CH.6

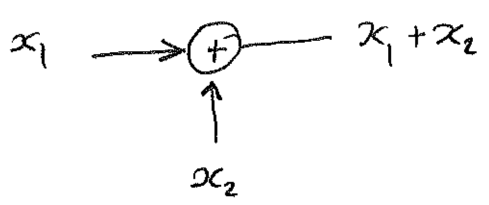
Structures for Discrete-Time Systems

* Note that, the difference equation, the impulse response, and the system function are equivalent characterization of the input-output relation of an LTI discrete-time system.

* For implementation by hardware or software, the difference equation or system function must be converted to an algorithm or structure that can be realized in the desired technology.

* Systems that are described by a LCCDE can be represented by structures consisting of an interconnection of basic operations of

① addition



② Multiplication

③ Delay

Example 6.1: Block diagram representation of a difference equation

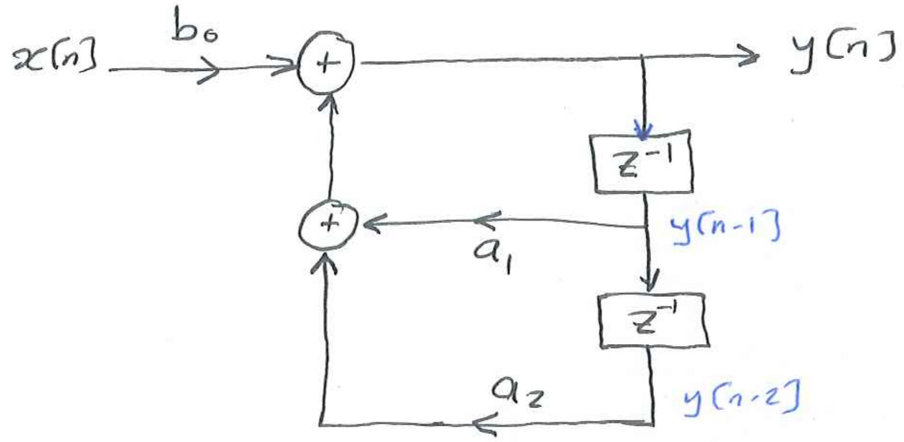
consider the 2nd order DE:

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

the corresponding System function:

$$H(z) = \frac{b_0}{1 - a_1 z^{-1} - a_2 z^{-2}}$$

the block diagram representation of the DE :-



Note: When the system is implemented on either the general purpose computer or digital Signal processing (DSP) chip, network structure (Block diagram) serves as the basis for a program that implements the system.

In the given block diagram, the computational complexity
two adders
three Multipliers
two storage elements.

The previous example can be generalized to higher-order Difference equations of the form

Note, in the previous chapters, we used: $\sum_{k=0}^{N} a_k y[n-k] = \sum_{k=0}^{M} b_k x[n-k]$

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

with the corresponding system function, $H(z) = \dfrac{\sum_{k=0}^{M} b_k z^{-k}}{1 - \sum_{k=1}^{N} a_k z^{-k}}$

Rewriting the DE as a recursive formula for $y[n]$ in terms of a linear combination of past values of the output sequence and current and past values of the input sequence leads to the relation

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

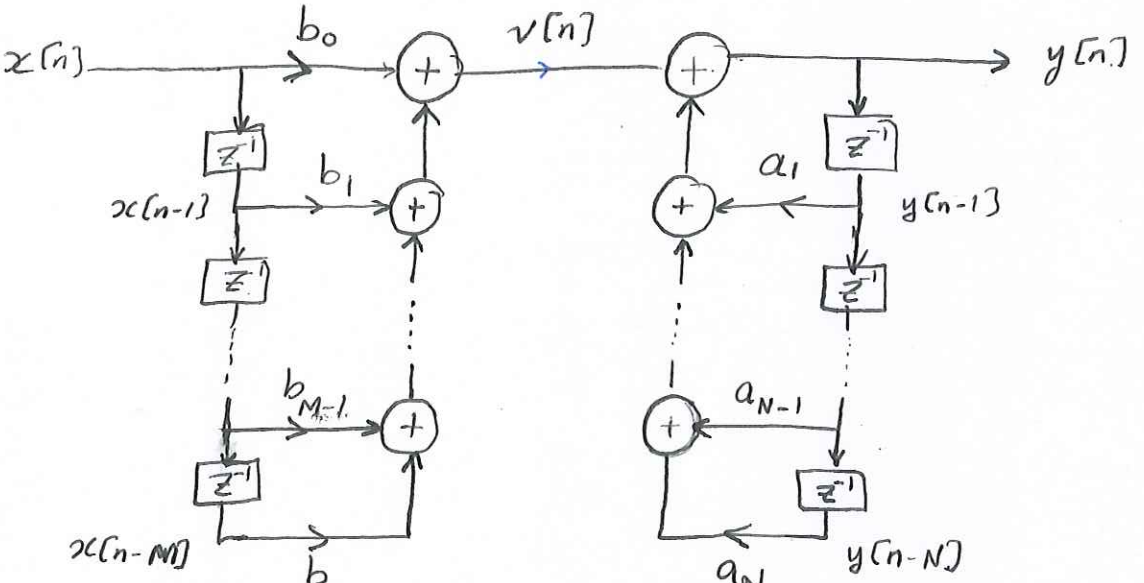
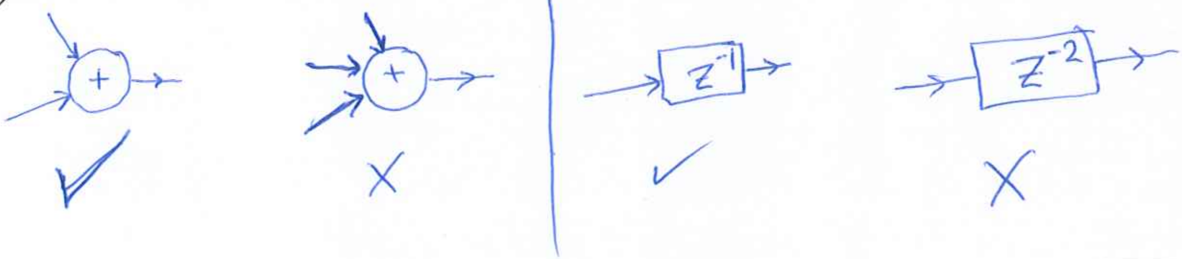


Fig. 1

- * This implementation is called Direct form I
- * we assumed (as a convention) two input adders and one delay element z^{-1}.

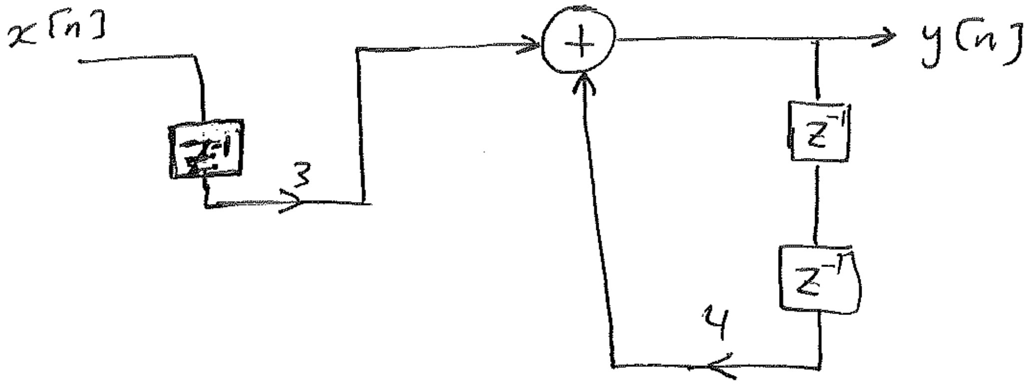


Example: let $H(z) = \dfrac{3z^{-1}}{1-4z^{-2}}$

then, $y[n] - 4y[n-2] = 3x[n-1]$

the recurrsive formula :-

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



In Matlab, the function filter(b, a, X] is used to implement the Direct form I.

X = [. . . .] input vector

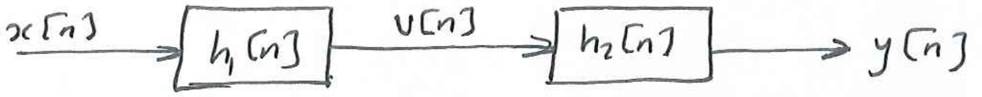
b = [0, 3]

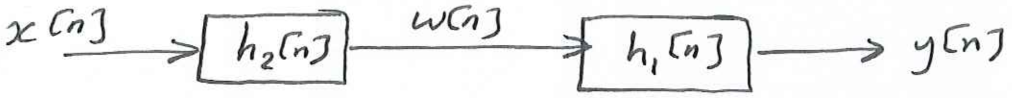
a = [1 , 0, -4]

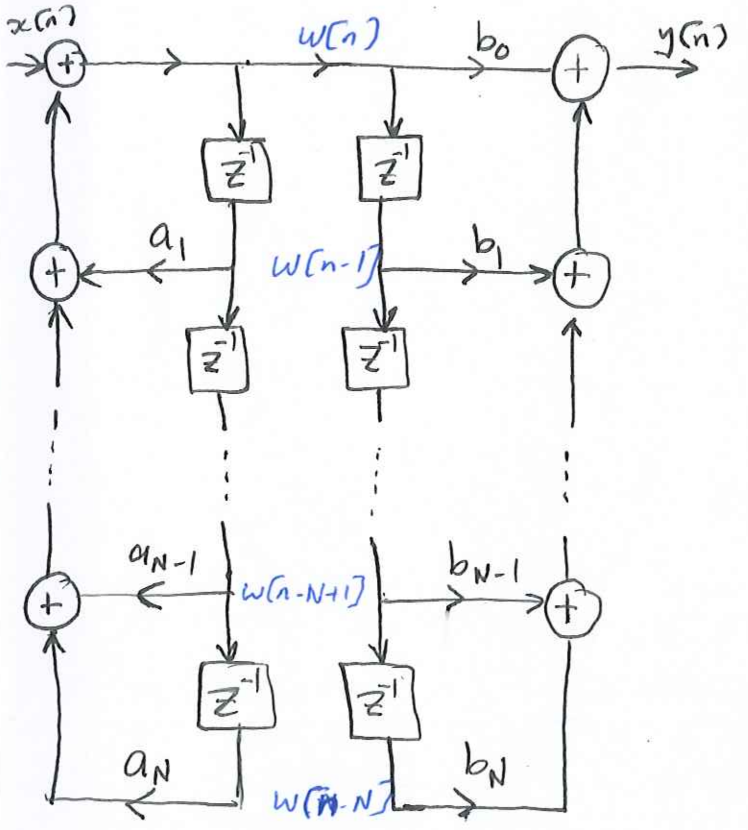
- Block diagram given in fig. 1 (Direct form I) can be rearranged in a variety of ways without changing the overall system function.

- Each rearrangement represents a different computational algorithm for implementing the same system.

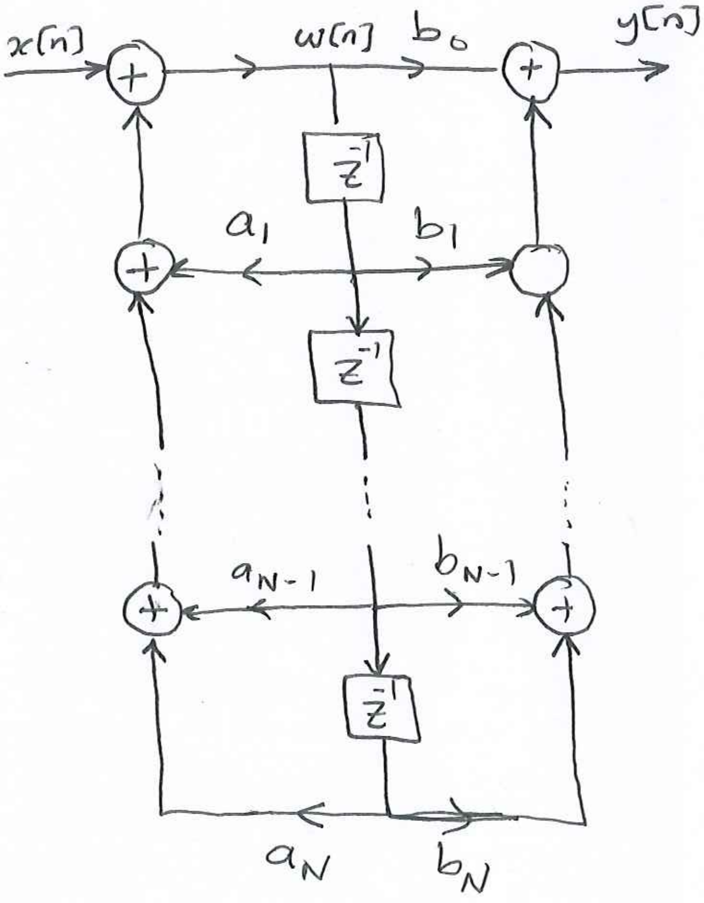
- Fig. 1 (Direct form I) can be viewed as a cascade of two subsystems







we assumed M = N



Direct form II
(canonical form) implementation

	(Adds + Mult.) Computational Complexity	(Memory) Delay elements
Direct form I	same	M + N
Direct form II	same.	max (N, M)

* An implementation with the minimum number of delay elements is commonly referred to as a canonic form implementation (Direct form II).

* From the generalization $H(z) = \dfrac{\sum_{k=0}^{M} b_k z^{-k}}{1 - \sum_{k=1}^{N} a_k z^{-k}}$

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

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

or equivalently, $W(z) = H_2(z)\, X(z)$

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

in t-domain,

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

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

Example 6.2 Direct Form I and Direct Form II implementations of an LTI system

$$H(z) = \frac{1 + 2z^{-1}}{1 - 1.5z^{-1} + 0.9z^{-2}}$$

Solution:

in comparison with the general form $H(z) = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1 - \sum_{k=1}^{N} a_k z^{-k}}$

$\Rightarrow$ $b_0 = 1$

$b_2 = 2$

$a_1 = 1.5$

$a_2 = -0.9$

} always have opposite sign (feedback coeffecients)

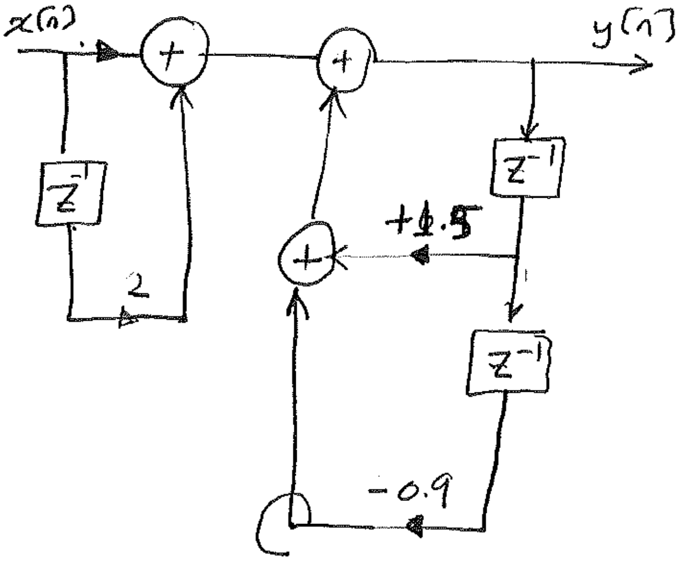
OR $\frac{Y(z)}{X(z)} = \frac{1 + 2z^{-1}}{1 - 1.5z^{-1} + 0.9z^{-2}}$

then, (apply $z^{-1}\{.\}$),

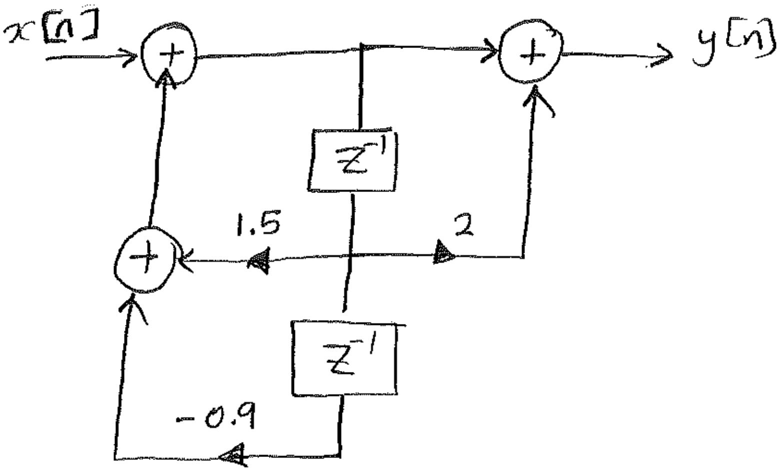
$$y[n] - 1.5y[n-1] + 0.9y[n-2] = x[n] + 2x[n-1]$$

the recursive form,

$$y[n] = 1.5y[n-1] - 0.9y[n-2] + x[n] + 2x[n-1]$$



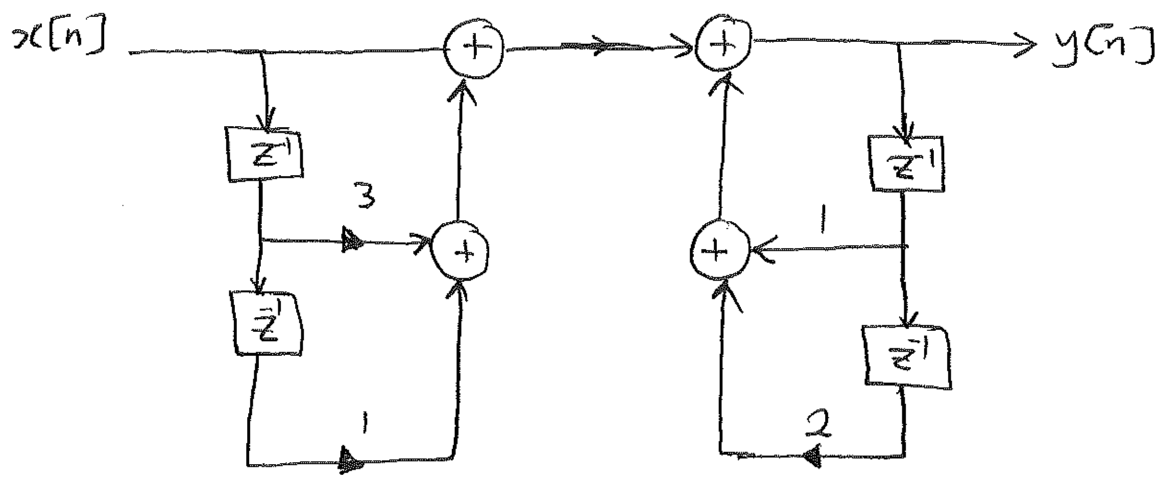
Direct Form I



Direct form II

Problem 6.5 (textbook)

An LTI system is realized by block diagram :-



(1) Write the Difference equation.

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

(2) What is the system function.

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

(3) How many real multiplications and real additions are required to compute each sample of y[n] ?
(Assume x[n] is real, and assume the multiplication by 1 does not count in the total complexity).

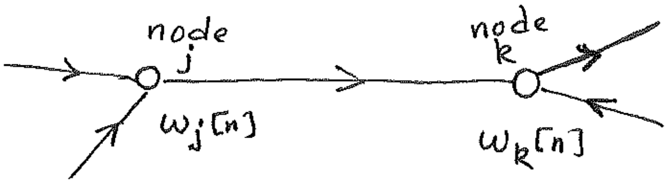
Four addition + 2 multiplication

(4) The given realization requires four storage registers. Is it possible to reduce the number of storage registers by using different structure?
If yes, draw the block diagram.

Solution : Yes, Direct form II.
it needs two registers
2 MwH + 4 adders.

6.2 Signal Flow Graph Representation of LCCDE

- * It is an alternative representation to the block diagram (similar to the block diagram, except for a few notational differences)
- * Signal flow graph is a network of directed branches that connect all nodes.



- * With each node, there is a variable associated with node value.

  (the value associated with node k is denoted as w_k, and since they are sequences, we can write $w_k[n]$).

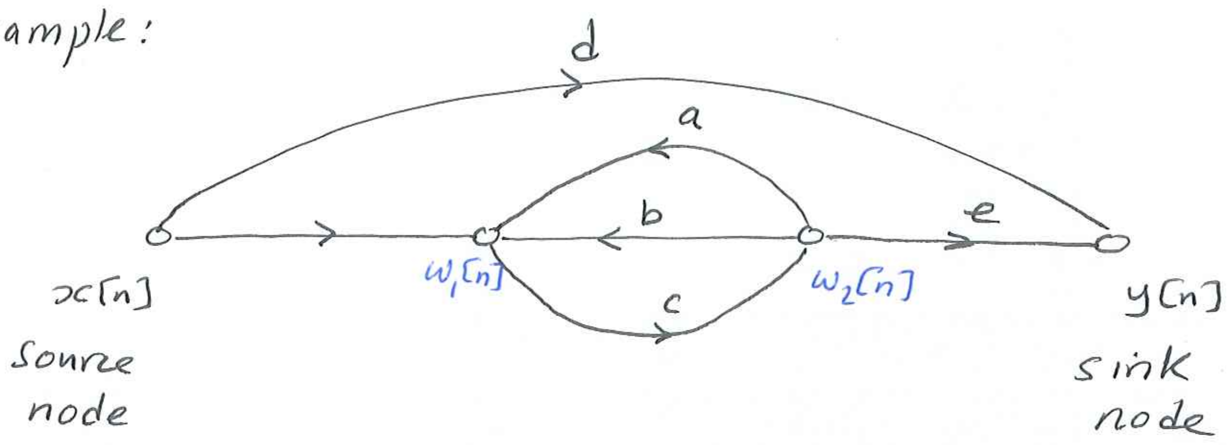
- * Branch (j, k) is a branch orginating at node j and terminating at node k.
- * Each node has an input signal and an output signal
- * The input signal from node j to branch (j, k) is the node value $w_j[n]$.
- * The value at each node in a graph is the sum of the outputs of all branches entering the node.

- * Two special types of nodes :-

  ① source nodes : nodes that have no entering braches

  ② sink nodes : 〃 〃 〃 only 〃 〃 .

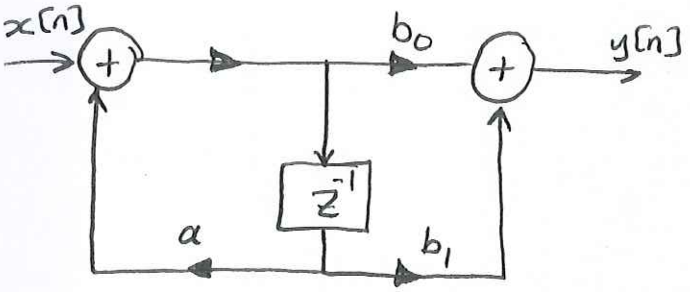
Example:



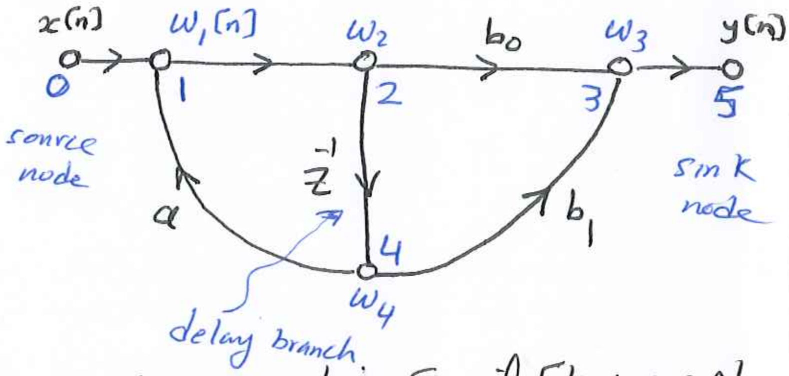
The linear equations that covers the graph:

$$w_1[n] = x[n] + a\,w_2[n] + b\,w_2[n]$$
$$w_2[n] = c\,w_1[n]$$
$$y[n] = d\,x[n] + e\,w_2[n]$$

Example: consider the block diagram representation of a 1^{st} order digital filter.



Direct Form II



Corresponding Signal Flow graph

node 0 is a source node whose value is determined by the input sequence $x[n]$.

node 5 is a sink node whose value is denoted by $y[n]$ (Note that the source and sink nodes are connected to the rest of the graph by unity-gain branches).

The flow graph represents a set of difference equations, with one equation being written at each node.

The reprerenting equations at each node:-

① $w_1[n] = a\,w_4[n] + x[n]$

② $w_2[n] = w_1[n]$

③ $w_3[n] = b_0\,w_2[n] + b_1\,w_4[n]$

④ $w_4[n] = w_2[n-1]$

⑤ $y[n] = w_3[n]$

we can eliminate some of the variables to obtain the pair of equations:-

sub. ④ in ① then the result in ② :

$\Rightarrow \quad w_2[n] = a\,w_2[n-1] + x[n]$ ——— ⊛

sub. ③ in ⑤ :

$\Rightarrow \quad y[n] = b_0\,w_2[n] + b_1\,w_2[n-1]$ ——— (**)

⊛ and (**) are in Direct Form II format.

initial rest condition would be imposed in this case by defining ~~imposing~~ $w_2[-1] = 0$.

* Note :- often, manipulation of the difference equations of a flow graph is difficult when dealing with the time domain variables, owing to the feedback of delayed variables.

In such cases, working with Z-transform is easier, since the delay branchs become simple gain by multiplying by z^{-1}.

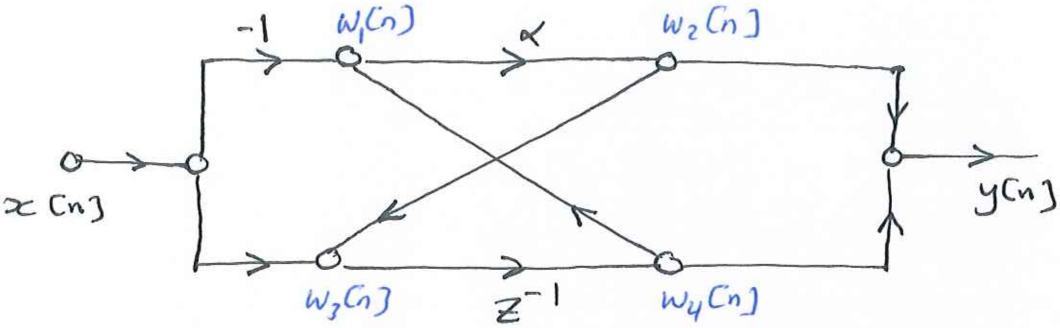
Example 6.3

a) Determine the system Function from the given graph flow

b) Find the Difference equation

c) Find the impulse response (assuming that the system is causal)

d) plot the Direct Form I Flow graph.



Note that the flow graph is not in direct form I or II, hence, we can't write $H(Z)$ by inspection

the set of difference equations represented by the graph and the corresponding z-transform equations :-

$$w_1[n] = w_4[n] - x[n] \qquad W_1(Z) = W_4(Z) - X(Z)$$
$$w_2[n] = \alpha\, w_1[n] \qquad W_2(Z) = \alpha\, W_1(Z)$$
$$w_3[n] = w_2[n] + x[n] \quad \xleftrightarrow{Z} \quad W_3(Z) = W_2(Z) + X(Z)$$
$$w_4[n] = w_3[n-1] \qquad W_4(Z) = Z^{-1} W_3(Z)$$
$$y[n] = w_2[n] + w_4[n] \qquad Y(Z) = W_2(Z) + W_4(Z)$$

We can eliminate $W_1(Z)$ and $W_3(Z)$ from this set of equations :-

sub ① into ② $\Rightarrow$ $W_2(Z) = \alpha\,(W_4(Z) - X(Z))$ —— Ⓐ

sub ③ into ④ $\Rightarrow$ $W_4(Z) = Z^{-1}(W_2(Z) + X(Z))$ —— Ⓑ

write ⑤ as it is $\Rightarrow$ $Y(Z) = W_2(Z) + W_4(Z)$ —— Ⓒ

Solve Ⓐ and Ⓑ for $W_2(Z)$ and $W_4(Z)$

$$\Rightarrow W_2(Z) = \frac{\alpha(Z^{-1} - 1)}{1 - \alpha Z^{-1}} X(Z) \quad , \quad W_4(Z) = \frac{Z^{-1}(1-\alpha)}{1-\alpha Z^{-1}} X(Z)$$

$$\Rightarrow \; Y(z) = \left[\frac{\alpha(z^{-1}-1)+z^{-1}(1-\alpha)}{1-\alpha z^{-1}}\right] X(z) = \frac{z^{-1}-\alpha}{1-\alpha z^{-1}} X(z)$$

$$\Rightarrow \; H(z) = \frac{z^{-1}-\alpha}{1-\alpha z^{-1}}$$

assume causality, the impulse response.

$$h[n] = \alpha^{n-1}u[n-1] - \alpha.\alpha^{n}u[n] = \alpha^{n-1}u[n-1] - \alpha^{n+1}u[n]$$

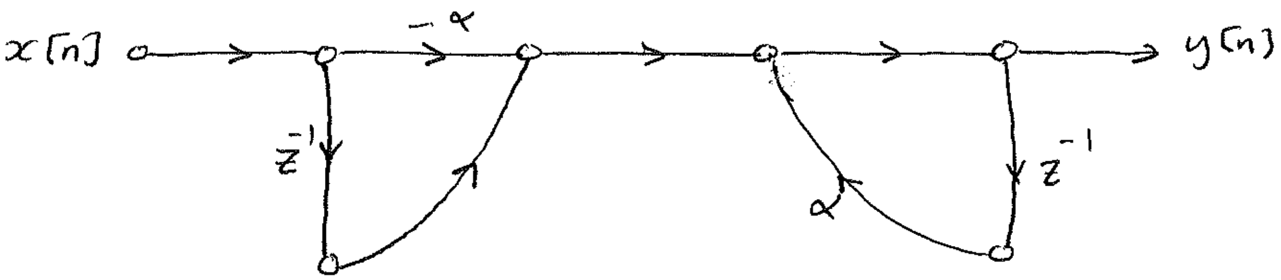
the difference equation

$$y[n] - \alpha y[n-1] = -\alpha x[n] + x[n-1]$$

the recursive equation

$$y[n] = +\alpha y[n-1] - \alpha x[n] + x[n-1]$$

the direct form I flow graph



Notes:-

* By Comparing the original implementation with the direct form I

	original	Direct form I	Direct form II
Mult.	1	2	2
Add.	3	2	2
Memory	1	2	1

* In this example, the z-transform converts the t-domain expressions, which involve feedback and thus are difficult to solve, into linear equations that can be solved by algebraic techniques.

6.3 Basic Structures for IIR systems

* This section demonstrates that for any given rational system function, a wide variety of equivalent sets of difference equations or network structures exist.

* One consideration in the choice among these different structures is computational complexity.

for example, in some digital implementations, structure with the fewest constant multipliers and delay branches are often most desirable.
(Mult. is time - consuming and costly operation in digital hardware.
also, each delay element corresponds to a memory register.)

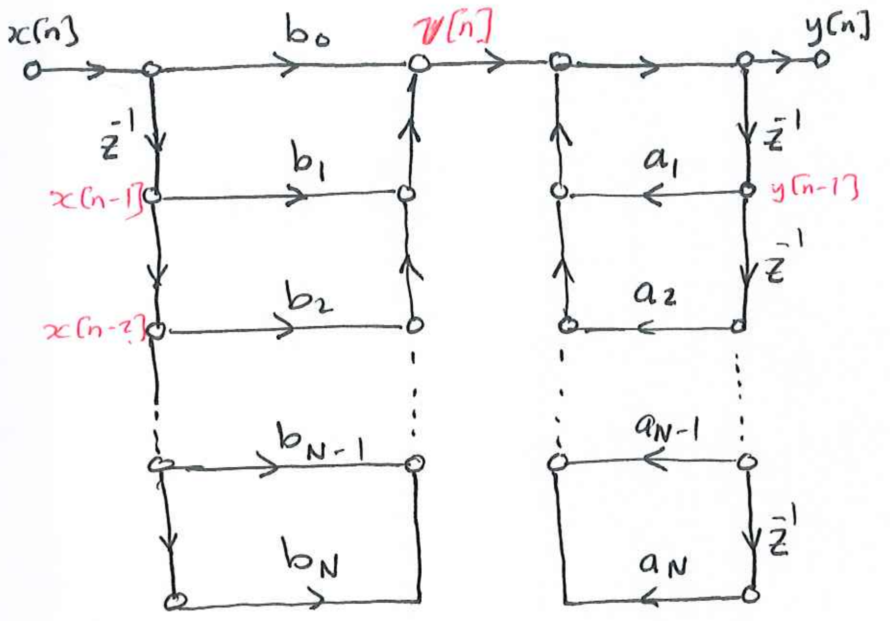
$$\Rightarrow \begin{cases} \text{less constant Multipliers} \Rightarrow \text{increase in speed} \\ \text{less delay elements} \Rightarrow \text{less memory requirments.} \end{cases}$$

In this section, we develop most common used forms for implementing an LTI IIR system and obtain their flow graph representation.

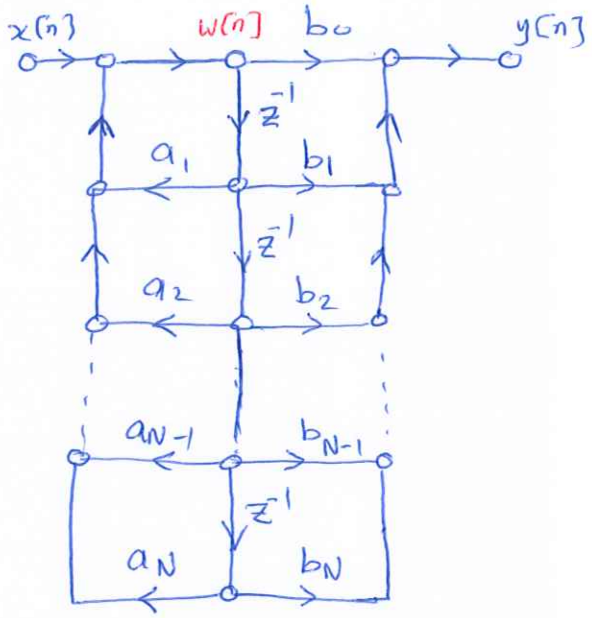
$\boxed{1}$ Direct Forms (I and II)

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

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



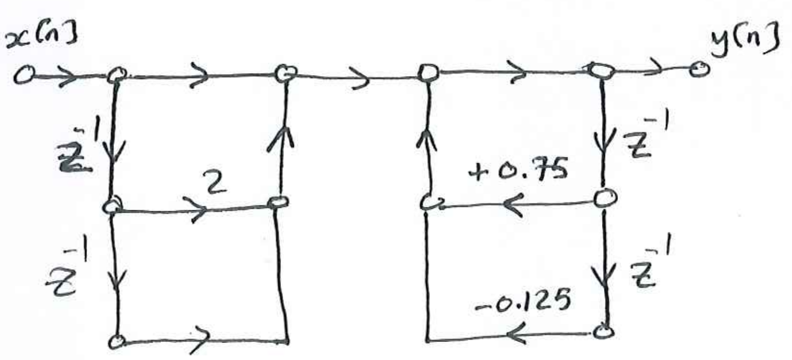
Signal Flow graph of direct form I



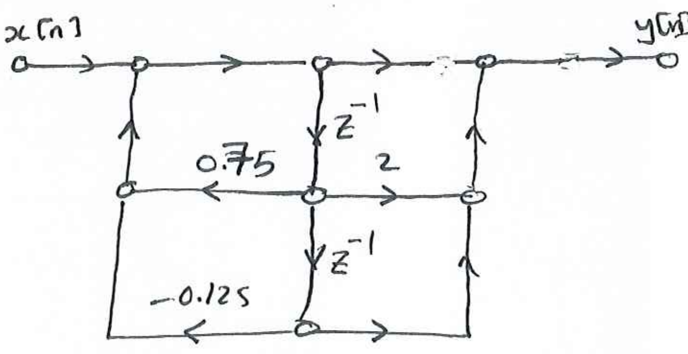
Direct Form II (Flow graph)

Example 6.4 plot signal Flow graph (Direct form I and II) for the system function $H(z) = \frac{1+2z^{-1}+z^{-2}}{1-0.75z^{-1}+0.125z^{-2}}$

solution: By inspection we can draw directly (taking into account the minus sign in the denominator coeffecients).



Direct Form I



Direct Form II

$\boxed{2}$ Cascade Form

If we factor the numerator and denominiator polynomial of $H(z)$ in the form:.

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

where
$$M = M_1 + 2M_2$$
$$N = N_1 + 2N_2$$

* The 1^{st} order factors represent real zeros at (f_k) and real poles at (c_k)

* The 2^{nd} order factors represent complex conjugate pairs of zeros at g_k and g_k^{*} and " " " of poles at d_k and d_k^{*}.

* This representation suggests a class of structures consisting of a cascade of 1^{st}-order and 2^{nd}-order systems.

* There is a freedom in the choice of composition of the subsystems and in the order in which the subsystems are cascaded.

* In practice, it is often desirable to implement the cascade realization using a minimum of storage and computation.

* The structure that is advantageous for many types of implementation is obtained by combining pairs of real factors and complex conjugate pairs into 2nd-order factors so that equation ⊛ can be expressed as

$$H(z) = \prod_{k=1}^{N_s} \frac{b_{0k} + b_{1k} z^{-1} + b_{2k} z^{-2}}{1 - a_{1k} z^{-1} - a_{2k} z^{-2}} \qquad \left(\text{assumeing that } M \leq N \right)$$

where $N_s = \lfloor (N+1)/2 \rfloor$ is the largest integer contained in $\left(\frac{N+1}{2}\right)$.

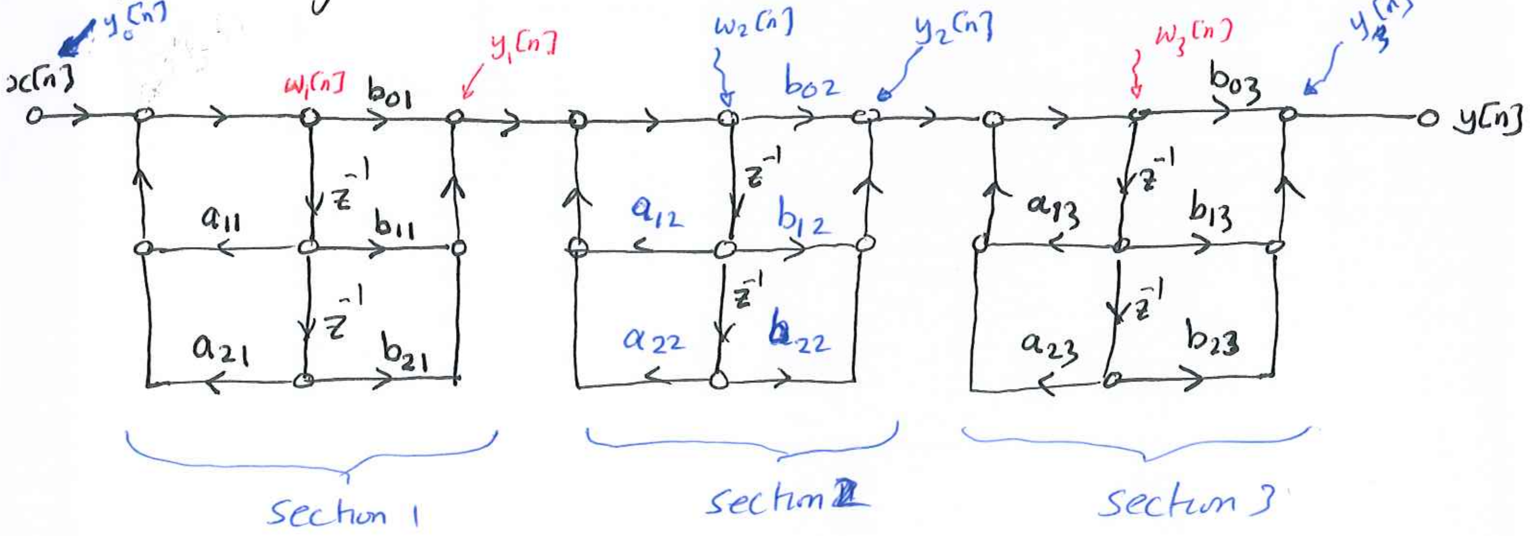
$$\left\lfloor \frac{3+1}{2} \right\rfloor = \lfloor 2 \rfloor = 2$$

$$\left\lfloor \frac{4+1}{2} \right\rfloor = \lfloor 2.5 \rfloor = 2$$

Notes:

* If there are an odd number of real zeros, one of the coeffeceints $b_{2k} = 0$
* " " " " " " " " poles, " " " " $a_{2k} = 0$
* The individual 2nd-order sections (subsystems) can be implemented using either of the direct form structures. (However, Form II is prefared over Form I due to less number of registers).

Example: consider the cascade structure for a 6th-order system using three direct form II 2nd order sections



$N = 6$

$N_s = \lfloor (6+1)/2 \rfloor = 3$ sections

The difference equations represented by a general cascade of direct form II 2nd-order sections are of the form

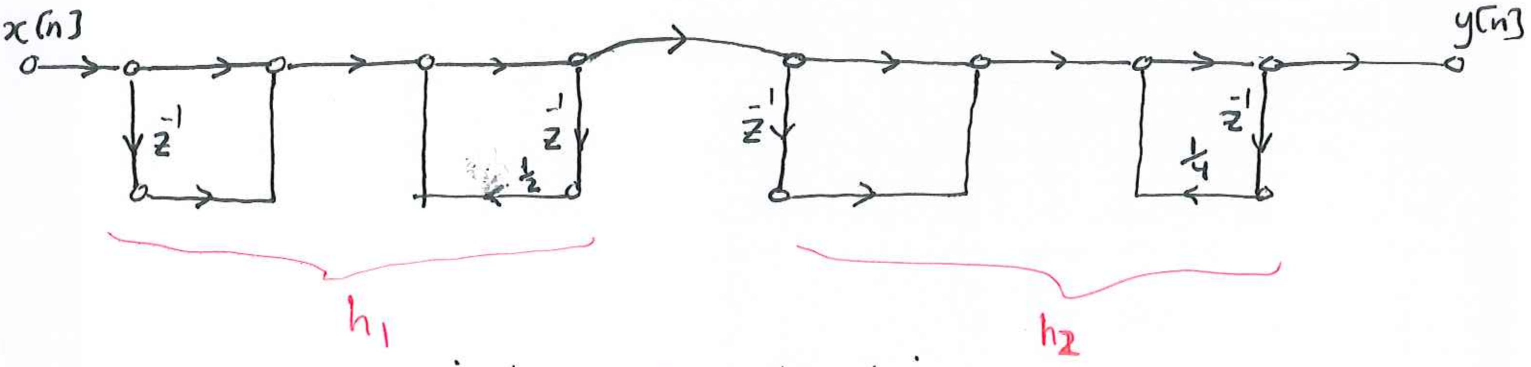
$$y_0[n] = x[n]$$
$$w_k[n] = a_{1k} w_k[n-1] + a_{2k} w_k[n-2] + y_{k-1}[n] \quad , k=1,2,..,N_s$$
$$y_k[n] = b_{0k} w_k[n] + b_{1k} w_k[n-1] + b_{2k} w_k[n-2] \quad , k=1,2,..,N_s$$
$$y[n] = y_{N_s}[n]$$

* It is easy to see that a variety of theoretically equivalent systems can be obtained by simply paring the poles and zeros in different ways and by ordering the 2nd-order sections in different ways.

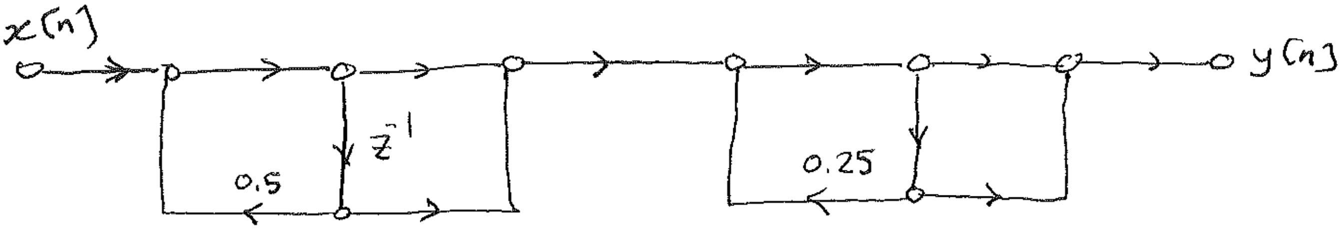
Example 6.5 : consider the 2nd order system discussed in the previous example (6.4)

$$H(z) = \frac{1+2z^{-1}+z^{-2}}{1-0.75z^{-1}+0.125z^{-1}} = \frac{(1+z^{-1})}{(1-\frac{1}{2}z^{-1})} \cdot \frac{(1+z^{-1})}{(1-\frac{1}{4}z^{-1})}$$

To illustrate the cascade structure, we can use 1st-order systems by expressing H(z) as a product of 1st-order factors.



Direct Form I subsections



Direct Form II subsections

Note: * since all of the poles and zeros are real, a cascade structure with 1st-order sections have real coefficients.

* If the poles and/or zeros were complex, only a 2nd-order section would have real coefficients.

---

3 Parallel Form

As an alternative to factoring the numerator and denominator polynomials of $H(z)$, we can express a rational system function as a partial fraction expansion in the form

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

where

$N = N_1 + 2N_2$

if $M \geq N$, then $N_p = M - N$

if $M < N$, then $N_p = 0$ (the first term is not included)

In this form, the system function can be interpreted as representing a parallel combination of 1st and 2nd-order IIR systems, with possibly N_p simple scaled delay paths.

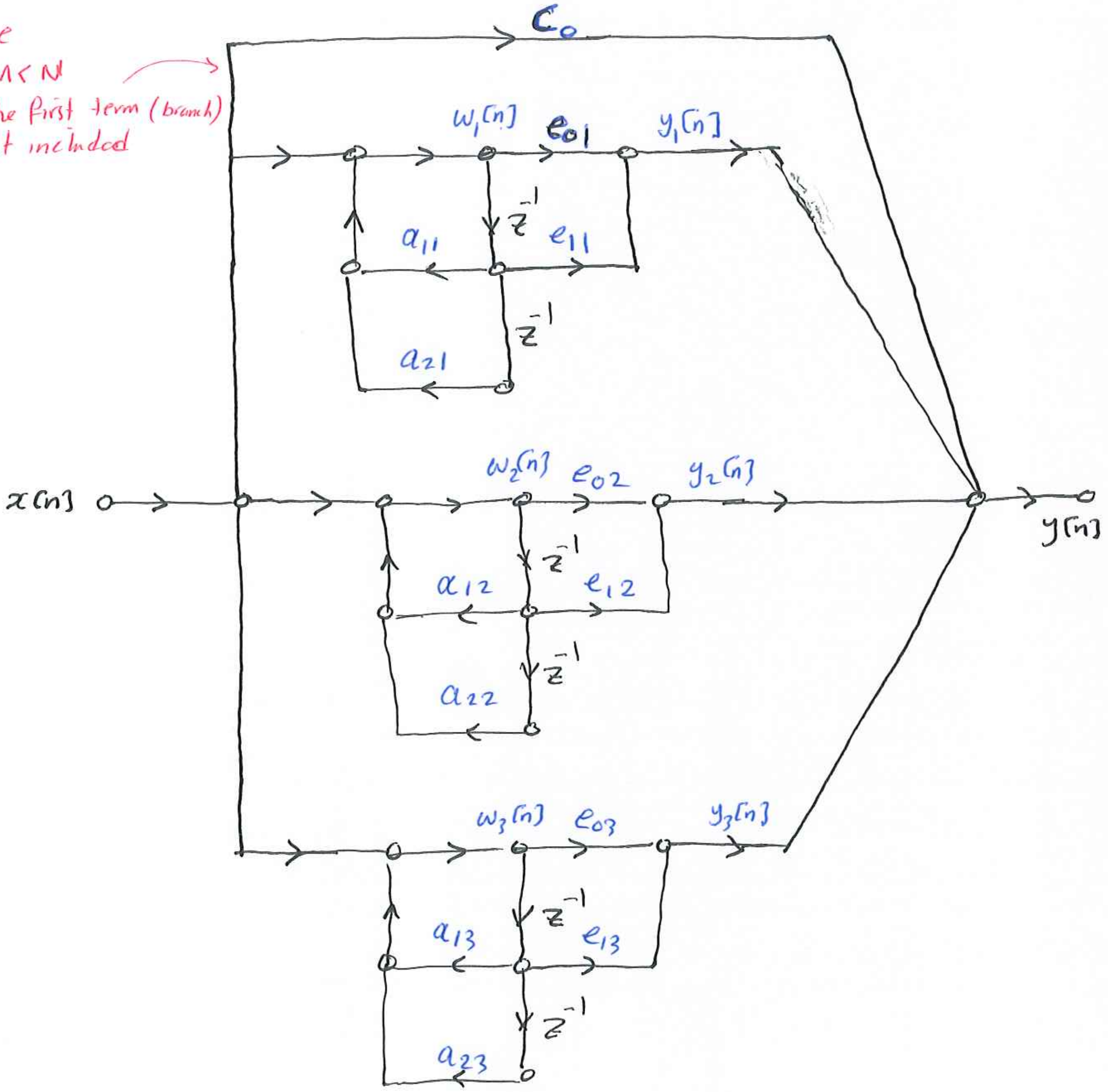
Alternatively, we may group the real poles in pairs, so that $H(z)$ can be expressed as

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

where, $N_s = \lfloor (N+1)/2 \rfloor$ is the largest integer contained in $(N+1)/2$

Example : let $M = N = 6$

then $N_s = \lfloor 7/2 \rfloor = 3$ 2nd order sections



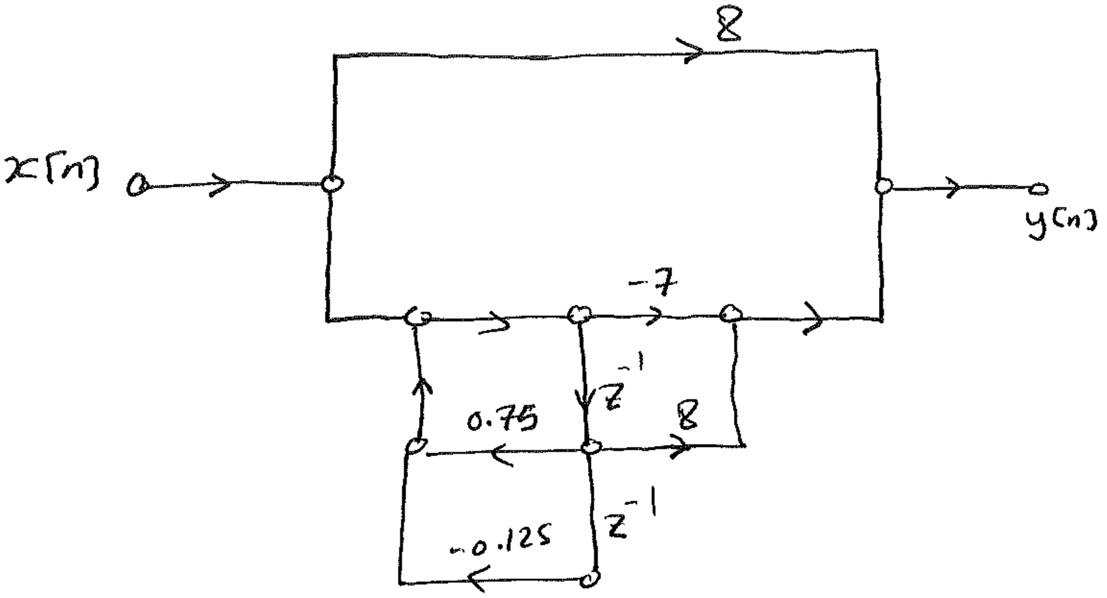
Note
If $M < N$
then, the first term (branch) is not included

Parallel Form structure with the real and complex poles grouped in pairs (6th order system $M = N = 6$)

Example 6.6 consider again the system function used in examples 6.4 and 6.5. ~~H(z)~~

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

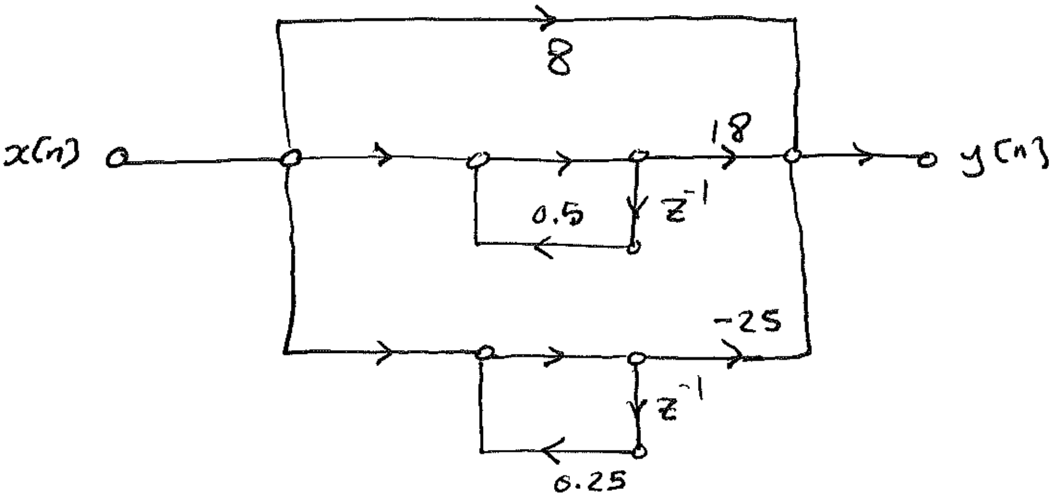
then, the parallel form realization with 2nd order section



since all the poles are real, we can obtain an alternative parallel form realization by expanding H(z) (partial fraction Expn.)

$$H(z) = 8 + \frac{18}{1-0.5z^{-1}} - \frac{25}{1-0.25z^{-1}}$$

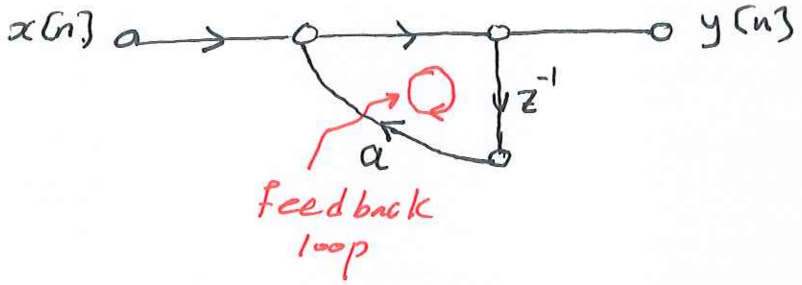
then, the parallel form realization with 1st-order section



4 Feedback in IIR systems

- Flow graph with feedback loops implies that a node variable in a loop depends directly or indirectly on itself.
- A feedback loops are closed paths that begin at a node and return to that node.

Example: $y[n] = a\, y[n-1] + x[n]$



- such loops are necessary (but Not sufficient) to generate infinitely long impulse responses.

- for a network with no feedback loop, the impulse response is no longer than the total number of delay elements

⇒ For Network with no loops, then the system function has only zeros (except poles at $z=0$), and the number of zeros can be no more than the number of delay elements in the network.

Example: for $y[n] = a y[n-1] + x[n]$

let $x[n] = \delta[n]$

$\Rightarrow h[n] = a^n u[n]$

this illustrates how feedback can create an infinite long impulse response.

$y[0] = 0 + 1 = 1$

$y[1] = a y[0] + 0 = a$

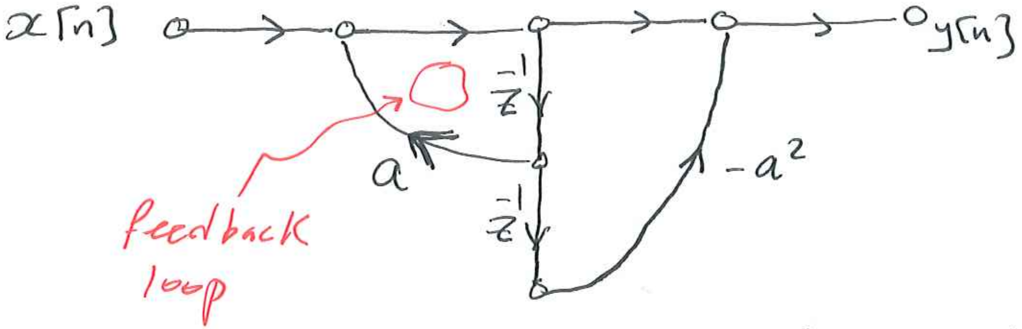
$y[2] = a y[1] + 0 = a^2$

$\vdots$

$y[n] = a^n$

* If the system function has poles, then the block diagram or signal flow graph will have feed back loop.

* On the other hand, neither poles in the system function nor loops in the network are sufficient for the impulse response h[n] to be infinitely long.



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

it is an FIR system with feedback loop!

$h[n] = \delta[n] + a\delta[n-1]$

h[n] is of finite length.

the pole of H(z) cancels with a zero.

6.4 Transposed Forms

- * flow graph reversal or transposition leads to a set of transposed system structures that provide some useful alternatives to structures discussed in the previous sections (Form I, II, cascade, ..) . ~~The overall system.~~
- * The overall system function between input and output is unchanged . (The formall proofe is not required !)

* Steps :-

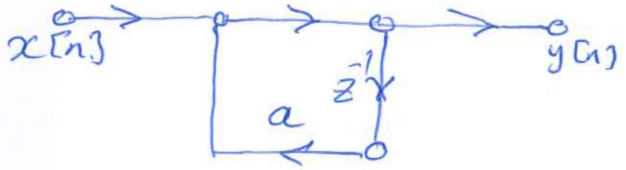
① Reverse the directions of all branches in the network

② Reverse the roles of the input and output so that the source nodes become sink nodes and vice versa .

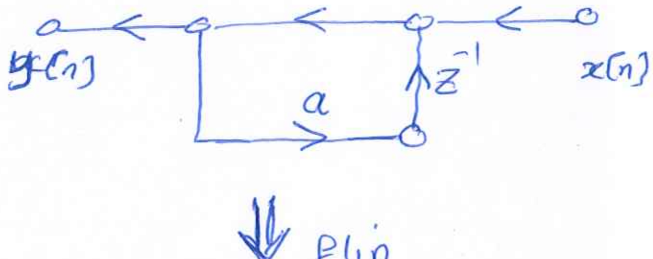
③ you can flip the result to make input in the LHS ,

Example 6.7

consider the following flow graph (1^{st} order with no zeros)



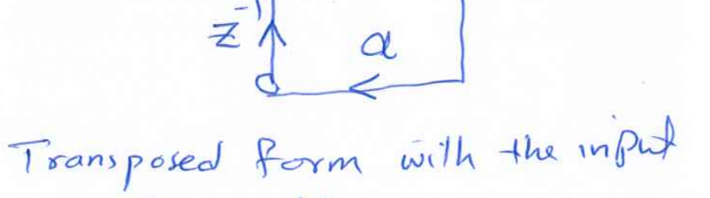
Reverse $\Longrightarrow$



$y[n] = a\,y[n-1] + x[n]$

$H(z) = \dfrac{1}{1 - a z^{-1}}$

$\Downarrow$ Flip



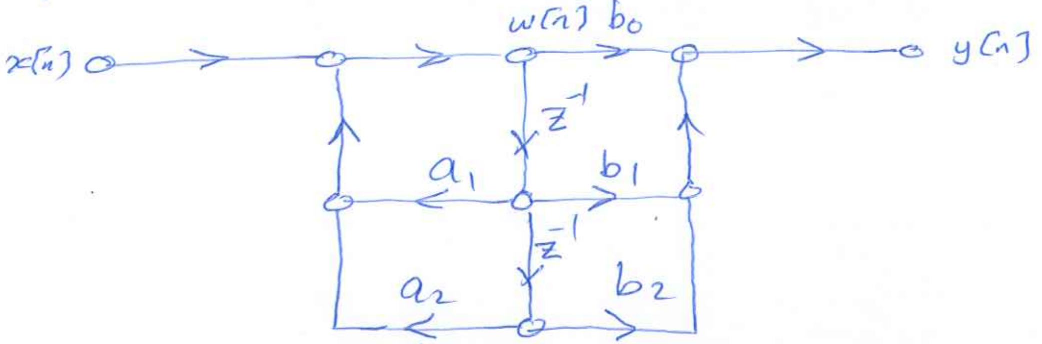
* The only difference is that in the original we Mult. the delayed output by a . while in the transposed form we mult. the output y[n] by a and then delay the resulting product.

Transposed form with the input on the left .

$y[n] = a\,y[n-1] + x[n]$

$H(z) = \dfrac{1}{1 - a z^{-1}}$

Example 6.8 consider the 2nd order section

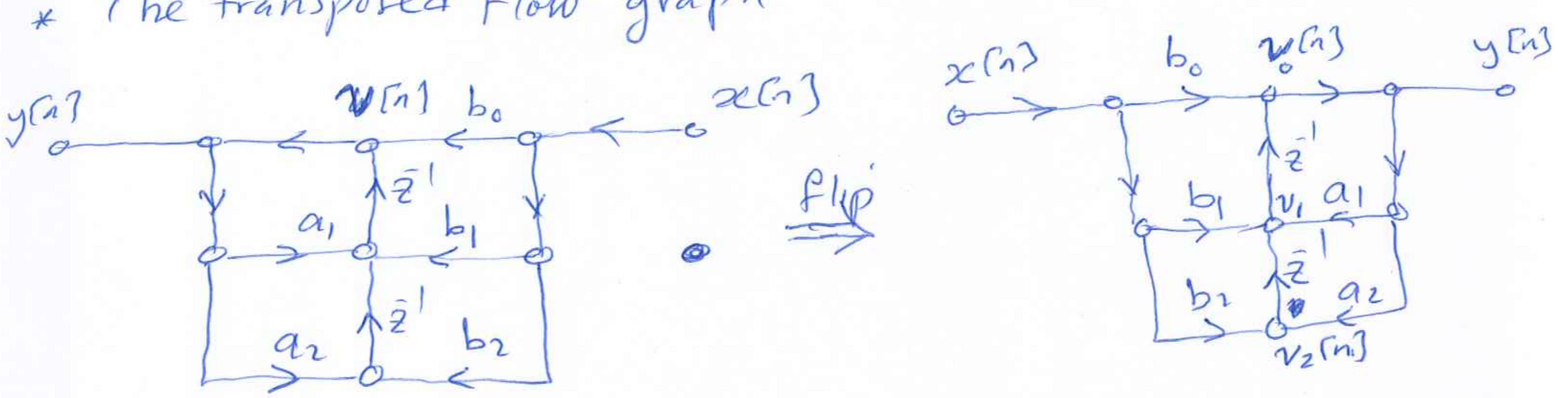


Direct form II structure.

* The corresponding difference equations :-

$$\left.\begin{aligned} w[n] &= a_1 w[n-1] + a_2 w[n-2] + x[n] \\ y[n] &= b_0 w[n] + b_1 w[n-1] + b_2 w[n-2] \end{aligned}\right) \text{set ①}$$

* The transposed flow graph



* its corresponding difference equations are

$$\left.\begin{aligned} v_0[n] &= b_0 x[n] + v_1[n-1] \\ y[n] &= v_0[n] \\ v_1[n] &= a_1 y[n] + b_1 x[n] + v_2[n-1] \\ v_2[n] &= a_2 y[n] + b_2 x[n] \end{aligned}\right) \text{set ②}$$

* equations set ① and ② are different ways to organize the compulation of y[n] from x[n].

* to show that two structures are equivalent, we can use z-transform of both sets to find H(z). or we can use substitution in set ② to show that the result is equivalent to set ①.

* The transposition theorem can be applied to any of the structures that we have discussed so far.

* An important point becomes clear through comparing the two structures in example 6.8. Whereas the direct form II structure implements the poles first and then the zeros, the transposed direct form II structure implements the ~~poles~~ zeros first and then the poles.

* When the transposition theorem is applied to cascade or parallel structures, the individual 2nd order systems are replaced by the transposed structures.

6.5 Basic Network structures for FIR systems

$\boxed{1}$ Direct Form

* For causal FIR systems, the system function has only zeros (except for poles at $z = 0$) ⇒ a_k's all are zeros.

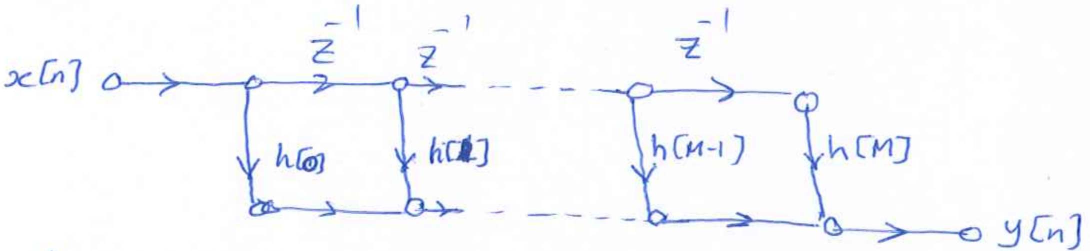
⇒ the difference equation:

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

thi can be recognized as the convolution sym of $x[n]$ with the impulse response $h[n]$

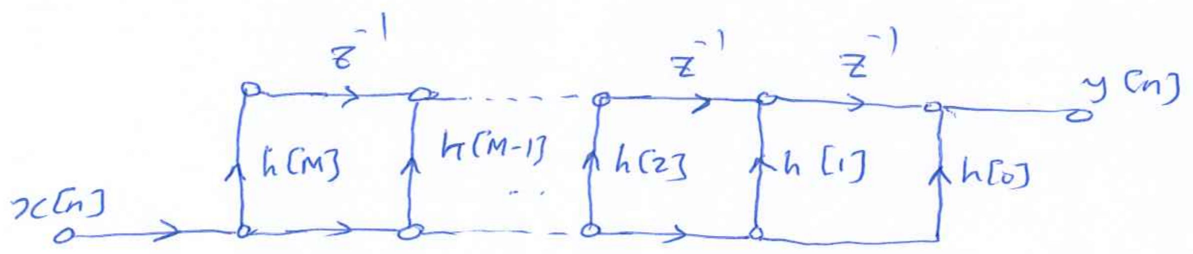
$$\Rightarrow \quad h[n] = \begin{cases} b_n & n = 0, 1, \cdots, M \\ 0 & \text{else} \end{cases}$$

⇒ Direct Form I and II reduce to the direct form FIR structure



Because of the delay element across the top of the diagram this structure is referred to as "tapped delay line" or a "transversal filter" structure.

* Applying the transposition theorem, we get the transposed direct form for the FIR



$\boxed{2}$ Cascade Form

* The cascade form for the FIR systems is obtamed by factoring the polynomial system function :

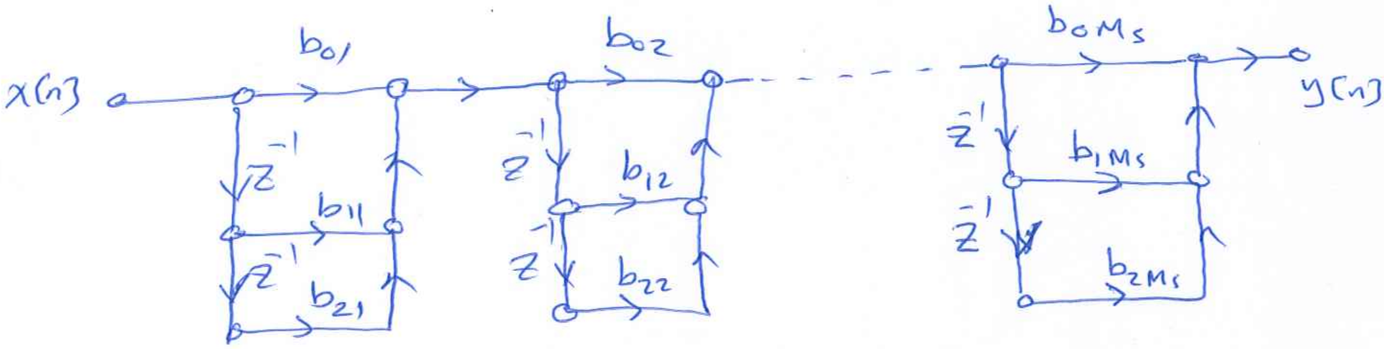
$$H(Z) = \sum_{k=0}^{M} h[n] z^{-k}$$

$$= \prod_{k=1}^{M_s} (b_{0k} + b_{1k} z^{-1} + b_{2k} z^{-2})$$

where $M_s = \lfloor (M+1)/2 \rfloor$ is the largest integer containld in $(M+1)/2$.

* if M is odd, one of the coeffecients b_{2k} will be zero.

* the corresponding flow graph (2^{nd}-order sections uses direct form)



cascade realization of an FIR system

---

* We have seen that a particular LTI system can be implemented by a variety of computational structures.

* One motivation for considering alternatives is that different structures that are equivalent for infinite precision arithmatic may behave differently when implemented with finite numerical precision. For more details referrs to sec. 6.7 / P. 443.