CIFB (Cascade of Integrators with Distributed Feedback)

- Cascade of delaying integrators:
  - Feedback coefficients $a$’s realize the zeros of $L_1$ and thus the NTF and STF poles.
  - Feed-in coefficients $b$’s determine zeros of $L_0$ and thus the STF zeros.
  - State scaling coefficients $c$’s are used for dynamic range scaling.
  - Implements Butterworth NTF.
Combine a non-delaying and a delaying integrator with local feedback around them, to form a stable resonator.

- Local feedback coefficients $g$’s realize the complex zeros in the NTF.
- Implements NTF with complex zeros: $z_i = e^{\pm j \sqrt{g_1}}$

For odd-order, use an integrator in the front to avoid noise coupling due to $g$. 

© Vishal Saxena
CIFB with Resonators

- A resonator can also be formed with two delaying integrators
  - Resonator poles outside the unit circle. \( z_i = e^{1\pm j\sqrt{g_1}} \)
  - Locally unstable but works fine in a stable loop-filter.
- Relaxes settling requirements on the op-amps and implements complex NTF zeros.
CIFF (Cascade of Integrators with Feed-Forward Summation)

Cascade of delaying integrators:
- Feedforward coefficients $a$’s realize the zeros of $L_1$ and thus the NTF and STF poles.
- Feed-in coefficients $b$’s determine zeros of $L_0$ and thus the STF zeros.
- State scaling coefficients $c$’s are used for dynamic range scaling.
- Implements Butterworth NTF.
CRFF (Cascade of Resonators with Feed-Forward Summation)

- Use resonators with feedforward summation.
  - Local feedback coefficients $g_i$’s realize the complex zeros in the NTF.
  - Implements NTF with complex zeros. $z_i = e^{\pm j \sqrt{g_i}}$
- For odd-order, use an integrator in the front to avoid noise coupling due to $g$.
CIFF with Resonators

- Uses resonators formed with two delaying integrators.
  - Resonator poles outside the unit circle. \( z_i = e^{1 \pm j \sqrt{g_1}} \)
Low-Distortion CIFF Topology

\[ b_1 = b_{N+1} = 1 \]

\[ STF(z) = 1 \]
MODULATOR MODEL DETAILS

A delta-sigma modulator with a single quantizer is assumed to consist of quantizer connected to a loop filter as shown in the diagram below.

![Diagram of a delta-sigma modulator](image)

The Loop Filter

The loop filter is described by an ABCD matrix. For single-quantizer systems, the loop filter is a two-input, one-output linear system and ABCD is an \((n+1)\times(n+2)\) matrix, partitioned into \(A\) \((n\times n)\), \(B\) \((n\times 2)\), \(C\) \((1\times n)\) and \(D\) \((1\times 2)\) sub-matrices as shown below:

\[
ABCD = \begin{bmatrix} A & B \\ C & D \end{bmatrix}.
\]  

(1)

The equations for updating the state and computing the output of the loop filter are

\[
x(n + 1) = Ax(n) + B \begin{bmatrix} u(n) \\ v(n) \end{bmatrix} \\
y(n) = Cx(n) + D \begin{bmatrix} u(n) \\ v(n) \end{bmatrix}.
\]  

(2)

This formulation is sufficiently general to encompass all single-quantizer modulators which employ linear loop filters. The toolbox currently supports translation to/from an ABCD description and coefficients for the following topologies:

- CIFB: Cascade-of-integrators, feedback form.
- CIFF: Cascade-of-integrators, feedforward form.
- CRFB: Cascade-of-resonators, feedback form.
- CRFF: Cascade-of-resonators, feedforward form.
- CRFBD: Cascade-of-resonators, feedback form, delaying quantizer.
- CRFFD: Cascade-of-resonators, feedforward form, delaying quantizer.

Multi-input and multi-quantizer systems can also be described with an ABCD matrix and Eq. (2) will still apply. For an \(n_i\)-input, \(n_o\)-output modulator, the dimensions of the sub-matrices are \(A\): \(n\times n\), \(B\): \(n\times(n_i+n_o)\), \(C\): \(n_o\times n\) and \(D\): \(n_o\times(n_i+n_o)\).
CIFB Structure

For odd order, first integrator is not paired up as a resonator.
CIFF Structure

Even Order

Odd Order
CRFB Structure

Even Order

Odd Order

Note that with NTFs designed using `synthesizeNTF`, omission of the $b_2$ coefficients in the CRFB structure will yield a maximally-flat STF.
CRFF Structure

Even Order

Odd Order

DAC

u(n)

v(n)

y(n)

x_2(n+1)

x_1(n)

1/z -1

-1/z -1

1/z -1

DAC

DAC

u(n)

v(n)

y(n)

x_3(n)

x_2(n+1)

x_1(n)

1/z -1

-1/z -1

1/z -1

DAC
CRFBD Structure

\[ u(n) \]

Even Order

\[
\begin{align*}
&b_1 \quad x_1(n) \quad c_1 \\
&-a_1 \quad -g_1 \quad z^{-1} \quad x_1(n) \quad + \quad z^{-1} \quad x_2(n+1) \\
&b_2 \quad c_2 \\
\end{align*}
\]

Odd Order

\[
\begin{align*}
&b_1 \quad x_1(n+1) \quad c_1 \\
&-a_1 \quad -g_1 \quad z^{-1} \quad x_2(n+1) \\
&b_2 \quad b_3 \quad c_3 \\
\end{align*}
\]

\[ v(n) \]

\[ b_{n+1} \] is not shown since it would have to be added to the quantizer input without delay, which is presumed to not be allowed in this structure. Note that this makes it impossible to have a unity STF.
CRFFD Structure

Even Order

Odd Order

DAC