Administrative

- MATLAB1 due @ midnight tonight
  - Turnitin problems? Check 'howto' guide in Lecture4 folder on Trunk. Most importantly, use the shibboleth website

- MATLAB2 is open on Trunk, due midnight 10/2 (related to today's lecture)

- See HW_Sampling.pdf in Lecture 5 folder – 2 small problems due next Monday

Tufts

Updated schedule, from Trunk

Date		Unit	Topic	book section # (P&M)	OUT-MATLAB	IN-MATLAB
6-Sep	1	review	Course overview; LTI systems	2.1-2.2	MATLAB1	
11-Sep	2	review	Convolution, start Z transform	2.3-2.4		
13-Sep	3	review	Z, Fourier transform	3.1-3.2; 4.1, 4.4		
18-Sep	4	review	Sampling	6.1, 6.2		
20-Sep	5	"review"	Reconstruction	"	MATLAB2	MATLAB1
25-Sep	6	LTI systems	LTI system analysis using Z-transform; Rational systems	3.5, some of 3.6		
27-Sep	7	LTI systems	LTI systems analysis using the Fourier transform	5.1,5.2		
2 Oct	8	LTI systems	Phase and group delay, geometric interpretation	5.4		MATLAB2
4-Oct	9	LTI systems	**15 min quiz, lectures 1-6**. Filter design by pole-zero placement, common simple filters	5.4	MATLAB 3	

Tufts

EE-125:
Digital Signal Processing

Lecture 5:
Review of Sampling (A/D) and Reconstruction (D/A)

Professor Tracey

Tufts

Lecture 5: Outline

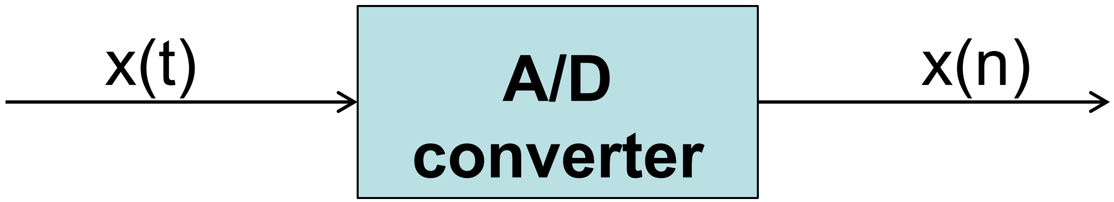
- Finish up sampling from (A/D)
- Signal reconstruction (Digital-to-analog, or D/A)
  - –Idealized vs. practical reconstruction
- Aliasing

Link to book:

- Idealized sampling and reconstruction is P&M 6.1
- Practical A/D and D/A is in 6.3

- **(Bonus topic)** Band-limited sampling (P&M 6.4.1)
  Worth knowing about, but not on HW/tests

Questions (from last lecture)



Continuous time (CT)

$x(t)$, $X(f)$

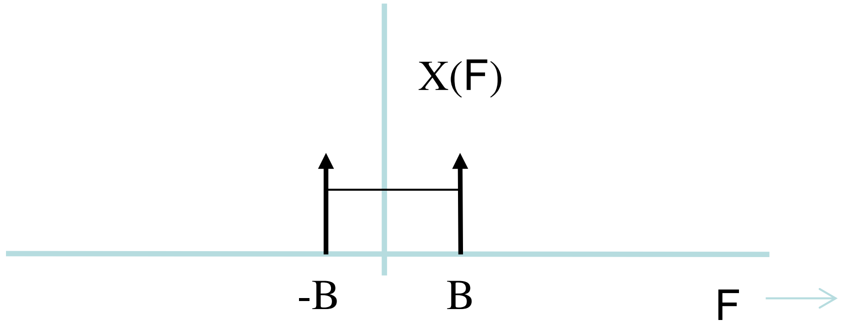
Discrete time (DT)

$x(n)$, $X(\omega)$

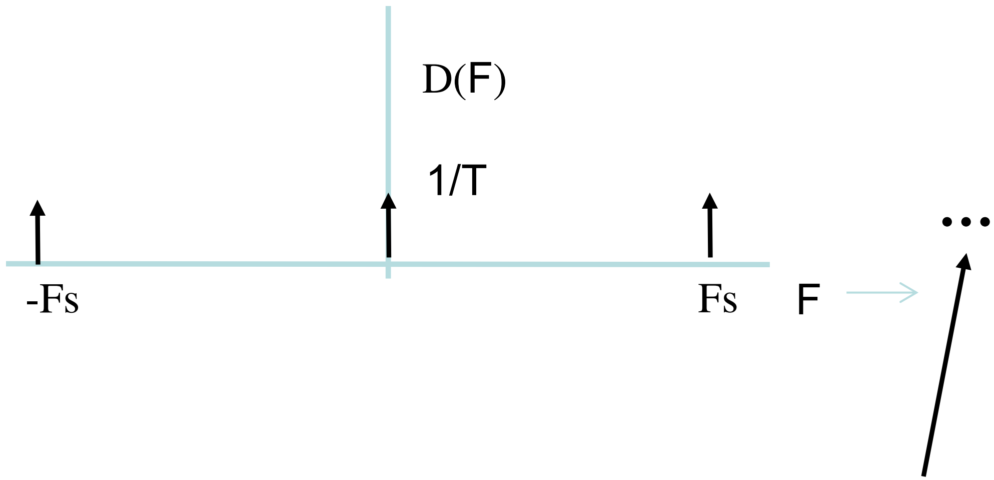
1) How are CT quantities (time t, frequency f) related to DT quantities (sample n, radial frequency ω)?
2) How does the process of sampling affect the frequency response?

Sampling, in frequency domain

1) We start with the true CT signal, in continuous frequency. Here the signal is a cosine (peaks) plus other content



2) We take samples at t = nT, (time domain pulse train) where T = 1/Fs, the sampling rate. Mathematically, we multiply x(t) by D(t), a pulse train. D(F) is shown here.

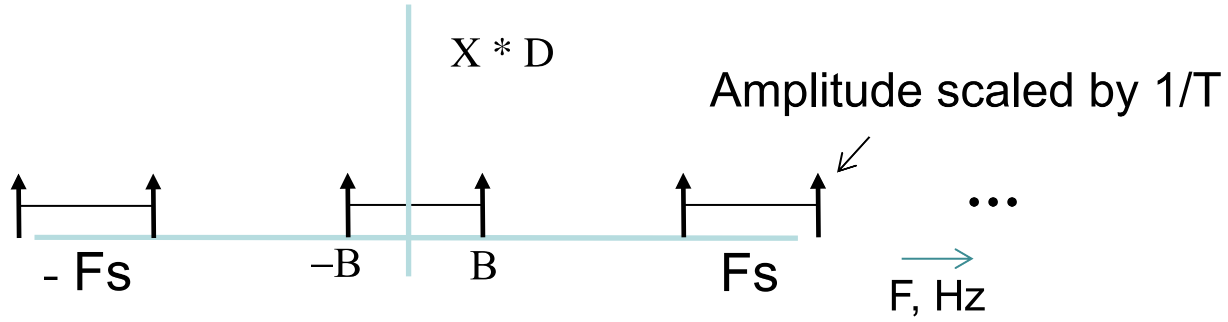


Infinite # of copies at +/- 2 Fs, +/- 3 Fs, etc

Sampling - continued

3) The sampled signal's spectrum is X * D.

- Periodic w/ period Fs
- Get overlap if B>Fs/2

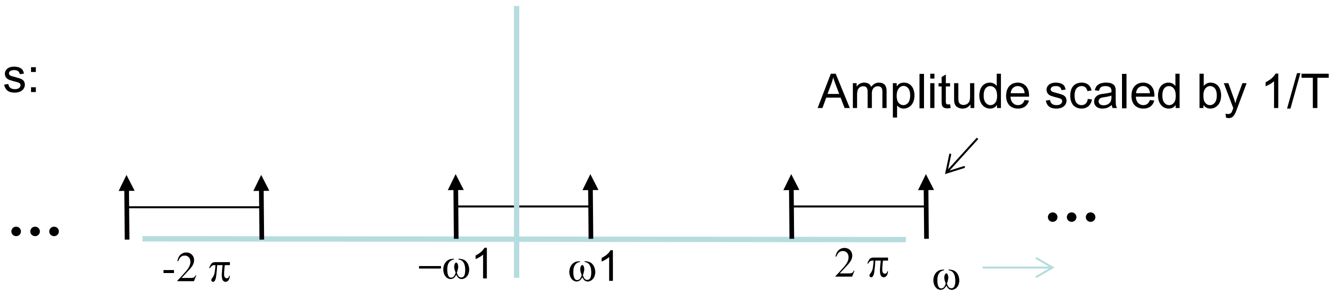


4) Mapping from F to ω is:

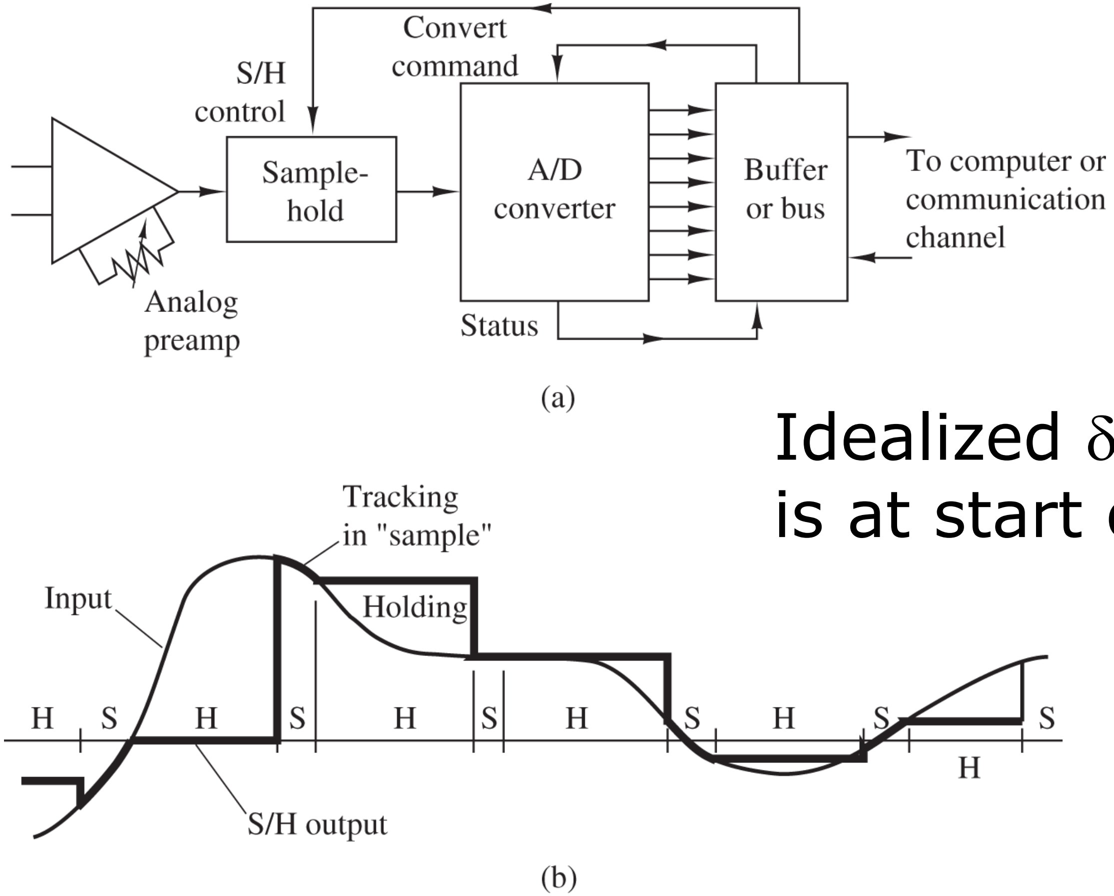
Fs -> 2π

Fs/2 ->π

B-> 2 π B/Fs = ω1



Practical A/D: Sample and Hold



Idealized $\delta(t)$ sample is at start of 'hold'

Figure 6.3.1 (a) Block diagram of basic elements of an A/D converter; (b) time-domain response of an ideal S/H circuit.

Practical A/D: quantization

- A/D output is in integer "counts", or levels
- Spacing of levels is the quantization step:
  Δ = (Xmax-Xmin)/(L-1)

 for L levels (figure shows L=8)

Lots more detail in P&M 6.3

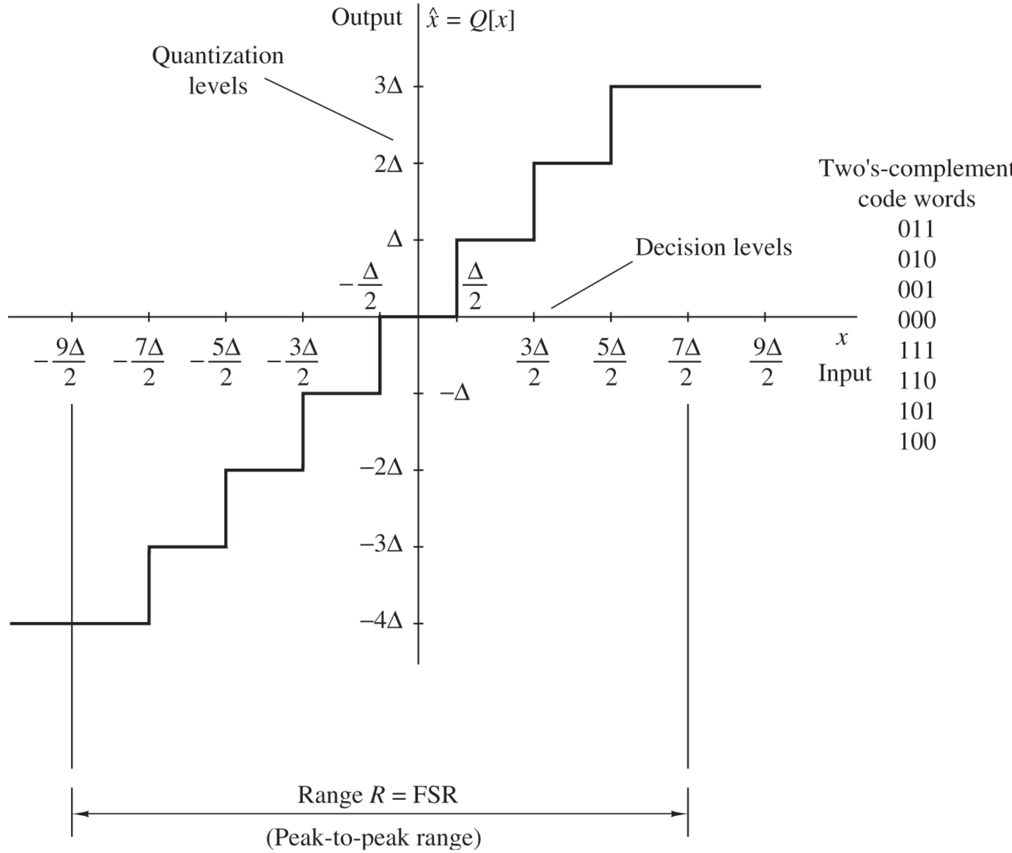


Figure 6.3.3 Example of a midtread quantizer.

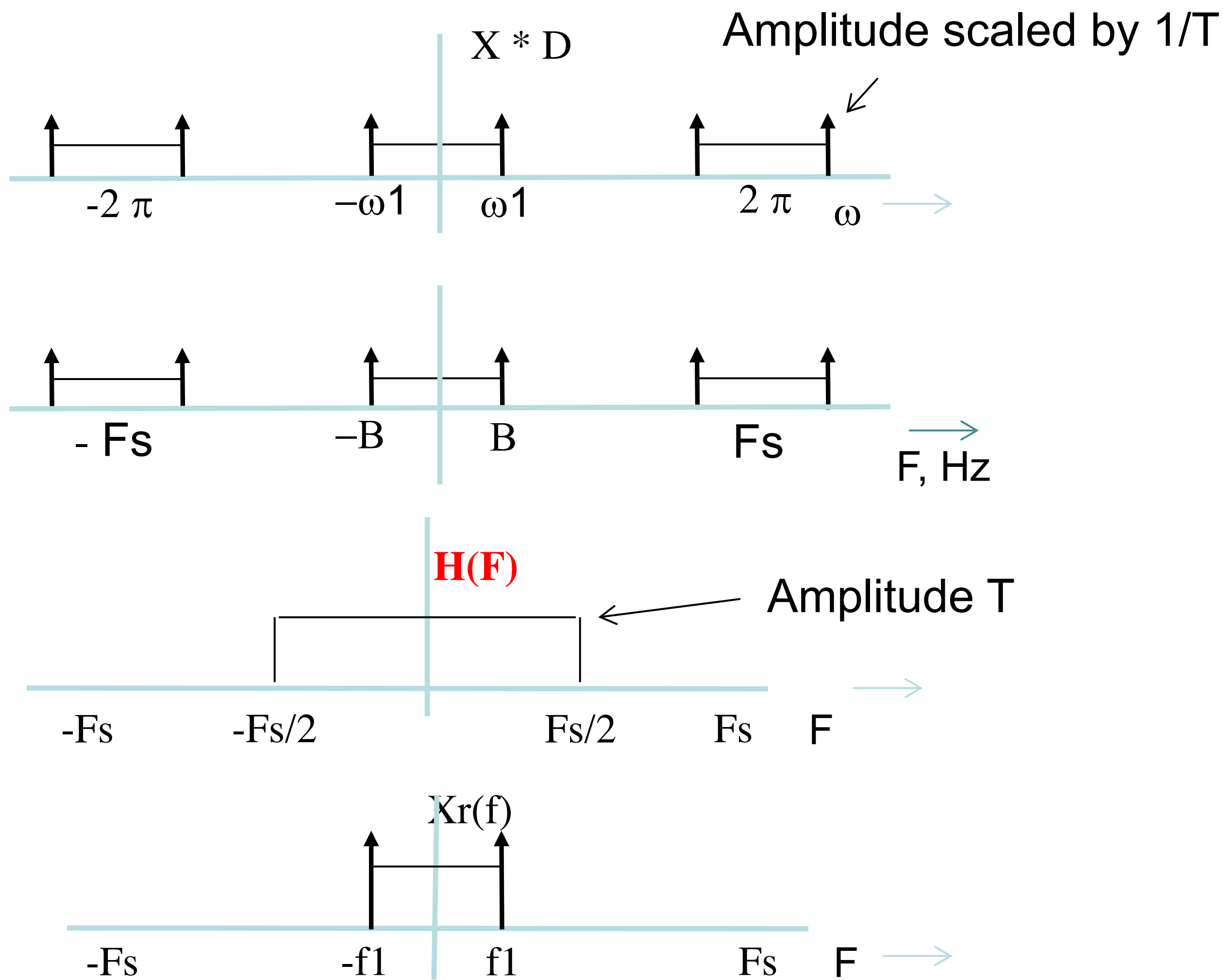
Reconstruction (D/A)
Really simple in pictures (in freq)….

The sampled signal's spectrum is X * D.

Mapping from ω to F:
- Fs -> 2π
- Fs/2 -> π
- f1 -> 2π f1/Fs = $\omega 1$

For idealized reconstruction, we use a perfect **ANALOG** low pass filter (LPF)…

… thus recovering the original spectrum



Sinc-based interpolation during reconstruction

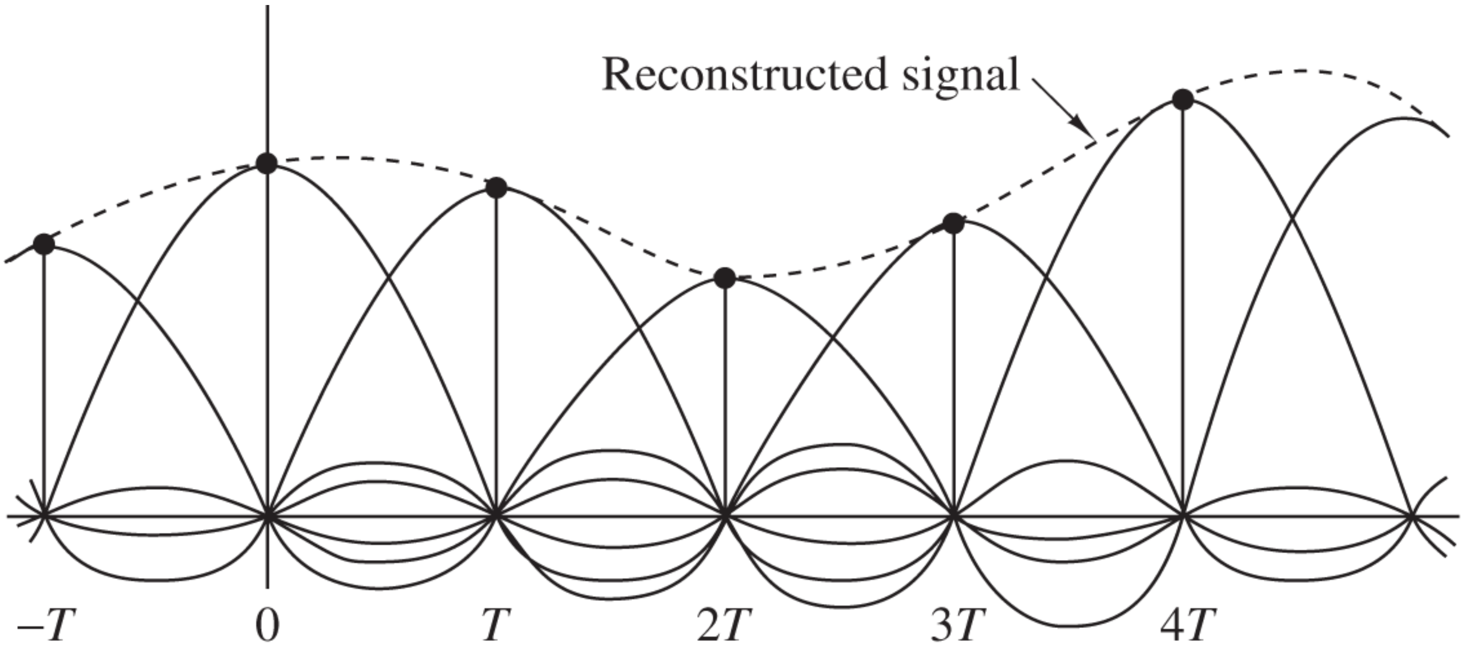


Figure 6.1.2 Reconstruction of a continuous-time signal using ideal interpolation.

See ‘recon.pdf’ on Trunk for derivation

Practical D/A: Sample & Hold + LPF

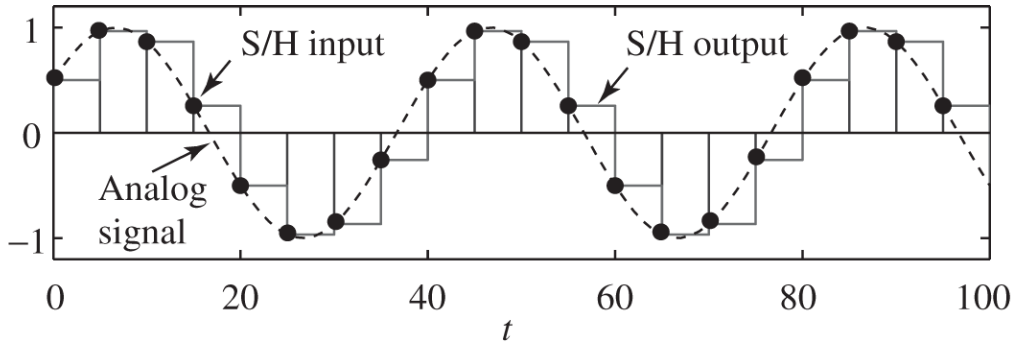


Figure 6.3.8 Response of an S/H interpolator to a discrete-time sinusoidal signal.

Then, pass S/H output through a low-pass filter to smooth out the rough edges

This is a fairly simple system that can be analyzed exactly….

Setup for Matlab2, 2^{nd} part

- D/A converters (this lecture): convert x(n) to x(t) by interpolating the original samples x(n) (at times t = nT) to other times t

- Matlab2, 2^{nd} part: goal is to study reconstruction without having you build hardware!
  - Write code that interpolates the original samples x(n) (at times t = nT) to other discrete time points between the original samples
  - A picture should help…

Sinc-based interpolation during reconstruction

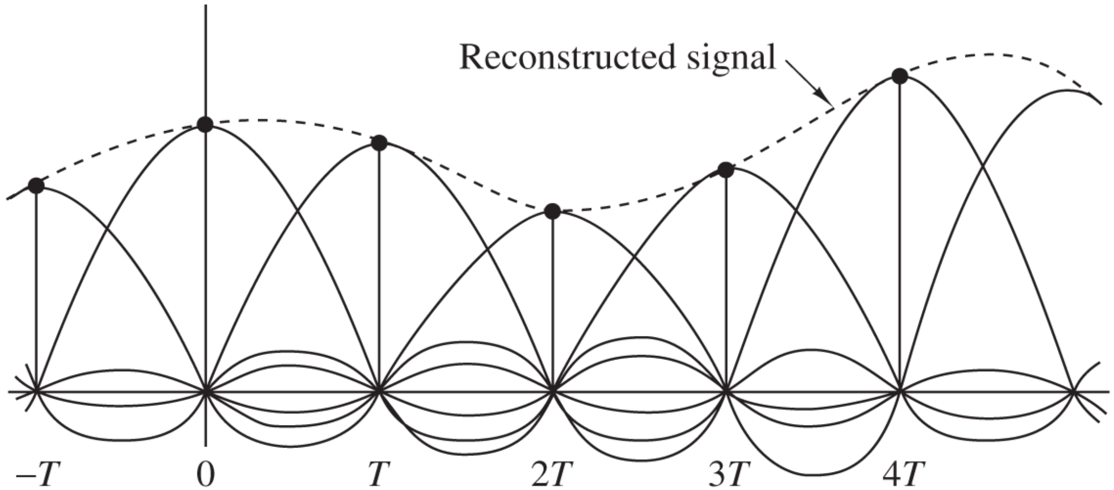


Figure 6.1.2 Reconstruction of a continuous-time signal using ideal interpolation.

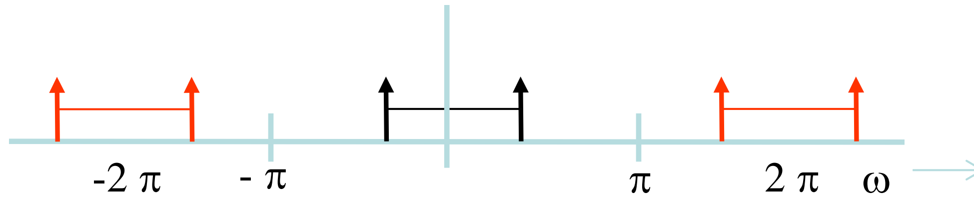
Question: how would linear interpolation in time look?

Aliasing (related to part 1 of MATLAB2)

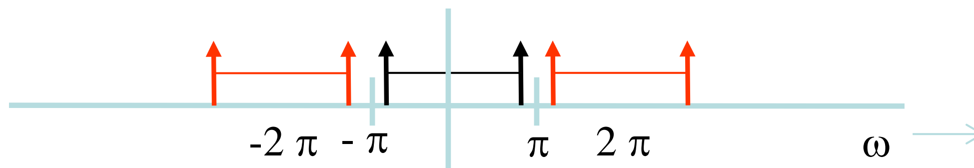
- Aliasing of sinusoidal components
- Aliasing of non-bandlimited signals
  - Definition of bandlimited
- Need for prefiltering before A/D

Sampling refresher – Aliasing in the frequency domain (DTFT)

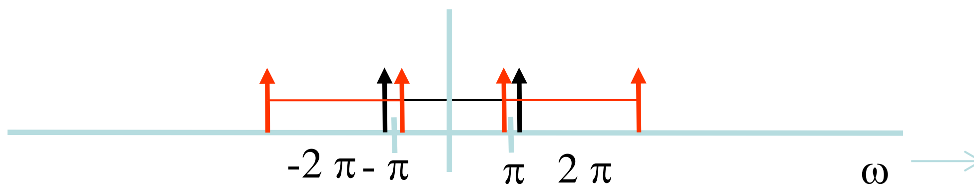
1) Here we are well sampled – high Fs



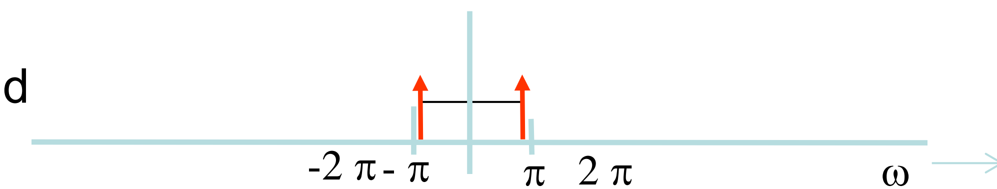
2) Here we've lowered sampling rate so Fs/2 ($\omega=\pi$) is just a little larger than the max frequency



3) Here we've lowered sampling rate so Fs/2 ($\omega=\pi$) is < signal frequency; signal is **aliased**



4) After low-pass filtering and reconstruction, **sinusoid appears at lower frequency**



[Video example of aliasing](https://www.youtube.com/watch?v=lEOF9VDE_kk)

https://www.youtube.com/watch?v=lEOF9VDE_kk

Aliasing of non-bandlimited signals (see example in book)

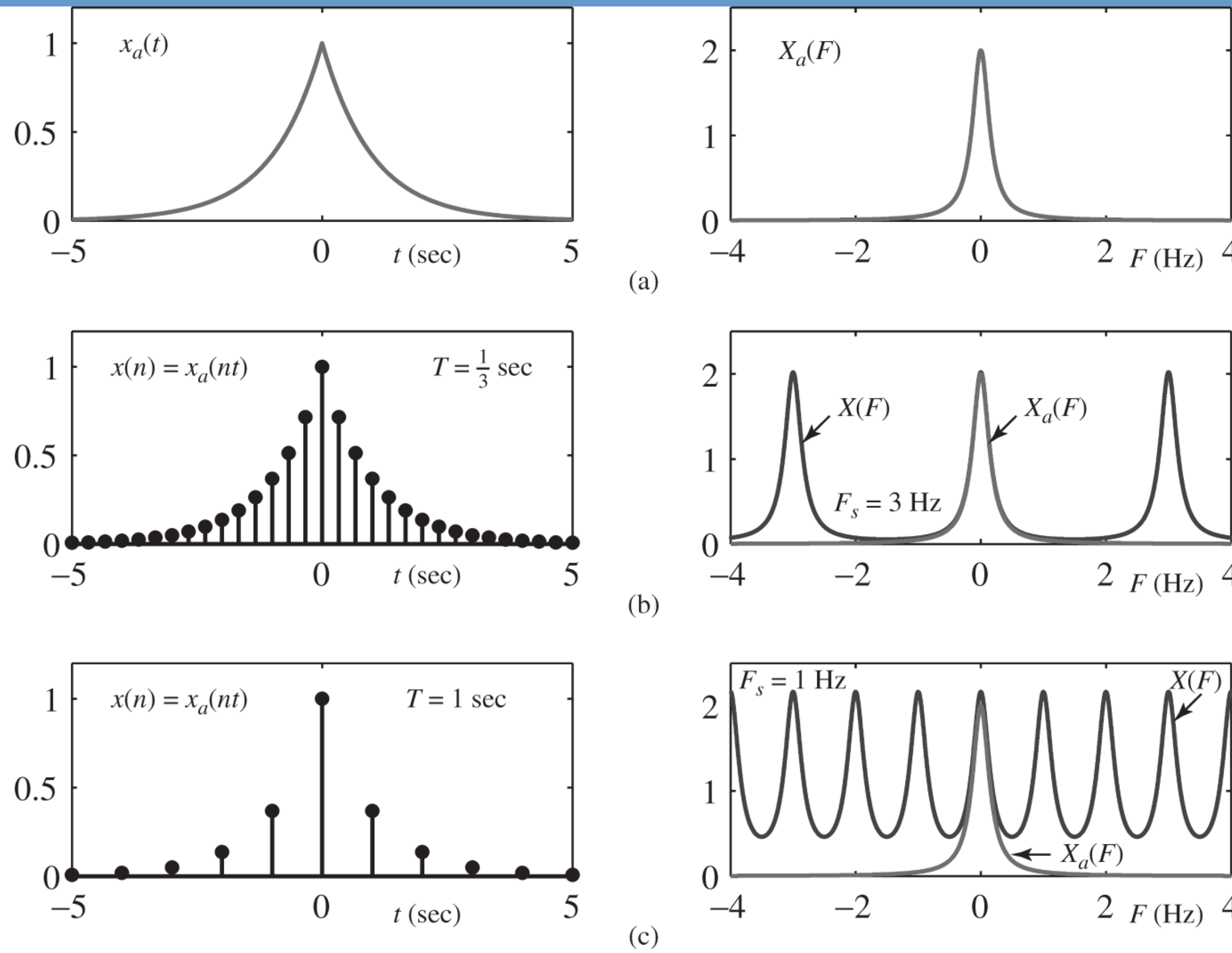


Figure 6.1.7 (a) Analog signal $x_a(t)$ and its spectrum $X_a(F)$; (b) $x(n) = x_a(nT)$ and its spectrum for $F_s = 3$ Hz; and (c) $x(n) = x_a(nT)$ and its spectrum for $F_s = 1$ Hz.

Lecture 5: Outline

- Review of sampling from last week (A/D)
- Signal reconstruction (Digital-to-analog, or D/A)

- Band-limited sampling (P&M 6.4.1)
  - For reading only…

A quick question

- True or false:

  The Nyquist criterion says that we should sample at twice the highest frequency in the data

A quick question

- True or false:

  The Nyquist criterion says that we should sample at twice the highest frequency in the data

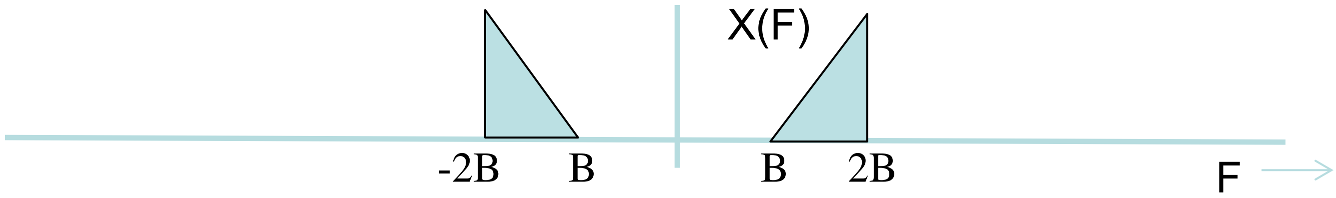
- It says to sample at Fs>2B

  where B is bandwidth – not necessarily the highest frequency

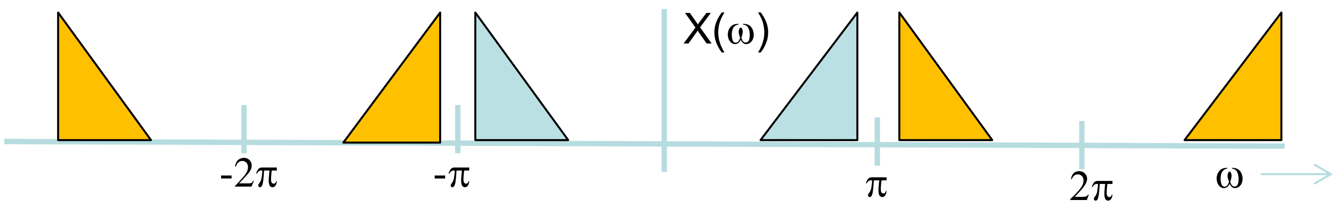
Sampling/reconstruction of bandpass signals

- We often think of the sampling theorem as stating "sample at twice the highest frequency of the signal"
- Really, it states "sample at twice the bandwidth".
  - For most signals, the bandwidth is from 0 Hz to the highest frequency of interest
  - However, when signals are *bandpass* (0 outside some range $F1<F<F2$) we can exploit this
- See section 6.4 for discussion of bandpass sampling
  - Integer band positioning (6.4.1) will be explored as part of MATLAB1.
  - Arbitrary band positioning is much more involved but same general idea

Integer band sampling, graphically

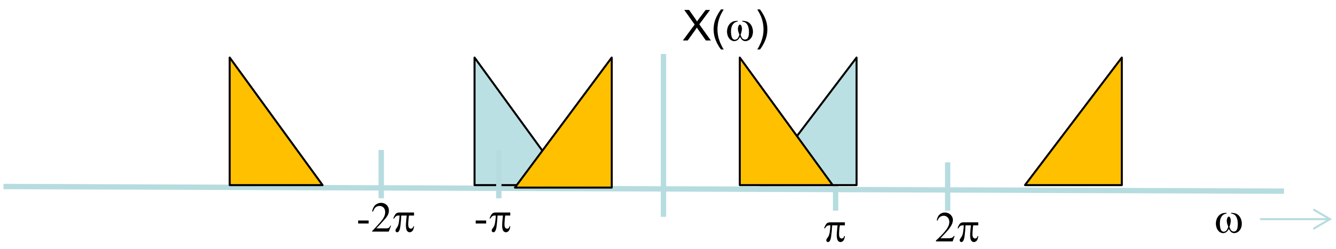


This bandpass signal is an example of integer band positioning, with highest frequency = integer multiple of the bandwidth; Fh = mB (here m=2)

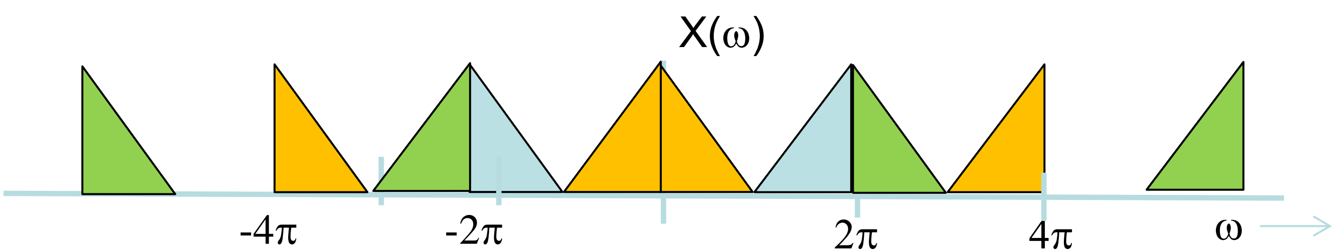


Sampled signal, Fs > 2*2B (>twice highest frequency). Here, the periodic repetitions of the signal (orange) don't encroach into the [-pi, pi] region.

Integer band sampling, graphically



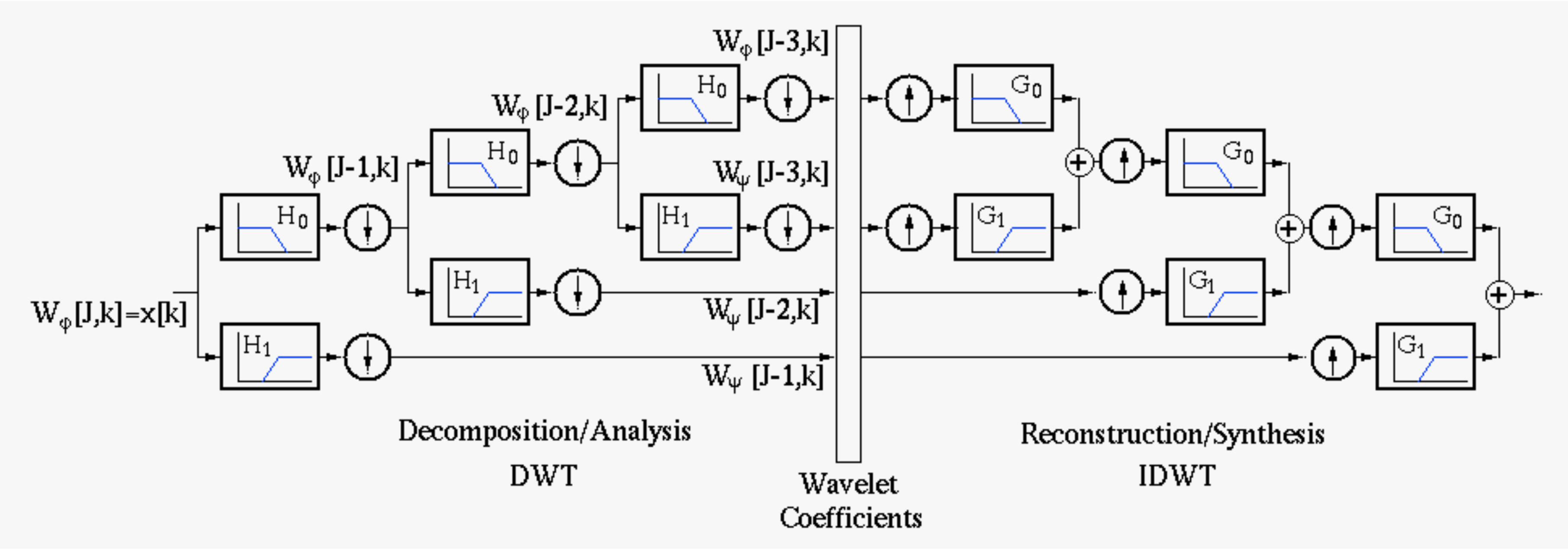
Sampled signal, lower sample rate: 2B < Fs < 2*2B. Aliasing is clear. Note, plot doesn't show signals centered around +/- 4 pi, etc.



Sampled signal, even lower sample rate: Fs = 2B. No overlap, so signal can be recovered!
Here, showing signals centered around +/- 4 pi in green, but not +/- 6 pi – those would fill in the other gaps

Tufts

Wavelet decomposition – one use of bandpass sampling

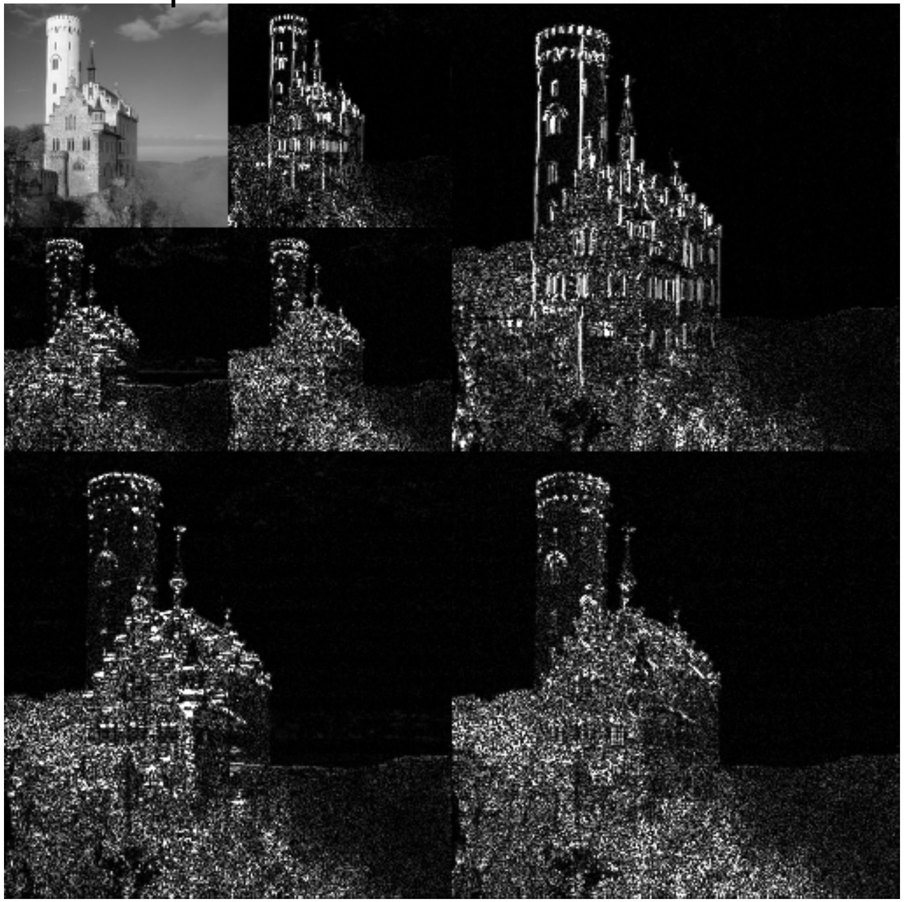


- With wavelets, we split a iteratively split a signal into low-pass and high-pass components. If the filters are designed correctly (special filters needed) then we can reconstruct the signal
- Bandpass sampling says that the highpass and lowpass signals at each stage can sampled at half the rate of the previous stage (as each contains half the bandwidth).
- This limits the data storage needed

Wavelet compression (JPEG2000)

- Example from image processing.
- The image can be split iteratively into bandpass versions w/o increasing storage
- Because many wavelet coefficients are zero, we don't need to store them – this leads to compression

Low freq



High freq

Tufts