**W.E.A.R**

**Wireless Energy Autonomous Robot**

**Department of Electrical Engineering and Computer Science**

**University of Central Florida**

**Spring 2016**

**Senior Design I**

**Group L**

**Blake Emerton (CpE) BEmerton@knights.ucf.edu**

**Wilson LaLeau (EE) Olivierlaleau@knights.ucf.edu**

**Trent Smith (EE) Trent.s.2012@knights.ucf.edu**

**Dimitri Wilks (EE) dimitriwilks@knights.ucf.edu**

**Table of Contents Page**

**1 Executive Summary 1**

**2 Project Description 1**

**2.1 Project Motivation 1**

**2.2 Objectives and Goals 2**

**2.3 Project Requirements 2**

**2.4 Block Diagrams 3 2.5 Materials and Budget 6**

**3 Research 7**

**3.1 Similar Projects 7**

**3.2 Receiver 9**

**3.3 Transmitter 28**

**3.4 Microcontroller 36**

**3.5 Microprocessors 42**

**3.6 Autonomous Control 44**

**3.7 Sensors 47**

**3.8 Robot Platform 52**

**3.9 Motor and Motor Driver 53**

**3.10 Batteries 55**

**3.11 Track Platform 57**

**4 Standards 60**

**4.1 Standards Search 60**

**4.2 Design Impact 60**

**5 Hardware Design 61**

**5.1 Microcontroller 61**

**5.2 Microprocessors 62**

**5.3 Receiver 63**

**5.4 Transmitter 66**

**5.5 Transmitting and Receiving Coils 67**

**5.6 Coils 69**

**5.7 Coils Enclosure 71**

**5.8 Oscillator Design 72**

**5.9 Oscillator Parts 75**

**5.10 Image Processing 76**

**6 Software Design 77**

**6.1 Computer Vision 78**

**6.2 Control System 80**

**6.3 Software Integration 82**

**7 Prototype Construction 83**

**7.1 Parts Acquisition 83**

**7.2 Breadboard Testing 84**

**7.3 PCB Design 84**

**7.4 Hardware Build Plan 97**

**7.5 Software Build Plan 98**

**7.6 Track Build Plan 99**

i

**8 Prototype Testing 100**

**8.1 Hardware Testing Environment 100**

**8.2 Hardware Specific Testing 100**

**8.3 Software Testing Environment 108**

**8.4 Software Specific Testing 108**

**8.5 Energy Transfer Testing 115**

**9 Administrative 119**

**9.1 Project Milestones 119 9.2 Project Budget 120**

**9.3 Project Summary 122**

**Appendices 123**

**A. Copyright Permissions 123**

**B. Works Cited 124**

**C. Diagrams 124**

ii

**1 Executive Summary**

The purpose of this project is to use the knowledge that we have accumulated during our time at the University of Central Florida to create a multi-dimensional project. We will oversee this project from its initial idea through to research, design, prototype build, testing, and finally its conclusion at the design showcase. We wanted to incorporate our interests along with current industry trends. There is currently a huge interest in electric vehicles and Tesla is evidence of this. The only drawback to electric vehicles is that they cannot travel long distances without have to stop and recharge. If they could somehow keep driving continuously if would solve this problem.

This led us to the idea of a wireless energy autonomous robot. It will navigate around a track which will provide it with wireless energy. The robot will be equipped with a camera unit which will handle the lane detection and send the corresponding data to a microcontroller. This microcontroller will then send signals to the robot motors to navigate around the figure eight track. The robot will also have an ultrasonic sensor to help it avoid a collision with a stopped object in its path. The wireless energy will be provided by induction coils placed a specific intervals underneath the track. There will be a receiving unit on the robot to continuously top off the battery and greatly extend its range.

By combining these different disciplines we will create a Wireless Energy Autonomous Robot or W.E.A.R.

**2 Project Description**

This section will contain a brief overview of the project. It will also include the project motivation, objectives and goals, requirements, and project responsibilities.

**2.1 Project Motivation**

Electricity is becoming one of the more essential sources of energy as more and more objects are being evolved to electric. In order to make the process easier, engineers are working on new ways to transmit energy wirelessly yet, this technology has been limited to powering static object. But, this project focuses on charging a moving robot wirelessly. The design will include a system capable of producing wireless signals to transmit energy over a small distance. The system will consist of an autonomous robot which will guide itself along a track and avoid any objects on the course.

Another motivation for this project comes from the need for extended range on electric vehicles. This is a miniaturized version that could incite the possibility of the use of autonomous electric vehicles in the future capable of being charged while moving.

**2.2 Objectives and Goals**

The robot will use a camera in order to view its environment using image processing and, it will guide itself along the track using sensors to detect the presence of any object to avoid them. As for the wireless energy, the use of radio waves seems like the best option available. To accomplish this, a transmitter will be used to produce small induced currents that will be running past a bandpass filter.

**2.3 Requirements and Specifications**

- Electronic Receiver

- Transmit Coil

- Receiver Coil

- Electronic Receiver (Robot car)

- Autonomous Camera System

- Track (for Robot)

- Addition- Solar Panel

**Constraints**

Some constraints will be the use of the transceiver to meet the desired voltage needed for the receiver such that it can produce normal results and how much wireless energy can be sent over to the robot, the loss of energy that will occur from whatever power source we decide due to eddy current, and the energy loss per distance.

**Transmitter**

In order to transfer the power to the robot wirelessly, we must build a transmitter circuit that will convert direct current power source into alternating current, so that it will pass through the transmitter coils. The transmitter circuit will consist of resistors, transistors, inductors, capacitors, Zener diodes and switches. This circuit will be connected to transmitter coils, where the current will be induced and create the magnetic field. The magnetic field carries the current to the receiving transmitter to be generated and apply power into the robotic system. To accomplish this the resonant frequency of the transmitter needs to be matched closely with the resonant frequency of the receiver circuit.

**Robot**

To be able to adjust the weight, a strain gauge will be inputted. However, if the strain gauge input exceeds the max voltage the IC is able to take, then a scale needs to be applied. To scale the voltage down, a voltage divider will be best. With this it will adjust the amount of wireless energy that may be needed to power the servo motors. One of the main issues that will be faced with this is the amount of wireless energy that we need to be able to get to power the servos.

**Milestones**

Spring 2016

1) Research and Development

2) Circuit Design

3) Begin Electrical Prototyping

4) Begin Software Prototyping

5) Obtain Parts

6) Design Paper

Fall 2016

1) Assemble Robot

2) Begin Programming for Mapping and Object Tracking

3) Start making Wireless Power Circuit

4) Assemble Robot Track

5) Finish Prototype

6) Test Prototype

7) Make Necessary Changes to Hardware and Software

**2.4 Block Diagrams**

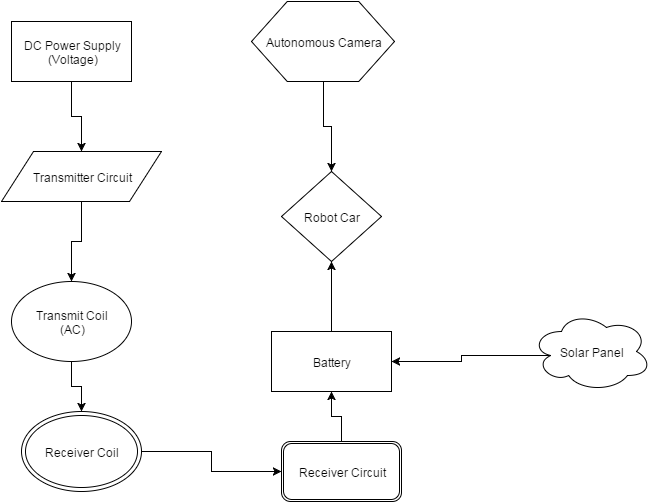


Figure 2.4a Project Overview

**Wireless Energy Block Diagram:**

Trent Smith



Figure 2.4b Wireless Energy

**Transmitter**

Dimitri Wilks

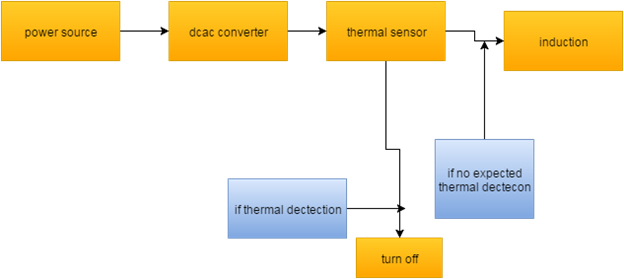


Figure 2.4c Transmitter

**Robot Hardware Block Diagram:**

Wilson Laleau

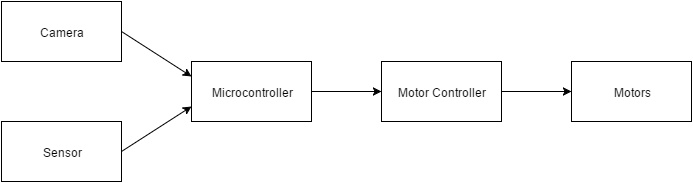


Figure 2.4d Hardware

**Software Block Diagram:**

Blake Emerton

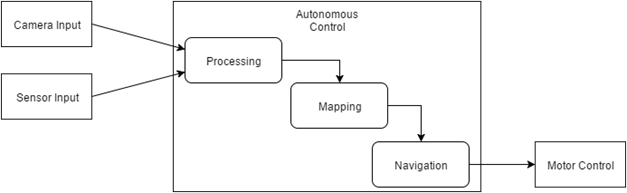


Figure 2.4e Software

**2.5 Project Budget and Materials**

|  |  |
| --- | --- |
| Material | Cost in Dollars |
| Resistors | .14 per type |
| Capacitors | .15 per each |
| Transformer | 13.81 |
| servos | 3.50 |
| inductors | 3.43 per each |
| Integrated Circuit | 5.00 |
| IR sensors  IR receivers | 2.00 each  2.00 each |
| Camera | 50.00 |
| Plywood | 3.57 |
| Plexiglass | 20.00 |
| Robot Platform | 70.00 |
| Total | 240.00-300.00 |

**3 Research**

This section includes all of the research we did before the design of the project. Many of these sections will not be used for design, but they were all vital in determining the correct parts and methods to use.

**3.1 Similar Projects**

This section describes many projects that are similar to the one we plan to build. We did not find any projects autonomous robots that used wireless energy as a supplemental power source, so we included the projects that incorporated an autonomous design or wireless energy.

**3.1.1 ASR**

The Autonomous Sentry Robot is a University of Central Florida Senior Design Project from 2015 and was created by Brian Dodge, Nicholas Musco, and Trevor Roman. It is able to map an unknown room and localize itself on the map. It can determine if a person has entered the room. Once the person is detected, it alerts the user and allows the user to control the vehicle. This project uses autonomous control, mapping and localization, object avoidance, object detection, and remote operation.

**3.1.2 BRAVO**

B.R.A.V.O. or “Bring Reliability to Autonomous Vehicle Operation” is a UCF Senior Design Project that was created by Christopher Cox, Douglas Akinola, Mirazam Usmanov, and Henry McWhorter. The robot is a fully autonomous vehicle that incorporates the use of a remote image processing unit as well as onboard sensors to navigate a predetermined course, navigate alternate routes involving 90 degree turns and stop signs, detect potential collisions, and obey the basic “rules of the road.”

**3.1.3 T-100 Watchdog**

The T­100 Watchdog is a University of Central Florida Senior Design project from 2014. The project was created by Ismael Rivera, Journey Sumlar, Chris Carmichael, and Warayut Techarutchatano. The T­100 Watchdog is a home security robot which can detect and track targets. One of the goals of the project was to use OpenCV algorithms and a thermal camera to detect movement and then track a specific target. As the vehicle tracks and moves towards the targets, it autonomously maneuvers across a room while avoiding obstacles that might be in its way. The Watchdog also has a webcam to relay images and video to the user via a wireless communication system. While this project has many similar goals to ours, like autonomous control, obstacle avoidance, and target detection and tracking using OpenCV, it does not share one of our main goals, which is the robot being supplied with wireless energy. Another difference is that the T-100 Watchdog only reacts to movement and our robot will be actively following a highway inspired track.

**3.1.4 Charge Spot**

The charge spot is a UCF design project from fall 2013 - spring 2014. The project was created by Ryan Johnson, Theophilus Essandoh, and Emelio Watson. The charge spot is a wireless charging station that is used to charge the battery of an electric vehicle. In the project they had a receiving and transmitting coil that would transmit to a battery in effectively charging it up. This project holds many similarities to the wireless energy system that we will be producing. While this project is similar to the wireless energy it does not hold the track nor the autonomous robot that we will be working on. This is an awesome project to be used as a guide so that we can determine what will be an issue and what will not be an issue with the wireless energy. I already learned so much from looking over this project such as inductive coupling and that we need to be weary for any heating issues.

**3.2 Receiver**

Receiver goals

1. Carry power from point a to point b from a wireless transmission of energy whether it is a laser, a light, a sound, or through induction.
2. Show the usefulness of wireless energy and how it can go green.

For the receiver there are a vast number of applications that can be put to use. The goal of this section is to explain some of these applications in further detail to find the best selection.

**INDUCTIVE COUPLING (COILS)**

A magnetic field is created when a conductor has a current running through it. To increase the magnetic strength one needs to put the conduct in loops such that the magnetic wave is amplified in a specific direction. Doing this can control the strength of the magnetic field being sent out.

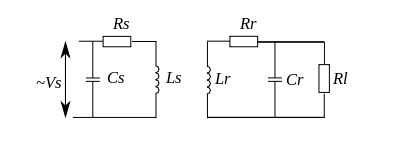


Figure 3.2a Inductive Coupling

**Example of inductive coupling**

Now if a sinusoidal wave is applied there is a frequency that the magnetic wave will have. This frequency can be used to detect a specified magnetic wave using a bandpass filter. With this methodology it is possible to use multiple inductors at a specified strength for different receivers to detect and harness the power of the magnetic wave.

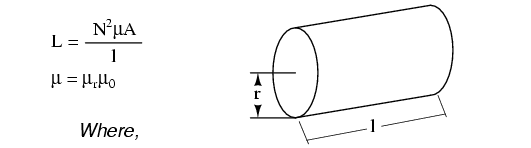


Figure 3.2b equation to determine the inductance

Since the inductor value is determined RLC band pass frequency and the ISM band frequency, we need to discuss some problems that might arise that the inductor may be too big for the robot.

So to prevent this problem the equation w0 = 1/sqrt(LxCx) can be used to adjust the capacitor so that the coil can be fit the robot and the frequency can meet the desired resonant frequency.

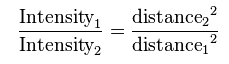
**Equations**

The goal of these equations is to find a good frequency and use it to find the right capacitors and resistors values. Not only that but these equations are to ensure that no noise will conflict with other parts of the receiver. This will help prevent spikes in the bandpass filters.

**Loss of energy**

The equation below is the loss of energy. As shown, the loss of energy in the system is huge and we need to somehow make up for the energy that is being loss. Not only that but careful consideration is needed for the track so that the energy sent over isn’t affected by the distance of the robot compared to the transmitter. Overall this loss is possibly the hardest part to take note of for the design of the robot.

To help compensate for the loss extra power will be sent, so that the loss of energy will not affect the voltages of the signal that is needed to charge the robot.



**3.2.1 BANDPASS**

To further understand the project, these are the elementary circuits that will be used and consider for the design of the robot’s wireless receiver and transmitter.

**Useful equations**

w0 = sqrt(wcl\*wch) βn =wcl-wch Q = w0/βn

H(s) = (s/w0)/((s/wo)2+(1/Q)\*(s/w0)+1))

X is a constant

((9+sqrt(17))/8)-1) wh(n) = beta(n)

w0=wh(sqrt(9+sqrt(17))/8)

wh(n+1) = wh(n)\*((9+sqrt(17))/8)

Wl = (9+sqrt(17))\*wh/8

W0 < wl

w0>wh

Wl > wh

**Spacing(x)**

Note: x has to be small enough such that the bandwidth is small, and importantly, x will be diagnosed as the spacing for frequency relative to the ISM band.

There are some ways that we would like to investigate to help prevent this problem. The first way is to isolate the first filter with a second filter which help narrow down the frequency. To do this we can use a HPF or a LPF to help narrow out the frequencies.

**Antenna length**

The wave equation velocity = lambda\*resonant frequency can be used to obtain the length of the receiving coil.

c = lambda\*f

lambda = c/f = c\*2\*pi\*Lx\*Cx

This should be the length of antenna that we will need.

**Critically Damped**

This method is the best and most efficient, however there is an issue in which there is an exact value, and because of this the condition will likely be unsatisfied mainly due to the small amount of noise caused by the resistance of a wire. Despite this, our main goal is to get as close as we can to this value.

The critically damped filter has the fastest time to reach the correct amplitude with little overshoot in time.

Q = ½ => wl = (9+(sqrt(17)))\*wh/8)=1.64\*wh

**Under Damped**

Underdamped can be used in the design of the bandpass filter because it does reach correct amplitude quickly, but has issues with overshoot in time.

q<½

wl<(9+(sqrt(17)))\*wh/8)<1.64\*wh

**Over Damped**

The overdamped filter has no oscillations, however, the time may be longer for the function to get to the amplitude.

Q >½

wl>(9+(sqrt(17)))\*wh/8)>1.64\*wh

**RC**

Wl = 1/(RLCL)

Wh = 1/(RHCH)=1/1.64\*(RLCL)

**RLC circuit**

Assume R and L are in the high pass

Wo =1/LC

wl = R/2L+sqrt((R/2L)^2+w0^2)=1.64(R/2L-sqrt((R/2L)^2+w0^2))

Assume R and L are in the low pass

wh = R/2L-sqrt((R/2L)^2+w0^2)

**RL**

Assume R and L are in the low pass

wl = RH/LH

**Frequencies**

**Band pass filters**

fL > fh for a band pass filter the frequency of the low pass filter has to be greater than the frequency of the high pass filter. Thus for this particular band pass filter the cutoff frequency has to be.

Now there are two types of bandpass filters. The first is the active filter and the second is the passive filter.

wh=RH/LH

**3.2.4**

In this section different RLC values for the different types of band pass filters will be shown as well as the simulation of how these filter act within our circuit.

**RLC**

In the table below is the range of frequencies that the ism band will use and their corresponding resistors, inductors, and capacitors

|  |  |  |  |
| --- | --- | --- | --- |
| Frequency range | Resistor | Capacitor | Inductor |
| 13553 kHz-13657kHz | 16MΩ | 15pF | .01mH |
| 26957kHz-27283kHz | 3.6kΩ | 3.3pF | 10u |
| 40.66Mhz-40.7Mhz | 51kΩ | .15pF | 100uH |
| 433.05MHz-434.79MHz | 510kΩ | .0015 pF | 100uH |
| 902MHz-928MHz | 1.1MΩ | .00033pF | 100mH |

**RLC continued**

In this section the resonant and cutoff frequencies are obtained using the equations from above. When dealing with higher values it is imperative to get as close as you can to the nearest Hz, because 1 KHz off and the filter will lose some power and could possibly show a decrease in voltage.

|  |  |  |
| --- | --- | --- |
| Resonant frequency | HP cutoff | LP cutoff |
| 13605kHz | 10623720.152 Hz | 17422901.04928 Hz |
| 27120kHz | 21177162 Hz | 347305450.68 Hz |
| 40.68MHz | 31.765743168MHz | 52.0958187 |
| 433.92MHz | 338.8345937MHz | 555.68873382137MHz |
| 915 MHz | 714.4949606403MHz | 1171.771735450211MHz |

Transmitter Inductor capacitor

In the table below is some more high and low pass frequencies for the ISM band range.

|  |  |  |  |
| --- | --- | --- | --- |
| Low | High | Center frequency | Frequency range (ISM band range) |
| 17364.32 khz | 10588 kHz | 13560 kHz | 13553 kHz-13657 kHz |
| 34730.28 khz | 21177 kHz | 27120 kHz | 6957 kHz-27283 kHz |
| 52.0946M | 31.765 MHz | 40.68 MHz | 40.66Mhz-40.7Mhz |
| 206.585 Mhz | 338.8 MHz | 433.92 MHz | 433.05MHz-434.79MHz |
| 1171 Mhz | 714.5 Mhz | 915 MHz | 902MHz-928MHz |

Note the desired cutoff frequencies are for making sure the circuit is critical damping

High pass filter

|  |  |  |
| --- | --- | --- |
| freq | Resistor high | Inductor high |
| 10588 kHz | 6.8MΩ | 100mH |
| 21177 kHz | 13MΩ | 100mH |
| 31.765 MHz | 20MΩ | 100mH |
| 338.8 MHz | 220MΩ | 100mH |
| 714.5 Mhz | 430MΩ | 100mH |

Low pass filter

|  |  |  |
| --- | --- | --- |
| freq | Resistor low (ohms) | Inductor low |
| 17364.32 kHz | 11MΩ | 100mH |
| 34730.28 kHz | 22MΩ | 100mH |
| 52.0946M | 33MΩ | 100mH |
| 206.585 MHz | 130MΩ | 100mH |
| 1171 Mhz | 750 MΩ | 100mH |

**RC high pass filter**

In the table below here is some of the resonant frequency and the high pass filter cutoff frequency for the RC band pass filter. In here is the resonant frequency where the most power is being delivered and

|  |  |  |  |
| --- | --- | --- | --- |
| Resonant frequency | Frequency (high cutoff) | Resistor | Capacitor |
| 13605 kHz | 10623720.152 | 120Ω | 100pF |
| 27120000 | 21177162 | 75Ω | 100pF |
| 40.68MHz | 31.765743168MHz | 39Ω | 100pF |
| 433.92M | 338.8345937MHz | 360Ω | 1pF |
| 915 MHz | 714.4949606403MHz | 180Ω | 1pF |

In the table below is the cutoff frequency for the electromagnetic wave that was derived from our resonant frequency using the previous equations.

RC low pass filter

|  |  |  |
| --- | --- | --- |
| Frequency (cutoff) | Resistor | Capacitor |
| 17422901.04928 | 91Ω | 100pF |
| 347305450.68 | 47Ω | 10pF |
| 52.0958187M | 300Ω | 10pF |
| 555.68873382137MHz | 30Ω | 10pF |
| 1171.771735450211MHz | 13Ω | 10pF |

**Simulations & Schematics**

In the simulation below is the receiver to wireless energy in it we tested the different types of filters.

**Passive Filter**

**RC**

Below shows the RC passive circuit with a unity gain buffer and a peak detector circuit.

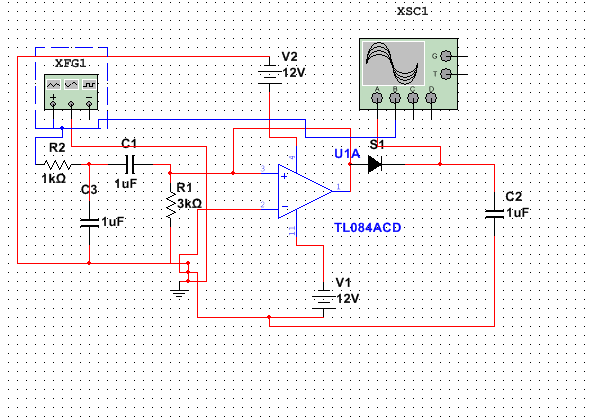


Figure 3.2.1a RC Passive Circuit

This is the result that was obtained when we ran it in the simulation to test how it will run differently from the other filters

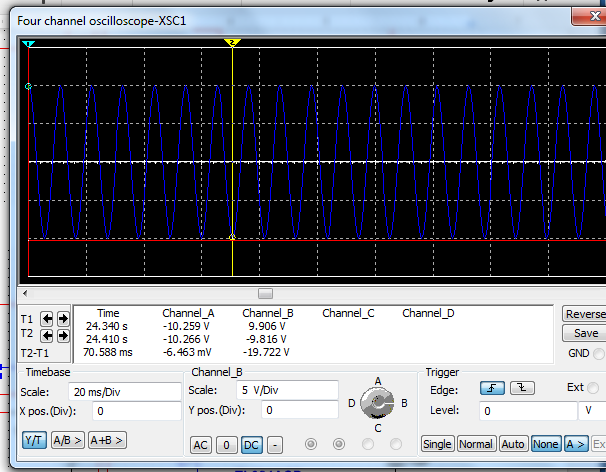


Figure 3.2.1b RC Passive Circuit Waveform

**RL**

Below shows the RL passive circuit with a unity gain buffer and a peak detector circuit.

This is the result that was obtained when we ran it in the simulation

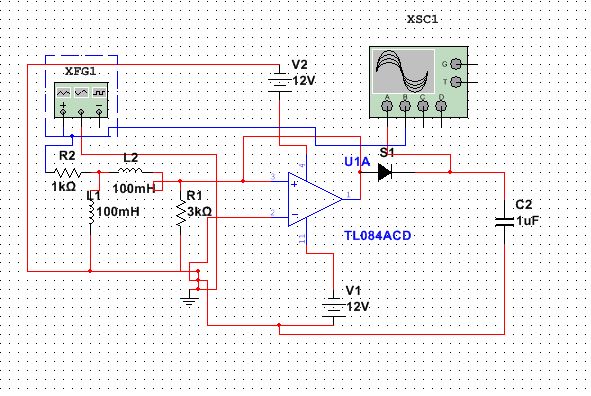


Figure 3.2.1c RL Passive Circuit

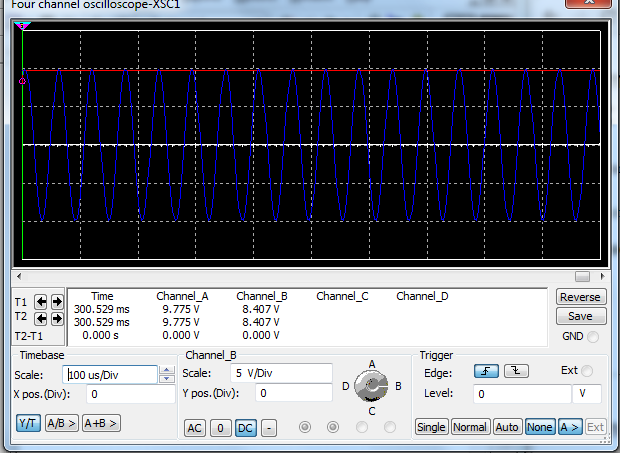
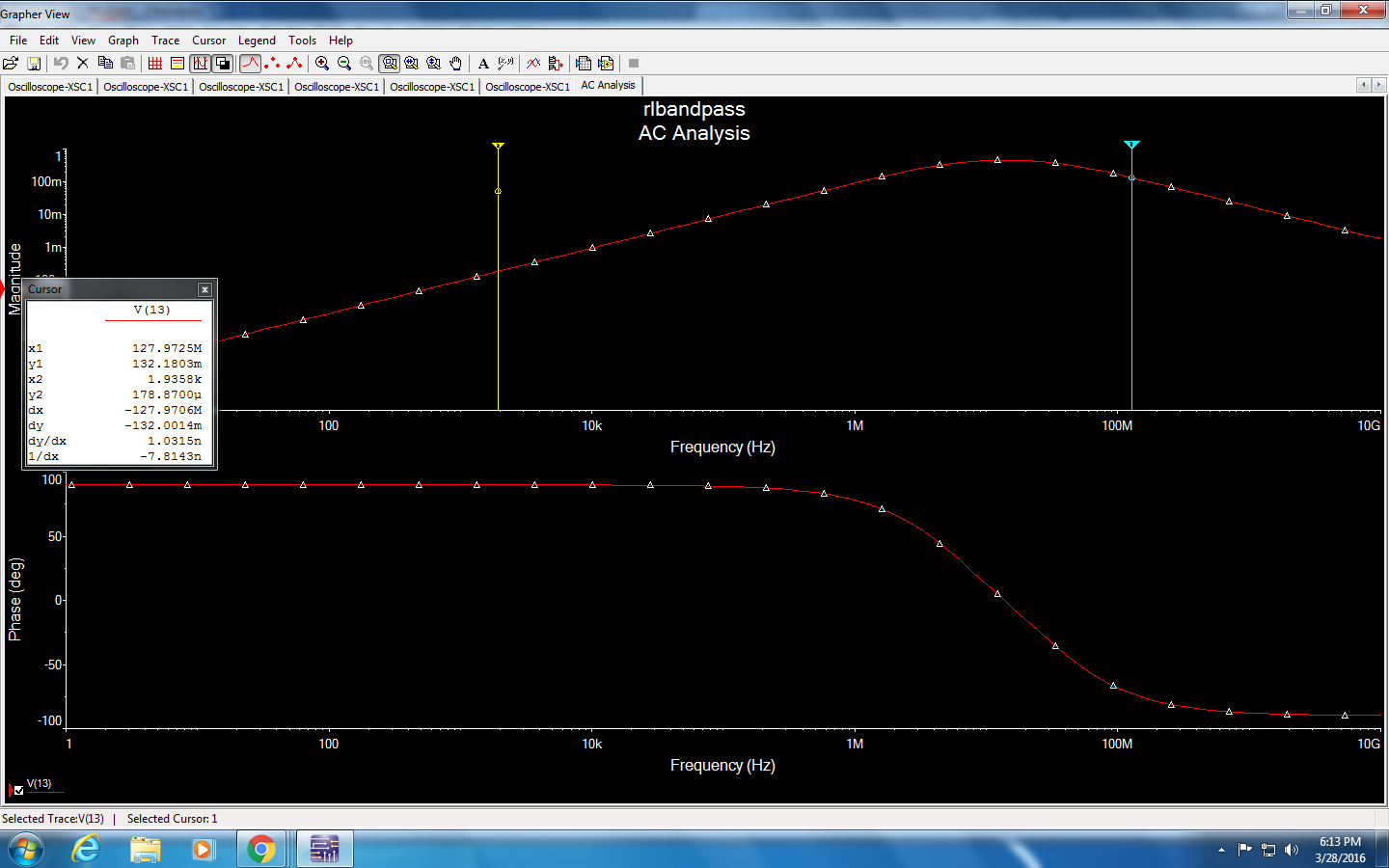


Figure 3.2.1d RL Passive Circuit Waveform

**RL circuit**



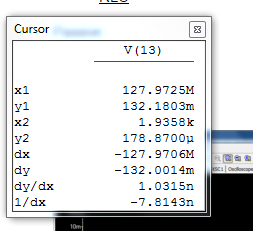


Figure 3.2.1e RL Bandpass

Assume R and L are in the high pass

**RLC**

Below shows the RLC passive circuit with a unity gain buffer and a peak detector circuit.

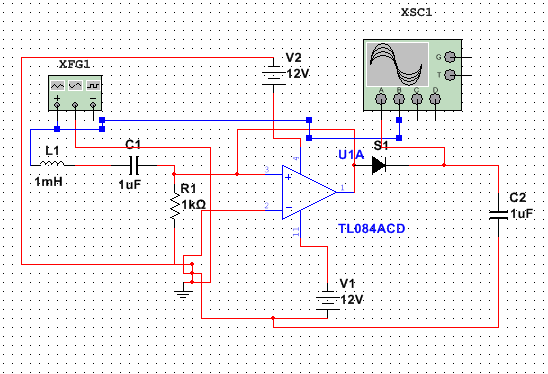


Figure 3.2.1f RLC Passive Circuit

This is the result that was obtained when we ran it in the simulation

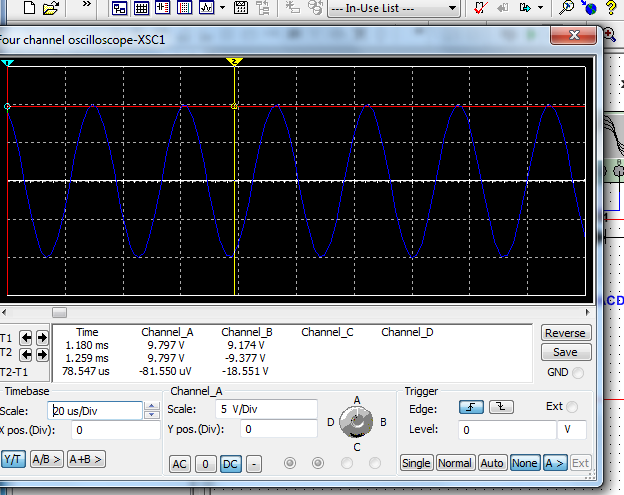


Figure 3.2.1g RLC Passive Circuit

**Active Filter**

There are three choices for the bandpass filter that we want to consider.

1. RC

* Since for the receiver we am going to need a receiving coil that will be able to receive the signal. The project will most likely use RL or RLC bandpass filter

1. RL

* Example is using the 1st resonant frequency

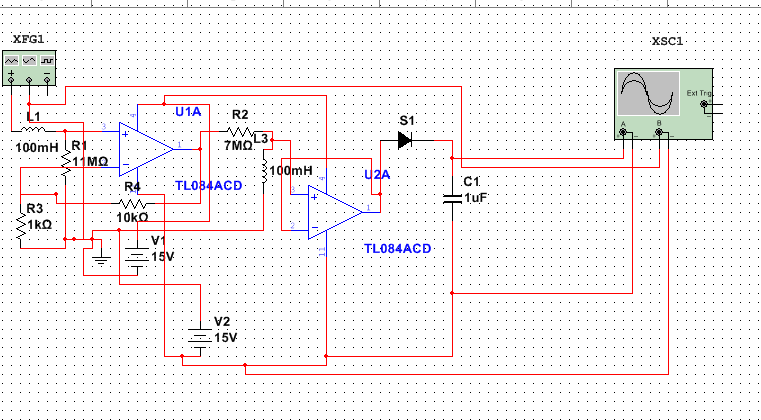


Figure 3.2.1h RL Active Filter Circuit

The RL active filter will look like the circuit shown above

1. RLC

* The RLC portion will look like that of the passive filter except it will be conntected to an amplifier that will amplify the voltage in the signal

**Errors**

While doing calculations we had some problems that we noticed beforehand. There is an exchange for one benefit compared to another benefit. For example, since the resonant frequency has to be different for our method to work, the bandpass cannot be the same, which is what our first method showed. It concluded that the bandpass must be similar for the method to work. So we found another way to put it in equations and fit into the constraints that we had specified to our project.

**Controller**

The method below is describing an approach that we will consider if we need a strong magnetic field. The purpose of this is to have a way transmitting power without damaging any circuit.

Since the amount of bandpass act in a similar way as bytes in the ASCII code, than any number up to the 2^nth v can be produced. The setup not only sends out huge voltages, but also helps prevent the problem of electromagnetic pulses that may cause eddy current in an another circuit. This makes the system able to adapt to the need of voltages by any one robot. While this is something most likely not going to be done in this senior project due to the complexity, it is that we want to be noted. Now to turn on or off these bits a single bit value can be used, to turn a byte on or off. To demonstrate this we used 4 bits.

|  |  |  |  |
| --- | --- | --- | --- |
| 8V | 4V | 2V | 1V |
| 1 | 0 | 1 | 0 |

Will give an amount of about 10 volts, just like the binary system.

With the receiver in the design, we think at most, it will do about 2-3 bits due to complexity. However depending on the amount of power that will be needed we can always adjust the configuration that we will need.

**Switch**

To be able to carry out the binary representation of this we will need to do something similar to switches.

The voltage source in here is used to represent the signal the controller will be using and the diode is used to make the voltage only positive. With this the comparator compares the 5V to the signal such that the signal switches on the 12 v which will be shown on the oscilloscope.

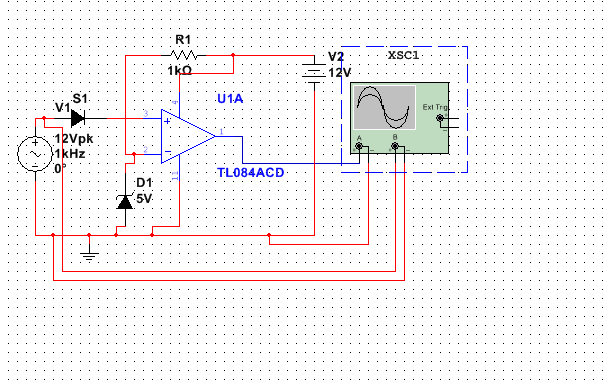


Figure 3.2.1i Switch Circuit

As seen below in the figure, the blue is the output and is only turn on when red is above a specified threshold.

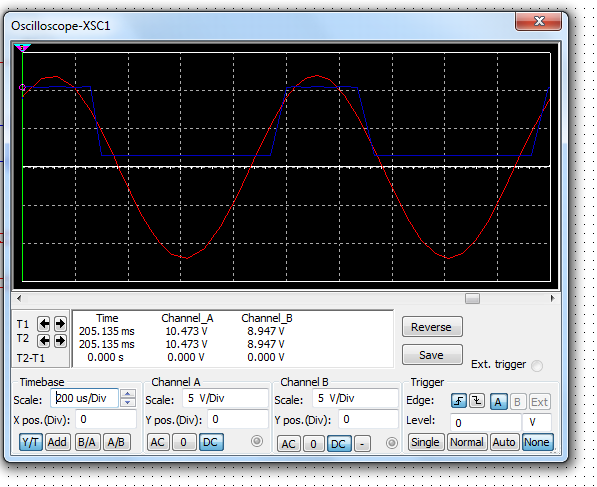


Figure 3.2.1j Switch Circuit Waveform

Also to note that there is a slope between switching this will cause issues for when the device needs to turn on rapidly, so in order to prevent this some research will be need to further investigate this problem.

**Multiple Receivers vs Singular**

**Advantages**

* More receivers are more easily controlled with a MCU than a single circuit.
* Can be used to adjust voltage to match various circumstances
* Great for circuits needing a lot of volt, but send a big magnetic wave out.

**Disadvantages**

* With multiple signals the power being used to transmit signals will diminish more than using a single receiver.
* For singular the main issue is that sending a huge magnetic field over can harm electronics.

**3.2.2 Voltage Follower**

One concern with having multiple bandpass filters is that it will have an internal resistance that will no doubt cause some issues. The voltage follower is mainly used to isolate the previous circuit such that it will not cause an internal resistance that will conflict with another circuit.

Despite this the voltage follower will not be used

**Unity Gain Buffer**

The unity gain buffer will isolate the circuit, such that no circuits interfere with each other. With this, the circuit should be safe from conflicting values so that the correct resonant frequency is obtained.

**3.2.3 Peak detector circuit**

The peak detector circuit is mainly used to change the AC to DC for use, this is incase such a thing needs to be taken into consideration

**3.2.4 Method 2**

Another way of going about designing the receiver is to utilize a mat which charges the robot battery while waiting.

For this particular method we wanted to research more into lasers mainly because of the focus and overall the low loss of energy this application will bring to the table. The little energy loss will give more area for the receiver and transmitter to be apart, and not only that but within a specified area the amount of energy to be transferred will be increased. Such that the energy transferred will be in overall better quality.

I believe this to be the most effective way over induction mainly because lases do not follow the inverse square law, and that if we have the robot over a matt than it will be able to charge its battery.

Now assuming the first method does not work, we want to try and put together this method as a back which was incorporated by Linear Tech.

For method 2, sending energy across with a laser will be most effective.

**3.3 Transmitter**

**3.3.1 Laser**

Laser power beaming is the act of using a laser beam to send concentrated light through space to a receiver that will convert the light into electricity. In 1968, American scientist Peter Glaser proposed using wireless power beaming for solar transmission from earth to space. The process is similar to solar energy but uses intense laser light on photovoltaic cells to convert the light into electricity. The laser is powered by a source, then shaped into optics to adjust beam size to its destination. Lasers power beaming can be manipulated to send energy or information over various distances.

**Pros**

* Narrow beams allows greater amounts of energy concentration at long distances.
* Power will be transmitted with zero radio frequency interferences.
* Flexibility in receiver size which can be integrated into small devices.
* Power can be transferred in areas where it’s uneconomical to have wires running. (Aircrafts. etc.)
* Can use existing power sources to power lasers
* Power can be delivered through free space.

**Cons**

* Laser Power Beaming requires a line-of-sight (direct or reflection off mirrors between the transmitter and receiver.
* Weather could affect the efficiency, range, and resulting in block transmission altogether.
* Longer distances could be achieved but could affect cost and efficiency.
* Stray or scatter lights from the beams can’t be completely eliminated he could affect plants and other animal life.
* Because of sensitivity of the eyes, laser beam can be hazardous to the eyes.

**3.3.2 Magnetron**

Microwave power transfer is the transfer of energy wirelessly through microwaves. In the 1950s electrical engineer William C. Brown, helped invent a crossed field amplifier. A pioneer in wireless microwave energy transmission. The microwave transfer system converts direct current (DC) power to microwaves, then the microwave radiations that is targeted passes through the Coax-Waveguide adaptor. The adaptor sends the radiation to the circulator to reduce exposure to outside power. The radiation passes through the tuner and directional coupler device, which separates the signal according to its direction. The radiation is transmitted over through the transmitter antennae and is received by the antenna at the rectenna. Then the radiation gets passed through a low pass filter, and its matching network. Finally through the rectifier as it gets converted to direct current power.

**Pro**

* The magnetron acts a lot like a laser except instead of photons, the magnetron uses electrons. This is useful because like the laser it is highly focus, but also loses more energy so the magnetron is not exactly the best even though it can be used for wireless energy.

**Cons**

* The ability to transfer microwaves through open space depends on line of sight between the source and the target which can inhibit the transfer of power.
* Factors such as weather and physical objects can affect power transfer.
* The magnetron is single directional so this means that only one target a time can be charging in a sense. Not only that, but due to the fact that the wave bounces any microwave emw could potentially be dangerous.

Microwave Trasmisson.png

Figure 3.3.2 High-Level Magnetron

**Conclusion**

For the magnetron to have any part would be creating a cavity which would have the beam of electrons bounce off the walls of a surface such that it will be hitting the robot.

**3.3.3 Induction**

Electromagnetic induction is the act of producing an electromotive force across a conductor which contains various magnetic fields. Michael Faraday was the great scientist who discovered electromagnetic induction in 1831. It began with a basic experiment using two wires where he wrapped two wires around a transformer. He connected and disconnected the wire to a battery where he saw a transient current, which is a wave of electricity. He explained that the induction happened when there is a change in the magnetic flux. Soon after he introduce Faraday’s law of induction, which means the electromotive force is equal to the rate of change of magnetic flux. In 1894, Nikola Tesla used resonant inductive coupling to wirelessly light incandescent and phosphorescent lambs. He pioneered the idea of wireless energy transfer and patented his high-voltage resonance transformer called the Tesla coil.

In our project, we will display how energy can be transferred through induction coupling. The induction coupling is the interaction of the transmitter and receiver coils. The power transfer happens by creating an alternating magnetic field or flux in the transmitter coil which then gets converted into electric current going into the receiver coil. The transmitter is a variation of components that take electrical energy and turn it into wireless energy for the receiver to pick up. The transmitter contains and induction coils and capacitors. The induction coils act as half of the bridge to transfer wireless energy. There’s capacitors and resistors in the system to help resonate the transmitter. The quality factor or “Q” factor is dependent on the relationship between the resistors and capacitors on the induction coil. The quality factor will determine the efficiency of the energy transfer.

To help with the transmitter we have been using linear technology wireless power user guide to help us with the transmitter. Now within the transmitter the frequency of the wave that is going to send out is w0= 1/(CxLx)

**Pros**

* Elimination of power cords
* Using induction as a wireless energy transmitter is great because it is able to transmit the energy over a couple of source which is great for short distances.

**Cons**

* Energy theft
* The problems with induction is because of the inverse square law which decreases the amount of energy being sent over an x amount of distances. Which is why for long distances using induction is a terrible idea.

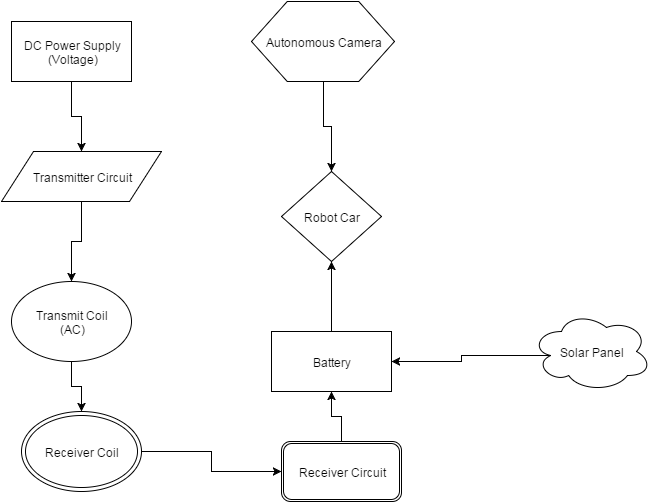


Figure 3.3.3 High Level Induction

**Conclusion**

Overall, the best way to design a working wireless energy to transmit over long distances will to use the laser due to the concentrated beam of photons. As for short distances the best option would be induction mainly because of the multiple targets that induction is able to hit.

**Electromagnetic Pulse (Emp)**

When it comes to energy or particles traveling over a space in time, it can cause some precautions and concerns. One of those concerns are electromagnetic pulses (Emp) that can be very erratic throughout a system. Electromagnetic pulses are short bursts of electrical disturbances that can damage electronics or components within the system. The higher the energy levels traveling throughout the system, the greater chances of eddy currents forming and potentially causes an interference with other components. So to help prevent this we will try to minimize the emps by using low energy such that no circuit will cease to stop operating.

In the guide it states that that emf have the potential to damage circuits so to prevent this it needs a thermal detection system to be integrated with the transmitter . This detection system will turn off whenever elevated temperatures are detected hence turn off the magnetic field.

**Precautions**

Thermal sensing is a great precaution to help prevent emps, and will turn off the transmitter when an elevation in temperature is sensed.

**3.3.4 Thermal**

Thermoelectric energy harvesting come from the thermoelectric effect. Thomas Johann Seebeck took the credit for discovering that electrical energy could be generated from the temperature gradient of thermoelectric material.

**Method 1**

For method 1, a sensor is used to detect an elevation in temperature. To accomplish this we want to hook a comparator to both an infrared signal to read energy reading, and the second to try and pick out a good average temperature of the room to compare it to. To accomplish the later we will need an analog to digital converter hook up to DSP processor such that it takes a bunch of analog values and determines which value is the most accurate. In this circuit there need to be a delay, such that the comparator can do its job.

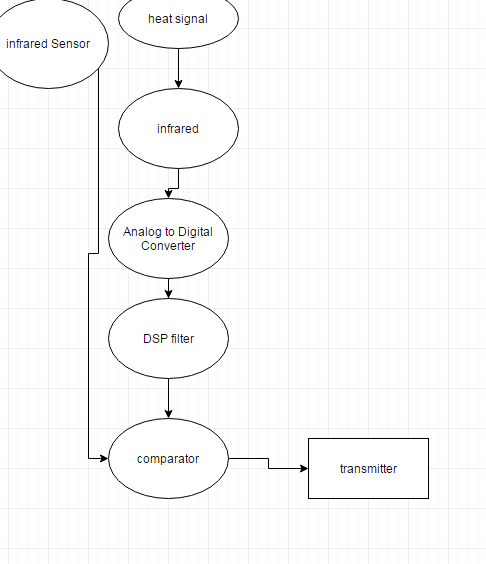


Figure 3.3.4a Thermal Method 1

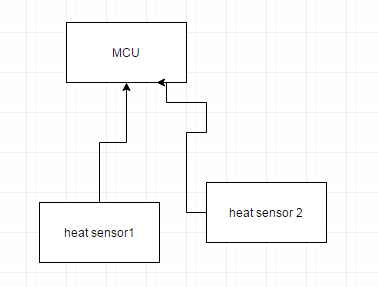
**Method 2**

**Thermistor**

Using a constant voltage source can use a voltage divider to determine the change in temperature compared to the voltage supplied by the temperature measured by a thermistor.

**Method 3**

The system simply has a sensor attach to the TX to give the temperature of the day over a set values measured over a specified time. This will allow the temperature to be adjusted even in the day. Than a second sensor is used to detect when an elevation is passed a specified threshold for the temperature.



3.3.4b Thermal Method 2

**Dangers**

In the guide it states that the product is dangerous to humans however we will disagree that the project that we are working on is dangerous mainly, because the wavelength that is most likely being using is most likely a radio wave which is nowhere near the amount needed to break chemical bonds . Now if the voltage being generated may be harmful. Yet, we took this in consideration and is why we made the design the way it is by using small voltages at different wavelength will help send out higher voltages in a safe way.

**Faraday Cage**

To help protect the electronics we will use what is known as a Faraday cage to protect from the dangers of any kind from the power surge that may happen in the wiring from the transmitter.

For the cage we will have a cage that is made out of a conductor, which will he used to send a radio frequency to interfere with the other radio frequency. However based on the loss of energy I even doubt this will be an issue but we need to take note if it.

Now to design a faraday cage we must consider the power of the Dbv that is travelling through the cage. With this in mind, wireless energy is going to be carry some heavy Dbv in the air. To fully protect the equipment, we will need to set up the faraday in laters to protect the equipment from the Dbv.

**Battery Receiver Interaction (bri)**

For the interface, we wanted to showcase how the battery is able to run longer than with the wireless energy than the battery has without the wireless energy. To make the system, we are going to have the wireless energy charge some batteries, this will act similar to a car’s alternator charging up the car’s batteries.

The advantages over using this system is that we can use more energy to power the robot gathering energy over the alternator. The advantage of the wireless energy charging the system is that it will last longer than a system using only batteries.

**3.4 Microcontroller**

A microcontroller is the simplest and most cost effective option to interface with the microprocessor, sensors, and motors of the robot. Its functions will be to use the data from the sensor to react to obstacles along the course and to send and receive sensor data with the microprocessor. The motors will be controlled by the data received from the microprocessor. The microcontrollers that are being considered are either used by the Arduino platform, which allows for many programming resources and libraries to be utilized, or do not have a steep learning curve. We are also familiar with the Arduino IDE and have experience with The MSP430 is used by UCF in its embedded systems course and will allow for professors and TAs to assist with the project.

**3.4.1 ATmega328**

|  |  |
| --- | --- |
| **Architecture** | 8 bits |
| **Clock Frequency** | 20 MHz |
| **Maximum Operating Voltage** | 5.5 V |
| **Program Memory** | 32 KB |
| **RAM** | 2 KB |
| **USART/SPI** | 1/1 |
| **I2C** | 1 |
| **I/O** | 23 |
| **Analog to Digital Converters** | 8 channels |

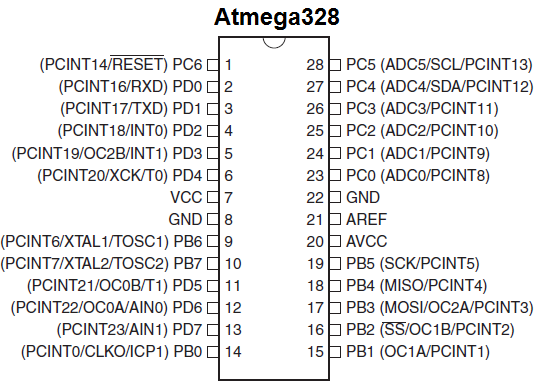


Figure 3.4.1 Atmega328

Reprinted with permission from Arduino.cc

**3.4.2 ATmega2560**

|  |  |
| --- | --- |
| **Architecture** | 8 bits |
| **Clock Frequency** | 16 MHz |
| **Maximum Operating Voltage** | 5.5 V |
| **Program Memory** | 256 KB |
| **RAM** | 8 KB |
| **USART/SPI** | 2/4 |
| **I2C** | 1 |
| **I/O** | 86 |
| **Analog to Digital Converters** | 16 channels |

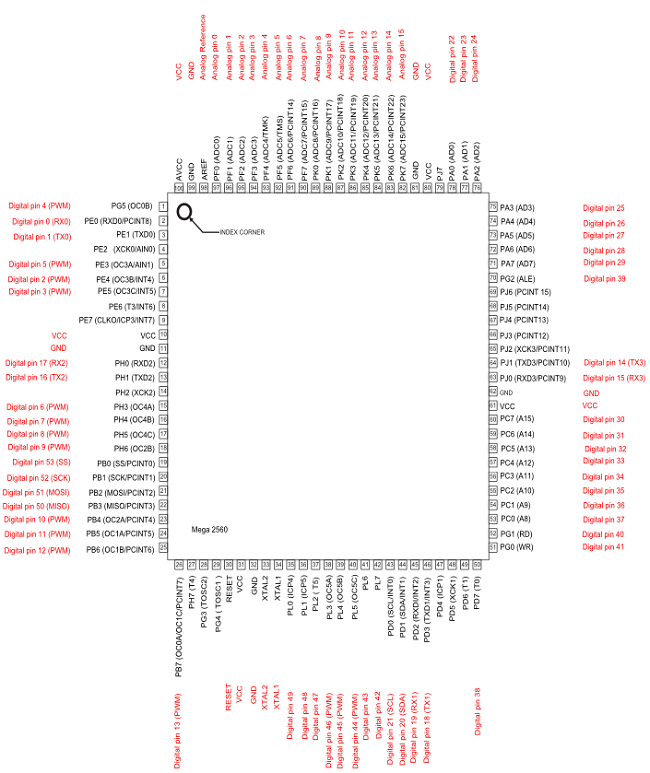


Figure 3.4.2 ATmega2560

Reprinted with permission from Arduino.cc

**3.4.3 MSP430G2553**

|  |  |
| --- | --- |
| **Architecture** | 16 bits |
| **Clock Frequency** | 16 MHz |
| **Maximum Operating Voltage** | 5.5 V |
| **Program Memory** | 16 KB |
| **RAM** | 0.5 KB |
| **UART/SPI** | 1/2 |
| **I2C** | 1 |
| **I/O** | 24 |
| **Analog to Digital Converters** | 8 channels |

**3.4.4 Atmel SAM3X8E**

|  |  |
| --- | --- |
| **Architecture** | 32 bits |
| **Clock Frequency** | 84 MHz |
| **Maximum Operating Voltage** | 3.3 V |
| **Program Memory** | 512 KB |
| **RAM** | 96 KB |
| **USART/SPI** | 4/1 |
| **I2C** | 2 |
| **I/O** | 54 |
| **Analog to Digital Converters** | 16 channels |
| **Digital to Analog Converters** | 2 channels |

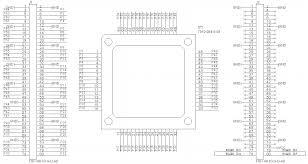
****

Figure 3.4.4 Atmel SAM3X8E

Reprinted with permission from Arduino.cc

**3.4.5 Atmel AT32UC3A**

|  |  |
| --- | --- |
| **Architecture** | 32 bits |
| **Clock Frequency** | 66 MHz |
| **Maximum Operating Voltage** | 3.3 V |
| **Program Memory** | Up to512 KB |
| **RAM** | 64 KB |
| **USART/SPI** | 4/1 |
| **I2C** | 2 |
| **I/O** | Up to 109 |
| **Analog to Digital Converters** | 8 channels |
| **Digital to Analog Converters** | 2 channels |

**3.5 Microprocessors**

The need for image processing and navigation for the autonomy of the robot may require the use of a microprocessor. A microprocessor will allow for fast image processing and can quickly communicate with the microcontroller. The microprocessor can also be used to send video and data to a nearby computer. The three microcontrollers below are developed by the Raspberry Pi Foundation and Texas Instruments.

**3.5.1 Raspberry Pi 2 Model B**

|  |  |
| --- | --- |
| **CPU** | Quad Core 32-bit ARM Cortex-A7 |
| **CPU Speed** | 900 MHz |
| **RAM** | 1 GB LPDDR2 |
| **Onboard Storage** | N/A |
| **Expandable Storage** | microSD |
| **GPIO/SPI** | 40 |
| **I2C** | 1 |
| **USB ports** | 4 |
| **Network Connection** | 10/100 Ethernet |
| **Operating System** | Various Linux/Debian, Android, BSD, RISC OS, webOS |
| **Power** | 5V, 2A |
| **Cost** | $35 |

**3.5.2 Raspberry Pi 3 Model B**

|  |  |
| --- | --- |
| **CPU** | Quad Core 64-bit ARM Cortex-A53 |
| **CPU Speed** | 1.2 GHz |
| **RAM** | 1 GB LPDDR2 |
| **Onboard Storage** | N/A |
| **Expandable Storage** | microSD |
| **GPIO/SPI** | 40 |
| **I2C** | 1 |
| **USB ports** | 4 |
| **Network Connection** | 10/100 Ethernet, 802.11n Wireless LAN, Bluetooth 4.1, Bluetooth Low Energy |
| **Operating System** | Various Linux/Debian, Android, BSD, RISC OS, webOS |
| **Power** | 5V, 2.5A |
| **Cost** | $35 |

**3.5.3 BeagleBone Black Rev B**

|  |  |
| --- | --- |
| **CPU** | ARM Cortex-A8 |
| **CPU Speed** | 1 GHz |
| **RAM** | 512MB DDR3 |
| **Onboard Storage** | 2GB Flash |
| **Expandable Storage** | microSD |
| **GPIO/SPI** | 69 |
| **I2C** | 1 |
| **USB ports** | 4+1 |
| **Network Connection** | 2 port 10/100 Ethernet |
| **Operating System** | Various Linux/Ubuntu, Debian, Gentoo, XNU, FreeBSO, Nintendo Symbian, QNX, Windows CE |
| **Power** | 5V, 3.3V |
| **Cost** | $45 |

**3.6 Autonomous Control**

**3.6.1 Image Processing**

**OpenCV**

OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. It is one of the most popular and widely used libraries in the computer vision industry. It uses more than 2500 optimized algorithms which can be used to “detect and recognize faces, identify objects, classify human actions in videos, track camera movements, track moving objects, extract 3D models of objects, produce 3D point clouds from stereo cameras, stitch images together to produce a high resolution image of an entire scene, find similar images from an image database, remove red eyes from images taken using flash, follow eye movements, recognize scenery and establish markers to overlay it with augmented reality, etc.”, according to the OpenCV website opencv.org.

OpenCV includes C++, C, Python, Java, and MATLAB interfaces and also supports Windows, Linux, and Android. This library is mainly used for real-time vision applications which makes it useful for companies such as Google, Yahoo, Microsoft, and Intel.

**SimpleCV**

SimpleCV is an open source framework, which means it builds off of a collections of OpenCV and other computer vision libraries. According to simplecv.org “With it, you get access to several high-powered computer vision libraries such as OpenCV – without having to first learn about bit depths, file formats, color spaces, buffer management, eigenvalues, or matrix versus bitmap storage.” SimpleCV exclusively uses Python which makes it more limited, however, it is very useful for prototyping.

**3.6.2 Object Detection**

**SLAM**

Simultaneous location and mapping is used to create a map of an environment while also simultaneously keeping track of the robot’s location.

**3.6.3 Algorithms**

**Hough Line Transform**

Hough Transform is a method to detect any shape which can be represented in mathematical form like a line or a circle. The algorithm is implemented using an array with a length equal to the number of unknown parameters. For implementation polar coordinates will be used where points are represented by parameters p and θ, where p is the perpendicular distance from the origin to the line and the angle θ is created by the perpendicular line and horizontal axis.

For point (x, y), a set of lines that go through this point can be determined using the formula: p = xcos(θ) + xsin(θ)

This results in a set of pairs (p, θ) that represent all lines that go through points (x, y). The result is a sinusoid wave when a group of lines is plotted that go through (x, y). The procedure is repeated for all points in the frame. If an intersection is detected between the curves of 2 different points in the plane that indicates that these 2 points belong to the same line. With this method, Hough Transform can effectively detect all intersection between points. For more precision, a threshold can be set to indicate a minimum number of intersections needed to detect a line in an image frame. But for the purpose of the project, the algorithm will be limited to detect only the lines on the road based on its color, and ignore any additional detected lines.

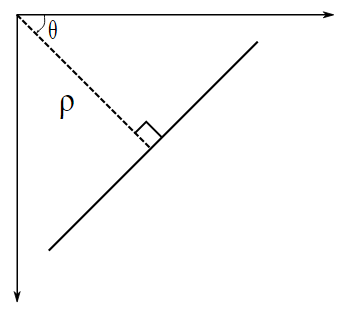


Figure 3.6.3 Hough Line Transform

Reprinted with from opencv.org

**Hough Circle Transform**

Unlike line detection, circle detection relies of 3 parameters: x, y, p, representing the center of the circle and the radius respectively. Using Canny edge detector operator, an image is pre-processed to detect all edges that are present in the frame. For every point that lies on the edge, a circle is plotted around that point, making it the center with radius r. An array is used as an accumulator to increment the radius until a desired value equal to r0 is achieved. Another accumulator counts the number of circles that pass through each edge point. At the end, the highest count is selected. Center coordinates of circles with the highest count in the image frame are presented as detected circles.

Explain advantages and disadvantages

**3.7 Sensors**

In order to navigate along the track, the robot will need a sensor. We will compare 3 different sensors to determine which one is the best fit for the project which will facilitate a fluid navigation. The criteria will be power consumption, range, weight, price, and data accuracy.

**3.7.1 Cameras**

The camera sensor

We are also considering using a camera sensor to detect and track an object or detecting of the numerous objects that can be presented in the track. Here are certain advantages that come from a camera sensor:

* Tracking
* Detecting colored objects
* Specific shapes detection
* Capable of detecting triangle corners from an image
* Spatial awareness on objects on 2D surfaces
* It can select which target to focus on

**Cmucam5 pixy**

A good prospect for a camera sensor would be the cmucam5 pixy. It is the latest version of CMUcam and it is labeled as the most popular vision system in history even though it was released 2 years ago.

Here are the specs for the cmucam5 pixy

|  |  |
| --- | --- |
| Processor | NXP LPC4330, 204 MHz, dual core |
| Image sensor | Omnivision OV9715, 1/4", 1280x800 |
| Lens field-of-view | 75 degrees horizontal, 47 degrees vertical |
| Lens type | standard M12 |
| Power consumption | 140 mA typical |
| Power input | USB input (5V) or unregulated input (6V to 10V) |
| RAM | 264K bytes |
| Flash: | 1M bytes |
| Available data outputs | UART serial, SPI, I2C, USB, digital, analog |
| Dimensions | 2.1" x 2.0" x 1.4 |
| Weight | 27 grams |

**3.7.2 Infrared**

The Infrared sensor or IR sensor works by sending a light signal which will be detected with the parabolic mirrors or Fresnel lenses it is equipped with. To detect an object, after the light signal is sent it will be reflected back to the sensor; the distance of the object will be determined by the intensity of the wavelength of the reflected light signal.

For the Ir sensor, we are considering the Sharp GP2Y0A710K0F IR Range Sensor

Here are the specifications

|  |  |
| --- | --- |
| Supply Voltage | -0.3V to +7 V |
| Measuring distance range | Min 100 to Max 550 cm |
| Operating supply voltage | 4.5 to 5.5 V |
| Operating temperature | -10 to +60 C |
| Large format | 58.0×17.6×22.5 mm |
| Average supply current | Min-0, Typ-30, Max-50 |
| Storage Temperature | -40C to + 70 C |
| Output terminal voltage | -0.3 to Vcc + 0.3 |

The next IR sensor we will be looking at is the GP2Y0A60SZ distance sensor

|  |  |
| --- | --- |
| Operating voltage | 5V version: 2.7 V to 5.5 V  3V version: 2.7 V to 3.6 V |
| Average current consumption | 33 mA |
| Distance measuring range | 10 cm to 150 cm |
| Output type | analog voltage |
| Output voltage differential over distance range | 5V version: 3.0 V  3V version: 1.6 V |
| Update period | 16.5 ± 4 ms |
| Size without header pins | 33 mm × 10.4 mm × 10.2 mm |
| Weight without header pins | 2.5 g (0.09 oz) |

**3.7.3 Sonar**

The most used sensor for a small autonomous robot is the sonar sensor. It works by sending sound waves on its surroundings which will be reflected back when they come in contact with a certain object. As a result, the reflected waves will get pick up by a receiver in the sensor the will determine the distance of the object with which the sound waves reached.

One of the sonar sensors being considered is the Maxbotix

HR-LV-MaxSonar-EZ0 High. It is said that it has the widest and most sensitive beam pattern of any unit from HRLV-MaxSonar-EZ sensor line. Here are the specifications for this sensor:

|  |  |
| --- | --- |
| reading rate | 10Hz |
| Operating voltage | 2.5V - 5.5V |
| Ultrasonic sensor | 42kHz |
| sensor output | Analog Voltage, RS232 or TTL Serial, and Pulse Width |
| Maximum range | 5000 mm |
| average current requirement | 3.1mA |
| Operating Temperature | -15ºC to +65ºC |

Other important specifications include:

* lower cost without sacrificing performance
* Higher tolerance for outside noise
* Simple integration into project
* Internal temperature compensation
* Real time automatic calibration
* Great for detecting people

Another good prospect for sonar sensors for this project is the Maxbotix LV-MaxSonar-EZ1 Sonar Range Finder MB1010. The selling point for this sonar sensor is that there is no dead zone and it is capable of detecting very small materials that are very close to the sensor face. Here are the specifications for this sensor:

|  |  |
| --- | --- |
| Size | 0.870" x 0.786" x 0.645" |
| Resolution | 2.5 cm |
| typical current consumption | 2 mA |
| Baud | 9600 |
| ultrasonic sensor | 42 kHz |
| reading rate | 20 Hz |
| Three interfaces | Serial output  Analog output  Pulse width output |
| Maximum range | 6.45 m |
| Minimum range: | 0 m |
| Weight | 4.3 g |

**HCSR04 Ultrasonic sensor**

The HCSR04 needs a 5 V power supply and is trigger by a pulsed input , and produce a pulse output.

|  |  |
| --- | --- |
| Working voltage | DC 5V |
| Working current | 15mA |
| Working frequency | 40Hz |
| Max range | 4m |
| Min range | 2cm |
| Measuring angle | 15 degree |
| Trigger input signal | 10uS TTL pulse |
| Echo output signal | Input TTl Lever signal |
| dimension | 45\*20\*15mm |

**3.8 Robot Platform**

For this project, one of the most important attributes we are looking for in the robot is space. The robot platform should have enough space to include the motor, sensor, battery and anything we might need to add. We settle on two robot platform which were great for do-it-yourself projects and they are both compatible with arduino.

CanaKit 4 Wheel Drive

The CanaKit 4 Wheel Drive is one of the robot chassis can can satisfy our requirements. This chassis is being illustrated in figure 3.8.1 below.

**Specifications**

* 4WD Arduino mobile robot development platform
* 4 high-quality micro-speed motor
* Tough aluminium body with mounting holes
* Variety of sensors can be added

Although this chassis allows some flexibility within its specifications, a problem still arises. Our project will require a decent amount of power and enough space for all the different materials that are needed to be present to ensure a smooth operation on the different task the robot may have to perform.

Arduino Compatible DIY Motor Smart Robot Car

The other robot chassis that is being considered is the Arduino Compatible DIY Motor Smart Robot Car. This chassis is being illustrated in figure 3.8.2 below.

**Specifications**

* Gear motor deceleration indicator
* With installation holes, easy to use
* Great for DIY project
* Compatible with Arduino
* Product weight: 0.479KG

This chassis seems a bit more convenient for our project than the CanaKit 4 Wheel Drive because it allows more space for other materials to be mounted onto it. Furthermore, the position of the batteries should contribute to a better connection with the wireless energy materials.

**3.9 Motor and Drivers**

Another important part that is detriment to the success of this project is the motor. Servo motors are the electric devices that are responsible for the movement of a chassis. But, they are also used in numerous objects from electronics and electronic appliances to cars and airplanes. To cause a robot to move, the servo motor will rotate or push parts of the chassis with great precision. Some of the important specifications in a servo motor that we look for this project weight, power consumption and reliability.

A servo motor uses a small dc motor which gets power from a battery. The efficiency of a servo motor depends on spinning at high revolutions per minutewhile using very low torque. When motor is going really fast the gears are responsible for its deceleration without an increase in torque. As a result, the type of gear that the motor will be using should be really effective for this project.

**HS-422 Servo Motor**

**Features**

* Voltage: 4.8-6.0 Volts
* Weight (g/oz): 45.5/1.6
* Torque: 45.82/56.93 oz-in. (4.8/6.0V)
* Speed: 0.21/0.16 sec/60° (4.8/6.0V)
* Direction: Clockwise/ Pulse Traveling 1500-1900usec
* Rotation: 180°
* Dual Oilite Bushing
* Nylon Gears
* 3-Pole Ferrite Motor
* C1 Standard Spline

**SainSmart Nano Servo motor: Tower pro SG90**

**Specifications**

* Weight: 9 g
* Dimension: 22.2 x 11.8 x 31 mm approx.
* Stall torque: 1.8 kgf·cm
* Operating speed: 0.1 s/60 degree
* Operating voltage: 4.8 V (~5V)
* Dead band width: 10 µs
* Temperature range: 0 ºC – 55 ºC
* Gear type: plastic
* Pulse Width: 500-2400 µs
* Rotation: 180°

**Motor Drive**r

The motor driver is an extension component that allows efficient control of the DC motors. We are considering the L298N.

Here are the specifications for it:

|  |  |
| --- | --- |
| Operating Voltage | 5V to 12V |
| Motor controller | L298N, Drives 2 DC motors or 1 stepper motor |
| Max current | 2A per channel or 4A max (with external power supply) |
| Current sensing | 1.65V/A |

Most microcontroller is not equipped to control multiple DC motors in a chassis because of its restrictions concerning output and input current. To solve this problem, we need an integrated circuit that is capable of allowing the DC motors being used to work independently to facility steering a chassis. It would require an H-bridge circuit to accomplish that feat. This is not going to be an issue when it comes to the motor shield.

The the L298P is an integrated monolithic circuit in a 15-lead Multiwatt and PowerSO20 packages. It is a full bridge driver capable of sustaining high voltages, high currents. It is also capable of allowing standard TTL logic (transistor-transistor logic) levels and drive inductive loads such as DC motors and stepping motors.

|  |  |
| --- | --- |
| Operating supply voltage | Up to 46 V |
| Total DC current | Up to 4 V |
| Saturation voltage | Low |
| Protection | Overtemperature protection |
| Logic level | Logical "0" |
| Input voltage | Up to 1.5 V |

Furthermore, the motor driver enables two inputs. These inputs are responsible to enable or disable the device regardless of the input signals that are being sent; this can allow it to control the wheels of a chassis independently. A combination of DC motors and a motor shield would be more beneficial to the success of the project.

**3.10 Batteries**

The projects need reliable rechargeable batteries in order to complete the wireless energy transfer. We have to figure out which of those batteries is the best fit for this particular project. Three types of batteries will be compared by presenting some of their characteristic and highlighting some of their advantages and disadvantages.

**3.10.1 Nickel-Cadmium**

Characteristics

The nickel-cadmium is composed of nickel hydroxide Ni(OH)2 for the cathode and, cadmium Cd as the anode and it is also equipped with the alkaline potassium hydroxide KOH electrolyte.

Advantages

* Low internal resistance
* High rate charge and discharge
* Tolerates deep discharges - can be deep cycled.
* Wide temperature range
* Typically last over 500 cycles.
* Cools during charging
* Fast charging
* Flat discharge characteristic
* Can be stored in different states without damage
* Available in a variety of sizes

Disadvantages

* Susceptible to limiting the amount of charge due to previous discharges
* Prone to damage caused by overcharging
* For better result with the type of battery, it should be fully charged but most importantly, it discharged fully once a month to avoid damages.

**3.10.2 Nickel-Metal-Hydride**

Characteristics

Nickel-metal-hydride batteries are similar to nickel-cadmium batteries but uses hydrogen instead of cadmium. In this case, Hydrogen is used in the anode.

Advantages

* High energy density
* Low internal impedance
* 3000 typical cycle life
* Can be deep cycled
* Capable to resist overcharge and overdischarge
* Flat discharge characteristic
* Can operate in many temperatures
* Can be fully charged within 1 hour
* Reconditioning
* Safer than most batteries

Disadvantages

* High self-discharge rate.
* Can be stored indefinitely either fully charged or fully discharged.
* High rate discharge not as good as NiCads
* Less tolerant of overcharging than NiCads
* Cell voltage is only 1.2 Volts
* Lower capacity

**3.10.3 Lithium-ion**

Lithium-ion batteries were designed to prevent the safety issues that can be created by a lithium metal. The essential feature of the Lithium ion battery is that at no stage in the charge-discharge cycle should there be any Lithium metal present. Rather, Lithium ions are intercalated into the positive electrode in the discharged state and into the negative electrode in the charged state and move from one to the other across the electrolyte.

Advantages

* Lithium-ion cells have no memory effect
* Long cycle life and
* Excellent discharge performance.
* Capable of operating under very high current
* Ideal for use in high power applications

Disadvantages

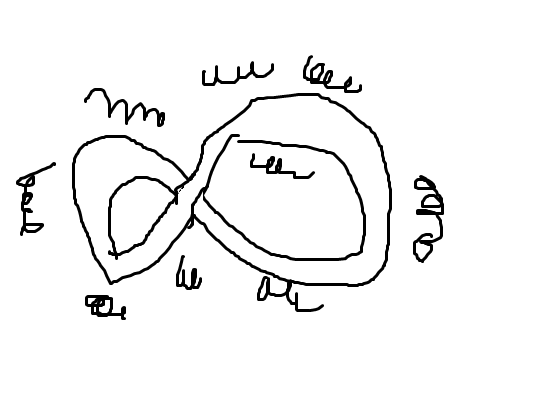
* Certain cell can be hazardous
* Weight more
* Limited in size

**3.11 Track**

The track will be in a figure 8 such that the robot will be traveling around. To help the energy transfer the coils need to be placed such that the robot has accessed to them.

As seen below

**3.11.1 Method 1**



**Advantages**

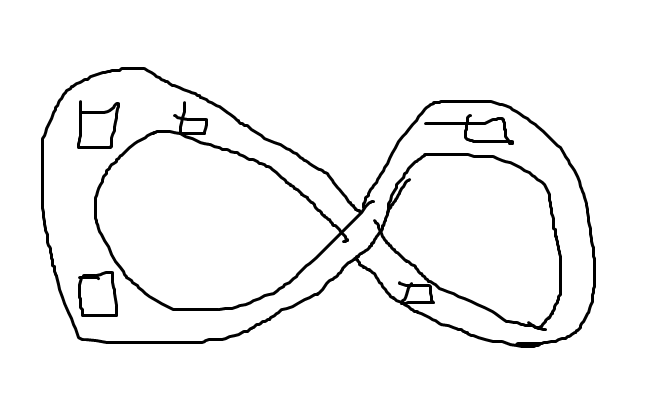
The main advantage to this is that the coils will always be charging up the battery.

**Disadvantages**

There will be energy theft and coupling will be a lot harder

**3.11.2 Method 2**

**Mats**



The mat will be placed at specified stops so that the robot will charge while waiting to pick up a passenger. These meets will use the same methodology as the system described to have multiple receivers. Not only that, but

**Receiver**

The receiver will still be the same on the robot.

**Advantages**

There is no energy theft from the track rather there is.

**Disadvantages**

The robot has to stop over the mat such that they have an inductance couple.

**3.11.3 Method 3**

For this method we have two big coils underneath the track such that they are at a set distance from the receiver. The distance that is being used is the radius of the circle.

**Advantages**

Having a set distance from the receiver will most likely be able to account for the inverse square law which will make our design much easier to work with.

**Disadvantages**

Some disadvantages is that the inductance is limited and can’t be rearranged.

**4 Standards**

**4.1 Standards Search**

ISM band

There are certifications that is required to use a range of frequencies. To avoid the hassle of getting a certification or if we are unable to get a certificate, we will need to use an ISM band frequency.

The ISM band is a range of frequencies that are allowed to be use without the use of a license. Within the ISM band there are a select few ranges that correspond to wireless signal communication, these band is what is going to be used in the design for the robot.

**4.2 Design Impact**

The impact the ISM band will have on the design is the resonant frequency of the couple of the wireless energy. This will affect the design by changing the resonant frequency. By changing this frequency, will have an effect on the bandwidth, as well as the component values for inductors, resistor, etc..

**5 Hardware design**

**5.1 Microcontrollers**

From the research it was found that the Atmel SAM3X8E would be the best processor for the project. The clock speed is more than four times more than the ATmega328P, while also having the benefits of the other microcontrollers I/O availability and communication. The Atmel SAM3X8E also has 512 KB of Flash memory for storing code and 96 KB of RAM. This microcontroller provides all of the benefits of the other three discussed and it operates at a lower voltage. Since the image processing is handled by the camera unit, the microprocessor only needs to interpret this data, and the data from the sensors, as quickly as possible to control the motors. This makes the Atmel SAM3X8E the best choice because of its high clock speed.

The microcontroller design will be made on a PCB and should emulate the Atmel SAM3X8E which is used in the Arduino Due.  Since the Arduino Due is an open-source board, we can reference the Eagle schematics for our design. The schematic is shown in Figure 5.1.1.

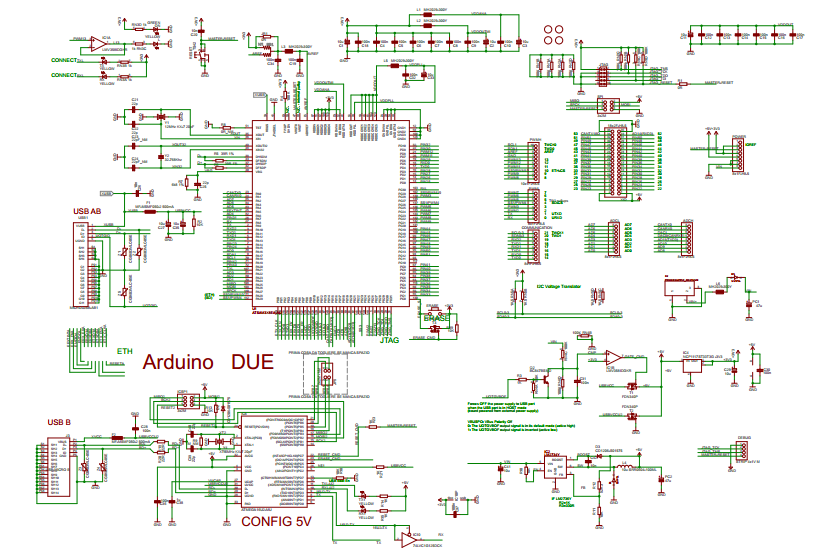


Fig 5.1.1 Arduino Due Schematic

Reprinted with permission from Arduino.cc

After we follow the design instructions, we will need to verify that these technical specs are reached with our design:

|  |  |
| --- | --- |
| Operating Voltage | 3.3V |
| Recommended Input Voltage | 7-12V |
| Limit for input Voltage | 6-16V |
| Digital I/O Pins | 54 |
| PWM outputs | 12 (Digital I/O Pins) |
| Analog Output Pins | 2 |
| Total DC Output Current on all I/O lines | 130 mA |
| DC Current for 3.3V Pin | 800 mA |
| DC Current for 5V Pin | 800 mA |

**5.2 Microprocessors**

When first discussing this project it was thought that a microprocessor such as the BeagleBone Black would be the best solution. However, the BeagleBone Black would provide an unnecessary amount of computational ability. Since we are only creating an autonomous vehicle which will navigate around a track and sense objects stopped in its path, everything we need can be achieved by using microcontrollers.

**5.3 Receiver**

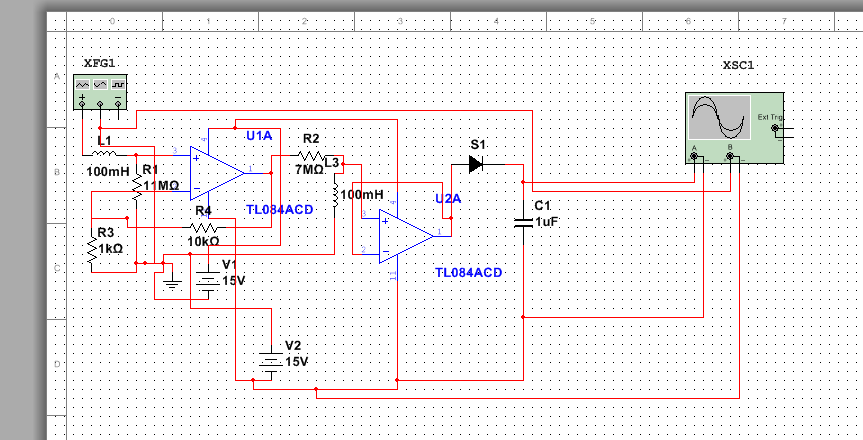
**Current to voltage converter**

To send out a signal with the transmitter with as much as accuracy as possible we need to convert all the dc voltage into current. Doing so, will make the circuit transmit the signal much smoother.

Eddy currents are being introduced into receiving coil, so in order to get the bandpass filter to be able to do its job it needs to transfer all the current into voltage such that it will be able to power the robot at the power we need it to be.

**Receiving Circuit**

Below is the circuit that will fit best for our design purposes.



The receiver will use only one Receiving coil to receive the power being delivered from the robot. The reason for this decision is that the inverse square law will not consume as much energy compared to using multiple receivers. Also since the voltage is small enough the magnetic field will likely be weak enough to not harm other electronics in the facility.

**Frequency**

The Resonant frequency that is desirable to send power over to the robot is the 13560000. The reason for this choice is

1.) The bandpass is smaller so that means less interference from other electromagnetic waves

2.) The length of the antenna will be smaller than using any of the other big frequencies. C = lambda\*13605000. This will

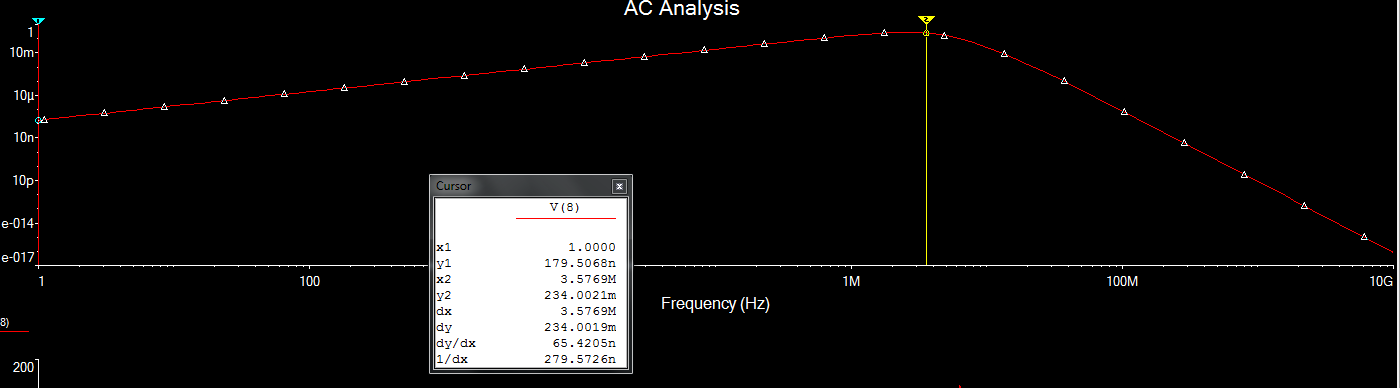
**Protection**

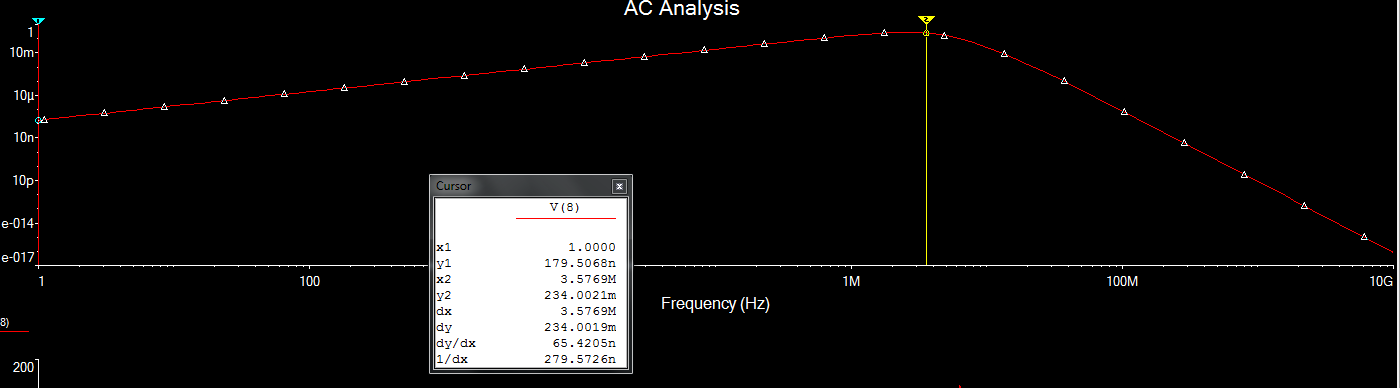
For protection, the robot will have a faraday cage surrounding all the electronics within the robot all except for the receiving coil. This will prevent any damage that could result in the robot’s electronics. While the receiver coil will not be protected because it will take the eddy current and transfer the current into power that the battery can use to charge.

**Bandpass**

For the bandpass filter, RL filter is what will be used. The reason being for this is because 1, it has amplification properties for the voltage such that the voltage that is loss can be regain to the circuit.

For the quality factor, we think the critical damping is the best, but since get it close to ½ is almost impossible having an underdamped circuit will work just as well. However, avoiding an overdamped circuit will be the most efficient method mainly because of the oscillation of the rate. This will make the design unpredictable when charging the battery and could potentially affect the robot.





**Precaution**

As a precaution, if high voltage is needed to be sent over the robot such that it could cause problems a second (possibly more) receiver and transmitter will be sent up. This will divide the voltage safely, and get the correct voltage across without frying any other electronics. One main concern is that if more than one couple is used an internal resistance of the receivers will most likely conflict with each other. In order to prevent this the circuit will need to be isolated such that the resistors will not interfere with any of the resonant frequencies of the receiver. To isolate the circuit a voltage follower will need to be applied before the peak detector circuit. This can be seen in the circuit above however if one receiver is enough it is possible to take out the voltage follower, and use only one op-amp for the circuit.

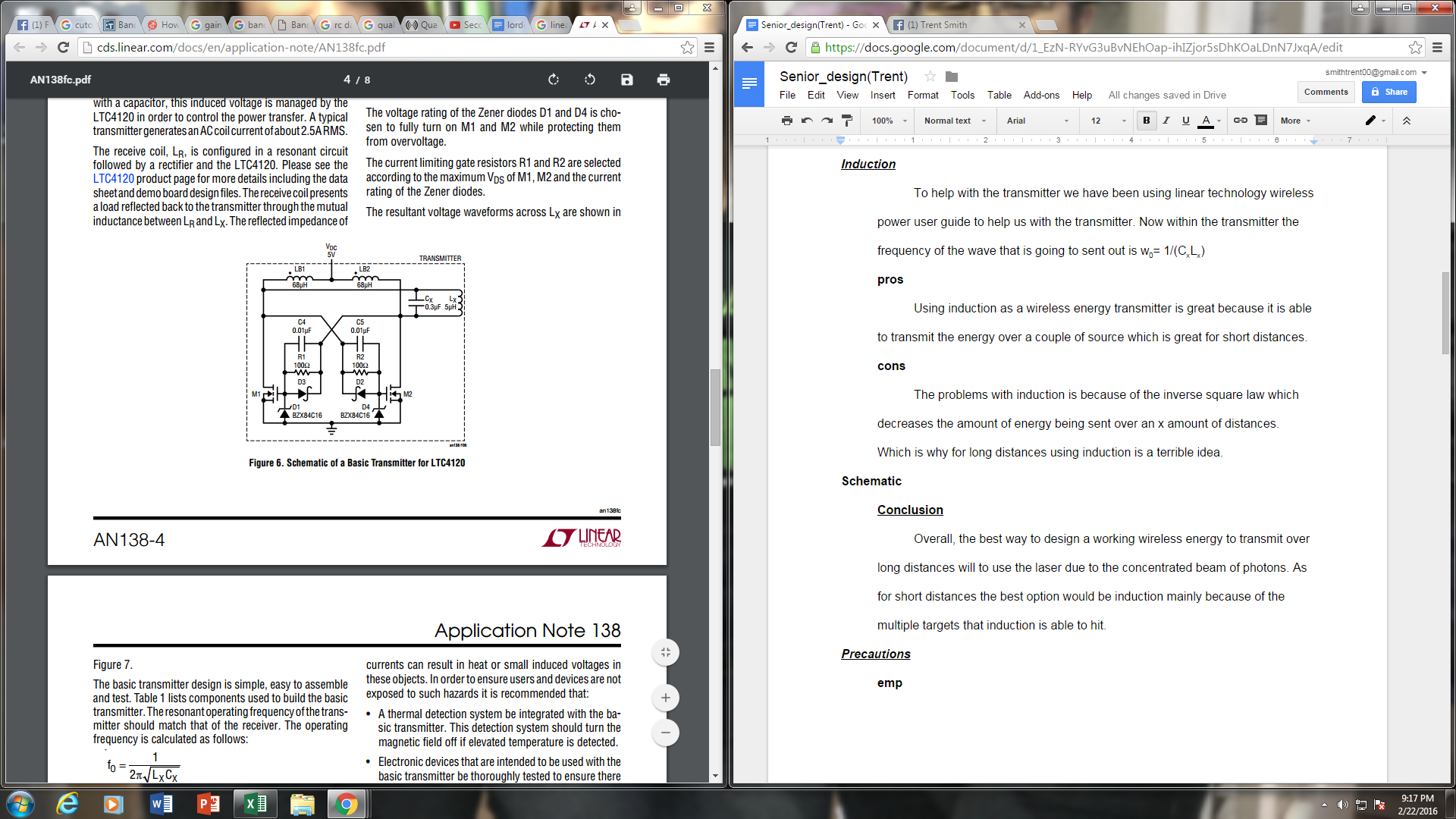
If more than one couple is needed one consideration is the method that was described above in the research section with the binary type system.

**5.4 Transmitter**

**Oscillators**

The circuit in the diagram takes a dc voltage and transfers it into an AC signal making it able to be sent out through a electromagnetic wave.

**Schematic**



**Note:** This schematic is from the wireless power user guide and is no way ours. Please note that some of the values being use will most likely change

**5.5 Transmitting and Receiving Coils**

Through various research and understanding different inductor coil types we feel that air core inductors will suit out designs much better. This is because air core inductors do not depend on ferromagnetic material to obtain a certain desired inductance. Ferromagnetic coils have a low electric conductivity compare to copper wires. This can impact the wire magnetic properties on the conductor through energy transfer. When AC is applied to the coil, it can affect the wires impedance and reduce the ferromagnetic material known as the skin effect. With higher frequency increase the inductance in the ferromagnetic wire has a loss of about ten percent, whereas there would be no loss in the copper wire. Since the production of harmonics can affect different electronics in our system, using the air core coils will prevent the production of harmonics. Air core coils are design to operate at high frequencies around 500MHz to 1GHz, where ferromagnetic ones tend to have great loss about 100MHz. This will result in a better Q-factor, handling of the power with less distortion, and increase in efficiency. We also have to maintain magnetic resonance between the transmitter coil and receiver coil, there will be fluctuation in our inductance because the current capacity would be affected. A common problem with ferromagnetic coils, is that the coils have iron losses with can affect our system and give our system a higher chance of unwanted behavior.

There are disadvantages to the air core coils that we will have to consider. For instance, without a permeability core you will have to have more and larger turns to achieve the desired inductance value. More turns mean lower self-resonance and higher copper loss. But since we will be using a high frequency voltage signal we will not need a high inductance. So the disadvantages they will not affect our system more than the benefits it will provide in our design.

The air core coil has an external radiated field at our resonance frequency that will be strong enough to be picked up by the receiving coil. We plan on using around 7 turns per coil which would provide a strong enough magnetic field with respect to the distance between transmitter and receiver coils. Figure # is an example of what the two coils will looked like**:**

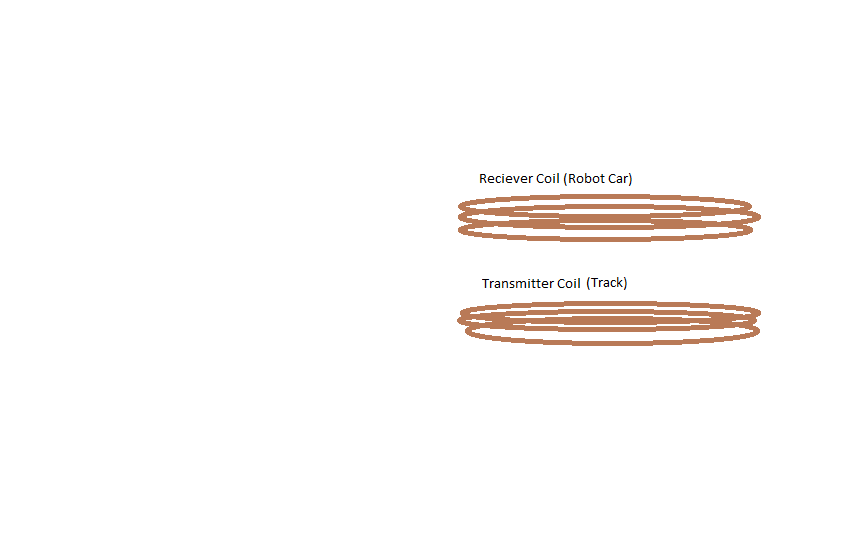


Figure 5.5a – Transmitter and receiver coils

Another design that will be greatly considered due to possible ground clearance and spacing between the track and the robot car is a flat coil design. The receiver coil will most likely be smaller than the transmitter coil because it will compensate for misalignment and optimize the transfer in energy so there will be less loss going into the car robot battery.

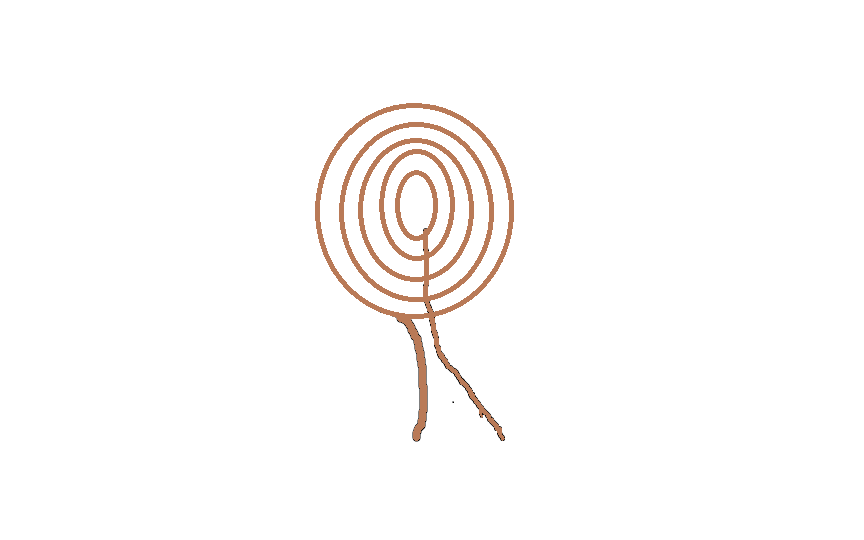
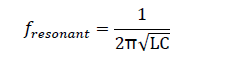


Figure 5.5b – Basic formation of transmitter coil (receiver coil is identical)

The resonance condition needed in our design will be accomplished my using a tank circuit design. This design is when the capacitor and inductor are in parallel to get the reactance of both components to be equal to each other. Because the inductor’s reactance increases with frequency and capacitor reactance decreases, when the components are in parallel there will be one frequency that will allow the reactance to be equal. Using the formula below, we will be able to obtain the resonance frequency desired for our design. Since the capacitor and inductor are in a relationship to obtain the resonant frequency, we will need to wind our inductor coils first and attach it to the circuit to find out our necessary capacitance to get our resonant frequency.



We may not be able to acquire the exact value of the capacitance needed for the resonant condition. In this case we will have to tweak our oscillator frequency or adjust the number of turns of the coil. The same condition for resonance must be adhered to in the tank circuit on the receiver side. The coil inductance and capacitor values must be such that it matches the resonant frequency of the transmitter coil.

**5.6 Coils**

The ground coil that’s connected to the track and the robot car coil will be used to transfer the energy over a given space. These coils will be wounded into the desired form that fits our constraints of the design. We will be using copper coils for our design. We can choose from various copper wiring such as, copper tubing, solid copper wire, litz wire. Aluminum wires would also be capable of producing the magnetic field necessary to transfer the energy between transmitter coil and receiver coil. We estimated using about 10 turns of coil material to create our designs and create the magnetic field over a 5 inch radius. So we looking at about 10 feet of conductor coils, give or take the size of the coil material we decide toy use.

10 AWG Solid Copper Conductor

Vendor: Home Depot

Manufacturer: CerroWire

Quantity: 25 ft.

Price/Unit $14.57

http://www.homedepot.com/b/Electrical-Wire/Bare-Copper/N-5yc1vZbm7vZ1z0uqq4

Table 5.6.1

The solid copper conductor is a very commonly used coil in our everyday electronics. It’s very easy to shape, has excellent conductivity, easily terminated. Our group is leaning towards We are most likely going to use solid copper.

Copper Tubing

Vendor: Lowes

Manufacturer: Cambridge-Lee

Price/Unit: $14.95

http://www.lowes.com/Plumbing/Pipe-Fittings/Copper-Pipe-Fittings/\_/N-1z10xv9/pl#!

Table 5.6.2

Copper tubing is commonly used in gas piping’s or HVAC’s. It has great conductivity, easily terminated, and very shapeable this material is considered but could not meet our spacing constraints in out design.

Litz Wire

Vendor: eBay

Quantity: 38 ft.

Price/Unit: $8.00

Table 5.6.3

http://www.ebay.com/itm/like/172068374435?lpid=82&chn=ps&ul\_noapp=true

Litz wire is a type of cable used to carry alternating current in electronics. This wire was designed to reduce the skin effect during operation. It also have a proximity effect losses up to 1MHz. The efficiency of this coil would be a great for out designs especially using low frequencies. The material is a bit expensive but it does not outweigh the benefit. The material is small strands already twisted and woven together. It has excellent conductivity, flexible and is easy to shape.

Aluminum

Vendor: Home Depot

Manufacturer: OOK

Quanitity: 50ft.

Price/Unit: $3.48

Table 5.6.4

http://www.homedepot.com/p/OOK-50-ft-Aluminum-Hobby-Wire-50176/100192917

Aluminum is also another option of material that can be used for our coil. Aluminum is also used in electronics. However coils wound with aluminum have more losses in performance compared to copper coils. Aluminum wire requires 1.6 times as much cross section as copper to carry an equivalent amount of current. Aluminum wound coils are typically larger than copper coils.

**5.7 Coil Enclosure**

After the coil designs test successfully, we will have to mount the transmitter coils on to the track and mount the receiver coils on to the car robot. We will screw the receiver coil to the track there for it stays in place so there is no misalignment during the energy transfer. We will also screw the receiver coil onto the robot car so that the coil will stay in place while the robot car travels over a distance.

Since the coils are the main component to complete the wireless energy transfer, we want the coils to protected and unseen for practical purposes. We will design a box to cover the transmitter circuit and coils. One side will be attached the plexi board (track) and the other sides will be boxed in with nonconductive material. The receiver coils and circuit will be boxed in as well, ensuring enough clearance between the track and the robot. Example illustration is shown below:



Figure 5.7 Coil Enclosures

**5.8 Oscillator Design**

The oscillator is what generates the resonant frequency to oscillate the transmitter coil. When resistance is taking into account that limits the oscillations required because of the resistor or the resistance in the inductor. A design we are considering is a design we have shown in Figure 2.1 taken from wireless power transmitter demonstration video. This circuit has a transmitter that oscillates at 63 KHz and power 12 volt loads from a distance of 10 centimeters, which is well over the clearance required for out track to robot design. It’s essential for the receiver on the car robot to be alignment for optimal performance.

R1 = R2 = 460Ω, R3 = R4 = 10KΩ

L1 = 1000µH, C1 = 0.47µF

The transmitter coil is center tapped and consists of 9+9 turns

This circuit has two IRFN054 power MOSFETs to take the power drawn from the transmitter coil. A heat sink will be mounted on to the system at cool temperatures. Heat will start to generate and will affect the circuit or other components. The 1N4740A Zener diodes are used to maintain constant amplitude and the BYV26E ultra-fast avalanche sinterglass diodes are able to handle high operating current and fast switching in the circuit. The receiver consist of 18 turns in parallel with a capacitor that has the same resonant frequency of the transmitter coil. For our design the capacitors are subject to change due to the inductance of our inductor.

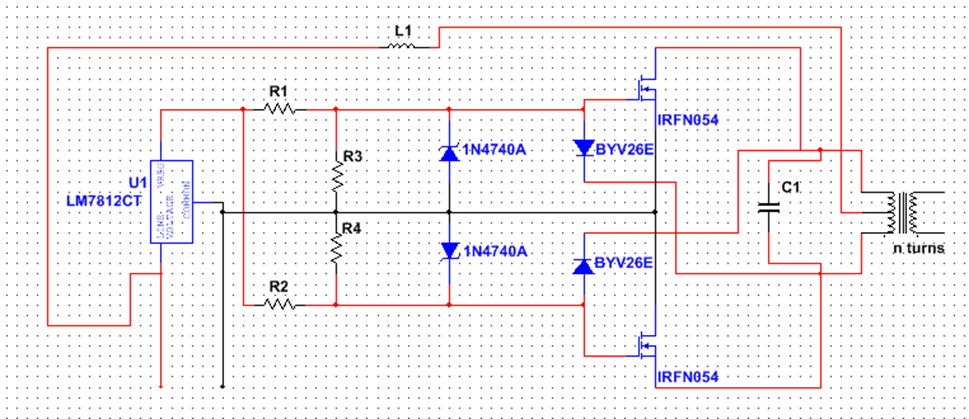


Figure 5.8a – Wireless energy transfer oscillator circuit diagram

Another oscillator design that is considered is the Colpitts oscillator show in figure “”. This circuit uses a single stage inverting amplifier and a LC phase shift network. The two capacitors in parallel (C1 & C2) are used to provide the feedback voltage. The voltage across C2 provides the regenerative feedback required to sustain oscillations. The parallel combination Re and Ce provides self-biasing with the R1 and R2 resistors. The supply voltage VCC is applied to the collector through a radio-frequency choke (RFC) which allows an easy flow of direct current but at the same time it offers very high impedance to the high frequency currents. The coupling capacitor Cc in the output the circuit does not allow the dc currents to go to the tank circuit. For this circuit a high Q-factor is desired but it’s reduced by the DC current in a tank. Through the capacitor Cc a radio-frequency energy is made across a RFC, which is capacitive coupled to the tank circuit.

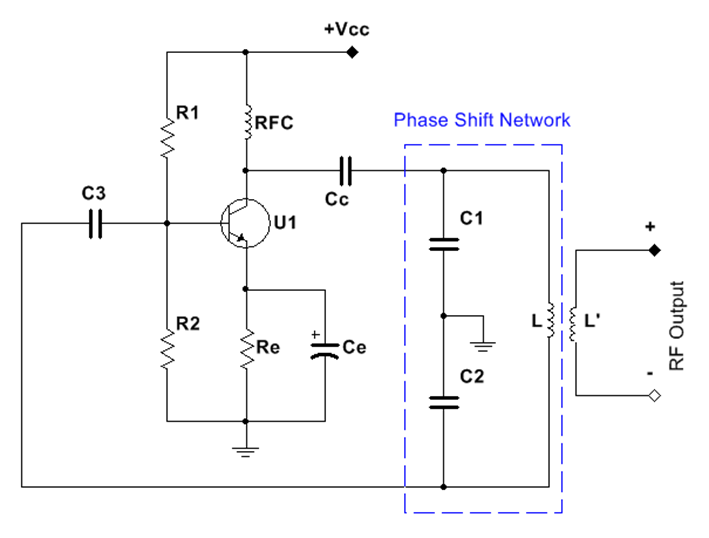
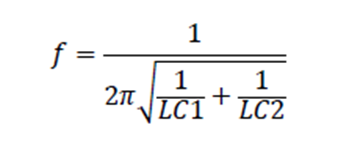


Figure 5.8b–Colpitts Oscillator

The coupling of L and C2 makes LC phase shift network, which is connected to the amplifier input. The input is coupled by the coupling capacitor CC to block the DC current going in and allow the AC current to pass through to the junction of L. There is two phase shifts of 180° produced by the transistor and by the capacitive feedback. The phase shift of a total of 360° is what’s needed to develop the conditions of oscillation. The voltage output comes from the second winding L coupled to the inductance L. To obtain the required frequency the capacitors C1 and C2 are tuned to simultaneously which is referred to as gang-turning. This is the equation to find the frequency of oscillations:



Another oscillator design we are considering is the Hartley oscillator showing in figure “”. This oscillator is another highly recommend design just as the Colpitts oscillator. This design is widely used in local oscillators in radio receivers. The Hartley Oscillator circuit is very similar to the Colpitts oscillator circuit but there is a difference in the configuration of the phase shift network. The Hartley contains two inductors L1 and L2 and a capacitor while the Colpitts has two capacitors and one inductor.

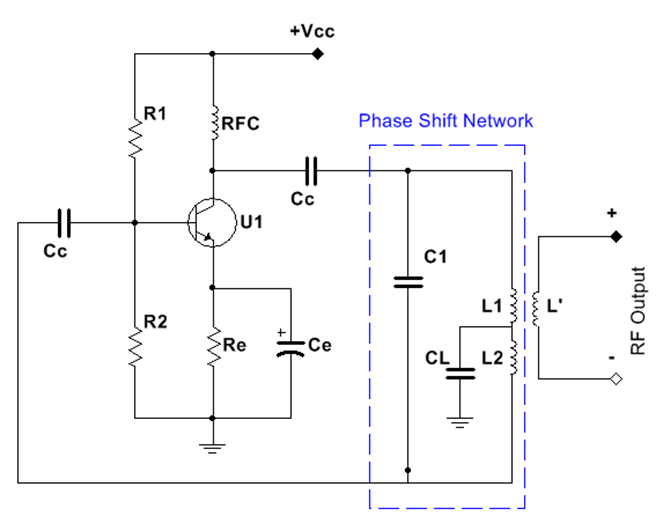


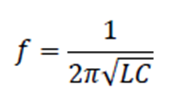
Figure 5.8c –Hartley Oscillator

The output is this amplifier is across the inductor L’ and the feedback voltage is produced by inductor L2. The coils L1 and L2 are inductively coupled to function as an auto-transformer. But since the L1 and L2 junction cannot be directly grounded, capacitor CL is connected to ground the system.

Since L1 and L2 are wounded on the same core, they share the same inductions between the coils and their net effective inductance is increased by M. The follow equation for the effective inductance is given by:



Therefore the resonant of oscillation frequency is given by:



Inductance can be adjusted easily to vary the frequency of the Hartley oscillator, which is achieved by making the core movable or varying the capacitance. Hartley oscillators are not used with low frequency because at low frequency, the value of inductance required becomes too large and it will not suit our design since we have size constraints for the inductor.

**5.9 Oscillator Parts**

In our design the oscillator is used to generate the resonant frequency of the transmitter coil and tank circuit. The circuit created by the transmitter coil has one resonant frequency. Oscillator means constant velocity bouncing back and forth between the sides of the resonator. To cause resonance, the phase of a sinusoidal wave must be equal to the initial phase so the waves self-reinforce. The waves of oscillation take a longer time to fade out. That is why in our design wireless energy transfer has a higher efficiency. See circuit in Figure 4.2.1 for the circuit diagram for the oscillator that will be used in this design.

We chose this circuit because of its simple design and its ability to produce the output signal that is needed. The main components of this oscillator circuit are listed below in Table “” along with the associated costs.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Component | Vendor | Manufacturer | Qty. | Cost/unit ($) |
| 1N4740A | Mouser | Fairchild Semiconductors | 2 | $0.22 |
| BYV26E | Mouser | Vishay Semiconductor | 2 | $0.59 |
| IRF2807 | Mouser | International Rectifier | 2 | $2.35 |
| LM7812 | Mouser | Fairchild Semiconductors | 1 | $0.64 |

**5.10 Image Processing**

The image processing unit is based off of a CMUcam5. This unit does all of the image processing at 50 frames per second and then sends only the necessary coordinate information to the microcontroller using any of the following: USART, SPI, I2C, USB, or digital/analog output.

|  |  |
| --- | --- |
| **Processor** | NXP LPC4330, dual core |
| **Clock Speed** | 204 MHz |
| **Image Sensor** | Omnivision OV9715, 1280x800 |
| **Power Consumption** | 140 mA for typical use |
| **RAM** | 264 KB |
| **Flash Memory** | 1 MB |
| **Data Outputs** | USART, SPI, I2C, USB, digital, analog |

**6 Software Design**

The high level design contains four functions which are shown in the following diagram. These functions are initialization, object detection, image processing, and control. The software will be divided into two subsystems. The NXP LPC4330 MCU will handle the image processing and the Atmel SAM3X8E will read the data from the NXP LPC4330 and the sensors, then control the motors accordingly.

**Untitled Diagram.png**

Figure 6 – High Level Design

**6.1 Computer Vision**

The computer vision will be handled by the NXP LPC4330 processor. Unfortunately, the CMUcam5 is configured to detect colors with a specific hue, making it initially unreliable to track white lines. To fix this, the firmware must be altered. NXP offers a development tool platform called LPCXpresso which is used to open the .hex firmware files and alter them. Since our project will be imitating a vehicle, the camera will be set to grayscale and the firmware will be altered to detect two lines. The location data will then be sent to the MCU which will decide how to control the robot.

**6.1.1 Lane Following Algorithm**

To have an autonomous robot follow lanes, it must first be able to detect lines. There are four basic steps for the algorithm to detect straight lines.

1. Edge detection (Can be accomplished using the Canny edge detector)

2. Map edge points to the Hough space

3. Interpretation of stored edge points to create lines of infinite length.

4. Converting the infinite lines to finite lines

Once the lines have been detected, the MCU can use the x coordinates to position the robot between the two lines.

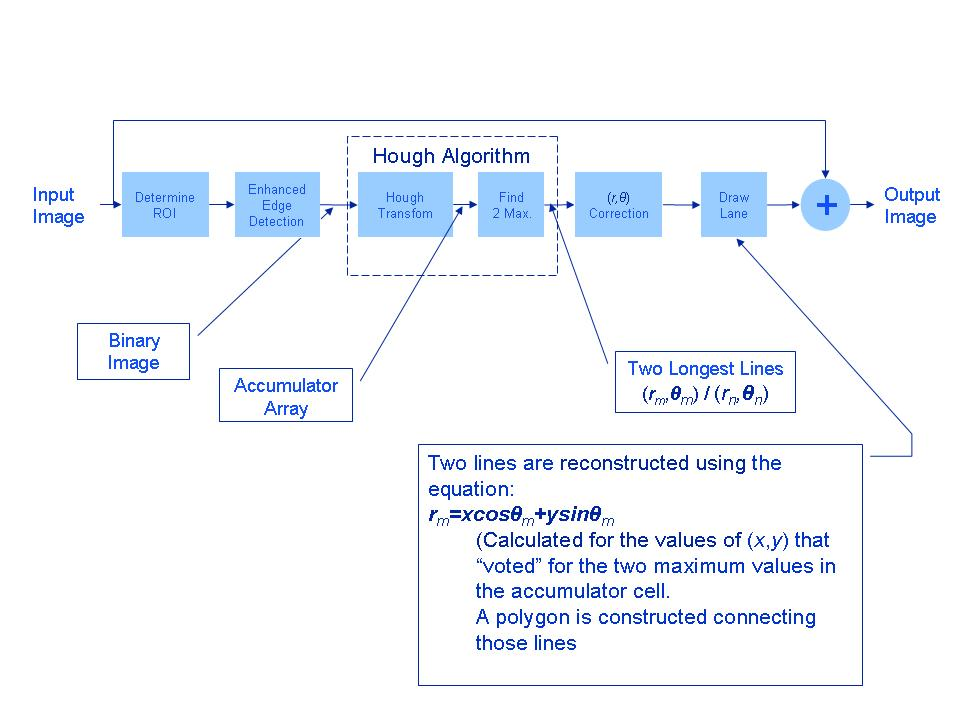
****

Fig. 6.1.1 Lane Detection Process

Reprinted with permission from cnx.org

**6.1.2 Object Detection**

To detect objects that may be stopped on the track, the robot will use an HC-SR04 ultrasonic sensor. This works by using sonar to determine the distance to an object. If the HC-SR04 sensor does not receive an echo, then the output is never low. If it does receive an echo the MCU will control the motors to stop the robot.

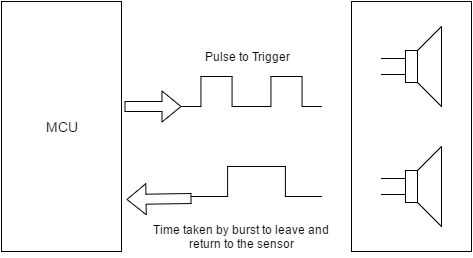
****

Figure 6.1.2 Sonar Object Detection

Although ultrasonic sensors are highly accurate, they are prone to interference from background noise causing false echos. The environment that this project will be presented in is most likely going to be full of interference from other students and faculty discussion. The best solution would be to have the NXP LPC4330 camera unit to perform the object detection as a backup in addition to the line tracking. That way if the ultrasonic sensor fails, the camera will still detect objects which are stopped on the track.

**6.2 Control System**

The control system is the Atmel SAM3X8E microprocessor. It will read the data sent by the NXP LPC4330 camera unit and control the motor drivers. The MCU will also stop the robot if any object are sensed by the HC-SR04.

**ControlSystem.png**

Figure 6.2 Control System

**6.3 Software Integration**

The following commands are useful in programming using the Arduino IDE

**Digital I/O**

pinMode(pin, [INPUT,OUTPUT]

digitalWrite(pin, value)

int digitalRead(pin)

**Analog I/O**

analogReference([DEFAULT,INTERNAL,EXTERNAL])

int analogRead(pin)

analogWrite(pin, value)

**Advanced I/O**

tone(pin, freqhz)

tone(pin, freqhz, duration\_ms)

noTone(pin)

shiftOut(dataPin, clockPin, [MSBFIRST,LSBFIRST], value)

unsigned long pulseIn(pin, [HIGH,LOW])

**Time**

unsigned long millis()

unsigned long micros()

delay(ms)

delay Microseconds(us)

**External Interrupts**

attachInterrupt(interrupt, function, [LOW,CHANGE,RISING,FALLING])

detachInterrupt(interrupt)

Interrupts()

noInterrupts()

**Servo Control**

attach(pin, [min\_uS, max\_uS])

write(angle)

writeMicroseconds(uS)

read()

attached()

detach()

**Serial**

SoftwareSerial(RxPin, TxPin)

begin(longSpeed)

char read()

print(myData)

or println(myData)

**I2C Control**

Wire(#include<Wire.h>)

begin()

begin(addr)

requestFrom(address, count)

beginTransmission(addr)

send(mybyte)

send(char \* mystring)

send(byte \* data, size)

endTransmission()

byte available()

byte receive()

onReceive(handler)

onRequest(handler)

**7 Prototype Construction**

**7.1 Parts Acquisition and Budget**

We are currently funding this project ourselves. The camera unit has already been obtained as well as the robot chassis and motors. We have an initial budget of $100 per person, or $400 total.

**7.2 Breadboard Testing**

For breadboard testing we will most likely be using the materials in the smart lab and in the TI lab.

|  |  |
| --- | --- |
| **Supplies** | **Purpose** |
| Oscilloscope | To measure the receiver, transmitter, etc. |
| Digital Multimeter | To measure the constant voltage being supplied to the robot. |
| Power Supply | To use as the dc input for the dc/ac converter. |
| Function Generator | To use to test the resonant frequency of the receiver. |
| Breadboard | To prototype our circuits |
| Soldering Iron | To solder and desolder our PCBs |

**7.3 PCB Design**

A requirement of Senior Design is that a PCB is used in this project. This will allow us to remove any unnecessary parts, as well as integrate those that will be beneficial to the project onto one board. We will be using EAGLE by CadSoft which stands for Easily Applicable Graphical Layout Editor. Some of the features include schematic capture, PCB layout, auto-router to transfer the schematic to PCB, computer-aided manufacturing, and bill of materials. Since we are inexperienced at building PCBs, it was decided that the group will use an online PCB manufacturer. Another advantage to using EAGLE is that almost all of these manufacturers accept EAGLE files and there are many resources to learn this software.

**7.3.1 Vendors**

Below is a list of PCB vendors. Most of the prices are comparable to each other.

[www.4pcb.com](http://www.4pcb.com)

[www.custompcb.com](http://www.custompcb.com)

[www.batchpcb.com](http://www.batchpcb.com)

[www.pcbexpress.com](http://www.pcbexpress.com)

[www.olimex.com](http://www.olimex.com)

**7.3.2 PCB Procedures**

Although we will be using EAGLE to design a PCB based on the Atmel SAM3X8E we thought it would be beneficial to know the steps involved in making a PCB ourselves. The pinout diagram in figure 7.3.2 is used as a guide for this process.

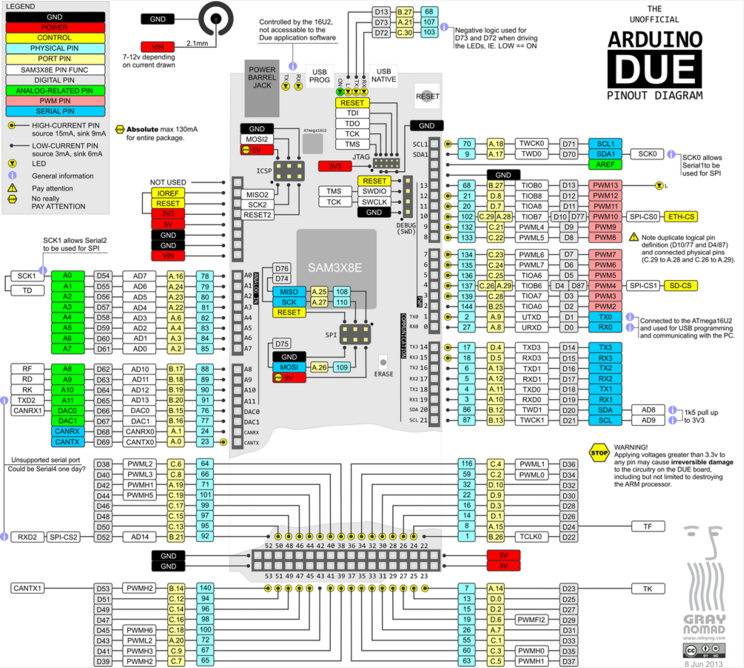


Figure 7.3.2 Arduino Due Pinout

Here are the list of the materials we will use to emulate the microcontroller if we choose not to use an online vendor for the design.

|  |  |
| --- | --- |
| Material | Anticipated quantity |
| button switch | 1 |
| crystal oscillator of 16 MHz frequency | 1 |
| tantalum capacitors with 10 uf capacitance | 2 |
| ceramic disk capacitors with a capacitance of 22 pf | 2 |
| 28 pin DIP IC socket | 1 |
| jumper wires | 12 |
| ceramic disk capacitor with a 0.1 uf capacitance | 1 |
| Atmel SAM3x8E | 1 |
| row of 6 male headers | 1 |
| resistor with a 10k ohms resistance | 1 |
| female header | 3 |
| red LED | 1 |
| resistors with 150 ohms resistance | 2 |
| green LED | 1 |
| LM1117T-3.3 voltage regulator with 3.3V | 1 |
| female wall wart power jack | 1 |
| electrolytic capacitors with 10uf capacitance | 2 |
| perfboard | 1 |

**The diagrams in step 1 through step 17 are all reprinted with permission from Arduino.cc**

**Step 1**

3 yellow LEDs and one green LED

4 1k resistors

1 LMV35IDGKR op-amp

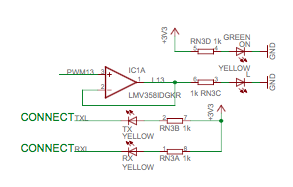


Figure reprinted with permission of Arduino

**Step 2**

1 10 nf non electrolytic capacitor

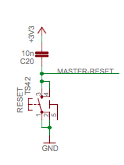


Figure reprinted with permission of arduino

**Step 3**

8 100nf non electrolytic capacitors

3 10uf electrolytic capacitors

2 MH2029-300Y filter beads

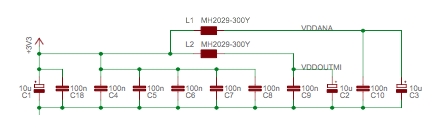


Figure reprinted with permission of arduino

**Step 4**

8 100k resistors

1 jtag

1 zero-ohm resistor

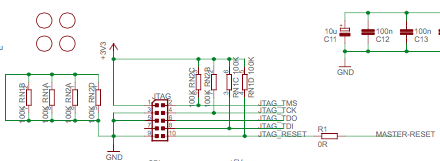


Figure reprinted with permission of Arduino.cc

**Step 5**

1 10 uf non electrolytic capacitor

6 100nf non electrolytic capacitors

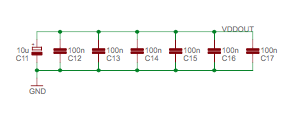


Figure reprinted with permission of Arduino.cc

**Step 6**

1 female header 8x1F-H8.5

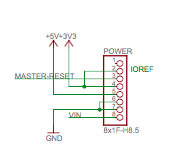


Figure reprinted with permission of Arduino.cc

**Step 7**

1 18x2F-H8.5 female header

1 3x2M male header

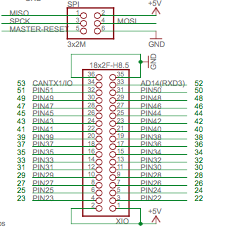


Figure reprinted with permission of Arduino.cc

**Step 8**

1 18x2F-H8.5 female header



Figure reprinted with permission of Arduino.cc

**Step 9**

1 18x2F-H8.5 female header

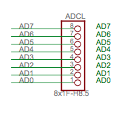


Figure reprinted with permission of Arduino.cc

**Step10**

1 1.5k resistor

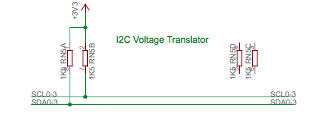


Figure reprinted with permission of Arduino.cc

**Step 11**

1 DC21 MMX power supply

1 L6MH2029-300Y filter bead

1 SS1P3L

1 47 electrolytic uf capacitor

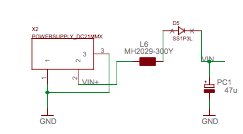


Figure reprinted with permission of Arduino.cc

**Step 12**

4 100k resistors

1 1k resistor

1 BC847BSMD npn transistor

1 LMV358IDGKR op-amp

1 NCP1117ST50T3G LDO voltage regulator

2 100nf non electrolytic capacitors

1 10 uf electrolytic capacitor

2 single P-channel mosfets

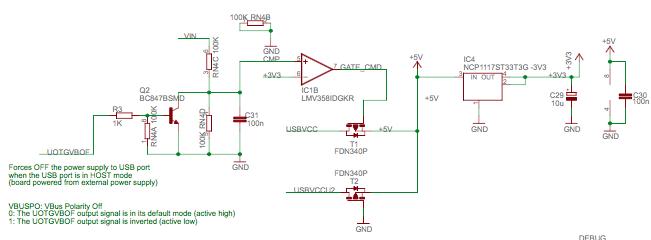


Figure reprinted with permission of Arduino.cc

**Step 13**

1 4x1V male header

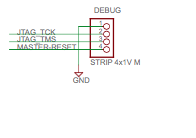


Figure reprinted with permission of Arduino.cc

**Step 14**

2 10k resistors

1 10uf non electrolytic capacitor

1 47uf electrolytic capacitor

1 10nf non-electrolytic capacitor

1 LM2734Y step down dc-dc regulator

1 52.3k resistor

1 SRR0604-100ML inductor power shielded wirewound

1 CD1206-S0175 diode

1 SS1P3L schottky diode

1 100nf non electrolytic capacitor

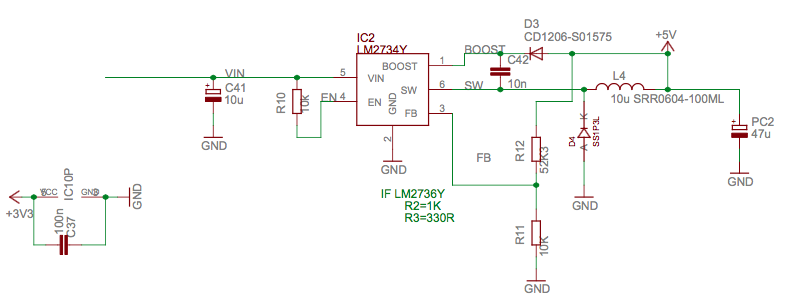


Figure reprinted with permission of Arduino.cc

**Step 15**

1 micro usb ab1

3 cg0603mlc-05e varistor

2 10uf electrolytic capacitors

1 mf-msmf050-2 resettable fuse

1 10nf non electrolytic capacitor

1 10k resistor

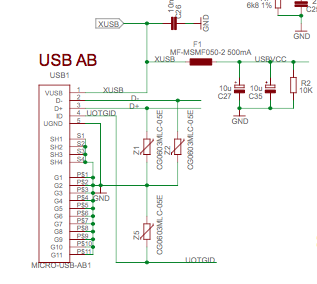


Figure reprinted with permission of Arduino.cc

**Step 16**

5 22pf non electrolytic capacitor

3 100nf non electrolytic capacitors

2 39 ohms resistor

1 zero-ohm resistor

1 6.8k resistor

1 12 MHZ kx-7 20pf crystal oscillator

1 10uf electrolytic capacitor

1 32.768 KHZ crystal oscillator

1 Atmel SAM3X8E microcontroller

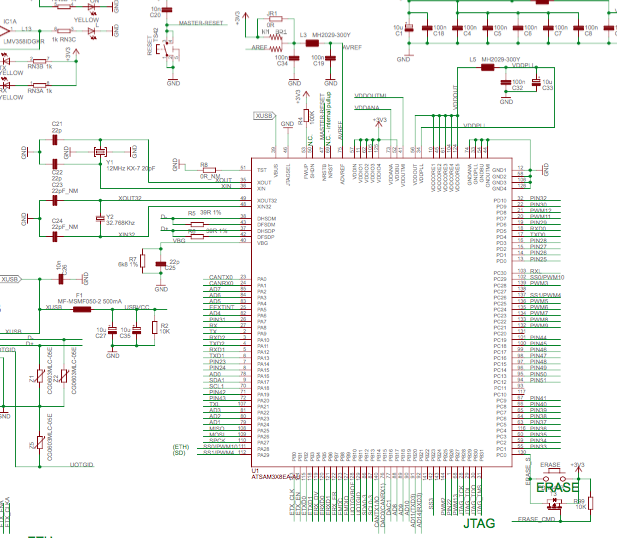


Figure reprinted with permission of Arduino.cc

**Step 17**

1 ATMEGA16U2-MV

2 yellow LEDs

1 10uf non electrolytic capacitor

2 100 fn non electrolytic capacitors

1 CG0603MLC-05E varistor

1 MF-MSMF050-2 resettable fuse

4 1k resistor

2 22ohms resistors

1 3x2 male header

1 2x2 male header

2 CD1206-S01575 diode rectifier

2 22pf non electrolytic capacitors

1 xt116MHZ kx-7 crystal oscillator

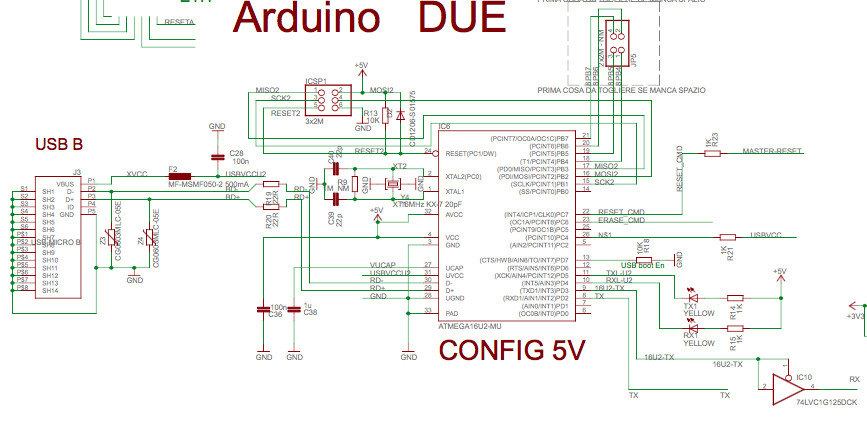


Figure reprinted with permission of Arduino.cc

All the components can be purchased from digikey.com and mouser.com. The estimated cost of all components is around 25 dollars.

After completing all the different circuit, we will then proceed with the connections and begin powering the device to test whether the PCB we designed is responsive. The next step will be to open the Arduino IDE and verify that the chip is being recognized. After that process, we can start implementing different codes to the microcontroller PCB.

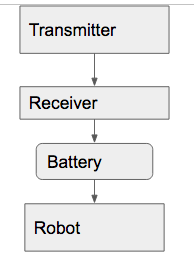
**7.4 Hardware Build Plan**

The success of this project will depend on being able to charge a moving robot wirelessly while it is moving. The concept of wireless charging has just recently been commercialized and the process may be complicated to do at once. We have decided to separate the project in two parts:

* Wireless energy
* Autonomous Robot

For the project to work smoothly, both of these parts will be built separately before being assembled together into a final prototype.

Components for prototyping wireless



Steps to build wireless energy

1. Build the necessary amount of induction coils needed along the track

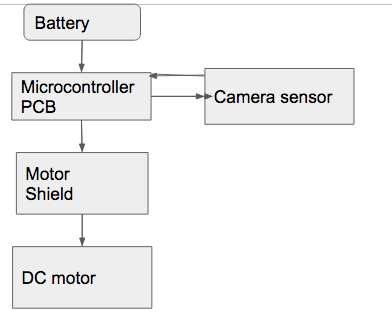
2. Apply voltage to transmitter and verify that the receiver is functioning

3. Connect the batteries and note if it is charging

4. Turn robot on

When the wireless energy functions, we can continue with the next step which is the autonomous control.

Components for prototyping autonomous control for the robot



Steps for autonomous control

1. When the battery is being used, confirm that the PCB is on
2. The camera sensor should start following the path designated on along the track, and send commands to the PCB
3. The PCB should receive the commands from the camera sensor and send to the motor shield
4. The motor shield should receive the commands and steer, speed or slow the robot down

When both of the parts can work simultaneously, we can begin recording the results to see whether we are on the right track and we can verify how fast and how effectively the batteries are being charged.

**7.5 Software Build Plan**

There will be two group members responsible for the software development. All software will be written in C and HEX, which is a high-level language with syntax very similar to Python. It also supports object oriented programming. In order to efficiently control the source code, we will be using GitHub. This will allow us to work from multiple locations and view revisions by both members. Below is the software build plan in order of what is to be completed first, as well as explanations of the importance of each for the success of the project.

1. Modifying the NXP LPC4330 Firmware

* Without the camera microcontroller detecting lines the robot will not be autonomous. This is going to be the most intensive part of the project with the steepest learning curve and should be completed first.
* The stock firmware for the NXP LPC4330 allows the camera unit to detect blobs of color. While this may be useful for other projects it is not useful for a robot that must navigate between two lines.

1. Sensor Processing

* The microcontroller must be able to process the data from both the camera unit and the ultrasonic sensor. Without this the microcontroller will not be able to send signals to the motor drivers.

1. Motor Control

* This can be implemented after part 2. Without the data from the sensors the microcontroller will not be able to send signals to control the motors.

1. Autonomous Integration

* This is the final step and should be the simplest to implement. There will be very little code written, as the majority of work will be to ensure that the previous three parts do not interfere with each other. This may require debugging and code correction.

**7.6 Track Build Plan**

For the track I personally believe method 3 stated in the research will be the best way to go, mainly because building the two coils will be easy to do at a low cost. The square law inverse must also be accounted for. This will make sending the power easier and to account for the loss of power, all that needs to be done is to increase the voltage being sent over for the power.

**8 Prototype Testing**

The following section will discuss how we will approach testing both the hardware and software prototypes.

**8.1 Hardware Testing Environment**

For the hardware testing environment we will use an oscilloscope, digital multimeter, and a power supply. For the board we will use a breadboard first to test the components, so that the circuit can be readjusted without the hassle of desoldering everything. Most of the testing will be done at either the smart lab or at the TI lab in the engineering atrium. We will also need a place where there are many people around to test our robot.

We also need a way to have a method to measure the electromagnetic field. To meet this requirement we will need to use an EMF meter or Gauss meter.

**8.2 Hardware Specific Testing**

**Receiver**

For this part of the circuit we plan to take it one step at a time, and make adjustments as we go.

To take it one step we will start with one part of the receiver and start adding on more parts or adjust it to get the working receiver that we will need.

1st we will test the bandpass filter such that it is able to receive any signals and deliver the power with little loss of energy/voltage.

2nd we will add a peak detector to obtain the goal for the system

3rd add a voltage follower to isolate the circuit from other internal resistances

**Test Name:** Receiver

**Objective:**

To deliver wireless energy to the robot, so that it is able to charge

**Supplies:**

* Oscilloscope
* Digital Multimeter
* Function Generator

**Procedure:**

1. Use the oscilloscope and DMM to measure the output of the receiver
2. Observe the output voltage

**Expected Result:**

Reciever should charge the battery

Before prototyping, we will need to test each material individually to make sure that they can work properly. Then we can make the connection needed of each component and ensure that the commands being inputted are being followed.

**Test Name**: Battery Testing

**Objective:**

The battery should be tested when it being charged and it should be tested when it is not being charged and verify that get the expected results. This process should be done with each individual battery.

**Supplies:**

* Nickel-Metal-Hydride rechargeable batteries
* Multimeter
* Load of one 1k resistor

**Preparation:**

To power the robot, we use 7 AA Nickel-Metal-Hydride rechargeable batteries that will be place in a battery holder. We would also try using 8 AA Nickel-Metal-Hydride batteries to and try to note which is more efficient for the project.

**Procedure:**

Steps to test for effectiveness of battery when charging

1. Connect multimeter to the battery
2. Check voltage
3. Check for current

Steps to test for effectiveness of battery when not charging

1. Connect a load to the battery
2. Connect multimeter to the load
3. Check voltage
4. Check for current

**Expected result:**

We expect to get the correct voltage and the correct current values when the battery is charging and when it is charging. The voltage for both parts should be 1.2V and the current with the load should be 0.12 A and without the load it should be 0 A.

**Test name: Receiver**

**Objective:**

For the Receiver’s objective we want to take an incoming signal, convert it to AC/DC and use it to power the robot.

**Supplies**

Oscilloscope

Digital multimeter

function generator

**Preparation**

1. Prepare the oscilloscope
2. Prepare the digital multimeter
3. Prepare the function generator

**Procedure**

For this part of the circuit we plan to take it one step at a time, and make adjustments as we go.

To take it one step we will start with one part of the receiver and start adding on more parts or adjust it to will get the working reciever that we will need.

1st we will test the bandpass filter such that it is able to receive any signals and deliver the power with little loss of energy/voltage.

2nd we will add a peak detector to obtain the goal for the system

3rd add a voltage follower to isolate the circuit from other internal resistances

**Expected Result**

For the expected result, we want it to be charging the battery.

**Summary**

|  |  |
| --- | --- |
| **What we are testing** | **Receiver** |
| Objective | To deliver wireless energy to the robot, so that it is able to charge |
| Supplies | Oscilloscope  Digital multimeter  function generator |
| Preparation | N/A |
| Procedure | Use the oscilloscope and DMM to measure the output of the receiver such that it is producing the correct output voltage. |
| Expected Result | Receiver should charge the battery up |

Steps to test efficient Receiver

1. Send a voltage t across the receiver at the same transmitter.

2. Measure the frequency to see it it is close to the resonant frequency.

3. Measure the output of the peak detector circuit to see if it matches the DC value.

**Objective:**

For the Transmitter we want it to take a DC input and convert it into AC.Than we want it to be coupled with the receiver, delivering the amount of voltage that we need to charge the battery.

**Supplies**

Oscilloscope

Digital multimeter

Function generator

**Preparation**

1. Prepare the oscilloscope
2. Prepare the digital multimeter
3. Prepare the function generator

**Procedure**

Use the oscilloscope and DMM to measure the output of the inductor such that it is producing the correct output voltage and frequency.

**Expected Result**A sine wave with 13605 KHz with amplitude of the sine wave being multiplied by 2\*distance\*distance

**Summary**

|  |  |
| --- | --- |
| **What we are testing** | **Transmitter** |
| Objective | Transmit DC to AC voltage |
| Supplies | Oscilloscope, Digital multimeter, and function generator |
| Preparation | N/A |
| Procedure | Use the oscilloscope and DMM to measure the output of the inductor such that it is producing the correct output voltage and frequency. |
| Expected Result | A sine wave with 13605Khz with amplitude of the sine wave being multiplied by 2\*distance\*distance |

Steps to test transmitter

1. Send Dc input signal
2. Measure the Ac output across the inductor
3. Check to see if the frequency matches up
4. Check to see if the voltage meets our requirements.

**Test Name:** Microcontroller Testing

**Objective:**

A microcontroller will be used for the prototype to ensure that the functionality of project that a PCB will be used representation will be used to emulate it.

**Supplies:**

* Microcontroller
* Computer with Arduino 1.6.5 software

**Preparation:**

The microcontroller will have two roles, which are to receive information about the surroundings and to use that information to steer the robot. To ensure its functionality, different simple codes should be tested.

**Procedure:**

1. Connect the microcontroller to a computer
2. Open arduino 1.6.5
3. Verify that the board is being recognized
4. Test some of the example codes

**Expected result:**

The microcontroller should be recognized by the Arduino software, and it should be able to perform the codes that were tested.

**Test Name:** Motor Driver Testing

**Objective:**

A L298N H bridge motor driver will be used as an intermediary between the microcontroller and the DC motors. Considering the different amount of materials that will be on the robot chassis, its total weight might cause it to move a little slower than we would like. Because the voltage pins on the microcontroller is 3.3 V and they provide a very low amount of output current, the L298N will be essential for this project.

The two main function of the motor driver shield are

* Control the directing of the chassis by controlling the DC motors
* Control the speed of the robot

The different amount of pins in a motor shield will allows the use for different quantities of voltages and a greater variety of currents. Its use will be very useful to prototype the robot because it takes away the need for to use a breadboard and the need for soldering. Also, it will protect the project from many different damages that can occur from mistakes or accidents that could have a negative impact on the different materials used on this project.

**Supplies:**

Microcontroller

Computer with Arduino 1.6.5 software

Motor shield

2 DC motors

**Preparation:**

To program the motor shield with the microcontroller, we researched online and we found the different steps indicated. We are going to follow these steps to test it and begin writing the codes necessary to control the DC motors.

**Procedure:**

Step 1: Installing the motor shield

Correctly place the motor shield onto the microcontroller by aligning the pins into the socket

Step 2: Shield Features

Identify the different channels and their function before making any connection to the microcontroller. The motor shield has 2 channels which allows for the control of two DC motors, or 1 stepper motor.

Here are the pin breakdowns:

Step 3: Program the motor shield

First we need to plug the microcontroller into the computer's USB port and open the Arduino software: Arduino 1.6.5. Then, we have to follow these three steps to initialize the motor channel.

1. Set high or low voltages to set the desired direction for the motor
2. Set the brake to pin for the motor to low
3. Use PWD command (analogWrite) to set speed
4. Verify if the motor is moving

To ensure functionality we will use codes for one motor, then for two motor just in case something doesn’t work well and we need to pinpoint the issue.

**Expected result:**

The motor shield should be smoothly connected with the microcontroller and it should be able to perform the functions it was assigned and move the DC motors

**Test Name:** DC motor testing

**Objective:**

The DC motor will be used to control the wheels of the chassis. It will be receiving the commands of the microcontroller from the the motor shield and determine the speeds of the wheels for steering. DC motors normally have just two leads, one positive and one negative. If you connect these two leads directly to a battery, the motor will rotate. If you switch the leads, the motor will rotate in the opposite direction.

**Supplies:**

* DC motors
* Battery
* Two wires

**Preparation:**

First we can verify the polarity of the motors being used. A DC motor has two leads on the top of it. By connecting one lead to the positive terminal of the battery and the other to the negative terminal, we can verify the polarity of the motor in question.

**Procedure:**

Connect the lead with the two wires to the battery in one direction

Switch the two wires

Connect the lead to the battery

**Expected results:**

When the battery is connected to the DC motor it should begin rotating in one direction. After the two wires are switched, we expect to see the motor rotate in the opposite direction. When all the motors are connected to the chassis, it gets more complex when we are faced with having to direction. Because the wheels are fixed and can only go one direction, we have to vary the speed on each side of the robot to change direction. This implies that the

two wheels on the right would need to be slower than those of the left to move right and faster to move left. The motor shield has individual pins for each of the 4 DC motors that will be used and we will implement different codes to test the movement to ensure that the motor are moving the chassis in the desired direction.

**8.3 Software Testing Environment**

The software will be tested using group members’ personal computers. This will allow tests to be done independently of each other and to quickly prepare for integration of the subsystems. The software will first be written and complied on any IDE which supports Arduino such as Eclipse, Visual Studio, or the Arduino IDE. The code itself will be hosted on Github which is a Git repository. Every member will be able to access the most up-to-date version of the code by accessing GitHub. This will also ensure the integrity of the code from hardware or user errors.

**8.4 Software Specific Testing**

The hardware and software are very closely related for the robot, so there may be some similar tests for the software. This section will include all testing and expected outcomes for the autonomous navigation of our robot which includes computer vision, sensor processing, autonomous navigation, and movement control.

**8.4.1 Computer Vision**

**Test Name:** Camera Initialization

**Objective:**

The objective of this test is to verify that the NXP LPC4330 camera microcontroller is recognized and connected to the Atmel SAM3X8E microcontroller. This is important because the firmware of the camera unit will be changed to run the proper algorithms. Because of this errors may occur when first connecting the camera. This is a simple test to see if the camera is successfully communicating after the firmware changes.

**Supplies:**

* NXP LPC4330 camera unit
* Atmel SAM3X8E PCB
* Computer
* SPI connection cable

**Preparation:**

The computer must have an Arduino compatible IDE installed. Download the required Arduino library for the camera and import it. Note: This is the default program and the camera will only print to console if a line with the correct color signature is detected. Connect the camera unit to the Atmel SAM3X8E microcontroller.

**Procedure:**

1. Open the Arduino IDE
2. Run the “hello\_world” program
3. Observe the output to the serial monitor

**Expected Result:**

The camera unit should output to the console successfully. The content of the data should be similar to:

Detected 1:

block 0: sig: 1 x: 159 y: 109 width: 15 height: 61

Detected 1:

block 0: sig: 1 x: 173 y: 114 width: 14 height: 61

Detected 1:

block 0: sig: 1 x: 146 y: 111 width: 15 height: 62

This means that the camera and MCU are communicating with each other.

**Test Name:** Line Detection

**Objective:**

The objective of this test is to verify that the NXP LPC4330 camera unit detects lines and sends the corresponding data back to the Atmel SAM3X8E microcontroller.

**Supplies:**

* NXP LPC4330 camera unit
* Atmel SAM3X8E PCB
* Computer
* SPI connection cable
* Lines on acrylic sheet

**Preparation:**

The computer must have the Arduino IDE and PixyMon software installed. Make sure the devices are turned on, then connect the camera unit to the Atmel SAM3X8E microcontroller and connect the PCB to the computer

**Procedure:**

1. Place the acrylic sheet with lines in front of the camera
2. In the IDE call the getBlocks() function
3. View the serial monitor and record the location data sent by the NXP LPC4330
4. Compare this to the location data on the PixyMon software

**Expected Result:**

The camera should detect the lines, process the image, and send back the corresponding data to the Atmel SAM3X8E microcontroller. The data should look similar to the lines below.

Detected 1:

block 0: sig: 1 x: 10 y: 109 width: 14 height: 150

Detected 1:

block 1: sig: 1 x: 173 y: 110 width: 14 height: 150

**Test Name:** Object Detection with Camera Unit

**Objective:**

The ultrasonic sensor will be the main method for object detection. However, ultrasonic sensors may not be reliable if the environment is too noisy. The camera can be used to provide a backup in the case of failure by the other sensor.

**Supplies:**

* NXP LPC4330 camera unit
* Atmel SAM3X8E PCB
* Computer
* SPI connection cable
* Stationary object
* Robot chassis

**Preparation:**

The computer must have the Arduino IDE and PixyMon software installed. Make sure the devices are turned on, then connect the camera unit to the Atmel SAM3X8E microcontroller and connect the PCB to the computer

**Procedure:**

1. Place the stationary object within the camera’s view
2. In the IDE call the getBlocks() function
3. Manually move the robot chassis forward towards the stationary object
4. View the serial monitor and observe the data sent by the camera unit

**Expected Results:**

The camera should detect the object and send back data that the width and height of the robot are increasing. This corresponds to the robot moving closer towards the stationary object. The recorded data should look similar to the example below.

Detected 1:

block 0: sig: 1 x: 159 y: 109 width: 70 height: 50

Detected 1:

block 0: sig: 1 x: 160 y: 111 width: 80 height: 60

Detected 1:

block 0: sig: 1 x: 146 y: 111 width: 90 height: 70

**8.4.2 Sensor Processing**

**Test Name:** Ultrasonic Sensor Initialization

**Objective:**

To test if the HC-SR04 ultrasonic sensor is connected and sending data to the Atmel SAM3X8E microcontroller

**Supplies:**

* HC-SR04 ultrasonic sensor
* Atmel SAM3X8E PCB
* Breadboard
* Jumper wires
* Computer with Arduino IDE

**Preparation:**

Connect the pins below using a breadboard, then import and include the NewPing library in the Arduino IDE.

|  |  |
| --- | --- |
| **HC-SR04** | **Atmel SAM3X8E PCB** |
| GND | GND |
| VCC | 5V |
| Echo | Pin 12 |
| Trig | Pin 13 |

**Procedure:**

1. Power on the Atmel SAM3X8E PCB
2. Put an object in front of the HC-SR04 and observe the data in the serial monitor.
3. Remove the object that is in front of the HC-SR04 and observe the data in the serial monitor.

**Expected Results:**

The ultrasonic sensor should detect the objects in front of it. If the HC-SR04 does not receive a return echo then the output will remain low. When an echo is detected then the corresponding data will show that for the echo parameter in the serial monitor.

**Test Name:** Ultrasonic Sensor Object Detection

**Objective:**

To test if the HC-SR04 ultrasonic sensor is detecting objects that are placed in the path of the robot.

**Supplies:**

* HC-SR04 ultrasonic sensor
* Atmel SAM3X8E PCB
* Computer with Arduino IDE
* Robot
* Acrylic track for the robot

**Preparation:**

Connect the pins using the same configuration as the Ultrasonic Sensor Initialization test, then import and include the NewPing library in the Arduino IDE.

**Procedure:**

1. Power on the Atmel SAM3X8E PCB
2. Manually push the robot forward with the HC-SR04 connected to the front of the robot
3. Put an object in front of the HC-SR04 and observe the data in the serial monitor.

**Expected Results:**

The ultrasonic sensor should detect the object in front of it. If the HC-SR04 is working properly the time between echoes will decrease in the serial monitor.

**8.4.3 Autonomous Navigation**

**Test Name:** Lane Following

**Objective:**

To objective of this test is to verify that the robot will stay between two lines without prompting from an external control source.

**Supplies:**

* Atmel SAM3X8E PCB
* NXP LPC4330 camera unit
* Fully assembled robot
* Acrylic test track

**Preparation:**

Using a fully assembled robot, ensure that the microcontrollers are communicating with each other as well as the motor drivers.

**Procedure:**

1. Place the robot on the track between the two lines
2. Power on the PCB and camera unit
3. Observe the robot’s behavior as it navigates through the course

**Expected Results:**

The robot should navigate through the course, detecting the lines and controlling the motors accordingly.

**Test Name:** Forward Collision Avoidance

**Objective:**

The objective of this test is to determine if the robot will stop if its path on the track is blocked. This will be tested in several locations for the ultrasonic sonar to check its effectiveness with varying sound levels.

**Supplies:**

* Atmel SAM3X8E PCB
* NXP LPC4330 camera unit
* HC-SR04 ultrasonic sensor
* Fully assembled robot
* Acrylic test track

**Preparation:**

Using a fully assembled robot, ensure that the microcontrollers are communicating with each other as well as the motor drivers. Also check that the HC-SR04 is communicating with the PCB.

**Procedure:**

1. Place the robot on the track
2. Once the robot is moving, place an object in its path
3. Observe the behavior of the robot
4. Repeat steps 1-3 in environments with different noise levels

**Expected Results:**

The robot should stop first using the ultrasonic sensor and in environments where interference is an issue it should stop using the camera as an emergency backup.

**8.5 Energy Transfer Testing**

The following tests will be used to check the legitimacy and stability of our designs performing the wireless energy transfer. We will be running distance test, alignment test, temperature test, coil inductance test, and efficiency test.

**Test Name:** Distance Check

**Objective:**

Checking the distance is an important aspect of our design because we need to ensure that maximum charging efficiency is obtained. Since the base of the robot car is approximately 5 inches from the top of track to our design should work easily within that range. The objective of this test is to verify that the output power received by the receiver coil is transferring when the height is varied.

**Supplies:**

* Voltmeter
* Measuring tape
* Prototype charging apparatus

**Procedure:**

* Disconnect rectifier circuit on robot car and connect the voltmeter across the rectifier output leads.
* Place the receiver coil on the track platform
* Make sure the receive coil is directly aligned to the transmitter coil mounted under the track.
* Power the transmitter coil.
* Vary the height of the receiver coil and record the output voltage seen on the voltmeter.

**Expected Results:**

This test is successful if we obtain 12 Volts or greater in the operating range which is between 2 inches to 5 inches above the ground. If this test fails enough power is not transferring to the receiver coil and so that means adjustment to our circuit will have to be made.

**Test Name:** Alignment Test

**Objective:**

Once we have determined the height of our robot car undercarriage, we would like to know implications of misalignment between transmitter and receiver coil while the robot car is in motion. We are testing is to verify the output power received by the receiver circuit when the alignment of the receiver coil is varied and moves along different positions of transmitter coil on the track.

**Supplies:**

* Voltmeter
* Measuring tape
* Figure- 8 Track
* Receiver Circuit

**Procedure:**

1. Disconnect rectifier (receiver) circuit on the robot car and connect the voltmeter across the rectifier output leads.
2. Place the receiver coil on the figure-8 track over the transmitter coil at the necessary height of the robot car.
3. Start with the coils perfectly aligned above each other.
4. Power the transmitter coil.
5. Move the receiver coil parallel to the track, and vary one inch at a time
6. Record the voltage output with the voltmeter.
7. For each movement over the track, record the voltage seen on the voltmeter.

This test is successful if we obtain 8 Volts or greater with a misalignment of 2 inches off center. If this is not accomplished the magnetic field is not producing enough power, therefore transmitter coil will have to be either re-wounded or changes to the design must be made.

**Test Name:** Temperature Test

**Objective:**

There will be copper losses in the coils and cause them to heat up. To monitor that we will time the operation and vary time without to predict how long it take the system to overheat. The objective is to verify that the coils operate within a safe temperature range.

**Supplies:**

* Thermometer
* Receiver circuit
* Stopwatch

**Procedure:**

1. Power the transmitter coil.
2. Use the thermometer to check the coil temperature.
3. Repeat, and increase in time of recording temperature readings.

**Expected Results:**

The test is successful if the coil temperature remains below 80 degrees Fahrenheit. If this test fails, the size of the coil wire is too small or the charging current is too high and needs to be adjusted.

**Test Name:** Coil Inductance Measurement

**Objective:**

To obtain the inductance of the coil after its final shape has been formed.

**Supplies:**

* Inductor
* Signal Generator
* Oscilloscope
* 100 ohm resistor precise to within 1% or less

**Procedure:**

1. Connect the inductor coil in series with the resistor.
2. Using a function generator and connect a sine wave of 1Vpeak-peak to the circuit
3. Monitor both the input voltage and the voltage where the indictor and resistor meet. Adjust the current until the junction voltage where the inductor and resistor meet is half the input voltage.
4. Record the frequency of the current.

**Expected Results:**

The relationship between R and L is:



**Test Name:** Efficiency Test

**Objective:**

The efficiency of the wireless charging system will tell us how much of the power being produced is actually being received during the wireless energy transfer and going into the receiver circuit. This will help us understand the practicality of this project.

**Supplies:**

* Voltmeter
* Receiver Circuit

**Procedure:**

1. Connect the voltmeter across the rectifier output leads
2. With the robot car in place power the transmitter coil by turning on the oscillator
3. Use the voltmeter to measure and record the voltage at the input of the oscillator (this is Vin)
4. Use the voltmeter to measure and record the voltage across the rectifier output leads on the car

**Expected Results:**

The efficiency of the system is calculated by the equation:

 .

The goal is to achieve about 40% of efficiency for out prototype.

**9 Administrative**

**9.1 Project Milestones**

**Wireless energy**

1. Build the Transmitter and receiver inductors such that they are able to couple and the correct inductance
2. Build the transmitter to inductor such that they are able send a sinusoidal wave at a specified frequency and voltage that we need to send the signal out.
3. Build a receiver such that they are able to receive that specified frequency at the correct amplitude that is needed..
4. Repeat for the next 2, 4, and 8 V cell so that we are able to get the desired power we need.

Note step for will be consider if we need to cut the power in the signals into two parts.

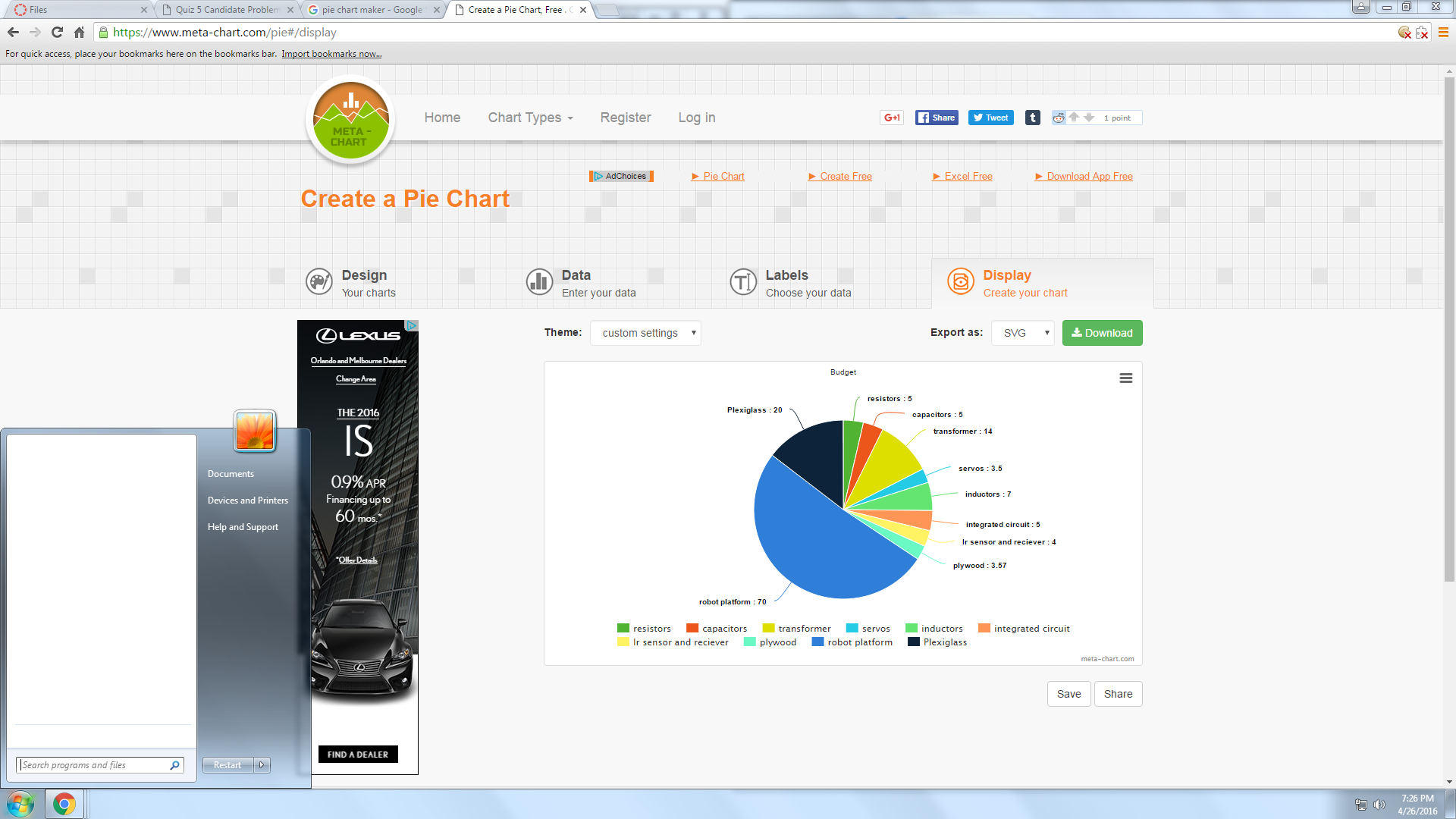
|  |  |  |  |
| --- | --- | --- | --- |
| **Task** | **Begin** | **End** | **Length** |
| Ideas for Project | 1/11/2016 | 1/18/2016 | 7 days |
| Define | 1/18/2016 | 1/25/2016 | 7 days |
| Research | 1/25/2016 | 3/7/2016 | 42 days |
| Design | 3/7/2016 | 3/28/2016 | 21 days |
| 1st Report | 3/28/2016 | 4/28/2016 | 30 days |
| Prototype | 8/15/2016 | 10/17/2016 | 63 days |
| Testing | 10/17/2016 | 11/7/2016 | 21 days |
| 2nd Report | 11/7/2016 | 11/28/2016 | 21days |

**9.2 Project Budget**

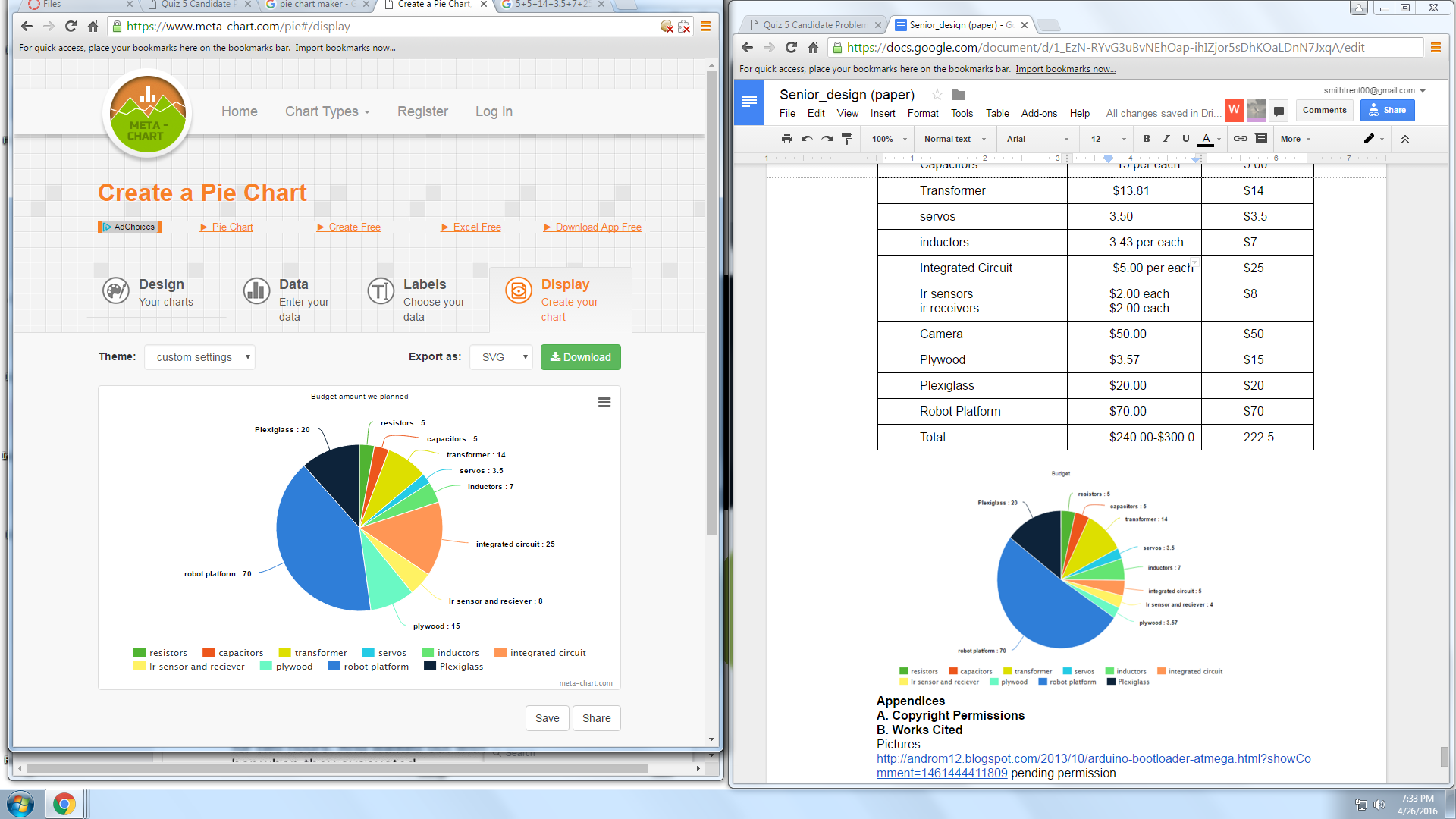
**Materials & Budget**

|  |  |  |
| --- | --- | --- |
| Material | Cost | Amount invested |
| Resistors | .14 per type | $5.00 |
| Capacitors | .15 per each | $5.00 |
| Transformer | $13.81 | $14 |
| Servos | $3.50 | $3.50 |
| Inductors | $3.43 per each | $7 |
| Integrated Circuit | $5.00 per each | $25 |
| HC-SR04 | $1.25 | $1.25 |
| NXP LPC 4330 Camera Unit | $70.00 | $70 |
| PCB | $66 | $66 |
| Plywood | $3.57 | $15 |
| Plexiglass | $20.00 | $20 |
| Robot Platform | $70.00 | $70 |
| Total | $240.00-$300.0 | $222.5 |

Below is the graph for the research parts of what we will most likely being spent on. As projected it looks like about one third of the money spent will go towards the robot platform.



Below is a pie chart of the money that we planned on spending.



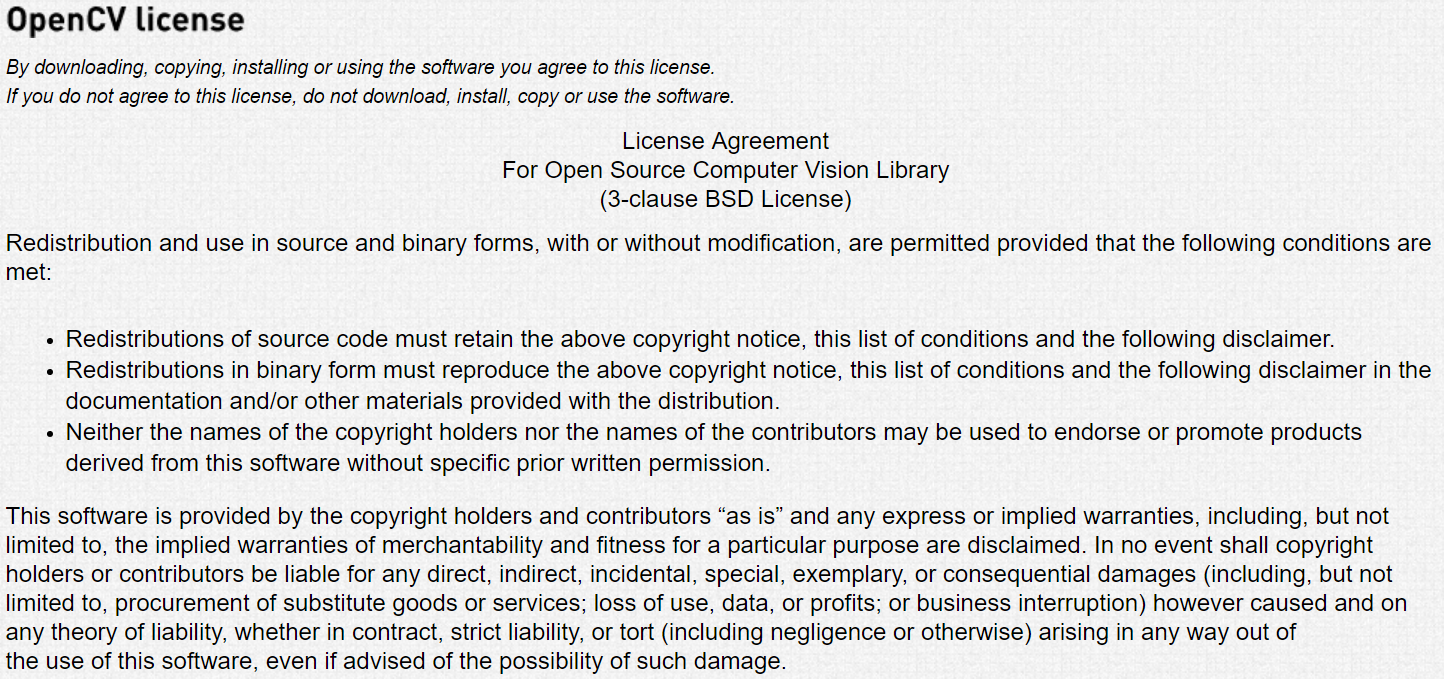
**9.3 Project Summary**

This project will be a major learning curve to engineering field of study. Not only will this project allow us to build device but it will allow us to learn how to work with a team. Learning how to work together, meet deadlines, organizing information, make schedules, be professionals and build character. We will beginning building the prototype of this summer. This will give us a chance to understand all the components and do the necessary testing, before we come up with the finish product. We will be able to show off our strengths and learn from our weaknesses. W.E.A.R is going to give us the full engineering experience before we graduate and go into our respective fields knowledgeable.

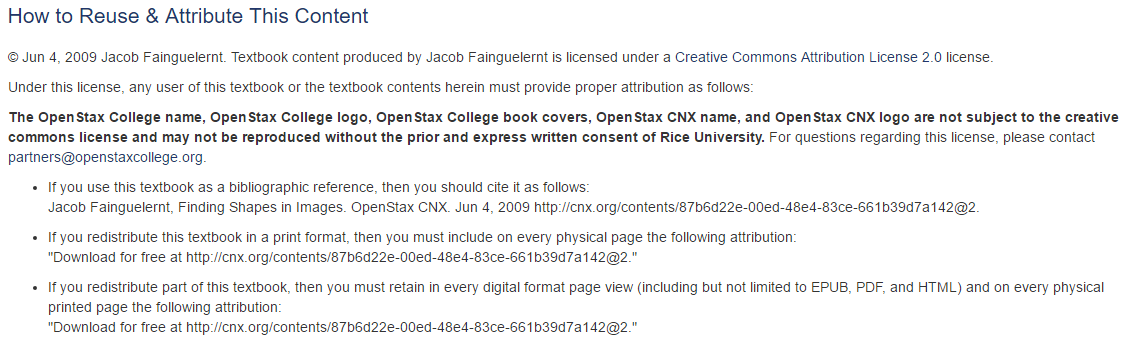
**Appendices**

**A. Copyright Permissions**

**Figure 3.6.3**



**Figure 6.1.1**



**Figures**

**3.4.1**

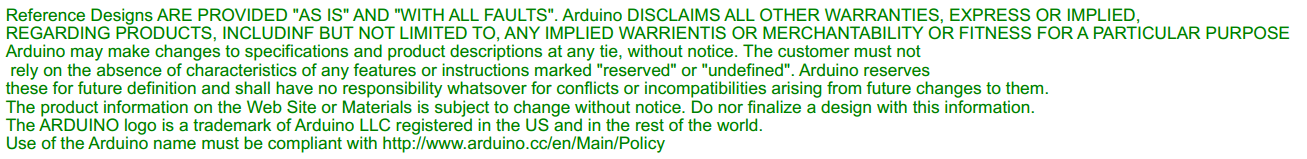
**3.4.2**

**3.4.4**

**5.1.1**

**7.3.2**

**7.3.2 Step 1-17**



**B. Works Cited**

Pictures

<http://androm12.blogspot.com/2013/10/arduino-bootloader-atmega.html?showComment=1461444411809> pending permission

<http://www.learningaboutelectronics.com/Articles/Atmega328-pinout.php>

Pending permission

<https://en.wikipedia.org/wiki/Resonant_inductive_coupling> pending permission

<https://www.sparkfun.com/datasheets/Robotics/L298_H_Bridge.pdf>

<https://learn.adafruit.com/adafruit-motor-shield/using-dc-motors>

<https://www.arduino.cc/en/Main/ArduinoBoardDue>

<http://www.robotshop.com/en/sharp-gp2y0a710k0f-ir-range-sensor.html>

<http://www.eecs.ucf.edu/seniordesign/sp2015su2015/g09/Archive/ASRSD1FinalReport.pdf>

<http://www.eecs.ucf.edu/seniordesign/sp2014su2014/g04/files/sd1-TI-100%20Watch%20Dog%20SD1%20FINAL%20DRAFT.pdf>

[www.arduino.cc](http://www.arduino.cc)

<http://cnx.org/contents/UAIFGC3y@1.2:k8WN2x28@2/Edge-Detection>

**C. Software Used**

Multisim

Meta-chart

draw.io