



Free Books

- Home
- Blogs
- Forums
- Quizzes!
- Courses
- Tutorials
- Books
- Free Books
- Free PDFs
- Code
- Login / Register

Home Blogs Forums Quizzes! Courses Tutorials Books Free Books Free PDFs Code

Home > Free Books > Introduction to Digital Filters >

DIRECT FORM II

The **signal flow graph** for the Direct-Form-II (DF-II) realization of the second-order **IIR filter** section is shown in Fig.9.2.

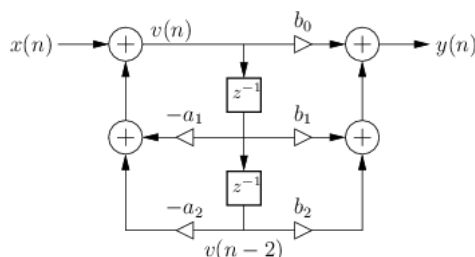


Figure 9.2: Direct-Form-II implementation of a 2nd-order **digital filter**.

The **difference equation** for the second-order DF-II structure can be written as

$$\begin{aligned} v(n) &= x(n) - a_1 v(n-1) - a_2 v(n-2) \\ y(n) &= b_0 v(n) + b_1 v(n-1) + b_2 v(n-2) \end{aligned}$$

which can be interpreted as a **two-pole filter** followed in series by a two-zero **filter**. This contrasts with the DF-I structure of the previous section (diagrammed in Fig.9.1) in which the two-zero FIR section precedes the two-pole recursive section in series. Since **LTI filters** in

Sign in

Sign in

☒ Remember me

[Forgot username or password?](#) | [Create account](#)

You might also like...

COURSE STARTS APRIL 25, 2024!

PYTHON APPLICATIONS FOR DIGITAL DESIGN AND SIGNAL PROCESSING

Dan Boschen

REGISTER NOW!

Upcoming Course - Python Applications for Digital Design and Signal Processing

About this Book



Introduction to Digital Filters

This book is a gentle introduction to digital filter theory including mathematical theory, illustrative examples,

series *commute* (§6.), we may reverse this ordering and implement an all-pole filter followed by an **FIR filter** in series. In other words, the zeros may come first, followed by the poles, without changing the **transfer function**. When this is done, it is easy to see that the delay elements in the two filter sections contain the same numbers (see Fig.5.1). As a result, a single **delay line** can be *shared* between the all-pole and all-zero (FIR) sections. This new combined structure is called "direct form II" [60, p. 153-155]. The second-order case is shown in Fig.9.2. It specifies exactly the same digital filter as shown in Fig.9.1 in the case of infinite-precision numerical computations.

In summary, the DF-II structure has the following properties:

1. It can be regarded as a two-pole filter section followed by a two-zero filter section.
2. It is *canonical with respect to delay*. This happens because delay elements associated with the two-pole and two-zero sections are *shared*.
3. In **fixed-point** arithmetic, overflow can occur at the **delay-line** input (output of the leftmost summer in Fig.9.2), unlike in the DF-I implementation.
4. As with all direct-form filter structures, the **poles and zeros** are sensitive to round-off errors in the coefficients a_i and b_i , especially for high transfer-function orders. Lower sensitivity is obtained using series low-order sections (e.g., second order), or by using ladder or **lattice filter** structures [86].

MORE ABOUT POTENTIAL INTERNAL OVERFLOW OF DF-II

Since the **poles** come first in the DF-II realization of an **IIR filter**, the **signal** entering the state **delay-line** (see Fig.9.2) typically requires a larger **dynamic range** than the output signal $y(n)$.

In other words, it is common for the feedback portion of a DF-II **IIR filter** to provide a large signal *boost* which is then compensated by *attenuation* in the feedforward portion (the zeros). As a result, if the input dynamic range is to remain unrestricted, the two delay elements may need to be implemented with high-order *guard bits* to accommodate an extended dynamic range. If the number of bits in the delay elements is doubled (which still does not guarantee impossibility of internal overflow), the benefit of halving the number of delays relative to the DF-I structure is approximately canceled. In other words, the DF-II structure, which is canonical with respect to delay, may require just as much or more memory as the DF-I structure, even though the DF-I uses twice as many addressable delay elements for the filter state memory.

Next Section:

🔗 [Transposed Direct-Forms](#)

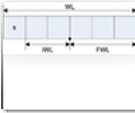
Previous Section:

🔗 [Direct-Form I](#)

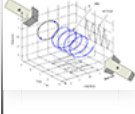
many, massive examples, some audio applications, and useful software start points.

[Read →](#) [Order -](#)


Blogs - Hall of Fame



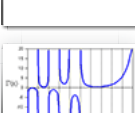
A FIXED-POINT INTRODUCTION BY EXAMPLE
Christopher Felton



HANDLING SPECTRAL INVERSION IN BASEBAND PROCESSING
Eric Jacobsen



UNDERSTANDING THE PHASING METHOD OF SINGLE SIDEBAND MODULATION
Rick Lyons



AN INTERESTING FOURIER TRANSFORM 1/F NOISE
Steve Smith

Free PDF Downloads



📄 **Design IIR Bandpass Filters**



📄 **Introduction to Real-Time Digital Signal Processing**



📄 **The Swiss Army Knife of Digi Networks**

All FREE PDF Downloads 📄

Quick Links

Home
Blogs
Forums
Quizzes!
Courses
Tutorials
Books
Free Books
Free PDFs
Code
comp.dsp

About DSPRelated.com

Advertise
Contact
Privacy Policy
Terms of Service
Cookies Policy

Social Networks



The Related Media Group

