11: Multirate Systems

Multirate Systems

Multirate systems include more than one sample rate

Multirate Systems

Multirate systems include more than one sample rate

Why bother?:

- May need to change the sample rate
  e.g. Audio sample rates include 32, 44.1, 48, 96 kHz

Multirate Systems

Multirate systems include more than one sample rate

Why bother?:

- May need to change the sample rate
  e.g. Audio sample rates include 32, 44.1, 48, 96 kHz
- Can relax analog or digital filter requirements
  e.g. Audio DAC increases sample rate so that the reconstruction filter can have a more gradual cutoff

Multirate Systems

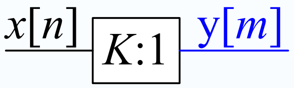
Multirate systems include more than one sample rate

Why bother?:

- May need to change the sample rate
  e.g. Audio sample rates include 32, 44.1, 48, 96 kHz
- Can relax analog or digital filter requirements
  e.g. Audio DAC increases sample rate so that the reconstruction filter can have a more gradual cutoff
- Reduce computational complexity
  FIR filter length $\propto \frac{f_s}{\Delta f}$ where Δf is width of transition band
  Lower $f_s \Rightarrow$ shorter filter + fewer samples $\Rightarrow$computation $\propto f_s^2$

Building blocks

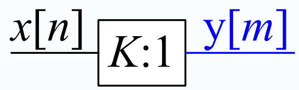
Downsample



$$y[m] = x[Km]$$

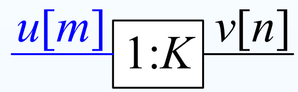
Building blocks

Downsample



$$y[m] = x[Km]$$

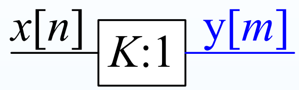
Upsample



$$v[n] = \begin{cases} u\left[\frac{n}{K}\right] & K \mid n \\ 0 & \text{else} \end{cases}$$

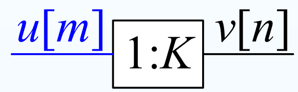
Building blocks

Downsample



$$y[m] = x[Km]$$

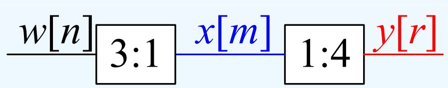
Upsample

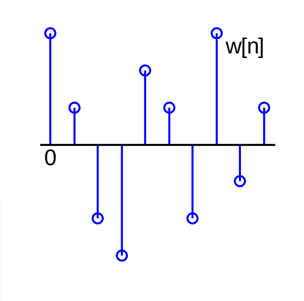


$$v[n] = \begin{cases} u\left[\frac{n}{K}\right] & K \mid n \\ 0 & \text{else} \end{cases}$$

Example:

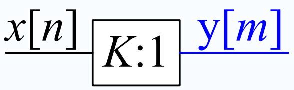
Downsample by 3 then upsample by 4





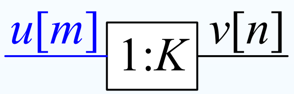
Building blocks

Downsample



$$y[m] = x[Km]$$

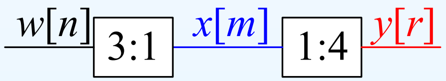
Upsample

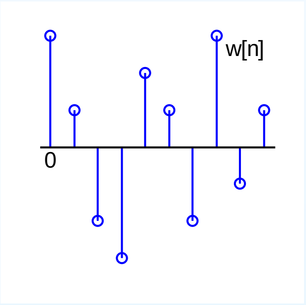


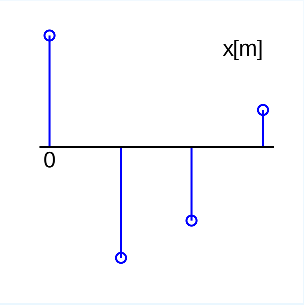
$$v[n] = \begin{cases} u\left[\frac{n}{K}\right] & K \mid n \\ 0 & \text{else} \end{cases}$$

Example:

Downsample by 3 then upsample by 4

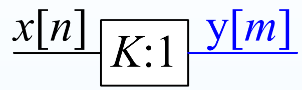






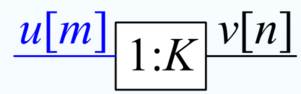
Building blocks

Downsample



$$y[m] = x[Km]$$

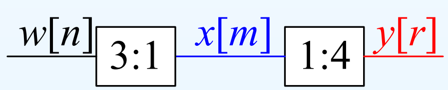
Upsample

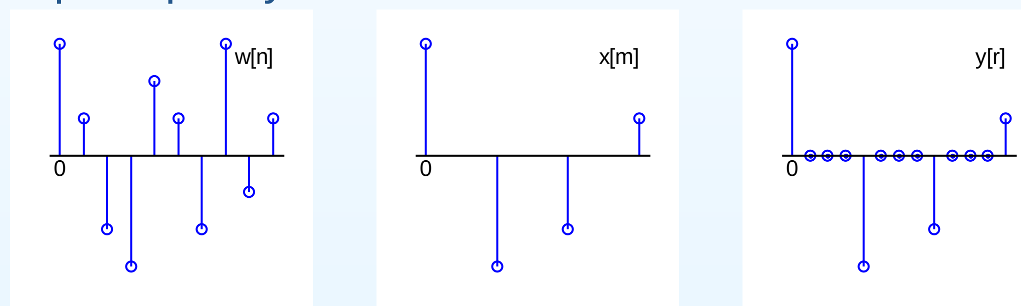


$$v[n] = \begin{cases} u\left[\frac{n}{K}\right] & K \mid n \\ 0 & \text{else} \end{cases}$$

Example:

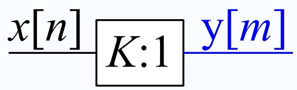
Downsample by 3 then upsample by 4





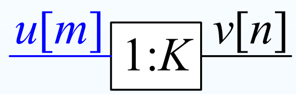
Building blocks

Downsample



$$y[m] = x[Km]$$

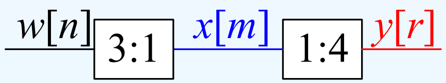
Upsample

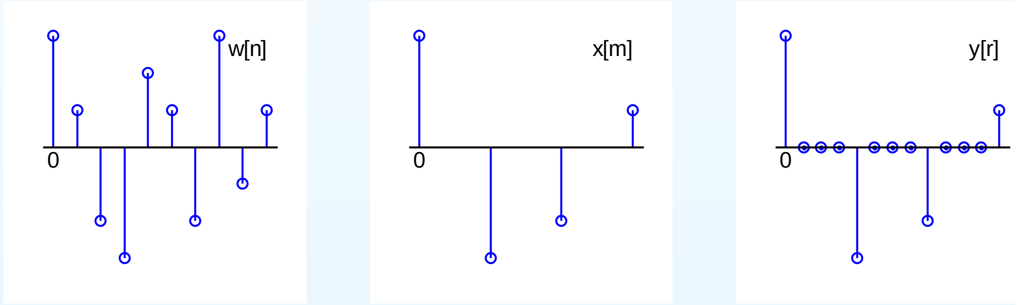


$$v[n] = \begin{cases} u\left[\frac{n}{K}\right] & K \mid n \\ 0 & \text{else} \end{cases}$$

Example:

Downsample by 3 then upsample by 4

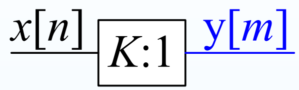




- We use different index variables (n, m, r) for different sample rates

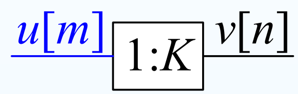
Building blocks

Downsample



$$y[m] = x[Km]$$

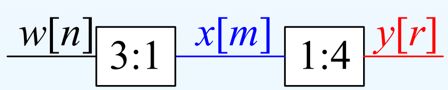
Upsample

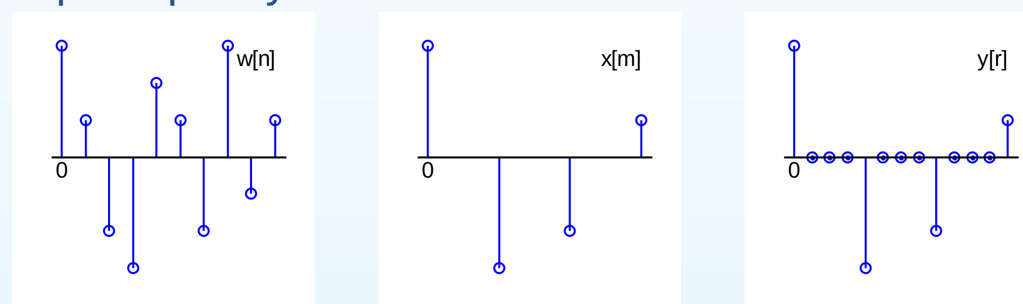


$$v[n] = \begin{cases} u\left[\frac{n}{K}\right] & K \mid n \\ 0 & \text{else} \end{cases}$$

Example:

Downsample by 3 then upsample by 4

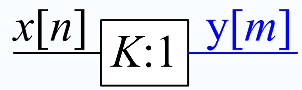




- We use different index variables (n, m, r) for different sample rates
- Use different colours for signals at different rates (sometimes)

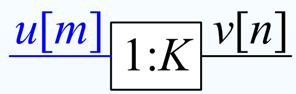
Building blocks

Downsample



$$y[m] = x[Km]$$

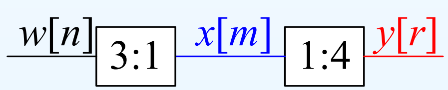
Upsample

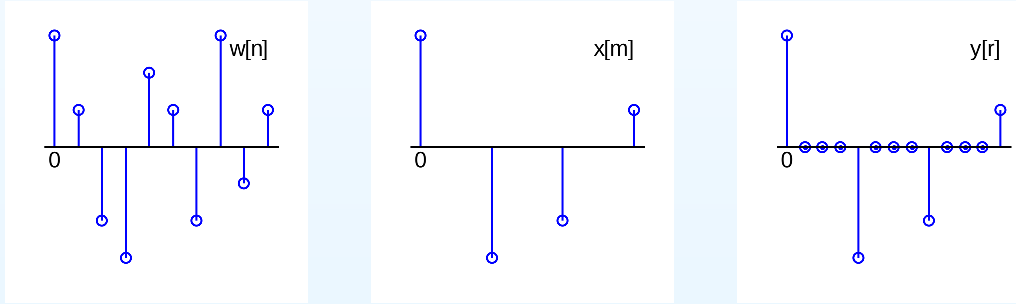


$$v[n] = \begin{cases} u\left[\frac{n}{K}\right] & K \mid n \\ 0 & \text{else} \end{cases}$$

Example:

Downsample by 3 then upsample by 4

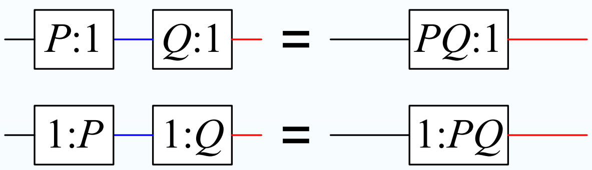




- We use different index variables (n, m, r) for different sample rates
- Use different colours for signals at different rates (sometimes)
- Synchronization: all signals have a sample at $n = 0$.

Resampling Cascades

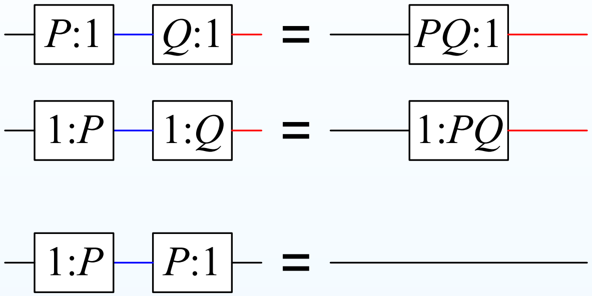
Successive downsamplers or upsamplers can be combined



Resampling Cascades

Successive downsamplers or upsamplers can be combined

Upsampling can be exactly inverted

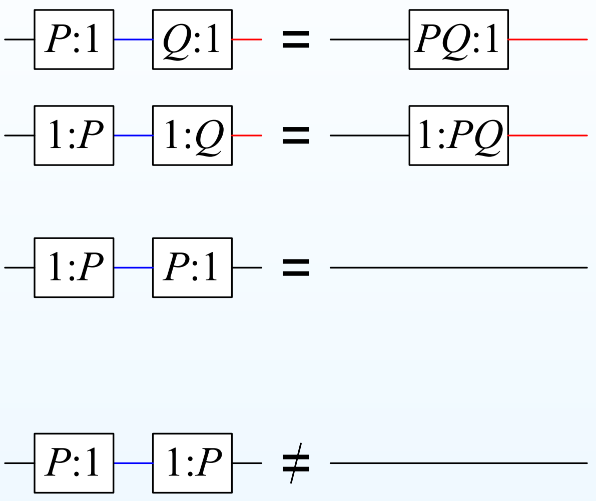


Resampling Cascades

Successive downsamplers or upsamplers can be combined

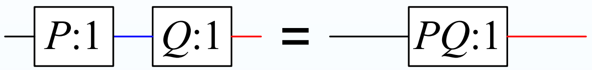
Upsampling can be exactly inverted

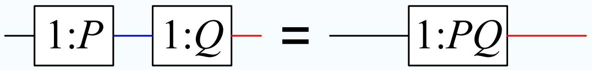
Downsampling destroys information permanently $\Rightarrow$ uninvertible



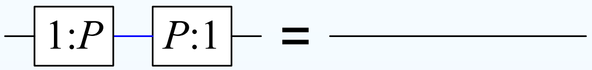
Resampling Cascades

Successive downsamplers or upsamplers can be combined

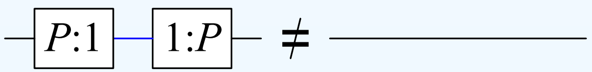




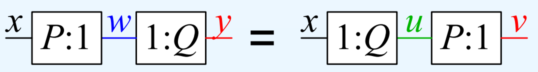
Upsampling can be exactly inverted



Downsampling destroys information permanently ⇒ uninvertible

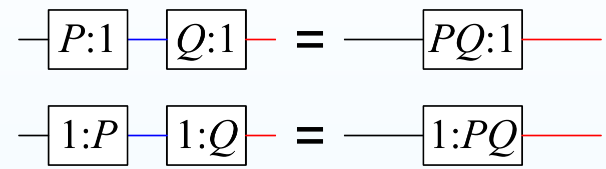


Resampling can be interchanged iff P and Q are coprime (surprising!)

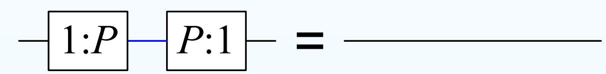


Resampling Cascades

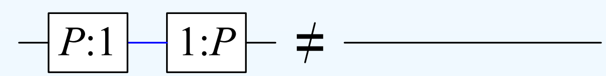
Successive downsamplers or upsamplers can be combined



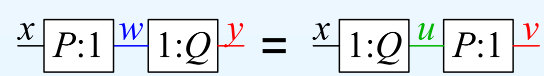
Upsampling can be exactly inverted



Downsampling destroys information permanently $\Rightarrow$ uninvertible



Resampling can be interchanged iff P and Q are coprime (surprising!)



Proof: Left side: $y[n] = w\left[\frac{1}{Q}n\right] = x\left[\frac{P}{Q}n\right]$ if $Q \mid n$ else $y[n] = 0$.

[Note: $a \mid b$ means "a divides into b exactly"]

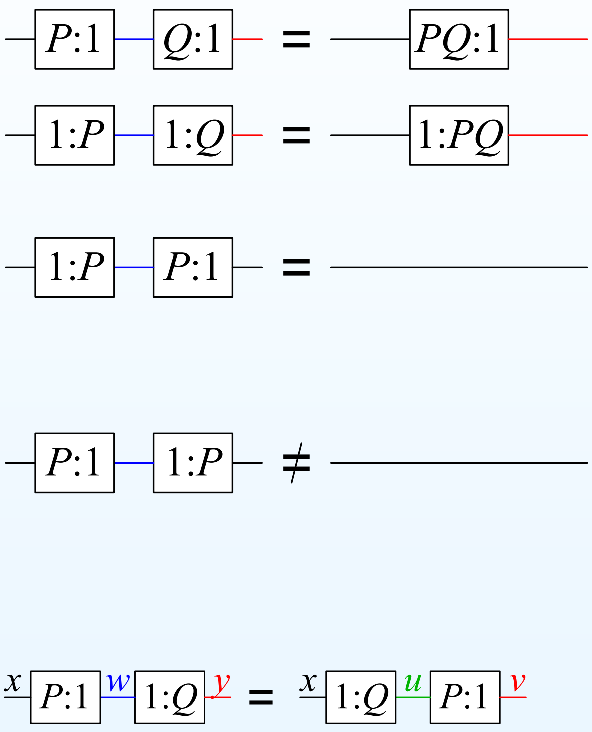
Resampling Cascades

Successive downsamplers or upsamplers can be combined

Upsampling can be exactly inverted

Downsampling destroys information permanently $\Rightarrow$ uninvertible

Resampling can be interchanged iff P and Q are coprime (surprising!)



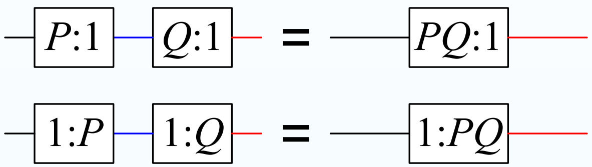
Proof: Left side: $y[n] = w\left[\frac{1}{Q}n\right] = x\left[\frac{P}{Q}n\right]$ if $Q \mid n$ else $y[n] = 0$.

Right side: $v[n] = u\left[Pn\right] = x\left[\frac{P}{Q}n\right]$ if $Q \mid Pn$.

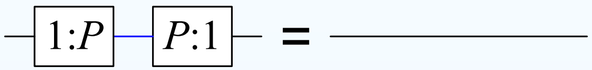
[Note: $a \mid b$ means "a divides into b exactly"]

Resampling Cascades

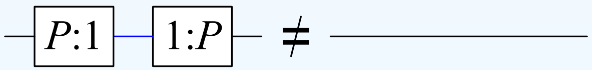
Successive downsamplers or upsamplers can be combined



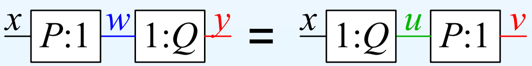
Upsampling can be exactly inverted



Downsampling destroys information permanently $\Rightarrow$ uninvertible



Resampling can be interchanged iff P and Q are coprime (surprising!)



Proof: Left side: $y[n] = w\left[\frac{1}{Q}n\right] = x\left[\frac{P}{Q}n\right]$ if $Q \mid n$ else $y[n] = 0$.

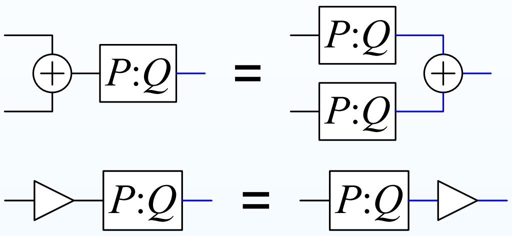
Right side: $v[n] = u\left[Pn\right] = x\left[\frac{P}{Q}n\right]$ if $Q \mid Pn$.

But $\{Q \mid Pn \;\Rightarrow\; Q \mid n\}$ iff P and Q are coprime.

[Note: $a \mid b$ means “a divides into b exactly”]

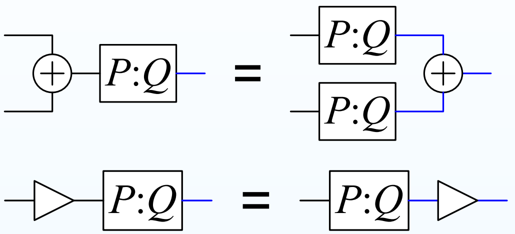
Noble Identities

Resamplers commute with addition and multiplication

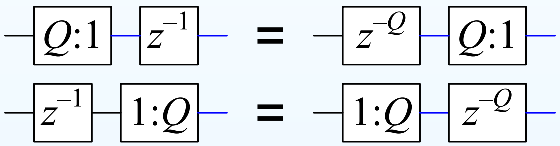


Noble Identities

Resamplers commute with addition and multiplication

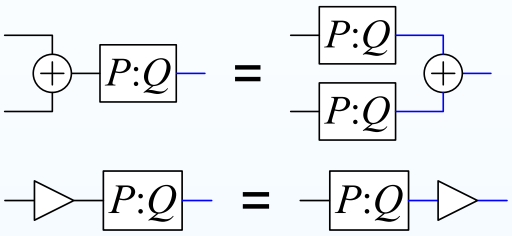


Delays must be multiplied by the resampling ratio

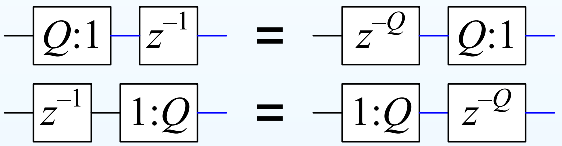


Noble Identities

Resamplers commute with addition and multiplication

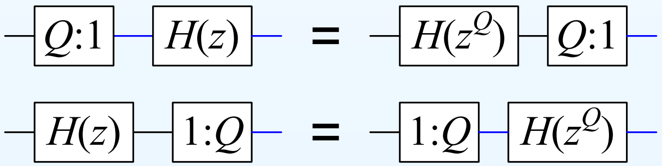


Delays must be multiplied by the resampling ratio



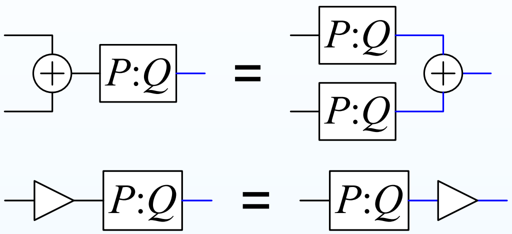
Noble identities:

Exchange resamplers and filters

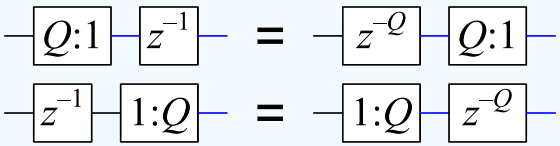


Noble Identities

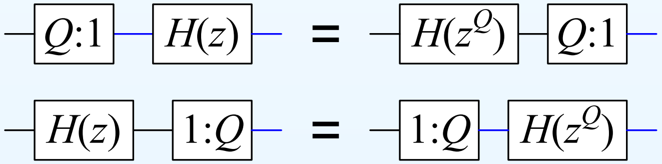
Resamplers commute with addition and multiplication



Delays must be multiplied by the resampling ratio



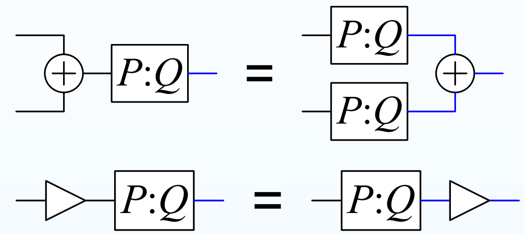
Noble identities:
Exchange resamplers and filters



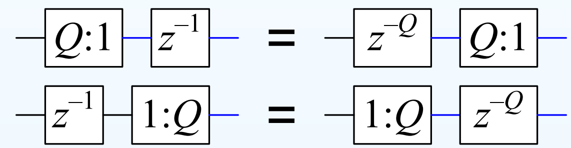
Example: $H(z) = h[0] + h[1]z^{-1} + h[2]z^{-2} + \cdots$
$H(z^3) = h[0] + h[1]z^{-3} + h[2]z^{-6} + \cdots$

Noble Identities

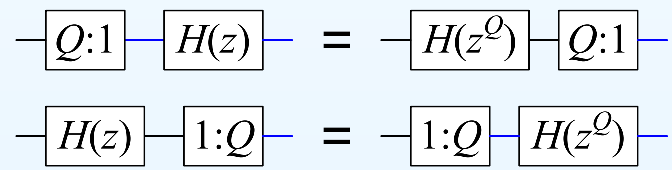
Resamplers commute with addition and multiplication



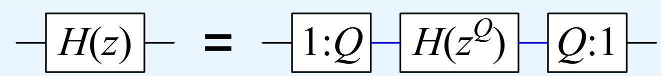
Delays must be multiplied by the resampling ratio



Noble identities:
Exchange resamplers and filters



Corrollary

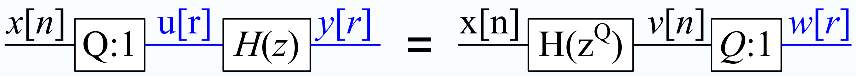


Example: $H(z) = h[0] + h[1]z^{-1} + h[2]z^{-2} + \cdots$

$$H(z^3) = h[0] + h[1]z^{-3} + h[2]z^{-6} + \cdots$$

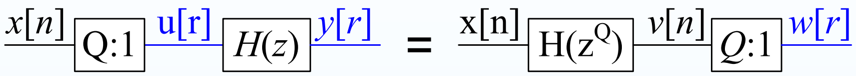
Noble Identities Proof

Define $h_Q[n]$ to be the impulse response of $H(z^Q)$.



Noble Identities Proof

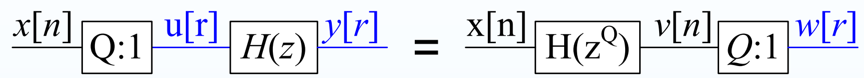
Define $h_Q[n]$ to be the impulse response of $H(z^Q)$.



Assume that $h[r]$ is of length $M + 1$ so that $h_Q[n]$ is of length $QM + 1$.

Noble Identities Proof

Define $h_Q[n]$ to be the impulse response of $H(z^Q)$.

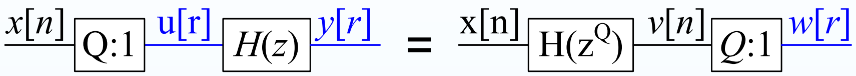


Assume that $h[r]$ is of length $M+1$ so that $h_Q[n]$ is of length $QM+1$.

We know that $h_Q[n] = 0$ except when $Q \mid n$ and that $h[r] = h_Q[Qr]$.

Noble Identities Proof

Define $h_Q[n]$ to be the impulse response of $H(z^Q)$.



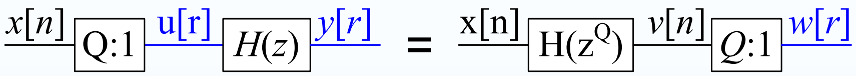
Assume that $h[r]$ is of length $M+1$ so that $h_Q[n]$ is of length $QM+1$.

We know that $h_Q[n] = 0$ except when $Q \mid n$ and that $h[r] = h_Q[Qr]$.

$$w[r] = v[Qr]$$

Noble Identities Proof

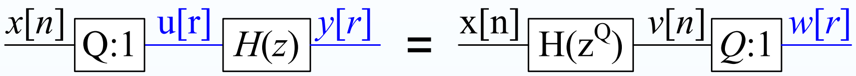
Define $h_Q[n]$ to be the impulse response of $H(z^Q)$.



Assume that $h[r]$ is of length $M+1$ so that $h_Q[n]$ is of length $QM+1$.
We know that $h_Q[n]=0$ except when $Q \mid n$ and that $h[r]=h_Q[Qr]$.

$$w[r]=v[Qr]=\sum_{s=0}^{QM} h_Q[s]x[Qr-s]$$

Noble Identities Proof



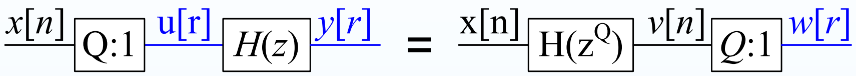
Define $h_Q[n]$ to be the impulse response of $H(z^Q)$.

Assume that $h[r]$ is of length $M+1$ so that $h_Q[n]$ is of length $QM+1$.
We know that $h_Q[n] = 0$ except when $Q \mid n$ and that $h[r] = h_Q[Qr]$.

$$\begin{aligned} w[r] = v[Qr] &= \sum_{s=0}^{QM} h_Q[s]x[Qr-s] \\ &= \sum_{m=0}^{M} h_Q[Qm]x[Qr-Qm] \end{aligned}$$

Noble Identities Proof

Define $h_Q[n]$ to be the impulse response of $H(z^Q)$.

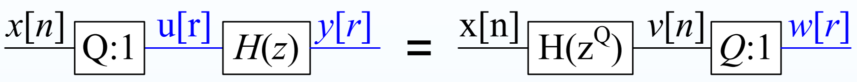


Assume that $h[r]$ is of length $M+1$ so that $h_Q[n]$ is of length $QM+1$.
We know that $h_Q[n]=0$ except when $Q \mid n$ and that $h[r]=h_Q[Qr]$.

$$w[r] = v[Qr] = \sum_{s=0}^{QM} h_Q[s]x[Qr-s]$$
$$= \sum_{m=0}^{M} h_Q[Qm]x[Qr-Qm] = \sum_{m=0}^{M} h[m]x[Q(r-m)]$$

Noble Identities Proof

Define $h_Q[n]$ to be the impulse response of $H(z^Q)$.



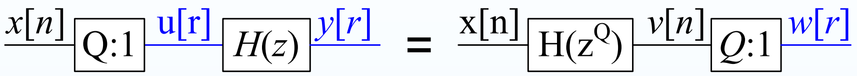
Assume that $h[r]$ is of length $M+1$ so that $h_Q[n]$ is of length $QM+1$.

We know that $h_Q[n]=0$ except when $Q \mid n$ and that $h[r]=h_Q[Qr]$.

$$\begin{aligned} w[r] = v[Qr] &= \sum_{s=0}^{QM} h_Q[s]x[Qr-s] \\ &= \sum_{m=0}^{M} h_Q[Qm]x[Qr-Qm] = \sum_{m=0}^{M} h[m]x[Q(r-m)] \\ &= \sum_{m=0}^{M} h[m]u[r-m] \end{aligned}$$

Noble Identities Proof

Define $h_Q[n]$ to be the impulse response of $H(z^Q)$.



Assume that $h[r]$ is of length $M+1$ so that $h_Q[n]$ is of length $QM+1$.

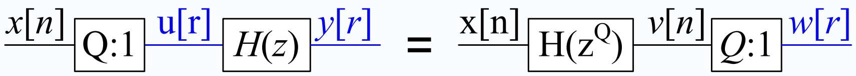
We know that $h_Q[n]=0$ except when $Q \mid n$ and that $h[r]=h_Q[Qr]$.

$$\begin{aligned} w[r] &= v[Qr] = \sum_{s=0}^{QM} h_Q[s]x[Qr-s] \\ &= \sum_{m=0}^{M} h_Q[Qm]x[Qr-Qm] = \sum_{m=0}^{M} h[m]x[Q(r-m)] \\ &= \sum_{m=0}^{M} h[m]u[r-m] = y[r] \end{aligned}$$

☺

Noble Identities Proof

Define $h_Q[n]$ to be the impulse response of $H(z^Q)$.

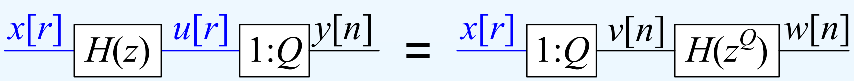


Assume that $h[r]$ is of length $M+1$ so that $h_Q[n]$ is of length $QM+1$.
We know that $h_Q[n]=0$ except when $Q \mid n$ and that $h[r]=h_Q[Qr]$.

$$\begin{aligned} w[r] = v[Qr] &= \sum_{s=0}^{QM} h_Q[s]x[Qr-s] \\ &= \sum_{m=0}^{M} h_Q[Qm]x[Qr-Qm] = \sum_{m=0}^{M} h[m]x[Q(r-m)] \\ &= \sum_{m=0}^{M} h[m]u[r-m] = y[r] \end{aligned}$$

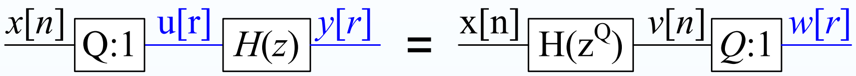
☺

Upsampled Noble Identity:



Noble Identities Proof

Define $h_Q[n]$ to be the impulse response of $H(z^Q)$.

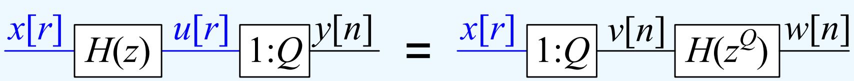


Assume that $h[r]$ is of length $M+1$ so that $h_Q[n]$ is of length $QM+1$.
We know that $h_Q[n]=0$ except when $Q \mid n$ and that $h[r]=h_Q[Qr]$.

$$\begin{aligned} w[r] = v[Qr] &= \sum_{s=0}^{QM} h_Q[s]x[Qr-s] \\ &= \sum_{m=0}^{M} h_Q[Qm]x[Qr-Qm] = \sum_{m=0}^{M} h[m]x[Q(r-m)] \\ &= \sum_{m=0}^{M} h[m]u[r-m] = y[r] \end{aligned}$$

☺

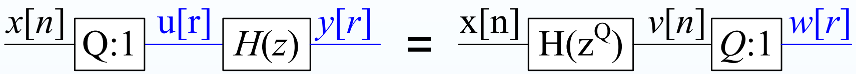
Upsampled Noble Identity:



We know that $v[n]=0$ except when $Q \mid n$ and that $v[Qr]=x[r]$.

Noble Identities Proof

Define $h_Q[n]$ to be the impulse response of $H(z^Q)$.

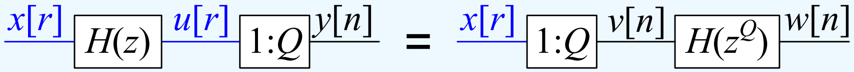


Assume that $h[r]$ is of length $M+1$ so that $h_Q[n]$ is of length $QM+1$.
We know that $h_Q[n]=0$ except when $Q \mid n$ and that $h[r]=h_Q[Qr]$.

$$\begin{aligned} w[r] = v[Qr] &= \sum_{s=0}^{QM} h_Q[s]x[Qr-s] \\ &= \sum_{m=0}^{M} h_Q[Qm]x[Qr-Qm] = \sum_{m=0}^{M} h[m]x[Q(r-m)] \\ &= \sum_{m=0}^{M} h[m]u[r-m] = y[r] \end{aligned}$$

☺

Upsampled Noble Identity:

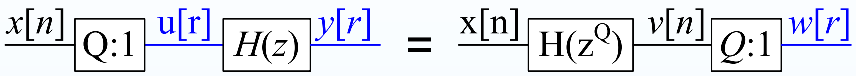


We know that $v[n]=0$ except when $Q \mid n$ and that $v[Qr]=x[r]$.

$$w[n] = \sum_{s=0}^{QM} h_Q[s]v[n-s]$$

Noble Identities Proof

Define $h_Q[n]$ to be the impulse response of $H(z^Q)$.

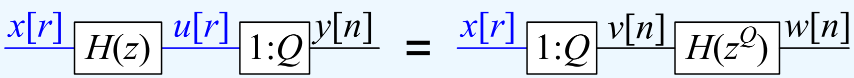


Assume that $h[r]$ is of length $M+1$ so that $h_Q[n]$ is of length $QM+1$.
We know that $h_Q[n]=0$ except when $Q \mid n$ and that $h[r]=h_Q[Qr]$.

$$\begin{aligned} w[r] &= v[Qr] = \sum_{s=0}^{QM} h_Q[s]x[Qr-s] \\ &= \sum_{m=0}^{M} h_Q[Qm]x[Qr-Qm] = \sum_{m=0}^{M} h[m]x[Q(r-m)] \\ &= \sum_{m=0}^{M} h[m]u[r-m] = y[r] \end{aligned}$$

☺

Upsampled Noble Identity:

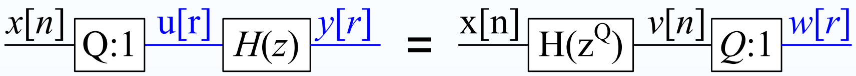


We know that $v[n]=0$ except when $Q \mid n$ and that $v[Qr]=x[r]$.

$$w[n] = \sum_{s=0}^{QM} h_Q[s]v[n-s] = \sum_{m=0}^{M} h_Q[Qm]v[n-Qm]$$

Noble Identities Proof

Define $h_Q[n]$ to be the impulse response of $H(z^Q)$.

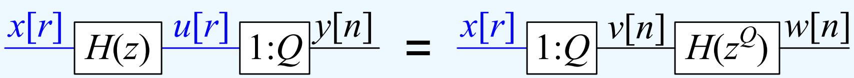


Assume that $h[r]$ is of length $M+1$ so that $h_Q[n]$ is of length $QM+1$.
We know that $h_Q[n]=0$ except when $Q \mid n$ and that $h[r]=h_Q[Qr]$.

$$\begin{aligned} w[r] = v[Qr] &= \sum_{s=0}^{QM} h_Q[s]x[Qr-s] \\ &= \sum_{m=0}^{M} h_Q[Qm]x[Qr-Qm] = \sum_{m=0}^{M} h[m]x[Q(r-m)] \\ &= \sum_{m=0}^{M} h[m]u[r-m] = y[r] \end{aligned}$$

☺

Upsampled Noble Identity:

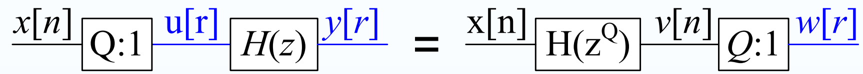


We know that $v[n]=0$ except when $Q \mid n$ and that $v[Qr]=x[r]$.

$$\begin{aligned} w[n] &= \sum_{s=0}^{QM} h_Q[s]v[n-s] = \sum_{m=0}^{M} h_Q[Qm]v[n-Qm] \\ &= \sum_{m=0}^{M} h[m]v[n-Qm] \end{aligned}$$

Noble Identities Proof

Define $h_Q[n]$ to be the impulse response of $H(z^Q)$.

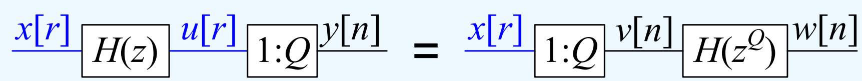


Assume that $h[r]$ is of length $M+1$ so that $h_Q[n]$ is of length $QM+1$.
We know that $h_Q[n]=0$ except when $Q \mid n$ and that $h[r]=h_Q[Qr]$.

$$\begin{aligned} w[r] = v[Qr] &= \sum_{s=0}^{QM} h_Q[s]x[Qr-s] \\ &= \sum_{m=0}^{M} h_Q[Qm]x[Qr-Qm] = \sum_{m=0}^{M} h[m]x[Q(r-m)] \\ &= \sum_{m=0}^{M} h[m]u[r-m] = y[r] \end{aligned}$$

☺

Upsampled Noble Identity:



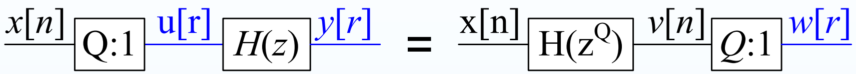
We know that $v[n]=0$ except when $Q \mid n$ and that $v[Qr]=x[r]$.

$$\begin{aligned} w[n] &= \sum_{s=0}^{QM} h_Q[s]v[n-s] = \sum_{m=0}^{M} h_Q[Qm]v[n-Qm] \\ &= \sum_{m=0}^{M} h[m]v[n-Qm] \end{aligned}$$

If $Q \nmid n$, then $v[n-Qm]=0\ \forall m$ so $w[n]=0=y[n]$

Noble Identities Proof

Define $h_Q[n]$ to be the impulse response of $H(z^Q)$.

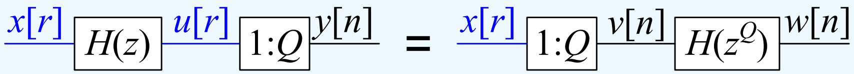


Assume that $h[r]$ is of length $M+1$ so that $h_Q[n]$ is of length $QM+1$.
We know that $h_Q[n]=0$ except when $Q \mid n$ and that $h[r]=h_Q[Qr]$.

$$\begin{aligned} w[r] = v[Qr] &= \sum_{s=0}^{QM} h_Q[s]x[Qr-s] \\ &= \sum_{m=0}^{M} h_Q[Qm]x[Qr-Qm] = \sum_{m=0}^{M} h[m]x[Q(r-m)] \\ &= \sum_{m=0}^{M} h[m]u[r-m] = y[r] \end{aligned}$$

☺

Upsampled Noble Identity:



We know that $v[n]=0$ except when $Q \mid n$ and that $v[Qr]=x[r]$.

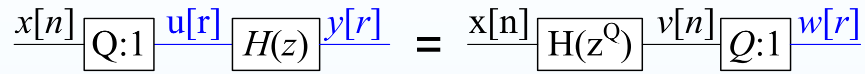
$$\begin{aligned} w[n] &= \sum_{s=0}^{QM} h_Q[s]v[n-s] = \sum_{m=0}^{M} h_Q[Qm]v[n-Qm] \\ &= \sum_{m=0}^{M} h[m]v[n-Qm] \end{aligned}$$

If $Q \nmid n$, then $v[n-Qm]=0 \; \forall m$ so $w[n]=0=y[n]$

If $Q \mid n=Qr$, then $w[Qr]=\sum_{m=0}^{M} h[m]v[Qr-Qm]$

Noble Identities Proof

Define $h_Q[n]$ to be the impulse response of $H(z^Q)$.

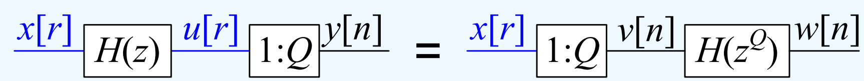


Assume that $h[r]$ is of length $M+1$ so that $h_Q[n]$ is of length $QM+1$.
We know that $h_Q[n]=0$ except when $Q \mid n$ and that $h[r]=h_Q[Qr]$.

$$\begin{aligned} w[r] = v[Qr] &= \sum_{s=0}^{QM} h_Q[s]x[Qr-s] \\ &= \sum_{m=0}^{M} h_Q[Qm]x[Qr-Qm] = \sum_{m=0}^{M} h[m]x[Q(r-m)] \\ &= \sum_{m=0}^{M} h[m]u[r-m] = y[r] \end{aligned}$$

☺

Upsampled Noble Identity:



We know that $v[n]=0$ except when $Q \mid n$ and that $v[Qr]=x[r]$.

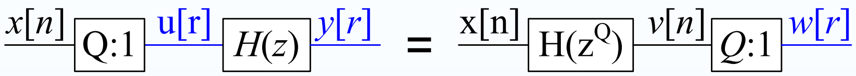
$$\begin{aligned} w[n] &= \sum_{s=0}^{QM} h_Q[s]v[n-s] = \sum_{m=0}^{M} h_Q[Qm]v[n-Qm] \\ &= \sum_{m=0}^{M} h[m]v[n-Qm] \end{aligned}$$

If $Q \nmid n$, then $v[n-Qm]=0\ \forall m$ so $w[n]=0=y[n]$

If $Q \mid n = Qr$, then $w[Qr] = \sum_{m=0}^{M} h[m]v[Qr-Qm]$
$= \sum_{m=0}^{M} h[m]x[r-m] = u[r]$

Noble Identities Proof

Define $h_Q[n]$ to be the impulse response of $H(z^Q)$.

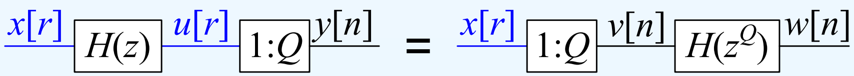


Assume that $h[r]$ is of length $M+1$ so that $h_Q[n]$ is of length $QM+1$.
We know that $h_Q[n]=0$ except when $Q \mid n$ and that $h[r]=h_Q[Qr]$.

$$\begin{aligned} w[r] = v[Qr] &= \sum_{s=0}^{QM} h_Q[s]x[Qr-s] \\ &= \sum_{m=0}^{M} h_Q[Qm]x[Qr-Qm] = \sum_{m=0}^{M} h[m]x[Q(r-m)] \\ &= \sum_{m=0}^{M} h[m]u[r-m] = y[r] \end{aligned}$$

☺

Upsampled Noble Identity:



We know that $v[n]=0$ except when $Q \mid n$ and that $v[Qr]=x[r]$.

$$\begin{aligned} w[n] &= \sum_{s=0}^{QM} h_Q[s]v[n-s] = \sum_{m=0}^{M} h_Q[Qm]v[n-Qm] \\ &= \sum_{m=0}^{M} h[m]v[n-Qm] \end{aligned}$$

If $Q \nmid n$, then $v[n-Qm]=0\ \forall m$ so $w[n]=0=y[n]$

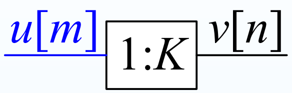
If $Q \mid n = Qr$, then $w[Qr] = \sum_{m=0}^{M} h[m]v[Qr-Qm]$

$$= \sum_{m=0}^{M} h[m]x[r-m] = u[r] = y[Qr]$$

☺

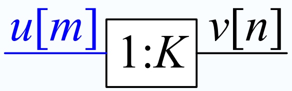
Upsampled z-transform

$$V(z) = \sum_n v[n] z^{-n}$$



Upsampled z-transform

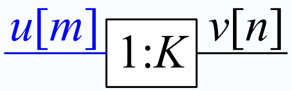
$$V(z) = \sum_n v[n] z^{-n} = \sum_{n \text{ s.t. } K|n} u[\tfrac{n}{K}] z^{-n}$$



Upsampled z-transform

$$V(z) = \sum_n v[n] z^{-n} = \sum_{n \text{ s.t. } K|n} u[\tfrac{n}{K}] z^{-n}$$

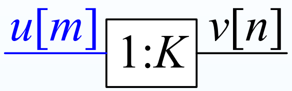
$$= \sum_m u[m] z^{-Km}$$

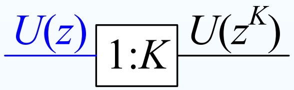


Upsampled z-transform

$$V(z) = \sum_n v[n] z^{-n} = \sum_{n \text{ s.t. } K|n} u[\tfrac{n}{K}] z^{-n}$$

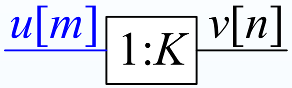
$$= \sum_m u[m] z^{-Km} = U(z^K)$$

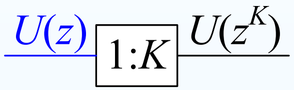




Upsampled z-transform

$$V(z) = \sum_n v[n] z^{-n} = \sum_{n \text{ s.t. } K|n} u[\tfrac{n}{K}] z^{-n}$$
$$= \sum_m u[m] z^{-Km} = U(z^K)$$

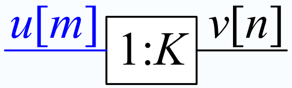


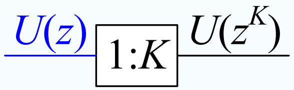


Spectrum: $V(e^{j\omega}) = U(e^{jK\omega})$

Upsampled z-transform

$$V(z) = \sum_n v[n]z^{-n} = \sum_{n \text{ s.t. } K|n} u[\tfrac{n}{K}]z^{-n}$$
$$= \sum_m u[m]z^{-Km} = U(z^K)$$



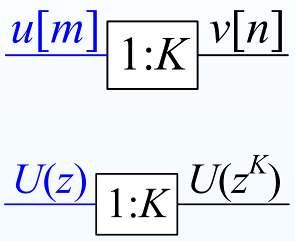


Spectrum: $V(e^{j\omega}) = U(e^{jK\omega})$

Spectrum is horizontally shrunk and replicated K times.

Upsampled z-transform

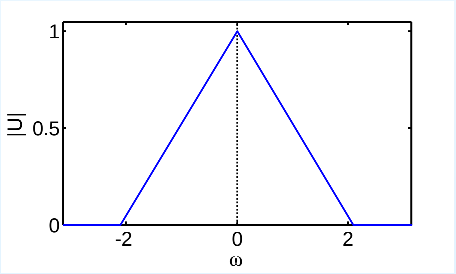
$$V(z) = \sum_n v[n] z^{-n} = \sum_{n \text{ s.t. } K|n} u[\frac{n}{K}] z^{-n}$$
$$= \sum_m u[m] z^{-Km} = U(z^K)$$



Spectrum: $V(e^{j\omega}) = U(e^{jK\omega})$

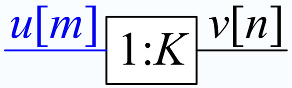
Spectrum is horizontally shrunk and replicated K times.

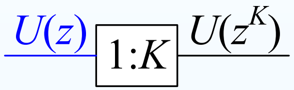
Example:



Upsampled z-transform

$$V(z) = \sum_n v[n] z^{-n} = \sum_{n \text{ s.t. } K|n} u[\tfrac{n}{K}] z^{-n}$$
$$= \sum_m u[m] z^{-Km} = U(z^K)$$



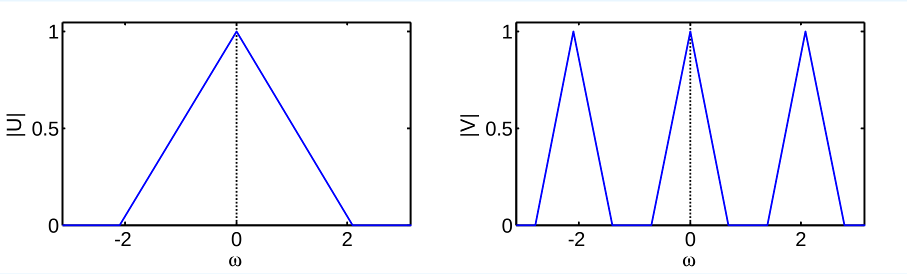


Spectrum: $V(e^{j\omega}) = U(e^{jK\omega})$

Spectrum is horizontally shrunk and replicated K times.

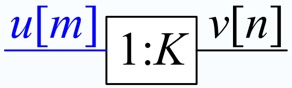
Example:

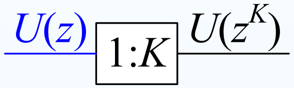
$K = 3$: three images of the original spectrum in all.



Upsampled z-transform

$$V(z) = \sum_n v[n]z^{-n} = \sum_{n \text{ s.t. } K|n} u[\frac{n}{K}]z^{-n}$$
$$= \sum_m u[m]z^{-Km} = U(z^K)$$





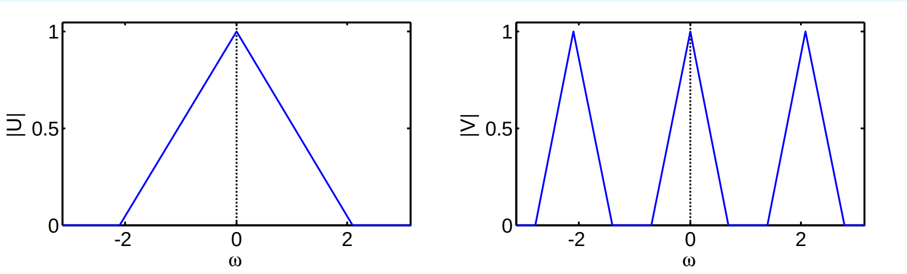
Spectrum: $V(e^{j\omega}) = U(e^{jK\omega})$

Spectrum is horizontally shrunk and replicated K times.

Total energy unchanged; power (= energy/sample) multiplied by $\frac{1}{K}$

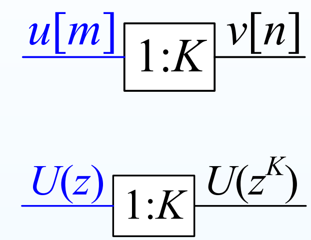
Example:

$K = 3$: three images of the original spectrum in all.



Upsampled z-transform

$$V(z) = \sum_n v[n] z^{-n} = \sum_{n \text{ s.t. } K|n} u[\frac{n}{K}] z^{-n}$$
$$= \sum_m u[m] z^{-Km} = U(z^K)$$



Spectrum: $V(e^{j\omega}) = U(e^{jK\omega})$

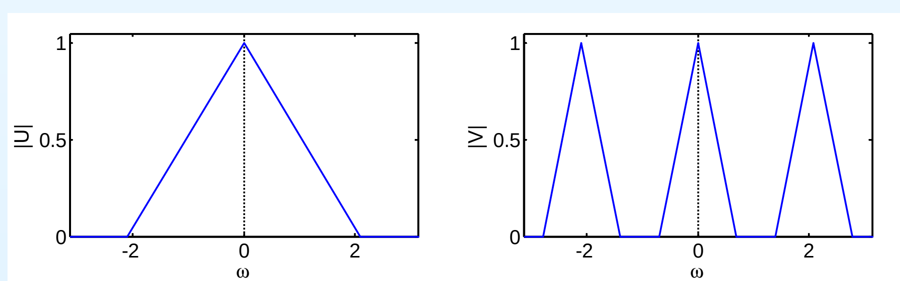
Spectrum is horizontally shrunk and replicated K times.

Total energy unchanged; power (= energy/sample) multiplied by $\frac{1}{K}$

Example:

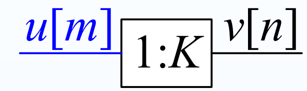
$K = 3$: three images of the original spectrum in all.

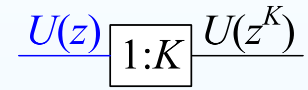
Energy unchanged: $\frac{1}{2\pi}\int \left|U(e^{j\omega})\right|^2 d\omega = \frac{1}{2\pi}\int \left|V(e^{j\omega})\right|^2 d\omega$



Upsampled z-transform

$$V(z) = \sum_n v[n]z^{-n} = \sum_{n \text{ s.t. } K|n} u[\tfrac{n}{K}]z^{-n}$$
$$= \sum_m u[m]z^{-Km} = U(z^K)$$





Spectrum: $V(e^{j\omega}) = U(e^{jK\omega})$

Spectrum is horizontally shrunk and replicated K times.

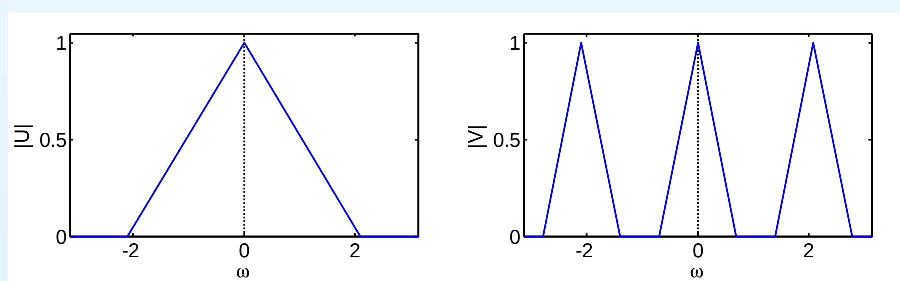
Total energy unchanged; power (= energy/sample) multiplied by $\frac{1}{K}$

Upsampling normally followed by a LP filter to remove images.

Example:

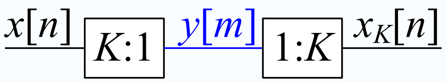
$K = 3$: three images of the original spectrum in all.

Energy unchanged: $\frac{1}{2\pi}\int \left|U(e^{j\omega})\right|^2 d\omega = \frac{1}{2\pi}\int \left|V(e^{j\omega})\right|^2 d\omega$



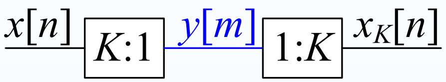
Downsampled z-transform

Define $c_K[n] = \delta_{K|n}[n]$



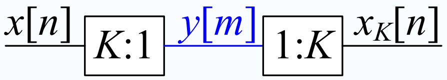
Downsampled z-transform

Define $c_K[n] = \delta_{K|n}[n] = \frac{1}{K}\sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}}$



Downsampled z-transform

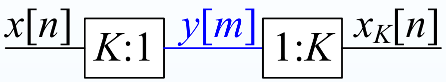
Define $c_K[n] = \delta_{K|n}[n] = \frac{1}{K}\sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}}$



Now define $x_K[n] = \begin{cases} x[n] & K \mid n \\ 0 & K \nmid n \end{cases}$

Downsampled z-transform

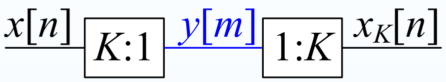
Define $c_K[n] = \delta_{K|n}[n] = \frac{1}{K}\sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}}$



Now define $x_K[n] = \begin{cases} x[n] & K \mid n \\ 0 & K \nmid n \end{cases} = c_K[n]x[n]$

Downsampled z-transform

Define $c_K[n] = \delta_{K|n}[n] = \frac{1}{K}\sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}}$

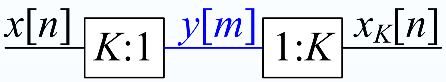


Now define $x_K[n] = \begin{cases} x[n] & K \mid n \\ 0 & K \nmid n \end{cases} = c_K[n]x[n]$

$X_K(z) = \sum_n x_K[n]z^{-n}$

Downsampled z-transform

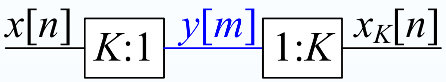
Define $c_K[n] = \delta_{K|n}[n] = \frac{1}{K}\sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}}$



Now define $x_K[n] = \begin{cases} x[n] & K \mid n \\ 0 & K \nmid n \end{cases} = c_K[n]x[n]$

$$X_K(z) = \sum_n x_K[n]z^{-n} = \frac{1}{K}\sum_n \sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}} x[n]z^{-n}$$

Downsampled z-transform



Define $c_K[n] = \delta_{K|n}[n] = \frac{1}{K}\sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}}$

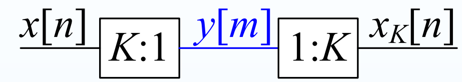
Now define $x_K[n] = \begin{cases} x[n] & K \mid n \\ 0 & K \nmid n \end{cases} = c_K[n]x[n]$

$$X_K(z) = \sum_n x_K[n]z^{-n} = \frac{1}{K}\sum_n \sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}} x[n]z^{-n}$$

$$= \frac{1}{K}\sum_{k=0}^{K-1}\sum_n x[n]\left(e^{\frac{-j2\pi k}{K}}z\right)^{-n}$$

Downsampled z-transform

Define $c_K[n] = \delta_{K|n}[n] = \frac{1}{K}\sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}}$



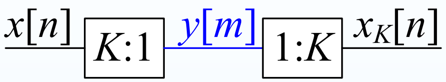
Now define $x_K[n] = \begin{cases} x[n] & K \mid n \\ 0 & K \nmid n \end{cases} = c_K[n]x[n]$

$$X_K(z) = \sum_n x_K[n]z^{-n} = \frac{1}{K}\sum_n \sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}} x[n]z^{-n}$$

$$= \frac{1}{K}\sum_{k=0}^{K-1}\sum_n x[n]\left(e^{\frac{-j2\pi k}{K}}z\right)^{-n} = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{-j2\pi k}{K}}z)$$

Downsampled z-transform

Define $c_K[n] = \delta_{K|n}[n] = \frac{1}{K}\sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}}$



Now define $x_K[n] = \begin{cases} x[n] & K \mid n \\ 0 & K \nmid n \end{cases} = c_K[n]x[n]$

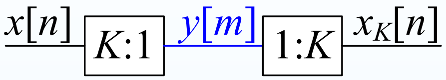
$$X_K(z) = \sum_n x_K[n]z^{-n} = \frac{1}{K}\sum_n \sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}} x[n]z^{-n}$$

$$= \frac{1}{K}\sum_{k=0}^{K-1}\sum_n x[n]\left(e^{\frac{-j2\pi k}{K}}z\right)^{-n} = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{-j2\pi k}{K}}z)$$

From previous slide:

$$X_K(z) = Y(z^K)$$

Downsampled z-transform



Define $c_K[n] = \delta_{K|n}[n] = \frac{1}{K}\sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}}$

Now define $x_K[n] = \begin{cases} x[n] & K \mid n \\ 0 & K \nmid n \end{cases} = c_K[n]x[n]$

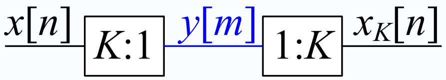
$$X_K(z) = \sum_n x_K[n]z^{-n} = \frac{1}{K}\sum_n \sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}} x[n]z^{-n}$$
$$= \frac{1}{K}\sum_{k=0}^{K-1}\sum_n x[n]\left(e^{\frac{-j2\pi k}{K}}z\right)^{-n} = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{-j2\pi k}{K}}z)$$

From previous slide:

$$X_K(z) = Y(z^K)$$
$$\Rightarrow Y(z) = X_K(z^{\frac{1}{K}})$$

Downsampled z-transform

Define $c_K[n] = \delta_{K|n}[n] = \frac{1}{K}\sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}}$

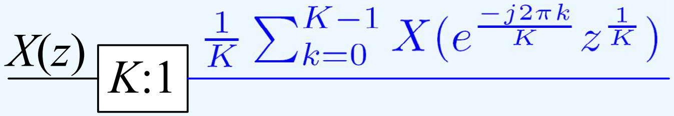


Now define $x_K[n] = \begin{cases} x[n] & K \mid n \\ 0 & K \nmid n \end{cases} = c_K[n]x[n]$

$$X_K(z) = \sum_n x_K[n]z^{-n} = \frac{1}{K}\sum_n \sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}} x[n]z^{-n}$$

$$= \frac{1}{K}\sum_{k=0}^{K-1}\sum_n x[n]\left(e^{\frac{-j2\pi k}{K}}z\right)^{-n} = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{-j2\pi k}{K}}z)$$

From previous slide:

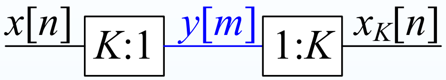


$$X_K(z) = Y(z^K)$$

$$\Rightarrow Y(z) = X_K(z^{\frac{1}{K}}) = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{-j2\pi k}{K}}z^{\frac{1}{K}})$$

Downsampled z-transform

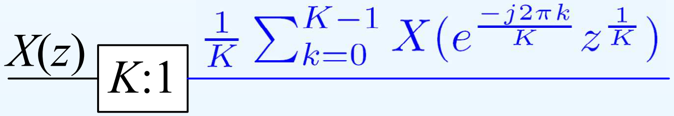
Define $c_K[n] = \delta_{K|n}[n] = \frac{1}{K}\sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}}$



Now define $x_K[n] = \begin{cases} x[n] & K \mid n \\ 0 & K \nmid n \end{cases} = c_K[n]x[n]$

$$X_K(z) = \sum_n x_K[n]z^{-n} = \frac{1}{K}\sum_n \sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}} x[n]z^{-n}$$
$$= \frac{1}{K}\sum_{k=0}^{K-1}\sum_n x[n]\left(e^{\frac{-j2\pi k}{K}}z\right)^{-n} = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{-j2\pi k}{K}}z)$$

From previous slide:

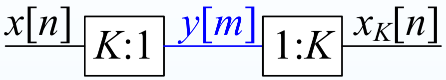


$$X_K(z) = Y(z^K)$$
$$\Rightarrow Y(z) = X_K(z^{\frac{1}{K}}) = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{-j2\pi k}{K}} z^{\frac{1}{K}})$$

Frequency Spectrum:

$$Y(e^{j\omega}) = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{j(\omega-2\pi k)}{K}})$$

Downsampled z-transform



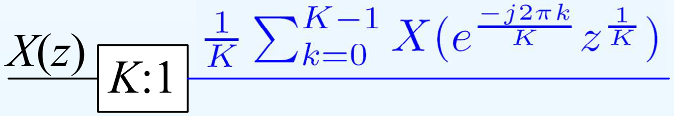
Define $c_K[n] = \delta_{K|n}[n] = \frac{1}{K}\sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}}$

Now define $x_K[n] = \begin{cases} x[n] & K \mid n \\ 0 & K \nmid n \end{cases} = c_K[n]x[n]$

$$X_K(z) = \sum_n x_K[n]z^{-n} = \frac{1}{K}\sum_n \sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}} x[n]z^{-n}$$

$$= \frac{1}{K}\sum_{k=0}^{K-1}\sum_n x[n]\left(e^{\frac{-j2\pi k}{K}}z\right)^{-n} = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{-j2\pi k}{K}}z)$$

From previous slide:



$$X_K(z) = Y(z^K)$$

$$\Rightarrow Y(z) = X_K(z^{\frac{1}{K}}) = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{-j2\pi k}{K}}z^{\frac{1}{K}})$$

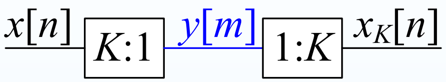
Frequency Spectrum:

$$Y(e^{j\omega}) = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{j(\omega-2\pi k)}{K}})$$

$$= \frac{1}{K}\left(X(e^{\frac{j\omega}{K}}) + X(e^{\frac{j\omega}{K}-\frac{2\pi}{K}}) + X(e^{\frac{j\omega}{K}-\frac{4\pi}{K}}) + \cdots\right)$$

Downsampled z-transform

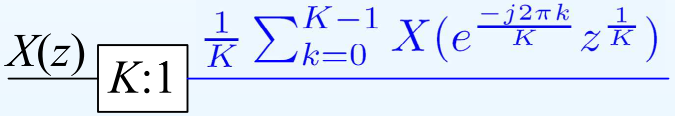
Define $c_K[n] = \delta_{K|n}[n] = \frac{1}{K}\sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}}$



Now define $x_K[n] = \begin{cases} x[n] & K \mid n \\ 0 & K \nmid n \end{cases} = c_K[n]x[n]$

$$X_K(z) = \sum_n x_K[n]z^{-n} = \frac{1}{K}\sum_n \sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}} x[n]z^{-n}$$
$$= \frac{1}{K}\sum_{k=0}^{K-1}\sum_n x[n]\left(e^{\frac{-j2\pi k}{K}}z\right)^{-n} = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{-j2\pi k}{K}}z)$$

From previous slide:



$$X_K(z) = Y(z^K)$$
$$\Rightarrow Y(z) = X_K(z^{\frac{1}{K}}) = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{-j2\pi k}{K}}z^{\frac{1}{K}})$$

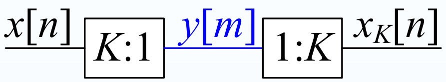
Frequency Spectrum:

$$Y(e^{j\omega}) = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{j(\omega-2\pi k)}{K}})$$
$$= \frac{1}{K}\left(X(e^{\frac{j\omega}{K}}) + X(e^{\frac{j\omega}{K}-\frac{2\pi}{K}}) + X(e^{\frac{j\omega}{K}-\frac{4\pi}{K}}) + \cdots\right)$$

Average of K aliased versions, each expanded in ω by a factor of K.

Downsampled z-transform

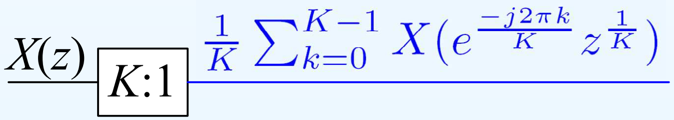
Define $c_K[n] = \delta_{K|n}[n] = \frac{1}{K}\sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}}$



Now define $x_K[n] = \begin{cases} x[n] & K \mid n \\ 0 & K \nmid n \end{cases} = c_K[n]x[n]$

$$X_K(z) = \sum_n x_K[n]z^{-n} = \frac{1}{K}\sum_n \sum_{k=0}^{K-1} e^{\frac{j2\pi kn}{K}} x[n]z^{-n}$$
$$= \frac{1}{K}\sum_{k=0}^{K-1}\sum_n x[n]\left(e^{\frac{-j2\pi k}{K}}z\right)^{-n} = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{-j2\pi k}{K}}z)$$

From previous slide:



$$X_K(z) = Y(z^K)$$
$$\Rightarrow Y(z) = X_K(z^{\frac{1}{K}}) = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{-j2\pi k}{K}}z^{\frac{1}{K}})$$

Frequency Spectrum:

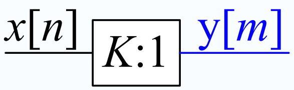
$$Y(e^{j\omega}) = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{j(\omega-2\pi k)}{K}})$$
$$= \frac{1}{K}\left(X(e^{\frac{j\omega}{K}}) + X(e^{\frac{j\omega}{K}-\frac{2\pi}{K}}) + X(e^{\frac{j\omega}{K}-\frac{4\pi}{K}}) + \cdots\right)$$

Average of K aliased versions, each expanded in ω by a factor of K.

Downsampling is normally preceded by a LP filter to prevent aliasing.

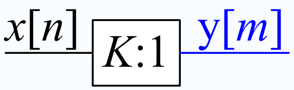
Downsampled Spectrum

$$Y(e^{j\omega}) = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{j(\omega-2\pi k)}{K}})$$



Downsampled Spectrum

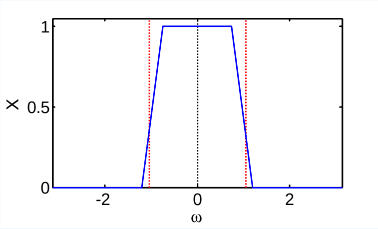
$$Y(e^{j\omega}) = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{j(\omega-2\pi k)}{K}})$$



Example 1:

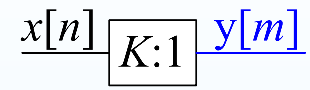
$K = 3$

Not quite limited to $\pm\frac{\pi}{K}$



Downsampled Spectrum

$Y(e^{j\omega}) = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{j(\omega-2\pi k)}{K}})$

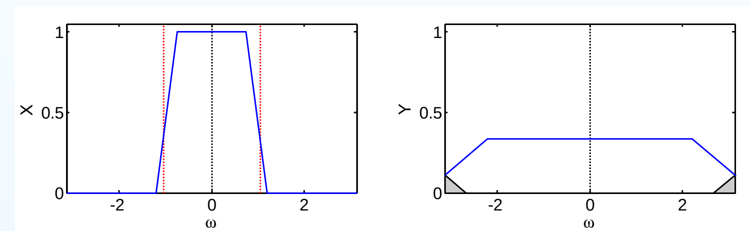


Example 1:

$K = 3$

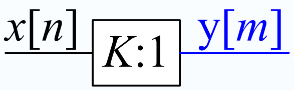
Not quite limited to $\pm\frac{\pi}{K}$

Shaded region shows aliasing



Downsampled Spectrum

$$Y(e^{j\omega}) = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{j(\omega-2\pi k)}{K}})$$

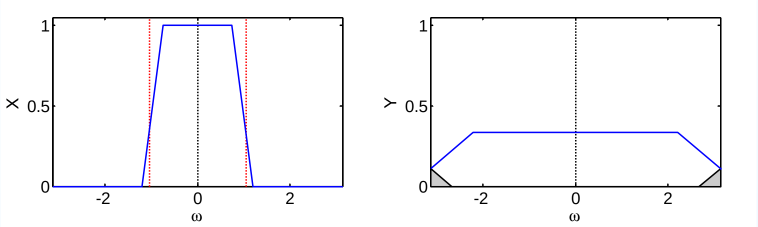


Example 1:

$K = 3$

Not quite limited to $\pm\frac{\pi}{K}$

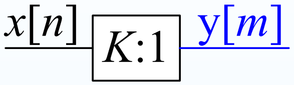
Shaded region shows aliasing



Energy decreases: $\frac{1}{2\pi}\int \left|Y(e^{j\omega})\right|^2 d\omega \approx \frac{1}{K} \times \frac{1}{2\pi}\int \left|X(e^{j\omega})\right|^2 d\omega$

Downsampled Spectrum

$$Y(e^{j\omega}) = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{j(\omega - 2\pi k)}{K}})$$

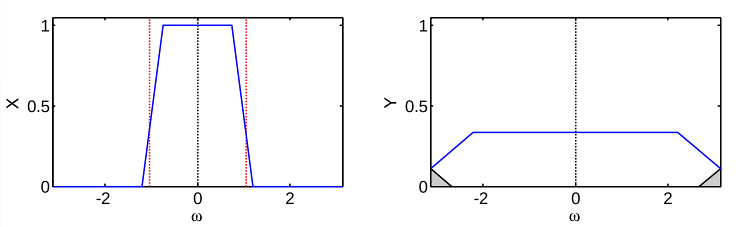


Example 1:

$K = 3$

Not quite limited to $\pm\frac{\pi}{K}$

Shaded region shows aliasing

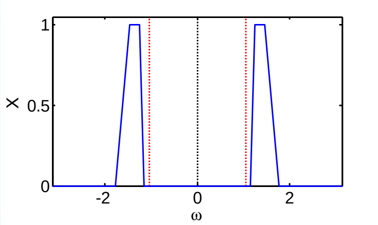


Energy decreases: $\frac{1}{2\pi}\int \left|Y(e^{j\omega})\right|^2 d\omega \approx \frac{1}{K} \times \frac{1}{2\pi}\int \left|X(e^{j\omega})\right|^2 d\omega$

Example 2:

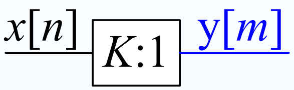
$K = 3$

Energy all in $\frac{\pi}{K} \le |\omega| < 2\frac{\pi}{K}$



Downsampled Spectrum

$$Y(e^{j\omega}) = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{j(\omega-2\pi k)}{K}})$$

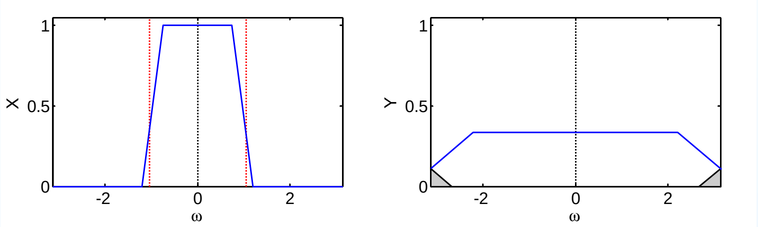


Example 1:

$K = 3$

Not quite limited to $\pm\frac{\pi}{K}$

Shaded region shows aliasing



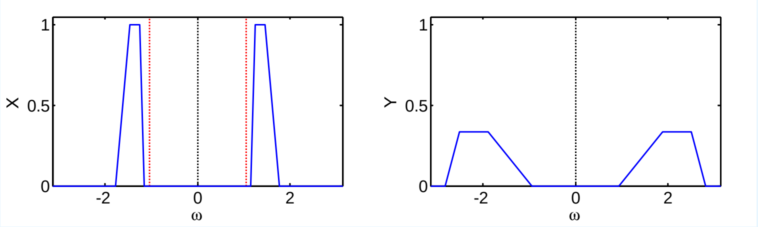
Energy decreases: $\frac{1}{2\pi}\int \left|Y(e^{j\omega})\right|^2 d\omega \approx \frac{1}{K} \times \frac{1}{2\pi}\int \left|X(e^{j\omega})\right|^2 d\omega$

Example 2:

$K = 3$

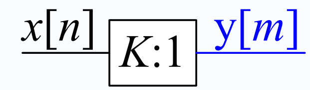
Energy all in $\frac{\pi}{K} \le |\omega| < 2\frac{\pi}{K}$

No aliasing: ☺



Downsampled Spectrum

$$Y(e^{j\omega}) = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{j(\omega-2\pi k)}{K}})$$

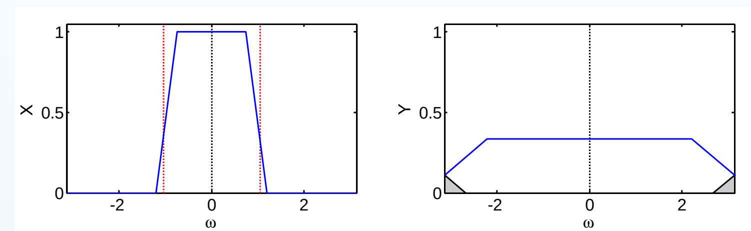


Example 1:

$K = 3$

Not quite limited to $\pm\frac{\pi}{K}$

Shaded region shows aliasing



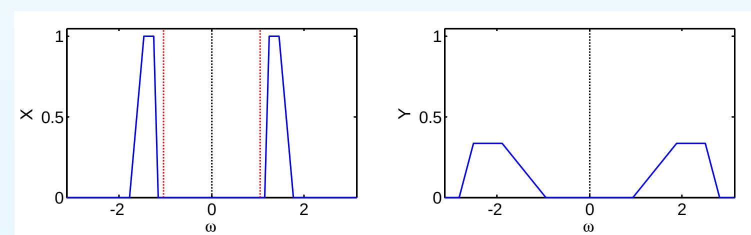
Energy decreases: $\frac{1}{2\pi}\int \left|Y(e^{j\omega})\right|^2 d\omega \approx \frac{1}{K} \times \frac{1}{2\pi}\int \left|X(e^{j\omega})\right|^2 d\omega$

Example 2:

$K = 3$

Energy all in $\frac{\pi}{K} \leq |\omega| < 2\frac{\pi}{K}$

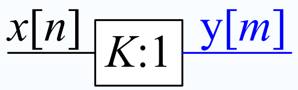
No aliasing: ☺



No aliasing: If all energy is in $r\frac{\pi}{K} \leq |\omega| < (r+1)\frac{\pi}{K}$ for some integer r

Downsampled Spectrum

$$Y(e^{j\omega}) = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{j(\omega-2\pi k)}{K}})$$

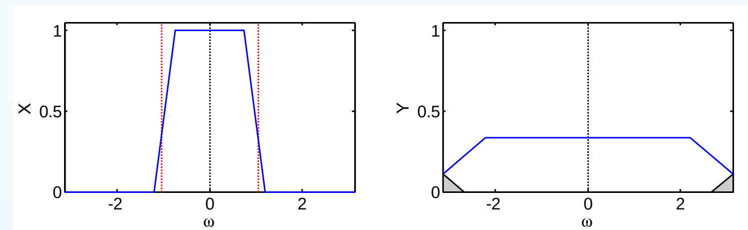


Example 1:

$K = 3$

Not quite limited to $\pm\frac{\pi}{K}$

Shaded region shows aliasing



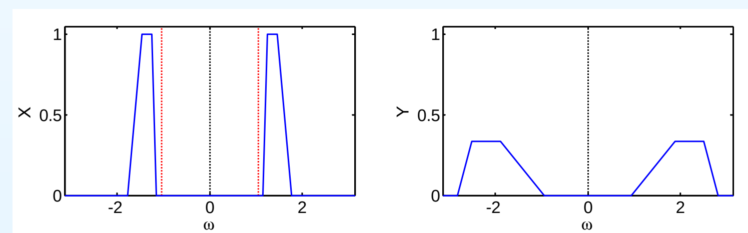
Energy decreases: $\frac{1}{2\pi}\int \left|Y(e^{j\omega})\right|^2 d\omega \approx \frac{1}{K} \times \frac{1}{2\pi}\int \left|X(e^{j\omega})\right|^2 d\omega$

Example 2:

$K = 3$

Energy all in $\frac{\pi}{K} \leq |\omega| < 2\frac{\pi}{K}$

No aliasing: ☺

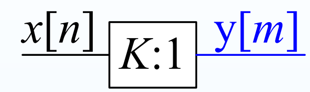


No aliasing: If all energy is in $r\frac{\pi}{K} \leq |\omega| < (r+1)\frac{\pi}{K}$ for some integer r

Normal case ($r = 0$): If all energy in $0 \leq |\omega| \leq \frac{\pi}{K}$

Downsampled Spectrum

$$Y(e^{j\omega}) = \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{j(\omega-2\pi k)}{K}})$$

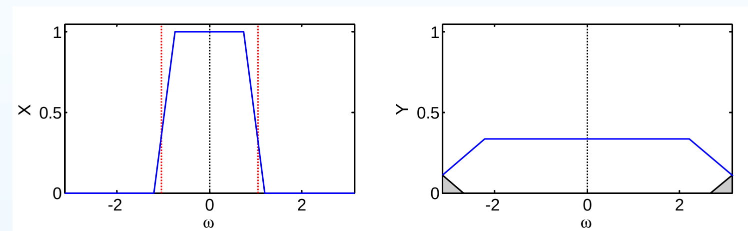


Example 1:

$K = 3$

Not quite limited to $\pm\frac{\pi}{K}$

Shaded region shows aliasing



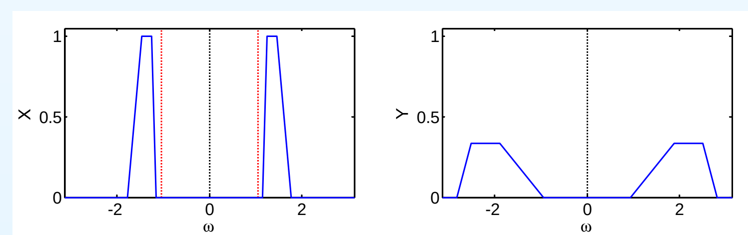
Energy decreases: $\frac{1}{2\pi}\int \left|Y(e^{j\omega})\right|^2 d\omega \approx \frac{1}{K} \times \frac{1}{2\pi}\int \left|X(e^{j\omega})\right|^2 d\omega$

Example 2:

$K = 3$

Energy all in $\frac{\pi}{K} \le |\omega| < 2\frac{\pi}{K}$

No aliasing: ☺



No aliasing: If all energy is in $r\frac{\pi}{K} \le |\omega| < (r+1)\frac{\pi}{K}$ for some integer r

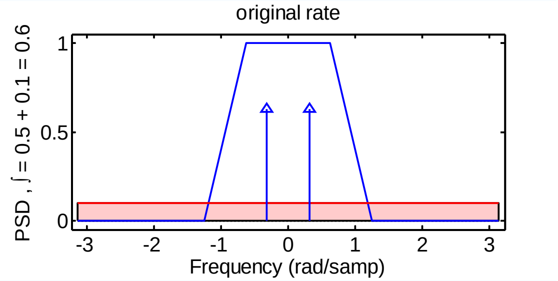
Normal case ($r = 0$): If all energy in $0 \le |\omega| \le \frac{\pi}{K}$

Downsampling: Total energy multiplied by $\approx \frac{1}{K}$ ($= \frac{1}{K}$ if no aliasing)

Average power $\approx$ unchanged ($=$ energy/sample)

Power Spectral Density

Example: Signal in $\omega \in \pm 0.4\pi$ + Tone @ $\omega = \pm 0.1\pi$ + White noise

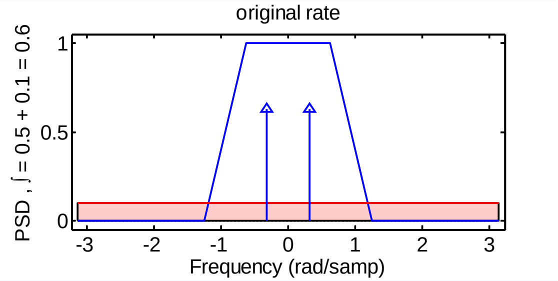


Power Spectral Density

+

Example: Signal in $\omega \in \pm 0.4\pi$ + Tone @ $\omega = \pm 0.1\pi$ + White noise

Power $=$ Energy/sample $=$ Average PSD



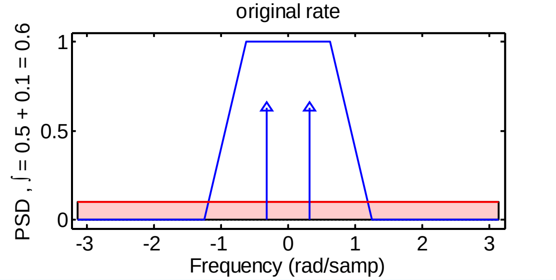
Power Spectral Density

+

Example: Signal in $\omega \in \pm 0.4\pi$ + Tone @ $\omega = \pm 0.1\pi$ + White noise

Power $=$ Energy/sample $=$ Average PSD

$$= \frac{1}{2\pi} \int_{-\pi}^{\pi} \text{PSD}(\omega) d\omega$$

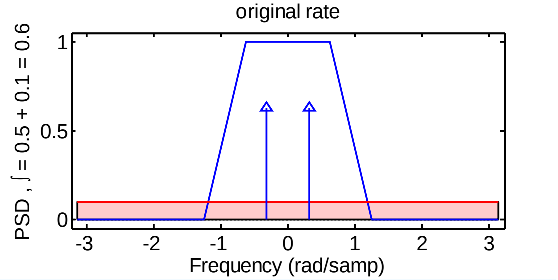


Power Spectral Density

Example: Signal in $\omega \in \pm 0.4\pi$ + Tone @ $\omega = \pm 0.1\pi$ + White noise

Power $=$ Energy/sample $=$ Average PSD

$$= \frac{1}{2\pi}\int_{-\pi}^{\pi} \mathrm{PSD}(\omega) d\omega \;= 0.6$$



Power Spectral Density

+

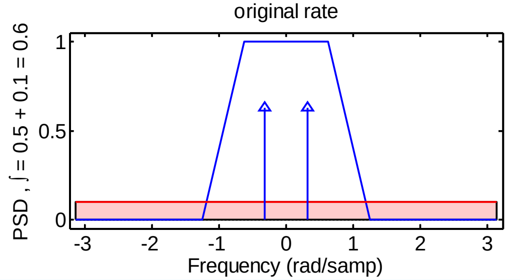
Example: Signal in $\omega \in \pm 0.4\pi$ + Tone @ $\omega = \pm 0.1\pi$ + White noise

Power $=$ Energy/sample $=$ Average PSD

$$= \frac{1}{2\pi}\int_{-\pi}^{\pi} \mathrm{PSD}(\omega)d\omega \ = 0.6$$

Component powers:

Signal = 0.3, Tone = 0.2, Noise = 0.1



Power Spectral Density +

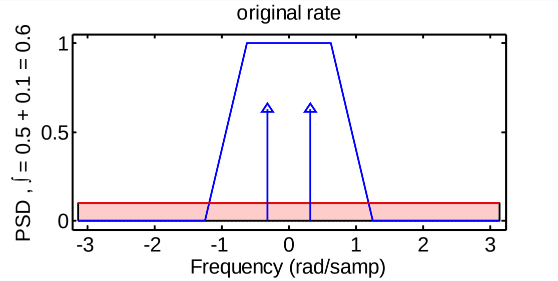
Example: Signal in $\omega \in \pm 0.4\pi$ + Tone @ $\omega = \pm 0.1\pi$ + White noise

Power $=$ Energy/sample $=$ Average PSD

$$= \frac{1}{2\pi}\int_{-\pi}^{\pi} \mathrm{PSD}(\omega)d\omega \; = 0.6$$

Component powers:

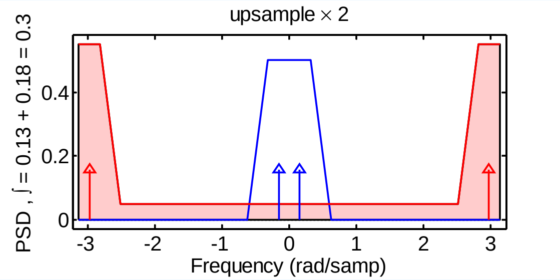
Signal = 0.3, Tone = 0.2, Noise = 0.1



Upsampling:

Same energy per second

$\Rightarrow$ Power is $\div K$



Power Spectral Density

+

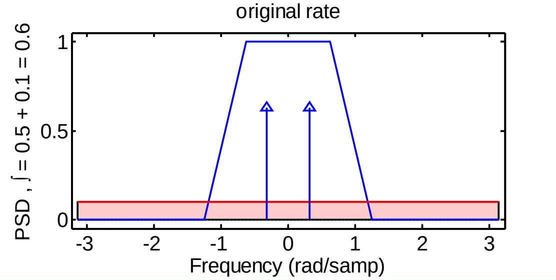
Example: Signal in $\omega \in \pm 0.4\pi$ + Tone @ $\omega = \pm 0.1\pi$ + White noise

Power $=$ Energy/sample $=$ Average PSD

$$= \frac{1}{2\pi}\int_{-\pi}^{\pi} \text{PSD}(\omega)d\omega \ = 0.6$$

Component powers:

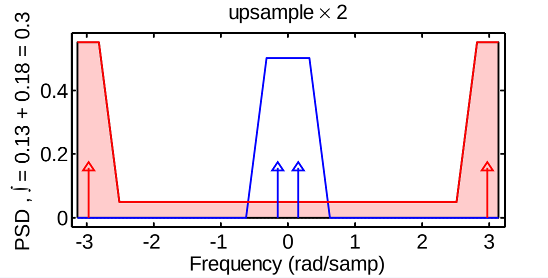
Signal = 0.3, Tone = 0.2, Noise = 0.1

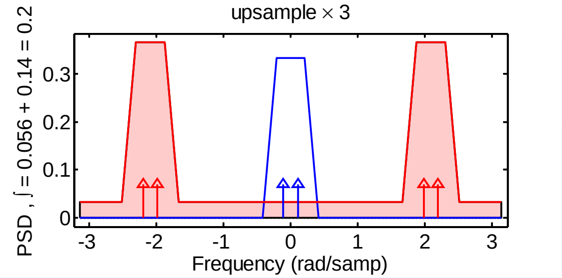


Upsampling:

Same energy per second

$\Rightarrow$ Power is $\div K$





Power Spectral Density

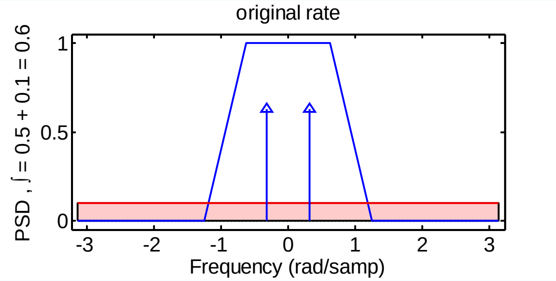
Example: Signal in $\omega \in \pm 0.4\pi$ + Tone @ $\omega = \pm 0.1\pi$ + White noise

Power $=$ Energy/sample $=$ Average PSD

$$= \frac{1}{2\pi}\int_{-\pi}^{\pi} \text{PSD}(\omega)d\omega \;= 0.6$$

Component powers:

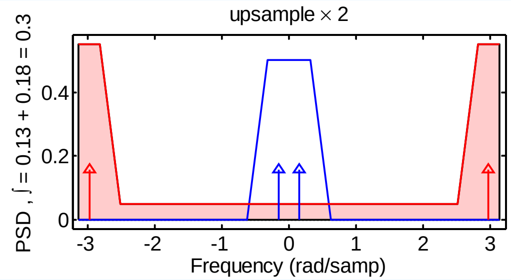
Signal = 0.3, Tone = 0.2, Noise = 0.1

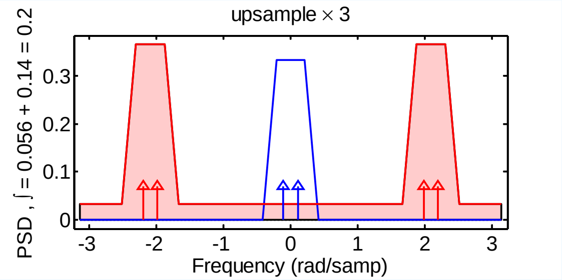


Upsampling:

Same energy per second

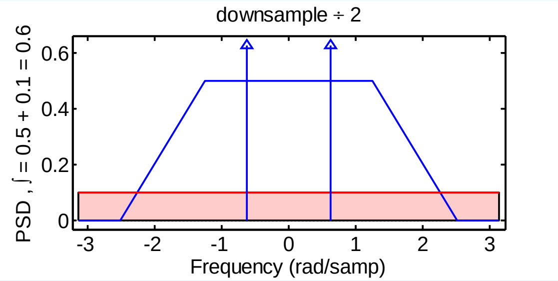
$\Rightarrow$ Power is $\div K$





Downsampling:

Average power is unchanged.



Power Spectral Density +

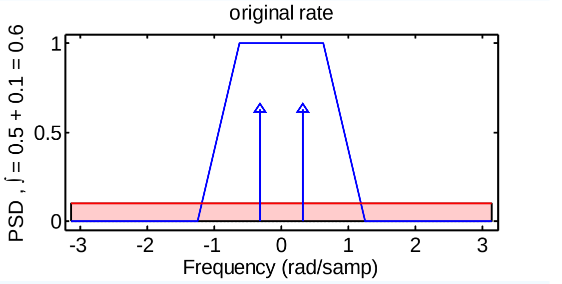
Example: Signal in $\omega \in \pm 0.4\pi$ + Tone @ $\omega = \pm 0.1\pi$ + White noise

Power $=$ Energy/sample $=$ Average PSD

$$= \frac{1}{2\pi}\int_{-\pi}^{\pi} \text{PSD}(\omega)d\omega \;= 0.6$$

Component powers:

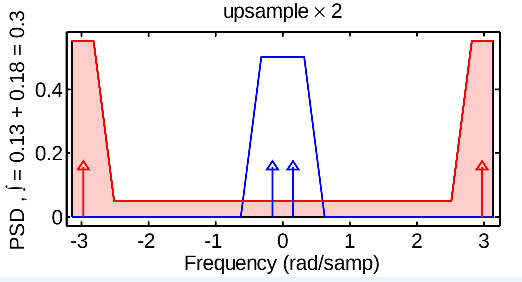
Signal = 0.3, Tone = 0.2, Noise = 0.1

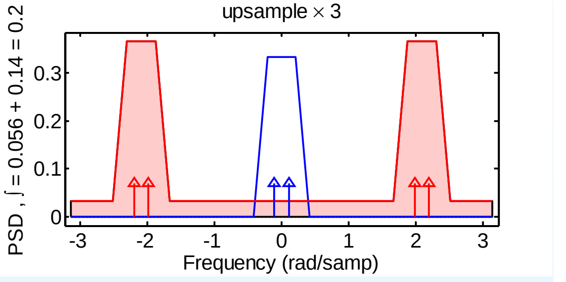


Upsampling:

Same energy per second

$\Rightarrow$ Power is $\div K$

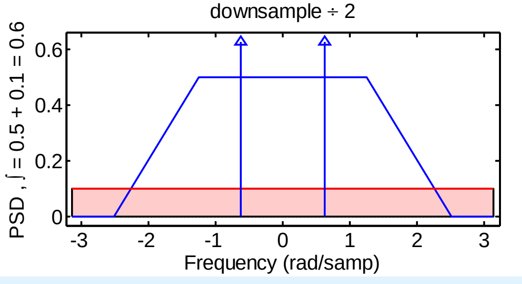


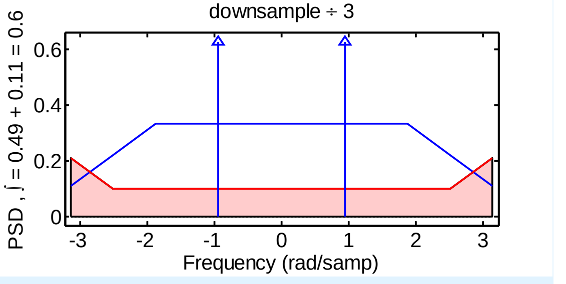


Downsampling:

Average power is unchanged.

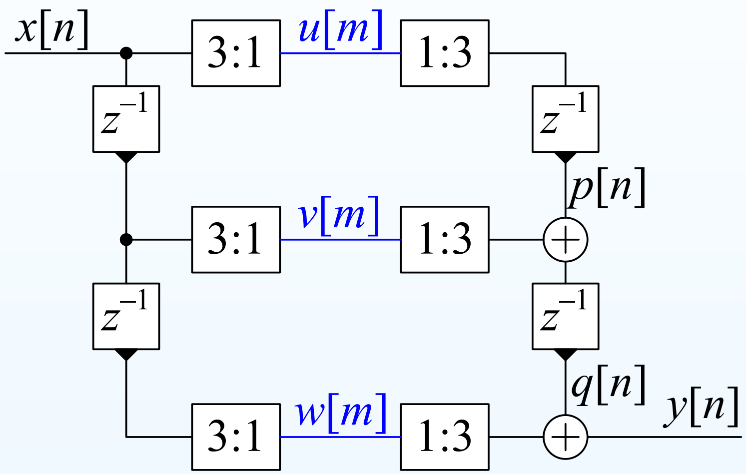
$\exists$ aliasing in the $\div 3$ case.





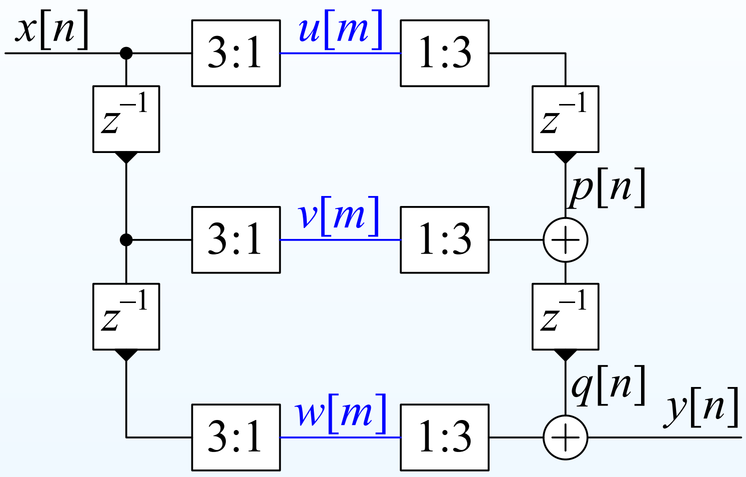
Perfect Reconstruction

$x[n]$ `cdefghijklmn`



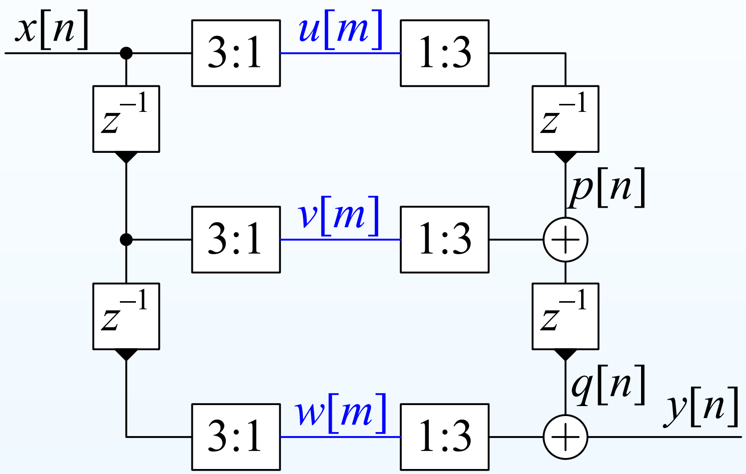
Perfect Reconstruction

$x[n]$	`cdefghijklmn`
$u[m]$	`c  f  i  l`



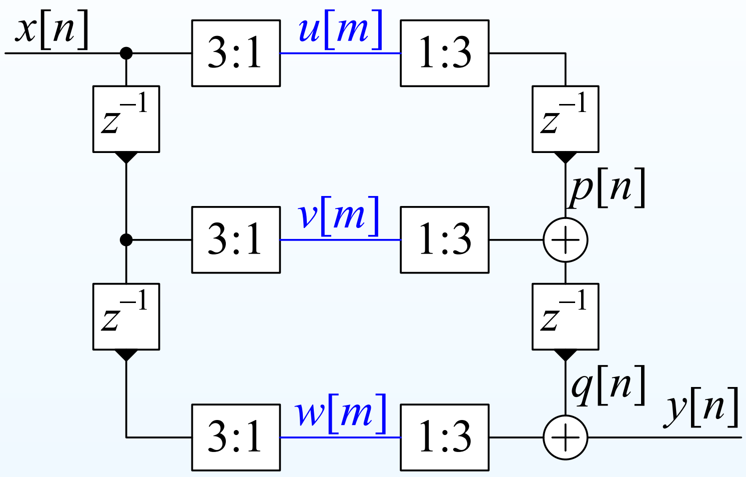
Perfect Reconstruction

$x[n]$	`cdefghijklmn`
$u[m]$	`c  f  i  l`
$p[n]$	`-c--f--i--l`



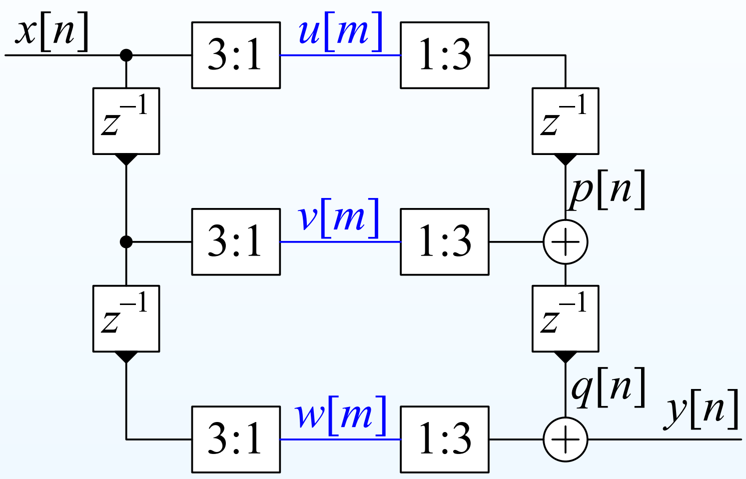
Perfect Reconstruction

$x[n]$	`cdefghijklmn`
$u[m]$	`c  f  i  l`
$p[n]$	`-c--f--i--l`
$v[m]$	`b  e  h  k`



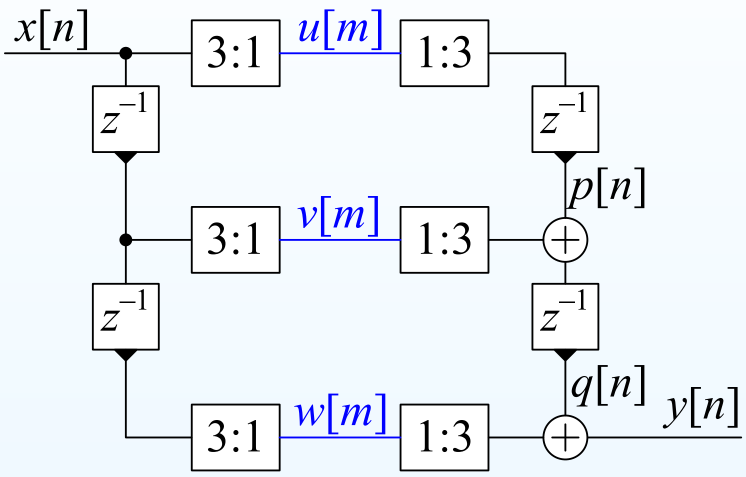
Perfect Reconstruction

$x[n]$	`cdefghijklmn`
$u[m]$	`c  f  i  l`
$p[n]$	`-c--f--i--l`
$v[m]$	`b  e  h  k`
$q[n]$	`-bc-ef-hi-kl`



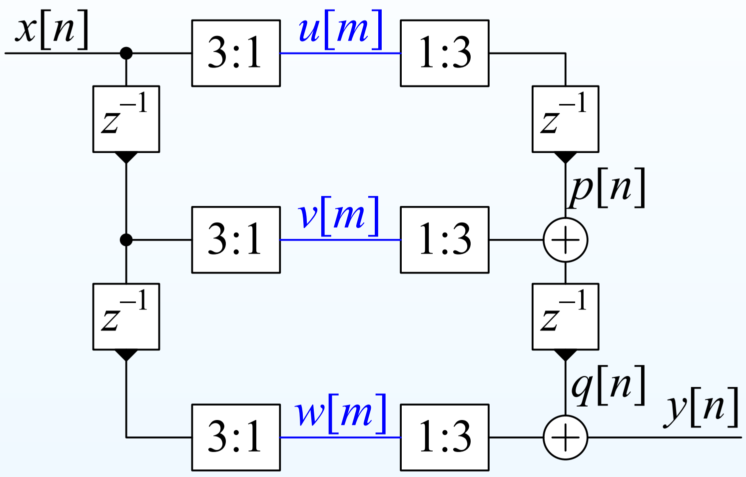
Perfect Reconstruction

$x[n]$	`cdefghijklmn`
$u[m]$	`c  f  i  l`
$p[n]$	`-c--f--i--l`
$v[m]$	`b  e  h  k`
$q[n]$	`-bc-ef-hi-kl`
$w[m]$	`a  d  g  j`



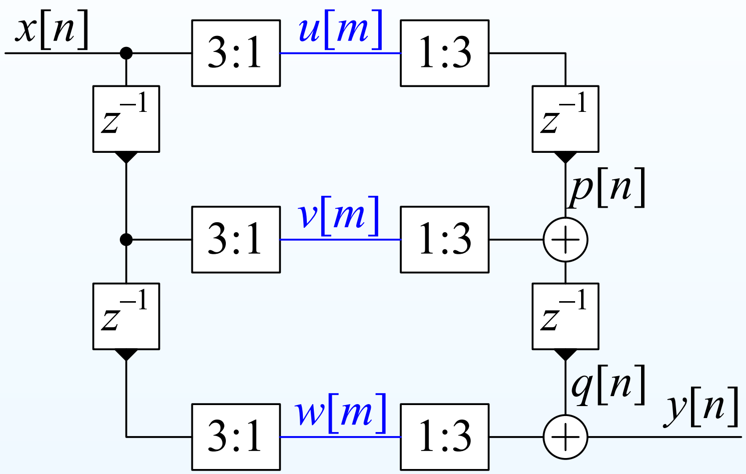
Perfect Reconstruction

$x[n]$	`cdefghijklmn`
$u[m]$	`c  f  i  l`
$p[n]$	`-c--f--i--l`
$v[m]$	`b  e  h  k`
$q[n]$	`-bc-ef-hi-kl`
$w[m]$	`a  d  g  j`
$y[n]$	`abcdefghijkl`



Perfect Reconstruction

$x[n]$	`cdefghijklmn`
$u[m]$	`c  f  i  l`
$p[n]$	`-c--f--i--l`
$v[m]$	`b  e  h  k`
$q[n]$	`-bc-ef-hi-kl`
$w[m]$	`a  d  g  j`
$y[n]$	`abcdefghijkl`

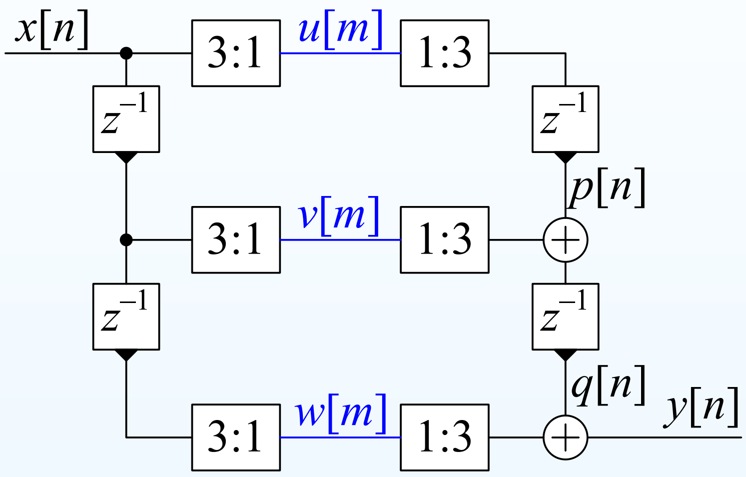


Input sequence $x[n]$ is split into three streams at $\frac{1}{3}$ the sample rate:

$$u[m] = x[3m],\ v[m] = x[3m-1],\ w[m] = x[3m-2]$$

Perfect Reconstruction

$x[n]$	`cdefghijklmn`
$u[m]$	`c  f  i  l`
$p[n]$	`-c--f--i--l`
$v[m]$	`b  e  h  k`
$q[n]$	`-bc-ef-hi-kl`
$w[m]$	`a  d  g  j`
$y[n]$	`abcdefghijkl`



Input sequence $x[n]$ is split into three streams at $\frac{1}{3}$ the sample rate:

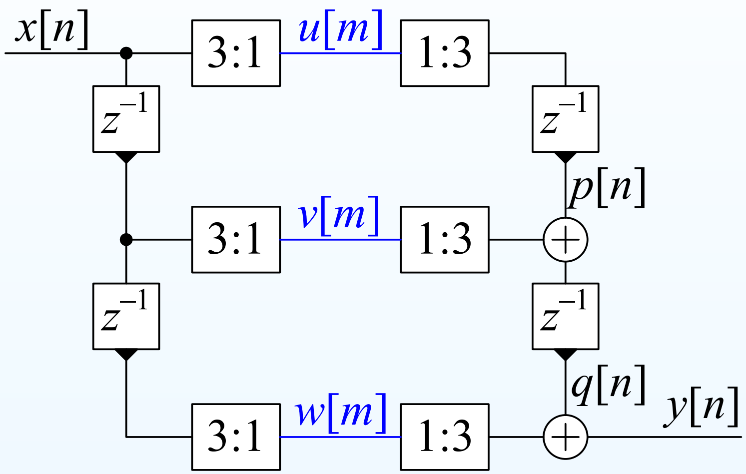
$$u[m] = x[3m],\ v[m] = x[3m-1],\ w[m] = x[3m-2]$$

Following upsampling, the streams are aligned by the delays and then added to give:

$$y[n] = x[n-2]$$

Perfect Reconstruction

$x[n]$	`cdefghijklmn`
$u[m]$	`c  f  i  l`
$p[n]$	`-c--f--i--l`
$v[m]$	`b  e  h  k`
$q[n]$	`-bc-ef-hi-kl`
$w[m]$	`a  d  g  j`
$y[n]$	`abcdefghijkl`



Input sequence $x[n]$ is split into three streams at $\frac{1}{3}$ the sample rate:

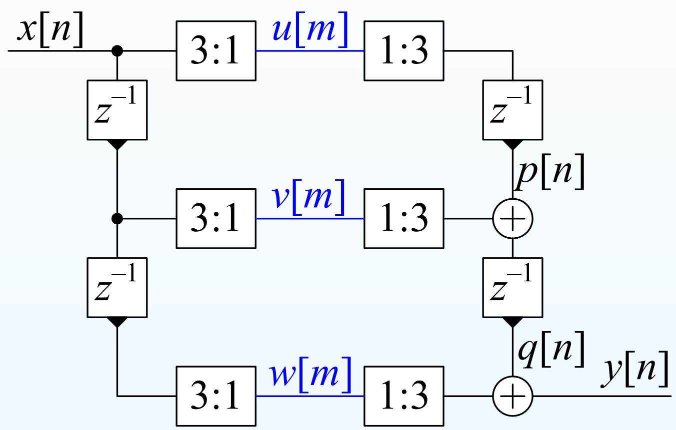
$$u[m] = x[3m],\ v[m] = x[3m-1],\ w[m] = x[3m-2]$$

Following upsampling, the streams are aligned by the delays and then added to give:

$$y[n] = x[n-2]$$

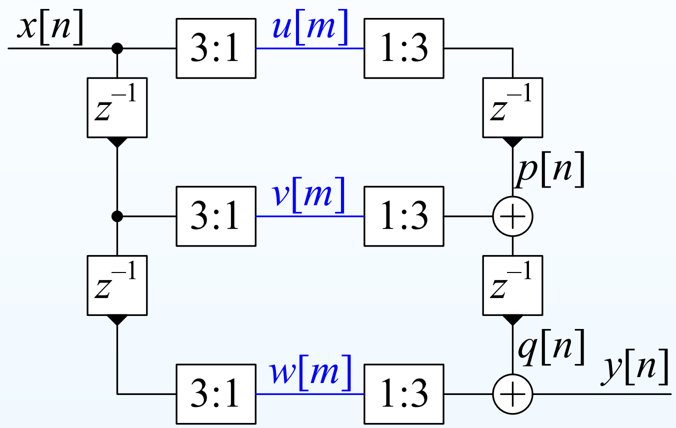
Perfect Reconstruction: output is a delayed scaled replica of the input

Commutators



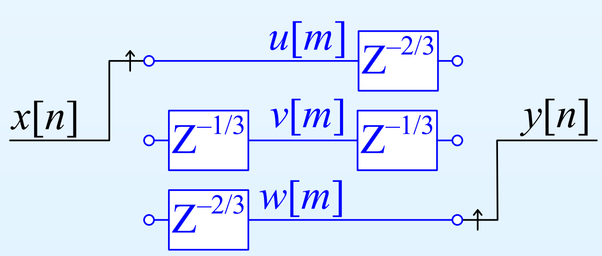
$x[n]$	cdefghijklmn
$u[m]$	c f i l
$v[m]$	b e h k
$w[m]$	a d g j
$y[n]$	abcdefghijkl

Commutators

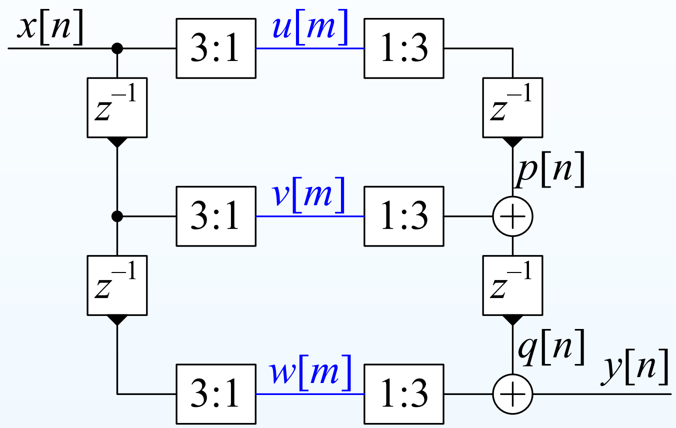


$x[n]$	`cdefghijklmn`
$u[m]$	`c f i l`
$v[m]$	`b e h k`
$w[m]$	`a d g j`
$y[n]$	`abcdefghijkl`

The combination of delays and downsamplers can be regarded as a commutator that distributes values in sequence to u, w and v.

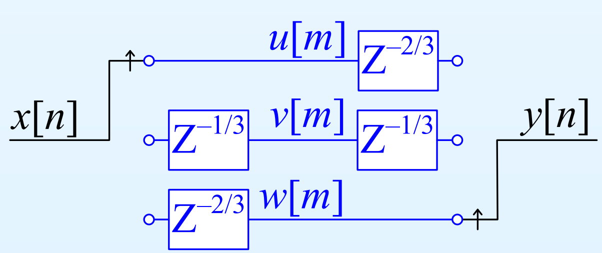


Commutators

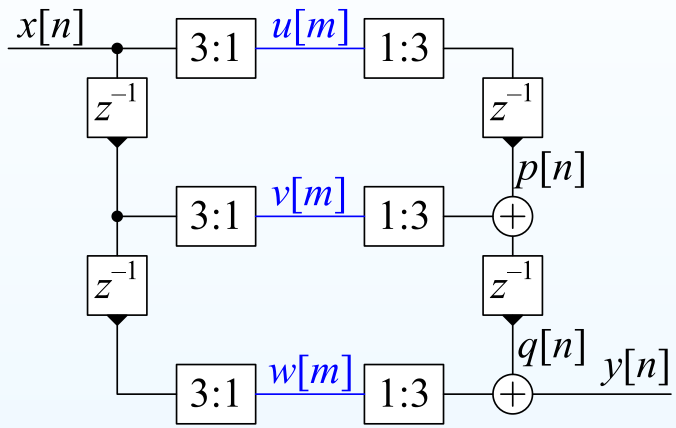


$x[n]$	`cdefghijklmn`
$u[m]$	`c  f  i  l`
$v[m]$	`b  e  h  k`
$w[m]$	`a  d  g  j`
$y[n]$	`abcdefghijkl`

The combination of delays and downsamplers can be regarded as a commutator that distributes values in sequence to u, w and v.
Fractional delays, $z^{-\frac{1}{3}}$ and $z^{-\frac{2}{3}}$ are needed to synchronize the streams.



Commutators

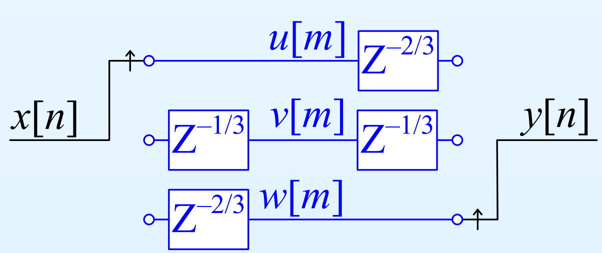


$x[n]$	`cdefghijklmn`
$u[m]$	`c  f  i  l`
$v[m]$	`b  e  h  k`
$w[m]$	`a  d  g  j`
$y[n]$	`abcdefghijkl`

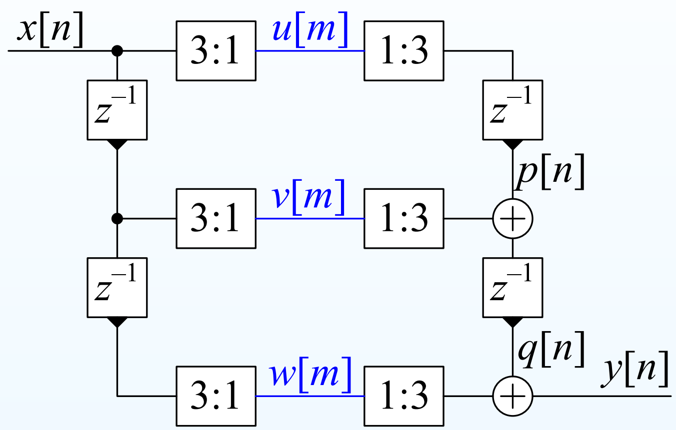
The combination of delays and downsamplers can be regarded as a commutator that distributes values in sequence to u, w and v.

Fractional delays, $z^{-\frac{1}{3}}$ and $z^{-\frac{2}{3}}$ are needed to synchronize the streams.

The output commutator takes values from the streams in sequence.

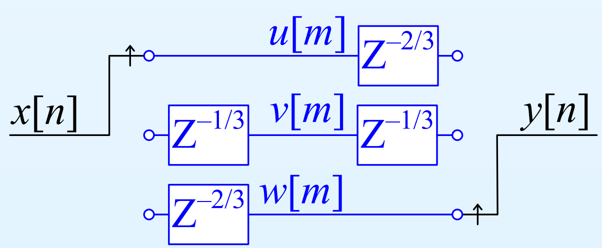


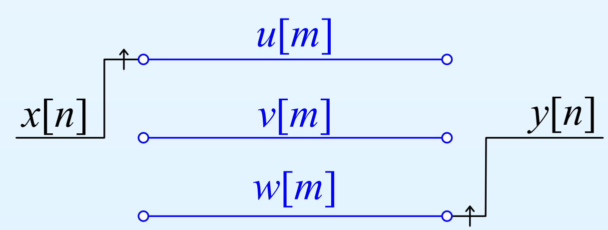
Commutators



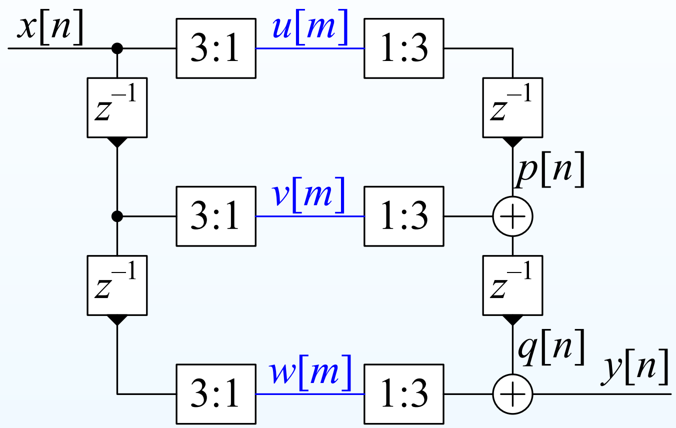
$x[n]$	`cdefghijklmn`
$u[m]$	`c  f  i  l`
$v[m]$	`b  e  h  k`
$w[m]$	`a  d  g  j`
$v[m+\frac{1}{3}]$	`e  h  k  l`
$w[m+\frac{2}{3}]$	`d  g  j  m`
$y[n]$	`abcdefghijkl`

The combination of delays and downsamplers can be regarded as a commutator that distributes values in sequence to u, w and v.
Fractional delays, $z^{-\frac{1}{3}}$ and $z^{-\frac{2}{3}}$ are needed to synchronize the streams.
The output commutator takes values from the streams in sequence.
For clarity, we omit the fractional delays and regard each terminal, $\circ$, as holding its value until needed.





Commutators



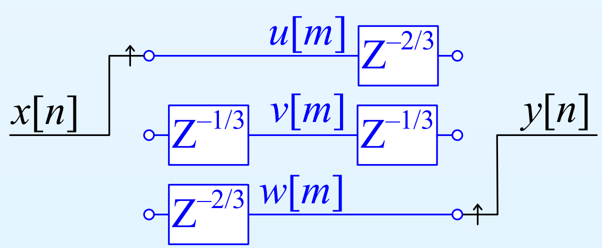
$x[n]$	c	d	e	f	g	h	i	j	k	l	m	n
$u[m]$	c			f			i			l		
$v[m]$	b			e			h			k		
$w[m]$	a			d			g			j		
$v[m+\frac{1}{3}]$			e			h			k			l
$w[m+\frac{2}{3}]$		d			g			j			m	
$y[n]$	a	b	c	d	e	f	g	h	i	j	k	l

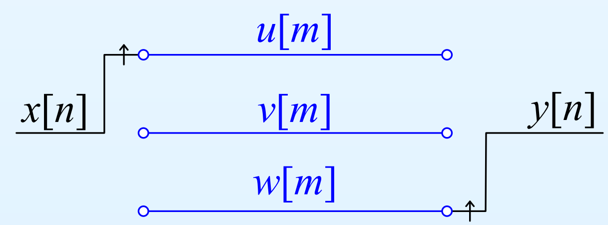
The combination of delays and downsamplers can be regarded as a commutator that distributes values in sequence to u, w and v.

Fractional delays, $z^{-\frac{1}{3}}$ and $z^{-\frac{2}{3}}$ are needed to synchronize the streams.

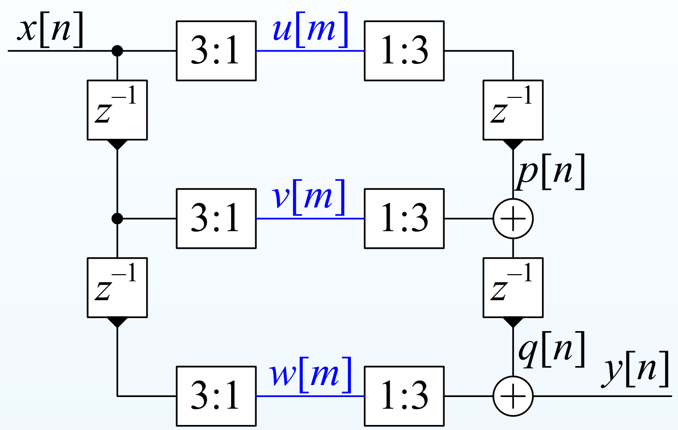
The output commutator takes values from the streams in sequence.

For clarity, we omit the fractional delays and regard each terminal, $\circ$, as holding its value until needed. Initial commutator position has zero delay.





Commutators



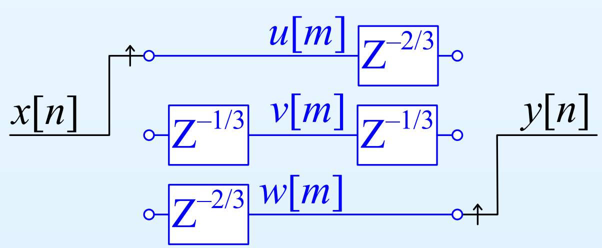
$x[n]$	`cdefghijklmn`
$u[m]$	`c  f  i  l`
$v[m]$	`b  e  h  k`
$w[m]$	`a  d  g  j`
$v[m+\frac{1}{3}]$	`e  h  k  l`
$w[m+\frac{2}{3}]$	`d  g  j  m`
$y[n]$	`abcdefghijkl`

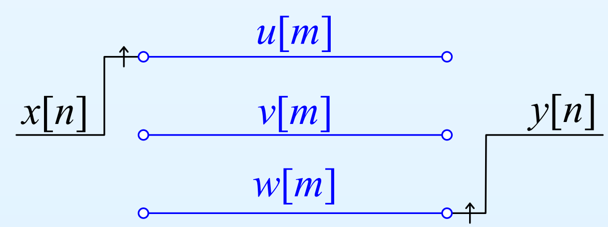
The combination of delays and downsamplers can be regarded as a commutator that distributes values in sequence to u, w and v.

Fractional delays, $z^{-\frac{1}{3}}$ and $z^{-\frac{2}{3}}$ are needed to synchronize the streams.

The output commutator takes values from the streams in sequence.

For clarity, we omit the fractional delays and regard each terminal, $\circ$, as holding its value until needed. Initial commutator position has zero delay.





The commutator direction is against the direction of the z^{-1} delays.

Summary

- Multirate Building Blocks
  - Upsample: $X(z) \stackrel{1:K}{\rightarrow} X(z^K)$
    Invertible, Inserts $K-1$ zeros between samples
    Shrinks and replicates spectrum
    Follow by LP filter to remove images

Summary

- Multirate Building Blocks
  - Upsample: $X(z) \stackrel{1:K}{\rightarrow} X(z^K)$
    Invertible, Inserts $K-1$ zeros between samples
    Shrinks and replicates spectrum
    Follow by LP filter to remove images
  - Downsample: $X(z) \stackrel{K:1}{\rightarrow} \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{-j2\pi k}{K}} z^{\frac{1}{K}})$
    Destroys information and energy, keeps every K^{th} sample
    Expands and aliasses the spectrum
    Spectrum is the average of K aliased expanded versions
    Precede by LP filter to prevent aliases

Summary

- Multirate Building Blocks
  - Upsample: $X(z) \stackrel{1:K}{\rightarrow} X(z^K)$
    Invertible, Inserts $K-1$ zeros between samples
    Shrinks and replicates spectrum
    Follow by LP filter to remove images
  - Downsample: $X(z) \stackrel{K:1}{\rightarrow} \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{-j2\pi k}{K}} z^{\frac{1}{K}})$
    Destroys information and energy, keeps every K^{th} sample
    Expands and aliasses the spectrum
    Spectrum is the average of K aliased expanded versions
    Precede by LP filter to prevent aliases
- Equivalences
  - Noble Identities: $H(z) \longleftrightarrow H(z^K)$
  - Interchange $P:1$ and $1:Q$ iff $P$and Q coprime

Summary

- Multirate Building Blocks
  - Upsample: $X(z) \stackrel{1:K}{\to} X(z^K)$
    Invertible, Inserts $K-1$ zeros between samples
    Shrinks and replicates spectrum
    Follow by LP filter to remove images
  - Downsample: $X(z) \stackrel{K:1}{\to} \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{-j2\pi k}{K}} z^{\frac{1}{K}})$
    Destroys information and energy, keeps every K^{th} sample
    Expands and aliasses the spectrum
    Spectrum is the average of K aliased expanded versions
    Precede by LP filter to prevent aliases
- Equivalences
  - Noble Identities: $H(z) \longleftrightarrow H(z^K)$
  - Interchange $P:1$ and $1:Q$ iff $P$and Q coprime
- Commutators
  - Combine delays and down/up sampling

Summary

- Multirate Building Blocks
  - Upsample: $X(z) \stackrel{1:K}{\rightarrow} X(z^K)$
    Invertible, Inserts $K-1$ zeros between samples
    Shrinks and replicates spectrum
    Follow by LP filter to remove images
  - Downsample: $X(z) \stackrel{K:1}{\rightarrow} \frac{1}{K}\sum_{k=0}^{K-1} X(e^{\frac{-j2\pi k}{K}} z^{\frac{1}{K}})$
    Destroys information and energy, keeps every K^{th} sample
    Expands and aliasses the spectrum
    Spectrum is the average of K aliased expanded versions
    Precede by LP filter to prevent aliases
- Equivalences
  - Noble Identities: $H(z) \longleftrightarrow H(z^K)$
  - Interchange $P:1$ and $1:Q$ iff $P$and Q coprime
- Commutators
  - Combine delays and down/up sampling

For further details see Mitra: 13.

MATLAB routines

resample	change sampling rate