# Simple PWM Tutorial

## *Overview*

This simple tutorial is meant as a quick introduction on how a simple Sinusoidal Pulse Width Modulation (SPWM) scheme can be implemented on the RTDS Simulator. The example is fairly trivial but is nevertheless instructive. The circuit shown in Figure 1 will be assembled.

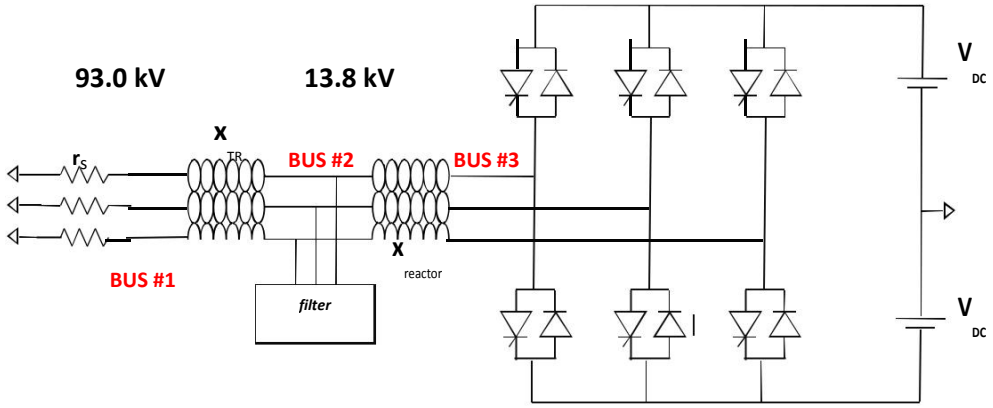


Figure 1

Ideal sources will be used for the DC Bus and a sinusoidal PWM scheme will be used to create a switched voltage waveform at Bus #3. That switched waveform will be filtered to produce a signal with reduced harmonic content at Bus #2. The voltage is then stepped-up to 93 kV and will supply a purely resistive load.

## *Step # 1*

The first step is to assemble the circuit which will be simulated. In this step the minimum amount of control components will be added. The circuit of Figure 2 should be assembled; a listing of component parameters is given. Explore the library to find the necessary components. In this first step no firing pulses should be provided to the two-level bridge.

*Reactor:*

L = 5.05e-4 H

*Filter:*

R = 0.90675 Ω

L = 1.1453e-4 H

C = 139.3 μF

*DC Bus:*

Voltage: 20kV

*Transformer:*

Primary Voltage: 53.69 kV

Secondary Voltage: 7.967 kV

Rating: 33.33 MVA

Winding Resistance = 0.0 pu

Winding Reactance = 0.1 pu

*Load:*

R = 86.49 Ω

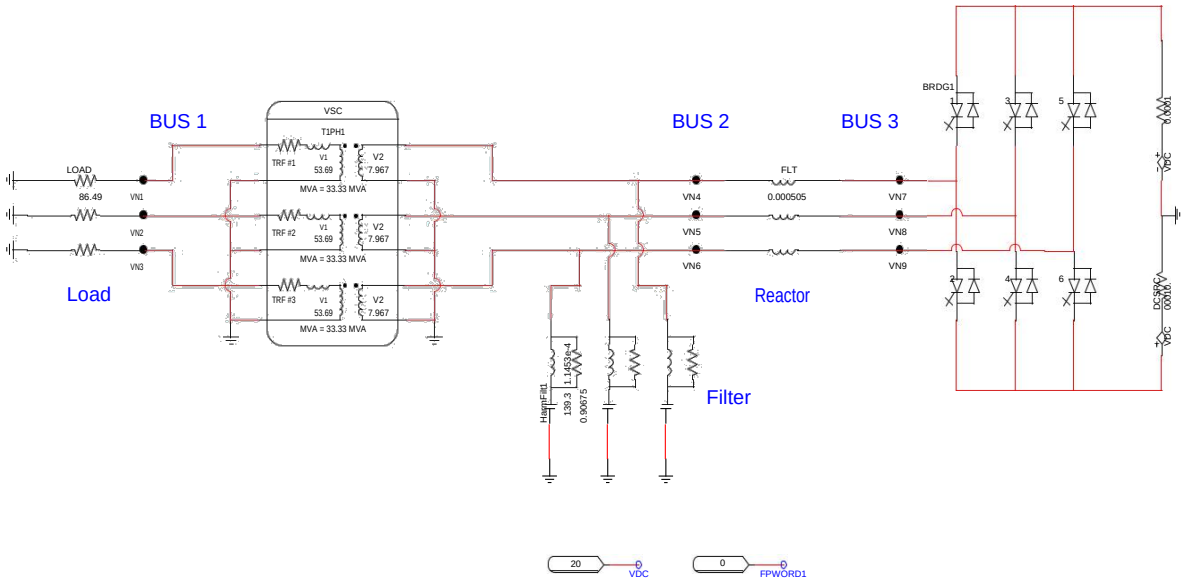


Figure 2

*Compile the case and create a simple RUNTIME interface where the 3-phase RMS voltages at Buses 1, 2 and 3 are measured. Are they as expected?*

## *Step # 2*

With the basic circuit assembled, the next step is to create the controls for our SPWM scheme. In a standard SPWM scheme a sinusoidal modulation signal is compared with a high frequency triangle wave as demonstrated in Figure 3B. When the value of the triangle wave exceeds that of the modulation signal then switch $S_1$ of Figure 3A will be turned off. Conversely, when the amplitude of the triangle wave is less than that of the modulation waveform then switch $S_1$ will be turned on. The switch $S_2$ will be controlled in a complimentary fashion.

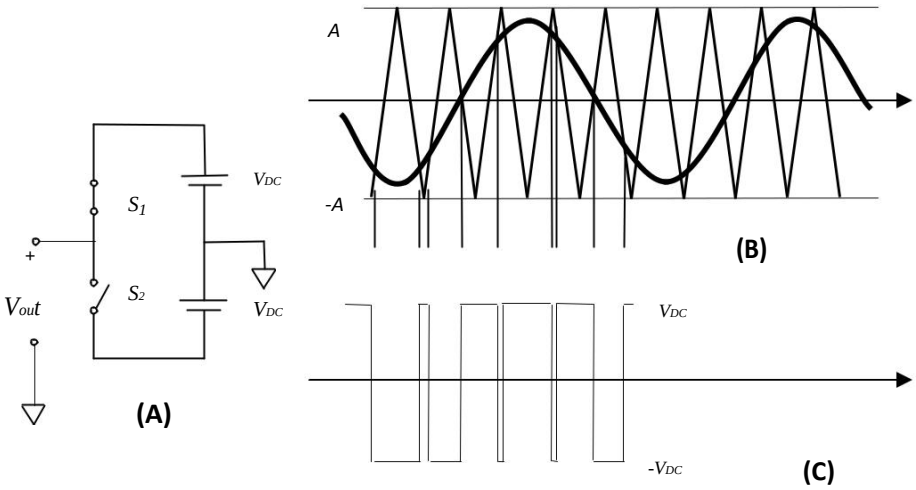


Figure 3

The signal $V_{out}$ from the circuit will be a switched waveform similar to that shown in Figure 3C. With the SPWM scheme that will be deployed it can be shown that the magnitude of the filtered signal that would ideally appear at each phase of Bus 2 will be ·, where *m* is the amplitude modulation ratio. Similar principles of operation apply to all three legs of the bridge.

The control circuit shown in Figure 4 can be used to create the firing pulses for the switches of all three phases. It consists of both large time-step and small time-step components. The large time-step components are needed to support the small time-step components.

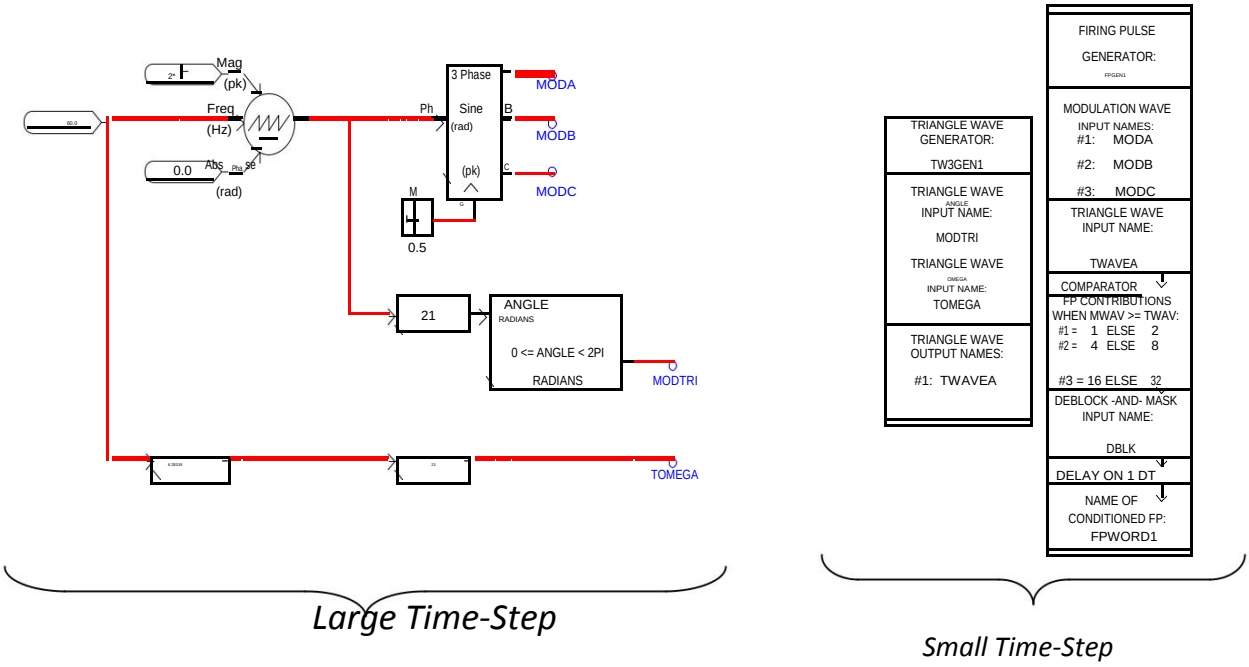


Figure 4

The large time-step controls create the modulation waveforms. These modulation waveforms will have the same wave-shapes as the voltages we are trying to create. The large time-step controls also generate two signals describing the high frequency triangle wave needed for the SPWM; these two signals are the triangle wave's phase (radians) and rate of change (radians/s).

The phase and rate of change information is used by the small time-step triangle wave generator (*rtds_vsc_TRIWAV3*) to create a high resolution triangle wave. The generated high resolution triangle wave is then used by another small time-step component (*rtds_vsc_3LGGIR*) and is compared to the modulation signals generated in the large time-step. This comparison must be done in the small time-step simulation in order to accurately determine the instant the switching event should occur. *Additional details can be found in the more comprehensive 'Simple STATCOM' exercise of the Small Time-Step Tutorial.*

*Compile your case and create a RUNTIME interface that looks similar to that of Figure 5. How do your simulation results compare to those shown? Change the amplitude modulation index, m. Does it affect the simulation as you expect it should?*

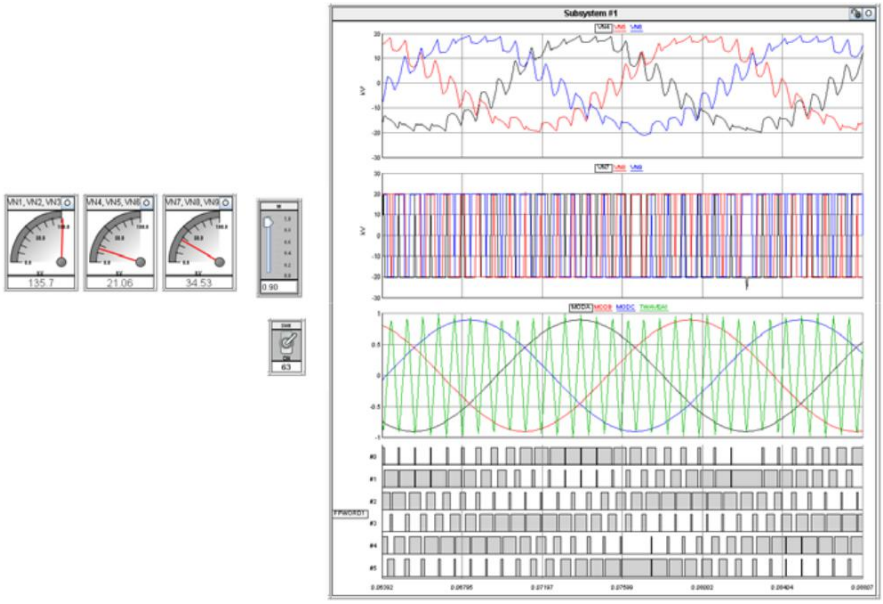


Figure 5

## *Step #3*

The control system from Step #2 can be modified so that a simple integral regulator keeps the voltage at Bus #2 at some desired set-point. Modify the controller as indicated in Figure 6 then compile and run the case again. *Does the regulator work?*

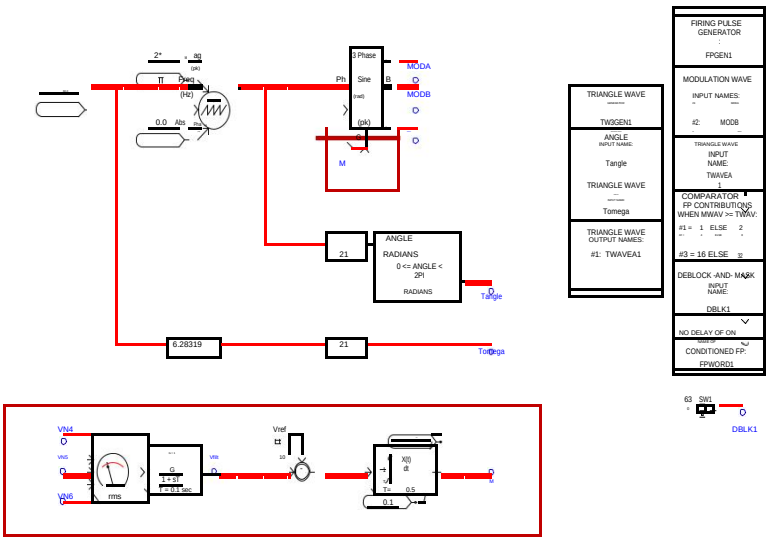


Figure 6

## *Step # 4*

Up until this point, all the simulations in this tutorial have consisted entirely of small time-step components. When possible this is a good option but it isn't always practical. The large time-step library is much more extensive than the small time-step library and sometimes the need for these models drives the need to interface the small and large time-step simulations. Also, small time-step simulation is relatively computationally intense when compared to the large time-step simulation so it doesn't always make sense to model everything with the level of detail provided by the small time-step simulation.

In order to interface the large and small time-step simulations a component called an *interface transformer* must be used. One side of the interface transformer will have only large time-step components connected to it; the other side will only be connected to small time-step components. **Whenever possible the circuit should be divided at a point where an actual transformer exists.** This is due to the fact that the interface transformer has a leakage reactance and a resistance. To minimize the impact of inserting the interface transformer into the circuit it is best to insert it a point where a leakage reactance and a resistance already exist. The presence of the leakage and resistance is a consequence of how the interface transformer is modeled.

Some slight modifications to the case from Step #3 will be made; the resistive load will be moved from the small-time step into the large time-step. Follow the instructions below:

(1) **Save the existing case under another name so that we can retain the case where everything is simulated using a small time-step.** (2) Replace the small time-step transformers with three single phase interface transformers (*rtds_vsc_IFCTRF1*). Figure 7 illustrates how this can be done.

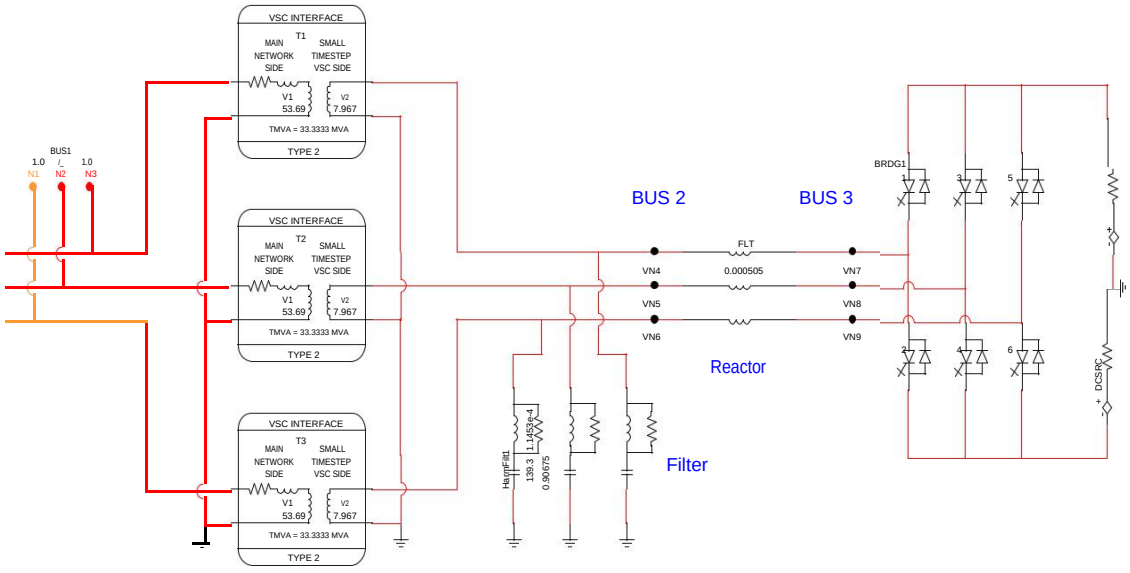


Figure 7

(3) Label the ‘Main Network Side’ of the interface transformers using a large time-step bus label. (4) Duplicate large time-step bus label outside the VSC Bridge Box and connect a resistive load as show Figure 8. (5) Compile the case and run it. *Does it work as expected?*

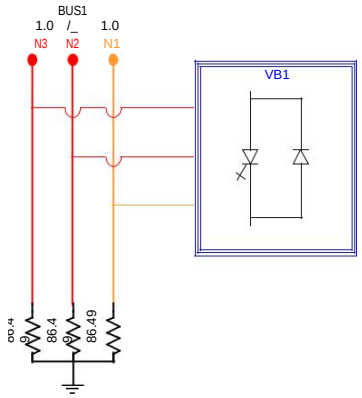


Figure 8