```
%% Specification of Parameters
% Fundamental parameter values used as the basis for subsequent
% calculations:
%
% * Permeability of free space, $\mu_0, \rm{H/m}$
% * Relative permeability of core, $\mu_r$
% * Number of primary turns, $N_1$
% * Number of secondary turns, $N_2$
% * Effective magnetic core length, $l_e, \rm{m}$
% * Effective magnetic core cross-sectional area, $A_e, \rm{m^2}$
% * Core saturation begins, $B_{sat_{begin}}, \rm{T}$
% * Core fully saturated, $B_{sat}, \rm{T}$

mu_0 = pi*4e-7;
mu_r = 3000;
N1 = 100;
N2 = 200;
le = 0.1;
Ae = 1e-4;
B_sat_begin = 0.6;
B_sat = 1.2;

%% Calculate Magnetic Flux Density Versus Magnetic Field Strength Characteristic
% Where:
%
% * Magnetic flux density, $B, \rm{T}$
% * Magnetic field strength, $H, \rm{A/m}$
%
% Linear representation:
%
% * $B = \mu_0 \mu_r H$
%
% Nonlinear representation (including coefficient, a):
%
% * $B = B_{sat} \tanh(a.H)$

% Use linear representation to find value of H corresponding to B_sat_begin
H_sat_begin = B_sat_begin/(mu_0*mu_r);
% Rearrange nonlinear representation to calculate coefficient, a
a = atanh( B_sat_begin/B_sat )/H_sat_begin;

% Nonlinear representation
H_nonlinear = -750:25:750;
B_nonlinear = B_sat*tanh(a*H_nonlinear);

%% Use Parameters in Simscape Model
% The parameters calculated can now be used in a Simscape model. Once
% simulated, the model is set to output a Simscape logging variable,
% simlog_ee_custom_transformer, and some signals using output ports, yout.
% Circuit parameters are:
```

```
%
% * Voltage source magnitude, $V_s=10 \rm{V}$
% * Voltage source frequency, $Freq_{Hz}=60 \rm{Hz}$
% * Voltage source resistance, $R_{Vs}=10 \rm{\Omega}$
% * Operational amplifier input resistance, $R_1=1 \rm{k\Omega}$
% * Operational amplifier feedback resistance, $R_2=1 \rm{M\Omega}$
% * Operational amplifier feedback capacitance, $C_2=1 \rm{\mu F}$

% Circuit parameters
Vs = 10;
Freq_Hz = 60;
R_Vs = 10;
R_1 = 1e3;
R_2 = 1e6;
C_2 = 1e-6;

% Simulate model
simOut = sim( modelName );
yout = get(simOut, 'yout');
simlog = get(simOut, 'simlog_ee_custom_transformer');

% Collect internal Simscape logging data for comparison
I_simscape = simlog.Transformer_B_H_curve.L_m.i.series.values;
phi_simscape = simlog.Transformer_B_H_curve.L_m.phi.series.values;

% Collect model output data for comparison (as used for oscilloscope)
I_primary = yout(:,1);
int_V_secondary = yout(:,2);

%% Calculations on Logging and Output Data
% The data needs to be processed to provide the Magnetic field strength and
% Magnetic flux density data for comparison.
% Where:
%
% * Magnetomotive force, $F_m, \rm{A}$
% * Magnetic flux, $\phi, \rm{Wb}$
% * Operational amplifier input voltage, $V_{in}, \rm{V}$
% * Operational amplifier output voltage, $V_{out}, \rm{V}$
%
% Equations to be used are as follows:
%
% * $F_m = I.N_1$
% * $H = F_m/l_e$
% * $B = \phi/A_e$
% * $V_{out}=\frac{1}{R_1C_2}\int V_{in}.dt+c$
% * $\phi =\frac{1}{N_2}\int v.dt$
%

% Internal logging data
H_simscape = I_simscape.*N1./le;
```

```
B_simscape = phi_simscape./Ae;

% Oscilloscope scaling and model output data
H_measured = I_primary.*N1./le;
phi_measured = (int_V_secondary.*R_1.*C_2)./N2;
B_measured = phi_measured./Ae;

%% Conclusion
% The three characteristics can now be overlaid:
%
% * Defined characteristic: calculated from fundamental parameters
% * Characteristic from logging: calculated from internal Simscape logging
% data
% * Characteristic from measurement: obtained by measurement and
% calculation using electronic test circuit
%
% Due to leakage and parasitic parameters, the characteristic obtained from
% the electronic test circuit differs to the defined characteristic.
% However, the test circuit and its parameterization is shown to find the
% characteristic for the given transformer within suitable tolerances.

if ~exist('h1_ee_custom_transformer', 'var') || ...
        ~isgraphics(h1_ee_custom_transformer, 'figure')
    h1_ee_custom_transformer = figure('Name', 'ee_custom_transformer');
end
figure(h1_ee_custom_transformer)
clf(h1_ee_custom_transformer)

plot( ...
    H_nonlinear,...
    B_nonlinear,...
    'o',...
    H_simscape,...
    B_simscape,...
    H_measured,...
    B_measured,...
    '-',...
    'LineWidth',2);
grid( 'on' );
title( 'Transformer B-H Curve' );
xlabel( 'Magnetic Field Strength, H (A/m)' );
ylabel( 'Magnetic Flux Density, B (T)' );
legend( 'Calculated', 'Simulated',...
    'From Sensor', 'Location', 'NorthWest' );
```