Kilo-Qubit Scale Quantum Computer Design

DESIGN DOCUMENT

Team #24
Project for: Dr. Durga Paudyal, Ames Lab
Advisors: Dr. Durga Paudyal, Dr. Jonathan Smith
Team Members:
Nicholas (Nick) Greenwood, Jacob Frieden,
Samuel (Sam) Degnan, Arvid Gustafson,
Chad (Colin) Gorgen, Emile Albert Kum Chi
sdmay23-24@iastate.edu
https://sdmay23-24.sd.ece.iastate.edu

Revised: December 1, 2022

Executive Summary

Development Standards & Practices Used

- **Waterfall Design Methodology (Development Practice)**
    - Waterfall Methodology is used because we do not have the knowledge to create many different sprints of designs and will need to put lots of knowledge into a final design.
- **P1730 - Standard for Quantum Computing Definitions (Development Standard)**
    - Through the development of this computer, we must communicate effectively. Adhering to standard definitions will be a must.
- **P1731 - Standard for Quantum Computing Performance Metrics & Performance Benchmarking (Development Standard)**
    - At the end of the development of the computer, we will use standard benchmarking procedures in our simulations to evaluate the performance of our proposed design.

Summary of Requirements

Fundamental:
- Design a quantum computer that can be scaled to hold thousands of qubits. The computer should utilize memory and computational ion traps and needs trampsort access. It is not required to be optical hardware addressable.

Physical:
- The design of the QC must be of a reasonable size (classical desktop sized), in line with the fabrication capabilities of Sandia[sic] labs, and capable of performing low-noise.

User Experiential:
- Control, error correction, and optimization of quantum gates and circuits need to reliably provide expected performance with minimal user overhead.
- Documentation on implementation of these features should be provided.

Economic / Market:
- We would need to utilize outside labor and outside funding to physically build any components.

Other:
- Submit a patent for our design if successful

Applicable Courses from Iowa State University Curriculum

- **"PHYS 422X/522X: Foundations of Quantum Computing"**
    - Highly relevant class, none of us have taken it
- **Other PHYS courses**
    - General physics classes can contain useful information on basic, non-quantum physics

Sources For New Skills/Knowledge Acquired (that was not taught in Iowa State University Curriculum)

- **Quantum Computation and Quantum Information**
  - Textbook by Isaac Chuang and Michael Nielsen
- **Microfabricated Ion Trap Junctions: 3D cross interchange***
  - Paper by Gavin Nop (TA)
- **On Stabilizer Techniques and Their Application to Simulation and Certification of Quantum Devices**
  - Paper on Error Correction
- **Honeywell Ion-Trap Quantum Computer Design Documentation/Review**
  - Presentation by Gavin Nop
- **Computational Physics 4860**
  - University of Northern Iowa physics class
- **Modern Physics 4100**
  - University of Northern Iowa physics class
- **Modern Physics Lab 4110**
  - University of Northern Iowa physics class
- **Various papers, lectures, virtual classes, and Youtube videos on ion traps, quantum computation, error correction, noise and other general quantum terminology and all associated topics discussed in this paper**

Table of Contents

List of figures/tables/symbols/definitions

- **Superposition**
    - Being a combination of both computational basis states (|0> and |1>)
- **Entanglement**
    - One qubit acts in a similar manner to another
- **Quantum Fidelity**
    - Commonly used in quiskit; Represented by capital 'F'
    - (bad) 0 <= F <= 1 (good)
        - An example of 1.0 Fidelity would be finding that collapsing a positive state |+> would result in 50% collapse into 0 and 50% into 1, a perfect distribution
        - If this collapse resulted in always 0s, that a F = 0.5
        - Scales exponentially with # of qubits (2 qubits = 2^2 states)
    - Basically a measure of how closely you can expect your qubit to behave irl compared to in quantum coding
    - Essentially the link between "quantum coding" and actually quantum computing
    - Fidelity < 1 is a result of noise
    - $<\psi|A|\psi>$ = Application of A to ψ, the basis which fidelity is determined off of
- **Quantum Laws**
    - Superposition (see above)
- **Clusters (Quantum and classical meanings)**
    - Set of computers that work together, as such can be viewed as a single system
    - Each node performs the same task
    - Deadlock
        - Two applications are fighting for the same resource, they prevent the other from accessing it, and so both programs cease
        - In single computers (ie nodes), deadlock is handled through algorithms and the OS, which means it probably isn't a big deal
- **Nodes (Quantum and classical meanings)**
    - Each perform the same task in traditional (meaning should be easiest to implement)
    - Each run their own OS
- **Quantum Supremacy**

    - Exactly how it sounds: The goal / idea that a quantum computer can solve a (potentially useless) problem that a classical computer can't in any feasible amount of time
    - Also known as "quantum advantage"
    - Is expected to be done with "near-term" QCs, as the goal doesn't require high-quality error correction or the problem to be useful; no impact or hurdle of commercially viable QCs; primarily scientific
    - Examples of potential problems:
        - Shor's Algo for factoring integers - prime factorization of an n-bit integer in O(n^3) time
        - Boson sampling - usage of boson scattering to evaluate expectation values of permanents of matrices
        - Sampling an output distribution of a quantum circuit - scales difficult exponentially with number of qubits
- **Quantum Volume ("Area")**
    - Metric of capability for a quantum computer, introduced by IBM in 2019
    - Maximum size of square quantum circuits that can be successfully implemented, is always a power of 2
    - As of September 2022, Honeywell's H1 is king at QV 8192
        - 13 square circuits
        - Did this by implementing arbitrary angle two-qubit gates
    - Honeywell has been increasing at a rate of 10x per year
- **Arbitrary angle two-qubit gates**
    - Currently, researchers are working with single qubit gates of fully entangled two qubit gates. Arbitrary angle gates mean we can operate with two partially entangled gates
    - Implemented by Honeywell in their Sept 2022 QV test
    - Instead of having to fully entangle then walk back, can now just add a slight but of entanglement

Very useful for fourier transform, where they can have ½ the number of arbitrary angle gates than traditional two qubit gates

Tables:

Figures:

1 Team

1.1 Team Members

- Nicholas Greenwood - Computer Engineering
- Jacob Frieden - Software Engineering
- Sam Degnan - Software Engineering
- Arvid Gustafson - Software Engineering
- Colin Gorgen - Electrical Engineering and Physics
- Emile Albert Kum Chi - Electrical Engineering

1.2 Required Skill Sets for Your Project

A basic understanding of quantum mechanics and a more thorough understanding of quantum computing are required. Knowledge in chip manufacturing techniques, coding techniques for simulation purposes, and robust capability of synthesizing technical writing are all paramount to the success of this project.

1.3 Skill Sets covered by the Team

Refer to 8.4.1 Team Contract for list of skills covered by each team member

1.4 Project Management Style Adopted by the team

We have opted for a distributed management style. Nicholas is the contact point for communication between the team and the client / professors and the management of deadlines for documents and other pieces of senior design work. When work items near due, Nicholas divides up the work evenly among team members.

For all non-senior design work (ie project based work), each team member undertakes their own assignment and sees that is completed to the clients standard. While assignments may overlap, this usually does not become apparent until the team meets again.

1.5 Initial Project Management Roles

- Nicholas Greenwood - Administration, Qubits
- Jacob Frieden - Administration, Quantum Machine LEarning (QML)
- Sam Degnan - Error Correction
- Arvid Gustafson - Coherence, Global Design
- Colin Gorgen - Quantum Gates and Physical Implementation
- Emile Albert Kum Chi - Physical Implementation

2 Introduction

2.1 Problem Statement

We will further the collective knowledge base of quantum computing and computer design by collectively contributing to the design and construction of a working quantum computer at ISU/Ames Lab over the next two semesters. This project will not conclude with us and will be carried on by future staff and students. We are doing so because quantum computing is a cutting edge technology, which offers opportunities to provide numerous advances in computational and scientific fields, and as a national lab and associated research university, Ames Lab and ISU's goals for furthering the state of science align with their construction of a quantum computer.

We will do this by subdividing and specializing into six sub-fields of quantum computer design with a focus on ion trap design and accumulating a knowledge base as we work. Robust communication and cross-educational sessions will be employed to ensure that along with our specializations, the knowledge necessary to address overall design concerns is accumulated across the entire team. Additionally, we have designated an integration specialist. Given the nature of our client, we will take guidance from them regarding the transition from research to development focus, at which point we will define our initial steps in construction in accordance with our research findings and available resources.

2.2 Intended Users and Uses

Ames Lab

- **Characteristics:** homogenous employment; heterogeneous expertise & explicit goals connected to their background, existing proposals/fundings, etc.; Highly technical individuals, likely interested in details and implementation as much as final product;
- **Needs:** Development of new techniques in the design of a quantum computer/proof of concept for existing techniques, verbose documentation/explanation of work
- **How They Use / Benefit:** They will be able to conduct research and forward the current knowledge base on quantum computer design and computing, increasing the productive potential and prestige of the institution.

Iowa State University students and faculty

- **Characteristics:** Large, diverse, and scholastic. A subset of students, likely in ECPE, Physics, ComSci, and related fields will be the most likely to be interested in this product. Within this subset, there are still wildly different areas of knowledge that will correspondingly result in different interests and concerns regarding our product. That said, they will all be technically inclined, though possibly to a lesser degree than the members of Ames Lab, and their access may be comparably limited.
- **Needs:** Access to quantum computing and/or quantum computer design starting at a possibly lower level of technical background than can be assumed of our other user base. This suggests the need for a full bodied "zero to hero" documentation structure.
- **How They Use / Benefit:** Involvement in the quantum computing domain - increases the prestige of the university and the real value offered to its associates through access/exposure to the computer and its design. **Students will have an expanded range of real world projects they can work on and take advantage of.** Will utilize knowledge and any components for furthering of our goal or for new discoveries

State-of-the art researchers:

- **Characteristics:** Continuous drive for improvement, working on science projects. Very open and sharing for the benefit of everyone. Diverse ethnicities, cultural backgrounds, first-languages
- **Needs:** Perform high level calculations, Develop new solutions to current problems using new techniques and technologies developed by themselves or others
- **How They Use / Benefit:** Personal or group glory by using these concepts for further development in the field, enhanced knowledge in the field; Will utilize knowledge and any components for furthering of our goal or for new discoveries

2.3 Requirements & Constraints

Kilo-qubit (scale) Ytterbium Ion-Trap Quantum Computer (QC) Design

Fundamental:

- Design a quantum computer that can be scaled to hold thousands of qubits
- The design should utilize memory ion traps that preserve qubits for longer times (10s of machine cycles). These need to have transport access, but need not be optical hardware addressable.
- The design should also utilize computational ion-traps, which are the standard within current ion-trap QC designs.

Resource:

- Mike and Ike quantum physics book
- Honeywell Ion-Trap Quantum Computer Design Documentation/Review
- Papers, lectures, and virtual classes on ion traps and quantum computation
- Quantum Computer design software
- A suitable word processor (Microsoft Office / G Suite) for documentation

Physical:

- The design of the QC must be of a reasonable size (classical desktop sized)
- The QC design must be in line with the fabrication capabilities of Sandia[sic] labs, our design implementation collaborators
- QC must be capable of performing low-noise / interference ion transport along the trap.
    - Note: Software based "transport" (swapping) mechanisms exist, but are impractically error-prone.
    - Physical ion-transport is the standard, and minimal ion transport distance is prioritized to decrease error from noise exposure along the transport channel.

Aesthetic:

- There are no aesthetic requirements, as the design will most likely be virtual on our end, and at a nano-meter scale when implemented.

User Experiential:

- The design should theoretically work as expected, according to the rules of quantum computation.
- Control, error correction, and optimization of quantum gates and circuits need to reliably provide expected performance with minimal user overhead. Documentation on implementation of these features should be provided, but not necessary for use.

Economic / Market:

- There are no economic requirements
- There may be economic constraints
    - The ability to product QC-level components is not one that Iowa State possesses

    - We would need to utilize outside labor and outside funding to physically build any components
        - Labor in the form of Sandia[sic] labs, our design partner
    - We may or may not get far enough to prototype anything, but if we do, we will work within the constraints of any grants and financing we can get.

Other:

- Submit a patent for our design if successful

2.4 Engineering Standards

Due to the nature of our design work, few existing standards will be utilized. The few that will be used are IEEE Quantum Standards

- **P1730 - Standard for Quantum Computing Definitions**
    - Through the development of this computer, we must communicate effectively. Adhering to standard definitions will be a must
- **P1731 - Standard for Quantum Computing Performance Metrics & Performance Benchmarking**
    - At the end of the development of the computer, we will use standard benchmarking procedures in our simulations to evaluate the performance of our proposed design

3 Project Plan

3.1 Project Management/Tracking Procedures

Due to the large amount of knowledge that we need to build up, we will be using a waterfall approach. Creating a viable product iteration simply isn't possible within a small portion of time such as a sprint. The requirements are also well worked out and shouldn't update as the semester progresses. Due to this being a largely physical design (in concept) of this Quantum Computer (QC), the iteration will come in further implementation of features into an overall computer design, as opposed to doing full minimum viable product (MVPs) and iterating on each one.

We are using GroupMe for communication and sharing knowledge amongst student team members. We utilize a mass-email chain for communication amongst all individuals involved in the project. We also have a shared drive with an extensive directory containing our accumulated knowledge base, presentations, design documents, and eventually our designs. Finally, we have weekly 2-3 hour meetings to talk about the progress that we have made with our project. We may use Git if we get to a point where software development becomes relevant, and are generally in an ad hoc stage of development tool selection as we become more acquainted with our needs.

3.2 Task Decomposition

- **Knowledge Acquisition: Reading papers, projects, journal, courses, etc. on quantum mechanics and computing. Can be broken into semi-distinct areas of research:**
    - General Quantum Mechanics Knowledge - Mike and Ike book on Quantum
    - Quantum Computer Design - Client Provided State of the Art Review, Honeywell and IonQ whitepapers, etc.
    - Quantum Simulation and Design - IBM Quantum, QISKIT, Quirk, more forthcoming

- **Initial computer design: Decomposing this step is contingent on adequate understanding of quantum computing components. Tentatively, as follows**
  - Defining Ion-trap Design
    - Memory vs. Computation Traps
    - Geometry Selection (tentatively: 2D linear, junctioned with trap inversion)
    - Electrode configuration (Quantum implications)
    - Laser Systems (Quantum implications):
      - Cooling: Doppler, Simulated Raman, Sideband
      - Operational: Gate Implementation
      - Global
  - Classical computer control systems
    - Laser system controls (Classical implementation)
    - Electrode controls (Classical implementation)
  - Stuff we don't know enough to list
- **Revised computer design(s)**
  - Client functional review and testing benchmarks
  - Optimization
    - MLfQ
    - Circuit optimization techniques
    - Noise reduction methods
    - Error correction implementation/revision
- **Simulate portions of the quantum computer.**
  - Simulation will be an ongoing component of the initial and revised computer design tasks, and does not subdivide well separate from the initial computer design subdivisions. We include it as a separate major on account of the fact that the initial design sections are not concretely or exclusively tied to simulation.
- **Rough draft of write up about design.**
  - Background Section
  - Highlight of novel developments
    - Methods, Goals, Results
  - Testing and performance evaluation
    - Methods, Results
  - Conclusion
    - Final results, fabrication discussion (if applicable), further work
  - Reference/Resource Section
- **Final write up with design and all documentation allowing for future continuation.**
  - Population of missing figures/experiments
  - Draft revision

3.3 Project Proposed Milestones, Metrics, and Evaluation Criteria

Due to the highly theoretical nature of our project -the construction of a quantum computer- , quantitative or physical milestones are difficult to create:

- **Knowledge Acquisition**
  - This step is purely measured by relative knowledge
  - In all likelihood, knowledge acquisition will continue to be the primary step of this entire project, and will extend until the end of the project in some fashion

    - A good yardstick to measure our success in this step is by our understanding of increasingly complex bodies of knowledge
    - Completion of this step can be marked when we feel comfortable enough to create an initial computer design
- **Initial Computer Design**
    - Milestones for this step can be measured based on relative completion on individual components:
        - ion trap (Electrode induced pseudopotential well, optical hardware system, doppler cooling system);
        - Computational, Memory, and possibly Transport Trap configuration (i..e. differentiating trap implementations/connections between traps)
        - Classical control systems for lasers and electrodes, etc.
    - Performance metrics will include: error, coherence time, noise, and quantum volume. These will be more important for the revised design, as we focus more on improvement.
    - This step will be complete when we have a single iteration, comprehensive QC design consisting of layout of ion traps, nodes, and clusters
- **Revised Computer Design**
    - Performance metrics will include: error, coherence time, noise, and quantum volume. <u>These are quantitative measures, but at this time describing them quantitatively would be unreasonable in the scope of this document.</u>
    - This step will be complete when we have tested and gone over multiple iterations of the initial computer and component designs
- **Simulation**
    - This will be an easy step to benchmark, as simulations can be run based on current QC benchmarks
    - Similar to knowledge acquisition, this step will not be complete until we say it is. As we come up with different designs for components, we can simulate each of them to any extent necessary.
- **Paper / Presentation Rough Draft**
    - This step will not have significant substeps or milestones other than completion
    - Evaluation can be done by completion of individual paper sections listed in 3.2
    - This step will be complete when the paper is written in its entirety
- **Paper / Presentation Final Draft**
    - This step will not have significant substeps or milestones other than completion
        - We will divide successive drafts between the initial rough draft and final draft as milestones,
    - Evaluation can be done by completion of sections listed in 2.2
    - A good evaluation metric would be whether or not we get published in a journal. This will not be required, but completion to the extent we could reasonably submit and be reviewed for publication, as well as publication, would serve as indicators of exceptional performance. This will be subject to the discretion of our client.

3.4 Project Timeline/Schedule

	August	September	October	November	December	January	February	March	April	May

Knowledge Acquisition										
Initial computer design										
Revised computer design										
Simulate										
Rough draft write up										
Final write up										

Table 1: Project time management throughout the year

3.5 Risks And Risk Management/Mitigation

Knowledge Acquisition: No risks present, risk factor of zero.

Initial design: Risk of software design not meeting the design specifications of the hardware. Very significant, risk factor of 0.75. We can mitigate this by peer-reviewing each of our solutions and having many stringent checking processes in place.

Revised design: Risk of implementing improper software, which could lead to a faulty fabrication. Risk factor: .1 due to lots of revisions done at each step **.**

Simulations: There are risks in not being able to manipulate whatever software we choose in the correct way, or getting the wrong software. The risk should be low due to our mentors knowledge therefore the risk is: .2

Rough draft write up: No risks present, risk factor of zero**.**

Final write up: There are severe risks with messing up the final design, which is why we mitigate the risk by having a rough draft. Risk: .1 since we take lots of care to mitigate risks to get to this step

3.6 Personnel Effort Requirements

	Explanation	Textual reference
Knowledge acquisition	Each individual will do 5 hours a week of knowledge acquisition and compiling the information along with a 3 hour meeting, accumulating in	*Mike and Ike Quantum Book, QisKit, Honeywell Quantum Computing, IBM Quantum Computing*

	about 224 hours over the course of 4 months.	
Initial Design	Each individual will spend 6 weekly hours to contribute to a large design document. Creating the design from our knowledge should take about 100 hours.	*Mike and Ike Quantum Book, QisKit, Honeywell Quantum Computing, IBM Quantum Computing*
Revised Design	Each individual will spend 3 weekly hours to review and revive aspects of our design. Revising should take about 75 hours.	*Mike and Ike Quantum Book, QisKit, Honeywell Quantum Computing, IBM Quantum Computing*
Simulation	Each individual will spend 4 weekly hours to assist in the construction of a simulation of our design. Simulation will take longer due to a new learning curve, making it take around 150 hours.	*Mike and Ike Quantum Book, QisKit, Honeywell Quantum Computing, IBM Quantum Computing*
Rough Draft	Each individual will spend 6 weekly hours to contribute to a rough draft of our paper. The rough draft should be able to take from things we have already done, though taking a good amount of work amounting to 100 hours.	*Mike and Ike Quantum Book*
Final write Up	Each individual will spend 6 weekly hours to revise the rough draft to produce a final draft. The final write up will have to be carefully produced, taking 90 hours.	*Mike and Ike Quantum Book*

Table 2: Personal Effort Requirements and Associated Textual Reference

3.7 Other Resource Requirements

- **Knowledge base entries:** textbooks, journal articles, whitepapers, etc.
- **Simulation softwares** (possibly local computational power for running simulations)
- **High-end fabrication facilities/tools** ⇒ Connections with Sandia establishe**d**
- **Physical System model** (i.e. implementing controls for optical hardware, trap electrodes, etc. will require a system model for these components… buy or build indeterminate)

4 Design

4.1 Design Context

4.1.1 Broader Context

Area	Description
Public health, safety, and welfare	Due to the purely theoretical and research-oriented nature of our design, it has minimal immediate impacts on public health, safety and welfare of people. The design is intended for students and faculty in the field of quantum computing / quantum physics, and stands to further the goals of the field. Down the line, it is our hope that quantum computers will be commercially viable and unlock a new paradigm of computing power for the general welfare. We believe we are doing our part in furthering this goal. Hopefully, our contributions to the field of quantum computing will enhance public welfare, as prior advancements in technology have.
Global, cultural, and social	The most significant impact of this project will be in global, cultural, and social areas. With this being a design of Iowa State origin, although our findings will be publicly available, the project stands to benefit Iowa State oriented individuals the most. Iowa State is very behind in the race for quantum computing power and the end goal of this project is to reduce that gap through further recognition of the school's efforts, increases in staff count and caliber in this department, and student engagement in the field. Doing so will increase the longstanding culture at Iowa State of ingenuity. As a team, we say "If the first digital computer was invented at Iowa State, why can't the first large-scale Quantum Computer?" While ambitious, there is a chance of global effects of our design. If our design works and performs as we intend it to, this could help the global field of quantum computing move forward. It could help open a door for the QC community into another school of design.
Environmental	Again, due to the purely theoretical and research-oriented nature of this design, the environmental impacts of it are negligible at the moment. A physical quantum computer of this specification would require a great amount of energy to run, similar to other Quantum Computers (QCs) of today. The power requirements and necessity of running the computer at 10 Kelvin would be significant relative to many other senior design projects. Our QC would utilize materials standard in other QCs, but procedures around obtaining such materials may still be harmful to society and the environment. This is not something we have much affect over, as we would not be constructing the computer, and as such, sourcing the materials.
Economic	Should this QC stay within the realm of a hypothetical design, the economic impact will be very minimal. We hope that our design will spur further ideation and design creation, further leading to the financial viability of quantum computers as a whole. If we begin to construct a physical QC, the economic impact will be more substantial, due to the requirement of designing such a cutting-edge machine. We couldn't do any sort of component production at Iowa State and would have to outsource it to a DoE lab. Even still, our computer would most likely not be financially viable, and as such, not have any large macroeconomic effects for us or associated parties.

Table 3: Ramifications of our project

4.1.2 Prior Work/Solutions

The Honeywell and IonQ Quantum Computers have been built, but they have very few qubits, and are therefore limited. Our design will be similar in many regards, but designed for a great many qubits, around 1,000. We will do this by arranging multiple ion traps together into a cluster, and using a special memory-type ion trap that allows qubits to last longer.

Our design will be more like the Honeywell QC. Both use Ytterbium (Yb) ions for qubits, each use electrodes to keep them in an ion trap, and each use beams of lights/lasers to set, address, cool and manipulate them. The Honeywell QC cools its Yb ions using Barium (Ba), while the IonQ QC does not. The Honeywell QC physically moves its qubits around to have them interact with each other, whereas the IonQ computer transfers information using light and swap gates.

Source: AVS Quantum Sci. 3, 044101 (2021); https://doi.org/10.1116/5.0065951

4.1.3 Technical Complexity

The complexity and scope of designing a quantum computer is apparent to us:

- The physics and mathematics of the operations inside a quantum computer are so complex (literally, they have a large focus on complex numbers, e.g. (a+bi,c+di)) that we have largely neglected them for the purposes of the design. We are treating ion-traps, an extremely new and relatively unproven design, to be a single, solid entity within our computer. Our computer will consist of a number of these ion traps in a specific layout to facilitate computing on a larger-scale than other QCs currently known in the public domain.
- Our problem is fundamentally an engineering problem - the layout and rough construction of a Quantum Computer with particular attributes. This holistically encompasses many aspects of the engineering design process - including ideation (see below), tradeoff consideration (technical and otherwise), and rough prototyping.
- Our Quantum Computer will consist of ion-traps laid out in a novel design to accomplish basic computing responsibilities: mutilation and storage of data for multiple cycles. This is very similar to designing a traditional computer, but with an extra helping of mathematics and physics.
- The handoff of ions between traps relies heavily on particle physics
- The scope of our design problem hinges on the creation of a kilo-qubit scale quantum computer - a novel concept yet to be successfully created. Designs for QCs with 10s or low hundreds of qubits do exist, but haven't been implemented due to cost, material, or technology constraints. We intend on making this computer with current technologies and materials, which will definitely be a challenge. Pushing the boundaries of a new field with existing technology will prove to be a sufficient challenge.

4.2 Design Exploration

4.2.1 Design Decisions

Number of clusters in the computer

- These are the largest layer of the QC. Depending on their role, the number of these may be important to not bottleneck the computer during operation. An adequate number of these is important to the "scale" aspect of our design, as multiplying these elements will quickly get our computer to the size we'd like it to be.

Number of nodes in a cluster

- These are the intermediate layer of the QC. Depending on their role, the number of these may be important to not bottleneck the computer during operation. The number of each of these in each computing node will set the upper limit of computations the computer could perform.

Number of traps per node

- These are the base layer of the QC. They serve a fundamentally different role than the larger two layers due to their importance to the physical operations of the computer. The number of each of these in each computing node will set the upper limit of computations each node unit could perform.

Function(s) of traps, nodes and/or clusters

- The decision to use all computing components as duplicates which do any one or number of functions would sway the number of each type of component needed to effectively perform computing operations.

Physical orientation of traps relative to other traps, nodes to other nodes, clusters to other clusters

- The orientation of each component is of paramount importance with respect to the physical tradeoff of ions, ability to hold information for multiple machine cycles, and ability to use quantum computing effectively.

4.2.2 Ideation

We must decide upon the physical orientation of traps relative to others. As previously mentioned, the orientation of the ion traps relative to other ones will be fundamental in the handoff of ions between traps, a crucial standpoint of our design and the driving force behind our scalable, modular design.

- **Square Grid design with upper and lower tracks**
    - This was the design recommended by the advisors of our project. This type of design is supported by modern, 2-level wafer electronics printing and will most likely be what we moved forward with. Similar to a street-level grid used in many newer towns, a meeting point would consist of two, three, or four “roads” (ion traps) converging in “intersections.” This design calls for explicit usage of the cardinal directions to maintain order
- **Tree design**
    - Two major factors permeate our design requirements. Firstly, the addition of ion traps at their intersections introduces additional noise and uncertainty, as a qubit traveling from one to another is less likely to go in the intended direction. Secondly, the movement of a qubit reduces its coherence time, and introduces error into the QC. Therefore, we would do good to minimize both the number of ion traps at each junction, and the distance between each qubit. A tree-like design, such as a binary tree, compromises between these constraints. It allows for each qubit to only need to travel O(log(n)) ion traps to get to any other qubit,as opposed to a line or circle, which requires O(n) ion traps to be traversed. It also requires less ion traps per junction between ion traps than a grid or wheel spoke. Even if the design is not strictly a tree, it can have aspects that are like one.
- **Wheel and spoke design**
    - This was a design thought of by a team member. It involves many ion traps converging at a central point, where ions could be handed off to any one of the

number of “spoke” ion traps. Around the outside of these spoke traps, we could have a “wheel” of ion traps providing a potentially different function.

- **Triangular grid design with upper and lower tracks**
  - This is a slightly different iteration of the grid design. Instead of having squares, we could have a triangular grid, with each connection being a meeting point of three ion traps instead of two or four. This design would not use the cardinal directions, and all intersections would have exactly three connections (outside of those on the corners of the grid).
- **Other 3+ Layer designs**
  - This is a subset of thoughts that we came up with when considering the binary tree design. This school of design requires the practical capability of electronics manufacturing with 3+ layer wafer design. We did not look much into this option, as to our knowledge, such wafer design is not possible.

4.2.3 Decision-Making and Trade-Off

Physical Layout of computer

- **Grid design with upper and lower tracks**
  - Pros:
    - Moderate junction density, at most 4 for a square grid
  - Cons:
    - Moderate-High O(n^(1/2)) distance between qubits
- **Tree design**
  - Pros:
    - Moderate-low O(logn) distance between qubits
    - Moderate-low junction density, at most 3 in the case of a binary tree.
    - Principles of the tree design may be applied to other designs.
  - Cons:
    - Leafs have the worst travel time, and they are the most prevalent.
    - Other designs have better qubits distance or junction density individually
- **Wheel and spoke design**
  - Pros:
    - Low distance between qubits
    - Low junction density on the wheel, 2 or 3 ion traps per junction.
    - Could support parallel computing in multiple “spoke” ion traps
  - Cons:
    - High junction density, particularly in the center of the wheel
    - Ions could get lost in the hub of the wheel, where the spokes all meet
- **Triangular grid design**
  - Pros:
    - Moderate-Low distance between qubits.
    - The structure would be most dense, and take up less space.
  - Cons:
    - High junction density, at most 6 for a triangular grid
- **Other 3+ Layer Designs**
  - Pros:

        - Denser, taking up less space.
        - Vertical traversal allows for shorter qubit traversal time.
    - Cons:
        - Likely high junction density
        - Probably not possible with current technology

We have not yet made an official decision on this or other considerations yet, as knowledge acquisition is still underway to help more thoroughly inform our decisions.

4.3 Proposed Design

4.3.1 Overview

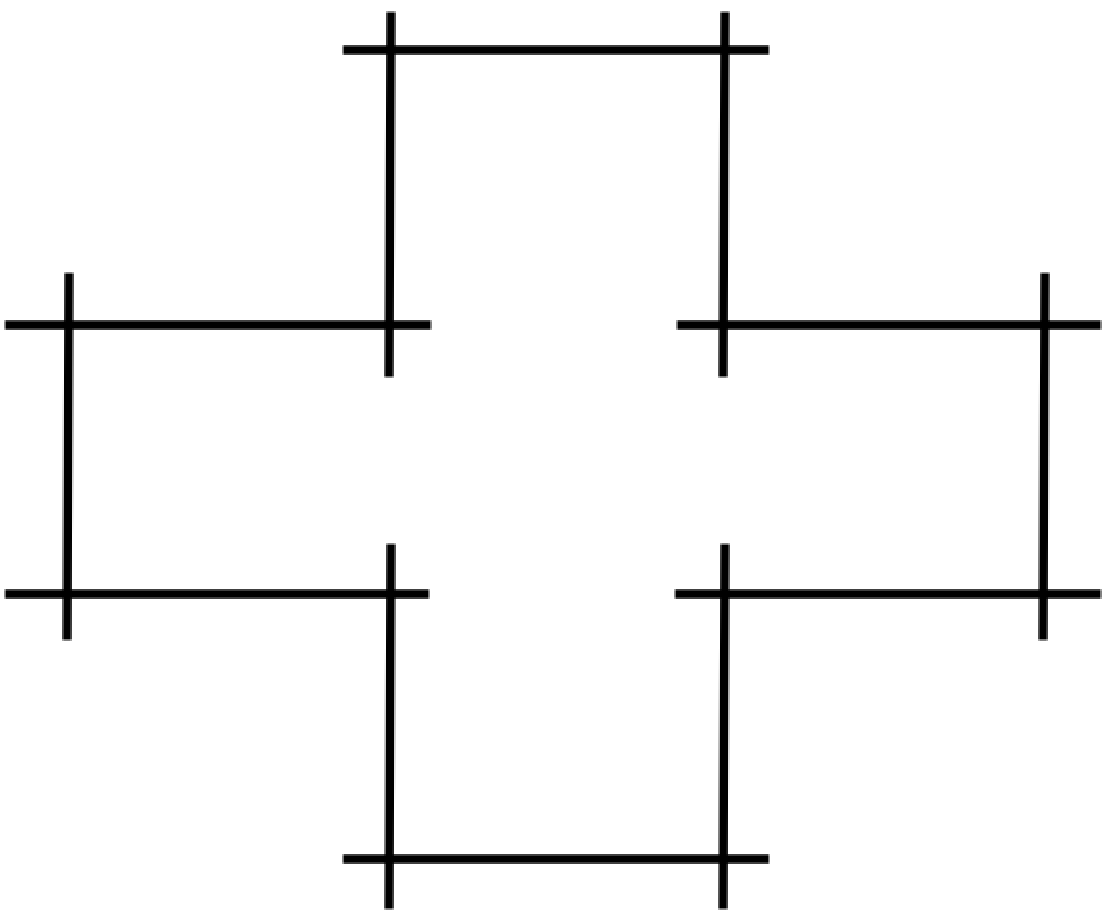

Figure 1: A brief overview of the shape of our RF ion trap design

We are designing a quantum computer schematic. Pictured above is our design for a node of this computer. This node will be composed of 12 ion traps, and each will hold 5-10 ions, each ion acting as a quantum bit ("qubit"). Each of these nodes will operate in the same fashion, and will connect to each other. To transfer information within a node, these qubits will continuously move between ion traps at junction points near their ends. Due to the state of current technology, it is best to implement junctions between two ion traps with a right angle. The part of each ion trap that extends past this junction helps to guide the qubit to its destination. Additionally, the ends of each ion trap contain a DC voltage stop - a wall that stops ions from flying off the ion trap. The cross design is composed of only right angle junctions, and is, to some degree, symmetric, so forces from electrodes may cancel each other out (which is desired).

4.3.2 Detailed Design and Visual(s)

We are designing a quantum computer schematic. The central part of the computer is a node, which itself is a quantum computer holding a number of qubits on the order of ~100. The node will be used in clusters to build a quantum computer of many clusters, which in total can hold on the order of 1000 qubits.

A "qubit" is a quantum bit in a superposition between 0 and 1. A single qubit may be represented as a vector of two complex numbers, the magnitudes of each are the probabilities that the qubit will be either 0 or 1 once it is measured, so the magnitude of the magnitudes of the elements of the vector is always 1. Qubits may be manipulated by quantum gates, which may be thought of as unary matrices, that is, any matrix when multiplied to its conjugate transpose becomes the identity matrix. This allows the probabilities of a qubit being 0 or 1 to be flipped, or for the polarity of one of the aforementioned complex numbers to be flipped, or for a qubit to gain an equal probability of yielding 0 or 1, from a state of exactly 0 or 1, but do nothing when applied twice. There are also two-qubit gates, such as the controlled not gate, which flips one qubit's value if and only if the other will yield 1 when measured. This allows for quantum entanglement, a situation where one qubit will not yield a different result than another qubit will when the same outside stimuli are applied to both. Complex quantum algorithms take advantage of these qualities to perform tasks faster than on digital computers. Though the complex numbers cannot be measured directly, their magnitudes can be measured by running the same algorithms multiple times, and measuring average responses.

As stated prior, qubit information is stored in Ytterbium cations, which are called "physical qubits." The last valence electron in the Ytterbium cation exists in a superposition between its usual orbital, and an excited state, allowing for us to use the ion (electron) as a qubit. Multiple physical qubits may be used together to model a single qubit with greater accuracy, called a logical qubit. We use lasers to emit photons, which then impart the Ytterbium atom with energy depending on the wavelength, as the last electron can absorb the incoming light, and jump to a higher state. Depending on the frequency of light provided, the electron could jump to a specific unstable state, and then go back down to its usual position. This allows us to set specific qubits to be either 0 or 1. Furthermore, we can measure the orbit of the electron by sending light to it, and then measuring what we get, because some light may be absorbed by the ion and will not go through it. Using these lasers, we can set and access the qubits.

In order for the qubits to be useful, the ions must exist at a very low temperature; otherwise, there will be too much noise, and the physical qubits we do have will retain information for less time. This temperature will be around 10 - 12 Kelvin. Therefore, we will employ multiple methods of cooling, including doppler cooling and passive Barium cooling. In doppler cooling, we emit light at a specific frequency to cause the Ytterbium ion to lose energy and slow down. In passive Barium cooling, we place Barium ions between the Ytterbium ions, and then cool the barium ions more aggressively, which then cool the Ytterbium ions. The image below depicts an ion gate with ions, and a beam of light going from the top to the bottom.

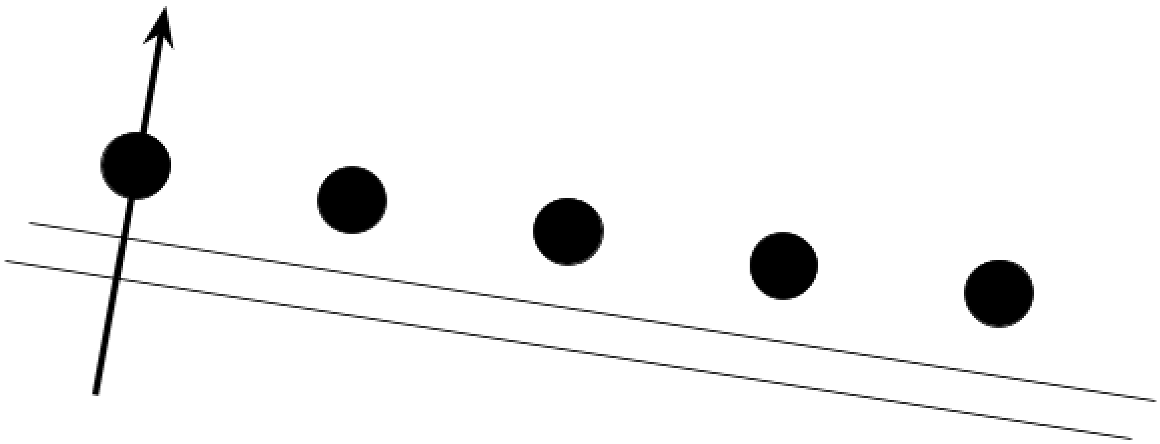

Figure 2: Ions suspended above an ion trap

Useful quantum algorithms are complex, and require many qubits to execute. Therefore, we aim to create a machine with a great many qubits, on the order of 1000, to advance the field of quantum computing. The primary challenge to accomplishing this feat is that qubits are very unstable, and exist for only a short amount of time. Usually, the addition of more qubits in a quantum computer introduces noise, and reduces the amount of time a qubit can hold accurate information. Therefore, we will use special memory-specific ion traps that are designed to only store qubits without doing gate operations on them. This is a key point where our design diverges from existing designs.

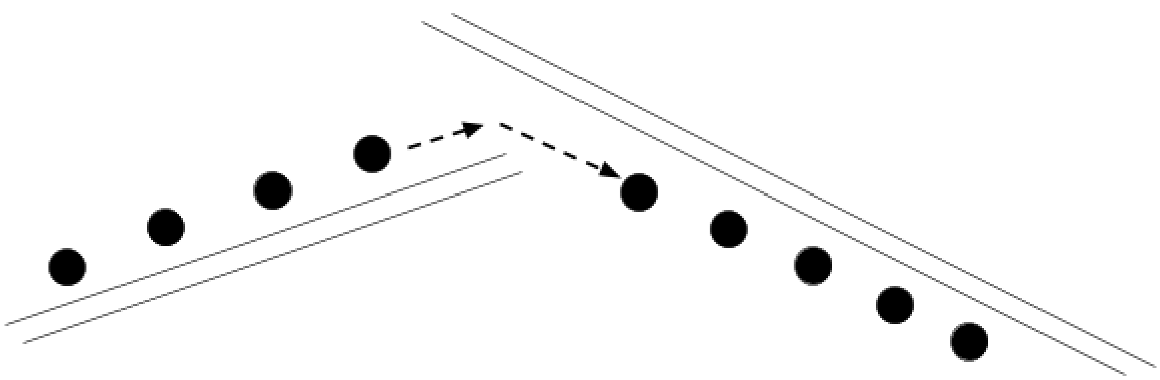

Figure 3: Visualization of an ion handoff between two ion traps

In addition to using memory specific ion traps, we will also use a vertical transfer, shown above. In this, qubits will transfer from one ion trap to another vertically, just hovering across. We expect that this will limit the noise that a qubit encounters while traversing ion traps.

4.3.3 Functionality

The goal of this project is to produce a viable quantum computer *design* that can achieve on the order of 1,000 qubits (referred to previously as "kilo-qubit"). Since the goal is a viable design, our client should expect to be able to apply tests or simulations against it to assess aspects of its functionality. Our guiding concern at the moment is the layout configuration of electrode rails called ion traps on top of which is where the qubits of the machine are located.

- Ion trap configuration testing: we will offer a proposed ion trap layout. While our current design is a result of our own thoughts, our final design may be generated by currently undetermined means. Our project will include a means to determine if ions configured in this layout behave as necessary for function in quantum computation

Additionally this *design* is meant to be viable for computation. We will have to make certain allowances for scale, but the computational capacity of our design will also need to be testable in some way, with respect to the trap configuration, since that's what's novel about our design.

- Computation testing: our design will need to provide computational viability against testing and simulation that pay respect to decoherence factors such as transport interference, laser addressal, etc.

4.3.4 Areas of Concern and Development

We are still very early in the pre-design phase, lots of unknowns. Some of the most important factors of our design thus far may not even work, which we refer to later in 4.5 Design Analysis. Further development on our design will be largely dependent on access to high-grade electro-dynamic simulations. With our existing simulation software, we can not be sure that the results of our simulations accurately represent a real world outcome. We must be certain that under scrutiny from more well-endowed individuals and institutions with access to more advanced modeling software, our design holds up.

We have numerous concerns regarding our client needs. Our ion-trap topology will need to be sufficiently well simulated and highly optimized to achieve the client's desired volume of qubits in a feasible configuration (i.e. without decoherence factors rendering the system useless). All the while, we must maintain a topology that can address the numerous functional requirements for ion-trap quantum computer design, e.g. laser cooling and addressal, loading, etc. The largest concern at the moment is trap topology and the accurate simulation thereof. Until we have determined feasible topologies, the other requirements are difficult or meaningless to attempt to address.

We must continue our dialogue with advisors and look into functionality of existing technologies to see if our existing design is even worth testing.

4.4 Technology Considerations

We will be making a quantum computer out of one modular component - a Ytterbium ion-trap. While a fairly recent development within the broader scope of computing, this is a tested way of performing computational operations. These ion traps have two segmented RF electrodes with DC voltage steps that create a "tube" above the trap that the ion sits in.

One of the most important parts of our project is that we will use existing technology, materials and methodologies. We are not in the place to create new and advanced materials or technologies, financially and educationally. The pro of doing this is that we don't have to invent something new, but the con is that utilizing existing technologies to come up with a design that outperforms existing ones in many respects will be a challenge. We believe that the continuous transference of the ion between traps could be a game changer, if it works.

We are using Ytterbium ion-trap design for our quantum computer because the current existing king, the Honeywell H1, has been continuously breaking records in terms of quantum volume - a metric introduced by IBM to show the computational power of a quantum computer. The H1 has set the last three records of quantum volume, and has 10x'd their quantum volume annually. Other types of quantum computers, such as those utilizing superconducting ions or use material defects to trap ions, are either too unsophisticated at this point in time or have had their recent progress slow.

4.5 Design Analysis

We have not built or implemented any components. Due to the fact that our project simply calls for a non-physical design, there will be no construction or physical implementation of the design. As of last week, we began rudimentary mathematical simulations of an ion trap. By the end of the next week, we should have uncovered whether continuous exchange of an ion between ion traps without degradation would be possible. We currently believe it to be possible, and our overlapping-ion-trap design hinges on it being so. For this reason, there are serious implications for the overall feasibility of our design.

As previously mentioned, the difficult part of this assignment is the creation of the design. There is a high likelihood that some of our sub-systems or entire design may not 1. Be functional 2. Push the boundaries of quantum computing or 3. Be feasible with current technology. Put another way, we are not only working in the paradigm of "Will this be good or change the game?" but also "Is this even possible?"

Our plans for future design and non-physical implementation include more extensive testing of the continuous handoff of ions, hopefully getting access to a more thorough (and expensive) modeling software with which to test larger sub-systems of our design, and continuous refinement of our sub-systems in a way that contributes to our overall goal of a kilo-qubit scale quantum computer. These items are in chronological order, and we do not expect to move on to the last item until next semester.

5 Testing

5.1 Unit Testing

- Qubits and Ion traps are being tested individually through a C++ simulation.
- Quantum component with ion traps through the same C++ code, though it would have to be upgraded.
- Structures through a 3D design software.
- Quantum program testing through algassert.com/quirk.
- Laser addressing simulation through whatever software we can find to simulate it.

5.2 Interface Testing

There are no interfaces within our project as we are making a quantum computer. The programming of this computer would be classical, and as such, outside of the scope of this project.

5.3 Integration Testing

- The ion traps will interact with each other. We will simulate ion trap units together.
- The quantum component will be in the traps.
- The ion traps will be assembled into a larger structure.
- The quantum programs will run on the computer. This is important because some problems can be solved with programs.
- Lasers will interact on the ion traps.

5.4 System Testing

The primary testing will be the overall structure and how it works together, as it is the most important part we are innovating. The 3D testing, quantum component of ion traps, and the laser addressing will be enough to be considered system testing.

5.5 Regression Testing

We will test as we go, at each step to ensure that no addition of components of code breaks the system. Whenever one change will impact others, we will retest. This shouldn't happen too frequently as most tests are relatively contained from the rest. Many components test an individual aspect of the design, and given correct testing in the previous iterations, we should be able to iterate conveniently.

5.6 Acceptance Testing

- If ions can be moved between ion traps, that would be a measure of proper function
- If ion traps can be organized in a way conducive to the tradeoff of ions between traps and nodes, that would be a measure of proper function
- If we can fit the ion traps within a reasonable form factors conducive to efficient computing, that would be a non-functional requirement met
- The designation of these things is not only based on the client's desire, but on physical properties (space, electro-dynamics)

5.7 Security Testing (if applicable)

There will not be a security component to our computer within the scope of our project. While all physical systems of this manner will have a cybersecurity component, the security component of this computer will end up being outside the scope of our program.

5.8 Results

We hope to have a working simulation of our quantum computer in c++ and have an accurate paper about it. The testing will verify whether or not our physical guesses translate well when simulated. Our requirements are designated by us, so our compliance with the requirements will be designated by us. Here is an example of the code we are using for rudimentary testing:

```cpp
//I'm generating this with mathematica code rn
//This should be first the mins, then the maxes in x y z order
//x is the axis for the ions, z is perpindicular to the trap surface
//so Controlm31 is in the -y direction
std::multimap<std::string, std::array<float, 6>> trapGeom{
  {"RF1", {{-800.0f, 245.0f, -2.0f, 800.0f, 500.0f, 0.0f}}},
  {"RF1", {{-800.0f, -500.0f, -2.0f, 800.0f, -245.0f, 0.0f}}},
  {"RFouter1", {{-800.0f, 150.0f, -2.0f, 800.0f, 245.0f, 0.0f}}},
  {"RFouter1", {{-800.0f, -245.0f, -2.0f, 800.0f, -150.0f, 0.0f}}},
  {"Ground1", {{-800.0f, 55.0f, -2.0f, 800.0f, 150.0f, 0.0f}}},
  {"Ground1", {{-800.0f, -150.0f, -2.0f, 800.0f, -55.0f, 0.0f}}},
  {"Controlm11", {{-100.0f, 30.0f, -14.0f, 0.0f, 55.0f, -12.0f}}},
  {"Controlm11", {{-100.0f, -55.0f, -14.0f, 0.0f, -30.0f, -12.0f}}},
  {"Controlm21", {{-200.0f, 30.0f, -14.0f, -100.0f, 55.0f, -12.0f}}},
  {"Controlm21", {{-200.0f, -55.0f, -14.0f, -100.0f, -30.0f, -12.0f}}},
  {"Controlm31", {{-300.0f, 30.0f, -14.0f, -200.0f, 55.0f, -12.0f}}},
  {"Controlm31", {{-300.0f, -55.0f, -14.0f, -200.0f, -30.0f, -12.0f}}},
  {"Control11", {{0.0f, 30.0f, -14.0f, 100.0f, 55.0f, -12.0f}}},
  {"Control11", {{0.0f, -55.0f, -14.0f, 100.0f, -30.0f, -12.0f}}},
  {"Control21", {{100.0f, 30.0f, -14.0f, 200.0f, 55.0f, -12.0f}}},
  {"Control21", {{100.0f, -55.0f, -14.0f, 200.0f, -30.0f, -12.0f}}},
  {"Control31", {{200.0f, 30.0f, -14.0f, 300.0f, 55.0f, -12.0f}}},
  {"Control31", {{200.0f, -55.0f, -14.0f, 300.0f, -30.0f, -12.0f}}},
  {"RF2", {{-245.0f, 800.0f, 122.0f, -500.0f, -800.0f, 120.0f}}},
  {"RF2", {{245.0f, 800.0f, 122.0f, 500.0f, -800.0f, 120.0f}}},
  {"RFouter2", {{-150.0f, 800.0f, 122.0f, -245.0f, -800.0f, 120.0f}}},
  {"RFouter2", {{150.0f, 800.0f, 122.0f, 245.0f, -800.0f, 120.0f}}},
  {"Ground2", {{-55.0f, 800.0f, 122.0f, -150.0f, -800.0f, 120.0f}}},
  {"Ground2", {{55.0f, 800.0f, 122.0f, 150.0f, -800.0f, 120.0f}}},
  {"Controlm12", {{-30.0f, 100.0f, 134.0f, -55.0f, 0.0f, 132.0f}}},
  {"Controlm12", {{55.0f, 100.0f, 134.0f, 30.0f, 0.0f, 132.0f}}},
  {"Controlm22", {{-30.0f, 200.0f, 134.0f, -55.0f, 100.0f, 132.0f}}},
  {"Controlm22", {{55.0f, 200.0f, 134.0f, 30.0f, 100.0f, 132.0f}}},
  {"Controlm32", {{-30.0f, 300.0f, 134.0f, -55.0f, 200.0f, 132.0f}}},
  {"Controlm32", {{55.0f, 300.0f, 134.0f, 30.0f, 200.0f, 132.0f}}},
  {"Control12", {{-30.0f, 0.0f, 134.0f, -55.0f, -100.0f, 132.0f}}},
  {"Control12", {{55.0f, 0.0f, 134.0f, 30.0f, -100.0f, 132.0f}}},
  {"Control22", {{-30.0f, -100.0f, 134.0f, -55.0f, -200.0f, 132.0f}}},
  {"Control22", {{55.0f, -100.0f, 134.0f, 30.0f, -200.0f, 132.0f}}},
  {"Control32", {{-30.0f, -200.0f, 134.0f, -55.0f, -300.0f, 132.0f}}},
  {"Control32", {{55.0f, -200.0f, 134.0f, 30.0f, -300.0f, 132.0f}}} };



//the first float is the intensity, the second is frequency for cos. 0 means cos(0) = 1
//since a linear program is possible, I don't need to specify any more control
std::map<std::string, std::array<float, 2>> program{
    {"RF1", {{40.0f,20.0f}}},
    {"RFouter1", {{0.114995f,0.0f}}},//0.1197f
    {"Ground1", {{-4.2f,20.0f}}},
    {"Controlm11", {{0.0f,0.0f}}},
    {"Controlm21", {{0.0f,0.0f}}},
    {"Controlm31", {{-2.0f,0.0f}}},
    {"Control11", {{0.0f,0.0f}}},
    {"Control21", {{0.0f,0.0f}}},
    {"Control31", {{-2.0f,0.0f}}},
```

Figure 4: A picture of the variables for the ion trap in our code

6 Implementation

Our project calls for the design of a kilo-qubit scale quantum computer design ("QC" or "KQC"), which we have made strides towards this semester. Because our project is only requesting a global design for this computer, we have already started working on smaller components. This will be a schematic / outline and not a physical example, even non-functional, so our final product will be a 2d or 3d schematic with accompanying text. We have an existing global-scale node drawings (pictured in 4.3.1), but that is predicated on the ion handoff between traps being viable. We will continue to simulate our ion handoff between 1+ traps and model the physics that would underpin our quantum computer design's functionality.

7 Professional Responsibility

This discussion is with respect to the paper titled " Contextualizing Professionalism in Capstone Projects Using the IDEALS Professional Responsibility Assessment", *International Journal of Engineering Education* Vol. 28, No. 2, pp. 416–424, 2012

7.1 Areas of Responsibility

Area of responsibility	Definition	NSPE Canon	Comparison to Other Codes	Importance in Our Design	Team's Current Performance
Work Competence	Perform work of high quality, integrity, timeliness and professional competence.	Perform services only in areas of their competence; avoid deceptive acts.	NSPE focuses on the public and how you interact with others such as caring about others safety, health, and being truthful to others. There are lots of similarities with the other codes.	High, as a design in quantum computing, by its nature, should be technically competent. This is probably the most important area of our project, as our design is seeking to do something novel and difficult.	Medium. We are not experts in this field, but are slowly learning. I suspect that over time, this will increase.
Financial Responsibility	Deliver products and services of realizable value and at reasonable costs.	Act for each employer or client as faithful agents or trustees.	NSPE does not directly talk about finances and any rules implicitly apply. The other codes are pretty similar. Things such as honesty and competence would mainly	Low. Our design will utilize existing technologies and has a high likelihood of not being physically constructed. Financial utilization will	Low to N/A. We have not even considered the cost of any of these components, and have not fully investigated whether some concepts are reasonable or financially viable.

			apply to finances.	be low.	
Communication Honesty	Report work truthfully, without deception, and as understandable to stakeholders.	Issue public statements only in an objective and truthful manner; avoid deceptive acts.	NSPE and other codes talk about honest communication with employers and the public.	Medium. Our computer will most likely never be physically constructed, but not willfully neglecting faulty parts of our design is important to our success.	N/A, design has not been started, so there's no willful ignorance to be perpetrated.
Health, Safety, Well-Being	Minimize risks to safety, health, and well-being of stakeholders.	Hold paramount the safety, health, and welfare of the public.	Health and safety are by far the most important things listed for all codes.	Low. Again, not a physical computer, low real-world impacts.	N/A, not considered a priority so no work done.
Property Ownership	Respect the property, ideas and information of clients and others.	Act for each employer or client as faithful agents or trustees.	Engineers are expected to be honest with employers and not deceive them, which would mean to not steal their work from the company.	Medium. We are not using bespoke materials or technology, so crediting existing designs that may be utilized is important.	High. We have been very cognizant of existing designs and their incorporation into our design. We will be sure to credit them when the time comes.
Sustainability	Protect the environment and natural resources locally and globally.		All codes talk about sustainability for the public and health.	Low. Again, not a physical computer, low real-world impacts.	N/A, not considered a priority so no work done.
Social Responsibility	Produce products and services that benefit society and communities.	Conduct themselves honorably, responsibly, ethically, and lawfully so as to enhance the honor, reputation, and usefulness of the profession.	Engineers have lots of social responsibility with being honest and making statements when necessary.	Low. Again, not a physical computer, low real-world impacts.	N/A, not considered a priority so no work done.

Table 4: A list of professional responsibilities and how they pertain to us

7.2 Project Specific Professional Responsibility Areas

See above table

7.3 Most Applicable Professional Responsibility Area

Work Competence, see explanation in table above

8 Closing Material

8.1 Discussion

Our results of our project are some beginning steps of a theorized quantum computer design and preliminary simulations provided by a graduate student who is attached to our project. These simulations produced promising results for our design as we were successful in our simulation of an ion handoff between two radiofrequency segments. We have created some diagrams of our proposed innovations to the industry's current leading ytterbium ion trap quantum computer design and have acquired lots of information as as group. This is what we expected to have by the end of the first semester. During the next semester we expect to expand what we currently have into a formal and thorough design with more robust simulations. Once we are successful in a full scale real-to-life simulation, we plan on submitting a patent for our design. Overall, we have not achieved our final requirements. This is expected because we are only half way through the process. This semester, we have achieved what we expected to have done and have made really good progress on our final requirements.

8.2 Conclusion

We have acquired lots of information and created the beginnings of a quantum computer design. We also have a simulation for individual ion traps and a design for the wafers of the computer. Our goals are to have a fully fleshed out design of a quantum computer by the end of the class. The best way to achieve this goal is by sticking to the timetable and plan that we have set out and continuing to use the resources at our disposal such as our graduate student colleague. We did not run into any large issues achieving our goals that would stand out. This is because we set out good and reasonable expectations from the get go.

8.3 References

A. Shaw, “Simulating Ion Trap Quantum Computers,” *Researchgate.net*, Oct-2022. [Online]. Available: https://www.researchgate.net/publication/364316902_Simulating_Ion_Trap_Quantum_Computers. [Accessed: 01-Dec-2022].

D.-I. “D. Cho, S. Hong, M. Lee, and T. Kim, “A Review of Silicon Microfabricated Ion Traps for Quantum Information Processing,” *Micro and Nano Systems Letters - A SpringerOpen Journal*, 2016.

D. J. Berkeland, J. D. Miller, J. C. Bergquist, W. M. Itano, and D. J. Wineland, “Minimization of Ion Micromotion in a Paul Trap,” *Journal of Applied Physics*, vol. 83, no. 10, pp. 5025–5033, May 1998.

G. N. Nop, D. Paudyal, and J. D. H. Smith, "Ytterbium Ion Trap Quantum Computing: The Current State-of-the-Art," dissertation, 2021.

M. Heinrich, "On Stabiliser Techniques and their Application to Simulation and Certification of Quantum Devices," dissertation, University of Cologne , Cologne , North Rhine-Westphalia, 2021.

M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*. Cambridge: Cambridge University Press, 2021.

P. Maunz, *High Optical Access Trap 2.0*. Sandia National Lab, 2016.

T. Jones, "Mathieu's Equations And The Ideal RF-Paul Trap." Accessed: Dec. 01, 2022. [Online]. Available: http://einstein.drexel.edu/~tim/open/mat/mat.pdf

W. C. Burton and B. Esty, "Transport of Multispecies Ion Crystals Through a Junction in an RF Paul Trap," *arXiv.org*, 24-Jun-2022. [Online]. Available: arXiv.org. [Accessed: 01-Dec-2022].

Z. An and D. L. Zhou, "Deep Reinforcement Learning for Quantum Gate Control," *EPL (Europhysics Letters)*, vol. 126, no. 6, p. 60002, 2019.

Z. D. Romaszko *et al.*, "Engineering of Microfabricated Ion Traps and Integration of Advanced on-Chip Features," *Nature Reviews Physics*, vol. 2, no. 6, pp. 285–299, May 2020, doi: 10.1038/s42254-020-0182-8.

8.4 Appendices

8.4.1 Team Contract

Team Members:

1) Nicholas Greenwood
2) Jacob Frieden
3) Sam Degnan
4) Arvid Gustafson
5) Colin Gorgen
6) Emile Albert Kum Chi

Team Procedures

1. Day, time, and location (face-to-face or virtual) for regular team meetings:
   - **Thursday, 2pm, SIC 3222**
   - **Backup meetings in library if we get kicked out**
   - **Further meetings upon request**
2. Preferred method of communication updates, reminders, issues, and scheduling (e.g., e-mail, phone, app, face-to-face):
   - **Email chain, GroupMe**
3. Decision-making policy (e.g., consensus, majority vote):
   - **Majority vote, compromise, Heavy client guidance**
4. Procedures for record keeping (i.e., who will keep meeting minutes, how will minutes be shared/archived):
   - **Everyone takes notes on their section**
   - **Nick will head communication and management**

Participation Expectations

1. Expected individual attendance, punctuality, and participation at all team meetings:
    - **Everyone comes to every meeting they possibly can, be on time, participate as needed by the professor and team. Advanced notice of absence is expected.**
2. Expected level of responsibility for fulfilling team assignments, timelines, and deadlines:
    - **Expectations for presentation and solid deadlines should be minimal; as such meeting these expectations is easy, but paramount. Group contribution on team assignments necessary as needed**
3. Expected level of communication with other team members:
    - **24 hour or less response unless otherwise specified, communicate any issues, scheduling or otherwise.**
4. Expected level of commitment to team decisions and tasks:
    - **Majority vote as necessary in decision making, with proponents expected to make arguments on behalf of their argument. Tasks have been loosely defined/separated by client.**

Leadership

1. Leadership roles for each team member (e.g., team organization, client interaction, individual component design, testing, etc.):
    - **Nick is spearheading organization, communication, and anything client facing or concerning itself with group organization.**
    - **Any individual tasks or academic deliverables will be a group effort.**
2. Strategies for supporting and guiding the work of all team members:
    - **Consulting the team - we are giving weekly updates/presentations on our assignments which should serve to highlight our individual needs and allow for collaboration and guidance**
3. Strategies for recognizing the contributions of all team members:
    - **Keep note of who has done what - individuals will be presenting their learning on a weekly basis to the team and professors, which should be a good proxy for recognizing contributions**

Collaboration and Inclusion

1. Describe the skills, expertise, and unique perspectives each team member brings to the team.
    - **Sam: I have good programming skills which should be useful for error correction.**
    - **Nick: has worked in business and project management roles, has a general interest in modern technology**
    - **Colin: I have previous research experience in the realm of the miniscule and I've taken courses on quantum mechanics**
    - **Jacob: I have some background in applications of machine learning, which is a technique used in the evaluation of quantum hardware.**
    - **Emile : I have a strong understanding of electrical and electronic circuit analysis that will be crucial to understand the physical implementation of our work.**

- **Arvid: I am a seasoned programmer, and have a comprehensive understanding of discrete mathematics, contemporary operating systems, and classical computers down to the logic gate level. These strengths can help me better understand how quantum computers work, and what applications they might have.**

2. Strategies for encouraging and support contributions and ideas from all team members:
    - **Each team member will be learning a unique field which will require collaboration in order to be successful. Since we know what the others will be doing, when we each run into problems we can propose them to the group and try to pool our information together.**
3. Procedures for identifying and resolving collaboration or inclusion issues (e.g., how will a team member inform the team that the team environment is obstructing their opportunity or ability to contribute?)
    - **If anyone has an issue with their inclusion, they should alert the team in the GroupMe. At that point we will speak as a group and potentially involve Dr. Paudyal, Dr. Smith or Dr. Fila**

Goal-Setting, Planning, and Execution

1. Team goals for this semester:
    - **Good grade (an A), learn something new, Potentially get published / funding (for help in grad school applications for some team members)**
2. Strategies for planning and assigning individual and team work:
    - **Communication, assessing individual workloads and assigning as we see fit, taking into account people's specialities**
3. Strategies for keeping on task:
    - **Setting goals, Adhering to and enforcing deadlines. Other strategies to be learned and decided upon**

Consequences for Not Adhering to Team Contract:

1. How will you handle infractions of any of the obligations of this team contract?
    - **Set a clear deadline that they are expected to meet and exactly what they are supposed to do. This will give them the opportunity to make right with the group**
2. What will your team do if the infractions continue?
    - **Team Intervention and discussion of issue / ways to alleviate, then escalate to Dr. Paudyal or Dr. Fila depending on type of infraction given repeated infractions**

****

a) I participated in formulating the standards, roles, and procedures as stated in this contract.

b) I understand that I am obligated to abide by these terms and conditions.

c) I understand that if I do not abide by these terms and conditions, I will suffer the

consequences as stated in this contract.

1)	Nicholas Greenwood	DATE: September 15, 2022
2)	Jacob Friden	DATE: September 15, 2022
3)	Sam Degnan	DATE: September 15, 2022
4)	Arvid Gustafson	DATE: September 15, 2022
5)	Colin Gorgen	DATE: September 15, 2022
6)	Emile Albert Kum Chi	DATE: September 15, 2022