COMP ENG 4TL4:

Digital Signal Processing

Notes for Lecture #26

Friday, November 7, 2003

6.9 Structures for Digital Filters

A digital filter described by a particular LCCD equation (or the corresponding z-domain transfer function) may be implemented in a DSP using a variety of standard structures made up of an interconnection of basic operations of addition, multiplication by a constant, and unit delays.

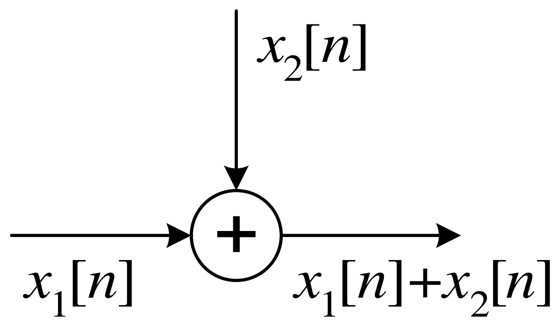
These structures may differ in:

- the number of basic operations required to implement a particular filter,
- their sensitivity to quantization of filter coefficient values (for finite-precision arithmetic), or
- their sensitivity to round-off noise because of finite-precision arithmetic.

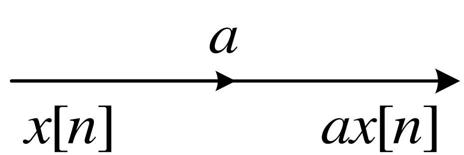
As a tool for investigating these different structures, we will utilize a block diagram representation of LCCD equations.

Block diagram representation of LCCD equations:

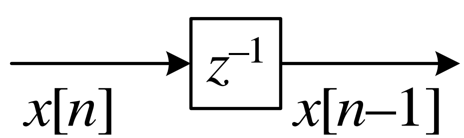
Using the basic building blocks shown below, a block diagram can be constructed to describe any LCCD equation.



Addition of two sequences



Multiplication of a sequence by a constant

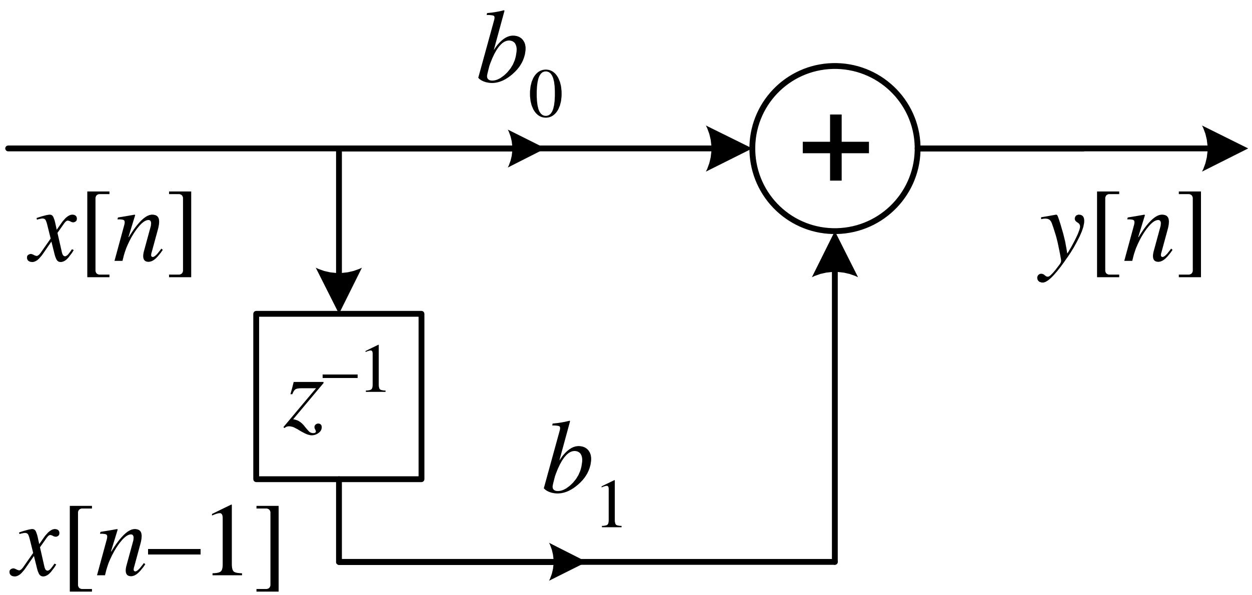


Unit delay

Example #1: 1st-order FIR filter:

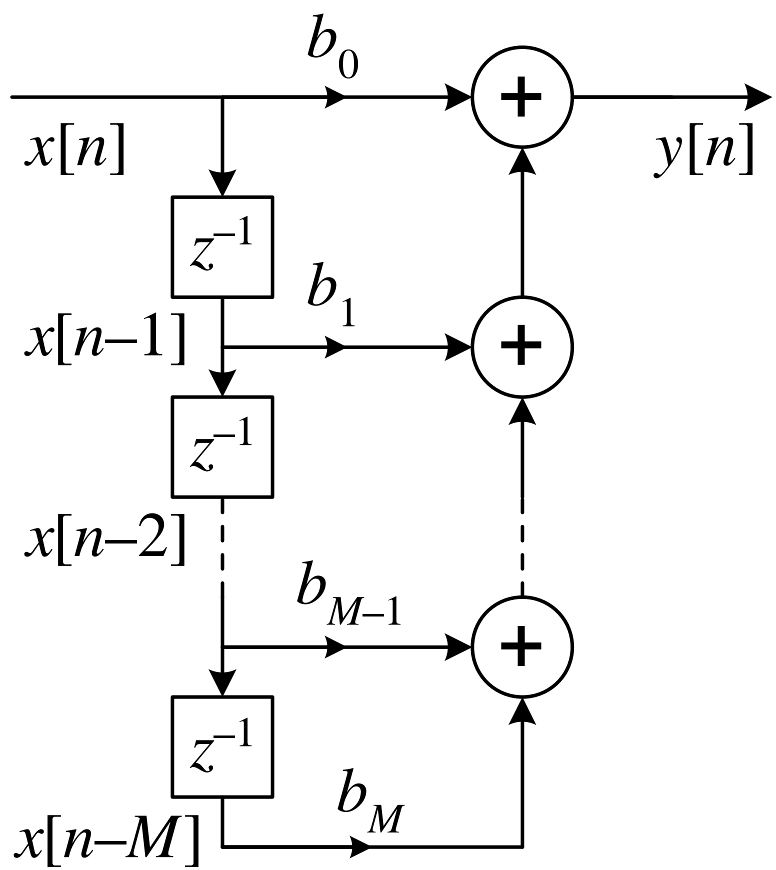
$$y[n] = b_0 x[n] + b_1 x[n-1]$$

$$\Rightarrow \quad H(z) = b_0 + b_1 z^{-1}.$$



This block diagram can be generalized to a higher-order FIR filter of the form:

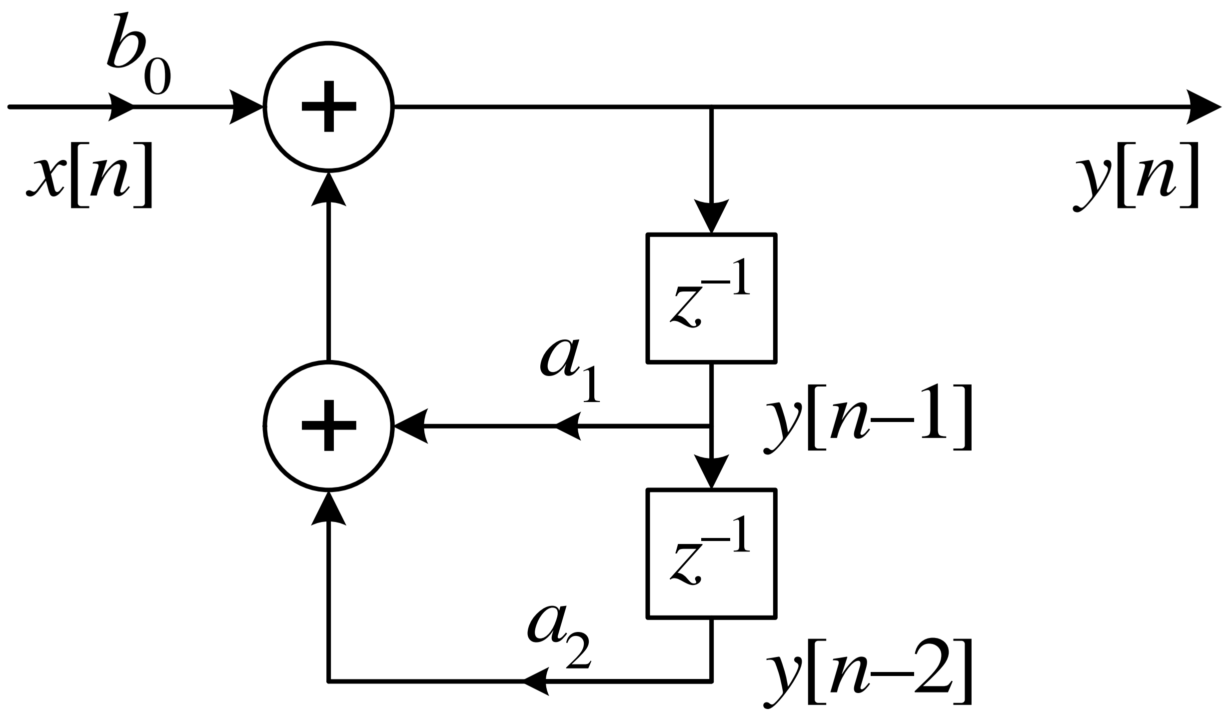
$$y[n] = \sum_{k=0}^{M} b_k x[n-k] \;\Rightarrow\; H(z) = \sum_{k=0}^{M} b_k z^{-k}.$$



Example #2: 2nd-order IIR filter:

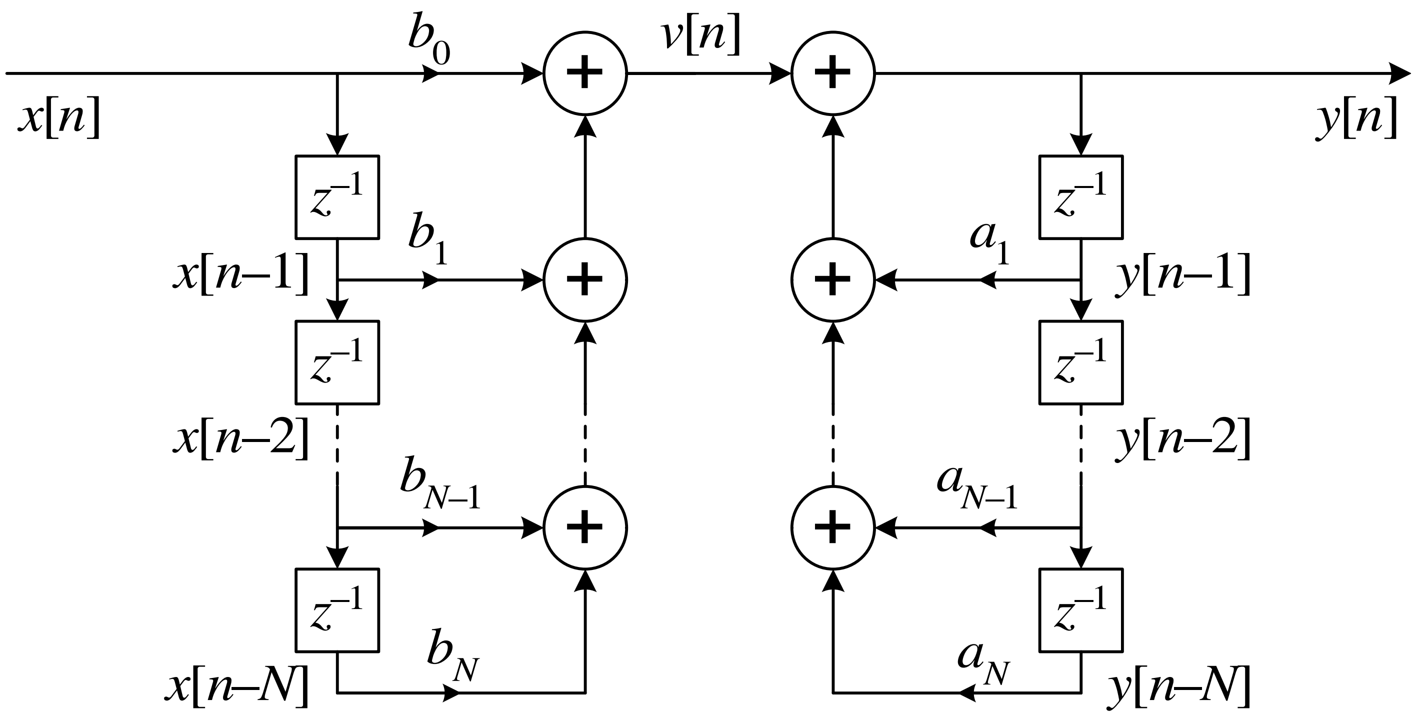
$$y[n] = a_1 y[n-1] + a_2 y[n-2] + b_0 x[n]$$

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



<u>Direct form I:</u> These block diagrams can be generalized to a higher-order difference equations of the form:

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



direct form I/
noncanonical form

This difference equation form is related to the standard ARMA system equation:

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

via the relationships:

$$b_k \;=\; b[k]\,, \qquad\qquad k = 0, \ldots, N\,,$$

$$a_k \;=\; \begin{cases} 1, & k = 0, \\ -a[k]\,, & k = 1, \ldots, N\,, \end{cases}$$

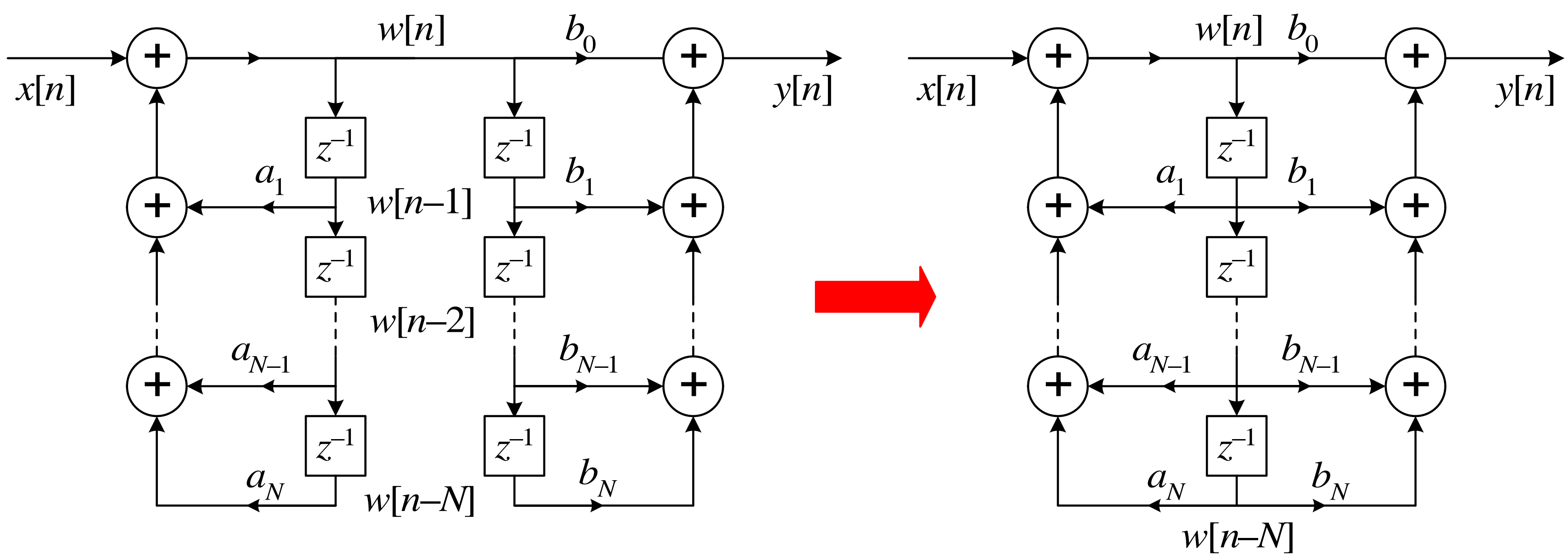
for the case of $M = N$.

If $M \neq N$ in the standard ARMA equation, then the order N of the *direct form I* equation should be set to $\max(M,N)$ and the appropriate coefficients a_k or b_k should be set to zero to achieve equivalence.

Direct form II: Note that *direct form I* can be viewed as two separate LTI subsystems placed in series, requiring a total of $2N$ unit delays.

Reversing the order of LTI subsystems does not affect the overall transfer function, so the unit delays from each subsystem can be combined, requiring a total of only N unit delays.

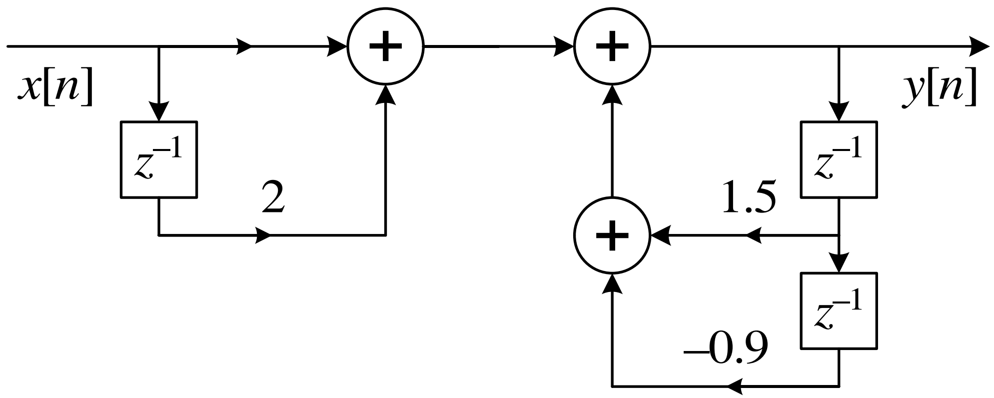
direct form II/
canonical form



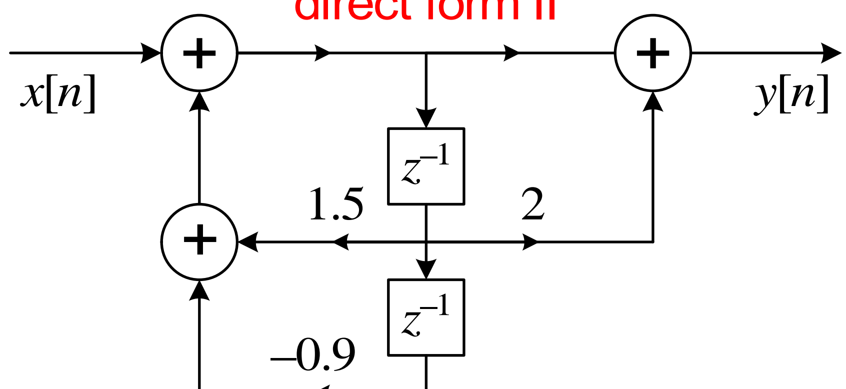
Example #3: 2nd-order IIR filter:

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

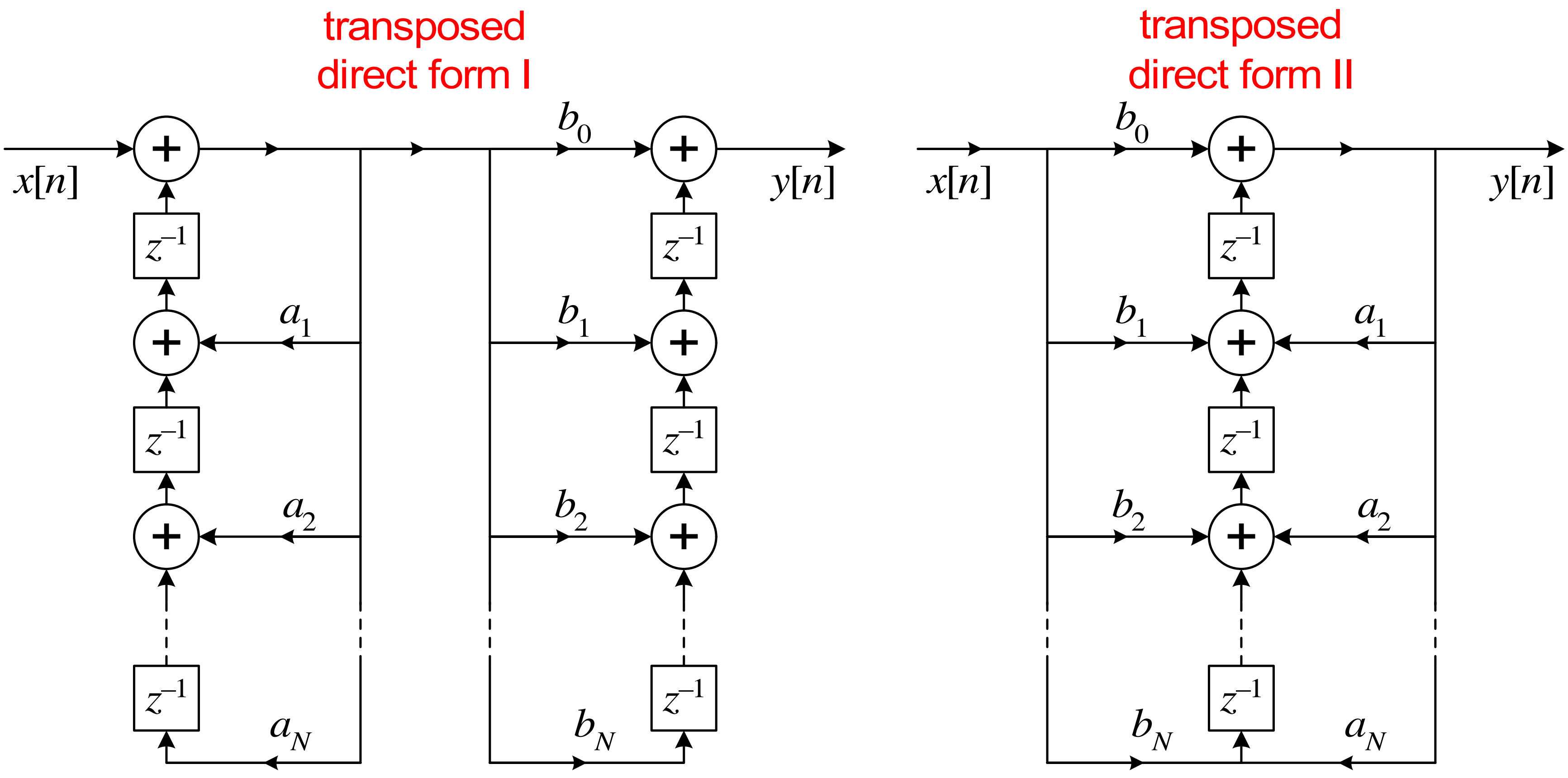
direct form I



direct form II



Transposed forms: It is also possible to reverse the order of *all the operations*, leading to *transposed* direct forms I and II.

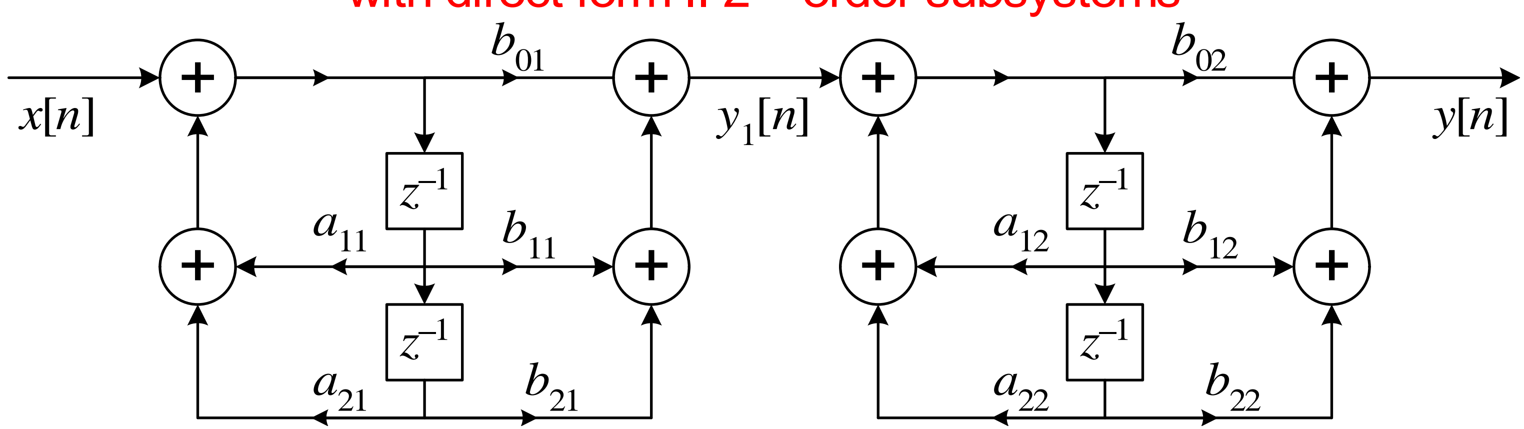


Cascade form: A high-order rational transfer function can be reformulated to be the product of 2nd-order factors, leading the 2nd-order subsystem *cascade form*:

$$H(z) = A\frac{\prod_{k=1}^{N}\left(1 - c[k]\,z^{-1}\right)}{\prod_{k=1}^{N}\left(1 - d[k]\,z^{-1}\right)} = A\prod_{k=1}^{N_s}\frac{b_{0k} + b_{1k}z^{-1} + b_{2k}z^{-2}}{1 - a_{1k}z^{-1} - a_{2k}z^{-2}}.$$

where N_s is the largest integer contained in $(N+1)/2$.

4th-order filter implemented in cascade form
with direct form II 2nd-order subsystems



Parallel form: Partial fraction expansion of a transfer function leads to the *parallel form*, e.g.:

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

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

parallel form using
1st-order subsystems