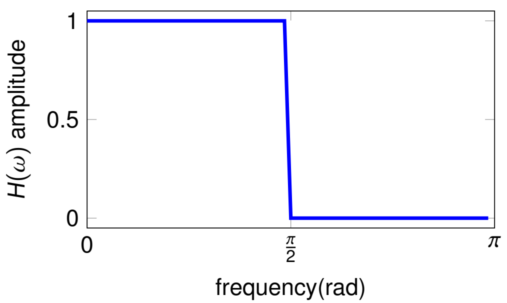
- - - Signal Processing Basics - - -

- Additionally, we usually convert $Y[\omega]$ to its time-domain representation by using the Inverse Fourier Transform, getting a filtered time-domain signal, i.e., $y[n] = \text{InverseFourier}(Y[\omega])$, which can be played, analysed, and so on...
- Notably, as we will verify later, $\text{Fourier}(x[n] * h[n]) = X[\omega] \cdot H[\omega]$.
- Therefore, **multiplication in the frequency-domain is equivalent to convolution in the time-domain**. For the frequency-domain, the desired filter frequency response is "visually" obvious as a set of 0s and 1s, however, to use this approach, two Fourier transformations (one direct and one inverse) are required. Contrary to this, the time domain procedure requires just a convolution but the time-domain filter $h[n] = \text{Inverse Fourier} Y[\omega]$ should be determined, being not "visually" obvious...

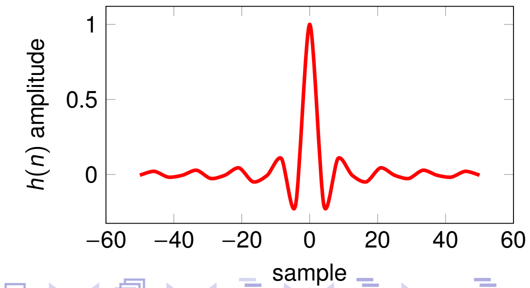
- - - Signal Processing Basics - - -

- Consequently, to be able to work in time-domain only, performing just a convolution to filter a signal, we have to specify the coefficients of $h[n]$. Since $h[n]$ is just the InverseFourier($H[\omega]$), we would need an inverse Fourier transformation... but we don't! This is because, for the well-known filter functions, i.e., low-pass, high-pass, band-pass and band-stop, the Fourier Transform pairs are well-defined and follow pre-defined tabulated functions.
- Let us start with the simplest case, i.e., that of the low-pass filter: the inverse Fourier Transform of a low-pass step function is the *sinc* function, where $\text{sinc}(t) = \frac{sin(t)}{t}$, as shown below.

step low-pass function (frequency domain)



Inverse Fourier Transform of the step low-pass function (time domain)



- - - Signal Processing Basics - - -

- Particularly, by adjusting the function parameters according to the cutoff frequency, we get the following low-pass FIR filter:

$$h[n] = \frac{\sin\left(\omega_c(n-\frac{M}{2})\right)}{\pi(n-\frac{M}{2})}, \text{ where:}$$

  - M is the FIR filter order, which corresponds to an FIR filter with $M+1$ coefficients indexed from 0 to M, i.e., $(0 \leqslant n \leqslant M)$.
  - ω_c is the cut-off frequency $(0 \leqslant \omega_c \leqslant \pi)$, being π the highest possible frequency in the input digital signal to be filtered.
- Example 1: Design an FIR filter of order $M = 1$ to allow for frequencies up to 10 KHz to pass through. The input signal to be filtered was sampled at 40000 samples per second.
- Example 2: Design an FIR filter of order $M = 3$ to allow for frequencies up to 1 KHz to pass through. The input signal to be filtered was sampled at 10000 samples per second.

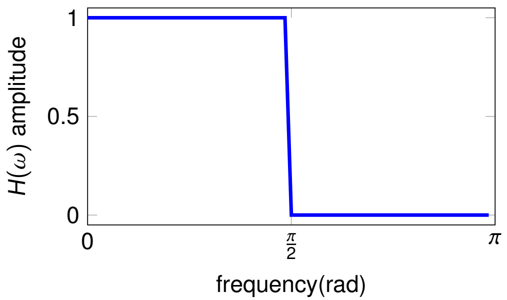
- - - Signal Processing Basics - - -

- The procedure used to find $h[n]$ produces a **symmetric** array!
- Since $Y[\omega] = X[\omega] \cdot H[\omega]$, then, $\frac{Y[\omega]}{X[\omega]} = H[\omega]$ is known as **system transfer function**. The filter **frequency response** corresponds to the transfer function ($H[\omega]$) assessed for different frequencies.
- The set of filter coefficients is known as filter **impulse response**. It represents the response of the filter when it receives an impulse as input, where impulse is the signal $\delta = \{1, 0, 0, 0, ..., 0\}$.
- The filter order affects the filter behaviour in such a way that, the higher the order, the closer the frequency response is to the ideal response. Usually, a step-like function matches the ideal response.
- **Today's Short Test (ST3)**: Design an FIR filter ($h[n]$) of order $M = 5$ to allow for frequencies up to 1500 Hz to pass through, assuming that the input signal to be filtered ($x[n]$) was sampled at 12000 samples per second. Considering, as an example, that $x[n] = \{-1, 1, -2, 2, 2\}$, filter it by using $h[n]$, obtaining the output signal.

- - - Signal Processing Basics - - -

- **High-pass FIR filters**: Different ways to define a high-pass filter exist. The easier consists of a simple trick: just consider that a high-pass frequency response is the mirrored low-pass response, in relation to the vertical axis, as shown below. Assuming that the equivalent time-domain operation is just the time-reversed digital filter with a subsequent alternating-signs procedure, we can simply get the desired high-pass filter by calculating a convenient low-pass filter and applying the trick.

low-pass filter frequency response



high-pass filter frequency response