Portable Magnetic Resonance Imaging Device

By

Yaokun Shi
Zexuan Cheng
Yujiang Han

May 2022

Abstract

This final report is dedicated to the design process and result of the Portable MRI device, which provides an alternative to the traditional MRI in some special conditions while being more convenient and less expensive. The device is divided into four subsystems, and elaborated accordingly. Then the report ends with a conclusion discussing the project's accomplishments and obstacles, along with several possible modifications.

Contents

1. Introduction

1.1 Problem

As image and signal processing becomes more developed and prevalent in the medical industry, advanced scanning devices are needed for the diagnosis of many diseases and such equipment is essential to many healthcare centers. However, a state-of-the-art MRI device that produces detailed images usually requires between 1.5 million to 3 million USD for purchase and installation depending on the strength of the magnetic field [1]. Moreover, a modern MRI device can take up a drastic amount of space and it is an arduous task to transport the device to a different location once it has been installed. It also requires the patient to be stationary inside the bore of the device for 15 to 90 minutes to complete an MRI scan depending on the region getting scanned [2]. In a nutshell, the spread and adoption of this technology on both the global and domestic scales have been dramatically impeded due to the cost, space, and time constraints that come along with the technology. Consequently, it is difficult for smaller clinics or medical-related organizations/groups to obtain an MRI device, and the usage of MRI technology during emergencies or on-site operations is improbable.

Several attempts in the field have been made in order to address this issue, and some prototypical devices that have been published include portable, low-field MRI devices. Nevertheless, these devices fail to produce images with high resolution and quality, and they continue to incorporate a cylindrical bore, resulting in a relatively enclosed geometry in the imaging volume. Furthermore, such prototypical devices are already heavy and had to be transported on wheels, which make them unsuitable for urgent-care scenarios.

1.2 Solution

To reduce the size, cost, and scan time of traditional MRI devices while providing an open geometry in the imaging region to accommodate for different scanning purposes, we propose a portable MRI device that incorporates a non-uniform, non-linear magnetic field and two sets of radiofrequency (RF) coils. The magnetic field provides unique spatial encodings through rotating two individual magnets, and a ferromagnetic core is placed on top of the two magnets to prevent magnetic flux loss. The imaging volume is the box-shaped region underneath the two magnets, and this region can be easily expanded through adding more transmitter and receiver coils. The two sets of RF coils are going to be surface coil arrays that act as transmitters and receivers, respectively, and they will be functioning at precomputed Larmor frequencies given the image voxel that they are responsible for. As figure 1 suggests, the data collected from these surface coil arrays will be analyzed and processed by the control unit, which performs current amplification as well as analog-to-digital conversion so the raw digitized signals can be sent to the image processing unit. The image processing unit will first package the serially transmitted data, run a digital signal processing algorithm to convert the image from time domain to frequency domain, and then use parallel imaging and deep learning techniques to ultimately produce and reconstruct an intelligible image of the scanning volume.

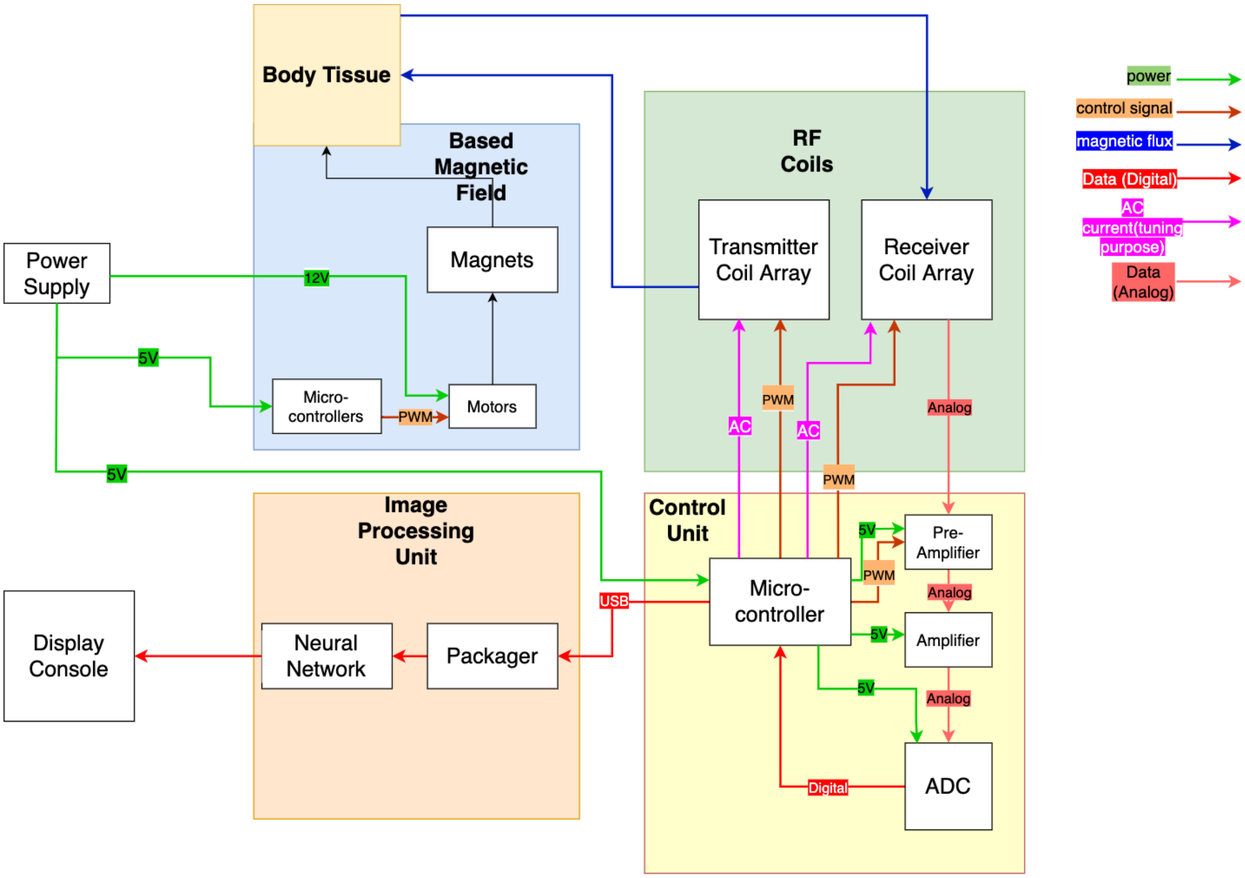


Figure 1. Block diagram of Portable MRI device

1.3 High-Level Requirements

- The final device should be able to generate an intelligible image scan of the target region and achieve an SSIM (Structural Similarity Index Measure) of above 0.5.
- The portable device needs to be relatively constrained in terms of its size and weight. The total size of the scanning device should not exceed 0.3m x 0.3m x 0.3m, and the total weight of the scanning device should not exceed 20kg.
- The power consumption of the entire system (scanning device + computing device) should be limited, and the device should not consume more than 500W.
- The overall scan time for the device should be limited due to the urgent-care nature of the device, and the current goal is a scan time of 5 to 15 minutes.

2. Design

2.1 Base Magnetic Field

2.1.1 Diametrically Magnetized Magnets

One major difference between the base magnetic field on the portable MRI device from traditional MRI devices is that the permanent magnets are being rotated to create a non-linear and non-uniform magnetic field. Using this technique, unique spatial encodings can be generated for all the surface coils in the RF Coil subsystem so that the receiver coils can locate themselves. The reason why a rotating magnetic field is adopted is due to the fact that activation of gradient coils often results in a delayed scanning time, and implementing this extra component in the device would encumber the physical design and further suppress the weight and space left for other components. In the case of the prototypical device, two diametrically magnetized magnets are placed parallel to each other, with 180 degrees of difference in their magnetization direction to ensure the maximum magnetic flux in the imaging volume. Note that the number, size, and weight of the magnets can be optimized to achieve an even stronger magnetic field in the imaging volume to enhance the image quality.

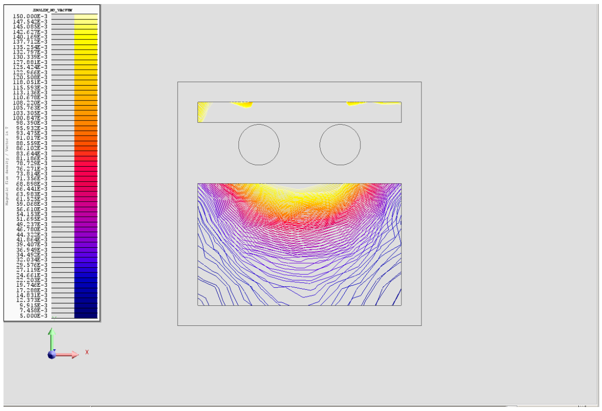


Figure 2. Zero Position XY-Plane

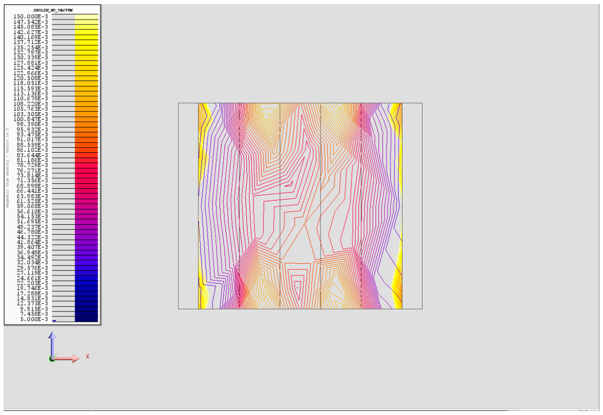


Figure 3. Zero Position XZ-Plane

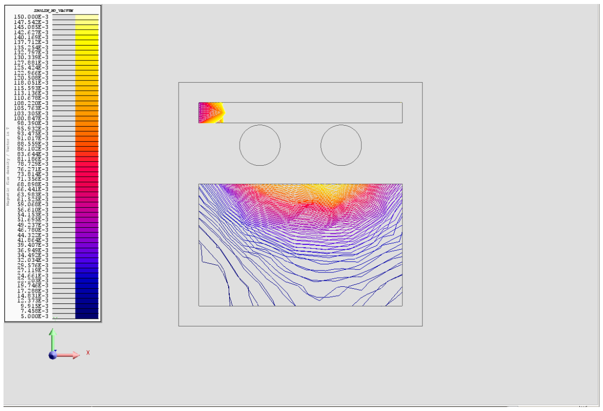


Figure 4. 30 Degree XY-Plane

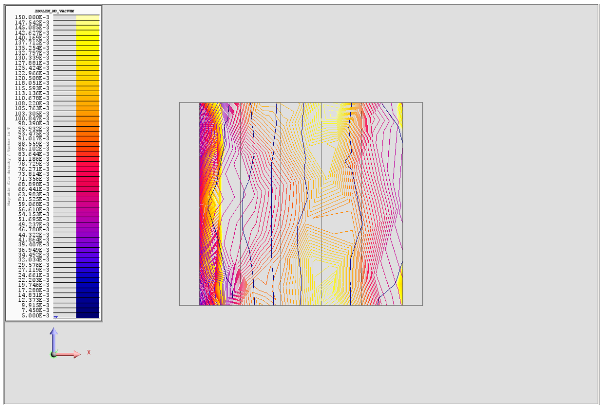


Figure 5. 30 Degree XZ-Plane

As Figure 2 to Figure 5 suggests, the simulation supports the assumption that the magnetic field generated by two individually rotating magnets have the potential to create uniqueness regarding the magnetic field strength at each imaging voxel. For image voxels that are 10mm x 10mm x 10mm, the usage of two magnets is sufficient to generate unique magnetic field strengths for a 30mm x 30mm x 10mm imaging volume.

2.1.2 Ferromagnetic Core

The ferromagnetic core used in this project is a plate made of ferromagnetic material, which is placed on top of the two rotating magnets to redirect the magnetic fluxes into the imaging volume, ultimately increasing the magnetic field strength in the scan region to improve coil performance. A ferrite plate with the dimension of 100mm x 100mm x 12mm is used in the prototype, and there are two major considerations accounting for this choice. First of all, ferrite is relatively cheap to purchase and available in many different shapes and sizes, which contributes to lowering the manufacturing cost of the device and provides flexibility in terms of structural design. Moreover, most ferrite is not electrically conductive, which helps reduce the eddy current loss in the surface coil arrays.

2.1.3 Motors and Drivers

The rotation of the two magnets is done through attaching the shaft of the two gear-motors to the 3D printed encasements for the magnets, and the PWM encoder drives the gear-motors at an appropriate pulse width to control the speed and time of rotation. Another degree of freedom can be provided through adding an extra gear motor on the top of the base structure so that the entire magnetic field component can be rotated to introduce extra magnetic field strength encodings, thus ensuring the uniqueness of field strength in the imaging volume. The angle as well as amount of rotations performed will be determined by the minimal amount of rotations needed to create a unique encoding for each surface receiver coil, as well as the minimal magnetic field strength our coils can operate in. The choice of gear motor is primarily based on the stalling torque on the motor due to the high magnetic pull force induced by the two magnets as well as the ferrite plate on top. Equation (1) through (3) illustrates the approximated torque generated on each shaft:

$$\tau_{total} = F_{magtomag} \times r + F_{magtoplate} \times r \tag{1}$$

$$\tau_{total} = 25N \times 0.01m + 50N \times 0.01m \tag{2}$$

$$\tau_{total} = 0.75N \cdot m \tag{3}$$

The gear motor purchased is capable of generating a stalling torque of 175 oz-in, which is equivalent to 1.236N·m; therefore the motors can effectively rotate the two magnets and maintain their position when the program temporarily halts to excite the body tissue and retrieve information. Another crucial limiting factor is the stalling current generated by the motors since the maximum current that these gear motors can operate in is 11.5A. To address this issue, a power distribution board with 10A fuses in each output channel is used for powering the motor such that the circuit would break in effort to protect the motors.

2.2 Radiofrequency Coils

2.2.1 PCB coil design

The radio frequency coil subsystem consists of two components, transmitter and receiver surface coil arrays. Both arrays will be made up of PCB coils due to the fact that the number of turns and layers of such PCB coils can be arbitrarily decided, consequently minimizing the area of the coils and allowing for increased flexibility in coil designs. Utilizing [3], the coils are generated as copper traces onto the PCB layout. The transmitter coil array functions as a nuclear spin excitation trigger, which sends a signal to the target imaging volume and excites the nuclear spin of hydrogen atoms in that region. Considering the limited area and that the transmitter needs to be placed perpendicular to the base magnetic field, only one row of transmitter coils are implemented, where each coil is responsible for a column of receiver coils. This is conceptually valid since each transmitter generates a uniform magnetic field along the axis, and atoms further from the coils can still be excited at a slightly dampened amplitude. The receiver coil array, on the other hand, detects the magnetic field change in the target region and records the strength as well as time spent for the nuclear spin to return to its original state (under the influence of the base magnetic field). According to [4], an MRI device requires a minimal external magnetic field on the 1μT-scale, and the copper trace in kicad is 0.16mm (the least requirement by PCBway), which is fit for approximately 0.5A current.

$$B = \frac{\mu_0 \cdot I}{2R}, \ \mu_0 = 4\pi \cdot 10^{-7} \tag{4}$$

As equation (4) (The Ampere's Law) indicates, a single loop with r=5mm holding 0.5A is able to generate a 60μT magnetic field, which satisfies the requirement of the device. Shown in Appendix B Figure 19 and Figure 20, the individual transmitter and receiver coils are of the same size where the radius is 5mm and approximately 20 turns using 0.16mm kicad copper trace.

2.2.2 Tuning and Larmor Frequency

In order to excite specific atoms, the radiofrequency (RF) coils targeting such atoms will have to operate at Larmor frequency, which is calculated from the base magnetic field strength as well as the gyromagnetic ratio of the target atoms in equation (5):

$$f = \gamma \cdot B_0 \tag{5}$$

Where f is the Larmor frequency(MHz), γ refers to the gyromagnetic ratio for specific atoms, and B_0 is the base magnetic field strength(T). The base magnetic field in the prototype ranges from 1mT to 200mT, and the gyromagnetic ratio for Hydrogen atoms is 42.58; therefore, the range of the working frequencies for the RF coils should be between 42.58kHz to 8.516MHz. Traditional MRI requires that the coil comprises conductive wire with two tuning capacitors to fine-tune the coil frequency, and a detuning trap (inductor and diode) to actively deactivate/detune the coil [5]. However, the portable MRI device requires rapid and precise control of frequency used for driving the RF coils, meaning that traditional LC circuits would not be a viable option because it requires manual adjustment of capacitor values in order to tune the frequency output. On the other hand, given the case that each coil needs at least two capacitors, the task of impeccably soldering all the components while maintaining a minimum space usage becomes incredibly arduous. Noting that the LC circuit used in the traditional tuning method does nothing more than changing the oscillating frequency of the current in the coil, the portable MRI device incorporates an on-chip function generator to produce the desired frequency to simplify the tuning process and reduce time needed for scanning. Additionally, during the experiment of coupling, it was found out that the transmitter coils need to be powered by a relatively high voltage compared to the receiver's voltage to create visible change in the receivers. The on-chip function generator in the prototype is incapable of producing a controlled, high voltage for the transmitter coils, and thus a function generator device is used as an alternative.

2.3 Control Unit

2.3.1 Microcontroller

The scanning operation requires several microcontrollers with sufficient computing power to realize the code that will be written with Arduino IDE. In the current version of MCU, Arduino Nano dev boards, which contain an ATMEGA32U4 as the main controller chip, are being used due to the global chip shortage. The primary microcontroller needs to inform the motors when to rotate, as well as the secondary microcontrollers when to excite and detect; then it sends the digitized data to the image processing unit through serial communication. On the other hand, the secondary microcontrollers share a similar script shown in Appendix D, except for the address of master. The secondary microcontrollers are responsible for driving the on-chip waveform generators, as well as utilizing their built-in ADCs to read inputs from the receiver coils at a proper time.

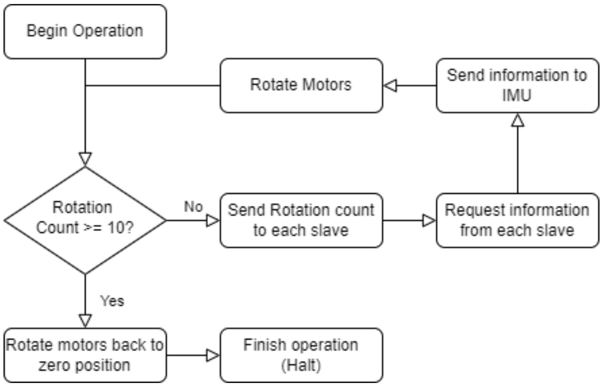

Begin Operation
Rotate Motors
Send information to IMU
Rotation Count >= 10?
No
Send Rotation count to each slave
Request information from each slave
Yes
Rotate motors back to zero position
Finish operation (Halt)

Figure 6. Control Logic Flow Chart for Primary Microcontroller

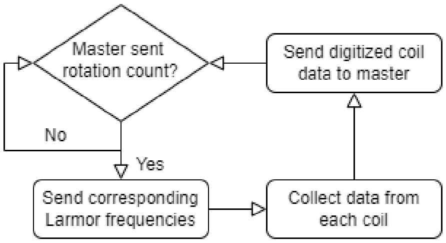

Master sent rotation count?
Send digitized coil data to master
No
Yes
Send corresponding Larmor frequencies
Collect data from each coil

Figure 7. Control Logic Flow Chart for One Secondary Microcontroller

Figure 6 and 7 gives a general idea on how each microcontroller functions as well as their interactions. I2C communication protocols are used between microcontrollers, and serial communication is used for transmitting data from the primary microcontroller to the image processing unit.

2.3.2 On-chip Waveform Generator

The waveform generator generates an analog sine wave, which is used to excite the transmitter and receiver coils. As mentioned in the previous subsystem, the waveform generator needs to be programmable so that the device can conveniently change the output frequency based on the embedded program. AD9837 is the on-chip waveform generator that is implemented in the prototype for several reasons including its easily programmable nature as well as the wide range of frequencies that it is able to generate, which covers the entire span of Larmor frequencies calculated from the magnetic field measured in the device.

2.3.3 Analog-to-Digital Converter

In order to convert the analog feedback signals from the receiver coils and transmit the signal to the image processing unit for further procedures, an ADC will be used in the portable MRI. Since the portable MRI has high requirements for the accuracy of the signal and the operation speed should be relatively fast, ADC should have a decent resolution and high sampling speed. However, due to the unstable performance of coils and lack of usable waveform generators, a complete RF coil system as well as a good feedback signal from the receiver coils is lacking. Therefore, A built- in ADC with 10-bit resolution and slow sampling rate on the microcontroller unit is used for testing purposes.

2.3.4 Preamplifier and Amplifier

Due to the relatively weak field strength and certain requirements of ADC, a preamplifier and an amplifier is required to increase the amplitude of the received signal to an appropriate level before digitizing and minimizing the effect from the offset.

$$V_1 = V_{in} + V_{os1} \tag{6}$$

$$V_2 = V_{os2} - G \cdot V_{os1} \tag{7}$$

$$V_{out} = V_1 + \frac{V_2}{G} = V_{in} + \frac{V_{os2}}{G} \tag{8}$$

Equations (6) through (8) illustrate the minimization of the offset with Vos1 and Vos2 being the offset for the input and output of stage1 and 2 and G being the amplifier gain. Thus, the output from the receiver coil can be amplified to a desired voltage such that the difference in signal received is easily detected between different body tissues.

2.4 Image Processing Unit

2.4.1 Data Packaging and Visualization

The control unit sends data serially to the image processing unit, and the data undergoes the first step of processing, which constructs an array for data from each coil so that it resembles a complete signal for further manipulation. Due to the low transmission rate of Arduino serial communication, the data collected by our prototypical device is truncated as a tremendous amount of data is lost during the procedure. With an appropriate communication protocol and a more powerful microcontroller, it is possible to reduce this loss to the minimum and reconstruct the signal in a more sophisticated manner. Furthermore, numeric clamping is implemented to filter out undesired outliers generated by the built-in ADC from the microcontroller, and the resulting data is visualized through plots.

2.4.2 Fast Fourier Transform

Fast Fourier Transform(FFT) needs to be performed because the packaged data showcases the digitized signal in the time domain. Converting the data into frequency domain will allow the program to identify the energy peak at the correct frequency, therefore diminishing the effect of neighboring responses as well as noise in the environment. As figure x in Appendix C shows, a simple FFT algorithm from the built-in numpy package in python is called to cast the information to the frequency domain.

2.4.3 MoDL Neural Network

The low magnetic field as well as Larmor frequencies inevitably impairs the accuracy and interpretability of data collected, and therefore some form of image reconstruction technique is required to generate an intelligible image output. Some of the reconstruction methods that are frequently used in the field include GRAPPA and SENSE; as [5] suggests, the MoDL algorithm is an alternative method that has proved to surpass most of its counterparts. It excelled in image reconstruction for MRI images, and the reconstructed image is able to achieve a Structural Similarity Index Measure of 0.92. Thus, this particular neural network [6] is adopted in our image processing unit to enhance the final image quality.

3. Verification

3.1 Base magnetic field

3.1.1 Rotation of Motor Shaft in Each Iteration

Despite basic functionalities including driving the motors at 10V and ensuring the current does not exceed 10A being verified, one of the requirements that the device is not able to meet is the 30 degree rotation of the motor shaft in each operation. The motor driver controls two gear motors using the pulse width of a PWM signal, and the smallest unit of delay by the microcontroller is 1ms, which results in a rotational angle that is greater than 1 degree, which means that precise angle control is unachievable.

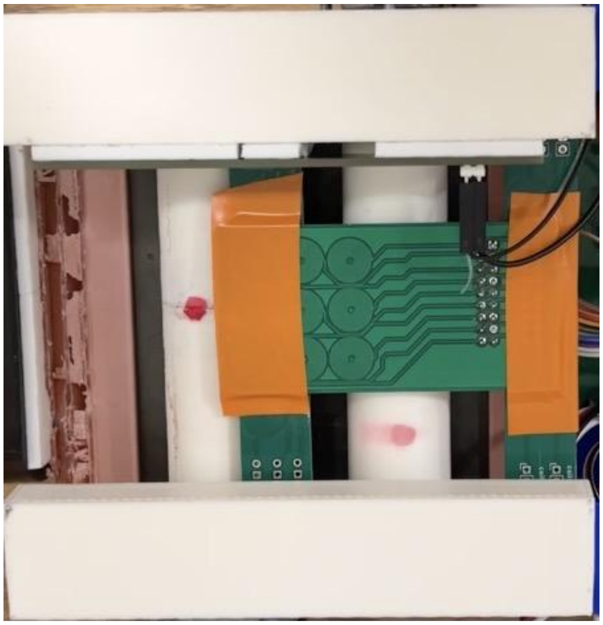


Figure 8. Motor at zero position

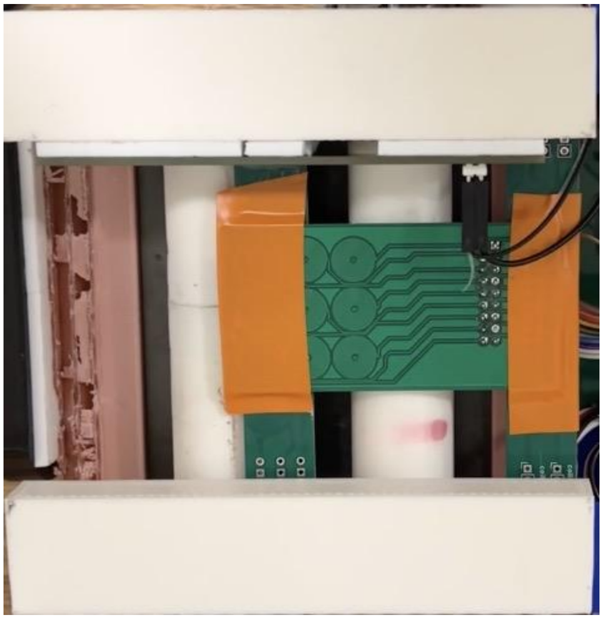


Figure 9. Motor after one rotation

Figure 8 & 9 demonstrates the current result after one rotation, and the angle of rotation is estimated to be 35 ± 5 degrees. However, the issue is accommodated by the solution of measuring the magnetic field at the angle of one rotation since it would still generate a unique magnetic field at this angle given the number of receiver coils does not surpass the limit.

3.1.2 Measurement of Base Magnetic Field

The measurement of the base magnetic field is done through placing the probe of a Gauss/Tesla meter at each imaging voxel, and performing rotation in one axis to acquire the maximum magnetic field value. Table 1 below records the magnetic field values that are measured, and the numbers have been approximated due to the fact that the readings on a Gauss meter are extremely unstable and floating. Several trials at each imaging voxel are taken, and the mean of each measurement is recorded; for symmetrical positions, voxels on both sides are measured and the mean is computed and recorded for both voxels to preserve the symmetry in ideal situations.

Table 1. Measured Magnetic Field in Imaging Region

90mT	160mT	90mT
75mT	100mT	75mT
90mT	160mT	90mT

Zero Position

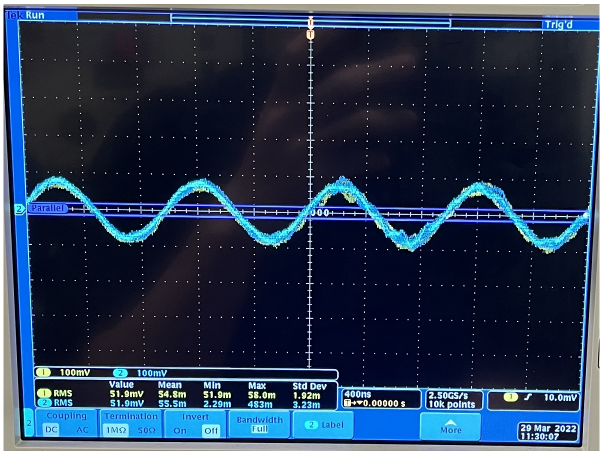
90mT	120mT	90mT
75mT	90mT	75mT
105mT	140mT	105mT

30 Degree Rotation

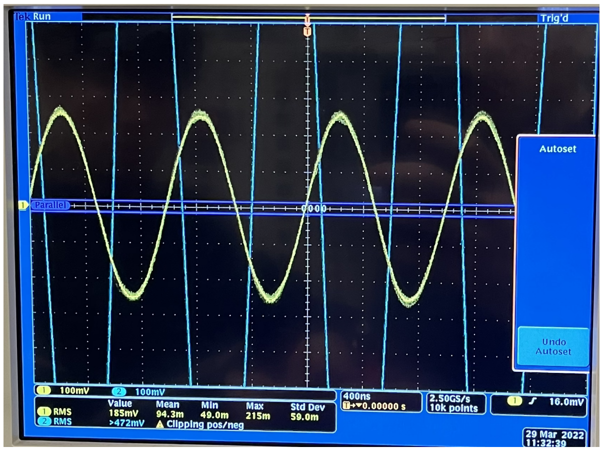
3.2 Radiofrequency coil

3.2.1 Tuning and Coupling

The fundamental requirement of our coil is to be tuned to the Larmor frequency with its position specified, and the coils have to be tested using parallel coupling methods to ensure their basic functionalities. Using a waveform generator IC to power a receiver coil and probing the receiver's output with an oscilloscope, a transmitter coil is powered by a separate waveform generator device at the same frequency and placed directly in front of the receiver coil. Observations are made that the coils' analog signal coherently follows the specifications during the setup stage.



	Value	Mean	Min	Max	Std Dev
1 RMS	51.9mV	54.8m	51.9m	58.0m	1.92m
2 RMS	51.9mV	55.5m	2.29m	483m	3.23m



	Value	Mean	Min	Max	Std Dev
1 RMS	185mV	94.3m	49.0m	215m	59.0m
2 RMS	>472mV	⚠ Clipping pos/neg			

Figure 10. Receiver coil signal without coupling

Figure 11. Receiver coil signal after coupling

As figure 10 shows, a clear sine wave signal output through the receiver coil is displayed on the oscilloscope; after bringing the transmitter and receiver coil together, which both work at the same frequency, a drastic change in the receiver's signal amplitude is seen. Figure 11 demonstrates this change as the amplitude of the blue channel signal increases dramatically (the yellow channel is merely a reference signal without coupling). This result is essential because it allows the implementation of traditional MRI coil's excitation and receiving procedures with an on-chip waveform generator, saving both space and scan time.

3.2.2 Scanning of Subjects

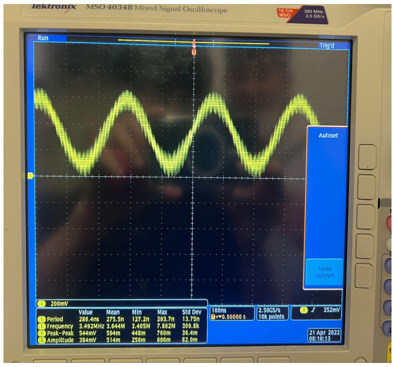


Figure 12. Receiver signal before scanning

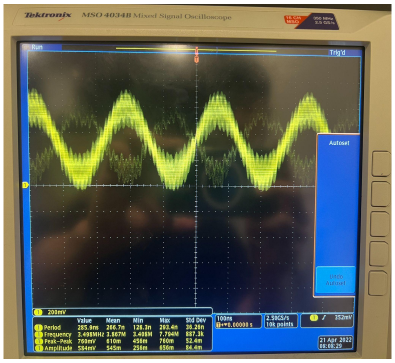


Figure 13. Receiver signal while scanning water

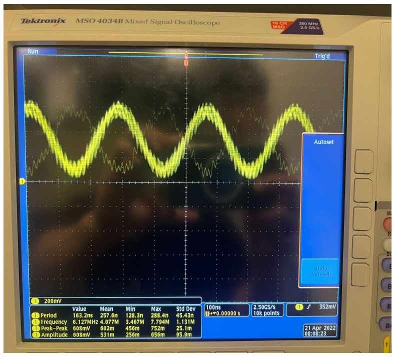


Figure 14. Receiver signal while scanning air

The above figures show the scanning results of an example receiver coil. Serving as a reference, figure 12 is dedicated to display the signal directly from the waveform generator IC. Figure 13 then illustrates the coil catching the power from the hydrogen atoms as a phantom containing water is placed in the scanning region, in which the original sine waveform is distorted by the power from water. Lastly, the water is removed while keeping the transmitter and receiver on, and a barely distorted signal is expected. Just as figure 14 shows, the scanning of air gives out a signal much similar to figure 12.

3.3 Control unit

3.3.1 Motor control

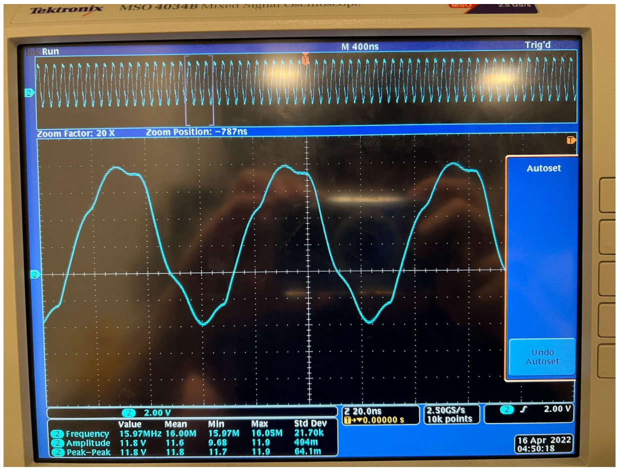
The most important functionality for the control unit would be generating the control signal for the motor, since the motor needs a PWM input within 16Hz to 200Hz. The RPM for the motor can be varied by changing the duty cycle of the PWM input to the motor driver, and the output voltage will be varied based on the duty cycle of the PWM input. Therefore, The MCU outputs the PWM signal from the PWM ports that have a duty cycle and frequency based on the embedded program to control the motor. As we demonstrated in the previous presentation, the motor can receive the control signal and have correct rotation as expected.

3.3.2 Communication between primary and secondary MCU

Another important functionality would be the communication between the microcontroller units since the MCU contains a secondary microcontroller unit that not only controls the on-chip function generators, but also reads the feedback from the receiver coils and sends it back to the primary control unit. Therefore, the communication between primary and secondary microcontrollers is needed so that both of them would be able to read and write the serial data from each other. During the subsystem test, we write some testing serial data to the secondary microcontroller and successfully read it from the target microcontroller. Also after the integration, the secondary microcontroller can be programmed and read the feedback from the primary microcontroller. The result will be shown in the data acquisition part.

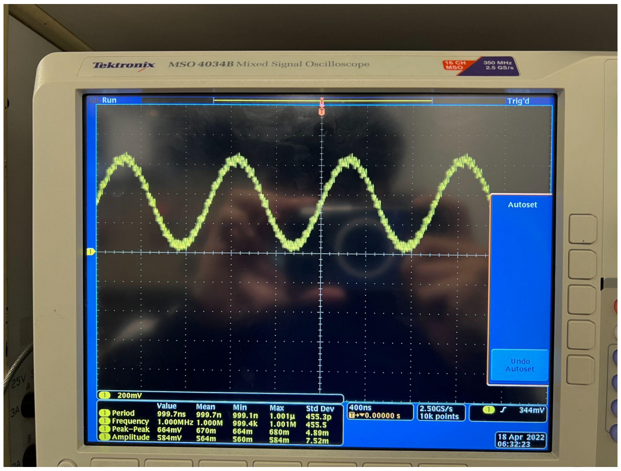
3.3.3 On-chip waveform generator

Last but not least, the microcontroller should be able to correctly program the on-chip function generator to generate a sine wave that has frequency that matches our calculated larmor frequencies. In order to achieve this, a relatively high frequency clock signal from the microcontroller is needed as the reference clock for the on-chip function generator. Originally, Arduino nano can only output 490 Hz to 976 Hz frequency output, so in order to output high frequency, the embedded configuration file for the microcontroller needs to be modified and burnloaded to output 16MHz signal from the arduino. The integration of the microcontroller and the function generator is successful, and the function generator can output the coil excitation signal based on our embedded program. Below is the example output of the Master microcontroller clock and waveform generator.



	Value	Mean	Min	Max	Std Dev
2 Frequency	15.97MHz	16.00M	15.97M	16.05M	21.70k
2 Amplitude	11.8 V	11.6	9.68	11.9	494m
2 Peak−Peak	11.8 V	11.8	11.7	11.9	64.1m

Figure 15. 16MHz MCLK output



	Value	Mean	Min	Max	Std Dev
1 Period	999.7ns	999.7n	999.1n	1.001µ	455.3p
1 Frequency	1.000MHz	1.000M	999.4k	1.001M	455.5
1 Peak−Peak	664mV	670m	664m	680m	4.89m
1 Amplitude	584mV	564m	560m	584m	7.52m

Figure 16. Waveform generator output

3.4 Image processing unit

3.4.1 Visualization of Data

Three separate trials are performed to validate the output as well as the functionality of data packaging and visualization, and each trial consists of both air and water as the scanning subject. In the 30 seconds trial, the transmitter coil is turned on for the first 15 seconds and turned off for the later 15 seconds to demonstrate the difference between scanning water and scanning air. Figure 17 & 18 showcases the results collected by the built-in ADC on the microcontroller, and though hardly noticeable, the peaks in signals while detecting air is less than the peaks in signals while detecting water. Although the change is minimal, it proves that the prototype is able to visualize the difference between water and air given a better ADC and the amplifying circuits are implemented.

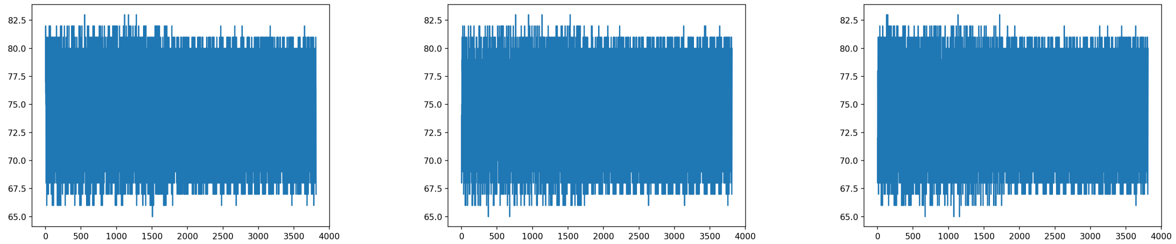


Figure 17. Data acquired from coil scanning air

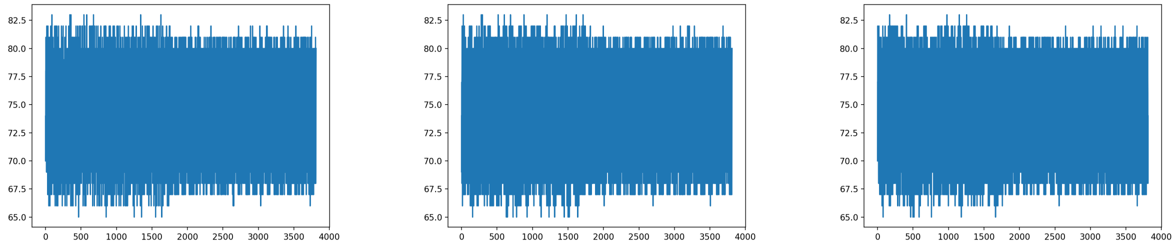


Figure 18. Data acquired from coil scanning water

3.4.2 FFT and Deep Learning

Due to the poor signal sampling rate and minimal change in excitation amplitude received by the receiver coils, it is decided that FFT and DL algorithms are not going to generate interpretable results, and thus these two algorithms are not included in this prototypical device.

4. Cost

4.1. Labor

The average salary received by undergraduate course assistants in the UIUC ECE department is 11$/hour after tax [8], and our estimation of weekly working hours is 10 hours. During the semester, around 11 weeks were dedicated to the designing and construction of our project. Thus, equation(6) demonstrates the approximation of the total labor cost:

$$11\ \$/hr\ \times\ 10\ hrs/week\ \times\ 11\ weeks\ \times\ 2.5\ \times\ 3\ =\ 9075\$ \qquad (6)$$

4.2. Parts

Table 2. Costs for all Purchased Parts

Part Name	Part Number	Qty	Unit Price ($)	Total Price ($)
NeveRest Orbit 20 Gearmotor	AndyMark: NeveRest Orbit 20 Gearmotor	3	35	105
Power Distribution Board	AndyMark: Powerpole Distribution Board	1	44	44
Anderson Power Cord	AndyMark: 12 AWG Bonded Cable PP45 Powerpole to Bare Wire	4	6	24
NdFB Diametrical Magnets	K&J Magnetics: DCCDIA	7	10.13	70.91
Ferrite Plate	Digi-Key: 399-FPL100/100/12-BH1T-ND	1	61.88	61.88
ATMEGA32U4 Dev Board	Digi-Key: 1050-1001-ND	30	5.27	158.1
AD9203 ADC	Digi-Key: AD9203ARUZRL7TR-ND	10	6.07	60.7
AD9837 Waveform generator	Digi-Key: 505-AD9837BCPZ-RL7TR-ND	10	6.01	60.1
1 to 16 Analog Multiplexer	Digi-Key: MC74LVXT4051DR2GOSTR-ND	25	0.87	21.75
Jumper Wires	ECE-Shop	60	0.5	30
Breadboard	ECE-shop	2	10	20
3D-Printer Filament	Gizmo Dorks: 3D PRINTER PLA FILAMENT	1	23.95	23.95
M3x10mm screws x 10	Amazon	2	10	20
Total				700.39

5. Conclusion

5.1 Accomplishments

Within the four high level requirements listed in the design document, three of them are achieved. The first high level requirement needs a SSIM (Structural Similarity Index) above 0.5. We are unable to fulfill this requirement since the current ADC's sampling rate doesn't match the frequency of the fetched data, which results in a poor image quality. The second requirement is fulfilled that the device's dimension is 30cm x 30cm x 15cm, and the device weighs approximately 5kg. The third requirement is also achieved. Most power is consumed by the motors which require 10 volts, and it draws current less than 1 Amp throughout the operation. Other power is consumed by the control unit which has a power consumption of 5V × 19mA [9]. As a result, the total power used won't exceed 11W. The last requirement limits the time used by a single scanning to 5 minutes. Based on our experiment, distinguishing water from air takes four scanning cycles, and it takes ten seconds to collect data each cycle to generate a valid output. And therefore, it is reasonable to assume the device is able to scan more complex targets within 5 minutes. As the verification section and figure 12-14 suggest, our device is able to distinguish the hydrogen atoms by exciting and receiving power from the atoms within the image volume. While the primary function is verified, other subsystems are closely following the specifications of the design document. The base magnetic field is able to create a strong field up to 200mT. As for motors, although their rotation angle cannot be precisely controlled, approximate magnets rotations are still enough to create different spatial encoding for our receiver coils. The control unit (which is referred to as CU below) is successfully designed and is able to carry out most of the tasks listed in the design document. By receiving the feedback from the motors' encoder, the CU can roughly control the motors' rotation angle. The CU is also able to switch on/off and tune the coils by communicating with the waveform generator IC. At last, after the analog signal is fetched by the receiver, the CU can digitize the signal and output it to the image processing unit.

5.2 Challenges

Low field portable MRI is a very challenging and research based project, and therefore, there are lots of challenges and difficulties for this project in hardware and software design and testing. Soldering could definitely be one of the most challenging steps. The on-chip function generator mentioned in the design introduction uses the QFN package, which is extremely tiny and hard to solder on the PCB. Also this chip is sensitive and the performance could be easily affected by the bad soldering. During the assembling and testing stage, we tried at least 20 times for soldering the on-chip function generator, but only the first one worked, which causes that only one of the coil in the coil arrays could be used for testing and the whole coil systems cannot fully operate since each coil should receive a excitation signal when it is on. Unstable performance of the PCB printed coils could also be one of the factors that could make the project challenging and difficult. Coils could be one of the most crucial parts for this project in the hardware design since it is the part that corporates with the base magnetic fields to scan and react with different materials. Therefore, in order to have a clean and reasonable feedback, the quality of coils could be important. Unfortunately, due to the manufacturing and designing defects, during the testing stage, the coils' performance were unstable, and the transmitter and receiver coils pairing were unstable because of that. Moreover, the result for scanning water and air varied differently in every testing due to the unstable performance of coils.

5.3 Ethics and safety

As our project is being progressively carried out, we should always prioritize ethics and safety. The [10] highlights the safety, health, and welfare of the public, which requires us to comply with the public rules and development practices, to protect the privacy of others, and to actively check for factors that might endanger the public or the environment. This piece of code applies to our projects which involves powerful magnets that could cause damage to public properties and even individual personnel. The magnets are stored individually in appropriate locations and embedded in the mechanical structure with extra caution. Once the magnets are enclosed in their individual chambers and kept stationary by the motor shaft, the physical structure should prevent further damage. Since the magnets that we are handling can create a surface magnetic field as strong as 200mT, it is also crucial to recognize the potential damage that such a powerful magnetic field can cause to the human body. In order to prevent this, the researchers who are working on the testing of the magnetic field as well as other operations that involve close exposure to the magnets should limit the time they spend within proximity to the magnets to below 2 minutes each time they perform such operations. Otherwise, as our project is heavily involved with electronic parts, there could be inevitable problems such as fire, electric, and chemical hazards. We will closely follow the research lab safety rules, for instance [11], "We will never work alone in the lab, will not bring food into the lab, will report any broken equipment, will clear off the working space so it is free of hazards, and will not use two hands on a circuit when powered." Additionally, our project explores MRI on a handheld scale, which is relatively novel and challenging. As a result, we should be rather delighted to take advice and criticism from other professionals, and make realistic assumptions based on researched data, obeying [12]. During construction and upon completion of our MRI device, we should keep in mind that it's specifically designed for medical usage and should only be operated by authorized personnel. Misconduct of the device such as intentionally keeping the strong magnetic fields within proximity to other people's electronic devices should be strictly prohibited. It is also crucial to acknowledge that the quality of the final output of our image will be greatly compromised due to the novel techniques that we are adopting as well as other constraints; therefore, the results should not be interpreted the same way as other MRI images and should only serve an educational purpose.

5.4 Future work

Due to the limited time and funding, we do not have a chance to finish all the functionalities and build a relatively complete prototype. There are lots of improvements that can be made in the future developing process. First of all, the testing of the magnetic field strength can be improved. Currently we are using a probe gaussmeter for the magnetic field testing, however, the accuracy cannot be guaranteed since the reading for the gauss meter is varying in a large range during testing. Furthermore, the control for the motor rotation could be improved and achieve more precise rotation. By implementing the sensor(encoder), we can set up a feedback loop control for the motor to perform a more precise rotation. Moreover, the quality of coils and ADC could be improved in the future since in the current version, the performance of the coil arrays and build in ADC was poor. Coil arrays were unstable and ADC does not have enough resolution and sampling speed. Last but not least, completing the Fast Fourier Transform algorithm as well as the Deep Learning algorithms would be essential for the device to generate a final image output and ultimately achieve the most crucial goal of the device.

References

[1] M. Sarracanie, C. D. LaPierre, N. Salameh, D. E. Waddington, T. Witzel, and M. S. Rosen, “Low-cost high-performance MRI,” Scientific Reports, vol. 5, no. 1, 2015.

[2] D. Yetman, “How long does an MRI take: Lumbar, heart, Pelvic, and more,” *Healthline*, 15-May-2021. [Online]. Available: https://www.healthline.com/health/how-long-does-an-mri-take. [Accessed: 03-May-2022].

[3] JoanTheSpark, (2015) KiCAD_CopperSpiral [Source code]. https://gist.github.com/JoanTheSpark/e3fab5a8af44f7f8779c#file-kicad_copperspiral_v2-py

[4] C. M. Collins and Z. Wang, “Calculation of radiofrequency electromagnetic fields and their effects in MRI of human subjects,” Magnetic Resonance in Medicine, vol. 65, no. 5, pp. 1470–1482, 2011.

[5] B. Gruber, M. Froeling, T. Leiner, and D. W. J. Klomp, “RF coils: A practical guide for nonphysicists,” Journal of Magnetic Resonance Imaging, vol. 48, no. 3, pp. 590–604, 2018.

[6] F. Knoll, K. Hammernik, C. Zhang, S. Moeller, T. Pock, D. K. Sodickson, and M. Akcakaya, “Deep-learning methods for Parallel Magnetic Resonance Imaging Reconstruction: A survey of the current approaches, trends, and issues,” *IEEE Signal Processing Magazine*, vol. 37, no. 1, pp. 128–140, Jan. 2020.

[7] Kaggarwal, (2020) modl [Source code]. https://github.com/hkaggarwal/modl.

[8] “University of Illinois at Urbana-Champaign Course ...,” glassdoor. [Online]. Available:https://www.glassdoor.com/Hourly-Pay/University-of-Illinois-at-Urbana-Champaign-Course-Assistant-Hourly-Pay-E142738_D_KO43,59.htm#:~:text=The%20typical%20University%20of%20Illinois,from%20%2411%20%2D%20%2417%20per%20hour. [Accessed: 25-Feb-2022].

[9]“Arduino Nano,” *Arduino Online Shop*. https://store-usa.arduino.cc/products/arduino-nano/

[10] “IEEE code of Ethics, I.1” IEEE, 2022. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 11-Feb-2022].

[11] E. I. T. S. Services, “Lab,” Lab :: ECE 445 - Senior Design Laboratory, 2022. [Online]. Available: https://courses.engr.illinois.edu/ece445/lab/index.asp. [Accessed: 11-Feb-2022].

[12]“IEEE code of Ethics, I.5” IEEE, 2022. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 11-Feb-2022].

Appendix A. Requirement and Verification Tables

Table 3. R&V Table for Base Magnetic Field

Requirement	Verification	Verified or Not
1. Power Distribution Board distributes 12V +/- 5% to each of the outlets with a 12V DC input from the function generator.	1A. Measure the output voltage from each outlet using an oscilloscope, ensuring that the output voltage stays within 5% of 12V.	Yes
2. 10A fuses on the Power Distribution Board are functional and will break the circuit if the current were to exceed 10A.	2A. Plug 10A fuses into the Power Distribution Board. 2B. Power the gear motor with a 12V power supply. 2C. Stall the motor using a torque test stand and check if the fuses break the circuit.	Yes
3. Magnetic field strength within the effective imaging volume should range from 1mT to 200mT.	3A. Rotate magnets to zero position (magnetization direction is 90 degrees and 270 degrees, respectively) 4B. Place the Gauss meter at the point of interest in the imaging region. 3C. Place the probe of the Gauss meter in parallel to the X-axis, rotate the probe by 360 degrees on the XY plane, and find the maximum reading. 3D. Rotate each magnet by 30 degrees, repeat step 4B-4C.	Yes
4. Motor can rotate by 30 degrees at each scanning iteration.	4A. Press the RESET button to return the motors to their zero position. 4B. Press the TEST button(debug) to perform one iteration on the motor. 4C. Measure the angle of rotation to ensure that motors rotate by the correct amount at each iteration.	Yes/No, since the angle can not be precisely controlled
5. The time needed for each rotation iteration (for each imaging step) should not exceed 1 minute	5A. Do step 5A-5B, and measure the time needed for one rotation using a stopwatch.	Yes
6. Time for motor stalling does not exceed 2 minutes and 54 seconds.	7A. When a motor begins stalling, use a stopwatch to record the time that it has been stalling and cut the power source immediately when the stall time	Yes

	exceeds 2 minutes and 45 seconds.	
7. The operating temperature does not exceed 80℃.	8A. During the time of operation, use an IR thermometer to make sure that the temperature of the motor as well as the microchip is below 80℃	Yes

Table 4. R&V Table for Radiofrequency Coils

Requirement	Verification	Verified or Not
1. Transmitter and receiver coils can be tuned by the on-chip function generator to the desired Larmor frequency.	1A. Use an on-chip function generator to generate the AC current oscillating at the Larmor frequency. 1B. Connect to the coil array and use an oscilloscope to measure the AC current through the coils, ensure that it’s corresponding with the Larmor frequency	Yes
2. Receiver is able to detect the magnetic field change.	2A. After the receiver coil is tuned and turned on, use an oscilloscope to measure the current through the coils. A significant increase of current amplitude indicates the magnetic field change of the target region. 2B. Observe that the current amplitude returns to the initial value, which indicates that the receiver is able to capture the nuclear’s returning to its original state.	Yes

Table 5. R&V Table for Control Unit

Requirement	Verification	Verified or Not
1. The microcontroller can generate correct control signals based on the embedded program.	1A. Motors are operated correctly based on the program. 1B. We can verify and monitor the signal output by using the oscilloscope for the output pins of the microcontroller, which should give us the output signal graph that aligns with the program command.	Yes
2. Preamplifiers should be able to minimize the offset and Noise caused during the transmission.	2A. Based on how much offset and Noise we will get, the output should be roughly smooth and clean. Detailed tolerance analysis will be calculated after we have the sample signal. Again, the signal can be verified by using an oscilloscope to get the waveform and checking the offset and Noise by comparing it with the ideal signal.	No
3. Amplifier should be able to detect the phase difference between the reference signal and input signal and the amplitude difference.	3A. Comparator should be able to compare the amplitude of the input and reference signal and output logic high if the input is higher. This can also be verified by using an oscilloscope to plot the waveform and compare them with the input and a reference relationship. 3B. The phase detector should be able to detect the phase difference and generate output signals that can visualize the delay between reference and input signals by using an oscilloscope.	No
4. ADC needs to have enough resolution (>= 12 bits) to be able to convert massive analog signals, and digital signals should be able to match the analog pattern.	4/5. The digital signal output pattern should match the analog input signal pattern, and we can compare them by plotting the output digital signal using python and reading the input from the oscilloscope.	Yes/No, built-in ADCs were used
5. ADC needs to have reasonable quantization error(<= +/- 0.5 bit) which can be calculated using Matlab after we have the sample signal.		

Table 6. R&V Table for Image Processing Unit

Requirement	Verification	Verified or Not
1. The Packager correctly receives digital signals from the microcontroller in a serial fashion.	1A. Program the controller to send arbitrary variables as output signals. 1B. Check the packager to make sure the constructed image in k-space corresponds to the arbitrary variables.	Yes
2. Data can be visualized in the time domain	2A. Collect data from each coil individually, and plot the digital signal	Yes
3. MoDL Neural Network is trained for the body part that is being scanned with a training accuracy of above 0.9	3A. Clone the MoDL repository from GitHub, and download hand MRI data from the USC hand MRI project. 3B. Run trn.py file with hand MRI data. 3C. Check to make sure that the training accuracy is above 0.9.	No
4. MoDL Neural Network can achieve a testing accuracy of above 0.5	4A. Generate a set of images of hand scan using our device in the k-space. 4B. Run test.py file with the generated images to get an output of the reconstructed image. 4C. Check to make sure that the testing accuracy is above 0.5.	No

Appendix B. PCB layouts and schematics

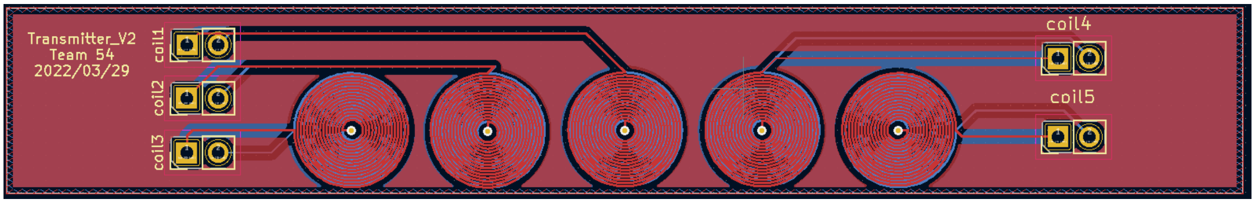


Figure 19. Transmitter coils

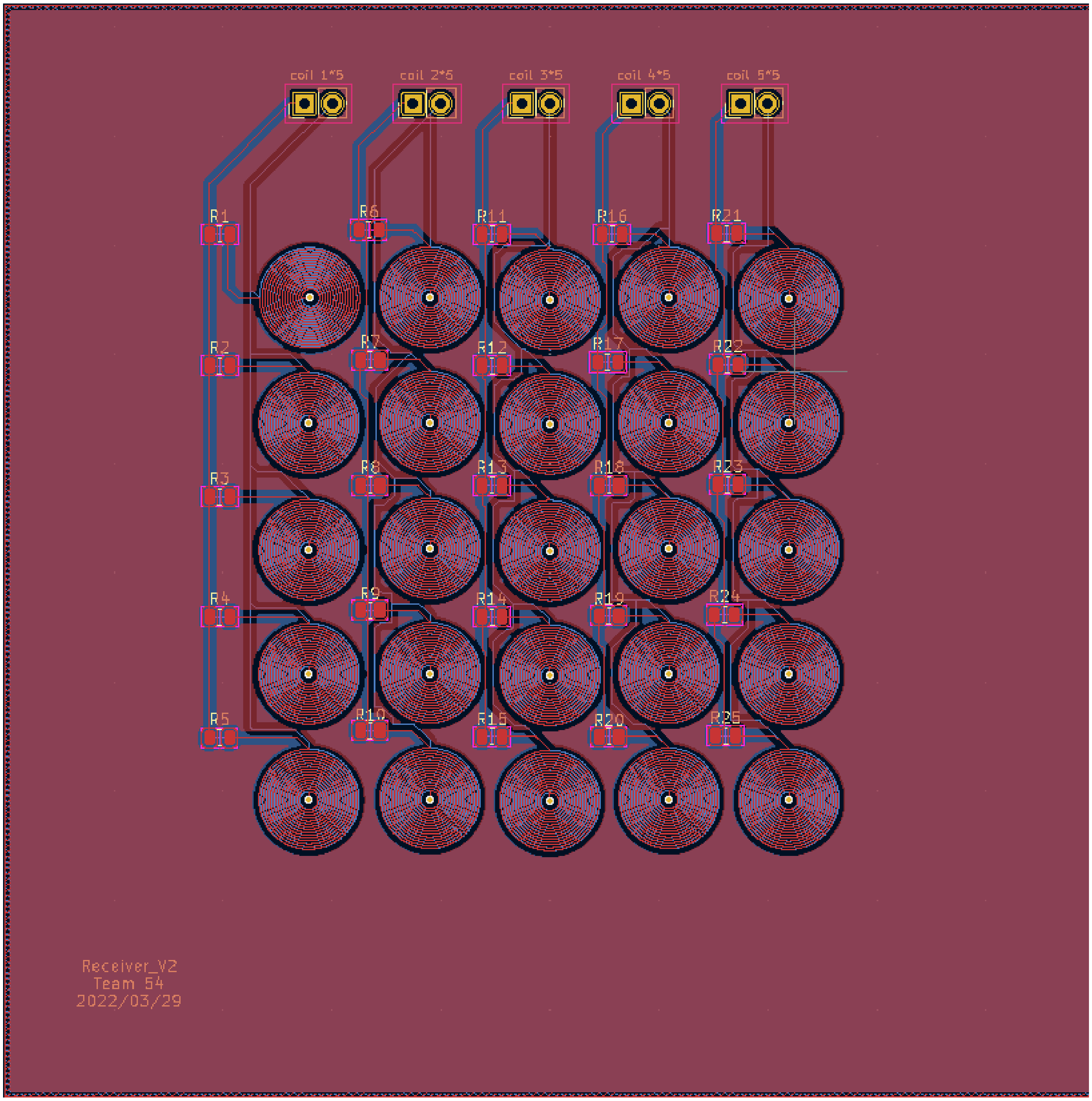


Figure 20. Receiver coils

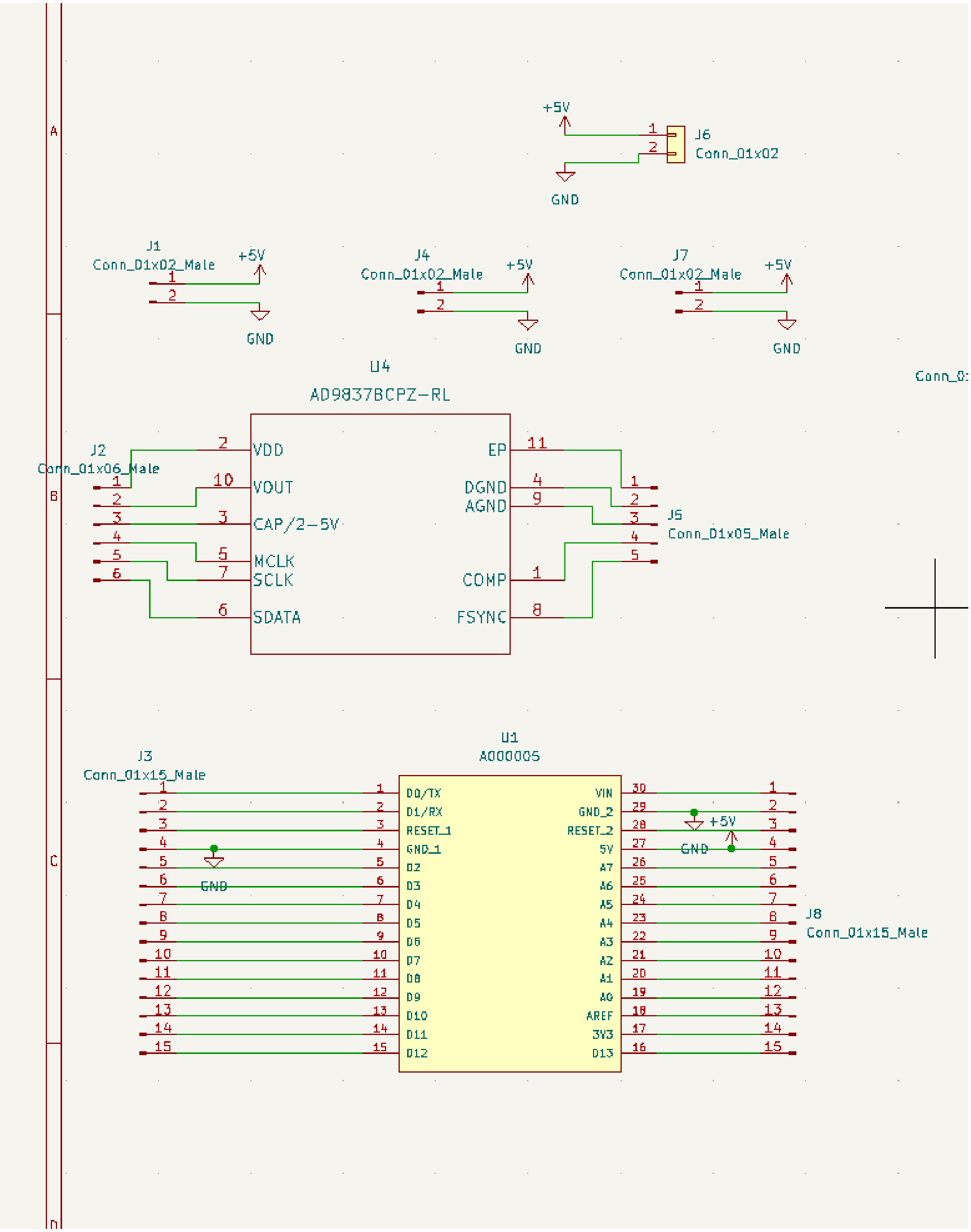


Figure 21. Schematic of Control Unit PCB

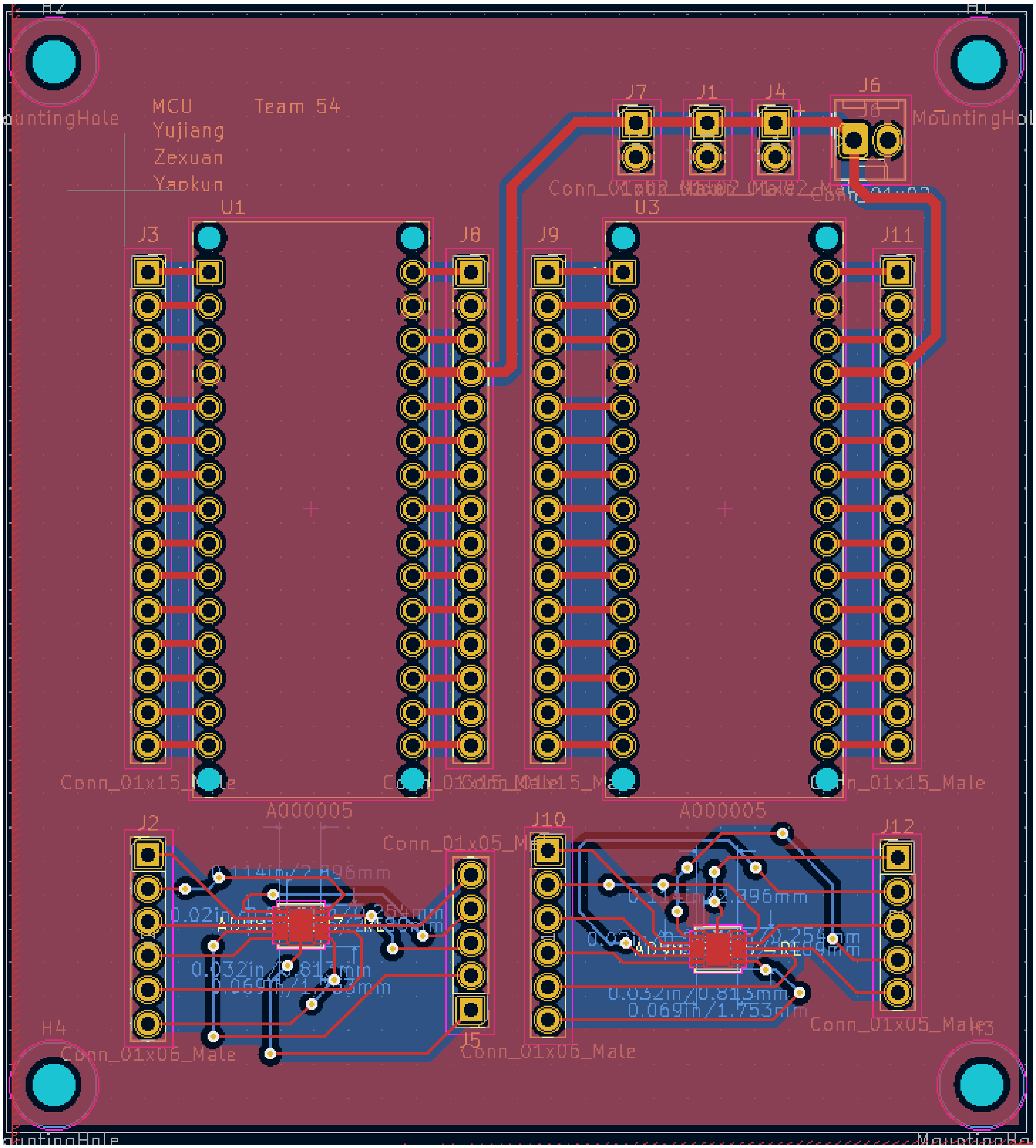


Figure 22. Layout of Control Unit PCB

Appendix C. Primary Control Unit Code

```
#include <Wire.h>
#include <AD9837.h>
#include <time.h>

// Addresses for the five slave MCUs
#define addr_slave1 1
#define addr_slave2 2
#define addr_slave3 3
#define addr_slave4 4
#define addr_slave5 5

// Pins for motor I/O
#define motor1_PWM 5
#define motor2_PWM 6

// Current number of rotations
int rotation_count = 0;
int slave_coil_count = 0;
int begin_flag = 0;

// Final image
int image[5][5] = {0};

void setup() {
  pinMode(13, OUTPUT);
  pinMode(motor1_PWM, OUTPUT);
  pinMode(motor2_PWM, OUTPUT);
//  pinMode(motor1_ENC, INPUT);
//  pinMode(motor2_ENC, INPUT);
  pinMode(A0, INPUT);
  pinMode(A1, INPUT);
  Wire.begin();
  Serial.begin(115200);
}

void loop() {
  /* Begin condition */
  if(!begin_flag) while(!Serial.available());
  begin_flag = 1;
```

```
/* Halt condition */
if(rotation_count >= 2) {
  while(true){digitalWrite(13,HIGH);}
}

/* Begin all operations in 5 slaves */
Wire.beginTransmission(addr_slave1);
Wire.write(rotation_count);
Wire.endTransmission();
Wire.beginTransmission(addr_slave2);
Wire.write(rotation_count);
Wire.endTransmission();
Serial.println(2);
Wire.beginTransmission(addr_slave3);
Wire.write(rotation_count);
Wire.endTransmission();
Wire.beginTransmission(addr_slave4);
Wire.write(rotation_count);
Wire.endTransmission();
Wire.beginTransmission(addr_slave5);
Wire.write(rotation_count);
Wire.endTransmission();

/* Obtain data from all 5 slaves */
// Data from slave #1
Wire.requestFrom(addr_slave3, 5, true);
slave_coil_count = 0;
while (Wire.available()) {
  int c = Wire.read();
  image[0][slave_coil_count] = c;
  slave_coil_count++;
}
// Data from slave #2
Wire.requestFrom(addr_slave3, 5, true);
slave_coil_count = 0;
while (Wire.available()) {
  int c = Wire.read();
  image[2][slave_coil_count] = c;
  slave_coil_count++;
}
```

```
// Data from slave #3
Wire.requestFrom(addr_slave3, 5, true);
slave_coil_count = 0;
while (Wire.available()) {
  int c = Wire.read();
  image[2][slave_coil_count] = c;
  slave_coil_count++;
}
// Data from slave #4
Wire.requestFrom(addr_slave4, 5, true);
slave_coil_count = 0;
while (Wire.available()) {
  int c = Wire.read();
  image[3][slave_coil_count] = c;
  slave_coil_count++;
}
// Data from slave #5
Wire.requestFrom(addr_slave5, 5, true);
slave_coil_count = 0;
while (Wire.available()) {
  int c = Wire.read();
  image[4][slave_coil_count] = c;
  slave_coil_count++;
}

/* Send data to computer through serial line */
for(int i=0; i<5; i++){
  for(int j=0; j<5; j++){
    Serial.print(image[i][j]);
  }
}

/* Rotate motors by 30 degrees */
// change clock frequency to 61.04Hz for motors
TCCR0B = TCCR0B & B11111000 | B00000101;
if(rotation_count == 0){
  // Rotate both motors for 30 degrees
  analogWrite(motor1_PWM, 19);
  delay(7);
  analogWrite(motor1_PWM, 23);
  delay(50);
  analogWrite(motor2_PWM, 26);
  delay(7);
  analogWrite(motor2_PWM, 23);
 }
 else if(rotation_count == 1){
  // Rotate both motors for -30 degrees
  analogWrite(motor1_PWM, 26);
  delay(7);
  analogWrite(motor1_PWM, 23);
  delay(50);
  analogWrite(motor2_PWM, 19);
  delay(7);
  analogWrite(motor2_PWM, 23);
 }
 // Change clock frequency back to default
 TCCR0B = TCCR0B & B11111000 | B00000011;

 /* Update rotation_count */
 rotation_count += 1;
}
```

Appendix D. Secondary Control Unit Code

```
#include <Wire.h>
#include <AD9837.h>

// Address for master controller
#define addr_master 1

// Variables
int rotation_count;
bool coil_on = false;
long int frequency_map[5] = {0};
long int frequency_word[5] = {0};
int result = 0;
AD9837 waveform_generator;

// Pins for MUX
#define MUX_BIT0 2
#define MUX_BIT1 4
#define MUX_BIT2 8
#define MUX_ENABLE 9
#define MUX_OUTPUT A0

void setup() {
  Wire.begin(addr_master);
  Wire.onReceive(receiveEvent);
  Wire.onRequest(requestEvent);
  Serial.begin(115200);
  pinMode(MUX_BIT0, OUTPUT);
  pinMode(MUX_BIT1, OUTPUT);
  pinMode(MUX_BIT2, OUTPUT);
  pinMode(MUX_ENABLE, OUTPUT);
  pinMode(MUX_OUTPUT, INPUT);
  waveform_generator.begin();
}

/*
 * When master MCU sends information, either turn coil on or record current rotation count
 */
void receiveEvent(int dummy){
  char c = Wire.read();
  rotation_count = c;
  /* Arbitrarily set all frequencies */
  switch (rotation_count) {
    case 0:
      waveform_generator.sineWave(waveform_generator.calcDDSTuningWord(3832200);
      break;
    case 1:
      waveform_generator.sineWave(waveform_generator.calcDDSTuningWord(4470900);
      break;
    default:
      waveform_generator.sineWave(waveform_generator.calcDDSTuningWord(3832200); // debug
      break;
  }
}

/*
 * When master MCU request information, send the receiver coil results
 */
void requestEvent(){
  Wire.write(result);
}

/*
 * Send corresponding control signals to the MUX
 */
void selectMUX(int num){
  switch (num) {
    case 0:
      digitalWrite(MUX_BIT0, LOW);
      digitalWrite(MUX_BIT1, LOW);
      digitalWrite(MUX_BIT2, LOW);
      digitalWrite(MUX_ENABLE, LOW);
    case 1:
      digitalWrite(MUX_BIT0, HIGH);
      digitalWrite(MUX_BIT1, LOW);
      digitalWrite(MUX_BIT2, LOW);
      digitalWrite(MUX_ENABLE, LOW);
    case 2:
      digitalWrite(MUX_BIT0, LOW);
      digitalWrite(MUX_BIT1, HIGH);
      digitalWrite(MUX_BIT2, LOW);
```

```
   digitalWrite(MUX_ENABLE, LOW);
  case 3:
   digitalWrite(MUX_BIT0, HIGH);
   digitalWrite(MUX_BIT1, HIGH);
   digitalWrite(MUX_BIT2, LOW);
   digitalWrite(MUX_ENABLE, LOW);
  case 4:
   digitalWrite(MUX_BIT0, LOW);
   digitalWrite(MUX_BIT1, LOW);
   digitalWrite(MUX_BIT2, HIGH);
   digitalWrite(MUX_ENABLE, LOW);
  case 5:
   digitalWrite(MUX_BIT0, HIGH);
   digitalWrite(MUX_BIT1, LOW);
   digitalWrite(MUX_BIT2, HIGH);
   digitalWrite(MUX_ENABLE, LOW);
  case 6:
   digitalWrite(MUX_BIT0, LOW);
   digitalWrite(MUX_BIT1, HIGH);
   digitalWrite(MUX_BIT2, HIGH);
   digitalWrite(MUX_ENABLE, LOW);
  case 7:
   digitalWrite(MUX_BIT0, HIGH);
   digitalWrite(MUX_BIT1, HIGH);
   digitalWrite(MUX_BIT2, HIGH);
   digitalWrite(MUX_ENABLE, LOW);
  default:
   digitalWrite(MUX_ENABLE, HIGH);
 }
}

void loop() {

 /* Activate Tx and Rx in order */
 // Tx on, Rx1 off
 selectMUX(2);
 delay(1000);
 // Tx off, Rx1 on
 selectMUX(3);
 result[0] = analogRead(MUX_OUTPUT);
```

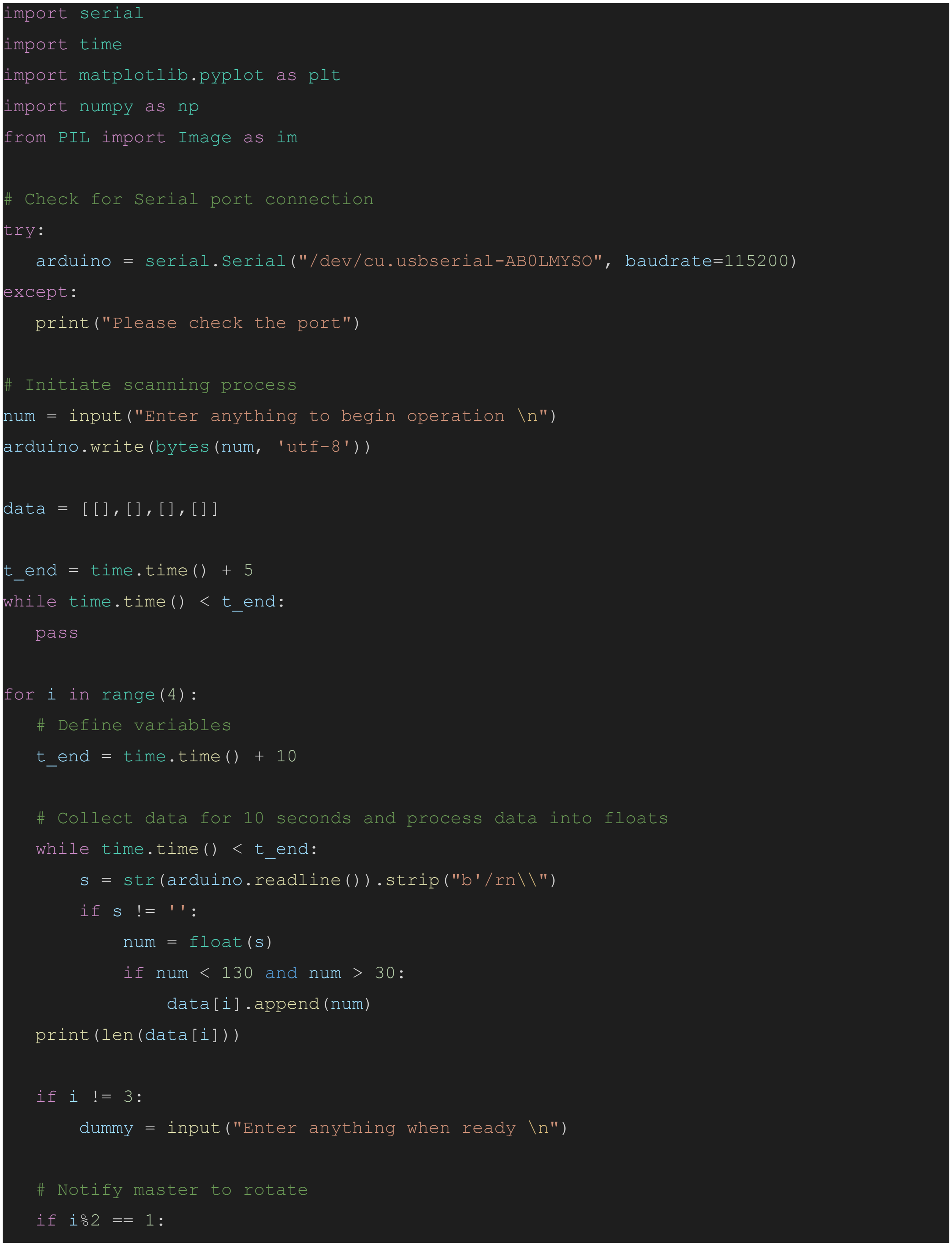
```
  // Tx on, Rx2 off
  selectMUX(2);
  delay(1000);
  // Tx off, Rx2 on
  selectMUX(5);
  result[1] = analogRead(MUX_OUTPUT);

  // Tx on, Rx3 off
  selectMUX(2);
  delay(1000)
  selectMUX(7);
  result[2] = analogRead(MUX_OUTPUT);

  // Tx on, Rx4 off
  selectMUX(2);
  delay(1000);
  selectMUX(6);
  result[3] = analogRead(MUX_OUTPUT);

  // Tx on, Rx5 off
  select(2);
  delay(1000);
  selectMUX(4);
  result[4] = analogRead(MUX_OUTPUT);
}
```

Appendix E. Image Processing Unit Code



```
import serial
import time
import matplotlib.pyplot as plt
import numpy as np
from PIL import Image as im

# Check for Serial port connection
try:
   arduino = serial.Serial("/dev/cu.usbserial-AB0LMYSO", baudrate=115200)
except:
   print("Please check the port")

# Initiate scanning process
num = input("Enter anything to begin operation \n")
arduino.write(bytes(num, 'utf-8'))

data = [[],[],[],[]]

t_end = time.time() + 5
while time.time() < t_end:
   pass

for i in range(4):
   # Define variables
   t_end = time.time() + 10

   # Collect data for 10 seconds and process data into floats
   while time.time() < t_end:
       s = str(arduino.readline()).strip("b'/rn\\")
       if s != '':
           num = float(s)
           if num < 130 and num > 30:
               data[i].append(num)
   print(len(data[i]))

   if i != 3:
       dummy = input("Enter anything when ready \n")

   # Notify master to rotate
   if i%2 == 1:
        arduino.write(bytes('s', 'utf-8'))

# Plot the output
data_1 = data[0]
x1 = np.linspace(0, len(data_1), len(data_1))
y1 = np.fft.fft(np.array(data_1))
plot_y1 = plt.subplot(221)
plot_y1.plot(x1, y1)
plot_y1.title.set_text('Water at 0')

data_2 = data[1]
x2 = np.linspace(0, len(data_2), len(data_2))
y2 = np.fft.fft(np.array(data_2))
plot_y2 = plt.subplot(222)
plot_y2.plot(x2, y2)
plot_y2.title.set_text('Air at 0')

data_3 = data[2]
x3 = np.linspace(0, len(data_3), len(data_3))
y3 = np.fft.fft(np.array(data_3))
plot_y3 = plt.subplot(223)
plot_y3.plot(x3, y3)
plot_y3.title.set_text('Water at 30')

data_4 = data[3]
x4 = np.linspace(0, len(data_4), len(data_4))
y4 = np.fft.fft(np.array(data_4))
plot_y4 = plt.subplot(224)
plot_y4.plot(x4, y4)
plot_y4.title.set_text('Air at 30')

plt.subplots_adjust(hspace = 0.5)
plt.show()
```