Discrete-Time Signals and Systems

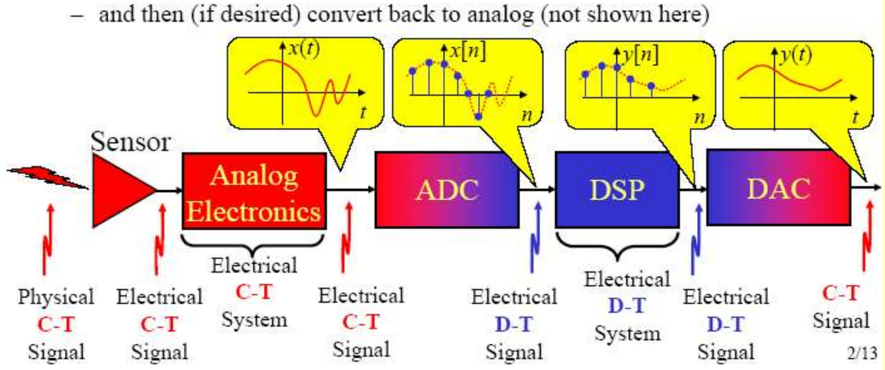
Quote of the Day

Mathematics is the tool specially suited for dealing with abstract concepts of any kind and there is no limit to its power in this field.

Paul Dirac

Discrete Signal and System

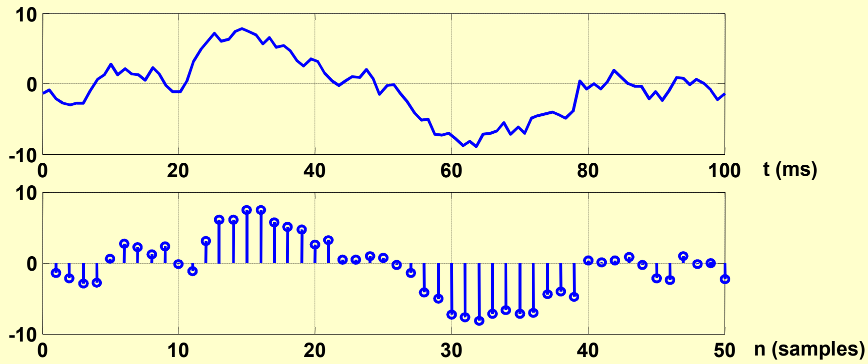
- Modern systems generally…
  - get a **continuous-time signal** from a sensor
  - a **cont.-time system** modifies the signal
  - an "analog-to-digital converter" (ADC or A-to-D) sample the signal to create a **discrete-time signal** … a "stream of numbers"
  - A **discrete-time system** to do the processing
  - and then (if desired) convert back to analog (not shown here)



2/13

Discrete-Time Signals: Sequences

- **Discrete-time signals** are represented by sequence of numbers
  - The n^{th} number in the sequence is represented with x[n]
- Often times sequences are obtained by sampling of continuous-time signals
  - In this case x[n] is value of the analog signal at $x_c(nT)$
  - Where T is the sampling period



Basic Sequences Types

- **Delaying** (Shifting) a sequence

$$y[n] = x[n - n_o]$$

- Unit sample (**impulse**) sequence

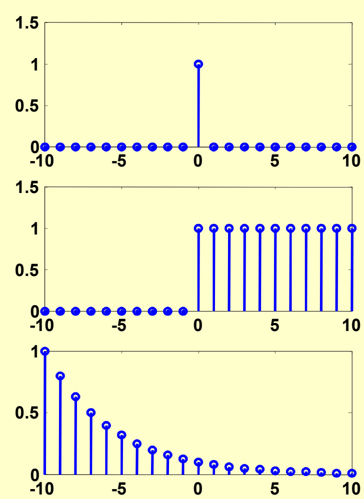
$$\delta[n] = \begin{cases} 0 & n \neq 0 \\ 1 & n = 0 \end{cases}$$

- Unit **step** sequence

$$u[n] = \begin{cases} 0 & n < 0 \\ 1 & n \geq 0 \end{cases}$$

- **Power** sequences

$$x[n] = A\alpha^n$$



Basic Sequences Types (cont')

- **Complex-valued exponential sequence**:

  $$x(n) = e^{(\sigma + j\omega_0)n}, \forall n$$

  where σ produces an attenuation (if <0) or amplification (if >0) and $\omega 0$ is the frequency in radians
- **Sinusoidal sequence**: $$x(n) = A\cos(\omega_0 n + \theta_0), \forall n$$

  where A is an amplitude and $\theta 0$ is the phase in radians.
- **Random sequences**:
- **Periodic sequence**:

  A sequence $x(n)$ is periodic if $x(n) = x(n + N)$, $\forall n$.

OPERATIONS ON SEQUENCES

- **Signal addition**:

  $$\{x_1[n]\} + \{x_2[n]\} = \{x_1[n] + x_2[n]\}$$
- **Signal multiplication**:

  $$\{x_1[n]\}.\{x_2[n]\} = \{x_1[n]x_2[n]\}$$
- **Scaling**:

  $$\alpha\ \{x[n]\} = \{\alpha x_1[n]\}$$
- **Shifting**:

  $$y[n] = \{x[n-k]\}$$
- **Folding**

  $$y[n] = \{x[-n]\}$$
- **Sample summation**:

  $$\sum_{n=n_1}^{n_2} x[n] = x[n_1] + \cdots + x[n_2]$$

OPERATIONS ON SEQUENCES (cont’

- **Sample products**:

$$\prod_{n_1}^{n_2} x[n] = x[n_1] \times \cdots \times x[n_2]$$

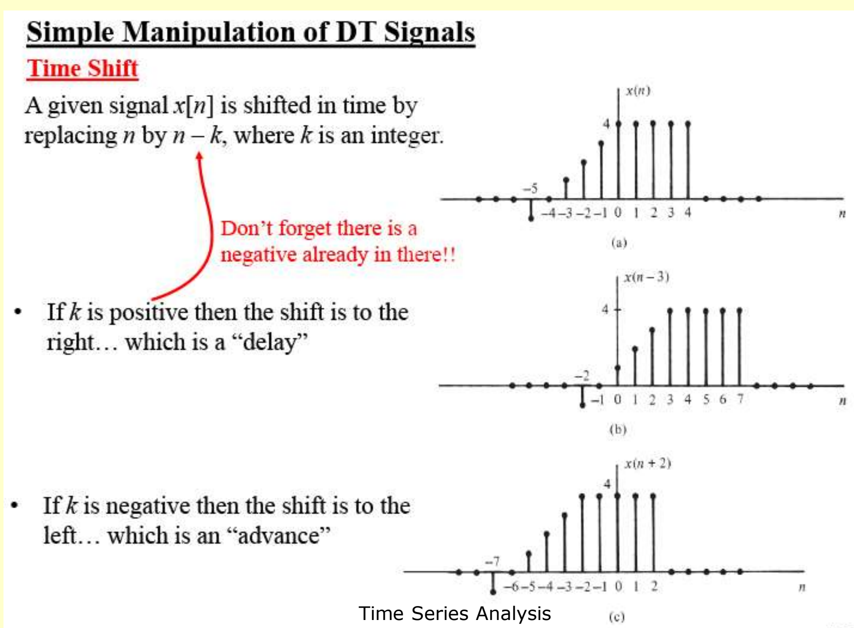
- **Signal energy**:

$$\varepsilon_x = \sum_{-\infty}^{\infty} x[n] x^*[n] = \sum_{-\infty}^{\infty} |x[n]|^2$$

- The **Signal power**:

$$P_x = \frac{1}{N} \sum_{0}^{N-1} |\tilde{x}|^2$$

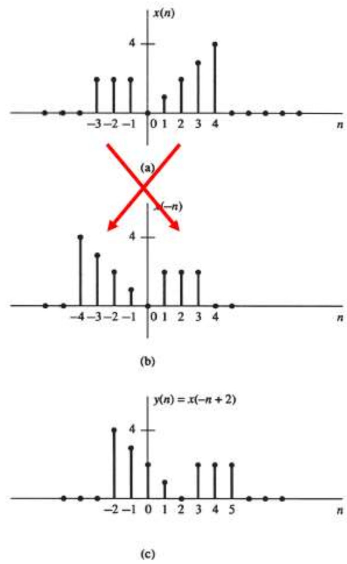
Example: time sifting



Example : time reversal

Time Reversal

A given signal $x[n]$ is reversed in time by replacing n by $-n$.



- Time reversal just "anchors" the signal at $n = 0$ and flips it
- Flipping and shifting can be combined… Careful here: Do the shift first then the flip.

$$x[n] \rightarrow \underbrace{x[n+2]}_{\text{Shift left by 2}} \rightarrow \underbrace{x[-n+2]}_{\text{Flip the shifted signal}}$$

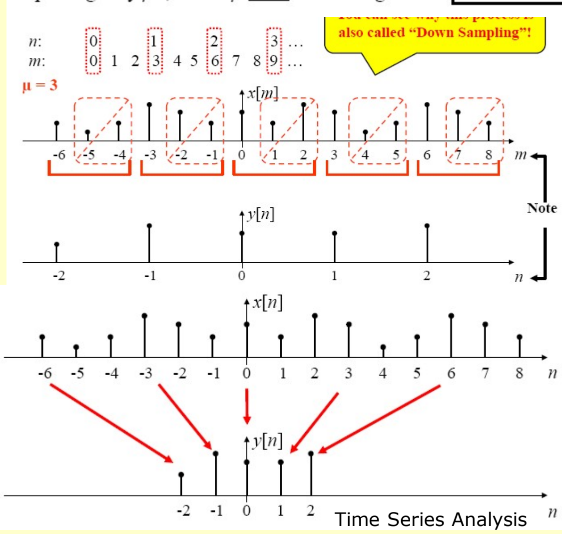
15

Example : Time Scaling

Time Scaling

A given signal $x[n]$ is scaled in time by replacing n by μn, where μ must be an integer.

$$\boxed{y[n] = x[\mu n]}$$



Sinusoidal Sequences

- Important class of sequences

$$x[n] = \cos(\omega_o n + \phi)$$

- An **exponential sequence** with complex $\alpha = |\alpha| e^{j\omega_o}$ and $A = |A| e^{j\phi}$

$$x[n] = A\alpha^n = |A| e^{j\phi} |\alpha|^n e^{j\omega_o n} = |A||\alpha|^n e^{j(\omega_o n + \phi)}$$

$$x[n] = |A||\alpha|^n \cos(\omega_o n + \phi) + j|A||\alpha|^n \sin(\omega_o n + \phi)$$

- x[n] is a sum of weighted sinusoids
- Different from continuous-time, discrete-time sinusoids
  - Have ambiguity of $2\pi k$ in frequency

$$\cos((\omega_o + 2\pi k)n + \phi) = \cos(\omega_o n + \phi)$$

  - Are not necessary periodic with $2\pi/\omega_o$

$$\cos(\omega_o n + \phi) = \cos(\omega_o n + \omega_o N + \phi) \text{ only if } N = \frac{2\pi k}{\omega_o} \text{ is an integer}$$

Discrete-Time Systems

- **Discrete-Time Sequence** is a mathematical operation that maps a given input sequence x[n] into an output sequence y[n]

$$y[n] = T\{x[n]\} \qquad x[n] \longrightarrow \boxed{T\{.\}} \longrightarrow y[n]$$

- **Example** Discrete-Time Systems
  - Moving (Running) Average

$$y[n] = \frac{1}{4}(x[n] + x[n-1] + x[n-2] + x[n-3])$$

  - Maximum

$$y[n] = \max\{x[n], x[n-1], x[n-2]\}$$

  - Ideal Delay System

$$y[n] = x[n - n_o]$$

Memoryless System

- Memoryless System
  - A system is **memoryless** if the output y[n] at every value of n depends only on the input x[n] at the same value of n

- Example (**Memoryless**)
  - Square

$$y[n] = (x[n])^2$$

  - Sign

$$y[n] = \mathrm{sign}\{x[n]\}$$

- Counter Example (**non memory less**)
  - Ideal Delay System

$$y[n] = x[n - n_o]$$

Time Series Analysis 13

Linear Systems

- **Linear System**: A system is linear if and only if

$$T\{x_1[n] + x_2[n]\} = T\{x_1[n]\} + T\{x_2[n]\} \quad \text{(additivity)}$$

and

$$T\{ax[n]\} = aT\{x[n]\} \quad \text{(scaling)}$$

- **Examples**
  - **Ideal Delay System**

$$y[n] = x[n - n_o]$$

$$\begin{aligned} T\{x_1[n] + x_2[n]\} &= x_1[n - n_o] + x_2[n - n_o] \\ T\{x_2[n]\} + T\{x_1[n]\} &= x_1[n - n_o] + x_2[n - n_o] \\ T\{ax[n]\} &= ax_1[n - n_o] \\ aT\{x[n]\} &= ax_1[n - n_o] \end{aligned}$$

Time Series Analysis 14

Time-Invariant Systems

- **Time-Invariant (shift-invariant) Systems**
  - A time shift at the input causes corresponding time-shift at output

$$y[n] = T\{x[n]\} \Rightarrow y[n - n_o] = T\{x[n - n_o]\}$$

- **Example (LTI)**
  - Square

$$y[n] = (x[n])^2$$

Delay the input the output is $y_1[n] = (x[n - n_o])^2$

Delay the output gives $y[n - n_o] = (x[n - n_o])^2$

- Counter Example (**non-LTI**)
  - Compressor System

$$y[n] = x[Mn]$$

Delay the input the output is $y_1[n] = x[Mn - n_o]$

Delay the output gives $y[n - n_o] = x[M(n - n_o)]$

Causal System

- **Causality**
  - A system is causal it's output is a function of only the current and previous samples

- Examples (**causal**)
  - Backward Difference

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

- Counter Example (**non-causal**)
  - Forward Difference

$$y[n] = x[n + 1] + x[n]$$

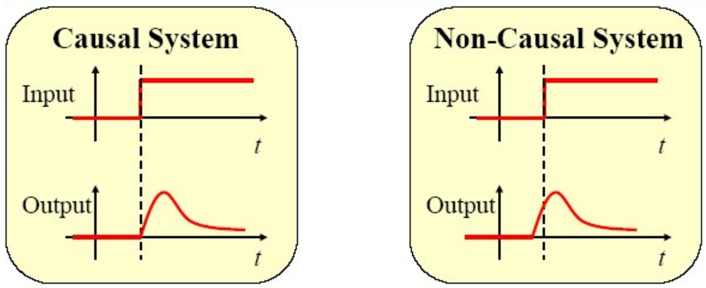
Causal System (cont')

Causal vs. Non-Causal

A causal (or non-anticipatory) system's output at a time t_1 does not depend on values of the input $x(t)$ for $t > t_1$

The "future input" cannot impact the "now output"

$\Rightarrow$ A Causal system (with zero initial conditions) cannot have a non-zero output until a non-zero input is applied.



"Real-time" systems must be causal…. But time-signals recorded processed off-line can in essence be non-causal. Images can be processed "non-causally".

24/.7

Stable System

- **Stability** (in the sense of bounded-input bounded-output **BIBO**)
  - A system is stable if and only if every bounded input produces a bounded output

$$|x[n]| \le B_x < \infty \Rightarrow |y[n]| \le B_y < \infty$$

- Example (**stable**)
  - Square

$$y[n] = (x[n])^2$$

if input is bounded by $|x[n]| \le B_x < \infty$

output is bounded by $|y[n]| \le B_x^2 < \infty$

- Counter Example (non-**stable**)
  - Log

$$y[n] = \log_{10}(|x[n]|)$$

even if input is bounded by $|x[n]| \le B_x < \infty$

output not bounded for $x[n] = 0 \Rightarrow y[0] = \log_{10}(|x[n]|) = -\infty$

Discrete Signal Analysis

Transforms & Notation

Proakis & Manolakis don't use this superscript Notation. I borrowed it from Porat's DSP Book

Fourier Transform for CT Signals

$$X^F(F) = \int_{-\infty}^{\infty} x(t) e^{-j2\pi Ft} dt$$

$$x(t) = \int_{-\infty}^{\infty} X^F(F) e^{j2\pi Ft} dF$$

Fourier Transform for DT Signals

$$X^f(\omega) = \sum_{n=-\infty}^{\infty} x[n] e^{-j\omega n}$$

Discrete **T**ime **F**ourier **T**ransform

$$x[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} X^f(\omega) e^{j\omega n} d\theta$$

Set $z = e^{j\omega}$

Z Transform for DT Signals

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

Inverse ZT done using partial fractions & a ZT table

Discrete Fourier Transform for DT Signals

Discrete **F**ourier **T**ransform

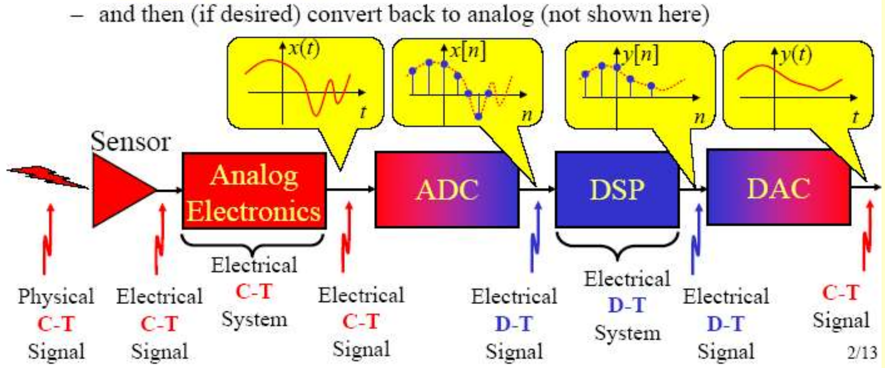
$$X^d[k] = \sum_{n=0}^{N-1} x[n] e^{-j2\pi kn/N} \quad k = 0, 1, 2, \ldots, N-1$$

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X^d[k] e^{j2\pi kn/N} \quad n = 0, 1, 2, \ldots, N-1$$

4/13

Discrete Signal and System

- Modern systems generally…
  - get a **continuous-time signal** from a sensor
  - a **cont.-time system** modifies the signal
  - an "analog-to-digital converter" (ADC or A-to-D) sample the signal to create a **discrete-time signal** … a "stream of numbers"
  - A **discrete-time system** to do the processing
  - and then (if desired) convert back to analog (not shown here)

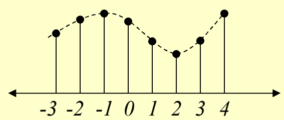


2/13

Time Series Analysis 20

Periodic (Uniform) Sampling

- **Sampling** is a continuous to discrete-time conversion



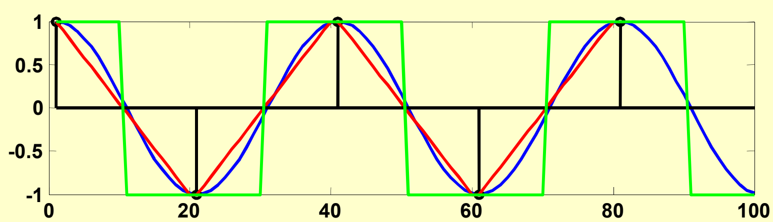
- Most common sampling is **periodic**

$$x[n] = x_c(nT) \quad -\infty < n < \infty$$

- T is the sampling period in second
- f_s = 1/T is the **sampling frequency** in Hz
- **Sampling frequency** in **radian-per-second** $\Omega_s=2\pi f_s$ rad/sec
- Use **[.]** for **discrete-time** and **(.)** for **continuous time signals**
- This is the ideal case not the practical but close enough
  - In practice it is implement with an analog-to-digital converters
  - We get digital signals that are quantized in amplitude and time

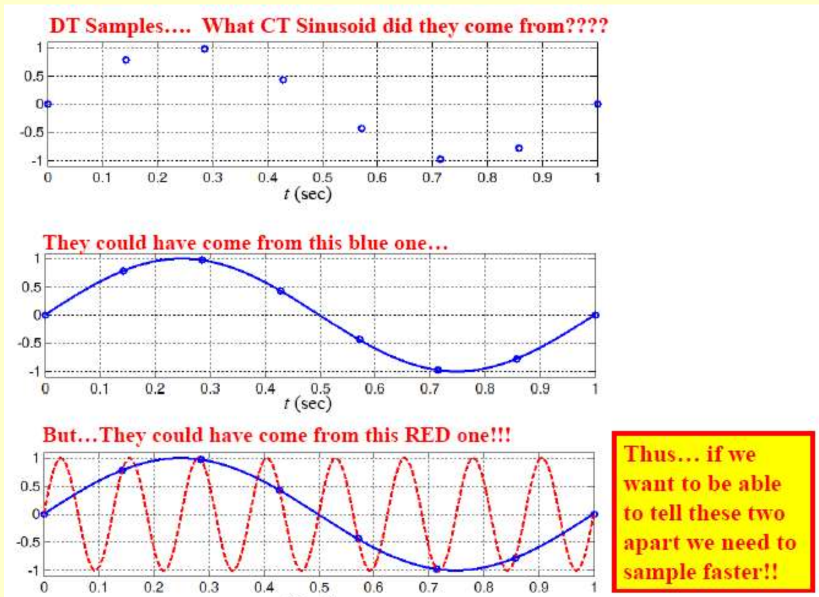
Periodic Sampling

- Sampling is, in general, **not reversible**
- Given a sampled signal one could fit infinite continuous signals through the samples



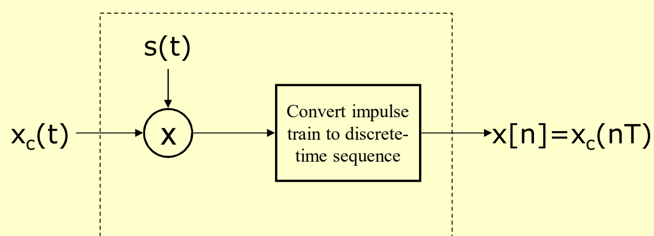
- Fundamental issue in digital signal processing
  - **If we loose information during sampling we cannot recover it**
- *Under certain conditions an analog signal can be sampled without loss so that it can be reconstructed perfectly*

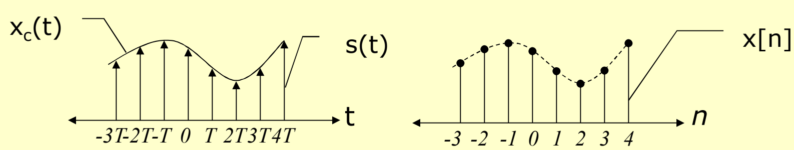
Sampling from Continuous signal



Representation of Sampling

- Mathematically convenient to represent in two stages
  - Impulse train modulator
  - Conversion of impulse train to a sequence





Continuous-Time Fourier Transform

- **Continuous-Time Fourier transform pair** is defined as

$$X_c(j\Omega) = \int_{-\infty}^{\infty} x_c(t) e^{-j\Omega t} dt$$

$$x_c(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X_c(j\Omega) e^{j\Omega t} d\Omega$$

- We write $x_c(t)$ as a weighted sum of complex exponentials
- Remember some Fourier Transform properties
  - **Time Convolution** (**frequency domain multiplication**)
    $$x(t) * y(t) \leftrightarrow X(j\Omega)Y(j\Omega)$$
  - **Frequency Convolution** (**time domain multiplication**)
    $$x(t)y(t) \leftrightarrow X(j\Omega) * Y(j\Omega)$$
  - **Modulation** (**Frequency shift**)
    $$x(t)e^{j\Omega_o t} \leftrightarrow X(j(\Omega - \Omega_o))$$

Frequency Domain Representation of Sampling

- Modulate (multiply) **continuous-time signal with pulse train**:

$$x_s(t) = x_c(t)s(t) = \sum_{n=-\infty}^{\infty} x_c(t)\delta(t - nT) \qquad s(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT)$$

- Let's take **the Fourier Transform** of $x_s(t)$ and $s(t)$

$$X_s(j\Omega) = \frac{1}{2\pi} X_c(j\Omega) * S(j\Omega) \qquad S(j\Omega) = \frac{2\pi}{T} \sum_{k=-\infty}^{\infty} \delta(\Omega - k\Omega_s)$$

- **Fourier transform of pulse train** is **again a pulse train**
- Note that multiplication in time is convolution in frequency
- We represent frequency with $\Omega = 2\pi f$ hence $\Omega_s = 2\pi f_s$

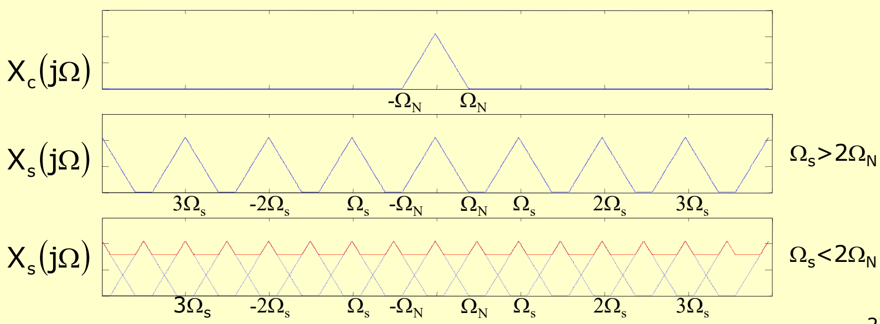
$$X_s(j\Omega) = \frac{1}{T} \sum_{k=-\infty}^{\infty} X_c(j(\Omega - k\Omega_s))$$

Frequency Domain Representation of Sampling

- Convolution with pulse creates replicas at pulse location:

$$X_s(j\Omega) = \frac{1}{T}\sum_{k=-\infty}^{\infty} X_c(j(\Omega - k\Omega_s))$$

- This tells us that the impulse train modulator
  - Creates images of the Fourier transform of the input signal
  - Images are periodic with sampling frequency
  - If $\Omega_s < \Omega_N$ sampling maybe irreversible due to aliasing of images



Nyquist Sampling Theorem

- Let $x_c(t)$ be a bandlimited signal with

$$X_c(j\Omega) = 0 \qquad \text{for } |\Omega| \geq \Omega_N$$

- Then $x_c(t)$ is uniquely determined by its samples $x[n] = x_c(nT)$ if

$$\Omega_s = \frac{2\pi}{T} = 2\pi f_s \geq 2\Omega_N$$

- Ω_N is generally known as the **Nyquist Frequency**
- The minimum sampling rate that must be exceeded is known as the **Nyquist Rate**