Signals and Systems

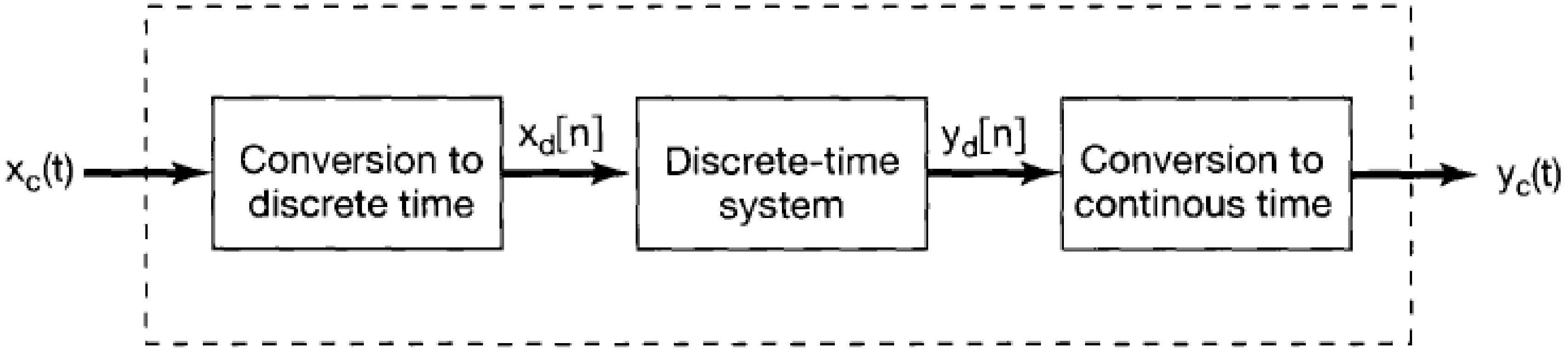
Lecture 10

Discrete-time processing of continuous-time signals

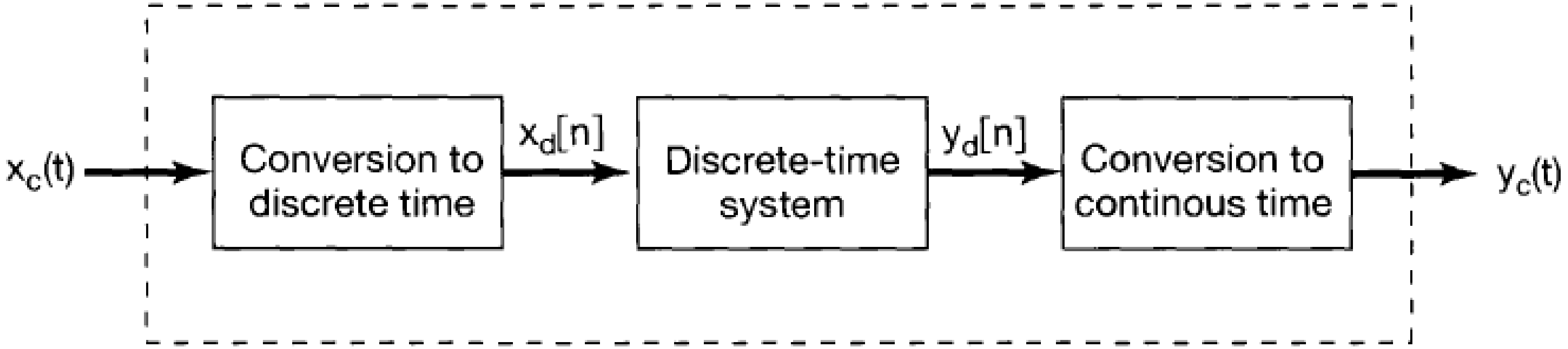
Discrete-time processing of continuous-time signals

In many applications, we process a continuous-time signal by first converting it to a discrete-time signal and, after discrete-time processing, convert it back to a continuous-time signal.

The discrete-time signal processing can be implemented with a general- or special-purpose computer, with microprocessors, or with any of the variety of devices that are specifically designed for discrete-time signal processing.



Discrete-time processing of continuous-time signals

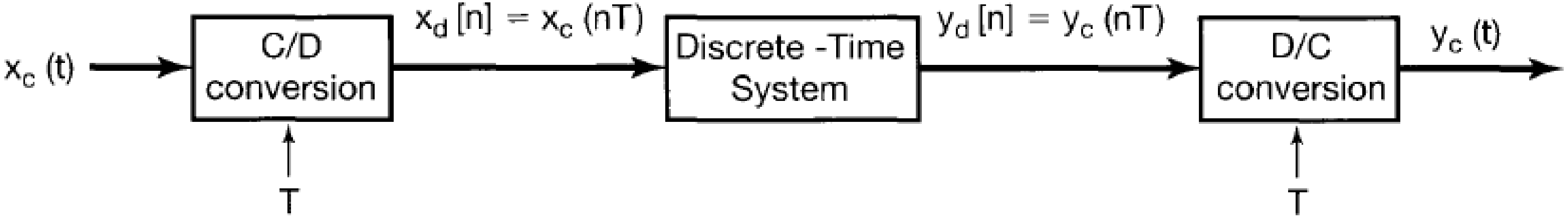


$x_c(t)$ and $y_c(t)$ are continuous-time signals and
$x_d[n]$ and $y_d[n]$ are the discrete-time signals corresponding to $x_c(t)$ and $y_c(t)$.

The overall system is a continuous-time system in the sense that its input and output are both continuous-time signals.

The theoretical basis for converting a continuous-time signal to a discrete-time signal and reconstructing a continuous-time signal from its discrete-time representation lies in the sampling theorem.

Discrete-time processing of continuous-time signals

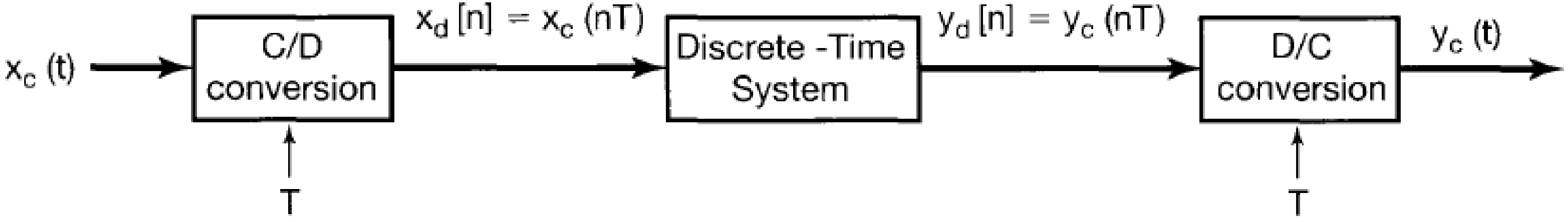


C/D : ***continuous-to-discrete-time conversion***

Through the process of periodic sampling with the sampling frequency consistent with the conditions of the sampling theorem, the continuous-time signal $x_c(t)$ is exactly represented by a sequence of instantaneous sample values $x_c(nT)$; that is, the discrete-time sequence $x_d[n]$ is related to $x_c(t)$ by

$$x_d[n] = x_c(nT).$$

Discrete-time processing of continuous-time signals



C/D : ***continuous-to-discrete-time conversion***

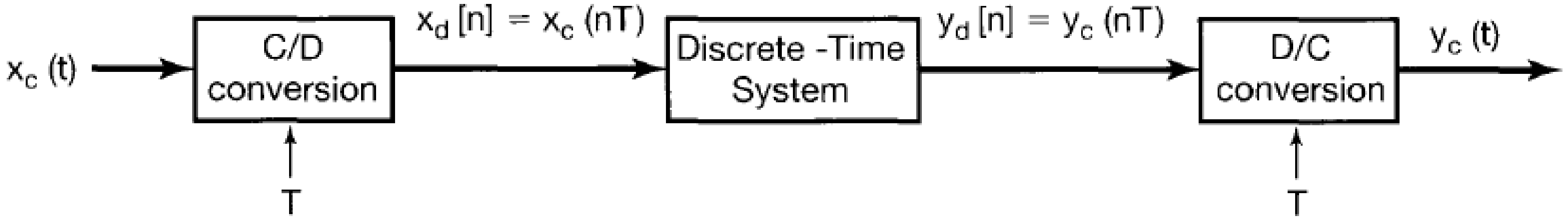
$$x_d[n] = x_c(nT).$$

D/C : ***discrete-to-continuous-time conversion***

The D/C operation performs an interpolation between the sample values provided to it as input. That is, the D/C operation produces a continuous-time signal $y_c(t)$ which is related to the discrete-time signal $y_d[n]$ by

$$y_d[n] = y_c(nT).$$

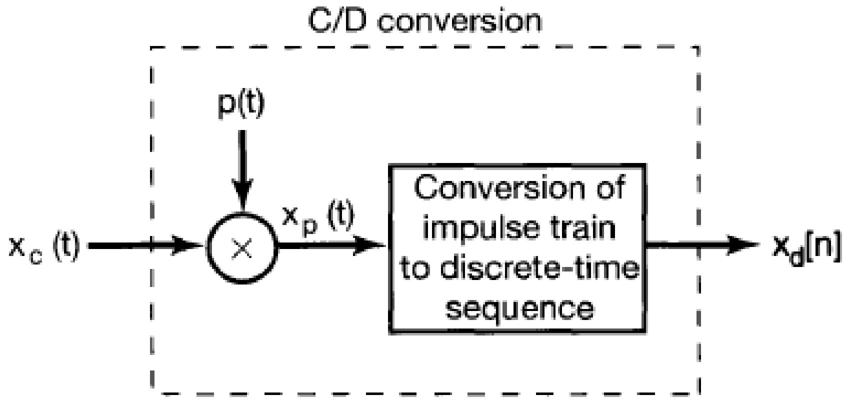
Discrete-time processing of continuous-time signals



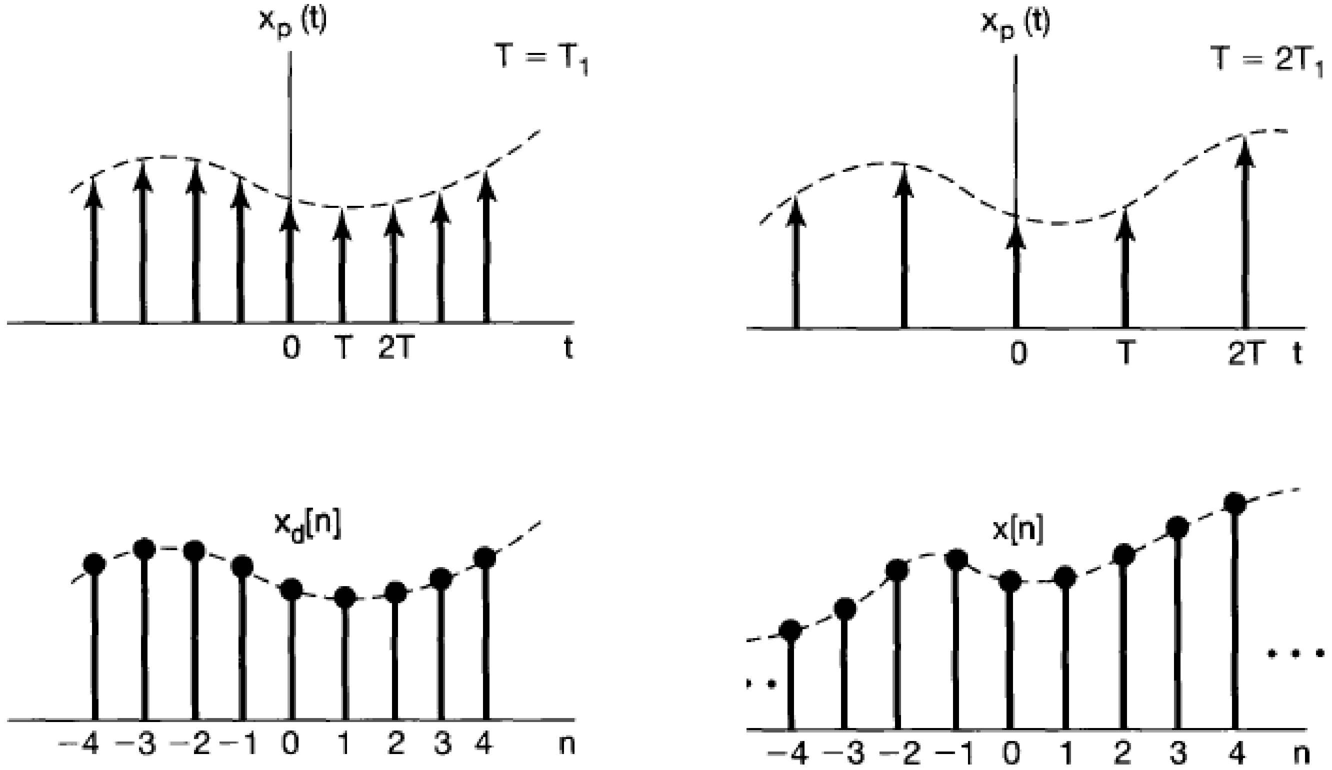
In systems such as digital computers and digital systems for which the discrete-time signal is represented in digital form, the device commonly used to implement the C/D conversion is referred to as an analog-to-digital (A-to-D) converter, and the device used to implement the D/C conversion is referred to as a digital-to-analog (D-to-A) converter.

Discrete-time processing of continuous-time signals

We can represent C/D as a process of periodic sampling followed by a mapping of the impulse train to a sequence.



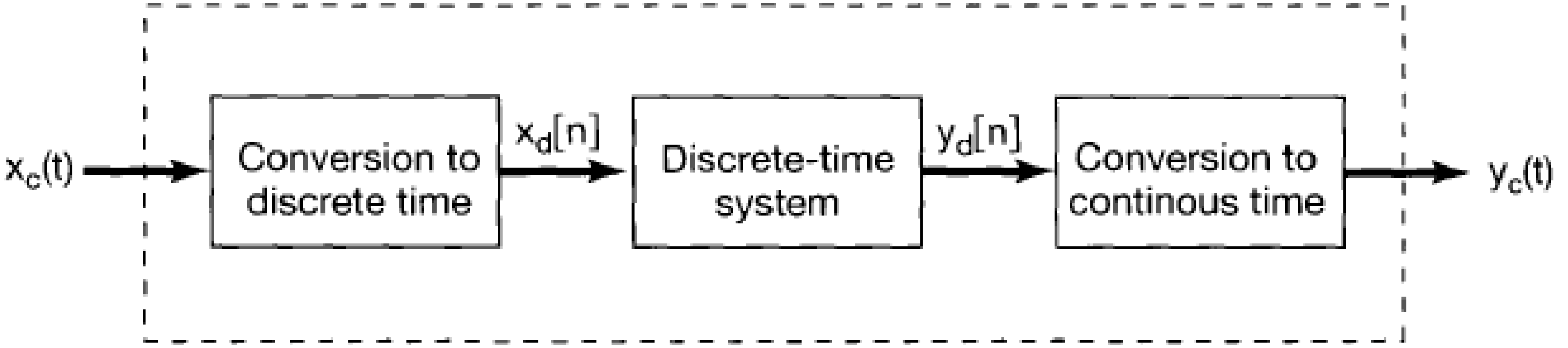
$x_p(t)$ and $x_d[n]$ for two different sampling rates. The dashed envelope represents $x_c(t)$.



$x_p(t)$: A sequence of impulses with amplitudes corresponding to the samples of $x_c(t)$ and with a time spacing equal to the sampling period T.

$x_d[n]$: A sequence of the samples with unity spacing in terms of the new independent variable n.

Discrete-time processing of continuous-time signals



We will examine this system in frequency domain.
To distinguish the continuous-time and discrete-time frequency variables by using ω in continuous time and Ω in discrete time.

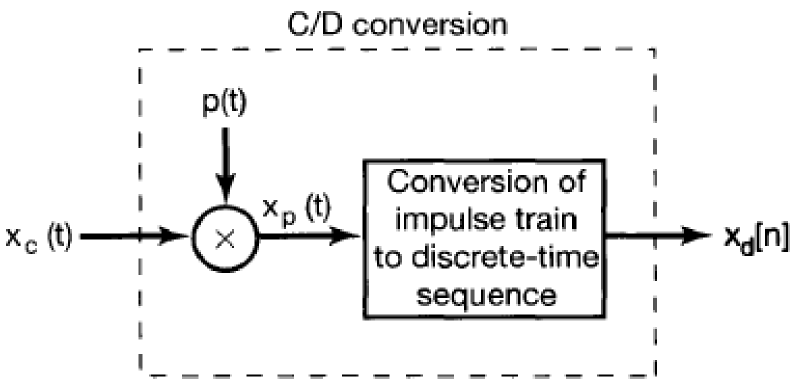
$X_c(j\omega)$: The continuous-time Fourier transform of $x_c(t)$
$Y_c(j\omega)$: The continuous-time Fourier transform of $y_c(t)$

$X_d(e^{j\Omega})$: The discrete-time Fourier transform of $x_d[n]$
$Y_d(e^{j\Omega})$: The discrete-time Fourier transform of $y_d[n]$

Discrete-time processing of continuous-time signals

$$\delta(t - nT) \longleftrightarrow e^{-j\omega nT}$$

$$x_p(t) = \sum_{n=-\infty}^{+\infty} x_c(nT)\delta(t - nT) \longleftrightarrow X_p(j\omega) = \sum_{n=-\infty}^{+\infty} x_c(nT)e^{-j\omega nT}.$$



Now consider the discrete-time Fourier transform of $x_d[n]$:

$$X_d(e^{j\Omega}) = \sum_{n=-\infty}^{+\infty} x_d[n]e^{-j\Omega n},$$

Since $x_d[n] = x_c(nT).$

$$X_d(e^{j\Omega}) = \sum_{n=-\infty}^{+\infty} x_c(nT)e^{-j\Omega n}.$$

Discrete-time processing of continuous-time signals

$$X_p(j\omega) = \sum_{n=-\infty}^{+\infty} x_c(nT)e^{-j\omega nT}.$$

$$X_d(e^{j\Omega}) = \sum_{n=-\infty}^{+\infty} x_c(nT)e^{-j\Omega n}.$$

Comparing the two equations:

$$X_d(e^{j\Omega}) = X_p(j\Omega/T)$$

Also, recall that

$$X_p(j\omega) = \frac{1}{T}\sum_{k=-\infty}^{+\infty} X_c(j(\omega - k\omega_s)).$$

As a result $X_c(j\omega)$ and $X_d\left(e^{j\Omega}\right)$ are related by

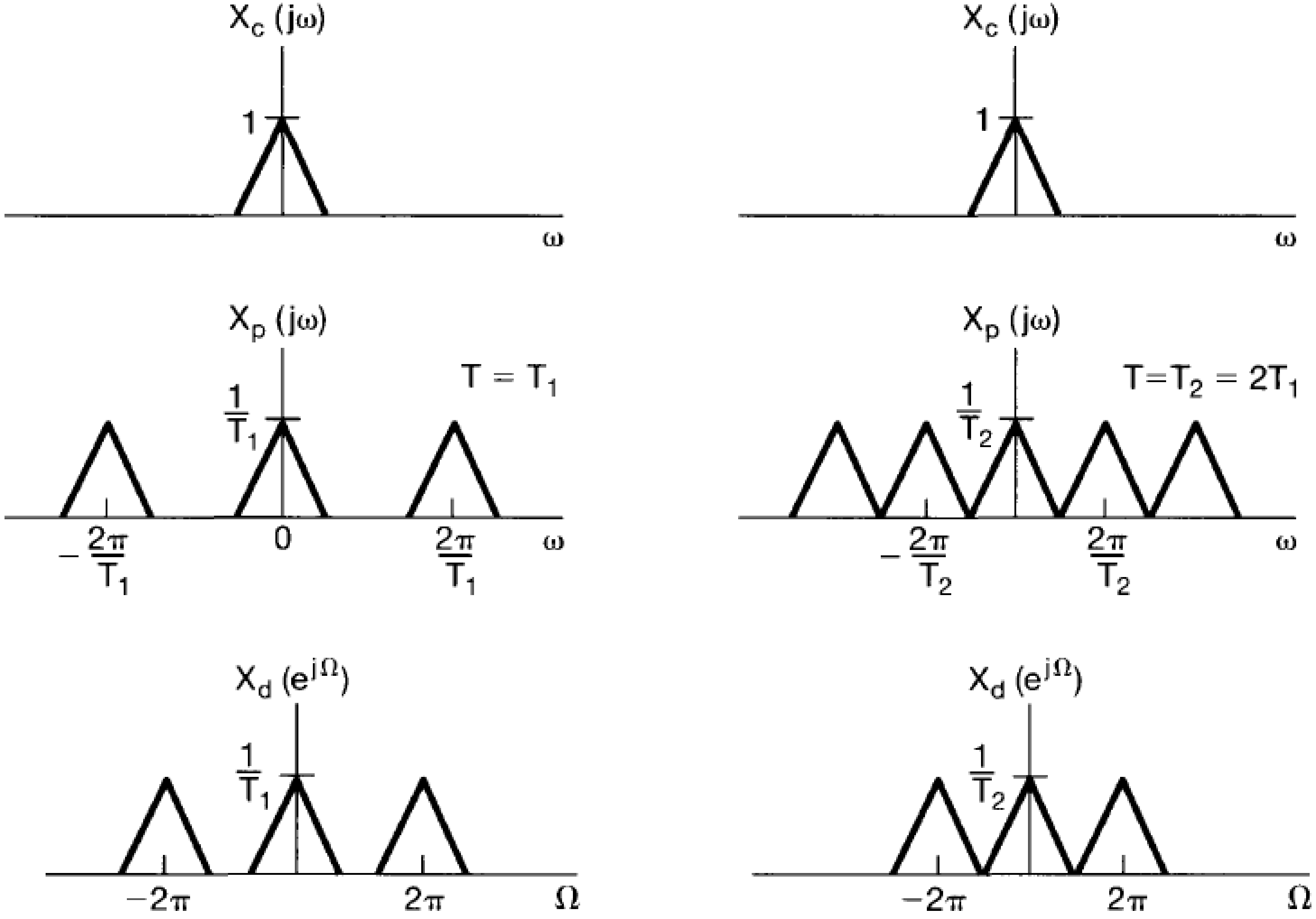
$$X_d(e^{j\Omega}) = \frac{1}{T}\sum_{k=-\infty}^{+\infty} X_c(j(\Omega - 2\pi k)/T)$$

Discrete-time processing of continuous-time signals

Relationship between $X_c(j\omega)$, $X_p(j\omega)$ and $X_d\left(e^{j\Omega}\right)$ for two different sampling rates:

Here, $X_d\left(e^{j\Omega}\right)$ is a frequency-scaled version of $X_p(j\omega)$

$X_d\left(e^{j\Omega}\right)$ is periodic in Ω with period 2π (as are all discrete-time Fourier transforms).



Discrete-time processing of continuous-time signals

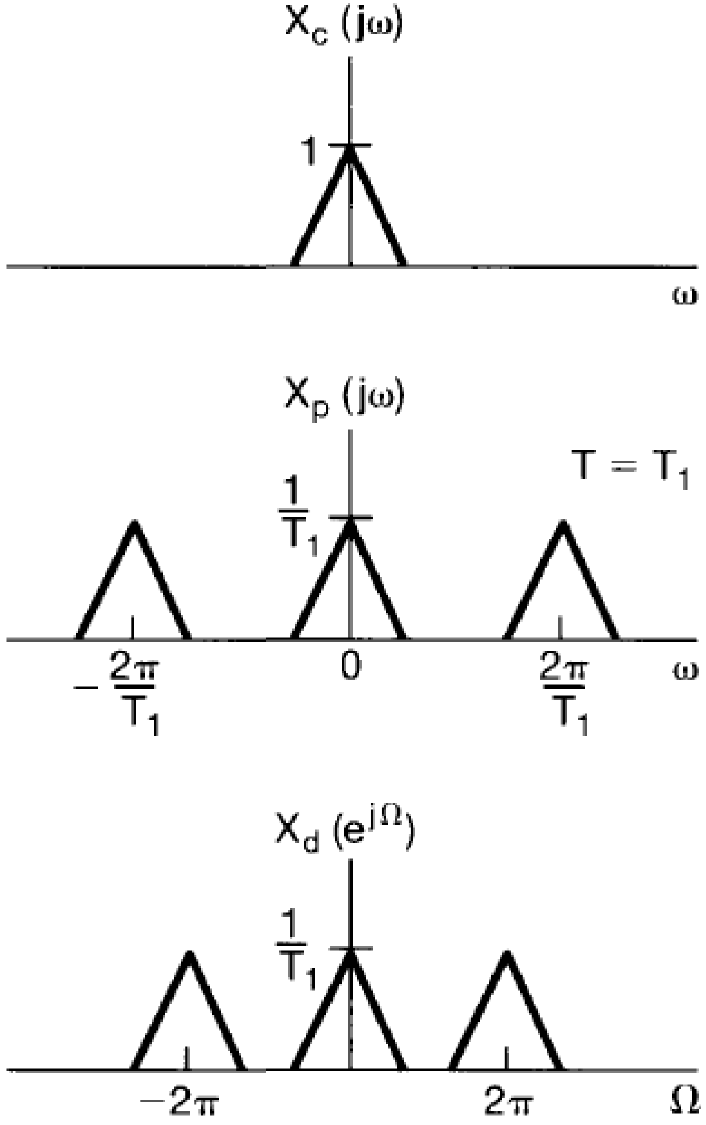
$X_d(e^{j\Omega})$ is a frequency-scaled version of $X_p(j\omega)$

$X_d(e^{j\Omega})$ is periodic in Ω with period 2π (as are all discrete-time Fourier transforms).

The spectrum of $x_d[n]$ is related to that of $x_c(t)$ through periodic replication, followed by linear frequency scaling.

The periodic replication is a consequence of the impulse-train sampling.

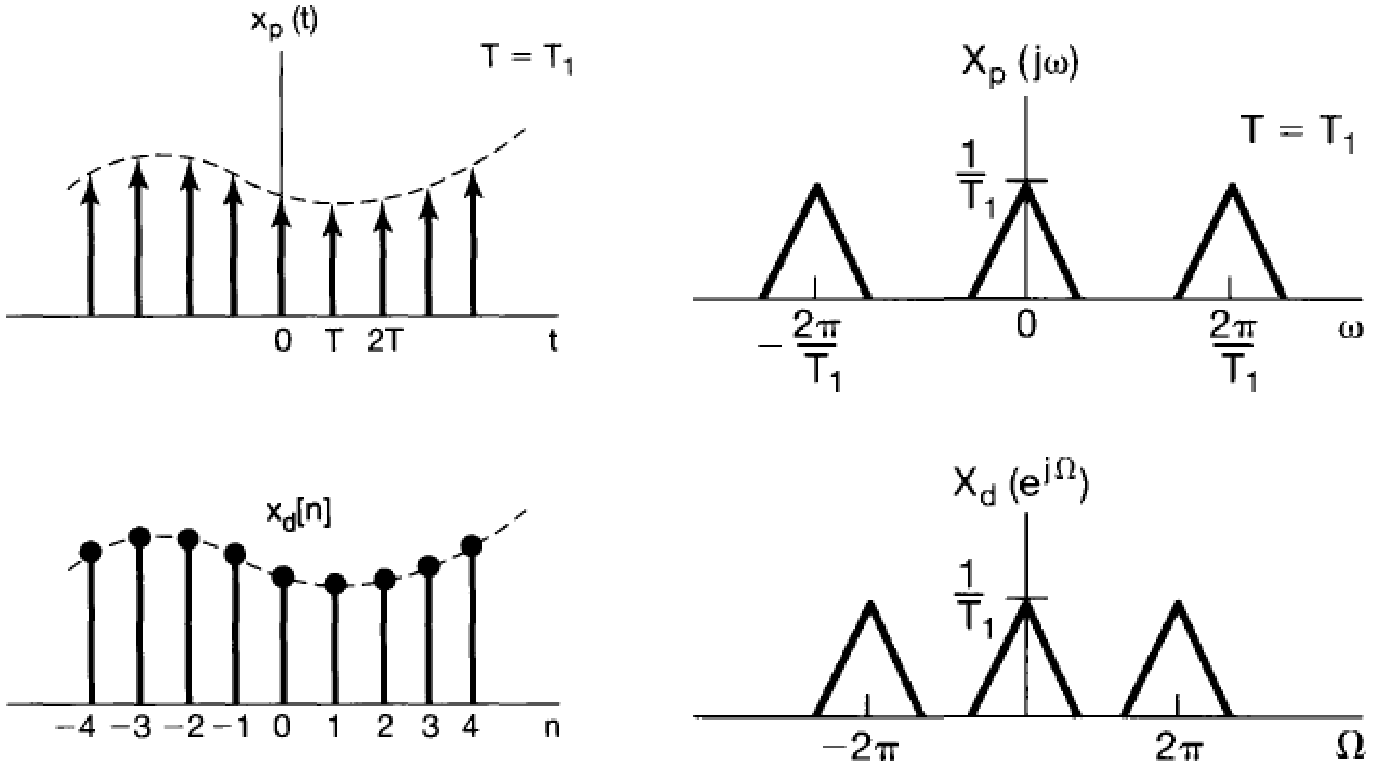
The linear frequency scaling can be thought of informally as a consequence of the normalization in time introduced by converting from the impulse train $x_p(t)$ to the discrete-time sequence $x_d[n]$.



Discrete-time processing of continuous-time signals

The linear frequency scaling can be thought of informally as a consequence of the normalization in time introduced by converting from the impulse train $x_p(t)$ to the discrete-time sequence $x_d[n]$.

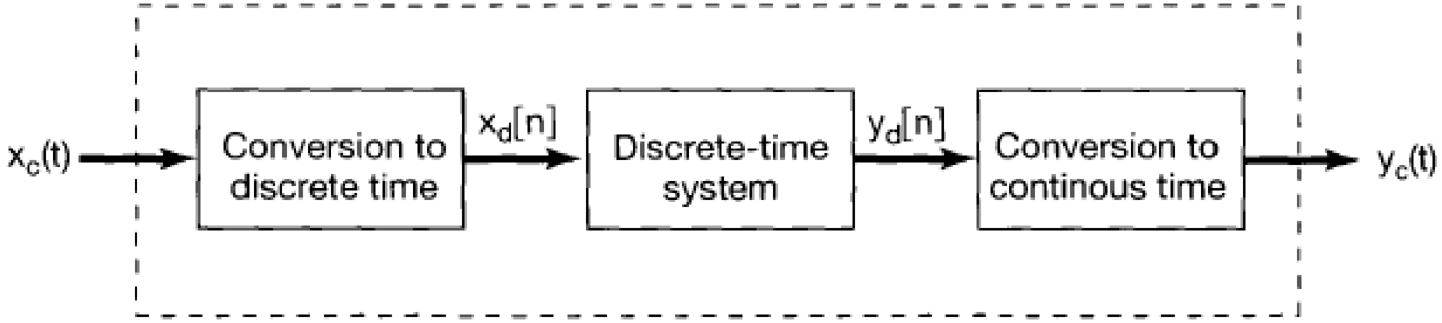
$$X_d(e^{j\Omega}) = X_p(j\Omega/T)$$



From the time-scaling property of the Fourier transform, scaling of the time axis by $1/T$ will introduce a scaling of the frequency axis by T.

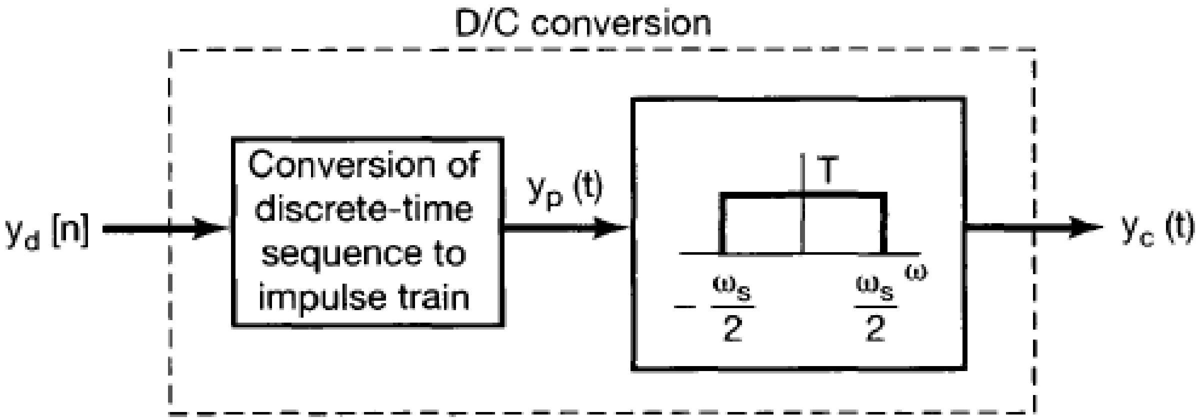
Thus, the relationship $\Omega = \omega T$ is consistent with the notion that, in converting from $x_p(t)$ to the discrete-time sequence $x_d[n]$, the time axis is scaled by $1/T$.

Discrete-time processing of continuous-time signals



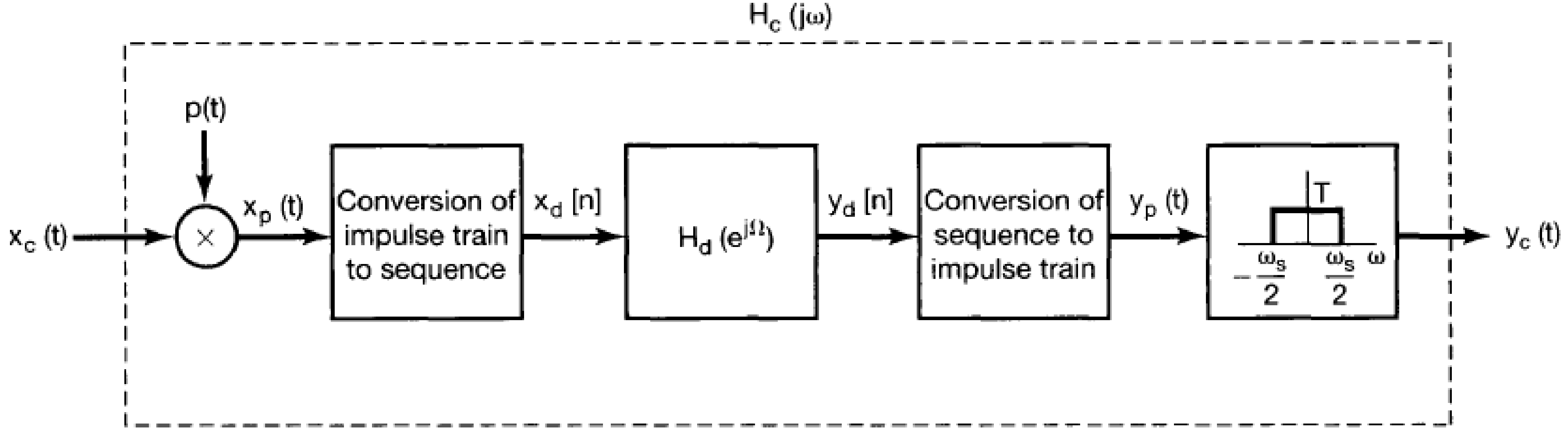
In the overall system, after processing with a discrete-time system, the resulting sequence is converted back to a continuous-time signal.

From the sequence $y_d[n]$, a continuous-time impulse train $y_p(t)$ can be generated.
Recovery of the continuous-time signal $y_c(t)$ from this impulse train is then accomplished by means of lowpass filtering



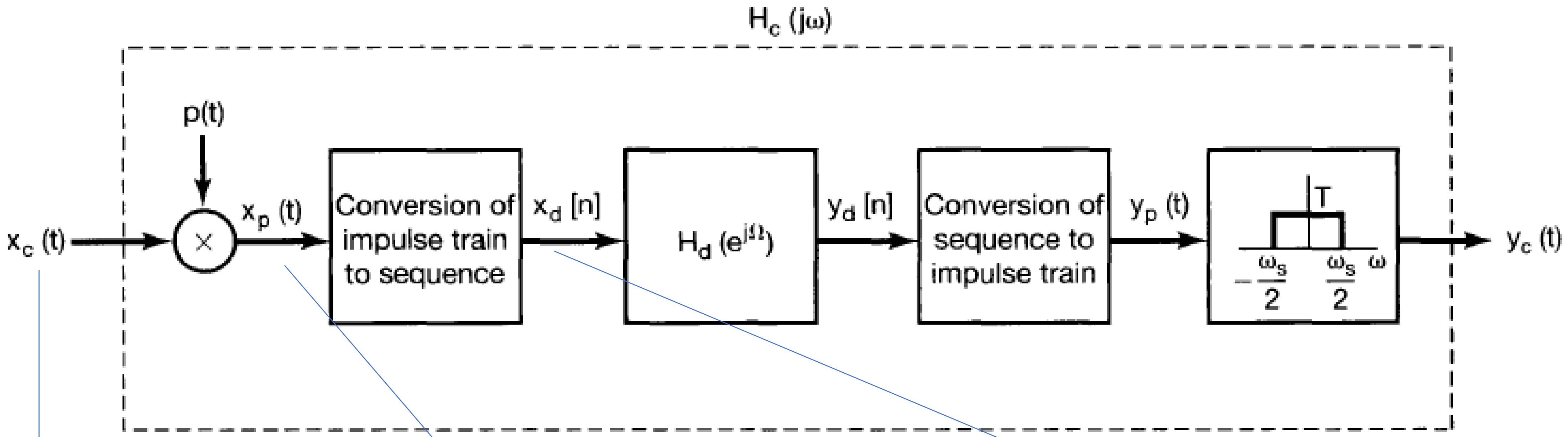
Discrete-time processing of continuous-time signals

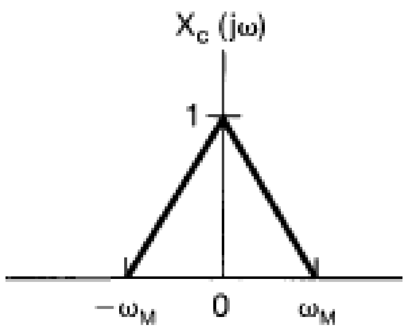
Overall system for filtering a continuous-time signal using a discrete-time filter:



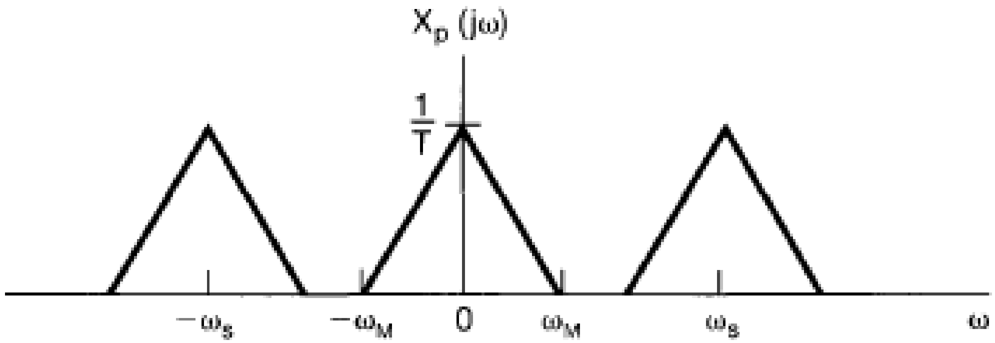
We assume that $\omega_M < \omega_s/2$, so that there is no aliasing.
We will examine the characteristics of the overall continuous-time system $H_c(j\omega)$ in relation to the frequency response of the discrete-time filter $H_d(e^{j\Omega})$.

Discrete-time processing of continuous-time signals

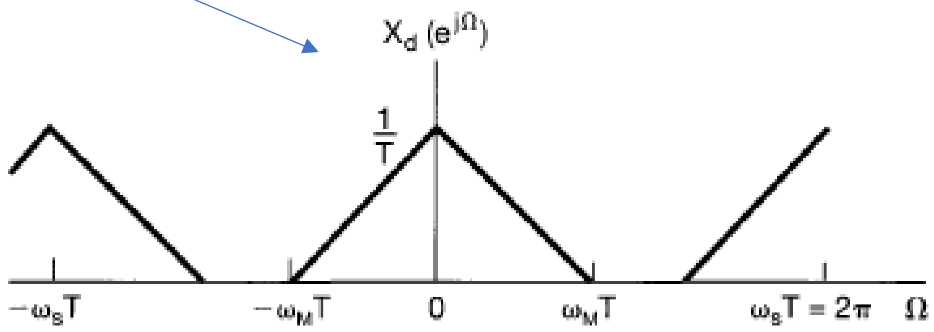




$X_c(j\omega)$: Spectrum of the input CT signal

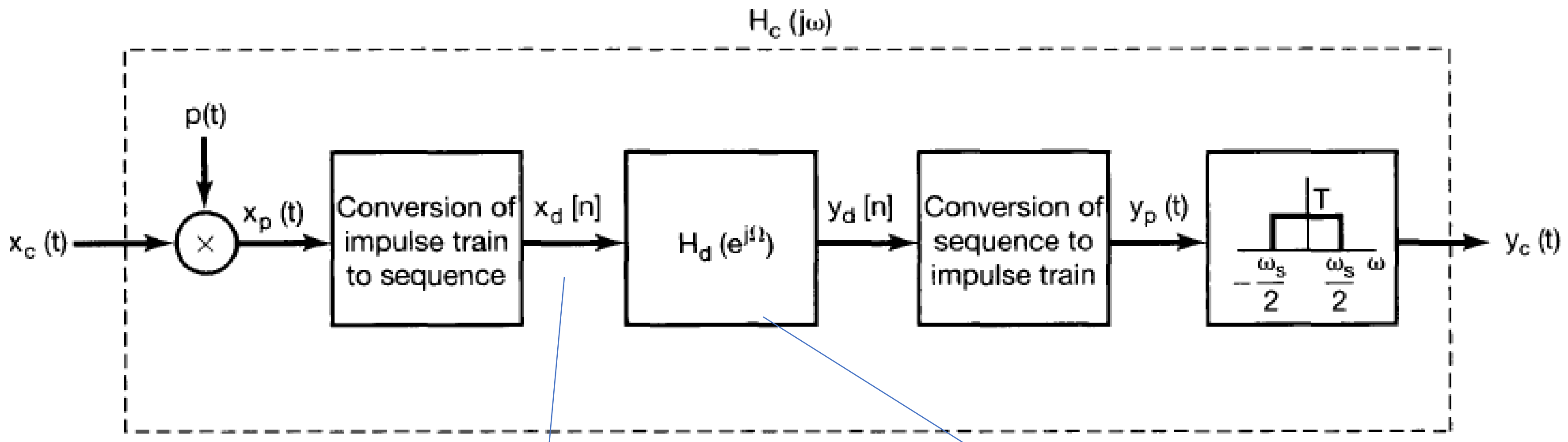


$X_p(j\omega)$: Spectrum of the signal after impulse train sampling.



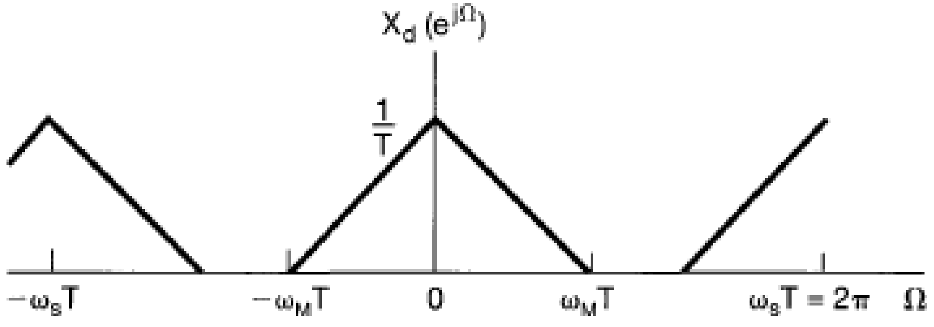
$X_d\left(e^{j\Omega}\right)$: Spectrum of the discrete-time sequence

Discrete-time processing of continuous-time signals

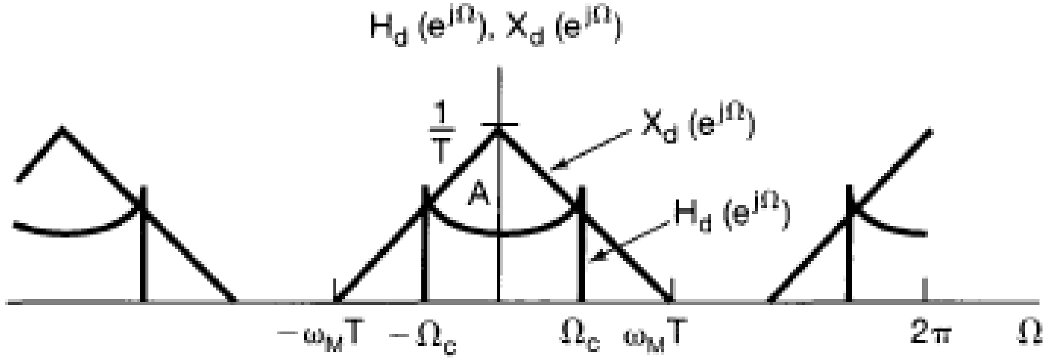


The spectrum of $y_d[n]$:

$Y_d(e^{j\Omega}) = H_d(e^{j\Omega})\, X_d(e^{j\Omega})$



$X_d(e^{j\Omega})$: Spectrum of the discrete-time sequence

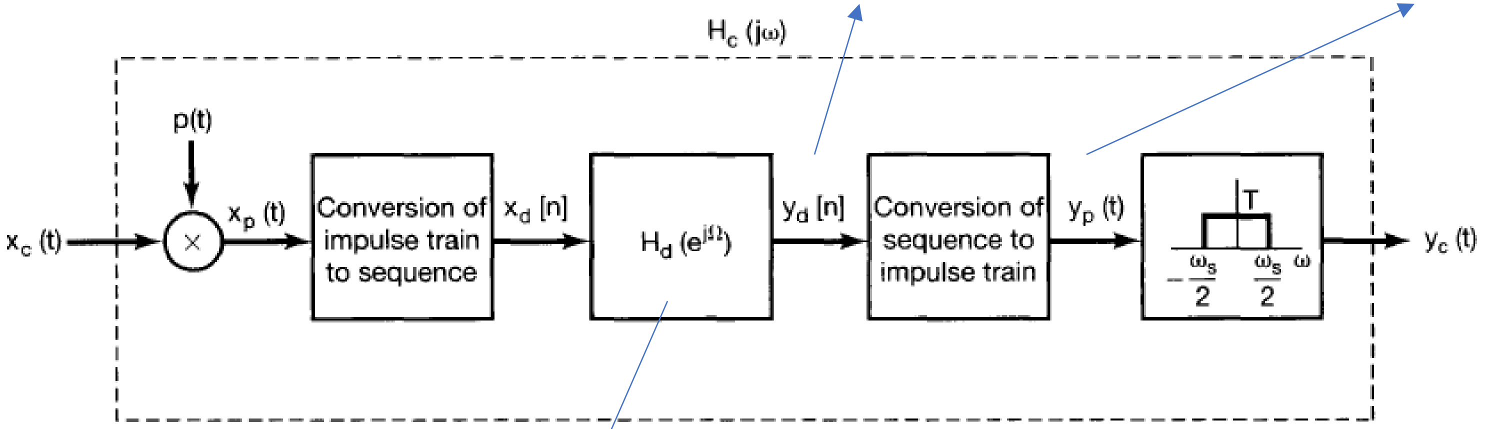


Discrete-time filtering of the sequence with a general frequency response $H_d(e^{j\Omega})$

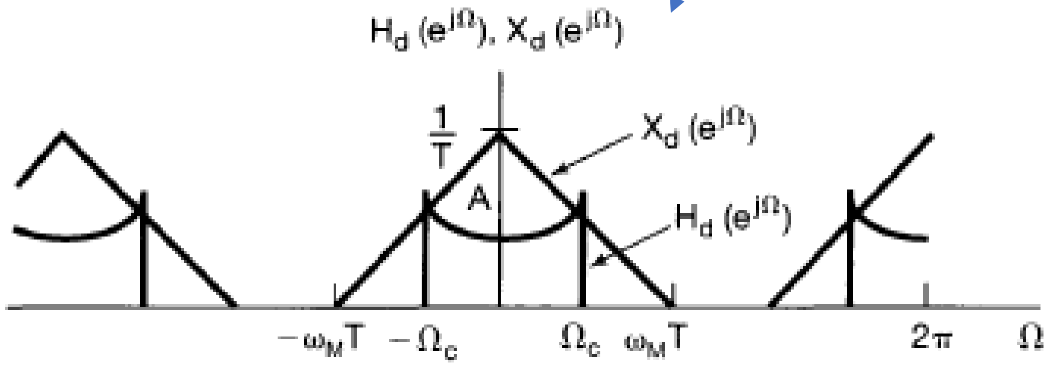
Discrete-time processing of continuous-time signals

The spectrum of $y_d[n]$:
$Y_d(e^{j\Omega}) = X_d(e^{j\Omega})\, H_d(e^{j\Omega})$

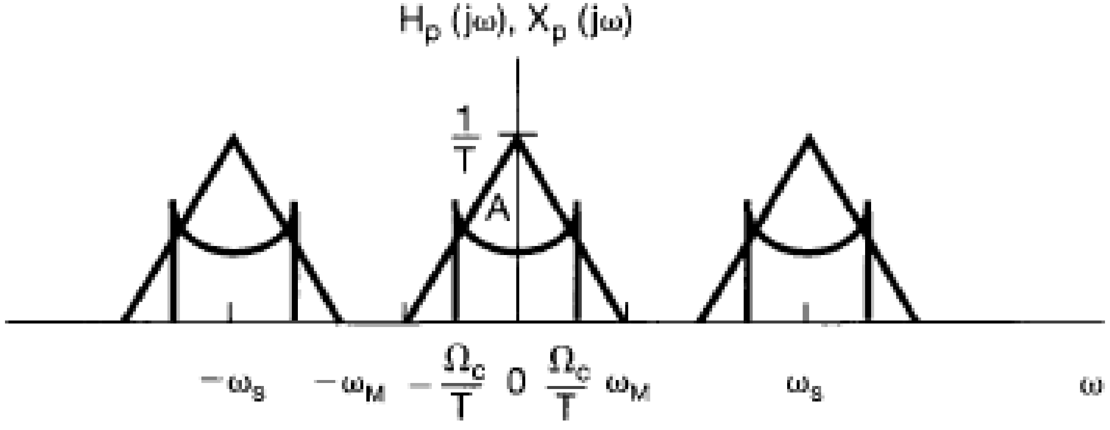
The spectrum of $y_p(t)$:
$Y_p(j\omega) = X_p(j\omega)\, H_p(j\omega)$



Applying frequency scaling $\Omega = \omega T$
$H_p(j\omega) = H_d(e^{j\omega T})$



Discrete-time filtering of the sequence with a general frequency response $H_d(e^{j\Omega})$

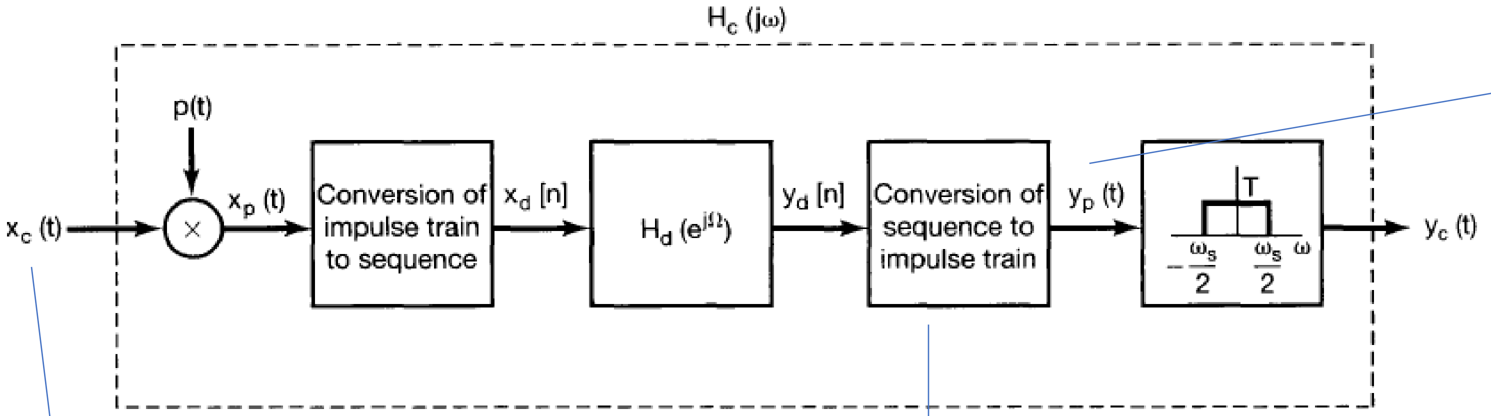


DT to CT conversion:
Apply frequency scaling to convert the overlay of $X_d(e^{j\Omega})$ and $H_d(e^{j\Omega})$ to the overlay of $X_p(j\omega)$ and $H_p(j\omega)$.

Discrete-time processing of continuous-time signals

Applying frequency scaling $\Omega = \omega T$

$H_p(j\omega) = H_d(e^{j\omega T})$



The spectrum of $y_p(t)$:

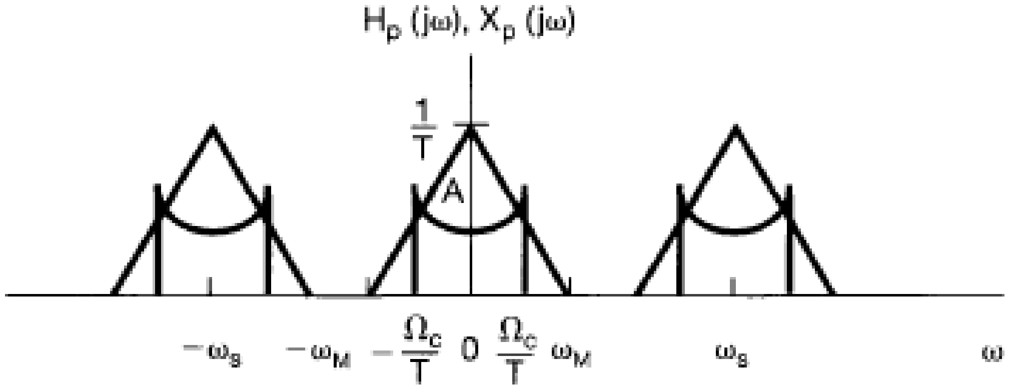
$Y_p(j\omega) = X_p(j\omega)\ H_p(j\omega)$

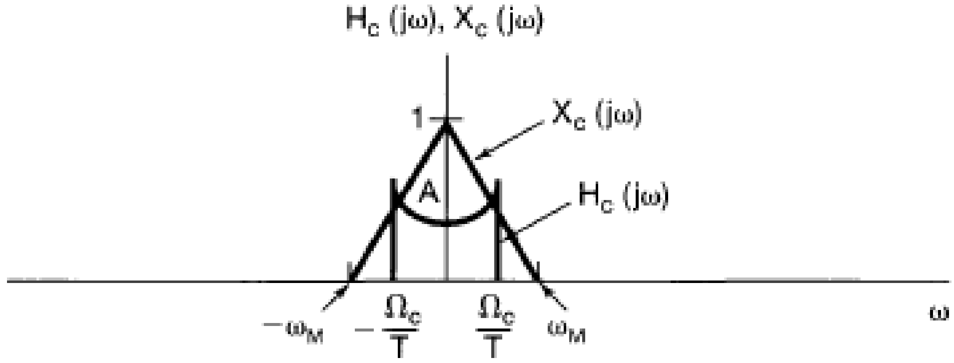
$Y_p(j\omega) = X_p(j\omega) H_d(e^{j\omega T})$

After lowpass filtering:

$Y_c(j\omega) = X_c(j\omega) H_d(e^{j\omega T})$

Overlay of spectrum of the input signal $X_c(j\omega)$ and frequency response of the overall system $H_c(j\omega)$.

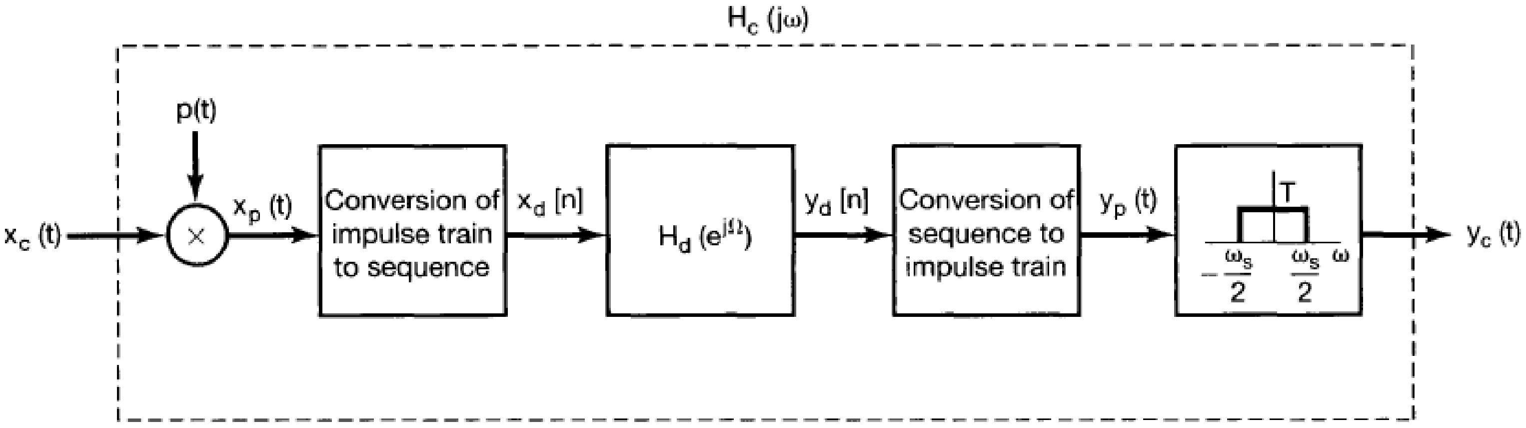


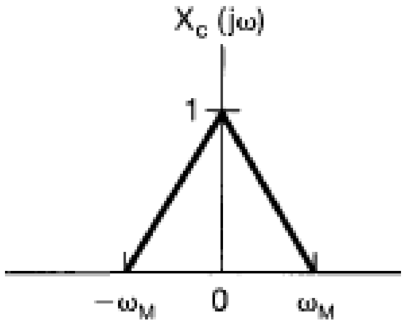


$X_c(j\omega)$:
Spectrum of the input CT signal

Overlay of spectrum of the input signal $X_c(j\omega)$ and frequency response of the overall system $H_c(j\omega)$.

Discrete-time processing of continuous-time signals



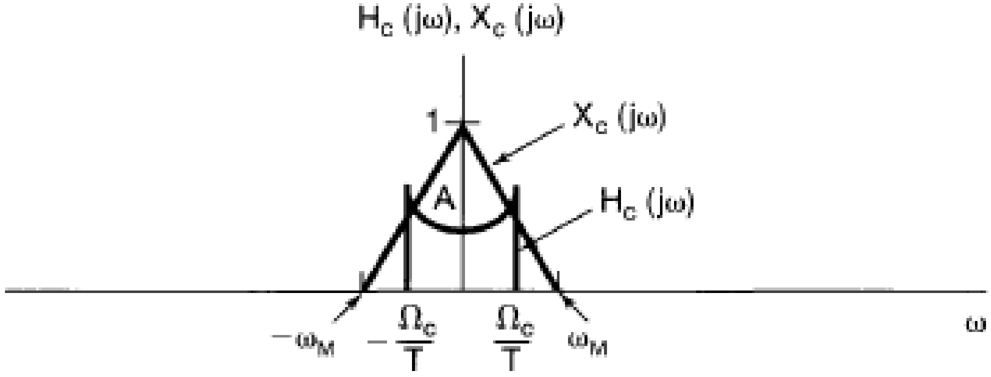


$X_c(j\omega)$: Spectrum of the input CT signal

$$Y_c(j\omega) = X_c(j\omega)H_d\left(e^{j\omega T}\right)$$

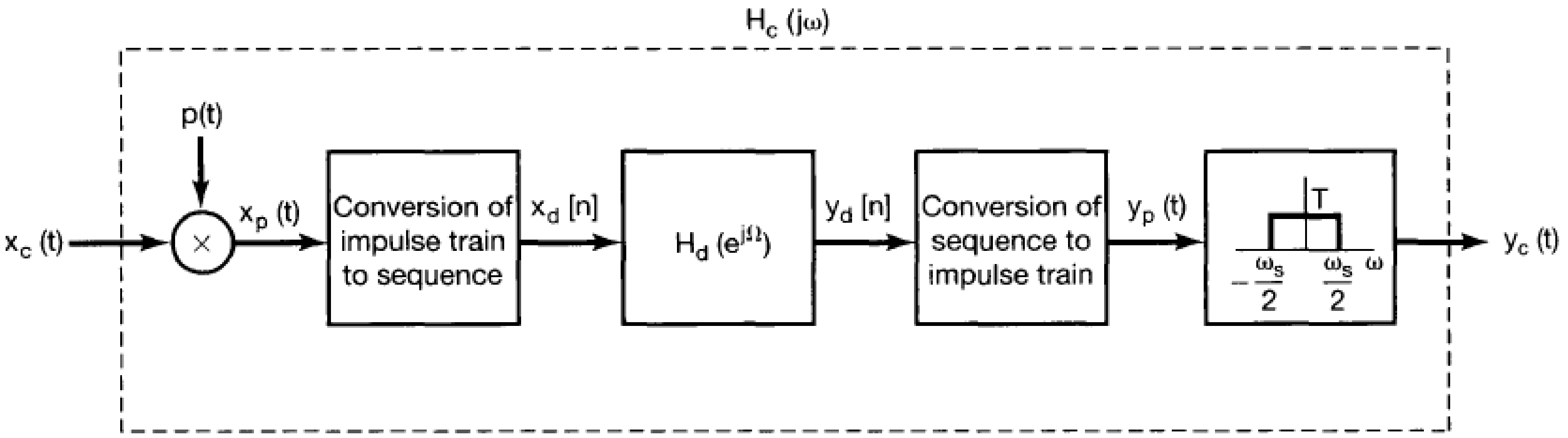
The overall system is equivalent to a continuous-time LTI system with frequency response $H_C(j\omega)$ which is related to the discrete-time frequency response $H_d\left(e^{j\Omega}\right)$ through

$$H_c(j\omega) = \begin{cases} H_d(e^{j\omega T}), & |\omega| < \omega_s/2 \\ 0, & |\omega| > \omega_s/2 \end{cases}$$



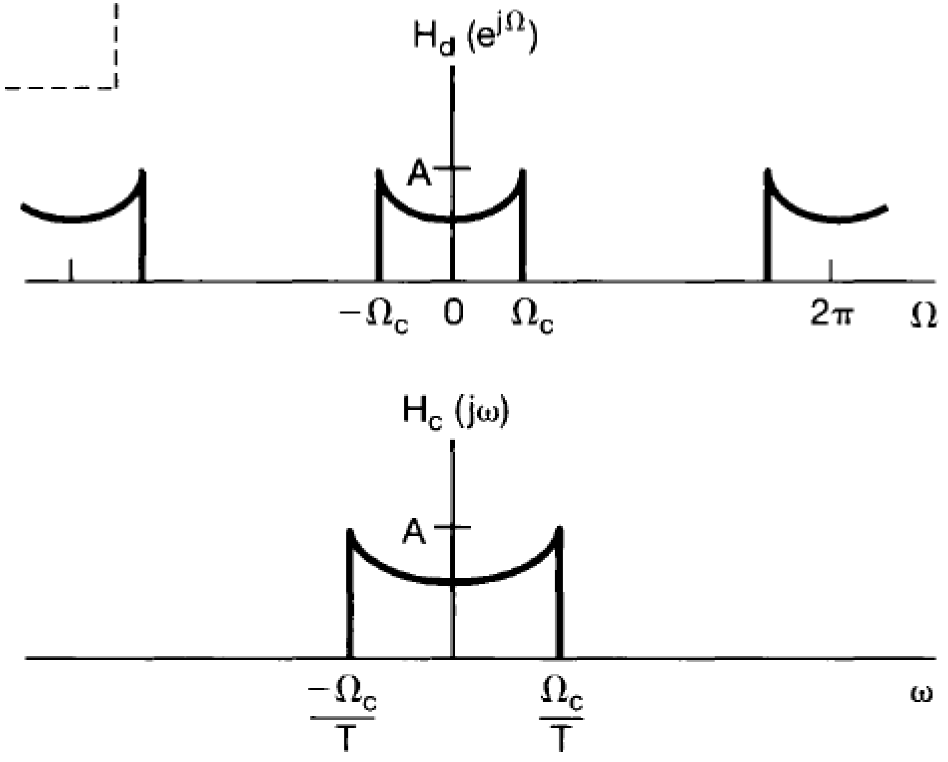
Overlay of spectrum of the input signal $X_c(j\omega)$ and frequency response of the overall system $H_C(j\omega)$.

Discrete-time processing of continuous-time signals



For inputs that are sufficiently band limited, so that the sampling theorem is satisfied, the overall system is equivalent to a continuous-time LTI system with frequency response $H_C(j\omega)$ which is related to the discrete-time frequency response $H_d\left(e^{j\Omega}\right)$ through

$$H_c(j\omega) = \begin{cases} H_d(e^{j\omega T}), & |\omega| < \omega_s/2 \\ 0, & |\omega| > \omega_s/2 \end{cases}$$



One period of the frequency response of the discrete-time filter with a frequncy scaling.

Example: Digital Differentiator

Consider the discrete-time implementation of a continuous-time band-limited differentiating filter.
The frequency response of a continuous-time differentiating filter is

$$H_c(j\omega) = j\omega$$

The frequency response of a band-limited differentiator with cutoff frequency ω_c is

$$H_c(j\omega) = \begin{cases} j\omega, & |\omega| < \omega_c \\ 0, & |\omega| > \omega_c \end{cases}$$

$$H_c(j\omega) = \begin{cases} H_d(e^{j\omega T}), & |\omega| < \omega_s/2 \\ 0, & |\omega| > \omega_s/2 \end{cases}$$

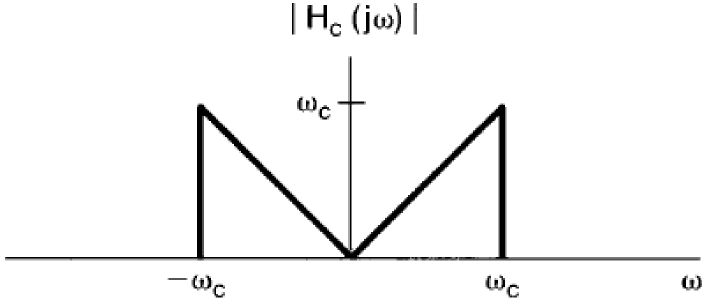
Using a sampling frequency $\omega_s = 2\omega_c$, we see that the corresponding discrete-time frequency response is

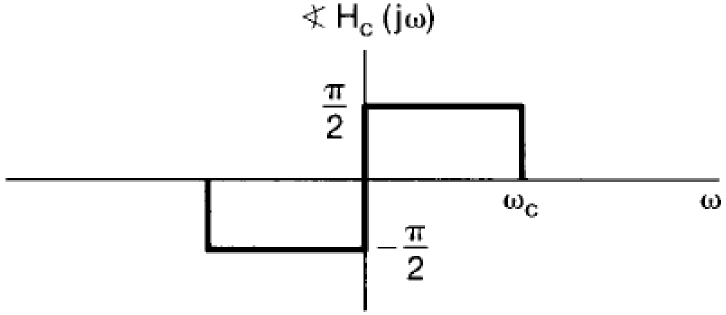
$$H_d(e^{j\Omega}) = j\left(\frac{\Omega}{T}\right), \quad |\Omega| < \pi$$

Example: Digital Differentiator

The frequency response of a band-limited differentiator with cutoff frequency ω_c is

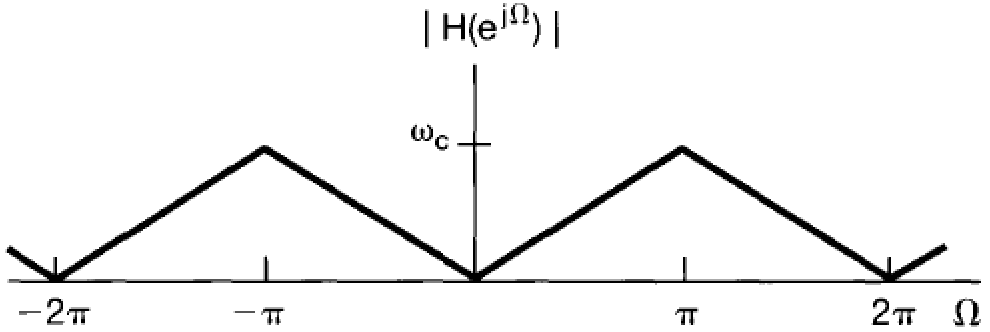
$$H_c(j\omega) = \begin{cases} j\omega, & |\omega| < \omega_c \\ 0, & |\omega| > \omega_c \end{cases}$$

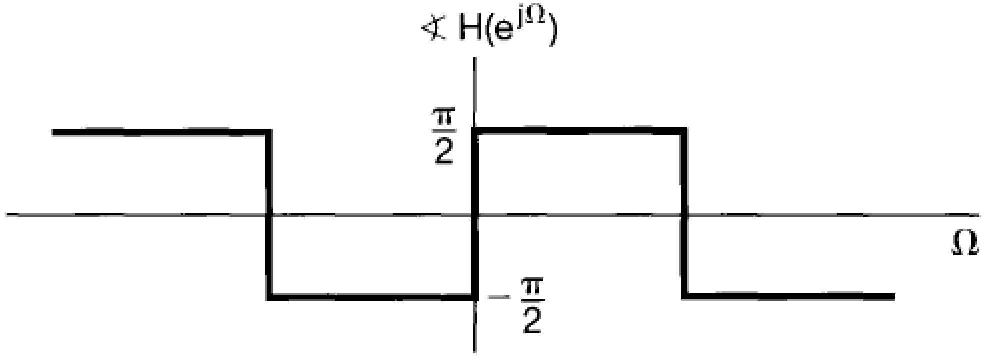




Frequency response of discrete-time filter used to implement a continuous-time band-limited differentiator:

$$H_d(e^{j\Omega}) = j\left(\frac{\Omega}{T}\right), \quad |\Omega| < \pi$$





Example: Digital Differentiator

Consider the following signal to the digital differentiator:

$$x_c(t) = \frac{\sin(\pi t/T)}{\pi t}$$

with sampling period T.

$$X_c(j\omega) = \begin{cases} 1, & |\omega| < \pi/T \\ 0, & \text{otherwise} \end{cases}$$

The signal is sufficiently band limited to ensure that sampling $x_c(t)$ at frequency $\omega_s = 2\pi/\mathrm{T}$ does not give rise to any aliasing.

When we convert this signal to discrete-time with sampling period T, we obtain:

$$x_d[n] = x_c(nT) = \frac{1}{T}\delta[n].$$

We use this result to find the impulse response $h_d[n]$ of the discrete-time filter.

Example: Digital Differentiator

The output of the differentiator should be

$$y_c(t) = \frac{d}{dt}x_c(t) = \frac{\cos(\pi t/T)}{Tt} - \frac{\sin(\pi t/T)}{\pi t^2}$$

The output of the discrete-time filter in response to $x_d[n] = \frac{1}{T}\delta[n]$ is

$$y_d[n] = y_c(nT) = \begin{cases} \frac{(-1)^n}{nT^2}, & n \neq 0 \\ 0 & n = 0 \end{cases}$$

Then, the impulse response $h_d[n]$ of the discrete-time filter is

$$h_d[n] = \begin{cases} \frac{(-1)^n}{nT}, & n \neq 0 \\ 0, & n = 0 \end{cases}$$

Example: Half-Sample Delay

We consider the implementation of a time shift (delay) of a continuous-time signal through discrete-time processing.
We require that the input and output of the overall system be related by

$$y_c(t) = x_c(t - \Delta)$$

when the input $x_c(t)$ is band limited and the sampling rate is high enough to avoid aliasing and where Δ represents the delay time.

From the time-shifting property of Fourier Transform:

$$Y_c(j\omega) = e^{-j\omega\Delta} X_c(j\omega)$$

The equivalent continuous-time system to be implemented must be band limited. Therefore, we take

$$H_c(j\omega) = \begin{cases} e^{-j\omega\Delta}, & |\omega| < \omega_c \\ 0, & \text{otherwise} \end{cases}$$

Example: Half-Sample Delay

$$H_c(j\omega) = \begin{cases} e^{-j\omega\Delta}, & |\omega| < \omega_c \\ 0, & \text{otherwise} \end{cases}$$

$H_c(j\omega)$ corresponds to a time shift for band-limited signals and rejects all frequencies greater than ω_c.