Title
New Target Methodology: Polymer-Assisted Deposition and Its Applications on Gas-Phase Nuclear Chemistry with Rutherfordium

Author
Garcia, Mitch Andre

Publication Date
2009

Peer reviewed|Thesis/dissertation

New Target Methodology: Polymer-Assisted Deposition and Its Applications on Gas-Phase Nuclear Chemistry with Rutherfordium

by

Mitch André Garcia

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Chemistry

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Heino Nitsche, Chair

Professor Joseph Cerny

Professor Joonhong Ahn

Fall 2009

The dissertation of Mitch André Garcia, titled New Target Methodology: Polymer-Assisted Deposition and Its Applications on Gas-Phase Nuclear Chemistry with Rutherfordiumthe is approved:

Chair

_____ Date _____

_____ Date _____

_____ Date _____

University of California, Berkeley

New Target Methodology: Polymer-Assisted Deposition and Its Applications on Gas-Phase Nuclear Chemistry with Rutherfordium

by

Mitch André Garcia

Abstract

New Target Methodology: Polymer-Assisted Deposition and Its Applications on Gas-Phase Nuclear Chemistry with Rutherfordium

by

Mitch André Garcia

Doctor of Philosophy of Chemistry

University of California Berkeley

Professor Heino Nitsche, Chair

This work focuses on three research topics that were distinctly different but broadly centered around developing new methods to perform transactinide gas-phase chemistry. First, the application of a new materials science methodology, Polymer-Assisted Deposition (PAD), to target manufacturing is described. Second, the construction of a new experimental apparatus to conduct gas-phase chemistry and the design of circuits and electronics for the measurement of alpha-decay energy is discussed. Third, a new nuclear reaction for the production of ^{261}Rf, an ideal candidate for gas-phase chemistry due to its relatively-long half life ($t_{1/2}$ = 69 s), is detailed.

The PAD method was used to make metal oxides of hafnium, europium, and thulium on silicon backings that were found to be homogenous and highly uniform metal oxide films. The films, after a single application of the PAD method, ranged in thickness from 30 to 320 nm. The reapplication of the PAD method on already existing metal oxide films (six times) using the 10% by weight hafnium (IV) solution resulted in equally homogeneous and uniform films with a total thickness of 600 nm.

The irradiation of targets made by the PAD method on silicon nitride backings (thickness

of 1000 nm, 344 μg/cm^2) were irradiated with an ^{40}Ar beam at a laboratory frame energy of 210 MeV (50 particle nA). The root mean squared (RMS) roughness prior to irradiation was 1.1 nm for a 250 nm (220 μg/cm2) Tm_2O_3 target, and a RMS roughness of 2.0 nm after irradiation was measured by Atomic Force Microscopy (AFM). These experiments demonstrated the resilience of the targets to heavy-ion irradiation and will prove useful when next generation ion sources come online with subsequently larger beam intensities.

The design, construction and testing of the alpha-wheel apparatus for gas-phase chemistry reactions was completed. Testing of the device was done with ^{205}Fr, produced from the nuclear reaction of ^{169}Tm(^{40}Ar, 4n)^{205}Fr. The francium atoms were quickly transported to the alpha-wheel apparatus, and a total of 164 decays were detected. They centered at an energy of 6.8 MeV which corresponded to the known 7.01 MeV alpha particle from ^{205}Fr; discrepancies in the energy are due to energy-loss from the helium atmosphere in the apparatus.

The first nuclear synthesis of Rutherfordium-261 using uranium targets (^{238}U) were conducted with the Berkeley Gas-filled Separator at the Lawrence Berkeley National Laboratory 88-Inch Cyclotron. The nuclear reaction consisted of irradiating the uranium targets with a ^{26}Mg beam. The production of ^{261}Rf went through a 3n exit channel, a channel not normally seen in hot-fusion reactions, and had a cross section of 28 pb.

In summary, this work shows that Polymer-Assisted Deposition is a good method to produce homogeneous, uniform, and crack-free metal oxide targets for nuclear science applications. It also showed that the construction of an alpha-wheel apparatus to measure the alpha-decay of radionuclides is reliable. Finally, it has been shown that the production of ^{261}Rf is possible with uranium targets at a sufficient cross-section to make a gas-phase chemistry reaction with this isotope feasible.

Table of Contents

List of Figures

Acknowledgments

The number of people who helped me complete this dissertation is quite large, and can not be adequately thanked within this small space.

My advisor, Heino Nitsche, has been a wonderful mentor throughout my graduate studies. I am truly thankful he took me into his group after I left my previous PI in synthetic chemistry for greener pastures. It was him that forced me to work on the resoundingly successful Polymer-Assisted Deposition (PAD) method for target production, when it seemed more of a fantastical idea than a successful research project. He encouraged, or at least looked the other way, while I worked on side projects in the world of Web 2.0 and my budding journalism hobby with my chemistry-blog.com website.

Ken Gregorich, who was an invaluable source of knowledge. He answered many-many questions regarding nuclear chemistry experimental design and was instrumental in any successful beamtime.

Maz Ali, Noel Chang, Jonathan Glidden, and Tashi Parsons-Moss who helped in a multitude of aspects in this dissertation. Maz worked hand-in-hand in the development of the PAD method, and took the lead in several of the circuit design aspects for the gas-phase apparatus. Noel Chang characterized the spin coater and did the first tests with the cap-on. Jonathan Glidden made circuits and helped out on beam-times. Tashi Parsons-Moss collected much of the laborious data for the PAD method chapters and taught me how to play Gin Rommy as we waited. Carolina Fineman-Sotomayor for being a wonderful office mate, conversation partner, and lunch buddy. Yung-Jin Hu for teaching me how to setup tab markers for this work.

Sarah Nelson, Jackie Gates, Irena Dragojevic, and Paul Ellison who were contemporary fellow graduate students and worked beamtime shifts with me. Thank you for making graduate

school an entertaining adventure.

Zuzanna, Dvorakova, Honza Dvorak, Cody Folden, Christoph Düllmann,and Ralf Sudowe who were senior people in the group and whose knowledge I milked for as long as I could.

Professor Cerny, who let me work with him as an undergraduate student many years ago. His advice to me in regards to joining Heino's group, after leaving synthetic chemistry, made all this seem possible.

My family, whose support emotionally and financially made my graduate student experience so pleasant. They rarely complained that I did not see them enough and have been quite proud of all my accomplishments.

Lastly, I would like to thank my beautiful and intelligent new bride Sandra Villa. She has been with me every step of my science career since I took freshman chemistry with her as an undergraduate. Her habit of studying constantly has forced me to also study, as I would do anything to simply be around her. I would not be the person I am today without her.

Chapter 1: Target Methodology and Rutherfordium Chemistry

1.1.1 Introduction

This dissertation is concerned with the study of a new target methodology with applications to transactinide chemistry, the construction of a gas-phase transactinide element chemistry apparatus, and the study of a new transactinide production reaction for ^{261}Rf. The transactinides comprise the elements between rutherfordium (Z = 104) to the recently confirmed element 114; they are found in the last period of the Periodic Table in Groups 4 through 14, and can be seen in Figure 1.1.[1] All the transactinides are inherently unstable to nuclear decay and, depending on the isotope, will decay by either alpha-decay, beta-decay, or spontaneous fission; the longest-living confirmed isotope has a half-life of ~1 min (Rf^{261}, $T_{1/2}$=75 s).[2] All transactinides are produced in a particle accelerator, whereby a beam of heavy-ions is accelerated and directed at a target; on occasion an atomic nucleus of the beam and target will fuse to form a compound nucleus of the transactinide of interest (several methods nuclear scientists use to produce targets are discussed in section 1.2.). The probability of producing the desired compound nucleus is very small; a sample calculation of the probability to produce a transactinide element is given later in this section.

The study of an element's chemistry is at the heart of all chemistry research. Chemists who study the chemical properties of the transactinides have to verify that the newly discovered elements indeed behave as one would expect for a member of their respective group on the Periodic Table. Deviations from predicted behavior need to be explored, and some examples of deviations are discussed in section 1.4. Because of the low probability of producing transactinide elements it is necessary to study their chemistry at an atom-at-a-time basis. This is usually done by separating the element of interest from the beam using velocity filters and/or

magnetic rigidity separators; this is discussed in more detail in chapter 5.

1												13	14	15	16	17	18
H 1	2																He 2
Li 3	Be 4											B 5	C 6	N 7	O 8	F 9	Ne 10
Na 11	Mg 12	3	4	5	6	7	8	9	10	11	12	Al 13	Si 14	P 15	S 16	Cl 17	Ar 18
K 19	Ca 20	Sc 21	Ti 22	V 23	Cr 24	Mn 25	Fe 26	Co 27	Ni 28	Cu 29	Zn 30	Ga 31	Ge 32	As 33	Se 34	Br 35	Kr 36
Rb 37	Sr 38	Y 39	Zr 40	Nb 41	Mo 42	Tc 43	Ru 44	Rh 45	Pd 46	Ag 47	Cd 48	In 49	Sn 50	Sb 51	Te 52	I 53	Xe 54
Cs 55	Ba 56	La 57	Hf 72	Ta 73	W 74	Re 75	Os 76	Ir 77	Pt 78	Au 79	Hg 80	Tl 81	Pb 82	Bi 83	Po 84	At 85	Rn 86
Fr 87	Ra 88	Ac 89	Rf 104	Db 105	Sg 106	Bh 107	Hs 108	Mt 109	Ds 110	Rg 111	Cn 112		114				

Ce 58	Pr 59	Nd 60	Pm 61	Sm 62	Eu 63	Gd 64	Tb 65	Dy 66	Ho 67	Er 68	Tm 69	Yb 70	Lu 71
Th 90	Pa 91	U 92	Np 93	Pu 94	Am 95	Cm 96	Bk 97	Cf 98	Es 99	Fm 100	Md 101	No 102	Lr 103

Figure 1.1 Current periodic table of the elements as of November 2009.

In a nuclear reaction, a target comprised of stationary nuclei is irradiated with heavy-ion beams (*i.e*, ^{18}O, ^{22}Ne, ^{26}Mg, ^{48}Ca) as shown schematically in Figure 1.2.

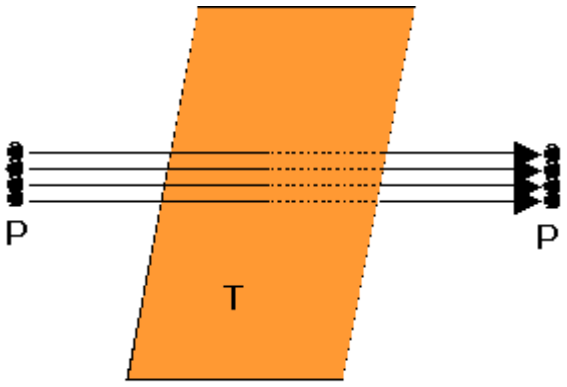

Figure 1.2 Schematic diagram of beam particles (P) going through a target (T).

The energy of the beam is often characterized in the literature as a value in the laboratory frame or as a value in the center-of-mass frame. In the laboratory frame, the projectile is characterized by its velocity, $V_{P,LF}$, and mass M_p, while the target nucleus is characterized by its velocity, $VT_{,LF}$ which equals zero, and has mass M_T. The total kinetic energy in the laboratory frame (T_{LF}) is thus,

$$T_{LF} = \tfrac{1}{2} M_P V_{P,LF}^2 + \tfrac{1}{2} M_T V_{T,LF}^2 \qquad \textbf{(1.1a)}$$

$$T_{LF} = \tfrac{1}{2} M_P V_{P,LF}^2 + 0 = \tfrac{1}{2} M_P V_{P,LF}^2 \qquad \textbf{(1.1b)}$$

Similarly the momentum ($P_{P,LF}$) for the projectile is given as $P_{P,LF} = M_P V_P$, and the momentum of the target nucleus is $P_{T,LF} = M_T V_T$ which is equal to zero in this reference frame. The center-of-mass frame is strictly defined as the system in which the total momentum of the projectile and target nucleus is zero. The velocity of the projectile and target nucleus in the center-of-mass frame are Galilean transformations from the laboratory frame, and are given respectively as thus,

$$V_{P,CM} = V_{P,LF} - V_{CM} \qquad \textbf{(1.2a)}$$

$$V_{T,CM} = V_{T,LF} + V_{CM} = V_{CM} \qquad \textbf{(1.2b)}$$

The momentum in the center-of-mass frame (P_{CM}) is equal to $M_P(V_{P,LF} - V_{CM})$; since as the momentum of the projectile is also equal to the target's momentum, it is also equal to $M_T V_{CM}$. By solving for the velocity of the center of mass (V_{CM}), a transformation from one reference frame

to the other is possible. This can be accomplished by setting both values of the momentum in the center-of-mass frame equal to each other as thus,

$$M_P V_{P,LF} - M_P V_{CM} = M_T V_{CM} \tag{1.3a}$$

$$M_P V_{P,LF} = (M_T + M_P) V_{CM} \tag{1.3b}$$

$$V_{CM} = \frac{M_P V_{P,LF}}{(M_T + M_P)} \tag{1.3c}$$

Using this result it is trivial to then solve for the relationships between the center-of-mass momentum and laboratory frame momentum, and between the total kinetic energy in the center-of-mass and the laboratory frame.

$$P_{CM} = \frac{M_T P_{P,LF}}{(M_T + M_P)} \tag{1.4a}$$

$$T_{CM} = \frac{M_T T_{P,LF}}{(M_T + M_P)} \tag{1.4b}$$

On occasion a beam particle A will strike a nucleus of B. The rate, R, of this occurrence is given by Equation 1.5.

$$R = \sigma \Phi N_T \tag{1.5}$$

Where σ is the cross section, Φ is the flux density of beam particles A in units of $cm^{-2}s^{-1}$, and N_T is the number of atoms of the target nuclide T. The cross section for simple scattering experiments can be described using the geometric cross section of the nucleus, which is on the order of 10^{-24} cm^2 (ignoring nuclear forces and Coulomb interactions). Aside from scattering experiments, cross sections can be defined for specific nuclear reactions, and can be thought of

as a probability or likelihood that a nuclear reaction will occur. The unit for cross sections are called barns: 1 barn (b) = 10^{-24} cm^2.

A sample rate calculation for the production of rutherfordium isotopes is as follows. The production cross-sections for rutherfordium are on the order of nanobarns 10^{-33} cm^2, typical beam intensities available are ~10^{11} particles $cm^{-2}s^{-1}$, and targets have ~10^{20} atoms within a typical area under irradiation. Multiplying these magnitudes together yields the rate of producing rutherfordium nuclei: somewhere on the order of ~0.01 s^{-1} or 36 nuclei per hour.

To conduct a rutherfordium chemistry experiment, it is necessary to produce targets that do not flake, can withstand large intensities of heavy-ion beams, and are uniform and homogenous. Recently, there has been a push for higher intensity stable and radioactive beams at Coulomb barrier energies. In order to advance studies of nuclear structure, reactions, and astrophysics, experiments with new isotopes produced at the Facility for Rare Isotope Beams (FRIB) and the Argonne Tandem Linac Accelerator System (ATLAS) require higher energy and higher intensity beams, as outlined in the Nuclear Science Advisory Committee (NSAC) Long Range Plan for 2007.[3] In the time leading up to the opening of the FRIB at Michigan State University, ATLAS at Argonne National Laboratory is being refit to provide a ten-fold intensity increase for all species.[4] The upgrade will allow studies of nuclei from the very neutron-rich to those beyond the proton dripline. Other accelerators, such as the UNILAC, a linear accelerator at the Gesellschaft für Schwerionenforschung Institute (GSI) in Germany, also desire high beam intensities for super-heavy element research.[5]

Furthermore, it is necessary to construct instruments that will assist with the exploration of the chemistry from the collected rutherfordium atoms, and to develop electronic, circuits, and data acquisition systems to record their decay characteristics. Finally, new nuclear reactions should be investigated in order to look for alternative, more attractive, methods for producing

long-lived rutherfordium isotopes (^{261}Rf).

1.2 Target Methodology

The preparation of targets has long occupied the time of experimental nuclear scientists. To date the main methods of creating targets are vacuum evaporation, molecular plating/electrodeposition, and painting/sedimentation. The work in this thesis will describe a new target methodology: polymer-assisted deposition. Each method is briefly described below.

One of the earliest reported uses of vacuum evaporation as a method to produce nuclear targets was by Randall and Smith in 1955.[6] The method involves pasting an oxide or carbonate of an alkali or alkaline earth metal of interest onto a tantalum foil; the foil is then heated, and as the metal is reduced, the elemental form is collected on a cool assembly within the chamber. A schematic of this process is shown in Figure 1.3.

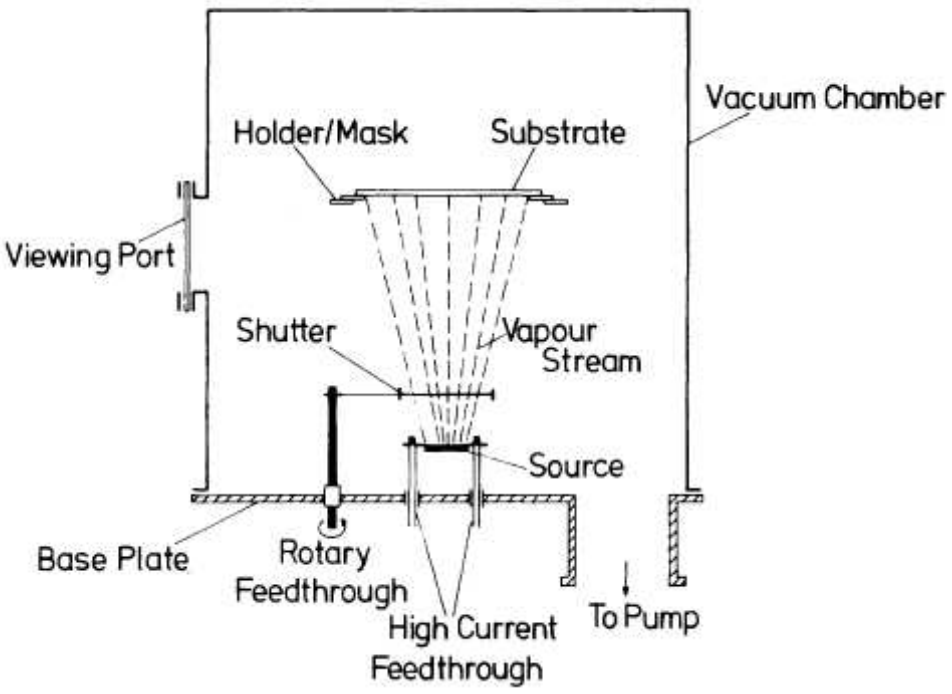

Figure 1.3 A generalized schematic of the evaporation method for target productions.[7]

The target thickness is a function of the rate of heating and the geometry of the vacuum evaporation apparatus. The chemical equations for this process with alkaline earth and alkali metals are given by Equation 1.6a-b and 1.7a-b respectively.

Alkaline Earth Metals

$$M_2O_{(s)} + \text{heat} \rightarrow 2M_{(s)} + \tfrac{1}{2}O_{2(g)} \quad \textbf{(1.6a)}$$

$$M_2(CO_3)_{(s)} + \text{heat} \rightarrow 2M_{(s)} + CO_{2(s)} + \tfrac{1}{2}O_{2(g)} \quad \textbf{(1.6b)}$$

Alkali Metals

$$MO_{(s)} + \text{heat} \rightarrow M_{(s)} + \tfrac{1}{2}O_{2(g)} \quad \textbf{(1.7a)}$$

$$M(CO_3)_{(s)} + \text{heat} \rightarrow M_{(s)} + CO_{2(s)} + \tfrac{1}{2}O_{2(g)} \quad \textbf{(1.7b)}$$

Molecular plating and electrodeposition are the electrochemist's solution to target preparation. Both methods employ an electrochemical cell with a solution comprised of the dissolved target metal of interest. Once a potential is applied across the cell, the solute deposits onto the target backing. Because both methods have a similar setup they are erroneously used interchangeably in the literature. In molecular plating, the molecular compound that is dissolved in the electrolyte is deposited onto the surface, while in electrodeposition it is the elemental form of the element. Another difference is the use of high voltages (50-2000 V) in molecular plating.[8]

A general procedure for molecular deposition is as follows: A solution is prepared of either hydrochloric or nitric acid (~3N) at a concentration of 10 mg/mL of the solute to be deposited. Anywhere between 25 to 250 mL is added to 10 mL of iso-propyl alcohol. This is then transferred to an an electrodeposition cell that has the target backing (e.g., thin aluminum) as the

cathode and a platinum wire as the anode. The voltage is then applied, and the compound deposits onto the target backing.

Painting produces metal oxide targets by dissolving a metal in a solution of alcohol and lacquer. The solution is applied drop-wise to a target backing which is slowly rotated. The process is complete after the foil is removed from the rotating mount and heated in oxygen, forming a metal oxide film on top of the backing material. This method was first used to produce uranium oxide films on platinum foils by Jorgensen in 1944.[9] The method has been successfully extended to *d*-transition metals,[10] and was the standard method of actinide target production at Lawrence Livermore National Laboratory from 1969-1978.[11]

This method has several limitations and has been described as the last option available when "…depositing refractory compounds and for all other sources when all else fails."[12] Painting will lead to a non-uniform metal oxide layer; the target will be thinner in the middle and thicker at the edges. It can not be used to make thick targets (>500 μg/cm^2), because the metal oxide will crack and flake off the substrate's surface.

The Polymer-Assisted Deposition (PAD) method was first described by Jia *et al.* in 2004.[13] The PAD method shares many similarities with the painting method of target production but was invented independently. In the PAD method, a metal chloride is dissolved in an aqueous solution of a multi-dentate polymer. The coordinating nature of the polymer helps dissolve a greater amount of the metal chloride than would be possible in a generic lacquer, see Figure 1.4. The solution is spin coated onto an atomically smooth surface (root mean surface height deviation ~1 nm). The resulting metal-organic film is heated in an oven to temperatures greater than 600 °C in air. The organic part of the metal-organic film depolymerizes and is removed by evaporation. The metal that is left on the surface is converted to a metal oxide by this process. The PAD method produces metal oxide films several orders of magnitude more homogenous

than comparable methods discussed above.

The metal-organic solutions are the key to why the method works well. The polymer used for this work was polyethylenimine (PEI). The organic molecule chelates to the dissolved metal, this forms the metal-organic covalent complex, presumably through the interaction of the lone-pair of electrons on nitrogen to the metal cations, see Figure 1.4. An advantage of the method is that the solutions are all aqueous and thus do not involve volatile or hazardous organic solvents.

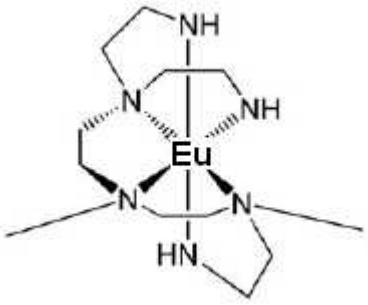

Figure 1.4 Putative structures of PEI complexed to a metal cation.

The viscosity of the solution can be adjusted by the addition or removal of water. The resulting metal-organic solutions are then spin-coated onto the substrate or target backing of interest.

The next step of the method is the thermal decomposition of the polymer after spin coating. During the thermal decomposition process the organic constituents of the film decompose and the surface is left with only a metal surface that quickly forms an oxide as it is heated in an oven in air. The surfaces created with the PAD method have been shown to be highly homogeneous, and studies of the film thickness as a function of the calcination temperature have shown that at ~500 °C only the metal oxide layer is left on the surface while any carbon, nitrogen, or hydrogen atoms are no longer present; which has only been seen for d- and f-transition metals.[7]

1.3 Techniques and Instruments of Transactinide Gas-Phase Chemistry

The transactinide elements have an atomic number (Z) greater than 103, and are comprised mostly of the fourth row d-transition metals. As heavy transition metals, they are predicted to have low volatility and high atomic melting points, much greater than 1000 °C. However, there are some inorganic compounds (*i.e.*, chlorides and bromides) that should have appreciable volatility and lower melting points, allowing for the study of their chemistry in the gas-phase.

The generalized method for studying such complexes in the gas-phase is as follows: 1) After irradiation of the target material with the appropriately selected heavy-ion the element is separated from the beam by the use of a velocity separator or a magnetic rigidity separator like the Berkeley Gas-filled Separator (BGS, chapter 5); the atoms are then stopped in the gas-phase in a chamber filled with a gas at a pressure that ensures all the atoms will reach the end of their range within it, this chamber is called a Recoil Transport Chamber (RTC) and is typically separated from the separator by a thin Mylar window. 2) In the RTC a complexing chemical agent is introduced into the gas in order to cause a chemical reaction with the atoms that have been stopped; from the RTC the new chemical compounds are studied by various chromatography methods, discussed in more detail later. 3) After chromatography of the new compound it is necessary to cluster it on aerosol particles and place it in front of detectors in order to measure the yield from the decay of the radionuclide. There are two types of gas-phase chromatography performed with transactinides, thermochromatography and isothermal chromatography. Thermochromatography has a variable temperature profile for the column while isothermal chromatography has a constant temperature, Figure 1.5. Thermochromatography is ideal for compounds that have their volatility within the narrow range

of temperatures that permit semiconductor detectors to operate, that is because the method often uses semiconductor detectors as the column material for chromatography. Isothermal chromatography is ideal for chemical compounds that are volatile at temperatures above the operating temperatures of semiconductor detectors, but a drawback is they often require more data to obtain physical meaning because data needs to be taken over several temperatures and only one temperature can be explored per experiment; isothermal chromatography will be the focus of all further discussions on transactinide chemistry in this thesis.

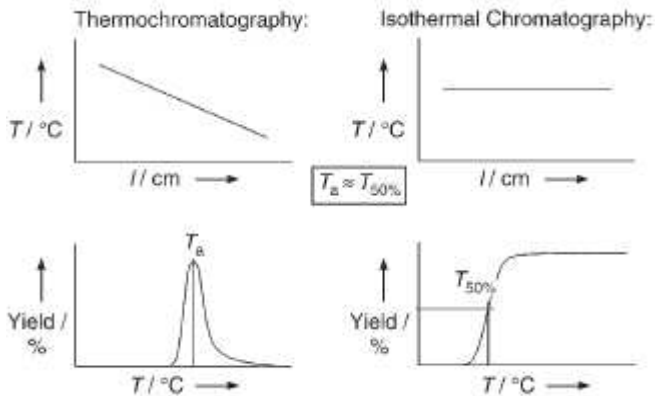

Figure 1.5 Left: Basic principles of thermochromatography. Right: Basic principles of isothermal gas chromatography. From Türler *et al.*[14]

The first experiments studying gas-phase transactinides were performed by Zvara and coworkers at Dubna.[15,16,17] In these experiments, the atoms of interest were not separated from the beam or fission fragments and the thermochromatographic column was connected directly to the the RTC. Now it is more common to couple a gas-jet transport system from the RTC to the chromatographic system because this acts as a chemical method to separate non-volatile compounds from the volatile transactinides of interest.[18]

The goal of gas-phase chemists has been to study these volatile chemical compounds in the gas-phase and in order to do this the On-Line Gas Chromatography Apparatus (OLGA II)

was made at GSI and serves as an illustrative example of how these experiments are performed. OLGA II was designed to "...continuously separate volatile products and to collect them by deposition in a detection system."[19] The OLGA II system starts as follows: 1) Recoiling nuclei are embedded onto potassium chloride aerosol particles in the gas-phase. 2) The aerosols are stopped in a quartz wool plug. 3) HBr or other strong halogenating gas is passed over the glass quartz wool plug which is heated to 900 °C. 4) The volatilized metal complex was then passed into a variable oven for chromatography studies. 5) After chromatography the products are embedded onto aerosols again and passed to a tape detector counting system. Figure 1.6

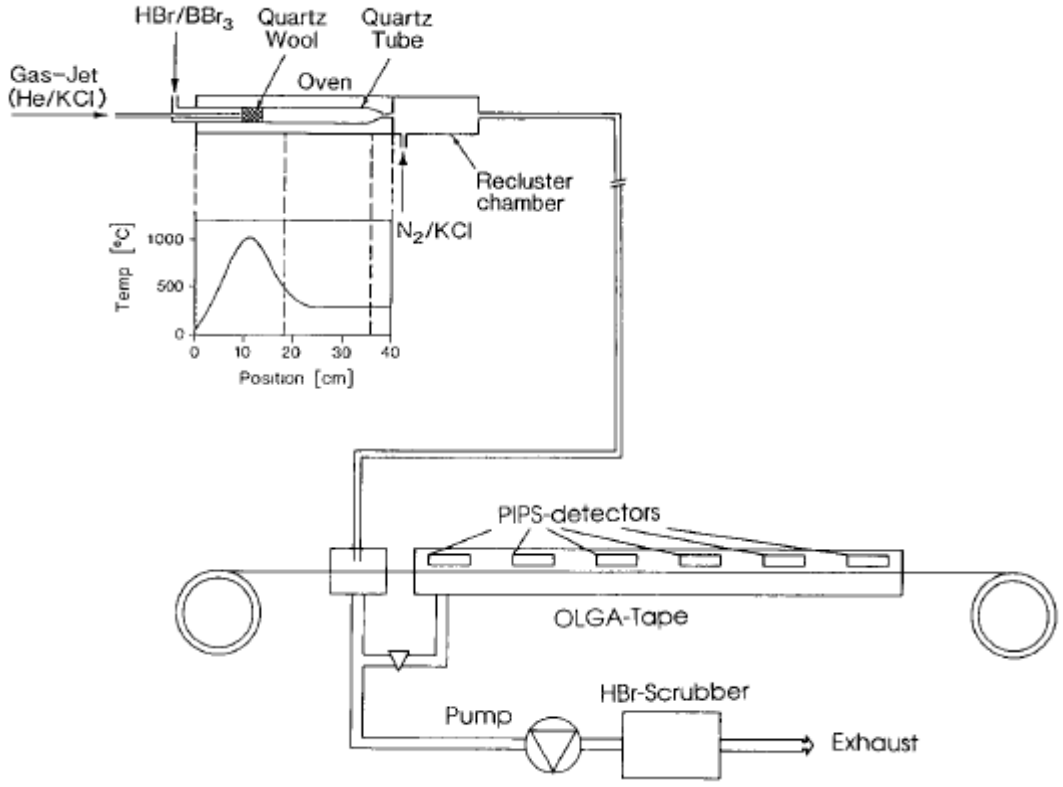

Figure 1.6 Schematic of OLGA II. (Gäggeler 1991)

1.4 Rutherfordium Gas-Phase Chemistry

Relativistic effects starts to play a dominant role in the chemistry of elements as early into the periodic table as mercury and gold, and can completely dominate the chemistry of transactinides-- the reason for their influence is described herein. Since the atomic number (Z) for each consecutive element increases, as one goes to heavier elements the core electrons within these elements move faster due to the presence of a larger positive nuclear charge. This has the effect of increasing the mass of the inner electrons and this mass increase is expressed in the following equation,

$$m = m_0\sqrt{(1 - \frac{v}{c}^2)} \tag{1.8}$$

Here m_0 is the rest mass of the electron, v is the speed of that electron, c is the speed of light and m is the new value of the mass. Relativistic effects also change the average distance an electron is from the nucleus, this is given by the effective Bohr radius equation,

$$a_B = a_B^0\sqrt{(1 - \frac{v}{c}^2)} \tag{1.9}$$

Here a^0_B is the normal Bohr radius of an electron in an atom, and a_B is the effective Bohr radius. The primary effect of relativistic effects is to bring the s and $p_{1/2}$-electrons closer to the nucleus, and thus lower their energy. This has the secondary effect of screening the nuclear charge from the other electrons, namely those in $p_{3/2}$, d, and f-orbitals, and thus raising their energy.

Originally it was thought that rutherfordium would behave like a p-block metal. Because

relativistic effects were known to lower the energy for $p_{1/2}$ orbitals and raise the energy of d-orbitals, the predicted electronic structure of rutherfordium was calculated to be dominantly described by $[Rn]5f^{14}7s^27p^2$.[20,21,22,23] After unsuccessful attempts were made to look for volatile atomic rutherfordium behavior,[24] a typical lead-like property, and after more refined calculations[25] showed that rutherfordium's electron configuration should be $[Rn]5f^{14}6d^27s^2$, it is no longer thought that rutherfordium behaves like a p-block metal.[26]

In the 1990s rutherfordium was reacted with chlorinating[27] and brominating[28] agents to make rutherfordium (IV) chloride and rutherfordium (IV) bromide, respectively. Rutherfordium (IV) chloride was of special interest to transactinide chemists because it was not known if it would be more or less volatile than its homologue, hafnium (IV) chloride. Hafnium (IV) chloride is less volatile than zirconium IV chloride and from extrapolating this trend, it was predicted that rutherfordium (IV) chloride would be the least volatile in this series.[29] Also non-relativistic calculations made a similar prediction.[30] However, relativistic calculations predicted that rutherfordium (IV) chloride would be more volatile than hafnium (IV) chloride, see Figure 1.7.[31,32]

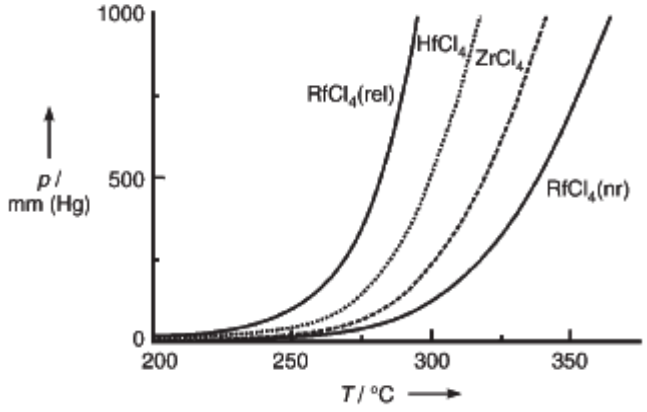

Figure 1.7 Volatility of zirconium (IV) chloride, hafnium (IV) chloride, non-relativistic

rutherfordium (IV) chloride (nr), and relativistic rutherfordium (IV) chloride (rel).[33]

The experiments with rutherfordium (IV) chloride established that rutherfordium (IV) chloride is more volatile than hafnium (IV) chloride, confirming the role relativistic effects play in transactinide chemistry.[34,35,36] The homolog bromide series for group 4 metals showed that rutherfordium (IV) bromide has a similar trend in volatility.[37,38] From those data it is possible to perform a Monte Carlo fit to determine the adsorption enthalpies on quartz (the material of the chromatographic coloumn) of both rutherfordium (IV) bromide and rutherfordium (IV) chloride, see Figure 1.8 and ref[39] for more details on how these calculations are performed.

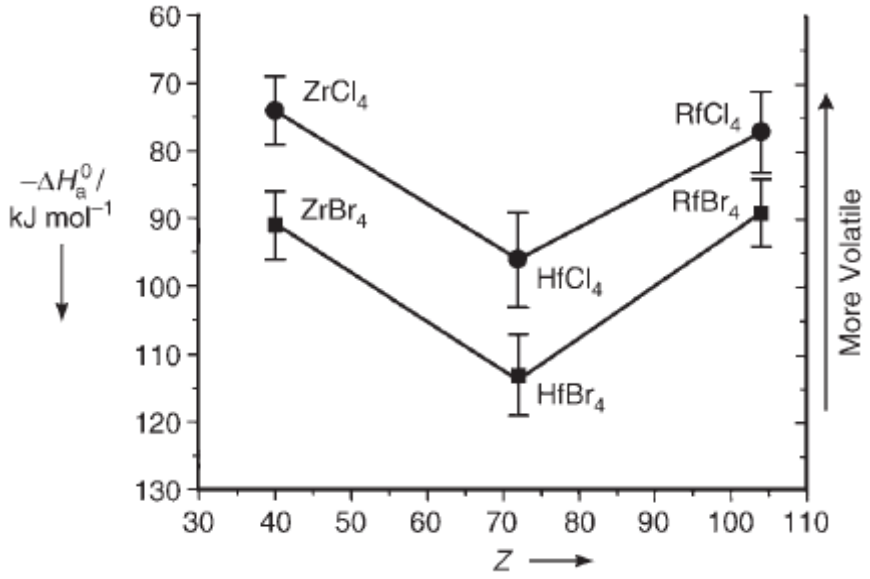

Figure 1.8 Adsorption enthalpies for group IV metal chlorides and bromides.[40]

Another physiochemical property, the heat of sublimation, can be estimated. A linear correlation exists between the heat of adsorption on quartz for a volatile compound and its heat of sublimation. Thus by studying this behavior for a large set of compounds, a reliable estimate of

the heat of sublimation can be made.[41],[42]

A problem that often arises when studying rutherfordium chemistry is its tendency to form oxy halide compounds ($RfOCl_2$). As oxy halide metal compounds are less volatile than pure halide metal compounds,[43] much care has to be taken to assure that oxygen does not contaminate these experiments. When $RfOCl_2$ and HfOCl2 were studied, it was found that they have comparable behavior, thus the effects of relativistic effects would have been less obvious.[44]

1.5.1 Production of Long-Lived Rutherfordium: ^{261}Rf

There are several isotopes of rutherfordium that have a half-life that is long enough to perform chemistry: *i.e.*, ^{259}Rf ($t_{1/2}$ = 2.5 s), ^{257}Rf ($t_{1/2}$ = 4 s), ^{255}Rf ($t_{1/2}$ = 1.5 s). However, working with isotopes that decay so quickly is not easy; working with isotopes that have half-lives of minutes is preferable. Rutherfordium-261 with a half-life of 75 s is the perfect candidate to explore the chemistry of rutherfordium.

In 1970 Albert Ghiorso and coworkers were the first to synthesize rutherfordium-261.[45] The nuclear reaction they studied was ^{248}Cm(^{18}O, 5*n*)^{261}Rf. They observed a broad alpha peak from 8.2-8.3 MeV. The broadness in the peak is presumably from the 8.22 MeV alpha-particle from ^{257}No and an 8.28 MeV alpha-particle from ^{261}Rf. Their measured cross section is 5 nb for a laboratory bombarding energy of 97 MeV. In the same year, Robert Silva and colleagues in collaboration with Ghiorso's team eluted ^{261}Rf from a cation exchange resin in the presence of tetravalent zirconium and hafnium, and trivalent actinide tracers.[46] At the time it was not known whether rutherfordium would behave like a group 4 metal or would follow the actinide series in its chemistry. The results of this experiment showed that 261Rf eluted with the tetravalent hafnium and zirconium tracers and solidified its place as a group 4 metal.

In 1995 ,Kadkhodyan *et al.* also synthesized ^{261}Rf via ^{248}Cm(^{18}O,5n)^{261}Rf reaction.[47] The

authors measured a half-life of 79 s. In 2000 Sylwester *et al.* using the same nuclear reaction were able to refine this half-life to 75 s by obtaining better statistics.[48] Other authors that have investigated ^{261}Rf production are Lazarev *et al*. in 2000 from the nuclear reaction ^{244}Pu(^{25}Ne, 5n)^{261}Rf.[49] Finally the work presented in this thesis discusses the production of ^{261}Rf using the nuclear reaction ^{238}U(^{26}Mg, 3n)^{261}Rf.

Chapter 2: Polymer-Assisted Deposition

2.1 Introduction

The preparation of homogenous metal oxide films (100 to750 nm) is of interest to nuclear science for use as targets in nuclear reactions. Metal oxide targets, prepared for nuclear science applications, are conventionally made by molecular plating.[50,51] However, this method suffers from poor adhesion to the backing material and lacks homogeneity at target thicknesses less than about 300 nm.[52] Jia *et al.* recently reported an alternative method, polymer-assisted deposition(PAD),[53] for producing crack-free homogenous metal oxide films with uniform thicknesses between 20 and 400 nm.[54,55] In the PAD method, a water-soluble multidentate polymer binds to metal precursors resulting in a homogenous distribution of the metal in solution. The solution is spin coated and then annealed to yield a high-quality metal oxide film. In this thesis, metal oxide films prepared by PAD were created as an alternative method of target production.

Targets composed of actinide oxides are necessary to synthesize the relatively long-lived and neutron-rich isotopes of transactinides (Z>103). The PAD method was used to study the oxide films of europium (Eu) and thulium (Tm) as models for actinides with an oxidation state of +3 (*e.g.*, americium and curium). Hafnium (Hf) was used as a model of +4 actinides(*e.g.*, uranium and plutonium). The metal oxide film thickness was determined as a function of the weight percent of the metal in solution. The reapplication of the PAD technique on an existing metal oxide layer to build thicker high-quality films was also investigated.

2.2.1 Solution Preparation

All solutions were composed of 15% polyethylenimine (PEI) by weight, with varying

weight percentages of metal chlorides. PEI (10 kDa, Aldrich) was dissolved in water and adjusted to a pH between 6 and 6.5 using 37% HCl and a pH meter with attached combination pH electrode (Model 231, Orion Research). Solutions were mixed using a vortex mixer, followed by stirring on a magnetic stir plate. Depending on the desired thickness of an appropriate weight percent of europium(III) chloride hexahydrate (99.99%, Aldrich), thulium(III) chloride hexahydrate (99.999+%, MV Laboratories), or hafnium(IV) chloride hexahydrate (98%, Aldrich) was added to the dissolved polymer. All solutions were prepared in a non-clean room environment and were stirred for at least 1 h before use.

2.2.2 Caution

Hafnium(IV) chloride (CAS# 013499053) is highly volatile and a strong irritant. Handling should be limited to ventilated environments.

2.2.3 **Single Layered Metal-Organic Films**

Silicon [100] wafers (WaferNet) were cut into rectangles (2.00 cm by 0.85 cm). These cut wafers were placed individually into a spin-coater, and 100 µL of the appropriate solution was evenly distributed onto the surface by pipette. The wafers were accelerated for 11 s to a maximum angular velocity of 1500 rpm and spun for 3.0 min in air to form a layer of metal-organic polymer. The angular acceleration was the same between trials; an image of the spin-coater is shown in Figure 2.1. The surface of each sample was scratched three times down to the silicon with sharp tweezers to determine film height. Samples were placed in a muffle oven. The temperature was increased by 50 °C every 15 min from 50 °C to 900 °C. After 15 min at 900 °C, the oven was turned off and allowed to cool to room temperature for several hours.

Figure 2.1 An image of the spin-coater device is shown. Samples were mounted onto a pc fan that served as the spin-coater.

2.2.4 Multiple layered metal-organic films (Reapplication)

Three circular silicon [100] wafers (WaferNet) with diameters of 10 cm were used to test the viability of forming thicker hafnium(IV) oxide films by reapplication of the PAD method. One aliquot of 3.0 mL from a solution composed of 10% Hf and 15% PEI was evenly distributed onto each wafer surface. The wafers were accelerated to a maximum angular velocity of 2500 rpm and spun for 3.0 min in air to form a layer of metal-organic polymer. The metal-organic coated wafers were annealed in the same manner as described above, and the entire

process was repeated six times.

2.2.5 Experimental Method – Film Thickness and Visual Assessment

The height of a single layered metal oxide film was determined by using a profilometer (Dektak 150, Veeco) to scan perpendicular to a scratch. Average film thicknesses were measured from three samples spun identically with the same solution. Each sample had three scratches and each scratch was scanned three times at different locations. A plain silicon wafer was scratched similarly to the samples described above, and the tweezers left no measurable indentation into the silicon surface. Outliers were Q-tested out of the data set at 90% confidence. The thickness of the multiple layered metal oxide films were determined by weight, because scratching and scanning by profilometry would have introduced an inhomogeneous surface for the spin coating of consecutive layers. A cartoon of how profilometry works is shown in Figure 2.2. A profilometer works similarly to an older 19th century technology known as a phonograph. A stylus, in this case diamond, is moved in contact with the surface of the sample at a specified force. The profilometer can measure vertical displacements from 10 nm to 1 mm. Thus the profilometer can quantitate the roughness of a surface, but in these experiments it was used to measure the height of the metal oxide film.A sample plot from the profilometer scanning over a scratch is shown in Figure 2.3; the film thickness for a sample is determined by subtracting the flat region of the plot (corresponding to the top of the film) from the bottom flat region (corresponding to the surface of the silicon substrate/backing); the large sharp peak on the right hand region is of a surface anonymity at that location of the film scanned, possibly a comet.

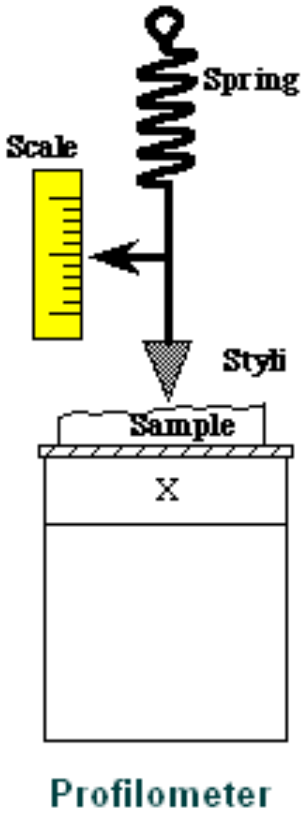

Figure 2.2 Shown is a cartoon of how profilometry works. Samples are places onto a sample stage and a stylus is passed over the sample. This allows us to measure the film height of a sample.

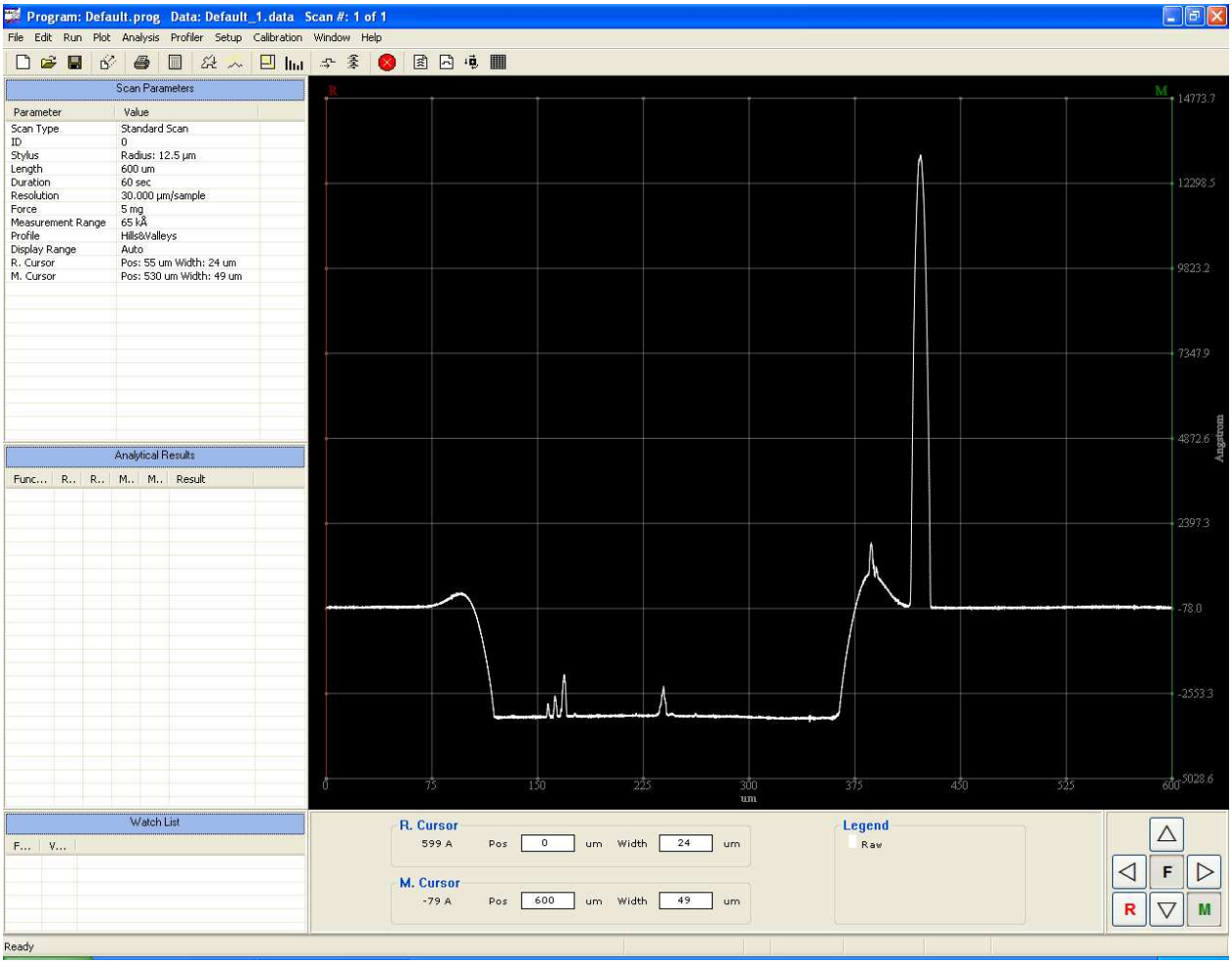

Figure 2.3 A screen shot of the program that was used with the profilometer. The black region plots the height of the film perpendicular to a scratch, the y-axis is in units of Angstroms and the x-axis is in micrometers. The left hand side of the image displays some parameter values that the instrument was using.

Mounted on the profilometer is an optical camera that can be used to take pictures of the surface of the samples. A picture of one of the samples showing a scratch placed onto the surface in order to determine film height is shown in Figure 2.4

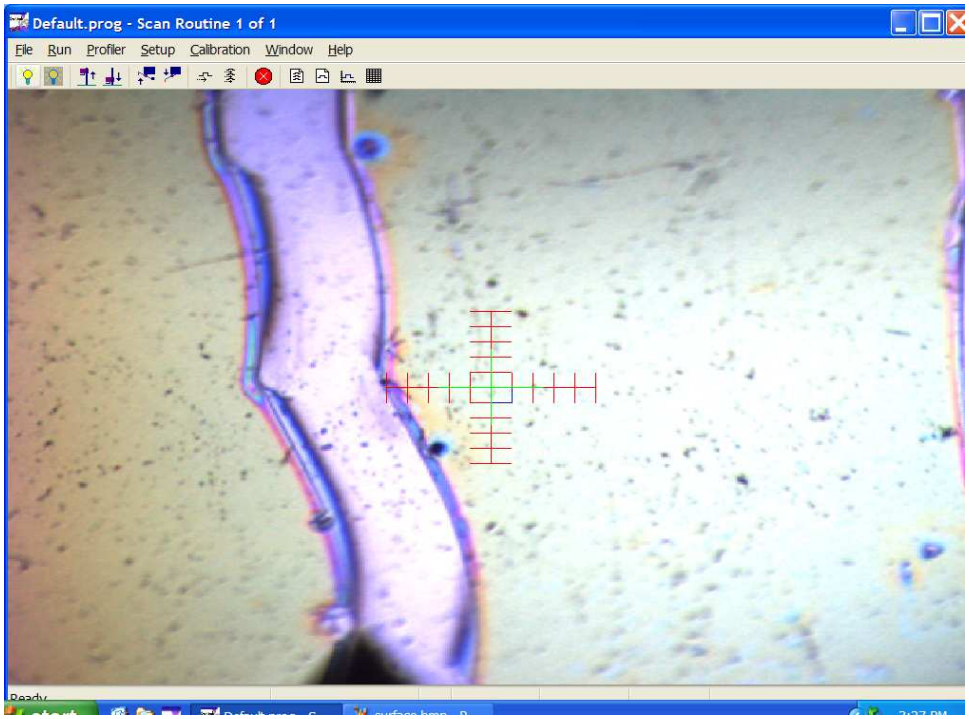

Figure 2.4 An optical image of one of the samples taken with a camera mounted onto the profilometer. The image shows the surface with a scratch that we placed onto the sample.

A scanning electron microscope (SEM) (Ultra 55VP Fesem, Zeiss) was used to determine surface homogeneity on the 200 nm to 200 μm scale by imaging the surface and the cross-section. An atomic force microscope (AFM) (MFP 3D, Asylum Research) was used to obtain a high resolution image of a 1 μm × 1 μm representative section of the surface. The AFM used cantilevers with a spring constant of 3 N/m (Multi75, Budget Sensors) for imaging in the attractive regime in AC mode. The crystal structure of the film created from the reapplication of PAD was determined with an X-ray Diffraction machine (Diffraktometer D500/501, Siemens). The wafer was scanned using 2θ values of 20 to 66° in 0.05 degree increments at one second per step.

2.3.1 Single Layered Metal-Organic Films

The thickness of the metal oxide film produced by the polymer-assisted deposition method is a function of several variables: angular acceleration of the spin-coater, viscosity of the solution, metal ion concentration, maximum velocity of the spin-coater, total time spun, and the annealing temperature profile.[56] To determine the optimum conditions for the PAD method, a systematic study was performed studying the effect that varying metal ion concentration has on film thickness for Eu(III), Tm(III), and Hf(IV).

The a-priori expectation of a linear relationship between film thickness and metal ion concentration was not observed, as seen in Figures 2.5-2.7. Spin coating yields film heights having an $\eta^{1/3}$ dependence,[57] where η is the viscosity. It is hypothesized that the deviation from linearity is due to the increase in viscosity from increasing metal ion weight percent in solution while fixing the polymer weight percent at 15%, thus decreasing the total amount of water available for solvation. Attempts at preparing solutions above 8% Eu, 10% Tm, and 12.9% Hf, all in 15% PEI, were unsuccessful due to a precipitation that could not be re-dissolved. For hafnium, the highest quality film was produced with the 10% by weight (b.w.) Hf solution. Likewise, 6% (b.w.) Eu and 8% (b.w.) Tm yielded the best results for those elements. The standard deviation for each sample was found to be less than 13% of its film height except for those derived from the 8% (b.w.) Eu solution, and the 2% (b.w.) metal ion solutions. The 8%-Eu film appeared inhomogeneous by SEM analysis, and consisted of micro-scale amorphous structures of Eu_2O_3. The 2%-metal ion solutions yielded films with a standard deviation of about 10 nm, which are similar to those produced by the more concentrated metal ion solutions.

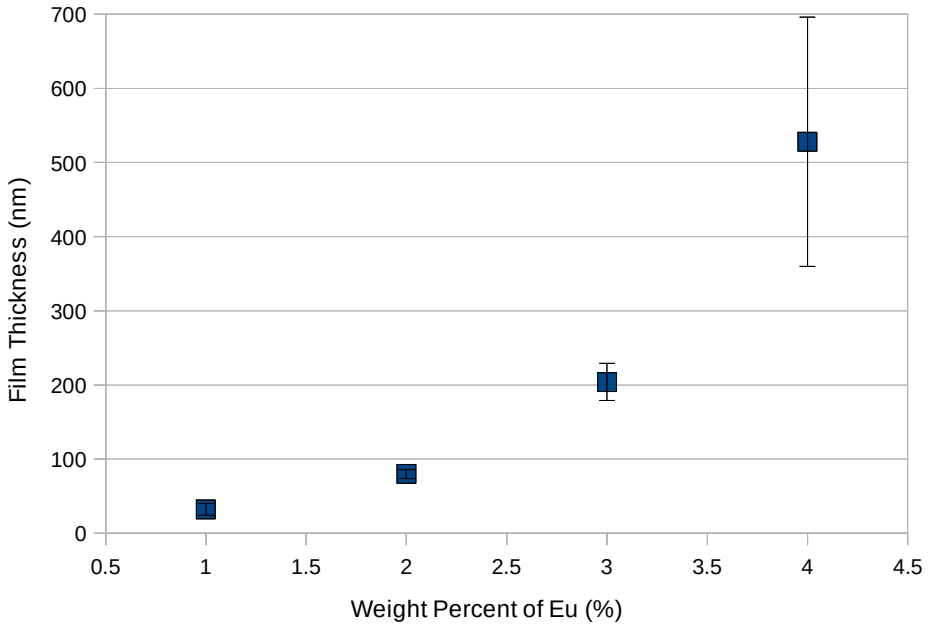

Figure 2.5 Film thickness as a function of metal ion concentration of europium(III) oxide. Error bars correspond to one standard deviation. Error for all solution preparations are negligible (< 0.05%).

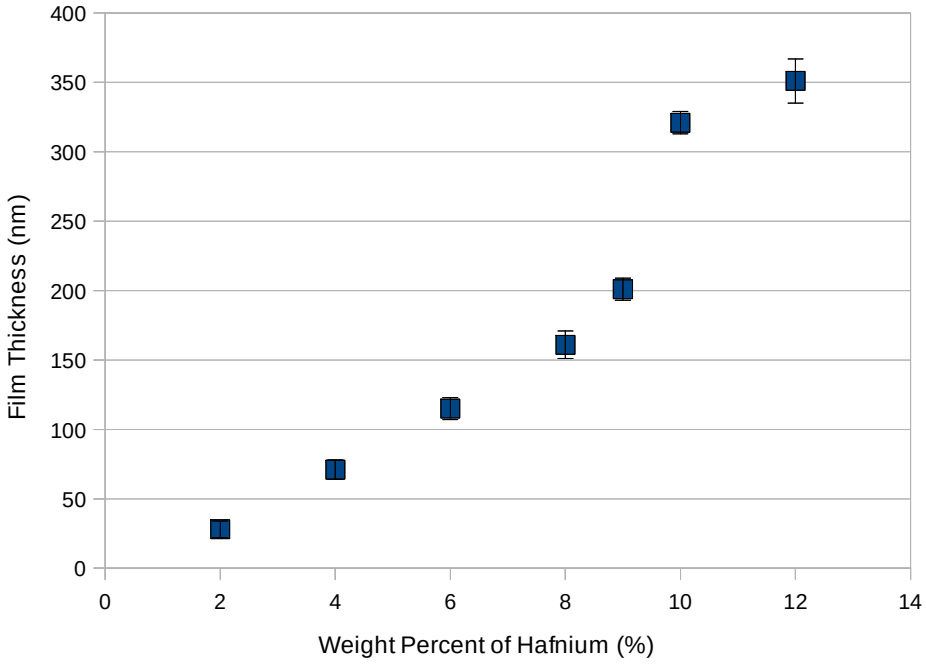

Figure 2.6 Film thickness as a function of metal ion concentration of hafnium(IV) oxide. Error bars correspond to one standard deviation. Error for all solution preparations are negligible (< 0.05%).

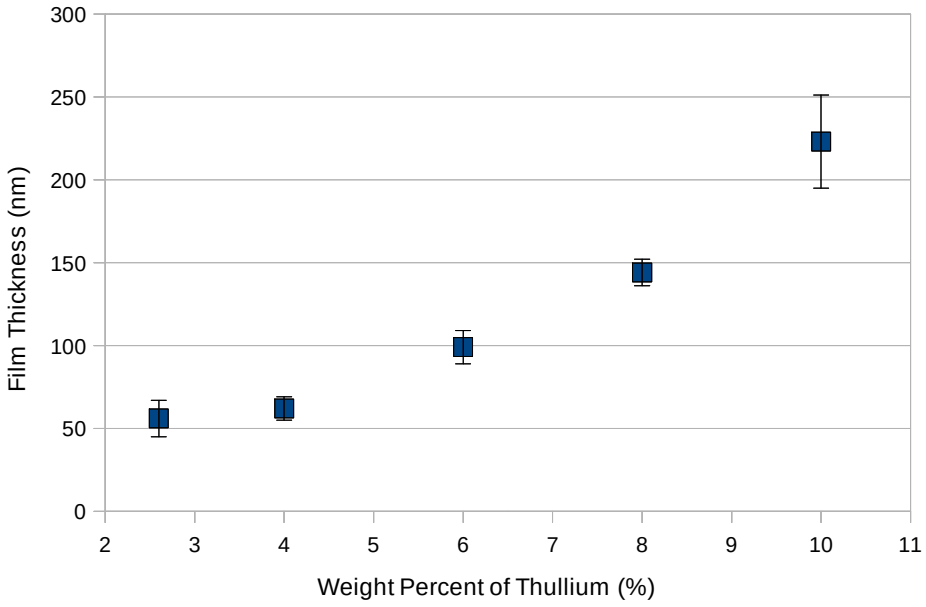

Figure 2.7 Film thickness as a function of metal ion concentration of thulium(III) oxide. Error bars correspond to one standard deviation.

Surface SEM and cross-section SEM images of europium, thulium, and hafnium are shown in Figure 2.8a-c. The surface of 6% Eu and 8% Tm were homogenous with an even distribution of metal oxide grains. However, there was a random distribution of surface aggregates of 2–5 μm in length in the 8%-Tm films, likely caused by being near the saturation point of Tm/PEI in water. The 10% Hf had the most homogenous surface of all the metals, although some minor surface aggregates are apparent as shown in Figure 2.8c. The cross sections for all metals, Figure 2.8a–c, show highly uniform films. AFM images of the metal oxide films show that the surfaces are composed of grains.

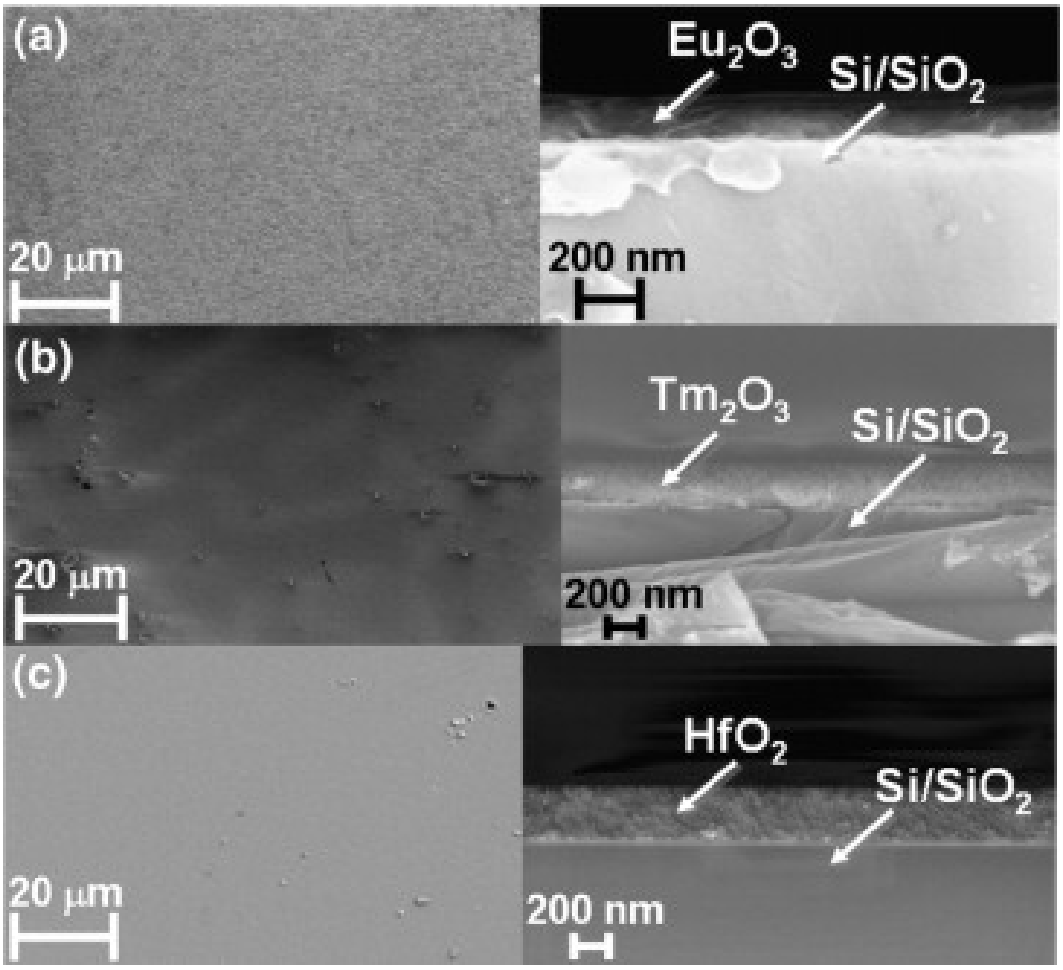

Figure 2.8 (a) Europium(III) oxide film (produced from a 6% by weight europium solution); (b) thulium(III) oxide film (produced from a 8% by weight thulium solution); (c) hafnium(IV) oxide film (produced from a 10% by weight hafnium solution.).

The largest grains occur in the Eu_2O_3 films (30–40 nm in diameter) and the smallest in the HfO_2 films (10–15 nm in diameter). The AFM images confirm the surface homogeneity and uniformity of all the films as shown in Figure 2.9. AFM and profilometry data verify that hafnium forms the most highly uniform and homogenous metal oxide film when SiO_2/Si is used as the backing material. The roughness, as measured by AFM, has a root mean square, rms, of

7.46 nm, 1.87 nm, and 1.45 nm for europium, thulium, and hafnium films, respectively, over 1 μm^2.

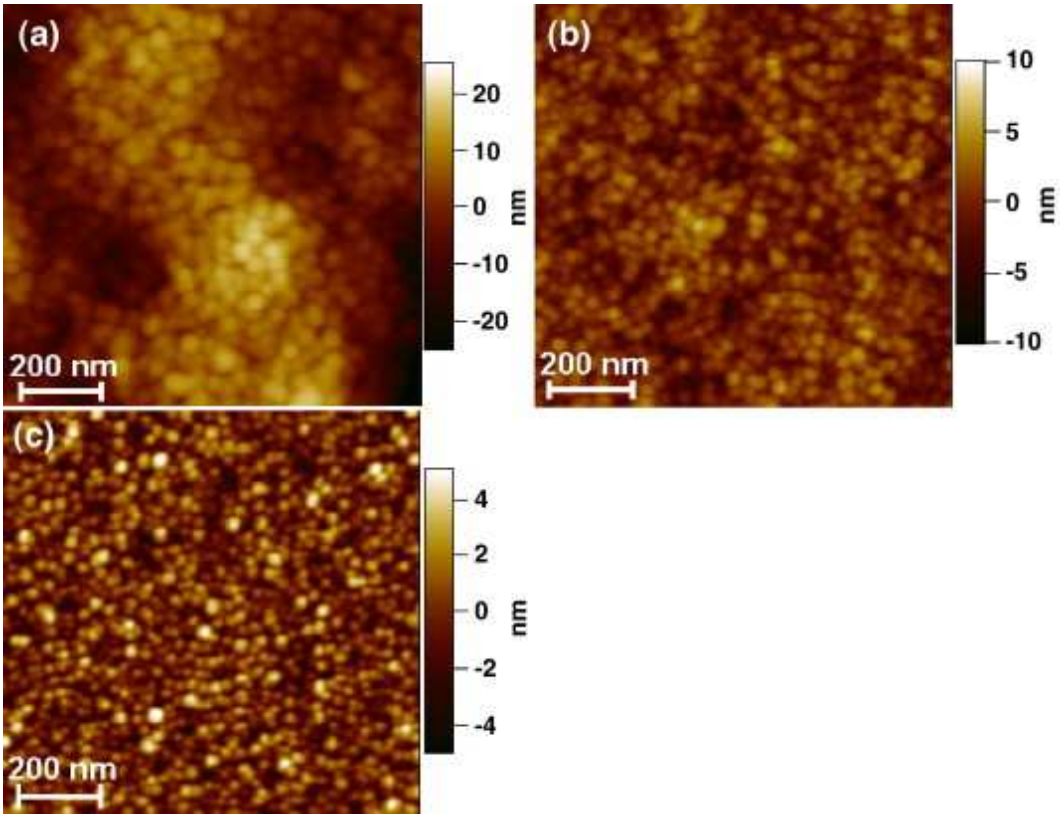

Figure 2.9 AFM images of metal oxide films after single application of the PAD method: (a) europium(III) oxide film produced from a 6% by weight europium solution, (b) thulium(III) oxide film produced by an 8% by weight thulium solution, and (c) hafnium(IV) oxide film produced from a 10% by weight hafnium solution.

2.3.2 Multiple Layered Metal-Organic Films (Reapplication)

The PAD method was applied to an already formed metal oxide layer to create thicker targets than were possible by spinning a concentrated metal ion solution only once. Due to the large size of the wafers, it was necessary to spin at an angular velocity of 2500 rpm to minimize comets and to form a uniform layer over the entire surface. The process resulted in multi-layered

films of HfO_2 with an average final height of 598 nm. Uniform film thickness across the width of the wafer was confirmed by imaging several locations by cross-section SEM. The change in height of hafnium(IV) oxide after each interval is summarized in Figure 2.10.

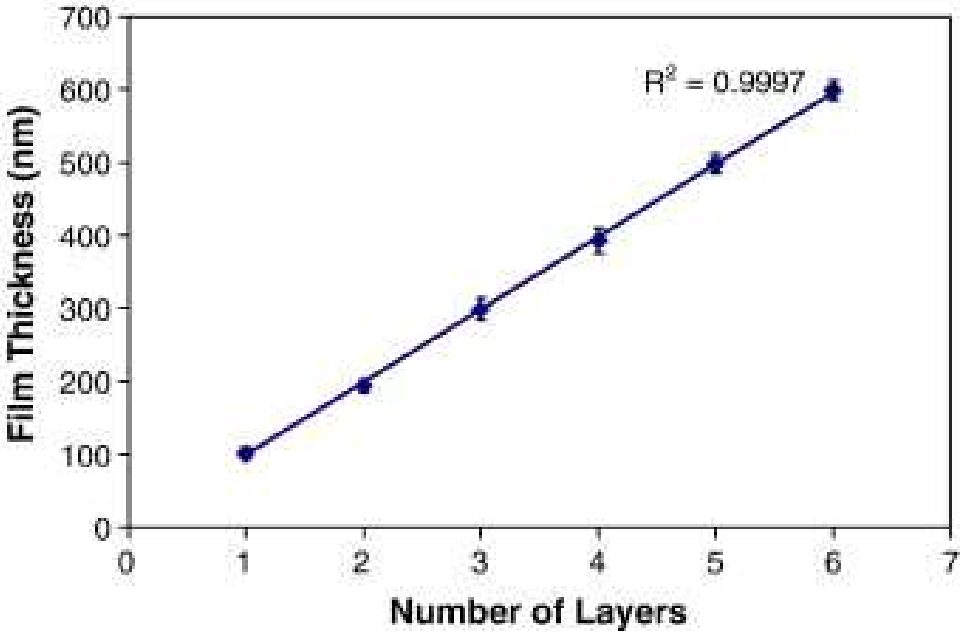

Figure 2.10 The graph shows the cumulative height of hafnium(IV) oxide after reapplication of the PAD method. The experiment was done in triplicate and the results averaged, except layer 6 which was done in duplicate. Uncertainty from the analytical balance translates to ± 2 nm.

Film thickness increased by 99 nm (± 2 nm) after each interval step. The upper limit of the method toward reapplication was not determined. Surprisingly, no striations were seen from a cross-section SEM (Figure 2.11) of the final wafer, indicating uniform growth.

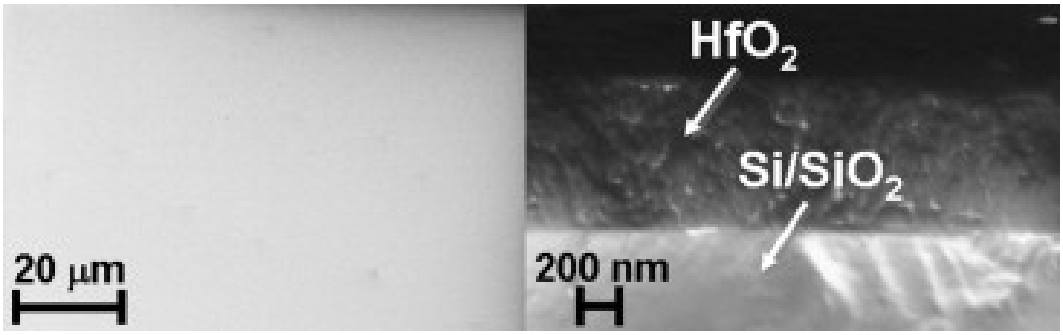

Figure 2.11 Left: Representative surface SEM of a hafnium oxide film produced from reapplication of the PAD method using a 10% by weight hafnium solution. Right: Representative cross-section SEM of the same hafnium(IV) oxide film.

An AFM image (Figure 2.12) of the film shows that the surface is composed of grains between 15 and 20 nm in diameter with a roughness rms of 2.35 nm over 1 μm^2.

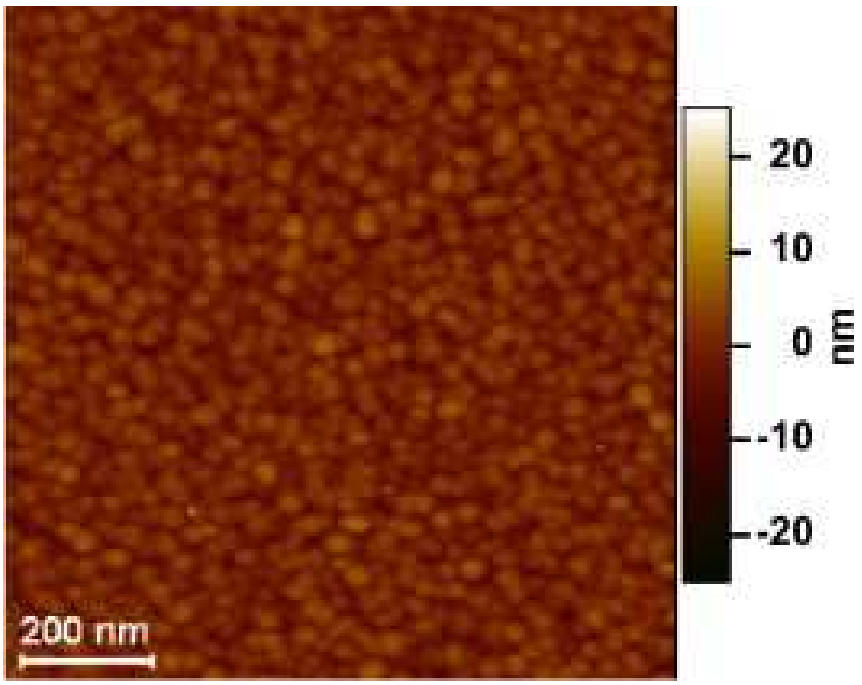

Figure 2.12 AFM image of a hafnium oxide film produced from reapplication of the PAD method using a 10% by weight hafnium solution.

X-ray Diffraction (XRD) analysis was conducted on the multi-layered HfO_2 film and compared with HfO_2 powder produced from baking several milliliters of a 9% (b.w.) Hf solution in a glass vial at 900 °C until a white powder formed. Figure 2.13 shows the close correlation between the HfO_2 film XRD pattern and the HfO_2 powder XRD pattern for polycrystalline HfO_2. The most intense peaks from the HfO_2 film are at 28.4°, 31.7°, 34.4°, 35.6°, and 55.8°. This agrees well with the library XRD pattern of simple monoclinic HfO_2 (PDF# 34-0104). The sharp peak at 33.1° is likely caused by the silicon backing. The relative intensities of the powder HfO_2 XRD pattern and the HfO_2 library XRD pattern were similar as seen in Figure 2.13a. However, differences in relative intensities in the HfO_2 film XRD pattern versus the HfO_2 powder XRD pattern were observed. This is likely due to the HfO_2 powder having a random distribution of lattice planes while the HfO_2 film has preferred orientations which skew the peaks. The broad widths of the peaks imply small grain sizes[58] in agreement with AFM and SEM analysis.

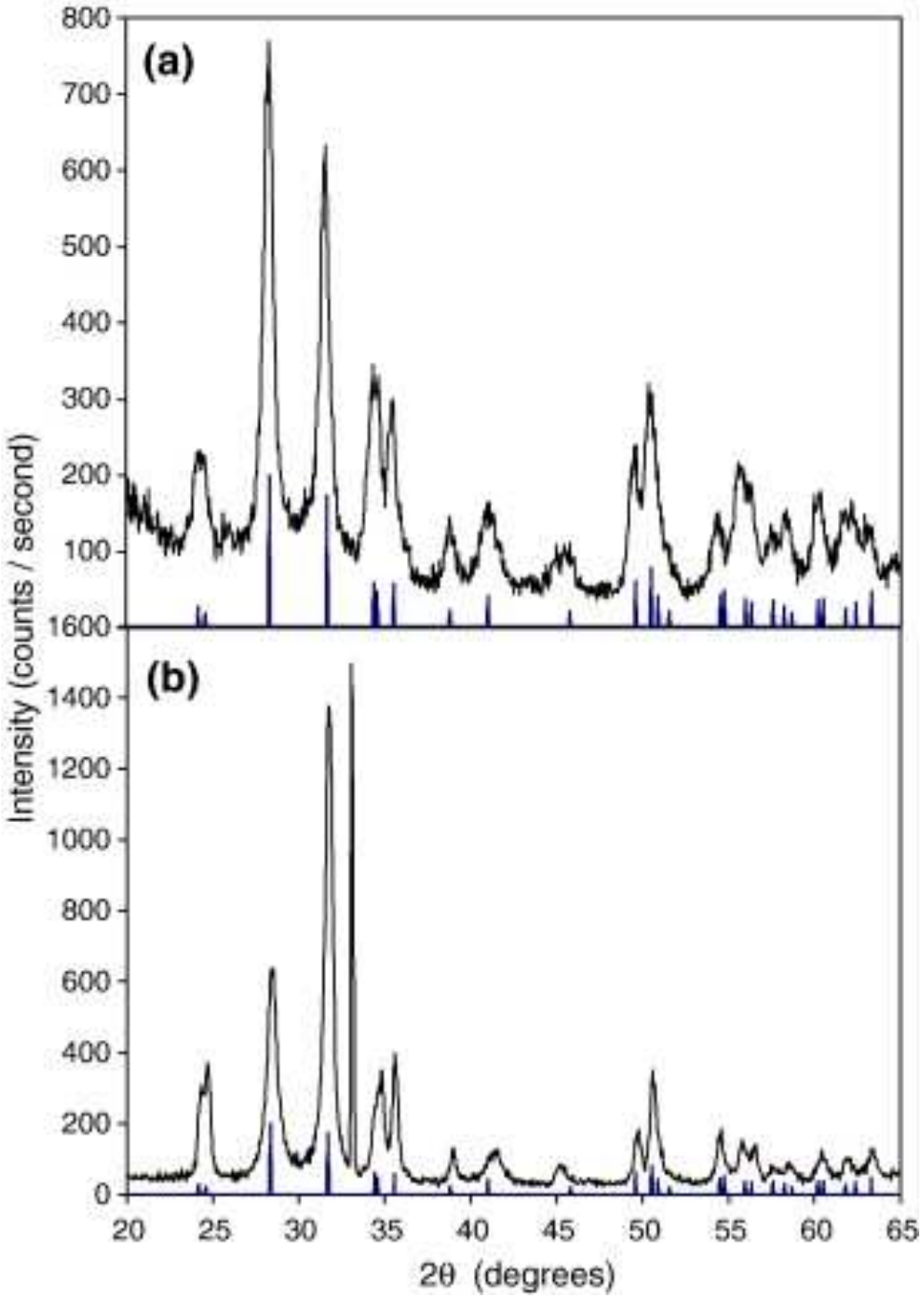

Figure 2.13 a.) X-ray Diffraction pattern of HfO_2 powder. b.) X-ray Diffraction pattern of a multi-layered film of HfO_2 on Si [100]. Below each XRD pattern is the relative intensities for simple monoclinic HfO_2.

2.4 Advantages of Polymer-Assisted Deposition

Molecular plating has been used by Trautmann et al.[59] and Mullen et al.[60] to create uranium, plutonium and curium oxide films. Their films ranged from 150 to 550 nm in thickness, which are typical for nuclear science applications. We have independently shown that the PAD method can produce films in a wider range of thicknesses than molecular plating. In addition, an advantage of using PAD is that it produces highly uniform and homogenous metal oxide films below 300 nm, while molecular plating method suffers in this region. A second advantage of the PAD method is that it can be reapplied to create thicker films without losing uniformity. A disadvantage of the PAD method is that the coating efficiency for a single application is ~ 33%. However, the solution that did not adhere to the surface can be collected and reused resulting in minimal loss of a precious metal, with the overall deposition efficiency approaching 100%. Molecular plating has ~ 90% efficiency for the deposition of metal out of solution and comparable to PAD.[61,62] Thus, the PAD method is likely a robust route to create metal oxide films suitable for nuclear science applications which require film uniformity and controlled film thickness.

It should be noted that the potential for silicon diffusion into the metal oxide films was not investigated, because this would not affect the overall energy loss of a high-energy beam passing through a metal oxide film and substrate. Future work will study the mechanical properties of metal oxide films of uranium, plutonium and curium created by the PAD method, before and after high intensity heavy-ion irradiation.

2.5 Development of Radioactive Spin Coating Methods

The process of spin coating applies a thin film of liquid evenly over a surface and is a crucial step in preparing targets by the PAD method. It is accomplished by utilizing centripetal force produced by rotating at angular velocities between 500 and 3000 rpm. Spin coating is used in many areas of technology, including fabrication of microcircuits, magnetic disks, compact disks, digital display devices, and other materials. The containment and recovery of the spin-coated solutions is not a common practice, and will be necessary when preparing radioactive targets by the PAD method. Prior designs for sealed spin-coating apparatuses are usually bulky and often require the use of vacuum pumps. Herein we show the design of a radioactive spin coating apparatus that is chemically inert and self contained.

Our radioactive spin coater design features a cylindrical sample-containment compartment made of poly(aryl- ether-ether)ketone (PEEK). A rectangular mount sits at the bottom of the compartment; it serves to hold the sample wafer in place while the apparatus is in use. A screw-on cap designed to fit snugly around the compartment, together with an o-ring of Viton® positioned at the outer rim of the cylindrical unit, ensure an air-tight seal and prevents leakage of the spin-coating solution when the spin coater operates at high rotations per minute. This compartment is attached to a 3-volt DC motor, which is connected to a variable voltage power supply. The DC motor's angular velocity dependence on the voltage setting of power supply is shown in Figure 2.14. The motor creates the rotating motion required to distribute the solute evenly.

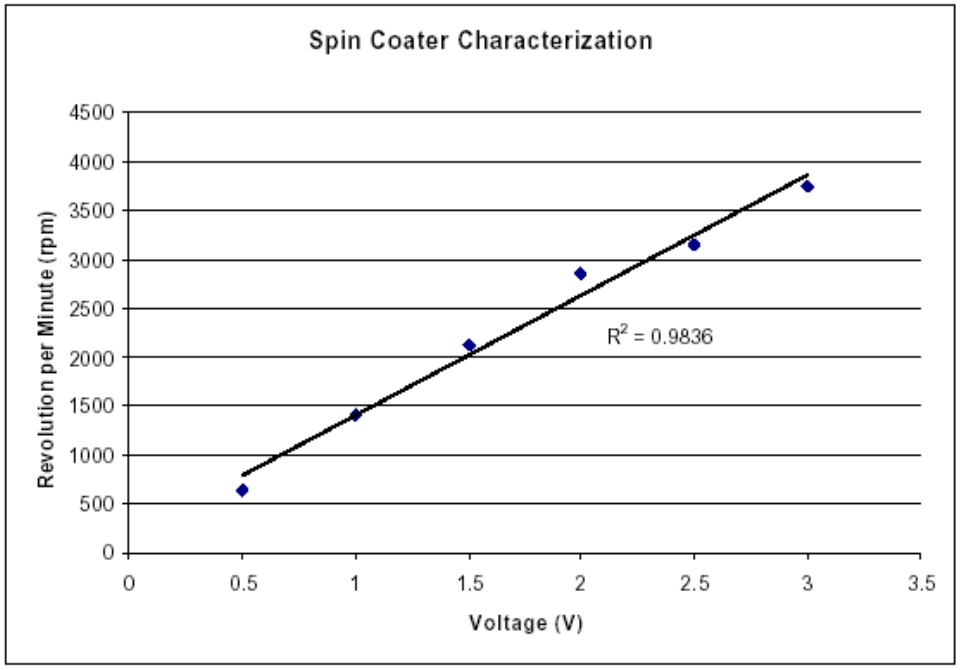

Figure 2.14 Angular velocity dependence on foltage of spin coater

The thickness of the film produced by spin coating is dependent on several variables including angular acceleration, speed of the rotation, air flow above the spin-coater, and the concentration and viscosity of the liquid used. A way to characterize a functional spin-coater is to observe a relationship between the concentration of the solution and the thickness of the film applied. In the particular case of the radioactive spin-coater characterization, the following procedure was used.

Polystyrene (MW: 350.00) was dissolved in toluene to prepare seven solutions with concentrations shown in Table 2.1. Glass wafers (1.85 cm x 0.85 cm x 1.2 mm) were rinsed briefly in acetone and dried with compressed air to clean dust and other contaminants from the surface. After being placed on the mount, 0.1 mL of a polystyrene solution was added to the top of a glass slide with an automatic pipette.

Series No.	Volume of Solution Added (μL)	Concentration (g/100mL)	Calculated Thickness (nm)
1	100	2.858	221.48
2	100	5.424	523.95
3	100	7.991	981.97
4	100	10.777	1959.28
5	100	13.337	2173.11
6	100	15.983	3743.99
7	100	18.808	5475.11

Table 2.1. Thickness of polystyrene films at various concentrations without cap

Once the solution was added in small aliquots to the glass slide , the spin-coater was switched on and set to rotate at 2,790±50 rpm. The samples were spun for 3 minutes each. They were then removed from the spin-coater and their thicknesses were measured by a UV-Vis spectrometer. Equation 2.1 and 2.2 were used to determine the thickness from spectral data. The calculation results are presented in Table 2.2.

$$\nu = \frac{\frac{n(\lambda_1)}{\lambda_1} - \frac{n(\lambda_2)}{\lambda_2}}{N_{cyc}} \tag{2.1}$$

$$d = \frac{1}{2\nu} \tag{2.2}$$

Constants $n(\lambda_1)$ and $n(\lambda_2)$ are the refractive indices at their corresponding wavelengths; the value of 1.6 was used for polystyrene of both molecular weights. N_{cyc} is the number of cycles in the interference fringes between wavelengths λ_1 and λ_2. For data consistency, λ_1 and λ_2 are chosen to cover the range that contains minimal noise interference (~360-790 nm). Finally, d represents the thickness of the film calculated. Wavelength range in the interpretation of UV-Vis

data is limited to between 350-1000 nm because glass has a strong absorption at wavelengths below 300 nm. A sample UV-Vis spectrum is shown below in Figure 2.15.

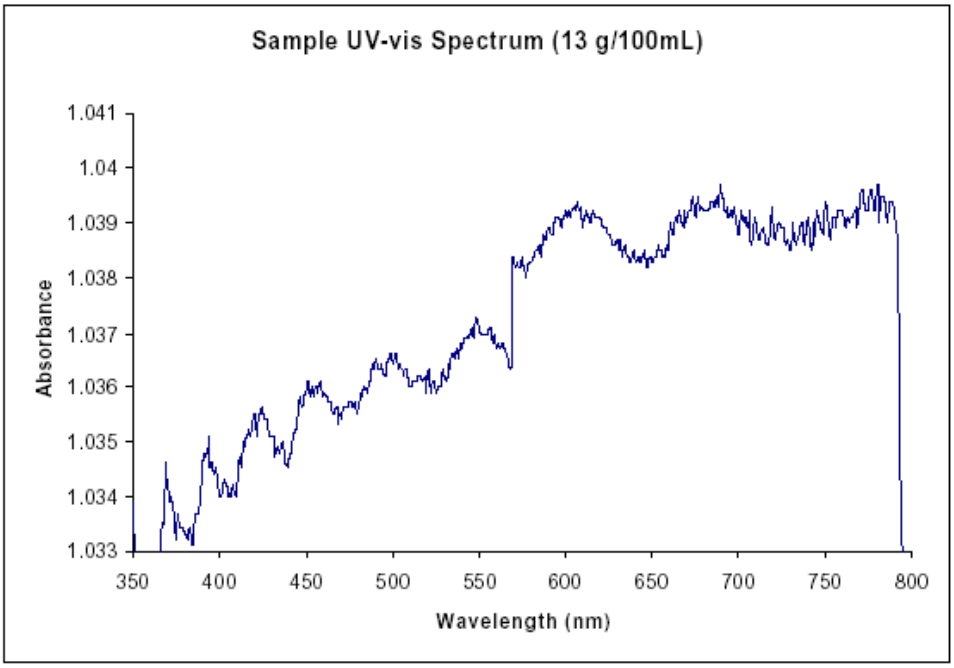

Figure 2.15. A representative UV-Vis spectrum of polystyrene film at 13 g / 100 mL.

By measuring film thicknesses of various polystyrene concentrations using a UV-Vis spectrometer, an exponential relationship is observed. Graph of concentration versus thickness is shown in Figure 2.16.

In order to ensure that the container and lid assembly would not leak, a highly fluorescent die (Rhodamine B) was spin coated. The same steps were repeated using 0.1 M of Rhodamine B solution. Its vibrant color and UV fluorescence allow an easy detection of any leak were it to occur during the procedure. By visual inspection, it is reasonable to conclude that no leak took place during the operation of the spin coater, both with and without the cap. All of the liquid was contained within the PEEK device and none was found to have been slung out into the chemistry

hood and surrounding lab benches.

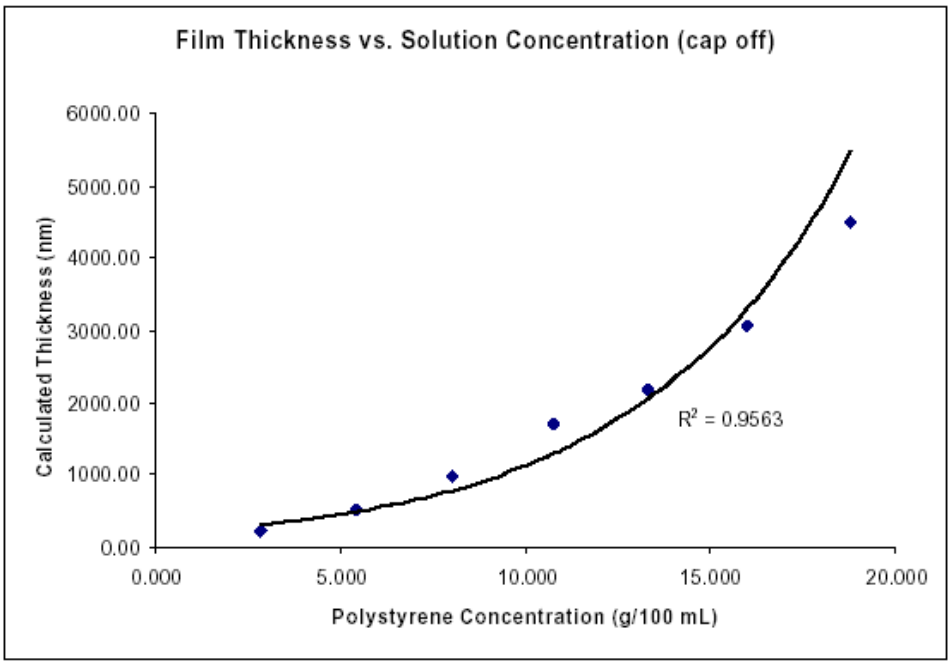

Figure 2.16 Film thickness vs. solution concentration with the cap off.

To ensure that the radioactive spin-coater will perform adequately even with an air-tight seal, the same procedure was executed with the cap on. Selected concentrations of polystyrene (MW: 350.00) dissolved in toluene were spun on glass slides. The experimental set up was the same as described previously, all except the cap was screwed on to produce an air-tight seal. In addition, the spin coater was set to operate for 1 minute at 1000 rpm. The film thickness and concentration of the corresponding solution demonstrate an exponential relationship. The numerical data and graph are shown in Table 2.3 and Figure 2.17. This result demonstrates that the cap allows the radioactive spin-coater to contain the solution used without compromising the quality of the film produced.

Series No.	Volume of Solution Added (µL)	Concentration (g/100mL)	Calculated Thickness (nm)
1	100	10.777	1239.401897
2	100	13.337	1725.05603
3	100	15.983	2779.263195
4	100	18.808	4541.995192

Table 2.3 Thickness of Polystyrene films at various concentrations with cap.

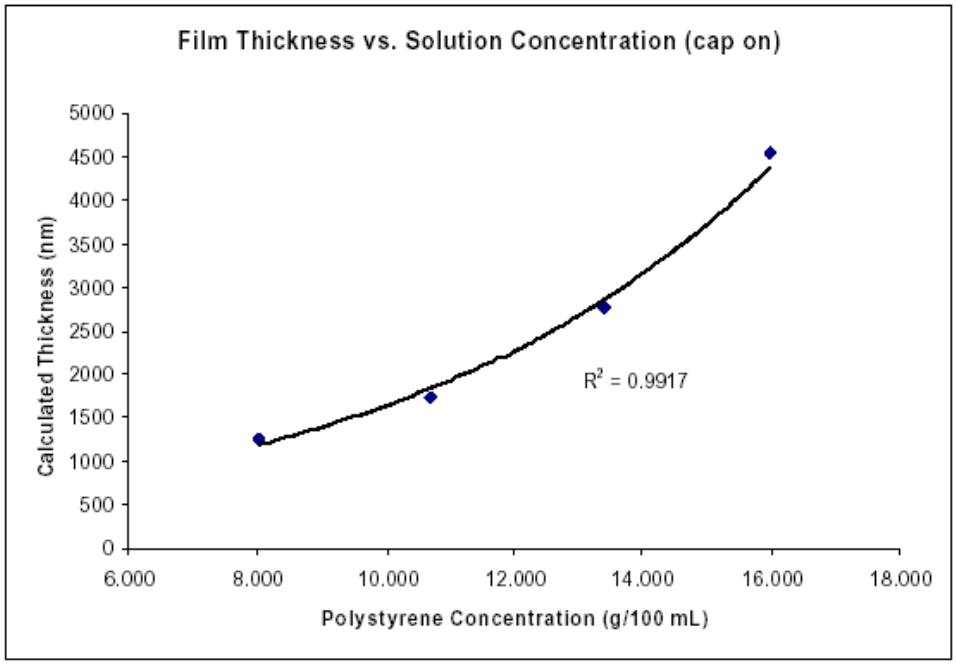

Figure 2.17 Film thickness vs. solution concentration with the cap on.

The mount size inside the spin-coater can be modified to fit individual needs. The current design accommodates rectangular glass slides with the dimension 1.00 cm x 2.90 cm, see Figure 2.18. With proper alteration, the apparatus can be used to spin coat wafers of various shapes and sizes. In addition, the height of the mount and the well volume can be adjusted to accommodate various experimental designs. The angular velocity of the radioactive spin-coater can be varied by adjusting the voltage setting on a power supply connected to the motor.

Figure 2.18 Images of the radioactive spin-coater.

The original intent of the design of a radioactive spin coating apparatus was to improve methods of nuclear target production. However, the radioactive spin-coater can also be a valuable asset in creating films from other hazardous or expensive solutions.

2.6 Conclusion

Europium, thulium and hafnium oxide films were annealed onto silicon substrates using polymer-assisted deposition. The films were characterized by SEM, AFM and profilometery. The characterizations showed that the films were crack free, uniform, and homogenous. Advantages of PAD over molecular plating include that it produces high-quality thin films below 300 nm in thickness and that reapplication of the method creates thicker metal oxide films of equal quality.

Chapter 3: Irradiation of Polymer-Assisted Deposition Targets

3.1 Introduction

The purpose of this chapter is to describe our experiments of irradiating targets made by the Polymer-Assisted Deposition (PAD) method. A modified procedure was used for the manufacturing of these targets than that described in chapter 2. The modification was the use of silicon nitride backing instead of the pure silicon that we used previously. The reason for the change is because silicon nitride can be made to be as thin as 100 nm but silicon can not be thinned to that thickness over a large area. The results from heavy-ion irradiation of several metal oxide targets, prepared by the PAD method on silicon nitride backings, at the 88-in Cyclotron of the Lawrence Berkeley National Laboratory (LBNL) will be discussed. Characterizations of the targets after irradiation were performed by scanning electron microscopy (SEM) and atomic force microscopy (AFM). Target integrity during irradiation was monitored by Rutherford scattering as a function of time.

3.2.1 Target Manufacturing

The metal oxide targets were prepared as follows. A low-pressure chemical vapor deposition (LPCVD) furnace (tystar, TYTAN) was used to epitaxially grow low-stress silicon nitride (Si_3N_4) onto both sides of 4-in single-sided polished (SSP) silicon wafers [1 0 0]. Dichlorosilane and ammonia with flow rates of 100 and 25 sccm, respectively, were used as the silicon nitrating source at 18.7 Pa and 835 °C. The film stress was about 200 MPa and the deposition rate was 40 Å/min. The Si_3N_4 film was produced to yield 1000±20 nm (344±7 $\mu g/cm^2$) before etching. Four rectangles (1.8×1.3 cm) were scratched through the

Si_3N_4 layer, on the unpolished (bottom) side of the silicon wafers, to expose the silicon surface. These patterned wafers were then placed for 8–10 h in a 20% potassium hydroxide bath at 80 °C, to etch away the silicon and leave only a rectangular surface of Si_3N_4, a scheme is shown in Figure 3.1.

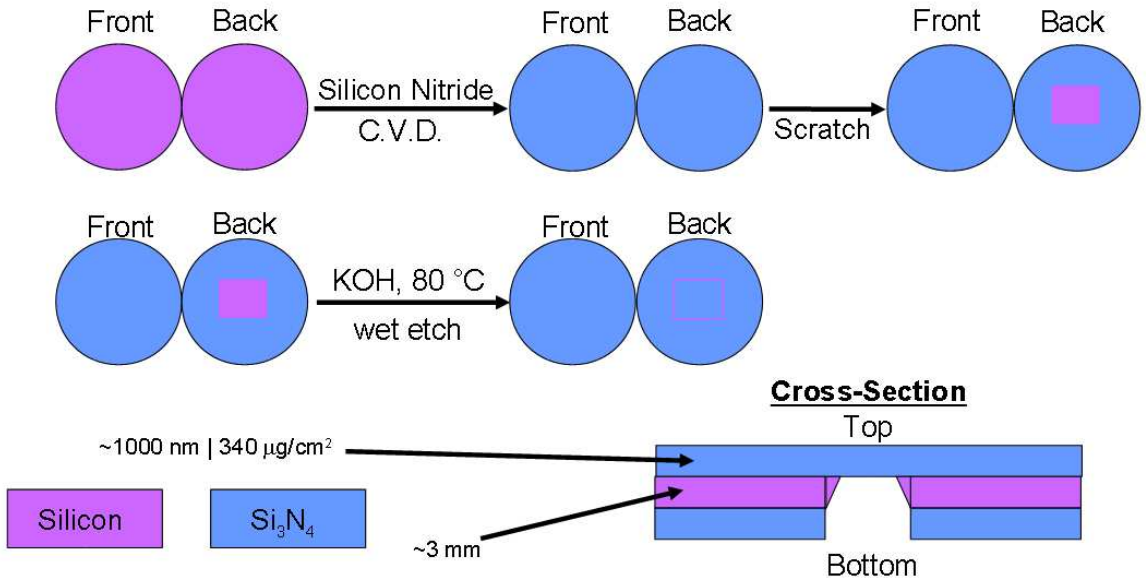

Figure 3.1 The process of making silicon nitride backings.

3.2.2 Solution Preparation

A solution of 8% by weight (b.w.) thulium with 15% (b.w.) polyethylenimine (PEI) in water was prepared exactly as previously described in chapter 2. A 3 mL aliquot of the thulium solution was spread on top of wafers containing the four Si_3N_4 windows. The wafers were spun at 1500 rpm and then baked over a period of 3 h in a muffle oven to a final temperature of 600 °C. The process creates a thin layer of thulium(III) oxide (Tm_2O_3), while the polymer decomposes to volatile oxides. A second layer of Tm_2O_3 was coated on the wafer by adding an additional 3 mL aliquot of the 8% (b.w.) thulium solution and repeating the steps described

above. The final thickness of the Tm_2O_3 film was ~250 nm (~220 μg/cm^2) as measured from taking the average of several cross-section images by SEM. The wafers were scribed and broken into pieces of 7.6×6.0 cm. The outer silicon was used as a frame to support the metal oxide film and silicon nitride window on a target ladder for heavy-ion irradiation.

3.2.3 Irradiation Conditions

Two Tm_2O_3 targets were each irradiated separately with a different heavy-ion beam of either 208 MeV Argon ($^{40}Ar^{10+}$), or 180 MeV Argon ($^{40}Ar^{8+}$). The first target was irradiated with the Ar^{10+} beam for 35 min at a beam current of 3.1×10^{11} particles/s (50 particle nA) for a total dose of 6.6×10^{14} particles. A second target was irradiated with the Ar^{8+} beam for 17 min at a maximum beam current of 1.2×10^{12} particles/s (200 particle nA) for a total dose of 1.3×10^{15} particles. The irradiated targets were easily removed from the target ladder and were subjected to further analysis.

3.3.1 Visual Inspection and Rutherford Scattering

Visual inspection of the Tm_2O_3 target irradiated by the $^{40}Ar^{10+}$ beam revealed a slightly darkened rectangular region which is shown in Figure 3.2 but difficult to see. This is hard to see due to the highly transparent nature of targets produced by this method, see Figure 3.3. The dimensions and location of the darkened rectangle are consistent with the known beam spot image after beam collimation. The Tm_2O_3 target irradiated by the higher intensity (200 particle nA) $^{40}Ar^{8+}$ beam had a melt spot after irradiation. This was done to test the limits of the method as stationary targets only receive a maximum intensity of ~40 particle nA during an irradiation to

prevent melting. The melt spot for the intensely irradiated Tm_2O_3 target indicates an upper-limit of beam intensity that stationary targets prepared by the PAD method can withstand. From monitoring the quantity of Rutherford scattered particles as a function of time, no decrease in intensity for either target was observed; this indicates target flaking did not occur during the irradiation.

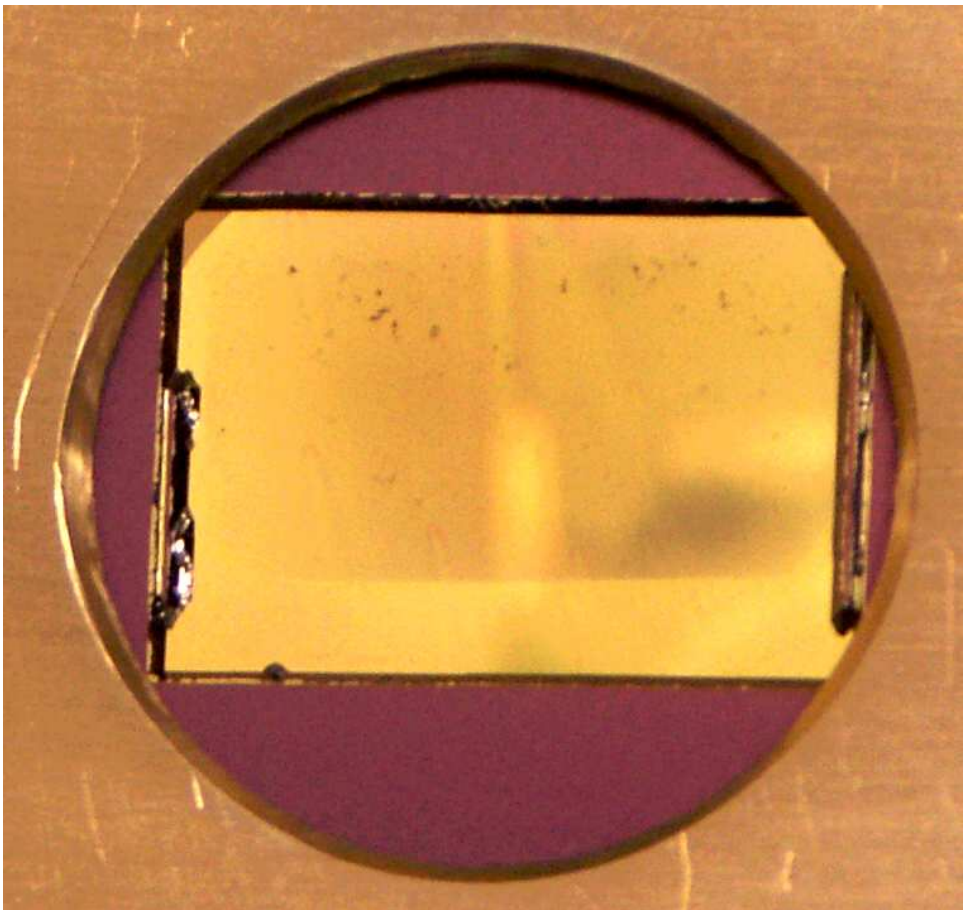

Figure 3.2 An irradiated Tm_2O_3 target mounted behind the target ladder is shown in the foreground. A slightly darkened rectangular region can be seen at the top of the silicon nitride that corresponded to the beam spot image. The purple color is the silicon frame used to support the silicon nitride layer. The metallic surface is the aluminum target ladder in which the targets were mounted. In the background is a non-focused image of the support frame.

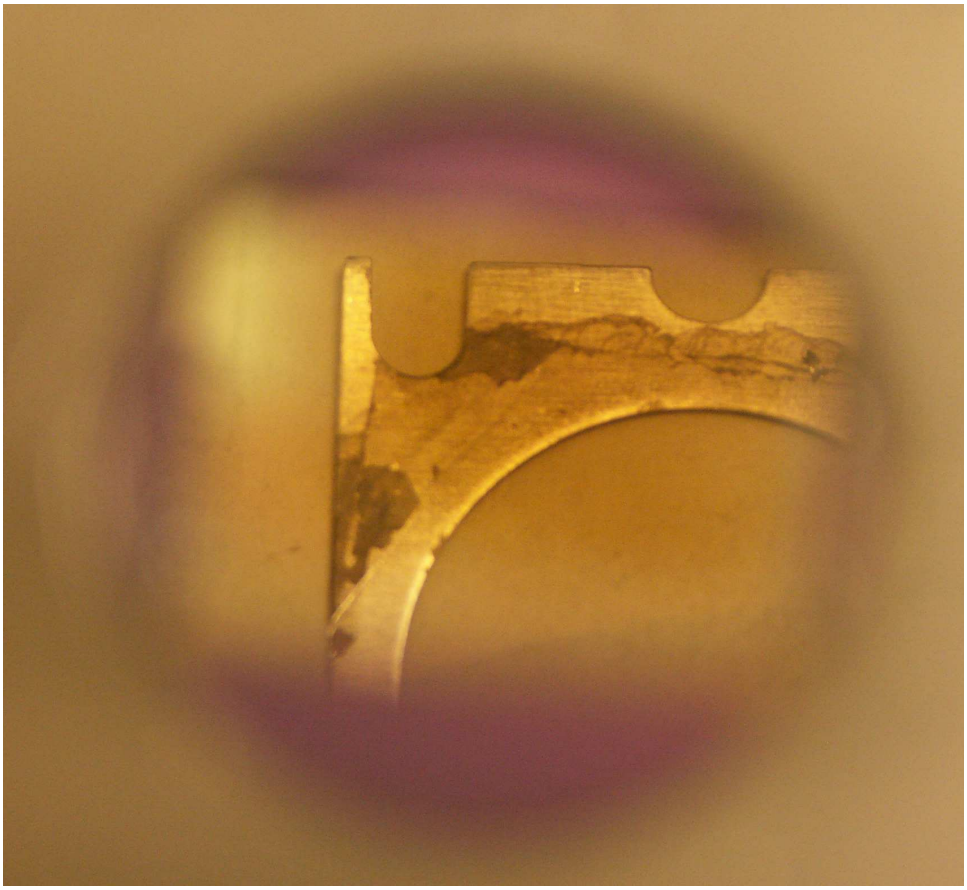

Figure 3.3 The same target is shown in Figure 3.2 except the image is focused on the background, highlighting the transparent nature of targets produced by the PAD method. The object shown in the background in the support frame used for the irradiation.

3.3.2 Scanning Electron Microscopy Inspection

An SEM (Ultra 55VP Fesem, Zeiss) with an acceleration voltage of 5 kV was used to determine surface homogeneity of the $^{40}Ar^{10+}$-irradiated target at a magnification of 274×. A cross-sectional SEM image was also taken to determine uniformity at a magnification of 31 k×. A representative image of the surface after the irradiation is shown in Figure 3.4a. Small surface aggregates randomly distributed on the surface can be seen, and were present before irradiation. This is likely due to using non-clean room techniques in the application of the PAD method. A

small pinhole ~7×4 μm is also visible in the upper-right-hand side of Figure 3.4a. Pinholes are seen prior to irradiation and are likely due to defects in the substrate, thus the wafers should be cleaned thoroughly before adding silicon nitride to the surface. A representative SEM cross-section of the target after irradiation is shown in Figure 3.4b; two layers of Tm_2O_3 can be clearly seen in the figure. The back edge was tilted ~8° toward the top of the wafer, in order to image the surface of Tm_2O_3 in tandem with viewing the striated Tm_2O_3 layers. The figure shows the high degree of uniformity achieved with the PAD method. It should be noted that the previous experiments described in chapter 2 did not show striations when utilizing the reapplication of the PAD method, this may be due to not baking the silicon nitride targets up to 950 °C but only to 500 °C (silicon nitride surfaces start to decompose near 600 °C). The occurrence of striation is likely dependent on the annealing temperature profile. The SEM analysis of the irradiated target show no significant deviation from those imaged prior to irradiation.

3.3.3 Atomic Force Microscopy Inspection

An AFM (MFP 3D, Asylum Research) was used to obtain a high-resolution image of a 1×1 μm^2 representative section of the irradiated target surface and a non-irradiated portion of the same target for $^{40}Ar^{10+}$ irradiation. The AFM used cantilevers with a spring constant of 3 N/m (Multi75, Budget Sensors) for imaging in the attractive regime in the AC mode. The AFM images are shown in Figure 3.5a and b. The homogeneity and granular appearance of the Tm_2O_3 layer on the non-irradiated portion of the target is consistent with previous experiments with a root mean square (RMS) roughness of 1.1 nm. The irradiated portion, Figure 3.5b, also shows a homogeneous surface with low surface variability with an RMS roughness of 2.0 nm.

After irradiation, the homogeneity of the film is only slightly reduced.

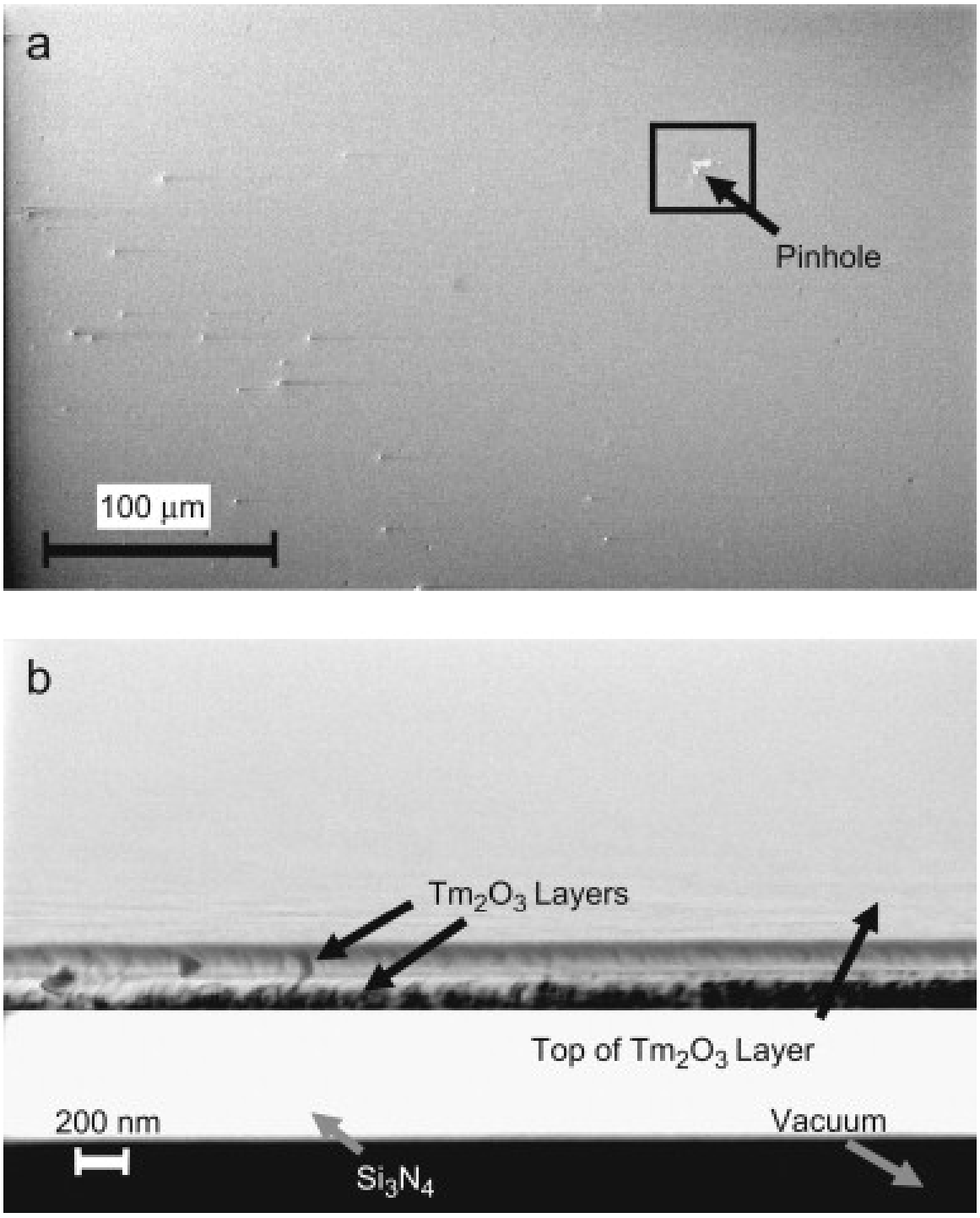

Figure 3.4 (a) Representative surface SEM of the irradiated Tm_2O_3 film produced from the PAD method using an 8% by weight Tm solution. (b) Representative cross-section SEM of the same

Tm_2O_3 film.

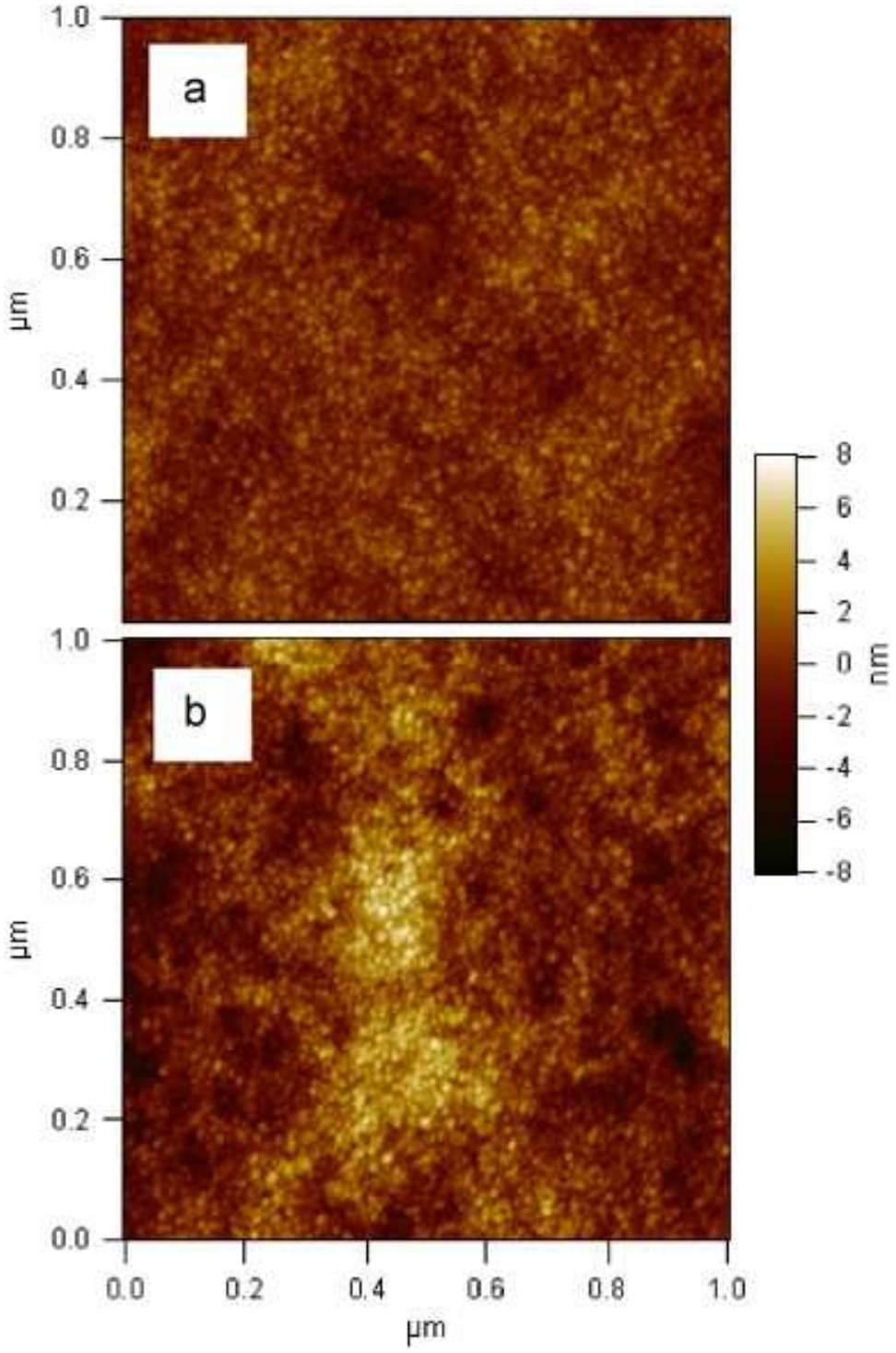

Figure 3.5 (a) Representative AFM of the non-irradiated area of the Tm_2O_3 film produced from

the PAD method, (b) representative AFM of the irradiated area of the Tm_2O_3 film produced by the PAD method.

3.4 Conclusion

From monitoring the Rutherford scattered during irradiation, it was observed that there was no loss of target material. This is in contrast to other methods that may produce targets that flake even before irradiation.[63] Loss of target material through the course of a beam irradiation will decrease the overall production of the desired nuclear reaction channel, and may prelude the undesired consequence of contamination if the target is composed of a radioactive isotope. Another consequence of target flaking is the loss of expensive enriched isotopic material. Therefore, the PAD method is an advantageous target methodology for preparing uniform, homogeneous crack-free metal oxides for heavy-ion irradiation. Further experiments are planned to determine the target temperature during irradiation and the optimum thickness of silicon nitride to minimize the energy-transfer from heavy ions, while retaining high structural integrity.

Chapter 4 Design and Testing of Gas-Phase Apparatus

4.1 Introduction

The research objective was to extract radionuclides from the Berkeley Gas-Filled Separator (BGS, discussed in detail in Chapter 5) perform chemistry on them and then measure the yield of the chemical process by monitoring their decay. In order to accomplish this, electronics, gas-jet systems, and detector arrays were designed and constructed. Computer programs were written to control these systems and scripts were written to analyze the data. A schematic of the process is shown in Figure 4.1

From the BGS radioactive atoms of interest were stopped in the gas-phase by the use of the Recoil Transport Chamber (RTC, described in chapter 1), the atoms were transported via a gas-jet to the alpha-wheel apparatus that is described in this chapter and the decay was monitored with the electronics, data acquisition systems, and data analysis codes described herein.

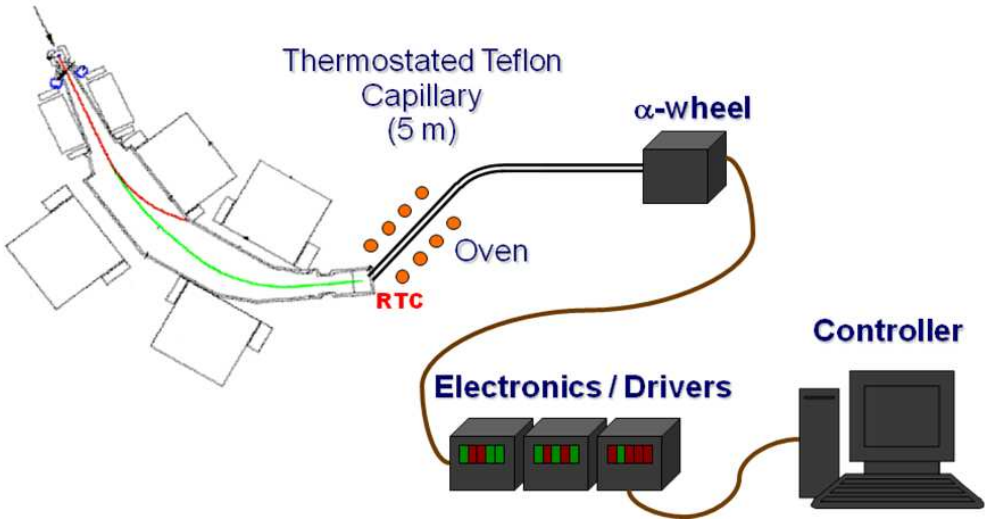

Figure 4.1 A cartoon schematic of the hardware constructed for the gas-phase testing experiments that were performed; the results are presented in this chapter.

4.2 The Alpha-Wheel Apparatus

The detectors were housed in the alpha-wheel apparatus. Radionuclides were transported from the BGS via gas-jet that sprayed them onto a rotating sample holder that passed in front of alpha-detectors; the sample holder plus alpha-detectors is called the alpha-wheel apparatus, see Figure 4.2.

Figure 4.2 This is a picture of the alpha-wheel apparatus. There are three of the four alpha-detectors positions filled in the device. The sample holder is the beige disk to the right. The cap is made from aluminum and is to the left with the hole for the capillary next to the red arrow..

The lid has a hole in the upper right corner (shown to the left of Figure 4.2 with a red arrow) to fit the capillary. The capillary was composed of Teflon and was the exit of the gas-jet. The gas-jet sprays a small spot onto the sample holder. The location it sprayed on the sample holder is covered in 1 μm Mylar foil. The radiation emitted from the radionuclides easily passes through the Mylar foil and strikes the detectors in the alpha-wheel apparatus. A figure of the Mylar covered sample holder in the alpha-wheel apparatus is shown in Figure 4.3. Also shown to the left of the figure is the nozzle that leads to a vacuum for exhaust.

Figure 4.3. A picture of the sample holder with the Mylar foil in the alpha-wheel apparatus.

The alpha-wheel utilized semiconductor detectors for the detection of ionizing radiation from alpha-decaying radionuclides. The semiconductors detectors used for this project were solid crystals of silicon. The valency for silicon is 4, and thus forms 4 bonds with other silicon neighboring atoms. Because all of the silicon's electrons are tied into bonding the band-structure

of silicon has a filled valence band and an unfilled conduction band. The separation in energy between a material's valence band and conduction band is called the energy gap, and this is what determines whether a material is considered an insulator or a conductor; the band gap for silicon is 1.1 eV. Due to the statistical nature of thermal energy there will be a small number of electrons in the valence band that will have significant energy to cross the energy gap and enter the conduction band. The atom that has lost a covalently bound electron is termed a “hole”. An electron from a neighboring atom will be dislodged and will fill the hole. However, this process creates a new hole at the second atom. Thus it will appear as if the hole migrates across a crystal, but it is electron's from neighboring atoms that move. The probability, p(T), that an electron-hole pair is created from thermal fluctuations in a semiconductor crystal is given in Equation 4.1.

$$p(T) = CT^{3/2}e^{-\frac{E_g}{2kT}} \tag{4.1}$$

where T is the temperature in kelvins, C is a constant of proportionality, k is Boltzmann's constant and Eg is the bandgap energy. The equation shows that the probability density of electron-hole pairs is strongly correlated to temperature; as the temperature decreases the concentration of electron-hole pairs will decrease exponentially. At any given temperature the process of electron-hole production will equal the rate of electron-hole destruction due to recombination; at equilibrium the number of electrons in the conductance band (n_i) will equal the number of holes in the valence band (p_i).

$$n_i = p_i \tag{4.2}$$

The subscript 'i' signifies that the semiconductor crystal is pure and has not been doped.

It is possible in semiconductors to control their electrical conductance by the addition of dopants into the crystal lattice of the semiconductor. Dopants are atoms with a valency different than 4, and often have a valency of either 3 or 5. By adding a dopant of valency 5 (*e.g.* phosphorus, arsenic, antimony) there will be four electrons in the valence band forming covalent bonds, but the fifth electron will be relatively free to travel and will be in a state only slightly beneath the conduction band termed donor states. The type of semiconductor crystal that has been doped with atoms of a valency of 5 or greater is called an n-type semiconductor. In contrast, if an atom of valency 3 or less is used to dope a semiconductor then those dopant atoms won't have enough electrons to form bonds with neighboring silicon atoms. This produces an abundance of holes in the material, and forms acceptor states just above the valence band; this type of crystal is called a p-type semiconductor, shown in figure 4.4.

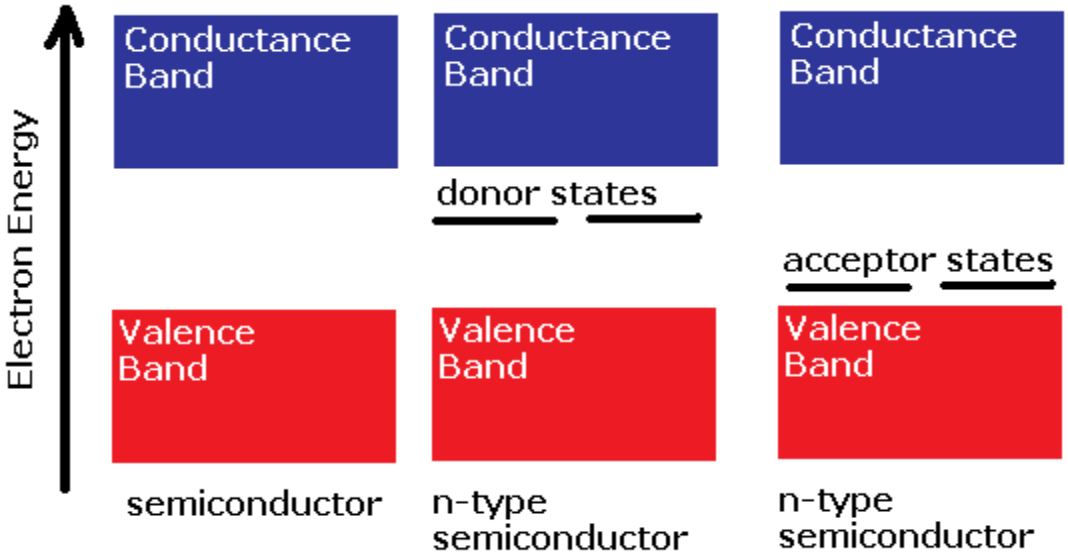

Figure 4.4 Electronic energy structure for various semiconductor materials.

The velocity (v) of the electrons and holes is governed by the strength of an applied

electrical field (E) and proportional to their respective mobilities (μ).

$$\nu_h = \mu_h E \quad \textbf{(4.3)}$$

$$\nu_e = \mu_e E \quad \textbf{(4.4)}$$

However, when an external field is high enough the velocity of the electrons and holes is saturated and are on the typical order of 10^7 cm/s.

The semiconductor crystals are useful in detecting ionizing radiation in the following way. When two semiconductor crystals of different types are brought together, electrons from the n-type semiconductor will diffuse into the p-type material; this continues until the flow of electrons is stopped by the electric field that is from the imbalance of charge, this area of the detector is called the depletion zone. When an ionizing radiation particle enters the depletion zone it creates a series of electron-hole pairs by displacing electrons from their atoms along the trajectory of the particle. Ideally, the ionizing particle should lose all of its energy within the depletion zone of the semiconductor crystal, because the electric field that has been in the depletion region plus the applied voltage will flow electrons in one direction and holes will flow in the opposite direction, shown in Figure 4.5. The charges are collected and converted to voltages using a charge-to-voltage amplifier, and the measured voltage is directly proportional to the energy of the radiation.

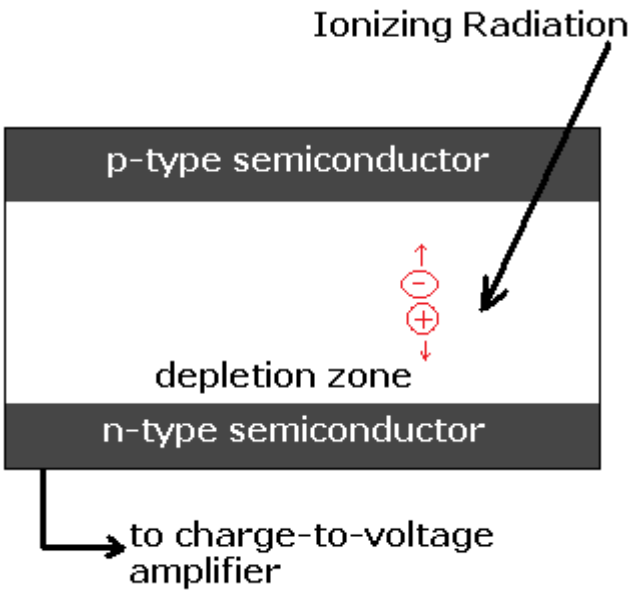

Figure 4.5 A cross sectional image of a semiconductor detector with ionizing radiation causing the formation of electron-hole pairs.

As mentioned earlier the depletion zone will naturally have an electric field whose potential (ϕ) is given by equation 4.5

$$\nabla^2\phi = -\frac{\rho}{\epsilon} \tag{4.5}$$

Where ρ is the charge density and ε the dielectric constant for silicon. The magnitude of the potential is often very near the bandgap of the material. The associated electric field (E) can easily be solved for by taking the derivative of the potential along the axis of p-to-n.

$$E(x) = -\frac{d\phi}{dx} \tag{4.6}$$

As the ionizing particle removes electrons from atoms it will begin to slow down and come to rest. During the slowdown phase the ionizing particle will knock-out electrons imparting

on them a sufficiently high energy that these electrons (termed delta rays) create electron-hole pairs as they decelerate. The average energy the ionizing particle loses to produce a single electron-hole pair is often called the ionization energy and has the symbol ε. The ionization energy is dependent on the kinetic energy of the ionizing particle. The ratio of electrons to holes formed from the energy deposited by the ionizing particle is 1:1, whether the crystal has an n-type or p-type dopant plays no role in the ratio.

The ionization energy is approximately 3 eV for silicon. This is small in contrast to other detector types like a typical gas-filled detector, which requires 30 eV to make an ion pair. As the number of electron-holes will be 10 times greater for silicon than a gas-filled detector this means that the attainable resolution is also greater. However, the ionizing energy is dependent on both the temperature of the silicon and the type of radiation measured. Thus, if you were to measure protons using a calibration from alpha particles the energy value could be off by as much as 2.2%.[64]

As said earlier, the number of electron-hole pairs created by an ionizing particle is directly proportional to the energy of that particle. However, the exact number of electron-hole pairs generated from another ionizing particle of the exact same energy and type will differ slightly. If we plotted the number of electron-hole pairs generated against the energy of the ionizing particles we will see a Gaussian distribution of values. The Gaussian distribution is given as,

$$G(H) = \frac{A}{\sigma(2\pi)^{1/2}} e^{-\frac{(H-H_0)^2}{2\sigma^2}} \qquad (4.7)$$

Here σ is related to the full width at half maximum (FWHM) by FWHM = 2.35σ, H_0 is the

centroid of the peak, and A is the area of the peak. Thus, we finally have the tools to quantify the observed distribution. It is often considered that the distribution should be due mainly to statistical noise, meaning that it arises from the random fluctuation of events even though the particle may have exactly the same energy and enter the material in essentially the same location. This distribution can be modeled as at the limit of a Poisson process, and the resolution is defined as the FWHM divided by the location of the centroid (H_0), see equation 4.8

.

$$R|_{Poisson} = \frac{FWHM}{H_0} \tag{4.8}$$

As the location of the centroid is equal to the number of electron-hole pairs (N) times a proportionality constant (K), and as mentioned earlier the FWHM equals 2.35σ where σ equals $KN^{1/2}$, then equation 4.8 becomes,

$$R|_{Poisson} = \frac{FWHM}{H_0} = \frac{2.35KN^{1/2}}{KN} = \frac{2.35}{N^{1/2}} \tag{4.9}$$

As the equation shows, the resolution is determined by the number of electron-hole pairs generated, so as N increases the resolution will increase (R will decrease). However, semiconductor detectors are odd in that the measured resolution is often 3 or 4 times better than that predicted from a statistical analysis. This simply means that the process described earlier for how electron-hole pairs are generated is not independent of each other. This increase in resolution is quantified by a variable called the Fano factor (F), and is related to equation 4.10 as,

$$R|_{Poisson} = 2.35\left(\frac{F}{N}\right)^{1/2} \tag{4.10}$$

For semiconductor crystal detectors the Fano factor (F) will be less than one.

As mentioned earlier these semiconductor crystal detectors often have an external potential applied to them. They will also work when there is no potential applied, but the resolution is greatly diminished. That is because the natural potential that is made across the detector, ~1 V, is not strong enough to sweep the electron-hole pairs to the electronics. Any delay in collecting the ion-pairs will cause a loss of information due to trapping and recombination. Thus, the detector is biased. The bias applied is usually the reverse of that generated naturally by the separation of ions. The p-side of the detector is biased negatively, and the n-side of the detector is biased positively. This ensures that the minor carriers of the n-side (holes) and the p-side (electrons) are attracted across the junction. As the concentration of carriers is so little across the junction the current across the detector is very low. So the entire detector can be thought of as a diode, and as with most diodes if the voltage is increased too much in the opposite direction it will permanently destroy the diode.

4.3 Driving the Stepper Motor

Precise control of the sample holder is crucial in these experiments, as the sample holder is stepped in front of the alpha-detectors. We used a stepper motor (Anaheim Automation, 1SY001S-LW4) to this end. A stepper motor is shown diagrammatically in Figure 4.6.

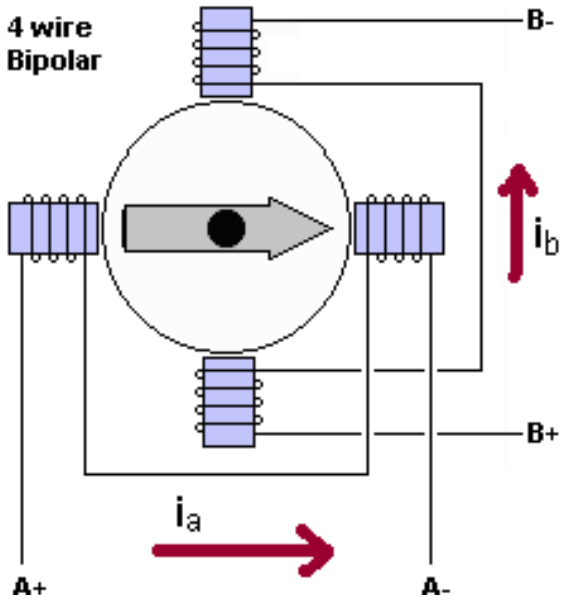

Figure 4.6 A diagram of the 4-wire stepper motor.

The beginning operation of the stepper motor is to pass current through points A+ to A- (winding A) and from points B+ to B- (winding B). In order to have the motor take a step it is necessary to change the current direction of either winding A or winding B but not both at the same time. An example current switch with an associated step is shown in Figure 4.7.

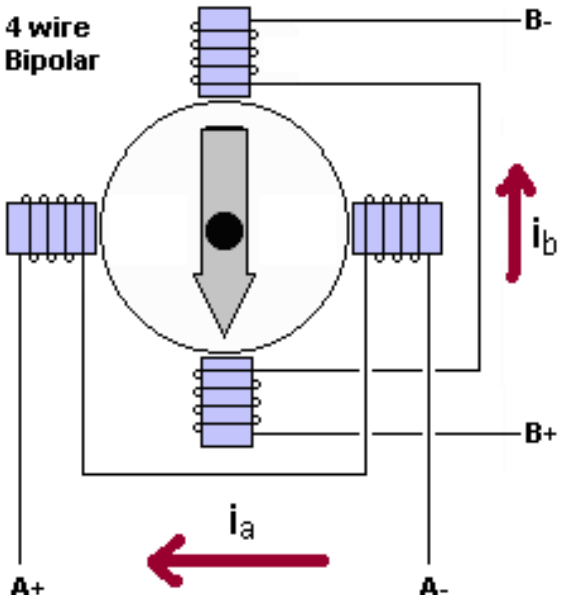

Figure 4.7 The stepper motor diagram after a current change showing the motor has stepped by 90 degrees.

By changing the direction of the current in the right order it is possible to make the motor rotate clockwise or counterclockwise. In order to do this a circuit was made and is sketched in Figure 4.8. The integrated circuit used to control the switching for V_1 and V_2 was the IC-4049. The logic of the circuit contains 2 bits of information for 4 total combinations: 00, 01, 10, 11. Each of the combinations corresponded to the current in winding A flowing in one direction or the other and similarly for the current flowing in winding B, to cover all possible combinations to make the wheel step. Thus with this simple logic it is possible to control the alpha-wheel apparatus with simple steps by manipulating the voltages sent to the integrated circuit with either 0 V or 5 V logic in order to change the flow of current through the windings. A picture of the final circuit is shown in Figure 4.9.

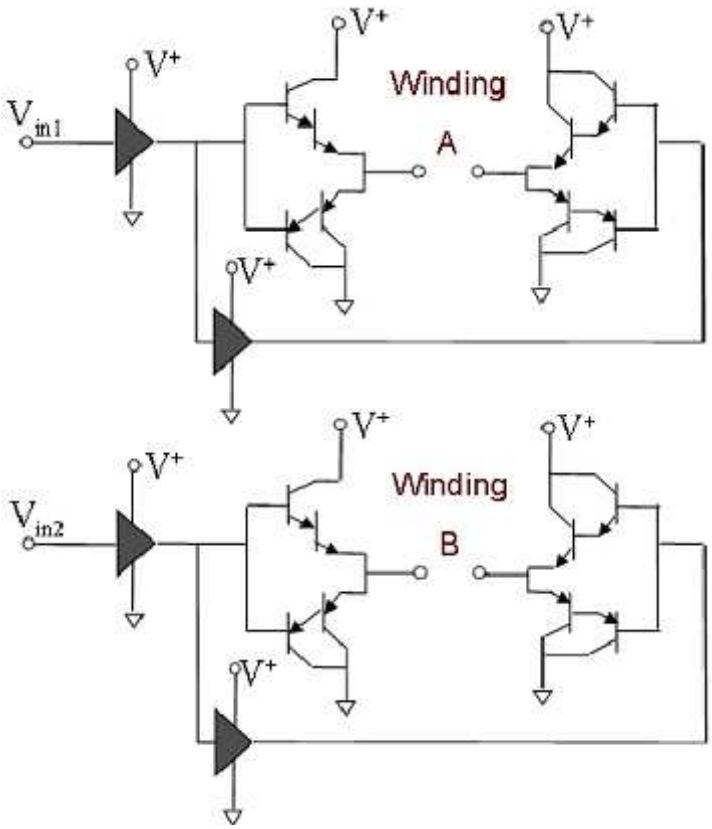

Figure 4.8 The circuit diagram of the stepper motor driver.

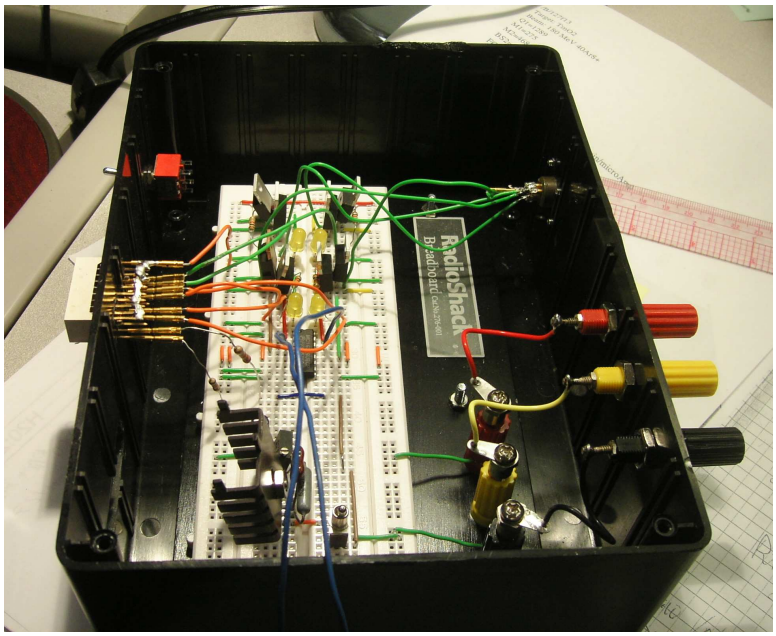

Figure 4.9 This shows the final constructed stepper motor circuit.

4.4 Peak Shaping From the Alpha-Detectors

The Data Acquisition System hardware and software used for the gas-phase experiments sampled voltages at a slow pace, approximately every 10 ms (discussed in more detail also in section 4.5). The pulse rise time which characterizes the signal from the detector is on the order of 10 ns. The 10 ns are due to two causes: The charge transit time, and the plasma time.

The charge transit time is simply the time it takes electron-hole pairs that are generated in the depletion zone to transverse it and be collected. The effects of the charge transit time can be minimized when a large electric field is generated across the detector and when the depletion zone is narrow. Since no electric field is ever homogenous, the migration of the electrons and holes will be a function of its location in the detector and the electric field it feels at that point.[65]

The plasma time is a characteristic for the types of radiation studied later in this chapter, which are alpha particles, but it also a significant component for any heavy charged particles. When an alpha-particles creates ion-pairs along its trajectory, it creates them at such a high density that the electron-hole pairs in the interior of the cloud are shielded from the effects of the electric field. Thus the electron-hole pairs on the outer edge will move first due to the electric field. As the edge ion-pairs migrate the shielding of the interior ions will decrease and they too will start migrating.[66,67,68,69,70,71,72,73] The time it takes for the plasma to be eroded is termed the plasma time, after which the detector behaves as a normal charge collection process. The amount of time associated with the plasma time is roughly 1-3 ns from the formation of tracks in the detector until the formation of the pulse rise time.[74,75,76,77,78,79]

When the charge is finally separated across the collector, the charge separation is not large enough to cause a significant voltage to measure. There needs to be an amplification step

before acquisition of data, this amplification is often termed the preamplification step. Preamplifiers are placed as near as possible to the detectors, to minimize the contribution of noise to the signal. A good preamplifier will not shape the pulse to any great extent, and its rise time should be as short as possible. Preamplifiers used in these experiments were the charge-sensitive (also known as charge-to-voltage) type. The basic concept of a preamplifier is shown in Figure 4.10

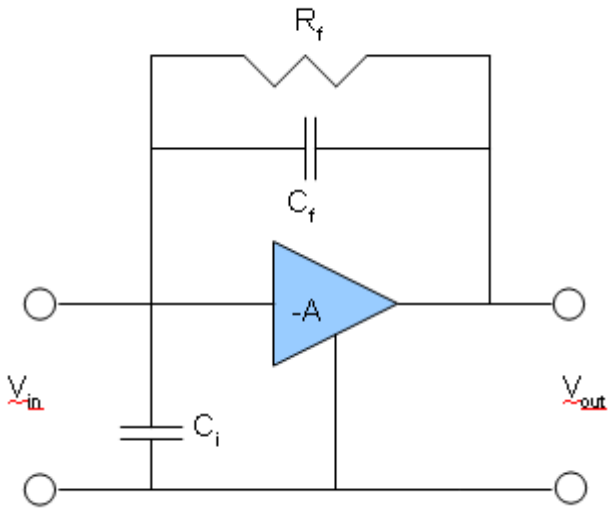

Figure 4.10 The elements of a charge-to-voltage preamplifier is shown.

The output voltage (V_{out}) is simply proportional to the amount of charge (Q) across the detector, that migrated to the applied voltage across it (V_{in}), the exact relationship is given in equation 4.11.[80]

$$V_{max} = \frac{Q}{C} \qquad (4.11)$$

R_f and C_f will control the decay rate or tail of the pulse out from the preamplifier, with a time constant that is equal to R_fC_f.

Once the voltage was generated with a preamplifier it is then necessary to construct a circuit that could hold that energy, collected from the alpha-detectors, long enough for the data acquisition system to measure it. The circuit constructed was a peak-and-hold circuit and utilized the integrated circuit PKD-01, the circuit made is shown in Figure 4.11.

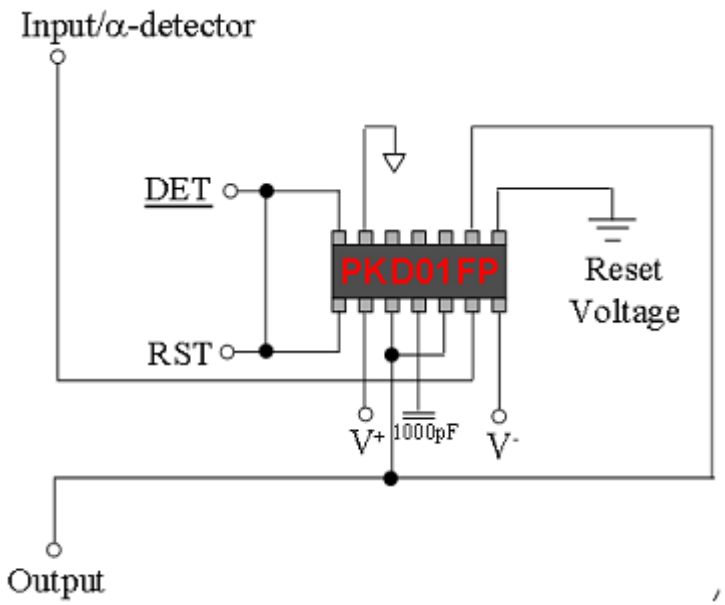

Figure 4.11 Circuit diagram of the shaping amplifier used for the gas-phase experiments.

The final shaping amplifier circuit board held three PKD-01s and was used to collect data from the alpha detectors on the alpha-wheel apparatus; a picture of the circuit is shown in Figure 4.12.

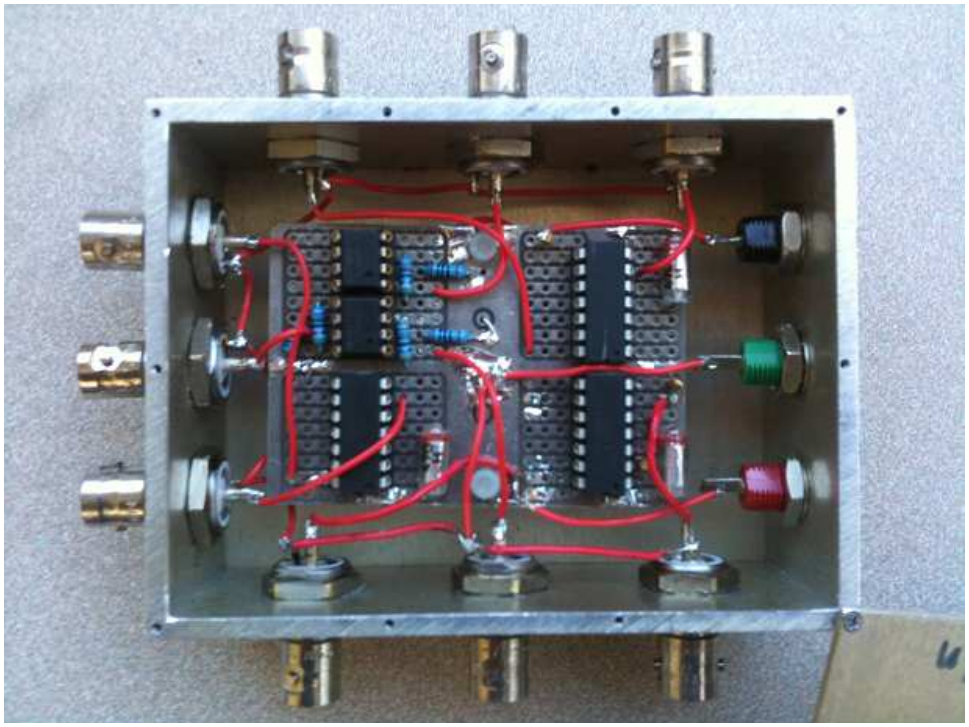

Figure 4.12 A picture of the 3-channel PKD-01 circuit designed, constructed, and used for gas-phase experiments.

The shaping amp could hold the maximum voltage collected from the alpha-detectors for several seconds until the signal began to droop. Since the DAQ was sampling every 10-15 ms it was more than enough for our needs. It should be noted that the number of events we were expecting were one event every minute in a low background, thus this type of setup would not be advisable for high-count rate experiments or high background experiments.

4.5 Data Acquisition System Hardware

Data are acquired using a modified National Instruments USB-6009, shown in Figure 4.13. The hardware was modified to accept digital signals from Bayonet Neill-Concelman (BNC) cables and to output digital signals using BNC cables. The USB-6009 stored information

in a its buffer and communicated the data being collected to the computer through a Universal Serial Bus (USB) connection.

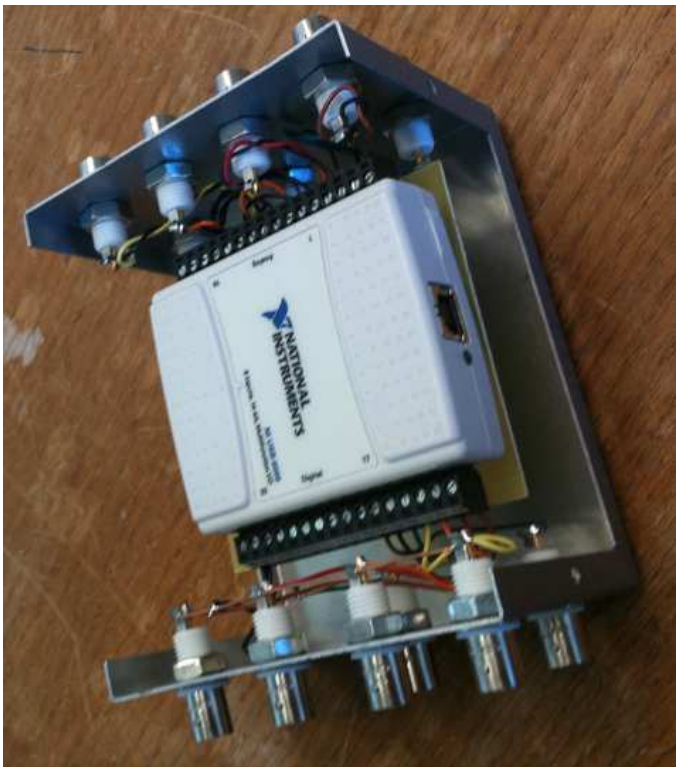

Figure 4.13 A picture of the modified NI USB-6009.

For any data acquisition system there will be a minimum time required so that when two separate events occur they are measured as two separate and distinct pulses. The minimum time is often called the dead time and for the NI USB-6009 it was sampled approximately every 15 ms. As radioactivity is a random statistical process, there is always a small probability that two nuclei will decay at nearly the same time, making separate observations of both separate decays nearly impossible; this can also happen in high-background experiments or any time a large count rate is expected. When a decay is lost in this fashion it is called a “dead time loss”.

There are two types of dead times, paralyzable and nonparalyzable response. In a detector

a fixed time τ follows any event until the detector becomes ready for more events, when this occurs the detector is termed *live*. In a paralyzable detector when the pulses come within τ, they will stack on top of each other and make the observed peak larger than what actually occurred. In a nonparalyzable response any events that come within τ will be ignored until the event has been sampled and the detector resets into a state where it can receive signals again. Both type of dead time responses are shown in Figure 4.14.

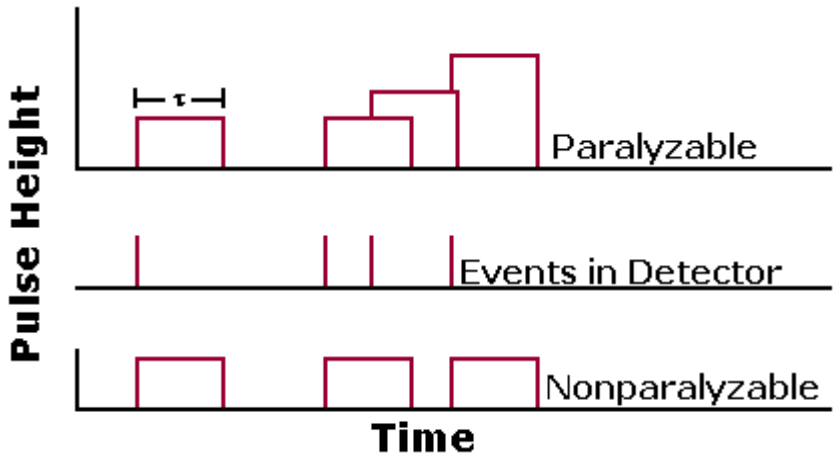

Figure 4.14 The measured pulse heights as seen for a fictitious series of events measured in the detector illustrating the concept of dead time.

In the figure there are 4 events at the detector, some within the dead time of the detector. For a paralyzable detector setup this would have meant only two events were observed, and the energy of the second would have been greater than that of any associated event. For a nonparalyzable decay the example would have led to 3 measured events, all corresponding to the energy of an event.

It is important to note that these models are idealized versions of how detectors respond during the dead-time. For the peak shaping circuit that I designed and explained in 4.3.1, the

response to a pulse in the dead time would have been nonparalyzable unless the second pulse was larger than the first, and in that case the peak intensity would have risen to that of the highest peak; which is typical for a peak-and-hold circuit.

There are expressions that describe the true rate of events (*n*) for both detector types if you recorded the count rate (*m*) and know the dead time (τ); see equation 4.8 for the nonparalyzable model and equation 4.9 for paralyzable model.

$$n = \frac{m}{1-m\tau} \tag{4.12}$$

$$m = ne^{-n\tau} \tag{4.13}$$

For the experiment presented later, the rate of events was low enough (5 per minute) that the dead time (~15 ms) was not a major concern.

4.6 Data Acquisition System Program (.exe)

The code for the data acquisition system program used for the Gas-Phase experiments (discussed later in this chapter) is shown in Appendix I. The code was written in Visual Basic (VB), because of its superior end user interface. Visual Basic is an event-driven programming language from Microsoft, meaning that with simple clicks and inputs a user has easy control over which part of the code that will be executed. The Visual basic language is considered one of the easier computer languages to learn, due to its origins in Beginner's All-purpose Symbolic Instruction Code (BASIC).

A nice feature of Visual Basic is the ability to make programs that are executable that

work across Windows platforms. So a program that is developed on one computer will work on any other windows computer without the need of installing any additional software; all that is needed is the one executable file. The first version of Visual Basic, v. 1.0, was released in 1991; the version used for this work is Visual Basic .Net which was released in 2008.

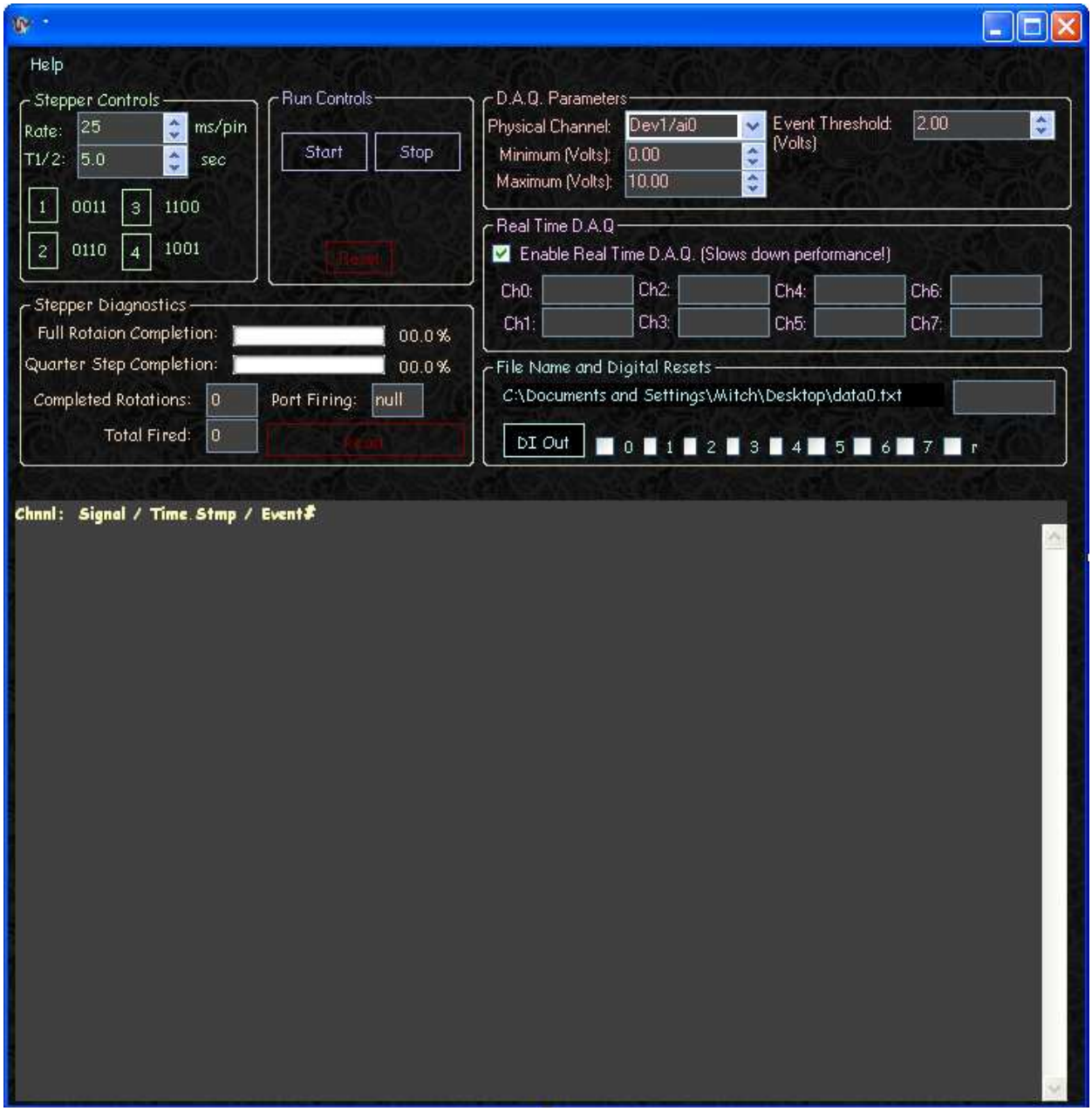

Figure 4.15 A screen shot of the data acquisition system used to acquire data.

The program is divided into different sections that have different functions, a different color is associated with the different sections. The sections are Stepper Controls (Green), Run Control (blue), D.A.Q. Parameters (Peach), Real Time D.A.Q. (Pink), Stepper Diagnostics (White), File Name and Digital Resets (cyan), Signal/Time.Stamp/Event# (Yellow).

The Stepper Controls section controls the rate of rotation of the circular sample holder across the alpha-detectors. The stepper motor makes 180 turns before it returns to its starting position. Because the alpha-wheel apparatus has four detectors, it needs to rotate 90 degrees and accomplishes this with 45 little steps. Once it reaches a detector, it stops rotating, collects data, and after a predetermined amount of time it begins to rotate in front of the next detector, and so on. The parameter listed as “Rate” is how long in milliseconds to wait until sending a signal to the motor to make one little step; the default setting is 25 ms. Thus it takes our device 45 x 25 ms, or 1.1 s, to turn and place the sample in front of the next detector. The length of time the sample is to be in front of a detector is determined by the “T1/2” parameter. The parameter is oddly named because the length of time we kept our samples in front of the detector was for the half-life of the element of interest. The buttons labeled “1”, “2”, “3”, “4” were used to manually turn the wheel if we so desired, but the typical use was for the program to use the default setting parameters of “Rate” and “T1/2”.

The Stepper Diagnostics section monitors the progress of the sample holder in the alpha-wheel apparatus. The “Full Rotation Completion” parameter keeps track of where the sample holder is in relation to where it began turning; the value is measured in percent from 0% to 100% and correlates with the sample wheel's rotation of 0 to 360 degrees. The “Quarter Step Completion” parameter keeps track of where the sample holder is when it begins to move the sample from one detector to the next detector; The value is measured in percent from 0 to 100% and correlates with the sample wheel's rotation of 0 – 90 degrees, or 90 – 180 degrees, or 180 –

270 degrees, or 270 – 360 degrees. The “Completed Rotation” parameter kept track of how many 360 degree turns the sample holder in the alpha-wheel apparatus made. The “Total Fired” parameter kept track of the total number of signals we sent to the stepper motor, each signal corresponded to a stepper motor step. The “port firing” parameter displays which pins of the parallel port is used to send the signal to the stepper-motor driver.

The Run Controls section deals with the actual running, and stopping of the program. Once the “start” button is pressed the stepper motor would begin to turn, using the predefined variables, and the program would begin to acquire the raw voltage signals from the DAQ device, see lines 1589 to 1604 in Appendix I for more information. Pressing “stop” would stop the motor from turning and stop collection of data from the DAQ device, see lines 1539-1545 in Appendix I for more information. The “Reset” button would return all editable parameters back to default.

The D.A.Q. Parameters section deals with the actual DAQ device. The device has eight analog inputs and the parameter labeled “Physical Channel” determines which one of the eight is sampled; the default is channel 0, and is shown as “Dev1/ai0”. In order to sample all channels the “Physical Pamater needs to be edited as so, “Dev1/ai0:7”. The “Minimum (Volts)” and “Maximum (Volts)” parameter tells the device from what range to expect signals, the device can accept any voltage between -10 V and 10 V. The design of our circuits insured only positive signals, thus the default “Minimum (Volts)” parameter is set to 0.00 V. The “Event Threshold (Volts)” parameter is the signal strength necessary in order for the program to consider an event real and record the data to a file.

The Real Time D.A.Q. Section is meant to allow the user to view the signals that the DAQ device's analog inputs is collecting in real time. It is only useful for debugging the device and was never used when collecting actual data as it was resource intensive and slowed down the computer considerably.

The File Name and Digital Resets section controlled where the data was saved and it could sent a signal to a peak-and-hold circuit to let go of the signal it collected. The location of the data file and the name of the data file could be edited by simply clicking the label displaying the default location (*i.e.* C:\Documents and Settings\Mitch\Desktop\data0.txt"). The design of the circuits prior to the DAQ device sampling involved a peak-and-hold circuit that held the voltage from the alpha-detector long enough for the DAQ device to sample it and send it to the DAQ program. Sampling was done every 10-15 ms on average. When the DAQ program detected an event above the "Event Threshold (Volts)" parameter it would record the event to file and then send a signal to the peak-and-hold to release that voltage. This process was visually seen by a check-mark that flashed on the screen corresponding to which channel had just been told to reset its peak-and-hold circuit. The "DI Out" button is for manually reseting the peak-and-hold and is used by check-marking the appropriate channel and then clicking the button.

The section with the label "Signal/Time.Stamp/Event#" is where the output of events the program believed to be real would appear in real-time. When the program comes across an event that is above the "Event Threshold (Volts)" parameter it will first save the data to file and then display it in this section of the program, see line1304-1310 in Appendix I for more information.

4.7 Data File Structure and Data Analysis Program

The data files stored 6 types of information: whether the heavy-ion beam was on target, the detector/channel an event occurred, the energy of that event, the unique time the event was detected in milliseconds, and a unique event number that was assigned, and at what time the sample holder started to make a 90 degree step, as shown in Figure 4.16.

```
beam-on      440331984
stepped      440331984
stepped      440337375
stepped      440342765
0:   7.02663959906771  440345203         2502
stepped      440348187
beam-on      440353578
stepped      440353578
stepped      440358968
stepped      440364359
stepped      440369750
2:   7.72490011947199  440374812         5415
```

Figure 4.16 An image of a typical data file.

After the run the files were placed onto FTP to allow other experimenters access to the files. A website was created to read the files and process it using a web browser. An image of the web browser interface is shown below Figure 4.17.

The files were processed using the language of php. By selecting for the parameters of interest above plots of counts versus alpha-energy were generated on the server and displayed back to the end user. The graphics program used was written by me and were generated pixel by pixel; the total cost in server time to run the script is less than 1s. The script also displays relevant information such as: the bin size, detector channel, largest peak, total events in the file, name of the data file analyzed, and associated image information, as shown in Figure 4.18.

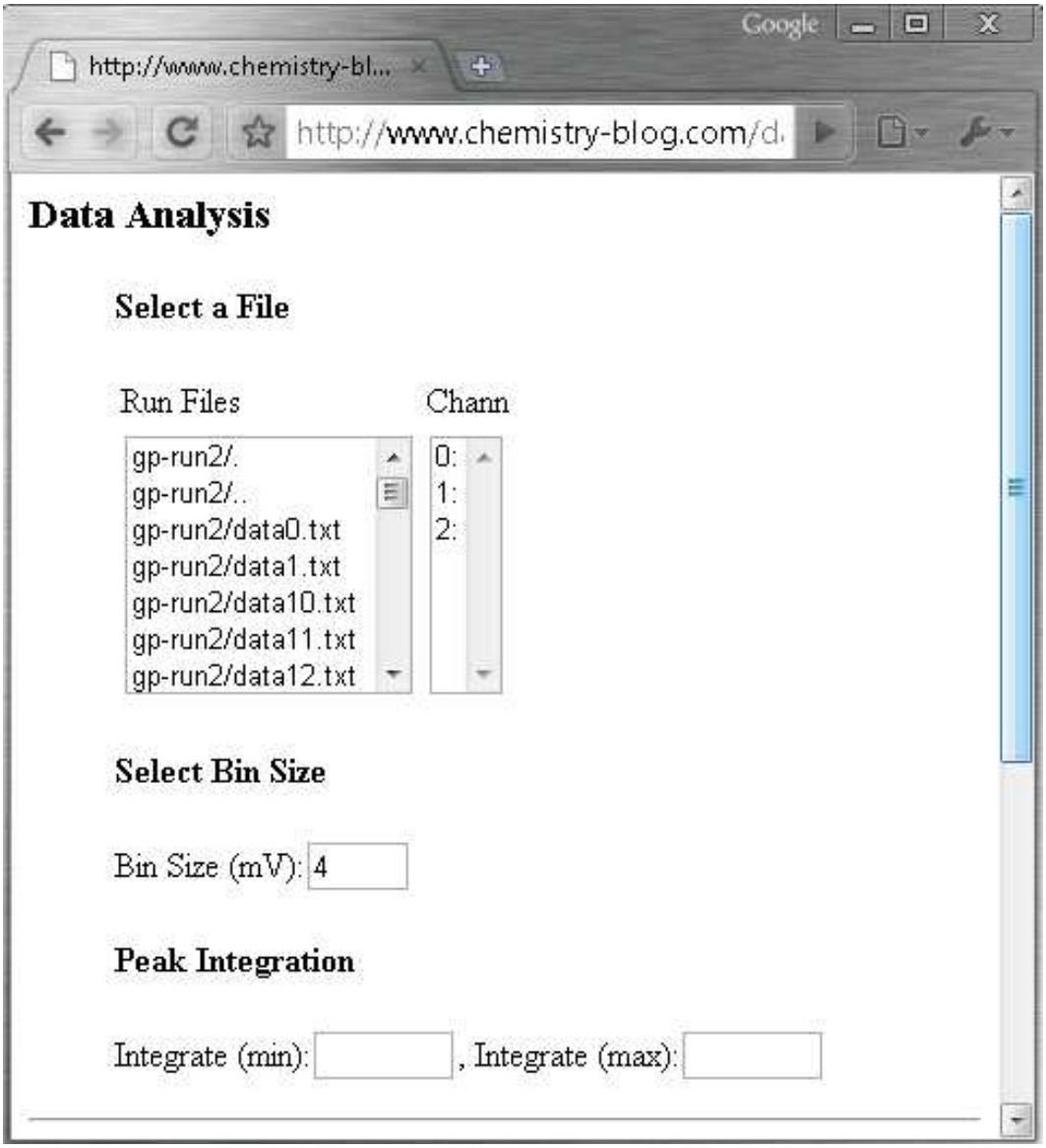

Figure 4.17 A screen shot of the web browser interface for analyzing data files.

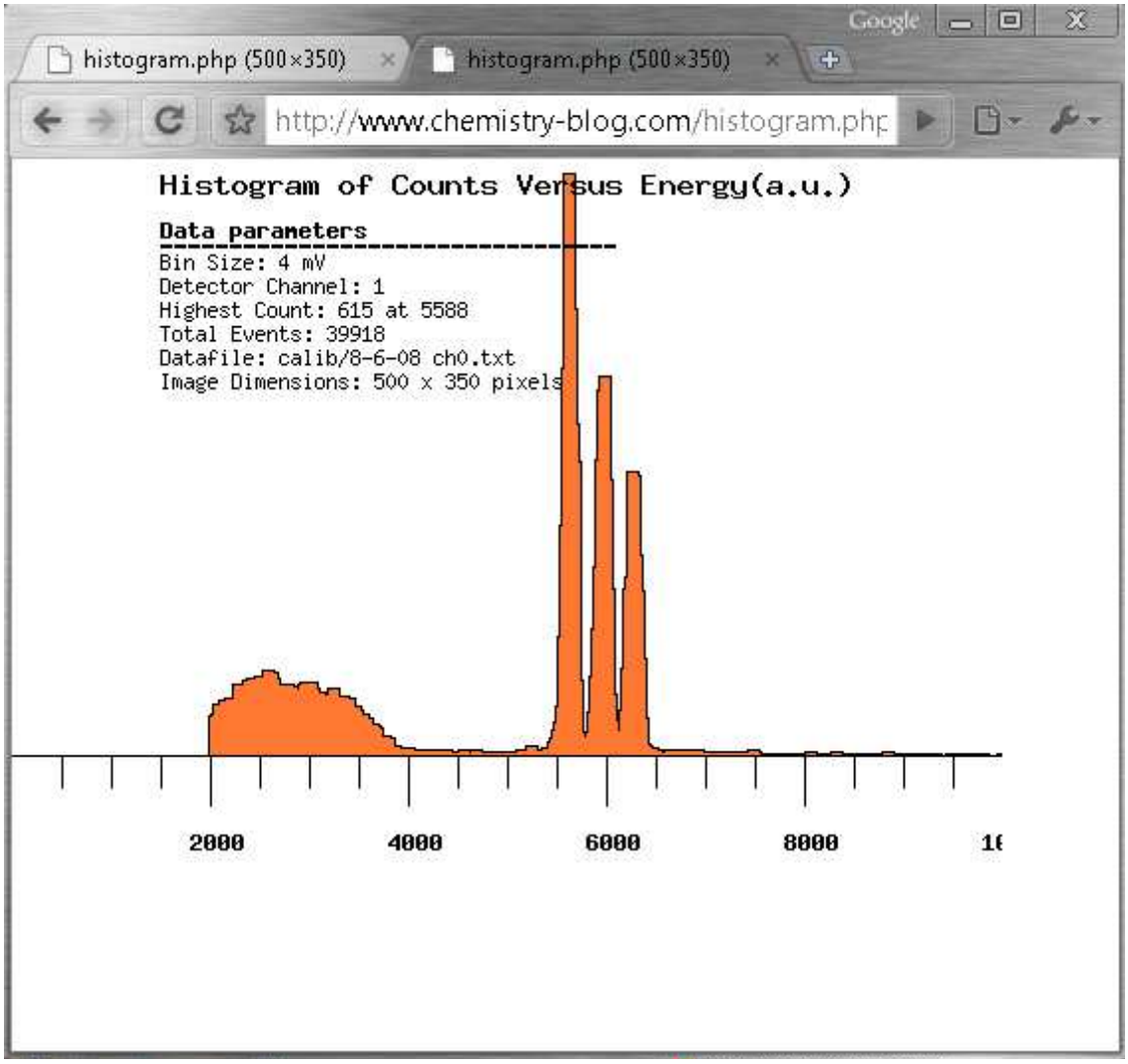

Figure 4.18 A screen shot of a typical spectrum generated from the data analysis program. The y-axis is counts and the x-axis is alpha-energy in (keV).

4.8 Experiments with the Alpha-Wheel Apparatus

At the 88-inch cyclotron, Fr-205 was produced to test the alpha-wheel apparatus. Francium-205 was used as a model for rutherfordium-261. Fr-205 has a half-life of 3.85 s and an alpha-energy of 7.01 MeV, while Rf-261 has a half-life of 75 s and an alpha-energy of 8.7 MeV.[81] The nuclear reaction was $^{169}Tm(^{40}Ar, 4n)^{205}Fr$. Once francium was made using the BGS it was

separated based on magnetic rigidity and stopped/thermalized in helium gas in the recoil transport chamber (RTC). The RTC then contained the francium atoms and an atmosphere of helium, into this chamber aerosol particles of potassium chloride were flushed in at a temperature of ~ 500 °C. The francium atoms would then attach to the aerosol particles and were transported from the BGS to the alpha-wheel apparatus via gas-jet. At the end of the gas-jet transport, the aerosols were deposited onto the thin Mylar film of the rotating sample holder and passed in front of the detectors as described earlier in this chapter.

The parameters of the experiment were as follows. The ^{40}Ar beam was turned on for 2 s and then kept off for 16 s. Thus with a step-time of 4 s the aerosol spot would travel in front of all the detectors and back to the collection spot, at which time the beam would be turned on again and the process would be repeated. At the 1st step the aerosol spot would be at detector-1 and the data recording would be collected. At the 2nd step the sample spot would be moved to detector-2 and data would be collected and so on. A typical spectrum collected from the experiment is shown in Figure 4.19

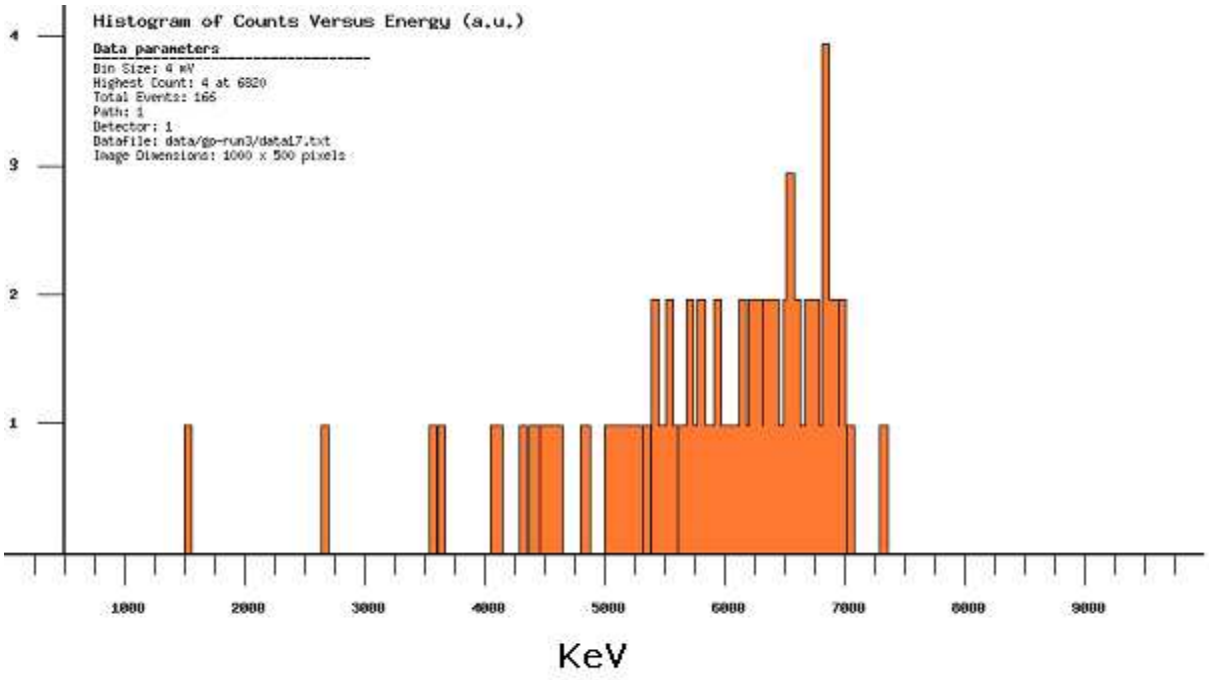

Figure 4.19 This shows the alpha spectrum collected from the $^{169}Tm(^{40}Ar, 4n)^{205}Fr$ nuclear reaction.

This histogram graphs the number of events recorded at specific alpha-energies. It uses a bin size of 4 mV corresponding 4 keV. As seen above, there is a peak at 6820 keV corresponding to Fr-205's 7.01 MeV alpha. The decrease in energy is likely due to the alpha-spectrum being taken in a flow of gas rather than in a vacuum. A table of all the events corresponding to Francium-205 is shown in Table 4.1.

Total Events	Detector 1	Detector 2	Detector 3
Group 1	129	21	35

Table 4.1 Shows a tabulation of all the events collected for the run.

A post run analysis on detector-2 found that channel to be unreliable and data collected from it was not incorporated in any subsequent analysis. The 129 events collected after turning the beam off and after waiting eight seconds corresponds to ~2 half-lives of Fr-205. By the time it reaches detector-3 approximately 4 half-lives have passed. As the number of events in detector-3 correspond roughly to ¼ of the number of events detected in detector-1, we have strong confidence that all our systems, aside from channel-2, were operating effectively.

4.9 Conclusion

From the $^{169}Tm(^{40}Ar, 4n)^{205}Fr$ nuclear reaction, we have shown that it is possible to transport radionuclides quickly and efficiently to the constructed alpha-wheel apparatus. We

have also shown that the electronics and circuits constructed are reasonably reliable and work well. These hardware and electronic systems may be on interest to future nuclear scientists as they were mostly constructed using common components found in local electronic stores; the electronics also utilized newer methods of communicating to computers like USB instead of complicated PC cards; finally the electronics were free of the nuclear instrumental module (NIM) standard and could often simply be plugged into a wall. The information learned from the gas-phase experiments will be useful for future investigations into the chemical properties of the heaviest elements in the gas-phase.

Chapter 5: Production of ^{261}Rf from ^{238}U(^{26}Mg, 3n)^{261}Rf

5.1 Introduction

The Berkeley Gas-Filled Separator (BGS) was used extensively for this work, and a summary of the device follows. The BGS utilizes three magnets to separate the heavy-ion beam from complete fusion evaporation products; the process has been described previously by Gregorich *et al.*.[82] A figure of the device is shown in Figure 5.1

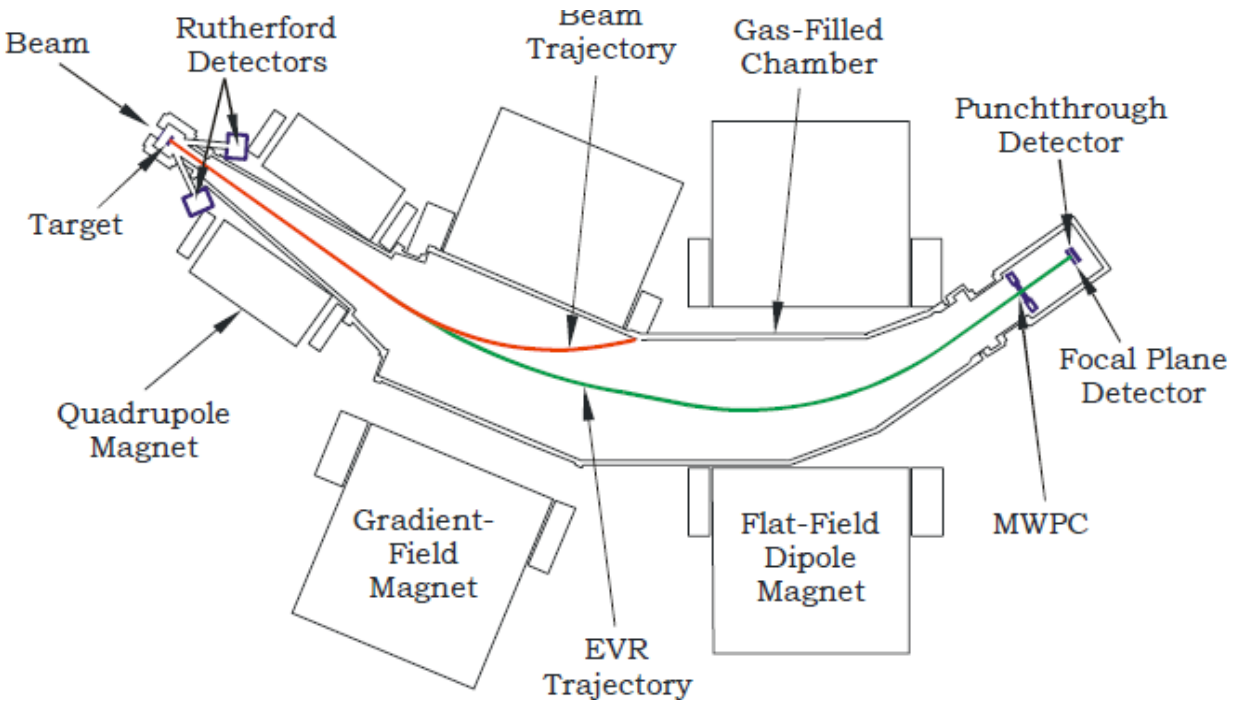

Figure 5.1 A schematic of the Berkeley Gas-Filled Separator in the configuration used for the work in this chapter.

The target section of the BGS housed a rotating wheel nine arc-shaped uranium(IV) tetrafluoride targets. The wheel usually rotated at 450 rpm. As a high intensity heavy-ion beam passes through a target it will deposit some of its energy as heat, as the wheel rotates it dissipates

a significant fraction of that heat across the other eight targets; this helps keep the uranium targets from melting. The uranium targets were deposited onto an aluminum backing. Details on the thicknesses of the uranium tetrafluoride targets and aluminum backings can be found in section 5.2.1

The Rutherford detectors are located after the target at an angle of ~27 degrees from the path of the beam (beam axis). The Rutherford detectors are silicon semiconductor detectors that have been thoroughly introduced in chapter 4. They are termed Rutherford detectors because they monitor the Rutherford scattering that occurs as the beam passes through the target. As the positive heavy-ions in the beam transverse the target they will often strike an atomic nucleus within the target and scatter as a simple 3-dimensional kinematics problem. The Rutherford detectors have four grids of wire mesh covering, this is to decrease the rate of scattered beam particles hitting the detectors; without the grids the scattered particles influx would be sufficient to cause excessive radiation damage to the detectors.

The rate of Rutherford scattering (R_{Ruth}) is a direct function of the cross-section of the scattering (σ_{Ruth}), the number of atoms in the target per unit area (N_t), and the intensity of the beam (I). Since the total rate of Rutherford scattering across all angles is not known (because we only have Rutherford detectors at one angle), it is necessary to use the differential Rutherford scattering model.[83]

$$\frac{d\sigma}{d\omega} = \left[\frac{e^2 Z_1 Z_2}{4\pi\epsilon_0 (2E_{lab})}\right] \frac{1}{\sin^4\theta} \frac{[\cos\theta + \sqrt{1-(m_1/m_2)^2 \sin^2\theta)}]^2}{\sqrt{1-(m_1/m_2)^2 \sin^2\theta}} \quad (5.1)$$

The term $d\omega$ is the solid angle differential, Z_1 and Z_2 are the atomic number of the projectile and target respectively, E_{lab} is the energy in the laboratory frame of the beam, m_1 and m_2 are the

masses of the projectile and target respectively, θ is the angle of the Rutherford detectors from the beam axis in the laboratory frame. Equation 5.1 only holds true when the mass of the projectile is less than the mass of the target.

As matter mainly consists of empty space the beam will mainly pass through the thin target material without any interaction. Compared to the rate of no reaction, the scattering rate is comparatively several magnitudes smaller, smaller yet is the magnitude of the rate of compound nucleus formation. When proper compound nucleus formation does occur, the evaporation of the proper number of neutrons without fission, it creates the isotope of interest (in the case of this chapter rutherfordium). The next task of the BGS is to separate the beam from the rutherfordium ions, this is accomplished through the use of the three BGS magnets. The magnets include a quadrupole magnet (Q1), followed by a gradient field dipole magnet (M1), and followed by a flat field dipole magnet (M2). The quadrupole magnet is oriented in such a way that it focuses the rutherfordium isotopes vertically while broadening out the horizontal distribution of ions. Rutherfordium ions get physically separated from the beam by the next magnets (M1 and M2). The magnets applies a different net Lorentz force on the beam and rutherfordium ions, the Lorentz force is given in equation 5.2.

$$F = qvB \tag{5.2}$$

The term q is the charge of the particle, v is the velocity of the particle, and B is the magnitude of the magnetic field. The Lorentz force is equally balanced by the centripetal force, given in equation 5.3.

$$F = \frac{mv^2}{\rho} \tag{5.3}$$

The term m is the particle's mass, v is the velocity of the particle, and ρ is the radius of curvature of the motion of the particle. As ρ is fixed by the physical design of the BGS, by careful selection of B it is possible to have high separation factors. The overall configuration of the BGS allows for a physical separation factor on the order of ~10^{14}, due in part to the large bending angle of the BGS 70°.

The charge state of the beam particles are known exactly, due to the cyclotron selecting for only one type of charge state. However, the charge state of rutherfordium as it leaves the target material is an other matter. The momentum of rutherfordium is much lower than the beam particles, so that interactions with the gas molecules within the BGS will cause rutherfordium to pick up several electrons. In order to account for this we use the concept of average charge state (q_{avg}). The average charge state is simply the average of the discrete charges on rutherfordium atoms as it transverses the BGS. An equation for the average charge state used in these experiments is given in equation 5.4.[84]

$$q_{avg} = px + r + s\sin\left(\tfrac{2\pi}{32}[Z - (px + r) - t]\right) \tag{5.4}$$

The equation is a semi-empirical fit to the measured q_{avg}. The term p is equal to 0.641, x is equal to $vZ^{1/3}$ where v is the velocity expressed in Bohr velocity and Z is the atomic number of the particle, r is -0.235, s is 0.517, t is 74.6.

The research group has developed equations to determine the amount of current needed to power each magnet in order to have good separation based on the magnetic rigidity of the

element of interest once the average charge state has been calculated.

$$I(Q1) = 765mv/q_{avg} \tag{5.5}$$

$$I(M1) = \frac{mv/q_{avg}}{0.002467+0.002016R} \tag{5.6}$$

$$I(M2) = \frac{mv/q_{avg}}{0.002467/R+0.002016} \tag{5.7}$$

The term *mv* is the momentum of the charged particle, and R is an experimentally determined constant and has been found to be 1.69. With all this information it is possible to separate out rutherfordium atoms and pass them to the detectors for detection.

The detectors used for the focal plane, at the end of the BGS, were of the silicon semiconductor variety. An picture of the detector array is shown in Figure 5.2

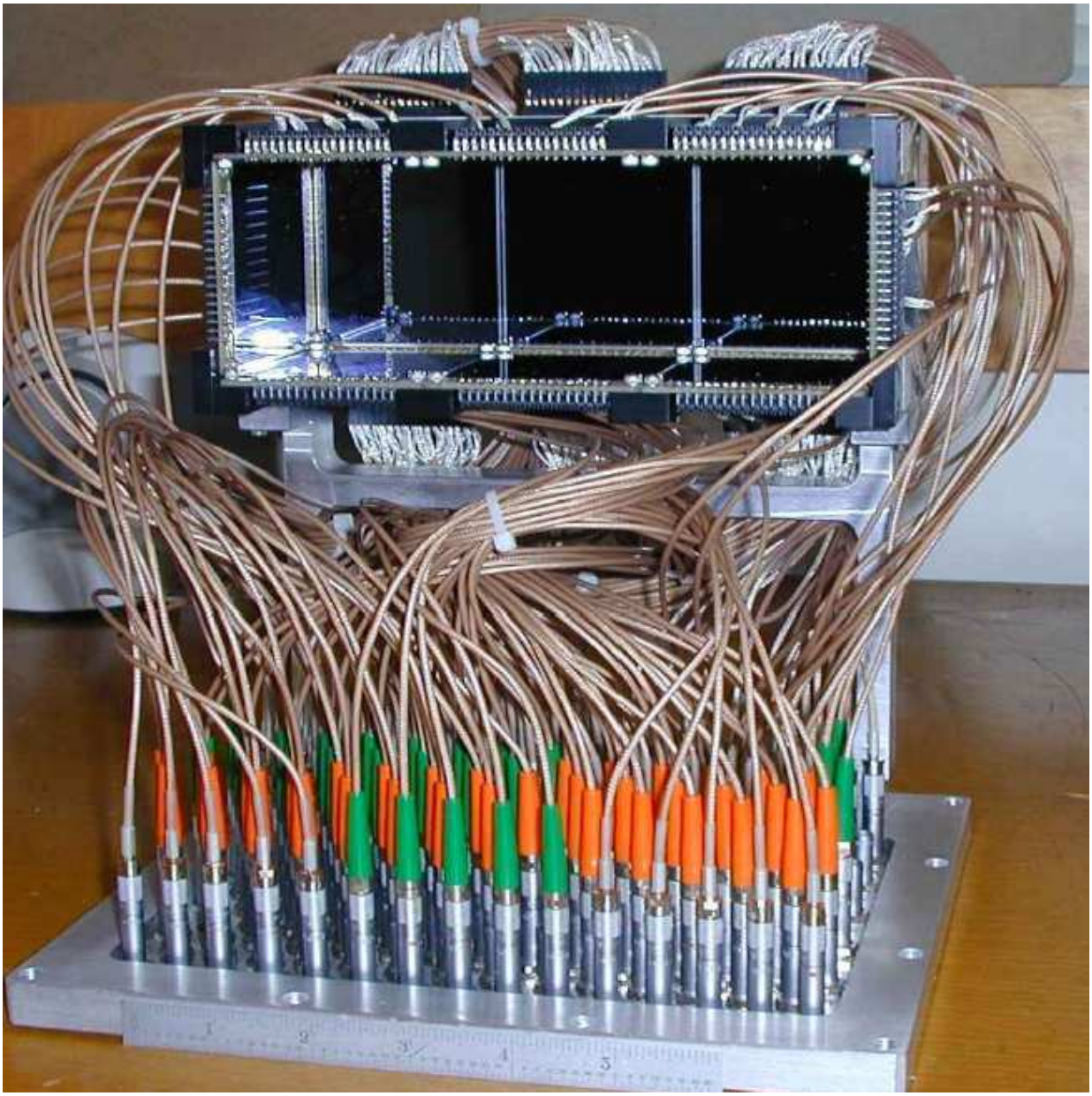

Figure 5.2 This shows a picture of the focal plane detector. The silicon semiconductor detectors can be seen to be arrayed in an open rectangular box.

The detectors are arrayed in an open rectangular box configuration of detector “cards”. The detector cards are 60 mm x 60 mm and each is divided equally into 16 strips. Each strip is 3.265 mm long and can be used to determine the position of a decay within the detector array. The

detectors are externally reverse biased to 40 V.

The data acquisition system differs from that explained in chapter 4. Similarly, as decays occur in the detector array the charge is converted to voltage by the use of preamplifiers. However, after the preamplifiers the signal is sent to a CAEN N568B amplifier. This module creates a fast pulse that is sent to a MSU 1806 constant fraction discriminator (CFD). In order for data collection to begin the CFD must receive a voltage that is above a minimal threshold; if such a signal exists it is sent to a trigger module.[85] The dead time for acquisition is 11 μs. A slow pulse is sent to the CAEN V785 analog digital converters (ADCs) for digitalization of the analog voltage. It is possible to view the data as it comes in real-time with the Multi Branch System (MBS) program.[86,87] The MBS is programmed to search for alpha-alpha correlations or alpha-spontaneous fission correlations under certain parameters described in the next section. When the MBS detects a possible correlation, it can shut off the beam in order to remove background noise caused by beam particles that were not effectiveley removed by the magnets. The MBS is installed on a RIO2 computer running the Lynx-OS operating system.

5.2 Experimental Setup

Beams of magnesium ($^{26}Mg^{6+}$) were produced from enriched metallic Mg in the Advanced Electron Cyclotron Resonance (AECR) ion source and then accelerated by the 88-Inch Cyclotron at Lawrence Berkeley National Laboratory(LBNL) to energies of 4.9–6.0 MeV/nucleon. The beam passed through a 45 μg/cm^2 thick carbon (C) window at the entrance of the Berkeley Gas-filled Separator (BGS)[88,89] that serves to separate the vacuum of the beam line from the 66-Pa helium(He) gas inside the BGS. Approximately one centimeter downstream of the entrance window was a rotating (~10 Hz)target wheel consisting of nine arc-shaped uranium(IV) tetrafluoride (UF_4, ~470μg/cm2) targets. These targets were prepared by

evaporation of UF_4 onto 580-μg/cm^2 aluminum(Al) foils. The energy thickness of the UF_4 layer on each target segment was approximately 2.0 MeV.

Typical beam intensities ranged from 0.5 to 1.0 particle-μA. Energy losses in carbon, aluminum, and uranium tetrafluoride were calculated with SRIM2006.02.[90] The product of target thickness and beam intensity was monitored on-line by the detection of Rutherford-scattered particles in two PIN diode detectors located at ±27° from the beam axis. Analysis of the pulse heights of the Rutherford-scattered projectiles from the various ^{26}Mg beam energies gives relative energies to within 0.1%. The resulting center-of-target beam energy was 121.8 MeV in the laboratory frame. Systematic uncertainty in the energies from the 88-Inch Cyclotron is estimated to be ~1%. Compound nucleus excitation energies were calculated using the relative beam energies with the experimental mass defects for ^{26}Mg and ^{238}U and the Thomas-Fermi mass defects for the compound nucleus.[91,92] The resulting ranges of compound nucleus excitation energy within the targets was 35.3 ± 0.9.

Rutherfordium compound nucleus evaporation residues(EVRs) were formed with the momentum of the projectile and recoil from the target. These EVRs were separated from the beam and other unwanted reaction products in the BGS based upon their differing magnetic rigidities in He gas. The magnetic rigidity of the rutherfordium EVRs was estimated as previously described.[93] The efficiency for collecting rutherfordium EVRs at the BGS focal plane was modeled using a Monte Carlo simulation of the EVR trajectories in the BGS, as described earlier,[94] and resulted in an efficiency of 15%. Details of the Si-strip detector array and data acquisition system have been reported elsewhere.[95] The rutherfordium EVRs have a short range and low implantation energy in the focal plane detector due to their low kinetic energy. Thus, a multiwire proportional counter could not be used and no information was available to help differentiate between implantation events and decay events of similar energy within the focal

plane detector.

The search for EVR-α and EVR-α-α correlations was expected to be hindered by random correlations of events unrelated to the decay of the desired nuclides. This problem was circumvented by using a pulsed beam that had a 50% duty factor with a period of 600 ms. Searching for α-particles between beam pulses led to a significant background reduction and allowed the identification of the α-emitting nuclides. To further reduce random correlations during the pulsed irradiations, α-decays were defined as events anticoincident with the upstream detectors (*i.e.*, the full energy of the α-decay was required to be in the focal plane detector and not split between the focal plane and upstream detectors). A fast shutoff mode was employed to search for ^{261}Rf. During this mode, the beam was shut off for 100 s after detection of an EVR [3.5 < E(MeV) < 10.0, during the beam pulse] followed by the observation of an α-particle [8.0 < E(MeV) < 9.0, in-between beam pulses] in the same strip and within 300 s and ±3.5 mm of the EVR. This allowed for the detection of the 25 s half-life of ^{257}No daughter in nearly background free conditions.[96]

5.3 Results

^{261}Rf was produced in the ^{26}Mg(^{238}U, 3n)^{261}Rf reaction at the lowest excitation energy of 35.3 MeV. The direct production of ^{261}Rf from a 3n hot-fusion exit-channel has not been observed previously. Two isomers are known for ^{261}Rf. The first isomer, designated ^{261a}Rf, decays by either SF or α-particle emission (Eα = 8.51 MeV, Iα =0.09, $T_{1/2}$ = 2.8 s).[97] The second isomer, designated ^{261b}Rf, has only been observed to decay by α-particle emission (Eα = 8.3 MeV, $T_{1/2}$ = 75 ± 7 s), with an upper limit of 11% for decay by SF.[98],[99]

We observed one correlated EVR-α-α decay chain with the following decay properties: ^{261}Rf (Eα = 8.34 ± 0.05 MeV, τ = 103.2 s), 257No (Eα = 8.30 ± 0.05 MeV, τ = 12.2 s). This event

was assigned to ^{261b}Rf and corresponds to a cross section of 28 pb. A second event consisted of an EVR-SF decay chain with a SF energy of 173.3 MeV and a life time of 9.4 s. This event could be assigned to either isomer of ^{261}Rf. However, as discussed in the following section, there is only a 6% probability for observing a random EVR-SF correlation within five half-lives of ^{261a}Rf. If confirmed, this would represent the first observation of this isomer produced as an EVR. The addition of this second event would only increase the 3n cross section by 14%, to 32 pb.

5.4 Conclusion

Historical precedence indicates that the 3n exit channel cross sections rapidly decrease with increasing $Z_{projectile}$.[100,101,102] Recently, Dvorak *et al.*[90] have made the surprising observation of a 3n exit channel in the ^{248}Cm(^{26}Mg, 3n)^{271}Hs reaction, with a cross section that is comparable in magnitude to the 4n cross section. In this work, a dedicated irradiation searching for the 3n exit channel of the ^{26}Mg + ^{238}U reaction was performed. ^{261}Rf was observed to have a cross section of 28 pb at a beam energy of 121.8 MeV. These discoveries open the possibility of accessing long-lived neutron-rich nuclides of the transactinides utilizing hot fusion reactions with neutron-rich targets.

Appendinx I – Visual Basic Code for Data Acquisition Sustem

```
Imports NationalInstruments.DAQmx
Imports System.Math

Public Class MainForm
    Inherits System.Windows.Forms.Form

    Private myTask As Task          'Main Task which is Assigned when the Start Button is Clicked
    Private myTask2 As Task          'Main Task which is Assigned when the Start Button is Clicked
    Private runningTask As Task     'Mitch added for the 100 sample catch
    Private analogInReader As AnalogMultiChannelReader 'Mitch added for the 100 sample catch
    Private analogCallback As AsyncCallback 'Mitch added for the 100 sample catch
    Private data100 As Double(,) 'Mitch added for the 100 sample catch
    Dim reader2 As AnalogMultiChannelReader 'Mitch added for the 100 sample catch
    Dim reader As AnalogMultiChannelReader
    Private totalSamples As Int32       'Global Container for the number samples to be acquired
    Private acquiredSamplesCount As Int32 = 0   'Iteration variable which hold the current sample being acquired

    Private dataColumn As DataColumn()
    Friend WithEvents runcontrols As System.Windows.Forms.GroupBox
    Friend WithEvents Label11 As System.Windows.Forms.Label
    Friend WithEvents halflife As System.Windows.Forms.NumericUpDown
    Friend WithEvents Label10 As System.Windows.Forms.Label
    Friend WithEvents reset As System.Windows.Forms.Button
    Friend WithEvents boxtimer As System.Windows.Forms.NumericUpDown
    Friend WithEvents Label2 As System.Windows.Forms.Label
    Friend WithEvents stopbutton2 As System.Windows.Forms.Button
    Friend WithEvents Label1 As System.Windows.Forms.Label
    Friend WithEvents GroupBox4 As System.Windows.Forms.GroupBox
    Friend WithEvents Label27 As System.Windows.Forms.Label
    Friend WithEvents Label28 As System.Windows.Forms.Label
    Friend WithEvents Label25 As System.Windows.Forms.Label
    Friend WithEvents Label26 As System.Windows.Forms.Label
    Friend WithEvents Label23 As System.Windows.Forms.Label
    Friend WithEvents Label24 As System.Windows.Forms.Label
    Friend WithEvents Label22 As System.Windows.Forms.Label
    Friend WithEvents Label21 As System.Windows.Forms.Label
    Friend WithEvents daq7 As System.Windows.Forms.TextBox
    Friend WithEvents daq6 As System.Windows.Forms.TextBox
    Friend WithEvents daq5 As System.Windows.Forms.TextBox
    Friend WithEvents daq4 As System.Windows.Forms.TextBox
    Friend WithEvents daq3 As System.Windows.Forms.TextBox
    Friend WithEvents daq2 As System.Windows.Forms.TextBox
    Friend WithEvents daq1 As System.Windows.Forms.TextBox
    Friend WithEvents daq0 As System.Windows.Forms.TextBox
    Friend WithEvents daqenable As System.Windows.Forms.CheckBox
    Friend WithEvents TextBox2 As System.Windows.Forms.TextBox
    Friend WithEvents TextBox1 As System.Windows.Forms.TextBox
    Friend WithEvents GroupBox3 As System.Windows.Forms.GroupBox
    Friend WithEvents Label17 As System.Windows.Forms.Label
    Friend WithEvents Labelbarstep As System.Windows.Forms.Label
    Friend WithEvents barstep As System.Windows.Forms.ProgressBar
    Friend WithEvents Label16 As System.Windows.Forms.Label
    Friend WithEvents Label15 As System.Windows.Forms.Label
    Friend WithEvents Labelbarfull As System.Windows.Forms.Label
```

```
53      Friend WithEvents boxcomfull As System.Windows.Forms.TextBox
54      Friend WithEvents Label14 As System.Windows.Forms.Label
55      Friend WithEvents Label13 As System.Windows.Forms.Label
56      Friend WithEvents barfull As System.Windows.Forms.ProgressBar
57      Friend WithEvents Button6 As System.Windows.Forms.Button
58      Friend WithEvents boxtfire As System.Windows.Forms.TextBox
59      Friend WithEvents Label9 As System.Windows.Forms.Label
60      Friend WithEvents boxport As System.Windows.Forms.TextBox
61      Friend WithEvents Label8 As System.Windows.Forms.Label
62      Friend WithEvents Label7 As System.Windows.Forms.Label
63      Friend WithEvents Label6 As System.Windows.Forms.Label
64      Friend WithEvents Label5 As System.Windows.Forms.Label
65      Friend WithEvents Label4 As System.Windows.Forms.Label
66      Friend WithEvents Button2 As System.Windows.Forms.Button
67      Friend WithEvents Button5 As System.Windows.Forms.Button
68      Friend WithEvents Button3 As System.Windows.Forms.Button
69      Friend WithEvents Button4 As System.Windows.Forms.Button
70      Friend WithEvents Label29 As System.Windows.Forms.Label
71      Friend WithEvents threshold As System.Windows.Forms.NumericUpDown
72      Friend WithEvents Timer1 As System.Windows.Forms.Timer
73      Friend WithEvents timerstep As System.Windows.Forms.Timer
74      Friend WithEvents Button1 As System.Windows.Forms.Button
75      Friend WithEvents MasterControls As System.Windows.Forms.GroupBox
76      Friend WithEvents MenuStrip1 As System.Windows.Forms.MenuStrip
77      Friend WithEvents HelpToolStripMenuItem As System.Windows.Forms.ToolStripMenuItem
78      Friend WithEvents samples100 As System.Windows.Forms.Timer
79      Friend WithEvents bit0CheckBox As System.Windows.Forms.CheckBox
80      Friend WithEvents bit1CheckBox As System.Windows.Forms.CheckBox
81      Friend WithEvents bit2CheckBox As System.Windows.Forms.CheckBox
82      Friend WithEvents bit3CheckBox As System.Windows.Forms.CheckBox
83      Friend WithEvents bit4CheckBox As System.Windows.Forms.CheckBox
84      Friend WithEvents bit5CheckBox As System.Windows.Forms.CheckBox
85      Friend WithEvents bit6CheckBox As System.Windows.Forms.CheckBox
86      Friend WithEvents bit7CheckBox As System.Windows.Forms.CheckBox
87      Friend WithEvents writeButton As System.Windows.Forms.Button
88      Friend WithEvents rstckbx As System.Windows.Forms.CheckBox
89      Friend WithEvents GroupBox1 As System.Windows.Forms.GroupBox
90      Friend WithEvents tbfilename As System.Windows.Forms.TextBox
91      Friend WithEvents beam As System.Windows.Forms.TextBox
92      'Channels of Data
93      Private dataTable As DataTable = New DataTable 'Table to Display Data


94  #Region " Windows Form Designer generated code "

95      Public Sub New()
96          MyBase.New()
97          Application.EnableVisualStyles()

98          'This call is required by the Windows Form Designer.
99          InitializeComponent()

100         'Add any initialization after the InitializeComponent() call
101         physicalChannelComboBox.Items.AddRange(DaqSystem.Local.GetPhysicalChannels(PhysicalChannelTypes.
102 AI, PhysicalChannelAccess.External))
103         If (physicalChannelComboBox.Items.Count > 0) Then
```

```
104             physicalChannelComboBox.SelectedIndex = 0
105         End If

106     End Sub

107     'Form overrides dispose to clean up the component list.
108     Protected Overloads Overrides Sub Dispose(ByVal disposing As Boolean)
109         If disposing Then
110             If Not (components Is Nothing) Then
111                 components.Dispose()
112             End If
113         End If
114         MyBase.Dispose(disposing)
115     End Sub

116     'Required by the Windows Form Designer
117     Private components As System.ComponentModel.IContainer

118     'NOTE: The following procedure is required by the Windows Form Designer
119     'It can be modified using the Windows Form Designer.
120     'Do not modify it using the code editor.
121     Friend WithEvents channelParametersGroupBox As System.Windows.Forms.GroupBox
122     Friend WithEvents maximumLabel As System.Windows.Forms.Label
123     Friend WithEvents minimumLabel As System.Windows.Forms.Label
124     Friend WithEvents physicalChannelLabel As System.Windows.Forms.Label
125     Friend WithEvents loopTimer As System.Windows.Forms.Timer
126     Friend WithEvents minimumValueNumeric As System.Windows.Forms.NumericUpDown
127     Friend WithEvents maximumValueNumeric As System.Windows.Forms.NumericUpDown
128     Friend WithEvents physicalChannelComboBox As System.Windows.Forms.ComboBox
129     <System.Diagnostics.DebuggerStepThrough()> Private Sub InitializeComponent()
130         Me.components = New System.ComponentModel.Container
131         Dim resources As System.ComponentModel.ComponentResourceManager = New
132 System.ComponentModel.ComponentResourceManager(GetType(MainForm))
133         Me.channelParametersGroupBox = New System.Windows.Forms.GroupBox
134         Me.Label29 = New System.Windows.Forms.Label
135         Me.threshold = New System.Windows.Forms.NumericUpDown
136         Me.physicalChannelComboBox = New System.Windows.Forms.ComboBox
137         Me.minimumValueNumeric = New System.Windows.Forms.NumericUpDown
138         Me.maximumValueNumeric = New System.Windows.Forms.NumericUpDown
139         Me.maximumLabel = New System.Windows.Forms.Label
140         Me.minimumLabel = New System.Windows.Forms.Label
141         Me.physicalChannelLabel = New System.Windows.Forms.Label
142         Me.loopTimer = New System.Windows.Forms.Timer(Me.components)
143         Me.runcontrols = New System.Windows.Forms.GroupBox
144         Me.Label11 = New System.Windows.Forms.Label
145         Me.Label7 = New System.Windows.Forms.Label
146         Me.halflife = New System.Windows.Forms.NumericUpDown
147         Me.Button5 = New System.Windows.Forms.Button
148         Me.Label6 = New System.Windows.Forms.Label
149         Me.Label10 = New System.Windows.Forms.Label
150         Me.Label5 = New System.Windows.Forms.Label
151         Me.Button4 = New System.Windows.Forms.Button
152         Me.boxtimer = New System.Windows.Forms.NumericUpDown
153         Me.Label4 = New System.Windows.Forms.Label
154         Me.Button3 = New System.Windows.Forms.Button
155         Me.Label2 = New System.Windows.Forms.Label
```

```
156         Me.Button2 = New System.Windows.Forms.Button
157         Me.Label1 = New System.Windows.Forms.Label
158         Me.reset = New System.Windows.Forms.Button
159         Me.stopbutton2 = New System.Windows.Forms.Button
160         Me.GroupBox4 = New System.Windows.Forms.GroupBox
161         Me.Label27 = New System.Windows.Forms.Label
162         Me.Label28 = New System.Windows.Forms.Label
163         Me.Label25 = New System.Windows.Forms.Label
164         Me.Label26 = New System.Windows.Forms.Label
165         Me.Label23 = New System.Windows.Forms.Label
166         Me.Label24 = New System.Windows.Forms.Label
167         Me.Label22 = New System.Windows.Forms.Label
168         Me.Label21 = New System.Windows.Forms.Label
169         Me.daq7 = New System.Windows.Forms.TextBox
170         Me.daq6 = New System.Windows.Forms.TextBox
171         Me.daq5 = New System.Windows.Forms.TextBox
172         Me.daq4 = New System.Windows.Forms.TextBox
173         Me.daq3 = New System.Windows.Forms.TextBox
174         Me.daq2 = New System.Windows.Forms.TextBox
175         Me.daq1 = New System.Windows.Forms.TextBox
176         Me.daq0 = New System.Windows.Forms.TextBox
177         Me.daqenable = New System.Windows.Forms.CheckBox
178         Me.TextBox2 = New System.Windows.Forms.TextBox
179         Me.TextBox1 = New System.Windows.Forms.TextBox
180         Me.GroupBox3 = New System.Windows.Forms.GroupBox
181         Me.Label17 = New System.Windows.Forms.Label
182         Me.Labelbarstep = New System.Windows.Forms.Label
183         Me.barstep = New System.Windows.Forms.ProgressBar
184         Me.Label16 = New System.Windows.Forms.Label
185         Me.Label15 = New System.Windows.Forms.Label
186         Me.Labelbarfull = New System.Windows.Forms.Label
187         Me.boxcomfull = New System.Windows.Forms.TextBox
188         Me.Label14 = New System.Windows.Forms.Label
189         Me.Label13 = New System.Windows.Forms.Label
190         Me.barfull = New System.Windows.Forms.ProgressBar
191         Me.Button6 = New System.Windows.Forms.Button
192         Me.boxtfire = New System.Windows.Forms.TextBox
193         Me.Label9 = New System.Windows.Forms.Label
194         Me.boxport = New System.Windows.Forms.TextBox
195         Me.Label8 = New System.Windows.Forms.Label
196         Me.Timer1 = New System.Windows.Forms.Timer(Me.components)
197         Me.timerstep = New System.Windows.Forms.Timer(Me.components)
198         Me.Button1 = New System.Windows.Forms.Button
199         Me.MasterControls = New System.Windows.Forms.GroupBox
200         Me.MenuStrip1 = New System.Windows.Forms.MenuStrip
201         Me.HelpToolStripMenuItem = New System.Windows.Forms.ToolStripMenuItem
202         Me.samples100 = New System.Windows.Forms.Timer(Me.components)
203         Me.bit0CheckBox = New System.Windows.Forms.CheckBox
204         Me.bit1CheckBox = New System.Windows.Forms.CheckBox
205         Me.bit2CheckBox = New System.Windows.Forms.CheckBox
206         Me.bit3CheckBox = New System.Windows.Forms.CheckBox
207         Me.bit4CheckBox = New System.Windows.Forms.CheckBox
208         Me.bit5CheckBox = New System.Windows.Forms.CheckBox
209         Me.bit6CheckBox = New System.Windows.Forms.CheckBox
210         Me.bit7CheckBox = New System.Windows.Forms.CheckBox
211         Me.writeButton = New System.Windows.Forms.Button
```

```
        Me.rstckbx = New System.Windows.Forms.CheckBox
        Me.GroupBox1 = New System.Windows.Forms.GroupBox
        Me.tbfilename = New System.Windows.Forms.TextBox
        Me.beam = New System.Windows.Forms.TextBox
        Me.channelParametersGroupBox.SuspendLayout()
        CType(Me.threshold, System.ComponentModel.ISupportInitialize).BeginInit()
        CType(Me.minimumValueNumeric, System.ComponentModel.ISupportInitialize).BeginInit()
        CType(Me.maximumValueNumeric, System.ComponentModel.ISupportInitialize).BeginInit()
        Me.runcontrols.SuspendLayout()
        CType(Me.halflife, System.ComponentModel.ISupportInitialize).BeginInit()
        CType(Me.boxtimer, System.ComponentModel.ISupportInitialize).BeginInit()
        Me.GroupBox4.SuspendLayout()
        Me.GroupBox3.SuspendLayout()
        Me.MasterControls.SuspendLayout()
        Me.MenuStrip1.SuspendLayout()
        Me.GroupBox1.SuspendLayout()
        Me.SuspendLayout()
        '
        'channelParametersGroupBox
        '
        Me.channelParametersGroupBox.BackColor = System.Drawing.Color.Transparent
        Me.channelParametersGroupBox.Controls.Add(Me.Label29)
        Me.channelParametersGroupBox.Controls.Add(Me.threshold)
        Me.channelParametersGroupBox.Controls.Add(Me.physicalChannelComboBox)
        Me.channelParametersGroupBox.Controls.Add(Me.minimumValueNumeric)
        Me.channelParametersGroupBox.Controls.Add(Me.maximumValueNumeric)
        Me.channelParametersGroupBox.Controls.Add(Me.maximumLabel)
        Me.channelParametersGroupBox.Controls.Add(Me.minimumLabel)
        Me.channelParametersGroupBox.Controls.Add(Me.physicalChannelLabel)
        Me.channelParametersGroupBox.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte),
Integer), CType(CType(192, Byte), Integer), CType(CType(192, Byte), Integer))
        Me.channelParametersGroupBox.Location = New System.Drawing.Point(321, 28)
        Me.channelParametersGroupBox.Name = "channelParametersGroupBox"
        Me.channelParametersGroupBox.Size = New System.Drawing.Size(398, 82)
        Me.channelParametersGroupBox.TabIndex = 2
        Me.channelParametersGroupBox.TabStop = False
        Me.channelParametersGroupBox.Text = "D.A.Q. Parameters"
        '
        'Label29
        '
        Me.Label29.AutoSize = True
        Me.Label29.Location = New System.Drawing.Point(193, 17)
        Me.Label29.Name = "Label29"
        Me.Label29.Size = New System.Drawing.Size(88, 26)
        Me.Label29.TabIndex = 18
        Me.Label29.Text = "Event Threshold:" & Global.Microsoft.VisualBasic.ChrW(13) &
Global.Microsoft.VisualBasic.ChrW(10) & "(Volts)"
        '
        'threshold
        '
        Me.threshold.BackColor = System.Drawing.Color.FromArgb(CType(CType(64, Byte), Integer),
CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
        Me.threshold.DecimalPlaces = 2
        Me.threshold.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte), Integer),
CType(CType(192, Byte), Integer), CType(CType(192, Byte), Integer))
        Me.threshold.Increment = New Decimal(New Integer() {1, 0, 0, 65536})
```

```
        Me.threshold.Location = New System.Drawing.Point(291, 15)
        Me.threshold.Maximum = New Decimal(New Integer() {10, 0, 0, 0})
        Me.threshold.Name = "threshold"
        Me.threshold.Size = New System.Drawing.Size(96, 20)
        Me.threshold.TabIndex = 17
        Me.threshold.Value = New Decimal(New Integer() {200, 0, 0, 131072})
        '
        'physicalChannelComboBox
        '
        Me.physicalChannelComboBox.BackColor = System.Drawing.Color.FromArgb(CType(CType(64, Byte),
Integer), CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
        Me.physicalChannelComboBox.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte),
Integer), CType(CType(192, Byte), Integer), CType(CType(192, Byte), Integer))
        Me.physicalChannelComboBox.Location = New System.Drawing.Point(96, 15)
        Me.physicalChannelComboBox.Name = "physicalChannelComboBox"
        Me.physicalChannelComboBox.Size = New System.Drawing.Size(96, 21)
        Me.physicalChannelComboBox.TabIndex = 1
        Me.physicalChannelComboBox.Text = "Dev1/ai0"
        '
        'minimumValueNumeric
        '
        Me.minimumValueNumeric.BackColor = System.Drawing.Color.FromArgb(CType(CType(64, Byte), Integer),
CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
        Me.minimumValueNumeric.DecimalPlaces = 2
        Me.minimumValueNumeric.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte), Integer),
CType(CType(192, Byte), Integer), CType(CType(192, Byte), Integer))
        Me.minimumValueNumeric.Location = New System.Drawing.Point(96, 36)
        Me.minimumValueNumeric.Maximum = New Decimal(New Integer() {10, 0, 0, 0})
        Me.minimumValueNumeric.Minimum = New Decimal(New Integer() {10, 0, 0, -2147483648})
        Me.minimumValueNumeric.Name = "minimumValueNumeric"
        Me.minimumValueNumeric.Size = New System.Drawing.Size(96, 20)
        Me.minimumValueNumeric.TabIndex = 3
        '
        'maximumValueNumeric
        '
        Me.maximumValueNumeric.BackColor = System.Drawing.Color.FromArgb(CType(CType(64, Byte), Integer),
CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
        Me.maximumValueNumeric.DecimalPlaces = 2
        Me.maximumValueNumeric.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte),
Integer), CType(CType(192, Byte), Integer), CType(CType(192, Byte), Integer))
        Me.maximumValueNumeric.Location = New System.Drawing.Point(96, 54)
        Me.maximumValueNumeric.Maximum = New Decimal(New Integer() {10, 0, 0, 0})
        Me.maximumValueNumeric.Minimum = New Decimal(New Integer() {10, 0, 0, -2147483648})
        Me.maximumValueNumeric.Name = "maximumValueNumeric"
        Me.maximumValueNumeric.Size = New System.Drawing.Size(96, 20)
        Me.maximumValueNumeric.TabIndex = 5
        Me.maximumValueNumeric.Value = New Decimal(New Integer() {100, 0, 0, 65536})
        '
        'maximumLabel
        '
        Me.maximumLabel.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte), Integer),
CType(CType(192, Byte), Integer), CType(CType(192, Byte), Integer))
        Me.maximumLabel.Location = New System.Drawing.Point(6, 56)
        Me.maximumLabel.Name = "maximumLabel"
        Me.maximumLabel.Size = New System.Drawing.Size(96, 16)
        Me.maximumLabel.TabIndex = 4
```

```
        Me.maximumLabel.Text = "Maximum (Volts):"
        '
        'minimumLabel
        '
        Me.minimumLabel.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte), Integer),
CType(CType(192, Byte), Integer), CType(CType(192, Byte), Integer))
        Me.minimumLabel.Location = New System.Drawing.Point(8, 38)
        Me.minimumLabel.Name = "minimumLabel"
        Me.minimumLabel.Size = New System.Drawing.Size(96, 16)
        Me.minimumLabel.TabIndex = 2
        Me.minimumLabel.Text = "Minimum (Volts):"
        '
        'physicalChannelLabel
        '
        Me.physicalChannelLabel.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte), Integer),
CType(CType(192, Byte), Integer), CType(CType(192, Byte), Integer))
        Me.physicalChannelLabel.Location = New System.Drawing.Point(0, 19)
        Me.physicalChannelLabel.Name = "physicalChannelLabel"
        Me.physicalChannelLabel.Size = New System.Drawing.Size(96, 16)
        Me.physicalChannelLabel.TabIndex = 0
        Me.physicalChannelLabel.Text = "Physical Channel:"
        '
        'loopTimer
        '
        Me.loopTimer.Interval = 1
        '
        'runcontrols
        '
        Me.runcontrols.BackColor = System.Drawing.Color.Transparent
        Me.runcontrols.Controls.Add(Me.Label11)
        Me.runcontrols.Controls.Add(Me.Label7)
        Me.runcontrols.Controls.Add(Me.halflife)
        Me.runcontrols.Controls.Add(Me.Button5)
        Me.runcontrols.Controls.Add(Me.Label6)
        Me.runcontrols.Controls.Add(Me.Label10)
        Me.runcontrols.Controls.Add(Me.Label5)
        Me.runcontrols.Controls.Add(Me.Button4)
        Me.runcontrols.Controls.Add(Me.boxtimer)
        Me.runcontrols.Controls.Add(Me.Label4)
        Me.runcontrols.Controls.Add(Me.Button3)
        Me.runcontrols.Controls.Add(Me.Label2)
        Me.runcontrols.Controls.Add(Me.Button2)
        Me.runcontrols.Controls.Add(Me.Label1)
        Me.runcontrols.Font = New System.Drawing.Font("Comic Sans MS", 8.25!,
System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.runcontrols.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer),
CType(CType(255, Byte), Integer), CType(CType(192, Byte), Integer))
        Me.runcontrols.Location = New System.Drawing.Point(8, 28)
        Me.runcontrols.Name = "runcontrols"
        Me.runcontrols.Size = New System.Drawing.Size(161, 131)
        Me.runcontrols.TabIndex = 16
        Me.runcontrols.TabStop = False
        Me.runcontrols.Text = "Stepper Controls"
        '
        'Label11
        '
```

```
        Me.Label11.AutoSize = True
        Me.Label11.Location = New System.Drawing.Point(120, 40)
        Me.Label11.Name = "Label11"
        Me.Label11.Size = New System.Drawing.Size(24, 15)
        Me.Label11.TabIndex = 17
        Me.Label11.Text = "sec"
        '
        'Label7
        '
        Me.Label7.AutoSize = True
        Me.Label7.Location = New System.Drawing.Point(95, 100)
        Me.Label7.Name = "Label7"
        Me.Label7.Size = New System.Drawing.Size(31, 15)
        Me.Label7.TabIndex = 23
        Me.Label7.Text = "1001"
        '
        'halflife
        '
        Me.halflife.BackColor = System.Drawing.Color.FromArgb(CType(CType(64, Byte), Integer),
CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
        Me.halflife.DecimalPlaces = 1
        Me.halflife.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer),
CType(CType(255, Byte), Integer), CType(CType(192, Byte), Integer))
        Me.halflife.Increment = New Decimal(New Integer() {10, 0, 0, 0})
        Me.halflife.Location = New System.Drawing.Point(39, 38)
        Me.halflife.Maximum = New Decimal(New Integer() {10000, 0, 0, 0})
        Me.halflife.Minimum = New Decimal(New Integer() {1, 0, 0, 0})
        Me.halflife.Name = "halflife"
        Me.halflife.Size = New System.Drawing.Size(75, 23)
        Me.halflife.TabIndex = 16
        Me.halflife.Value = New Decimal(New Integer() {5, 0, 0, 0})
        '
        'Button5
        '
        Me.Button5.BackColor = System.Drawing.Color.Transparent
        Me.Button5.FlatAppearance.MouseDownBackColor = System.Drawing.Color.FromArgb(CType(CType(64,
Byte), Integer), CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
        Me.Button5.FlatAppearance.MouseOverBackColor = System.Drawing.Color.FromArgb(CType(CType(64,
Byte), Integer), CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
        Me.Button5.FlatStyle = System.Windows.Forms.FlatStyle.Flat
        Me.Button5.Location = New System.Drawing.Point(69, 98)
        Me.Button5.Name = "Button5"
        Me.Button5.Size = New System.Drawing.Size(20, 25)
        Me.Button5.TabIndex = 19
        Me.Button5.Text = "4"
        Me.Button5.UseVisualStyleBackColor = False
        '
        'Label6
        '
        Me.Label6.AutoSize = True
        Me.Label6.Location = New System.Drawing.Point(95, 72)
        Me.Label6.Name = "Label6"
        Me.Label6.Size = New System.Drawing.Size(31, 15)
        Me.Label6.TabIndex = 22
        Me.Label6.Text = "1100"
        '
```

```
436         'Label10
437         '
438         Me.Label10.AutoSize = True
439         Me.Label10.Location = New System.Drawing.Point(116, 18)
440         Me.Label10.Name = "Label10"
441         Me.Label10.Size = New System.Drawing.Size(42, 15)
442         Me.Label10.TabIndex = 15
443         Me.Label10.Text = "ms/pin"
444         '
445         'Label5
446         '
447         Me.Label5.AutoSize = True
448         Me.Label5.Location = New System.Drawing.Point(32, 101)
449         Me.Label5.Name = "Label5"
450         Me.Label5.Size = New System.Drawing.Size(31, 15)
451         Me.Label5.TabIndex = 21
452         Me.Label5.Text = "0110"
453         '
454         'Button4
455         '
456         Me.Button4.BackColor = System.Drawing.Color.Transparent
457         Me.Button4.FlatAppearance.MouseDownBackColor = System.Drawing.Color.FromArgb(CType(CType(64,
458 Byte), Integer), CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
459         Me.Button4.FlatAppearance.MouseOverBackColor = System.Drawing.Color.FromArgb(CType(CType(64,
460 Byte), Integer), CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
461         Me.Button4.FlatStyle = System.Windows.Forms.FlatStyle.Flat
462         Me.Button4.Location = New System.Drawing.Point(69, 68)
463         Me.Button4.Name = "Button4"
464         Me.Button4.Size = New System.Drawing.Size(20, 25)
465         Me.Button4.TabIndex = 18
466         Me.Button4.Text = "3"
467         Me.Button4.UseVisualStyleBackColor = False
468         '
469         'boxtimer
470         '
471         Me.boxtimer.BackColor = System.Drawing.Color.FromArgb(CType(CType(64, Byte), Integer),
472 CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
473         Me.boxtimer.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer),
474 CType(CType(255, Byte), Integer), CType(CType(192, Byte), Integer))
475         Me.boxtimer.Increment = New Decimal(New Integer() {5, 0, 0, 0})
476         Me.boxtimer.Location = New System.Drawing.Point(39, 16)
477         Me.boxtimer.Maximum = New Decimal(New Integer() {1000, 0, 0, 0})
478         Me.boxtimer.Minimum = New Decimal(New Integer() {1, 0, 0, 0})
479         Me.boxtimer.Name = "boxtimer"
480         Me.boxtimer.Size = New System.Drawing.Size(75, 23)
481         Me.boxtimer.TabIndex = 13
482         Me.boxtimer.Value = New Decimal(New Integer() {25, 0, 0, 0})
483         '
484         'Label4
485         '
486         Me.Label4.AutoSize = True
487         Me.Label4.BackColor = System.Drawing.Color.Transparent
488         Me.Label4.Location = New System.Drawing.Point(32, 72)
489         Me.Label4.Name = "Label4"
490         Me.Label4.Size = New System.Drawing.Size(31, 15)
491         Me.Label4.TabIndex = 20
```

```
        Me.Label4.Text = "0011"
        '
        'Button3
        '
        Me.Button3.BackColor = System.Drawing.Color.Transparent
        Me.Button3.FlatAppearance.MouseDownBackColor = System.Drawing.Color.FromArgb(CType(CType(64,
Byte), Integer), CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
        Me.Button3.FlatAppearance.MouseOverBackColor = System.Drawing.Color.FromArgb(CType(CType(64,
Byte), Integer), CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
        Me.Button3.FlatStyle = System.Windows.Forms.FlatStyle.Flat
        Me.Button3.Location = New System.Drawing.Point(6, 97)
        Me.Button3.Name = "Button3"
        Me.Button3.Size = New System.Drawing.Size(20, 25)
        Me.Button3.TabIndex = 17
        Me.Button3.Text = "2"
        Me.Button3.UseVisualStyleBackColor = False
        '
        'Label2
        '
        Me.Label2.AutoSize = True
        Me.Label2.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer),
CType(CType(255, Byte), Integer), CType(CType(192, Byte), Integer))
        Me.Label2.Location = New System.Drawing.Point(0, 40)
        Me.Label2.Name = "Label2"
        Me.Label2.Size = New System.Drawing.Size(35, 15)
        Me.Label2.TabIndex = 11
        Me.Label2.Text = "T1/2:"
        '
        'Button2
        '
        Me.Button2.BackColor = System.Drawing.Color.Transparent
        Me.Button2.FlatAppearance.MouseDownBackColor = System.Drawing.Color.FromArgb(CType(CType(64,
Byte), Integer), CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
        Me.Button2.FlatAppearance.MouseOverBackColor = System.Drawing.Color.FromArgb(CType(CType(64,
Byte), Integer), CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
        Me.Button2.FlatStyle = System.Windows.Forms.FlatStyle.Flat
        Me.Button2.Location = New System.Drawing.Point(6, 67)
        Me.Button2.Name = "Button2"
        Me.Button2.Size = New System.Drawing.Size(20, 24)
        Me.Button2.TabIndex = 16
        Me.Button2.Text = "1"
        Me.Button2.UseVisualStyleBackColor = False
        '
        'Label1
        '
        Me.Label1.AutoSize = True
        Me.Label1.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer),
CType(CType(255, Byte), Integer), CType(CType(192, Byte), Integer))
        Me.Label1.Location = New System.Drawing.Point(0, 21)
        Me.Label1.Name = "Label1"
        Me.Label1.Size = New System.Drawing.Size(33, 15)
        Me.Label1.TabIndex = 9
        Me.Label1.Text = "Rate:"
        '
        'reset
        '
```

```
        Me.reset.BackColor = System.Drawing.Color.Transparent
        Me.reset.FlatAppearance.MouseDownBackColor = System.Drawing.Color.FromArgb(CType(CType(64, Byte),
Integer), CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
        Me.reset.FlatAppearance.MouseOverBackColor = System.Drawing.Color.FromArgb(CType(CType(64, Byte),
Integer), CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
        Me.reset.FlatStyle = System.Windows.Forms.FlatStyle.Flat
        Me.reset.ForeColor = System.Drawing.Color.Maroon
        Me.reset.Location = New System.Drawing.Point(39, 103)
        Me.reset.Name = "reset"
        Me.reset.Size = New System.Drawing.Size(45, 22)
        Me.reset.TabIndex = 14
        Me.reset.Text = "Reset"
        Me.reset.UseVisualStyleBackColor = False
        '
        'stopbutton2
        '
        Me.stopbutton2.BackColor = System.Drawing.Color.Transparent
        Me.stopbutton2.FlatAppearance.MouseDownBackColor =
System.Drawing.Color.FromArgb(CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer),
CType(CType(64, Byte), Integer))
        Me.stopbutton2.FlatAppearance.MouseOverBackColor = System.Drawing.Color.FromArgb(CType(CType(64,
Byte), Integer), CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
        Me.stopbutton2.FlatStyle = System.Windows.Forms.FlatStyle.Flat
        Me.stopbutton2.Font = New System.Drawing.Font("Comic Sans MS", 8.25!,
System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.stopbutton2.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer),
CType(CType(192, Byte), Integer), CType(CType(255, Byte), Integer))
        Me.stopbutton2.Location = New System.Drawing.Point(72, 30)
        Me.stopbutton2.Name = "stopbutton2"
        Me.stopbutton2.Size = New System.Drawing.Size(58, 26)
        Me.stopbutton2.TabIndex = 10
        Me.stopbutton2.Text = "Stop"
        Me.stopbutton2.UseVisualStyleBackColor = False
        '
        'GroupBox4
        '
        Me.GroupBox4.BackColor = System.Drawing.Color.Transparent
        Me.GroupBox4.Controls.Add(Me.Label27)
        Me.GroupBox4.Controls.Add(Me.Label28)
        Me.GroupBox4.Controls.Add(Me.Label25)
        Me.GroupBox4.Controls.Add(Me.Label26)
        Me.GroupBox4.Controls.Add(Me.Label23)
        Me.GroupBox4.Controls.Add(Me.Label24)
        Me.GroupBox4.Controls.Add(Me.Label22)
        Me.GroupBox4.Controls.Add(Me.Label21)
        Me.GroupBox4.Controls.Add(Me.daq7)
        Me.GroupBox4.Controls.Add(Me.daq6)
        Me.GroupBox4.Controls.Add(Me.daq5)
        Me.GroupBox4.Controls.Add(Me.daq4)
        Me.GroupBox4.Controls.Add(Me.daq3)
        Me.GroupBox4.Controls.Add(Me.daq2)
        Me.GroupBox4.Controls.Add(Me.daq1)
        Me.GroupBox4.Controls.Add(Me.daq0)
        Me.GroupBox4.Controls.Add(Me.daqenable)
        Me.GroupBox4.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte), Integer),
CType(CType(192, Byte), Integer), CType(CType(255, Byte), Integer))
```

```
        Me.GroupBox4.Location = New System.Drawing.Point(321, 114)
        Me.GroupBox4.Name = "GroupBox4"
        Me.GroupBox4.Size = New System.Drawing.Size(398, 92)
        Me.GroupBox4.TabIndex = 31
        Me.GroupBox4.TabStop = False
        Me.GroupBox4.Text = "Real Time D.A.Q"
        '
        'Label27
        '
        Me.Label27.AutoSize = True
        Me.Label27.Location = New System.Drawing.Point(286, 66)
        Me.Label27.Name = "Label27"
        Me.Label27.Size = New System.Drawing.Size(29, 13)
        Me.Label27.TabIndex = 42
        Me.Label27.Text = "Ch7:"
        '
        'Label28
        '
        Me.Label28.AutoSize = True
        Me.Label28.Location = New System.Drawing.Point(286, 44)
        Me.Label28.Name = "Label28"
        Me.Label28.Size = New System.Drawing.Size(29, 13)
        Me.Label28.TabIndex = 41
        Me.Label28.Text = "Ch6:"
        '
        'Label25
        '
        Me.Label25.AutoSize = True
        Me.Label25.Location = New System.Drawing.Point(194, 66)
        Me.Label25.Name = "Label25"
        Me.Label25.Size = New System.Drawing.Size(29, 13)
        Me.Label25.TabIndex = 40
        Me.Label25.Text = "Ch5:"
        '
        'Label26
        '
        Me.Label26.AutoSize = True
        Me.Label26.Location = New System.Drawing.Point(194, 44)
        Me.Label26.Name = "Label26"
        Me.Label26.Size = New System.Drawing.Size(29, 13)
        Me.Label26.TabIndex = 39
        Me.Label26.Text = "Ch4:"
        '
        'Label23
        '
        Me.Label23.AutoSize = True
        Me.Label23.Location = New System.Drawing.Point(102, 65)
        Me.Label23.Name = "Label23"
        Me.Label23.Size = New System.Drawing.Size(29, 13)
        Me.Label23.TabIndex = 38
        Me.Label23.Text = "Ch3:"
        '
        'Label24
        '
        Me.Label24.AutoSize = True
        Me.Label24.Location = New System.Drawing.Point(102, 43)
```

```
660         Me.Label24.Name = "Label24"
661         Me.Label24.Size = New System.Drawing.Size(29, 13)
662         Me.Label24.TabIndex = 37
663         Me.Label24.Text = "Ch2:"
664         '
665         'Label22
666         '
667         Me.Label22.AutoSize = True
668         Me.Label22.Location = New System.Drawing.Point(11, 66)
669         Me.Label22.Name = "Label22"
670         Me.Label22.Size = New System.Drawing.Size(29, 13)
671         Me.Label22.TabIndex = 36
672         Me.Label22.Text = "Ch1:"
673         '
674         'Label21
675         '
676         Me.Label21.AutoSize = True
677         Me.Label21.Location = New System.Drawing.Point(9, 44)
678         Me.Label21.Name = "Label21"
679         Me.Label21.Size = New System.Drawing.Size(29, 13)
680         Me.Label21.TabIndex = 35
681         Me.Label21.Text = "Ch0:"
682         '
683         'daq7
684         '
685         Me.daq7.BackColor = System.Drawing.Color.FromArgb(CType(CType(64, Byte), Integer), CType(CType(64,
686 Byte), Integer), CType(CType(64, Byte), Integer))
687         Me.daq7.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte), Integer),
688 CType(CType(192, Byte), Integer), CType(CType(255, Byte), Integer))
689         Me.daq7.Location = New System.Drawing.Point(316, 62)
690         Me.daq7.Name = "daq7"
691         Me.daq7.Size = New System.Drawing.Size(62, 20)
692         Me.daq7.TabIndex = 34
693         '
694         'daq6
695         '
696         Me.daq6.BackColor = System.Drawing.Color.FromArgb(CType(CType(64, Byte), Integer), CType(CType(64,
697 Byte), Integer), CType(CType(64, Byte), Integer))
698         Me.daq6.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte), Integer),
699 CType(CType(192, Byte), Integer), CType(CType(255, Byte), Integer))
700         Me.daq6.Location = New System.Drawing.Point(316, 40)
701         Me.daq6.Name = "daq6"
702         Me.daq6.Size = New System.Drawing.Size(62, 20)
703         Me.daq6.TabIndex = 33
704         '
705         'daq5
706         '
707         Me.daq5.BackColor = System.Drawing.Color.FromArgb(CType(CType(64, Byte), Integer), CType(CType(64,
708 Byte), Integer), CType(CType(64, Byte), Integer))
709         Me.daq5.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte), Integer),
710 CType(CType(192, Byte), Integer), CType(CType(255, Byte), Integer))
711         Me.daq5.Location = New System.Drawing.Point(224, 62)
712         Me.daq5.Name = "daq5"
713         Me.daq5.Size = New System.Drawing.Size(62, 20)
714         Me.daq5.TabIndex = 32
715         '
```

```
        'daq4
        '
        Me.daq4.BackColor = System.Drawing.Color.FromArgb(CType(CType(64, Byte), Integer), CType(CType(64,
Byte), Integer), CType(CType(64, Byte), Integer))
        Me.daq4.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte), Integer),
CType(CType(192, Byte), Integer), CType(CType(255, Byte), Integer))
        Me.daq4.Location = New System.Drawing.Point(224, 40)
        Me.daq4.Name = "daq4"
        Me.daq4.Size = New System.Drawing.Size(62, 20)
        Me.daq4.TabIndex = 31
        '
        'daq3
        '
        Me.daq3.BackColor = System.Drawing.Color.FromArgb(CType(CType(64, Byte), Integer), CType(CType(64,
Byte), Integer), CType(CType(64, Byte), Integer))
        Me.daq3.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte), Integer),
CType(CType(192, Byte), Integer), CType(CType(255, Byte), Integer))
        Me.daq3.Location = New System.Drawing.Point(132, 62)
        Me.daq3.Name = "daq3"
        Me.daq3.Size = New System.Drawing.Size(62, 20)
        Me.daq3.TabIndex = 30
        '
        'daq2
        '
        Me.daq2.BackColor = System.Drawing.Color.FromArgb(CType(CType(64, Byte), Integer), CType(CType(64,
Byte), Integer), CType(CType(64, Byte), Integer))
        Me.daq2.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte), Integer),
CType(CType(192, Byte), Integer), CType(CType(255, Byte), Integer))
        Me.daq2.Location = New System.Drawing.Point(132, 40)
        Me.daq2.Name = "daq2"
        Me.daq2.Size = New System.Drawing.Size(62, 20)
        Me.daq2.TabIndex = 29
        '
        'daq1
        '
        Me.daq1.BackColor = System.Drawing.Color.FromArgb(CType(CType(64, Byte), Integer), CType(CType(64,
Byte), Integer), CType(CType(64, Byte), Integer))
        Me.daq1.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte), Integer),
CType(CType(192, Byte), Integer), CType(CType(255, Byte), Integer))
        Me.daq1.Location = New System.Drawing.Point(40, 62)
        Me.daq1.Name = "daq1"
        Me.daq1.Size = New System.Drawing.Size(62, 20)
        Me.daq1.TabIndex = 28
        '
        'daq0
        '
        Me.daq0.BackColor = System.Drawing.Color.FromArgb(CType(CType(64, Byte), Integer), CType(CType(64,
Byte), Integer), CType(CType(64, Byte), Integer))
        Me.daq0.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte), Integer),
CType(CType(192, Byte), Integer), CType(CType(255, Byte), Integer))
        Me.daq0.Location = New System.Drawing.Point(40, 40)
        Me.daq0.Name = "daq0"
        Me.daq0.Size = New System.Drawing.Size(62, 20)
        Me.daq0.TabIndex = 27
        '
        'daqenable
```

```
772         '
773         Me.daqenable.AutoSize = True
774         Me.daqenable.Checked = True
775         Me.daqenable.CheckState = System.Windows.Forms.CheckState.Checked
776         Me.daqenable.Location = New System.Drawing.Point(6, 18)
777         Me.daqenable.Name = "daqenable"
778         Me.daqenable.Size = New System.Drawing.Size(276, 17)
779         Me.daqenable.TabIndex = 0
780         Me.daqenable.Text = "Enable Real Time D.A.Q. (Slows down performance!)"
781         Me.daqenable.UseVisualStyleBackColor = True
782         '
783         'TextBox2
784         '
785         Me.TextBox2.BackColor = System.Drawing.Color.FromArgb(CType(CType(64, Byte), Integer),
786     CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
787         Me.TextBox2.BorderStyle = System.Windows.Forms.BorderStyle.None
788         Me.TextBox2.Font = New System.Drawing.Font("Comic Sans MS", 8.25!, System.Drawing.FontStyle.Bold,
789     System.Drawing.GraphicsUnit.Point, CType(0, Byte))
790         Me.TextBox2.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte), Integer),
791     CType(CType(255, Byte), Integer), CType(CType(192, Byte), Integer))
792         Me.TextBox2.Location = New System.Drawing.Point(5, 306)
793         Me.TextBox2.Name = "TextBox2"
794         Me.TextBox2.Size = New System.Drawing.Size(711, 16)
795         Me.TextBox2.TabIndex = 30
796         Me.TextBox2.Text = "Chnnl:  Signal / Time.Stmp / Event#"
797         '
798         'TextBox1
799         '
800         Me.TextBox1.BackColor = System.Drawing.Color.FromArgb(CType(CType(64, Byte), Integer),
801     CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
802         Me.TextBox1.BorderStyle = System.Windows.Forms.BorderStyle.None
803         Me.TextBox1.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte), Integer),
804     CType(CType(255, Byte), Integer), CType(CType(192, Byte), Integer))
805         Me.TextBox1.Location = New System.Drawing.Point(5, 322)
806         Me.TextBox1.Multiline = True
807         Me.TextBox1.Name = "TextBox1"
808         Me.TextBox1.ScrollBars = System.Windows.Forms.ScrollBars.Vertical
809         Me.TextBox1.Size = New System.Drawing.Size(711, 389)
810         Me.TextBox1.TabIndex = 29
811         '
812         'GroupBox3
813         '
814         Me.GroupBox3.BackColor = System.Drawing.Color.Transparent
815         Me.GroupBox3.Controls.Add(Me.Label17)
816         Me.GroupBox3.Controls.Add(Me.Labelbarstep)
817         Me.GroupBox3.Controls.Add(Me.barstep)
818         Me.GroupBox3.Controls.Add(Me.Label16)
819         Me.GroupBox3.Controls.Add(Me.Label15)
820         Me.GroupBox3.Controls.Add(Me.Labelbarfull)
821         Me.GroupBox3.Controls.Add(Me.boxcomfull)
822         Me.GroupBox3.Controls.Add(Me.Label14)
823         Me.GroupBox3.Controls.Add(Me.Label13)
824         Me.GroupBox3.Controls.Add(Me.barfull)
825         Me.GroupBox3.Controls.Add(Me.Button6)
826         Me.GroupBox3.Controls.Add(Me.boxtfire)
827         Me.GroupBox3.Controls.Add(Me.Label9)
```

```
828         Me.GroupBox3.Controls.Add(Me.boxport)
829         Me.GroupBox3.Controls.Add(Me.Label8)
830         Me.GroupBox3.Font = New System.Drawing.Font("Comic Sans MS", 8.25!,
831     System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
832         Me.GroupBox3.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte), Integer),
833     CType(CType(224, Byte), Integer), CType(CType(192, Byte), Integer))
834         Me.GroupBox3.Location = New System.Drawing.Point(8, 166)
835         Me.GroupBox3.Name = "GroupBox3"
836         Me.GroupBox3.Size = New System.Drawing.Size(307, 117)
837         Me.GroupBox3.TabIndex = 28
838         Me.GroupBox3.TabStop = False
839         Me.GroupBox3.Text = "Stepper Diagnostics"
840         '
841         'Label17
842         '
843         Me.Label17.AutoSize = True
844         Me.Label17.Location = New System.Drawing.Point(279, 42)
845         Me.Label17.Name = "Label17"
846         Me.Label17.Size = New System.Drawing.Size(16, 15)
847         Me.Label17.TabIndex = 14
848         Me.Label17.Text = "%"
849         '
850         'Labelbarstep
851         '
852         Me.Labelbarstep.AutoSize = True
853         Me.Labelbarstep.Location = New System.Drawing.Point(253, 42)
854         Me.Labelbarstep.Name = "Labelbarstep"
855         Me.Labelbarstep.Size = New System.Drawing.Size(31, 15)
856         Me.Labelbarstep.TabIndex = 13
857         Me.Labelbarstep.Text = "00.0"
858         '
859         'barstep
860         '
861         Me.barstep.BackColor = System.Drawing.Color.Black
862         Me.barstep.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte), Integer),
863     CType(CType(224, Byte), Integer), CType(CType(192, Byte), Integer))
864         Me.barstep.Location = New System.Drawing.Point(144, 42)
865         Me.barstep.Maximum = 51
866         Me.barstep.Name = "barstep"
867         Me.barstep.Size = New System.Drawing.Size(103, 13)
868         Me.barstep.TabIndex = 12
869         '
870         'Label16
871         '
872         Me.Label16.AutoSize = True
873         Me.Label16.Location = New System.Drawing.Point(0, 40)
874         Me.Label16.Name = "Label16"
875         Me.Label16.Size = New System.Drawing.Size(138, 15)
876         Me.Label16.TabIndex = 11
877         Me.Label16.Text = "Quarter Step Completion:"
878         '
879         'Label15
880         '
881         Me.Label15.AutoSize = True
882         Me.Label15.Location = New System.Drawing.Point(279, 21)
883         Me.Label15.Name = "Label15"
```

```
884         Me.Label15.Size = New System.Drawing.Size(16, 15)
885         Me.Label15.TabIndex = 10
886         Me.Label15.Text = "%"
887         '
888         'Labelbarfull
889         '
890         Me.Labelbarfull.AutoSize = True
891         Me.Labelbarfull.Location = New System.Drawing.Point(253, 21)
892         Me.Labelbarfull.Name = "Labelbarfull"
893         Me.Labelbarfull.Size = New System.Drawing.Size(31, 15)
894         Me.Labelbarfull.TabIndex = 9
895         Me.Labelbarfull.Text = "00.0"
896         '
897         'boxcomfull
898         '
899         Me.boxcomfull.BackColor = System.Drawing.Color.FromArgb(CType(CType(64, Byte), Integer),
900     CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
901         Me.boxcomfull.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte), Integer),
902     CType(CType(224, Byte), Integer), CType(CType(192, Byte), Integer))
903         Me.boxcomfull.Location = New System.Drawing.Point(126, 61)
904         Me.boxcomfull.Name = "boxcomfull"
905         Me.boxcomfull.Size = New System.Drawing.Size(35, 23)
906         Me.boxcomfull.TabIndex = 8
907         Me.boxcomfull.Text = "0"
908         '
909         'Label14
910         '
911         Me.Label14.AutoSize = True
912         Me.Label14.Location = New System.Drawing.Point(7, 64)
913         Me.Label14.Name = "Label14"
914         Me.Label14.Size = New System.Drawing.Size(113, 15)
915         Me.Label14.TabIndex = 7
916         Me.Label14.Text = "Completed Rotations:"
917         '
918         'Label13
919         '
920         Me.Label13.AutoSize = True
921         Me.Label13.Location = New System.Drawing.Point(9, 19)
922         Me.Label13.Name = "Label13"
923         Me.Label13.Size = New System.Drawing.Size(129, 15)
924         Me.Label13.TabIndex = 6
925         Me.Label13.Text = "Full Rotaion Completion:"
926         '
927         'barfull
928         '
929         Me.barfull.BackColor = System.Drawing.Color.Black
930         Me.barfull.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte), Integer),
931     CType(CType(224, Byte), Integer), CType(CType(192, Byte), Integer))
932         Me.barfull.Location = New System.Drawing.Point(144, 22)
933         Me.barfull.Maximum = 200
934         Me.barfull.Name = "barfull"
935         Me.barfull.Size = New System.Drawing.Size(103, 14)
936         Me.barfull.Step = 1
937         Me.barfull.TabIndex = 5
938         '
939         'Button6
```

```
        '
        Me.Button6.FlatAppearance.MouseDownBackColor = System.Drawing.Color.FromArgb(CType(CType(64,
Byte), Integer), CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
        Me.Button6.FlatAppearance.MouseOverBackColor = System.Drawing.Color.FromArgb(CType(CType(64,
Byte), Integer), CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
        Me.Button6.FlatStyle = System.Windows.Forms.FlatStyle.Flat
        Me.Button6.ForeColor = System.Drawing.Color.Maroon
        Me.Button6.Location = New System.Drawing.Point(167, 88)
        Me.Button6.Name = "Button6"
        Me.Button6.Size = New System.Drawing.Size(134, 22)
        Me.Button6.TabIndex = 4
        Me.Button6.Text = "Reset"
        Me.Button6.UseVisualStyleBackColor = True
        '
        'boxtfire
        '
        Me.boxtfire.BackColor = System.Drawing.Color.FromArgb(CType(CType(64, Byte), Integer),
CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
        Me.boxtfire.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte), Integer),
CType(CType(224, Byte), Integer), CType(CType(192, Byte), Integer))
        Me.boxtfire.Location = New System.Drawing.Point(126, 85)
        Me.boxtfire.Name = "boxtfire"
        Me.boxtfire.Size = New System.Drawing.Size(35, 23)
        Me.boxtfire.TabIndex = 3
        Me.boxtfire.Text = "0"
        '
        'Label9
        '
        Me.Label9.AutoSize = True
        Me.Label9.Location = New System.Drawing.Point(54, 88)
        Me.Label9.Name = "Label9"
        Me.Label9.Size = New System.Drawing.Size(66, 15)
        Me.Label9.TabIndex = 2
        Me.Label9.Text = "Total Fired:"
        '
        'boxport
        '
        Me.boxport.BackColor = System.Drawing.Color.FromArgb(CType(CType(64, Byte), Integer),
CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
        Me.boxport.ForeColor = System.Drawing.Color.FromArgb(CType(CType(255, Byte), Integer),
CType(CType(224, Byte), Integer), CType(CType(192, Byte), Integer))
        Me.boxport.Location = New System.Drawing.Point(238, 61)
        Me.boxport.Name = "boxport"
        Me.boxport.Size = New System.Drawing.Size(35, 23)
        Me.boxport.TabIndex = 1
        Me.boxport.Text = "null"
        '
        'Label8
        '
        Me.Label8.AutoSize = True
        Me.Label8.Location = New System.Drawing.Point(167, 64)
        Me.Label8.Name = "Label8"
        Me.Label8.Size = New System.Drawing.Size(65, 15)
        Me.Label8.TabIndex = 0
        Me.Label8.Text = "Port Firing:"
        '
```

```
        'Timer1
        '
        Me.Timer1.Interval = 1
        '
        'timerstep
        '
        '
        'Button1
        '
        Me.Button1.BackColor = System.Drawing.Color.Transparent
        Me.Button1.FlatAppearance.MouseDownBackColor = System.Drawing.Color.FromArgb(CType(CType(64,
Byte), Integer), CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
        Me.Button1.FlatAppearance.MouseOverBackColor = System.Drawing.Color.FromArgb(CType(CType(64,
Byte), Integer), CType(CType(64, Byte), Integer), CType(CType(64, Byte), Integer))
        Me.Button1.FlatStyle = System.Windows.Forms.FlatStyle.Flat
        Me.Button1.Font = New System.Drawing.Font("Comic Sans MS", 8.25!, System.Drawing.FontStyle.Regular,
System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.Button1.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer),
CType(CType(192, Byte), Integer), CType(CType(255, Byte), Integer))
        Me.Button1.Location = New System.Drawing.Point(9, 30)
        Me.Button1.Name = "Button1"
        Me.Button1.Size = New System.Drawing.Size(58, 26)
        Me.Button1.TabIndex = 8
        Me.Button1.Text = "Start"
        Me.Button1.UseVisualStyleBackColor = False
        '
        'MasterControls
        '
        Me.MasterControls.BackColor = System.Drawing.Color.Transparent
        Me.MasterControls.Controls.Add(Me.Button1)
        Me.MasterControls.Controls.Add(Me.stopbutton2)
        Me.MasterControls.Controls.Add(Me.reset)
        Me.MasterControls.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer),
CType(CType(192, Byte), Integer), CType(CType(255, Byte), Integer))
        Me.MasterControls.Location = New System.Drawing.Point(176, 28)
        Me.MasterControls.Name = "MasterControls"
        Me.MasterControls.Size = New System.Drawing.Size(139, 133)
        Me.MasterControls.TabIndex = 32
        Me.MasterControls.TabStop = False
        Me.MasterControls.Text = "Run Controls"
        '
        'MenuStrip1
        '
        Me.MenuStrip1.BackColor = System.Drawing.Color.Transparent
        Me.MenuStrip1.Font = New System.Drawing.Font("Comic Sans MS", 8.25!,
System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.MenuStrip1.Items.AddRange(New System.Windows.Forms.ToolStripItem()
{Me.HelpToolStripMenuItem})
        Me.MenuStrip1.Location = New System.Drawing.Point(0, 0)
        Me.MenuStrip1.Name = "MenuStrip1"
        Me.MenuStrip1.Size = New System.Drawing.Size(728, 24)
        Me.MenuStrip1.TabIndex = 33
        Me.MenuStrip1.Text = "MenuStrip1"
        '
        'HelpToolStripMenuItem
        '
```

```
1052        Me.HelpToolStripMenuItem.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte),
1053    Integer), CType(CType(255, Byte), Integer), CType(CType(255, Byte), Integer))
1054        Me.HelpToolStripMenuItem.Name = "HelpToolStripMenuItem"
1055        Me.HelpToolStripMenuItem.Size = New System.Drawing.Size(42, 20)
1056        Me.HelpToolStripMenuItem.Text = "Help"
1057        '
1058        'samples100
1059        '
1060        '
1061        'bit0CheckBox
1062        '
1063        Me.bit0CheckBox.AutoSize = True
1064        Me.bit0CheckBox.BackColor = System.Drawing.Color.Transparent
1065        Me.bit0CheckBox.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer),
1066    CType(CType(255, Byte), Integer), CType(CType(255, Byte), Integer))
1067        Me.bit0CheckBox.Location = New System.Drawing.Point(76, 53)
1068        Me.bit0CheckBox.Name = "bit0CheckBox"
1069        Me.bit0CheckBox.Size = New System.Drawing.Size(33, 19)
1070        Me.bit0CheckBox.TabIndex = 35
1071        Me.bit0CheckBox.Text = "0"
1072        Me.bit0CheckBox.UseVisualStyleBackColor = False
1073        '
1074        'bit1CheckBox
1075        '
1076        Me.bit1CheckBox.AutoSize = True
1077        Me.bit1CheckBox.BackColor = System.Drawing.Color.Transparent
1078        Me.bit1CheckBox.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer),
1079    CType(CType(255, Byte), Integer), CType(CType(255, Byte), Integer))
1080        Me.bit1CheckBox.Location = New System.Drawing.Point(105, 53)
1081        Me.bit1CheckBox.Name = "bit1CheckBox"
1082        Me.bit1CheckBox.Size = New System.Drawing.Size(31, 19)
1083        Me.bit1CheckBox.TabIndex = 36
1084        Me.bit1CheckBox.Text = "1"
1085        Me.bit1CheckBox.UseVisualStyleBackColor = False
1086        '
1087        'bit2CheckBox
1088        '
1089        Me.bit2CheckBox.AutoSize = True
1090        Me.bit2CheckBox.BackColor = System.Drawing.Color.Transparent
1091        Me.bit2CheckBox.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer),
1092    CType(CType(255, Byte), Integer), CType(CType(255, Byte), Integer))
1093        Me.bit2CheckBox.Location = New System.Drawing.Point(132, 53)
1094        Me.bit2CheckBox.Name = "bit2CheckBox"
1095        Me.bit2CheckBox.Size = New System.Drawing.Size(33, 19)
1096        Me.bit2CheckBox.TabIndex = 37
1097        Me.bit2CheckBox.Text = "2"
1098        Me.bit2CheckBox.UseVisualStyleBackColor = False
1099        '
1100        'bit3CheckBox
1101        '
1102        Me.bit3CheckBox.AutoSize = True
1103        Me.bit3CheckBox.BackColor = System.Drawing.Color.Transparent
1104        Me.bit3CheckBox.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer),
1105    CType(CType(255, Byte), Integer), CType(CType(255, Byte), Integer))
1106        Me.bit3CheckBox.Location = New System.Drawing.Point(161, 53)
1107        Me.bit3CheckBox.Name = "bit3CheckBox"
```

```
1108        Me.bit3CheckBox.Size = New System.Drawing.Size(33, 19)
1109        Me.bit3CheckBox.TabIndex = 38
1110        Me.bit3CheckBox.Text = "3"
1111        Me.bit3CheckBox.UseVisualStyleBackColor = False
1112        '
1113        'bit4CheckBox
1114        '
1115        Me.bit4CheckBox.AutoSize = True
1116        Me.bit4CheckBox.BackColor = System.Drawing.Color.Transparent
1117        Me.bit4CheckBox.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer),
1118    CType(CType(255, Byte), Integer), CType(CType(255, Byte), Integer))
1119        Me.bit4CheckBox.Location = New System.Drawing.Point(190, 53)
1120        Me.bit4CheckBox.Name = "bit4CheckBox"
1121        Me.bit4CheckBox.Size = New System.Drawing.Size(33, 19)
1122        Me.bit4CheckBox.TabIndex = 39
1123        Me.bit4CheckBox.Text = "4"
1124        Me.bit4CheckBox.UseVisualStyleBackColor = False
1125        '
1126        'bit5CheckBox
1127        '
1128        Me.bit5CheckBox.AutoSize = True
1129        Me.bit5CheckBox.BackColor = System.Drawing.Color.Transparent
1130        Me.bit5CheckBox.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer),
1131    CType(CType(255, Byte), Integer), CType(CType(255, Byte), Integer))
1132        Me.bit5CheckBox.Location = New System.Drawing.Point(219, 53)
1133        Me.bit5CheckBox.Name = "bit5CheckBox"
1134        Me.bit5CheckBox.Size = New System.Drawing.Size(33, 19)
1135        Me.bit5CheckBox.TabIndex = 40
1136        Me.bit5CheckBox.Text = "5"
1137        Me.bit5CheckBox.UseVisualStyleBackColor = False
1138        '
1139        'bit6CheckBox
1140        '
1141        Me.bit6CheckBox.AutoSize = True
1142        Me.bit6CheckBox.BackColor = System.Drawing.Color.Transparent
1143        Me.bit6CheckBox.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer),
1144    CType(CType(255, Byte), Integer), CType(CType(255, Byte), Integer))
1145        Me.bit6CheckBox.Location = New System.Drawing.Point(250, 53)
1146        Me.bit6CheckBox.Name = "bit6CheckBox"
1147        Me.bit6CheckBox.Size = New System.Drawing.Size(33, 19)
1148        Me.bit6CheckBox.TabIndex = 41
1149        Me.bit6CheckBox.Text = "6"
1150        Me.bit6CheckBox.UseVisualStyleBackColor = False
1151        '
1152        'bit7CheckBox
1153        '
1154        Me.bit7CheckBox.AutoSize = True
1155        Me.bit7CheckBox.BackColor = System.Drawing.Color.Transparent
1156        Me.bit7CheckBox.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer),
1157    CType(CType(255, Byte), Integer), CType(CType(255, Byte), Integer))
1158        Me.bit7CheckBox.Location = New System.Drawing.Point(279, 53)
1159        Me.bit7CheckBox.Name = "bit7CheckBox"
1160        Me.bit7CheckBox.Size = New System.Drawing.Size(33, 19)
1161        Me.bit7CheckBox.TabIndex = 42
1162        Me.bit7CheckBox.Text = "7"
1163        Me.bit7CheckBox.UseVisualStyleBackColor = False
```

```
        '
        'writeButton
        '
        Me.writeButton.BackColor = System.Drawing.Color.Black
        Me.writeButton.FlatStyle = System.Windows.Forms.FlatStyle.Flat
        Me.writeButton.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer),
CType(CType(255, Byte), Integer), CType(CType(255, Byte), Integer))
        Me.writeButton.Location = New System.Drawing.Point(14, 46)
        Me.writeButton.Name = "writeButton"
        Me.writeButton.Size = New System.Drawing.Size(55, 23)
        Me.writeButton.TabIndex = 43
        Me.writeButton.Text = "DI Out"
        Me.writeButton.UseVisualStyleBackColor = False
        '
        'rstckbx
        '
        Me.rstckbx.AutoSize = True
        Me.rstckbx.BackColor = System.Drawing.Color.Transparent
        Me.rstckbx.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer),
CType(CType(255, Byte), Integer), CType(CType(255, Byte), Integer))
        Me.rstckbx.Location = New System.Drawing.Point(311, 53)
        Me.rstckbx.Name = "rstckbx"
        Me.rstckbx.Size = New System.Drawing.Size(31, 19)
        Me.rstckbx.TabIndex = 44
        Me.rstckbx.Text = "r"
        Me.rstckbx.UseVisualStyleBackColor = False
        '
        'GroupBox1
        '
        Me.GroupBox1.BackColor = System.Drawing.Color.Transparent
        Me.GroupBox1.Controls.Add(Me.beam)
        Me.GroupBox1.Controls.Add(Me.tbfilename)
        Me.GroupBox1.Controls.Add(Me.rstckbx)
        Me.GroupBox1.Controls.Add(Me.writeButton)
        Me.GroupBox1.Controls.Add(Me.bit0CheckBox)
        Me.GroupBox1.Controls.Add(Me.bit7CheckBox)
        Me.GroupBox1.Controls.Add(Me.bit1CheckBox)
        Me.GroupBox1.Controls.Add(Me.bit6CheckBox)
        Me.GroupBox1.Controls.Add(Me.bit2CheckBox)
        Me.GroupBox1.Controls.Add(Me.bit5CheckBox)
        Me.GroupBox1.Controls.Add(Me.bit3CheckBox)
        Me.GroupBox1.Controls.Add(Me.bit4CheckBox)
        Me.GroupBox1.Font = New System.Drawing.Font("Comic Sans MS", 8.25!,
System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.GroupBox1.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer),
CType(CType(255, Byte), Integer), CType(CType(255, Byte), Integer))
        Me.GroupBox1.Location = New System.Drawing.Point(321, 208)
        Me.GroupBox1.Name = "GroupBox1"
        Me.GroupBox1.Size = New System.Drawing.Size(395, 75)
        Me.GroupBox1.TabIndex = 46
        Me.GroupBox1.TabStop = False
        Me.GroupBox1.Text = "File Name and Digital Resets"
        '
        'tbfilename
        '
        Me.tbfilename.BackColor = System.Drawing.Color.Black
```

```
        Me.tbfilename.BorderStyle = System.Windows.Forms.BorderStyle.None
        Me.tbfilename.Font = New System.Drawing.Font("Comic Sans MS", 8.25!,
System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.tbfilename.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer),
CType(CType(255, Byte), Integer), CType(CType(255, Byte), Integer))
        Me.tbfilename.Location = New System.Drawing.Point(14, 19)
        Me.tbfilename.Name = "tbfilename"
        Me.tbfilename.Size = New System.Drawing.Size(298, 16)
        Me.tbfilename.TabIndex = 45
        Me.tbfilename.Text = "C:\Documents and Settings\Mitch\Desktop\data0.txt"
        '
        'beam
        '
        Me.beam.BackColor = System.Drawing.Color.FromArgb(CType(CType(64, Byte), Integer), CType(CType(64,
Byte), Integer), CType(CType(64, Byte), Integer))
        Me.beam.ForeColor = System.Drawing.Color.FromArgb(CType(CType(192, Byte), Integer),
CType(CType(255, Byte), Integer), CType(CType(255, Byte), Integer))
        Me.beam.Location = New System.Drawing.Point(318, 17)
        Me.beam.Name = "beam"
        Me.beam.Size = New System.Drawing.Size(69, 23)
        Me.beam.TabIndex = 47
        '
        'MainForm
        '
        Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
        Me.BackgroundImage = CType(resources.GetObject("$this.BackgroundImage"), System.Drawing.Image)
        Me.ClientSize = New System.Drawing.Size(728, 714)
        Me.Controls.Add(Me.GroupBox1)
        Me.Controls.Add(Me.MasterControls)
        Me.Controls.Add(Me.GroupBox4)
        Me.Controls.Add(Me.runcontrols)
        Me.Controls.Add(Me.TextBox2)
        Me.Controls.Add(Me.TextBox1)
        Me.Controls.Add(Me.GroupBox3)
        Me.Controls.Add(Me.channelParametersGroupBox)
        Me.Controls.Add(Me.MenuStrip1)
        Me.FormBorderStyle = System.Windows.Forms.FormBorderStyle.FixedDialog
        Me.Icon = CType(resources.GetObject("$this.Icon"), System.Drawing.Icon)
        Me.MainMenuStrip = Me.MenuStrip1
        Me.MaximumSize = New System.Drawing.Size(2560, 4560)
        Me.MinimumSize = New System.Drawing.Size(256, 320)
        Me.Name = "MainForm"
        Me.StartPosition = System.Windows.Forms.FormStartPosition.CenterScreen
        Me.Text = "`"
        Me.channelParametersGroupBox.ResumeLayout(False)
        Me.channelParametersGroupBox.PerformLayout()
        CType(Me.threshold, System.ComponentModel.ISupportInitialize).EndInit()
        CType(Me.minimumValueNumeric, System.ComponentModel.ISupportInitialize).EndInit()
        CType(Me.maximumValueNumeric, System.ComponentModel.ISupportInitialize).EndInit()
        Me.runcontrols.ResumeLayout(False)
        Me.runcontrols.PerformLayout()
        CType(Me.halflife, System.ComponentModel.ISupportInitialize).EndInit()
        CType(Me.boxtimer, System.ComponentModel.ISupportInitialize).EndInit()
        Me.GroupBox4.ResumeLayout(False)
        Me.GroupBox4.PerformLayout()
        Me.GroupBox3.ResumeLayout(False)
```

```
1276            Me.GroupBox3.PerformLayout()
1277            Me.MasterControls.ResumeLayout(False)
1278            Me.MenuStrip1.ResumeLayout(False)
1279            Me.MenuStrip1.PerformLayout()
1280            Me.GroupBox1.ResumeLayout(False)
1281            Me.GroupBox1.PerformLayout()
1282            Me.ResumeLayout(False)
1283            Me.PerformLayout()

1284        End Sub

1285    #End Region

1286        Private Sub loopTimer_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
1287    loopTimer.Tick

1288            Try
1289                'Read one point of data per channel during every execution of the while loop
1290                Dim data() As Double = reader.ReadSingleSample()

1291                time = My.Computer.Clock.TickCount

1292                'Plot your data here
1293                If daqenable.Checked = True Then
1294                    'dataToDataTable(data, dataTable)

1295                    daq0.Text = data(0)
1296                    'daq1.Text = data(1)
1297                    'daq2.Text = data(2)
1298                    'daq3.Text = data(3)
1299                    'daq4.Text = data(4)
1300                    'daq5.Text = data(5)
1301                    'daq6.Text = data(6)
1302                    'daq7.Text = data(7)
1303                End If

1304                If data(0) > threshold.Value Then
1305                    TextBox1.AppendText("       0: " & "  " & Format(data(0), "0.0000") & "      " & time & "         " &
1306    (eventnumber + 0) & vbCrLf)
1307                    TextBox1.ScrollToCaret()

1308                    PrintLine(1, "0: " & "  " & data(0) & "  " & time & "         " & (eventnumber + 0))
1309                    bit0CheckBox.Checked = True

1310                End If
1311


1312                'If data(3) > 1 And beamstate = 0 Then
1313                'beam.Text = "beam on"
1314                'TextBox1.AppendText("beam-on " & "      " & time & vbCrLf)
1315                'PrintLine(1, "beam-on" & "  " & time)
1316                'beamstate = 1
1317                'End If

1318                'If data(3) < 1 And beamstate = 1 Then
1319                'beam.Text = "beam on"
```

```
1320            'TextBox1.AppendText("beam-off " & "      " & time & vbCrLf)
1321            'PrintLine(1, "beam-off" & "  " & time)
1322            'beamstate = 0
1323            'End If

1324            If bit0CheckBox.Checked = True Or bit1CheckBox.Checked = True Or bit2CheckBox.Checked = True Then

1325                'Digital Resets below
1326                System.Windows.Forms.Cursor.Current = Cursors.WaitCursor
1327                Dim digitalWriteTask As Task = Nothing

1328                Try
1329                    digitalWriteTask = New Task()
1330                    digitalWriteTask.DOChannels.CreateChannel("Dev1/Port0/line0:7", "port0", _
1331                        ChannelLineGrouping.OneChannelForAllLines)

1332                    Dim dataArray(7) As Boolean
1333                    CheckLines(dataArray)

1334                    Dim writer As New DigitalSingleChannelWriter(digitalWriteTask.Stream)
1335                    writer.WriteSingleSampleMultiLine(True, dataArray)

1336                Catch exception As DaqException
1337                    MessageBox.Show(exception.Message)

1338                Finally
1339                    digitalWriteTask.Dispose()
1340                    System.Windows.Forms.Cursor.Current = Cursors.Default

1341                End Try

1342                bit0CheckBox.Checked = False
1343                bit1CheckBox.Checked = False
1344                bit2CheckBox.Checked = False
1345                rstckbx.Checked = True

1346            End If

1347            If rstckbx.Checked = True Then

1348                'Digital Resets part 2 below
1349                System.Windows.Forms.Cursor.Current = Cursors.WaitCursor
1350                Dim digitalWriteTask As Task = Nothing

1351                Try
1352                    digitalWriteTask = New Task()
1353                    digitalWriteTask.DOChannels.CreateChannel("Dev1/Port0/line0:7", "port0", _
1354                        ChannelLineGrouping.OneChannelForAllLines)

1355                    Dim dataArray(7) As Boolean
1356                    CheckLines(dataArray)

1357                    Dim writer As New DigitalSingleChannelWriter(digitalWriteTask.Stream)
1358                    writer.WriteSingleSampleMultiLine(True, dataArray)

1359                Catch exception As DaqException
```

```
1360                MessageBox.Show(exception.Message)

1361            Finally
1362                digitalWriteTask.Dispose()
1363                System.Windows.Forms.Cursor.Current = Cursors.Default

1364            End Try

1365            rstckbx.Checked = False

1366        End If

1367    Catch exception As DaqException
1368        'Dispose the Task and Disable the Timer
1369        loopTimer.Enabled = False
1370        myTask.Dispose()
1371        MessageBox.Show(exception.Message)
1372    End Try

1373    eventnumber = eventnumber + 3

1374 End Sub

1375 Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Timer1.Tick

1376    Timer1.Interval = boxtimer.Value

1377    If motortimer = 1 Then
1378        Out(888, 3)

1379        ' If barstep.Value = 0 Then
1380        'Out(888, 19)
1381        'End If


1382        boxport.Text = 3
1383        boxtfire.Text = boxtfire.Text + 1
1384        barfull.Value = barfull.Value + 1
1385        barstep.Value = barstep.Value + 1

1386        If barfull.Value = 200 Then
1387            barfull.Value = 0
1388            Out(888, 19)
1389            PrintLine(1, "beam-on" & "      " & time)
1390            TextBox1.AppendText("beam-on" & "      " & time & vbCrLf)
1391            boxcomfull.Text = boxcomfull.Text + 1
1392        End If

1393        Labelbarfull.Text = Format$(barfull.Value / 200 * 100, "00.0")
1394        Labelbarstep.Text = Format$(barstep.Value / 50 * 100, "00.0")

1395        If barstep.Value = 51 Then
1396            PrintLine(1, "stepped" & "      " & time)
1397            TextBox1.AppendText("stepped" & "      " & time & vbCrLf)
1398            barstep.Value = 0
1399            timerstep.Interval = halflife.Value * 1000
```

```
            timerstep.Enabled = True
            loopTimer.Enabled = True
            Timer1.Enabled = False
        End If
    End If

    If motortimer = 2 Then
        Out(888, 6)

        'If barstep.Value = 0 Then
        'Out(888, 22)
        'End If

        boxport.Text = 6
        boxtfire.Text = boxtfire.Text + 1
        barfull.Value = barfull.Value + 1
        barstep.Value = barstep.Value + 1

        If barfull.Value = 200 Then
            barfull.Value = 0
            Out(888, 22)
            PrintLine(1, "beam-on" & "      " & time)
            TextBox1.AppendText("beam-on" & "      " & time & vbCrLf)
            boxcomfull.Text = boxcomfull.Text + 1
        End If

        Labelbarfull.Text = Format$(barfull.Value / 200 * 100, "00.0")
        Labelbarstep.Text = Format$(barstep.Value / 50 * 100, "00.0")

        If barstep.Value = 51 Then
            PrintLine(1, "stepped" & "      " & time)
            TextBox1.AppendText("stepped" & "      " & time & vbCrLf)
            barstep.Value = 0
            timerstep.Interval = halflife.Value * 1000
            timerstep.Enabled = True
            loopTimer.Enabled = True
            Timer1.Enabled = False
        End If
    End If

    If motortimer = 3 Then
        Out(888, 12)

        '   If barstep.Value = 0 Then
        'Out(888, 28)
        'End If

        boxport.Text = 12
        boxtfire.Text = boxtfire.Text + 1
        barfull.Value = barfull.Value + 1
        barstep.Value = barstep.Value + 1

        If barfull.Value = 200 Then
            barfull.Value = 0
            Out(888, 28)
            PrintLine(1, "beam-on" & "      " & time)
```

```
1446              TextBox1.AppendText("beam-on" & "      " & time & vbCrLf)
1447              boxcomfull.Text = boxcomfull.Text + 1
1448           End If

1449           Labelbarfull.Text = Format$(barfull.Value / 200 * 100, "00.0")
1450           Labelbarstep.Text = Format$(barstep.Value / 50 * 100, "00.0")

1451           If barstep.Value = 51 Then
1452              PrintLine(1, "stepped" & "      " & time)
1453              TextBox1.AppendText("stepped" & "      " & time & vbCrLf)
1454              barstep.Value = 0
1455              timerstep.Interval = halflife.Value * 1000
1456              timerstep.Enabled = True
1457              loopTimer.Enabled = True
1458              Timer1.Enabled = False
1459           End If
1460        End If

1461        If motortimer = 4 Then
1462           Out(888, 9)

1463           '   If barstep.Value = 0 Then
1464           'Out(888, 25)
1465           'End If

1466           boxport.Text = 9
1467           boxtfire.Text = boxtfire.Text + 1
1468           barfull.Value = barfull.Value + 1
1469           barstep.Value = barstep.Value + 1

1470           If barfull.Value = 200 Then
1471              barfull.Value = 0
1472              Out(888, 25)
1473              PrintLine(1, "beam-on" & "      " & time)
1474              TextBox1.AppendText("beam-on" & "      " & time & vbCrLf)
1475              boxcomfull.Text = boxcomfull.Text + 1
1476           End If

1477           Labelbarfull.Text = Format$(barfull.Value / 200 * 100, "00.0")
1478           Labelbarstep.Text = Format$(barstep.Value / 50 * 100, "00.0")

1479           If barstep.Value = 51 Then
1480              PrintLine(1, "stepped" & "      " & time)
1481              TextBox1.AppendText("stepped" & "      " & time & vbCrLf)
1482              barstep.Value = 0
1483              timerstep.Interval = halflife.Value * 1000
1484              timerstep.Enabled = True
1485              loopTimer.Enabled = True
1486              Timer1.Enabled = False
1487           End If
1488        End If

1489        motortimer = motortimer + 1
1490        If motortimer = 5 Then
1491           motortimer = 1
1492        End If
```

```
1493        End Sub

1494        Private Sub timerstep_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
1495    timerstep.Tick

1496            barstep.Value = 0
1497            Labelbarstep.Text = timerstep.Interval
1498            If boxport.Text = "3" Then
1499                motortimer = 1
1500                Timer1.Enabled = True
1501            End If
1502            If boxport.Text = "6" Then
1503                motortimer = 2
1504                Timer1.Enabled = True
1505            End If
1506            If boxport.Text = "12" Then
1507                motortimer = 3
1508                Timer1.Enabled = True
1509            End If
1510            If boxport.Text = "9" Then
1511                motortimer = 4
1512                Timer1.Enabled = True
1513            End If
1514            loopTimer.Enabled = False
1515            timerstep.Enabled = False

1516        End Sub
1517        Private Sub stopbutton2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
1518    stopbutton2.Click
1519            Timer1.Enabled = False
1520            timerstep.Enabled = False

1521            'DAQ Stop Actions
1522            'Dispose the Task and Disable the Timer
1523            loopTimer.Enabled = False
1524            myTask.Dispose()
1525            FileClose(1)

1526        End Sub

1527        Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
1528    Button2.Click
1529            Out(888, 3)
1530        End Sub

1531        Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
1532    Button3.Click
1533            Out(888, 6)
1534        End Sub

1535        Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
1536    Button4.Click
1537            Out(888, 12)
1538        End Sub
```

```
1539    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
1540 Button5.Click
1541        Out(888, 9)
1542    End Sub

1543    Private Sub reset_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles reset.Click
1544        Out(888, 0)
1545    End Sub

1546    Private Sub Button6_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
1547 Button6.Click
1548        boxtfire.Text = 0
1549        barfull.Value = 0
1550        Labelbarfull.Text = 0.0
1551        barstep.Value = 0
1552        boxport.Text = 0
1553    End Sub

1554    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
1555 Button1.Click
1556        'Digital Resets Prevariables

1557        'Stepper Prevariables
1558        Timer1.Enabled = True

1559        'DAQ Prevariables
1560        FileOpen(1, tbfilename.Text, OpenMode.Append)
1561        beamstate = 0

1562        acquiredSamplesCount = 0
1563        Try
1564            'Create a new task
1565            myTask = New Task()
1566            'Create a virtual channel
1567            myTask.AIChannels.CreateVoltageChannel(physicalChannelComboBox.Text, "", _
1568                    AITerminalConfiguration.Rse, Convert.ToDouble(minimumValueNumeric.Value), _
1569                    Convert.ToDouble(maximumValueNumeric.Value), AIVoltageUnits.Volts)
1570            'Verify the Task
1571            myTask.Control(TaskAction.Verify)

1572            reader = New AnalogMultiChannelReader(myTask.Stream)

1573            'enable the timer
1574            'loopTimer.Enabled = True

1575        Catch exception As DaqException
1576            'Dispose the Task and Disable the Timer
1577            loopTimer.Enabled = False
1578            myTask.Dispose()
1579            MessageBox.Show(exception.Message)
1580        End Try
1581    End Sub

1582    Private Sub HelpToolStripMenuItem_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
1583 Handles HelpToolStripMenuItem.Click
1584        MsgBox("If you clicked this then you obviously don't know what your doing and you need to call either Mitch
```

```
1585    at (510)684-8092 or Maz at (925)286-4907 immediately!", , "You've really done it now...")
1586        End Sub

1587        Private Sub samples100_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
1588    samples100.Tick

1589        End Sub
1590        Private Sub CheckLines(ByRef dataArray() As Boolean)
1591            dataArray(0) = bit0CheckBox.Checked
1592            dataArray(1) = bit1CheckBox.Checked
1593            dataArray(2) = bit2CheckBox.Checked
1594            dataArray(3) = bit3CheckBox.Checked
1595            dataArray(4) = bit4CheckBox.Checked
1596            dataArray(5) = bit5CheckBox.Checked
1597            dataArray(6) = bit6CheckBox.Checked
1598            dataArray(7) = bit7CheckBox.Checked
1599        End Sub


1600        Private Sub writeButton_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
1601    writeButton.Click
1602            System.Windows.Forms.Cursor.Current = Cursors.WaitCursor
1603            Dim digitalWriteTask As Task = Nothing

1604            Try
1605                digitalWriteTask = New Task()
1606                digitalWriteTask.DOChannels.CreateChannel("Dev1/Port0/line0:7", "port0", _
1607                      ChannelLineGrouping.OneChannelForAllLines)

1608                Dim dataArray(7) As Boolean
1609                CheckLines(dataArray)

1610                Dim writer As New DigitalSingleChannelWriter(digitalWriteTask.Stream)
1611                writer.WriteSingleSampleMultiLine(True, dataArray)

1612            Catch exception As DaqException
1613                MessageBox.Show(exception.Message)

1614            Finally
1615                digitalWriteTask.Dispose()
1616                System.Windows.Forms.Cursor.Current = Cursors.Default

1617            End Try


1618        End Sub

1619    End Class
```

1 References

1 L. Stavsetra, K. E. Gregorich, J. Dvorak, P. A. Ellison, I. Dragojević, M. A. Garcia, H. Nitsche *Physical Review Letters* **2009**, *103*, 132502.

2 E. R. Sylwester, K. E. Gregorich, M. D. Lee, B. Kadkhodayan, A. Türler, J. L. Adams, C. D. Kacher, M. R. Lane, C. A. Laue,C. A. McGrath, D. A. Shaughnessy, D. A. Strellis, P. A. Wilk, D. C. Hoffman, *Radiochimica.Acta* **2000**, 88, 837.

3 "The Frontiers of Nuclear Science, A Long Range Plan", Nuclear Science Advisory Committee, 2007

4 G. Savard, R. Janssens, "Proposed ATLAS efficiency and intensity upgrade", ATLAS Workshop, 2009

5 H. Podlech, A. Bechtold et. al, "A cw Heavy Ion Linac Based on Superconducting CH-Structures", GSI Scientific Report 2008, GSI-Accelerators-24.

6 D. H. Randall, M. L. Smith, *Nature* **1955**, *175*, 1041. doi:10.1038/1751041b0

7 H.J. Maier, *IEEE transactions on nuclear science* **1981**, NS-28 (2), 1575.

8 W. Parker, R. Falk, *Nuclear Instruments and Methods* **1962**, *16*, 355.

9 T. Jorgensen, *Los Alamos Scientific Laboratory* **1944**, Report LA-56.

10 M. G. Brown, S. J. Lyle, G. R. Martin, *Nuclear Instruments and Methods* **1965**, 35(2), 353.

11 J. W. Behrens, *Nuclear Instruments and Methods in Physics Research* **1982**, 200(1), 67.

12 K. M. Glover, F. J. G. Rogers, T. A. Tuplin, *Nuclear Instruments and Methods* **1972**, 102(3), 443.

13 Q. X. Jia, M. McClesky, A. K. Burrell, Y. Lin, G. E. Collis, H. Wang, A. D. Q. Li, S. R. Foltyn, *Nature* Materials **2004**, 3, 529.

14 A. Türler, K.E. Gregorich in *The Chemistry of Superheavy Elements* (Ed.: M. Schädel), Kluwer Academic Publishers, Dordtrecht, **2003**, pp. 117-157.

15 I. Zvara, V.Z. Belov, L.P. Chelnokov, V.P. Domanov, M. Hussonnois, Yu. S. Korotkin, V.A. Sheolev, M.R. Shalaevsky, *Inorganic Nuclear Chemistry Letters* **1971**, 7, 1109.

16 I Zvara, Yu.T. Chuburkov, V.Z. Belov, G.V. Buklanov, B.B. Zakhvataev, T.S. Zvarova, O.D. Maslov, R. Caletka, M.R. Shalaevsky, *Journal of Inorganic Nuclear Chemistry* **1970**, 32, 1885.

17 I. Zvara in *Transplutonium Elements* (Eds.: W. Müller, R. Lindler), North-Holland, Amsterdam, **1976**, pp. 11-20.

18 U. Hickmann, N. Greulich, N. Trautmann, H. Gäggeler, H. Gäggeler-Koch, B. Eichler, G. Herrmann, *Nuclear Instruments and Methods* **1980**, 174, 507.

19 H.W. Gäggeler, D. T. Jost, U. Baltensperger, A. Weber, A. Kovacs, D. Vermeulen, A. Türler, *Nuclear Instruments and Methods in Physics Research A* **1991**, 309, 201.

20 O. L. Keller, Jr., *Radiochimica Acta* **1984**, 37, 169.

21 J. P. Desclaux, B. Fricke, *Journal of Physics* **1980**, 41, 943.

22 V. A. Glebov, L. Kasztura, V. S. Nefedov, B. L. Zhuikov, *Radiochimica Acta* **1989**, 46, 117.

23 B. L. Zhuikov, V. A. Glebov, V. S. Nefdov, I. Zvara, *Journal of Radioanalytical Nuclear Chemistry* **1990**, 143, 103.

24 B. L. Zhuikov, Yu. T. Chuburkov, S. N. Timokhin, K. U. Jin, I. Zvara, *Radiochimica Acta* **1989**, 46, 113.

25 E. Eliav, U. Kaldor, Y. Ishikawa, *Physical Review Letters* **1995**, 74, 1079.

26 M. V. Ryzhkov, V. A. Gubanov, I. Zvara, *Radiochimica Acta* **1992**, 57, 11.

27 B. Kadkhodayan, A. TüIrler, K. E. Gregorich, P. A. Baisden, K. R. Czerwinski, B. Eichler, H.W. Gäggeler, T. M. Hamilton,D. T. Jost, C. D. Kacher, J. Kovacs, S. A. Kreek, M. R. Lane, M. Mohar, M. P. Neu,N. J. Stoyer, E. R. Sylwester, M. D. Lee, M. J.Nurmia, G. T. Seaborg, D. C. Hoffman, *Radiochimica Acta* **1996**, 72, 169.
28 E. R. Sylwester, K. E. Gregorich, M. D. Lee, B. Kadkhodayan, A. Türler, J. L. Adams, C. D. Kacher, M. R. Lane, C. A. Laue,C. A. McGrath, D. A. Shaughnessy, D. A. Strellis, P. A. Wilk, D. C. Hoffman, *Radiochimica.Acta* **2000**, 88, 837.
29 B. Eichler, R. Eichler in *The Chemistry of Superheavy Elements* (Ed.: M. SchVdel), Kluwer Academic Publishers, Dordrecht, **2003**, pp. 205 – 236.
30 V. Pershina, B. Fricke in *Heavy Elements and Related New Phenomena* (Eds.: W. Greiner, R. K. Gupta), World Scientific, Singapore, **1999**, pp. 194 – 262.
31 B. L. Zhuikov, V. A. Glebov, V. S. Nefedov, I. Zvara, *Journal Radioanalytical Nuclear Chemistry* **1990**, 143, 103.
32 V. Pershina, B. Fricke, *Journal of Physical Chemistry* **1994**, 98, 6468.
33 V. Pershina, B. Fricke in *Heavy Elements and Related New Phenomena* (Eds.: W. Greiner, R. K. Gupta), World Scientific, Singapore, **1999**, pp. 194 – 262.
34 B. Kadkhodayan, A. TüIrler, K. E. Gregorich, P. A. Baisden, K. R. Czerwinski, B. Eichler, H.W. Gäggeler, T. M. Hamilton,D. T. Jost, C. D. Kacher, J. Kovacs, S. A. Kreek, M. R. Lane, M. Mohar, M. P. Neu,N. J. Stoyer, E. R. Sylwester, M. D. Lee, M. J.Nurmia, G. T. Seaborg, D. C. Hoffman, *Radiochimica Acta* **1996**, 72, 169.
35 A. Türler, G. V. Buklanov, B. Eichler, H. W. Gäggeler, M.Grantz, S. Hübener, D. T. Jost, V. Ya. Lebedev, D. Piquet, S. N.Timokhin, A. B. Yakushev, I. Zvara, *J.Alloys Compd.* **1998**, 271–273, 287.
36 A. Türler, H. W. Gäggeler, K. E. Gregorich, H. Barth, W. Brüchle, K. R. Czerwinski, M. K. Gober, N. J. Hannink, R. A. Henderson, D. C. Hoffman, D. T. Jost, C. D. Kacher, B. Kadkhodayan, J. Kovacs, J. V. Kratz, S. A. Kreek, D. M. Lee,J. D. Leyba, M. J. Nurmia, M. Schädel, U. W. Scherer, E. Schimpf, D. Vermeulen, A. Weber, H. P. Zimmermann, I. Zvara, *Journal of Radioanalytical Nuclear Chemistry* **1992**, 95, 327.
37 E. R. Sylwester, K. E. Gregorich, M. D. Lee, B. Kadkhodayan, A. Türler, J. L. Adams, C. D. Kacher, M. R. Lane, C. A. Laue,C. A. McGrath, D. A. Shaughnessy, D. A. Strellis, P. A. Wilk, D. C. Hoffman, *Radiochimica.Acta* **2000**, 88, 837.
38 A. Türler, H. W. Gäggeler, K. E. Gregorich, H. Barth, W. Brüchle, K. R. Czerwinski, M. K. Gober, N. J. Hannink, R. A. Henderson, D. C. Hoffman, D. T. Jost, C. D. Kacher, B. Kadkhodayan, J. Kovacs, J. V. Kratz, S. A. Kreek, D. M. Lee,J. D. Leyba, M. J. Nurmia, M. Schädel, U. W. Scherer, E. Schimpf, D. Vermeulen, A. Weber, H. P. Zimmermann, I. Zvara, *Journal of Radioanalytical Nuclear Chemistry* **1992**, 95, 327.
39 A. Türler, *Radiochim.Acta* **1996**, 72, 7.
40 K. E. Gregorich in *Proc.of The Robert A. Welch Foundation 41st Conference on Chemical Research*, The TransactinideElements, October 27 – 28, **1997**, Houston, Texas, pp. 95–124.
41 I. Zvara, Yu. T. Chuburkov, V. Z. Belov, G. V. Buklanov, B. B.Zakhvataev, T. S. Zvarova, O. D. Maslov, R. Caletka, M. R.Shalaevsky, Journal of Inorganic Nuclear Chemistry **1970**, 32, 1885.
42 B. Eichler, R. Eichler in *The Chemistry of Superheavy Elements* (Ed.: M. SchVdel), Kluwer Academic Publishers, Dordrecht, **2003**, pp. 205 – 236.
43 V. P. Domanov, U. Zin Kim, *Radiokhimiya* **1989**, 31, 12.

44 H.W. Gäggeler, A. Türler in *The Chemistry of Superheavy Elements* (Ed.: M. Schädel), Kluwer Academic Publishers, Dordrecht, **2003**, pp. 237 – 289.
45 A. Ghiorso, M. Nurmia, K. Eskola, P. Eskola Physics Letters **1970**, 32B, 95.
46 R.J. Silva, J. Harris, M. Nurmia, K. Eskola, A. Ghiorso. Inorganic Nuclear Chemistry Letters **1970**, 6, 871.
47 B. Kadkhodayan, A. TüIrler, K. E. Gregorich, P. A. Baisden, K. R. Czerwinski, B. Eichler, H.W. Gäggeler, T. M. Hamilton,D. T. Jost, C. D. Kacher, J. Kovacs, S. A. Kreek, M. R. Lane, M. Mohar, M. P. Neu,N. J. Stoyer, E. R. Sylwester, M. D. Lee, M. J.Nurmia, G. T. Seaborg, D. C. Hoffman, *Radiochimica Acta* **1996**, 72, 169.
48 E. R. Sylwester, K. E. Gregorich, M. D. Lee, B. Kadkhodayan, A. Türler, J. L. Adams, C. D. Kacher, M. R. Lane, C. A. Laue,C. A. McGrath, D. A. Shaughnessy, D. A. Strellis, P. A. Wilk, D. C. Hoffman, *Radiochimica.Acta* **2000**, 88, 837.
49 Yu. A. Lazarev, V. Lobanov, Yu. Ts. Oganessian, V. K. Utyonkov, F. Sh. Abdullin, A. N. Polyakov, J. Rigol, I. V. Shirovsky, Yu. S. Tsyganov, S. Iliev, V. G. Subbotin, A. M. Sukhov, G. V. Buklanov, A. N. Mezentsev, K. Subotic *Physical Review Letters* **2000**, 62, 064307.
50 N. Trautmann, H. Folger, *Nucl. Instrum. Methods Phys. Res., Sect. A* **1989**, *282*, 102.
51 G. Mullen, D.C. Aumann, *Nucl. Instrum. Methods* **1975**, *128*, 425.
52 .M. Glover, T.A. Tuplin and F.J.G. Rogers, *Nucl. Instrum. Methods* **1972**, *102*, 443.
53 Q.X. Jia, T.M.McCleskey, A.K. Burrell, Y. Lin, G.E. Collis, H.Wang, A.D.Q.Li, S.R. Foltyn, Nat. Mater. **2004**, *3*, 529.
54 Y. Lin, H. Wang, M.E. Hawley, S.R. Foltyn, Q.X. Jia, G.E. Collis, A.K.Burrell, T.M. McCleskey, Appl. Phys. Lett. **2004**, *85*, 3426.
55 Y. Lin, J. Xie, H. Wang, Y. Li, C. Chavez, S.Y. Lee, S.R. Foltyn, S.A.Crooker, A.K. Burrell, T.M. McCleskey, Q.X. Jia, Thin Solid Films **2005**, *492*, 101.
56 Q.X. Jia, T.M.McCleskey, A.K. Burrell, Y. Lin, G.E. Collis, H.Wang, A.D.Q.Li, S.R. Foltyn, Nat. Mater. **2004**, *3*, 529.
57 D.W. Schubert and T. Dunkel, *Mater. Res. Innov.* **2003**, *7*, 314.
58 B.D. Cullity and S.R. Stock, "Elements of X-ray Diffraction (3 ed.)", Prentice Hall, Menlo Park CA **2001** p. 664.
59 N. Trautmann, H. Folger, *Nucl. Instrum. Methods Phys. Res., Sect. A* **1989**, *282*, 102.
60 G. Mullen, D.C. Aumann, *Nucl. Instrum. Methods* **1975**, *128*, 425.
61 N. Trautmann, H. Folger, *Nucl. Instrum. Methods Phys. Res., Sect. A* **1989**, *282*, 102.
62 G. Mullen, D.C. Aumann, *Nucl. Instrum. Methods* **1975**, *128*, 425.
63 W.S. Aaron, M. Petek, L.A. Zevenbergen and J.R. Gibson, *Nucl. Instr. and Meth.***1989**, *282*, 147.
64 R. A. Langley *Nucl Instrum. Meth.* **1973**, *113*, 109.
65 H. M. Mann, J. W. Haslett, G. Ottaviani, *IEEE Transactions on Nuclear Science* **1969**, *NS-16*, 35.
66 W. Seibt, K. E. Sundstorm, P. A. Tove, *Nuclear Instruments and Methods* **1973**, *113*, 317.
67 E. C. Finch, *Nuclear Instruments and Methods* **1974**, *121*, 431.
68 E. C. Finch, *Nuclear Instruments and Methods* **1975**, *129*, 617.
69 C.F.G. Delaney, E. C. Finch, *Nuclear Instruments and Methods* **1983**, *215*, 219.
70 I. Kanno, *Review of Scientific Instruments* **1987**, *58*, 1926.
71 I. Kanno, T. Inbe, S. Kanasawa, I. Kimura, Review of Scientific Instruments **1994**, *65*, 3050.
72 R. N. Williams, E. M. Lawson, *Nuclear Instruments and Methods* **1974**, *120*, 261.

73 H. Henschel, H. Hipp, A. Kohnle, F. Gonnenwein, *Nuclear Instruments and Methods* **1975**, *125*, 365.
74 R. N. Williams E. M. Lawson, *Nuclear Instruments and Methods* **1974**, *120*, 261.
75 H. Henschel, H. Hipp, A. Kohlne, F. Gonnenwein, *Nuclear Instruments and Methods* **1975**, *125*, 365.
76 E. C. Finch, C. F. G. Delaney, M. Asghar, *IEEE Transactions on Nuclear Science* **1980**, *NS-27*, 286.
77 E. C. Finch, A. A. Cafolla, M. Asghar, *Nuclear Instruments and Methods* **1982**, *198*, 547.
78 H. O. Neidel, H. Henschel, H. Geissel, Y. Laichter, *Nuclear Instruments and Methods* **1983,** *212*, 299.
79 R. Butsch, J. Pochodzalla, B. Heck, *Nuclear Instruments and Methods* **1985,** *228*, 586.
80 J. W. T. Dabbs, F. J. Walter, *Semiconductor Nuclear Particle Detectors*, Publication No. 871, Washington DC: NAS-NRC, 237 (1961).
81 E. R. Sylwester, K. E. Gregorich, M. D. Lee, B. Kadkhodayan, A. Türler, J. L. Adams, C. D. Kacher, M. R. Lane, C. A. Laue,C. A. McGrath, D. A. Shaughnessy, D. A. Strellis, P. A. Wilk, D. C. Hoffman, *Radiochimica.Acta* **2000**, 88, 837.
82 K. E. Gregorich, V. Ninov, Superheavy Elements with the Berkeley Gas-Filled Separator, *J. Nucl. Radiochem. Sci* **2000**, *1*, 1.
83 E. Segré, *Nuclei and Particles: An Introduction to Nuclear and Subnuclear Physics*, 2nd Ed. (Benjamin/ Cummings, Reading, Massachusetts, 1977), pp. 26-27.
84 K. E. Gregorich (private communication).
85 H. G. Essel, J. Hoffmann, R. Richter, D. Schall, *VME Trigger Description*, http://www-gsi-vms.gsi.de/goodoc/GM_VME_TRIG.ps
86 H. G. Essel, J. Hoffmann, N. Kurz, R. S. Mayer, W. Ott, D. Schall, *IEEE Trans. Nucl. Sci.* **1996**, *NS-43*, 132.
87 H.G. Essel, N. Kurtz, *IEEE Trans. Nucl. Sci.* **2000**, *NS-47*, 337.
88 W. Loveland, K. E. Gregorich, J. B. Patin, D. Peterson,C. Rouki, P. M. Zielinski, and K. Aleklett *Phys. Rev. C* **2002**, *66*, 044617.
89 K. E. Gregorich, W. Loveland, D. Peterson, P. M. Zielinski, S. L. Nelson, Y. H. Chung, Ch. E. Düllmann, C. M. Folden III, K. Aleklett, R. Eichler, D. C. Hoffman, J. P. Omtvedt, G. K. Pang, J. M. Schwantes, S. Soverna, P. Sprunger, R. Sudowe, R. E. Wilson, H. Nitsche *Phys. Rev. C* **2005**, *72*, 014605.
90 J. F. Ziegler *Nucl. Instrum. Methods B* **2004**, *219*, 1027.
91 G. Audi, O. Bersillon, J. Blachot, and A. H.Wapstra *Nucl. Phys.A* **2003**, *729*, 3.
92 W. D. Myers and W. J. Swiatecki, Lawrence Berkeley National Laboratory Report; LBNL-36803 **1994**; http://ie.lbl.gov/txt/ms.txt.
93 K. E. Gregorich, W. Loveland, D. Peterson, P. M. Zielinski, S. L. Nelson, Y. H. Chung, Ch. E. Düllmann, C. M. Folden III, K. Aleklett, R. Eichler, D. C. Hoffman, J. P. Omtvedt, G. K. Pang, J. M. Schwantes, S. Soverna, P. Sprunger, R. Sudowe, R. E. Wilson, H. Nitsche *Phys. Rev. C* **2005**, *72*, 014605.
94 K. E. Gregorich , T. N. Ginter, W. Loveland, D. Peterson, J. B. Patin, C. M. Folden III, D. C. Hoffman, D. M. Lee, H. Nitsche, J. P. Omtvedt, L. A. Omtvedt, L. Stavsetra, R. Sudowe, P. A. Wilk, P. M. Zielinski, K. Aleklett, *Eur. Phys. J. A* **2003**, *18*, 633.
95 C. M. Folden III, K. E. Gregorich, Ch. E.Düllmann, H. Mahmud, G. K. Pang, J. M. Schwantes, R. Sudowe, P. M. Zielinski,H. Nitsche, D. C. Hoffman, *Phys. Rev. Lett.* **2004**, *93*, 212702.

96 *Table of Isotopes*, 8th ed., edited by R. B. Firestone and V. S.Shirley (Wiley, New York, **1996**).
97 J. Dvorak, W. Brüchle, M. Chelnokov, Ch. E. Düllmann, Z. Dvorakova, K. Eberhardt, E. Jäger, R. Krücken, A. Kuznetsov, Y. Nagame, F. Nebel, K. Nishio, R. Perego, Z. Qin, M. Schäde, B. Schausten, E. Schimpf, R. Schuber, A. Semchenkov, P. Thörle, A. Türler, M. Wegrzecki, B. Wierczinski, A. Yakushev, A. Yeremin *Phys. Rev. Lett.* **2008**, 100, 132503.
98 B. Kadkhodayan, A. TüIrler, K. E. Gregorich, P. A. Baisden, K. R. Czerwinski, B. Eichler, H.W. Gäggeler, T. M. Hamilton,D. T. Jost, C. D. Kacher, J. Kovacs, S. A. Kreek, M. R. Lane, M. Mohar, M. P. Neu,N. J. Stoyer, E. R. Sylwester, M. D. Lee, M. J.Nurmia, G. T. Seaborg, D. C. Hoffman, *Radiochimica Acta* **1996**, 72, 169.
99 E. R. Sylwester, K. E. Gregorich, M. D. Lee, B. Kadkhodayan, A. Türler, J. L. Adams, C. D. Kacher, M. R. Lane, C. A. Laue,C. A. McGrath, D. A. Shaughnessy, D. A. Strellis, P. A. Wilk, D. C. Hoffman, *Radiochimica.Acta* **2000**, 88, 837.
100 K. Eskola, P. Eskola, M. Nurmia, and A. Ghiorso, *Phys. Rev. C* **1971**, *4*, 632.
101 E. D. Donets, V. A. Shchegolev, and V. A. Ermakov, *Sov. J. Nucl. Phys.* **1966**, *2*, 723 .
102 J. B. Patin, Lawrence Berkeley National Laboratory Report; LBNL-49593 (2002).