بسم الله الرحمن الرحیم

دانشگاه تهران

پردازش سیگنال دیجیتال

درس ۱۸

ساختارهایی برای سیستم‌های گسسته–زمان

Structures for Discrete-Time Systems

کاظم فولادی قلعه
دانشکده مهندسی، پردیس فارابی
دانشگاه تهران

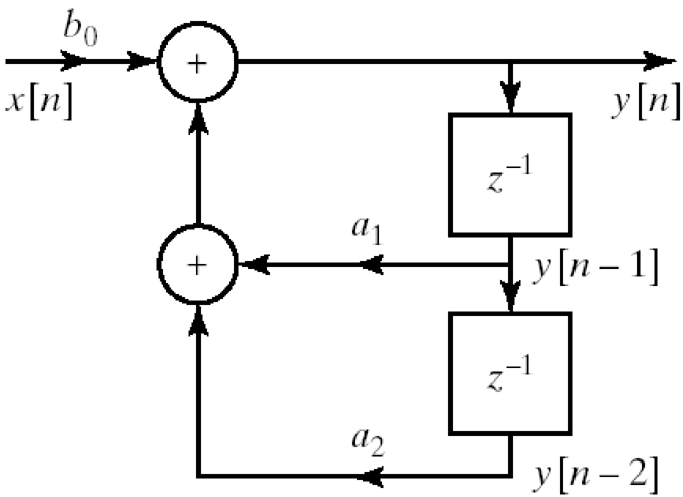
http://courses.fouladi.ir/dsp

Structures for Discrete-Time Systems

Example

- Block diagram representation of

$$y[n] = a_1 y[n-1] + a_2 y[n-2] + b_0 x[n]$$



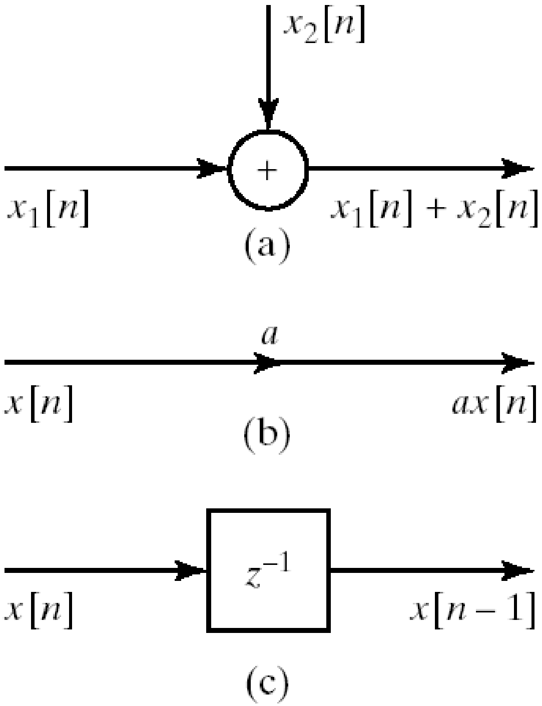
Block Diagram Representation

- LTI systems with rational system function can be represented as constant-coefficient difference equation

- The implementation of difference equations requires **delayed values** of the
  - input
  - output
  - intermediate results

- The requirement of delayed elements implies need for **storage**

- We also need means of
  - addition
  - multiplication



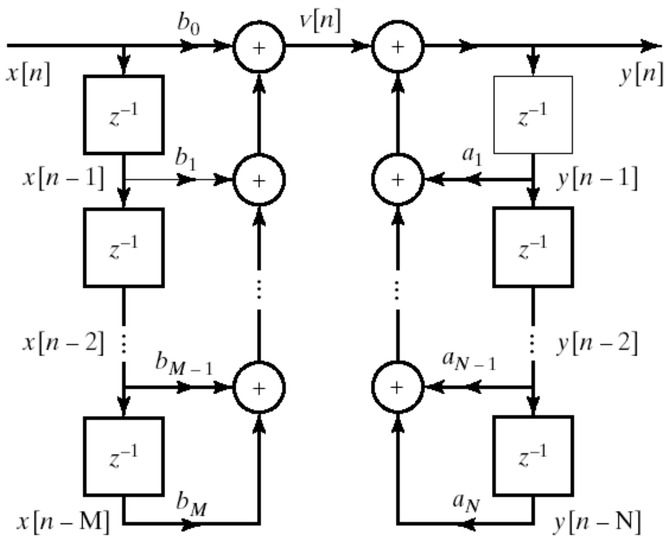
Direct Form I

- General form of difference equation

$$\sum_{k=0}^{N} \hat{a}_k y[n-k] = \sum_{k=0}^{M} \hat{b}_k x[n-k]$$

- Alternative equivalent form

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



$$v[n] = \sum_{k=0}^{M} b_k x[n-k]$$

$$y[n] = \sum_{k=1}^{N} a_k y[n-k] + v[n]$$

Direct Form I

- Transfer function can be written as

$$H(z)=\frac{\sum_{k=0}^{M} b_k z^{-k}}{1-\sum_{k=1}^{N} a_k z^{-k}}$$

- Direct Form I Represents

$$H(z)=H_2(z)H_1(z)=\left(\frac{1}{1-\sum_{k=1}^{N} a_k z^{-k}}\right)\left(\sum_{k=0}^{M} b_k z^{-k}\right)$$

$$V(z)=H_1(z)X(z)=\left(\sum_{k=0}^{M} b_k z^{-k}\right)X(z)$$

$$Y(z)=H_2(z)V(z)=\left(\frac{1}{1-\sum_{k=1}^{N} a_k z^{-k}}\right)V(z)$$

$$v[n]=\sum_{k=0}^{M} b_k x[n-k]$$

$$y[n]=\sum_{k=1}^{N} a_k y[n-k]+v[n]$$

Alternative Representation

- Replace order of cascade LTI systems

$$H(z) = H_1(z)H_2(z) = \left(\sum_{k=0}^{M} b_k z^{-k}\right)\left(\frac{1}{1-\sum_{k=1}^{N} a_k z^{-k}}\right)$$

$$W(z) = H_2(z)X(z) = \left(\frac{1}{1-\sum_{k=1}^{N} a_k z^{-k}}\right)X(z)$$

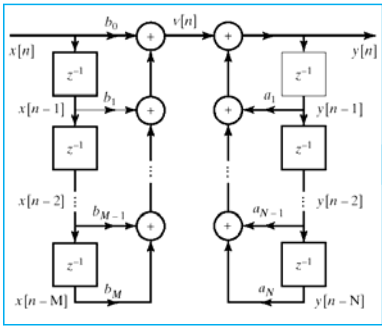
$$Y(z) = H_1(z)W(z) = \left(\sum_{k=0}^{M} b_k z^{-k}\right)W(z)$$

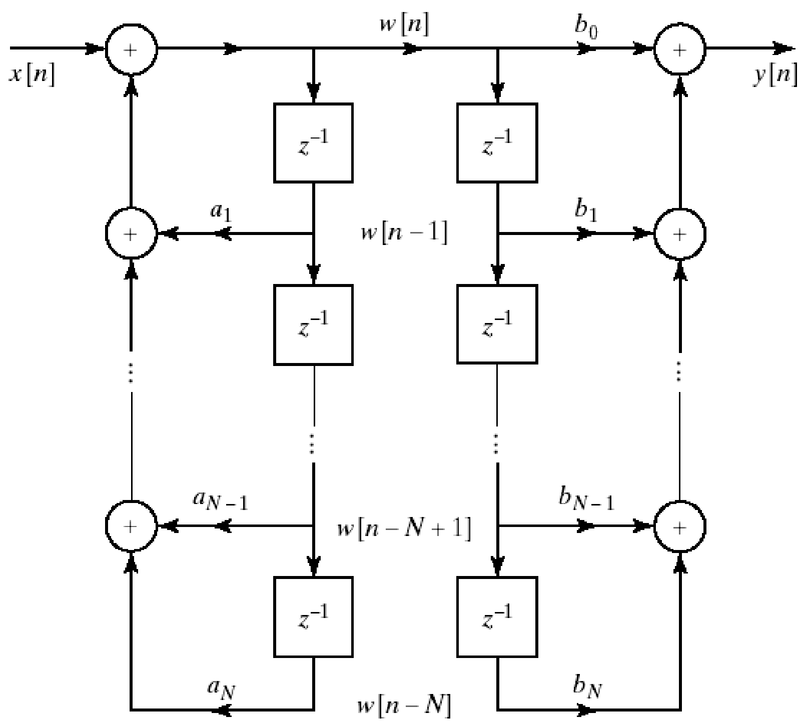
$$w[n] = \sum_{k=1}^{N} a_k w[n-k] + x[n]$$

$$y[n] = \sum_{k=0}^{M} b_k w[n-k]$$

Alternative Block Diagram

- We can change the order of the cascade systems



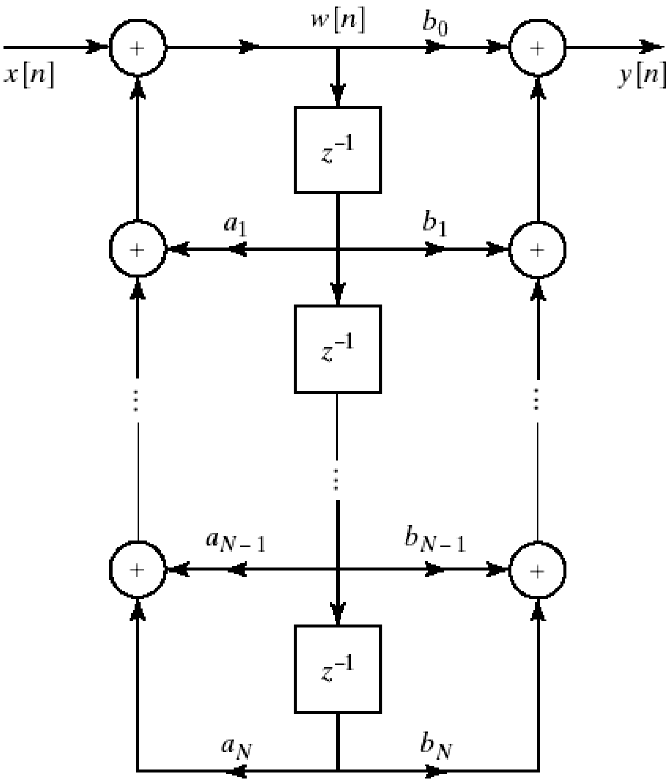


$$w[n] = \sum_{k=1}^{N} a_k w[n-k] + x[n]$$

$$y[n] = \sum_{k=0}^{M} b_k w[n-k]$$

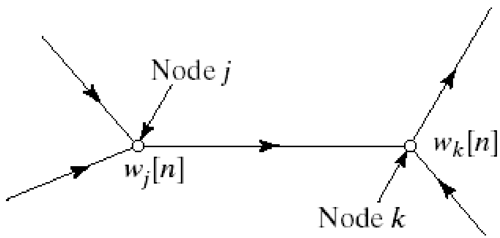
Direct Form II

- No need to store the same data twice in previous system
- So we can collapse the delay elements into one chain
- This is called Direct Form II or the **Canonical Form**
- Theoretically no difference between Direct Form I and II
- Implementation wise
  - Less memory in Direct II
  - Difference when using finite-precision arithmetic

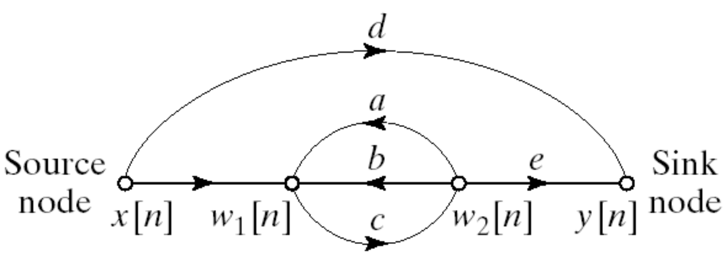


Signal Flow Graph Representation

- Similar to block diagram representation
  - Notational differences
- A network of directed branches connected at nodes



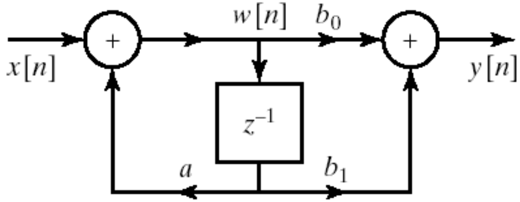
- Example representation of a difference equation



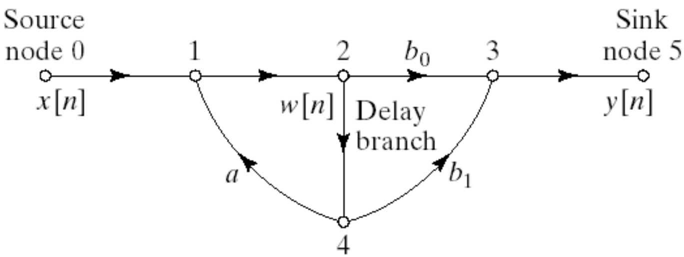
$$w_1[n] = x[n] + aw_2[n] + bw_2[n]$$
$$w_2[n] = cw_1[n]$$
$$y[n] = dx[n] + ew_2[n]$$

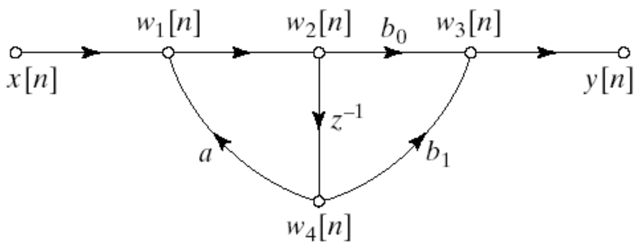
Example

- Representation of Direct Form II with signal flow graphs



(a)

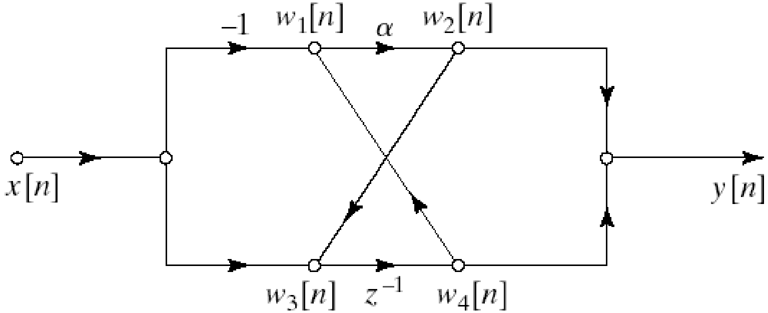




$$w_1[n] = aw_4[n] + x[n]$$
$$w_2[n] = w_1[n]$$
$$w_3[n] = b_0 w_2[n] + b_1 w_4[n]$$
$$w_4[n] = w_2[n-1]$$
$$y[n] = w_3[n]$$

$$w_1[n] = aw_1[n-1] + x[n]$$
$$y[n] = b_0 w_1[n] + b_1 w_1[n-1]$$

Determination of System Function from Flow Graph



$$w_1[n] = w_4[n] - x[n]$$
$$w_2[n] = \alpha w_1[n]$$
$$w_3[n] = w_2[n] + x[n]$$
$$w_4[n] = w_3[n-1]$$
$$y[n] = w_2[n] + w_4[n]$$

$$W_1(z) = W_4(z) - X(z)$$
$$W_2(z) = \alpha W_1(z)$$
$$W_3(z) = W_2(z) + X(z)$$
$$W_4(z) = W_3(z) z^{-1}$$
$$Y(z) = W_2(z) + W_4(z)$$

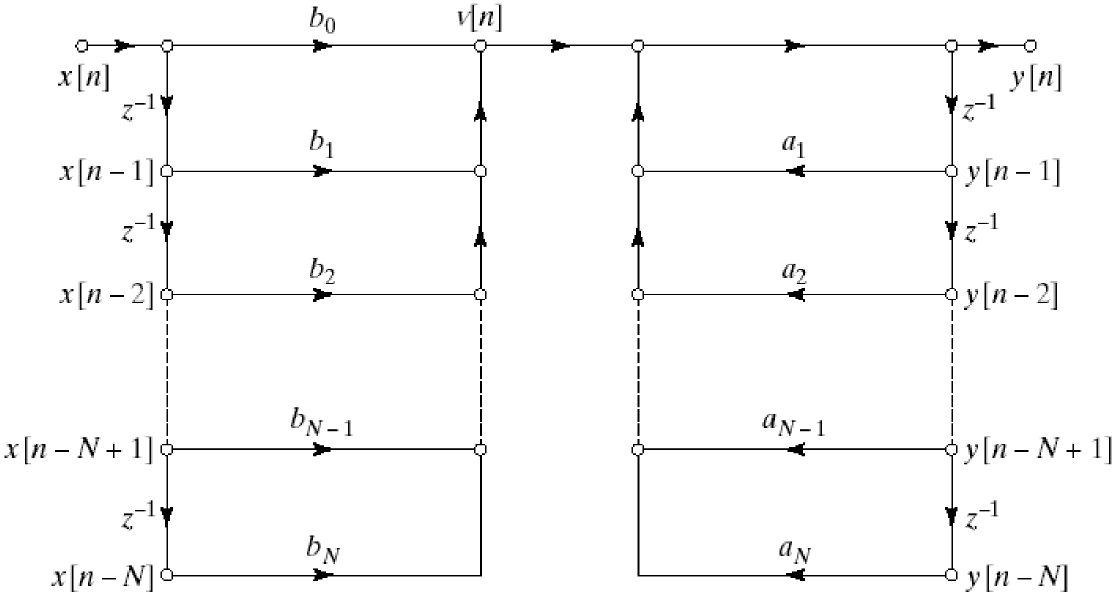
$$\longrightarrow$$

$$W_2(z) = \frac{\alpha X(z)\left(z^{-1} - 1\right)}{1 - \alpha z^{-1}}$$
$$W_4(z) = \frac{X(z) z^{-1}(1 - \alpha)}{1 - \alpha z^{-1}}$$
$$Y(z) = W_2(z) + W_4(z)$$

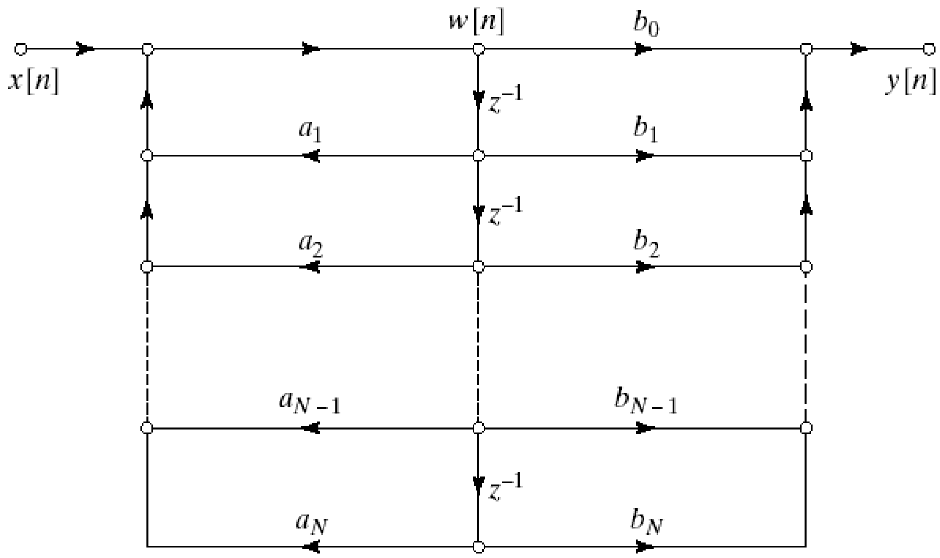
$$\longrightarrow$$

$$H(z) = \frac{Y(z)}{X(z)} = \frac{z^{-1} - \alpha}{1 - \alpha z^{-1}}$$
$$h[n] = \alpha^{n-1} u[n-1] - \alpha^{n+1} u[n]$$

Basic Structures for IIR Systems: Direct Form I



Basic Structures for IIR Systems: Direct Form II



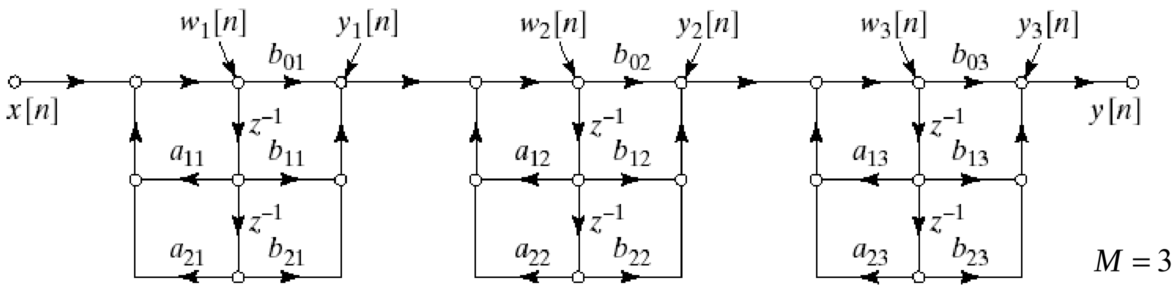
Basic Structures for IIR Systems: Cascade Form

- General form for cascade implementation

$$H(z) = A\frac{\prod_{k=1}^{M_1}\left(1-f_k z^{-1}\right)\prod_{k=1}^{M_2}\left(1-g_k z^{-1}\right)\left(1-g_k^* z^{-1}\right)}{\prod_{k=1}^{N_1}\left(1-c_k z^{-1}\right)\prod_{k=1}^{N_2}\left(1-d_k z^{-1}\right)\left(1-d_k^* z^{-1}\right)}$$

- More practical form in 2nd order systems

$$H(z) = \prod_{k=1}^{M_1}\frac{b_{0k}+b_{1k}z^{-1}+b_{2k}z^{-2}}{1-a_{1k}z^{-1}-a_{2k}z^{-2}}$$

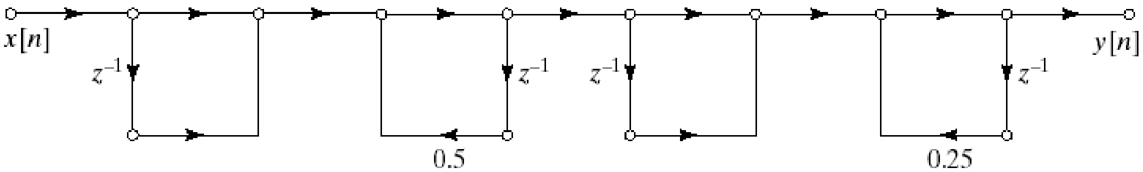


Example

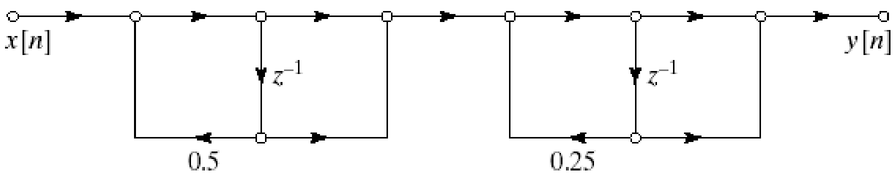
$$H(z)=\frac{1+2z^{-1}+z^{-2}}{1-0.75z^{-1}+0.125z^{-2}}=\frac{(1+z^{-1})(1+z^{-1})}{(1-0.5z^{-1})(1-0.25z^{-1})}$$

$$=\frac{(1+z^{-1})}{(1-0.5z^{-1})}\frac{(1+z^{-1})}{(1-0.25z^{-1})}$$

- Cascade of Direct Form I subsections



- Cascade of Direct Form II subsections



Basic Structures for IIR Systems: Parallel Form

- Represent system function using partial fraction expansion

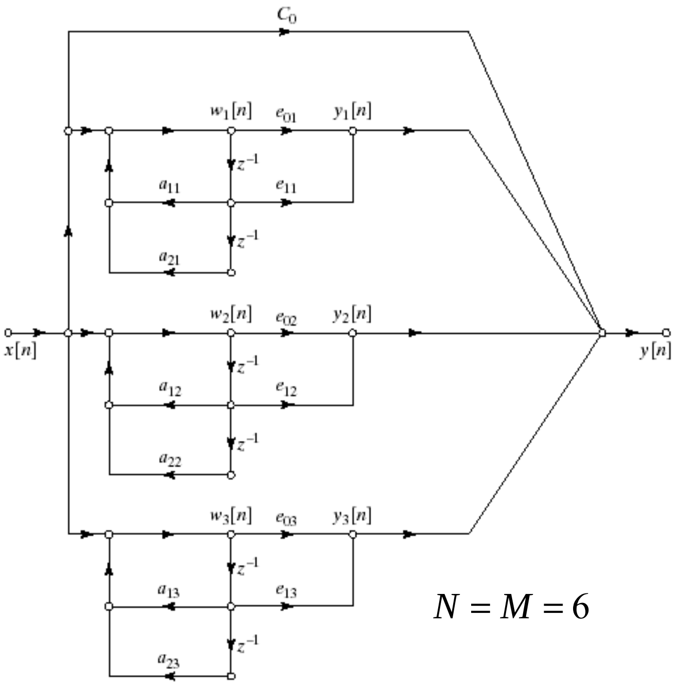
$$H(z) = \sum_{k=0}^{N_P} C_k z^{-k} + \sum_{k=1}^{N_1} \frac{A_k}{1 - c_k z^{-1}} + \sum_{k=1}^{N_2} \frac{B_k\left(1 - e_k z^{-1}\right)}{\left(1 - d_k z^{-1}\right)\left(1 - d_k^* z^{-1}\right)}$$

$$N = N_1 + 2N_2$$
$$N_P = M - N \quad (M \ge N)$$

- Or by pairing the real poles

$$H(z) = \sum_{k=0}^{N_P} C_k z^{-k} + \sum_{k=1}^{N_S} \frac{e_{0k} + e_{1k} z^{-1}}{1 - a_{1k} z^{-1} - a_{2k} z^{-2}}$$

$$N_S = \left\lfloor \frac{N+1}{2} \right\rfloor$$



$N = M = 6$

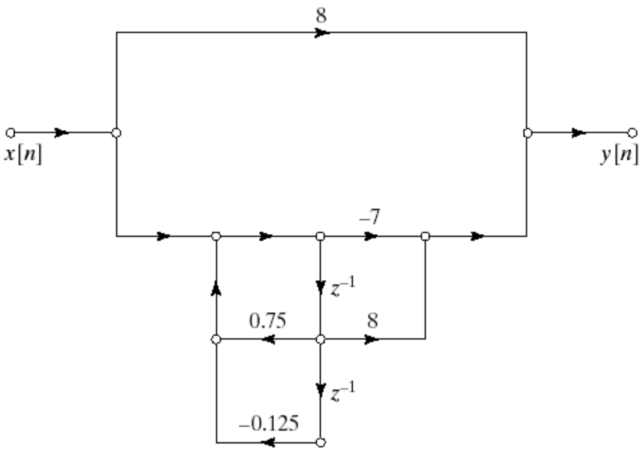
Example

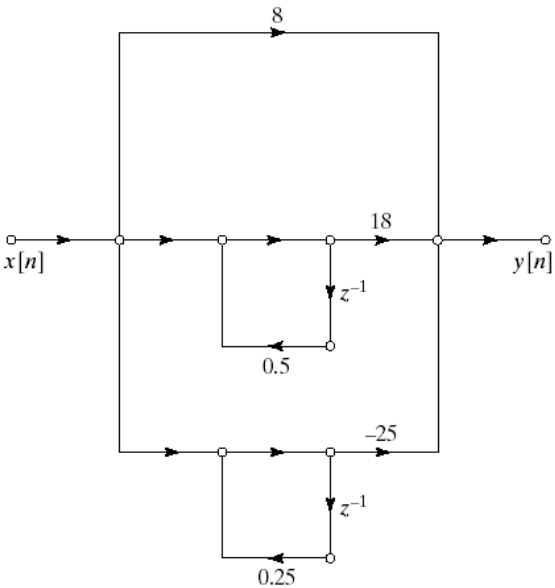
- Partial Fraction Expansion

$$H(z)=\frac{1+2z^{-1}+z^{-2}}{1-0.75z^{-1}+0.125z^{-2}}=8+\frac{18}{\left(1-0.5z^{-1}\right)}-\frac{25}{\left(1-0.25z^{-1}\right)}$$

- Combine poles to get

$$H(z)=8+\frac{-7+8z^{-1}}{1-0.75z^{-1}+0.125z^{-2}}$$

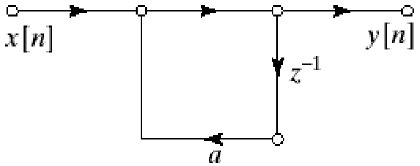




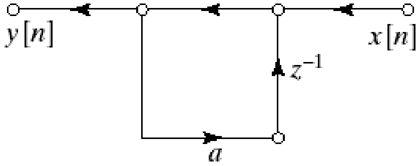
Transposed Forms

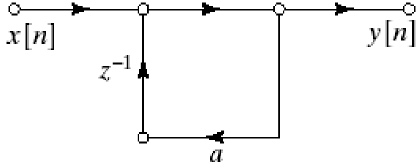
- Linear signal flow graph property:
  - **Transposing doesn't change the input-output relation**
- Transposing:
  - **Reverse** directions of all branches
  - **Interchange** input and output nodes
- Example:

$$H(z) = \frac{1}{1 - az^{-1}}$$

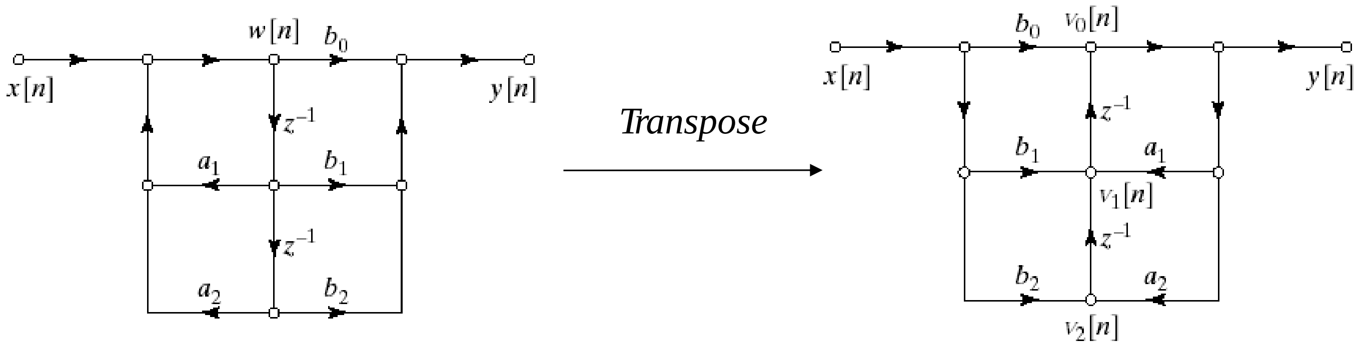


  - Reverse directions of branches and interchange input and output





Example

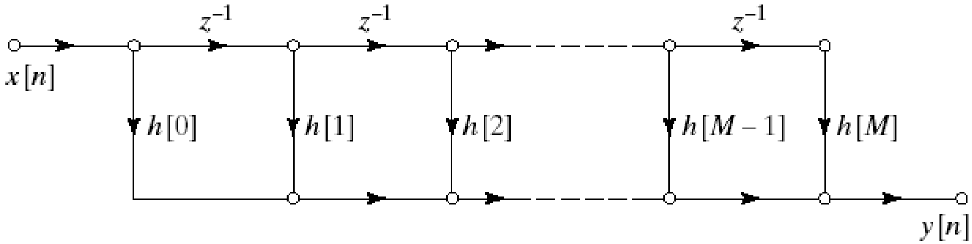


- Both have the same system function or difference equation

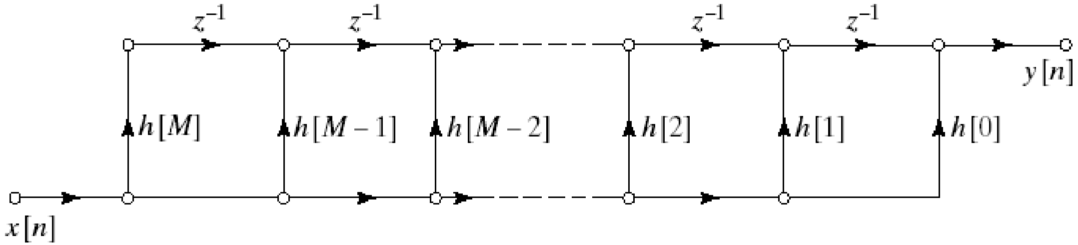
$$y[n] = a_1 y[n-1] + a_2 y[n-2] + b_0 x[n] + b_1 x[n-1] + b_2 x[n-2]$$

Basic Structures for FIR Systems: Direct Form

- Special cases of IIR direct form structures



- **Transpose** of direct form **I** gives direct form **II**
- Both forms are equal for FIR systems

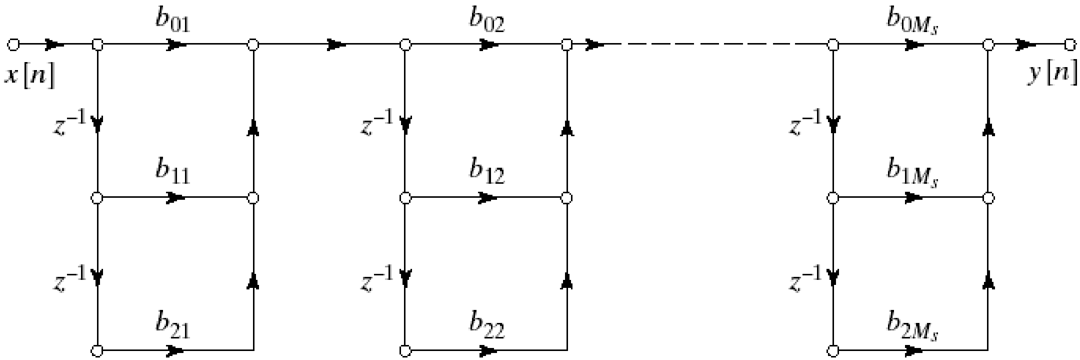


- Tapped delay line

Basic Structures for FIR Systems: Cascade Form

- Obtained by factoring the polynomial system function

$$H(z) = \sum_{n=0}^{M} h[n] z^{-n} = \prod_{k=1}^{M_S} \left(b_{0k} + b_{1k} z^{-1} + b_{2k} z^{-2} \right)$$



$$M_S = \left\lfloor \frac{M+1}{2} \right\rfloor$$

Structures for Linear-Phase FIR Systems

- **Causal FIR system** with generalized linear phase are symmetric:

$$h[M-n] = h[n] \qquad n = 0,1,\ldots,M \quad \text{(type I or III)}$$

$$h[M-n] = -h[n] \qquad n = 0,1,\ldots,M \quad \text{(type II or IV)}$$

- Symmetry means we can half the number of multiplications
- **Example**:
  For even M and type I or type III systems:

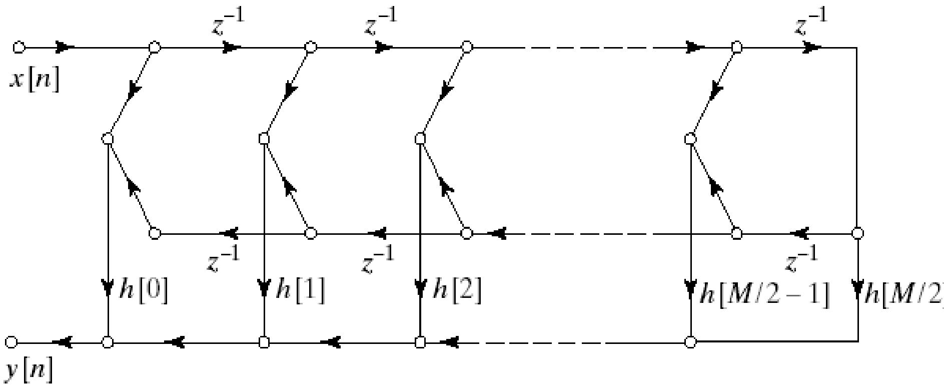
$$y[n] = \sum_{k=0}^{M} h[k]x[n-k] = \sum_{k=0}^{M/2-1} h[k]x[n-k] + h[M/2]x[n-M/2] + \sum_{k=M/2+1}^{M} h[k]x[n-k]$$

$$= \sum_{k=0}^{M/2-1} h[k]x[n-k] + h[M/2]x[n-M/2] + \sum_{k=0}^{M/2-1} h[M-k]x[n-M+k]$$

$$= \sum_{k=0}^{M/2-1} h[k]\left(x[n-k] + x[n-M+k]\right) + h[M/2]x[n-M/2]$$

Structures for Linear-Phase FIR Systems

- Structure for even M



- Structure for odd M