Development of Digital Mass Spectrometry Instrumentation for the Advancement of Cryogenic Ion Vibrational Spectroscopy Techniques

By

Grace O. Capek

A dissertation submitted in partial fulfillment of

the requirements for the degree of

Doctor of Philosophy

(Chemistry)

at the

UNIVERSITY OF WISCONSIN-MADISON

2025

Date of final oral examination: 12/15/2025

The dissertation is approved by the following members of the Final Oral Committee:

Etienne Garand, Professor, Chemistry

Timothy Bertram, Professor, Chemistry, Atmospheric and Oceanic Sciences

Susanna Widicus Weaver, Professor, Chemistry, Astronomy

Blaise Thompson, Instrumentation Lab Director, Chemistry

Abstract

Development of Digital Mass Spectrometry Instrumentation for the Advancement of Cryogenic Ion Vibrational Spectroscopy Techniques

Grace O. Capek

Under the supervision of Professor Etienne Garand

at the

UNIVERSITY OF WISCONSIN-MADISON

Cryogenic Ion Vibrational Spectroscopy (CIVS) is a technique which combines the sensitivity and selectivity of mass spectrometry (MS) with the direct structural information gained from infrared (IR) spectroscopy. Messenger-tagged cryogenic complexes are irradiated with an IR laser in the Reflectron time-of-flight (Re-TOF) region of a custom mass spectrometer, and the photodissociation of the ion-tag complex is monitored in order to obtain the IR spectrum. Here, we present an alternative instrument for CIVS where the complexes are irradiated inside of a cryogenic linear quadrupole ion trap driven by radiofrequency (RF) square waves, otherwise known as digital waves.

In contrast to traditional CIVS where only one mass-to-charge (m/z) ion packet can be spectroscopically probed in the Reflectron region, this instrument is equipped to perform a multiplexed spectroscopy experiment where multiple m/z species are probed in one laser scan. In the Garand group, we are particularly interested in studying solvation processes in the gas phase

by clustering amino acids and small polypeptides with water in cryogenic ion traps. The CIVS technique gives specific insight into how molecular structure changes with solvation. Expanding to a multiplexed spectroscopy workflow will allow for a higher throughput study of these water clusters, where every other m/z cluster is investigated in one experiment.

The Cryogenic Digital Linear Ion Trap Mass Spectrometer (CD-LIT-MS) described here is equipped with digital ion trapping (DIT) technology which allows the frequency to be scanned for mass analysis, fixing the amplitude of the RF square waves low to reduce the arcing potential under cryogenic conditions. The frequency and duty cycle of the RF square waves can be adjusted for software-controlled mass filtration and isolation. In addition, a novel square complex waveform has been developed to perform parametric resonant excitation. We demonstrate how this waveform can perform all the typical functions of sinusoidal complex waveforms such as selective m/z elimination or pre-ejection, mass isolation, and MS resolution enhancement. The square parametric excitation technique allows us to precisely eliminate selected m/z and therefore perform the proof-of-concept multiplexed experiment on every other water cluster.

Table of Contents

List of Figures

List of Tables

Dedication

To all the weird, inquisitive kids who are unafraid to ask how things work and why.

Acknowledgements

The gratitude that I have for the people in my life is a difficult thing to put into words, but I'm going to try and do that here anyway. I could never have achieved this PhD without the countless teachers, mentors, friends, and family who helped me to believe that it was possible. Graduate school is not particularly easy, but I can honestly say that my experience has been full of fun and meaningful moments, all because of my wonderful community. Thank you.

First, I want to thank my undergraduate research mentors at Purdue University. Thank you to Graham Cooks, who really got me hooked on mass spectrometry. Thank you for giving me numerous opportunities to get hands-on experience with research and teaching as an undergraduate. The way that you believed in me helped me to start trusting myself as a research scientist. Thank you to Brett and Kiran, my mentors in the Cooks group, who gave me ownership over my own research and helped me get into graduate school. Brett, you were right – Madison really is the best place to live.

Thank you to my PhD advisor, Etienne, for encouraging me to pursue my interests and make the PhD experience my own. I sincerely enjoyed all our conversations dreaming up instruments we could build, and all the times we tinkered around or did instrument maintenance in lab together (working in your lab has in fact solidified my identity as a lifelong tinkerer). A special thank you to Steve, Matt, and James from the UW-Madison chemistry machine shop and Blaise from the electronics shop, without whom that tinkering would not have been possible. I'm so grateful for your expertise and advice in making our instrument dreams come true.

Thank you to my PhD committee members: Tim Bertram, Gil Nathanson, Susanna Widicus Weaver, and Blaise Thompson. I would not be here without the support and recommendation letters from my committee. Tim and Gil, thank you for all your advice over the years, talking with you

both helped me to decide what career to pursue after my PhD. Thank you to Susanna and Blaise for joining my defense committee, I greatly appreciated your thoughtful questions and support.

Thank you to Brian Clowers at Washington State University, as well as Peter Reilly and his group. In this thesis, you will read a lot about square waves and digital ion trapping, none of which would have been possible without the Reilly group's circuit designs and programs. Thank you to the Reilly and Clowers groups for making your designs open source, allowing us and others to use this technology. Brian, thank you for introducing me to people in the mass spectrometry community and being so willing to help me out. I'm looking forward to seeing what comes next from the Clowers group.

Now this will be a long one, because I want to thank all my mentors and friends in the Garand group. You were so much more than colleagues. Thank you for encouraging me every day, laughing with me, sometimes crying with me on the office floor, and picking me back up again. The camaraderie and fun we had truly made grad school some of the best years of my life. Thank you to Casey, although we didn't have much overlap in the group, I appreciate how you paved the way for the work in this thesis. Summer, thank you for your energy and enthusiasm, which made me feel so welcomed when I joined the group. Cole, thank you for your excellent advice, humor, and delicious coffee (which my palette was not distinguished enough to truly appreciate).

To Gina, my mentor in the group – thank you times a million. Thank you for your boundless, infectious enthusiasm which made being your mentee such a joy. Thank you for sharing your great wealth of knowledge with me and believing in me no matter what. A special shout out to all the silly times we spent together, on the floor of our shared office or jamming to tunes in the electronics lab. Those times really got me through the first and most challenging years of grad school. Unstrumentation ladies forever. Thank you to Kathy and Katharina, who also mentored me

in many ways in and out of lab. Kathy, thank you for your sound advice and the refreshing little breaks we took to get treats. I appreciate the way you kindly challenged my assumptions and made me a better researcher. Katharina, thank you for helping me through some really tough scientific and personal times. Your tenacity is truly unmatched and continues to inspire me. Also, thank you for always telling me to calm down, I really needed that.

Thank you to Kenny, who joined the Garand group with me in fall 2020. I really appreciated your calm energy and willingness to help with anything. Your passion for science and life is so evident, and I can't wait to see the wonderful science you continue to do. Thank you to my very dear friend, Bri. You are one of the kindest people I have ever known. Thank you for always being there when I needed help, in lab and life. Thank you for allowing me to poke my head in your office anytime with questions or jokes or half-baked thoughts (often interrupting you, hard at work). Thank you for being one of my lifelines. Your quiet confidence and intelligence is going to bring you far, and I'm excited to watch you succeed at whatever you do next.

To Dillon, I'm so glad that we got to overlap for some time. I am so impressed with your confidence and how you've taken initiative in the group, already tackling so many challenges by yourself. Thank you for bringing some loudness back to the office during my last year, and please never stop sharing your strongly held beliefs. Jeffrey, your determination has truly been wonderful to see. Thank you for reminding me how exciting and awe-inspiring scientific research can be, and good luck as you continue on your (often confusing) theory journey. I'm certain you will prevail. Finally, thank you to our newest members Olivia and Tamar. It's been a very short time, but I know you both are going to do very well. I hope you enjoy the Garand group as much as I have.

Thank you to my family, who have supported all my crazy ideas, including the decision to get a chemistry PhD. To my parents, thank you for helping me to believe in myself. Dad, thank

you for showing me what hard work and perseverance looks like. I have always been inspired by you as an entrepreneur and fellow scientist. Mom, thank you for sharing your creative gifts and encouraging me to create. All the little crafts we did together may seem insignificant at first, but I believe those creative pursuits were essential to make me the person and scientist I am today. I'm so grateful that I got to grow up in a family that valued these things. Thank you to my brothers, Luke and Matthew. Thank you for fully accepting me and supporting me when I needed it the most. It was such a unique and wonderful experience being PhD students together, and I'm so grateful that we got to share the struggles, milestones, and inside jokes of grad school.

Finally, I want to thank my Madison family. I'm so grateful for the lifelong friends that I have made in grad school and in the city of Madison. Sam, thank you for being my friend from almost the very beginning. Thank you for all of our deep and insightful chats, often over a good whiskey at Cask & Ale. Thank you for grounding me, giving me no-nonsense advice, and inspiring me with your confidence. Jonathan, thank you for all the musical jams and meaningful conversations. Our open mic performances were some of the highlights of my time in Madison (shoutout to our grad school band, That Sweet Lead Vapor). Thank you for bringing so much light and music into my life, in more ways than one. Chloe, thank you for being the most supportive roommate and friend. Our home has been the warmest place, where I can always come back and relax after long days in the lab. Thank you for being the sister I never had, giving me PhD and life advice, and taking me on so many adventures (I'm sure we'll have many more). And to anyone I didn't mention, who crossed my path during this PhD experience – thank you. Without your encouragement and companionship, this achievement would not have been possible.

Chapter 1: The Past and Future of Cryogenic Ion Vibrational Spectroscopy (CIVS)

In this chapter, a combined mass spectrometry and vibrational spectroscopy technique called CIVS is introduced in section 1.1. The traditional home-built Reflectron time-of-flight mass spectrometer used to perform this technique is described in section 1.2, in comparison to a newly developed digital ion trap mass spectrometer which will be the focus of this thesis in section 1.3. The distinct functionalities of this instrument and a proposed multiplexed spectroscopy workflow are discussed in section 1.3, illustrating how innovation in CIVS instrumentation can increase throughput and make the technique more accessible for broader applications.

Section 1.1: Combining Mass Spectrometry with Vibrational Spectroscopy

Mass spectrometry (MS) is a widely used analytical technique which measures the mass-to-charge ratio (m/z) of gas-phase molecular ions. The output of this analysis is a mass spectrum, a plot of intensity vs. m/z which gives mass information for every species in a sample. Advances in MS instrumentation have made it an incredibly powerful technique for the identification of unknown species, with high specificity, sensitivity, throughput, and versatility. As a result, MS (usually combined with liquid or gas chromatography) is the gold standard for molecular analysis in industries such as drug discovery, clinical diagnostics, environmental monitoring, and forensics, among many others.[1-4] It is also a very useful technique for fundamental studies in areas such as chemical reactivity and kinetics.[5-7]

Although the high mass resolving power of MS instruments allows for precise molecular formula identification, there are limitations to the technique when it comes to elucidating

molecular structure. Molecular structure is a crucial consideration in chemical identification, since the structure of a molecule is directly related to its function, and there may exist many isomeric structures with the exact same molecular formula. In response to this limitation, MS-based methods such as tandem mass spectrometry (also known as MS/MS or MS^n) were developed to give indirect structural information. In tandem MS, molecular ions are energetically fragmented inside the mass spectrometer so that one can gain insight on connectivity and bonding from the fragments.[8-9] Ion/molecule and ion/ion reactions, often alongside tandem MS, have also been performed inside mass spectrometers in order to elucidate the presence of specific structural features such as the ordering of amino acids in a peptide, and the locations of double bonds in lipids.[5,10]

Nevertheless, there are cases where direct structural information is necessary and the results of these MS-only methods are unclear. One example is the case of distinguishing between polysaccharides, which are sugar molecules that can act as biomarkers. These molecules often have very similar formulae and bonding, leading to indistinguishable fragmentation patterns in tandem MS.[11-12] However, they are not the same – they have different structures which point to their unique origin in the body. Misidentification of one of these polysaccharides in a biological sample would lead to an incorrect identification of its bodily source, making a molecular analysis technique with direct structural information highly desirable.

One of the most common analytical techniques which *does* give direct structural information is infrared (IR) spectroscopy. IR spectroscopy is a technique which measures the absorption of IR light of a sample, yielding information about the vibrational transitions which are present and therefore the structure of its molecular components. This process is governed by Beer's Law shown in Figure 1.1, where absorbance (A) is directly proportional to concentration (c).[13] If

we want this extra dimension of information, we can ask ourselves the question: can IR absorption spectroscopy be done inside a mass spectrometer? By combining these techniques, one could have the sensitivity and controlled gas-phase environment of MS along with the direct structural information from IR spectroscopy. Unfortunately, the initial answer is no. The number of ions inside a mass spectrometer is on the order of 10^5, corresponding to concentrations of ~0.16 nM, which would be far too dilute to measure direct IR absorption.

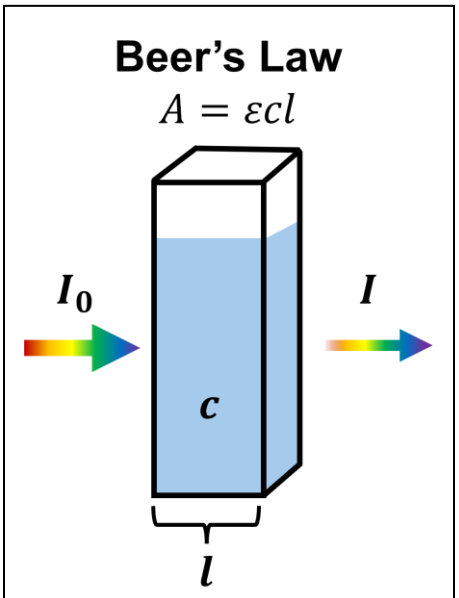


Figure 1.1 – Depiction of condensed-phase IR absorption spectroscopy, where I_0 is incident IR light, I is transmitted light, ε is molar absorptivity, c is concentration, and l is path length of the cuvette.

In the Garand group, we circumvent this problem using action spectroscopy. Instead of measuring absorption itself, we measure an *action* which is directly related to absorption. Inside a custom mass spectrometer, we measure the IR photodissociation of a messenger-tagged cryogenic complex. An inert "tag" molecule is cryogenically condensed onto the ions before irradiation with

a tunable IR laser system. When there is a resonant vibrational transition in the complex, the loosely bound tag photo-dissociates, and the tag loss is monitored in the mass spectrum. By integrating this tag loss with respect to wavelength, we get the IR spectrum.[14-15] This process is illustrated in Figure 1.2. We call this technique Cryogenic Ion Vibrational Spectroscopy (CIVS), a combined MS-IR experiment which gives both direct m/z and structural information.

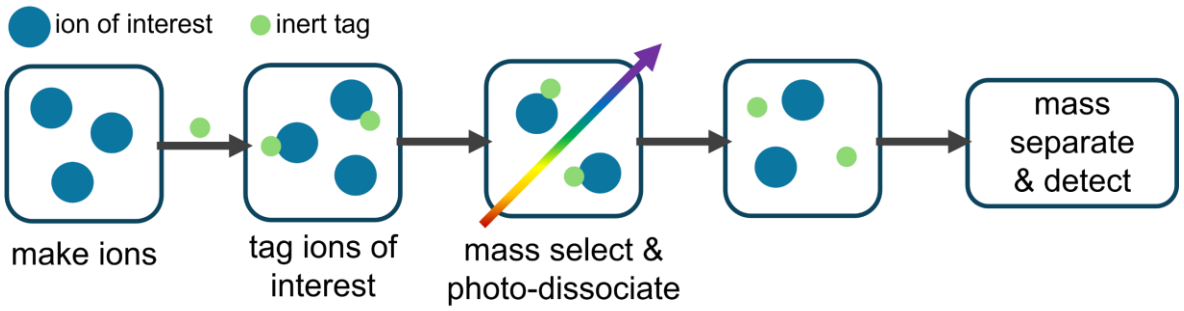


Figure 1.2 – Flow chart depicting the primary steps of the CIVS process, which occurs in the gas phase inside a custom mass spectrometer.

It is important to note that there are two variations of CIVS which are relevant to this thesis: IR photodissociation (IRPD) and IR multiple photon dissociation (IRMPD). IRPD is a single photon process which can be performed with loosely bound tags that only adsorb onto ions via van Der Waals interactions (H_2, D_2, N_2, etc.).[15] This is the most common mode of operation on our CIVS instrument. However, IRMPD is required to photo-dissociate more strongly bound tags such as water, since they form stronger interactions with the ions such as hydrogen bonds. IRMPD action spectroscopy can also be done by fragmenting the ions themselves, but this requires higher power lasers such as CO_2 discharge or free electron lasers.[12,16] Although these studies may be mentioned as relevant background, it is important to note that our specific CIVS instrument is equipped with a pulsed laser system for the messenger-tagging spectroscopy scheme.

One of the additional benefits of CIVS is the controlled gas-phase environment inside the mass spectrometer. The features in our IR spectra are sharp and well resolved since the species of interest is kept at cryogenic temperatures and isolated in the gas phase, without the broadening effects of solution. We can use ion traps in our custom MS instruments as gas-phase reaction vessels to study complex systems such as reaction intermediates or solvent clusters, with much more control than one has in the dynamic solution phase. This allows us to gain insight into fundamental processes such as solvation with a high level of precision.[15] In the next section, I will show the home-built instruments we use to do this and outline the advancements in instrumentation we have made in order to probe more complex chemistry and make this hybrid technique more accessible for further applications.

Section 1.2: Current CIVS Instrumentation and its Limitations

Our primary instrument for CIVS in the Garand group is a home-built Reflectron Time-of-Flight (ReTOF) mass spectrometer shown in Figure 1.3.[17-18] Species are first ionized via positive mode electrospray ionization (ESI) outside the instrument, then they enter a capillary-skimmer setup in Stage I. In Stage II, the ions pass through a hexapole ion guide and aperture before entering Stage III where the first ion trap is located. There are two identical linear quadrupole ion traps with cylindrical rods in Stages III and IV, which are both cooled to 77 K by home-built liquid nitrogen (LN_2) cryostats and are equipped with pulsed valves to introduce Helium buffer gas. We often refer to these as "reaction traps" since the buffer gas can be seeded with a reactant, making these traps gas-phase reaction vessels for ion/molecule chemistry or solvent clustering. These dual traps can also be driven by square or digital waves, allowing for unique mass filtration capabilities.[18] See Chapter 2 of this thesis for more information on digital ion trapping.

After passing through the reaction traps, ions enter a 3D quadrupole or Paul trap in Stage V which is cooled by a closed-cycle Helium cryostat (10 K). This is the trap where the inert "tag" molecule (H_2, D_2, N_2, etc.) is introduced for IRPD spectroscopy. Ions are collisionally cooled by a Helium buffer gas pulse seeded with the tag molecule, and the tag adsorbs onto the ions. The tagged species exit the cold Paul trap and are injected into the Stage VI TOF region, where they are separated by m/z in the drift tube. At the end of the TOF tube is a mass gate right before the Reflectron region, so that any untagged species can ideally be eliminated (like everything in life, tagging is unfortunately not 100% efficient). The complexes are then irradiated with a 10 Hz pulsed tunable IR laser system (Nd:YAG pumped OPO/OPA) and the parent ions and photofragments are separated by the Reflectron so that they can be detected. A custom LabVIEW program is used to integrate the photofragment signal with respect to wavelength to create the IRPD spectrum.

This CIVS instrument design[17-18] is based upon that by Wang and Wang[19] and the Johnson group at Yale University.[14] The instrumentation is well developed and understood, but there are some limitations to performing CIVS in these instruments which make it a less desirable technique for certain applications. First, the custom ReTOF-MS are home-built and large in size, requiring ample lab space and specialized equipment. High voltage power supplies are required for the ion optics, and expensive vacuum pumps are also needed to reach the low pressures required for the TOF region. Second, since the laser interaction region lies in the TOF, only one m/z species can be spectroscopically interrogated at a time. This severely limits the throughput of analysis, since it on average takes ~1 day to obtain a well-optimized spectrum of a single m/z species. After the spectrum is taken, there is also a significant data analysis component where calculated IR spectra must be matched to the experimental spectra to interpret the results. This process is usually done by a person, creating another bottleneck in the CIVS workflow.

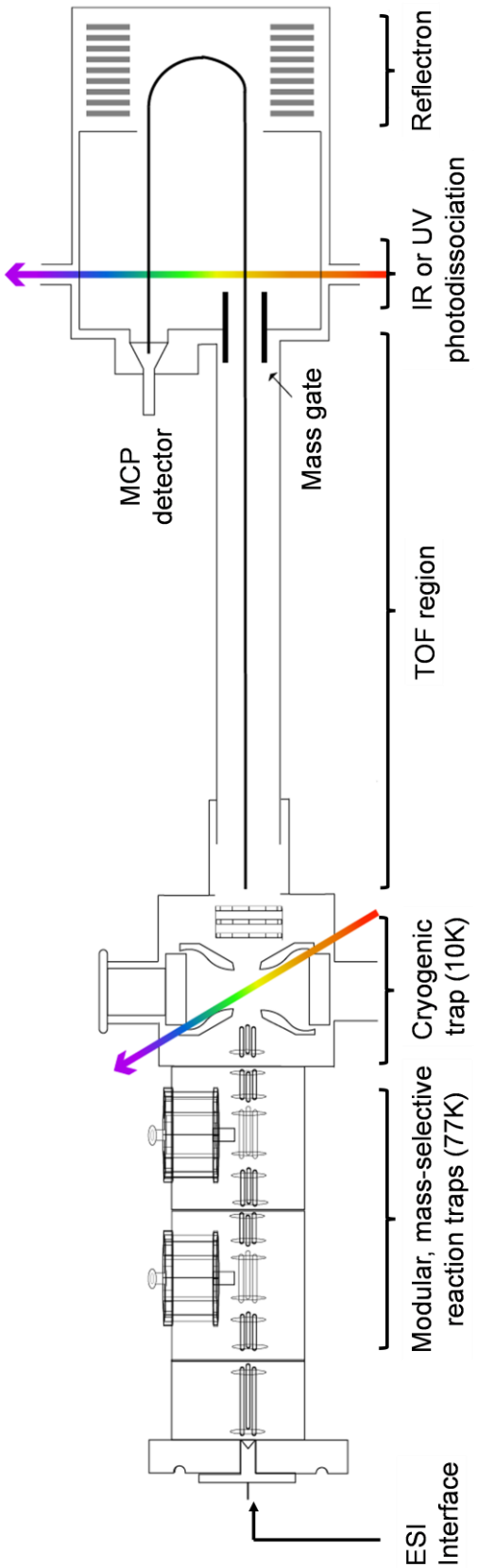


Figure 1.3 – Instrument schematic of the traditional CIVS instrument in the Garand group, a custom ReTOF-MS. Solution-phase species are ionized by a positive-mode ESI source and introduced into the instrument via a capillary-skimmer setup. There are two LN_2-cooled quadrupole reaction traps, followed by a He cryostat-cooled Paul trap for messenger tagging and a TOF drift tube with a dual stage Reflectron. The primary laser interaction region is in the Reflectron, but a second laser is available in the Paul trap for conformer-specific spectroscopy.

Significant strides have been made to develop instrumentation and methods which address these points and make CIVS more approachable for commercial applications. In order to reduce the size and cost of these instruments, cryogenic IR spectroscopy has been performed inside custom and commercial ion trap mass spectrometers. Ion trap instruments are much smaller in size and operate at higher pressures, reducing the pumping capacity needs. Nicolas Polfer's group at the University of Florida pioneered some of this work, developing a custom cryogenic rectilinear ion trap for IR action spectroscopy.[20-21] They interfaced a tunable IR laser into the ion trap, spectroscopically probing the structures of para-aminobenzoic acid (PABA) and tyramine ions cryogenically tagged with acetonitrile (ACN). Polfer also proposed that a multiplexed spectroscopy scheme should be possible in an ion trap, where multiple m/z ions are probed in one laser scan, therefore increasing throughput.[20]

Thomas Rizzo at EPFL (Ecole Polytechnique Fédérale de Lausanne) developed an instrument that combines ion mobility spectrometry (IMS) with cryogenic IR spectroscopy in a flat ion trap, with a focus on analyzing glycans.[22-23] Rizzo co-founded the company Isospec Analytics, which is working towards commercializing cryogenic infrared ion spectroscopy (CIRIS) in collaboration with Agilent Technologies.[24] More recently, Nguyen et al. demonstrated room temperature IR action spectroscopy of protonated amino acids tagged with water in a modified commercial Thermo Fisher LTQ instrument.[25]

The Garand group has been a part of this development, working to design instruments with different capabilities than the traditional ReTOF-MS and automate the CIVS data analysis workflow. We have employed digital ion trapping techniques with precise mass filtration capabilities in our instruments so that more complex mixtures can be studied without additional sample preparation. Garand group alum Dr. Gina Roesch developed the prototype which became

the digital reaction traps on CIVS, as well as a custom triple quadrupole mass spectrometer with a continuous tagging scheme to increase speed and throughput.[18,26] Another Garand alum Dr. Kathy Nickson addressed the data analysis bottleneck, developing a program to automate the assignment of calculated to experimental spectra.[27] In this thesis, I will describe my contribution to this advancement of technology for CIVS, namely, a home-built cryogenic digital linear ion trap mass spectrometer which is introduced in the next section.

Section 1.3: Cryogenic Digital Linear Ion Trap Mass Spectrometer (CD-LIT-MS)

Over the course of my PhD, I have developed an instrument which performs the CIVS experiment inside of a digital linear ion trap, opening the possibility for a multiplexed spectroscopy scheme as suggested by the Polfer group.[20] The schematic for this instrument is shown in Figure 1.4. You will notice that the front of this instrument is very similar to the ReTOF-MS from Figure 1.3. A positive-mode ESI source ionizes the sample, and the ions go through a capillary-skimmer setup in Stage I and two regions of hexapole ion guides and apertures in Stages II-III. In Stage IV there is an octupole ion trap which is cooled by a commercial LN_2 cryostat (Janis Research Co.) to 77 K. This is our "reaction trap" (RT), which is primarily utilized for forming solvent clusters. This trap was present on the original CIVS instrument, and the geometry was experimentally shown to be ideal for clustering experiments.[15,17]

Finally, the ions enter Stage V where there is a linear quadrupole ion trap with radial ejection, which is cooled by a closed-cycle Helium cryostat to 32 K. Ions enter the trap and are ejected at 90° from the trap entrance through a slit in one of the rods, where they meet a channel electron multiplier (CEM) for detection. This linear ion trap (LIT) is equipped with digital ion

trapping technology which can perform frequency and duty cycle filtering, as well as a square complex waveform for resonant excitation.[28] Chapters 2 and 3 discuss these capabilities in detail.

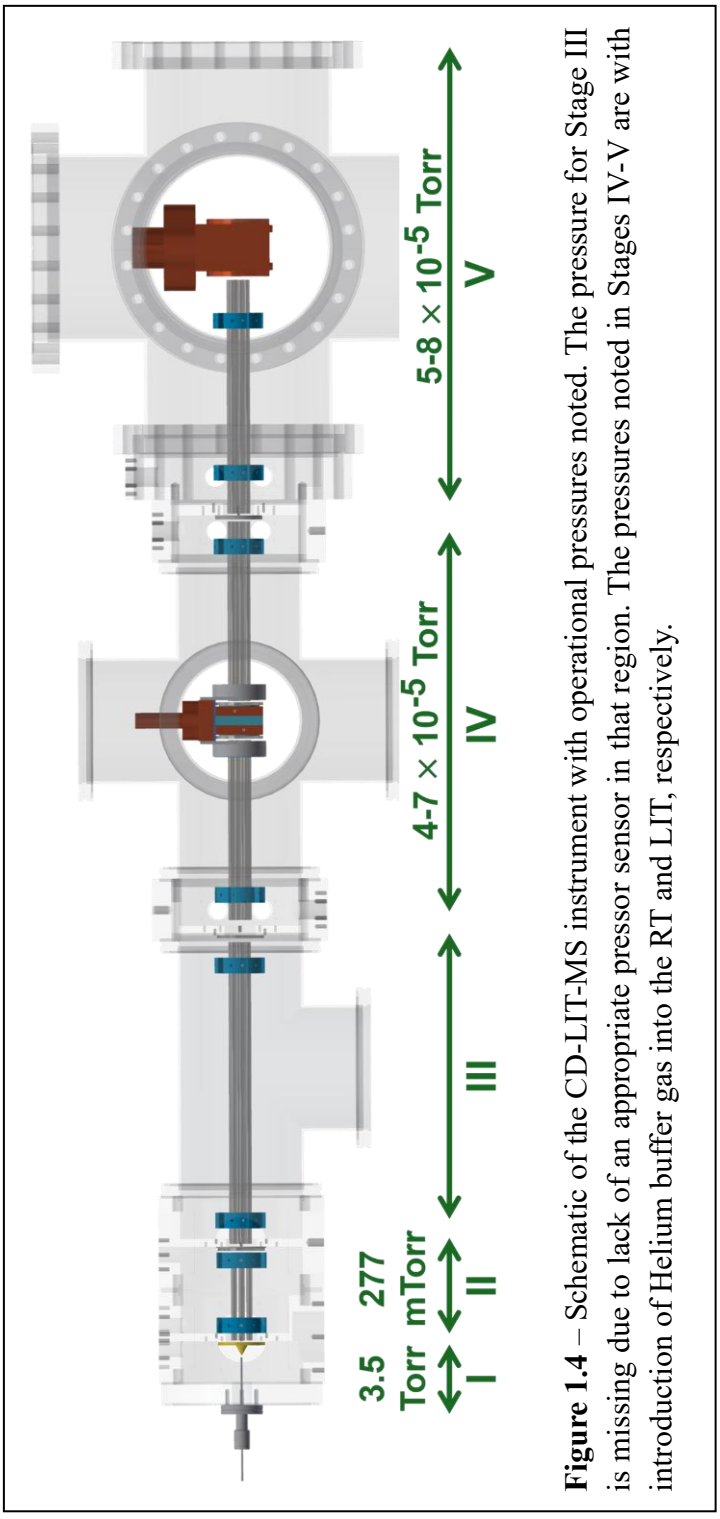


Figure 1.4 – Schematic of the CD-LIT-MS instrument with operational pressures noted. The pressure for Stage III is missing due to lack of an appropriate pressor sensor in that region. The pressures noted in Stages IV-V are with introduction of Helium buffer gas into the RT and LIT, respectively.

The instrument described here has several advantageous features which are distinct from existing CIVS instrumentation in the Garand group. First, it is a more compact instrument as demonstrated by the schematic in Figure 1.4. Replacing the ReTOF mass analyzer with the LIT cuts the overall instrument length in half and reduces the required pumping capacity. Second, irradiating the ion-tag complexes inside an ion trap allows us to probe multiple m/z species in one laser scan, increasing the throughput of the CIVS technique.

Figure 1.5 illustrates a proposed workflow for multiplexed spectroscopy in the LIT using water clusters as an example. In the Garand group, we are particularly interested in studying the solvation of amino acids and polypeptides in the gas phase as model systems for proteins. By seeding the Helium buffer gas in our LN_2-cooled ion traps, we can form cryogenic clusters of ions with water or other solvents. The gas-phase environment allows up to probe each cluster individually, gaining information about how its conformational structure changes with solvation. Understanding these structural changes is an important pursuit for biological applications, where the solvation of proteins affects their structure and function.[29]

Water clusters are an ideal proof-of-concept system for multiplexed spectroscopy in the CD-LIT-MS because we can form numerous clusters inside our ion traps.[17] As a result, spectroscopically interrogating each cluster with traditional CIVS can be a tedious and time-intensive process. A multiplexed workflow would improve on this process since every other water cluster could be probed in one scan, yielding the IR spectra of all the even or odd clusters in one experiment as shown in Figure 1.5. This scheme is an IRMPD experiment where water is used as the tag molecule. A significant portion of my PhD research focused on developing a square wave-based resonant excitation waveform to achieve Step 3 of the workflow shown in Figure 1.5; this

technique is discussed in Chapters 3-4.[28] Water clustering results and progress towards IRMPD spectroscopy are described in Chapter 6.

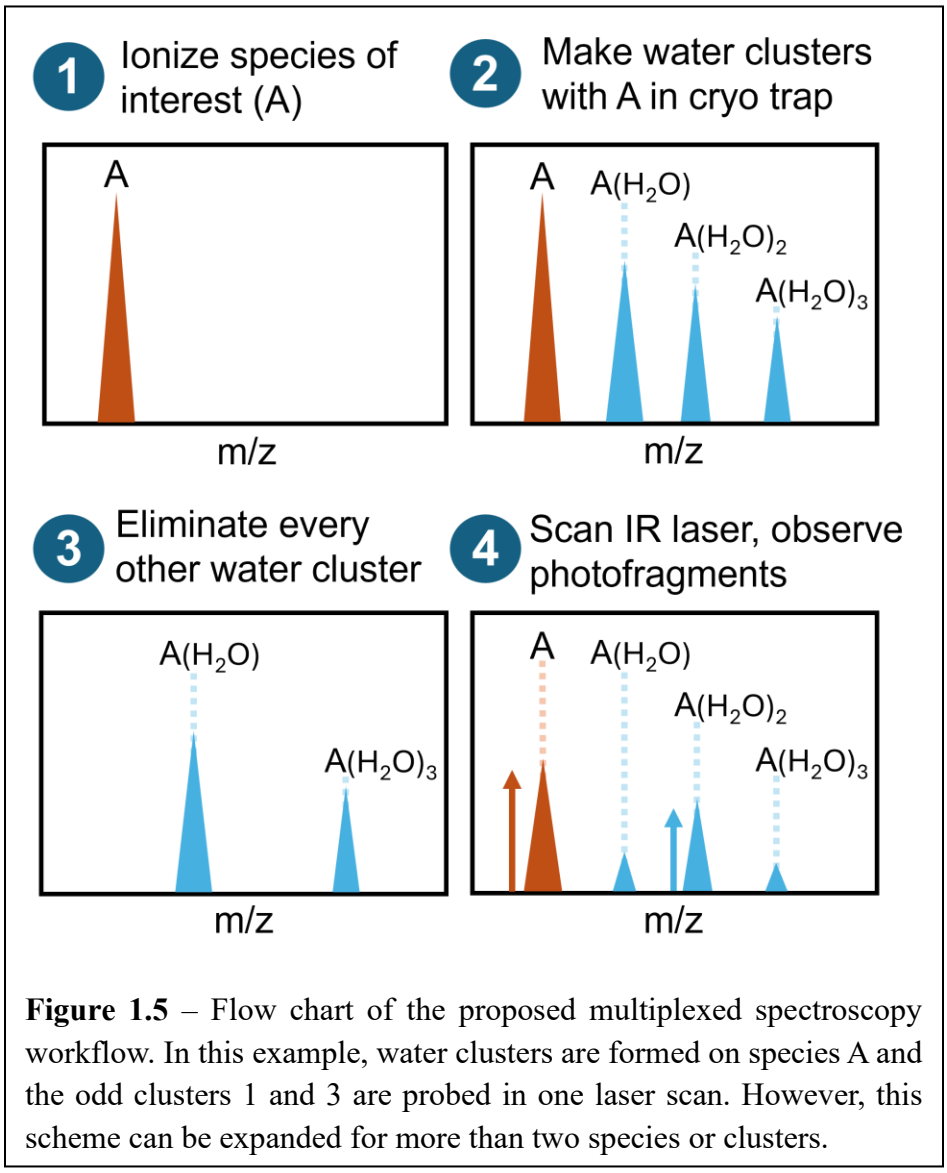


Figure 1.5 – Flow chart of the proposed multiplexed spectroscopy workflow. In this example, water clusters are formed on species A and the odd clusters 1 and 3 are probed in one laser scan. However, this scheme can be expanded for more than two species or clusters.

Finally, another advantage of the CD-LIT-MS is the digital ion trapping (DIT) technology which drives the LIT. DIT replaces traditional radiofrequency (RF) sine waves with square waves, introducing software-controlled mass filtration capabilities.[26,30-31] It also allows us to fix the amplitude low (100 V_{pp}) and scan frequency for mass analysis. This is a very important capability

under cryogenic solvent clustering conditions, where ices unavoidably form on the trap interior and introduce a higher possibility of arcing. Fixing the low amplitude of the driving waveforms allows us to reduce this arcing potential while still using an LIT as a mass analyzer. In the next chapter, I will discuss the difference between sine wave and square wave-driven quadrupole ion traps and the theory which allows us to predict the stability of ion motion.

References

(1) Glish, G. L.; Vachet, R. W. The Basics of Mass Spectrometry in the Twenty-First Century. *Nat. Rev. Drug Discov.* **2003**, *2* (2), 140–150. https://doi.org/10.1038/nrd1011.

(2) Chait, B. T. Mass Spectrometry in the Postgenomic Era. *Annu. Rev. Biochem.* **2011**, *80* (Volume 80, 2011), 239–246. https://doi.org/10.1146/annurev-biochem-110810-095744.

(3) Lebedev, A. T. Environmental Mass Spectrometry. *Annu. Rev. Anal. Chem.* **2013**, *6* (Volume 6, 2013), 163–189. https://doi.org/10.1146/annurev-anchem-062012-092604.

(4) M. Brown, H.; J. McDaniel, T.; W. Fedick, P.; C. Mulligan, C. The Current Role of Mass Spectrometry in Forensics and Future Prospects. *Anal. Methods* **2020**, *12* (32), 3974–3997. https://doi.org/10.1039/D0AY01113D.

(5) Green, M. K.; Lebrilla, C. B. Ion-Molecule Reactions as Probes of Gas-Phase Structures of Peptides and Proteins. *Mass Spectrom. Rev.* **1997**, *16* (2), 53–71. https://doi.org/10.1002/(SICI)1098-2787(1997)16:2%253C53::AID-MAS1%253E3.0.CO;2-8.

(6) Gronert, S. Mass Spectrometric Studies of Organic Ion/Molecule Reactions. *Chem. Rev.* **2001**, *101* (2), 329–360. https://doi.org/10.1021/cr9900836.

(7) Schröder, D. Applications of Electrospray Ionization Mass Spectrometry in Mechanistic Studies and Catalysis Research. *Acc. Chem. Res.* **2012**, *45* (9), 1521–1532. https://doi.org/10.1021/ar3000426.

(8) McLafferty, F. W. Tandem Mass Spectrometry. *Science* **1981**, *214* (4518), 280–287.

(9) Cooks, R. G.; Hand, O. W. Tandem Mass Spectrometry at Low Kinetic Energy. *Nucl. Instrum. Methods Phys. Res., Sect. B* **1987**, *29* (1), 427–436. https://doi.org/10.1016/0168-583X(87)90277-1.

(10) Brown, S. H. J.; Mitchell, T. W.; Blanksby, S. J. Analysis of Unsaturated Lipids by Ozone-Induced Dissociation. *Biochim. Biophys. Acta, Mol. Cell Biol. Lipids* **2011**, *1811* (11), 807–817. https://doi.org/10.1016/j.bbalip.2011.04.015.

(11) Masellis, C.; Khanal, N.; Kamrath, M. Z.; Clemmer, D. E.; Rizzo, T. R. Cryogenic Vibrational Spectroscopy Provides Unique Fingerprints for Glycan Identification. *J. Am. Soc. Mass Spectrom.* **2017**, *28* (10), 2217–2222. https://doi.org/10.1007/s13361-017-1728-6.

(12) van Outersterp, R. E.; Kooijman, P. C.; Merx, J.; Engelke, U. F. H.; Omidikia, N.; Tonneijck, M.-L. H.; Houthuijs, K. J.; Berden, G.; Peters, T. M. A.; Lefeber, D. J.; Willemsen, M. A. A. P.; Mecinovic, J.; Jansen, J. J.; Coene, K. L. M.; Wevers, R. A.; Boltje, T. J.; Oomens, J.; Martens, J. Distinguishing Oligosaccharide Isomers Using Far-Infrared Ion Spectroscopy: Identification of Biomarkers for Inborn Errors of Metabolism. *Anal. Chem.* **2023**, *95* (26), 9787–9796. https://doi.org/10.1021/acs.analchem.3c00363.

(13) *Infrared Spectroscopy*. Chemistry LibreTexts. https://chem.libretexts.org/Bookshelves/Physical_and_Theoretical_Chemistry_Textbook_Maps/Supplemental_Modules_(Physical_and_Theoretical_Chemistry)/Spectroscopy/Vibrational_Spectroscopy/Infrared_Spectroscopy/Infrared_Spectroscopy (accessed 2025-12-07).

(14) Wolk, A. B.; Leavitt, C. M.; Garand, E.; Johnson, M. A. Cryogenic Ion Chemistry and Spectroscopy. *Acc. Chem. Res.* **2014**, *47* (1), 202–210. https://doi.org/10.1021/ar400125a.

(15) Garand, E. Spectroscopy of Reactive Complexes and Solvated Clusters: A Bottom-Up Approach Using Cryogenic Ion Traps. *J. Phys. Chem. A* **2018**, *122* (32), 6479–6490. https://doi.org/10.1021/acs.jpca.8b05712.

(16) C. Polfer, N. Infrared Multiple Photon Dissociation Spectroscopy of Trapped Ions. *Chem. Soc. Rev.* **2011**, *40* (5), 2211–2221. https://doi.org/10.1039/C0CS00171F.

(17) Marsh, B. M.; Voss, J. M.; Garand, E. A Dual Cryogenic Ion Trap Spectrometer for the Formation and Characterization of Solvated Ionic Clusters. *J. Chem. Phys.* **2015**, *143* (20), 204201. https://doi.org/10.1063/1.4936360.

(18) Roesch, G. C.; Garand, E. Tandem Mass-Selective Cryogenic Digital Ion Traps for Enhanced Cluster Formation. *J. Phys. Chem. A* **2023**, *127* (36), 7665–7672. https://doi.org/10.1021/acs.jpca.3c04706.

(19) Wang, X.-B.; Wang, L.-S. Development of a Low-Temperature Photoelectron Spectroscopy Instrument Using an Electrospray Ion Source and a Cryogenically Controlled Ion Trap. *Rev. Sci. Instrum.* **2008**, *79* (7), 073108. https://doi.org/10.1063/1.2957610.

(20) Cismesia, A. P.; Tesler, L. F.; Bell, M. R.; Bailey, L. S.; Polfer, N. C. Infrared Ion Spectroscopy inside a Mass-Selective Cryogenic 2D Linear Ion Trap. *J. Mass Spectrom.* **2017**, *52* (11), 720–727. https://doi.org/10.1002/jms.3975.

(21) Tesler, L. F.; Cismesia, A. P.; Bell, M. R.; Bailey, L. S.; Polfer, N. C. Operation and Performance of a Mass-Selective Cryogenic Linear Ion Trap. *J. Am. Soc. Mass Spectrom.* **2018**, *29* (11), 2115–2124. https://doi.org/10.1007/s13361-018-2026-7.

(22) Ben Faleh, A.; Warnke, S.; Rizzo, T. R. Combining Ultrahigh-Resolution Ion-Mobility Spectrometry with Cryogenic Infrared Spectroscopy for the Analysis of Glycan Mixtures. *Anal. Chem.* **2019**, *91* (7), 4876–4882. https://doi.org/10.1021/acs.analchem.9b00659.

(23) Rizzo, T.; Warnke, S.; Ben Faleh, A.; Scutelnic, V. High-Throughput Cryogenic Spectroscopy for Glycan Analysis. US10522337B2, 2019.

(24) *Isospec Analytics*. https://www.isospecanalytics.com/news-article/isospec-and-agilent-announce-collaboration-in-molecular-identification (accessed 2025-11-30).

(25) Nguyen, T. M.; Ober, D. C.; Balaji, A.; Maiwald, F. W.; Hodyss, R. P.; Madzunkov, S. M.; Okumura, M.; Nemchick, D. J. Infrared Photodissociation Spectroscopy of Water-Tagged Ions with a Widely Tunable Quantum Cascade Laser for Planetary Science Applications. *Anal. Chem.* **2024**, *96* (22), 8875–8879. https://doi.org/10.1021/acs.analchem.4c01023.

(26) Roesch, G. C. Innovation in Mass Spectrometry Instrumentation for Applications in Cryogenic Ion Vibrational Spectroscopy. Ph. D. Dissertation, University of Wisconsin-Madison, 2023.

(27) Nickson, K. Developing Automated Spectral Assignment Programs for Cryogenic Ion Vibrational Spectroscopy. Ph. D. Dissertation, University of Wisconsin-Madison, 2024.

(28) Capek, G. O.; Howdieshell, C. J.; Garand, E. Square Parametric Excitation: A Digital Resonant Method for the Quadrupole Ion Trap. *J. Am. Soc. Mass Spectrom.* **2024**, *35* (8), 1846–1853. https://doi.org/10.1021/jasms.4c00169.

(29) Makarov, V.; Pettitt, B. M.; Feig, M. Solvation and Hydration of Proteins and Nucleic Acids: A Theoretical View of Simulation and Experiment. *Acc. Chem. Res.* **2002**, *35* (6), 376–384. https://doi.org/10.1021/ar0100273.

(30) Brancia, F. L.; McCullough, B.; Entwistle, A.; Grossmann, J. G.; Ding, L. Digital Asymmetric Waveform Isolation (DAWI) in a Digital Linear Ion Trap. *J. Am. Soc. Mass Spectrom.* **2010**, *21* (9), 1530–1533. https://doi.org/10.1016/j.jasms.2010.05.003.

(31) Singh, R.; Jayaram, V.; Reilly, P. T. A. Duty Cycle-Based Isolation in Linear Quadrupole Ion Traps. *Int. J. Mass Spectrom.* **2013**, *343–344*, 45–49. https://doi.org/10.1016/j.ijms.2013.02.012.

Chapter 2: Digital Ion Trapping (DIT) in Mass Spectrometry

This chapter serves to introduce linear quadrupole ion trap MS, starting with a brief history and description of modern quadrupole devices in section 2.1. In section 2.2, traditional quadrupole theory for sine wave-driven linear quadrupoles is described, along with details about mass-selective instability mode and a note on 3D quadrupole or Paul traps. The mathematical basis for ion motion in a linear quadrupole field based on the Mathieu equations is given in section 2.2.

In section 2.3, DIT methods are introduced where the driving sine waves are replaced with square or rectangular waveforms. The mathematical basis for ion motion in a square wave-driven quadrupole field is detailed in section 2.4 based on matrix solutions to the Mathieu-Hill equations, along with a description of a custom Python program to plot digital Mathieu stability diagrams. Finally, in section 2.5, the linear quadrupole ion trap in the CD-LIT-MS is shown and our method for digital mass analysis by frequency scan is described.

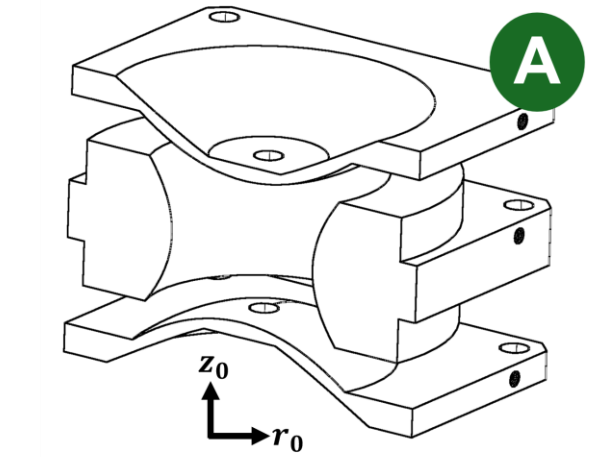
Section 2.1: A Brief History of Ion Trap Mass Spectrometry

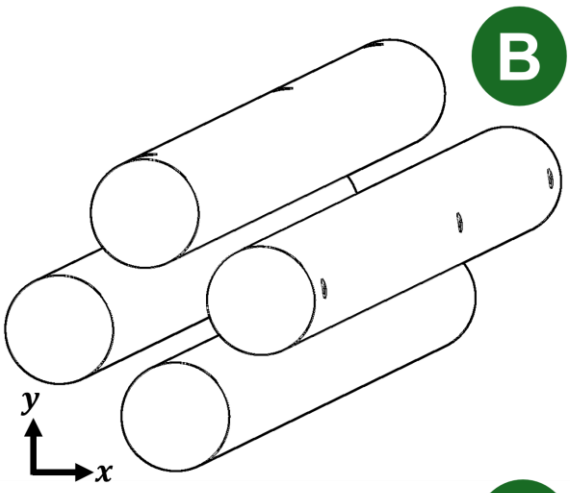
In 1897, physicist J.J. Thomson "discovered" the electron by deflecting cathode rays with a magnetic field and measuring their mass-to-charge ratio (m/z), therefore proving the existence of electrons as particles. This landmark experiment not only answered a 50-year debate about the nature of cathode rays but also laid the groundwork for the field of mass spectrometry (MS).[1] The premise of MS analysis is that one can measure the m/z of charged particles or ions by manipulating them in magnetic and electric fields. Francis Aston, a protégé of Thomson, further developed this technique to identify charged atoms and prove the existence of elemental isotopes in the early 20th century. Physicists such as Alfred Nier were pivotal in developing early MS

instrumentation, making it more accessible, and promoting the technique to fields outside of fundamental physics. One of the first chemical applications of MS was in the petroleum industry, where it was used to determine the molecular components in crude oil.[1-2]

Today, MS is one of the most powerful techniques in the modern landscape of chemical analysis, with applications in clinical diagnostics, biomedical research, natural products, pharmaceuticals, forensics, and environmental testing, to name just a few.[3-6] Although there are many types of mass spectrometry for these different applications, all MS instruments contain three key components: a source, mass analyzer, and detector. The source ionizes the sample and if it is a liquid or solid, also puts it into the gas phase. The mass analyzer then discriminates between ions of differing m/z by deflecting, separating, or manipulating the ions. The first mass analyzers were large magnets which deflected ions, and one could elucidate a relationship between the degree of deflection and m/z.[1-2] Due to the development of modern electronics and desire for more compact instrumentation, many modern mass analyzers instead generate electric fields to confine and analyze ions. The focus of this thesis is one such mass analyzer, the linear quadrupole ion trap. Finally, the mass spectrometer must have a detector after the mass analyzer to detect ions once they have been separated by m/z. This analysis results in a mass spectrum, which is a plot of intensity vs. m/z which contains peaks for all the components in the sample.

Ion trap MS instruments have become ubiquitous in mass spectrometry due to their compact nature, relatively simple operation, and the precise control of ion motion that they offer.[7-9] In the 1950s, Paul and Steinwedel invented devices now known as quadrupole mass filters and ion traps which use radiofrequency (RF) signals and direct current (DC) voltages to contain ions.[10] Figure 2.1 shows the designs of three common quadrupole devices, the 3D quadrupole ion trap or Paul trap, the quadrupole mass filter with cylindrical rods, and the linear quadrupole ion trap.





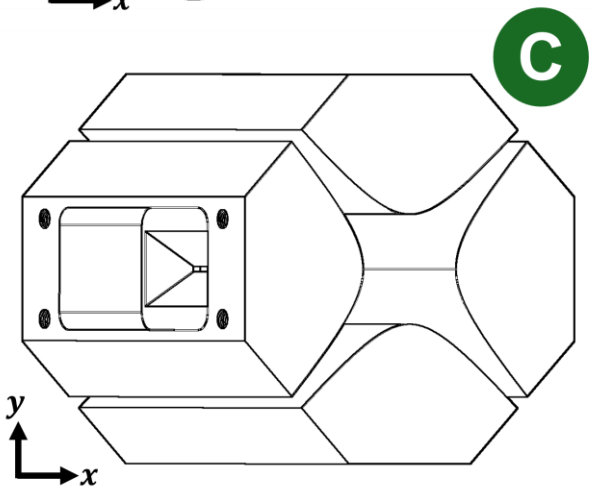


Figure 2.1 – Garand group schematics of A) 3D quadrupole or Paul trap, B) quadrupole mass filter with cylindrical rods for easier machining, C) linear quadrupole ion trap with radial ejection (endcaps are not visible).

Both quadrupole mass filters and ion traps create a quadrupole field which is described by the same theory, but the primary distinction is whether you can store ions within. A quadrupole ion trap can contain ions for a period of time, while a quadrupole mass filter allows ions to pass through continuously.[11] Since the ion trap can store ions, it can act not only as a mass analyzer but as a gas-phase reaction vessel where you can study phenomena such as ion-molecule chemistry and, in our case, cryogenic clusters.

The 3D Paul trap has one hyperbolic ring electrode with two hyperbolic end caps as shown in Figure 2.1A. A disadvantage of this geometry is that the small volume and confinement of the 3D quadrupole field limit the ion storage capacity, creating space charge repulsion which degrades its performance as a mass analyzer.[7,11] The linear quadrupole ion trap with radial ejection was developed in 1995 by Bier, Park, and Syka to increase trap volume and reduce space charge effects, in turn, enhancing injection efficiency and mass resolution.[9,12] This variation of the quadrupole trap has four hyperbolic rods similar to the quadrupole mass filter, but with endcaps or end sections on either side to which static DC voltages are applied. This creates a 2D quadrupole field in the XY plane between the rods, and a static electric field in the Z plane which keeps ions contained within the device.[12] The next section will focus on the theory and principles behind the linear quadrupole ion trap and how mass-selective radial ejection can be used to obtain a mass spectrum.

Section 2.2: Traditional Linear Quadrupole Theory

The Mathieu Equations

In the traditional linear quadrupole configuration, there are two RF sine waves applied to the X rod pair and the Y rod pair which are 180° phase shifted from each other. The phase shift is

necessary to effectively trap ions, so that they oscillate back and forth between the X and Y rods. These RF signals can also have a DC voltage component as shown in Figure 2.2. Static DC voltages on the endcaps serve to create a potential well for ions to remain trapped. However, the language "trapped" can be misleading, because it does not mean stationary. Once ions from the source enter this electric field, they experience complex trajectories of motion. These trajectories can be unstable, meaning that the ions collide with the rods and lose their charge, or stable, such that they oscillate inside the trap continuously. By adjusting the RF and DC voltages, one can make ions of specific m/z have stable or unstable trajectories.[13] This is the principle which drives mass analysis in a quadrupole ion trap or mass filter.

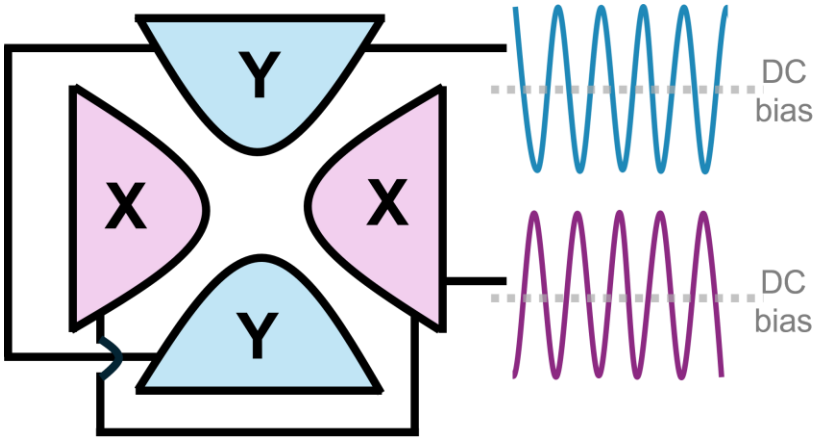


Figure 2.2 – Arrangement of traditional RF since waves on a linear quadrupole ion trap, 180° out-of-phase with each other. Optional DC bias represented by the gray dashed line.

Ion motion inside a quadrupole field can be described by solutions to a set of second-order linear differential equations known as the Mathieu equations. In 1868, Émile Mathieu derived these equations by vibrating animal skins, and described the solutions in terms of regions of stability and instability.[14] In general, the Mathieu equations can be used to describe physical phenomena that involve periodic motion. By solving these equations under the conditions of a

quadrupole field, one can determine whether an ion's trajectory will be stable or unstable. For the 2D quadrupole field in a linear ion trap, there are two Mathieu equations shown below which describe ion motion independently in the *x* and *y* dimensions.

$$\frac{d^2x}{d\xi^2} + (a_x - 2q_x cos2\xi)x = 0 \qquad (2.1a)$$

$$\frac{d^2y}{d\xi^2} + \left(a_y - 2q_y cos2\xi\right)y = 0 \qquad (2.1b)$$

Equations 1a-b contain three dimensionless parameters (a, q, and ξ), defined as

$$a_x = -a_y = \frac{8eU}{m\Omega^2 r_0^2} \qquad (2.2)$$

$$q_x = -q_y = \frac{4eV_{RF}}{m\Omega^2 r_0^2} \qquad (2.3)$$

$$\xi = \frac{\Omega t}{2} \qquad (2.4)$$

Here, e is the charge of an electron, U is the DC voltage component of the RF, m is mass, Ω is radial RF frequency, r_0 is the trap radius, V_{RF} is the RF amplitude, and t is time. The ξ parameter in Equation 2.4 is simply a different expression of time. The Mathieu a and q parameters are very important, because they allow us to interrelate the mass-to-charge ratio of an ion (notice m/e in equations 2-3) to the experimental conditions and stability of its trajectory.[9]

Plotting a vs. q results in Figure 2.3, the Mathieu stability diagram, which shows the regions where ions are stable in *x* and *y*, only stable in *x*, only stable in *y*, or unstable in both *x* and *y*. In most cases with quadrupole ion traps, the DC voltage component U is set to zero, therefore $a = 0$ and the ions are lying on the q axis.[11] The frequencies at which ions are freely oscillating (ω

in radians) are described by the following equation, where *u* represents the coordinate axis (either *x* or *y* in the case of a linear quadrupole)

$$\omega_{u,n} = (2n + \beta_u)\frac{\Omega}{2} \tag{2.5}$$

$$0 \leq \beta_u \leq 1 \qquad n = 0, \pm 1, \pm 2$$

β is a dimensionless parameter related to a and q, defined by the continued-fraction expression

$$\beta_u = a_u + \frac{q_u^2}{(\beta_u + 2)^2 - a_u - \dfrac{q_u^2}{(\beta_u + 4)^2 - a_u - \dfrac{q_u^2}{(\beta_u + 6)^2 - a_u - \cdots}}} + \frac{q_u^2}{(\beta_u - 2)^2 - a_u - \dfrac{q_u^2}{(\beta_u - 4)^2 - a_u - \dfrac{q_u^2}{(\beta_u - 6)^2 - a_u - \cdots}}} \tag{2.6}$$

Since this expression is infinite, β must be approximated in order to solve for ion motion in the sine wave-driven quadrupole field. For low values of q when $a = 0$, the Dehmelt approximation can be used to reduce β to

$$\beta_u \approx \frac{q_u}{\sqrt{2}} \tag{2.7}$$

The Mathieu β parameter defines the stability boundaries shown in Figure 2.3, also referred to as the iso-Beta lines $\beta = 0$ and $\beta = 1$.[9,13] Therefore, by using the approximation in Equation 2.7, one can map out the stability of an ion's trajectory in *a*-*q* space and relate that directly to the experimental parameters of the experiment such as RF frequency and amplitude.

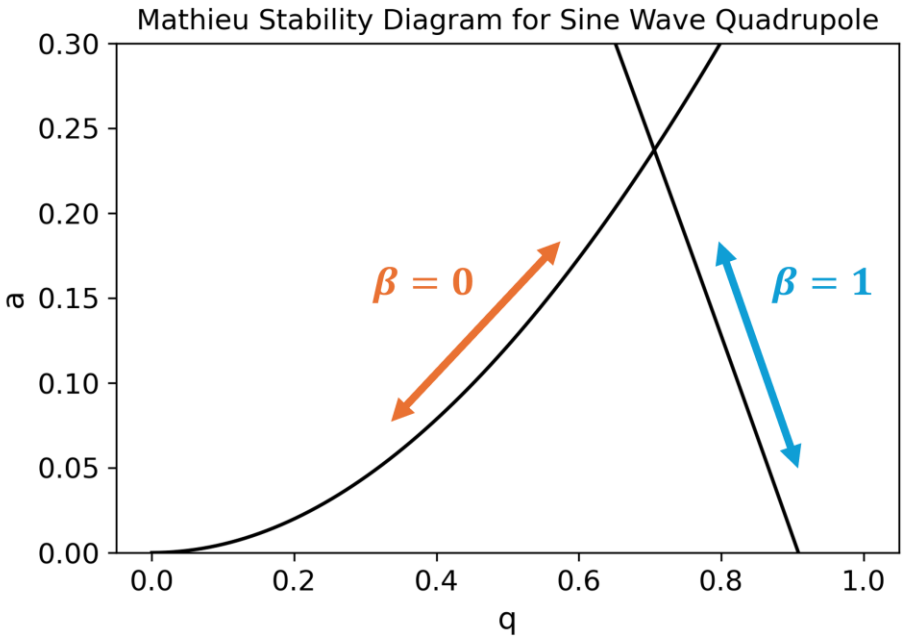


Figure 2.3 – Mathieu stability diagram for a sine wave-driven linear quadrupole, where the region of both X and Y stability is within the β stability boundaries. On the q axis, the leftmost boundary lies at $\beta = 0$ while the rightmost boundary is at $\beta = 1$. Recall that a is related to the DC voltage, while q is related to the RF amplitude of the driving waveforms.

A Note on 3D Quadrupole or Paul Traps

The equations given above are for a linear quadrupole ion trap, since that is the trap geometry in my instrument and the focus of this thesis. Nevertheless, since we also use 3D Paul traps in the Garand group, I am including a brief note here on how the Mathieu equations differ for this case. In the Paul trap, an alternating RF signal is applied to the hyperbolic ring electrode while the hyperbolic endcaps are usually grounded or floated at a constant DC potential. Unlike the 2D field of the linear ion trap, ions experience a quadrupolar potential in all three dimensions of the Paul trap.[7-9]

There are two mathematical dimensions defined in the 3D quadrupole field: the axial dimension *z* through which ions enter and exit through the endcaps, and *r*, the radial dimension in the plane of the ring electrode (see Figure 2.1A). One can solve for ion motion in the *z* and *r* dimensions independently starting with the same linear differential equations, but the *a* and *q* parameters are defined slightly differently,[15]

$$a_z = -2a_r = \frac{-16eU}{m(r_0^2 + 2z_0^2)\Omega^2} \tag{2.8}$$

$$q_z = -2q_r = \frac{-8eV_{RF}}{m(r_0^2 + 2z_0^2)\Omega^2} \tag{2.9}$$

To make a pure 3D quadrupole field, the relation $r_0^2 = 2z_0^2$ must be geometrically satisfied, simplifying these equations to the more well-known versions,

$$a_z = -2a_r = \frac{-8eU}{mr_0^2\Omega^2} \tag{2.10}$$

$$q_z = -2q_r = \frac{-4eV_{RF}}{mr_0^2\Omega^2} \tag{2.11}$$

However, geometries which deviate from the $r_0^2 = 2z_0^2$ relation are common and known as a "stretched" 3D quadrupole. Depending on the degree of the "stretch", the effects upon the Mathieu stability region may be slight.[15] The same β parameter approximation in Equation 2.7 can then be used to plot the stability diagram for the 3D quadrupole case. Therefore, especially if you are a quadrupole user in the Garand group, keep in mind these subtle differences between the *a* and *q* definitions in the linear quadrupole vs. Paul trap.

Mass-Selective Radial Instability Mode

One of the common methods of mass analysis in linear quadrupole ion traps (and the method used in the CD-LIT-MS) is called mass-selective radial instability mode. This is a form of mass analysis where one makes ions become sequentially unstable in order to mass-selectively eject them from the trap. Since $a = 0$ and the ions are lying on the q axis, scanning the RF sine wave amplitude (V_{RF}) higher moves the ions to higher q coordinates until eventually they cross the stability boundary at $\beta = 1$. Crossing the rightmost β boundary means that the ions move from a region of x and y stability to a region of x instability, so they will collide with the x rods.[9]

Exit slits are machined in one or both x rods of the linear ion trap (see Figure 2.1C) so that ions can exit and hit the detector as they become unstable.[12] After recording the ion signal hitting the detector in time, one can calculate the m/z of the ions by rearranging the Mathieu q equation (Eqn. 3) and obtain a full mass spectrum for the contents of the ion trap. Note that the term "radial" in this case represents the xy plane, while "axial" refers to the z dimension. Methods have also been developed for mass-selective *axial* ejection in linear ion traps, but that discussion is outside the scope of this thesis.[9]

Figure 2.4 shows what this process looks like on the Mathieu stability diagram, along with an illustrative metaphor taken from a tutorial article by Raymond E. March.[7] March compares the potential well inside the quadrupole ion trap to a parabolic wine glass full of liqueurs with different densities. The liqueurs arrange themselves in layers by density, and you can imagine that these layers represent different m/z ions. Scanning the RF amplitude is like slowly tipping the bowl, pouring the liqueurs out one-by-one into another vessel which represents the detector. This metaphor illustrates a few important details about the quadrupole ion trap to keep in mind, 1) similar to a bowl or test tube for liquids, ion traps are gas-phase reaction vessels, 2) ions are trapped

inside a parabolic potential well within the quadrupole field, and 3) the mass-selective radial instability scan results in sequential ejection of ions by their m/z.

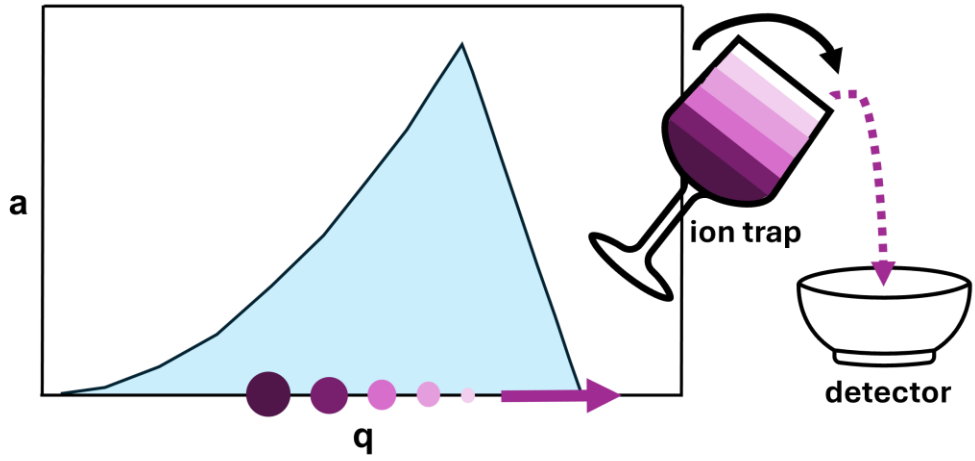


Figure 2.4 – Depiction of the mass-selective radial instability scan on the Mathieu diagram (left), liqueur metaphor from March (right).

Section 2.3: What is Digital Ion Trapping (DIT)?

In the previous section, the behavior of ions in a traditional sine wave-driven linear quadrupole was described mathematically. RF sine waves were the first electrical signal used to drive quadrupole ion traps and mass filters, but it was discovered by scientists such as Sheretov and Terent'ev and Richards et al. in the 1970s that alternate signals such as pulsed waveforms could be used and may offer some advantages.[16-17] Waveforms which are created from a series of high and low DC pulses are known as rectangular, square, or digital waves. In this thesis I refer to these waveforms as square waves for simplicity, even though they are more rectangular in shape as shown in Figure 2.5.

A key difference between these signals is that RF sine waves are variable amplitude, fixed frequency waveforms, while RF square waves are fixed amplitude, variable frequency waveforms.[16] As described in the previous section, mass analysis in a sine wave-driven quadrupole requires the amplitude to be scanned, and there is a positive linear relationship between m/z and RF amplitude (see Eqn. 2.3). Therefore, to analyze larger m/z ions, you need larger RF amplitudes. These amplitudes can get up to tens of kV, making the size of the power supply and capability of the RF circuitry the determining factor for the upper m/z limit.

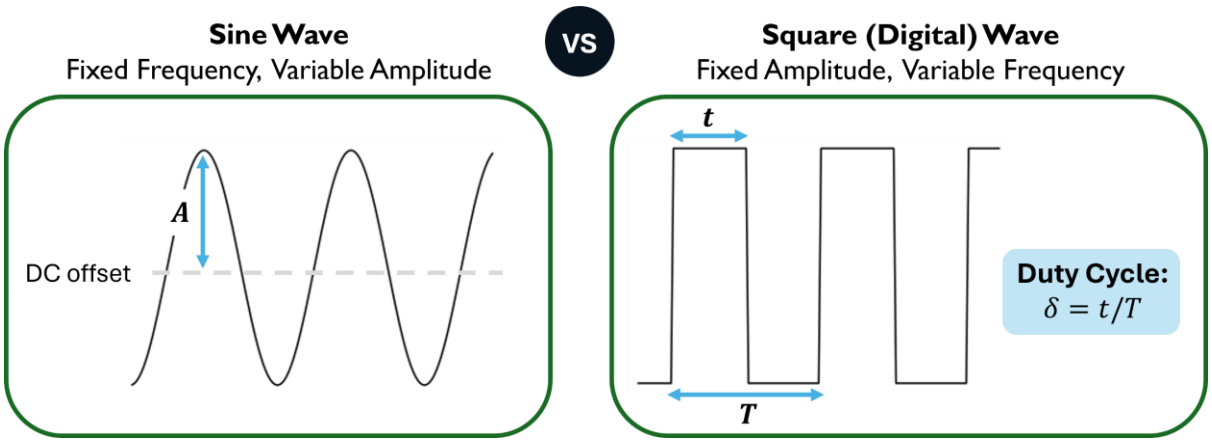


Figure 2.5 – Comparison of a sine wave with amplitude A and DC offset to a square or digital wave with duty cycle δ.

In contrast, the digital circuitry which produces RF square waves allows for facile adjustment of the frequency, since these waves consist of a series of discrete pulses in time. Square wave amplitude can be kept constant and low, while frequency is scanned for mass analysis. Due to the inverse relationship between RF frequency (Ω) and m/z defined by Mathieu q (Eqn. 2.3), lower frequencies eject higher m/z species. This means that a square wave-driven or digital ion trap has a theoretically infinite upper m/z limit since there is no lower frequency limit imposed by digital circuitry.[18] Researchers such as Li Ding and Chuan-Fan Ding developed this square wave

frequency scan in a 3D quadrupole ion trap,[18-19] resulting in the MALDImini-1, the first commercial digital mass spectrometer by Shimadzu in 2018.

Square waves have an additional parameter called the duty cycle, defined in Figure 2.5 as $\delta = t/T$, the length of time the square wave spends in the high state over the total period of the waveform. Adjustment of the square wave duty cycle affects the landscape of the quadrupole field, and in turn, changes the shape of the Mathieu stability diagram.[16,20-21] Figure 2.6 shows Mathieu *a-q* diagrams for a digital linear quadrupole with duty cycles of 50:50 and 45:55 (noted as X:Y % high), highlighting this change in shape and location of the β stability boundaries. The reasoning for this behavior will be explained in more detail in the next section. It can also be seen from comparing Figure 2.3 to 2.6 that the stability diagrams for a sine wave quadrupole and a 50:50 square wave quadrupole are not the same. Most notably, the apex and location of the $\beta = 1$ boundary are at lower q values for the square wave versus sine wave case.[16]

Adjustment of this duty cycle parameter allows for instantaneous m/z filtration or isolation, referred to as Digital Asymmetric Waveform Isolation (DAWI) by Ding and colleagues.[22] Peter Reilly's group at Washington State University further developed this technology, designing square wave generators[23] and implementing them on triple quadrupole and ion trap instruments for mass isolation, mass analysis, and tandem mass spectrometry (MS/MS).[24-26] Collision-induced dissociation (CID) in digital ion traps for MS/MS experiments has been demonstrated as an alternate method to current CID techniques, offering comparable and even enhanced fragmentation efficiency.[27-28] DIT technology has also showed promise in miniature mass spectrometers due to the simplicity and compact nature of modern digital circuitry.[17,27]

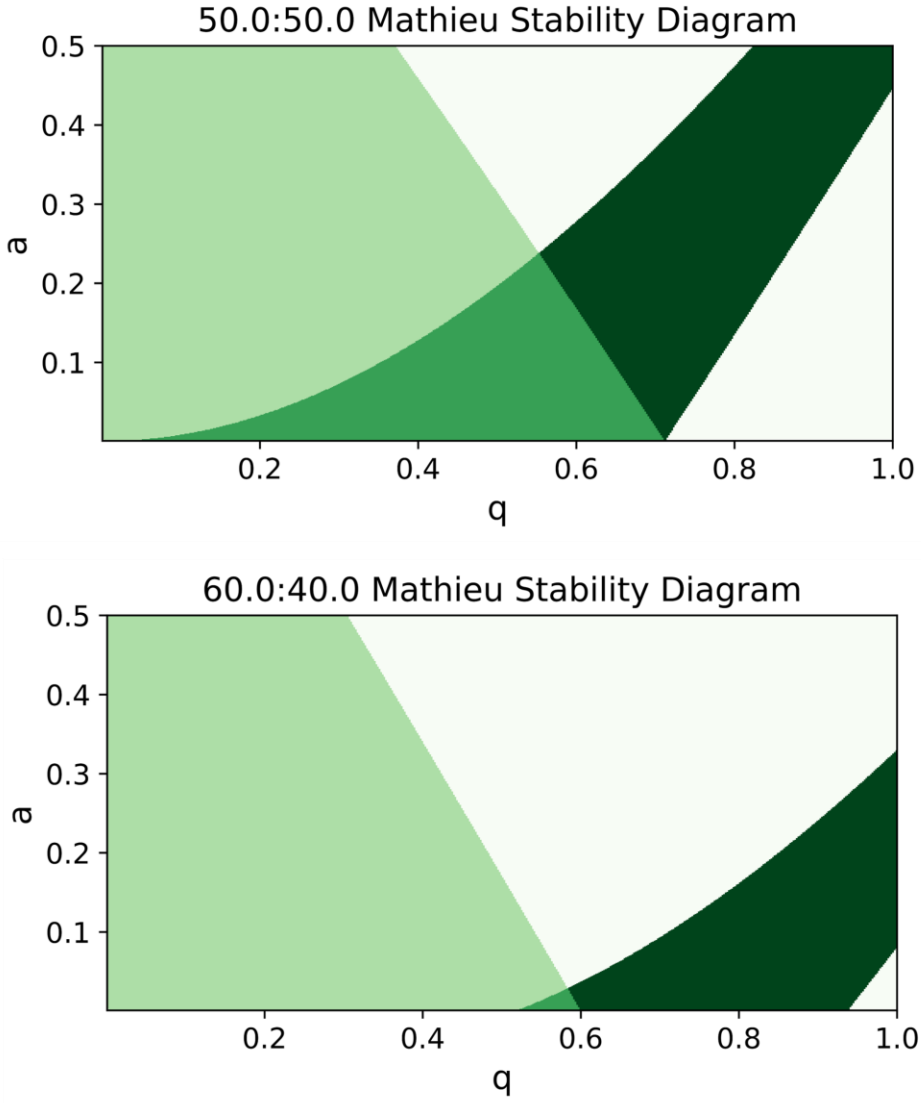


Figure 2.6 – Comparison of Mathieu stability diagrams for a digital linear quadrupole for two X:Y duty cycle combinations, 50:50 (top) and 60:40 (bottom), illustrating how the duty cycle can be used for instantaneous mass isolation.

The Reilly group has done considerable work explaining ion stability in digitally driven quadrupoles, calculating solutions to ion motion in a series of open-source spreadsheets.[29] These spreadsheets were the basis for my Python program to plot Mathieu stability diagrams, which is described in the next section. This thesis would not have been possible without help from the Reilly group, directly sharing their waveform generator designs which are currently in use on the CD-LIT-MS and paving the way for DIT technology to become more widely available. The Reilly spreadsheets illustrate another advantage of DIT: unlike sine wave-driven quadrupoles, there are analytical solutions for ion motion in a square wave-driven quadrupole.[16,29] In the next section, I will discuss these mathematical solutions in detail.

Section 2.4: DIT Theory – Matrix Solutions to the Mathieu-Hill Equations

The mathematical framework to describe ion motion in a digital quadrupole field has similarities to that described in section 2.2, as the basis is still a set of linear differential equations known as the Hill equations and the parameters a and q are still used to define ion stability.[18,29] However, the difference is that one can calculate an exact analytical solution for an ion's trajectory in a square wave-driven quadrupole field, without the approximations that are necessary in the sine wave case (see Eqns. 2.6-2.7). This is possible because square waves are discontinuous waveforms, so there are a discrete number of periods of constant potential which an ion can experience in the quadrupole field. If you solve for an ion's stability at each of these potential time points during a full period of the waveform, then you can get the complete trajectory and stability information for an ion under a set of experimental conditions.

These constant potential time points in the square wave quadrupole field are best described visually as shown in Figure 2.7. Imagine you have two square waves for a digital linear quadrupole ion trap, which are 180° phase shifted from each other. When the duty cycles of both X and Y square waves are 50:50, there are only two potential zones: X potential is high while Y is low (t_1), and X potential is low while Y is high (t_3). This changes when you adjust duty cycle, and you may end up with a period of time when X and Y are both low or both high (t_2). During this brief period of equal constant potential, there is actually no quadrupolar field present, so these two potential conditions can be mathematically considered as one time period t_2. These three voltage states sum to the total period of the waveform ($t_1 + t_2 + t_3 = T$) and the contribution of t_2 varies depending on the selected X and Y duty cycles.[29] Note how different this is from the sine wave case, where there are infinite voltage states present within one period of the continuous trapping waveforms.

Matrix solutions to Mathieu-Hill type equations were derived by Pipes in 1953 and applied to motion in a quadrupole field by Konenkov et al.[30] and later, Brabeck & Reilly.[29] Since they are in use again, here are the Mathieu *a* and *q* parameters for a linear quadrupole introduced in Section 2.2 as Equations 2.2 - 2.3:

$$a = \frac{8zeU}{m\Omega^2 r_0^2} \qquad q = \frac{4zeV_{RF}}{m\Omega^2 r_0^2}$$

Note that in this case, the DC voltage component *U* is defined as the DC voltage difference between the X and Y rods, $\left(V_X^{high} + V_X^{low}\right) - \left(V_Y^{high} + V_Y^{low}\right)$ and V_{RF} is defined as the zero-to-peak voltage of the square waves. The other variables are as follows: *z* is the charge state of the ion, *e* is electronic charge in Coulombs, *m* is the mass of ion in kilograms, Ω is the angular RF frequency in radians, and r_0 is the inner radius of the quadrupole in meters. Figure 2.7 shows the amplitude

of the square waves for a linear quadrupole defined in terms of a and q, which will be used in the following matrix solutions.[29]

Since voltage is constant during the time segments t_1, t_2, and t_3, you can calculate the position and velocity of an ion for each time segment. This is all the information you need to determine whether an ion's trajectory is stable or unstable, since the time segments encompass the whole period T and it repeats continuously. Each time segment can be defined by the following matrices in terms of the time parameter $\tau = \pi t/T$ and the parameter f

$$V(f,\tau) = \begin{bmatrix} \cos(\tau\sqrt{f}) & \frac{1}{\sqrt{f}}\sin(\tau\sqrt{f}) \\ -\sqrt{f}\sin(\tau\sqrt{f}) & \cos(\tau\sqrt{f}) \end{bmatrix} if\ f > 0 \tag{2.12}$$

or

$$V(f,\tau) = \begin{bmatrix} \cosh(\tau\sqrt{-f}) & \frac{1}{\sqrt{-f}}\sinh(\tau\sqrt{-f}) \\ \sqrt{-f}\sinh(\tau\sqrt{-f}) & \cosh(\tau\sqrt{-f}) \end{bmatrix} if\ f < 0 \tag{2.13}$$

where the f values are defined in terms of Mathieu a and q as shown in Figure 2.7 for the three time segments,

$$f_1 = a + 2q \tag{2.14}$$

$$f_2 = a \tag{2.15}$$

$$f_3 = a - 2q \tag{2.16}$$

The product of all three matrices then gives you the matrix M, which defines ion stability over the entire period.

$$M = V(f_1,t_1) \times V(f_2,t_2) \times V(f_3,t_3) = \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} \tag{2.17}$$

This matrix *M* is commonly known in ion optics as the transfer matrix, although it may also be referred to as the monodromy or transmission matrix. *M* defines the global stability of the solutions to the Hill equations, and therefore, the global stability of ions in the quadrupole field.[30] The trace of *M* is defined as

$$tr(M) = m_{11} + m_{22} \tag{2.18}$$

and the stability parameter β depends on the trace of *M* as follows:

$$\beta = \frac{1}{\pi}\arccos\left[\frac{tr(M)}{2}\right] \tag{2.19}$$

When $|tr(M)| > 2$, Equation 2.19 produces a real number for β which means that ion motion is stable. When $|tr(M)| < 2$, Equation 2.19 produces an imaginary number for β which means that ion motion is unstable.[29-30] Therefore, by calculating the matrix *M* and taking its trace, you can determine whether an ion is stable or unstable at a specific a, q coordinate. To determine the full stability landscape and plot the Mathieu diagram for a digital quadrupole, you must create a grid of a vs. q values and solve for *tr(M)* at each point.[29] Remember that ion motion is independent in the X and Y dimensions of a linear quadrupole, so there will be a transfer matrix for both X and Y. This results in the four possible stability solutions: unstable in Y but stable in X, stable in both X and Y, unstable in X but stable in Y, and unstable in both X and Y.

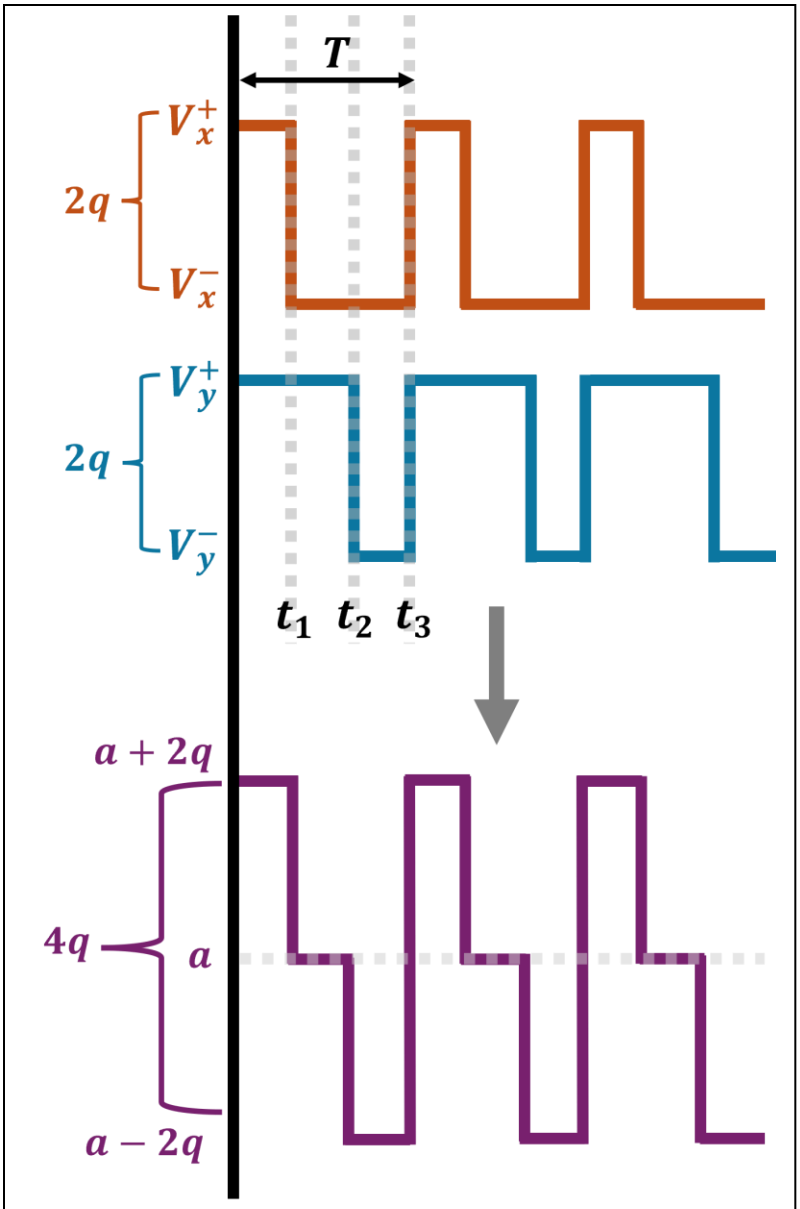


Figure 2.7 – Representation of the three potential states (t_1, t_2, t_3) in the square wave linear quadrupole field. The orange trace represents the X square wave, while the blue trace represents the Y square wave, both with non-50:50 duty cycles to introduce a t_2 period. The purple trace represents the combined potential of X and Y in terms of the Mathieu stability parameters *a* and *q*, which are input into the matrix solutions as *f* values.

Python Program for Digital Quad Mathieu Stability Diagrams

I wrote a Python program based on the Reilly spreadsheets to calculate the matrix solutions to the Mathieu-Hill equations as described above and plot the Mathieu stability diagram for a selected *a* and *q* range (file name: *DigitalQuad_MathieuStabilityDiagrams.py*). Screenshots of this program can be found in Appendix section A1. Here I will briefly walk through the steps of the calculation in this code. The user input variables are t1, t3, and numpy arrays of a and q values. The program will take the input t1 and t3, defined as the decimal fraction of t/T and then will calculate t2/T from that user input. Recall that t1/T and t3/T are the high duty cycles for the X and Y rods, respectively, as shown in Figure 2.7. The primary use of this code is to visualize the resulting Mathieu stability diagram when you change duty cycle settings.

Next, a series of functions are defined. The functions "transM_pos" and "transM_neg" correspond to the two V(f,τ) matrices in Equations 2.12-2.13, so the inputs are the time parameter tau (calculated from t1, t2, and t3) and the f value. Then two more functions called "X_trace" and "Y_trace" are defined which perform the rest of the calculation for the X and Y rods separately. In both these functions you see the f values defined from Equations 2.14-2.16 and a series of if/else statements which determine whether you should calculate V(f,τ) for the positive or negative f value case. Finally, these functions multiply the matrices for t1, t2, and t3 together and take the trace of the final transfer matrix as shown in Equations 2.17 and 2.18. The "X_trace" and "Y_trace" functions return tr(M), the trace of the transfer matrix, whose value determines ion stability.

In the last two sections of the code, matrices are initialized for the X and Y tr(M) values. There is a "for loop" which iterates through the a,q matrix (created from the user input a and q arrays) to calculate those traces at each coordinate. A grid is then initialized to contain the results of the calculations. Then there is another "for loop" which assigns the three potential solutions to

an integer: X stable Y unstable assigned to 1, X stable Y stable assigned to 2, and X unstable Y stable assigned to 3. The fourth option (X unstable Y unstable) is not assigned an integer, because it can simply not be plotted and the white space of the plot will correspond to that region. Finally, the grid which contains these integers (solution 1, 2, or 3) is plotted using the "matplotlib.pyplot" library "imshow" function. This function will plot the grid using a selected color palette (currently set to "Greens" but you can change this to match your aesthetic).

Keep in mind that the code will take longer to run depending on the size of the a and q arrays that you input. The code defaults to only a 100 × 100 matrix which will not take long to calculate but will look pixelated. To get a nice image of the Mathieu diagram with clear Beta lines, you will need arrays of ~500 or longer. In my experience, a 500 × 500 matrix can be calculated and plotted in ~15 seconds using the original version of this code in the Spyder IDE. There is another variation of this program available in a Jupyter notebook file entitled "DQ_Stability_Diagrams_v2" which was improved by Brian Clowers to run faster and has the capability to do an animation of the stability diagram as duty cycle is scanned. This version of the program is certainly more evolved and available to you if you would like to use Jupyter.

Section 2.5: The Digital Linear Ion Trap in the CD-LIT-MS

In the final section of this chapter, I will briefly introduce the home-built digital linear ion trap which is present in the CD-LIT-MS and explain how the mass-selective radial instability scan is set up. Throughout this thesis, I refer to this trap simply as the Linear Ion Trap or LIT. Note this trap was originally designed by Garand group alum Dr. Casey Howdieshell.[31] An Inventor CAD schematic and picture of the LIT cross section is shown in Figure 2.8, with the endcaps of the trap

removed to reveal the interior. Based upon previous linear quadrupole ion trap designs such as that by Schwartz et al.,[12] the LIT is 38 mm long in the axial Z dimension and has a traditional hyperbolic cross-section with a circumscribed inner radius (r_0) of 5 mm. There is one 18 × 0.5 mm exit slit machined in one of the X-dimension rods for radial ejection, with a cavity also machined into the exit side to allow for extraction electrodes to be placed directly after the slit. The missing endcap electrodes are symmetrical, each with a 2.5 mm diameter aperture which allows ions to enter through the front cap and to interface an IR laser through the back cap.

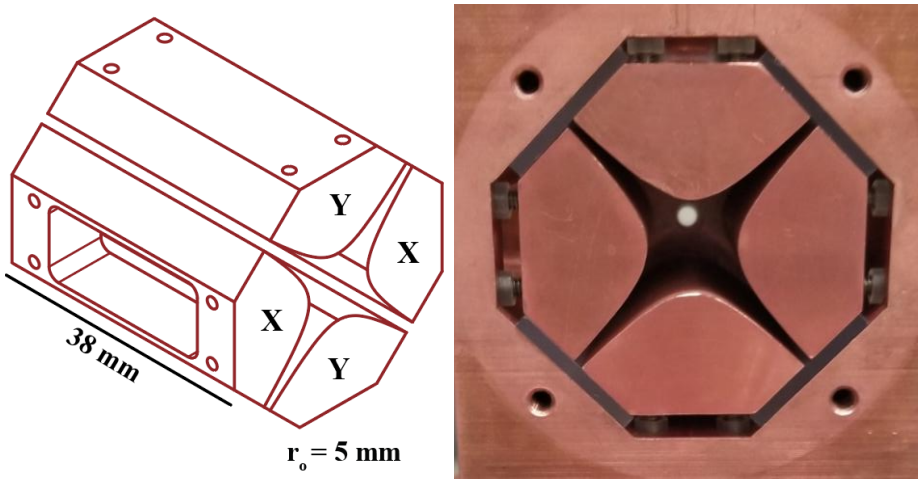


Figure 2.8 – Schematic of the LIT from Autodesk Inventor (left) alongside a picture of the hyperbolic cross section of this linear quadrupole ion trap (right).

The LIT uses mass-selective radial instability mode as the method for mass analysis. As described in Section 2.3, in digital quadrupole devices the square wave RF frequency is scanned simultaneously on the X and Y rods in order to sequentially eject ions by their m/z. Since there is a non-linear relationship between m/z and RF frequency (see Equation 2.3), a linear scan of frequency would create a non-linear mass calibration, which is not ideal. For this reason, we employ a reverse stepwise scan of the square wave frequency as previously reported for digital ion traps.[19,26] Each frequency step is calculated from the rearranged Mathieu *q* equation from a user

input *m/z* range and step size. The frequency is stepped from high to low, which corresponds to the reverse trend in *m/z*. Details about the square wave control programs and data analysis for this process are given in Chapter 5. In addition to square wave RF frequency and duty cycle manipulation, the LIT is equipped with a square auxiliary waveform for resonant excitation which will be described in the next chapter.

References

(1) Griffiths, J. A Brief History of Mass Spectrometry. *Anal. Chem.* **2008**, *80* (15), 5678–5683. https://doi.org/10.1021/ac8013065.

(2) Smoluch, M.; Silberring, J. A Brief History of Mass Spectrometry. *Mass Spectrometry;* Smoluch, M., Grasso, G., Suder, P., Silberring, J., Eds.; John Wiley & Sons, Inc., 2019; pp 5-8. https://doi.org/10.1002/9781119377368.ch2

(3) Glish, G. L.; Vachet, R. W. The Basics of Mass Spectrometry in the Twenty-First Century. *Nat. Rev. Drug Discov.* **2003**, *2* (2), 140–150. https://doi.org/10.1038/nrd1011.

(4) Chait, B. T. Mass Spectrometry in the Postgenomic Era. *Annu. Rev. Biochem.* **2011**, *80* (Volume 80, 2011), 239–246. https://doi.org/10.1146/annurev-biochem-110810-095744.

(5) Lebedev, A. T. Environmental Mass Spectrometry. *Annu. Rev. Anal. Chem.* **2013**, *6* (Volume 6, 2013), 163–189. https://doi.org/10.1146/annurev-anchem-062012-092604.

(6) M. Brown, H.; J. McDaniel, T.; W. Fedick, P.; C. Mulligan, C. The Current Role of Mass Spectrometry in Forensics and Future Prospects. *Anal. Methods* **2020**, *12* (32), 3974–3997. https://doi.org/10.1039/D0AY01113D.

(7) March, R. E. An Introduction to Quadrupole Ion Trap Mass Spectrometry. *J. Mass Spectrom.* **1997**, *32* (4), 351–369. https://doi.org/10.1002/(SICI)1096-9888(199704)32:4%253C351::AID-JMS512%253E3.0.CO;2-Y.

(8) Wong, P. S. H.; Cooks, R. G. Ion trap mass spectrometry. *Curr. Sep.* **1997**, *16*, 85-92.

(9) Douglas, D. J.; Frank, A. J.; Mao, D. Linear Ion Traps in Mass Spectrometry. *Mass Spectrom. Rev.* **2005**, *24* (1), 1–29. https://doi.org/10.1002/mas.20004.

(10) Paul, W.; Steinwedel, H. Apparatus for Separating Charged Particles of Different Specific Charges. US2939952A, 1960.

(11) March, R. E. Quadrupole Ion Traps. *Mass Spectrom. Rev.* **2009**, *28* (6), 961–989. https://doi.org/10.1002/mas.20250.

(12) Schwartz, J. C.; Senko, M. W.; Syka, J. E. P. A Two-Dimensional Quadrupole Ion Trap Mass Spectrometer. *J. Am. Soc. Mass Spectrom.* **2002**, *13* (6), 659–669. https://doi.org/10.1016/S1044-0305(02)00384-7.

(13) March, R. E.; Todd, J. F. J. Quadrupole Ion Trap Mass Spectrometry, 2005.

(14) Campana, J.E. Elementary Theory of the Quadrupole Mass Filter. *Int. J. Mass Spectrom. Ion Phys.* **1980**, *33,* 101-117.

(15) Johnson, J. V.; Peddeer, R. E.; Yost, R. A.; March, R. E. The Stretched Quadrupole Ion Trap: Implications for the Mathieu Au and Qu Parameters and Experimental Mapping of the Stability Diagram. *Rapid Commun. Mass Spectrom.* **1992**, *6* (12), 760–764. https://doi.org/10.1002/rcm.1290061210.

(16) Richards, J. A.; Huey, R. M.; Hiller, J. A New Operating Mode for the Quadrupole Mass Filter. *Int. J. Mass Spectrom. Ion Phys.* **1973**, *12,* 317-339.

(17) Sheretov, E. P.; Gurov, V. S.; Safonov, M. P.; Philippov, I. W. Hyperboloid Mass Spectrometers for Space Exploration. *Int. J. Mass Spectrom.* **1999**, *189* (1), 9–17. https://doi.org/10.1016/S1387-3806(99)00041-X.

(18) Ding, L.; Sudakov, M.; Kumashiro, S. A Simulation Study of the Digital Ion Trap Mass Spectrometer. *Int. J. Mass Spectrom.* **2002**, *221* (2), 117–138. https://doi.org/10.1016/S1387-3806(02)00921-1.

(19) Ding, L.; Sudakov, M.; Brancia, F. L.; Giles, R.; Kumashiro, S. A Digital Ion Trap Mass Spectrometer Coupled with Atmospheric Pressure Ion Sources. *J. Mass Spectrom.* **2004**, *39* (5), 471–484. https://doi.org/10.1002/jms.637.

(20) Brabeck, G. F.; Reilly, P. T. A. Mapping Ion Stability in Digitally Driven Ion Traps and Guides. *Int. J. Mass Spectrom.* **2014**, *364*, 1–8. https://doi.org/10.1016/j.ijms.2014.03.008.

(21) Berton, A.; Traldi, P.; Ding, L.; Brancia, F. L. Mapping the Stability Diagram of a Digital Ion Trap (DIT) Mass Spectrometer Varying the Duty Cycle of the Trapping Rectangular Waveform. *J. Am. Soc. Mass Spectrom.* **2008**, *19* (4), 620–625. https://doi.org/10.1016/j.jasms.2007.12.012.

(22) Brancia, F. L.; McCullough, B.; Entwistle, A.; Grossmann, J. G.; Ding, L. Digital Asymmetric Waveform Isolation (DAWI) in a Digital Linear Ion Trap. *J. Am. Soc. Mass Spectrom.* **2010**, *21* (9), 1530–1533. https://doi.org/10.1016/j.jasms.2010.05.003.

(23) Hoffman, N. M.; Gotlib, Z. P.; Opačić, B.; Clowers, B. H.; Reilly, P. T. A. A Comparison Based Digital Waveform Generator for High Resolution Duty Cycle. *Rev. Sci. Instrum.* **2018**, *89* (8), 084101. https://doi.org/10.1063/1.5004798.

(24) Singh, R.; Jayaram, V.; Reilly, P. T. A. Duty Cycle-Based Isolation in Linear Quadrupole Ion Traps. *Int. J. Mass Spectrom.* **2013**, *343–344*, 45–49. https://doi.org/10.1016/j.ijms.2013.02.012.

(25) Brabeck, G. F.; Chen, H.; Hoffman, N. M.; Wang, L.; Reilly, P. T. A. Development of MSn in Digitally Operated Linear Ion Guides. *Anal. Chem.* **2014**, *86* (15), 7757–7763. https://doi.org/10.1021/ac501685v.

(26) Reece, M. E.; Huntley, A. P.; Moon, A. M.; Reilly, P. T. A. Digital Mass Analysis in a Linear Ion Trap without Auxiliary Waveforms. *J. Am. Soc. Mass Spectrom.* **2020**, *31* (1), 103–108. https://doi.org/10.1021/jasms.9b00012.

(27) Xue, B.; Sun, L.; Huang, Z.; Gao, W.; Fan, R.; Cheng, P.; Ding, L.; Ma, L.; Zhou, Z. A Hand-Portable Digital Linear Ion Trap Mass Spectrometer. *Analyst* **2016**, *141* (19), 5535–5542. https://doi.org/10.1039/C6AN01118G.

(28) Xu, F.; Wang, W.; Ding, L.; Fang, X.; Ding, C.-F. Synchronized Reverse Scan Collision Induced Dissociation in Digital Ion Trap Mass Spectrometer for Improving Fragment Ion Detection. *Anal. Chem.* **2022**, *94* (51), 17827–17834. https://doi.org/10.1021/acs.analchem.2c03524.

(29) Brabeck, G. F.; Reilly, P. T. A. Mapping Ion Stability in Digitally Driven Ion Traps and Guides. *Int. J. Mass Spectrom.* **2014**, *364*, 1–8. https://doi.org/10.1016/j.ijms.2014.03.008.

(30) Konenkov, N. V.; Sudakov, M.; Douglas, D. J. Matrix Methods for the Calculation of Stability Diagrams in Quadrupole Mass Spectrometry. *J Am. Soc. Mass Spectrom.* **2002**, *13* (6), 597–613. https://doi.org/10.1016/S1044-0305(02)00365-3.

(31) Howdieshell, C. J. Linear Digital Ion Trap Mass Spectrometry and its Application for Cryogenic Ion Spectroscopy. Ph.D. Dissertation, University of Wisconsin-Madison, 2021.

Chapter 3: Resonance Methods in Quadrupole Ion Traps

In this chapter, resonance methods for mass spectrometry are introduced since they are utilized in the LIT. Section 3.1 gives a general definition of resonance and introduces resonances of ion motion within a quadrupole ion trap. The equations which define resonance in a quadrupole field are also given in section 3.1. Section 3.2 shows the typical arrangements of alternating current (AC) signals for resonance methods, referred to as dipolar and quadrupolar/parametric excitation. The configurations of these signals are shown for both Paul traps and linear quadrupole ion traps or mass filters. Finally, resonance methods in digital ion traps are discussed in section 3.3, and a novel square parametric excitation technique in the LIT is described along with a corresponding Python program which calculates parametric resonant frequencies from m/z.

Section 3.1: What are Resonance Methods?

Resonances are a commonly discussed topic in physics and physical chemistry, because they are a part of many processes which involve periodic, wave-like motion. When an object oscillates in space and time, it does so at a specific frequency or set of frequencies depending on the complexity of the motion. If you strike that object with an oscillating signal (light, sound, electromagnetic pulse, etc.) of a *matching* frequency, that is called resonance. Resonance causes the object to oscillate or vibrate at a greater amplitude, which is often referred to as excitation.[1]

One of this author's favorite examples of resonance is the humble playground swing. Imagine your best friend, enjoying the pendulum action of the swing on their local playground. On their own, they are oscillating at a specific frequency. Resonance is what occurs when you, the external resonant signal, apply a time-dependent pushing force onto your swinging friend. This

resonant action causes your friend's oscillations to increase in amplitude, and in turn, hopefully increases their enjoyment. Keep in mind, however, that resonance can lead to instability. With continued force, your friend's amplitude may increase enough such that they eject from the swing. This pendulum or swinging motion is an example of parametric resonance, a term which will become relevant later in this chapter.[2]

In Chapter 2, quadrupole ion traps and the periodic trajectories that ions experience within these devices were introduced. Ions oscillate inside the quadrupole field at a fundamental frequency known as the *secular frequency*, but there are actually mathematically infinite, discrete resonant frequencies of ion motion in the quadrupole field.[3] Excitation of these resonances is used for many mass-selective operations in quadrupole ion traps, such as ion isolation, fragmentation, and ejection. These techniques which utilize resonant excitation are generally referred to as resonance methods.[4]

Resonant frequencies in quadrupole ion traps can be calculated using the Mathieu-Hill equations which were introduced in Chapter 2. There is a meaningful relationship between m/z and resonant frequency, allowing these techniques to be mass selective and highly specific to the m/z ion of interest.[5-6] Resonance methods are widely used in ion trap mass spectrometry due to the specificity and precise control over ion motion which they offer. Resonant excitation of one m/z can be done with little to no perturbation of other m/z ions which are present.[4]

The fundamental secular frequency of ions in a quadrupole field (ω) is defined by the following equation, where u represents the coordinate axis (x, y, or z), β is the Mathieu stability parameter, and Ω is the RF trapping frequency.

$$\omega_{u,0} = \frac{\beta_u \Omega}{2} \tag{3.1}$$

Recall from Chapter 2 that the β parameter ($0 \leq \beta_u \leq 1$) is dependent on Mathieu q, and therefore dependent on m/z, RF amplitude, frequency, and trap radius. Therefore, from Equation 3.1, secular frequency is inversely proportional to m/z. It is also important to note that β is defined differently for sine wave and square wave-driven quadrupoles (see Chapter 2 equations 2.7 and 2.15), making the resonant frequencies dependent on the *type* of trapping waveform along with the experimental parameters. While the secular frequency is the fundamental component ion motion, ions also have *parametric* resonances,

$$\omega_{u,n}^{K} = \frac{\Omega}{K}|n + \beta_u| \tag{3.2}$$

$$n = 0, \pm 1, \pm 2, \ldots \quad K = 1, 2, 3, \ldots$$

where K is the order of the resonance and n is an integer.[7] Equation 3.2 shows that there are infinite m/z-specific resonances available to excite ions. However, the strongest parametric resonance is the first order resonance at $K = 1, n = 0$, which is twice the secular frequency. Excitation of ions via higher order resonances ($K > 1$) is possible but may necessitate higher amplitude signals since those resonances are weaker components of ion motion. The presence of buffer gas also appears to have a greater broadening effect on the higher order resonant frequencies.[8]

Additional electrical signals besides the RF and DC voltages for trapping are required to perform resonance methods. Since they are supplementary to the trapping waveforms, these signals are called *auxiliary waveforms* but may also be referred to as supplementary AC waveforms or the "tickle" voltage.[4,9-10] Auxiliary waveforms are often low voltage (3-6 V) and frequency adjustable so that they can be tuned to the specific frequency of resonance, and they can be applied to quadrupole ion traps in different configurations to excite the secular or parametric frequencies.[5]

The implementation and common configurations of auxiliary waveforms in the Paul trap and linear quadrupole ion trap will be discussed in the next section.

Section 3.2: Typical Configurations in Quadrupole Ion Traps

The primary configurations of auxiliary waveforms for resonance methods are dipolar and quadrupolar or parametric excitation. Dipolar excitation primarily excites the secular frequencies (equation 3.1), while quadrupolar/parametric excitation excites the parametric resonances (equation 3.2). In practice, the difference between these methods is where and how the electrical signals are applied. Here I will discuss the traditional way that auxiliary waveforms are applied to the sine wave-driven Paul trap and linear quadrupole for these different resonance methods.

3D Quadrupole or Paul Traps

As shown in Figure 3.1, the Paul trap consists of one hyperbolic ring electrode and two hyperbolic endcaps. An alternating RF waveform is applied to the ring electrode while the endcaps are floated at a DC potential or grounded. Ions enter and exit through the endcaps in the *z* dimension and also move in the radial dimension defined as *r*, which is in the *xy* plane of the ring electrode. The *z* and *r* resonant frequencies differ in the 3D quadrupole field, allowing for selective resonant excitation along either dimension depending on how auxiliary waveforms are applied.[5-6]

Variations of the Paul trap with simplified geometries have also been developed, such as the cylindrical ion trap which has planar electrodes that are easier to machine and miniaturize.[11] The auxiliary waveform arrangements described here can also be utilized for these simplified geometries. However, one must keep in mind that deviating from the more “pure quadrupole” field

created by the hyperbolic cross-section introduces higher order field components. Therefore, additional calculations and simulations will likely be necessary to accurately predict resonances in these simplified traps.[11]

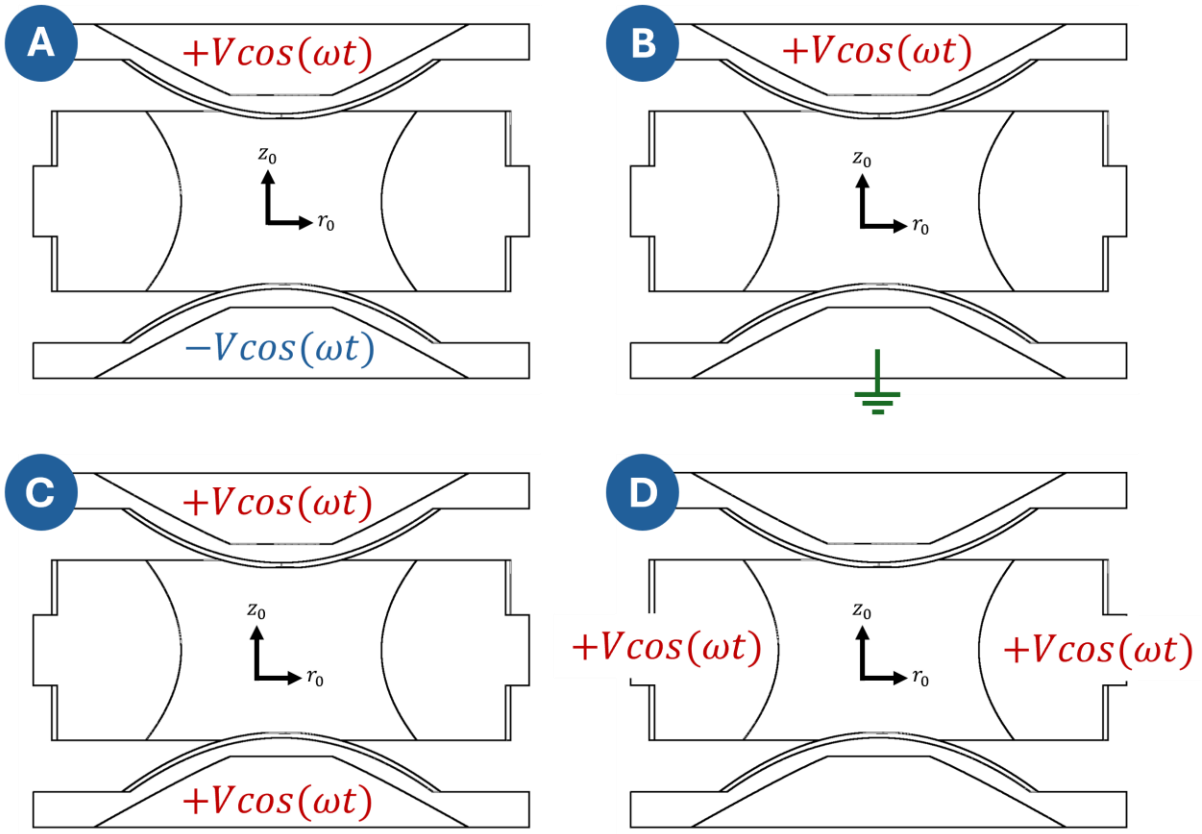


Figure 3.1 – Arrangements of auxiliary waveforms for resonance methods in the 3D Paul trap. A) dipolar excitation, B) monopolar excitation, C) quadrupolar/parametric excitation using the endcaps, D) quadrupolar/parametric excitation using the ring electrode.

The most common arrangement of auxiliary waveforms in Paul traps is dipolar excitation, shown in Figure 3.1A. Dipolar excitation is when two out-of-phase auxiliary waveforms are applied to the endcaps from either side of a bipolar AC supply.[5] This arrangement is simple to implement in the Paul trap since the two endcaps are already electrically isolated from each other, and results in strong excitation of the secular frequency component. The auxiliary AC signal can

also be applied to only one endcap while the other is held at constant potential (usually grounded). This arrangement is called monopolar excitation, shown in Figure 3.1B. The dipolar and monopolar configurations cause different electric field effects, but both have been shown to excite secular frequencies.[3]

Finally, there are two arrangements for parametric excitation in a Paul trap depicted in Figure 3.1C-D. An in-phase AC waveform can be applied to both endcaps in parallel *or* an AC waveform can be superimposed onto the driving RF potential on the ring electrode.[3] In either case, this parametric excitation arrangement has quadrupolar symmetry and therefore can excite ions in either the *z* or *r* dimension if the amplitude of the auxiliary waveform is strong enough. This arrangement allows one to access the parametric resonances from Equation 3.2 as well as the secular frequencies, since they overlap with the K = 2 resonance.[5]

Linear Quadrupole Ion Traps

Auxiliary waveform arrangements in linear quadrupole ion traps or mass filters look similar to those for Paul traps, however, there are practical considerations which make parametric excitation more desirable. Figure 3.2 shows the dipolar excitation setup in a linear quadrupole, where two out-of-phase AC waveforms are superimposed onto the RF waveforms of each X rod (the Y rod pair can also be used).[12-13] This dipolar arrangement requires the two X rods to be electrically isolated from each other, which is not the case under typical use since the same in-phase RF potential is applied to both rods for ion trapping. Therefore, to perform dipolar excitation in a linear quadrupole, significant modifications to the driving circuitry must be made.[14] To this author's knowledge, monopolar excitation has not been employed in linear quadrupoles since the RF potential must be kept on all four rods to maintain the quadrupole field.

In contrast, quadrupolar or parametric excitation in a linear quadrupole does not require that the X or Y rod pair be electrically isolated from its mate. This setup is depicted in Figure 3.2, where an AC waveform is superimposed in-phase with the RF on the X rods, and another AC waveform is superimposed in-phase with the RF on the Y rods. This allows for transformer coupling of the low-voltage auxiliary AC waveform with the already present RF waveform for each rod pair.[14-15] Due to this simpler implementation, parametric excitation is the most common arrangement for quadrupoles of linear geometry, even though it primarily accesses the parametric resonances instead of the fundamental secular frequencies.[14]

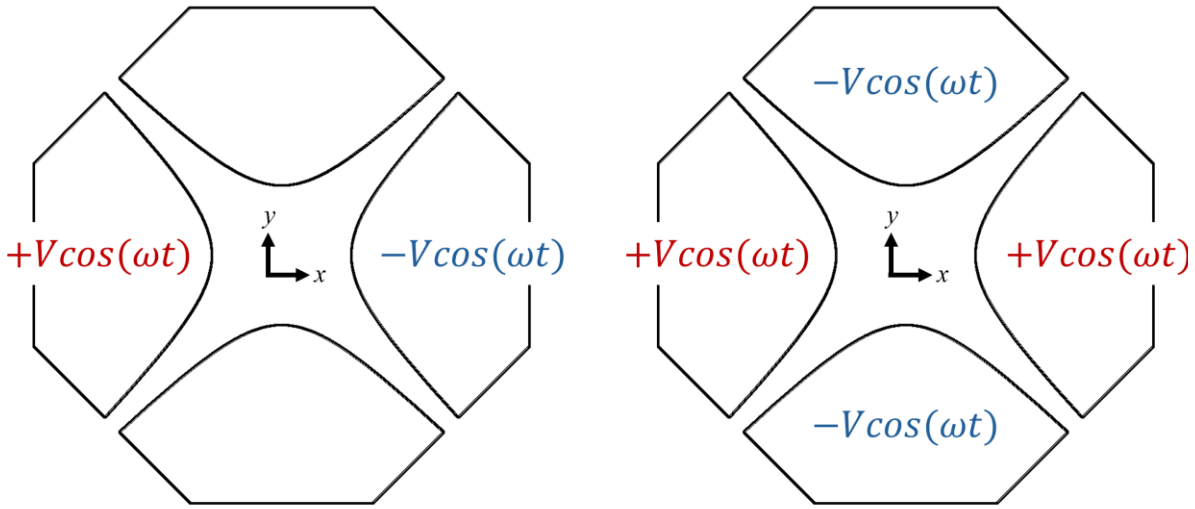


Figure 3.2 – Arrangements of auxiliary waveforms for resonance methods in the linear ion trap or mass filter. Dipolar excitation (left) and quadrupolar/parametric excitation (right).

Section 3.3: Resonance Methods in Digital Ion Traps

In sine wave-driven linear quadrupoles, the auxiliary waveform can be superimposed onto the base RF waveform for parametric or dipolar excitation by sending both signals through a transformer. This results in a complex waveform similar to that shown in Figure 3.3, which contains both the high frequency RF and lower frequency AC components.[14-15] This transformer coupling method has been used sparingly for digital ion traps,[16-17] but results in distortion of the square wave signal. This can make the square wave more rounded, losing the sharp duty cycle control which is desirable for other digital trapping techniques. Therefore, if you want the precise control of ion motion offered by resonance methods in your digital ion trap, an alternate method to transformer coupling would be advantageous.

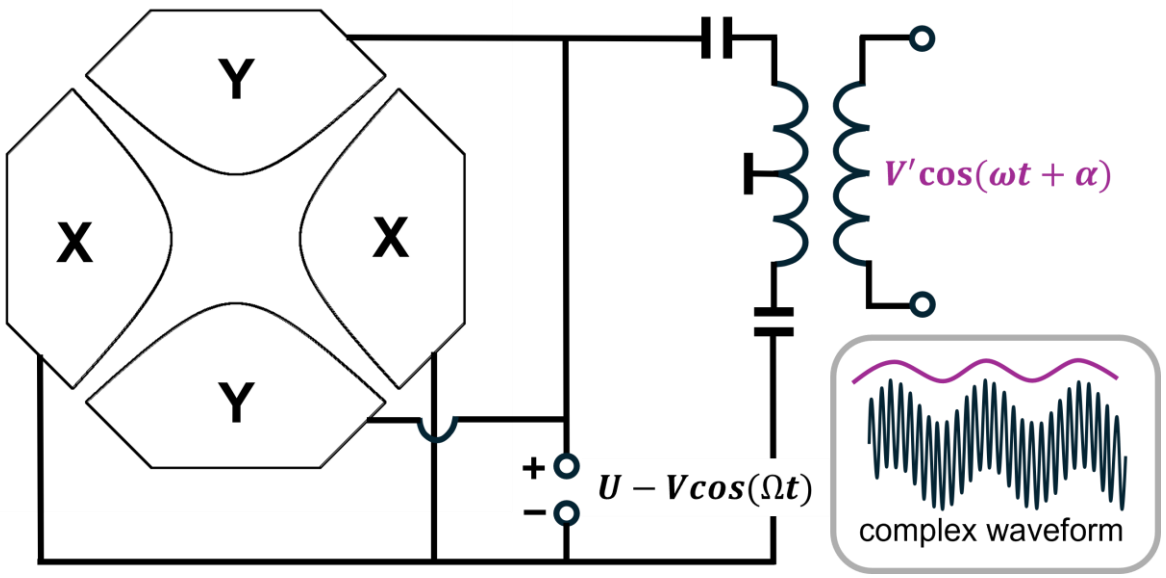


Figure 3.3 – Application of sine wave complex waveform for parametric excitation in a linear ion trap or guide, which uses a transformer to couple the RF and auxiliary AC signals. An example of what the complex waveform may look like is shown in the bottom right corner, with the purple trace representing the auxiliary frequency used for resonance.

Over the course of my PhD research, I developed a method to address this concern which utilizes only digital circuitry to produce a complex waveform for resonant excitation.[18] This square complex waveform is produced by sending TTL-level signals from an Arduino-controlled low-voltage waveform generator through a series of high-voltage MOSFET (metal oxide field effect transistor) switches to couple the base RF and auxiliary waveforms together. A schematic of this process is shown in Figure 3.4, with more details about the circuitry given in the CD-LIT-MS user's manual in Chapter 5. The superposition of the auxiliary waveform onto the RF is represented in Figure 3.4 by the blue trace.

In our current setup in Figure 3.4, the same complex square waveform is applied in-phase to both X rods of the linear quadrupole ion trap, making this a parametric arrangement. Usually for this arrangement, an additional in-phase auxiliary waveform would be applied to the Y rods as shown in Figure 3.3. However, I discovered that this additional waveform was not necessary to successfully perform parametric excitation experiments. The current setup therefore uses the minimal number of bipolar MOSFET switches (4) but can access parametric resonances in both the X and Y dimensions of the linear ion trap.

Figure 3.5 shows oscilloscope traces of the square complex waveform output with an RF frequency of 300 kHz and two different auxiliary applied frequencies. The upper plot contains a trace with an auxiliary frequency of 50 Hz which is an even division of the RF, while the lower plot shows a waveform with auxiliary frequency of 177 kHz which is not an even division of the RF. The auxiliary amplitude was set to 7 V in this case for visual clarity, although it is usually kept to 3-4 V in practice. Figure 3.5 illustrates that our circuitry allows for independent control of the RF and auxiliary frequency, which is necessary for resonance methods such as parametric excitation where the auxiliary and RF frequencies are not always even divisions of one another.

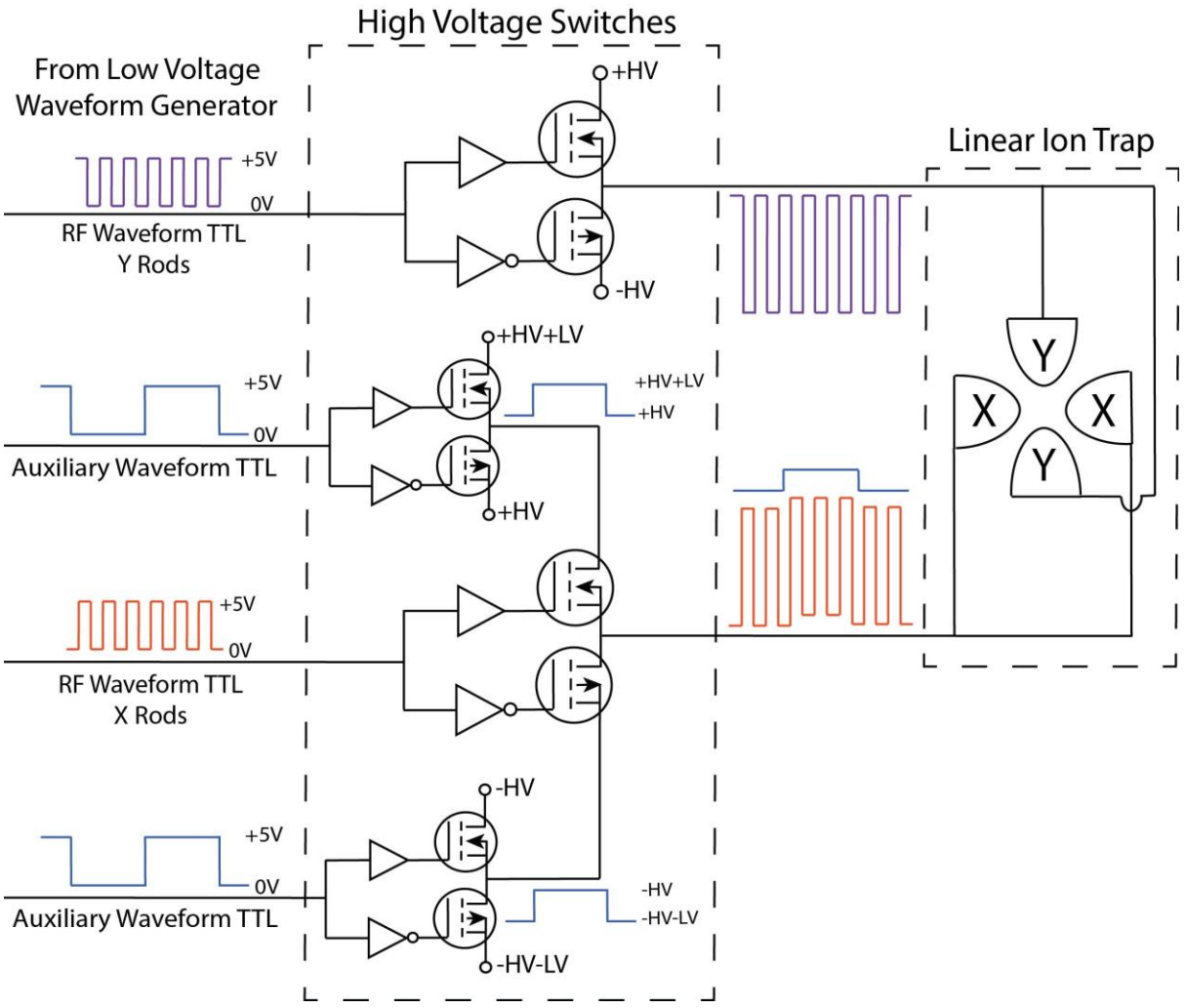


Figure 3.4 – Schematic of the square wave driver which produces the complex square waveform for resonant excitation, applied only to the X rods as shown. The superposition of the lower frequency auxiliary component is represented by the blue trace. HV denotes the +50/-50 V outputs of a 100 V DC power supply, while LV denotes the outputs of 0 -7 V DC supplies used to adjust the auxiliary waveform amplitude.

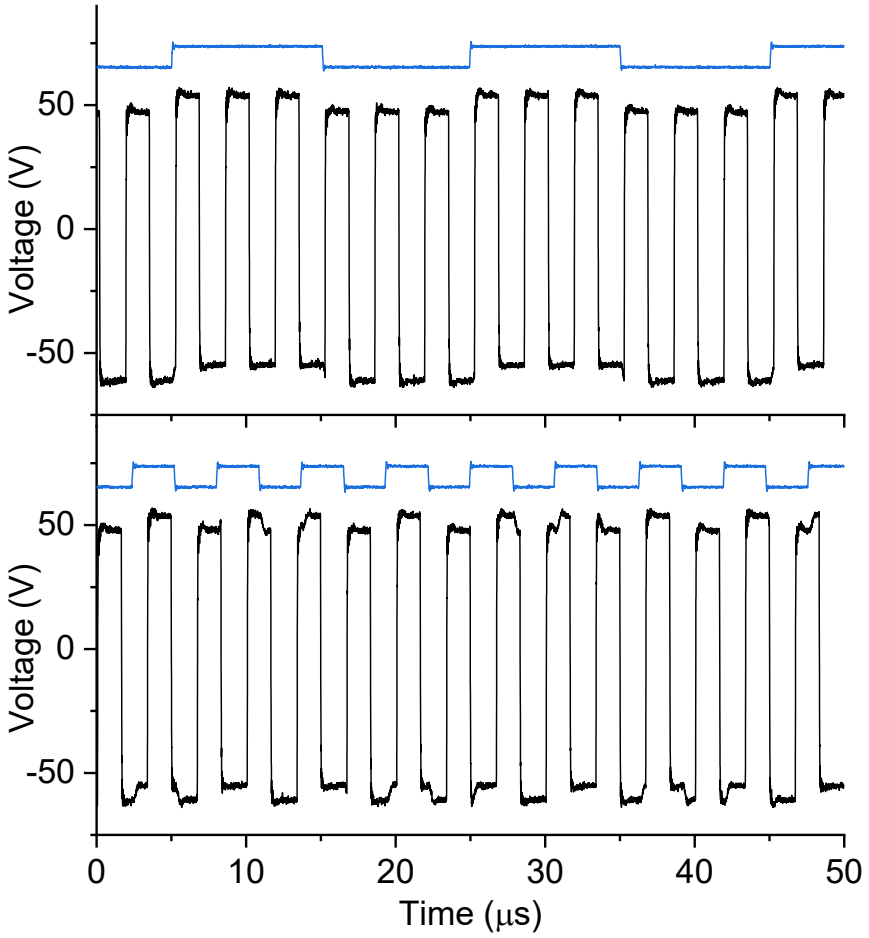


Figure 3.5 – Oscilloscope traces of the complex square waveform applied (black) and the corresponding auxiliary TTL (blue). The RF frequency is 300 kHz for both plots, and the auxiliary frequencies are 50 kHz (top) and 177 kHz (bottom), with an auxiliary excitation voltage of 7 V.

Although the current resonance arrangement in the CD-LIT-MS is for square parametric excitation on the X rods, more traditional parametric excitation on both rod pairs and dipolar excitation is possible. For more traditional parametric excitation, you would only need to duplicate the bipolar MOSFET switch arrangement on the Y rods (total of 6 MOSFET switches instead of 4). Similarly, the dipolar arrangement could be achieved by electrically isolating the two X rods and adding an additional out-of-phase auxiliary TTL input, resulting in 3 switches for each X rod

and a total of 7 for the entire linear quadrupole setup. In this way, the square complex waveform is a digital alternative to currently available sinusoidal resonance methods. Experimental results demonstrating the MS capabilities of this technique are shown and discussed in Chapter 4.

Python program for calculating parametric resonances in a digital quadrupole

In order to predict the parametric frequencies in my digital linear ion trap, I wrote a Python program (file name: *Matrix_calc_quad_stability.py*) to calculate the first order parametric resonances using the β parameter for the square wave case (Chapter 2, Equation 2.19). Simplifying Equation 3.2 for the first order resonance at $K = 1, n = 0$ leads to following equation,

$$\omega_{u,0}^{1} = \beta_u \Omega \tag{3.3}$$

Where β can be calculated for the X and Y dimensions from the matrix solutions to the Mathieu-Hill equations, and Ω is once again the RF frequency in radians. This equation defines the strongest parametric resonance, and the only one that I have clearly observed experimentally in the LIT. However, this program could easily be modified to calculate higher order parametric resonances or secular frequencies according to Equations 3.2-3.2.

This program looks similar to the one described in Chapter 2 for plotting digital quad Mathieu stability diagrams (file name: *Mathieu_stability_diagrams_2.0.py*), since the majority of the code is there to calculate the transfer matrices for X and Y so that we can find the β parameter. First, the same two functions are present to calculate the matrices when $f > 0$ and $f < 0$, called "transM_pos" and "transM_neg". Next there is a function called "K1freq_calc" which takes the inputs: array of m/z values, square wave RF frequency in kHz, t1/T (decimal fraction X high duty cycle), and t3/T (decimal fraction Y high duty cycle). This function first calculates the Mathieu *a*

and q parameters and f values for a linear quadrupole, then creates the transfer matrices for the three time points (t1, t2, and t3) and multiplies them together to get the total transfer matrix (M). Finally, it calculates β from Equation 2.19 and the first order parametric resonance ω from Equation 3.3 above. The function returns an array of parametric resonances in kHz corresponding to the m/z array that you input.

You will see from the screenshots of this code in Appendix section A2 that below the function definitions is the code to plot parametric frequency trends. You can modify that code below to plot different trends or simply copy the functions to do your own resonance calculations for a digital quadrupole. In the next chapter, the calculated and experimental trends in parametric excitation frequency versus RF frequency and X:Y duty cycles are shown in plots made from this Python program.

References

(1) *4.3: Resonance*. Physics LibreTexts. https://phys.libretexts.org/Bookshelves/Conceptual_Physics/Conceptual_Physics_(Crowell)/04%3A_Conservation_of_Momentum/4.03%3A_Resonance (accessed 2025-12-06).

(2) Seyranian, A. P. The Swing: Parametric Resonance. *J. Appl. Math. Mech.* **2004**, *68* (5), 757–764. https://doi.org/10.1016/j.jappmathmech.2004.09.011.

(3) March, R. E.; Todd, J. F. J. Quadrupole Ion Trap Mass Spectrometry, 2005.

(4) Snyder, D. T.; Pulliam, C. J.; Wiley, J. S.; Duncan, J.; Cooks, R. G. Experimental Characterization of Secular Frequency Scanning in Ion Trap Mass Spectrometers. *J. Am. Soc. Mass Spectrom.* **2016**, *27* (7), 1243–1255. https://doi.org/10.1007/s13361-016-1377-1.

(5) March, R. E.; McMahon, A. W.; Londry, F. A.; Alfred, R. L.; Todd, J. F. J.; Vedel, F. Resonance Excitation of Ions Stored in a Quadrupole Ion Trap. Part 1. A Simulation Study. *Int. J. Mass Spectrom. Ion Processes* **1989**, *95* (2), 119–156. https://doi.org/10.1016/0168-1176(89)83037-X.

(6) March, R. E. Quadrupole Ion Traps. *Mass Spectrometry Reviews* **2009**, *28* (6), 961–989. https://doi.org/10.1002/mas.20250.

(7) Zhao, X.; Ryjkov, V. L.; Schuessler, H. A. Parametric Excitations of Trapped Ions in a Linear Rf Ion Trap. *Phys. Rev. A* **2002**, *66* (6), 063414. https://doi.org/10.1103/PhysRevA.66.063414.

(8) Collings, B. A.; Douglas, D. J. Observation of Higher Order Quadrupole Excitation Frequencies in a Linear Ion Trap. *J. Am. Soc. Spectrom.* **2000**, *11* (11), 1016–1022. https://doi.org/10.1016/S1044-0305(00)00171-9.

(9) Vedel, F.; Vedel, M.; Evans March, R. New Schemes for Resonant Ejection in r.f. Quadrupolar Ion Traps. *Int. J. Mass Spectrom. Ion Processes* **1990**, *99* (1–2), 125–138. https://doi.org/10.1016/0168-1176(90)85025-W.

(10) Vettori, U.; Traldi, P. How to Achieve a Well-Defined Resonance Excitation in Ion-Trap Mass Spectrometry. *Rapid Commun. Mass Spectrom.* **1993**, *7* (4), 312–314. https://doi.org/10.1002/rcm.1290070409.

(11) Badman, E. R.; Johnson, R. C.; Plass, W. R.; Cooks, R. G. A Miniature Cylindrical Quadrupole Ion Trap: Simulation and Experiment. *Anal. Chem.* **1998**, *70* (23), 4896–4901. https://doi.org/10.1021/ac980908w.

(12) Collings, B. A.; Stott, W. R.; Londry, F. A. Resonant Excitation in a Low-Pressure Linear Ion Trap. *J. Am. Soc. Mass Spectrom.* **2003**, *14* (6), 622–634. https://doi.org/10.1016/S1044-0305(03)00202-2.

(13) Snyder, D. T.; Pulliam, C. J.; Wiley, J. S.; Duncan, J.; Cooks, R. G. Experimental Characterization of Secular Frequency Scanning in Ion Trap Mass Spectrometers. *J. Am. Soc. Mass Spectrom.* **2016**, *27* (7), 1243–1255. https://doi.org/10.1007/s13361-016-1377-1.

(14) Sudakov, M.; Konenkov, N.; Douglas, D. J.; Glebova, T. Excitation Frequencies of Ions Confined in a Quadrupole Field with Quadrupole Excitation. *J. Am. Soc. Mass Spectrom.* **2000**, *11* (1), 10–18. https://doi.org/10.1016/S1044-0305(99)00111-7.

(15) Collings, B. A.; Douglas, D. J. Observation of Higher Order Quadrupole Excitation Frequencies in a Linear Ion Trap. *J. Am. Soc. Spectrom.* **2000**, *11* (11), 1016–1022. https://doi.org/10.1016/S1044-0305(00)00171-9.

(16) Xu, F.; Wang, L.; Dai, X.; Fang, X.; Ding, C.-F. Resonance Activation and Collision-Induced-Dissociation of Ions Using Rectangular Wave Dipolar Potentials in a Digital Ion Trap Mass Spectrometer. *J. Am. Soc. Mass Spectrom.* **2014**, *25* (4), 556–562. https://doi.org/10.1007/s13361-013-0804-9.

(17) Duan, C.; Zhang, J.; Xian, T.; Li, L.; Zhang, Y.; He, X.; Li, P. Direct Performance of Triple-Stage Tandem Mass Spectrometry Analysis Using Dual-Direction Dipolar Excitation in a Digital Linear Ion Trap. *J. Am. Soc. Mass Spectrom.* **2024**, *35* (3), 551–560. https://doi.org/10.1021/jasms.3c00406.

(18) Capek, G. O.; Howdieshell, C. J.; Garand, E. Square Parametric Excitation: A Digital Resonant Method for the Quadrupole Ion Trap. *J. Am. Soc. Mass Spectrom.* **2024**, *35* (8), 1846–1853. https://doi.org/10.1021/jasms.4c00169.

Chapter 4: Capabilities of Square Parametric Excitation

The previous chapter gave a broad overview of resonance methods in quadrupole ion traps, introduced the complex square waveform, and showed how this waveform is currently applied in the parametric arrangement to the LIT. In this chapter, previously published results of the square parametric excitation technique on the CD-LIT-MS are presented.[1]

A brief introduction is given in section 4.1. We demonstrate that this digital alternative waveform can perform the same functions as its sine wave counterparts, starting with selective excitation of one m/z species in section 4.2. Section 4.2 also illustrates that there is good agreement between experimental results and calculated parametric frequencies from the Python program described in Chapter 3. In section 4.3, we describe a precise mass isolation scheme which utilizes a RF frequency and duty cycle switch and a notched scan of the parametric excitation signal. Finally, section 4.4 shows the results of a parametric resonance-enhanced mass scan for increased signal and mass resolution, followed by overall conclusions in section 4.5.

Section 4.1: Introduction and Methods

As introduced in Chapter 2 of this thesis, the development of modern digital circuitry has led to a renewed interest in square wave-drive quadrupole devices in recent decades.[2-5] Digital ion traps have become a viable alternative to sine wave-driven traps for certain applications, especially for mass analysis of heavy species due to their higher upper m/z limit.[6] Adjustment of the square wave frequency and duty cycle has been shown to change the shape of the Mathieu stability diagram, allowing for simultaneous mass filtration or isolation.[7-8] However, there are still ion trap operations such as ion pre-ejection or dissociation where resonance methods are preferred due to

the selectivity and precision that they offer. As discussed in detail in Chapter 3, resonance methods employ an additional frequency to excite m/z-specific resonances of ion motion in the quadrupole field.[9-10]

Here, we present the results of a nontraditional digital resonance method which does not require transformer coupling of the driving RF and auxiliary waveforms. Our complex square waveform was produced using only digital circuitry and applied to the X rods of the LIT in the parametric configuration as shown in Chapter 3. The custom mass spectrometer used for these studies was described previously in Chapter 1, with more instrumental and data analysis details provided in the following Chapter 5.

The electrosprayed solution used for this study consisted of a set of amino acids and small polypeptides (Diglycine, L-Histidine, L-alanyl-L-alanine, L-Phenylalanine, L-Tyrosine, Triglycine, and Tri-L-Alanine) purchased from Millipore Sigma and combined in HPLC grade methanol at a total concentration of 1 mM with 0.1% formic acid to aid with solubility and ionization. The concentrations of individual species are tabulated in Appendix section A3 along with details on the mass calibration for these spectra in A4. Ions were generated by ESI at 2-3 kV using 5-20 μm diameter borosilicate glass tips pulled by a filament micropipette puller (MicroData Instrument PMP102).

Section 4.2: Selective Mass Elimination and Parametric Frequency Trends

First, we demonstrate that the complex square waveform can selectively activate a narrow range of m/z species while leaving other m/z species undisturbed, therefore eliminating certain species from the ion trap. This scheme is analogous to experiments in sine wave based linear quadrupoles and 3D Paul traps where the resonant signal is set to one fixed frequency to activate a small m/z range for pre-ejection or fragmentation.[11-12] If the amplitude of the auxiliary signal and buffer gas pressure are high enough, we can observe fragmentation from collision-induced dissociation (CID) in the LIT, although this capability was not rigorously explored. The primary use of this mode in the CD-LIT-MS is to selectively eliminate any untagged species or undesired solvent clusters before spectroscopic analysis (see Chapter 6 for results on water clusters).

Figure 4.1 shows the sequential elimination of species at 156 m/z, 161m/z, and 166 m/z from a mixture. These m/z species were eliminated individually by applying a complex square waveform consisting of a RF driving square wave with 550 kHz frequency and 52:45 duty cycles and auxiliary frequency of 183 kHz, 175 kHz, and 170 kHz, respectively, for 10 ms before acquiring the mass spectrum. The appropriate auxiliary frequencies were determined experimentally, and their amplitude was set to 3.5 V to avoid fragmentation of the m/z ion of interest while still causing enough excitation for complete elimination.

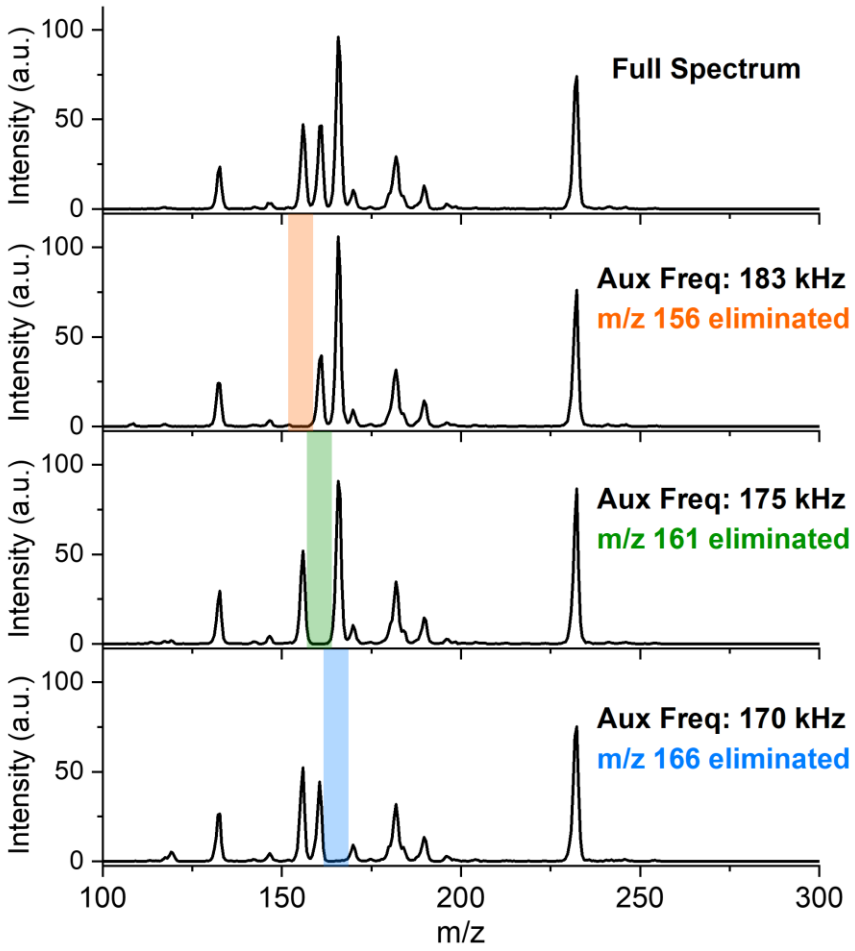


Figure 4.1 – Elimination of specific m/z species in a mixture by their resonant parametric frequencies. The driving square wave frequency (550 kHz) and X:Y duty cycles (52:45) were fixed for the 10 ms excitation window.

To confirm that we were exciting the first order (K = 1) parametric resonance, we experimentally determined the excitation frequency for the elimination of 7 different m/z at various RF frequencies and fixed 52:45 duty cycles, as shown in Figure 4.2. The error bars on the experimental data points correspond to the range of experimental parametric frequencies which were effective at resonantly exciting and eliminating that m/z peak. The expected first order

parametric resonances in a square wave quadrupole field were calculated using the Python program described in Chapter 3 for comparison with the experimental data.

Figure 4.2 shows that there is good agreement between the calculated first order parametric resonances and experimental frequencies found for resonant mass elimination. This confirms that these are in fact first order (K = 1) parametric resonances. Excitation was not observed at any higher order (K > 1) resonant frequencies, most likely because they are weaker and typically require larger auxiliary amplitudes to excite.[13-14]

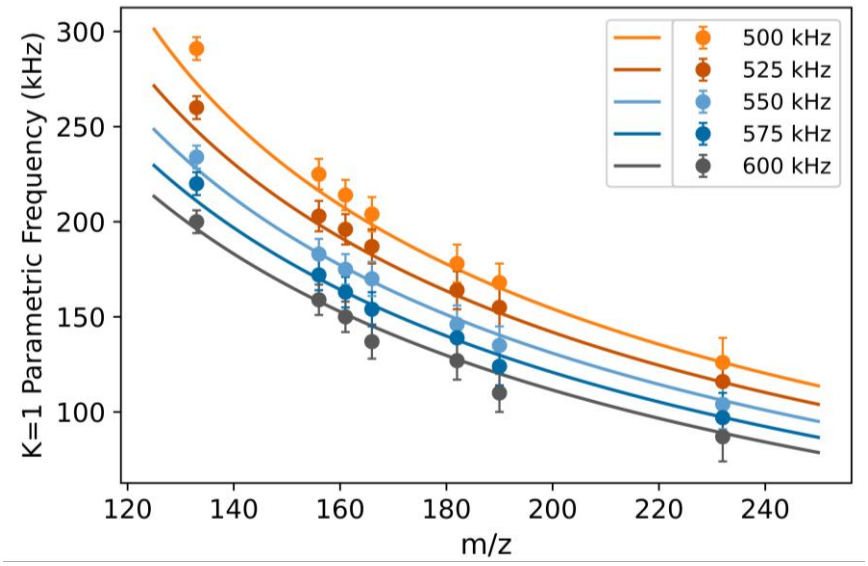


Figure 4.2 – Trend in first order parametric resonance vs. m/z at different driving square wave frequencies for the Y dimension of the LIT. Lines are calculated frequencies and points are experimental data. The X and Y duty cycles were fixed to 52:45.

Additionally, we mapped the trend in parametric excitation frequency vs. m/z along with X and Y duty cycles as shown in Figure 4.3. In our experiment, there is a small non-zero DC potential (U) component which occurs from asymmetry in the high/low voltages coming from the driving circuitry. This leads to a nonzero Mathieu a parameter and causes the X and Y parametric

excitation frequencies to diverge, a phenomenon which has been explained previously for sine wave-based resonance methods.[13] Although the square parametric excitation waveform is only applied to the X rods, we have shown that it can selectively excite both the X and Y parametric resonances independently. To experimentally differentiate between X and Y frequencies, ion signal on the detector was recorded during the mass elimination window. If ions were ejected during that window at a given excitation frequency while a specific m/z peak was simultaneously eliminated from the mass spectrum, the excitation was only along the X axis. This was further confirmed by switching the X and Y duty cycles to visualize that no excitation was occurring on the opposing axis.

Figure 4.3 also shows that as the X and Y duty cycles approach 50:50, their first order parametric frequencies approach each other. We believe this is due to the fact that asymmetric duty cycles act similarly to a time-averaged DC potential (*U*). This *U* contribution diminishes as the duty cycles become more symmetric between X and Y, resulting in closer parametric excitation frequencies. Essentially, as duty cycles approach 50:50, the quadrupole field becomes closer to the completely symmetrical case where the X and Y resonances are equal. Once again, the experimental parametric excitation frequencies in Figure 4.3 are in excellent agreement with the calculated first order values.

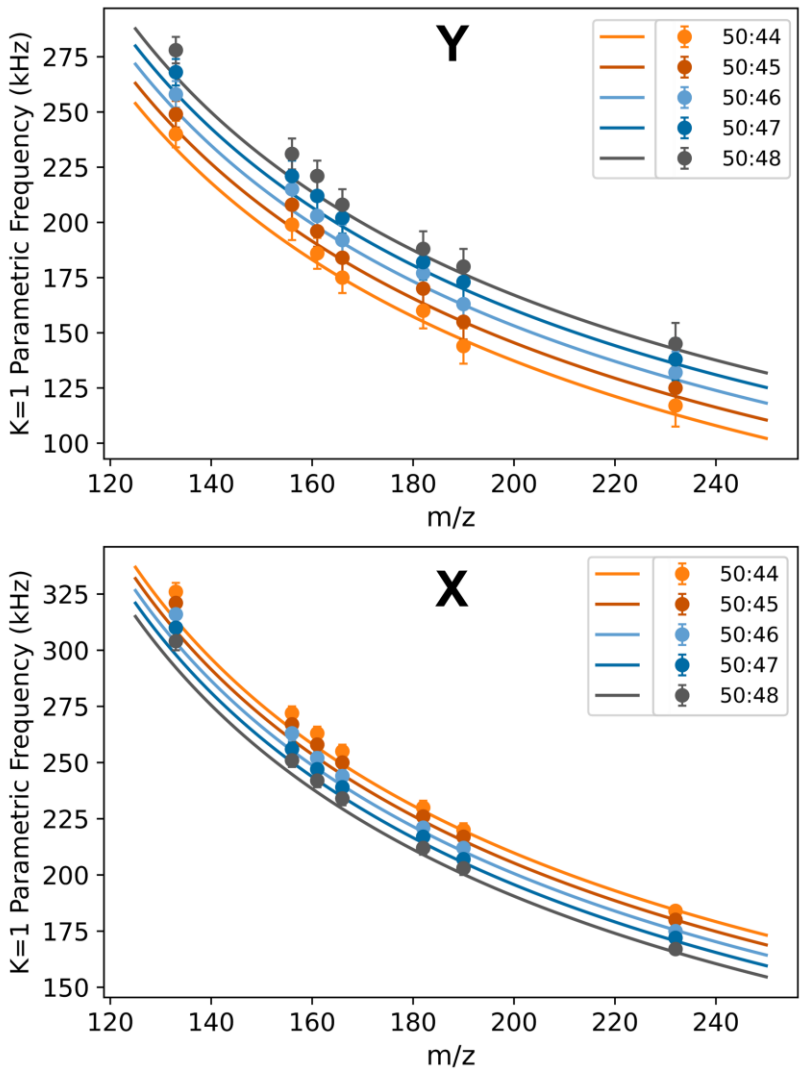


Figure 4.3. The trend in first order parametric resonance vs. m/z for the X dimension (bottom) and Y dimension (top) of the LIT at different duty cycle combinations. Lines are calculated frequencies and points are experimental data. Duty cycles are denoted as d_x: d_y in the upper right, while the driving square wave frequency was fixed to 550 kHz.

Section 4.3: Mass Isolation

The square parametric excitation waveform can also be used to isolate a selected m/z species. Mass isolation in a digital ion trap by RF frequency and duty cycle manipulation has been reported previously[2,15] and utilized by our group.[16] In short, this method works by moving the boundaries of the Mathieu stability diagram so that the desired m/z ion(s) are isolated in the apex of the stable region (see Chapter 2 of this thesis). Therefore, a combined frequency and duty cycle switch can be used to achieve mass isolation with only the driving RF square waves. However, isolation of a narrow m/z range results in significant loss of signal, likely due to the fact that ions can become prematurely unstable when they approach the stability boundary.[17] Maintaining the intensity of the peak of interest is possible, but typically results in incomplete isolation, as shown in panel 2 of Figure 4.4. This trade-off between narrow mass isolation and signal intensity is illustrated in panel 3 of Figure 4.4, where the most complete isolation solely by frequency and duty cycle manipulation resulted in >50% loss in peak intensity.

In contrast to frequency and duty cycle manipulation of the driving waveform, parametric excitation does not change the overall shape of the Mathieu *a*-*q* stability region, but instead creates additional unstable lines within it.[18] Resonant mass isolation can be performed by scanning the parametric excitation frequency and skipping the region which contains the ion of interest. Similar resonance-based methods have been used in RF sine wave quadrupoles and are referred to as notched frequency scans.[19-20]

In order to avoid long scans of the parametric excitation frequency over the entire mass range, we performed a 5 ms duty cycle and frequency switch to filter out most of the unwanted species. Then we followed it by a 3.5 ms scan of the parametric excitation frequency from 200 kHz to 165 kHz with a notch between 184 kHz and 174 kHz. This method allows for the isolation

of a narrower m/z range while maintaining more signal integrity in comparison to frequency and duty cycle manipulation alone. The protonated L-alanyl-L-alanine (m/z 161) isolated in panel 6 of Figure 4.4 underwent only a ~12% loss in intensity using this combined method, and nearby peaks (+/-5 m/z) were completely filtered out in the mass spectrum.

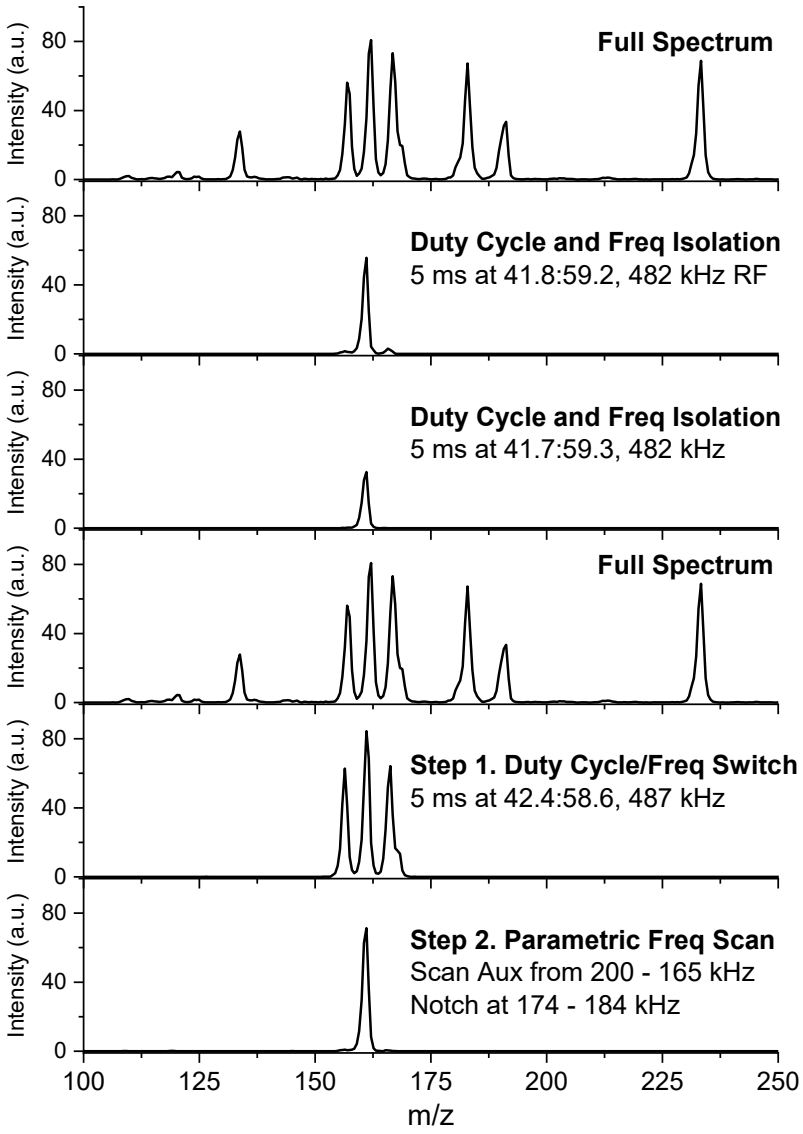


Figure 4.4. Mass isolation of Ala_2H^+ (m/z 161) by RF duty cycle and frequency manipulation (top 3 panels), compared to a combined approach with a RF duty cycle and frequency switch followed by a notched frequency scan of the auxiliary frequency (bottom 3 panels).

Section 4.4: Parametric Resonance-Enhanced Mass Scan

Finally, we applied the square parametric excitation waveform during mass ejection to improve S/N and mass resolution performance of the LIT. The auxiliary frequency was scanned along with the RF frequency to resonantly excite ions when they are already close to the stability boundary, causing them to exit the trap with higher kinetic energy and in a more compact ion packet. This is made possible by the circuitry which employs the same waveform generator for both the RF and auxiliary waveforms. We found that for a narrow mass range (100 – 250 Da), the auxiliary frequency could be scanned linearly with respect to m/z and RF frequency and still have the desired effect, even though the true mathematical relationship is non-linear (this is evident in Figures 4.2-4.3).

The results are shown in Figure 4.5, where the auxiliary frequency was scanned at the experimentally determined rate: $auxiliary\ frequency\ (kHz) = 1.12(RF\ in\ kHz) - 109$. This auxiliary frequency scan corresponds to a linear scan of the Mathieu β parameter from 0.93 to 0.83 as the RF frequency is scanned for the mass range 100 – 250 m/z. Under typical operation, as the RF frequency is scanned, ions are sequentially moved to higher q coordinates until they reach the X-axis $\beta = 1$ stability boundary where they become unstable and eject out of the trap. By setting the auxiliary frequency close to the $\beta = 1$ boundary and scanning downward, we are pre-exciting the ions as they approach that ejection boundary. This can result in a slight upper shift in RF ejection frequency, since ions may eject at lower β and q values. This effect can be seen with m/z 232 in Figure 4.5, which undergoes a slight shift to higher ejection RF frequency, corresponding to lower m/z in the initial mass calibration.

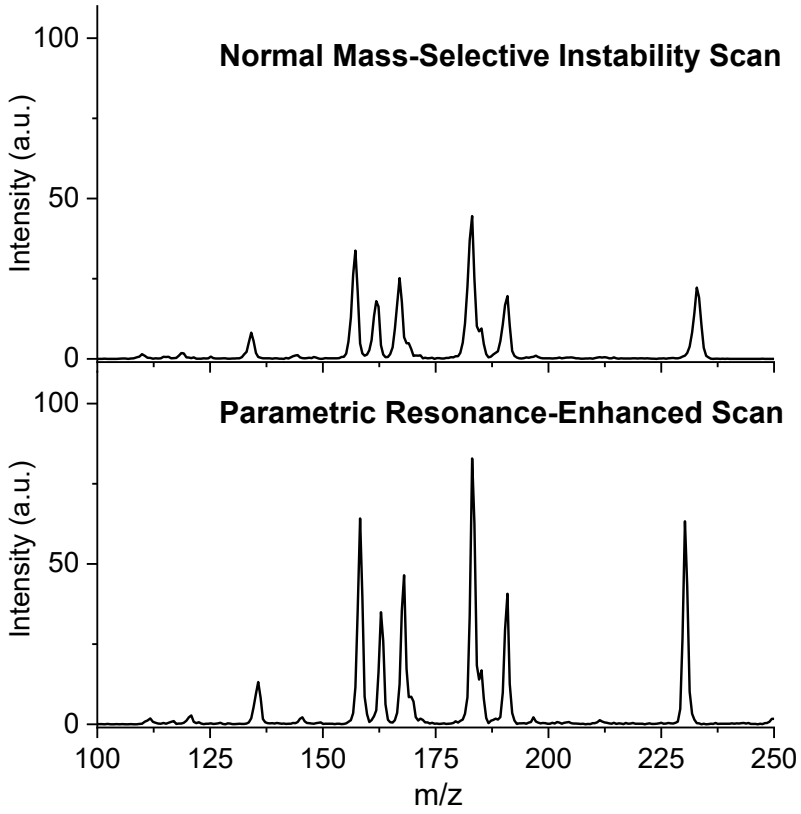


Figure 4.5. Comparison of the mass resolution and S/N of a normal mass-selective instability frequency scan (top) versus the parametric resonance-enhanced scan (bottom). The experimentally determined relationship used here was: $auxiliary\ frequency\ (kHz) = 1.12(RF\ in\ kHz) - 109$ with 52:45 duty cycles.

Application of this waveform decreased the average full-width half-max (FWHM) by ~25% (from 1.6 to 1.2) and increased the average peak intensity by ~50% (from 24 to 49). The best improvement in mass resolution occurred with the highest mass species in this mixture at m/z 232 (Ala_3H^+), whose FWHM was reduced by ~35% (from 1.7 to 1.1) during the parametric resonance-enhanced scan. We believe this technique holds promise for enhancing mass resolution performance at higher mass ranges where the digital ion trap can be clearly advantageous.

The effects of duty cycle and scan rate on the parametric resonance-enhanced scan were also explored and the results are summarized in Appendix section A5. For minor adjustments in X or Y duty cycles (1-2%), the same linear scan of parametric frequency can be used and results in comparable enhancement to what is shown in Figure 4.5. Decreasing the scan rate also results in better mass resolution, which is consistent with results from sine wave-based resonant ejection.[9] Note that for these experiments, it was important to ensure that only the X-dimension first order parametric resonances were being excited at the duty cycles selected for ejection, since ions are ejected along the X axis in our instrument. Any excitation of the Y-dimension first order parametric resonances resulted in no change or reduced mass resolution compared to normal operation.

Section 4.5: Conclusions

With the continuing development of faster, more compact, and more accurate digital circuitry, RF square waves are proving to be a viable alternative to traditional quadrupolar ion trapping for specific applications. Here, we presented a digitally produced complex square waveform which can excite the first order parametric resonances of ion motion in both the X and Y dimensions of a linear ion trap. We showed that by applying this parametric excitation signal at a fixed frequency to one set of the quadrupole rods, specific m/z ions can be excited for pre-ejection or elimination. Additionally, the implementation of a notched parametric frequency scan resulted in more precise mass isolation than what was possible with duty cycle and driving frequency manipulation. Finally, we saw an up to 35% decrease in FWHM by applying the parametric excitation signal during the mass-selective radial instability scan, demonstrating the ability of this resonant technique for mass resolution enhancement in digital ion traps.

The results presented here are limited examples of how this square parametric excitation waveform may be implemented to expand the functionality of digital ion traps. The digital driving circuity described herein allows for ease of adjustment of the driving frequency and duty cycles, resonant excitation frequency and duty cycle, and timing of application. The relatively simple and repetitive circuit design could also support different configurations of resonance methods, such as dipolar or monopolar excitation. In the Garand group, this technology will allow for more selective mass filtration than what is available on current CIVS instrumentation, while maintaining the functionality of our cryogenic digital ion traps. The selective mass elimination capability shown here in section 4.2 will also be used to remove every other water cluster or any untagged species during the multiplexed spectroscopy workflow in the LIT.

References

(1) Capek, G. O.; Howdieshell, C. J.; Garand, E. Square Parametric Excitation: A Digital Resonant Method for the Quadrupole Ion Trap. *J. Am. Soc. Mass Spectrom.* **2024**, *35* (8), 1846–1853. https://doi.org/10.1021/jasms.4c00169.

(2) Brancia, F. L.; McCullough, B.; Entwistle, A.; Grossmann, J. G.; Ding, L. Digital Asymmetric Waveform Isolation (DAWI) in a Digital Linear Ion Trap. *J. Am. Soc. Mass Spectrom.* **2010**, *21* (9), 1530–1533. https://doi.org/10.1016/j.jasms.2010.05.003.

(3) Xue, B.; Sun, L.; Huang, Z.; Gao, W.; Fan, R.; Cheng, P.; Ding, L.; Ma, L.; Zhou, Z. A Hand-Portable Digital Linear Ion Trap Mass Spectrometer. *Analyst* **2016**, *141* (19), 5535–5542. https://doi.org/10.1039/C6AN01118G.

(4) Reece, M. E.; Huntley, A. P.; Moon, A. M.; Reilly, P. T. A. Digital Mass Analysis in a Linear Ion Trap without Auxiliary Waveforms. *J. Am. Soc. Mass Spectrom.* **2020**, *31* (1), 103–108. https://doi.org/10.1021/jasms.9b00012.

(5) Simke, F.; Fischer, P.; Marx, G.; Schweikhard, L. Simulations of a Digital Ion Filter and a Digital Ion Trap for Heavy Biomolecules. *Int. J. Mass Spectrom.* **2022**, *473*, 116779. https://doi.org/10.1016/j.ijms.2021.116779.

(6) Ding, L.; Sudakov, M.; Brancia, F. L.; Giles, R.; Kumashiro, S. A Digital Ion Trap Mass Spectrometer Coupled with Atmospheric Pressure Ion Sources. *J. Mass Spectrom.* **2004**, *39* (5), 471–484. https://doi.org/10.1002/jms.637.

(7) Brancia, F. L.; McCullough, B.; Entwistle, A.; Grossmann, J. G.; Ding, L. Digital Asymmetric Waveform Isolation (DAWI) in a Digital Linear Ion Trap. *J. Am. Soc. Mass Spectrom.* **2010**, *21* (9), 1530–1533. https://doi.org/10.1016/j.jasms.2010.05.003.

(8) Brabeck, G. F.; Reilly, P. T. A. Mapping Ion Stability in Digitally Driven Ion Traps and Guides. *Int. J. Mass Spectrom.* **2014**, *364*, 1–8. https://doi.org/10.1016/j.ijms.2014.03.008.

(9) Snyder, D. T.; Peng, W.-P.; Cooks, R. G. Resonance Methods in Quadrupole Ion Traps. *Chem. Phys. Lett.* **2017**, *668*, 69–89. https://doi.org/10.1016/j.cplett.2016.11.011.

(10) Chen, G.; Wang, Z.; Yan, H.; Li, A.; Xie, R.; Li, D. Recent Advances in the Application of Ion Resonance Methods in Ion Trap Mass Spectrometers. *Anal. Methods* **2025**. https://doi.org/10.1039/D5AY01078K.

(11) McLuckey, S. A.; Goeringer, D. E.; Glish, G. L. Selective Ion Isolation/Rejection over a Broad Mass Range in the Quadrupole Ion Trap. *J. Am. Soc. Spectrom.* **1991**, *2* (1), 11–21. https://doi.org/10.1016/1044-0305(91)80056-D.

(12) Cousins, L. M.; Thomson, B. A. MS3 Using the Collision Cell of a Tandem Mass Spectrometer System. *Rapid Commun. Mass Spectrom.* **2002**, *16* (11), 1023–1034. https://doi.org/10.1002/rcm.674.

(13) Collings, B. A.; Douglas, D. J. Observation of Higher Order Quadrupole Excitation Frequencies in a Linear Ion Trap. *J. Am. Soc. Mass Spectrom.* **2000**, *11* (11), 1016–1022. https://doi.org/10.1016/S1044-0305(00)00171-9.

(14) Zhao, X.; Ryjkov, V. L.; Schuessler, H. A. Parametric Excitations of Trapped Ions in a Linear Rf Ion Trap. *Phys. Rev. A* **2002**, *66* (6), 063414. https://doi.org/10.1103/PhysRevA.66.063414.

(15) Gotlib, Z. P.; Brabeck, G. F.; Reilly, P. T. A. Methodology and Characterization of Isolation and Preconcentration in a Gas-Filled Digital Linear Ion Guide. *Anal. Chem.* **2017**, *89* (7), 4287–4293. https://doi.org/10.1021/acs.analchem.7b00356.

(16) Roesch, G. C.; Garand, E. Tandem Mass-Selective Cryogenic Digital Ion Traps for Enhanced Cluster Formation. *J. Phys. Chem. A* **2023**, *127* (36), 7665–7672. https://doi.org/10.1021/acs.jpca.3c04706.

(17) Paradisi, C.; Todd, J. F. J.; Traldi, P.; Vettori, U. Boundary Effects and Collisional Activation in a Quadrupole Ion Trap. *Org. Mass Spectrom.* **1992**, *27* (3), 251–254. https://doi.org/10.1002/oms.1210270316.

(18) Konenkov, N. V.; Cousins, L. M.; Baranov, V. I.; Sudakov, M. Yu. Quadrupole Mass Filter Operation with Auxiliary Quadrupolar Excitation: Theory and Experiment. *Int. J. Mass Spectrom.* **2001**, *208* (1), 17–27. https://doi.org/10.1016/S1387-3806(01)00375-X.

(19) Soni, M. H.; Cooks, R. Graham. Selective Injection and Isolation of Ions in Quadrupole Ion Trap Mass Spectrometry Using Notched Waveforms Created Using the Inverse Fourier Transform. *Anal. Chem.* **1994**, *66* (15), 2488–2496. https://doi.org/10.1021/ac00087a013.

(20) Song, Q.; Smith, S. A.; Gao, L.; Xu, W.; Volný, M.; Ouyang, Z.; Cooks, R. G. Mass Selection of Ions from Beams Using Waveform Isolation in Radiofrequency Quadrupoles. *Anal. Chem.* **2009**, *81* (5), 1833–1840. https://doi.org/10.1021/ac802213p.

Chapter 5: User's Guide to the CD-LIT-MS

The intention of this chapter is to provide a practical description of how to operate the CD-LIT-MS for future users. If you intend to walk into Daniels 4309 and fire up this digital mass spectrometer, then this chapter was written with you in mind. *Godspeed*. A general overview of the instrument is given in section 5.1, followed by a detailed explanation of the steps to start and run the instrument in section 5.2. Next in section 5.3, the digital electronics which make square waves for the LIT are described. Section 5.4 gives a detailed explanation of the Arduino programs which are available to you to run the LIT, including their functionalities and tabulated variables and commands. Section 5.5 discusses the LabVIEW data acquisition software and user interface. In the final section 5.6, the overall instrument timing scheme is illustrated and discussed.

Section 5.1: General Overview

As it stands today, the CD-LIT-MS consists of five vacuum stages, shown in Figure 5.1 with approximate base pressures noted when the capillary is open. First there is an electrospray ionization (ESI) source, then a capillary and skimmer in Stage I, hexapole ion guides and apertures in Stages II-III, a LN_2-cooled octupole reaction trap (abbreviated RT) in Stage IV, and finally, the He cryostat-cooled LIT and detector in Stage V. Ions are mass-selectively ejected from the LIT at 90° from the entrance axis and detected by a channel electron multiplier (CEM) in Stage V.

All control voltages are set up for operation in positive mode, although in theory the instrument can be switched to negative mode with minimal adjustment. The DC voltage boxes are housed on the electronics rack. The RF boxes which control all the ion guides and the octupole trap are on the front upper shelf on the instrument, while the boxes which house the HV square

waves to drive the LIT are on the rear upper shelf. ***Before operating this instrument, please take note that the ±100 V, 3A Acopian power supply for the square waves is on the rear upper shelf and is highly dangerous.*** The HV switch which controls the power supply is a separate box labeled with a red sticker. A more thorough overview of the digital electronics on the CD-LIT-MS is provided in section 5.3.

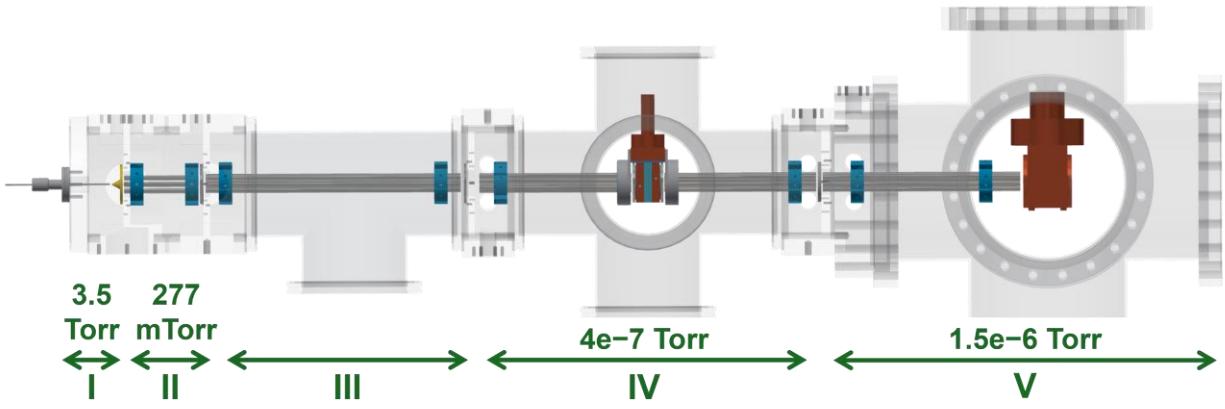


Figure 5.1 – Schematic of the CD-LIT-MS instrument (same figure as Chapter 1) with base pressures noted when the capillary is open. Note that these are not the operational pressures, since Helium gas is introduced into the ion traps in Stage IV and V as described in Section 5.2. The open pressure in Stage III is unknown because it bottoms out the current Pirani pressure gauge.

Section 5.2: Step-by-Step Operation

Optional Step 0. Cool down the octupole trap and LIT

If you intend to make solvent clusters in the octupole reaction trap or tag in the LIT, you will have to cool down both traps before performing the following steps to obtain a mass spectrum and an IR spectrum. The octupole trap is cooled via a Janis Research Co. LN_2 cryostat which you fill with liquid nitrogen via a funnel as is standard practice in the Garand group. This trap and cryostat used to be on the previous version of CIVS (c. 2023), so other group members can certainly assist with any issues you may encounter. It takes ~2 standard dewars of LN_2 to fill, and

you will see the excess come out of the rubber hose when filled. The LN_2 cryo is equipped with an RTD temperature sensor and cartridge heater which you control via Channel A on the Lake Shore 340 Temperature Controller on the upper front shelf.

The closed-cycle Helium cryostat on the LIT is slightly different from the other He cryostats in the group. The CIVS and SEVI main traps have Gifford-McMahon cryocoolers, while this one is a pulse tube cryocooler. The cold heads work differently for these two types, but the principles are similar, and the He compressors operate in the exact same manner. This He cryo is very uncomplicated to operate – you simply press the big green button on the compressor to start and stop it. The compressor makes a loud sound when it starts, so prepare yourself (and maybe others in 4309). If the compressor overheats or loses too much He pressure it will shut itself off, and there are reset switches on the lower part of the front panel which must be pressed before restarting (see the manual on the group server).

Always listen for the rhythmic cryo chirp, and if the chirping ever becomes inconsistent in tempo you should likely consult Etienne as it may need maintenance. Sometimes the chirp will sound inconsistent for a few minutes as it starts up, then will become a consistent tempo. One perk about this cryo is that it cools very fast, and will be down to the base temperature of 32 K in ~25 minutes. Channel C on the Lake Shore controller is connected to a thermocouple on the LIT housing which reads the temperature. I have found that this cryo typically gets all the way down to its base temperature (~32 K), but then very often goes up and settles around 60-70 K spontaneously. I do not currently have a solution for this problem, but it may be due to contamination within the cold head (hopefully you will be able to find a solution).

The LIT is also equipped with two 25 Ω, 100 W cartridge heaters (Lake Shore Cryotronics HTR-25-100) wired in parallel so that it can be heated to room temperature. Unfortunately, this

Lake Shore controller only has one heater, so you must heat one trap at a time. To heat the LIT, simply disconnect the 10-pin connector cable from the RT and connect to the LIT cold head. Turn the Lake Shore heater all the way up (50 W) because the LIT housing is large and will heat slowly, usually taking 3+ hours. Most of the time you can let both traps come to room temperature naturally overnight after a long day of cryo experiments, but the heaters do come in handy if de-icing is required when water clustering or if you wish to run the cryo higher than base temperature.

Step 1. Introduce Helium gas into both ion traps

The instrument is equipped with one gas manifold system which connects two separate He mixing bottles for the RT and LIT. I redesigned this manifold system in Spring 2025 so that there could be more control over introducing water into the RT without icing up the LIT, and there is the option to put different gases in the LIT for cryogenic tagging experiments. The RT bottle is connected to a pulsed valve, while the LIT bottle is connected to a precision leak valve outside the vacuum of the instrument. I recommend opening the LIT bottle first because the pressure takes some time to stabilize in Stage V.

Open the LIT regulator valve then slowly open the precision leak valve above it to allow gas into the LIT. The display reads 50 when the leak valve is closed as shown in Figure 5.2, and the numbers increase as it opens. I typically set the regulator valve to ~10 psi, however, the exact backing pressure is not of high importance since the leak valve allows for precise pressure control. Slowly open the leak valve using the knob on the right until the desired pressure is reached.

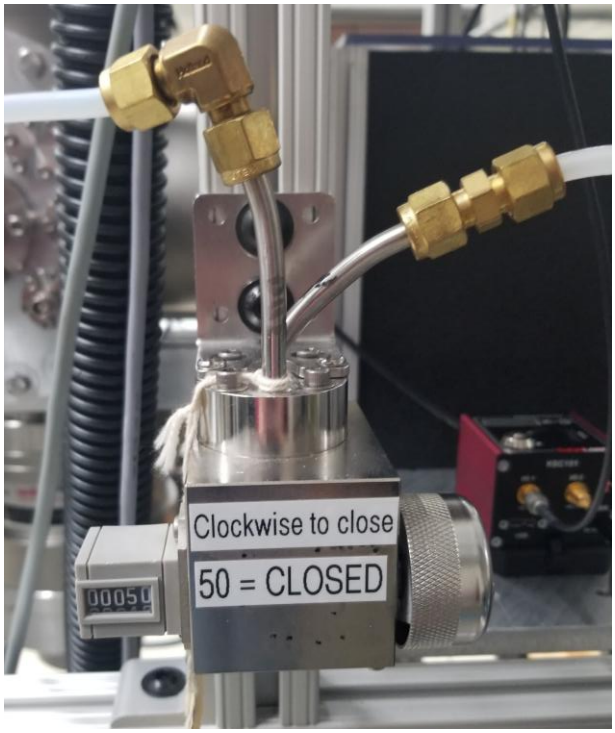


Figure 5.2 – Picture of the precision leak valve which allows gas into the LIT in Stage V.

Once you reach operational pressure in Stage V, I recommend waiting a couple minutes for it to equilibrate. I have found that the pressure always drops slightly after initially opening, and I must open the leak valve more before it stays at the intended pressure for the rest of my running time. The exact LIT pressure you want depends on the experiment you have planned, but I have operated the LIT at pressures from $\sim 5 \times 10^{-5}$ to $\sim 1 \times 10^{-4}$ Torr with comparable performance as a mass analyzer. Pressure control in the LIT is a more important consideration when performing resonant excitation experiments which can cause fragmentation at higher pressures ($> 5 \times 10^{-5}$ Torr).

Gas is introduced into the RT via a more standard Garand group solenoid pulsed valve. Next, open the regulator valve on the smaller RT bottle (also kept at ~10 psi) and adjust the pressure with the pulsed valve controller located at the top of the electronics rack. The pressure can be increased in Stage IV by adjusting the amplitude or width of the pulse. Note that there are two Stanford boxes which control the timing of the instrument, and the RT pulsed valve is currently triggered by the 10 Hz box on the bottom. I discovered that the RT must be run at 10 Hz or faster to make water clusters, and it does not affect the instrument under mass spec only operation. While the pulsed valve is run at 10 Hz, the RT exit pulse is triggered by a second Stanford box according to the slower cycle of LIT mass analysis (1-2 Hz). See section 5.6 for more details about instrument

timing. Operational pressures in Stage IV vary significantly based on your current trapping settings but typically stay in the range of $\sim 1 \times 10^{-6} - 7 \times 10^{-5}$ Torr.

Step 2. Start the Arduino and LabVIEW programs

The low voltage waveform generator (LVWFG) box sits on the upper rear shelf of the instrument and is usually left on 24/7 with the main Arduino program continuously running. It is my preference to leave this box on to maintain Arduino connection with the computer. This is not required; however, it is safe to do so since this is a low voltage circuit (see section 5.3). Therefore, I do not typically have to press "Run" on the Arduino program unless I am switching to a different program. If you do need to start the program, click "Run" and initiate serial communication with the Arduino. See section 5.4 for more specific instructions and details about the Arduino programs. Note that the Arduino program must be started before the LabVIEW data analysis program, since the data collection is triggered by the LVWFG.

Once the Arduino program is started, you can then start the LabVIEW data collection program by pressing the "Run" ► button in the ribbon at the top of the window. To stop the LabVIEW program, you must press the red "STOP" button *in the program interface*, that way it will correctly stop communication with the Picoscope. See Section 5.5 for more detailed information about the LabVIEW program, and a breakdown of all the important parameters in the user interface.

Step 3. Turn on electronics and electrospray (ESI)

The next step is to turn on all the driving electronics and set up the ESI. On the electronics rack you will see supplies labeled as the detector bias, capillary voltage, LIT exit plates, RT exit pulser, and two 10-output DC voltage boxes. There is one RF voltage box on the upper front shelf, then the square wave high voltage high frequency (HVHF) box and ±100 V Acopian supply and switch on the upper rear shelf.

Before turning on the detector bias, you should turn off the cold cathode pressure gauge in Stage V (Channel A1 on the Balzers pressure controller). This gauge causes an incredible amount of noise in the mass spectrum, which I believe is due to ions from the cold cathode discharge hitting the detector. For the time being while that gauge is in use, it must be turned off while the instrument is running and the backing Pirani pressure gauge for Stage V (B2 on Balzers) can be used to monitor any major pressure changes. Once that gauge is replaced with a newer one, I do not expect this to be an issue.

As opposed to other detectors in the Garand lab, the CEM detector in this instrument *should not be biased greater than -2 kV*, and I have found that a bias voltage of -1.5 to -1.8 kV is more than enough to get good ion signal. In fact, increasing the detector bias greater than -1.8 kV often results in over-broadening of the peaks in your mass spectrum. For the lifetime of the CEM, it is recommended to bias the detector at the lowest voltage magnitude while maintaining good signal. If you start to notice that the detector bias must be greater and greater in magnitude to get the same ion signal, this could be an indicator of detector contamination or failure.

I highly recommend turning on the Acopian last, since it is the most dangerous supply, and the user should be cognizant of when it is turned on. If you are not actively running the instrument and must leave the room for a significant time, I would also recommend turning off the Acopian

to prevent any accidental encounters. A simple check to know if the ±100 V supply is on and the HV square waves are working correctly is by looking at the baseline noise on the raw data output in LabVIEW. You should see the noise increase significantly when you switch on the Acopian. This is why I typically turn on the LabVIEW program before all the voltages, so that I can see that baseline increase.

The instrument is also equipped with a commercial 100x bipolar amplifier which lies in a small box near the detector flange (left side of the instrument when you face the inlet). This box is kept along with the Picoscope in this specific location to limit the length of the BNC cables and reduce the amount of noise that it picks up from the HV square wave drivers. The amplifier is typically left on 24/7, but if you ever observe a completely flat baseline in the LabVIEW program, you should certainly check to make sure this box is switched on. Finally, once all the voltages are turned on, you can set up the ESI syringe and turn on the syringe pump (~2.5-3.5 μL/hr) and ESI voltage (~2-4 kV).

Optional Step 4. Make water clusters in the RT

The primary proof-of-concept experiment I have performed in the CD-LIT-MS is water clustering on small polypeptides and amino acids (results shown in Chapter 6). The RT on this instrument came from CIVS and has proven to be an excellent vessel for water clustering. To water cluster, you must seed the RT He buffer gas with water vapor and make sure the RT pulsed valve is operating at 10 Hz as described in Step 1. I have discovered that a 10 Hz or faster gas pulse is crucial to create water clustering conditions, and a slower pulse will not result in cluster formation. Therefore, even though the LIT and laser operation are slower, the RT must remain at 10 Hz for this experiment. You can pulse faster than 10 Hz, but I have found little benefit in doing so. RT

pressures of roughly $3 \times 10^{-5} - 7 \times 10^{-5}$ Torr are ideal for clustering, and pulsing faster can result in pressures that are too high. Additionally, letting multiple pulses of water clusters into the LIT tends to result in very broad peaks in the mass spectrum and more difficulty in selectively filtering them, likely due to the broader kinetic energy distribution and additional space charge.

To seed the RT buffer gas with water, first attach a tube (made by Etienne and shared with CIVS) full of Milli-Q water to the gas manifold. I typically disconnect the green N_2 line and attach it there where it can be in full view. Next, vent the RT mixing bottle if Helium is still present and then vacuum it out fully with the manifold pump and close the valve on the bottle. To get water vapor into the manifold, pull vacuum on the water tube by opening the valve briefly ~5 times. You should see the water bubble in the tube as you vacuum on it, and the level of water should decrease significantly in the tube by the time you are done. Open the valve a few more times if this is not the case. Then close the valve to the manifold pump and turn the pump off. Open the valves from the water tube to the RT mixing bottle so that it fills with water vapor. Due to the very small amount of vapor, you unfortunately will not see any pressure change on the regulator so at this point you must trust that it is in there (if water did not get in it will become clear when you run, however, this process has been pretty foolproof so far). Finally, you will fill the rest of the mixing bottle with Helium and run as usual.

Optional Step 5. Mass filter or isolate using square parametric excitation

Another optional step in the CD-LIT-MS workflow is mass filtration or isolation using the RF square wave duty cycle and frequency and/or square parametric excitation. In section 5.4, I describe the different Arduino programs which are available to you for these operations. If you

have a very crowded mass spectrum or want to remove any untagged or unwanted species, you can use these techniques to do so.

During this step, you will need to do some optimization of whichever technique you choose to find the correct settings. For example, if you want to use square parametric excitation to resonantly eliminate one m/z, you can use my Python program to calculate the parametric frequency as described in Chapter 3. However, math and real life often do not line up exactly, so adjustments will need to be made to find the best possible elimination conditions in the LIT. Make sure that as you optimize, you record all instrument settings (concentrations, ESI, pressures, etc.) because all these things need to be very similar to have repeatable mass filtration results.

(Future) Step 6. Obtain an IR spectrum

This step of the instrument cycle is the only unfinished step, since I was not able to obtain an IR spectrum from the CD-LIT-MS at the end of my PhD. Alas, instrument maintenance issues and life delayed things, as it often does. However, everything is physically set up for *you* to do spectroscopy. The second IR laser from CIVS is aligned into the LIT, such that CIVS and this instrument can run at the same time, unless CIVS is doing their dual-laser experiment. We did shoot the laser into the LIT and see photofragmentation, so the next steps will be to optimize this and set up the LabVIEW data analysis program to record IR spectra.

The small optics box on the rear shelf of the instrument contains a shutter, turning and focusing mirror, and KBr window on the back flange. The shutter is used to block laser shots, since the CD-LIT-MS cannot run as fast as the 10 Hz laser (see Section 5.7 for details on instrument timing). The solenoid valve controller for the shutter is also located on the back shelf right outside

the optics box and must be switched on before running. In order to take IR spectra like CIVS does, their LabVIEW program which integrates the mass spectrum will have to be integrated into the LabVIEW program for this instrument as well.

Final Step. Shut down the instrument

To shut down the instrument, you can generally perform the start-up tasks in reverse order. Turn off the ESI voltage and syringe pump, then make sure to turn off the detector and Acopian power supply before turning the Stage V cold cathode pressure gauge back on. Turn the amplitude of the RT pulsed valve down to zero and close the LIT leak valve to 50 before also closing both regulator valves. Then you can turn off all the electronics and cover the capillary entrance. The closed-cycle He cryostat on Stage V can be shut off simply by pressing the big green button on the compressor – the same one you pressed to turn it on.

Section 5.3: Digital Circuitry

Here I will provide a brief description of the digital electronics which run the LIT. The same circuit boards are present on CIVS and described in Gina Roesch's 2023 thesis,[1] however, there is a unique setup in the DC-LIT-MS for parametric resonant excitation. There are 2 boxes containing the square wave circuits, one which produces the low voltage TTL signals, and one which amplifies that signal to 100 V_{pp} and creates the complex waveform for resonant excitation. All the circuits on this instrument have worked reliably for years, but I will also comment in this section on issues you could encounter and the recommended troubleshooting steps. Note that you can find the printed circuit board (PCB) designs for all these circuits in the Garand group Box

folder titled Electronic Circuits, and even more documentation for electronics testing in Microsoft Teams (once again, thanks to the wonderful Gina Roesch).

Low Voltage Waveform Generator (LVWFG)

The LVWFG is a waveform generator based on designs by Hoffman et. al.[2] which uses Direct Digital Synthesis (DDS) modules to create TTL level (0-5 V) square waves with adjustable frequencies and duty cycles. This generator has 3 DDS modules (labeled 0,1, and 2) which each produce an independent frequency-adjustable sine wave. Each DDS module then sends this sine wave through two comparators, outputting two square waves which are 180° out-of-phase with ease other. In each comparator, the sine wave signal is compared to a DC reference signal from its own Digital-to-Analog Converter (DAC), producing a set of high and low DC pulses with adjustable lengths. This allows the user to adjust the duty cycle of each square wave independently. The entire LVWFG is controlled by an Arduino DUE microcontroller by serial communication in the Arduino IDE interface (software control is described in the next section, 5.4). A block diagram of the LVWFG is shown in Figure 5.3.

The LVWFG for this instrument has 3 operational DDS modules but only uses two to run the LIT (DDS 0 and 1). DDS 1 produces the primary X and Y waveforms for trapping in the LIT, while DDS 0 produces the auxiliary square wave signal for resonant excitation. You will see that the quartz clock oscillators for DDS 0 and 1 are wired to each other to keep the base and auxiliary waveforms as in-phase as possible. This is the only difference between my LVWFG, and the others present in our lab. Note that frequency control happens in the DDS modules, but duty cycle control happens by changing the DC reference from the DACs. There is a naming convention for each of the DDS modules and their corresponding DACs which will become important when you look at

the Arduino code to adjust frequency and duty cycle. The naming convention is as follows: DDS 0 is connected to DACs 0 and 1, DDS 1 to DACs 2 and 3, and DDS 2 to DACs 4 and 5.

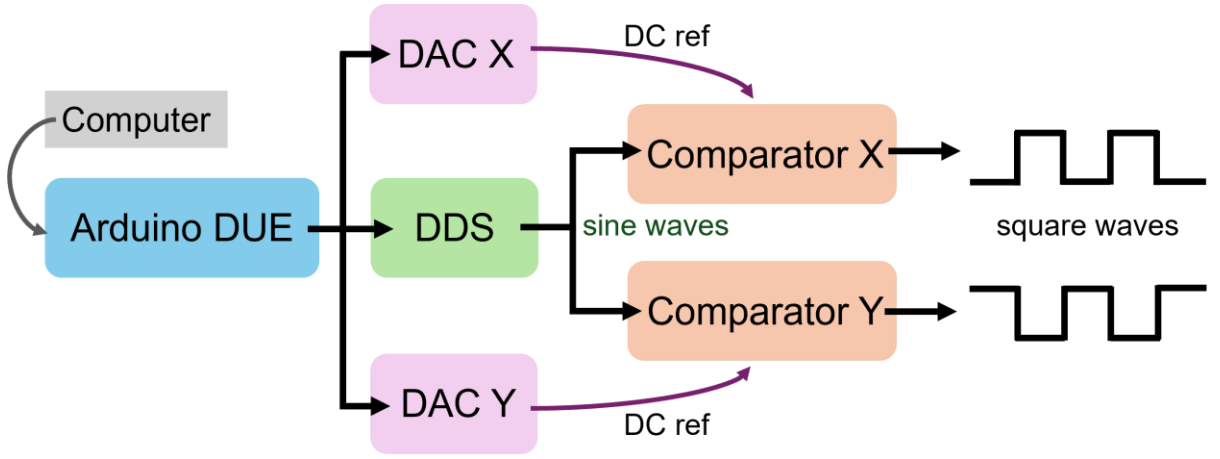


Figure 5.3 – Block diagram of one output channel on the LVWFG. The Arduino controls all the DDS modules, and each DDS module controls the square wave frequency. Each DDS module has two DACs and comparators which each output a 180° out-of-phase square wave.

The other TTL signals in use from the LVWFG are labeled "SCAN", "TRIG", and "Signal 53" on the PCB and exterior of the box. "SCAN" and "TRIG" are used as triggers for the LabVIEW data collection, which is described further in Section 5.5. "Signal 53" is used to trigger the LVWFG with a Stanford box as the master timer. The LVWFG is currently externally triggered by a Stanford box so that it can be interfaced with the 10 Hz pulsed IR laser. Note that in the past, the LVWFG was the master timer and therefore internally triggered, so Signal 53 was not in use. See section 5.4 for a description of which Arduino programs use which triggering scheme, and section 5.6 for an overview of instrument timing. Finally, an additional trigger called Signal 42 on the PCB is set up to trigger the divide-by-N circuit designed by Casey,[3] but that circuit is not currently in use. A 50 Ω BNC terminator is kept on the front of the box where Signal 42 would be connected.

The LVWFG is a robust circuit which does not tend to have issues, so I do not anticipate that you will need to do much troubleshooting. In fact, you may remember from earlier that I leave this box running 24/7 since it is very safe and easier to maintain communication between the Arduino and computer. In any case, it is important to note that many of the parts on this circuit are surface mounted, and if those fail, they are nearly impossible to replace yourself. The Arduino DUE and DDS modules can be removed and replaced, and the DDS 2 channel is currently open and operational in case you have an unfixable issue with one of the other channels, or if you want to make even more square waves.

High Voltage High Frequency (HVHF)

The TTL-level square waves which come from the LVWFG are then sent into the HVHF box for amplification. The red HVHF boards have two sides, one for the positive half of the square wave and one for the negative half. There are a series of high-voltage MOSFET switches on either side that switch on and off in accordance with the input TTL. This HVHF design is based on Hoffman et. al.[4] with modifications to the MOSFETs to be able to handle higher current and voltage. Therefore, the red HVHF boards for driving this ion trap (see Figure 5.5) look slightly different than the green PCBs around lab which are used for other pulsing applications (such as pulsing the DC voltage on an ion trap exit).

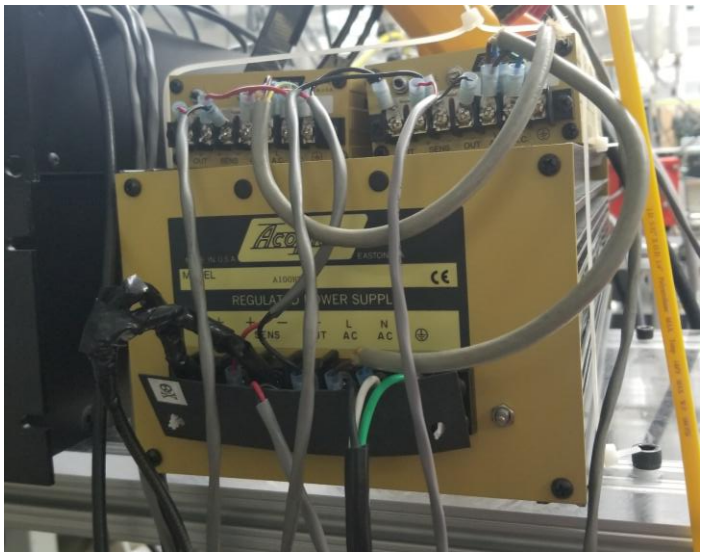


Figure 5.4 – Picture of the Acopian setup for the LIT. The big Acopian on the bottom is the ±100 V, 3 A supply with two 0-7 V supplies on the top for the auxiliary waveform.

The high voltage connections are at the back of each HVHF board for both the positive and negative halves, and these voltages come from the Acopian power supply shown in Figure 5.4. There are two resistors on the main Acopian output acting as a voltage divider (the part wrapped in black electrical tape) to split it into approximately -50 and +50 V. Note that the actual voltages are not as such since there is a DC float voltage for the LIT, however, these numbers are easy to use as illustration and represent the 100 V being divided evenly.

On top of the large Acopian are two adjustable +/- 0-7 V Acopian power supplies which are used for the complex square waveform generation. These are the mini Acopians shown in Figure 5.4. Let's assume for a moment that the little adjustable Acopians are set to -5 V and +5 V. Then there would be 4 distinct DC voltages generated: V++ (+55 V), V+− (+50 V), V −+ (−50 V), and V−− (−55 V). This notation is used to designate the 4 voltages and can be a little confusing, so take extra care to make sure you are connecting the right BNCs to the back of the HVHF box.

The HVHF box for this instrument contains 4 boards: one for the Y base waveform, one for the X base waveform, one to produce the top half of the X complex waveform, and one to produce the bottom half of the X complex waveform. To run a quadrupole ion trap or guide under typical conditions, you would only need two HVHF boards: one for the X rods and one for the Y rods. The HVHF drivers on CIVS have this setup for each digital trap. The Y rods of the LIT are also setup in this typical way, with one HVHF board driven by V+− and V−+ to produce the driving square wave with a V_{pp} of ~100 V. For the LIT, I made two additional boards so that there could be a complex waveform for resonant excitation on the X rods (the one described in Chapter 3).

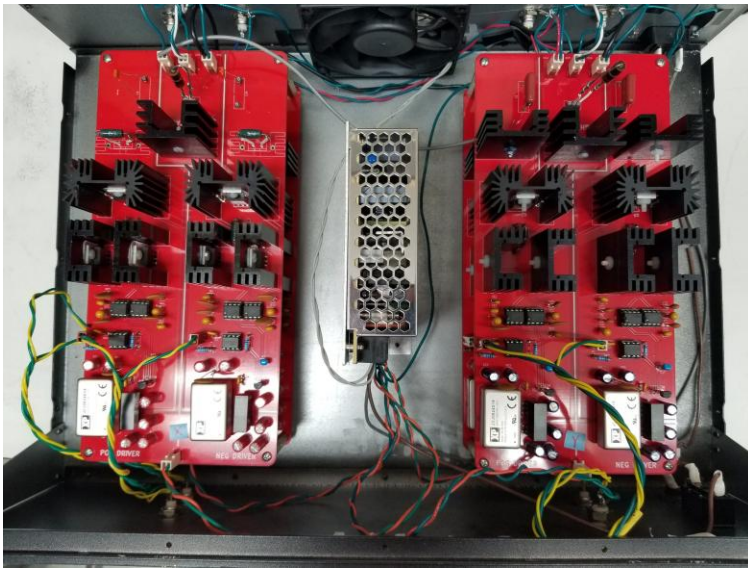


Figure 5.5 – Picture of the HVHF box for the LIT, which contains 4 red HVHF boards. The top of the photo is the back of the box, where the HV power entry is located. The bottom of the photo is the front of the box, where the square wave TTLs are input from the LVWFG.

The auxiliary waveform TTL comes from the LVWFG and triggers both bonus HVHF boards, one which has the high voltage inputs V++ and V+− and the other which has inputs V−+ and V−−. You can think of this as one board for the top half of the auxiliary component (+50 to

+55) and the other board for the bottom half (-50 to -55). The output of these additional boards is then wired to the high voltage connections at the back of the main X board, where the top (V++/V+−) output backs the positive side and the bottom (V−+/V−−) backs the negative side. This process is illustrated in Figure 5.6 and results in a square waveform with two frequency components – the high driving frequency which comes from the X TTL (DDS 1 DAC 2), and the lower auxiliary frequency coming from the auxiliary TTL (DDS 0 DAC 1).

There are several important operational notes and pitfalls for the HVHF boards. First, it is very important that you never switch the positive and negative high voltages on the back. The MOSFETs on each side are specifically designed for either polarity, so they cannot handle the opposite. If this does occur (and yes, I have done it before), it will likely blow the tuning resistors which are right after the high voltage entry and potentially some MOSFETs as well.

To troubleshoot, first remove the tuning resistors and measure the resistance, since they are connected to the board via socket connectors and do not need to be de-soldered. When these resistors break, they typically measure infinite resistance. If the resistors appear okay and the square waves are still not working, inspect the MOSFETs on the board to see if any of them are burnt. You should see (or smell) signs of melted plastic or the glue which connects the MOSFET to the heat shield will be burnt and no longer white in color. You can also beep test the MOSFET to see if the outer two legs are shorted, which is a common way that they fail. Look up the pin diagram for that specific MOSFET to make sure it is correctly connected and not shorted. If a MOSFET needs to be replaced, it is not the end of the world – you can do that by carefully de-

soldering and soldering on a new one. You can find further documentation on troubleshooting HVHF boards on the Garand group Microsoft Teams channel entitled “Things we should know”.

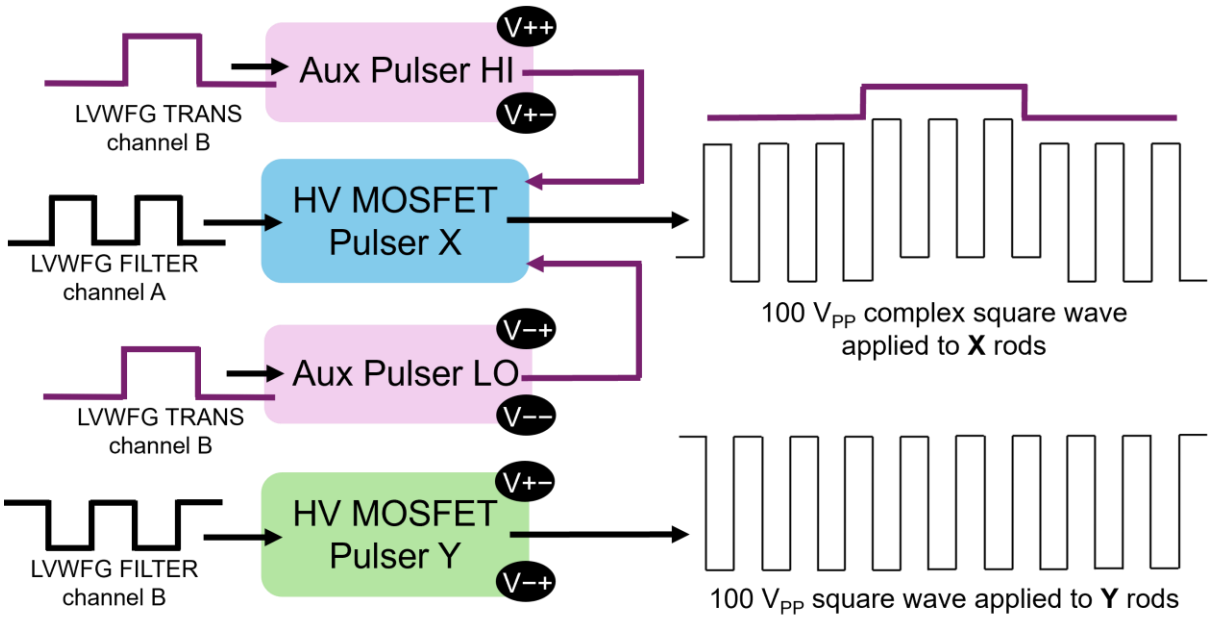


Figure 5.6 – Block diagram of 4-board HVHF setup for the LIT, with the HV inputs labeled as they are on the instrument. V+− and V−+ are the split outputs from the 100 V Acopian supply, while V++ and V—are those outputs plus or minus the auxiliary amplitude from the adjustable 0-7 V supplies, respectively. Purple trace represents the superimposed auxiliary frequency component of the complex waveform.

The other crucial pitfall to avoid with HVHF boards is inputting a too-high frequency. The MOSFETs are designed to withstand up to 700 kHz, but usually they cannot quite reach that value. With these specific boards, you cannot go above 675 kHz without seeing the square waves start to cut out, indicating that a MOSFET is close to failure. Note that there are protections in the Arduino code that should prevent the LVWFG from going above 675 kHz, and those are intentional so that the user cannot accidentally input too high of a frequency.

Finally, I want to note that the 5-Watt tuning resistors on the back of each HVHF board are important. These resistors are present to reduce ringing noise on the square waves and are mounted in sockets so that they can be readily adjustable. The tuning resistance value was optimized to match the capacitance of the ion trap and chosen to reduce ringing noise without rounding the square wave excessively. For the LIT, I have found that a 80 Ω resistance is ideal. However, you'll see that the two boards for the top and bottom of the complex waveform have different resistance and capacitance values on the back, which I adjusted by trial and error to reduce dampening of those smaller amplitude (5 V_{pp}) waveforms. If you are having issues with the square waves cutting out, it is a good idea to check that these removable components are still fully inserted into their sockets. Also note that when you look at the square wave output *off* the instrument, the ringing noise will look a lot worse since that 80 Ω resistance is matched to the trap. To get nice oscilloscope images off the instrument, I typically put a higher value tuning resistor (on the order of 500 Ω) on the back of the main X or Y outputs.

Section 5.4: Arduino Programs and LIT Modes

Over the years, I have modified and produced many versions of the Arduino code which runs this instrument. The important versions are compiled in the Garand group server folder entitled "Arduino Code Final Versions" for you to access. Here I will overview what you should know about the Arduino code and how serial communication with the LVWFG works. I will outline what Arduino code file you need for the different MS-only experiments I have done in the LIT: mass analysis only, mass isolation by frequency and duty cycle switch combined with square parametric excitation, selective mass elimination by square parametric excitation, and parametric resonance-enhanced mass scan. Finally, I will talk about the Arduino code for selective mass

elimination of multiple m/z species which was modified to interface with the tunable IR laser system and therefore has a different triggering scheme.

Serial Communication and Interval Code

All communication with the LVWFG occurs within the serial monitor in the Arduino IDE using a series of commands. A screenshot of the Arduino serial monitor is shown in Figure 5.7. To adjust variables while running, you type a command into the serial monitor followed by your selected numerical value. These commands vary across the different codes, but there are some which are ubiquitous since every code must contain the trapping cycle and the frequency scan for mass-selective ejection. In between the trapping time and mass scan one can do many things (mass elimination or isolation, shoot a laser in there, etc.), which necessitates different codes for different experiments. Table 5.1 contains the file names of the final versions of each code I have modified along with a brief description. Before running, make sure the baud rate (the speed of communication with the Arduino) is set to 115200, because that is the rate required for the Arduino Due in the LVWFG.

The main body of each program is in the Interval Code section, ~2590 lines down. Everything before the interval loop is there to initialize important things like serial communication, variables, commands, and functions that you need. Note that this code originated from Pete Reilly's group at Washington State and was modified by Garand group students before me. Therefore, there are certainly some variables/commands/etc. which are initialized before the intervals and unused in the body of the code. Here I will describe the function of the interval code and tabulate the important variables and commands that you should know as the user. You can find screenshots of the interval code for each program in Appendix section A6.

Table 5.1 – File names of each Arduino code, with a brief description of the function and how the trigger is currently set up in the code.

	File Name	Description	Trigger Setup
A	*2021-7-20-LIT_Mass_Scan_Only*	Basic program to obtain a mass spectrum, no mass filtering	SCAN output triggers Stanford box
B	*2024-5-30-LIT_Parametric_Mass_Isolation*	Mass isolation by combination frequency/duty cycle switch and notched parametric frequency scan	SCAN output triggers Stanford box
C	*2023-7-3-LIT_Parametric_Mass_Elimination*	Selective mass elimination of one m/z species by square parametric excitation	SCAN output triggers Stanford box
D	*2023-10-24-LIT_Parametric_Mass_Scan*	Parametric resonance-enhanced mass scan by scanning the auxiliary frequency along with base frequency	SCAN output triggers Stanford box
E	*2025-10-8-ParaMassElimination_3regions*	Mass elimination of 3 m/z species by square parametric excitation, with triggers set up for 10 Hz laser	Stanford box triggers the LVWFG on Signal 53 input

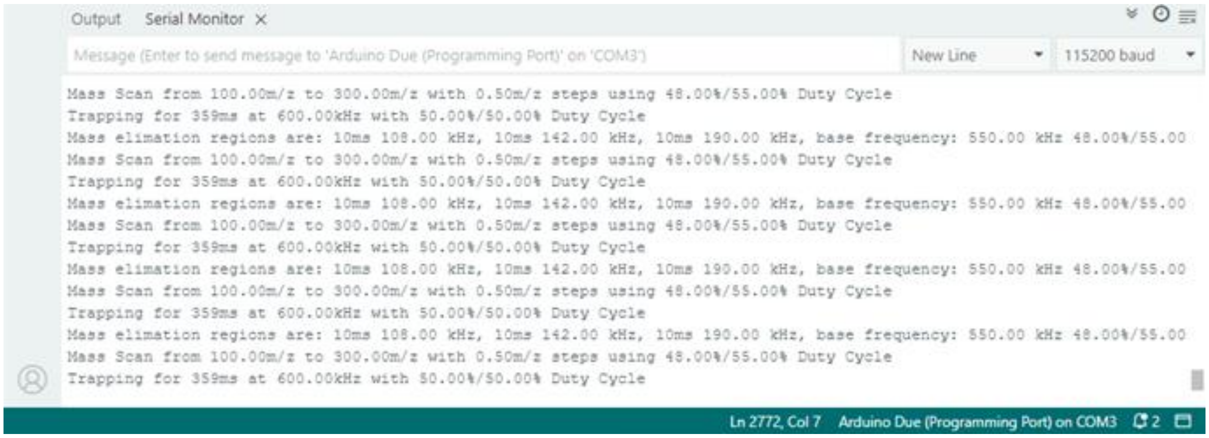


Figure 5.7 – Arduino serial monitor output for program E from Table 5.1. Note the 115200 baud rate in the upper right-hand corner.

When you get to the interval section of every code, you will see intervals 0, 1, 2, 3, and 4. You can think of the intervals as steps in the overall instrument cycle, which runs continuously in a loop. Interval 1 is the first interval of the instrument cycle, so it starts the serial communication and checks to see if you have input any commands. Therefore, to start every Arduino program, you enter the command “i1” into the serial monitor to tell the program to enter interval 1. Interval 0 exits you out of the loop, so if you want to stop the program from running you enter “i0”. These commands and all others related to general trapping and mass analysis are shown in Table 5.2.

After interval 1, the next step is trapping in interval 2. Interval 2 initializes the frequency and duty cycle of the square waves to stable trapping conditions. I typically have them set to 600 kHz, 50:50 duty cycles, but this is something you can adjust using the commands in Table 5.2. For example, there are occasions when you might want to lower the trapping frequency and therefore create a high mass pass filter as ions enter the LIT. If you dramatically change the mass range of analytes, it may also be advantageous to adjust the trapping conditions. These adjustments will likely be slight, but you can do this while running in the loop. Note that in the serial monitor, you

input duty cycles as decimal fractions (0.50 = 50%) and what you input is the LOW half of the waveform. This is something that you can change in the code, but I kept it the same for consistency when you see reported duty cycles in my notebook. After switching to stable square wave conditions, then there is a time delay (W0) to allow for collisional cooling and trapping.

Interval 3 is next and is an optional step that is commented out in some of the codes. When present, interval 3 is used for duty cycle filtering or parametric mass elimination or isolation. More details about this interval are given below for programs B-E. The main thing you need to know is that you can take any Arduino code and simply swap out interval 3 to do one of the types of mass filtering I have described here in this thesis.

Interval 4 is the final interval of the instrument cycle and contains the code for the downward stepped frequency scan which mass-selectively ejects all the ions. This interval is the same for all codes except D, which has a modified scan with an additional parametric frequency component, described more in detail below. At the beginning of interval 4, you will see that several temporary variables are initialized for use during the frequency scan. Next there is an if-else statement to protect the circuitry from failure. Since the HVHF boards have a strict upper frequency limit as described in section 5.3, the low mass limit is calculated from that user-input value. If the input m/z is lower than the limit, then the program will automatically exit the loop. This is a failsafe in case you accidentally input a m/z which is too small and will probably fry a MOSFET or two. In the "else" statement is a loop which calculates all the frequency steps. The "qCalcMulti" function calculates each frequency value from the user input m/z and other variables using the rearranged Mathieu q equation, shown below.

$$\Omega\,(Hz) = \frac{1}{2\pi}\sqrt{\frac{4eV_{RF}}{mr_0^2 q}} \tag{5.1}$$

As the user, you must input the start m/z, stop m/z, delta m/z, Mathieu q value, trap radius r_0, and square wave zero-to-peak voltage V_{RF}. The commands, variable names, and units for these parameters are shown in Table 5.2. Once the frequency is calculated, the LVWFG must sit at this frequency for a delay time to excite and eject the ions of that m/z. The length of each step is calculated from a user-selected number of periods, default set to 100 periods per step (command W5). Inside the scan loop, the current m/z is incremented by the chosen delta m/z so it will calculate the next frequency step until the stop m/z is reached. Note that you can also see the "SCAN" trigger go HIGH at the beginning of the loop to signal data collection for the Pico, and the "TRIG" signal go high at the beginning of each frequency step to make the frequency bins for data analysis (LabVIEW data analysis described in section 5.5).

Changing the user-input variables for q, trap radius r_0, and square wave zero-to-peak voltage V_{RF} will adjust the frequency landscape of the scan and subsequently the mass calibration. Although the trap radius and zero-to-peak voltage are known, slight adjustments can make a big difference in calibration. Additionally, the Mathieu q value of ejection is less definitively known and will change depending on duty cycle. If you change the duty cycle during the mass scan, I recommend looking at the Mathieu a-q diagram for that duty cycle combo (perhaps using my Python program) to determine where the q value should approximately lie. Remember that ions are ejecting out the X-axis of the LIT, so they are crossing the rightmost $\beta = 1$ boundary. The value of q at that boundary close to $a = 0$ is the "qCliff" that ions fall off, and the q which you should select and input into this program.

Table 5.2 – Serial commands for the trapping and mass scan intervals along with their corresponding variable name in the code, units, and functions. These are the only commands required for program A. Programs B-E also use these commands for the mass scan, plus those shown in Tables 5.3-5.6.

Serial Command	Variable Name	Unit	Function
i0	interval0	N/A	Exits the loop
i1	interval1	N/A	Enters the loop
W0:	trapDuration	milliseconds	Length of trapping time
r0:	trapFreq	Hertz	Trapping frequency
)6:	DACTrapDuty2	Decimal fraction LOW half	X duty cycle during trapping
)7:	DACTrapDuty3	Decimal fraction LOW half	Y duty cycle during trapping
%	scanStartMass	m/z	Starting m/z for mass scan
^	scanStopMass	m/z	Ending m/z for mass scan
&	scanMassDelta	m/z	Step size for mass scan
)2:	DACScanDuty2	Decimal fraction LOW half	X duty cycle during mass scan
)3:	DACScanDuty3	Decimal fraction LOW half	Y duty cycle during mass scan
)13:	qVoltsFilter	unitless	Mathieu q parameter for the frequency calculation
@2:	qVoltsFilter	Volts	Zero-to-peak voltage of square waves
#2:	qRodSizeFilter	meters	Trap radius r_0
W5:	triggerUpPeriods	# of periods	Number of periods the square waves sit at each frequency in scan
s	N/A	N/A	Prints the status of parameters
i7	interval7	N/A	Prints the values of all frequency steps in scan

I found that the on-the-fly mass calibration by adjusting q, r_0, and V_{RF} can be rather tedious and sometimes inaccurate, especially when you look at wider mass ranges. This may be because of the effects of pressure and space charge, which the Mathieu equation does not take into account. Therefore, for publication-level data I tend to fit the frequency vs. m/z relationship to a function using a spectrum of known m/z ions to produce more reliable mass calibration results. If you want to do this, you need to have a list of all the frequencies in your current mass scan, and you can get that from interval 7. The "i7" command will stop the program and print out all the frequencies under the current settings in the serial monitor, that way you can copy them and do your own mass calibration in Origin if you so desire. Intervals 5 and 6 also exist in many of the Arduino programs but are left over from the initial code and not in use for this instrument.

B. Mass isolation by combined frequency/duty cycle switch and notched parametric frequency scan

Program B is a code that you should select if you want to isolate one m/z or a small range and filter out everything else. This interval has two steps: a duty cycle and frequency switch (known as apex isolation or DAWI), followed by a notched scan of the parametric frequency for more precise isolation. If you simply want to use the frequency and duty cycle switch alone for rough isolation, and you can use this code and comment out the parametric scan section. The commands for code B interval 3 are shown in Table 5.3, starting with the frequency (r1:), duty cycles (n, N), and duration (b) of the apex isolation switch.

The rest of the commands in Table 5.3 are related to the notched scan of parametric frequency. You will notice some similarities between the code for this scan and the stepwise frequency scan for mass-selective ejection, as they are done in a similar manner. You select the

start frequency (<), stop frequency (>), delta frequency (M), and step length (W2:) for the stepwise parametric scan. The scan goes from high to low frequency (low to high m/z), therefore the delta value is subtracted at each step. Then you can select the location of a “notch” in the scan where the parametric frequency signal will turn off. This allows the ion(s) inside the notch to be preserved. The notch is created by selecting a frequency center (r3:) and width (r4:) which will be dependent on the m/z range of peaks you are trying to isolate. The final variable is the base frequency (r2:) which remains constant throughout the scan.

You can calculate a parametric frequency center for your m/z ion of interest using my Python program described in Chapter 3, but some experimental optimization of the center and width of the notch will be necessary. Complete isolation by only parametric frequency scan is also possible, if you do not want to do an apex isolation step beforehand. However, this is primarily useful when you have a small mass range, since scanning parametric frequency is a lengthier process than the apex switch. For a wide mass range, the combination of apex isolation first followed by parametric scan to further isolate one m/z will save you time.

Table 5.3 – Serial commands for interval 3 of program B, which is used for mass isolation by 1) apex isolation by frequency and duty cycle switch, followed by 2) notched scan of the parametric frequency for more precise isolation.

Serial Command	Variable Name	Unit	Function
r1:	filterFreq	Hertz	Frequency for apex isolation step
n	DACFilterDuty0	Decimal fraction LOW half	X duty cycle for apex isolation step
N	DACFilterDuty1	Decimal fraction LOW half	Y duty cycle for apex isolation step
W1:	filterDuration	milliseconds	Duration of apex isolation step
<	scanStartFreq	Hertz	Starting frequency of parametric scan (high)
>	scanStopFreq	Hertz	Ending frequency of parametric scan (low)
M	scanDeltaFreq	Hertz	Step size for parametric frequency scan
r3:	isolatedFreq	Hertz	Center of the notch where isolated m/z must be
r4:	isolatedRange	Hertz	Width of the notch
W2:	scanRate	microseconds	Length of each frequency step in the scan
r2:	baseFreq	Hertz	Base frequency during the parametric scan

C. Selective mass elimination of one m/z

Program C is the most basic program for selective mass elimination using square parametric excitation. If you want to selectively filter out one m/z in your mass spectrum, this is the interval code you want to use. Interval 3 of program C contains one filtering region where you have control over the parametric frequency, base frequency, base duty cycles, and filter duration. These commands are tabulated in Table 5.4. There is also a duty cycle for the auxiliary waveform which can be adjusted, but I have not included those commands in Table 5.4 since adjusting it is not particularly useful. After trapping, the program will switch to these conditions for mass

elimination. To filter out multiple m/z species, you simply need more filtering regions of this same structure to be added, and additional variables set up for them. Program E does this multi-elimination with three regions and a different triggering scheme for the laser interface.

Table 5.4 – Serial commands for interval 3 of program C, which is used for selective mass elimination of one m/z species by square parametric excitation.

Serial Command	Variable Name	Unit	Function
r1:	filterFreq	Hertz	Base frequency during elimination
r2:	auxFreq	Hertz	Parametric auxiliary frequency during elimination
)0:	DACScanDuty0	Decimal fraction LOW half	X base waveform duty cycle during elimination
)1:	DACScanDuty1	Decimal fraction LOW half	Y base waveform duty cycle during elimination
W1:	filterDuration	milliseconds	Length of elimination interval

D. Parametric resonance-enhanced mass scan

Program D is different from the others described here because there is no interval 3 mass filtration, just a modified mass scan in interval 4. The mass-selective instability scan of square wave frequency is kept the same and a linear scan of the auxiliary frequency is added to it for resolution enhancement. I discovered that for a narrow mass range such as 100-300 m/z, one can scan the auxiliary frequency linearly with respect to m/z or base frequency and see positive effects upon resolution and signal intensity. The actual mathematical relationship is non-linear, but this approximation allows for simpler implementation. This interval 4 code takes the linear equation shown below and calculates the auxiliary frequency at each step with respect to m/z.

$$auxiliary\ frequency\ (Hz) = base\ frequency(Hz) * [A - B(m/z)] \qquad (5.2)$$

A and B are experimental parameters which I arrived at by fitting the plotted relationship between parametric frequency vs. m/z for a given mass range, followed by experimental optimization. In the case of 100-300 m/z (my favorite mass range during this PhD), A = 1.2 and B = 0.00215 appeared to be the best fit which showed clear enhancement results. These terms are the only additional adjustable variables in the code, shown in Table 5.5.

Table 5.5 – Additional serial commands for interval 4 of program D, which is used for a parametric resonance-enhanced mass scan. These commands are in addition to those in Table 5.3 for the typical mass-selective instability scan of base RF frequency.

Serial Command	Variable Name	Unit	Function
d1:	Div1	N/A	Fit parameter A from Eqn. 5.2
d2:	Div2	N/A	Fit parameter B from Eqn. 5.2

E. Selective mass elimination of 3+ species with laser timing

Program E is a code for selective mass elimination of multiple m/z species. It is set up to be externally triggered by a Stanford box so that there can be consistent timing between the laser pulse, shutter, and LIT instrument cycle. This code is triggered on Signal 53 as the input similarly to Gina Roesch's code for CIVS. The main difference is the trigger starts at interval 3, right before the mass elimination and laser shot. More explanation about this instrument timing is in section 5.6. There are a few extra lines of code in the set up before interval code to make this triggering happen, which do not currently exist in programs A-D. Therefore, if you are keeping the current laser timing scheme and want to use another mass filtering function than what is currently available in this interval, I would recommend starting with this program and switching out the interval 3 code. Note that you will have to make sure all the proper variables and commands are initialized, since I did reuse some variables and commands throughout B-D.

Table 5.6 contains the commands and variables for this program. The premise is the same as the selective mass elimination in program C, simply with three distinct filtering regions so that you can eliminate three separate m/z peaks. This is an ideal filtering scheme for spectroscopically probing every other water cluster, or up to three untagged species in a mixture. For each region, you select the parametric frequency (r2, r3, r4) and duration (W2, W3, W4). You also select the base frequency (r1), which is currently hard coded to be same for all three regions (you can of course change this, just need more variables/commands). If you need more than three elimination regions, all you must do is copy the same code and make more variables and commands. After the mass elimination, there is one more delay term (W7) where the program switches to stable square wave conditions (the same as trapping interval 2) for the moment of laser interaction. After that, the program transitions to interval 4 for the mass scan per usual.

One of the important user's notes I have for this program is that I did not code a command which turns the auxiliary frequency off. There is probably a way to write a function which does this if you would like, or you can use my "hack". If I want to stop filtering in one of the regions, I simply input a very low frequency (like 10 Hz) which is going to have little to no effect upon the ions. In addition, you can change the delay to be very short so that the effect of the auxiliary frequency is as minimal as possible. This seems to be an effective strategy when you want to stop filtering while running, or if you did not want to use one or two of the filtering regions.

Table 5.6 – Serial commands for interval 3 of program E, which has three regions of selective mass elimination using square parametric excitation. Note that this is the only program set up to be externally triggered on Signal 53.

Serial Command	Variable Name	Unit	Function
r1:	filterFreq	Hertz	Base RF frequency during all 3 isolation regions
r2:	auxFreq1	Hertz	Parametric frequency for elimination region #1
r3:	auxFreq2	Hertz	Parametric frequency for elimination region #2
r4:	auxFreq3	Hertz	Parametric frequency for elimination region #3
W2:	filterDuration1	milliseconds	Duration of elimination region #1
W3:	filterDuration2	milliseconds	Duration of elimination region #2
W4:	filterDuration3	milliseconds	Duration of elimination region #3
W7:	filterDuration7	milliseconds	Delay for laser interaction after switch to stable conditions

Section 5.5: LabVIEW Data Analysis Program

The LabVIEW data analysis program for the DC-LIT-MS was written by Casey Howdieshell[3] and modified slightly by me. Raw ion signal is sent from the CEM detector to the 100x bipolar amplifier, then to the Picoscope Channel A. The LabVIEW program works by taking in data from the Pico, which is triggered by the LVWFG when the mass scan starts in the LIT. There are two trigger signals from the LVWFG to the Pico: “SCAN” which goes to the external trigger (labeled EXT) and “TRIG” which goes to Channel B. The “SCAN” trigger goes high when the mass scan starts and low when it ends, signaling to the program when to start and stop taking data. The “TRIG” trigger goes high at the beginning of a frequency step and low at the end of each

step while the DDS module switches to the next frequency, creating frequency "bins". These frequency bins show up as the red trace on the upper raw signal plot in the user interface, shown in Figure 5.8.

The values of the frequency bins are calculated in the Arduino code from the rearranged Mathieu q equation, as explained above in Section 5.4. In the upper left-hand corner of the LabVIEW interface, you will also input the start m/z and delta m/z to match what you input in the Arduino code. This will allow the program to take the upper plot of raw signal in time and transform it into the lower intensity vs. m/z plot. The "QTPI Stick Figure MS" sub vi (file name: *Stick Figure Mass Spec_ion_count_binning_GOC*) counts all the lines present in each bin above a certain threshold of noise and sums them to make the lower plot.

The binning noise threshold is an adjustable variable in the upper left side panel. I typically use a threshold of 0.015, and all the data I have recorded with this LabVIEW code is with that value. However, if the baseline noise changes in the upper plot, you can adjust the threshold while running. Note that increasing this threshold will decrease your baseline noise in the intensity vs. m/z plot, but it will also decrease your ion signal since less ions will be counted. Therefore, it is important to carefully select this threshold and keep it consistent across comparable data sets. The counting process described here is the only operational difference between my LabVIEW code (file name: *QTPI_MS_7_25_22_ionCountBins_GOC*) and previous versions, which instead summed the intensities of the raw ion signal in each bin. I changed the data processing to be more consistent with typical methods of ion trap MS analysis, and to have more user control over the noise threshold.

There are two modes for data analysis in the LabVIEW code: streaming and set number of averages. In streaming mode, the Pico will continuously take a rolling averages of data (adjustable

in the blue box, 8 minimum) which is displayed in the "Raw Output" plot (see Figure 5.5). You will continuously see the ions hit the detector in the upper plot, with a slight delay between the upper and lower plot since the binning process takes time. This is the mode you use while optimizing ion signal and adjusting instrument conditions.

Once you are ready to record a mass spectrum, you will stop the program using the "STOP" button in the user interface and switch to the set number of averages mode. *While the program is off*, click the Binned Data Mode "Streaming" button in the blue box titled "MUST BE CHANGED BEFORE RUNNING". The button will then read "Set Number of Averages". You must also change the "Take Data # Averages" value below the button before running (50-100 averages is standard). Once you click "Run" ► in the LabVIEW header, the program will run for that many averages and stop automatically when finished. Then you can export the binned intensity vs. m/z plot for further analysis. The easiest way to do so is to right click on the plot and select "Export" then "Export Data to Clipboard" and copy into an Origin file. You can also select "Export" then "Export Data to Excel". If you ever desire to export the upper raw signal plot, you *must* utilize "Export Data to Excel" because there are many more data points and the program tends to crash if you try to copy to clipboard.

Figure 5.8 shows the current LabVIEW user interface. In the upper lefthand corner is the start m/z, delta m/z, and "Voltage Threshold for Binning" as described above, as well an indicator that the program is triggered. You can see the upper "Raw Output" plot coming directly from the Pico, where the red trace is the frequency bins and the white trace is raw ion signal. When you turn on the high voltage square waves, this white baseline will increase dramatically due to high frequency noise. You should select your voltage threshold depending on where that baseline lies. Now, let us continue going down the lefthand side of the interface in Figure 5.8.

Below there is a “Cursors” button which turns adjustable cursors on in the “Binned Ion Signal” plot. These cursors can be useful for finding the m/z center of peaks and keeping track of mass filtered peaks on the fly. Next are settings for the Pico data collection: the sampling interval and number of samples that the Pico records. You will have to adjust the “Post-trigger Samples” whenever you increase the mass range, since more data points need to be recorded. For my mass range of 100-300 m/z and typical mass scan of 100 periods/frequency step, I have used settings of 104 ns sample interval and 1.4 million samples. Note that after typing in new Pico settings, you *must* press the “Apply New Pico Settings” button to actually change these values.

Below the Pico settings is a button to turn on/off a digital lowpass filter which comes from a LabVIEW library, along with settings. This lowpass filter is no longer in use, so the button should read “Lowpass Filter OFF.” When you restart the program, it will default on and you will see a very flat baseline, so make sure to press the button to turn it off. This feature was used to filter out noise, but is no longer necessary since acquiring a new, much less noisy detector. I would recommend not using it since it artificially attenuates both the noise and signal. Instead, you can now adjust the noise threshold for binning when there are noticeable changes in the baseline.

Finally, at the bottom of the screen is the blue box with parameters you must change before running, the m/z locations of cursors, and timings for the sample interval and total loop time. The parameters you must change before running are the mode of data collection described above (streaming or set # averages), the input number of averages for each mode, and other variables related to the Pico channels. These Pico variables should not need to be adjusted, since they are typical oscilloscope settings such as voltage range and trigger type.

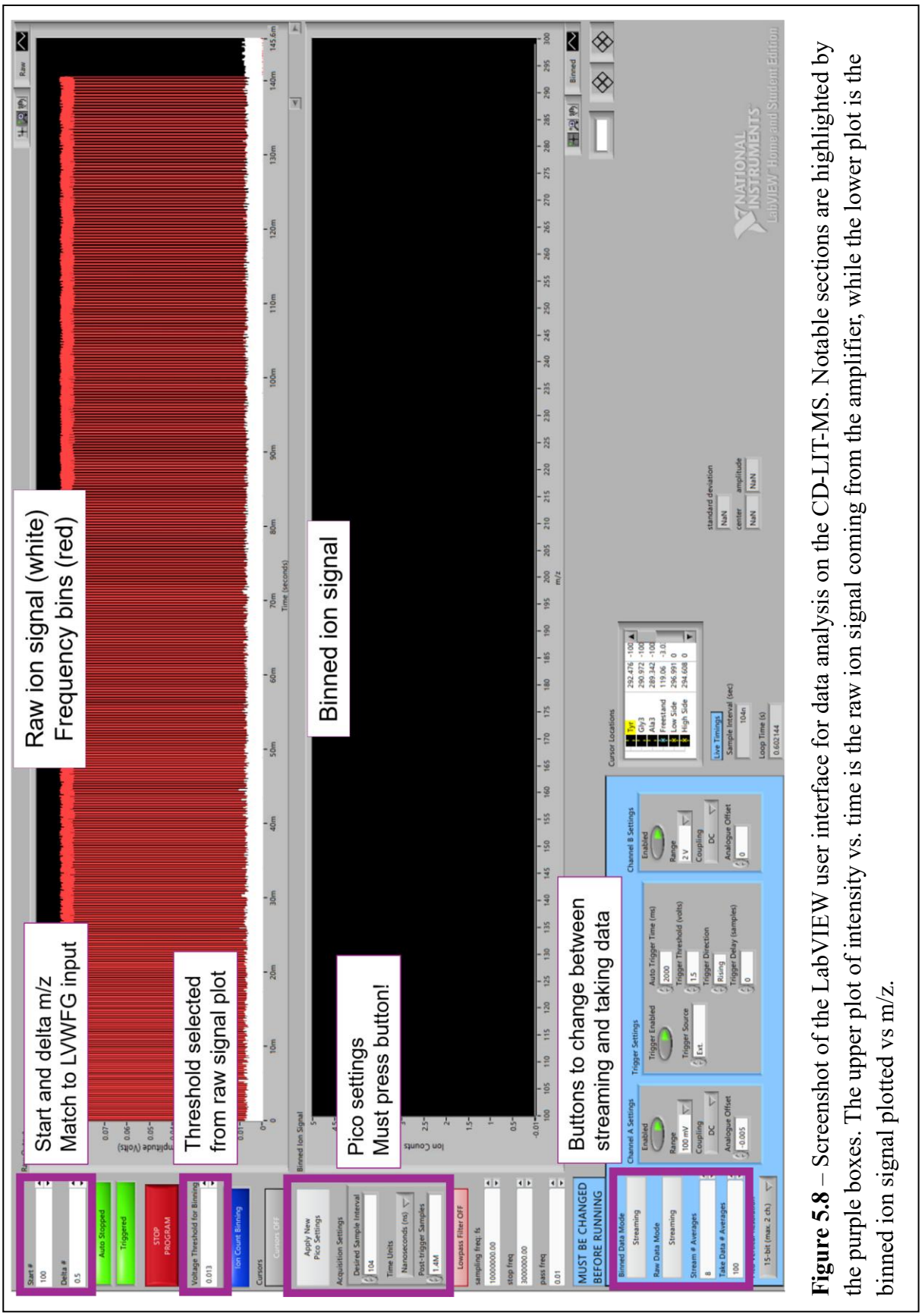


Figure 5.8 – Screenshot of the LabVIEW user interface for data analysis on the CD-LIT-MS. Notable sections are highlighted by the purple boxes. The upper plot of intensity vs. time is the raw ion signal coming from the amplifier, while the lower plot is the binned ion signal plotted vs m/z.

Section 5.6: Overview of Instrument Timings

In this final section, I will provide an overview of current instrument timings and triggers. Figure 5.9 shows the two Stanford boxes for this instrument, Stanford box #1 which runs at 10 Hz and Stanford box #2 which runs at the much slower LIT cycle rate. The 10 Hz box triggers the pulsed valve, laser, and Stanford box #2. The laser is always pulsing at 10 Hz, and a shutter is used to block the beam so that only every fifth pulse goes into the trap. You will see in Figure 5.9 that the lower 10 Hz box on the shelf uses 3 connections: T0 split so that it goes to the RT pulsed valve and Stanford box #2, AB going to the laser flash lamp, and CD going to the Q switch.

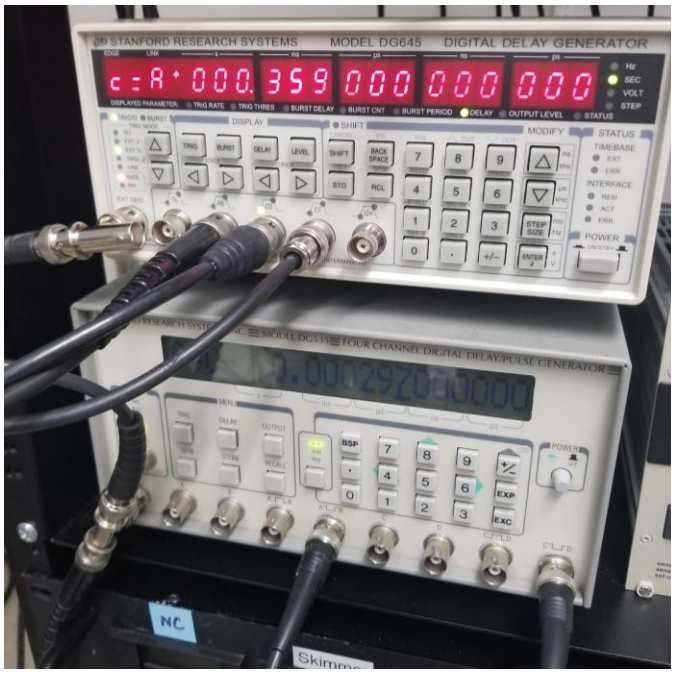


Figure 5.9 – Picture of the two Stanford boxes for this instrument. The lower box is internally triggered at 10 Hz and externally triggers the upper box. The upper box runs much slower according to the instrument cycle of the LIT, so the long delays will cause it to skip triggers from the 10 Hz box.

A line diagram in Figure 5.10 shows all of the signals from the slow Stanford box in time, starting with the pulsed valve trigger in the RT. I find it helpful to think of the RT pulsed valve as the starting point, since it is the first thing the ions "experience" in the instrument. Ions are created via ESI and guided through the first 3 stages, then enter the RT and experience the gas pulse. After the ions are trapped in the RT for a 10 ms delay period, they experience a 1 ms DC pulse out of the RT. Then there is a long 359 ms delay for the LIT trapping cycle. During this delay, ions will be transferred to the LIT and trapped under stable square wave conditions (usually set to 600 kHz, 50:50 duty cycles).

After LIT trapping is the period for square parametric excitation. The current timing scheme shown in Figure 5.10 is for Arduino program E which contains three regions for selective mass elimination (see Table 5.6). Therefore, the total filtering time is 30 ms, 10 ms for each of the three m/z species. Program E is also set to be triggered at the beginning of the selective mass elimination interval (interval 3), so the CD output of Stanford box #2 goes to Signal 53 on the LVWFG. Keep in mind that if you change the Arduino program, you will need to adjust these timings on the Stanford box. Also remember that it is crucial that the Stanford box timings match your input into the Arduino code – for example, the trapping time (W0) must match the 359 ms on the Stanford box or else the laser shot will not fire at the correct time.

At the same time that the LVWFG is triggered before selective mass elimination, there is a trigger (EF) to open the shutter in preparation for the laser. The laser pulse is very fast (ns), but the shutter takes some time to open, so we make sure it is opened in time by triggering it at the start of the parametric excitation interval and leaving it open for 60 ms. The shutter does have a ~5 ms delay time for opening, and ~8 ms delay for closing. However, this is not a problem since you have a generous 100 ms window during the 10 Hz cycle to open and close it for a single laser shot. The

EF trigger on Stanford box #2 goes directly to externally trigger the solenoid controller for the shutter, which sits on the back lower shelf of the instrument right outside the optics box.

After the CD trigger for the LVWFG and EF trigger for the shutter is one more delay for the mass scan in the LIT. The current mass scan is set to 100 – 300 m/z with 0.5 m/z steps, corresponding to ~150 ms. This delay can be less precise, because it simply has to get you past the next 10 Hz trigger. The instrument cycle will then begin again starting with the sixth trigger from the 10 Hz box, which would come after the time range shown in Figure 5.10. The Arduino code is programmed to go back to stable square wave conditions after the mass scan, that way everything can just be waiting for the next trigger to start it all over again. If you change the mass scan length by a considerable amount, you may need to look at the scope output to determine its approximate length and adjust these triggers accordingly.

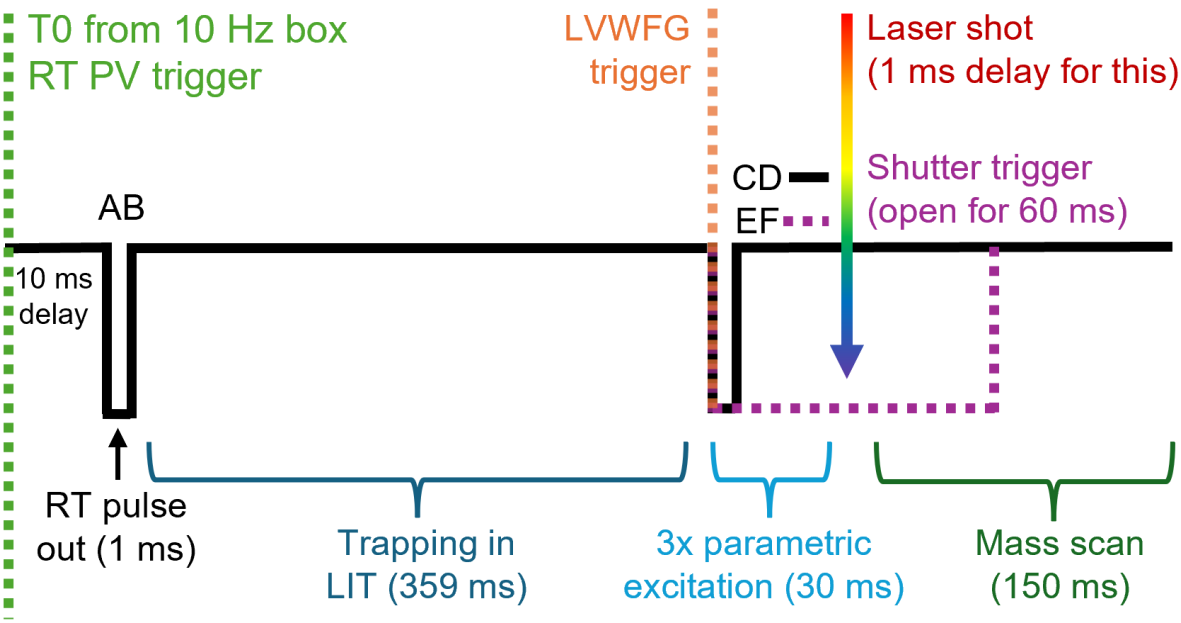


Figure 5.10 – Current timing schematic for the CD-LIT-MS with laser interface (not really to scale). This shows the outputs of Stanford box #2 which runs at the slower LIT cycle rate and is externally triggered by another 10 Hz Stanford box. This timing is for Arduino program E, which has a region for selective mass elimination of 3 m/z species.

References

(1) Roesch, G. C. Innovation in Mass Spectrometry Instrumentation for Applications in Cryogenic Ion Vibrational Spectroscopy. Ph. D. Dissertation, University of Wisconsin-Madison, 2023.

(2) Hoffman, N. M.; Gotlib, Z. P.; Opačić, B.; Clowers, B. H.; Reilly, P. T. A. A Comparison Based Digital Waveform Generator for High Resolution Duty Cycle. *Rev. Sci. Instrum.* **2018**, *89* (8), 084101. https://doi.org/10.1063/1.5004798.

(3) Howdieshell, C. J. Linear Digital Ion Trap Mass Spectrometry and its Application for Cryogenic Ion Spectroscopy. Ph.D. Dissertation, University of Wisconsin-Madison, 2021.

(4) Hoffman, N. M.; Opačić, B.; Reilly, P. T. A. Note: An Inexpensive Square Waveform Ion Funnel Driver. *Rev. Sci. Instrum.* **2017**, *88* (1), 016104. https://doi.org/10.1063/1.4974345.

Chapter 6: Towards a Multiplexed CIVS Workflow

In this final chapter, the experimental challenges and results of microsolvation experiments on the CD-LIT-MS are described. Section 6.1 introduces microsolvation studies in cryogenic ion traps, including an IRMPD spectra of water clusters from our Re-TOF CIVS instrument. In section 6.2, the formation and selective mass elimination by square parametric excitation of polypeptide water clusters is demonstrated, satisfying the mass filtration step which is required for the multiplexed spectroscopy workflow. We also demonstrate that parametric excitation results in enhancement of specific clusters via gentle dissociation. Finally, the progress towards performing IR spectroscopy in this instrument is discussed in section 6.3, along with potential future experiments and applications.

Section 6.1: Microsolvation Studies in Cryogenic Ion Traps

The solvation of molecules is an important fundamental process to understand, especially in biological systems where the interactions between biomolecules and water can result in notable structural changes which affect their function.[1] However, since solvation is such a dynamic process, it is challenging to investigate in the solution phase. Cryogenic ion spectroscopy of gas-phase solvation clusters has proven to be a viable method to elucidate the effects of solvation upon molecular structure in a more controlled environment.[2] This approach has been used successfully to investigate biomolecules, for example, the antibiotic gramicidin S with 1-50 water molecules by Rizzo and colleagues.[3]

The Garand group has produced significant results in this area, documenting the conformational changes and intermolecular interactions of water with amino acids such as

protonated glycine and alanine[4-5] and protonated tripeptides containing those residues.[6-7] We have also demonstrated a more universal approach to solvent cluster formation in cryogenic ion traps, where the buffer gas can be seeded with solvent to cluster with the ion(s) of interest in the gas phase.[3] IRMPD spectra on water clusters from the Re-TOF CIVS instrument are shown in Figure 6.1. These spectra correspond to water clusters 1-9 of protonated betaine, also called glycine betaine or trimethylglycine. Figure 6.1 serves to highlight the distinct spectral features which allow us to distinguish between different-numbered water clusters and their binding effects.

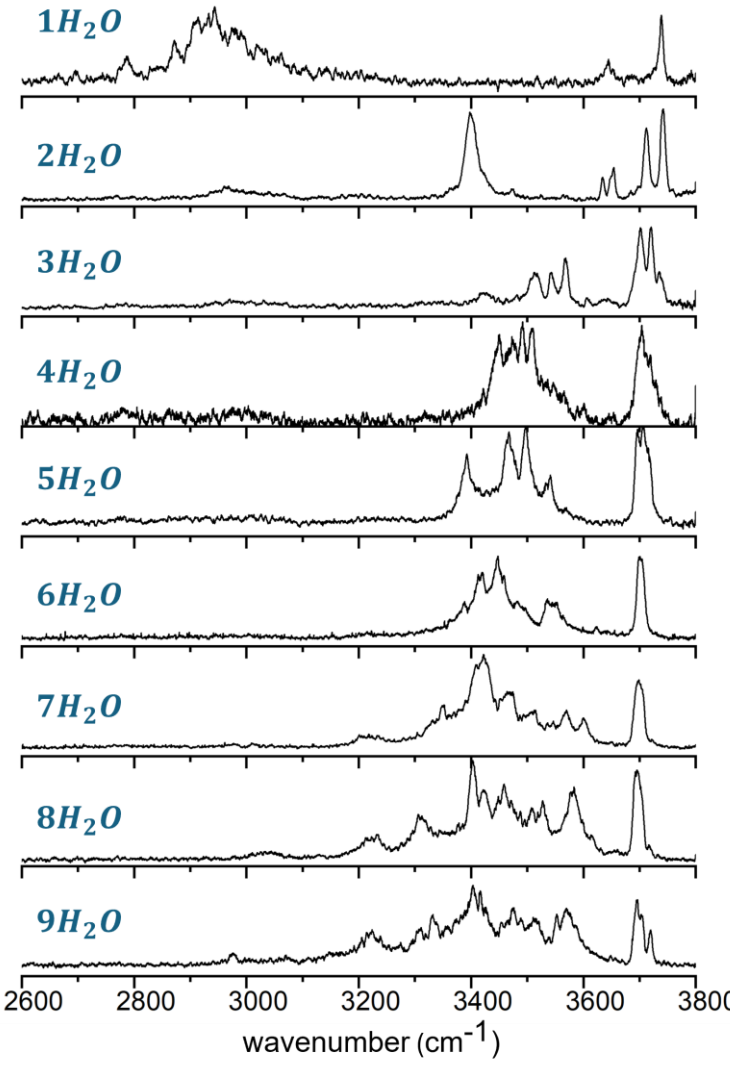


Figure 6.1 – IRMPD spectra of protonated betaine water clusters 1-9 from our traditional CIVS instrument.

Section 6.2: Mass Selection of Water Clusters in the CD-LIT-MS

In order to achieve the proof-of-concept water clustering experiment in the CD-LIT-MS, I added the LN_2-cooled octupole ion trap from the original CIVS instrument to the instrument. This trap has demonstrated very effective clustering capabilities, including up to 30 water clusters produced at one time in the original CIVS instrument.[8] The first step was therefore to reproduce these clustering results in the CD-LIT-MS.

However, water clustering in this instrument was not a trivial pursuit. There are two key differences between the LIT and TOF mass analyzers which present challenges for cluster formation and survival. First, the LIT runs at higher pressure (up to 1×10^{-4} Torr) than a TOF tube ($\sim 1 \times 10^{-7}$ Torr), introducing the concern that clusters could experience too many gas collisions and dissociate if conditions are too harsh. Second, the timing of the Re-TOF and LIT instruments is dramatically different. Re-TOF CIVS runs on a 10 Hz cycle to match the laser, but the LIT requires a trapping time of $\geq$ 300 ms and a mass scan on the order of 100 ms, making the overall instrument cycle only 1-2 Hz. To make water clusters, we typically seed the Helium buffer gas with water and pulse it into the RT at the instrument cycle rate. Since the high initial pressure and subsequent decay of the gas pulse are crucial for cluster formation, this change in timing was another uncertain variable.

After much optimization of the RT trapping conditions and pressures, I was only able to form 1-2 water clusters when pulsing gas at the 1-2 Hz LIT rate, and the results were not reproducible day-to-day. By switching the RT to a 10 Hz cycle separate from the LIT cycle, I was able to see protonated betaine water clusters 1-6 reliably, and a small number of clusters 8-10 depending on RT pressure as shown in Figure 6.2. RT gas pulse rates up to 30 Hz resulted in cluster formation, while pulse rates below 10 Hz resulted in little to no clusters. This discovery confirmed

that timing is very important in the dynamic cluster forming process. We believe that during the slow 1-2 Hz gas pulse, the pressure inside the RT likely did not reach the same peak value and duration which is required to cause enough collisions for optimal cluster formation.

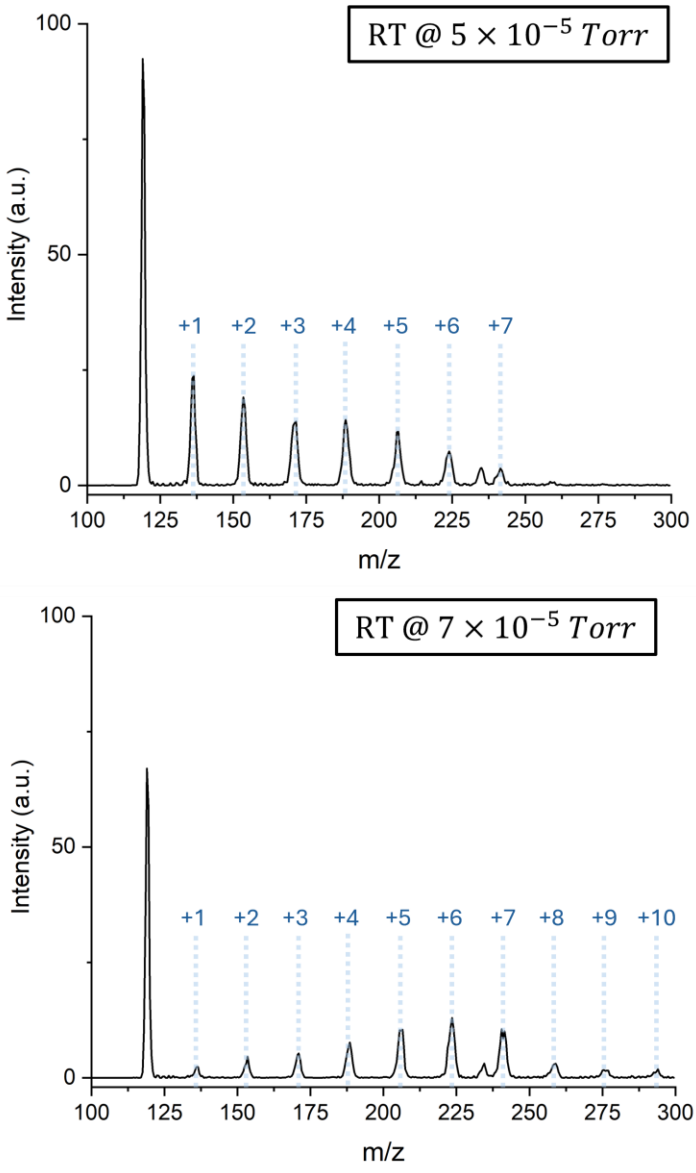


Figure 6.2 – Initial results of water clusters on betaine (m/z 118) at two different RT pressures after implementing the 10 Hz gas pulse rate. At the higher RT pressure on the bottom, small amounts of clusters 8-10 are produced. For this experiment, the RT was at 78 K and LIT at 52 K.

Although clusters will also form at RT gas pulse rates > 10 Hz, I discovered that there was not much advantage in doing so. Faster pulses resulted in much higher overall pressure in the RT, causing a deleterious effect upon the clusters at pulse rates > 30 Hz. Additionally, allowing multiple fast pulses of clusters in the LIT resulted in lower mass resolution, likely due to less effective trapping from a broadening of the kinetic energy distribution of the ionic clusters. This also made selective mass elimination by square parametric excitation in the LIT less effective, since the peaks were broader and encompassed a wider resonant frequency range. Therefore, the RT gas pulse rate of 10 Hz was selected for optimal cluster formation and mass spectrometry performance.

After forming abundant signal of water clusters on betaine at the RT 10 Hz pulse rate, the pressure in the LIT did not appear to have much of an effect. At the primary operational pressure range of the LIT ($5 \times 10^{-5} - 8 \times 10^{-5}\ Torr$), there was no visible difference in the number of clusters or their intensities. This confirmed that to the best of our knowledge, clusters are not being destroyed by gas collisions in the higher pressure environment of the LIT. The mass spectra in Figure 6.2 were taken at a lower LIT pressure of 5×10^{-5} Torr, while the spectra in Figure 6.3 were optimized and taken at a more typical LIT pressure of 7×10^{-5} Torr, demonstrating no significant adverse effects upon the clusters.

The next step of the proposed multiplexed spectroscopy workflow was to perform selective mass elimination of every other water cluster so that their photofragments could be clearly observed in that m/z space. The results of this are shown in Figure 6.3, where a set of betaine water clusters 1-8 were formed and the even clusters 2, 4, and 6 were selectively eliminated by square parametric excitation. Parametric frequencies of 108, 142, and 190 kHz were applied for 10 ms each to eliminate clusters 6, 4, and 2, respectively. The base RF waveform was fixed to 550 kHz, 52:45 duty cycles for these experiments and the auxiliary excitation voltage was set to 4 V.

It appears that resonantly exciting one m/z cluster results in "shaking off" a water molecule, which can be seen from the increase in peak heights and areas for clusters 1, 3, and 5 in Figure 6.3. Analysis of the peak areas indicates that only one water molecule is typically lost during this resonant excitation, e.g., most of the excited $6H_2O$ cluster funnels into the $5H_2O$ peak, $4H_2O$ to $3H_2O$, and so on. This technique may have future applications in solvation studies, since it allows for very selective enhancement of one or more clusters. Roesch & Garand saw a similar effect with square wave duty cycle and frequency filtering, where water molecules were "shaken off" after excitation.[9] This similarity is not unexpected, since both these processes induce gentle resonant motion of the complex.

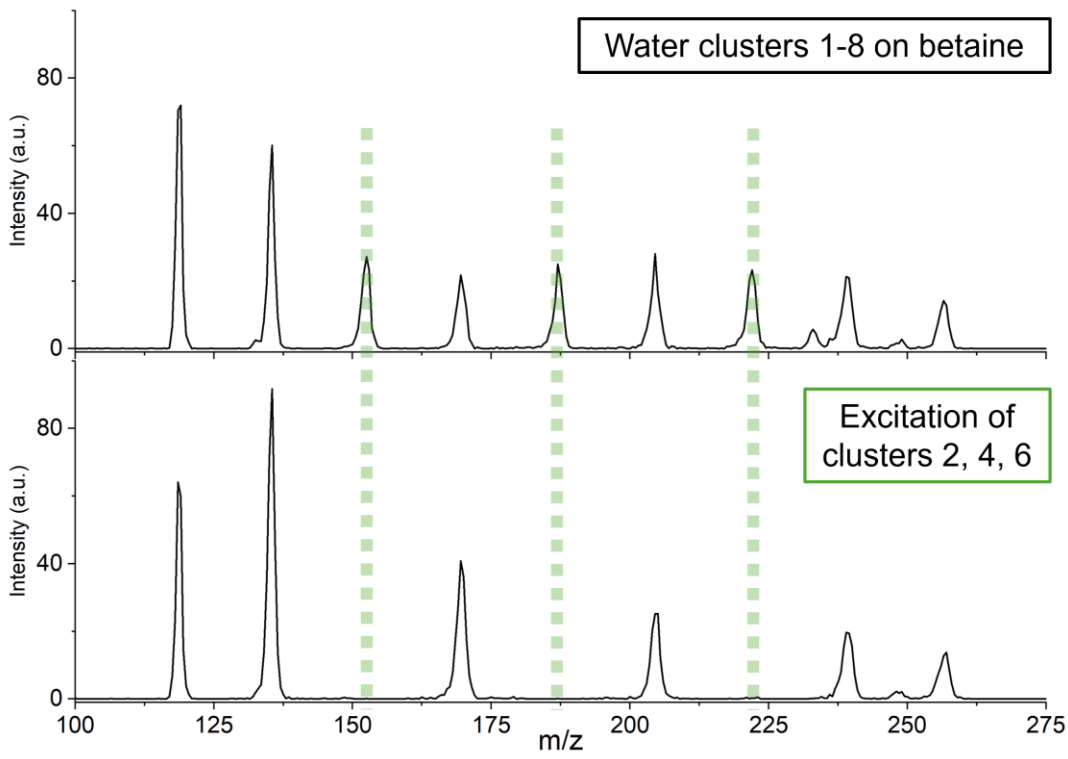


Figure 6.3 – Selective mass elimination of protonated betaine water clusters 2, 4, and 6 using parametric excitation frequencies of 190, 142, and 108 kHz, respectively. The RF frequency was fixed to 550 kHz with X:Y duty cycles of 52:45. The excitation signal was applied for 10 ms per cluster, corresponding to 30 ms total filtering time.

Section 6.3: Future IR Spectroscopy and Applications

Now that the formation and selective mass elimination of water clusters has been achieved in the CD-LIT-MS, the final step is to perform IRMPD spectroscopy. Optics were added to send the second laser from our primary CIVS instrument to a small optics box on the back shelf which contains a shutter, turning/focusing mirror, and a KBr window into the back of the LIT. The shutter is used to block 4/5 laser pulses during the instrument cycle, since the LIT must currently run at a slower rate than 10 Hz (see Chapter 5 for more detailed timings). The laser was successfully aligned and photodissociation was observed, but unfortunately, a complete IR spectrum could not be taken during my PhD timeline. Since all the required physical components are successfully installed, it is my sincere hope that a new Garand group student will take on this project and optimize the CIVS experiment in this new instrument, starting with multiplexed spectroscopy of water clusters as proof-of-concept.

In the future, the CD-LIT-MS instrument would be ideal for investigating even larger solvent clusters due to the high upper m/z limit of the digital trap and multiplexed workflow. The current 100 V_{pp} square waves on the LIT can only sufficiently trap and analyze species up to ~600 m/z with sufficient resolution, therefore a higher voltage power supply will be necessary for these studies. It is likely that greater V_{pp} square waves would also allow the instrument cycle to be made faster, since higher amplitude square waves deepen the potential well and should therefore shorten the required collisional cooling time. Therefore, further advancements in the square wave drivers will enhance overall instrument performance and efficiency, as well as allow higher mass species to be studied.

Beyond fundamental solvation studies, the CD-LIT-MS shows promise for investigating more complex mixtures where the digital mass filtration and square parametric excitation

capabilities would be advantageous. Applications in metabolomics or lipidomics, for example, could benefit highly from the IR spectroscopy information gained from the CIVS technique. Little sample preparation of such biological samples would be required since gas-phase manipulation in the LIT and IRPD/IRMPD spectroscopy results can aid in deconvoluting complex spectra and distinguish between isomeric species. In this way, the CD-LIT-MS can be a step towards making the CIVS technique more widely accessible and compatible with MS-based workflows.

References

(1) Makarov, V.; Pettitt, B. M.; Feig, M. Solvation and Hydration of Proteins and Nucleic Acids: A Theoretical View of Simulation and Experiment. *Acc. Chem. Res.* **2002**, *35* (6), 376–384. https://doi.org/10.1021/ar0100273.

(2) Garand, E. Spectroscopy of Reactive Complexes and Solvated Clusters: A Bottom-Up Approach Using Cryogenic Ion Traps. *J. Phys. Chem. A* **2018**, *122* (32), 6479–6490. https://doi.org/10.1021/acs.jpca.8b05712.

(3) Nagornova, N. S.; Rizzo, T. R.; Boyarkin, O. V. Interplay of Intra- and Intermolecular H-Bonding in a Progressively Solvated Macrocyclic Peptide. *Science* **2012**, *336* (6079), 320–323. https://doi.org/10.1126/science.1218709.

(4) Voss, J. M.; Fischer, K. C.; Garand, E. Accessing the Vibrational Signatures of Amino Acid Ions Embedded in Water Clusters. *J. Phys. Chem. Lett.* **2018**, *9* (9), 2246–2250. https://doi.org/10.1021/acs.jpclett.8b00738.

(5) Fischer, K. C.; Sherman, S. L.; Garand, E. Competition between Solvation and Intramolecular Hydrogen-Bonding in Microsolvated Protonated Glycine and β-Alanine. *J. Phys. Chem. A* **2020**, *124* (8), 1593–1602. https://doi.org/10.1021/acs.jpca.9b11977.

(6) Fischer, K. C.; Voss, J. M.; Zhou, J.; Garand, E. Probing Solvation-Induced Structural Changes in Conformationally Flexible Peptides: IR Spectroscopy of Gly3H+·(H2O). *J. Phys. Chem. A* **2018**, *122* (41), 8213–8221. https://doi.org/10.1021/acs.jpca.8b07546.

(7) Sherman, S. L.; Fischer, K. C.; Garand, E. Effects of Methyl Side Chains on the Microsolvation Structure of Protonated Tripeptides. *J. Phys. Chem. A* **2023**, *127* (30), 6275–6281. https://doi.org/10.1021/acs.jpca.3c03711.

(8) Marsh, B. M.; Voss, J. M.; Garand, E. A Dual Cryogenic Ion Trap Spectrometer for the Formation and Characterization of Solvated Ionic Clusters. *J. Chem. Phys.* **2015**, *143* (20), 204201. https://doi.org/10.1063/1.4936360.

(9) Roesch, G. C.; Garand, E. Tandem Mass-Selective Cryogenic Digital Ion Traps for Enhanced Cluster Formation. *J. Phys. Chem. A* **2023**, *127* (36), 7665–7672. https://doi.org/10.1021/acs.jpca.3c04706.

Appendix

A1: Screenshot of Python program for plotting digital quad Mathieu stability diagrams, described in Chapter 2.
File name: *DigitalQuad_MathieuStabilityDiagrams.py*

```
# -*- coding: utf-8 -*-
"""
November 2023
Author: Grace Capek, Garand Group, University of Wisconsin - Madison
Questions and suggestions welcome, please contact at gcapek@wisc.edu

This is a simple program to plot grid-based Mathieu a-q stability diagrams
from the matrix solutions of the Mathieu-Hill equations for a linear quadrupole.
The user should adjust the duty cycle and a-q range for their experiment.
Important references: Konenkov et al 2002, Brabeck et al 2014

"""

#%% Import the necessary libraries
import numpy as np
import matplotlib.pyplot as plt
import functools as ft

#%% User-adjustable variables: X and Y duty cycles in the form of t1 and t3, a and q ranges for stability plot
### Adjust the # of points in the a-q arrays for a less pixelated image
### Note that it runs a little slow when you go higher than a 100x matrix - she's working, I promise

t1 = 0.600                      # HIGH duty cycle for the X rods as a fraction of period T
t3 = 0.400                      # HIGH duty cycle for the Y rods as a fraction of period T

# Note that when a = 0 you divide by zero in the math, so I start at very small a to avoid this
# (if I mathed this wrong, somebody please let me know, thanks)
# Also note that if you change the a-q range, you may need to adjust the plot scaling in line 135
a = np.linspace(0.001,0.5,500)    # array of Mathieu a values
q = np.linspace(0.001,1,500)      # array of Mathieu q values

#%% Small calculations from user-input variables
qq,aa = np.meshgrid(q,np.flip(a))   # grid of a and q values to use for the calculation later
t2 = np.absolute(1-t1-t3)           # calculate t2, the time that the quadrupolar potential is zero
tau1 = t1*np.pi                     # calculate tau for all 3 time points during the square wave period
tau2 = t2*np.pi
tau3 = t3*np.pi

#%% Define the functions which calculate the transfer matrices
### Inputs are tau and f values, output is the 4by4 matrix M

# create a function to calculate the matrix when f > 0
def transM_pos(f,t):
    m11 = np.cos(t*np.sqrt(f))
    m12 = (1/np.sqrt(f))*np.sin(t*np.sqrt(f))
    m21 = -np.sqrt(f)*np.sin(t*np.sqrt(f))
    m22 = np.cos(t*np.sqrt(f))
    M = [[m11,m12],[m21,m22]]
    return M
```

```
# create a function to calculate the matrix when f < 0
def transM_neg(f,t):
    m11 = np.cosh(t*np.sqrt(-f))
    m12 = (1/np.sqrt(-f))*np.sinh(t*np.sqrt(-f))
    m21 = np.sqrt(-f)*np.sinh(t*np.sqrt(-f))
    m22 = np.cosh(t*np.sqrt(-f))
    M = [[m11,m12],[m21,m22]]
    return M

#%% Define the functions which calculate the transmission matrices and their traces for X and Y
### Inputs are the arrays of Mathieu a and q values, output is the trace of the matrix M

# Function to calculate X dimension traces
def X_trace(q,a):
    # Define f values for X dimension
    f1x = a + 2*q
    f2x = a
    f3x = a - 2*q
    # Calculate the transmission matrices from functions defined above
    if f1x > 0:
        M1x = transM_pos(f1x,tau1)
    else:
        M1x = transM_neg(f1x,tau1)
    if f2x > 0:
        M2x = transM_pos(f2x,tau2)
    else:
        M2x = transM_neg(f2x,tau2)
    if f3x > 0:
        M3x = transM_pos(f3x,tau3)
    else:
        M3x = transM_neg(f3x,tau3)
    # Multiply all 3 matrices by each other to get the total transmission matrix
    MX_total = ft.reduce(np.dot,[M1x,M2x,M3x])
    MX_trace = np.absolute(np.trace(MX_total))  # Take the trace of that matrix
    return MX_trace
```

```
# Function to calculate Y dimension traces
def Y_trace(q,a):
    # Define f values for Y dimension
    f1y = -a - 2*q
    f2y = -a
    f3y = -a + 2*q
    # Calculate the transmission matrices from functions defined above
    if f1y > 0:
        M1y = transM_pos(f1y,tau1)
    else:
        M1y = transM_neg(f1y,tau1)
    if f2y > 0:
        M2y = transM_pos(f2y,tau2)
    else:
        M2y = transM_neg(f2y,tau2)
    if f3y > 0:
        M3y = transM_pos(f3y,tau3)
    else:
        M3y = transM_neg(f3y,tau3)
    # Multiply all 3 matrices by each other to get the total transmission matrix
    MY_total = ft.reduce(np.dot,[M1y,M2y,M3y])
    MY_trace = np.absolute(np.trace(MY_total))  # Take the trace of that matrix
    return MY_trace
```

```
#%% Iterate through the a and q matrices and calculate the trace of the transmission matrix
### for each a,q combination in both dimensions

# Initialize matrices for the X and Y traces
X_traces = np.zeros(np.shape(qq))
Y_traces = np.zeros(np.shape(qq))
# For the specified a-q range, calculate the X and Y traces and put them in the matrices
for i in range(len(q)):
    for j in range(len(q)):
        X_traces[i,j] = X_trace(qq[i,j],aa[i,j])
        Y_traces[i,j] = Y_trace(qq[i,j],aa[i,j])
# Initialize a grid to plot later
grid = np.zeros(np.shape(qq))
# Assign integer values for each grid item based on X and Y stability
# Ions only have a stable trajectory in that dimension if the trace < 2
for i in range(len(q)):
    for j in range(len(q)):
        if X_traces[i,j] < 2 and Y_traces[i,j] > 2:     # X stable, Y unstable assigned to 1
            grid[i,j] = 1
        if X_traces[i,j] < 2 and Y_traces[i,j] < 2:     # X stable, Y stable assigned to 2
            grid[i,j] = 2
        if X_traces[i,j] > 2 and Y_traces[i,j] < 2:     # X unstable, Y stable assigned to 3
            grid[i,j] = 3

#%% Plot the grid of integer values based on the stability conditions defined above
plt.imshow(grid, cmap = 'Greens', extent=[0.001,1,0.001,0.5])
plt.xlabel("q", fontsize = 14)
plt.ylabel("a", fontsize = 14)
# Title contains the duty cycles as high X:Y in %
plt.title(str(t1*100) + ":" + str(t3*100) + " Mathieu Stability Diagram", fontsize = 14)
plt.xticks(fontsize = 12)
plt.yticks(fontsize = 12)
# Uncomment the next line if you want to export a higher res pic
plt.savefig("60_40_a-q.png",dpi=1200)
```

A2: Screenshot of Python program to calculate the first order parametric resonances in a linear digital quadrupole, described in Chapter 3.
File name: *Matrix_calc_quad_stability.py*

```
# -*- coding: utf-8 -*-
"""
Created on Tue Oct 31 15:02:34 2023

@author: grace
"""
#%% This program calculates the transmission matrix for a linear quadrupole using
### matrix methods (see Konekov et al 2002) in order to get the stability parameter
### Beta, quadrupolar resonant frequencies, and the Mathieu a-q stability diagram
import numpy as np
import matplotlib.pyplot as plt
import functools as ft
```

```
#%% Define the functions to calculate f values and the first order (K1) quadrupolar frequencies

# create a function to calculate the matrix when f > 0
def transM_pos(f,t):
    m11 = np.cos(t*np.sqrt(f))
    m12 = (1/np.sqrt(f))*np.sin(t*np.sqrt(f))
    m21 = -np.sqrt(f)*np.sin(t*np.sqrt(f))
    m22 = np.cos(t*np.sqrt(f))
    M = [[m11,m12],[m21,m22]]
    return M
# create a function to calculate the matrix when f < 0
def transM_neg(f,t):
    m11 = np.cosh(t*np.sqrt(-f))
    m12 = (1/np.sqrt(-f))*np.sinh(t*np.sqrt(-f))
    m21 = np.sqrt(-f)*np.sinh(t*np.sqrt(-f))
    m22 = np.cosh(t*np.sqrt(-f))
    M = [[m11,m12],[m21,m22]]
    return M
```

```
def K1freq_calc(mass_array,baseF_kHz,t1,t3):
    # Define the constants and calculated variables
    t2 = np.absolute(1-t1-t3)               # calculate t2, the time that the quadrupolar potential is zero
    e_C = 1.602176634e-19                   # charge of one electron in Coulombs
    z = 1                                   # in our case, usually only one charge
    U = (49+-55)-(46+-53)                   # DC potential in Volts
    #U = 1
    V = 50                                  # zero-to-peak voltage of square wave in Volts
    r0 = 0.005                              # trap radius in meters
    baseF_rad = baseF_kHz*1000*2*np.pi      # convert base frequency to radians
    Beta_array = []

    for m_Da in mass_array:
        # Convert mass in Da to kg
        m_kg = m_Da*1.66054e-27
        # Calculate Mathieu a and q stability parameters
        a = (8*e_C*z*U)/(m_kg*(baseF_rad**2)*(r0**2))
        q = (-4*e_C*z*V)/(m_kg*(baseF_rad**2)*(r0**2))
        # Calculate the input to the matrices, tau and f
        tau1 = t1*np.pi
        tau2 = t2*np.pi
        tau3 = t3*np.pi
        f1 = a + 2*q
        f2 = a
        f3 = a - 2*q
        # f1 = -a - 2*q
        # f2 = -a
        # f3 = -a + 2*q
        # Calculate the three transmission matrices for t1, t2, and t3
        if f1 > 0:
            M1 = transM_pos(f1,tau1)
        else:
            M1 = transM_neg(f1,tau1)
        if f2 > 0:
            M2 = transM_pos(f2,tau2)
        else:
            M2 = transM_neg(f2,tau2)
        if f3 > 0:
            M3 = transM_pos(f3,tau3)
        else:
            M3 = transM_neg(f3,tau3)
        # Multiply the transmission matrices by each other to get the total
        M_total = ft.reduce(np.dot,[M1,M2,M3])
        M_trace = np.trace(M_total)
        # Calculate Beta from the trace of the total transmission matrix
        Beta = np.arccos(np.complex(M_trace/2))/np.pi
        Beta_real = np.real(Beta)
        #Beta = np.arccos(np.absolute(M_trace)/2)/np.pi
        Beta_array.append(Beta_real)
```

```
    # Calculate the first order quadrupolar resonant frequency from Beta
    K1_freq_rad = [x*baseF_rad for x in Beta_array]
    K1_freq_kHz = [y/(2*np.pi*1000) for y in K1_freq_rad]

    return K1_freq_kHz

#%% Plot the results of the calculation with respect to ion m/z for different base frequencies
mass_array = np.linspace(125,250,500)     # Define the mass range in Da

#Plot the predicted K1 frequencies vs. m/z using the K1freq_calc function
#Inputs are the mass array (in Da), base frequency (kHz), t1, and t3
fig, ax = plt.subplots()#
line1, = ax.plot(mass_array,K1freq_calc(mass_array,500,0.52,0.45),color="#FF800E",label=" ")
line2, = ax.plot(mass_array,K1freq_calc(mass_array,525,0.52,0.45),color="#C85200",label=" ")
line3, = ax.plot(mass_array,K1freq_calc(mass_array,550,0.52,0.45),color="#5F9ED1",label=" ")
line4, = ax.plot(mass_array,K1freq_calc(mass_array,575,0.52,0.45),color="#006BA4",label=" ")
line5, = ax.plot(mass_array,K1freq_calc(mass_array,600,0.52,0.45),color="#595959",label=" ")
legend1 = ax.legend(handles=[line1,line2,line3,line4,line5],bbox_to_anchor=(0.65, 0.6))
ax.add_artist(legend1)
plt.xlabel("m/z")
plt.ylabel("First order parametric frequency (kHz)")

# Plot the experimental values
m_exp = [133,156,161,166,182,190,232]
w1_exp = [291,225,214,204,178,168,126]  # 48/55 Y base 500 kHz
w2_exp = [260,203,196,187,164,155,116]  # 48/55 Y base 525 kHz
w3_exp = [234,183,175,170,146,135,104]  # 48/55 Y base 550 kHz
w4_exp = [220,172,163,154,139,124,97]   # 48/55 Y base 575 kHz
w5_exp = [200,159,150,137,127,110,87]   # 48/55 Y base 600 kHz

e = [6,8,8,9,10,10,13]        # approximate errors in frequency for each m/z
line6 = ax.errorbar(m_exp,w1_exp,yerr=e,fmt="o",elinewidth=1,capsize=2,color="#FF800E",label="500 kHz")
line7 = ax.errorbar(m_exp,w2_exp,yerr=e,fmt="o",elinewidth=1,capsize=2,color="#C85200",label = "525 kHz")
line8 = ax.errorbar(m_exp,w3_exp,yerr=e,fmt="o",elinewidth=1,capsize=2,color="#5F9ED1",label = "550 kHz")
line9 = ax.errorbar(m_exp,w4_exp,yerr=e,fmt="o",elinewidth=1,capsize=2,color="#006BA4",label = "575 kHz")
line10 = ax.errorbar(m_exp,w5_exp,yerr=e,fmt="o",elinewidth=1,capsize=2,color="#595959",label = "600 kHz")
legend2 = ax.legend(handles=[line6,line7,line8,line9,line10], bbox_to_anchor=(0.74, 0.6))
ax.add_artist(legend2)
plt.savefig("freq_trends.png",dpi=1200)
plt.show()
```

```
#%% Plot the calculated K1 frequencies vs. m/z for different duty cycles

# Plot the predicted first order resonant frequencies for the Y dimension
fig, ax = plt.subplots()
line1, = ax.plot(mass_array,K1freq_calc(mass_array,550,0.50,0.44),color="#FF800E",label=" ")
line2, = ax.plot(mass_array,K1freq_calc(mass_array,550,0.50,0.45),color="#C85200",label=" ")
line3, = ax.plot(mass_array,K1freq_calc(mass_array,550,0.50,0.46),color="#5F9ED1",label=" ")
line4, = ax.plot(mass_array,K1freq_calc(mass_array,550,0.50,0.47),color="#006BA4",label=" ")
line5, = ax.plot(mass_array,K1freq_calc(mass_array,550,0.50,0.48),color="#595959",label=" ")
legend1 = ax.legend(handles=[line1,line2,line3,line4,line5],bbox_to_anchor=(0.69, 0.6))
ax.add_artist(legend1)
plt.xlabel("m/z")
plt.ylabel("First order parametric frequency (kHz)")

# Plot the experimental first order resonant frequencies for the Y dimension
m_exp = [133,156,161,166,182,190,232]
w1_exp = [240,199,186,175,160,144,117]  # 50/56 Y base 550 kHz
w2_exp = [249,208,196,184,170,155,125]  # 50/55 Y base 550 kHz
w3_exp = [258,215,203,192,177,163,132]  # 50/54 Y base 550 kHz
w4_exp = [268,221,212,202,182,173,138]  # 50/53 Y base 550 kHz
w5_exp = [278,231,221,208,188,180,145]  # 50/52 Y base 550 kHz
e = [6,7,7,7,8,8,9.5]         # approximate errors in frequency for each m/z
line6 = ax.errorbar(m_exp,w1_exp,yerr=e,fmt="o",elinewidth=1,capsize=2,color="#FF800E",label = "50:44")
line7 = ax.errorbar(m_exp,w2_exp,yerr=e,fmt="o",elinewidth=1,capsize=2,color="#C85200",label = "50:45")
line8 = ax.errorbar(m_exp,w3_exp,yerr=e,fmt="o",elinewidth=1,capsize=2,color="#5F9ED1",label = "50:46")
line9 = ax.errorbar(m_exp,w4_exp,yerr=e,fmt="o",elinewidth=1,capsize=2,color="#006BA4",label = "50:47")
line10 = ax.errorbar(m_exp,w5_exp,yerr=e,fmt="o",elinewidth=1,capsize=2,color="#595959",label = "50:48")
legend2 = ax.legend(handles=[line6,line7,line8,line9,line10], bbox_to_anchor=(0.78, 0.6))
ax.add_artist(legend2)
#plt.title("duty cycles trends of Y resonant frequencies")
plt.savefig("Y_DC_trends.png",dpi=1200)
plt.show()
```

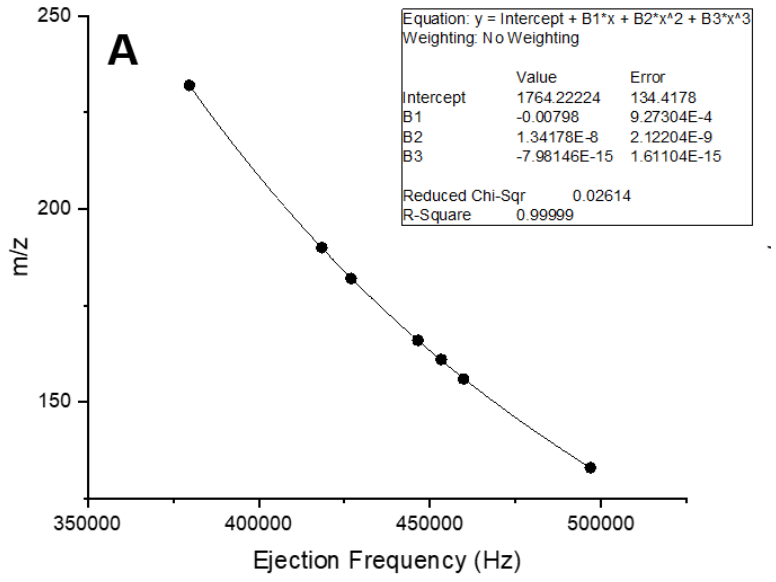
```
#%%
# Plot the predicted first order resonances for the X dimension
fig, ax = plt.subplots()
line1, = ax.plot(mass_array,K1freq_calc(mass_array,550,0.44,0.50),color="#FF800E",label=" ")
line2, = ax.plot(mass_array,K1freq_calc(mass_array,550,0.45,0.50),color="#C85200",label=" ")
line3, = ax.plot(mass_array,K1freq_calc(mass_array,550,0.46,0.50),color="#5F9ED1",label=" ")
line4, = ax.plot(mass_array,K1freq_calc(mass_array,550,0.47,0.50),color="#006BA4",label=" ")
line5, = ax.plot(mass_array,K1freq_calc(mass_array,550,0.48,0.50),color="#595959",label=" ")
legend1 = ax.legend(handles=[line1,line2,line3,line4,line5],bbox_to_anchor=(0.83, 1))
ax.add_artist(legend1)
plt.xlabel("m/z")
plt.ylabel("First order parametric frequency (kHz)")

# Plot the experimental first order resonant frequencies for the X dimension
w6_exp = [326,272,263,255,230,220,184]  # 50/56 X base 550 kHz
w7_exp = [321,267,258,250,226,217,180]  # 50/55 X base 550 kHz
w8_exp = [316,263,252,244,221,212,175]  # 50/54 X base 550 kHz
w9_exp = [310,256,247,239,217,207,172]  # 50/53 X base 550 kHz
w10_exp = [304,251,242,234,212,203,167]  # 50/52 X base 550 kHz
e = [4,3,3,3,3,3,2]        # approximate errors in frequency for each m/z
line6 = ax.errorbar(m_exp,w6_exp,yerr=e,fmt="o",elinewidth=1,capsize=2,color="#FF800E",label = "50:44")
line7 = ax.errorbar(m_exp,w7_exp,yerr=e,fmt="o",elinewidth=1,capsize=2,color="#C85200",label = "50:45")
line8 = ax.errorbar(m_exp,w8_exp,yerr=e,fmt="o",elinewidth=1,capsize=2,color="#5F9ED1",label = "50:46")
line9 = ax.errorbar(m_exp,w9_exp,yerr=e,fmt="o",elinewidth=1,capsize=2,color="#006BA4",label = "50:47")
line10 = ax.errorbar(m_exp,w10_exp,yerr=e,fmt="o",elinewidth=1,capsize=2,color="#595959",label = "50:48")
legend2 = ax.legend(handles=[line6,line7,line8,line9,line10], bbox_to_anchor=(1, 1))
ax.add_artist(legend2)
#plt.title("duty cycle trends of X resonant frequencies")
plt.savefig("X_DC_trends.png",dpi=1200)
plt.show()
```

A3: Concentrations of amino acid and polypeptide solution from Chapter 4.

Identity	**Gly_2**	**His**	**Ala_2**	**Phe**	**Tyr**	**Gly_3**	**Ala_3**
Nominal m/z	133	156	161	166	182	190	232
Concentration (mM)	211	105	105	53	211	211	105

Individual concentrations of the protonated amino acids and polypeptides in the 1 mM solution.

A4: Mass calibration settings for Chapter 4 spectra



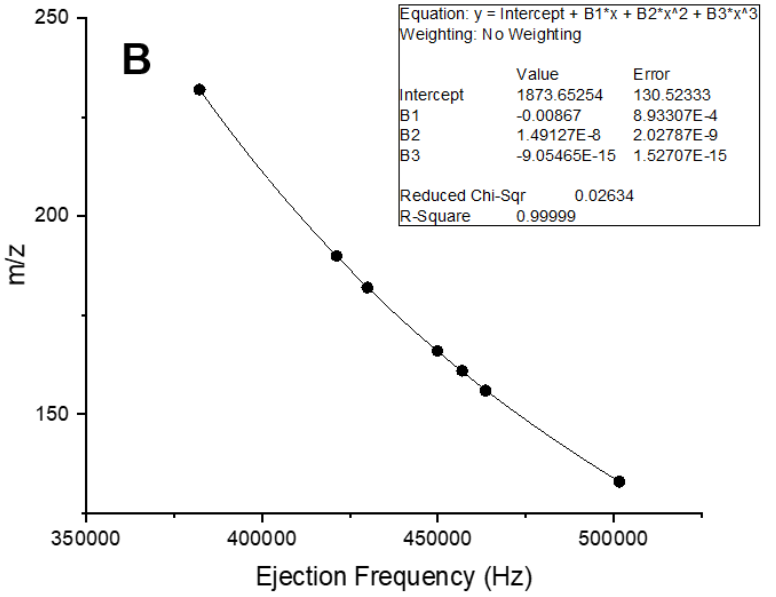


Figure A1. Plot of nominal m/z versus ejection RF frequency for the mass-selective instability scan, along with the cubic polynomial fit for mass calibration. Plot A is the mass calibration used for Figures 4.1 and 4.4, Plot B is the mass calibration for Figure 4.5. A new mass calibration was performed due to adjustment of instrument settings over time which caused a slight drift in ejection frequency.

A5: Comparison of parametric resonance-enhanced mass scan performance at different duty cycles and scan rates

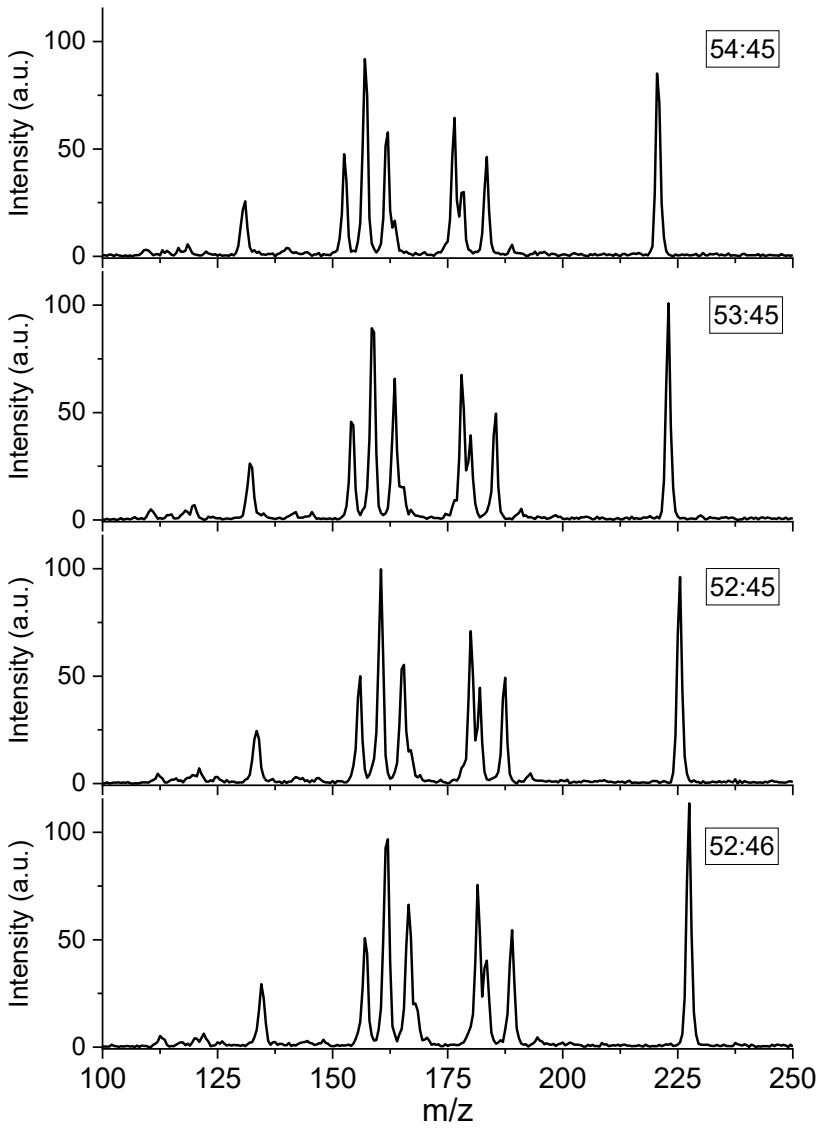


Figure A2. Parametric resonance-enhanced scans at different X:Y duty cycles, shown in the upper right. The experimentally determined relationship used here was: $auxiliary\ frequency\ (kHz) = 1.12(RF\ in\ kHz) - 109$. The average FWHM values from top to bottom are 1.31, 1.29, 1.33, and 1.37, and the average intensity values are 59.8, 63.0, 63.7, and 69.5, respectively.

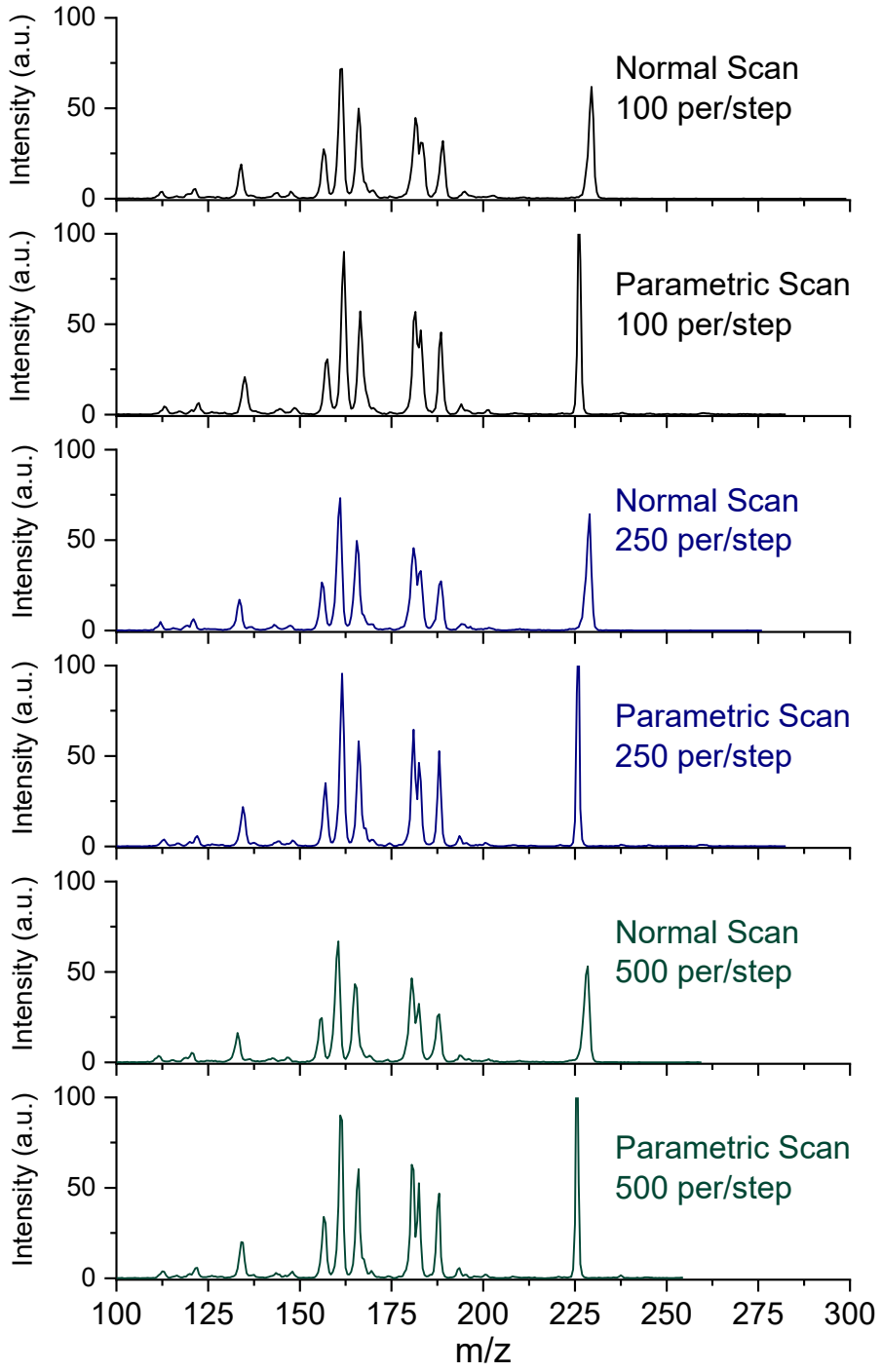


Figure A3. Comparison of parametric resonance-enhanced scan performance at different scan rates. The number of periods of the RF waveform spent at each frequency step was adjusted from 100 – 500 for the normal RF scan and the parametric resonance-enhanced scan. The experimentally determined relationship used here was: $auxiliary\ frequency\ (kHz) = 1.12(RF\ in\ kHz) - 109$ and the duty cycles were 52:45.

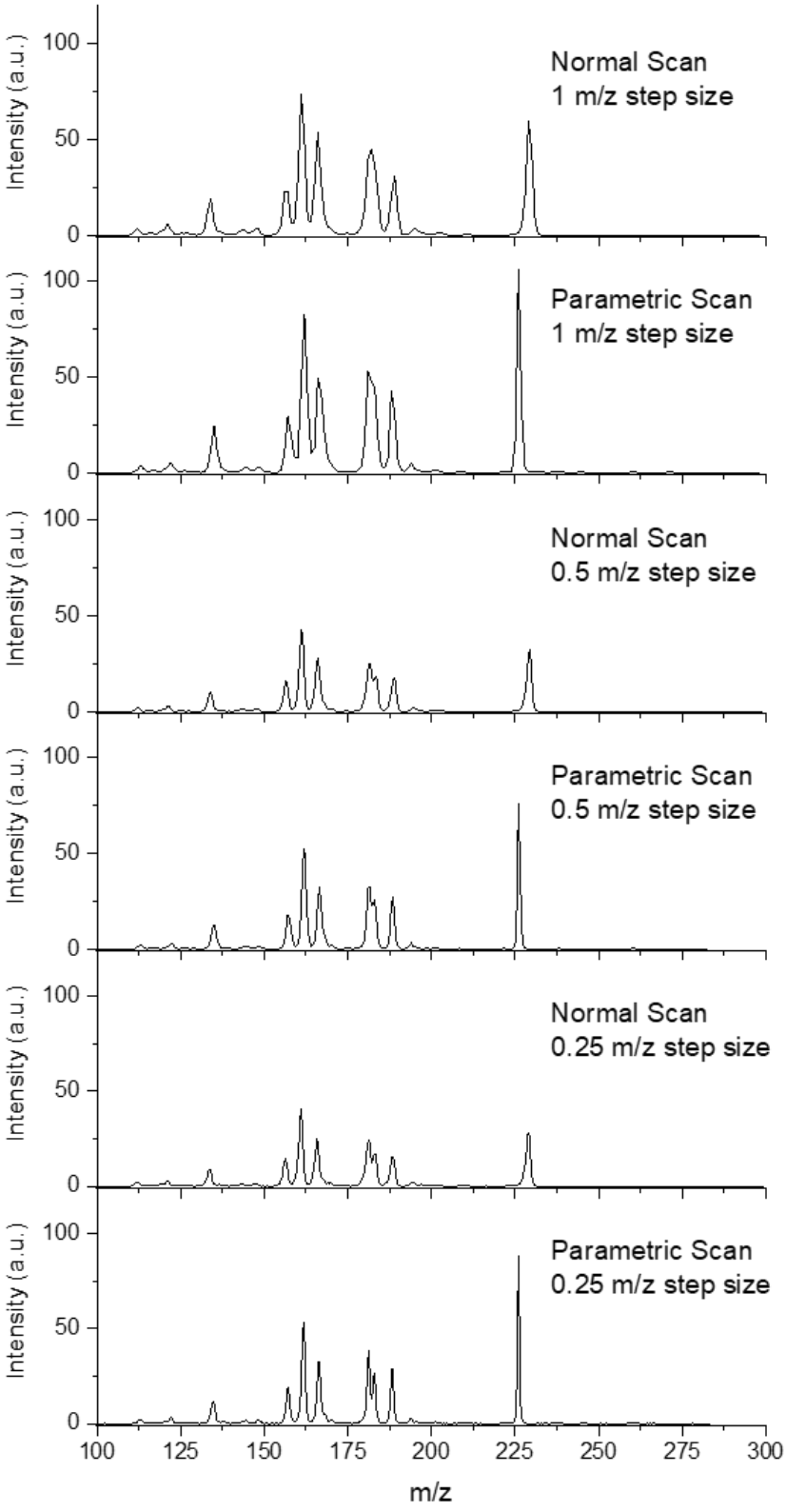


Figure A4. Comparison of parametric resonance-enhanced scan performance at different scan rates. Here, the m/z step size was adjusted and the number of periods per frequency step was fixed to 100. The duty cycles were 52:45 and the experimentally determined linear scan used was: $auxiliary\ frequency\ (kHz) = 1.12(RF\ in\ kHz) - 109.$

A6: Screenshots of the interval code in Arduino programs A-E, described in Chapter 5 section 5.4.

Program A: *2021-7-20-LIT_Mass_Scan_Only*

```
/*////////////////////////////////////////
  INTERVAL CODE
  ////////////////////////////////////////*/

void interval0()  //Exit Loop Interval
{
  digitalWrite(visual0, HIGH);                              // indicate serial communication is active
  if (Serial.available() > 0)                               // detect if serial data is available, value of 0 is empty
  {
    readSerialData();                                       // there is serial data available--read it in
  }
  digitalWrite(visual0, LOW);                               // indicate serial communication is no longer active
}

void interval1()  //Enter Loop Interval
{
  digitalWrite(visual0, HIGH);                              // indicate serial communication is active
  if (Serial.available() > 0)                               // detect if serial data is available, value of 0 is empty
  {
    readSerialData();                                       // there is serial data available--read it in
  }
  else if (interval != 0)
  {
    interval = trappingInterval;
  }
  digitalWrite(visual0, LOW);                               // indicate serial communication is no longer active
}
```

```
void interval2() //Mass Collection Trapping Interval
{
  Serial.print("Trapping for ");
  Serial.print((trapDuration));
  Serial.print("ms at ");
  Serial.print(trapFreq / 1000);
  Serial.print("kHz with ");
  Serial.print(DACTrapDuty2 * 100);
  Serial.print("%/");
  Serial.print(DACTrapDuty3 * 100);
  Serial.println("% Duty Cycle");
  volatile int trapDurationDummy = trapDuration * 1000;   // dummy variable to fix user input for trapDuration

  DACCMD2 = DACcalcVari(calAmp2, calCor2, DACTrapDuty2, 2);  // determine DAC2 command for holding
  DACCMD3 = DACcalcVari(calAmp3, calCor3, DACTrapDuty3, 1);  // determine DAC3 command for holding
  commandSyncFreqChange(trapFreq, DDSAND1);                // synchronously change filter frequency
  syncUpdateDDS(xfrFltDDSmask);                                 // perform synchronous output frequency updates for both DDS modules
  cmdDACSyncOut(DACXFROutputCMD, DACCS2, DACCMD2);              //
  cmdDACSyncOut(DACXFROutputCMD, DACCS3, DACCMD3);              //
  cmdDACSyncOut(DACXFROutputCMD, DACCS0, shhWord);              // turn off DAC0 dipolar output
  cmdDACSyncOut(DACXFROutputCMD, DACCS1, shhWord);              // turn off DAC1 dipolar output
  latchFilterDACs();                                            // latch all DACs associated with the filter
  latchXfrDACs();                                 // latch all DACs associated with the transfer

  //digitalWrite(signal42, HIGH);
  microsDelay(trapDurationDummy);
  interval = filterInterval;                                 // send system to holding interval that monitors serial communication

}
```

```
void interval4()  // Downward Stepped MASS Scan Interval
{
  Serial.print("Mass Scan from ");
  Serial.print((scanStartMass));
  Serial.print("m/z to ");
  Serial.print(scanStopMass);
  Serial.print("m/z with ");
  Serial.print(scanMassDelta);
  Serial.print("m/z steps using ");
  Serial.print((DACScanDuty2) * 100);
  Serial.print("%/");
  Serial.print((DACScanDuty3) * 100);
  Serial.println("% Duty Cycle");
  double cliffFreq;                                          // frequency value for target m/z at cliff edge q value
  double cliffExPeriod;                                      // duration of excitation at cliff edge
  double deepExPeriod;
  double MinMz;
  int scanHoldCliff;
  int scanHoldDeep;
  int totalMassSteps = (((scanStopMass - scanStartMass) / scanMassDelta) + 1);

  DACCMD2 = DACcalc(DACScanDuty2, 2);                        // determine DAC2 command for analysis
  DACCMD3 = DACcalc(DACScanDuty3, 1);                        // determine DAC3 command for analysis
  commandSyncFreqChange(YholdFreq, DDSAND2);                  // synchronously change Frag frequency
  cmdDACSyncOut(DACXFROutputCMD, DACCS2, DACCMD2);           //
  cmdDACSyncOut(DACXFROutputCMD, DACCS3, DACCMD3);           //
  latchFilterDACs();                                           // latch all DACs associated with the filter

  digitalWrite(signal42, HIGH);

  double currentmz = scanStartMass;                          // set current m/z value to scan start value
  MinMz = qCalcMinMZ(qCliff, maxSerialFreq);
  if (currentmz <= MinMz)
  {
    scanFireball = 1;
    interval = exitLoopInterval;                                    // set next interval for holding
    Serial.println(donahoeWarning);
    Serial.print("Minimum Acceptable m/z = ");
    Serial.println(MinMz);
    Serial.println("Exiting loop...");
  }
else
{
  for (int mzInt = 0; mzInt < totalMassSteps; mzInt = mzInt + 1)  // count until total frequency steps has been reached
  {
    cliffFreq = qCalcMulti(qCalcConstantFilter, qCliff, currentmz);
    cliffExPeriod = ((1 / cliffFreq) * triggerUpPeriods) * 1000000.0;  // determine duration of exitation by waveform period and required number of waveforms
    deepExPeriod = ((1/cliffFreq) * triggerDownPeriods) * 1000000.0;

    if (cliffExPeriod <= triggerUpTime)                        // compare the cliff excitation period to the alternative minimum time tax
    {
      scanHoldCliff = triggerUpTime;                                  // set the scanHold parameter to one, resulting in the alternative minimum time tax
    }
    else
    {
      scanHoldCliff = cliffExPeriod;              // set the scanHold parameter to the determined value less the time tax
    }
    currentmz = currentmz + scanMassDelta;                    // increment the current m/z value by the delta mass value
    digitalWrite(scanTTL, HIGH);                               // signal mass scan in progress
    commandFreqChange(cliffFreq, DDSAND1, DDSUPD1);           // pass new frequency to AD9851
    commandFreqChange(YholdFreq, DDSAND2,DDSUPD2);                  // keep Frag frequency (Y-axis) at the set value
    digitalWrite(triggerTTL, HIGH);                             // signal ion ejection to data acquisition system
    microsDelay(scanHoldCliff);                               // initiate microsDeleay to hold waveform


    if (deepExPeriod <= triggerDownTime)
    {
      scanHoldDeep = triggerDownTime;
    }
    else
    {
      scanHoldDeep = deepExPeriod;
    }
    digitalWrite(triggerTTL, LOW);                              // signal end of data acquisition at currentMZ
    microsDelay(scanHoldDeep);                                  // delay for data transmission
  }

  digitalWrite(scanTTL, LOW);                                 // signal end of mass scan
  digitalWrite(signal42, LOW);
  interval = crapInterval;                                   // return to the holding pattern
 }
}
```

Program B: *2024-5-30-LIT_Parametric_Mass_Isolation*

Only interval 3 is shown here, since it contains the code for parametric mass isolation. Intervals 0, 1, and 4 remain the same as above for Program A.

```
2596  void interval3() // Combined Apex and Parametric Excitation Mass Filter Interval
2597  {
2598    // First do a high and low mass pass filter using base frequency and duty cycle settings --> apex filtering
2599    digitalWrite(signal42,LOW);
2600    Serial.print("Apex filtering for ");
2601    Serial.print(filterDuration);
2602    Serial.print("us with duty cycle ");
2603    Serial.print(DACFilterDuty0 * 100);
2604    Serial.print("%/");
2605    Serial.print(DACFilterDuty1 * 100);
2606    Serial.print("% and frequency ");
2607    Serial.print(filterFreq / 1000);
2608    Serial.println(" kHz");
2609    volatile int filterDurationDummy = filterDuration * 1000; // dummy variable to fix user input for filterDuration
2610
2611    // Do a simultaneous duty cycle and frequency switch to isolate a small range of m/z (apex isolation, DAWI)
2612    DACCMD2 = DACcalcVari(calAmp2, calCor2, DACFilterDuty0, 2);  // determine DAC2 command for holding
2613    DACCMD3 = DACcalcVari(calAmp3, calCor3, DACFilterDuty1, 1);  // determine DAC3 command for holding
2614    commandSyncFreqChange(filterFreq, DDSAND1);                // synchronously change filter frequency
2615    syncUpdateDDS(xfrFltDDSmask);                                 // perform synchronous output frequency updates for both DDS modules
2616    cmdDACSyncOut(DACXFROutputCMD, DACCS2, DACCMD2);              //
2617    cmdDACSyncOut(DACXFROutputCMD, DACCS3, DACCMD3);              //
2618    latchFilterDACs();                                            // latch all DACs associated with the filter
2619    microsDelay(filterDurationDummy);
2620
```

```
2621    // Next do a short scan of the auxiliary frequency to resonantly filter out nearby mass peaks and get better mass isolation
2622    Serial.print("Parametric filtering from ");
2623    Serial.print(scanStartFreq/1000);
2624    Serial.print("kHz to ");
2625    Serial.print(scanStopFreq/1000);
2626    Serial.print(" kHz for ");
2627    Serial.print(scanDeltaFreq/1000);
2628    Serial.print(" kHz, ");
2629    Serial.print(scanRate);
2630    Serial.print(" us steps with base ");
2631    Serial.print(baseFreq/1000);
2632    Serial.print(" kHz, ");
2633    Serial.print(DACScanDuty0 * 100);
2634    Serial.print("%/");
2635    Serial.println(DACScanDuty1 * 100);
2636    Serial.print("Isolating the peak at aux frequency ");
2637    Serial.print(isolatedFreq/1000);
2638    Serial.print(" +/- ");
2639    Serial.print((isolatedRange/1000)/2);
2640    Serial.println(" kHz");
2641    double currentFreq = scanStartFreq;                                    // set current secular frequency value to scan start value input by the user
2642    int totalFreqSteps = (((scanStartFreq - scanStopFreq) / scanDeltaFreq) + 1);    // total number of parametric frequency steps determined by user input
2643
2644    // Below is the code for doing a short parametric frequency sweep around the isolated m/z
2645    // Turn on the auxiliary TTL output (transfer) to do filtering
2646    DACCMD2 = DACcalc(DACScanDuty0, 2);                    // determine DAC2 command for analysis
2647    DACCMD3 = DACcalc(DACScanDuty1, 1);                    // determine DAC3 command for analysis
2648    cmdDACSyncOut(DACXFROutputCMD, DACCS2, DACCMD2);       // turn on DAC2 for base waveform output
2649    cmdDACSyncOut(DACXFROutputCMD, DACCS3, DACCMD3);       // turn on DAC3 for base waveform output
2650    latchFilterDACs();                                     // latch all DACs associated with the filter
2651
```

```
// Scan the parametric frequency from user input start (scanStartFreq) and end (scanStopFreq) values with a step size (scanDeltaFreq)
if (currentFreq >= maxSerialFreq)                     // if the user input frequency is higher than the hard-coded maximum frequency, exit the loop
{                                                        // (so we don't break stuff by switching to a frequency which is too high for the circuit)
  scanFireball = 1;
  interval = exitLoopInterval;
  Serial.println(donahoeWarning);
  Serial.print("Maximum Acceptable Frequency = ");
  Serial.println(maxSerialFreq);
  Serial.println("Exiting loop...");
}
else
{
  commandFreqChange(baseFreq, DDSAND1, DDSUPD1);                          // pass new STATIC base frequency to DDS1 Filter outputs (r1)
  commandFreqChange(currentFreq, DDSAND0, DDSUPD0);                       // pass new SCANNED parametric frequency to DDS0 Transfer outputs
  for (int freqInt = 0; freqInt < totalFreqSteps; freqInt = freqInt + 1)  // count until total parametric frequency steps has been reached
  {
    if (currentFreq <= (isolatedFreq+(0.5*isolatedRange)) and currentFreq >= (isolatedFreq-(0.5*isolatedRange)))
    {
      // If the frequency is within the selected range, turn off the auxiliary TTL to keep that peak!
      cmdDACSyncOut(DACXFROutputCMD, DACCS0, shhWord);          // turn off DAC0 auxiliary output - DAC0
      cmdDACSyncOut(DACXFROutputCMD, DACCS1, shhWord);          // turn off DAC1 auxiliary output - DAC1
      latchXfrDACs();                                           // latch all DACs associated with the transfer
    }
    else
    {
      // Keep the auxiliary signal on to filter out nearby unwanted peaks
      cmdDACSyncOut(DACXFROutputCMD, DACCS0, DACCMD0);          // turn on the auxiliary TTL from the transfer outputs - DAC0
      cmdDACSyncOut(DACXFROutputCMD, DACCS1, DACCMD1);          // turn on the auxiliary TTL from the transfer outputs - DAC1
      latchXfrDACs();                                           // latch all DACs associated with the transfer
      microsDelay(scanRate);                                    // sit at that frequency for the user input length of time
    }
    currentFreq = currentFreq - scanDeltaFreq;                 // increment the value for the parametric frequency
    commandFreqChange(currentFreq, DDSAND0, DDSUPD0);          // pass new SCANNED parametric frequency to DDS0 Transfer outputs
  }
}
```

Program C: *2023-7-3-LIT_Parametric_Mass_Elimination*

Only interval 3 is shown here, since it contains the code for parametric mass elimination of one m/z species. Intervals 0, 1, and 4 remain the same as above for Program A.

```
void interval3() // Parametric Mass Elimination Interval
{
  Serial.print("Filtering for ");
  Serial.print((filterDuration));
  Serial.print("ms with dipolar freq ");
  Serial.print(auxFreq);
  Serial.print(" Hz and base waveform at ");
  Serial.print(filterFreq/1000);
  Serial.print("kHz ");
  Serial.print(DACScanDuty0 * 100);
  Serial.print("%/");
  Serial.println(DACScanDuty1 * 100);
  volatile int filterDurationDummy = filterDuration*1000; // dummy variable to fix user input for filterDuration

  cmdDACSyncOut(DACXFROutputCMD, DACCS0, DACCMD0);          // turn on the auxiliary TTL from the transfer outputs - DAC0
  cmdDACSyncOut(DACXFROutputCMD, DACCS1, DACCMD1);          // turn on the auxiliary TTL from the transfer outputs - DAC1
  DACCMD2 = DACcalc(DACScanDuty0, 2);                       // determine DAC2 command for analysis
  DACCMD3 = DACcalc(DACScanDuty1, 1);                       // determine DAC3 command for analysis
  cmdDACSyncOut(DACXFROutputCMD, DACCS2, DACCMD2);          // turn on DAC2 for base waveform output
  cmdDACSyncOut(DACXFROutputCMD, DACCS3, DACCMD3);          // turn on DAC3 for base waveform output
  DACCMD0 = DACcalcVari(calAmp0, calCor0, DACTrapDuty0, 1); // determine DAC0 command for holding
  DACCMD1 = DACcalcVari(calAmp1, calCor1, DACTrapDuty1, 2); // determine DAC1 command for holding
  commandSyncFreqChange(filterFreq, DDSAND1);               // synchronously change filter frequency
  commandSyncFreqChange(auxFreq, DDSAND0);                  // synchronously change transfer frequency to the user input auxiliary frequency
  syncUpdateDDS(xfrFltDDSmask);                             // perform synchronous output frequency updates for DDS1 (base waveform)
  latchFilterDACs();                                        // latch all DACs associated with the filter
  latchXfrDACs();                                           // latch all DACs associated with transfer
  microsDelay(filterDurationDummy);                         // delay for the user input time period
  cmdDACSyncOut(DACXFROutputCMD, DACCS0, shhWord);          // turn off DAC0 auxiliary output
  cmdDACSyncOut(DACXFROutputCMD, DACCS1, shhWord);          // turn off DAC1 auxiliary output
  latchXfrDACs();                                           // latch all DACs associated with the transfer
  interval = scanInterval;                                  // send system to the mass scan interval

}
```

Program D: *2023-10-24_Parametric_Mass_Scan*

Only interval 4 is shown here, since this code contains the combined base and auxiliary frequency scan in that interval. Intervals 0 and 1 are the same for all programs, and interval 3 is not used in this code.

```
2681  void interval4()  // Downward Stepped Mass Scan Interval WITH Parametric Excitation
2682  {
2683    Serial.print("Mass Scan from ");                  // print updated user variables in the serial monitor
2684    Serial.print((scanStartMass));
2685    Serial.print("m/z to ");
2686    Serial.print(scanStopMass);
2687    Serial.print("m/z with ");
2688    Serial.print(scanMassDelta);
2689    Serial.print("m/z steps using ");
2690    Serial.print((DACScanDuty2) * 100);
2691    Serial.print("%/");
2692    Serial.print((DACScanDuty3) * 100);
2693    Serial.print("% Duty Cycle");
2694    Serial.println("");
2695    Serial.print("Auxiliary waveform Beta #1: ");
2696    Serial.print(Div1);
2697    Serial.print(" Beta #2: ");
2698    Serial.println(Div2);
2699    double cliffFreq;                                 // frequency value for target m/z at cliff edge q value
2700    double auxFreq;                                   // frequency of auxiliary waveform
2701    double cliffExPeriod;                             // duration of excitation at cliff edge (data bin)
2702    double deepExPeriod;                              // duration of time in deep well (time in between bins)
2703    double MinMz;                                     // minimum m/z value determined by frequency limits of circuit
2704    int scanHoldCliff;                                // length of time to hold for m/z frequency excitation
2705    int scanHoldDeep;                                 // length of time to hold for data acquisition
2706    int totalMassSteps = (((scanStopMass - scanStartMass) / scanMassDelta) + 1);    // total number of mass steps determined by user input
```

```
2707
2708    DACCMD2 = DACcalc(DACScanDuty2, 2);                       // determine DAC2 command for analysis
2709    DACCMD3 = DACcalc(DACScanDuty3, 1);                       // determine DAC3 command for analysis
2710    DACCMD0 = DACcalc(DACScanDuty0, 1);                       // determine DAC0 command for analysis
2711    DACCMD1 = DACcalc(DACScanDuty1, 2);                       // determine DAC1 command for analysis
2712    cmdDACSyncOut(DACXFROutputCMD, DACCS2, DACCMD2);          // turn on DAC2 for base waveform (filter outputs)
2713    cmdDACSyncOut(DACXFROutputCMD, DACCS3, DACCMD3);          // turn on DAC3 for base waveform (filter outputs)
2714    cmdDACSyncOut(DACXFROutputCMD, DACCS0, DACCMD0);          // turn on DAC0 auxiliary output
2715    cmdDACSyncOut(DACXFROutputCMD, DACCS1, DACCMD1);          // turn on DAC1 auxiliary output
2716    latchXfrDACs();                                           // latch all DACs associated with the Transfer outputs
2717    latchFilterDACs();                                        // latch all DACs associated with the Filter outputs
2718    digitalWrite(signal42, HIGH);                             // turn signal 42 HIGH to trigger the external divide-by-N board
2719    double currentmz = scanStartMass;                         // set current m/z value to scan start value input by the user
2720    double auxMZ = scanStartMass;
2721    MinMz = qCalcMinMZ(qCliff, maxSerialFreq);
2722    if (currentmz <= MinMz)                                   // if the user input minimum m/z is lower than the allowed minimum, exit the loop
2723    {                                                         // (so we don't break stuff by switching to a frequency which is too high for the circuit)
2724      scanFireball = 1;
2725      interval = exitLoopInterval;
2726      Serial.println(donahoeWarning);
2727      Serial.print("Minimum Acceptable m/z = ");
2728      Serial.println(MinMz);
2729      Serial.println("Exiting loop...");
2730    }
```

```
2731    else
2732    {
2733      cliffFreq = qCalcMulti(qCalcConstantFilter, qCliff, currentmz);
2734      auxFreq = qCalcMulti(qCalcConstantFilter, qCliff, currentmz)*(Div1-(Div2*currentmz));
2735      commandFreqChange(cliffFreq, DDSAND1, DDSUPD1);                    // pass new base frequency to DDS1 Filter outputs
2736      commandFreqChange(auxFreq, DDSAND0, DDSUPD0);                      // pass new auxiliary frequency to DDS0 Transfer outputs
2737      for (int mzInt = 0; mzInt < totalMassSteps; mzInt = mzInt + 1)     // count until total frequency (m/z) steps has been reached
2738      {
2739        digitalWrite(scanTTL, HIGH);                                     // signal mass scan in progress (SCAN output, triggers the stanford box)
2740        // BEGIN DATA BIN
2741        digitalWrite(triggerTTL, HIGH);                                  // signal ion ejection to data acquisition system (TRIGGER output, goes to Pico for data analysis)
2742        cliffExPeriod = ((1 / cliffFreq) * triggerUpPeriods) * 1000000.0;  // determine duration of exitation by waveform period and required number of waveforms (length of bin)
2743        deepExPeriod = ((1/cliffFreq) * triggerDownPeriods) * 1000000.0;   // determine duration of time in between frequency bins
2744        if (cliffExPeriod <= triggerUpTime)     // compare the cliff excitation period to the alternative minimum time tax
2745        {
2746          scanHoldCliff = triggerUpTime;        // set the scanHold parameter to triggerUpTime, resulting in the alternative minimum time tax
2747        }
2748        else
2749        {
2750          scanHoldCliff = cliffExPeriod;        // set the scanHold parameter to the determined value if it is greater than the time tax
2751        }
2752        microsDelay(scanHoldCliff);             // initiate microsDelay to hold waveform, this is the length of the data bin
2753        if (deepExPeriod <= triggerDownTime)    // compare the deep well period to the alternative minimum time tax
2754        {
2755          scanHoldDeep = triggerDownTime;       // set the scanHold parameter to triggerDownTime, resulting in the alternative minimum time tax
2756        }
2757        else
2758        {
2759          scanHoldDeep = deepExPeriod;          // set the scanHold parameter to the determined value if it is greater than the time tax
2760        }
2761        // END DATA BIN
2762        digitalWrite(triggerTTL, LOW);                            // signal end of data acquisition at current m/z
2763        microsDelay(scanHoldDeep);                                // delay to apply auxiliary for longer in between bins, hold the last base frequency
2764        currentmz = currentmz + scanMassDelta;                    // increment the current m/z value by the delta mass value
2765        auxMZ = auxMZ + scanMassDelta;                            // increment the m/z value for the auxiliary frequency
2766        auxFreq = qCalcMulti(qCalcConstantFilter, qCliff, currentmz)*(Div1-(Div2*currentmz));     //calculate the auxiliary frequency from a linear relationship
2767        cliffFreq = qCalcMulti(qCalcConstantFilter, qCliff, currentmz);    // calculate the ejection frequency for the current m/z
2768        commandFreqChange(auxFreq, DDSAND0, DDSUPD0);                     // pass new auxiliary frequency to DDS0 Transfer outputs
2769        commandFreqChange(cliffFreq, DDSAND1, DDSUPD1);                   // pass new base frequency to DDS1 Filter outputs, hold auxiliary frequency
2770        microsDelay(1);                                                   // delay 1 us for data transmission
2771      }
2772
```

```
2771      }
2772
2773      digitalWrite(scanTTL, LOW);     // signal end of mass scan
2774      digitalWrite(signal42, LOW);    // signal end of trigger for the external divide-by-N board
2775      interval = crapInterval;        // return to the holding pattern
2776    }
2777  }
```

Program E: *2025-10-8_ParaMassElimination_3regions*

Only interval 3 is shown here, since it contains the code for parametric mass elimination of three m/z species. Intervals 0, 1, and 4 remain the same as above for Program A. However, this program is externally triggered at the beginning of interval 3 by an input from a Stanford box to Signal 53.

```
2658  void interval3() // Parametric Mass Elimination Interval - 3 separate regions to eliminate 3 m/z species in one instrument cycle
2659  {
2660    Serial.print("Mass elimation regions are: ");
2661    Serial.print(filterDuration1);
2662    Serial.print("ms ");
2663    Serial.print(auxFreq1/1000);
2664    Serial.print(" kHz, ");
2665    Serial.print(filterDuration2);
2666    Serial.print("ms ");
2667    Serial.print(auxFreq2/1000);
2668    Serial.print(" kHz, ");
2669    Serial.print(filterDuration3);
2670    Serial.print("ms ");
2671    Serial.print(auxFreq3/1000);
2672    Serial.print(" kHz, base frequency: ");
2673    Serial.print(filterFreq/1000);
2674    Serial.print(" kHz ");
2675    Serial.print(DACScanDuty0 * 100);
2676    Serial.print("%/");
2677    Serial.println(DACScanDuty1 * 100);
2678    volatile int filterDurationDummy1 = filterDuration1*1000;           // dummy variable to fix user input for filterDuration region #1
2679    volatile int filterDurationDummy2 = filterDuration2*1000;           // dummy variable to fix user input for filterDuration region #2
2680    volatile int filterDurationDummy3 = filterDuration3*1000;           // dummy variable to fix user input for filterDuration region #3
2681    volatile int filterDurationDummy7 = filterDuration7*1000;           // dummy variable to fix user input for filterDuration laser delay
2682    //digitalWrite(scanTTL, HIGH);                                      // Set SCAN trigger output to HIGH here if you want to see ions hit the detector while filtering
2683
2684    cmdDACSyncOut(DACXFROutputCMD, DACCS0, DACCMD0);          // turn on the auxiliary TTL from the transfer outputs - DAC0
2685    cmdDACSyncOut(DACXFROutputCMD, DACCS1, DACCMD1);          // turn on the auxiliary TTL from the transfer outputs - DAC1
2686    DACCMD2 = DACcalc(DACScanDuty0, 2);                       // determine DAC2 command for analysis
2687    DACCMD3 = DACcalc(DACScanDuty1, 1);                       // determine DAC3 command for analysis
2688    cmdDACSyncOut(DACXFROutputCMD, DACCS2, DACCMD2);          // turn on DAC2 for base waveform output
2689    cmdDACSyncOut(DACXFROutputCMD, DACCS3, DACCMD3);          // turn on DAC3 for base waveform output
2690    DACCMD0 = DACcalcVari(calAmp0, calCor0, DACTrapDuty0, 1);  // determine DAC0 command for holding
2691    DACCMD1 = DACcalcVari(calAmp1, calCor1, DACTrapDuty1, 2);  // determine DAC1 command for holding
2692    commandSyncFreqChange(filterFreq, DDSAND1);          // synchronously change filter frequency
2693
```

```
2693
2694    // Filtering Region #1
2695    commandSyncFreqChange(auxFreq1, DDSAND0);             // synchronously change transfer frequency to the user input auxiliary frequency
2696    syncUpdateDDS(xfrFltDDSmask);                          // perform synchronous output frequency updates for DDS0 (aux waveform)
2697    latchFilterDACs();                                     // latch all DACs associated with the filter
2698    latchXfrDACs();                                        // latch all DACs associated with transfer
2699    microsDelay(filterDurationDummy1);                      // delay for the user input time period
2700    // Filtering Region #2
2701    commandSyncFreqChange(auxFreq2, DDSAND0);             // synchronously change transfer frequency to the user input auxiliary frequency
2702    syncUpdateDDS(xfrFltDDSmask);                          // perform synchronous output frequency updates for DDS0 (aux waveform)
2703    latchFilterDACs();                                     // latch all DACs associated with the filter
2704    latchXfrDACs();                                        // latch all DACs associated with transfer
2705    microsDelay(filterDurationDummy2);                      // delay for the user input time period
2706    // Filtering Region #3
2707    commandSyncFreqChange(auxFreq3, DDSAND0);             // synchronously change transfer frequency to the user input auxiliary frequency
2708    syncUpdateDDS(xfrFltDDSmask);                          // perform synchronous output frequency updates for DDS0 (aux waveform)
2709    latchFilterDACs();                                     // latch all DACs associated with the filter
2710    latchXfrDACs();                                        // latch all DACs associated with transfer
2711    microsDelay(filterDurationDummy3);                      // delay for the user input time period
2712
2713    // Shut off the auxiliary waveform after filtering
2714    cmdDACSyncOut(DACXFROutputCMD, DACCS0, shhWord);          // turn off DAC0 auxiliary output
2715    cmdDACSyncOut(DACXFROutputCMD, DACCS1, shhWord);          // turn off DAC1 auxiliary output
2716    latchXfrDACs();                                        // latch all DACs associated with the transfer
2717
2718    // Return to stable conditions for laser interaction -- default 1 ms delay time
2719    commandSyncFreqChange(trapFreq, DDSAND1);               // synchronously change filter frequency
2720    latchFilterDACs();                                     // latch all DACs associated with the filter
2721    microsDelay(filterDurationDummy7);                      // delay for the user input time period
2722
2723    interval = scanInterval;                               // send system to the mass scan interval
2724
2725  }
2726
```