Application Note

# Simulating TDK's µPOL Modules

by Herman Neufeld

*Abstract: This app note explores the use of QSPICE® Simulator by Qorvo and SIMPLIS® by SIMPLIS Technologies for simulating TDK's Micro Point-of-Load (µPoL) modules. It is primarily intended for FPGA Hardware Designers as well as Power Management Engineers.*

## Introduction

QSPICE is a recently announced simulation (sim) tool, which is continuously being updated as new models are created and uploaded. It is a free software recommended for all general needs. A noteworthy advantage of QSPICE is its fast execution speed. SIMPLIS, on the other hand, is a well-established tool with a very high degree of functionality and a sophisticated analysis capability. Technical support at SIMPLIS is also excellent. To use SIMPLIS, however, you need to purchase a license and a USB dongle. The purpose of this app note is not to educate the user on these tools but rather show how to use them effectively with µPoL modules. TDK has already made a number of simulations which have been organized into tables and cataloged by FPGA on TDK's website. A section of a typical table is shown in Fig. 1.

In this table Rail 1 is the core rail which can be set from 0.7 to 0.9V. Therefore, three sets of simulations are provided, namely for 0.7, 0.8, and 0.9V. The necessary files to run these simulations have been zipped and embedded in the second column from the right. Note: not included in the zip file is the license file, which SIMPLIS requires to run the schematic. Also note that TDK models are proprietary and have been encrypted. After extracting the files from the zip file, click on the file with the SIMPLIS logo which, for example, would look like as follows:


Agilex 7 SIM Library SIMPLIS

| Rail Nr | Voltage rail name | Nom Vin | Max Iin | Max DC ripple | DC reg | AC reg | Load step | Slew rate | TDK µPOL | SIMPLIS model | Transient response plot |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | VCCL 0.8V: VCC, VCCP, VCCL_HPS, VCCPLLDIG_HPS* | 0.70V | 25A | 35mV | | ±3% | 30% | 140A/µs | FS1525-0600 | 0V7 25A 30pc 140Aµs SIM.zip | Plot 0V7 25A 30pc 140Aµs SIM.pdf |
| | | 0.80V | 25A | 40mV | | ±3% | 30% | 140A/µs | FS1525-0600 | 0V8 25A 30pc 140Aµs SIM.zip | Plot 0V8 25A 30pc 140Aµs SIM.pdf |
| | | 0.90V | 25A | 45mV | | ±3% | 30% | 140A/µs | FS1525-0600 | 0V9 25A 30pc 140Aµs SIM.zip | Plot 0V9 25A 30pc 140Aµs SIM.pdf |
| 2 | P0V8_GR1 0.8V: VCC_HSSI_GXF, VCCL_SDM/VCCH, VCCH_SDM, VCCPLLDIG_SDM* VCCLPLL_NOC* VCCPLLDIG_NOC* | 0.8V | 25A | 40mV | | ±3% | 20% | 70A/µs | FS1525-0600 | 0V8 25A 20pc 70Aµs SIM.zip | Plot 0V8 25A 20pc 70Aµs SIM.pdf |
| 3 | VCC_HSSI_GXR | 0.9V | 25A | 45mV | | ±3% | 20% | 70A/µs | FS1525-0600 | 0V9 25A 20pc 70Aµs SIM.zip | Plot 0V9 25A 20pc |

**Fig. 1 Partial view of a SIMPLIS simulation table for Altera's Agilex 7 FPGA.**

The filename is composed of the four parameters needed to specify the sim, namely Vout [= 0.8V], Iout [=25A], Load Step [= 30 percent], and Slew Rate [=140A/µs]. SIM stands for SIMPLIS. QSPICE sims have filenames ending with QS. If the same conditions can be met by more than one µPoL, a suffix is added to the filename to indicate this. If the input voltage is 5V instead of 12V, 5VIN is added as a prefix in the filename.

Once the four required design parameters have been defined, one needs to configure the output filter section, which typically consists of a group of capacitors (caps) connected in parallel. The cap values vary, as well as their equivalent series resistance (ESR), a frequency-dependent parasitic component. When these caps are connected in parallel, their values add, so they are lumped up into one equivalent capacitor and one ESR. Note that this is a first-order approximation. Also note that cap values go down as a dc voltage is applied to them. Derating curves for these caps can be found in the manufacturers' datasheets. Because there is a wide range of cap and ESR values, four typical values have been chosen for the sims presented here: 3 caps and 2 ESRs. One should choose the sim that closely approximates the equivalent cap in the design.

The results of the sims are shown in the plot files embedded in the right column of the table of Fig. 1. Once the correct sim for the application is chosen, it can be fine tuned by modifying the schematic file. One such modification is adding a secondary LC filter stage, where the inductor may be a ferrite bead. Parasitic components can also be extracted from the circuit board based on the length, width, and thickness of the solder traces from the µPoL to the actual load to provide more accurate results.

The transient responses of the dcdc converters have been captured via screenshots and displayed for every sim in the plot files. These screenshots are also highly useful for estimating loop stability. Fig. 2, for example, illustrates the case of an excellent transient response. When a dynamic load is applied, Vout drops to an acceptable level and is allowed to slightly exceed the steady state output voltage before it finally swings back to it.

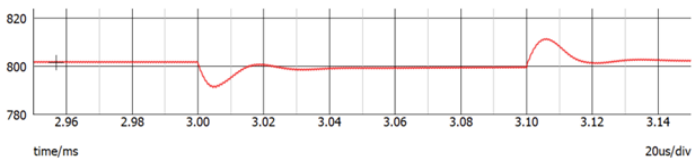



**Fig. 2 Transient response for a dcdc converter with excellent loop stability.**

If excessive capacitance is used, a load step could cause the output voltage to oscillate before finally settling down to its steady state value. This condition should be avoided. Conversely, lowering the output cap too much and increasing the ESR improves stability but at the expense of a much higher output ripple and a worse AC regulation. One should therefore weigh all the trade offs before deciding what filter to use.

Zooming in with the mouse on the output ripple, reveals another interesting observation. Fig 3a illustrates the inductor current flowing through the ESR producing a voltage in phase with this current, which is proportional to the ESR value. This output ripple is said to be **ESR-dominated**. This condition provides for a fast loop response and a good stability, but at the expense of a higher dc ripple voltage, which could be a problem in, for example, SERDES rails. In Fig. 3b, however, the ESR is so low that its effect on the output ripple is barely noticeable. The output cap also introduces a 90° phase lag on the output voltage. This output ripple is said to be **Cap-dominated.** Although the output ripple is low and smooth, this phase lag actually slows down the converter's transient response, consequently lowering its phase margin. The output will exhibit some ringing as already mentioned. Again, trade offs should be considered, hence the main purpose of the sim tools. TDK's µPOLs also provide additional flexibility for tailoring the loop response via external components.

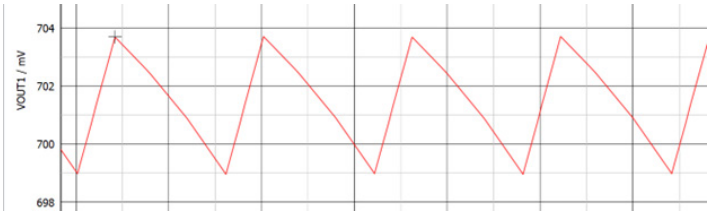



**Fig. 3a ESR-dominated output ripple.**

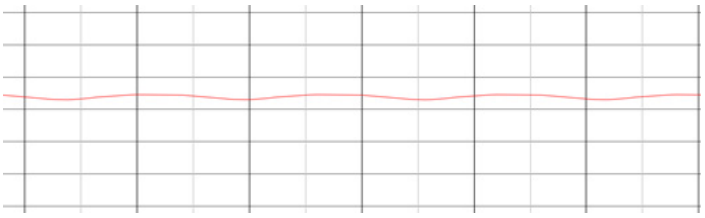


**Fig. 3b Cap-dominated output ripple.**

## The Schematic File

### QSPICE

After extracting the files from the ZIP file for QSPICE, the circuit file can be opened by clicking on the file with the QSPICE logo on it as shown below:


At the time this app note was written, you may encounter a window saying, “Update now”. If you have recently updated QSPICE, chances are you will not notice any difference with the new update. Therefore, it is recommended to save time by avoiding doing frequent updates.

The corresponding schematic file is shown in Fig 4. Assuming that the sim parameters match the design requirements, adjust the value of C4 by double-clicking on its value and editing it. The four sims shown in the plot file correspond to Cout = 50µF, 100µF and 500µF at ESR = 1mΩ. The fourth sim is for Cout = 500µF and ESR = 0.1mΩ.

If any of the four key parameters, Vout, Iout, Step Load and Slew Rate, need to be modified, proceed as follows:

1. Change Vout by clicking on VOUT_TARGET and edit it to the new desired value.
2. Calculate R4 by dividing Vout by the steady state output current. In this example, the steady state current is 75% of Iout, or 4.5A. This results in R4 becoming 0.18Ω.
3. Enter the dynamic current value, in this case 1.5A, into the pulse statement. This is the second value from the left of the pulse statement. Notice the minus sign, meaning that current is leaving the upper node (IEEE notation).
4. Calculate ton and toff by dividing the dynamic part of Iout by the slew rate. Enter these into positions 4 and 5 of the pulse statement.

The .tran directive is responsible for the data that is to be displayed. Since QSPICE is a new software, one will probably experience the sim not displaying the correct results for certain parameter and loading conditions. For this reason, a full record of the sim has been specified in the .tran directive.

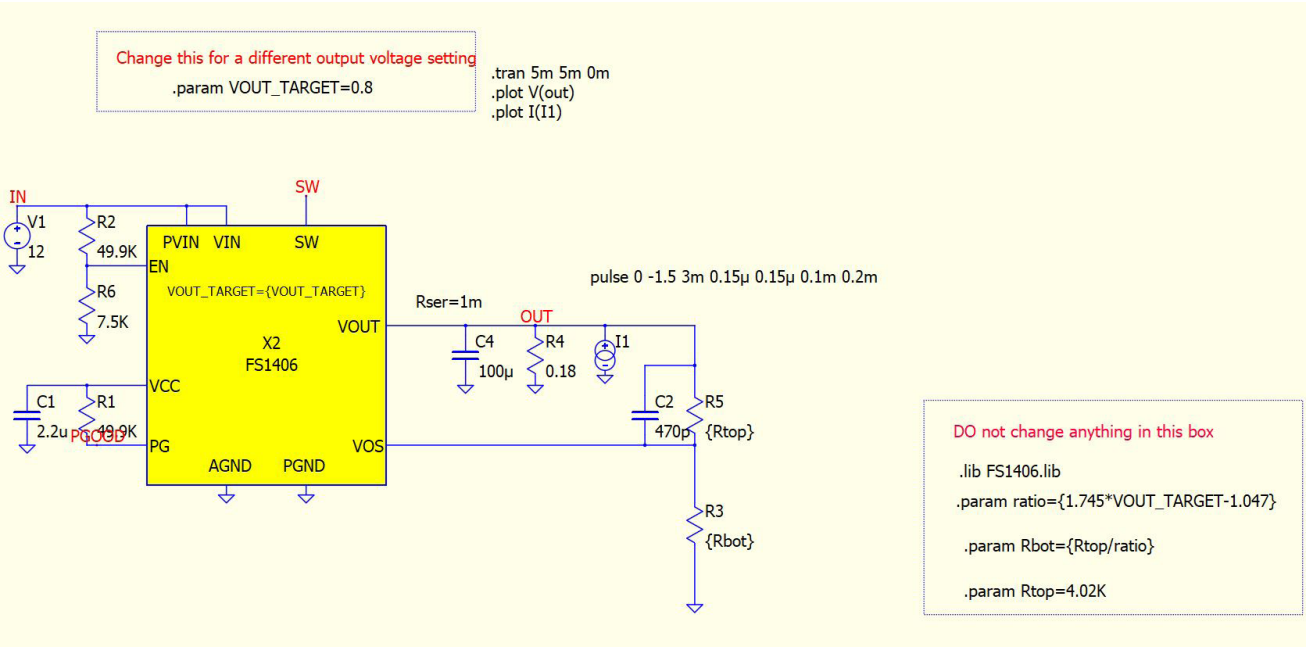



**Fig. 4 Schematic file for QSPICE.**

Fig. 5 shows a successful sim execution. The top waveform is the dynamic part of load current. It turns ON at 3ms, as specified in the pulse statement, after the output (bottom trace) has reached steady state.

Notice the glitch on the voltage during start-up. This and the steady state part of output voltage during the pulsed load condition need to be observed as not all Cout and ESR values will produce the expected results. Further developments are currently being made to address this issue.

To observe the transient response, zoom in on Vout by selecting one loading cycle on Vout. This case is shown in Fig. 6. If more detail is required on the Y-axis, right-click on the area left of the Y-axis. A window will pop up in which you can select the range of values to display on the Y-axis (Fig. 7 left side). You can also hit the Autorange button to fill up the vertical scale on the graph. Right-clicking on the axis label while pressing the CTRL key offers some interesting measurement utilities (Fig. 7 right side), something which can also be specified by a .meas directive inside the schematic file. The peak-to-peak measurement on Vout, for example, allows for the calculation of the AC regulation. By inspection of Fig. 6 one can see that the DC regulation is nearly ideal. Finally, the output ripple can be determined by further zooming in on it.

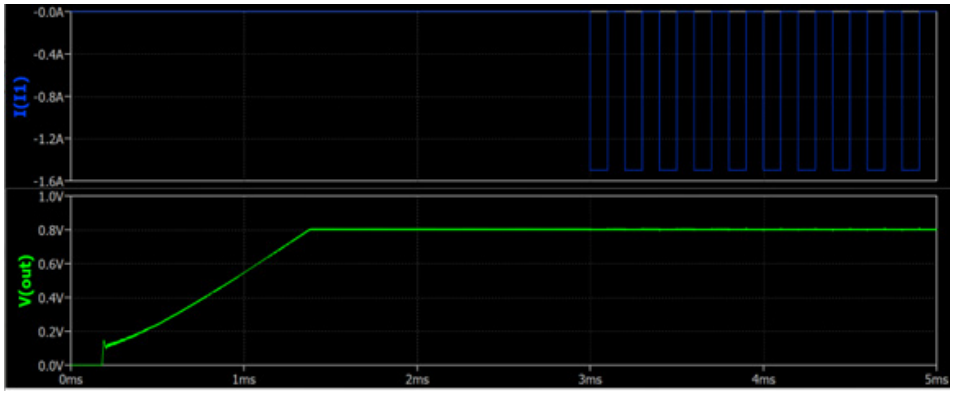



**Fig. 5 SIM output from QSPICE.**

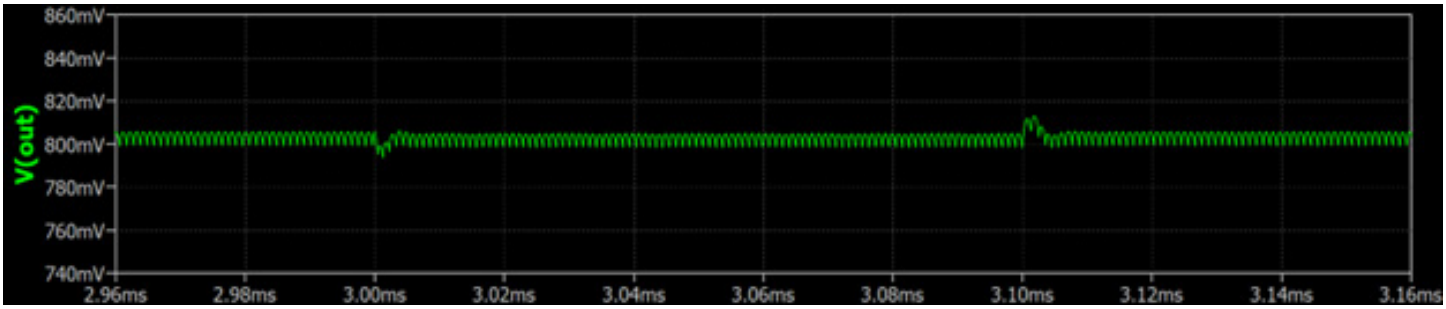



**Fig. 6 Zoom of Vout from Fig 5 during one load switching cycle.**

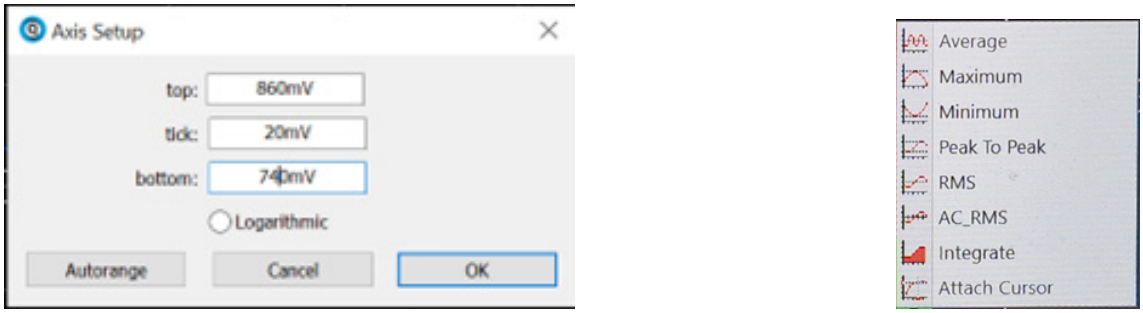
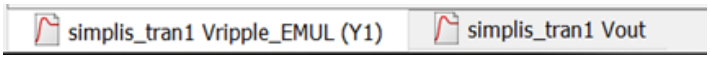



**Fig. 7 Windows showing how to modify the Y-axis values and perform measurements.**

## SIMPLIS

On SIMPLIS, the schematic file is shown in Fig 8. This schematic corresponds to the following parameters: Vout = 0.9V, Iout = 6A, Step Load = 40%, and Slew Rate = 4A/µs. Changing the values of C1 and R5 produces the results shown in the plot file for this sim.

After running the schematic, two tabs appear on the bottom of the screen as seen by the screenshot below. SIMPLIS displays the default results of the left tab. Click on the right tab to open the plot of interest, shown on Fig. 2.

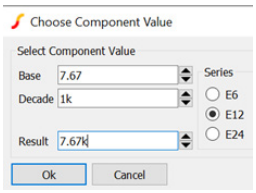



These plots have already been preconfigured to display the desired results. Below the graph you also see the peak-to-peak value of the transient response and the output ripple (see below):

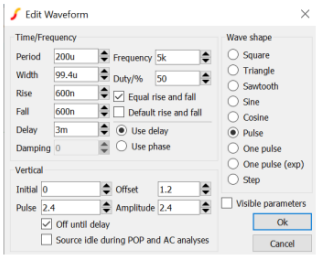
| Curve label | Name | Value |
|---|---|---|
| Vout | Peak To Peak | 27.341675mV |
| Vout | Peak To Peak (2.96ms - 2.98ms) | 5.0286455mV |

If the standard-defined parameters need to be modified, proceed as follows.

1. Select R2 from the values given in the datasheet. You may need to perform some iterations on R2 to display the target output voltage. Clicking on R2 you will see the following window:




2. Change the Base value; the Result will automatically adjust by itself.
3. Calculate R3 by dividing Vout by the steady state dc current. Enter this value into R3.
4. Specify the dynamic load parameters by clicking on the active dynamic load symbol. The following window will appear:




5. Since the dynamic load is 40% of Iout, 2.4A appears in the Pulse box. The values on the right boxes will automatically update. Now, divide 2.4A by the slew rate, which is 4A/µs. This gives 600ns. Because equal rise and fall times have been specified in the window, entering the rise time will automatically cause the fall time to also update.

## Conclusion

One can save a significant amount of time by simulating transient response. The two simulation tools presented here are SIMPLIS and QSPICE. The library tables containing the sims, and the models, will be updated on TDK's website as improvements become available.

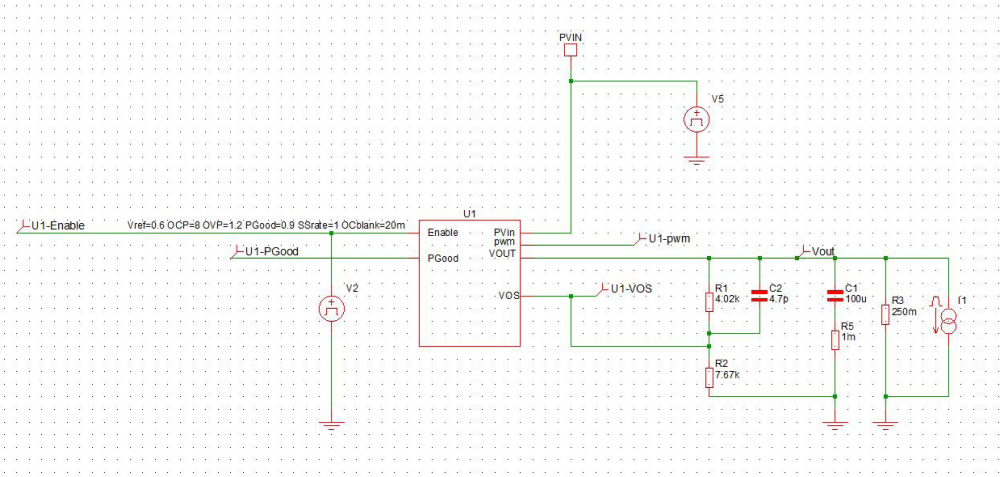



**Fig. 8 Example of a SIMPLIS schematic file.**