Design of an SEMG Signals-Robotic Arm Human Machine Interface for Arm Trajectory Teleportation

Mwangi Alex W   Brian Ndambia M   Brian Munene K

October 5, 2023

Introduction

Our project involves developing a collaborative robotic arm that imitates human arm movements.The essence behind creating such a system is primarily to offer safety in industrial operations where human involvement pose either a health or accidental risk.The human hence can conduct the operations from a safe distance.We pick up sEMG signals from different regions of the arm of the human, filter and process the signals, extract necessary features from the signals and subject the features to a ML model that eventually controls the robotic arm.An example of a proper use case would be in nuclear power plants where operation in some areas could expose the human operator to harmful radiations.RBEC sought to actualize this project using the available resources in the university while not compromising the performance of the system therefore, optimizing cost and performance.

Abstract

The project can be divided into four main parts - Signal acquisition and conditioning which makes up the data acquisition unit(DAQ),digital signal processing(DSP),feature extraction, modelling and robotic arm kinematic model development.The DAQ is responsible for collecting surface signals from the muscles using specialized electrodes and circuitry. The signal is digitized and the magnitude vs time real-time graph of the digitized signal can then be plotted on an oscilloscope or a microcontroller serial plotter. Here specialized electrodes are properly attached to the skin on an area where the surface electromyographic (sEMG) signals are required and the signal is picked. The signal is directed to an instrumentation amplifier.The instrumentation amplifier in addition to a subsequent amplifier stage acts as a preamplifier. After acquisition, conditioning follows, which entails filtering the signal such that its frequency bandwidth falls within that of sEMG signals by eliminating noise.The preamplifier stage ensures processability of the signal.The raw signal from the instrumentation amplifier belongs in the mV range and would not be reliable for processing.Conditioning can be divided into two parts; analogue filtering and digital signal processing. Analogue filtering involves designing a particular configuration of analog electrical components inform of resistors, capacitors and amplifiers to fit the signal within the bandwidth 0-500Hz.Here, we implemented a lowpass filter and a notch filter as will later be seen.Digital signal processing happens on the MCU and is implemented as an embedded application.

Before passing the signal to DSP schemes, it has to be digitized. In this case, we utilized the MCU's ADC.

In DSP we implement the moving average filter algorithm to the acquired and digitized signal.The moving average filter is a low pass filter that eliminates noise coming from power line interference and motion artefacts.With an MA filter, the signal also achieves more stability. For the feature extraction stage,we implement the recursive moving RMS algorithm.The moving RMS technique utilizes the present and past samples of the signal for a particular length to calculate the RMS values in real time.

Modelling is the technique employed to analyze and evaluate the features extracted from the sEMG signal so that intended instructions are passed to the robotic arm.It is the bridge between the determined features of the collected signal and the robotic arm's actuation algorithm.That way,modelling can work in two ways. One way is taking the data from the feature extraction stage such as the moving RMS data from the moving RMS algorithm as the input and producing the corresponding spatial dimensions, speed, direction of movement of the volunteer's arm characterized by the collected signal. The outputs are then fed to a inverse kinematics algorithm that determines the individual motor angles and positions and angular velocities which are directly passed to the actuators of the robotic arm.This can be achieved either by adopting machine learning algorithms that would require training a model to determine characteristics of the movement from the extracted feature data.Another way would be to use the extracted features to directly determine the corresponding angles for the motors. This would need less computing power but would produce unreliable movement teleportation.For status, our group decided that it would be important to approach this stage at the beginning in a simplistic manner and begin by detecting when there is any kind of movement prior to adopting machine learning models to test responsiveness and accuracy of the acquired data. Movement of any kind is characterized by a change of amplitude of the signal for a particular time when the muscles flex that would in turn correspond so particular length of the moving RMS data being higher than normal. The normal case would be when muscles aren't flexing. We implemented as simple model referred to as the thresholding technique that produces binary outputs, 1 or 0 when the RMS values are above or below a determined threshold value to indicate whether there is movement or not.The binary outputs are then utilized to trigger any reactions such as trajectory functions or LED blinking.It is however important to remember that the thresholding model only plays the role of identifying when there is movement(even attempted) or not.

Modelling produces spatial coordinates for the position that the patient's arm is at or intends to be. In order to teleport the same position to the robotic arm, an inverse kinematic algorithm is necessary to translate the spatial coordinates into motor angular positions that would take the robotic arm to the same position.Creating a kinematic model involves developing an inverse kinematic algorithm which generates required motor angles to reach a desired spatial dimensional position. This can be achieved either by using algebraic techniques in simplified manipulators or analytical techniques as with the Newton-Raphson method.In this case, algebraic techniques were employed since the robotic arm we used (5 DoF) was simplified to a 4 DoF by ignoring one joint for simplicity. The trajectory algorithm was developed on the basis of system of equations that were solved to find the solutions in form of motor angles to achieve the same trajectory as the patient. However angle limits need to be defined to match with those of the servo motors in the robotic arm.

Data Acquisition Unit

Preamplification

Figure 2 shows the data acquisition unit that was used to acquire and condition the SEMG signals. INA125P was our choice of instrumentation amplifier (IA) owing to its reliable performance and voltage offset characteristics.It offers an edge over other IAs such as AD620, INA128 and LT1167 in that it has additional voltage offset pins that can be connected to make the collected signal have desired voltage level characteristics.sEMG signals are picked up by the positive and negative terminals of the INA125P labeled by the EMG-1+ and EMG-1-nets on figure2.It was observed that the output signal was dependent on the order in which the input signals were placed.That is to mean that by interchanging the order of placing the input signals from the electrodes at the non-inverting and inverting terminals, the output was different. Hence if one configuration produces an unreliable or no output, the order can be changed. The reference electrode is connected to the IA at the reference pin through a voltage divider. Capacitors are connected in parallel to the resistors of the divider to decrease inductive noise and eliminate DC offsets.

The output is tapped with reference to the floating ground rail.Both the floating ground rail potential and the reference pin are raised in voltage to a level half the supply voltage(+2.5V) which now acts as our ground.This is because the output swings about this potential and in order to match the voltage specifications of the output to those of the ADC/MCUs ,we raise the output to a level that can be readable by the ADC/MCU.One must be careful however not to connect the reference pin (or its rail) to the floating ground rail otherwise the signal will not be seen after the amplification stage in the serial plotter. Another way is to connect the floating ground to a variable voltage by choosing one of divider resistors as a potentiometer that can be varied to achieve any voltage within the supply voltage range.The wires connected to the electrodes needs to be as still as possible during collection of sEMG signals. Motion artefacts causes the reference voltage of the signals to 'bounce', a phenomenon we referred to as ground bouncing.However, this effect can be mitigated by implementing a moving average filter in the DSP stage as shall be seen.The Arduino serial plotter was preferred for signal waveform representation over the oscilloscope. After many attempts with the oscilloscope, it was observed that it was incapable of representing the sEMG signal properly owing to power line interference such that the sEMG signal was masked by noise. Upon using the Arduino serial plotter, a smooth signal that resembles the convectional sEMG signal was achieved.This is somewhat counter intuitive as one would expect the oscilloscope to be a better option to represent this signal.

Filtering

To design appropriate filters, the signal had to be analyzed in the frequency domain.This means that we needed to subject collected samples of the signal after the preamplification stage to an FFT algorithm that would convert it into the frequency domain. This was achieved by saving the acquired Arduino serial monitor values and passing them to MATLAB. In MATLAB we designed an FFT algorithm that uses the passed samples to generate an FFT plot as shown in figure 1.

The output was then obtained as a graph that depicts where most of the frequency component of the signal resides in. The FFT graph in figure 1 shows that the most dominant frequency of the signal was approximately 10Hz.Beyond

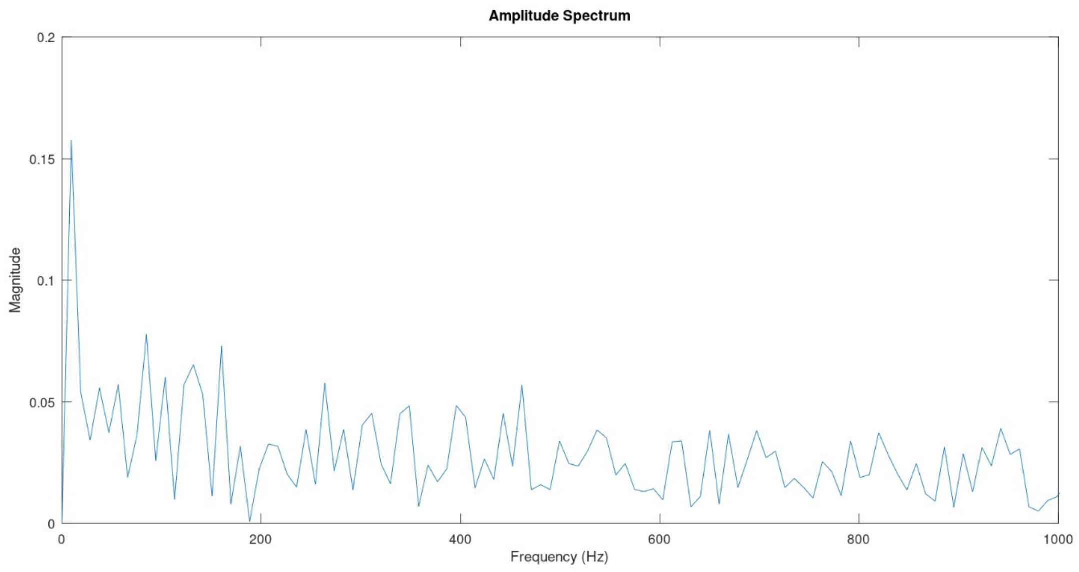

Figure 1: An FFT plot of the acquired EMG signal. The highest peak is observed to be at around 10Hz.The magnitude of the plot becomes insignificant at frequencies beyond 500Hz

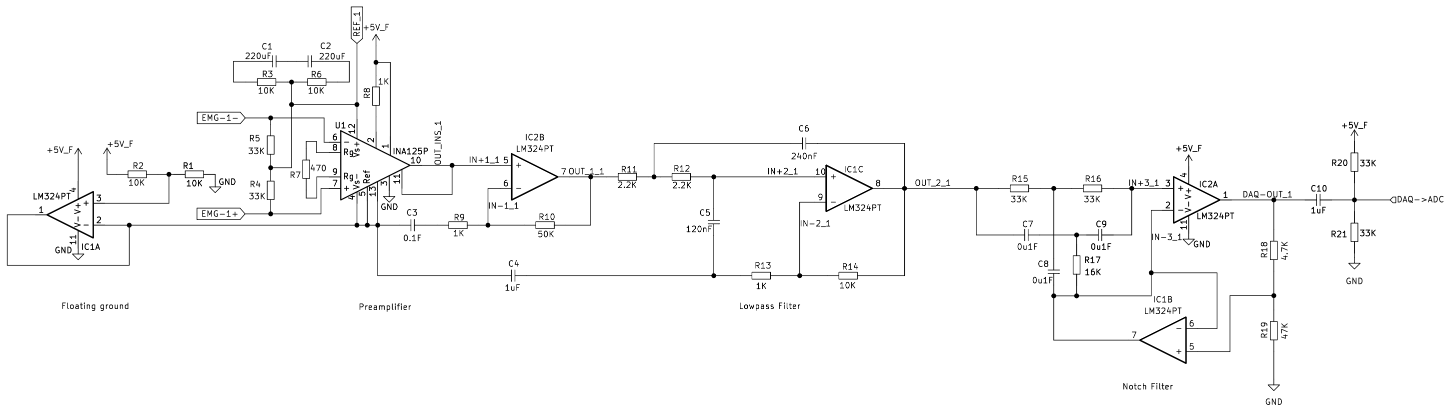

Figure 2: A schematic for the Preamplifier, low pass and notch filter,entirely referred to as the DAQ.

500Hz, the signal loses its power.Therefore, we decided to design a low pass filter with a cut-off frequency of 500Hz. Designing a high pass filter with a cutoff frequency of below 10Hz means it had to be of high order since that would require a high roll-off filter. High order in turn means multiple stages that would unnecessarily add real estate to the DAQ yet the frequencies are not substantial. We designed a butter-worth sallen key topology low pass filter to pick up the SEMG signals and condition them such that their gain and frequency characteristics are within the required characteristics.

The schematic for the low pass filter can be seen on the third stage of figure 2.

We also needed to eliminate 50Hz noise from power line interference. The best candidate for such a task would be a notch filter designed to attenuate 50Hz noise to a desired level.The notch filter we designed can be seen on the fourth stage of figure 2.After designing all the three stages(Preamplifier,Lowpass filter and notch filter), we combined them and simulated them to obtain their frequency characteristics. Figure 3 shown the AC sweep of the three stages using Multism.Shown by the upper graph of 3,the IA is indicated by a red line, low pass filter indicated by a green line and the notch filter indicated by a blue line. The frequency response for the DAQ is indicated by the blue line which is the output from the notch filter which corresponds to the output of the DAQ. A sharp descent is observed at around 50Hz for the blue line which indicates attenuation of 50Hz noise from the power line. The blue and green line intercept with the red line at approximately 472Hz above which the signals are attenuated below the input signal level. The lower graph depicts the phase response which happens to be of least importance in this case.

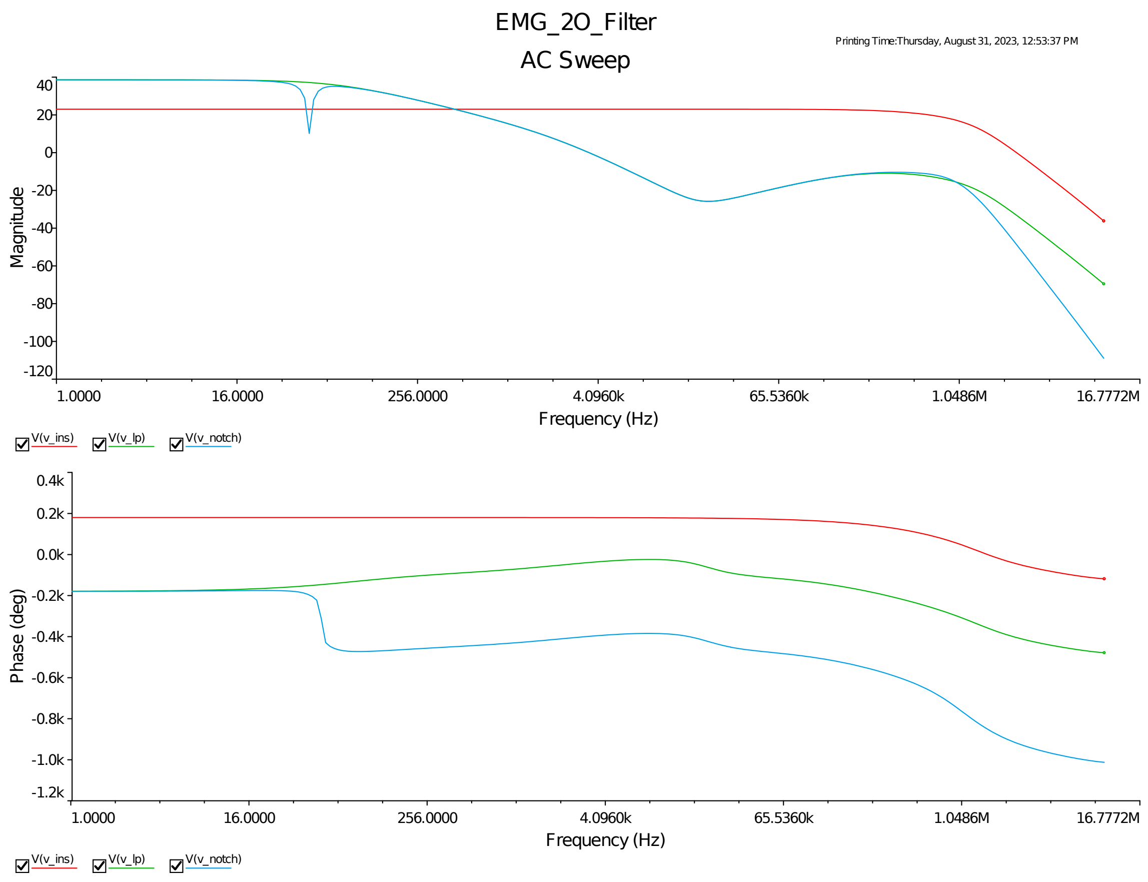

Figure 3: A Multism simulation for the combined three stages of the DAQ.

Digital Signal Processing

Moving Average Filter

The Moving average system is essentially a low roll off low pass digital filter.It eliminates power line interference and motion artefact disturbances.Represented as an equation, it has the form;

$$y[n] = \frac{1}{M_1 + M_2 + 1} \sum_{k=-M_1}^{M_2} x[n-k] \tag{1}$$

where M1 and M2 are constants that determines how much past samples the filter considers during computation. y[n] is the output while x[n-k] are the delayed versions of the input samples.Upon deriving the impulse response of the filter, which is equivalent to the output when unit samples are fed to the system as inputs we get;

$$h[n] = \begin{cases} \frac{1}{M_1+M_2+1} & -M_1 \leq n \leq M_2 \\ 0 & Elsewhere \end{cases} \tag{2}$$

To implement the moving average filter, we would need to convoluted the impulse response with delayed versions of the input samples as follows;

$$y[n] = \sum_{k=-\infty}^{\infty} h[n]x[n-k] \tag{3}$$

Equation 4 is implemented as an embedded system on a microcontroller.With the knowledge of structures, arrays, pointers and functions the moving average filter was implemented using embedded C both in Arduino MEGA and Nucleo-F446RE

Feature Extraction

Moving Recursive RMS

The recursive moving RMS feature is an improvised way to implement the moving RMS such that memory utilization and speed of computation are optimized. The moving RMS is realised by implementing the equation,

$$y[n] = \sqrt{\frac{1}{L}\sum_{l=0}^{L-1} x^2[n-l]} = \sqrt{\frac{1}{L}(x^2[n] + x^2[n-1] + x^2[n-2] + ... + x^2[n-(L-1)])} \tag{4}$$

Where,x[n-l] is the delayed versions of the input samples of the digitized signal.L is the length of the window that is considered during computation. L has to be chosen optimally in order to achieve reliable results.Such an equation implemented would however utilize quite a lot of memory and time during implementation in an embedded real time system.Utilizing recursive techniques would solve those problems.

let eq 4 be written in following two equivalent forms

$$y^2[n] = \frac{1}{L}(x^2[n] + x^2[n-1] + x^2[n-2] + ... + x^2[n-(L-1)]) \tag{5}$$

equation 5 can be shifted to get,

$$y^2[n-1] = \frac{1}{L}(x^2[n-1] + x^2[n-2] + ... + x^2[n-L+1] + ... + x^2[n-L]) \tag{6}$$

eqs 5 and 6 can be used to get the form,

$$y^2[n] - y^2[n-1] = \frac{1}{L}(x^2[n] + x^2[n-L]) \tag{7}$$

finally

$$y^2[n] = y^2[n-1] + \frac{1}{L}(x^2[n] + x^2[n-L]) \tag{8}$$

Equation 8 was implemented both on an Arduino MEGA and a Nucleo-F446RE.The goal of the recursive moving RMS is to ensure that for a particular number of past input samples(defined by the choice of L) of the digitized sEMG signal being collected and stored in the microcontroller memory in real time, we evaluate the addition of the magnitudes of the square of the samples and add them together. This values can then be utilized in the modelling stage to assess when the volunteer's nature of movement characterized by the collected signal.Ultimately,the recursive moving RMS serves the role of a filter that gives a signal that can be easily used to evaluate the characteristics of the sEMG signal

Modelling

Modelling involves utilizing the collected features of the signal for the purpose of producing necessary commands to the robotic arm.There are two ways to achieve this.One is by adopting machine learning techniques where a data set of collected feature data referred to as the training data set is used to train a model that produces a trajectory in terms of a set of spatial coordinates that match that of the human arm. By so doing, the model serves as the link between the collected sEMG signal and the robotic arm.The other way is to employ thresholding. In this context, we determined as particular value during the feature extraction stage where if the feature data surpassed a given condition, a desired action was undertaken.Thresholding becomes a suitable way of modelling a system for status before employing machine learning so as to test the accuracy and reliability of the data from the DAQ. In this case the feature we utilized was the RMS. Since the value RMS corresponds to the extent of muscle contraction/movement on the human arm, the average RMS value was determined before the volunteer made any significant arm movements.This value is what we refer to as the threshold and the basis upon which we adopt the name thresholding.When the user began moving his arm, the RMS values increased depend on muscles contraction level and the angle of movement. Here we focused on the angle of movement. We sort to map particular values of RMS to to particular joint angles on the robotic arm by tracing the difference between the RMS values and the threshold while mapping the difference to the servo motor controll algorithm that in turn controls the motor joints.

Trajectory Planning

Inverse kinematics entails determining the angular values that will help us achieve desired spatial coordinates for the end effector. I this case, we intend to develop an inverse kinematics model for a 4 DoF robotic arm. There are 2 techniques that can be employed to determine inverse kinematics; numerical techniques and algebraic solutions. The numerical method entails having initial guessed angles and the desired spatial coordinates. The initial guess is fed to a forward kinematics model that calculates the position.The error between the calculated position and true expected position is calculated.The error is minimized iteratively by adjusting the angles guesses until they are close enough to the angles that correspond to the true position.However, this method entails a lot of iterations that presents a running time problem therefore it is not preferred.Additionally,it depends on the reliability of the initial guess of the angles upon which the correct angles are calculated. In cases where the initial guess deviates too much from the true angles, the solution diverges hence becoming problematic since solutions cannot be found.The algebraic method entails using a system of equations that are analytically derived from knowledge of trigonometry. This is the technique we employ in this case.The angles we intend to determine are 4; the base joint, the shoulder joint, and the elbow joint. The figures we used for analysis are as shown on figure x and y. They present the geometry of the robotic arm. In figure x r is the length of the shoulder link.