

Electrical & Electronic Principles Report

Stanley Vass – 2306978 – ACDA4006

Contents

Table of Figures.....	2
Abbreviations.....	2
Abstract	3
Introduction.....	4
The LC Tank	4
LC Tank Applications.....	4
LC Tank Theory	5
Simulation	6
Method.....	6
Results.....	7
Discussion.....	8
Testing	8
Method.....	8
Results.....	9
Discussion.....	9
Parasitic Effects.....	11
Parasitic Effects Introduction.....	12
Parasitic Effects Methodology	12
Parasitic Effects Results.....	13
Parasitic Effects Discussion	14
Parasitic Effects Conclusion.....	14
Conclusion	15
References	16
Appendices.....	17
Appendix A: Python code for calculating Q-factor from a .csv file	17
Appendix B: Real Circuit Performance Data.....	19

Table of Figures

Figure 1:	6
Figure 2:	7
Figure 3:	7
Figure 4:	9
Figure 5:	10
Figure 6:	10
Figure 7:	11
Figure 8:	12
Figure 9:	13
Figure 10:	14

Abbreviations

Q-factor	Quality Factor
ESR	Equivalent Series Resistance
EPR	Equivalent Parallel Resistance
EPC	Equivalent Parallel Capacitance
ESL	Equivalent Series Inductance

Abstract

The goal of this report is to perform research into LC tank circuits and their simulated behaviour vs real-world behaviour.

To this end the theory behind LC tanks will be discussed. Then an LC tank circuit will be simulated in Proteus, a schematic capture and circuit simulation suite. Once the simulated behaviour has been observed and correlated with predicted theoretical behaviour, the circuit will be built and its performance measured, the results of this being compared to the simulation.

Discrepancies in the behaviours of the circuits will then be identified and alterations made to the simulation, correcting for the aforementioned discrepancies. Then a discussion on which real world parasitic effects affect the greatest change in circuit performance, and thus are most important to model.

The results obtained from the course of this research show that real world performance and simulated performance can often deviate quite significantly. The results also suggest that the largest parasitic factor affecting simulated circuit performance is the inductor's ESR value and that, for the component values tested, the other effects offer only marginal changes in performance.

Introduction

An LC Tank circuits is an arrangement of inductor and capacitor in parallel. Which when exposed to an AC signal will respond with frequency variable impedance.

Analysing the exact behaviour of this variability, and how real components as opposed to ideal components alters it is the principle goal of this report.

To this end an LC tank circuit will first be constructed in Proteus and its response measured over a range of frequencies.

Then the performance of this simulated circuit will be compared to a real example of an LC tank circuit built on breadboard and the differences analysed.

Finally, potential corrections to the simulation model will be discussed and tested, identifying the largest affecting factor.

The LC Tank

The LC tank's frequency response is due to the opposing actions of inductive and capacitive reactance. At most frequency input values, either the inductor or the capacitor is the dominant circuit element, offering little impedance and allowing significant currents to flow.

However, at the frequency where both the inductor and capacitor exhibit the same reactance, known as the resonant frequency, the overall reactance is zero and the circuit as a whole will act as an open circuit. Permitting no current to flow and exhibiting its minimum attenuation of the input AC signal.

LC Tank Applications

The most prominent application historically for LC tank circuits has been within AM radio receivers [1]. This is because in an AM radio transmission, carrier frequency is not variable. So, if the LC circuit is tuned to have its resonant frequency coincide with this carrier frequency, then the LC circuit can be used as an extremely narrow bandwidth bandpass filter, filtering out nearby frequency bands/channels. In such an application LC tank circuits are typically called a tuned circuit. Tuned Circuits are typically constructed with a variable inductor or capacitor, allowing the operator to tune the circuit, and thus receiver, to different frequencies/channels. [2]

LC tank circuits are still used today in non-AM applications but are often referred to as resonant filters, which are useful as narrow-band bandpass filters, or as similarly narrow-band notch filters. [1]

LC tank circuit can also be used for measuring inductance, this is done by using a known value of capacitor and measuring the resonant frequency of the LC tank formed. [3] [4]

LC Tank Theory

Ideal LC tank circuits can be described quite completely with only a few relatively simple equations:

$$X_L = 2\pi fL$$

Equation 1: Inductive Reactance

$$X_C = \frac{1}{2\pi fC}$$

Equation 2: Capacitive Reactance

$$X_t = \frac{1}{\frac{1}{X_L} - \frac{1}{X_C}}$$

Equation 3: Parallel Reactance for Ideal LC Tank

Note: Equation 3 will work in all cases except when at resonance, as X_L and X_C are equal at resonance causing the bottom half of the fraction to evaluate to 0 leaving $\frac{1}{0}$, which is undefined.

By combining these three equations most aspects of the tank circuit can be derived, for example the resonant frequency can be given by:

$$2\pi fL = \frac{1}{2\pi fC}$$

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

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

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

Equation 4: LC Tank Resonant Frequency

However, the Q-factor of the circuit cannot be easily derived from these simple equations.

Q-factor is a property of resonant circuit which describes the frequency response of the circuit. It considers the resonant frequency and the gradient of the response to surrounding frequencies, as such it can be an extremely useful tool in resonant filter design.

The following equation is typically used to calculate Q-factor:

$$Q = \frac{f_r}{f_2 - f_1}$$

Equation 5: Q-factor equation

f_1 in this case represents the frequency below resonance at which the gain of the LC tank is 3dB below the gain at resonance.

f_2 is much the same except that it is the -3dB frequency above the resonant frequency.

Important note when applying these equations - These equations are only accurate for ideal components, once this circuit is translated into reality parasitic effects can significantly alter the performance of the circuit.

Some examples of these parasitic effects include:

- Parasitic EPC
- Parasitic EPR
- Parasitic ESL
- Parasitic ESR
- Parasitic Inductance of PCB traces
- Parasitic Capacitance of PCB traces
- Oscilloscope Probe Capacitance
- Oscilloscope Probe Termination Resistance
- Component value inaccuracies

Some of these parasitic effects may affect only a negligible change in circuit performance. However, some effects, for instance the ESR of the inductor, may to a rather large degree, alter the circuits properties.

Simulation

Method

In order to simulate the LC tank circuit under test, it was drawn in Proteus with an additional resistor R1. This was added to limit the flow of current through the tank circuit, causing the varying impedance of the LC tank to be seen as a variation in peak to peak voltage at C1(2).

The input at R1(2) was set to 2V peak to peak.

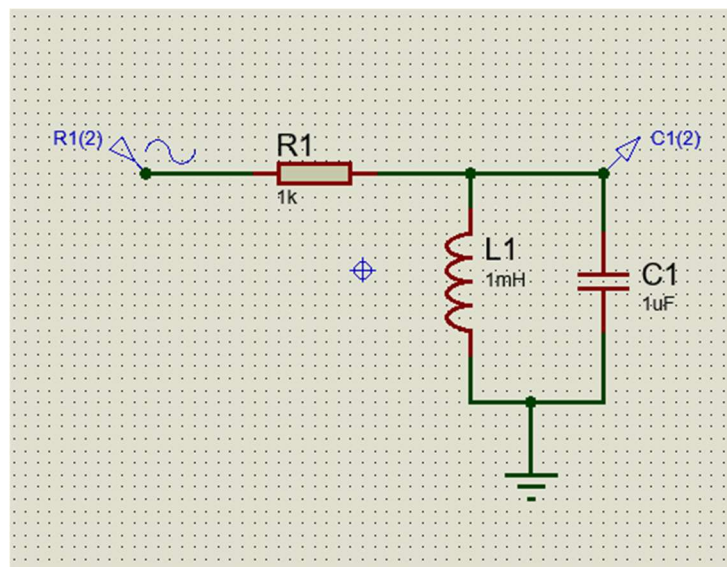


Figure 1: The Circuit

A frequency response graph was then added to the Proteus drawing. The Frequency Response graph is a feature of Proteus which iterates the frequency generated at R1(2), then measures the gain on that signal at C1(2).

The results of this simulation can then be exported to a standard CSV file and plotted in Microsoft Excel.

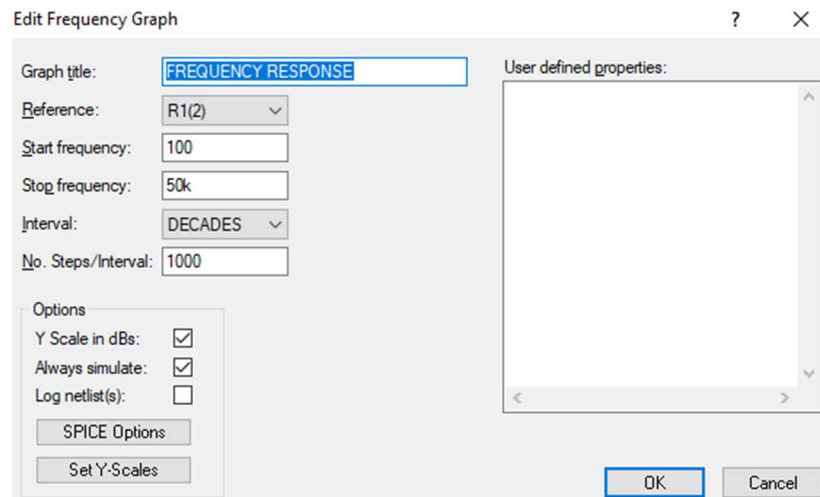


Figure 2: Frequency Response Sweep Configuration

Results

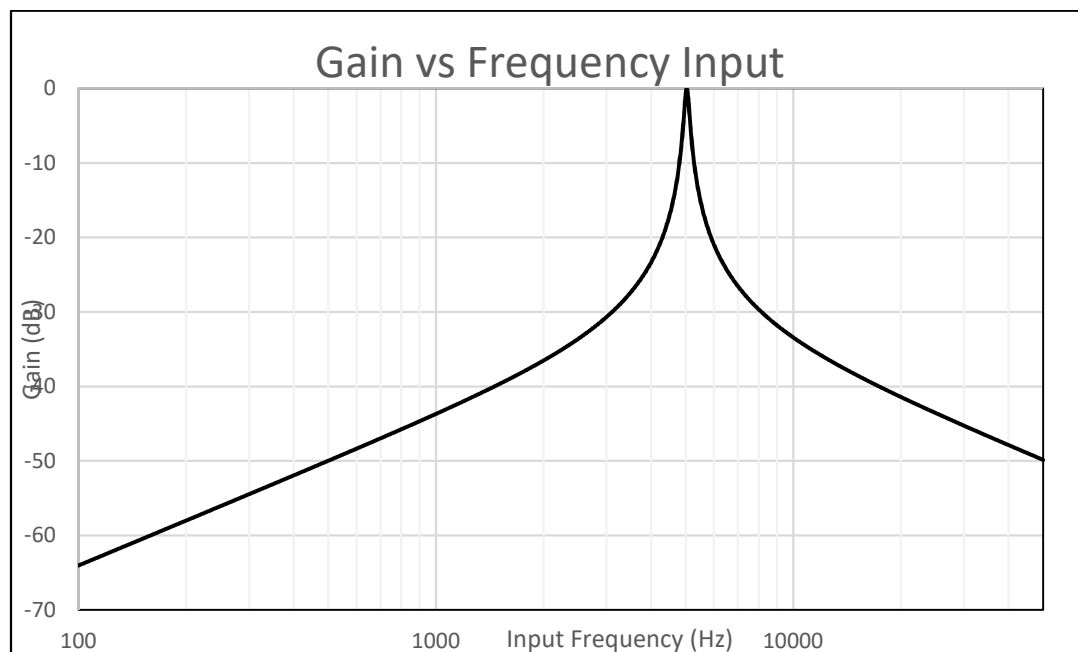


Figure 3: Graph of Simulation Results

Figure 3 shows how the gain at C1(2) rapidly drops off as the input frequency deviates from the resonant frequency of the LC tank circuit.

Discussion

Figure 3 shows a frequency response graph that matches with what the theory predicts.

This can be shown to be true by calculating the LC tank's impedance and treating it and R1 as a potential divider. This potential divider's output can then be compared with the output from the simulation.

At 100Hz Equations 1, 2 and 3 predict an impedance of approximately 629mΩ.

Note: As this is an ideal circuit with ideal components no resistive element is present within the tank circuit, this means that impedance can be considered to be equal to reactance; For any scenario featuring a resistive element, this shortcut will not work.

$$V_{out} = V_{in} \frac{R_2}{R_1 + R_2}$$

Equation 6: Potential Divider Output Voltage

Using Equation 6 the predicted output is approximately 1.256mVp-p, or when converted into dB of gain relative to 2Vp-p input a result of -64.04 dB is obtained, which almost exactly matches the recorded -64.03dB.

Repeating the calculation again for 10kHz an impedance of 21.31Ω is reached producing a voltage of 41.73mVp-p or a gain of -33.61dB again very closely matching the -33.43dB recorded by the simulation.

In both the 100Hz and 10kHz cases the small difference in results is most likely to be the result of cumulative rounding errors when calculating the gain from the theoretical equations.

This simulated circuit exhibits a Q-factor of 31.69 which is another indicator of the sharp frequency response / narrow bandwidth of a resonant circuit. This value was calculated using the python code in Appendix A.

Testing

Method

The circuit was first built on a breadboard and connected to a signal generator so as to be the equivalent of the signal input at R1(2) in Figure 1. This signal generator was configured to output at 2V peak to peak, sine wave at 100 Hz.

Then an oscilloscope was connected at the same position as C1(2) in Figure 1 as well as at the signal generator's output to allow for corrections to be made for potential signal generator instabilities. The oscilloscope was then configured to measure the peak to peak voltage of both connected channels.

Once this setup had been constructed the output frequency of the signal generator was incremented in logarithmic steps from 100Hz through to 20kHz recording both the input to the circuit and the output of the circuit into a spreadsheet.

Results

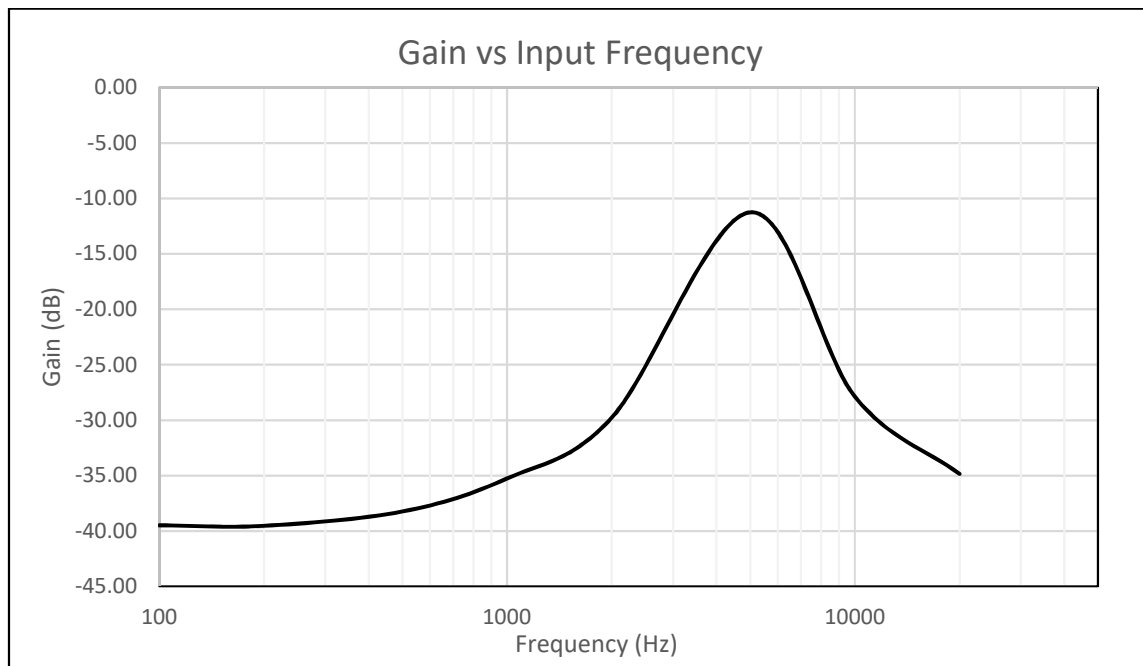


Figure 4: Test Results

These results show that the behaviour of the circuit does broadly match the behaviour of the simulated circuit, but with some significant differences; Notably the reduced Q-Factor and the lower frequency region plateauing towards -40dB.

Discussion

This difference in behaviour can most likely be attributed to a combination of parasitic effects on the circuit and component value variations. However, in this instance LC variance can be mostly ruled-out as a potential source of this discrepancy; As if the capacitor or inductor had different values, then in line with equation 4 the resonant frequency should change and this change is not reflected in the data.

This can be evidenced by how the simulated circuits behaviour can be brought in line with the measured results.

This was done by adding R2 which represents the internal inductor resistance, and by systematically editing every component value until the behaviour closely matched the collected data.

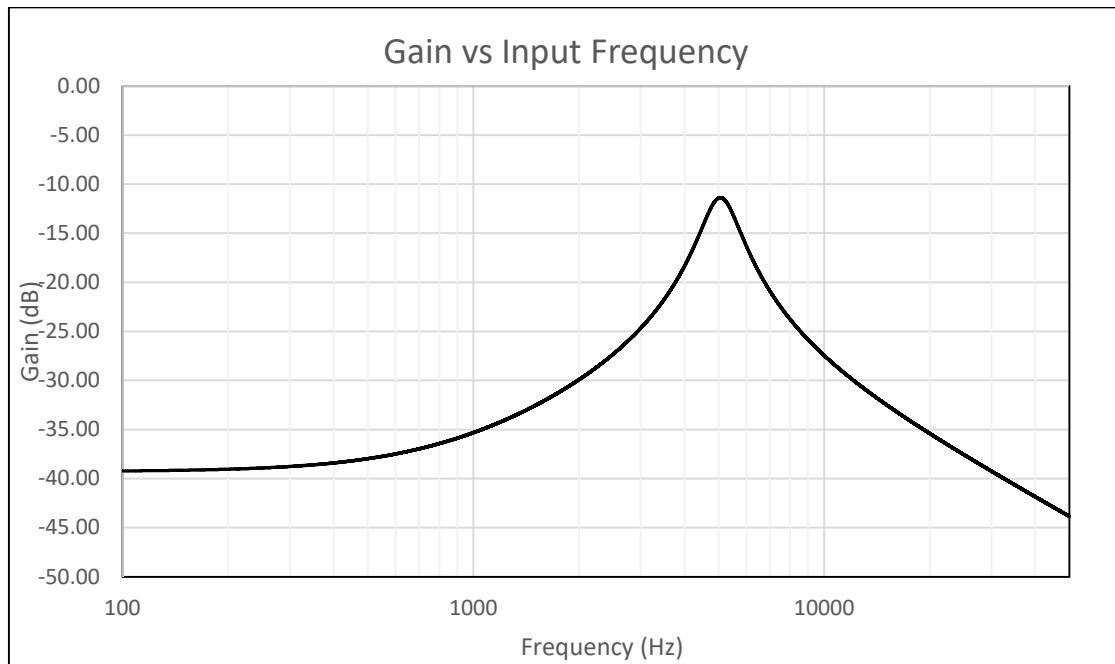


Figure 5: Corrected Simulation

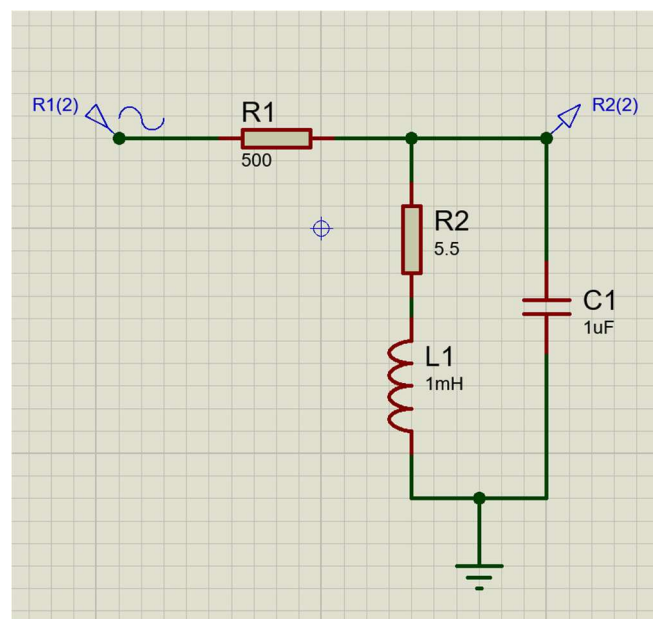


Figure 6: Corrected Circuit

This corrected circuit which has a response curve within $\pm 0.5\text{dB}$ of all measured points on the real circuit, implies that R1 may have been faulty and had much less resistance than indicated by the markings present.

The 5.5Ω R2 was added to approximate the real-life internal resistance of the inductor (ESR). This resistor has the effect of providing a lower limit to the inductor's impedance, raising the gain of the circuit where $f_{in} < f_{resonant}$, the region where the inductor is the dominant circuit element.

If inductor's internal resistance is indeed 5.5Ω then the minimum gain can be estimated by treating the inductor as if it has 0 impedance. At this point the total impedance of L1 and R2 will be equal to R2. R1 and R2 can then be treated as a potential divider producing a gain of:

$$20 * \log_{10}\left(\frac{5.5}{500 + 5.5}\right) = -39.267dB$$

Which matches the collected data trending towards approximately -39.5dB.

The corrected simulation also approximately matches the Q-factor of the collected data with a value of 4.25 compared to the 3.60 of the collected data. As before this was calculated using the python code in Appendix A.

Parasitic Effects

The corrected circuit and graph in figures 5 and 6 only account for the effect of the inductors ESR. Yet there are many more discontinuities between ideal components and their real counterparts.

In order to determine the impact of these effects, an analysis into the change affected by the introduction of parasitic effects to the model was undertaken.

In order to facilitate the measurement of the influence of the different parasitic effects of real-world components, the ideal simulation was redone with a corrected R1 value of 5000Ω . This data will then be used as a baseline to measure the magnitude of change affected by different parasitic phenomena. The results of this simulation are shown in figure 7.

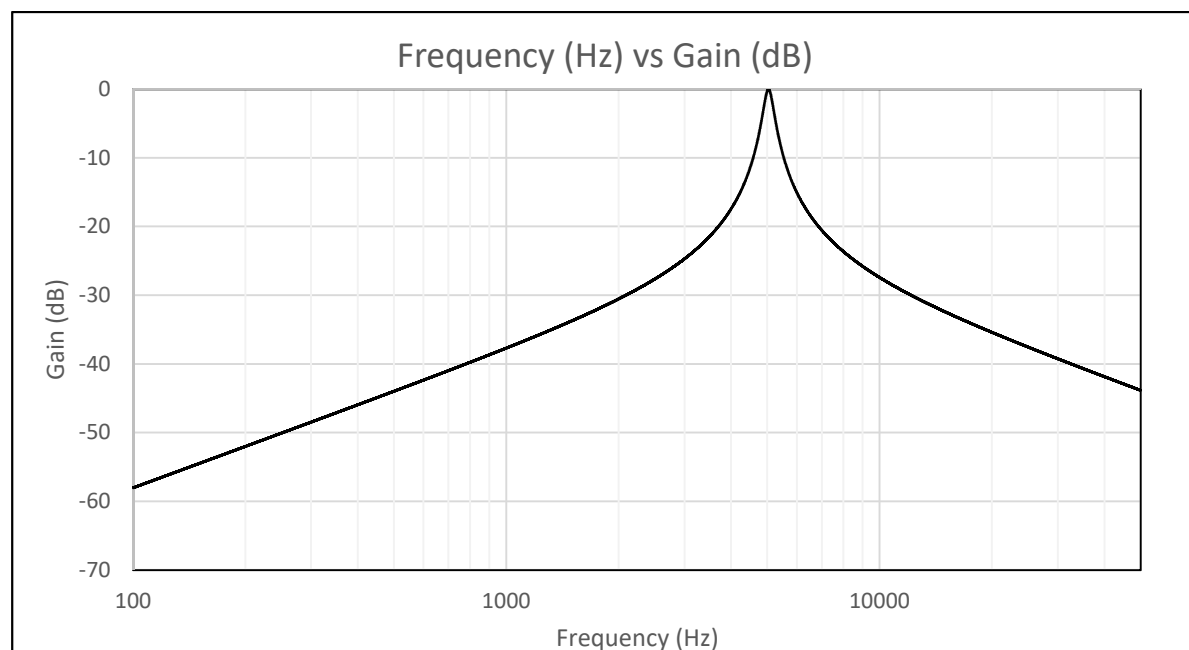


Figure 7: Corrected Ideal Simulation

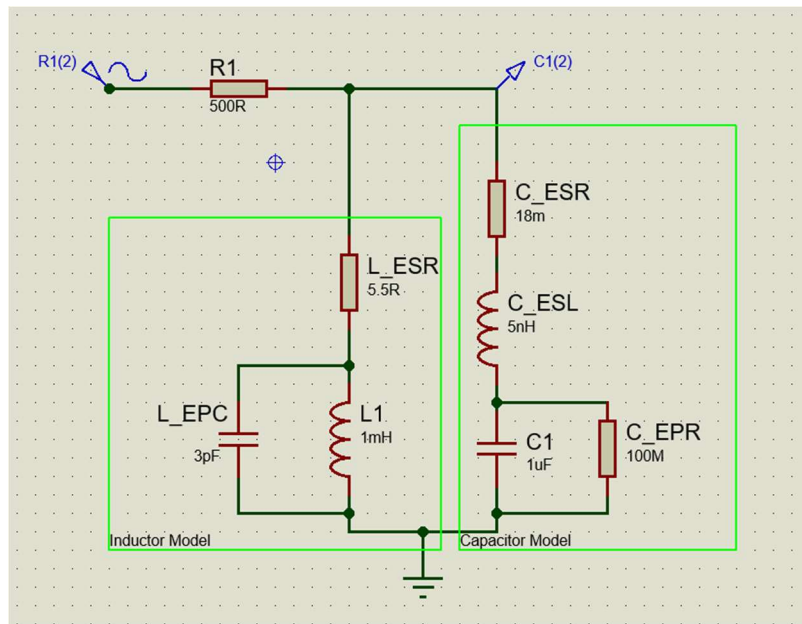


Figure 8: More Accurate Model

Parasitic Effects Introduction

The goal was to determine the magnitude of behavioural change affected by modelling parasitic influences on the circuit. With the hypothesis that the inductor's ESR would exhibit, by a wide margin, the most dominant effect on circuit performance.

A parasitic effect is one which is not necessarily intended by the designer, however is present in the circuit regardless.

For example, a PCB designer running two tracks in parallel may have no intention of capacitively and inductively coupling the two traces together. Yet it will occur to some degree regardless, and under the correct circumstances may massively alter the design's performance.

Parasitic Effects Methodology

In order to determine whether the Inductor's ESR does indeed exhibit the largest influence on performance, the circuit was first simulated as seen in Figure 6. Then simulated again with the additional parasitic effects, represented as additional ideal components.

The data produced from just adding L_ESR was compared to Figure 7 by finding the difference in gain at every frequency point.

The same was then repeated between the data including L_ESR and the data generated with all the parasitic effects modelled, so as to identify any additional change.

Inductor Model

The inductor's EPC and ESR effects were modelled, EPR was not modelled due to it not being commonly provided information by manufacturers as measuring EPR is extremely difficult due to the behaviour of an inductor in relation to DC currents [5].

The L_ESR value was chosen to closely match the simulation's behaviour to that of reality. This element is representative of the resistance inherent to the wire composing the inductor.

L_EPC is a representation of the capacitance between neighbouring windings in the inductor's coil. Its value was chosen so as to be simply a reasonable value [6], as, if the hypothesis that it would have little to no effect is true then the precise value should not matter and merely being within the same order of magnitude should be sufficient.

Capacitor Model Value Selection

C_EPR approximates the resistance across the dielectric separating the two conductive elements of the capacitor. The value was again chosen simply to be within the realm of plausibility [7].

C_ESL is a model for the inductance introduced by the legs of the through hole capacitor used. The value was chosen assuming a 5mm lead length on the capacitor and a rule of thumb of ~1nH of inductance per 1mm of lead length [7].

C_ESR is a representation of the resistance of the legs of the capacitor. Its value was selected as it was within the range of ESR values returned by a search on mouser for an equivalent capacitor.

Parasitic Effects Results

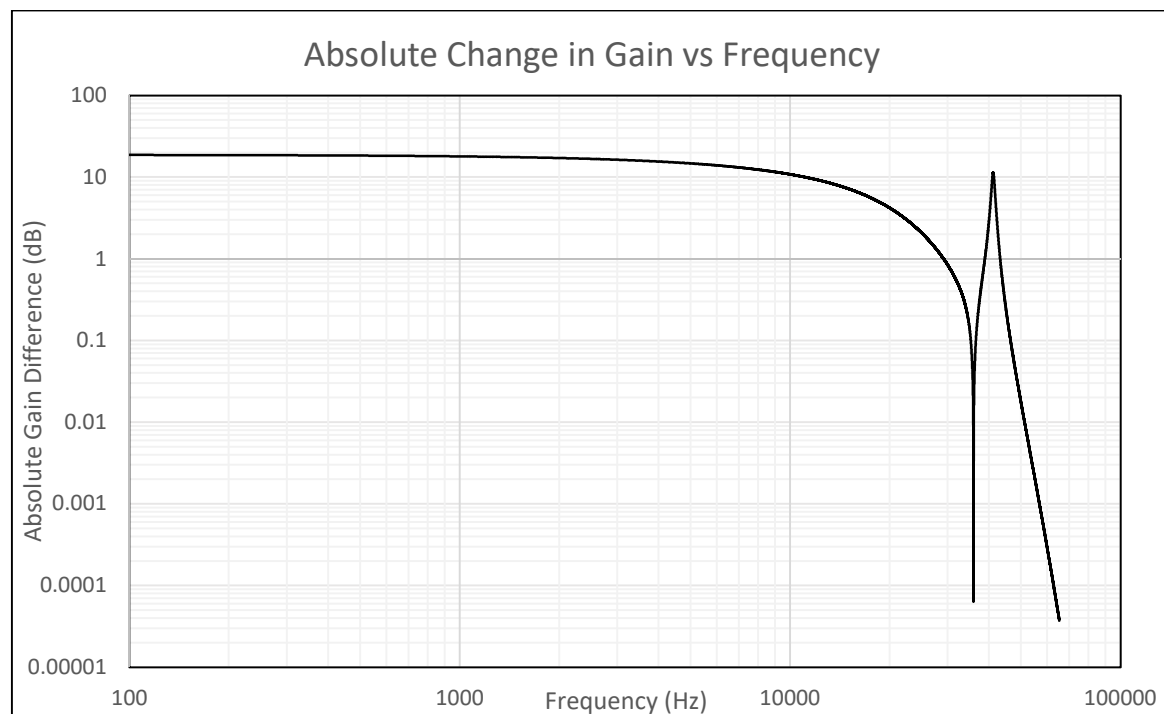


Figure 9: Absolute Change in Gain from L_ESR

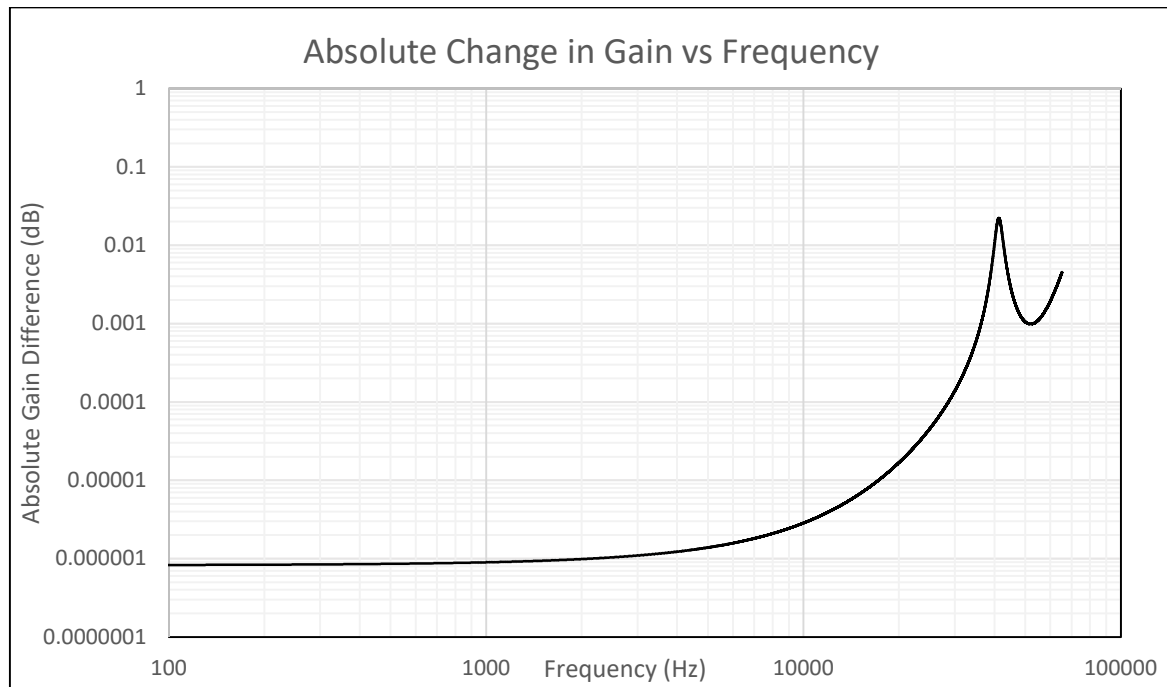


Figure 10: Absolute Change in Gain from Additional Effects

These results in Figure 9 illustrate that the change in behaviour from simulating the effect of L_{ESR} is quite significant resulting in a change in gain of nearly 11dB, across the majority of the measured frequency range.

Whereas, simulating the additional parasitic effects mentioned prior at most induces a change in gain of 0.02dB, a difference of 3 orders of magnitude.

Parasitic Effects Discussion

These results are likely to be a result of the extreme magnitude of the parasitic effects, for almost every effect except the inductor's ESR, the value of the equivalent ideal component is either extremely large or extremely small, to such a point that they influence almost zero actual change to circuit performance for the component values being tested.

For smaller principle component values, the parasitic effects may represent a larger portion of the overall circuit properties, and thus command a larger influence over performance. But for the circuit being tested this does not appear to be the case.

However, Figure 10 does suggest that some of the parasitic effects added after L_{ESR} may have an increased influence at higher frequencies above resonance. Although further testing would be needed to confirm or deny this hypothesis.

Parasitic Effects Conclusion

The extreme difference in the magnitude of change affected by these parasitic effects, in the frequency range tested, provides adequate justification for simply ignoring the majority of these effects and modelling only the inductor's ESR.

Conclusion

In this report it has been explored how real-world effects and properties can influence and alter the behaviour of resonant circuits.

Through the analysis done in Figures 9 and 10 it has been shown that, for the components values and frequencies tested, the ESR of the inductor exhibits the largest influence on circuit performance as compared with other parasitic effects. It should be noted that parasitic phenomena may have increased affect at higher frequencies, as suggested by Figure 10. However, more testing is needed to confirm or deny this.

Variations in component principle values may also provide a cause for alterations in circuit performance, as shown through R1 which by comparing Figures 3 and 7 shows that this variation can exhibit a significant change in circuit performance.

References

- [1] P. Horowitz and W. Hill, *The Art of Electronics*, 3rd ed., New York: Cambridge University Press, 2015.
- [2] P. Rufus P.Turner, "Range Coverage," in *LC Circuits*, Indianapolis, Howard W.Sams & Co.,Inc., 1980, pp. 51-53.
- [3] P. RufusP.Turner, "Resonant Circuit as Measuring Device," in *LC Circuits*, Indianapolis, Howard W.Sams & Co.,Inc., 1980, pp. 84-85.
- [4] Texas Instruments, "Configuring Inductive-to-Digital-Converters for Parallel Resistance (RP) Variation in L-C Tank Sensors," April 2015. [Online]. Available: <https://www.ti.com/lit/an/snaa221b/snaa221b.pdf?ts=1714225538973>. [Accessed 27 April 2024].
- [5] Coilcraft, "Working Voltage Ratings Applied to Inductors," 25 July 2008. [Online]. Available: https://www.coilcraft.com/getmedia/393e18e1-adbc-45b6-bbe7-5923255e72fc/doc712_inductor_voltage_ratings.pdf. [Accessed 23 April 2024].
- [6] J. A. Woody, "Modeling of Parasitic Effects in Discrete Passive Components," ROME AIR DEVELOPMENT CENTER Air Force Systems Command Griffiss Air Force Base, NY 13441, New York, February 1983.
- [7] Vishay Roederstein, "General Technical Information - Film Capacitors," 24 March 2022. [Online]. Available: <https://www.vishay.com/docs/26033/gentechinfofilm.pdf>. [Accessed 23 April 2024].

Appendices

Appendix A: Python code for calculating Q-factor from a .csv file

```
import sys
import csv

#being completely honest this is some hacky code, should probably not be solely
# relied upon

#pretty self explanatory finds the highest value within an array
def peak_search(data_arr, index):
    max = float(data_arr[0][index])
    max_i = 0
    for i in range(len(data_arr)):
        if float(data_arr[i][index]) > max:
            max = float(data_arr[i][index])
            max_i = i

    return (max, max_i)

#finds the nearest 2 values to a target value
def value_neighbor_search(data_arr, index, target_val):
    #find highest value below the target
    max = target_val - 9999999999 #arbitrary low number
    max_j = -1
    for i in range(len(data_arr)):
        if float(data_arr[i][index]) > max and float(data_arr[i][index]) < target_val:
            max = float(data_arr[i][index])
            max_j = i

    hn = 0
    ln = max
    hi = 0
    li = max_j

    if float(data_arr[max_j+1][index]) > target_val:
        hn = float(data_arr[max_j+1][index])
        hi = max_j+1
    else:
        hn = float(data_arr[max_j-1][index])
        hi = max_j-1

    return (ln, hn, li, hi)

#open the input .csv file for reading
file = sys.argv[1]
data = []
with open(file, "r") as f:
    reader = csv.reader(f)
    data = list(reader)

#find the peak gain
(max, max_i) = peak_search(data[1:-1], 1)

#getting the low and high neighbor values to the -3dB points
target = max - 3
max_i = max_i

(lln, lhn, lli, lhi) = value_neighbor_search(data[1:max_i+2], 1, target)

(cln, chn, cli, chi) = value_neighbor_search(data[max_i:len(data)+1], 1, target)

#interpolate between the points
#low freq side -3dB point
```

```

lowlf = float(data[l1i+1][0])
lowhf = float(data[lhi+1][0])

#get the total frequency difference
frange = lowhf - lowlf

#get the total gain difference
grange = abs(abs(lhn) - abs(lln))

#get the delta between the low neighbor and the target gain
gtdelta = abs(abs(target) - abs(lln))

#linear interpolation
lowfreq = (gtdelta / grange) * frange + lowlf

#high freq side -3dB point
#same as low frequency code, should really be a function for neatness
#but doesn't really matter it not being a function
highlf = float(data[cli+max_i][0])
highhf = float(data[chi+max_i][0])
frange = highhf - highlf
grange = abs(abs(chn) - abs(cln))
gtdelta = abs(abs(target) - abs(cln))
highfreq = (gtdelta / grange) * frange + highlf

#qfactor = peak_freq / bandwidth
qfactor = float(data[max_i+1][0]) / abs( highfreq - lowfreq )

#output data
print("Q-factor:          " + str(qfactor))
print("Resonant Frequency: " + str(data[max_i+1][0]))
print("Peak Gain:          " + str(max))
print("High -3dB Intersect: " + str(highfreq))
print("Low -3dB Intersect:  " + str(lowfreq))

#some systems run python in a seperate window, in which case this is needed as
# otherwise said window appears, and immediately closes leaving no time to
# see the data
input()

```

Appendix B: Real Circuit Performance Data

F_{INPUT} (HZ)	V_{P-P} (V)	V_{IN} (V)	GAIN (DB)
100	2.10E-02	1.98	-39.49
200	2.10E-02	1.99	-39.53
500	2.40E-02	1.96	-38.24
1000	3.40E-02	1.97	-35.26
2000	6.40E-02	1.97	-29.77
5000	5.37E-01	1.96	-11.25
10000	8.00E-02	1.98	-27.87
20000	3.60E-02	1.99	-34.85

Table A1