Digital Sensors and Digital Filtering with CyDAQ

Introduction:

In this lab, we'll use the CyDAQs digital sensor capability to collect data from the Sparkfun MPU-9250 **IMU**(or **I**nertial **M**easurement **U**nit). The IMU is capable of detecting digital signals from both an accelerometer (which measures acceleration (rate of change of velocity over time)), and a gyroscope (which measures the angular velocity). For this lab, we experiment with using the accelerometer to determine when the direction of motion of the sensor changes.

Motivation:

We consider an interface that requires the user to quickly move their hand as input. For the purpose of this lab experiment, we can assume that the IMU is securely attached to the users hand. While typically a user would not have an issue moving their hand such that our system can differentiate between moving or not moving, some users may suffer from Essential Tremors, a common neurological disorder that can result in an impaired control of the hands or fingers. As a result, users with this condition may not be able to provide input that is clearly identified as either 'Moving', or 'Not Moving'.

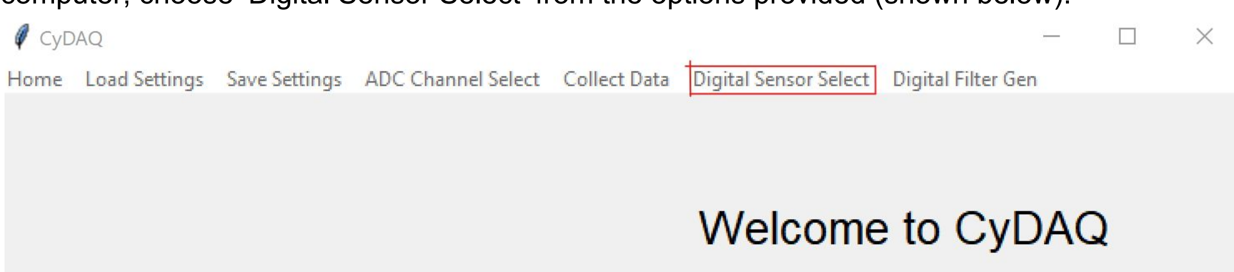
INSTRUCTOR VERIFICATION:

Prelab: Briefly read about Essential Tremor on Wikipedia. The article states a specific range of frequencies the tremor occurs in. What is this range?

We wish to apply a digital FIR filter using the CyDAQ Digital Filtering tool to clean up data collected from the accelerometer sensor. In doing so, we will further explore uses of digital filtering.

Part 1: Collecting Data:

To collect data, we will use the CyDAQ. After opening the CyDAQ GUI on your computer, choose 'Digital Sensor Select' from the options provided (shown below).



By default, the 'IMU_ACCEL' option is already selected. We'll use this option to tell the CyDAQ that we are collecting information from the Accelerometer on the IMU, rather than the Gyroscope or Temperature sensor. Press the 'Select' button to proceed.
On the next screen, we'll see that the CyDAQ will sample on the digital sensor at a rate of 100 Hz.

After selecting the appropriate COM port, you will press the ‘START’ button to begin recording data. Then you will finish collecting data by pressing the ‘STOP’ button.

Start by simply collecting data from the sensor as it sits on the table. Once you are able to plot this data in MATLAB, you will see that even when at rest, there is some amount of noise in our measurement.
Next, Collect Data by holding the sensor in your hand and moving your hand left and right in a fast motion. For example, move between left and right positions every second. One possible strategy would be to use a metronome set to 60 beats per minute. This would count out seconds, allowing you to time your movements to some sort of reference.

INSTRUCTOR VERIFICATION:
1(a): Plot the resulting data in both Time and Frequency Domain.

Next, we repeat the same data collection while attempting to simulate a tremor in our hands. We should still try to move from left to right with a consistent period, while also contributing a noticeable amount of noise from shaking.

INSTRUCTOR VERIFICATION:
1(b): Plot the resulting data in both Time and Frequency Domain.

Part 2: Cleaning Data:
Next, we wish to clean our noisy signal. To do so, we’ll need to design a low pass filter. First, we note that an ideal Low Pass Filter has the following impulse response:

$$h[n] = \frac{\omega_c}{\pi}\frac{sin(\omega_c n)}{\omega_c n} \quad \text{(Eqn. 1)}$$

Unfortunately, this is not practical as is. Reflect on this and be able to verbally explain why this implementation is NOT practical.

In order to implement our low pass filter, we’ll truncate our transfer function such that:

$$h[n] = \frac{\omega_c}{\pi}\frac{sin(\omega_c n)}{\omega_c n} \text{ for } |n| \leq M, \quad \text{(Eqn. 2)}$$

And $h[n] = 0$ otherwise.

Where ω_c is the cutoff frequency. Note, that this frequency is based on both the cutoff frequency in hertz, and our sampling frequency in hertz.

INSTRUCTOR VERIFICATION:
2(a): Determine an appropriate impulse response to use for our problem.
Hint: The parameters we change should be the cutoff frequency and our truncation value M. Otherwise, we should just use the form of Eqn. 2 above.
Note: What happens as n approaches 0? What happens when n = 0? (Recall idea from calculus courses)

Once you have the impulse response, implement the corresponding non-ideal Low Pass Filter in MATLAB. Hint: $Y(\Omega) = H(\Omega)X(\Omega) \leftrightarrow x[n] \star h[n] = y[n]$

Experiment with different values of M, and find an appropriate value such that you no longer see the noise.

In order to plot the resulting data against a time axis, you would need to pad the time vector from the x[n] we loaded in earlier.

Consider the following provided code:

```
>> load('tremor.mat'); % This gives vectors 'data' and 'time' to the workspace
>> x = data;
>> endT = max(time); % This gives us the last value we stored in our time vector
>> padLength = length(y) - length(x); % this gives the additional length we need
>> time_y = [time, (0.0100 * ones(1,padLength)) + endT] % This gives us the extended time
```

This gives us an extended time vector that we can use to plot the filtered data against.

INSTRUCTOR VERIFICATION:
2(b): Plot the resulting data in both time and frequency domain. Demonstrate that you have filtered out the noise caused by the hand tremors. Be able to explain the effects of changing the selected value for M that you observe, as well as why these changes occur.

Part 3: Applying Digital Filters Using CyDAQ
Finally, we'll use the CyDAQ Digital Filtering tool to apply a digital low pass filter using the filter coefficients.

To start, we'll open the 'Digital Filter Gen' tab on the CyDAQ GUI. This is just to the right of the 'Digital Sensor Select' tab from part 1. Press the 'Design FIR Filter' button. This will prompt you to select a file to provide as an input to the filter. You'll need to select the file containing the signal demonstrating the Hand Tremors that you collected in part 1.

Next, we provide the FIR Filter coefficients. Download the provided FilterVec.csv file, and within MATLAB, right click and select 'Open as Text'. Copy and paste these values into the FIR Coefficients field in the window. Then click 'Save Filtered Data' to save the resulting signal as a .mat file. Finally, plot the resulting signal in both time and frequency domain, and observe that it successfully filters out the added noise caused by the hand tremors.

INSTRUCTOR VERIFICATION:
3: Plot the resulting data in both time and frequency domain. Demonstrate that you have filtered out the noise caused by the hand tremors.

EE 224 INSTRUCTOR VERIFICATION SHEET:

NAME: ____________________________
DATE: ____________________________
LAB SECTION: ________, LAB #: _____

PRELAB:
The expected range we would see the hand tremors occur in is between: ___ and ___ (Hz)

Instructor Initials: _________ Date: _____________

PART 1:
Plot in both time and frequency domain.
(a) Regular Hand Motion:

Instructor Initials: _________ Date: _____________
(b) Tremor Hand Motion:

Instructor Initials: _________ Date: _____________

PART 2:
(a) Determine an appropriate impulse response to use for this application:
Value for omega_c: ____________
Value for M: _______

Expression for h[n]:

____

Instructor Initials: _________ Date: _____________

(b): Plot the resulting signal in both time and frequency domain. Be able to explain any results, as well as how changing the value of M changes the resulting signal.

Instructor Initials: _________ Date: _____________

PART 3:
Plot the CyDAQ filtered data in both time and frequency domain.

Instructor Initials: _________ Date: _____________