In early SDK versions the HWA and DSP implementations of the Doppler FFT module were different in scaling in the processing steps.
In the doppler FFT DPU we perform Doppler FFT per virtual antenna, and then do noncoherent accumulation across antennas. In particular, we sum the log2 magnitudes of the FFT outputs. In HWA this summation across antennas is done using the FFT engine, and save only the DC-bin (zero-bin) which represents the sum. Also while doing this we enable radix scaling in the FFT so that the output zero bin represents the average instead of the sum (but note in case of 12 virtual antennas scale is 1/16). For example

Number of virtual antennas	FFT size (2^ceil(log2(numVirtAnt))	number of radix stages	Output element of the range doppler matrix
2	2	1	1/2 SUM(256 * log2 $\|Y_k\|$)
4	4	2	1/4 SUM(256 * log2 $\|Y_k\|$)
8	8	3	1/8 SUM(256 * log2 $\|Y_k\|$)
12	16	4	1/16 SUM(256 * log2 $\|Y_k\|$)

where Y_k is Doppler FFT complex output symbol.
Also note that log2 magnitudes are coming out of the FFT engine in Q11 format, and after the noncoherent summation in Q8 format.
So, in visualizer, to convert to dB we divide the output by 256 (i.e. 256*3/4 for 12 antenna case) and multiply by 6 (6=20log10(2)).

However, in the early versions, in the DSP implementation of the Doppler FFT module this noncoherent accumulation was just summation, (not the average).
So, to convert to dB we divide by 256*number of virtual antennas and multiply by 6.

At some point we decided to unify the scaling of the these implementations and changed the DSP version to be the same as HWA version, i.e. use average instead of sum.
Because of that the formula in the visualizer changed.