Biped Patrol - Task 1.1

Signal Processing

Signal processing is a field of engineering that focuses on analysing, modifying and synthesizing signals such as sound, images, biological measurements etc. Signal processing techniques can be used to improve transmission, storage efficiency and subjective quality and to also emphasize or detect components of interest in a measured signal. There various ways to classify these signals, such as: Analog, Digital, Continuous Time, Discrete Time etc.

In signal processing, a **"Filter"** is a device or process that removes some unwanted components or features from a signal. Filtering is a class of signal processing, the defining feature of filters being the complete or partial suppression of some aspect of the signal. Most often, this means removing some **frequencies** or **frequency bands**.

There are many different bases of classifying filters and these overlap in many different ways; there is no simple hierarchical classification. Filters may be: Non-Linear or Linear, Time-variant or Time-invariant, Causal or Not-causal, Analog or Digital, Passive or Active etc…..

Frequency filter circuits (such as low-pass, high-pass, band-pass, and band-reject) shape the frequency content of signals by allowing only certain frequencies to pass through. You can describe these filters based on simple circuits or logics.

Low Pass Filter

The low-pass filter has a gain response with a frequency range from zero frequency (DC) to cutoff frequency (ω_c), which means that any input that has a frequency below the ω_c gets a pass, and anything above it gets attenuated or rejected. The gain approaches zero as frequency increases to infinity. Refer to the image below:

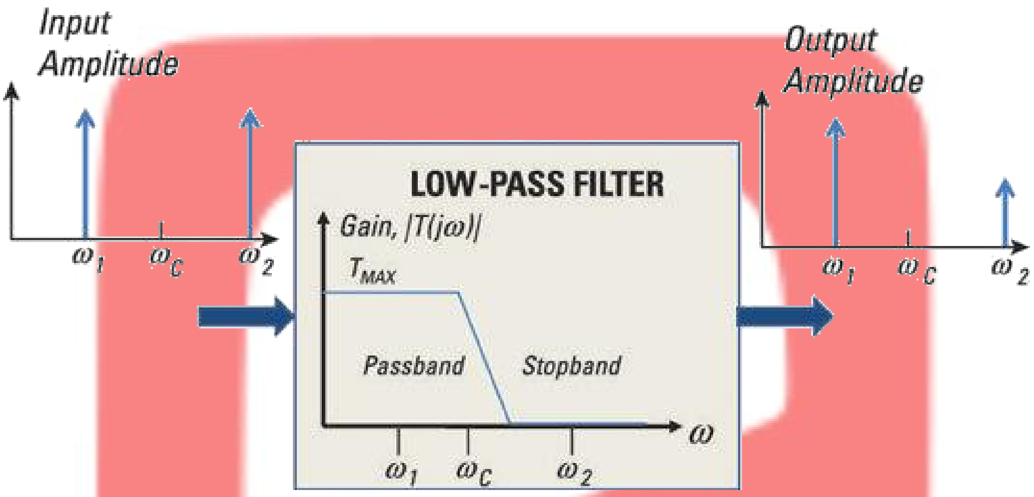

Figure 1

The input signal of the filter shown here has equal amplitudes at frequencies ω_1 and ω_2. After passing through the low-pass filter, the output amplitude at ω_1 is unaffected because it's below the cutoff frequency ω_c. However, at ω_2, the signal amplitude is significantly decreased because it's above ω_c.

High Pass Filter

Opposite to the Low-pass filter, the high-pass filter has a gain response with a frequency range from the cutoff frequency (ω_c) to infinity. Any input having a frequency below ω_c gets attenuated or rejected. Anything above ω_c passes through unaffected. Refer to the image below:

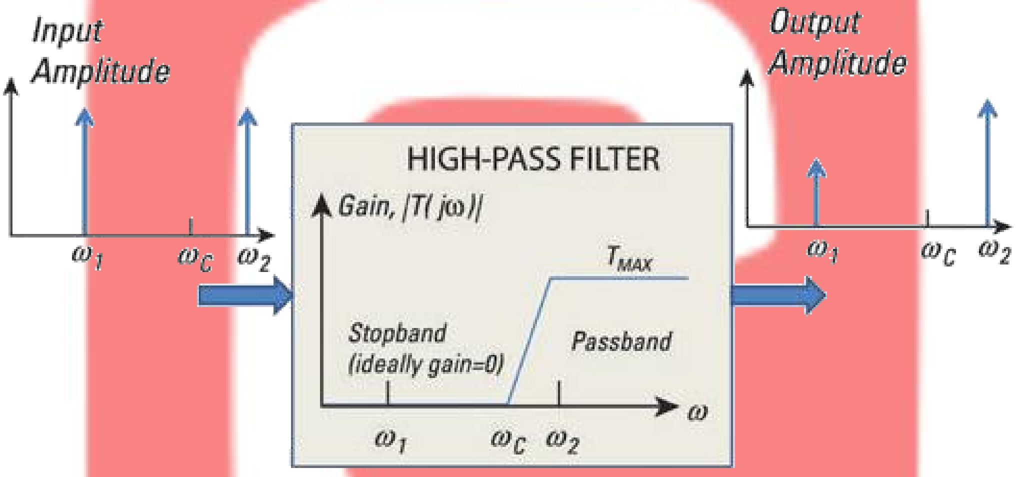

Figure 2

The input signal of the filter shown here has equal amplitude at frequencies ω_1 and ω_2. After passing through the high-pass filter, the output amplitude at ω_1 is significantly decreased because it's below ω_c, and at ω_2, the signal amplitude passes through unaffected because it's above ω_c.

Complimentary Filter

Complementary filter is a simple estimation technique to combine measurements. The basic complementary filter is shown in the figure below where x and y are noisy measurements of signal **z**; **ẑ** is the estimate of **z** produced by the filter. Assume that the noise in y is mostly high frequency, and the noise in x is mostly low frequency. Then G(s) can be made a low pass filter to filter out the high-frequency noise in y. If G(s) is low-pass, [1-G(s)] is the compliment, i.e., a high pass filter which filters out the low-frequency noise in x. No detailed description of the noise process is considered in complementary filtering.

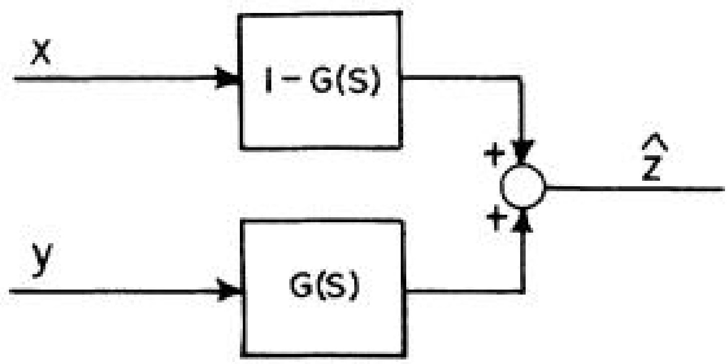

Figure 3

Why do we need to use complementary filter in our task?

After studying the characteristics of both gyro and accelerometer, we know that they have their own strengths and weaknesses. The calculated tilt angle from the accelerometer data has slow response time, while the integrated tilt angle from the gyro data is subjected to drift over a period of time. In other words, we can say that the accelerometer data is useful for long term while the gyro data is useful for short term. Idea behind complementary filter is to take slow moving signals from accelerometer and fast moving signals from a gyroscope and combine them. Complementary filter is designed in such a way that the strength of one sensor will be used to overcome the weakness of the other sensor which is complementary to each other.

Accelerometer gives a good indicator of orientation in static conditions. Gyroscope gives a good indicator of tilt in dynamic conditions. So the idea is to pass the accelerometer signals through a low-pass filter and the gyroscope signals through a high-pass filter and combine them to give the final rate.

The key-point here is that the frequency response of the low-pass and high-pass filters add up to 1 at all frequencies. This means that at any given time the complete signal is subject to either low pass or high pass.

Equation for low-pass filter:

y[n] = (1-alpha).x[n] + alpha.y[n-1] //use this for angles obtained from accelerometers

x[n] is the pitch/roll/yaw that you get from the accelerometer

y[n] is the filtered final pitch/roll/yaw which you must feed into the next phase of your program

Equation for high-pass filter:

y[n] = (1-alpha).y[n-1] + (1-alpha)(x[n]-x[n-1]) //use this for angles obtained from gyroscopes

x[n] is the pitch/roll/yaw that you get from the gyroscope

y[n] is the filtered final pitch/roll/yaw which you must feed into the next phase of your program

n is the current sample indicator.

A quick way of implementing a complementary filter:

angle = (1- alpha)(angle + gyro * dt) + (alpha)(acc)

First reading is the angle as obtained from gyroscope integration. Second reading is the one from accelerometer.

How to choose alpha?

alpha **= (**tau**)/(**tau + dt**)** where tau is the desired time constant (how fast you want the readings to respond) and dt = 1/fs where fs is your sampling frequency.