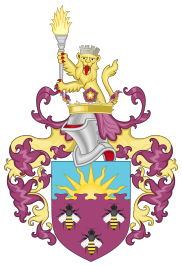
**Master's Dissertation**

# Signal processing & pulse control in Earth's field NMR spectrometry

*as part of the ongoing development of a portable, low-cost Earth's field NMR spectrometer.*



*Author:* Sajid Ahmed

*Supervisor:* Dr Patrick Parkinson

*In fulfilment of the requirements for the degree of Integrated Master's of Physics*

*in the*

**Department of Physics and Astronomy**

**The University of Manchester**

December 2024

**Acknowledgements:** The experimental work for this project was conducted in collaboration with Emma Hattersley, George Champion, and Joseph McAuley under the supervision of Dr Patrick Parkinson at the University of Manchester.

## Abstract

Earth's-field nuclear magnetic resonance (EFNMR) trades signal strength for portability and a price tag two to three orders of magnitude lower than high-field NMR, which permits use cases that high-field instruments cannot serve, such as hands-on physics education in schools. This report presents the pulsing half of the latest iteration of the departmental low-cost EFNMR project, which targets a portable, sub-£250 spectrometer for use in schools and colleges. Three investigations were carried out: an evaluation of four receiver-coil configurations within the pre-polariser, the design and characterisation of a smoothed-square-wave pulsing circuit as a simpler alternative to the dedicated sine-wave drive used by previous cohorts, and a comparison of C++ and MicroPython firmware on the Raspberry Pi Pico 2 microcontroller.

A counter-wound double-coil configuration was selected, matching the noise floor of a single coil while doubling the usable sample volume to 580 mL. A smoothed-square-wave drive, produced by passing the microcontroller's output through a simple RLC resonant circuit (0.15 µF, 47.4 mH), delivered sufficient power at the 2103 Hz proton Larmor frequency to drive the CPMG sequence, establishing it as a viable lower-cost alternative to a dedicated sine-wave drive. Upgrading the microcontroller to the Raspberry Pi Pico 2 and replacing the inherited MicroPython firmware with a compiled C++ equivalent reduced per-pulse overhead from 347.1 µs to 23.5 µs (a 93.2% reduction) and brought the mean pulse frequency to within 40 Hz of the Larmor target, though end-to-end power-transfer efficiency between the two firmware languages improved only marginally.

The assembled spectrometer met the cost target at £178.65 with full CPMG implementation on the pulsing side; portability and detection-circuit integration remain outstanding. The headline contribution is the first demonstration within the departmental project of the CPMG sequence using a C++ based firmware for the microcontroller, substantially reducing the performance overheads and bringing the CPMG pulse sequence frequency drastically closer to the targeted Larmor frequency.

# Contents


Abstract . . . . . 1

**1 Introduction** **6**

1.1 Nuclear Magnetic Resonance . . . . . 6

1.2 Earth's-Field NMR . . . . . 7

1.3 Aims and Objectives . . . . . 7

1.3.1 Motivation . . . . . 7

1.3.2 Design requirements . . . . . 8

1.3.3 Hypotheses . . . . . 8

1.3.4 Scope of this report and report structure . . . . . 9

**2 Theory** **10**

2.1 Quantum Mechanical Foundations . . . . . 10

2.2 Spin echo . . . . . 13

2.2.1 Hahn echo . . . . . 14

2.2.2 Carr-Purcell-Meiboom-Gill sequence . . . . . 15

2.3 Fourier transform . . . . . 17

2.4 Signal noise . . . . . 18

2.4.1 Environmental Electromagnetic Interference . . . . . 18

2.4.2 Johnson-Nyquist noise . . . . . 19

2.4.3 Shot noise . . . . . 20

2.4.4 1/f Noise . . . . . 21

2.4.5 Geomagnetic and Environmental Field Fluctuations . . . . . 22

2.4.6 Mechanical (Vibrational) Noise . . . . . 22

**3 Methodology** **23**

3.1 Apparatus design . . . . . 23

3.2 Polarising field strength and homogeneity . . . . . 24

3.2.1 Method for measuring field homogeneity . . . . . 24

3.3 Noise in the receiver coils . . . . . 25

3.3.1 Method for measuring noise levels . . . . . 26

3.4 Power Transfer Efficiency . . . . . 27

3.4.1 Method for measuring Power Transfer Efficiency . . . . . 27

3.5 Hardware and firmware optimisation . . . 28
3.5.1 Method for testing hardware & firmware speed . . . 29
3.5.2 Method for measuring pulse frequency deviation . . . 30

**4 Results and Discussion** **31**
4.1 Project aims and overall outcomes . . . 31
4.2 Coil configurations . . . 32
4.2.1 Hypothesis . . . 32
4.2.2 Magnetic field strength and homogeneity in the pre-polariser . . . 32
4.2.3 Noise amplitudes across the four configurations . . . 34
4.2.4 Configuration choice . . . 35
4.2.5 Limitations . . . 35
4.2.6 Next steps . . . 36
4.3 Square-wave pulsing . . . 37
4.3.1 Hypothesis . . . 37
4.3.2 Power transfer efficiency vs. smoothing capacitance . . . 37
4.3.3 Comparison with the sine-wave approach . . . 38
4.3.4 Limitations . . . 38
4.3.5 Next steps . . . 38
4.4 Pulsing code and microcontrollers . . . 39
4.4.1 Hypothesis . . . 39
4.4.2 Hardware comparison: Pico 1 vs. Pico 2 . . . 39
4.4.3 Firmware comparison: MicroPython vs. C++ . . . 40
4.4.4 Limitations . . . 41
4.4.5 Next steps . . . 42
4.5 Implications for educational outreach . . . 42

**5 Conclusion** **44**

**Bibliography** **49**

**A Appendix - Table** **50**

**B Appendix - Figures** **51**

**C Appendix - Code** **53**
C.1 C++ Microcontroller Code . . . 53
C.2 MicroPython microcontroller firmware . . . 57

# List of Figures

3.1 Schematic of the NMR apparatus design. . . . . 23
3.2 3D rendering of the pre-polariser and reciever coils design (made by Michal Wlodarczyk[18]). . . . . 23

4.1 Magnetic field strength and homogeneity in the pre-polariser. Error bars correspond to 1.96 standard deviations of the measured field strength at each position (95% confidence interval). . . . . 33
4.2 Mean and maximum noise peak amplitudes for the four tested coil configurations. Error bars represent the standard error of the mean. . . . . 34
4.3 Power transfer efficiency vs. smoothing capacitance for the resonant pulsing circuit. Error bars correspond to 1.96 standard deviations (95% confidence interval); for most data points they are too small to be visible on the graph. . 37

B.1 Fourier transform of noise floor data for counter-wound configuration . . . . 51
B.2 Schematic showing tested coil arrangements . . . . . 51
B.3 Previous cohort's pulsing circuit[17–20,22] . . . . . 52

# List of Tables

1.1 Design requirements for the EFNMR spectrometer. . . . . 8

4.1 Evaluation of the final spectrometer against the design requirements set out in the project aims. . . . . 31
4.2 Comparison of overhead times and overhead percentages for MicroPython and C++ on the Pico 1 and Pico 2 microcontrollers. . . . . 39
4.3 Comparison of pulsing performance between MicroPython and C++ on the Raspberry Pi Pico 2. . . . . 40

A.1 Summary of components used in spectrometer design and their costs. . . . 50

# List of Equations


2.1 Spin quantum number . . . 10
2.2 Spin angular momentum . . . 10
2.3 Spin energy levels . . . 11
2.4 Boltzmann distribution . . . 11
2.5 Net magnetisation . . . 11
2.6 Magnetic moment torque . . . 11
2.7 Larmor angular frequency . . . 12
2.8 T1 decay . . . 12
2.9 T2 decay . . . 12
2.10 T2* relation . . . 13
2.11 Hahn echo pulse sequence . . . 14
2.12 Spin precession phase . . . 14
2.13 Spin echo amplitude . . . 14
2.14 Bloch equation (x axis) . . . 14
2.15 Bloch equation (y axis) . . . 15
2.16 Bloch equation in complex form . . . 15
2.17 Bloch equation solution . . . 15
2.18 Carr-Purcell sequence . . . 15
2.19 Carr-Purcell-Meiboom-Gill sequence . . . 16
2.20 CPMG echo amplitude . . . 16
2.21 CPMG echo amplitude with diffusion . . . 16
2.22 CPMG echo amplitude with diffusion as a function of time . . . 16
2.23 Carr-Purcell-Meiboom-Gill effective decay rate . . . 16
2.24 Fourier transform . . . 17
2.25 Inverse Fourier transform . . . 17
2.26 Faraday's law . . . 18
2.27 Induced electromotive force . . . 19
2.28 Solenoid magnetic flux . . . 19
2.29 Nyquist formula . . . 19
2.30 Johnson-Nyquist noise RMS voltage . . . 19
2.31 Schottky formula . . . 20
2.32 Shot noise RMS current . . . 20

2.33 SNR of shot-noise-limited counting . . . . . 21
2.34 Flicker noise . . . . . 21
3.1 Pulsing overhead time . . . . . 30

# Chapter 1

# Introduction

Nuclear Magnetic Resonance (NMR) is one of the most powerful techniques in modern chemistry and medicine, but the high-field spectrometers that perform it cost upwards of a million pounds and weigh several tonnes[1]. Earth's-field NMR (EFNMR) offers an alternative that trades signal strength for portability, simplicity, and a price tag two to three orders of magnitude lower, opening the door to applications that high-field instruments cannot serve such as hands-on physics education in schools.

This chapter introduces the physics of NMR (Section 1.1) and the specific characteristics of its Earth's-field variant (Section 1.2), and then sets out the aims, design requirements, and hypotheses for our contribution to the ongoing departmental EFNMR project (Section 1.3).

## 1.1 Nuclear Magnetic Resonance

NMR exploits the precession of nuclear spins in an applied magnetic field to perform non-destructive analysis of materials and chemical samples. A nucleus with spin quantum number $s = \frac{n}{2}$ placed in a static field $B_0$ occupies one of two energy eigenstates separated by $\Delta E = \gamma \hbar B_0$, where $\gamma$ is the gyromagnetic ratio of the nucleus. The foundational example of this is the proton, or a $^1H$ nucleus. The net magnetisation precesses about $B_0$ at the Larmor frequency $\omega_0 = \gamma B_0$; a radiofrequency pulse tuned to $\omega_0$ flips the magnetisation vector into the transverse plane, and the subsequent free precession induces a measurable voltage in a detection coil. Further theoretical detail relevant to this project including the values of $\gamma$ for $^1$H, the role of the CPMG pulse sequence in extracting the transverse relaxation time $T_2$, and the calculation of the Larmor frequency for water in the Earth's field in Manchester, UK (the location of the research laboratory) is developed alongside the apparatus design in Chapter 3.

The technique was first demonstrated in 1938 by Rabi, who measured nuclear resonance in molecular beams as an extension of the Stern-Gerlach experiment[2], and was extended to liquids and solids by Bloch and Purcell in 1946[3]. It has since become a mainstay of two

distinct fields. In chemistry, NMR spectroscopy resolves the local environments of individual nuclei from their resonance shifts, allowing molecular composition to be reconstructed; recent examples include the characterisation of ion dynamics in battery cathode materials[4] and microstructural studies of polymers[5]. In medicine, NMR underpins magnetic resonance imaging (MRI), in which contrast between the $T_1$ and $T_2$ relaxation times of different tissues produces the soft-tissue images used in routine diagnosis[6].

## 1.2 Earth's-Field NMR

Earth's-field NMR was first demonstrated in 1954 by Packard and Varian, who used a 20 mT pre-polarising field to obtain a proton NMR signal from water in the geomagnetic field[7]. The technique has since found commercial application in detecting petroleum and water reserves in geological formations[8] and in polymer characterisation[5], and has been deployed in field environments inaccessible to high-field instruments, including in situ measurements of brine content in Arctic sea ice[9].

The appeal of EFNMR is straightforward. Removing the superconducting magnet eliminates by far the most expensive and environmentally demanding component of a conventional spectrometer[10]. Commercial EFNMR teaching instruments are available for £5,000–£10,000[11], against the >£1 million typical of a high-field research system[1], and bench-top EFNMR devices have been built from off-the-shelf components for around £250[12]. The resulting instruments are small, robust, and operable outside controlled laboratory environments.

The cost of these advantages is signal. Research-grade high-field NMR systems typically operate at 6 to 23.5 T[13], with the highest commercially available instruments reaching approximately 28.2 T[14], whereas the Earth's field varies from roughly 25 to 65 µT depending on geographic location[15] - a difference of five to six orders of magnitude. In a direct NMR experiment the induced signal scales as $B_0^2$: one factor of $B_0$ from the thermal-equilibrium magnetisation, and one from the Larmor frequency at which that magnetisation is detected[16]. EFNMR setups recover some signal by pre-polarising the sample in a higher field before detection in Earth's field, decoupling magnetisation from detection frequency[10], but the resulting signal-to-noise ratio remains low and managing it is the central engineering challenge of every EFNMR design.

## 1.3 Aims and Objectives

### 1.3.1 Motivation

The strategic aim of this MPhys project, ongoing since 2020[17–22], is to develop a portable, low-cost EFNMR spectrometer suitable for educational outreach in local high schools and

colleges. NMR physics is excellent for teaching as it brings together spin, quantisation, precession, resonance, and relaxation in a single experiment, but is normally invisible to school-age students because of the cost and complexity of the instruments required. A portable, working spectrometer that fits in a car and runs on battery power gives students direct contact with apparatus they would otherwise only encounter in textbooks, and demonstrates that frontier physics is not the exclusive preserve of multi-million-pound facilities. The project was originally inspired by Michal's 2010 demonstration that a functional EFNMR instrument can be built for around $250[12].

### 1.3.2 Design requirements

The design targets that follow from this aim are summarised in Table 1.1.

Table 1.1: Design requirements for the EFNMR spectrometer.

| **Requirement** | **Specification** |
|---|---|
| Total cost | ≤£250.00 |
| Footprint | $\lesssim 1.0\,\mathrm{m}^2$ |
| Portability | Transportable in a standard passenger vehicle; operable outdoors |
| Time resolution | ∼µs |
| Pulse sequence | CPMG |
| Operability | Usable by non-experts |

### 1.3.3 Hypotheses

Building on the design inherited from previous cohorts, we identified two subsystems where targeted improvements were most likely to yield measurable gains: the pulsing circuit and the amplification/detection chain. The specific hypotheses tested under the pulsing half of the project, which is the focus of this report, are:

1. That re-arranging the receiver coils within the pre-polariser to place the sample in the region of strongest and most uniform field, and adopting a counter-wound double-coil arrangement, will increase the spin-echo signal amplitude and gradiometrically cancel ambient external magnetic interference at the detection coil.

2. That square-wave pulses generated directly by the microcontroller, smoothed through a simple RLC resonant circuit, can deliver sufficient power at the proton Larmor frequency to drive the CPMG sequence, replacing the dedicated sine-wave generation circuit used by previous cohorts at substantially lower component cost and complexity.

3. That upgrading the pulsing microcontroller from the Raspberry Pi Pico (RP2040) to the Raspberry Pi Pico 2 (RP2350), and replacing the inherited MicroPython firmware

with a compiled C++ equivalent, will reduce per-pulse overhead and bring the mean pulse frequency closer to the proton Larmor target.

### 1.3.4 Scope of this report and report structure

The project was carried out by a team of four students under the supervision of Dr Parkinson. Emma Hattersley and I investigated the pulsing circuit; George Champion and Joseph McAuley investigated the amplification and detection circuits, the results of which are reported separately[23–25]. The present report is concerned exclusively with the pulsing-side investigations.

The remainder of the report is organised as follows. Chapter 2 details the quantum mechanics underlying NMR including Larmor precession, spin echo, Hahn echo and CPMG pulsing. It also describes the Fast Fourier Transform algorithm and potential sources of signal noise, both of which play a major role in EFNMR signal processing. The Chapter 3 describes the experimental apparatus and the methods used to test each of the hypotheses listed above, alongside the theoretical background where it is needed for context. Chapter 4 presents and discusses the results of those investigations and evaluates the spectrometer against the design requirements of Table 1.1. Chapter 5 synthesises the contribution and identifies next steps for the following cohort.

# Chapter 2

# Theory

## 2.1 Quantum Mechanical Foundations

The theoretical physics underpinning Nuclear Magnetic Resonance relies on the quantum phenomenon of Larmor precession. Larmor precession is the precession of the magnetic moment vector of a particle about the direction of an external magnetic field. It is analogous to the classical phenomenon of a gyroscope precessing about its axis due to the torque exerted onto it by the gravitational field.

In quantum mechanics, elementary particles possess 'spin', which is a component of the total angular momentum of the particle alongside the orbital angular momentum. The spin quantum number, $s$, takes values

$$s = \frac{1}{2}n \tag{2.1}$$

where $n \in \mathbb{Z}_{\geq 0}$, so that $s \in \{0, \frac{1}{2}, 1, \frac{3}{2}, \ldots\}$. The magnitude of the spin angular momentum $S$ is then

$$S = \hbar\sqrt{s(s+1)} \tag{2.2}$$

where $\hbar = \frac{h}{2\pi}$ is the reduced Planck constant.

Only particles with non-zero spin possess a magnetic moment[26] and therefore exhibit Larmor precession, making them eligible for use in NMR.

We will use water as the material sample to test functionality of our EFNMR spectrometer. $H_2O$ contains two hydrogen nuclei (protons) and one oxygen nucleus. $^{16}O$ has an equal number of protons and neutrons, so it has net zero spin, while the hydrogen nucleus (a single proton) has a spin of $\frac{1}{2}$. This is called proton NMR.

In the absence of an external magnetic field, the magnetic moments of the protons are randomly oriented. However, when the water sample is subjected to an external magnetic

field $B_0$, the Zeeman effect causes the energy levels of the protons will split into two eigenstates[27]. The protons in the lower energy level eigenstate align parallel to the direction of $B_0$, while the protons in the higher energy level eigenstate align antiparallel to the direction of $B_0$[13].

The two energy levels are given by

$$E_{\pm} = \pm\frac{1}{2}\hbar\gamma B_0 \tag{2.3}$$

where $\gamma$ is the gyromagnetic ratio of the proton, given by $\gamma/(2\pi) = 42.577\,\mathrm{MHz\,T^{-1}}$[28] (equivalently, $\gamma = 2.675 \times 10^8\,\mathrm{rad\,s^{-1}\,T^{-1}}$), $\hbar$ is the reduced Planck constant[29], and $B_0$ is the magnetic field strength. The energy gap between the two states is therefore $\Delta E = E_+ - E_- = \hbar\gamma B_0$. The proportions of protons in each energy level, at thermal equilibrium, is given by

$$\frac{N_+}{N_-} = e^{-\frac{\Delta E}{k_B T}} \tag{2.4}$$

where $k_B$ is the Boltzmann constant and $T$ is the temperature of the sample in Kelvin[13,30].

At room temperature there are slightly more protons in the lower energy level than the higher energy level[13].

In EFNMR, pre-polarisation is used to enhance the net magnetisation of the spins in the sample. This is achieved by applying a strong magnetic field perpendicular to the Earth's magnetic field for a short period of time. This process increases the population difference between the spin energy levels, thereby amplifying the NMR signal[10]. A stronger pre-polarising magnetic field leads to greater sample magnetisation, thereby improving both the uniformity of alignment and the signal strength.

Once the pre-polarisation field is applied, the water sample will have a net magnetisation parallel to the pre-polarising field. The net magnetisation can be expressed as

$$M = N_+\mu - N_-\mu \tag{2.5}$$

where $\mu = \frac{1}{2}\hbar\gamma$ is the magnitude of the $z$-component of the proton magnetic moment[31].

The external magnetic field will apply a torque[31] onto the magnetic moment of the proton

$$\vec{\tau} = \vec{\mu} \times \vec{B_0} \tag{2.6}$$

This torque causes the net magnetisation to precess with a characteristic resonant angular frequency:

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

called the Larmor frequency[31]; the corresponding linear frequency is $f_L = \omega_L/(2\pi) = (\gamma/2\pi)B_0$. In NMR, the Larmor frequency is unique to each nuclear spin and is influenced by its chemical environment, with factors such as electron shielding affecting the resonance frequency.

Once the pre-polarising field is switched off rapidly (non-adiabatically), the net magnetisation vector - which was aligned with the pre-polarising field - now sits perpendicular to the Earth's field and begins to precess at the Larmor frequency about $\vec{B}_{\text{Earth}}$, with its amplitude decaying back to the equilibrium state[12,31]. The gradual loss of magnetisation energy allows for the measurement of relaxation times, which describe the time scales over which the net magnetisation decays. These relaxation times provide critical information about the dynamics of the spins and their surrounding chemical environment[13]. The three types of relaxation times in NMR are:

1. **T1 (Spin-Lattice Relaxation Time):**
   The relaxation of the nuclear spin magnetisation vector parallel to the external magnetic field. It reflects the time required for the proton spin energy levels to return to thermal equilibrium with the lattice. During this process, the excess energy absorbed by the spins during magnetisation is dissipated into the lattice, which causes the spins to relax back to thermal equilibrium. In essence, T1 is a measure of how quickly the spins release energy to their surroundings. The amplitude of the recovered longitudinal magnetisation is described by the equation

   $$M_z(t) = M_{eq}(1 - e^{-t/T_1}) \tag{2.8}$$

   where $M_z(t)$ is the net magnetisation at time $t$, $M_{eq}$ is the equilibrium magnetisation, and $T_1$ is the spin-lattice relaxation time[13].

2. **T2 (Spin-Spin Relaxation Time):**
   Corresponds to the loss of coherence in the transverse magnetisation of the nuclear spins. Random fluctuations in the local magnetic field cause variations in the precession frequencies of proton spins, leading to the dephasing of the spins[32]. The amplitude of the transverse magnetisation is described by the equation

   $$M_{xy}(t) = M_{xy}(0)e^{-t/T_2} \tag{2.9}$$

where $M_{xy}(t)$ is the transverse magnetisation at time $t$, $M_{xy}(0)$ is the initial transverse magnetisation, and $T_2$ is the spin-spin relaxation time. This shows the T2 time is the time it takes for the transverse magnetisation vector to decay to $1/e$ of its original magnitude[33].

$T_2$ relaxation time is dependent on the molecular structure of the sample that is being measured, thereby allowing us to identify unique chemical environments through its measurement. For example, in pure water $T_2$ is approximately 3 to 4 seconds whereas in water that contains contaminants such as ions it is on the order of 100 to 200 milliseconds, this is called T2 shortening[34].

3. **T2* (Effective Spin-Spin Relaxation Time):**
   T2* arises from inhomogeneities in the local magnetic field, which cause additional dephasing among the spins.

$$\frac{1}{T_2^*} = \frac{1}{T_2} + \frac{1}{T_2^{inhom}} \tag{2.10}$$

where $T_2$ is the true (intrinsic) transverse relaxation time and $T_2^{inhom}$ captures dephasing from field inhomogeneity[13,35]. The spin echo separates these effects.

These relaxation times are fundamental in both medical and chemical applications. In medicine, T1 and T2 shortening is used to generate contrast in magnetic resonance imaging (MRI), allowing for detailed imaging of biological tissues such as the brain[34]. In chemistry, these relaxation times enable the analysis of material structures[5].

One significant advantage of Earth's field NMR devices is that the $T_2^*$ contribution is naturally lower due to the uniformity of Earth's magnetic field[10]. Furthermore, the non-adiabatic (sudden) switching off of the pre-polariser creates an equivalent of the initial 90° pulse in the CPMG sequence: because the pre-polarising field is applied perpendicular to $\vec{B}_{\text{Earth}}$, switching it off faster than the Larmor period leaves the magnetisation pinned in the pre-polarising direction, which is now transverse to $\vec{B}_{\text{Earth}}$. Only the subsequent 180° pulses are therefore required to achieve spin echoes in our experiment[12]. The spin echo and CPMG sequence are explained in the next section.

## 2.2 Spin echo

Earth's field NMR employs several techniques designed to counteract the challenges posed by operating in a very weak magnetic field. One we have already mentioned is the application of a 'pre-polarising' magnetic field perpendicular to the Earth's magnetic field, which significantly strengthens nuclear spin magnetisation, leading to a stronger NMR signal[12].

Another is the use of pulsing sequences that generate a spin echo. A spin echo is a phase refocusing phenomenon that recovers transverse magnetisation lost due to static magnetic field inhomogeneities[36]. First described by Erwin Hahn in 1950, it remains foundational to modern MRI and NMR spectroscopy[37]. The two pulsing sequences relevant to this report are the Hahn (spin) echo[37] and its multi-echo extension, the Carr-Purcell-Meiboom-Gill (CPMG) sequence[38,39].

### 2.2.1 Hahn echo

After a single 90° excitation pulse, the transverse magnetisation gives rise to a free induction decay (FID) signal whose envelope decays on the timescale $T_2^*$ due to both intrinsic $T_2$ processes and static field inhomogeneities. Hahn (1950) showed that a subsequent 180° pulse refocuses the inhomogeneous broadening, producing a spin echo whose decay reflects $T_2$ alone. The Hahn echo pulse sequence is therefore:

$$90^\circ \rightarrow \tau \rightarrow 180^\circ \rightarrow \tau \rightarrow \text{echo} \tag{2.11}$$

The 90° pulse at $t = 0$ rotates $M$ from $\hat{z}$ into the $xy$-plane. During the time $\tau$, free precession causes the spins to dephase due to static $B_0$ field inhomogeneities[37,40]. A spin at resonance offset $\Delta\omega$ acquires a phase:

$$\phi = \Delta\omega \cdot \tau \tag{2.12}$$

The 180° pulse at $t = \tau$ inverts all transverse spin phases ($\phi \rightarrow -\phi$), effectively reversing the sense of precession. This leads to refocusing over the next $\tau$ time period of free precession. Each spin accumulates phase $+\Delta\omega \cdot \tau$, exactly cancelling the earlier $-\Delta\omega \cdot \tau$. At $t = 2\tau$, all static phase offsets cancel and the spins rephase, producing the spin echo[37,40].

The spin echo amplitude at $t = 2\tau$ is:

$$S(2\tau) = S_0\, e^{-2\tau/T_2} \tag{2.13}$$

The inhomogeneous broadening is fully refocused, so the decay depends only on the intrinsic $T_2$ relaxation time. The spin dynamics are governed by the Bloch equations[35]. In the frame rotating at $\omega_0$, the transverse components evolve as:

$$\frac{dM_x}{dt} = \Delta\omega\, M_y - \frac{M_x}{T_2} \tag{2.14}$$

$$\frac{dM_y}{dt} = -\Delta\omega\, M_x - \frac{M_y}{T_2} \tag{2.15}$$

where $\Delta\omega = \omega - \omega_0$ is the resonance offset[35]. In complex notation with $M_+ = M_x + iM_y$:

$$\frac{dM_+}{dt} = -\left(i\Delta\omega + \frac{1}{T_2}\right) M_+ \tag{2.16}$$

which has the solution

$$M_+(t) = M_+(0)\, e^{-i\Delta\omega\, t}\, e^{-t/T_2} \tag{2.17}$$

The 180° pulse maps $M_+ \rightarrow M_+^*$ (complex conjugate), reversing the phase factor $e^{-i\Delta\omega t}$ while the $T_2$ decay continues monotonically[35,37,40]. The echo forms when the reversed phase winds back to zero. However, $T_2$ decay is irreversible and cannot be refocused, hence the $e^{-2\tau/T_2}$ envelope. $T_2$ relaxation arises from stochastic, time-varying local fields (molecular tumbling, diffusion), whose random phase contributions do not cancel under the 180° pulse. Only static (or slowly varying) inhomogeneities are refocused[37,40]. This is the fundamental physical distinction the spin echo exploits.

In principle, $T_2$ can be extracted from a Hahn echo by repeating the experiment at many different values of $\tau$ and fitting the peak amplitudes to equation 2.13. In practice this is slow and wasteful; the CPMG sequence described next samples the same $T_2$ envelope in a single shot.

### 2.2.2 Carr-Purcell-Meiboom-Gill sequence

The Carr-Purcell-Meiboom-Gill (CPMG) sequence is an extension of the Hahn echo designed to measure $T_2$ more accurately and efficiently. It was developed in two stages: Carr & Purcell (1954)[38] identified the multi-echo approach, and Meiboom & Gill (1958)[39] introduced a critical phase correction that dramatically reduced cumulative errors. The Carr-Purcell idea was to apply repeated 180° pulses after a single 90° pulse, generating a train of echoes in one shot:

$$90^\circ_x - \tau - [180^\circ_x - \tau - \text{echo} - \tau]_n \tag{2.18}$$

However, Carr and Purcell noted that any small rotation angle error in the 180° pulse accumulates with each repetition. If the pulse is $180° + \epsilon$ instead of exactly 180°, the error $\epsilon$ grows linearly across the echo train[38], severely distorting the measured $T_2$. Meiboom and

Gill solved the cumulative error problem by shifting the phase of the 180° pulses by 90° relative to the initial 90° pulse[39]:

$$90^\circ_x - \tau - [180^\circ_y - \tau - \text{echo} - \tau]_n \tag{2.19}$$

The initial pulse is applied along the $\hat{x}$ axis, tipping the net magnetisation $M$ onto the $\hat{y}$ axis. The refocusing pulses are then along the $\hat{y}$ axis. This causes pulse errors to self-compensate on even-numbered echoes so the error introduced on one echo is cancelled by the next, keeping the magnetisation on the transverse plane and preventing error accumulation[38,39].

The amplitude of the $n$-th echo, occurring at time $t = 2n\tau$ (where $\tau$ is the time between the initial 90° pulse and the first 180° pulse), is:

$$S(2n\tau) = S_0\, e^{-2n\tau/T_2} = S_0\, e^{-t/T_2} \tag{2.20}$$

This is identical in form to the single Hahn echo result, but now we sample the decay envelope at many time points $t = 2\tau, 4\tau, 6\tau, \ldots, 2n\tau$ in a single experiment. Fitting the peak amplitudes of the echo train to this exponential yields $T_2$ directly[38,39] - this is the method we use in this work.

In practice, molecules diffuse through residual field gradients $G$ during each $\tau$ interval[38,41]. This causes an additional irreversible dephasing that the 180° pulse cannot refocus. The corrected signal amplitude for the $n$-th echo is:

$$S(2n\tau) = S_0 \exp\left(-\frac{2n\tau}{T_2}\right) \exp\left(-\frac{2}{3}\gamma^2 G^2 D \tau^3 \cdot n\right) \tag{2.21}$$

where $D$ is the diffusion coefficient and $G$ is the gradient strength[38,41]. Rewriting in terms of the total time $t = 2n\tau$:

$$S(t) = S_0 \exp\left(-\frac{t}{T_2}\right) \exp\left(-\frac{1}{3}\gamma^2 G^2 D \tau^2 \cdot t\right) \tag{2.22}$$

The diffusion term scales as $\tau^2$, so using short $\tau$ spacing minimises diffusion weighting and recovers a cleaner $T_2$ measurement. This is the practical reason CPMG uses many closely spaced pulses rather than a few widely spaced ones[38,39,41].

The two exponential factors combine into an apparent decay rate:

$$\frac{1}{T_2^{eff}} = \frac{1}{T_2} + \frac{1}{3}\gamma^2 G^2 D \tau^2 \tag{2.23}$$

By running CPMG at several different $\tau$ spacings and extrapolating to $\tau \to 0$, one can isolate the true $T_2$ from diffusion contributions[42].

The electromagnetic signal measured immediately after a spin echo is induced is the 'NMR signal', containing the information needed for molecular structure analysis. The ultimate objective of the NMR spectrometer is to measure the spin echo signal with maximal signal-to-noise ratio (SNR).

## 2.3 Fourier transform

In our spectrometer design, the Raspberry Pi Pico 2 microcontroller ADC measures the spin echo signal as a function of time. However, to extract physically useful information from the signal, it must be analysed in the frequency domain[43]. For this, we must perform a Fourier transform on the time-series data:

$$\tilde{f}(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(t) e^{-i\omega t}\, dt \tag{2.24}$$

where $f(t)$ is the time-domain data and $\tilde{f}(\omega)$ is the frequency-domain data, $\omega$ is the angular frequency, and $t$ is the time[44].

The inverse Fourier transform[44] can be used to convert frequency-domain data back into the time domain:

$$f(t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \tilde{f}(\omega) e^{+i\omega t}\, d\omega \tag{2.25}$$

We use the Fast Fourier transform (FFT) algorithm[43,45] which performs a discrete Fourier transform (DFT) on time-series data into the frequency domain (and the inverse). The FFT is used as it can be programmed very easily on the microcontroller by using pre-existing modules such as `scipy.fft`[46] for Python or `PocketFFT`[47] for C++. Furthermore, an FFT function is built in to the oscilloscope that we used in testing and development[48]. The FFT algorithm has been chosen because of its $O(n \log n)$ time complexity, which allows us to compute large amounts of data in significantly shorter time than the naive discrete Fourier transform (DFT), which has quadratic $O(n^2)$ time complexity[43,45].

The DFT itself maps a sequence of $N$ complex numbers in the time domain to a corresponding sequence in the frequency domain, decomposing a signal into its constituent sinusoidal components. The Cooley-Tukey algorithm[45] is the most widely employed FFT formulation and operates on the principle of recursive divide-and-conquer: it partitions the DFT of a sequence into smaller DFTs of subsequences, exploiting the periodicity and symmetry properties of the complex roots of unity to eliminate redundant computation[45].

## 2.4 Signal noise

This section provides brief descriptions of the different sources of electronic noise that affect signal processing in Earth's field NMR spectrometry.

An Earth's field NMR (EFNMR) spectrometer operates at an extremely low Larmor frequency - typically in the range of 1-4 kHz, corresponding to the weak geomagnetic field of approximately 50 µT - and as such is subject to a distinct and challenging noise environment that differs considerably from that encountered in conventional high-field NMR instrumentation.

### 2.4.1 Environmental Electromagnetic Interference

Environmental noise arises from external electromagnetic fields inducing currents in the coils. Key sources of this noise include fluctuations in the Earth's magnetic field caused by nearby metal objects and electromagnetic fields from the mains electricity grid and electronic equipment. Unlike systematic noise, this noise can fluctuate unpredictably depending on the local environment and setup conditions[12].

Perhaps the most practically challenging noise source in EFNMR is external electromagnetic interference. Since the Larmor frequency coincides with the audio and very-low-frequency (VLF) band, the spectrometer is highly susceptible to anthropogenic interference sources, including mains power supply harmonics (50 Hz and their overtones), electrical machinery, and radio transmissions[49]. In urban or laboratory environments, such interference can overwhelm the intrinsically weak NMR signal entirely. Mitigation strategies typically include operating in electrically quiet rural locations, employing shielded enclosures, and utilising gradiometric coil configurations that discriminate against spatially uniform interference fields[49].

The spectrometer's receiver coils act like a solenoid so a small but potentially influential electromotive force (EMF) is induced in them whenever they experience a change in magnetic flux. This induced EMF may interfere with the electromagnetic pulses if it does not dissipate quickly. The induction of EMF in the coils is described by Faraday's Law of inductance:

$$\oint_C \mathbf{E} \cdot d\mathbf{l} = -\frac{d}{dt} \iint_S \mathbf{B} \cdot d\mathbf{A} \tag{2.26}$$

where $C$ is a closed curve in space, $\mathbf{E}$ is the electric field, $d\mathbf{l}$ is a vector representing an infinitesimal segment of the curve, $S$ is a surface bounded by $C$, $\mathbf{B}$ is the magnetic field, and $d\mathbf{A}$ is a vector representing an infinitesimal element of the surface[50].

Specifically, the induced electromotive force (EMF) in a coil can be expressed as

$$\mathcal{E} = -N\frac{d\Phi_B}{dt} \tag{2.27}$$

where $\mathcal{E}$ is the induced EMF, $N$ is the number of turns in the coil, and $\Phi_B$ is the magnetic flux through one turn[50].

For our receiver coils wrapped around a cylindrical reservoir of water, the magnetic flux through one turn is given by the solenoid equation:

$$\Phi_B = \mu_0 \frac{N}{L} A I \tag{2.28}$$

where $\mu_0$ is the magnetic permeability of free space (water is diamagnetic and has a relative permeability of $\approx 1$), $A$ is the cylinder's cross-sectional area, and $I$ is the current in the solenoid[50].

### 2.4.2 Johnson-Nyquist noise

Johnson-Nyquist noise is the induced voltage or current that is generated as a result of the microscopic thermal vibrations of charge carriers (electrons) in electrically conductive materials at equilibrium[51,52]. First characterised experimentally by John B. Johnson[51] and theoretically explained by Harry Nyquist[52] in 1928, this noise is an intrinsic property of any resistive element at a temperature above absolute zero. Unlike other noise sources, Johnson-Nyquist noise is entirely independent of any applied voltage or current; it exists solely as a consequence of thermodynamic activity within the material[51,52]. Considering our design objective of maximising signal-to-noise ratio, Johnson-Nyquist noise poses an obstacle in that it cannot be removed at room temperatures (the only way to reduce it is to cool the receiver electronics, as is done in high-end cryoprobe systems). Therefore, the only reliable and affordable course of action in our setup is to maximise the signal strength to achieve a favourable SNR.

The power spectral density of Johnson-Nyquist noise is given by the Nyquist formula[52]. For a resistor of resistance $R$ at absolute temperature $T$, the open-circuit noise voltage spectral density is expressed as:

$$S_v(f) = 4 k_B T R \tag{2.29}$$

where $k_B$ is the Boltzmann constant ($1.38 \times 10^{-23}$ J/K) and $f$ is frequency. From this, the root-mean-square (RMS) noise voltage across a bandwidth $\Delta f$ can be derived as:

$$V_{rms} = \sqrt{4 k_B T R \, \Delta f} \tag{2.30}$$

This relationship reveals several important engineering implications. Firstly, noise power

increases linearly with both temperature and resistance, suggesting that minimising these parameters is critical in low-noise circuit design. Secondly, the noise is spectrally flat across the frequencies of practical interest - a characteristic that classifies it as white noise. It is worth noting, however, that this flat approximation holds only up to frequencies on the order of $k_B T/h$ (where $h$ is Planck's constant), beyond which quantum corrections become non-negligible - a regime far outside the scope of conventional electronic engineering applications[52].

In practical terms, Johnson-Nyquist noise sets a fundamental lower bound on the noise floor of any electronic system. This has profound consequences in the design of electronics where the signal-to-noise ratio (SNR) is a critical performance metric, such as our Earth's field NMR spectrometer.

### 2.4.3 Shot noise

Shot noise, sometimes referred to as Poisson noise, is another fundamental noise mechanism encountered in electronic systems, arising from the discrete and statistically independent nature of charge carriers[53]. First described by Walter Schottky in 1918, shot noise manifests whenever a current flows across a potential barrier, such as those in semiconductor junctions found in diodes and bipolar junction transistors (BJTs). Unlike Johnson-Nyquist noise, which is present even in the absence of current flow, shot noise is intrinsically linked to the passage of current; it is a direct consequence of the quantisation of electric charge[53], where current is not a smooth, continuous flow but rather a succession of discrete electron arrivals, each carrying charge $q = 1.6 \times 10^{-19}$ C.

The statistical nature of this process is well modelled by the Poisson distribution, as the arrival of individual charge carriers at a barrier can be treated as independent random events occurring at a known average rate[53]. From this statistical framework, the power spectral density of shot noise current is given by the Schottky formula:

$$S_i(f) = 2qI_{DC} \tag{2.31}$$

where $q$ is the elementary charge and $I_{DC}$ is the mean (DC) current flowing through the device[53]. The corresponding RMS shot noise current over a bandwidth $\Delta f$ is therefore:

$$I_{rms} = \sqrt{2qI_{DC}\,\Delta f} \tag{2.32}$$

This result implies shot noise scales with the square root of the average current, meaning that higher bias currents produce greater absolute noise, though the relative noise (characterised by the ratio $I_{rms}/I_{DC}$) diminishes with increasing current[53]. Furthermore,

like Johnson-Nyquist noise, shot noise exhibits a white noise spectrum, remaining spectrally flat up to frequencies comparable to the inverse of the carrier transit time through the device, beyond which the noise power rolls off.

Due to the Poisson distribution, the signal-to-noise ratio of shot-noise-limited counting (i.e., when $N$ discrete charge carriers are detected) is given by:

$$\text{SNR} = \frac{N}{\sqrt{N}} = \sqrt{N} \tag{2.33}$$

It is important to note that shot noise is not universally present in all current-carrying conductors. In a metallic resistor, for instance, the Coulomb interactions between electrons and the constraining influence of the lattice suppress the independence of carrier arrivals, meaning that current flow in such materials does not exhibit shot noise in the classical sense[53] (the dominant noise mechanism remains Johnson-Nyquist). Shot noise is most significant in devices where carriers traverse a potential barrier or depletion region with a high degree of statistical independence, rather than traversing through a continuous medium[53]. Examples of this are p-n junctions, BJTs, and photodiodes. In our spectrometer design, the operational amplifiers (op-amps) in the detection circuit may contribute significant levels of shot noise, especially if they have a BJT input stage. This requires the careful selection of alternative op-amp technologies such as junction field-effect transistors (JFET) input op-amps, though such decisions are based on multiple factors that will be considered by the group working on the detection circuit.

### 2.4.4 1/f Noise

One of the most significant low-frequency noise sources is flicker noise, commonly referred to as 1/f noise due to its characteristic power spectral density, which scales inversely with frequency:

$$S_v(f) \propto \frac{K}{f^\alpha} \tag{2.34}$$

where $K$ is a device-dependent constant and $\alpha$ is an exponent typically close to unity[54]. Given that EFNMR operates in the low kilohertz regime, the spectrometer's operating frequency falls squarely within the range where 1/f noise dominates in many electronic components[54], particularly in the detection stages (preamplifier and signal conditioning). Therefore, it is critical to carefully select low-noise components such as JFETs, which exhibit considerably lower flicker noise than MOSFET-based alternatives.

### 2.4.5 Geomagnetic and Environmental Field Fluctuations

Unlike high-field instruments where the static field is generated by a superconducting magnet, Earth's field NMR relies upon the Earth's geomagnetic field as its polarising medium. This field is subject to temporal fluctuations arising from geomagnetic micropulsations, ionospheric activity, and local magnetic disturbances[55]. All of these contribute directly to instability in the Larmor frequency and effective dephasing of the NMR signal. These fluctuations manifest as an additional broadening of the resonance linewidth, degrading both spectral resolution and SNR[55].

### 2.4.6 Mechanical (Vibrational) Noise

In sensitive inductive detection schemes, any mechanical vibration of the receiver coil within the ambient magnetic field will induce a spurious electromotive force (EMF) via Faraday's law, constituting a form of vibrational noise[56]. This is particularly relevant in portable or field-deployed EFNMR instruments, where isolation from ground vibration and acoustic excitation cannot always be guaranteed. Rigid mechanical construction and the use of vibration-damping mounts can mitigate this.

In summary, the EFNMR spectrometer presents a uniquely demanding noise environment in which environmental and low-frequency noise sources frequently dominate over the fundamental thermal limits. Achieving adequate SNR in such instruments demands a holistic approach encompassing careful electronic design, physical shielding, intelligent coil geometry, and - where possible - judicious choice of operating environment.

# Chapter 3

# Methodology

## 3.1 Apparatus design

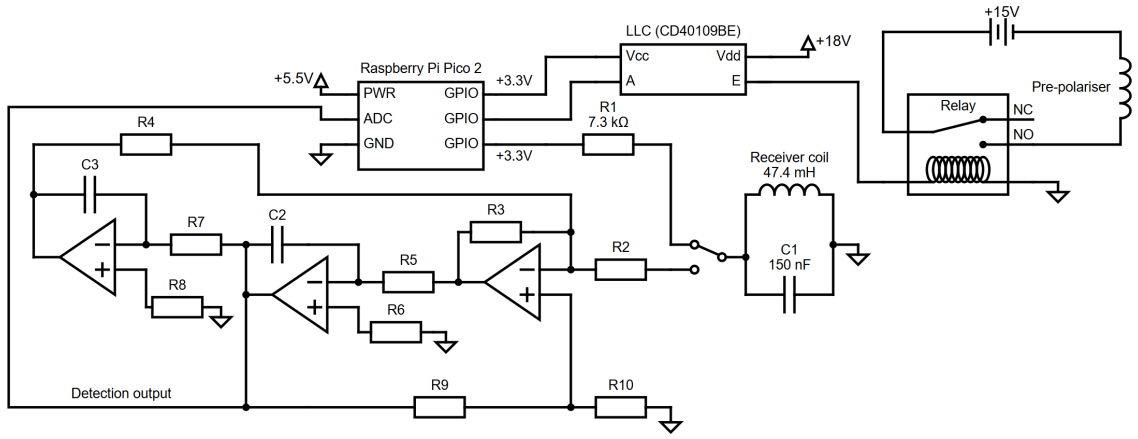



Figure 3.1: Schematic of the NMR apparatus design.

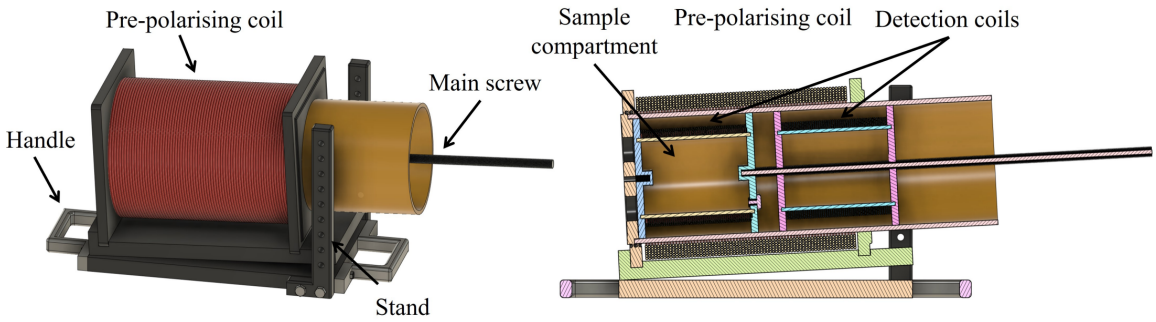



Figure 3.2: 3D rendering of the pre-polariser and reciever coils design (made by Michal Wlodarczyk[18]).

Figure 3.1 illustrates the key components of the NMR apparatus, including the Raspberry Pi Pico 2 microcontroller, which is the central connection and processing point of the pre-polarising, pulsing, and detection circuits.

The Raspberry Pi Pico 2 is connected by a General-Purpose Input/Output pin (GPIO) to a Texas Instruments CD401109BE Logic Level Converter/Shifter (LLC)[57] that uses an external battery to ramp up the Pico 2's output voltage from 3.3 V to 18 V, before connecting via a single pole double throw relay switch to the pre-polarising coils. The voltage is ramped up to to ensure the relay switches reliably as it was unreliable initially. A GPIO pin with an output voltage of 3.3 V is used to connect in series to a 7310 Ω resistor to a single pole double throw relay switch that allows switching between the pulsing circuit and detection circuit.

The pulsing circuit is completed with a 150 nF capacitor in parallel to the receiver coils. This forms an RLC resonant circuit because the receiver coils themselves act as inductors with a measured inductance of 47.4 mH. The RLC circuit design was chosen because it acts a harmonic oscillator, which smooths the square wave pulses emitted by the Pico 2, allowing them act more like sinusoidal pulses. The use of sinusoidal pulses in creating efficient power transfer will be discussed later in this section. An LC circuit can also be used for resonance and smoothing but the resistor in the RLC circuit provides the additional benefit of damping - which causes the signal oscillations to decay more quickly than an LC circuit - which helps reduce noise created by the pulses. Furthermore, the RLC circuit reduces the peak resonant frequency, and a resistance is chosen to fix the peak resonant frequency as close as possible to the Larmor frequency. The value of the Larmor frequency for distilled water under the Earth's magnetic field in Manchester (estimated as 50 µT) was calculated to be 2103 Hz.

## 3.2 Polarising field strength and homogeneity

We formed a hypothesis that by placing the water sample in a region of stronger and more uniform magnetic field, the difference between the higher and lower spin energy levels would increase, leading to a higher spin echo signal amplitude once the pre-polarising field is switched off.

To investigate this hypothesis we measured the magnetic field strength and homogeneity of the pre-polariser.

### 3.2.1 Method for measuring field homogeneity

When measuring the magnetic field strength in the pre-polariser, we performed the following procedure:

1. Zeroed the Gauss meter using a Zero Gauss Chamber to ensure accurate measurements.

2. Created a plastic mount to hold the Gauss meter's probe in the centre of the pre-polariser exactly parallel to the cylinder's axis, ensuring consistent placement for every measurement.

3. Using a flexible plastic ruler, we moved the Gauss meter probe along the length of the pre-polariser at approximately 1 cm intervals, recording the magnetic field strength at each point.

4. Started measurements by placing the Gauss meter at the bottom of the pre-polariser cylinder (0 cm position) and finished at the top of the cylinder reaching a length of 21 cm.

5. We repeated this process three times for reliable results and calculating measurement standard deviations.

To minimise systematic errors, we removed as many metal objects as possible from the vicinity of the pre-polariser, including electronic circuits, components, tools, and other ferromagnetic materials. However, it was not feasible to remove the metal desk on which the pre-polariser was placed, which may have introduced an error in the measurements.

## 3.3 Noise in the receiver coils

This section outlines an investigation adjacent to the tests described in section 3.2. In consideration of our research objective of maximising the signal-noise ratio (SNR), we decided to investigate the presence of electronic noise in the receiver coils (used for pulsing and detection).

Noise plays a particularly critical role in the performance of an EFNMR system because the signal being measured is inherently weak due to the use of the Earth's magnetic field, which is only about 50 $\mu T$. As a result, the signal is much smaller than that observed in high-field NMR systems. If the noise in the system is too high, it can easily overpower the weak NMR signal, making it extremely difficult to detect CPMG spin echo or Hahn echo signals. Therefore, identifying, understanding, and mitigating noise sources is essential for achieving a usable signal-to-noise ratio in an EFNMR system. The presence of noise in the receiver coils during the pulsing phase may create interference with the Larmor frequency pulses which may be an obstacle to successfully achieving spin echo with the CPMG sequence. Additionally, noise in the detection phase will reduce the signal-noise ratio, making it harder for us to accurately measure T2 relaxation times.

Therefore, we aimed to quantify noise in the receiver coils by testing the following receiver coil configurations:

- Two counter-wound coils (parallel and antiparallel).

  - Water in only one coil.
  - Water in both coils.
- Two co-wound coils (parallel) with water in both coils.
- A single coil in the centre of the pre-polariser.

These configurations were tested because they were the only arrangements achievable with the available receiver coils and pre-polariser design. Due to strict time constraints, no structural changes to the receiver coils nor the pre-polariser were made.

The purpose of testing two counter-wound coils was to investigate the hypothesis that counter-winding may cancel out noise in the coils by inducing EMFs in anti-phase with one another. We tested this with water in either one or both coils to assess the potential impact of the sample's presence on noise reduction. With water being diagmagnetic and having a relative magnetic permeability $\approx 1$, we hypothesised this would not have a significant effect. The purpose of testing two co-wound coils containing water was to assess the impact of maximising the sample volume. Each coil was wrapped around a cylindrical reservoir capable of holding 290 mL of water. Adding water to both coils provided a combined water volume of 580 mL, which was expected to generate a stronger spin echo signal, as more nuclear spins would be polarised and precessing. The co-wound configuration was also used as a control to assess if the counter-wound configuration shows a measurable reduction in noise. The purpose of testing a single coil in the middle of the pre-polariser was to position the water sample in the region of the highest magnetic field strength and homogeneity. The single coil configuration was filled with 290 mL of water.

### 3.3.1 Method for measuring noise levels

We measured the voltage amplitude of noise in the coils with the following procedure:

1. Isolated the receiver coils from the rest of the circuit.
2. Switched off the pre-polariser to ensure no active magnetic field was present (to isolate the base noise levels due to circuit components & environmental magnetic fields).
3. Connected a Tektronix TBS1052C oscilloscope[48] in parallel to the receiver coils.
4. Set up the receiver coils within the pre-polariser (while keeping the pre-polariser turned off).
5. For each coil configuration, we recorded the amplitude of voltage fluctuations for 2000 ms with a sample rate of 3125 Hz.
6. Repeated measurements 3 times for each configuration.

7. Performed a Fourier transform on the noise data to analyse it in the frequency domain to attempt to identify specific sources of noise (Eg. mains noise is 50 Hz).

## 3.4 Power Transfer Efficiency

One of the most critical parts of the pulsing sequence is to ensure the CPMG pulses are able to create an adequate transfer in power to the nuclear magnetic moments to create the 180° rotations of the magnetic moment that rephase the precessing spins for spin echo. The previous year's group hypothesised that square wave pulses would not be able to transfer power adequately, thus they designed a complex pulsing circuit (see appendix B.3) which aimed to transform the microcontroller's digitally created square waves into sinusoidal waves. A continuous sinusoidal waveform creates a highly efficient power transfer at a given frequency because its Fourier Transform produces a pair of narrow delta functions, and finite-duration sinusoidal pulses create a sinc ($sincx = \frac{sinx}{x}$) shaped peak in the frequency domain. This narrow peak allows for power to be transferred more efficiently to the spins at the targeted Larmor frequency.

In contrast, the Fourier Transform of a square wave contains multiple harmonics, meaning the power transfer is spread over a broader range of frequencies. This causes less power to be transferred at the Larmor frequency. Inefficient power transfer creates two main problems; the first that the pulses may not be strong enough to create 180° rotations of the nuclear magnetic moments. The second issue is that the more power transfer occurs over a broader range of frequencies around the Larmor frequency, the more phase differences are introduced to the precessing spins. This would create noise in the spin echo signal. In contrast to the previous years group, we believe that the square wave pulses from the microcontroller can be sufficiently smoothed to imitate sinusoidal waves through the use of a simple RLC harmonic oscillator desgin rather than the highly complex pulse generation circuit that they designed, which included the use of op-amps and low-pass filtering on top of an RLC harmonic oscillator B.3[18,22]. This would massively simplify development and reduce production and maintenance costs. Therefore, in this section we test the hypothesis that square waves pulses in the receiver coils are sufficiently smoothed using the RLC resonant circuit design to mimic sinusoidal waves. In particular, we will measure the power transfer efficiency (PTE) of the RLC smoothed square waves and compare them with the theoretical PTE of finite-duration sine waves.

### 3.4.1 Method for measuring Power Transfer Efficiency

Data was obtained to test the above hypothesis using the following process:

1. Constructed the RLC resonant circuit with a 47.4 mH inductor (the receiver coils) and a 7310 Ω resistor.

2. A variable capacitor was added in parallel to a single receiver coil placed in the middle of the pre-polariser.

3. The capacitance was varied from 0.01 µF to 0.31 µF.

4. Multiple readings from the oscilloscope were taken for each capacitance value.

5. We wrote a code to analyse the waveform data. The waveform, initially in the time domain, was converted to the frequency domain using a Fourier Transform, allowing us to visualise how power was distributed across different frequencies.

6. We identified the peak and fitted it to a Gaussian distribution (which is an adequate approximation of the *sinc* function & allows for easier distribution fitting).

7. The area under this peak was calculated to represent the power transferred by the pulses.

8. We calculated the ratio of the area under the Larmor frequency peak to the theoretical power transferred by a perfect finite-duration sine wave at 2103 Hz.

9. This ratio is the Power Transfer Efficiency and was used to assess how effectively the square wave pulses had been smoothed to approximate a sine wave.

## 3.5 Hardware and firmware optimisation

The final development hypothesis we tested is related to the potential limitations that suboptimal hardware and firmware may impose on the spectrometer function. The CPMG pulsing sequence is the crux of the spectrometer's functionality, without its successful implementation, spin echo cannot be created and the T2 relaxation times cannot be measured. The pulses of the CPMG sequence must be delivered with the highest possible temporal precision in order to transfer power at the targeted Larmor frequency. If the real pulse frequency deviates significantly from the targeted Larmor frequency, spin echo will not be achievable, even if we were to use a highly complex circuit design that creates maximal power transfer efficiency through perfect imitation of sinusoidal pulses. Therefore, it is imperative to test and optimise the microcontroller hardware and firmware to ensure there is minimal deviation from the Larmor frequency (within the budget constraints). We hypothesised the previous year's implementation of a Raspberry Pi Pico (RP2040) microcontroller has suboptimal temporal resolution compared to the newer Raspberry Pi Pico 2 (RP2350). The RP2350 features a more powerful core processing unit and almost double the flash memory and SRAM, but can be purchased for a similar price. Furthermore, the RP2350 is considered a drop-in replacement, so no changes are required to the circuit to make this improvement.

Furthermore, we believe the pulsing can be optimised by flashing it with C++ firmware rather than MicroPython firmware. MicroPython is a microcontroller optimised version of the Python programming language. Python is an interpreted language, in contrast to C++ which is a compiled language. Interpreted languages are generally slower than compiled languages because the code is translated to bytecode line by line, whereas compiled languages are compiled into assembly at runtime and can perform very closely to the native hardware speeds. Additionally, in embedded systems engineering, the majority of performance-critical firmware is written in C++ and C. This is not only due to improved performance, but also allows greater control over memory and data registers. Transitioning to C++ will align this project with industry standards and creates space for future optimisations to the direct memory access of the microcontroller to maximise the data sampling rate for detecting spin echo signals. The maximum data sampling rate of the RP2350 analogue-digital-converter (ADC) is 500 000 Hz, which gives a native time resolution of 2 µs. Using C++ firmware may allow us to sample at a rate much closer to native speeds than MicroPython. We hypothesised that these changes would result in higher temporal precision in pulsing, thereby improving power transfer at the Larmor frequency. This is based on the reasoning that small deviations in the pulse timings, even on the order of a few hundred microseconds, may cause the power to be transferred at frequencies other than the Larmor frequency.

### 3.5.1 Method for testing hardware & firmware speed

We measured the overhead times of the RP2040 and RP2350 microcontrollers and MicroPython and C++ firmwares using the following procedure:

1. Before testing, I first wrote an equivalent version of the pre-existing MicroPython firmware (carried over from the previous year's group[17,18]) in C++.
2. Then I isolated the pulsing logic for both sets of firmware, as this is the performance critical area of interest of this report.
3. This isolated code tracked the start and finish times of the CPMG pulse sequence using a nanosecond resolution timer.
4. We ran this code for all four combinations of the microcontroller & firmware configurations
   (a) RP2040 with MicroPython
   (b) RP2040 with C++
   (c) RP2350 with MicroPython
   (d) RP2350 with C++

5. We repeated these measurements 10 times for each configuration to ensure reliability.

6. We then calculated the mean overhead time and standard deviation for each configuration.

The overhead time of the CPMG sequence was calculated as the difference between the measured time and the programmed sequence duration:

$$T_{overhead} = T_{measured} - T_{CPMG} \tag{3.1}$$

### 3.5.2 Method for measuring pulse frequency deviation

The pulsing sequence is created by the microcontroller, so it is imperative that the microcontroller is able to deliver the pulses with the highest possible temporal precision in order to transfer power at the targeted Larmor frequency. We measured the frequency deviation of the pulses for each combination using the following procedure:

1. Programmed a sequence of 10 pulses for each hardware-firmware configuration.

2. Measured the actual pulse frequencies for each combination using a Tektronix TBS1052C oscilloscope[48].

3. Calculated the mean and standard deviation of the pulse frequencies for each combination.

4. Calculated the differences between the mean frequencies and the intended Larmor frequency.

5. Calculated the Power Transfer Efficiency for each combination using the method outlined in subsection 3.4.1.

The measurements of timing, frequency deviation, and power transfer efficiency provide a comprehensive view of how effectively our optimisations improved the Earth's field NMR spectrometer's pulsing performance.

# Chapter 4

# Results and Discussion

This chapter presents and discusses the experimental findings of the three investigations that comprised the pulsing half of the project: receiver-coil configurations within the pre-polariser, the development of a square-wave pulsing circuit, and a comparison of microcontroller hardware and firmware for pulse-sequence generation. Section 4.1 evaluates the assembled spectrometer against the design requirements set out in Section 1.3, after which Sections 4.2–4.4 present and discuss each investigation in turn. Each of those sections states the hypothesis under test, presents the central findings, evaluates the hypothesis verdict, compares the result to previous iterations of the project where possible, and identifies targeted next steps for the following cohort. The chapter closes (Section 4.5) with a synthesis of the findings in the context of the project's educational-outreach aim.

## 4.1 Project aims and overall outcomes

Table 4.1 summarises the outcome of each design requirement set out in Section 1.3.

Table 4.1: Evaluation of the final spectrometer against the design requirements set out in the project aims.

| **Requirement** | **Target** | **Outcome** |
|---|---|---|
| Total cost | ≤£250 | £178.65 (see appendix A.1) |
| Footprint | $\lesssim 1.0\,\mathrm{m}^2$ | 0.5 to $1.0\,\mathrm{m}^2$ |
| Time resolution | ~µs | ≥2 µs |
| CPMG pulse sequence | Implemented | Implemented |
| Portability and outdoor use | Required | Not portable |

The cost ceiling was met comfortably, with the total parts bill of £178.65 leaving £71.35 of headroom against the £250 target. Furthermore, for the pulsing circuit the CPMG sequence is fully implemented. For the detection circuit, the RP2350 ADC provides a potential time resolution of 2 µs. Two requirements were not satisfied: the spectrometer is not yet sufficiently portable for off-site use, and the functional time-resolution cannot be

evaluated until the detection circuit (the subject of the companion report[24,25]) has been tested. The instrument is therefore best described as a partially-complete educational platform; the pulsing chain is in place, and the cost and pulse-sequence targets are met, but classroom deployment is not yet possible without further integration and packaging work.

## 4.2 Coil configurations

### 4.2.1 Hypothesis

We hypothesised that placing the water sample in a region of stronger and more uniform magnetic field within the pre-polariser would increase the Zeeman splitting of the proton spin energy levels and therefore the spin-echo signal amplitude. Adjacent to this, we hypothesised that a counter-wound double-coil arrangement, in which two coaxial coils are wound in antiparallel and connected in series, would cancel external magnetic interference through gradiometric coupling - lowering the noise floor without sacrificing sample volume. To test both hypotheses simultaneously we measured the polarising field profile along the bore of the pre-polariser and the noise amplitude at the detection node for four candidate receiver-coil configurations: a single coil at the geometric centre of the pre-polariser, a co-wound double coil with water in both coils, a counter-wound double coil with water in both coils, and a counter-wound double coil with water in only one coil (the last included to test the sub-hypothesis that water, being diamagnetic with relative permeability $\approx 1$, would not measurably affect the noise floor). The configuration giving the best combination of field exposure, sample volume, and noise floor was to be selected for the final spectrometer.

### 4.2.2 Magnetic field strength and homogeneity in the pre-polariser

Figure 4.1 shows the variation of the polarising magnetic field along the length of the pre-polariser. The field peaks at approximately 7.42 mT at the centre of the pre-polariser, around the 10 cm mark, and tapers gradually on either side, falling rapidly below 6 cm and above 15 cm. The region of most homogeneous and highest field strength is therefore a roughly 9 cm window centred on the geometrical midpoint of the pre-polariser; it accommodates a single receiver coil comfortably but only marginally accommodates two coils stacked coaxially.

This profile imposes a direct trade-off on the coil-configuration decision. A single receiver coil placed at the centre of the pre-polariser sits entirely within the high-field plateau, maximising the Zeeman splitting of the proton spin energy levels and therefore the thermal-equilibrium magnetisation contributing to the detected signal. A double-coil arrangement extends outside this window: the outer halves of each coil sample regions of weaker and less homogeneous field, where nuclear spins precess at slightly different Larmor

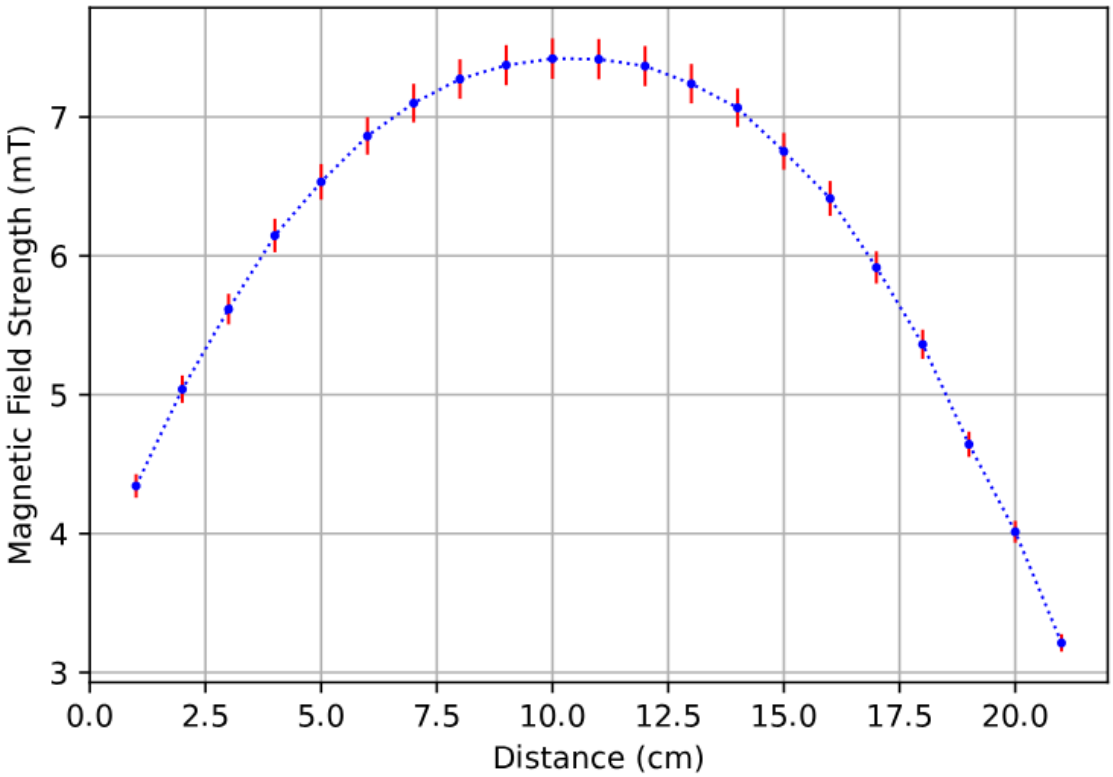



Figure 4.1: Magnetic field strength and homogeneity in the pre-polariser. Error bars correspond to 1.96 standard deviations of the measured field strength at each position (95% confidence interval).

frequencies and dephase relative to those at the centre, broadening the resonance and reducing the coherent signal. Against this, the double-coil arrangement doubles the usable sample volume from approximately 290 mL to 580 mL, and sample volume directly increases signal amplitude in EFNMR. The single-coil and double-coil arrangements therefore exchange field uniformity for sample volume in roughly equal measure, and the choice between them must be made on the basis of which factor dominates the achieved signal-to-noise ratio in practice.

### 4.2.3 Noise amplitudes across the four configurations

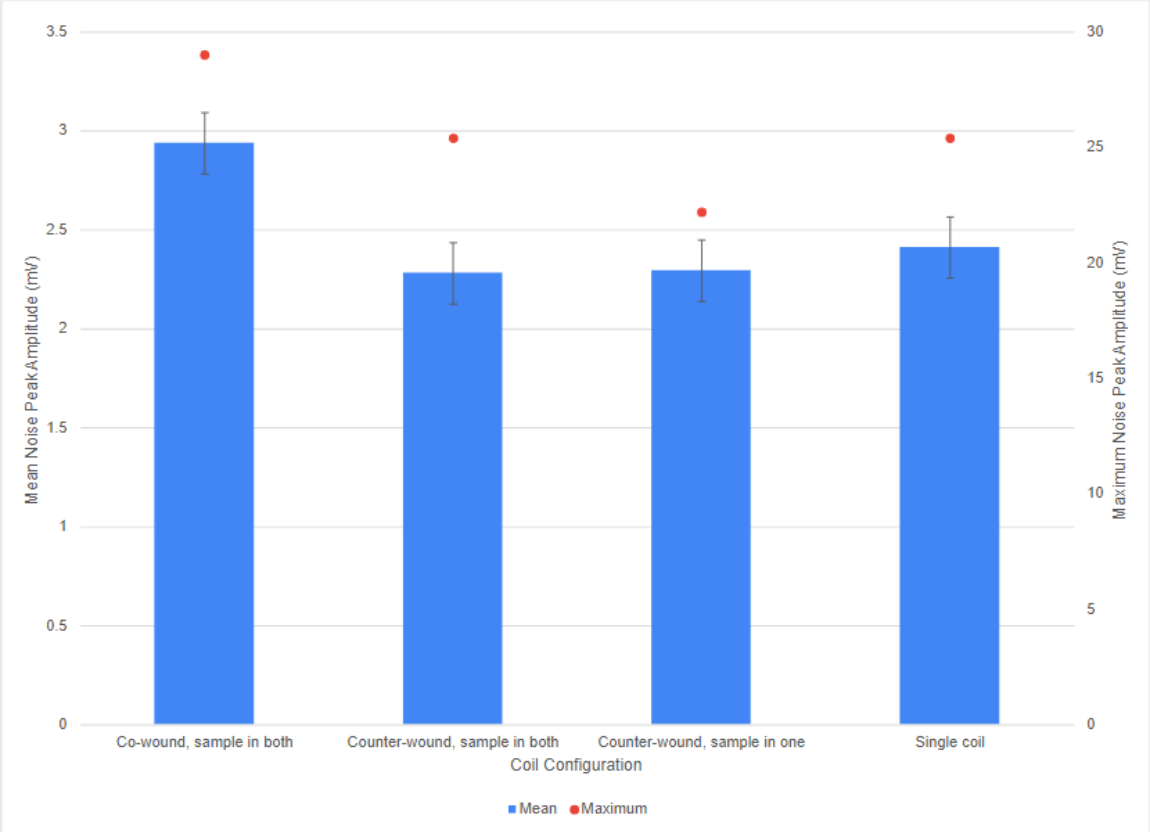



Figure 4.2: Mean and maximum noise peak amplitudes for the four tested coil configurations. Error bars represent the standard error of the mean.

Figure 4.2 displays the mean and maximum noise peak amplitudes for each of the four tested configurations. The co-wound configuration exhibits a mean noise amplitude approximately 0.5 mV higher than the other three, which lie within their respective standard errors of one another. At the precision of this measurement, the single coil and the two counter-wound variants are therefore indistinguishable from one another; the co-wound configuration is clearly the noisiest. The water-in-one-coil and water-in-both-coils counter-wound variants give the same noise floor, supporting the sub-hypothesis that water's diamagnetism does not measurably affect the noise behaviour of this arrangement.

The counter-wound double coil achieving a noise floor comparable to the single coil is consistent with the gradiometric principle underlying its design. The two coils are stacked coaxially with their windings in antiparallel and connected in series, so any spatially uniform external magnetic field induces EMFs of equal magnitude but opposite sign in the two coils, which cancel at the output. The NMR signal, by contrast, is preserved because the two samples sit in opposite poles of the pre-polariser and therefore begin Larmor precession 180° out of phase: the induced EMFs from the two precessing samples are of opposite sign, and when combined with the antiparallel winding direction they sum constructively at the output. The noise cancelled by this arrangement is therefore inferred to be ambient environmental magnetic interference from unknown sources in the laboratory; local, uncorrelated sources intrinsic to each coil - Johnson–Nyquist, shot, and $1/f$ flicker noise - are not coherent between the two coils and contribute equally to all four configurations tested. Fourier transforms of the time-series noise data show no peaks in the 50 Hz region (see appendix B.1), which rules out mains-grid pickup as the dominant external source of noise in our setup.

### 4.2.4 Configuration choice

The configuration decision was weighted on three factors: magnetic-field strength and homogeneity at the sample position, sample volume, and noise amplitude at the detection coil. The single-coil configuration sits entirely within the strongest and most uniform region of the pre-polariser field but is restricted to half the sample volume of the double-coil alternatives. The co-wound double coil benefits from the larger sample volume but pays a penalty in both noise amplitude and field homogeneity. The counter-wound double coil retains the 580 mL sample volume and matches the single-coil configuration's low noise floor through gradiometric cancellation of external interference. Theoretically this arrangement should approximately double the SNR over the single-coil configuration; in practice the gain will be smaller, because the outer halves of the two coils sample regions of weaker and less homogeneous field and therefore contribute reduced and partially incoherent signal.

On the balance of these factors we selected the counter-wound double-coil configuration, with water in both coils, for the final spectrometer; both hypotheses are supported.

### 4.2.5 Limitations

Several methodological choices limit the strength of these conclusions. The configuration space explored was small: only four arrangements were tested, and the coils themselves were not modified - only their relative placement and the volume of water occupying them. A more impactful investigation would have varied turn count, wire gauge, and coil geometry, since these directly affect inductance, resistance, and noise pickup. Our results therefore characterise the best of the available options rather than the best achievable design.

The noise metric was also weaker than it could have been. Reading the oscilloscope's baseline voltage (with a minimum step of 1 mV) characterises the total noise at one node but does not separate the contributions of Johnson-Nyquist noise, electromagnetic interference, and amplifier input noise. Established measures of electronic noise such as noise power spectral density (in $V^2\,Hz^{-1}$) and noise figure would have allowed us to attribute differences between configurations to specific physical mechanisms and to compare quantitatively with the EFNMR literature. A swept-tuned (superheterodyne) spectrum analyser would measure noise directly in the frequency domain, avoiding the propagation of errors from Fourier-transforming time-series data, and would itself contribute substantially less instrument noise than the oscilloscope used here.

The pre-polariser homogeneity measurements were limited by three factors. First, position along the bore was set with a flexible plastic ruler, giving a distance uncertainty of order ±1 mm on top of the nominal 1 cm sampling interval. Second, although a bespoke plastic mount was used to hold the Hall probe parallel to the cylinder's axis, the mount's fit varied along the bore, so some residual tilt of the probe's sensitive axis relative to the local field cannot be excluded. Third, the pre-polariser sat on a metal-topped desk that could not be removed for the measurement; the ferromagnetic material of the desk distorts the field in a manner that is difficult to quantify. The three repeat trials at each position, from which we report means and standard deviations, mitigate random error but do not address any of these systematic effects. A final restriction is that we sampled only the central axis: a useful homogeneity figure needs radial as well as axial coverage, since spins throughout the sample volume contribute to the signal. All four configurations also remain sensitive to axial misalignment of the receiver coils relative to the pre-polariser, a residual systematic uncertainty that can only be eliminated by re-designing the pre-polariser as a whole.

### 4.2.6 Next steps

Two improvements would substantially strengthen the next iteration of this measurement. A redesigned probe jig with a locating rod, a fixed-pitch graduated scale, and a snug fit along the full bore would simultaneously fix the probe orientation and reduce positional uncertainty to roughly ±0.1 mm. Repeating the scan at several radii would then yield a two-dimensional map of $|B|$ over the sample volume, allowing direct computation of $\Delta B/B_0$ in the form quoted by the EFNMR literature. Performing the measurement on a non-magnetic bench would remove the metal-desk uncertainty and is a simple change to make. Beyond this, the configuration sweep should be widened to include changes in turn count and coil geometry and the noise floor should be re-characterised on a spectrum analyser to give a frequency-resolved breakdown of contributions.

## 4.3 Square-wave pulsing

### 4.3.1 Hypothesis

We tested the hypothesis that the square-wave pulses emitted by the microcontroller, smoothed through a simple RLC resonant circuit (a 47.4 mH receiver coil, a 7.31 kΩ resistor, and a tunable capacitor in parallel with the coil), could mimic a sine wave at the proton Larmor frequency closely enough to deliver sufficient power to drive EFNMR - providing a substantially simpler and lower-cost alternative to the dedicated sine-wave generation circuit used by the previous cohort B.3[17,18]. The motivation was practical: the previous group's sine-wave circuit was complex enough that we judged we could not reliably reproduce and debug it within the project timeframe.

### 4.3.2 Power transfer efficiency vs. smoothing capacitance

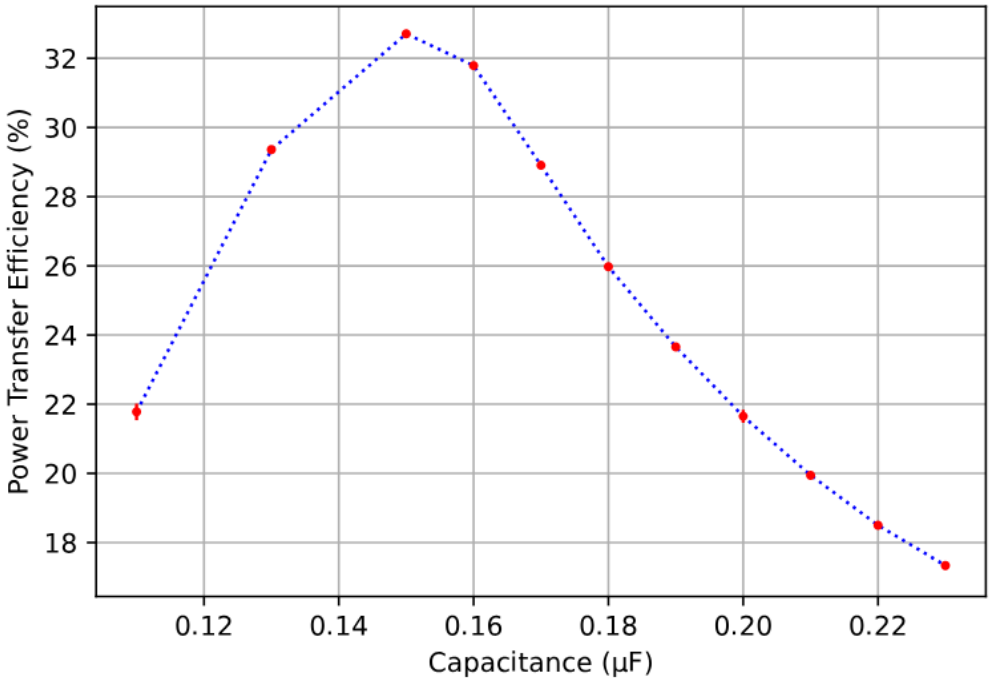



Figure 4.3: Power transfer efficiency vs. smoothing capacitance for the resonant pulsing circuit. Error bars correspond to 1.96 standard deviations (95% confidence interval); for most data points they are too small to be visible on the graph.

We define the power transfer efficiency as the ratio of the area under the spectral peak of the smoothed square wave at the Larmor frequency (2103 Hz) to the area under the spectral peak of an idealised finite-duration sine wave at the same frequency. Figure 4.3 shows how this efficiency varies as the smoothing capacitor is swept from 0.01 µF to 0.31 µF. The

efficiency exhibits a clear peak at a smoothing capacitance of 0.15 µF, at which value the smoothed square wave most closely approximates a sinusoid at the Larmor frequency and the spectral overlap with the resonance is maximised. This value was therefore adopted for the final apparatus, where it appears as the 150 nF capacitor in parallel with the receiver coil in the pulsing circuit (Section 3.1). The peak efficiency is sufficient to drive the CPMG sequence on the pulsing side, supporting the hypothesis: the square-wave approach with simple RLC smoothing is viable for EFNMR at this field strength.

### 4.3.3 Comparison with the sine-wave approach

What we cannot conclude from this measurement alone is whether the smoothed square-wave method is superior, inferior, or equivalent to a true sine-wave drive in absolute power-transfer efficiency. A direct comparison would require constructing both circuits and measuring the spectral overlap between each drive waveform and the spin resonance under identical conditions - specifically, the integral of the drive's power spectral density within the resonance bandwidth. We did not construct the sine-wave circuit, so this comparison is left open.

Independent of relative efficiency, the smoothed-square-wave configuration has practical advantages that matter directly for the project's cost and operability requirements. The drive circuit comprises three separate sections including numerous op-amps, resistors and capacitors, at a total component cost of approximately £20 more than our design. Pulse amplitude in our design is set by a single resistor, rather than by tuning a resonant LC network, which makes amplitude adjustment and field-strength calibration straightforward. These attributes are well aligned with the educational-outreach use case: a school-based operator should not need to retune a resonant circuit between samples.

### 4.3.4 Limitations

The principal limitation of this investigation is the absence of a back-to-back comparison with a sine-wave drive built and characterised under identical conditions. Without it, claims about relative efficiency rest on theoretical expectations rather than measurement. A second limitation is that the square wave's harmonic content (odd harmonics at $3f_0$, $5f_0$, $7f_0$, ...) deposits power outside the resonance window; quantifying this loss would require measuring the waveform's full Fourier spectrum and integrating the out-of-band fraction, rather than relying only on the central-peak overlap reported here.

### 4.3.5 Next steps

The natural extension of this work is to re-construct the previous cohort's sine-wave drive, paired to the same coils and detection chain, and to drive each circuit through the same CPMG sequence on the same sample. Comparing the resulting echo amplitudes directly is

the cleanest test of relative efficiency. Measuring each waveform's spectrum on a spectrum analyser, and computing the fraction of input power that falls within the resonance bandwidth, would convert the efficiency comparison from theoretical to quantitative.

## 4.4 Pulsing code and microcontrollers

### 4.4.1 Hypothesis

We hypothesised that the original Raspberry Pi Pico (RP2040) microcontroller used by previous cohorts has suboptimal temporal resolution for delivering CPMG pulses at the proton Larmor frequency, and that upgrading to the Raspberry Pi Pico 2 (RP2350) - a drop-in replacement with a more powerful core, double the flash memory and SRAM, at a similar price point - would reduce per-pulse overhead. Independently, we hypothesised that compiled C++ firmware would produce more accurate Larmor-frequency pulses than the MicroPython implementation inherited from the previous cohort[17,18], because compiled languages execute substantially closer to native hardware speeds than interpreted ones. Together, these changes were expected to bring the mean pulse frequency closer to the target Larmor frequency and improve power transfer at that frequency.

### 4.4.2 Hardware comparison: Pico 1 vs. Pico 2

| Microcontroller | Pico 1 | | Pico 2 | |
|---|---|---|---|---|
| Language | MicroPython | C++ | MicroPython | C++ |
| Overhead time (µs) | 347.1 | 35.5 | 192.0 | 23.5 |
| Error in overhead time | 6.27 | 0.991 | 0.00 | 1.01 |
| Overhead (%) | 19.2 | 1.97 | 10.6 | 1.30 |
| Error in overhead | 0.347 | 0.0548 | 0.00 | 0.0557 |

Table 4.2: Comparison of overhead times and overhead percentages for MicroPython and C++ on the Pico 1 and Pico 2 microcontrollers.

Table 4.2 shows the mean per-pulse overhead measured for each combination of microcontroller and firmware language, averaged over 10 iterations per condition. For either language, the Pico 2 substantially reduces the overhead compared with the Pico 1: MicroPython falls from 347.1 µs to 192.0 µs (a 44.7% reduction), and C++ from 35.5 µs to 23.5 µs (a 33.8% reduction). The Pico 2's faster clock speed and improved instruction throughput therefore translate directly into more precise pulse timing, independent of language. For a given microcontroller, C++ reduces overhead by roughly an order of magnitude over MicroPython - 89.8% on the Pico 1 and 87.8% on the Pico 2 - which

is consistent with the expected performance gap between interpreted and compiled code. The differences between hardware and between languages are both large compared to the reported uncertainties, so the hardware-upgrade hypothesis is supported, and the Pico 2 was selected for all subsequent measurements.

### 4.4.3 Firmware comparison: MicroPython vs. C++

| Language | MicroPython | C++ |
|---|---|---|
| Mean pulse frequency (Hz) | 2001.7 | 2062.9 |
| Pulse frequency error | 3.0 | 5.6 |
| Coefficient of variation | 0.00479 | 0.00860 |
| Standard deviation from Larmor (Hz) | 101.7 | 43.5 |
| Power transfer efficiency (%) | 32.71 | 32.85 |
| Power transfer efficiency error | 0.0113 | 0.0136 |

Table 4.3: Comparison of pulsing performance between MicroPython and C++ on the Raspberry Pi Pico 2.

Table 4.3 examines the downstream consequences of the overhead reduction for actual pulsing performance on the Pico 2, averaged over 10 pulses per condition. The two languages exhibit a notable trade-off between accuracy and precision in pulse frequency. C++ has a higher shot-to-shot coefficient of variation (0.008 60 vs. 0.004 79 for MicroPython) and a larger pulse-frequency error (5.6 Hz vs. 3.0 Hz), making it the less precise of the two. However, the mean C++ pulse frequency of 2062.9 Hz sits only ≈40 Hz below the target Larmor frequency of 2103 Hz, whereas MicroPython produces a mean of 2001.7 Hz - a systematic bias of approximately 100 Hz. As a result, the overall deviation from the Larmor frequency is more than halved with C++ (43.5 Hz vs. 101.7 Hz). In other words, MicroPython produces pulses that are tightly clustered around the wrong frequency, while C++ produces pulses that are more scattered but centred closer to the target.

For driving a resonant pulsing circuit, accuracy matters more than precision: power transfer at the Larmor frequency is governed by the spectral overlap of the pulse train with the resonance, which is dominated by the offset of the mean pulse frequency from Larmor rather than by the shot-to-shot scatter (provided the latter remains small compared to the circuit's bandwidth). The measured power transfer efficiencies are consistent with this expectation: 32.85% for C++ versus 32.71% for MicroPython. The difference is marginal - lying only just above the combined uncertainty range - and on its own cannot be regarded as a statistically significant improvement in power transfer. Its direction, however, is consistent with the accuracy-versus-precision argument, and the substantial reduction in overhead and the

improvement in mean pulse-frequency accuracy are independently valuable. On this basis we adopted C++ in the final implementation. The firmware hypothesis is therefore supported with caveats: C++ improves the metrics that matter most for resonant pulsing (overhead and mean-frequency accuracy) while underperforming on shot-to-shot precision, and the change in end-to-end power-transfer efficiency is at the edge of measurement uncertainty.

A practical secondary advantage of the lower overhead and higher throughput of the C++ firmware is that they ease subsequent time-resolution demands on the detection circuit: the Larmor period of approximately 0.5 ms sets the timescale on which the FID and CPMG echo must be sampled, and achieving the necessary sampling rate with high precision becomes substantially easier when the pulse generator imposes only ∼23.5 µs of overhead per pulse rather than ∼192 µs.

### 4.4.4 Limitations

For the comparisons reported above the statistical robustness is adequate: the Pico 1 vs. Pico 2 overhead difference and the C++ vs. MicroPython mean-frequency difference both exceed the reported uncertainties by well over an order of magnitude. The end-to-end power-transfer efficiency difference between the two languages, however, sits only just outside the combined uncertainty range, and at 10 iterations per condition we cannot resolve a more subtle effect of this size with confidence. With $n = 10$, the standard error on the mean is $s/\sqrt{10} \approx 0.32\,s$, where $s$ is the per-iteration standard deviation, so resolving a difference of a few tenths of a percentage point in power-transfer efficiency would require either an order of magnitude more iterations or a substantial reduction in $s$. Python's `timeit` module, which would routinely run $10^3$–$10^6$ iterations, is not directly available for the C++ side; the equivalent approach is to time a tight pulse-sequence loop using hardware-timestamped edges (the Pico's PIO or DMA-driven timer counters), which we did not implement.

The frequency-deviation measurement was limited by the oscilloscope's frequency-readout resolution rather than by sample size alone. Readings clustered on discrete values separated by approximately 10 Hz at our sampling settings, so differences smaller than this step cannot be detected by this method regardless of how many traces are averaged.

A separate methodological issue concerns how power transfer was inferred from the drive waveform. We approximated the central peak of each drive's Fourier spectrum with a Gaussian and integrated under that fit. This is a useful first approximation but it does not exactly describe the spectrum of our pulses. The Fourier transform of a continuous sinusoid is a pair of delta functions at $\pm f_0$; truncating that sinusoid to a finite-duration pulse of length $T$ broadens those deltas into a *sinc*-shaped lineshape whose full width scales as $\sim 1/T$. A square-wave pulse train introduces additional *sinc*-shaped peaks at every odd harmonic of $f_0$. A Gaussian fits the central lobe of a *sinc* tolerably near its peak, but underestimates the wings and ignores the harmonics entirely, so the Gaussian-area estimate is biased downward

by an amount that depends on pulse duration and harmonic content. A more faithful estimate would integrate the measured spectrum numerically within a defined resonance bandwidth, or fit the expected $sinc^2$ power spectrum directly. The C++ and MicroPython efficiencies share the same systematic bias from this approximation, so their ranking is more reliable than their absolute magnitudes, but the small absolute difference between them makes the metric less informative than the underlying overhead and pulse-frequency measurements.

### 4.4.5 Next steps

Three improvements are worth prioritising. The Pico's on-chip hardware can timestamp pulse edges with single-clock-cycle resolution (8 ns on the default 125 MHz clock), which would remove the oscilloscope as a precision bottleneck and allow the iteration count to grow into the thousands. The frequency-domain analysis should be done by direct numerical integration of the measured spectrum rather than by Gaussian fitting; this also opens the door to quantifying the harmonic-power loss noted in Section 4.3. Finally, a tight per-iteration loop with on-chip timestamping would yield a per-pulse jitter histogram, which is a more informative description of timing reproducibility than a single mean and standard deviation.

## 4.5 Implications for educational outreach

The findings above bear directly on whether the spectrometer is fit for its intended use in schools and colleges. Three points stand out. First, the smoothed-square-wave drive's simplicity means that a non-expert operator can change samples and re-acquire by re-calibrating the single resistor and capacitor very easily - a non-trivial advantage over a dedicated sine-wave design in a classroom context where teachers and students cannot be expected to debug an entire pre-amplifier circui mid-demonstration. Second, the C++/Pico 2 firmware combination reduces the per-pulse overhead to 23.5 µs and brings the mean pulse frequency to within 40 Hz of the proton Larmor frequency, giving the pulse generator enough margin to deliver the CPMG sequence reliably within a short acquisition time. Third, the counter-wound double-coil configuration extracts the largest available signal from the pre-polariser while keeping the noise floor at the same level as a single-coil arrangement, which improves the chances of a clear, classroom-visible CPMG echo once the detection circuit is integrated into the final design.

Two important caveats temper this verdict. The cost ceiling of £250 was met comfortably (£178.65) and the CPMG sequence is fully implemented; against these, the spectrometer is not yet portable enough for off-site use, and the time-resolution requirement cannot be evaluated until the detection circuit characterised in the companion report[24,25] is integrated. The methodological limitations identified earlier in this chapter - noise characterisation, field homogeneity, and timing precision - affect the spectrometer's quantitative scientific accuracy

but not its qualitative behaviour: provided the detection chain delivers a usable signal, the pulsing side is now configured to drive a clear CPMG echo, and that is the relevant threshold for the outreach use case.

The two priorities for the next cohort, as far as outreach readiness is concerned, are therefore (i) the integration and packaging work needed to make the device genuinely portable in the sense required by Table 4.1, and (ii) the back-to-back comparison with a sine-wave drive recommended in Section 4.3, which would either confirm the smoothed-square-wave drive as the preferred long-term design or motivate a return to a (now better-baselined) sine-wave approach.

# Chapter 5

# Conclusion

This project set out to advance the departmental EFNMR spectrometer toward a state in which it can be used reliably for educational outreach at a total cost below £250. The headline contribution is the successful implementation of the CPMG sequence using a C++ based firmware for the RP2350 microcontroller. This substantially reducde the performance overheads and brought the CPMG pulse sequence frequency drastically closer to the targeted Larmor frequency. Furthermore, we demonstrated that a smoothed-square-wave drive, generated directly by the microcontroller through a simple RLC resonant circuit, delivers sufficient power at the proton Larmor frequency to drive the CPMG sequence. This provides a substantially simpler and lower-cost alternative to the dedicated sine-wave generation circuit designed by the previous cohort B.3. Additionally, we demonstrated that optimal receiver coil configuration to maximise the SNR is the counter-wound double coil arrangement. This takes advantage of gradiometric noise cancellation while doubling the sample volume relative to a single receiver coil. Three investigations supported these outcomes: an evaluation of receiver-coil configurations within the pre-polariser, the construction and characterisation of the smoothed-square-wave drive itself, and a comparison of C++ and MicroPython firmware on the Raspberry Pi Pico 2 for pulse-sequence timing.

Measured against the design requirements set out in Table 1.1, the instrument met the cost target at £178.65, and fully implemented the CPMG sequence on the pulsing side; the full evaluation is given in Table 4.1. Quantitatively, the counter-wound double-coil configuration matched the low noise floor of a single coil while doubling the usable sample volume to 580 mL, and the C++/Pico 2 firmware reduced per-pulse overhead by a factor of approximately eight relative to the MicroPython implementation, bringing the mean pulse frequency to within 40 Hz of the 2103 Hz proton Larmor target. No statistically resolvable difference in end-to-end power-transfer efficiency between the two firmware languages was found. This suggests further development effort is better directed at the drive circuit and detection chain than at the firmware.

Taken together, the three findings address a common problem: making EFNMR work reliably at low cost without the tuning expertise that high-field instruments demand. The

coil-configuration result keeps the noise floor at its single-coil minimum while restoring the larger sample volume needed to lift a weak Earth's-field signal above it; the smoothed-square-wave drive removes the resonant LC pre-amplifier network that would otherwise need complicated re-tuning between samples; and the C++/Pico 2 firmware reduces pulse-timing overhead to the point at which the detection circuit, rather than the pulse generator, becomes the limiting factor on time resolution. In combination these gains move the instrument meaningfully closer to a device a non-expert can operate, and represent the first iteration of the project to demonstrate the CPMG sequence using a smoothed-square-wave drive in place of a dedicated sine-wave circuit. Subject to the caveats discussed in Chapter 4, mainly the integration of the detection circuit[24,25] and the packaging work needed for portability - the spectrometer's pulsing circuit is functional.

The most productive next steps for the following cohort follow directly from the limitations identified in Chapter 4. The two priorities are: (i) the integration and packaging work needed to make the device genuinely portable, since cost and pulsing functionality are now meeting design targets but portability is not; and (ii) a back-to-back comparison of the smoothed-square-wave drive against the previous cohort's sine-wave drive, which would close the question of relative power-transfer efficiency that this report leaves open. Beyond these, a two-dimensional field map of the pre-polariser using a fixed-orientation probe on a graduated stage would replace the central-axis homogeneity figure with something directly comparable to the EFNMR literature, and replacing oscilloscope-based pulse timing with hardware-timestamped edges on the Pico's PIO subsystem would lift the precision ceiling on the firmware comparison by several orders of magnitude. With these in place, the spectrometer should be in a position to leave the laboratory for the schools and colleges it was designed to reach.

# Bibliography


[1] JEOL Ltd, *NMR spectrometer ECZ Luminous ™ JNM-ECZL series FT NMR | Products | JEOL Ltd.* NMR spectrometer ECZ Luminous™ JNM-ECZL series FT NMR | Products | JEOL Ltd., https://www.jeol.com/products/scientific/nmr/JNM-ECZL.php (visited on 12/06/2024).

[2] I. I. Rabi, J. R. Zacharias, S. Millman, and P. Kusch, "A new method of measuring nuclear magnetic moment", Physical Review **53**, 318 (1938).

[3] E. M. Purcell, H. C. Torrey, and R. V. Pound, "Resonance absorption by nuclear magnetic moments in a solid", Physical Review **69**, 37 (1946).

[4] E. N. Bassey, P. J. Reeves, I. D. Seymour, and C. P. Grey, "17O NMR Spectroscopy in Lithium-Ion Battery Cathode Materials: Challenges and Interpretation", Journal of the American Chemical Society **144**, 18714 (2022).

[5] H. N. Cheng, T. Asakura, K. Suganuma, J. M. Lagaron, B. Melendez-Rodriguez, and A. Biswas, "NMR Analyses and Statistical Modeling of Biobased Polymer Microstructures—A Selected Review", Polymers **16**, 620 (2024).

[6] National Institute of Health, *Magnetic resonance imaging (MRI)*, (2022) https://www.nibib.nih.gov/science-education/science-topics/magnetic-resonance-imaging-mri.

[7] M. Packard and R. Varian, "Free nuclear induction in Earth's magnetic field", Physical Review **A93** (1954).

[8] B. Kenyon, R. Kleinberg, C. Straley, G. Gubelin, and C. Morriss, "Nuclear magnetic resonance imaging - technology for the 21st century", Oilfield Review **7**, 19 (1995).

[9] P. T. Callaghan, C. D. Eccles, T. G. Haskell, P. J. Langhorne, and J. D. Seymour, "Earth's field NMR in Antarctica: A pulsed gradient spin echo NMR study of restricted diffusion in sea ice", Journal of Magnetic Resonance **133**, 148 (1998).

[10] P. T. Callaghan and M. L. Gros, "Nuclear spins in the earth's magnetic field", American Journal of Physics **50**, 709 (1982).

[11] TeachSpin Inc., *Pricing — TeachSpin*, (2024) https://www.teachspin.com/pricing.

[12] C. A. Michal, "A low-cost spectrometer for NMR measurements in the Earth's magnetic field", Measurement Science and Technology **21**, 105902 (2010).

[13] J. Hornak, *The basics of NMR*, (2017) https://www.cis.rit.edu/htbooks/nmr/inside.htm.

[14] Bruker Corp, *GHz Class NMR | Ultra High Magnetic Field*, https://www.bruker.com/en/products-and-solutions/mr/nmr/ascend-ghz-class.html (visited on 12/08/2024).

[15] C. C. Finlay et al., "International geomagnetic reference field: The eleventh generation", Geophysical Journal International **183**, 1216 (2010).

[16] D. I. Hoult and R. E. Richards, "The signal-to-noise ratio of the nuclear magnetic resonance experiment", Journal of Magnetic Resonance (1969) **24**, 71 (1976).

[17] M. Włodarczyk, "Low-cost Earth's field NMR spectrometer" (The University of Manchester, 2024).

[18] M. Włodarczyk, "Pulse sequencing in low-cost Earth's field NMR spectrometer" (The University of Manchester, 2024).

[19] E. Asquith, *Constructing a low-cost Earth's field NMR device*, 2024.

[20] E. Asquith, "Spin-echo pulsing in a low-cost earth's field NMR device", MA thesis (University of Manchester, 2024).

[21] L. Mestre, "Signal collection system for earth's field NMR", MA thesis (University of Manchester, 2021).

[22] A. Raif, "Designing a low-cost earth's field NMR system", MA thesis (University of Manchester, 2024).

[23] E. Hattersley, "Low-Cost Earth's Field Nuclear Magnetic Resonance", MA thesis (The University of Manchester, 2024).

[24] G. Champion, "Master's Dissertation", MA thesis (The University of Manchester, 2024).

[25] J. McAuley, "Master's Dissertation", MA thesis (The University of Manchester, 2024).

[26] J. W. Hennel and J. Klinowski, in (Longman Scientific and Technical, 1993), p. 61.

[27] P. Zeeman, "XXXII. *On the influence of magnetism on the nature of the light emitted by a substance*", The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science **43**, 226 (1897).

[28] S. D. Serai, "Basics of magnetic resonance imaging and quantitative parameters T1, T2, T2*, t1rho and diffusion-weighted imaging", Pediatric Radiology **52**, 217 (2022).

[29] National Institute of Standards, *Codata, the NIST reference on constants, units and uncertainty*, (2024)

[30] D. B. Newell and E. Tiesinga, *The international system of units (si)* (National Institute of Standards Special Publication 330, 2019).

[31] M. H. Levitt, *Spin dynamics: Basics of nuclear magnetic resonance* (Wiley, 2001).

[32] G. Planinšič, J. Stepišnik, and M. Kos, "Relaxation-time measurement and imaging in the Earth's magnetic field", Journal of Magnetic Resonance, Series A **110**, 170 (1994).

[33] L. Parkes, *Physics of medical imaging lectures*, University of Manchester, 2022.

[34] D. G. Mitchell, D. L. Burk, S. Vinitski, and M. D. Rifkin, "The biophysical basis of tissue contrast in extracranial MR imaging", AJR. American journal of roentgenology **149**, 831 (1987).

[35] F. Bloch, "Nuclear induction", Physical Review **70**, 460 (1946).

[36] M. Schle, *NMR signal : Free induction decay and spin echo*, (2009)

[37] E. L. Hahn, "Spin echoes", Physical Review **80**, 580 (1950).

[38] H. Y. Carr and E. Purcell, "Effects of diffusion on free precession in nuclear magnetic resonance experiments", Physical Review **94** (1954).

[39] S. Meiboom and D. Gill, "Modified spin-echo method for measuring nuclear relaxation time", Review of Scientific Instruments **29** (1958).

[40] James. Keeler, *Understanding NMR spectroscopy*, 2nd ed. (John Wiley and Sons, Chichester, U.K, 2010).

[41] H. C. Torrey, "Bloch Equations with Diffusion Terms", Physical Review **104**, 563 (1956).

[42] G. B. Chavhan, P. S. Babyn, B. Thomas, M. M. Shroff, and E. M. Haacke, "Principles, techniques, and applications of t2*-based mr imaging and its special applications1", Radiographics : a review publication of the Radiological Society of North America, Inc **29**, 1433 (2009).

[43] R. R. Ernst and W. A. Anderson, "Application of Fourier Transform Spectroscopy to Magnetic Resonance", Review of Scientific Instruments **37**, 93 (1966).

[44] J. B. J. Fourier, *Théorie Analytique de la Chaleur*, 1st ed. (Cambridge University Press, July 20, 2009).

[45] J. W. Cooley and J. W. Tukey, "An Algorithm for the Machine Calculation of Complex Fourier Series", (1965).

[46] SciPy Developers, *Fft — SciPy v1.17.0 Manual*, `https://docs.scipy.org/doc/scipy/reference/generated/scipy.fft.fft.html#scipy.fft.fft` (visited on 11/28/2024).

[47] Martin Reinecke, *Martin Reinecke / pocketfft · GitLab*, GitLab, (May 10, 2019) https://gitlab.mpcdf.mpg.de/mtr/pocketfft (visited on 11/30/2024).

[48] Tektronix Inc., *Digital storage oscilloscope TBS1000C series datasheet*, manual (Tektronix Inc., 2020).

[49] X. Huang, H. Dong, Q. Tao, M. Yu, Y. Li, L. Rong, H.-J. Krause, A. Offenhäusser, and X. Xie, "Sensor Configuration and Algorithms for Power-Line Interference Suppression in Low Field Nuclear Magnetic Resonance", Sensors (Basel, Switzerland) **19**, 3566 (2019).

[50] J. C. Maxwell, "VIII. A dynamical theory of the electromagnetic field", Philosophical Transactions of the Royal Society of London, 459 (1865).

[51] J. B. Johnson, "Thermal Agitation of Electricity in Conductors", Physical Review **32**, 97 (1928).

[52] H. Nyquist, "Thermal Agitation of Electric Charge in Conductors", Physical Review **32**, 110 (1928).

[53] C. A. Mack, "Reducing roughness in extreme ultraviolet lithography", Journal of Micro/Nanolithography, MEMS, and MOEMS **17**, 041006 (2018).

[54] W. Schottky, "Small-Shot Effect and Flicker Effect", Physical Review **28**, 74 (1926).

[55] M. B. Cohen, R. K. Said, and U. S. Inan, "Mitigation of 50-60 Hz power line interference in geophysical data: POWER LINE INTERFERENCE MITIGATION", Radio Science **45**, n/a (2010).

[56] D. Sinha, "Generation of acoustic-Brownian noise in nuclear magnetic resonance under non-equilibrium thermal fluctuations", Scientific Reports **10**, 21406 (2020).

[57] Texas Instruments Inc., *CMOS quad low-to-high voltage level shifter*, manual (Texas Instruments, 2003).

# Appendix A

# Appendix - Table

| Component | Unit cost [£] | No. used | Total Cost [£] |
|---|---|---|---|
| Pre-polariser coil | 100 | 1 | 100 |
| Receiver coil | 21.05 | 2 | 42.10 |
| PVC cylinder | 15.00 | 1 | 15.00 |
| Pico 2 (RP2350) | 4.80 | 1 | 4.80 |
| LLC (CD40109BE) | 0.52 | 1 | 0.52 |
| SPDT Relay (PE014009) | 3.45 | 2 | 6.90 |
| Resistor | 0.02 | 11 | 0.22 |
| Capacitor (S503Z69Z5UL63L0R) | 1.47 | 1 | 1.47 |
| 9 V Battery | 2.90 | 2 | 5.80 |
| Op-amp (UA741CP) | 0.6 | 3 | 1.84 |

Table A.1: Summary of components used in spectrometer design and their costs.

The total cost of the whole NMR spectrometer design is £178.65 as calculated by Emma Hattersley[23].

# Appendix B

# Appendix - Figures

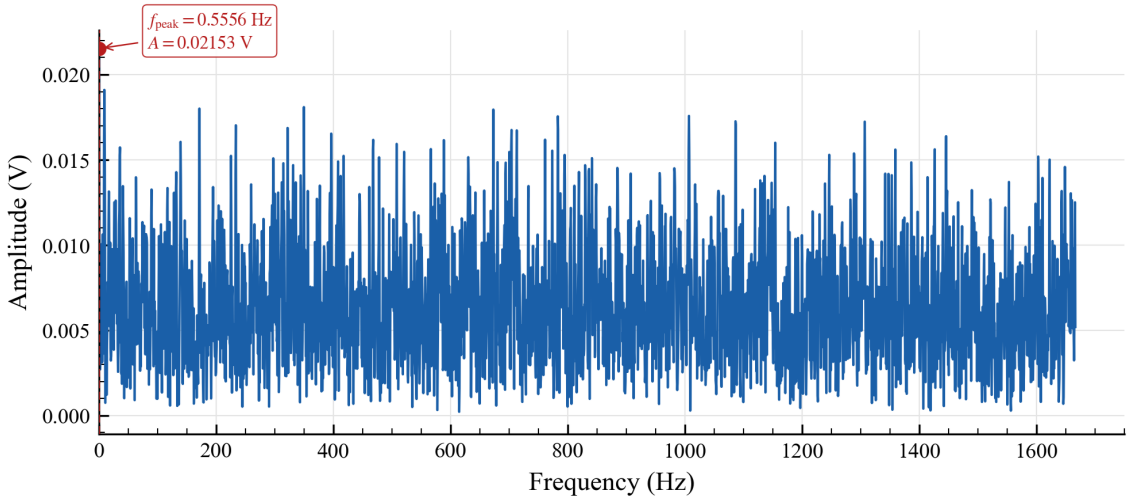



Figure B.1: Fourier transform of noise floor data for counter-wound configuration

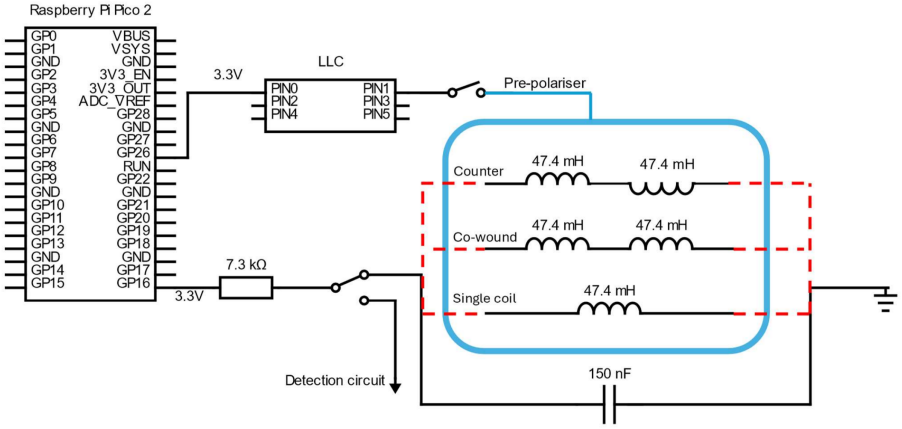



Figure B.2: Schematic showing tested coil arrangements

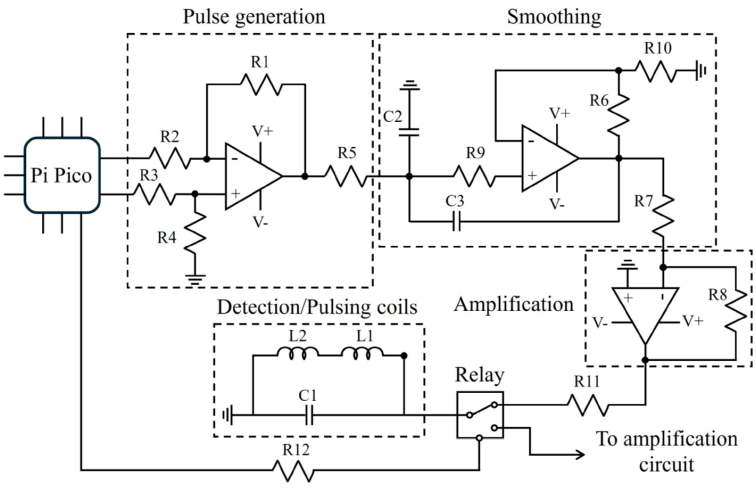



Figure B.3: Previous cohort's pulsing circuit[17–20,22]

# Appendix C

# Appendix - Code

## C.1 C++ Microcontroller Code

The code for the C++ firmware was written entirely by myself (Sajid Ahmed), and is a functionally equivalent translation of the MicroPython code in the next section. Therefore, I have made this code available under the MIT open-source license so anyone can freely use, modify, and distribute it.

```
#include <Arduino.h>
// Define all pins
#define LED_PIN 25          // LED pin
#define ADC_PIN A2          // ADC pin (corresponds to Pin 28 in MicroPython)
#define PULSE_PIN 16        // Pulse pin
#define PP_COIL_PIN 26      // Prepolarisation coil pin
#define DET_SWITCH_PIN 22   // Detection switch pin

// Variables
int sleep_time = 0;
int datasize = 0;
const float freq = 2210.0;              // Larmor frequency in Hz
const float period = 1000000.0 / freq; // Period in microseconds
const int pulse_time = (int)(period / 2); // Pulse time in microseconds
const int turn_on_time = 20;             // Delay before switching modes
↪  (microseconds)

// Function prototypes
void pulse_gen(int size);
float find_freq();
void fid_command(int sleep_time, int datasize);
void echo_command(int sleep_time, int datasize, int wait_time);

void setup() {
  // Initialise serial communication
  Serial.begin(9600);
```

```
26
27    // Initialise pins
28    pinMode(LED_PIN, OUTPUT);
29    pinMode(PULSE_PIN, OUTPUT);
30    pinMode(PP_COIL_PIN, OUTPUT);
31    pinMode(DET_SWITCH_PIN, OUTPUT);
32    pinMode(ADC_PIN, INPUT);
33
34    Serial.println("Ready for commands...");
35  }
36
37  void loop() {
38    if (Serial.available() > 0) {
39      String command = Serial.readStringUntil('\n'); // Read input command
40      command.trim();                                // Remove extra whitespace
41
42      String list_of_commands[4];
43      int idx = 0;
44
45      // Split the command into parts
46      int start = 0;
47      for (int i = 0; i < command.length(); i++) {
48        if (command[i] == ',') {
49          list_of_commands[idx++] = command.substring(start, i);
50          start = i + 1;
51        }
52      }
53      list_of_commands[idx] = command.substring(start);
54
55      // Handle commands
56      if (list_of_commands[0] == "FID") {
57        sleep_time = list_of_commands[1].toInt();
58        datasize = list_of_commands[2].toInt();
59        fid_command(sleep_time, datasize);
60      } else if (list_of_commands[0] == "echo") {
61        sleep_time = list_of_commands[1].toInt();
62        datasize = list_of_commands[2].toInt();
63        int wait_time = list_of_commands[3].toFloat() * 1e6;
64        echo_command(sleep_time, datasize, wait_time);
65      } else {
66        Serial.println("Incorrect command string");
67      }
68    }
69  }
70
71  void pulse_gen(int size) {
72    for (int i = 0; i < size; i++) {
73      digitalWrite(PULSE_PIN, HIGH);
```

```
    delayMicroseconds(pulse_time);
    digitalWrite(PULSE_PIN, LOW);
    delayMicroseconds(pulse_time);
  }
}

float find_freq() {
  int y[10], x[10];
  unsigned long start = micros();
  for (int i = 0; i < 10; i++) {
    y[i] = analogRead(ADC_PIN);
    x[i] = micros() - start;
    delayMicroseconds(sleep_time);
  }

  int sum = 0;
  for (int i = 0; i < 9; i++) {
    sum += (x[i + 1] - x[i]);
  }

  return 9.0 / sum;
}

void fid_command(int sleep_time, int datasize) {
  int values[datasize], times[datasize];

  digitalWrite(LED_PIN, HIGH);
  digitalWrite(PP_COIL_PIN, HIGH);
  delay(3000);
  digitalWrite(PP_COIL_PIN, LOW);

  pulse_gen(1);
  digitalWrite(DET_SWITCH_PIN, HIGH);

  unsigned long start = micros();
  for (int i = 0; i < datasize; i++) {
    values[i] = analogRead(ADC_PIN);
    times[i] = micros() - start;
    delayMicroseconds(sleep_time);
  }

  digitalWrite(DET_SWITCH_PIN, LOW);
  digitalWrite(LED_PIN, LOW);

  // Output data as CSV
  for (int i = 0; i < datasize; i++) {
    Serial.print(times[i]);
    Serial.print(",");
```

```
    Serial.println(values[i]);
  }
}

void echo_command(int sleep_time, int datasize, int wait_time) {
  int values[datasize], times[datasize];
  digitalWrite(LED_PIN, HIGH);
  int no_samples = wait_time * find_freq();
  int samples_taken = 0;

  digitalWrite(PP_COIL_PIN, HIGH);
  delay(3000);
  digitalWrite(PP_COIL_PIN, LOW);

  pulse_gen(1);
  digitalWrite(DET_SWITCH_PIN, HIGH);

  unsigned long start = micros();
  while (samples_taken < datasize) {
    for (int i = 0; i < min(no_samples, datasize - samples_taken); i++) {
      values[samples_taken + i] = analogRead(ADC_PIN);
      times[samples_taken + i] = micros() - start;
      delayMicroseconds(sleep_time);
    }
    samples_taken += no_samples;

    digitalWrite(DET_SWITCH_PIN, LOW);
    delayMicroseconds(turn_on_time);
    pulse_gen(2);
    digitalWrite(DET_SWITCH_PIN, HIGH);
    delayMicroseconds(turn_on_time);
  }

  digitalWrite(LED_PIN, LOW);

  // Output data as CSV
  for (int i = 0; i < datasize; i++) {
    Serial.print(times[i]);
    Serial.print(",");
    Serial.println(values[i]);
  }
}
```

## C.2 MicroPython microcontroller firmware

The MicroPython code in this section has been modified by me (Sajid Ahmed) and is based on the previous cohort's iteration written by Michal Wlodarczyk[17,18], which itself was based on an early iteration written by Dr Patrick Parkinson. I am unsure whether this version is sufficiently different from their private codebases to be considered a separate work, therefore I present this code only for review & evaluation purposes and advise against unauthorised usage, modification and distribution.

```
import time  # for time functions
import ujson as json  # imports module for data serialisation
import array  # for storing data in efficient, fixed type arrays
import utime  # timing
from machine import Pin, ADC  # controls GPIO pins of Pico
import usys as sys  # access system level functionality

# Defining all the pins
led = Pin(25, Pin.OUT)
# ADC on Pin 26 is being used to take readings, converting the analogue spin echo
↪  signals to binary data.
adc = ADC(Pin(28))
pulse = Pin(16, Pin.OUT)
pp_coil = Pin(26, Pin.OUT)  # prepol
det_switch = Pin(22, Pin.OUT)  # turns relay for detection circuit

datasize = None
sleep_time = None
freq = 2210  # Larmor frequency in Hz
period = 1_000_000/freq
pulse_time = int(period/2)  # pulse time in microseconds
turn_on_time = 20  # delay before switching modes, in microseconds


def pulse_gen(size):
    # Sends a pulse that can either be size = 1 or 2 , coresponding to 90 and 180
    ↪  degrees.
    for i in range(size):
        pulse.on()
        utime.sleep_us(pulse_time)
        pulse.off()
        utime.sleep_us(pulse_time)


def find_freq(a_d_c):
    # Measures and computes the frequency of signal acquisition from the ADC.
    # Uses timestamps and values to estimate frequency based on time differences
    ↪  between readings.
```

```
    y = array.array('i', [0] * 10)
    x = array.array('i', [0] * 10)
    start = utime.ticks_us()
    for i in range(10):
        y[i] = a_d_c.read_u16() >> 4
        x[i] = utime.ticks_diff(utime.ticks_us(), start)
        utime.sleep_us(sleep_time)
    sum = 0
    i = 0
    while i < len(x) - 1:
        sum += x[i+1] - x[i]
        i += 1
    return 9 / sum


while (True):
    command = sys.stdin.readline()  # listens for commands from system inputs
    list_of_commands = command.split(',') # split each command with a comma

    if list_of_commands[0] == 'FID':  # if the 1st input == 'FID'
        # get values and define necessary lists
        # makes sleep_time the 2nd input, this is the interval between pulses and
        ↪  readings
        sleep_time = int(list_of_commands[1])
        # makes datasize the 3rd input, this is the number of readings to be
        ↪  taken
        datasize = int(list_of_commands[2])
        values = array.array('i', [0] * datasize)  # set up array for readings
        # set up array for intervals between readings
        times = array.array('i', [0] * datasize)

        # signal that the measurement has started
        led.on()

        # pre-polarise the sample
        pp_coil.on()
        time.sleep_ms(10000)  # using milliseconds for more precise timing
        pp_coil.off()

        # apply a 90 pulse
        pulse_gen(1)

        # switch to detection mode
        det_switch.on()

        # record data
        start = utime.ticks_us()
```

```
        # Reads data from ADC at intervals defined by sleep_time, recording both values and timestamps
        for i in range(datasize):
            values[i] = adc.read_u16() >> 4
            times[i] = utime.ticks_diff(utime.ticks_us(), start)
            utime.sleep_us(sleep_time)
        print(json.dumps(times + values))

        det_switch.off()  # turns detection circuit off
        led.off()

    elif list_of_commands[0] == 'echo':  # else if 1st system input command == echo
        # get values and define necessary lists
        sleep_time = int(list_of_commands[1])
        datasize = int(list_of_commands[2])
        # 4th command, wait_time for delaying subsequent pulses
        wait_time = int(round(float(list_of_commands[3]) * 10 ** 6, 0))
        values = array.array('i', [0] * datasize)
        times = array.array('i', [0] * datasize)

        # signal that the measurement has started
        led.on()
        no_samples = wait_time * find_freq(adc)
        samples_taken = 0

        # polarise the sample
        pp_coil.on()
        time.sleep_ms(10000)
        pp_coil.off()

        # apply a 90 pulse
        pulse_gen(1)

        # switch to detection mode
        det_switch.on()

        # record data
        start = utime.ticks_us()
        while samples_taken < datasize:  # records data in bursts, applying 180 pulse between each burst
            for i in range(min(no_samples, datasize - samples_taken)):
                values[int(i + samples_taken)] = adc.read_u16() >> 4
                times[int(i + samples_taken)
                      ] = utime.ticks_diff(utime.ticks_us(), start)
                utime.sleep_us(sleep_time)
            samples_taken += no_samples  # for loop counter

```

```
126             det_switch.off()  # detection circuit off so can apply 180 pulse
127             time.sleep_ms(turn_on_time)  # delay pulse to reduce interference
128 
129             pulse_gen(2)  # 180 pulse
130 
131             det_switch.on()  # detection circuit on
132             time.sleep_ms(turn_on_time)  # delay to reduce interference
133 
134         print(json.dumps(times + values))
135         led.off()
136 
137     else:  # error handling, if didn't input correct commands
138         print("Incorrect command string")
139 
```