RMS and PSD Estimation with the FFT
version 14 February 2019

1. Write a MATLAB function that estimates the RMS value of a time-domain voltage signal from evenly spaced samples of that signal. The only input to the function is a 1D array of the samples.
2. Write a MATLAB function that (a) computes the N point DFT from N time samples and (b) plots the magnitude and phase of the DFT values vs. frequency. You may not use any built-in MATLAB FFT functions. The input to the function is an array of N samples and the sampling frequency f_s. The function returns [f DFTresult] where DFTresult contains the DFT values corresponding to the frequencies in f.
3. Write a MATLAB function that (a) computes the N point DFT from N time samples and (b) plots the magnitude and phase of the DFT values vs. frequency. Use the MATLAB fft() function. The input to the function is an array of N samples and the sampling frequency f_s. The function returns [f DFTresult] where DFTresult contains the DFT values corresponding to the frequencies in f.
4. Very that the results of steps (2) and (3) match for voltage samples {8, 1, 7, 8, 5, 3, 2, 8, 2, 1} acquired at a sampling rate of f_s=100 Hz.
5. Write a MATLAB function that estimates the RMS value of a time-domain signal using the N point MATLAB FFT values. The input to the function is a 1D array of the time samples.
6. Sample the voltage signal
   $$v(t) = 120\,\sqrt{2}\,\sin(2\,\pi\,60\,t)\ \text{(V)}$$
   at f_s=1000 Hz for 10 cycles. Using the code from steps (1) and (5), estimate the RMS value. Compare to the theoretical RMS value.
7. Repeat step (6) for f_s=6000 Hz. Did the RMS estimate improve? Why or why not?
8. Repeat step (6) for f_s=100 Hz. Did the RMS estimate improve? Why or why not?
9. Repeat step (6) for a 60 Hz square wave with amplitude 120 (V).
10. Write a MATLAB function that returns samples of an N term Fourier Series approximation of a bipolar square wave of a given frequency and amplitude sampled at an arbitrary frequency. Verify your function by plotting your N=20 Fourier Series approximation for one cycle of the square wave of step (9). Then, compare the first 5 non-zero Fourier Series coefficient values to the largest FFT values scaled by $2\,N^{-1}$ for f_s=6000 Hz where N is the number of samples. What do you notice? Can you explain this?
11. Use the N=20 Fourier Series coefficients to estimate the RMS value of the square wave.
12. Write a MATLAB function that provides an estimate of signal power spectral density using equation (13.4.5) of [2]. The input to the function is [time samples sampling frequency]. The output is [f PSD] where PSD is an array of power values at the frequencies in f. Verify your that your PSD routine provides the correct RMS values for the sine wave and square wave samples of steps (7) and (9).
    Caution: this approach ignores the negative effects of using a finite number of samples. These effects can be reduced by using a 'windowing' function, see [2].

Prepare a technical report describing your results. Attach a listing of all code. Be sure your code contains explanatory comments. **You are cautioned that you must write your own code.**

References

[1] Frank L. Severance, *System Modeling and Simulation: An Introduction*, John Wiley & Sons, New York, 2001.
[2] Press et al., *Numerical Recipes in C*, 2nd ed., Cambridge University Press, 1992.
[3] J. A. Cadzow and H. F. Van Landingham, *Signals, Systems, and Transforms*, Prentice-Hall, Inc., New Jersey, 1985.