Modeling Skin Effect in Inductors

Ken Kundert

Version 1a, October 2001

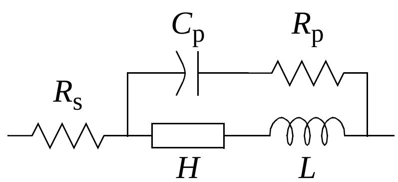
Presents a technique for modeling skin effect losses in inductors. Includes a model written in Verilog-A as well as a model based on a new type of basic element named a *fracpole*.

This manuscript was originally written in October 2001. It was last updated on January 1, 2004. You can find the most recent version at www.designers-guide.com. Contact the author via e-mail at ken@designers-guide.com.

1.0 Background

In the Coilcraft application note on modeling RF inductors [2], the inductor is modeled as shown in Figure 1. Component H in the model represents the skin-effect loss. In the ap-note, it is modeled in two different ways. In the PSPICE version, it is written as

FIGURE 1. *RF inductor model.*



$$Z(f) = \frac{V(f)}{I(f)} = \frac{\sqrt{1+jf}}{H}. \tag{1}$$

For other simulation software, the model is written as

$$Z(f) = k\sqrt{f}. \tag{2}$$

These two models are not consistent in two ways. First, (1) has a corner at 1 Hz whereas (2) does not. Second, $Z(f)$ in (1) is complex while in (2) it is real. Once significantly above 1 Hz the real and imaginary are equal for the PSPICE model. Thus, for measurements made well above 1 Hz, k and H are related by

$$k = \frac{1}{\sqrt{2}H} \tag{3}$$

Having a purely real impedance that varies with $\sqrt{f}$ is not physically realizable (it would be non-causal, as shown in the appendix) and so is clearly incorrect. However, the corner frequency at 1 Hz in the PSPICE model seems artificial. I expect that it was included to prevent the impedance from going to zero at DC, which causes problems for PSPICE.

Instead of (1) and (2), skin effect is modeled here in a more traditional way,

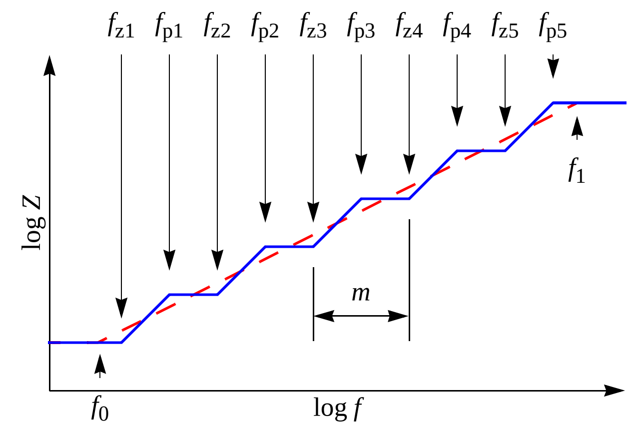
$$Z(f) = \frac{\sqrt{jf}}{H}. \tag{4}$$

This model involves a non-integer power for f and so is a distributed model [1]. In other words, $Z(f)$ cannot be exactly represented using a finite number of poles and zeros and the model cannot be implemented exactly by combining a finite number of lumped components (resistors, capacitors, and inductors).

2.0 Modeling Skin Effect

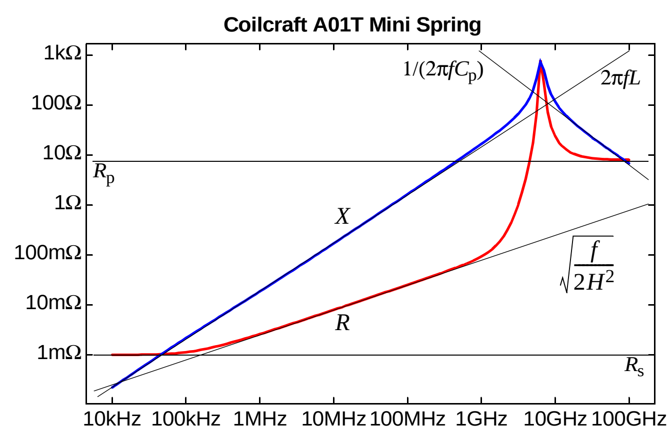
A transfer function of $\sqrt{jf}$ is approximated over a finite range of frequencies with an equal number of real poles and zeros alternating and evenly spaced in a logarithmic sense over that range, as shown in Figure 2. The range of the approximation is from f_0 to

FIGURE 2. *Modeling the $\sqrt{jf}$ nature of skin effect using a collection of real poles and zeros.*



f_1, with the impedance flattening out at frequencies outside of this range. The range of frequencies over which skin effect must be accurately modeled can be determined by examining its contribution to the overall impedance of the inductor. The impedance of a representative inductor is shown in Figure 3. Notice that the impedance is separated into

FIGURE 3. *Modeled impedance of an RF inductor separated into real (R) and imaginary (X) parts.*

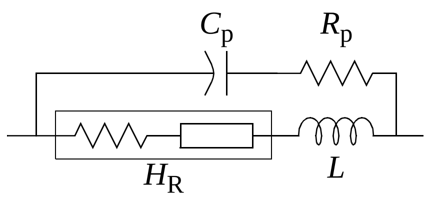


its real and imaginary parts (the resistance and the reactance). The resistance represents the loss. It is important to accurately model the resistive portion in the operating frequency range even where the reactive portion is much larger because it determines the Q of resonators and because designer sometimes use tuning to cancel out the reactive portion.

Because we do not know the operating frequency range a priori, we must model the skin effect over the range of frequencies where it is significant. The low frequency bound, f_0, is chosen to be the frequency where the impedance of the skin effect dominated over R_s.

Since our lumped approximation of skin effect naturally flattens out at low frequency, R_s will be combined with H to form H_R as shown in Figure 4. Notice that the structure

FIGURE 4. *Modified RF inductor model.*



of the model was modified slightly to accommodate the combining of R_s and H. Since R_p is always much larger than R_s, the difference is not expected to be significant.

The low frequency bound, f_0, is the frequency where the resistance (the real part) of the skin effect begins to dominate over R_s.

$$f_0 = 2(R_s H)^2 \tag{5}$$

The high frequency bound, f_1, is chosen to be the resonant frequency of LC_p, because above this frequency the capacitive path through the inductor dominate over the inductive path, and so R_p dominates over H.

$$f_1 = \frac{1}{2\pi\sqrt{LC_p}} \tag{6}$$

Placing f_1 at the resonant frequency L and C_p does act to increase the modeled Q of this parasitic resonance, but does not significantly affect the Q in the operating frequencies of the inductor.

The number of lumps, n, is chosen to provide a sufficient level of accuracy. The larger n, the closer the impedance of the lumped approximation matches $\sqrt{jf}$. Typically, using 1½ lumps per decade of frequency range give a good fit (but 1 lump per decade is often sufficient).

$$n = \left\lceil \frac{3}{2}\log\frac{f_1}{f_0} \right\rceil \tag{7}$$

The frequency span of one lump is then

$$m = \sqrt[n]{\frac{f_1}{f_0}} \tag{8}$$

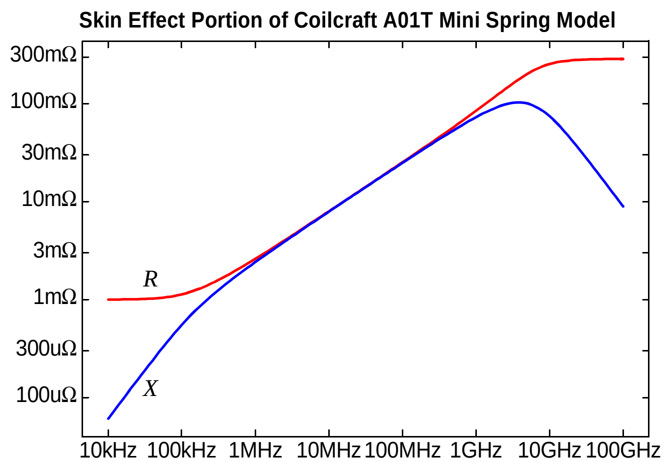
From Figure 2, the zero and pole frequencies are simply

$$f_{zi} = m^{i-\frac{3}{4}} f_0 \tag{9}$$

$$f_{pi} = m^{i-\frac{1}{4}} f_0 \tag{10}$$

with i ranging from 1 to n. For the inductor shown in Figure 3, impedance of the model for just the skin effect is shown in Figure 5. In this example, f_0 = 25 kHz and

FIGURE 5. *Impedance of skin effect model.*



f_1 = 65 GHz. It is clear that accuracy degrades at the ends of the range. If accuracy at the ends is important, consider lowering f_0 and/or raising f_1 by and order of magnitude.

3.0 Verilog-A Inductor Model

A model is derived assuming that the Coilcraft parameter values are available. If they are not, one can extract them graphically from measured data as illustrated in Figure 3. The model of the inductor itself implements Figure 4 and is given in the form of a Verilog-A netlist in Listing 1. It uses (5) and (6) to compute f_0 and f_1 for the skin effect model. The parameter *esr* specifies R_s and *rcp* specifies R_p. Both H or k are parameters, even though they specify the same thing. Just give one. Finally, the netlist used to test the model is given in Listing 2.

Listing 3 shows the module that implements the skin effect model. It is based on (7), (8), (9), and (10). The model starts by computing the location of the poles and zeros needed to achieve the transfer function shown in Figure 2, and then implements the transfer function using the *laplace_zp* function.

3.1 Results

Figure 6 shows the impedance exhibited by the model compared with measured data. Both the model parameters and the measured *s*-parameter data are provided by Coilcraft [2]. A large part of the discrepancy results either from a poor choice of model parameters or from poor *s*-parameter measurement technique (the odd shape of the *s*-parameter data suggests that it is of questionable accuracy). For example, the *s*-parameter data suggests R_s = 80 mΩ, whereas Coilcraft specifies it as 1 mΩ. They may have provided R_s = 1 mΩ because DC measurements show this to be the value and they trust DC measurements more than *s*-parameter measurements at low frequencies and impedance levels

LISTING 1. *Verilog-A model of the inductor.*

```
‘include “discipline.h”
‘include”constants.h”

module lossy_ind(t1, t2);

electrical t1, t2;
parameter real esr=0.01 from (0:inf);
parameter real rcp=1.0 from (0:inf);
parameter real cp=10f from (0:inf);
parameter real l=1n from (0:inf);
parameter real k=1e–5 from (0:1);
parameter real h=1/(1.414∗k) from (1:inf);
electrical n1, n2;

    resistor #(.r(rcp)) Rp (n2, t2);
    skin_effect #(.r0(esr), .f0(2∗esr∗esr∗h∗h), .f1(1/(2∗‘M_PI∗sqrt(l∗cp)))) Hr (t1, n1);
    capacitor #(.c(cp)) Cp (t1, n2);
    inductor #(.l(l)) L (n1, t2);
endmodule
```

LISTING 2. *Spectre netlist used to test the inductor.*

```
// test bench for inductor model with skin-effect
simulator lang=spectre

ahdl_include “lossy-inductor.va”

L1 (n 0) lossy_ind l=2.6nH esr=1m rcp=8 h=704k cp=230fF
I1 (n 0) isource mag=1

acswp ac start=10kHz stop=100GHz dec=20 title=“Coilcraft A01T Mini Spring”
```

(accuracy of *s*-parameters tend to degrade when impedance levels are far from the reference impedance of 50Ω). The loss modeling could be improved by decreasing H (or increasing k) by a factor of 4. However, a substantial discrepancy will clearly remain after correcting the parameter values because there is more detail in the measured data than can be represented with this model.

3.2 Limitations of the Verilog-A Model

The model as implemented exhibits two annoying short comings. First, since it is implemented using the *laplace_zp* function, the noise produced by the inductor is not modeled. To accurately model noise, an RC equivalent to the skin effect model would have to be synthesized.

Second, the number of lumps must be hard-coded in the model. There are two aspects of Verilog-A that combine to create this limitation. First, *laplace_zp* determines the order of the transfer function using the allocated size of the pole and zero arrays. Second, when declaring an array, the size cannot depend on the value of a parameter; it must be a constant. This constraint is a limitation of the current implementation of Verilog-A in Spectre, and should be relaxed in future versions.

LISTING 3. *Verilog-A model of the skin effect.*

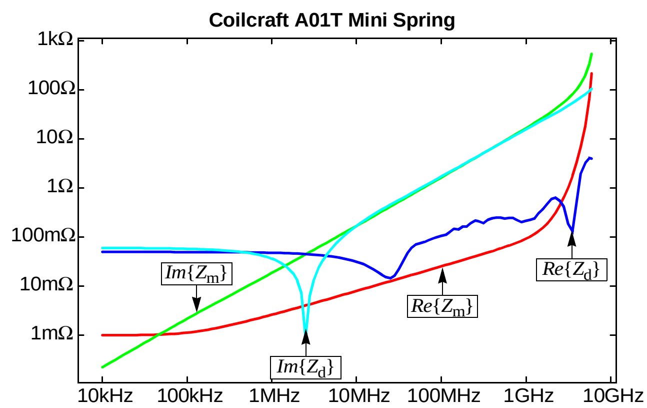
```
‘include “discipline.h”
‘include”constants.h”

module skin_effect(p, n);

electrical p, n;
parameter real f0=1 from (0:inf);
parameter real f1=10 from (f0:inf);
parameter real r0=1 from (0:inf);
‘define lumps 10
real mult, mult2, wp, wz;
real zeros[0:2*‘lumps–1], poles[0:2*‘lumps–1];
integer i;

    analog begin
        @(initial_step(“static”)) begin
            mult = pow(f1/f0, 1.0/(4*‘lumps));
            mult2 = mult*mult;
            wz = 2*‘M_PI*mult*f0;
            wp = mult2*wz;
            for (i=0; i < ‘lumps; i=i+1) begin
                zeros[2*i] = –wz;
                zeros[2*i+1] = 0;
                poles[2*i] = –wp;
                poles[2*i+1] = 0;
                wz = mult2 * wp;
                wp = mult2 * wz;
            end
        end
        V(p,n) <+ r0*laplace_zp( I(p,n), zeros, poles );
    end
endmodule
```

FIGURE 6. *The measured impedance (Z_d) of an RF inductor versus the impedance exhibited by the model with vendor provide parameter values (Z_m).*



4.0 FracPole

The two issues with the Verilog-A model could be resolved if Spectre supported a new type of basic element that exhibits the following impedance,

$$Z(s) = \alpha s^{\beta}. \tag{11}$$

As described before, if β is not an integer such a component is distributed and has time constants that are spread over an infinite range, from infinitely long to infinitely short. In a transient-based simulator, it must be approximated over a finite range of frequencies. So, consider a new basic component type, named a *fracpole*, which is short for fractional impedance pole. It approximates (11) over the range of frequencies from f_0 to f_1. Choose

$$\alpha = \frac{1}{\sqrt{2\pi}H}, \tag{12}$$

and β = ½. Doing so results in the over all impedance for the *fracpole* being

$$Z(f) = \frac{\sqrt{jf}}{H}. \tag{13}$$

A Spectre subcircuit implementation of the complete inductor model is given in Listing 4.

LISTING 4. *Inductor model.*

```
simulator lang=spectre

subckt lossyind (1 2)
    parameters l esr cp rp h f0 f1
    L (1 3) inductor l=l
    S (3 4) fracpole coef=1/sqrt(2*M_PI)*h slope=0.5 f0=f0 f1=f1 profile=dd
    Cp (1 5) capacitor c=cp
    Rp (5 4) resistor r=rp
    Rs (4 2) resistor r=esr
ends lossyind
```

4.1 Interim Approach

Since the *fracpole* component is not yet available in Spectre, an alternative is available on an interim basis. A program, separate from Spectre, is available that takes the *fracpole* parameters and generates a Spectre subcircuit that implements it [3]. This program is limited to generating fracpoles with negative slope, so a gyrator would be needed to effectively flip the sign of the slope. Also available is a program that takes a more direct approach. It takes the Coilcraft parameters and uses the *fracpole* program to directly generate a Spectre subcircuit description of the inductor of the form shown in Figure 1.

Appendix I

In this section, it is proved that the model of (2) is non-causal and therefore is not physically realizable.

Consider a system with frequency response $H(\omega)$. The impulse response is $h(t)$.

Postulate that $H(\omega)$ is purely real. From this we conclude that $h(t)$ is either zero for all $t \neq 0$, or it is non-causal.

Proof: $H(\omega) = H(-\omega)$ because of symmetry relations [$h(t)$ must be real, implying $H(\omega) = H(-\omega)^*$, and H(ω) is real by assumption]. Consider Fourier transform pair:

$H(\omega)$ is real and even $\Leftrightarrow$ $h(t)$ is real and even.

therefore, either $h(t)$ is zero for all $t \neq 0$, which only occurs if H is independent of ω, or $h(t)$ is non-zero for $t < 0$, in other words, it is non-causal. ■

Incidentally, for specific case of

$$H(\omega) \sim \frac{1}{\sqrt{\omega}} \tag{14}$$

$$h(t) \sim \frac{1}{\sqrt{|t|}}. \tag{15}$$

This pair is found in tables of sine/cosine transforms.

References

An alternative approach to modeling skin effect is given in [4].

[1] L. O. Chua. Device modeling via basic nonlinear circuit elements. *IEEE Transactions on Circuits and Systems*, vol. CAS-27, no. 11, Nov. 1980.

[2] *Modeling Coilcraft RF Inductors*. Coilcraft document 158, May 1999, *www.coilcraft.com*.

[3] K. Kundert. *The FracPole Suite*. Available from *www.designers-guide.com*. Includes executables and documentation.

[4] C. S. Yen, Z. Fazarinc, R. L. Wheeler. Time-domain skin-effect model for transient analysis of lossy transmission lines. *Proceedings of the IEEE*, vol. 70, pp. 750-757, July 1982.