Multirate Digital Signal Processing (Optional)

(Final Exam will not cover this Topic)

EE4015 Digital Signal Processing

Dr. Lai-Man Po
Department of Electrical Engineering
City University of Hong Kong

Content

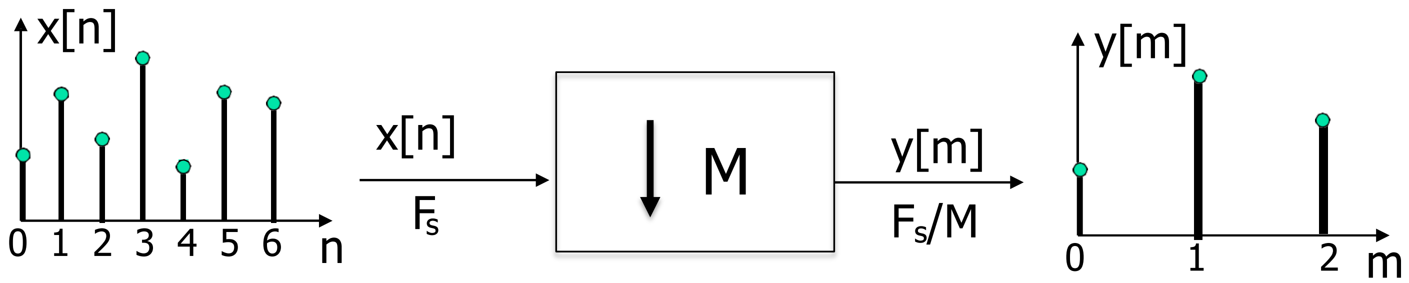
- Decimation : Down-Sampler
- Interpolation : Up-Sampler
- Sampling Rate Conversion by Non-Integer Factors
- Computational Requirement of MDSP
- Modulation

Multirate Digital Signal Processing

- The increasing need in modern digital systems to process data at more than one sampling rate has led the development of a new sub-area in DSP known as multirate digital signal processing (MDSP).
- The two primary operations in MDSP are:
  - **Decimation** (Down Sampling) : decrease the sampling rate F_s of a given signal x[n]
  - **Interpolation** (Up Sampling) : increase the sampling rate F_s of a given signal x[n]

Decimation (Down Sampling)

- Decimation is used to decrease the sampling rate of an input signal.
- The decimation factor is confined to an integer such as M=3



M=3; F_s = sampling frequency

- The output signal y[m] is obtained by taking every Mth sample of the input signal.
- If M=3, we should just take every third sample of x[n] to form the desired signal y[m]

Decimation Example

- x[n] = {1, 2, 4, 3, 5, -6, -8, 2, -3, 2, 6, 8, 9, 7, 5, 2}

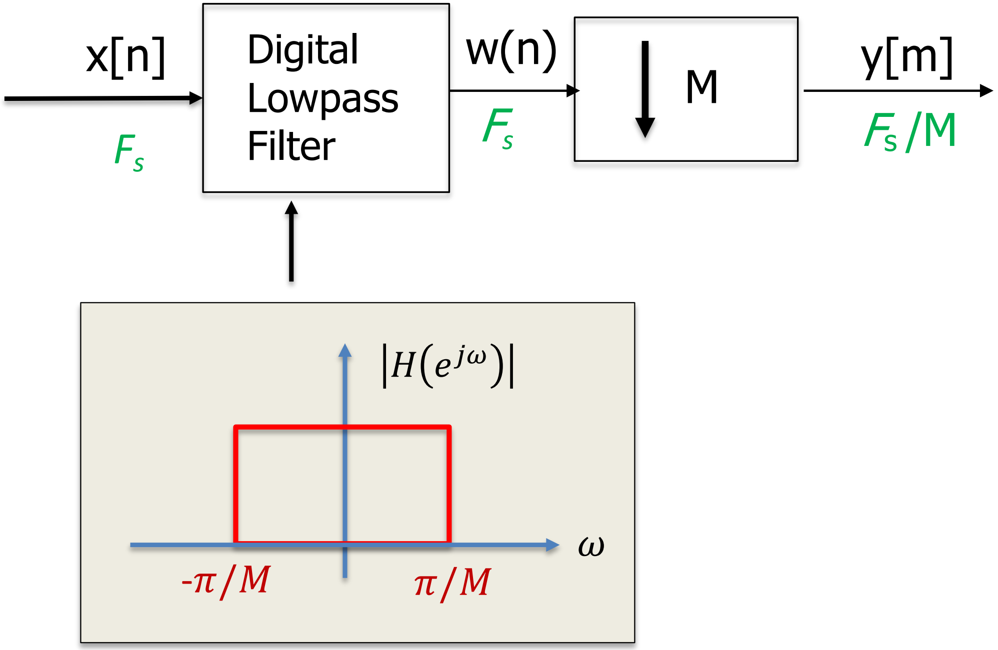
↓ Down sample by 3

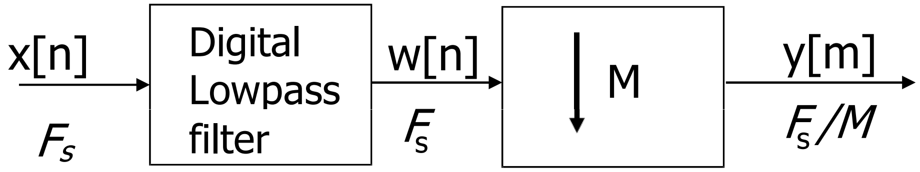
- y[m] = {1, 3, -8, 2, 9, 2}

- The output signal y[n] is obtained by taking every Mth sample of the input signal. If M = 3, we should just take every second sample of x[n] to form the desired signal y[m].
- 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 $F_s/6$ (Half the desired sampling rate)
  - It is because the spectral components above this frequency will be aliased into frequencies below $F_s/6$ according to the sampling rule.

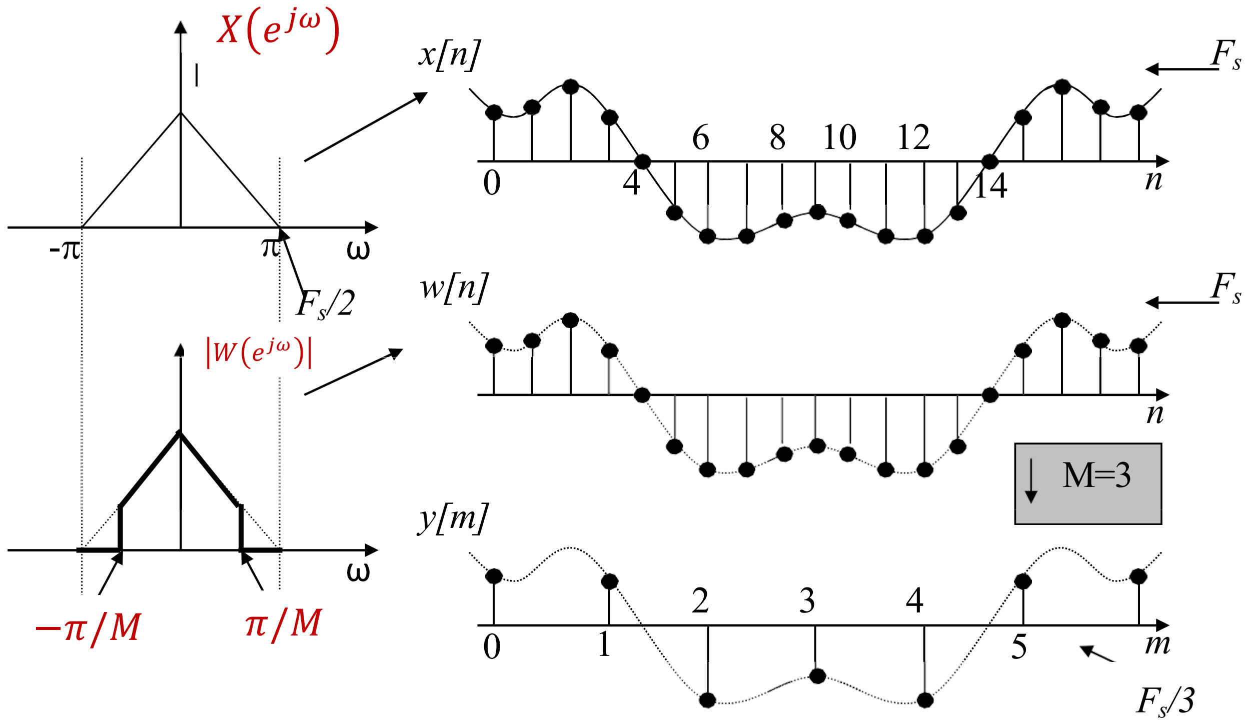
A Times M Decimator Configuration

- The signal x[n] is first passed through a digital lowpass filter that attenuates the band from $|\omega| > \pi/M$ (F_s/2M to F_s/2) to prevent aliasing.

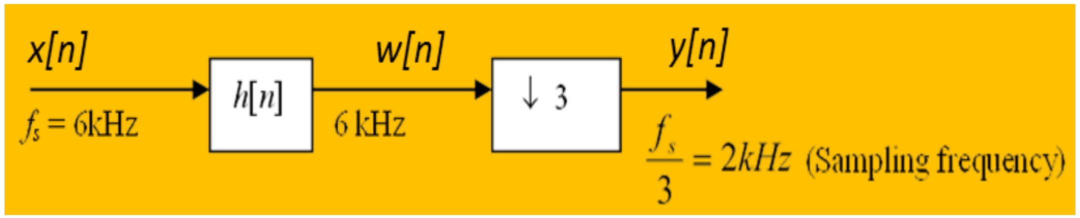




x[n]
F_s
Digital Lowpass filter
w[n]
F_s
M
y[m]
F_s/M

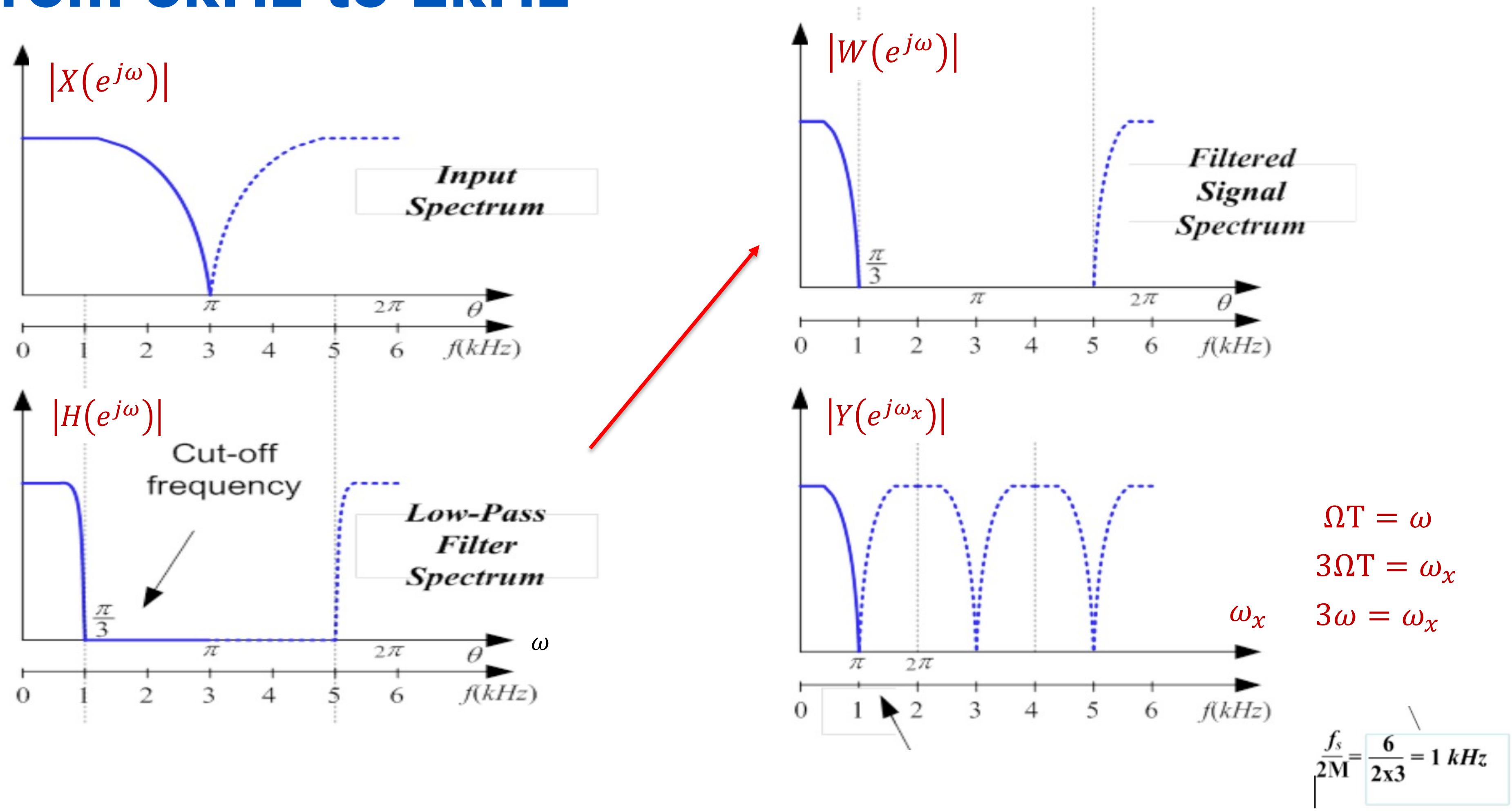


$X(e^{j\omega})$
-π
π
ω
$F_s/2$
$|W(e^{j\omega})|$
ω
$-\pi/M$
π/M
x[n]
0
4
6
8
10
12
14
n
F_s
w[n]
n
F_s
M=3
y[m]
0
1
2
3
4
5
m
$F_s/3$

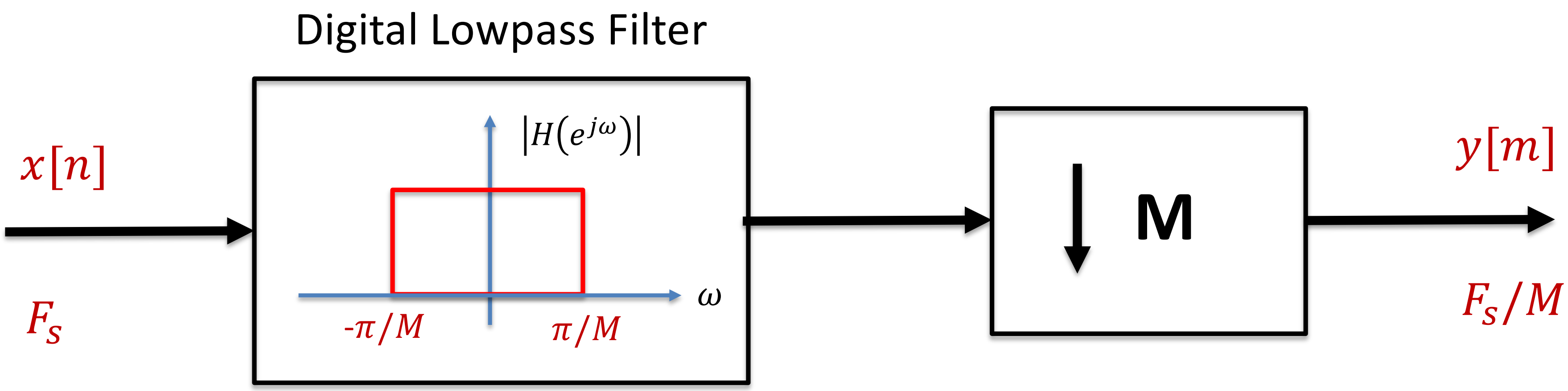


x[n]
f_s = 6kHz
h[n]
w[n]
6 kHz
↓ 3
y[n]
$\frac{f_s}{3} = 2kHz$ (Sampling frequency)

Spectral interpretation of decimation of a signal from 6kHz to 2kHz



Decimator Configuration



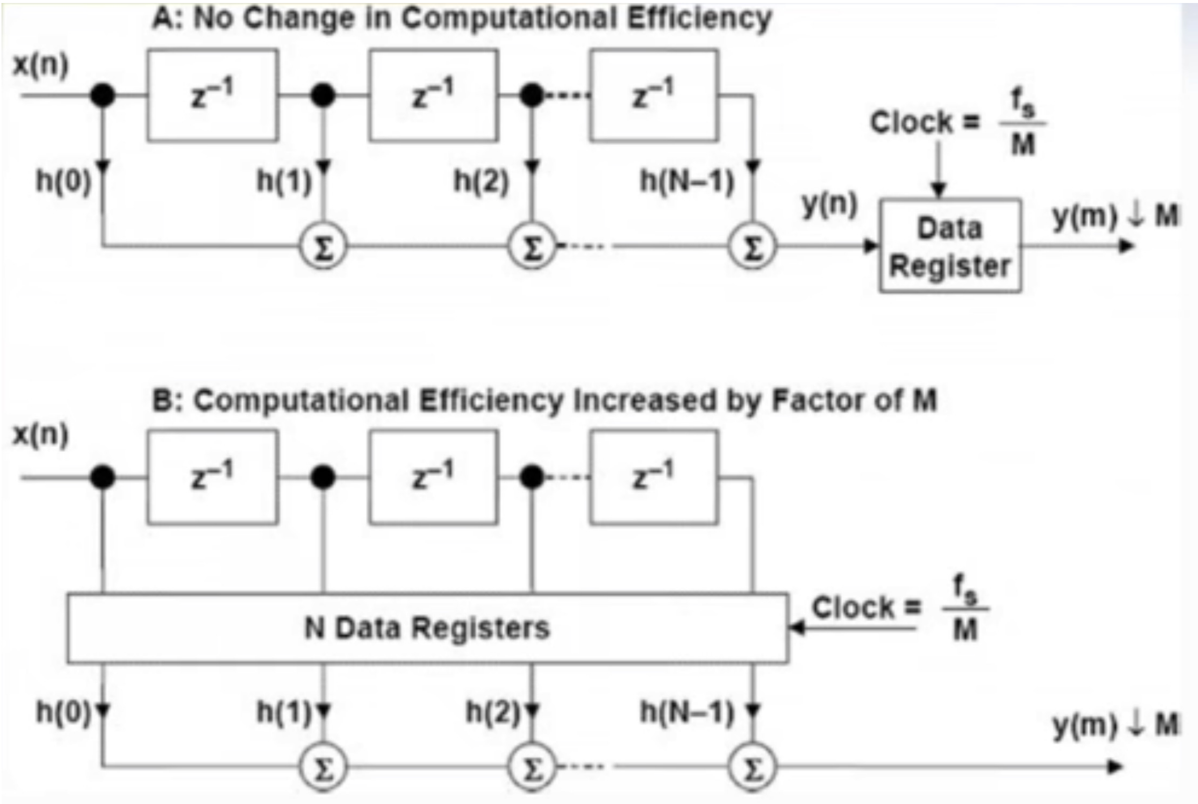
Why Decimation?

- In practice, decimator may require, for example when an audio signal is over sampled 4 times at F_s = 176.4kHz for releasing the analog anti-aliasing lowpass filter requirement.
- In order to match the standard Compact Disc Audio sample rate of 44.1kHz, we need to down sample the digital signal.
- So the first step in the decimation process must be the digital filtering of the signal x[n] is band limited to $F_s/(2\times 4)$ = 22.05kHz

Which Type of Digital Filter (IIR or FIR) should be Used?

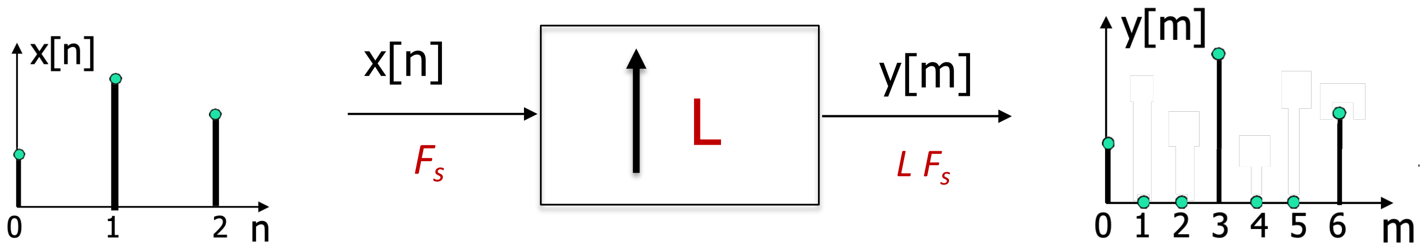
- **IIR filter** has an obvious shortcoming. We cannot take advantage of the fact that we only have to compute every Nth output, since previous outputs are required to compute the Mth output. No saving is realized.
- **FIR filter** implies that we can do the computations at the rate of fs/M. Thus, using an FIR filter in the decimation process will lead to a significantly lower computation rate. 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.

Decimation of FIR Filtering improve Efficiency



Interpolation (Up Sampling)

- The process of interpolation involves a sampling rate increase such as L=3



- The sequence x[n] was derived by sampling x(t) at a sampling rate F_s and we want to obtain a sequence y[n] that approximates as closely as possible the sequence that would have been obtained had we sampled x(t) at the rate $L\ F_s$.
- Interpolation involves inserting (L-1) zero samples between samples x[n] and x[n-1]. (Zero Insertion)

Interpolation Examples by Zeros Insertion

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

$$\uparrow 2$$

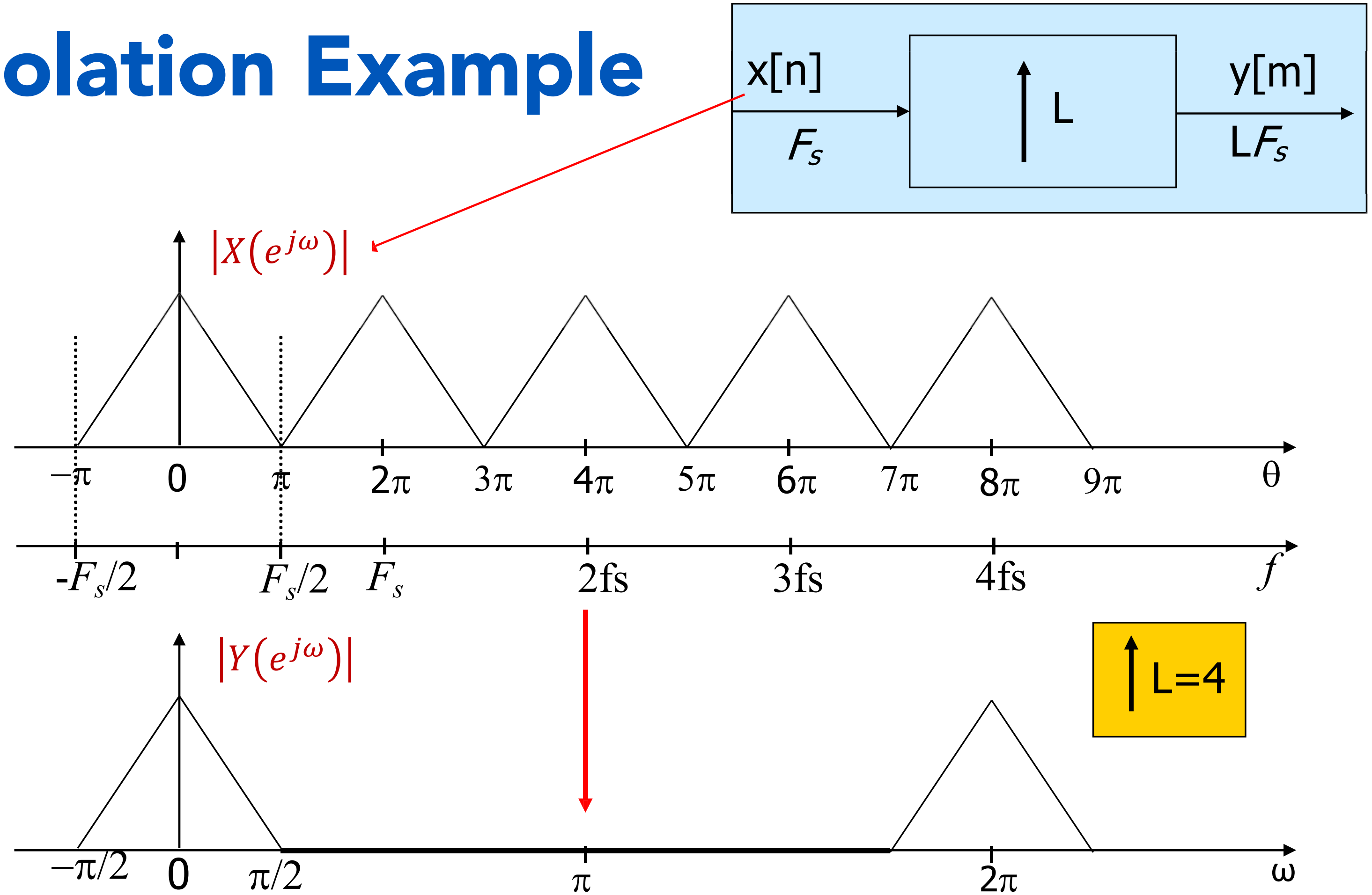
$$y[m]=\{1,0,2,0,4,0,3,0,-5,0,6,0,-7,0,2,0,4,0,3,0\}$$

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

$$\uparrow 3$$

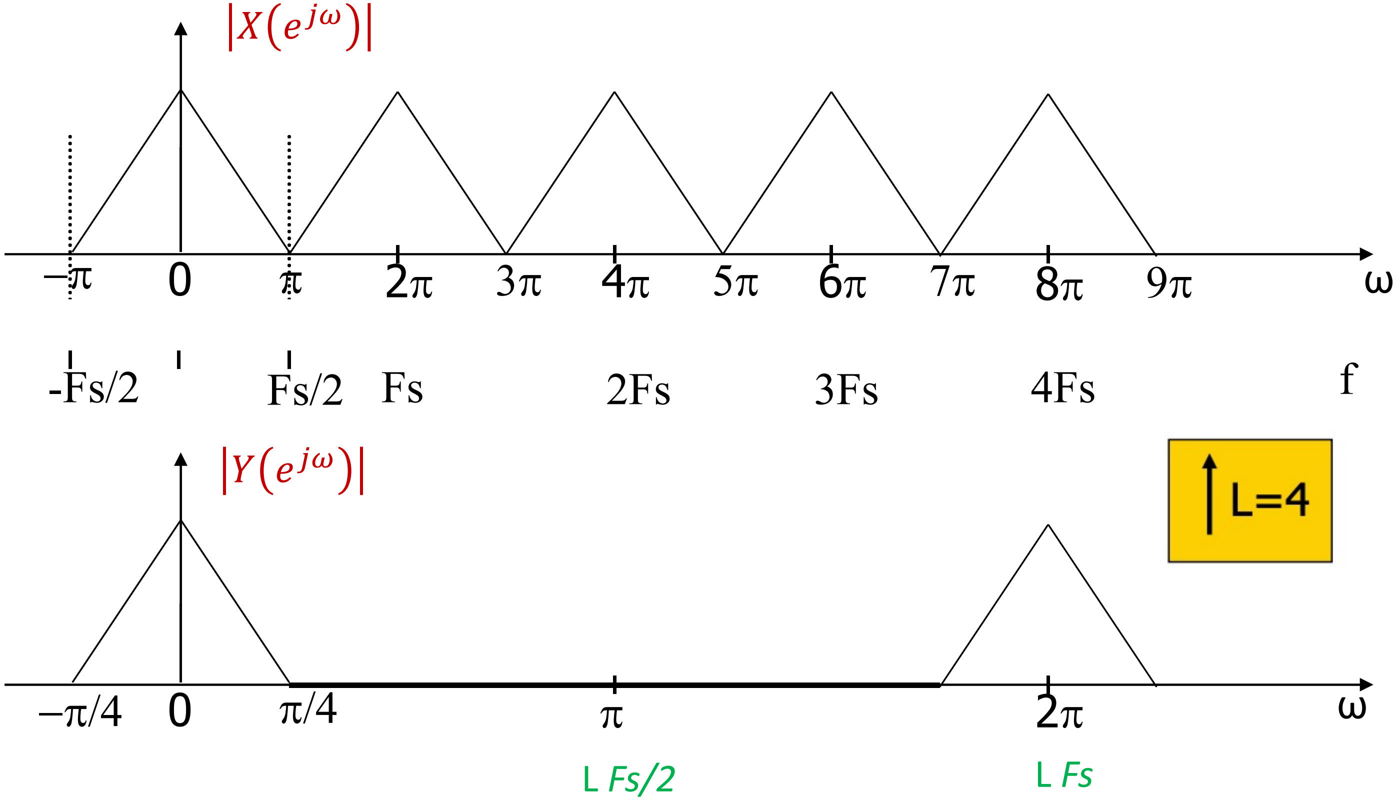
$$y[m]=\{1,0,0,2,0,0,4,0,0,3,0,0,-5,0,0,6,0,0,-7,0,0,2,0,0,4,0,0,3,0,0\}$$

Interpolation Example



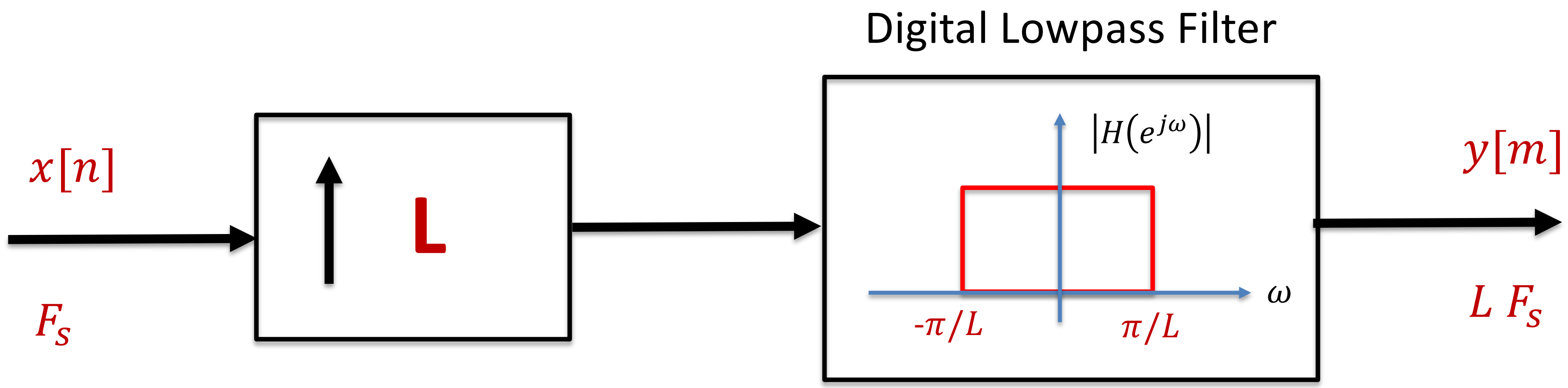
Sampling frequency of y[m] = $4F_s$; Signals must be band limited to $2F_s$

- We observe that to go from $X(e^{j\omega})$ to $Y(e^{j\omega})$, we have to pass x[n] through a lowpass digital filter designed at the $L\ F_s$ sampling rate that attenuates sufficiently any frequency components above $F_s/2$.

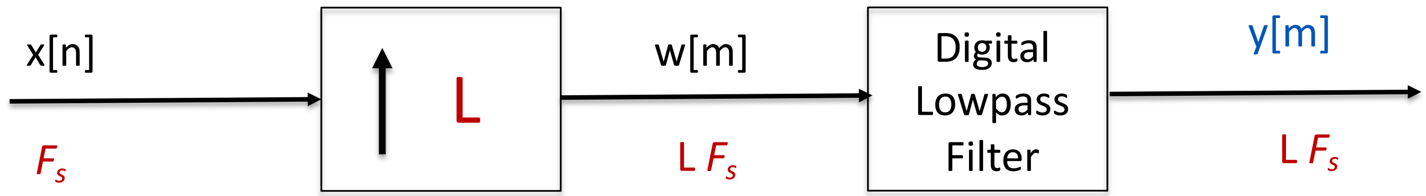


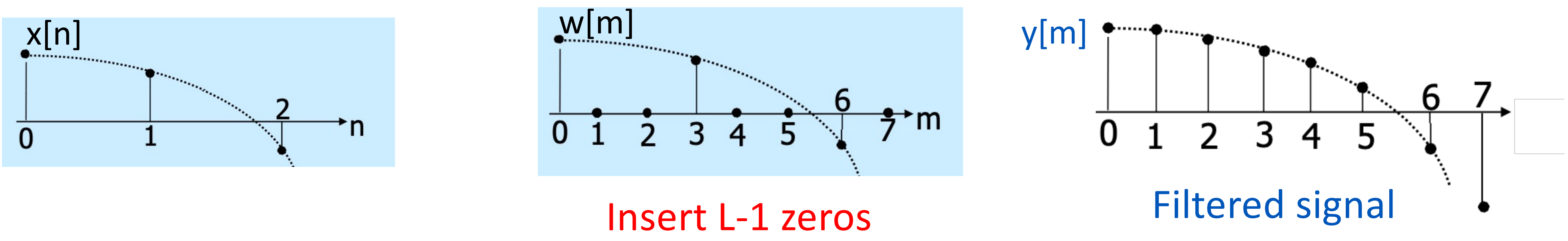
Interpolator Configuration

- To recover the original signal, the upsampled sequence is required to pass through a digital lowpass filter that attenuates the band from $|\omega| > \pi/L$.



xL Interpolator Configuration



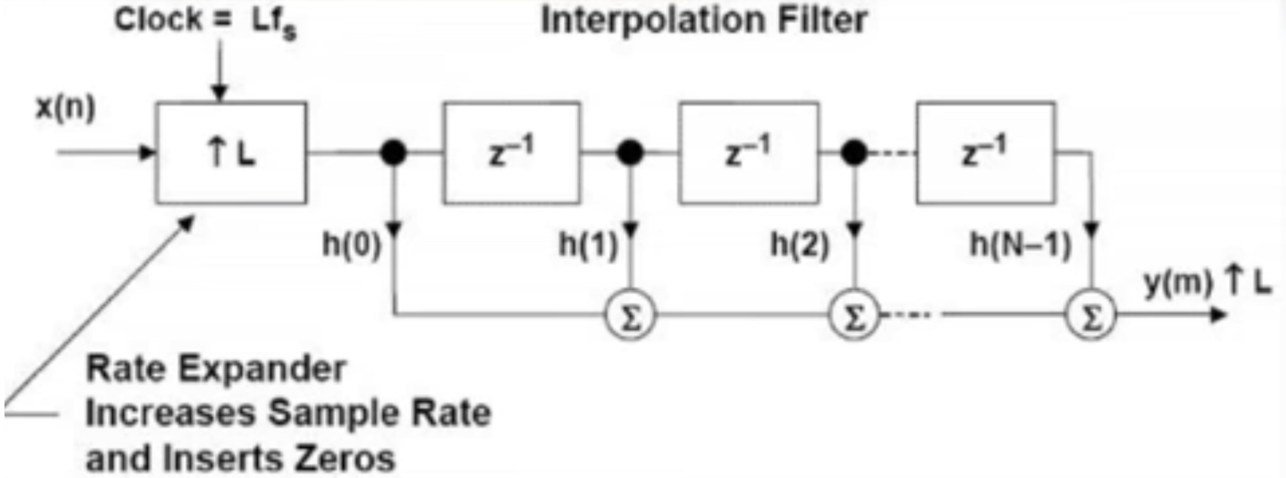


- Example: x[n] = {1, 0.9, -0.5}, Let L = 3, then, w[m] = {1, 0, 0, 0.9, 0, 0, -0.5, 0, 0}
- The digital lowpass filter joins all the samples of w[m] to produce a waveform as if x[n] has been sampled at $L\ F_s$

Interpolator Characteristics

- We assume that behind each x[n], there are L-1 zero samples when we computing an output w[n]
- Note that for each sample of x[n], three output samples y[n] are obtained
- Obviously, the same reasoning that led us to believe that FIR filters are preferable in the decimation process holds here also.

Typical Interpolation Implementation

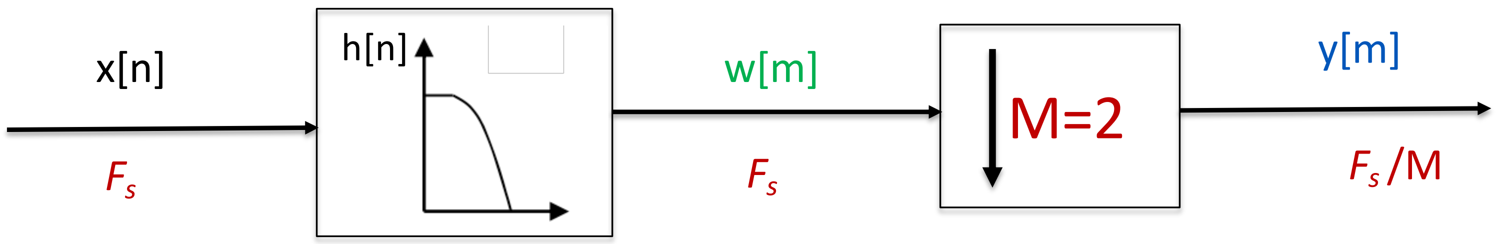


Efficient DSP algorithms take advantage of:

- Multiplications by zero
- Circular Buffers
- Zero-Overhead Looping

Example of x2 Decimator Design

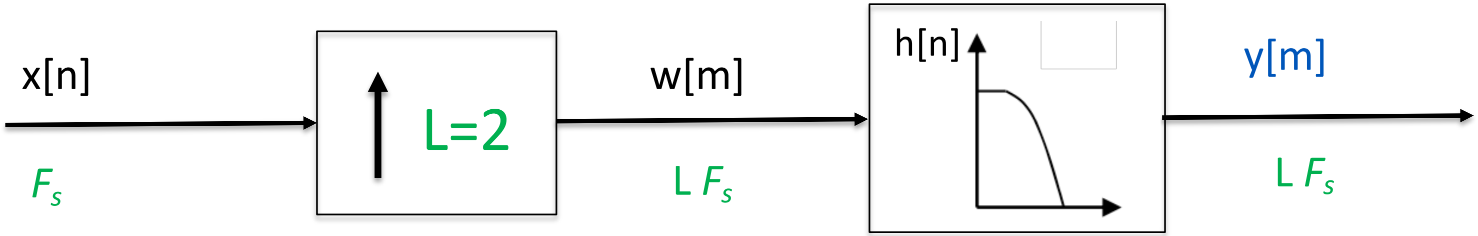
- Decimation of x[n] = {2, 6, 4, 2, 6, 8, 4, 2, 4, 4} with M=2



- Digital Lowpass Filter with impulse response of h[n] = {1/2, 1/2}
- w[m] = x[n]*h[n] = {4, 5, 3, 4, 7, 6, 3, 3, 4, 2}
- y[m] = {4, 3, 7, 3, 4}

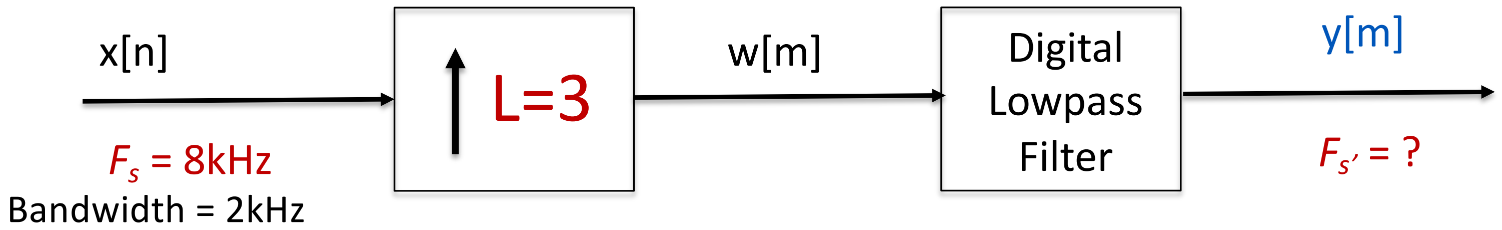
Example of x2 Interpolator Design

- Interpolation of x[n] = {1, 3, 5, 3, 7} with L=2



- Digital Lowpass Filter with impulse response of h[n] = {1/2, 1, 1/2}
- w[m] = {1, 0, 3, 0, 5, 0, 3, 0, 7, 0}
- y[m] = w[m]*h[m] = {1, 2, 3, 4, 5, 4, 3, 5, 7, 3.5}

Interpolator Design Example



- What should be the sample rate of the output signal y[m]?
  - $F_{s'}$ = 3x8 = 24kHz
- What should be the cut-off frequency of the digital lowpass filter?
  - The cut-off frequency should be $\omega_c = \pi/3$, which corresponding to $F_{s'}/6$ = 24k/6 = 4kHz

Sampling Rate Conversion by Non-Integer Factors

- In some applications, the need often arises to change the sampling rate by a non-integer factor
  - An example is transferring data from the compact disk (CD) system at a rate of 44.1kHz 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 48/44.1, a non-integer
  - In practice, such a non-integer factor is represented by a rational number, that is a ration of two integers say L and M
  - The sampling frequency change is then achieved by first interpolating the data by L and then decimating by M

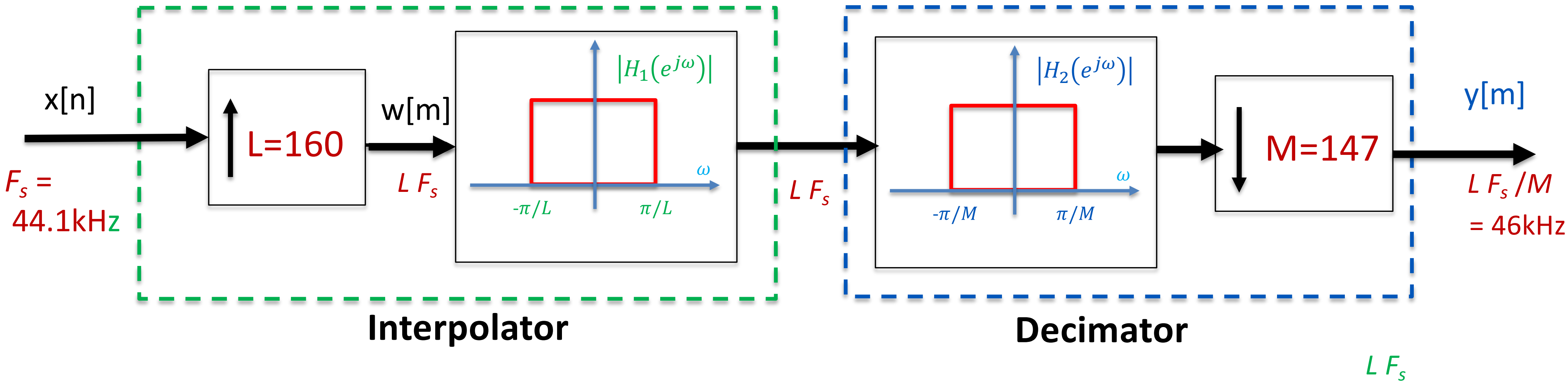
Sampling Rate Conversion of 44.1kHz to 48kHz

- The interpolation process must be performed before decimation, otherwise the decimation process will remove some of the desired frequency components
- CD at 44.1kHz => DAT at 48kHz, which can be converted by

$$\frac{L}{M} = \frac{48000}{44100} = \frac{2^7 \cdot 3 \cdot 5^3}{2^2 \cdot 3^2 \cdot 5^2 \cdot 7^2} = \frac{160}{147}$$

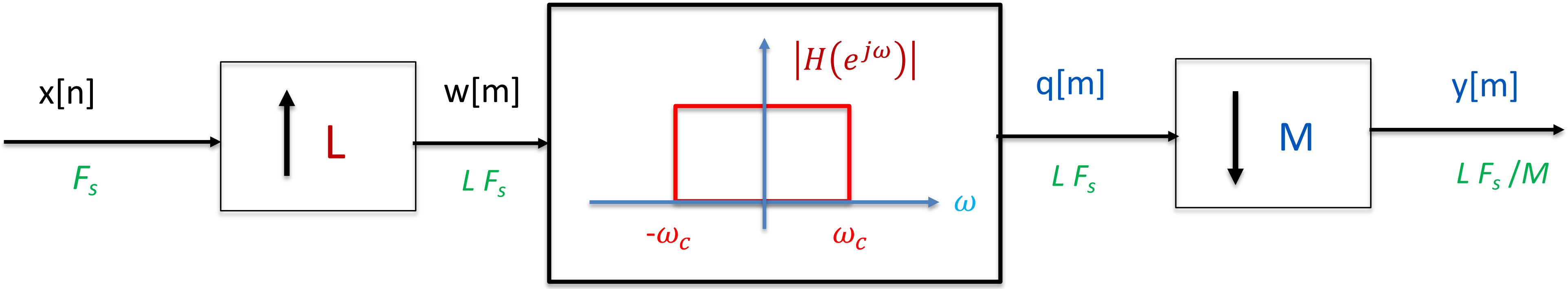
- Therefore if we up sample by L=160 and then down sample by M=147, we can achieve the desired sample rate conversion.

Interpolator and Decimator Configuration



- The two Digital Lowpass Filters, $H_1(e^{j\omega})$ and $H_2(e^{j\omega})$ can be combined into a single filter since they are in cascade and have a common sampling frequency
  - If M > L, then the resulting operation is a decimation process by a non-integer
  - If M < L, then the resulting operation is an interpolation

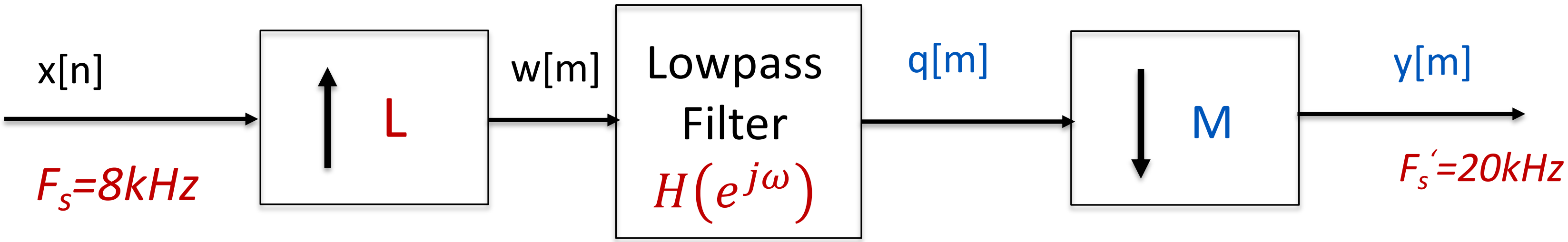
Sampling Rate Conversion by Non-Integer Factors



- The two Digital Lowpass Filters, $H_1(e^{j\omega})$ and $H_2(e^{j\omega})$ can be combined into a single lowpass filter $H(e^{j\omega})$ with cut-off frequency ω_c :
  - $\omega_c = \min\left(\frac{\pi}{L}, \frac{\pi}{M}\right)$

Sampling Rate Conversion Example

- Figure below shows sampling rate conversion by non-integer factors.
- Calculate the values of *L* and *M* as well as the cut-off frequency of the digital lowpass filter.



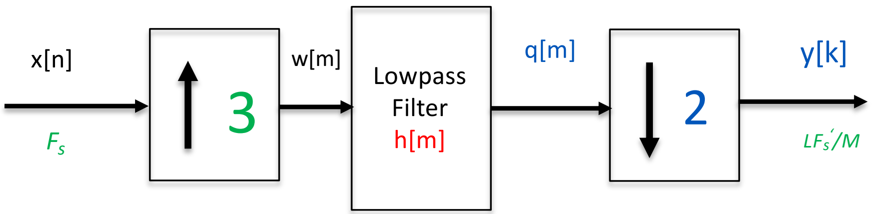
$$\frac{L}{M} = \frac{20}{8} = \frac{2^2 \cdot 5}{2^3} = \frac{5}{2}$$

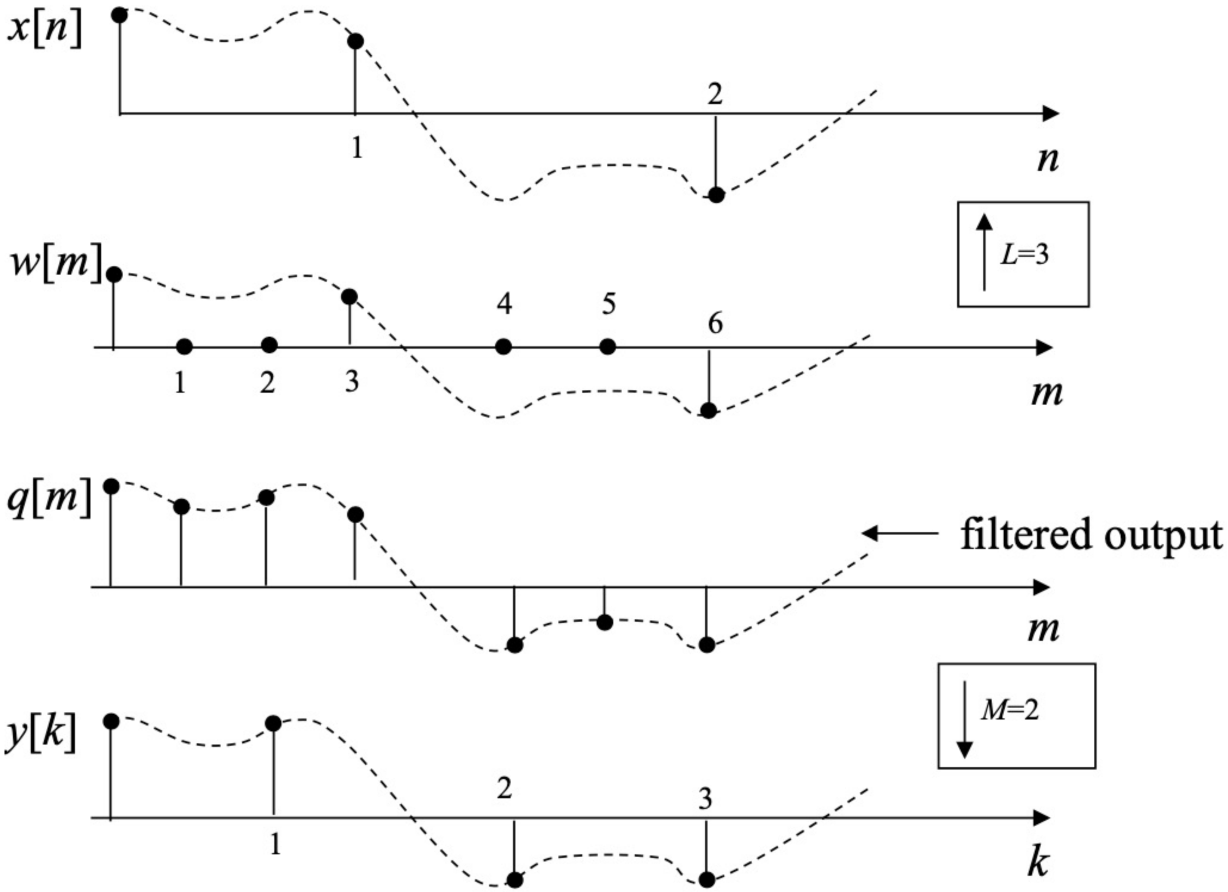
$$\omega_c = \min\left(\frac{\pi}{5}, \frac{\pi}{2}\right) = \frac{\pi}{5} \Rightarrow f_c = \frac{8000}{2 \cdot 5} = 800Hz$$

Illustration of Interpolation by a factor 3/2

The sample rate is first increased by 3, by inserting two zero-value samples for each sample of x[n] and low-pass filtered to yield q[m].

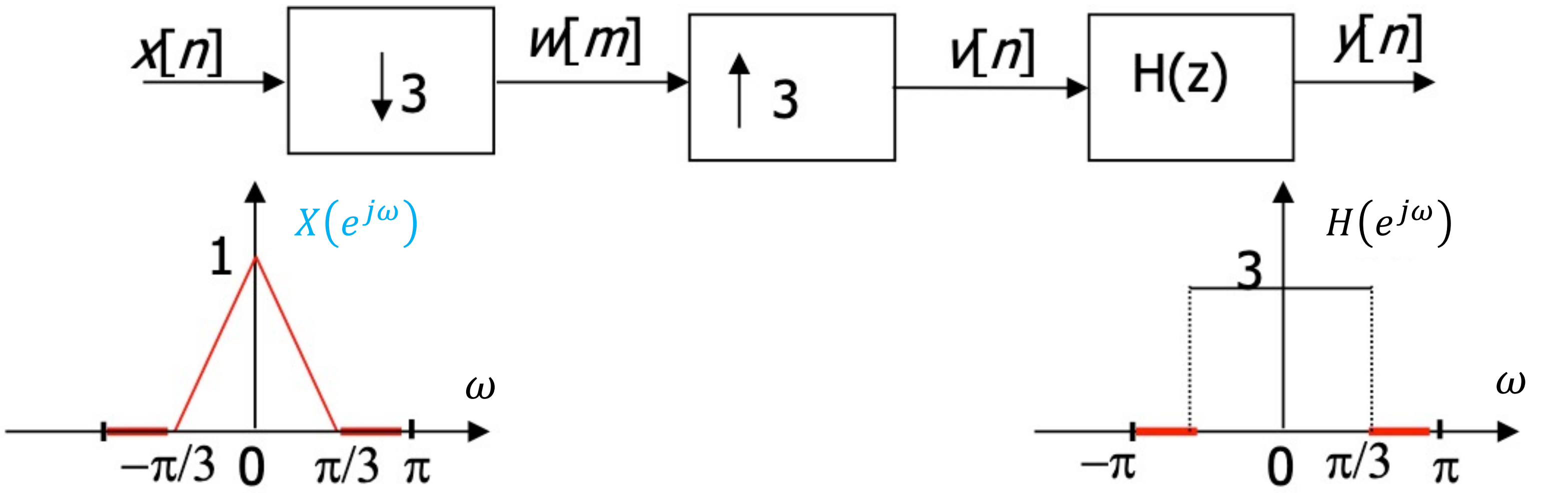
The filtered data is then reduced by a factor of 2 by retaining only one sample for every two samples of q[m].



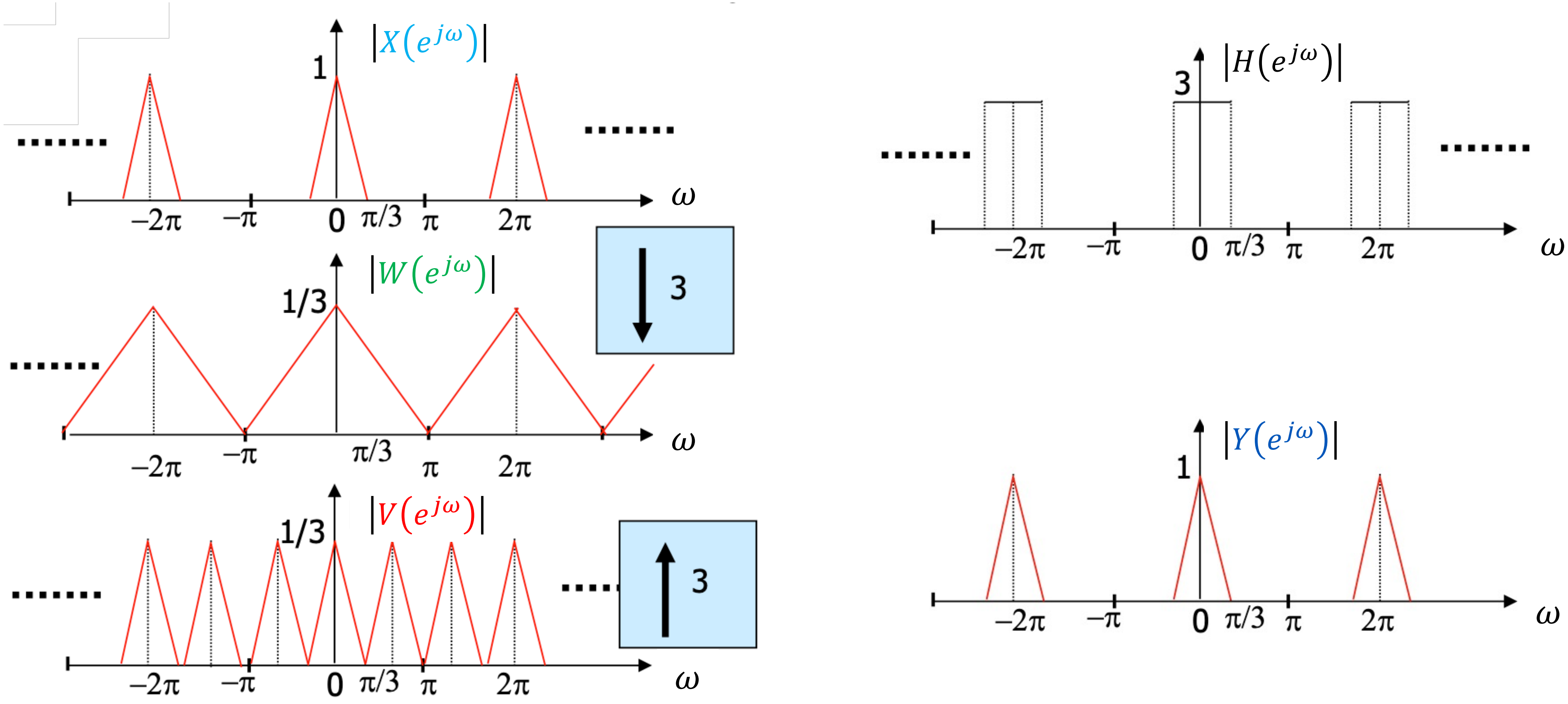


MDSP Example 1

- An input signal $x[n]$ with spectrum $X(e^{j\omega})$ is shown below. The input signal is applied to the system shown below. Sketch $|X(e^{j\omega})|$, $|W(e^{j\omega})|$, $|V(e^{j\omega})|$, $|Y(e^{j\omega})|$ against ω.

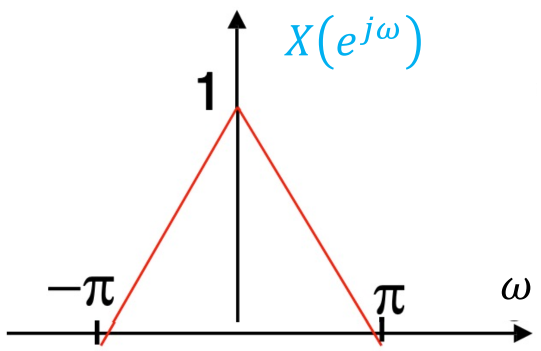


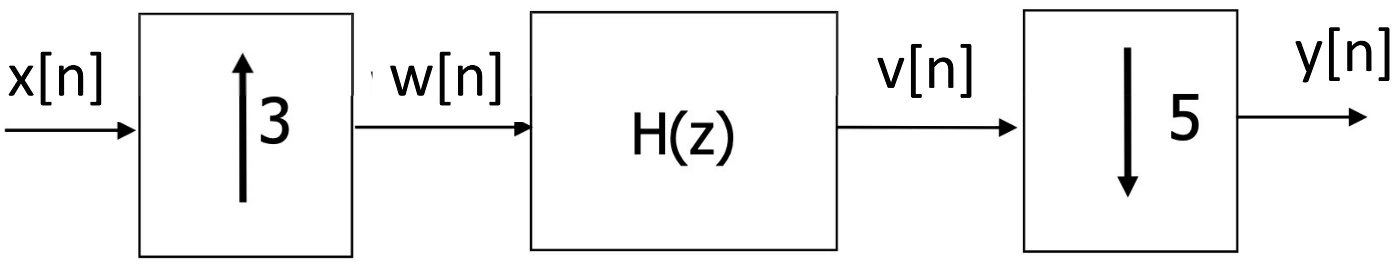
Solution of Example 1

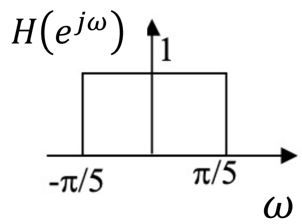


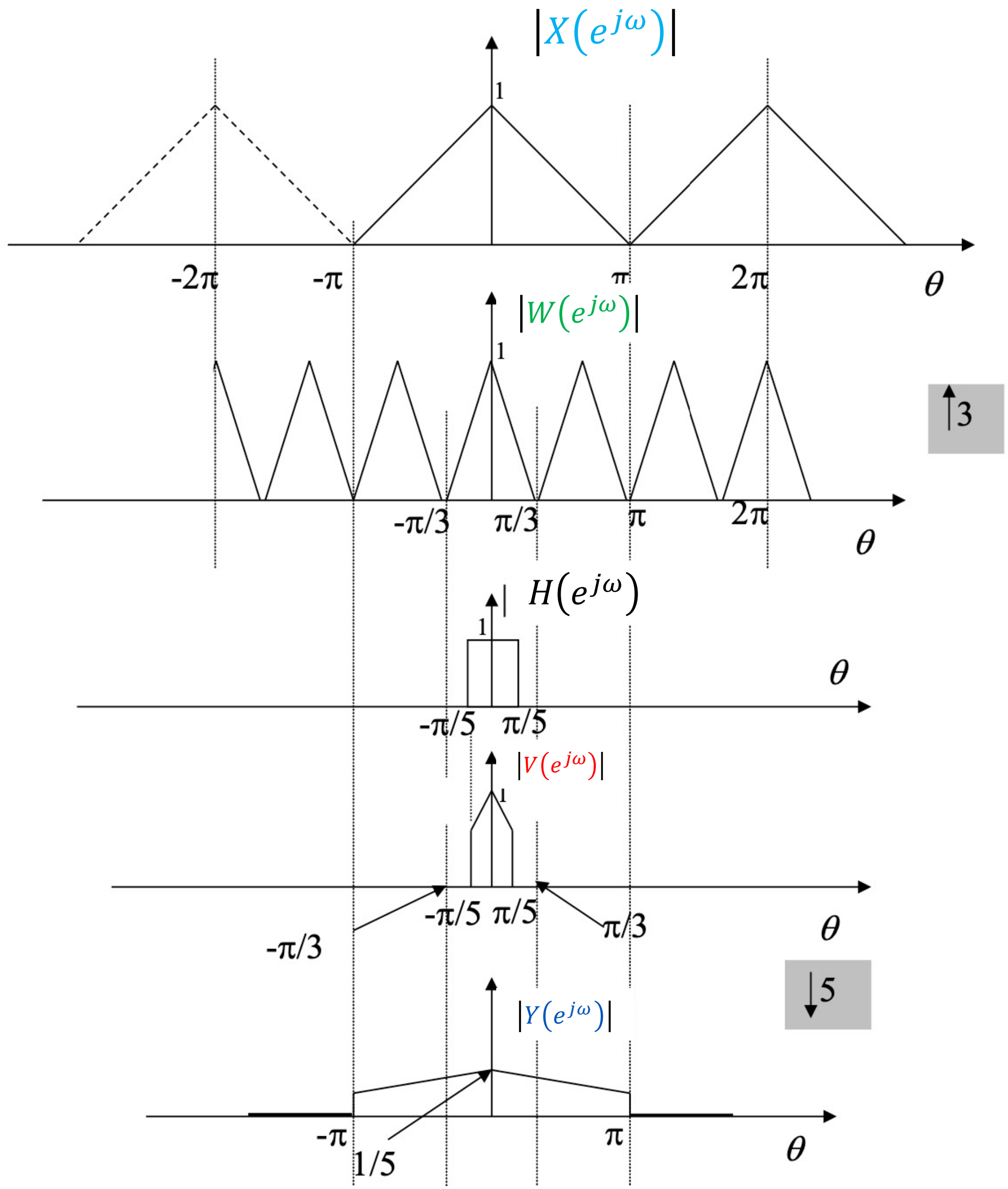
MDSP Example 2

- An input signal $x[n]$ with spectrum $X(e^{j\omega})$ is shown below. The input signal is applied to the system shown below.
- The ideal lowpass filter H(z) has a gain factor of 1 in the passband and a cut-off frequency $\omega = \pi/5$.
- Sketch $|X(e^{j\omega})|$, $|W(e^{j\omega})|$, $|V(e^{j\omega})|$, $|Y(e^{j\omega})|$ against ω.









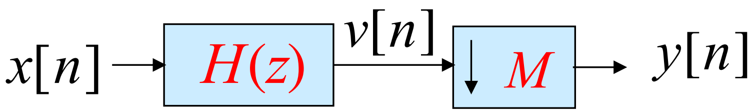
|X(e^{jω})|
1
-2π
-π
π
2π
θ
|W(e^{jω})|
1
-π/3
π/3
π
2π
θ
↑3
H(e^{jω})
1
θ
-π/5
π/5
|V(e^{jω})|
1
-π/5
π/5
-π/3
π/3
θ
↓5
|Y(e^{jω})|
-π
π
1/5
θ

Computational Requirement (1)

- The lowpass decimation or interpolation filter can be designed either as an FIR or an IIR digital filter
- In the case of single-rate digital signal processing, IIR digital filters are, in general, computationally more efficient than equivalent FIR digital filters, and are therefore preferred where computational cost needs to be minimized

Computational Requirement (2)

- This issue is not quite the same in the case of multirate digital signal processing
- To illustrate this point further, consider the factor-of-*M* decimator shown below



- If the decimation filter *H*(*z*) is an FIR filter of length *N* implemented in a direct form, then

$$v[n] = \sum_{m=0}^{N-1} h[m]x[n-m]$$

Computational Requirement (3)

- Now, the down-sampler keeps only every M-th sample of $v[n]$ at its output
- Hence, it is sufficient to compute $v[n]$ only for values of n that are multiples of M and skip the computations of in-between samples
- This leads to a factor of M savings in the computational complexity
- Now assume $H(z)$ to be an IIR filter of order K with a transfer function

$$\frac{V(z)}{X(z)} = H(z) = \frac{P(z)}{D(z)} \qquad P(z) = \sum_{n=0}^{K} p_n z^{-n} \qquad D(z) = 1 + \sum_{n=1}^{K} d_n z^{-n}$$

Computational Requirement (4)

- Its direct form implementation is given by

$$w[n] = -d_1 w[n-1] - d_2 w[n-2] - \cdots - d_K w[n-K] + x[n]$$

$$v[n] = p_0 w[n] + p_1 w[n-1] + \cdots + p_K w[n-K]$$

- Since $v[n]$ is being down-sampled, it is sufficient to compute $v[n]$ only for values of n that are integer multiples of M

Computational Requirement (5)

- However, the intermediate signal $w[n]$ must be computed for all values of n
- For example, in the computation of

$$v[M] = p_0 w[M] + p_1 w[M-1] + \cdots + p_K w[M-K]$$

  K+1 successive values of $w[n]$ are still required
- As a result, the savings in the computation in this case is going to be less than a factor of M

Computational Requirement (6)

- For the case of interpolator design, very similar arguments hold
- If $H(z)$ is an FIR interpolation filter, then the computational savings is by a factor of L (since $v[n]$ has L-1 zeros between its consecutive nonzero samples)
- On the other hand, computational savings is significantly less with IIR filters

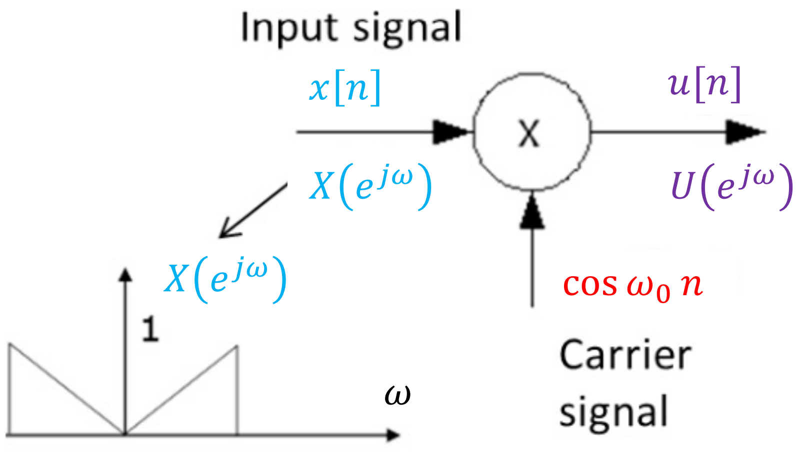
Modulation

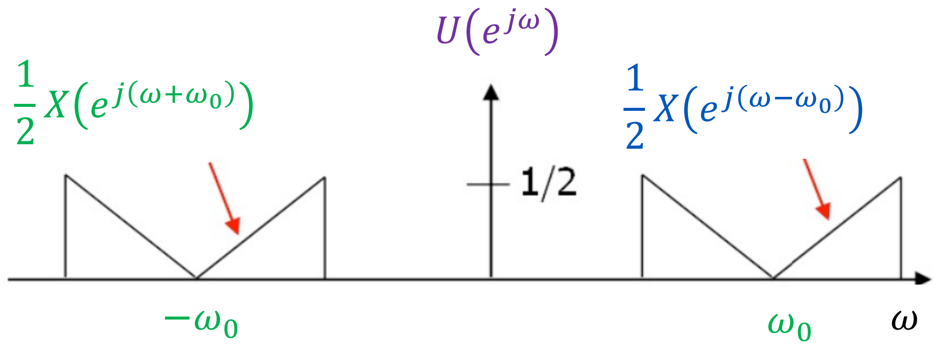
Modulation (1)

- In the time domain, modulation is the process of multiplying an input signal $x[n]$ with a sinusoidal signal known as the carrier, illustrated in the diagram on the right.
- According to the modulation property of the discrete-time Fourier transform:

$$x[n]\cos\omega_0 n \;\leftrightarrow\; \frac{1}{2}X\left(e^{j(\omega+\omega_0)}\right)+\frac{1}{2}X\left(e^{j(\omega-\omega_0)}\right)$$

- In the frequency domain, the modulated signal comprises two shifted versions of the original signal, translated by the carrier frequency.





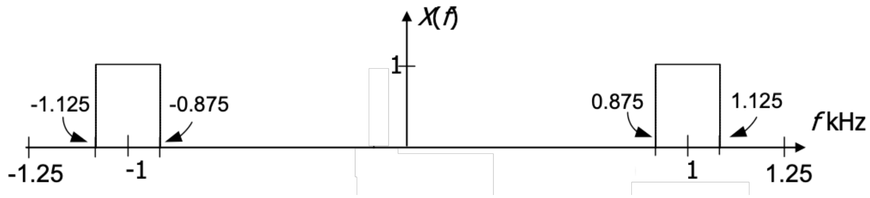
Modulation (2)

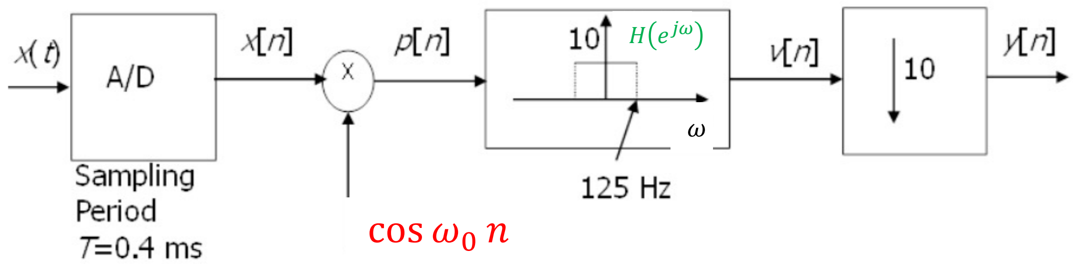
Time Domain	Frequency Domain
$x[n]\cos\omega_0 n$	$\frac{1}{2}\left[X\left(e^{j(\omega+\omega_0)}\right)+X\left(e^{j(\omega-\omega_0)}\right)\right]$
$x[n]\sin\omega_0 n$	$\frac{1}{2j}\left[X\left(e^{j(\omega+\omega_0)}\right)+X\left(e^{j(\omega-\omega_0)}\right)\right]$
$x[n]e^{j\omega_0 n}$	$e^{j(\omega+\omega_0)}$

Modulation Example 1

- $x(t)$ is the input signal for the system shown below. The analogue signal $x(t)$ has the spectrum $X(f)$ given by:

$$X(f)=\begin{cases}1 & 0.875kHz \le f \le 1.125kHz \\ 1 & -1.125kHz \le f \le -0.875kHz \\ 0 & \text{elsewhere}\end{cases}$$





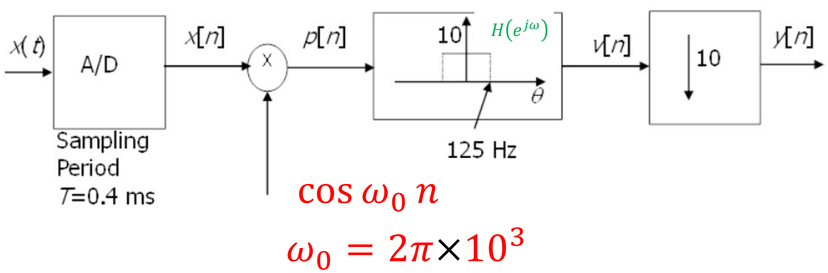
$\cos \omega_0 n$

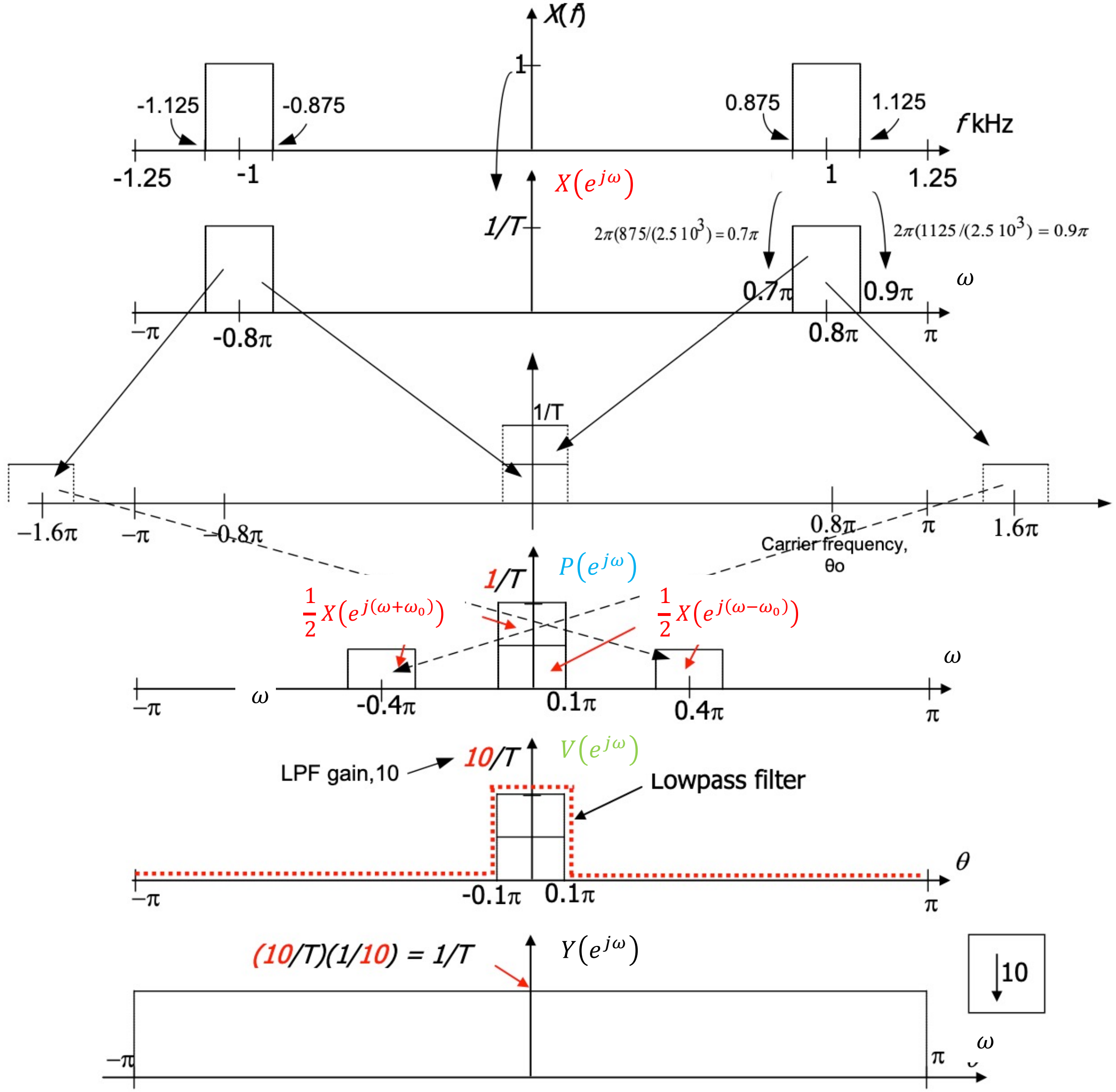
$\omega_0 = 2\pi \times 10^3$

- $H(e^{j\omega})$ is an ideal lowpass filter (gain=10) with cut-off frequency $f_c = 125$ Hz. Sketch, one above another, $|X(e^{j\omega})|$, $|P(e^{j\omega})|$, $|V(e^{j\omega})|$, $|Y(e^{j\omega})|$ against ω .

Solution

- $F_s = \frac{1}{T} = 2.5$ kHz
- Carrier Frequency :
  - $\omega_0 = 2\pi 10^3 T = 0.8\pi$
- LPF Cut-off Frequency :
  - $\omega_c = 2\pi\left(\frac{125}{2.5\times10^3}\right) = 0.1\pi$

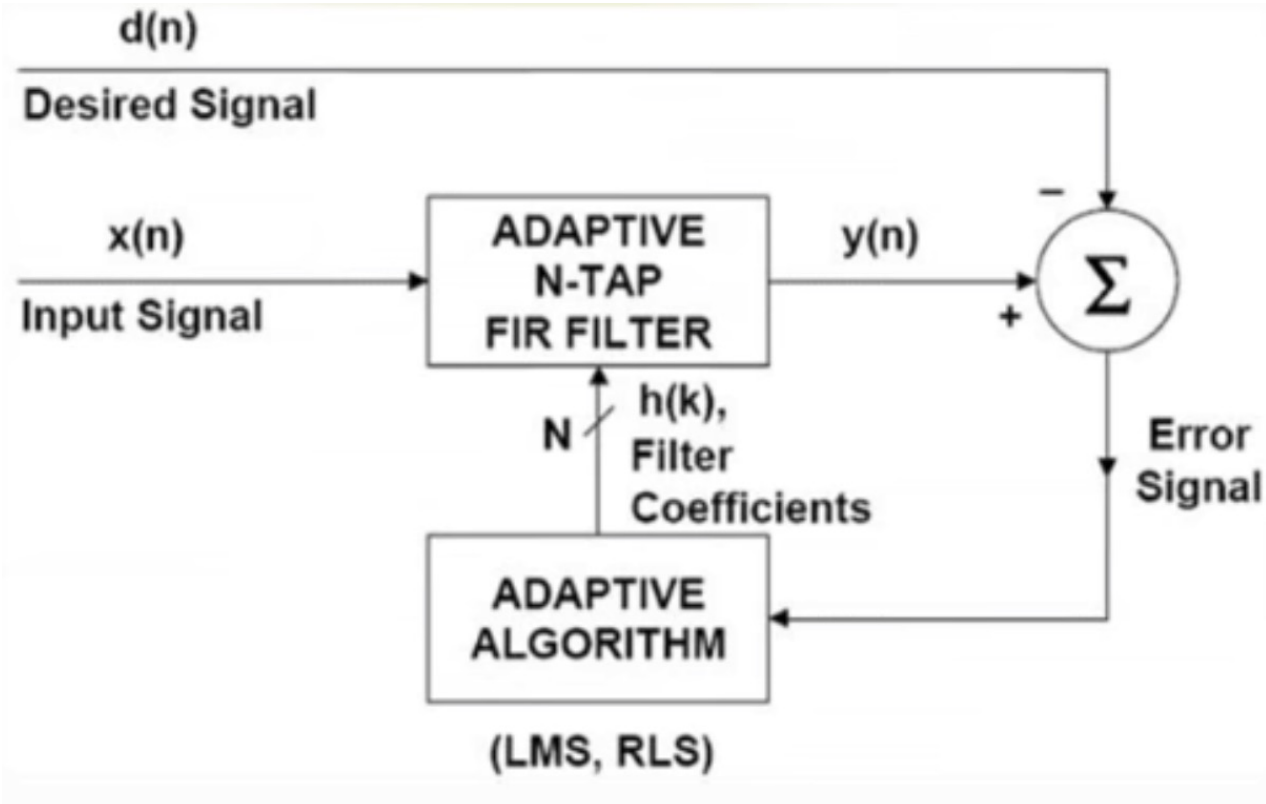




Modulation Example 2

- The sampling period T of the input signal shown in the figure below is 125 μs. The relative frequency is $\theta = \omega T$. The first oscillator generates a carrier with a relative frequency $\theta 1 = \omega 1T$, where $\omega 1 = 2\pi .2.103$ rad / sec. The second
- oscillator generates a carrier with a relative frequency $\theta =(\omega c+\omega)T$, where$\omega =2\pi.103$rad/sec. The low-pass
- 21c
- filter has the following characteristics:

Adaptive Filters



Digital Transmission using Adaptive Equalization