```
% Nathan Schilling
% 02/19/20
% outputs t,I_1,I_2,V_Cap,d_vec
% Requires L1, L2 (transformer inductances) R1, R2, l_1, l_2 (loss inductances),
% magnetic coupling constants k_1, C, L_c, constant eta, B, m_p,
% R_Fcc
% need inital plasma ball radius (r_0), plasma velocity (v_0), I_1_0, V_cap_0,
% I_2_0, I_4_0
function [t,I_1,V_Cap,I_2,I_4,R_p,V_p] = circuitModelFunction_V_Cassibry3↙
(circuitInputParams)

    % Circuit parameters
    input.L_1=circuitInputParams.L_1;
    input.L_2=circuitInputParams.L_2;
    input.l_1=circuitInputParams.l_1;
    input.l_2=circuitInputParams.l_2;
    if ~isfield(circuitInputParams,'R1')
        input.R1=0;
    else
        input.R1=circuitInputParams.R_1;
    end
    if ~isfield(circuitInputParams,'R2')
        input.R2=0;
    else
        input.R2=circuitInputParams.R_2;
    end
    input.k_1=circuitInputParams.k_1;
    M_1_=input.k_1*sqrt(input.L_1*input.L_2);
    input.L_c=circuitInputParams.L_Fcc;
    input.C_load=circuitInputParams.C;
    input.eta=circuitInputParams.Eta;
    input.P_mag_hand=@(I) circuitInputParams.P_mag(I);
    input.m_p=circuitInputParams.m_p;
    input.R_fcc=circuitInputParams.r_Fcc;

    % Display parameters
    tauPerc=1;

    % Seed current
    I1_0=circuitInputParams.I0;
    Rp_0=circuitInputParams.Rp0;
    Vp_0=circuitInputParams.vp0;

    % Set up input functions
    tau=circuitInputParams.tau;
    input.Lp_hand=@(r) circuitInputParams.Lp_r_hand(r);
    input.M2_hand=@(Lp) circuitInputParams.M2_Lp_hand(Lp);
    input.dLp_hand=@(v) circuitInputParams.dLp_dt_v_hand(v);
    input.dM2_hand=@(Lp, v) circuitInputParams.dM2_Lp_v_hand(input.L_c,v);

    % Inital Condition [I1_0, Vcap_0, I2_0, I4_0, r_0, v_0]
```

```
    Iode=[I1_0;0;0;0;Rp_0;Vp_0];

    tSpan=linspace(0,tauPerc*tau,1e4);

    %options=odeset('RelTol',1e-20,'AbsTol',1e-20);
    options=odeset('Events',@zeroRpStopEvent);

    % Run until cap voltage starts to decrease
    sol=ode45(@(t,y) fcgfuns(t,y,input),tSpan,Iode,options);
    y=sol.y';
    t=sol.x;

    % Run again with R_2=1M Ohm
    %input.R2=1e6;
%     options=odeset('Events',@zeroRpStopEvent);
%     sol=ode45(@(t,y) fcgfuns(t,y,input),tSpan,y(end,:)',options);
%     y=[y;sol.y'];
%     t=[t sol.x];

    I_1=y(:,1);
    V_Cap=y(:,2);
    I_2=y(:,3);
    I_4=y(:,4);
    R_p=y(:,5);
    V_p=y(:,6);

    %this is the ode circuit solver
    function dI = fcgfuns(t,y,holderArray)
        I1=y(1);
        V_cap=y(2);
        I2=y(3);
        I4=y(4);
        r_p=y(5);
        v_p=y(6);

        if ~isfield(holderArray,'Lp_hand')
            msgID = 'fcgfuns:BadInput';
            msg = 'No Lp function specified.';
            baseException = MException(msgID,msg);
            throw(baseException);
        else
            Lp=holderArray.Lp_hand(r_p);
        end

        if ~isfield(holderArray,'M2_hand')
            msgID = 'fcgfuns:BadInput';
            msg = 'No M_2 function specified.';
            baseException = MException(msgID,msg);
            throw(baseException);
        else
            M_2=holderArray.M2_hand(Lp);
```

```
end

if ~isfield(holderArray,'dLp_hand')
    msgID = 'fcgfuns:BadInput';
    msg = 'No dLp_dt function specified.';
    baseException = MException(msgID,msg);
    throw(baseException);
else
    dL_p=holderArray.dLp_hand(v_p);
end

if ~isfield(holderArray,'dM2_hand')
    msgID = 'fcgfuns:BadInput';
    msg = 'No dM2_dt function specified.';
    baseException = MException(msgID,msg);
    throw(baseException);
else
    dM2=holderArray.dM2_hand(Lp,v_p);
end

if ~isfield(holderArray,'L_1')
    L_1=1;
else
    L_1=holderArray.L_1;
end

if ~isfield(holderArray,'L_2')
    L_2=1;
else
    L_2=holderArray.L_2;
end

if ~isfield(holderArray,'k_1')
    M_1=0.85*sqrt(L_1*L_2);
else
    M_1=holderArray.k_1*sqrt(L_1*L_2);
end

if ~isfield(holderArray,'l_1')
    l_1=0;
else
    l_1=holderArray.l_1;
end

if ~isfield(holderArray,'l_2')
    l_2=0;
else
    l_2=holderArray.l_2;
end

if ~isfield(holderArray,'R1')
```

```
    R_1=0;
else
    R_1=holderArray.R1;
end

if ~isfield(holderArray,'R2')
    R_2=0;
else
    R_2=holderArray.R2;
end

if ~isfield(holderArray,'C_load')
    C=1;
else
    C=holderArray.C_load;
end

if ~isfield(holderArray,'L_c')
    L_c=1;
else
    L_c=holderArray.L_c;
end

if ~isfield(holderArray,'P_mag_hand')
    msgID = 'fcgfuns:BadInput';
    msg = 'No P_mag function specified.';
    baseException = MException(msgID,msg);
    throw(baseException);
else
    P_mag=holderArray.P_mag_hand(I1);
end

if ~isfield(holderArray,'m_p')
    m_p=0;
else
    m_p=holderArray.m_p;
end

if ~isfield(holderArray,'eta')
    eta=0;
else
    eta=holderArray.eta;
end

if ~isfield(holderArray,'R_fcc')
    R_fcc=0;
else
    R_fcc=holderArray.R_fcc;
end

eta_l=eta*2*pi*r_p;
```

```
        dI=zeros(6,1);

        dI(1,1)=-(Lp*M_1*V_cap + I1*L_2*Lp*R_1 + I2*Lp*M_1*R_2 - I4*L_2*Lp*dM2...
            + I4*L_2*M_2*dL_p - I1*L_2*M_2*dM2 + I4*L_2*M_2*eta_l + I1*Lp*R_1*l_2...
            - I4*Lp*dM2*l_2 + I4*M_2*dL_p*l_2 - I1*M_2*dM2*l_2 + I4*M_2*eta_l*l_2)...
            /(L_1*L_2*Lp - Lp*M_1^2 - M_2^2*l_2 - L_2*M_2^2 + L_2*L_c*Lp + ...
            L_1*Lp*l_2 + L_2*Lp*l_1 + L_c*Lp*l_2 + Lp*l_1*l_2);

        dI(2,1)=I2/C;

        dI(3,1)=-(L_1*Lp*V_cap - M_2^2*V_cap + L_c*Lp*V_cap + Lp*V_cap*l_1 ...
            - I2*M_2^2*R_2 + I2*L_1*Lp*R_2 + I2*L_c*Lp*R_2 + I1*Lp*M_1*R_1 ...
            - I4*Lp*M_1*dM2 + I4*M_1*M_2*dL_p - I1*M_1*M_2*dM2 + I4*M_1*M_2*eta_l...
            + I2*Lp*R_2*l_1)/(L_1*L_2*Lp - Lp*M_1^2 - M_2^2*l_2 - L_2*M_2^2 ...
            + L_2*L_c*Lp + L_1*Lp*l_2 + L_2*Lp*l_1 + L_c*Lp*l_2 + Lp*l_1*l_2);

        dI(4,1)=-(M_1*M_2*V_cap - I4*M_1^2*dL_p + I1*M_1^2*dM2 - I4*M_1^2*eta_l...
            + I1*L_2*M_2*R_1 + I2*M_1*M_2*R_2 + I4*L_1*L_2*dL_p + I4*L_2*L_c*dL_p...
            - I1*L_1*L_2*dM2 - I1*L_2*L_c*dM2 - I4*L_2*M_2*dM2 + I4*L_1*L_2*eta_l...
            + I4*L_2*L_c*eta_l + I1*M_2*R_1*l_2 + I4*L_1*dL_p*l_2 + I4*L_2*dL_p*l_1...
            + I4*L_c*dL_p*l_2 - I1*L_1*dM2*l_2 - I1*L_2*dM2*l_1 - I1*L_c*dM2*l_2...
            - I4*M_2*dM2*l_2 + I4*L_1*eta_l*l_2 + I4*L_2*eta_l*l_1 + I4*L_c*eta_l*l_2...
            + I4*dL_p*l_1*l_2 - I1*dM2*l_1*l_2 + I4*eta_l*l_1*l_2)...
            /(L_1*L_2*Lp - Lp*M_1^2 - M_2^2*l_2 - L_2*M_2^2 + L_2*L_c*Lp + ...
            L_1*Lp*l_2 + L_2*Lp*l_1 + L_c*Lp*l_2 + Lp*l_1*l_2);

        dI(5,1)=v_p;

        dI(6,1)=(-P_mag/m_p)*(2*pi*r_p*R_fcc);
    end

    % Stops integration if Rp is 0
    function [value,isterminal,direction] = zeroRpStopEvent(t,y)
        value=y(5);
        isterminal=1;
        direction=[];
    end

    % Stop the integration if the voltage on the capacitor starts to
    % decrease
    function [value,isterminal,direction] = capBackStopEvent(t,y)
        value=y(3);
        isterminal=1;
        direction=-1;
    end
end
```