ADVANCED SIGNAL ANALYSIS AND PROCESSING LAB

Lab sheet. No: 10

NAME: Pedasingu Rajesh **Roll. No:** EE21M019

Q1.

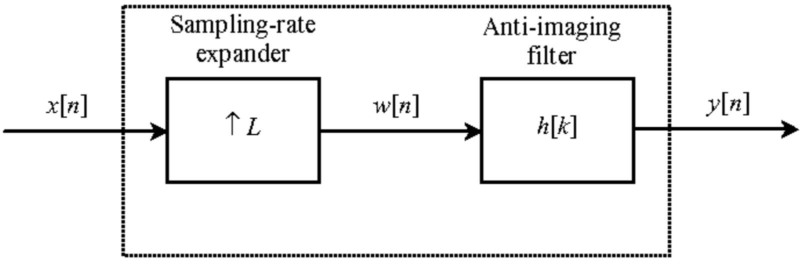
(a) **Let the continuous signal xc(t) = cos(10πt) + cos(20πt) is sampled at 20 Hz to obtain x(n). Perform up sampling by 5, perform appropriate low pass filtering to get back the original signal spectrum and then down sample by 4. Compare x(n) and xd(n) and their spectrum. What is sampling frequency of the processed signal.**

(b) **If you perform the reverse operations i.e performing down sampling by 4 first and then up sampling do you think you can get back the signal, with any appropriate low-pass filtering?**

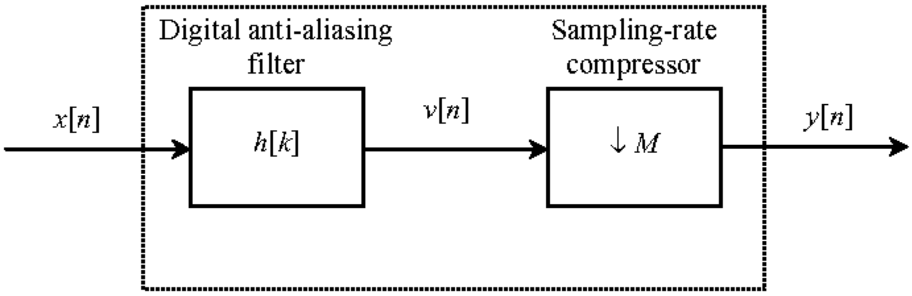
AIM: To generate the discrete signal by sampling the given signal with a sampling frequency 20Hz. Perform up sampling by 5, perform appropriate low pass filtering to get back the original signal spectrum and then down sample by 4 And compare the down sampled spectrum with original spectrum.

Short Theory:

up-sampling: Increasing sampling rate of a signal by an integer factor I is known as Interpolation or up-sampling. An increase in the sampling rate by an integer factor I may be done by interpolating (I-1) new samples between successive values of the signals.



Down-sampling: The process of reducing the sampling rate by an integer factor(D) is called decimation of the sampling rate. It is also called down sampling by factor(D).Decimator consists of decimation filter to band limit the signal and down sampler to decrease the sampling rate by an integer factor (D).



Key Commands:

filter % filter(b,a,x) filters the input data x using a rational transfer function defined by the numerator and denominator coefficients b and a.

fir1 %fir1(n,Wn) uses a Hamming window to design an nth-order lowpass, bandpass, or multiband FIR filter with linear phase.

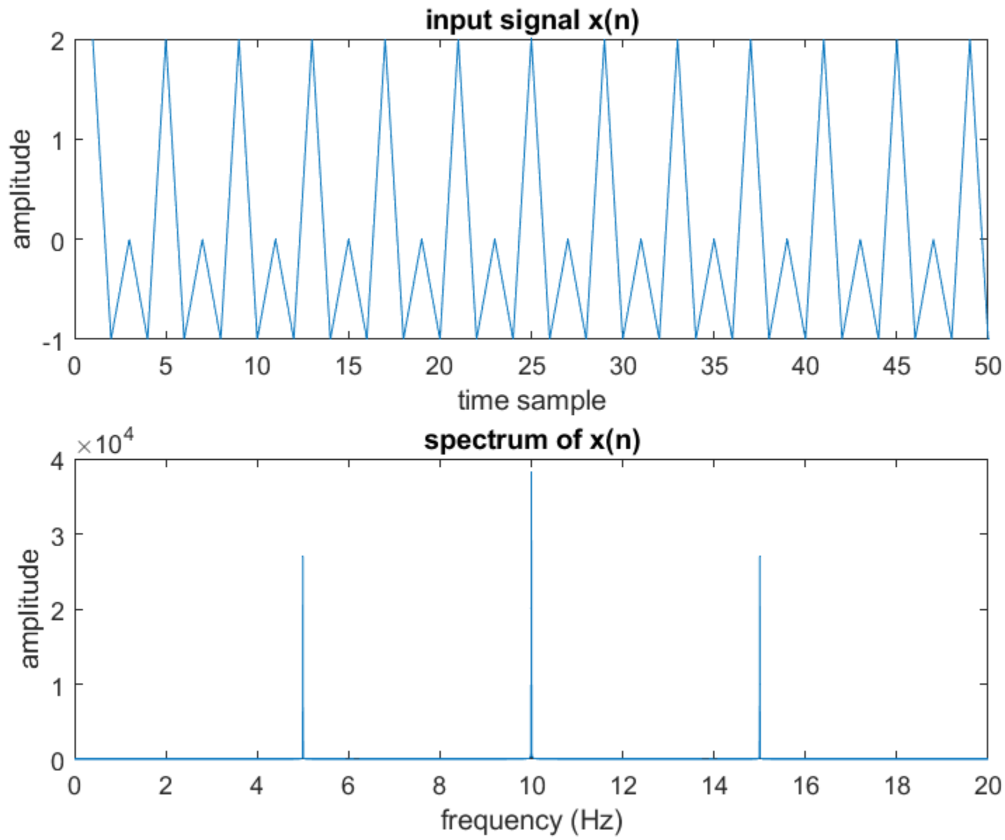
cos % cos(X) returns the cosine for each element of X

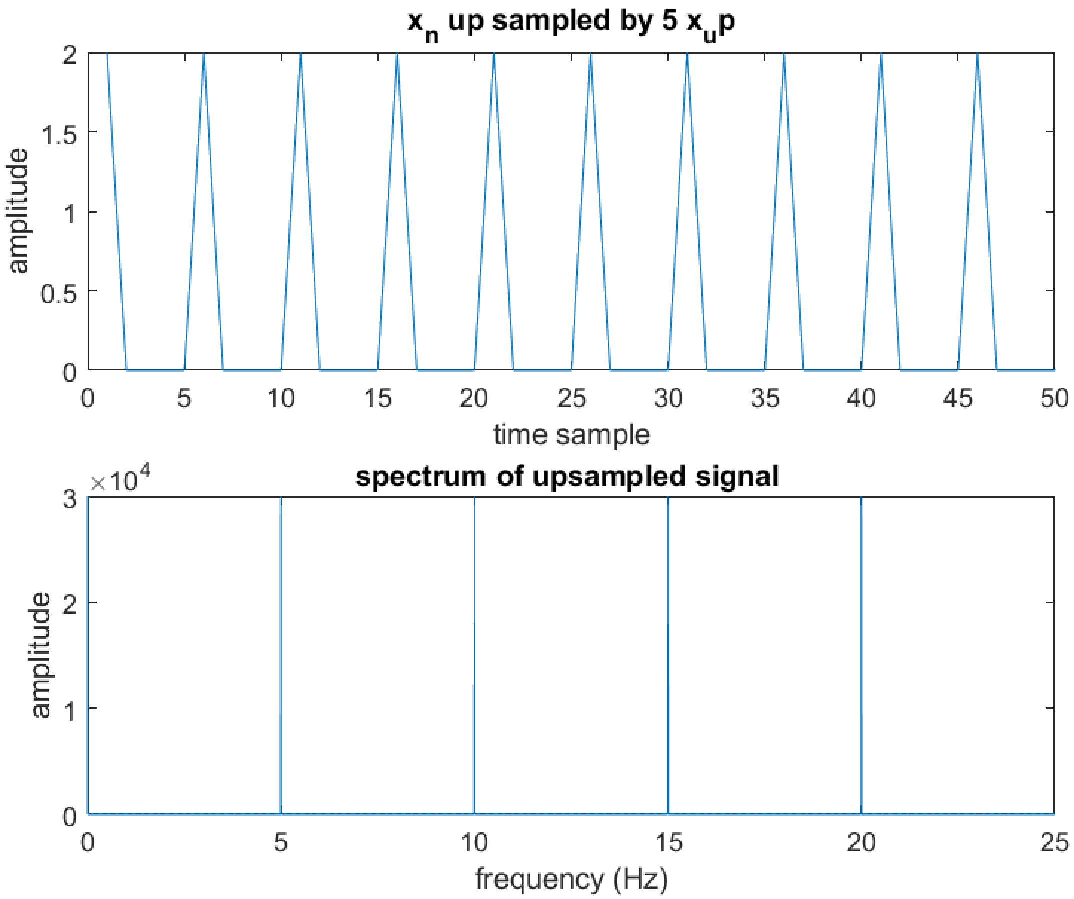
fft % fft(X) computes the discrete Fourier transform (DFT) of X using a fast Fourier transform (FFT) algorithm.

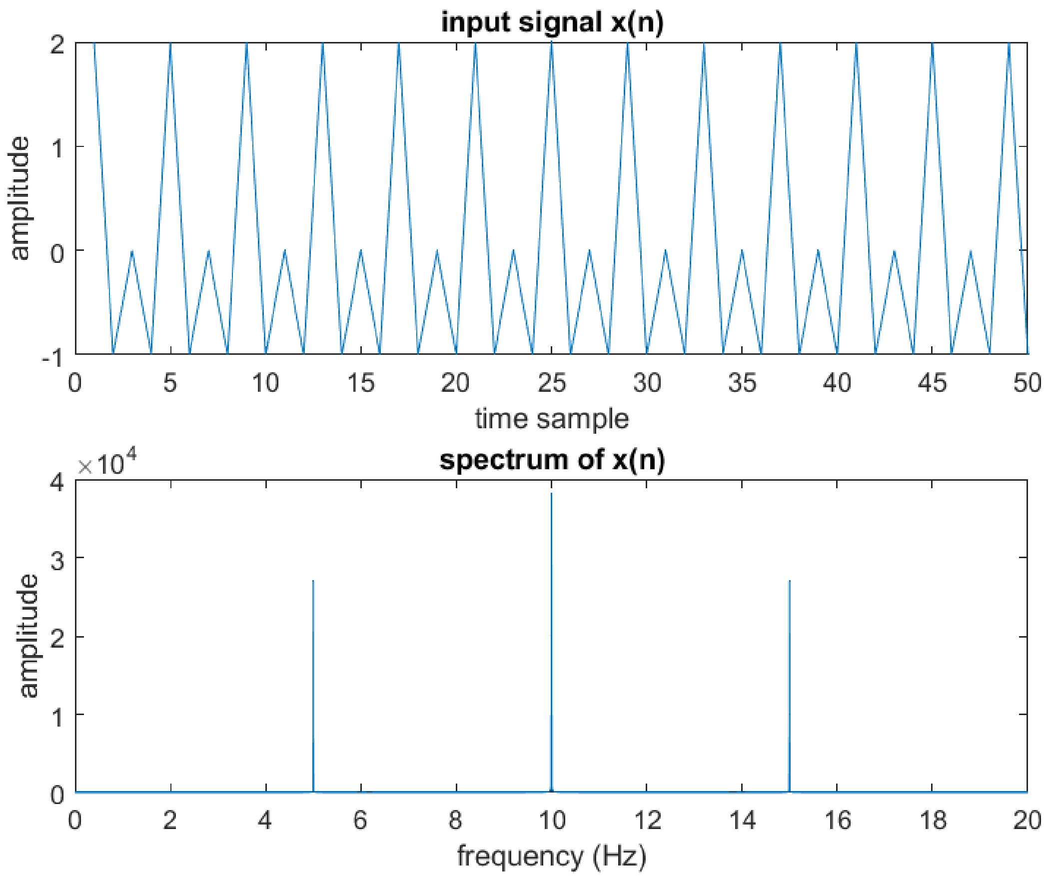
downsample(x,n) decreases the sample rate of x by keeping the first sample and then every nth sample after the first.

upsample(x,n) increases the sample rate of x by inserting n – 1 zeros between samples.

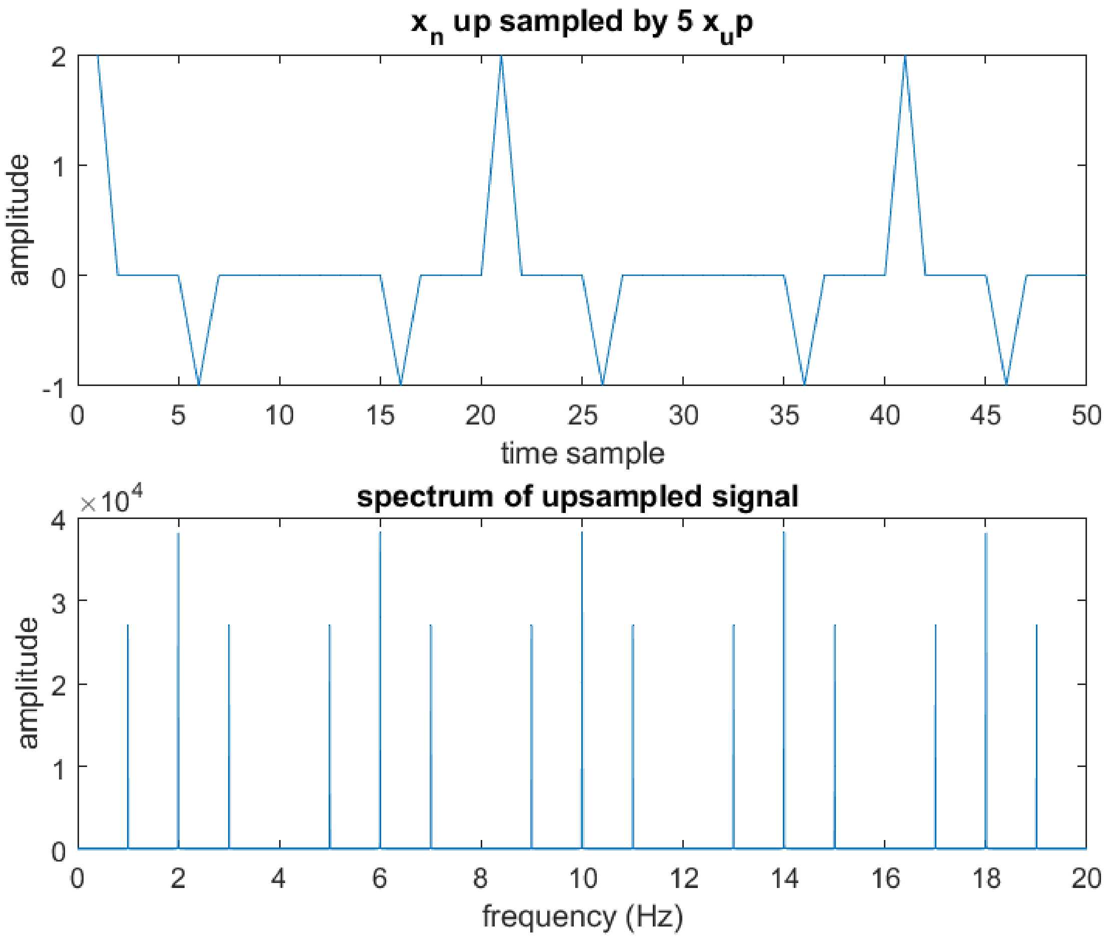
Plots:


input signal x(n)
amplitude
time sample
0 5 10 15 20 25 30 35 40 45 50
-1 0 1 2
spectrum of x(n)
×10^4
amplitude
frequency (Hz)
0 2 4 6 8 10 12 14 16 18 20
0 1 2 3 4

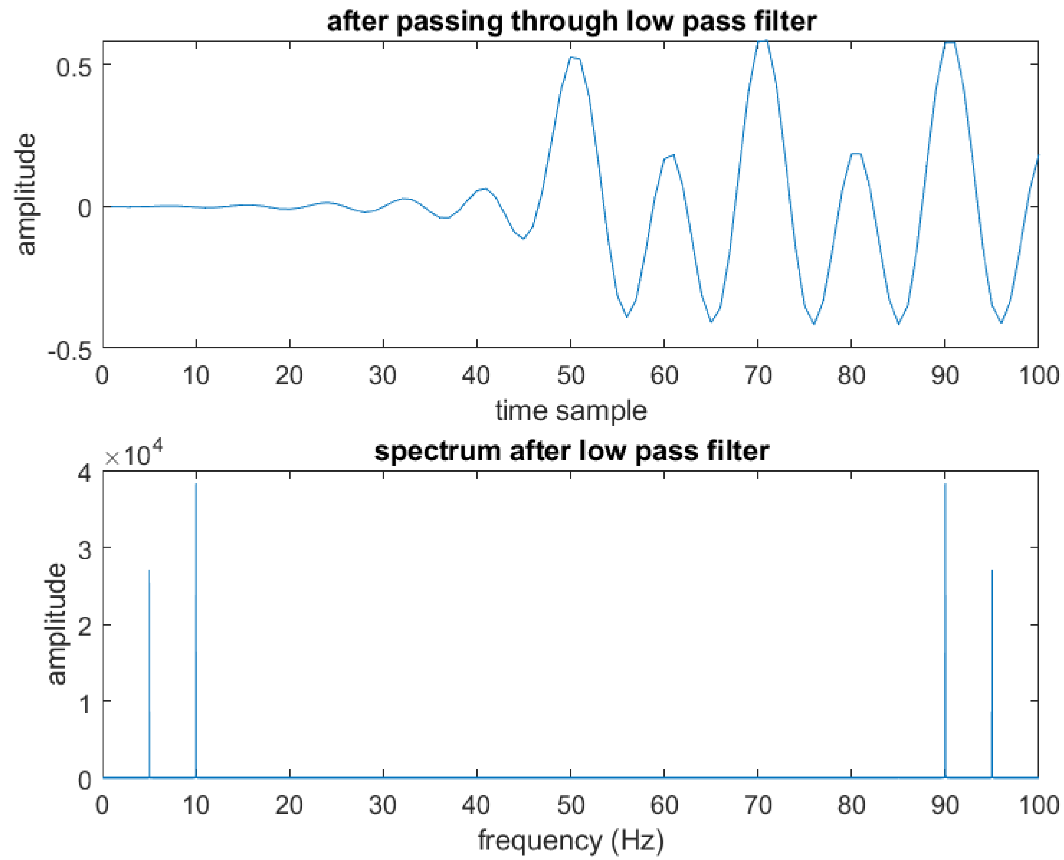

x_n up sampled by 5 $x_u p$
amplitude
time sample
0 5 10 15 20 25 30 35 40 45 50
0 0.5 1 1.5 2
spectrum of upsampled signal
×10^4
amplitude
frequency (Hz)
0 5 10 15 20 25
0 1 2 3

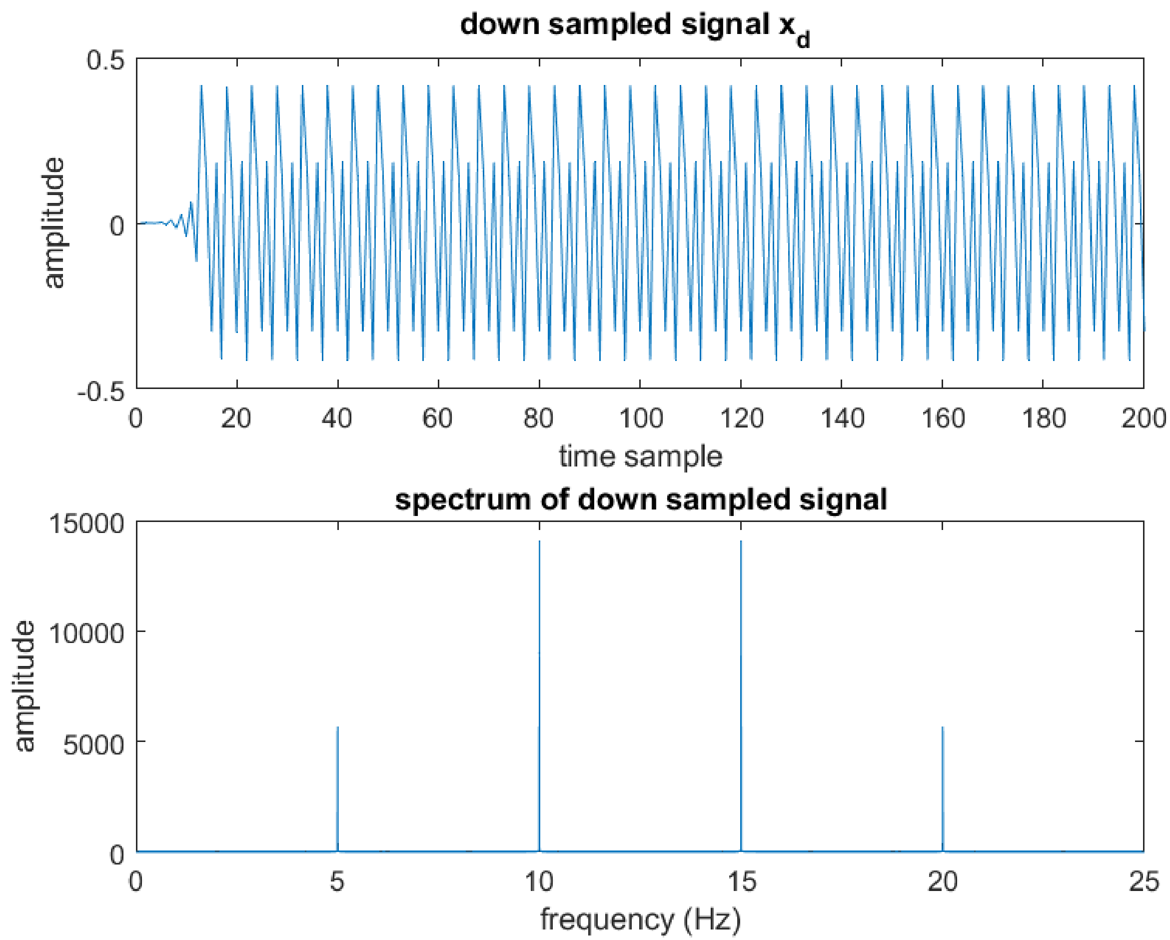


input signal x(n)
amplitude
time sample
spectrum of x(n)
×10^4
amplitude
frequency (Hz)



x_n up sampled by 5 x_up
amplitude
time sample
spectrum of upsampled signal
×10^4
amplitude
frequency (Hz)


after passing through low pass filter
amplitude
time sample
spectrum after low pass filter
amplitude
frequency (Hz)


down sampled signal x_d
amplitude
time sample
spectrum of down sampled signal
amplitude
frequency (Hz)

Inferences/comments:

1) After performing the up sample by 5 the sampling frequency is Fs1= 5*Fs=100Hz
2) After performing the down sample by 4 the sampling frequency is Fs2= Fs1/4=25Hz
3) Final sampling frequency after performing up sample by 5 followed by down sample by 4 is Fs=25 Hz.
4) After performing the Up sampling the spectrum is gets reated in frequency domain, so to get back the original signal spectrum by passing through the low pass filter.
5) If the sampling frequency is Fs<2*max(Fm) then After performing the Down sampling we get the aliasing in the frequency domain. We can not recover the original signal.
6) Given signal is down sampled by M , if the signal is not band-limited to π / M, down-sampling results in aliasing . Aliasing can be avoided if x(n) is low-pass signal band-limited to the region |w| < π / M

Q2.

(a) Let the continuous signal xc(t) = sin(20πt) + sin(50πt) is sampled at 80 Hz to obtain x(n). Perform down sampling by 2 to obtain xd(n) and compare x(n) and xd(n) and their spectrum. What is sampling frequency after down sampling.

(b) Does aliasing occurred in the down sampled spectrum? Design a hamming window based FIR low pass filter (of appropriate cut off frequency), and filter the original signal, in frequency domain, before down sampling to avoid aliasing. Compare the spectrum and the signals before and after down sampling, with pre-filtering.

AIM: To generate the discrete signal by sampling the given signal with a sampling frequency 80Hz. Perform down sampling by 2, to obtain xd(n) and compare x(n) and xd(n) and their spectrum. To Design a hamming window based FIR low pass filter (of appropriate cut off frequency), and filter the original signal, in frequency domain, before down sampling to avoid aliasing. Compare the spectrum and the signals before and after down sampling, with pre-filtering.

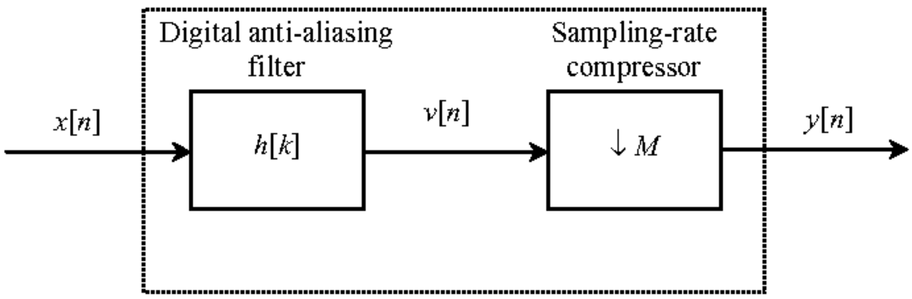
Short Theory:

Down-sampling: The process of reducing the sampling rate by an integer factor(D) is called decimation of the sampling rate. It is also called down sampling by factor(D).Decimator consists of decimation filter to band limit the signal and down sampler to decrease the sampling rate by an integer factor (D).

Decimation

– Reduce the sampling rate of a discrete-time signal.

– Low sampling rate reduces storage and computation requirements.



Aliasing can be avoided if x(n) is low-pass signal band-limited to the region |w| < p / M.. In most applications, the down-sampler is preceded by a low-pass digital filter called "decimation filter".

To prevent aliasing at a lower rate, the digital filter h[k] is used to band-limit the input signal to less than Fs /2M beforehand. Sampling rate reduction is achieved by discarding M-1 samples for every M samples of the filtered signal w[n].

Key Commands:

filter % filter(b,a,x) filters the input data x using a rational transfer function defined by the numerator and denominator coefficients b and a.

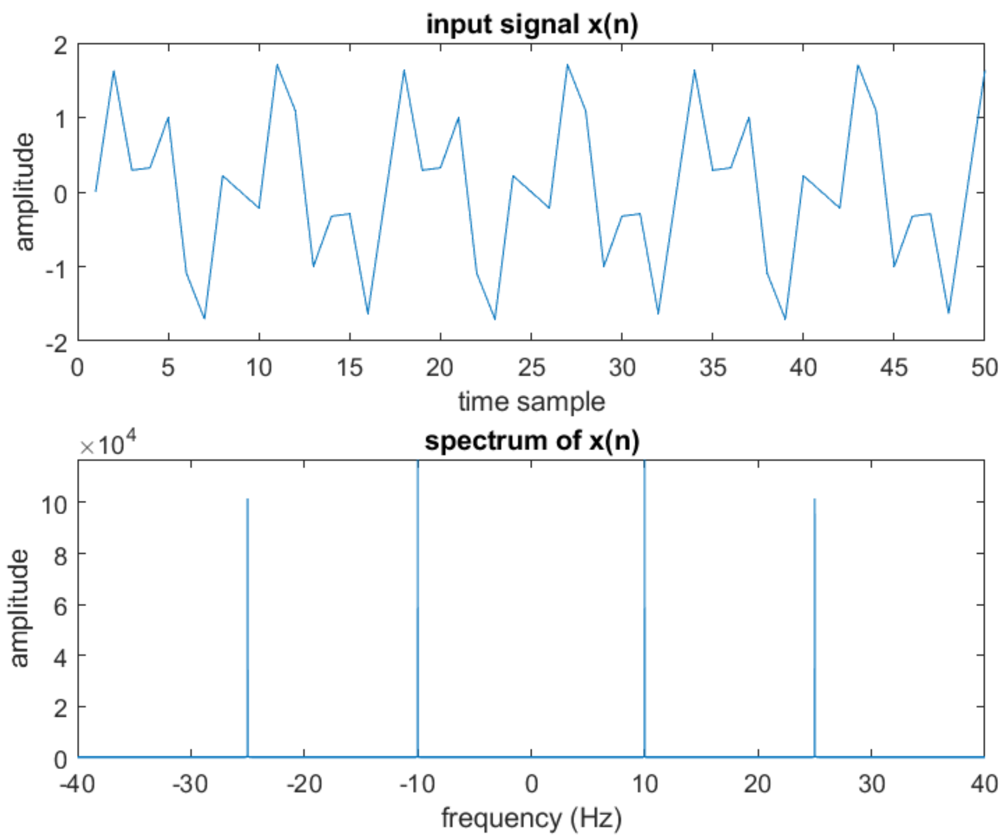
fir1 %fir1(n,Wn) uses a Hamming window to design an nth-order lowpass, bandpass, or multiband FIR filter with linear phase.

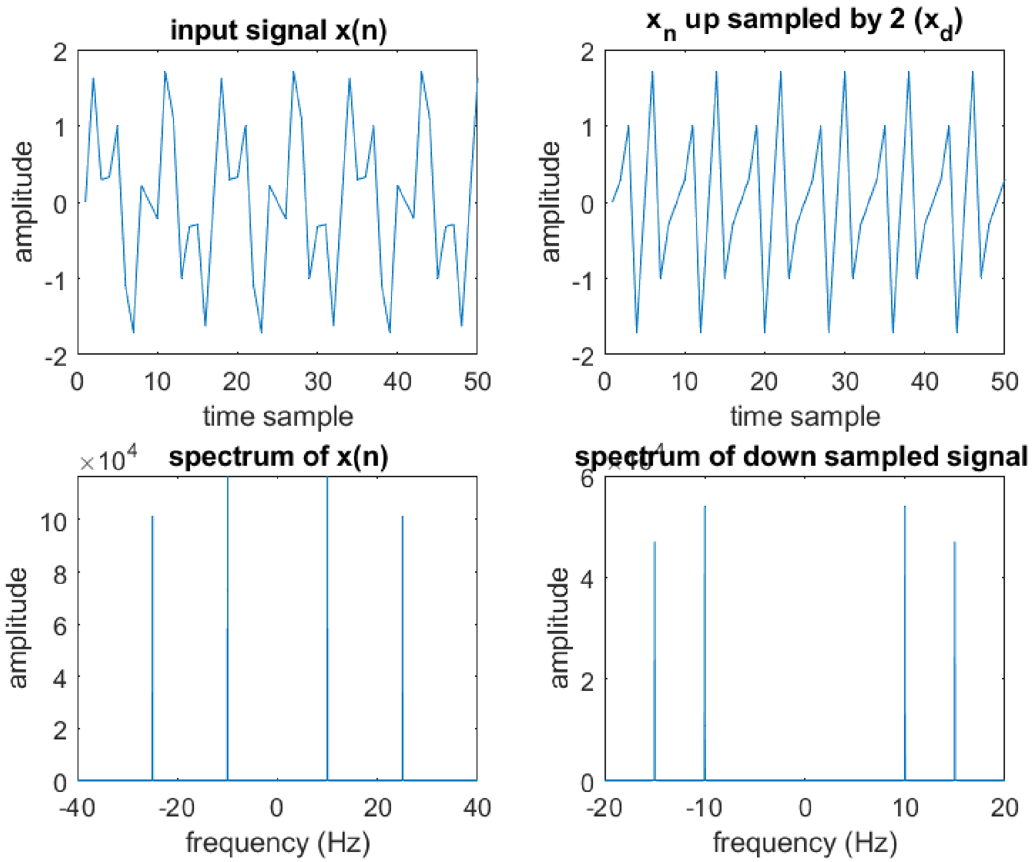
cos % cos(X) returns the cosine for each element of X

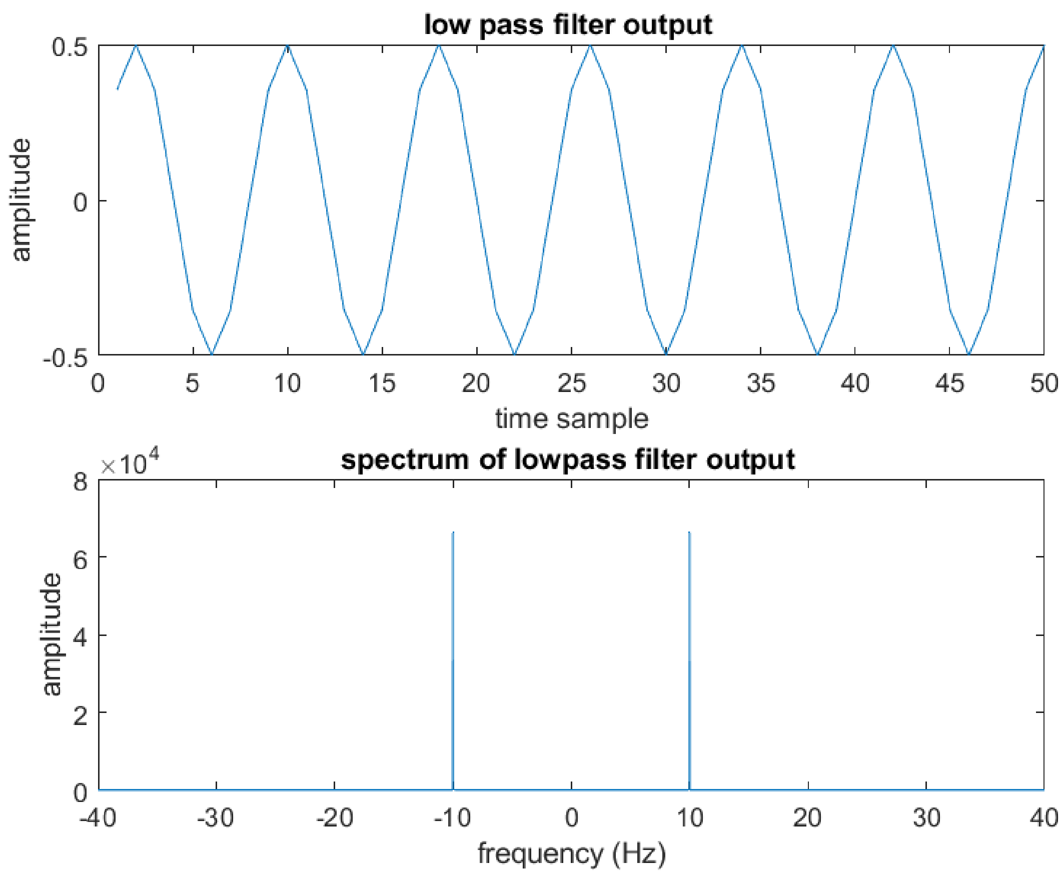
fft % fft(X) computes the discrete Fourier transform (DFT) of X using a fast Fourier transform (FFT) algorithm.

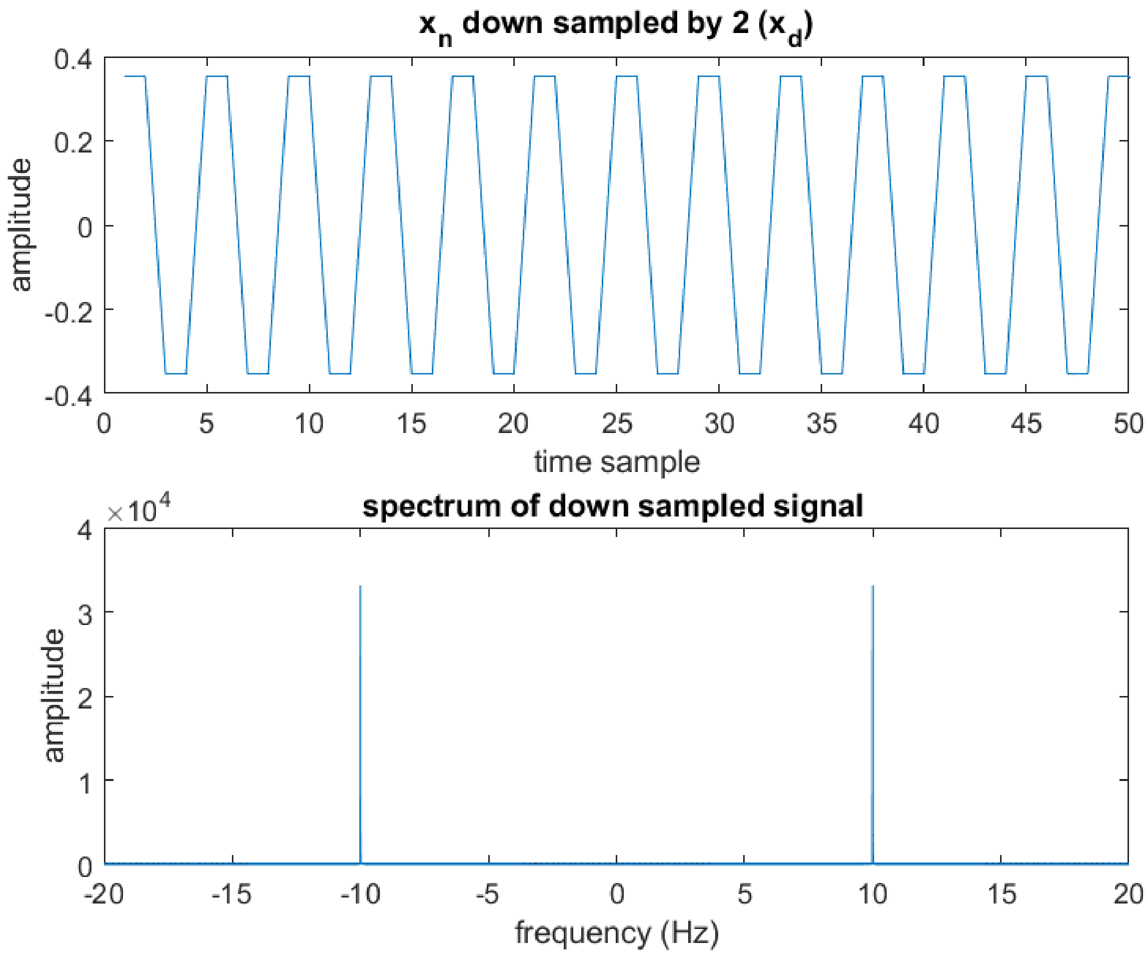
downsample(x,n) decreases the sample rate of x by keeping the first sample and then every nth sample after the first.

Plots:




input signal x(n)
amplitude
time sample
x_n up sampled by 2 (x_d)
amplitude
time sample
spectrum of x(n)
$\times 10^4$
amplitude
frequency (Hz)
spectrum of down sampled signal
$\times 10^4$
amplitude
frequency (Hz)


low pass filter output
amplitude
time sample
spectrum of lowpass filter output
$\times 10^4$
amplitude
frequency (Hz)



Inferences/comments:

1) After performing the down sample by 2 the sampling frequency is Fs1= 2/Fs=40Hz
2) If the sampling frequency is Fs<2*max(Fm) then After performing the Down sampling we get the aliasing in the frequency domain. We can not recover the original signal.
3) Given signal is down sampled by M , if the signal is not band-limited to π / M, down-sampling results in aliasing . Aliasing can be avoided if x(n) is low-pass signal band-limited to the region |w| < π / M

Q3.

(a) **Let the continuous signal xc(t) = sin(20πt) + sin(50πt) is sampled at 80 Hz to obtain x(n). Perform up sampling by 2 to obtain xe(n) and compare x(n) and xe(n) and their spectrum. What is sampling frequency after up sampling.**

(b) **Does aliasing occurred in the up sampled spectrum? Design a hamming window based FIR low pass filter (of appropriate cut off frequency), and filter the up sampled signal in frequency domain to get back the original spectrum.**

AIM: To generate the discrete signal by sampling the given signal with a sampling frequency 80Hz. Perform up sampling by 2, to obtain xd(n) and compare x(n) and xd(n) and their spectrum. To Design a hamming window based FIR low pass filter (of appropriate cut off frequency), and filter the original signal, in frequency domain, after up sampling to get back the original signal.

Short Theory:

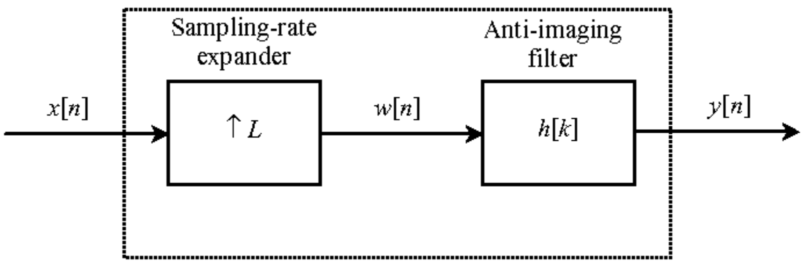
Multi-rate signal processing: The process of converting a signal from a given rate to a different rate is called sampling rate conversion. Systems that employ multiple sampling rates in the processing of digital signals are called multi rate digital signal processing.

up-sampling: Increasing sampling rate of a signal by an integer factor I is known as Interpolation or up-sampling. An increase in the sampling rate by an integer factor I may be done by interpolating (I-1) new samples between successive values of the signals.

- **Interpolation**

– Increase the sampling rate of a discrete-time signal.

– Higher sampling rate preserves fidelity



Key Commands:

filter % filter(b,a,x) filters the input data x using a rational transfer function defined by the numerator and denominator coefficients b and a.

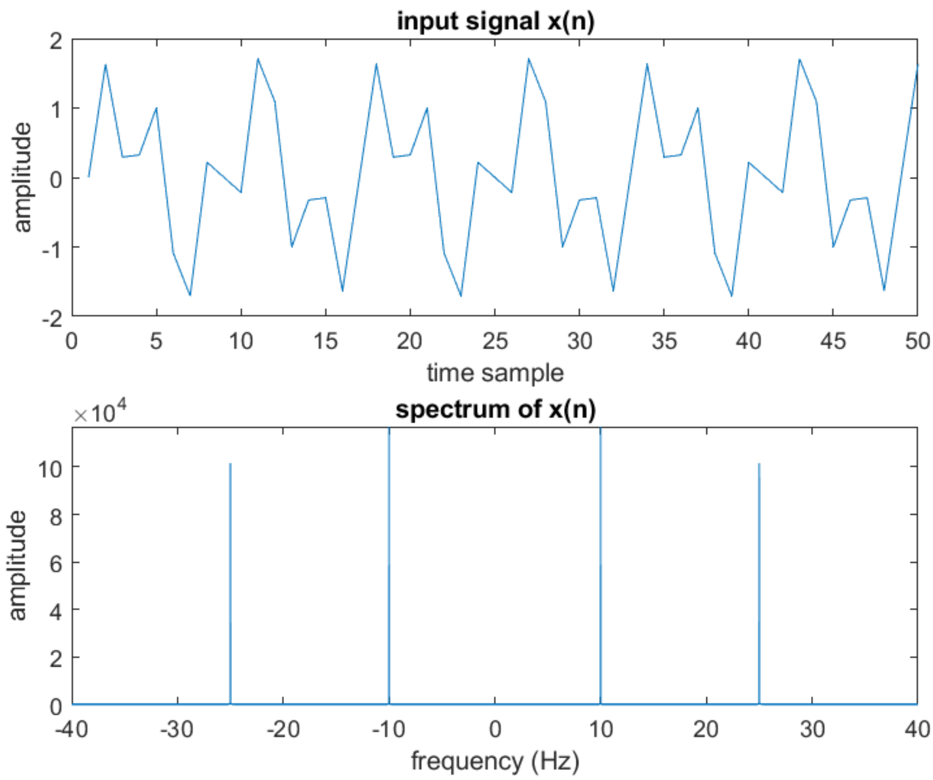
fir1 %fir1(n,Wn) uses a Hamming window to design an nth-order lowpass, bandpass, or multiband FIR filter with linear phase.

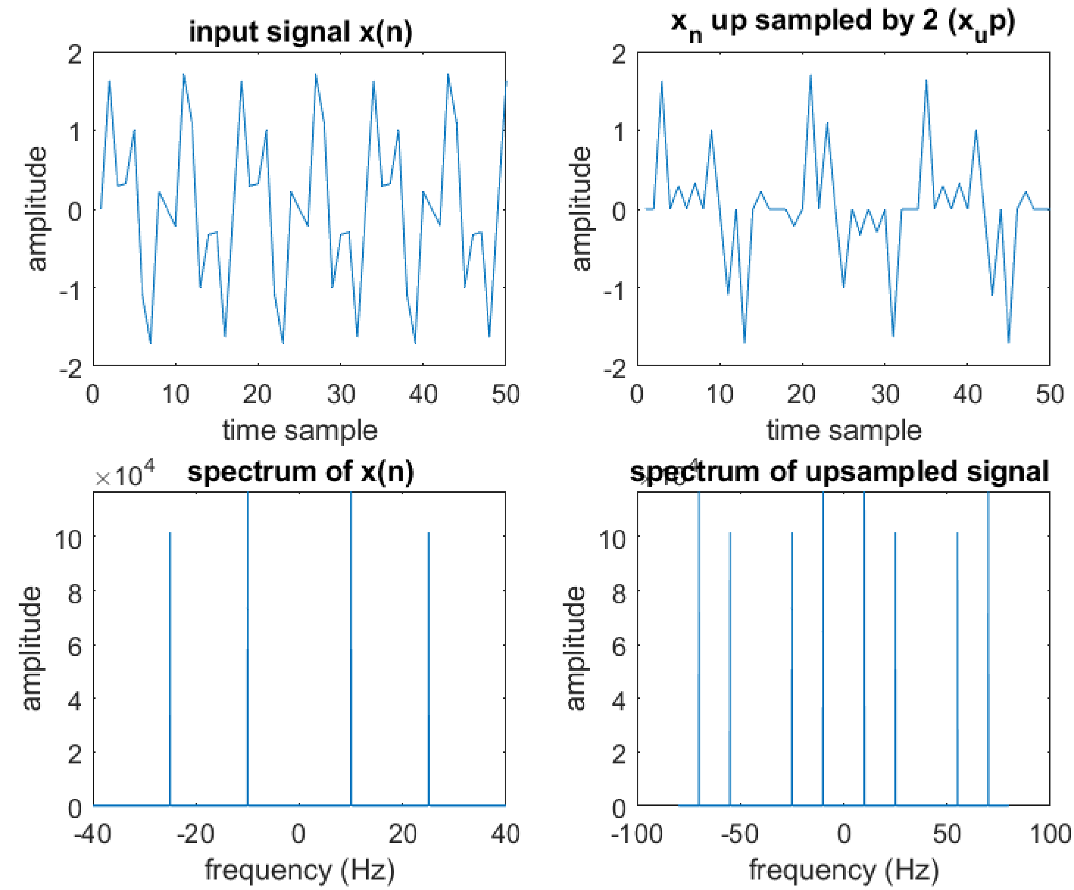
cos % cos(X) returns the cosine for each element of X

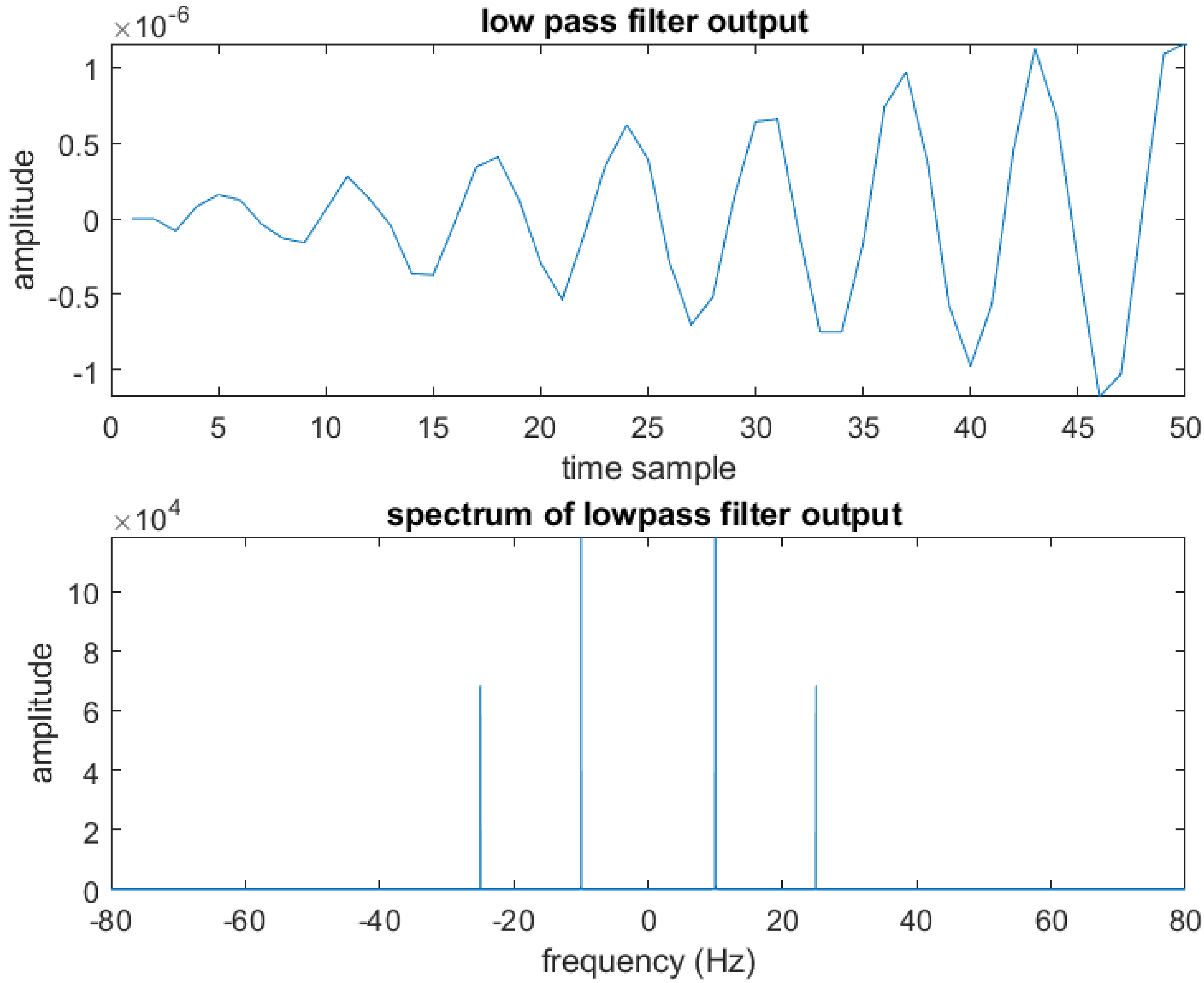
fft % fft(X) computes the discrete Fourier transform (DFT) of X using a fast Fourier transform (FFT) algorithm.

upsample(x,n) increases the sample rate of x by inserting n – 1 zeros between samples.

Plots:




input signal x(n)
amplitude
time sample
x_n up sampled by 2 ($x_u p$)
amplitude
time sample
spectrum of x(n)
$\times 10^4$
amplitude
frequency (Hz)
spectrum of upsampled signal
amplitude
frequency (Hz)


low pass filter output
$\times 10^{-6}$
amplitude
time sample
spectrum of lowpass filter output
$\times 10^4$
amplitude
frequency (Hz)

Inferences/comments:

1) After performing the up sample by 2 the sampling frequency is Fs1= 2*Fs=160Hz
2) After performing the Up sampling, the spectrum is gets repeated in frequency domain, so to get back the original signal spectrum by passing through the low pass filter.
3) Aliasing doesn't occur in the up sampled spectrum.

Q4. Let the continuous signal xc(t) = sin(20πt)+sin(50πt) is sampled at 80 Hz to obtain x(n). Design an interpolation and decimation processing algorithm to change the sampling rate to 120 Hz.

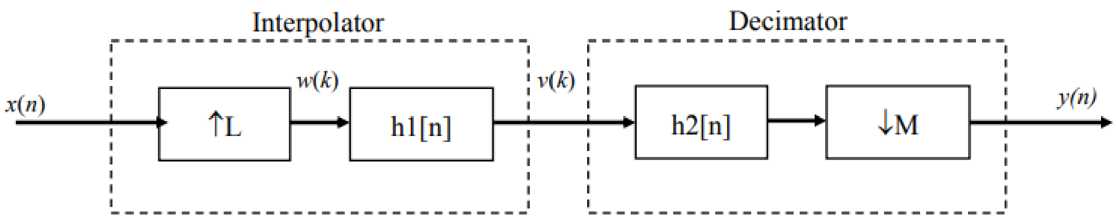
(a) Perform Interpolation first followed by decimation. Apply low pass filtering appropriately to remove the high frequency components after interpolation or as anti-aliasing filter to get back the original signal;

(b) Perform decimation first and then interpolation. In this case also you can design appropriate low pass filters at any stage to get back original signal (if possible). Are the results in question 4a and 4b different? Which is better option to get back the original signal after sampling rate conversion?

AIM: To generate the discrete signal by sampling the given signal with a sampling frequency 80Hz. Design an interpolation and decimation processing algorithm to change the sampling rate to 120 Hz.

Short Theory:

we now consider the general case of sampling rate conversion by a rational factor I/D. we can achieve this sampling rate conversion by first performing interpolation by the factor I and then decimating the output of the interpolator by the factor D. a sampling rate conversion by the rational factor I/D is accomplished by cascading an interpolator with a decimator. We emphasize that the importance of performing the interpolation first and the decimation second is to preserve the desired spectral characteristics of x(n).



We perform Interpolation precedes Decimation because Decimation removes some of the desired frequency components.

Key Commands:

filter % filter(b,a,x) filters the input data x using a rational transfer function defined by the numerator and denominator coefficients b and a.

fir1 %fir1(n,Wn) uses a Hamming window to design an nth-order lowpass, bandpass, or multiband FIR filter with linear phase.

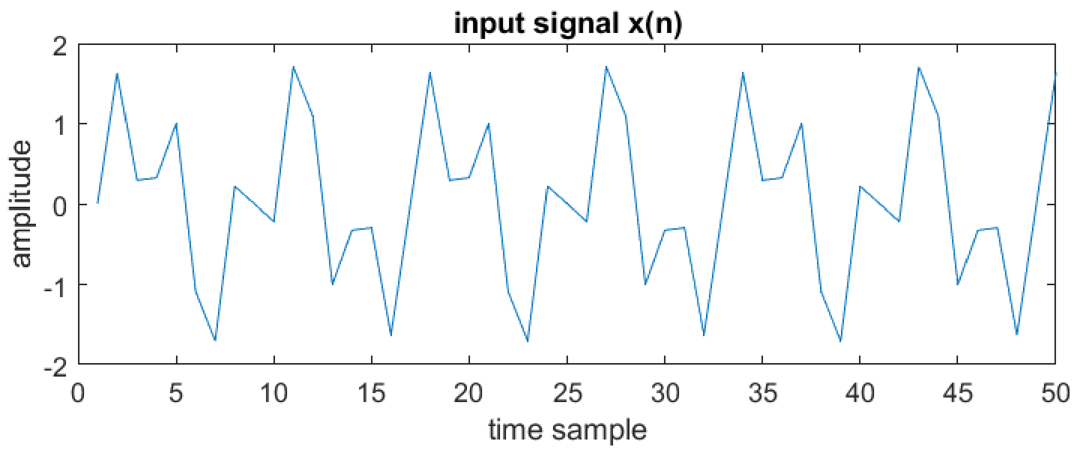
cos % cos(X) returns the cosine for each element of X

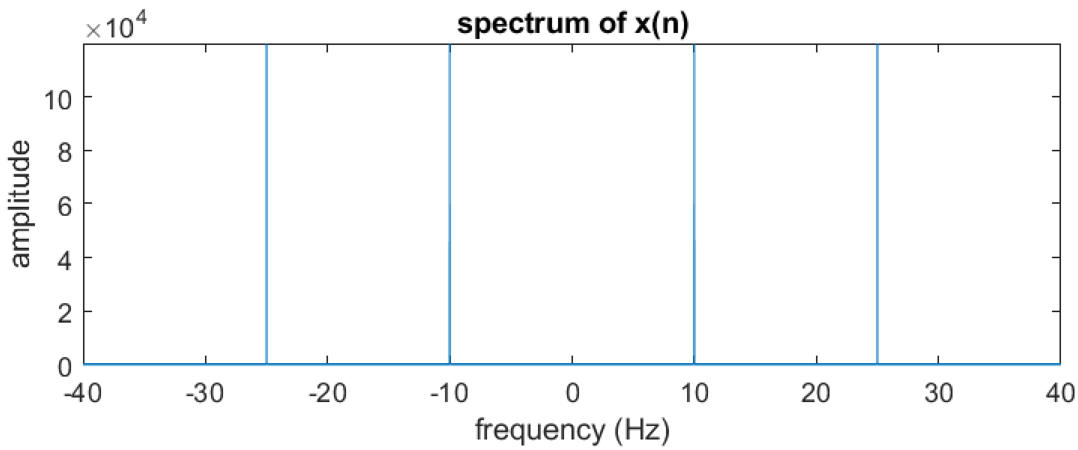
fft % fft(X) computes the discrete Fourier transform (DFT) of X using a fast Fourier transform (FFT) algorithm.

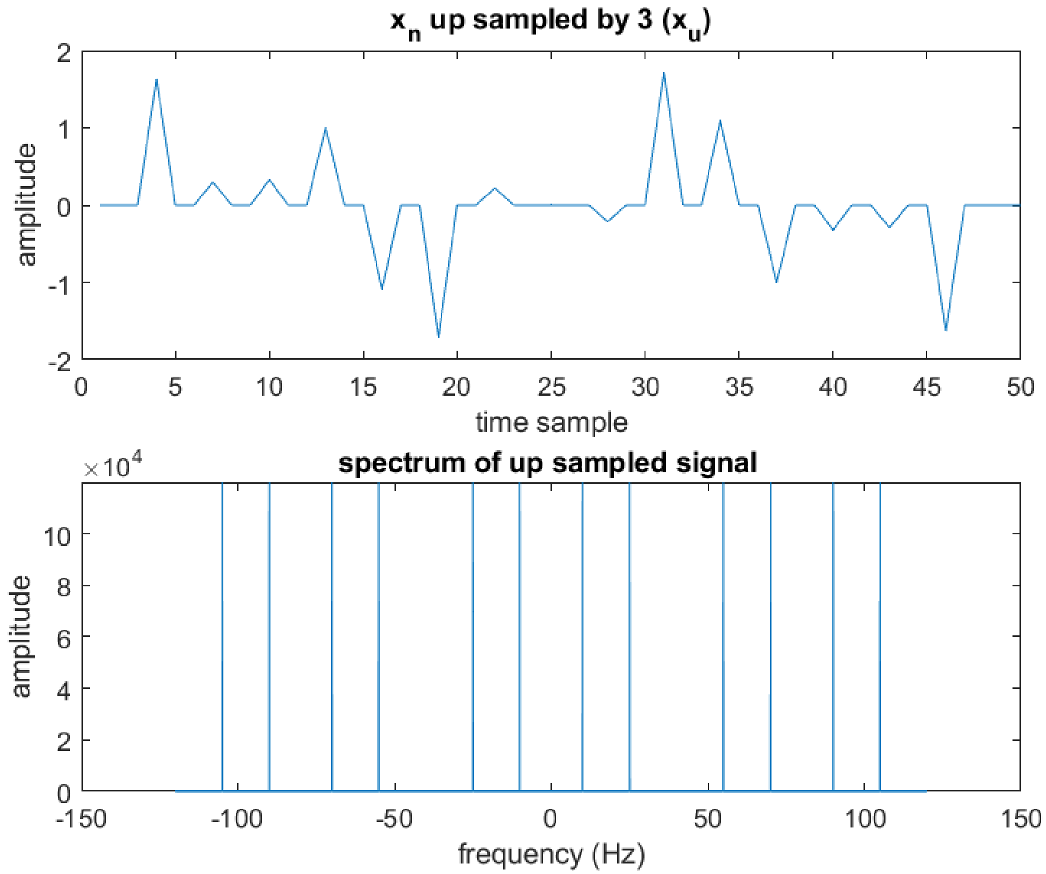
downsample(x,n) decreases the sample rate of x by keeping the first sample and then every nth sample after the first.

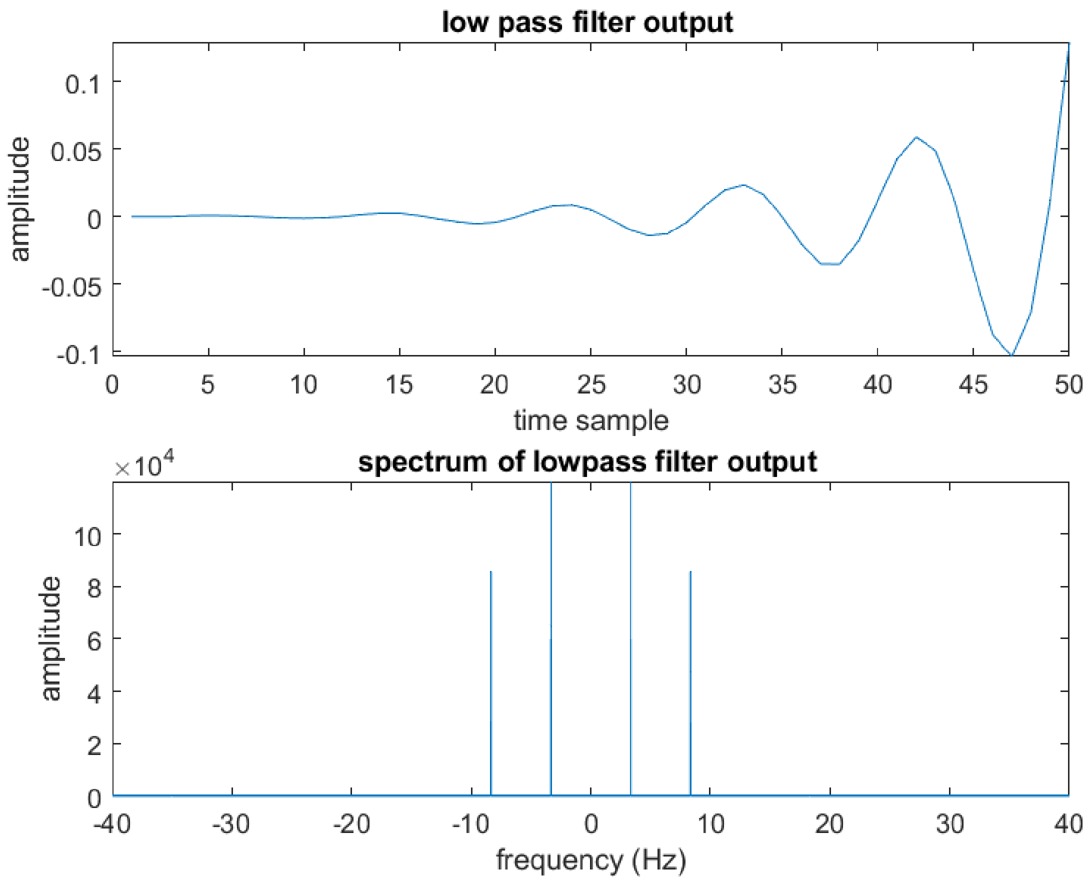
upsample(x,n) increases the sample rate of x by inserting n – 1 zeros between samples.

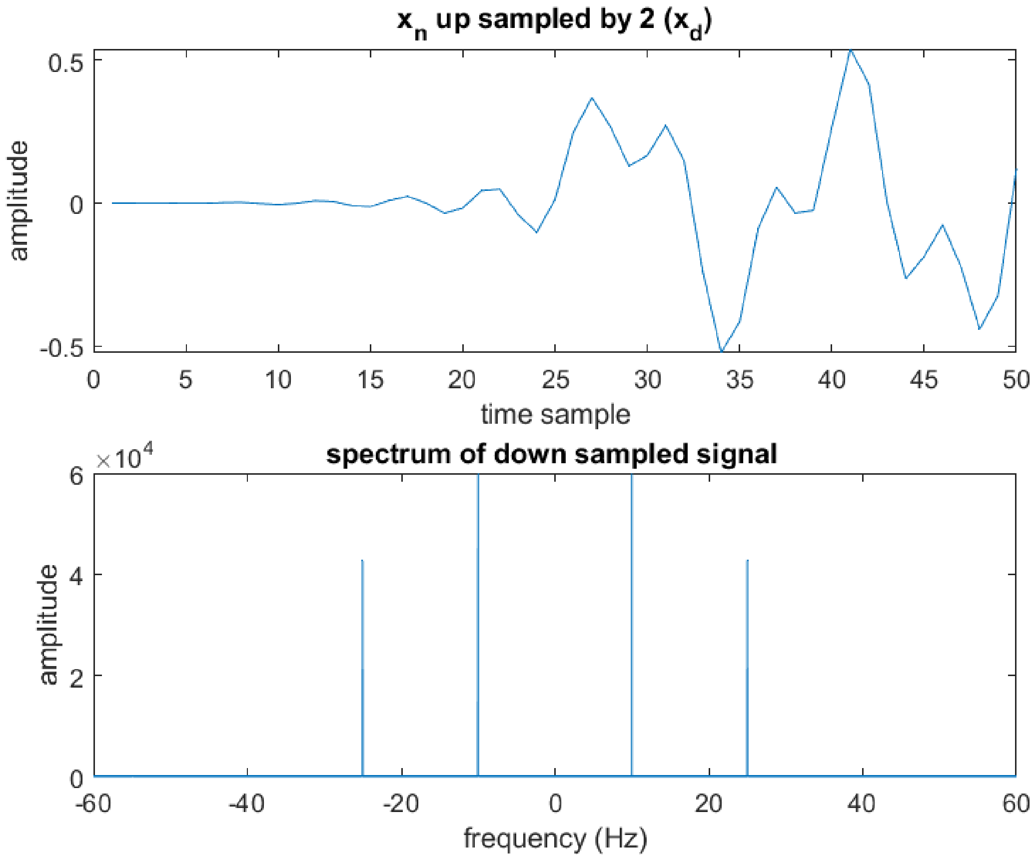
Plots:

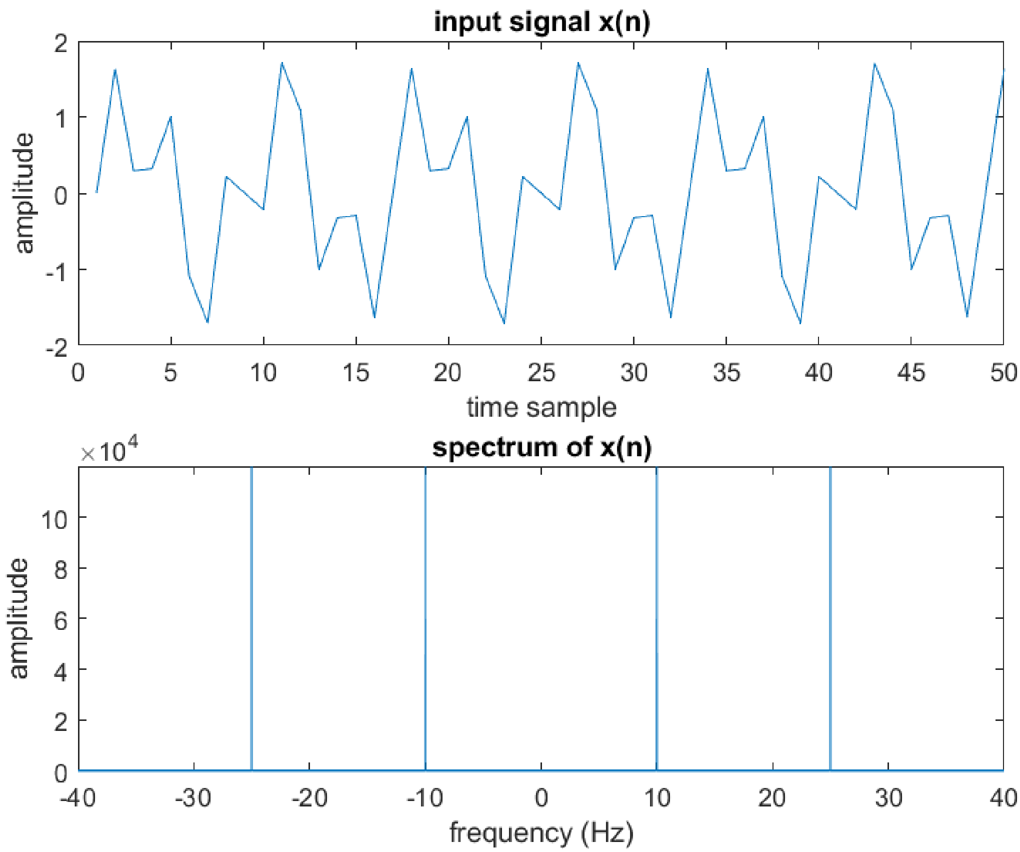


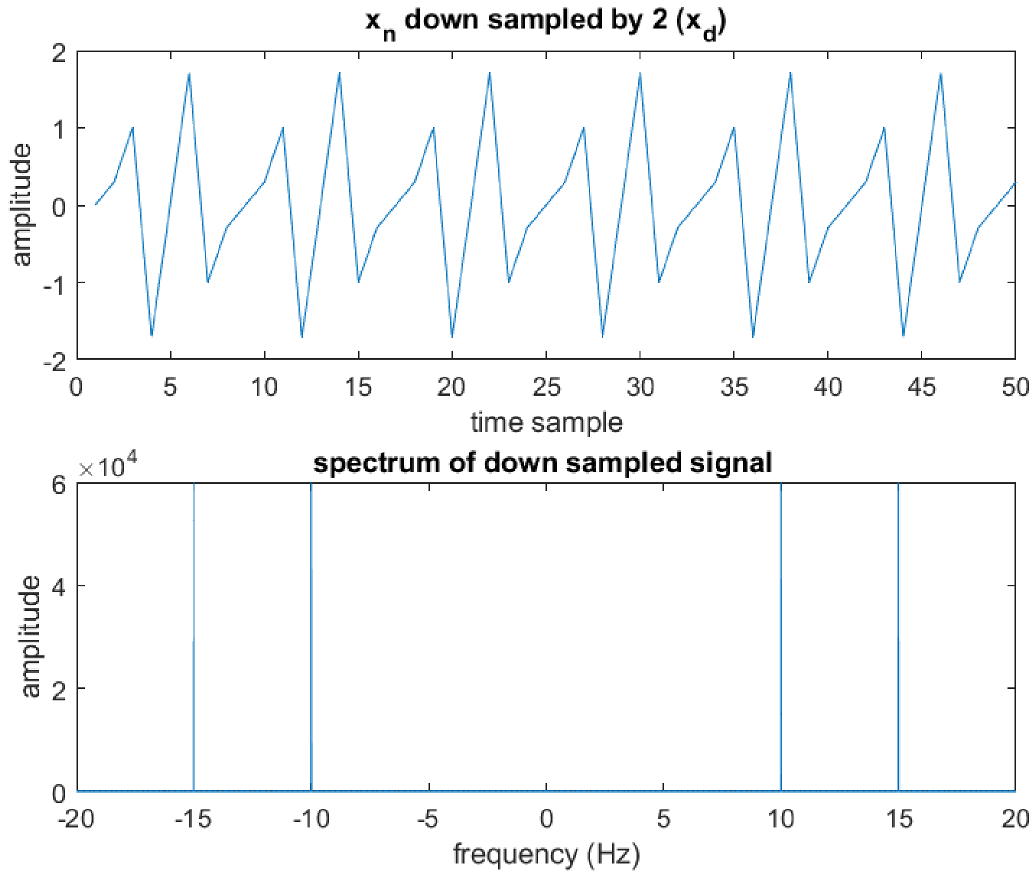



x_n up sampled by 3 (x_u)
amplitude
time sample
spectrum of up sampled signal
×10^4
amplitude
frequency (Hz)

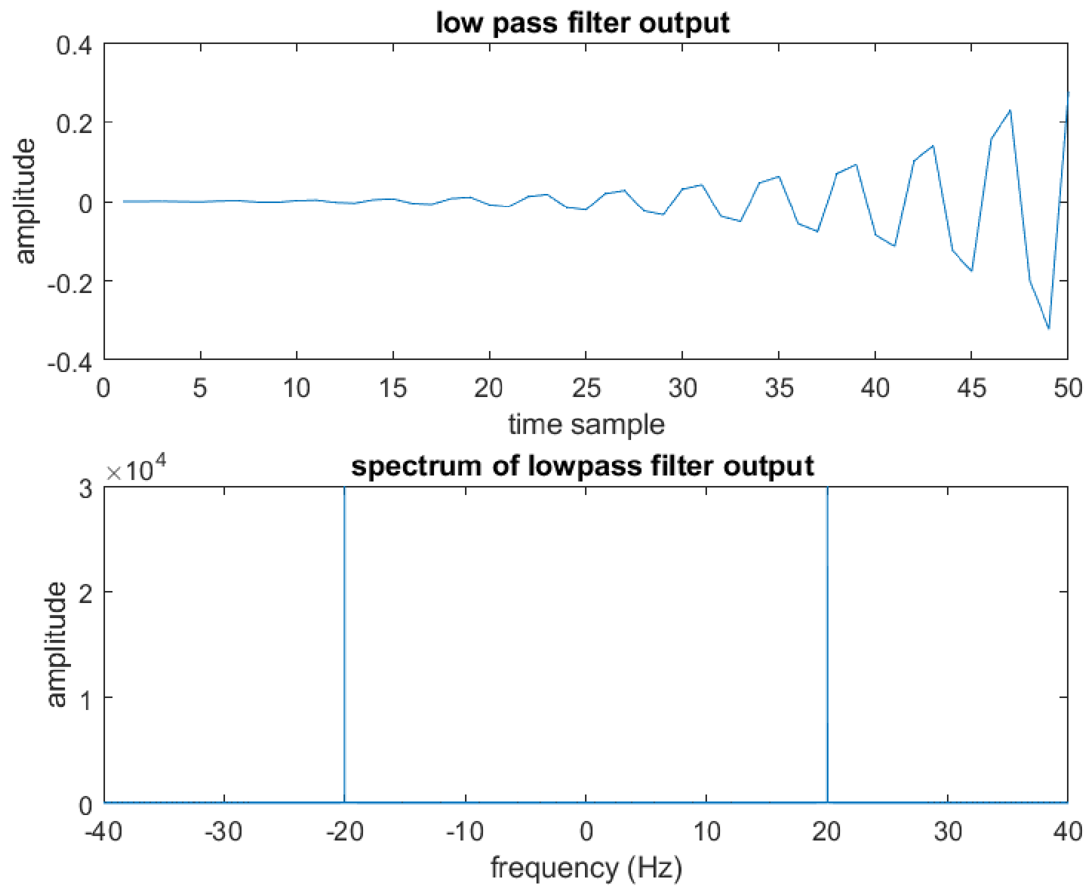

low pass filter output
amplitude
time sample
spectrum of lowpass filter output
×10^4
amplitude
frequency (Hz)


x_n up sampled by 2 (x_d)
amplitude
time sample
spectrum of down sampled signal
$\times 10^4$
amplitude
frequency (Hz)

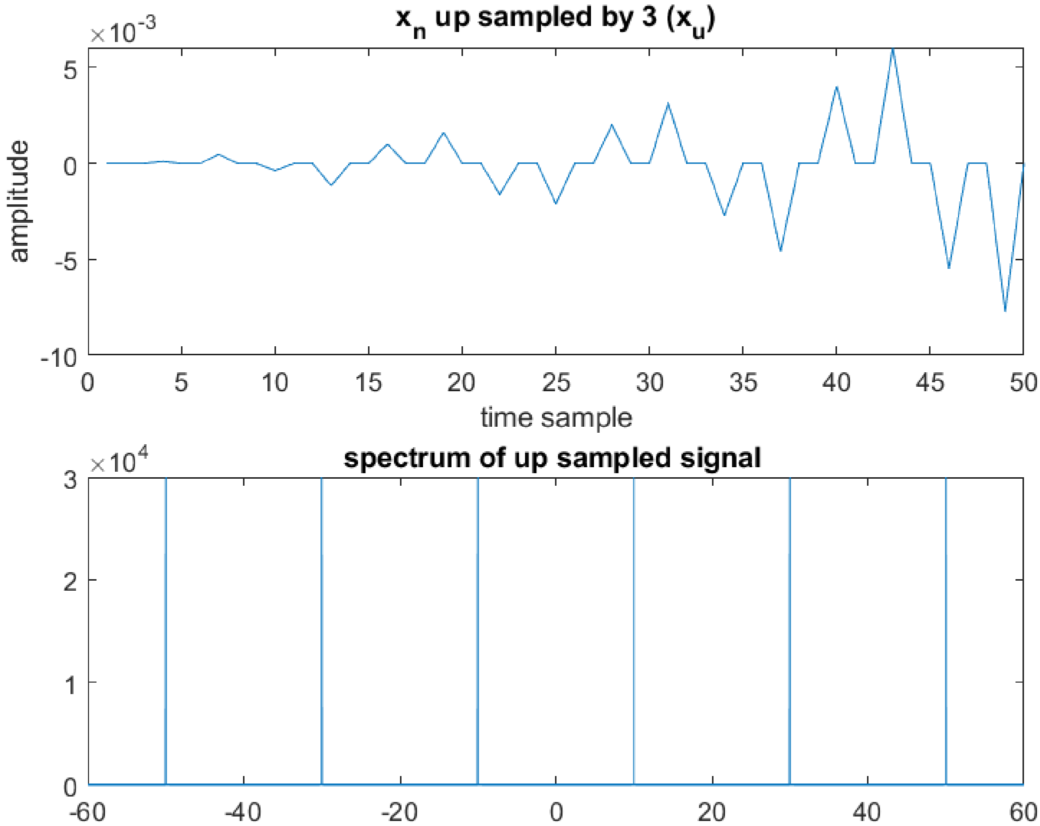

input signal x(n)
amplitude
time sample
spectrum of x(n)
$\times 10^4$
amplitude
frequency (Hz)



x_n down sampled by 2 (x_d)
amplitude
time sample
spectrum of down sampled signal
amplitude
frequency (Hz)



low pass filter output
amplitude
time sample
spectrum of lowpass filter output
amplitude
frequency (Hz)



Inferences/comments:

1) first performing the interpolation followed by decimation then the sampling frequency after interpolation and decimation are respectively are Fs1= 3*Fs=240Hz,Fs2=120Hz.
2) first performing the decimation followed by interpolation then the sampling frequency after interpolation and decimation are respectively are Fs1= 3*Fs=40Hz,Fs2=120Hz.
3) Final sampling frequency after performing up sample by 5 followed by down sample by 4 is Fs=120 Hz.
4) After performing the Up sampling, the spectrum is gets related in frequency domain, so to get back the original signal spectrum by passing through the low pass filter.
5) If the sampling frequency is Fs<2*max(Fm) then After performing the Down sampling we get the aliasing in the frequency domain. We cannot recover the original signal.
6) Given signal is down sampled by M , if the signal is not band-limited to π / M, down-sampling results in aliasing . Aliasing can be avoided if x(n) is low-pass signal band-limited to the region |w| < π / M.