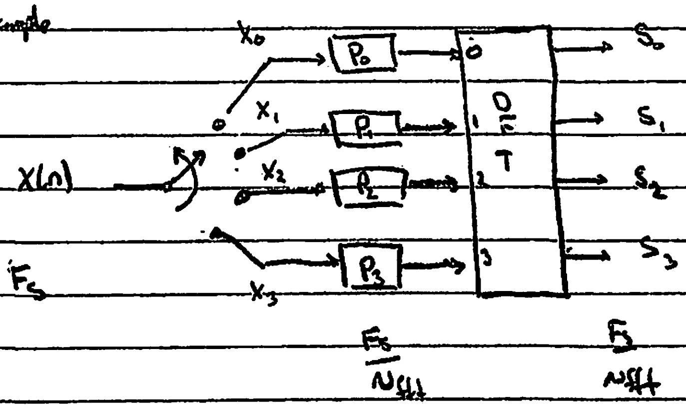
Polyphase Filter → TMUX

$$\boxed{\text{Polyphase Filter} \rightarrow \text{MFFT}} = \text{TMUX}$$

when

$F_{s_{out}} = \frac{F_s}{N_{fft}}$, polyphase filter is most efficient.

example

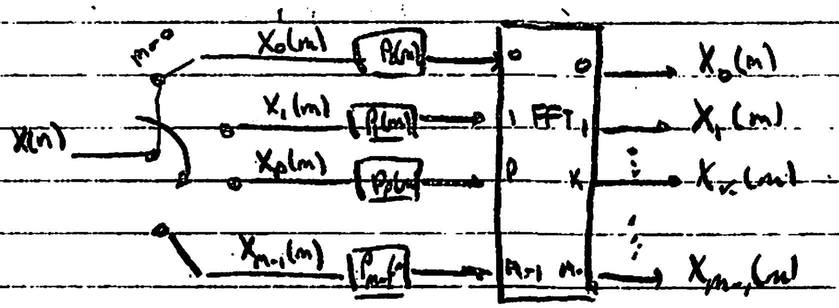


P is a lowpass filter split into parallel paths using polyphase filtering concepts. $h(n)$ is filter impulse response.

output filter centers are at $f_{c_k} = \frac{k F_s}{N_{fft}}$

If input blocks are overlapped then output is oversampled at a rate $\frac{F_s}{D}$, where $D = (1 - \text{overlap}) N_{fft}$.

Polyphase Filterbank w/ FFT modulator



$$\cancel{X_k(m) = \sum_{p=0}^{M-1} \sum_{r=-\infty}^{\infty} P_p(r)\, W_M^{-kp}\, x_p(m-r)}$$

Data Input Structure:

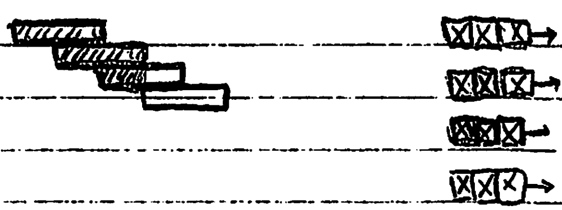
$\longrightarrow$ $x(0)$ $x(M)$ $x(2M)$

$\longrightarrow$ $x(1)$ $x(M+1)$ $x(2M+1)$

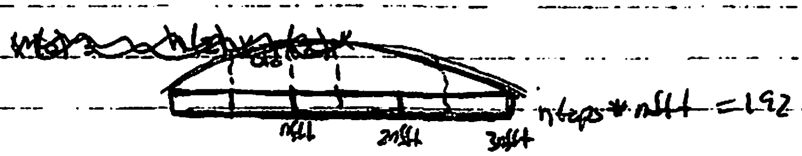
$\longrightarrow$ $x(2)$ . . .

$\vdots$

$\longrightarrow$ $x(M-1)$ $x(2M-1)$ $x(3M-1)$



3-Tap



$$\bar{y}(k) = \sum_{l=0}^{L-1} h(l)\,p(k-l) \qquad L = \text{Length of filter}$$

$$= \sum_{l=0}^{L-1} h(l)\,x(k-l)\,e^{-j2\pi f_0 (k-l)T}$$

$$= \sum_{l=0}^{L-1} h(l)\,x(k-l)\,e^{-j2\pi f_0 kT}\,e^{+j2\pi f_0 lT}$$

$$= e^{-j2\pi f_0 kT} \sum_{l=0}^{L-1} h(l)\,x(k-l)\,e^{j2\pi f_0 lT}$$

Decimate by $M \Rightarrow k = rM$

$$y(r) = e^{-j2\pi f_0 rMT} \sum_{l=0}^{L-1} h(l)\,x(rM-l)\,e^{j2\pi f_0 lT}$$

1- Assume $f_s \equiv N \cdot \Delta f$ and $f_0 \equiv n \cdot \Delta f$

2- Assume $L \equiv Q \cdot N$

3- Assume $M \equiv N$ or $M \equiv N/2$

So, using 1-

$$y(r) = e^{-j2\pi n \Delta f rMT} \sum_{l=0}^{L-1} h(l)\,x(rM-l)\,e^{j2\pi n \Delta f lT}$$

$$y(r) = e^{-j2\pi \frac{nrM}{N}} \sum_{l=0}^{L-1} h(l)\,x(rM-l)\,e^{+j2\pi \frac{nl}{N}} \qquad f_s = \frac{1}{T} \quad T = \frac{1}{N\Delta f}$$

using 2- $\quad l \equiv qN + p \qquad 0 \le q \le Q-1 \;\;\&\;\; 0 \le p \le N-1$

$$y(r) = e^{-j2\pi \frac{nrM}{N}} \sum_{q=0}^{N-1} \sum_{p=0}^{Q-1} h(qN+p)\,x(rM-qN-p)\,e^{j2\pi \frac{n(qN+p)}{N}}$$

$$y(r) = e^{-j2\pi \frac{nrM}{2}} \sum_{q=0}^{N-1} \sum_{p=0}^{Q-1} h(qN+p)\,x(rM-qN-p)\,\overset{1}{\cancel{e^{j2\pi n q \frac{N}{N}}}}\,e^{j2\pi \frac{np}{N}}$$

$$y(r) = e^{-j2\pi n r \frac{M}{2}} \sum_{p=0}^{Q-1} e^{j2\pi \frac{np}{N}} \underbrace{\left[\sum_{q=0}^{N-1} h(qN+p)\,x(rM-qN-p)\right]}_{=\,v(r,p)}$$

$$y(r) = e^{-j2\pi \frac{nrM}{N}} \sum_{p=0}^{N-1} e^{j2\pi \frac{np}{N}} v(r,p)$$

using 3 - $M = \frac{N}{K}$, $K = 1, 2, \text{or } 4$

$$e^{-j2\pi \frac{nrN}{NK}} = e^{-j2\pi \frac{nr}{K}} = \left[e^{-j\frac{2\pi}{K}}\right]^{nr} = \left[-j^{4/K}\right]^{nr}$$

So,

$$y(r) = \left[-j^{\frac{4}{K}}\right]^{nr} \sum_{p=0}^{N-1} e^{j2\pi \frac{np}{N}} v(r,p)$$

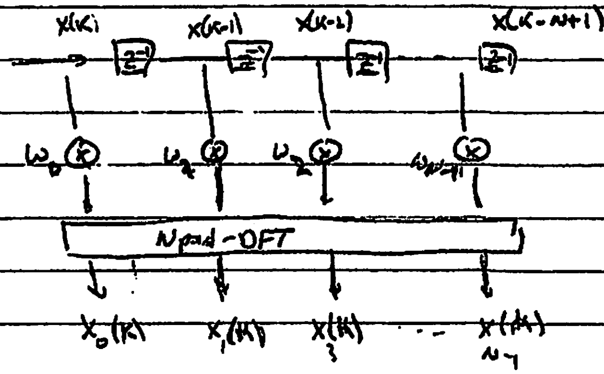
If $K=1$ then

$$y(r) = \sum_{p=0}^{N-1} e^{j2\pi \frac{np}{N}} v(r,p)$$

*Note: The computation of $v(r,p)$ is often called "Polyphase filtering".

What if we want multiple channels?

DFT Filterbank Approach - Another way of looking at it.



$$X_m(k) = \sum_{p=0}^{N-1} x(k-p)\, w(p)\, e^{j2\pi\frac{mp}{N}}$$

Let $\bar{w}_m(p) = w(p)\, e^{j2\pi\frac{mp}{N}}$

$$\Rightarrow X_m(k) = \sum_{p=0}^{N-1} x(k-p)\, \bar{w}_m(p)$$

This is output of a FIR filter with x(k) as the input & $\bar{w}_m(p)$ as the impulse response.

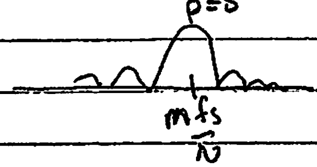
$$X_m(k) = x(p) * \bar{w}_m(p)$$

also,

$$W_m(\omega) = \sum_{p=0}^{N-1} \bar{w}_m(p)\, e^{-j\omega pT}$$ Transfer Function of Filter

If $w(p)=1$ then,

$$W_m(\omega) = \sum_{p=0}^{N-1} e^{j2\pi\frac{mp}{N}} e^{-j\omega pT} = e^{-j\frac{\pi m}{N}} e^{-j\frac{(N-1)\omega T}{2}} \cdot \frac{\sin\frac{N\omega T}{2}}{\sin\left(\frac{\pi m}{N} - \frac{\omega T}{2}\right)}$$

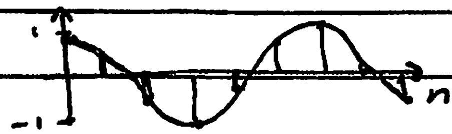


other bins (m) have the same response.

- Decimation Demo 13.2.m 0.042Hz

- Interpolation

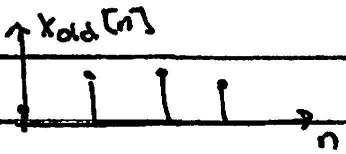
$x_{old}[n]$

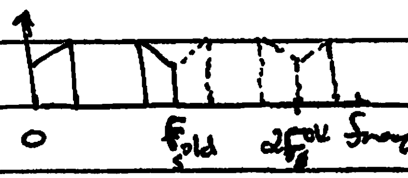


$x_{old}[n] = x[0], x[1], x[2], x[3]$

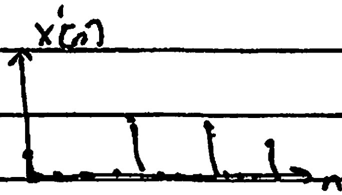
$x'_{new}[n] = x_{old}[0]\ 0\ 0\ 0\ x_{old}[1]\ 0\ 0\ 0\ x_{old}[2]\ 0\ 0\ 0\ \dots$

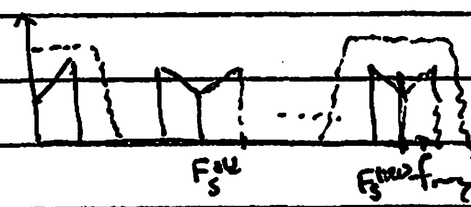
Spectrum -





M=4





Demo 13.1.m
13.3.m

- lowpass filter $x'[n]$ to attenuate spectral Images.
- This filter is called "Interpolation filter".

Note- Accuracy of interpolation process depends on stopband attenuation of lowpass interpolation filter.

Note- Inherent amplitude loss factor of M. To achieve unity gain between $x_{old}[n]$ and $x_{new}[n]$ the filter must have gain of M.

- Combining decimation and interpolation

Q: How do we change sample rate by any rational fraction ~~amount~~ M/D ?

A: $x_{old}[n] \rightarrow \boxed{\uparrow M} - \boxed{LPF_M} \rightarrow \boxed{LPF_D} - \boxed{\downarrow D} \rightarrow x_{new}[n]$

Example

Change sample rate ~~to~~ by 7.125.

$\Rightarrow M = 57, \ D = 8 \qquad \frac{M}{D} = \frac{57}{8} = 7.125$

We can combine LPF_M & LPF_D.

$x_{old}[n] \rightarrow \boxed{\uparrow M} \rightarrow \boxed{LPF_{M/D}} \rightarrow \boxed{\downarrow D} \rightarrow x_{new}$

Called "multirate" filter.

This is again a job of lowpass filter design!

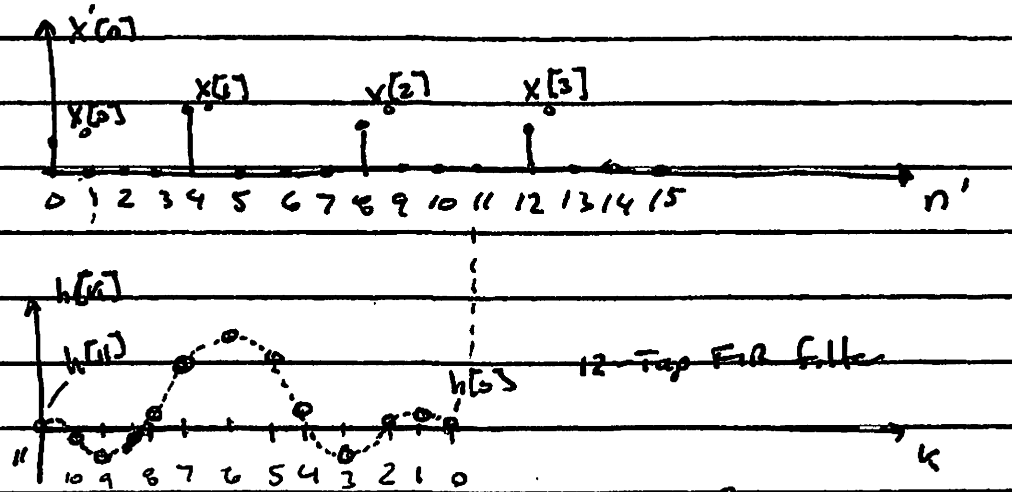
This process is very inefficient.

Example $\frac{M}{D} = \frac{4}{3} \Rightarrow$ stuff 3 zeros then LP-filter.

$\frac{3}{4}$ of multiplies will be ~~zeros~~ zero'd

Next, discard $2/3$ of filter outputs .... very ~~ineff~~ inefficient!

Polyphase Filters



$$y[n] = \sum_{k=-\infty}^{\infty} x[k]\, h[n-k]$$

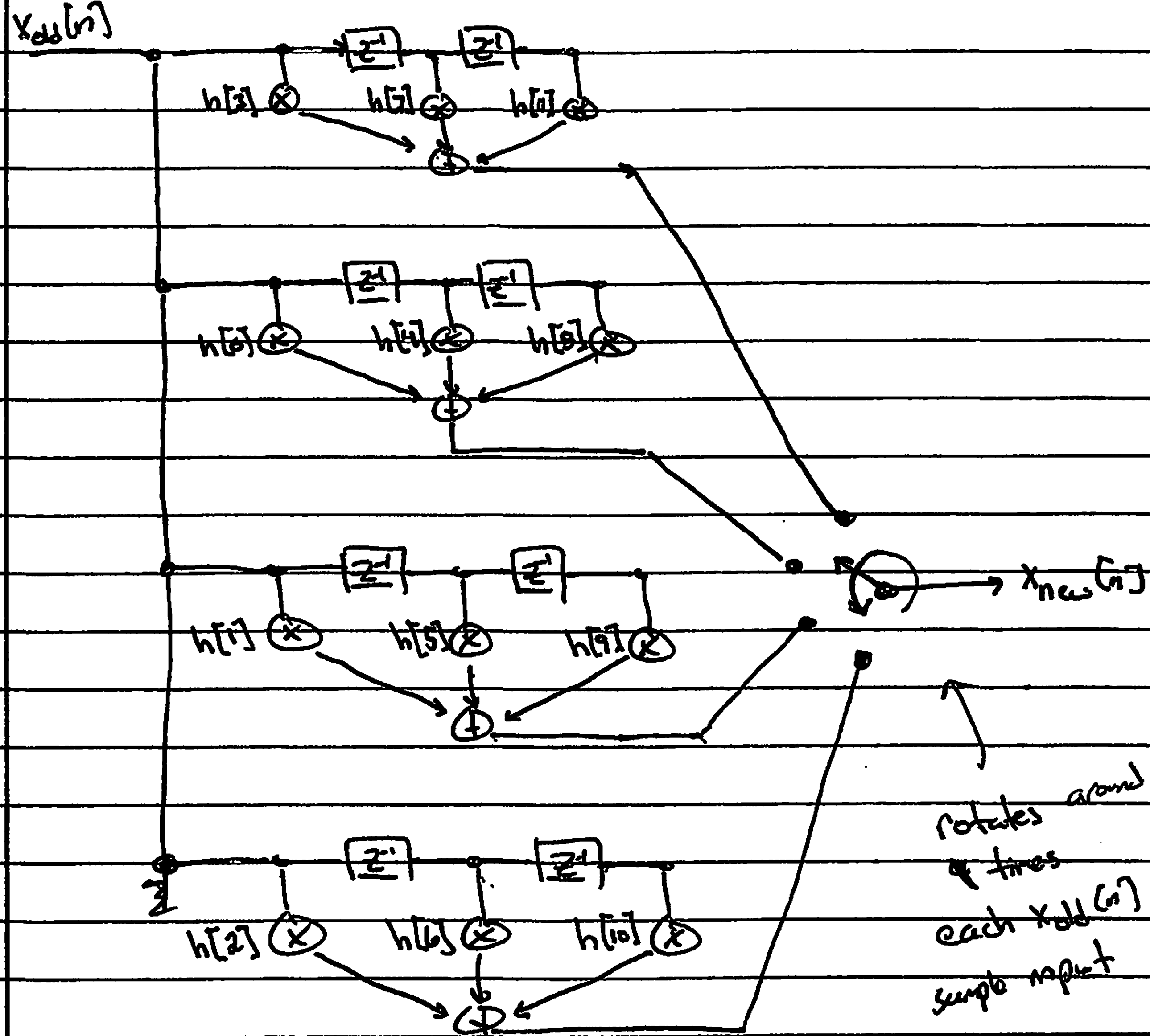
- 12 mults for each output $x_{new}[n]$
- 9 mults are always zero

With $M=4$, there are 4 different sets of coeff used to compute $x_{new}[n]$

$$x_{new}[0] = h[3]\,x_0[2] + h[7]\,x_0[1] + h[11]\,x_0[0]$$
$$x_{new}[1] = h[0]\,x_0[3] + h[4]\,x_0[2] + h[8]\,x_0[1]$$
$$x_{new}[2] = h[1]\,x_0[3] + h[5]\,x_0[2] + h[9]\,x_0[1]$$
$$x_{new}[3] = h[2]\,x_0[3] + h[6]\,x_0[2] + h[10]\,x_0[1]$$
$$x_{new}[4] = h[3]\,x_0[3] + h[7]\,x_0[2] + h[11]\,x_0[1]$$
$$\vdots$$

We can implement this with a bank of 4 sub-filters

see next page



$$H(z) = h[0] + h[4]z^{-1} + h[8]z^{-2}$$
$$+ \left[h[1] + h[5]z^{-1} + h[9]z^{-2}\right] z_c^{-1}$$
$$+ \left[h[2] + h[6]z^{-1} + h[10]z^{-2}\right] z_c^{-2}$$
$$+ \left[h[3] + h[7]z^{-1} + h[11]z^{-2}\right] z_c^{-3}$$

Note – $z^{-1} = z_c^{-4}$ & $z^{-2} = z_c^{-8}$

$$\Rightarrow H(z) = h[0] + h[4]z_c^{-4} + h[8]z_c^{-8}$$
$$+ h[1]z_c^{-1} + h[5]z_c^{-5} + h[9]z_c^{-9}$$
$$+ h[2]z_c^{-2} + h[6]z_c^{-6} + h[10]z_c^{-10}$$
$$+ h[3]z_c^{-3} + h[7]z_c^{-7} + h[11]z_c^{-11}$$

$$H(z) = \sum_{k=0}^{11} h[k] z_c^{-k}$$

12-Tap FIR filter.

What about decimation?