NESLA Coil

Team 32 -- Julian Goldstein, Payton Baznik, Shane Zhao
ECE 445 Project Proposal -- Spring 2018
TA: Zipeng Wang

Table of Contents

1. Introduction

1.1 Objective

In an effort to develop efficient methods of wireless power transmission, Nikola Tesla invented the Tesla Coil, a high voltage resonant transformer circuit that emits electromagnetic energy into the air. Although the Tesla Coil is not successful in achieving its intent to wirelessly transmit power, its lightning discharge effects provide a source of mesmerizing entertainment to observers. In recent years, engineers have redesigned the coil using solid state devices and signal processing techniques in order to use Tesla Coils to create a new form of novelty lightning music. Using the electrical breakdown phenomenon of air as a means of creating sound imposes limitations on the resolutions of sound the coil can emit. Coincidentally, this shortcoming is parallel to the producible sets of sounds that can be synthesized under the hardware constraints of the Nintendo Entertainment System. Thus, the Tesla Coil can be transformed into a clever accessory to the NES. Our objective is to extend the Nintendo Entertainment System's Audio Processing Unit's functionality beyond its designed limits, so that the NES' audio sub-systems are fully compatible with a musical solid-state Tesla Coil.

1.2 Background

The evolution of video game soundtracks is an auditory reflection of the overall advancement in developing microprocessor architectures and fabrication techniques over the past three decades. A revolution beginning with the release of 8-bit masterpieces like *Super Mario Bros.* and *The Legend of Zelda* on the Nintendo Entertainment System has made video game music an iconic, well appreciated, cultural symbol in the digital era. The computer hardware limitations of the 1980s meant that the NES could only synthesize a limited number of sounds composed of basic geometric waveforms, musical compositions that reflect these limitations have been termed "Chiptunes." Viewing the NES, not as a video game console, but rather as an instrument that produces chiptunes.

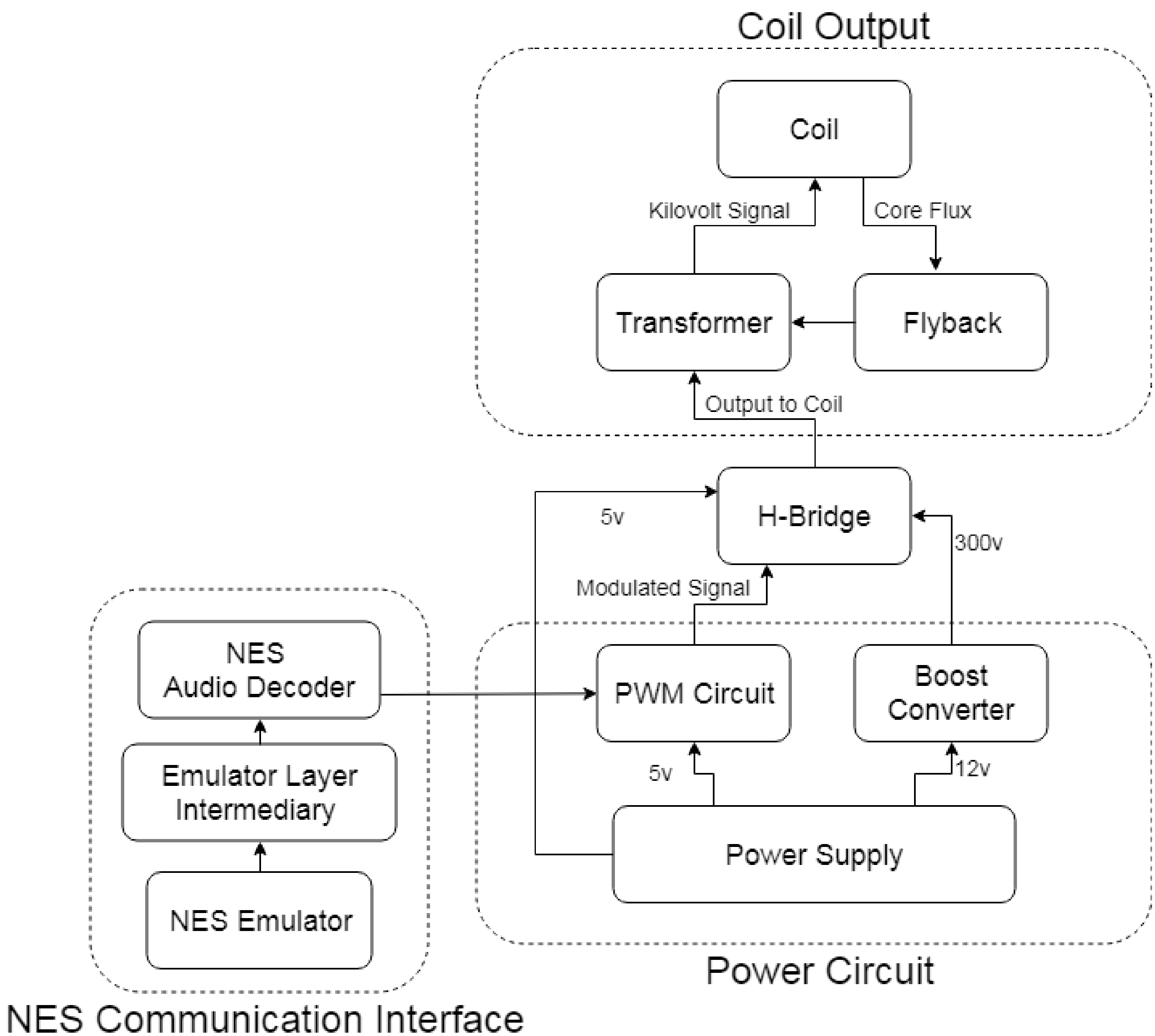
Because the NES revived the gaming world after the crash of the 80s, it became a cultural icon that signified the rebirth of video games. Because of this many people have created passion project devoted to the gaming system that started it all. In a way our project strives to do the same. By taking the classic sounds of the NES and transforming them into arcs in the air, we can create something unique that still make the NES the focus of the project. People will hear their favorite Super Mario songs, but through electricity itself. This is a way to praise the NES for its revolutionary impact on the gaming world but acknowledge it through an ECE perspective. It shows that engineering can still be used as an art form to express one's own love for something.

When people see the arcs of lightning, it will give them a new appreciation for something that they already knew about, the NES.

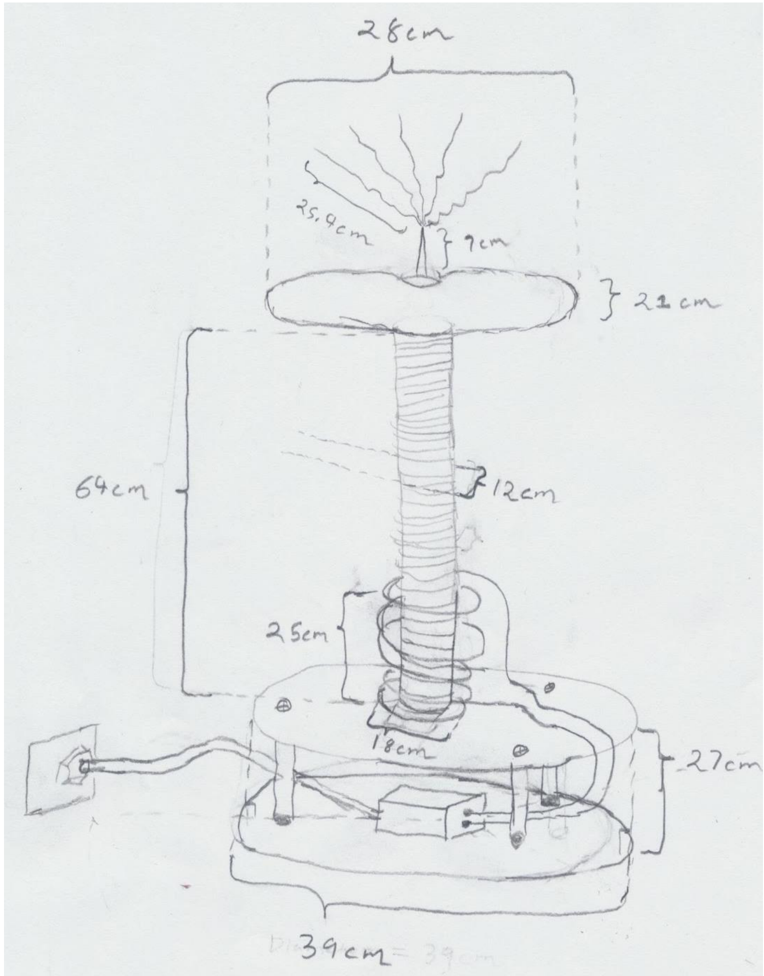
1.3 High Level Requirements

- The Tesla Coil's toroidal top load should be able to emit sparks between 8 to 10 inches in length.
- The Tesla Coil's electrical discharge should be able to emit sound waves that cover the entire set of audio waveforms from the Pulse 1, Pulse 2, Triangle and Noise channels on the Nintendo Entertainment System's APU.
- The Tesla Coil's audio modulation inputs should be designed in such a way that they natively interface with the NES APU. Meaning that as far as the NES APU is concerned, the Tesla Coil outputting the sound is equivalent to the standard Television Diaphragm speaker it was originally designed for.

2. Design


Coil Output
Coil
Kilovolt Signal
Core Flux
Transformer
Flyback
Output to Coil
H-Bridge
5v
300v
Modulated Signal
NES
Audio Decoder
PWM Circuit
Boost
Converter
Emulator Layer
Intermediary
5v
12v
Power Supply
NES Emulator
Power Circuit
NES Communication Interface

Physical Design



Above is a sketch with approximate dimension which reflects what the NESLA coil should look like once it is finished and operational. Note that auxiliary dashed lines were added to convey orientation in space, and three-dimensional axis that measurements were made along. All components other than the coil itself sit below two 39 cm diameter wooden disks which are separated from one another using three 27 cm wooden rods placed 120 degrees apart from one another near the perimeter of the disks. The rods should be orthogonal to both wooden disks. The top surface of the upper disk has support for the primary and secondary coils to stand orthogonal to the surface. The coil 64cm in height will be wrapped around PVC pipe which has a diameter of 12 cm. The top toroidal load interconnects with the wiring of the secondary coil and has a 7 cm high needle to concentrate the electric field lines and yield the point in which the electric discharge is emitted.

2.1 NES Communication Interface

The audio's data path begins within a Raspberry Pi software environment as emulated APU instructions encoded within NES ROM images, and ends with the intended APU sound being output as electrical discharge emanating from a Tesla coil. In this sub-module we will outline the processing steps required to transform a stream of discrete digitized NES ROM APU instructions into an analog, continuous sound amplitude signal.

2.1.1 NES Emulator

The NES emulator executing within the Raspberry Pi environment will read a NES ROM image and output the data to the Raspberry Pi's GPIO interface using the Wiring Pi GPIO library. The emulator should be able to output 8-bit deep sound amplitudes as a discretized vector sequence of eight GPIO pin voltages in real time.

Requirements	Verification
The Emulator must read NES ROM images and communicate with the Raspberry Pi's GPIO interface using the Wiring Pi library at the clock speed of the NES.	1. Set the appropriate linker flags in the Make File so that Wiring Pi's code is included in the emulator's address space. 2. Then write a function to strobe the GPIO pin voltages 3. Measure the pin voltages with an oscilloscope 4. Confirm the physical behavior matches the test strobe signal specified by software.
Must use the Simple Direct Media Layer library as a way of generating multimedia output.	1. View sections of the emulator's source code, specifically areas where the emulated APU is implemented. 2. Verify that the Simple DirectMedia Layer is integrated.
Must execute smoothly within the Raspberry Pi's Linux Environment and accurately reflect the runtime behavior of the NES.	1. Implement a software clock to measure the Emulator's cycle speed, and log clock measurements before and after critical sections. 2. Confirm that the interval displacing the two clocks approximates the same timing measurements made on the NES.

2.1.2 Emulator Media Layer Intermediary

The Emulator Simple DirectMedia Layer (SDL) Intermediary will provide a seemingly transparent interface between the Emulator's APU sound output stream and the memory buffer used by SDL to queue sound amplitude samples into the operating system's designated audio output peripheral. The intermediary will be transparent with respect to both the SDL library and the emulator, thus its presence should not affect their interaction. The purpose of this intermediary is to silently duplicate the stream of audio samples being transmitted from the APU into the audio buffer during the NES emulators run-time. The duplicated stream is then re-routed to a separate memory buffer. Within the emulator's initialization routine, we add instructions to map this memory buffer into the process address space and create a POSIX thread. The thread's purpose is to read from the mapped memory buffer either periodically or upon receiving a signal. When the thread's waiting conditions are not satisfied, it flushes the duplicated APU audio samples out of the intermediary buffer. Upon being flushed, the data is removed from the buffer and sent to the GPIO writing interface provided by the WiringPi library we link into the executable. By having a separate thread periodically schedule GPIO write operations, we can make sure audio samples are synchronously transmitted into the coil through the GPIO peripheral.

Requirements	**Verification**
Must function as a transparent intermediary interface located between APU's audio output and SDL's sound buffer.	1. Measure the performance cost of the intermediary by measuring clock displacement, every time the NES is asked to fetch an instruction. 2. Verify that the intermediary marginal cycle performance cost is 0.05 milliseconds or lower.
The Raspberry Pi environment must have the POSIX thread library available for linking at compile time.	1. Compile the intermediary with the linker flag -lpthread set. 2. Verify that the POSIX threads are supported, when compilation should proceed without returning an error.

2.1.3 NES Audio Decoder

The decoder module effectively functions as a digital audio converter (DAC). This module is input with the discrete real-time 8-bit sound amplitude signal encoded by the GPIO pin voltages and accurately reconstructs a continuous time analog. The produced analog signal must exhibit properties that are fully compatible with the operating conditions of the PWM module.

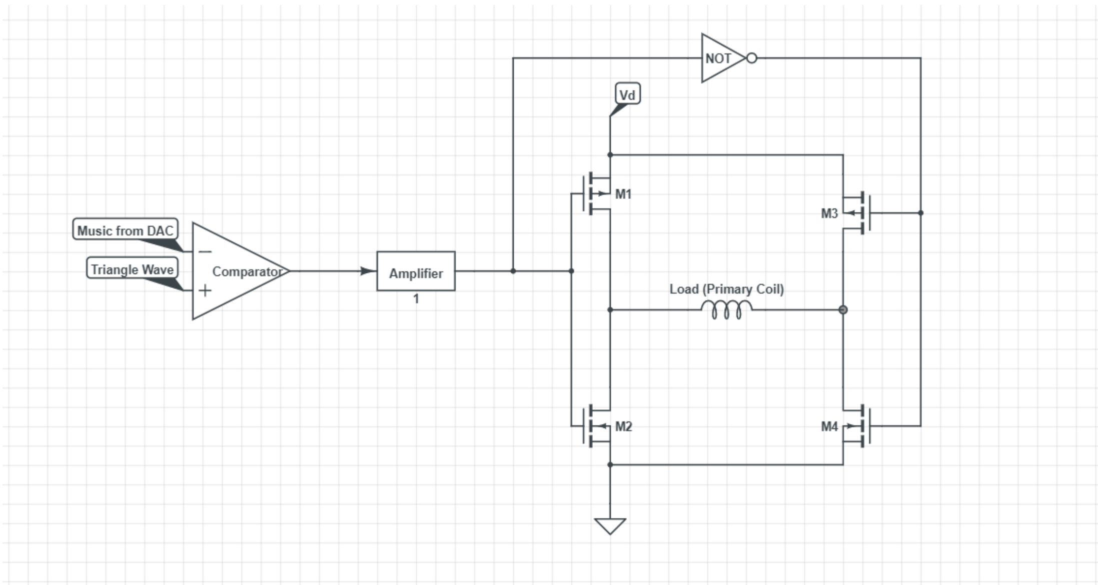
The 8-bit deep sound amplitude signal will output audio data at a rate 9,600 samples per second. The DAC IC driving the analog conversion must be able to operate at input frequencies between 1 kHz - 100 kHz.

Requirements	Verification
Must satisfy frequency operating conditions within the 1 kHz - 100 kHz to correctly sample audio data.	1. Directly reference the operational amplifiers datasheet and locate the Gain Bandwidth Product. 2. Check that the specified bounds form a valid subset of the frequency interval formed between 1 kHz and 100 kHz.
Must be able to produce continuous time analog signals of the digitized audio signal created by the GPIO pins.	1. Measure the discrete GPIO pin voltages and the reconstructed analog output signal and capture the behavior of both over a 10 second interval of time. 2. Create normalized plots of the two-time series on the same pair of axis and verify they only exhibit phase difference.

2.2 Tesla Coil Audio Signal Driver

2.2.1 PWM circuit

This block uses the basic form of a PWM circuit. We will create a switching circuit with our DC voltage that we created earlier in order to switch at a frequency high enough to mask our music signal coming in from the other section of our project. We will use NMOS circuitry in order to create our carrier frequency. We will then power the gates by using our new audio signal coming from our NES in order to modulate this carrier wave so when it comes out of the Tesla coil, you will be able to hear the music. A triangle wave rated within 10-15% of the Tesla Coil's resonant frequency is continuously applied to a comparator. The other input of this comparator will be our audio signal. Out of this comparator will be our gate signal that we will send to the MOSFETs. But before we do this we must lower the voltage output of our comparator. We do this by adding a capacitor at the end of the operational amplifier to make it so only the triangle wave will be output. This output will not be large enough to send to the gate, so we will put it through an amplifier to increase the voltage range of the triangle wave. Finally, we will send this wave into a gate driver to supply enough current into the gate to ensure that the gate turns on and off in a reasonable amount of time.



Requirements	Verification
Be able to handle a high enough frequency to operate at the resonance frequency of the coil	1. Find the resonance frequency of our coil 2. Confirm that the PWM circuit can operate at such a frequency
Be able to supply multiple amps worth of current into our gate (2-3 amps)	1. Select a gate driver based on our FET that we use 2. Run current through this gate driver using a bench power supply to make sure that it can run at the rated current for the FET

2.2.2 H Bridge

The H Bridge topologically consists of 4 high power MOSFETs in a Wheatstone bridge configuration symmetric about the primary side of the Tesla Coil's step-up transformer. The frequent on/off application of gate voltage rapidly switches the voltage source driving the Tesla Coil between the regulated high voltage DC power supply and ground. The gate signal is pulse width modulated, such that it encodes the NES audio signal but is carried at the resonant frequency. Encoding the sound through pulse width modulation specifies the resulting sound we observe.

Requirements	**Verification**
The MOSFETs must be able to handle at least 300 volts across source terminals without the occurrence of junction breakdown.	1. Apply 300 volts across the MOSFETs with a load that draws the correct amount of current to simulate the coil 2. Verify that MOSFETs behave correctly
The MOSFETs must be able to carry out the switching behavior within 10-15% of the Tesla Coil's resonant frequency.	1. Attempt switching the FETs at the rated frequency and clean the waves 2. Ensure that MOSFETs operate correctly at resonant frequency
The MOSFETs must be able to remain within their appropriate operating temperature ranges as specified by their datasheet.	1. Operate MOSFETs as specified during previous requirements 2. Verify using a heat gun on the FETs that the temperature does not exceed ratings

2.2.3 Fly-back Protector

A buffer circuit needs to be placed between the step-up air transformer's primary winding and the output terminals of the H bridge. The reason a buffer is required is because the rapidly alternating high voltage waveform being fed into the air-core transformer from the output of the H bridge attempts to exhibit jump discontinuities at the points of switching. Since the amount of magnetic flux circulating in the transformer coils over time must be differentiable, it decreases smoothly with time. As a result, a large spike of current can be induced in the opposing direction potentially fry our H bridge. This phenomenon is called current fly-back. By incorporating power diodes as a blockage mechanism against this hazardous fly-back effect, we can protect our coil driver circuitry from the damaging effects.

Requirements	Verification
The diodes used to reroute current flow must be able to handle the voltage applied upon the frequent occurrence of current fly-back without junction breakdown.	1. Operate the diodes in test cases where they are forced to deal with reverse voltage 2. Verify that the diodes are not damaged and function correctly
The diodes must maintain temperatures that are within their appropriate ranges as indicated by their data sheets (25-75 C) throughout the Tesla Coil's entire operating cycle.	1. Operate the diode under a 300 Watt load setting 2. Use a heat gun to ensure proper cooling of the system and temperature operation

2.3 Power Supply

This section of the design deals with taking wall voltage and converting it into a DC Voltage that we can use in our circuit. By doing this we then have the ability to power our many circuit parts, along with create the high voltage that we need in order to run the coil at 250W and 300V.

2.3.1 PSU

The power supply that we will be using is an off the shelf power supply. We will be using a 350 W power supply with 12v and 5v outputs. We will be using the ATX-350PN-B204 power supply, which will be able to use 12v, 5v, and 3.3v. The current coming out of these ports are rated very high, for example the 12v rail is rated for 18 amps. The 12v output will be used for our voltage going into the primary coil. Before we use this voltage, however, we must step this DC voltage up to around 300 volts, so we can get into the Kilovolts range when the voltage in the primary transfers into the secondary coil.

Requirements	Verification
Must be rated for 350 W	1. Check specifications given for the power supply 2. Ensure that it is rated for at least 350W
Must have access to a 5V rail, and a 12V rail	1. Check that the power supply has a 5V rail and 12V rail 2. Ensure both rails are operating at their respective voltages using oscilloscope
The 12V rail must be able to output 300 W	1. Test the 12V using a power resistor that can withstand 250W of power. 2. Ensure that the power supply will be able to supply its maximum current rating without overheating or varying voltage

2.3.2 Boost Converter

A boost converter will be used in our circuit to boost our 12V output from our power supply to get 300v out. The boost converter is a simple converter in that it takes the input voltage and applies a duty cycle to it to force its voltage to increase. It does this by powering up an inductor, then when the gate turns off the inductor will shoot its current into the load, creating an increase in voltage. A problem that we could run into would be our duty cycle for the boost converter will be too high and our boost circuit will break down. For a standard boost converter, you want to operate in the 20-80% cycle range. This is because we will be switching at around 200kHz, so the turn on and turn off time delay for the FETs will be long compared to the switching period. This means that if we operate at too high a frequency, our FETs will simply stay on, burning our FETs. If we ever exceed this range, then we would need to add a second boost after the first one in order to get the scaling that we need.

Requirements	Verification
This circuit must not operate in the 20-80%	1. Calculate the theoretical duty cycle,

duty cycle range, because a cycle past this will put excess strain on our FETs in the boost.	2. Ensure that it is outside the duty cycle range
The FETs used in this circuit must not exceed their current limits, the current will be above 5 amps	1. The FETs we use will be rated for 18A, and we will use a heat gun to constantly monitor the temperature of the FETs
The ripple of the output voltage must not exceed 2% under our coils' load	1. Verify using an oscilloscope to ensure the voltage ripple is less than 2% under coil load

2.4 Tesla Coil

2.4.1 Step Up Air Core Transformer

The air core transformer is responsible for creating the high voltage that produces the desired electrical discharge. A helical air core separates the primary and secondary transformer terminals. The secondary inductance in ratio to the primary inductance is set such that a voltage of several kilovolts in magnitude is induced in the secondary coil when the primary coil operates at 300 volts. We can achieve this by altering the shape and amount of turns composing the secondary. We also must pay attention to the distance between the concentric coils that make up the step-up air core transformer.

Requirements	Verification
The transformer should have an air core and step up the secondary terminal voltage 180-220 times the primary terminal voltage.	1. Look at our primary and secondary coil winding ratio to ensure a proper step-up voltage ratio 2. Measure the output voltage 3. Verify that the output voltage 180-220 times the input voltage

The transformer should be helical and the length between the inner and outer radii of the concentric coils should be small enough such that the primary and secondary coils magnetically interact, but far enough such that the primary and secondary coils electrically interact.	1. Measure all defining attributes of the transformer: like magnetizing inductance, DC resistance, primary and secondary line impedance, core resistance 2. Use these measurements to ensure that the transformer will not saturate
The wires used should be able to account for the skin effect when operating at our coils resonant frequency	1. Calculate the current density of copper at the switching frequency to ensure that we our not current saturating our wires 2. Use enough wires in parallel to allow enough space for the current to transfer through

2.4.2 Toroidal Top load

This part of the circuit acts as the Tesla Coil's capacitive load and is driven by the high voltage secondary winding. We will place a 7 cm high needle at the center of the toroid such that the electric field lines at the needle's tip converge to a point. By adding this geometric feature to our top load, we can be almost certain that the expected electrical discharge that creates the arcing effects is emitted from the point of highest electric field concentration. The toroid and needle should have aluminum covering its surface area in order to gather charge.

Requirements	Verification
The toroidal top load should function as a capacitor so it can store and release charge into the air. We want a capacitance in the range of 10-20 pF.	1. Measure the inductance and capacitance value of the coil, along with its parasitics 2. Ensure that we are seeing the capacitance granted by the top load
The Toroidal Top Load's aluminum coating should not noticeably deteriorate as a result of the electrical discharge.	1. Run the coil and look at the top load to ensure it does not break down. 2. After resting for an hour, re-run and verify that it still functions.

The toroidal top load should be the region of emission for the desired electrical discharge creating the arcs from 8-10 inches in length.	1. Run the coil 2. Ensure that the arcs produced are visible at the toroidal top and around 8-10 inches in length

2.5 Thermal Control Block

2.5.1 Heat Sink

This component is primarily concerned with the thermal regulation of the H Bridge and the Fly-back Protector. The heat sink functions by enclosing the heat producing devices, namely the power MOSFETs and diodes, and serves as a thermal energy deposit for the excess heat energy produced. A 12 V DC fan will be integrated into the heat sink, such that device cooling occurs through convective heat transfer.

Requirements	Verification
The heat sink must regulate device temperatures such that they stay within their appropriate ranges as indicated by their data sheets throughout the entire Tesla Coil operation cycle.	1. Measure the temperature of all of our sensitive equipment for a 2 minute test at full load and ensure that they stay within specified temperature ranges
The heat sink should cover the devices we want to thermally regulate.	1. Verify that the heatsink has a large surface area with fins on the outside to ensure maximum heat transfer

2.6 Tolerance Analysis

The block that will cause the most problems in terms of its complexity will be the PWM circuit that masks our audio signal into our carrier signal. This requires an H bridge circuit that will be switching at a high frequency. Because of this we will need to make sure that our copper wires will be able to handle the current being sent into it. The skin depth of copper at that frequency may force us to use multiple wires to carry our signals. We also need to worry about taking our 4 different audio channels that come from the NES and adding them together into audio signal to be sent into the PWM circuit. This may be problematic and require some added circuitry in order to make sure that the audio signal that comes out will be clean to the ears. This also requires us to make sure that the carrier frequency is fast enough to get the level of definition that we want out of our NES audio. This added switching loss that may require us to have some form of cooling system in our circuit to make sure that the FETs do not burn up when they are switching. Not only that, but in order to get FETs that can switch relatively fast and can handle large amounts of current will be a potential problem. It's often hard to find FETs that can handle a lot of power but still have the ability to switch fast without extreme switching losses.

The tolerance of our passives will play a large part in our design as well. Because we are switching power FETs at a above 50kHz, we will need to cancel out the interference of the output. In order to do this, we need to use the resonance frequency of the disturbance and calculate the proper capacitance and resistance we need to cancel. But because our resistors and capacitors are rated to vary from 5-20% error, we will need to measure the values of the resistors and capacitors to make sure that we are getting as close to the rated frequency as possible to cancel out the interference.

3. Costs

Our fixed development costs are estimated to be $35/hr, 10hr/week for three people. We are spending approximately 15 weeks in this class working on our design. This equates to **$39,375**.

$$\boldsymbol{\$35/hour * 10hours/week * 15\,weeks * 3\ people * 2.5 = \$39,375}$$

Our parts and manufacturing prototypes costs as follows:

Item	**Model**	**Price**
Power supply	ATX-350PN-B204	$45
Power Components	-	$20

4. Schedule

Week	**Shane**	**Julian**	**Payton**
2/19/18	-Complete initial design	- Obtain a time series of	- Work on the design

	of DAC -Work on Design document	APU audio samples produced by specific NES ROM files at run-time. - Use the obtained sequence of audio samples, to create LTSPICE WAV files that can be used as test input signals for simulating the Tesla Coil's drive circuits.	document power sections - Begin to model our PWM circuit within LTSpice - Purchase '741' op amps in order to test the triangle wave generator circuit
2/26/18	-Model the DAC design in SPICE software to measure performance -Look at IC DAC and compared performance across different vendors and specifications	- Analyze the software interface between the NES APU's sound output, and SDL's audio output buffer over a variety of open-source NES emulators. - Recognize the design patterns implemented and begin prototyping the intermediary layer. - Understand the locking mechanisms needed to synchronize multi-threaded access to SDL's sound output buffer. - Create a non-blocking sound buffer read interface that can provide a continuous stream of real-time audio samples.	-Work on the design of the PCB layout for the power circuit -Look at MOSFETs that we could be using and select the parts that will work best with our design -Work on the triangle wave generator on a breadboard to get it working
3/05/18	-Do final simulation of DAC -Work with Julian to simulate input values into DAC simulation that are similar from that output of the GPIO	- Assist Payton in producing accurate LTSPICE simulations to test the acoustic constraints of sounds produced by electric arcs.	- Enhance my LTSpice simulations to account for the rated values of my FETs that I Choose -Work with Julian to help model our Tesla coil in the simulation

		- Create LTSPICE models that abstracts the audio transfer characteristics of the Tesla Coil into a linear time invariant circuit.	
3/12/18	-Purchase requirement parts for the DAC -Test the circuit using oscilloscope and simple input values	- Use the previously collected acoustical simulation data and devised audio transfer function models to identify and insert appropriate filtration cascades along the signals input/output path.	-Purchase the required parts for my power design, ensure that I need to use 2 boost converters for my 12v output
3/19/18	-Test power performance of circuits	- Obtain the Raspberry Pi and port the prototyped SDL/Emulator intermediary to the Linux Environment. - Perform clock measurements on various open source emulators that make use of SDL and use this data to decide which emulator candidate will perform best within the Raspberry Pi environment.	-begin testing my power components, run basic tests -These tests will be used to see if I need to add heatsinks onto my FETs for not -potentially buy heatsinks as needed
3/26/18	-Test the DAC with the GPIO interface and ensure correct functionality	- Integrate the Emulator/SDL intermediary into the best performing emulator, and modify the emulator's Make File, so that the compiler knows to link in WiringPi. - Use WiringPi to implement a non-blocking write interface that can output a real-	- Fully build the converters and the H-bridge, this will begin the power circuits main testing phase

		time stream of audio data to the GPIO pins.	
4/02/18	-Analyze results of previous test and decide on changes, if necessary to improve performance.	- Connect the GPIO output interface on the board to the NES audio decoders' input interface.	-Begin choosing passive elements that will cancel out our harmonics in the switching frequency to produce clean waves that will translate into working audio.
4/09/18	-Finalize potential changes to the DAC and start overall PCB layout	- Perform physical measurements along the signal processing path and compare them with corresponding LTSPICE simulation results.	-Implement the passives and make sure that our FETs are switching at the required frequency without a breakdown at the gate level
4/16/18	-Finish final PCB layout and	- Identify sources of error between measured and simulated results and tune sub-circuit parameters to minimize undesired phenomenon.	-Continue my verifications of the power unit to ensure that everything is working correctly
4/23/18	-Fix any possible bugs in DAC circuit -Make any needed changes in partners' circuits	- Measure the power transfer efficiencies between signal processing steps and tune sub-circuit parameters to minimize power losses.	-Buffer time to work on the power circuit
4/30/18	-Prepare for final design presentation -Write final paper	- Apply all verifications within this design document to the final result and yield actual operating measurements to provide documentation showing each functional block of this design satisfying the constraints asserted in this document.	-Buffer time to work on the power circuit and ensure that everything is being powered correctly

5. Safety and Ethics

There are several safety concerns with our project. The high power that our device uses can be potentially life threatening if the device is incorrectly implemented or lab safety procedures are not followed.

A Tesla coil has enough voltage that if anyone were to actually touch it, it would deliver a potentially lethal shock. We do not take this precaution lightly and will make every effort to make sure this cannot happen. Under no circumstances will anyone be allowed near the Tesla coil during operation, and long after operation to make sure that the entire circuit is discharged correctly. We also understand that our coil can be scaled up to any amount of power, but we will be designing a coil that will be just large enough for us to create arcs that will allow us to hear the audio and no further. Any larger and we will only get larger arc lengths, which will make the coil more dangerous to others around it.

Another thing worth considering is the initial testing of the coil. This will arguably be the most dangerous part of the project. Because at this point it will be unknown if the circuit will work to its fullest potential we will need to make extra precautions to make sure that if something does go wrong there is zero possibility for someone to get hurt. We will isolate the coil from everything around and make sure to tell all others around us that we will be working with high voltages and arcs. We also will have a kill switch far away from the circuit itself to make sure that if something catastrophic does occur we can end the test immediately before things get too out of hand. If the coil does break it will be a massive setback for our design so getting it right will be critical to our success.

IEEE Code of Ethics, #1: “To hold paramount the safety, health, and welfare of the public, to strive to comply with ethical design and sustainable development practices, and to disclose promptly factors that might endanger the public or environment” [1]. To ensure lab safety and the safety of our peers, we will ensure that our device safely is first simulated in SPICE before bringing the device for testing to the lab. We do not want any possible chances of causing harm to the others or the lab. We will also properly communicate to teaching assistants and teachers our intended usage of the lab and absorb potential feedback to ensure safety.

Due to our use of an NES emulator and Nintendo’s 8-bit sound format, we do not intend to commercialize our product in any form. We understand that this project if commercialized might have conflicts of interest with Nintendo. Adhering to the IEEE Code of Ethics, #2: “To avoid conflict of interest whenever possible, and to disclose them to affected parties when they do exist" [1], we will not try to commercialize or infringe on Nintendo's rights in anyway. Our project is intended for our educational and creative purposes only.

6. Equations

$$C_S = \begin{cases} \dfrac{1.8\frac{pF}{cm}(D-d)}{\ln\left(8\frac{D-d}{d}\right)}; \dfrac{d}{D} < \dfrac{1}{4} \\ 0.37D + 0.23d; \dfrac{d}{D} > \dfrac{1}{4} \end{cases} \quad (1)$$

Let C_S be the toroidal top load's capacitance [pF]
Let D be the toroidal top load's outer diameter [cm]
Let d be the toroidal top load's inner diameter [cm]

$$L_w = \frac{r^2 N^2}{9r + 10l} \quad (2)$$

Let L_w be the solenoid inductance [μH]
Let r be the radius of the coil [in]
Let l be its length [in]
Let N be the number of windings around the solenoid's core

$$\delta = \frac{1}{\sqrt{\pi f \mu \sigma}} \quad (3)$$

Let f be current frequency through conductor[Hz]
Let μ be the permeability of the conductor [H/m]
Let σ be the conductivity of the conductor [S]

References

[1] "IEEE Code of Ethics", *Ieee.org,* 2016. [Online]. Available: http://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: Feb. 7, 2018].

[2] "APU," *wiki.nesdev.com*, 24, Aug. 2017. [Online]. Available: https://wiki.nesdev.com/w/index.php/APU. [Accessed Feb. 7, 2018].

[3] “Class D Amplifiers: Fundamentals of Operation and Recent Developments”, Maximum Integrated, Jan. 2017. [Online]. Available: https://www.maximintegrated.com/en/app-notes/index.mvp/id/3977. [Accessed: Feb. 8, 2018]

[4] “Triangle Wave generator”, *PCBheaven*, July. 2009. [Online] Available: http://pcbheaven.com/circuitpages/Triangle_Wave_Generator. [Accessed: Feb. 9, 2018]

[5] “Solid State Tesla Coil”, Johnson, Gary L. March 15, 2016. [Online] Available: http://www.emsri.org/TeslaBook.pdf. [Accessed: Feb 21, 2016]