Qspice for "PWL" Power Electronic Simulation

Rev. 2 - 2026/02/04

Acknowledgement to KSKelvin (Kelvin Leung)
for the discussion and suggestions

Arief Noor Rahman

Principle Engineer

https://www.linkedin.com/in/arief-noor-rahman-3880a536
https://forum.qorvo.com/u/physicboy/summary
https://github.com/physicboy/QSPICE

Outline

- Introduction of (Q)spice
- Analog Electronics, Power Electronics, and The simulation
- The basic of Spice Solver
- Challenge of Spice for Power Electronic Simulations
- Adopting Qspice for (near) PWL Power Electronic Simulations
- Component model library for improved convergence - 1
- Example on making simple circuit with ANR Q-lib
- Further more advance technique

Introduction of (Q)spice

- **SPICE (Simulation Program with Integrated Circuit Emphasis)** is a general-purpose, open-source analog electronic circuit simulator. It is a program used in integrated circuit and board-level design to check the integrity of circuit designs and to predict circuit behavior - Wikipedia.
- The SPICE was first released back in 1973, with the latest release was Spice 3F5 back in 1993. These days, the original spice has been no longer maintained, instead it has been forked into various software packages. The direct descendent of Spice is now ngSpice being the only one being continuously maintained open source with sizeable community. There are a lot more other commercial-propietary descendant of Spice (Pspice, Simetrix, Topspice, etc) and free-propietary (LTspice and Qspice).
- This article will specifically focus on Qspice by Mike Engelhardt, freely available via https://www.qorvo.com/design-hub/design-tools/interactive/qspice. It features a further improved solver, renewed user interface, and most significantly (for me at least) the introduction of advance digital control block (supports both C-code and Verilog).

Analog electronics, Power electronics, and The simulation

- Analog electronic is generally defined as a circuit contained linear element (RLC) and nonlinear semiconductors (transistor BJT/MOSFET/JFET/etc and diode) primarily operates within its linear-ish region.
- Power electronic circuit operates with different principle compared to analog electronic. In analog circuit, the emphasize is on using semiconductor devices within its linear range. While, in power electronic the semiconductor device are operating in either fully OFF and fully ON (saturated) and also minimize the switching transition duration between OFF and ON down to tens of nano-second or less. By minimizing the switching transition the device operates with low resistance thus minimize the losses and maximize efficiency.

And about the simulation:

- Spice was originally developed for analog electronic simulation. It uses non-linear newton rhapson based solver to obtain the detail of the circuit operating point at each time step and use an integrator to compute the state parameter at each step. Two different convergence criteria are used to evaluate if the newton-rhapson solver and if the transient error by the integator both have error within the required tolerance by which the simulation can accept the solution at each particular timestep then move-on to the next timestep.

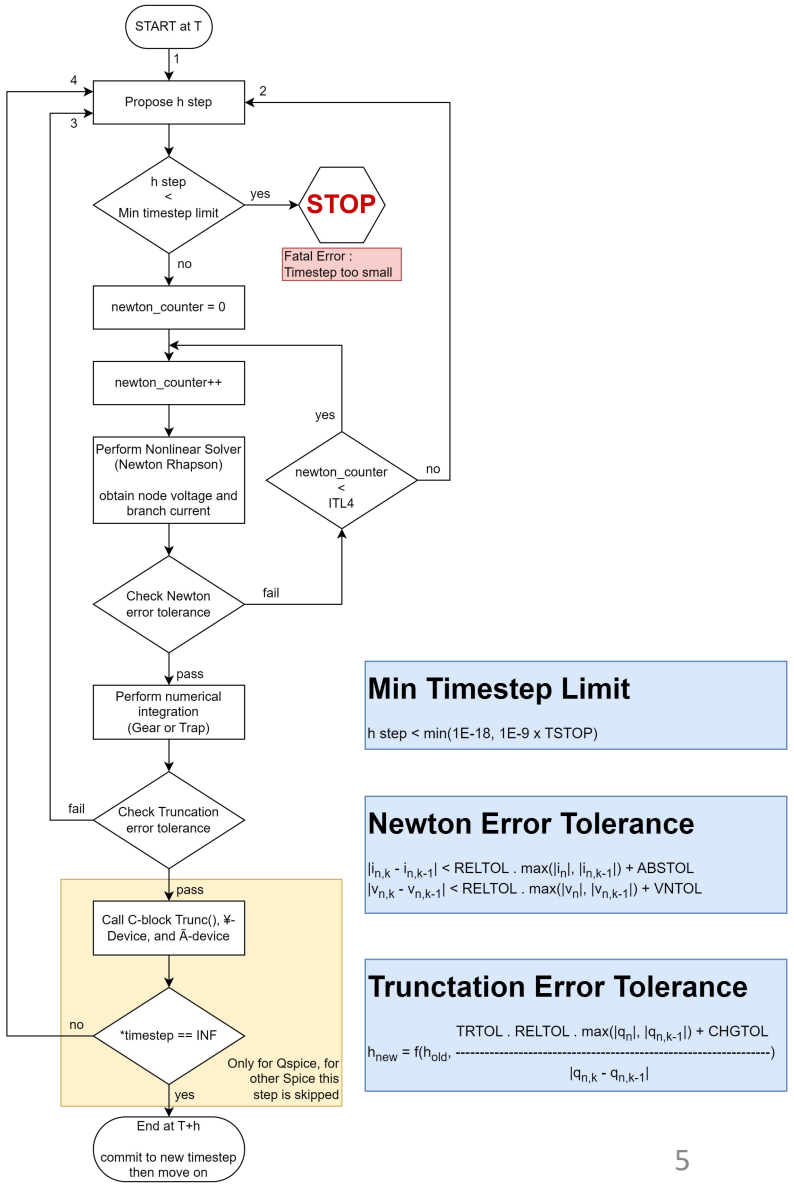
The Basic of Spice Solver

Disclaimer: based on my very limited current knowledge, take it with a spoon of salt

- Transient solver used in Spice consisted of 2 solvers which work together following the program flow as described in the flow chart on the right side.
  - First at each simulation instantaneous runtime, the solver will propose a time step “h”.
  - Then nonlinear circuit solver is called to compute an estimated circuit node voltage. The estimation is iterated using Newton-Rhapson algorithm and it will continue until the error tolerance fall below the target. If the error on all nonlinear branches voltage and current does not converge within the prescribed iteration count limit (ITL4), the solver will propose a new “h” agressively much smaller.

    → convergence parameters: RELTOL, ABSTOL, VNTOL
  - Once the nonlinear solver converged, transient solver will perform numerical integration then evaluate the stored charge on all capacitors, capacitive branches (such as MOSFET gate) and some other circuit states (so called Local Truncation Error, LTE) then evaluate if the proposed $h_{new} > 0.9 \times h_{old}$ (or different convergence criteria) to either accept the existing “h” or shrink the “h” then re-compute the solution by the nonlinear solver again.

    → convergence parameters: RELTOL, CHGTOL
  - Lastly after the transient solver converged, then C-block, ¥-Device, or Ã-device check if these devices can accept the “h” value.



Challenge of Spice for Power Electronics Simulation

- Spice is originally designed for analog electronic with particular emphasize on integrated circuit analysis.

  Meaning:

  1. Circuit elements for simulation are mostly aimed for semiconductors within OFF, linear, and saturated range. Thus detail of device transients are simulated
  2. The small circuit elements also means small voltage and current magnitude being simulate

  Thus:

  1. Circuit elements for simulation are mostly aimed for semiconductors within OFF, linear, and saturated range. Thus detail of device transients are simulated
  2. The small circuit elements also means small voltage and current magnitude being simulate

- Meanwhile, for power electronic the circuit operates with switching operation where the switching transient duration minimized as much as possible (typically within ns ~ less than 100 ns for most MOSFETs, or a bit larger for IGBT). Due to the relatively very short (or negligible) switch transition duration, for most of power electronic simulation it is common to treat MOSFET/IGBT as an ideal switch to speed up the simulation. Due to this contrast difference on the requirement, different kind of software package for switching power electronic are emerged (PLECS, PSIM, NL5, SIMBA, SIMPLIS) where the switching components (MOSFET/IGBT/Diode) are completely replaced with ideal switch with instantaneous switching transition (also called Piecewise Linear, PWL equivalent). By using PWL based approach, the newton rhapson iteration can be eliminated, thus speeding up the analysis and also improve the robustness for this target application.

  Concern of Spice:

  1. Default convergence criteria was designed for IC is overly strict for switching power circuit.
  2. Newton rhapson maybe very slow to solve rapid ON→OFF or OFF→ON transition for repetitive switching events. Such steep transition may cause Jacobian derivation to fail and the whole Newton iteration to fail.

     → Qspice has improved solver robustness.

Adopting Qspice for PWL Power Electronics - 1

- Right of the bat with the default solver options, Qspice should work just fine for most of the simple-ish circuit. Anyway, with a rather complex circuit we can adjust simulation options as following (and the suggested range for tuning):

	Default	Modified (range)	Explanation
Solver selection			
fastmath	true	false	True: 64-bit math, False: 80-bit math, use 80-bit math for improved numerical precision
method	trap	gear	Trapezoidal (trap) is more accurate than gear, it ensure accurate integration value without any artifical damping making it essential for resonator/oscillator simulation. However, trap suffers from artificial ringing. In switching circuit, gear is more suitable to reduce excessive ringing and thus generally more robust for the application.
uic	not set	set	ensure to add uic by the end of .tran command to skip bias point analysis and use only state initial condition provided by the user. By default, spice solver will compute the bias (or operating point) point of the circuit to speed up the transient analysis. However, the bias point analysis is unreliable for switching converter thus the estimate result may be return wildly extreme value. Turning off this option ensure all state start from zero (or user provided initial condition).
ITL4	10	100	ITL4 corresponds to the maximum iteration count of newton rhapson, if the newton solver didnt converge by the end of ITL4 the proposed timestep will be reduced (rather aggressively) to help the convergence. By increasing ITL4, we are forcing the solver to keep trying, and hope that it can solve the newton without tightening the timestep.

Adopting Qspice for PWL Power Electronics - 2

	Default	Modified (range)	Explanation
Convergence criteria			
reltol	1m	0.1m (0.1m ~ 1m)	For general simulation reltol=1E-3 is sufficient. However when detail accuracy of circuit waveform is required (such as to analyze circuit THD) tighten reltol to 1E-4 improve the result.
abstol	1`p	1µ (0.1µ ~ 1µ)	In IC electronic, the current flow through the transistors are generally in µA range, thus current accuracy down to 6 order is required to obtain good result. For power electronic, 1E-12 is unnecessarily too tight, so we can set abstol to be 6 order of smaller than the nominal current in simulation, 1E-7 ~ 1E-6 seems to be reasonable for 100mA ~100A range.
vntol	1µ	10µ (10µ ~ 1m)	Loosen the accuracy range for same reason with abstol. Anyway for vntol, a bit lower accuracy (compared to abstol) of ~5 order is generally enough due to the physic of the circuit solving method which is done by guessing the nonlinear voltage state to obtain accurate current.
chgtol	10f	1p (1p ~ 1n)	Since no transistor (and thus gate charge) exist, thus no tiny capacitor requiring solving exist. We can loosen the chgtol to achieve the target voltage accuracy requirement on capacitors, we can adjust this chgtol by checking the minimum voltage accuracy on the smallest capacitor used in the circuit. If current source + timing capacitor is used (like in L6599 LLC controller), we can simply use this timing capacitor and target voltage accuracy to set this parameter. (CHGTOL = C_{min} x V_{tol}). Generally, using CHGTOL = ~1E-9 is a good value to keep track of small capacitors charge though loosen it to 1E-12 doesnt hurt the speed very signficantly either.
TRTOL	2.5	7 (2.5 ~ 7)	Basically TRTOL is a constant used to loosen the RELTOL used in tracking charge transient error. Smaller value resulting in stricter truncation error criteria thus more accurate but also slower simulation. We can use TRTOL = 7 following the default value used by TINA, PSPICE which I believe its reasonable to loosen following the value used in these softwares.

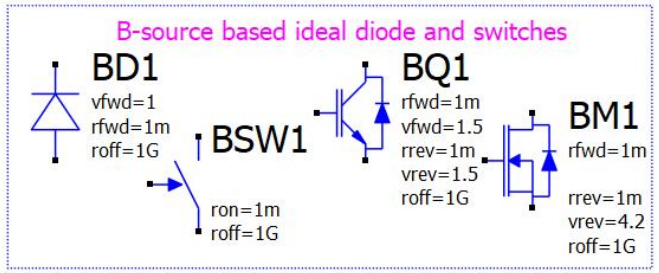
As a rule of thumb:

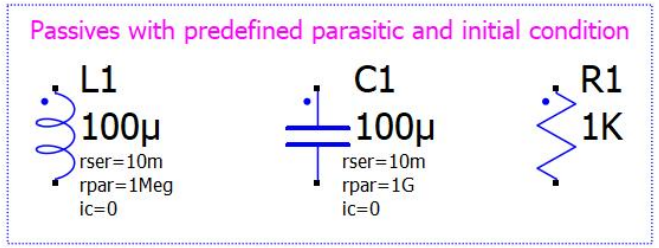
- Timestep too small failure mostly caused by Newton iteration failure -> fixed by loosen the abstol and vntol.
- To trade off speed and waveform accuracy -> adjust RELTOL and TRTOL mostly

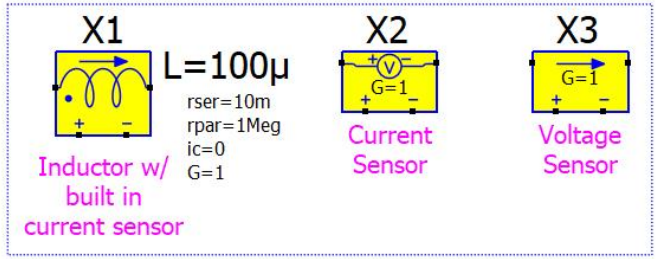
Adopting Qspice for PWL Power Electronics - 3

	Default	Modified (range)	Explanation
Convergence aid			
cshunt	0	1f	Adding small circuit impedance and leakage to the ground (cshunt = capacitor to gnd and gshunt = admittance to gnd) and also adding large parallel admittance (gmin) across inductor and semiconductor device help the jacobian matrix solver in newton iteration to converge. 1E-15 for both cshunt and gshunt and 1E-9 for GMIN is practically negligible for to affect any noticable change to circuit behavior. In some very particular application where it matters, then user may just consider to revert back to default value. For cshunt ensure to keep the charge stored in the cshunt at the nominal operating point to be smaller than the chgtol to avoid timestep tightening due to LTE non-convergence.
gshunt	0	1f (1f ~ 1p)	
gmin	1p	1n	
Limit the stored data			
culltime	2p	100n (any value)	limit the sampling of the waveform data storage, thus the time of every subsequent stored waveform data is ≥ CULLTIME
.save	-	-	By default, Qspice will store all waveform data including for every subckt. Add a .save command to tell the spice to only store the waveform as specified on this command.

Component model library for improved convergence - 1





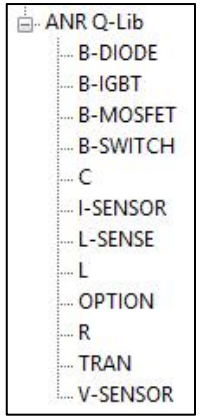


Transient setup

By Arief Noor Rahman

Berkeley Syntax : .tran

Tstop	: 100m
Tstart	: 0
MaxTimeStep	: 10u
Uninitialized IC	: uic



ANR Q-lib power component library and drag-drop option and tran syntax has been developed following the suggestion provided.

The options provided in this library should work for vast majority of circuit with good convergence and reasonable computation speed. If necessary, adjustment following the guideline provided can be performed.

Transient option for ideal PWL power electronic

By Arief Noor Rahman

Berkeley Syntax : .opt

Solver option
- fastmath=false
- method=gear
- ITL4=100

Convergence option

reltol=0.1m	(0.1m~1m)
abstol=1u	(0.1u~1u)
vntol=10u	(10u~1m)
chgtol=1p	(1p~1n)
trtol=7	(2.5~7)

Convergence aid

cshunt=1f	(1f)
gshunt=1f	(1f~1p)
gmin=1n	(1n)

Waveform save option
- culltime=100n

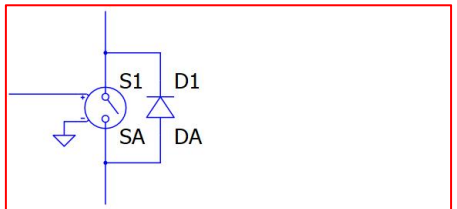
cshunt, gshunt, gmin, and cull time can be removed if not needed

Component model for better convergence - 2

L and C, avoid these two from forming high-Q resonant circuit by adding damping with small series resistance and large parallel resistor.
Note:

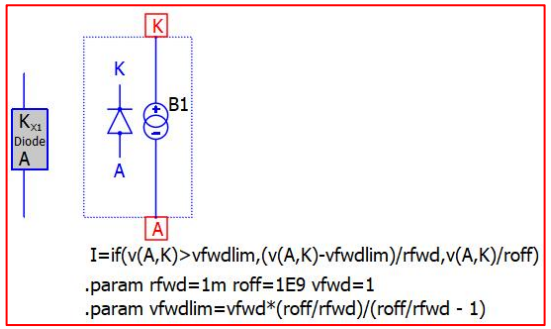
1. Dont use extremely small rser as it may cause convergence problem, (rser > 1m or use a practical cap ESR is suggested)
2. For inductor rpar=1Meg is reasonable to emulate a very small core loss and aid convergence
3. For capacitor rpar=10Meg ~ 1G is beneficial to aid convergence

Behavioral Model for Switches

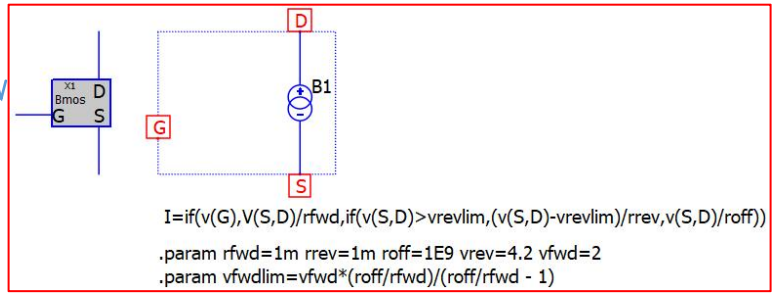



```
.model SA SW ron=1m roff=1E9 vt=0.5 vh=0
.model DA D vfwd=1m ron=1m roff=1E9
```

Originally, I used to use this model as MOSFET with body diode.



I guess, the newer model is somewhat easier to solve as it combine the forward and reverse path of the MOSFET and IGBT into single B-source, and further due to less part count.

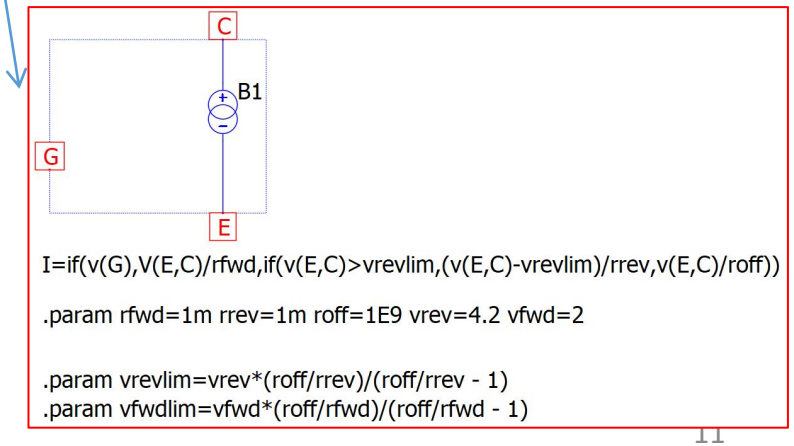


Anyway, I recently developed this new model using behavioral source to create MOSFET/IGBT/Diode.

Note: the gate signal is ground reference with value of 0 (off) or 1 (on)

In some cases, the older model runs a bit faster, but overall the newer model experience far less “Timestep too small” issues

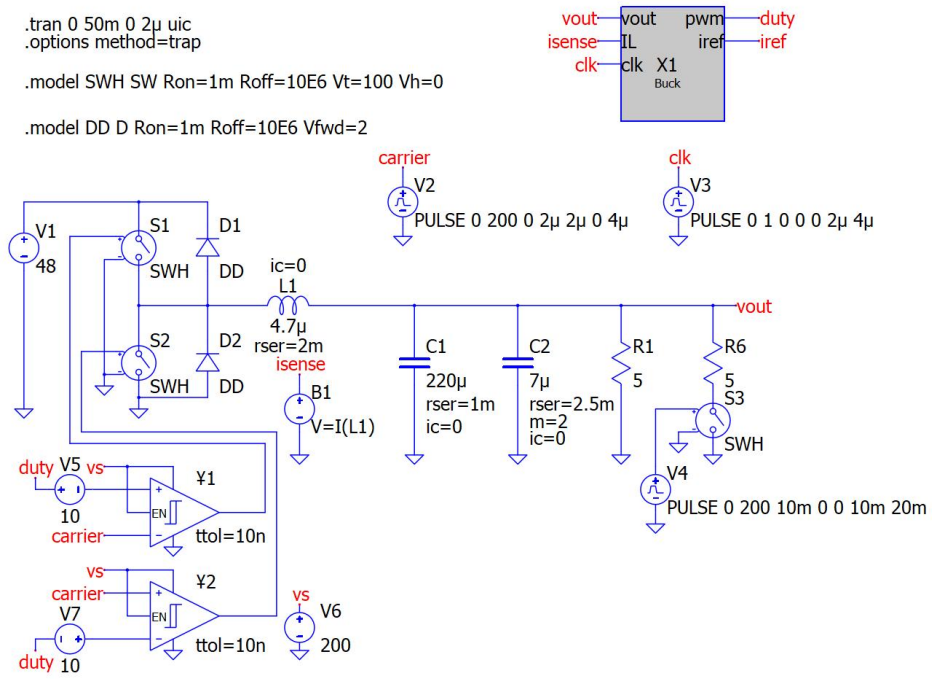
In the ANR Q-lib library, the model developed has been converted into symbol



Example on making simple circuit with ANR Q-lib

Further more advance technique

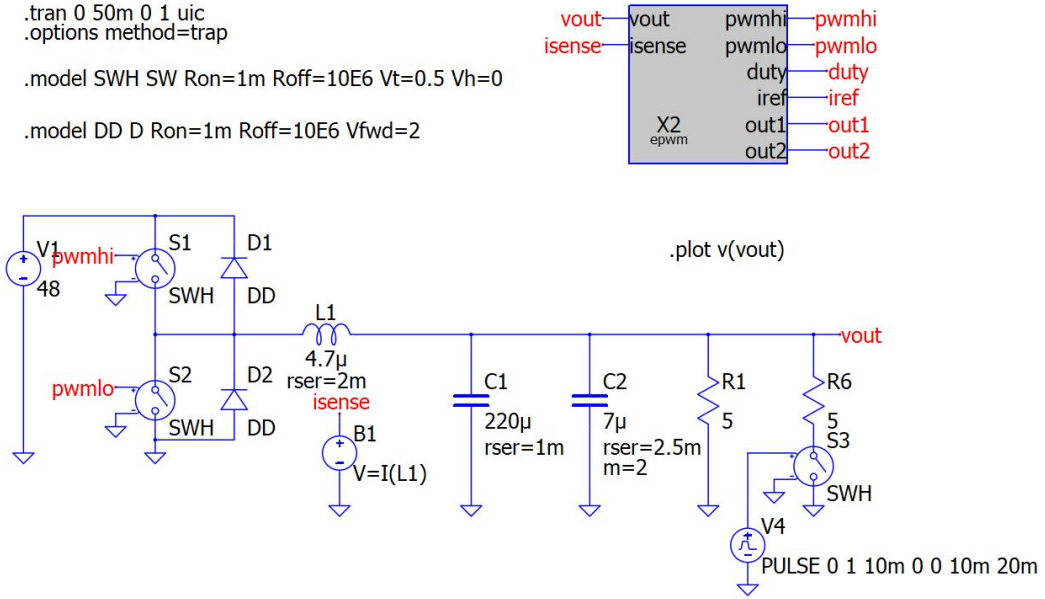
Conventional PWM with ¥-Device comparator



https://github.com/physicboy/QSPICE/tree/main/PWM_example_SRbuck/Level1D%20SR-Buck%20External%20Compare%20PWM

9.2 sec runtime

PWM by C-block utilizing advance time control

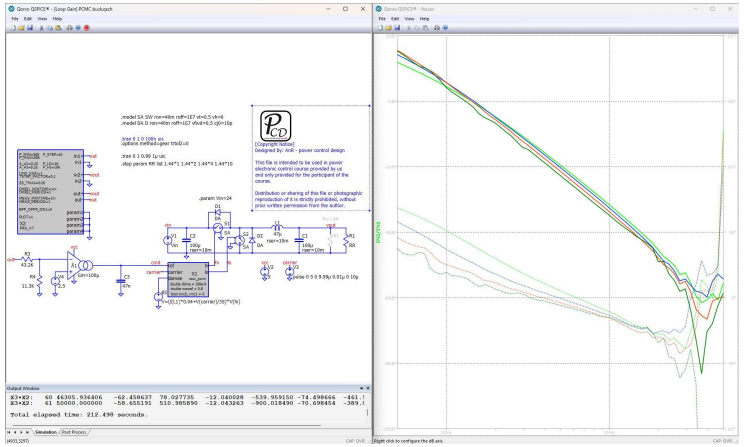


https://github.com/physicboy/QSPICE/tree/main/PWM_example_SRbuck/Level4%20SR-Buck%20Internal%20Cblock%20PWM%20with%20Deadtime

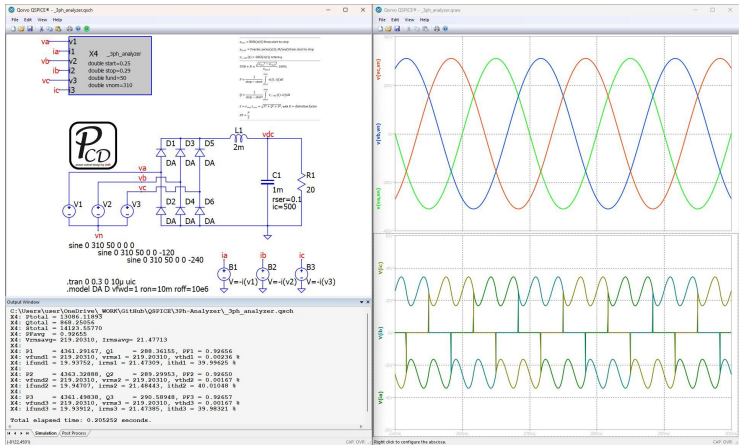
3.5 sec runtime

Note:
The simulation was simple enough, thus the method described previously for improved convergence are not necessary

Other Important Analysis Tool for Power Electronic



Bode Plot Analyzer
https://github.com/physicboy/QSPICE/tree/main/FRA_project



3Phase power and THD analyzer
https://github.com/physicboy/QSPICE/tree/main/3Ph-Analyzer

Practical Modification of Switching Losses Estimation from Datasheet Parameter for SiC MOSFET

By Arief Noor Rahman – Power Control Design

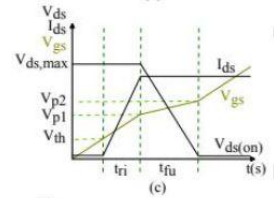
Note: the method presented here may have some discrepancy when compared to Eon and Eoff curve provided by the datasheet. The measurement of Eon is also greatly affected by the freewheeling diode characteristic (typically body diode of same MOSFET) used in the datasheet measurement.

Section 1 -> Background:

Some of the most common method to estimate the switching losses in SiC MOSFET is based on the time required to charge the MOSFET parasitic capacitance (Ciss, Coss, Crss) as most famously first described in [1] which then later refined for SiC MOSFET that have non planar miller plateau in [2].

[1] D. Graovac, M. Purschel, and A. Kiep, , "MOSFET power losses calculation using datasheet parameters," 2006 Infineon application note V 1.1

[2] B. Agrawal, M. Preindl, B. Bilgin and A. Emadi, "Estimating switching losses for SiC MOSFETs with non-flat miller plateau region," 2017 IEEE Applied Power Electronics Conference and Exposition (APEC), Tampa, FL, USA, 2017, pp. 2664-2670, doi: 10.1109/APEC.2017.7931075.



When defining losses estimation as a function of parasitic capacitance the turn on and turn off losses (E_{on} and E_{off}) is then calculated as:

$$E_{on} = V_{ds} \cdot I_d \cdot \frac{t_{ri} + t_{fu}}{2} \quad (1)$$

$$E_{off} = V_{ds} \cdot I_d \cdot \frac{t_{ru} + t_{fi}}{2} \quad (2)$$

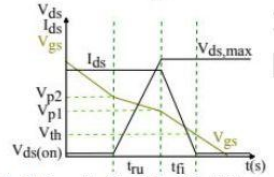


Fig. 1. Theoretical depiction of switching dynamic according to [2]

Power Control Design - 2025 1

https://github.com/physicboy/QSPICE/tree/main/Ideal%20PWL%20Transient%20Losses%20Analysis

For everything else, check my Github!

References and Further Read

1. Ken Kuncert and Ian Clifford, “Achieving Accurate Results with a Circuit SImulator”, Cadence Design Systems.
2. K.G. Nichols, T.J. Kazmierski, M. Zwolinski, A.D. Brown, “Overview of SPICE-like Circuit Simulation ALgorithms”, IEEE Proc. Circuits Devices System, Vol 141, No. 4, 1994.
3. Laurence W. Nagel, “SPICE 2: A Computer Program to SImulate Semiconductor Circuits”, UC. Berkeley, 1975.
4. Kenneth S. Kundert, “The Designer’s Guide to SPICE and SPECTRE”, Cadence Design Systems, Kluwer Academic Publishing 1995.
5. Andrei Vladimirescu, “The Spice Book”, Wiley, 1994
6. K.S. Kelvin, “Qspice - Command Reference Guide”, https://github.com/KSKelvin-Github/Qspice/blob/main/Guideline/Qspice%20-%20Device%20Reference%20Guide%20by%20KSKelvin.pdf
7. Arief Noor Rahman, “https://github.com/physicboy/QSPICE/tree/main”