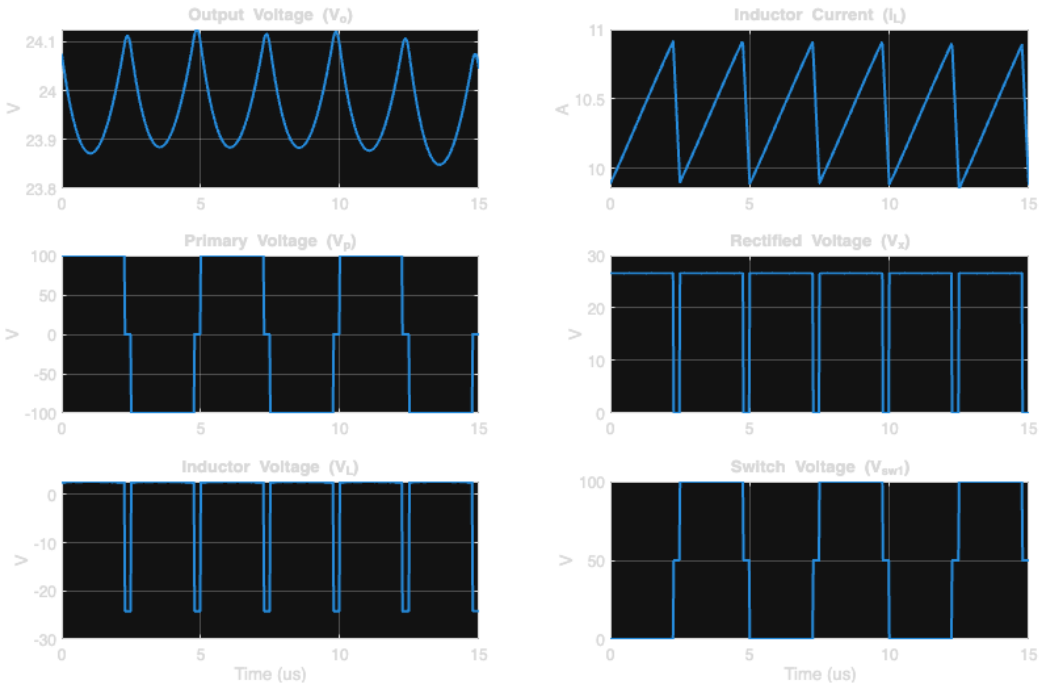
```
% EE463 HW3 - Part 3 Test Script
% We finished the calculations for Part 1 and 2, but we are going to double
% check the resistance values of the cable tomorrow just to be sure.
% I wrote this script to test if the RK4 solver works and to see what the
% plots look like with our current values. We wrote this code for checkşng
% the plots we wll obtain tomorrow.
% I will update the R_pri and R_sec values tomorrow and run the final
% simulation on Simulink then. For now, this is just to verify the topology.


Vin = 100;
Vout_req = 24;
P_rated = 250;
fsw = 200e3;

Np_Ns_ratio = 15/4;
L_filter = 5.76e-6;
C_out = 1.36e-6;
% Resistance values (using estimates for now, will update with exact calc)
R_pri = 0.012;
R_sec = 0.004;
R_ind = 0.005;

R_load = (Vout_req^2) / P_rated;
Duty = (Vout_req * Np_Ns_ratio) / (2 * Vin);

fprintf('Duty Cycle: %.4f \n', Duty);

T_sim = 0.001;
dt = (1/fsw) / 200;
time = 0:dt:T_sim;
N = length(time);

x = zeros(2, N);
x(:, 1) = [Vout_req/R_load; Vout_req];

vp_arr = zeros(1, N);
vx_arr = zeros(1, N);
vl_arr = zeros(1, N);
vsw1_arr = zeros(1, N);

for k = 1:N-1
    t_curr = time(k);
    state_curr = x(:, k);

    [dx1, aux1] = get_derivatives(t_curr, state_curr, Vin, Duty, fsw,
Np_Ns_ratio, L_filter, C_out, R_load, R_pri, R_sec, R_ind);
    [dx2, ~] = get_derivatives(t_curr + dt/2, state_curr + dx1*dt/2, Vin,
Duty, fsw, Np_Ns_ratio, L_filter, C_out, R_load, R_pri, R_sec, R_ind);
    [dx3, ~] = get_derivatives(t_curr + dt/2, state_curr + dx2*dt/2, Vin,
Duty, fsw, Np_Ns_ratio, L_filter, C_out, R_load, R_pri, R_sec, R_ind);
    [dx4, ~] = get_derivatives(t_curr + dt, state_curr + dx3*dt, Vin, Duty,
```

```
fsw, Np_Ns_ratio, L_filter, C_out, R_load, R_pri, R_sec, R_ind);

    x(:, k+1) = state_curr + (dt/6) * (dx1 + 2*dx2 + 2*dx3 + dx4);

    vp_arr(k) = aux1.Vp;
    vx_arr(k) = aux1.Vx;
    vl_arr(k) = aux1.VL;
    vsw1_arr(k) = aux1.Vsw1;
end

vp_arr(N) = vp_arr(N-1);
vx_arr(N) = vx_arr(N-1);
vl_arr(N) = vl_arr(N-1);
vsw1_arr(N) = vsw1_arr(N-1);

cycles = 3;
samples = round((cycles/fsw) / dt);
start_idx = N - samples;
range = start_idx:N;
t_plot = (time(range) - time(start_idx)) * 1e6;

figure('Color', 'w');

subplot(3, 2, 1);
plot(t_plot, x(2, range), 'LineWidth', 1.5);
title('Output Voltage (V_o)'); ylabel('V'); grid on;

subplot(3, 2, 2);
plot(t_plot, x(1, range), 'LineWidth', 1.5);
title('Inductor Current (i_L)'); ylabel('A'); grid on;

subplot(3, 2, 3);
plot(t_plot, vp_arr(range), 'LineWidth', 1.5);
title('Primary Voltage (V_p)'); ylabel('V'); grid on;

subplot(3, 2, 4);
plot(t_plot, vx_arr(range), 'LineWidth', 1.5);
title('Rectified Voltage (V_x)'); ylabel('V'); grid on;

subplot(3, 2, 5);
plot(t_plot, vl_arr(range), 'LineWidth', 1.5);
title('Inductor Voltage (V_L)'); ylabel('V'); xlabel('Time (us)'); grid on;

subplot(3, 2, 6);
plot(t_plot, vsw1_arr(range), 'LineWidth', 1.5);
title('Switch Voltage (V_{sw1})'); ylabel('V'); xlabel('Time (us)'); grid on;

iL_ss = x(1, range);
vp_ss = vp_arr(range);

i_pri = iL_ss / Np_Ns_ratio;
i_sw1 = i_pri;
i_sw1(vp_ss <= 0) = 0;
```

```
I_rms = sqrt(mean(i_sw1.^2));
V_ripple = max(x(2, range)) - min(x(2, range));

fprintf('\nCheck Results:\n');
fprintf('MOSFET RMS: %.3f A\n', I_rms);
fprintf('Ripple: %.3f V\n', V_ripple);
fprintf('Mean Vout: %.3f V\n', mean(x(2, range)));

function [dx, aux] = get_derivatives(t, state, Vin, D, fsw, n, L, C, Rload,
Rpri, Rsec, Rind)
    iL = state(1);
    vC = state(2);
    T = 1/fsw;
    t_cyc = mod(t, T);

    if t_cyc < (D * T)
        Vp = Vin;
        Req = Rsec + (Rpri / n^2);
        Vx = (Vin/n) - (iL * Req);
        Vsw1 = 0;
    elseif (t_cyc >= 0.5*T) && (t_cyc < (0.5*T + D*T))
        Vp = -Vin;
        Req = Rsec + (Rpri / n^2);
        Vx = (Vin/n) - (iL * Req);
        Vsw1 = Vin;
    else
        Vp = 0;
        Vx = 0 - (iL * Rsec);
        Vsw1 = Vin/2;
    end

    vL_val = Vx - (iL * Rind) - vC;
    di_dt = vL_val / L;
    dv_dt = (iL - vC/Rload) / C;

    dx = [di_dt; dv_dt];
    aux.Vp = Vp; aux.Vx = Vx; aux.VL = vL_val; aux.Vsw1 = Vsw1;
end
```

```
Duty Cycle: 0.4500

Check Results:
MOSFET RMS: 1.866 A
Ripple: 0.277 V
Mean Vout: 23.962 V
```



Published with MATLAB® R2025a