- - - Signal Processing Basics - - -

- **FIR filters subtypes**:
  - **subtype I**: odd length and symmetric impulse response
  - **subtype II**: even length and symmetric impulse response
  - **subtype III**: odd length and antisymmetric impulse response
  - **subtype IV**: even length and antisymmetric impulse response

	subtype I	subtype II	subtype III	subtype IV
low-pass	X	X		
high-pass	X			X
band-pass	X	X	X	X
band-stop	X			

Red Xs: the possibilities used in this course.

- - - Signal Processing Basics - - -

- **Very important remark**: in our previous class, we designed a band-stop filter by adding a low-pass filter and a high-pass filter. The general procedure is correct **but** the impulse responses of those band-stop filters are neither symmetric nor antisymmetric. Consequently, the filters' phase responses are **not** linear.
- Hence, we could try to define a way to get linear-phase band-stop FIR filters. To do so, we need symmetric or antisymmetric impulse responses.
- Assume that our band-stop filter is $q[n] = \{q_0, q_1, q_2, q_3\}$ where $q[n]$ is a set with an **even** number of coefficients. Then, its frequency response is $Q[z] = q_0 + q_1 z^{-1} + q_2 z^{-2} + q_3 z^{-3}$. Now, let us evaluate the filter frequency response just in two particular points: $\omega = 0$ and $\omega = \pi$.
- Recalling that $z = e^{j\omega}$, then $\omega = 0 \rightarrow z = 1$ and $\omega = \pi \rightarrow z = -1$, we can proceed easily.

- - - Signal Processing Basics - - -

- Thus, we have:
  - for $z = 1 \rightarrow Q[z] = q_0 + q_1(1)^{-1} + q_2(1)^{-2} + q_3(1)^{-3} = q_0 + q_1 + q_2 + q_3$. If the coefficients are antisymmetric, i.e., $q_3 = -q_0$ and $q_2 = -q_1$, then $Q[z] = q_0 + q_1 - q_1 - q_0 = 0$. But this is **incorrect** for a band-stop filter!!! Thus, the filter **cannot** be antisymmetric in case it has an even number of coefficients.
  - for $z = -1 \rightarrow Q[z] = q_0 + q_1(-1)^{-1} + q_2(-1)^{-2} + q_3(-1)^{-3} = q_0 - q_1 + q_2 - q_3$. If the coefficients are symmetric, i.e., $q_3 = q_0$ and $q_2 = q_1$, then $Q[z] = q_0 - q_1 + q_1 - q_0 = 0$. But this is **incorrect** for a band-stop filter!!! Thus, the filter **cannot** be symmetric in case it has an even number of coefficients.
- Therefore, to exhibit linear phase responses, band-stop FIR filters **cannot** have an even number of coefficients!!!!
- Let us see what happens if we reformulate our problem, stating that the filter $q[n] = \{q_0, q_1, q_2, q_3, q_4\}$, i.e., it has an **odd** number of coefficients.

- - - Signal Processing Basics - - -

- In such a case, we have:
  - for $z = 1 \rightarrow Q[z] = q_0 + q_1(1)^{-1} + q_2(1)^{-2} + q_3(1)^{-3} + q_4(1)^{-4} = q_0 + q_1 + q_2 + q_3 + q_4$. If the coefficients are antisymmetric, i.e., $q_4 = -q_0$ and $q_3 = -q_1$, then $Q[z] = q_0 + q_1 + q_2 - q_1 - q_0 = q_2$, i.e., only the coefficient at the central position is effectively considered. But, according to the definition of an antisymmetric odd sequence, the central coefficient must be zero... but this usually does **not** happen... Thus, the filter **cannot** be antisymmetric in case it has an odd number of coefficients.
  - for $z = -1 \rightarrow Q[z] = q_0 + q_1(-1)^{-1} + q_2(-1)^{-2} + q_3(-1)^{-3} + q_4(-1)^{-4} = q_0 - q_1 + q_2 - q_3 + q_4$. If the coefficients are symmetric, i.e., $q_4 = q_0$ and $q_3 = q_1$, then $Q[z] = q_0 - q_1 + q_2 - q_1 + q_0 \neq 0$. Thus, the filter **can** be symmetric in case it has an odd number of coefficients.
- Therefore, to exhibit linear phase responses, band-stop FIR filters **must** be symmetric with an odd number of coefficients!!!!

- - - Signal Processing Basics - - -

- **Remark**: the same kind of calculations can be performed to check the entire table on page 42.
- So, understanding that we need an odd number of symmetric coefficients to get linear-phase band-stop FIR filters, what are the design possibilities? There are many, such as:
  - (i) we just add a low-pass filter with a high-pass one, as we studied earlier, however, to do so, we need both filters to have an odd length... But how can we design a high-pass filter with an odd number of coefficients? One simple solution is to complement the frequency response of a low-pass filter, i.e., $g[n] = \frac{sin(\pi(n - \frac{M}{2}))}{\pi(n - \frac{M}{2})} - \frac{sin(\omega_c(n - \frac{M}{2}))}{\pi(n - \frac{M}{2})}$, where the term $\frac{sin(\pi(n - \frac{M}{2}))}{\pi(n - \frac{M}{2})}$ is an all-pass filter.
  - (ii) we just complement the frequency response of a band-pass filter by using an all-pass filter, i.e., $q[n] = \frac{sin(\pi(n - \frac{M}{2}))}{\pi(n - \frac{M}{2})} - p[n]$.
- Both procedures produce a symmetric set of coefficients with any desired length. Let us take a look at an example.

- - - Signal Processing Basics - - -

- Example: design a band-stop filter, with order $M = 4$, to block all frequencies within the interval $1000 \sim 2000$ Hz. The signal to be filtered was sampled at 10000 samples per second.
- **The Bel scale and the decibels**: sometimes, when checking a filter's frequency response or analyzing the spectral content of a signal, based on Fourier Transform, we get a set of numbers containing a few extremely high values and many extremely low values. If we use the regular decimal scale to plot them, the high values will hamper the visualization of the low ones. Thus, it is usual to plot those values by using a log scale known as Bel scale. Particularly useful is the decibel (dB) conversion, defined as follows:
  - for voltage, current and pressure: $\text{db} = 20 \cdot \log(\frac{\text{output}}{\text{input}}) = 20 \cdot \log(\text{gain})$.
  - for power: $\text{db} = 10 \cdot \log(\frac{\text{output}}{\text{input}}) = 10 \cdot \log(\text{gain})$.

- - - Signal Processing Basics - - -

- In the context of this course, we are interest in voltage gains, in case of filters' frequency responses, and in voltage output, in case of spectral content of a signal. Thus, we will use db $= 20 \cdot \log(\text{gain})$.
- Typical dB values in signal processing appear in the table below:

Decimal	$20 \cdot \log(\text{gain})$		Decimal	$20 \cdot \log(\text{gain})$
0.00001	-100 dB		1	0 dB
0.0001	-80 dB		10	20 dB
0.001	-60 dB		100	40 dB
0.01	-40 dB		1000	60 dB
0.1	-20 dB		10000	80 dB
0.71	-3 dB		100000	100 dB

- **Remark**: For a low-pass / band-stop filter to have a 0 dB gain in the pass-band, its coefficients must sum up 1. Thus, if they do not sum up 1, they can be normalized accordingly.

- - - Signal Processing Basics - - -

- **Remark**: there is a **very important** way to realize filtering procedures, which is indeed the way convolution is commonly implemented in computers. It is called **difference equation**. To obtain it, let us take a look at the next example. But first, one brief detail, which corresponds to a relevant property of the Z Transform: if $X[z]$ is the Z Transform of $x[n]$, then $z^{-k}X[z]$ is the Z Transform of $x[n-k]$.
- Example: Assuming that $q[n] = \{0.000, -0.060, 0.739, 0.161, 0.360, -0.180\}$ is a filter, write down the difference equation to filter the input signal $x[n]$ by using $q[n]$.
- **Today's Short Test (ST6)**: Design an FIR filter ($q[n]$), subtype I, of order $M = 4$, to cut-off frequencies within the range 2500 Hz $\sim$ 3500 Hz, allowing for all the others to pass through. Assume that the input signal to be filtered ($x[n]$) was sampled at 10000 samples per second. Normalize the filters' coefficients in such a way that the filter presents a gain of 0 dB in the pass-band. Lastly, write down the difference equation to filter an input signal $x[n]$ by using $q[n]$.

- - - Signal Processing Basics - - -

- **Remark**: it is important to handle the particular case in which a division by zero might occur when computing $\frac{\sin\left(\omega_c(n-\frac{M}{2})\right)}{\pi(n-\frac{M}{2})}$. It happens with $n = \frac{M}{2}$ for odd length filters. To solve this problem, we recall that:
  - $\omega_c = \alpha\pi$, where $0 \leqslant \alpha \leqslant 1$, implying that the equation becomes $\frac{\sin\left(\alpha\pi(n-\frac{M}{2})\right)}{\pi(n-\frac{M}{2})}$.
  - $\sin(x) \approx x$ for small values of $|x|$.

  Thus, for <u>**small values**</u> of the argument, i.e., values approaching zero, $\sin\left(\alpha\pi(n-\frac{M}{2})\right) \approx \left(\alpha\pi(n-\frac{M}{2})\right)$. Consequently, in this particular case, the equation becomes $\frac{\left(\alpha\pi(n-\frac{M}{2})\right)}{\pi(n-\frac{M}{2})} = \alpha \cdot 1 = \alpha = \frac{\text{cut-off frequency}}{\text{maximum frequency}}$. Therefore, this must be taken into account when designing FIR filters with an odd number of coefficients.