# Chapter-4: Discrete Filter Structures

## ❑ FIR Filter, Structures for FIR Filter:

### 1. Introduction:

- For the ***design of digital filters***, the ***system function H(z)*** or the ***impulse response h[n]*** must be specified. Then the ***digital filter structure*** can be implemented in ***hardware/software*** form by its ***difference equation*** obtained directly from H(z) or h[n].

- To implement the specified ***difference equation*** of the system, the required basic operations are ***addition, delay and multiplication by a constant***.

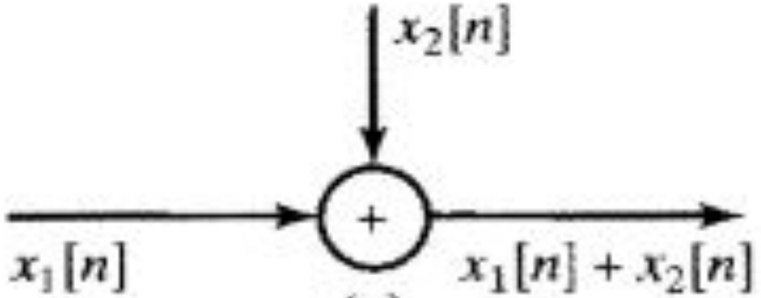
- The ***structures are derived*** on the basis of ***computational complexity***, ***ease of implementation of finite word length effect*** etc.

## 2. Block Diagram Representation:

- When the system function H(z) or the impulse response h[n] is specified then the digital filters can be implemented or realized using ***block diagram***. The following are the ***basic elements required for the implementation***.

### a. An adder:

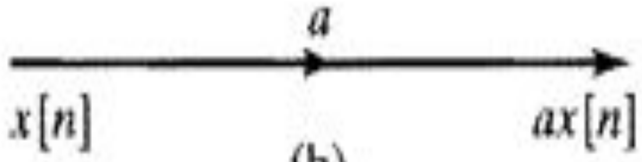
- It performs the addition of two signals.



(a)

### b. A Constant Multiplier:

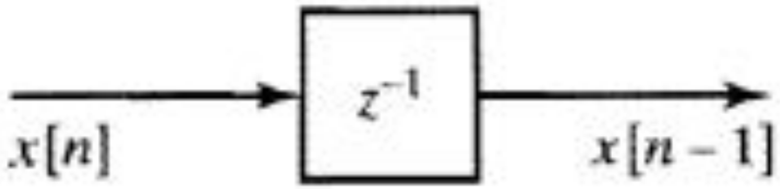
- It multiplies a signal with a constant ***'a'***.



(b)

c. **A Unit Delay Element:**

- It delays the input sequence by one element.



(c)

## 3. Advantages of Representing the Digital System (i.e. Filters) in Block Diagram Form:

i. The computation algorithm can be easily written just by inspection.
ii. The hardware requirement can be easily determined.
iii. The relationship between input and output can be easily determined.

## 4. Canonic and Non-canonic Structures:

- If the ***number of delays*** in the structure or realization block diagram is ***equal*** to the ***order of the difference equation or the order of the system function*** of a digital filter, then the structure is canonic otherwise non-canonic.

## 4. Structures for FIR Systems:

- A ***causal FIR system*** can be described by the ***difference equation***

$$y[n] = \sum_{k=0}^{M-1} b_k x[n-k] \qquad \text{.....i}$$

Or, equivalently, by the ***system function***

$$H(z) = \sum_{k=0}^{M-1} b_k z^{-k} \qquad \text{.....ii}$$

- If we replace $b_k$ by $h[k]$, we obtain

$$y[n] = \sum_{k=0}^{M-1} h[k]\, x[n-k] \qquad \text{.....iii}$$

( *convolution sum*)

Or, $$H(z) = \sum_{k=0}^{M-1} h[k]\, z^{-k} \qquad \text{.....iv}$$

- Therefore, we can write

$$h[n] = \begin{cases} b_n, & 0 \le n \le M-1 \\ 0\,, & Otherwise \end{cases} \qquad \text{.....v}$$

- **Note that FIR filter is called all-zero filter ( or comb filter)**.

- There are several methods for implementation of FIR system. They are:

a. Direct form structures
b. Cascade form structures
c. Frequency sampling structures
d. Lattice structures
e. Linear phase structures

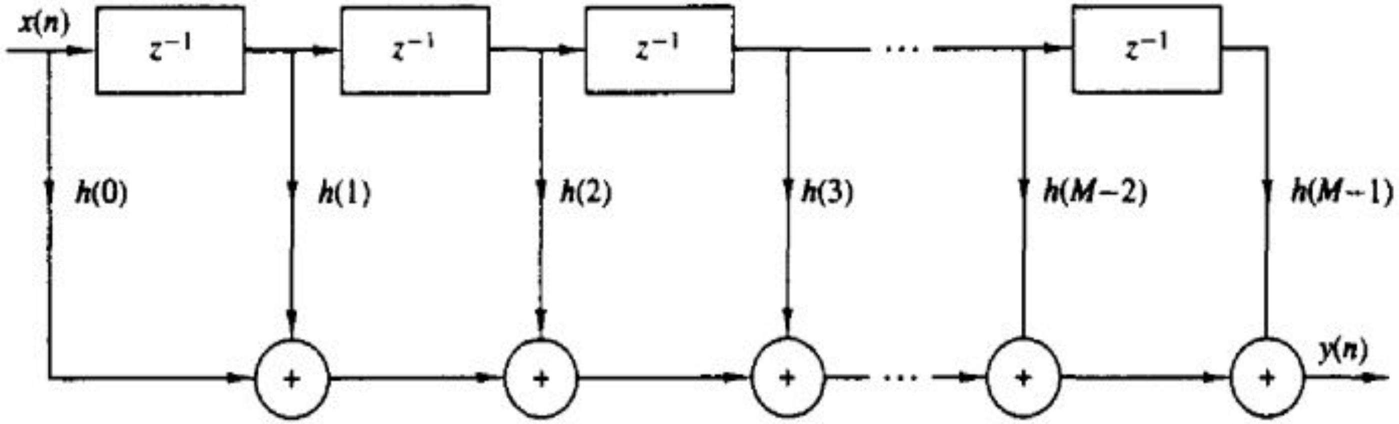
## a. Direct Form Structures:

- The direct-form realization follows immediately from the non-recursive difference equation or, equivalently, by the convolution summation

$$y[n] = \sum_{k=0}^{M-1} h[k]\, x[n-k] \qquad \text{.....i}$$

Or, $y[n] = h[0]\, x[n] + h[1]\, x[n-1] + \cdots + h[M-1]\, x[n-(M-1)]$ .....ii

- The direct form structure can be realized as shown in figure below:




- Because of the ***chain of delay elements across the top of the diagram***, this structure is also referred to as a ***tapped delay line structure or a transversal filter structure***.

## b. Cascade Form Structures:

- The ***cascade* form structure** of FIR system is obtained by ***factoring the polynomial system function*** as:

$$H(z) = \prod_{k=1}^{K} H_k(z)$$

where

$$H_k(z) = b_{k0} + b_{k1}z^{-1} + b_{k2}z^{-2} \qquad k = 1, 2, \ldots, K$$

where, $K$= integer part of $\frac{M+1}{2}$.

- If ***M is odd*** then $\boldsymbol{K = \frac{M+1}{2}}$.
- If ***M is even*** then ***K***= $\boldsymbol{\frac{M}{2}}$ with $\boldsymbol{b_{k2} = 0}$.

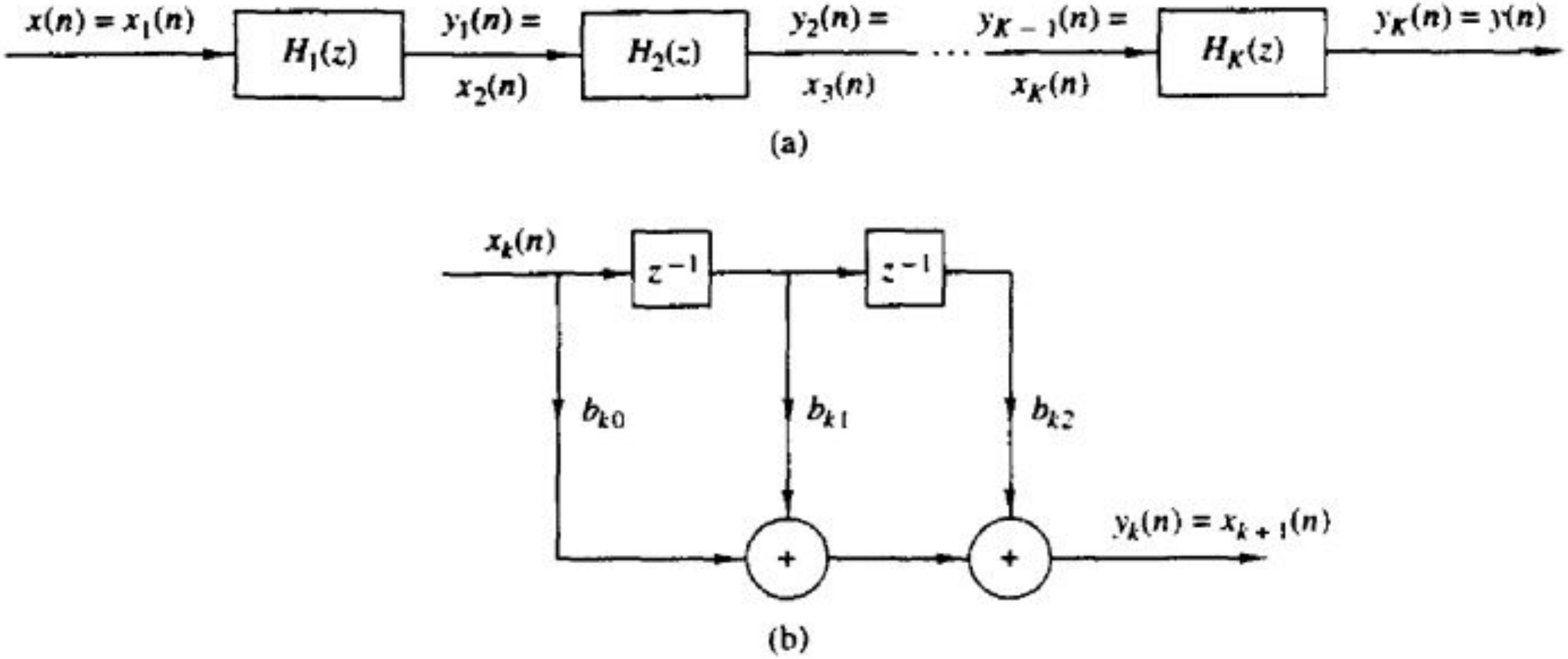



Fig: Cascade realization of an FIR system

Example 5.5: Obtain the direct-form and cascade-form realizations for the transfer function of an FIR system given by

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

(b) $H(z) = 1+\frac{5}{2}z^{-1}+2z^{-2}+2z^{-3}$

Solution:

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

(i) Direct-form realization:

→ For this, we have

$$H(z) = (1+2z^{-1}-z^{-2})(1+z^{-1}-z^{-2})$$
$$= 1+z^{-1}-z^{-2}+2z^{-1}+2z^{-2}-2z^{-3}-z^{-2}-z^{-3}+z^{-4}$$

$$\therefore H(z) = 1+3z^{-1}-3z^{-3}+z^{-4}$$

The direct-form realization is shown in fig 5.22.

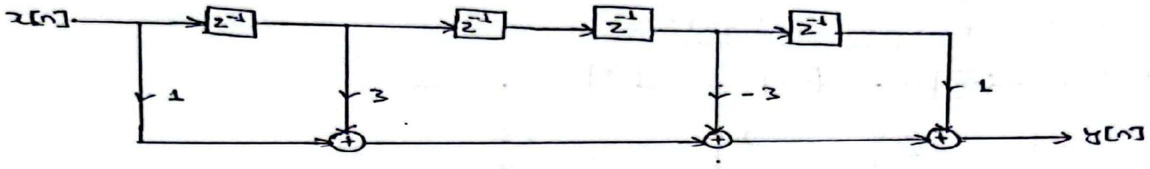



Fig 5.22: Direct-form realization

(ii) cascade-form realization:

→ For this, we have

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

$$H(z) = H_1(z)\,H_2(z)$$

where, $H_1(z) = (1+2z^{-1}-z^{-2})$

and $H_2(z) = (1+z^{-1}-z^{-2})$

The cascade-form realization is shown in fig 5.23.

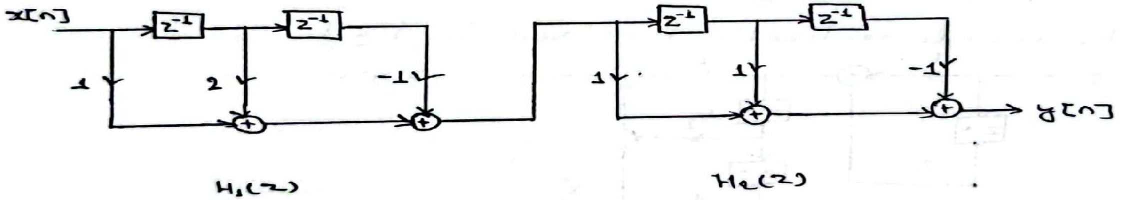
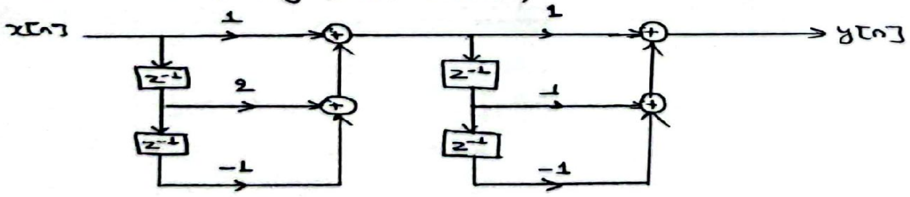



Fig 5.23: cascade-for realization

**Note:** Fig 5.23 can be redrawn for cascade-form realization as shown in fig 5.24 below.




$H_1(z)$ $H_2(z)$

Fig 5.24

(b) $H(z) = 1 + \frac{5}{2} z^{-1} + 2z^{-2} + 2z^{-3}$

(i) Direct-Form realization:

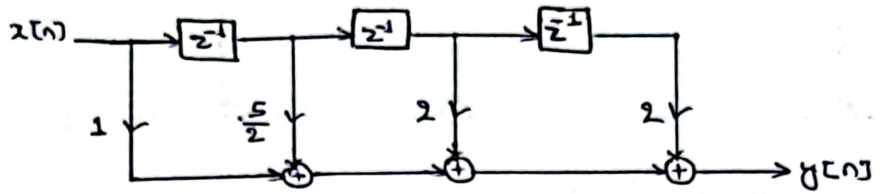



Fig 5.25: Direct-form realization

(ii) Cascade-form realization:

→ For this, we have

$$H(z) = 1 + \frac{5}{2} z^{-1} + 2z^{-2} + 2z^{-3}$$

or $H(z) = (1+2z^{-1})\left(1 + \frac{1}{2} z^{-1} + z^{-2}\right) = H_1(z)\, H_2(z)$

$\therefore$ $H_1(z) = (1+2z^{-1})$

and $H_2(z) = \left(1 + \frac{1}{2} z^{-1} + z^{-2}\right)$

The cascade-realization is shown in fig

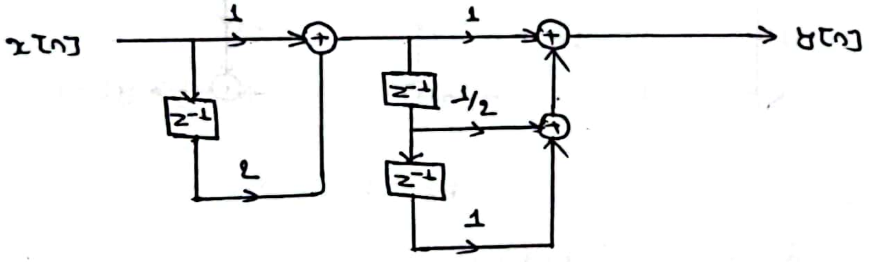



Fig 5.26: cascade-form realization

## c. Frequency-Sampling Structures:

- The ***frequency-sampling realization*** is an alternative structure for an FIR filter in which the ***parameters that characterize the filter are the values of the desired frequency response instead of the impulse response h(n)***.

- To ***derive the frequency sampling structure***, we specify ***the desired frequency response at a set of equally spaced frequencies***, namely

$$\omega_k = \frac{2\pi}{M}(k+\alpha) \qquad k = 0,1,\dots\dots\frac{M-1}{2} \text{ for } \boldsymbol{M} \text{ odd.}$$

$$k = 0,1,\dots\dots\frac{M}{2}-1 \text{ for } \boldsymbol{M} \text{ even.}$$

$$\alpha = 0 \text{ or } \frac{1}{2}.$$

and ***solve for the*** $\boldsymbol{h[n]}$.

- The ***frequency response*** is

$$\mathbf{H}(\boldsymbol{e^{j\omega}}) = \boldsymbol{H(\omega)} = \sum_{\boldsymbol{n=0}}^{\boldsymbol{M-1}} \boldsymbol{h[n]\, e^{-j\omega n}}$$

and values of $\mathbf{H}(\boldsymbol{e^{j\omega}})$ at $\omega_k = \frac{2\pi}{M}(k+\alpha)$ are

$$H(k+\alpha) = H\left(\frac{2\pi}{M}(k+\alpha)\right)$$

$$= \sum_{n=0}^{M-1} h(n) e^{-j2\pi(k+\alpha)n/M} \qquad k = 0, 1, \ldots, M-1$$

- The set of values $\frac{2\pi}{M}(k+\alpha)$ are called the ***frequency samples of*** $\mathrm{H}(e^{j\omega})$. In the case, where $\alpha = 0$, $\{H(k)\}$ corresponds to the $M$ **-point DFT of** $\{h[n]\}$.

- The ***impulse response*** $h[n]$ of above equation is

$$h(n) = \frac{1}{M} \sum_{k=0}^{M-1} H(k+\alpha) e^{j2\pi(k+\alpha)n/M} \qquad n = 0, 1, \ldots, M-1$$

When $\alpha = 0$, above equation is simply the ***IDFT of*** $H(k)$.

- Now, if we use above equation to substitute for $h[n]$ in the z-transform $H(z)$, we have

$$H(z) = \sum_{n=0}^{M-1} h(n) z^{-n}$$

$$= \sum_{n=0}^{M-1} \left[ \frac{1}{M} \sum_{k=0}^{M-1} H(k+\alpha) e^{j2\pi(k+\alpha)n/M} \right] z^{-n}$$

$$H(z) = \sum_{k=0}^{M-1} H(k+\alpha) \left[ \frac{1}{M} \sum_{n=0}^{M-1} (e^{j2\pi(k+\alpha)/M} z^{-1})^n \right]$$

$$= \frac{1 - z^{-M} e^{j2\pi\alpha}}{M} \sum_{k=0}^{M-1} \frac{H(k+\alpha)}{1 - e^{j2\pi(k+\alpha)/M} z^{-1}}$$

- Thus, the ***system function*** $\boldsymbol{H(z)}$ ***is characterized by the set of frequency samples*** $\boldsymbol{H(k+\alpha)}$ ***instead of*** $\{\boldsymbol{h}[\boldsymbol{n}]\}$.

- We view this FIR filter realization as a ***cascade of two filters*** [i.e., $\boldsymbol{H(z)} = \boldsymbol{H_1(z).H_1(z)}$]

$$H_1(z) = \frac{1}{M}\left(1 - z^{-M} e^{j2\pi\alpha}\right) \qquad \text{( \textit{\textbf{all zero filter or comb filter}} )}$$

Its ***zeros*** are located at equally spaced points on the unit circle at

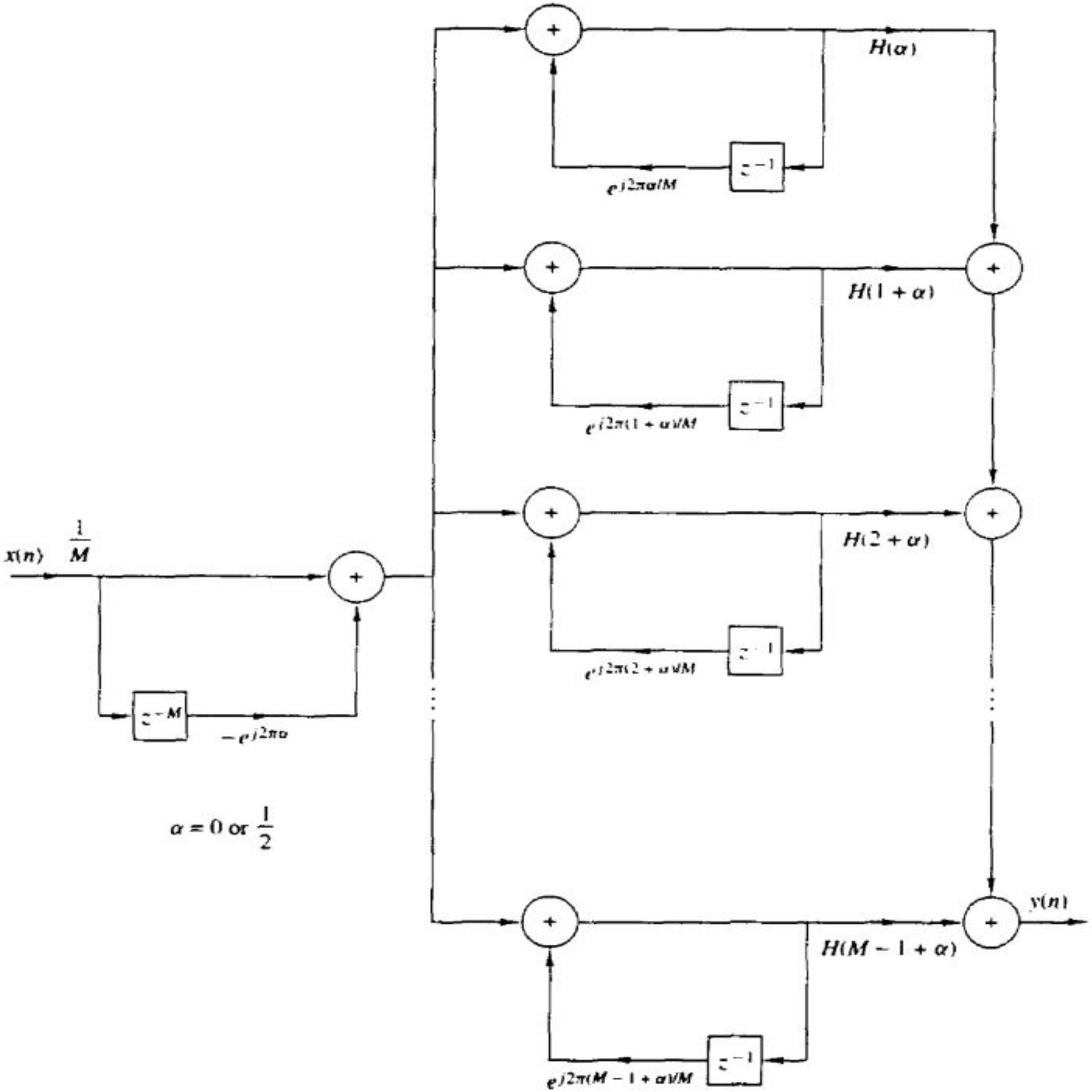
$$z_k = e^{j2\pi(k+\alpha)/M}, \quad k = 0, 1, \dots \dots M-1$$

and $$H_2(z) = \sum_{k=0}^{M-1} \frac{H(k+\alpha)}{1 - e^{j2\pi(k+\alpha)/M} z^{-1}}$$

consists ***parallel banks of single-pole filters*** with resonant frequencies

$$p_k = e^{j2\pi(k+\alpha)/M}, \quad k = 0, 1, \dots \dots M-1$$

- The cascade realization is shown in figure below:



x(n)
1/M
z^{-M}
-e^{j2πα}
α = 0 or 1/2
H(α)
e^{j2πα/M}
z^{-1}
H(1 + α)
e^{j2π(1 + α)/M}
H(2 + α)
e^{j2π(2 + α)/M}
H(M - 1 + α)
e^{j2π(M - 1 + α)/M}
y(n)

- For ***linear phase***, $H(k) = H^*(M-k)$ for $\alpha = 0$, and

$$H\left(k+\frac{1}{2}\right) = H^*\left(M-k-\frac{1}{2}\right) \quad \text{for } \alpha = \frac{1}{2}$$

As a result, ***a pair of single pole filters can be combined to form a single two-pole filter*** with ($\boldsymbol{\alpha = 0}$).

$$H_2(z) = \frac{H(0)}{1-z^{-1}} + \sum_{k=1}^{(M-1)/2} \frac{A(k)+B(k)z^{-1}}{1-2\cos(2\pi k/M)z^{-1}+z^{-2}}$$

$$H_2(z) = \frac{H(0)}{1-z^{-1}} + \frac{H(M/2)}{1+z^{-1}} + \sum_{k=1}^{(M/2)-1} \frac{A(k)+B(k)z^{-1}}{1-2\cos(2\pi k/M)z^{-1}+z^{-2}}$$

where, by definition

$$A(k) = H(k) + H(M-k)$$

$$B(k) = H(k)e^{-j2\pi k/M} + H(M-k)e^{j2\pi k/M}$$

Similar expressions can be obtained for $\boldsymbol{\alpha = \frac{1}{2}}$.

## d. Lattice Structures:

- ***Lattice filters*** are used extensively in ***digital speech processing*** and in the implementation of ***adaptive filters***. Let an FIR filter with system function

$$\boldsymbol{H(z) = A_m(z), m = 0, 1, \dots\dots, M-1}$$

$$\boldsymbol{H(z) = A_m(z) = 1 + \sum_{k=1}^{m} a_m(k) z^{-k}}, \quad \boldsymbol{m \geq 1} \qquad \text{.....1}$$

with $\boldsymbol{A_0(z) = 1}$, and impulse response

$$h[n] = \begin{cases} 1, & n = 0 \\ a_m(n), & n = 1,2,\dots,m \end{cases}$$

OR,

$$Y(z) = X(z)\left[\, 1 + \sum_{k=1}^{m} a_m(k) z^{-k} \,\right]$$

$$Y(z) = X(z) + X(z) \sum_{k=1}^{m} a_m(k) z^{-k}$$

- Taking inverse z-transform, we get

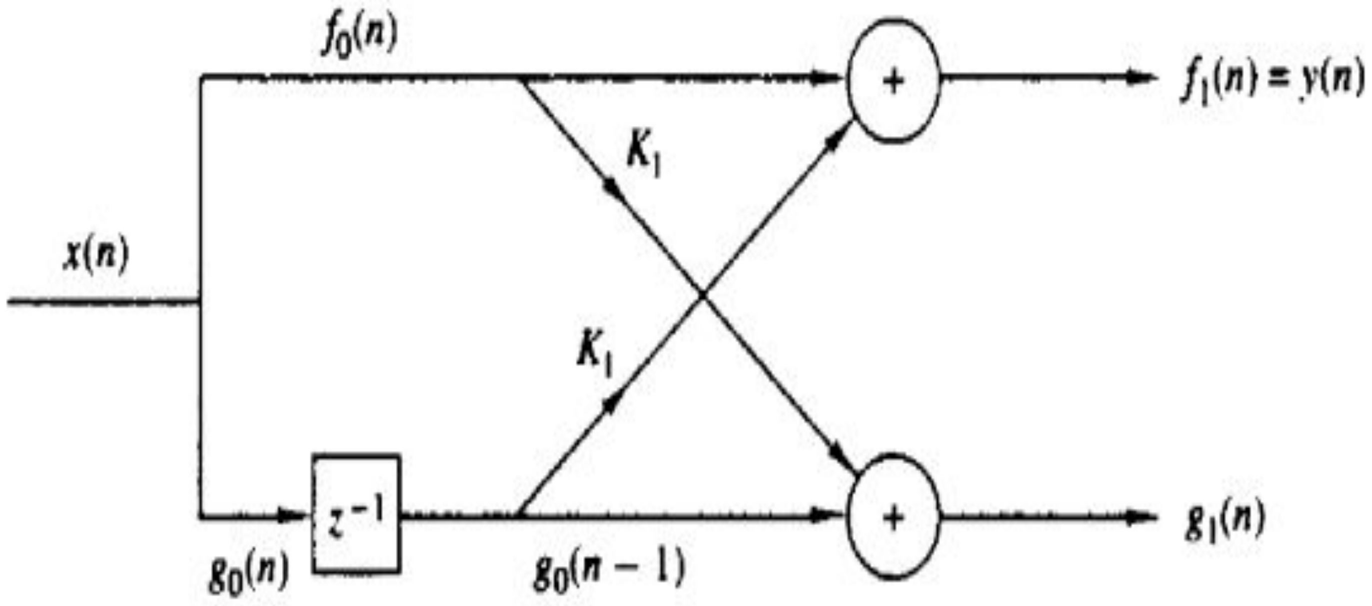
$$\boldsymbol{y[n] = x[n] + \sum_{k=1}^{m} a_m(k) x[n-k]} \qquad \text{.....2}$$

***( m is the degree of polynomial )***

- For $\boldsymbol{m = 1}$:
- Equation 2 reduces to

$$\boldsymbol{y[n] = x[n] + a_1(1) x[n-1]} \qquad \text{.....3}$$

This ***single stage lattice filter*** can be realized as shown in figure below:




- From above figure, we have

$$x[n] = f_0[n] = g_0[n] \qquad .....4$$

$$y[n] = f_1[n] = f_0[n] + k_1\, g_0[n-1] = x[n] + k_1 x[n-1] \qquad .....5$$

$$g_1[n] = k_1 f_0[n] + g_0[n-1] = k_1 x[n] + x[n-1] \qquad .....6$$

- Comparing equations 3 and 4, we get

$$a_1(0) = 1, \quad a_1(1) = k_1 \qquad .....7$$

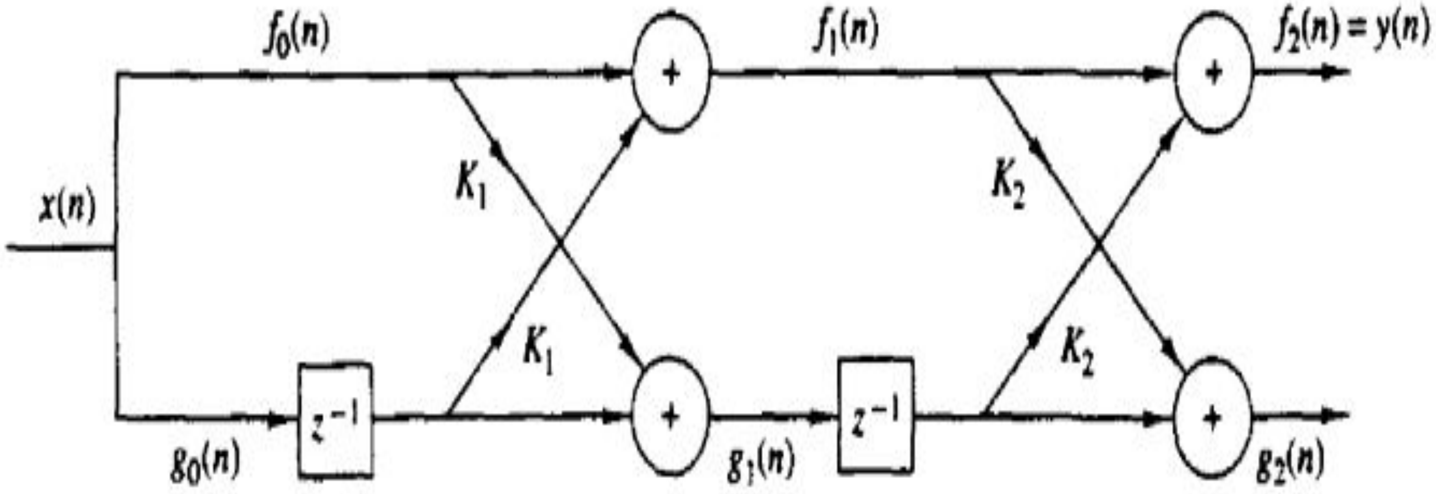
- Where, $k_1$ is ***lattice coefficient***. Similarly, $a_1(0)$ and $a_1(1)$ are ***direct form coefficients.***

- For $m = 2$:

- Equation 2 reduces to

$$y[n] = x[n] + a_2(1)x[n-1] + a_2(2)x[n-2] \qquad .....8$$

the ***two-stage lattice structure*** for this is shown in figure below:



- From above figure, we have

$$\boldsymbol{y[n] = f_2[n] = f_1[n] + k_2 g_1[n-1]}$$

But, $f_1[n] = f_0[n] + k_1 g_0[n-1] = x[n] + k_1 x[n-1]$

and, $g_1[n-1] = k_1 f_0[n-1] + g_0\,[n-2] = k_1 x[n-1] + x[n-2]$

therefore, $y[n] = f_2[n] = x[n] + k_1 x[n-1] + k_2\{k_1 x[n-1] + x[n-2]\}$

$$\boldsymbol{y[n] = f_2[n] = x[n] + k_1(1+k_2)x[n-1] + k_2 x[n-2]} \quad .....9$$

- From equations 8 and 9, we get

$$\boldsymbol{a_2(0) = 1, \quad a_2(1) = k_1(1+k_2), \quad a_2(2) = k_2}$$

- Also, from above figure

$$\boldsymbol{g_2[n] = k_2 f_1[n] + g_1[n-1]}$$

- Therefore,

$$g_2[n] = k_2x[n] + k_1(1 + k_2)x[n-1] + x[n-2] \quad .....10$$

Note that two sets of filter coefficients in $f[n]$ and $g[n]$ are in reverse order.

- For $m = M - 1$:
- For M-1 stage filter, we have

$$f_0(n) = g_0(n) = x(n)$$

$$f_m(n) = f_{m-1}(n) + K_m g_{m-1}(n-1) \qquad m = 1,2,\dots\dots,M-1 \quad .....11$$

$$g_m(n) = K_m f_{m-1}(n) + g_{m-1}(n-1) \qquad m = 1,2,\dots\dots,M-1$$

- The output of M-1 stage is

$$y[n] = f_{M-1}[n] \quad .....12$$

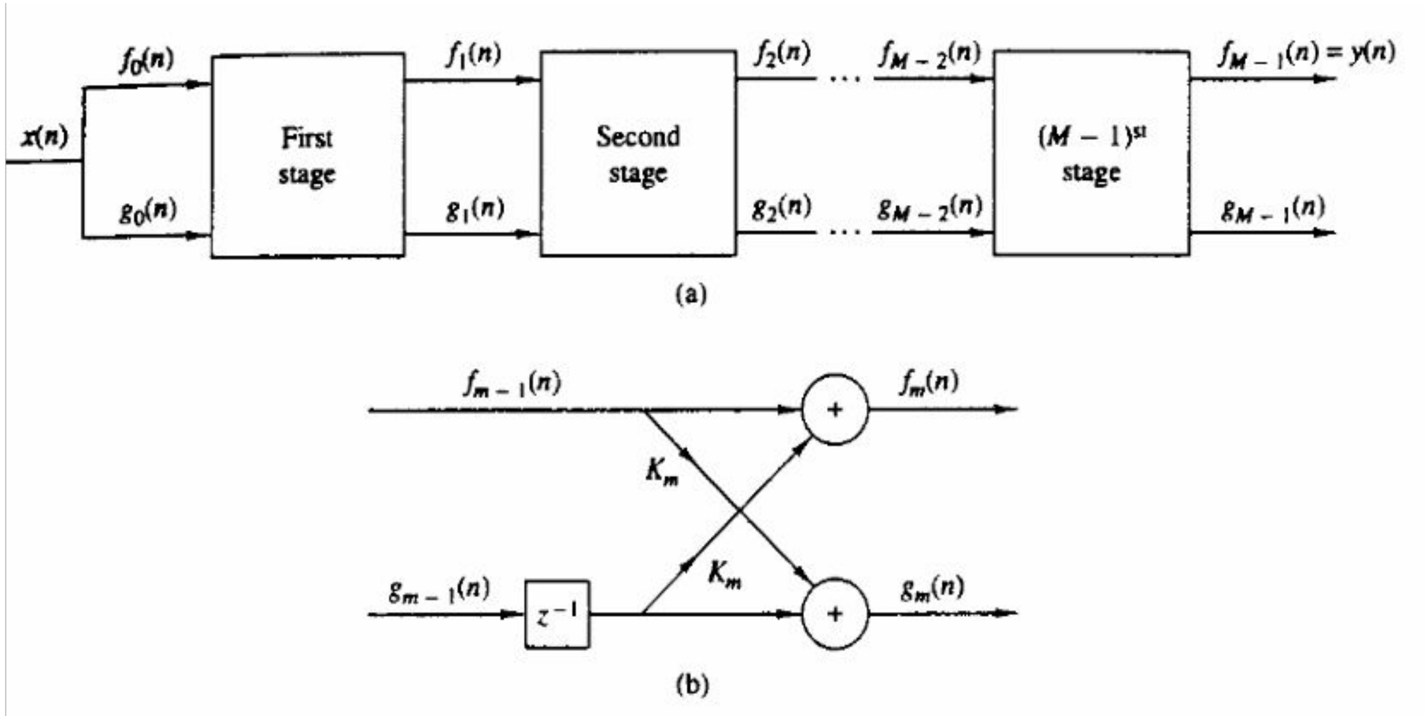



Fig: $(M-1)$-stage lattice filter.

- **Conversion of Lattice Coefficients to Direct Form Coefficients:**

➢ In general,

$$a_m(0) = 1$$
$$a_m(m) = k_m$$
$$a_m(k) = a_{m-1}(k) + a_m(m)\, a_{m-1}(m-k)$$

where, $1 \le k \le m-1$ and $m = 1,2, \dots \dots, M-1$

- **Conversion of Direct Form Coefficients to Lattice Coefficients :**

➢ In general,

$$a_m(0) = 1$$

$$k_m = a_m(m)$$

$$a_m(k) = a_{m-1}(k) + a_m(m)\, a_{m-1}(m-k)$$

$$a_{m-1}(k) = \frac{a_m(k) - a_m(m) a_m(m-k)}{1 - \{a_m(m)\}^2}$$

where, $1 \leq k \leq m-1$ and $m = 1,2, \dots \dots, M-1$

## ❑ IIR Filter, Structures for IIR Filter:

- Causal IIR systems are characterized by the difference equation as:

$$\boldsymbol{y[n] = -\sum_{k=1}^{N} a_k y[n-k] + \sum_{k=0}^{M} b_k x[n-k]} \quad \text{.....1}$$

Or, $$\sum_{k=0}^{N} a_k y[n-k] = \sum_{k=0}^{M} b_k x[n-k]\,,\ a_0 = 1$$

- Taking z-transform on both sides of equation 1, we get

$$Y(z) = -\sum_{k=1}^{N} a_k z^{-k} Y(z) + \sum_{k=0}^{M} b_k z^{-k} X(z)$$

$$Y(z)[1 + \sum_{k=1}^{N} a_k z^{-k}] = \sum_{k=0}^{M} b_k z^{-k} X(z)$$

$$\boldsymbol{H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1+\sum_{k=1}^{N} a_k z^{-k}}} \quad \text{.....2}$$

where, $H(z)$ = system function, $a_0 \neq 0$ and $M \leq N$

- From 1 and 2, we observe that the ***realization of IIR systems, i. e. filters , involves a recursive computational algorithm***.

- **Notes:**

## A. Non-recursive and Recursive Systems:

- **Non-recursive System:**
- If the ***output of a system is the function of the present and past values of the inputs*** only then ***the system is known as non-recursive system***. Mathematically,

$$\boldsymbol{y[n] = F\{x[n], x[n-1], \dots \dots x[n-M]\}} \qquad \text{.....1}$$

- A ***causal FIR system is non-recursive*** system. consider a causal FIR system,

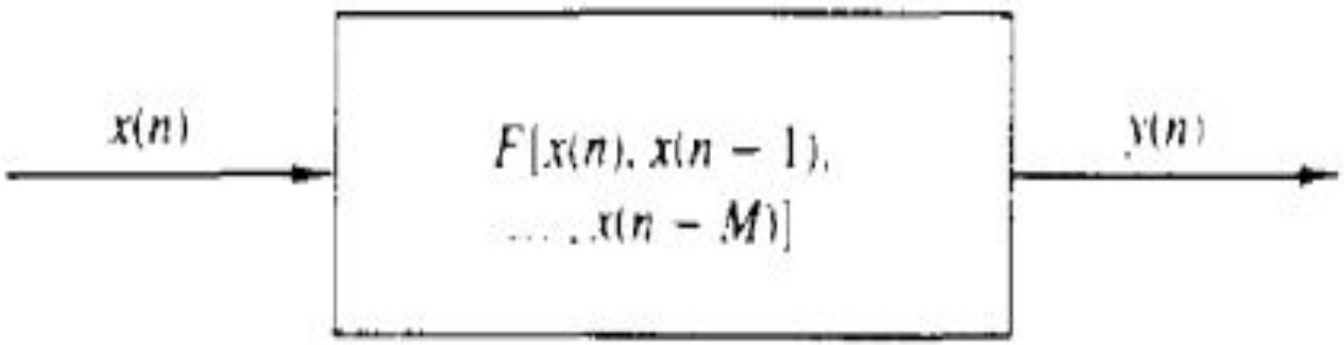
$$\boldsymbol{y[n] = \sum_{k=0}^{M-1} h[k]x[n-k]} \qquad \text{.....2}$$

where the output of the system is the function of present and past inputs. Thus, the causal FIR system is non-recursive.

- A ***non-recursive system*** can be represented in terms of ***difference equation*** as

$$\boldsymbol{y[n] = \sum_{k=0}^{M} b_k x[n-k]} \quad .....3$$

- Non-recursive systems ***does not have feedback path***.




- **Recursive System:**

- If the ***output of a system is the function of the present and past values of the inputs as well past outputs*** then ***the system is known as recursive system***. Mathematically,

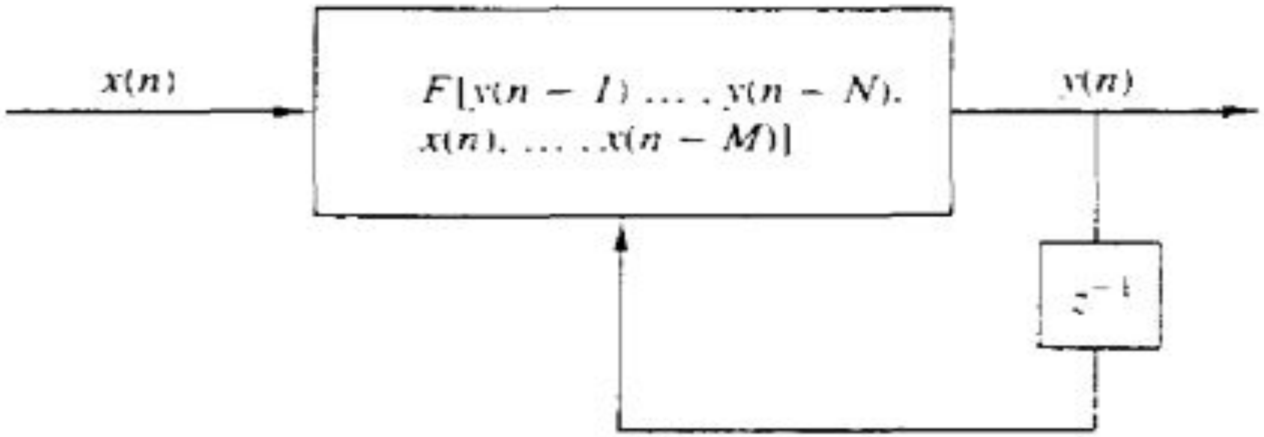
$$\boldsymbol{y[n] = F\{y[n-1], \dots \dots y[n-N], x[n], x[n-1], \dots \dots x[n-M]\}} \quad .....4$$

( causal recursive system)

- A ***recursive LTI system*** is characterized by ***difference equation*** as

$$y[n] = -\sum_{k=1}^{N} a_k y[n-k] + \sum_{k=0}^{M} b_k x[n-k] \qquad .....5$$

- ***Recursive realization reduces*** the ***memory requirements, additions, and multiplications***.
- Recursive systems ***have feedback path***.



## B. FIR ( Finite-Duration Impulse Response ) and IIR (Infinite-Duration Impulse Response ) System:

- Let h[n] be the impulse response of a LTI system. Then LTI system can be subdivided into two types

1. **FIR ( Finite-Duration Impulse Response ) System, and**
2. **IIR ( Infinite-Duration Impulse Response ) System**

### 1. FIR System:

- For causal FIR system, the convolution sum formula is

$$\boldsymbol{y[n] = \sum_{k=0}^{M-1} h[k]x[n-k]} \qquad .....1$$

An ***FIR system has finite memory of length M and non-recursive***.

- The difference equation of FIR system is

$$\boldsymbol{y[n] = \sum_{k=0}^{M} b_k x[n-k]} \qquad .....2$$

**2. IIR System:**

- For causal IIR system, the **convolution sum** formula is

$$\boldsymbol{y[n] = \sum_{k=0}^{\infty} h[k]x[n-k]} \qquad \text{.....1}$$

An ***FIR system has finite memory of length M and non-recursive***.

- The difference equation of IIR system is

$$\mathbf{y[n]} = -\sum_{k=1}^{N} \boldsymbol{a_k y[n-k]} + \sum_{k=0}^{M} \boldsymbol{b_k x[n-k]} \qquad \text{.....2}$$

- An ***IIR system cannot be realized using convolution sum*** but it is ***realized using difference equation (LCCDE )***.

## 1. Methods for the Implementation of IIR Systems:

a. Direct form structures
b. Cascade form structures
c. Parallel form structures
d. Lattice and lattice-ladder structures

### a. Direct Form Structures:

- The ***system function for IIR*** system is

$$H(z)=\frac{Y(z)}{X(z)}=\frac{\sum_{k=0}^{M} b_k z^{-k}}{1+\sum_{k=1}^{N} a_k z^{-k}}\,,\ a_0=1 \qquad \text{.....1}$$

- Since, the ***multiplier coefficients*** ($a_k$ and $b_k$) in the structures are exactly the ***coefficients of the system function***, they are called ***direct form structures***.
- Direct form structures can be studied under:

i. Direct form I structure
ii. Direct form II structure

## i. Direct form I structure:

- We know that the multiplier coefficients are the coefficients of system function

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1+\sum_{k=1}^{N} a_k z^{-k}}, a_0 = 1 \qquad .....1$$

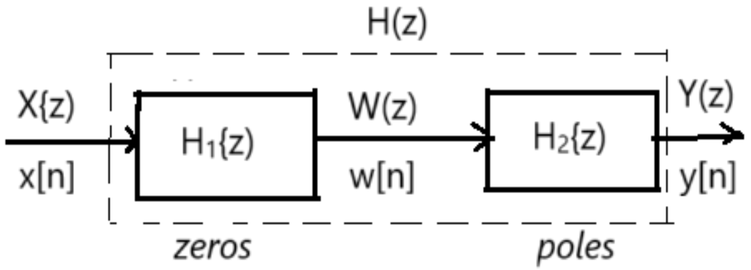
Or, $$H(z) = H_1(z).H_2(z)$$

where, $H_1(z) = \sum_{k=0}^{M} b_k z^{-k}$ , all- zero system (non-recursive) .....2

and $H_2(z) = \frac{1}{1+\sum_{k=1}^{N} a_k z^{-k}}$ , all-pole system (recursive) .....3

- Then the ***direct form I structure*** is obtained by ***cascading the structures for $H_1(z)$ and $H_2(z)$***.

- **Note:** $H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1-\sum_{k=1}^{N} a_k z^{-k}}, a_0 = 1$ (in Oppenheim book)




- From figure, we have

$$\frac{W(z)}{X(z)} = H_1(z)$$
$$\frac{W(z)}{X(z)} = \sum_{k=0}^{M} b_k z^{-k}$$
$$W(z) = X(z) \sum_{k=0}^{M} b_k z^{-k}$$

- Or,

$$W(z) = b_0\, X(z) + b_1 z^{-1} X(z) + b_2 z^{-2} X(z) + \cdots + b_M z^{-M} X(z) \qquad .....1$$

- Taking inverse z-transform, we get

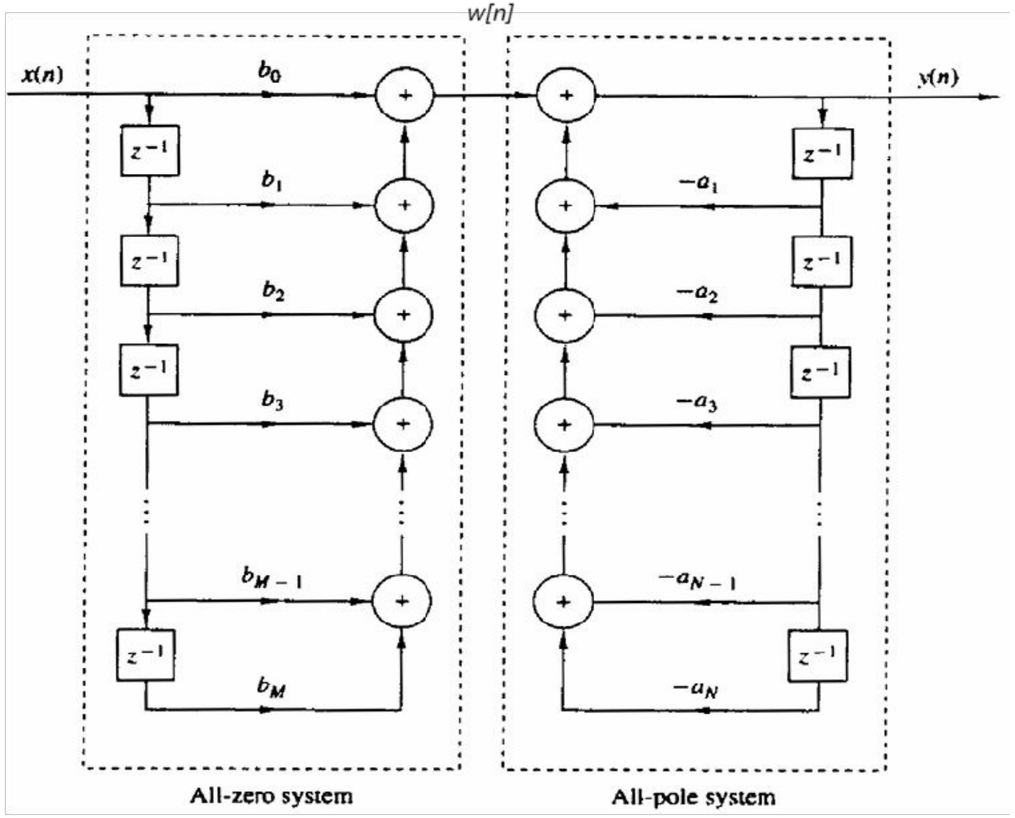
$$w[n] = b_0\, x[n] + b_1\, x[n-1] + b_2\, x[n-2] + \cdots + b_M x[n-M]$$
$$\therefore \quad w[n] = \sum_{k=0}^{M} b_k x[n-k]$$

.....2

- Similarly,

$$\frac{Y(z)}{W(z)} = H_2(z)$$
$$\frac{Y(z)}{W(z)} = \frac{1}{1+\sum_{k=1}^{N} a_k z^{-k}}$$

- Or, $$Y(z) + \sum_{k=1}^{N} a_k z^{-k} Y(z) = W(z)$$
$$Y(z) = -\sum_{k=1}^{N} a_k z^{-k} Y(z) + W(z) \qquad .....3$$
- Taking inverse z-transform, we get
$$y[n] = -\sum_{k=1}^{N} a_k y[n-k] + w[n] \qquad .....4$$
$$y[n] = -a_1\, y[n-1] - a_2\, y[n-2] - \cdots\ - a_N y[n-N] + w[n]$$
- Therefore, direct form-I realization becomes as shown in fig below:



- **Computational Complexity:**

➢ This realization requires:

i. Number of additions = M+N
ii. Number of multiplications = M+N+1
iii. Number of memory locations ( delay elements) = M+N+1

## ii. Direct form II structure:

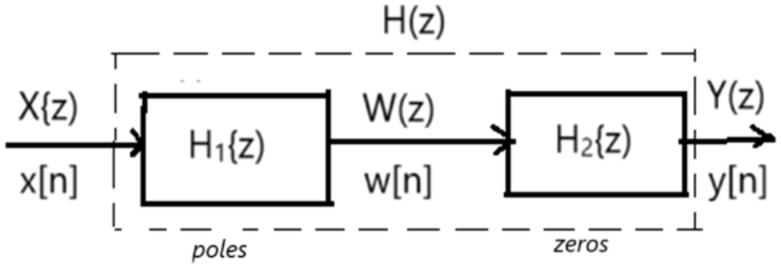
➢ Since, we are dealing with the LTI systems, we can interchange the positions of $H_1(z)$ and $H_2(z)$. This property gives the direct form-II structure.

➢ In direct form-II realization( or structure), poles of $H(z)$ is realized first and then the zeros second.

$$\boldsymbol{H(z) = H_1(z).H_2(z)}$$

where, $\boldsymbol{H_1(z) = \frac{1}{1+\sum_{k=1}^{N} a_k z^{-k}}}$ , all-pole system (recursive)

and $\boldsymbol{H_2(z) = \sum_{k=0}^{M} b_k z^{-k}}$ , all- zero system (non-recursive)




- From figure, we have

$$\frac{W(z)}{X(z)} = H_1(z)$$

$$\frac{W(z)}{X(z)} = \frac{1}{1+\sum_{k=1}^{N} a_k z^{-k}}$$

$$W(z) = X(z)\frac{1}{1+\sum_{k=1}^{N} a_k z^{-k}}$$

- Or,

$$W(z)\{1+\sum_{k=1}^{N} a_k z^{-k}\} = X(z)$$

$$\boldsymbol{W(z) = -\sum_{k=1}^{N} a_k z^{-k} W(z) + X(z)} \qquad \textbf{.....1}$$

- Taking inverse z-transform, we get

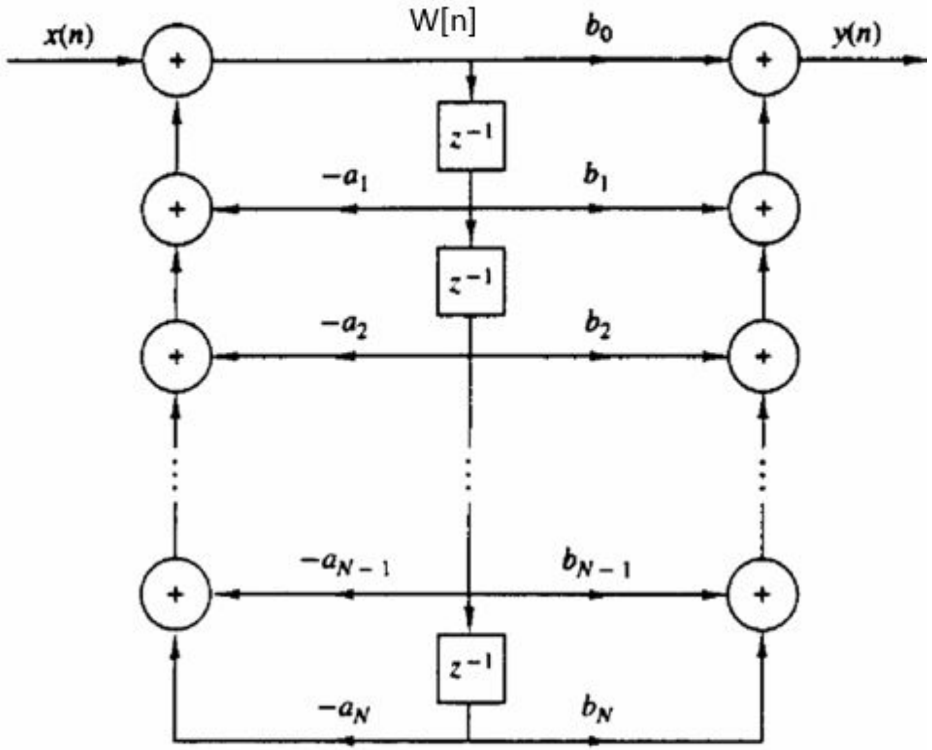
$$\boldsymbol{w[n] = -\sum_{k=1}^{N} a_k w[n-k] + x[n]} \qquad .....2$$

- Similarly,

$$\frac{Y(z)}{W(z)} = H_2(z)$$

$$\frac{Y(z)}{W(z)} = \sum_{k=0}^{M} b_k z^{-k}$$

- Or, $$\frac{Y(z)}{W(z)} = \sum_{k=0}^{M} b_k z^{-k}$$
$$Y(z) = \sum_{k=0}^{M} b_k z^{-k} W(z)$$
- Taking inverse z-transform, we get
$$\boldsymbol{y[n] = \sum_{k=0}^{M} b_k w[n-k]} \qquad .....3$$
- The direct form-II realization (or structure) is shown in figure below (for **N=M**):




- **Computational Complexity:**
- This realization requires:

i. Number of additions = M+N

ii. Number of multiplications = M+N+1

iii. Number of memory locations ( delay elements) is equal to the order of the filter (or, order of the system function or difference equation), hence canonical structure

### b. Cascade Form Structures:

- The cascade form realization of an IIR system or filter is obtained by decomposing the system function $H(z)$ into a product of simpler transfer functions as:

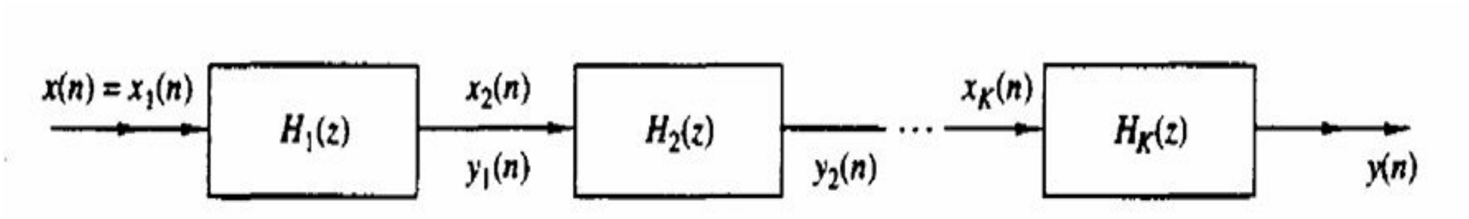
$$H(z) = AH_1(z)H_2(z) \ldots \ldots H_K(z)$$

$$H(z) = A\prod_{k=1}^{K} H_k(z) \qquad \ldots..1$$

where, $A =$ a constant

$k =$ integer part of $\frac{N+1}{2}$

and it is assumed that $M \leq N$



## c. Parallel Form Structures:

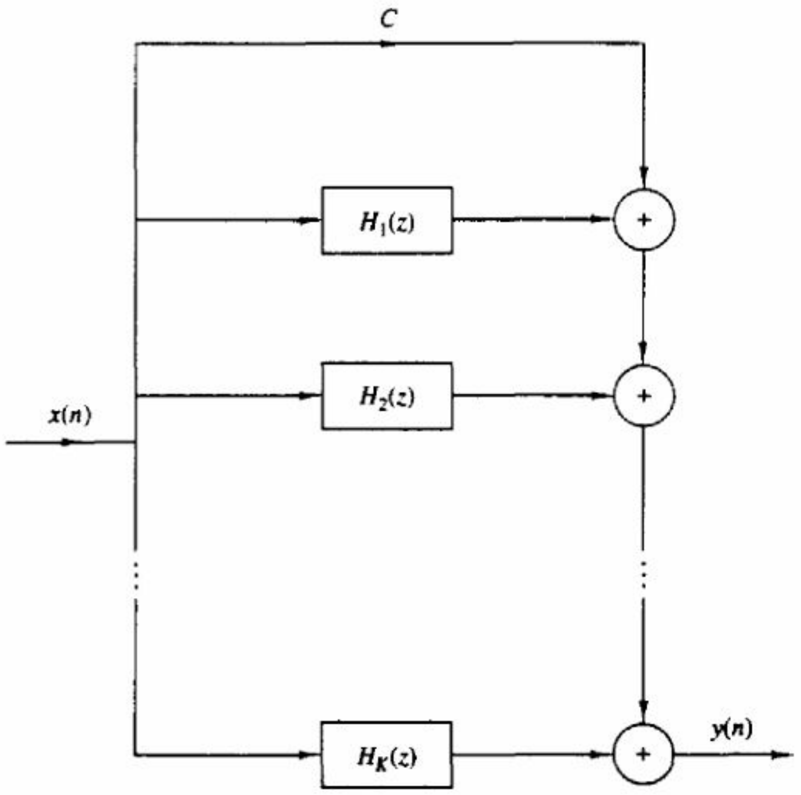
➢ By using partial fractionn epansion, the overall system function $H(z)$ can be expressed as:

$$\boldsymbol{H(z) = C + H_1(z) + H_2(z) + \cdots \ldots + H_K\,(z)} \quad \ldots..1$$

where, $C$ = a constant

$H_1(z), H_2(z), \ldots \ldots, H_K\,(z)$ = second order sub-systems

and, $H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1+\sum_{k=1}^{N} a_k z^{-k}}, a_0 = 0$




- Application:

➢ For ***high-speed filtering***. Because the processing of filtering operation is performed parallelly.

## d. Lattice and Lattice-ladder Structure:

➢ Lattice filters are used in ***digital speech processing*** and the implementing of ***adaptive filters***.

## a. Lattice Structure:

➢ An all-pole system with system function

$$H(z) = \frac{1}{1 + \sum_{k=1}^{N} a_N(k) z^{-k}} = \frac{1}{A_N(z)} \qquad .....1$$

and the difference equation is

$$y[n] = -\sum_{k=1}^{N} a_N(k)\, y[n-k] + x[n] \qquad .....2$$

Or, $$x[n] = y[n] + \sum_{k=1}^{N} a_N(k)\, y[n-k] \qquad .....3$$

- **For $N = 1$:**

➢ We have, $$x[n] = y[n] + a_1(1)\, y[n-1] \qquad \because N = k = 1$$

$$y[n] = x[n] - a_1(1)\, y[n-1] \qquad .....4$$

- We have, $x[n] = y[n] + a_1(1)\, y[n-1] \quad \because N = k = 1$

$$\boldsymbol{y[n] = x[n] - a_1(1)\, y[n-1]} \quad \textbf{.....4}$$

- From figure, we have

$$x[n] = f_1[n]$$

$$y[n] = f_0[n] = f_1[n] - k_1 g_0[n-1]$$

Or, $\boldsymbol{y[n] = x[n] - k_1 y[n-1]}$ **.....5**

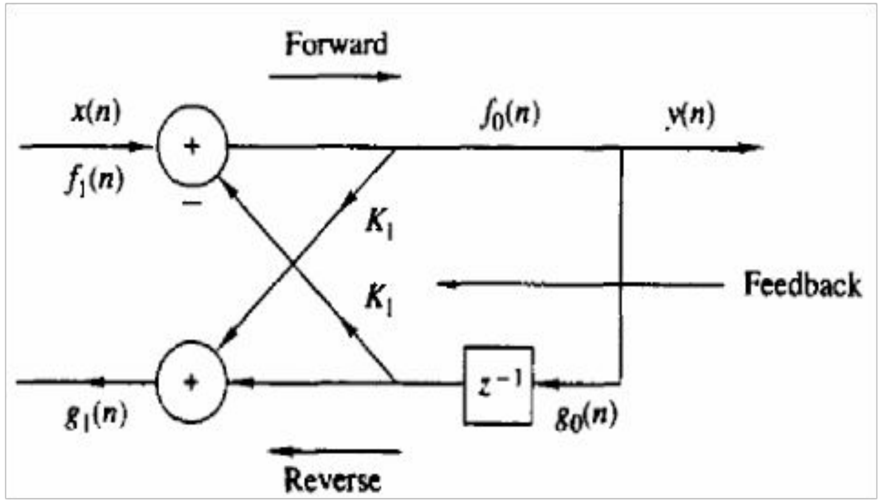



Fig: Single-stage all-pole lattice filter

- Also, $\boldsymbol{g_1[n] = k_1 y[n] + y[n-1]}$
- From equations (4) and (5), we know

$$\boldsymbol{k_1 = a_1(1)} \quad \textbf{.....6}$$

- **For $N = 2$:**

- We have, $x[n] = y[n] + a_2(1)\, y[n-1] + a_2(2)\, y[n-2] \quad \because N = k = 1$

$$\boldsymbol{y[n] = x[n] - a_2(1)\, y[n-1] - a_2(2)\, y[n-2]} \quad \textbf{.....7}$$

- This output can be achieved from two-stage lattice structure as shown in figure below:

- From figure, we have

$$x[n] = f_2[n]$$

$$f_1[n] = f_2[n] - k_2 g_1[n-1]$$
$$g_2[n] = k_2 f_1[n] + k_2 g_1[n-1]$$
$$f_0[n] = f_1[n] - k_1 g_0[n-1]$$
$$g_1[n] = k_1 f_0[n] + g_0[n-1]$$

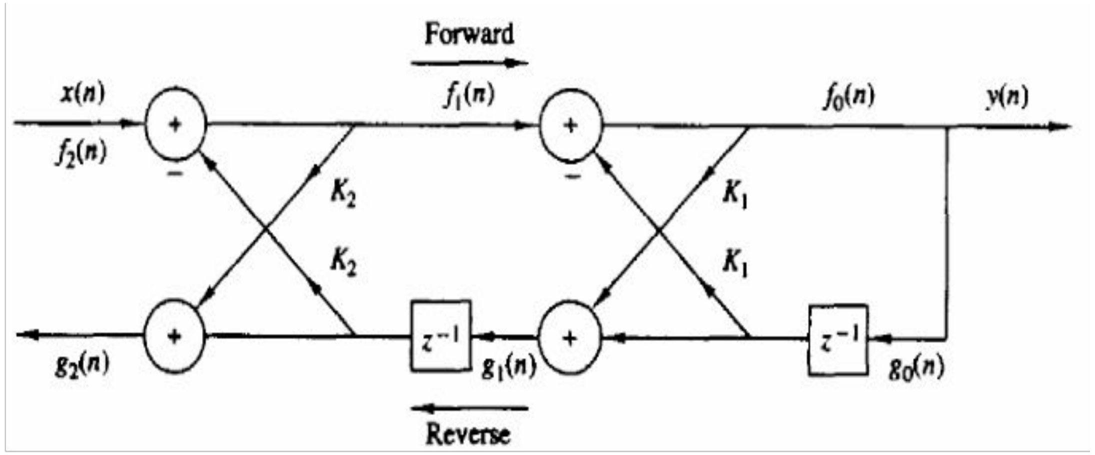



Fig: Two-stage all-pole lattice filter

- Therefore,

$$y[n] = f_0[n] = g_0[n]$$
$$y[n] = f_1[n] - k_1 g_0[n-1]$$
$$y[n] = \{f_2[n] - \ k_2 g_1[n-1]\} - k_1 g_0[n-1]$$
$$y[n] = f_2[n] - k_2\{k_1 f_0[n-1] + g_0[n-2]\} - k_1 g_0[n-1]$$
$$y[n] = f_2[n] - k_2 k_1 f_0[n-1] - k_2 g_0[n-2] - k_1 g_0[n-1]$$
$$y[n] = x[n] - k_2 k_1 y[n-1] - k_2 y[n-2] - k_1 y[n-1]$$

- Therefore, $\boldsymbol{y[n] = x[n] - k_1(1 + k_2)y[n-1] - k_2 y[n-2]}$ …..8

- In similar manner, we obtain

$$g_2[n] = k_2 y[n] + k_1(1 + k_2)y[n-1] + y[n-1] \quad \text{.....9}$$

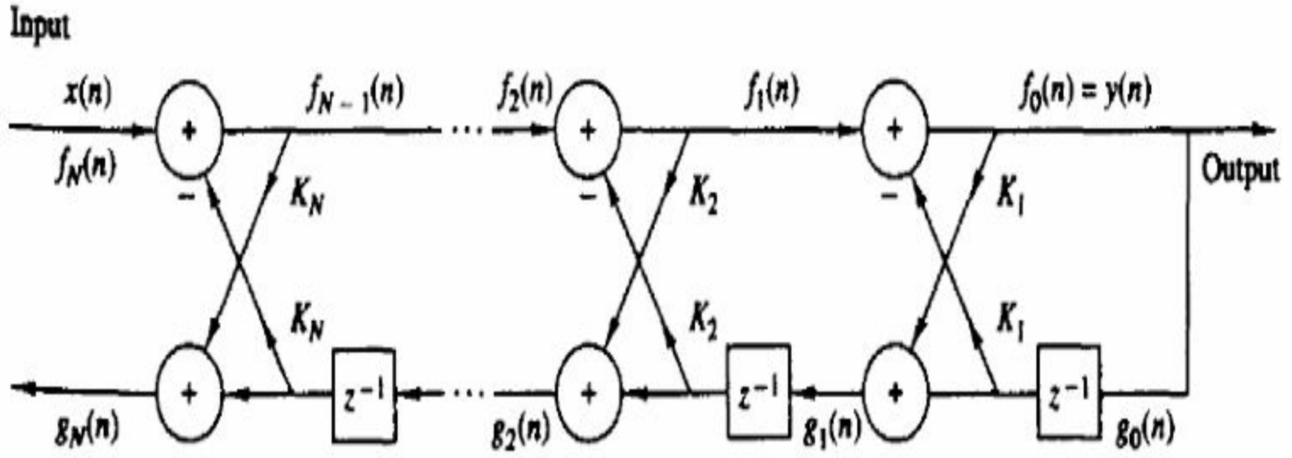
- From equations (7) and (8), we have

$$a_2(0) = 1$$
$$a_2(1) = k_1(1 + k_2)$$

and, $k_1(1 + k_2)$

$$a_2(2) = k_2$$

- Therefore, N-stage lattice structure of IIR filter is obtained as shown in figure below:



- From figure, we have

$$
\begin{aligned}
& f_N[n] = x[n] \\
& f_{m-1}[n] = f_m[n] - k_m g_{m-1}[n-1], \qquad m = N, N-1, \ldots, 1 \\
& g_m[n] = k_m f_{m-1}[n] + g_{m-1}[n-1], \qquad m = N, N-1, \ldots, 1 \\
& y[n] = f_0[n] = g_0[n]
\end{aligned}
$$

## b. Lattice-ladder Structure:

- A general IIR filter connecting both poles and zeros can be realized ( or implemented) using all-pole lattice as building block. To develop appropriate structure, let us consider an IIR system with system function:

$$H(z) = \frac{\sum_{k=0}^{M} c_M(k) z^{-k}}{1 + \sum_{k=1}^{N} a_N(k) z^{-k}} = \frac{C_M(z)}{A_N(z)} \qquad \ldots..1$$

where, $N \geq M$

- The lattice structure for equation (1) is constructed first by realizing all-pole lattice coefficients $\boldsymbol{k_m}$ , where, $\mathbf{1} \leq \boldsymbol{m} \leq \boldsymbol{N}$ for the denominator $\boldsymbol{A_N(z)}$, and then adding the ladder part by taking the output as a weighted linear combination of $\boldsymbol{g_m[m]}$.

- The result is the pole-zero IIR (lattice-ladder) structure.
- The lattice-ladder structure for $\mathrm{M} = \boldsymbol{N}$ is shown in figure below:

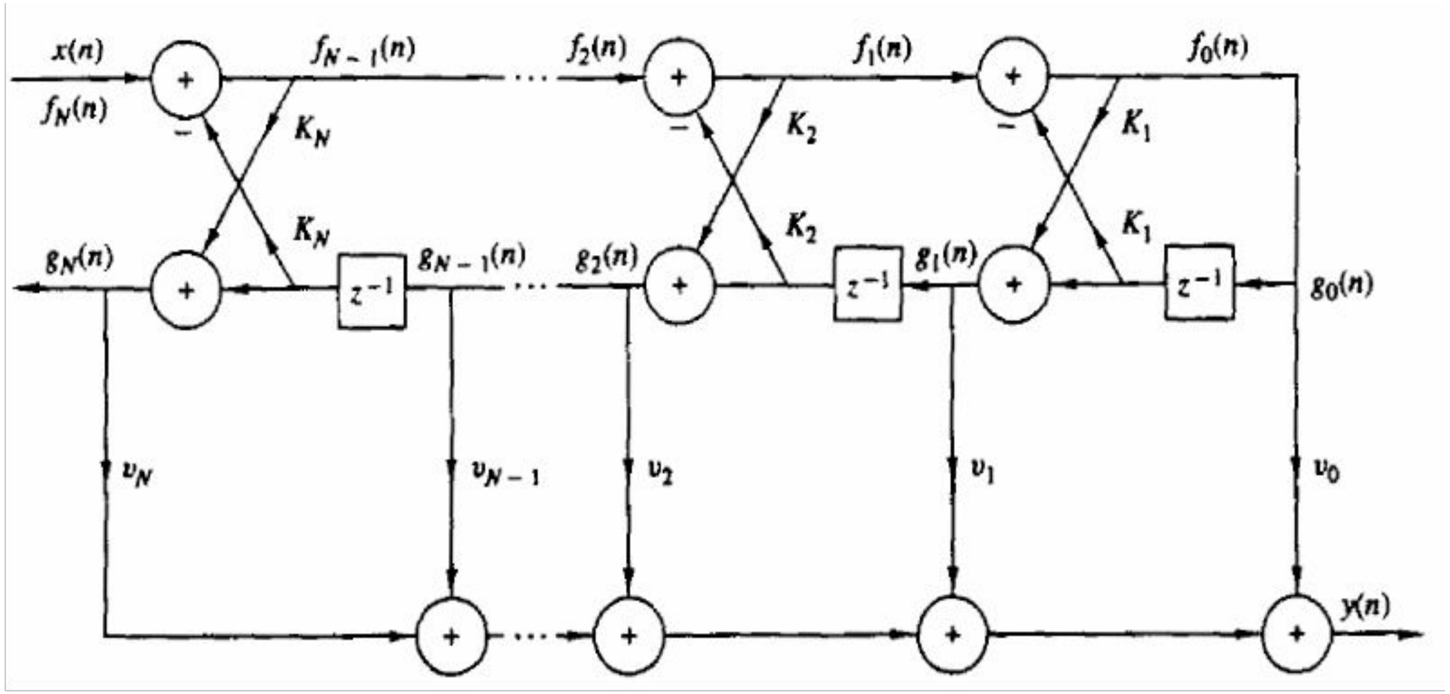



Fig: Lattice-ladder structure for the pole-zero system

- The output is given by

$$y(n) = \sum_{m=0}^{M} v_m g_m(n) \qquad .....2$$

where, $v_m$ = ladder coefficients and obtained by the equation

$$\boldsymbol{v_m = c_m - \sum_{i=1+m}^{M} v_i a_i(i-m), \quad m = M, M-1, \dots.0} \qquad \boldsymbol{.....3}$$

- **Conversion from Lattice Structure to Direct-form Structure:**
- In general,

$$\boldsymbol{a_m(0) = 1}$$
$$\boldsymbol{a_m(m) = k_m}$$
$$\boldsymbol{a_m(k) = a_{m-1}(k) + a_m(m)\, a_{m-1}(m-k)}$$

- **Conversion of Direct Form Coefficients to Lattice Coefficients :**

➢ In general,

$$a_m(0) = 1$$

$$k_m = a_m(m)$$

$$a_{m-1}(k) = \frac{a_m(k) - a_m(m)a_m(m-k)}{1 - \{a_m(m)\}^2}$$

## ❑ Quantization Effect (Truncation and Rounding):