# MIDI Zeusophone

PROJECT PLAN

Team Number: SDMAY19-11
Client: Dr. Joseph Zambreno
Advisor: Craig Rupp

Team:
Jacob Feddersen - Communications Specialist
William Brandt - Pulse Width Modulation Expert
Luke Heilman - Technical Architect
Gregory Harmon - Tesla Coil Construction Expert
Leo Freier - Interrupter and Microcontroller Lead
Gunnar Andrews - Webmaster

Team Email: sdmay19-11@iastate.edu
Team Website: http://sdmay19-11.sd.ece.iastate.edu

Revised: 09-19-2018 - Version 1.0

# Table of Contents

## List of Figures



## List of Tables



## List of Symbols

## List of Definitions

MIDI: Stands for Musical Instrument Digital Interface. It is a technical standard for playing sounds through a digital interface. MIDI can also refer to the file type that computers use to play sounds based on the MIDI standard.

EcpE: Electrical and Computer Engineering. Usually refers to the EcpE Department at Iowa State University, which includes Electrical, Computer, and Software Engineering

PPE: Personal protective equipment.

Tesla Coil: A resonating transformer circuit that produces very high voltages, generating arcs into the air.

DRSSTC: Double Resonant Solid State Tesla Coil - a tesla coil design which can be modulated, producing audio

Zeusaphone: A special Tesla coil that releases voltages at specific frequencies, creating sound like a musical instrument

WAP: WiFi Access Point

# 1 Introductory Material

## 1.1 Acknowledgement

The MIDI Zeusaphone team would like to extends thanks to our client Dr. Joseph Zambreno for providing the project, as well as the full financial support and other technical assistance during the project. The team would also like to thank our advisor Craig Rupp for being a reliable expert on the subject matter, being a professional mentor for the team, and always being available for us.

## 1.2 Problem Statement

When prospective students are given a tour through Iowa State, they are shown the accomplishments and senior design projects of past undergrad students. The Electrical and Computer Engineering Department currently has two inoperable arcade cabinets that were constructed by previous electrical and computer engineers. In order to continue attracting students to ECprE, the department needs a new showpiece to demonstrate what prospective students could be capable of if they choose to attend Iowa State.

Our solution to this problem is to construct a Tesla Coil that plays music, also called a Zeusaphone. The Zeusaphone will be able to play preset songs as well as have the ability to be played with a piano keyboard so that prospective students are engaged with the demonstration. Because it will be shown on tours, an operating manual will be written to ensure the operator is using the Zeusaphone properly. A safety manual, proper signage, and proper personal protective equipment (PPE) will also be provided so that no injuries occur when the device is in operation.

## 1.3 Operating Environment

The MIDI Zeusaphone will always be demonstrated indoors. It will be stored in Coover Hall and will be operated in the same place. There is no threat of moisture since it will not go outside. There may be a problem with dust build-up if it is stored for an extended period of time, but this can easily be handled by quickly dusting the project off or blowing the dust off.

## 1.4 Intended Users and Intended Uses

As the goal of the MIDI Zeusaphone is to be a showcase item for the EcpE Department, the operator of the Zeusaphone will always be a faculty member of the EcpE Department. However, the operator may not always be someone with previous knowledge or operation experience with the device. Therefore the MIDI Zeusaphone should be designed with simplicity and intuitive operation in mind.

The MIDI Zeusaphone will be used in demo scenarios in front of an audience. This audience could be a small private viewing or a large demo in front of a lecture hall.

### 1.5 Assumptions and Limitations

Assumptions:

On Usage

- The operator will be able to play a MIDI keyboard to produce sounds
- The operator can play pre-loaded MIDI songs to play through the web client.
- The operator can load MIDI songs through the web client to be played later.

On Safety

- The primary use of the Zeusaphone will be as a showcase item.
- The operator will be fully aware of the safety considerations and proper use of the Zeusaphone.
- During operation, all safety standards will be followed by the operator and the audience.
- When not being shown, the operator assumes responsibility as laid out by the provided safety standards.

On Reliability

- The system can be safely stored in any room safe enough to store high voltage circuits.
- The full project will be able to be reliably moved to and from storage with minimal assembly and disassembly
- Improper input will not result in a dangerous situation.

Limitations:

- The end product will be no larger than 2 ft tall with a 1 x 1 square foot area
- It must be able to be run off of a wall outlet. (120V 60Hz)
- Can only play two different tones at once
- Operators must be associated with the EcpE Department.
- The Tesla coil will only be able to be activated using the project interfaces.

### 1.6 Expected End Product and Other Deliverables

- MIDI Zeusaphone (May 2019)
    - This will be the final product of our project. This will include a Tesla coil or coils that will play frequencies to make music while electricity arcs out of them. This will all be made by us and programmed by us. This device will be portable and easy to work so it can be used by a large number of people.
- Operating Manual (May 2019)
    - This will be a very detailed guide for working the zeusaphone. It will include all the steps to turn on the zeusaphone and make it play through all of the different interfaces. This manual will also include extensive safety details, so that whoever handles the project will know exactly what steps to take in order to ensure that the zeusaphone is operated safely. Finally the operating manual will also explain how to play the zeusaphone through all of the available interfaces (keyboard, MIDI, bluetooth, etc).
- Keyboard (May 2019)
    - Along with the zeusaphone a keyboard will be provided. There will be instructions inside the operating manual on how to connect the keyboard to the zeusaphone. This keyboard will be used to make music through the Zeusaphone.

### 1.7 Previous Work And Literature

Zeusaphones already exist - they form a niche in the commercial market. There is also a relatively large hobbyist community centered around building these devices. There are even a few that are also controlled by MIDI keyboards. However, most designs only load songs or actually have the songs written into the code. These designs are common because the engineer and creator of the project will also know how to write the code that will create the tones. However, our team's design will be more reliable and easy to use for the operator. This way, someone who does not have extensive knowledge of how the system works can still play notes through the keyboard and songs through the user interface.

While the general principles behind the physics of a tesla coil are fairly well understood, building one can be tricky and take some time for anyone new to the field. Thus, we will be primarily following the work of Steve Ward (http://www.stevehv.4hv.org). He has experimented extensively with tesla coils, including musical variants, as well as a range of other high voltage devices, and has made his research and designs available for the community to analyze and follow in their own designs.

# 2 Proposed Approach and Statement of Work

### 2.1 Objective of the Task

In this project, the team will construct from scratch a working tesla coil. This tesla coil will be controlled through software running on a Raspberry Pi. The Raspberry Pi will allow control of the coil through a MIDI keyboard located a safe distance away, as well as through a web interface accessible to the presenter from a WIFI access point hosted from the Raspberry Pi.

### 2.2 Functional Requirements

- Power the Tesla coil with a standard 120V, 60Hz plug - this allows the coil to be easy to set up by not having special power requirements
- Create electric arcs that are at least 10 inches long - the arcs need to be big enough to be seen by people sitting in the back of a lecture hall
- Coil will play two notes at the same time - being able to play two notes at once allows the coil to play more complex music
- MIDI Keyboard will play the coil in real time - there should be no significant delay, otherwise it will be difficult for the keyboard player
- MIDI files can be loaded onto the microcontroller and played by the coil - this allows a presenter to easily show off the coil without musical experience

### 2.3 Constraints Considerations

Non-Functional Requirements

- The total cost of the project should be at most $1,000
- Safety cannot be compromised for functionality
- Size constraints - no taller than 2 feet, with a 1 x 1 square foot area
- Easy to move and setup
- Reliable for demonstrations

Standards/Protocols

- Code documentation will follow Doxygen standards - this will allow anyone to read our code and easily understand the comments - it will also let us generate API documentation automatically
- Keyboard and files will follow MIDI standard - the MIDI standard has existed for a long time, and many electronic instruments use it. This will make the coil more flexible
- FCC standards for EM fields - The RF interference generated by the coil will be kept to a minimum, by either a faraday cage or other means.

- Following WiFi/WPA2 standards - the WiFi access point generated by the microcontroller will be protected by WPA2 security. This will prevent audience members from controlling it without the presenters consent.
- HTTP/HTTPS communication - By using these standard protocols, any device on the microcontroller's WAP which has a web browser will be able to connect to the control page. This will make it easier for the presenter to use it.

### 2.4 Technology Considerations

There are two main choices for the construction of a tesla coil: Spark gap and solid state. The Spark gap tesla coil is the original design. It involves a bright and high voltage spark through air while the required oscillations between the capacitor and inductor occur. The solid state tesla coil uses integrated circuits to mimic the oscillations that would occur in place of having the spark. We chose to create a solid state coil because we will have more control of the oscillating which will give us the ability to create the required music. Another perk of choosing solid state over spark gap is safety. An uncontrollable, dangerous, and bright spark occurring in the base of the coil would make testing a hazard if we were having to adjust the the circuitry or have measurement probes on the circuit.

The choice of microcontroller will have a noticeable impact on how the solution can be implemented. It ended up being a choice between an Arduino and a Raspberry Pi, with a brief look at an extremely low level microcontroller chip that was not chosen as it would be too complex to implement. Speed will be a big factor for the microcontroller's job, which leans the choice toward an Arduino, but a Pi was chosen since it should be able to achieve the speed needed and is a little easier to use. All of the microcontroller code will be written in C because it runs faster than Python, and because the system requires low latency. In the future, a different microcontroller may be used if the Pi poses issues or has more functionality overhead than is needed.

The main software needs some way to interact with the microcontroller that will control the coil's circuit. The microcontroller will create a WiFi access point which can be connected to through a web client, which can load and play MIDI files. This solution was chosen for ease of use. Operators should encounter little to no issues when demonstrating the Zeusaphone, and the design of the web client should facilitate use of the system.

### 2.5 Security Considerations

Since the microcontroller will be acting as a WiFi access point (WAP), it should be protected. This will be done with WPA2, since it will be easy to configure. This network will not be connected to the rest of the internet, since it does not need to be.

Where the device is stored should also be considered. It should be kept in a location where only people that are authorized/trained to use it can access it. This will help with safety issues as well.

## 2.6 Safety Considerations

While the MIDI Zeusaphone is designed to be entertaining to watch and listen to, there are several important precautions that must be taken in order for the system to be safe. First, it is imperative that the coil is off before it is plugged in and that everyone is kept from touching the coil while it is in operation. Second, the area around the coil must be kept clear to avoid being struck by arcs, unless the objects are properly grounded and intended to be arc targets. Special consideration is needed for those with hearing aids and pacemakers, as these devices could potentially suffer from electromagnetic interference produced by the tesla coil and should be kept a safe distance away. The capacitors in the system can hold a high charge for some time even after the tesla coil has been powered off. Finally, the tesla coil creates ozone and certain nitrous oxides. Prolonged use within an enclosed space must be avoided.

## 2.7 Possible Risks And Risk Management

We will attempt to use a Raspberry Pi running Raspbian for the microcontroller. Since this is not a real time operating system, it may be difficult to get the precise timing required to control the GPIO pin. If this proves to be an issue, we may need to get a separate piece of hardware to control this accurately.

Using the Raspberry Pi as a WAP may create an issue if it needs to be protected from the coil. Our original plan to put the Pi in a Faraday Cage will not work if we use it as a WAP. Along these lines, the WAP also introduces a new security issue to tackle.

Designing the interrupter as to not blow out the coil is a very real risk. Of course, this is a design requirement, but the team will need to be extra cautious when testing on the coil circuit. Our tests will need to be safe enough as to not destroy the coil while the circuit is still being tested.

Design of the Zeusaphone will involve researching different methods of designs and constructions. The varying designs found can possibly skip certain safety precautions so in order to ensure that the larger components don't fail and the first prototype and subsequent iterations work safely, simulations will be done before circuits are powered up.

## 2.8 Project Proposed Milestones and Evaluation Criteria

- MIDI translation software
  - The software will be able to translate MIDI into a usable format. It will be piped to the Raspberry Pi where it can read the notes being generated. There will be two ways to port the MIDIs: through a keyboard, and sending files to the Pi over its WAP.
- Driver software to convert digital signals to analog signals
  - Software running on the Pi will translate the signals from the translation software into analog voltage signals for the coil circuit. This can be completed without knowledge of the translation software by using a temporary input method.

- Construct prototype Tesla coil circuit
  - The coil circuit will be able to function as a tesla coil without any input from the other components. Safety and reliability are key in completing the coil. All three main circuits must be functioning as intended.
- Develop user interface
  - A MIDI keyboard will send signals to be played. This interface will need to be developed. A web application will also be able to load and select songs to play by interacting with the Raspberry Pi.
- Construct prototype Zeusaphone
  - Combining all previous milestones will create the system as a whole, where the user interface will interact with the driver which sends analog signals to the circuit.
- Testing Zeusaphone
  - Once a prototype is completed and base functionality exists, intensive testing should be performed by the team. During this testing, the primary objectives will be to correct reliability, performance, and safety issues with the prototype.
- Stretch Goals
  - Once the prototype is finished, the team could work on stretch goals such as multiple coils, different interfaces, or cleaning up the project's implementation. Stretch goals could be completed simultaneously with overall intensive testing, depending on when the final product is finished.

### 2.9 Project Tracking Procedures

To track progress and report it to the client, the team will complete weekly status reports detailing research conducted and progress made, as well as obstacles and hurdles that they faced. These reports will be delivered to both the client and the advisor for review. In addition, weekly meetings will be held with the advisor to discuss progress that has been made and discuss design challenges and potential solutions.

### 2.10 Proposed Design

The overall design of the system (Figure 1) is to pipe MIDI signals into a microcontroller that will control the Tesla coil. A MIDI keyboard, when played, will send MIDI messages wirelessly over Bluetooth. These messages will be received by the microcontroller, which will translate these into on/off signals for the interrupter. The interrupter will respond to these signals by pulsing the power of the Tesla coil on and off, thus creating voltages at the right frequencies that are emitted as light and sound.

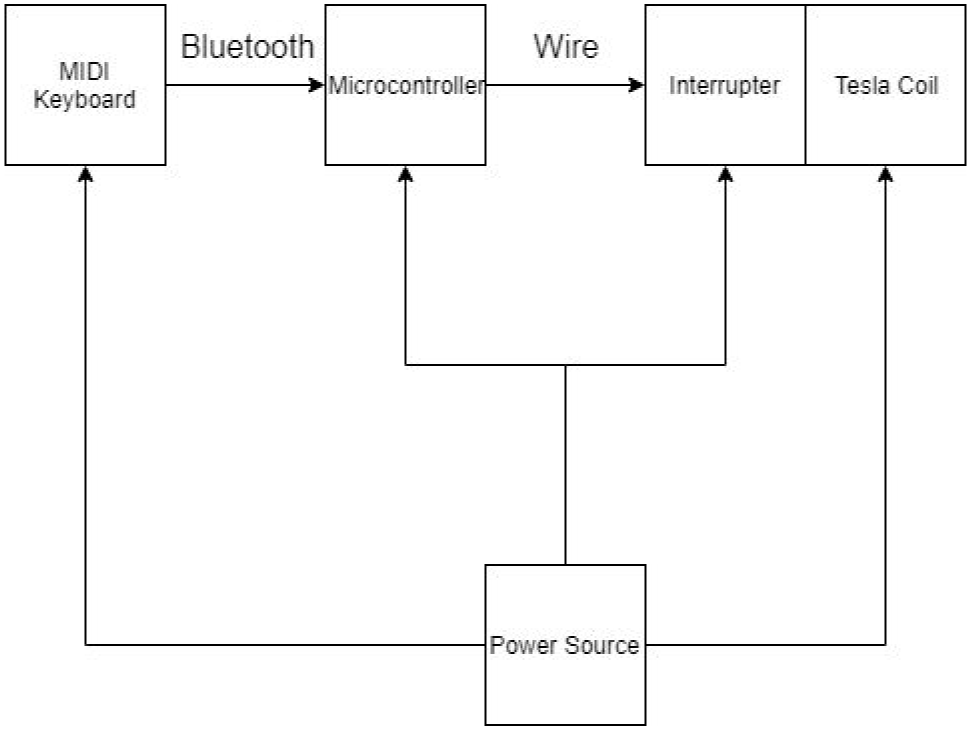


*Figure 1: Project Schematic*

All of these components will require power. They should be able to be supplied by normal 120 V wall plug-ins (even the Tesla coil itself).

We will use a standard, inexpensive MIDI keyboard to control the tesla coil because tone and sound quality are irrelevant in this application. The only inputs we need are key down and key up.

A Raspberry Pi will be used as the microcontroller because it is economical and easy to program. It easily supports the C language, which will be used due to its speed. It will interpret the MIDI messages it receives and translate them into output through one of its GPIO pins. This simple voltage output will be sent to the interrupter circuit of the Tesla coil.

The Raspberry Pi will also need to store sample MIDI files that can be played by the Zeusaphone. These files will be accessible and playable through a web console. The Pi will create its own WiFi access point, which when connected to will allow access to the web console. This access point will not be connected to the rest of the internet, since this would add an unnecessary security risk.

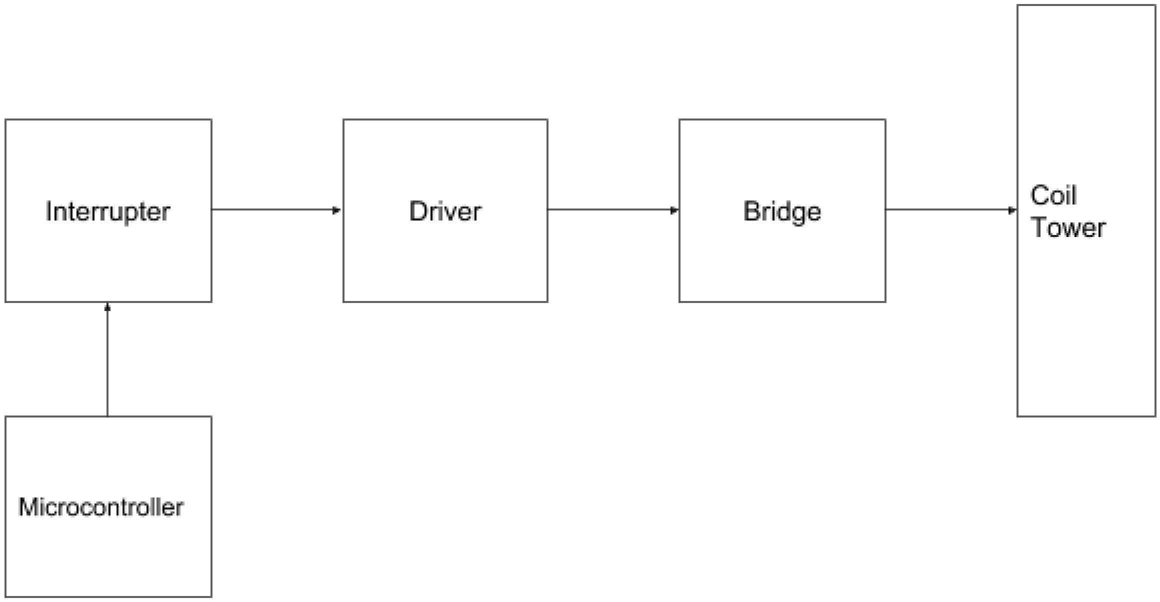


*Figure 2: Zeusaphone*

The Zeusaphone comprises of four main components: The interrupter, driver, bridge, and coil tower.

The interrupter gets the audio signal from the microcontroller and modulates it and sends it to the driver over a fiber optic connection. The driver will be using the logic from the interrupter and from a feedback control and overcurrent protection to create an output that will be sent to the bridge. Then the bridge will take the alternating signal from the driver and that allows the primary coil to link with the coil tower and step up the voltage hundreds of times.

The software will follow the general API shown below in Figure 3:

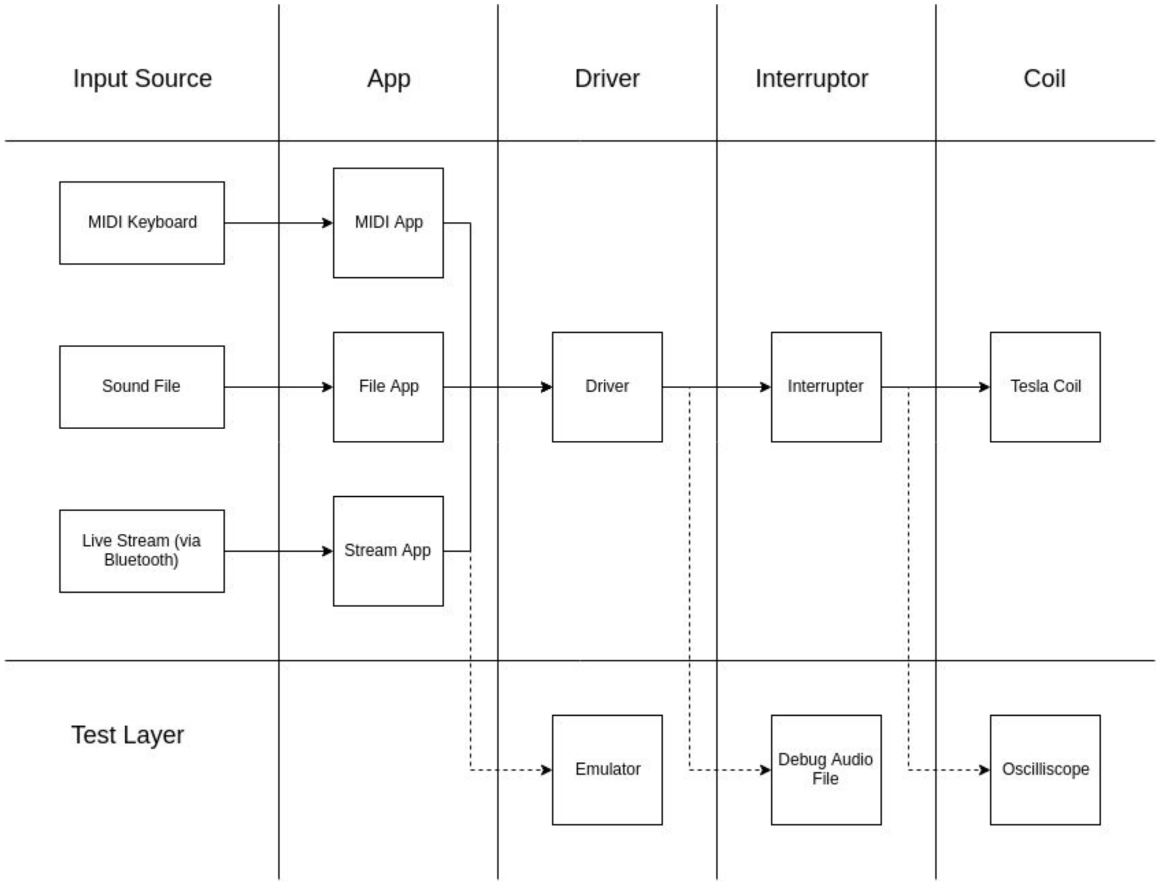


*Figure 3: Software Layer APIs*

Each layer in the software is designed to be modular and testable without the need of the next layer. This allows easier flexibility and testing of each component. An input source will send messages to an App on the microcontroller. This app will translate the input source into simple messages of turn this frequency on or off. These messages will be sent to the Driver layer on the microcontroller, which will determine the frequency to pulse the voltage on the GPIO pin.

There is an extra input source shown here - this is a audio input stream via Bluetooth that would come, for example, from a phone. This would allow the Zeusaphone to effectively be a Bluetooth speaker. This is a stretch goal, however, so it is not focused on much here.

### 2.11 Expected Results and Validation

The expected end product is a tesla coil which can produce an impressive visual and musical display for a live audience, playing both pre-recorded MIDI files and live input from a MIDI keyboard. The system should be simple to move, store, and set up, and it should be safe for all people present.

The tesla coil system should be capable of playing at least two tones simultaneously, to make the demonstration more interesting and exciting for the audience.

Input should be delivered through a wireless MIDI keyboard, located a safe distance away from the tesla coil, or from saved songs which are activated through the web interface.

Validation of individual components will be accomplished as described in the Test Plan below, through a combination of automated software testing where possible and manual testing otherwise.

### 2.12 Test Plan

The components of the MIDI Zeusaphone are modular, so testing of each component can be done individually. These tests can be both automated unit tests, run on every iteration of the software to prevent regressions, or manual tests to see how the pieces of the system respond under various input conditions.

The MIDI applications - reading from a MIDI file and from a MIDI keyboard - can be tested by a "driver emulator." This emulator would read the same input from the application as the real driver which controls the coil, but would interpret the input and output it directly to a sound file, which can be manually verified or compared against a predetermined scenario.

The driver layer can be tested using another emulator at the hardware level - outputting the high-low pin outputs into a sound file as with the driver emulator. Again this sound file can be listened to in order to verify the correct sound frequencies are being generated.

The microcontroller output can be verified in a lab using an oscilloscope to manually view the voltage output waveform that will be sent to the tesla coil. Most modern oscilloscopes provide simple tools to determine wave frequency and pulse width, which are the parameters we want to verify against our expectations.

The tesla coil can be tested without the microcontroller by using a function generator in the lab, set to produce the same output voltage as the microcontroller pin. In this way, the tesla coil can be manually tested to produce sound at a number of different frequencies before the microcontroller is connected.

Finally, when each component has been tested and works individually, we will connect all of the components and conduct end-to-end manual testing of the entire project.

# 3 Project Timeline, Estimated Resources, and Challenges

### 3.1 Personnel Effort Requirements

On our team, we have four Computer Engineers and two Electrical Engineers. We hope to balance our time accordingly between hardware and software tasks. We currently estimate that around 575 hours of effort will be required to complete the tasks strictly related to the construction, programming, and testing of the MIDI Zeusaphone. This is a reasonable time estimate to be accomplished by the team in two semesters.

| Task | Description | Estimated Time |
|---|---|---|
| Research Tesla Coil Designs | Research needed to evaluate competing tesla coil designs and determine which will be used | 20 hours |
| Research Tesla Coil Implementation | Research needed to determine the parts and design details of tesla coil | 40 hours |
| Research MIDI File Format | Research needed to understand the concepts of how music is transferred in MIDI format | 5 hours |
| Develop MIDI emulator | An emulator which can synthesize MIDI events to sound is needed for testing | 5 hours |
| MIDI File Reader | Develop a program which can read MIDI files and extract the note events and timing | 5 hours |
| Build Prototype Tesla Coil | Build an initial prototype tesla coil model, based on parts determined from research | 50 hours |
| Build Driver Circuit | Build the control circuit for the tesla coil | 50 hours |
| Develop MIDI Keyboard Application | Develop a program which reads live input from the MIDI keyboard | 20 hours |
| Develop Tesla Coil Driver | Develop a program which translates MIDI events into an output signal to control the tesla coil | 50 hours |
| Develop Raspberry Pi Configuration | Configure Raspberry Pi for automatically running software, enabling remote login and control | 10 hours |
| Develop Raspberry Pi Web Interface | Build a web interface to play, load, and delete MIDI files from the Raspberry Pi | 20 hours |
| Hardware Testing | Test and tune the coil for optimal performance | 100 hours |
| Software Testing | Test the software through many scenarios to validate performance and correctness | 100 hours |
| Safety Documentation | Document safety concerns and compile a manual for the tesla coil | 100 hours |

*Table 1: Personnel Time Requirements*

### 3.2 Other Resource Requirements

There are a number of hardware requirements for this project. As the hardware components list continues to develop, the team will inform the client for approval of all cost changes, to ensure that the project remains within financial limits and funds are being used appropriately.

Most of the hardware required will be used toward the construction of the tesla coil itself. Materials are required to build the main structure of the coil as well as the circuit to drive it.

Costs listed for each item are the approximate expected costs, found through comparison of products online. We looked for suppliers and components which would minimize the overall cost while remaining adequate for the quality and safety requirements of the project. There may also be shipping and handling costs for some components depending on where they are sourced and how they are obtained.

| Reference No. | Item | Cost Per Unit x Number of Units | Total Cost |
|---|---|---|---|
| 1 | Raspberry Pi Model 3 B+ | $35.00 x 1 | ~ $35.00 |
| 2 | 2.5A Power Adapter for Raspberry Pi Model 3 | $9.95 x 1 | ~ $9.95 |
| 3 | MIDI Keyboard | $199.99 x 1 | ~ $199.99 |
| 4 | Custom-build Midi Coil | ~$200 | ~200 |

*Table 2: Hardware Cost Requirements*

### 3.3 Financial Requirements

The only financial requirements for the construction of the MIDI Zeusaphone are listed in the table above. All of the software required for the project can be either developed by the team or found in open source or freely available software. The total cost of the system is anticipated at this time to be ~$444.94. This leaves us with plenty of our ~$1000.00 budget to spare as the needs and hardware requirements of the project develop.

## 3.4 Project Timeline

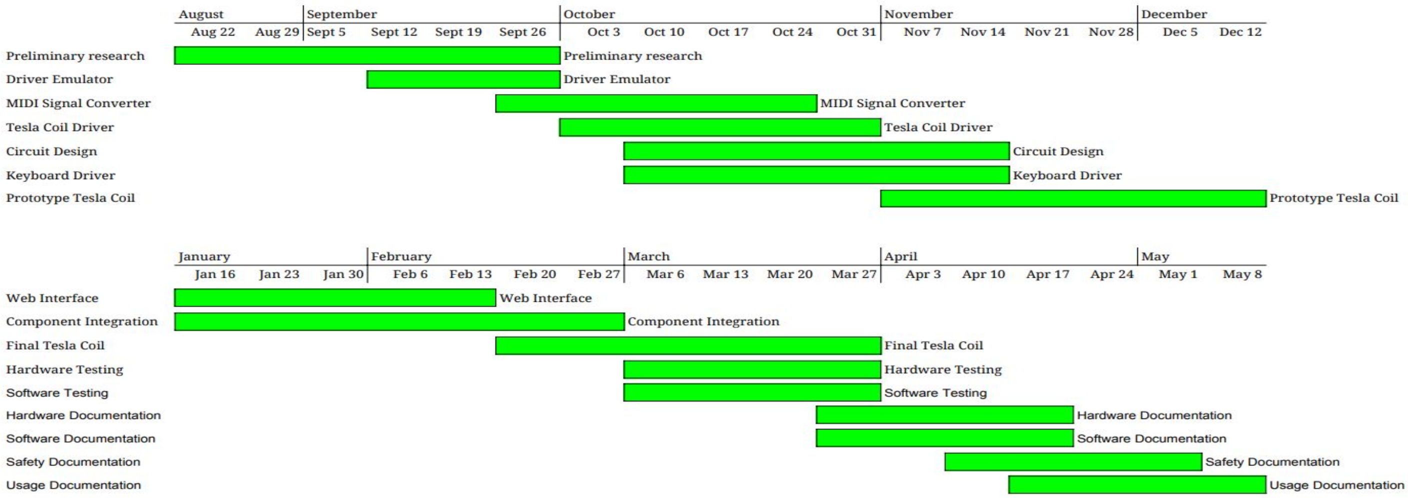


*Figure 4: Proposed Project Timeline*

The proposed timeline for the first semester is a plan for developing a prototype by the end of the semester. This will put us in a good position to test and develop potentially better methods during the second semester so that the end result works at peak efficiency. Our plan starts with basic research designed to familiarize ourselves with the project, and other people's work so that we a starting point for our own design. Moving forward we work on developing the software used to convert signals to midi as well as run them on the tesla coil, and the actual hardware of the tesla coil. Once these are complete we will be able to construct our prototype by the end of the semester.

The proposed timeline for the second semester has much to do with the usability of the tesla coil, and is best worked on after we have our working coil. It will also be for finalizing our design. We will develop the interface for the tesla coil as well as test and integrate each component into the final design. Once we have our final tesla coil we will work on the creating and finalizing any documentation needed so that the tesla coil can be operated and maintained by the client.

## 3.5 Feasibility Assessment

Designing the system in a modular way helps with the feasibility of the project as a whole. Each part can be designed and tested separately in order to achieve the most reliable system possible. The whole team is dedicated to seeing the project completed better than anticipated. Financial costs are low, providing one less barrier for completion. Many other projects similar to this have also been completed, even by one person alone. Of course, our Zeusaphone should have more functionality and reliability than a hobbyist's project, but the point stands that this is not necessarily treading new territory.

Risks do exist in the development, many to do with safety and reliability.  The Tesla Coil needs to be reliable when being controlled by the microcontroller.  Keeping the arcs on for too long can overload the circuit and break the system.  Preventative measures will need to be taken in order to assure this accident does not occur.  Fail safe off switches can be implemented into the driver program, the circuit can have limiting aspects, or the driver could be designed to never bring the circuit close to overload.

A Tesla Coil is dangerous when handled improperly.  Safety is always forefront in the team's collective minds and can be a huge roadblock in development.  The importance of safety comes from the system's purpose as a show tool.  The Zeusaphone should be an awesome show for an audience, but any accident is far less than awesome and could result in legal ramifications.  The absolute authority of safety in the project means no shortcuts can be taken in those regards.  For development, this means stretch goals or extra functionality could be discarded in order to assure safety.  It could potentially prove problematic in the normal design process.  However, it will be nothing the team cannot handle.

# 4 Closure Materials

## 4.1 Closing Summary

As our society grows more embedded with technology, we will need more engineers with an electrical and computer background. To attract more students to the ECpE department at ISU, a musical Tesla coil (Zeusaphone) will be created. This Zeusaphone will be playable both by a MIDI keyboard and by MIDI files stored on a microcontroller. The microcontroller will emit its own WAP, allowing the presenter of the coil to easily connect to it and control it. The dazzling displays from the Zeusaphone will inspire prospective students and encourage them to join the ECpE department.

### 4.2 References


Price References:

1. https://www.canakit.com/raspberry-pi-3-model-b-plus.html?cid=usd&src=raspberrypi&src=raspberrypi
2. https://www.canakit.com/raspberry-pi-adapter-power-supply-2-5a.html
3. https://www.amazon.com/Akai-Professional-LPK25-WIRELESS-Bluetooth/dp/B01J66AM0G/ref=sr_1_6?s=musical-instruments&ie=UTF8&qid=1537994088&sr=1-6&keywords=bluetooth+midi+keyboard
4. http://kaizerpowerelectronics.dk/tesla-coils/drsstc-design-guide/drsstc-faq-frequently-asked-questions/

Design References:

1. Steve Ward's Website: http://www.stevehv.4hv.org
2. Kaiser Power Electrics DRSSTC design: http://kaizerpowerelectronics.dk/tesla-coils/kaizer-drsstc-ii/

Software Libraries:

1. Midifile: https://github.com/craigsapp/midifile
2. Wavfile: https://www3.nd.edu/~dthain/courses/cse20211/fall2013/wavfile/

## 4.3 Appendices

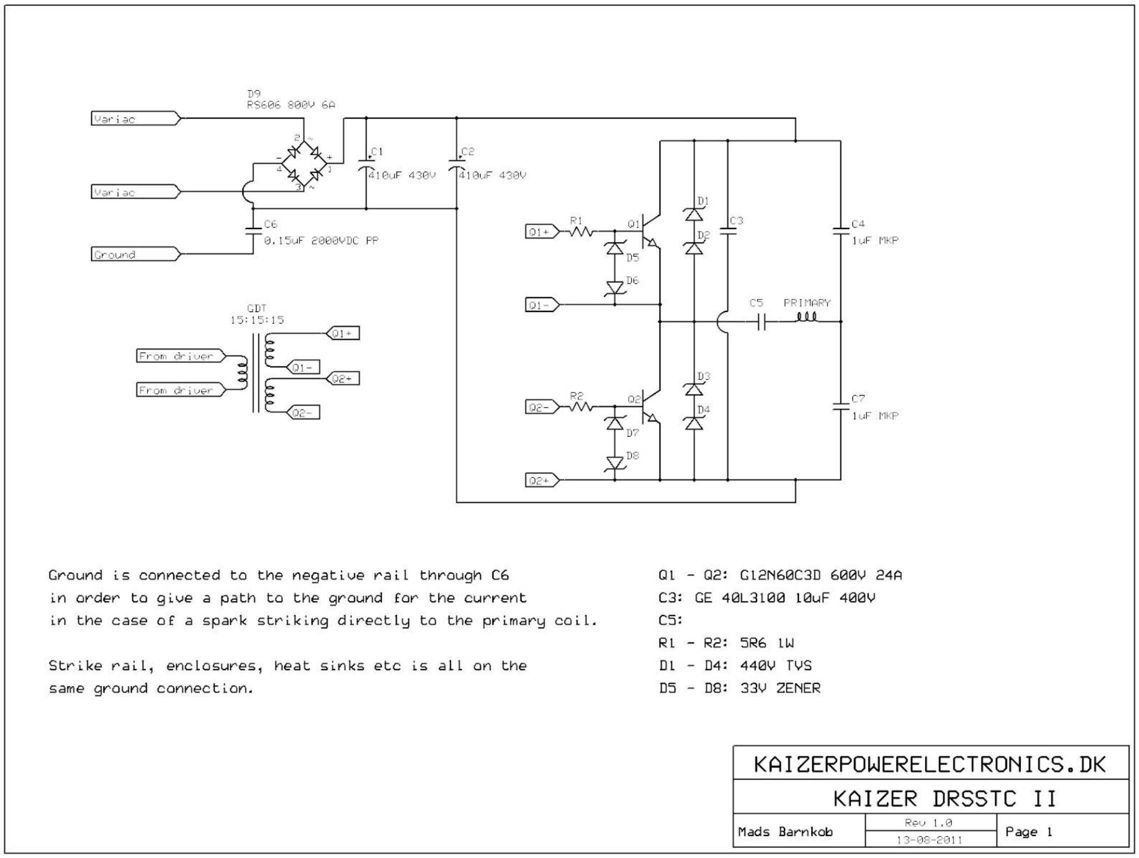


*Figure 5: Bridge Circuit Schematic, from Kaiser Power Electrics*