

# Microcontroller Based Inductance Meter

By

David Nguyen

Advisor: William Ahlgren

Senior Project

ELECTRICAL ENGINEERING DEPARTMENT

California Polytechnic State University

San Luis Obispo

Spring 2011 – Fall 2011

## Table of Contents

List of Figures .....	iii
Acknowledgements .....	iv
Abstract .....	v
I. Introduction .....	1
II. Background .....	2
III. Requirements .....	3
IV. Design .....	4
V. Results .....	6
VI. Conclusion .....	9
VII. Future Works .....	10
Citations .....	11
Appendix A.....	12
Source Code .....	12
Appendix B .....	16
Inductor Derivation .....	16
Sample Calculations .....	16
Appendix C .....	17
Component Legend .....	17
Appendix D: Analysis of Senior Project Design .....	18
Summary of Functional Requirements.....	18
Primary Constraints .....	18
Economic.....	19
If Manufactured on a Commercial Basis.....	20
Environmental .....	20
Manufacturability .....	20
Sustainability .....	21
Ethical.....	21
Health and Safety .....	21
Social and Political.....	22
Development .....	22

## List of Figures

Figure 1: General Design .....	4
Figure 2: Overall Black Box System .....	5
Figure 3: Resonant LC Input Frequency .....	6
Figure 4: Output Comparator Voltage without Pull-Up Resistor .....	7
Figure 5: Output Comparator Voltage with Pull-Up Resistor .....	7
Figure 6: Capacitor .....	17
Figure 7: Ground.....	17
Figure 8: Inductor .....	17
Figure 9: Comparator Operational Amplifier .....	17
Figure 10: Resistor.....	17
Figure 11: Voltage Source .....	17

## Acknowledgements

Friends and Family:

I would like to thank my friends and family for the support they have given me through the years. Without their help, I would not have been successful in completing my project and college career. Thank you everyone.

William Ahlgren:

Thank you for picking up my senior project idea. Initially I did not know what to create as my senior project. With your ideas and guidance, I was able to come up with an idea that I was able to successfully complete and I learned so much in the process. Thank you very much, Professor Ahlgren.

## Abstract

The scope of this project is to design and create a microcontroller based inductor meter. By taking advantage of the natural resonant frequency of an LC circuit, the inductance of an unknown inductor can be easily measured if capacitance is defined. The system is constructed by converting the natural sinusoidal waveform of the circuit to a square wave where the microcontroller can measure the period of one cycle. Multiple theories and design circuits were researched before the most efficient and precise time measurements were implemented. Software required for this project includes Atmel's AVR Studio, while hardware includes the STK600 development board and ATmega2560 microcontroller and other additional components. With this project, one can instantaneously change inductors and measure its value without any hassle. This design will convert the sinusoidal resonance of the LC circuit to a square wave waveform in order for the microcontroller to read the time period. This is driven by an interrupt signal based timer that counts for every instance the interrupt happens. After completing this project, I learned about the limitations of the microcontroller used and what effect it has on the overall calculations of the inductor.

## I. Introduction

Creating a system to measure inductance requires very precise instruments. Hardware selection should be selected with the least amount of tolerance. This is to ensure that all parts, when programmed, are working near ideal nominal values. There are many different approaches we can take to measure the inductance of an unknown inductor; however the most widely used technique is to use the resonant frequency between the unknown inductor and constant capacitor to your advantage. Essentially, the microcontroller will be able to measure how long a period of this resonance to determine the frequency between the two components. From there, it is simply a single formula one must use to calculate the mysterious inductance. Since this technique is mostly component dependent, measured values used in calculations must be used during the actual programming of the microcontroller. This means that all components in this project will be measured beforehand and any formulas used by the program will have these values stored internally. The program will not assume that all components are at nominal values, which will result in a higher and more accurate measurement. There are many factors that will contribute greatly to the outcome of the calculated inductance. It is important to reduce these factors in order to achieve the closest measurement of the inductor.

## II. Background

Between any known inductor and capacitor in parallel, there is a frequency that is always present. This frequency is independent of the source given to these two components. Resonance can be measured using:

$$f = \frac{1}{2\pi\sqrt{LC}}$$

Since this calculates the resonant frequency of the parallel LC circuit, we can use the same equation to calculate an unknown inductance value by simply rearranging the equation and if the resonant frequency has been measured beforehand. Instead, we can use the following formula as a basis of our project:

$$L = \frac{1}{4\pi^2 f^2 C}$$

The resonant frequency can be measured and seen by using an oscilloscope, however only its period is important to determining the unknown inductance, L. The microcontroller cannot measure the high and low of this waveform without the help of an ADC, and even then the times and values are not precise. It is then necessary to convert a sinusoid frequency to a square wave, while at the same time preserving the frequency. This can be done with the help of a high speed operational amplifier that will do most of the work to convert the signal to something more readable. When using a comparator and pull up resistor, the output will become a clear square wave that the microcontroller can work with. This setup sets the foundation for the entire project and the only thing left is to write instructions to record the period of a single cycle.

### III. Requirements

The requirement of this project is to simply create an inductor meter that uses some kind of microcontroller. The microcontroller will be the sole provider of power to the circuit. This means that the controller will have to be able to provide a constant 5 V square wave with a 50% duty cycle while at the same time have enough time to accurately measure the period of the output. Since each of our components is non-ideal, each will have a value that will be different than the stated nominal values. With all parts being within 10% tolerance, the goal of this project is to measure and calculate the unknown inductance within a 10% margin as well. This will simply prove that the formula and theory works out correctly and if one were to recreate the project, lower tolerance values can be used to achieve a lower percent error while at the same time the code for the program may be reused and untouched. After measuring the time period of the output square wave, the program will ultimately calculate the inductance and show the value on an LCD display. The controller has to easily communicate with the LCD so that it updates efficiently without any delay. Since the frequency of resonance acts at a much higher frequency than the input, there has to be a limit to how frequent the measured value can be updated in the LCD. For this project, the LCD should be able to update every one second to avoid any ghosting issues with the LCD display.



## IV. Design

The overall design of this project can be seen in Figure 1 below. This diagram utilizes a function generator that energizes the circuit. Current is drawn through the input resistor as it is given an AC voltage source. From the circuit, the unknown inductor is in parallel with a known capacitor. This will act as an input waveform to the Comparator Operational Amplifier. When the oscillating waveform of the LC circuit is compared to ground, the output of the Comparator Operational Amplifier is simply a square wave with the same period as the LC's sinusoidal waveform.

We can use this knowledge to our advantage when writing the program. Since the output is simply a square wave waveform, a microcontroller can simply detect this as a high or low. When the voltage change from low to high, a timer is initiated. It will continue to record until the voltage has changed from high to low. This measurement is simply half of the oscillating period, so doubling it and inversing the time will give us the resonant frequency we desire.

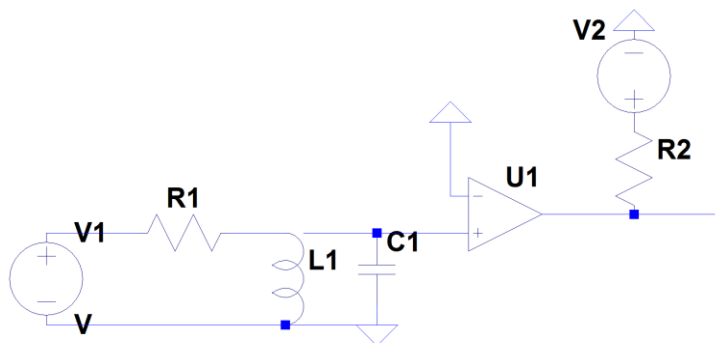


Figure 1: General Design

To implement the function generator for the input waveform, a simple while loop that triggers high and low every 1ms is sufficient to energize the input inductor and capacitor. In Figure 1, V1 is the result of this while loop. This connects to the physical components on the rest of the board.

The Comparator Operational Amplifier will compare the resonating frequency of L1 and C1 with ground. When the voltage is higher than ground, the output will become high. This is because the output of the comparator is connected to a pull-up resistor. However if the sinusoid is lower than ground, the output of the comparator will be zero. This makes it so that the microcontroller will be able to measure and calculate the period of the sinusoid without having to measure each exact point of the sinusoid. As a result, the microcontroller will easily measure the resonant frequency and the code required to measure a single period is dramatically decreased.

The black box diagram of the overall system can be seen below. Essentially, V1 and V<sub>out</sub> is part of the microcontroller code, while the rest of the circuit are actual components with non-ideal tolerances.

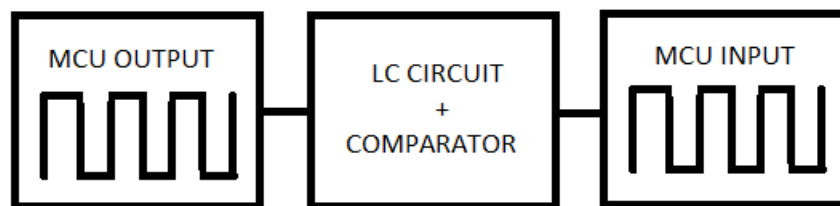


Figure 2: Overall Black Box System

## V. Results

The theory behind the resonant LC frequency can be seen in the next three figures. For educational purposes, 3 different types of inductors were used and measured using the finalized system. For the purpose of this write up, a 100  $\mu\text{H}$  inductor was used as L1 in Figure 1. In Figure 3, we can see the resonant frequency of the LC input voltage. At this point, the microcontroller is unable to measure the period because of the difference in amplitude of the decaying voltage. It is important to note that the input frequency is only 1kHz. However the frequency at which

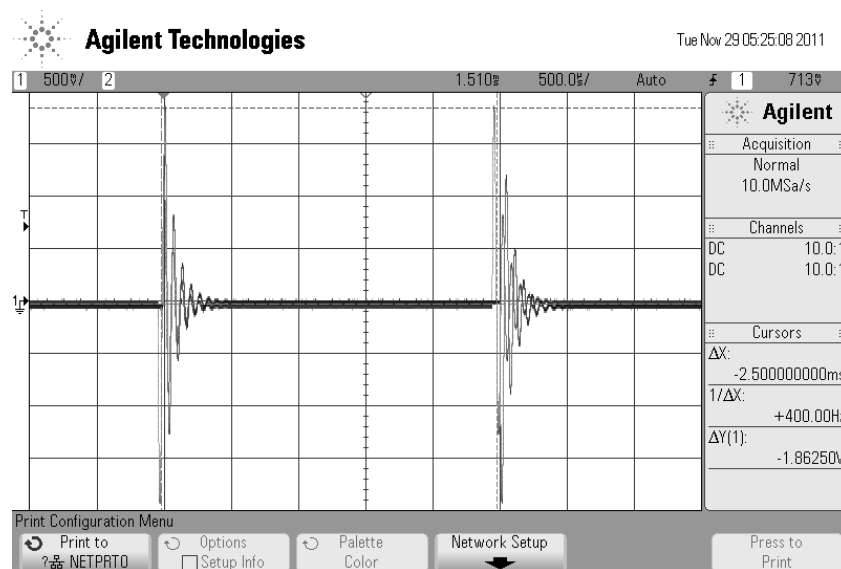


Figure 3: Resonant LC Input Frequency

Figure 3 is the input voltage of the comparator, where it is compared to ground. At the output, Figure 4 is measured without any pull up resistors. At this point, the amplitude of the comparator's output is simply 500 mV. The amplitude is not important to us because any voltage above ground will create a signal high if it a pull-up resistor is attached. The only measurement that is important to us is the period. We can see that the period is measured at 15.337 kHz. When

this resonant frequency is used in determining the inductance of our unknown inductor, the calculated value is 107.686  $\mu\text{H}$  (see Appendix B for calculations).

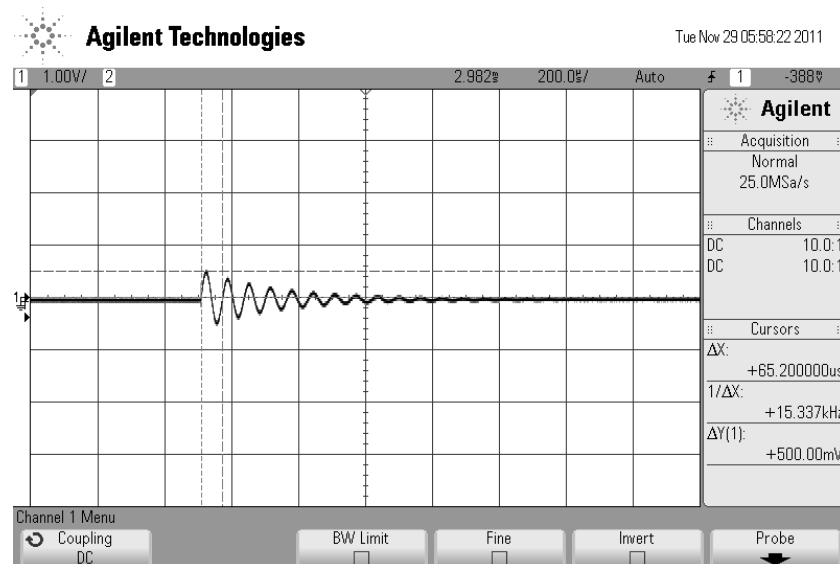


Figure 4: Output Comparator Voltage without Pull-Up Resistor

Figure 5 shows what happens when we add a small pull-up resistor to the output of the comparator. This essentially makes the entire output waveform a digital signal that the microcontroller can easily read and translate.

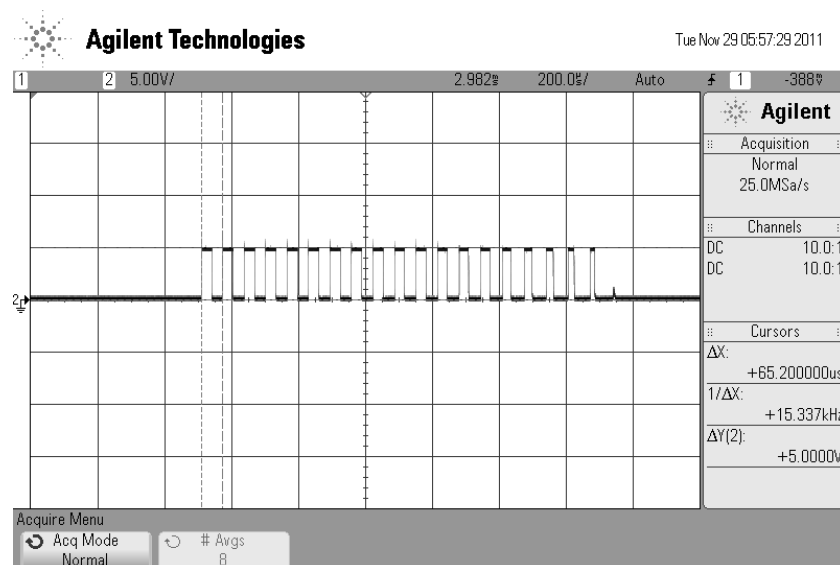


Figure 5: Output Comparator Voltage with Pull-Up Resistor

It is important to note that the voltage at this point is 5V. This is because the pull up resistor attached to the output waveform is another 5V source. This is simply tied to one of the power sources on the STK600 development board. Instead of 500 mV, the voltage will be 5V. The unique thing about this design is that the frequency between Figure 4 and Figure 5 is maintained. Both are read as 15.337 kHz, however Figure 5 is more applicable and friendly to the microcontroller.

This part ends the hardware portion of the project. Software on the other hand requires the use of timers and interrupts to measure the 15.337 kHz. In the final measurement, the 100 uH inductor read as 112 uH. This means that there are timer limitations of the microcontroller that adds to the inaccuracies of the final measurement. From the oscilloscope graphs in the previous page, we can see that the microcontroller can ideally measure the inductance as 107.686 uH, however the final display measurement shown on the LCD display is 112uH. This means that there is a 4% error due to the timing of the microcontroller. This is because we used a polling type interrupt instead of an immediate interrupt type. The microcontroller could possibly start the counter half a step too early or half a step too late. Since each cycle takes around 16 us, there can be a maximum of 32 us delay between the actual time and the recorded time. Different approaches and methods are explained in the Future Works section of this document that will decrease the percent error and increase the accuracy of the measurements.

## VI. Conclusion

For this senior project, its intent was to easily calculate the inductance of an unknown inductor. Its purpose is to simulate a portable LC meter. Future works into this project can possibly make it ergonomic. Instead of using an entire development board, we can program additional integrated circuits that have similar inputs and output pins for detecting the comparator's output waveform. Although there were some issues in the measurement of the period of the sinusoidal waveform, those issues can easily be addressed in the Future Works section. The majority of the issue is the way in which the timer was handled. The interrupt based timer explained in the next section is much more accurate and efficient. Components are also important, however those can simply be addressed by using an external impedance measurement tool to accurately measure the capacitor used in parallel with our unknown inductor. This project sets a basis for future projects that can improve upon all the errors seen in this project. Timers can be used more efficiently than how it was implemented here and actual values should be used in the program to receive the highest precision.

## VII. Future Works

There are many things this project could improve upon. First of all, the components used in this project were of low quality. The parts were at 10% tolerance from nominal values. This would mean that the calculated inductance would have a 10% error even if the time measurement of the controller was precise.

Another thing this project could improve upon is the approach taken to start the timer. The way the program is set up right now, the timer is started and every time there is an overflow in the counter, the program will check and see if the comparator voltage is high. It will continue to check the voltage, incrementing the counter with every instance when it is high. The counter will stop after the voltage comes low and stops counting. We then use this number and multiply it by 2 and the time it takes for 1 instance to occur. A different approach can be to use the output comparator voltage as an interrupt signal. When this signal turns high, we can start the timer, and then stop it when the voltage suddenly turns low. This approach is much more accurate and precise compared to the “polling” option used in this project.

The function generator for this project was solely created by using a while loop. This works relatively well, however it can always be improved. Instead of being stuck in a while loop just to create some source for the circuit, we can use a separate interrupt and timer specifically for the function generator. This timer will work in the background simultaneously in parallel with written instructions of the program.

## Citations

“ATMEL AVR2560 INFO – CPE 329 Course Materials.” *Dr. Slivovsky*. Spring 2011

“EE212 Lecture Notes – Electric Circuit Analysis III.” *Dr. Taufik*. Spring 2009

“LC Circuit.” *Wikipedia*. 12 July 2011.

< [http://en.wikipedia.org/wiki/LC\\_circuit](http://en.wikipedia.org/wiki/LC_circuit) >



## Appendix A

### Source Code

```
//output/input and delay header files
#include<avr/io.h>
#define F_CPU 16000000
#include<util/delay.h>
#include<string.h>
#include <avr/interrupt.h>
#include<stdlib.h>
#include<math.h>

void init_lcd(); //function for initializing the LCD screen, based off of PMD CLP datasheet
void display(char* word_print); //writes a string to the LCD screen
void clr_disp(); // this function clears the display

//function prototype
void initTimer0(void);

// global variable declaration!
volatile unsigned char myCounter;
volatile unsigned int test;

int main(void)
{
    DDRB = 0xFF; //This is port connects to the data lines of the LCD
    //PB(7:0) = DB(7:0);
    DDRC = 0xFF; //This port connects to the control lines of the LCD
    //PC(7) = RS, PC(5) = R/W, PC(3) = E
    DDRD = 0x00; //This port connects to the buttons
    DDRE = 0xFF; //Function Generator
    DDRF = 0x00; //Interrupt Signal

    char first_line[16];
    char second_line[16];

    PORTC = 0x00;
    test = 2;

    double period = 0;
    double frequency = 0;
    double inductor = 0;
    double capacitor = 0.000001; //1uF Capacitor

    sei();
    init_lcd();
```

```

        initTimer0();

while(1)
{
    if(test != 1)
    {
        test++;
    }

    PORTE = 0x00;
    _delay_ms(1);
    PORTE = 0xFF;
    _delay_ms(1);

    period = 2*myCounter*16*pow(10,-6);
    frequency = 1/period;
    inductor = pow(10,6)/(4*pow(frequency,2)*pow(3.1416,2)*capacitor); //displayed in uH
    itoa(myCounter,first_line,10);
    display(first_line);
}
return 0;
//END
}

void initTimer0(void)
{
    TCCR0A = 0;           // timer normal mode
    TCCR0B = 0x03;        // timer clk = system clk / 1024
    TIFR0 = 0x01;         // clear previous timer overflow
    TIMSK0 = 0x01;        // timer compare interrupt enabled
}

ISR(TIMER0_OVF_vect)
{
    if((PINF == 0x01) && (test > 50))
    {
        myCounter++; //increment global val
    }
    else if((PINF == 0x00) && (myCounter != 0))
    {
        test = 1;
    }
}

```

```

void init_lcd()
{
    _delay_ms(50); //wait after power up
    PORTC = 0x00;

    PORTC = 0x0F; // set function
    PORTB = 0x38;
    PORTC = 0x00;

    _delay_us(100); //display set: display on, cursor off
    PORTC = 0x00;
    PORTC = 0x0F;
    PORTB = 0x0C;
    PORTC = 0x00;

    clr_disp();

    PORTC = 0x00;
    PORTC = 0x0F; //Set the display to shift cursor right
    PORTB = 0x06;
    PORTC = 0x00;
}

/* this function works by using a while loop to write each character of the input string to the
LCD screen. The LCD screen automatically increments the address, so the correct string is
displayed. */
void display(char* word_print)
{
    int w_length = strlen(word_print);
    int i;
    unsigned char letter;

    clr_disp();

    for(i = 0; i < w_length; i++)
    {
        PORTC = 0x8F;
        letter = word_print[i];
        PORTB = letter;
        PORTC = 0x80;
        _delay_us(500);
    }
}

```

```
void clr_disp()
{
    _delay_us(100); //display clear code based on PMOD CLP datasheet
    PORTC = 0x00;
    PORTC = 0x0F;
    PORTB = 0x01;
    PORTC = 0x00;
    _delay_ms(3);
}
```

## Appendix B

### Inductor Derivation

$$f = \frac{1}{2\pi\sqrt{LC}}$$

$$2\pi f = \frac{1}{\sqrt{LC}}$$

$$(2\pi f)^2 = \left(\frac{1}{\sqrt{LC}}\right)^2$$

$$4\pi^2 f^2 = \frac{1}{LC}$$

$$1 = \frac{1}{4\pi^2 f^2 LC}$$

$$L = \frac{1}{4\pi^2 f^2 C}$$

### Sample Calculations

$$L = \frac{1}{4\pi^2 f^2 C}$$

$$f = 15.337 \text{ kHz}$$

$$C = 1 \mu F$$

$$L = \frac{1}{4\pi^2 (15.337 \text{ kHz})^2 (1 \times 10^{-6})}$$

$$L = \frac{1}{9.286 \times 10^{-3}}$$

$$L = 107.686 \mu H$$

## Appendix C

### Component Legend

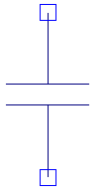


Figure 6: Capacitor

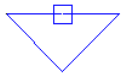


Figure 7: Ground

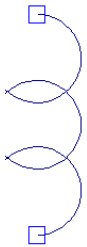


Figure 8: Inductor

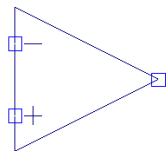


Figure 9: Comparator Operational Amplifier



Figure 10: Resistor

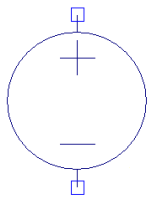


Figure 11: Voltage Source

## Appendix D: Analysis of Senior Project Design

**Project Title:** Microcontroller Based Inductance Meter

**Student:** David Nguyen: \_\_\_\_\_

**Advisor:** William Ahlgren: \_\_\_\_\_

### Summary of Functional Requirements

The goal of this project is to create an inductance meter use of a microcontroller. The user will be able to connect their inductor to the device and an LCD will display the value conveniently for the user. The board will act as the sole source of power for the detection circuit. There will be no additional external sources other than the microcontroller's function generator and the detection circuit. This project aims at building a tool that is self-sufficient so that in the future, products can be mobile and convenient to the user.

### Primary Constraints

Components are a huge part to determining the accuracy of the entire system. Components used in this project were measured within 10% tolerances; however the final measurements and calculations were approximately 15% from the expected values. Since the system had to be self-sufficient, a working function generator had to be coded to be on at all times. The function generator code had to work in parallel with the measurement sequence; this takes resources away from the controller, which then in turn may affect the timing measurements of the circuit. In addition to the issue of parallel execution, the microcontroller supplies limited current to the circuit. During the time of measurement, the circuit is not fully energized due to the maximum

allowed current of the board. Measurement could still take place, however accuracies could be improved if these limitations are removed.

## Economic

Costs started to arise after the research and design stages of the project. The development board and components had to be purchased in order to have a functional working product. Cost for the development board was approximately \$230, while the parts were \$20. Parts include various resistors, inductors for testing, and a single capacitor and comparator operational amplifier.

There was an additional \$20 budget set aside for different alternative methods to determine the inductance, however those ideas were scrapped after the interrupt-based timer method was chosen. Other than the component and development boards, no additional costs were necessary.

There were no profits tied to this senior project, however if there are any interest in commercialize this idea, there is a slight potential to obtain profits.

	<i>Original Estimated Time</i>	<i>Actual Development Time</i>
<b>Research</b>	30 Hours	45 Hours
<b>Design</b>	30 Hours	35 Hours
<b>Programming</b>	40 Hours	50 Hours
<b>Debugging</b>	10 Hours	30 Hours
<b>Total</b>	110 Hours	160 Hours



## If Manufactured on a Commercial Basis

- ~1000 units will be sold per year
- Each device would be approximately \$10 to manufacture and produce
- Each device would be sold for approximately \$20 in order to cover the design and manufacturing costs
- Estimated profits at the end of the year would be approximately \$10,000
- User would need 2 AA batteries to power the circuit, so costs would cost around \$10 to operate annually

## Environmental

If future products were to be built, then the most damage done to the environment would be through the use of batteries. Manufacturing and putting the parts together would not have a greater effect on the environment because users will tend to use regular alkaline batteries instead of the environmentally friendly rechargeable batteries. The only environment that might be affected would more than likely be the landfills because battery consumption would increase as a result of producing this product.

## Manufacturability

The design of this product would need to be improved upon before proceeding to the manufacture stage. The code for the input and output of the system needs to be ported to a smaller integrated chip. Other than the porting of the code, there shouldn't be any more issues that arise to the development of this product.

## Sustainability

If this product becomes mobile, the sales and consumption of batteries will increase. It is up to the consumer to determine the sustainability of this product. Whether or not they choose to use rechargeable batteries or regular alkaline batteries will determine how much of an impact it will have on the environment. The atmosphere and landfills will more than likely take a more drastic hit because of the harm production does to the atmosphere and batteries do not degrade in a single lifetime.

## Ethical

There should not be any ethical issues that arise in the use or misuse of this device. The system uses a low voltage (5V) system that is not enough to do harm to others. There will only be a single place for the user to place their unknown inductor. At these open points, the voltage is not high enough to even shock a person. The only issue with manufacturing a product like this will result in the degradation of the atmosphere.

## Health and Safety

The only way for a person to be harmed with this product is when they physically open this product apart and accidentally consume the components. Once again, the atmosphere and landfill are the most harmed during the production of this device. There should be little to no harm done to a single human being by using this product improperly.

## Social and Political

There shouldn't be many social or political issues that arise with the manufacturing of this product. Since there is only an estimated 1000 units sold initially, there isn't going to be much of an impact on the environment for people to raise awareness.

## Development

I was able to combine what I have learned in separate classes and combine them together in a single project. In doing so, I was able to learn and relearn tricks while I programmed the timing sequence. I was able to use the development board to its extent and code more efficiently than before. Since correct timing is crucial to the accuracy of the measurement, instructions sent by the controller had to be maximized efficiently to have the least amount of errors.