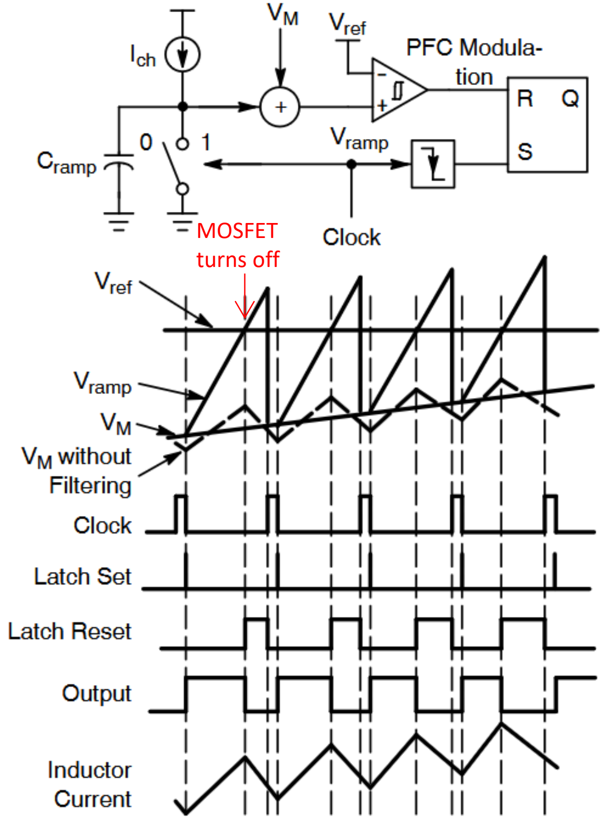
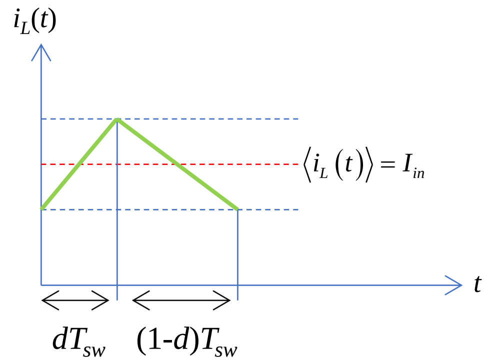
The NCP1654 has been released after the NCP1653 which came out in 2005. Both engines include what is called a *predictive control law*. Unlike the former series of CCM PFCs like the UC1854, there is no need to sense the input voltage for generating the duty ratio. The 1654 senses the input voltage, but for implementing brown-out protection and scaling the control voltage at low and high lines.



This is the internal modulator of the NCP1653/54



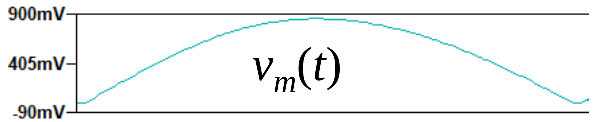
Boost dc transfer characteristic

$$\frac{V_{out}}{V_{in}} = \frac{1}{1-d} \Rightarrow V_{in} = V_{out}(1-d)$$

$$Z_{in} = \frac{V_{in}}{I_{in}} = \frac{V_{in}}{\langle i_L(t)\rangle} = \frac{V_{out}}{\langle i_L(t)\rangle}(1-d)$$

Z_{in} has to be constant if we want to emulate a resistance

The modulator is made of a capacitor C_{ramp} charged by I_{ch}. The MOSFET turn-off occurs when the ramp voltage intersects with the reference voltage. The ramp is actually seating on a component, V_m, which is the absolute value of the sinusoidal envelope:



The transition occurs when $V_{ramp} = V_{ref}$ or:

$$V_m + \frac{\boxed{I_{ch}} dT_{sw}}{C_{ramp}} = V_{ref}$$

The ramp duration lasts a switching period and is easily determined as:

$$T_{sw} = \frac{C_{ramp}V_{ref}}{I_{ch}} \rightarrow I_{ch} = \frac{C_{ramp}V_{ref}}{T_{sw}}$$

Replace $\boxed{I_{ch}}$ in the above expression

$$\Rightarrow V_{ref} = V_m + \frac{\left(C_{ramp}V_{ref}/T_{sw}\right)dT_{sw}}{C_{ramp}} \rightarrow V_m = V_{ref}(1-d) \Rightarrow Z_{in} = \frac{V_{out}}{\langle i_L(t)\rangle}\frac{V_m}{V_{ref}}$$

Because V_{ref} and V_{out} are nearly constant over time, the voltage V_m will be made proportional to the average inductor current for emulating a resistive input:

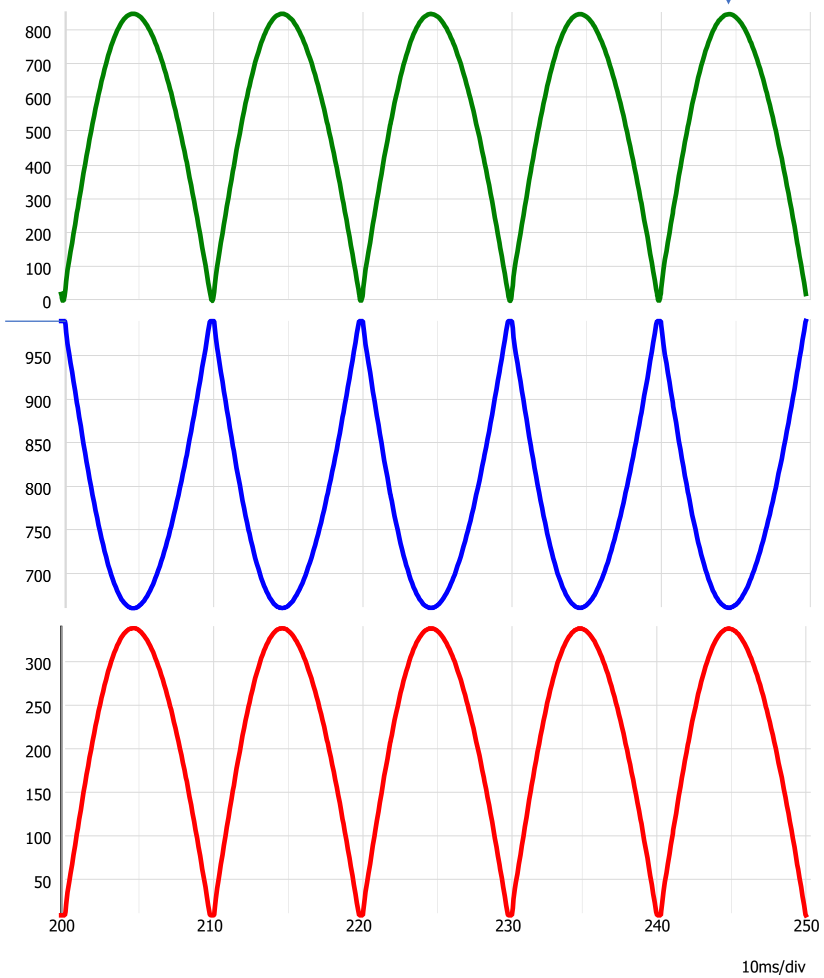
Averaged value

$$I_L(t) = \frac{V_{out}}{Z_{in}} \frac{|v_m \sin \omega t|}{V_{ref}}$$

Programmed for P_{out}

Fixed to 2.5 V

The height of v_m is adjusted based on the input voltage supplied to the PFC. The error voltage will do this through a multiplier in the chip.

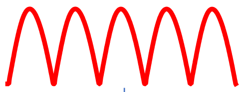


$v_m(t)$

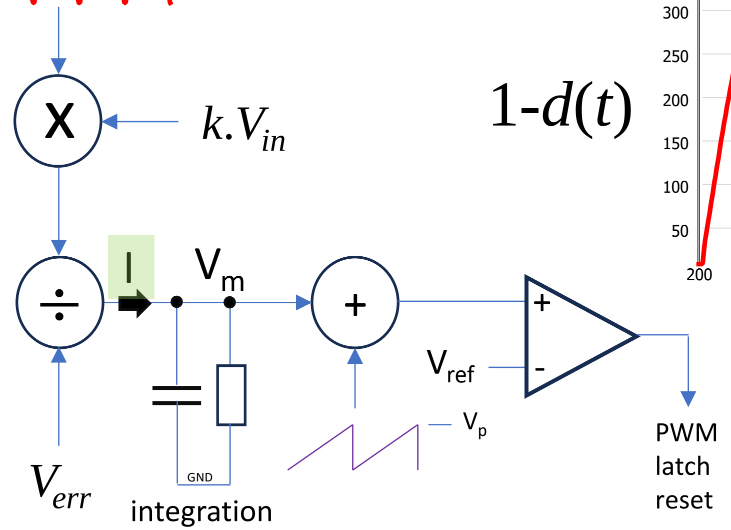
$d(t)$

$1\text{-}d(t)$

Scaled rectified inductor current



$|i_L(t)|$



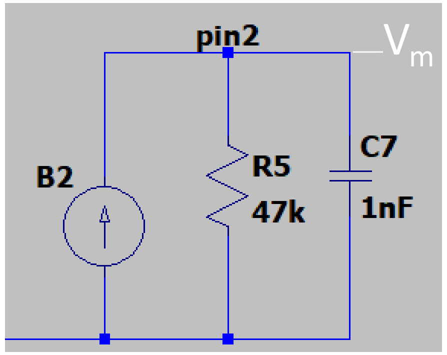
P = 1.3 kW

$V_{m,peak}$ = 2 V, V_{in} = 230 V

$V_{m,peak}$ = 0.82 V, V_{in} = 100 V

V_m drives the off-time

→ It reduces if V_{in} decreases



I=((-V(Rs)/{ROCP})*V(BO))/(4*(V(CTRL)-0.55))

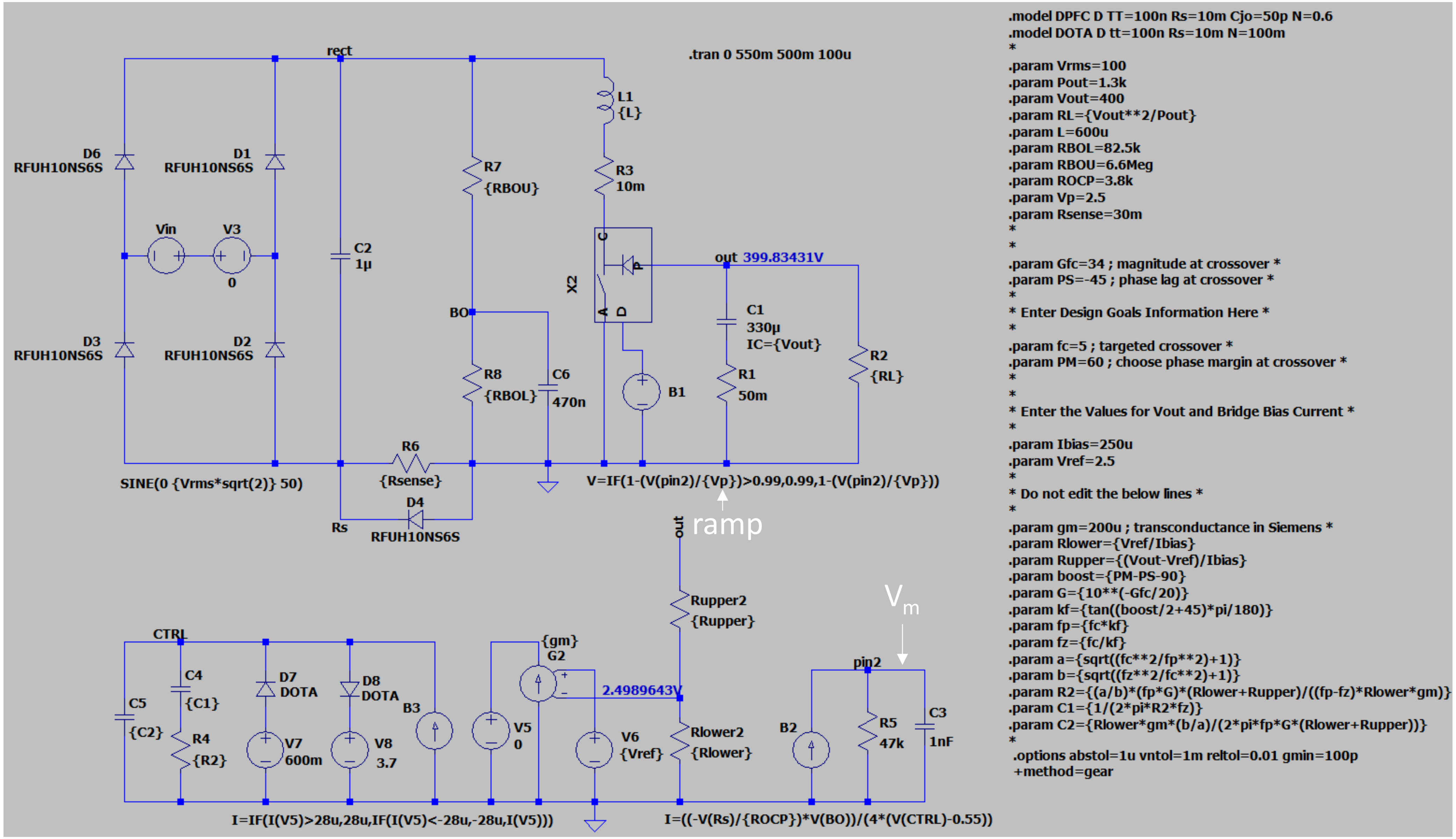
$i_L(t)$ scaled by R_{sense}

Scales with input voltage

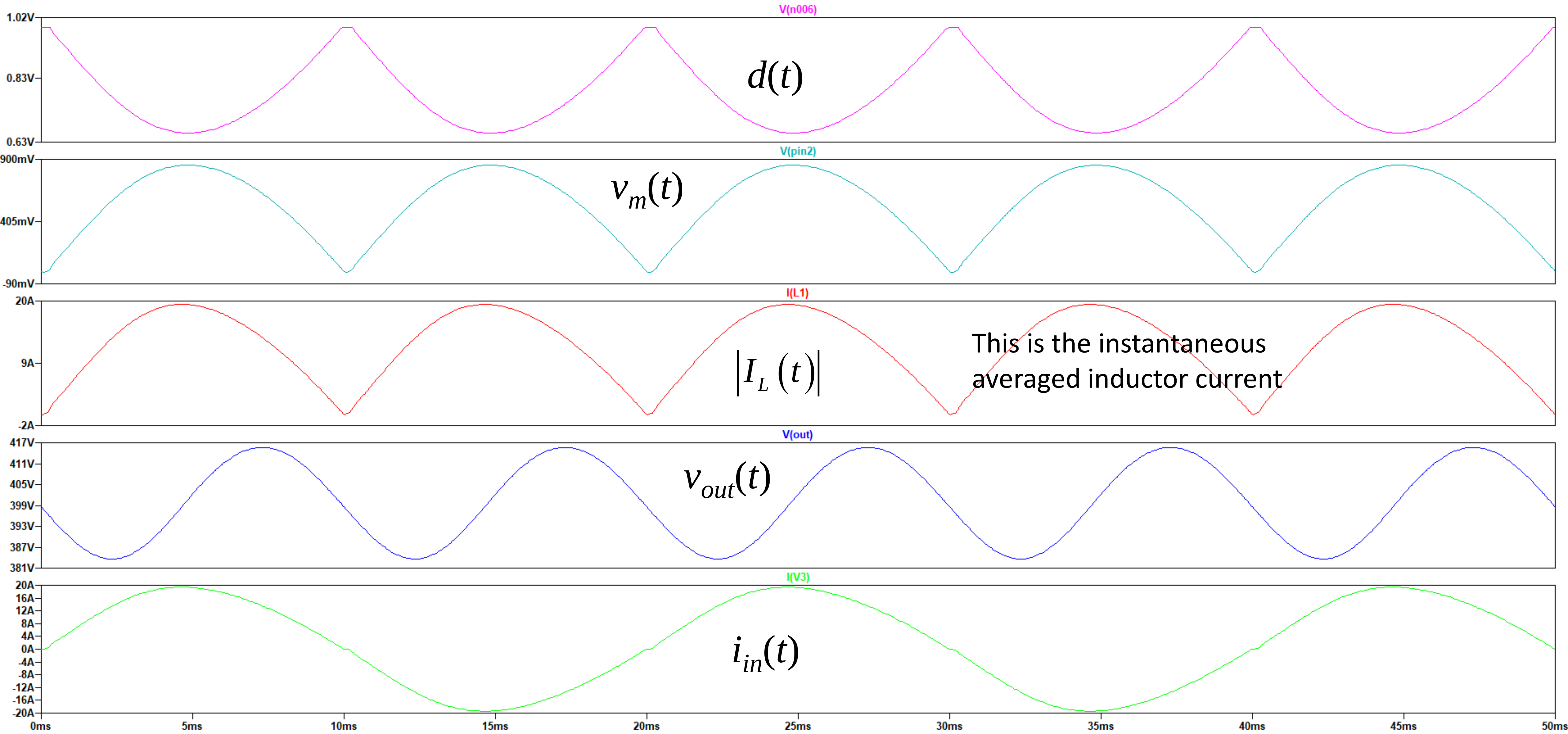
Error voltage

Homogeneous to power

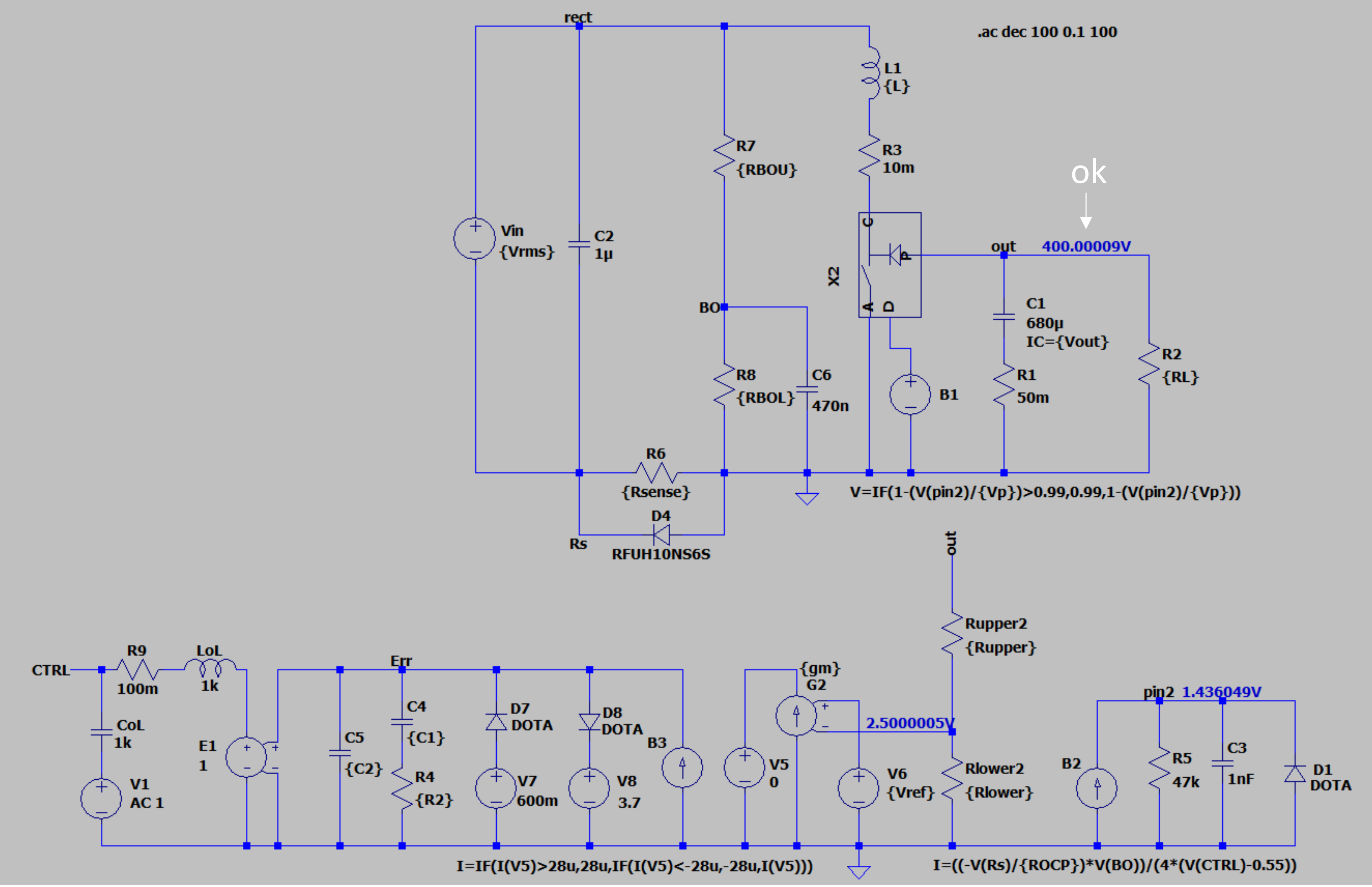
It is possible to build an averaged model with the voltage-mode CCM PWM switch model:



These are the simulated waveforms for a 1.3-kW CCM PFC supplied from a 100-V rms input line

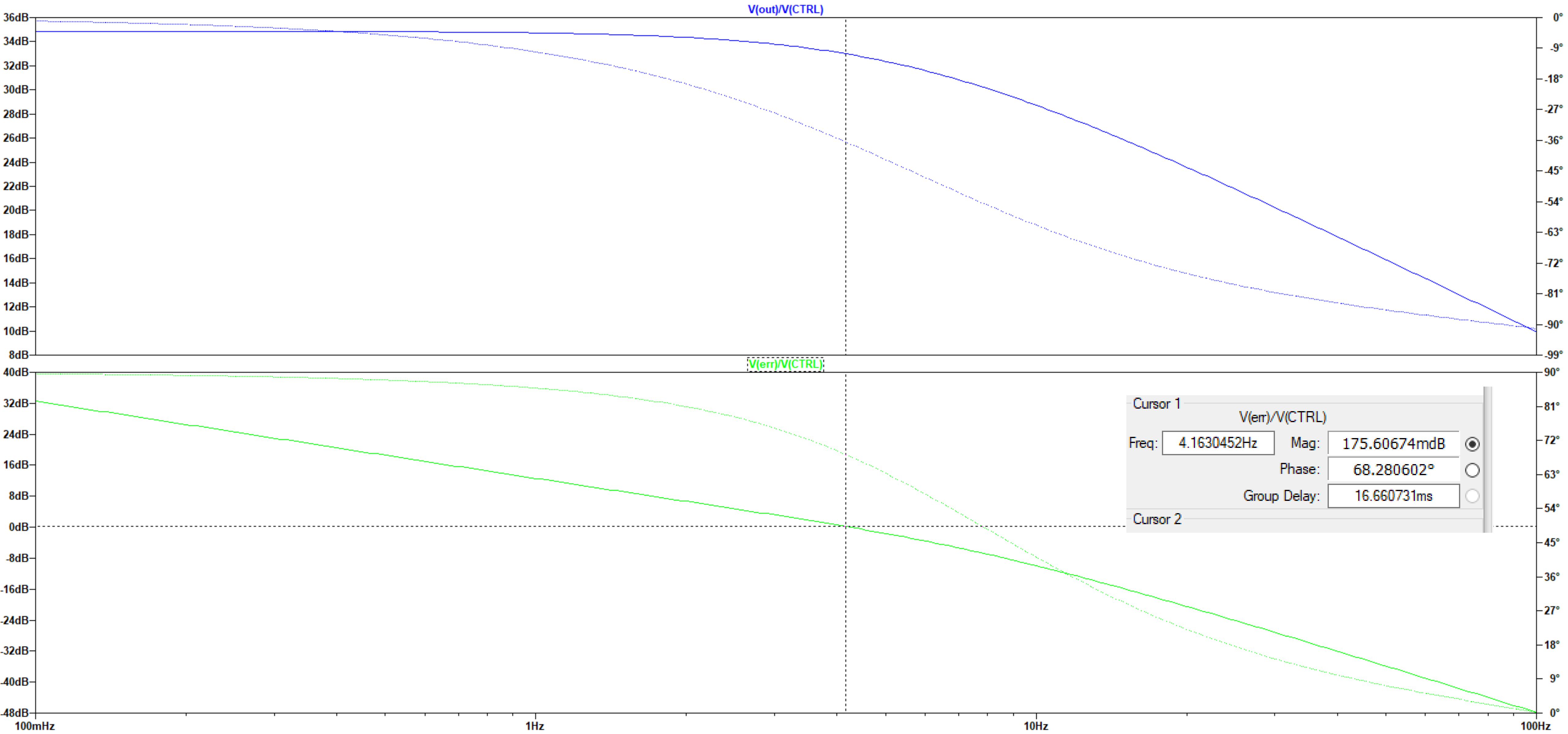


We can use the averaged model to extract the ac response of the power stage and stabilize it. Check bias points are correct.

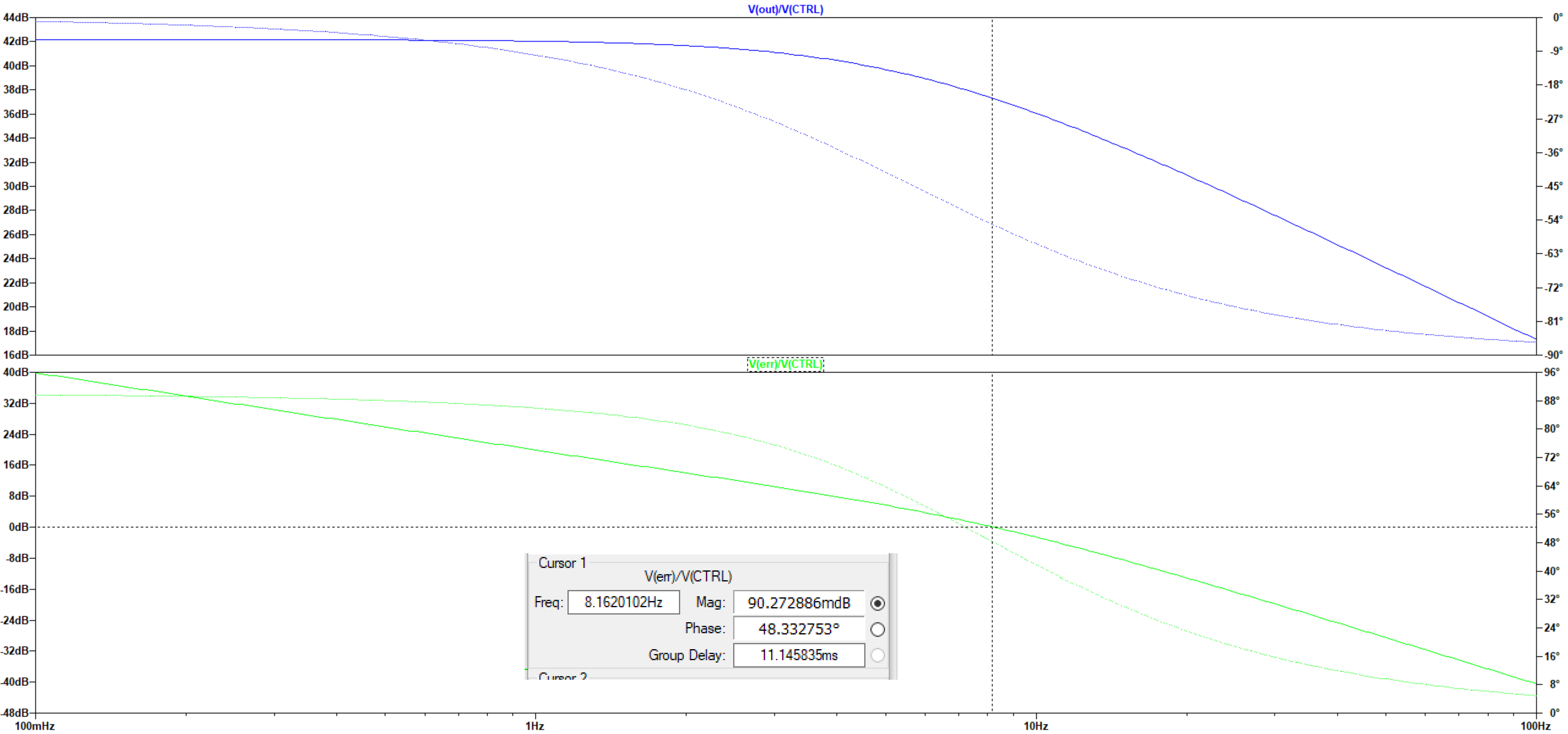



```
.model DPFC D TT=100n Rs=10m Cjo=50p N=0.6
.model DOTA D tt=100n Rs=10m N=100m BV=5
*
.param Vrms=230
.param Pout=1.3k
.param Vout=400
.param RL={Vout**2/Pout}
.param L=600u
.param RBOL=82.5k
.param RBOU=6.6Meg
.param ROCP=3.8k
.param Vp=2.5
.param Rsense=30m
*
*
.param Gfc=34 ; magnitude at crossover *
.param PS=-45 ; phase lag at crossover *
*
* Enter Design Goals Information Here *
*
.param fc=5 ; targeted crossover *
.param PM=60 ; choose phase margin at crossover *
*
*
* Enter the Values for Vout and Bridge Bias Current *
*
.param Ibias=250u
.param Vref=2.5
*
* Do not edit the below lines *
*
.param gm=200u ; transconductance in Siemens *
.param Rlower={Vref/Ibias}
.param Rupper={(Vout-Vref)/Ibias}
.param boost={PM-PS-90}
.param G={10**(-Gfc/20)}
.param kf={tan((boost/2+45)*pi/180)}
.param fp={fc*kf}
.param fz={fc/kf}
.param a={sqrt((fc**2/fp**2)+1)}
.param b={sqrt((fz**2/fc**2)+1)}
.param R2={(a/b)*(fp*G)*(Rlower+Rupper)/((fp-fz)*Rlower*gm)}
.param C1={1/(2*pi*R2*fz)}
.param C2={Rlower*gm*(b/a)/(2*pi*fp*G*(Rlower+Rupper))}
*
.options abstol=1u vntol=1m reltol=0.01 gmin=100p
+method=gear
```

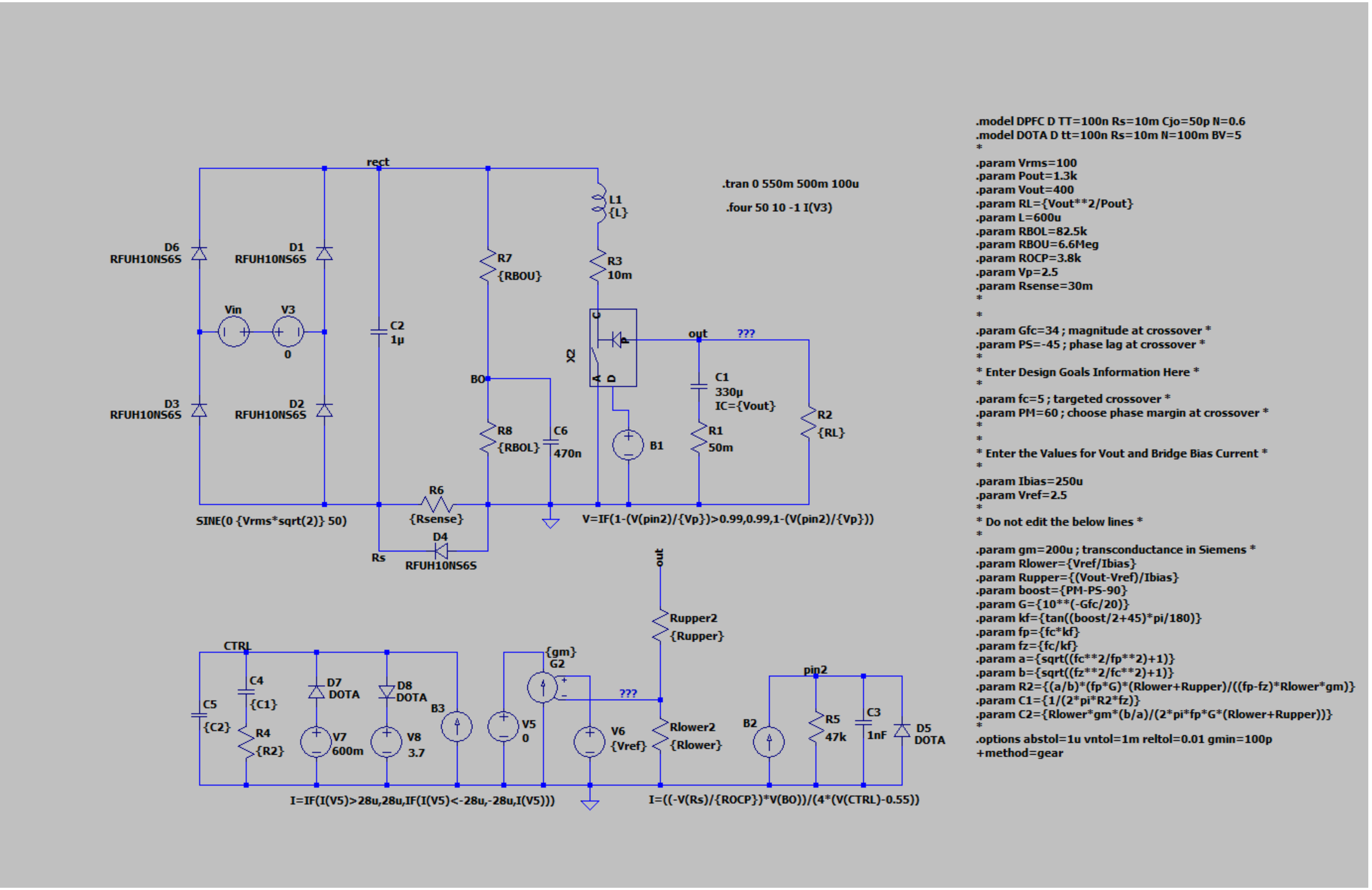
The loop is stabilized for a 4-Hz crossover frequency at 100 V rms. It will naturally increase when V_{in} is 230 V.



The crossover increases to 8 Hz at 230 V rms. The BO sensing helps reducing the variability a little.



The averaged model lends itself well for assessing the current distortion at different line levels



V_{in} = 100 V rms

```
Total Harmonic Distortion:   3.649093%

Total elapsed time: 2.749 seconds.
```

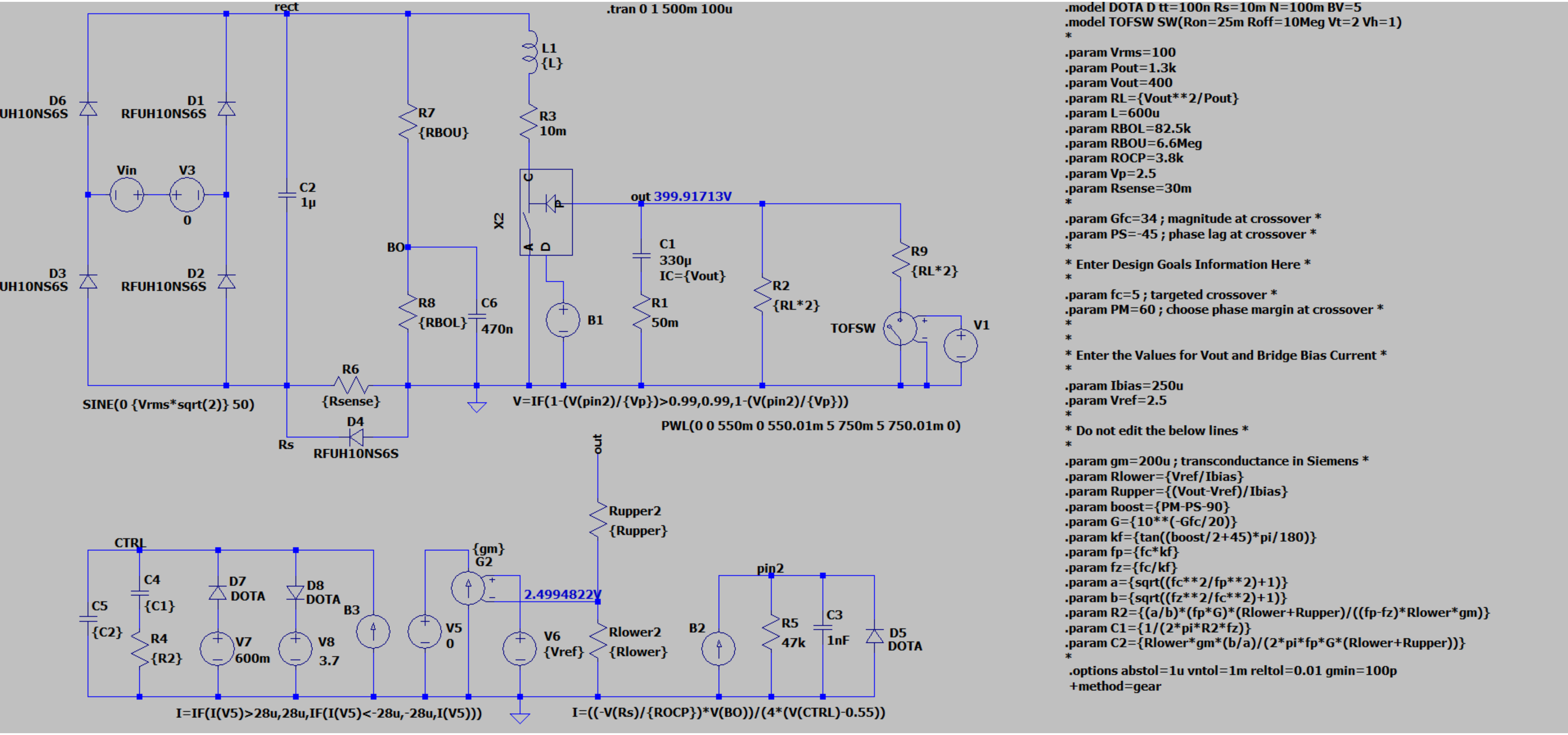
V_{in} = 230 V rms

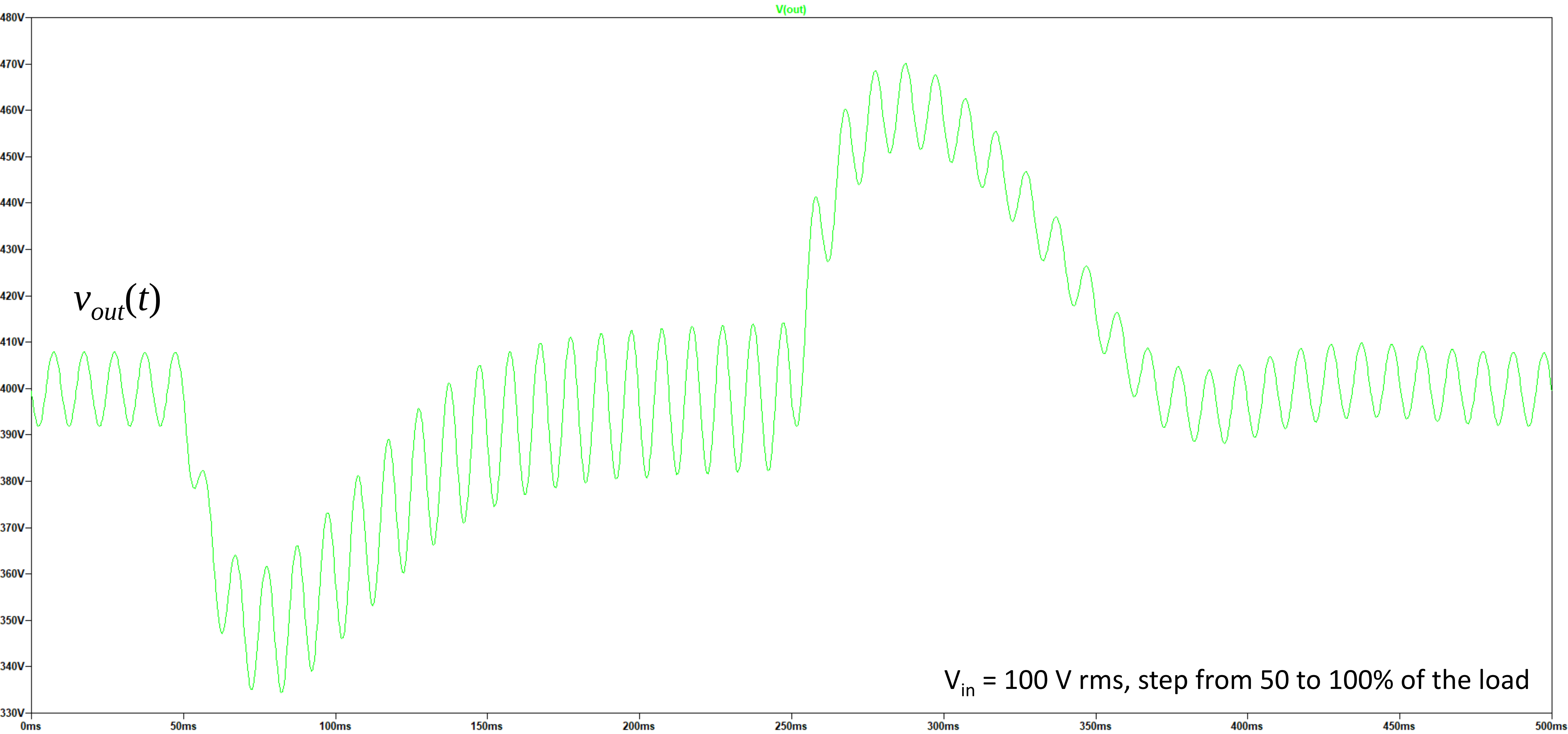
```
Partial Harmonic Distortion: 3.604100%
Total Harmonic Distortion:   3.635482%

Total elapsed time: 7.161 seconds.
```

The average model is useful to assess the low-frequency rms component of the output capacitor:

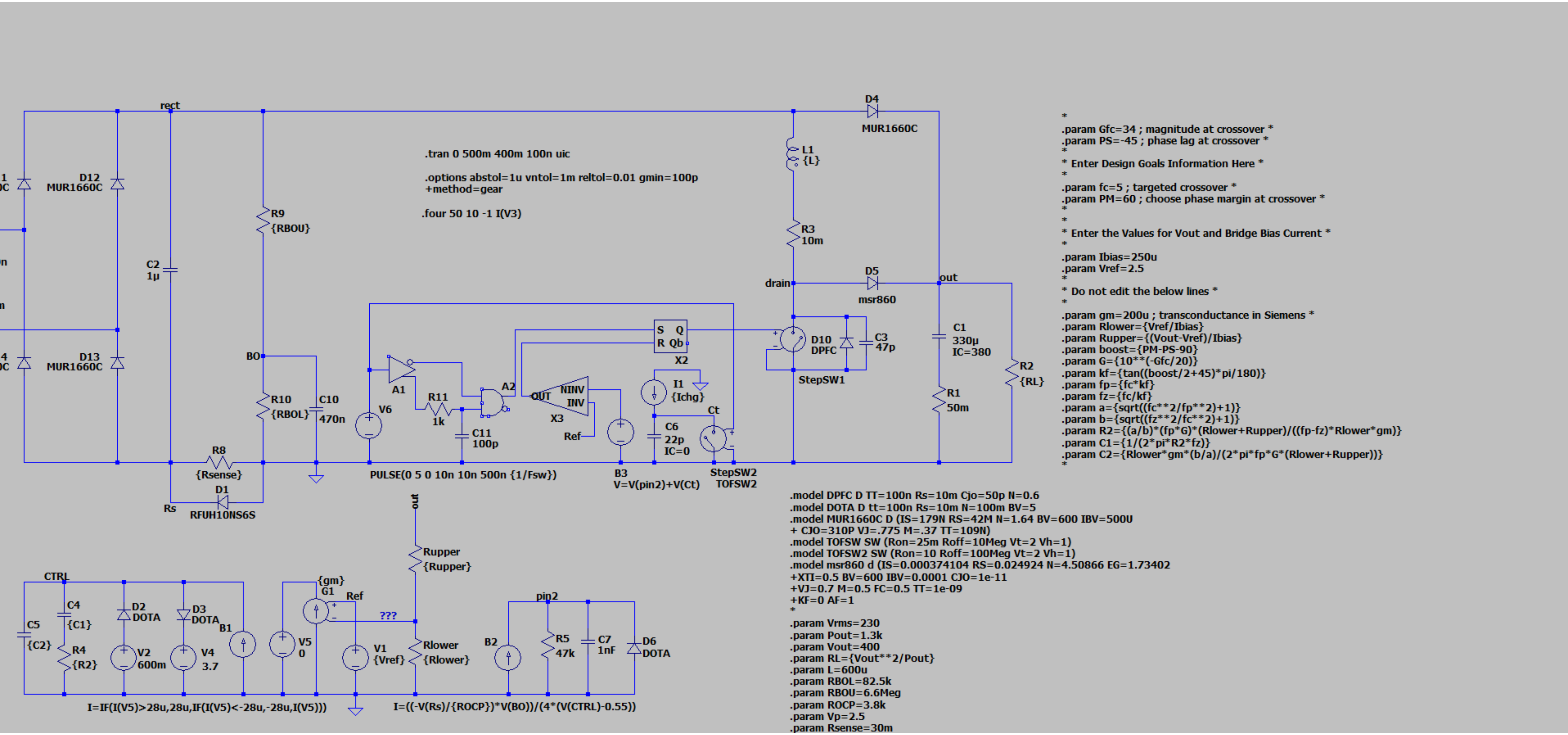
$I_{C,rms}$ = 2.33 A, V_{in} = 100 V rms

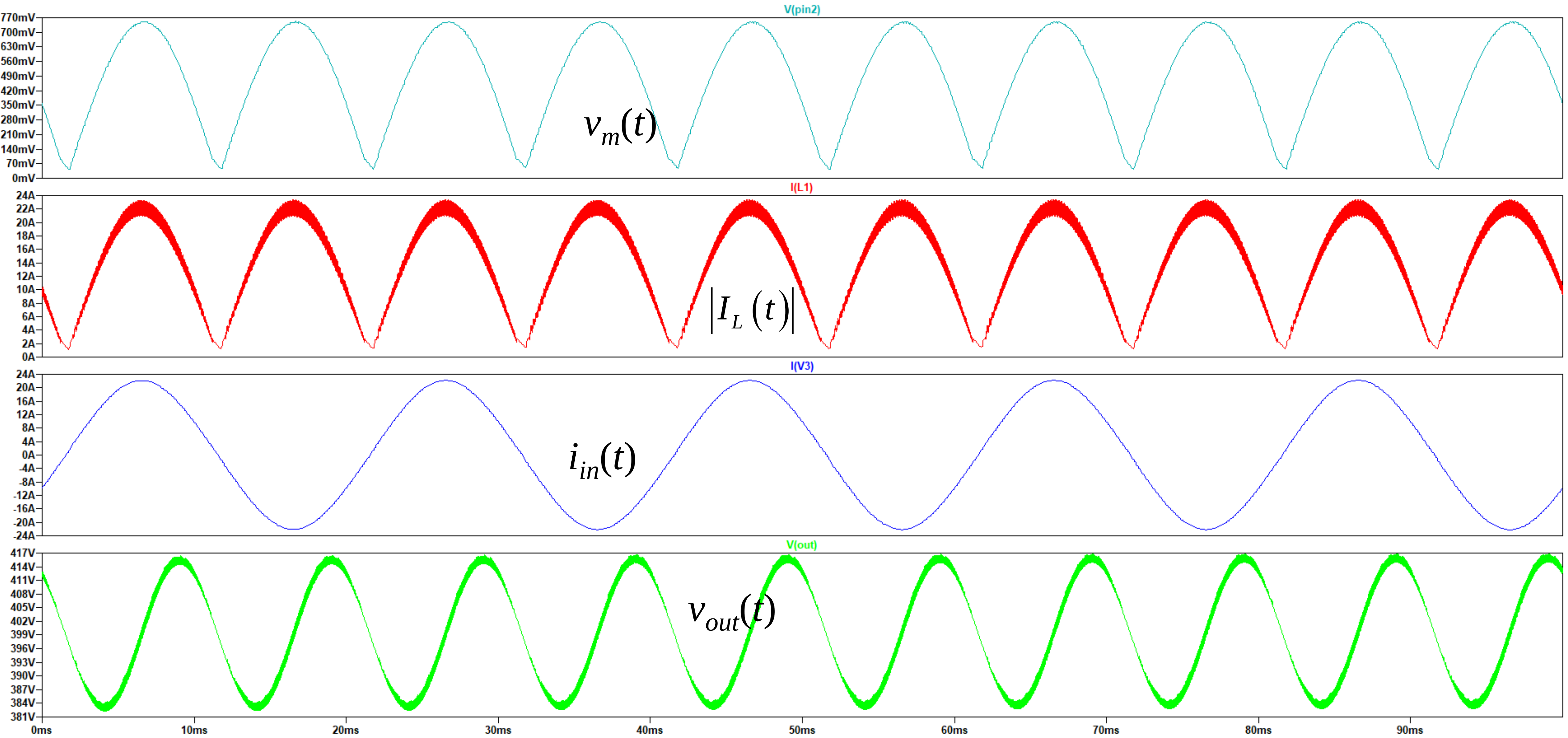
Once the loop is closed, you can test the transient response:




V(out)
480V
470V
460V
450V
440V
430V
420V
410V
400V
390V
380V
370V
360V
350V
340V
330V
0ms
50ms
100ms
150ms
200ms
250ms
300ms
350ms
400ms
450ms
500ms
$v_{out}(t)$
V_{in} = 100 V rms, step from 50 to 100% of the load

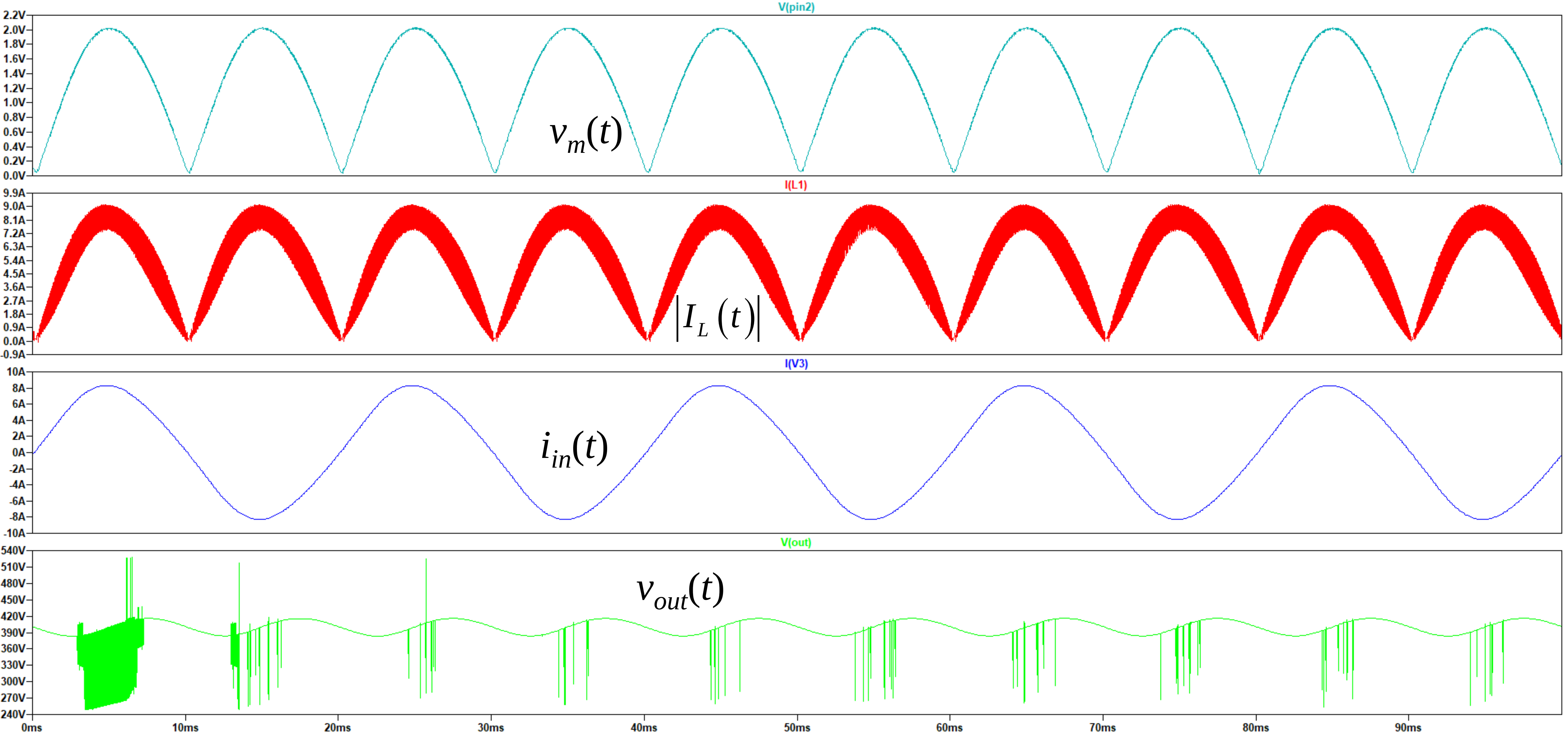
This is the complete cycle-by-cycle CCM PFC operated as in the NCP1654:





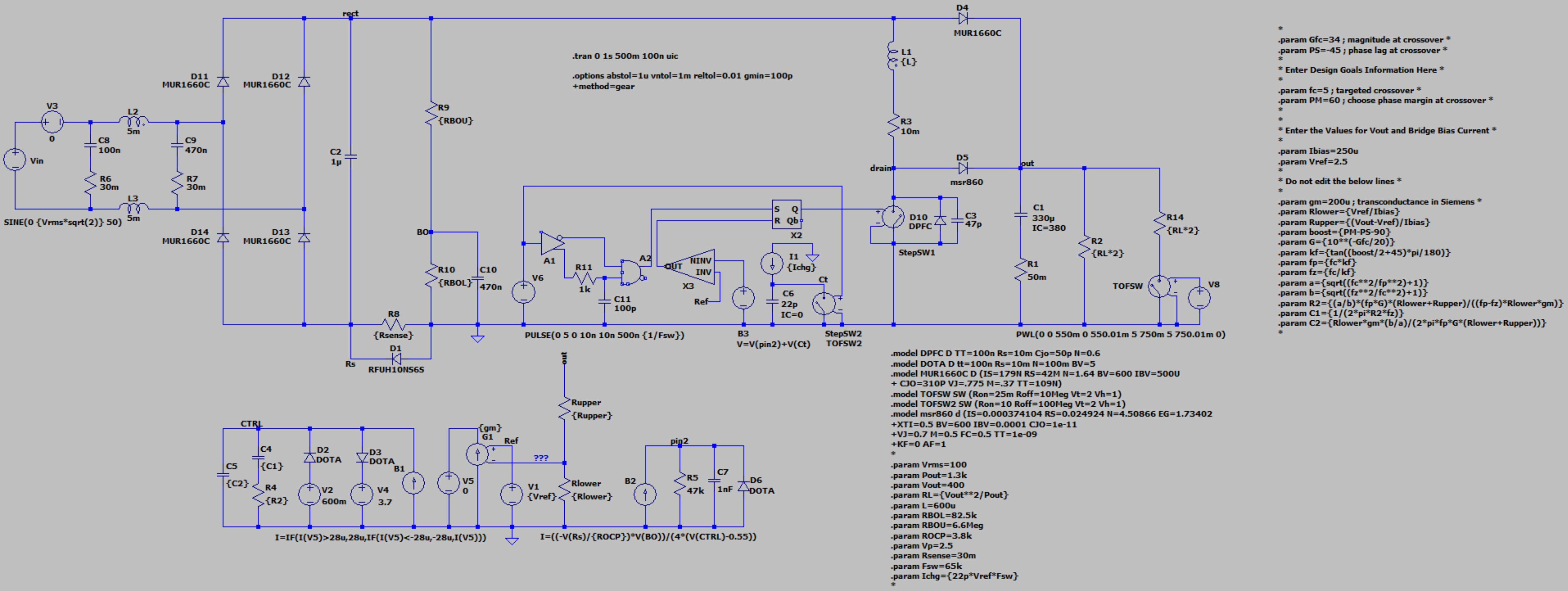
This is the cycle-by-cycle simulation of the 1.3-kW CCM PFC supplied from a 100-V rms input voltage. The THD is better than with the averaged model.

```
10              5.000e+2
Partial Harmonic Distortion: 1.521069%
Total Harmonic Distortion:   1.536531%
```

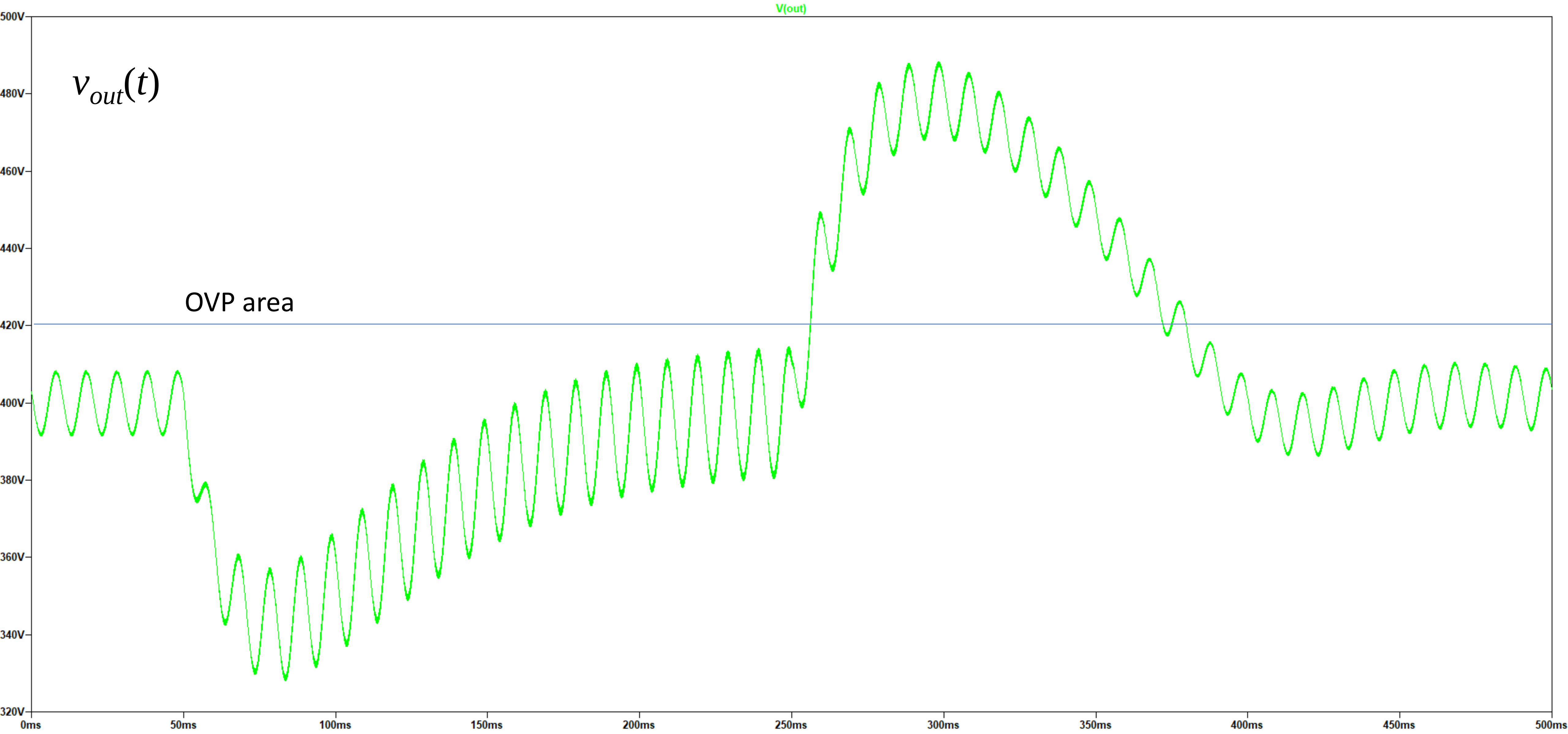


This is the cycle-by-cycle simulation of the 1.3-kW CCM PFC supplied from a 230-V rms input voltage. The THD is better than with the averaged model.

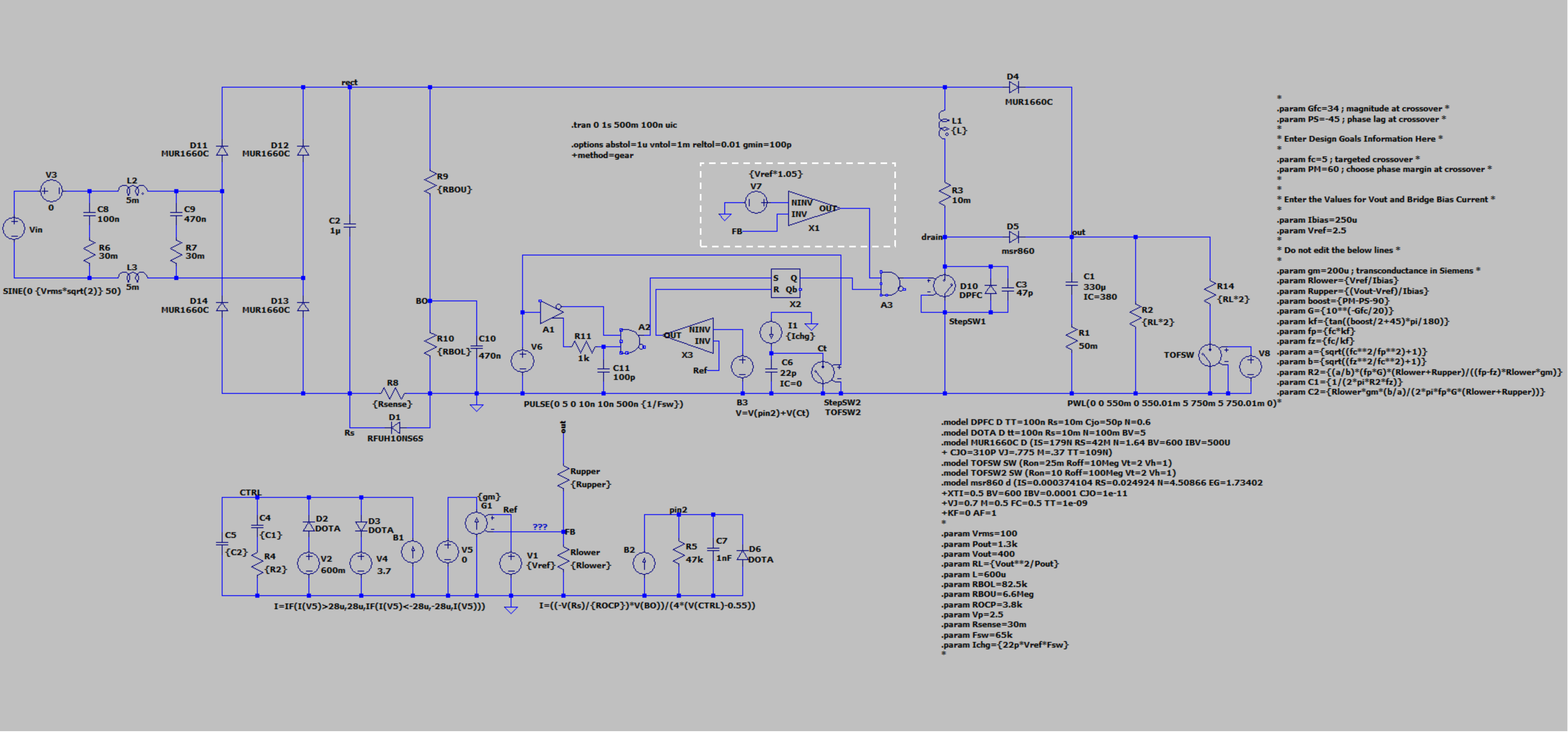
```
Partial Harmonic Distortion: 3.582166%
Total Harmonic Distortion:   3.585183%

Total elapsed time: 179.235 seconds.
```



rect
D4
MUR1660C
.tran 0 1s 500m 100n uic
.options abstol=1u vntol=1m reltol=0.01 gmin=100p
+method=gear
D11
MUR1660C
D12
MUR1660C
V3
0
L2
5m
C8
100n
C9
470n
R6
30m
R7
30m
Vin
L3
5m
SINE(0 {Vrms*sqrt(2)} 50)
D14
MUR1660C
D13
MUR1660C
C2
1µ
R9
{RBOU}
BO
R10
{RBOL}
C10
470n
R8
{Rsense}
D1
RFUH10NS6S
Rs
L1
{L}
R3
10m
drain
D5
msr860
out
A1
V6
R11
1k
C11
100p
A2
X3
OUT
NINV
INV
Ref
S Q
R Qb
X2
I1
{Ichg}
Ct
C6
22p
IC=0
B3
V=V(pin2)+V(Ct)
StepSW2
TOFSW2
D10
DPFC
C3
47p
StepSW1
C1
330µ
IC=380
R1
50m
R2
{RL*2}
R14
{RL*2}
TOFSW
V8
PULSE(0 5 0 10n 10n 500n {1/Fsw})
PWL(0 0 550m 0 550.01m 5 750m 5 750.01m 0)
out
Rupper
{Rupper}
CTRL
C5
{C2}
C4
{C1}
R4
{R2}
D2
DOTA
D3
DOTA
V2
600m
V4
3.7
B1
V5
0
{gm}
G1
Ref
???
V1
{Vref}
Rlower
{Rlower}
pin2
B2
R5
47k
C7
1nF
D6
DOTA
I=IF(I(V5)>28u,28u,IF(I(V5)<-28u,-28u,I(V5)))
I=((-V(Rs)/{ROCP})*V(BO))/(4*(V(CTRL)-0.55))
.model DPFC D TT=100n Rs=10m Cjo=50p N=0.6
.model DOTA D tt=100n Rs=10m N=100m BV=5
.model MUR1660C D (IS=179N RS=42M N=1.64 BV=600 IBV=500U
+ CJO=310P VJ=.775 M=.37 TT=109N)
.model TOFSW SW (Ron=25m Roff=10Meg Vt=2 Vh=1)
.model TOFSW2 SW (Ron=10 Roff=100Meg Vt=2 Vh=1)
.model msr860 d (IS=0.000374104 RS=0.024924 N=4.50866 EG=1.73402
+XTI=0.5 BV=600 IBV=0.0001 CJO=1e-11
+VJ=0.7 M=0.5 FC=0.5 TT=1e-09
+KF=0 AF=1
*
.param Vrms=100
.param Pout=1.3k
.param Vout=400
.param RL={Vout**2/Pout}
.param L=600u
.param RBOL=82.5k
.param RBOU=6.6Meg
.param ROCP=3.8k
.param Vp=2.5
.param Rsense=30m
.param Fsw=65k
.param Ichg={22p*Vref*Fsw}
*
*
.param Gfc=34 ; magnitude at crossover *
.param PS=-45 ; phase lag at crossover *
*
* Enter Design Goals Information Here *
*
.param fc=5 ; targeted crossover *
.param PM=60 ; choose phase margin at crossover *
*
*
* Enter the Values for Vout and Bridge Bias Current *
*
.param Ibias=250u
.param Vref=2.5
*
* Do not edit the below lines *
*
.param gm=200u ; transconductance in Siemens *
.param Rlower={Vref/Ibias}
.param Rupper={(Vout-Vref)/Ibias}
.param boost={PM-PS-90}
.param G={10**(-Gfc/20)}
.param kf={tan((boost/2+45)*pi/180)}
.param fp={fc*kf}
.param fz={fc/kf}
.param a={sqrt((fc**2/fp**2)+1)}
.param b={sqrt((fz**2/fc**2)+1)}
.param R2={(a/b)*(fp*G)*(Rlower+Rupper)/((fp-fz)*Rlower*gm)}
.param C1={1/(2*pi*R2*fz)}
.param C2={Rlower*gm*(b/a)/(2*pi*fp*G*(Rlower+Rupper))}
*



V(out)
$v_{out}(t)$
OVP area
500V
480V
460V
440V
420V
400V
380V
360V
340V
320V
0ms
50ms
100ms
150ms
200ms
250ms
300ms
350ms
400ms
450ms
500ms



An OVP circuitry is added: it interrupts the driving pulses as soon as the output voltage exceeds the nominal value by 5%.

Another simple predictive control law was described by Dr. Sam Ben-Yaakov in a paper he published in 1998, PWM Converters with Resistive Input. The principle lies in generating an off-time control proportional to the average inductor current:

In a PFC

$$\langle i_L(t)\rangle = i_{in}(t) \propto v_{in}(t)$$

$$\frac{V_{out}}{V_{in}} = \frac{1}{1-D} = \frac{1}{d_{off}}$$

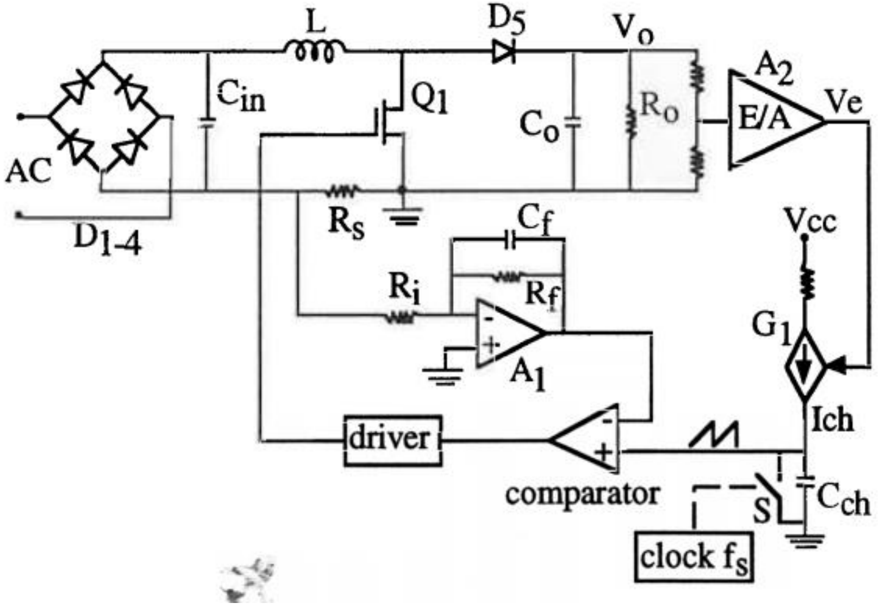
Applying V.s balance:

$$\longrightarrow d_{off}(t) = \frac{v_{in}(t)}{V_{out}} \Rightarrow d_{off}(t) \propto \langle i_L(t)\rangle$$

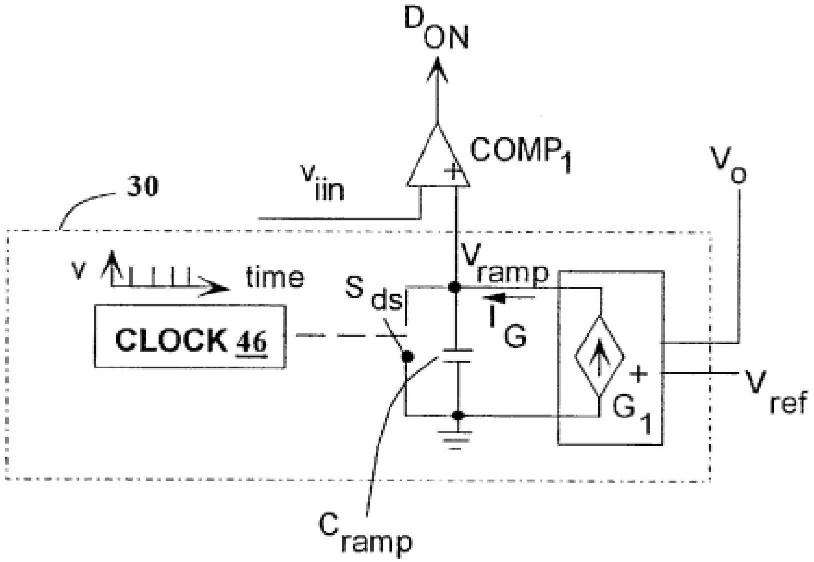
Ensures a sinusoidal absorption

That is, a resistive input will be observed if D_{off} is programmed according to the rule:

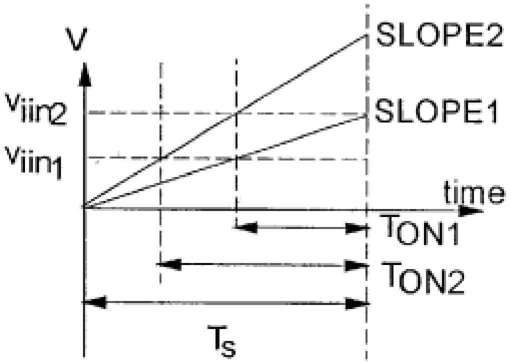
$$D_{off} = \underbrace{\boxed{\left(\frac{R_e}{V_o(av)}\right)}}_{\text{constant}} I_L(av) \quad , \; 0 < D_{off} < 1 \qquad (4)$$



The original patent dates from October 2001 and describes a claim linked the charging method of capacitor C_{ramp} based on the error voltage.

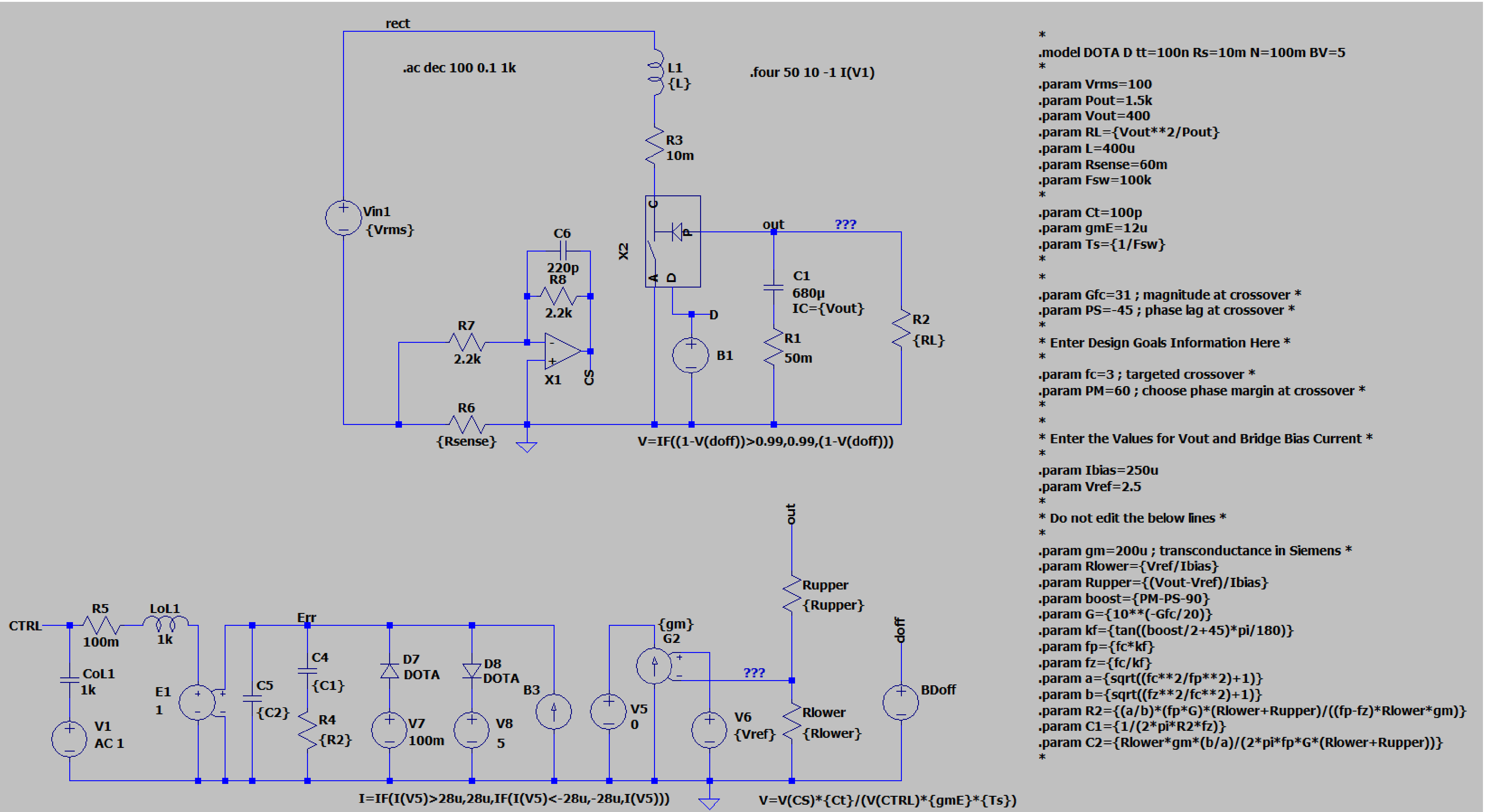


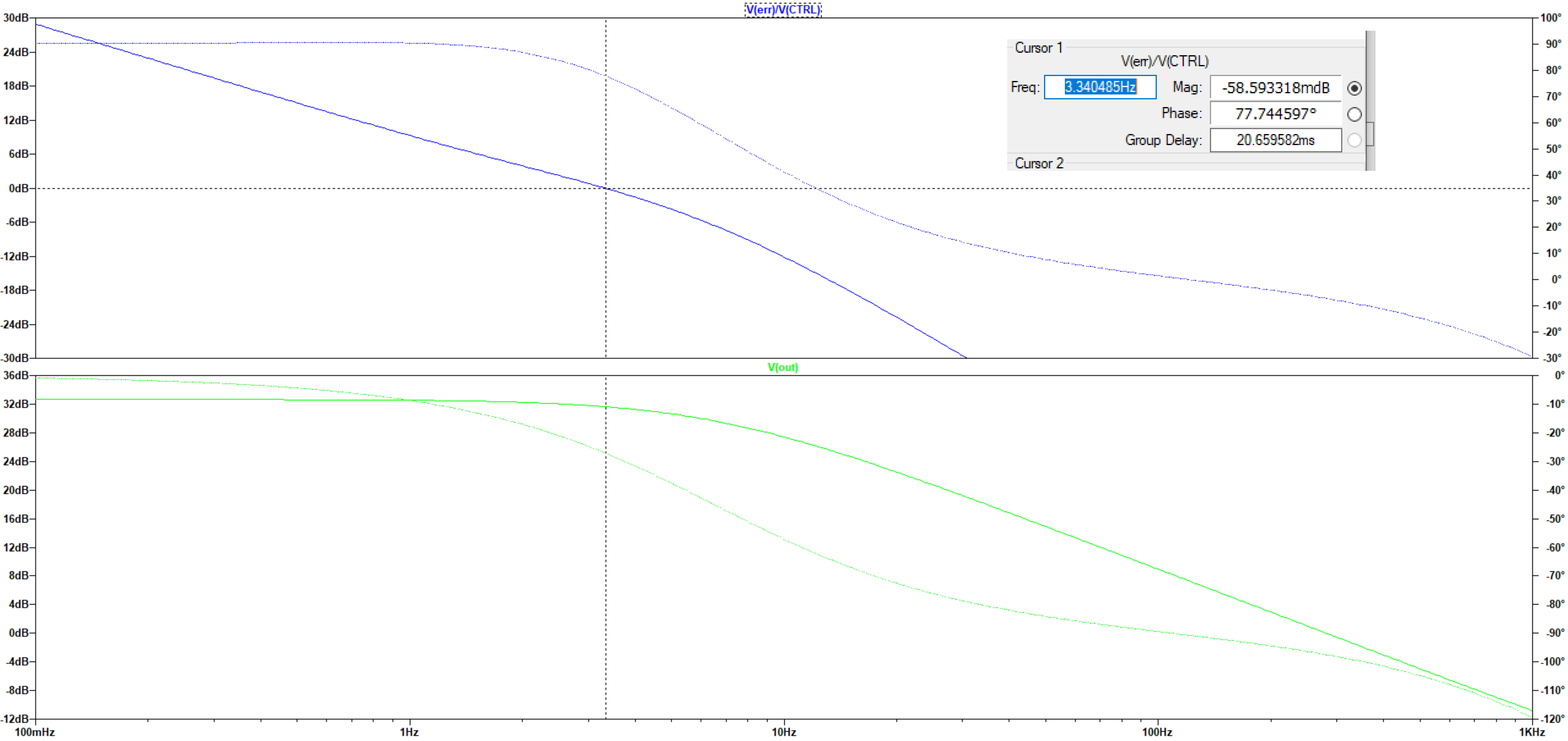
Capacitor voltage



6,307,361 (Oct 2001) and 6,728,121 (April2004)

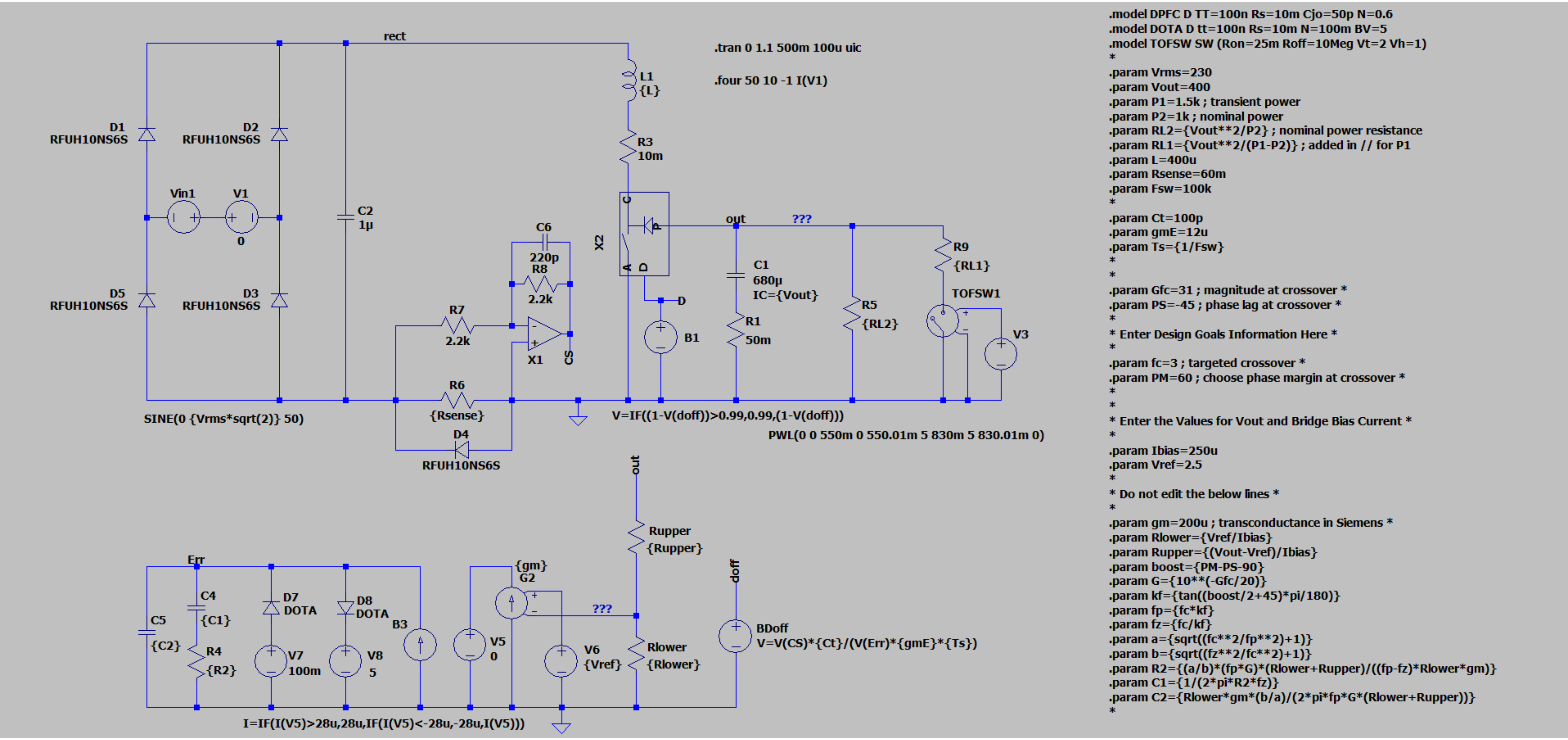
I built a large-signal averaged model for determining the voltage-loop response and stabilize the converter.



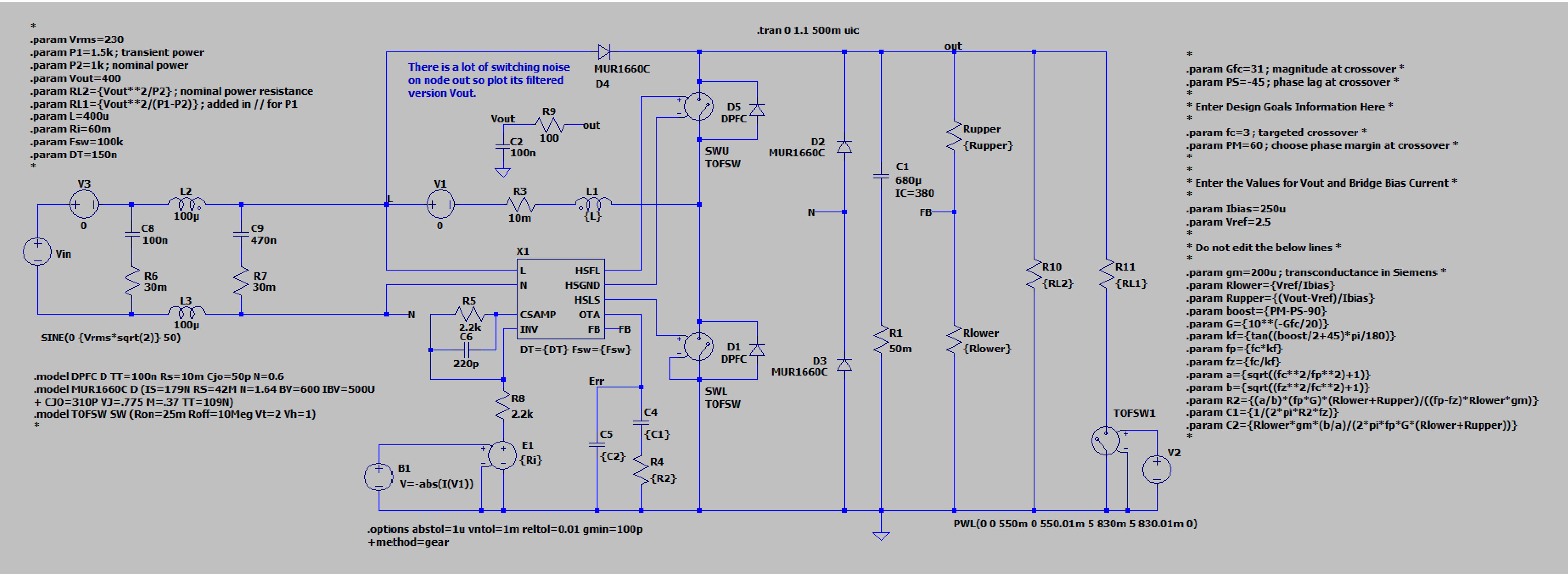


The voltage loop is stabilized with a 3.3-Hz crossover frequency and a comfortable phase margin.

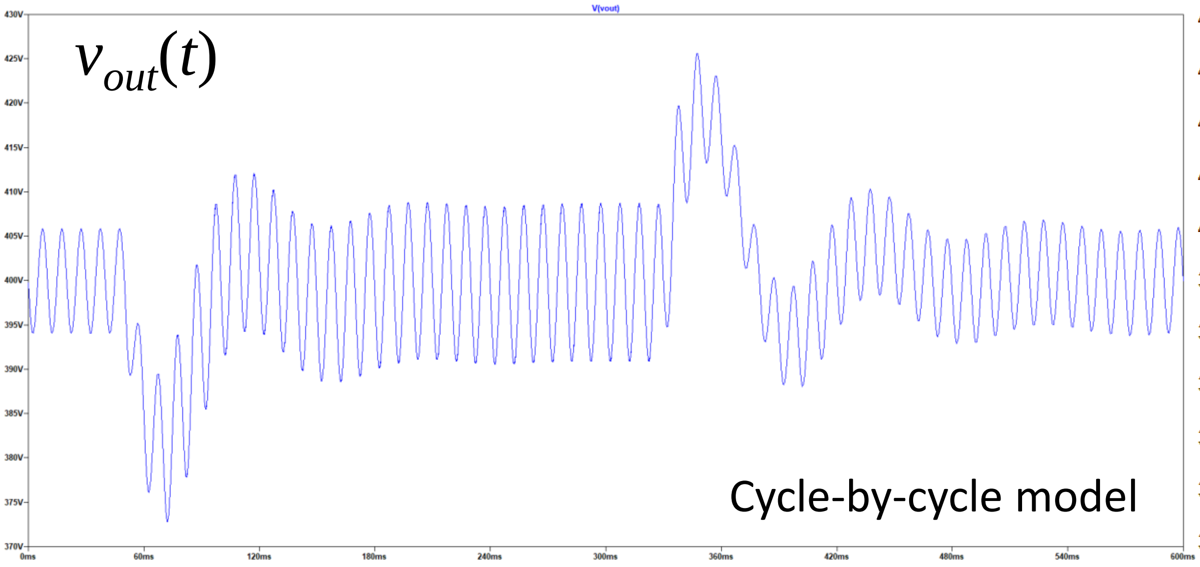
The large-signal model can be used to check the transient response once the loop has been stabilized. Averaged models are interesting in evaluating PFCs as they simulate very fast considering the absence of switching component.

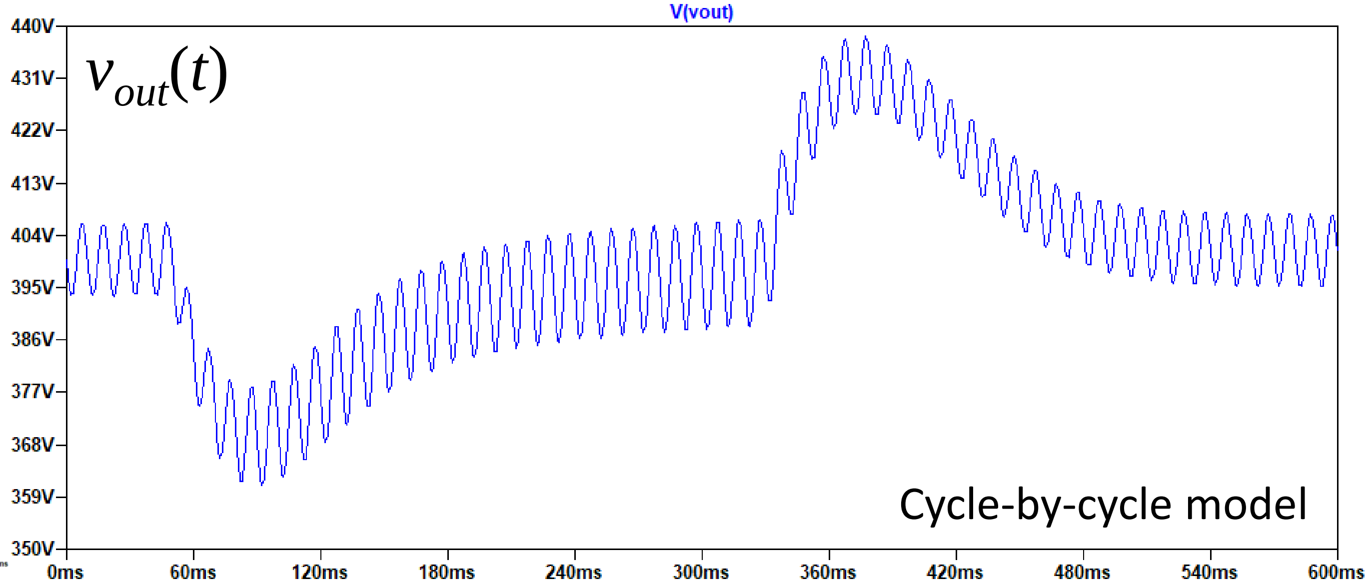


I then built a cycle-by-cycle version to check the loop was well stabilized and I compared the step response with that of the averaged model:



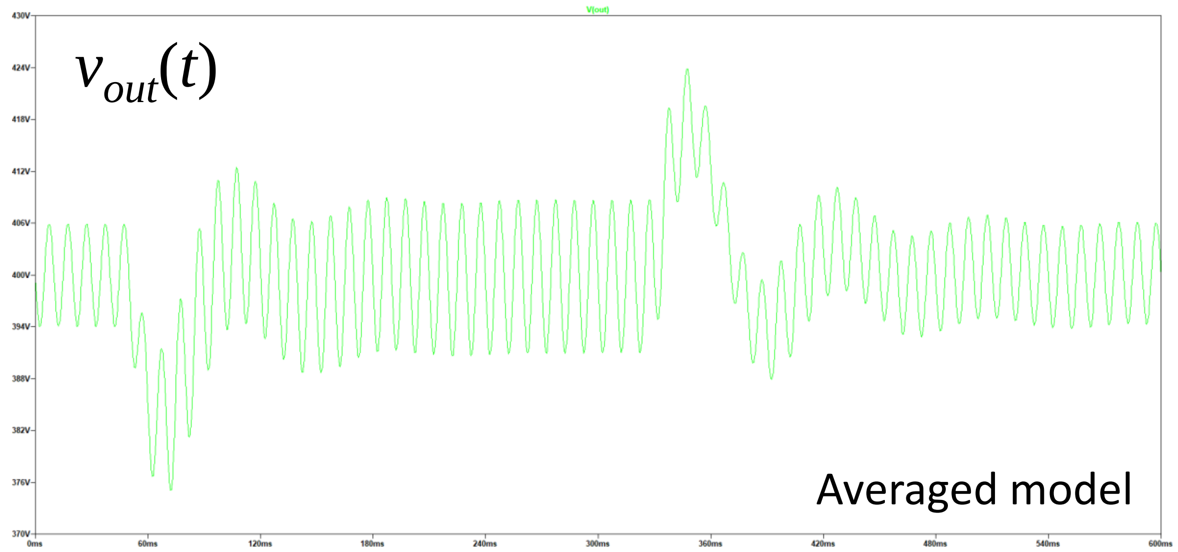
It is a TPPFC circuit where the inductor current is observed via the dummy 0-V source V_1 then shaped in a positive-only waveform via B_1.

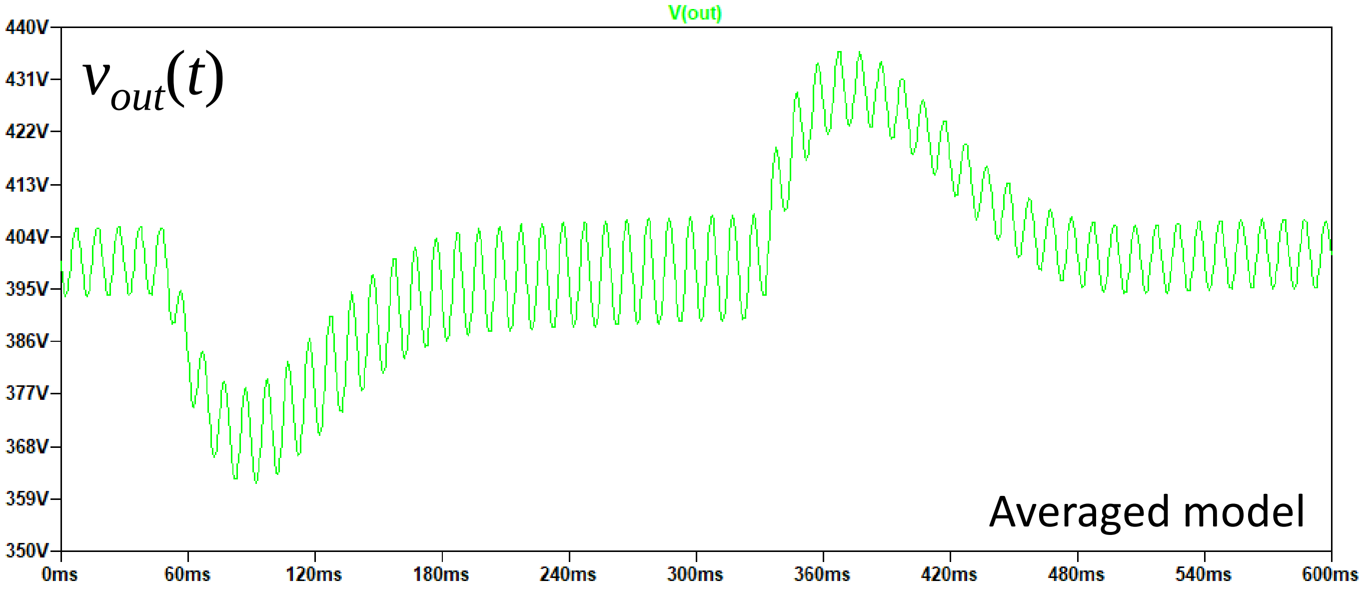




The output power is stepped from 1 kW to 1.5 kW with a 230-V input voltage

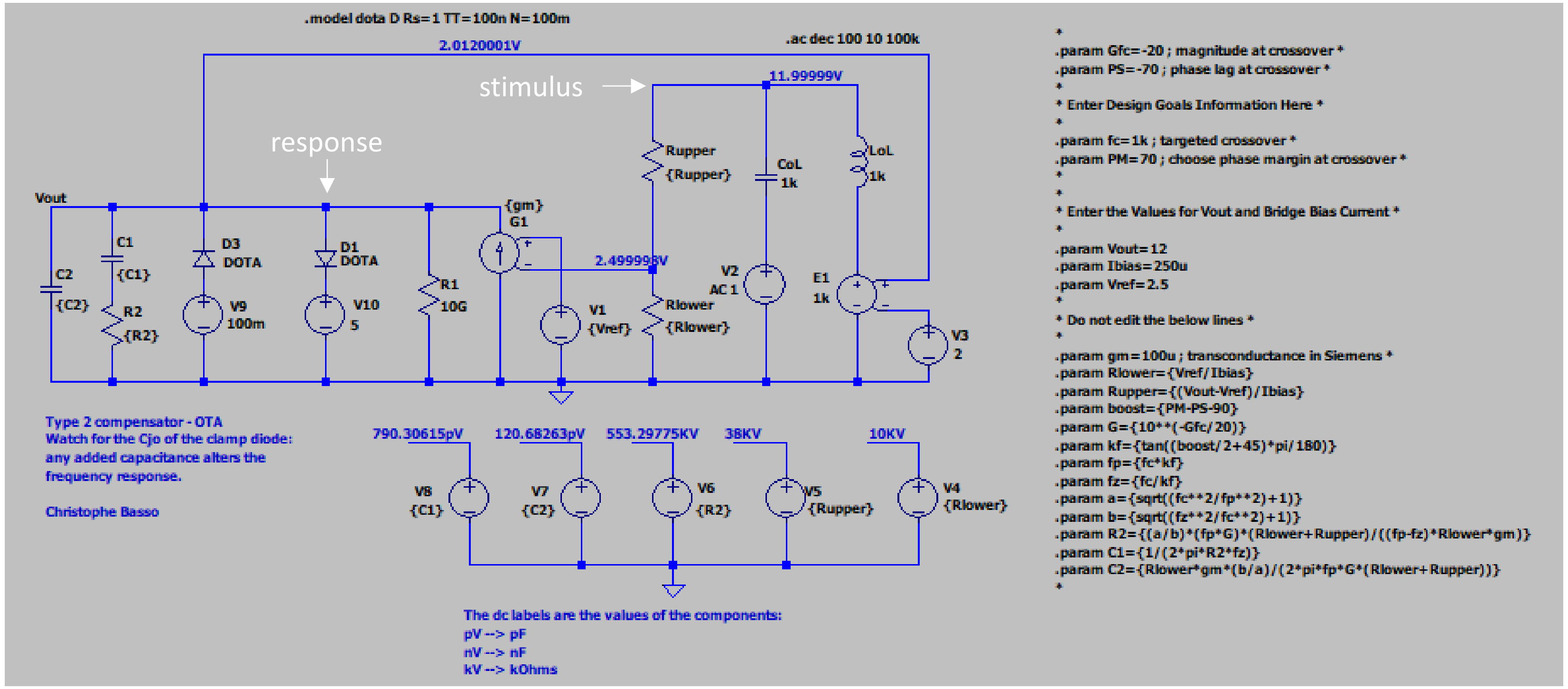
The output power is stepped from 1 kW to 1.5 kW with a 100-V input voltage

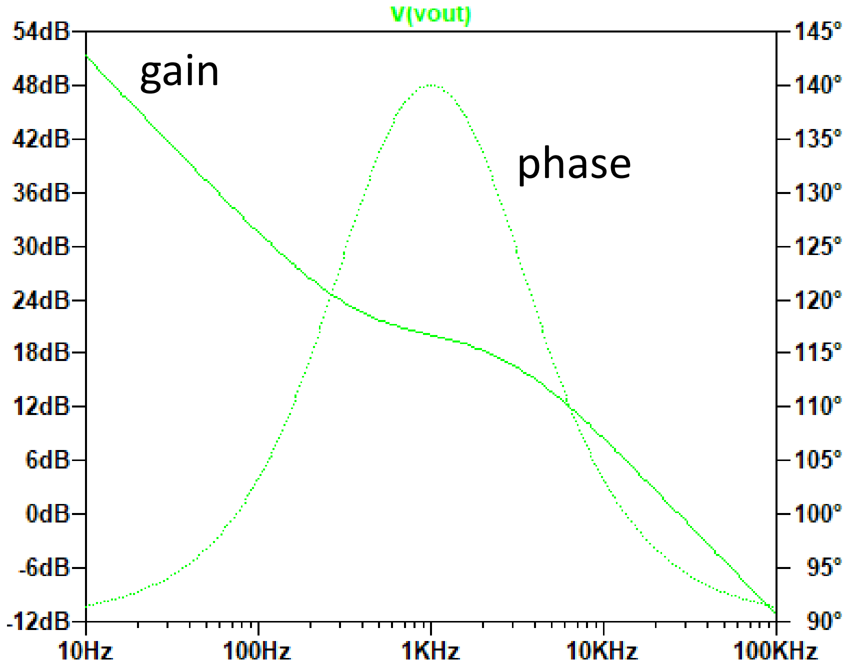




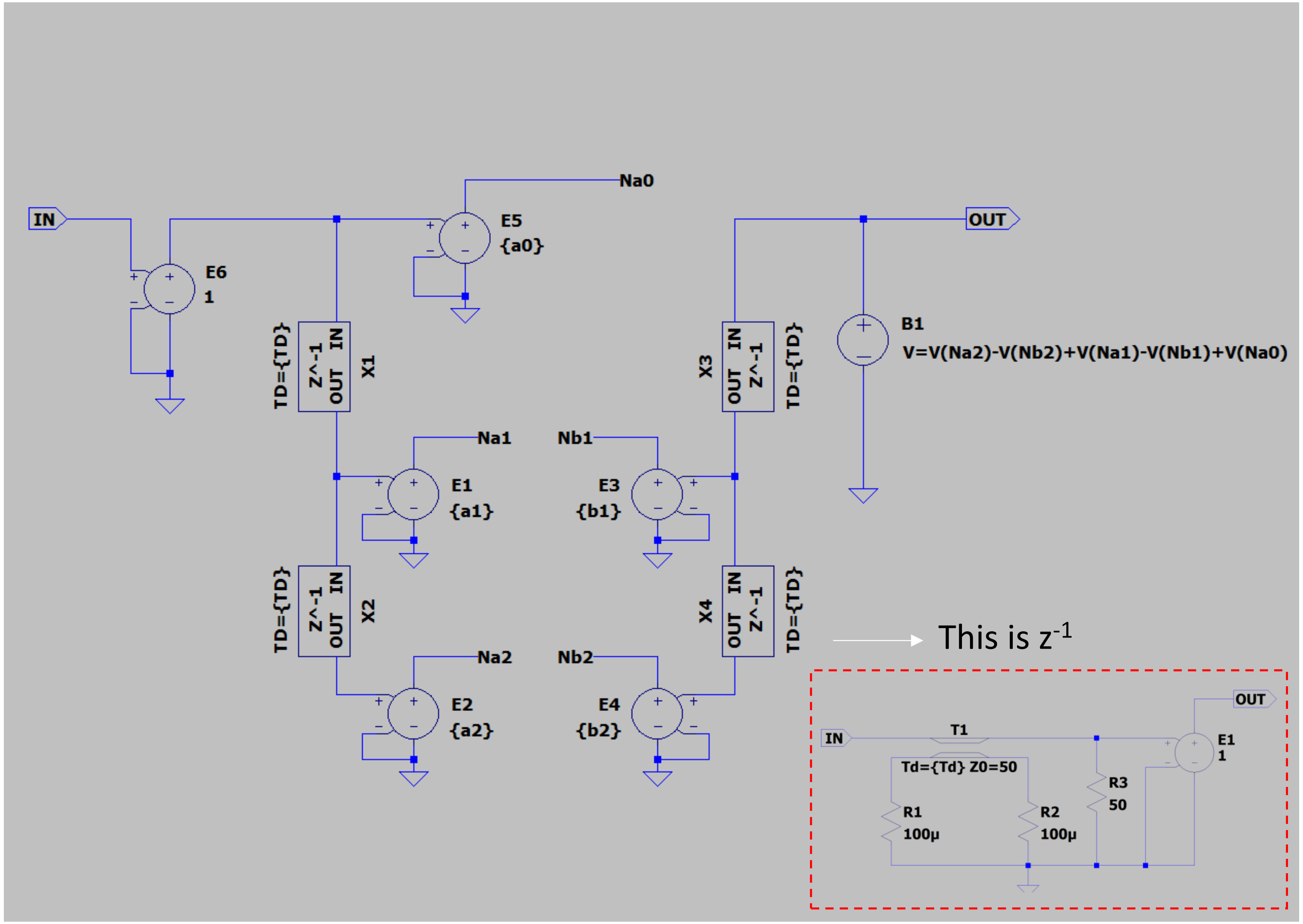
Many new integrated circuits now include a digital filter for compensating the loop. For instance, in a PFC controller, where the crossover frequency is set to a few hertz, building the filter in the analog way, would imply large time constants. These time constants would require resistances of high value considering the small capacitance one can integrate in a die. Hard-wiring a digital filter, on the other hand, represents a viable option, especially if one wants to dynamically change the mid-band gain in relationship with the input voltage and keep an almost constant crossover frequency at high- and low-line condition.

In a PFC circuit, the power stage often exhibits an excess of gain in the vicinity of 5-10 Hz, implying a compensator providing an attenuation at this frequency. The analog version of the compensator can be thought of an operational transconductance amplifier (OTA) driving a network setting the time constants for two poles and the zero. Below is what can be simulated in LTspice (files are here):





A digital version of a type 2 can be assembled with delay lines which have a SPICE primitive. You can check slide 103 of last APEC 2021 seminar.



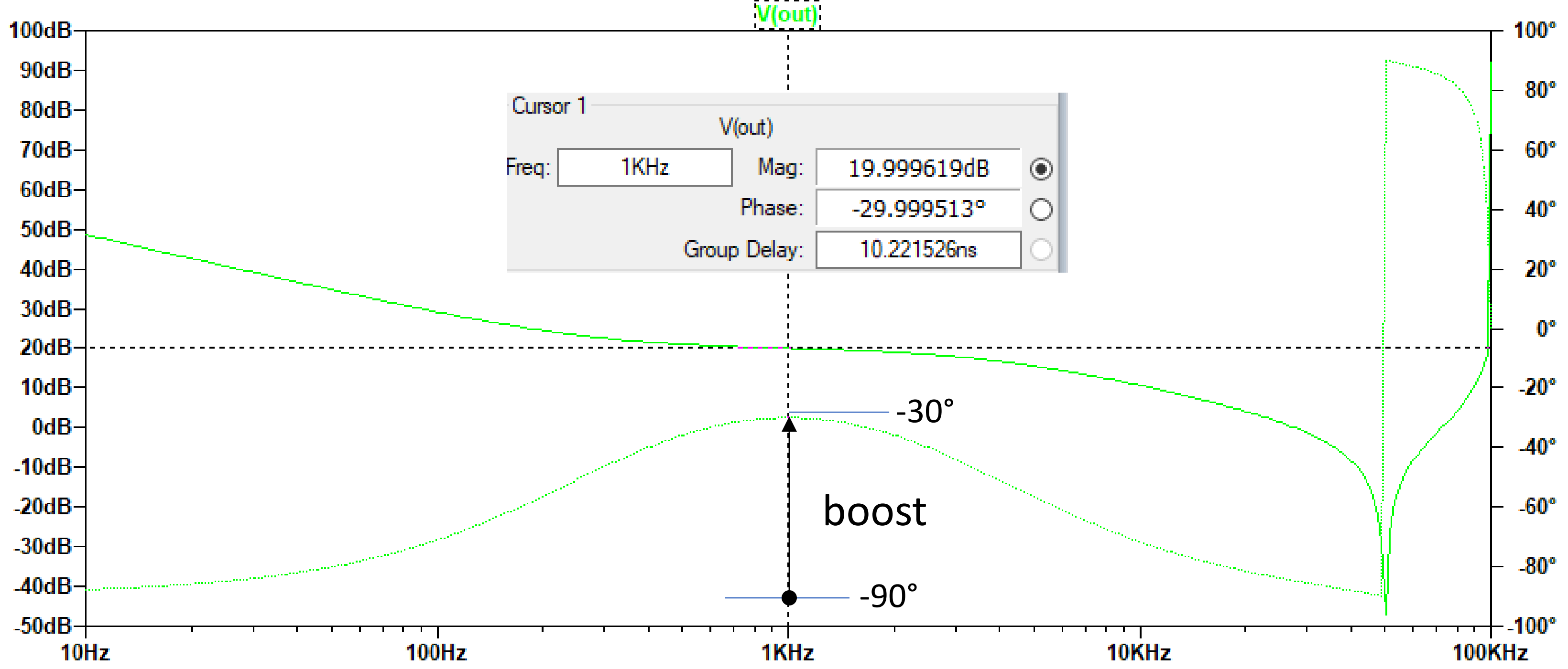
The coefficients determination is automated in the right-side macro

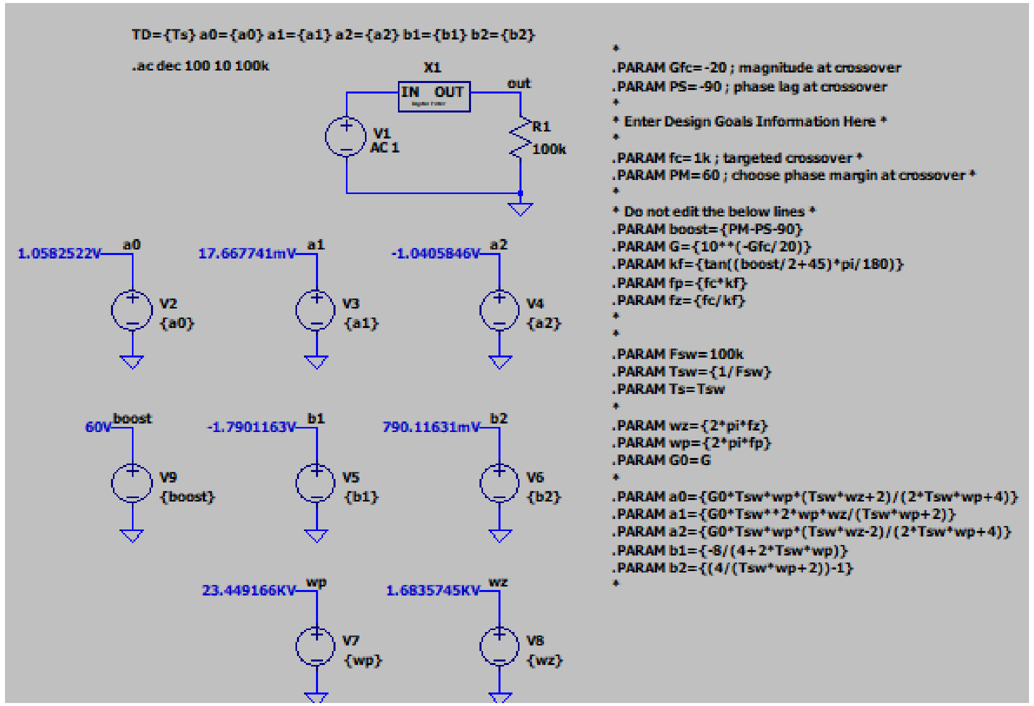
```
*
.param Vin=100 ; input voltage
.param Vout=400 ; regulated output voltage
*
*
.param Gfc=34 ; magnitude at crossover *
.param PS=-50 ; phase lag at crossover *
*
* Enter Design Goals Information Here *
*
.param fc=10 ; targeted crossover *
.param PM=60 ; choose phase margin at crossover *
*
*
.param Ibias=250u
.param Vref=2.5
*
* Do not edit the below lines *
*
.param Rlower={Vref/Ibias}
.param Rupper={(Vout-Vref)/Ibias}
*
* Do not edit the below lines *
*
.param boost={PM-PS-90}
.param G1={10**(-Gfc/20)*(Rlower+Rupper)/Rlower}
.param kf={tan((boost/2+45)*pi/180)}
.param fp={fc*kf}
.param fz={fc/kf}
.PARAM G2={IF(Vin<150,G1,G1/4)}
*
*
.PARAM Fsw=10k ; internal sampling frequency
.PARAM Tsw={1/Fsw}
.PARAM Ts=Tsw
.PARAM Fline=50 ; line frequency
.PARAM TsN=1/(4*Fline) ; moving average sampling freq.
*
.PARAM wz={2*pi*fz}
.PARAM wp={2*pi*fp}
.PARAM G0=G2
*
.PARAM a0={G0*Tsw*wp*(Tsw*wz+2)/(2*Tsw*wp+4)}
.PARAM a1={G0*Tsw**2*wp*wz/(Tsw*wp+2)}
.PARAM a2={G0*Tsw*wp*(Tsw*wz-2)/(2*Tsw*wp+4)}
.PARAM b1={-8/(4+2*Tsw*wp)}
.PARAM b2={(4/(Tsw*wp+2))-1}
*
```

No virtual ground

Gain change with input line

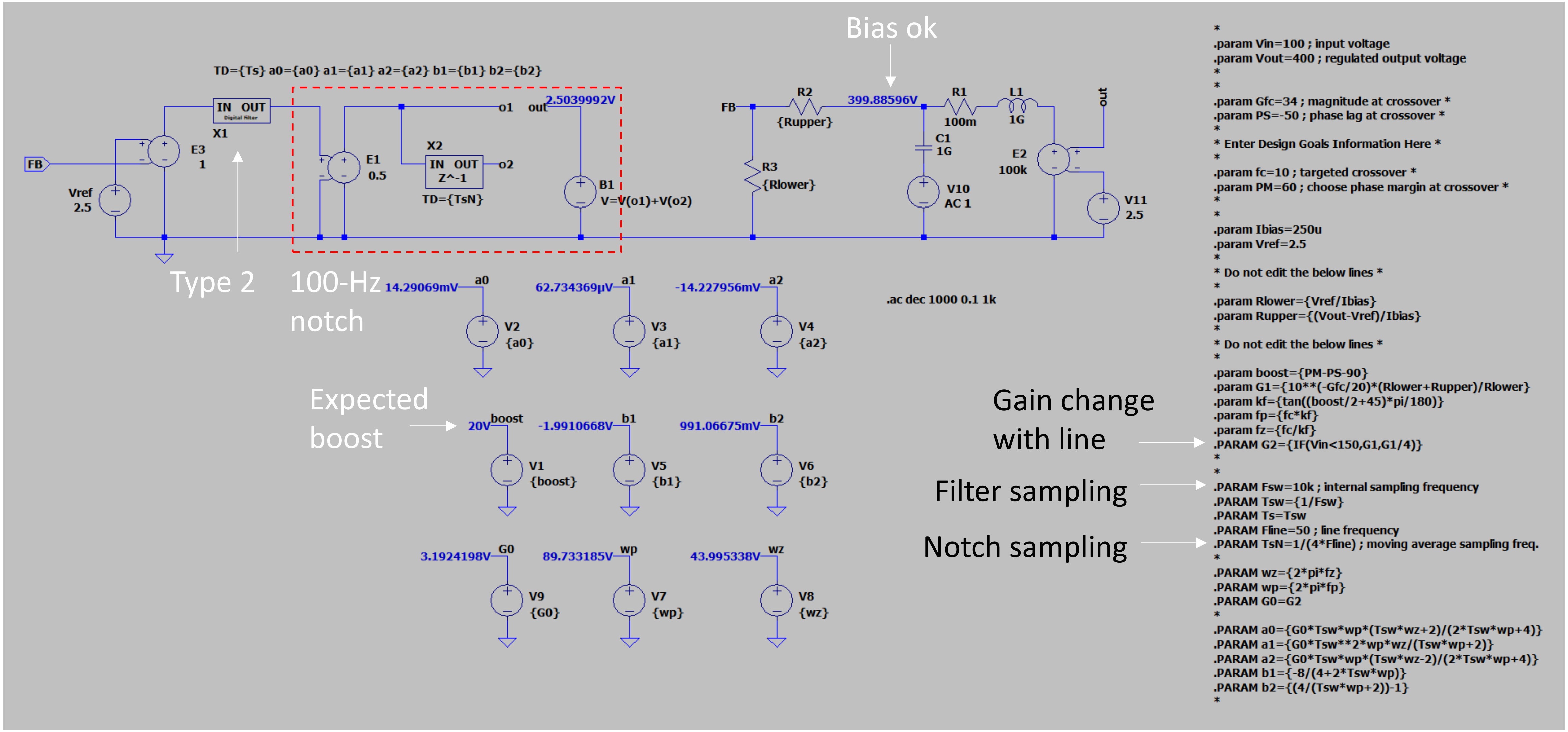
Example for a PFC



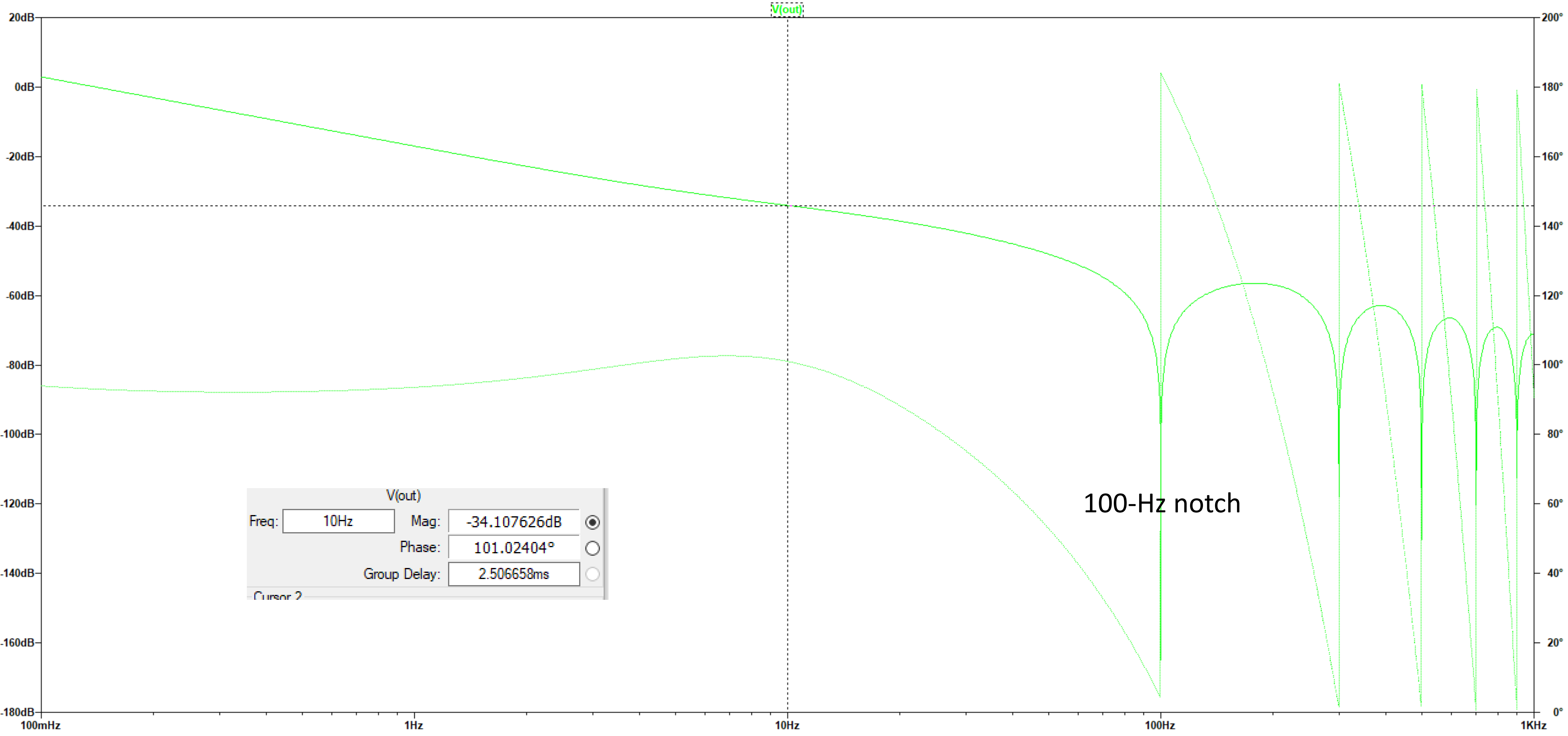


This is the type 2 response with a sampling frequency of 100 kHz. The coefficients are calculated to bring a 20-dB gain at 1 kHz with a phase boost of 60°. This is exactly what the Bode plot shows with a phase of -30° corresponding to the 60° boost you want. The dc sources give the calculated coefficients as LTspice, unfortunately, does not generate a file in which the calculated parameters could be displayed.

In a PFC circuit, the 100- or 120-Hz ripple shows up on the error signal which controls the duty ratio. Too much of ripple – if you push the crossover frequency – induces higher distortion. One way to counteract this phenomenon, is to include a notch filter set at 100 or 120 Hz. It can be done by implementing a moving-average filter whose sampling frequency is 4 times the line frequency (200 Hz or 240 Hz):

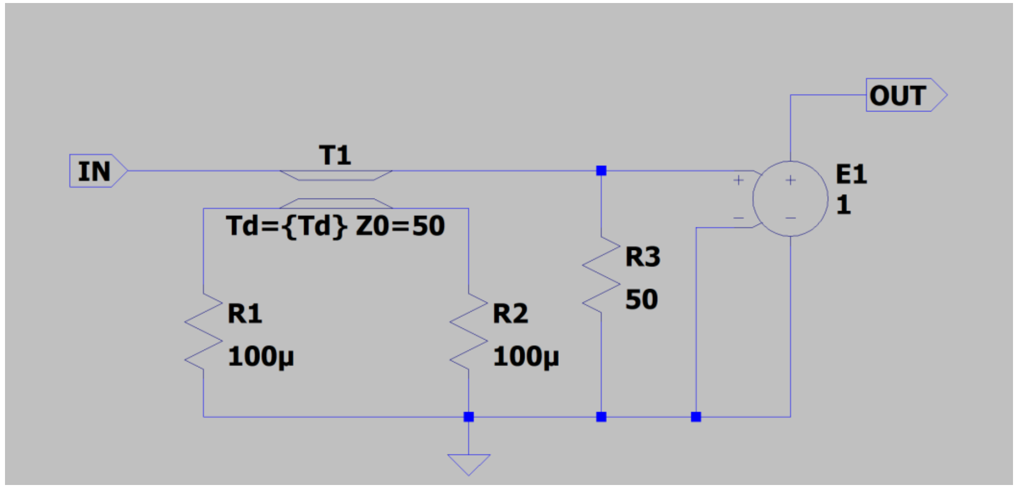


The notch rejects the 100-Hz components but erodes the phase boost by 10°

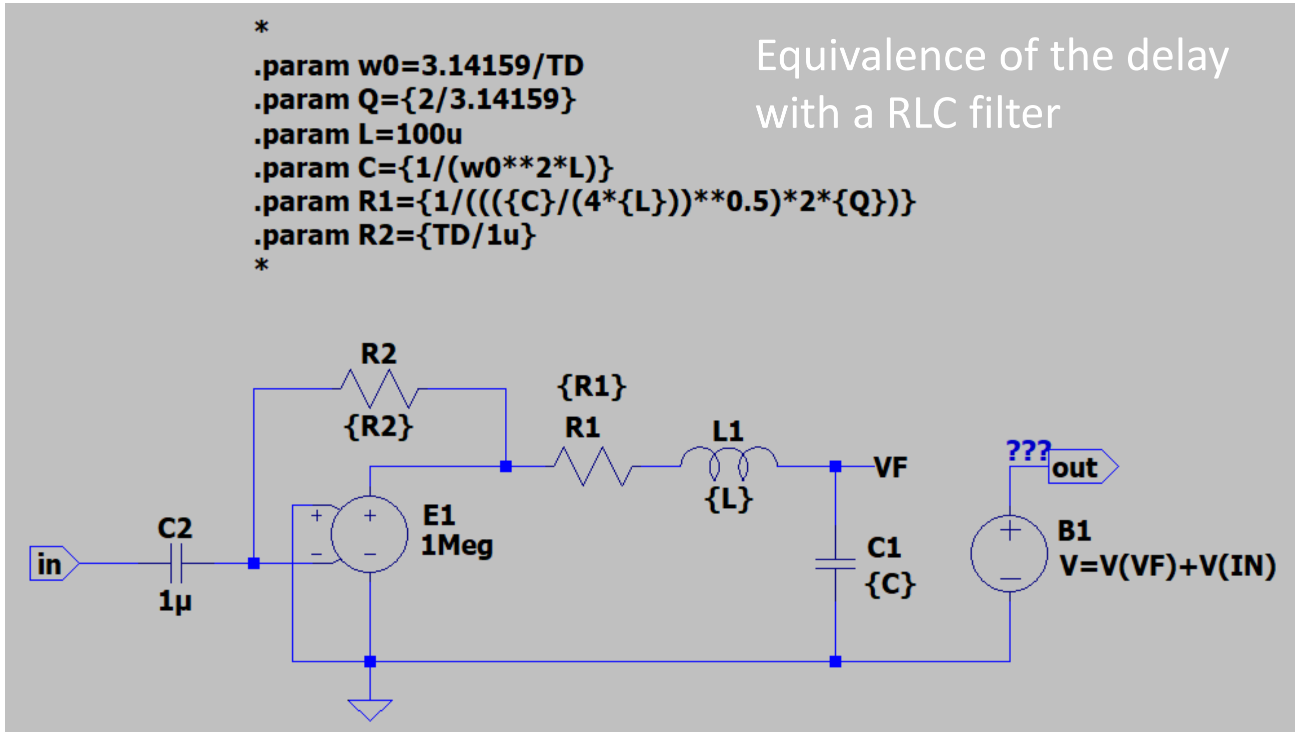


The issue with the delay line is the computational burden it brings on the SPICE engine. And when you simulate PFCs, it dramatically increases the simulation time. One option to attenuate this effect, is to get rid of the delay line and replace it with an equivalent circuit. The 1st-order Padé approximant is an option but I have adopted a different one, consisting of a second-order polynomial:

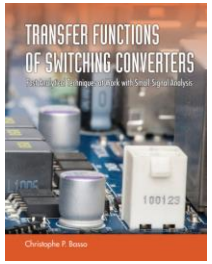
$$H(s) = 1 - \frac{s \cdot t_D}{1 + \frac{s}{\omega Q} + \left(\frac{s}{\omega}\right)^2} \qquad Q = \frac{2}{\pi} \qquad \omega = \frac{\pi}{t_D}$$



≈

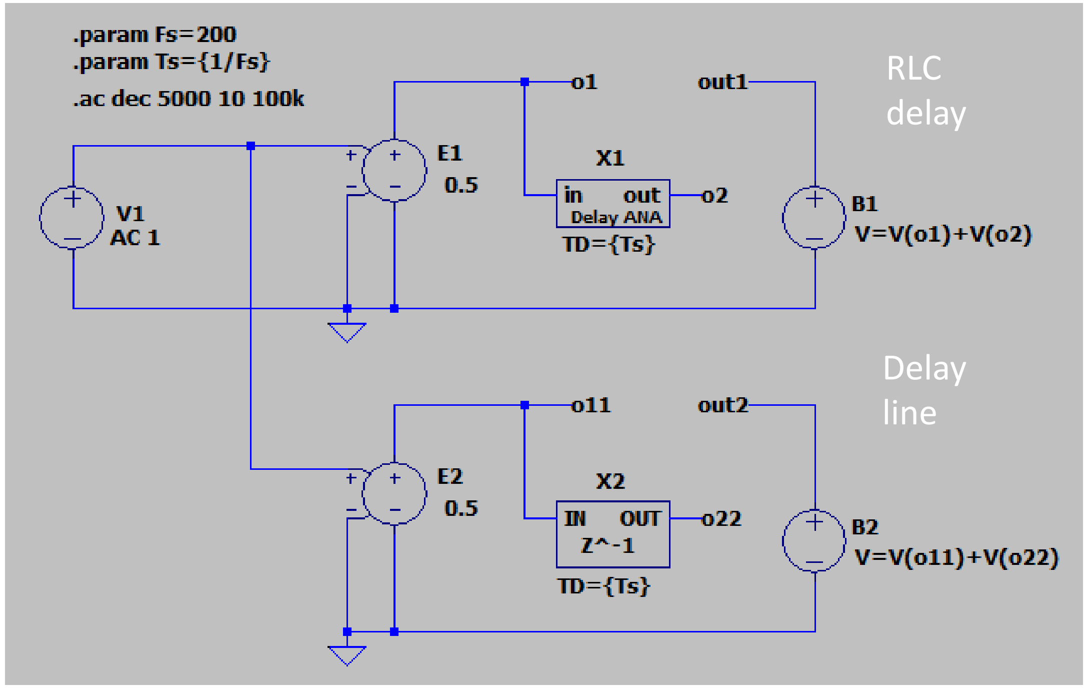


The agreement with the pure delay is acceptable in my opinion and I tested it in my book on small-signal modeling. The simulation truly benefits from this approach and the simulation time is reduced with the pure analog version.

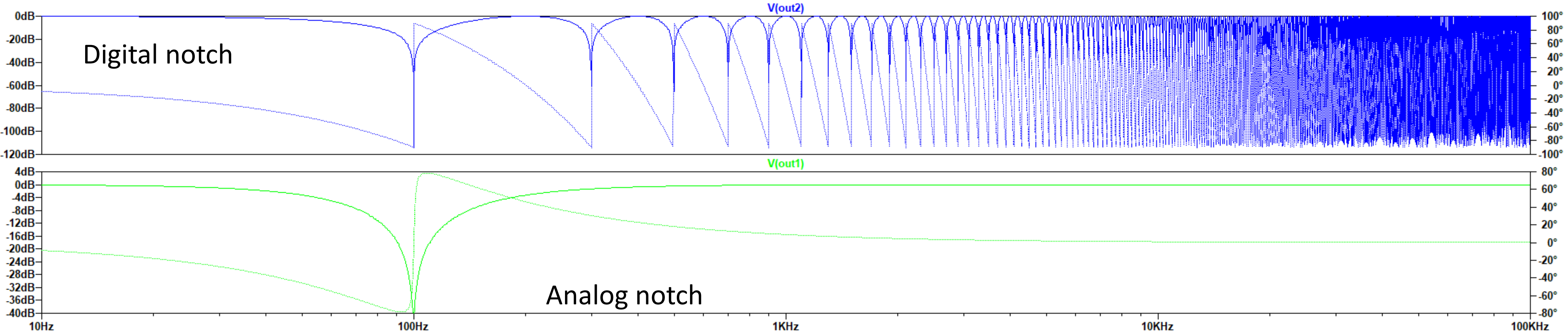


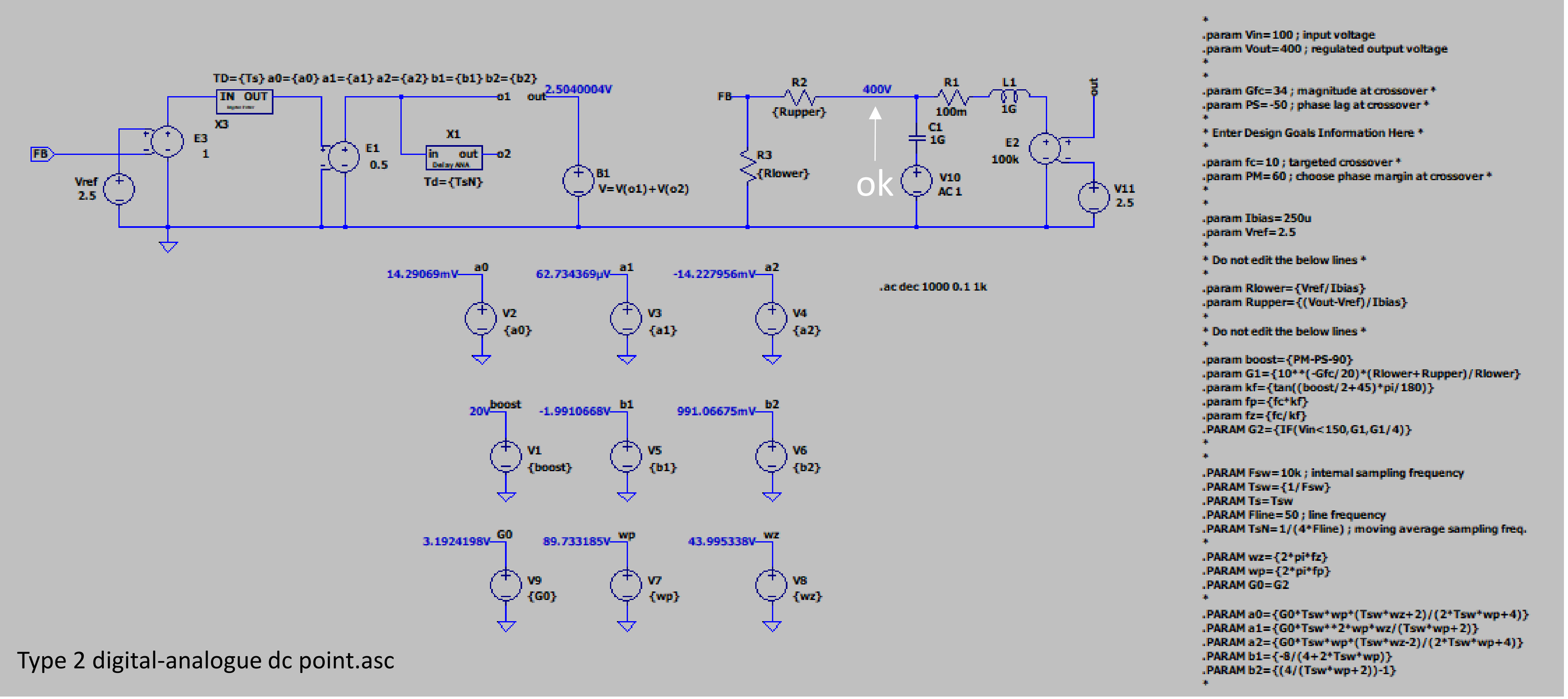
It took me a while to figure this out; the first attempts did not work for the analog delay in LTspice: the same circuit worked in IsSpice but not in LTspice! The guilty was resistor R_1 calculated to exactly 987 µΩ if L is arbitrarily set to 1 µH. LTspice minimum resistance is set to 1 mΩ so it messed up the results by not allowing a smaller value. By increasing L to 100 µH, R_1 is now equal to 98.7 mΩ and it rejects the 100-Hz component ok.

We can now compare the analog notch response with that including the delay line:

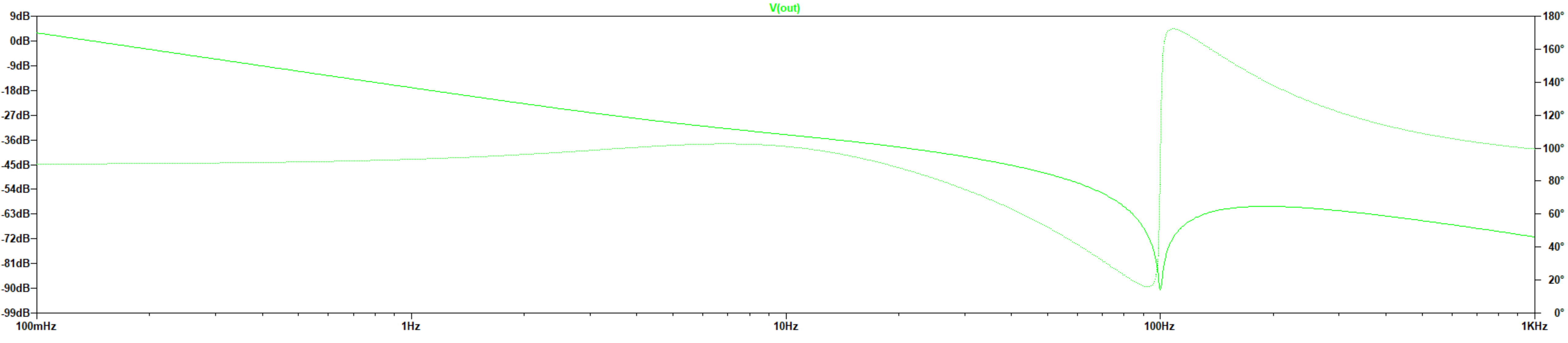


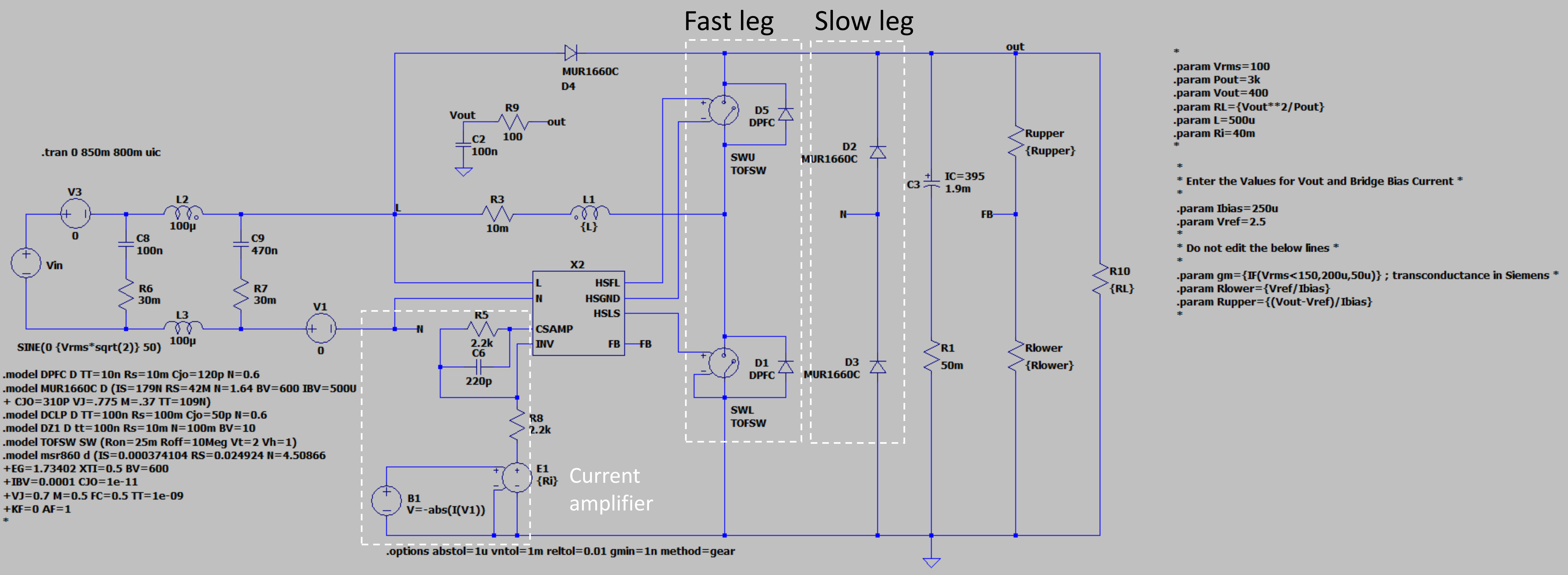
The selectivity is less with the analog notch but it provides a 40-dB attenuation at 100 Hz.



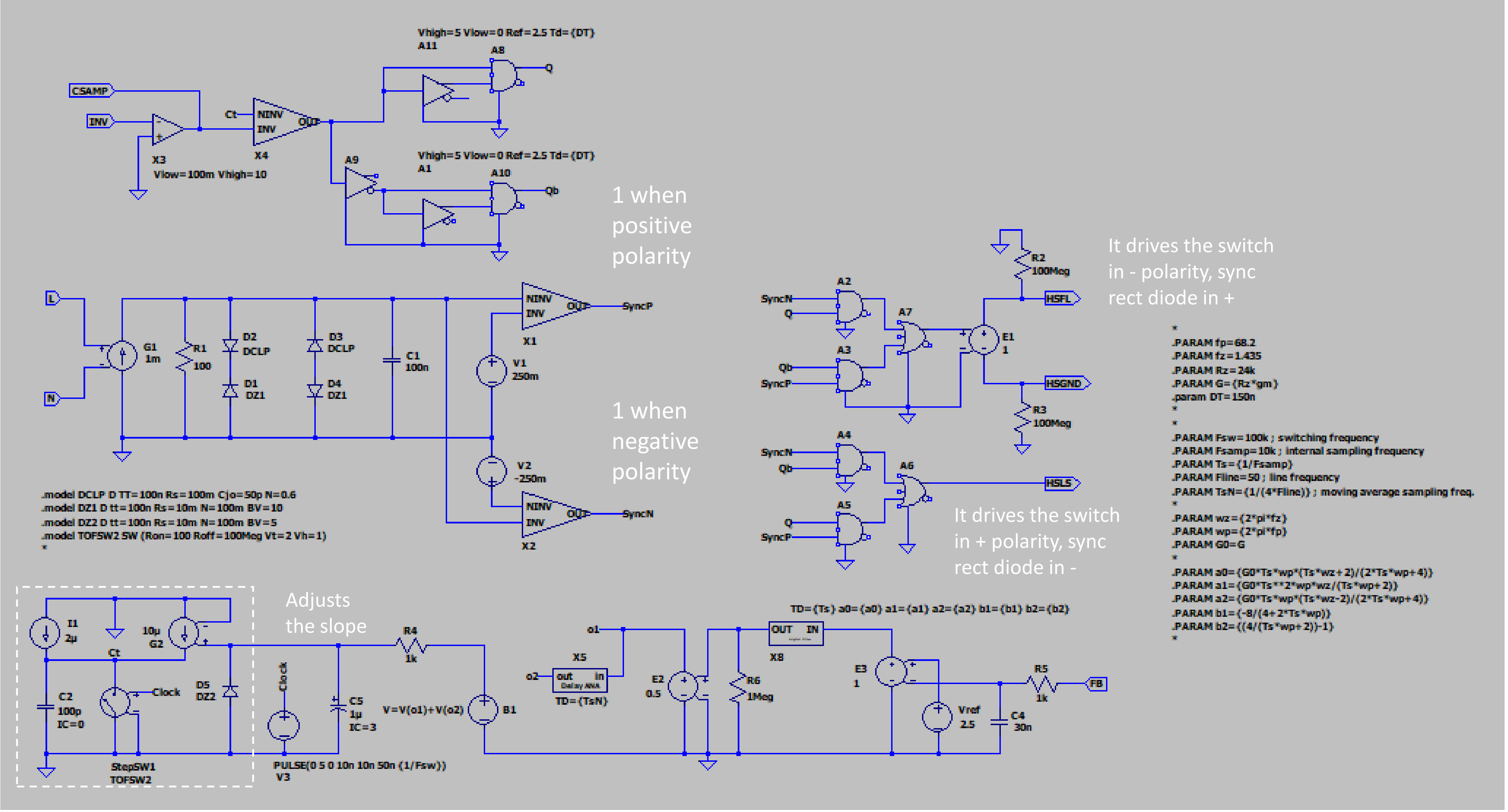


The digital type 2 compensator is now coupled with the 100-Hz notch and the component is rejected by 90 dB. By changing the line frequency to 60 Hz, this notch will be tuned at 120 Hz.

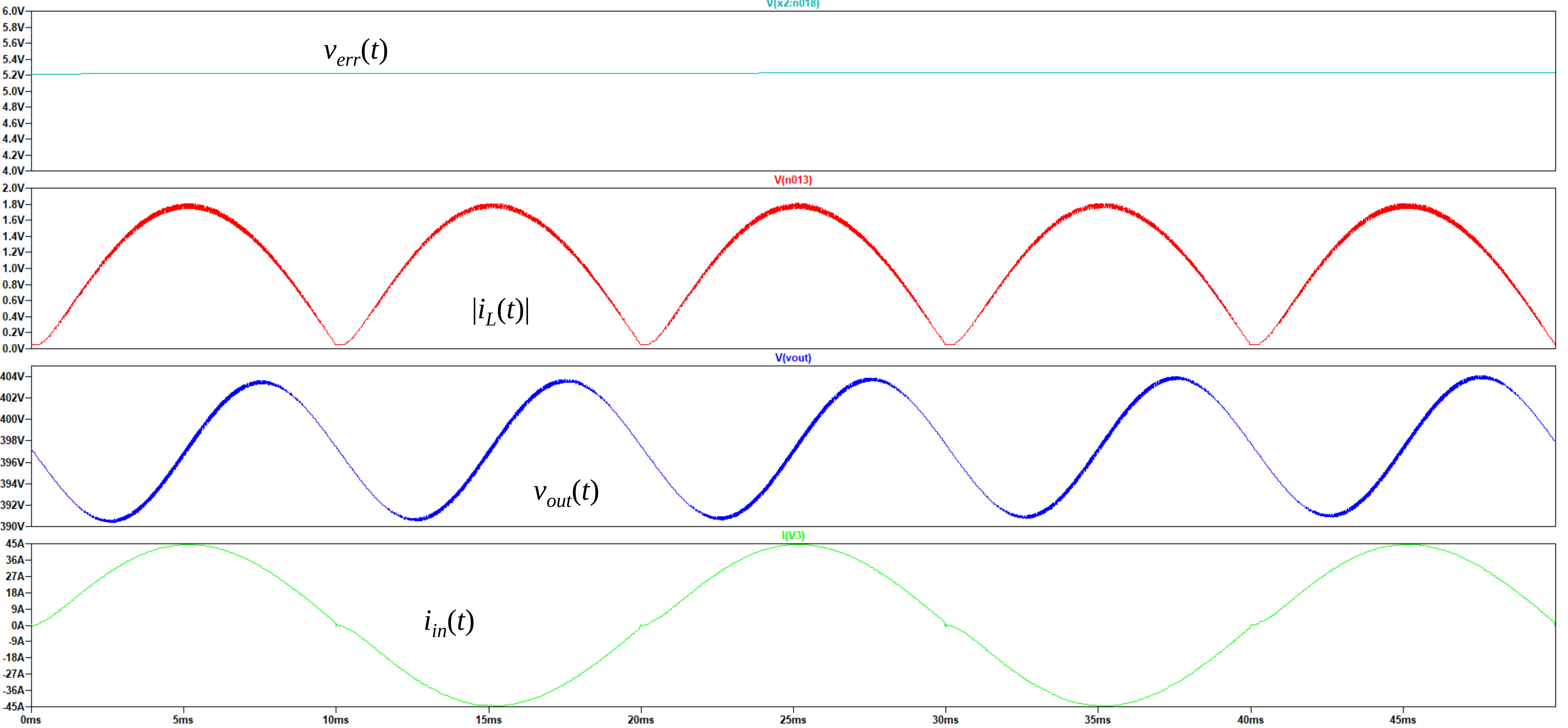




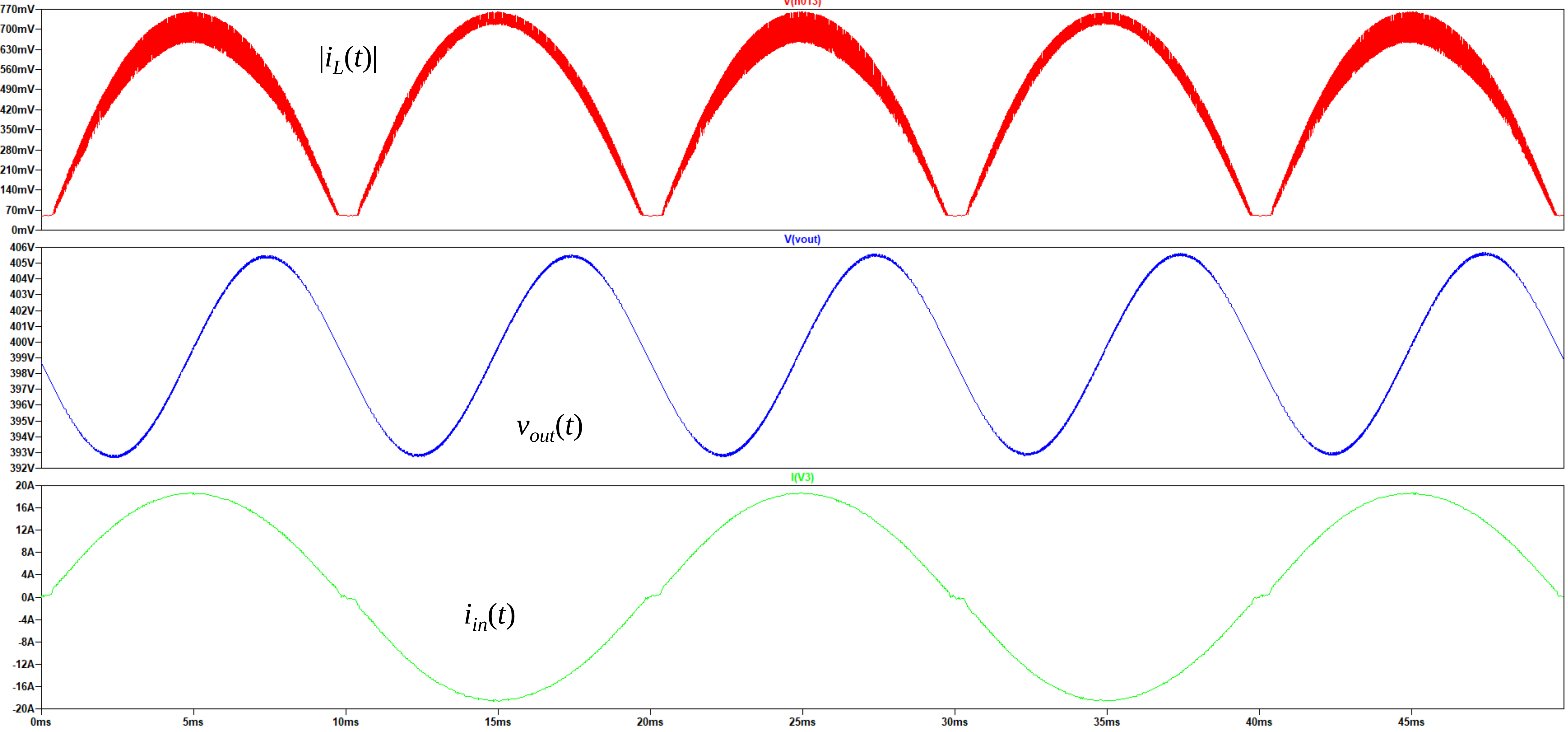
The filter is now included in a totem-pole PFC using a variable-t_{off} modulator as described by Sam Ben-Yaakov in his 1998 paper. The fast leg uses simplified switches while the slow one implements a pair of diodes. They can also be sync-rectified but the model does not include these outputs. Please note the g_m change in relationship with the input voltage to adjust the gain on the fly and keep a constant crossover at low- and high-line operations. The NCP1681 typically operates this way.



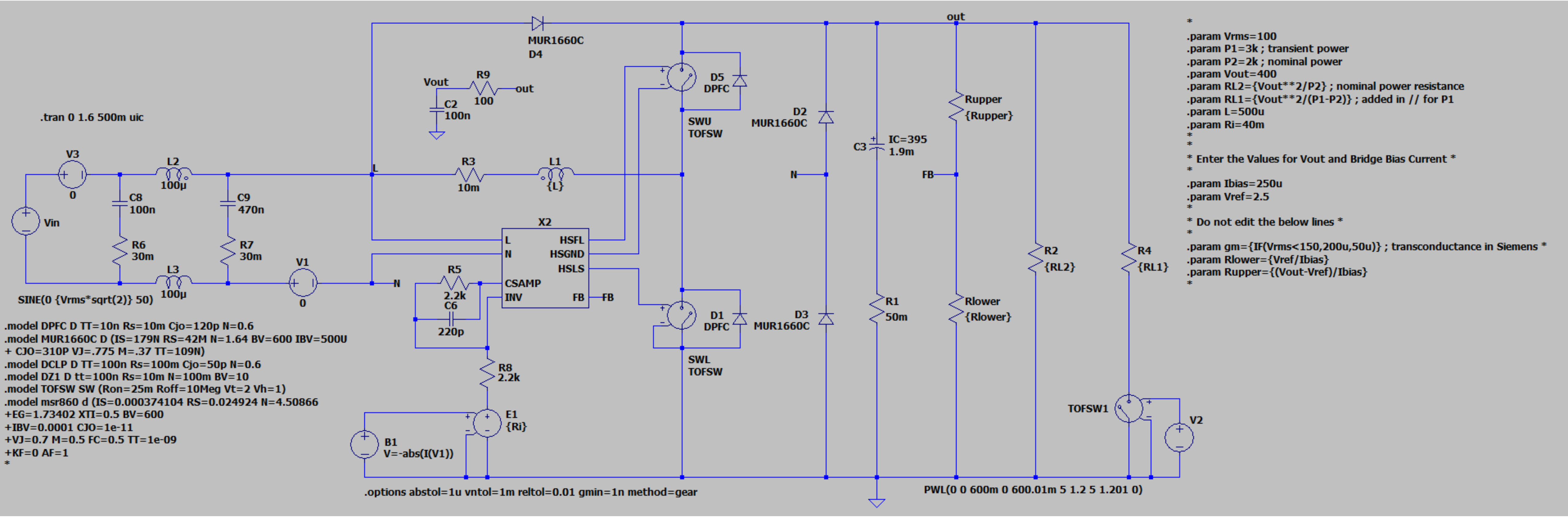
The TPPFC requires a logic circuit decoding the input line polarity for actuating the power switches. In this approach, only the fast leg is switching, the slow leg uses classical diodes. Please note the absence of virtual ground at FB input, hence the necessity to factor the divider ratio in.



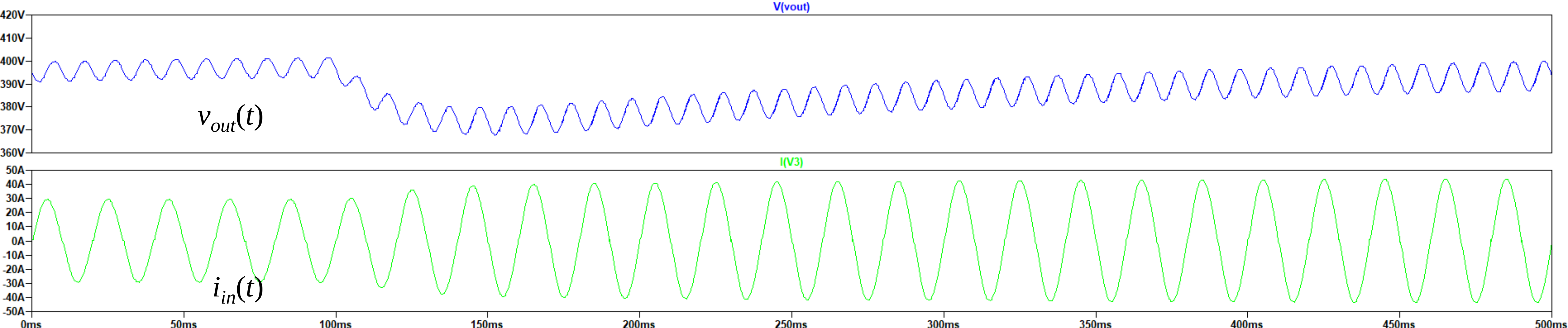
This is the input waveform in a 3-kW PFC supplied from a 100-V rms input voltage.
The error voltage is almost flat, without ripple.

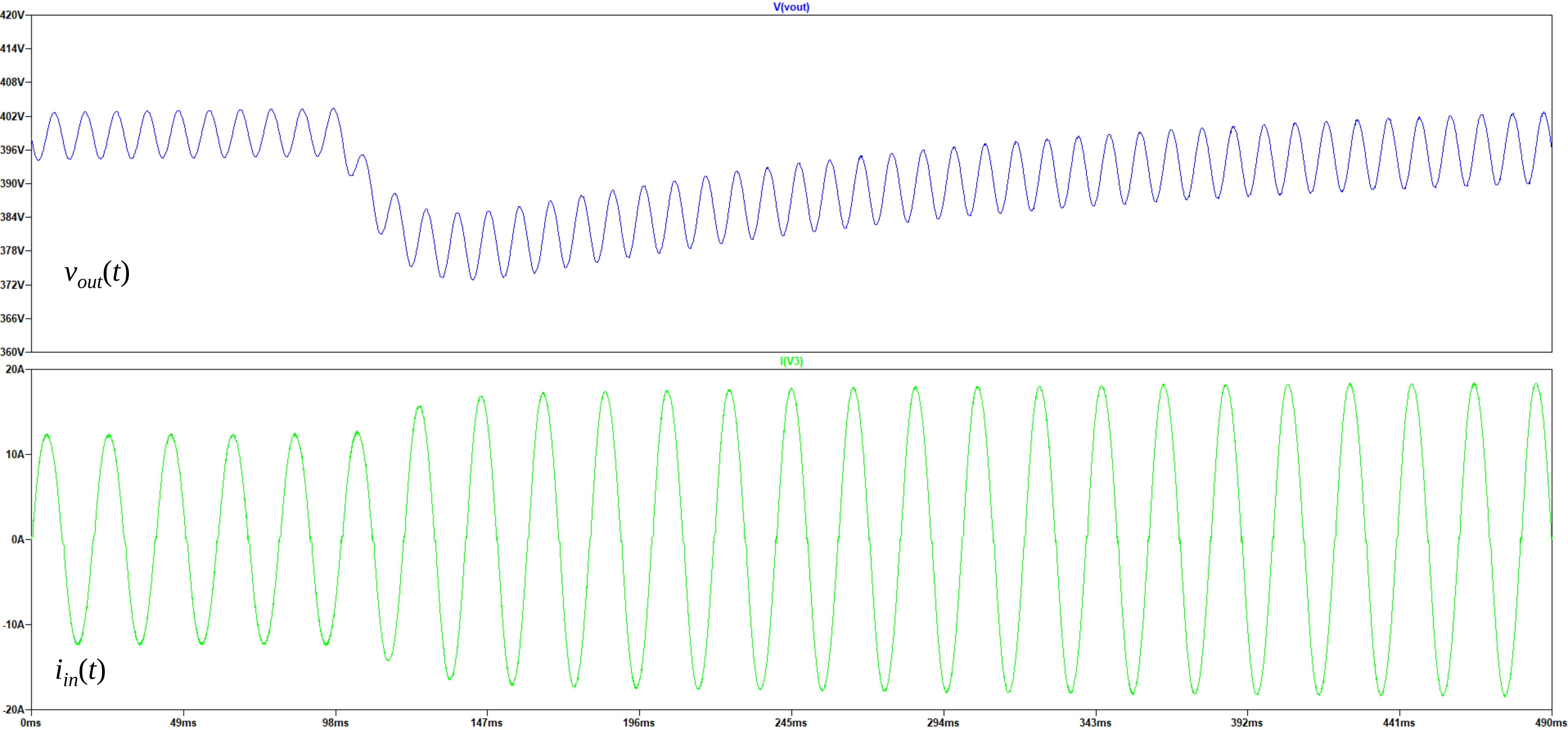


This is the input waveform in a 3-kW PFC supplied from a 230-V rms input voltage.



In this mode, the load is stepped from 2 kW to 3 kW in a short period of time. V_{in} is 100 V rms in this mode, F_{sw} = 65 kHz.





In this mode, the load is stepped from 2 kW to 3 kW in a short period of time. V_{in} is 230 V rms in this mode. The load release is not shown as no over-voltage protection is installed on this circuit and the overshoot is extremely large.