ELEC3104: Digital Signal Processing

Chapter 8: Filter Design

8.1 Introduction

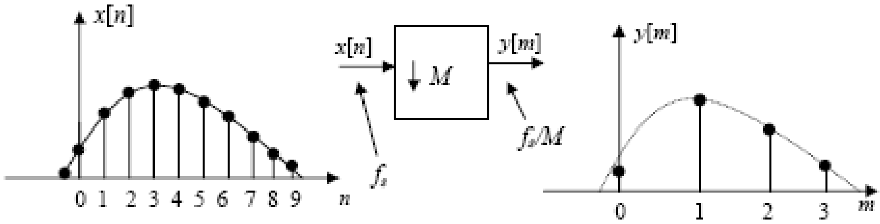
The increasing need in modern digital systems to process data at more than one sampling rate has lead the development of a new sub-area in DSP known as multirate processing. The two primary operations in multirate processing are:

- **Decimation:** The process of decreasing the sampling rate of a given signal. This is sometimes called **down-sampling**.
- **Interpolation:** The process of increasing the sampling rate of a given signal. This is sometimes called **up-sampling**.

The process of decimation and interpolation are the fundamental operations in multirate signal processing.

8.2 Decimation

We confine our attention to a decrease by an integer factor M (e.g., $M = 3$).



The output signal is obtained by taking every M^{th} sample of the input signal hence removing the $M - 1$ interleaving (in-between) samples. If $M = 3$, we should just take every third sample of $x[n]$ to form the desired signal $y[n]$. Generally,

$$y[n] = x[nM]$$

Example 8.1

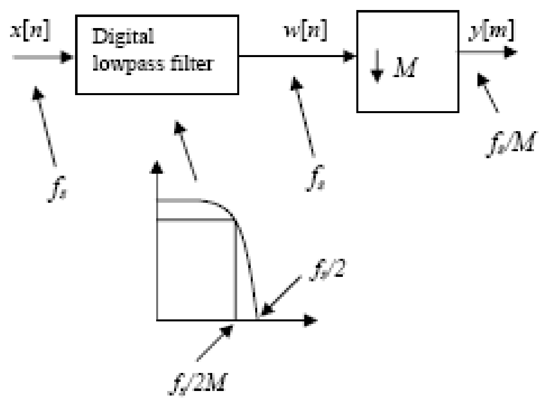
$$x[n] = \{1, 2, 4, 3, 5, -6, -8, 2, -3, 2\}$$

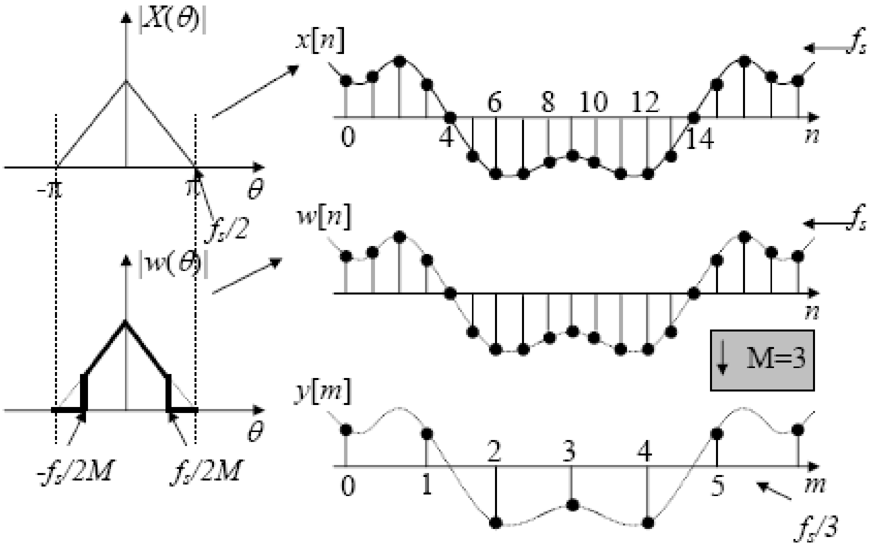
$\downarrow$ Down sample by 2

$$y[m] = \{1,4,5,-8,-3\}$$

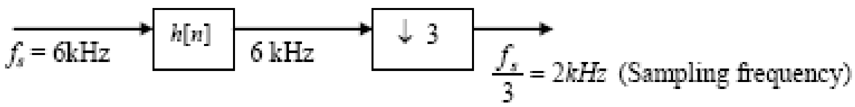
Obviously, it only makes sense to reduce the sampling rate if the information constant of the signal we wish to preserve is band-limited to $f_s/2M$, half the desired sampling rate, since the spectral components above this frequency will be aliased into frequencies below $f_s/2M$ according to the sampling theorem.

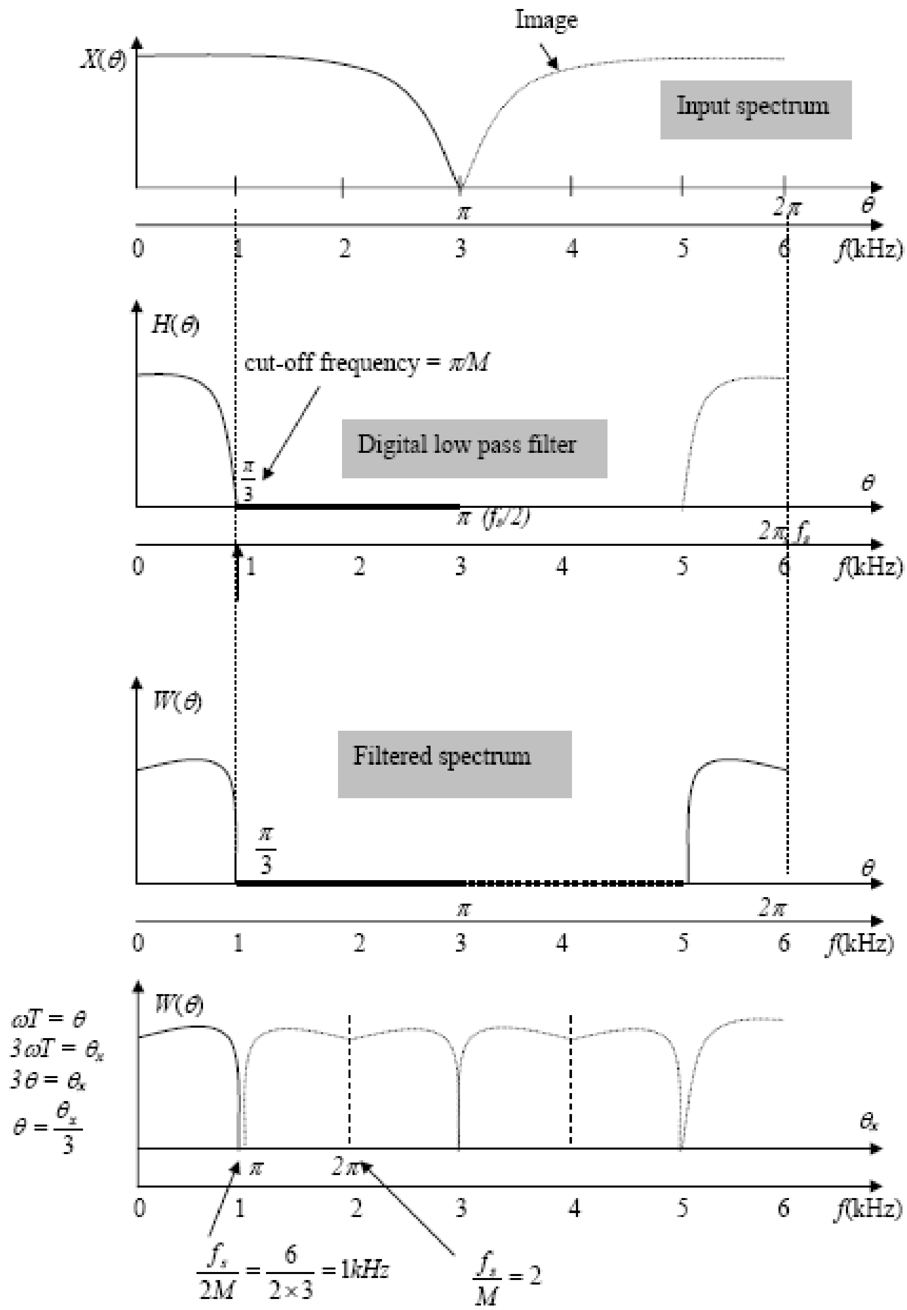
The following figure shows a block diagram representation of an *M*-times decimator. We see that the signal $x[n]$ is first passed through a low-pass filter that attenuates the band from $\frac{f_s/2}{M}$ to $f_s/2$ to prevent aliasing.




|X(θ)|
-π
π
θ
fs/2
|w(θ)|
θ
-fs/2M
fs/2M
x[n]
0
4
6
8
10
12
14
n
fs
w[n]
n
fs
↓ M=3
y[m]
0
1
2
3
4
5
m
fs/3

Example 8.2


fs = 6kHz
h[n]
6 kHz
↓ 3
fs/3 = 2kHz (Sampling frequency)



Note: Such a processing requirement may arise, for example when a speech signal is over sampled at $f_s = 32kHz$. Since we are interested only in a band of $0 - 4\ kHz$, sampling rate can be reduced to $8kH$, so the first step in the decimation process has to be the digital filtering of the signal $x[n]$ to bandlimit it to $f_s/8 = 4kHz$.

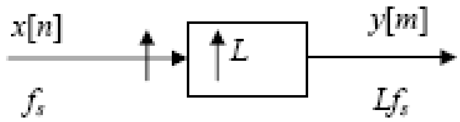
Should we use IIR (recursive) or FIR (non-recursive) for the low pass filtering required ?

Using an IIR filter in this case has an obvious shortcoming. We cannot take advantage of the fact that we only have to compute every N^{th} output, since previous outputs are required to compute the M^{th} output. Thus no saving is realized.

On the other hand, using an FIR filter, in this case implies that we can do our computations at the rate f_s/M. Thus, using an FIR filter in the decimation process will lead to a significantly lower computation rate for filters of the same order. Another advantage of using an FIR filter is the fact that we can easily design linear phase filters and this is desirable in many applications. The main drawback of FIR filters in this application is that considerably higher filter orders are required to realize the low pass filtering than for IIR filters.

8.3 Interpolation

The process of upsampling involves a sampling rate increase.



The sequence $x[n]$ was derived by sampling $x(t)$ at a sampling rate of f_s, and we want to obtain a sequence $y[m]$ that approximates as closely as possible the sequence that would have been obtained had we sampled $x(t)$ at the rate Lf_s. This involves inserting between any two samples $x[n]$ and $x[n-1]$ an additional L-1 samples. Generally,

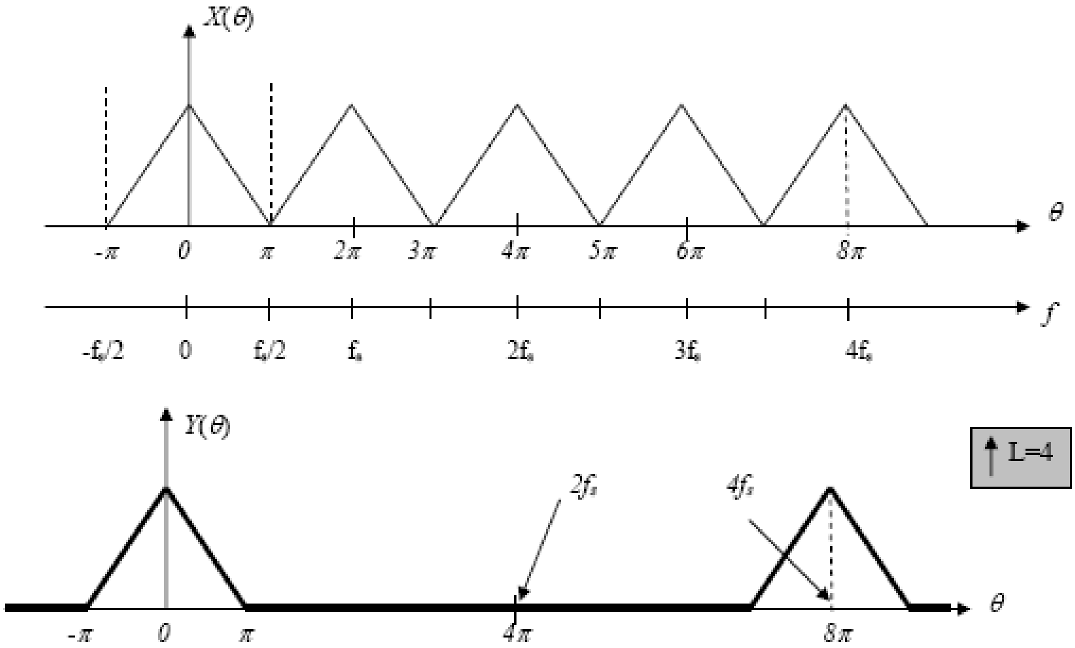
$$y[n] = \begin{cases} x\left[\frac{n}{L}\right], & n = 0, \pm L, \pm 2L, \ldots \\ 0, & otherwise \end{cases}$$

Example 8.3

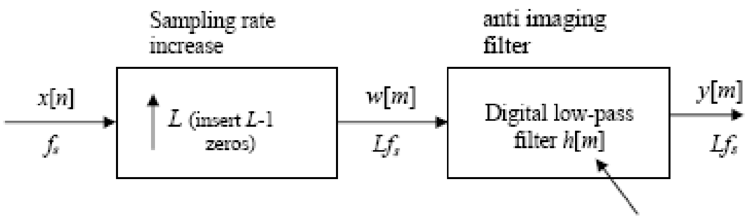
$$x[n]=\{1,2,4,3,-5,6,-7,2,4,3\}$$

$$\uparrow 2$$

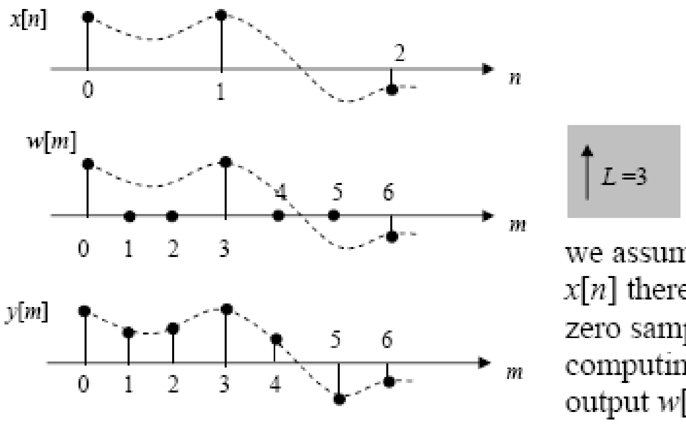
$$y[m]=\{1,0,2,0,4,0,3,0,-5,0,6,0,-7,0,2,0,4,0,3,0\}$$



We observe that to go from $X(\theta)$ to $Y(\theta)$, we have to pass $x[n]$ through a low-pass digital filter designated at the Lf_s sampling rate that attenuates sufficiently any frequency components above $f_s/2$. The time-domain interpretation of the low-pass filter is that it interpolates the samples of $w[m]$ to produce the waveform as if $x(t)$ had been sampled at Lf_s.

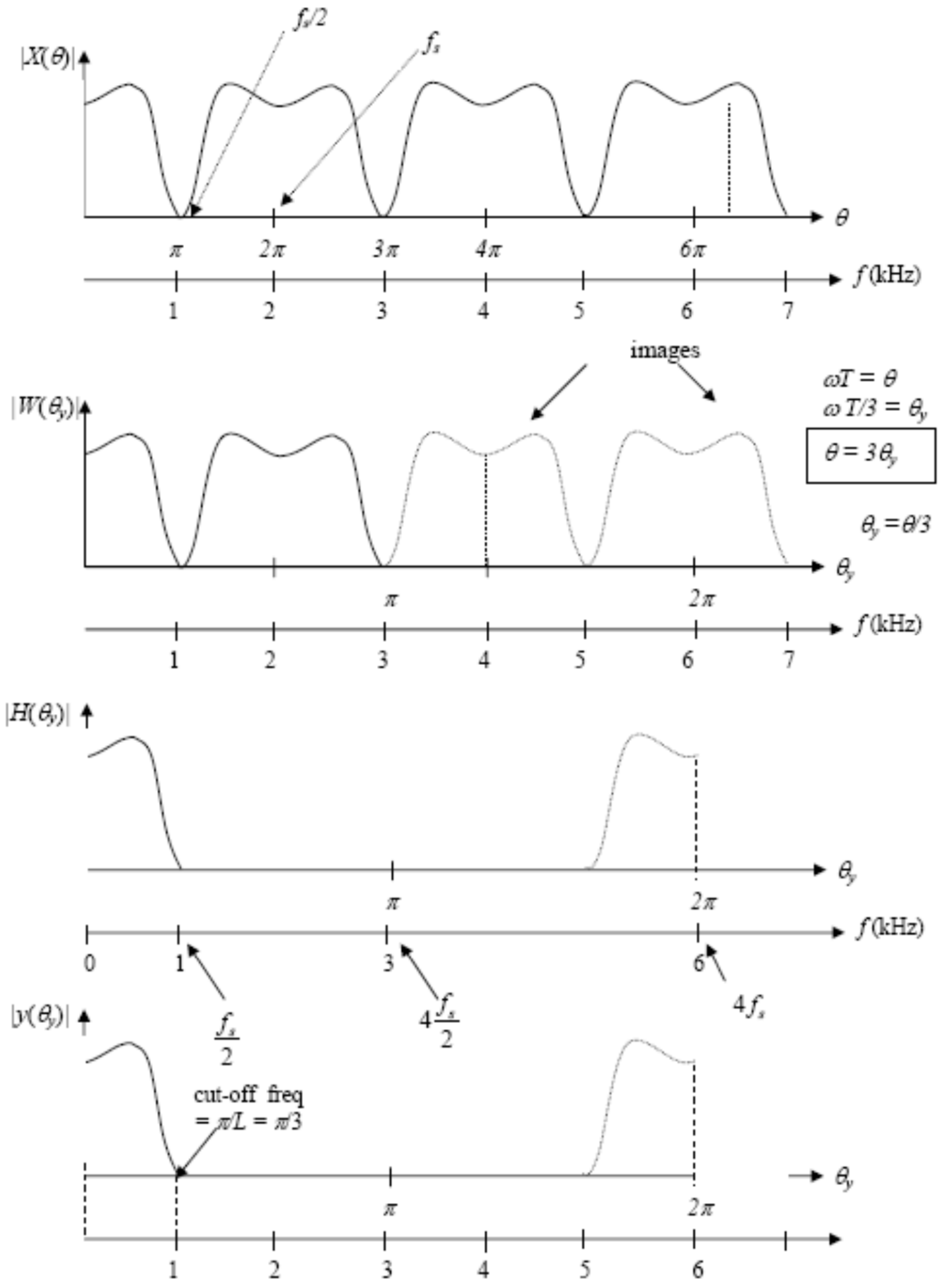


The LPF joins all the samples of $w[m]$ to produce a waveform as if $x[n]$ had been sampled at Lf_s.



we assume that $x[n]$ there are N-1 zero samples when computing an output $w[n]$.

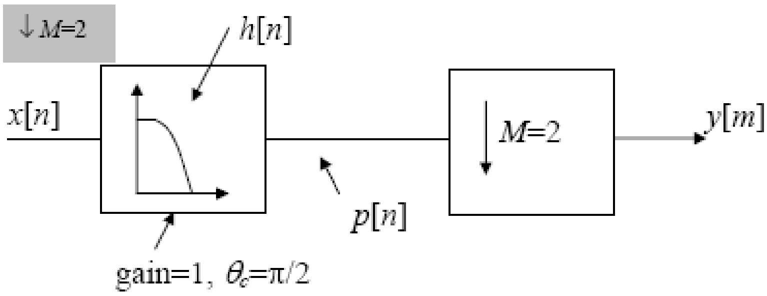
The frequency domain representation of the process is given below



Note: Obviously the same reasoning that led us to believe that FIR filters are preferable in the decimation holds here also.

Example 8.4

Downsampling $x[n] = \{2,6,4,2,6,8,4,2,4,4\}$ by a factor of 2.

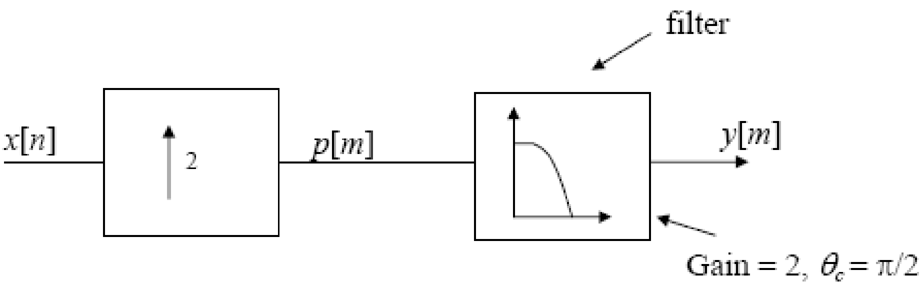


$h[0]$=1/2, $h[1]$=1/2 $\qquad$ $h[n]$= {1/2, 1/2}

$$p[n] = x[n] * h[n] = \{4,5,3,4,7,6,3,3,4,2\}$$
$$y[n] = \{4,3,7,3,4\}$$

8.3.1 Linear Interpolation

Linear interpolation of $x[n] = \{1,3,5,3,7\}$ by a factor of 2.



$$p[n] = \{1,0,3,0,5,0,3,0,7,0\}, \qquad (insert\ zeros)$$
$$h[n] = \left\{\frac{1}{2}, 1, \frac{1}{2}\right\}$$
$$y[n] = p[n] * h[n] = \{1,2,3,4,5,4,3,5,7,3,5\}$$

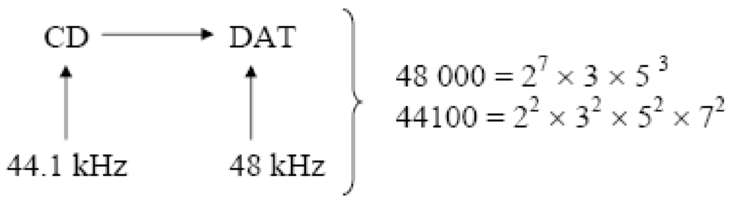
8.4 Rate Conversion by Non-integer Factor

In some applications, the need often arises to change the sampling rate by a non-integer factor. An example is in digital audio applications where it may be necessary to transfer data from one storage system to another, where both systems employ different rates.

An example is transferring data from the compact disk system at a rate of 44.1 kHz to a Digital Audio Tape (DAT) at 48 kHz. This can be achieved by increasing the data rate of the CD by a factor of $\frac{48}{44.1}$, a non-integer.

In practice, such a non-integer factor is represented by a rational number that is a ratio of two integers, say L and M. The sampling frequency change is thus achieved by first interpolating the data by L and then decimating by M.

Note: It is necessary that the interpolation process precedes the decimation, otherwise the decimation process would remove some of the desired frequency components, specifically all frequency components between $f_s/2M$ and $f_s/2$ in the input signal. This is almost always unacceptable.

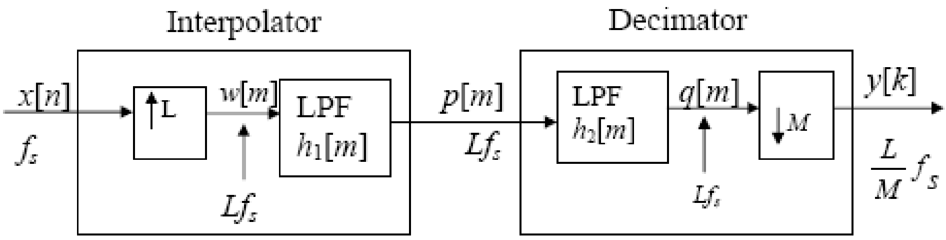


To change the sampling rate we require:

$$\frac{L}{M} = \frac{2^7 \times 3 \times 5^3}{2^2 \times 3^2 \times 5^2 \times 7^2} = \frac{160}{147}$$

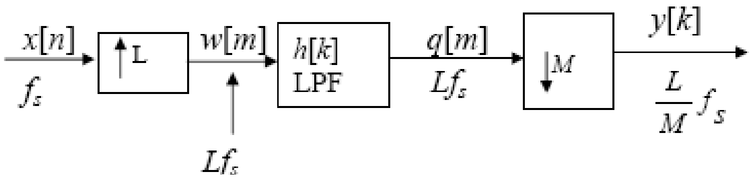
Therefore if we up-sample by L = 160 and then down-sample by M = 147, we achieve the desired sampling rate conversion by a non-integer factor.

The following figure shows the sampling frequency change being achieved by first interpolating the data by L and then decimating by M.



The two low-pass filters $h_1[m]$ and $h_2[m]$ can be combined into a single filter since they are in cascade and have a common sampling frequency.

- If $M > L$, the resulting combined operation is effectively a decimation process by a non-integer.
- If $M < L$, it is effectively an interpolation process by a non-integer.

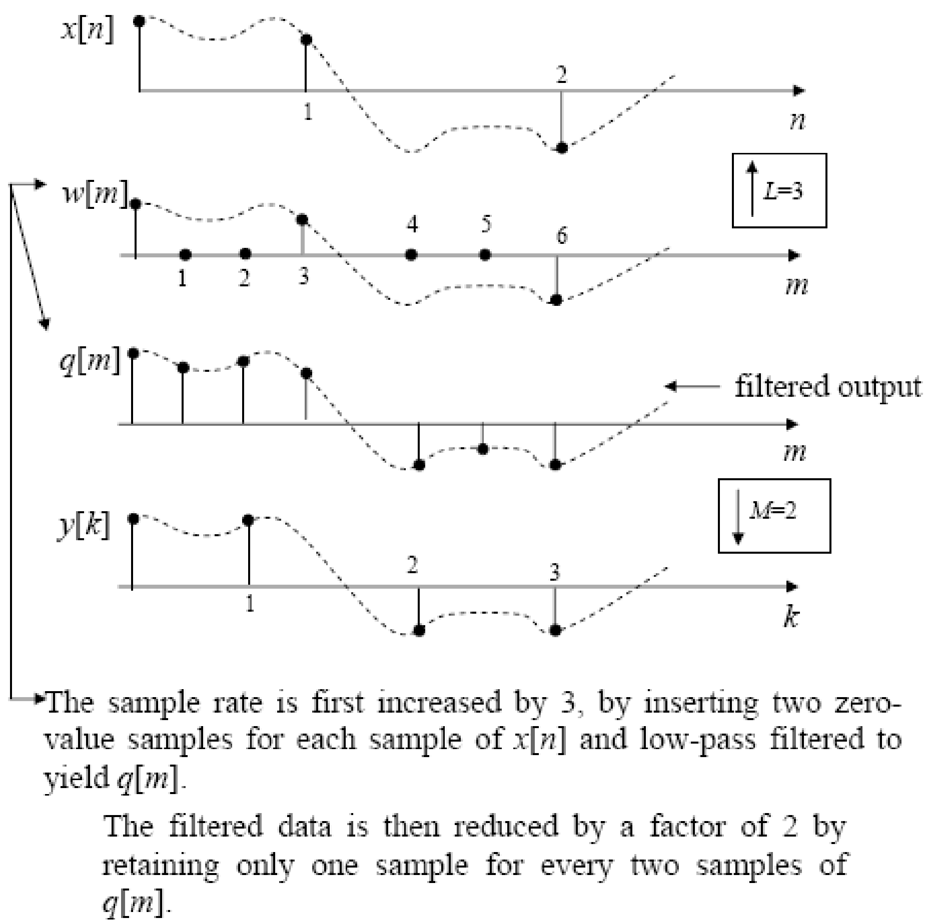


The low-pass filter $h[m]$ that we require is the one that has a cut-off frequency:

$$\theta_c = \min\left(\frac{\pi}{L}, \frac{\pi}{M}\right)$$

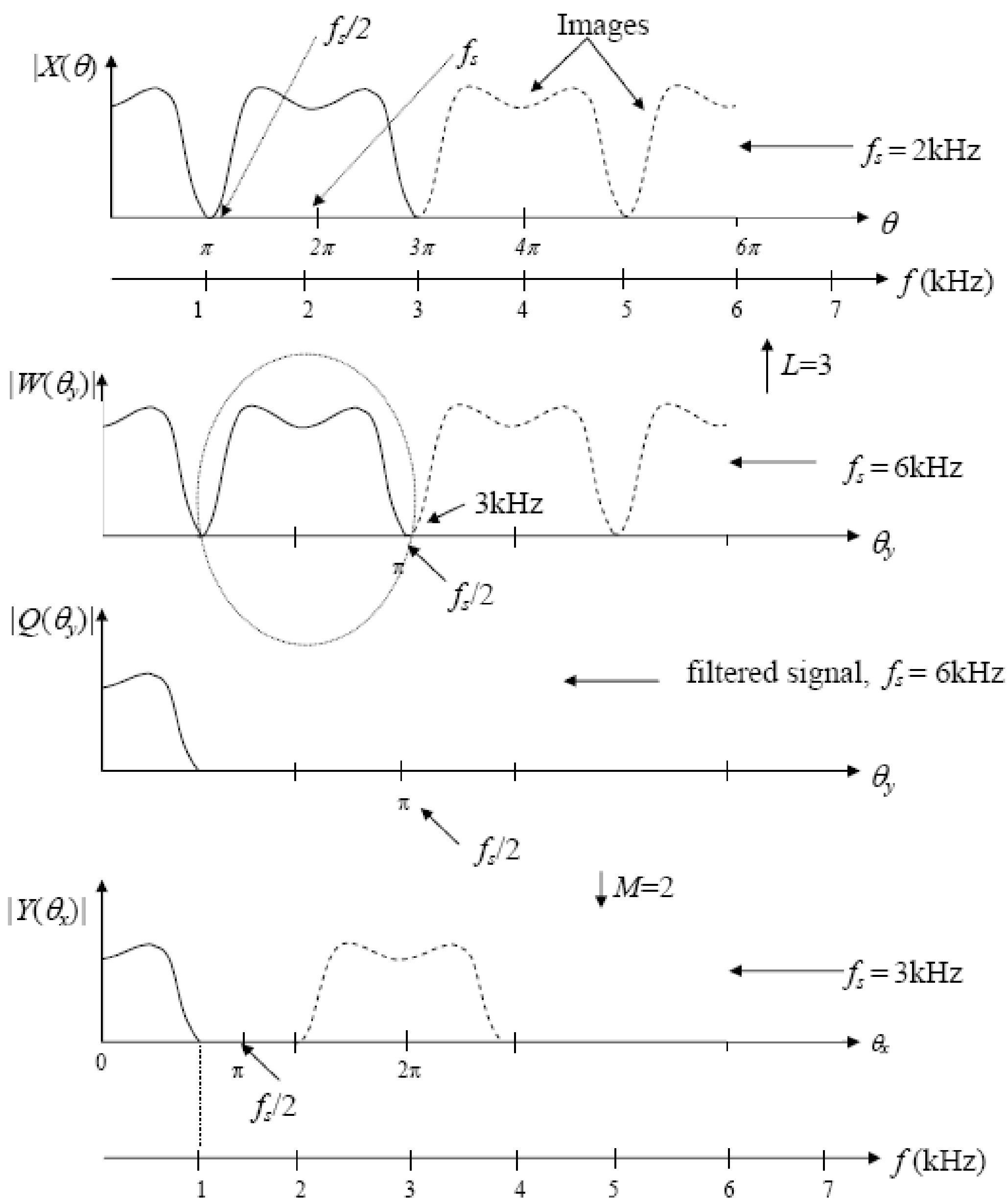
Example 8.5

Illustration of interpolation by a factor of 3/2

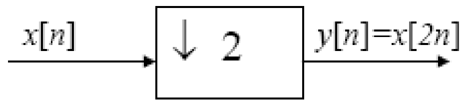


The Figure below illustrates the process of interpolation by 3/2 in the frequency domain. The input signal $x[n]$ sampling rate of 2 kHz is first increased, by a factor of 3, to 6 kHz. It is then filtered to remove the image frequencies which would otherwise cause aliasing distortion, and then reduced by a factor of 2 to 3 kHz.

Frequency domain process of interpolation by 3/2 $\quad \frac{L}{M} = \frac{3}{2}$



8.5 Decimation by 2



$$Y(z) = \frac{1}{2}\left[X(z^{\frac{1}{2}}) + X(-z^{\frac{1}{2}}) \right]$$

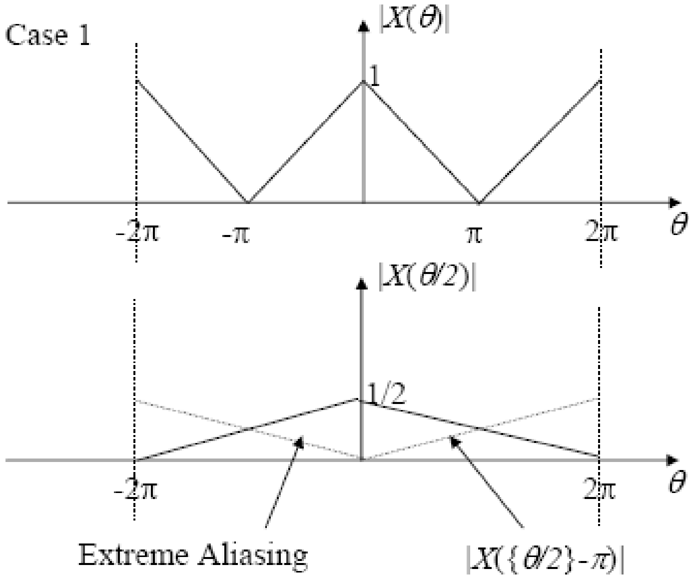
Aliasing term

$$Y(\theta) = \frac{1}{2}\left[X(e^{\frac{j\theta}{2}}) + X(-e^{\frac{j\theta}{2}}) \right] = \frac{1}{2}\left[X(e^{\frac{j\theta}{2}}) + X(e^{j(\frac{\theta}{2}-\pi)}) \right]$$

$$Y(\theta) = \frac{1}{2}\left[X(\frac{\theta}{2}) + X(\frac{\theta}{2} - \pi) \right]$$

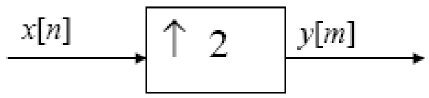
Aliasing term

Stretch $X(\theta)$ by a factor 2 to obtain $X(\theta/2)$

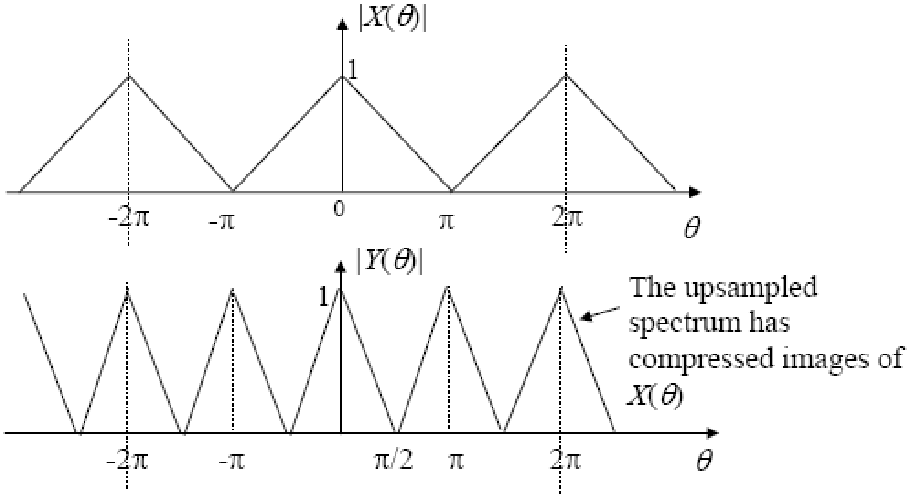


Note: The spectrum is stretched by down sampling.

8.6 Interpolation by 2



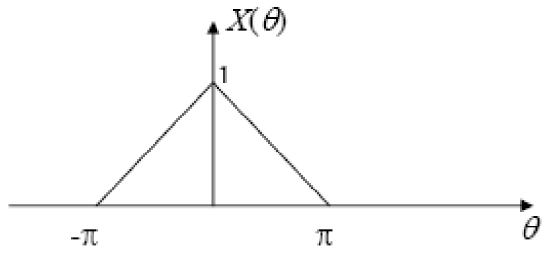
$$Y(z) = \left\lfloor X(z^2) \right\rfloor; \qquad Y(\theta) = \left\lfloor X(e^{j\theta}) \right\rfloor = \left[X(2\theta) \right]$$

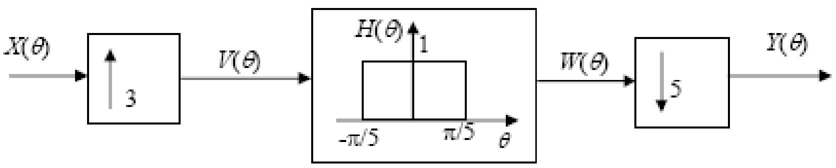


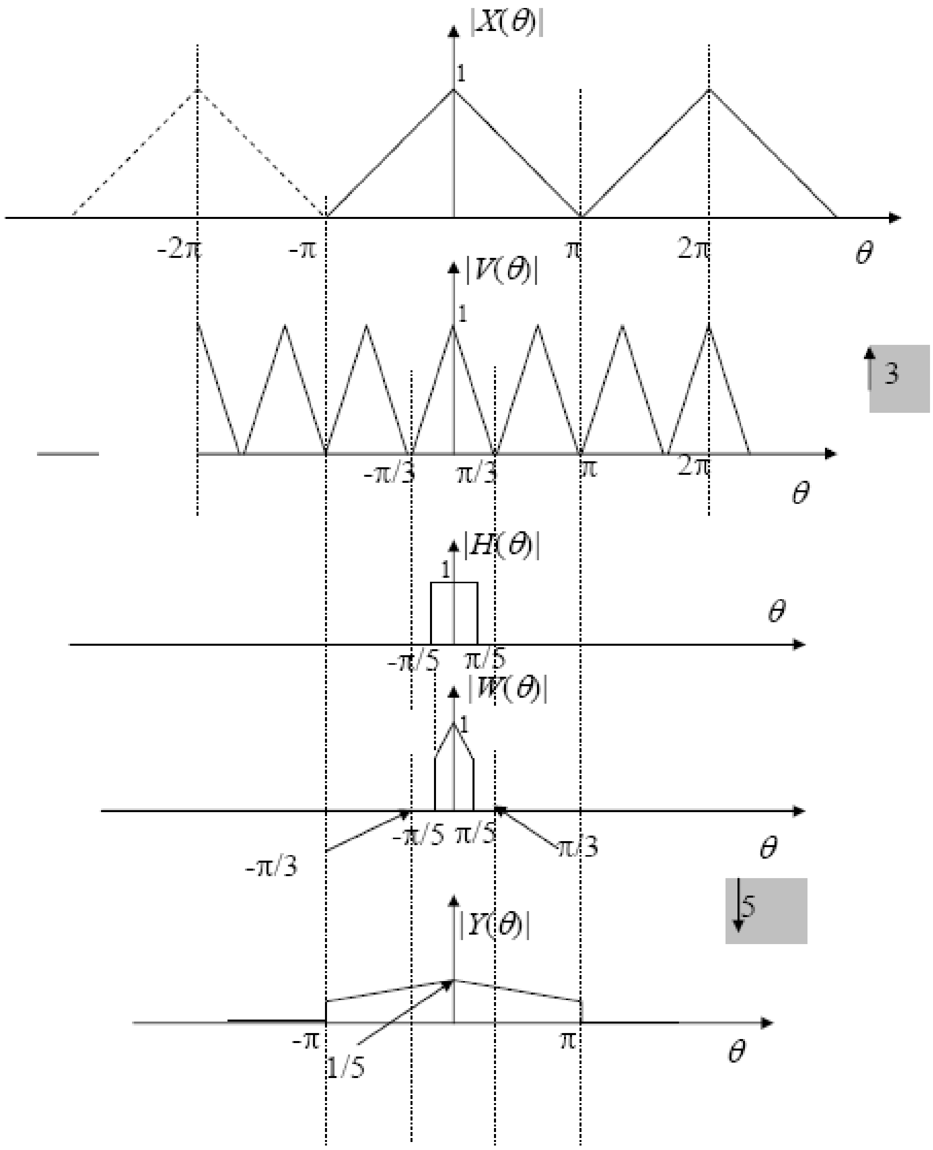
Note: The spectrum is compressed by upsampling.

<u>Example 8.6</u>

An input signal $x[n]$ with the spectrum $X(\theta)$ is applied to the system shown below. Sketch, one above another, $|X(\theta)|$, $|V(\theta)|$, $|W(\theta)|$, and $|Y(\theta)|$ against θ.



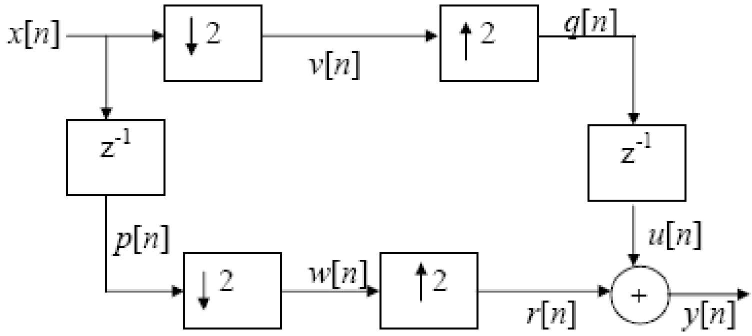




|X(θ)|
1
-2π
-π
π
2π
θ
|V(θ)|
1
3
-π/3
π/3
π
2π
θ
|H(θ)|
1
θ
-π/5
π/5
|W(θ)|
1
-π/5
π/5
-π/3
π/3
θ
5
|Y(θ)|
-π
π
1/5
θ

Example 8.7

Express the output $y[n]$ of the figure below as a function of the input $x[n]$. By simplifying the expression derived show that $y[n] = x[n-1]$.



$$p[n] = x[n-1]$$
$$\Rightarrow P(z) = z^{-1}X(z)$$
$$\Rightarrow P\left(z^{\frac{1}{2}}\right) = z^{-\frac{1}{2}}X\left(z^{\frac{1}{2}}\right)$$

$$w[n] = p[2n]$$
$$\Rightarrow W(z) = \frac{1}{2}\left[P\left(z^{\frac{1}{2}}\right) + P\left(-z^{\frac{1}{2}}\right)\right]$$
$$\Rightarrow W(z) = \frac{1}{2}\left[z^{-\frac{1}{2}}X\left(z^{\frac{1}{2}}\right) - z^{-\frac{1}{2}}X\left(-z^{\frac{1}{2}}\right)\right]$$

$$v[n] = x[2n]$$
$$\Rightarrow V(z) = \frac{1}{2}\left[X\left(z^{\frac{1}{2}}\right) + X\left(-z^{\frac{1}{2}}\right)\right]$$

$$q[n] = v\left[\frac{n}{2}\right]$$
$$\Rightarrow Q(z) = V(z^2)$$
$$U(z) = Q(z)z^{-1}$$

Similarly,

$$R(z) = W(z^2)$$
$$Q(z) = V(z^2) = \frac{1}{2}[X(z) + X(-z)]$$

$$R(z) = W(z^2) = \frac{z^{-1}}{2}X(z) - \frac{z^{-1}}{2}X(-z)$$

$$U(z) = z^{-1}Q(z) = \frac{1}{2}z^{-1}[X(z) + X(-z)]$$

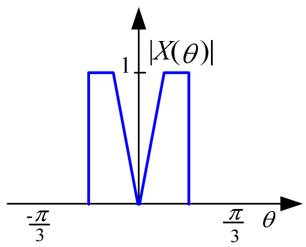
$$\begin{aligned} Y(z) &= U(z) + R(z) \\ &= \frac{z^{-1}}{2}X(z) - \frac{z^{-1}}{2}X(-z) + \frac{1}{2}z^{-1}[X(z) + X(-z)] \\ &= z^{-1}X(z) \end{aligned}$$

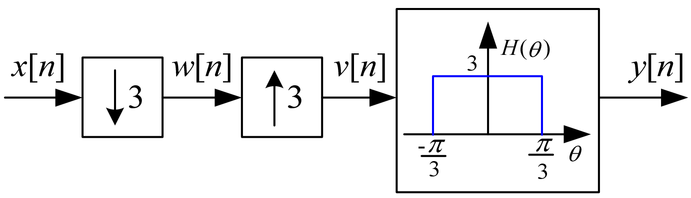
$$\Rightarrow y[n] = x[n-1]$$

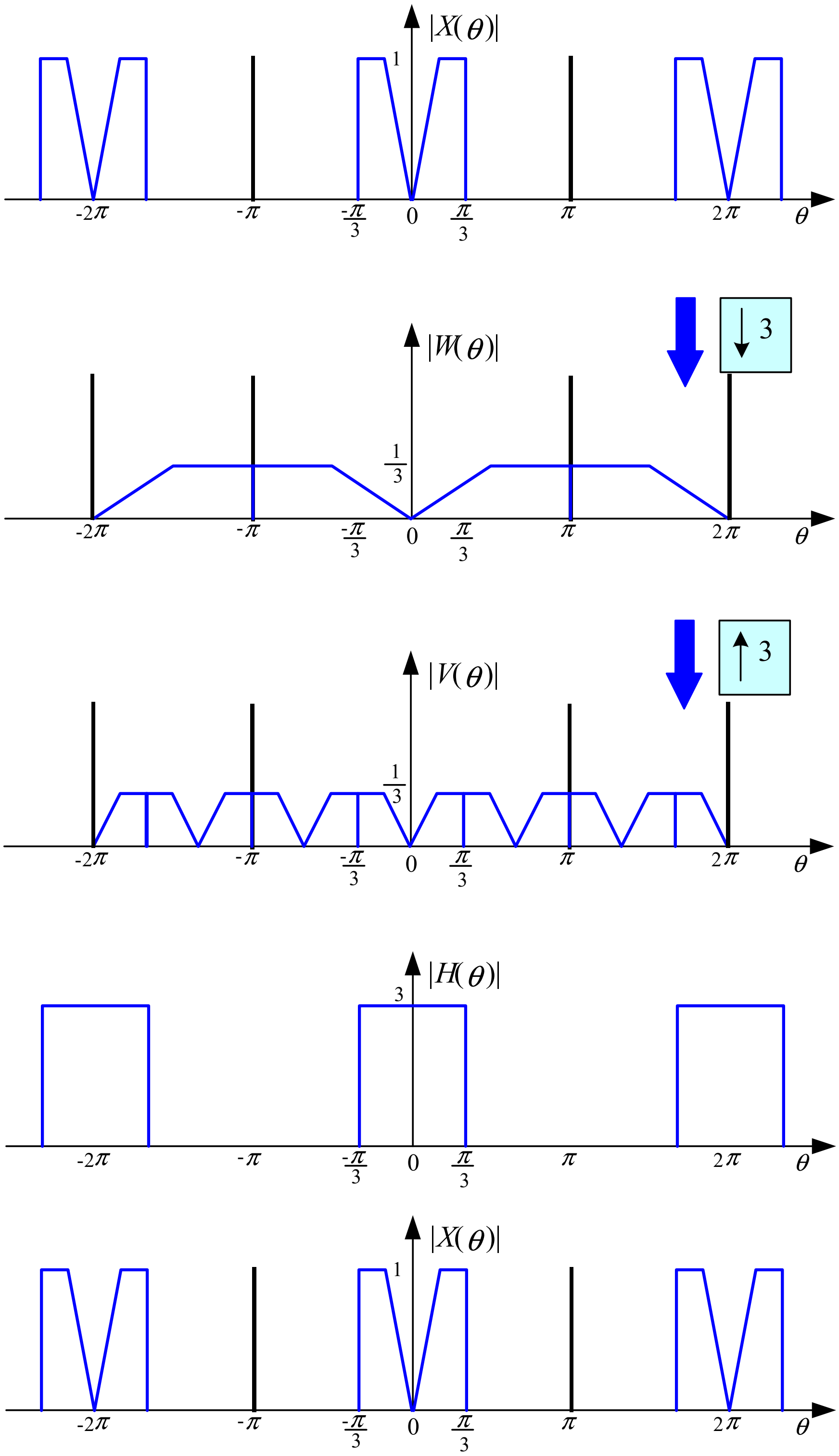
The original signal $x[n]$ is reconstructed. The above structure is a perfect reconstruction system. During sample rate conversion, interpolation should precede decimation, as stated earlier in the section. In this example, downsampling was able to precede upsampling because $v[n]$ and $w[n]$ are simply the even and odd samples of $x[n]$, so no information is lost from $x[n]$.

Example 8.8

An input signal $x[n]$ with the spectrum $X(\theta)$ is applied to the system shown below. Sketch, one above another, $|X(\theta)|$, $|V(\theta)|$, $|W(\theta)|$, and $|Y(\theta)|$ against θ.



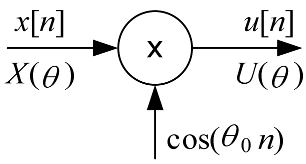



|X(θ)|
1
-2π
-π
-π/3
0
π/3
π
2π
θ
|W(θ)|
↓3
1/3
|V(θ)|
↑3
|H(θ)|
3

Note: During sample rate conversion, interpolation should precede decimation, as stated earlier in the section. In this example, decimation was able to precede interpolation only because $x[n]$ originally had a bandwidth of $\frac{f_s}{2M}$ or $\frac{\pi}{3}$.

8.7 Modulation

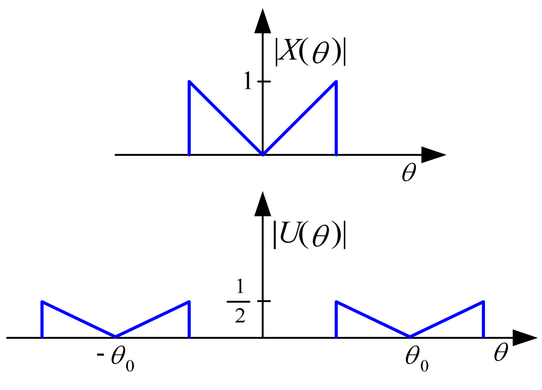
In the time domain, modulation is the process of multiplying an input signal $x[n]$ with a sinusoidal signal known as the carrier, and is illustrated below.



$$u[n] = x[n] \cos \theta_0 n$$

Taking the Fourier transform,

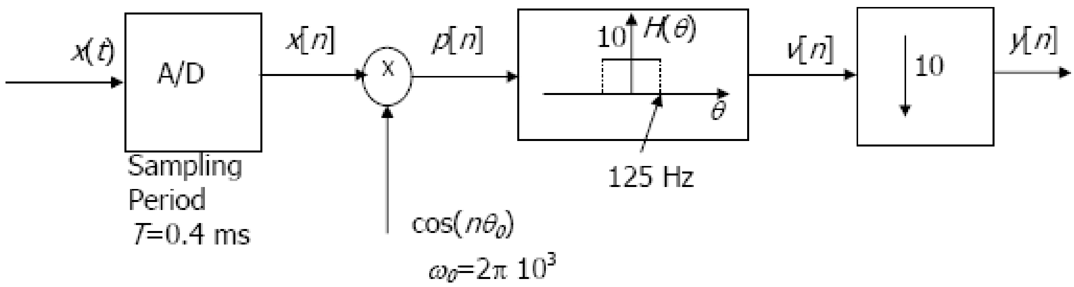
$$U(\theta) = \frac{1}{2} X(\theta + \theta_0) + \frac{1}{2} X(\theta - \theta_0)$$



Example 8.9

$x(t)$ is the input signal for the system shown below. The spectrum of the analogue signal $x(t)$ is given by

$$X(f) = \begin{cases} 1, & 0.9kHz \le f \le 1.1kHz \\ 1, & -1.1kHz \le f \le -0.9kHz \\ 0, & otherwise \end{cases}$$



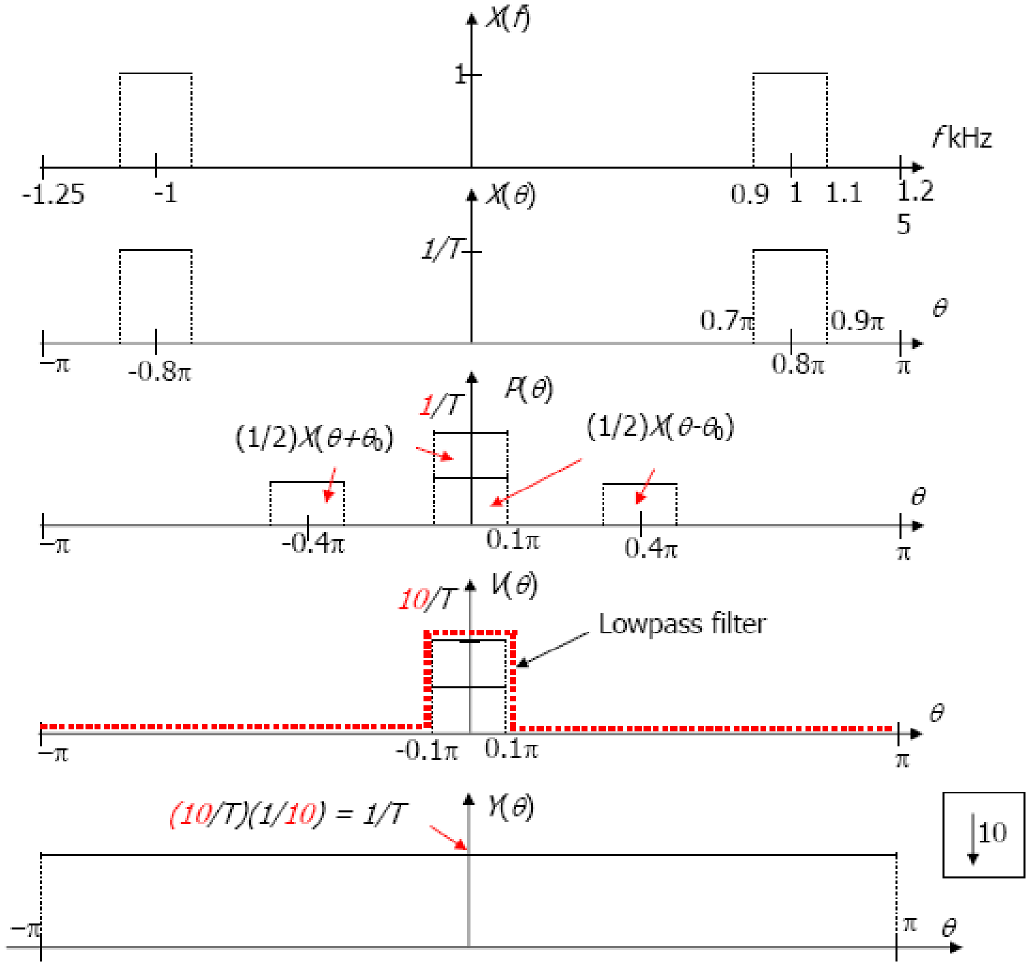
$H(\theta)$ is an ideal low-pass filter (gain=10) with cut-off frequency f_c=250Hz. Sketch, one above another, $|X(\theta)|$, $|P(\theta)|$,$|V(\theta)|$, $|Y(\theta)|$ against θ.

$$P(\theta) = \frac{1}{2}[X(\theta + \theta_0) + X(\theta - \theta_0)]$$

$$f_s = \frac{1}{T} = 2.5kHz$$

$$\theta_0 = 2\pi \times 10^3(0.4 \times 10^{-3}) = 0.8\pi$$

$$\theta_c = 2\pi \times \frac{125}{2.5 \times 10^3} = 0.1\pi$$



X(f)
1
f kHz
-1.25
-1
0.9
1
1.1
1.25
X(θ)
1/T
θ
0.7π
0.9π
−π
-0.8π
0.8π
π
P(θ)
1/T
(1/2)X(θ+θ0)
(1/2)X(θ-θ0)
-0.4π
0.1π
0.4π
V(θ)
10/T
Lowpass filter
-0.1π
0.1π
(10/T)(1/10) = 1/T
Y(θ)
10