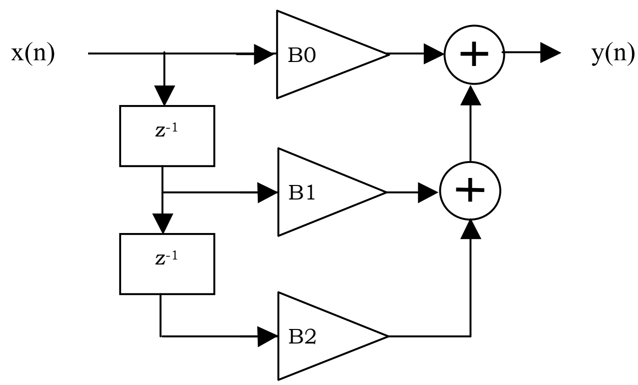
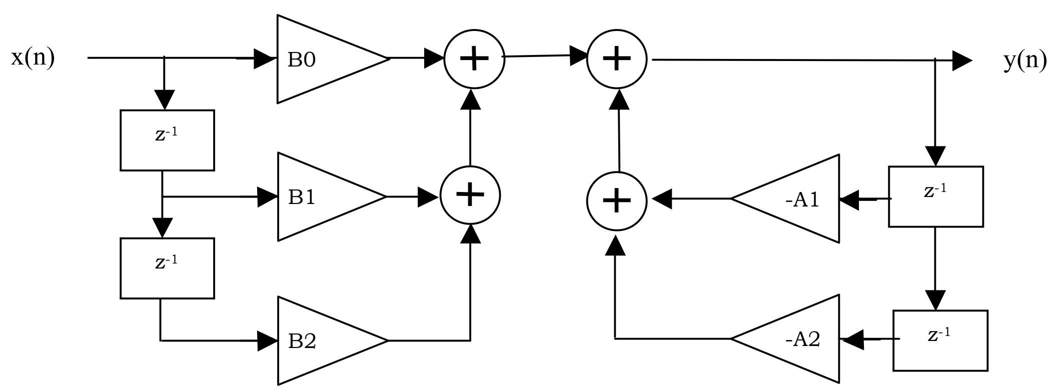
Hardware Realizations of Difference Equations – DePiero

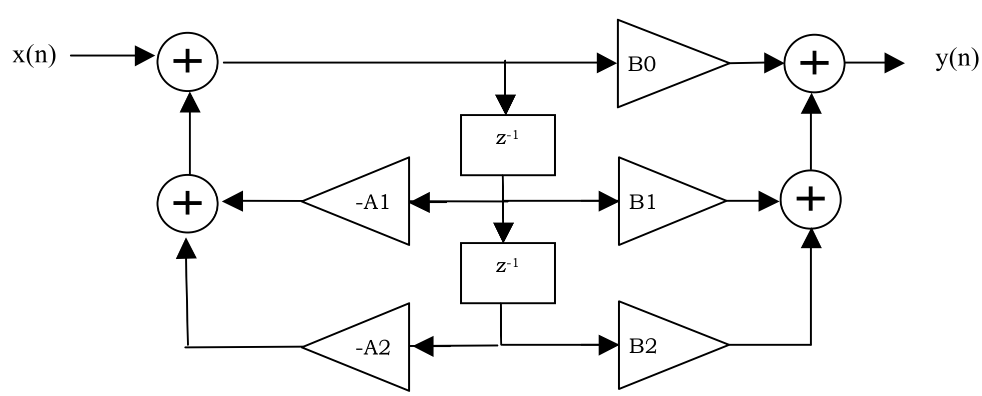
The following examples are shown for 2^{nd} order systems. The diagrams below contain unit delays (denoted by z^{-1}), as well as multipliers (denoted by Bk and Ak), and adders.



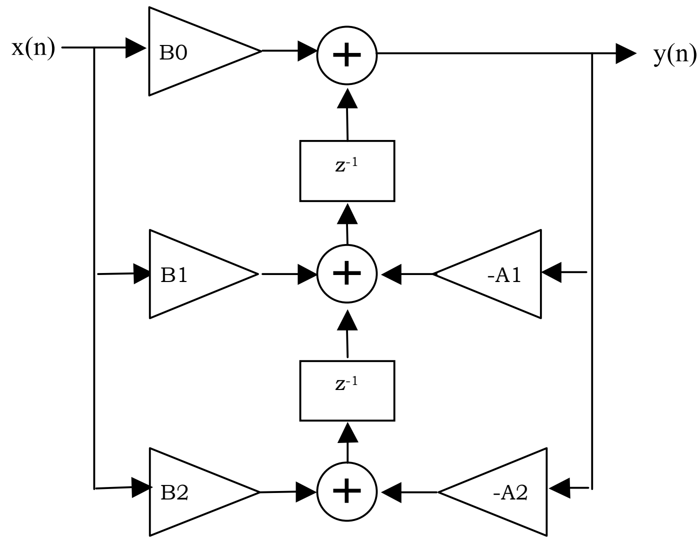
Direct Form I, for a 2^{nd} order FIR system



Direct Form I, for a 2^{nd} order IIR system



Direct Form II, for a 2^{nd} order IIR system



Direct Form II Transposed, for a 2nd order IIR system

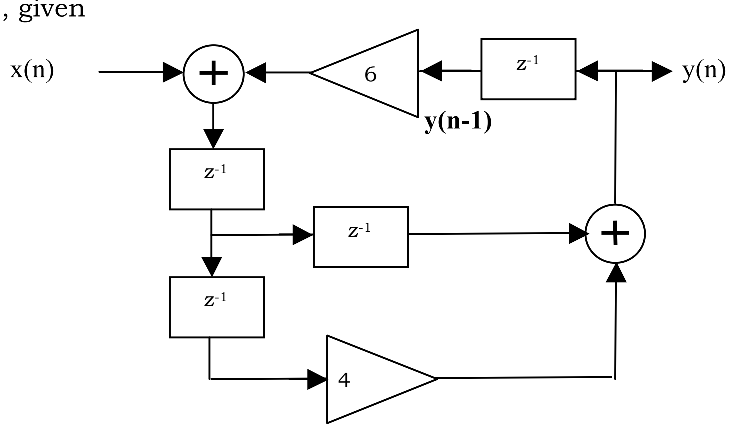
The above forms are termed 'direct' because the coefficients of a difference equation are transcribed directly into the hardware multipliers. The above forms differ in the number of delay elements (memory) and adders required. These are readily extended to higher order versions.

Finding an Equivalent Difference Equation

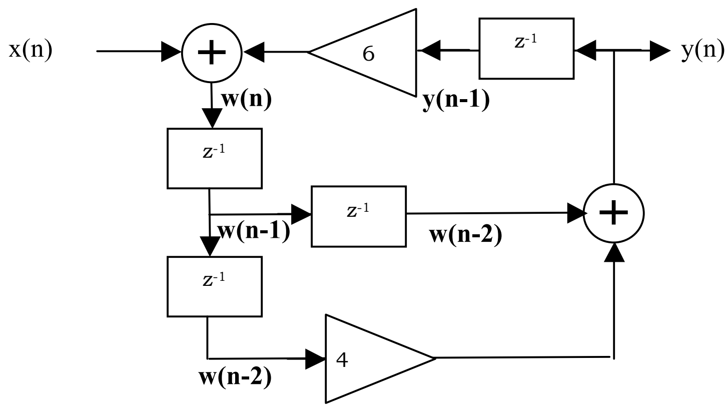
Given a hardware realization with arbitrary structure (not matching a standard form), it may be possible to describe the system using a single difference equation. This is not always true – sometimes more than one difference equation is required. A general procedure for this translation is:

1. Label delayed versions of the input and output, if present.
2. If the output can not be expressed in terms of x(n-k) and y(n-k) labels that are present, then introduce new labels. New labels could be introduced at the output of adders for example, or as deemed appropriate.
3. Algebraically eliminate any new signal labels by substitution until a difference equation involving x(n-k) and y(n-k) is obtained – if possible.

For example, given



There is only one signal, y(n-1), that may be labeled as simply a delayed version of x(n) or y(n). Next, introduce w(n) and its delayed versions, as shown below.



Hence the following expressions may now be identified from the diagram:

y(n) = w(n-2) + 4 w(n-2)
w(n) = x(n) + 6 y(n-1)

Substituting for w(n) yields

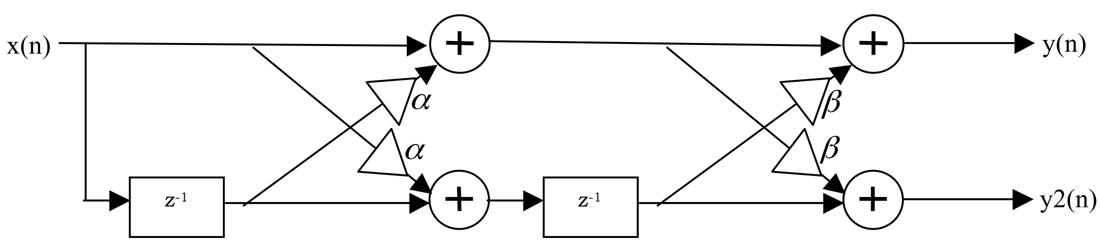
y(n) = 5 w(n-2) = 5 [x(n-2) + 6 y(n-3)] = 5 x(n-2) + 30 y(n-3)

Comparing to the standard form of a difference equation reveals, B2=5 and A3 = -30.

Lattice Filters

Lattice filters are an advanced type of structure. Lattice filters are not a direct realization, meaning that the multiplying factors in the structure are not simply the Ak and Bk values. IIR lattice filters have an advantage over direct realizations, generally being less sensitive to the effects of coefficient rounding. When coefficients are rounded, this implies a change in the pole/zero locations, which implies a change in the shape of the frequency response. Lattice filters typically suffer less change for a given amount of rounding than do direct filters.

The following structure depicts a 2^{nd} order FIR lattice filter. Conversion formulas follow, that relate difference equation coefficients to the gain parameters of the lattice.



FIR Lattice Structure, for a 2^{nd} order system having B0 = 1.0

For the above structure,

$$B0 = 1, \quad B1 = \alpha(1+\beta), \quad B2 = \beta$$

Or, rearranging

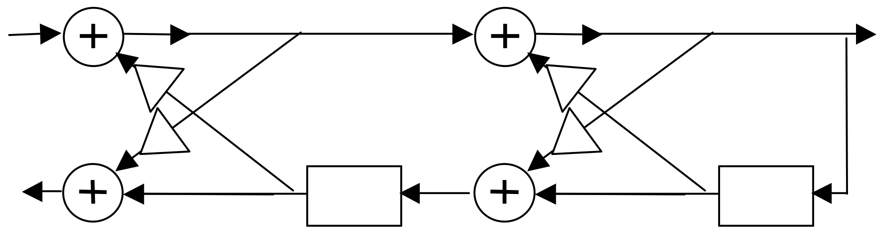
$$\alpha = B1/(1+B2), \quad \beta = B2$$

x(n) y(n)

-G -F

-G -F

y3(n) A 2^{nd} order IIR lattice filter follows, with associated conversions. z^{-1} z^{-1}



IIR Lattice Structure, for a 2^{nd} order system having B0 = 1 and Bk = 0, k>0

In this case,

$$A0 = 1, \quad A1 = F\,(1\text{-}G), \quad A2 = G$$

Or, rearranging

$$F = A1/(1\text{-}A2), \quad G = A2$$

Higher order lattice filters employ repeated stages, similar to above. However the conversion formulae are more involved.