Digital Filters

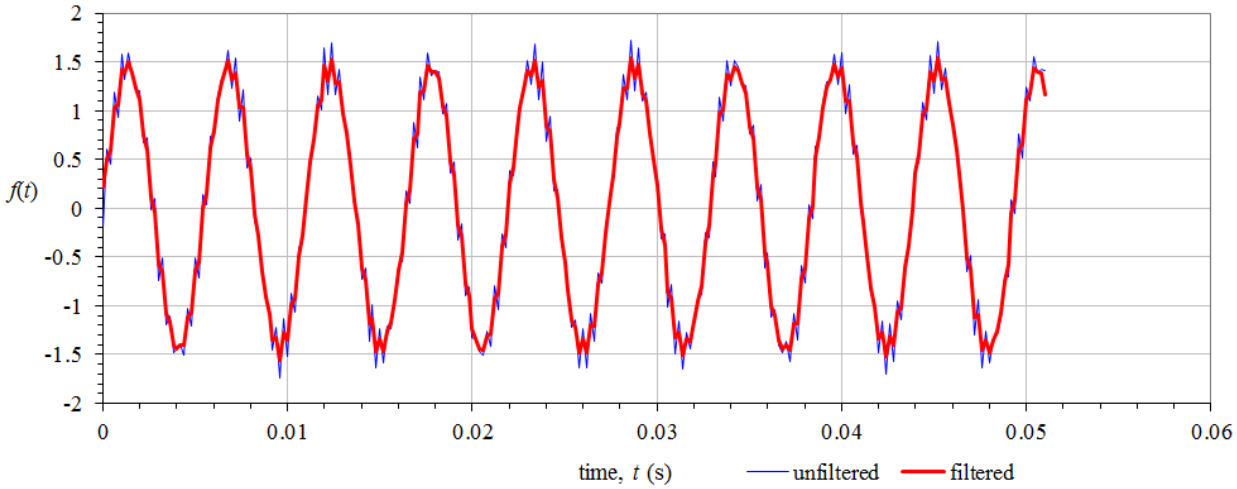
Author: John M. Cimbala, Penn State University
Latest revision: 15 October 2013

Introduction

- The filters discussed in the previous learning module are ***analog filters*** – **they filter the original *analog* signal *before* data are sampled digitally with a DAQ**.
- It is also possible to filter unwanted frequency components *digitally*, using ***digital filters*** – **they filter the *digital* signal *after* data have already been sampled digitally with a DAQ**.

Digital Filters

- There are many ways to generate a digital filter.
- A simple low-pass digital filter can be created by calculating a running average of the signal, beginning a few time steps *before* the current time step, and ending a few time steps *after* the current time step. The effect is to average out the high frequency oscillations in the signal, leaving only the lower frequency components.
- A more sophisticated method involves using FFTs and *inverse* FFTs, as described below:
  - Take an FFT of the time trace data.
  - Reduce the amplitude of the *undesired* frequency components of the FFT, using a simulated filter.
  - Take an *inverse* FFT of the reduced-amplitude FFT, which results in a new time trace. This new time trace is *filtered* – it has reduced amplitudes at the frequency components that were filtered.
- Some "tricks" are required to get this FFT/inverse FFT technique to work in Excel, and a sample Excel file is provided (with many comments) to use as a guide.
- The time trace before and after filtering with a first-order low-pass Butterworth digital filter (in Excel) is shown below.



- As can be seen, the digital filter has attenuated much of the high frequency noise, but has not removed it.
- A higher-order digital filter can be used, and it is possible to completely remove the high-frequency noise from the original signal. An example is shown below for the same case, but using a 4^{th}-order filter.