Appendix – 1 [calc.mlx]

```
clearvars
u = symunit;
```

```
syms d turnsRatio
format shortEng
% format short
v_o = 48
```

```
v_o =
    48.0000e+000
```

```
d_min = 0.278; v_d_minduty = 18;
d_max = 0.366; v_d_maxduty = 12
```

```
v_d_maxduty =
    12.0000e+000
```

```
turnsRatio_minduty = ( (d_min/(1-d_min)) * (v_d_minduty/v_o) )^-1
```

```
turnsRatio_minduty =
     6.9257e+000
```

```
turnsRatio_maxduty = ( (d_max/(1-d_max)) * (v_d_maxduty/v_o) )^-1
```

```
turnsRatio_maxduty =
     6.9290e+000
```

```
U_o = v_o;
v_t = d_max;
f_sw = 100e3;
i_out = 1;
i_avgSec = i_out/(1-v_t);
xformerCurrRipple = 0.5; % percent
L_sec = (U_o*(1-v_t))/(xformerCurrRipple*i_avgSec*f_sw)
```

```
L_sec =
   385.8778e-006
```

```
L_pri = L_sec/(turnsRatio_maxduty^2)
```

```
L_pri =
     8.0374e-006
```

```
% (turnsRatio_maxduty^2)*2.814e-6
```

```
syms priTurns secTurns
AL = 51e-9 % nH/T^2; minimal
```

```
AL =
    51.0000e-009
```

```
priTurns = double(solve(L_pri == AL*priTurns^2))
```

```
priTurns = 2×1
   -12.5537e+000
    12.5537e+000
```

```
secTurns = double(solve(L_sec == AL*secTurns^2))
```

```
secTurns = 2×1
   -86.9841e+000
    86.9841e+000
```

```
% make sure core is not saturated
ampTurns = i_out*secTurns
```

```
ampTurns = 2×1
   -86.9841e+000
    86.9841e+000
```

AWG selection

```
p_o = i_out * v_o
```

```
p_o =
    48.0000e+000
```

```
i_in_max = v_o/v_d_maxduty
```

```
i_in_max =
     4.0000e+000
```

```
selectedAWGRating = 0.226;
num_of_paralles_sec = i_out/selectedAWGRating
```

```
num_of_paralles_sec =
     4.4248e+000
```

```
num_of_paralles_pri = i_in_max/selectedAWGRating
```

```
num_of_paralles_pri =
    17.6991e+000
```

Fill Factor Calculation

```
windowArea_mm2 = 427;
priTurns = ceil(priTurns(priTurns>0))
```

```
priTurns =
    13.0000e+000
```

```
secTurns = ceil(secTurns(secTurns>0))
```

```
secTurns =
    87.0000e+000
```

```
num_of_paralles_pri = ceil(num_of_paralles_pri)
```

```
num_of_paralles_pri =
    18.0000e+000
```

```
num_of_paralles_sec = ceil(num_of_paralles_sec)
```

```
num_of_paralles_sec =
     5.0000e+000
```

```
cableArea_mm2 = 0.080;

primaryArea_mm2 = priTurns*num_of_paralles_pri*cableArea_mm2
```

```
primaryArea_mm2 =
    18.7200e+000
```

```
secondaryArea_mm2 = secTurns*num_of_paralles_sec*cableArea_mm2
```

```
secondaryArea_mm2 =
    34.8000e+000
```

```
totalCableArea_mm2 = primaryArea_mm2 + secondaryArea_mm2
```

```
totalCableArea_mm2 =
    53.5200e+000
```

```
fillFactor_perc = 100*totalCableArea_mm2/windowArea_mm2
```

```
fillFactor_perc =
    12.5340e+000
```

Cable Resistance Calculation

```
windingLengthPerTurn_mm = 68.2
```

```
windingLengthPerTurn_mm =
    68.2000e+000
```

```
ohms_per_meter = 212.872 / 1e3
```

```
ohms_per_meter =
   212.8720e-003
```

```
primaryLength_m = windingLengthPerTurn_mm * priTurns * 1e-3
```

```
primaryLength_m =
   886.6000e-003
```

```
secondaryLength_m = windingLengthPerTurn_mm * secTurns * 1e-3
```

```
secondaryLength_m =
     5.9334e+000
```

```
primary_DC_resistance_ohm = ohms_per_meter * primaryLength_m / num_of_paralles_pri
```

```
primary_DC_resistance_ohm =
    10.4851e-003
```

```
secondary_DC_resistance_ohm = ohms_per_meter * secondaryLength_m / num_of_paralles_sec
```

```
secondary_DC_resistance_ohm =
   252.6109e-003
```

Copper Loss Calculation

```
diameter_mm = vpa(0.32004*u.mm)
```

diameter_mm = 0.32004 mm

```
radius_mm = diameter_mm/2
```

radius_mm = 0.16002 mm

```
skinDepth_cm = vpa(7.5/sqrt(f_sw)*u.cm)
```

skinDepth_cm = 0.023717082451262844989991701583245 cm

```
skinDepth_mm = unitConvert(skinDepth_cm, u.mm)
```

skinDepth_mm = 0.23717082451262844989991701583245 mm

```
% skin depth is greater than radius.
% Therefore, AC reistance equals DC resistance
DC_to_AC_ratio = 1
```

```
DC_to_AC_ratio =
     1.0000e+000
```

```
primary_AC_resistance_ohm = primary_DC_resistance_ohm*DC_to_AC_ratio
```

```
primary_AC_resistance_ohm =
    10.4851e-003
```

```
secondary_AC_resistance_ohm = secondary_DC_resistance_ohm*DC_to_AC_ratio
```

```
secondary_AC_resistance_ohm =
   252.6109e-003
```

```

resistancePri_ohm = vpa(primary_AC_resistance_ohm * u.Ohm)
```

resistancePri_ohm = 0.010485128622222223207516300647058 Ω

```
resistanceSec_ohm = vpa(secondary_AC_resistance_ohm * u.Ohm)
```

resistanceSec_ohm = 0.25261094496000002784796834021108 Ω

```
copperLossPri = vpa(unitConvert((i_in_max*u.A)^2 * resistancePri_ohm, u.W))
```

copperLossPri = 0.16776205795555557132026081035292 W

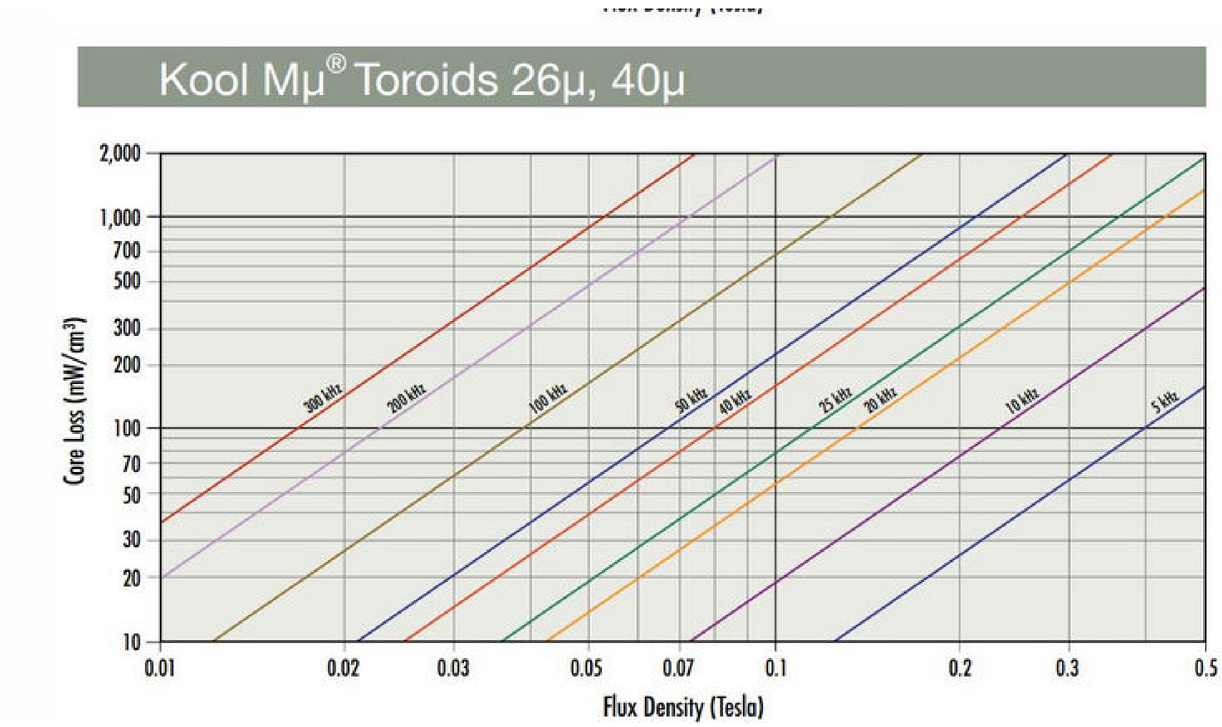
```
copperLossSec = vpa(unitConvert((i_out*u.A)^2 * resistanceSec_ohm, u.W))
```

copperLossSec = 0.25261094496000002784796834021108 W

```
copperLoss_W = copperLossPri + copperLossSec
```

copperLoss_W = 0.42037300291555559916822915O564 W

Core Loss Calculation




```
permeability = 26;
mu_zero = 1.25663706212e-6;
pathLength_m = 107e-3;
fluxDensity_Tesla = mu_zero * permeability * ampTurns / pathLength_m
```

```
fluxDensity_Tesla = 2×1
   -26.5607e-003
    26.5607e-003
```

```
% using graph above, 0.03 Tesla @ 100 kHz corresponds to
wattLoss_mW_cm3 = 60*u.mW/u.cm^3
```

wattLoss_mW_cm3 =

$60 \frac{\mathrm{mW}}{\mathrm{cm}^3}$

```
volume_mm3 = 21300;
volume_cm3 = vpa(unitConvert(volume_mm3*u.mm^3, u.cm^3))
```

```
volume_cm3 = 21.3 cm^3
```

```
coreLoss_w = vpa(unitConvert(wattLoss_mW_cm3 * volume_cm3, u.W))
```

```
coreLoss_w = 1.278 W
```

```
magnetizingResistance = v_d_minduty^2/p_o
```

```
magnetizingResistance =
     6.7500e+000
```

```
Lm = 8*10^-6; f_sw = 100*10^3;

% DCM
% for Vs = 12V
Vs = 12; D = 0.366;
deltaI_lm = Vs*D/(Lm*f_sw);
P_min = Vs^2 * D^2 / (2*Lm*f_sw);
I_load_min = P_min / 48
```

```
I_load_min =
   251.1675e-003
```

```

% for Vs = 18V
Vs = 18; D = 0.278;
deltaI_lm = Vs*D/(Lm*f_sw);
P_min = Vs^2 * D^2 / (2*Lm*f_sw);
I_load_min = P_min / 48
```

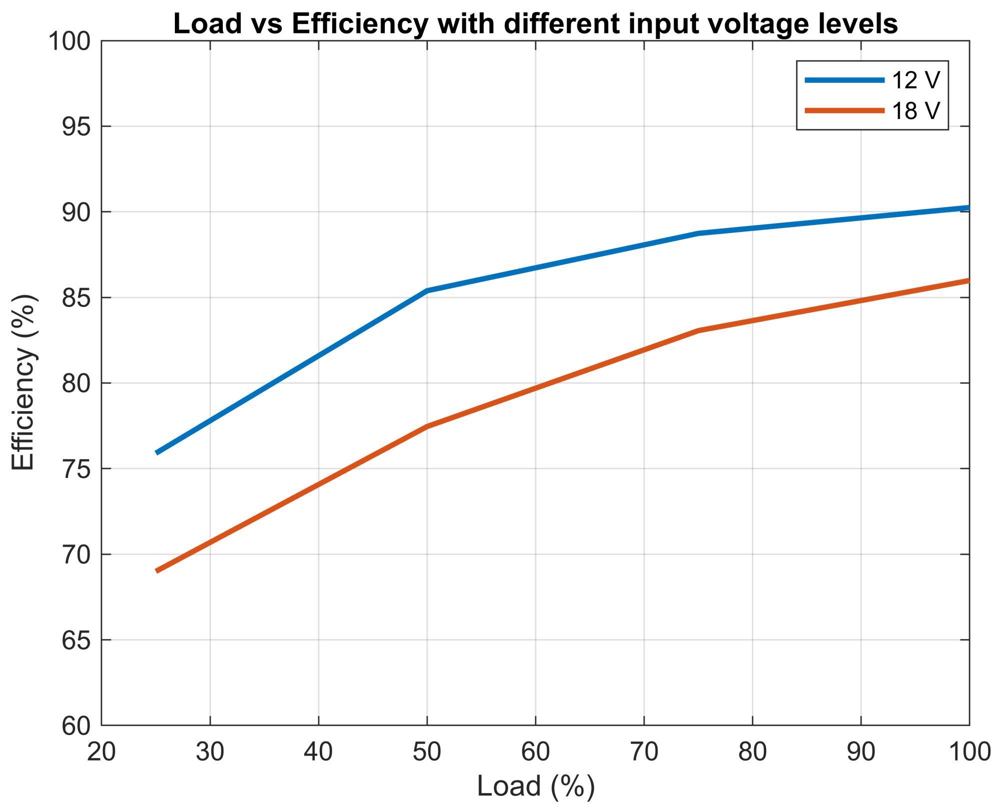
```
I_load_min =
   326.0419e-003
```

```
% max I_Lm current occurs when input voltage is 12V and at 100% load
Vs = 12; D = 0.366;
turnsRatio = 6.92; R = 48;
deltaI_lm = Vs*D/(Lm*f_sw);
P_out = Vs^2 * D^2 * turnsRatio^2 / ((1-D)^2 * R);
I_Lm_max = deltaI_lm/2 + P_out/(Vs*D)
```

```
I_Lm_max =
    13.6457e+000
```

```
% efficiency plot
plot([25 50 75 100], [75.9 85.39 88.74 90.25], 'LineWidth', 2);
hold on
```

```
plot([25 50 75 100], [69 77.46 83.06 85.99], 'LineWidth', 2);
grid on
legend("12 V", "18 V");
xlabel("Load (%)");
ylabel("Efficiency (%)")
title("Load vs Efficiency with different input voltage levels");
ylim([60 100])
```




```
% loss calculations
P_out = [12 24 36 48];  Vs = [12; 18];   Vf = 0.5; Rds_on = 3.6*10^-3; Q = 22.6*10^-9;
I_in_avg = P_out./Vs;    I_out_avg = P_out./48;
P_c = 1.278;

P_mosfet_conduction = I_in_avg.^2*Rds_on;    P_mosfet_switching = 19*I_in_avg*Q*f_sw;
P_diode_conduction = I_out_avg*Vf;
P_copper = I_in_avg.^2*primary_AC_resistance_ohm + I_out_avg.^2*secondary_AC_resistance_ohm;

P_total_loss = P_c + P_mosfet_conduction + P_mosfet_switching + P_diode_conduction + P_copper;
efficiency = 100-P_total_loss./P_out*100
```

```
efficiency = 2×4
    87.7016e+000    92.7776e+000    94.3037e+000    94.9422e+000
    87.8860e+000    93.0273e+000    94.6186e+000    95.3223e+000
```

```
% component selections
% mosfet
duty = [d_min; d_max]
```

```
duty = 2×1
   278.0000e-003
   366.0000e-003
```

```
V_sw = Vs + 1/turnsRatio*48
```

```
V_sw = 2×1
    18.9364e+000
    24.9364e+000
```

```
I_sw = 1./(1-duty).*turnsRatio.*I_out_avg + 1/turnsRatio.*(1-duty)./2./L_pri./f_sw*48
```

```
I_sw = 2×4
     5.5116e+000     7.9078e+000    10.3039e+000    12.7000e+000
     5.4645e+000     8.1932e+000    10.9219e+000    13.6506e+000
```

```
% diode
V_d_max = Vs*turnsRatio + 48
```

```
V_d_max = 2×1
   131.0400e+000
   172.5600e+000
```

```
I_d_max = I_sw./turnsRatio
```

```
I_d_max = 2×4
   796.4784e-003     1.1427e+000     1.4890e+000     1.8353e+000
   789.6655e-003     1.1840e+000     1.5783e+000     1.9726e+000
```

```
% output voltage ripple
R = 48; ripple = 0.03; ESR = 2
```

```
ESR = 
     2.0000e+000
```

```
C_min = duty/f_sw/R/ripple
```

```
C_min = 2×1
     1.9306e-006
     2.5417e-006
```

```
V_ESR = ESR * I_out_avg
```

```
V_ESR = 1×4
   500.0000e-003     1.0000e+000     1.5000e+000     2.0000e+000
```

Appendix – 2 [TI Webench Design]