ESE 531: Digital Signal Processing

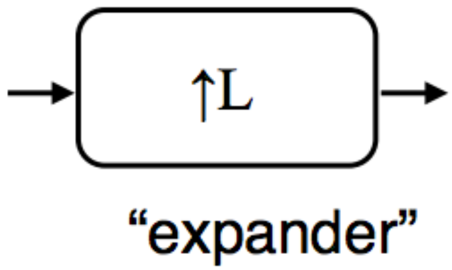
Lec 11: February 20, 2020

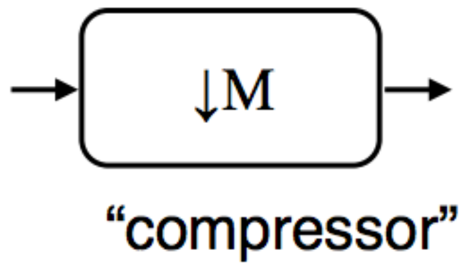
Polyphase Decomposition and Multi-rate Filter Banks

Lecture Outline

- Review: Interchanging Operations
- Polyphase Decomposition
- Multi-Rate Filter Banks

Expander and Compressor





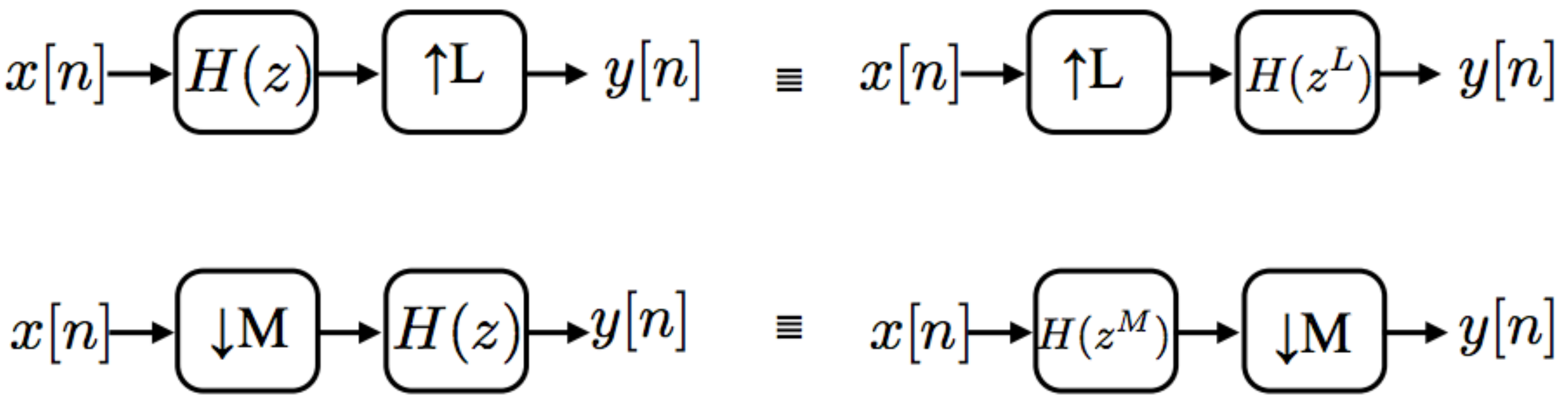
not LTI!

Upsampling
-expanding in time
-compressing in frequency

Downsampling
-compressing in time
-expanding in frequency

Interchanging Operations - Summary

Filter and expander — Expander and expanded filter*



Compressor and filter — Expanded filter* and compressor

*Expanded filter = expanded impulse response, compressed freq response

Polyphase Decomposition

- The polyphase decomposition of a sequence is obtained by representing it as a superposition of M subsequences, each consisting of every Mth value of successively delayed versions of the sequence.
- When this decomposition is applied to a filter impulse response, it can lead to efficient implementation structures for linear filters in several contexts.

Polyphase Decomposition

- We can decompose an impulse response (of our filter) to:

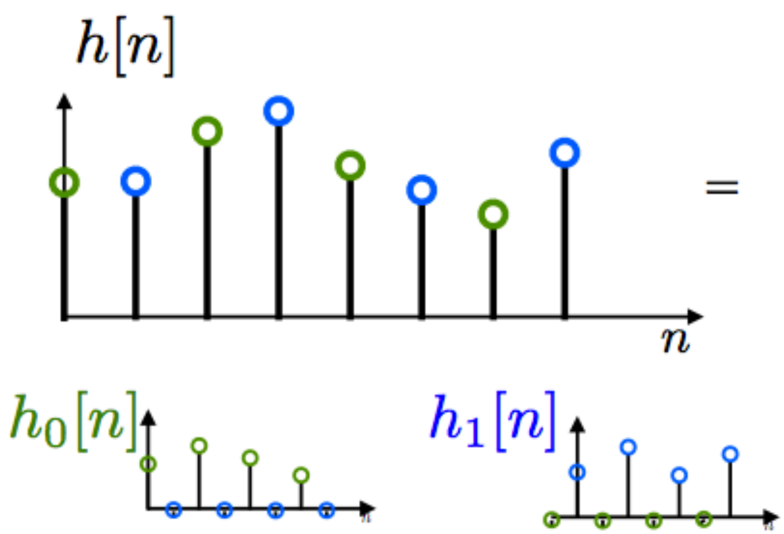
$$h[n] = \sum_{k=0}^{M-1} h_k[n-k]$$

Polyphase Decomposition

- We can decompose an impulse response (of our filter) to:

$$h[n] = \sum_{k=0}^{M-1} h_k[n-k]$$

M=2

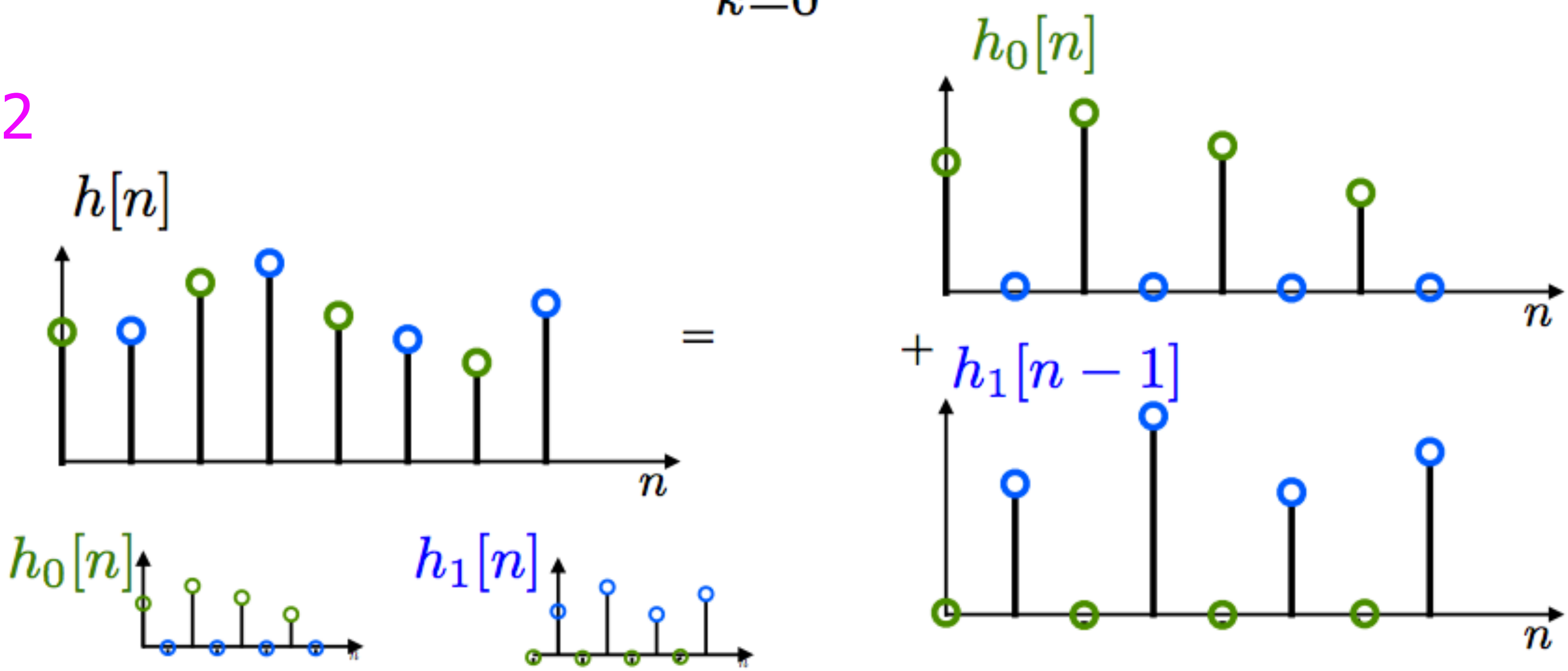


Polyphase Decomposition

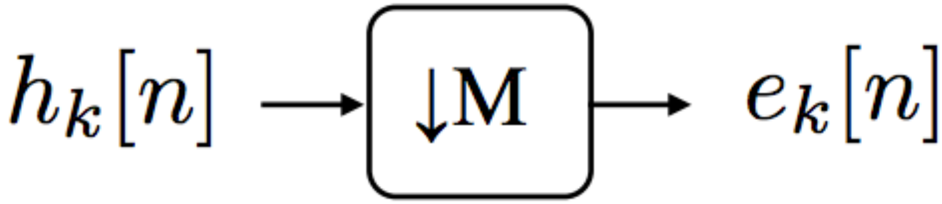
- We can decompose an impulse response (of our filter) to:

$$h[n] = \sum_{k=0}^{M-1} h_k[n-k]$$

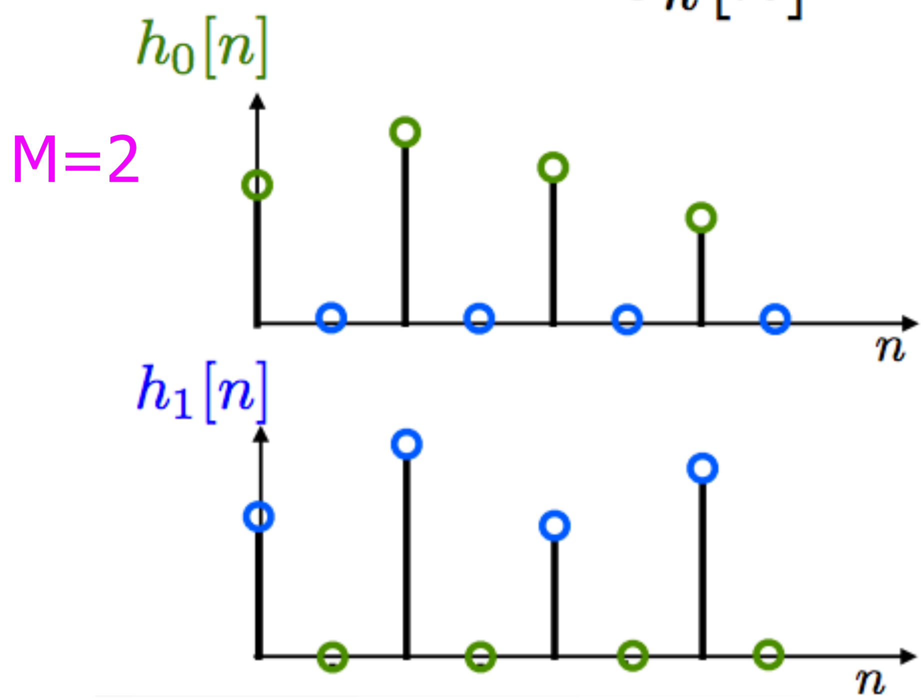
M=2



Polyphase Decomposition



$$e_k[n] = h_k[nM]$$

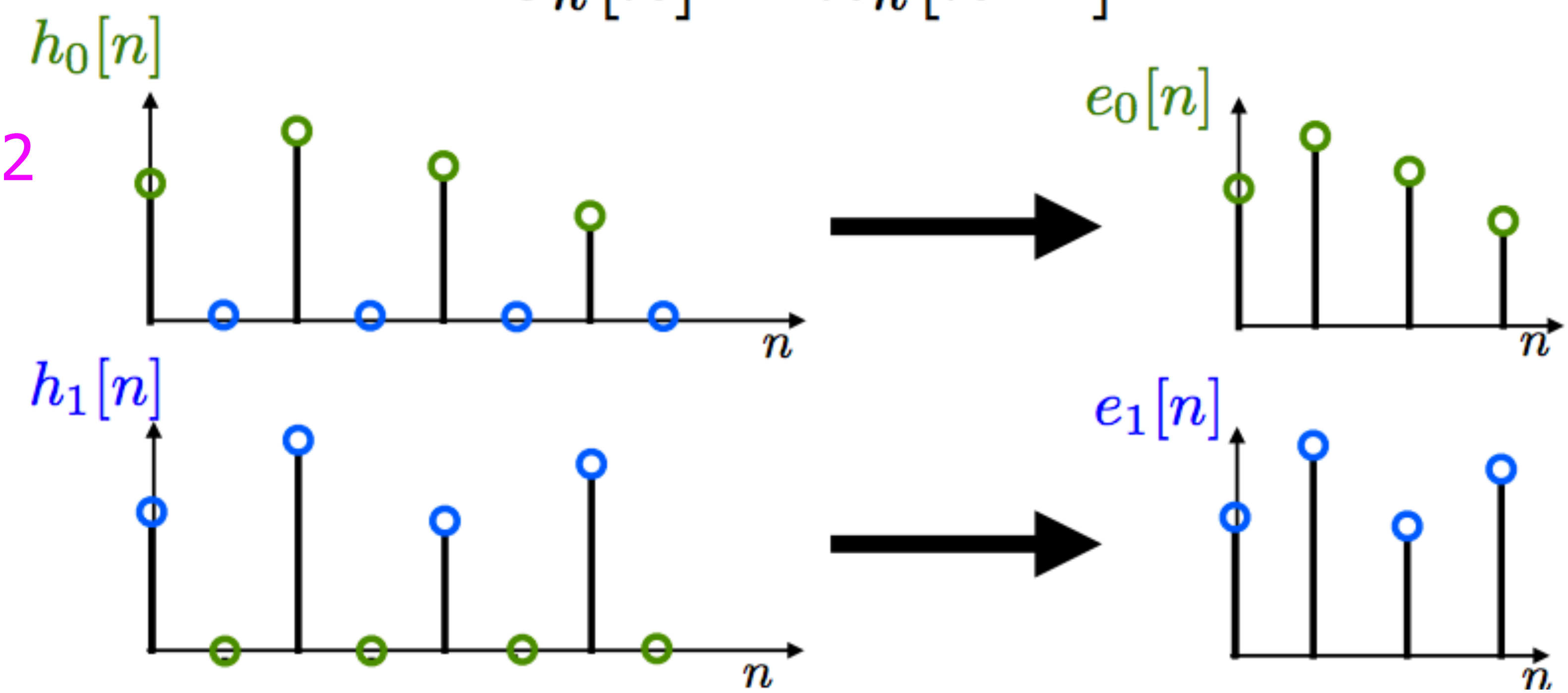


Polyphase Decomposition

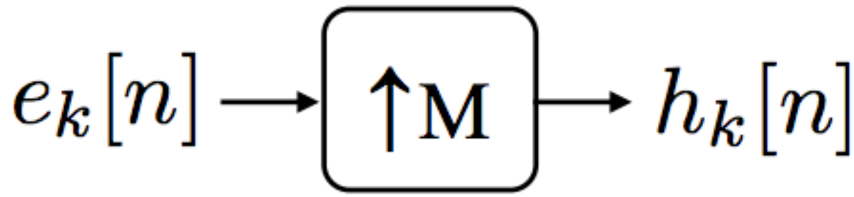
$$h_k[n] \rightarrow \boxed{\downarrow M} \rightarrow e_k[n]$$

$$e_k[n] = h_k[nM]$$

M=2



Polyphase Decomposition



recall upsampling ⇒ scaling

$$H_k(z) = E_k(z^M)$$

Polyphase Decomposition

$$e_k[n] \longrightarrow \boxed{\uparrow M} \longrightarrow h_k[n]$$

recall upsampling ⇒ scaling

$$H_k(z) = E_k(z^M)$$

Also, recall:

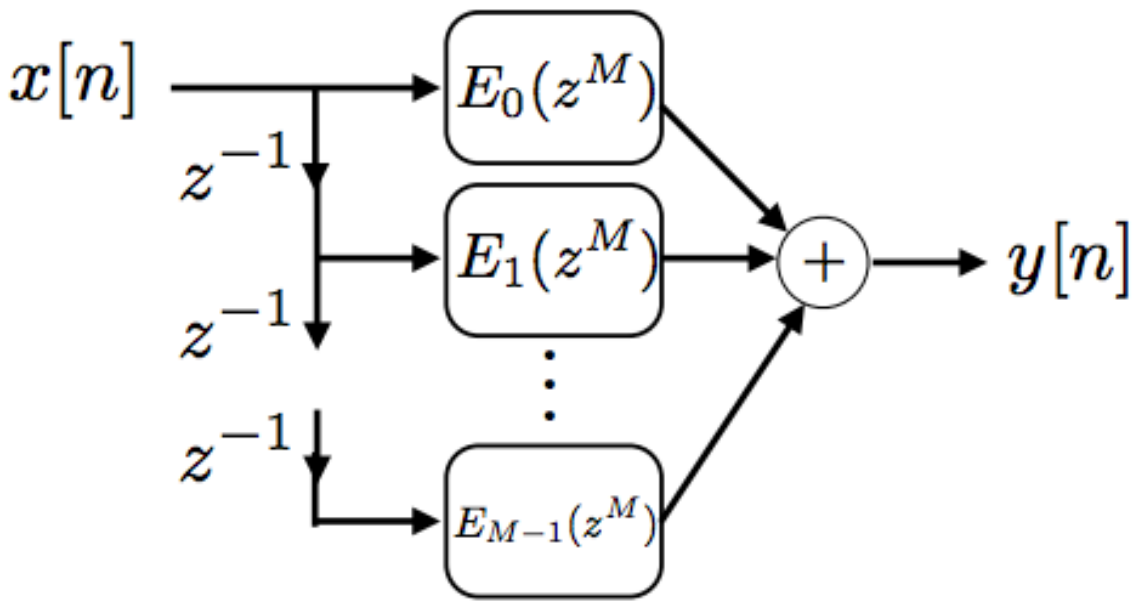
$$h[n] = \sum_{k=0}^{M-1} h_k[n-k]$$

So,

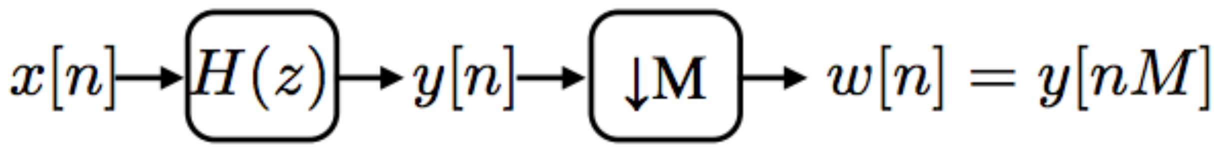
$$\boxed{H(z) = \sum_{k=0}^{M-1} E_k(z^M) z^{-k}}$$

Polyphase Decomposition

$$H(z) = \sum_{k=0}^{M-1} E_k(z^M) z^{-k}$$



Polyphase Implementation of Decimation



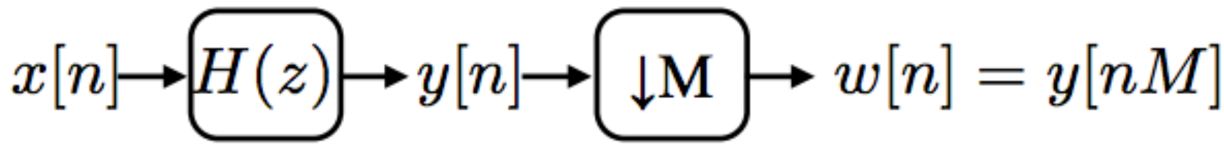
- Problem:
  - Compute all y[n] and then throw away -- wasted computation!

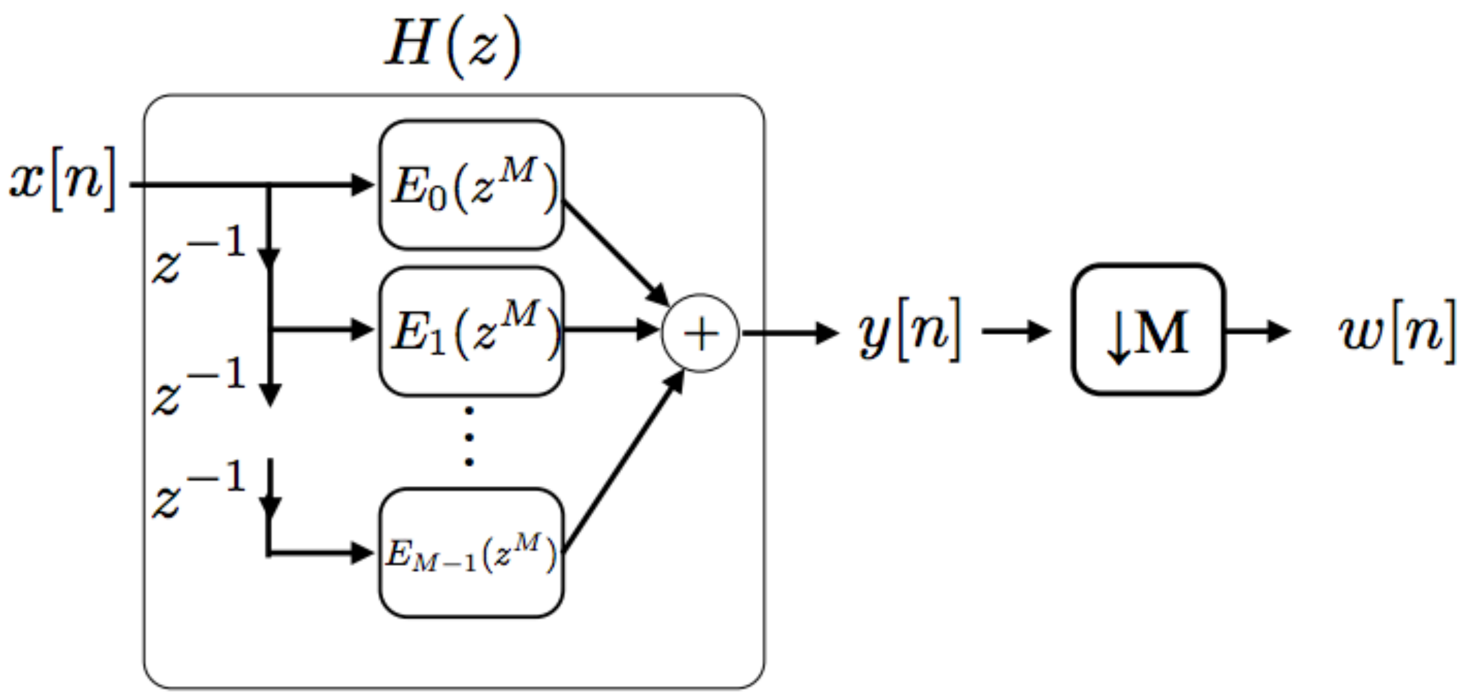
Polyphase Implementation of Decimation

$$x[n] \rightarrow \boxed{H(z)} \rightarrow y[n] \rightarrow \boxed{\downarrow M} \rightarrow w[n] = y[nM]$$

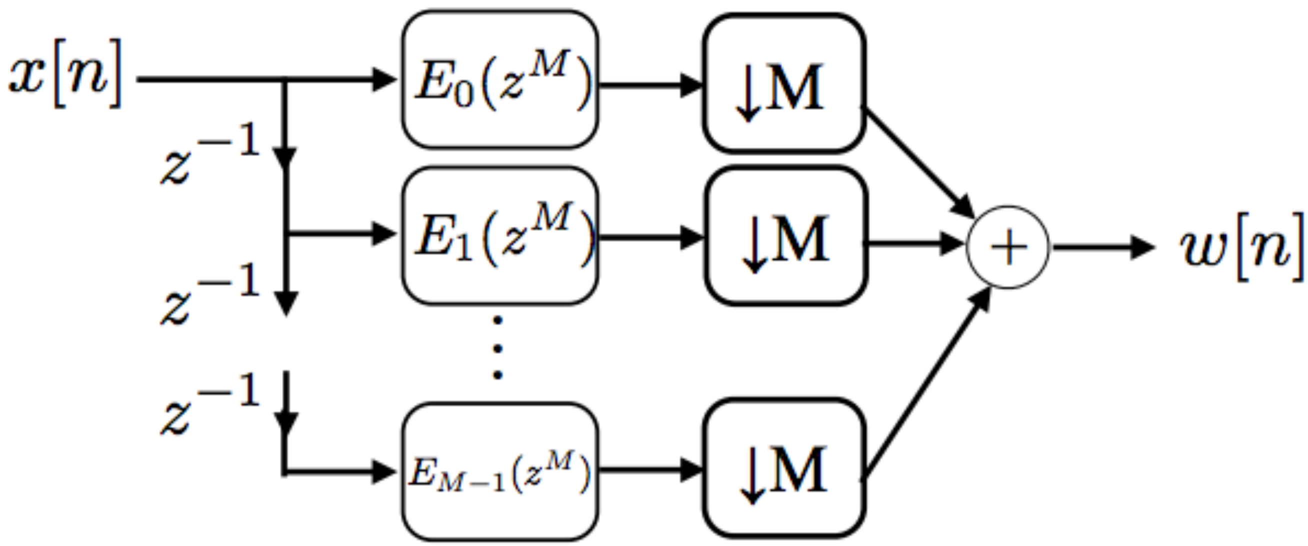
- Problem:
  - Compute all y[n] and then throw away -- wasted computation!
  - For FIR length N → N multiplications/unit time

Polyphase Implementation of Decimation

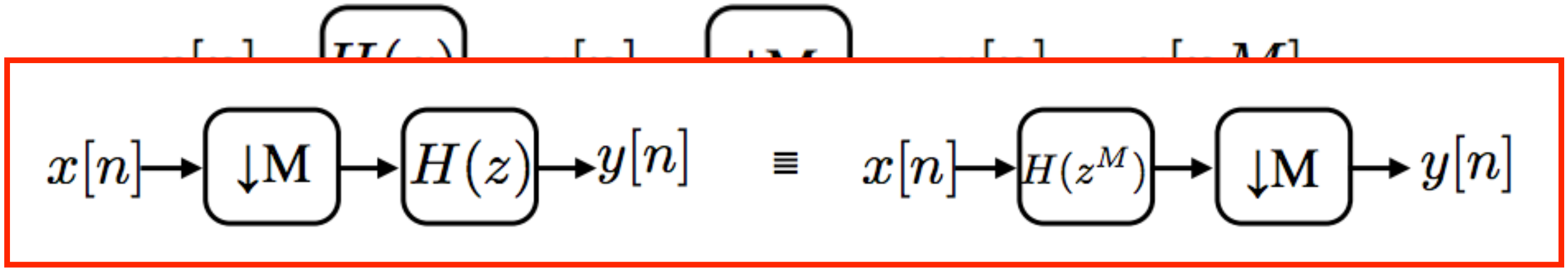


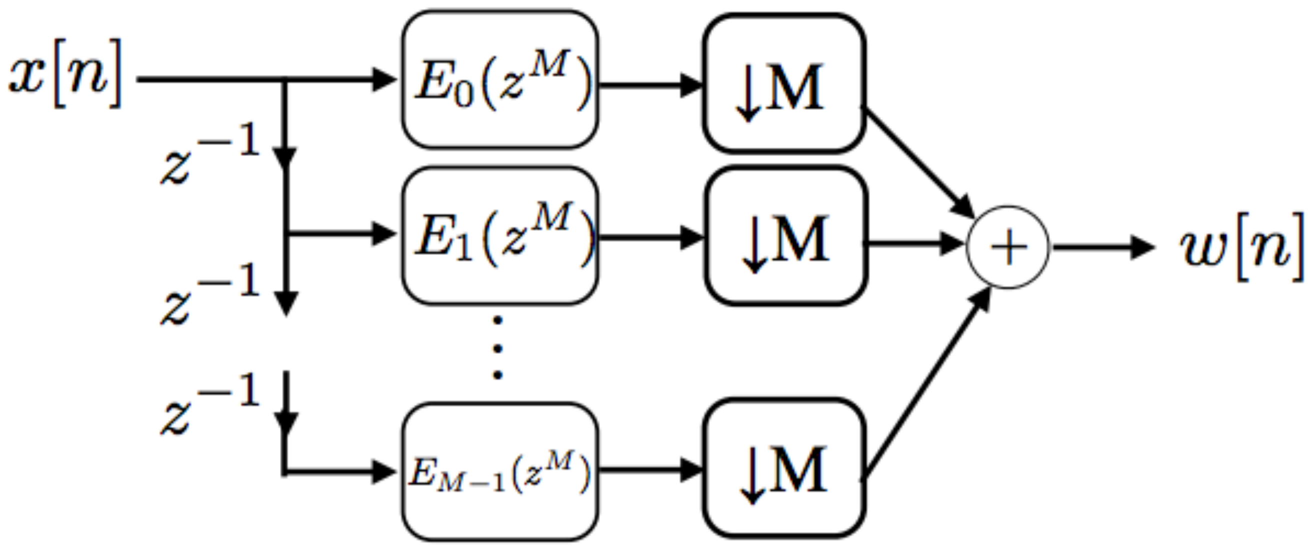


Polyphase Implementation of Decimation

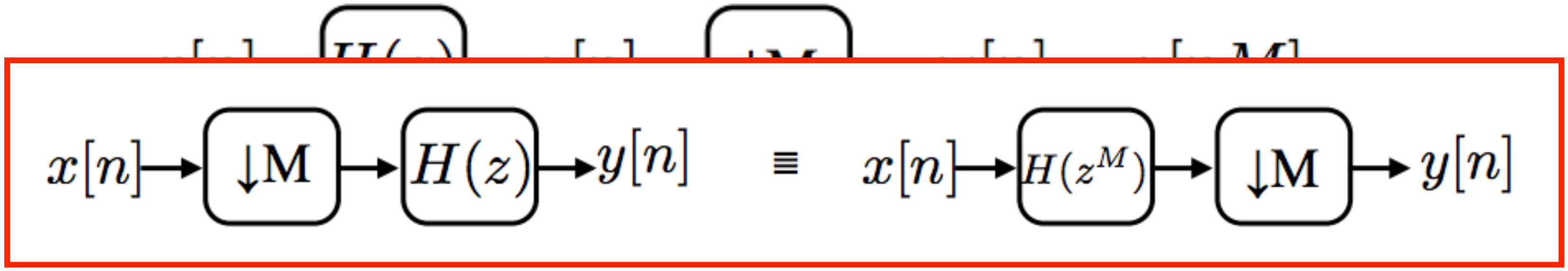


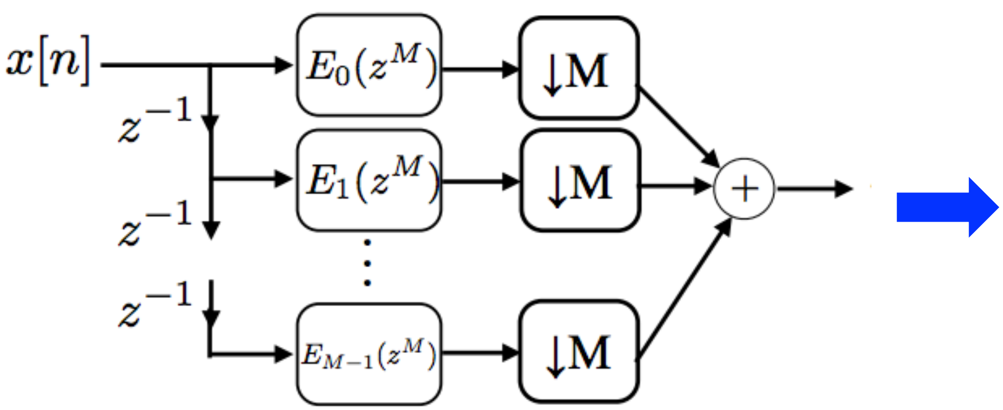
Polyphase Implementation of Decimation



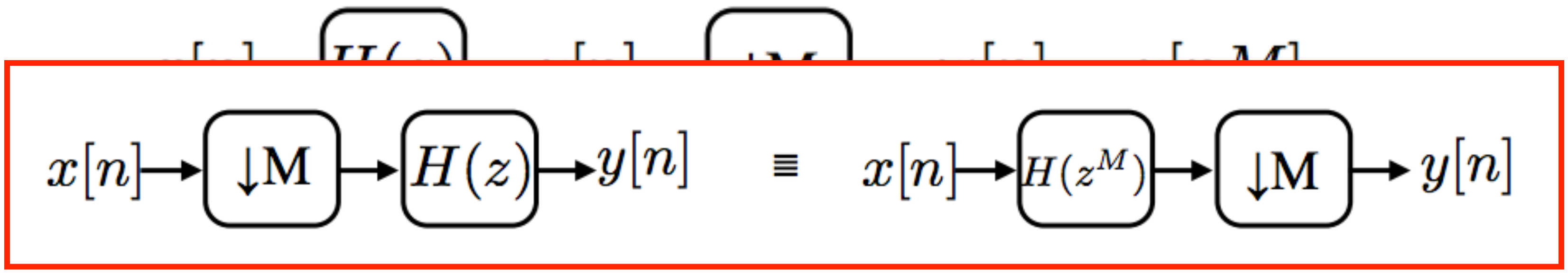


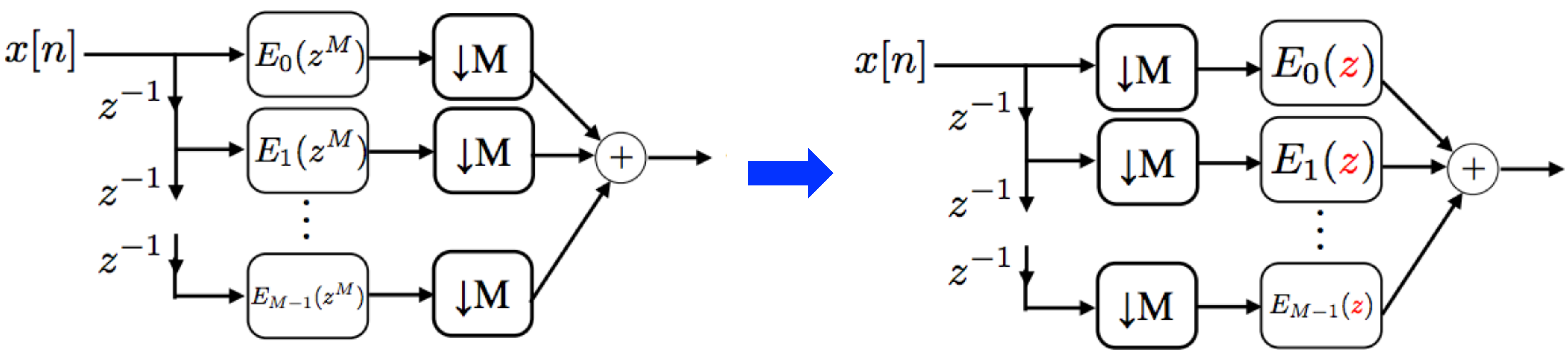
Polyphase Implementation of Decimation



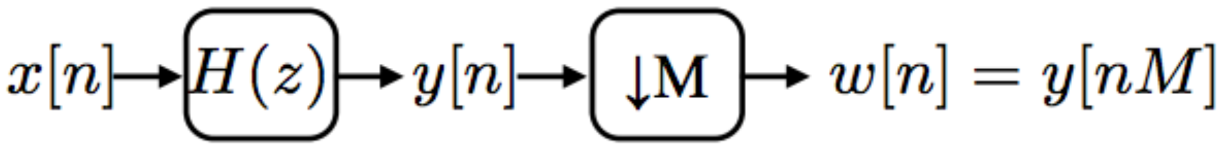


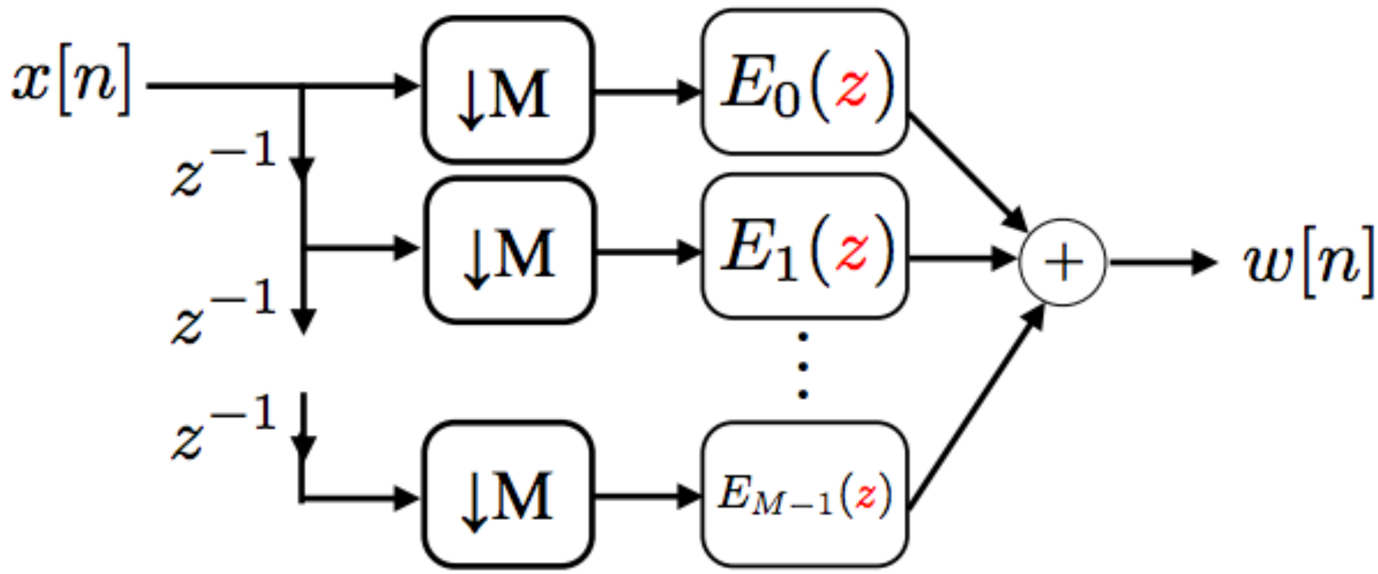
Polyphase Implementation of Decimation



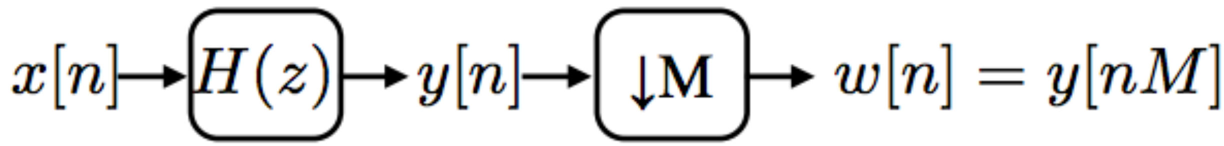


Polyphase Implementation of Decimation

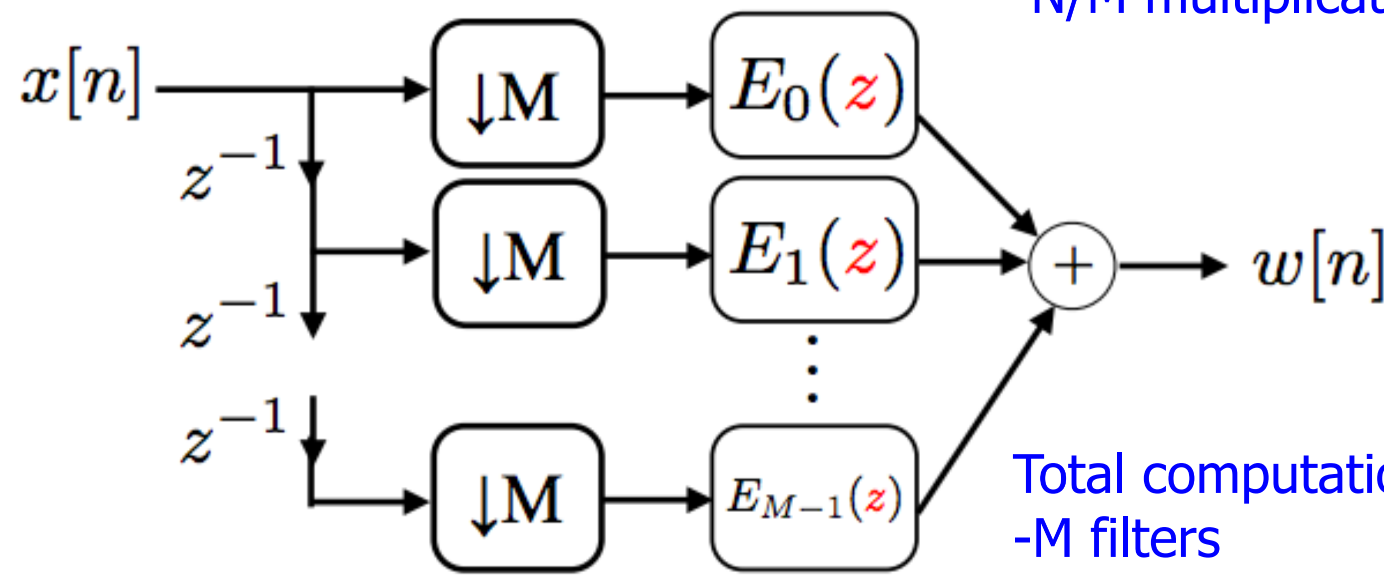




Polyphase Implementation of Decimation

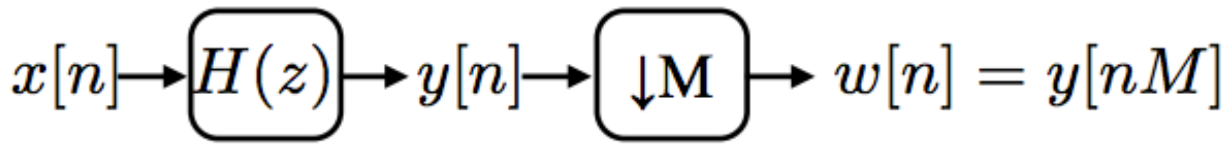


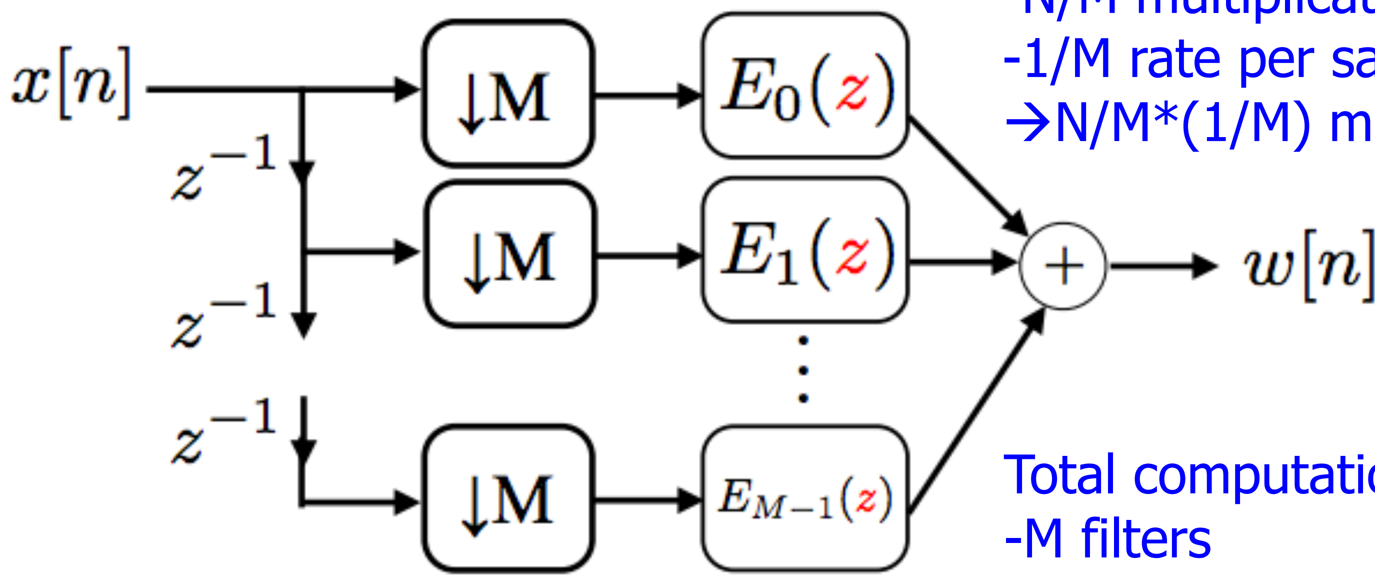
Each filter computation:
-N/M multiplications



Total computation:
-M filters
→N mults??

Polyphase Implementation of Decimation

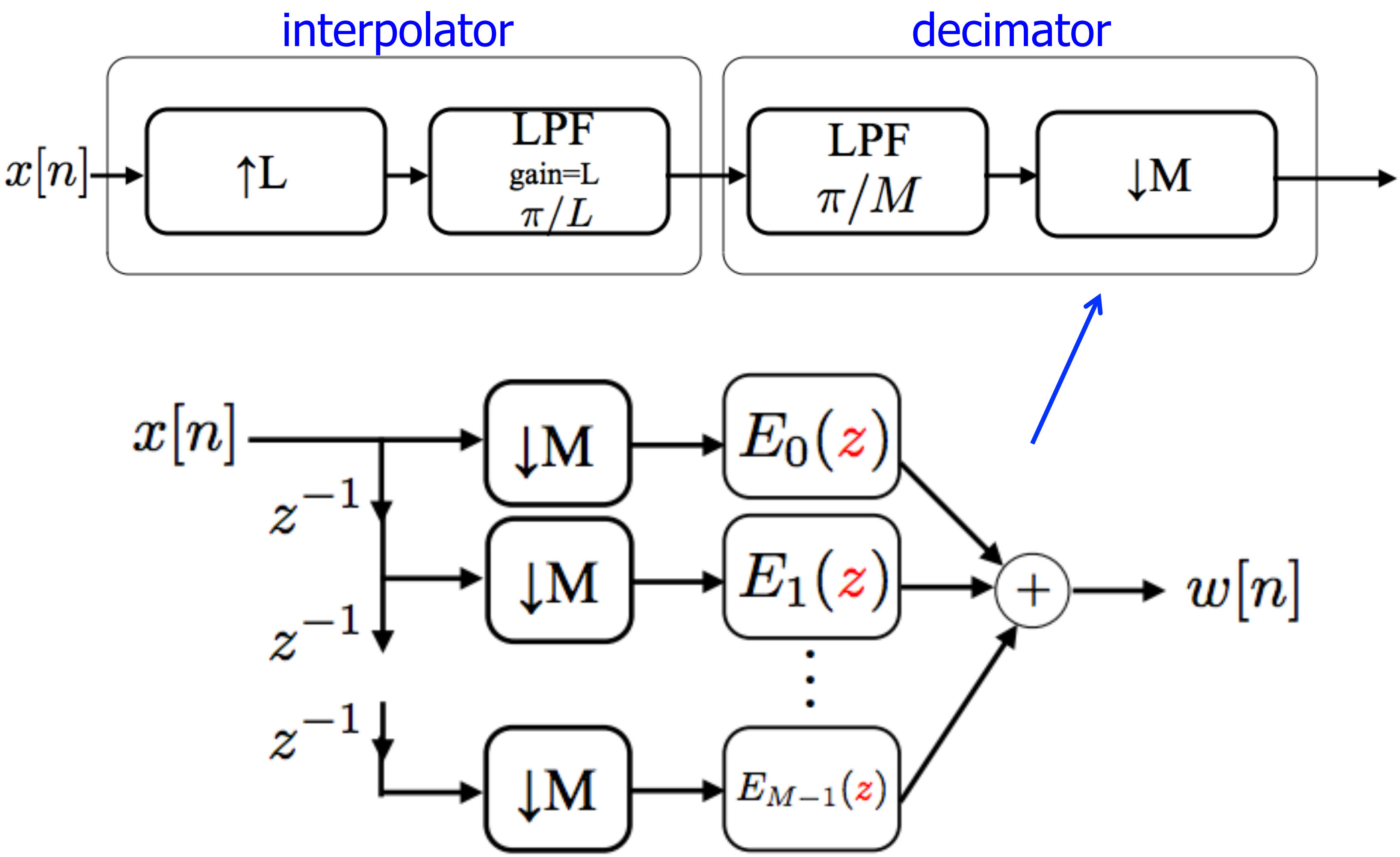




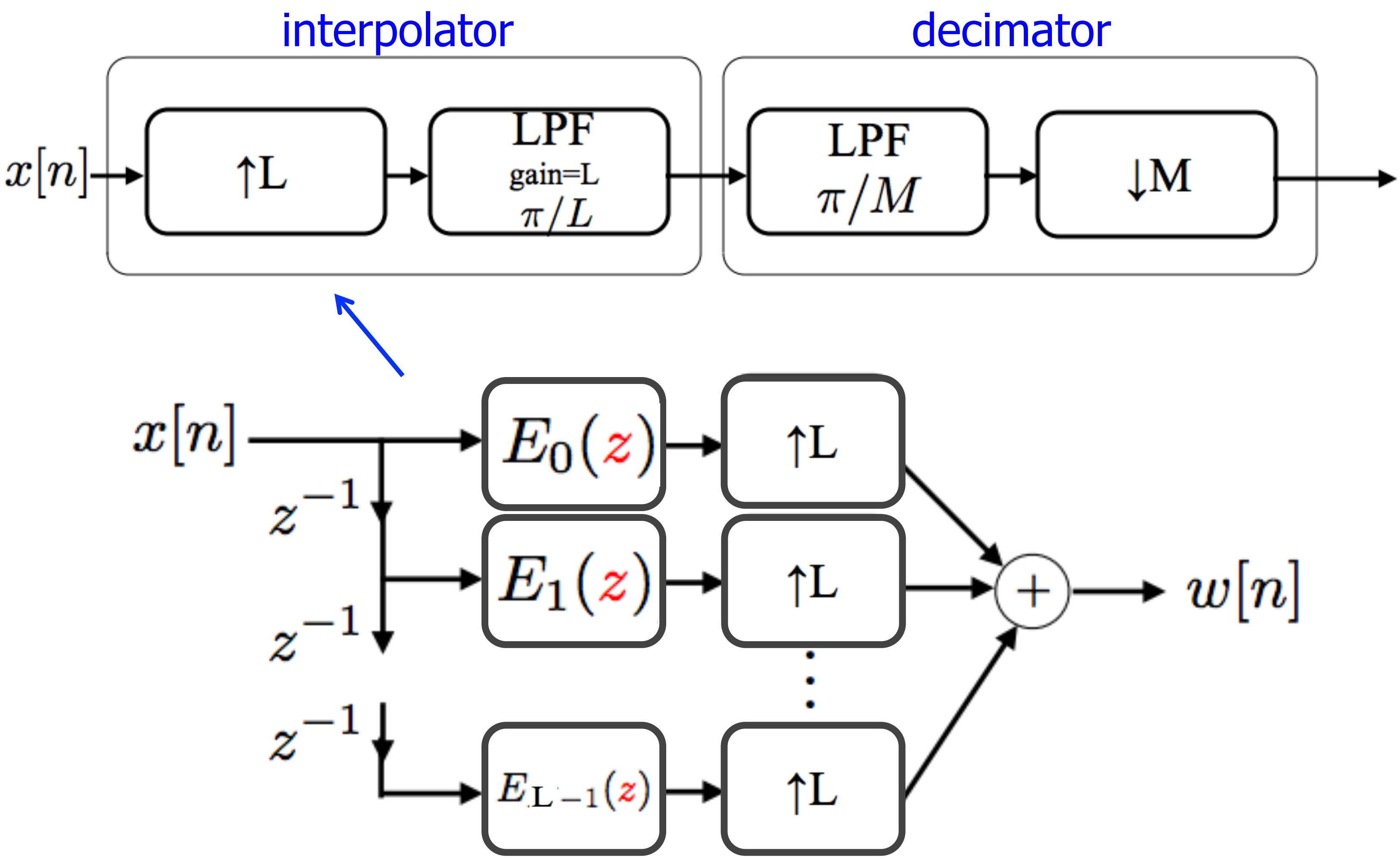
Each filter computation:
-N/M multiplications
-1/M rate per sample
→N/M*(1/M) mults/unit time

Total computation:
-M filters
→N/M mults/unit time

Polyphase Implementation of Decimator



Polyphase Implementation of Interpolation



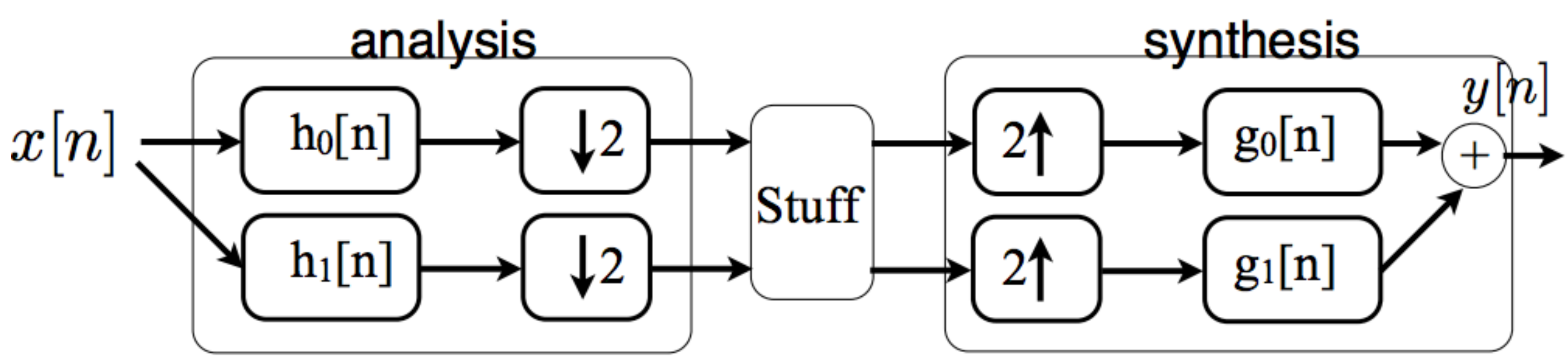
Multi-Rate Filter Banks

- Use filter banks to operate on a signal differently in different frequency bands
  - To save computation, reduce the rate after filtering

Multi-Rate Filter Banks

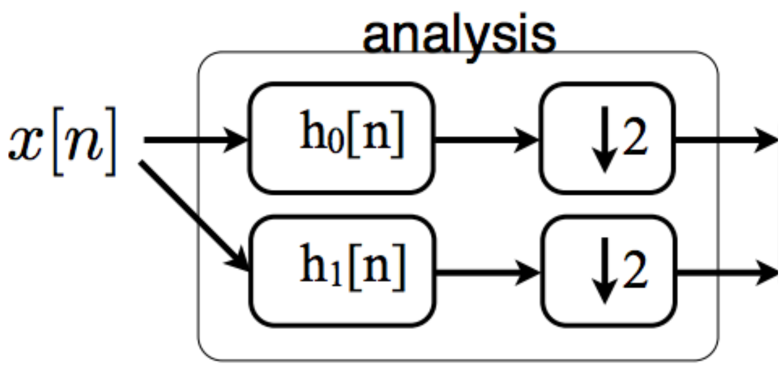
- Use filter banks to operate on a signal differently in different frequency bands
  - To save computation, reduce the rate after filtering

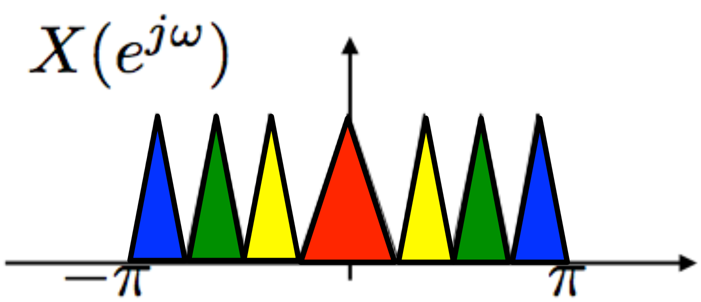
- $h_0[n]$ is low-pass, $h_1[n]$ is high-pass
  - Often $h_1[n]=e^{j\pi n}h_0[n]$ ← shift freq resp by π



Multi-Rate Filter Banks

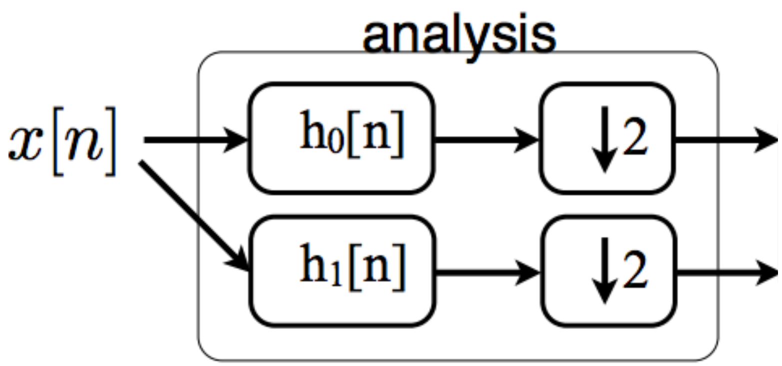
- Assume h_0, h_1 are ideal low/high pass with $\omega_C = \pi/2$

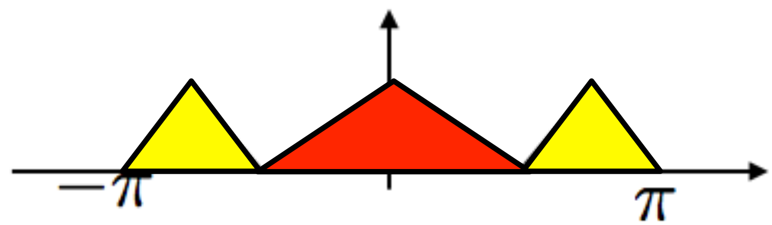


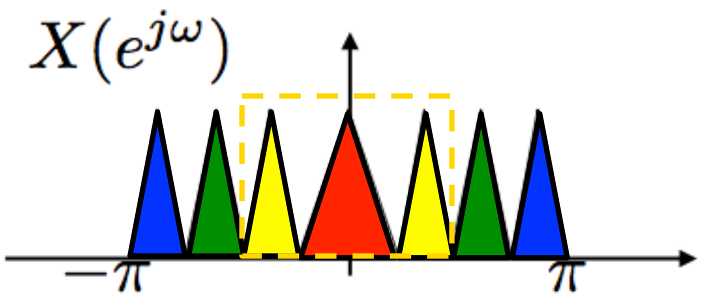


Multi-Rate Filter Banks

- Assume h_0, h_1 are ideal low/high pass with $\omega_C = \pi/2$

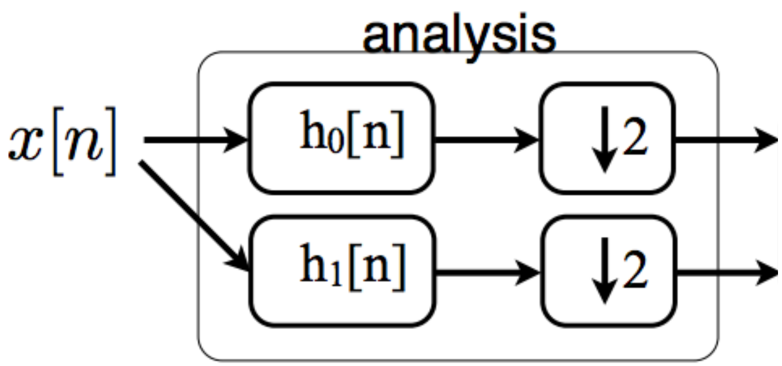


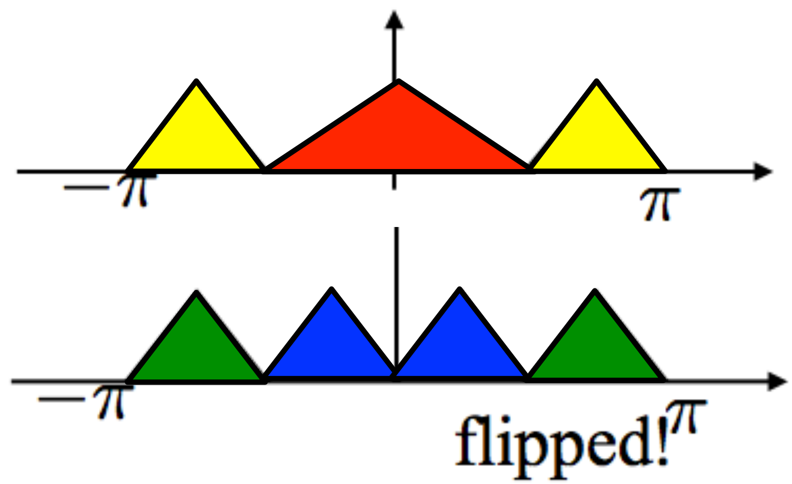


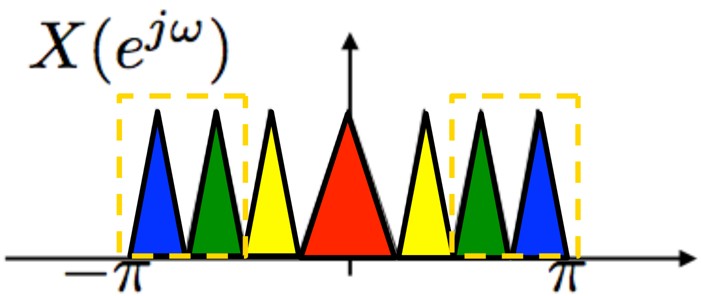


Multi-Rate Filter Banks

- Assume h_0, h_1 are ideal low/high pass with $\omega_C = \pi/2$

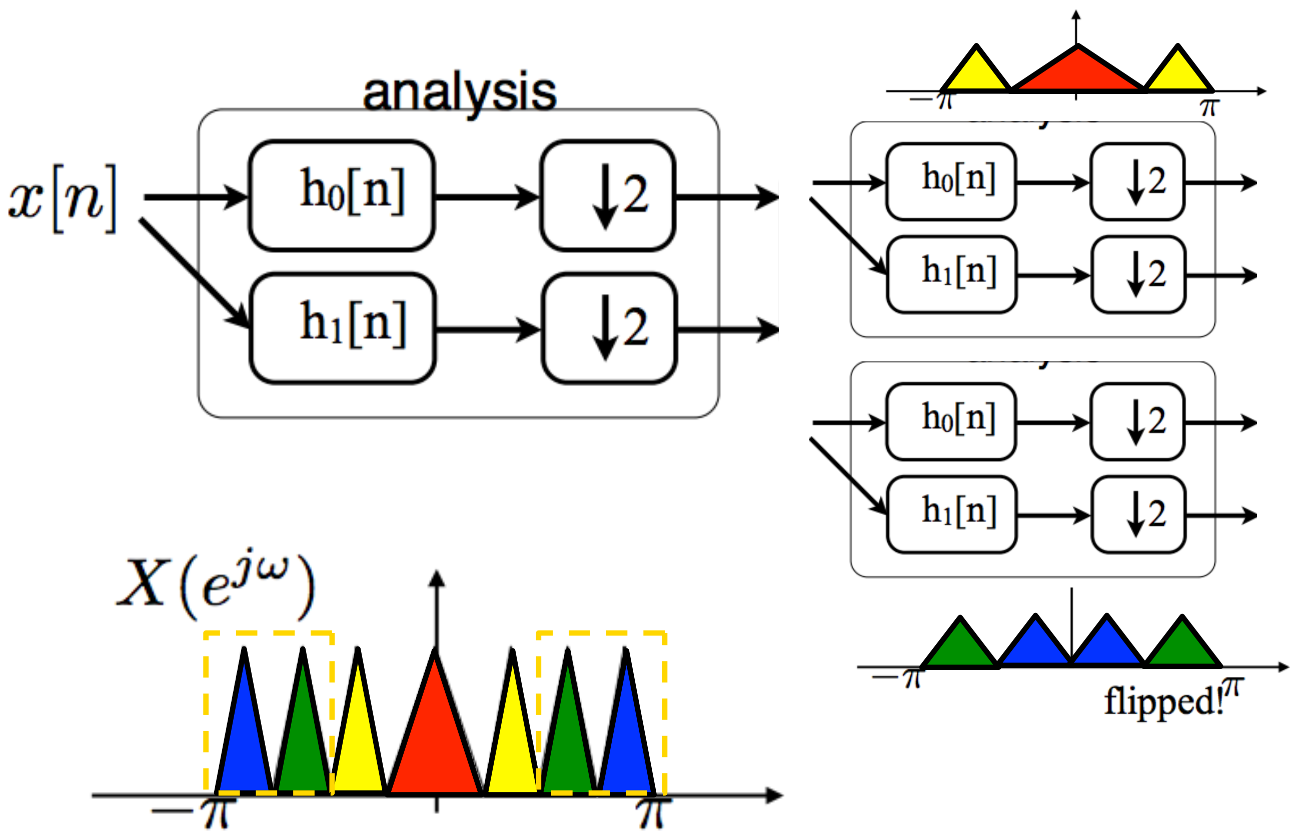






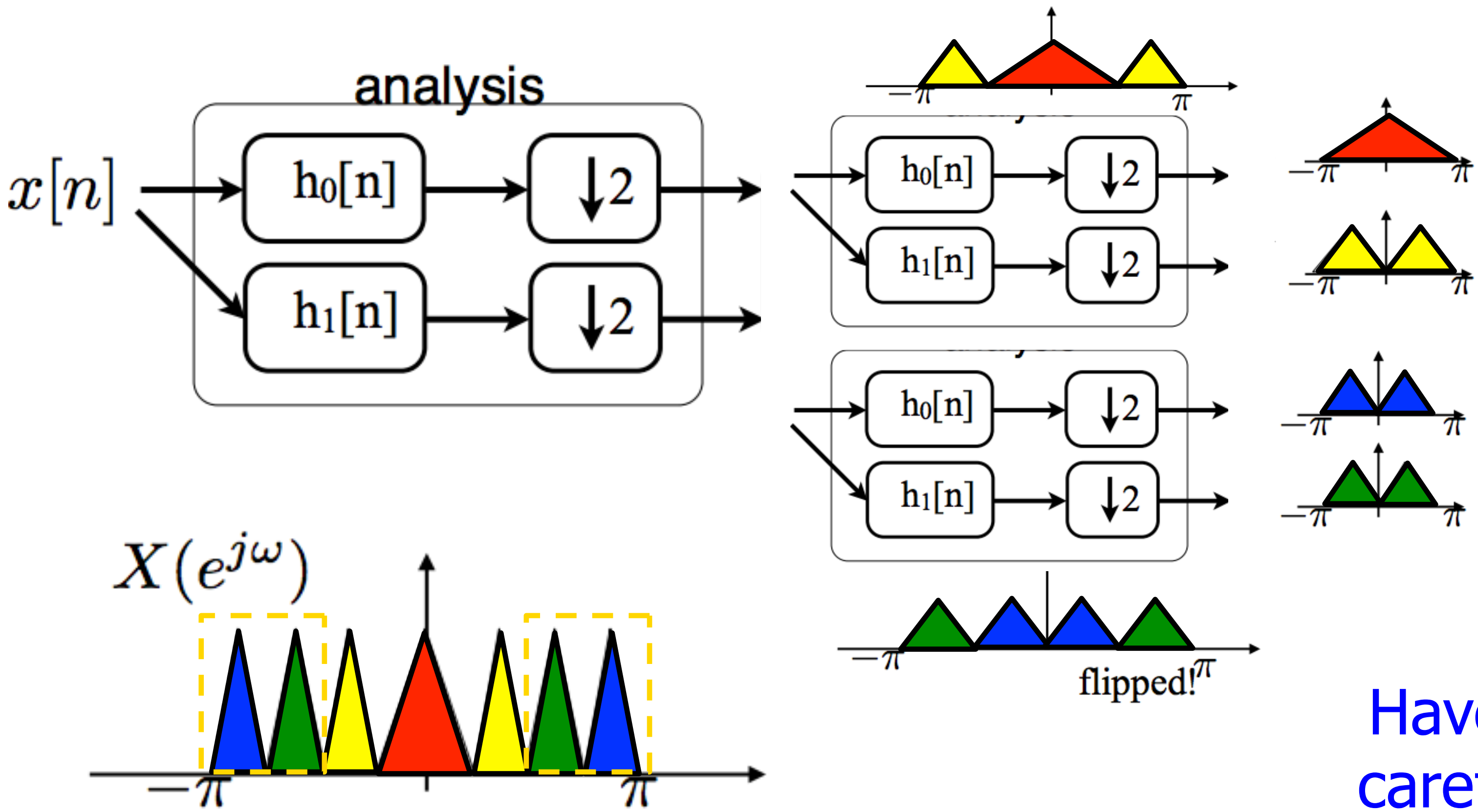
Multi-Rate Filter Banks

- Assume h_0, h_1 are ideal low/high pass with $\omega_C = \pi/2$



Multi-Rate Filter Banks

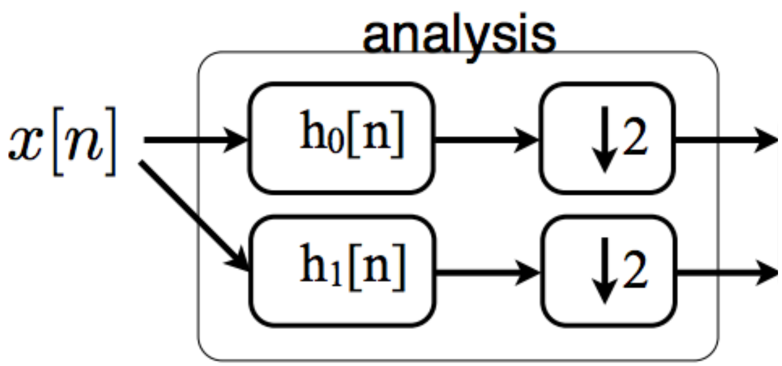
- Assume h_0, h_1 are ideal low/high pass with $\omega_C = \pi/2$

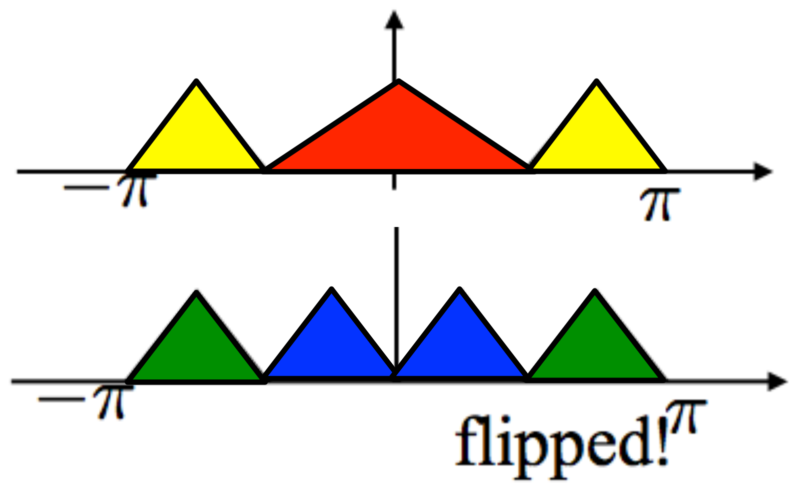


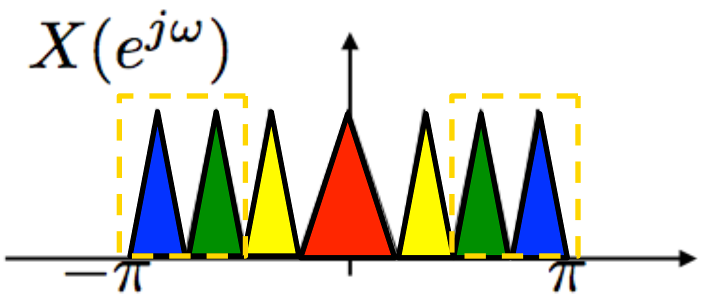
Have to be careful with order!

Multi-Rate Filter Banks

- Assume h_0, h_1 are ideal low/high pass with $\omega_C = \pi/2$

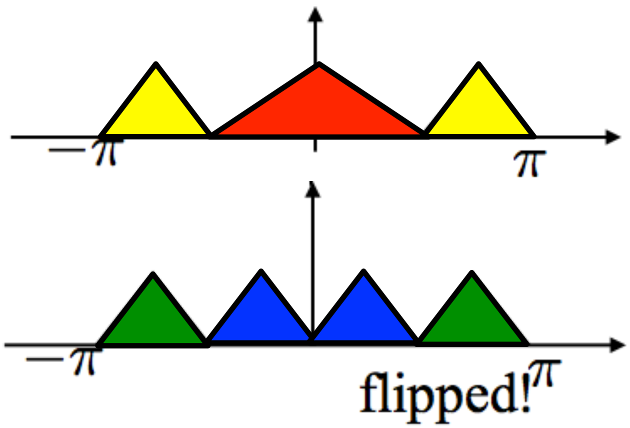


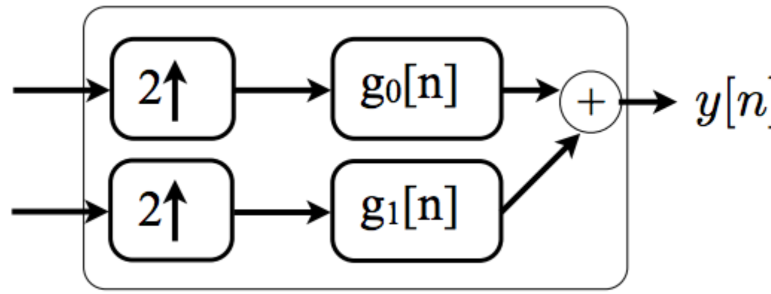




Multi-Rate Filter Banks

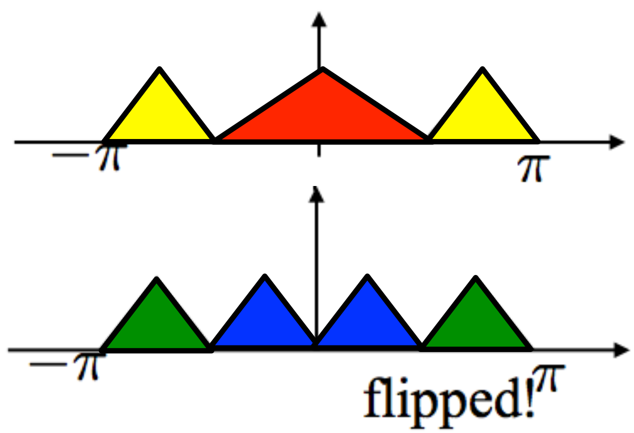
- Assume h_0, h_1 are ideal low/high pass with $\omega_C = \pi/2$

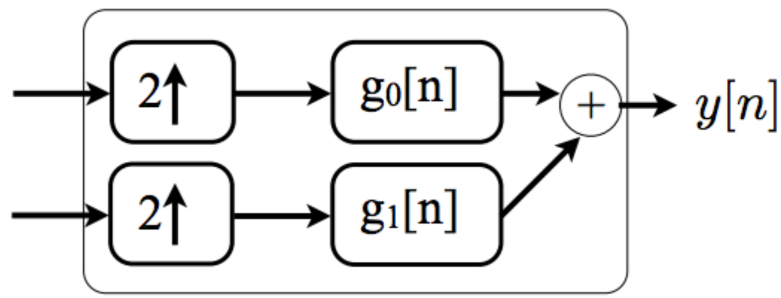


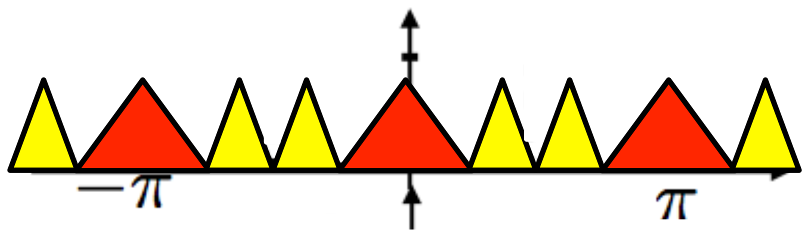


Multi-Rate Filter Banks

- Assume h_0, h_1 are ideal low/high pass with $\omega_C = \pi/2$

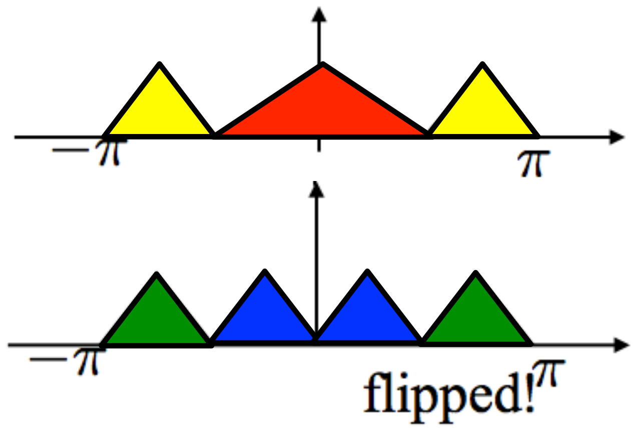


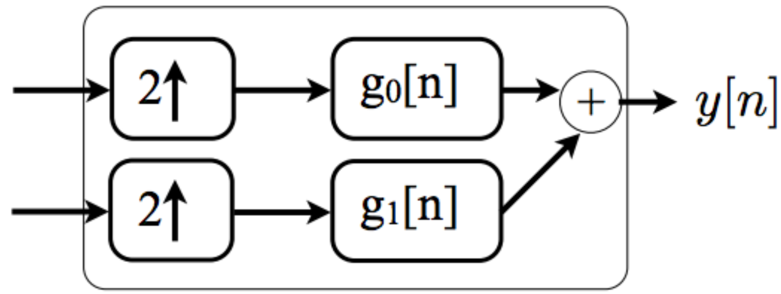


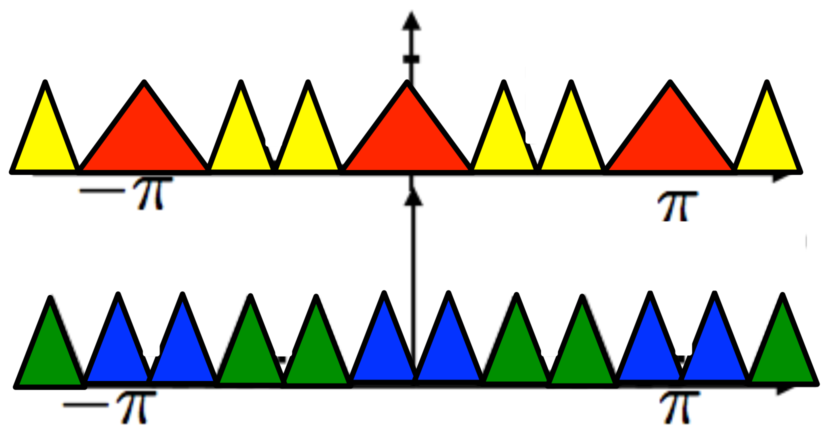


Multi-Rate Filter Banks

- Assume h_0, h_1 are ideal low/high pass with $\omega_C = \pi/2$

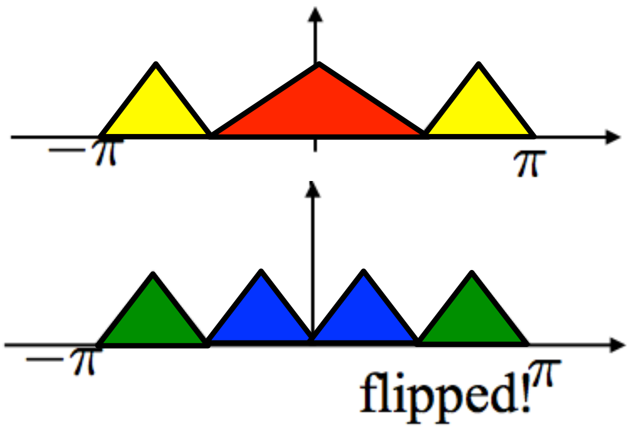


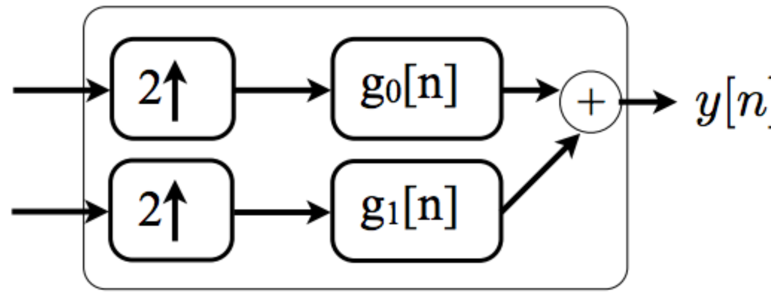


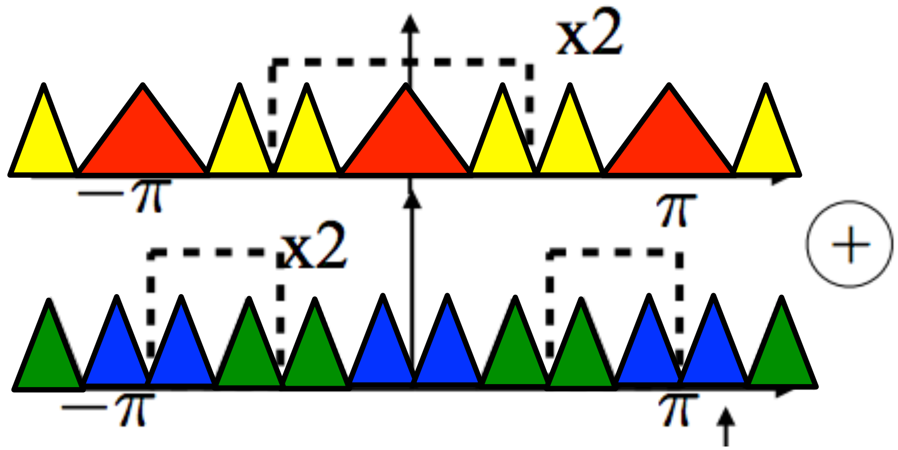


Multi-Rate Filter Banks

- Assume h_0, h_1 are ideal low/high pass with $\omega_C = \pi/2$

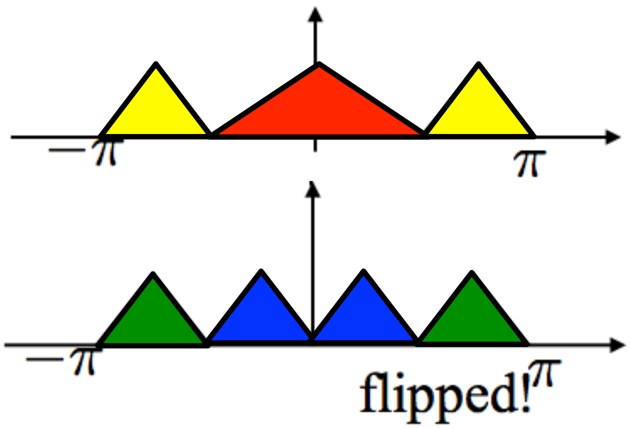


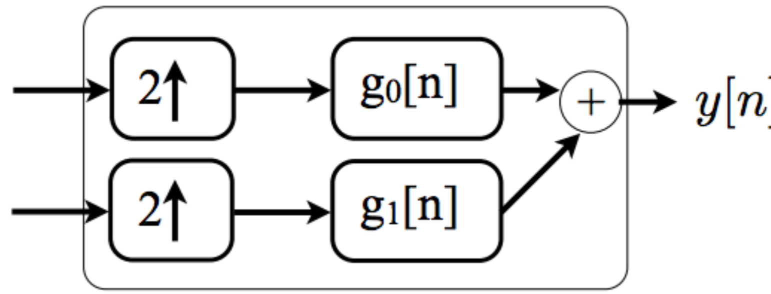


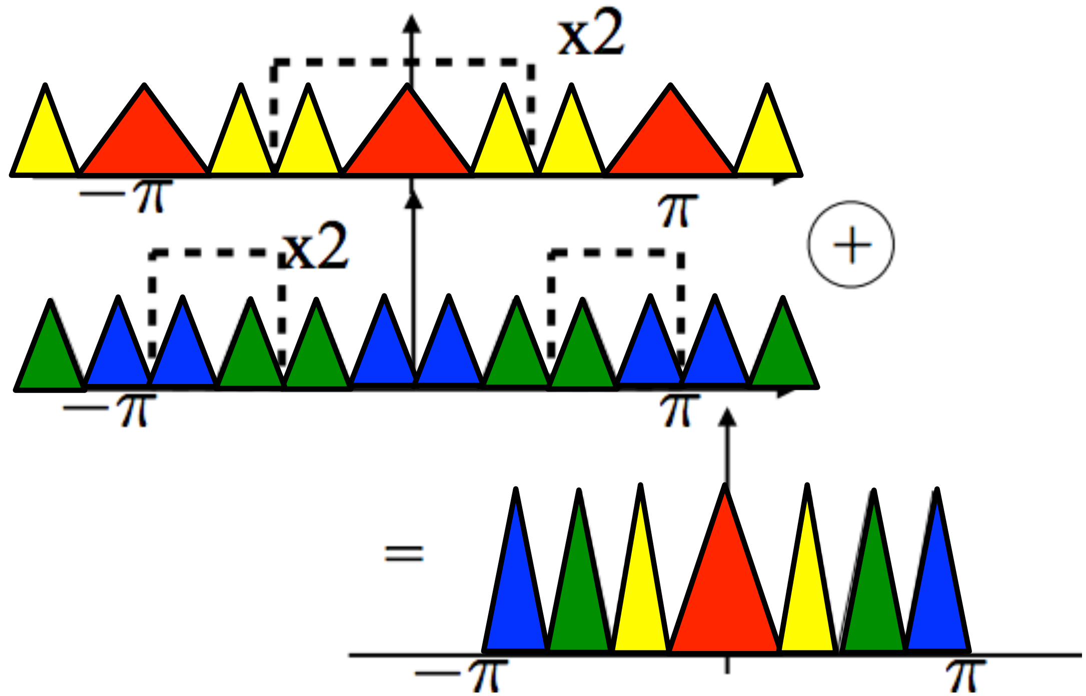


Multi-Rate Filter Banks

- Assume h_0, h_1 are ideal low/high pass with $\omega_C = \pi/2$

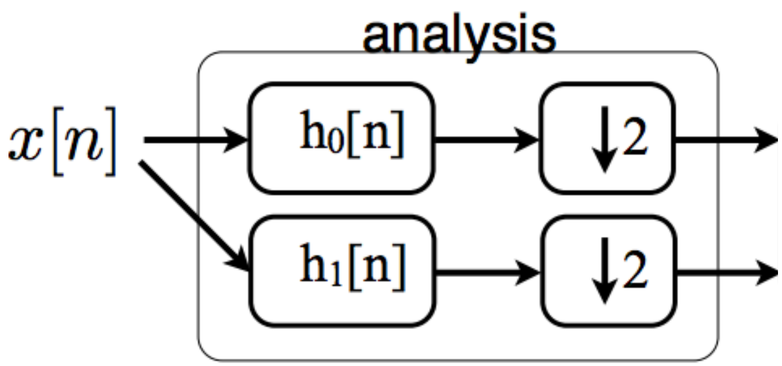


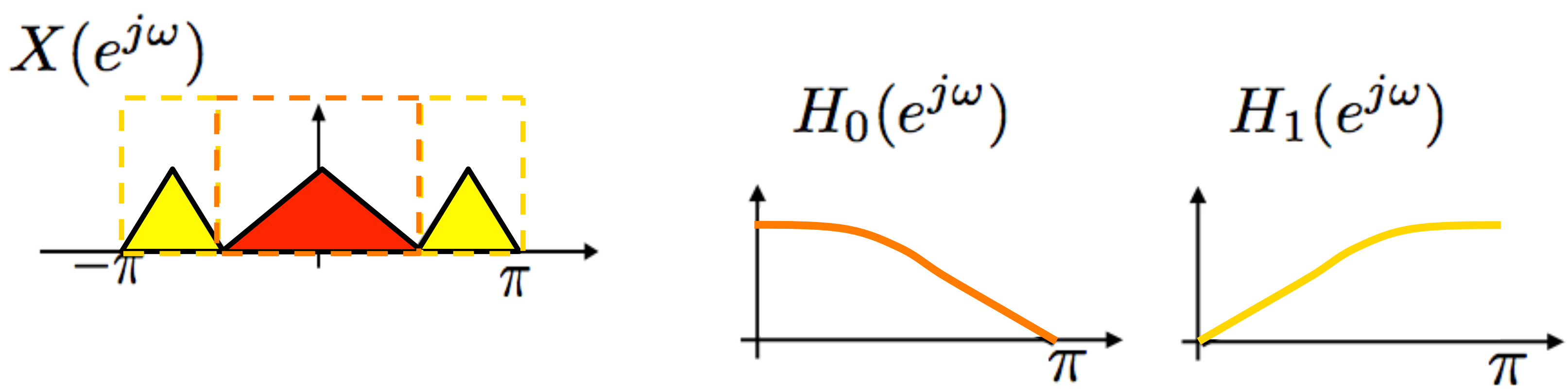




Multi-Rate Filter Banks

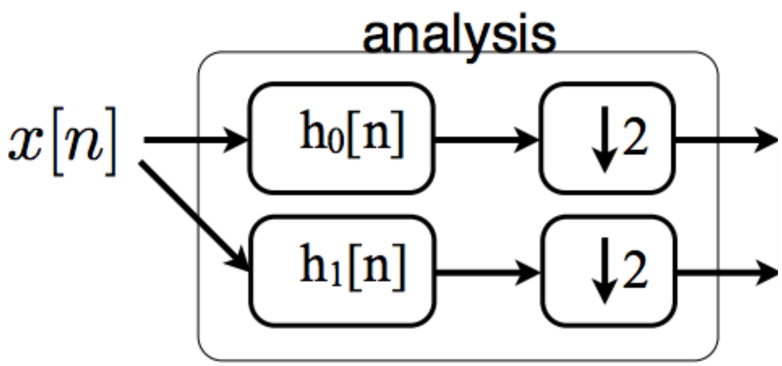
- h_0, h_1 are NOT ideal low/high pass

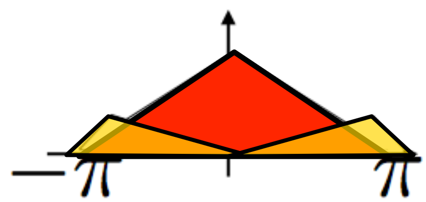


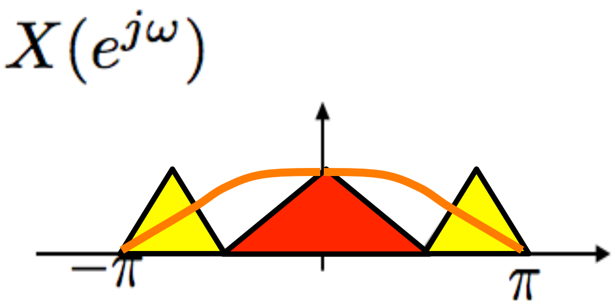


Non Ideal Filters

- h_0, h_1 are NOT ideal low/high pass

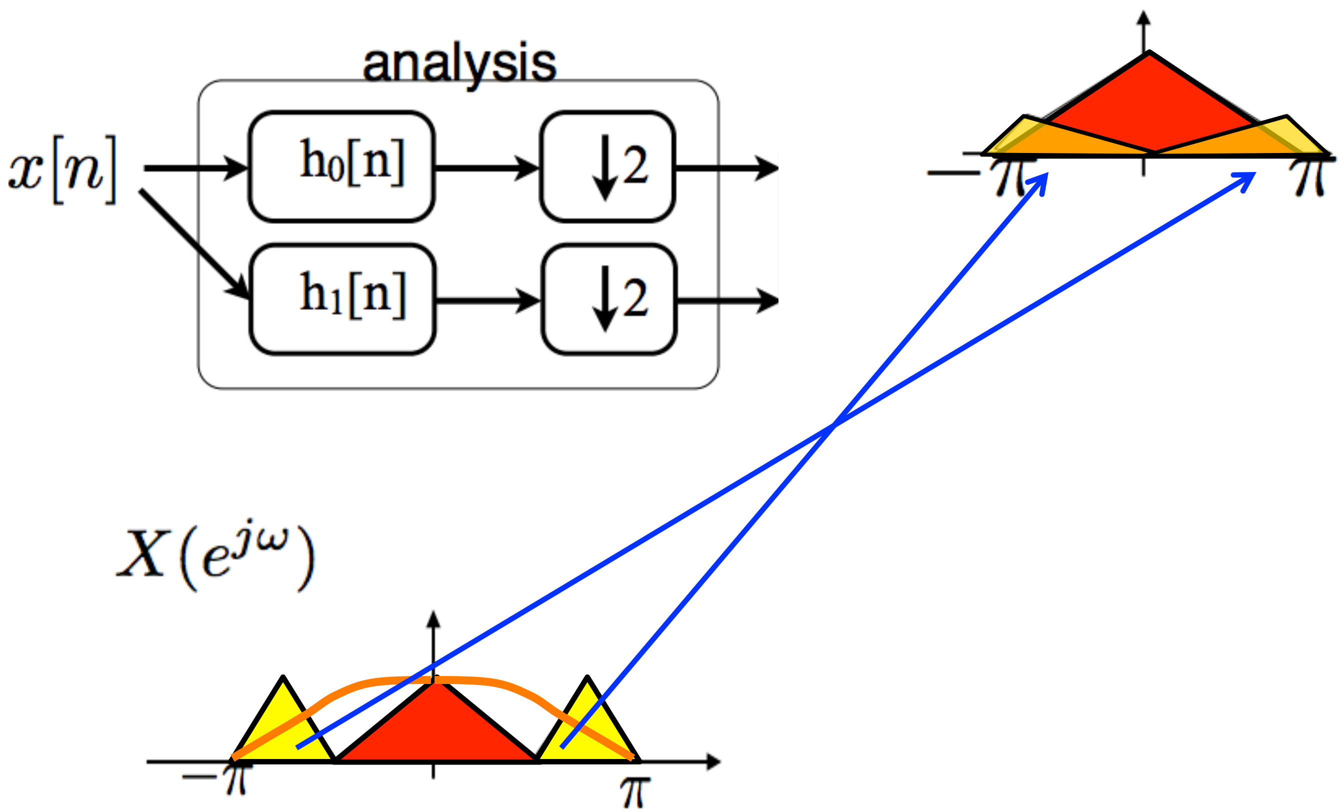






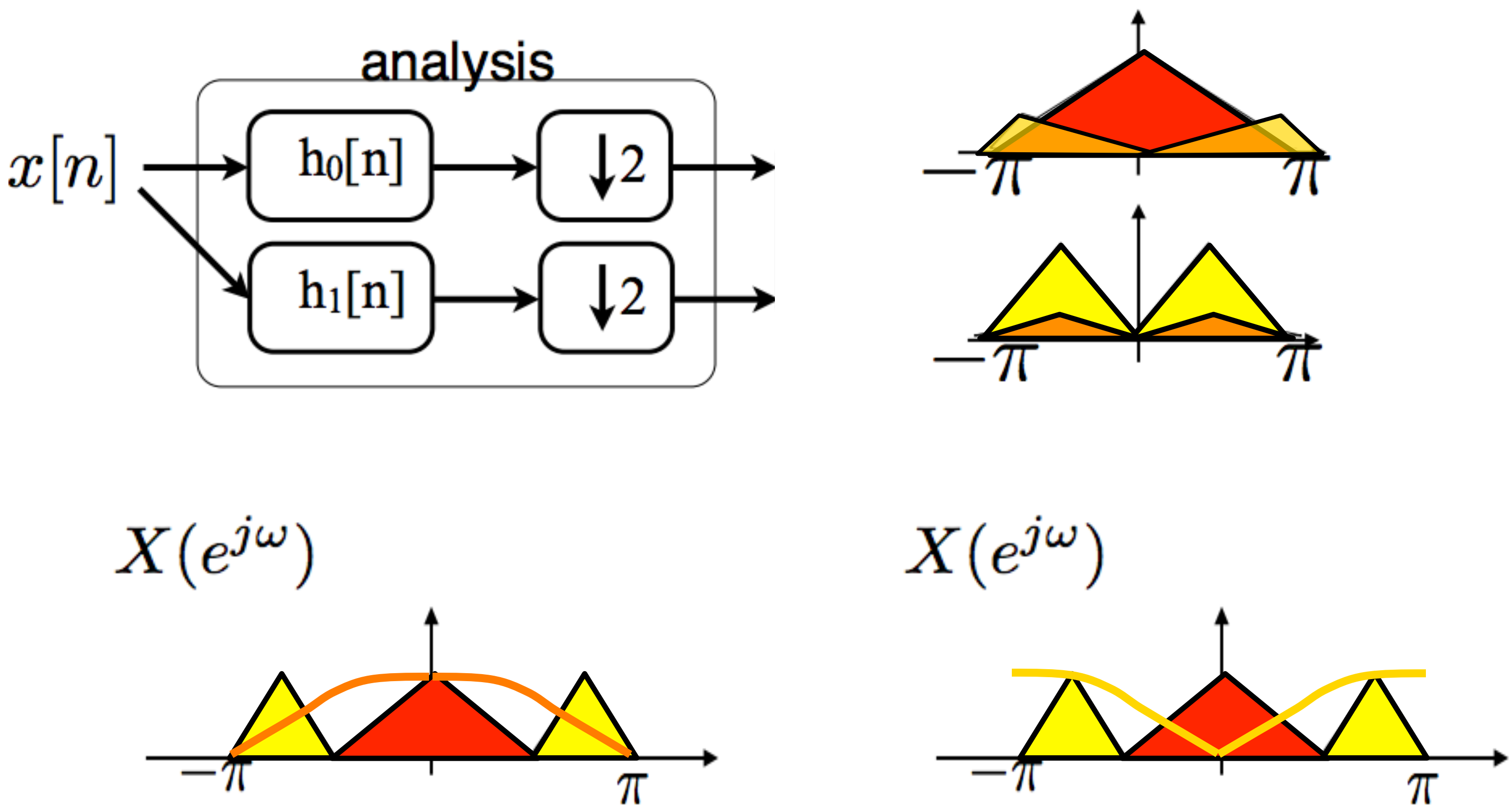
Non Ideal Filters

- h_0, h_1 are NOT ideal low/high pass



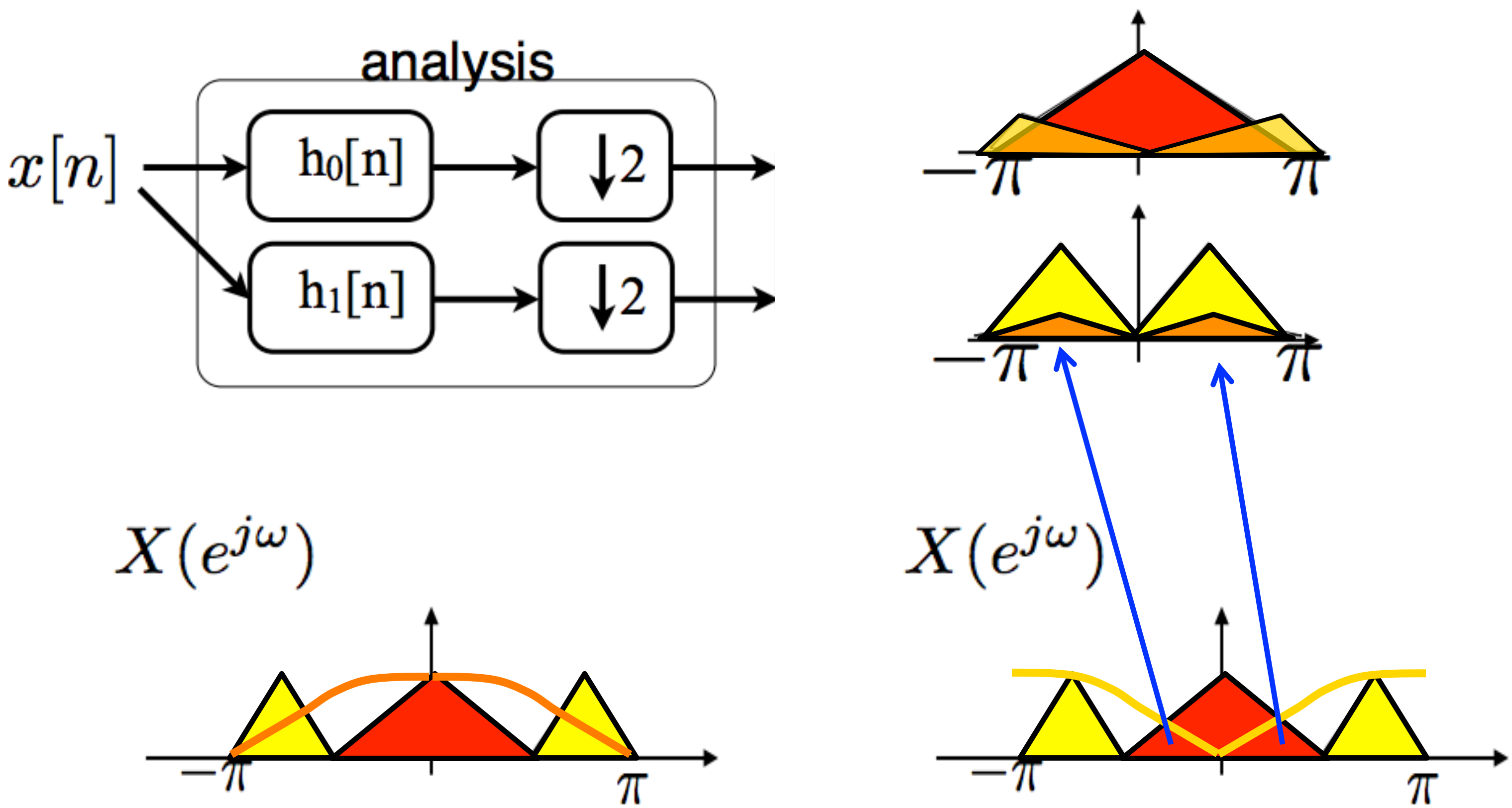
Non Ideal Filters

- h_0, h_1 are NOT ideal low/high pass

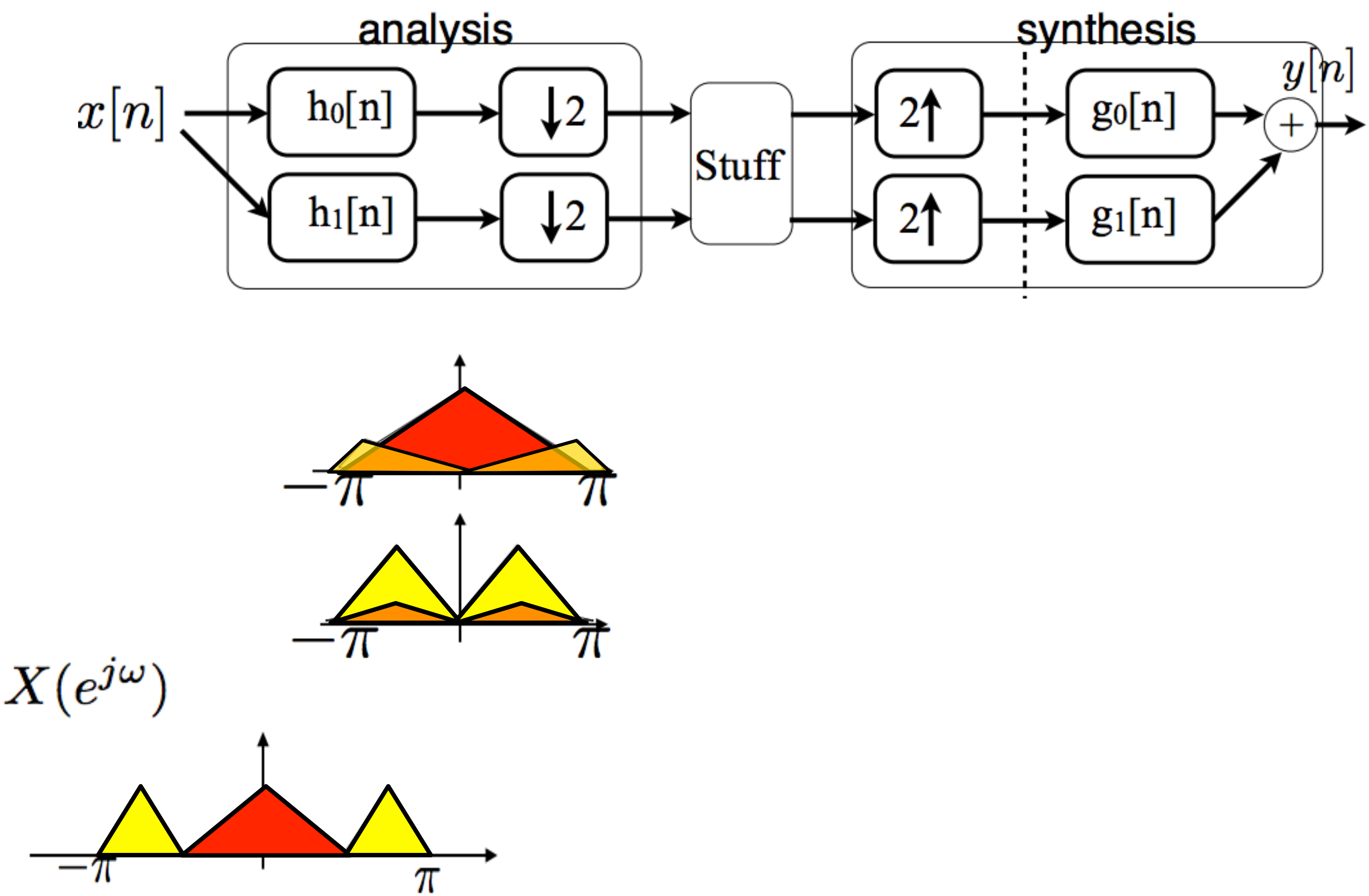


Non Ideal Filters

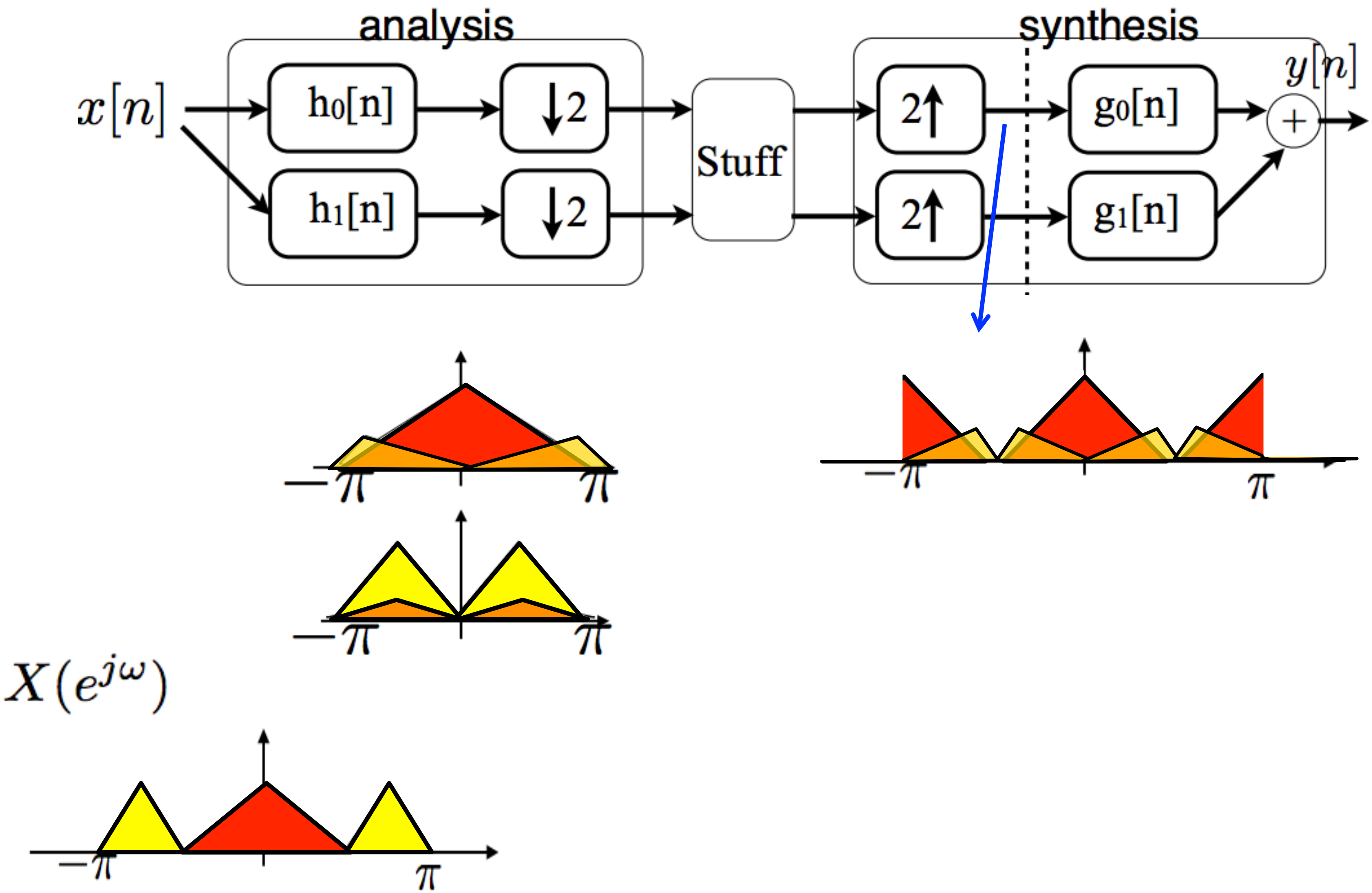
- h_0, h_1 are NOT ideal low/high pass



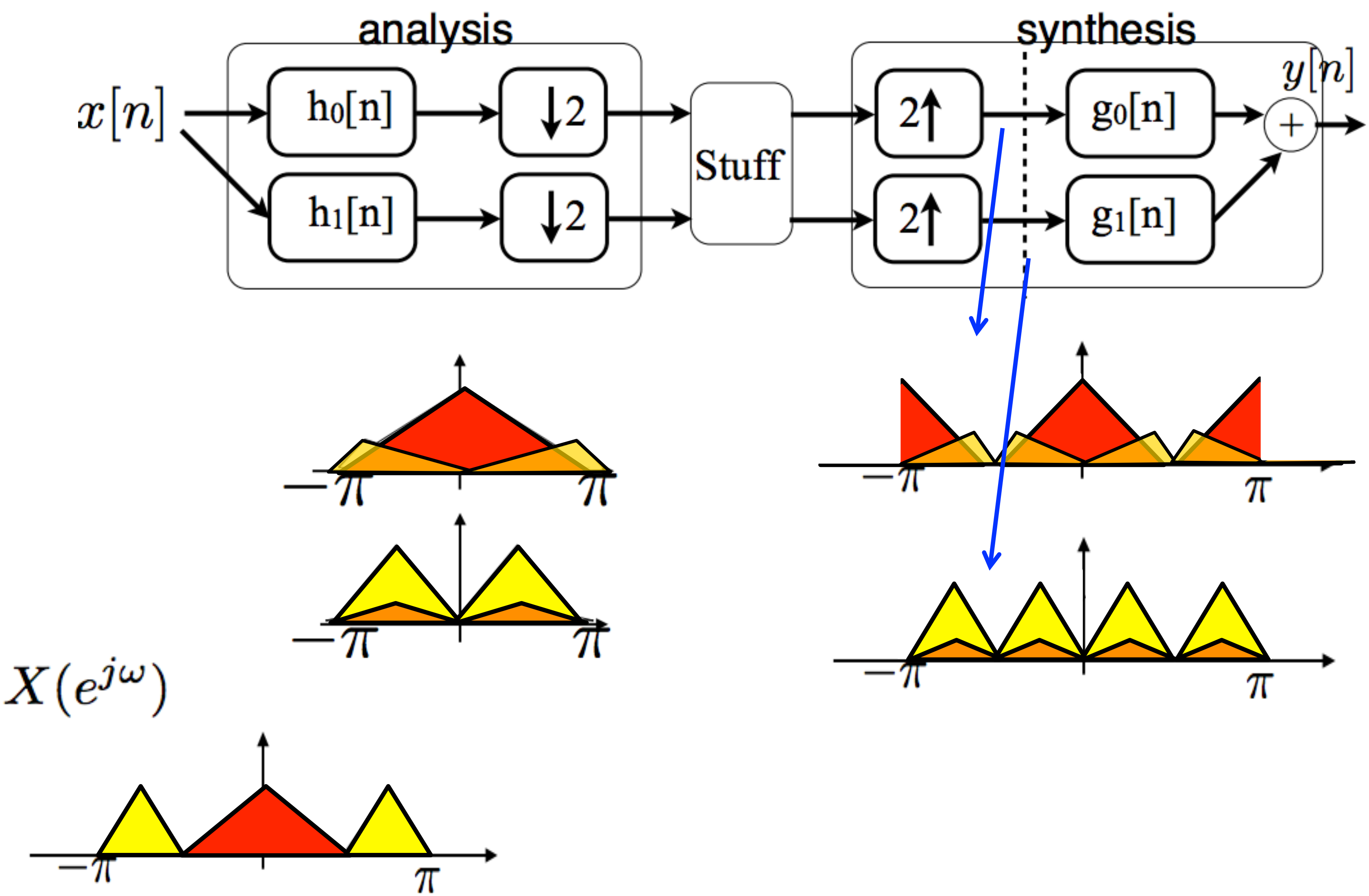
Non Ideal Filters



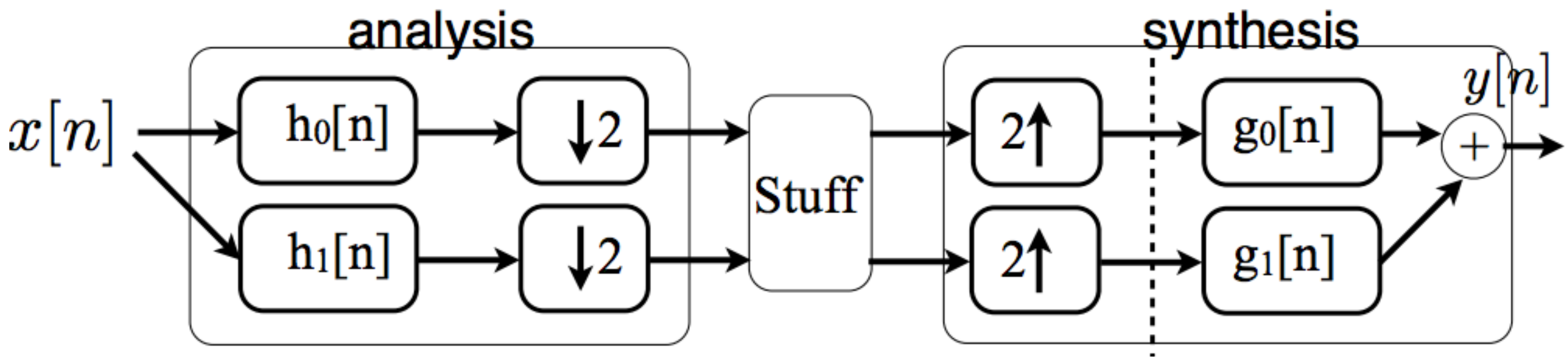
Non Ideal Filters



Non Ideal Filters



Perfect Reconstruction non-Ideal Filters

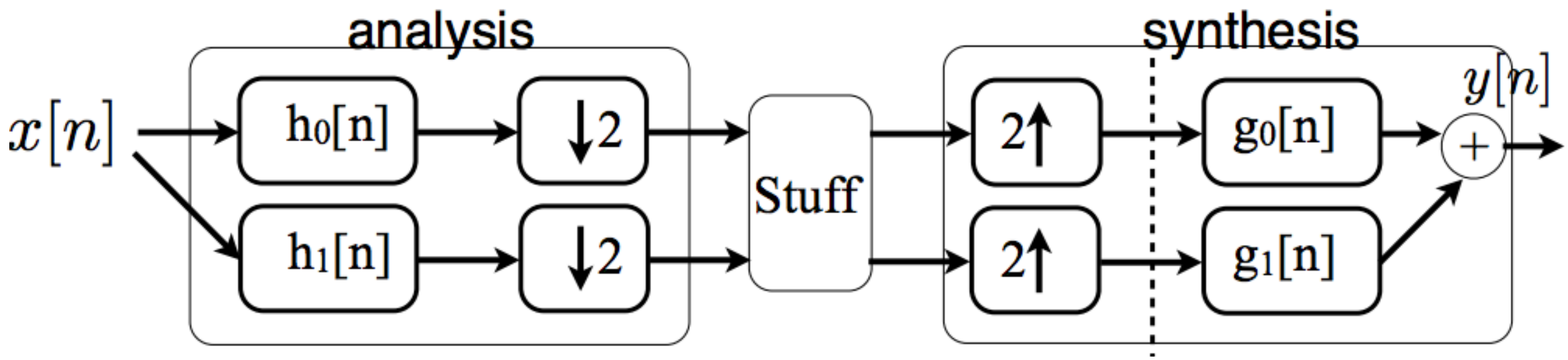


$$Y(e^{j\omega}) = \frac{1}{2}\left[G_0(e^{j\omega})H_0(e^{j\omega}) + G_1(e^{j\omega})H_1(e^{j\omega})\right] X(e^{j\omega})$$
$$+\frac{1}{2}\left[G_0(e^{j\omega})H_0(e^{j(\omega-\pi)}) + G_1(e^{j\omega})H_1(e^{j(\omega-\pi)})\right] X(e^{j(\omega-\pi)})$$

need to cancel!

aliasing

Quadrature Mirror Filters



Quadrature mirror filters

$$
\begin{aligned}
H_1(e^{j\omega}) &= H_0(e^{j(\omega-\pi)}) \\
G_0(e^{j\omega}) &= 2H_0(e^{j\omega}) \\
G_1(e^{j\omega}) &= -2H_1(e^{j\omega})
\end{aligned}
$$

Perfect Reconstruction non-Ideal Filters

$$Y(e^{j\omega}) = \frac{1}{2}\left[G_0(e^{j\omega})H_0(e^{j\omega}) + G_1(e^{j\omega})H_1(e^{j\omega})\right] X(e^{j\omega})$$
$$+\frac{1}{2}\left[G_0(e^{j\omega})H_0(e^{j(\omega-\pi)}) + G_1(e^{j\omega})H_1(e^{j(\omega-\pi)})\right] X(e^{j(\omega-\pi)})$$

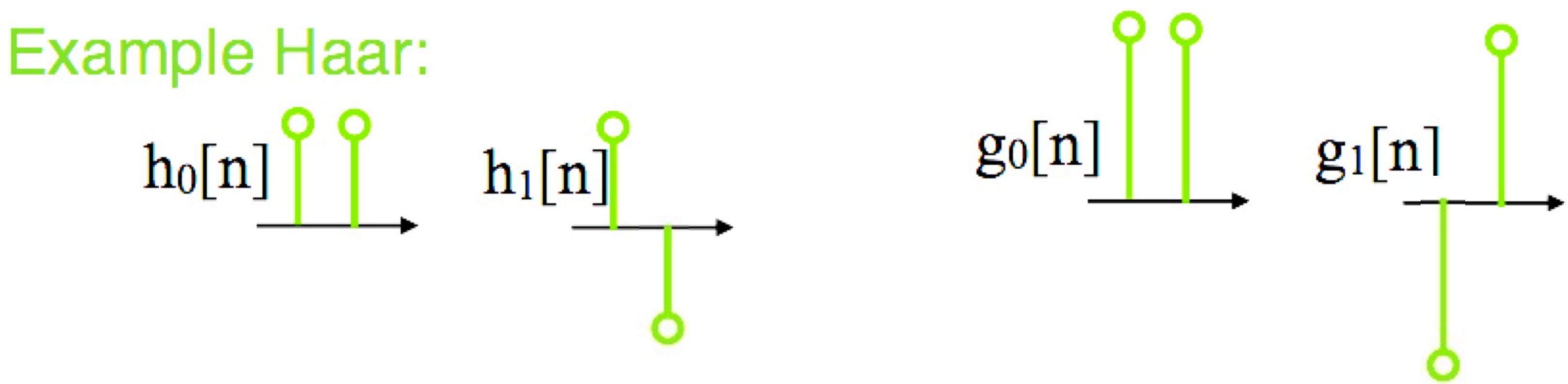
need to cancel!

aliasing

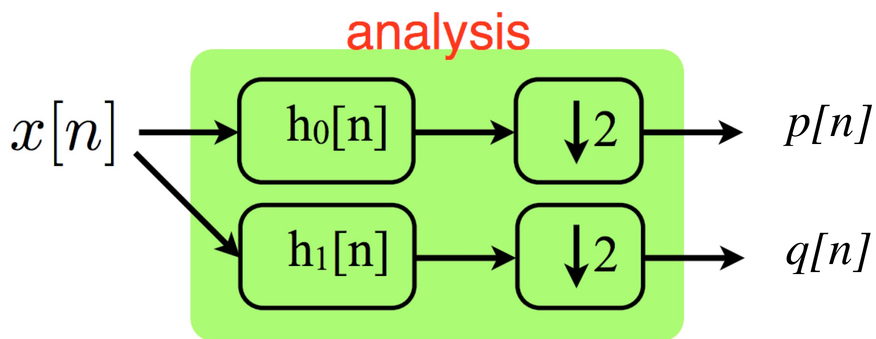
$$H_1(e^{j\omega}) = H_0(e^{j(\omega-\pi)})$$
$$G_0(e^{j\omega}) = 2H_0(e^{j\omega})$$
$$G_1(e^{j\omega}) = -2H_1(e^{j\omega})$$

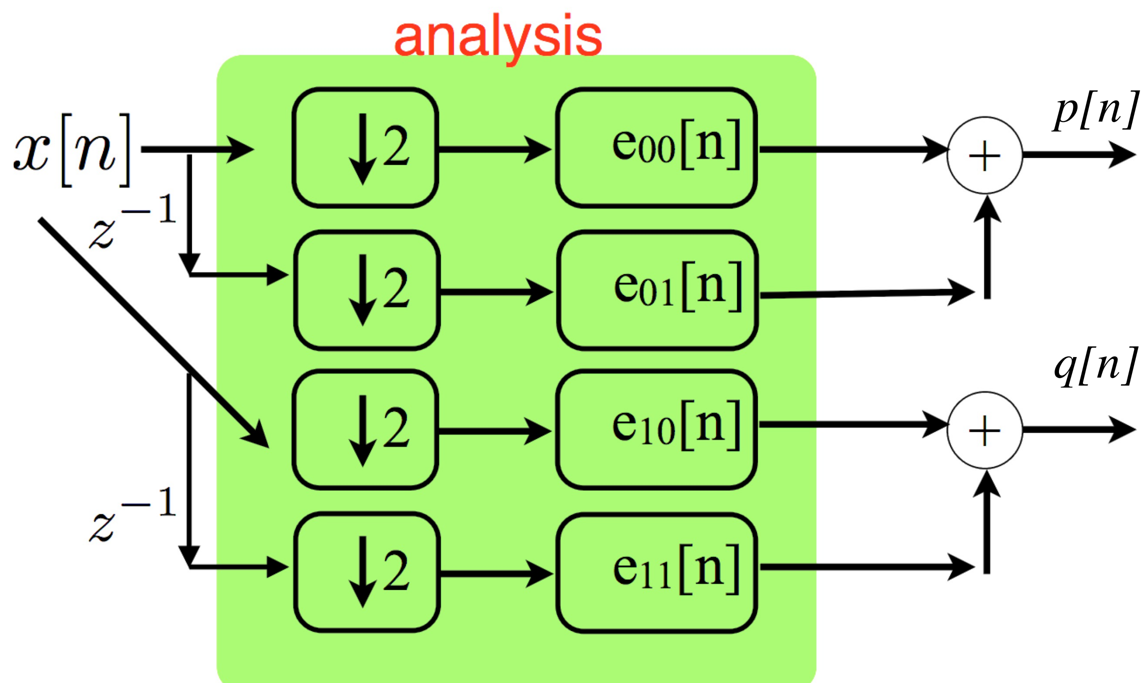
Haar Filter Example

$$
\begin{aligned}
H_1(e^{j\omega}) &= H_0(e^{j(\omega-\pi)}) \\
G_0(e^{j\omega}) &= 2H_0(e^{j\omega}) \\
G_1(e^{j\omega}) &= -2H_1(e^{j\omega})
\end{aligned}
$$

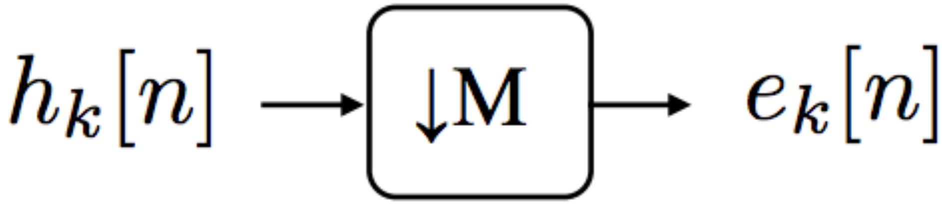


Polyphase Filter Bank

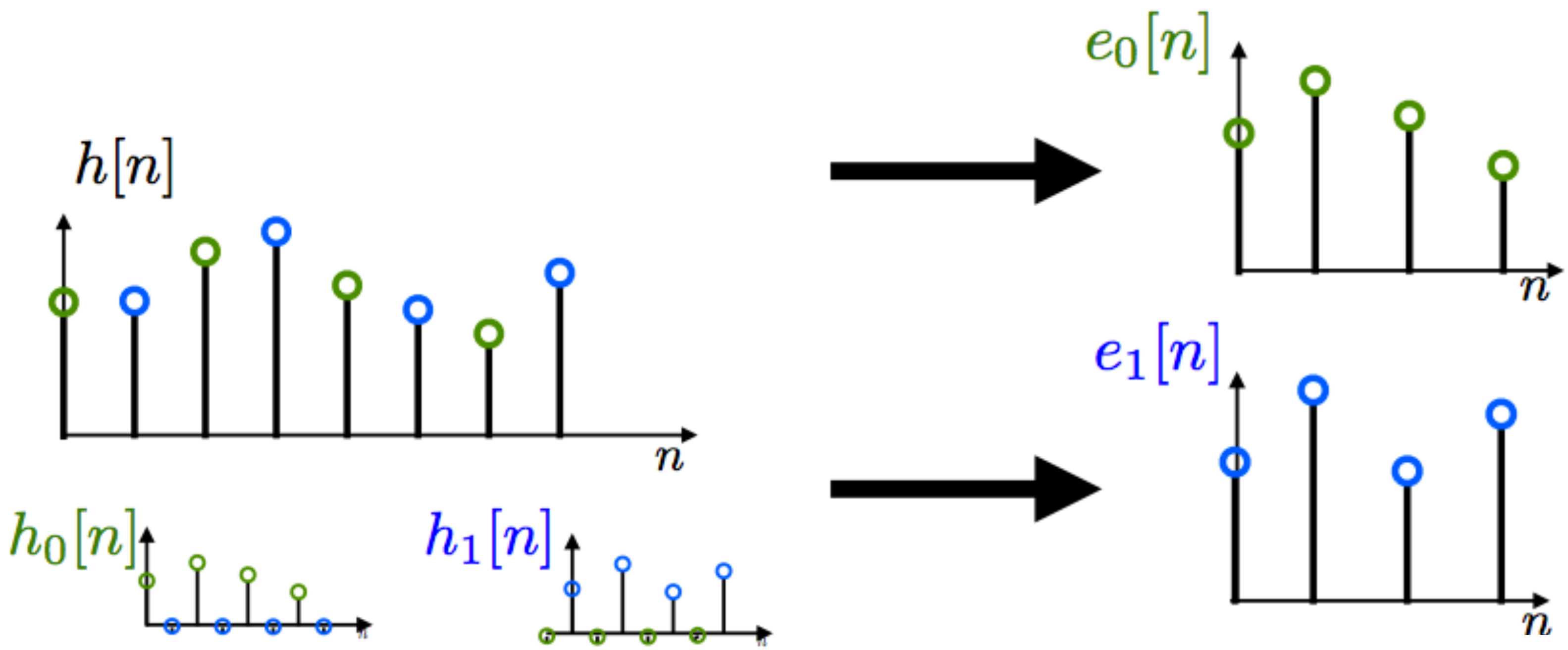




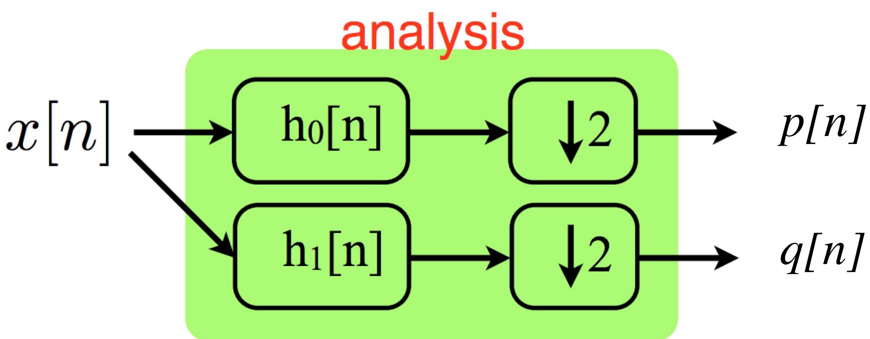
Polyphase Decomposition



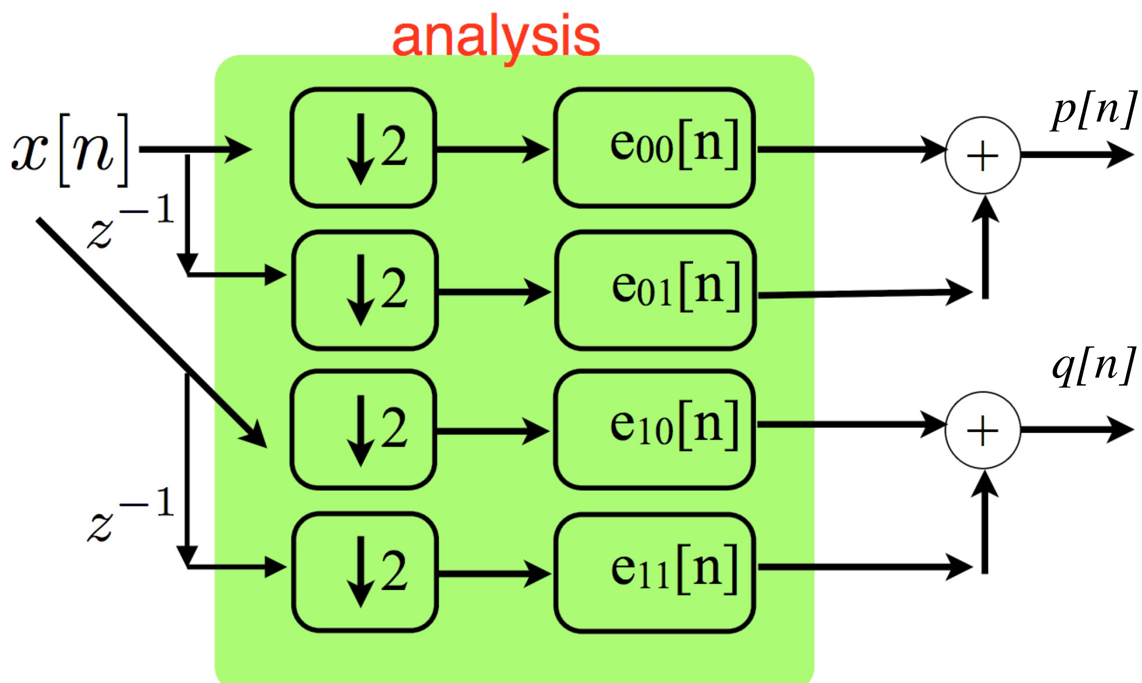
$$e_k[n] = h_k[nM]$$



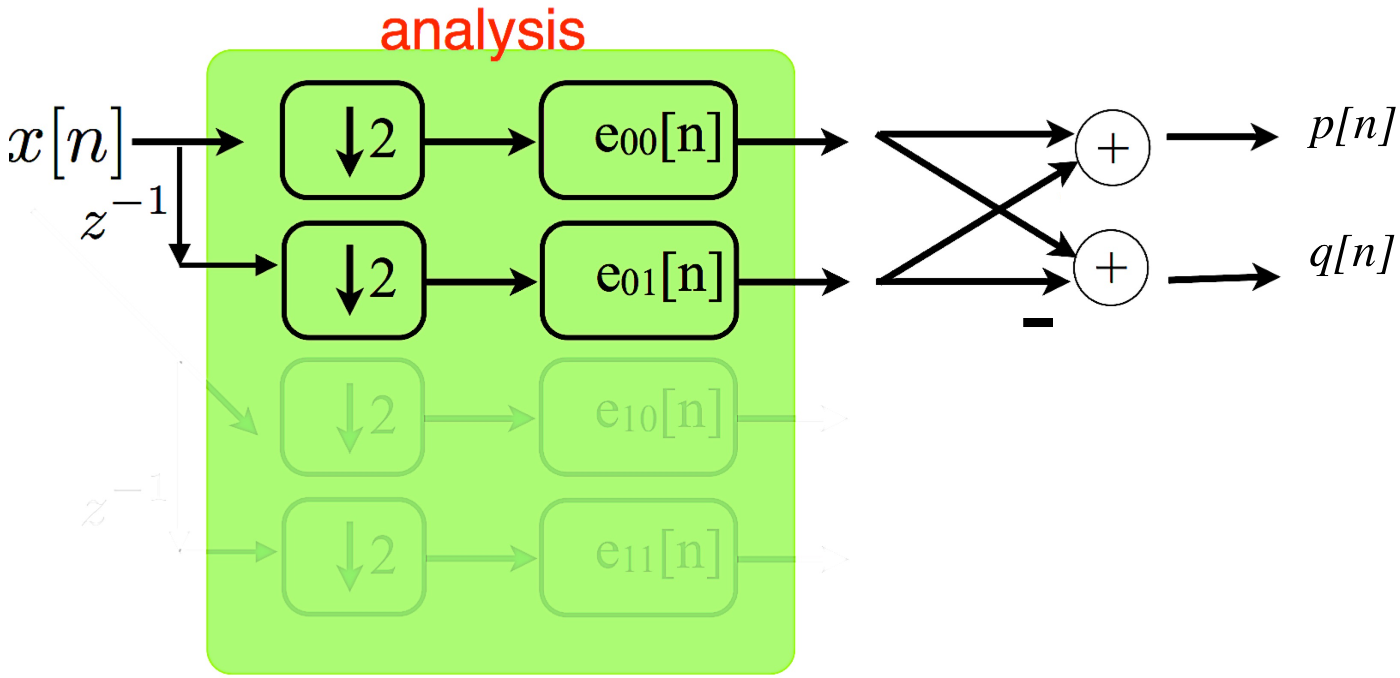
Polyphase Filter Bank



$$
\begin{aligned}
e_{00} &= h_0[2n] \\
e_{01} &= h_0[2n+1] \\
e_{10} &= e_{00}[n] \\
e_{11} &= -e_{01}[n]
\end{aligned}
$$



Polyphase Filter Bank

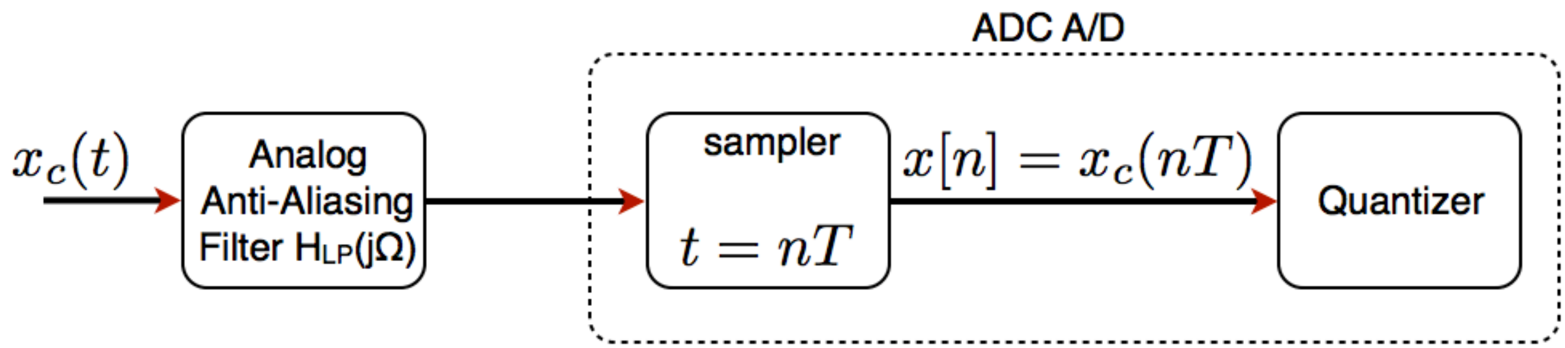


$$
\begin{aligned}
e_{00} &= h_0[2n] \\
e_{01} &= h_0[2n+1] \\
e_{10} &= e_{00}[n] \\
e_{11} &= -e_{01}[n]
\end{aligned}
$$

ADC

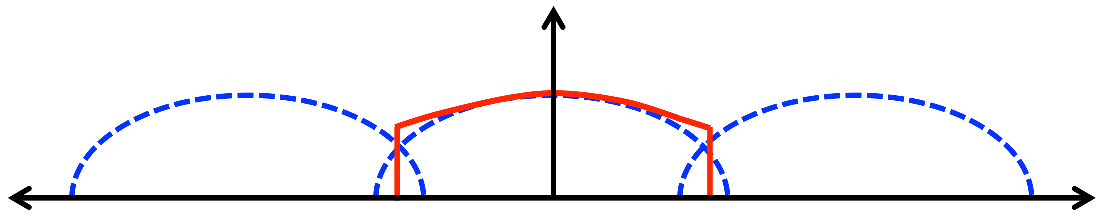
Analog to Digital Converter

Anti-Aliasing Filter with ADC



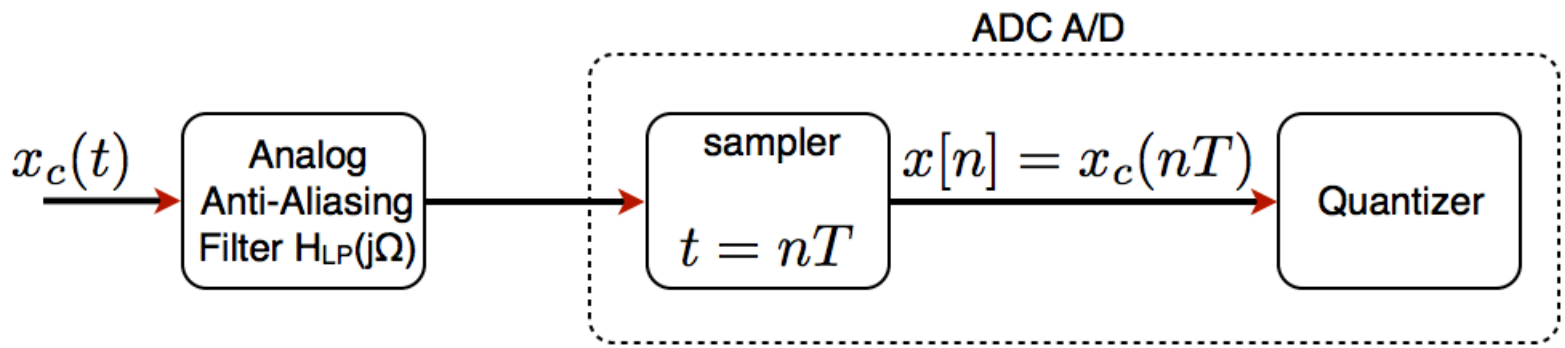
Aliasing

- If $\Omega_N > \Omega_s/2$, $x_r(t)$ an aliased version of $x_c(t)$

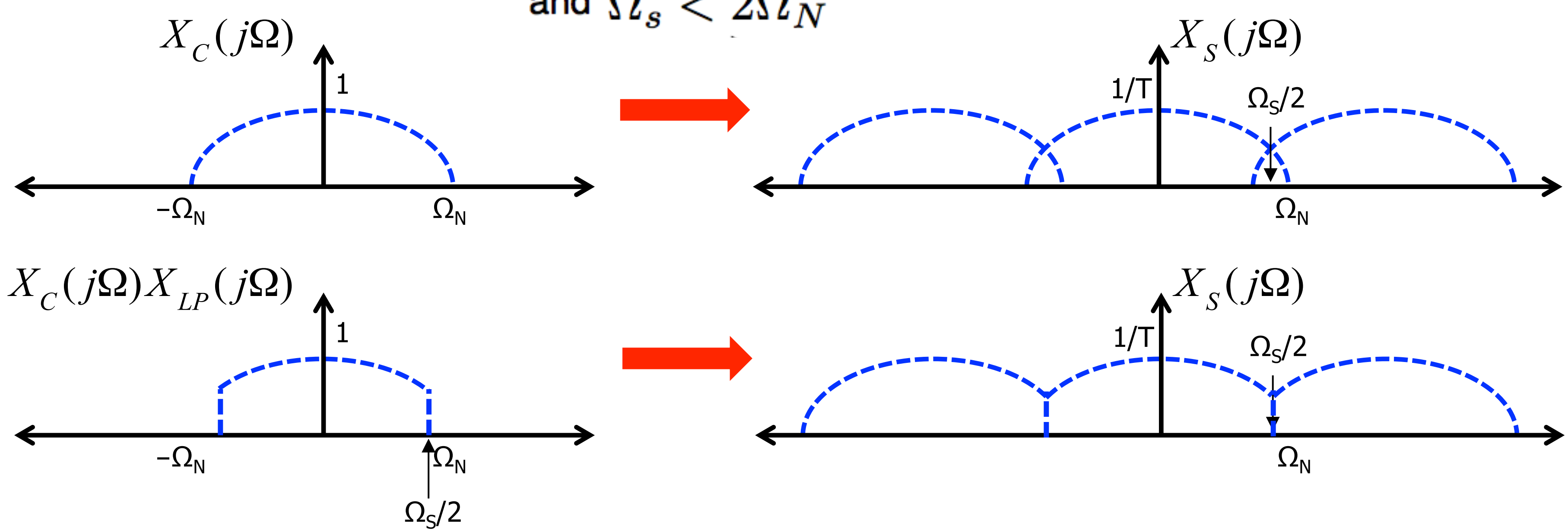


$$X_r(j\Omega) = \begin{cases} TX_s(j\Omega) & \text{if} \quad |\Omega| \le \Omega_s/2 \\ 0 & \text{otherwise} \end{cases}$$

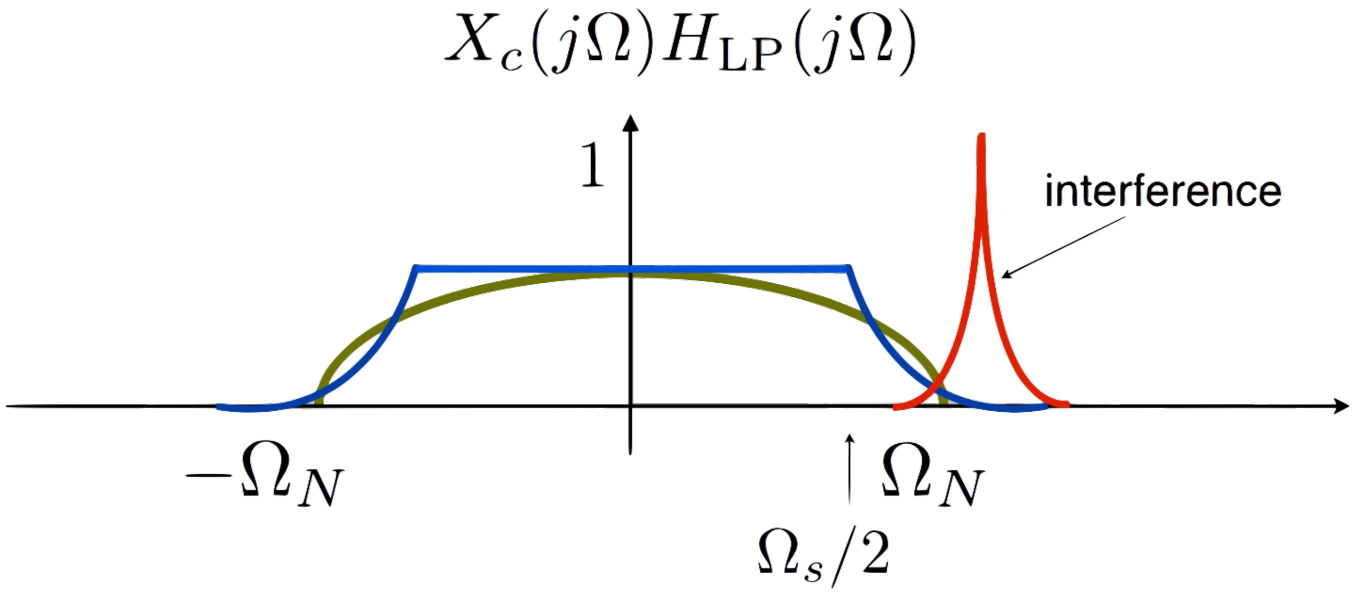
Anti-Aliasing Filter with ADC



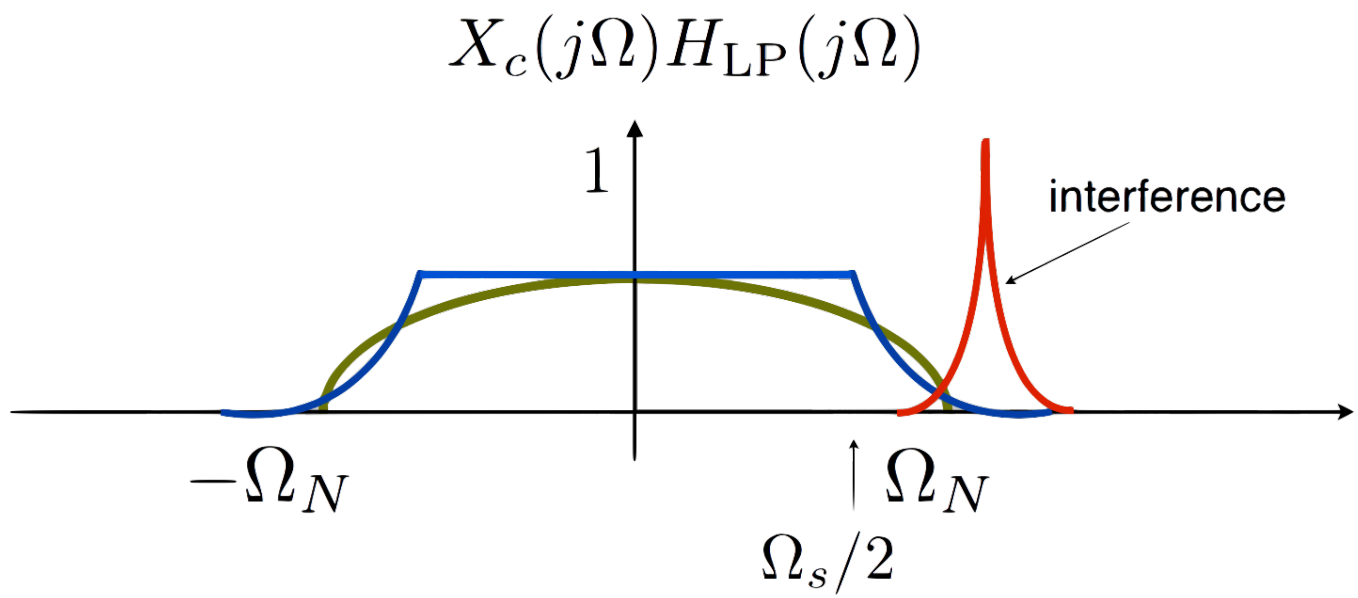
and $\Omega_s < 2\Omega_N$

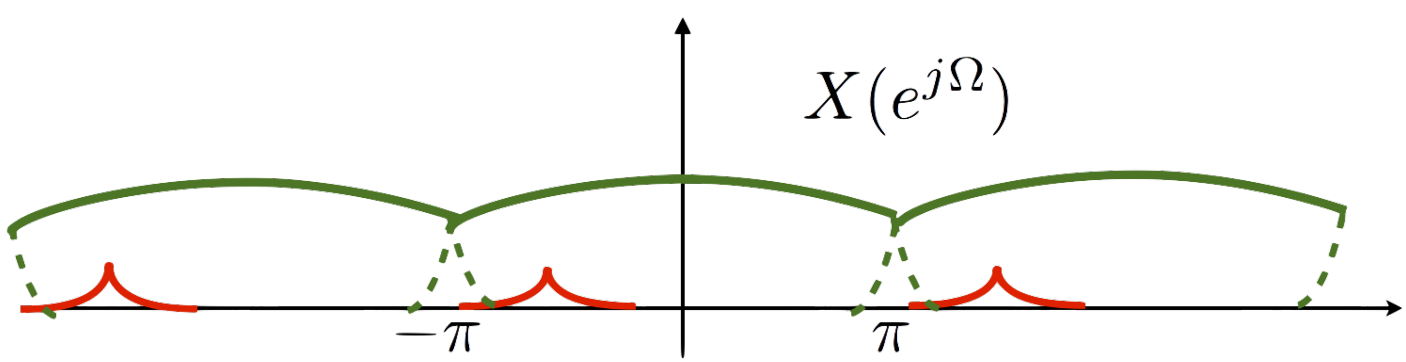


Non-Ideal Anti-Aliasing Filter

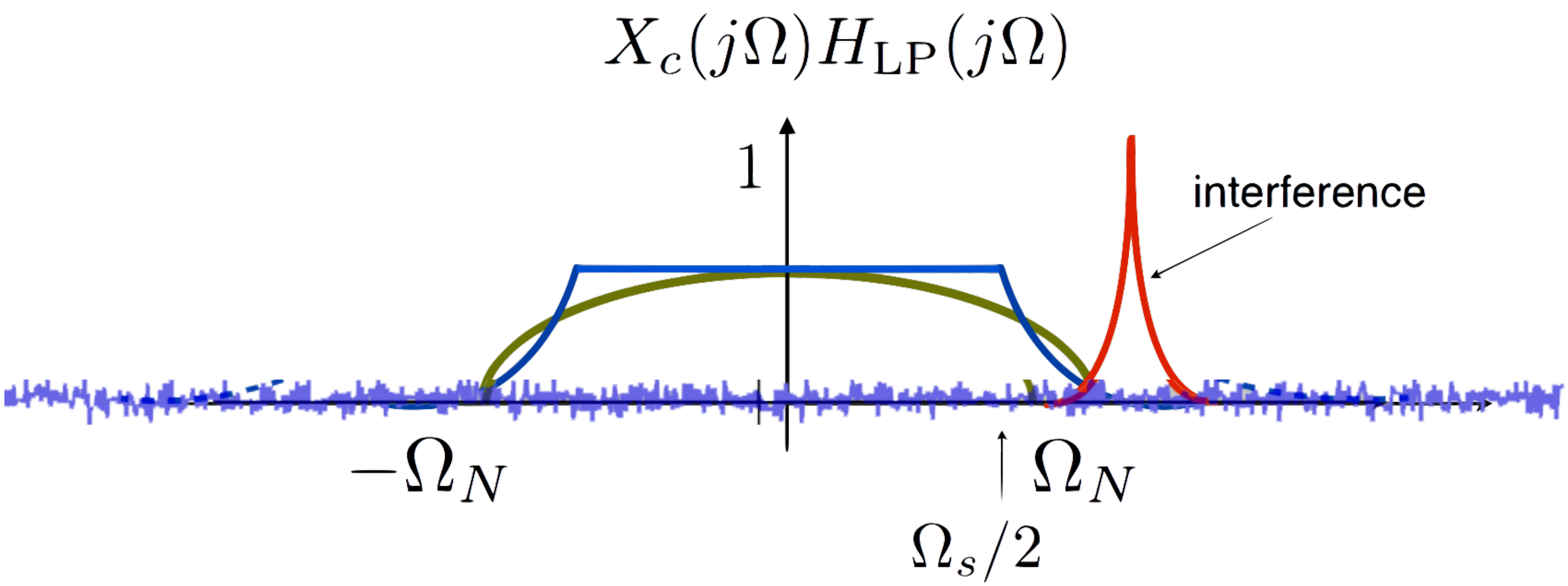


Non-Ideal Anti-Aliasing Filter



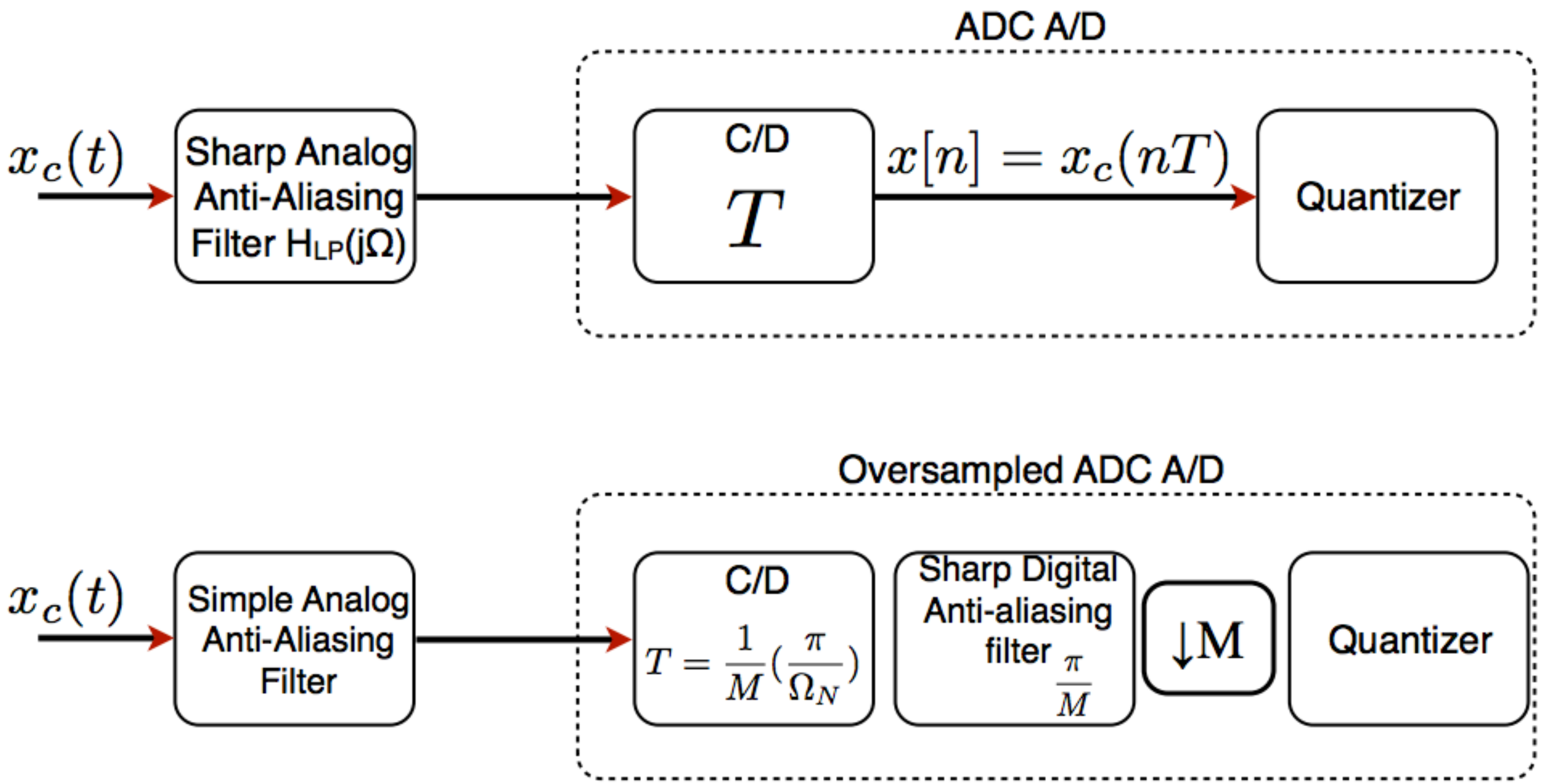


Non-Ideal Anti-Aliasing Filter

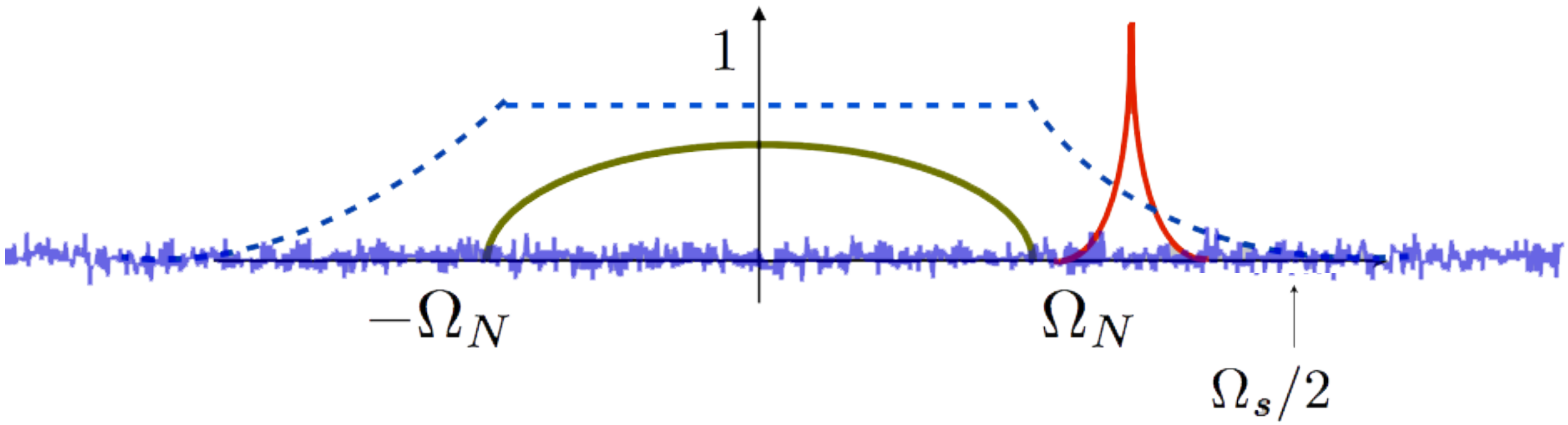


- Problem: Hard to implement sharp analog filter
- Consequence: Crop part of the signal and suffer from noise and interference

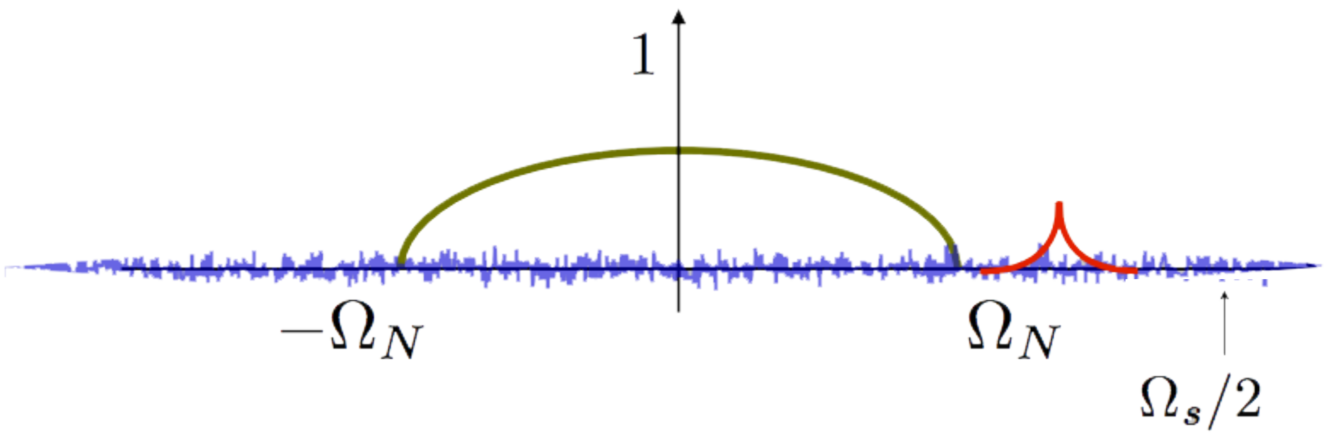
Oversampled ADC



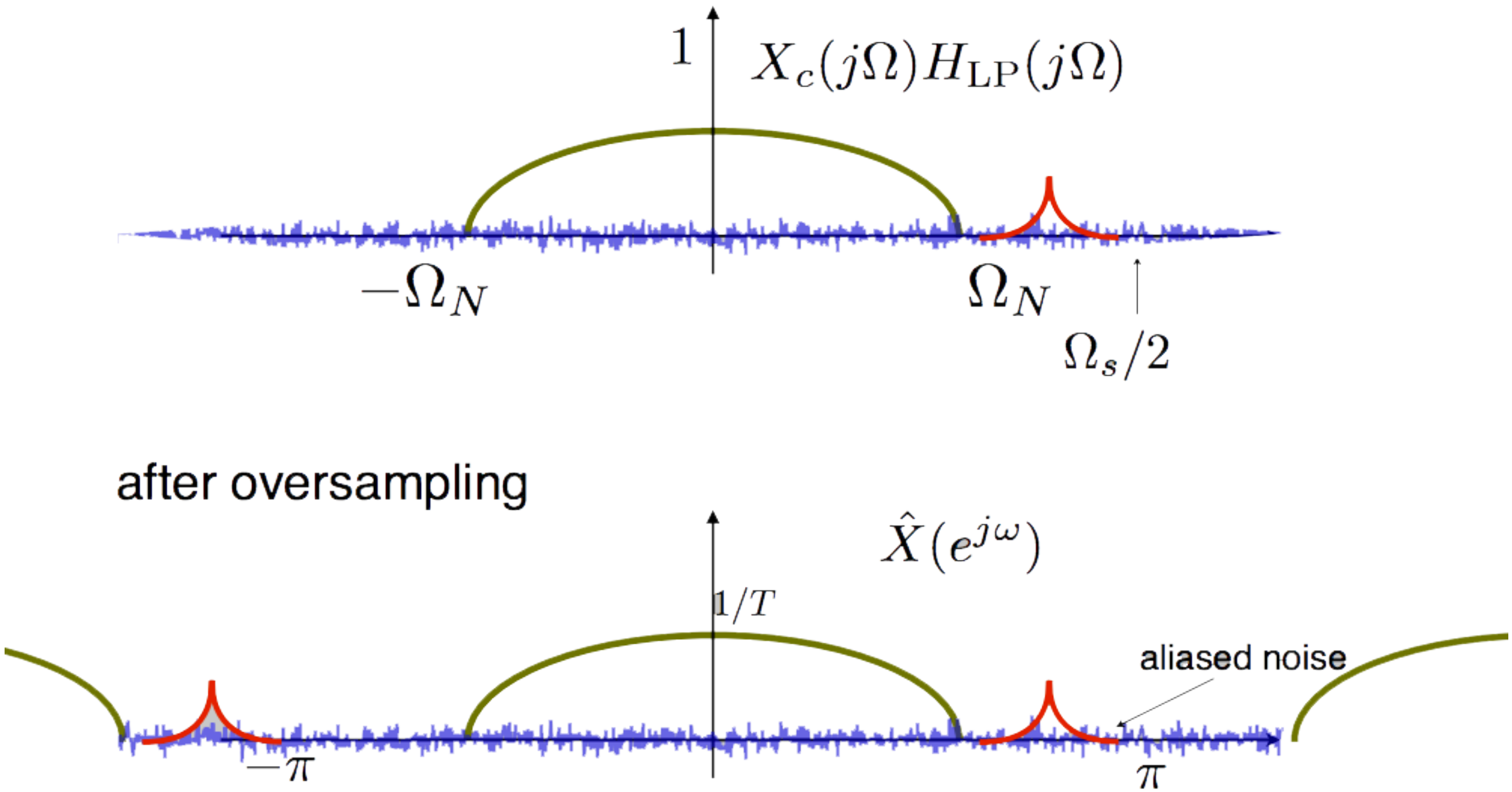
Oversampled ADC – Simple filter



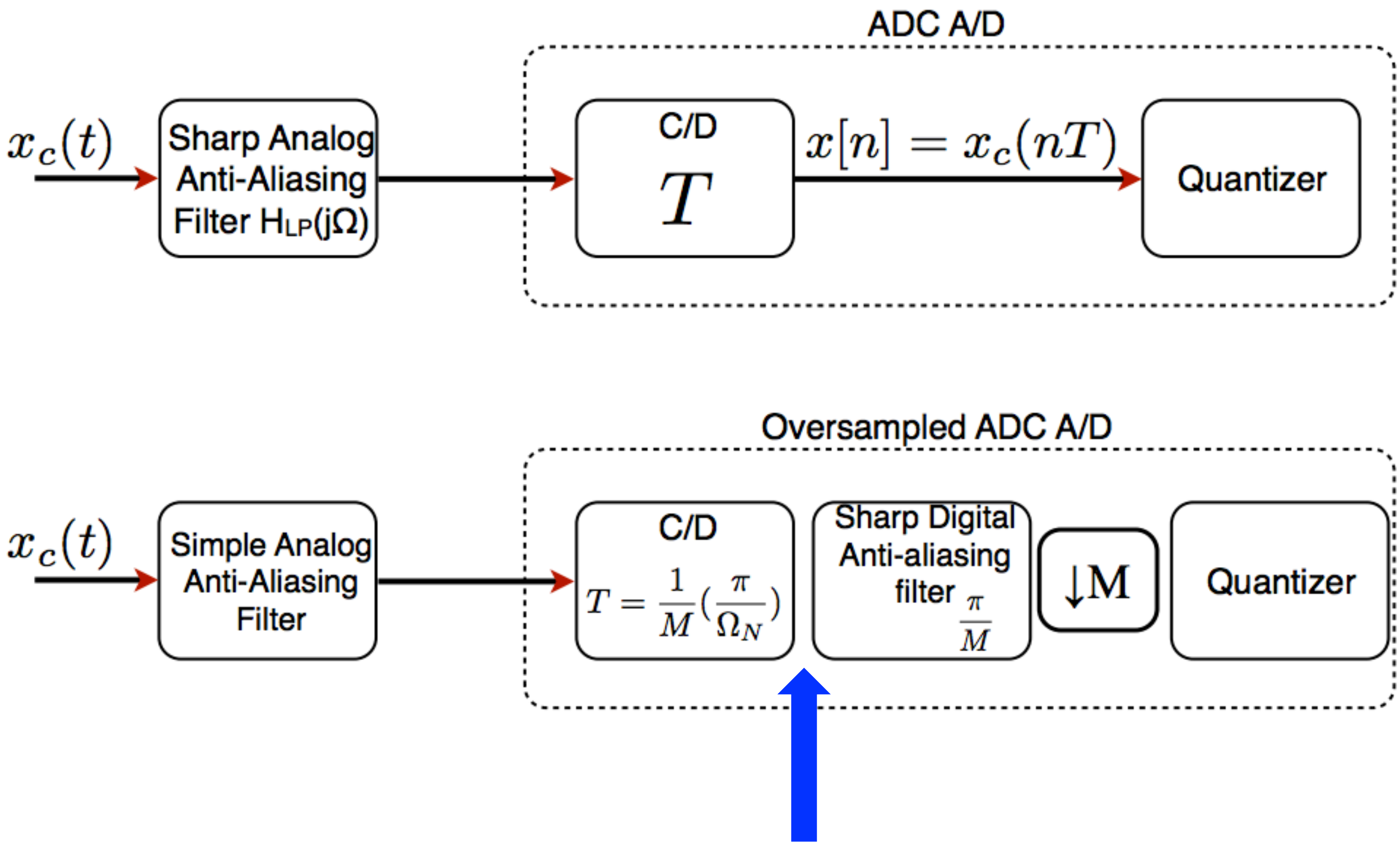
$X_c(j\Omega)H_{\text{LP}}(j\Omega)$



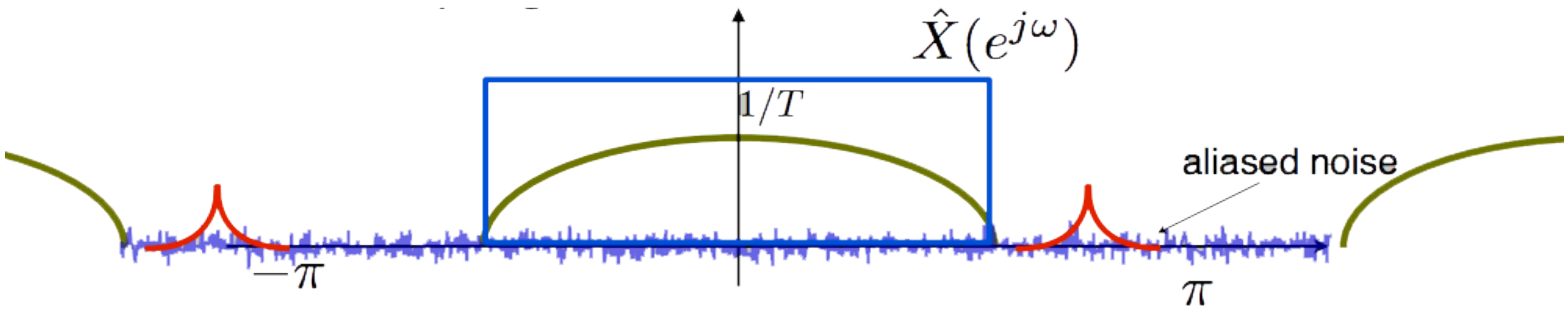
Oversampled ADC – M=2



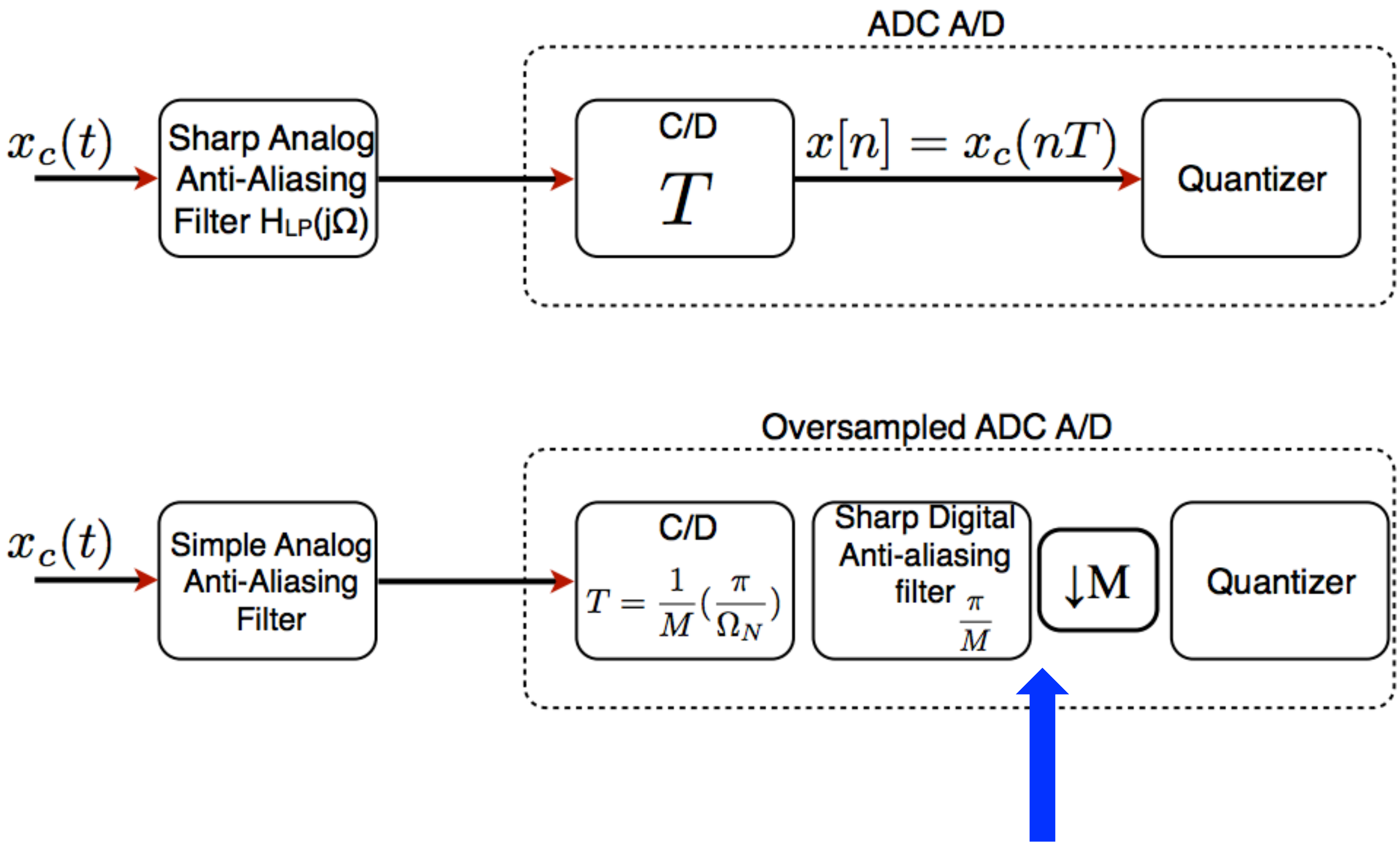
Oversampled ADC



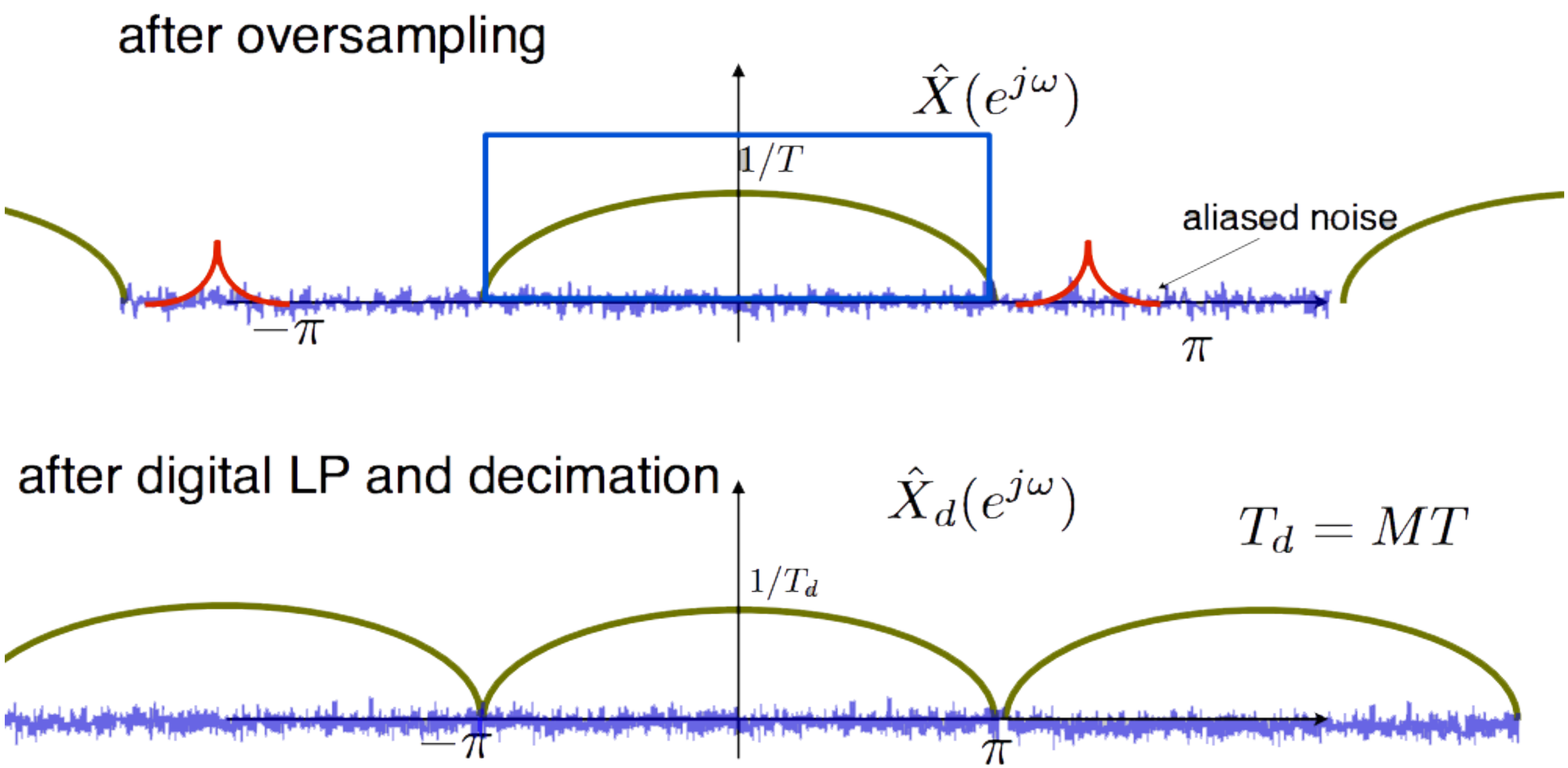
Oversampled ADC – Sharp digital filter/Downsample



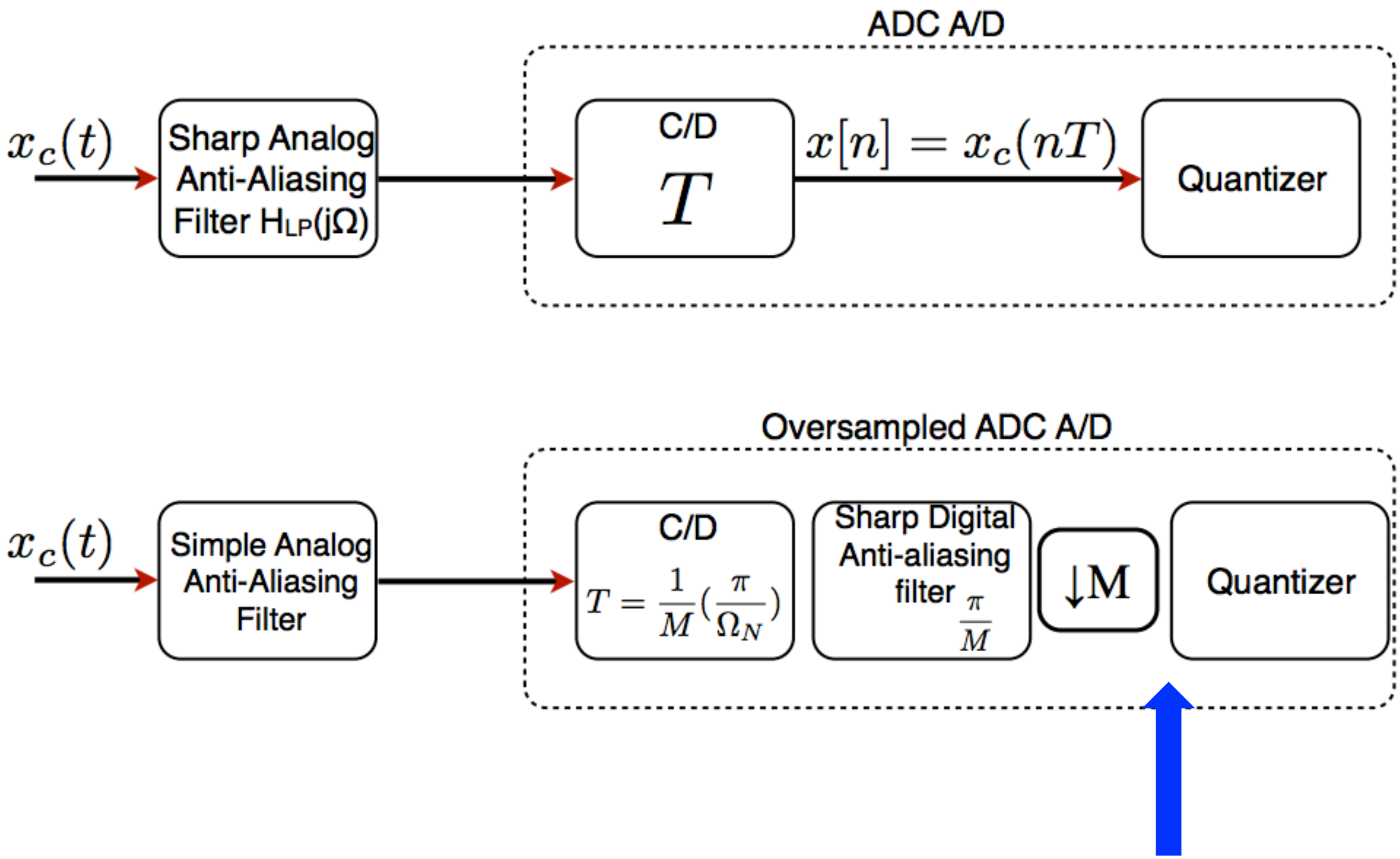
Oversampled ADC



Oversampled ADC – Sharp digital filter/Downsample

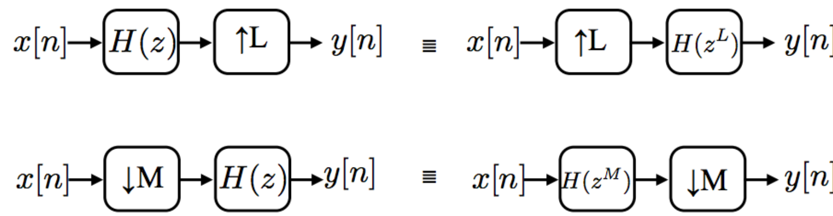


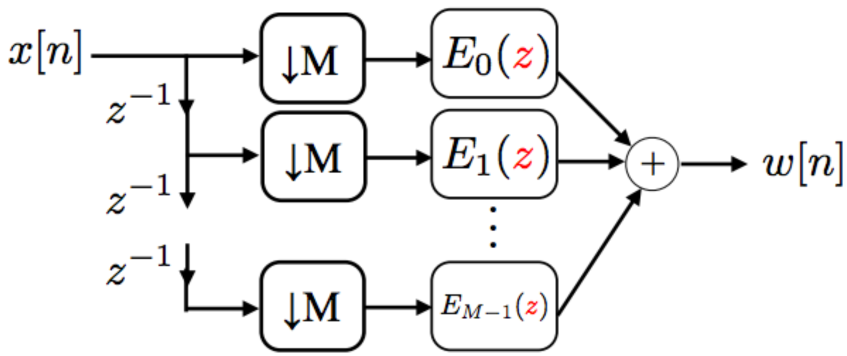
Oversampled ADC

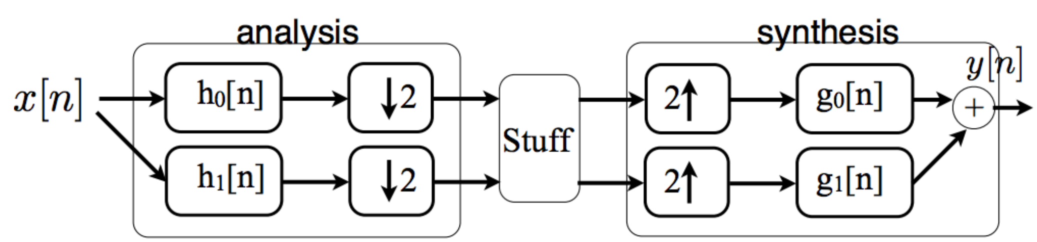


Big Ideas

- Interchanging Operations
- Polyphase Decomposition
- Multi-Rate Filter Banks







Admin

- HW 4 due Sunday