Technical University of Cluj-Napoca
Faculty of Automation and Computer Science

Subject:

Control Engineering II

Project:

Servo motor control using EMG:
Myoelectric Trainer

Student Name: Brezae Tudor
Group: 30331

2023-2024

General Description, Setup and Goals:

The QNET Myoelectric Trainer (MYOELECTRIC) is a specifically engineered device that facilitates the comprehension and practical demonstration of electromyographic signal processing principles. This system offers customizable configurations to employ diverse filtering techniques and control strategies for interpreting muscular signals, subsequently regulating the positioning of clamps on the servo mechanism.

Figure 1: QNET Myoelectric Trainer (MYOELECTRIC)

The device is operable through National Instruments' LabVIEW software, renowned for its graphical programming interface tailored for test system development. LabVIEW offers distinct productivity enhancements, including an intuitive programming paradigm, seamless instrument connectivity, and comprehensive user interface integration. Leveraging LabVIEW, users can employ a range of filters and signal processors to refine the input signal received by the Myoelectric Trainer, thus optimizing its performance.

Refocusing on the principal objective of this project, the primary aim is to enable users to effectively process electromyographic signals derived from muscle contractions sensed by the two connectors. The overarching goal is to engineer a signal transmission system capable of reliably actuating the servo-motor clamp, minimizing susceptibility to both electrical artifacts and ambient environmental noise.

The Power of Filters:

To advance toward our objectives, a pivotal focus lies in the exploration of filters. Within the realm of signal processing and noise mitigation, filters stand as indispensable tools, meticulously crafted to address diverse challenges and cater to the unique needs of users.

Within the LabVIEW application, users can harness the Express Signal Analysis Filter Block, offering an array of configuration options. Subsequent paragraphs will delineate these options in detail.

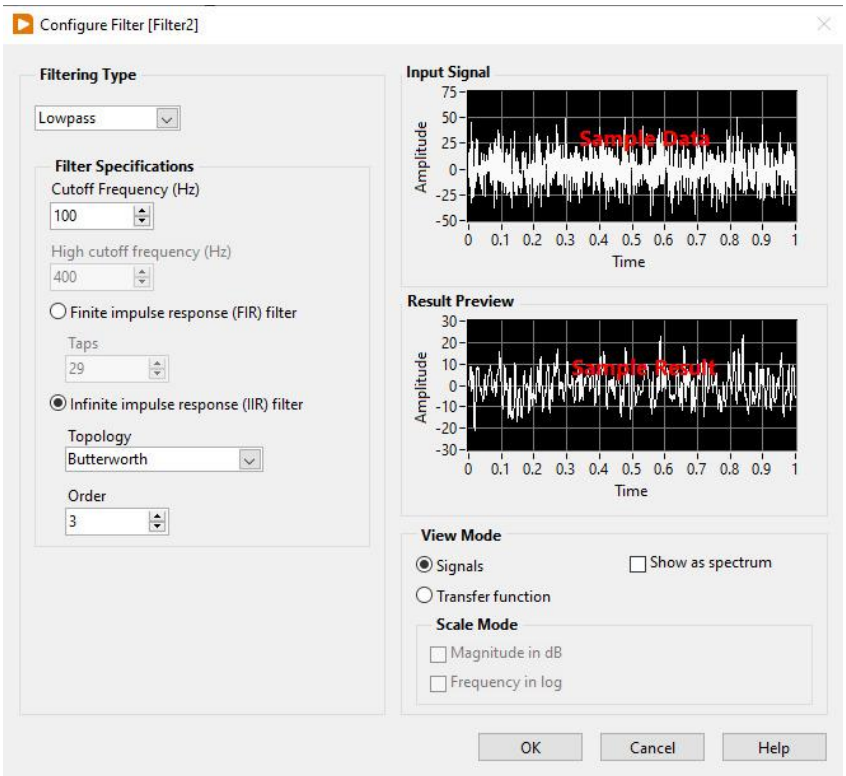

Figure 2: The Filter Block Configuration Window

Filtering Types are of 5 main categorizations:

- Lowpass Filter: A filter that attenuates signals with frequencies higher than a specified cutoff frequency, allowing signals with lower frequencies to pass through.
- Highpass Filter: A filter that attenuates signals with frequencies lower than a specified cutoff frequency, permitting signals with higher frequencies to pass through.
- Bandpass Filter: A filter that selectively passes signals within a specified frequency range while attenuating frequencies outside this range.
- Bandstop Filter: A filter that attenuates signals within a specified frequency band while allowing frequencies outside this band to pass through.
- Smoothing Filter: A filter designed to reduce noise and fluctuations in a signal by averaging neighboring data points, resulting in a smoother signal.

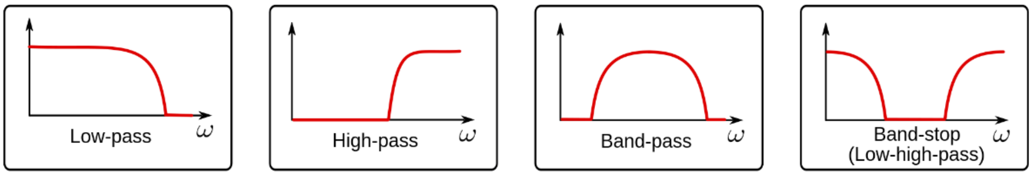

Figure 3: Visualizations for the four signal filtering types

Advancing our investigation into filter specifications, users can tailor cutoff frequencies following their chosen filtering type, and opt between Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) filters:

- Finite Impulse Response (FIR) Filter: A type of digital filter characterized by a finite duration impulse response, typically offering linear phase response and precise control over filter characteristics. FIR filters are computationally efficient and provide stable performance but may require more computational resources compared to IIR filters.
- Infinite Impulse Response (IIR) Filter: A digital filter characterized by an impulse response of infinite duration, commonly featuring feedback loops in its design. IIR filters are computationally efficient and can achieve sharper roll-off characteristics compared to FIR filters, but they may introduce phase distortion and stability issues, particularly in designs with high order or narrow transition bands.

Infinite Impulse Response (IIR) filters encompass various topologies, each with distinct advantages and disadvantages. The selection of a particular topology should be driven by the specific requirements of the user's application:

Type	Passband	Stopband	Roll-off	Step Response
Butterworth	Flat	Monotonic	Good	Good
Chebyshev	Rippled	Monotonic	Very Good	Poor
Inverse Chebyshev	Flat	Rippled	Very Good	Good
Elliptic	Rippled	Rippled	Best	Poor
Bessel	Flat	Monotonic	Poor	Best

Figure 4: Comparison of Filter Approximations
(credits: @ALLABOUTELECTRONICS on YouTube)

Achieving Results:

To accurately analyze the signal generated by muscle contractions, users must initially assess the unfiltered data using a Fourier Transform MATLAB code. This facilitates the examination of the spectral composition of the signal, aiding in the identification of various frequencies present and assisting in determining an approximate cutoff frequency for subsequent filtering.

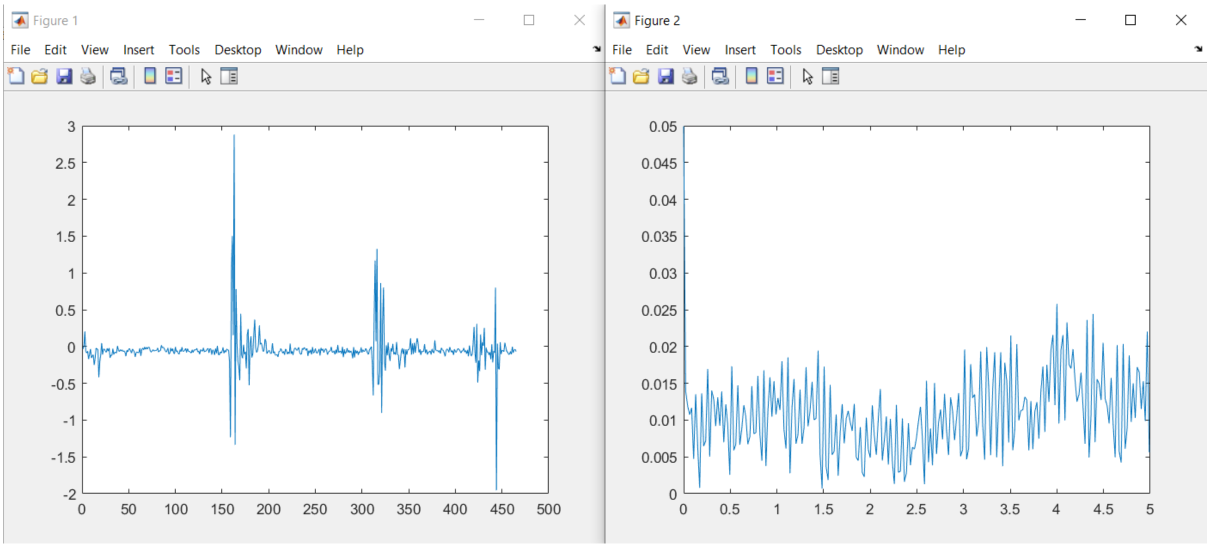

Figure 5: The Unfiltered Signal and its' Fourier Transform graph

Based on the Fourier Transform graph and considering filter roll-off, a suitable cutoff frequency of 1Hz is determined. Subsequently, opting for the Infinite Impulse Response (IIR) Filter aligns with our requirements. Further analysis of the Topology table indicates that the Bessel topology is ideal due to its flat passband and superior step response. Determining the optimal filter order entails iterative refinement to achieve the most favorable outcome.

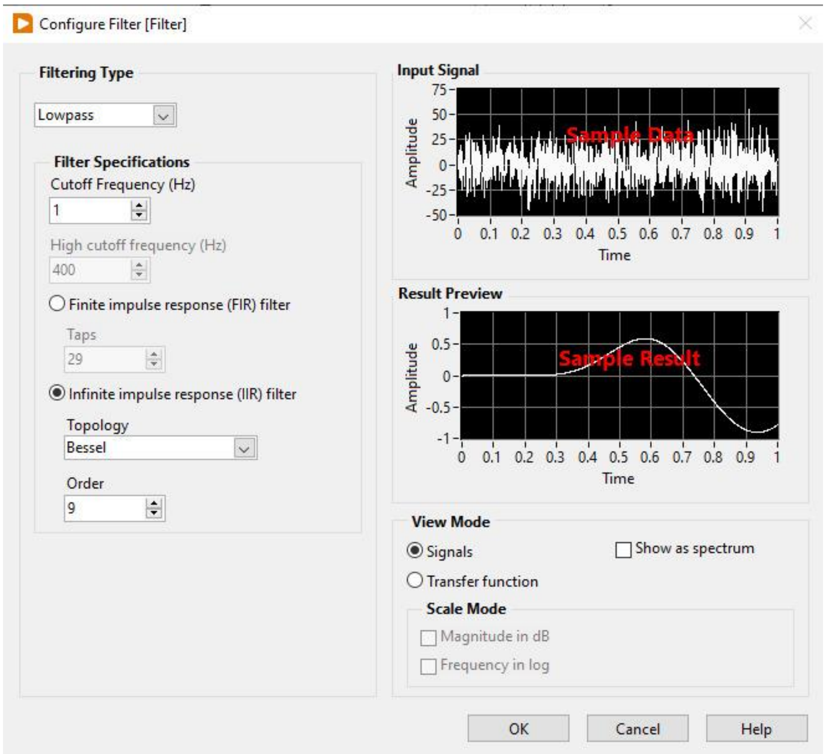

Figure 6: The most favorable configuration for the Filter Block

With the chosen configurations, we effectively attenuate noise, enabling the servo-clamp to exhibit optimal and discernible motion during both opening and closing phases.

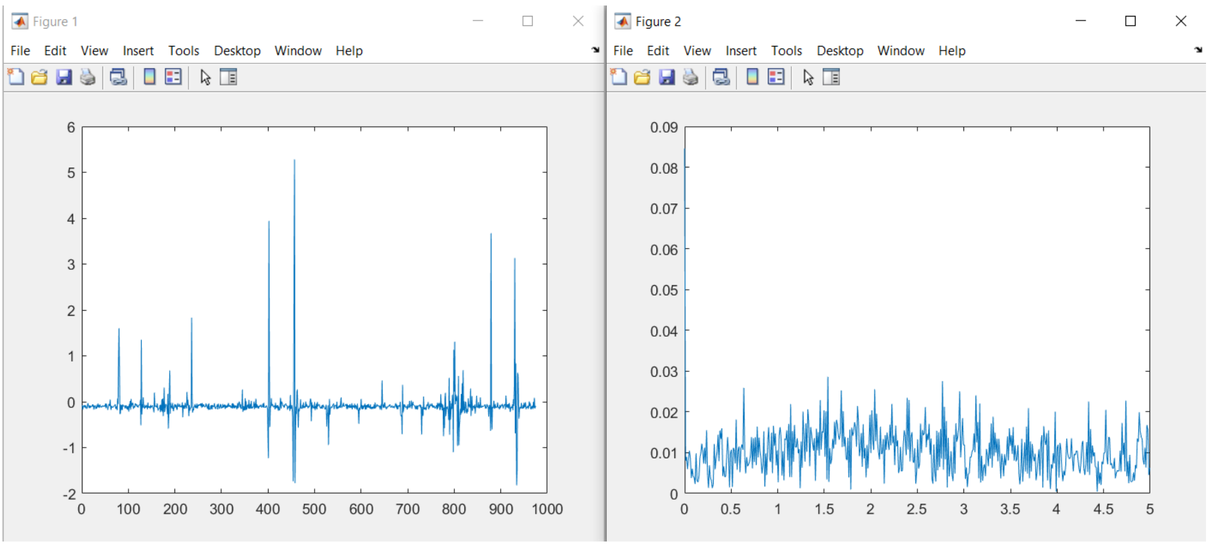

Figure 7: The Filtered Signal and its' Fourier Transform graph

Conclusions:

In conclusion, this project has served as an invaluable platform for acquiring knowledge and conducting in-depth research into signal filtering and its intricacies. By meticulously considering various parameters and influencing factors, we have achieved a result that is both gratifying and commendable. The Myoelectric Trainer emerges as an exceptional tool for delving into real-world signal processing applications, offering users a foundational understanding of muscle signal processing—fundamental to advancements in fields such as artificial limbs, prosthetics, robotics, and engineering at large.

Appendix:

Matlab Code for Fourier Transform:

```
Input = DataFiltered.VoltageVPosition;
Ts = 0.1;
Fs = 1/Ts;
N = length(Input)
f = Fs*(0:(N/2))/N;
figure
plot([1:N],Input);
figure;
Y = fft(Input);
P2 = abs(Y/N);
P1 = P2(1:N/2+1);
plot(f,P1)
```

LabView Simulation Interface: