

21/11/2016

**Ex.1 (Pt.11)**

A signal is sampled at 80kHz. We want to transform the sampled digital signal into a signal sampled at 120kHz.

- 1) Define the whole pipeline to get this result providing accurate description of the filters adopted both for sampling and re-sampling.

We want to re-sample the signal in real-time working on blocks of 1024 samples and in the frequency domain.

- 2) Describe how the DFT can be obtained, provide how the transform matrix can be realized for this specific case and the operations to get the DFT of the input signal
- 3) Define a proper filter in the frequency domain for the specific task of question 1 and how it has to be applied to the signal.
- 4) Describe how the filtered signal can be transformed back into the time domain. Are there issues related to circular convolution for periodicity assumption for this specific case?

**Ex.2 (Pt.11)**

The following signal:

$$x(t) = 5 + 3 \cos(2\pi f_1 t) + \cos(2\pi f_2 t), \quad (1)$$

where  $f_1 = 10\text{kHz}$  and  $f_2 = 40\text{kHz}$  is sampled at 80kHz and is filtered with the following filter:

$$H(z) = \frac{1 - \sqrt{2}z^{-1} + z^{-2}}{1 - 0.7\sqrt{2}z^{-1} + 0.49z^{-2}}$$

Plot the filter in the z-plane and provide the output signal in the time domain.

**Ex.3 (Pt. 11 - MATLAB code)**

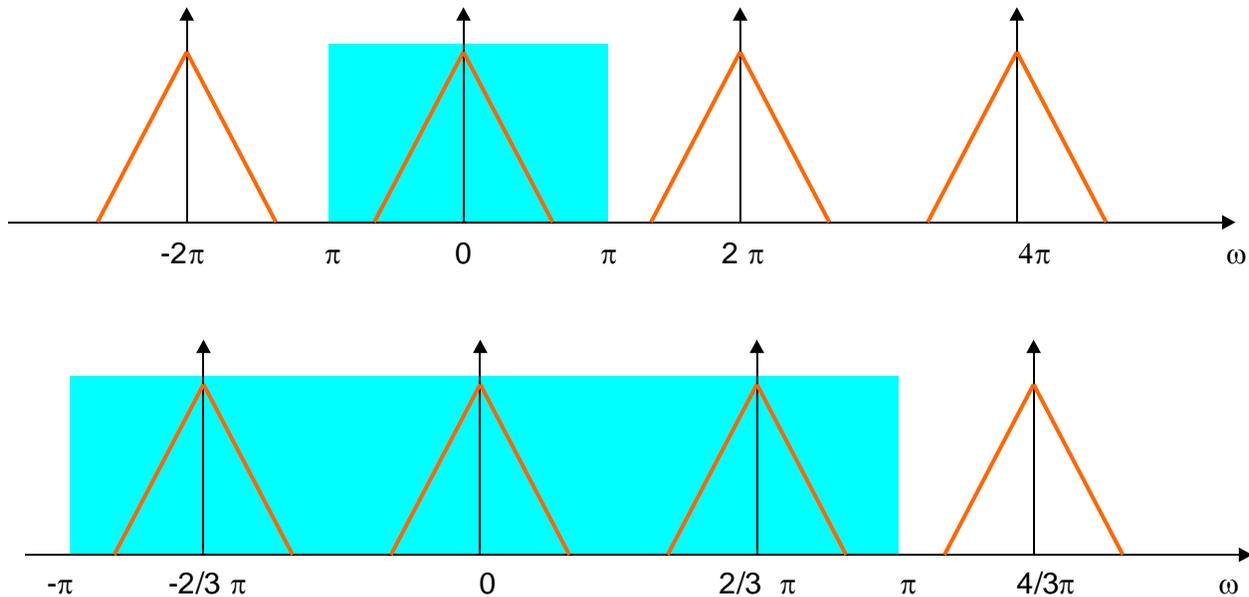
- 1) Generate 1024 from the signal (1) of exercise 2.
- 2) Apply all the steps in order to change the sample rate as described in Exercise 1. NOTE: the DFT must be implemented from scratch without using the fft function.
- 3) Plot the first 512 samples of the output signal

## Solutions

### Ex.1

The signal has to be upsampled of an order of 3 and downsampled of an order of 2. This will allow to move from 80kHz -> 240kHz->120kHz.

During upsampling the spectrum will be compressed of an order or 3 and a replica will appear below the Nyquist frequency as depicted in the following figure:



in cyan it is represented the region below the Nyquist frequencies before and after the upsampling.

The low pass filter for upsampling must then cut frequencies above  $\frac{\pi}{3}$  to remove replicas below the Nyquist

frequencies. For the downsampling step the low-pass filter should cut frequencies above  $\frac{\pi}{2}$  to prevent

aliasing, so, the low pass filter for the upsampling (which is already removing frequencies above  $\frac{\pi}{3}$ ) is already removing aliasing risk also for the downsampling stage.

In order to apply the upsampling and then the filter in the frequency domain I need to take 1024 samples from the signal, add two samples with zero value every sample and then define the DFT Matrix:

The DFT kernel is  $w = e^{j\frac{2\pi}{3 \cdot 1024}} = e^{j\frac{\pi}{1536}}$  so, the W matrix for the DFT will be a 3072x3072 matrix where

$$\mathbf{W}_{r,c} = w^{-r \cdot c}$$

Once I multiply this matrix with the set of upsampled samples I get the DFT of the upsampled signal, in order

to apply the low pass filter in the frequency domain, I have to remove all the frequencies between  $\frac{\pi}{3}$  and  $\frac{5}{3}\pi$

(since now I have to consider that my spectrum is represented in the frequency range between 0 and  $2\pi$ ).

This can be done simply setting to zero all frequencies between the 513<sup>th</sup> sample ( $3072/6+1$ ) and the 2560<sup>th</sup> sample ( $3072 \cdot 5/6$ ) of the DFT of my signal which is exactly a low-pass filter rectangular in the frequency domain.

In order to go back into the time domain we have to pre-multiply the 3072 samples in the frequency domain

with the matrix:  $\mathbf{W}_{r,c}^* = \frac{1}{3072} w^{r \cdot c}$

Analyzing what we are performing in the time domain, since we are multiplying the spectrum of upsampled signal with a low pass rectangle filter we are implicitly performing a circular convolution with the iDFT of the rectangle filter:  $h(n) = \text{sinc}\left(\frac{n}{3}\right)$  whose range of the first lobe is 6 samples. We can then conclude that for a generic signal the circular convolution error is present and, due to the narrowness of the filter its effect operates significantly just of a few samples close to the signal tails. However, since in this case the signal is input periodic and the number of samples considered is a multiple of the samples in a period no error will be introduced.

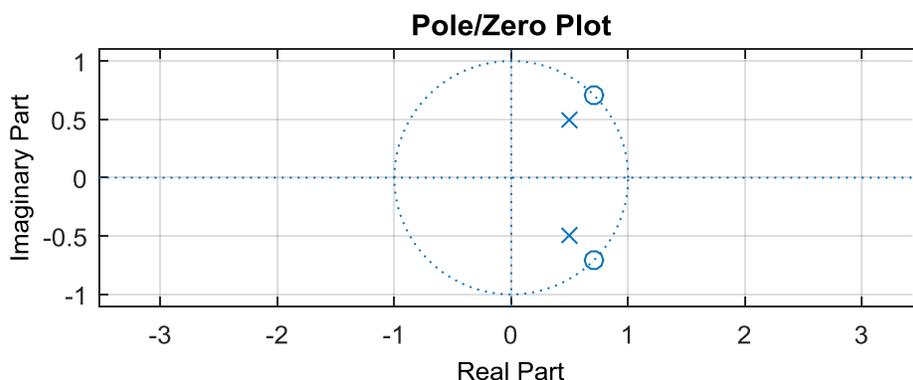
### Ex.2

The spectrum of the signal will be the following:  $X(f) = 5\delta(f) + \frac{3}{2}\delta(f \pm f_1) + \frac{1}{2}\delta(f \pm f_2)$ ; after

sampling in normalized frequencies it will become:  $X(f) = 5\delta(f) + \frac{3}{2}\delta\left(f \pm \frac{f_1}{f_s}\right) + \frac{1}{2}\delta\left(f \pm \frac{f_2}{f_s}\right) =$

$$= 5\delta(f) + \frac{3}{2}\delta\left(f \pm \frac{f_1}{f_s}\right) + \frac{1}{2}\delta\left(f \pm \frac{f_2}{f_s}\right) = 5\delta(\omega) + \frac{3}{2}\delta\left(\omega \pm \frac{\pi}{4}\right) + \delta(\omega - \pi).$$

Applying the filter  $H(z)$  that has the following zeros and poles: two zeros on the unit circle at  $z_{1,2} = e^{\pm j\frac{\pi}{4}}$  and two poles  $p_{1,2} = 0.7e^{\pm j\frac{\pi}{4}}$ .



Such a filter will remove the two conjugate poles associated to the sinusoid at  $f_1$ , the other two components (the continuous component and the Nyquist component) will get the following amplification:

$$H(\omega = 0 \rightarrow z = 1) = \frac{2 - \sqrt{2}}{1.49 - 0.7\sqrt{2}} \approx 1.17$$

$$H(\omega = \pi \rightarrow z = -1) = \frac{2 + \sqrt{2}}{1.49 + 0.7\sqrt{2}} \approx 1.38$$

The output will then be:

$$y(n) = 5 \cdot 1.17 + 1.38 \cdot \cos\left(2\pi \frac{f_2}{f_s} n\right) = 5.85 + 1.38 \cdot (-1)^n$$

### Ex.3

```
n=[0:1023];
fs=80000;
f1=10000;
f2=40000;
x=5+3*cos(2*pi*f1/fs*n)+cos(2*pi*f2/fs*n);
x_up=zeros(1,3*length(n));
x_up(1:3:3*1024)=x;
index=[0:1024*3-1];%row and column index
W=exp(-j*2*pi/(3*1024)*(index'*index));
X_up=W*x_up';
% the low pass will remove all frequencies between pi/3
and 5/3 pi
Y=X_up;
Y(1024*3/6+1:1024*3*5/6)=0;
y=(1/(3*1024))*conj(W)*Y;%inverse DFT
y_down=y(1:2:end);
stem(y_down(1:512));
```