UNIVERSIDADE DE SÃO PAULO
INSTITUTO DE FÍSICA DE SÃO CARLOS

Julia Marcolan Teixeira

Digital magnetic resonance spectrometer (DMRS) from CIERMag: calibration and relaxometry measurements methodology

São Carlos

2022

Julia Marcolan Teixeira

Digital magnetic resonance spectrometer (DMRS) from CIERMag: calibration and relaxometry measurements methodology

Dissertation presented to the Graduate Program in Physics at the Instituto de Física de São Carlos da Universidade de São Paulo, to obtain the degree of Master in Science.

Concentration area: Applied Physics

Option: Computational Physics

Advisor: Prof. Dr. Alberto Tanús

Corrected Version
(Original version available on the Program Unit)

São Carlos

2022

Teixeira, Julia Marcolan
Digital magnetic resonance spectrometer (DMRS) from CIERMag: calibration and relaxometry measurements methodology / Julia Marcolan Teixeira; advisor Alberto Tannus - corrected version -- São Carlos 2023.
112 p.

Dissertation (Master's degree - Graduate Program in Computational Physics) -- Instituto de Física de São Carlos, Universidade de São Paulo - Brasil , 2023.

1. Nuclear magnetic resonnace. 2. Relaxometry. 3. Digital magnetic resonance spectrometer. 4. FPGA. I. Tannus, Alberto, advisor. II. Title.

I dedicate this work to all those who left us during the coronavirus pandemic.

ACKNOWLEDGEMENTS

I'm very honored and grateful for the opportunity to be part of the team of *Centro de Imagens e Espectroscopia por Ressonância Magnética* (CIERMg) and I would like to thank my advisor Prof. Dr. Alberto Tannús for all the patience and knowledge shared during these two years. I would also like to thank the entire CIERMag research group for all their hard work and dedication and also for providing a welcoming learning environment. In special, I would like to thank Dr. Edson Vidoto and Dr. Mateus Martis for all the support and dedication to CIERMag, and for the direct influence, you had on this work, from monitoring in the laboratory to corrections and suggestions in the text. Thanks to students Gabrielly Rios and Hendrik Louzada for all their help during the lab days, I wish you a bright future.

I would like to thank all my family for all the support they have always given me in choosing my career. Thank you to all my uncles and aunts, cousins, and cousins. In particular, thanks to my grandmother Luzia, my mother Aline, my father Marinelso, and my brother Murilo for all their support and for never letting me give up. Thanks to my friend Isys and my goddaughter Gaia for always being around and supporting me.

I am very grateful to all the good friends I made in São Carlos and for that, I could not forget to thank them here. In particular, thanks to my neighbors Camila and Bruna for all their company and support during the difficult years of the COVID-19 pandemic. Thanks also to my dear friends Lilianne and Kamila for being my career inspirations and for all the academic support you've given me since we met. In particular, thanks to my partner Julia Jaccoud for being my inspiration, for all her support, and for her companionship. Also, I would like to thank Luciano and Claudia Jaccoud for all their love. Thanks to my friend Thash, for his help with the LateX and also to my friend Rafael for the tea ceremony that calmed me down during the dissertation delivery week.

This master's degree took place in a very difficult time marked by the COVID-19 pandemic, so I could not forget to thank all the health professionals who worked incessantly this year, as well as Brazilian science and the SUS.

Finally, I would like to thank *Universidade de São Paulo* (USP), *Instituto de Física de São Carlos* (IFSC), to the to Center for Magnetic Resonance Research (CMRR) and all our collaborating colleagues, and the *Conselho Nacional de Desenvolvimento Científico e Tecnológico* (CNPq) for the possibility of developing this research.

"Ever tried. Ever failed. No matter. Try again. Fail again. Fail better."
Samuel Beckett

ABSTRACT

TEIXEIRA, J. M. **Digital magnetic resonance spectrometer (DMRS) from CIERMag: calibration and relaxometry measurements methodology**. 2022. 112p. Dissertation (Master in Science) - Instituto de Física de São Carlos, Universidade de São Paulo, São Carlos, 2022.

Nuclear Magnetic Resonance (NMR) has become a powerful and ubiquitous analytical tool scientific and technological areas, such as medicine, industry, agriculture, materials science, etc. However, according to a review by Centro de Imagens e Espectroscopia por Ressonância Magnética (CIERMag), although practical to use in most applications, current NMR commercial equipment are restrictive for modifications and does not allow new pulse sequences, or pulse shapes - i.e. adiabatic pulses - to be introduced easily. To avoid these limitations present in commercial NMR equipment the CIERMag group developed the Digital Magnetic Resonance Spectrometer (DMRS) based on Field-Programmable Gate Arrays (FPGAs), and a software based on the Python Magnetic Resonance Framework (PyMR) for planning and control of the experiment. This development included an Integrated Development Environment (IDE), a graphical instruction interface, the Console for acquiring, organizing, visualizing and storing data, a primitive "F" language, used to develop the pulse sequences, which has a compiler and a language-sensitive editor. In CIERMag context, an NMR Method represents an experiment project as a modern software project, containing all information about the experiment from pre-processing to post-processing. In this study, NMR Methods for DMRS calibration such as spectrometer's resonance frequency adjustment, and flip angle (B_1 amplitude) calibration were developed, in addition to the implementation of classic NMR sequences such as Carr-Purcell-Meiboom-Gill (CPMG) and Inversion Recovery (IR). The relaxation times of water at 35°C were calculated with the use of CPMG and Inversion recovery sequences. The values for the spin-spin and spin-lattice relaxation times are respectively $T_2 = 2.095 \pm 0.006$ s and $T_1 = 3.49 \pm 0.05$ s. In addition, as predicted by the literature, we found linear relationships between the concentrations of copper sulfate in water for four different samples with the inverse of the transverse and longitudinal relaxation times. A survey on the influence of temperature variation on the spectrometer operating frequency is also provided. The results for the calibration methodologies comply with the pre-experiment adjustments necessary for measurement reliability. Furthermore, measurements of relaxation times that are following the literature demonstrate that the equipment is functional and ready for use. Therefore, DMRS is an alternative that avoids obsolescence and simplifies the development of new NMR techniques.

Keywords: Nuclear-magnetic-resonance. Relaxometry. Digital-magnetic-resonance-spectrometer. FPGA.

RESUMO

TEIXEIRA, J. M. **Calibração e metodologia de medidas de relaxometria aplicadas ao Espectrômetro de Ressonâmcia Magnética Digital (DMRS) do CIERMag**. 2022. 112p. Dissertação (Mestrado em Ciências) - Instituto de Física de São Carlos, Universidade de São Paulo, São Carlos, 2022.

A Ressonância Magnética Nuclear (RMN) está presente em várias aplicações científicas e tecnológicas em áreas, como na medicina, na indústria, na agricultura, ciências de materiais, etc. Apesar disto, de acordo com um levantamento feito pelo Centro de Imagens e Espectroscopia por Ressonância Magnética (CIERMag), os equipamentos comerciais de RMN, embora práticos para o uso em diversas aplicações, são restritivos quanto a drásticas modificações e não permite que novas sequências de pulso, ou formas de pulso - ou seja, pulsos adiabáticos - sejam introduzidos facilmente. O CIERMag também desenvolveu o *software* baseado no *Python Magnetic Resonance Framework* (PyMR) para planejamento e controle do experimento que abrange: um *Integrated Development Environment* (IDE), uma interface de prescrição gráfica, o Console para aquisição, organização, visualização e armazenamento de dados, e a linguagem primitiva "F", utilizada para desenvolver as sequências de pulsos, que conta com um compilador e um editor sensível à linguagem. No contexto do CIERMag, um método representa um projeto de sequência de pulsos de RMN que contém toda a informação sobre ele. Neste trabalho, foram desenvolvidos métodos de RMN para calibração do DMRS, como ajuste da frequência de ressonância do espectrômetro e calibração do ângulo de *flip* (amplitude de B_1), além da implementação de sequências clássicas de RMN, como *Carr-Purcell-Meiboom-Gill* (CPMG) e Inversão Recuperação (IR). Os tempos de relaxação da água a 35°C foram calculados com o uso das sequências CPMG e IR. Os valores para os tempos de relaxação spin-spin e spin-rede são, respectivamente, $T_2 = 2.095 \pm 0.006$ s e $T_1 = 3.49 \pm 0.05$ s. Além disso, conforme previsto pela literatura, encontramos relações lineares entre as concentrações de sulfato de cobre na água para quatro diferentes amostras com o inverso dos tempos de relaxação transversal e longitudinal. Um levantamento sobre a influência da variação de temperatura na frequência de operação do espectrômetro também é fornecido. Os resultados das metodologias de calibração atendem aos ajustes pré-experimentais necessários para a confiabilidade das medidas. Além disso, medidas dos tempos de relaxamento que seguem a literatura demonstram que o equipamento está funcional e pronto para uso. Portanto, o DMRS é uma alternativa que evita a obsolescência do *hardware* e simplifica o desenvolvimento de novas técnicas de RMN.

Palavras-chave: Ressonância-magnética-nuclear. Relaxometria. Espectrômetro-digital. FPGA.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

NMR	Nuclear Magnetic Resonance
MRI	Magnetic Resonance Imaging
MRS	Magnetic Resonance Spectroscopy
IFSC	Instituto de Física de São Carlos
ToRM	Magnetic Resonance Tomography
SUS	Unified Health System
DMRS	Digital Magnetic Resonance Spectrometer
PyMR	Python Magnetic Resonance Framework
IDE	Integrated Development Environment
LSP	Language Server Protocol
CPMG	Carr-Purcell-Meiboom-Gill
IR	Inversion Recovery
MRR	Magnetic Resonance Relaxometry
ESR	Electron Spin Resonance
FID	Free Induction Decay
TE	Echo Time
TR	Repetition Time
RF	Rafiofrequency Pulse
FPGA	Field-Programmable Gate
VHSIC	Very-High-Speed Integrated Circuit
VHDL	Hardware Description Language
ADC	Analog-to-Digital converter
DAC	Digital-to-Analog converter
DHF5	Hierarchical Data Format

LIST OF SYMBOLS

γ	Gyromagnetic ratio
τ	Torque
$\vec{\mu}$	Magnetic Moment
$\vec{L}$	Angular Momentum
H	Hamiltonian
$\vec{B}$	Magnetic Field
ω_0	Larmor Frequency
K_B	Boltzmann constant
I	Spin Quantum number
T_1	Longitudinal Relaxation Time
T_2	Transversal Relaxation Time
τ_I	Inversion Time

CONTENTS

1 INTRODUCTION

In the early 1920s, observations in the Hydrogen spectrum revealed spectral lines split into closely spaced lines, giving rise to a fine structure. The Bohr-Sommerfeld atomic model could not explain the extra spectral lines and observations indicated some corrections should be made to the model. In 1922, Pauli suggested the use of half-integer numbers as magnetic quantum numbers, and, in 1924, proposed the existence of a nuclear angular momentum (spin) for explaining the lines in the Hydrogen spectrum.(1)

Interested in investigating the nuclear magnetic properties of crystals, Isidor Rabi developed methods for evaluations of the precession frequency of atomic systems subjected to magnetic fields and was able to measure the magnetic moment of the proton. The researcher was awarded the Nobel Prize in 1944 for the method of recording magnetic resonance properties of atomic nuclei. (2)

Working independently, Edward Purcell and Felix Bloch demonstrated the precession frequency of nuclei depended on the strength of the magnetic field applied. (3) Bloch precisely measured the Hydrogen gyromagnetic ratio and nuclei precession moments and proposed a method for the detection of Nuclear Magnetic Resonance (NMR) signals with receiver coils. (4) Bloch and Purcell were awarded the Nobel Prize in 1952 for those contributions.

In 1966, Richard Ernest used the Fourier Transform to convert the time-dependent decay signal into a frequency domain NMR one. In 1973, Peter Mansfield and Paul Lauterbur published methods for NMR imaging generation with the use of gradient fields and, in 1974, Raymond Damadian patented the NMR scanner idea for pathological tissue detection, obtaining the first successful human NMR imaging in 1977. Lauterbur, a physical chemist, and Mansfield, a physicist won the Nobel Prize in Medicine in 2003.

Techniques that use NMR (e.g., Magnetic Resonance Imaging (MRI) and Magnetic Resonance Spectroscopy (MRS)) are powerful, since they enable investigations on molecular structures and biochemical profiles of tissues, thus helping the identification of diseases. In addition to its wide applicability, NMR is a non-invasive technique that causes danger to neither the sample nor the patient.

In 1983, the Magnetic Resonance group of the *Instituto de Física de São Carlos* IFSC (later on the *Centro de Imagens e Espectroscopia por Ressonância Magnética* - CIERMag) started to build a pulsed Magnetic Resonance Spectrometer with a resistive magnet and a rudimentary control interface. In 1984, images of small vegetables (okra)

were obtained (5) and were considered the first images of the world's southern hemisphere. The software that controlled the CIERMag spectrometer was developed in FORTRAN, C, Assembly, and Pascal, and an event generator controlled the pulse sequence. In 1987, ToRM (Magnetic Resonance Tomography) project was initiated towards improving and automating the spectrometer control (6) and, in 1995, a clinical system developed by the group began operating at *Santa Casa* hospital, in São Carlos. The system operated for eight years and generated more than eight thousand exams mostly performed by the Unified Health System (SUS).

NMR techniques and equipment have evolved and are now present in several sectors of society (e.g., medical, industrial, and agricultural). However, according to a review conducted by CIERMag, the current NMR commercial equipment does not have tools for facilitating the development of new NMR techniques. Although practical for use in most applications, it is restricted to drastic modifications required by research - as an example, it does not provide an easy way to introduce new pulse sequences. (7) Another difficulty faced by researchers from CIERMag was the introduction of adiabatic pulses, i.e. frequency- and amplitude-modulated pulses that rotate the magnetization vector differently from conventional pulses (8), in commercial equipment. The lack of complete knowledge of the specs of the full Radiofrequency path made it difficult to predict the AM and PM (or FM) shapes that would produce the desired action on the spin system, crucial to develop research on Adiabatic Pulses.

Towards solving the problems related to the restrictions imposed on commercial equipment, the CIERMag team began to develop the Digital Magnetic Resonance Spectrometer (DMRS) with the use of reconfigurable hardware powered by Field-Programmable Gate Arrays (FPGAs). CIERMag team has also developed software based on Python Magnetic Resonance Framework (PyMR) (9) for experiment planning and control, which includes an Integrated Development Environment (IDE), a Console for acquisition, organization, visualization, and storage of data. (10) Enriching these software packages and allowing seamless control of the synthesized hardware, the CIERMag team created the "F-language" with its own compiler and a Language Server Protocol (LSP), making it the very first MR Methods developing environment to use modern computational tools to deal with the MR experiment as a software project. F is a primitive language created for NMR pulse sequence development and intends to be robust and reusable for its specialized applications. (11)

This dissertation addresses the development of Methods compatible with the DMRS, here divided in two categories, namely calibration, and analysis. The first encompasses Methods that adjusted the spectrometer resonance frequency, and the radiofrequency flip angle (B_1 amplitude), whereas the second comprises analysis methods, including classic NMR ones such as CPMG and Inversion Recovery.

The dissertation is structured as follows:

- Chapter 1 review the main concepts and instruments of NMR;
- Chapter 2 provides a detailed contextualization of DMRS;
- The Materials and Methods chapter discusses the development of methods compatible with DMRS and describes the experimental apparatus and the experiments performed;
- Finally, Chapter 4 is devoted to the results, a comparison between two different frequency adjustment methods, a discussion on flip angle adjustment, and a survey on the influence of temperature variation on the spectrometer operating frequency.

The relaxation times of water at 35°C, were calculated with the use of CPMG and Inversion recovery sequences. The values for the spin-spin and spin-lattice relaxation times are respectively $T_2 = 2.095 \pm 0.006$ s and $T_1 = 3.49 \pm 0.05$ s. In addition, as predicted by the literature, we found the linear relationships between the concentrations of copper sulfate in water for four different samples with the inverse of the transverse and longitudinal relaxation times.

The results for the calibration methodologies comply with the pre-experiment adjustments necessary for measurement reliability. Furthermore, measurements of relaxation times that are in accordance with the literature (12) demonstrate that the equipment is functional and ready for use. Therefore, DMRS is an alternative that avoids obsolescence and simplifies the development of new NMR techniques.

The dissertation includes an appendix with a Reference Manual, which contains an instructional reference, detailed description of the F language.

2 THEORETICAL FOUNDATION

2.1 Magnetic Resonance

Nuclear Magnetic Resonance (NMR) is a physical phenomenon that describes the behavior of particles with a magnetic moment in the presence of a magnetic field, which is proportional to their angular momentum. Techniques that use NMR like Magnetic Resonance Imaging (MRI), Magnetic Resonance Spectroscopy (MRS), and Magnetic Resonance Relaxometry (MRR) are powerful and present in diverse scientific and technological applications in areas such as medicine, industry, agriculture, and materials sciences, among others. In addition to its wide applicability, NMR is a non-invasive technique and offers danger to neither the sample nor the patient.

2.1.1 Basic atomic nuclei magnetic properties and quantum description

The atomic nuclei have two important properties, namely, spin and charge. Spin refers to the intrinsic angular momentum and the possible quantum orientations of nuclei and subatomic particles and is characterized by quantum number I. Protons and neutrons in the atomic nuclei have spin ½. Since spin pairs tend to cancel each other, in this way only nuclei with an odd number of protons and neutrons show a spin other than zero - i.e. the Carbon (^{12}C) has 6 protons and 6 neutrons in its nuclei; therefore, the nuclear spin is zero. Because of an intrinsic angular momentum, it can be thought that the nuclei experience a rotational movement around their axis and, consequently, the electrical charge creates a small current loop that produces a magnetic field in the nucleus axis direction, giving rise to the magnetic moment. The atomic nucleus behaves as a magnetic dipole; the magnetic moment and the angular momentum are related by equation 2.1.

$$\vec{\mu} = \gamma\vec{L} = \gamma\vec{I} \tag{2.1}$$

where γ is the gyromagnetic ratio, which is a constant characteristic of every chemical species and reflects the magnetic nuclei properties.

When an atomic nucleus is placed in a magnetic field, the interaction between the magnetic field and the magnetic moment causes a splitting of atomic energy levels. Such an interaction is described by the Zeeman Hamiltonian 2.2.

$$H = -\vec{\mu}.\vec{B} \tag{2.2}$$

In the presence of an external magnetic field, $\vec{B} = B_0\hat{z}$, and considering equation 2.1, the equation Zeeman Hamiltonian equation 2.2 can be rewritten as 2.3.

$$H = -\gamma I_z B_0 \tag{2.3}$$

Hydrogen (^1H)is the most abundant chemical element in both nature and the human body and has the simplest nuclei, formed by only one proton. Consequently, due to its high sensitivity, it is commonly used in NMR experiments. Table 1 shows the gyromagnetic ratio (γ) for spin ½ nuclei that are interesting for NMR.

Table 1 – Gyromagnetic ratio in MHz/T from nuclei that are interesting for NMR.

Núcleo	**Constante giromagnética (MHz/T)**
^{1}H	42,58
^{13}C	10,71
^{15}N	4,31
^{19}F	40,05
^{31}P	17,23

Source: By the author.

The spin component of the angular momentum has only some possible values imposed by the angular momentum quantization. The intrinsic angular momentum of Hydrogen is a half-integer and the possible values of $I_z = \pm 1/2$. Only two possible values of energy, defined by 2.4, can be found for the Hydrogen case:

$$E = \left(\pm\frac{1}{2}\right)\gamma\hbar B_0 \tag{2.4}$$

The spins are aligned parallel with the magnetic field, $\vec{B} = B_0\hat{z}$ at the lowest energy level, determined by $E_{+1/2}$, and aligned antiparallel with the magnetic field for the highest energy level, determined by $E_{-1/2}$. The transitions between those two energy levels are possible only if the system receives or releases energy, whose amount should be equal to 2.5.

$$\Delta E = h\nu = \hbar w \tag{2.5}$$

The amount of energy to be exchanged for the Hydrogen two atomic two-level system is 2.6,

$$\Delta E = E_{+1/2} - E_{-1/2} = \gamma B_0 \hbar \tag{2.6}$$

Equations 2.5 and 2.6 are used to calculate the relation 2.7.

$$w_0 = \gamma B_0 \tag{2.7}$$

The equation is called Larmor relation and frequency w_0 is called Larmor frequency and depends only on the gyromagnetic ratio and the magnetic field intensity. (13) Notice that this formalism borrows a similar interpretation from the Electron Spin Resonance (ESR), where a similar equation is observed. Since electrons and nuclei have opposite charges, we need to introduce a vector representation of this equation to reflect the true behavior of the nucleus in the presence of the magnetic field. And since ESR formalism came first, the electron precession was defined positive proportional to the field, whereas in NMR the vector interpretation of this equation carries a negative sign.

The absence of Planck's constant in the equation 2.7 enables a classical physics phenomenon description, explained in section 2.1.2. Figure 1 displays an energy levels transition scheme.

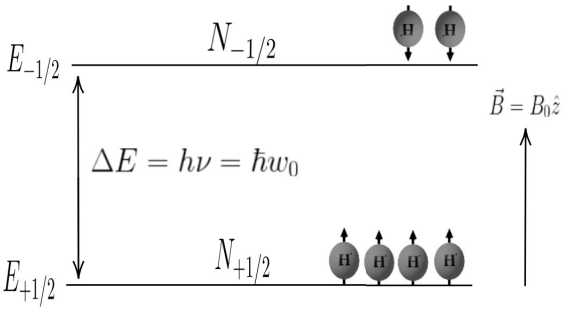

Figure 1 – Energy levels transition scheme for a Hydrogen nucleus.

Source: By the author.

According to Boltzmann's statistics, thermal equilibrium systems tend to accommodate more spins in the lowest energy state; therefore, more spins will be parallel to the external magnetic field rather than antiparallel. The nuclei distribution in the possible energy states is given by Boltzmann's distribution 2.8, where E is the energy associated between the quantum level energy transitions, K_B is Boltzmann's constant, and T is the system temperature.

$$\frac{N_p}{N_a} = e^{\frac{E}{K_B T}} = e^{\frac{\gamma \hbar B_0}{K_B T}} \tag{2.8}$$

2.1.2 Classical Description

The following classical approach can be adopted for the description of the NMR phenomenon: classically a charged body with angular momentum, such as a proton, placed

in the presence of a magnetic field experiences a magnetic force action, and the magnetic moment suffers a torque proportional to the angular momentum temporal variation 2.9.

$$\vec{\tau} = \frac{d\vec{L}}{dt} = \vec{\mu} \times \vec{B} \tag{2.9}$$

Since the magnetic moment can be expressed by 2.1, equation 2.9 provides expression 2.10 for the magnetic moment precession evolution around the magnetic field $\vec{B} = B_0\hat{z}$, with $w_0 = \gamma B_0$ frequency, as illustrated in Figure 2.

$$\frac{d\vec{\mu}}{dt} = -\gamma\vec{\mu} \times \vec{B} \tag{2.10}$$

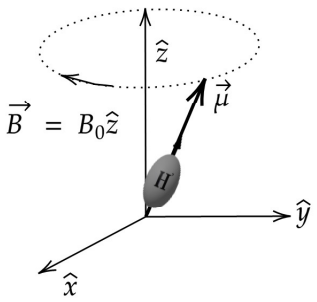

Figure 2 – Precession evolution of the hydrogen nucleus, with magnetic moment μ in the presence of a magnetic field.

Source: By the author.

Macroscopic magnetization vector 2.11 can be defined as the sum of the magnetic moment for each system's atomic nucleus.

$$\vec{M} = \sum_i \vec{\mu_i} \tag{2.11}$$

In a thermal equilibrium situation, the magnetic moment evolution has no phase coherence in the transverse plane, i.e. the plane orthogonal to the external magnetic field direction. In this case, the magnetization transverse component is null. Otherwise, the magnetization vector has phase coherence in the longitudinal plane, i.e. the plane parallel to the magnetic field direction, and the net magnetization points to the external magnetic field direction, as shown in Figure 3.

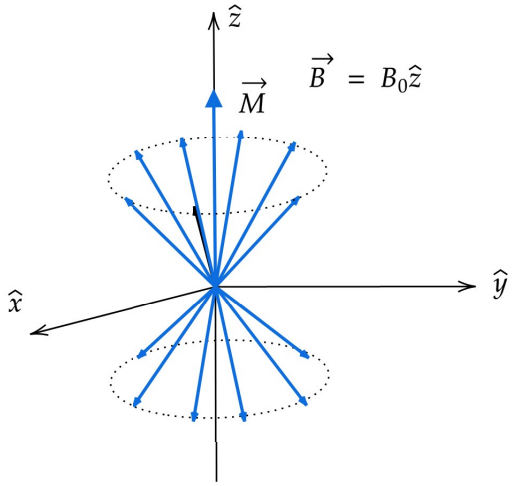

Figure 3 – Precession evolution of the hydrogen nucleus, with magnetic moment μ in the presence of a magnetic field.

Source: By the author.

The temporal evolution for the magnetization vector is given by Bloch equation 2.12 for a non-interagent system.

$$\frac{d\vec{M}}{dt} = -\gamma\vec{M} \times \vec{B} \tag{2.12}$$

2.1.3 Rotation frame of reference and the Radiofrequency Pulses

The rotating frame of reference is considered for a better understanding of the NMR signal. Let us take a reference frame with a unit vector $\hat{x}', \hat{y}', \hat{z}'$, where $\hat{x}', \hat{y}'$ rotates at frequency Ω, as illustrated in Figure 4.

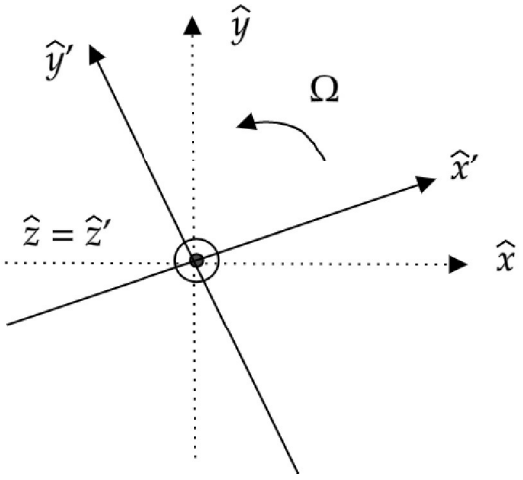

Figure 4 – Rotating frame of reference.

Source: Adapted from BROWN *et al.* (13)

In the rotation frame, equation 2.10 for the magnetic moment evolution can be rewritten as 2.13 in the function of the effective external magnetic field.

$$\left(\frac{d\mu'}{dt}\right)_{\hat{x'}\hat{y'}\hat{z'}} = -\gamma\vec{\mu'} \times \vec{B}_{eff} \tag{2.13}$$

where

$$\vec{B}_{eff} = \vec{B_0} + \frac{\vec{\Omega}}{\gamma} \tag{2.14}$$

If frequency Ω from the rotating frame is equal to the Larmor frequency $\vec{\Omega} = -\omega_0\hat{z}$, the magnetic moment temporal variation is null as can be seen from the equation 2.13. Therefore, $\mu = constant$ and points into $\hat{z} = \hat{z'}$ in this reference frame. Such a condition is called resonance condition.

The detection of the NMR signal requires the removal of magnetization from equilibrium. The signal observed in the NMR experiments consists of the temporal evolution from the transverse magnetization when it returns to the equilibrium situation. A radio frequency (RF) pulse applied to the system removes the magnetization from its equilibrium.

Let us consider an oscillating field of type 2.15

$$\vec{B_1} = B_1[cos(wt)\hat{x} = sen(wt)\hat{y}] \tag{2.15}$$

where $\hat{x}, \hat{y}, \hat{z}$ and $\hat{x'}, \hat{y'}, \hat{z'}$ relationship in the rotarion frame is given by 2.16.

$$\begin{aligned} \hat{x} &= \hat{x'}cos(wt) + \hat{y'}sen(wt)) \\ \hat{y} &= -\hat{x'}sen(wt) + \hat{y'}cos(wt) \end{aligned} \tag{2.16}$$

In the rotation frame, B_1 field 2.15 can be described as 2.17.

$$\vec{B_1} = B_1\hat{x} \tag{2.17}$$

According to equation 2.17, the oscillating field 2.15 is represented by a static field in the rotating frame, thus facilitating the magnetization vector description. The effective field felt by the magnetization after the application of the RF pulse is given by 2.18

$$\vec{B_{eff'}} = B_0\hat{z} - \frac{\Omega}{\gamma}\hat{z} + B_1\hat{x'} \tag{2.18}$$

and the equation for the magnetic moment evolution in the rotation frame is 2.19.

$$\frac{d\mu'}{dt} = -\vec{\mu'} \times \vec{B_1}\hat{x'} \tag{2.19}$$

Equation 2.19 shows the magnetic moment, and the magnetization vector performs a precessional motion around the RF field. When the RF field is applied, the spins rotate around the $\hat{x}'$ with a flip angle equal to 2.20

$$\theta = \gamma \int_0^{Tp} B_1(t)dt \tag{2.20}$$

where T_p is the pulse duration. If B_1 is a constant in T_p, then the flip angle can be written as 2.21.

$$\theta = \gamma B_1 T_p \tag{2.21}$$

2.1.4 NMR Signal and relaxation

After pulse duration T_p, the system suffers only the static magnetic field B_0 influence. The magnetization vector tends to return to the thermodynamic equilibrium and releases energy into the system. This process is called relaxation and involves two different mechanisms, namely transverse relaxation, and longitudinal relaxation, which occur at different rates. Both longitudinal and transverse relaxation times are intrinsic to each system or tissue, and Relaxometry techniques use them to infer the physical, chemical, and biological properties of a spin system.

Longitudinal relaxation, also called spin-lattice relaxation, represents the return of the longitudinal magnetization to the equilibrium situation aligned to the external magnetic field at time T_1. (13) During the return to the equilibrium situation, the spins release energy to the lattice interacting with the neighborhoods.

The longitudinal relaxation effects were phenomenologically introduced in Bloch equations, and the change rate for the interaction of nuclei is expressed by 2.22.

$$\frac{dM_z}{dt} = \frac{1}{T_1}(M_0 - M_z) \tag{2.22}$$

The return of the longitudinal magnetization to the equilibrium situation is illustrated in 5.
The longitudinal magnetization regrows according to equation 2.23

$$M_z = M_0[1 - e^{-t/T_1}] \tag{2.23}$$

The transverse relaxation, or 'spin-spin' relaxation, represents lost of phase coherence from the transversal magnetizations, i.e. return to zero in the transverse plane; it is caused by the influence of local magnetic field inhomogeneities and occurs at time T_2. (13)

An isochromat is a small volume of sample in which all spins must be conventionally processed at the same frequency. Once magnetization starts appearing in the transverse

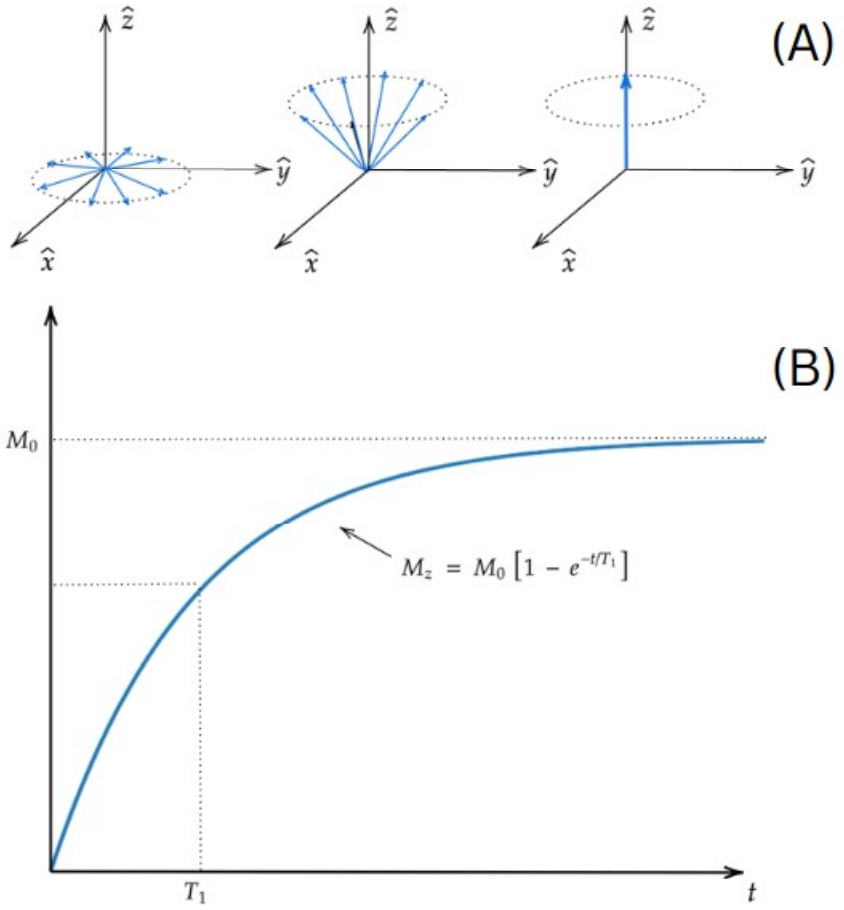

Figure 5 – A: net magnetization vector behavior during returning to the equilibrium situation in the longitudinal plane. B: longitudinal magnetization relaxation signal.

Source: By the author.

plane, it experiences local variations in the magnetic field, which locally alter the precession frequency. Consequently, each spin in an isochromat performs the precessional movement at narrowly different frequencies, quickly losing phase coherence in the transverse plane.

Local inhomogeneities in the static magnetic field (T_2' - decay) and stochastic and temporal interactions between the magnetic moments of the spins and their neighbors that change the local field (decay T_2) are two dependent processes that can influence the local fields and the transverse magnetization dephasing. Equation 2.24 expresses the real transverse decay time T_2*.

$$\frac{1}{T_2^*} = \frac{1}{T_2} + \frac{1}{T_2'} \tag{2.24}$$

Due to the stochastic nature of T_2, the contribution to dephasing cannot be recovered. However, the T_2' contribution the phase can be reversed by the application of a refocusing pulse. The differential equation for the transverse magnetization rate of change

2.26 receives the addition of empirical decay rate term T_{2^*}, determined by Bloch.

$$\frac{dM_{xy}}{dt} = \gamma \vec{M_{xy}} \times \vec{B} - \frac{1}{T_{2^*}} \vec{M_{xy}} \tag{2.25}$$

The loss of phase coherence in the transverse magnetization for the equilibrium situation is illustrated in 6.

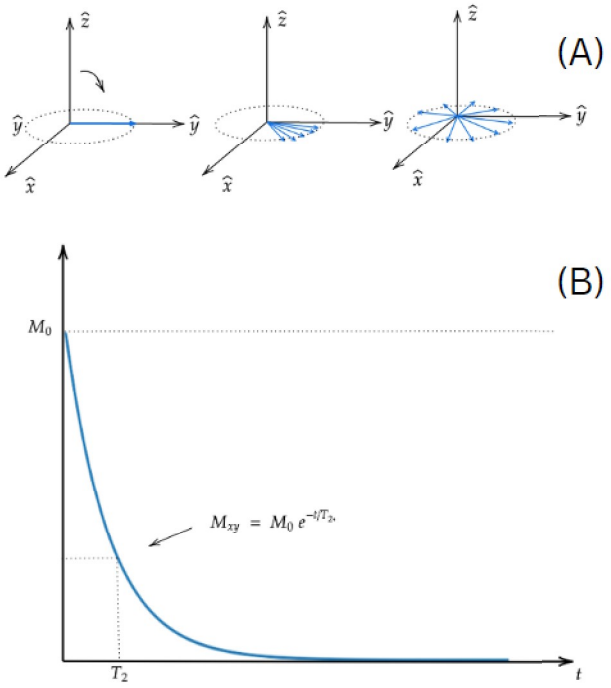

Figure 6 – A: net magnetization vector behavior when losing the phase coherence in the transverse plane. B: transverse magnetization relaxation signal.

Source: By the author.

The transverse magnetization decay according to equation 2.26

$$M_{xy} = M_0 e^{-t/T_{2^*}} \tag{2.26}$$

Transverse relaxation time T_2 is typically much shorter than longitudinal relaxation time T_1. The NMR signal persists as long as the magnetization transverse component is not null. The Larmor precession in the transverse plane induces a magnetic flux variation, which gives rise to an electromotive force that a receiver coil can detect. The NMR signal consists of the voltage (or current) detected by the receiver coil and is an exponential signal called Free Induction Decay (FID), as shown in figure 7.

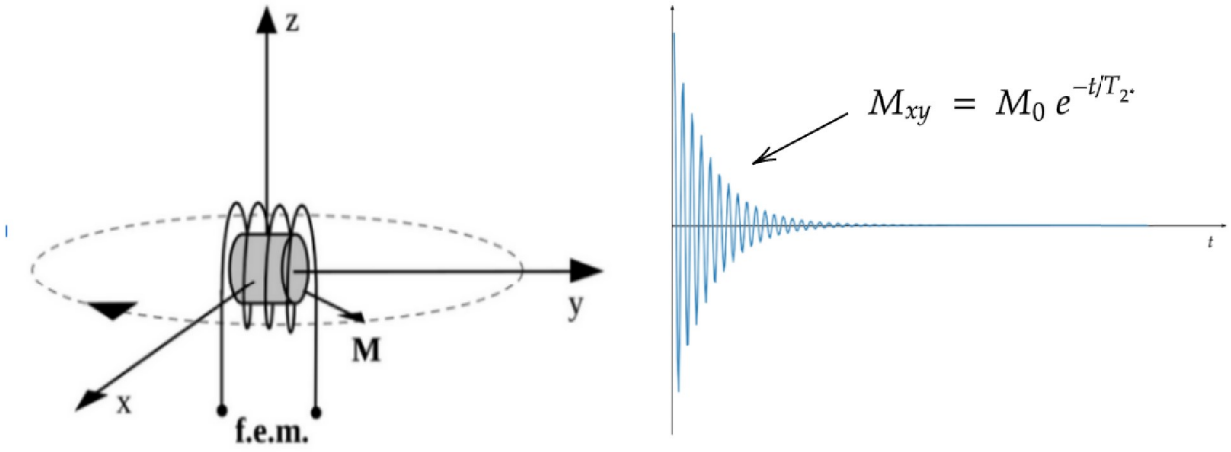

Figure 7 – A: receiver coil. B: Free Induction Decay (FID).

Source: Adapted from VIDOTO. (14)

Solving equations 2.22 and 2.26 leads to the following behaviors from the magnetization:

$$M_z(t) = M_z(0)e^{-t/T_1} + M_0(1 - e^{-t/T_1}) \tag{2.27}$$

$$M_x(t) = e^{-t/T_2}(M_x(0)cos(w_0t) + M_y(0)sen(w_0t)) \tag{2.28}$$

$$M_y(t) = e^{-t/T_2}(M_y(0)cos(w_0t) - M_x(0)sen(w_0t)) \tag{2.29}$$

The transverse magnetization composed of M_x and M_y can be expressed as a complex signal were:

$$M_{xy} = M_x + iM_y \tag{2.30}$$

Once the signal is complex, it can also be represented by 2.31

$$M(t) = |M_{xy}|\, e^{-i\phi} \tag{2.31}$$

where $\phi(t) = -w_0 + \phi(0)$ is the phase of the complex representation.

Once the FID represents the transverse magnetization decay in the time domain, the Fourier Transform can be applied to the complex signal obtaining the spectral representation in the frequency domain. Figure 8 displays a simplified representation of the

FID in the time domain and the Spectrum representation after the Fourier transform applications.

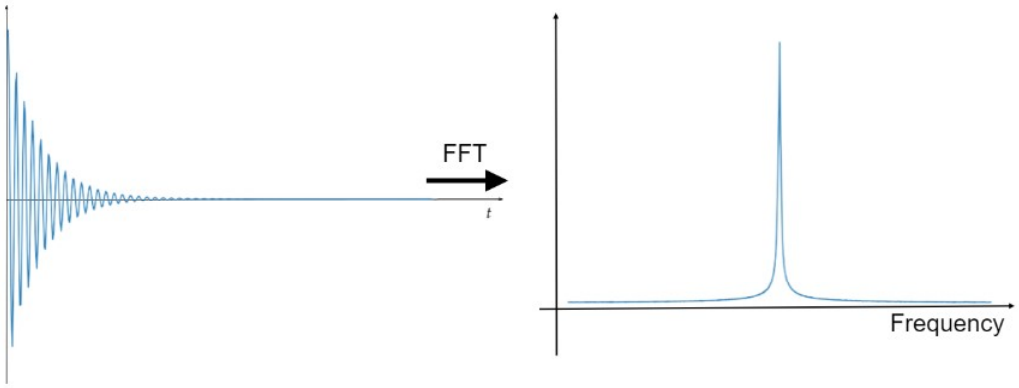

Figure 8 – FID representation in the time domains and the Spectrum representation after the application of the Fourier Transform.

Source: By the author.

2.1.5 Echo Formation

After the application of the excitation pulse, the magnetization is tipped into the transverse plane, and in this plane, the isochromat loses phase coherence and starts to accumulate phase and rotate in the transverse plane, decaying very fast. A π RF pulse, called refocusing pulse, may be applied to reverse the direction of the spin speed in the transverse plane, thus, at a time TE (echo time) the isochromats recovery phase coherence and the echo signal is formed. Figure 9 displays the steps to the echo formation. (15)

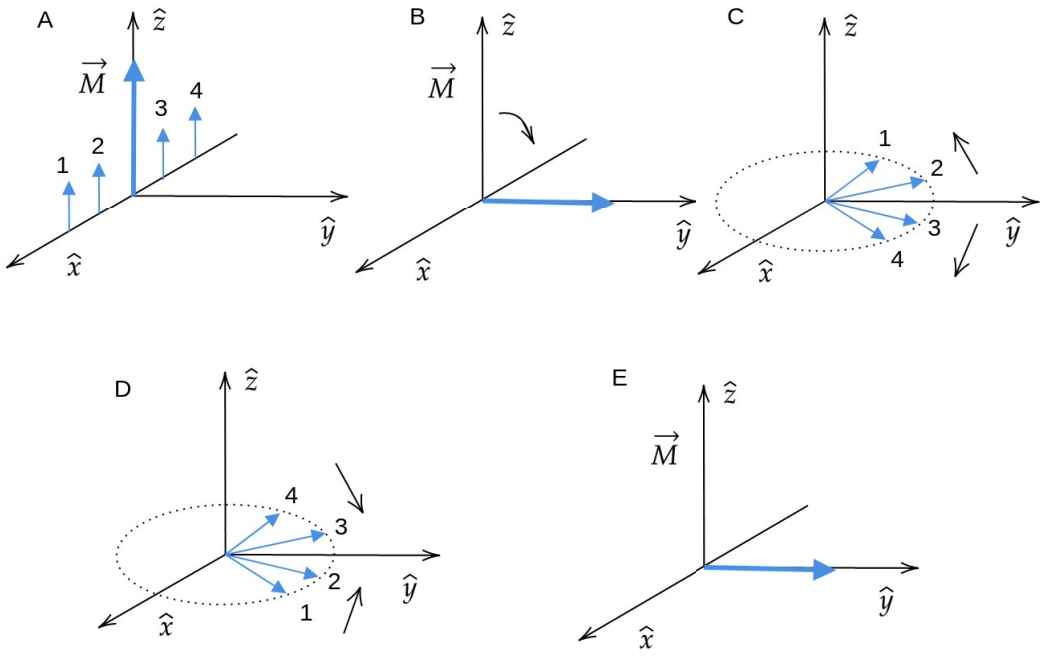

Figure 9 – Echo formation diagram. A: Four spins in a isochromat aligned to the external magnetic field in the equilibrium situation and the resulting magnetization. B: The magnetization flipped in the transverse plane after the applications of the excitation pulse. C: The spins in the isochromat start to lose phase coherence and rotates in the transverse plane. D: After the application of the refocusing pulse the spins rotation speed changes the directions and starts to accumulate phase again. D: The spins recovered phase coherence and the echo signal is acquired by a receiver coil.

Source: By the author.

2.1.6 Relaxometry techniques: T_2 measurements

Due to possible field inhomogeneities, discussed in section 2.1.4, transverse relaxation time T_2 cannot be obtained directly from the FID. In 1954, Carr and Purcell suggested acquiring several consecutive echoes by a $\frac{\pi}{2}$ excitation pulse followed by a sequence of π pulses. (16) The methodology enables the decay in the amplitude of the echo to be directly related to the transverse relaxation time.

The pulse sequence proposed by Carl and Purcell has led to another problem, i.e., a bad calibration in the π pulses, or even an inhomogeneity in the radiofrequency field, which may generate a cumulative deviation in the inversion angle. Figure 10 (A) shows the magnetization reversed by an incorrectly calibrated pi-pulse, and Figure 10 (B) displays the result of the accumulation of phases after a sequence of two pulses.

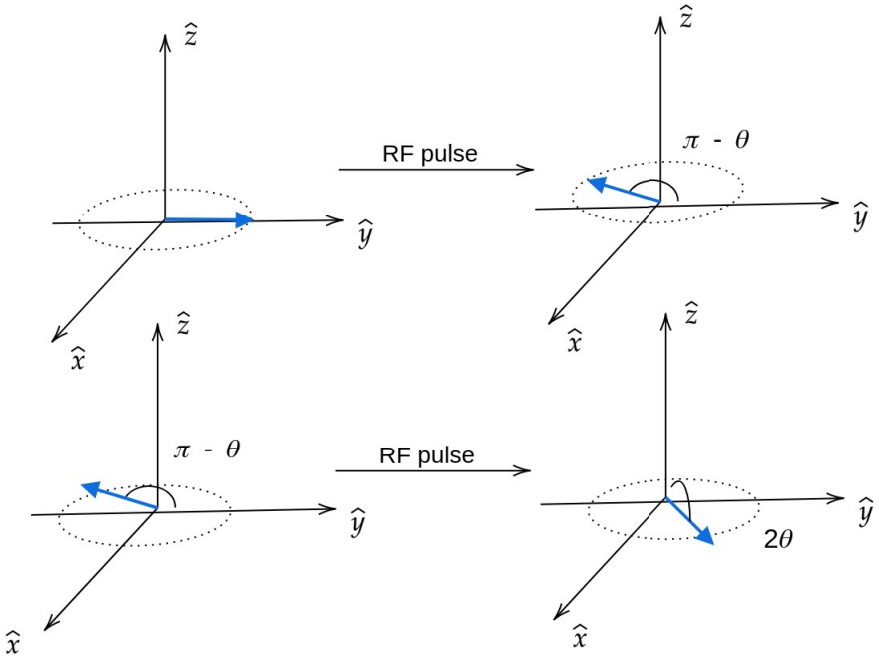

Figure 10 – A: the magnetization reversed by an incorrectly calibrated pi-pulse. B: a result of the accumulation of phases after a sequence of two incorrectly calibrated pi-pulses.

Source: Adapted from BROWN *et al.* (13)

Towards circumventing the aforementioned problem, in 1958, Meiboom and Gill introduced a 90° phase difference between the $\frac{\pi}{2}$ pulse and the π pulses. The Carr-Purcell-Meiboom-Gill (CPMG) sequence is a combination of the two techniques and obtains transverse relaxation time T_2 from the exponential decay of the echo signal amplitude acquired (17). Figure 11 shows a schematic of the CPMG sequence, where the time 2τ represents the echo time.

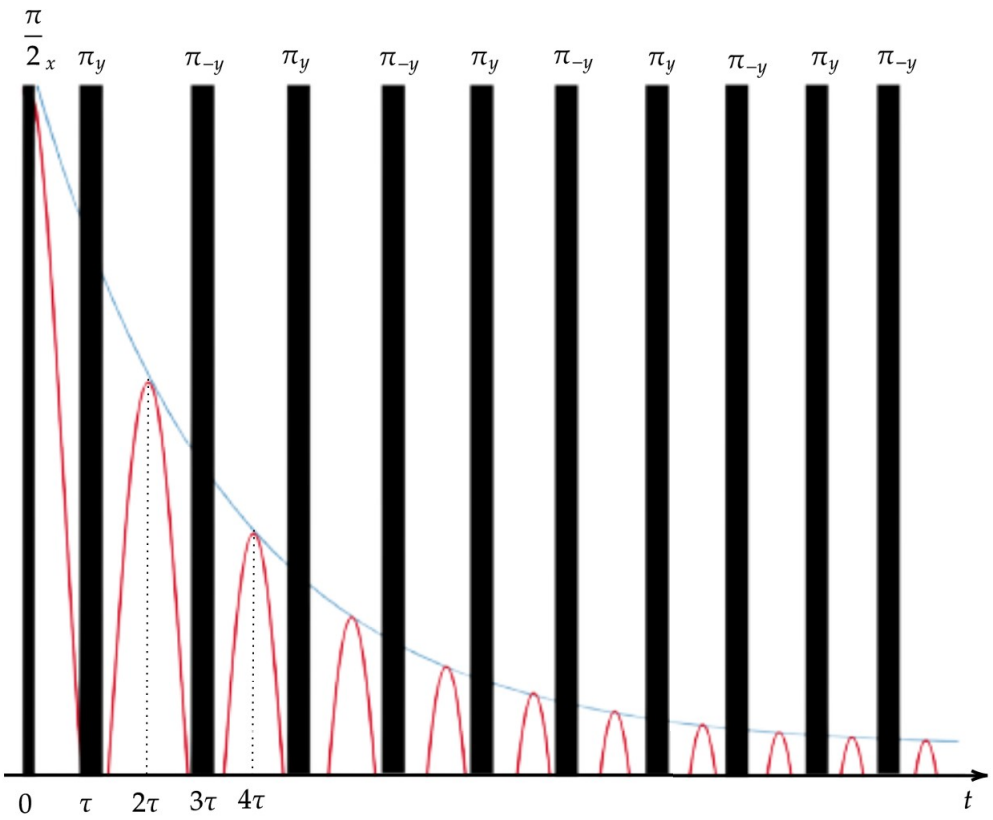

Figure 11 – Schematic drawing of the CPMG sequence. A signal is acquired after each pulse and its decay intensity is directly related to the transverse relaxation time.

Source: By the author.

Since a sequence of refocusing pulses is used, the decay occurs with T_2, as seen in figure 12.

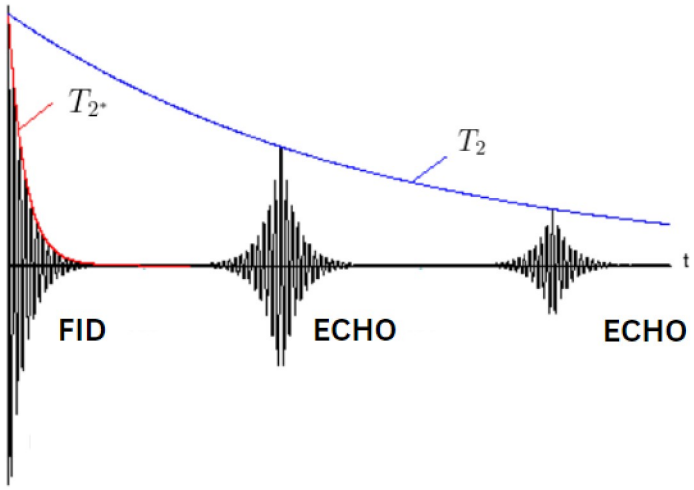

Figure 12 – A signal decay in the CPMG sequence.

Source: By the author.

2.1.7 Relaxometry techniques: T_1 measurements

The longitudinal relaxation time T_1 can be measured by the Inversion-Recovery (IR) sequence, which consists in applying a π pulse towards inverting the magnetization prior to the application of the $\frac{\pi}{2}$ pulse that flips the magnetization in the transverse plane. The transverse magnetization after the π pulse is the negative of equilibrium value 2.32.

$$M_z = -M_0 \tag{2.32}$$

The interval τ_I between pulses is increased with each acquisition; therefore, the longitudinal magnetization recovers a little more with each acquisition. Figure 13 shows a schematic of the Inversion Recovery sequence and the return of magnetization to the equilibrium value where TR is the repetition time between π pulses. TR must be long enough to ensure that the longitudinal magnetization returns to equilibrium before the next π pulse (i.e. $TR >> T_1$).

Figure 14 shows the module of the magnetization signal in the Inversion Recovery sequence. Time $\tau_{I,null}$ is the time at which the magnetization regrew from $-M_0$ to zero.

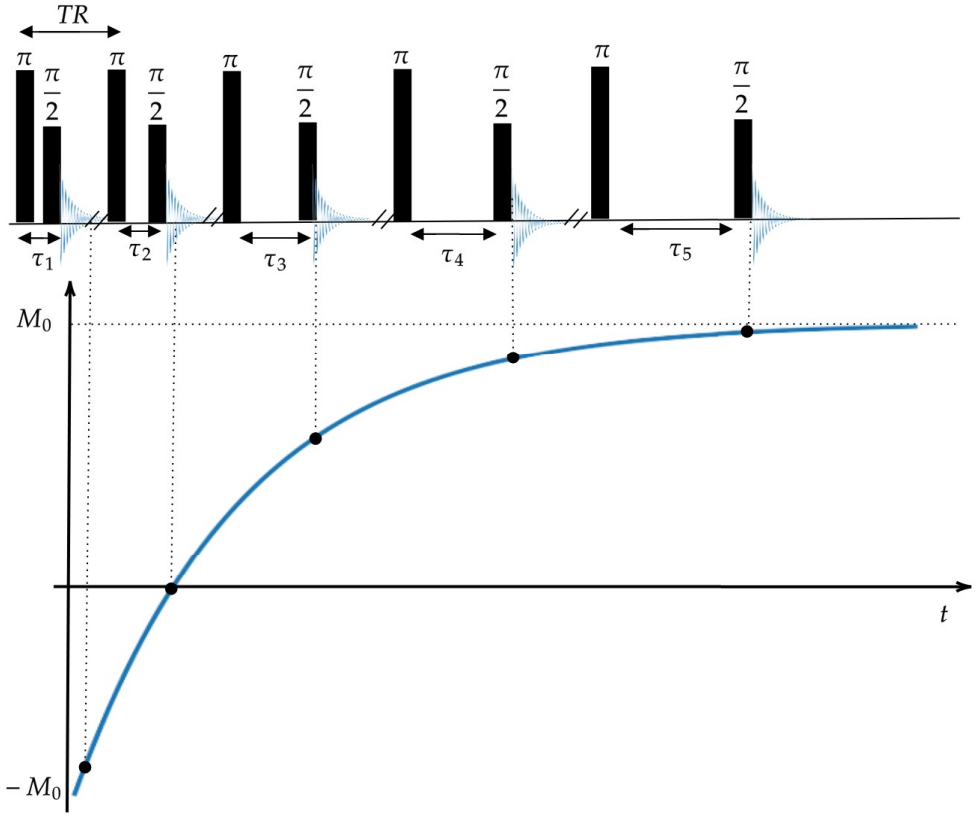

Figure 13 – Schematic drawing of the Inversion Recovery sequence. Variation in the signal magnitude as a function of the inversion time.

Source: By the author.

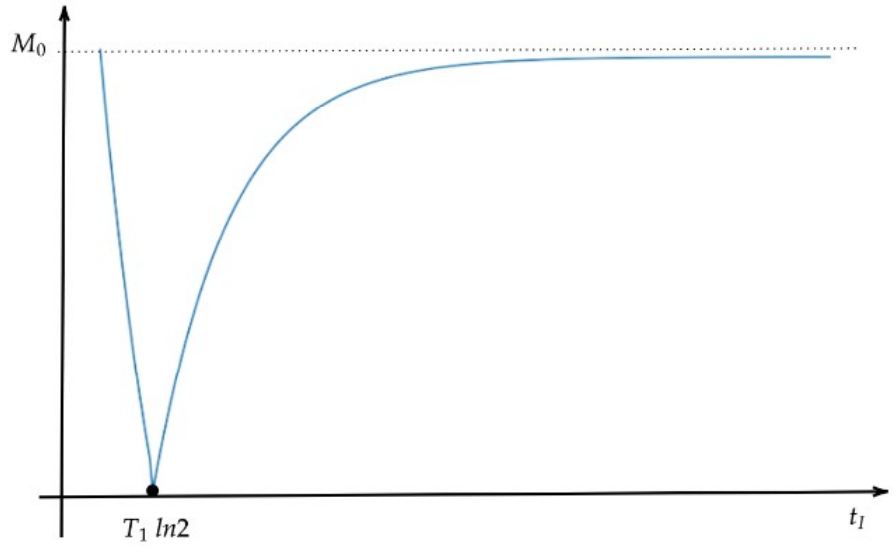

Figure 14 – Module of the signal in an Inversion Recovery as a function of the inversion time between the pulses.

Source: By the author.

Therefore, it is possible to determine T_1 using relationship 2.33.

$$\tau_{I,null} = T_1 ln2 \tag{2.33}$$

2.2 NMR Instrumentation

The NMR system can be divided into the following categories: (i) a magnet, (ii) a probe containing at least one coil, (iii) a Radiofrequency (RF) transmitter/receiver system for the generation of B1 field and NMR signal detection, and (iv) a computer system for instrument control and data processing. (18)

The components responsible for the system operation and control, e.g., those that generate the radiofrequency pulse or those that receive the signal and store data, constitute the spectrometer. Its different parts should work synchronously and the spectrometer can include subsystems such as power amplifiers, and a set of gradient coils.

An NMR experiment can be described as a combination of a pulse sequence that develops instructions for RF pulse generation and data acquisition for exemplifying the way the components are used. Instructions for the generation of RF are converted into analog signals (voltage) applied through the probe coil for the production of the RF magnetic field that excites the sample. Such communication is established through the transmitter system. The voltage induced in the sample coil is detected by the receiver system and the signal is amplified and then converted from analog to digital. The digital signal is then stored in the computer system that processes the data. Figure 15 shows a schematic of the NMR system's main components.

The magnet handles the generation of the static magnetic field B_0. Most NMR experiments rely on the static field to be homogeneous since a lack of homogeneity can cause a loss of sensitivity and resolution. Three types of magnets, namely superconductive, resistive, and permanent can be employed in NMR applications.

In a superconductive magnet, the magnetic field is generated by the passage of current through a superconductor material. Superconductivity is a phenomenon in which materials such as mercury or a niobium-titanium metal alloy show zero electrical resistance at low temperatures. While the temperature remains below the critical one (T_c), the current can flow through the wire without dissipating heat and the magnetic field is maintained. The wire must be immersed in liquid helium for maintaining the superconductor state making cooling costs high.

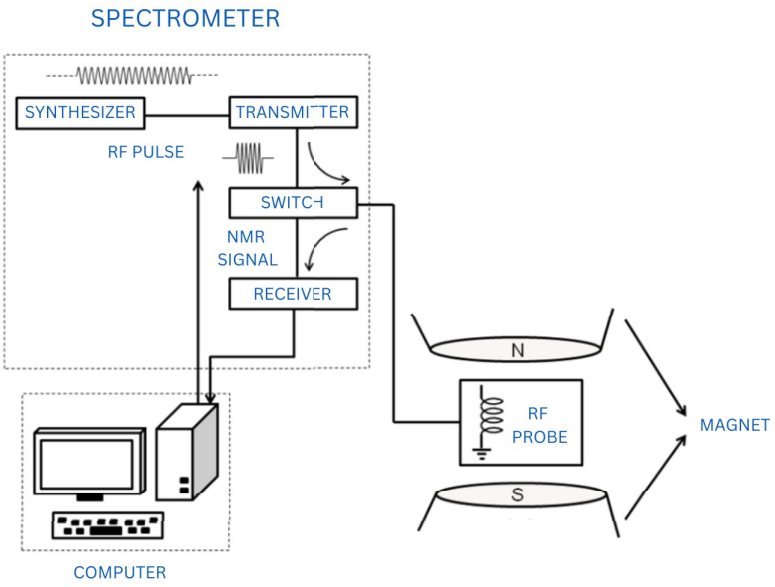

Figure 15 – Schematic of the NMR equipment.

Source: Adapted from MONTRAZI. (19)

Magnetic fields from resistive magnets are also formed by a current passage through a wire. However, since the material is not in a superconductive state immersed in liquid helium, the wire has resistance and dissipates heat for the system. As a result, such a magnet requires a cooling system and demands high, constantly applied power. As an example, according to (20), the power required for a 0.15T resistive magnet is approximately 50kW.

Permanent magnets are composed of blocks of magnetic materials, of which Neodymium-Iron-Boron (NdFeB) is one of the most common. The design aims at raising a maximum magnetic flux density between the magnetic poles using only a short portion of permanent material. Therefore, permanent magnets are constructed in different shapes and are more sensitive to temperature fluctuation, for illustration purposes, a NdFeB magnetNdFeB has a negative variation coefficient of remanence as a function of temperature, i.e. about $-0.1\%/°C$ (21), so that as the temperature increases the field strength decreases and vice versa.

3 DIGITAL MAGNETIC RESONANCE SPECTROMETER (DMRS)

Nuclear Magnetic Resonance can be widely applied in medicine, agriculture, materials science, engineering, and chemistry, among other areas. However, according to the Center for Images and Magnetic Resonance Spectroscopy (CIERMag) (9), the existing NMR commercial equipment is usually applied exclusively to either Spectroscopy, Relaxometry, or Imaging. Despite its practicality, it is restricted to drastic modifications and enables no new pulse sequences or new waveforms to be introduced in a simple way, and its updates are costly and challenging.

Towards circumventing the aforementioned issues, CIERMag has developed the Digital Magnetic Resonance Spectrometer (DMRS) based on reconfigurable hardware powered by a Field-Programmable Gate Array (FPGA) and ancillary software for performing and controlling NMR experiments. Python Magnetic Resonance (PyMR), the framework behind the software, aims at filling the gaps left by existing NMR tools, discussed in 1. The new CIERMag spectrometer is looking for a flexible and reusable system that provides an easy-to-use interface

DMRS is a magnetic resonance system of high flexibility in both software and hardware. The next section describes the hardware and software of the Digital Magnetic Resonance Spectrometer and provides an overview of its benefits to NMR research.

3.1 CIERMag Hardware

As discussed in section 2.2, the spectrometer is part of the NMR system that generates the signals used in nuclei excitation, signal acquisition, and digitization The systems also accomplishes magnet and additional systems such as gradient coils, and all spectrometer synchronous subsystems.

The commercial spectrometer components are composed mostly of logic circuits and hardware specific for fast switching. The processor's architecture is generally rigid and comprised of non-flexible structures. Over time, such systems become obsolete and face problems such as a lack of maintenance assistance and insufficient flexibility and adaptability.

The current electronic equipment uses integrated circuits for its operation and, in most devices (e.g., televisions, cellular telephones, radios, etc.), such circuits are pre-programmed by their manufacturers for performing certain functions. The Field-Programmable Gate Array (FPGA) emerged in the 1980s as a device to be programmed

according to user applications (reconfigurable hardware). It is an integrated circuit that enables the implementation of digital circuits with the use of the description of its internal connections in Very-High-Speed Integrated Circuit (VHSIC) Hardware Description Language (VHDL).

The DMRS comprises a Field-Programmable Gate Array (FPGA) with synthesized digital hardware subsystems that generate and acquire NMR signals. (22) The FPGA approach used in the DMRS solves the problem of inadaptability faced by commercial spectrometers, providing a flexible system controller and a highly malleable arrangement of digital and analog hardware once the firmware developed by VHDL can be adapted to new FPGA generations without major and extensive modifications. The DMRS uses a holistic approach according to which a set of essential subsystems for any configuration, i.e., from a Relaxometry spectrometer to a more complicated MRI scanner setup, can be synthesized in a single FPGA. All hardware subsystems reside in the single FPGA and keep the whole hardware project integrated and parameterized. The new system is scalable and replicable since it integrates subsystems by instantiating them in the code and reconfiguring the inputs and outputs. It enables, for example, the introduction of new transmitters/receivers limited only by the FPGA model resources. The approach adopted by CIERMag in the DMRS avoids obsolescence since its hardware system is almost independent of FPGA model, family, and vendor.

The DMRS was configured towards containing a loader, a temporal controller, an RF pulse generator, a module for RF modulations, and synchronous receivers. Analog-to-Digital converter (ADC), Digital-to-Analog converter (DAC), and Ethernet/USB media converters are also incorporated in the system. In what follows is a brief description of each of such components.

- **Loader:** responsible for receiver commands (through USB or Ethernet), it loads commands and programs, reads values, and verifies the functionality of the spectrometer pointing to possible errors.

- **Timing Sequencer (TS):** a temporal controller that enables the generation of real-time signals and a module processor that performs groups of instructions and understands loops and interruptions in the code.

- **RF pulse generator:** a module that generates digital signals from an RF signal and enables a dynamic change of parameters such as base frequency, flip angle, and offset and signal modulations in AM/PM/FM.

- **RF modulator:** an arbitrary waveform generator with AM/PM/FM modulation.

- **Synchronous receivers:** modules responsible for the synchronous demodulation of captured signals, which pass a combined Cascaded Integrator Comb (CIC) filter and a Finite Impulse Response (FIR) filter before being stored.

The spectrometer can also encompass an arbitrary waveform generator for the creation of a magnetic field gradient for applications that require a magnetic field with spatial variation, as in the imaging case. In MR applications, such gradients are called logical gradients: Read, Phase, and Selection Gradients, for example. The spectrometer includes a rotation module that converts logical gradients into physical gradients and communication modules for USB and Ethernet ports. The spectrometer also contains auxiliary modules configured according to the user's application to be used for simpler applications such as the implementation of an FPGA temperature controller, as well as for more complex functionalities (e.g., channels that correct the homogeneity of the main magnetic field (Shimming)). Figure 16 shows a schematic of DMRS and its peripherals.

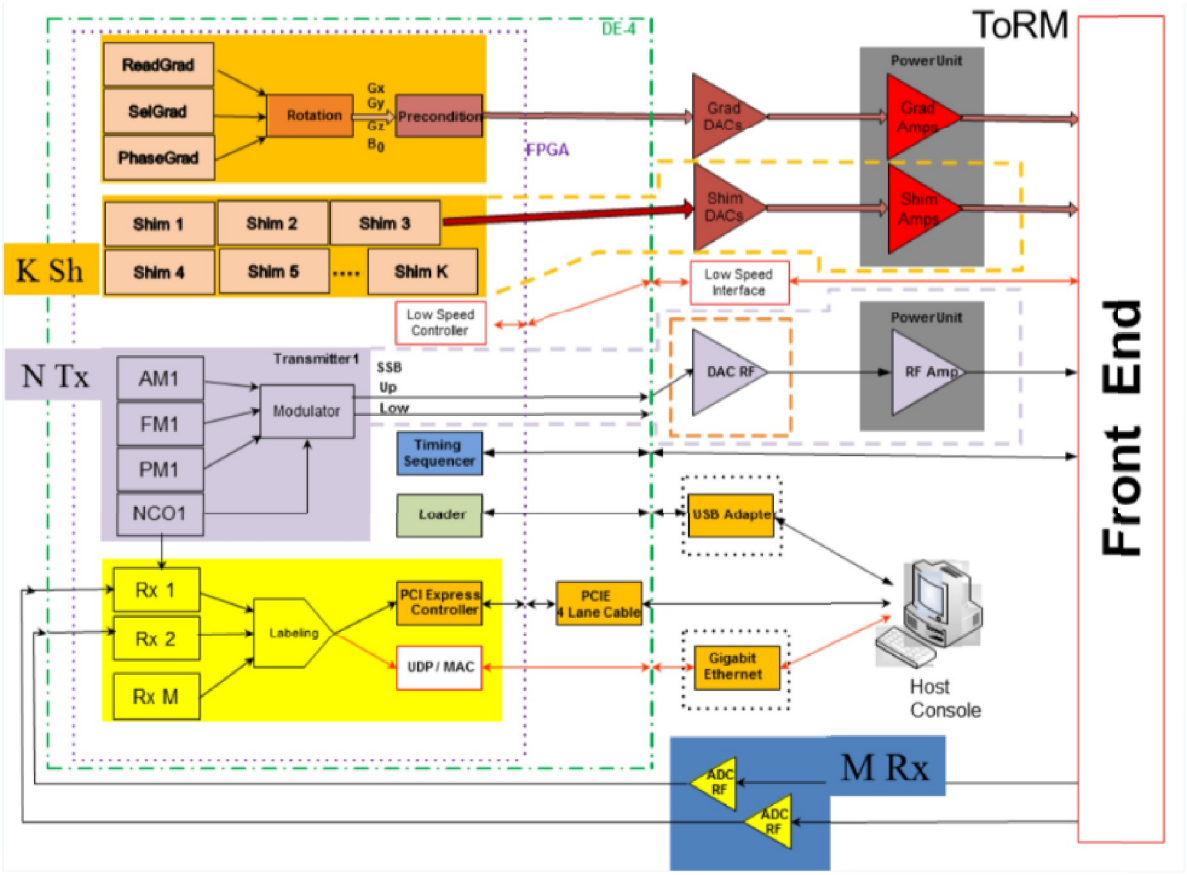

Figure 16 – Block diagram of DMRS and its peripherals. The green dashed contour represents the limit of the FPGA Development Kit and the other components responsible for the conditioning of analog and digital signals (generation of both gradients and RF pulses and digitization of MR signals) connected to the experiment through a Front End.

Source: CIERMag.(23)

The current version of the DMRS combines 8 receiver channels (ADCs), 4 transmitter channels (DACs), and 4 gradient channels (DACs) and communications are established through either Gigabit Ethernet or PCI Express. The synthesized hardware contains approximately 150 thousand lines of code, 32 heterogeneous processors under the "F" compiler, and 20ns of clock. Figure 17(A) illustrates an old version of the FPGAs Mezzanine boards, containing two ADCs and two DACs each. A modern version of the system has independent Mezzanine boards for receive and transmit stages as shown in 17(B).

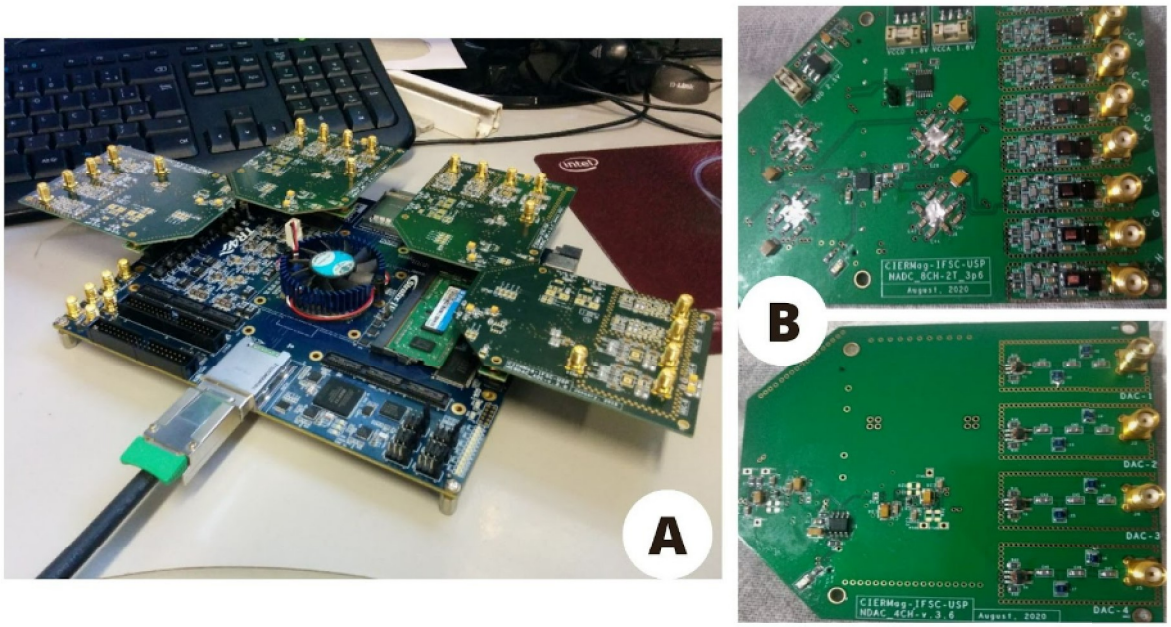

Figure 17 – FPGA system with ADC/DAC boards.

Source: CIERMag. (23)

3.2 CIERMag Software

In addition to the spectrometer hardware, CIERMag has also developed software with the use of PyMR framework that performs and controls NMR experiments using the DMRS. The software contains an Integrated Development Environment (IDE), F-Language with its own compiler and a Language Server Protocol (LSP), a Console, and an ACQ Server. The language server enables the use of two different IDE's, namely Spyder and Eclipse, for the edition of codes in F language with syntax highlighting, autocomplete code, variable examination etc. PyMR promoted the organization of parameters into three different levels, of which the first is composed of two specifications, namely Method and System.

In the CIERMag context, a Method represents an experiment project with all information on the experiment, from pre-processing to post-processing scripts, including pulse sequence, sequence-specific parameters definition, validation scripts, and acquisition, execution, and processing protocols. It also comprehends the graphical representation of parameter edition boxes, lists, tables, Booleans etc. Therefore, developing a new method compatible with the DMRS means developing all scripts necessary for validation, creation of tables, data processing, and the pulse sequence written in F Language. Figure 18 displays the folder structure with the Method files.

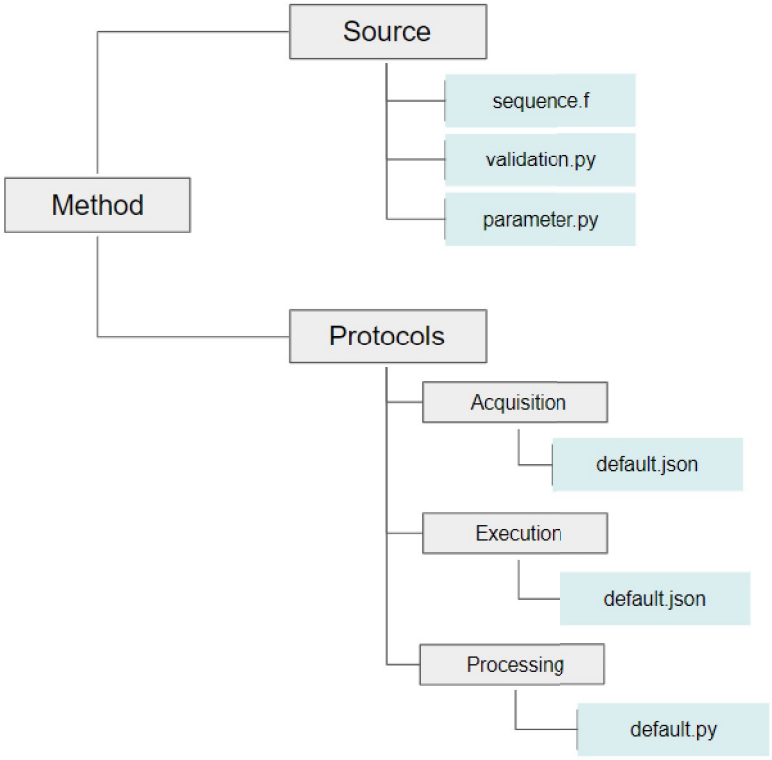

Figure 18 – Method folder and files structure. Grey structures represent folders and green structures denote files.

Source: CIERMag. (23)

Sequence file "sequence.f" - the sequence file contains the NMR sequence code written in F- Language.

Acquisition protocol – a JSON file that stores all necessary information to be used in the acquisition system. The acquired data are saved in a Hierarchical Data Format (HDF5) file structure and HDF5 is a high-performance data software library and file format that manages, processes, and stores data in a file called "final_data.h5"

Execution protocol - a JSON file that contains the system and all Method parameters.

Processing protocol/Script - a python script that manages and obtains information from the dataset and uses PyMR data and protocol managers to access and read data, parameter values, and files from the execution protocol. Managers are a PyMR resource that provides functions for the reading of such data and parameter values. The protocol manager enables access to execution and acquisition protocols and the data manager helps the reading of data obtained in the experiment.

Validation script – a Python script in which several tests check if the designer created sequence parameters according to both equipment and sequence restrictions and if they respect their internal relationship. It also updates waveforms to be used in the Method and verifies if the delays were correctly calculated by the programmer with duration multiples of 20 ns. A template for the validation script is generated when the method is created by PyMR; however, the sequence designer adapts the template so that it corresponds to the current Method, i.e., the designer verifies if all shapes used in the sequence were loaded and all delays are being verified, etc.

Parameter file "parameters.py" – a Python file that contains all the parameters created as python objects and their attributes.

Secondary files (e.g., table files for specific applications such as store pulse shapes and gradient amplitudes) can also be created. Figure 19 shows a diagram representing all software components of the system. PyMR Framework provides the main elements for all software programs developed in python using PyQT to develop the Graphical User Interface (GUI).

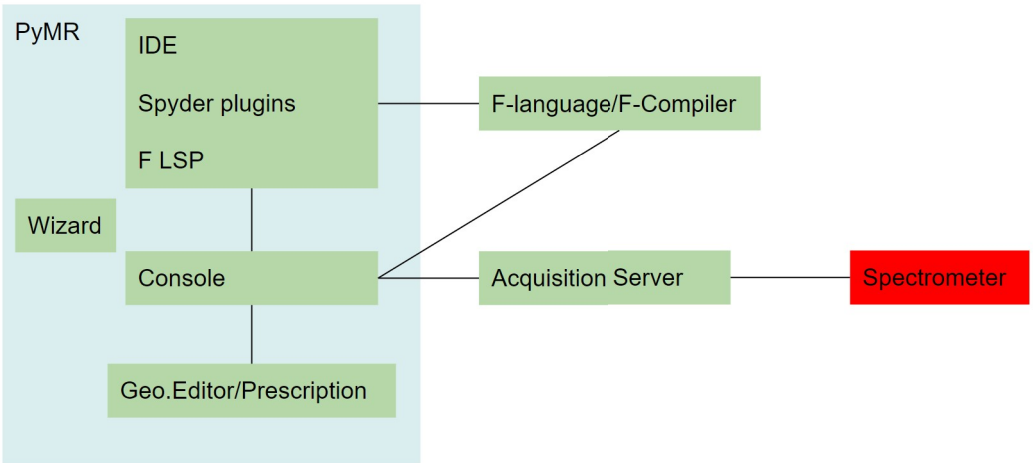

Figure 19 – Diagram of software components from CIERMag spectrometer.

Source: CIERMag. (23)

The next sections describe the software entities developed by CIERMag and their interrelationship.

3.2.1 Installation Wizard

PyMR installation wizard is an integrated tool that installs, updates, and reports data. It communicates with the GitLab platform, clones all CIERMag repositories, installs CIERMag software (see Figure 20), and also creates shortcuts for CIERMag applications.

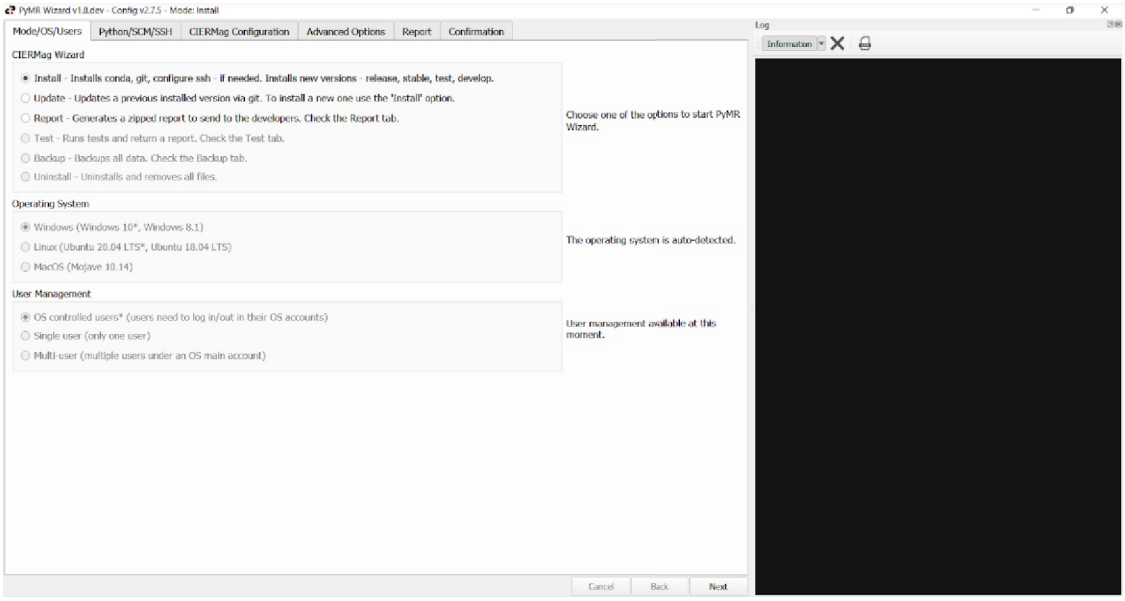

Figure 20 – PyMR installation wizard.

Source: CIERMag. (23)

Moreover, it installs Conda Python, creates environments, installs Git, clone packages, and installs them.

3.2.2 F Language and Language Server Protocol

F language was developed for programming pulse sequences used in MR experiments and its several features facilitate the development of new techniques. It has its own compiler, which generates codes for the specialized processors in FPGA, and a Language Server Protocol (LSP).

A sequence written in F language can be split into three parts, namely modules, events, and command sequence. Events are the smallest part of a sequence and represent

an event in the NMR sequence. Modules control the flow of the sequence program, contain statements to repeat instructions, perform conditional processing, and start events and delays. The sequence command represents the part of the program that selects a single module as the entry point and sets the parameters used by it. Figure 21 shows a sequence code written in F language.

```
module CPMG(
        NumAver,
        NumEchoes,
        tRead,
        tExcEvent,
        tRefEvent,
        TE,
        TR
        ):

    start GlobalReset1();
    loop Average = 0, NumAver-1, type = average:
        start GlobalReset1();
        start ExcPulse(tExcEvent, RT_ExcAMAmp[0], TX1Phase1[0], RT_ExcOffset[0]);
        delay  DL1 = TE/2 - (tExcEvent/2 - tRefEvent/2);
        loop Echo = 0, NumEchoes-1, type = echo:
            start RefPulse( tRefEvent, RT_RefAMAmp[0], TX1Phase2[0], RT_RefOffset[0] );
            delay DL3 = TE/2 - (tRefEvent/2 + tRead/2);
            start Readout(tRead, RT_ExcOffset[0], RX1Phase[0]);
            delay DL3;
        end //echo
        delay DLTR = TR - (unit(NumEchoes)*TE + TE/2 + tExcEvent/2 + tRefEvent/2);
    end // Average
end //cpmg
sequence :
    #if C_CPMG == true:
        start CPMG(
            NumAver = C_Averages,
            NumEchoes = C_NumEchoes_FIX,
            tRead = C_tRead,
            tExcEvent = C_tExcEvent,
            tRefEvent = C_tRefEvent,
            TE = C_TE,
            TR = C_TR
        );
    #end
end // Sequence
```

Figure 21 – Part of the CPMG sequence code written in F Language.

Source: By the author.

The central idea of the language focuses on the concept of tables that can be used in different applications, i.e., to define pulse shapes, gradient segments, etc. The language also allows the use of real-time variables that give access to changing variable values in scan time. Loops are another important structure in F language and can be used when parts of the code need to be repeated. Loops (typed) are a very useful tool since it allows the sequence designer to select the format of the data structure. F language has several loop types and its nesting order is almost free and the acquired data is automatically

organized based on loop types. The A brings a detailed F language reference guide, with examples and applications.

3.2.3 PyMR IDE

PyMR Integrated Development Environment (IDE) (9) is dedicated to the creation and development of new NMR Methods. Since in the CIERMag context each NMR experiment is seen as a software project, in PyMR IDE, the project folders organization is through a project explorer. Moreover, it enables access to libraries and all files from the Method, including tables, graphical edition, and the creation of parameters with the aid of an easy-to-use interface. PyMR IDE also contains a log window for debugging codes, running validation scripts, and compiling sequences. Figure 22 shows PyMR IDE with a Method opened.

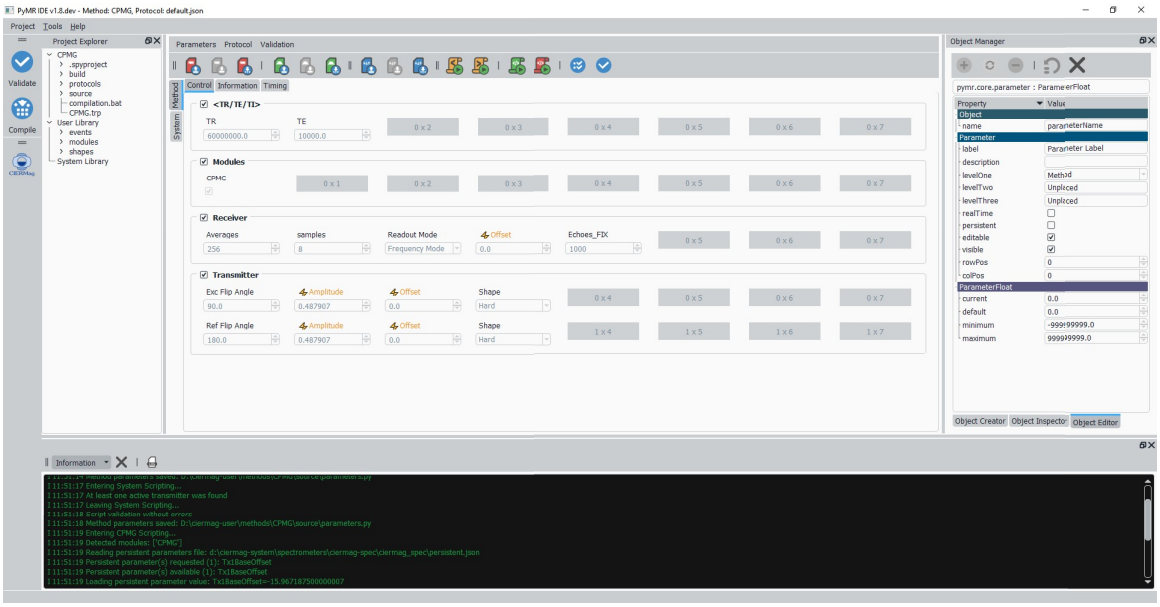

Figure 22 – PyMR Integrated Development Environment (IDE).

Source: CIERMag. (23)

3.2.4 PyMR - Spyder IDE Plugin

Spyder IDE plugin enables the use of Spyder resources for syntax highlighting and other coding tools for Python and F language codes and provides integrated tools for debugging, inspecting, and processing data. It has been integrated with PyMR IDE, thus containing all PyMR IDE functionalities. Figure 23 shows the Spyder PyMR interface with an F language code.

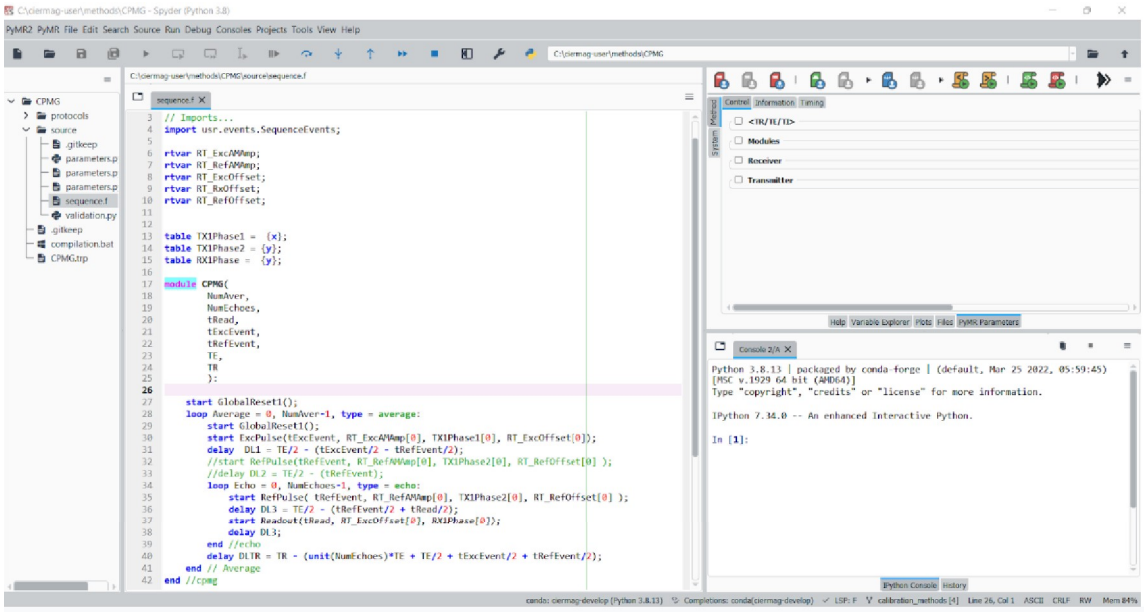

Figure 23 – Spyder IDE Plugin.

Source: CIERMag. (23)

3.2.5 Console

The Console was developed for the acquisition, organization, visualization, and storage of data and supports the running of sequences, acquisition, and visualization of the NMR signal (10). The Console is the application thought the user would perform the experiment, using the console it is also possible to change the value of parameters and method protocol. After acquiring data, it runs automatic processing scripts, manipulates the data, and analyzes experiments. Figura 24 displays its window.

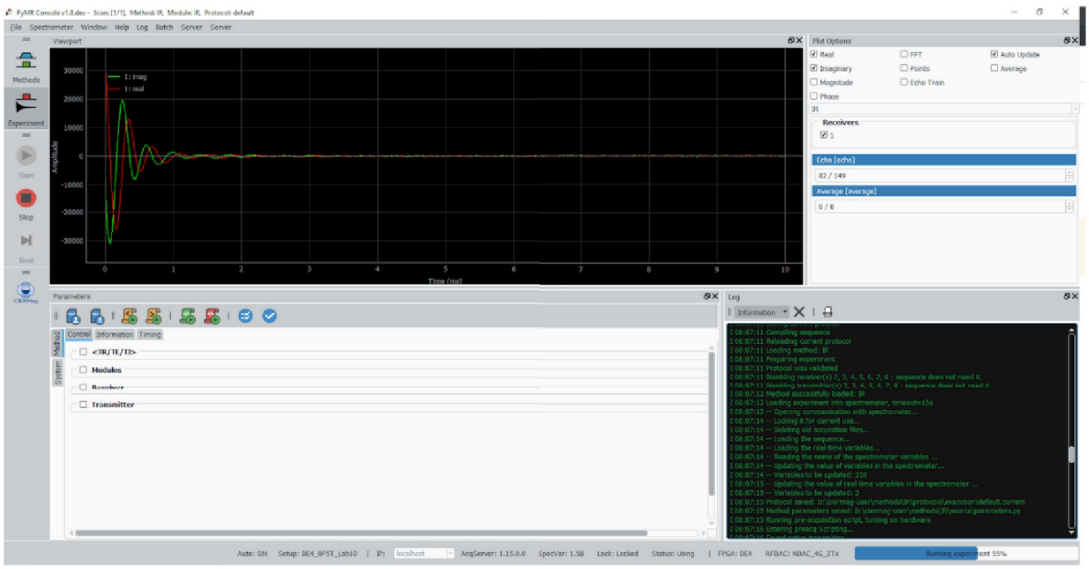

Figure 24 – Console interface.

Source: CIERMag. (23)

The console enables the use of a processing script for the creation of plot windows for analyses of data after their processing by Qt Widgets. Figure 25 shows a window plotted by the Console through the processing script.

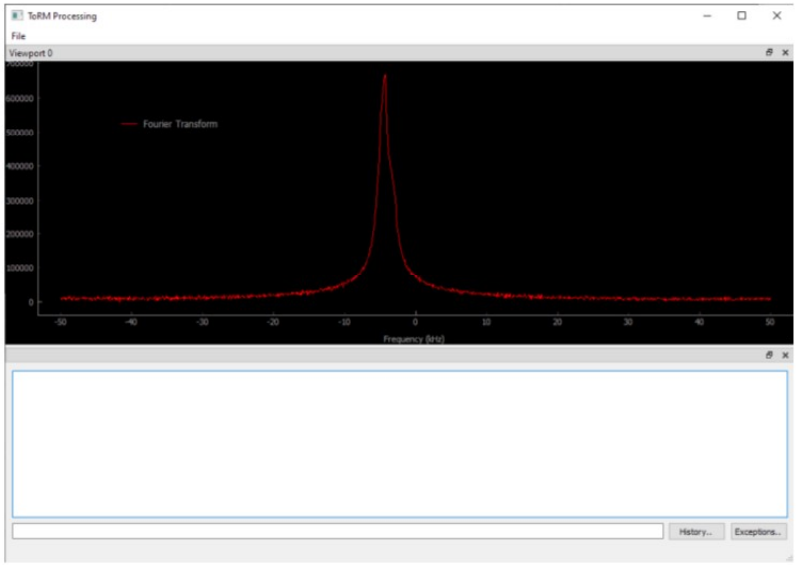

Figure 25 – Console plot window.

Source: CIERMag. (23)

3.2.6 Acquisition Server

ACQ-Server is a stand-alone application that communicates with the Spectrometer/FPGA and controls the spectrometer and the temporary data storage in the acquisition process. Figure 26 displays the ACQ Server interface.

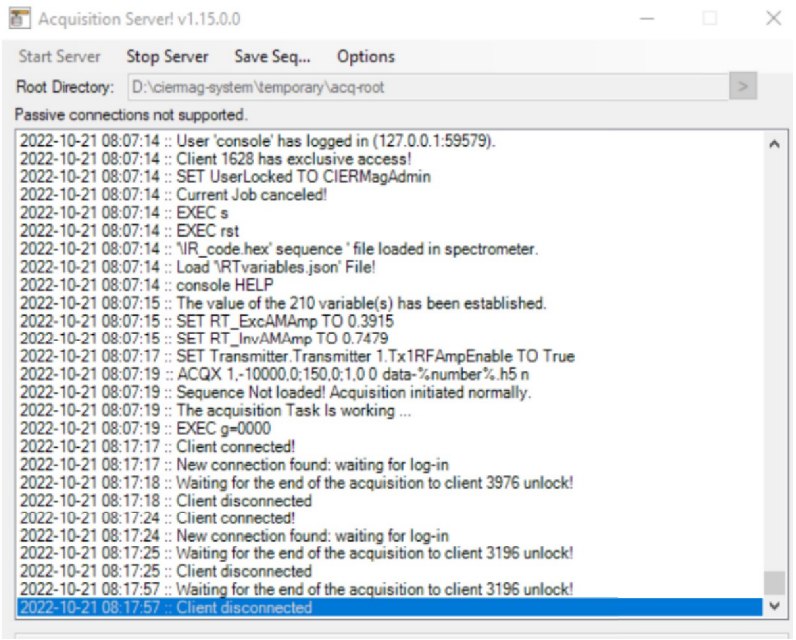

Figure 26 – ACQ-Server interface.

Source: CIERMag. (23)

3.3 How to create a new DMRS Method

An NMR Method in the DMRS context represents an experiment Project that contains the following files: sequence.f, acquisition, execution, and processing protocols, validation script, and parameter file. The diagram in Figure 27 represents the steps to be followed in the Method creation.

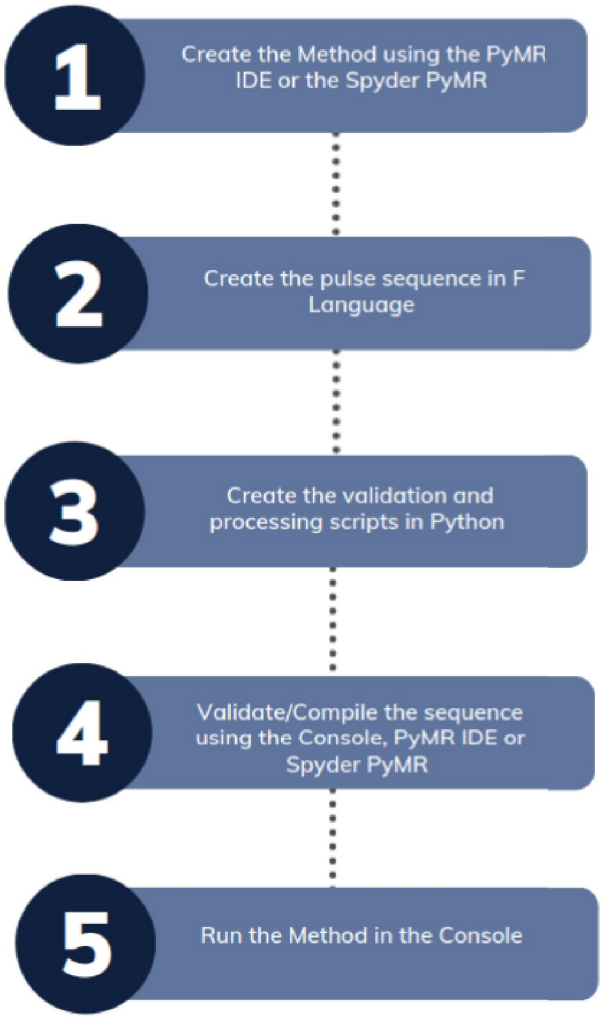

Figure 27 – Diagram with the steps for the creation of an NMR Method in DMRS.

Source: By the author.

Before the design of the Method itself, it is considered a good practice to draw all possible graphical representations of its Pulse Sequence. There, the user should detail (horizontally) the lines of the DMRS that will hold desired actions, such as RF Transmit channels, logical gradients, phase advances, phase resets and phase coherence copy, delays, loops, acquisition intervals, anyway, all possible information necessary to tell the DMRS what, how and when to do. With practice, the Method designer will end up holding this stage of Method construction as essential to readily building the F code and proceeding with a clean compilation. The team of CIERMag developed all basic tools to help with this part of the design, in the form of graphical building blocks constructed in Microsoft Visio 2019 (24), and several Methods were designed using its help. The Visio tool for sequence illustration is described in the appendix A. This feature will be later incorporated into the software package suite, and made available to all users. Translating the graphical building

blocks into F code is natural and straightforward since the F language itself was thought to bring a graphical interpretation of the sequence.

The first step for the creation of a new Method with the use of CIERMag's structure is to create a project in either PyMR IDE, or Spyder PyMR. After clicking on "New", a window is displayed so that information on the method is filled. Category, project name, and author fields must be provided by the sequence author. The full path field is also a required parameter that is automatically filled by the system. A short description and a description are two optional fields in the window, as is shown in Figure 28.

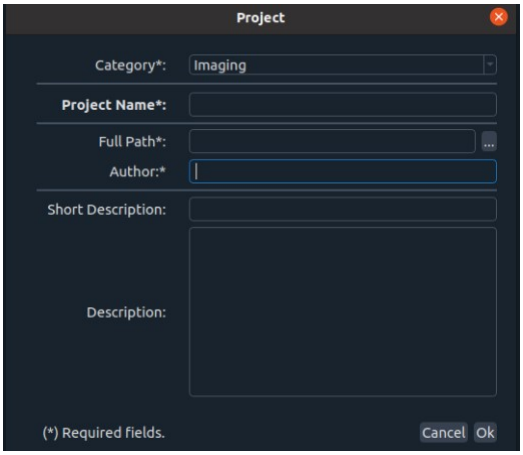

Figure 28 – New method creation window in PyMR.

Source: CIERMag. (23)

The creation of the project follows one of the three templates that generate the necessary folder structure and the initial files as pre-encoded structures. The user should choose among Imaging, Spectroscopy (Analytical), and Relaxometry Methods templates. The parameters required for the Method development are also created and manipulated in PyMR IDE.

The second step is to develop a pulse sequence in F Language A provides a detailed F language reference guide, with examples and applications, and the third one is to generate the validation and processing scripts in Python. Since those are python scripts, all python libraries such as Numpy (25), Scipy (26), and NMR Glue (27) - used to process NMR data - can be used in the files. Moreover, ciermag_spec parameters that access values from the execution protocol and function libraries containing special functions developed for a special use in CIERMag can be imported towards the use of the persistent protocol. A persistent attribute is a parameter Boolean attribute when using the functions to read and write persistent, the current values from the persistent parameters will be automatically saved in the system protocol and loaded also automatically. It enables the value of a

parameter to permeate between subsequent scans, either for the same or for different sequential Methods.

Finally, the Method must be validated and compiled. PyMR IDE, console, and Spyder PyMR have buttons for those actions. When the validate is clicked on, the validate script runs and verifies all tests in the script – error, warning, or info messages are displayed in the logger window in case of an error, warning or info is selected to call the attention of the user. After validation, the execution protocol is updated with the new values. The compile command compiles the sequence and checks if the syntax is correct. The last step is in the Method in the console, where it is possible to control the experiment and visualize the data in scan time. After the experiment finishes, the console runs the processing script, which may contain commands to return values for the console through the logger window, or using the persistent protocol, it is also possible to include commands to open plot windows to show the processing data.

4 MATERIALS AND METHODS

4.1 Materials and infrastructure

All procedures were performed at the Instituto de Física de São Carlos (IFSC) from Universidade de São Paulo (USP), by a Digital Magnetic Resonance Spectrometer (DMRS) developed by Centro de Imagens e Espectroscopia por Ressonância Magnética (CIERMag), (see chapter 3 for its description. The NMR Methods in CIERMag context used in the experiments are described in section 4.2.

The experiments were conducted in a Neodymium-Iron-Boron (NdFeB) kept in a box for temperature control. The solenoidal RF coil has an optimal operating region of 15 mm diameter by 15 mm thickness with higher than 96% homogeneity for a region of 12 mm diameter by 14 mm length. The magnet field intensity is 0.5463 T and operates at 23.256 MHz for (^1H) nuclei. A TOMCO amplifier model BT00500-Gamma was used with a homemade RF probe developed by Dr. Edson Vidoto for use for the permanent magnet with a solenoid coil of 28 mm diameter and 40 mm length is formed by 14 equally spaced turns of enameled copper wire. Figures 29 A, B, and C show an inner vision of the probe, its external view, and the probe inserted into the magnet with a sample holder. The DMRS Console performed the experiment control and data acquisition. Figure 30 displays the magnet in a thermally insulated box.

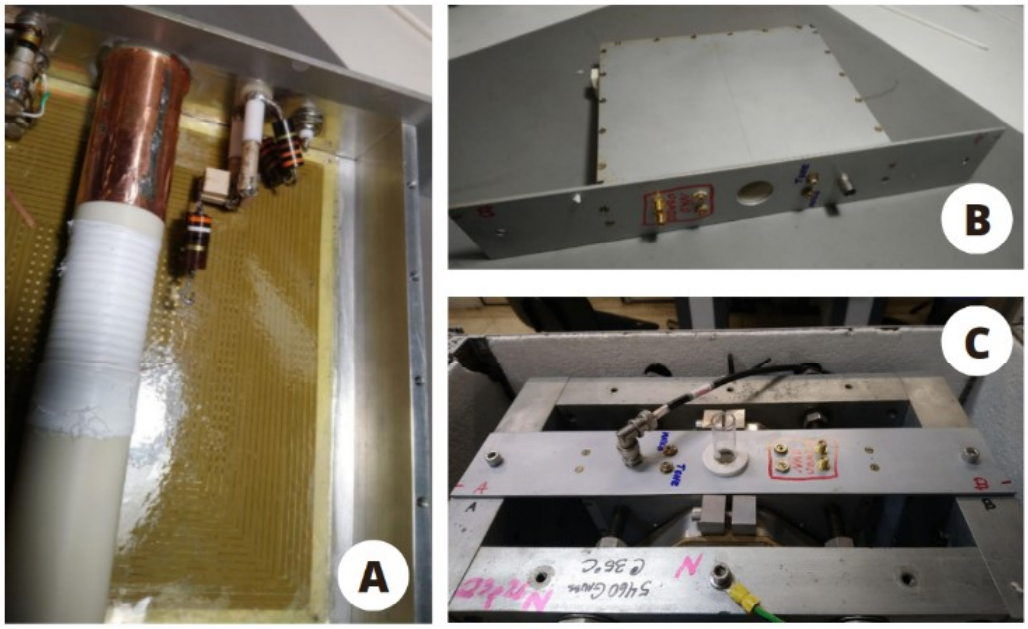

Figure 29 – A: internal view of the probe. B: external view of the probe. C: probe inserted into the magnet with a sample holder.

Source: CIERMag.(23)

Figure 30 – CIERMag Permanent magnet in a thermally insulated box.

Source: CIERMag. (23)

4.2 Implementation of NMR Methods in DMRS

The main objective of this research is the development of Methods * that can operate in the CIERMag Digital Magnetic Resonance Spectrometer. The pulse sequences chosen for the implementation are simple and known, since the emphasis is on the implementation in the DMRS, thus certifying that the equipment is functional and ready for use in NMR experiments. The Methods developed were divided in two categories, namely calibration, and analysis. The former refers to Methods that calibrate experiments to be run in the DMRS - calibration processes are essential for the proper functioning of the spectrometer towards ensuring the correctness of the experiments. The second category is composed of classic Relaxometry NMR Methods, such as CPMG, and Inversion Recovery. F-Language was used in the development of the pulse sequences, whereas Python was adopted for other scripts, including validation and processing.

4.2.1 Calibration Methods

Calibration Methods are used in pre-experiment routines. Those that adjust the spectrometer resonance frequency and RF power are ready for use in the routine of the CIERMag digital spectrometer. The following sections describe the methodology of implementation of the calibration Methods in F-Language.

4.2.1.1 Frequency Adjustment

According to Larmor Equation 2.7, the nuclei precession frequency depends on gyromagnetic ratio, γ, and field intensity. The external field applied may be inhomogeneous due to, for example, magnet imperfections, or variations caused by external factors such as temperature, as discussed in section 2.2. Therefore, the spectrometer frequency operation may undertake variations. Two Methods, namely *Find Resonance* and *Frequency Adjust* have been developed towards adjusting it.

Find Resonance searches for the system's resonance frequency, sending an RF pulse and acquiring an FID at each repetition time. Each RF pulse that excites the sample operates at a different excitation frequency, i.e., the one used in the receiver (Figure 31 illustrates the sequence). The searching bandwidth can be chosen in the Console by setting initial and final values for the frequency. The magnitude order of the frequency variation between each repetition can also be determined in the Console since the number of repetitions desired can be chosen.

* In CIERMag context, an NMR Method represents an experiment project as a software project, containing all information about the experiment from pre-processing to post-processing.

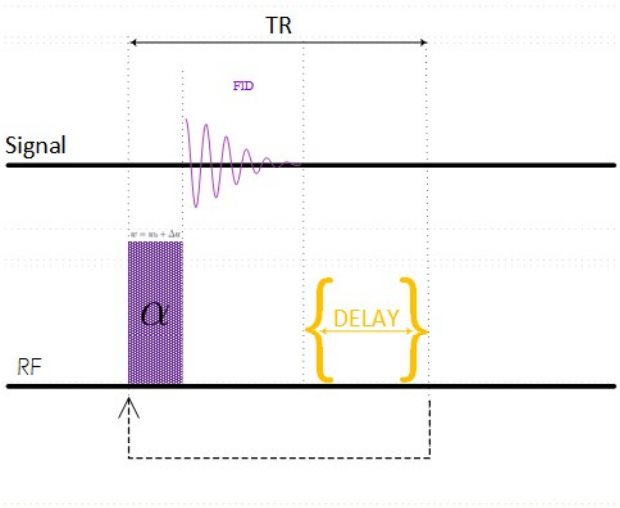

Figure 31 – *Find Resonance* pulse sequence. A $\frac{\pi}{2}$ flip angle pulse is applied at each TR with a different frequency offset.

Source: By the author using MICROSOFT. (24)

As addressed in section 3.2, F-language enables the use of different loop types - Echo and Averages were used in the *Find Resonance* sequence. Events GenExcPulse and Readout, described in A, sent the excitation pulse and received data. Figure 32 displays the sequence including hardware-related structures available for users in F-Language, i.e., loop type index, delays, gate's unblocking, 't_pre' and 't_pos' related to hardware switching, etc. The validation file was used for the calculation of the TR minimum value, as discussed in section 3.2, and the offsets incremented the spectrometer operating frequency. The validation script calculates the offsets and saves them in a table file, whereas the other entities of the Method, i.e., sequences and processing script, can use the values through the tables. The total time duration inside the average loop should be equal to TR; consequently, the delay was calculated to be equal to TR minus the time from the other sequence events.

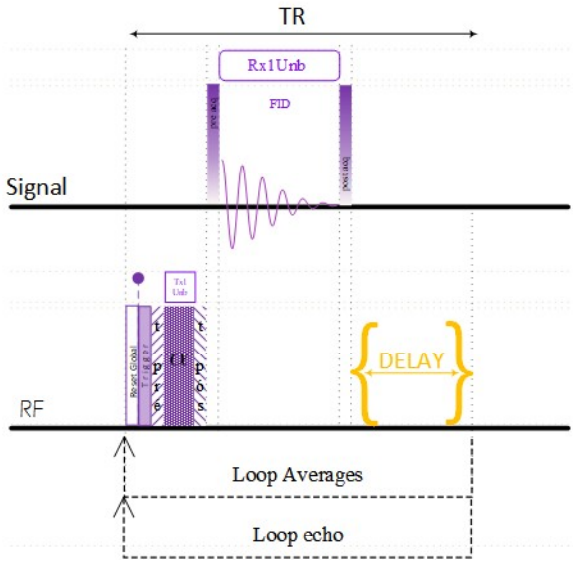

Figure 32 – *Find Resonance* pulse sequence including hardware-related structures. A $\frac{\pi}{2}$ flip angle pulse is applied at each TR with a different frequency offset.

Source: By the author using MICROSOFT. (24)

The echo-typed loop was chosen to profit from a special feature of this loop type. When in the Console, the user can choose the EchoTrain mode to plot all echoes sequentially, allowing the visualization of echo behavior during the individual acquisitions.

Find Resonance processing script reads the acquired data and the offset frequency table from the HDF5 final data file. Both data and parameter values are accessed in the processing script by managers implemented by CIERMag and described in section 3.2. The script calculates the maximum intensity of each FID in the echo loop - the maximum value represents the FID in resonance. Once the resonance frequency value has been found, the script returns the offset value to the console through the persistent parameter protocol discussed in section 3.2.

Frequency Adjust calculates the frequency offset value using Fourier Transform. Figure 33 shows the sequence diagram that excites the sample with an RF pulse and acquires an FID. Differently from *Find Resonance*, *Frequency Adjust* uses only one FID for finding the resonance frequency.

Events GenExcPulse and Readout, described in A, sent the excitation pulse and received data. The Echo loop type controls the number of repetitions performed. Although the method requires only one FID to work properly, more than one repetition can be used

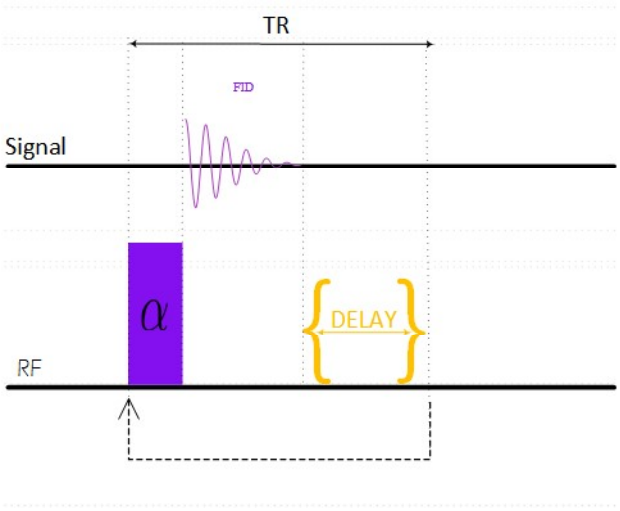

Figure 33 – *Frequency Adjust* pulse sequence. A α flip angle pulse is applied and an FID is acquired for each TR.

Source: By the author using MICROSOFT. (24)

for different applications (i.e., monitoring the resonance frequency variation over time). Figure 34 displays the sequence, including hardware-related structures.

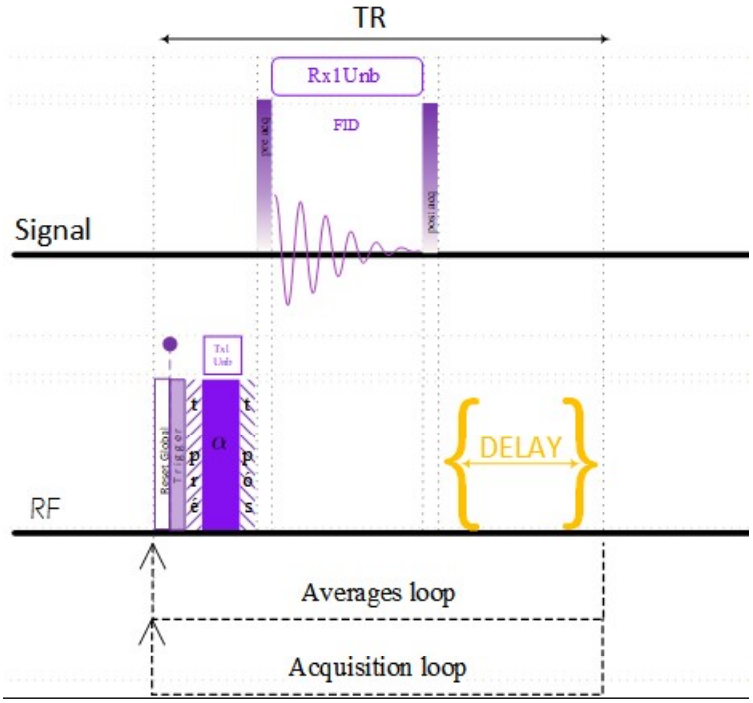

Figure 34 – *Frequence Adjust* pulse sequence including hardware-related structures. A α flip angle pulse is applied and an FID is acquired for each TR.

Source: By the author using MICROSOFT. (24)

The processing scripts zero-fill the data until 64×10^3 points using the python library NMR Glue and calculate Fourier Transform employing the Numpy library. The offset value is calculated with the use of a spectral line peak. In pure water, where the Hydrogen spectrum shows only one peak, the frequency offset is easily calculated through the calculation of the shift in the spectral line for position zero. Once the frequency offset value has been found, the script calculates the spectrometer base operating frequency and returns the value to the console through the persistent parameter protocol.

4.2.1.2 Radiofrequency power Adjustment

Ideally, the radio frequency transmit and receive coil should have a uniform response, i.e., the entire sample should be excited under the same flip angle. However conventionally, it does not occur and the power needs to be adjusted toward minimizing the possible effects of RF (B_1) inhomogeneities. *Flip Adjust* was developed for such a purpose.

The flip angle is determined by 2.20, according to which it depends on B_1 amplitude/power and pulse duration. An amplitude scan is performed for the RF hard pulse in the radiofrequency power adjustment Method - the sequence used sends pulses with a small variation in amplitude, keeping the pulse duration constant, and acquires an FID for each pulse. Figure 35 shows the sequence.

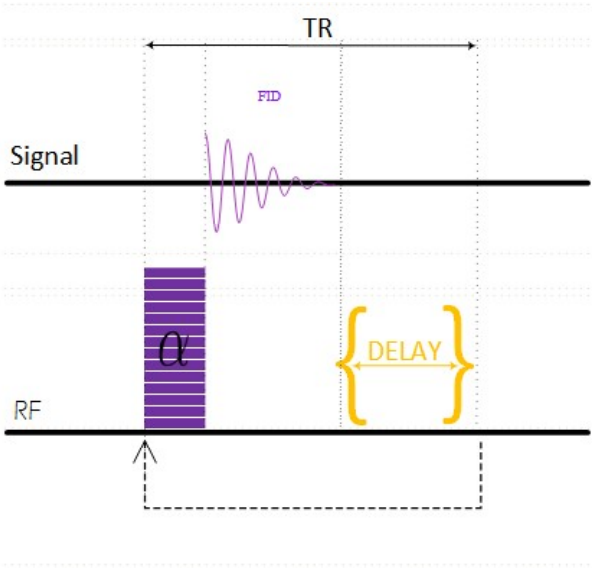

Figure 35 – *Flip Adjust* pulse sequence.

Source: By the author using MICROSOFT. (24)

Events GenExcPulse and Readout, described in A, sent the excitation pulse and received the data, and the Echo loop type controlled the number of repetitions performed.

GenExcPulse received a different pulse amplitude in each repetition - all amplitudes were previously calculated in the validation file and the values were stored in a table.

The system chooses the maximum flip angle of 180° to calculate the amplitudes used in the scan and the corresponding amplitude considering no experimental condition such as field inhomogeneities. The step size in the amplitude variation is calculated by dividing the maximum amplitude for the maximum flip angle by the number of repetitions. Python script uses a loop to the index to multiply the step size and calculate the amplitudes and the tables can be instantiated by the sequence file. The total time duration in the average loop should be equal to TR; consequently, the delay was calculated as being equal to TR minus the time from the other sequence events. Figure 36 shows the sequence, including hardware-related structures available for users in F- Language, i.e., loop type index, delays, gate's unblocking, 't_pre' and 't_pos' times related to hardware switching, etc.

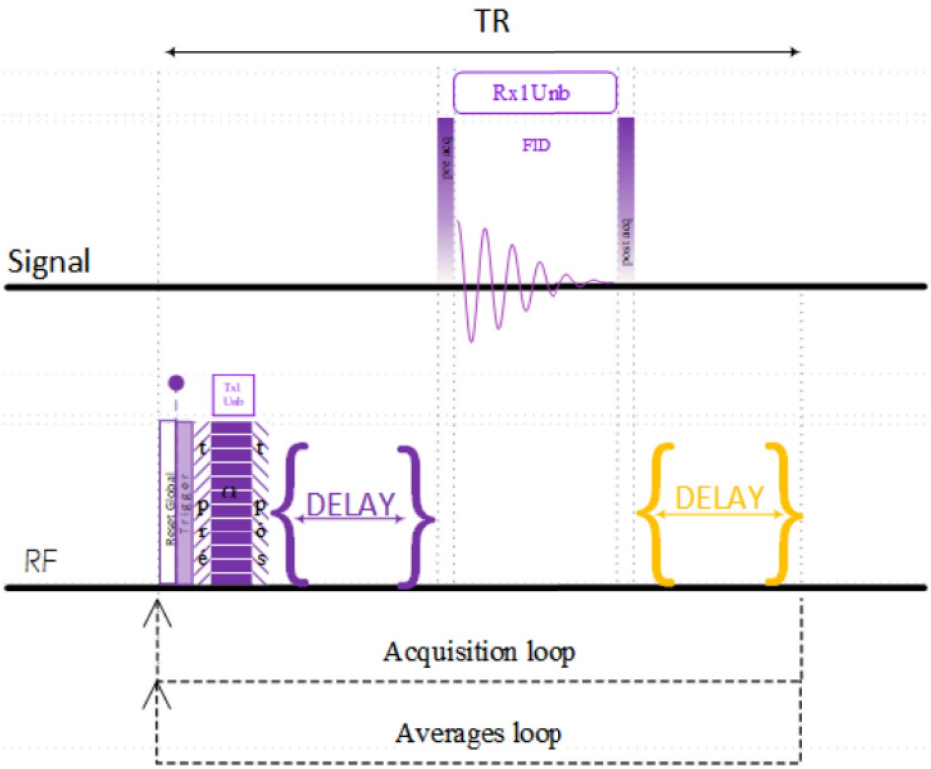

Figure 36 – *Flip Adjust* pulse sequence including hardware-related structures.

Source: By the author using MICROSOFT. (24)

The processing scripts read the data from the HFD5 file and then calculate the amplitude for the 90° or 180° pulse, according to the option set in the Console. Both data and parameter values are accessed in the processing script by the managers implemented by CIERMag and described in section 3.2. The processing protocol uses the maximum

of the real part of the Fourier transform from the acquired FIDs to calculate the pulse amplitude for the 90° pulse. When the flip angle reaches 90°, the signal is maximum and the corresponding amplitude will be the calibration amplitude of the 90° pulse. Since the 180° pulse inverts the magnetization phase, the methodology adopted searches for the phase inversion in the signal for calibrating the 180° pulse.

4.2.2 Methods of Relaxometry Measurements

The Methods Inversion Recovery (IR) and Carr-Purcell-Meiboom-Gill (CPMG) were implemented in F language compatible with the DMRS. Both are well-established Methods for T_1 and T_2 measurements respectively. The following sections describe the methodology of implementation of the calibration Methods in F-Language.

4.2.2.1 Carr-Purcell-Meiboom-Gill (CPMG)

The CPMG sequence uses a $\frac{\pi}{2}$ pulse to flip the magnetization into the transverse pulse and a series of refocusing pulses π. After each refocusing pulse, an echo is acquired and the exponential decay of the transverse magnetization can be measured by decaying the intensities of the echoes. Figure 11 displays the decaying and Figure 37 displays the sequence diagram.

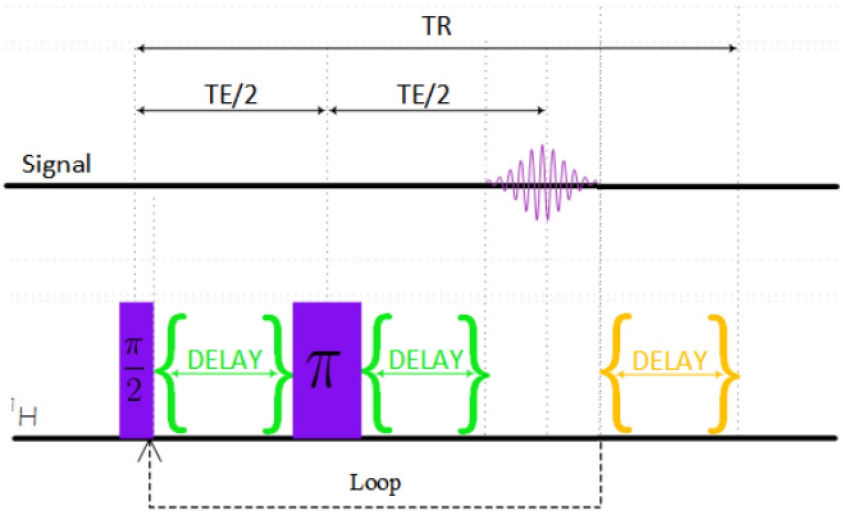

Figure 37 – CPMG pulse sequence.

Source: By the author using MICROSOFT. (24)

The CPMG sequence in F - Language uses an echo loop type to control the number of π pulses. An Average loop type is also used to control the average number. Figure 38 displays the sequence diagram of CPMG including F- Language structures hardware

related. The delays in the sequence were precisely calculated in the validation script to ensure that the time between the π pulse and the acquisition is equal to $\frac{TE}{2}$, where TE is the echo time.

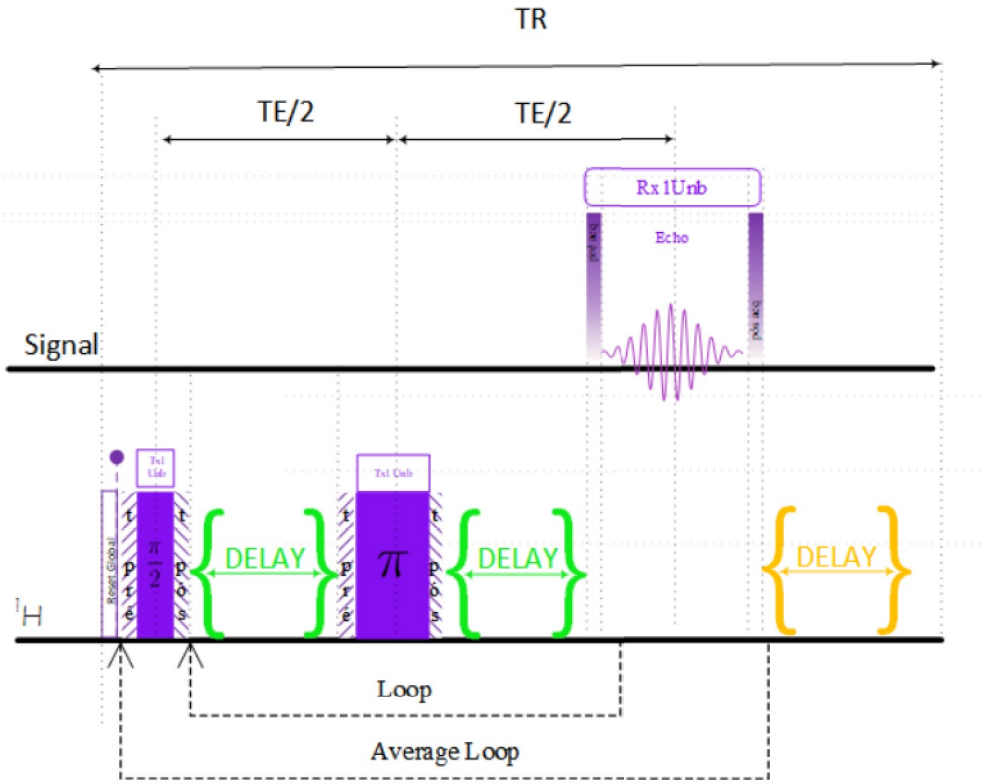

Figure 38 – CPMG pulse sequence including hardware-related structures.

Source: By the author using MICROSOFT. (24)

The CPMG processing script calculates the maximum of the real part from each acquired signal, after a phase correction, and uses the function opt.curve_fit from scipy library to calculate exponential fitting, using the equation 2.26 to determine the T_2 value. The opt.curve_fit from scipy uses non-linear least squares to fit the function. The processing script also opens a plot window showing the decay and returns the T_2 as information in the logger window.

The Meiboom-Gill contribution for the sequence, introducing a 90° phase difference between the $\frac{\pi}{2}$ pulse and the π pulses avoids the effect of incorrectly calibrated π pulse in the signal. Nonetheless, a background interference from the spectrometer electronics may arise. It is possible to distinguish between the NMR signal and the electronic background by RF phase alternation. For example, a successive phase alternation between the π pulses generates successive addition and subtraction of the signal that will lead to the coherent superposition of the NMR signal while the interference background is nullified. (28) This

process namely phase cycling and the successive phases of the transmitter and receivers were stipulated using tables declared intrinsically in the F code. Each average uses one different value from the phase cycling table according to the loop index.

4.2.2.2 Inversion Recovery (IR)

The inversion recovery experiment consists of a combination of two RF pulses. The first pulse inverts the longitudinal magnetization and the second tips the longitudinal magnetization into the transverse plane so an FID signal may be measured. The interval τ_I, called inversion time, between pulses, is increased with each acquisition; therefore, the longitudinal magnetization recovers a little more with each acquisition. Figure 13 shows a schematic of the Inversion Recovery sequence and the return of magnetization to the equilibrium value.

The user can define a parameter called the number of repetitions that represents the total number of acquisitions in the IR experiment. The value τ_I grows with the number of repetitions. To calculate it is defined a variable called evolution time (T_{evol}), whose value can be chosen by the user. The validation script calculates the inversion times based on the number of repetitions loop indexing, and this script generates a delay table - with the extension ".tbl" that can be read by the sequence file - in which each inversion time is calculated according to 4.1 where i is the number of repetition/acquisition.

$$\tau_{I_i} = i^2.T_{evol} \tag{4.1}$$

Figure 39 displays the Inversion Recovery sequence diagram.

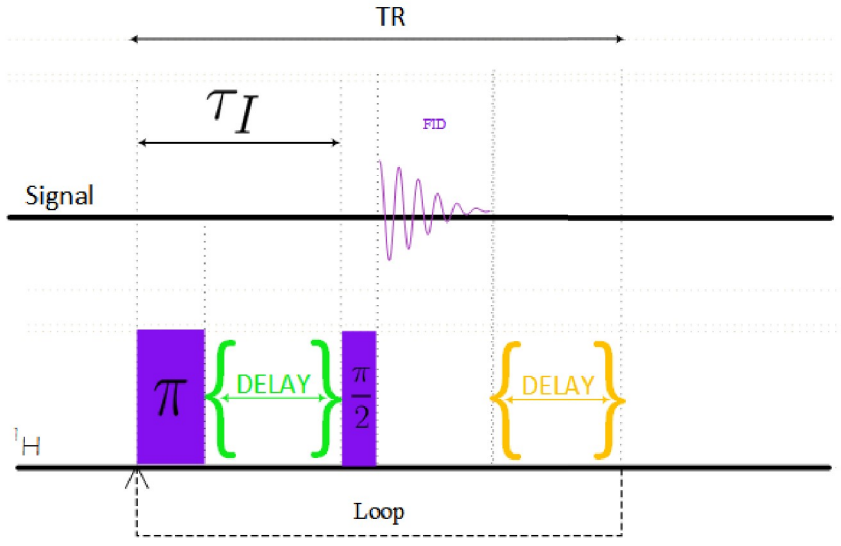

Figure 39 – IR pulse sequence.

Source: By the author using MICROSOFT. (24)

The IR sequence in F - Language uses an echo loop type to control the number of repetitions of the two pulses application. An Average loop type is also used to control the average number. Figure 40 displays the sequence diagram of IR including F- Language structures hardware related. In each repetition of the echo loop type, a different inversion time is used for the sequence.

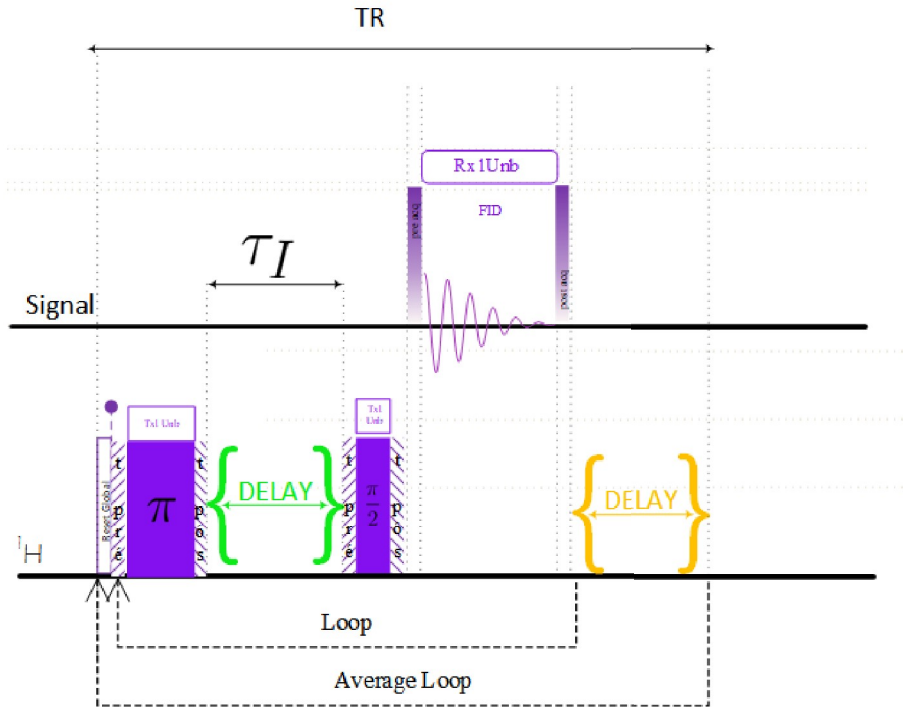

Figure 40 – IR pulse sequence including hardware-related structures.

Source: By the author using MICROSOFT. (24)

In the IR processing script the T_1 values can be calculated using two different methodologies:

A: The processing script calculates the maximum of the FFT module for each acquisition and uses the function opt.curve_fit from scipy (26) library to calculate exponential fitting, using the equation 2.23 to determine the T_1 value. The opt.curve_fit from scipy uses non-linear least squares to fit the function.

B: The processing script calculates the T_1 value using the equation 2.33. In this case, the $T_{I,null}$ is defined as the minimum intensity signal for the FFT module of all the acquisitions.

4.3 Description of the experiments

Different experiments were performed with the Methods described in section 4.2 and implemented in the spectrometer. The next section describes the experimental protocol.

4.3.1 Signal to Noise ratio

The signal-to-noise ratio was calculated by the CIERMag Digital Magnetic Resonance Spectrometer and measurements were taken by a simple sequence present in the CIERMag suite of Methods, called *1DSimple*, which consists of a pulse with predetermined amplitude and duration followed by an FID acquisition.

According to the methodology adopted, the signal strength was the maximum from the real part of the Fourier Transform calculated after the FID phase correction. The 90° excitation flip angle was calibrated with a 15 μ s hard pulse and 20 Watts peak power for the signal acquisition. A new acquisition was performed for the noise amplitude measurement under the same conditions adopted for the signal acquisition, but with RF power set to zero. The noise value was calculated with the root to mean square (RMS) for the real part of the Fourier Transform from the acquired noise.

The sample was 1 ml of mineral oil placed in the center of the coil. A TR = 20 s was chosen and the number of points per acquisition was equal to 1000 with zero fill for 64000 points. The acquisition bandwidth used was 100 kHz with no average and no dummy scan.

4.3.2 The efficiency of the Frequency adjustment Methods

Find Resonance and *Frequency adjust*, described in section 4.2.1.1, searched for the spectrometer operating frequency. The sample chosen for both was 1 ml water and the repetition time used was TR = 10 seconds. A 90° excitation flip angle of 15 μs duration was used and the pulse amplitude was adjusted to 0.4879 a.u., corresponding to approximately 20 Watts of peak power. The Methods ran with no signal average and no dummy scan.

A 100 kHz bandwidth was chosen for the search by *Find Resonance*, setting the search initial and final frequencies to - 50 kHz and 50 kHz, respectively. The Method ran with 1000 repetitions, representing a 0.1 kHz ($\approx$ 4.34 ppm) variation in the RF pulse excitation frequency for each repetition. Only one repetition was performed for *Frequency Adjust*.

Besides comparing the efficiency of the two Methods, the experiment aimed at finding a good methodology for calibrating the spectrometer operation frequency and ensuring the resonance frequency would be correctly calculated in the pre-experiment CIERMag routine prior to measurements.

4.3.3 Temperature and Field variation

Frequency Adjust, described in section 4.2.1.1, was used to evaluate the effect of room temperature variation on the spectrometer operating frequency. The Method was run in the spectrometer for 16.6 hours and the ambient temperature was monitored during that time.

A digital thermometer measured the room temperature and a webcam recorded the values displayed. A python program, developed by student Hendrik Louzada, recorded the data read from the thermometer in a file and the temperature values were acquired at 2.5 s intervals and averaged every 20 acquisitions.

The repetition time used in the pulse sequence was TR = 60 seconds with 500 repetitions using two averages and no dummy scan. A 90° flip angle with 15 μs duration was used for the excitation, and the pulse amplitude was set to 0.4879 a.u., corresponding to approximately 20 Watts of peak power. The sample chosen for the experiment was 1 ml of mineral oil.

Measurements were taken between 8 pm and 1 pm in April when the temperature in São Carlos normally drops during the early morning hours. The magnet is commonly maintained in a thermally insulated box, as shown in Figure 30, and the room's air conditioning is kept on at 22°C. The experiment aimed at tracking the relationship between frequency variation and room temperature and understanding if temperature drops interfere with the measurements.

4.3.4 Flip Angle calibration by Radiofrequency Power Adjustment Method

Flip Adjust, described in section 4.2.1.2, adjusted the radiofrequency power and calibrated the flip angle. The sample chosen was 1 ml of mineral oil and the repetition time used was TR = 10 seconds. The number of points per acquisition was equal to 200 and the number of repetitions was 50 with 2 averages.

The Method uses the amplitude for the 180° maximum flip angle automatically calculated for the system as a reference for calculating the amplitudes used in the search. The automatic value for the 180° pulse amplitude was 0.8696 a.u. and a 15 μs pulse duration was kept constant for all acquisitions.

4.3.5 Measurement of T_2 from water using CPMG

CPMG Method, described in section 4.2.2.1 were used to measure the T_2 from the demineralized water at 35ºC temperature. The sample used was 1ml from demineralized water, the repetition time TR = 60 s, and the echo time TE = 10 ms. The number of points per acquisition was equal to 16, with zero fill for 64000 points, and the number of acquired echoes was 1000 with 4 averages. Table 2 illustrates the phase cycling used.

Table 2 – Pulse phases in the CPMG sequence.

Average	π	$\frac{\pi}{2}$	**Receptor**
1	x	y	y
2	-x	-y	-y
3	y	-x	-x
4	-y	x	x

Source: By the author.

The pulses were calibrated using the *Flip Adjust* Method, the 90° flip angle was adjusted to 15 μs duration and 0.7221 a.u. of amplitude which corresponds to approximately 20 Watts of peak power, and the 180° flip angle was adjusted to 15 μs duration, and 0.4020 a.u. of amplitude that corresponds to approximately 80 Watts of peak power.

4.3.6 Measurement of T_2 from different concentrations of copper sulfate in water using CPMG

CPMG Method, described in section 4.2.2.1 was used to measure the T_2 from four solutions with different concentrations of copper sulfate in water at a temperature of 35ºC. For all the procedures, 1 ml of the solution was used. The number of points per acquisition was equal to 16, with zero fill for 64000 points, and the number of acquired echoes was 500 with 4 averages. Table 3 illustrates the solutions concentration and the values from TR, and TE used for each one.

Table 3 – Different concentrations of the four solutions of copper sulfate in water and values from TR and TE used in each experiment.

Concentration (g/L)	**TR (s)**	**TE (ms)**
1.25	60	10
2.50	60	5
5.00	60	2.5
10.00	60	1.25

Source: By the author.

Table 2 illustrates the phase cycling used. The pulses were calibrated using the *Flip Adjust* Method, the 90° flip angle was adjusted to 15 μs duration and 0.4879 a.u. of amplitude which corresponds to approximately 20 Watts of peak power, and the 180° flip angle was adjusted to 27 μs duration, and 0.4879 a.u. of amplitude that corresponds to approximately 80 Watts of peak power.

4.3.7 Measurement of T_1 from water using IR

IR Method, described in section 4.2.2.2 was used to measure the T_1 from the demineralized water at 35ºC temperature. The sample used was 1ml from demineralized water, the repetition time TR = 60 s and the evolution time Tevol = 1.5 ms was used to calculate the different values from the inversion time (T_I), and the number of acquisitions was 150 with no averages, in this context, since the delayed growth is non-linear, as described in 4.2.2.2 table 4 displays the inversion time use in each acquisition. The number of points per acquisition was equal to 100, with zero fill for 64000 points.

Table 4 – Inversion times (T_I) used in each acquisition between the π and the $\frac{\pi}{2}$ pulse in the IR experiment from water.

i	T_I (s)	i	T_I **(s)**	i	T_I **(s)**	**i**	T_I (s)	**i**	T_I (s)
0	0.0	30	1.35	60	5.04	90	11.61	120	21.60
1	1.5×10^{-3}	31	1.44	61	5.22	91	11.88	121	21.96
2	6.0×10^{-3}	32	1.53	62	5.40	92	12.15	122	22.32
3	1.35×10^{-2}	33	1.63	63	5.58	93	12.42	123	22.69
4	2.4×10^{-2}	34	1.73	64	5.76	94	12.69	124	23.06
5	3.75×10^{-2}	35	1.83	65	5.95	95	12.97	125	23.43
6	5.40×10^{-2}	36	1.94	66	6.14	96	13.25	126	23.81
7	7.35×10^{-2}	37	2.05	67	6.33	97	13.53	127	24.19
8	9.60×10^{-2}	38	2.16	68	6.53	98	13.82	128	24.57
9	1.21×10^{-1}	39	2.28	69	6.73	99	14.11	129	24.96
10	1.50×10^{-1}	40	2.40	70	6.93	100	14.40	130	25.35
11	1.81×10^{-1}	41	2.52	71	7.14	101	15.00	131	25.74
12	2.16×10^{-1}	42	2.64	72	7.35	102	15.30	132	26.13
13	2.53×10^{-1}	43	2.64	73	7.56	103	15.60	133	26.53
14	2.94×10^{-1}	44	2.77	74	7.77	104	15.91	134	26.93
15	3.37×10^{-1}	45	2.90	75	7.99	105	16.22	135	27.33
16	3.84×10^{-1}	46	3.03	76	8.21	106	16.53	136	27.74
17	4.33×10^{-1}	47	3.17	77	8.43	107	16.85	137	28.15
18	4.86×10^{-1}	48	3.17	78	8.66	108	17.17	138	28.56
19	5.41×10^{-1}	49	3.31	79	8.89	109	17.82	139	28.98
20	6.00×10^{-1}	50	3.45	80	9.12	110	18.15	140	29.40
21	6.61×10^{-1}	51	3.60	81	9.36	111	18.48	141	29.82
22	7.26×10^{-1}	52	3.75	82	9.60	112	18.81	142	30.24
23	7.93×10^{-1}	53	3.90	83	9.84	113	19.15	143	30.67
24	8.64×10^{-1}	54	4.05	84	10.08	114	19.49	144	31.10
25	9.37×10^{-1}	55	4.21	85	10.33	115	19.83	145	31.53
26	1.01	56	4.37	86	10.58	116	20.18	146	31.97
27	1.09	57	4.53	87	10.83	117	20.53	147	32.41
28	1.17	58	4.70	88	11.09	118	20.88	148	32.85
29	1.26	59	4.87	89	11.35	119	21.24	149	33.33

Source: By the author.

The pulses were calibrated using the *Flip Adjust* Method, the 90° flip angle was adjusted to 15 μs duration and 0.46 a.u. of amplitude that correspond to approximately 20 Watts of peak power, and the 180° flip angle was adjusted to 26 μs duration and 0.46 a.u. of amplitude that corresponds to approximately 80 Watts of peak power.

4.3.8 Measurement of T_1 from different concentrations of Copper Sulfate in water using IR

IR Method, described in section 4.2.2.2 was used to measure the T_1 from four solutions with different concentrations of copper sulfate in water at a temperature of 35ºC. For all the procedures, 1 ml of the solution was used, The number of points per acquisition was equal to 100, with zero fill for 64000 points, and the number of acquisitions was 150 with no averages. Table 5 illustrates the solutions concentration and the values from TR, evolution time (T_{evol}) used to calculate the inversion time in each case according to the methodology described in 4.2.2.2.

Table 5 – Different concentrations for the four solutions of copper sulfate in water and values from TR and evolution time (T_{evol}) used in each experiment.

Concentração (g/L)	**TR (s)**	T_{evol} **(ms)**
1.25	60	2.00
2.50	60	1.50
5.00	60	0.50
10.00	60	0.25

Source: By the author.

The pulses were calibrated using the *Flip Adjust* Method, the 90° flip angle was adjusted to 15 μs duration and 0.400059 a.u. of amplitude which corresponds to approximately 20 Watts of peak power, and the 180° flip angle was adjusted to 30 μs duration, and 0.400059 a.u. of amplitude that corresponds to approximately 80 Watts of peak power.

5 RESULTS AND DISCUSSION

This section discusses the results of the experiments described in section 4.3.

5.1 Signal to Noise ratio

The signal strength in the methodology adopted was the maximum from the real part of the Fourier Transform calculated after the signal phase correction. Figure 41 shows the graphic representation of the Fourier Transform. The noise value was calculated with the use of the root mean square for the real part of the Fourier Transform from the acquired noise data.

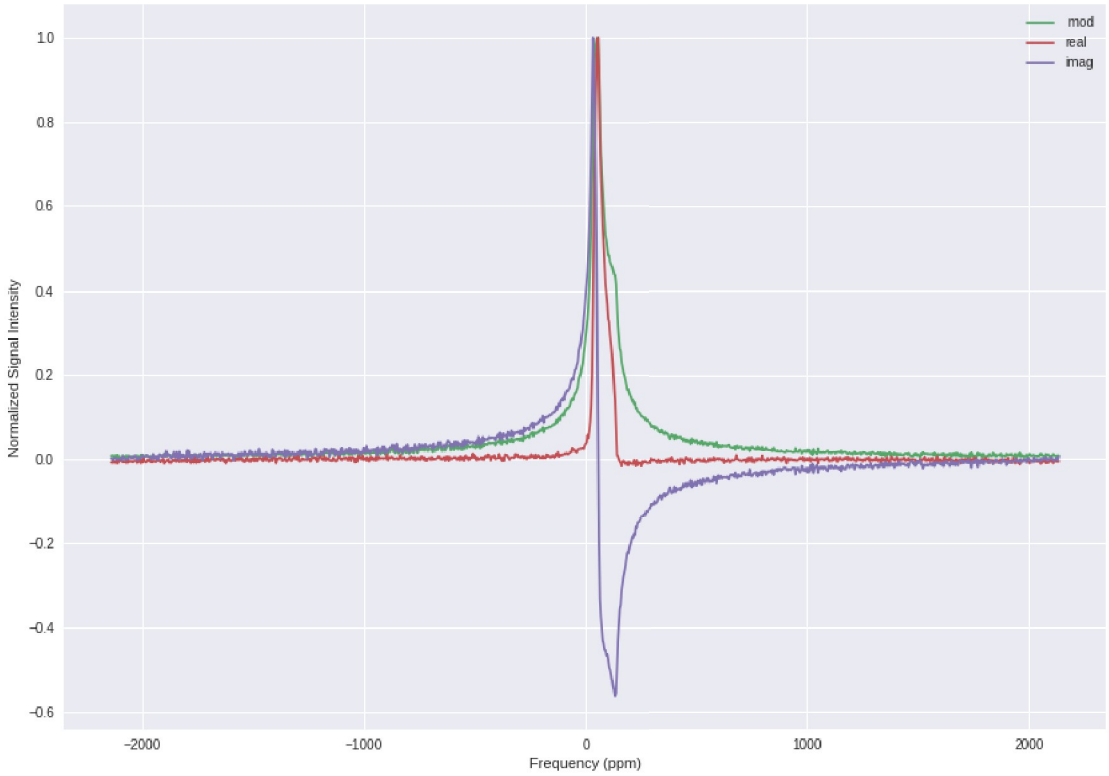

Figure 41 – Module and real and imaginary parts from the Fourier Transform calculated from the acquired FID.The sampe uses was 1 ml from mineral oiil at 35°C.

Source: By the author.

A statistic was calculated for 10 different acquisitions under the same conditions. The averages for the signal intensity and for the RMS noise were S = 260389.27 a.u. and

N = 798.29 a.u., respectively, and the mean value found for the signal-to-noise ratio was S/N = 332.26, with a 5.73 standard deviation.

5.2 Analysis of the Efficiency of Frequency Adjustments Methods

Frequency Adjust found a −13.4 KHz frequency offset and the spectrometer operation frequency was corrected from 23.260 MHz to 23.246 MHz. *Find Resonance* found a −32.2 KHz frequency offset and the operation frequency was corrected from 23.260 MHz to 23.227 MHz. The condition under which the experiments were carried out as described in section 4.3.2.

Figure 42 displays the Fourier Transform calculated from the FID signals acquired before and after the spectrometer operation frequency had been corrected by the two different calibration Methods. The spectral line is centered in the zero position in *Frequency Adjust* and shifted approximately 1000 ppm from that position in*Find Resonance*.

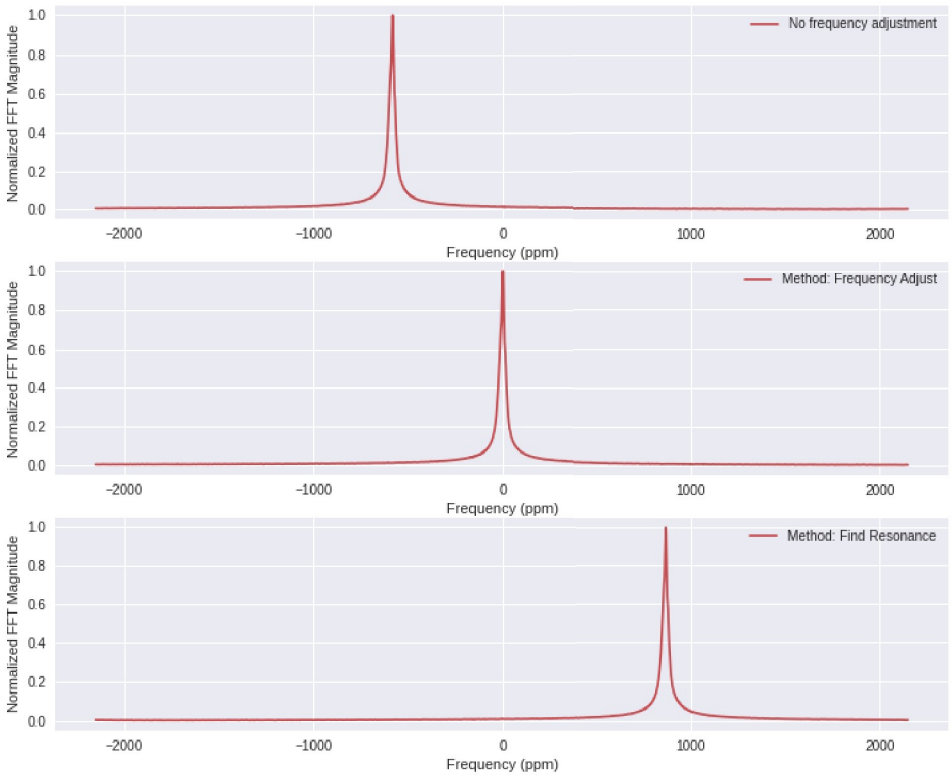

Figure 42 – Fourier Transforms calculated from FID signals – the spectrometer operation frequency had been corrected by the two calibration Methods. In the upper part of the image is the spectral peak prior to corrections; in the middle is the spectral peak after the correction by Frequency Adjust, and in the last graph is the spectral peak after the correction performed by Find Resonance. The sample used was 1 ml from demineralized water ate 35°C.

Source: By the author.

Since *Frequency Adjust* uses only one repetition to calculate the frequency offset, the execution was fast and the experiment duration was approximately 20 seconds. On the other hand, *Find Resonance* required 2 hours and 45 minutes to perform 1000 repetitions.

There are no functional differences between the two preceding Methods, in the sense that both can be used to find the resonance condition. What differs is the accuracy, higher in the latter. According to the results, *Frequency Adjust* is more efficient in finding the fine resonance frequency, whereas *Find Resonance* can find the coarse resonance frequency. The user could first seek for the resonance when it is completely unknown, such as when first calibrating the system after a new field up, using the *Find Resonance*, and subsequently refining the search using the *Frequency Adjust*.

Improvements in the accuracy of *Find Resonance* calculated value require a reduction in the size of the frequency variation between repetitions. Nevertheless, the approach is not efficient, since the duration of the experiment for the current configuration is almost three orders of magnitude bigger than that of *Frequency Adjust*. A possible solution to improve the efficiency of *Find Resonance*, keeping the execution time short, refers to the possibility of exploring explore more efficient search algorithms, such as an interactive binary search for the resonance frequency.

5.3 Analysis of Temperature and Field variation

The experiment described in section 4.3.3 analyzes whether the magnet thermal insulation was efficient and whether temperature drops interfered with the measurements. A survey from the variation in the spectrometer operating frequency over the room temperature variation was therefore conducted and the results are shown in the graphic in Figure 43.

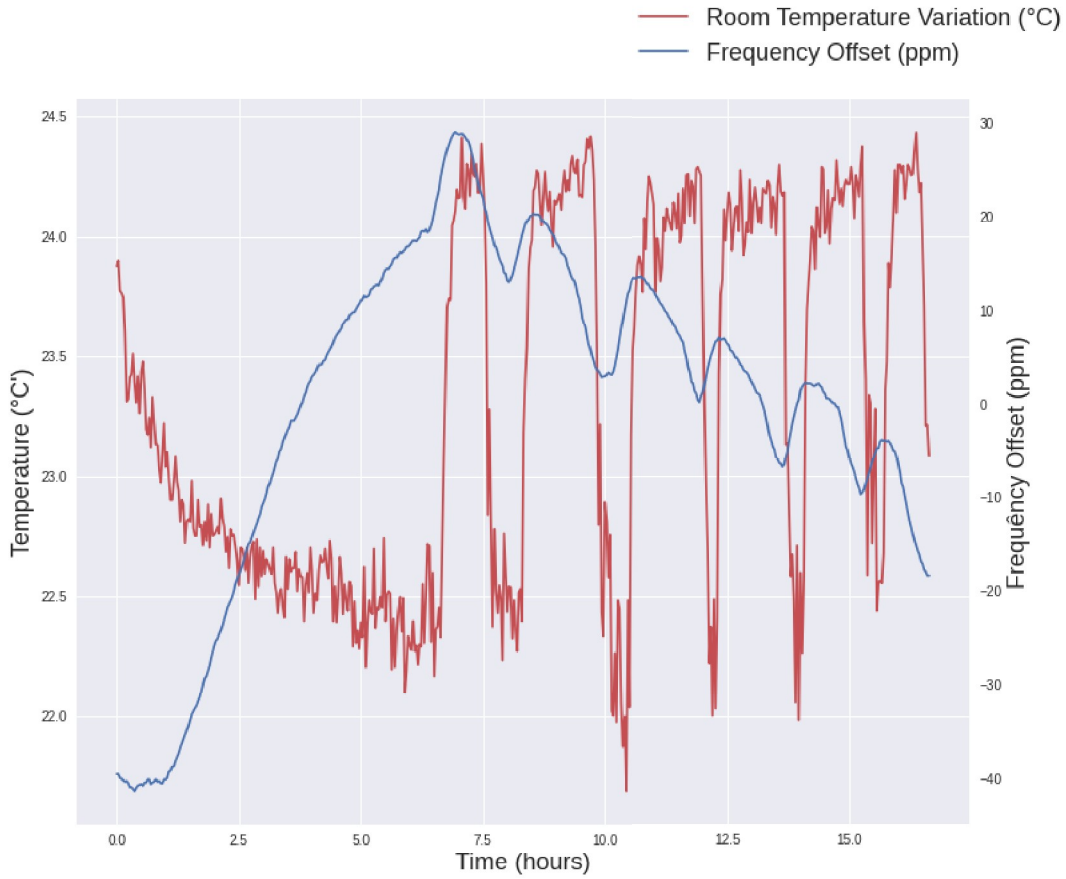

Figure 43 – Time variation in hours over temperature variation (°C) (left side of the abscissa axis) and frequency offset (right side) in ppm.

Source: By the author.

Figure 43 displays a relationship between temperature and frequency offset, with a decrease in room temperature and an increase in the spectrometer operating frequency. An approximately 2.5 °C variation in temperature generates a 70 ppm variation in frequency, which represents a 4.3 ppm/h average rate of change. The first part of the graph where there is a more sudden drop in temperature represents the night and dawn, while the second part represents the morning.

Permanent magnets are sensitive to temperature changes, as discussed in section 2.2, in this case, it is possible to observe the behavior of the negative temperature coefficient of the magnetic material from the NdFeB. The magnetic field strength decreases as temperature increases and vice versa. Moreover, the insulation for both magnet and thermal control may not fulfill the expected function, even with internal control and thermal insulation (at 35°C) the magnet still feels the room temperature. This study has led to the conclusion the system is ideal for short measurements, due to the frequency variation in the long-time measurements.

5.4 Flip Angle calibration by the Radiofrequency Power Adjustment Method

Table 6 shows the values for the amplitudes of 90° and 180° flip angle pulses calculated automatically by Flip Adjust.

Table 6 – Values for pulse duration and amplitude calculated by Flip Adjust for 90° and 180° flip angles.

θ	**Amplitude (u.a)**	**Duração do pulso (μs)**
90°	0.3915	15
180°	0.7479	15

Source: By the author.

A statistic was calculated for 10 different acquisitions under the same conditions, and the result was 0.0297 u.a standard deviation for the amplitude of the 90° flip angle. The same amplitude value was found for 180° for all acquisitions. The graphic in Figure 44 shows signal amplitudes versus RF amplitude. The blue line in the upper part represents the maximum signal amplitude, which corresponds to the 90° flip angle, and the blue line in the lower part denotes the amplitude from the 180° flip angle.

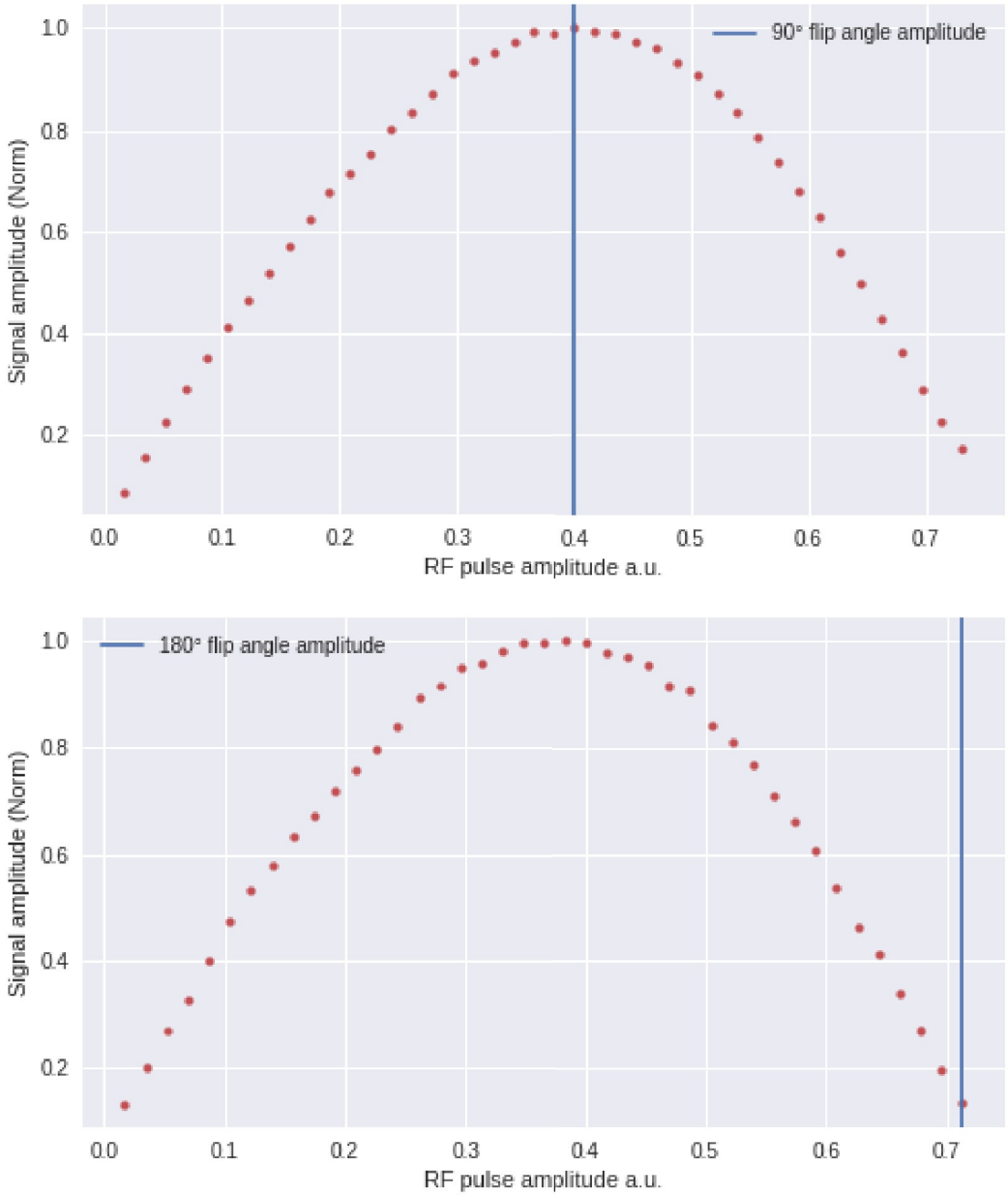

Figure 44 – Signal amplitudes versus RF amplitude in Flip Angle Search Method. The blue line in the upper part represents the maximum signal amplitude, which corresponds to the 90° flip angle, whereas the blue line in the lower part denotes the signal amplitude, which corresponds to the 180° flip angle.The sample used was 1 ml from mineral oil at 35°C.

Source: By the author.

Figure 45 depicts two FIDs acquired with the use of the flip angles calibrated by Flip Adjust. The upper image shows the FID using the 90° flip angle, and the lower one uses the 180° flip angle. Apart from the different flip angles, all other conditions were identical, i.e., the two acquisitions used no averages and acquired 300 points. Red and

blue signals represent their real and imaginary parts, respectively. The drop in the signal amplitude with a reduction in the signal-to-noise ratio and the phase inversion is observed between the two acquisitions.

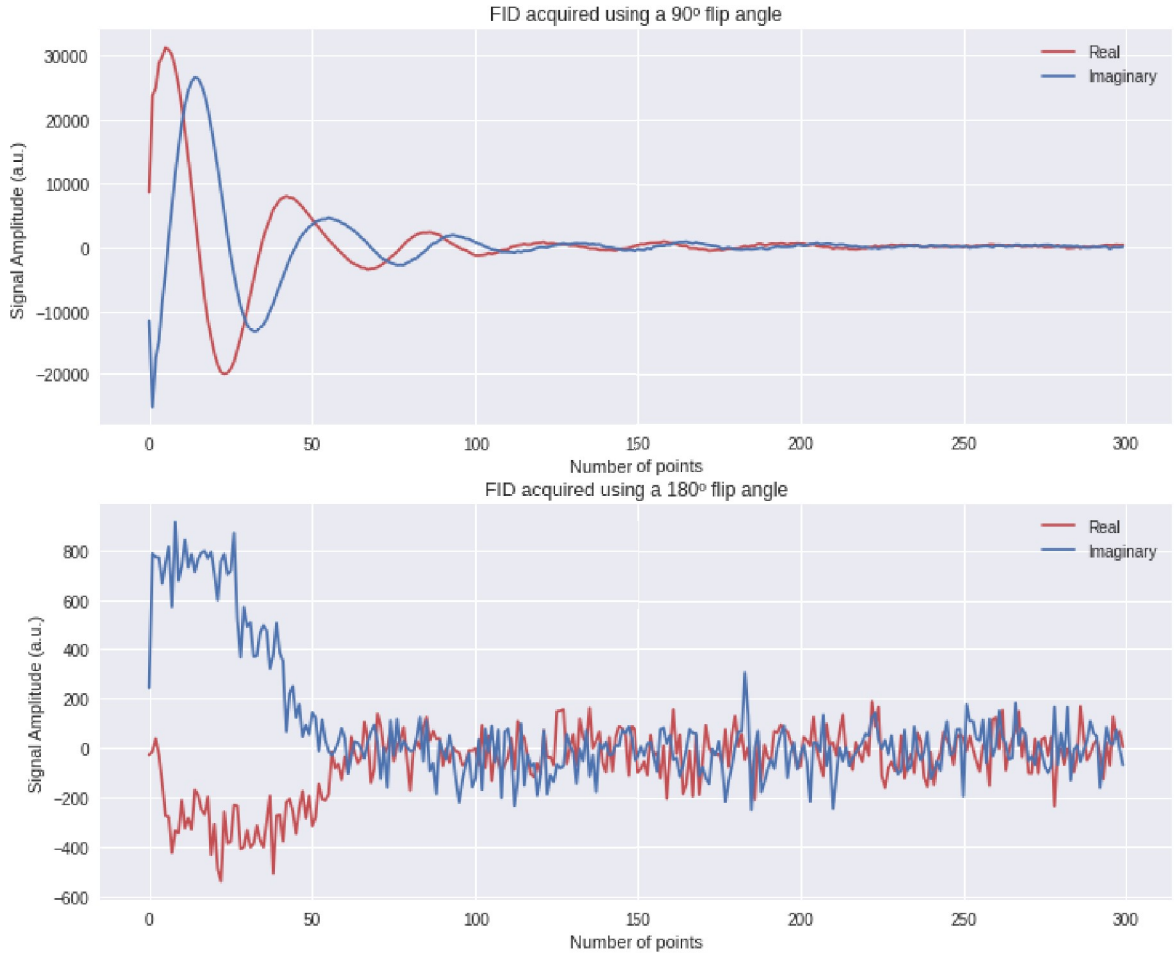

Figure 45 – FIDs acquired with the use of the flip angles calibrated by Flip Adjust. The upper part shows the FID acquired with the 90° flip angle and the lower one display it acquired with the 180° flip angle.

Source: By the author.

5.5 Results from the measurement of T_2 from water using CPMG

The methodology described in section 4.3.5 was used to calculate the T_2 value from 1 ml of demineralized water at a temperature of 35ºC. A statistic was calculated for 10 different acquisitions under the same conditions and an average was taken, the value found was equal to $T_2 = 2.095$ s with a 0.006 s standard deviation. The result is in accordance with the expected value of the transverse relaxation time from demineralized water at the

given temperature. (12) Figure 46 displays the exponential fitting for the water transverse relaxation decay.

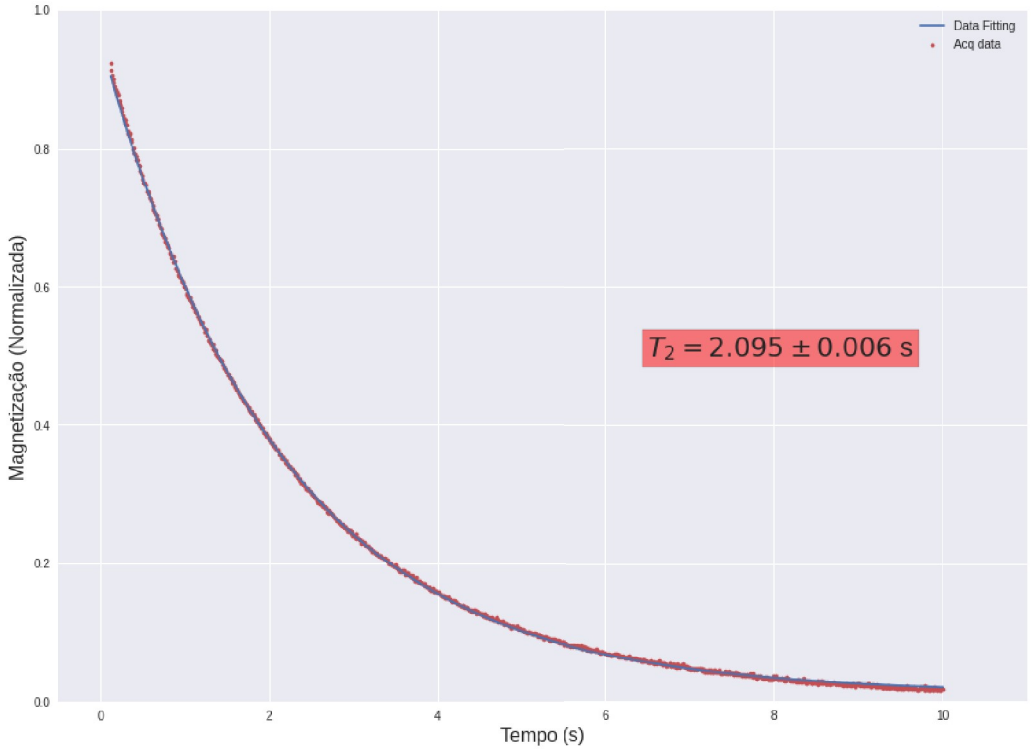

Figure 46 – Exponential decay fitting for the transverse relaxation decay from demineralized water.

Source: By the author.

The T_2 value was obtained through the exponential fitting from the acquired data using the equation 2.26 as reference for the fitting, the R^2 calculated for this fitting was 0.999 which means that the model represents 99.9% of the observed data. The satisfactory result found for the transverse relaxation time of the water demonstrates that the DMRS is functional and ready for use.

5.6 Results from the measurement of T_2 from different concentrations of Copper Sulfate in water using CPMG

The methodology described in section 4.3.6 was used to measure the T_2 relaxation time for four solutions with different concentrations of water in copper sulfate. Table 7 displays the value of T_2 found for each concentration of water in copper sulfate.

Table 7 – Four different concentrations of water in copper sulfate in g/l and the corresponding T_2 value in ms.

Concentration (g/L)	T_2 **(ms)**
1.25	237.71 ± 3.19
2.50	108.46 ± 0.70
5.00	53.31 ± 1.51
10.00	26.86 ± 0.17

Source: By the author.

Figure 47 displays the linear relationship between the inverse of the transverse relaxation time and the concentration of water and copper sulfate. The R^2 calculated for this fitting was 0.999 which means that the model represents 99.9% of the observed data.

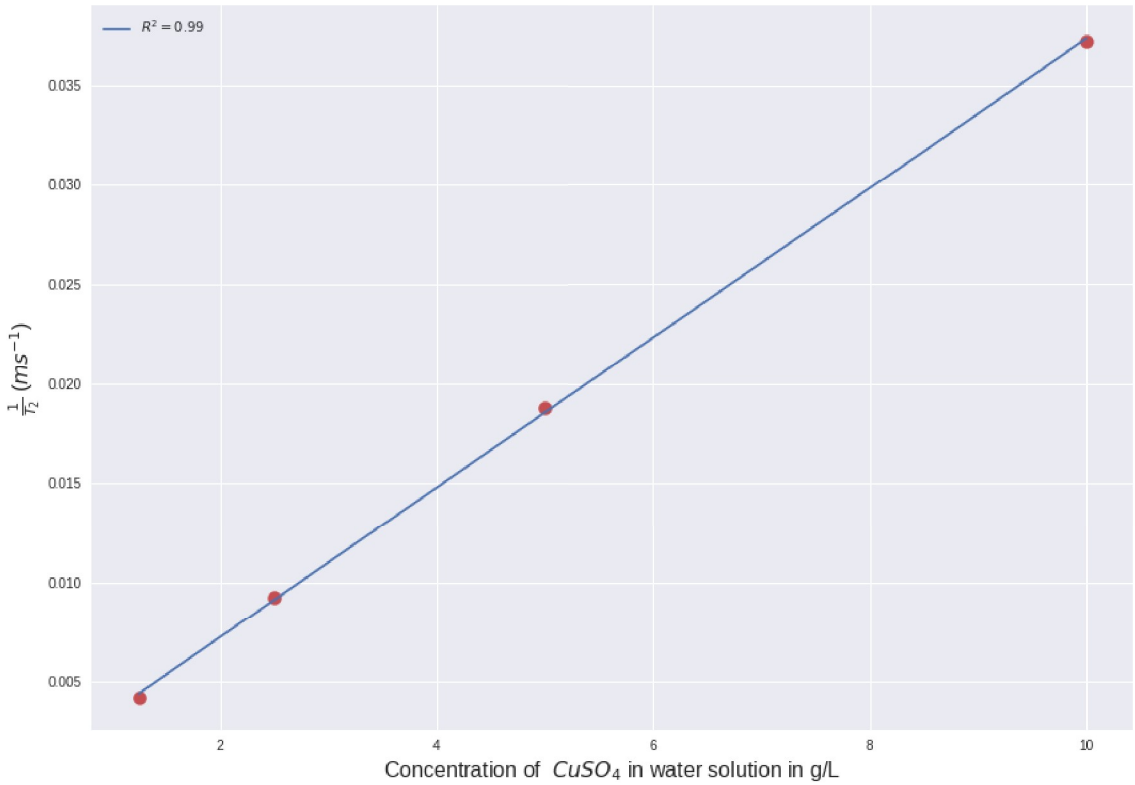

Figure 47 – Linear regression of four concentrations of copper sulfate in water and the inverse of the transverse relaxation time.

Source: By the author.

The linear relationship between the concentrations of copper sulfate in the water of the four different samples with the inverse of the transverse relaxation time is predicted by the literature. (15) Thus, the reproduction of this result is an indication that the DMRS developed by CIERMag is efficient equipment that is ready for use in NMR experiments.

5.7 Results from the measurement of T_1 from water using IR

The methodology described in section 4.3.7 was used to calculate the T_1 value from 1 ml of demineralized water at a temperature of 35ºC. Figure 48 shows the module from the magnetization value behavior during the experiment. Comparing the signal intensities from the beginning to the end of the experiment, it is possible to notice that the magnetization is fully recovered, and the inversion time chosen was sufficient.

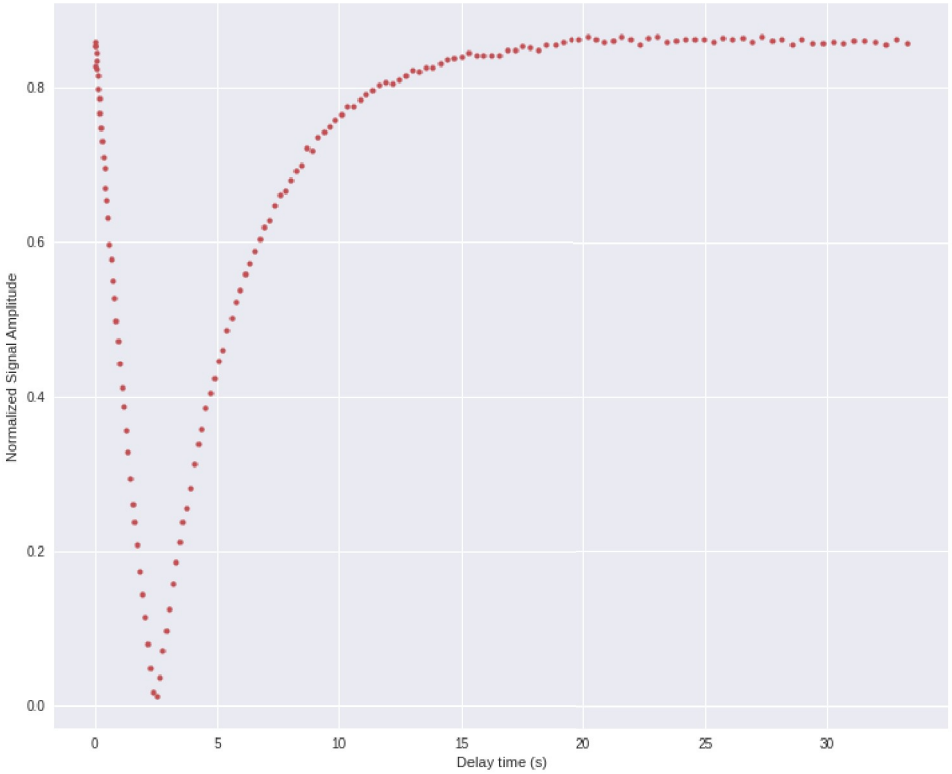

Figure 48 – Module from the signal magnetization behavior from water in the Inversion Recovery experiment.

Source: By the author.

Calculating the value of the longitudinal relaxation time using the equation 2.33 and making a statistic for 10 measurements, the value was found $T_1 = 3.58$ s with a 0.08 s standard deviation. The T_1 value was also obtained through the exponential fitting from the acquired data using the equation 2.23 as a reference for the fitting. A statistic was calculated for the same 10 acquisitions, the value found was equal to $T_1 = 3.49$ s with a 0.05 s standard deviation. Note that the two values found are in agreement when taking into account the standard deviation. The value found using the minimum signal has a slightly higher standard deviation, which is explained by a possible difference in the noise level between acquisitions. This can be explained by the fact that the noise has

more influence on the measurement using the signal minimum since for the fitting it is uniformly distributed to all points. The results are in accordance with the expected value of the longitudinal relaxation time from demineralized water at the given temperature (12). Figure 49 displays the exponential fitting for the water magnetization recovery.

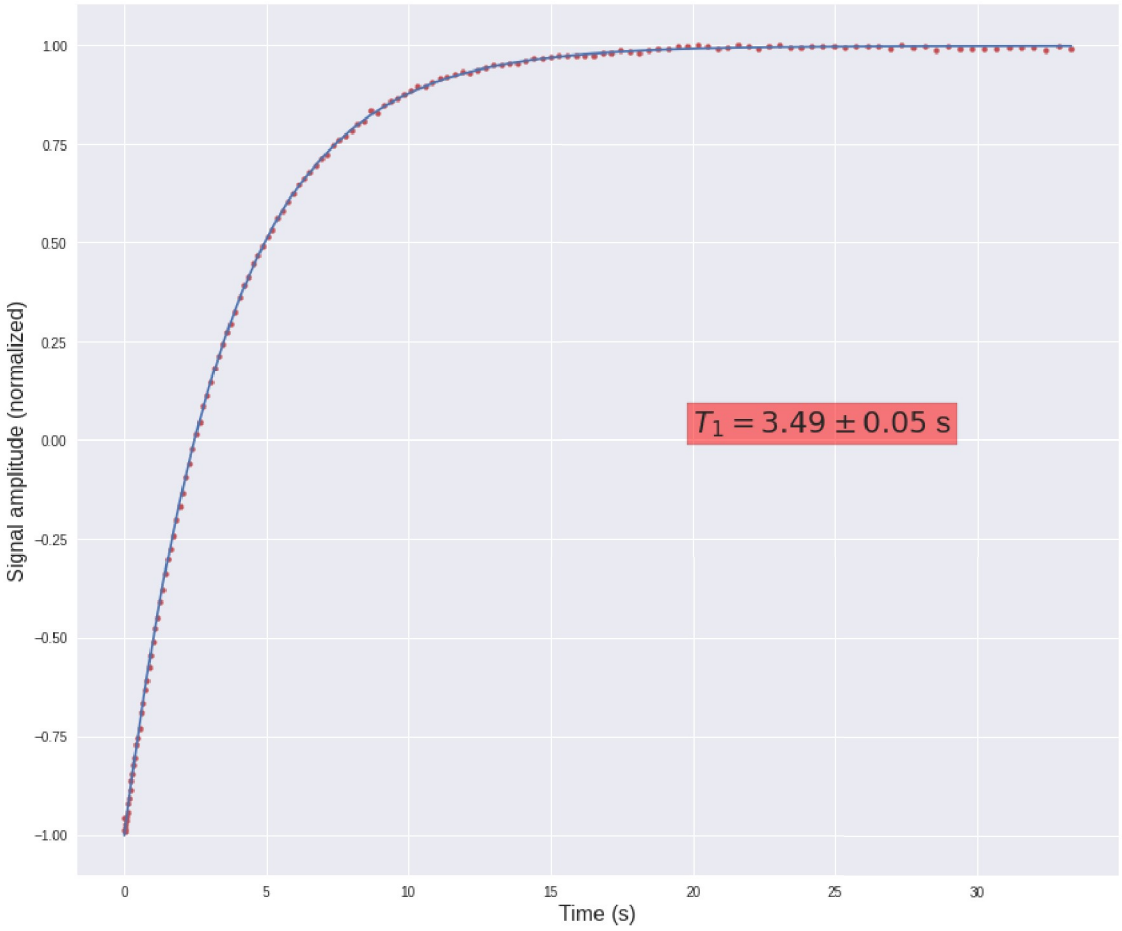

Figure 49 – Exponential fitting for the water magnetization recovering in the IR experiment.

Source: By the author.

5.8 Results from the measurement of T_1 from different concentrations of Copper Sulfate in water using IR

The methodology described in section 4.3.8 was used to measure the T_1 relaxation time for four solutions with different concentrations of water in copper sulfate. Table 8 displays the value of T_1 found for each concentration of water in copper sulfate.

Table 8 – Four different concentrations of water in copper sulfate in g/l and the corresponding T_1 value in ms

Concentração (g/L)	T_1 **(ms)**
1.25	231.9857 ± 5.4747
2.50	107.1201 ± 3.4216
5.00	58.4292 ± 0.0002
10.00	29.2145 ± 0.0014

Source: By the author.

Figure 50 displays the linear relationship between the inverse of the longitudinal relaxation time and the concentration of water and copper sulfate. The R^2 calculated for this fitting was 0.999 which means that the model represents 99.9% of the observed data.

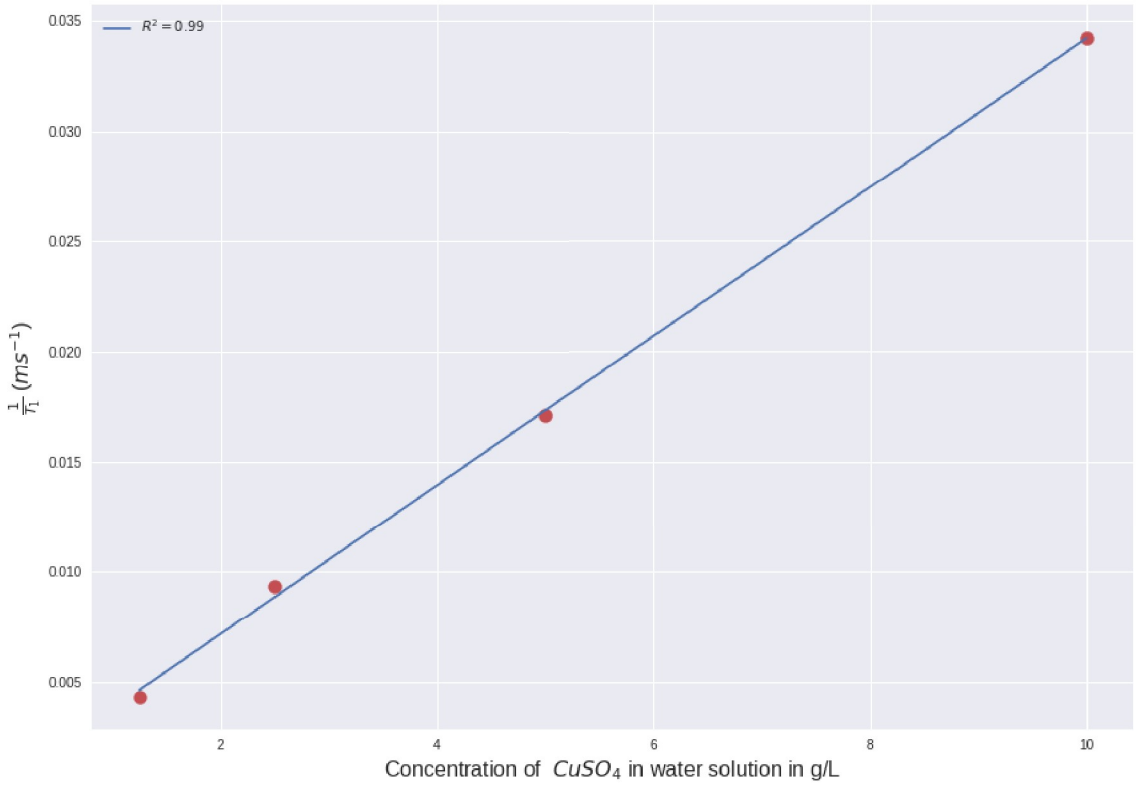

Figure 50 – Linear regression of four concentrations of copper sulfate in water and the inverse of the longitudinal relaxation time.

Source: By the author.

6 CONCLUSION

The focus of this work was the implementation of Magnetic Resonance methods compatible with the Digital Magnetic Resonance Spectrometer (DMRS) from CIERMag. In this sense, calibration methods were implemented for experiments performed on this equipment, such as adjusting the resonance frequency and the flip angle (B1 amplitude). In addition, well-known methods in the area of Magnetic Resonance Relaxometry, i.e. CPMG and IR, were implemented to measure longitudinal and transverse relaxation times. As the focus of the work was the implementation of the Methods in the CIERMag innovative equipment, the samples chosen were simple samples whose behavior was already known, in this way, the results obtained validated the effectiveness of the equipment and the implementation sought to show that the DMRS is ready for use and with basic functions for a routine of experiments implemented.

Two Methods for frequency adjustment were developed: *Find Resonance* and *Frequency Adjust*. There are no functional differences between the Methods, in the sense that both can be used to find the resonance condition. The difference between them is the accuracy, higher in the latter. The Frequency Adjust is more efficient in finding the fine resonance frequency, whereas Find Resonance can find the coarse resonance frequency. The flip angle (B_1 amplitude) results show that the Method is efficient and provides a well-calibrated flip angle. The results for the calibration methodologies comply with the pre-experiment adjustments necessary for measurement reliability.

Frequency Adjust was used to evaluate the effect of room temperature variation on the spectrometer operating frequency. The Method was run in the spectrometer for 16.6 hours and the ambient temperature was monitored during that time. The results show a relationship between temperature and frequency offset, with a decrease in room temperature and an increase in the spectrometer operating frequency. An approximately 2.5 °C variation in temperature generates a 70 ppm variation in frequency, which represents a 4.3 ppm/h average rate of change. This relationship is in accordance with the literature, once a permanent magnet is sensitive to temperature variations. This study has led to the conclusion the system is ideal for short measurements, due to the frequency variation in the long-time measurements.

The relaxation times of water at 35°C were calculated with the use of CPMG and Inversion recovery sequences. The values for the spin-spin and spin-lattice relaxation times are respectively $T_2 = 2.095 \pm 0.006$ s and $T_1 = 3.49 \pm 0.05$ s. In addition, as predicted by

the literature, we found linear relationships between the concentrations of copper sulfate in water for four different samples with the inverse of the transverse and longitudinal relaxation times.

All the results found for the relaxation times are in accordance with the expected for these samples. In this way, it is possible to conclude that the methods were implemented correctly as well as the experimental procedures. In addition, the DMRS can be considered ready to use and functional since the calibration Methods, which are essential for the proper functioning of the spectrometer towards ensuring the correctness of the experiments, are correctly implemented.

The CIERMag spectrometer and the structure for NMR sequence development is a good alternative to conventional standard equipment. This arrangement allows new techniques and new NMR Methods to be developed by researchers, in addition, to providing an easy-to-use and the F language gives the sequence designer a pictorial view of the sequence. In this way, this approach may represent an advantage for the development of NMR techniques and research.

REFERENCES

1 MINSKY N. W.; YÁÑEZ, A. S. Breve historia de la resonancia magnética nuclear: desde el descubrimiento hasta la aplicación en imagenología. **Educación Química**, v. 30, n. 2, p. 129, 2019.

2 RABI, I. I.; KELLOGG, J. M. B.; ZACHARIAS, J. R. The Magnetic Moment of the Proton. **Physical Review**, v. 46, n. 3, p. 157–163, 8 1934. DOI: 10.1103/PhysRev.46.157.

3 PURCELL, E. M.; TORREY, H. C.; POUND, R. V. Resonance absorption by nuclear magnetic moments in a solid [7]. **Physical Review**, v. 69, n. 1-2, p. 37–38, 1946. DOI: 10.1103/PHYSREV.69.37/FIGURE/1/THUMB.

4 BLOCH, F. Nuclear Induction. **Physical Review**, v. 70, n. 7-8, p. 460–474, 1946.DOI: 10.1103/PhysRev.70.460.

5 TANNUS, A. **Desenvolvimento da tecnologia de tomografia por ressonância magnética nuclear**, 1987. Tese (Doutorado) - Instituto de Física e Química de São Carlos, Universidade de São Paulo, São Carlos, 1987. DOI: 10.11606/T.54.1987.tde-21052007-164959.

6 MARTINS, M. J. **Desenvolvimento de um tomógrafo de ressonância magnética:** integração e otimização, 1995. Tese (Doutorado) - Instituto de Física de São Carlos, Universidade de São Paulo, São Carlos, 1995. DOI: 10.11606/T.76.1995.tde-04062007-150439.

7 PIZETTA, D. C. **Biblioteca, API e IDE para o desenvolvimento de projetos de metodologias de Ressonância Magnética**, 2014. 32 p. Dissertação (Mestrado em Ciências) - Instituto de Física de São Carlos, Universidade de São Paulo, São Carlos, 2014. DOI: 10.11606/D.76.2014.tde-28042014-160738.

8 TANNÚS, A.; GARWOOD, M. Adiabatic pulses. **NMR in Biomedicine**, v. 10, n. 8, p. 423–434, 8 1997. DOI: 10.1002/(SICI)1099-1492(199712)10:8<423::AID-NBM488>3.0.CO;2-X.

9 PIZETTA, D. C. **PyMR:** a framework for programming magnetic resonance systems, 2019. Tese (Doutorado em Ciências) - Instituto de Física de São Carlos, Universidade de São Paulo, São Carlos, 2019. DOI: 10.11606/T.76.2019.tde-06052019-103714.

10 SILVA, D. M. D. D. d. **Desenvolvimento de console multiplataforma para aquisição, organização e visualização de dados do espectrômetro digital de RM do CIERMag:** ToRM Console, 2014. 143 p. Dissertação (Mestrado em Ciências) - Instituto de Física de São Carlos, Universidade de São Paulo, São Carlos, 2014. DOI: 10.11606/D.76.2014.tde-24042014-172009.

11 COELHO F. B. *et al.* Gerador de eventos em lógica programável - abordagem com processador digital de sinais (DSP) configurável: linguagem de programação e compilador. *In*: IV Workshop do CInAPCe, 4., **Anais [. . .]**, São Carlos: IFSC, 2010.

12 TSUKIASHI, A. *et al.* Application of spin-crossover water soluble nanoparticles for use as MRI contrast agents. **Scientific Reports**, v. 8, n. 1, p. 14911, 2018. DOI: 10.1038/s41598–018–33362–6.

13 BROWN, R. W. *et al.* **Magnetic resonance imaging**. Chichester, UK: John Wiley & Sons, 2014. DOI: 10.1002/9781118633953.

14 VIDOTO, E. L. G. **Projeto de transdutores e otimização do sistema de recepção do tomógrafo de RMN de campo magnético ultrabaixo**, 1995. 123 p. Dissertação (Mestrado em Ciências) - Instituto de Física de São Carlos, Universidade de São Paulo, São Carlos, 1995. DOI: 10.11606/D.76.1995.tde-30012008-083030.

15 HAHN, E. L. Spin Echoes. **Physical Review**, v. 80, n. 4, p. 580–594, 1950. DOI: 10.1103/PhysRev.80.580.

16 CARR, H. Y.; PURCELL, E. M. Effects of diffusion on free precession in nuclear magnetic resonance experiments. **Physical Review**, v. 94, p. 630–638, 5 1954. DOI: 10.1103/PhysRev.94.630.

17 MEIBOOM, S.; GILL, D. Modified spin-echo method for measuring nuclear relaxation times. **Review of Scientific Instruments**, v. 29, n. 8, p. 688–691, 1958. DOI: 10.1063/1.1716296.

18 BAI, S. Nuclear Magnetic Resonance Instrumentation. *In*: **Encyclopedia of Analytical Chemistry**. Chichester, UK: John Wiley & Sons, 2016. DOI: 10.1002/9780470027318.a6108.pub3.

19 MONTRAZI, E. T. **Estudo de cerâmicas porosas de alumina através da medida de tempos de relaxação via ressonância magnética nuclear**,2012. Dissertação (Mestrado em Ciências) - Instituto de Física de São Carlos, Universidade de São Paulo, São Carlos, 2012. DOI: 10.11606/D.76.2012.tde-01062012-145924.

20 GRAAF, R. A. **In vivo NMR spectroscopy:** principles and techniques. 3rd ed. Chichester: Wiley, 2019. 577 p. DOI: 10.1002/9781119382461.

21 HOFFMANN, M. S. T. **Estabilidade térmica de ímãs compósitos à base de Nd-Fe-B**. 2021. 82 p. Dissertação (Mestrado) - Centro Tecnológico, Programa de Pós-Graduação em Ciência e Engenharia de Materiais, Universidade Federal de Santa Catarina, Florianópolis, 2021.

22 UNIVERSIDADE DE SÃO PAULO. Instituto de Física de São Carlos. Mateus José Martins; Edson Luiz Gea Vidoto; Alberto Tannús. **Espectrômetro para uso em sistemas de ressonância magnética.** BR1020150006241, 12 jan. 2015, 19 jun. 2016.

23 CENTRO de Imagens e espectroscopia por Ressonância Magnética (CIERMag). **"F" Language Reference Guide**. Não publicado.

24 MICROSOFT CORPORATION. **Microsoft visio**. 2019. Disponível em: https://products.office.com/en/visio/flowchart-software. Acesso em: 15 de dezembro de 2022.

25 HARRIS, C. R. *et al.* Array programming with NumPy. **Nature**, v. 585, n. 7825, p. 357–362, 2020.

26 VIRTANEN, P. *et al.* SciPy 1.0: fundamental algorithms for scientific computing in Python. **Nature Methods**, v. 17, n. 3, p. 261–272, 2020.

27 HELMUS, J. J.; JARONIEC, C. P. Nmrglue: an open source Python package for the analysis of multidimensional NMR data. **Journal of Biomolecular NMR**, v. 55, n. 4, p. 355–367, 2013.

28 CALLAGHAN, P. T. **Principles of nuclear magnetic resonance microscopy**. Oxford: ClarendonPress, 1993.

APPENDIX A – F - LANGUAGE

The tem at the CIERMag developed the F language that has features for programming pulse sequences used in MR experiments, several of them peculiar in the sense that they facilitate the development of new techniques. An important feature of this language is the possibility of having a pictorial view of the pulse sequence that is under development, to make the sequence programmer closer to the experiment. The F language has its own compiler and a Language Server Protocol (LSP).

In this appendix, the main elements of the F Language will be described and exemplified. A complete F Language description containing all the functionalities can be found in the internal documentation of CIERMag (Language Reference).

A.1 Language elements

It is possible to define the whole sequence structure- i.e. the events, modules, and the sequence command - in a single file, with the extension "xx.f". However, some objects such as modules and events can be reused and can be placed together in libraries, also named with the xx.f extension - i.e. SequenceEvents.f and SequeceModules.f. In a sequence file, it is then possible to reference those modules or events by importing them at the beginning of the file, as shown in Figure 51.

```
// Imports...
import usr.events.SequenceEvents;
import usr.modules.SequenceModules;
```

Figure 51 – How to import SequenceEvents.f and SequenceModules.f in the F code.

Source: By the author.

Once imported, all the events can be instantiated by the user in the sequence. The SequenceEvents.f library has a significant amount of events already developed that cover most of the requirements needed for NMR experiments. Events can have their functionality locally overridden - using the command OVERRIDE before the event name - inside a sequence without changing how they show in the event libraries. This simplifies their use in sequences allowing the preservation of the mnemonic effect of the symbol names internally defined.

In addition to events and modules, the F Language presents other important directives such as tables, delays, and different loop types. Some of these directives will be described in more detail below.

A.1.1 Modules

Modules control the flow of the sequence program, contain statements to repeat instructions, i.e., loops, perform conditional processing, and start events, and delays set the rotation matrix, and execute other operations. Figure 52 displays an example of a module in F Language.

```
module module_name(parameter_list):
    statement;
    statement; ...
end
```

Figure 52 – Example of a module in F Language.

Source: By the author.

A.1.2 Sequence Command

The sequence command is the entry point into a sequence program, they are used to select a single module as the entry point and set the parameters used by it. Figure 53 displays the format of the command.

```
module module_name(arg1, arg2, argument3, argument4, argument5):
    ...
end
sequence:
start module_name(0.5,
                  1.0,
                  argument3 = 0.3,
                  argument4 = 0.4,
                  argument5 = 254
                  );
end
```

Figure 53 – Format of the sequence command.

Source: By the author.

A.1.3 Tables

Tables are reusable structures that can be applied in different situations, for example, to define pulse shapes, gradient segments, storage of durations in some variable

delays experiments, and finally for the generation of rotation, specifically for transforming Logical Gradients (Readout, Phase Encoding, Slice Selection) in Physical ones (Gx, Gy, Gz) at a given orientation by using the rotation processor.

One special type of table used to index the Numerical Oscillator phase advance both for RF pulses and for readout, is the phase table, the only one that admits numerical and symbolic values, as we will see later in this section. Tables can be defined in files with the extension ".tbl" and loaded into the code or can be defined intrinsically in the code. Figure 54 shows examples of table definitions.

```
// Orientation Tables
table PlaneOrientation = file("protocols/execution/OrientSet.tbl");
 or
table PlaneOrientation = { 1.0, 0.0, 0.0, 0.0, 1.0, 0.0, 0.0, 0.0, 1.0 }; // Identity...
```

Figure 54 – Example showing the two possible forms of defining tables in Language F: using files with the extension .tbl or defining the tables intrinsically in the code.

Source: By the author.

Tables must be declared/loaded in the context in which they will be used,i.e., a table used by a given module must be declared/loaded in the same file that the module was declared, and a table used by a given event must be declared/loaded in the same file that the event was declared. An important property of tables in the F language is cyclic redundancy, this means that, once tables are defined with a value, that value is repeated cyclically for all necessary points. The rotation command associates the logical gradients with physical gradients. The rotations or orientation tables, which are also cyclical, must be defined in multiples of 9 elements. This happens because each element of the rotation is a 3x3 matrix. Thus, to define a table with two rotation elements, you need two 3x3 matrices and 18 elements. Most indexing tables that directly affect the spectrometer processors' registers can only admit float in the interval (-1.0, 1.0), although other tables can assume any value in their elements.

A.1.4 Real-time variables

The real-time variables represent a tool to modify the conditions under which the experiment is being carried out in real-time. That is, there is no need to wait for the experiment to finish before changing the conditions and then restarting it. The real-time variables can be modified on the console while the Method runs, if the Method is being executed, it waits for the end of the cycle and updates the variable. To define the Real Time variables we use the rtvar stroke, like in Figure 55.

```
rtvar RT_TxFMOffset = { 0.0 };
rtvar RT_RxFMOffset = { 0.0 };
```

Figure 55 – Definition of a real-time variable in F Language.

Source: By the author.

A.1.5 Rotation

The rotation command represents the rotation processor responsible for transforming physical gradients into logical gradients in the following order: GRead = Gx, GPhase = Gy, and GSelection = Gz. It uses the rotation matrix as an argument and an expression that relates the current slice to the size of the rotation matrix. Figure 56 displays an example of the use of the rotation command with the rotation tables.

```
table rotation_matrices = { 1, 0, 0,  0, 1, 0,  0, 0, 1,  1, 0, 0,  0, 0, 1,  0, 1, 0 };
module module_name():
    loop I = 0, 1:
        rotation(rotation_matrices[9 * i]);
        statement;
        statement;
        ...
    end
end
```

Figure 56 – Example of using the rotation command.

Source: By the author.

A.1.6 Loop Types

In the F language there exist different types of loops and each one has a specific functionality. The loop types nesting order is almost free which is a very useful tool since it allows the sequence designer to select the format of the data structure. Figure 57 shows a generic definition of a loop that contains the start, end, the step size - which must always be a positive value, and the loop type.

The possible loop types will be described below:

- **Average:** Additively process internal acquisition command data, no matter how many other loops it contains. Consequently, it does not generate any additional dimensions, being itself collapsed on average.

```
loop variable=start,end [,step] [,type=type_description] :
    statement;
    statement;
    ...
end
```

Figure 57 – Generic definition of loop in F Language.

Source: By the author.

- **catAcq:** When this loop type is used, the Console will iteratively concatenate any acquisition command inside it until the end of the loop. The result is equivalent to a single acquisition with a total number of data equal to the number of concatenations multiplied by the number of individual data acquisitions.
- **catAvgAcq:** Very similar to the catAcq, but it concatenates the average and consequent collapse into a single point, the final number of data points will be the number of concatenations.
- **dim2 or phEnc2 or phEnc:** place the collection of data acquired in this loop along the second dimension of the data matrix.
- **dim3 or phEnc3 or slice:** According to the classical multislice acquisition it populates the third dimension of the dataset.
- **dim4 or echo:** This type is reserved for acquisitions where the concept of echo is dominant - i.e. in non-imaging sequences such as CPMG. The main iteration is to produce multiple echoes.
- **dim5 or evolution:** Same meaning as discussed before in dim4, for the main type of dimension 5.
- **ordinary:** Used when the sequence designer wants to evolve certain entities such as the amplitude of a given gradient and requires the use of a loop variable to do so. As long as no acquisition is involved in the process (inside the loop), its use is allowed. In that case, since we made no acquisition within, the loop structure parser in the compiler generates no dimension in the dataset. This is the default loop type, assumed when no loop type is explicitly declared.

In addition to the loop types presented above, another important loop type in Language F is the Join loop. The join type is an F language mechanism to preserve the TR value stipulated by the sequence designer. When users define many slices, most existing commercial programs end up modifying the value of TR to perform the execution, however, TR is a parameter that defines contrast, so once defined it should not be modified. In the F language, join loops offer an alternative so that does not happen by creating “partitions”

in the number of slices. Thus, it is possible to have 16 slices and have the program execute the events for 4 slices ate time, in 4 join steps. In this way, Language F still relies on loops: joinDim3 or joinphEnc3 or joinSlice, joinDim4 or joinEcho, and joinDim5 or joinEvolution.

A.1.7 Delays

The delay command stops processing of all lines for a specified time, line outputs are blocked and any instruction that produces outputs will not finish execution before the end of the delay time. The resulting time is computed as if expressed in microseconds, with resolution determined by the Timing Sequencer clock period of 20 ns. In the validation file, the delay values were verified to ensure that they are multiple of 20 ns and positive. To define a delay in the code, it is necessary to use the command delay followed by a delay expression, Figura 58 show different forms for delay definition and how to insert them in the code.

```
DL0 = 0.0 us; //Observe that the unit is necessary to state this variable as a time interval
loop Fid = 0, NumFIDs-1, type = echo:
    delay DL1 = InterFID[Fid];
    start ...;
    DL0 = DL0 + DL1;
end // Fid
delay DL4 = TR - (tTrig + ... + DL0)
```

Figure 58 – Different forms for delay definition inserted in the F code.

Source: By the author.

A.1.8 Events

Events are a collection of lines, each line describes a waveform during the event duration for a particular output. Events may contain several segments, and all the lines must show the same total duration. Events occur one after the other, so for a new event to be executed, the previous one must be terminated. Figure 59 displays the general structure of an event.

The following lines that generate analog outputs are available:

- **G_READ , G_SEL , G_PHASE:** These are read, selection, and phase gradients. These lines have aliases GR, GS, and GP respectively.
- **AM_1 , AM_2, AM_3, AM_4:** These are the AM modulators in main transmitters 1, 2, 3 and 4;

```
event event_name(parameter_list):
    line LINE_NAME:
        statement;
        ...
    end
    line OTHER_LINE:
        statement;
        ...
    end
    ...
end
```

Figure 59 – General events structure in F Language.

Source: By the author.

- **FM_1 , FM_2, FM _3, FM_4:** These are the FM modulators in main transmitters 1, 2, 3 and 4;
- **AM_PAR, AM_PAR1, AM_PAR2, AM_PAR3, AM_PAR4; FM PAR, FM_PAR1, FM_AR2, FM_PAR3, FM_PAR4, PM_PAR, PM_PAR1, PM_PAR2, PM_PAR3, PM_PAR4:** These are the lines to be used when the parallel transmission is required.
- **ACQ_1, ACQ_2, ACQ_3, ACQ_4, ACQ_PAR, ACQ_PAR1, ACQ_PAR2, ACQ_PAR3, ACQ_PAR4**: These lines are used to set acquisition devices. Setting it to 1 enables acquisition, and 0 disables it.
- **RXUNB_1, RXUNB_2, RXUNB_3, RXUNB_4, RXUNB_PAR; RXUNB_PAR1, RXUNB_PAR2, RXUNB_PAR3, RXUNB_PAR4; TXUNB_1, TXUNB_2, TXUNB_3, TXUNB_4, TXUNB_PAR; TXUNB_PAR1, TXUN_PAR2, TXUNB_PAR3, TXUNB_PAR4:** These lines control the unblank gates of the receivers (blanked by default) and the unblank gates of the transmitters (normally shut off).
- **AM_@, FM_@, PM_@, ACQ_@, TXUNB_@, RXUNB_@:** This is a special construct idealized to allow the sequence designer (and the end user) to choose the transmitter and receiver channels at execution time.
- **FETrIg, FEAux, RXTXSEC, FlagsON:** By managing these lines the user can access bits that are available either inside or outside the spectrometer.

The command start precedes the events and indicates that they should start. An example of an event is shown in figure 60.

```
event PulseHard( tHard, AMamp, Offset, Phase ):

  line AM_1:
    constant 0.0 for C_tPre;
    constant AMamp for tHard - C_tPre - C_tPos;
    constant 0.0 for C_tPos;
  end

  line FM_1:
    constant Offset for tHard;
  end
  .
  .
  .
```

Figure 60 – Example of the event *PulseHard* in F Language.

Source: By the author.

A.2 Sequence Illustration

To help the sequence designer represent the sequences graphically, stencils that contain all the F language structures were developed by the CIERMag team in Microsoft Visio 2019 (24). The Visio stencils for drawing NMR pulse sequences developed by CIERMag are available in the CIERMag internal documentation. Microsoft Visio is an application for creating diagrams for the Windows environment. The program is used to generate diagrams of different types, such as organization charts, flowcharts, data modeling, network diagrams, floor plans, posters, etc.

The structures in Visio stencils for drawing NMR pulse sequences are categorized and organized into different stencil types. For example, “Temporal” stencils represent events related to time, such as delays, and also related to hardware, like phase reset, transmitter, and receiver. “Trapezoidal Gradient stencil”, “Sinusoidal Gradient stencil”, and “Rectangular Gradient stencil” represent respectively the trapezoidal, sinusoidal, and rectangular gradient forms. Figure 61 displays examples of these structure representations.

In Visio stencils for drawing NMR pulse sequences, colors differentiate time scales. The green structures represent events with a duration of microseconds, the purple structures events with a duration of milliseconds and the yellow ones represent events with a duration of seconds. The Visio Stencils for NMR Pulse Sequence help F Language programmers visualize and understand all the structures related to the hardware control that appears in the sequence code. All the sequence diagrams elaborate in this dissertation were made using Visio.

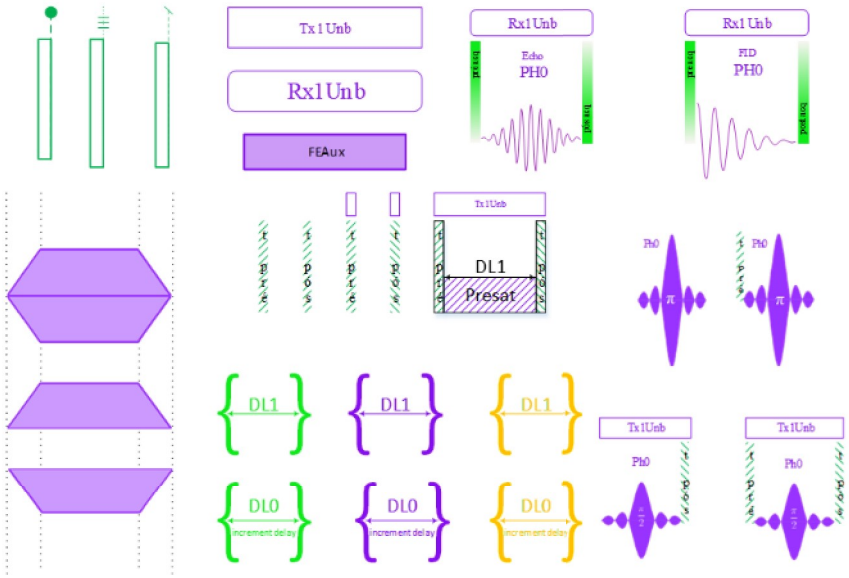

Figure 61 – Examples of structures from Visio stencils for drawing MR sequence pulses.

Source: By the author.

A.3 Examples

For a better understanding of Language F, the construction of some events - used in the pulse sequences developed for this work - with illustrations and the code will be shown below

- **GenExcPulse:** This event is responsible for sending a generic RF pulse, i.e., the pulse shape, amplitude, phase modulation, and frequency can be defined by the user in the PyMR interface. Figure 62 displays the F code for the GenExcPulse event. The event receives the parameters tExc, AMamp, Phase, and Offset, and use four lines, AM_1, PM_1, FM_1 and TXUNB_1. Lines AM_1, PM_1, and FM_1 are related to the amplitude, phase, and frequency modulation of the pulse. The line AM_1 uses the pulse shape uploaded by the validation file in the GenAMShape multiplied by the parameter AMamp that represents the pulse amplitude during. On the PM_1 line, something similar to the AM_1 line occurs, however, the generic waveform table GenPMShape is used plus the Phase variable. The frequency modulation line FM_1 remains constant for the entire time of the event. The last line used is the TXUNB_1 line responsible for opening the transmission gates at the correct time.

```
event ExcPulse(tExc, AMamp, Phase, Offset):
        line AM_1::
                Constant OFF for C_tPre;
                tabel GenAMShape*AMamp  for tExc - C_tPre - C_tPos;
                Constant OFF for C_tPos;
        end
        line PM_1:
                Constant OFF for C_tPre;
                tabel GenPMShape + Phase  for tExc - C_tPre - C_tPos;
                Constant OFF for C_tPos;
        end
        line FM_1:
                Constant Offset for tExc;
        end
        line TXUNB_1:
                Constant ON  for tExc;
        end
end
```

Figure 62 – F code for the GenExcPulse event.

Source: By the author.

The parameter tExc represents the total duration of the event, in this way, the duration of each line is equal to tExc. In Figure 63 it is possible to observe the durations of each different part of the event.

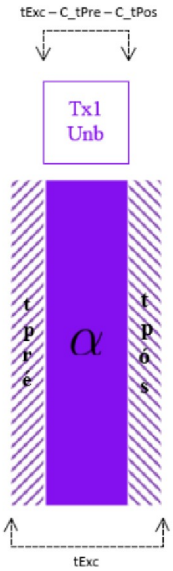

Figure 63 – Representation of the GenExcPulse event

Source: By the author.

- **Readout:** The Readout event is an event developed to be used only for acquisitions without gradients. This event receives the parameters tRead, RxFMOffset, and RxPMamp and uses the lines ACQ_1, PM_1, FM_1, REUNB_1, and FEAux. Line ACQ_1 is used to set acquisition device 1. Line RXUNB_1 controls the unblanking of receiver 1 and line FEAux is responsible for triggering the Echo Synthesizer. Figure 64 displays the F code for the Readout event.

```
event Readout(tRead, RxFMOffset, RxPMamp):
        line ACQ_1::
                Constant OFF for C_tPreACQ;
                Constant ON  for tRead - C_tPreACQ -C_tPosACQ;
                Constant OFF for C_tPosACQ;
        end
        line PM_1:
                Constant RxPMamp for tRead;
        end
        line FM_1:
                Constant RxFMOffset for tRead;
        end
        line RXUNB_1:
                Constant ON  for tRead;
        end
        line FEAux:
                Constant ON  for tRead;
                Constant ON*C_SybthSignal for tRead - C_tPreACQ - C_tPosACQ;
                Constant OFF  for C_tPosACQ;
        end
end
```

Figure 64 – F code for the Readout event.

Source: By the author.

The parameter tRead represents the total duration of the event, in this way, the duration of each line is equal to tRead. In Figure 65 it is possible to observe the durations of each different part of the event.

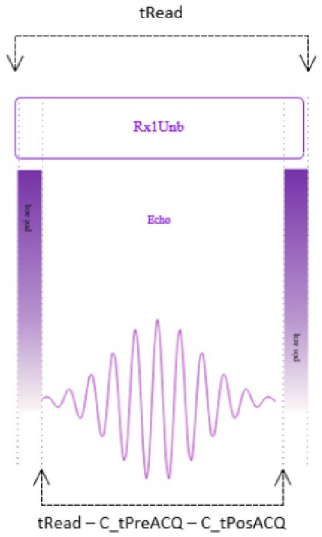

Figure 65 – Representation of the Readout event.

Source: By the author.