DEVELOPMENT AND CHARACTERIZATION OF AN FT-ICR/MS

SYSTEM FOR A NON-NEUTRAL PLASMA.

by

Brent Hicks

A senior thesis submitted to the faculty of

Brigham Young University

in partial fulfillment of the requirements for the degree of

Bachelor of Science

Department of Physics and Astronomy

Brigham Young University

April 2008

BRIGHAM YOUNG UNIVERSITY

DEPARTMENT APPROVAL

of a senior thesis submitted by

Brent Hicks

This thesis has been reviewed by the research advisor, research coordinator, and department chair and has been found to be satisfactory.

_____   _____
Date   Bryan Peterson, Advisor

_____   _____
Date   Eric Hintz, Research Coordinator

_____   _____
Date   Ross Spencer, Chair

ABSTRACT

DEVELOPMENT AND CHARACTERIZATION OF AN FT-ICR/MS

SYSTEM FOR A NON-NEUTRAL PLASMA.

Brent Hicks

Department of Physics and Astronomy

Bachelor of Science

This paper discusses the development of a Fourier transform ion cyclotron resonance mass spectrometry (FT-ICR/ms) system capable of differentiating and finding the relative abundances of different ion species in a plasma confined in a Malmberg-Penning trap. FT-ICR is a mass spectrometry technique frequently used by chemists to identify protein components with masses of several thousand amu. Unique to this system is its ability to identify lightweight ions with masses near 10 amu, corresponding to cyclotron frequencies near 1 MHz, and its ability to differentiate between cyclotron frequencies differing by about 20 Hz. Ultimately this system will be used to determine the effect of ionization on the half-life of ^{7}Be and the formation of its daughter ^{7}Li during long term confinement.

ACKNOWLEDGMENTS

Contents

List of Figures

Chapter 1

Introduction

1.1 Beryllium-7

Beryllium-7 is a radioactive isotope with an accepted half life of 53.28 days. It forms from collisions of cosmic rays with the oxygen and nitrogen in the atmosphere. Its peak production is at an altitude of about 20 km [1]. At higher altitudes production is limited because of the lack of oxygen and nitrogen in the atmosphere; production at altitudes lower than 20 km is significantly reduced because the atmosphere tends to attenuate the cosmic rays.

Beryllium-7 is unique because it is the lightest element that only decays by electron capture. Electron capture is one of the ways that a radioactive element can decay into another element. It occurs when a proton from the nucleus of the atom captures an electron and the two combine to form a neutron. The electrons most frequently involved in electron capture are the electrons closest to the nucleus, primarily the 1s electrons [2]. Since an electron is required for ^{7}Be to decay, it is possible to decrease the nuclear decay rate by ionizing the sample. Current research indicates that the half-life of ^{7}Be is dependent on chemical form [3], but no data detailing the effect of

ionization on its half-life has been published.

1.2   Long Duration Exposure Facility

In 1991 the Long Duration Exposure Facility (LDEF) satellite was retrieved from its 300 km orbit. Concentrations of beryllium-7 samples on its leading edge [4] were about 1000 times greater than can be accounted for with current knowledge of atmospheric flows and production rates. Since most ^{7}Be production is around an altitude of 20 km, it is possible that the ^{7}Be found on LDEF is mostly the result of production from lower altitudes. After being formed it is carried in atmospheric flows up to LDEF's orbit altitude.

Incident cosmic rays are energetic enough to form and ionize ^{7}Be in the atmosphere. It is possible that the half-life of ^{7}Be is greater in an ionized state, thus allowing it more time to reach the upper atmosphere and possibly explain why the LDEF satellite had such high concentrations.

A greater understanding of ^{7}Be is interesting because its formation is one of the processes that protects life on the surface of the earth from the harmful effects of cosmic rays. It is a radioactive isotope and can be used as an atmospheric transport tracer [5]. Measurements of ^{7}Be concentrations could be collected from numerous locations in the atmosphere and used to calculate how long it has been circulating. With this data, models of atmospheric currents could be created eventually leading to a better understanding of how the atmosphere works and perhaps even a better understanding of weather patterns. Beryllium-7 can also be used to study soil erosion [6] because it gets caught in water droplets in the atmosphere and precipitates in rain.

1.3   Overview

Ion containment is accomplished using a Malmberg-Penning trap. Once contained, measurements of the relative abundance of the contained ions are obtained using a technique known as Fourier transform ion cyclotron resonance mass spectrometry (FT-ICR/ms). The ions are driven by an excitation signal at a frequency equal to the cyclotron frequency. After driving the ions, detection rings detect induced charge from the orbiting ions and a Fast Fourier Transform (FFT) is used to determine the constituent frequencies. Each frequency represents a different mass in the plasma. For the eventual study of beryllium-7 the masses will correspond to either ^{7}Be or its decay product ^{7}Li.

Limitations of the containment trap will eventually be characterized as well as the accuracy and ability of the software to correctly identify the ions present in the plasma. Boron will initially be used to test the equipment since it is not radioactive. Its mass is similar enough to beryllium to allow for accurate characterization, but different enough that it will not skew data collected in the future if some of the boron remains in the device.

Chapter 2

Experimental Set-up

2.1   Ion containment

To accurately measure the half-life of ionized atoms, a device must be capable of containing the atoms for long periods of time. A half-life measurement of ^{7}Be requires a couple of months. To isolate the plasma from the outside environment, a Malmberg-Penning trap is used, see Fig. 2.1.
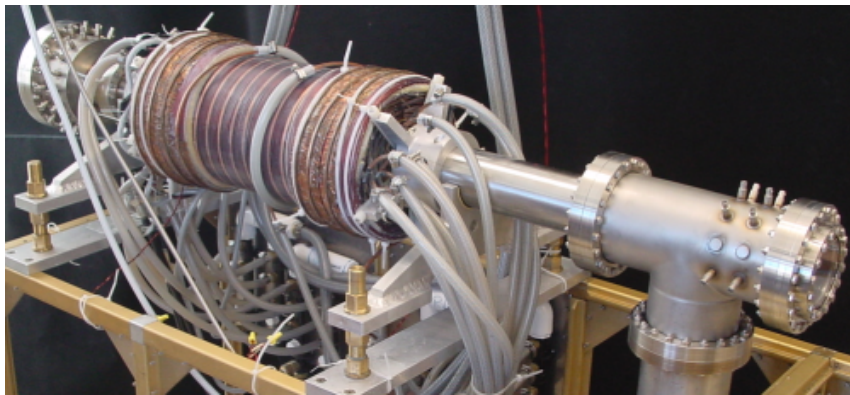
Figure 2.1 Our Malmberg-Penning trap, the plasma is contained in a 10 cm region in the center of the magnet. The copper wire is wound on a vacuum chamber and the trapping region is inside the vacuum. The magnet is 60 cm long.

A Malmberg-Penning trap is a combination of a solenoid and two high voltage end rings. Combining the solenoid with some correction coils creates a nearly uniform magnetic field parallel to the cental axis, depicted in Fig. 2.2.
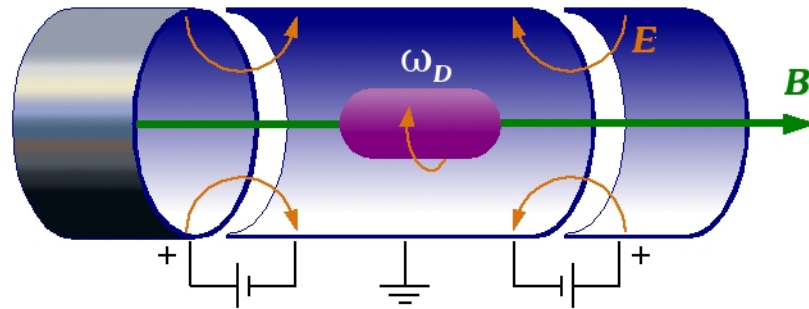
Figure 2.2 Diagram of the plasma containment chamber. Magnetic fields contain the ions axially and positive electrostatic potentials on the end rings prevent them from escaping the ends.

The uniform magnetic field in the solenoid provides plasma containment in two dimensions. Charged particles in a uniform magnetic field experience the Lorentz force $\mathbf{F} = q\left(\mathbf{v} \times \mathbf{B}\right)$ that causes them to rotate around the magnetic field lines. Since the magnetic field lines are parallel to the central axis of the solenoid, the ions are confined toward the central axis of the solenoid.

To prevent the plasma from escaping the ends of the solenoid, the end rings are given a high voltage that repels the contained ions. Since the positive ions will be studied, the voltage on the ends will be positive with a magnitude around 80-150 V.

At full field strength the magnet generates 15 kW of thermal energy. To prevent it from overheating, cooling pipes have been built into and layered with the solenoid coils. Water is pumped through the pipes to help remove excess heat. Unfortunately this has caused minor ripples [7] in the internal magnetic field, but they should not affect our ability to contain the plasma.

Over time the plasma drifts apart and would eventually come in contact with the containment chamber. Maintaining a low pressure helps prevent dispersion; our trap is a sealed chamber with a pressure of 10^{-9} - 10^{-10} torr.

2.2 Fourier transform ion cyclotron resonance mass spectrometry

Once contained, measurements of the relative abundance of the contained plasmas are obtained using FT-ICR/ms [8, 9]. The technique involves finding the cyclotron frequency,

$$f_c = \frac{q\mathbf{B}}{2\pi m} \tag{2.1}$$

from electric signals generated by the contents of the trap. The variables of the equation are m, the mass of the ion, q is the charge of the ion, and $\mathbf{B}$ is the strength of the containment magnetic field. Each frequency corresponds to a specific mass to charge ratio. Provided that the charge is known, the mass to charge ratio can be used to determine a given molecule or atom.

FT-ICR/ms is often used by chemists to identify complex protein components. Unique to our experiment is that we intend to contain a higher plasma density than typically used by chemists. A high plasma density is preferable when measuring a half-life because a greater number of ions will yield a greater number of decays per unit time. This provides more complete data for half-life calculations and reduces statistical error. The drawback of a higher plasma density is that the contained ions will interact with each other more causing frequency shifts that complicate analysis and peak differentiation.

Also different in our approach is the mass of the ions that we will contain and study. Large proteins have masses of the order of 10^4 daltons compared to the mere

7 daltons that we will contain and study. Containing smaller ions does not require magnetic fields as strong as those required to contain heavier ions. Many devices used to analyze proteins use a superconducting magnet with a field strength of 5-10 T. Since our experiment will need to be run over a couple of months, using a superconducting magnet is not possible because the field must be turned off when refilling the coolant to avoid quenching the superconductor. Our electromagnet is capable of creating a 0.4 T field, which will be sufficient to contain the ^{7}Be plasma. The drawback of lighter ions is that their cyclotron frequency is much higher (see Eq. 2.1) and faster hardware is required to excite the ions and collect the resulting signal. Using a weaker field helps reduce the cyclotron frequency and brings it into a more measurable frequency. We expect that the cyclotron frequency of ^{7}Be will be near 938 438 Hz and the frequency for ^{7}Li will be near 938 558 Hz in our magnetic field.

2.2.1 Ion cyclotron excitation

Unexcited ions have a thermal distribution of energies with each energy corresponding to a different orbital radius. The signals induced as the ions pass near the detection plates are out of phase and do not create a strong enough signal to be detected and analyzed. To detect a measurable signal, the ions need to be excited by an alternating electric field. The AC electric field necessary is the same frequency as the ion cyclotron frequency and is applied in a direction perpendicular to the magnetic field. This resonance causes the ions to accelerate and their orbit diameter to spiral outward and become in phase (Fig. 2.3).

The containment chamber has a section composed of four ring segments (Fig. 2.4). An excitation signal is applied to two opposing segments for a user specified drive time and the other two ring segments detect the transient signal after excitation. Since
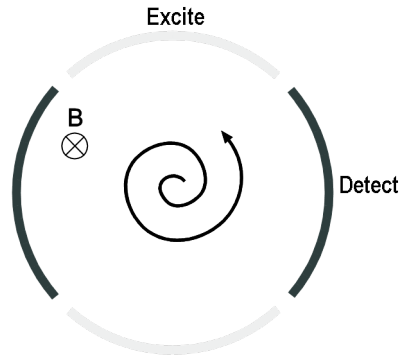
Figure 2.3 Orbit path of ions during excitation

both ^{7}Be and ^{7}Li have comparable masses, they have similar cyclotron frequencies.
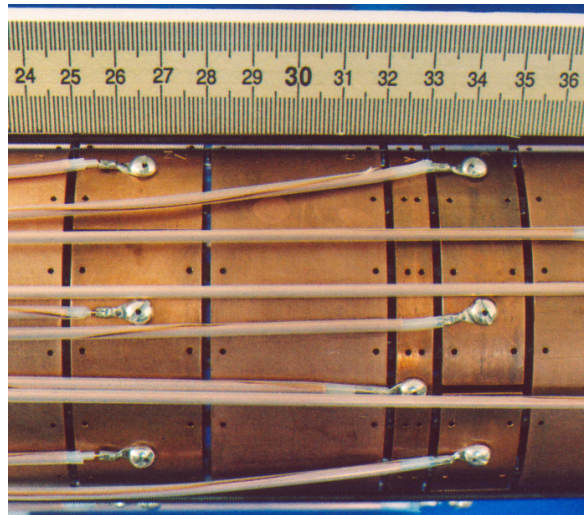
Figure 2.4 The far right of the containment center between 33 cm and 35 cm is a set of eight ring sections used to excite the ions and detect their signals.

A simple way to simultaneously excite multiple frequencies is to specify a square frequency profile and then compute the inverse fourier transform to get the required time domain excitation signal. This approach requires a very large dynamic range from the excitation hardware and suffers from Gibb's oscillations [10]. These weaknesses can be overcome by using a quadratic phase function and a smoothed square frequency profile as inputs to the inverse fourier transform [11].

First the user defines the desired square frequency profile $F(\omega)$ as diagrammed in Fig. 2.5(a). To remove the sharp discontinuities, $F(\omega)$ is smoothed as explained by Guan in [12] by using

$$F_1(\omega) = \frac{1}{\Delta\omega} \int_{\omega-\Delta\omega/2}^{\omega+\Delta\omega/2} F(\omega')\, d\omega'. \tag{2.2}$$

Equation 2.2 is a moving average. It steps through the frequency profile $F(\omega)$. At each frequency ω, it redefines the function value as the average of the profile over a frequency range $\Delta\omega$ defined by the user. The smoothing results in a profile similar to Fig. 2.5(b).

After smoothing the function, $F_1(\omega)$ is used to find the phase function $P(\omega)$. The phase function, when recombined with $F_1(\omega)$, serves to reduce the required dynamic range and spreads the excitation power over a longer time period. Guan and McIver state in [11] that the quadratic phase function can be calculated from the *power spectral profile* $G(\omega) = |F_1(\omega)|^2$ and

$$P(\omega) = \frac{t_1 - t_0}{\int_{\omega_0}^{\omega_1} G(y)\, dy} \int_{\omega_0}^{\omega_1} \int_{\omega_0}^{y} G(x)\, dx\, dy + t_0(\omega - \omega_0) + P_0 \tag{2.3}$$

where $t_1 - t_0$ is the time period that the excitation power is spread over, ω_0 and ω_1 signify the frequency range over which the phase function is desired, and P_0 is the initial phase which may be set to zero. The final phase function is similar to Fig. 2.5(c). Once $F_1(\omega)$ and $P(\omega)$ are known, their values are combined into a complex number and a complex inverse Fourier transform generates the final time domain excitation signal, see Fig. 2.5(d).

The LabVIEW code that I have written to calculate the excitation signal is included in Fig. A.4. It's input and outputs are shown in Fig. 2.5.

(a) User specifies the desired frequency profile

(b) The program smooths the initial profile to reduce Gibb's oscillations

(c) The phase modulation is calculated

(d) The phase and magnitude information is combined and an inverse Fourier transform yields the time signal
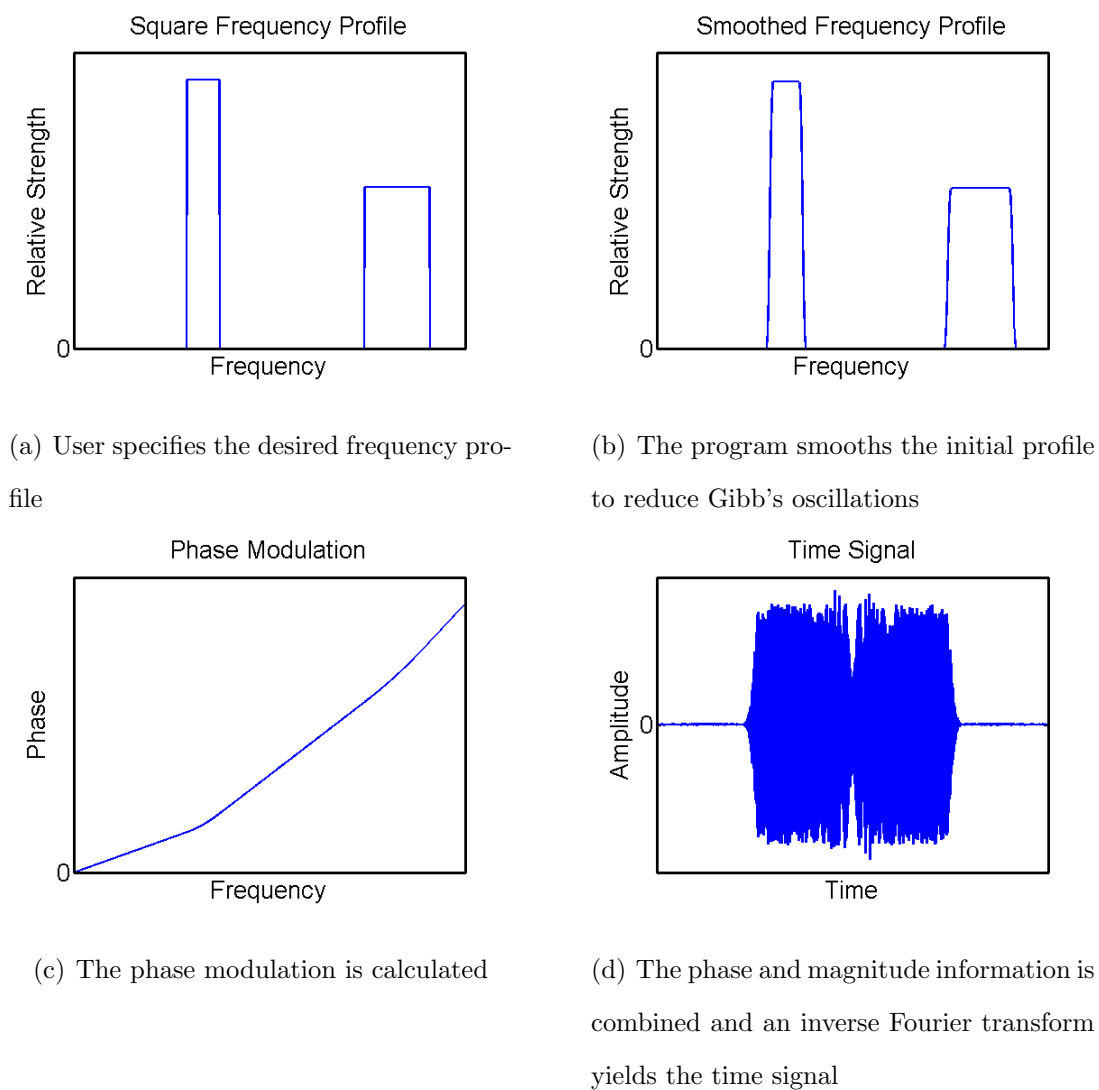
Figure 2.5

2.2.2 Ion cyclotron detection

After driving the plasma, the transient signal is received by two ring segments in the containment chamber, amplified through a series of electronics, and recorded by a computer. To prevent aliasing, data must be collected at a higher frequency than the highest frequency expected in the signal. This frequency is known as the Nyquist frequency. Data sampling rates must be at least twice the Nyquist frequency to correctly measure time signals. Since the highest frequency that we expect to detect is about 1 MHz, comparable to the cyclotron frequency of the contained ^{7}Be and ^{7}Li ions, we need a sample rate of at least 2 MHz.

The current hardware is a National Instruments PCI-6110 multifunction data acquisition board (DAQ). It has 12 bit input resolution and is capable of collecting 5 million samples per second. With this set-up, signals up to about 2.5 MHz can reasonably be measured within a few Hz accuracy. The DAQ board will also be used to excite the contained plasma. The board's output resolution is 16 bits and is capable of outputting 4 million samples per second.

Driving Circuit

The hardware used to generate the excitation signal does not provide sufficient current to properly drive the high capacitance of the cables and ring segments. An additional amplifier is used to boost the current and to split the signal into two channels that are 180 degrees out of phase. The two channels are used to drive opposing ring segments in the containment chamber. Figure 2.6 is the electronic schematic used for the driving circuit. Resistors 3 and 4 are required for stability. Resistor 1 was given a 12.4 kΩ value to limit the current drawn from the DAQ board.
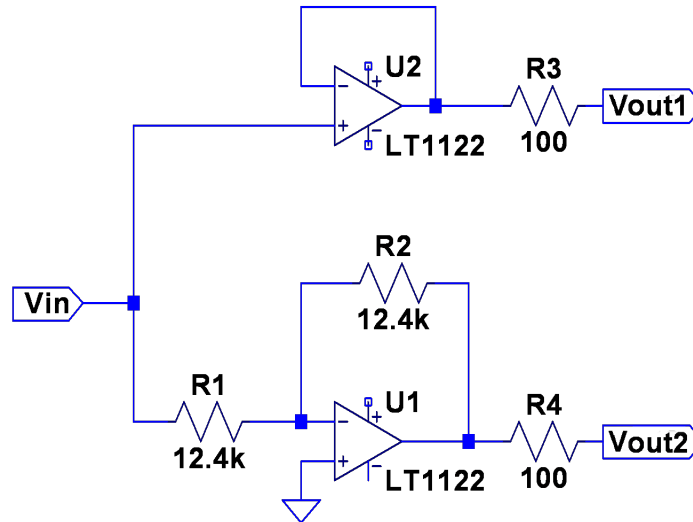
Figure 2.6 Plasma driving circuit

Receiving Circuit

The receiving circuit is designed to take a 10 μA current signal from two channels (two opposing ring segments), find the difference between the two signals, and amplify the result so that it outputs a ± 0.5 V signal for collection by the computer DAQ. Figure 2.7 is the electronic schematic used for the receiving circuit. The 110 pF capacitors are necessary to maintain circuit stability and also remove a resonance at about 300 kHz that is present when smaller capacitors are used.

2.2.3 Circuit Characterization

A signal generator connected to a 7.6 m coaxial cable was used as the input signal to characterize the drive circuit. The outputs of the drive circuit were connected to two of the ring segments in the ion trap to provide the capacitive load for the circuit. Two 1 m cables were then connected from each of the rings to an oscilloscope for the
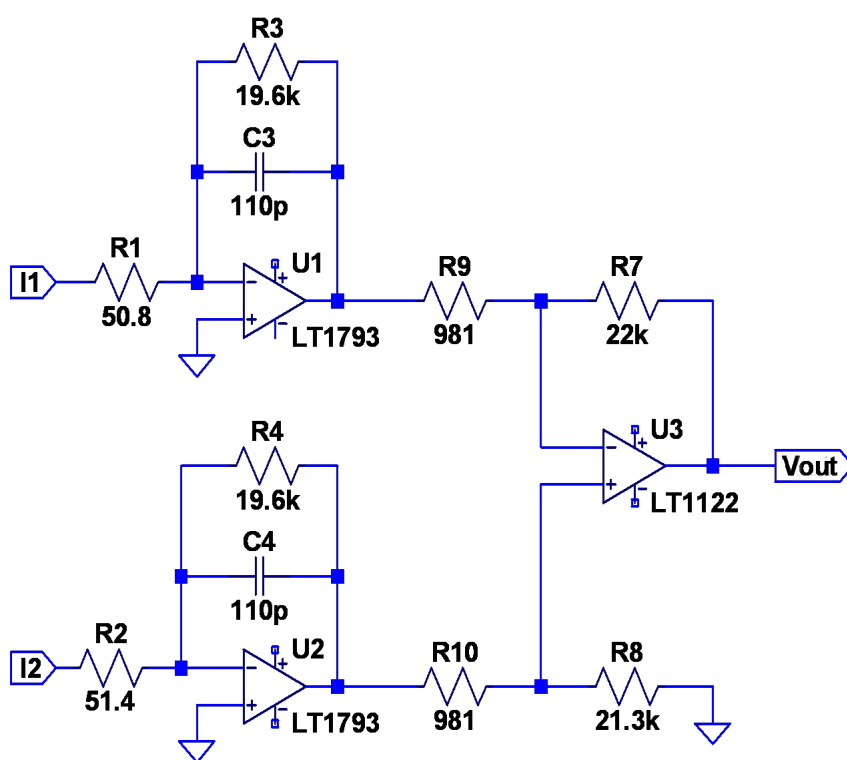
Figure 2.7 Transient receiving circuit

characterization. The signal generator was programmed to output sinusoidal signals varying from 100 kHz to 2 MHz in 25 kHz steps. After the first pass, a 99.8 Ω resister was added to the signal generator output to measure the output impedance. The characterization software used the input signal as a reference to calculate and record the gain and phase of the output signal.

The receiving amplifier is designed to accept a 10 μA input current rather than an input voltage. To calibrate the receiving amplifier a signal generator was connected directly to a ring segment adjacent to the ring segment connected to the receiving amplifier. The rings are close enough that they are capacitively coupled. The magnitude of the signal from the signal generator was adjusted until the coupled setup provided the same output as a previous setup using a 39.8 pF blocking capacitor. The output current of the capacitor is calculated from Eq. 2.4; at 1 MHz, a 50 mV signal from the signal generator provides 12.5 μA.

$$I = C\omega V \tag{2.4}$$

2.2.4   Signal analysis

Since both ^{7}Be and ^{7}Li will be present in the trap, the signal detected will be a jumbled mess of both their frequencies. Figure 2.8(a) is a simulated signal composed of both ^{7}Be and ^{7}Li frequencies with a damping term added. Damping occurs as the excited ions collide with other molecules which causes their orbital radius to decrease. Damping can be reduced by keeping the chamber at a low pressure to minimize the number of rogue molecules.

To extract which frequencies are present and figure out how much ^{7}Be and ^{7}Li is contained, a mathematical method known as a Fast Fourier Transform (FFT) is used. It determines the component frequencies of a transient signal. These can

then be plotted as in Fig. 2.8(b). Notice that this graph has peaks at the cyclotron frequencies of both ^{7}Be and ^{7}Li.
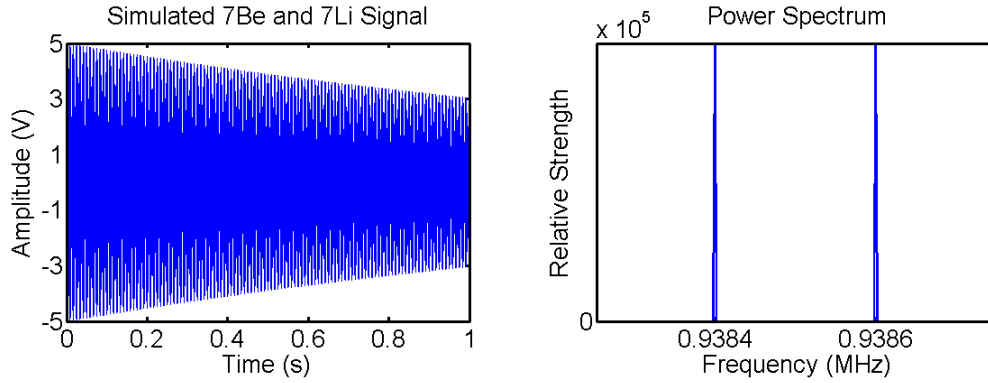
(a) Simulated time signal with damping   (b) Result of FFT

Figure 2.8 Example of an input signal and the FFT output of its frequency spectrum.

After analyzing the signal and finding its component frequencies, we are able to compare the component frequencies with the cyclotron frequencies of ^{7}Be and ^{7}Li. The area under the peaks in the frequency plot is representative of the relative abundance of the mass present. Since the end result of our research is to calculate the nuclear decay rate of ^{7}Be, we will excite the ions periodically over the course of several months, compute the FFT, and then calculate how many of the contained ions are ^{7}Be and how many are ^{7}Li. From that data, the half-life of ionized ^{7}Be can be calculated.

2.3   System Characterization

After developing all of the code and circuits necessary to excite and detect the component plasmas, all of the pieces needed to be tested to ensure that they all worked together and would yield the desired results. A 0.25 s excitation signal based on a

1 V amplitude square frequency profile from 940 kHz to 960 kHz was calculated using the signal calculation VI diagrammed in Fig. A.4. The signal was sent through the drive amplifier, connected to capacitively coupled rings in the chamber and returned through the receiving amplifier to the computer input (Fig. 2.9).
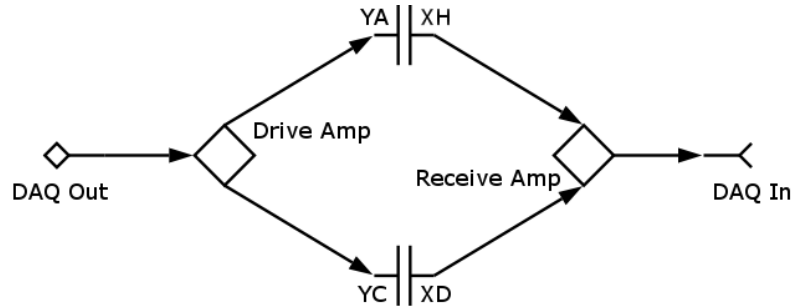
Figure 2.9 Layout used to test the FT-ICR system. The two-letter combinations next to the capacitor plates correspond to adjacent rings in the chamber.

This set-up was used to ensure that the computer could successfully send the computed excitation signal through the amplifiers and then collect the same signal on the other end. After the signal returned to the computer its FFT was calculated to ensure that the original frequency profile was returned.

After testing the 940 kHz to 960 kHz frequency band through the amplifiers, another test was conducted to determine the frequency resolution of the system. Since we expect that the difference between the cyclotron frequency of ^{7}Be and ^{7}Li to be near 100 Hz another excitation signal with 1 V peaks at 950 000 Hz and 950 100 Hz was generated and sent through the amplifiers. The resulting signal was then run through the MATLAB scripts contained in Appendix A to verify the correct peak frequency locations and their relative magnitudes.

Chapter 3

Results and Conclusion

3.1   MATLAB Characterization

All of the MATLAB scripts used to identify the cyclotron frequencies of the transient signal are included in Appendix A.1. A stand alone signal generator was used to generate different frequency and amplitude signals, which were then summed together and processed through the scripts. Table 3.1 contains the components of the input signal to the scripts and the results from the scripts. Notice that although the output magnitudes are not equal to the input magnitudes, the relative magnitudes are preserved. This is acceptable for our application because we do not need to know exactly how much ^{7}Be or ^{7}Li is in the chamber at any given time. Over time the relative amplitudes may be compared and exponentially fit to determine the ^{7}Be half-life.

It is also very important for the MATLAB scripts to be able to correctly identify the cyclotron frequencies because they will be used to confirm what ion is present in the chamber. The results in Table 3.1 again affirm the script accuracy to within a few hertz. There are small differences between the input and output frequencies but it appears to be the result of either the signal generator or the DAQ's ability to

correctly clock out the signal. The absolute error increases with increasing frequency but even at 1 MHz the error is still only 10 Hz. Since the expected difference between the cyclotron frequencies of ^{7}Be and ^{7}Li is about 100 Hz there should be no problem.

Input		Output	
0.5 V	333 000 Hz	1	333 003 Hz
1 V	1 000 000 Hz	1.976	1 000 010 Hz
2 V	500 000 Hz	3.979	500 005 Hz
3 V	5 000 Hz	6.022	4 999 Hz
5 V	1 000 Hz	10.047	999 Hz

Table 3.1 The MATLAB script works correctly over a wide frequency range

3.2 Amplifier Characterization

Since the computer DAQ is limited in its ability to output large enough current to drive the excitation rings, an additional amplifier is required to not only amplify the signal but to also split it into two channels 180° out of phase. Figure 3.1(a) shows the ratio between the gain of the two outputs from the circuit diagrammed in Fig. 2.6. Ideally the gain ratio would be a constant value of one over all frequencies. At the cyclotron frequencies we are interested in, the gain ratio is relatively flat but it has a value of 0.9. This means that one of the rings will accelerate the ions more but we do not expect that to be detrimental to the system.

More important is that the amplifier outputs are 180° out of phase. The phase difference between the two is diagrammed in Fig. 3.1(b). Again at the cyclotron frequencies of ^{7}Be and ^{7}Li the phase difference is within one or two degrees of 180°.

Interesting things do start to happen above 1 MHz but even up to about 1.25 MHz the phase difference is still within five degrees.

(a) Gain Ratio   (b) Phase Difference
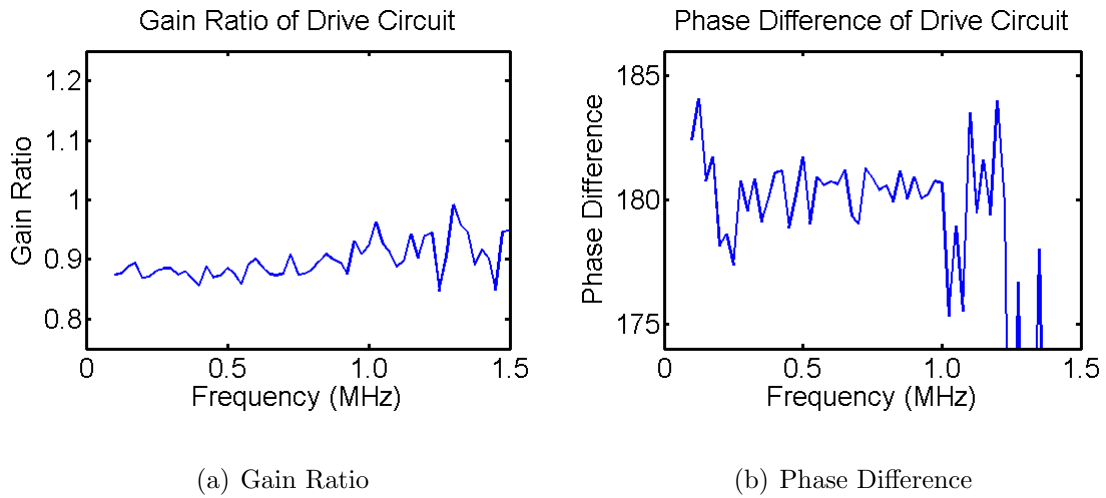
Figure 3.1 Drive Circuit Characterization

As with the driving circuit, the ideal receiving circuit would have a gain ratio of one and a phase difference of 180°. Figures 3.2(a) and 3.2(b) show that again, perfection is not attained, however at the desired cyclotron frequencies the gain ratio is close enough to one and the phase difference is within five degrees of 180°.

(a) Gain Ratio   (b) Phase Difference
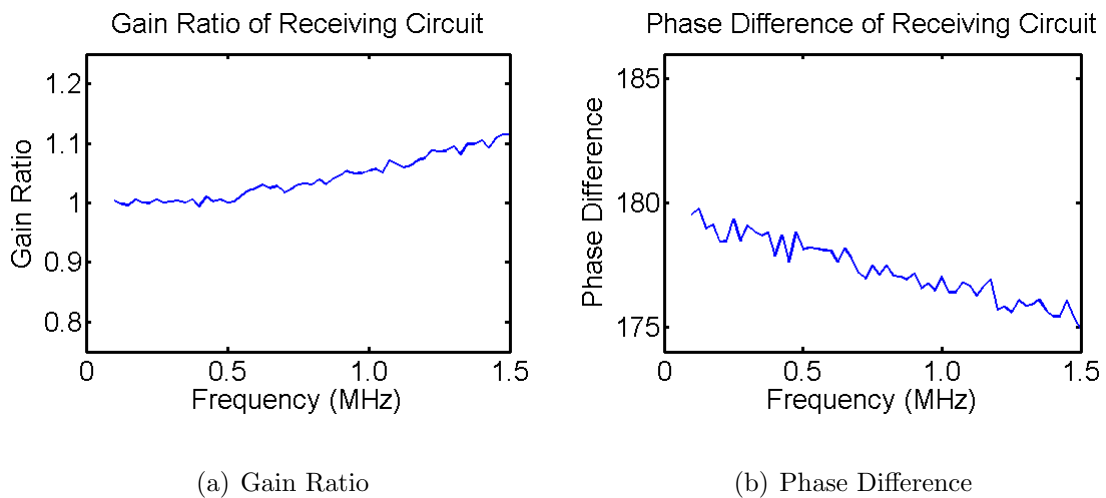
Figure 3.2 Receiving Circuit Characterization

3.3   System Characterization

This section contains the results from specifying the desired frequency profile in Lab-VIEW, generating the excitation time signal, outputting the signal through the drive amplifier, through capacitively coupled rings, back through the receiving amplifier, and then through the MATLAB scripts. In short, these results show how well the entire FT-ICR system works at exciting defined frequencies and detecting the ion cyclotron frequencies.

3.3.1   Frequency Bands

Since we are not certain what the exact ion cyclotron frequencies of ^{7}Be and ^{7}Li will be in the trap, it will be much easier to excite an entire band of frequencies in the right ballpark. Figure 3.3(a) is the result of a defined frequency profile centered on 950 kHz and a bandwidth of 10 kHz. We expect that both the beryllium and lithium cyclotron frequencies will fall somewhere within this range. This result was acquired by outputting the signal and then immediately reading it again without the amplifiers or the excitation rings. Ideally this would result in a clean, square profile between 940 kHz and 960 kHz without any spikes to zero. We believe that the spikes to zero are an artifact of the data acquisition and signal output.

Figure 3.3(b) is the result of the same excitation signal as the data given in Fig. 3.3(a), but it also includes the effect of the amplifiers and the capacitive coupling. Again we believe that the spikes to zero are artifacts. Different in this test is the amplitude decay with increasing frequency. This is likely the result of the combined effects of the driving and receiving amplifiers since they do not provide perfectly flat gain amplitudes. Fortunately the amplitude decrease over this 20 kHz range is minimal. Again, since the difference between the ^{7}Be and ^{7}Li cyclotron frequencies

(a) Spectra result without amplifiers   (b) Spectra result with amplifiers
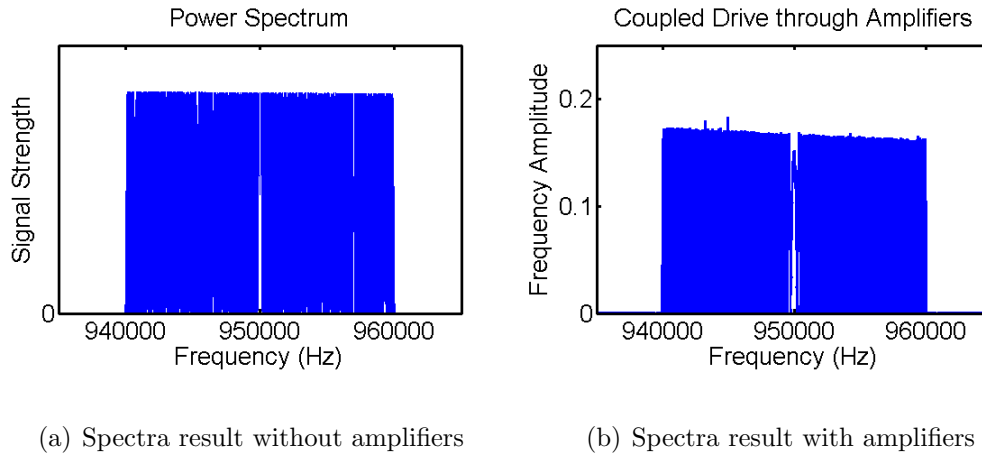
Figure 3.3 Results from frequency band excitation

is only expected to be about 100 Hz, the frequency dependence of the amplitude will be insignificant.

3.3.2 Frequency Resolution

Perhaps one of the most important results for the FT-ICR system is to know whether it is capable of detecting distinct frequencies as close as 100 Hz apart and determining their relative amplitudes. To determine this an excitation signal with 1 V peaks at 950 000 Hz and 950 100 Hz was generated, sent through the amplifiers, and then run through the MATLAB scripts. As shown in Fig. 3.4 two distinct peaks were easily resolved. Likely the system will be able completely resolve signals as close as 20 Hz apart ensuring sufficient ability to distinguish between contained ^{7}Be and ^{7}Li.

Notice the oscillations at the base of the 950 100 Hz peak that are not present in the 950 000 Hz peak. This is likely the result of the same discretization as exhibited in Figs. 3.3(a) and 3.3(b).

In addition to being able to distinguish the two cyclotron frequencies between each other, knowing what is in the trap, the system needs to determine how strong each
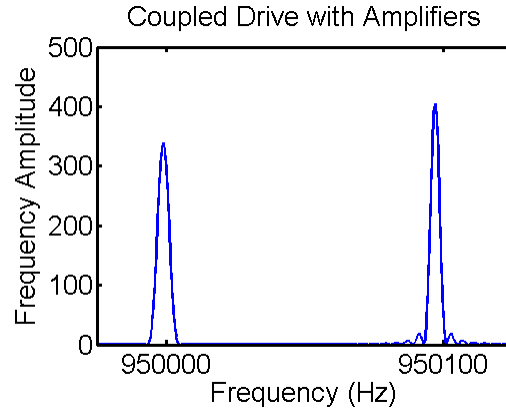
Figure 3.4 Frequency resolution of capacitively coupled rings

of the frequencies is, how much of what is in the trap. Table 3.2 shows the results from a generated signal through the amplifiers, detected, and analyzed in MATLAB. Notice that the frequencies generated and the frequencies detected are within a few Hz of each other. Similarly, the relative amplitudes of both are comparable as well. Since the amplitude is found by summing the area under the curve, it is possible that the amplitude difference in the detected signals is the result of the oscillations in the 950 100 Hz. Thus not only can the FT-ICR system determine what ion species are present, but it can also accurately determine the relative quantities of the contained species.

Generated Signal		Detected Signal	
1 V	950 000 Hz	437	949 998 Hz
1 V	950 100 Hz	425	950 096 Hz

Table 3.2 Results showing the ability of the FT-ICR system to correctly generate excitation signals and detect the components and powers of incoming transient signals.

3.4   Conclusion

I am quite confident that the FT-ICR system is capable of creating correct ion exci-tation signals, exciting the ions, and then determining contained ion species and their relative abundances. It should be straightforward enough to use the system to collect the required data to determine the half-life of ionized ^{7}Be. The most likely obstacle the FT-ICR system may face is insufficient ability to detect the induced charge on the detection rings as the ions cycle in the trap. Since the receiving amplifier was designed to collect a 10 μA current, it could potentially be modified to overcome this obstacle.

If greater amplitude accuracy is desired, the MATLAB scripts could be modified to not just find the area under the power spectrum, but to also check if the oscillations are present. If they are present, it could fit a curve along the peaks of the oscillations and then find the area. If this were done, it is possible that the detected amplitudes as listed in Table 3.2 would be the same.

Bibliography

[1] J. Arnold and H. Ali Al-Salih, "Beryllium-7 Produced by Cosmic Rays," Science **121,** 451–453 (1955).

[2] P. A. Voytas *et al.*, "Direct Measurement of the L/K Ratio in ^{7}Be Electron Capture," Physical Review Letters **88,** 012501 (2001).

[3] C. Huh, "Dependence of the decay rate of ^{7}Be on chemical forms," Earth and Planetary Science Letters **171,** 325–328 (1999).

[4] G. Fishman *et al.*, "Observation of ^{7}Be on the surface of LDEF spacecraft," Letters to Nature **349,** 678–680 (1991).

[5] D. Koch and D. Rind, "Beryllium 10/beryllium 7 as a tracer of stratospheric transport," Journal of Geophysical Research **103,** 3907–3917 (1998).

[6] W. Blake, D. Walling, and Q. He, "Fallout beryllium-7 as a tracer in soil erosion investigations," Applied Radiation & Isotopes **51,** 599–605 (1999).

[7] D. K. Olson, "Construction and Characterization of a Specialized Malmberg-Penning Trap for Confinement of a Nonneutral Radionuclide Plasma," Senior Thesis (Brigham Young University, Provo, UT, 2005).

[8] M. B. Comisarow, "Fundamental Aspects and Applications of Fourier-Transform Ion-Cyclotron Resonance Spectroscopy," Analytica Chimica Acta **178,** 1–15 (1985).

[9] *FT-ICR/MS Analytical Applications of Fourier Transform Ion Cyclotron Resonance Mass Spectrometry*, B. Asamoto, ed., (VCH Publishers, New York, NY, 1991), pp. 45–46.

[10] S. Guan and A. G. Marshall, "Stored waveform inverse Fourier transform (Swift) ion excitation in trapped-ion mass spectrometry: theory and applications," International Journal of Mass Spectromety and Ion Processes **157,** 5–37 (1996).

[11] S. Guan and R. McIver, "Optimal phase modulation in a stored wave form inverse Fourier transform excitation for Fourier transform mass spectrometry. I. Basic algorithm," Journal of Chemical Physics **92,** 5841–5846 (1990).

[12] S. Guan, "Optimal phase modulation in a stored wave form inverse Fourier transform excitation for Fourier transform mass spectrometry. II. Magnitude Spectrum Smoothing," Journal of Chemical Physics **93,** 8442–8445 (1990).

Appendix A

FT-ICR Analysis Code

A.1 MATLAB Codes

A.1.1 Main Analysis Code

This is the main MATLAB analysis code used to find the component frequencies and their relative amplitudes from the time signal received after ion excitation. The functions *readfticr*, *dofft*, and *findextreme* are user built functions whose descriptions follow.

```
% Code used to analyze the signal from the FT-ICR

%Import the transient signal from the LabVIEW binary
filename=input('Enter filename: ');
[data, timebase] = readfticr(filename)

%Compute the power spectrum of the signal and plot P vs. frequency
[P, f, df] = dofft(data, timebase, 1);
clear data;

%Find location of the peaks in the power spectrum and corresponding
%magnitudes
[xmax,imax] = findextreme(P);

%Calculate the area under the peaks of the power spectrum and output
%summary

%integration range max and min
irmin=ceil(15*df); %How far left of the peak you want to integrate
```

```
irmax=floor(15*df); %How far right of the peak you want to integrate
for n=1:length(imax)
    if imax(n) < irmin | imax(n) > (length(P)-irmax)
        fprintf('Error - Peak too close to grid edge to integrate.\n')
        continue
    end
    A(n,1)=f(imax(n));  %load the frequency
    A(n,2)=xmax(n); %load the magnitude
    A(n,3)=trapz(P(imax(n)-irmin:imax(n)+irmax));   %find the area
end
A=sortrows(A,3);
fprintf('Frequency \t Relative Area \n');
m=1;
[I J]=size(A);
for n=1:I
    if A(n,1)==0    %remove DC offset component
        m=m+1;
        continue
    end
    A(n,4)=sqrt(A(n,3)/A(m,3));
    fprintf('%6.0f \t\t %6.6f\n',A(n,1),A(n,4));
end

% Matrix A composition: [Frequency, Magnitude, Area, Relative Area]
```

A.1.2   Read LabVIEW Binary

Since greater FFT resolution is achieved by collecting data at higher sampling rates and longer periods of time, it is beneficial to save the transient signal following ion excitation as a binary file. The binary files are smaller than their ASCII equivalents and LabVIEW is able to save and open binary files more quickly. However, MATLAB needs to know how to read the data from the binary files before it is able to analyze it. This script opens the binary file output by LabVIEW and creates a time-step size variable and an array that contains all of the equally spaced data.

```
function [data, timebase] = readfticr(filename)

%File used to read the binary output of LabVIEW from the FT-ICR data

fid=fopen(filename,'rb');
npts=fread(fid,1,'int32','ieee-be');
timebase=fread(fid,1,'float32','ieee-be');
```

```
npts=npts-1;
data=fread(fid,npts,'float32','ieee-be');
fclose(fid);
```

A.1.3   Compute FFT

This function computes the power spectrum as a function of frequency from the input time signal. It will also plot the result if desired, however the plot limits will likely need to be adjusted to make the plot option more useful.

```
function [P, f, df] = dofft(x, dt, makeplot)
%P:power spectrum, f:frequency array of powerspectrum,
%df:frequency spacing %x:input data array, dt:data spacing,
%makeplot:plot power spectrum if true (1)

N=length(x);

g=fft(x);
g=g(1:ceil(N/2));
P=dt/(2*pi)*(abs(g).^2);
clear g;

df=1/(N*dt);
f=0:df:df*(length(P)-1);

if makeplot==1
    figure
    plot(f,P,'b-','LineWidth',1.5)
    aa=gca;
    set(aa,'XLimMode','manual','YLimMode','manual',...
        'Xlim',[938350 938650],'Ylim',[0 max(P)],'FontSize',20,...
        'LineWidth',2.5,'XTick',[938438 938558],...
        'XTickLabel',{'938438','938558'},'YTick',[0])
    xlabel('Frequency (Hz)')
    ylabel('Relative Strength')
    title('Power Spectrum')
end
```

A.1.4   Find Peaks of FFT

I found this function publicly posted online at the MATLAB Central File Exchange. Its purpose is to locate extreme values from a set of data and return the values along with their indexes. I've modified it so that it only returns the maximums,

although it does still go through and find all of the minima as well. The line "a = find(dx>(sqrt(max(x(2:end))))/5e2));" may still need to be modified to tweak how sharp a local slope needs to be to be considered a peak. I played around with it a little bit until I got what I consider acceptable results but it may still need work.

```
function [xmax,imax] = findextreme(x)
%EXTREME   Extreme points of a vector.
%   [XMAX,IMAX,XMIN,IMIN] = EXTREME(X) returns the minima and maxima
%   points of the vector X ignoring NaN's, where
%    XMAX - maxima points in descending order
%    IMAX - indexes of the XMAX
%    XMIN - minima points in descending order
%    IMIN - indexes of the XMIN
%
%   Example:
%      x = 2*pi*linspace(-1,1);
%      y = cos(x) - 0.5 + 0.5*rand(size(x)); y(40:45) = 1.85;
%      y(50:53)=NaN;
%      [ymax,imax,ymin,imin] = extreme(y);
%      plot(x,y,x(imax),ymax,'g.',x(imin),ymin,'r.')
%
%   See also EXTREME2, MAX, MIN


%   Written by
%   Lic. on Physics Carlos Adrin Vargas Aguilera
%   Physical Oceanography MS candidate
%   UNIVERSIDAD DE GUADALAJARA
%   Mexico, 2004
%
%   nubeobscura@hotmail.com


%   original code has been modified for use in ft-icr by
%   Brent Hicks, Brigham Young University, April 2007

xmax = [];
imax = [];
xmin = [];
imin = [];

% Vector input?
Nt = numel(x);
if Nt ~= length(x)
 error('Entry must be a vector.')
```

```
end

% NaN's:
inan = find(isnan(x));
indx = 1:Nt;
if ~isempty(inan)
 indx(inan) = [];
 x(inan) = [];
 Nt = length(x);
end

% Difference between subsequent elements:
dx = diff(x);

% Is an horizontal line?
if ~any(dx)
 return
end

% Flat peaks? Put the middle element:
a = find(dx>(sqrt(max(x(2:end)))/5e2)); % Indexes where x changes a lot
lm = find(diff(a)~=1) + 1;    % Indexes where a do not changes
d = a(lm) - a(lm-1);          %5e4 Number of elements in the flat peak
a(lm) = a(lm) - floor(d/2);   % Save middle elements
a(end+1) = Nt;

% Peaks?
xa  = x(a);              % Serie without flat peaks
b = (diff(xa) > 0);      % 1  =>  positive slopes (minima begin)
                         % 0  =>  negative slopes (maxima begin)
xb  = diff(b);           % -1 =>  maxima indexes (but one)
                         % +1 =>  minima indexes (but one)
imax = find(xb == -1) + 1; % maxima indexes
imin = find(xb == +1) + 1; % minima indexes
imax = a(imax);
imin = a(imin);

nmaxi = length(imax);
nmini = length(imin);

% Maximum or minumim on a flat peak at the ends?
if (nmaxi==0) && (nmini==0)
 if x(1) > x(Nt)
```

```
  xmax = x(1);
  imax = indx(1);
  xmin = x(Nt);
  imin = indx(Nt);
 elseif x(1) < x(Nt)
  xmax = x(Nt);
  imax = indx(Nt);
  xmin = x(1);
  imin = indx(1);
 end
 return
end

% Maximum or minumim at the ends?
if (nmaxi==0)
 imax(1:2) = [1 Nt];
elseif (nmini==0)
 imin(1:2) = [1 Nt];
else
 if imax(1) < imin(1)
  imin(2:nmini+1) = imin;
  imin(1) = 1;
 else
  imax(2:nmaxi+1) = imax;
  imax(1) = 1;
 end
 if imax(end) > imin(end)
  imin(end+1) = Nt;
 else
  imax(end+1) = Nt;
 end
end
xmax = x(imax);
xmin = x(imin);

% NaN's:
if ~isempty(inan)
 imax = indx(imax);
 imin = indx(imin);
end

% Same size as x:
imax = reshape(imax,size(xmax));
```

```
imin = reshape(imin,size(xmin));

% Descending order:
[temp,inmax] = sort(-xmax); clear temp
xmax = xmax(inmax);
imax = imax(inmax);
[xmin,inmin] = sort(xmin);
imin = imin(inmin);



% Carlos Adrin Vargas Aguilera. nubeobscura@hotmail.com
```

A.2 LabVIEW Programs

LabVIEW VIs have been written to take a user desired excitation frequency profile and compute the required time domain excitation signal. Once the signal is calculated it is saved as a binary file for later use since it is rather time consuming to compute the signal. There is also a VI that outputs the time signal to the drive amplifier and a VI to collect the post excitation transient signal from the receiving amplifier and save it to a binary file for export to MATLAB.

A.2.1 Frequency Profile

A user specifies the desired excitation frequency profile in this VI. The profile is specified by choosing a center frequency, a bandwidth and an amplitude; these three parameters are stored in a cluster which is then stored in an array. Multiple frequency bands can be added to the profile by defining additional array elements. All frequencies are given in Hz. The bandwidth is the number of Hz on either side of the center frequency.
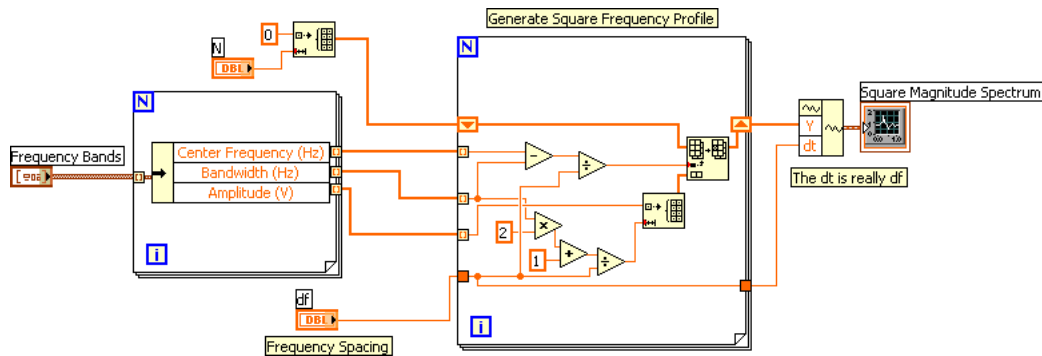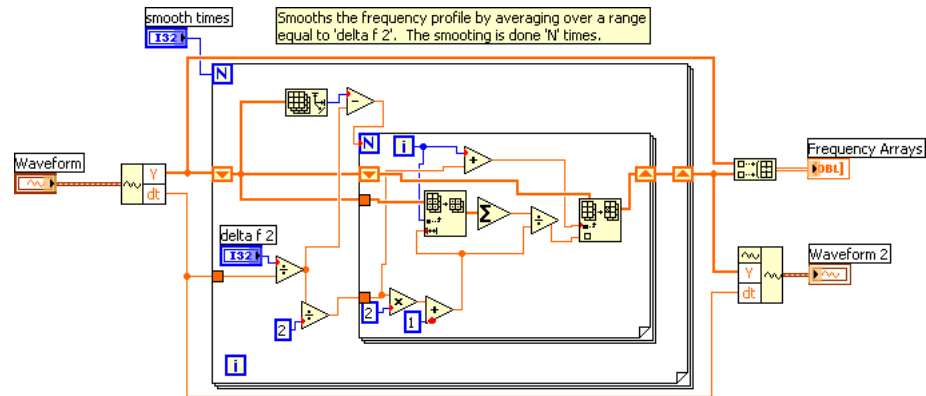

Figure A.1 Virtual instrument used to build a square frequency profile.

A.2.2 Smooth Profile

To minimize Gibb's phenomenon during the Fourier transforms, it is beneficial to smooth the sharp edges of the frequency profile beforehand. This VI steps through the frequency array discussed in Sec. A.2.1 element by element averaging it with the next few elements comprising the frequency width defined by *delta f 2*. The smoothing will be completed N times, again user definable.

Figure A.2 Virtual instrument used to smooth the square frequency profile to eliminate Gibb's oscillations.

A.2.3 Compute Signal

After the signal has been smoothed it is can be used to calculate the corresponding time signal. As currently written, this is the only VI that needs to be run to compute the time signal. The user defines all necessary parameters in this VI and then it calls the *Frequency Profile* and *Smooth Profile* VIs as discussed in the previous sections. The phase function is found and combined with the frequency profile, the combination of the two is mirrored based on whether there are an even or odd number of elements, and then the inverse complex Fourier transform is called to get the time signal. The VI normalizes all of the amplitudes on the greatest amplitude, giving it a magnitude of 1 V so that the time signal does not contain any amplitudes exceeding the 5 V limit of the DAQ. Last the excitation signal is saved as a binary file for later use.
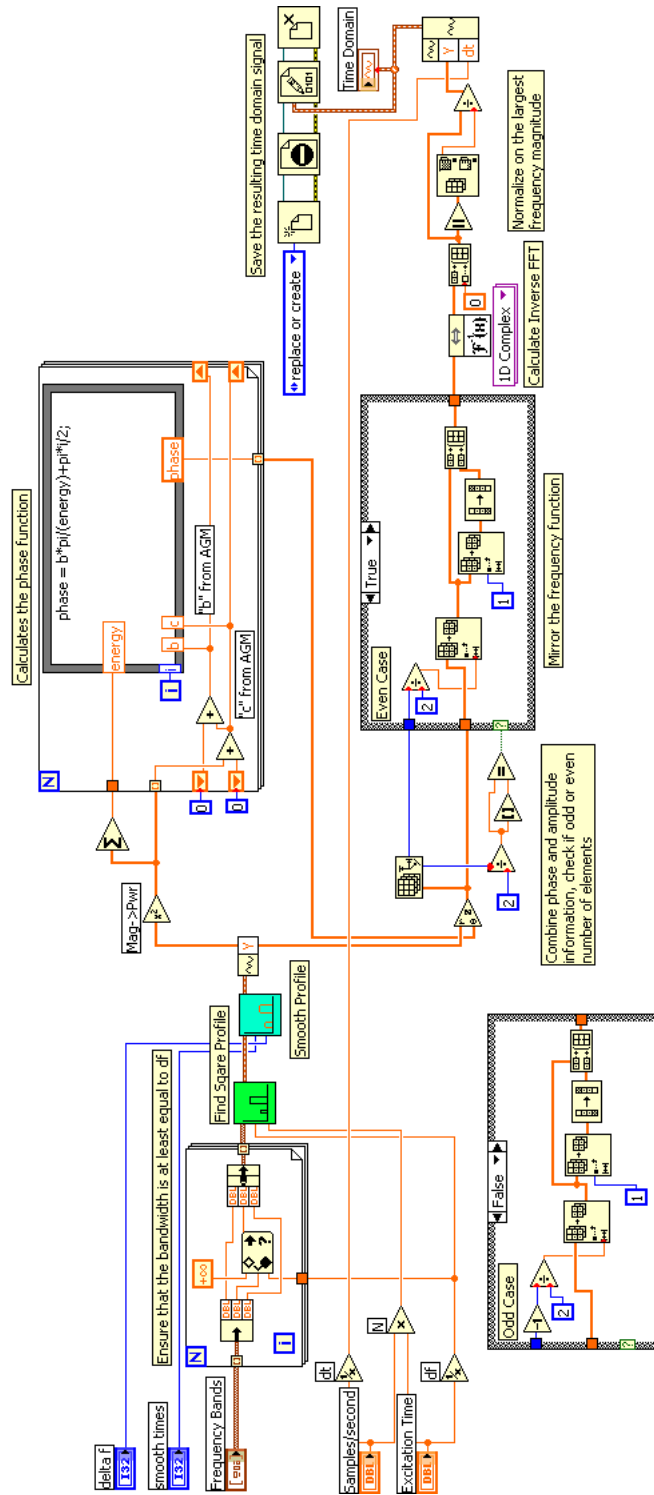
Figure A.3 Virtual instrument used to calculate the ion excitation signal and save it as a binary file.

A.2.4 Open Signal

Since the excitation signal is stored as a binary file, this simple VI is necessary to tell LabVIEW how to extract the data for further use.

Figure A.4 Virtual instrument used to open the saved binary signal file.

A.2.5 Output Signal

This VI takes as input the opened binary time signal and outputs it on the user defined analog output on the DAQ board. The *Sample Clock* function ensures that the board's hardware timing is used to get accurate signal representation.

Figure A.5 Virtual instrument used to output the ion excitation signal.

A.2.6 Signal Input

After the plasma is excited the computer collects the transient signal using this VI. Again it is hardware timed. The transient signal is converted to single precision since it provides sufficient precision and yields a smaller file size. Again to increase file write speed and minimize file size, the data is saved as a binary file for later analysis in MATLAB. The file name is automatically generated based on the date and time the data is collected.

Figure A.6 Virtual instrument used to measure the ion cyclotron resonance frequencies and record the data to a single precision binary file.

Index