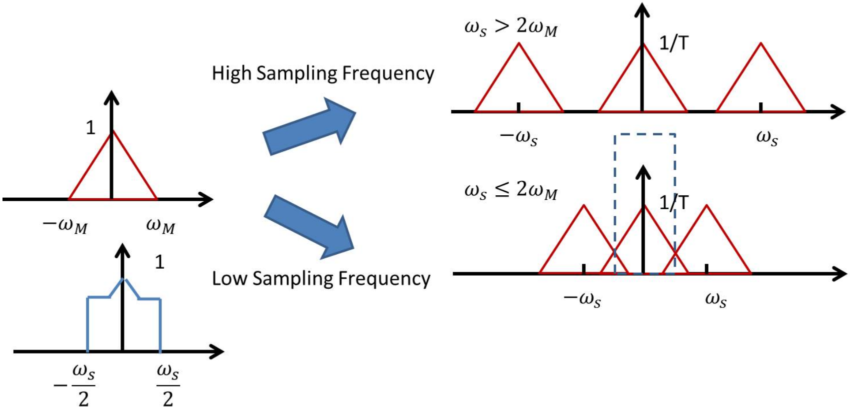
Homework: 4.50 & 4.51 of the attachment

Tutorial Problems: 7.41, 7.44, 7.47, 7.49

Undersampling & Aliasing

- Undersampling: insufficient sampling frequency $\omega_s < 2\omega_M$
- Perfect reconstruction is impossible with undersampling.
- Aliasing: distortion due to undersampling



Aliasing: Example

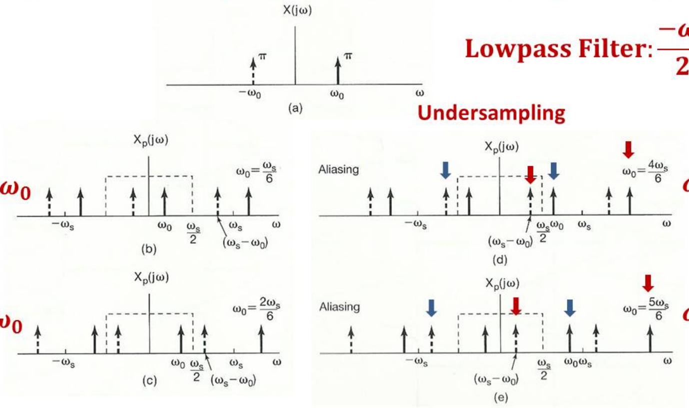
Signal before sampling: $\cos\omega_0 t$ **Sampling rate: ω_s**

Lowpass Filter: $\frac{-\omega_s}{2} \sim \frac{\omega_s}{2}$

Undersampling

$\omega_s = 6\omega_0$ $\omega_s = \frac{3}{2}\omega_0$

$\omega_s = 3\omega_0$ $\omega_s = \frac{6}{5}\omega_0$



$\cos\omega_0 t$ **Aliasing: $\cos(\omega_s - \omega_0)t$**

Low-pass filtering: Interpret the samples by cosine function with frequency lower than $\omega_s/2$

Original:

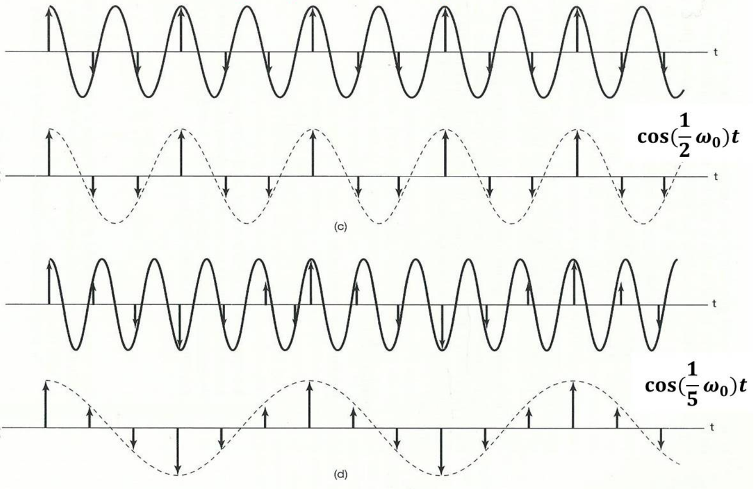
$$\omega_s = \frac{3}{2}\omega_0$$

Reconstructed:

Original:

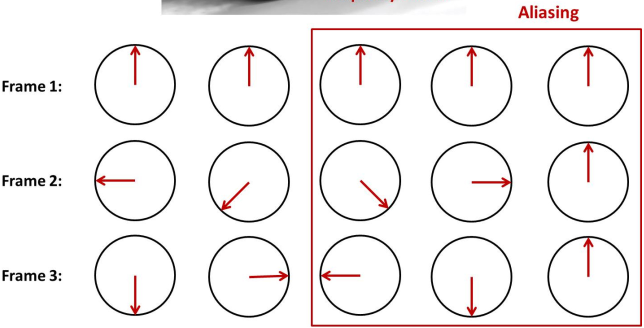
$$\omega_s = \frac{6}{5}\omega_0$$

Reconstructed:



Aliasing in Movies

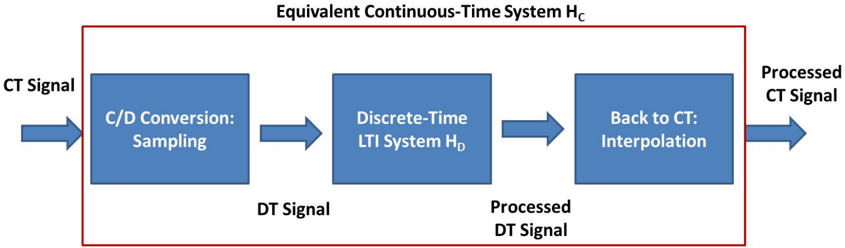
- Wheel's rotation in movies



Process Continuous-Time Signals Discretely

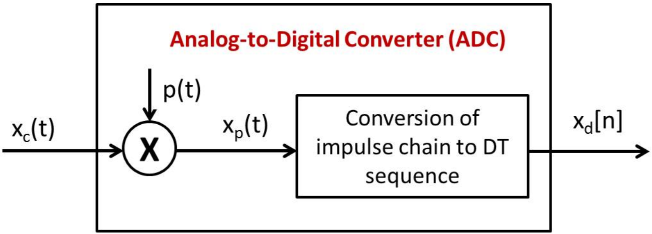
- People would like to process continuous-time signal in discrete-time (digital) domain

Block Diagram



- It is much easier to design DT system.
- What's the relation between H_C and H_D?

Discretization: C/D Conversion



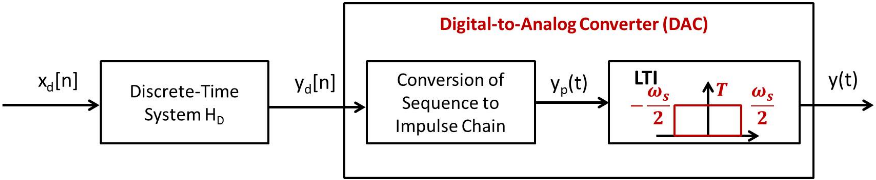
- Mathematical Interpretation (Fourier Transform)

$$x_c(t) \longleftrightarrow X_c(j\omega)$$

$$x_p(t) = \sum_{n=-\infty}^{+\infty} x_c(nT)\delta(t-nT) \longleftrightarrow X_p(j\omega) = \frac{1}{T}\sum_{k=-\infty}^{+\infty} X_c(j(\omega - k\omega_s))$$

$$x_d[n] = x_c(nT) \longleftrightarrow X_d(e^{j\omega}) = X_p(j\omega/T)$$

DT Processing and Conversion



- Mathematical Interpretation (Fourier Transform)

$$y_d[n] = x_d[n] * h_D[n] \quad \longleftrightarrow \quad Y_d(e^{j\omega}) = X_d(e^{j\omega})H_D(e^{j\omega})$$

$$y_p(t) = \sum_{n=-\infty}^{\infty} y_d[n]\delta(t - nT) \quad \longleftrightarrow \quad Y_p(j\omega) = Y_d(e^{j\omega T})$$

$$y(t) = y_p(t) * h_{LP}(t) \quad \longleftrightarrow \quad Y(j\omega) = Y_p(j\omega)H_{LP}(j\omega)$$

Input vs. Output

$$
\begin{aligned}
Y(j\omega) &= Y_p(j\omega)H_{LP}(j\omega) = Y_d(e^{j\omega T})H_{LP}(j\omega) \\
&= X_d(e^{j\omega T})H_D(e^{j\omega T})H_{LP}(j\omega) \\
&= X_p(j\omega)H_D(e^{j\omega T})H_{LP}(j\omega) \\
&= \left[\frac{1}{T}\sum_{k=-\infty}^{+\infty} X_c(j(\omega - k\omega_s))\right] H_D(e^{j\omega T})H_{LP}(j\omega) \\
&= X_c(j\omega)H_D(e^{j\omega T}) \\
&= X_c(j\omega)\widetilde{H}_D(e^{j\omega T}) \qquad (1)
\end{aligned}
$$

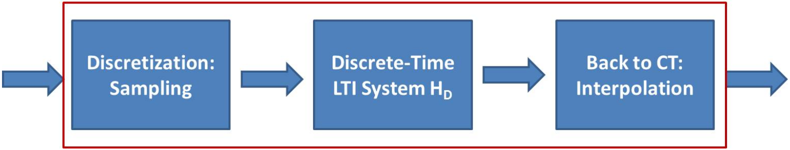
where

$$
\widetilde{H}_D(e^{j\omega T}) = \begin{cases} H_D(e^{j\omega T}) & |\omega| < \omega_s/2 \\ 0 & otherwise \end{cases} \qquad (2)
$$

- It is equivalent to a continuous-time LTI system $H_C(j\omega) = \widetilde{H}_D(e^{j\omega T})$
- $H_D(e^{j\omega T})$ is a periodic extension of $\widetilde{H}_D(e^{j\omega T})$ with period $\omega_s = 2\pi/T$

System Design

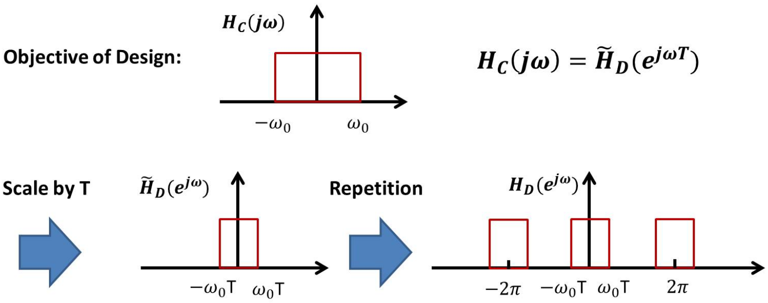
$$H_C(j\omega) = \widetilde{H}_D(e^{j\omega T})$$



- How can we design a CT LTI system with frequency response H_C via DT LTI system?
- Step 1: Sampling frequency ω_s or $2\pi/T$ should be larger than Nyquist rate
- Step 2: $\widetilde{H}_D(e^{j\omega T}) = H_C(j\omega)$
- Step 3: Frequency response of DT LTI system
  $H_D(e^{j\omega T}) = \sum_{k=-\infty}^{\infty} \widetilde{H}_D(e^{j(\omega - k\omega_s)T})$ or
  $H_D(e^{j\omega}) = \sum_{k=-\infty}^{\infty} \widetilde{H}_D(e^{j(\omega - k\omega_s T)}) = \sum_{k=-\infty}^{\infty} H_C(j\frac{\omega - 2k\pi}{T})$

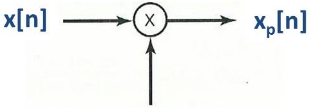
System Design Example

- How to implement an ideal CT lowpass filter?



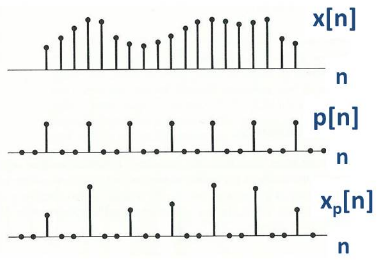
Sampling on Discrete-Time Signals

- In digital signal processing, we may sample the discrete-time signals
- Example: image, audio, video compression
- Its mathematical model is give below:



$$\mathbf{p[n]} = \sum_{\mathbf{k}=-\infty}^{+\infty} \delta[\mathbf{n}-\mathbf{kN}]$$

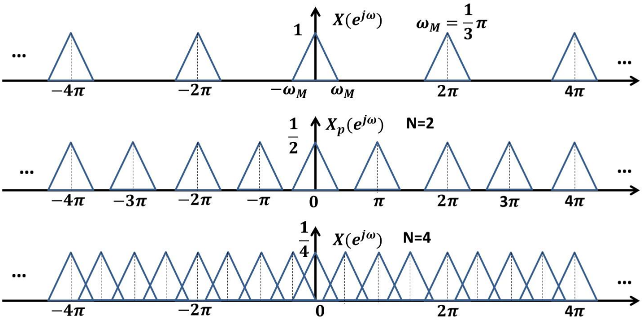
$$\mathbf{x_p[n]} = \sum_{\mathbf{k}=-\infty}^{+\infty} x[kN]\delta[\mathbf{n}-\mathbf{kN}]$$



Frequency Analysis

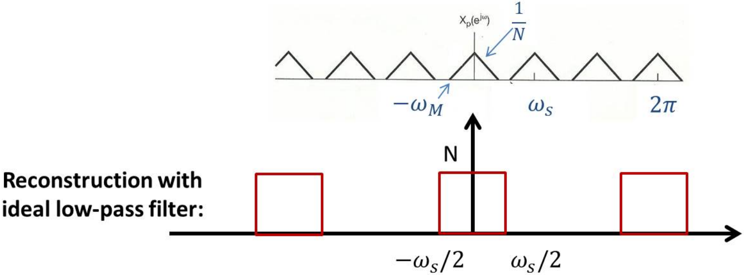
$$P(e^{j\omega}) = \frac{2\pi}{N}\sum_{k=-\infty}^{\infty}\delta(\omega - k\omega_s) \quad where\ \omega_s = 2\pi/N$$

$$X_p(e^{j\omega}) = \frac{1}{2\pi}\int_{2\pi} P(e^{j\theta})X(e^{j(\omega-\theta)})d\theta = \frac{1}{N}\sum_{k=0}^{N-1} X(e^{j(\omega-k\omega_s)})$$



Reconstruction

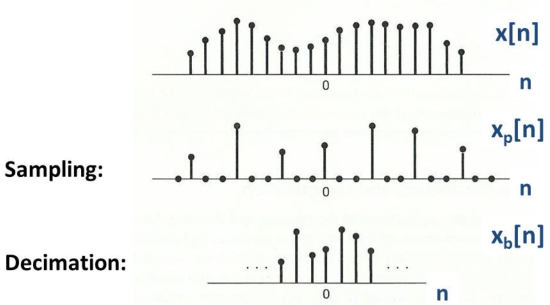
- Perfect reconstruction is applicable when $\omega_s > 2\omega_M \leftrightarrow N < \frac{\pi}{\omega_M}$



- Aliasing occurs when $\omega_s < 2\omega_M$

Decimation

- After sampling, there will be a great amount of redundancy
- Decimation: discrete-time sampling + remove zeros



$$x_b[n] = x_p[nN] = x[nN]$$

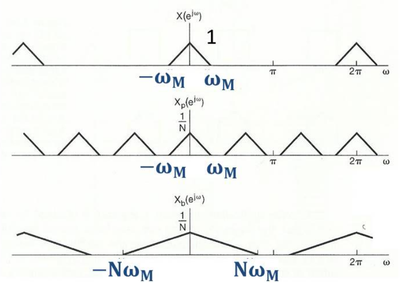
Frequency Analysis

$$X_b(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x_b[n]e^{-j\omega n} = \sum_{n=-\infty}^{\infty} x_p[n]e^{-j\omega n/N} = X_p(e^{j\omega/N})$$

$$= \frac{1}{N}\sum_{k=0}^{N-1} X(e^{j\omega/N - k\omega_s})$$

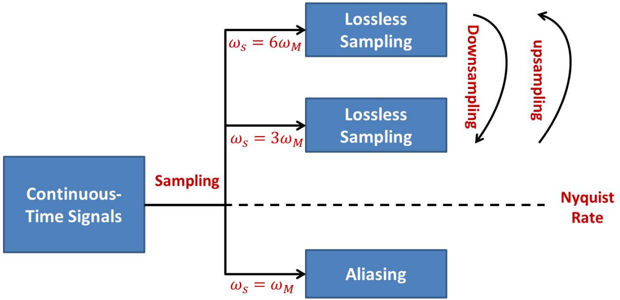
Sampling:

Decimation:



Condition for Perfect Reconstruction: $N\omega_M < \pi$

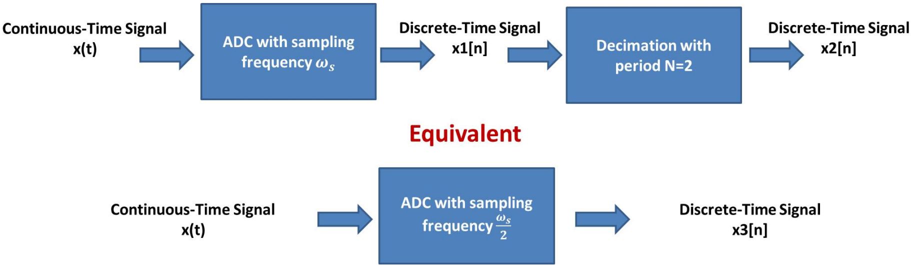
Anything Else?



- Downsampling: to reduce the sampling frequency (decimation)
- Upsampling: to generate a DT signal with higher sampling frequency
- As long as Nyquist rate is satisfied, the transform between low-sampling-frequency version and high-sampling-frequency versions is lossless.

Downsampling

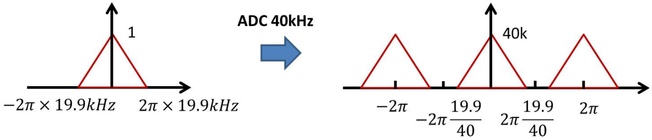
- Downsampling: a general procedure to reduce the sampling frequency



When do we need downsampling?

Downsampling Example (1/2)

- Suppose we have a clip of voice, $x(t)$, with bandwidth =19.9kHz
- It can be converted to DT signal with sampling frequency 40kHz, denoted as $x_1[n]$

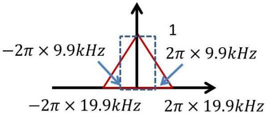


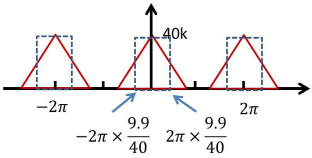
- Based on $x_1[n]$, if we want to save the voice information within 9.9kHz into another DT signal, what can we do?

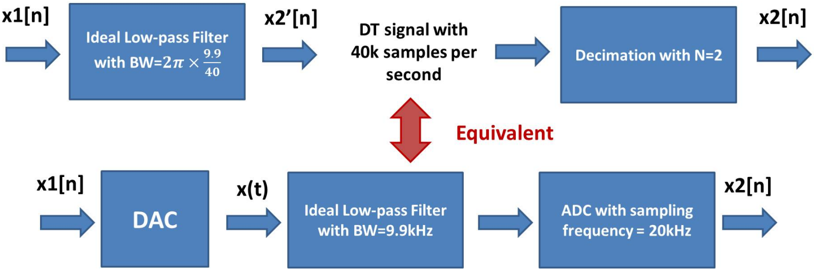
One Choice:

How can we generate x2[n] in discrete-time domain?

Downsampling Example (2/2)

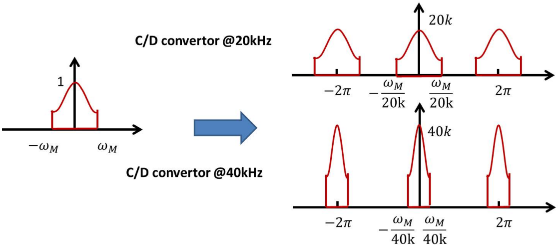






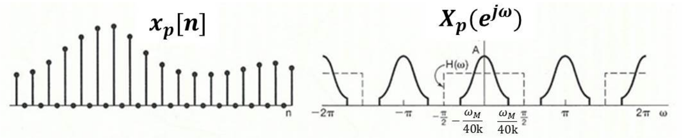
Upsampling

- Upsampling: a procedure to generate a sequence with higher sampling frequency
- Superpose the following two digital sound clips
  - Audio clip 1: Bandwidth= 19.9kHz, sampled at 40kHz
  - Audio clip 2: Bandwidth= 9.9kHz, sampled at 20kHz
- Double the sampling frequency of audio clip 2 (40kHz)
- How to do upsampling in discrete-time domain?



- Time expansion (Insert zeros):

$$x_p[n] = x_{b(2)}[n] \longleftrightarrow X_p(e^{j\omega}) = X_b(e^{j2\omega})$$



- Low-pass filtering:

Ideal Low-Pass Filter: