Basic FIR Digital Filter Structures

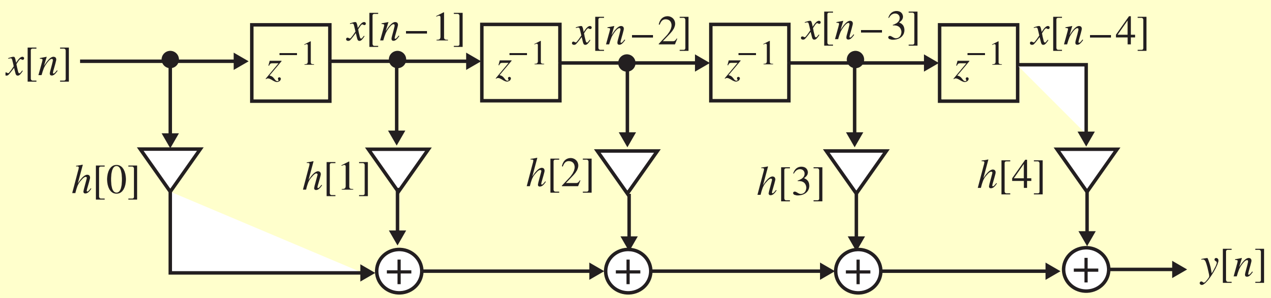
- A causal FIR filter of order N is characterized by a transfer function $H(z)$ given by
$$H(z) = \sum_{n=0}^{N} h[n] z^{-n}$$
which is a polynomial in z^{-1}
- In the time-domain the input-output relation of the above FIR filter is given by
$$y[n] = \sum_{k=0}^{N} h[k] x[n-k]$$

Direct Form FIR Digital Filter Structures

- An FIR filter of order N is characterized by $N+1$ coefficients and, in general, require $N+1$ multipliers and N two-input adders
- Structures in which the multiplier coefficients are precisely the coefficients of the transfer function are called **direct form** structures

Direct Form FIR Digital Filter Structures

- A direct form realization of an FIR filter can be readily developed from the convolution sum description as indicated below for $N = 4$



Direct Form FIR Digital Filter Structures

- An analysis of this structure yields

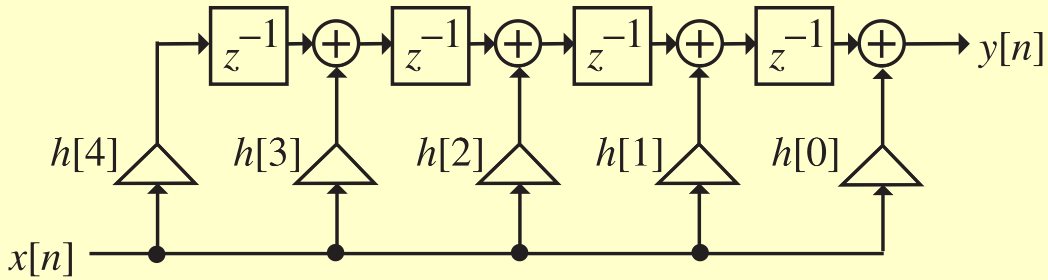
$$y[n] = h[0]x[n] + h[1]x[n-1] + h[2]x[n-2] + h[3]x[n-3] + h[4]x[n-4]$$

which is precisely of the form of the convolution sum description

- The direct form structure shown on the previous slide is also known as a **transversal filter**

Direct Form FIR Digital Filter Structures

- The transpose of the direct form structure shown earlier is indicated below



- Both direct form structures are canonic with respect to delays

Cascade Form FIR Digital Filter Structures

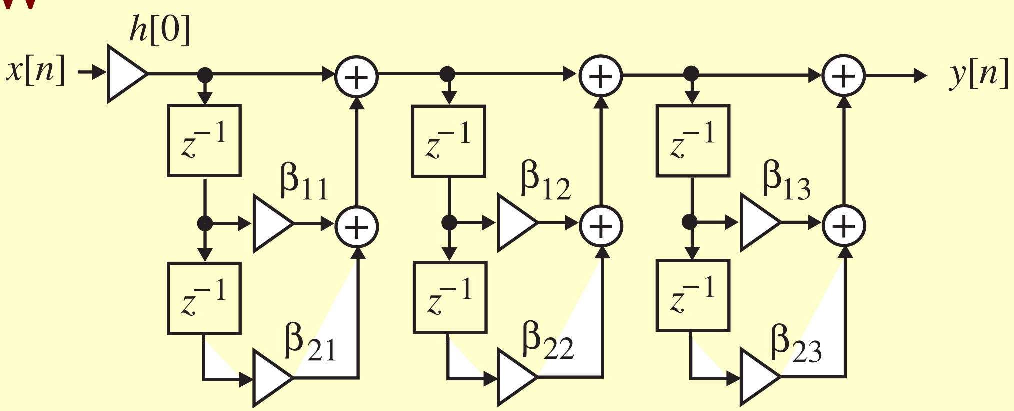
- A higher-order FIR transfer function can also be realized as a cascade of second-order FIR sections and possibly a first-order section
- To this end we express $H(z)$ as

$$H(z) = h[0]\prod_{k=1}^{K}(1+\beta_{1k}z^{-1}+\beta_{2k}z^{-2})$$

where $K = \frac{N}{2}$ if N is even, and $K = \frac{N+1}{2}$ if N is odd, with $\beta_{2K} = 0$

Cascade Form FIR Digital Filter Structures

- A cascade realization for $N = 6$ is shown below



- Each second-order section in the above structure can also be realized in the transposed direct form

Polyphase FIR Structures

- The polyphase decomposition of $H(z)$ leads to a parallel form structure
- To illustrate this approach, consider a causal FIR transfer function $H(z)$ with $N = 8$:

$$H(z) = h[0] + h[1]z^{-1} + h[2]z^{-2} + h[3]z^{-3} + h[4]z^{-4}$$
$$+ h[5]z^{-5} + h[6]z^{-6} + h[7]z^{-7} + h[8]z^{-8}$$

Polyphase FIR Structures

- $H(z)$ can be expressed as a sum of two terms, with one term containing the even-indexed coefficients and the other containing the odd-indexed coefficients:

$$
\begin{aligned}
H(z) &= (h[0] + h[2]z^{-2} + h[4]z^{-4} + h[6]z^{-6} + h[8]z^{-8}) \\
&\quad + (h[1]z^{-1} + h[3]z^{-3} + h[5]z^{-5} + h[7]z^{-7}) \\
&= (h[0] + h[2]z^{-2} + h[4]z^{-4} + h[6]z^{-6} + h[8]z^{-8}) \\
&\quad + z^{-1}(h[1] + h[3]z^{-2} + h[5]z^{-4} + h[7]z^{-6})
\end{aligned}
$$

Polyphase FIR Structures

- By using the notation

$$E_0(z) = h[0] + h[2]z^{-1} + h[4]z^{-2} + h[6]z^{-3} + h[8]z^{-4}$$

$$E_1(z) = h[1] + h[3]z^{-1} + h[5]z^{-2} + h[7]z^{-3}$$

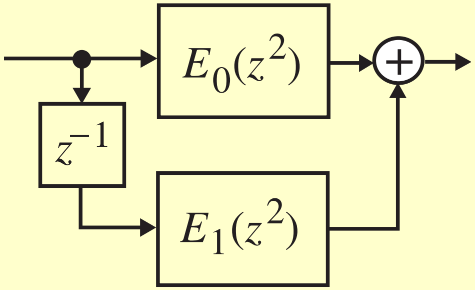
we can express $H(z)$ as

$$H(z) = E_0(z^2) + z^{-1}E_1(z^2)$$

- The above decompostion is more commonly known as the 2-branch **polyphase decomposition**

Polyphase FIR Structures

- A realization of $H(z)$ based on the 2-branch polyphase decomposition is thus as shown below



Polyphase FIR Structures

- In a similar manner, by grouping the terms in the original expression for $H(z)$, we can reexpress it in the form

$$H(z) = E_0(z^3) + z^{-1}E_1(z^3) + z^{-2}E_2(z^3)$$

where now

$$E_0(z) = h[0] + h[3]z^{-1} + h[6]z^{-2}$$

$$E_1(z) = h[1] + h[4]z^{-1} + h[7]z^{-2}$$

$$E_2(z) = h[2] + h[5]z^{-1} + h[8]z^{-2}$$

Polyphase FIR Structures

- The decomposition of $H(z)$ in the form

$$H(z) = E_0(z^2) + z^{-1}E_1(z^2)$$

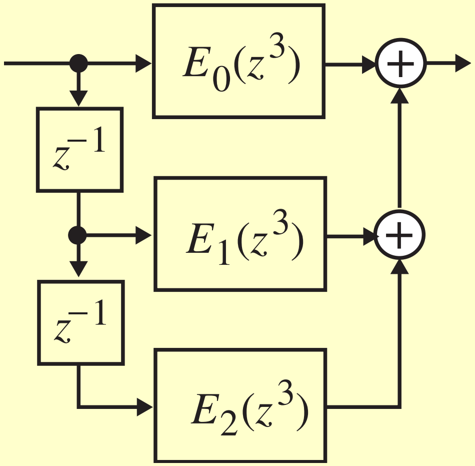
or

$$H(z) = E_0(z^3) + z^{-1}E_1(z^3) + z^{-2}E_2(z^3)$$

is more commonly known as the 3-branch **polyphase decomposition**

Polyphase FIR Structures

- A realization of $H(z)$ based on the 3-branch polyphase decomposition is thus as shown below



Polyphase FIR Structures

- In the general case, an L-branch polyphase decomposition of an FIR transfer function of order N is of the form

$$H(z) = \sum_{m=0}^{L-1} z^{-m} E_m(z^L)$$

where

$$E_m(z) = \sum_{n=0}^{\lfloor (N+1)/L \rfloor} h[Ln+m] z^{-m}$$

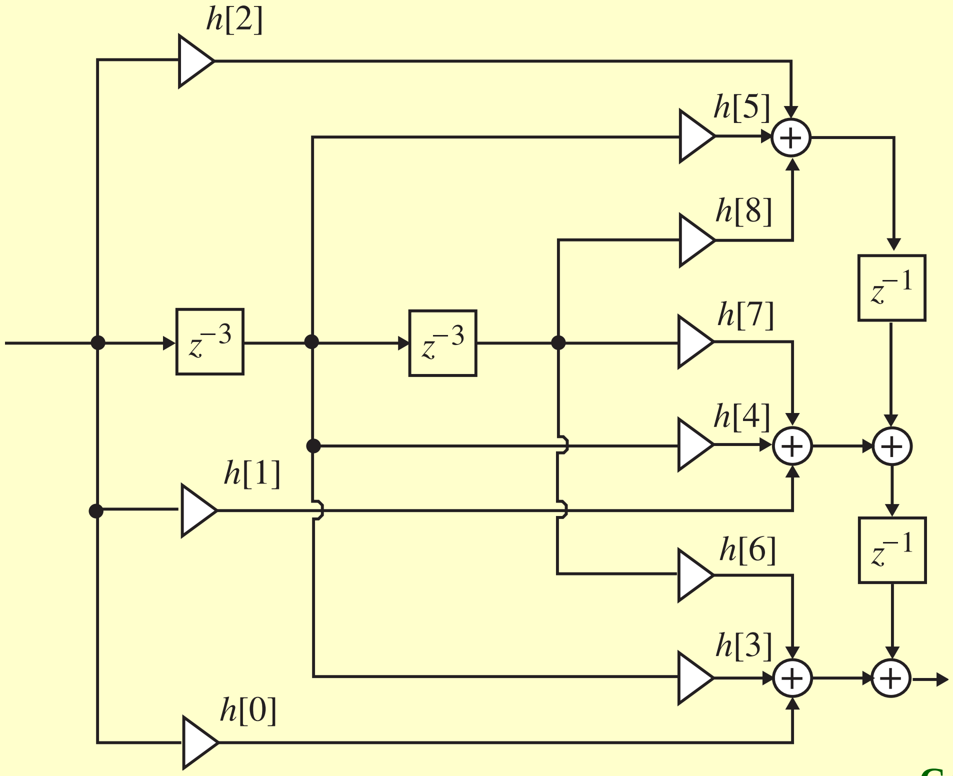
with $h[n]=0$ for $n > N$

Polyphase FIR Structures

- The subfilters $E_m(z^L)$ in the polyphase realization of an FIR transfer function are also FIR filters and can be realized using any methods described so far
- However, to obtain a canonic realization of the overall structure, the delays in all subfilters must be shared

Polyphase FIR Structures

- Figure below shows a canonic realization of a length-9 FIR transfer function obtained using delay sharing



Linear-Phase FIR Structures

- The symmetry (or antisymmetry) property of a linear-phase FIR filter can be exploited to reduce the number of multipliers into almost half of that in the direct form implementations
- Consider a length-7 Type 1 FIR transfer function with a symmetric impulse response:

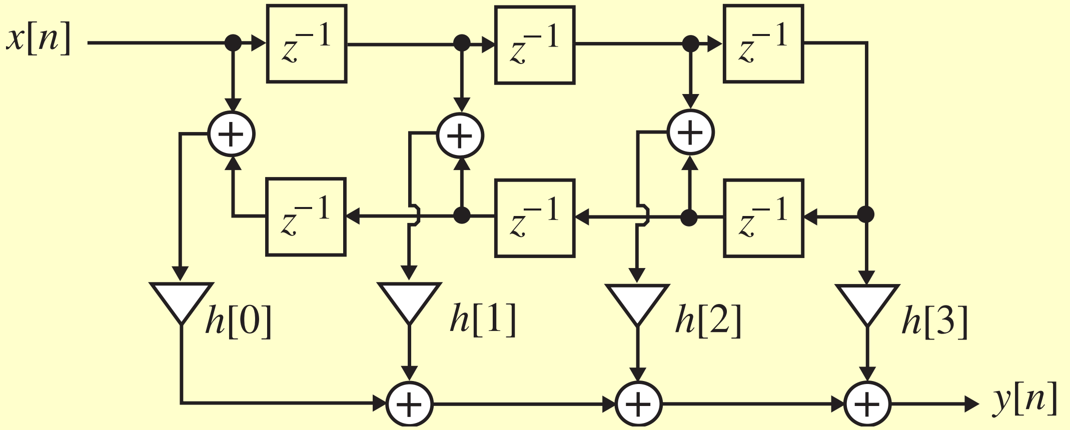
$$H(z) = h[0] + h[1]z^{-1} + h[2]z^{-2} + h[3]z^{-3} + h[2]z^{-4} + h[1]z^{-5} + h[0]z^{-6}$$

Linear-Phase FIR Structures

- Rewriting $H(z)$ in the form

$$H(z) = h[0](1+z^{-6}) + h[1](z^{-1}+z^{-5}) + h[2](z^{-2}+z^{-4}) + h[3]z^{-3}$$

we obtain the realization shown below



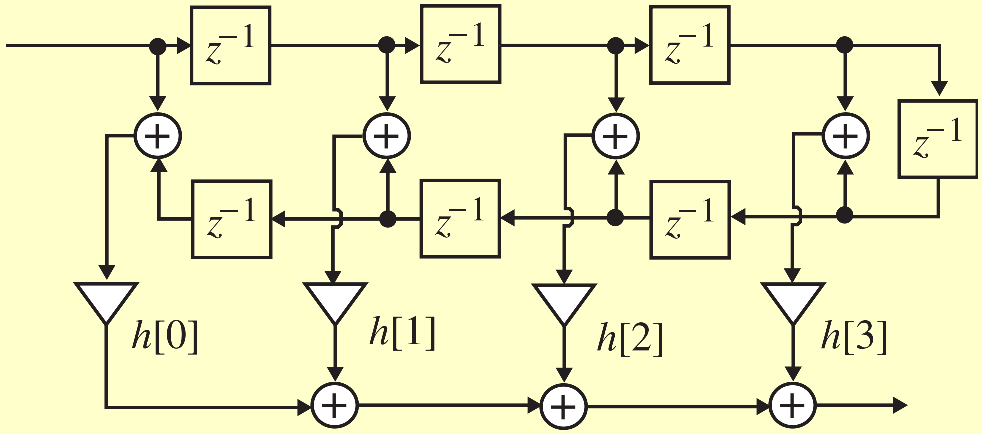
Linear-Phase FIR Structures

- A similar decomposition can be applied to a Type 2 FIR transfer function
- For example, a length-8 Type 2 FIR transfer function can be expressed as

$$H(z) = h[0](1 + z^{-7}) + h[1](z^{-1} + z^{-6}) + h[2](z^{-2} + z^{-5}) + h[3](z^{-3} + z^{-4})$$

- The corresponding realization is shown on the next slide

Linear-Phase FIR Structures



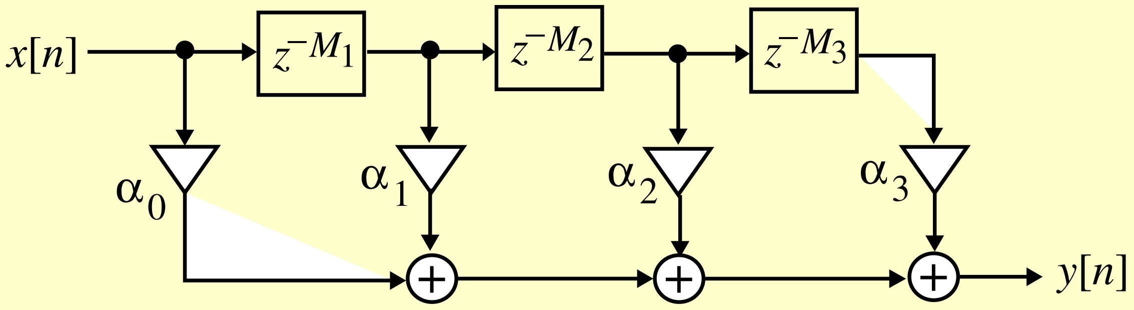
- Note: The Type 1 linear-phase structure for a length-7 FIR filter requires 4 multipliers, whereas a direct form realization requires 7 multipliers

Linear-Phase FIR Structures

- Note: The Type 2 linear-phase structure for a length-8 FIR filter requires 4 multipliers, whereas a direct form realization requires 8 multipliers
- Similar savings occurs in the realization of Type 3 and Type 4 linear-phase FIR filters with antisymmetric impulse responses

Tapped Delay Line

- In some applications, such as musical and sound processing, FIR filter structures of the form shown below are employed



Tapped Delay Line

- The structure consists of a chain of $M_1 + M_2 + M_3$ unit delays with taps at the input, at the end of first M_1 delays, at the end of next M_2 delays, and at the output
- Signals at these taps are then multiplied by constants α_0, α_1, α_2, and α_3 and added to form the output

Tapped Delay Line

- Such a structure is usually referred to as the tapped delay line
- The direct form FIR structure in slide no. 37 is seen to be a special case of the tapped delay line, where there is a tap after each unit delay