EE3731C:
Signal Processing Methods

Lecture II-1:
Multirate Digital Signal Processing

National University of Singapore

Outline

- Introduction to Multirate Digital Signal Processing
- Interpolation (Up-sampling)
- Decimation (Down-sampling)
- Sampling Rate Conversion
- Multirate Identities
- Filters in Multirate Systems

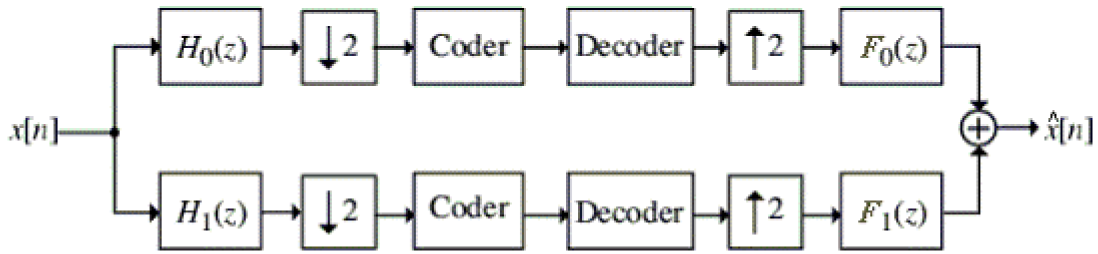
Multirate Digital Signal Processing

- Multirate Digital Signal Processing (DSP)
  - In single-rate DSP systems, all data is sampled at the same rate, i.e., no change of rate within the system.
  - In multirate DSP systems, sampling rates are changed.
- Example: audio sampling rate conversion
  - Various systems in digital audio signal processing often operate at different sampling rates. The connection of such systems requires a conversion of sampling rate.
    - Recording studios: 192 kHz
    - CD: 44.1 kHz
    - DVD movie sound track: mostly 48kHz
    - DVD-audio: 96kHz

Demo

Multirate DSP

- Advantages of multirate DSP:
  - –Decimation (down-sampling) reduces the sampling rate.
  - –Interpolation (up-sampling) increases the sampling rate.
  - –Reduced computational complexity
  - –Reduced transmission data rate
- Applications:
  - –Subband coding of speech, audio, and video signals
  - –Fast transforms using digital filter banks and wavelet analysis of signals



Fundamental Multirate Operations

- Interpolation (up-sampling)
  - Increase the sampling rate by an integer factor of L
  - L-fold expander/up-sampler and lowpass filter

$$x[n] \rightarrow \boxed{\uparrow L} \rightarrow \boxed{H(z)} \rightarrow y[n]$$

- Decimation (down-sampling)
  - Decrease the sampling rate by an integer factor of M
  - Lowpass filter and M-fold decimator/down-sampler

$$x[n] \rightarrow \boxed{H(z)} \rightarrow \boxed{\downarrow M} \rightarrow y[n]$$

Up-sampling

- For an input sequence $x[n]$, insert $L-1$ zeros in-between every two samples
- The output sequence has a sampling rate L times that of the input sequence.
- L-fold expander (up-sampler)

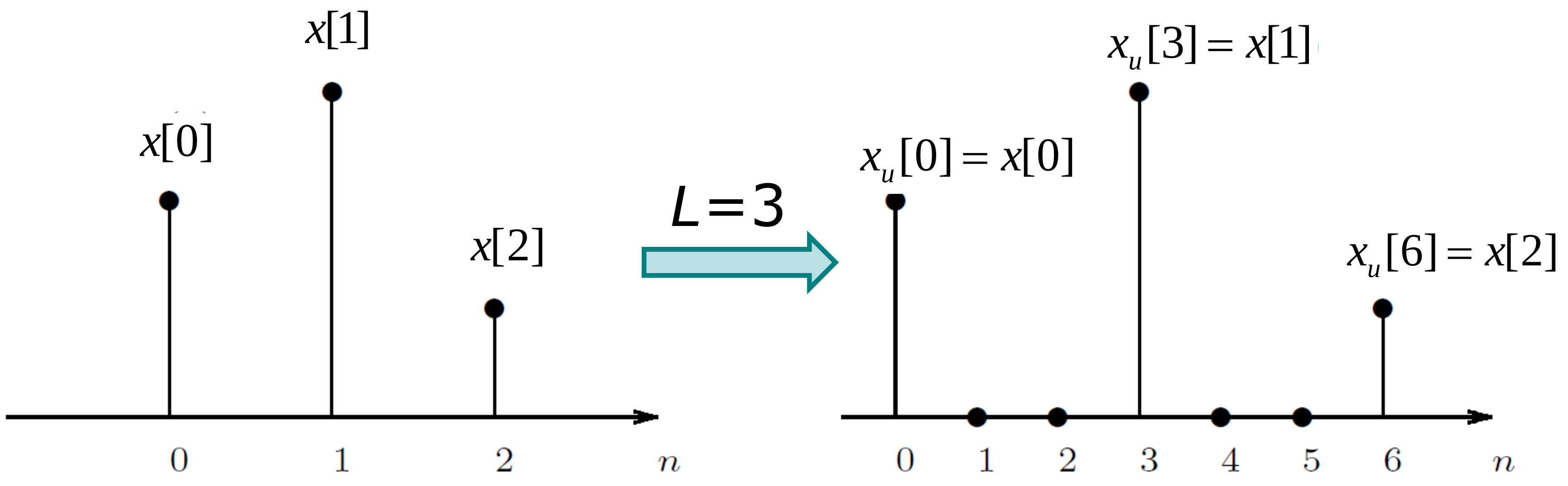
$$x[n] \longrightarrow \boxed{\uparrow L} \longrightarrow x_u[n]$$

$$x_u[n] = \begin{cases} x[n/L], & n = 0, \pm L, \pm 2L, \cdots \\ 0, & \text{otherwise} \end{cases}$$

MATLAB: y = upsample(x,L) increases the sampling rate of x by inserting L-1 zeros between samples.

Up-sampling: Example

$$x_u[n] = \begin{cases} x[n/L], & n = 0, \pm L, \pm 2L, \cdots \\ 0, & \text{otherwise} \end{cases}$$



- $x[n]$ can always be recovered from $x_u[n]$
- No loss of information

Frequency Domain View

- Z transform of $x_u[n] = \begin{cases} x[n/L], & n = 0, \pm L, \pm 2L, \cdots \\ 0, & \text{otherwise} \end{cases}$

$$X_u(z) = \sum_{n=-\infty}^{\infty} x_u[n] z^{-n} = \sum_{k=-\infty}^{\infty} x_u[kL] z^{-kL}$$

$$= \sum_{k=-\infty}^{\infty} x[k] z^{-kL} = \sum_{k=-\infty}^{\infty} x[k] \left(z^L\right)^{-k}$$

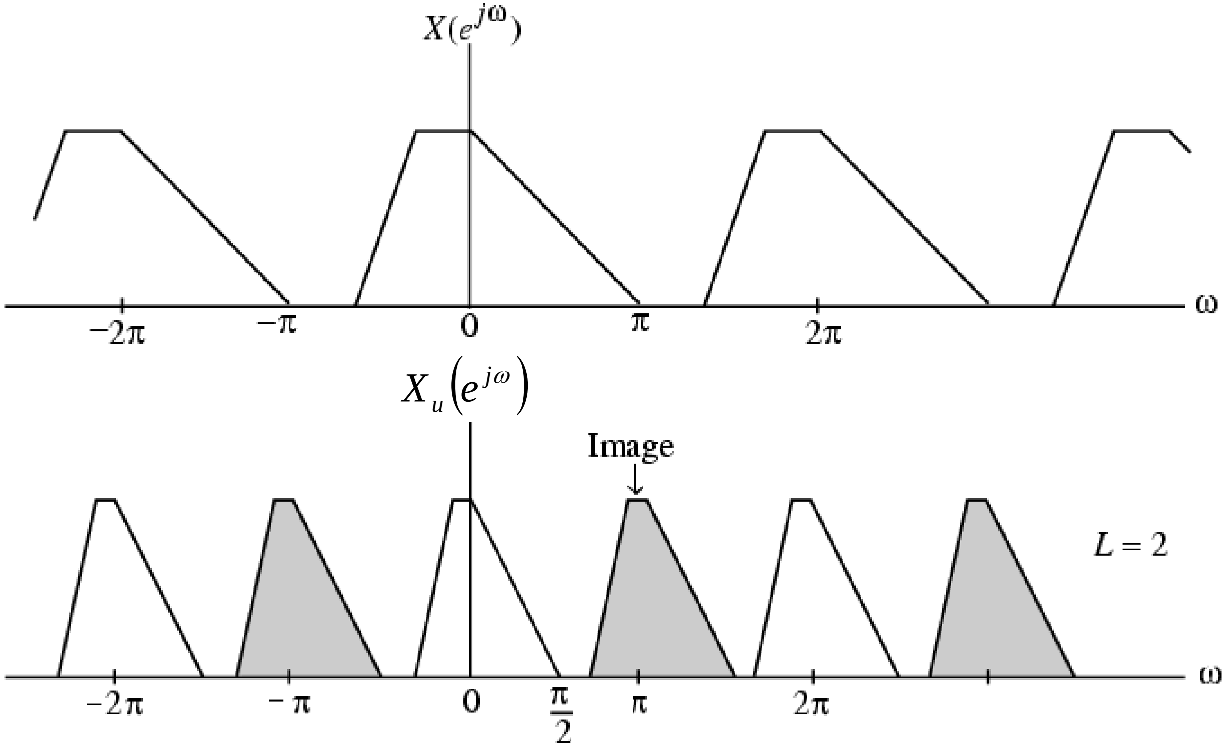
$$n = \begin{cases} kL \\ \text{otherwise} \end{cases}$$

- Z transform of $x[n]$: $X(z) = \sum_{n=-\infty}^{\infty} x[n] z^{-n}$

$$X_u(z) = X\left(z^L\right) \xrightarrow{z=e^{j\omega}} X_u\left(e^{j\omega}\right) = X\left(e^{j\omega L}\right)$$

  - X_u is a compressed version of X
  - Multiple images of $X\left(e^{j\omega}\right)$ are created between 0 and 2π

Frequency Domain Illustration



For interpolation, a lowpass filter is applied to remove the images. In effect, it “fills in” the zero-valued samples with interpolated sample values.

Down-sampling

- Keep every *M*th sample of $x[n]$ and discard $M-1$ in-between samples
- The output sequence has a sampling rate $1/M$ that of the input sequence.
- *M*-fold decimator (down-sampler)

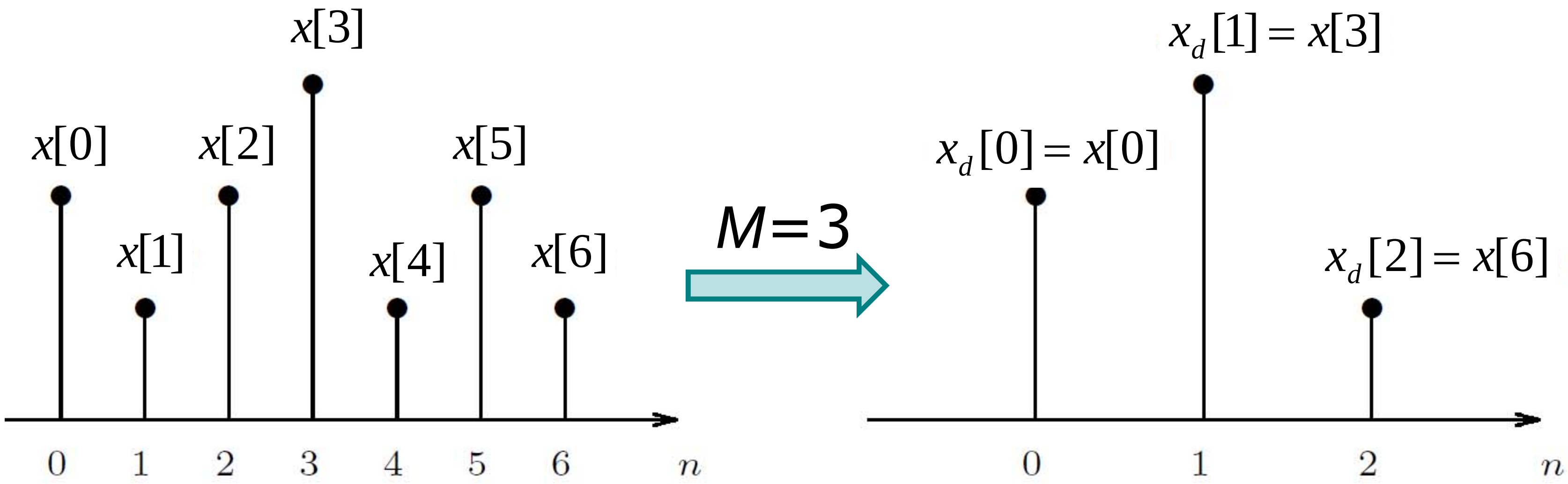
$$x[n] \longrightarrow \boxed{\downarrow M} \longrightarrow x_d[n]$$

$$x_d[n] = x[nM]$$

MATLAB: y = downsample(x,M) decreases the sampling rate of x by keeping every M-th sample starting with the first sample.

Down-sampling: Example

$$x_d[n] = x[nM]$$



- Aliasing will occur in $x_d[n]$ unless $x[n]$ is sufficiently bandlimited.
- Loss of information

Frequency Domain View

- Z transform of $x_d[n] = x[nM]$

$$X_d(z) = \sum_{n=-\infty}^{\infty} x_d[n] z^{-n} = \sum_{n=-\infty}^{\infty} x[nM] z^{-n}$$

Let $x_1[n] = \begin{cases} x[n], & n = 0, \pm M, \pm 2M, \ldots \\ 0, & \text{otherwise} \end{cases}$ $\quad k = \begin{cases} nM \\ \text{otherwise} \end{cases}$

Then $X_d(z) = \sum_{n=-\infty}^{\infty} x_1[nM] z^{-n} = \sum_{k=-\infty}^{\infty} x_1[k] z^{-k/M} = X_1\left(z^{1/M}\right)$

since $x_1[k] = 0$ unless k is a multiple of M.

What is $X_1(z)$?

Frequency Domain View

Let $c[n] = \begin{cases} 1, & n = 0, \pm M, \pm 2M, \ldots \\ 0, & \text{otherwise} \end{cases}$

Then $x_1[n]$ can be related to $x[n]$ as $x_1[n] = c[n] \cdot x[n]$

Rewrite $c[n]$ as $c[n] = \frac{1}{M} \sum_{k=0}^{M-1} W_M^{-kn}$, where $W_M = e^{-j2\pi/M}$

$$X_1(z) = \sum_{n=-\infty}^{\infty} c[n]x[n]z^{-n} = \frac{1}{M} \sum_{n=-\infty}^{\infty} \left(\sum_{k=0}^{M-1} W_M^{-kn} \right) x[n] z^{-n}$$

$$= \frac{1}{M} \sum_{k=0}^{M-1} \left(\sum_{n=-\infty}^{\infty} x[n] W_M^{-kn} z^{-n} \right) = \frac{1}{M} \sum_{k=0}^{M-1} X\left(z W_M^k\right)$$

Therefore, $X_d(z) = X_1\left(z^{1/M}\right) = \frac{1}{M} \sum_{k=0}^{M-1} X\left(z^{1/M} W_M^k\right)$

Interpretation

- Frequency response:

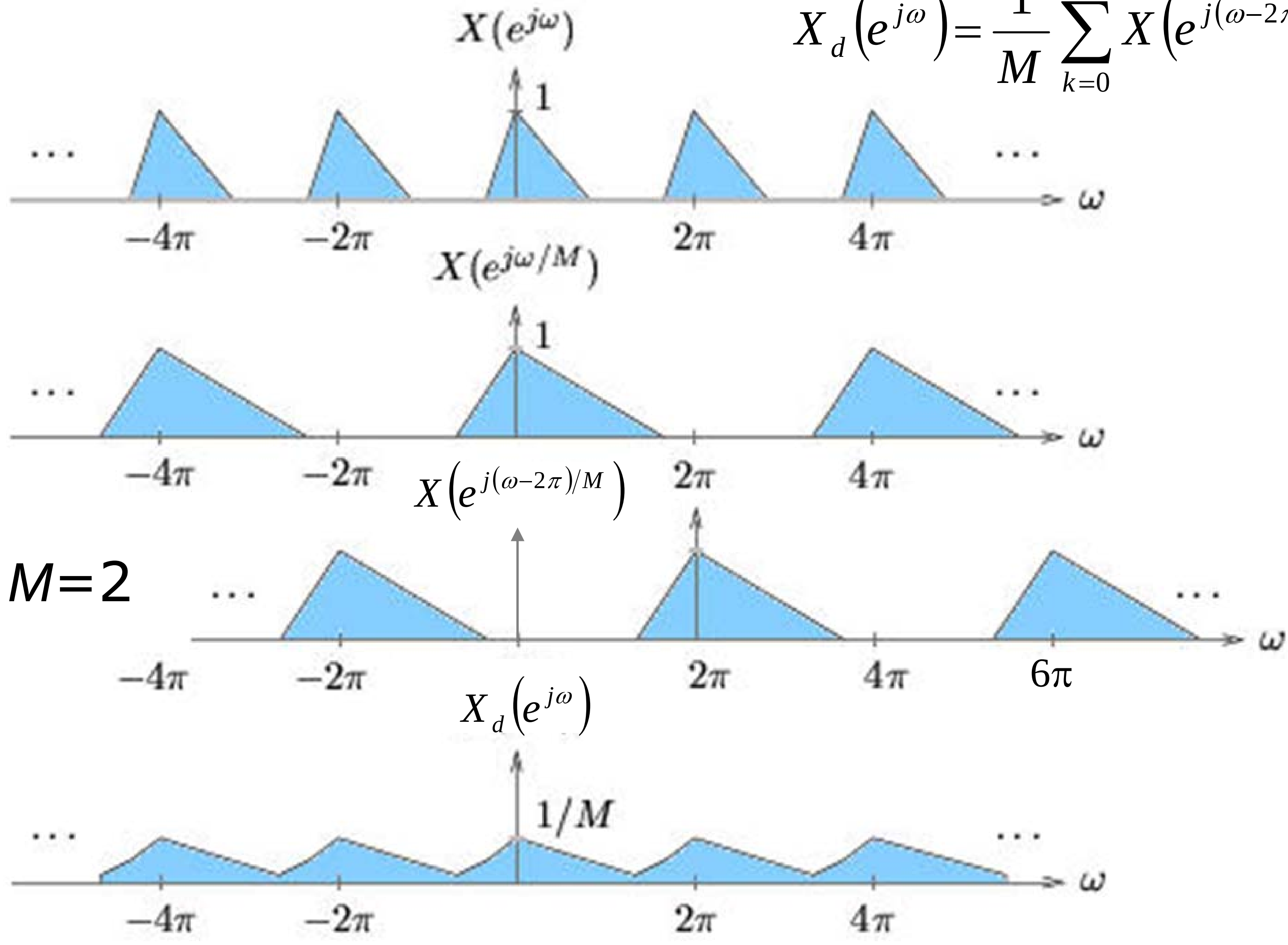
$$X_d(z) = \frac{1}{M}\sum_{k=0}^{M-1} X\left(z^{1/M} W_M^k\right) \xrightarrow{z=e^{j\omega}} X_d\left(e^{j\omega}\right) = \frac{1}{M}\sum_{k=0}^{M-1} X\left(e^{j(\omega-2\pi k)/M}\right)$$

- What does it represent?
  - Stretch $X\left(e^{j\omega}\right)$ to $X\left(e^{j\omega/M}\right)$
  - Summing *M* copies of the stretched version that are shifted by integer multiples of 2π, i.e., 0, 2π,…, 2π(*M*-1)
  - Dividing the result by *M*

$X_d\left(e^{j\omega}\right)$ is a sum of *M* uniformly shifted and stretched versions of $X\left(e^{j\omega}\right)$ and scaled by a factor of 1/*M*.

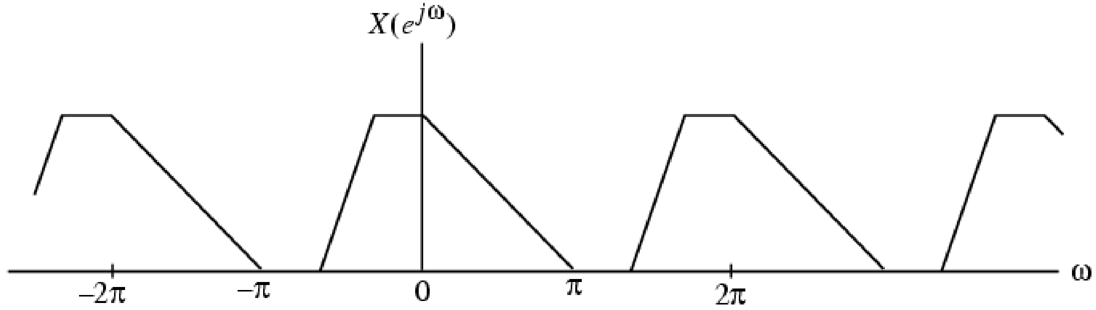
Frequency Domain Illustration

$$X_d\left(e^{j\omega}\right)=\frac{1}{M}\sum_{k=0}^{M-1}X\left(e^{j(\omega-2\pi k)/M}\right)$$



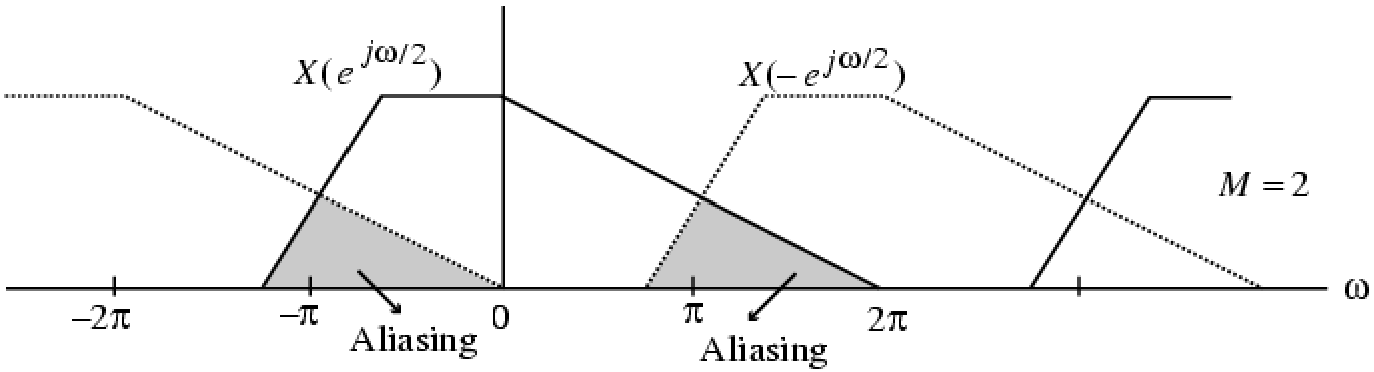
Frequency Domain Illustration

- Spectrum of the input signal



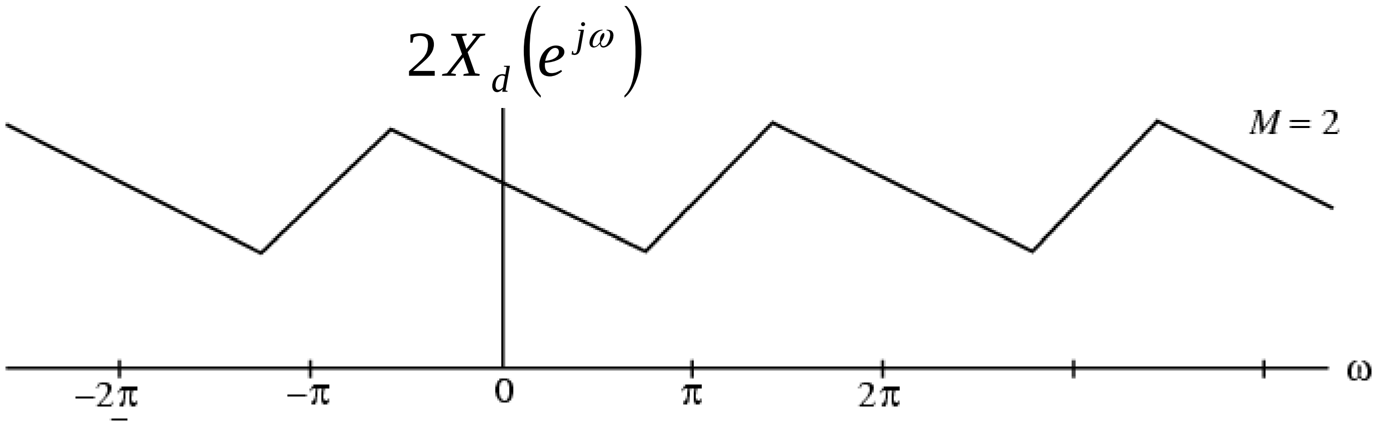
- Down-sample by a factor of 2, i.e., *M*=2

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



Frequency Domain: Example

- Spectrum of the down-sampled signal

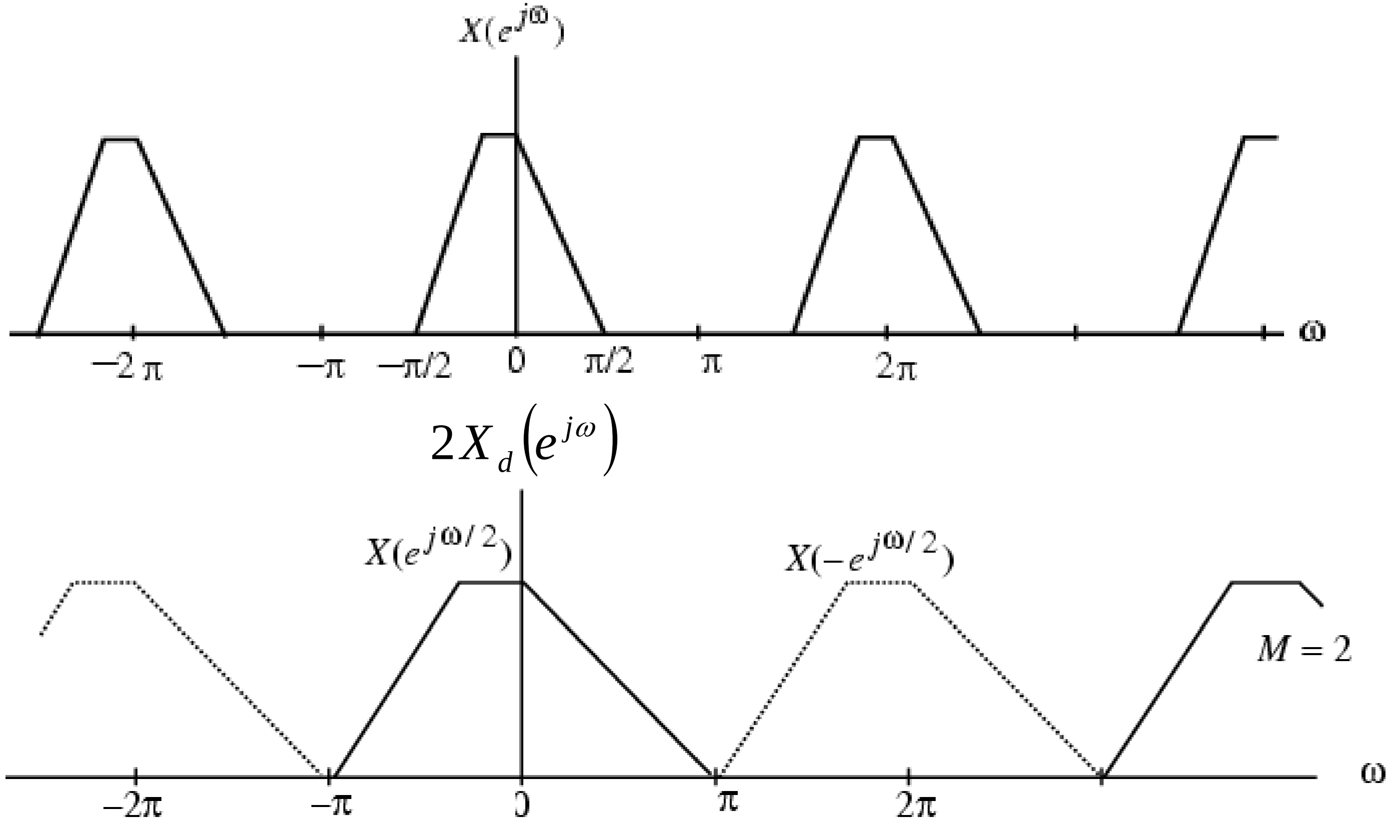


- The original "shape" of $X\left(e^{j\omega}\right)$ is lost due to overlap/aliasing.
- For *M*=2, no overlap/aliasing, only if $X\left(e^{j\omega}\right)=0 \text{ for } |\omega| \geq \frac{\pi}{2}$
- For any *M*, aliasing is absent if and only if

$$X\left(e^{j\omega}\right)=0 \text{ for } |\omega| \geq \frac{\pi}{M}$$

Frequency Domain: Example

- *M*=2, $X\left(e^{j\omega}\right)=0 \text{ for } |\omega| \geq \frac{\pi}{2}$



no overlap/aliasing!

Sampling Rate Conversion

- How to change a sampling rate by a non-integer factor of *L*/*M*?

$$x[n] \longrightarrow \boxed{\uparrow L} \longrightarrow \boxed{H(z)} \longrightarrow \boxed{\downarrow M} \longrightarrow y[n]$$

  - The signal is first interpolated by a factor of *L*
  - Then decimated by a factor of *M*
  - Before decimation the signal must be filtered by a low-pass filter which acts as not only an interpolator but also an anti-aliasing filter.
  - When either *M* or *L* = 1, it reduces to interpolation or decimation filtering.

Multirate Identities

- Interchange of Filtering and Up-sampling

$$x[n] \rightarrow \boxed{\uparrow L} \rightarrow \boxed{H(z^L)} \rightarrow y_1[n] \equiv x[n] \rightarrow \boxed{H(z)} \rightarrow \boxed{\uparrow L} \rightarrow y_2[n]$$

- Interchange of Filtering and Down-sampling

$$x[n] \rightarrow \boxed{\downarrow M} \rightarrow \boxed{H(z)} \rightarrow y_1[n] \equiv x[n] \rightarrow \boxed{H(z^M)} \rightarrow \boxed{\downarrow M} \rightarrow y_2[n]$$

- Interchange of Up-sampling and Down-sampling

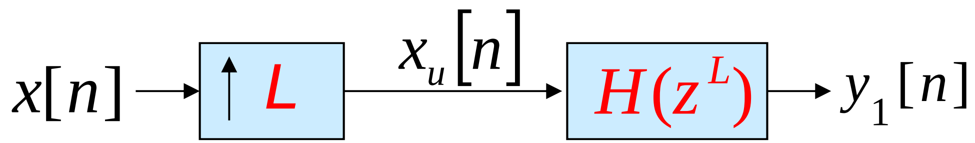
$$x[n] \rightarrow \boxed{\downarrow M} \rightarrow \boxed{\uparrow L} \rightarrow y_1[n] \equiv x[n] \rightarrow \boxed{\uparrow L} \rightarrow \boxed{\downarrow M} \rightarrow y_2[n]$$

if and only if *M* and *L* are relatively prime, i.e., *M* and *L* share no common positive factors except 1

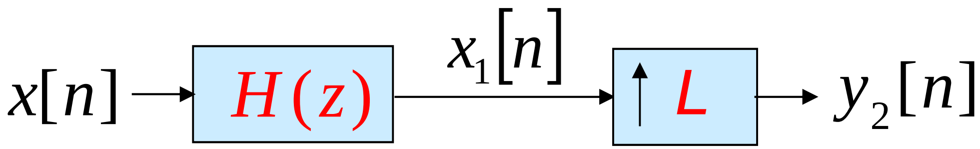
Multirate Identities

- Interchange of Filtering and Up-sampling

Proof:



$$X_u(z) = X\left(z^L\right) \qquad Y_1(z) = X\left(z^L\right)H\left(z^L\right)$$



$$X_1(z) = X(z)H(z) \qquad Y_2(z) = X\left(z^L\right)H\left(z^L\right)$$

$$Y_1(z) = Y_2(z)$$

Multirate Identities

- Interchange of Filtering and Down-sampling

Proof:

$$x[n] \rightarrow \boxed{\downarrow M} \xrightarrow{x_d[n]} \boxed{H(z)} \rightarrow y_1[n]$$

$$X_d(z) = \frac{1}{M}\sum_{k=0}^{M-1} X\left(z^{1/M} W_M^k\right) \qquad Y_1(z) = H(z)\frac{1}{M}\sum_{k=0}^{M-1} X\left(z^{1/M} W_M^k\right)$$

$$x[n] \rightarrow \boxed{H(z^M)} \xrightarrow{x_1[n]} \boxed{\downarrow M} \rightarrow y_2[n]$$

$$X_1(z) = H\left(z^M\right)X(z)$$

$$Y_2(z) = \frac{1}{M}\sum_{k=0}^{M-1} H\left(\left(z^{1/M} W_M^k\right)^M\right) X\left(z^{1/M} W_M^k\right)$$

$$= \frac{1}{M}\sum_{k=0}^{M-1} H(z) X\left(z^{1/M} W_M^k\right)$$

$$= H(z)\frac{1}{M}\sum_{k=0}^{M-1} X\left(z^{1/M} W_M^k\right)$$

Multirate Identities

- Interchange of Up-sampling and Down-sampling

$$x[n] \rightarrow \boxed{\downarrow M} \rightarrow \boxed{\uparrow L} \rightarrow y_1[n] \quad \equiv \quad x[n] \rightarrow \boxed{\uparrow L} \rightarrow \boxed{\downarrow M} \rightarrow y_2[n]$$

if and only if *M* and *L* are relatively prime, i.e., *M* and *L* share no common positive factors except 1

Proof:

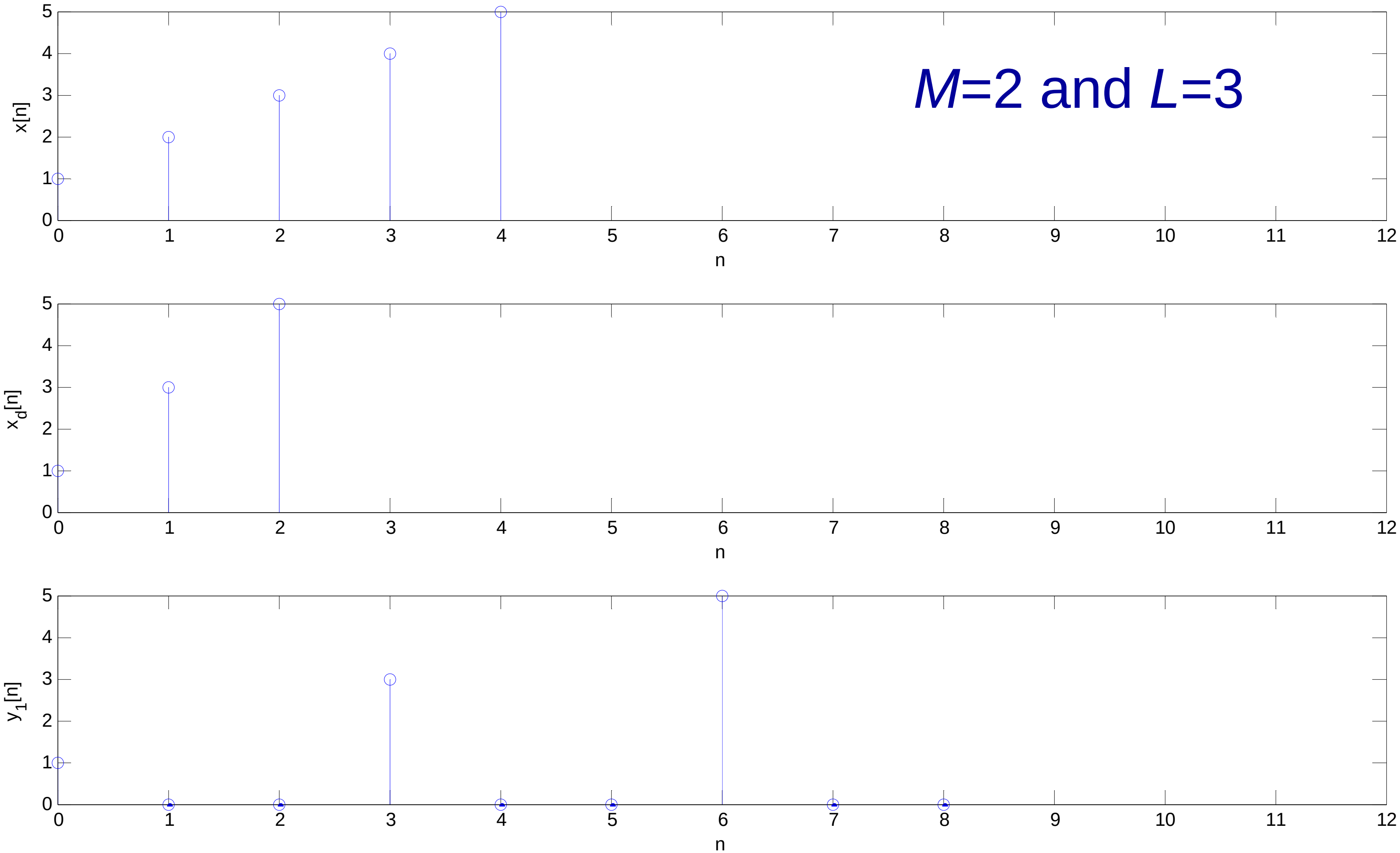
$$X_d(z) = \frac{1}{M}\sum_{k=0}^{M-1} X\left(z^{1/M} W_M^k\right) \qquad Y_1(z) = X_d\left(z^L\right) = \frac{1}{M}\sum_{k=0}^{M-1} X\left(z^{L/M} W_M^{kL}\right)$$

$$X_u(z) = X\left(z^L\right) \qquad Y_2(z) = \frac{1}{M}\sum_{k=0}^{M-1} X_u\left(z^{1/M} W_M^k\right) = \frac{1}{M}\sum_{k=0}^{M-1} X\left(z^{L/M} W_M^{kL}\right)$$

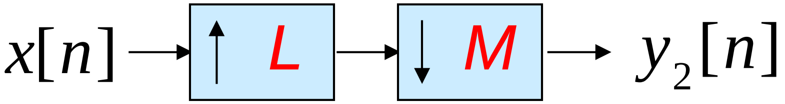
For the two expressions to be equal, *M* and *L* must be relatively prime.

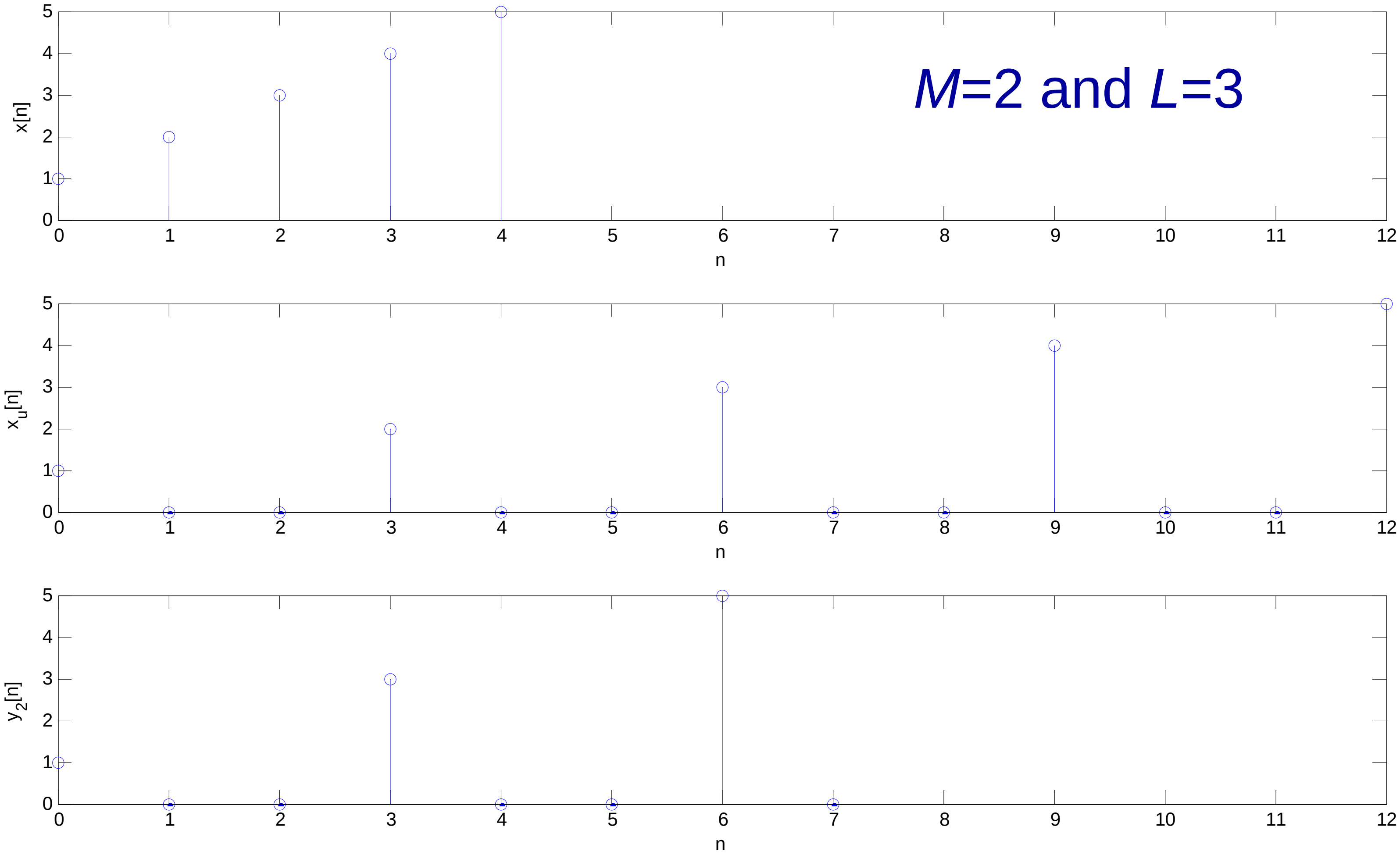
Multirate Identities: Example

$$x[n] \rightarrow \boxed{\downarrow M} \rightarrow \boxed{\uparrow L} \rightarrow y_1[n]$$



Multirate Identities: Example





Filters in Multirate Systems

- Interpolation filter

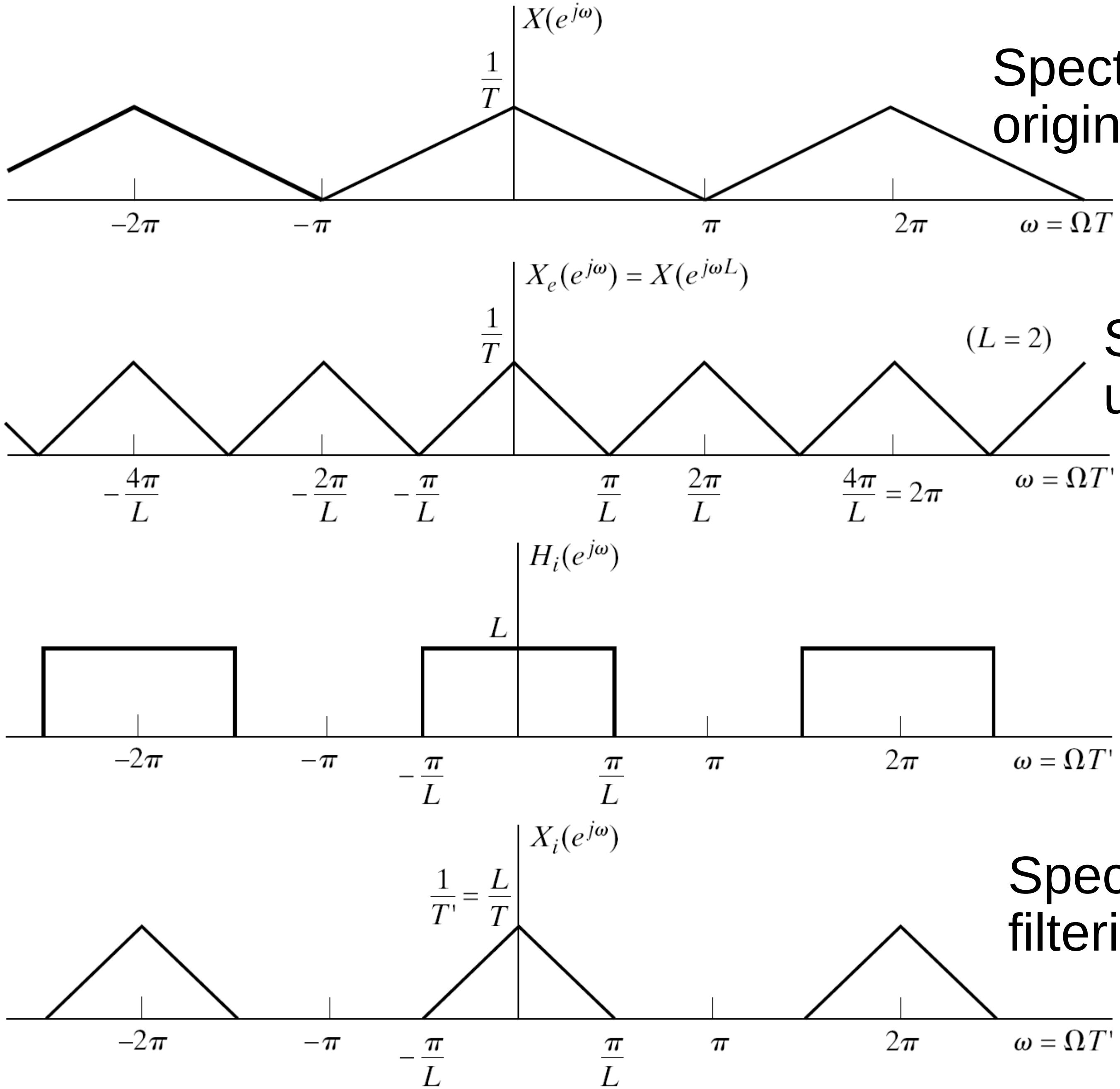
$$x[n] \longrightarrow \boxed{\uparrow L} \longrightarrow \boxed{H(z)} \longrightarrow y[n]$$

  - Since up-sampling causes periodic repetition of the basic spectrum, the unwanted images in the spectra of the up-sampled signal $x_u[n]$ must be removed by using a lowpass interpolation filter $H(z)$.
  - The cutoff frequency of the lowpass interpolation filter

$$\omega_c = \frac{\pi}{L} \qquad f_c = \frac{f_s}{2L}$$

f_s: the input sampling rate

MATLAB: y = interp(x,r) increases the sampling rate of x by a factor of r.



Spectrum of the original signal $x[n]$

Spectrum after up-sampling

$$T' = \frac{T}{L}$$

Spectrum after filtering

Filters in Multirate Systems

- Decimation filter

$$x[n] \longrightarrow \boxed{H(z)} \longrightarrow \boxed{\downarrow M} \longrightarrow y[n]$$

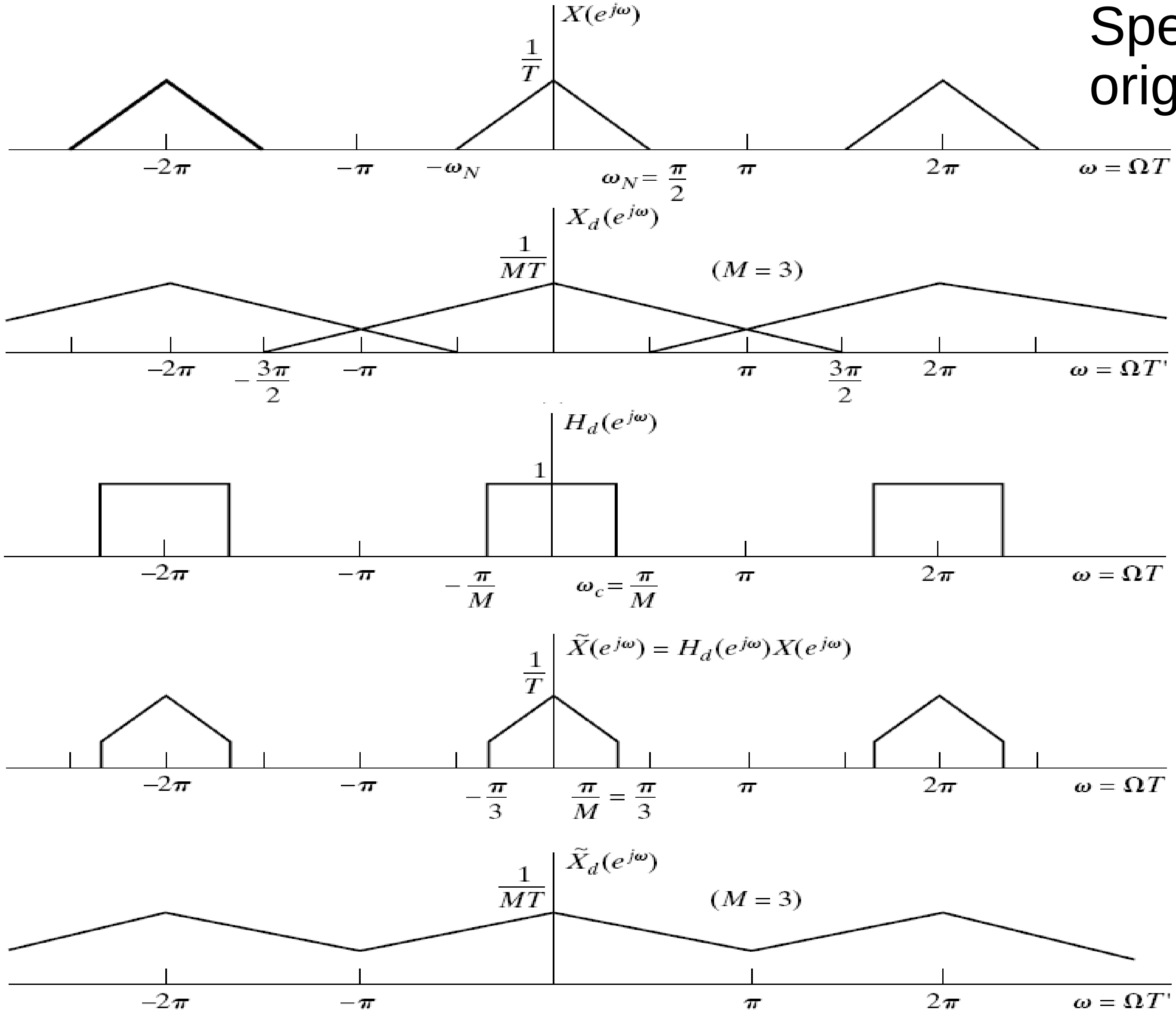
– Prior to down-sampling, the signal should be bandlimited to $|\omega| < \pi / M$ by means of a lowpass decimation filter, to avoid aliasing caused by down-sampling.

– The cutoff frequency of the lowpass decimation filter:

$$\omega_c = \frac{\pi}{M} \qquad f_c = \frac{f_s}{2M}$$

f_s: the input sampling rate

MATLAB: y = decimate(x,r) reduces the sample rate of x by a factor r.



Spectrum of the original signal *x*[*n*]

Spectrum after down-sampling with aliasing

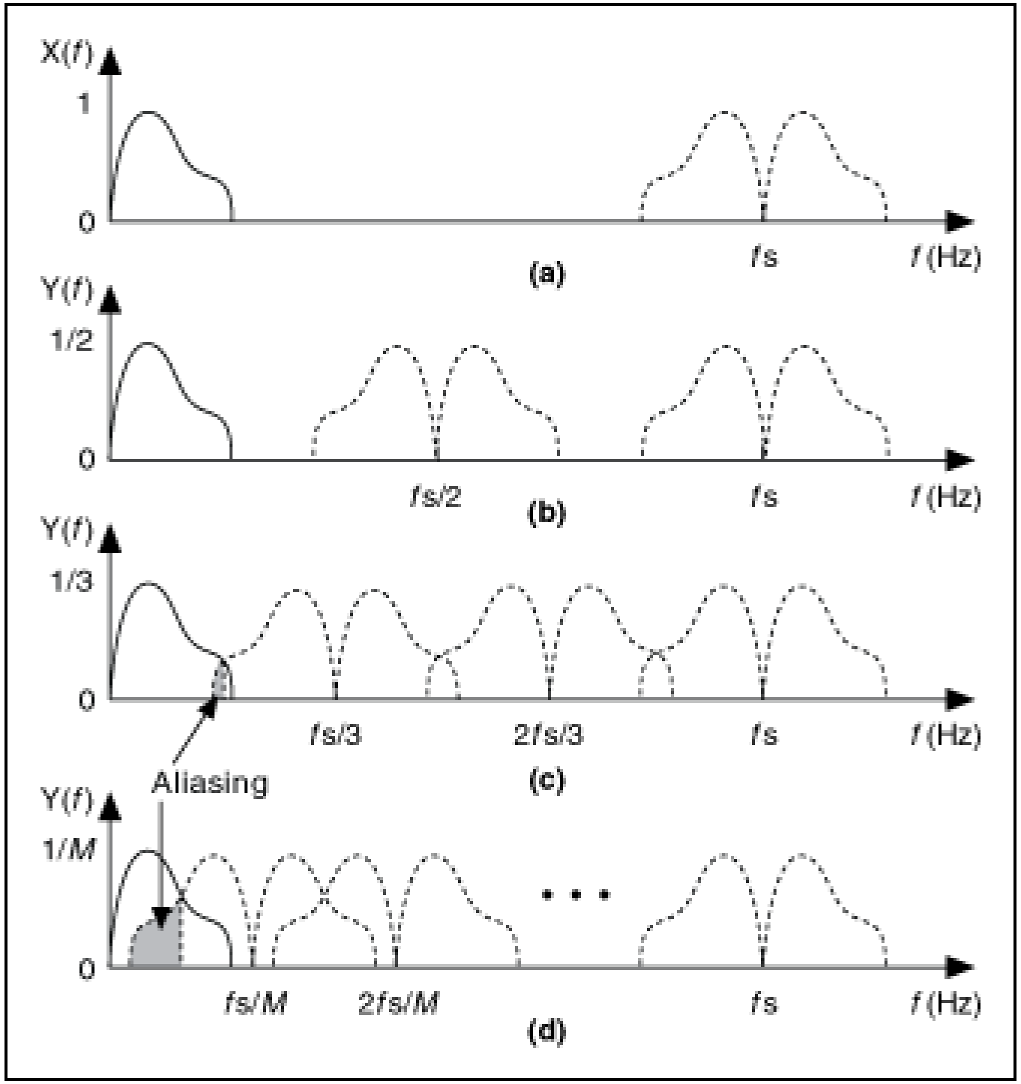
Lowpass filtering

Spectrum after filtering

Spectrum after down-sampling without aliasing

$$T' = MT$$

The Cut-off Frequency



Spectrum of the original signal *x*[*n*]

Spectrum of the output signal *y*[*n*], decimated by a factor of 2

Spectrum of the output signal *y*[*n*], decimated by a factor of 3

Spectrum of the output signal *y*[*n*], decimated by a factor of *M*

Filters in Multirate Systems

- Decimation and interpolation filter

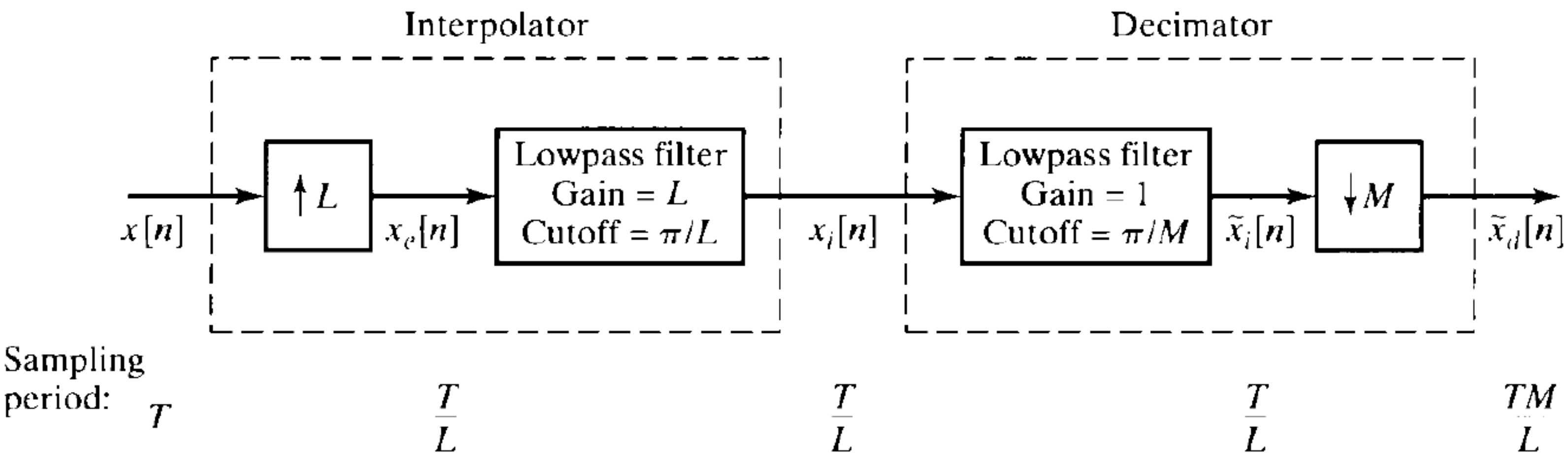
$$x[n] \longrightarrow \boxed{\uparrow L} \longrightarrow \boxed{H(z)} \longrightarrow \boxed{\downarrow M} \longrightarrow y[n]$$

  - *H*(*z*) serves as both the interpolation and the decimation filter.
  - *H*(*z*) should have a cut-off frequency given by

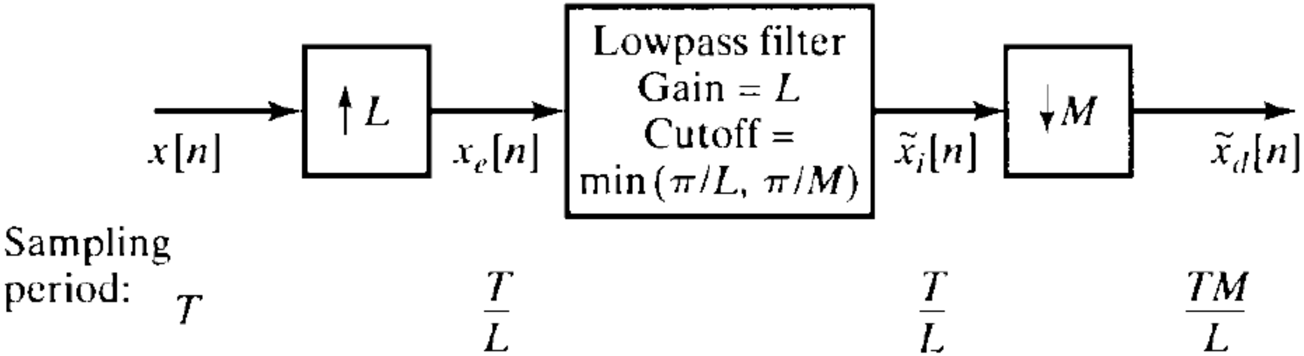
$$\omega_c = \min\left(\frac{\pi}{L}, \frac{\pi}{M}\right) \qquad f_c = \min\left(\frac{f_s}{2L}, \frac{f_s}{2M}\right)$$

MATLAB: y = resample(x,L,M) resamples the sequence in vector *x at L*/*M* times the original sampling rate.

Sampling Rate Conversion



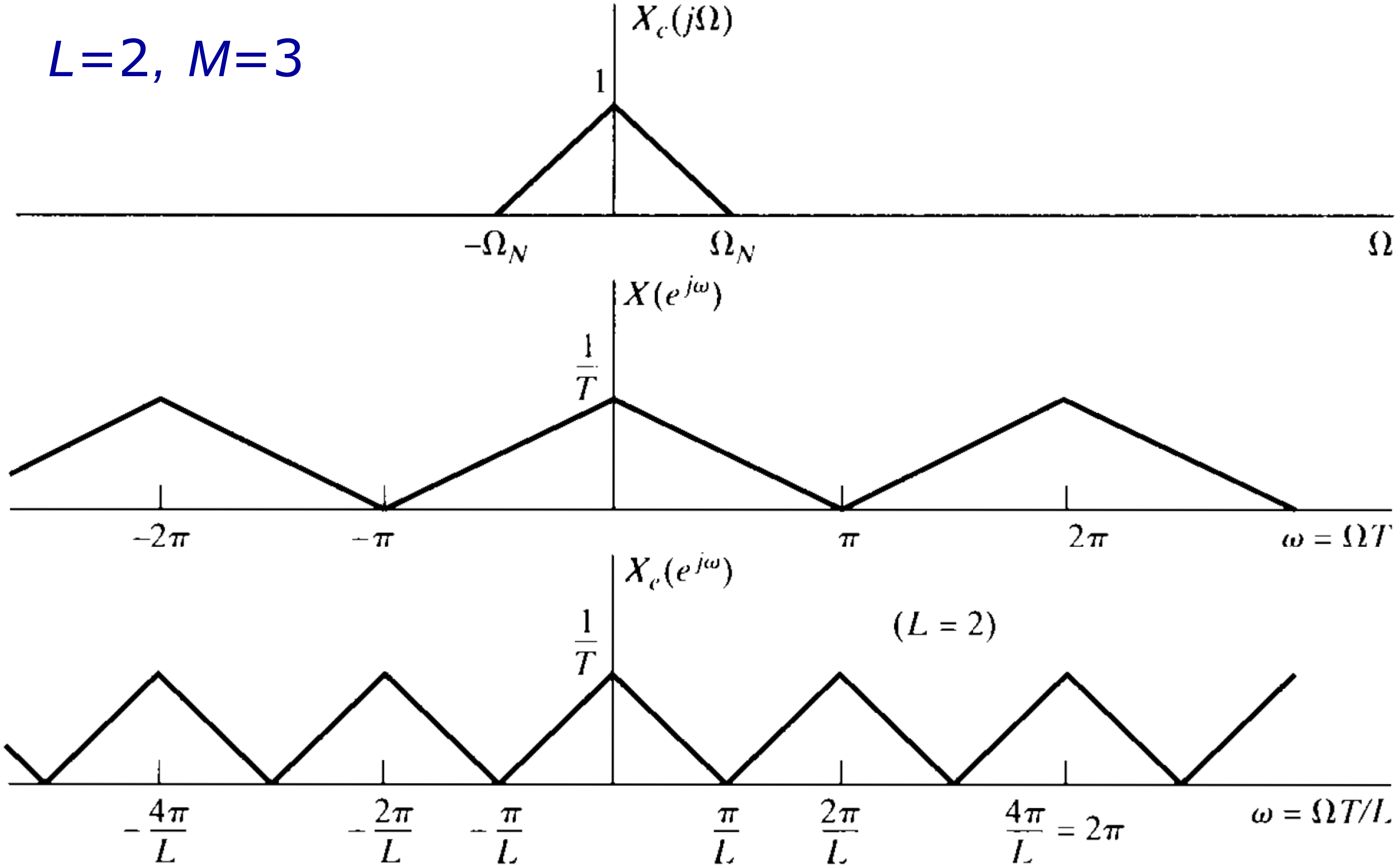
System for changing the sampling rate by a noninteger factor



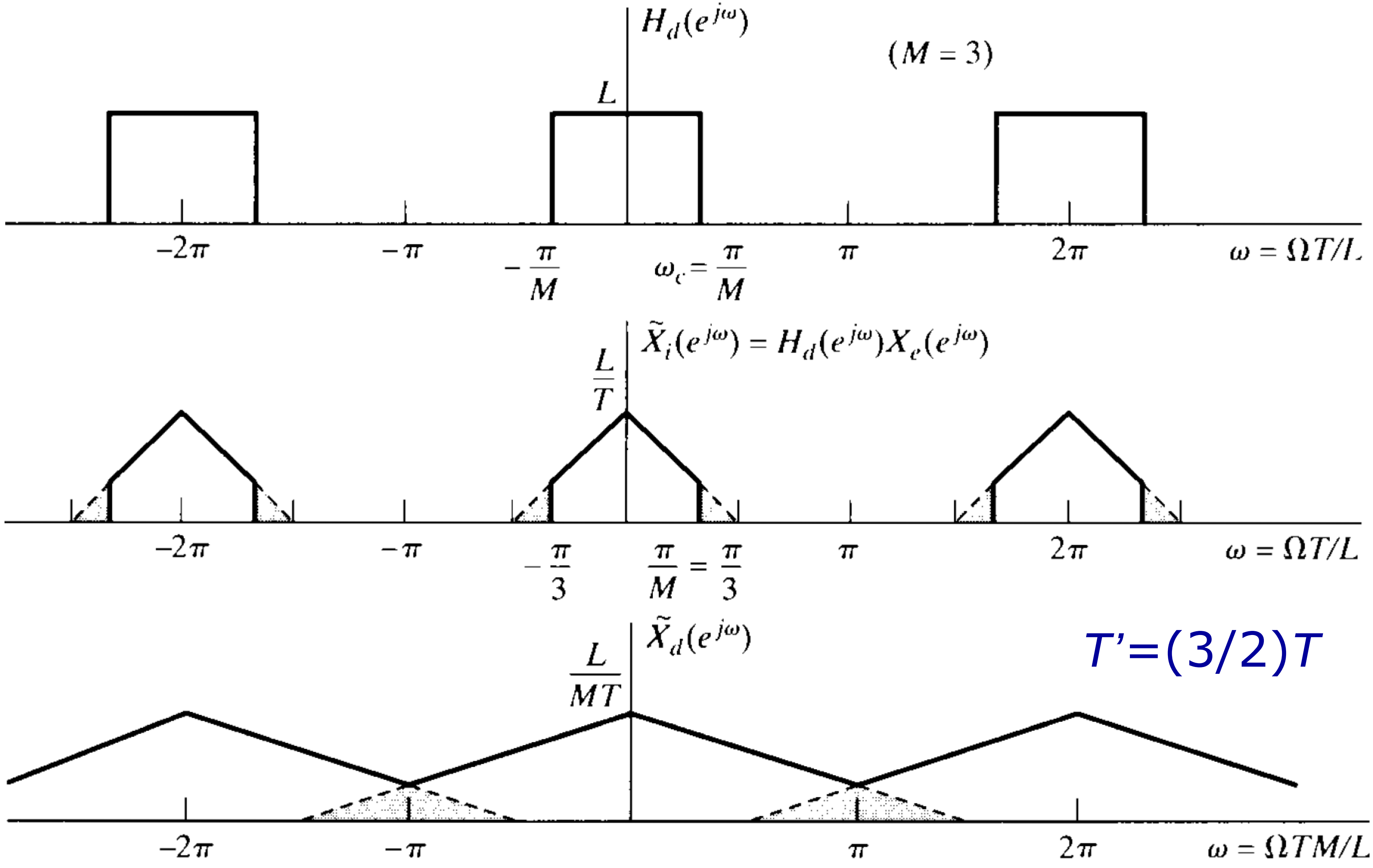
Simplified system: the decimation and interpolation filters are combined.

Frequency Domain Illustration

$L=2$, $M=3$



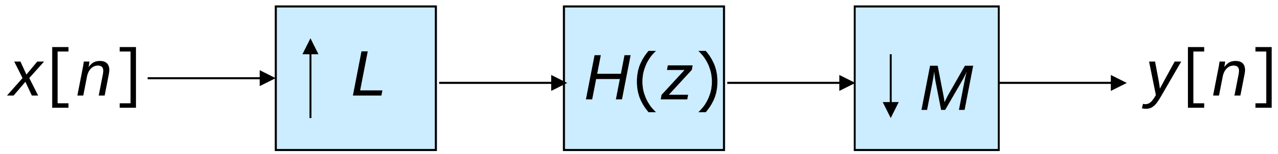
Frequency Domain Illustration



$T' = (3/2)T$

Summary

- Interpolation (Up-sampling)
- Decimation (Down-sampling)
- Sampling Rate Conversion



- Multirate Identities
- Filters in Multirate Systems