

# Design + Analysis of a Wireless Tool Changing System to Transfer both Power + Data Through a Single Coil to a Servo Motor

**Aditya Mehrotra**

Final Project Report w/ Independent Inquiry (ii)  
6.1311 Power Electronics Laboratory  
Department of Electrical Engineering and Computer Science  
Massachusetts Institute of Technology  
Email: adim@mit.edu

**Abstract.** This final project for 6.1311 Power Electronics Lab Independent Inquiry (ii) presents the design and analysis of a circuit to transfer both power and data through a single wireless charging coil set attempting to power and control a small servo motor.

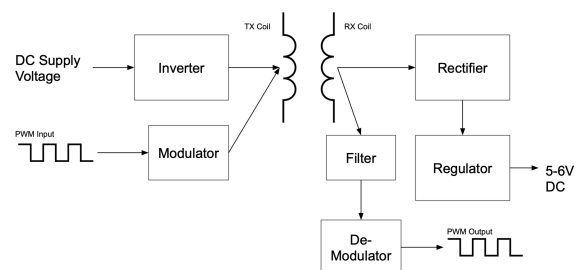
## I. Introduction

A large topic in modern robotics centers around the idea of robot versatility—future advances in robotics are going to heavily depend on our ability to design systems that can perform many tasks instead of one, repeated task. This would drive system cost down as it would require a lower number of robots to automate a complicated task.

One method to increase the versatility of a robotic system is end-effector replacement, or tool-changing. A quick example of this is a CNC machine's ability to swap between end-mills, facing bits, and other types of cutting devices to produce different finishes and shapes of a produced part. Currently, tool-changer systems are mostly mechanical as it is challenging to disconnect the electrical system within a tool and reconnect to a new one safely. Attempts have been made to do this with exposed—but electrically protected—contacts that can transfer both power and data to different end-effectors. This works in

some situations but is dangerous in agricultural robotics and very problematic—if not impossible—in underwater robotics due to the presence of water. Contacts also require precise alignment of the end-effector to the robotic system slowing down the tool-changing process.

In this project, we implemented a system that can transfer both power and data between a robot and its replaceable end-effector wirelessly through the use of inductive coupling. To save material, expense, and size, both power and data were transferred through the same pair of coils using a technique based-off a paper published in IEEE in 2017. A very basic system block diagram is shown below.



## II. Background Research (ii)

**II.1.0 II Papers Consulted.** We started by looking at previous work in the area including *Wireless Power Transfer—An Overview* (IEEE, 2019) [1]. [1] presented different methods that can be used to transfer power wirelessly and considerations related to them. The paper focused on near-field transmission methods including Inductive Power Transfer (IPT) and Capacitively Coupled Power Transfer (CCPT) as opposed to more radiative technologies. Much of the discussion is not presented in this paper as it pertains mostly to EV charging. We chose to focus on the part of [1] describing Inductive Power Transfer (IPT) as this has the most relevance to our system.

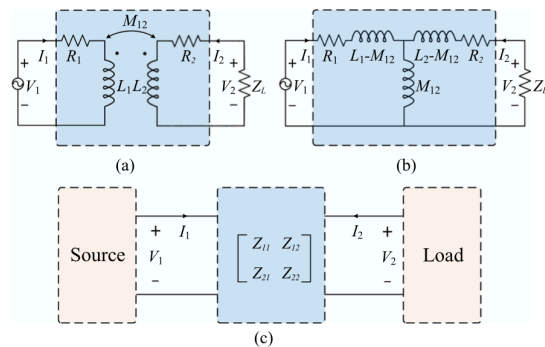


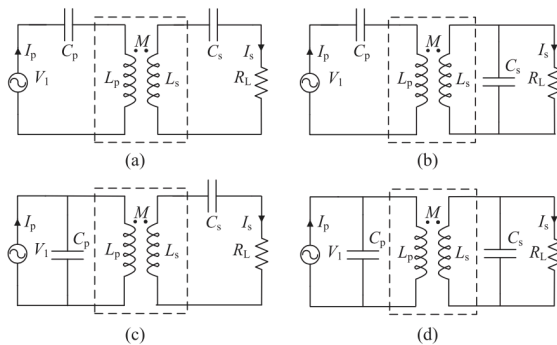
Fig. 1. IPT. (a) Circuit model. (b) Equivalent T-model. (c) Two-port network model.

[1] started by describing the above figure as the most basic IPT system with two magnetically coupled coils, one driven by a sinusoidal source and the other connected to some load resistor. They used this model to derive the relations between input voltage, output voltage, input current, and output current for this simple system. Also derived was the power delivered to the load and the system's coupling coefficient. The mathematics for this system will not be presented in this paper, but the math informed some characteristics of IPT systems that will be presented.

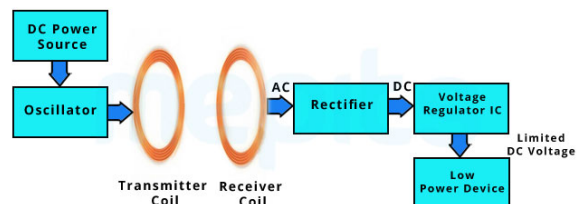
$$\kappa_{12} = \frac{2r}{[\ln(8r/a) - 2] \sqrt{d^2 + 4r^2}} \int_0^{\pi/2} \frac{(2\sin^2\phi - 1)}{\sqrt{1 - \frac{4r^2\sin^2\phi}{d^2 + 4r^2}}} d\phi$$

First, the discussion showed coils in an IPT system should be as flat as possible relative to each other and as close together as possible to maximize the coupling coefficient (see the equation above from [1]). We will keep this in mind when building our system, though it was not a surprise.

Second, the discussion about impedances and the efficiency of power transfer presented in the paper describe the need for capacitive compensation on both sides of the system. [1] explained that to maximize the voltage transferred, capacitive compensation can be used in the forms below. The capacitors also compensate for the imaginary part of the coil impedances.



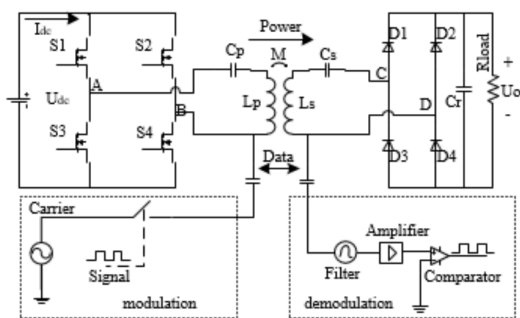
After reading [1] we still felt that we needed a little more of a breakdown on what capacitive compensation meant and how the design of an IPT system would look. We consulted some resources from the Wireless Power Consortium such as [9], [10], and [11]. We came across with the following, highly-generalized diagram of an IPT system from [13].



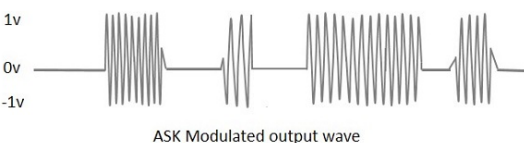
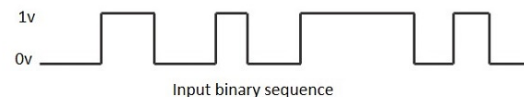
We determined based on this that a wireless power transfer system could simply be comprised of an inverter, some form of LC resonant circuit, and a rectifier on the other

side of the coils. Some questions including whether to drive the system at resonance and what affects the mutual and reflected inductances have on the system were still unanswered but are left for later discussion.

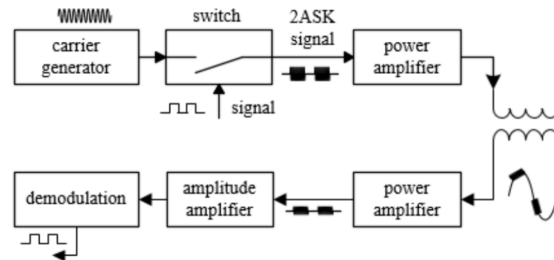
*Data Transmission through Energy Coil of a Wireless Power Transfer System (IEEE 2017)* [2] gave us some insight into how to approach data transfer. When this project was initially proposed the goal was to transfer both data and power wirelessly but not necessarily through the same coil system. The topology presented in [2] was relatively simple and we determined we could attempt to execute it. We had also briefly looked into using NFC to transfer data but deemed this to be too unnecessarily complicated for this project as we are only trying to transfer a simple PWM signal [14].



The above diagram shows the system presented by [2]. In addition to the basic schematic offered, [2] also provided the key insight that the data transfer system must operate at a frequency several orders of magnitude higher than the power transfer system in order to utilize the same set of coils. Then, a technique called Amplitude Shift Keying (ASK) can be employed to



transfer the data where a high-frequency signal sent through the coil represents a “HIGH” value and no signal represents a “LOW” value. The previous diagram, from [15], illustrates this concept.



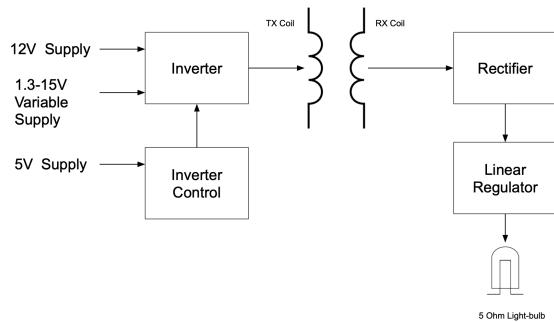
The above figure, from [2], presents a block diagram describing how the signal gets transferred through the entire system. The signal is modulated using ASK and passed through the coil. The coil system’s output is passed through a high-pass filter to isolate the modulated signal. The signal is possibly amplified, and then de-modulated. The paper did not present any information on demodulation and limited information on modulation. Further research on those topics was done and is presented in a later section.

### III. Power Transfer System Design

During the design and implementation phase of the project we split the system into two parts, implemented these parts individually first, and then attempted to combine them together. This section presents the design and construction of just the power transfer component.

**III.1.0 High-Level System Overview.** The goal of the power system is to be able to deliver a regulated 5V at a maximum of 1A to a servo motor on the receiving side of an IPT system. We will connect the servo motor in sections IV and V, but for now we will use an incandescent lightbulb with a resistance of around 5Ω. We will assume (for now) that if the IPT system can power this lightbulb then it will be able to power a small servo. The 1A specification came from [16] which states that the average servo has a stall current of

approximately 1A and draws (on average) a few hundred mA.



The above block diagram shows an overview of the power system. We will design an inverter that takes the 1.3-15V variable supply voltage and converts it to a wave to be sent across the transmitter coil. A rectifier and reservoir capacitor on the other side of the receiver coil will produce a DC output 2-3V higher than our desired 5V output. A linear regulator will then be employed to produce the final 5V output across the lightbulb.\*

*\*Note that we chose to use a linear regulator in this case for simplicity more than efficiency. Due to this fact we will purposefully design the input voltage to the regulator close to the 5V target output voltage. The regulator is essentially present to prevent blowing the servo.*

**III.2.0 Selecting the Wireless Coils.** Our first step was to select the coils we'd be using so we could start building a circuit model of our system for design purposes. We were initially unsure of how to go about selecting the coils so resources such as [17] were consulted but gave us little information on how the inductance of the system and other factors came into play. [4] was interesting but was more of a discussion on how to improve coils and not how to select them from DigiKey listings. So we started with some constraints.

- Select both the receiving (RX) and transmitting (TX) coil from the same manufacturer from the same product line to ensure compatibility.
- Select the coils with a high enough current rating.
- Select a TX/RX coil combination that costs around \$25.00 together.

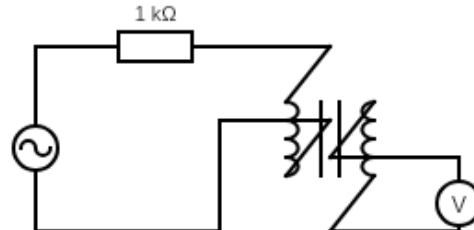
We then found [18] which told us that we should keep the TX and RX coils close to the same size to maximize our coupling factor. Based on all of this we selected the following coils.

Coil	Current Rating	Inductance	Series Resitance	Cost
Würth Elektronik 760308110 (TX)	6 Amps	24 uH	100 mOhms	12.76 USD
Würth Elektronik 7603081022 07 (RX)	5 Amps	8 uH	60 mOhms	9.86 USD

The above table summarizes some of the important information about the coils we will use in our design.

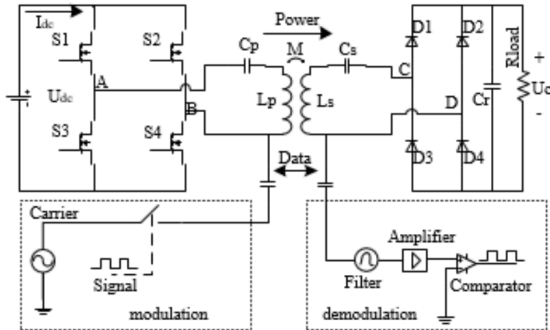
### III.2.1 Initial Measurements on the Coils.

After we selected the coils we wanted to collect some basic information on them experimentally. This included basic information such as the coupling coefficient  $k_c$  of the system. We started with the circuit below.

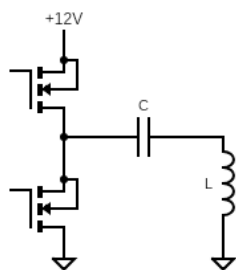


The 1kΩ resistor served as a current limiting resistor for this setup and the system was driven with a 12V @ 60Hz sine wave. We measure the voltage across both the primary and secondary coils when the coils were placed right on top of each other to attempt to determine the coupling between them. We determined 60Hz was too low a frequency to measure the coupling at and that we would get more accurate measurements if the frequencies were in the 80-200kHz range. The scope photos from this test were not included because they are uninformative to the discussion. We decided to move onto constructing our own inverter and measure the coupling and other parameters after this was done.

**III.3.0 Design of the Inverter + Supporting Circuitry.** Initially, we proposed to utilize the schematic presented by [2] which is reproduced below.



This design proposed a dual-totem inverter (essentially an H-Bridge) that utilized four MOSFETs to create a  $-V_{in}$  to  $+V_{in}$  square wave across the coil and resonant capacitor. We started to design and construct this circuit but chose not to go with this design because the supporting circuitry would require two break-before-make circuits, and may require up to (4x) IR2125 drivers if the IR2104 could not be used at the frequencies of operation. We decided to go with a slightly simpler inverter schematic similar to what was used in [19].

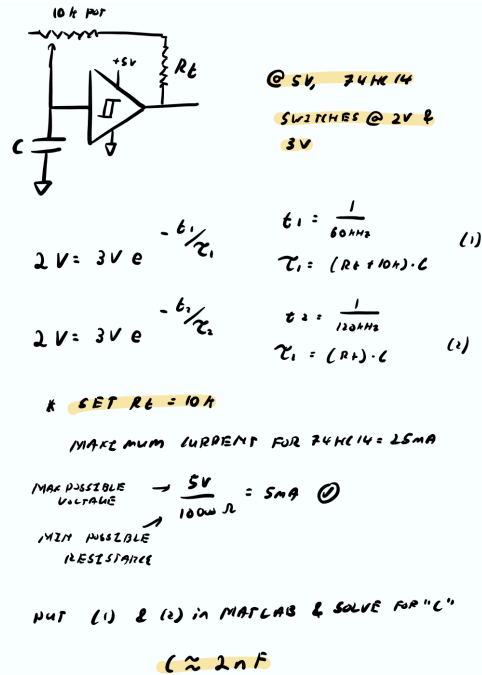


We will instead use a single totem stage powered by the variable supply voltage rail. The capacitor “C” will act as both the resonant and the blocking capacitor for the TX coil.\*

\*Note that the +12V in the figure is replaced with a +1.3-15V variable supply in our system.

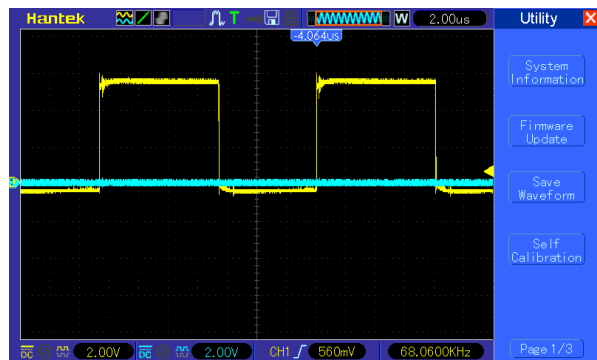
**III.3.1 Design of the Supporting Circuitry.** The inverter needs two pieces of supporting circuitry to operate: the clock, and the break-

before-make circuitry. The clock was initially designed to have a frequency that could vary between 60-120kHz because [2] ran the inverter circuit at around 80kHz. We will use a standard 74HC14 timer circuit to build the clock.



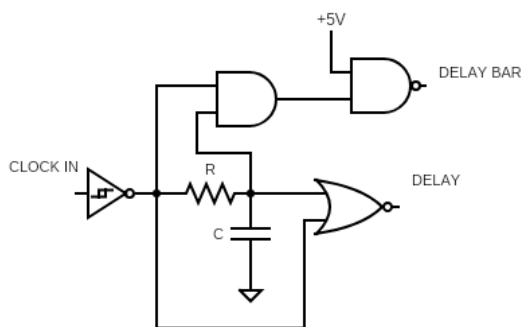
To pick the value of  $R_{trim}$ , we simply picked a resistor value that satisfied the current requirements of the 74HC14. The “C” value was chosen based on the value of  $R_{trim}$  and the limits of frequency we desired.\*

\*Should we need higher or lower frequencies during testing we could simply change the capacitor, decreasing “C” will increase the frequency and vice versa.



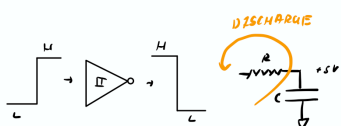
The above waveform shows the output of the 74HC14 clock circuitry running at around 68kHz.

We initially designed and implemented the break-before-make using a schematic that utilized only Inverting Schmitt Triggers. When we tested this circuit we were getting outputs from the circuit that were not desired and we estimated this was due to the high frequency of operation. We switched to the schematic below which utilizes logic gates as these gates are designed to operate at higher frequencies.



The only values to design were the R and C value which set the length of the delay. If we have a switching frequency up to 250kHz, the period of the signal is around 4ms. We probably don't want more than 1ms of delay in this case. The below calculation shows our specifications for R and C.

ANALYZE ON A LOW FREQUENCY



NEED TO KNOW "LOW" TRIGGER OF GATES  $\rightarrow \approx 0.8V$

$$0.8V = 5V e^{-t/\tau} \quad \tau = R \cdot C$$

$$t \approx 1ms$$

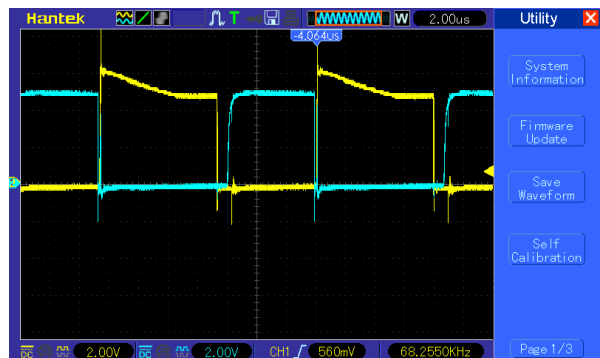
Let's say  $R \approx 470\Omega$  b/c we don't want "C" to be too small

$$\frac{5V}{470\Omega} \approx 11mA \text{ } \textcircled{O} \text{ } \text{Ok FOR 1202T}$$

$$0.8V = 5V e^{-1ms/470 \cdot C} \quad C \approx 1.16 \times 10^{-9}$$

$$C \approx 1nF$$

With  $R = 470\Omega$  and  $C = 1nF$ , we should get a delay of around 1ms which is adequate for our system.



The above waveform shows the output from the break-before-make circuit.\*

\*Note we used a NAND gate instead of an Inverting Schmitt Trigger at the DELAY\_BAR output for compactness when breadboarding. Also note in the waveform above while the delay is small for the first half of the cycle, it does exist.

**III.3.2 Design of the Totem Itself.** The next step was to design the FETs and FET drivers. Because of the high switching frequency, we thought it might be best to go with the IR2125 as the FET driver due to its high drive current capability. The IR2125s would need to drive N-Channel power MOSFETs and the only ones we have available were IRF540s.

ABSOLUTE MAXIMUM RATINGS (T <sub>C</sub> = 25 °C, unless otherwise noted)			
PARAMETER	SYMBOL	LIMIT	UNIT
Drain-Source Voltage	V <sub>DS</sub>	100	V
Gate-Source Voltage	V <sub>GS</sub>	±20	V
Continuous Drain Current	V <sub>GS</sub> at 10 V	T <sub>C</sub> = 25 °C	28
		T <sub>C</sub> = 100 °C	20
Pulsed Drain Current <sup>a</sup>	I <sub>DM</sub>	110	A
Linear Derating Factor		1.0	W/°C
Single Pulse Avalanche Energy <sup>b</sup>	E <sub>AS</sub>	230	mJ
Repetitive Avalanche Current <sup>a</sup>	I <sub>AR</sub>	28	A
Repetitive Avalanche Energy <sup>a</sup>	E <sub>AR</sub>	15	mJ
Maximum Power Dissipation	P <sub>D</sub>	150	W
Peak Diode Recovery dV/dt <sup>c</sup>	dV/dt	5.5	V/ns
Operating Junction and Storage Temperature Range		T <sub>J</sub> , T <sub>STG</sub>	-55 to +175
Soldering Recommendations (Peak Temperature)			300 <sup>d</sup>
Mounting Torque		6-32 or M3 screw	10 lbf · in 1.1 N · m

THERMAL RESISTANCE RATINGS			
PARAMETER	SYMBOL	TYP.	MAX.
Maximum Junction-to-Ambient	R <sub>θJA</sub>	-	62
Case-to-Sink, Flat, Greased Surface	R <sub>θCS</sub>	0.50	-
Maximum Junction-to-Case (Drain)	R <sub>θJC</sub>	-	1.0

- Heatsink power dissipation is 6.0°C/W
- FET power dissipation 62.0°C/W
- Max internal temperature of FET is 175°C
- Assume FET casing maximum temperature of 160°C
- Assume high ambient temperature of 30°C
- Assume a constant 1A drive and draw current from the IR2125

We now want to calculate the losses in the transistor to determine if the system is capable of performing switching at 250kHz. The calculation for this is shown below and is based on the previous assumptions and data.

**SPECIFYING TRANSISTOR LOSSES**

① FET w/o SINK RATED  $\sim 62^\circ\text{C/WATT}$

$T_{max} = 160^\circ\text{C} - 30^\circ\text{C} = 130^\circ\text{C}$   
AMBIENT TEMP

$\therefore 130^\circ\text{C} / 62^\circ\text{C/W} = 2.1 \text{ W possible dissipation}$

② @  $160^\circ\text{C}$   $R_{DS} = 2.0 \cdot 0.077 \Omega = 0.154 \Omega$   
 ASSUME 2A OF CURRENT THROUGH FET @ MAX

$P_{dis} = (2A)^2 \cdot (0.154 \Omega) = 0.616 \text{ W}$

③ WATTS ALLOTTED FOR SWITCHING: **1.484 W**

1 NOW SWITCHING LOSSES!!

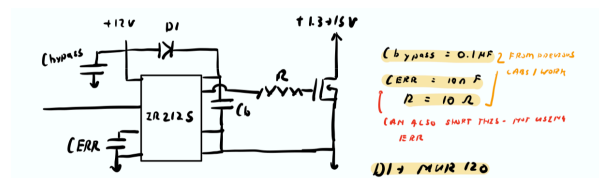
$t_{on} = t_{off} = \frac{115 \text{ nC}}{1A} = 115 \text{ ns}$   
LATE (TABLE Q<sub>1</sub> Q<sub>2</sub> + Q<sub>3</sub>)

$2 \cdot \frac{1}{2} (115 \text{ ns}) (12 \text{ W}) \cdot A_w \leq 1.484 \text{ W}$

$v_{sw} \leq 1 \text{ MHz}$  Ⓞ

The above calculation shows that the FET without the heatsink will be able to handle the 250kHz of switching with a 2A drive current (2A is twice what we expect).

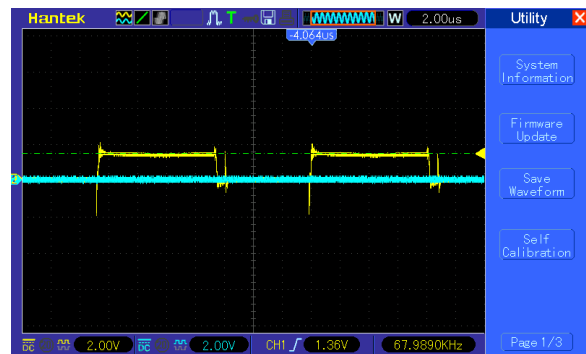
We now need to design the FET driver circuitry using the IR2125. We will use the MUR120 power diode as the charge diode



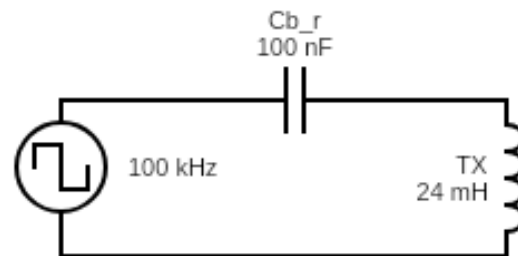
- ① SIZE  $C_b$  such that it does not lose more than 5% voltage...
- $\Delta Q_d = 115 \text{ nC}$     $\Delta Q = C \Delta V$
- $\Delta V = 0.05 (12V) = 0.6V$
- $115 \text{ nC} = C \cdot 0.6V$     $C \geq 0.1 \mu F$
- $\therefore C_b \approx 1 \mu F$  Ⓞ

for the bootstrap capacitor. We will use a standard bypass capacitor of around 0.1μF and an error capacitor of 10nF. The value of C\_ERR doesn't really matter since we aren't using the error feature we can short the error pin to ground. The damping resistor will be set at the standard 10Ω.

We sized the bootstrap capacitor so that it doesn't lose more than 5% of its voltage after delivering the charge needed to turn the IRF540 on. We chose to go with around 1μF for this capacitor as it exceeds that specification. The output of the assembled inverter is shown below with an input voltage of 1.3V at the top of the totem structure.



**III.4.0 The Resonant Cell + Coils.** The completion of a working inverter meant that we could now come back to the initial questions we had about the coil system and start attempting to answer them. From this point forwards the project switched from a more mathematical-based design to more of an experimental-based design.



**III.4.1 Finding Resonance.** The resonant cell takes the inputted square wave and produces a high-voltage AC-sine wave across the coil which is then transmitted to

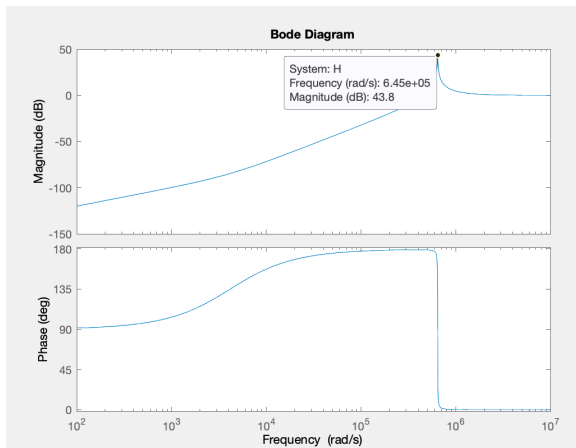
the receiver coil thereby transmitting power. The  $C_{b\_r}$  capacitor acts as a blocking capacitor as well as the resonant capacitor in this case. We chose  $C_{b\_r}$  to be 100nF as that puts the resonant frequency of this circuit at around 102kHz.

$$f = \frac{1}{2\pi\sqrt{LC}} = \frac{1}{2\pi\sqrt{24 * 10^{-6} * 100 * 10^{-9}}} \approx 102kHz$$

Next we made a bode plot of the system in MATLAB based on the following transfer function and adding in the resistance of the TX coil which is around 100mΩ according to [c] (“s” is the Laplace transform variable).

$$H(s) = \frac{sL + R}{sL + R + \frac{1}{sC}}$$

The bode plot of this system from MATLAB is shown below.



We can see a peak at 645 rad/s ~ 102kHz of 43.8dB.

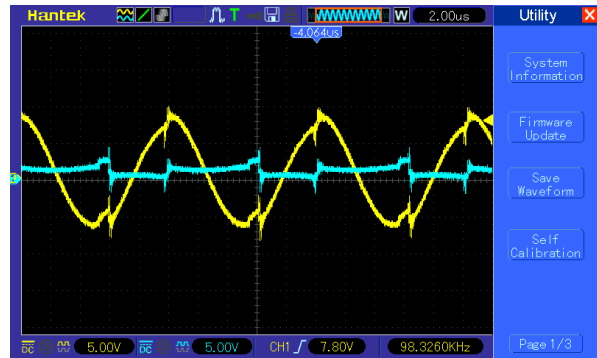
$$dB = 20 \log_{10} G$$

$$43.8 = 20 \log_{10} G$$

$$10^{\frac{43.8}{20}} = G$$

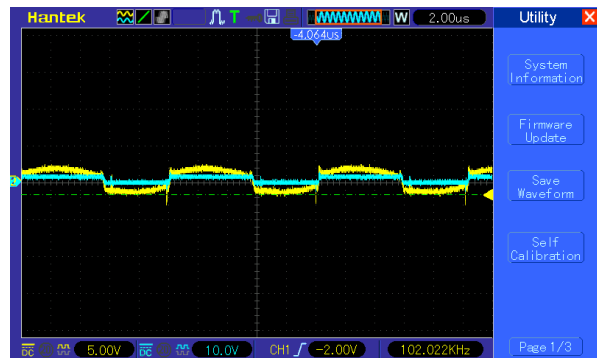
$$G \approx 155$$

With such a high theoretical gain from this circuit we started at a frequency far from resonance with an input voltage to the totem rail of +1.3V and tuned the clock until we either found the resonant peak or the voltage was unsafe. The waveform below shows the output of the system at resonance where we

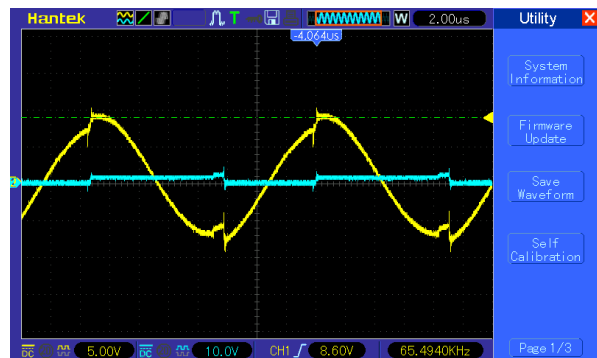


see 7.8V peak being produced across the coil from a +1.3V square wave input.

Now if we pass the receiving coil over the transmitting coil at resonance (load the system) the voltage drops dramatically as shown by the waveform below.



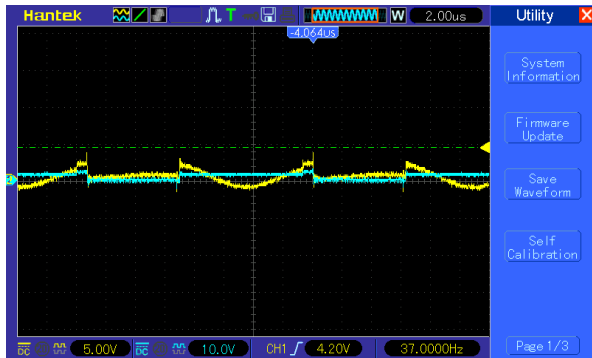
This is dangerous as if we construct the rest of the system using this frequency, when the load is removed the voltage across the TX coil will spike to a voltage that is likely higher than the rating of some of the components. We want to find the frequency where when the RX coil is passed over the TX coil the system resonates but when it is removed the





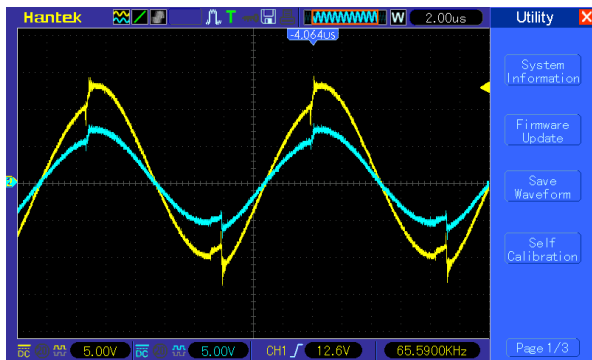
voltage drops. Essentially the opposite behavior of what we are currently seeing.

We held the RX coil on top of the TX coil and tuned the frequency until we found the resonant peak above. This peak happened at around 65kHz and we can see 8.60V being produced across the TX coil from a +1.3V totem rail. When we removed the RX coil, the below waveform was produced on the scope.



Based on the above experiments, we chose to drive the power system at 65kHz as at this frequency the system resonates when loaded but when the load is removed the voltage returns to a safe voltage.

**III.4.2 Determining Coupling.** We could also use the experiment to determine the coupling between the coils for later calculations. We setup measurements across the TX coil (in yellow) and the RX coil (in blue) and increased the rail voltage to approximately +2V. We got the following waveforms.



From a +2V totem rail we got 12.6V across the TX coil and 6.6V across the RX coil

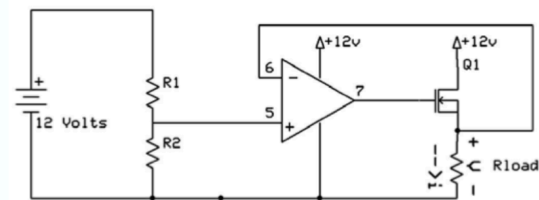
meaning we can calculate the approximate coupling for this system.

$$k \approx \frac{6.6V}{12.6V} = 0.524$$

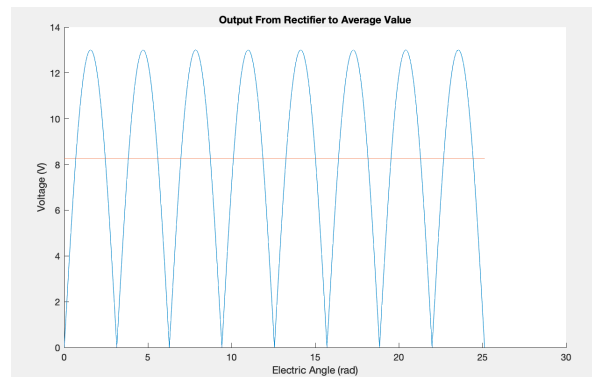
**III.4.3 Summary of Details.** The table below summarizes some basic findings.

System	Resonant Frequency	Coupling
TX Coil (only)	98-102 kHz	0 (no voltage across the RX coil)
Loaded RX on TX	63-68 kHz	~0.524

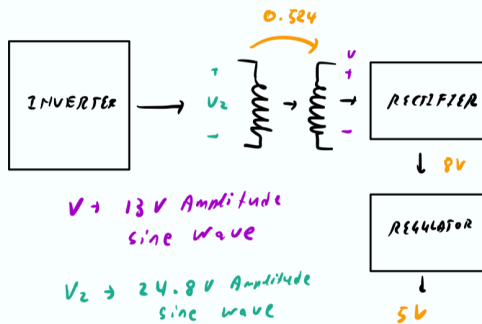
**III.5.0 Designing the Remainder of the Power System.** We can now move on to the rectifier as well as the linear regulator components of the power transfer system.



**III.5.1 System Characterization.** Let's say we decided to use a linear regulator with a FET assist like the one shown above from [20]. In these type of regulators the voltage input must be at least 2V above the desired output voltage otherwise the FET cannot turn on. So if we desire a 5V DC output from this type of linear regulator, we would like to have an 8V input to ensure we actually get 5V at the output (3V margin instead of 2V).



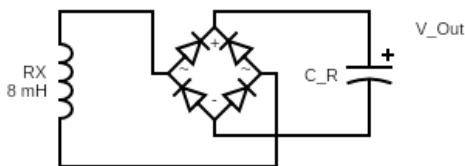
To have an 8V DC input to the linear regulator, the rectifier must produce a waveform with an average value of 8V DC. The above graph shows that a 13V amplitude sine-wave input to the rectifier would produce a DC input of around 8V to the regulator from the rectifier. The blue waveform is the output of the rectifier and red waveform is the average value of the blue waveform.



We can follow the voltages we require through each part of the system to determine we need approximately a 25V amplitude sine wave across the TX coil to generate 5V at the output. To determine what input rail voltage will generate a 25V amplitude sine wave depends on the current we are drawing so we will start with a 2V input and raise the voltage as necessary after the load has been added.\*

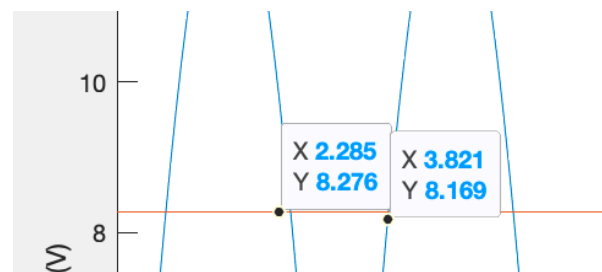
\*Note that we are choosing to experimentally determine the input rail voltage because the presence of a rectifier and regulator make it complicated to build a simple circuit model for this system where we can reflect the impedances over the TX/RX coil transformer.

**III.5.2 Rectifier Design.** For the rectifier we will use the topology of a simple full-wave-rectifier with reservoir capacitor shown below.



For the diodes we will use MUR120 power diodes which have current ratings up to 1A.

For the reservoir capacitor, we want to size this so that we minimize the voltage ripple. Looking at the graph we made of average voltage and rectifier output, we know that this system will only draw current from the line when the diodes are forward biased meaning when the voltage of the line is above the voltage of the capacitor (the average DC voltage). For the rest of the time, the load is drawing current directly from the capacitor.



Looking at the approximate points in the graph where the line voltage decreases below the average voltage we can approximate the amount of time that current is being drawn directly from the capacitor at 65kHz.

$$d = \frac{\theta_f - \theta_i}{2\pi} * 2 = \frac{3.821 - 2.285}{2\pi} * 2 = 0.4889$$

$$t = \frac{d}{f_{sw}} = \frac{0.4889}{65000} \approx 7.5\mu s$$

Then if we assume that a constant current of 1A is being drawn from the capacitor during this time, we can then calculate the charge removed from the capacitor.

$$Q = ti = (1A) * (7.5\mu s) = 7.5\mu C$$

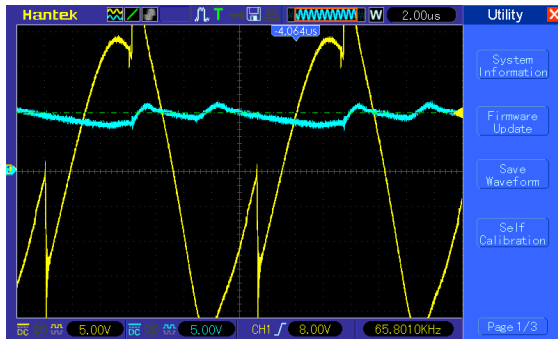
We can then size the capacitor based on the maximum “droop” in voltage we allow for our system. Let’s say that maximum droop is around 0.6V.

$$\Delta Q = \Delta VC$$

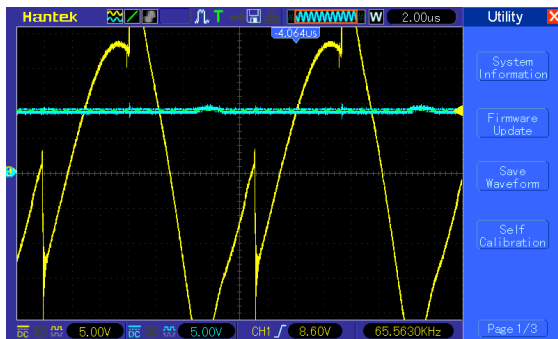
$$-7.5\mu C = -0.6 * C$$

$$C = 12.5\mu F$$

To be prepared for current spikes and other anomalies we multiplied this value by approximately four, choosing a 47uF 50V electrolytic capacitor to be the reservoir capacitor. The waveform for the output with a small load is shown below. We can see the rise and fall in voltage of the capacitor around a nominal value of 8V.

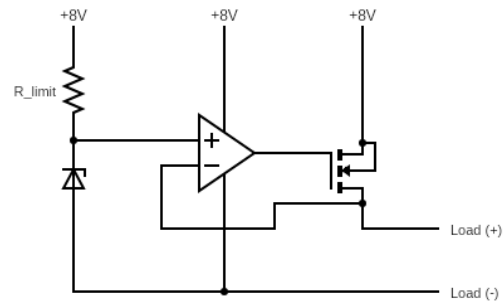


We weren't satisfied with this level of voltage ripple and the asymmetry of the waveform. Because we aren't concerned with system cost and much bigger capacitors are available to us, we experimented with a 1000uF 16V electrolytic capacitor and got the following waveform with the same load. This essentially eliminated all voltage ripple.



There was no reason to not use the 1000uF capacitor so that became the final reservoir capacitor for the system.

**III.5.3 Regulator Design.** When designing the regulator to produce a reliable and stable 5V-6V DC output we combined two topologies from [20] to ensure the regulator output was not affected by changes in the voltage input as long as the voltage remained above the minimum for a FET-assisted linear regulator.



We combined the use of a Zener Voltage Reference and a FET-assisted regulator to produce the following design for our linear regulator. This design is very similar to that of a 7805. We will use an IRF540 N-Channel MOSFET for the FET and an LM358 for the Op-Amp.

The zener reference requires a 5V-6V zener voltage. We wanted a zener diode rated to a higher power rating than 1W for safety more than necessity. We decided to go with a 1N5340BRGL where the key specifications are summarized in the table below. The 6V zener voltage may allow the system to power larger servos during further testing.

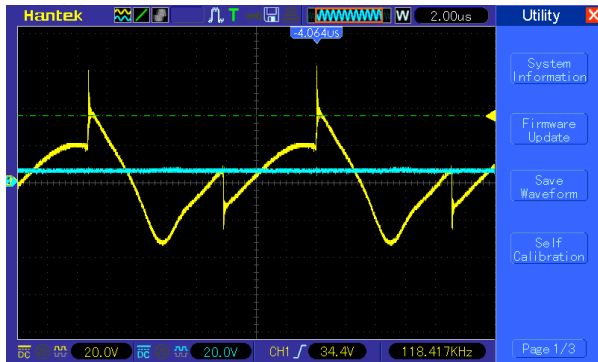
Diode	Zener Voltage	Wattage Rating	Iz Max
1N5340BRGL	6 V	5 W	790 mA

The R\_limit resistor was sized based on the current limits of the zener diode. A 1K resistor was used for R\_limit for efficiency.

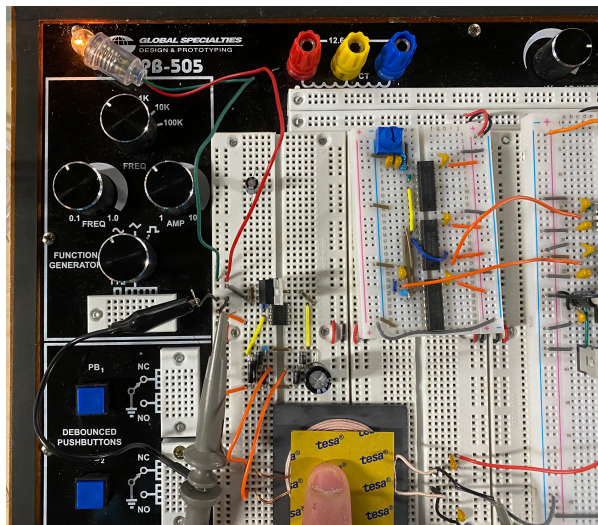
**III.5.4 Tuning + Testing.** A detailed final schematic of the power system is presented in Appendix B with the designed values included. To test the system, we loaded the output of the linear regulator with a single lightbulb with the resistance of about 5Ω. The +1.3-15V variable power supply was set at 1.3V and slowly increased until one of two things happened.

- The system delivers 5V to the lightbulb load.
- The voltage across any component exceeds 40V peak (at this point we shut the system off).

Especially of concern is the capacitor on the TX side of the circuit which is acting as both a blocking and resonant capacitor meaning it will have both the resonant and blocking components of the voltage across it. That capacitor is a ceramic capacitor rated to 50V. We found that a 12V rail voltage will produce the following waveforms across the TX coil and the lightbulb.



The above waveform shows a peak of 34.4V in the waveform across the coil and a 5V DC output across the lightbulb. The voltage across the capacitor would be 40.4V if we add the 6V amplitude sine wave created from the 12V square wave input.



The above picture shows the results of the lightbulb being powered by the wireless system with the rail voltage set at 12V. Note the lightbulb in the top left corner of the image.

**III.5.5 Notes not Discussed.** In the final schematic a 220uF electrolytic capacitor is present at the top of the totem rail connected between the source and ground. This capacitor is present to help with current spikes as the 1.3-15V variable supply is not rated for high currents and is prone to “droop.” Various bypass capacitors placed in the final schematic were also not discussed in this section.

It is also important to note that inductors require freewheeling current paths to prevent arc-ing across parts of the circuit that can damage components. Freewheeling on the TX side is presented by the in-built diodes in the IRF540 MOSFETs. On the RX side, the rectifier itself provides a freewheeling path.

## IV. Data Transfer System Design

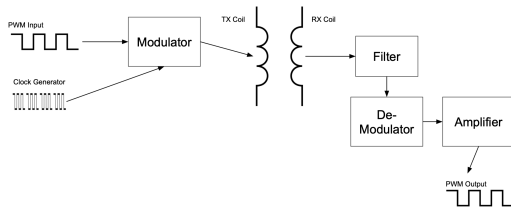
This section presents the design and construction of just the data transfer component of the system.

**IV.1.0 High-Level System Overview.** Using information presented in [2], the data transfer system creates a signal to transfer across the coil and modulates it with a high frequency clock wave. This signal is transferred across the coil system, filtered, demodulated, and then amplified to produce the final output signal on the other side of the coil. The block diagram on the next page shows this pictorially.

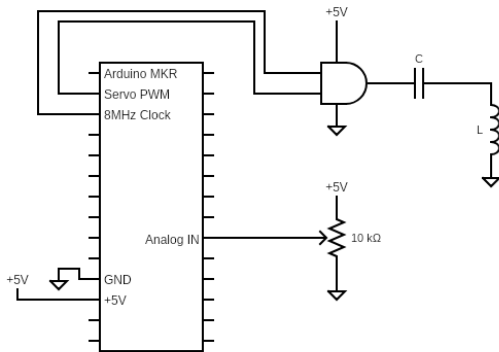
**IV.2.0 Transmitting Side Hardware Design.** On the transmitting side, the system needs to perform ASK as described in section II. This involves signal generation and modulation.

The simplest way to generate a PWM signal that could control a servo as well as a high-frequency clock signal several orders of magnitude higher than the 65kHz the power transfer system is operating at was to use a microcontroller. We had a choice between a Cypress PSoC and an Arduino MKR WiFi 1010. We chose the Arduino because of its familiarity but either would have worked for our application.

For ASK to work, the high frequency clock signal can only be sent through the coil when the PWM is in a “HIGH” state, otherwise no



signal is sent through the coil. If both the clock and the PWM signal are 5V square-

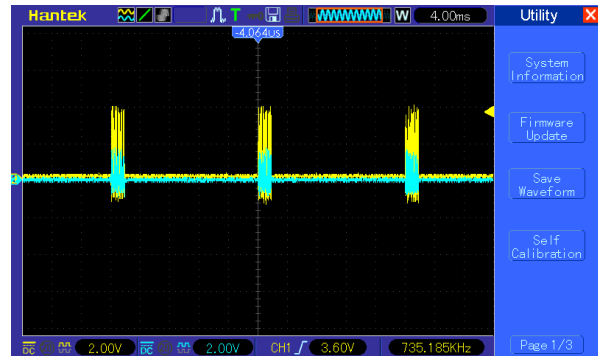


waves the easiest way to achieve this is to use an AND gate. We chose a 74LS08N dual-input AND gate with a maximum operating frequency of 35MHz. The below schematic shows the transmitting side of the data transfer system. The programming of the microcontroller was fairly straightforward. To generate the 8MHz clock signal we used code from [8], the Arduino Community. To generate the servo pulse, the code reads the analog voltage of a potentiometer and uses that to set the duty cycle of a servo using the Arduino Servo Library. See Appendix A for commented code.

For the value of the resonant and blocking capacitor in this case, we can use the same procedure from Section III but this time attempt to make the system resonate at 8MHz knowing  $L=24\mu\text{H}$ . This gave us a C value of 16pF. But according to [21] we can expect between 2-3pF of parasitic capacitance from the breadboard alone so

we started with a 10pF ceramic capacitor for C.

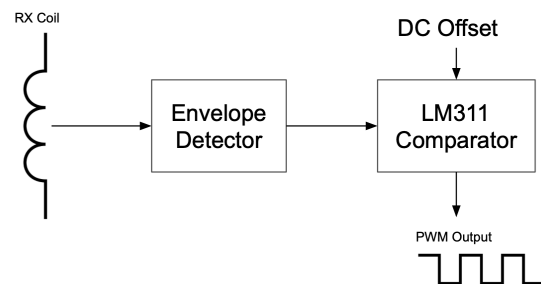
When we built and tested the circuit, however, the system produced almost no output across the RX coil when  $C=10\text{pF}$ . We experimented with a few other values of C until we settled on 1nF which gave us the following waveforms. The yellow is the modulated signal and the blue is the output from the RX coil.



Because of the demodulation technique used in IV.3.0, this signal amplitude will be enough to pass data through the coil.

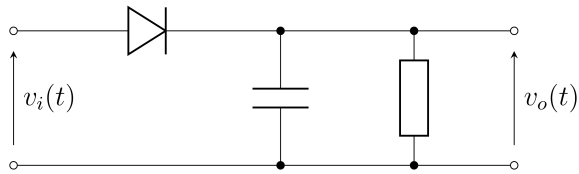
### IV.3.0 Receiving Side Hardware Design.

The RX side of the data transfer system has three main components: filtering, demodulation, and amplification. The design of these systems was done mostly through trial and error seeing what worked and what didn't. The end goal was to simply take the signal seen across the RX coil in IV.2.0 and turn it into a 5V PWM square wave that could be sent to a servo. The below block diagram describes the system in more detail.

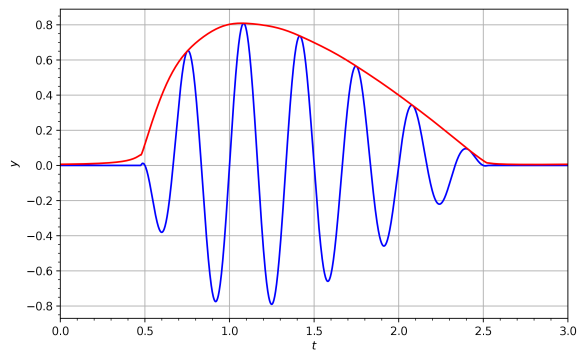


**IV.3.1 Envelop Detector Design.** For now, filtering was not employed in the data transfer system as it was not needed. So the

Envelope Detector became the first part of the circuitry to design.



An envelop detector is a very simple circuit that takes a high frequency waveform as its input and produces a demodulated form of this input. A schematic is shown above and an example is shown below (both are from [22]).



This kind of detector works by quickly charging up the capacitor to maintain a DC voltage at  $V_o(t)$ , and when the high frequency signal at  $V_i(t)$  stops the capacitor discharges through the resistor. The inputted signal must exceed the turn-on voltage of the diode for this to work. We used a 1N4148 signal diode which has a turn-on voltage of around 500mV according to the below graph from [f].

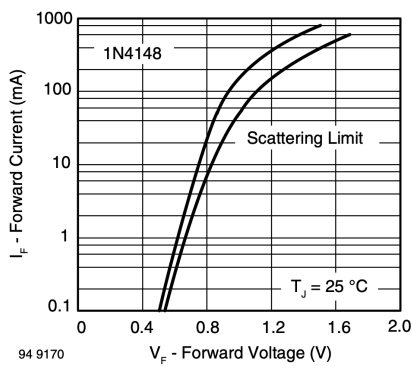
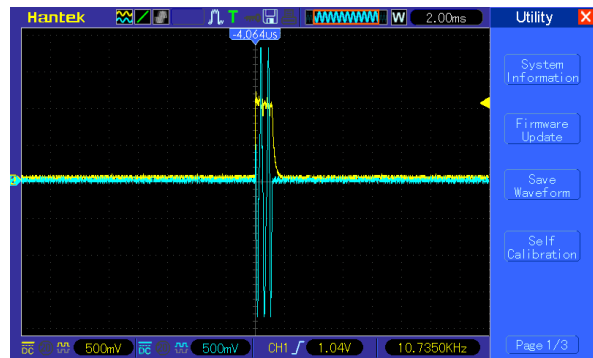


Fig. 2 - Forward Current vs. Forward Voltage

To specify the values of the resistor and capacitor for this circuit we use the following ideas:

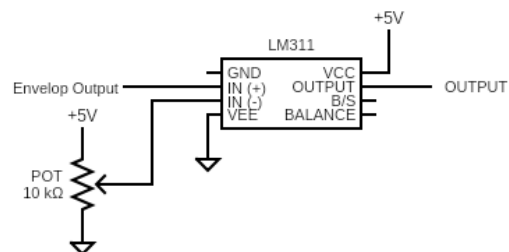
- The capacitor should be relatively small so it can charge quickly.
- The resistor should be large enough so that most of the current goes into charging the capacitor, but small enough so the capacitor discharges quickly.

We took a trial and error strategy to get this system to work. We started with a 680pF capacitor and we changed the value of the resistor until we were satisfied with the performance of the system. We wanted at least a 500mV peak for the square wave as any lower would be difficult to precisely amplify with the LM311.

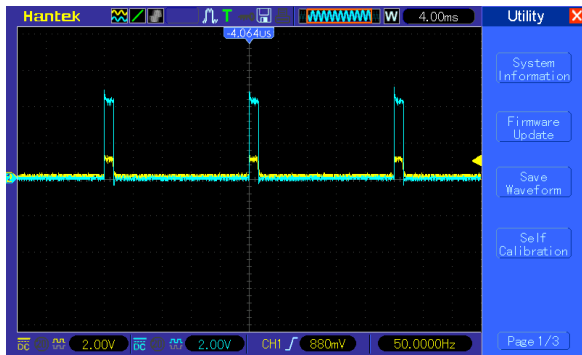


In the above waveform, blue is the output from the coil and yellow is the output from the envelope detector. This waveform was taken with  $R=220k\Omega$  and  $C=680pF$ . Larger resistors provided too slow of a discharge and smaller resistors dropped the peak voltage of the waveform below 500mV.

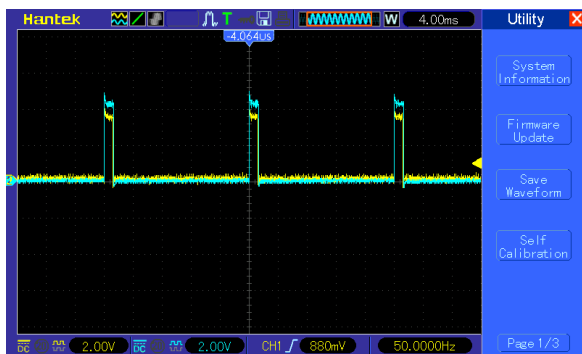
**IV.3.2 Comparator Amplifier Design.** The last step on the receiving side in this testing phase was to amplify the signal using an LM311 comparator.



The above schematic shows the circuit of the comparator amplifier. For now, the +5V and GND will be connected to the Arduino's +5V and ground to test signal transfer separately from power transfer. The servo will also be powered from the Arduino during this phase of testing. The potentiometer in this circuit is setting the comparison voltage, in our case we want that to be ~250mV. The below waveform is the input to the LM311 in yellow and the output in blue.



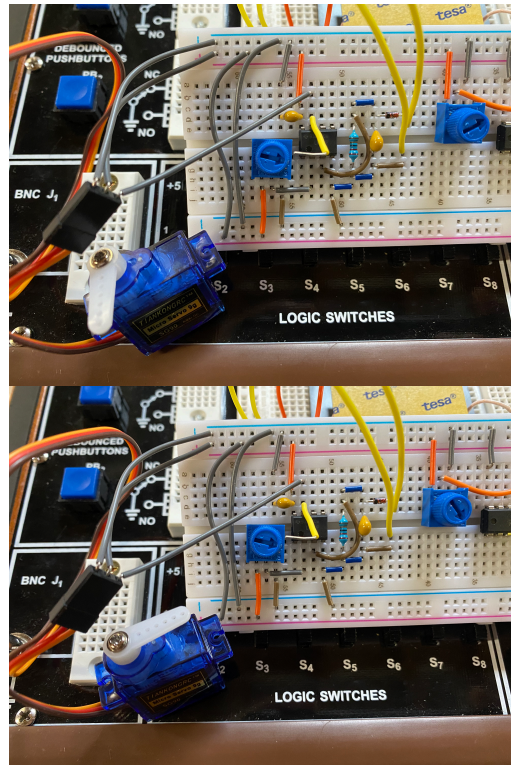
And the final test of the system, the below waveform is the servo signal from the Arduino (yellow) compared to the output from the LM311 (blue).



**IV.4.0 System Test w/ a Servo.** As mentioned in IV.3.2, we ensured the system was outputting a valid servo signal that could be used for control by hooking up a servo to the signal transfer system. The servo was powered directly from the Arduino but the signal was transferred through the wireless coil.

The following images show that setting the input potentiometer that changes the duty cycle of the servo PWM wave, changes the

position of the servo after the signal is transferred through the coil system.

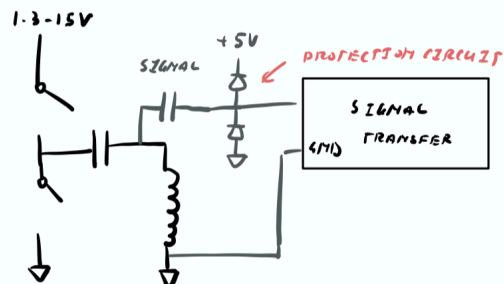


Note the position of the right potentiometer and the servo horn in the above two images.

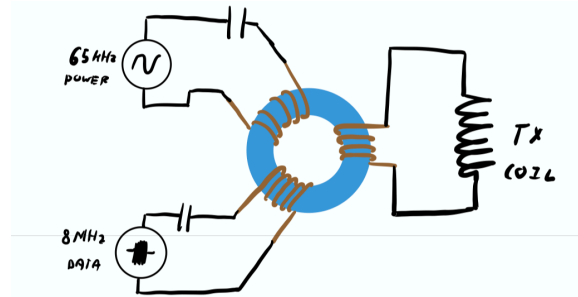
## V. Combining Power + Data

This section presents the attempt to combine both the power and data transfer systems into a single system.

**V.1.0 Initial Ideas + Attempts.** There are two major problems to solve when combining the data and the power systems. The first is how to actually superpose the data and power signals on the TX side. And the second is how to separate them on the RX side.

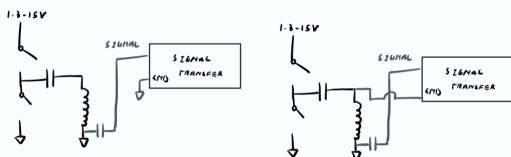


The above diagram shows a possible idea for superposing the signals on the TX side. This was never constructed or tested as we were worried that the high voltage being produced across the TX coil for power transfer may pass through the signal transfer capacitor and damage the ICs on the other side.



The drawing above is an idea that came from Eric Ponce, which is to use a 3-way transformer with a 1:1:1 ratio to combine the signals. We weren't sure what the effects of the transformer would be on this system so this idea was left as an alternative should no other solution be found.

**V.1.1 Discussion from [2].** So far, we have not discussed the data transfer system presented in [2] as that involved injecting the data at the bottom of the transmitter coil which (in our power system design) is connected to ground.



We believed the paper was describing one of the two configurations above. In the first configuration, the capacitor would essentially block the signal from passing through the TX coil and the base of the TX coil would remain at ground since the grounds of both the signal transfer and power transfer circuits are connected. In the second circuit, we are connecting a floating ground of the signal transfer circuit to the varying sinusoidal voltage at the top of the coil which is bound to cause issues.\*

*\*It's important to note that the entire signal transfer system is being powered off of the separate USB power supply of the Arduino, independent of the power transfer system's power supply.*

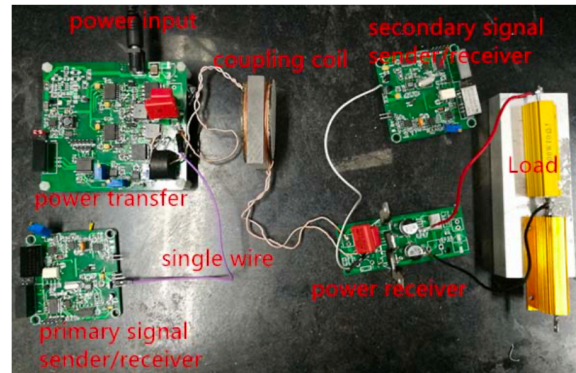
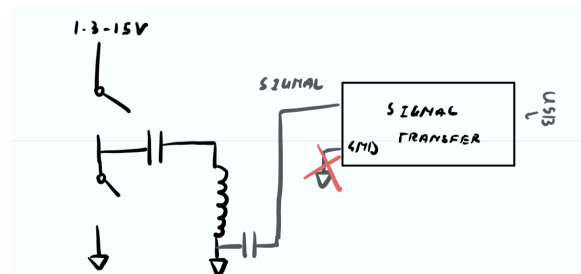
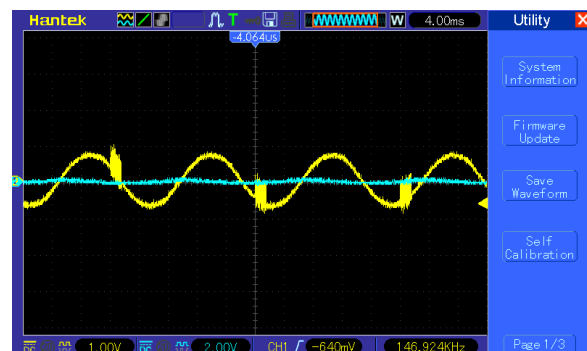


Fig.6 the experimental platform of system

The above figure from [2] gave us some insight. Note that in the above figure, the signal and power transfer circuits are powered completely independently and the grounds of the systems are not connected (see the diagram below).



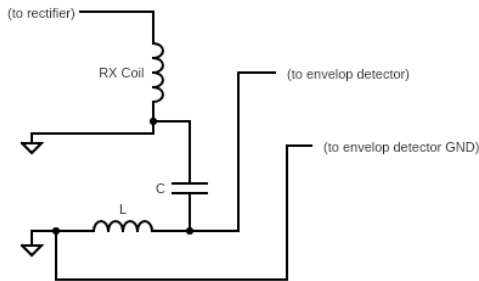
In this case (since the grounds are not connected) the 8MHz data waveform can "ride" the 65kHz power waveform. Shown below is a purposefully generated demonstration of this phenomenon and not an actual example of what we would see





across the transmitting coil. But the example illustrates what the phenomenon may look like.

**V.2.0 Separating Data from Power on the RX Side.** To separate the signals on the RX side, we will employ the use of an LC filter with a high-frequency cutoff.



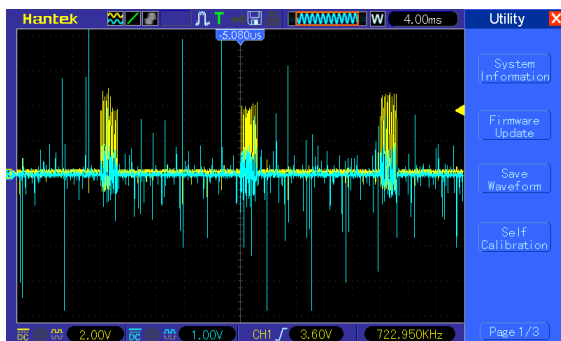
The design of this LC filter is fairly straightforward. The filter simply needs to have a cutoff frequency high enough where only the 8MHz signal is able to pass through. We will set the cutoff to be high as in addition to the 65kHz power waveform, we have been consistently noticing some high frequency noise when both power and data have been transferred through the coil.

$$f = \frac{1}{2\pi\sqrt{LC}} = \frac{1}{2\pi\sqrt{L * 680pF}} = 6MHz$$

$$f = \frac{1}{2\pi\sqrt{LC}} = \frac{1}{2\pi\sqrt{1uH * 680pF}} \approx 6.1MHz$$

$L = 1uH$

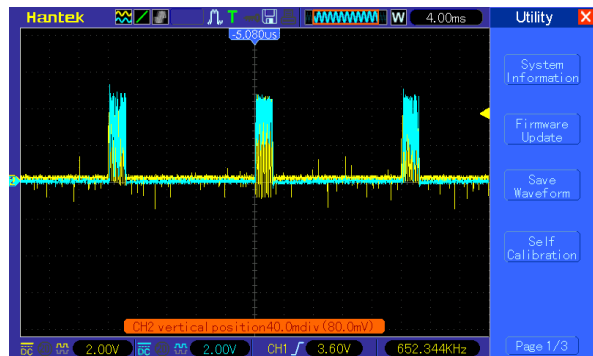
The 680pF capacitor was specified using trail and error. The 680pF capacitor gave us the highest voltage of the signal waveform after it had passed through the coil. If we place the 680pF capacitor with a 1uH



inductor we get a frequency cutoff of 6.1MHz. The scope photo above shows the still-very-noisy input to the envelop detector (blue waveform).



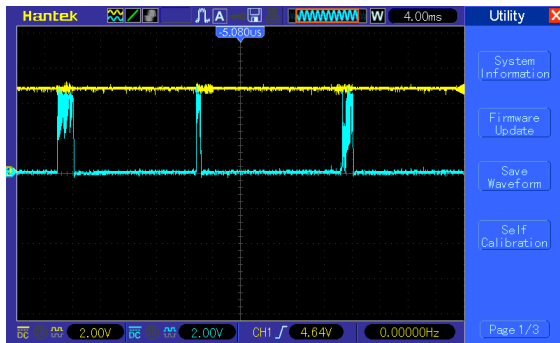
The output of the envelop detector is shown above (blue waveform). An interesting phenomena occurs where because the noise is much larger than the signal, the signal pulls down the output of the envelop detector just enough where we might be able to think about separating it using the LM311 circuit.



Above is the output from the LM311 after very carefully tuning the potentiometer (blue waveform).\*

*\*One important note is that this level of noise in the system is common at higher power levels. At low power levels (where the voltage output from the regulator is nearing zero and the input voltage to the inverter is 1.3V) we get a very clean-square wave of data being transferred. The above results are when the input to the inverter is set so that we see a 5V output from the regulator.*

**V.3.0 Waveforms for Power + Data.** The final experiment was to hook up the servo to see if we could control and power the servo with the outputs we were getting.



In the above photo, the yellow waveform is the ~5V output from the linear regulator meaning power is being transferred. The blue waveform is the output from the LM311 meaning data is also being transferred but not as cleanly as we would hope.

The servo accepted power from this source but exhibited jittering motions and didn't respond to changes in the control command. This suggests that this waveform is much too noisy for the servo to recognize it.\*

*\*For a full schematic of the combined data and power transfer system, please see Appendix B.*

## VI. Analysis + Conclusions

The goal of this project was to inductively transfer both power and data through a wireless power coil system to both power and control a servo. We were not able to finally power and control the servo motor but both power and data were transferred through the coil set both together and individually.

### VI.1.0 Issues w/ the System Presented.

The final system was able to transfer both power and data through the coil but the data that was transferred was much too noisy to be recognized by the servo. There are many places this noise could have originated from.

- General parasitics from the breadboard such as the capacitances between the rails.
- Parasitics inductances that could have originated from the loops of wire in the circuit.
- Noise on the input +1.3V-15V power supply rail.

The lack of amplification of the data signal before it was sent across the coil also became an issue as this produced very small voltages on the receiving side (<1V) which made it hard to separate the noise from the signal.

It's difficult to pinpoint the possible issues because there are many. It is also hard to say whether this system would've worked had it been constructed on a PCB as opposed to a breadboard. With better managing of parasitics and better breadboarding, as well as taking extreme care to generate clean signals in all parts of the circuit, it's possible this system could both wirelessly power and control a servo.\*

*\*Another possible issue discussed during the Final Project Checkoff was that the choice to use a full-wave rectifier means there is no reference "ground" for the servo for signal transfer to operate correctly. Using a half-wave rectifier with a larger capacitor could solve this problem in the future.*

### VI.2.0 Analysis on Viability of the System Presented.

Onto the question of applying this system to the problem of wireless tool-changing, we think this system falls short of a system that could reliably work for the applications initially presented (even if you ignore the issues already discussed). One of the problems with this system is that it does not work when the ground rail of the data transmitter circuit is connected to the ground rail of the inverter as discussed in V.1.1. We don't know if this issue comes from the way the system was built or if it's a fundamental design issue, but if we wanted to design a compact tool changing system we might not want to isolate the grounds or may find it difficult to do so. We say this speaking about the system in terms of a product rather than in a research sense.

There are other methods we could look into to transferring data and power at the same time (RFID is one) and there are other ways we could combine these two signals and send the combination across the coil with the grounds connected as discussed in V1.1.0. Ultimately, more work would need to be done to figure out what is the best method to convert this system into one that could work for our tool-changer idea. It's

hard to say if this system is viable or not for a tool-changer without further testing.

### VI.3.0 Summary of what was Achieved.

Throughout this project we gained a deeper understanding of both Inductive Power Transfer and Inductive Signal Transfer, two areas in which we previously had no experience. We were able to construct a wireless power system that transferred 5V 1A to a lightbulb and returned to a safe voltage when the load was removed. We were also able to send signals across a wireless coil to control a servo. We were able to successfully combine the two systems based on a technique found in [2] in the sense that both power and data were transferred across the coil system at the same time, and the data could be demodulated and a 5V output was obtained. And though the combined system was ultimately too noisy to achieve the goal of powering and controlling a servo, the combination taught us much about systems that can transfer both power and data and has opened up avenues for future work.

## VII. Special Thanks To

We'd like to give a special thanks to Professor Steven B. Leeb and the 6.1311 course staff as well as Eric Ponce, Manuel Gutierrez, and Daniel Monagle as this project and learning experience would not have been possible without them. We loved this class and have learned so so so much from it.

## VIII. References

- [1] *Wireless power transfer overview*: <https://ieeexplore-ieee-org.libproxy.mit.edu/document/8357386>
- [2] *Data transmission through energy coil of wireless power transfer system*: <https://ieeexplore-ieee-org.libproxy.mit.edu/document/7959373>
- [3] *An overview of capacitively coupled power transfer*: <https://ieeexplore-ieee-org.libproxy.mit.edu/document/6566413>
- [4] *Paper about special coil geometry*: <https://ieeexplore-ieee-org.libproxy.mit.edu/document/5278713>
- [5] *Creating a 5MHz clock-signal from an Arduino*: <https://forum.arduino.cc/index.php?topic=432538.0>
- [6] *Build your own Induction Charger (Nuts & Volts)*: [https://www.nutsvolts.com/magazine/article/august2013\\_Bates](https://www.nutsvolts.com/magazine/article/august2013_Bates)
- [7] *Zener Diode as a Voltage Regulator*: [https://www.electronics-tutorials.ws/diode/diode\\_7.html](https://www.electronics-tutorials.ws/diode/diode_7.html)
- [8] *Arduino MKR 8MHz Clock Output*: <https://forum.arduino.cc/index.php?topic=620903.0>
- [9] *Inductive Power Transfer*: <https://www.wirelesspowerconsortium.com/knowledge-base/magnetic-induction-technology/how-it-works/basic-principle-of-inductive-power-transmission.html>
- [10] *Coupling Factor Basics*: <https://www.wirelesspowerconsortium.com/knowledge-base/magnetic-induction-technology/how-it-works/coupling-factor.html>
- [11] *Basics of Qi Wireless Power Systems*: <https://www.wirelesspowerconsortium.com/knowledge-base/magnetic-induction-technology/how-it-works/how-it-works.html>
- [12] *Inductive vs. Resonant Charging*: <https://www.digikey.com/en/articles/inductive-versus-resonant-wireless-charging>
- [13] *Mepits, Wireless Charging*: <https://www.mepits.com/project/171/wireless/wireless-power-transmission-mobile-charger-circuit-using-inductive-coupling>
- [14] *NFC Reader Design*: <https://www.themobileknowledge.com/wp-content/uploads/2017/05/NFC-Reader-Design-How-to-build-your-own-reader.pdf>
- [15] *Amplitude Shift Keying*: [https://www.tutorialspoint.com/digital\\_communication/digital\\_communication\\_amplitude\\_shift\\_keying.htm](https://www.tutorialspoint.com/digital_communication/digital_communication_amplitude_shift_keying.htm)
- [16] *Servo Stall Currents*: <https://www.pololu.com/blog/16/electrical-characteristics-of-servos-and-introduction-to-the-servo-control-interface>
- [17] *Key Factors for Wireless Power Coils*: [https://www.wonline.com/web/en/electronic\\_components/news\\_pbs/blog\\_pbcn/blog\\_detail-worldofelectronics\\_109210.php](https://www.wonline.com/web/en/electronic_components/news_pbs/blog_pbcn/blog_detail-worldofelectronics_109210.php)
- [18] *Selecting the right coils for WPT systems*: [www.youtube.com/watch?v=MvHSU9FNe8w](https://www.youtube.com/watch?v=MvHSU9FNe8w)
- [19] *Lab 3 single-totem inverter for the ballast circuit.*
- [20] *Lab 1 Linear Regulator w/ NFET Assist.*
- [21] *Parasitic Capacitance of a breadboard*: <https://electronics.stackexchange.com/questions/302850/what-order-of-magnitude-should-i-expect-the-parasitic-capacitance-to-be-for-a-so>
- [22] *Envelope Detector*: [https://en.wikipedia.org/wiki/Envelope\\_detector](https://en.wikipedia.org/wiki/Envelope_detector)

[23] *Servo PWM Current Draw*: <https://electronics.stackexchange.com/questions/217439/required-servo-control-current>

[24] *Lab 2 220uF Cap on Switching Rail + IR2125 Schematics*.

## IX. Data-sheets Not Provided

[a] *1N5340BRGL 6V 5W Zener Diode*: [https://www.mouser.com/datasheet/2/308/1N5333B\\_D-1801430.pdf](https://www.mouser.com/datasheet/2/308/1N5333B_D-1801430.pdf)

[b] *Arduino MKR WiFi 1010 Pinout*: [https://content.arduino.cc/assets/Pinout-MKRwifi1010\\_latest.pdf](https://content.arduino.cc/assets/Pinout-MKRwifi1010_latest.pdf)

[c] *WE-WPCC 760308110 Wireless TX Coil*: <https://www.w-online.de/katalog/datasheet/760308110.pdf>

[d] *WE-WPCC 760308102207 Wireless RX Coil*: <https://media.digikey.com/pdf/Data%20Sheets/Wurth%20Electronics%20PDFs/760308102207.pdf>

[e] *74-Series Logic Ratings*: <https://www.esr.co.uk/components/digital-74.htm>

[f] *1N4148 Datasheet*: <https://www.vishay.com/docs/81857/1n4148.pdf>

## Appendix A. Commented Code for Arduino MKR WiFi 1010

The following code was used on the Arduino MKR WiFi 1010 to generate an 8MHz clock signal, read from a potentiometer using that to set the duty cycle of a servo PWM-pulse, and outputting these waveforms to pins 7 and 6 respectively. **Code marked in RED came from source [8] and is not our own.**

```
/*
 * Aditya Mehrotra | ADIM@MIT.EDU
 * Final Project w/ Independent Inquiry
 * 6.1311 Power Electronics Lab
 * Department of Electrical Engineering and Computer Science
 * Massachusetts Institute of Technology
 *
 * Code Description:
 * This code takes a 0-5V input from a potentiometer,
 * uses it to set the duty cycle of a servo and sends
 * that signal to output pin 6.
 *
 * At the same time code by "MartinL" on Arduino Forums
 * outputs an 8MHz clock signal from pin7.
 *
 * Outside the Arduino, a 74LS08N (non-inverting AND Gate)
 * compares the 8MHz clock and the Servo signal to perform
 * modulation. This then goes into a capacitor before transfer-
 * ing through the charging coil.
 *
 * We chose this gate as it has a max operating frequency of
 * 35MHz.
 *
 * The goal is to wirelessly control a servo motor with this
 * signal after the demodulation on the other side.
 */

//Just some generic setup code
#include <Servo.h>
int servopin = 6;
int inpin = A1;
Servo output_signal = Servo(); //create servo/PWM generator

//setup code to run once
void setup()
{
  pinMode(servopin, OUTPUT); //sets the servo pin at output
  pinMode(inpin, INPUT); //sets the potentiometer pin as input
  output_signal.attach(servopin); //attaches servo object to pin

  //ALL OF THIS IS TO GENERATE THE 8MHZ CLOCK
  //DID NOT WRITE THIS CODE - CANT TAKE CREDIT
  // Output 8MHz, 50% duty-cycle PWM on timer TCC0

  GCLK->GENDIV.reg = GCLK_GENDIV_DIV(1) | // Divide the 48MHz clock source by divisor 1: 48MHz/1=48MHz
    GCLK_GENDIV_ID(4); // Select Generic Clock (GCLK) 4
  while (GCLK->STATUS.bit.SYNCBUSY); // Wait for synchronization

  GCLK->GENCTRL.reg = GCLK_GENCTRL_IDC | // Set the duty cycle to 50/50 HIGH/LOW
    GCLK_GENCTRL_GENEN | // Enable GCLK4
    GCLK_GENCTRL_SRC_DFLL48M | // Set the 48MHz clock source
    GCLK_GENCTRL_ID(4); // Select GCLK4
  while (GCLK->STATUS.bit.SYNCBUSY); // Wait for synchronization

  // Feed GCLK4 to TCC0 and TCC1
  GCLK->CLKCTRL.reg = GCLK_CLKCTRL_CLKEN | // Enable GCLK4 to TCC0 and TCC1
```

```

        GCLK_CLKCTRL_GEN_GCLK4 | // Select GCLK4
        GCLK_CLKCTRL_ID_TCC0_TCC1; // Feed GCLK4 to TCC0 and TCC1
while (GCLK->STATUS.bit.SYNCBUSY); // Wait for synchronization

// Enable the port multiplexer for the TCC0 PWM channel 3 (digital pin D7), SAMD21 pin PA21
PORT->Group[g_APinDescription[7].ulPort].PINCFG[g_APinDescription[7].ulPin].bit.PMUXEN = 1;

// Connect the TCC0 timer to the port outputs - port pins are paired odd PMUO and even PMUXE
// F & E specify the timers: TCC0, TCC1 and TCC2
PORT->Group[g_APinDescription[7].ulPort].PMUX[g_APinDescription[7].ulPin >> 1].reg |= /*PORT_PMUX_PMUXO_F */
PORT_PMUX_PMUXO_F;

// Normal (single slope) PWM operation: timer continuously count up to PER register value and then is reset to 0
TCC0->WAVE.reg |= TCC_WAVE_WAVEGEN_NPWM; // Setup single slope PWM on TCC0
while (TCC0->SYNCBUSY.bit.WAVE); // Wait for synchronization

// Each timer counts up to a maximum or TOP value set by the PER register,
// this determines the frequency of the PWM operation:
TCC0->PER.reg = 5; // Set the frequency of the PWM on TCC0 to 8MHz
while(TCC0->SYNCBUSY.bit.PER); // Wait for synchronization

// The CCx register value corresponds to the pulsewidth in microseconds (us)
TCC0->CC[3].reg = 3; // TCC0 CC3 - 50% duty cycle on D7
while(TCC0->SYNCBUSY.bit.CC3); // Wait for synchronization

// Divide the 48MHz signal by 1 giving 48MHz (20.8ns) timer tick and enable timer TCC0
TCC0->CTRLA.reg |= TCC_CTRLA_PRESCALER_DIV1; // Divide GCLK4 by 1
TCC0->CTRLA.bit.ENABLE = 1; // Enable the TCC0 output
while (TCC0->SYNCBUSY.bit.ENABLE); // Wait for synchronization

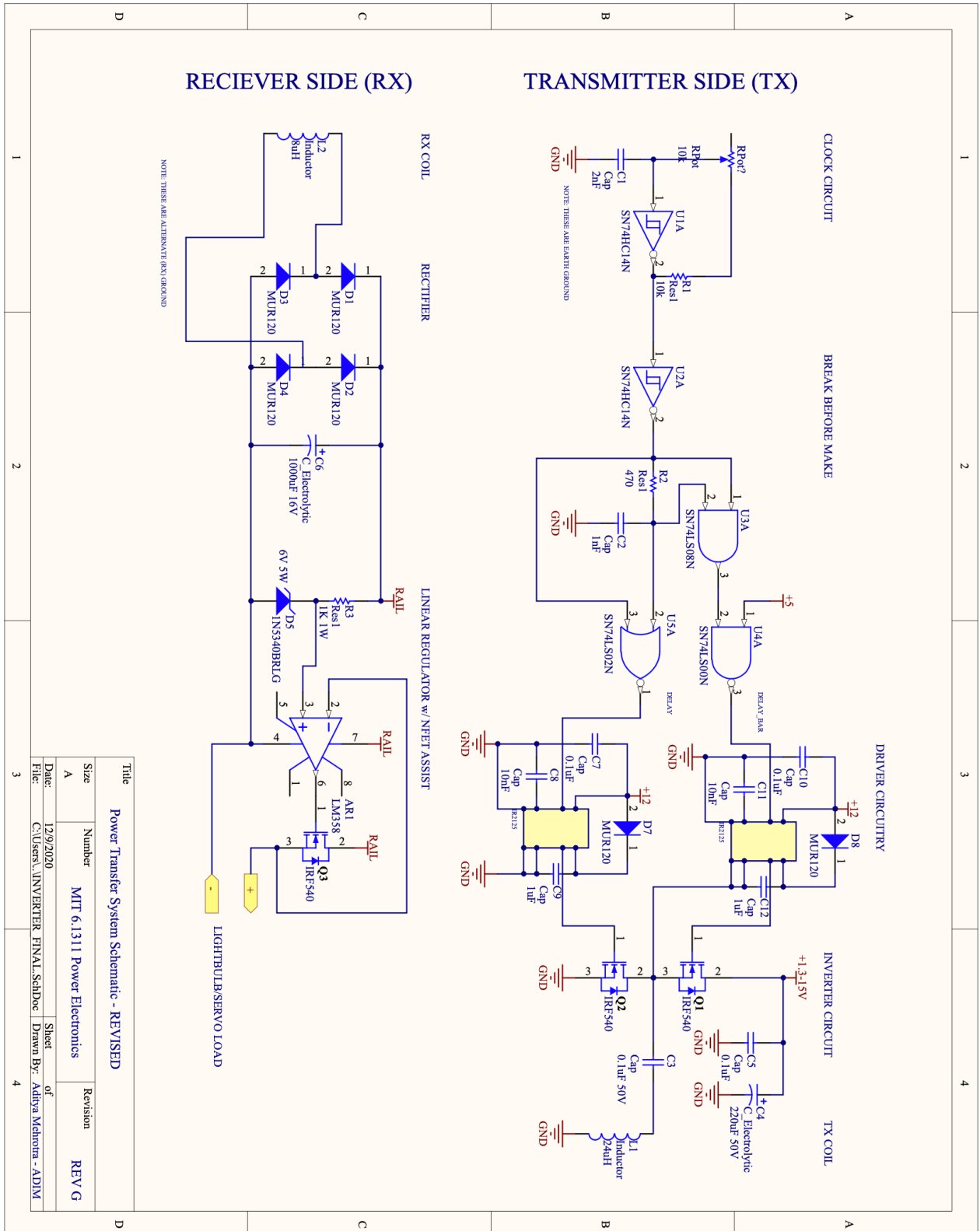
//END_CODE WE DIDN'T WRITE - REST IS OURS
}

//loop code to run while on
void loop() {
  int val = analogRead(inpin); //read from potentiometer
  int servo_val = map(val, 0, 1023, 0, 255); //map analog to digital 0-255 value
  output_signal.write(servo_val); //write value to servo
}

```

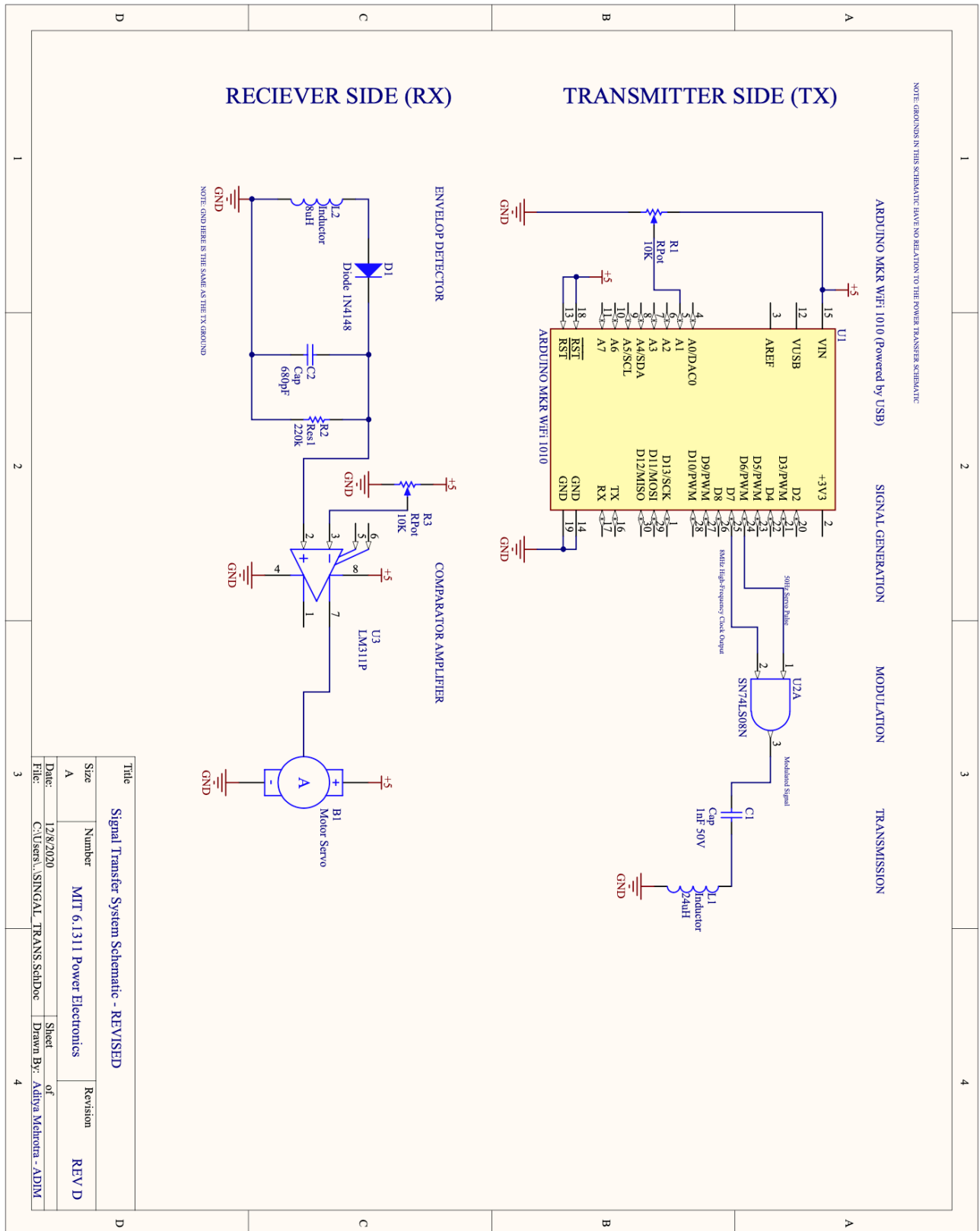
# Appendix B. Large-Format Schematics

## A.B.I Large Format Schematic of Power System in Altium Designer.



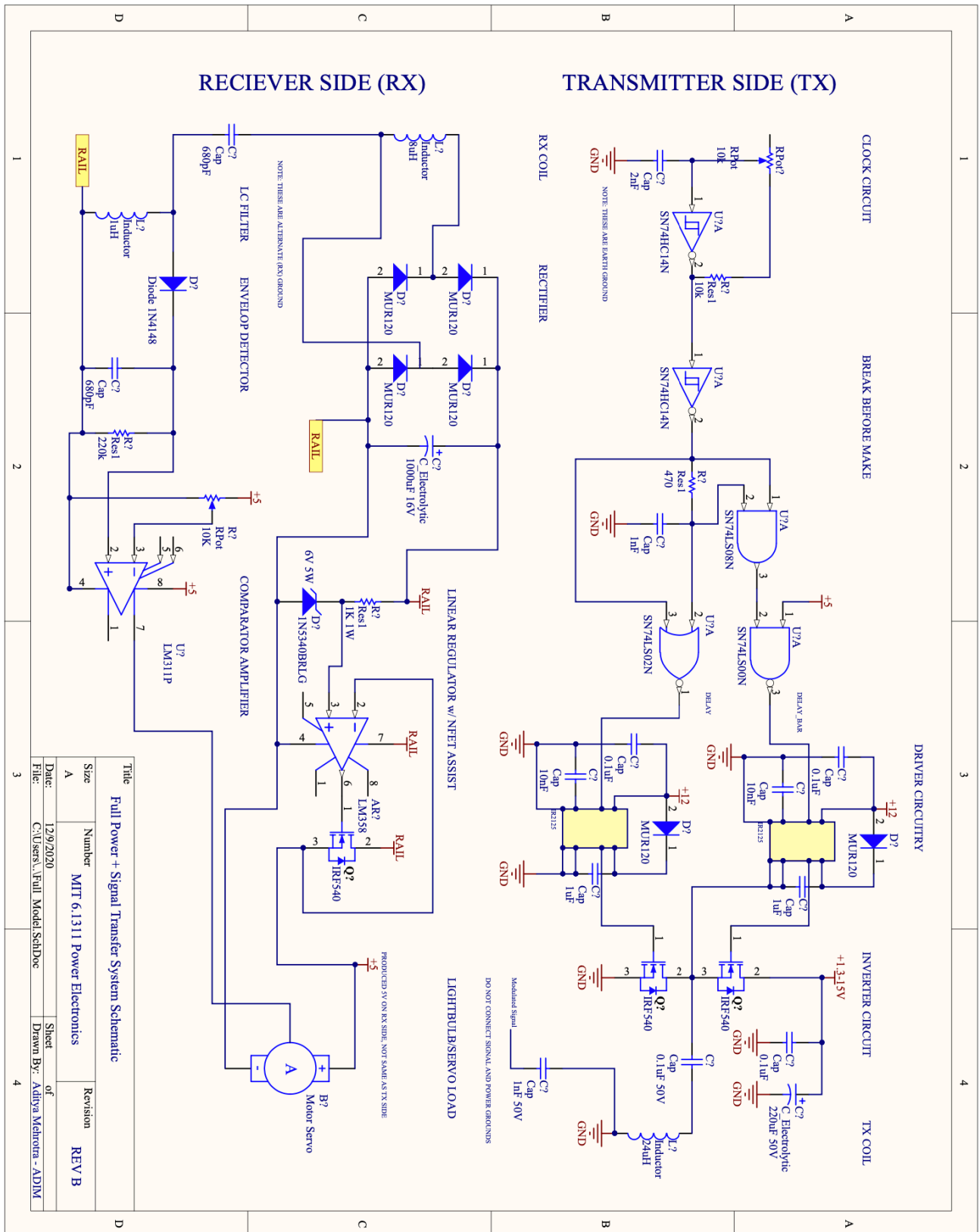
Title		Revision	
Power Transfer System Schematic - REVISED		REV G	
Size	Number	MIT 6.1311 Power Electronics	Revision
A			
Date:	12/9/2020	Sheet	of
File:	C:\Users\ADIVYA\INVERTER_FINAL_SchDoc	Drawn By:	Aditya Mehra - ADIM

**A.B.II Large Format Schematic of Signal Transfer System in Altium Designer.**





**A.B.III Large Format Schematic of Combined Signal + Power Transfer System in Altium Designer.**



## Appendix C. MATLAB Code for Various Graphs + Calculations

```
clear all
close all
clc

%CODE TO CREATE THE BODE PLOT
s = tf('s')

%w = 2*pi*82000;
C=100*10^(-9);
L=24*10^(-6);
R=0.1; %max of 100mOhms

Z1 = (s*L+R)

H = Z1/(1/(s*C)+Z1);
figure;
bode(H)

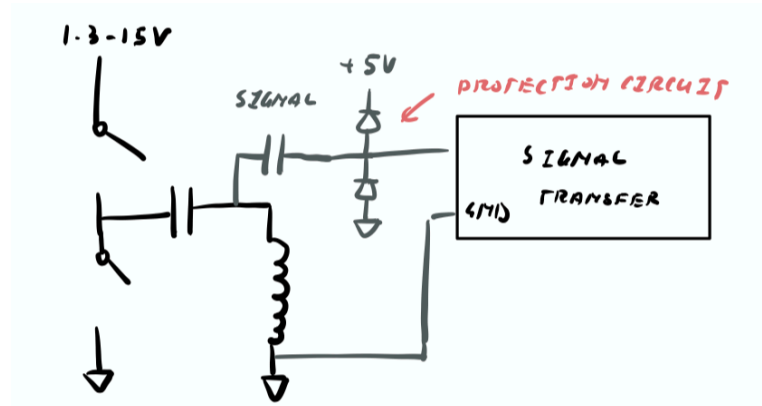
%CODE TO PLOT THE RECTIFIER OUTPUTS
syms x
figure;
hold on
title('Output From Rectifier to Average Value')
ylabel('Voltage (V)')
xlabel('Electric Angle (rad)')
V = 13
fplot(abs(V*sin(x)),[0,8*pi])
avg = int(abs(V*sin(x)), x, 0, 2*pi)/2/pi
fplot(avg, [0,8*pi])

%CODE TO SOLVE A SIMPLE LC FREQUENCY PROBLEM
syms C1
L1 = 24*10^(-6)
eq1 = 1/sqrt(L1*C1)/2/pi == 8000000
solve(eq1,C1)
```

## Appendix D. Proposed Revisions to the Project

After this project was completed we do have some proposed revisions that may allow it to work more successfully in the future should it be re-attempted.

(1) First, utilize the below schematic (or similar) for injecting data into the coil so that the ground for data transfer can be connected to the ground for the power circuit.



(2) Second, amplify the data signal on the TX side of the system so that we may be able to filter it from noise more easily on the RX side of the coil.

(3) On the RX side of the coil, utilize a half-wave rectifier instead of a full-wave rectifier and a larger storage capacitor to ensure that we can make one of the rails of the circuit “ground” so the data signal has a reference. We determined the lack of signal “ground” could have been one of the reasons the servo didn’t end up accepting the signal.

(4) When constructing the circuit, just generally manage the parasitics better to reduce noise. Find a way to reduce noise on the input power rail as well, possibly using better or more bypassing.

We think that by doing the above, not only would we receive both power and signal through the coil system, but also be able to accurately control the servo motor as was the end goal of the project. These are possibilities for future work.