FastHenry2 Creator

Table of Contents

file_name=fasthenry_creator(file_name,coils,freq)

This function will take coils as input and generate a FastHenry2 compatible output Beware that file_name must be a string 'This_is_a_File_Name' Coils must be a cell array of an arbitrary number of coils units in meters [m]

Parameters

- @param **file_name** Name of file to be created
- @param **coils** Cell array of compatible structs -> see generate_coil
- @param **freq** Frequencies to Evaluate the coils
- @retval **file_name** Original file_name with .inp extension

Code

```
function file_name=fasthenry_creator(file_name,coils,freq)
 file_name=[file_name '.inp'];
 fast_henry=fopen(file_name,'w');
 fprintf(fast_henry,'* FastHenry2 File Automatically Generated....
 JCCopyrights 2019\n');
 fprintf(fast_henry,'.Units M\n');%S.I.
 fprintf(fast_henry,'.Default z=0 sigma=5.8e4 w=%g h=%g nhinc=%g
 nwinc=%g\n',1,1,2,2);
 index=1;%Will assure no overlaping between Nodes for different coils
 %Generates Nodes and Segments for every Coil Introduced
 for j=1:1:size(coils,2)
  %Nodes ONLY depend of geometry
  fprintf(fast_henry,'\n*Coil %d Nodes:\n', j);
  coil=cell2mat(coils(j));
  X=coil.X; %coil geometry
  for i=index:1:index+size(X,2)-1
   fprintf(fast_henry,'N%d x=%g y=%g z=%g\n',i,X(1,i-index+1),X(2,i-
index+1),X(3,i-index+1));
  end

  %Segments include materials, coductors and discretization
 propierties
  fprintf(fast_henry,'\n*Coil %d Segments:\n', j);
  for i=index:1:index+size(X,2)-2
   fprintf(fast_henry,'E%d N%d N%d w = %g h = %g sigma = %g nhinc = %g
 nwinc = %g rh = %g rw = %g\n', ...
```

```
   i,i,i
+1,coil.w,coil.h,coil.sigma,coil.nhinc,coil.nwinc,coil.rh,coil.rw);

  end
  fprintf(fast_henry,'\n*Coil %d "%s" Electric Port:\n',
 j,coil.coil_name);
  fprintf(fast_henry,'.external N%d N%d %s' ,index,index+size(X,2)-1,
 coil.coil_name);
  fprintf(fast_henry,'\n');
  index=index+size(X,2);
 end
 %Notice that only works for ONE frequency
 fprintf(fast_henry,'.freq fmin=%d fmax=%d ndec=1', freq, freq);
 fprintf(fast_henry,'\n');
 fprintf(fast_henry,'.end');
 fclose(fast_henry);
end
```

Published with MATLAB® R2018b