IEE1711 Applied signal processing

Julia Berdnikova
julia.berdnikova@taltech.ee

2019 spring

TAL
TECH

Books and Slides

- John G. Proakis, Dimitris K Manolakis, Digital Signal Processing (4th Edition) (Chapter 9.2, 10.1, 10.2)
- Filter Introduction in Moodle from previous course

IEE1710 Signal processing methods and algorithms (Lecture 11)

Filtering

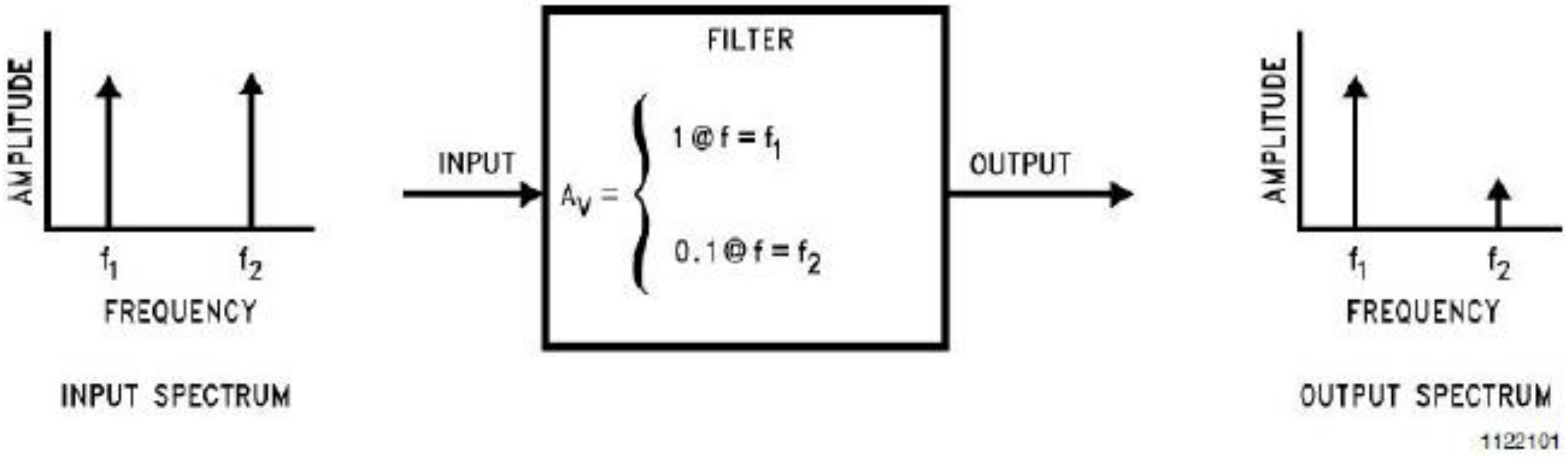


FIGURE 1. Using a Filter to Reduce the Effect of an Undesired Signal at Frequency f_2, while Retaining Desired Signal at Frequency f_1

http://www.ti.com/lit/an/snoa224a/snoa224a.pdf

Filter types

There are many different bases of classifying filters and these overlap in many different ways; there is no simple hierarchical classification. Filters may be:

- linear or non-linear
- time-invariant or time-variant, also known as shift invariance. If the filter operates in a spatial domain then the characterization is space invariance.
- analog or digital
- discrete-time (sampled) or continuous-time
- passive or active type of continuous-time filter
- infinite impulse response (IIR) or finite impulse response (FIR) type of discrete-time or digital filter.

https://en.wikipedia.org/wiki/Filter_(signal_processing)

FIR vs. IIR Filters

- Digital filters with finite-duration impulse response (all-zero, or FIR filters) have both advantages and disadvantages compared to infinite-duration impulse response (IIR) filters.

FIR filters have the following primary **advantages**:

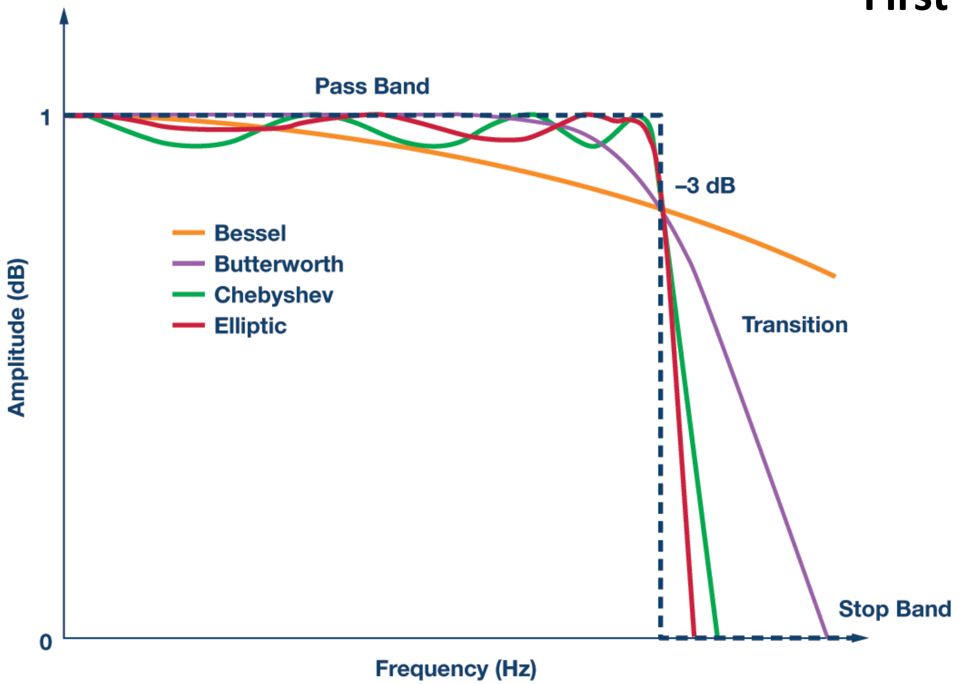
- They can have exactly linear phase. FIR filters can achieve linear phase response and pass a signal without phase distortion.
- They are always stable.
- The design methods are generally linear.
- They can be realized efficiently in hardware.
- They are easier to implement than IIR filters.
- The filter startup transients have finite duration.

The primary **disadvantage** of FIR filters is that they often require a much higher filter order than IIR filters to achieve a given level of performance. Correspondingly, the delay of these filters is often much greater than for an equal performance IIR filter.

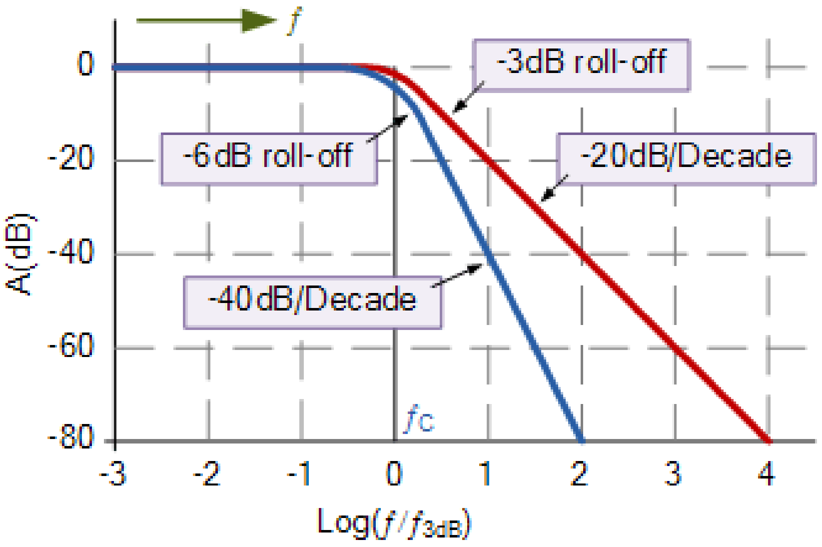
TAL TECH

https://www.mathworks.com/help/signal/ug/fir-filter-design.html
http://zone.ni.com/reference/en-XX/help/370858N-01/genmaths/genmaths/calc_filterfir_iir/

Filter order (digital filter)



First vs second -order roll-off



The maximum delay, in samples, used in creating each output sample is called the ***order* of the filter**. For example,

$$y(n) = x(n) - x(n-1) - 2y(n-1) + y(n-2)$$

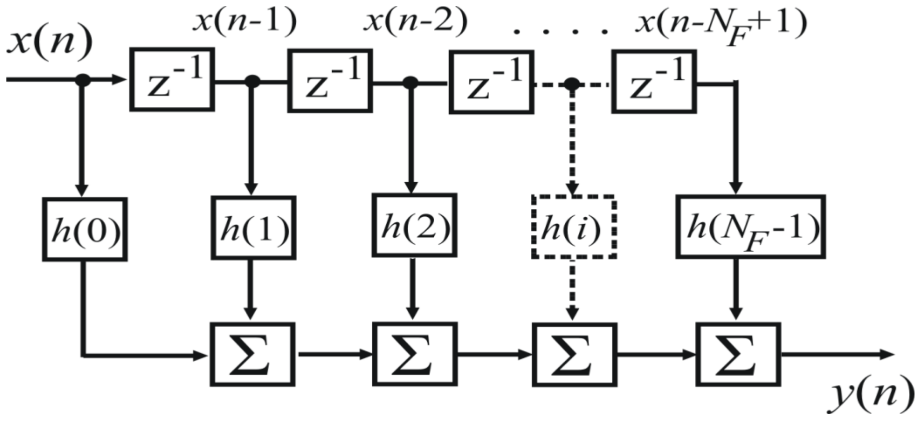
specifies a particular second-order filter.

https://ccrma.stanford.edu/~jos/fp/Filter_Order.html
http://www.analog.com/en/analog-dialogue/articles/practical-filter-design-precision-adcs.html

Finite Impulse Response (FIR) filter

A finite impulse response (FIR) filter is a filter whose impulse response (or response to any finite length input) is of finite duration, because it settles to zero in finite time. This is in contrast to infinite impulse response (IIR) filters, which may have internal feedback and may continue to respond indefinitely (usually decaying).

$$y(n) = \sum_{i=0}^{N_F-1} h(i)x(n-i)$$



An FIR filter of order *N* is characterized by *N+1* coefficients and, in general, require *N+1* multipliers and *N* two-input adders.

https://en.wikipedia.org/wiki/Finite_impulse_response

Ideal Impulse Response

Table 7.1 Summary of Ideal Impulse Responses for Standard FIR Filters

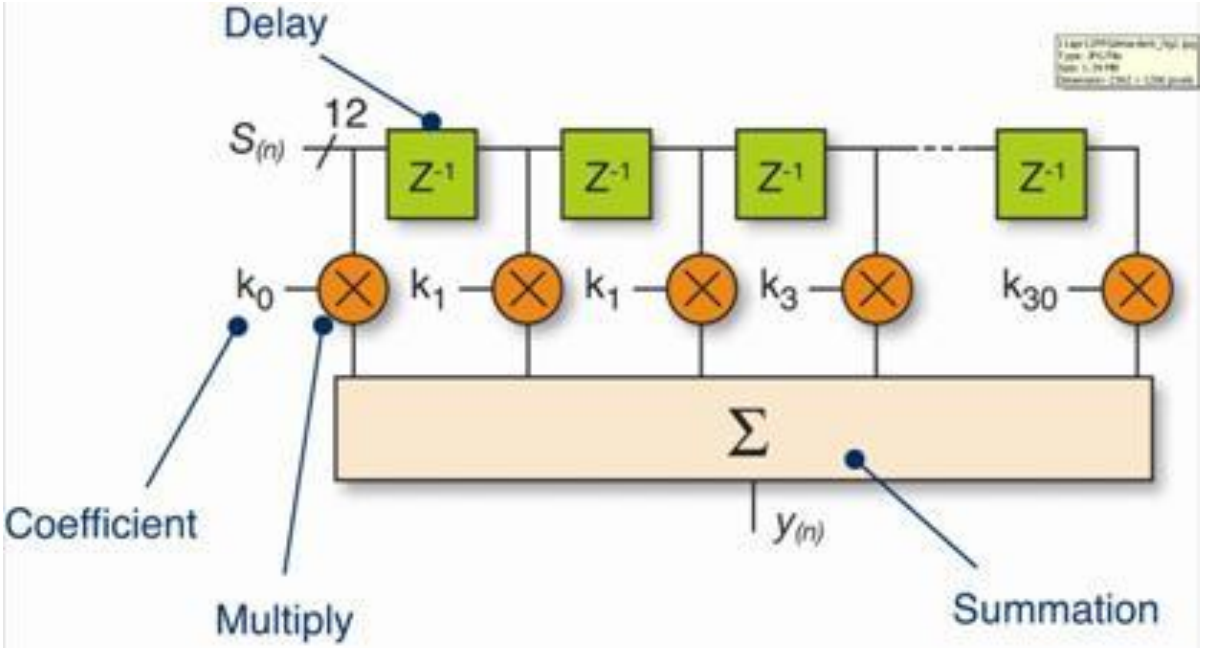
Filter	Type Ideal Impulse Response $h(n)$ (noncausal FIR coefficients)
Lowpass:	$h(n) = \begin{cases} \dfrac{\Omega_c}{\pi} & \text{for} \quad n = 0 \\ \dfrac{\sin(\Omega_c n)}{n\pi} & \text{for} \quad n \neq 0 \end{cases} \quad -M \le n \le M$
Highpass:	$h(n) = \begin{cases} \dfrac{\pi - \Omega_c}{\pi} & \text{for} \quad n = 0 \\ -\dfrac{\sin(\Omega_c n)}{n\pi} & \text{for} \quad n \neq 0 \end{cases} \quad -M \le n \le M$
Bandpass:	$h(n) = \begin{cases} \dfrac{\Omega_H - \Omega_L}{\pi} & \text{for} \quad n = 0 \\ \dfrac{\sin(\Omega_H n)}{n\pi} - \dfrac{\sin(\Omega_L n)}{n\pi} & \text{for} \quad n \neq 0 \end{cases} \quad -M \le n \le M$
Bandstop:	$h(n) = \begin{cases} \dfrac{\pi - \Omega_H + \Omega_L}{\pi} & \text{for} \quad n = 0 \\ -\dfrac{\sin(\Omega_H n)}{n\pi} + \dfrac{\sin(\Omega_L n)}{n\pi} & \text{for} \quad n \neq 0 \end{cases} \quad -M \le n \le M$

Causal FIR filter coefficients: shifting $h(n)$ to the right by M samples.
Transfer function:
$H(z) = b_0 + b_1 z^{-1} + b_2 z^{-2} + \cdots b_{2M} z^{-2M}$
where $b_n = h(n - M), n = 0, 1, \cdots, 2M.$

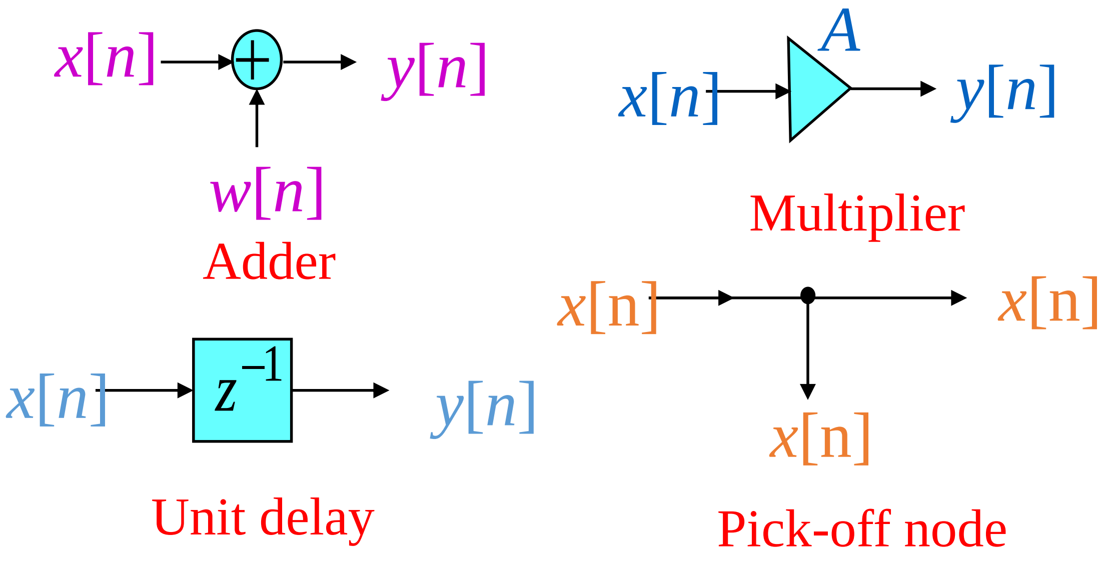
https://dsp.stackexchange.com/questions/13327/fir-filter-design-by-the-fourier-transform-method

TAL TECH

FIR filter block diagramm (N=31)

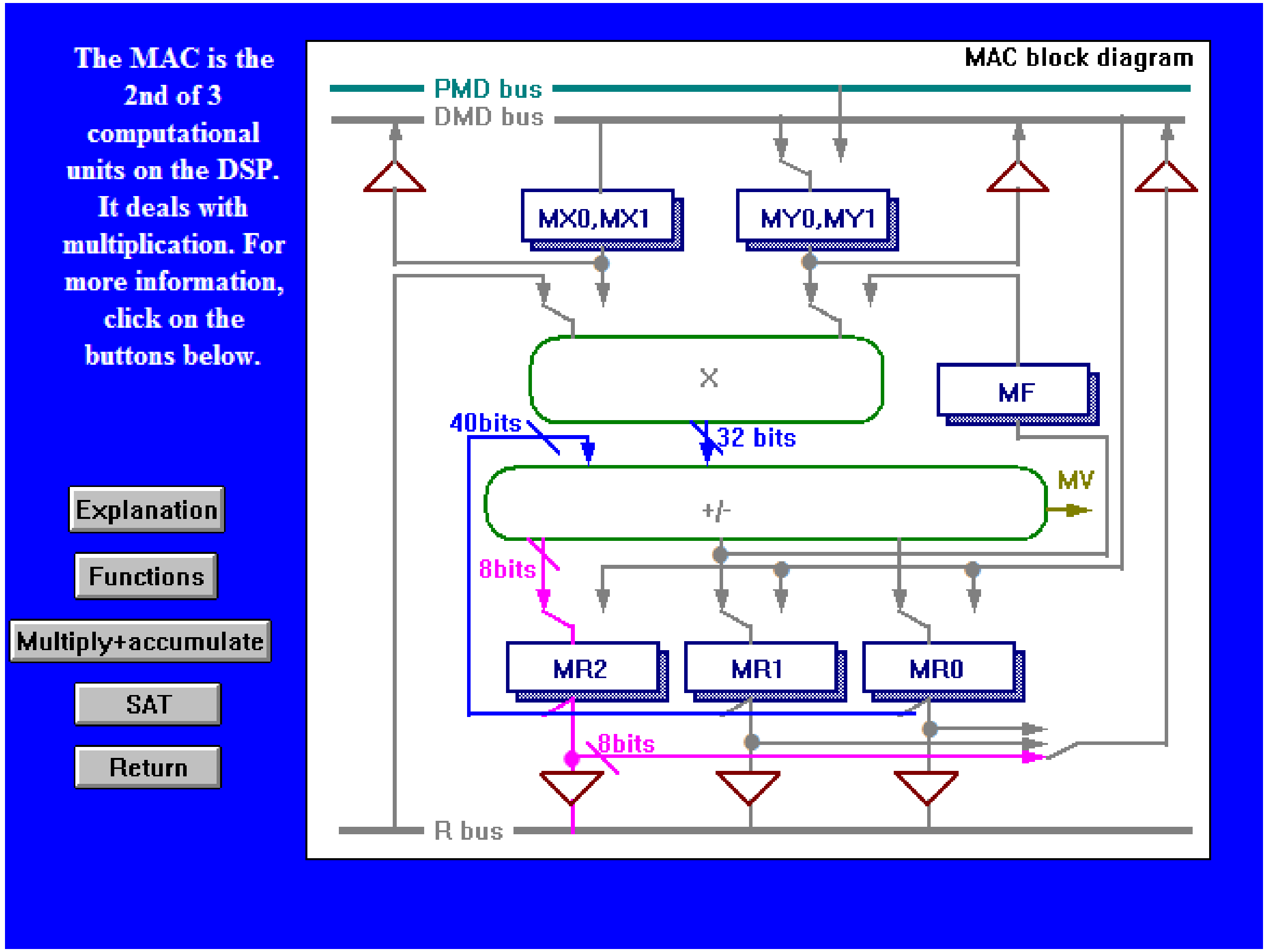


The filter coefficients is sometimes called “taps”



https://static.electronicsweekly.com/news/wp-content/uploads/sites/16/2012/05/itemid-55066-getasset.jpg
eeweb.poly.edu/iselesni/EL713/zoom/Ch6(1).ppt

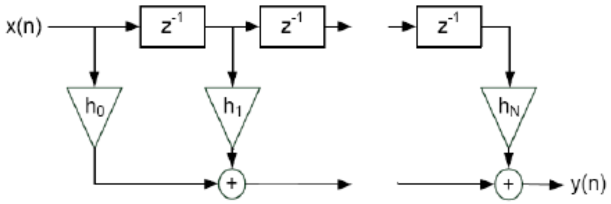
DSP(Digital Signal Processor) implementation of FIR filter (Analog Devises ADSP21xx DSP 16bits Fixed Point)

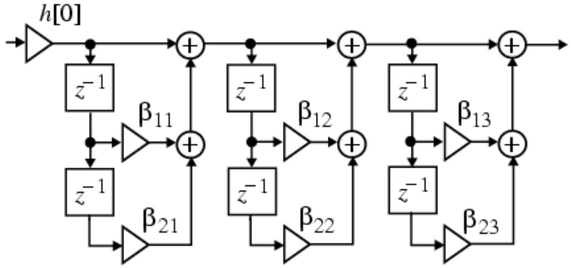


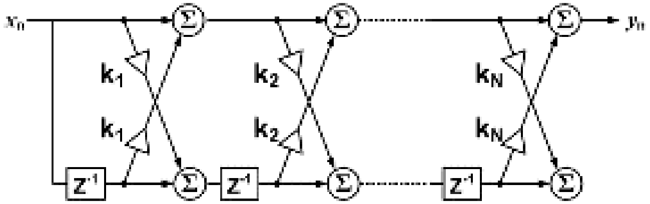
TAL TECH

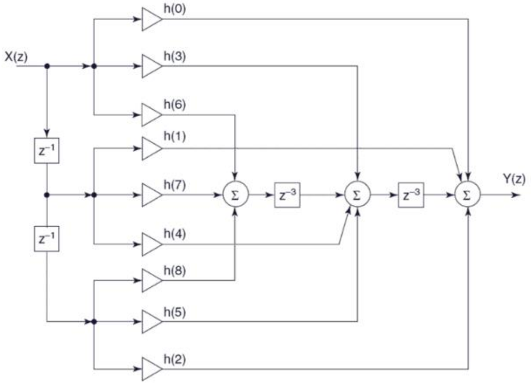
FIR filter structures

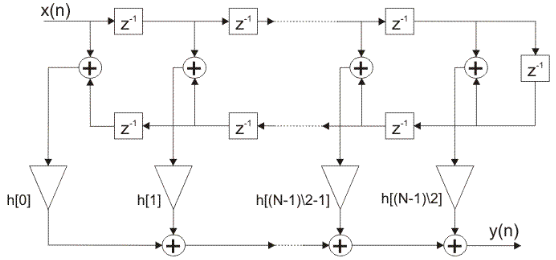
- Direct Form FIR
- Cascade Form FIR
- Lattice Form FIR
- Polyphase FIR
- Linear-Phase FIR











eeweb.poly.edu/iselesni/EL713/zoom/Ch6(1).ppt
https://commons.wikimedia.org/wiki/File:FIR_Lattice_Filter.png
https://commons.wikimedia.org/wiki/File:Fir_filter_polyphase.png
https://www.researchgate.net/profile/Marina_Topa/publication/224111908/figure/fig1/AS:302801168158721@1449204832464/Direct-form-FIR-structure.png
https://e2e.ti.com/cfs-file/__key/communityserver-discussions-components-files/64/7120.FIR_5F00_Structure_5F00_EvenSymetric.gif

Filter design methods:

- Window method
- Frequency sampling method
- Optimal method

Matlab FIR filter summary

Filter Design Method	Description	Filter Functions
Windowing	Apply window to truncated inverse Fourier transform of specified "brick wall" filter	fir1, fir2, kaiserord
Multiband with Transition Bands	Equiripple or least squares approach over sub-bands of the frequency range	firls, firpm, firpmord
Constrained Least Squares	Minimize squared integral error over entire frequency range subject to maximum error constraints	fircls, fircls1
Arbitrary Response	Arbitrary responses, including nonlinear phase and complex filters	cfirpm
Raised Cosine	Lowpass response with smooth, sinusoidal transition	rcosdesign

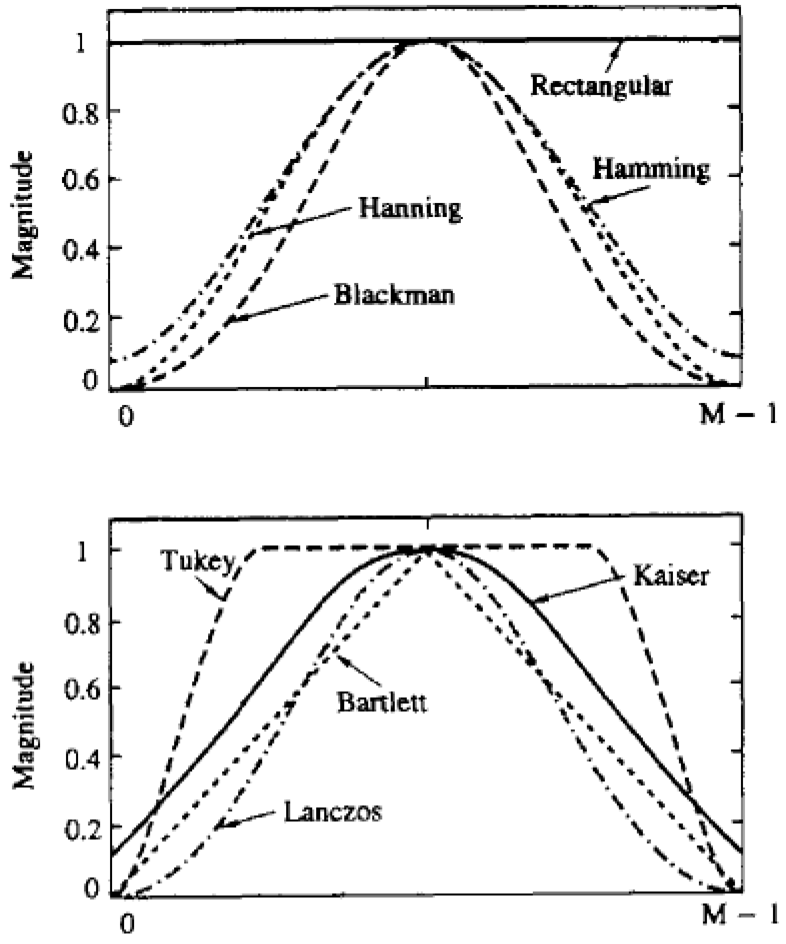
https://www.mathworks.com/help/signal/ug/fir-filter-design.html

Designing a FIR filter (Windowing method)

- Determine the filter specification
- Select a suitable window function and obtain the window function values
- Specify an ideal frequency response (amplitude and phase responses)
- Compute the ideal filter coefficient values (impulse response) from frequency response
- Multiply the window function values by ideal filter coefficients
- Evaluate the frequency response of the filter
- Implementation

Window functions for FIR filter design

Name of window	Time-domain sequence, $h(n),\ 0 \le n \le M-1$
Bartlett (triangular)	$1 - \dfrac{2\left\lvert n - \frac{M-1}{2}\right\rvert}{M-1}$
Blackman	$0.42 - 0.5\cos\dfrac{2\pi n}{M-1} + 0.08\cos\dfrac{4\pi n}{M-1}$
Hamming	$0.54 - 0.46\cos\dfrac{2\pi n}{M-1}$
Hanning	$\dfrac{1}{2}\left(1 - \cos\dfrac{2\pi n}{M-1}\right)$
Kaiser	$\dfrac{I_0\left[\alpha\sqrt{\left(\frac{M-1}{2}\right)^2 - \left(n - \frac{M-1}{2}\right)^2}\right]}{I_0\left[\alpha\left(\frac{M-1}{2}\right)\right]}$
Lanczos	$\left\{\dfrac{\sin\left[2\pi\left(n - \frac{M-1}{2}\right)\Big/(M-1)\right]}{2\pi\left(n - \frac{M-1}{2}\right)\Big/\left(\frac{M-1}{2}\right)}\right\}^L \quad L > 0$
Tukey	$1,\ \left\lvert n - \frac{M-1}{2}\right\rvert \le \alpha\frac{M-1}{2} \quad 0 < \alpha < 1$ $\dfrac{1}{2}\left[1 + \cos\left(\dfrac{n - (1+\alpha)(M-1)/2}{(1-\alpha)(M-1)/2}\pi\right)\right]$ $\alpha(M-1)/2 \le \left\lvert n - \frac{M-1}{2}\right\rvert \le \frac{M-1}{2}$



John G. Proakis, Dimitris K Manolakis, Digital Signal Processing

Windowing method: Matlab FIR design example:

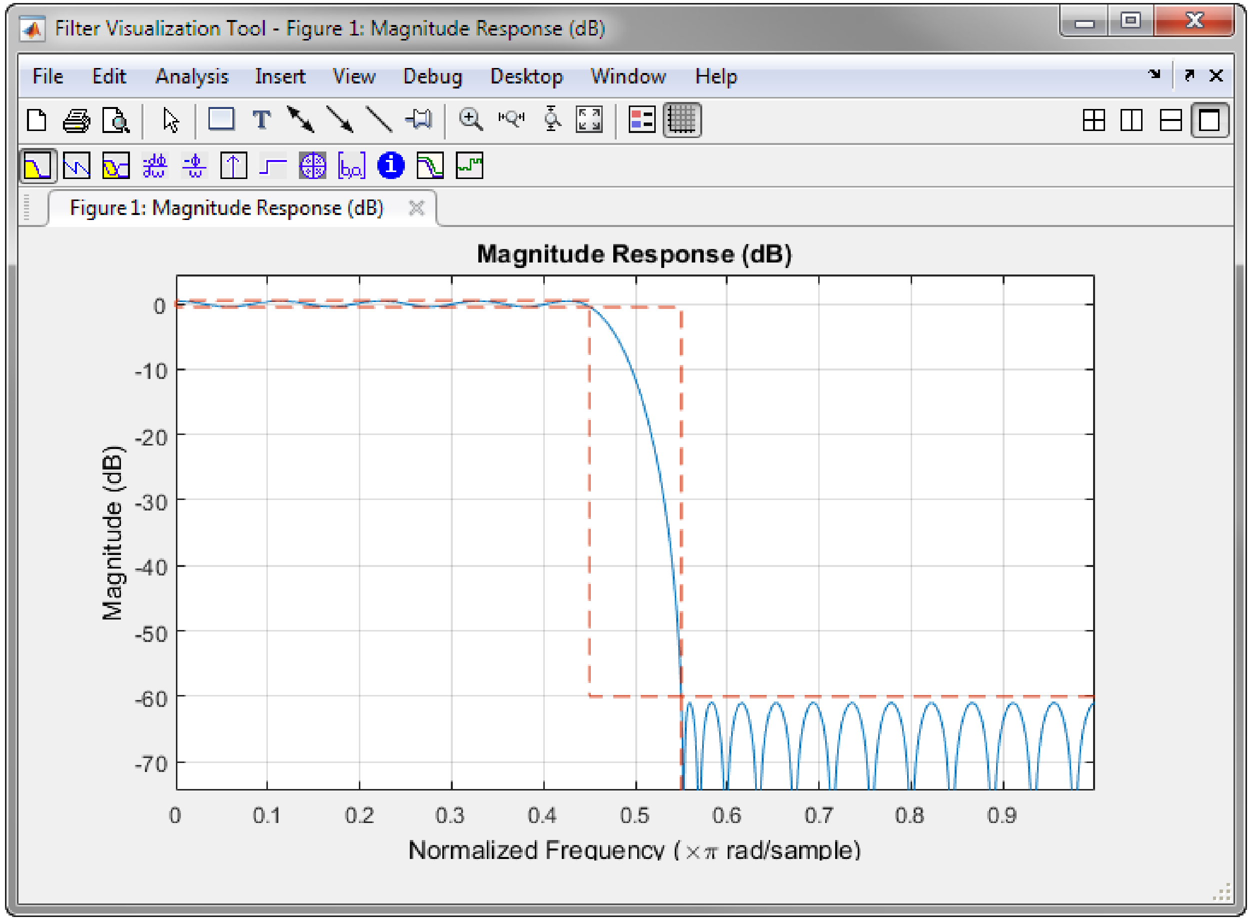
1.) Window-based FIR filter design (returns filter coefficients, as a row vector of length n + 1)

```
bhi = fir1(34,0.48,'high',chebwin(35,30));
freqz(bhi,1)
outhi = filter(bhi,1,x)
```

2.) Frequency sampling-based FIR filter design (returns an *n*th-order FIR filter with frequency-magnitude characteristics specified in the vectors *f* and *m*.)

```
f = [0 0.48 0.48 1];
mhi = [0 0 1 1];
bhi = fir2(34,f,mhi);
freqz(bhi,1)
```

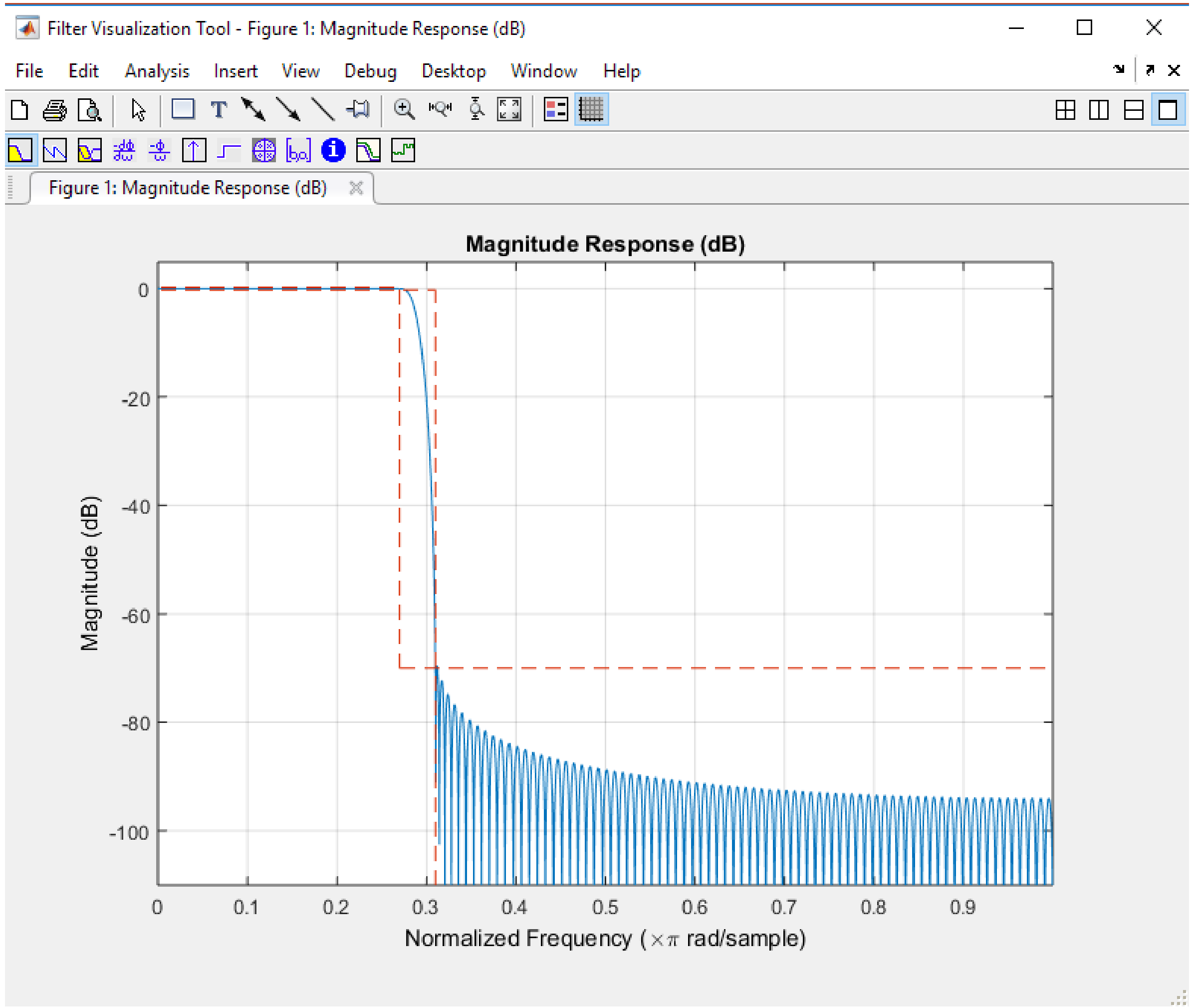
Matlab: Filter Visualization Tool (fvtool)



https://www.mathworks.com/help/signal/ref/fvtool.html

Matlab: Filter Designer example

```
myFilt = designfilt('lowpassfir','PassbandFrequency',0.27, ...
      'StopbandFrequency',0.31,'PassbandRipple',0.5, ...
      'StopbandAttenuation',70,'DesignMethod','kaiserwin');
fvtool(myFilt)
```



Design of FIR filter by the Frequency Sampling Method

- It consists simply of uniformly *sampling* the desired frequency response, and performing an inverse DFT to obtain the corresponding (finite) impulse response.

- When the desired frequency-response is *undersampled*, which is typical, the resulting impulse response will be *time aliased* to some extent. It is important to evaluate the final impulse response via a simulated DTFT (FFT with lots of zero padding), comparing to the originally desired frequency response.

TAL TECH

https://www.dsprelated.com/freebooks/sasp/Frequency_Sampling_Method_FIR.html

Design of Optimum Equiripple FIR filters

The goal of the algorithm is to minimize the error in the pass and stop bands by utilizing the Chebyshev approximation

Read

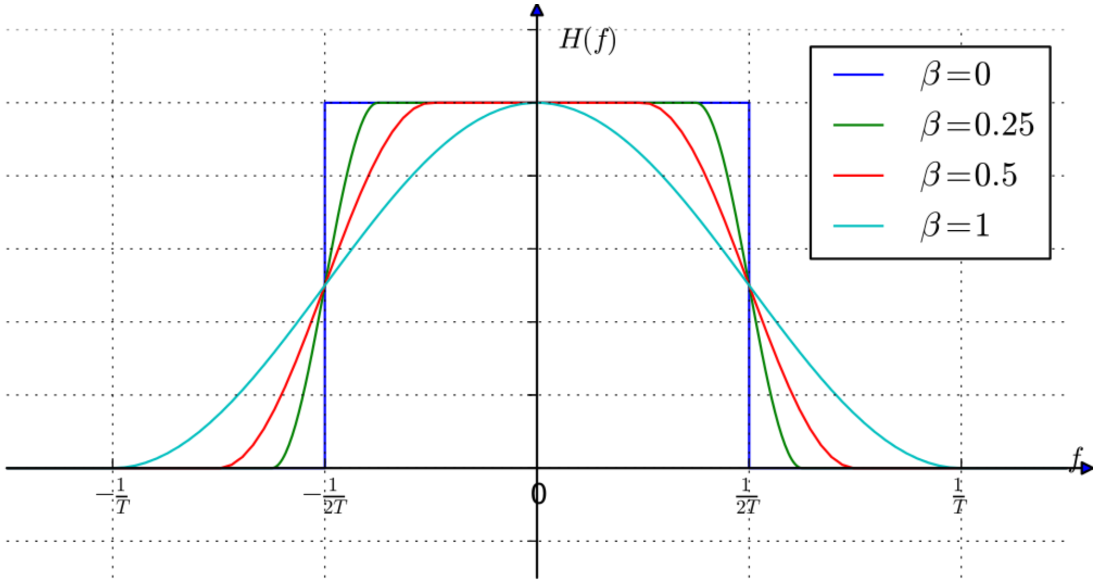
1) Parks–McClellan filter design algorithm

https://en.wikipedia.org/wiki/Parks%E2%80%93McClellan_filter_design_algorithm

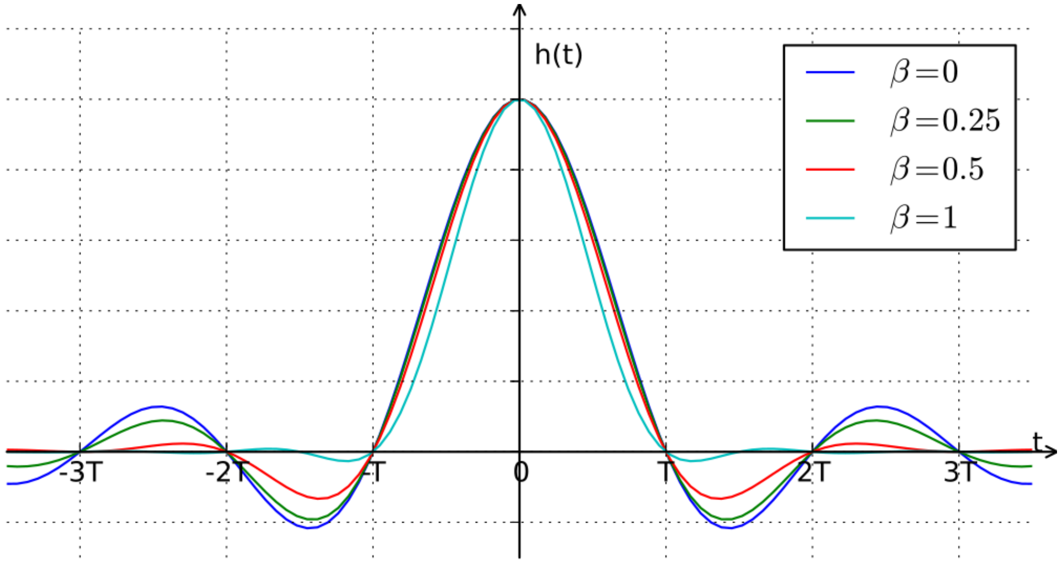
2) Matlab Parks-McClellan optimal FIR filter design (***firpm()***)

https://www.mathworks.com/help/signal/ref/firpm.html

Raised-cosine filter



Frequency response of raised-cosine filter with various roll-off factors



Impulse response of raised-cosine filter with various roll-off factors

https://en.wikipedia.org/wiki/Raised-cosine_filter

Raised-cosine filter

The raised-cosine filter is a filter frequently used for pulse-shaping in digital modulation due to its ability to minimise intersymbol interference (ISI). The roll-off factor, β is a measure of the excess bandwidth of the filter,

Impulse responce:

$$h(t) = \begin{cases} \frac{\pi}{4T}\,\mathrm{sinc}\left(\frac{1}{2\beta}\right), & t = \pm\frac{T}{2\beta} \\ \frac{1}{T}\,\mathrm{sinc}\left(\frac{t}{T}\right)\frac{\cos\left(\frac{\pi\beta t}{T}\right)}{1-\left(\frac{2\beta t}{T}\right)^2}, & \text{otherwise} \end{cases}$$

Frequency responce:

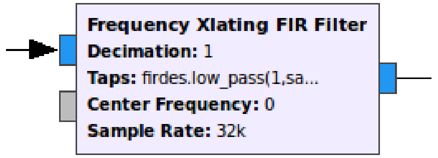
$$H(f) = \begin{cases} 1, & |f| \leq \frac{1-\beta}{2T} \\ \frac{1}{2}\left[1+\cos\left(\frac{\pi T}{\beta}\left[|f| - \frac{1-\beta}{2T}\right]\right)\right], & \frac{1-\beta}{2T} < |f| \leq \frac{1+\beta}{2T} \\ 0, & \text{otherwise} \end{cases}$$

https://en.wikipedia.org/wiki/Raised-cosine_filter

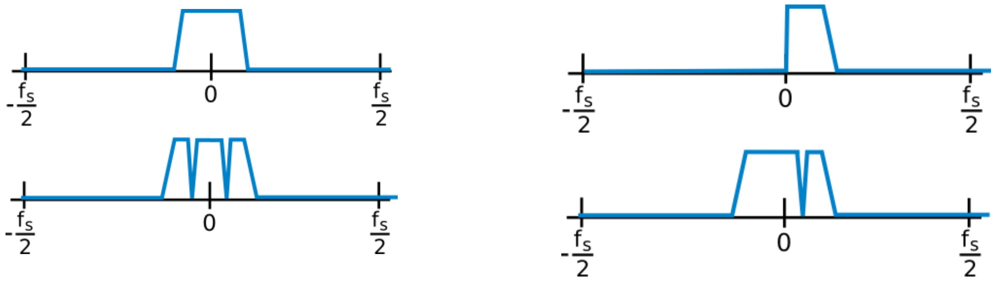
GnuRadio Filters

- Low Pass Filter
- High Pass Filter
- Band Pass Filter
- Band Reject Filter
- DC Blocker
- Hilbert
- Rational Resampler
- Frequency Xlating FIR Filter

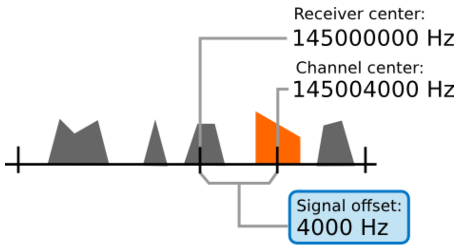
GNURadio filter: Frequency Xlating FIR Filter



- *Real taps, Complex taps*



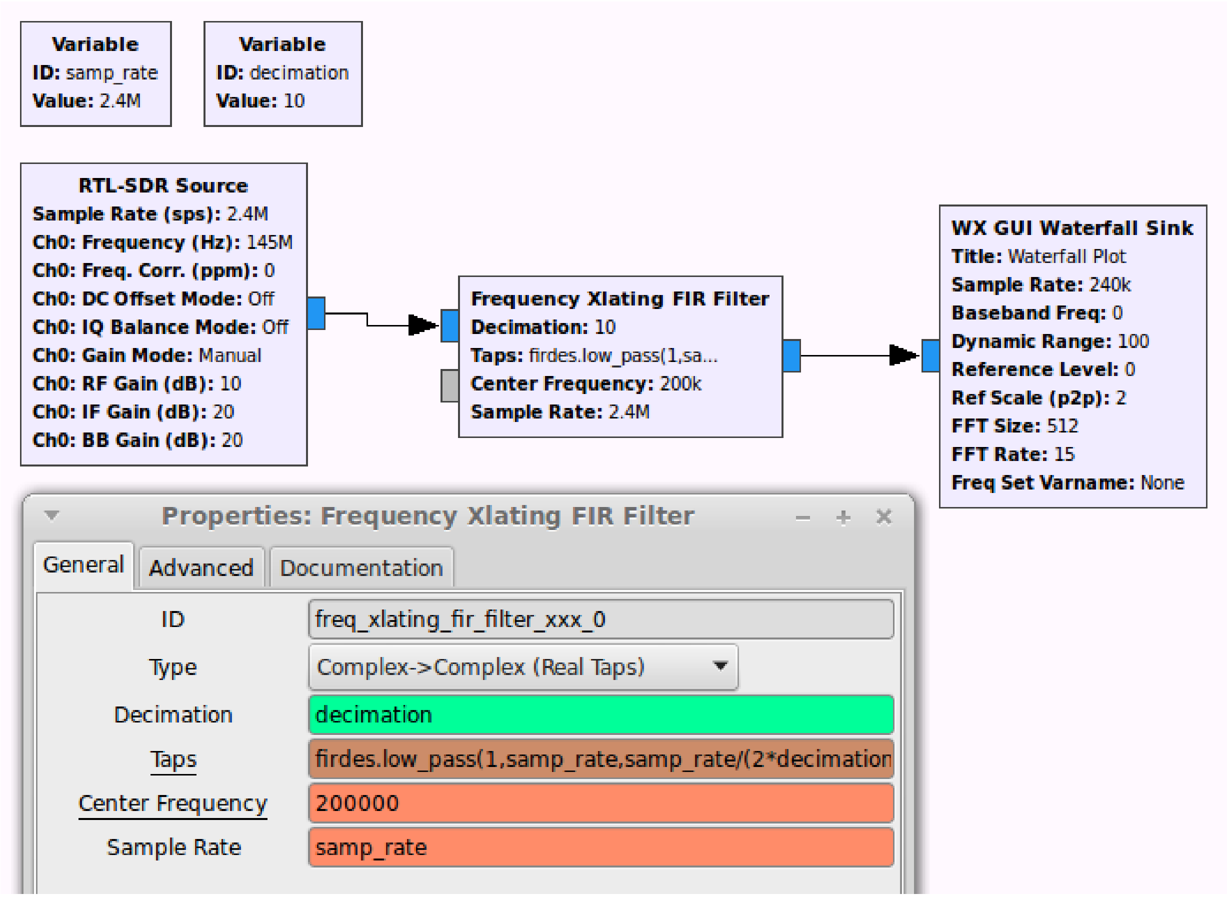
- **Decimation:** the integer ratio between the input and the output signal's sampling rate. *(Resampling)*
- **Center frequency:** the frequency translation offset frequency.



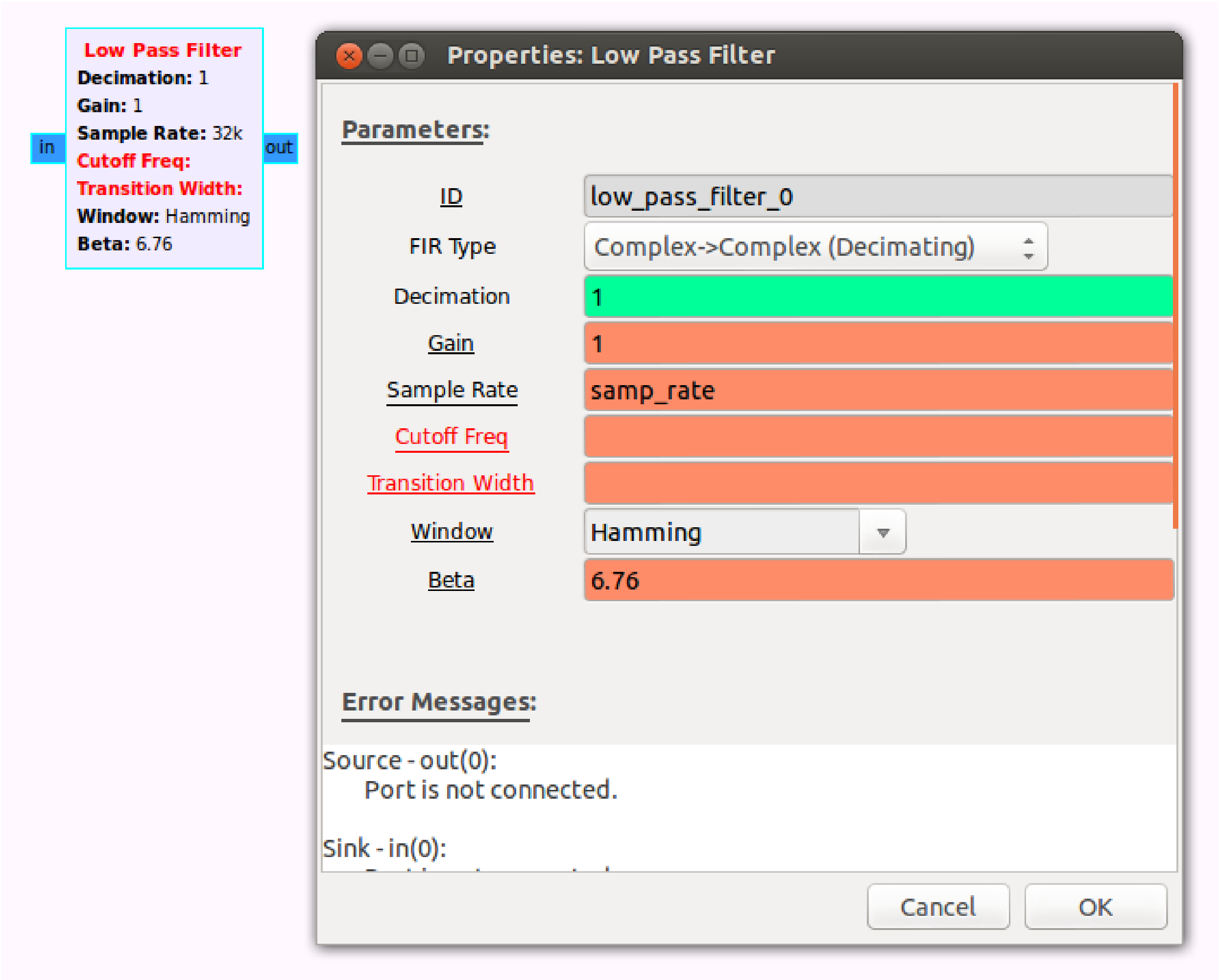
- **Taps**: the vector of the complex or real taps of the FIR filter (can be generated with Matlab or python code)
  - *firdes.low_pass(1,samp_rate,samp_rate/(2*decimation), transition_bw)* *// Real taps*
  - *firdes.complex_band_pass(1, samp_rate, -samp_rate/(2*decimation), samp_rate/(2*decimation), transition_bw)* *// Complex taps*
- **transition_bw** is the transition bandwidth of the filter in Hz. **(This parameter will determine the CPU usage and thus the execution speed of the block.)**
- **Sample rate:** The sample rate of the input signal.

http://blog.sdr.hu/grblocks/xlating-fir.html

GNURadio example :



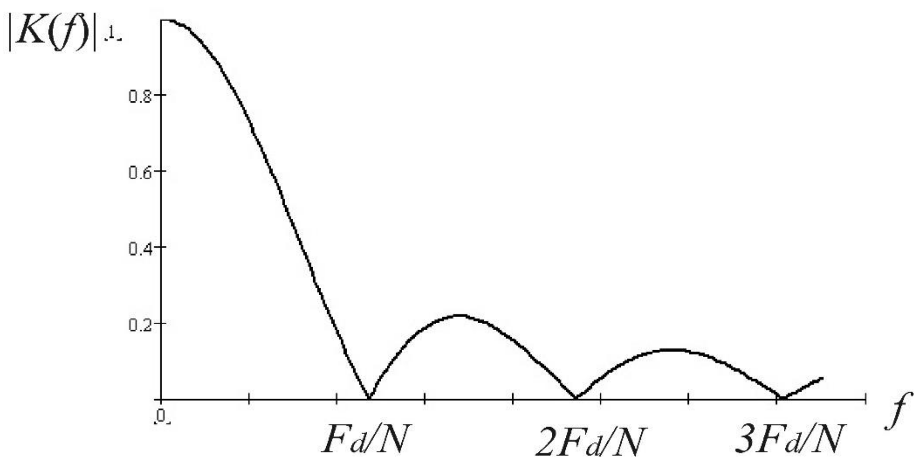
GNURadio example: Low pass filter



https://www.ece.uvic.ca/~ece350/grc_doc/ar01s12s01.html

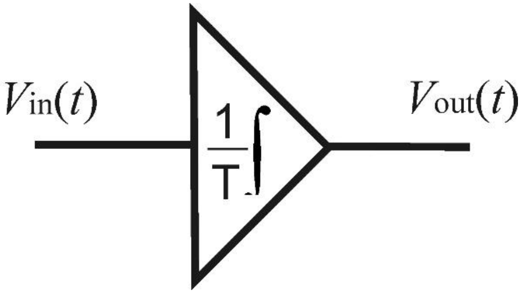
TAL TECH

Example: *First-order low-pass FIR filter* Sliding window



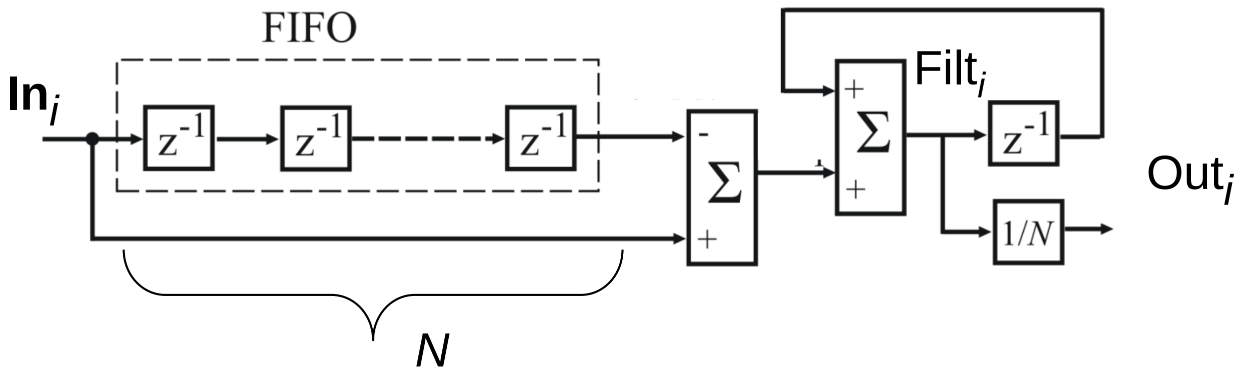
$$|K(f)| = \frac{\sin(\pi \cdot T \cdot f)}{\pi \cdot T \cdot f}$$

Analog filter



$$V_{out}(t) = \frac{1}{T}\int_{t-T}^{t} V_{in}(t)dt$$

Digital Filter



$$Out_i = \frac{1}{N}\sum_{k=i-N}^{i} In_k$$

$$T_A(f) = \frac{\sin(\pi \cdot N \cdot f / F_d)}{N \cdot \sin(\pi \cdot f / F_d)}$$