**Introduction**

A digital filter is an algorithm or a mathematical function which converts digital input to digital output with desired signal characteristics. Different types of filtering that can be performed on the input data include

* Low-pass filtering
* High-pass filtering
* Band-pass filtering
* Band-stop filtering
* Band-stop filtering
* Data smoothing
* Averaging



Figure Low pass filter



Figure High pass filter



Figure Band pass filter



Figure Band stop filter

As this filtering technique does not require hardware components, digital filter design is less complicated when compared to analog filter design. The parameters of the filter can be tweaked and a suitable filter, for a particular application, can be developed iteratively. Digital filters are used for various applications, some of them include,

* Digital anti-alias filtering in data acquisition systems
* Decimation
* Noise reduction
* Fixed and adaptive interpolation
* Order tracking

This is not an exhaustive list of applications as digital filters are used absolutely where signal processing is involved. When compared to analog filters, digital filters have certain advantages as discussed below

* Accuracy – Digital filters can be developed either in Laplace domain or by performing an inverse fft on the desired frequency spectrum. As digital filters are mathematical equations, characteristics of a digital filter can be fully understood
* Repeatability – As there are no hardware components involved, there is no problem of aging and the results acquired by a particular digital filter are highly repeatable
* Implementation – Digital filters can be hard-coded into the chip or can be coded for application after the data is acquired by DAQ
* Flexibility – It’s the ultimate in flexibility. It is possible to design almost any shape using digital filters. Most signal processing toolboxes have a frequency versus gain plot which can be manipulated to get a filter of desired shape
* Cost effective – Digital filter is just a piece of software of code. As this does not require any hardware components for implementation it is the most cost-effective
* No channel to channel variation – All that matters is to check if the filter behaves properly and as desired. If same filtering application is performed on all channels, same piece of code can be used without introducing any variation in the filtering process

Different types of digital filters available are

* Finite impulse response – FIR
* Infinite impulse response – IIR
* Adaptive IIR and Adaptive FIR
* Kalman filtering
* Long FFT filtering

Important parameters of a filter design



Figure Filter design parameters

The filter design schematic in figure 1 describes a low-pass filter

Passband – These are frequencies through which data passes without any change in the frequency. Ideally, this band should not introduce phase delay. However, all the filters introduce some phase delay, be it linear or non-linear

Passband ripple – This is the variance in gain in pass-band

Stopband – These are the frequencies in which the gain is attenuated. The algorithms are not programmed for agreeable phase characteristics in stop-band.

Stopband ripple – This is the variation of gain in a stop-band

Transition band – It can be seen in figure 1 that the transition from start band to stop band occurs over a small band of frequencies. This band is called the transition band. Transition band changes when different filters are used. For example, Butterworth filter has a greater transition band when compared to elliptical filters.

Stopband Attenuation – Stopband attenuation is defined as the attenuation in the gain of the filter in the stopband. It is advisable to have the stopband attenuated to the noise floor. This parameter is also filter dependent and varies drastically when different filters are used

FIR filter

An FIR filter is mathematically defines as

|  |  |  |
| --- | --- | --- |
|  |  |  |

where, y(n) is the filtered data, x(n) is the actual data and denotes the coefficient that each data point is multiplied by. In equation 1, k denotes the number of data points used to calculate the filtered value at a particular data point. Higher the value of k, higher the number of coefficients and higher is the order of the filter. These coefficients are also called as taps.

From equation 1, it can be observed that an FIR filter relies on the data to calculate filtered output. It does not rely on any filtered output values. This makes the filtered output stable. It can also be deduced from equation 1 that and FIR filter has a finite startup transient. Startup transient is defined as