http://elec3004.org

Sampling & More

ELEC 3004: **Digital Linear Systems**: Signals & Controls
Dr. Surya Singh

Lecture 3

elec3004@itee.uq.edu.au
http://robotics.itee.uq.edu.au/~elec3004/

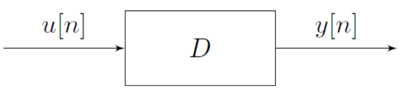
March 16, 2015

Lecture Schedule:

Week	Date	Lecture Title
1	2-Mar	Introduction
	3-Mar	Systems Overview
2	9-Mar	Signals as Vectors & Systems as Maps
	10-Mar	[Signals]
3	16-Mar	Sampling & Data Acquisition & Antialiasing Filters
	17-Mar	[Sampling]
4	23-Mar	System Analysis & Convolution
	24-Mar	[Convolution & FT]
5	30-Mar	Frequency Response & Filter Analysis
	31-Mar	[Filters]
6	13-Apr	Discrete Systems & Z-Transforms
	14-Apr	[Z-Transforms]
7	20-Apr	Introduction to Digital Control
	21-Apr	[Feedback]
8	27-Apr	Digital Filters
	28-Apr	[Digital Filters]
9	4-May	Digital Control Design
	5-May	[Digitial Control]
10	11-May	Stability of Digital Systems
	12-May	[Stability]
11	18-May	State-Space
	19-May	Controllability & Observability
12	25-May	PID Control & System Identification
	26-May	Digitial Control System Hardware
13	31-May	Applications in Industry & Information Theory & Communications
	2-Jun	Summary and Course Review

Interpretations of Systems as Maps

Then a System is a **MATRIX**



$$y = Du.$$

$$\begin{bmatrix} y[1] \\ y[2] \\ \vdots \\ y[M] \end{bmatrix} = \begin{bmatrix} D_{11} & D12 & \cdots & D_{1N} \\ D_{21} & D_{22} & \cdots & D_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ D_{M1} & D_{M2} & \cdots & D_{MN} \end{bmatrix} \begin{bmatrix} u[1] \\ u[2] \\ \vdots \\ u[N] \end{bmatrix}.$$

$$y[i] = \sum_j D_{ij} u[j].$$

System Analysis

[Chapter 2, Lathi]

Linear Differential Systems

$$\frac{d^n y}{dt^n} + a_{n-1}\frac{d^{n-1}y}{dt^{n-1}} + \cdots + a_1\frac{dy}{dt} + a_0 y(t) = b_m\frac{d^m f}{dt^m} + b_{m-1}\frac{d^{m-1}f}{dt^{m-1}} + \cdots + b_1\frac{df}{dt} + b_0 f(t) \qquad (2.1a)$$

where all the coefficients a_i and b_i are constants. Using operational notation D to represent d/dt, we can express this equation as

$$(D^n + a_{n-1}D^{n-1} + \cdots + a_1 D + a_0)\, y(t) = (b_m D^m + b_{m-1}D^{m-1} + \cdots + b_1 D + b_0)\, f(t) \qquad (2.1b)$$

or

$$Q(D)y(t) = P(D)f(t) \qquad (2.1c)$$

where the polynomials $Q(D)$ and $P(D)$ are

$$Q(D) = D^n + a_{n-1}D^{n-1} + \cdots + a_1 D + a_0 \qquad (2.2a)$$

$$P(D) = b_m D^m + b_{m-1}D^{m-1} + \cdots + b_1 D + b_0 \qquad (2.2b)$$

Linear Differential System Order

$$Q(D)y(t) = P(D)f(t)$$

$$Q(D) = D^n + a_{n-1}D^{n-1} + \cdots + a_1 D + a_0$$

$$P(D) = b_m D^m + b_{m-1}D^{m-1} + \cdots + b_1 D + b_0$$

→ *y(t)=P(D)/Q(D) f(t)*
P(D): M
Q(D): N
(yes, N is de**N**ominator)

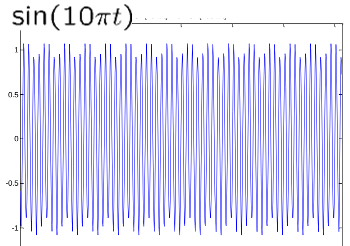
- In practice: $m \leq n$

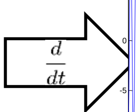
∵ **if** m > *n:*
then the system is an
(m - n)th -order differentiator of high-frequency signals!

- Derivatives magnify noise!

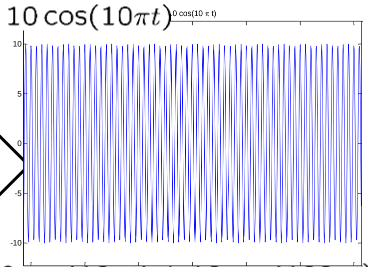
Derivatives magnify noise!

- $\sin(10\pi t)$

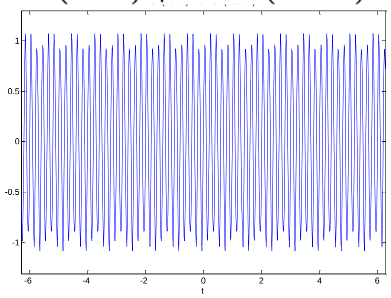




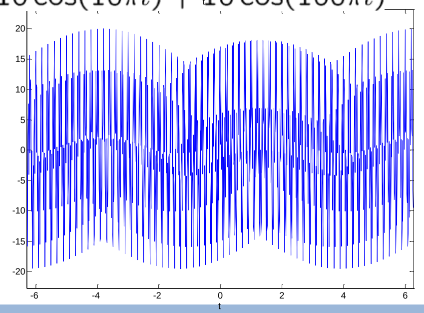
- $10\cos(10\pi t)$



- $\sin(10\pi t) + 0.1\sin(100\pi t)$



- $10\cos(10\pi t) + 10\cos(100\pi t)$



Zero-Input | Zero-State

Total response = zero-input response + zero-state response

Zero Input

- = The system response when the input *f(t)* = 0 so that it is the result of internal system conditions (such as energy storages, initial conditions) alone.
- It is **independent of the external input**.

Zero-State

- = the system response to the external input *f (t)* when the system is in zero state, meaning the absence of all internal energy storages; that is, all initial conditions are zero.

System Stability

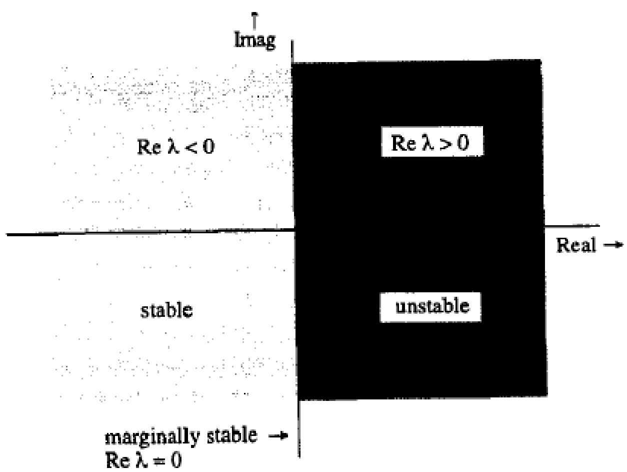
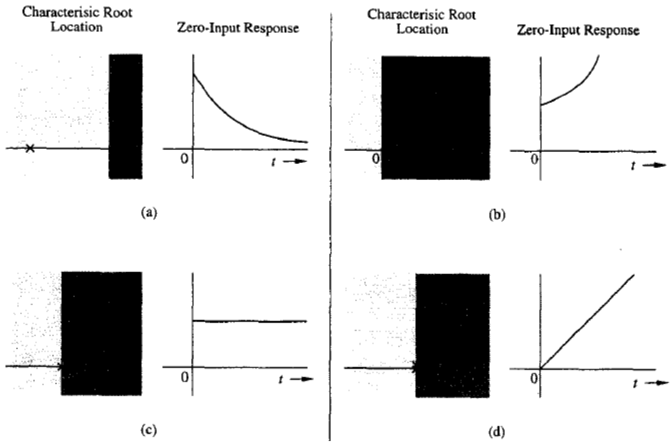


Fig. 2.15 Characteristic roots location and system stability.

Lathi, p. 149

System Stability [II]



Lathi, p. 150

System Stability [III]

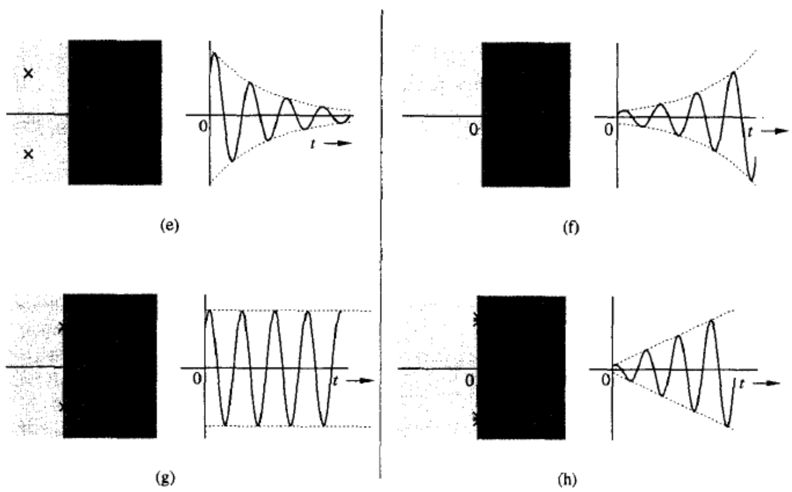


Fig. 2.16 Location of characteristic roots and the corresponding characteristic modes.

Signals Review

Signal: A carrier of (desired) information [1]

- Need **NOT** be electrical:
  - Thermometer
  - Clock hands
  - Automobile speedometer

- Need **NOT** always being given
  - – "Abnormal" sounds/operations
  - – Ex: "pitch" or "engine hum" during machining as an indicator for feeds and speeds

Signal: A carrier of (desired) information [2]

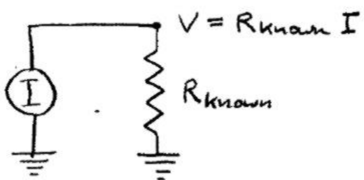
- Electrical signals
  - Voltage
  - Current

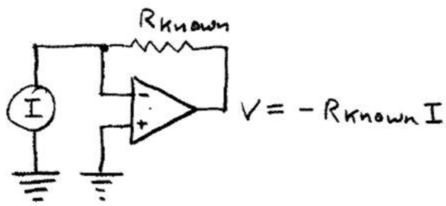
- **Digital signals**
  - **Convert analog electrical signals to an appropriate digital electrical message**
  - **Processing by a microcontroller or microprocessor**

Ex: Current-to-voltage conversion

- simple:
  Precision Resistor
- better:
  Use an "op amp"

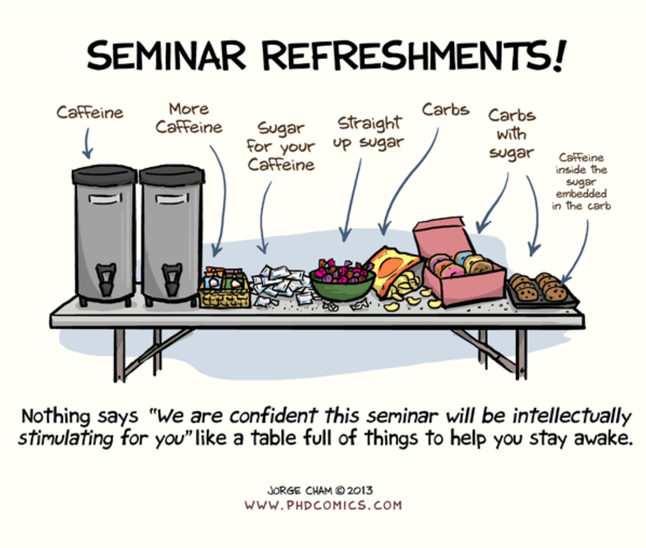
$$i = \frac{V_{\text{measured}}}{R_{\text{known}}}$$



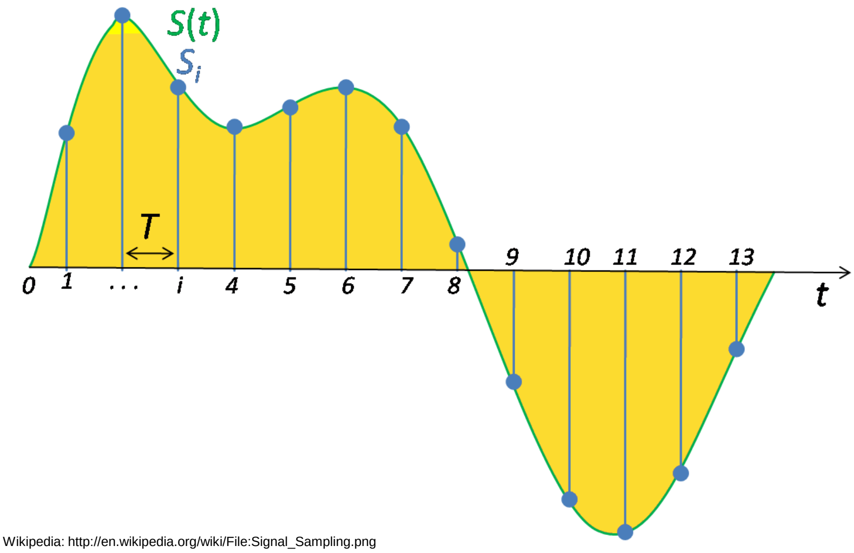


Sampling!

Not this type of sampling …



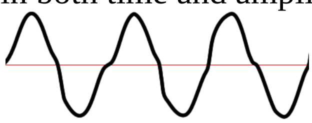
This type of sampling…



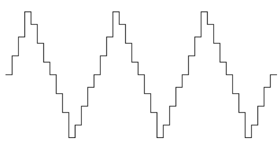
Source: Wikipedia: http://en.wikipedia.org/wiki/File:Signal_Sampling.png

Analog vs Digital

- Analog Signal: An analog or analogue signal is any variable signal continuous in both time and amplitude



- Digital Signal: A digital signal is a signal that is both discrete and quantized



E.g. Music stored in a CD: 44,100 Samples per second and 16 bits to represent amplitude

Digital Signal

- Representation of a signal against a discrete set
- The set is fixed in by computing hardware

$$s \in \mathbb{Z}$$

- Can be scaled or normalized … but is limited

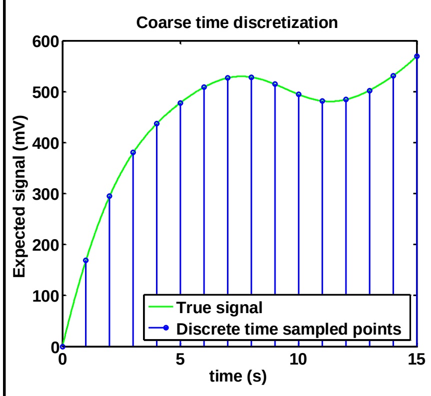
$$s \in \mathbb{Z}(0, \ldots, 2^{16})$$

- Time is also discretized

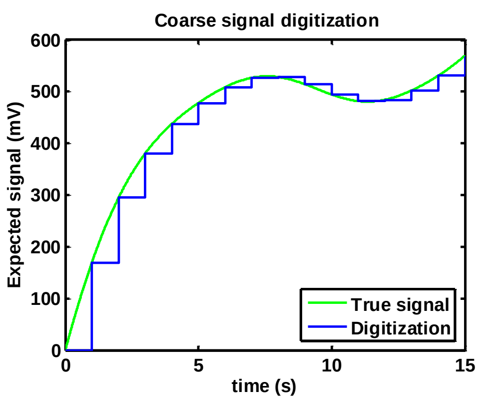
$$s' \in \frac{\mathbb{Z}(0,\ldots,2^{16})}{2^{16}}$$

Representation of Signal

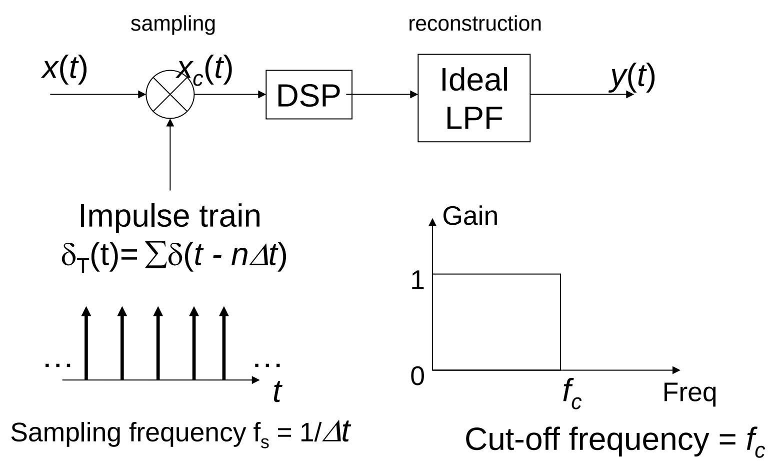
- Time Discretization



- Digitization



Mathematics of Sampling and Reconstruction

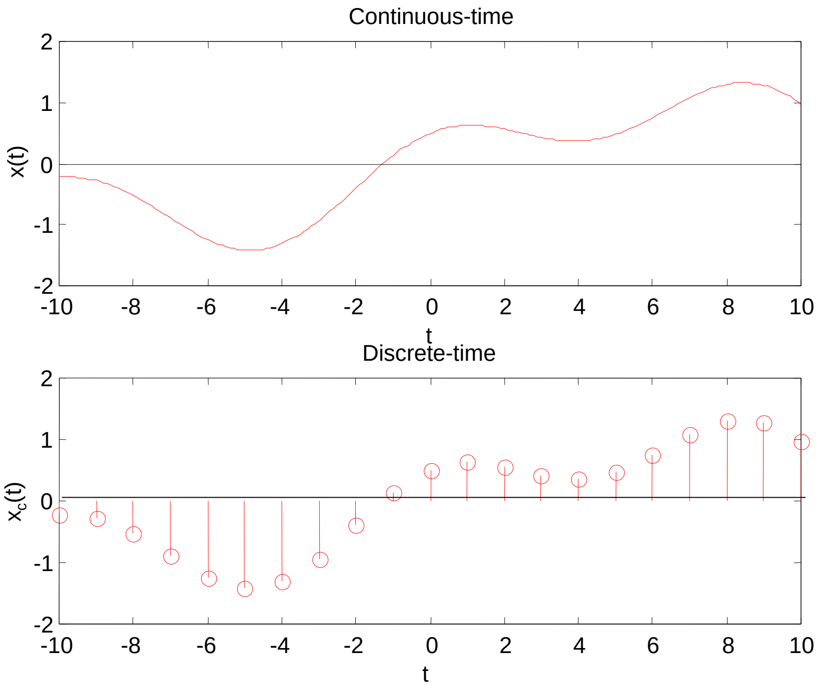


Mathematical Model of Sampling

- x(t) multiplied by impulse train δT(t)

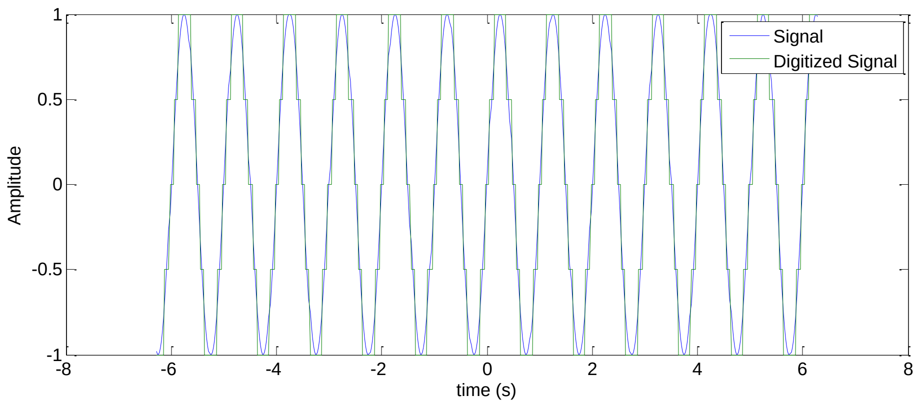
$$
\begin{aligned}
x_c(t) &= x(t)\delta_T(t) \\
&= x(t)\left[\delta(t) + \delta(t-\Delta t) + \delta(t-2\Delta t) + \cdots\right] \\
&= \sum_n x(n\Delta t)\delta(t - n\Delta t)
\end{aligned}
$$

- $x_c(t)$ is a train of impulses of height $x(t)|_{t=n\Delta t}$



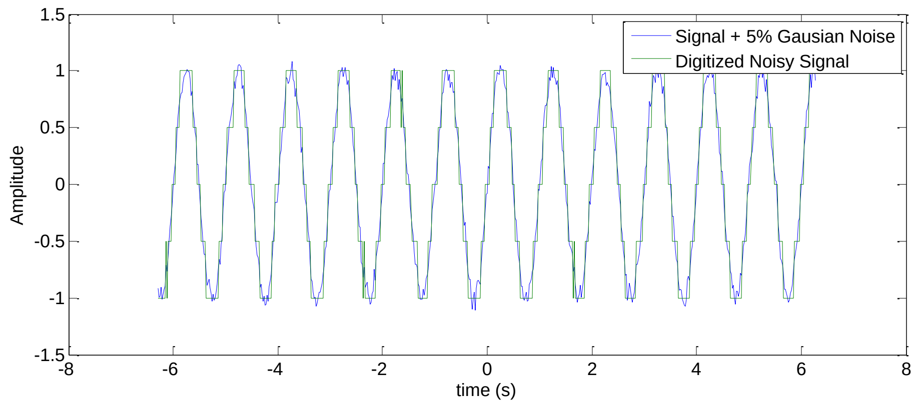
Discrete Time Signal

- Image a signal…



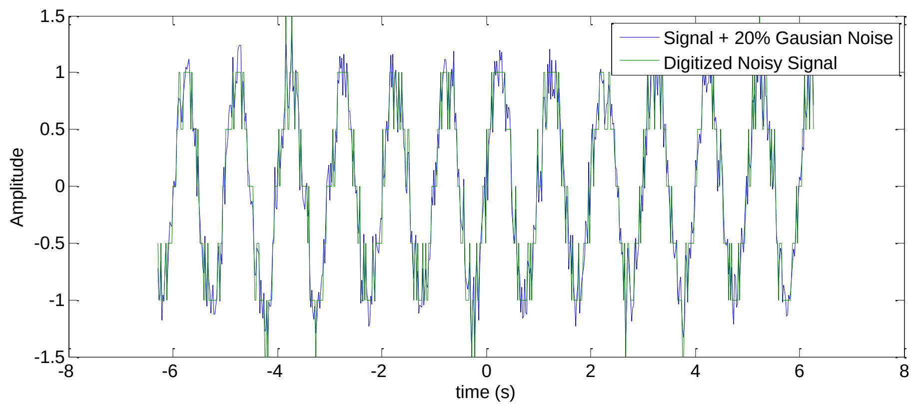
Discrete Time Signals

- Digitization helps beat the Noise!



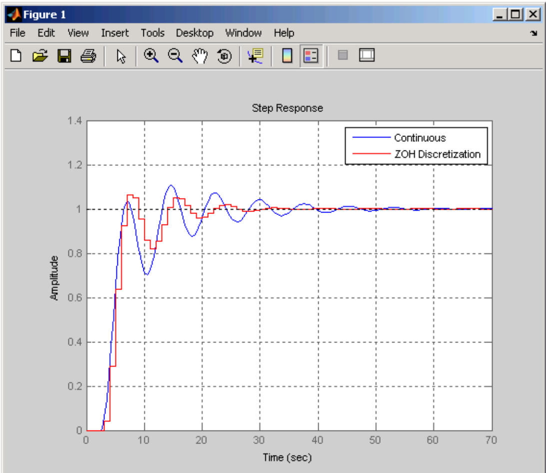
Discrete Time Signals

- But only so much…



Discrete Time Signals

- Can make control tricky!



Signal Manipulations

- Shifting

$$y(n) = x(n - n_0)$$

- Reversal

$$y(n) = x(-n)$$

- Time Scaling

(Down Sampling)

$$y(M) = x(Mn)$$

(Up Sampling)

$$y(n) = x\left(\frac{n}{N}\right)$$

Frequency Domain Analysis of Sampling

- Consider the case where the DSP performs no filtering operations
  - i.e., only passes xc(t) to the reconstruction filter
- To understand we need to look at the frequency domain
- Sampling: we know
  - multiplication in time ≡ convolution in frequency
  - F{x(t)} = X(w)
  - F{δT(t)} = ∑δ(w - 2πn/Δt),
  - i.e., an impulse train in the frequency domain

Frequency Domain Analysis of Sampling

- In the frequency domain we have

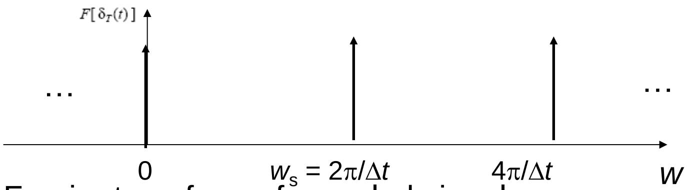
$$X_c(w) = \frac{1}{2\pi}\left(X(w) * \frac{2\pi}{\Delta t}\sum_n \delta\left(w - \frac{2\pi n}{\Delta t}\right)\right)$$
$$= \frac{1}{\Delta t}\sum_n X\left(w - \frac{2\pi n}{\Delta t}\right)$$

Remember convolution with an impulse? Same idea for an impulse train

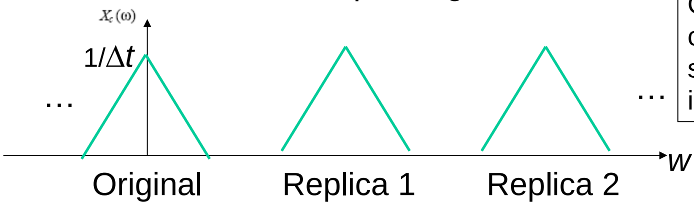
- Let's look at an example
  - where X(w) is triangular function
  - with maximum frequency w_m rad/s
  - being sampled by an impulse train, of frequency w_s rad/s

Fourier transform of original signal X(ω) (signal spectrum)

Fourier transform of impulse train $\delta_T(\omega/2\pi)$ (sampling signal)

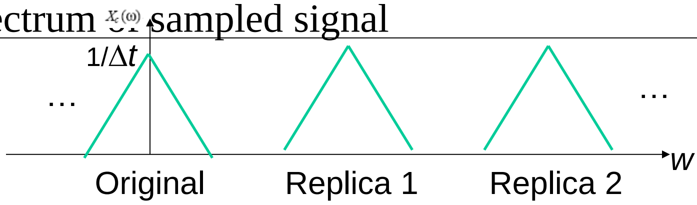


Fourier transform of sampled signal

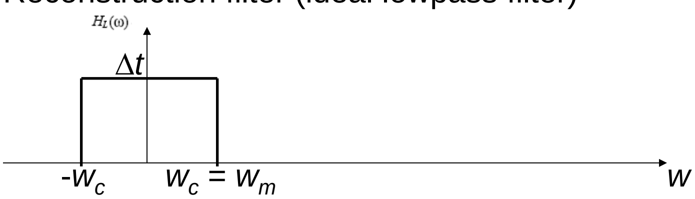


Original spectrum convolved with spectrum of impulse train

Spectrum of sampled signal

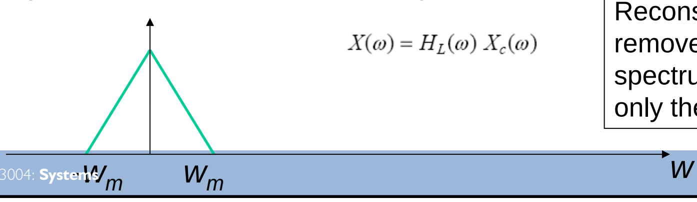


Reconstruction filter (ideal lowpass filter)



Spectrum of reconstructed signal

$$X(\omega) = H_L(\omega)\, X_c(\omega)$$

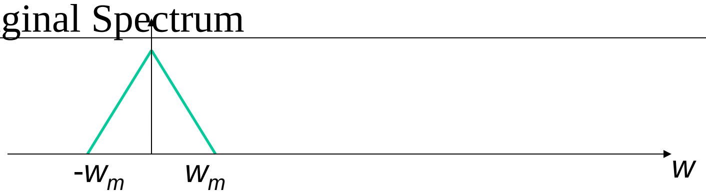


Reconstruction filter removes the replica spectrums & leaves only the original

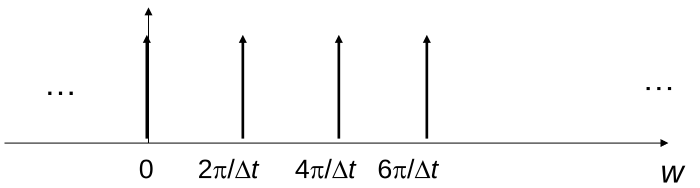
Sampling Frequency

- In this example it was possible to recover the original signal from the discrete-time samples
- But is this always the case?
- Consider an example where the sampling frequency w_s is reduced
  - i.e., Δt is increased

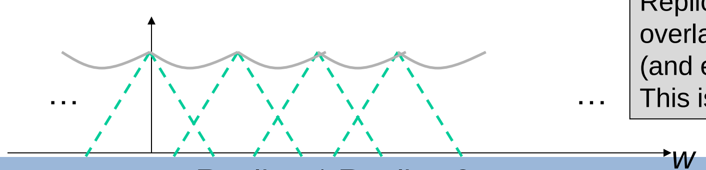
Original Spectrum



Fourier transform of impulse train (sampling signal)



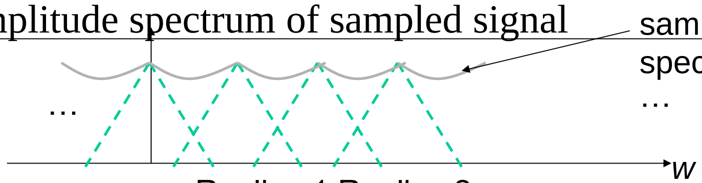
Amplitude spectrum of sampled signal



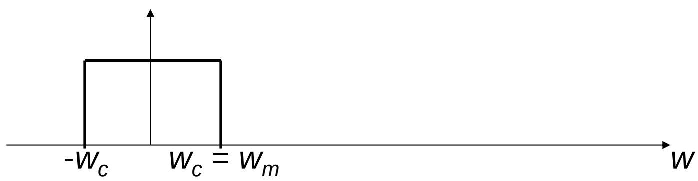
Original Replica 1 Replica 2 …

Replica spectrums overlap with origina
(and each other)
This is **Aliasing**

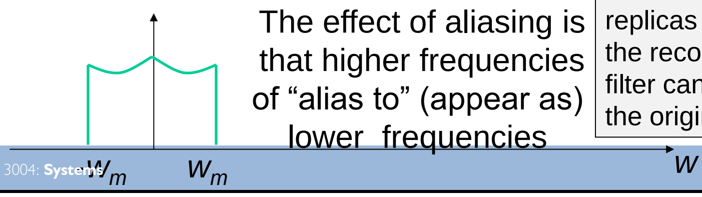
Amplitude spectrum of sampled signal



Reconstruction filter (ideal lowpass filter)



Spectrum of reconstructed signal



The effect of aliasing is that higher frequencies of "alias to" (appear as) lower frequencies

Due to overlapping replicas (aliasing) the reconstruction filter cannot recover the original spectrum

Sampling Theorem

- The Nyquist criterion states:

To prevent aliasing, a ***bandlimited*** *signal of bandwidth* w_B *rad/s must be sampled at a rate greater than* $2w_B$ *rad/s*

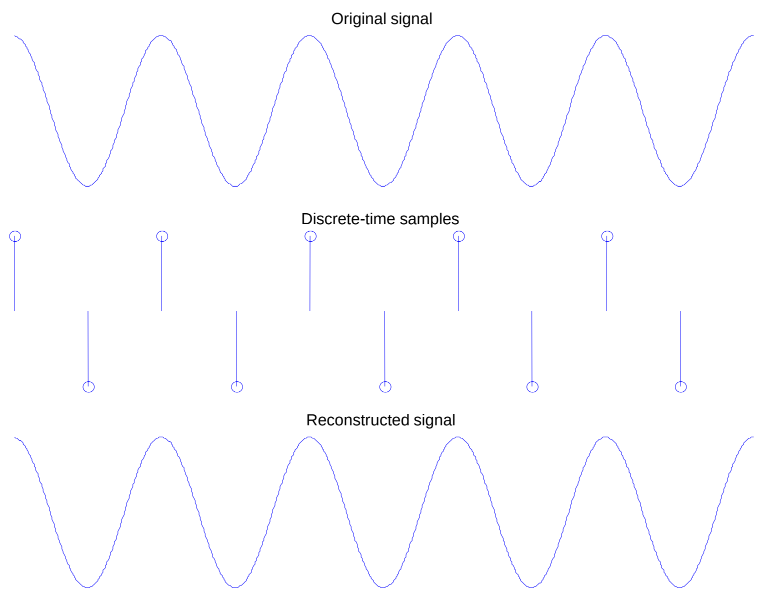
$$-w_s > 2w_B$$

Note: this is a > sign not a ≥

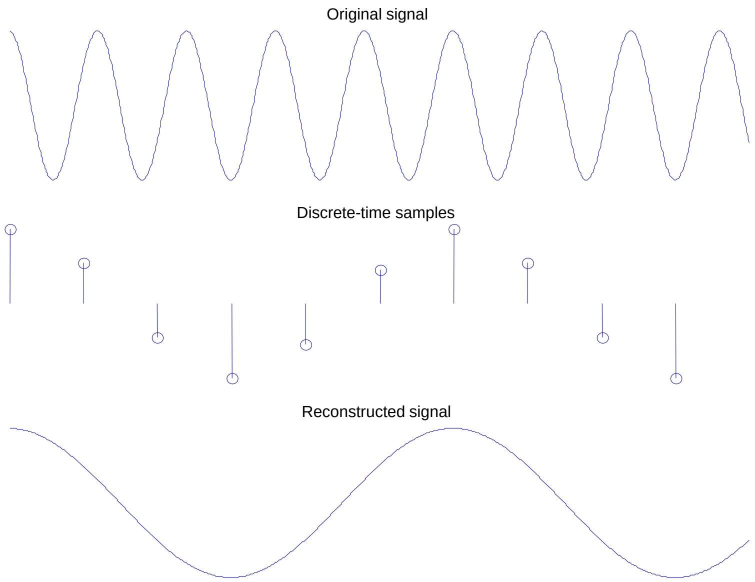
Also note: Most real world signals require band-limiting with a lowpass (anti-aliasing) filter

Time Domain Analysis of Sampling

- Frequency domain analysis of sampling is very useful to understand
  - sampling ($X(w) * \sum \delta(w - 2\pi n/\Delta t)$)
  - reconstruction (lowpass filter removes replicas)
  - aliasing (if $w_s \leq 2w_B$)
- Time domain analysis can also illustrate the concepts
  - sampling a sinewave of increasing frequency
  - sampling images of a rotating wheel

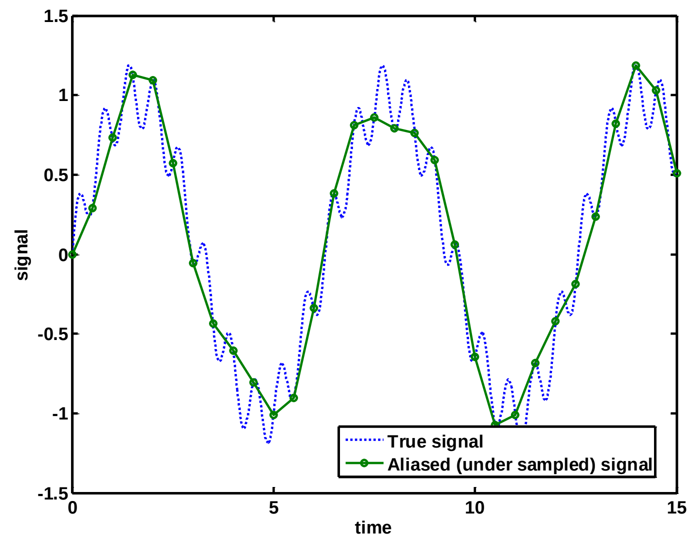


A signal of the original frequency is reconstructed

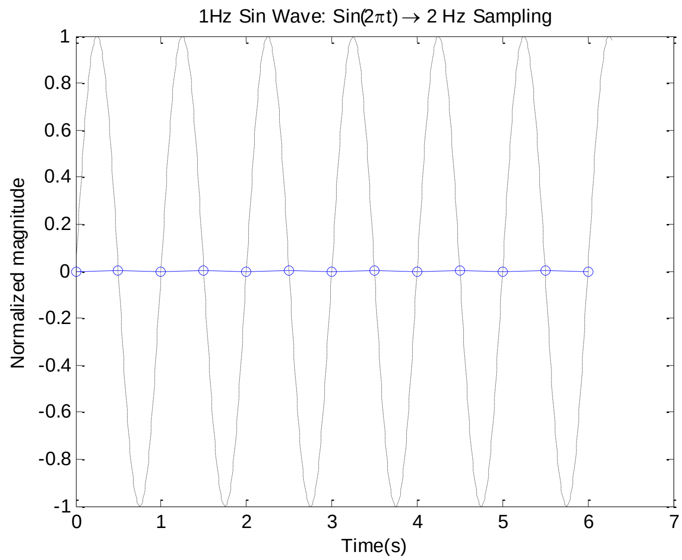


A signal with a reduced frequency is recovered, i.e., the signal is aliased to a lower frequency (we recover a replica)

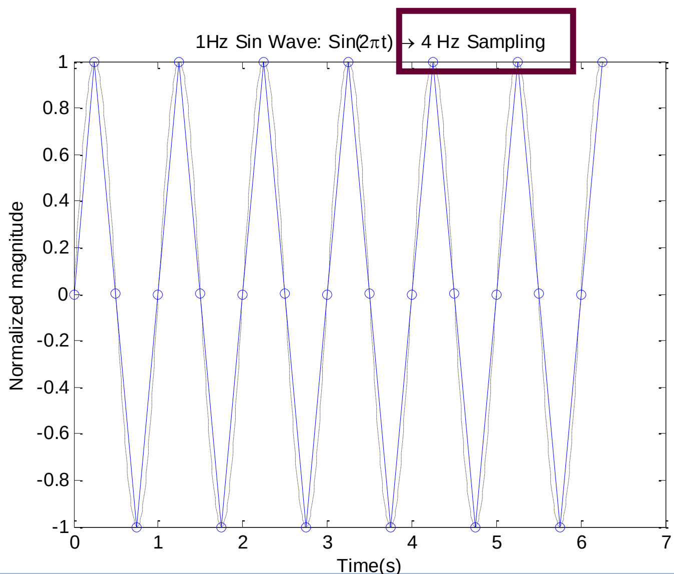
Sampling < Nyquist → Aliasing



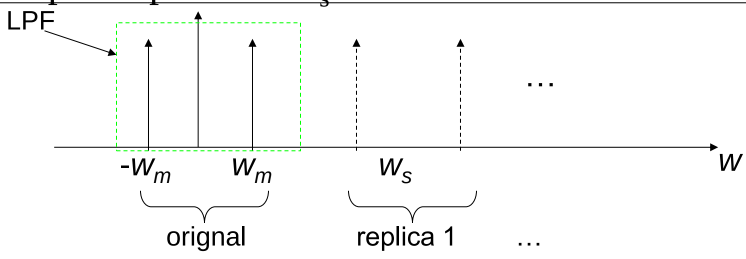
Nyquist is not enough …



A little more than Nyquist is not enough …

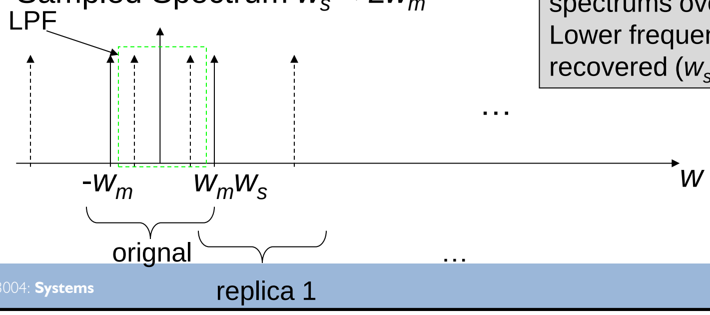


Sampled Spectrum $w_s > 2wm$



original freq recovered

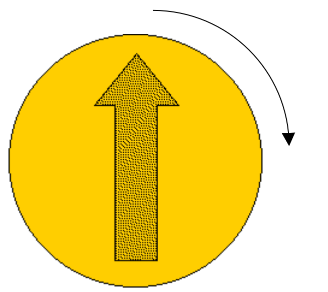
Sampled Spectrum $w_s < 2w_m$



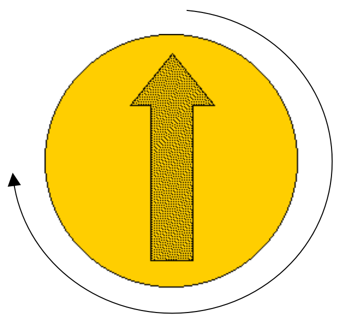
Original and replica spectrums overlap
Lower frequency recovered ($w_s – w_m$)

Temporal Aliasing

90^o clockwise rotation/frame
clockwise rotation perceived



270^o clockwise rotation/frame
(90^o) anticlockwise rotation
perceived i.e., aliasing



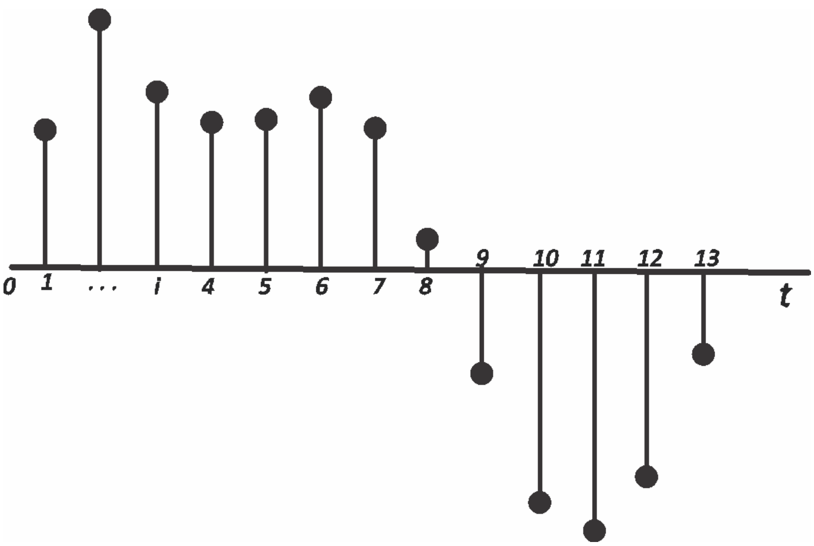
Require LPF to 'blur' motion

Time Domain Analysis of Reconstruction

- Frequency domain: multiply by ideal LPF
  - ideal LPF: ‘rect’ function (gain Δt, cut off w_c)
  - removes replica spectrums, leaves original
- Time domain: this is equivalent to
  - convolution with ‘sinc’ function
  - as $F^{-1}\{\Delta t \ \mathrm{rect}(w/w_c)\} = \Delta t \, w_c \, \mathrm{sinc}(w_c t/\pi)$
  - i.e., weighted sinc on every sample
- Normally, $w_c = w_s/2$

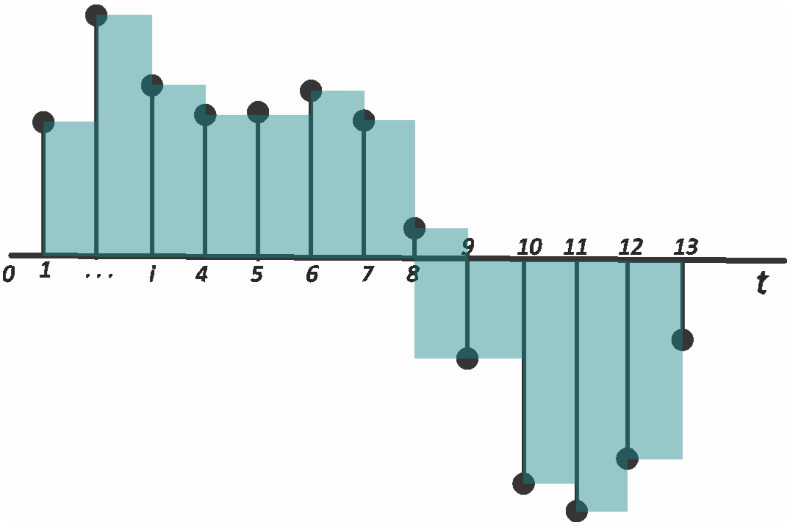
$$x_r(t) = \sum_{n=-\infty}^{\infty} x(n\Delta t)\Delta t w_c \,\mathrm{sinc}\left(\frac{w_c(t - n\Delta t)}{\pi}\right)$$

Reconstruction



Reconstruction

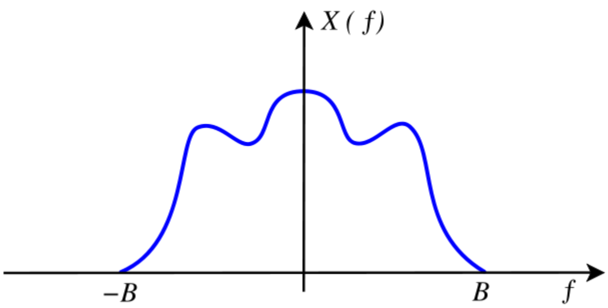
- Zero-Order Hold [ZOH]



Reconstruction

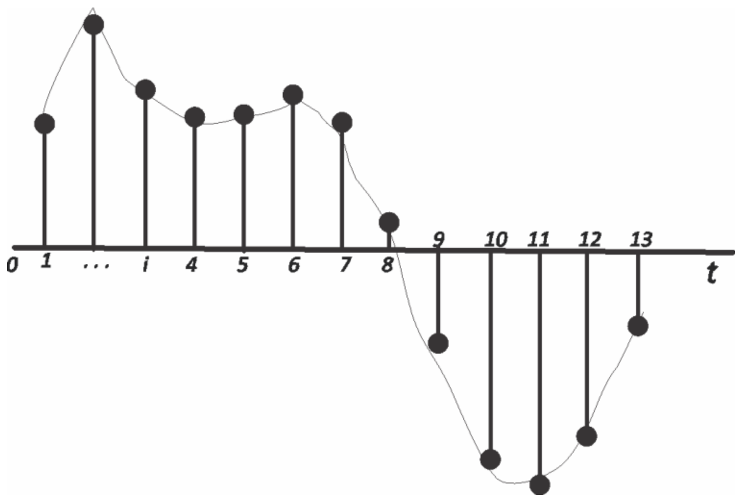
- Whittaker–Shannon interpolation formula

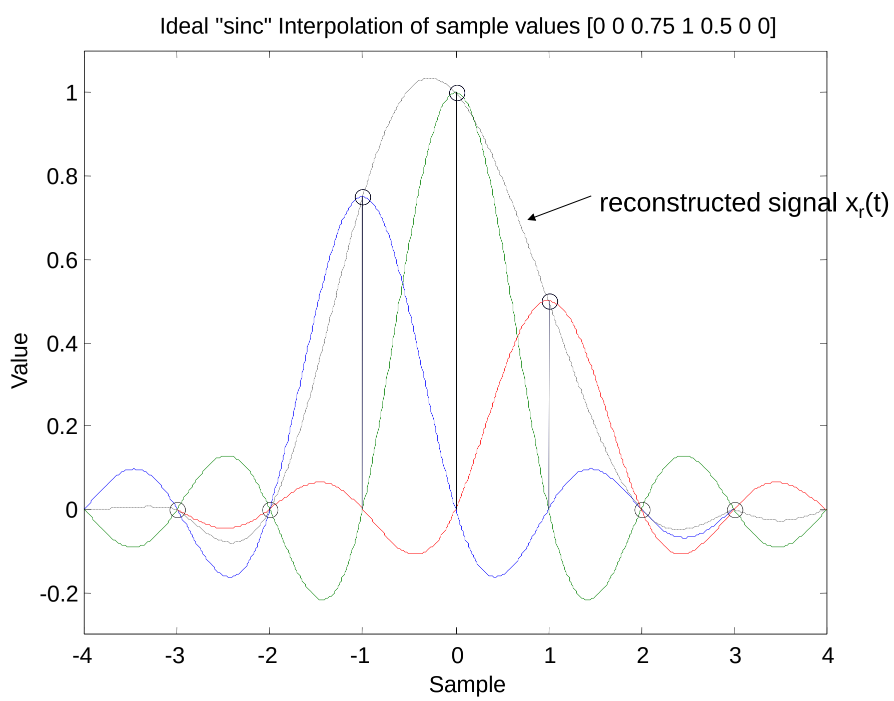
$$x(t) = \sum_{n=-\infty}^{\infty} x[n] \cdot \mathrm{sinc}\left(\frac{t-nT}{T}\right)$$



Reconstruction

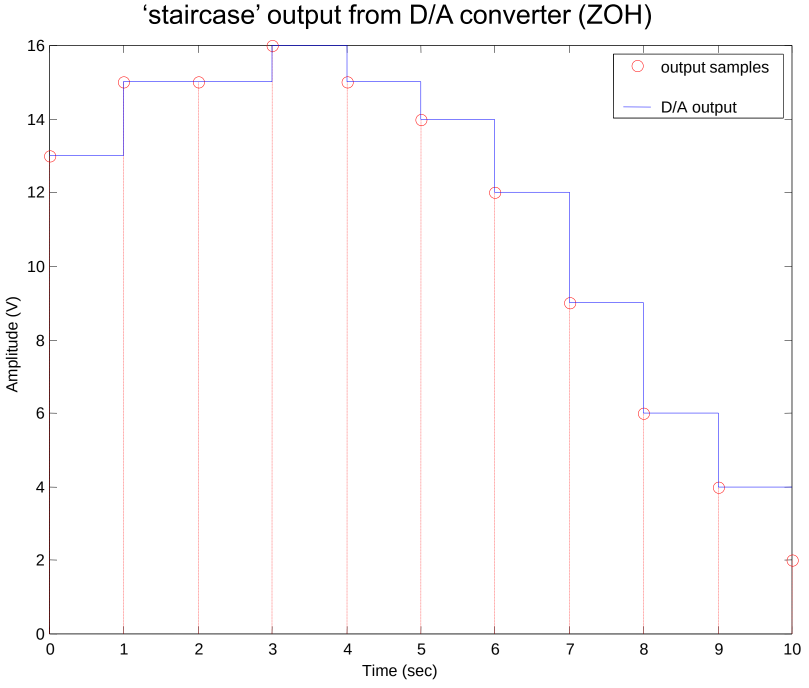
- Whittaker–Shannon interpolation formula



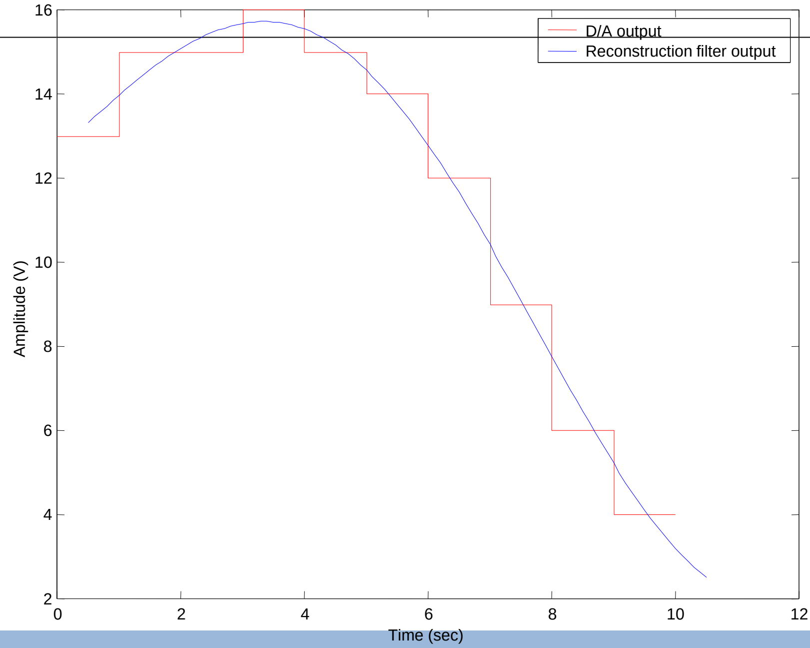


Sampling and Reconstruction Theory and Practice

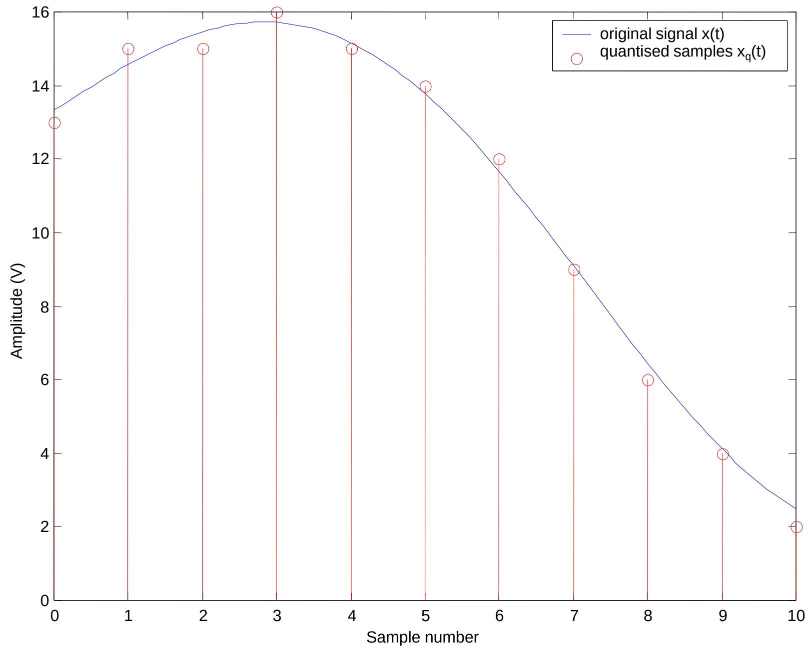
- Signal is bandlimited to bandwidth WB
  - Problem: real signals are not bandlimited
    - Therefore, require (non-ideal) anti-aliasing filter
- Signal multiplied by ideal impulse train
  - problems: sample pulses have finite width
  - and not ⊗ in practice, but sample & hold circuit
- Samples discrete-time, continuous valued
  - Problem: require discrete values for DSP
    - Therefore, require A/D converter (quantisation)
- Ideal lowpass reconstruction (‘sinc’ interpolation)
  - problems: ideal lowpass filter not available
    - Therefore, use D/A converter and practical lowpass filter

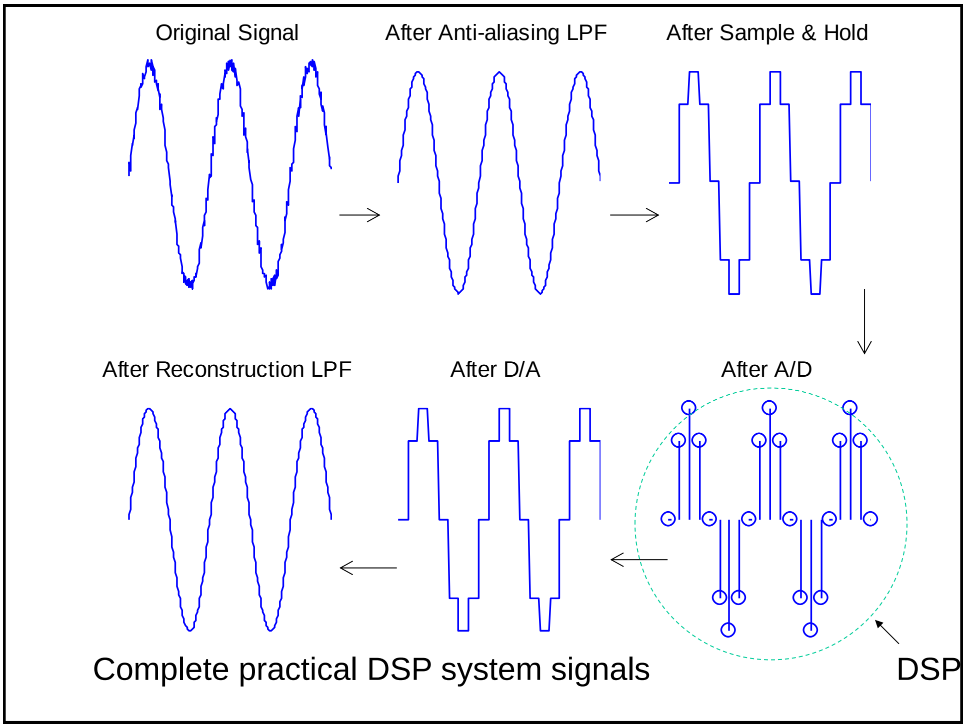


Smooth output from reconstruction filter



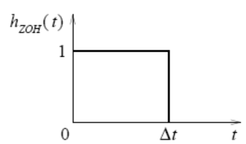
Example: error due to signal quantisation



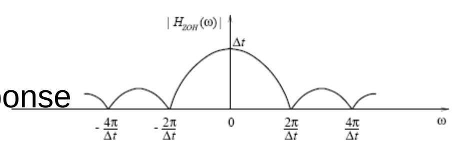


Zero Order Hold (ZOH)

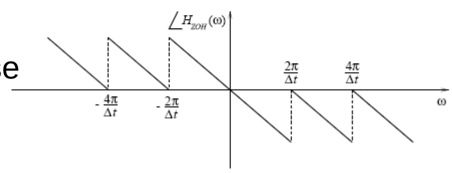
ZOH impulse response



ZOH amplitude response



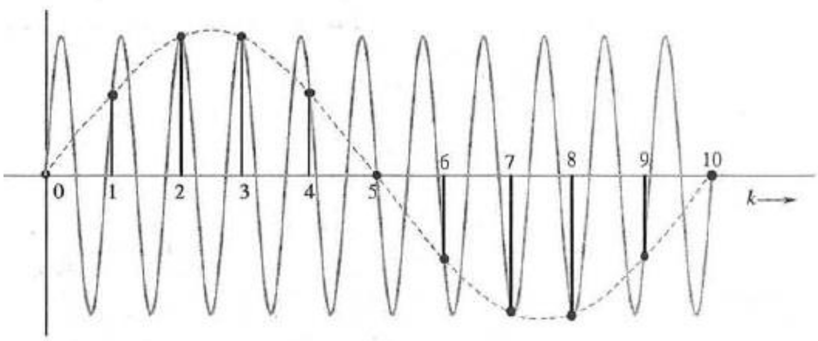
ZOH phase response



Finite Width Sampling

- Impulse train sampling not realisable
  - sample pulses have finite width (say nanosecs)
- This produces two effects,
- Impulse train has sinc envelope in frequency domain
  - impulse train is square wave with small duty cycle
  - Reduces amplitude of replica spectrums
    - smaller replicas to remove with reconstruction filter ☺
- Averaging of signal during sample time
  - effective low pass filter of original signal
    - can reduce aliasing, but can reduce fidelity ☹
    - negligible with most S/H ☺

Aliasing: Another view of this



Alliasing

- Aliasing - through sampling, two entirely different analog sinusoids take on the same "discrete time" identity

For f[k]=cosΩk,   Ω=ωT:

The period has to be less than Fh (highest frequency): $T \leq \frac{1}{2\mathcal{F}_h}$

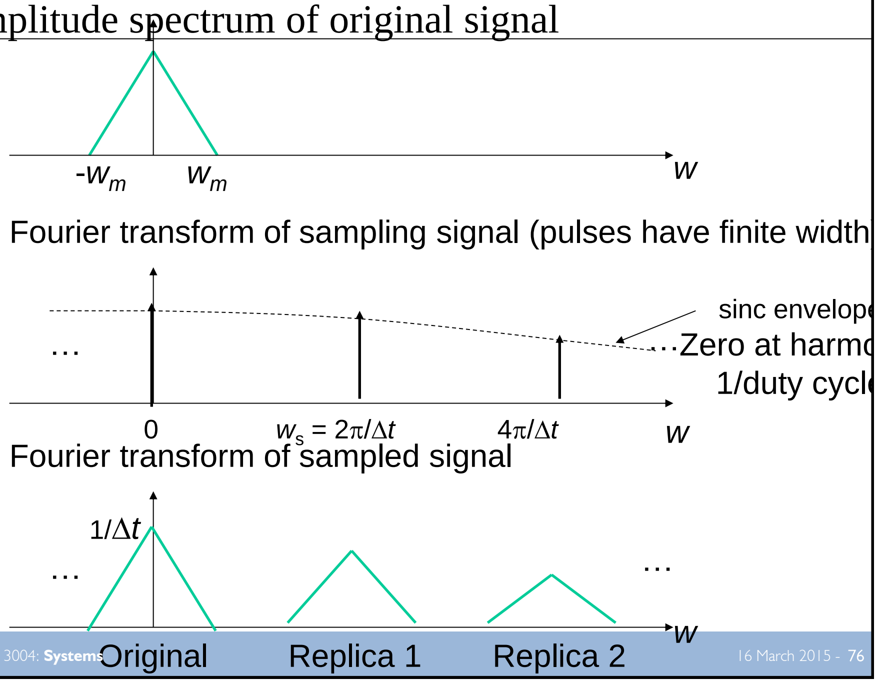
Thus: $0 \leq \mathcal{F} \leq \frac{\mathcal{F}_s}{2}$

ω_f: aliased frequency: $\omega T = \omega_f T + 2\pi m$

Practical Anti-aliasing Filter

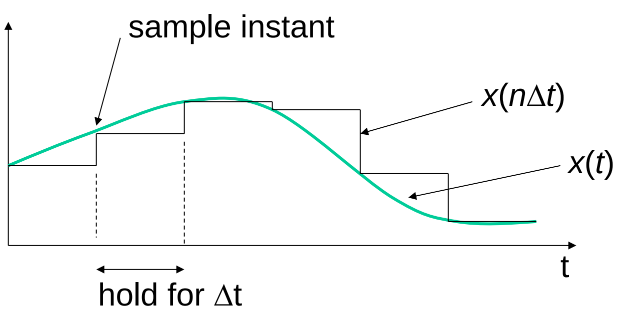
- Non-ideal filter
  - wc = ws /2
- Filter usually 4th – 6th order (e.g., Butterworth)
  - so frequencies > wc may still be present
  - not higher order as phase response gets worse
- Luckily, most real signals
  - are lowpass in nature
    - signal power reduces with increasing frequency
  - e.g., speech naturally bandlimited (say < 8KHz)
  - Natural signals have a (approx) 1/f spectrum
  - so, in practice aliasing is not (usually) a problem

Amplitude spectrum of original signal



Practical Sampling

- Sample and Hold (S/H)
  1. takes a sample every Δt seconds
  2. holds that value constant until next sample
- Produces 'staircase' waveform, $x(n\Delta t)$



Quantisation

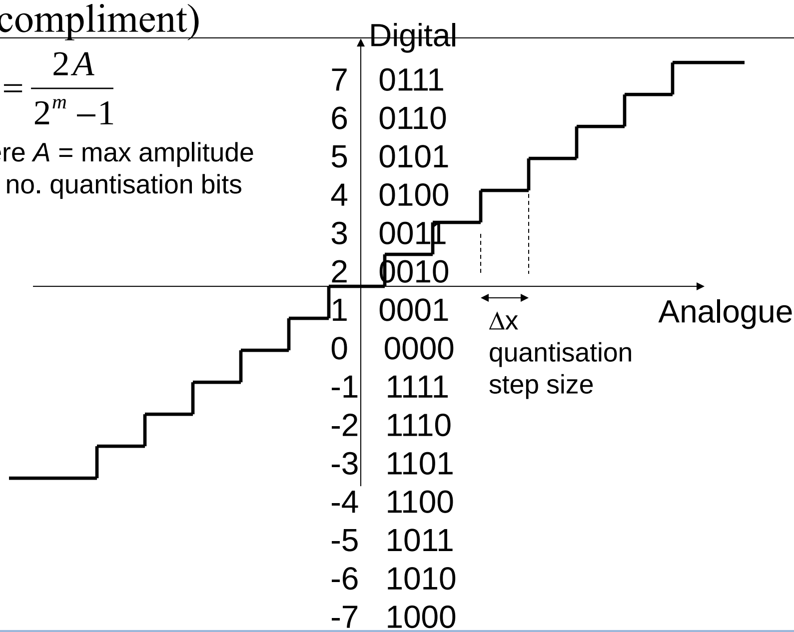
- Analogue to digital converter (A/D)
  - Calculates nearest binary number to $x(n\Delta t)$
    - $x_q[n] = q(x(n\Delta t))$, where q() is non-linear rounding fctn
  - output modeled as $x_q[n] = x(n\Delta t) + e[n]$
- Approximation process
  - therefore, loss of information (unrecoverable)
  - known as 'quantisation noise' ($e[n]$)
  - error reduced as number of bits in A/D increased
    - i.e., Δx, quantisation step size reduces

$$|e[n]| \leq \frac{\Delta x}{2}$$

Input-output for 4-bit quantiser (two's compliment)

$$\Delta x = \frac{2A}{2^m - 1}$$

where A = max amplitude
m = no. quantisation bits



Signal to Quantisation Noise

- To estimate SQNR we assume
  - *e*[*n*] is uncorrelated to signal and is a
  - uniform random process
- assumptions not always correct!
  - not the only assumptions we could make…
- Also known a 'Dynamic range' (R_D)
  - expressed in decibels (dB)
  - ratio of power of largest signal to smallest (noise)

$$R_D = 10\log_{10}\left(\frac{P_{signal}}{P_{noise}}\right)$$

Dynamic Range

Need to estimate:

1. Noise power
   - uniform random process: $P_{noise} = \Delta x^2/12$
2. Signal power
   - (at least) two possible assumptions
   1. sinusoidal: $P_{signal} = A^2/2$
   2. zero mean Gaussian process: $P_{signal} = \sigma^2$
      - Note: as $\sigma \approx A/3$: $P_{signal} \approx A^2/9$
      - where σ^2 = variance, A = signal amplitude

1 extra bit halves Δx
i.e., 20log10(1/2) = 6dB

Regardless of assumptions: R_D increases by 6dB for every bit that is added to the quantiser

Practical Reconstruction

Two stage process:

1. Digital to analogue converter (D/A)
   - zero order hold filter
   - produces 'staircase' analogue output
2. Reconstruction filter
   - non-ideal filter: $w_c = w_s/2$
   - further reduces replica spectrums
   - usually $4^{th} - 6^{th}$ order e.g., Butterworth
     - for acceptable phase response

D/A Converter

- Analogue output y(t) is
  - convolution of output samples $y(n\Delta t)$ with $h_{ZOH}(t)$

$$y(t) = \sum_n y(n\Delta t) h_{ZOH}(t - n\Delta t)$$

$$h_{ZOH}(t) = \begin{cases} 1, & 0 \le t < \Delta t \\ 0, & \text{otherwise} \end{cases}$$

$$H_{ZOH}(w) = \Delta t \exp\left(\frac{-jw\Delta t}{2}\right) \frac{\sin(w\Delta t/2)}{w\Delta t/2}$$

D/A is lowpass filter with sinc type frequency response
It does not completely remove the replica spectrums
Therefore, additional reconstruction filter required

Summary

- Theoretical model of Sampling
  - bandlimited signal (wB)
  - multiplication by ideal impulse train (ws > 2wB)
    - convolution of frequency spectrums (creates replicas)
  - Ideal lowpass filter to remove replica spectrums
    - wc = ws /2
    - Sinc interpolation
- Practical systems
  - Anti-aliasing filter (wc < ws /2)
  - A/D (S/H and quantisation)
  - D/A (ZOH)
  - Reconstruction filter (wc = ws /2)

Don't confuse theory and practice!