0.0.1 Equivalent structures

Equivalent structures

Another important topic in digital filter design is the equivalence. Two digital filter structures are defined to be **equivalent** if they have the same transfer function.
In the following discussion we describe several methods for the generation of equivalent structures. However, we state now that a fairly simple way to do it from a given realization is via the **transpose operation**, which consists in:

Transpose operation

- reversing all the paths;

- replacing pick-off nodes by adders, and viceversa;

- interchanging the input and output nodes.

Let us consider an example of this technique. This is showed in Figure 1 along with the redrawn transposed structure.
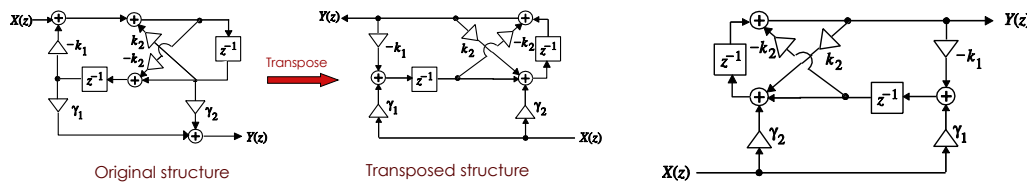

Figure 1: Example of the application of the transpose operation (left) and redrawn transposed structure (right).

We have to remark that all the other methods for developing equivalent structures are based on a specific algorithm for each structure. Moreover, there are literally an infinite number of equivalent structures realizing the same transfer function. Thus, it is impossible to develop all the equivalent realizations and in this course we will restrict our attention to a discussion of some commonly used structures.

Equivalent structures and finite precision arithmetics

Under infinite precision arithmetic, any given realization of a digital filter behaves identically to any other equivalent structure. However, in practice, due to the finite wordlength limitations, a specific realization behaves totally differently from its other equivalent realizations. Hence, it is important to choose a structure that has the least quantization effects when implemented. One way to arrive at such a structure is to determine a large number of equivalent structures, analyze the finite wordlength effects in each case, and select the one showing the least effects. In ceratin cases, it is possible to edvelop a structure that by construction has the least quantization effects. Here, we review some simple realizations that in many applications are quite adequate.

0.0.2 Basic FIR digital filter structures

Basic FIR digital filter structures

A **causal FIR filter of order** N is characterized by a transfer function $H(z)$ given by:

$$H(z) = \sum_{n=0}^{N} h[n]z^{-n} \tag{1}$$

which is a polynomial in z^{-1}. In the time-domain the input-output relation of the FIR filter in Eq. 1 is given by:

$$y[n] = \sum_{n=0}^{N} h[k]x[n-k] \tag{2}$$

It is important to remark that an FIR filter of order N is characterized by $N + 1$ coefficients and, in general, requires $N + 1$ multipliers and N two-input adders. Structures in which the multiplier coefficients are precisely the coefficients of the transfer function are called **direct form structures**.
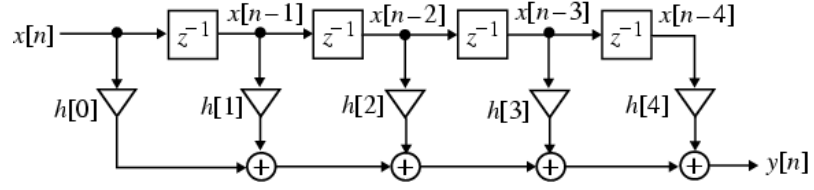

Figure 2: Example of FIR filter structure.

In particular, an analysis of the structure with $N = 4$, as showed in Figure 2, yields:

$$y[n] = h[0]x[n] + h[1]x[n - 1] + h[2]x[n - 2] + h[3]x[n - 3] + h[4]x[n - 4] \qquad (3)$$

Transversal filter

which is precisely of the form of the convolution sum description. The direct form structure in Figure 2 is also known as a **transversal filter**. The transpose of this direct form structure is showed in Figure 3. Both direct form structures are canonic with respect to the delays.
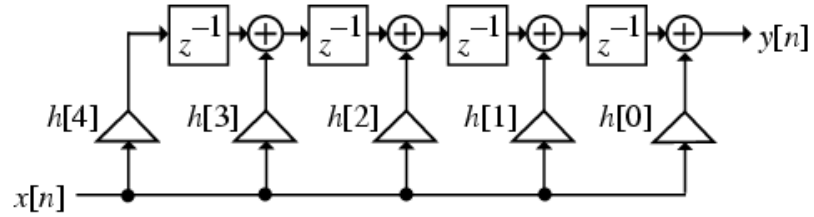

Figure 3: Transpose of the direct form structure in Figure 2.