Operating Point BODE100

Table of Contents

model=model_bode100(data,freq_L)

Author: JCCopyrights Summer 2019 Project: CRANE: Medical WPT for Deep Brain Stimulation Implants Creates a struct with power inductor model data extracted from raw data from Bode100 This function is created to work with the import_bode100 function. It is expeted to be used as data.model=model_bode100(data,freq_L);

Parameters

- @param **data** Struct of dara extracted from Bode100.
- @param **freq_L** Is the frequency which Xs/w is exactily the L of the coil.
- @retval **model** Returns the model values for each frequency measured in data struct

Code

```
function model=model_bode100(data,freq_L)

 %Look for closest frequency to freq_op
 %[ trash, i_aprox ] = min(abs( data.raw.f-freq_op));
 %f_aprox=data.raw.f(i_aprox);
 %text=sprintf('Aproximating f=%i to f_aprox=%i',freq_op,f_aprox);
 %Because disp function sucks dick.
 %disp(text);
 model.f=data.raw.f;
 %%Resonance calculations
 [model.Rp, i_res]=max(data.raw.Rs); %Damping Resistance (Rfe). Notice that Rp is also dependent of frequency.
 model.Rp=model.Rp*ones(length(data.raw.f),1);
 if i_res<length(data.raw.f) %Check if resonance is in the measured range
  model.f_res=data.raw.f(i_res);
 else
  warning('Resonance freq not in measurement range. Cp is WHATEVER');
  model.f_res=1.15*data.raw.f(length(data.raw.f)); %Random shit.
 end;
 % Notice that here we are ignoring the parasitic effects of Cp and R when assuming Ls is L at freq_op
 % This should be checked and studied. I am not going to. Because Fuck You.
 if nargin>1 %freq_L exist
  [ trash, i_L ] = min(abs( data.raw.f-freq_L));
  model.L=data.raw.Ls(i_L)*ones(length(data.raw.f),1);
 else
  model.L=data.raw.Ls(i_aprox)*ones(length(data.raw.f),1);
 end
 model.Cp=(1./
((2*pi*model.f_res)^2.*model.L)).*ones(length(data.raw.f),1);

 %% Real Coil Model
 % The parasitic effects of Cp and L over the Real impedance cannot be
 ignored
 % The parasitic effect of Rp near resonance cannot be ignored, but
 adding Rp to the impedance eliminates
 % any simple mathematical solution. So try not to work near resonance
 maybe?
 % @TODO: Introduce a numerical method to approximate model values
 withoud analytical constrains.
 w=2*pi*data.raw.f;
 a=w.^2.*model.Cp.^2.*data.raw.Rs;
 b=-1;
 c=data.raw.Rs.*((1-w.^2.*model.L.*model.Cp).^2);
 % This equation for a R>0 has two solutions one bigger than the
 other,
 % The correct solution shall be the smaller one, because of math and
 stuff (Convergence at 0+ when inf).
 model.R=(-b-sqrt(b^2-4*a.*c))./(2*a);
 % Model impedance
 Z_C=1./(j*w.*model.Cp);
 Zp=model.Rp;
 Z_LR=model.R+j*w.*model.L;
 model.Z=Z_C.*Z_LR./(Z_C+Z_LR);
 model.Z_damp=model.Z.*Zp./(model.Z+Zp);
 % Compare the model impedance with the measured impedance
 % Relative error compared with real measurements
 % If error>0 means model value is bigger
 model.error.err_real_damp=(real(model.Z_damp)-data.raw.Rs)./
data.raw.Rs;
 model.error.err_L_damp=(imag(model.Z_damp./w)-data.raw.Ls)./
data.raw.Ls;
 model.error.err_real=(real(model.Z)-data.raw.Rs)./data.raw.Rs;
 model.error.err_L=(imag(model.Z./w)-data.raw.Ls)./data.raw.Ls;
 %text=sprintf('Error model vs measured:\nReal %f\nImag
 %f',op.error.err_rel_Real*100,op.error.err_rel_L*100); %Because disp
 function sucks dick.
 %disp(text)



%    | - Name: Original csv file
%    |
%    |  |-f: frequency range
%    |  |-Z: Impedance in ohm
%    |- raw: |-theta: Impedance phase in degrees
%    |  |-Ls: Series Inductance
%    |  |-Rs: Real Impedance
%    |  |-Q: Quality factor
% data  |
%    |  |-f: frequency
%    |  |-f_res: resonance frequency
%    |-model:|-L: Model inductance
%    |  |-R: Copper losses
%    |  |-Cp: Parasitic capactitor
%    |  |-Rp: Damping Resistance
%    |  |-error: Relative error between model and measured data
%    |
%    |
```

Published with MATLAB® R2019a