
Circular Planar Inductor

Table of Contents

Parameters	1
Code	2
Geometry	3

`X = circular_planar_inductor(N,r0,ri,d,phi0,RES,h,x0,y0,z0,phix,phiy,phiz,view)`

Author: JCCopyrights Summer 2019 This function generates a planar circular multilayer spiral - PCB Inductor The coil will have enough layers to acomodate all N turns. The first layer will be generated with center in (0,0,0) in XY plane The layers will be generated below ($z < 0$) the first layer. It can be moved using the `x0,...,phix...` parameters

Parameters

- @param **N** Number of Turns
- @param **r0** External radius of the coil
- @param **ri** Internal radius of the coil
- @param **d** Distane bewtween turns
- @param **phi0** Angle at which the turns start
- @param **RES** Number of nodes of the Geometry (Discretization)
- @param **h** Distance between layers of the Coil. Can be interoduced as a single value (equidistant) or an array

```
%      With different distances between each layer.
%
% * @param  *x0* Center position X
%
% * @param  *y0* Center position Y
%
% * @param  *z0* Center position Z
%
% * @param  *phix* Turn respect X axis
%
% * @param  *phiy* Turn respect Y axis
%
% * @param  *phiz* Turn respect Z axis
%
% * @param  *view* Optional parameter, if true generates figure with
%                  geometry
%
% * @retval *X*      Geometry nodes
```

Code

```
function X =
circular_planar_inductor(N,r0,ri,d,phi0,RES,h,x0,y0,z0,phix,phiy,phiz,view)
Rx=[1,0,0;0,cos(phix),-sin(phix);0,sin(phix),cos(phix)];
Ry=[cos(phiy),0,sin(phiy);0,1,0;-sin(phiy),0,cos(phiy)];
Rz=[cos(phiz),-sin(phiz),0;sin(phiz),cos(phiz),0;0,0,1];

Nremaining=N; Nmax=floor((r0-ri)/d);
i=1;
while Nremaining>0 %Calculate turns per Layer
    if Nremaining>Nmax
        Nlayer(i)=Nmax;
        Nremaining=Nremaining-Nmax;
    else
        Nlayer(i)=Nremaining;
        Nremaining=Nremaining-Nremaining;
    end
    i=i+1;
end

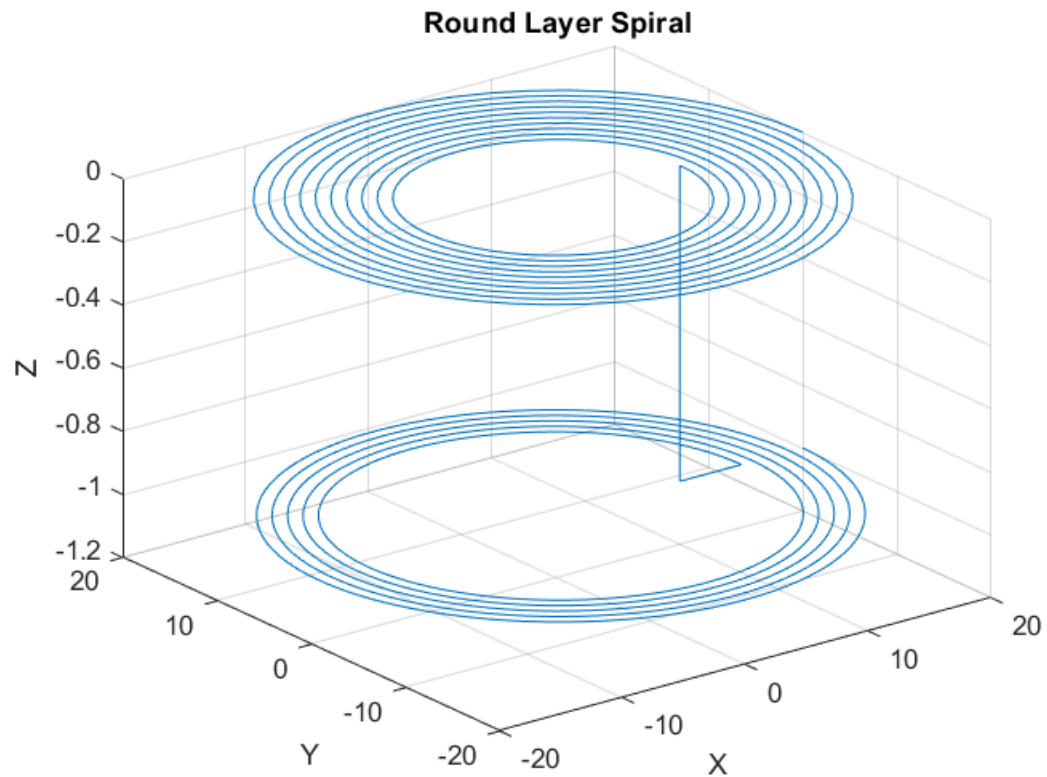
if length(h)==1
    hlayer=h/(size(Nlayer,2)-1); %Height of each layer
    zlayer=hlayer.*(0:1:(size(Nlayer,2)-1));
else
    hlayer=h;
    zlayer(1)=0;
    for i=2:1:(size(Nlayer,2))
        zlayer(i)=sum(hlayer(1:(i-1)));
    end
end

X=round_spiral(Nlayer(1), r0, d, phi0, RES, 0, 0, 0, 0, 0, 0, false);
for i=2:1:size(Nlayer,2)
    if mod(i,2)==1 %Assures the correct direction of the turns
        X=[X,round_spiral(Nlayer(i), r0, d, phi0, RES, 0, 0, -zlayer(i), 0,
0, 0, false)];
    else
        if Nlayer(i)== Nmax
            X=[X,flipplr(round_spiral(Nlayer(i), r0, d, phi0, RES, 0, 0, -
zlayer(i), pi, 0, 0, false))];
        else %Connection to the last turn has to be manually made
            Xaux=X(:,size(X,2))+[0;0;-hlayer]; %@TODO: Warning two points of
the inductor could overlap
            X=[X,Xaux,flipplr(round_spiral(Nlayer(i), r0, d, phi0, RES, 0, 0, -
zlayer(i), pi, 0, 0, false))];
        end
    end
end

for i=1:size(X,2)
    X(:,i)=transpose(Rx*[X(1,i);X(2,i);X(3,i)]);
    X(:,i)=transpose(Ry*[X(1,i);X(2,i);X(3,i)]);
end
```

```
X(:,i)=transpose(Rz*[X(1,i);X(2,i);X(3,i)]);  
X(:,i)=X(:,i)+[x0;y0;z0];  
end  
  
if nargin>13  
    if view  
        plot3(X(1,:),X(2,:),X(3,:))  
        grid on  
        xlabel('X')  
        ylabel('Y')  
        zlabel('Z')  
        title('Round Layer Spiral');  
    end  
end  
end
```

Geometry



Published with MATLAB® R2018b