DSP Lecture 14

Continuous-time filtering with digital systems; upsampling and down sampling.

Last time is Sampling theorem

recall a continuous-time filter, How to do it in discrete-time?

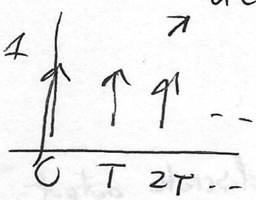
→ How to change the Sampling rate? A Rate conversion.

How to make two ~~systems~~ with different Rates Talk To Each other?

continuous-time Sampled signal

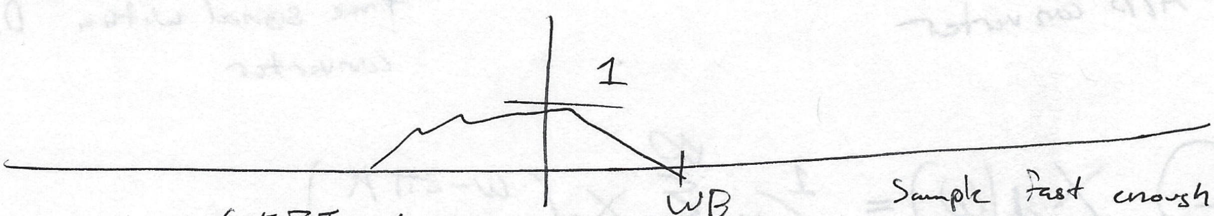
$$x_c(t) \longrightarrow \otimes \longrightarrow x_s(t) \longrightarrow \boxed{\text{Convert Impulse Train Into Discrete time Sequen}} \longrightarrow x[n] = x_c(nT)$$

$d(t)$



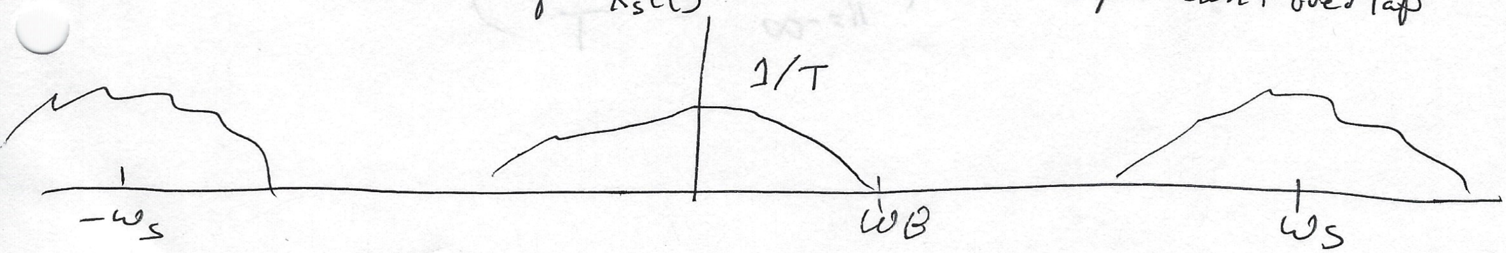
$$X_s(\omega) = \frac{1}{T}\sum_{k=-\infty}^{\infty} X_c(\omega - k\omega_s) \qquad \omega_T = \frac{2\pi}{T}$$

$X_c(\omega)$ CTFT



Sample fast enough that the copies don't overlap

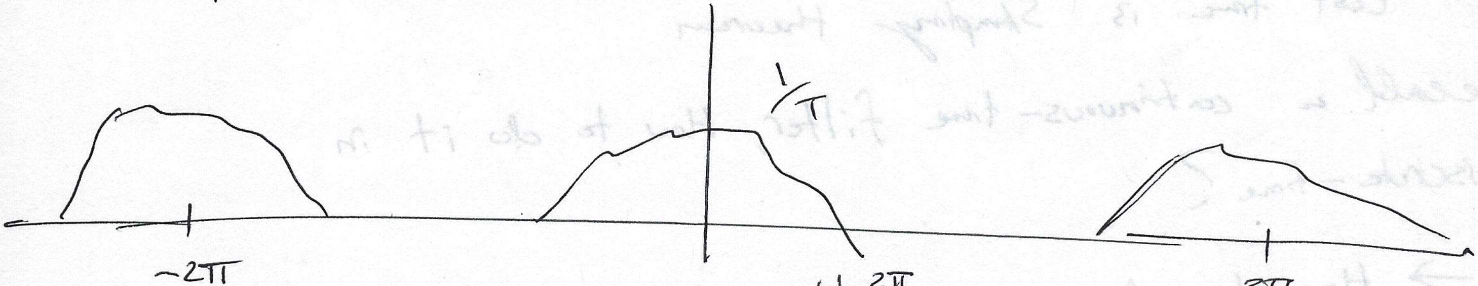
CTFT of $x_s(t)$



② So what of DTFT

DTFT of X[n]

must be 2π periodic, So scale the axis so that The copies occur at 2π instead of the multiples of the Sampling frequency (W_s)



So if we had a bandwidth from before, W_B we need to modify it such that ~~if~~ $W_B \to W_B \frac{2\pi}{W_s} = W_B T$

$$W_s = \frac{2\pi}{T}, \quad \to \underline{T W_s = 2\pi}$$

S: S1

DISCRETE - Time processing of Continuous-time Signals ①

$X_c(t) \to$ C/D $\to X_d[n] \to$ Discrete Time System ② $\to Y_d[n] \to$ D/C ③ $\to Y_r(t)$

C/D: T — D/C: T — Sampling rate

discrete output

Some continuous-time signal coming in

Sample the signal by some sampling rate T

Continuous to Discrete A/D converter

We get a Discrete-time Signal

Process with discrete-Time System

Process back into a continuous time signal with a D/A converter

① $$X_d(W) = \frac{1}{T} \sum_{K=-\infty}^{\infty} X_c\left(\frac{W - 2\pi K}{T}\right)$$

How to reconstruct this signal, the reconstruction filter, go from D to A

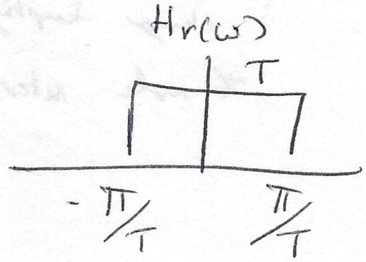
(3)

(3) $\sum_{n=-\infty}^{\infty} Y_d[n] \operatorname{sinc}\left(\frac{t-nT}{T}\right)$ $\qquad Y_r(\omega) = H_r(\omega) Y_d(\omega)$

The spectrum of $Y_d[n]$ to the spectrum of $Y_r(t)$ is ~~just~~ a low pass filter

→ what we want to do in the frequency domain is Notch out the middle copy and Take the Inverse CTFT.

→ $H_r(\omega)$ is a Lowpass filter



(2) is saying $Y_d(\omega) = H(\omega) X_d(\omega)$

↑ Freq response of D.T System

we know

reconstructed filter (Pulse)

$$Y_r(\omega) = H_r(\omega) Y_d(\omega)$$

$$= \cancel{H_r(\omega) H(\omega) X_d(\omega)}$$

$$= H_r(\omega) H(\omega T) X_d(\omega T)$$

$$= H_r(\omega) H(\omega T) \frac{1}{T} \sum_{k=-\infty}^{\infty} X_c\left(\omega - \frac{2\pi k}{T}\right)$$

(4)

IF $X_c(\omega) = 0$ for $|\omega| > \pi/T$, Then

$$Y_r(\omega) = \begin{cases} H(\omega T)\, X_c(\omega) & |\omega| < \pi/T \\ 0 & \text{Else} \end{cases}$$

12:25

So The Effective Continuous-time frequency response is

$$Y_r(\omega) = H_{eff}(\omega)\, X(\omega) \quad (\text{CTFT})$$

discrete time filter scaled by T

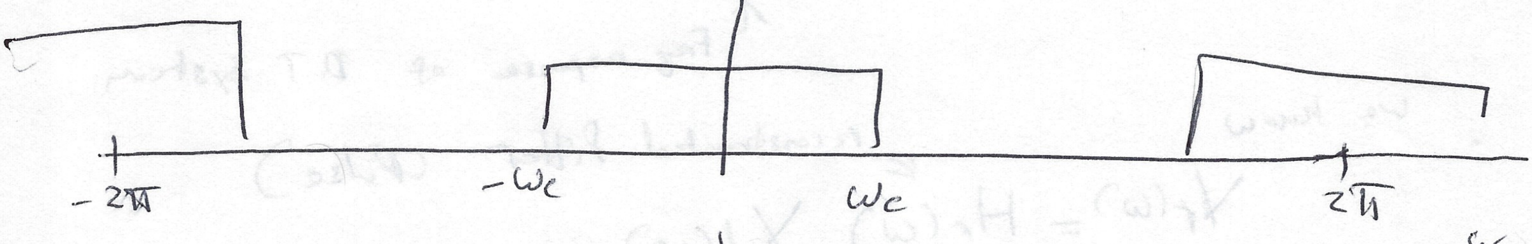
$$\underline{H_{eff}} = \begin{cases} \underline{H(\omega T)} & \underline{|\omega| < \pi/T} \quad \text{raised region} \\ 0 & \text{Else} \end{cases}$$

change sampling rate to make interval wider

---

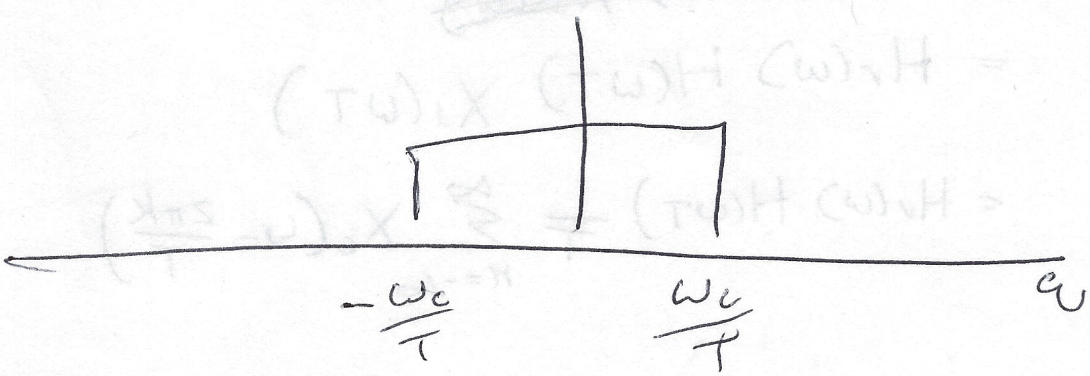
consider $H(\omega)$ as a Digital Low Pass Filter

$H(\omega)$ DTFT



$\downarrow$

$H_{eff}(\omega)$ CTFT



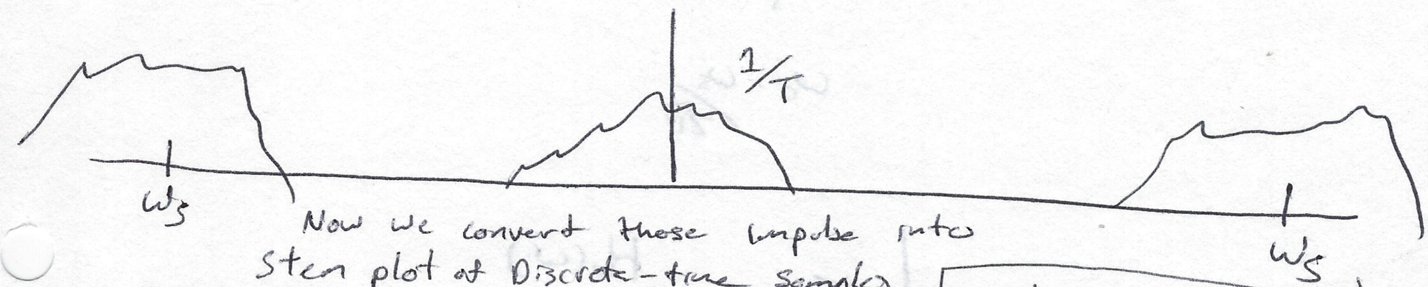
(in continuous world)

& if we want a cutoff of Γ, then Design a Low Pass filter in the discrete world that has a cutoff of $\underline{\Gamma \cdot T}$

$X_c(\omega)$

Sampled signal's spectrum

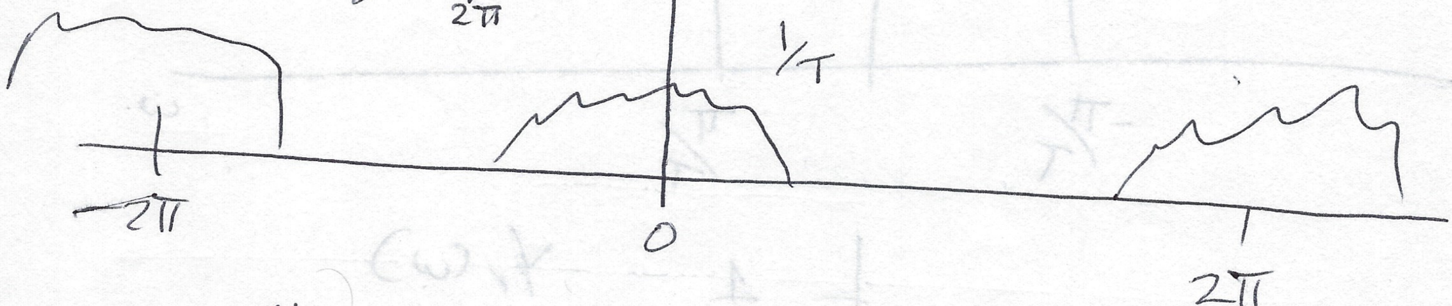
CTFT of $X_s(\omega)$



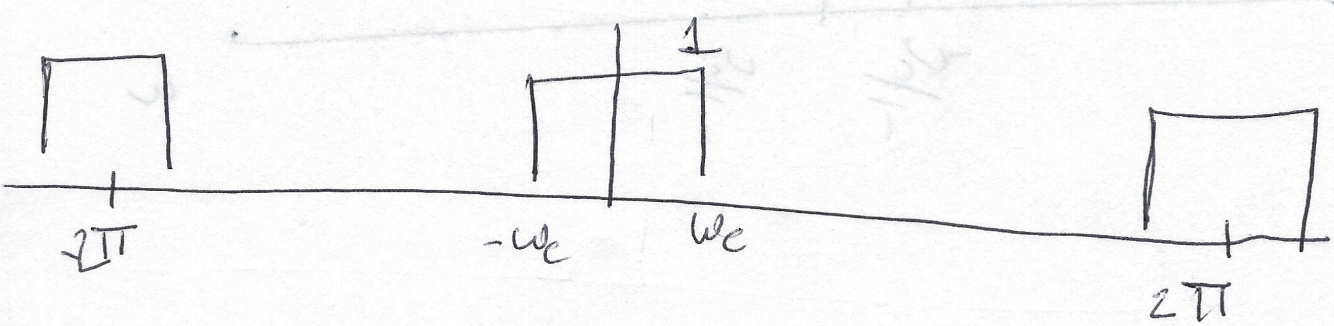
Now we convert these impulse into Stem plot of Discrete-time samples

We scale the ω-axis so that the sampling frequency is 2π

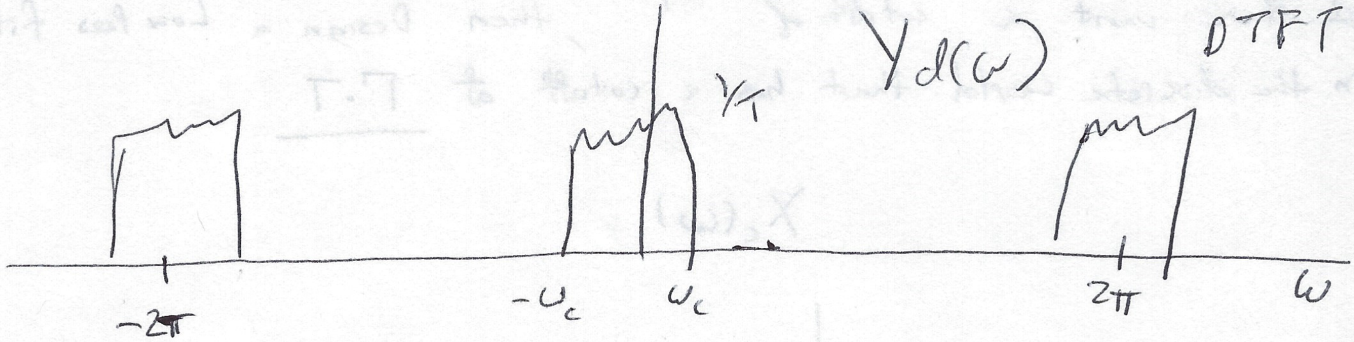
$\boxed{X_d(\omega) \quad \text{DTFT}}$

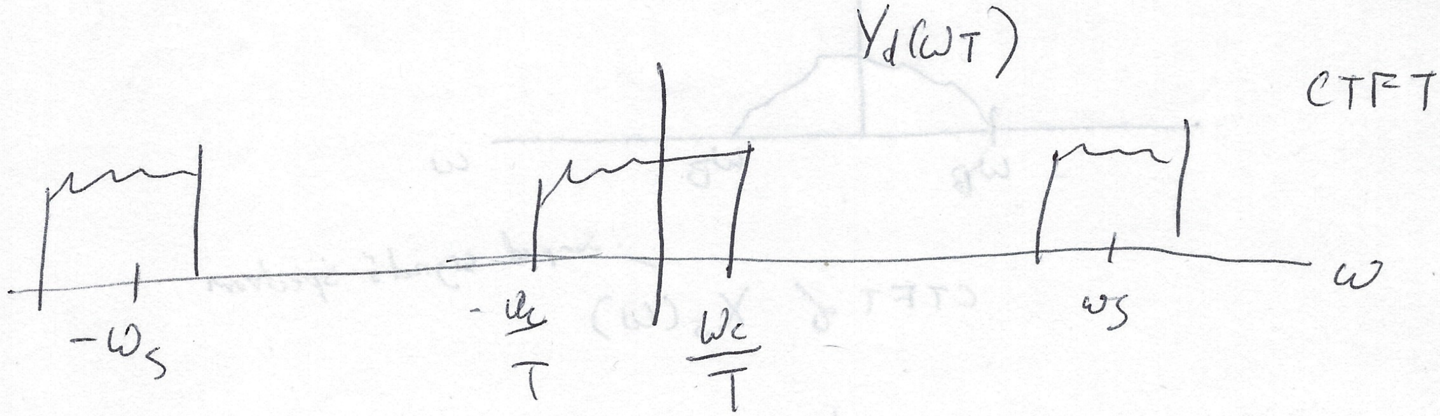


Now design a <u>digital</u> filter $H(\omega)$

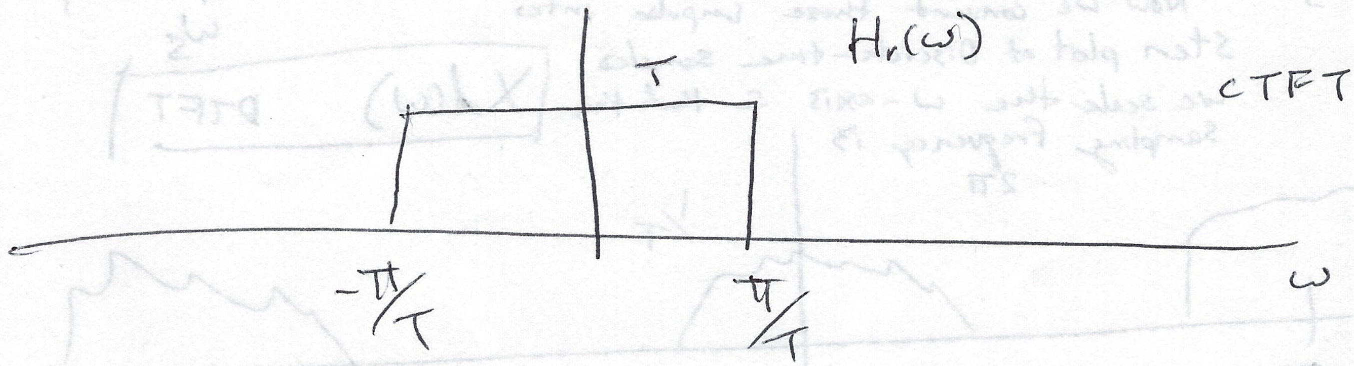


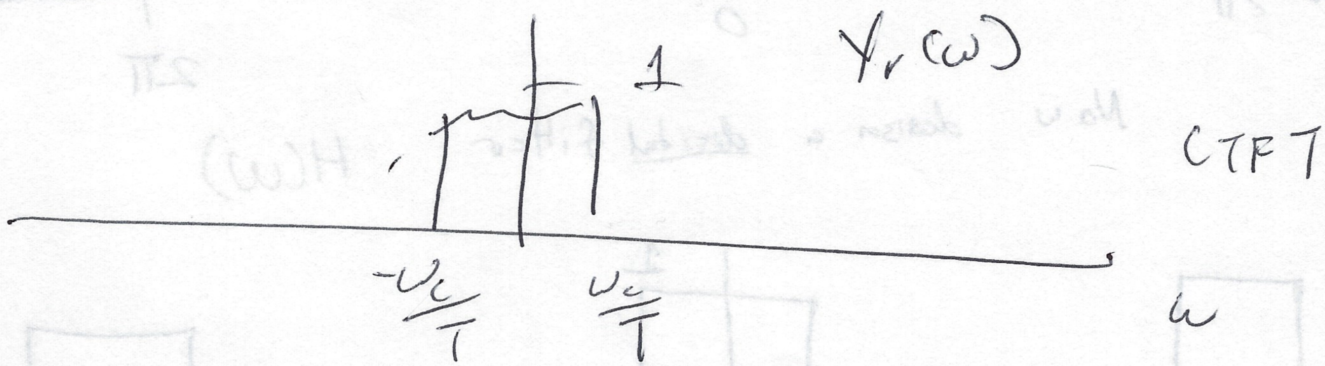
B





$\omega_c \cdot \omega_s / 2\pi$





Discrete time cutoff ω_c

Continuous-time cutoff $\frac{\omega_c}{T}$

Can accomplish many continuous-time filters simply by designing one really good digital filter with a cutoff then running it through sampling systems with different sampling rates. If I want to vary the continuous-time cutoff, all I need to do is change the sampling rate of the input and run it through this digital filter we designed.

$\Rightarrow$ Design a Great fixed Digital filter

Modify the effective continuous-time filter By changing the sampling rate

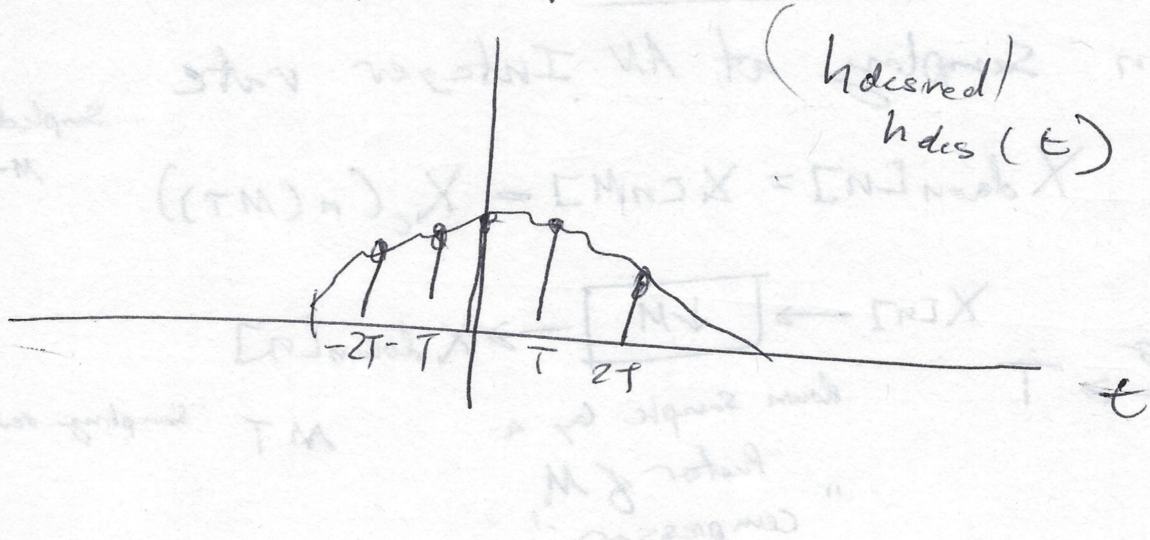
1) Discrete-time system must be LTI

2) Sampler must be above Nyquist rate of Input

How are the impulse responses related

(hdesired)

hdes (t)



$h[n] = T\, h_{des}(nT)$

"Impulse Invariance"

Next Exam question! I want to design a filter that in the continuous time domain has a cutoff at 10 Hz, → I sample that signal at 40 Hz → what should the cutoff in the digital domain be?

"convert back and forth in between cutoffs and sampling rates 2π's and ω_s's"

---

changing the sampling rate

Common problem!

We have $X[n] = X_c(nT)$

We want $X'[n] = X_c(nT')$ ← sampling at some different rate

$T' \neq T$

Long way assuming fast enough sampling

reconstruct continuous-time signal then re sample

don't really want to $X_c(t)$ do this.

Is there an equivalent Discrete-time approach

---

down sampling at AN Integer rate

$M > 0$
$M \in \mathbb{Z}$

$X_{down}[n] = X[nM] = X_c(n(MT))$

Sampled X_c every MT units

$X[n] \rightarrow \boxed{\downarrow M} \rightarrow X_{down}[n]$

Sampling rate → T

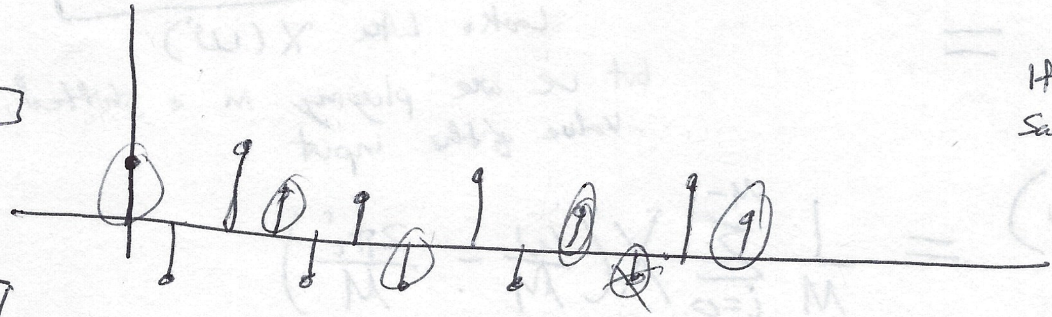
down sample by a factor of M,
"compressor"

MT Sampling rate MT

Take original signal and take every M th sample

Original Signal And Say $M=3$

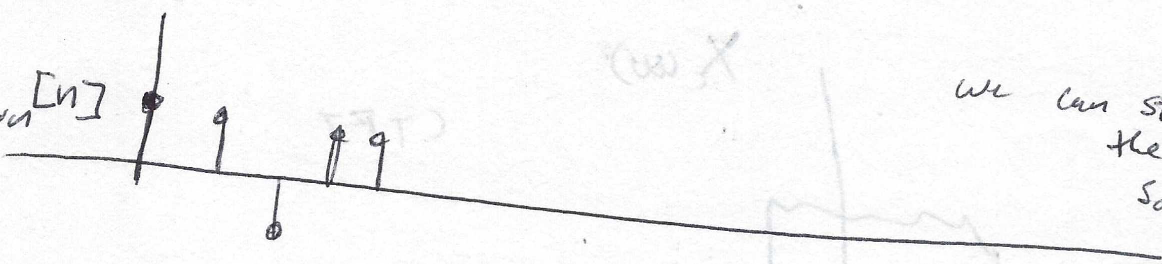
$X[n]$



If this signal was sampled at $M(2\omega_B)$

$\boxed{\downarrow 3}$

$X_{down}[n]$



we can still reconstruct the signal from these samples

possible to remove critical information to reconstruct the Signal

Frequency domain relationship

DTFT

FT of $x_c(t)$

$$X(\omega) = \frac{1}{T} \sum_{k=-\infty}^{\infty} X_c\left(\frac{\omega}{T} - \frac{2\pi k}{T}\right)$$

$$X_{down}(\omega) = \frac{1}{T'} \sum_{k=-\infty}^{\infty} X\left(\frac{\omega}{T'} - \frac{2\pi k}{T'}\right)$$

$$\cancel{= \frac{1}{MT} \sum_{\infty}^{\infty}}$$

$$= \frac{1}{MT} \sum_{k=-\infty}^{\infty} X_c\left(\frac{\omega}{MT} - \frac{2\pi k}{T}\right)$$

$$= \frac{1}{M} \sum_{i=0}^{M-1} \frac{1}{T} \sum_{k=-\infty}^{\infty} X_c\left(\frac{\omega}{MT} - \frac{2\pi(i+kM)}{MT}\right)$$

⑩

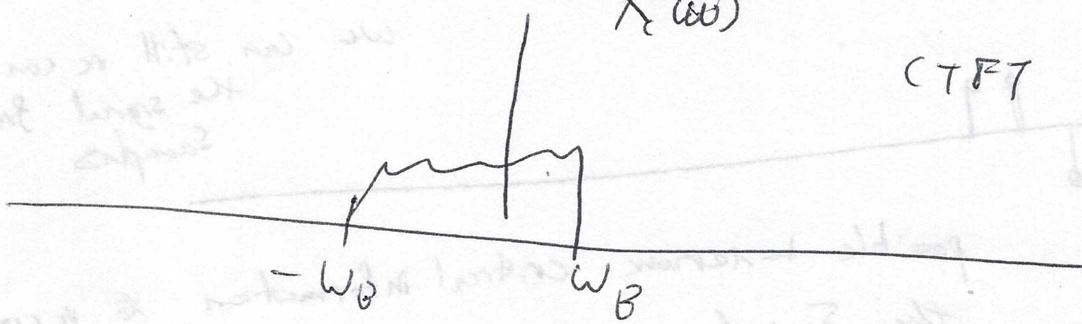
$$= \frac{1}{M}\sum_{i=0}^{M-1}\underbrace{\left[\frac{1}{T}\sum_{k=-\infty}^{\infty} X_c\left(\frac{\omega-2\pi i}{MT}-\frac{2\pi k}{T}\right)\right]}_{\text{Looks like } X(\omega')}$$

$$=$$

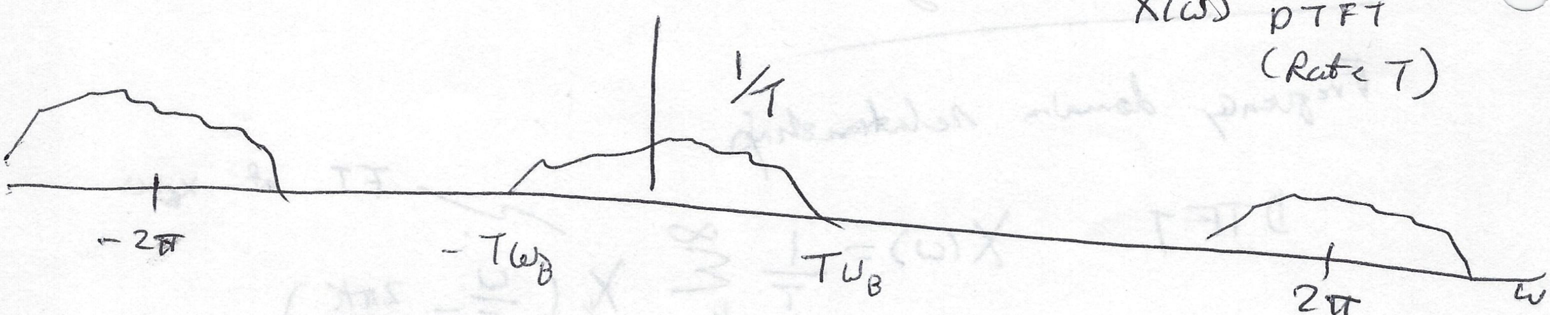
but we are plugging in a shifted value of the input

$$X_{down}(\omega) = \frac{1}{M}\sum_{i=0}^{M-1} X\left(\frac{\omega}{M}-\frac{2\pi i}{M}\right)$$

"Scaled and shifted copies"

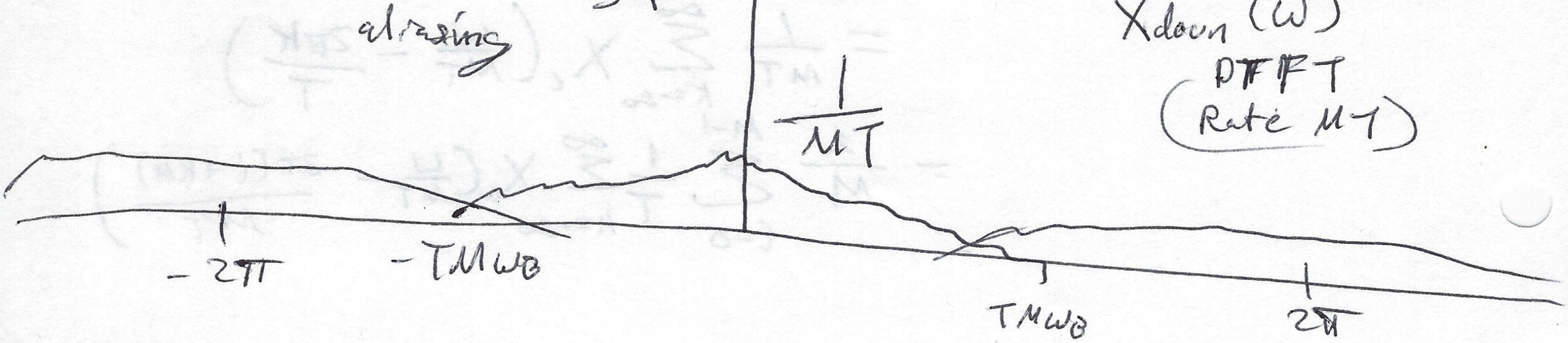


(DTFT of $x_d[n]$ right? yes!)



fundamentally we are making copies and stretching the frequency axis out by a factor of M

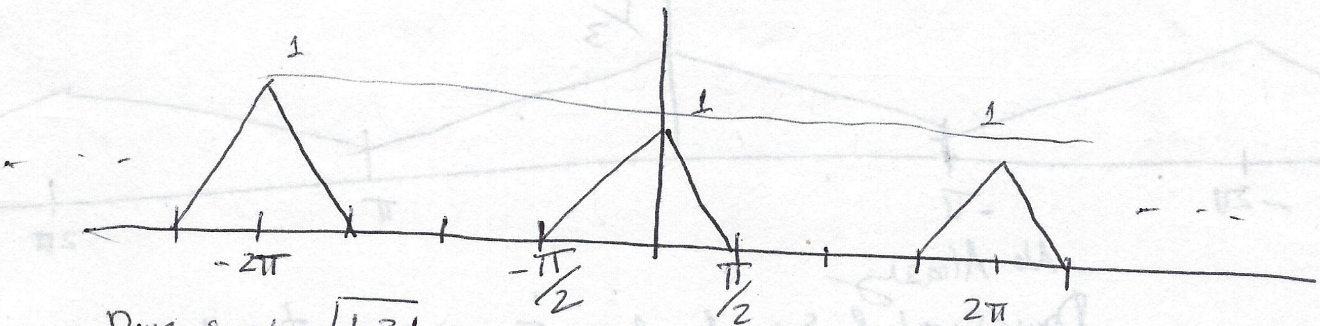
Highly possible to get aliasing



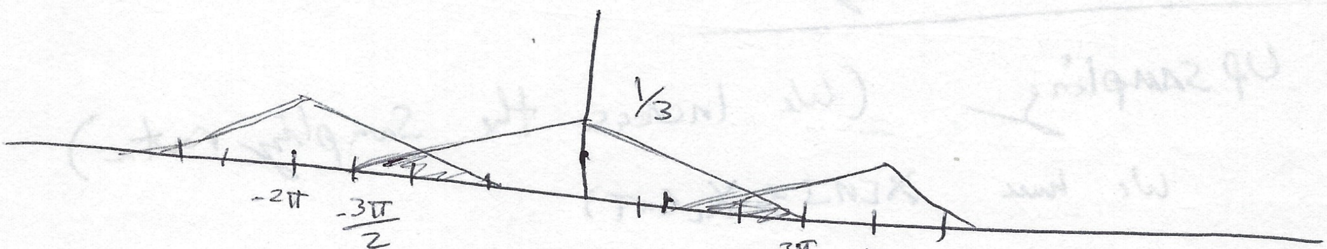
Sample: take discrete-time Fourier transform

MT: Like Sampling CtoS

Aliasing can occur! We would have needed to Sample the original signal at at least M * Nyquist rate

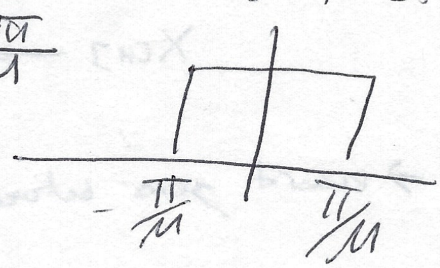


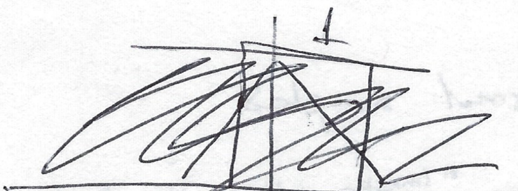
Down sample $\boxed{\downarrow 3}$ shrink and spread out by a factor of 3

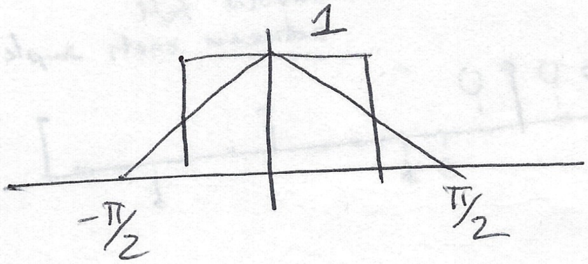


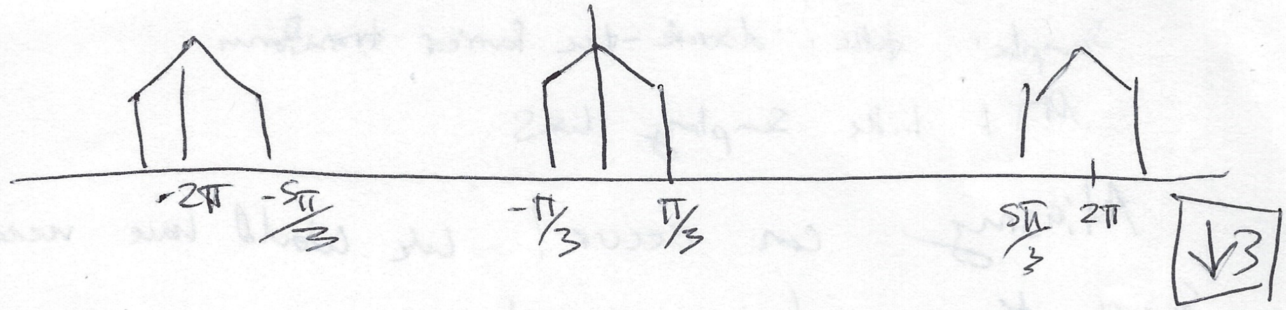
"Solution" Mitigating the problem

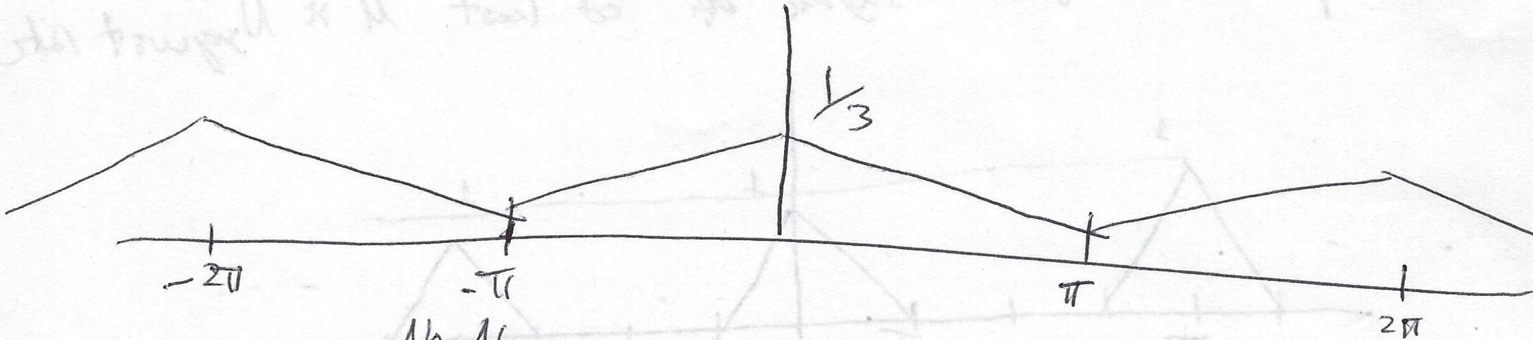
Pre filter the original discrete-time signal with a Lowpass filter with bandwidth $\frac{\pi}{M}$











No Aliasing

Downsampled Signal doesn't represent the Original

Up sampling (We Increase the Sampling rate)

We have $x[n] = x_c(nT)$

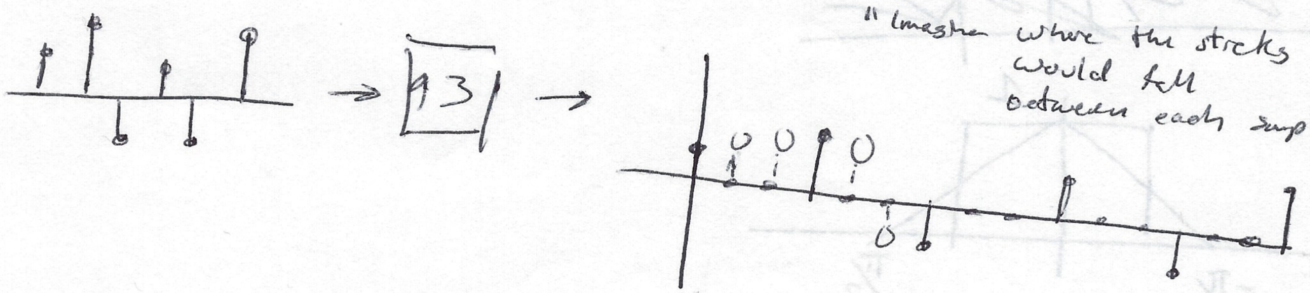
we want $x_{up}[n] = x_c(n\frac{T}{L})$ $L \in \mathbb{Z}$

"up sampling"

$$x[n] \longrightarrow \boxed{\uparrow L} \longrightarrow x_e[n]$$

Expander

→ insert zero between the original samples



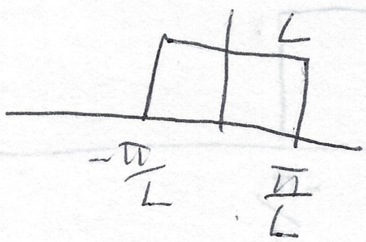
"Imagine where the sticks would fall between each sample"

$$X_e[n] = \begin{cases} x[n/L] & n = 0, \pm L, \pm 2L \\ 0 & \text{Else} \end{cases}$$

How to create upsampled version of x from the expanded version of x

$x_e[n]$ $\quad$ $x_{up}[n]$

we just pass it through a LPF



$H_i(\omega)$ Interpolation Filter, "fancy for connecting the dots"

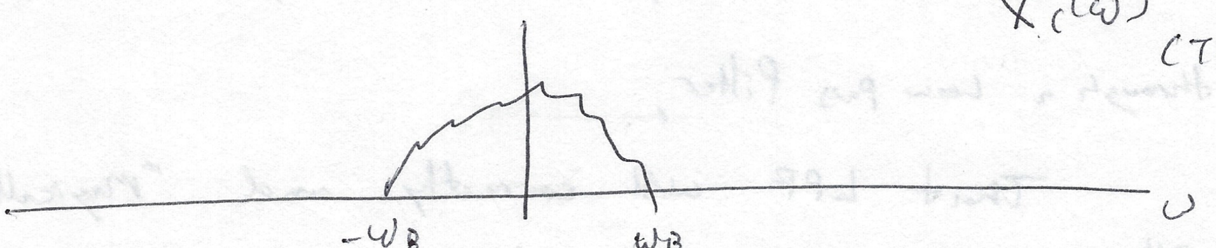
$$X_e(\omega) = \sum_{n=-\infty}^{\infty} x_e[n] e^{-j\omega n}$$

nL is the only value that is non zero

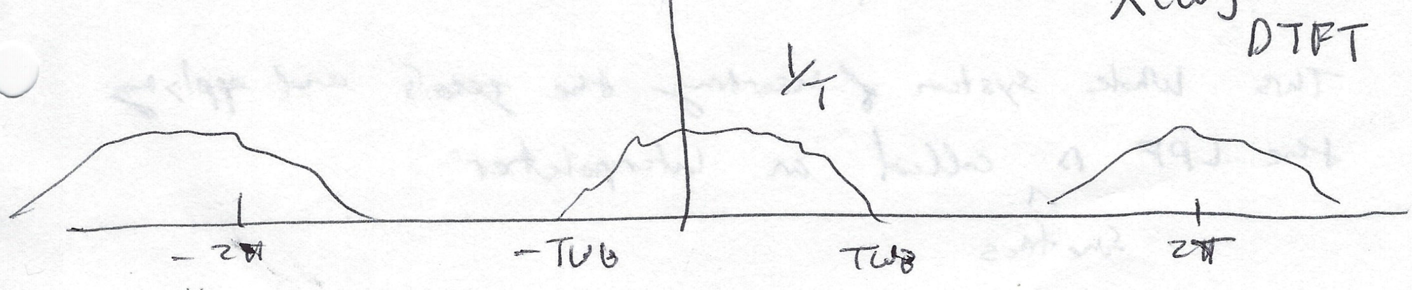
$$= \sum_{n=-\infty}^{\infty} x[nL] e^{-j\omega nL}$$

$$= \underline{X(\omega L)}$$

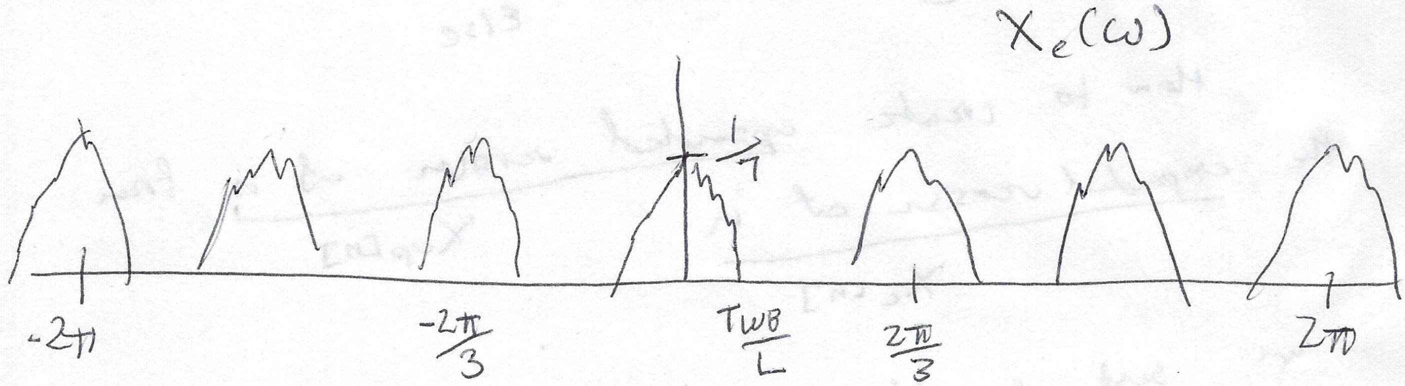
$X_c(\omega)$ CTFT

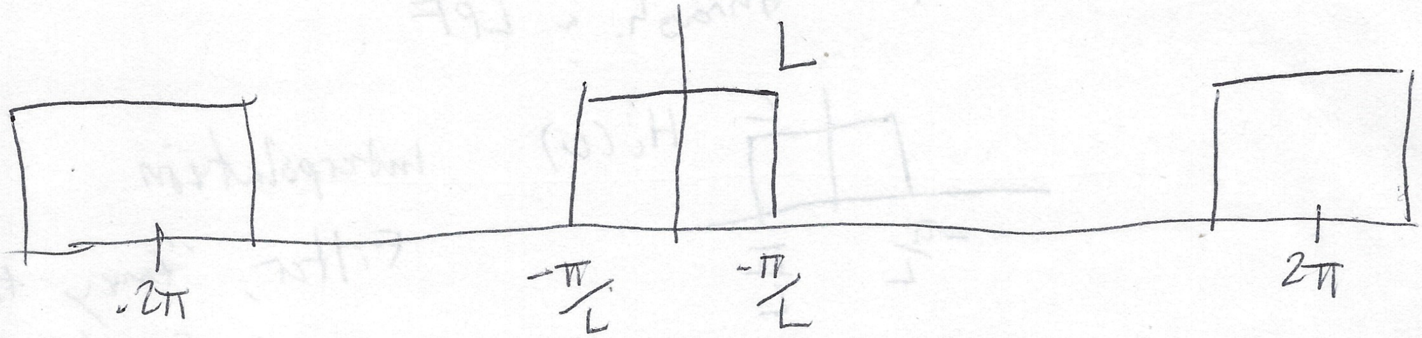


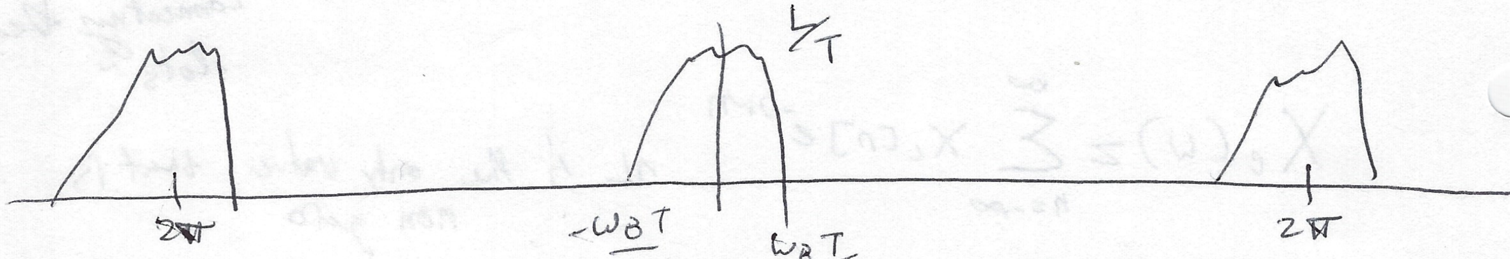
$X(\omega)$ DTFT



(14) Take the DTFT $X(\omega)$ and squish the frequency axis by a factor of L







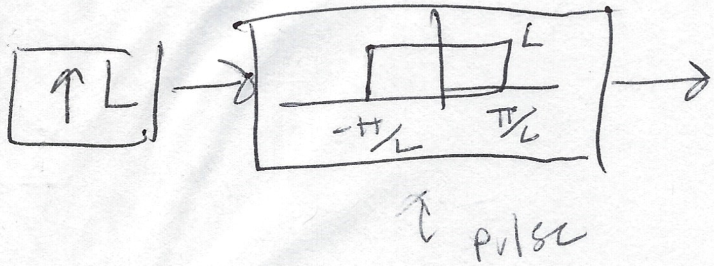
Exactly what we'd get by sampling $\frac{T}{L}$ in the first place!

Take original samples, spread them out, put zeros in there, run through a low pass filter,

That LPF will correctly and "magically" fill in the right samples in between

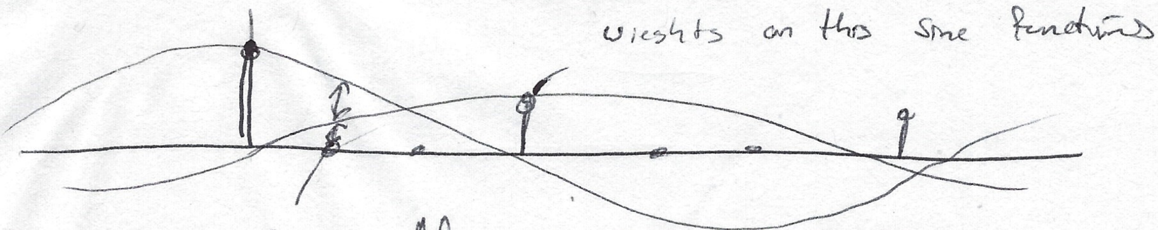
This whole system of inserting the zeros and applying the LPF is sometimes called an interpolater

Interpolation



$h_i[n] = \text{sinc}$

Take the sample like this



SAME Interpolation Idea to reconstruct a signal

I turns out that the effective filter for linear Interpolation looks like

$$H_{lin}(\omega) = \frac{1}{L}\left[\frac{\sin(\omega L/2)}{\sin(\omega/2)}\right]^2$$