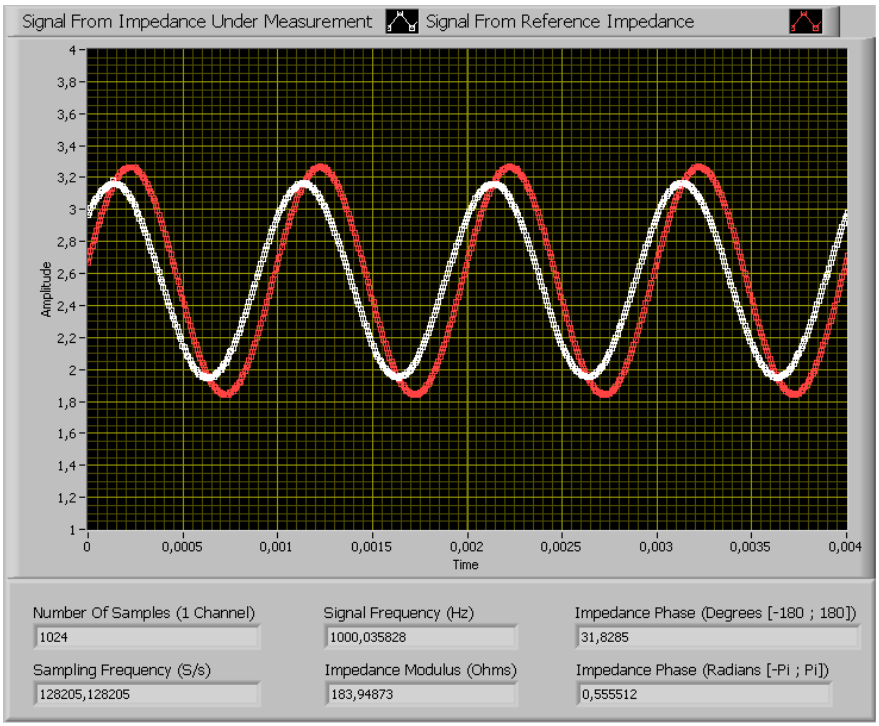


Fig. 5.5 - Acquired signals and measurement results for R = 120, L = 15 mH.

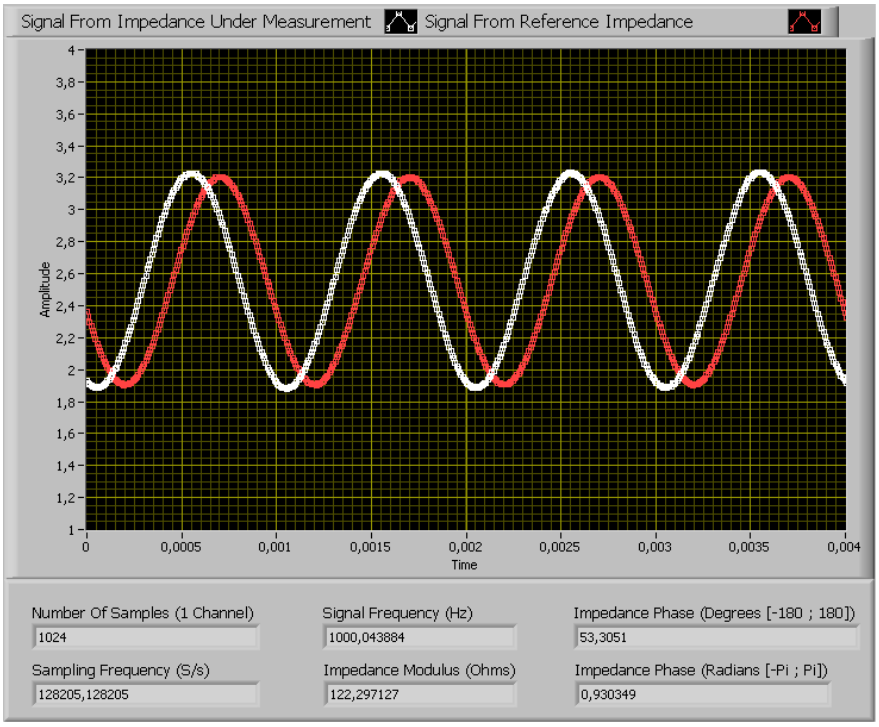


Fig. 5.6 - Acquired signals and measurement results for R = 36 Ω, L = 15 mH.

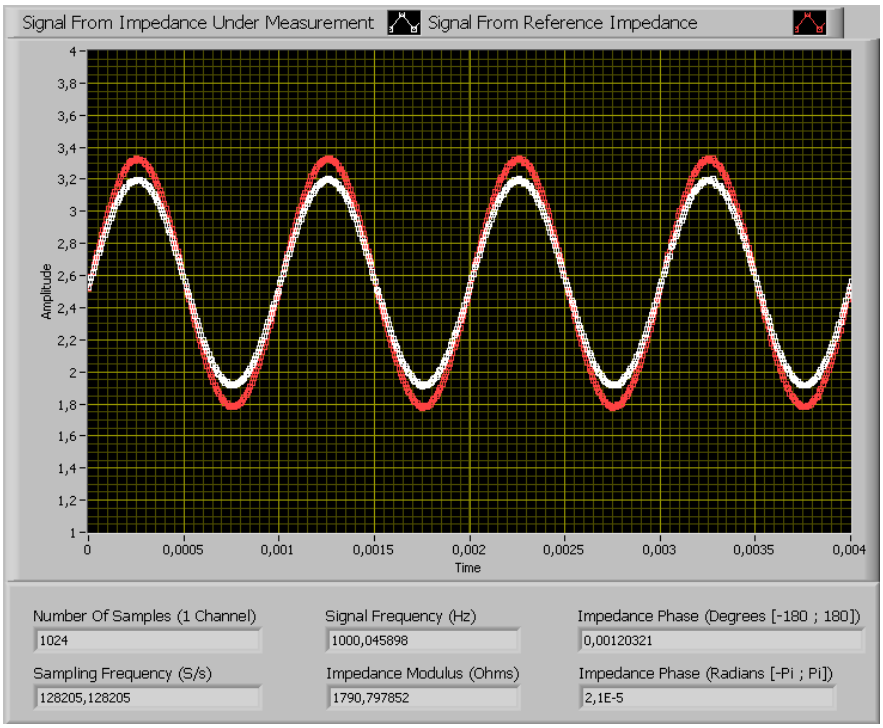


Fig. 5.7 - Acquired signals and measurement results for R = 1.8 kΩ.

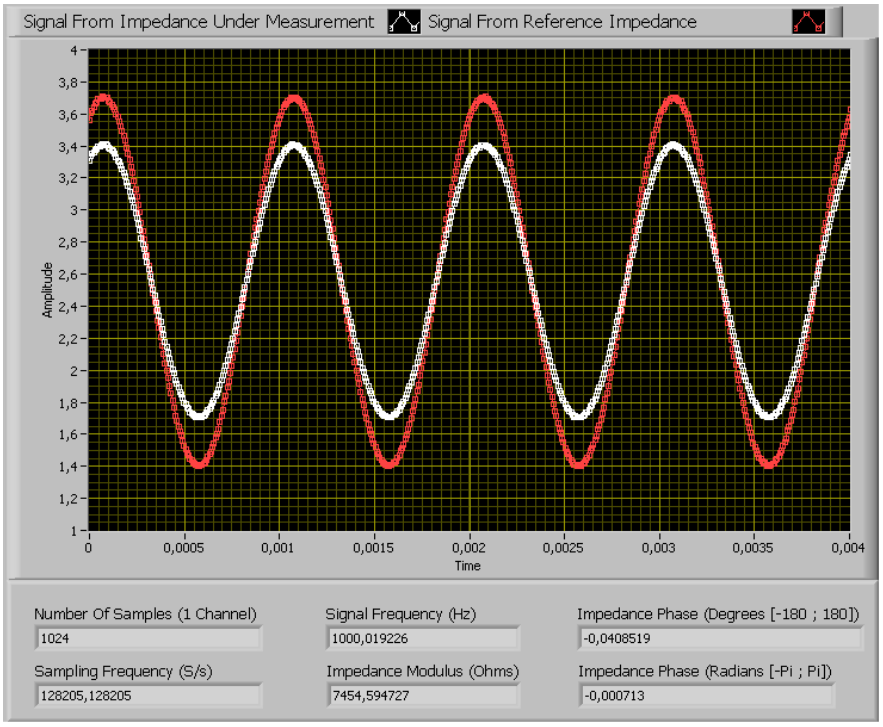


Fig. 5.8 - Acquired signals and measurement results for R = 7.5 kΩ.

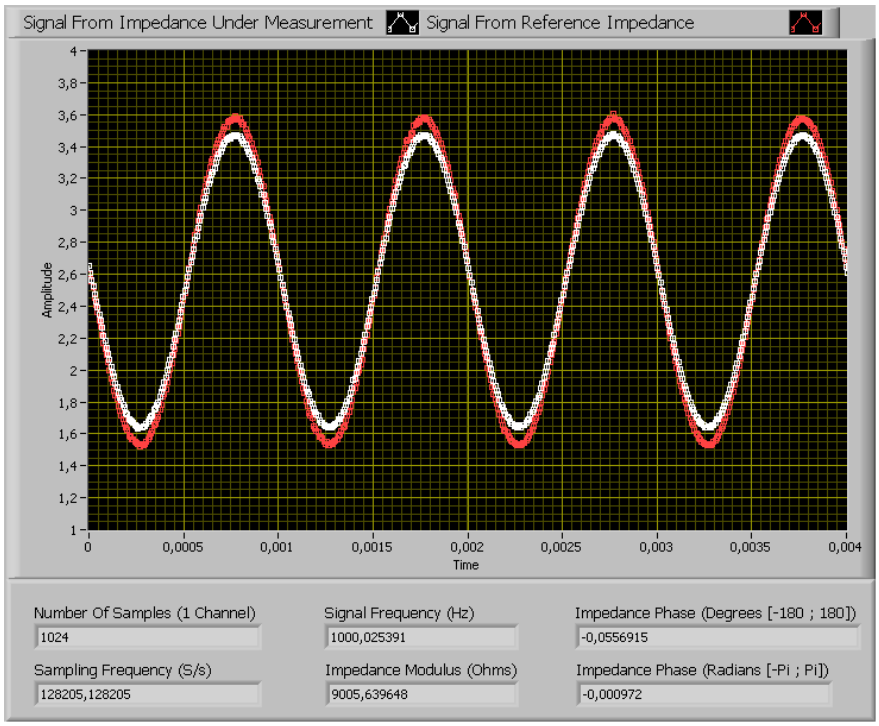


Fig. 5.9 - Acquired signals and measurement results for R = 9.1 kΩ.

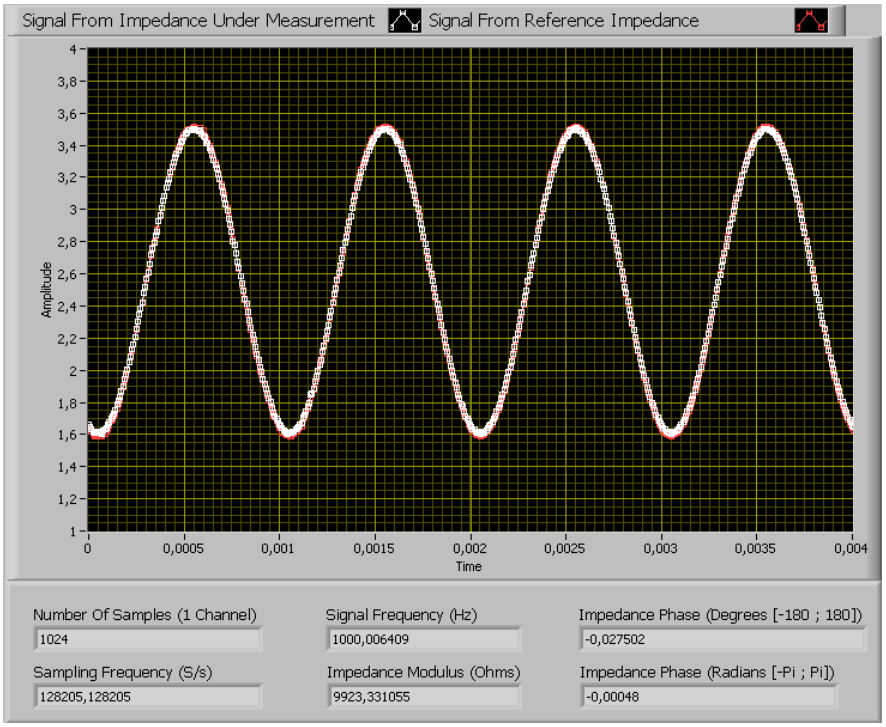


Fig. 5.10 - Acquired signals and measurement results for R = 10 kΩ.

Five hundred measurements of the same impedance were taken at a measurement frequency of 1 kHz. The impedance tested was a resistor with a nominal value of 7.5 kΩ. With these measurements it was possible to obtain the histograms of the measurement frequency the amplitude and the phase of the impedance. It was also obtained the dispersion of the complex value of the impedance in the Argand plane. These results are presented in Fig. 5.11 to Fig. 5.14.

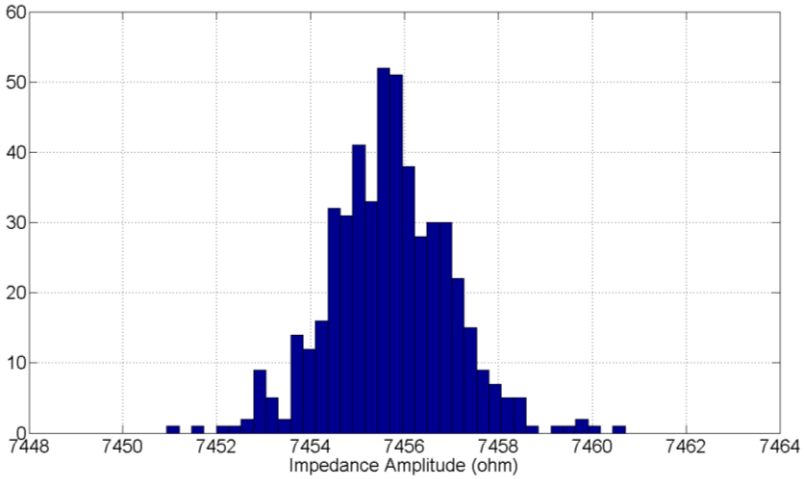


Fig. 5.11 - Histogram of impedance amplitude.

The measurements regarding the impedance amplitude present a mean of 7455,68 Ω, a standard deviation of 1,29 Ω and a relative standard deviation of 0,017 %.

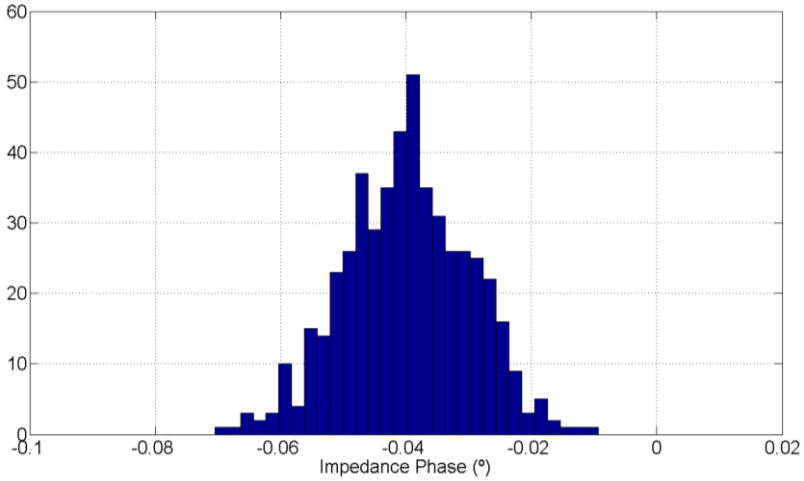


Fig. 5.12 - Histogram of impedance phase.

The measurements regarding the impedance phase present a mean of -0,0399 º and a standard deviation of 0,0999 º.

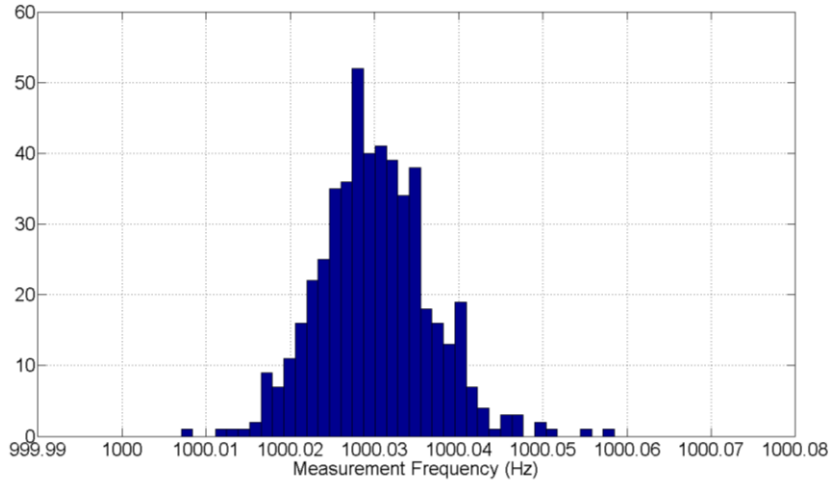


Fig. 5.13 - Histogram of measurement frequency.

The measurements regarding the measurement frequency present a mean of 1000,03005 Hz, a standard deviation of 0,00664 Hz and a relative standard deviation of 0,00069 %.

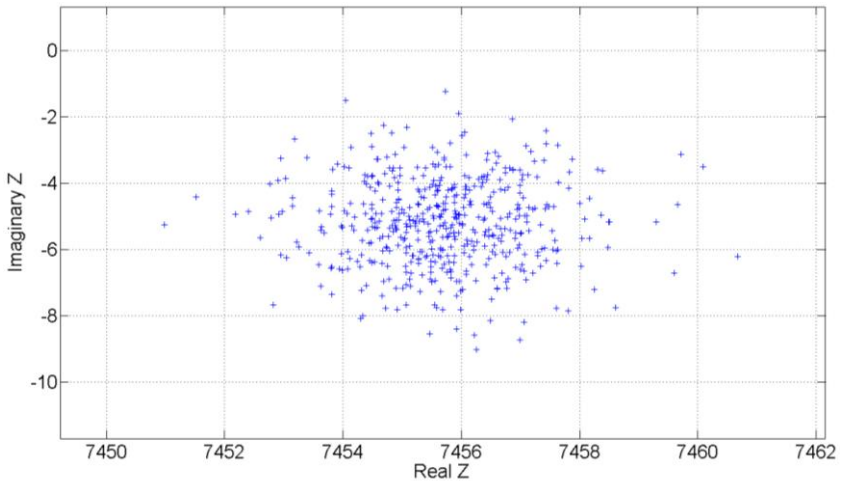


Fig. 5.14 - Dispersion of the complex value of the impedance in the Argand plane.

6 – System's Control Software

It was implemented a control software which runs in a PC. The software acts as an interface between the user and the device, giving its user the means to interact with the device. This software allows the user to choose the frequency at which to perform the measurement of the unknown impedance. Another feature of the software is the ability to communicate with the device in order to obtain information regarding the device's maximum operating ranges. An important aspect of the software is the fact that it allows the user to view and interpret the acquired samples of the measurements done.

The software was developed in the LabVIEW environment, which provides simple and intuitive programming tools to implement even the most complex program, and also has the capability to easily create an application that can run in a PC, even if it does not have the LabVIEW environment installed.

The main window of the implemented software is represented in Fig. 6.1.

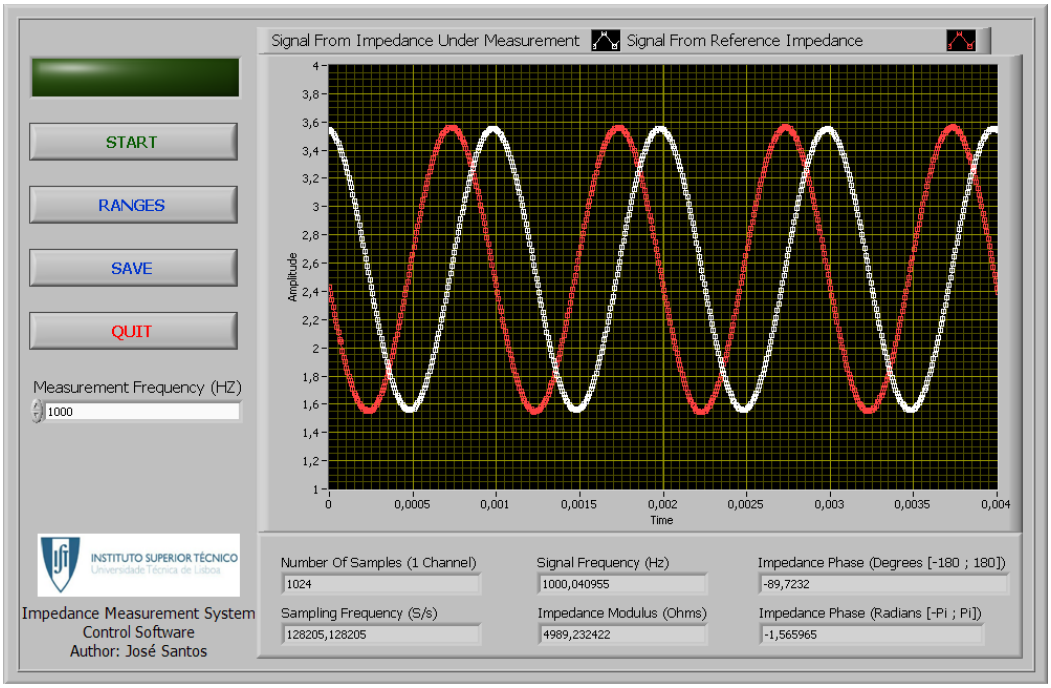


Fig. 6.1 - Control software main window.

Once the program is running and the desired measurement frequency has been introduced, the user can begin the measurement of the unknown impedance by clicking the start button.

The window of the control software has a LED which indicates the status of the program. When the user starts the measurement procedure the LED lights up, indicating that the PC is communicating with the device. After the measurement is complete and all of the data has been successfully received by the PC, the LED goes off indicating that the program is standing by and ready for the next measurement.

After the measurement results have been received by the PC, the user can choose to save the results, along with the acquired samples of the signals across the impedances. To do so he has only to click the save button, which results in the appearance of the window presented in Fig. 6.2.

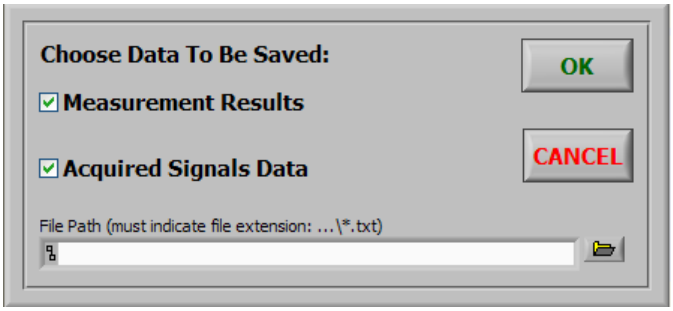


Fig. 6.2 - Save data program window.

In the window presented in Fig. 6.2 the user chooses the information to be saved and the location of the file containing the data (File Path). The file is saved as a text file with the extension txt.

The user can also use the control software to ask the device its maximum operating ranges by clicking the ranges button. Just like during the measurement procedure, while the PC is communicating with the device the LED remains lit until the data has not been received. Afterwards, when the ranges of the device have been received, the LED goes off and the program returns to the standby mode.

The ranges of the device appear in the window presented in Fig. 6.3.

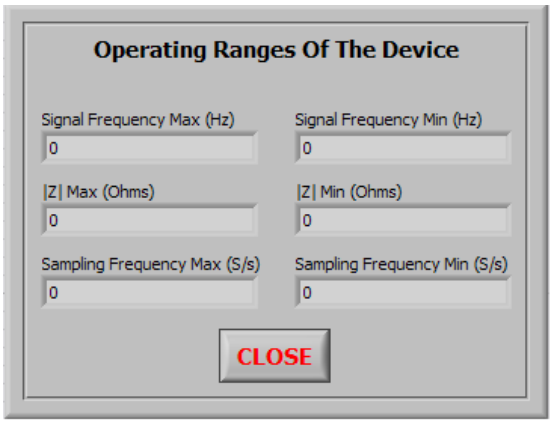


Fig. 6.3 - Program window that presents the ranges of the device.

The ranges of the device correspond to: measurement frequency, impedance modulus (|Z|) and sampling frequency.

7 – Conclusions

Impedance measurements have a great number of applications in different fields of science, such as medicine, civil engineering and applications where it is necessary to characterize sensor outputs, thus making it an important field of investigation.

Given the high cost of sophisticated equipment dedicated to impedance measurement, it is important to develop new and cheaper ways to make the same type of measurements at a more reduced cost.

The measurement setup proposed in this work is an attempt to do just that. It uses a number of low cost equipment, which by operating together can produce results with comparable accuracy to those obtained with dedicated measurement equipment. This is possible because of the great processing capability of the processing unit used, the dsPIC, combined with the powerful signal processing techniques applied, the three-parameter sine fitting algorithm and the seven-parameter sine fitting algorithm.

The implemented system facilitates the interaction with its user, since it is possible to connect it to a PC, through a RS-232 connection, to receive commands from the user and allow him to monitor the device and save the results of the measurements done.

The system implemented in this thesis operates in a wide frequency range, 500 Hz to 200 kHz, and is capable of measuring impedances in the amplitude range from 100 Ω to 10 kΩ.

To operate in this frequency range it was necessary to apply under sampling techniques because of the constraints imposed by the dsPIC in terms of maximum sampling frequency.

The experimental results produced in this work were compared with a commercially available device, the 3522-50 LCR HiTESTER from HIOKI, by measuring ten different impedances in the amplitude range from 100 Ω to 10 kΩ and in phase range from -90º to 90º at a measurement frequency of 1 kHz. Given the experimental results presented, it was demonstrated that it is possible to implement a low-cost device operating at a wide frequency range, but still with measurement precision comparable to that of sophisticated high-cost dedicated impedance measurement systems.

8 – Future Work

In order to improve the work presented in this thesis it is proposed a few research directions.

One of the proposed modifications to be done regards the upgrade of the device to act as a stand-alone system without the need of a PC to control it. This involves the implementation of a keypad to give the user the ability to control the device independently. This way the user would have the choice to control the device without using the PC.

Another important modification is the upgrade of the processing unit from a dsPIC to a DSP. This would improve the processing capabilities of the device in terms of throughput and available data memory, thus allowing the device to perform the estimation of the unknown impedance in less time. The increase in data memory would allow the acquisition of more samples, and so a better resolution of the acquired signals.

It would also be interesting to give the device the ability to measure the impedances at different frequencies in a single acquisition. To do so it is suggested the study of multi-harmonic signals to stimulate the measurement circuit.

9 – References

[1] *The Impedance Measurement Handbook – A Guide to Measurement Technologies and Techniques*, Agilent Technologies Co., 2000.

[2] Olumuyiwa T. Ogunnika, Michael Scharfstein, Roshni C. Cooper, Hongshen Ma, Joel L. Dawson and Seward B. Rutkove, "A Handheld Electrical Impedance Myography Probe for the Assessment of Neuromuscular Disease", *30th Annual International IEEE EMBS Conference, Vancouver, British Columbia, Canada*, August 20-24, 2008.

[3] Michael Scharfstein, "A Reconfigurable Electrode Array for Use in Rotational Electrical Impedance Myography", *Thesis for the Degree of Master of Engineering in Electrical Engineering and Computer Science*, Massachusetts Institute of Technology, September 2007.

[4] M. Mancio, J. Zhang and P.J.M. Monteiro, "Nondestructive surface measurement of corrosion of reinforcing steel in concrete", *Canadian Civil Engineer*, v. 21, no. 2, pp. 12-14, May 2004.

[5] Agilent 4294A Precision Impedance Analyzer, Data Sheet, available at www.agilent.com April 2008.

[6] Agilent 4287A RF LCR Meter 1 MHz - 3 GHz, Data Sheet, available at www.agilent.com October 2004.

[7] Agilent 4395A Network/Spectrum/Impedance Analyzer, Data Sheet, available at www.agilent.com February 2008.

[8] *IEEE Std. 1057-1994, Standard for Digitizing Waveform Records*, The Institute of Electrical and Electronics Engineers, New York, December 1994.

[9] Pedro M. Ramos and A. Cruz Serra, "A new sine-fitting algorithm for accurate amplitude and phase measurements in two channel acquisition systems", *Measurement, Elsevier*, Vol. 41, No. 2, pp. 135-143, Feb. 2008.

[10] P. Händel, "Parameter estimation employing a dual-channel sine-wave model under a gaussian assumption," *IEEE Trans. Instr. Meas.,* vol. 57, no. 8, pp. 1661–1669, Aug. 2008.

[11] Pedro M. Ramos, Fernando M. Janeiro, Mouhaydine Tlemçani and A. Cruz Serra, "Recent Developments on Impedance Measurements With DSP-Based Ellipse-Fitting Algorithms", *IEEE Transactions on Instrumentation and Measurement*, vol. 58, no. 5, May 2009.

[12] R. Halír and J. Flusser, "Numerically stable direct least squares fitting of ellipses," in *Proc. WSCG*, Feb. 1998, pp. 125–132.

[13] M. Fonseca da Silva, Pedro M. Ramos, A. Cruz Serra, "Impedance Measurement Using Sine Fitting Algorithms", *12th IMEKO TC4 International Symposium Electrical Measurements and Instrumentation*, September 25−27, 2002.

[14] Neil P. Albaugh, *The Instrumentation Amplifier Handbook*, Burr-Brown Corporation.

[15] Analog Devices AD8250 PGA, Data Sheet, available at www.analog.com June 2009.

[16] Analog Devices AD7980 ADC, Data Sheet, available at www.analog.com June 2009.

[17] Tamás Zoltán Bilau, Tamás Megyeri, Attila Sárhegyi, János Márkus, István Kollár, “Four Parameter Fitting Of Sine Wave Testing Result: Iteration and Convergence”, *Computer Standards and Interfaces*, Vol. 26, pp. 51-56, 2004.

[18] Pedro M. Ramos, Fernando M. Janeiro, Tomáš Radil, “Comparison of Impedance Measurements in a DSP using Ellipse-fit and Seven-Parameter Sine-fit Algorithms”, *Elsevier Editorial System(tm) for Measurement*, 2009.

A – Reference Impedance Measurement Results

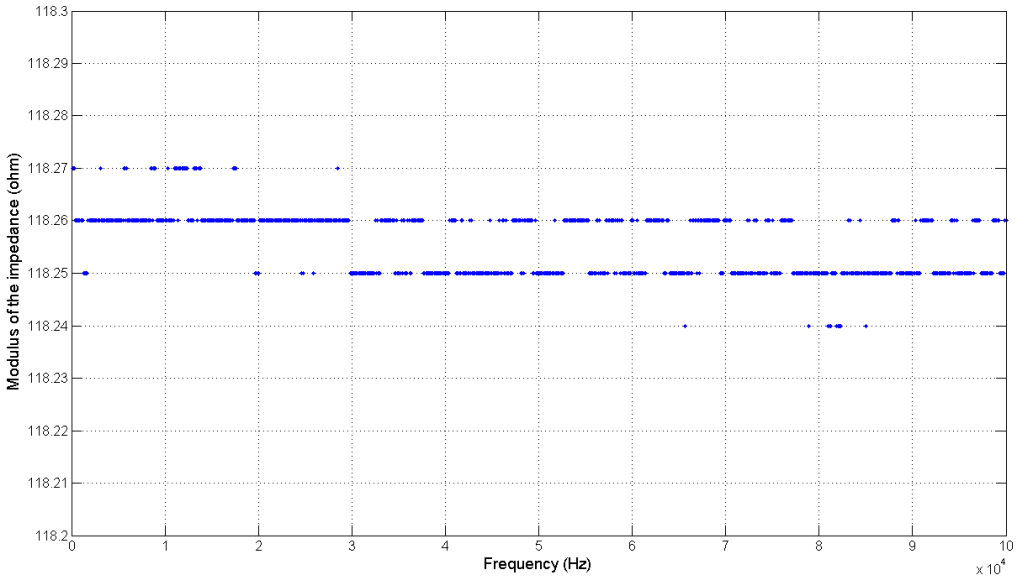


Fig. A.1 - Modulus of the 120 Ω reference impedance.

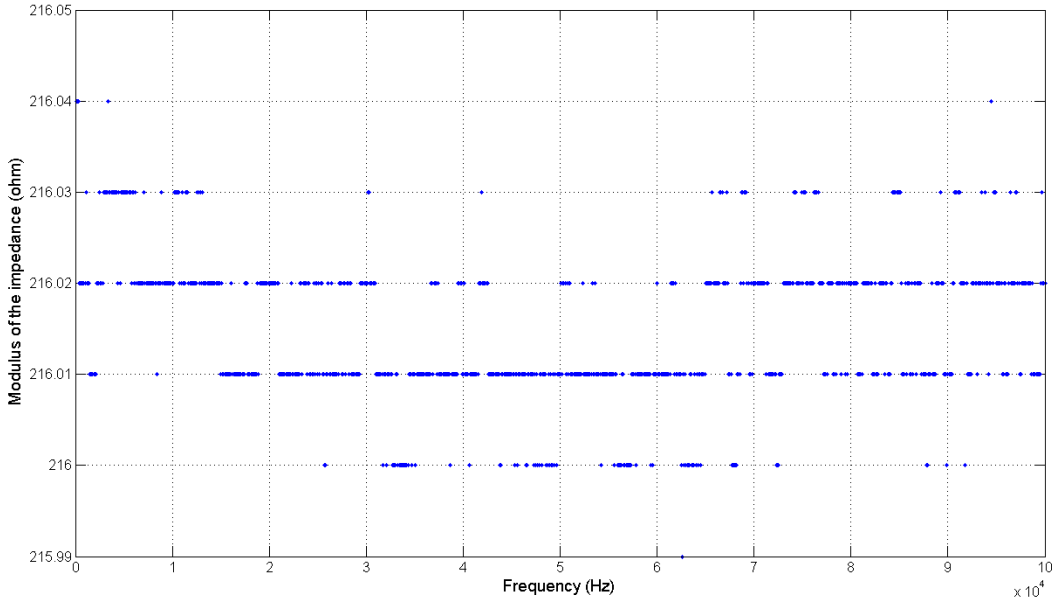


Fig. A.2 - Modulus of the 220 Ω reference impedance.

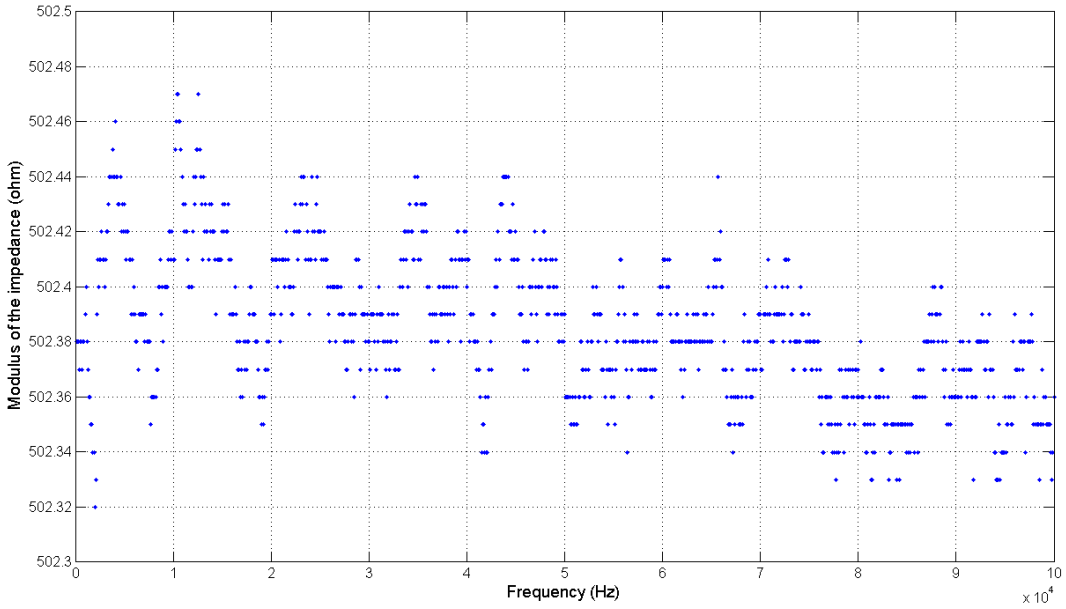


Fig. A.3 - Modulus of the 510 Ω reference impedance.

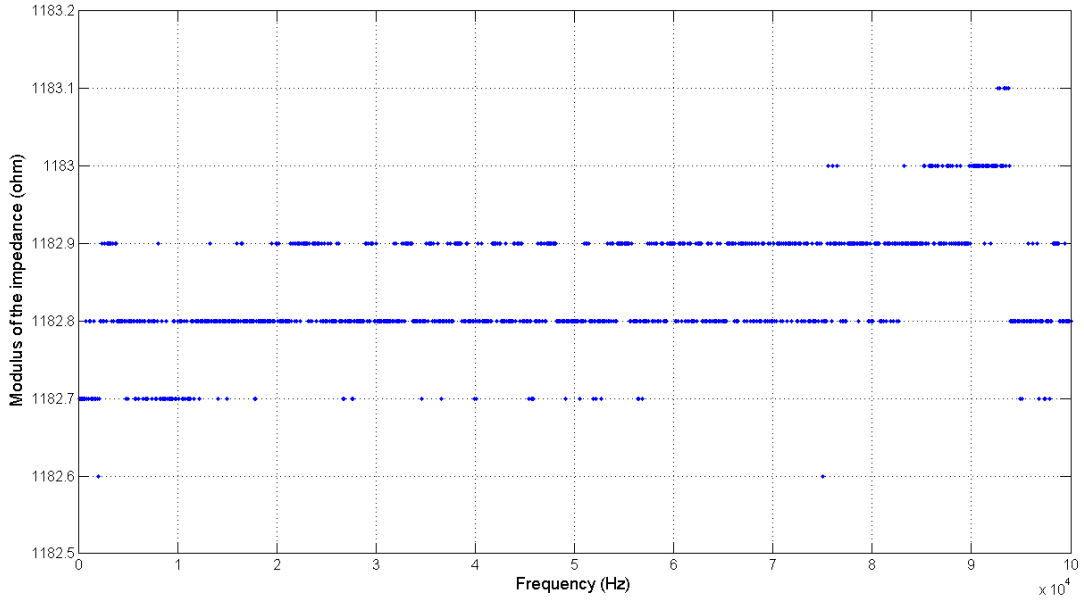


Fig. A.4 - Modulus of the 1200 Ω reference impedance.

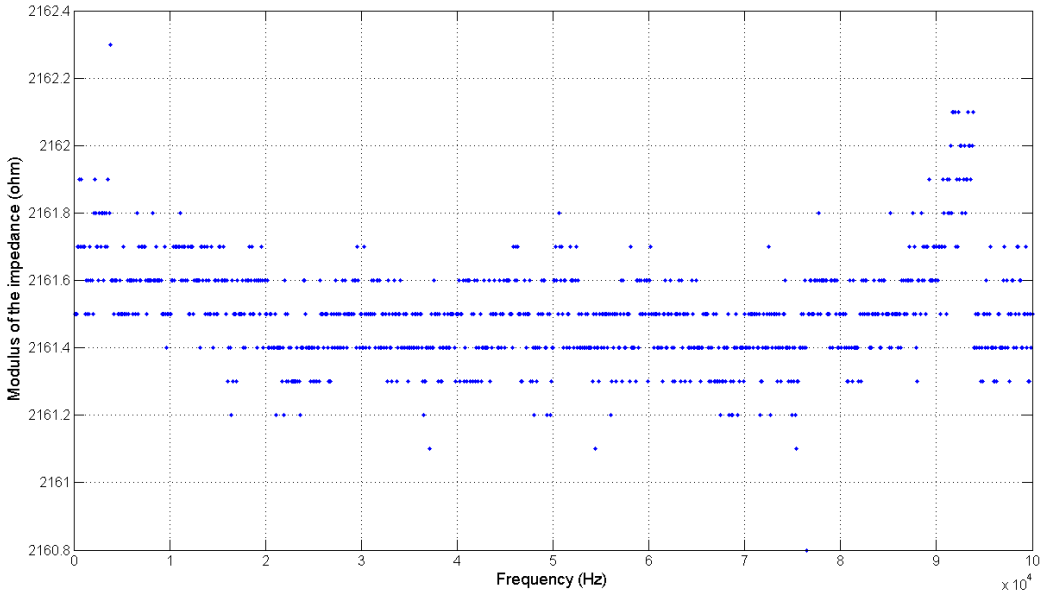


Fig. A.5 - Modulus of the 2200 Ω reference impedance.

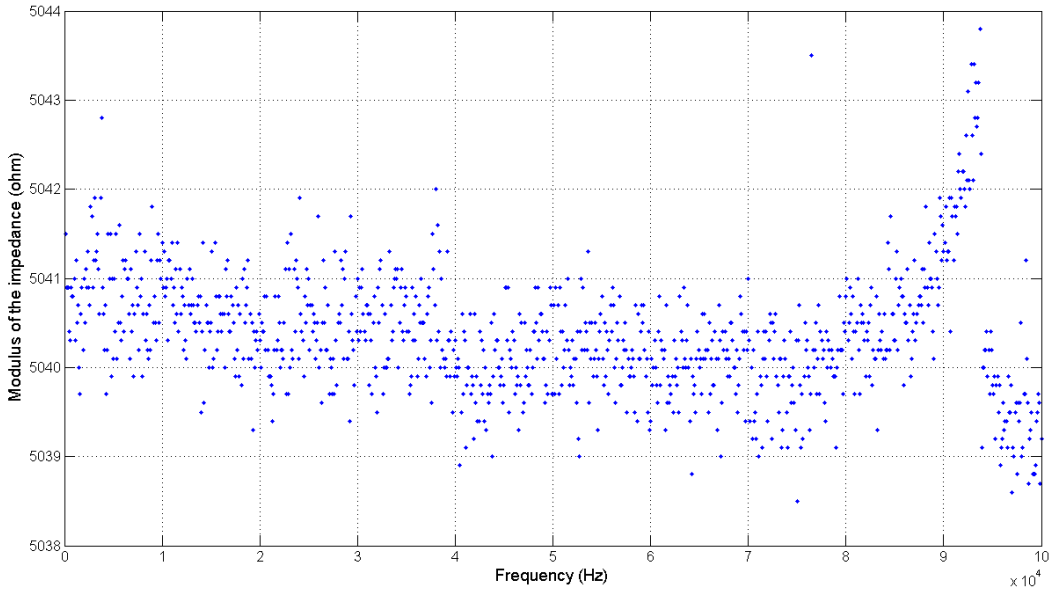


Fig. A.6 - Modulus of the 5100 Ω reference impedance.

B – Encoding of Acquired Data to Transmit to PC

Upon the completion of the measurement procedure of the unknown impedance, it is necessary to transmit the measurement results to the computer. The acquired samples of the signals across the impedances must also be sent to the computer.

Given the amount of data to be sent to the computer, 1024 samples per channel with 16 bits per sample, it was necessary to implement an algorithm to encode the acquired data so it could be sent to the computer in an efficient way, in order to reduce the necessary time for the data transmission.

The acquired data is in the unsigned integer format. In order to transmit the data to the computer the data has to be converted into the ASCII (American Standard Code for Information Interchange) format. Each character of the ASCII table is represented in 7 bits, but each sample is represented in 16 bits, so, it is necessary to break each sample into “pieces”, three to be exact: two “pieces” of 6 bits and one of 4 bits. So, for each sample it is necessary to transmit three characters. Two characters corresponding to two integers of 6 bits and one character corresponding to an integer of 4 bits. Fig. B.1 illustrates the process for an ADC sample of 65535.

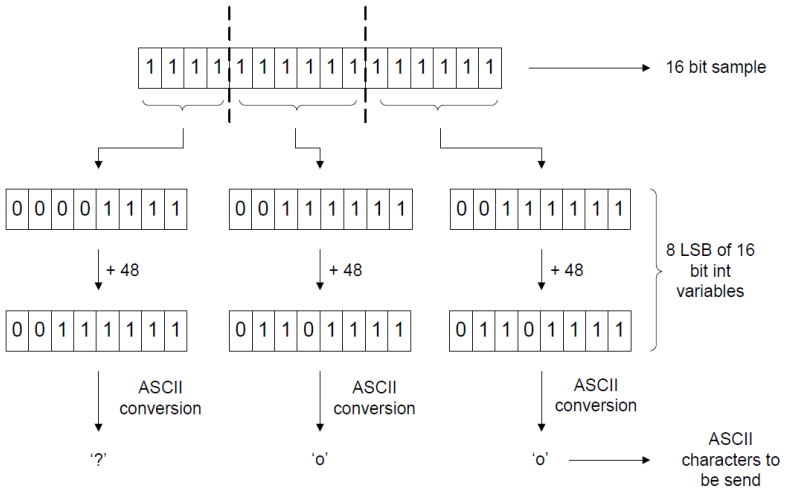


Fig. B.1 - Encoding of an ADC sample.

Although each ASCII character can be represented in 7 bits, it was decided to use a maximum of 6 bits to represent each part of the sample, because before each integer is converted into the ASCII format, each integer is added a value of 48 to avoid using any of the special characters of the ASCII table, some examples are: STX (Start Of Text), ACK (Acknowledge) and ESC (Escape).

The conversion from the integer values to ASCII characters are done by assigning the integer values to char type variables.

The flowchart from Fig. B.2 describes the execution of the algorithm.

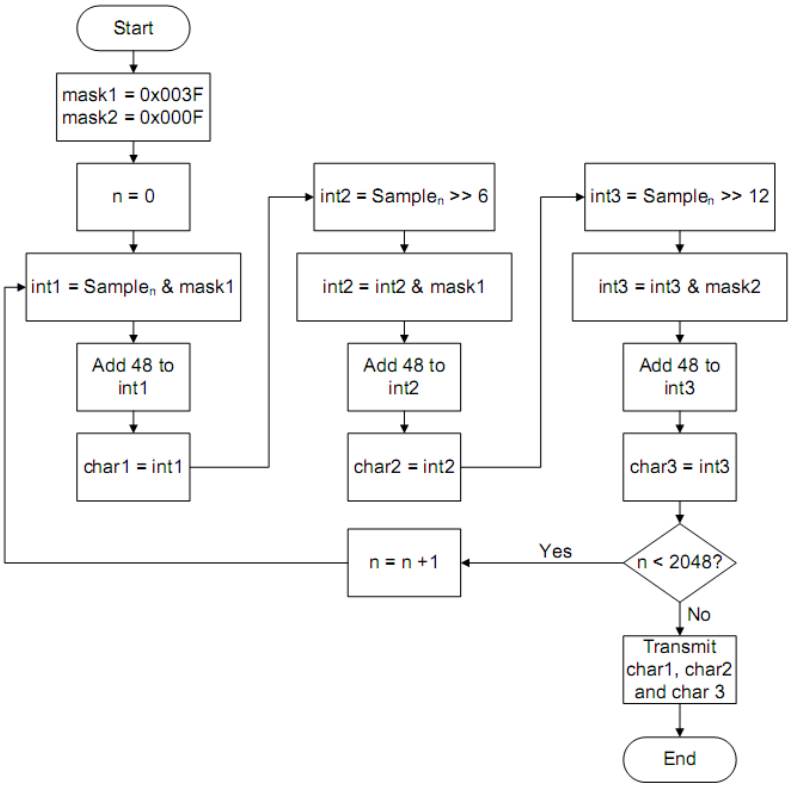


Fig. B.2 - Flowchart describing the encoding of the acquired samples.

In the computer end, it is necessary to perform the inverse operations described, in order to obtain the correct values of the decoded samples transmitted by the dsPIC.

C – Schematic of the Measurement Circuit

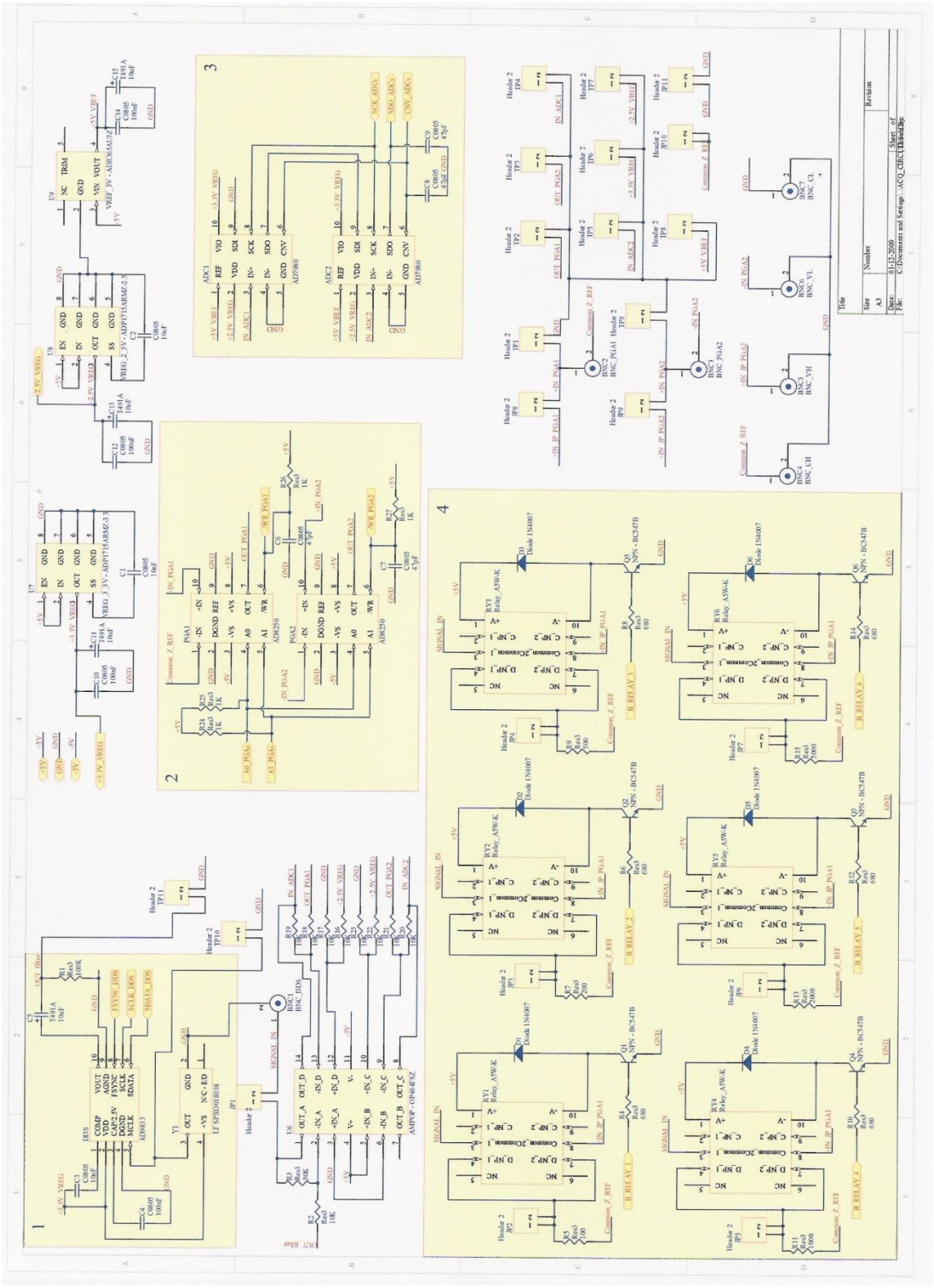


Fig. C.1 - Schematic of the measurement circuit.

D – Schematic of the Complete System

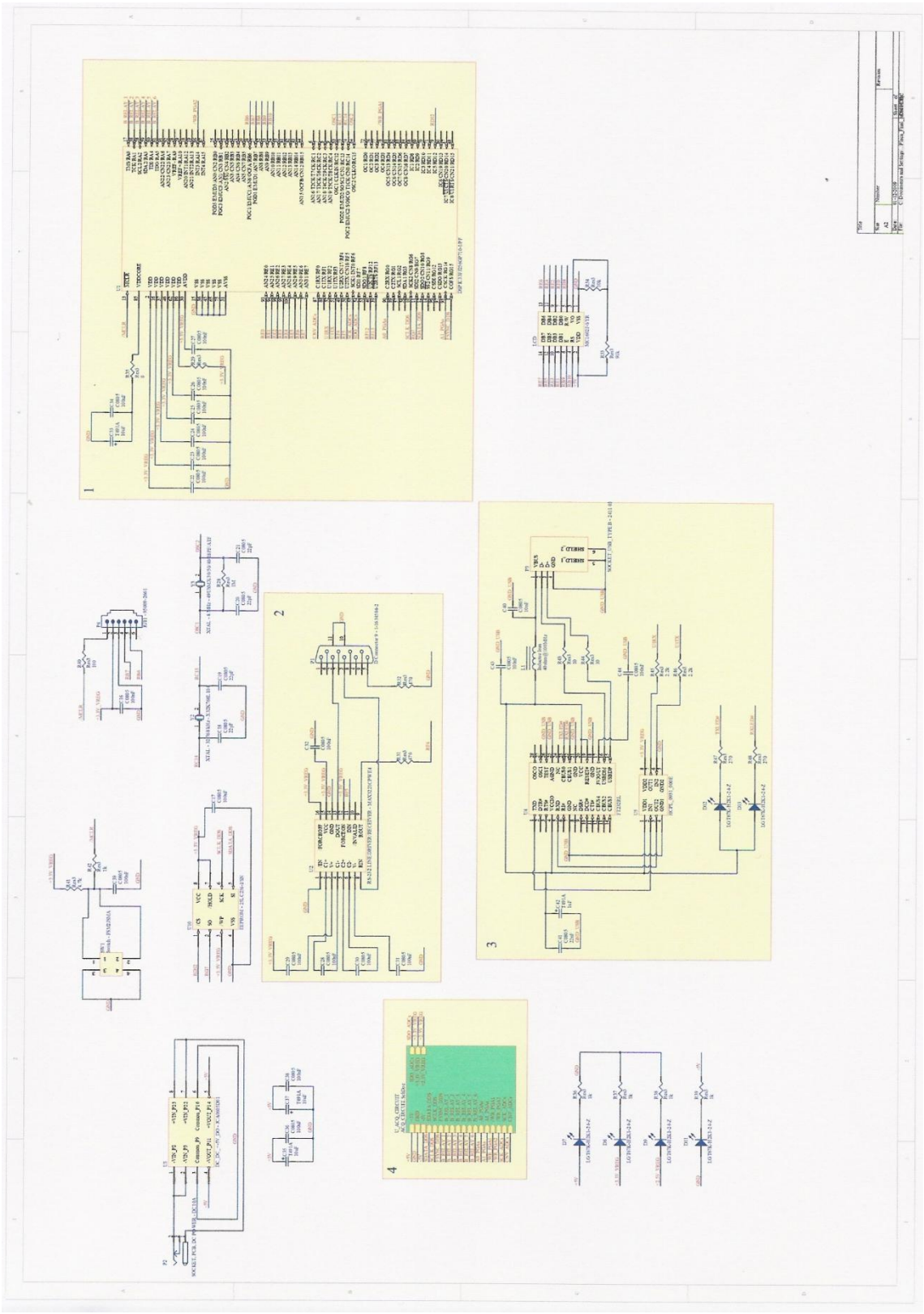


Fig. D.1 - Schematic of the complete system. Square 3 represents the hardware for the USB interface. The symbol in square 4 represents the circuit from Fig. C.1 in appendix C.

E – PCB of the Complete System

The entire system was implemented in a PCB which includes the processing unit and the digitizing channels.

The layout of the board is presented in Fig. E.1 (top layer) and Fig. E.2 (bottom layer).

Since the board is still in production, it was not possible to test it in order to obtain experimental results regarding the measurement of impedances, so only the layout of the board is presented in this appendix.

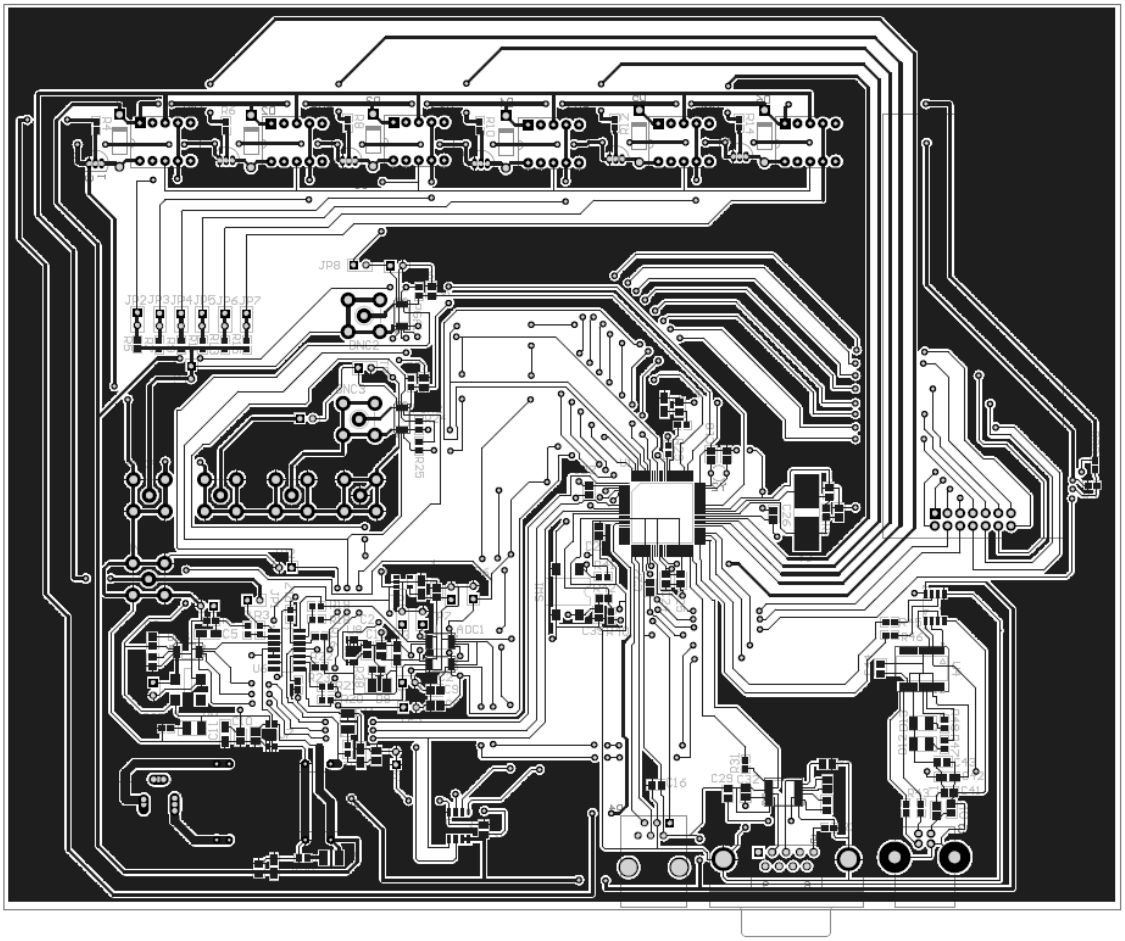


Fig. E.1 - Layout of the PCB of the entire system - top layer.

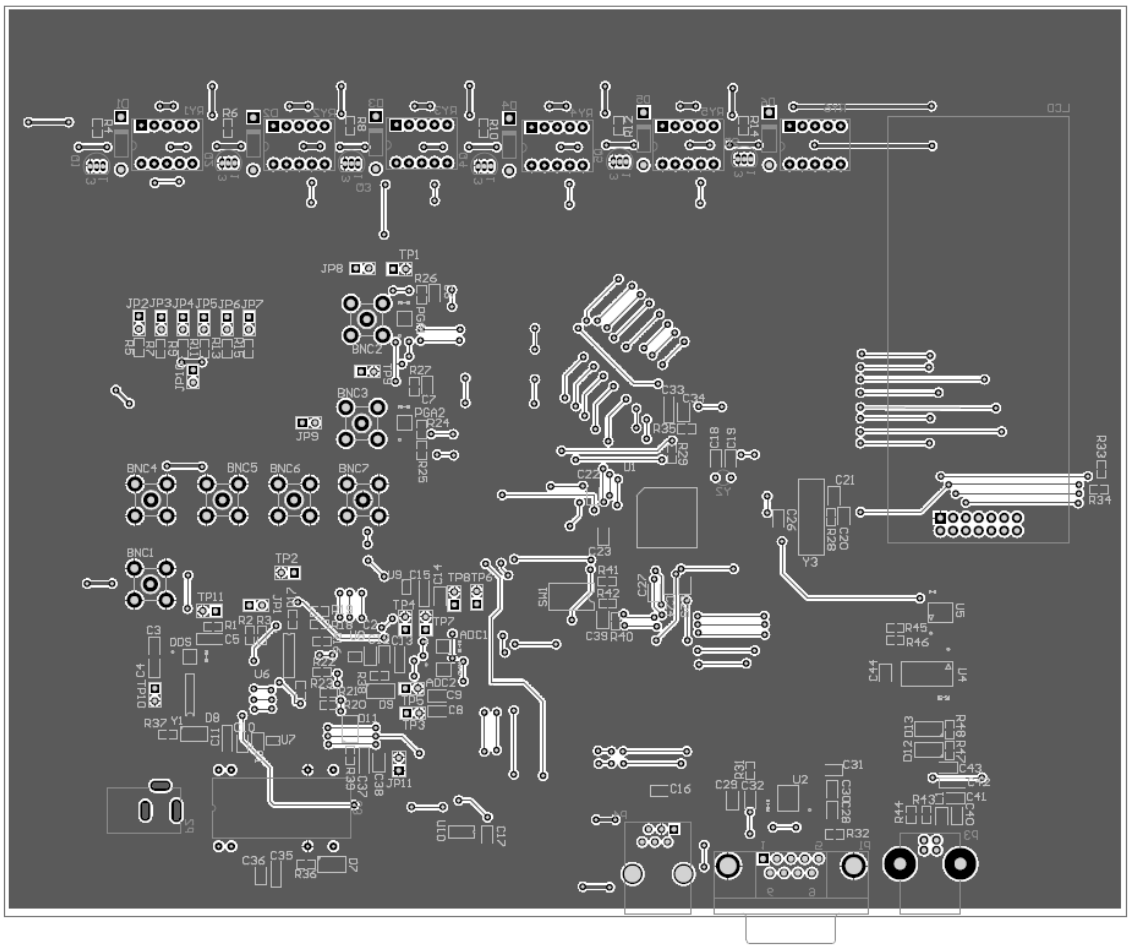


Fig. E.2 - Layout of the PCB of the entire system - bottom layer.

Although the board includes the USB interface given by the FT232RL USB to UART device, this feature was not tested yet because, as stated before the board is still in production.