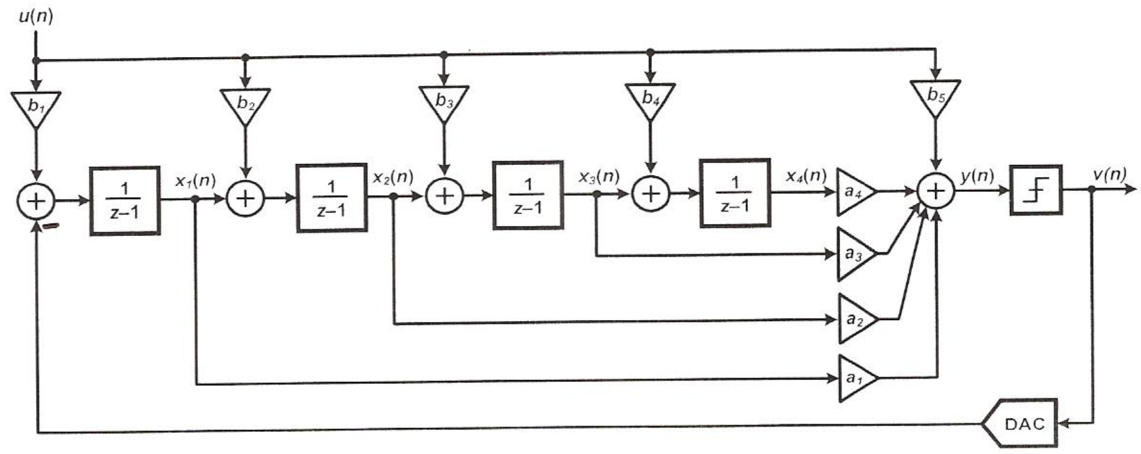
Feedforward Loop-filter Topologies

CIFF



Cascade of Integrators with feed forward summation (CIFF)

↳ Use feedforward paths rather than feedback ones to create the zeros of the NTF.

By inspection:

If $I(z) = \frac{1}{z-1} = \frac{z^{-1}}{1-z^{-1}}$ ← delaying integrator / accumulator.

$L_1(z) = -a_1 I(z) - a_2 I^2(z) - \cdots - a_N I^N(z)$

$L_0(z) = b_1 (a_1 I + a_2 I^2 + \cdots + a_N I^N) + b_2 (a_2 I + \cdots + a_N I^{N-1}) + b_3 (a_3 I + \cdots + a_N I^{N-2}) + \cdots + b_{N+1}$

$\Rightarrow L_1(z) = -\frac{a_1 + a_2(z-1) + \cdots + a_N (z-1)^{N-1}}{(z-1)^N}$ ← L_1 same as in CIFB ⇒ Same NTF

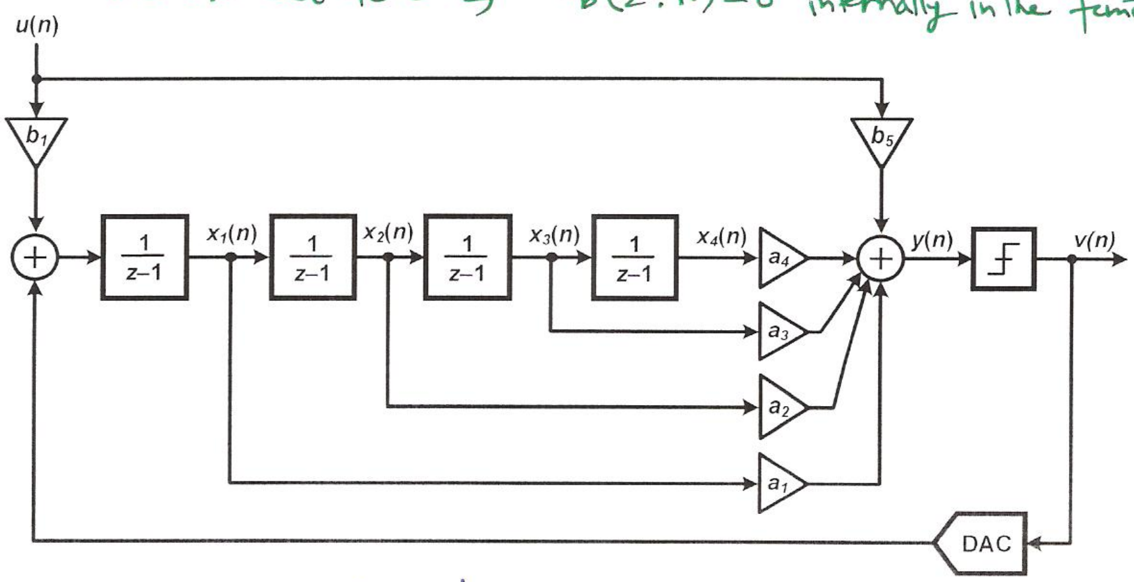
$L_0(z) = \frac{\gamma_1 + \gamma_2 (z-1) + \cdots + \gamma_N (z-1)^{N-1} + \gamma_{N+1} (z-1)^N}{(z-1)^N}$,

$\gamma_{N+1} = b_{N+1}$
$\gamma_N = b_1 a_1 + b_2 a_2 + b_3 a_3 + \cdots$
$\gamma_{N-1} = b_1 a_2 + b_2 a_3 + b_3 a_4 + \cdots$
$\vdots$
$\gamma_1 = b_1 a_N$

} Complicated ⇐ This generic case is not used in the Toolbox.

• Now, consider the case when $b_2 = b_3 = \cdots = b_N = 0$ and $b_1 = b_{N+1} = 1$ ②
→ input coupling, and full input feedforward.

Note:
• Toolbox realizeNTF() function only uses b_1 and b_{N+1}, rest b's are set to 0 ⇒ $b(2:N) = 0$ internally in the function.



Now, the $L_0(z)$ simplifies to

$$L_0(z) = (a_1 I + a_2 I^2 + \cdots + a_N I^N) + 1$$
$$= 1 - (-a_1 I - a_2 I^2 - \cdots - a_N I^N)$$
$$= 1 - L_1(z) \longrightarrow ①$$

$$\Rightarrow \begin{cases} L_0(z) = 1 - L_1(z) & \text{when } b_{N+1} = 1 \\ L_0(z) = -L_1(z) & \text{when } b_{N+1} = 0. \end{cases}$$

From ①, we have when $b_{N+1} = 1$ (Full input feedforward case)

$$\boxed{STF(z) = \frac{L_0(z)}{1 - L_1(z)} = 1}, \quad \text{Also } NTF(z) = \frac{1}{1 - L_1(z)}$$

when $b_{N+1} = 0$, (only input is coupled)

$$\boxed{STF(z) = 1 - NTF(z) = \frac{-L_1(z)}{1 - L_1(z)}}, \quad NTF(z) = \frac{1}{1 - L_1(z)}$$

- when $b_1 = b_{N+1} = 1$

  STF = 1

  $$\Rightarrow \quad U(z) - V(z) = U(z) - \underbrace{[U(z) + NTF(z)E(z)]}_{V(z)}$$

  $$= NTF(z) \cdot E(z)$$

  ⇒ Loop-filter doesn't process the input signal $u[n]$ at all.

  ↳ low distortion case

  ↳ Same results apply to relaxed integrator implementation as seen for the CIFB low-distortion case.

- Note that implementing the low-distortion condition is easier with CIFF (only 2 branches) as opposed to (N+1) input coupling branches required for the CIFB case.

- when $b_{N+1} = 0$ ⇒ only input coupling is used then we saw that

  $STF(z) = 1 - NTF(z)$

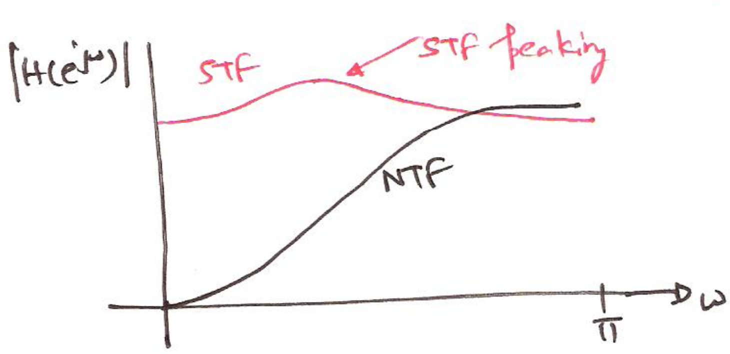
  ⇒ results in high-frequency peaking in the STF.

  ↳ add pre-filter to the inputs in this range to avoid overloading in the modulator.

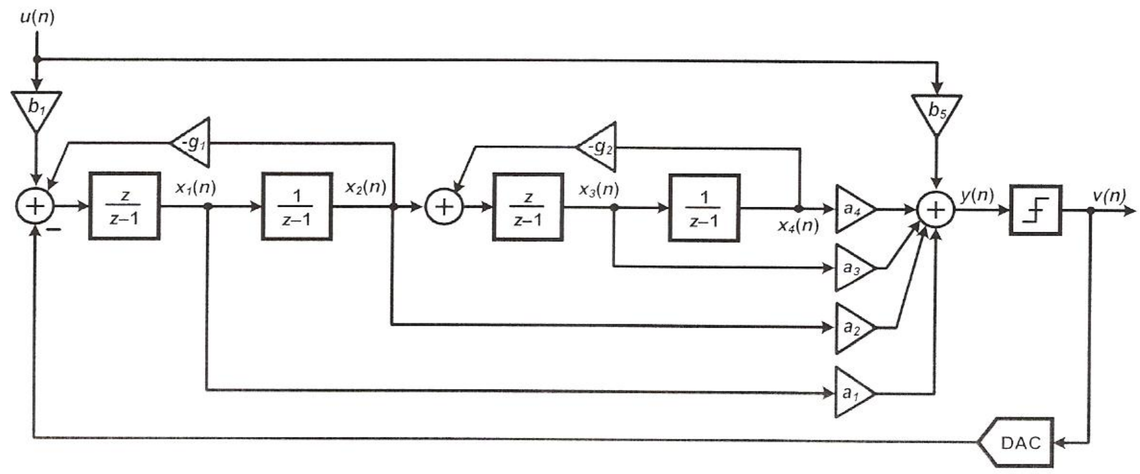
  or modify the NTF/STF

  ↳ can use other b_i's to attenuate the noise peaking.

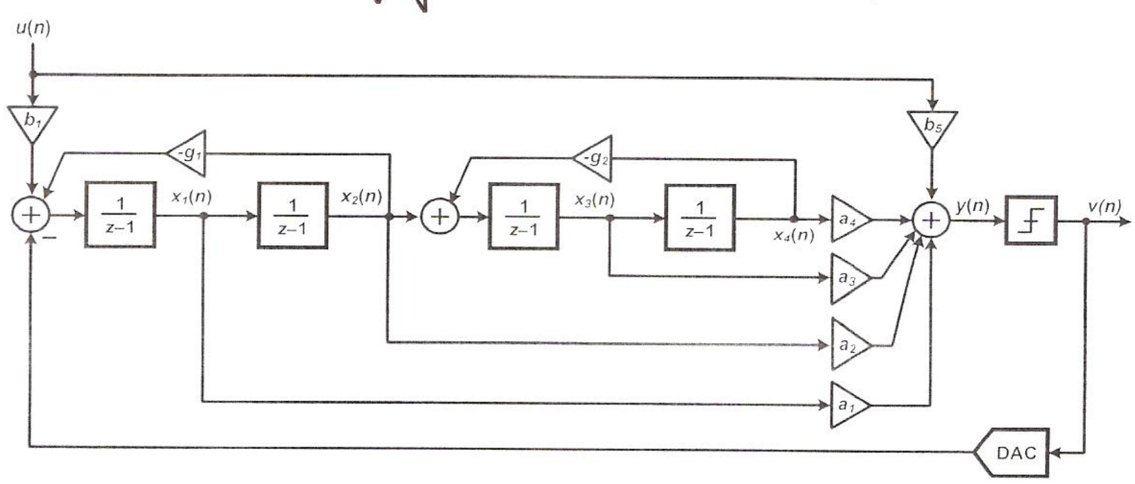
  ↳ figure this out for CIFF/CRFB topologies.



- Add resonator feedbacks to create complex NTF zeros
  ↳ CRFF (Cascade of Resonators with feedforward Summation).



- Can also use double-delaying resonators to modify the CIFF topology.



- Same analysis for resonators as seen for the CRFB/CIFB cases.
- Here, multiple input feed-in branches are also possible using b(2:N)
  ↳ modify toolbox function?

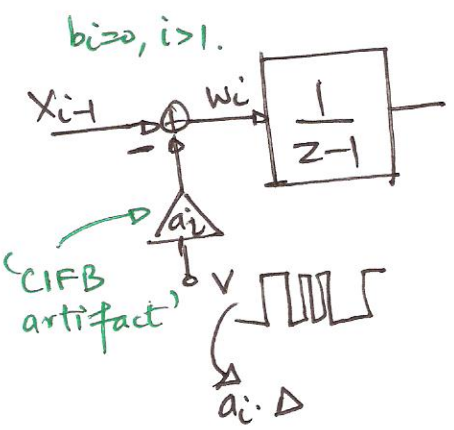
Comparison between FB and FF topologies:

#	Feed Forward	Feedback
(1)	FF has relaxed dynamic range requirements.	Integrator outputs contain significant amount of input signal as well as filtered quantization noise (Typically)
(2)	If full-FF is not used ($b_{N+1}=0$) STF peaking occurs. ↳ a problem in CT-DSMs	STF has good attenuation beyond the signal band ↳ Better AAF in CT-DSMs.
(3)	only one DAC required	Requires many feedback DACs.
(4)	Needs a summation block	No extra summer required.
(5)	Timing can be tricky ↳ need to quantize u and feed it back in zero time.	Not an issue
(6)	First integrator is fastest	Last integrator is fastest (has large signal content)
(7)	First opamp is power hungry (Noise reasons)	First opamp is power hungry (noise reasons)
	(6) & (7) → Best optimization for opamp power consumption	(6) & (7) ⇒ first and last opamps are power hungry → more power is burnt.
(8)	Small capacitor area	Large capacitor area to accomodate the large signal swings ↳ DRS results in large C's. ↳ more power burnt in opamps ↳ Larger layout size.

If STF/AAF is not an issue, CIFF (CRFF) architecture is generally preferred.

- Intuition for larger integrator swings in CIFB/CRFB topology.

Consider a DC input to the CIFB modulator with $b_i = 0$ for $i > 1$.
and 1-bit quantizer feedback.



- Each integrator has ∞ gain at DC
  - ⇒ Sum of all inputs $= 0$ to prevent any dc component from appearing at the integrator input → else loop-filter will saturate
  - ⇒ one input is the 1-bit feedback path
    another is the output of the previous integrator i.e. (X_{i-1}).

⇒ X_{i-1} must have a DC component to counteract the DC component of the 1-bit feedback.

⇒ Each integrator output contains a combination of filtered quantization noise and a low frequency component equal to the input signal.

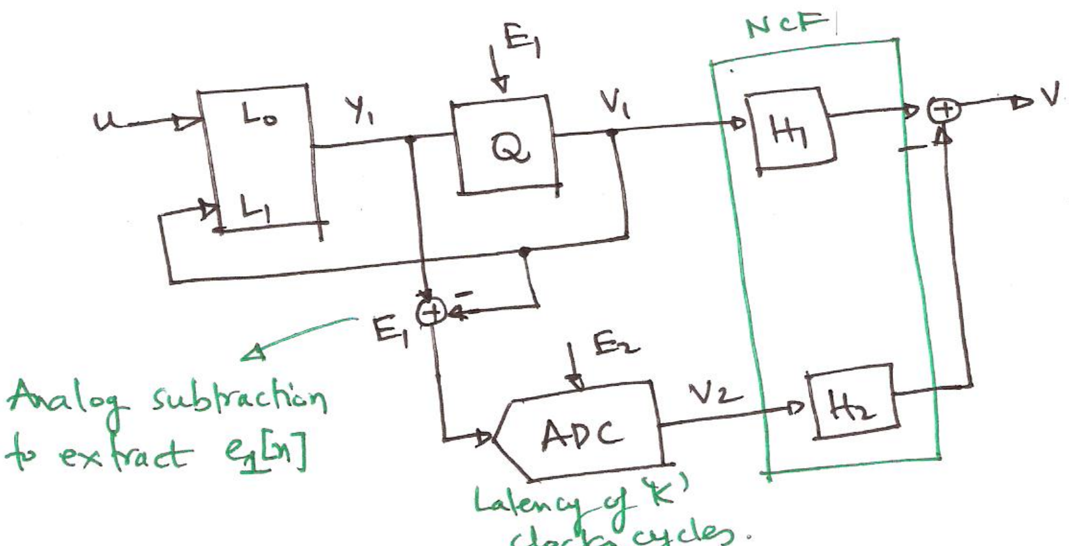
- From simulations, the signal component in X_i is significantly larger than the noise component
  - ↳ larger integrator swings ($\max(X_i) > \max(X_{i-1})$)
    (verify yourself)
  - ↳ need larger integrator feedback caps to limit the swings, as a result of dynamic range scaling (DRS)
    - ↳ requires large 'c_i' values.

⇒ FB circuits tend to be larger and more power hungry than the FF ones.

Multi-stage Modulators

- For low-OSR values, it's no longer possible to obtain high SQNR values in a single (quantizer) loop modulator by raising the order of the loop-filter.
  - ↳ MSA ↓ as order ↑
  - ↳ can increase quantizer resolution
    - ↳ Flash ADC design becomes complicated
    - ↳ DAC → linearity issues (complex DEM for large nLev)
    - ↳ complexity ↑ as 2^N
      - max 4-5 bits quantizer is employed.
- Use digital cancellation of noise, rather than filtering, by using a multistage (cascade) structure of the modulator.

(1) L-0 Cascade (Leslie-Singh Structure).



- L^{th} order ΔΣ Modulator in the first stage, followed by a static (zero-order) ADC as the second-stage → eg. a pipelined ADC of 10-bits resolution.
- Combine the outputs of the two stages V_1 & V_2 to obtain the overall output V.

↳ $e_1[n]$ is extracted in analog form by
$e_1 = V_1 - y_1$

⇒ e_1 is then converted to digital form by a multibit (say 10 bits) ADC forming the 2^{nd}-stage.

↳ introduces a second quantization noise $e_2[n]$.

- $|e_2[n]| << |e_1[n]|$
- 2^{nd}-stage can have arbitrary latency (no feedback loop around it)

↳ realized using a low complexity pipelined ADC.

- V_1 and V_2 are filtered by digital stages H_1 and H_2 respectively and then "added" together.

↳ $H_1(z) = z^{-k}$ simply ⇒ matches the latency of the 2^{nd}-stage.

↳ $H_2(z)$ ⇒ digital equivalent of the first-stage NTF.

$$\Rightarrow V(z) = H_1(z) \cdot V_1(z) - H_2(z) \cdot V_2(z)$$

$$= \underbrace{z^{-k}}_{H_1}\left[STF_1 \cdot U + NTF_1 \cdot \cancel{E_1}\right] - \underbrace{NTF_1}_{H_2} \cdot z^{-k} \cdot \left[\underbrace{\cancel{E_1}}_{U_2 = e_1} + E_2\right].$$

$$\simeq z^{-k}\left[STF_1 \cdot U - NTF_1 \cdot E_2\right]$$

if E_1 is perfectly cancelled. i.e. $H_2 = NTF_1$

- Compare $V(z)$ with $V_1(z)$:

Except for the delay of z^{-k},

- $E_1(z)$ is replaced by $-E_2(z)$
- PSD level of $E_2(z)$ is much smaller than that of $E_1(z)$

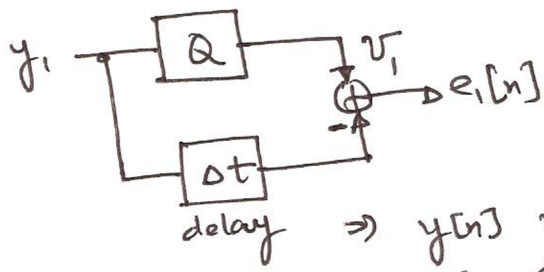
↳ its cheaper to construct a pipelined ADC than a multibit-loop quantizer.

↳ Enhances SQNR by as much as 25-30 dB.

Subtraction Implementation:

- To obtain $e_1[n]$ by subtraction, the operation of the quantizer must be delay-free, which may not be practical.

$\because$ $e_1[n] = \underbrace{v_1[n] - y_1[n]}$ not possible with finite quantizer delay



$\Rightarrow$ $y[n]$ must be delayed before subtraction can be carried out
↳ done using switched-cap techniques.

- To avoid the subtraction altogether, the input signal of the second-stage can be chosen as $y_1[n]$, instead of $e_1[n]$.

Then, we have

$$Y_1(z) = V_1(z) - E_1(z) = STF_1 \cdot U + (NTF_1 - 1) E_1$$

keeping $H_1(z) = z^{-k}$, but choosing H_2 as

$$H_2(z) = \frac{NTF_1(z)}{NTF_1(z) - 1}$$, the overall output now becomes

$$V(z) = z^{-k}[STF_1 \cdot U + NTF_1 \cdot E_1] - \frac{NTF_1}{NTF_1 - 1} z^{-k} \cdot \{STF_1 \cdot U + (NTF_1 - 1)E_1 + E_2\}$$

- Assuming ideal cancellation of terms, we get

$$V(z) = \frac{z^{-k} STF_1(z)}{1 - NTF_1(z)} \cdot U(z) + \frac{z^{-k} NTF_1(z)}{1 - NTF_1(z)} \cdot E_2(z)$$

$\because$ In the signal band $|NTF| \ll 1$ then

$$\frac{NTF_1}{1 - NTF_1} \simeq NTF_1 \text{ in the signal band}$$

↳ can get SQNR close to the ~~value~~ one obtained earlier.

- Disadvantage:

  $y_1(n)$ contains $u(n)$ as well as $e_1(n)$

  → second-stage must be able to handle a larger input signal.

  → Also the second-stage must have lower distortion in processing the $u(n)$ input.

  → Not very attractive overall.

- Consider one of the low distortion structures

  - Eg. CIFB with $b_i = a_i$ and $b_{N+1} = 1$,
    or CIFF with $b_1 = b_{N+1} = 1$ and $b(2:N) = 0$

  ⇒ both cases have STF=1.

  → output of the last integrator in the low-dist CIFB case is:

$$X_N(z) = Y(z) - b_N U(z)$$

$$= STF(z)\cdot U(z) - (1 - NTF(z))E(z) - b_N U(z)$$

$$\Rightarrow \boxed{X_N(z) = -(1 - NTF(z))\cdot E(z)}$$

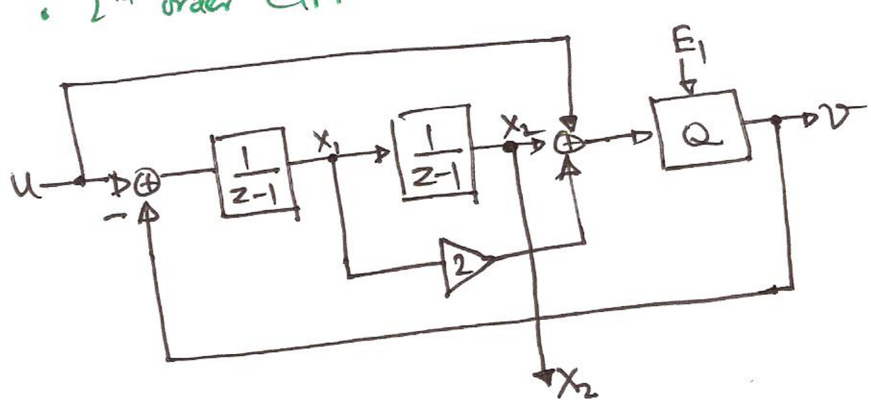
This signal $X_N(z)$ can be used as the input for the second-stage of the L-O cascade.

→ doesn't contain any $u(n)$

→ second-stage need not very linear

Example topology:

- 2nd order CIFF



$NTF = (1 - z^{-1})^2$

$STF = 1$

$X_2 = -z^{-2} E_1(z)$

→ feed directly to the second stage input.

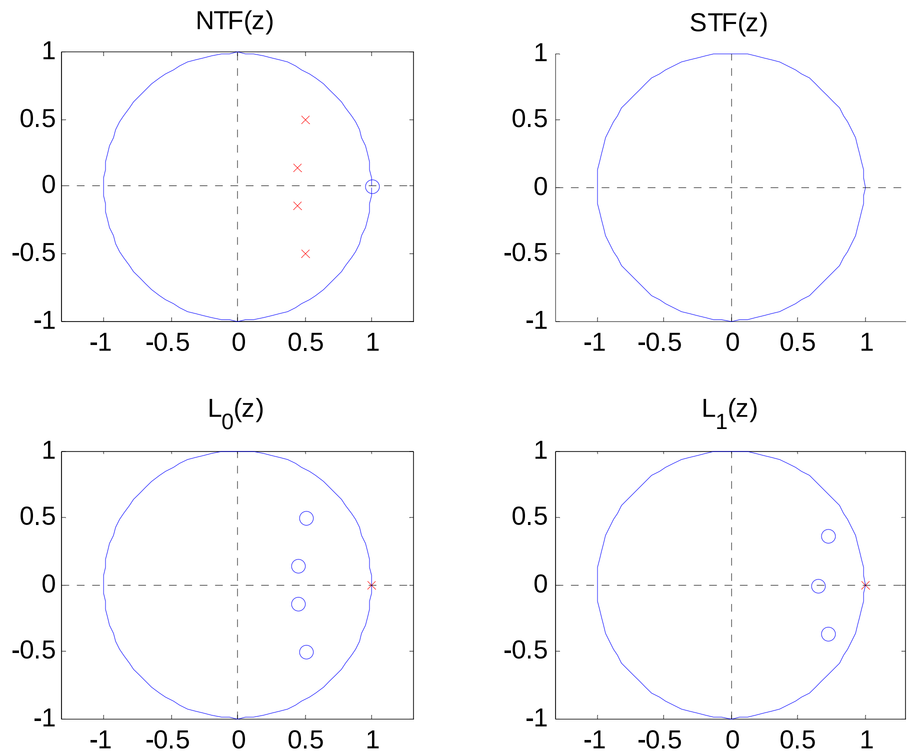
ECE 697 Delta-Sigma Converters Design

Lecture#16 Slides

Vishal Saxena
(vishalsaxena@u.boisetstate.edu)

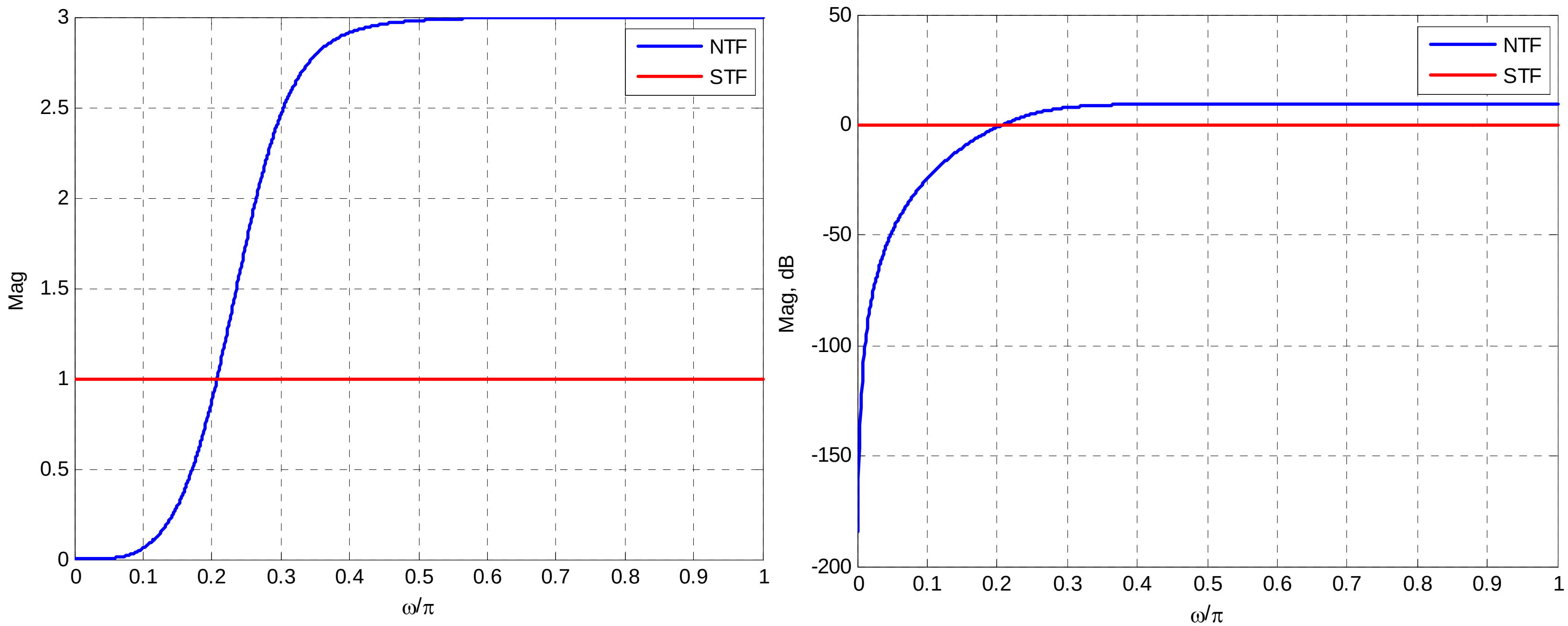
CIFF Example 1

- CIFF, order = 4
- All NTF zeros at z=1, i.e. opt =0.
- OBG = 3, OSR = 16, nLev = 15.
- Low-distortion topology
  - b(1) =b(5)= 1
  - b(2:4)=0
- **a** = [2.1 1.9 0.86 0.16]
- **b** = [1 0 0 0 1]
- **c** = [1 1 1 1]
- **g** = [0 0]



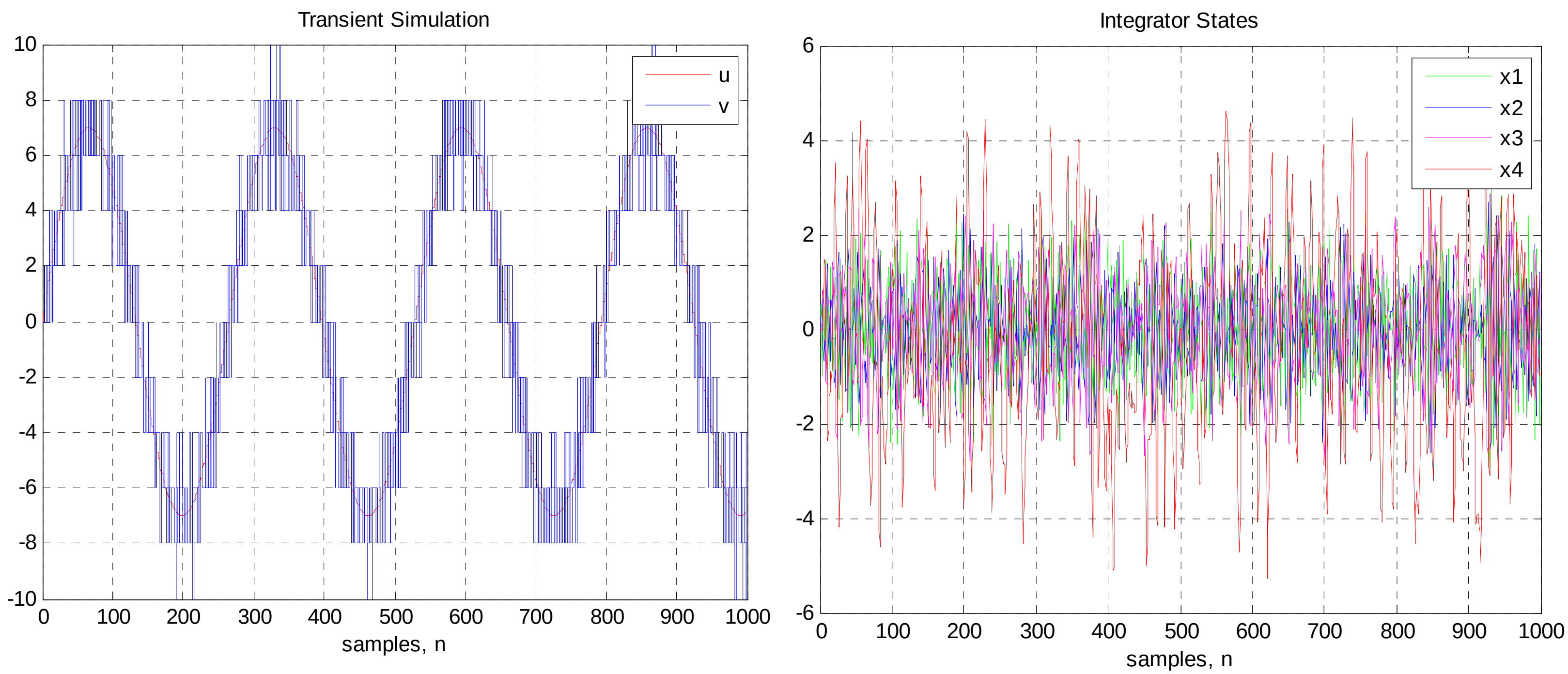
File: CIFB_4th_Order_1.m

CIFF Example 1 contd. : NTF and STF



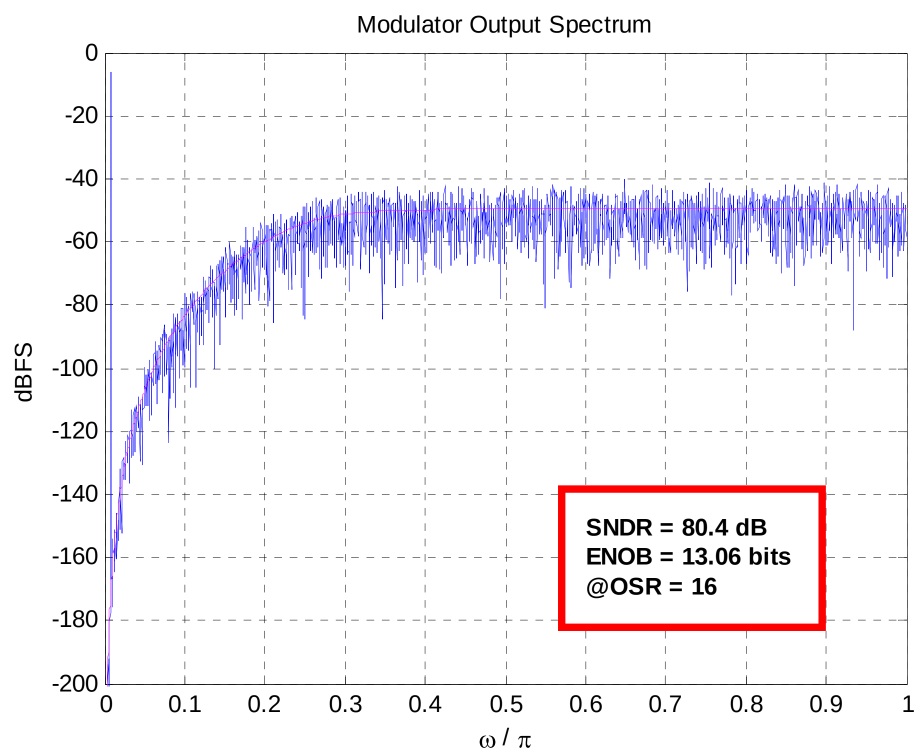
File: CIFF_4th_Order_1.m

CIFF Example 1 contd. : Loop-Filter States



File: CIFF_4th_Order_1.m

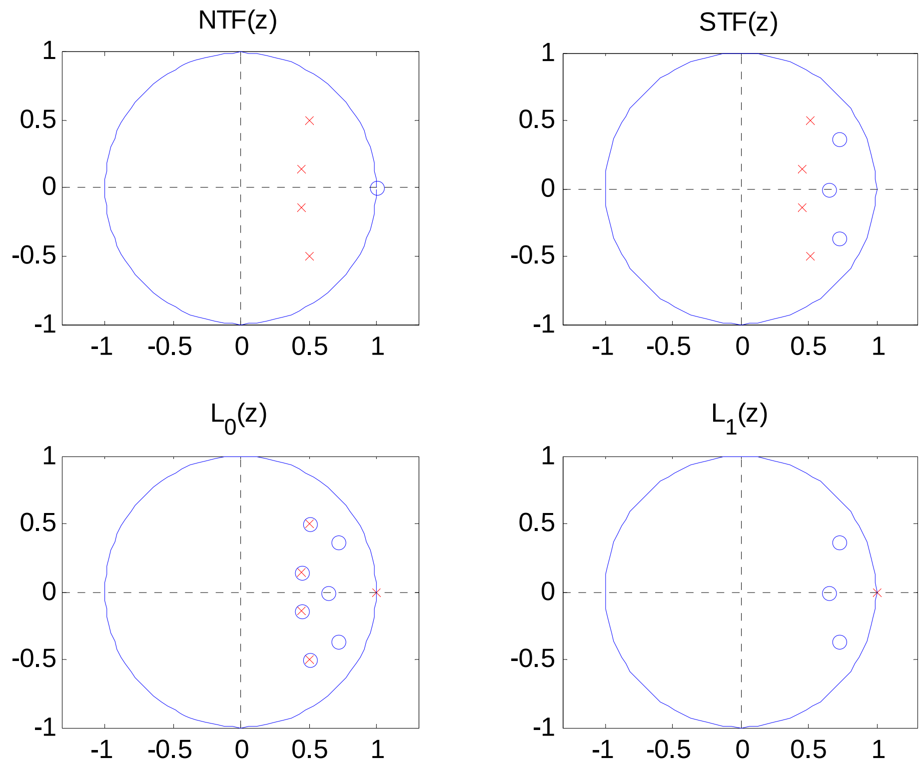
CIFF Example 1 contd. : Simulated Spectrum



File: CIFF_4th_Order_1.m

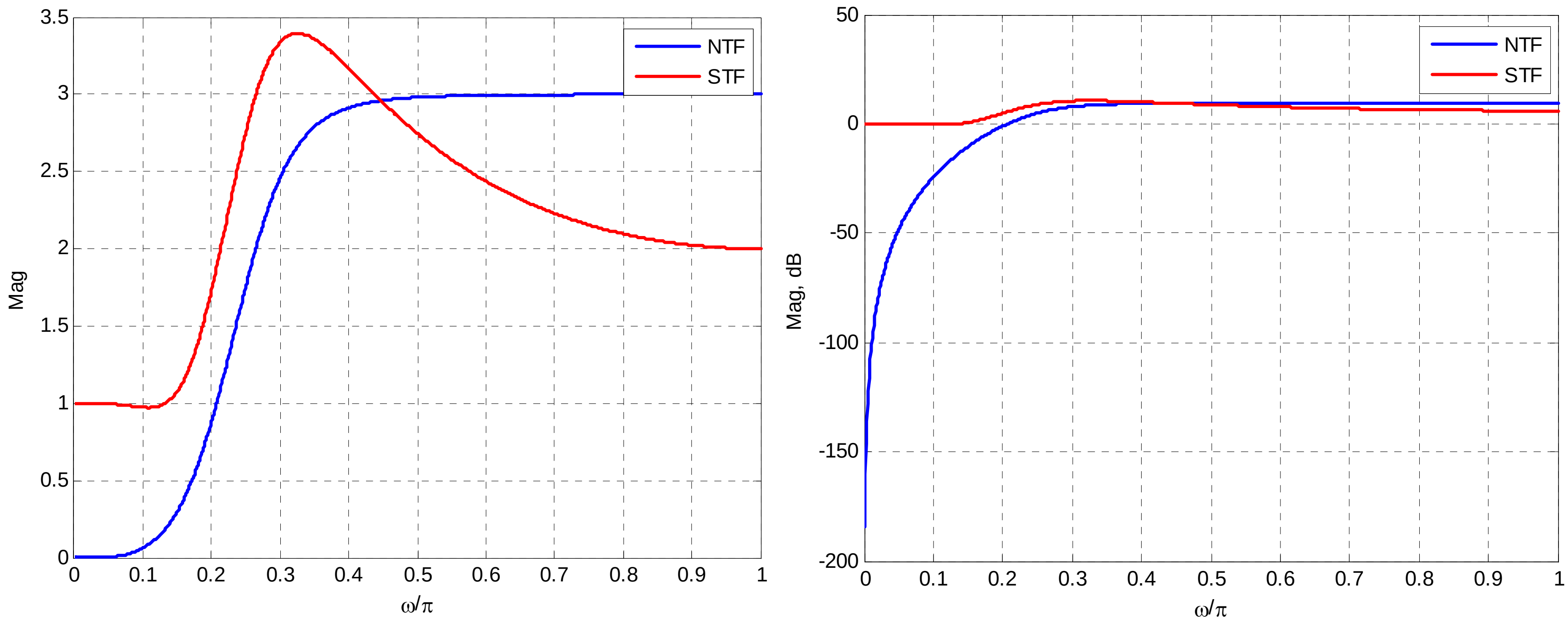
CIFF Example 2

- CIFF, order = 4
- All NTF zeros at z=1, i.e. opt =0.
- OBG = 3, OSR = 16, nLev = 15.
- Only single input feed-in used
  - b(2:end)=0
- **a** = [2.1 1.9 0.86 0.16]
- **b** = [1 0 0 0 0]
- **c** = [1 1 1 1]
- **g** = [0 0]



File: CIFB_4th_Order_2.m

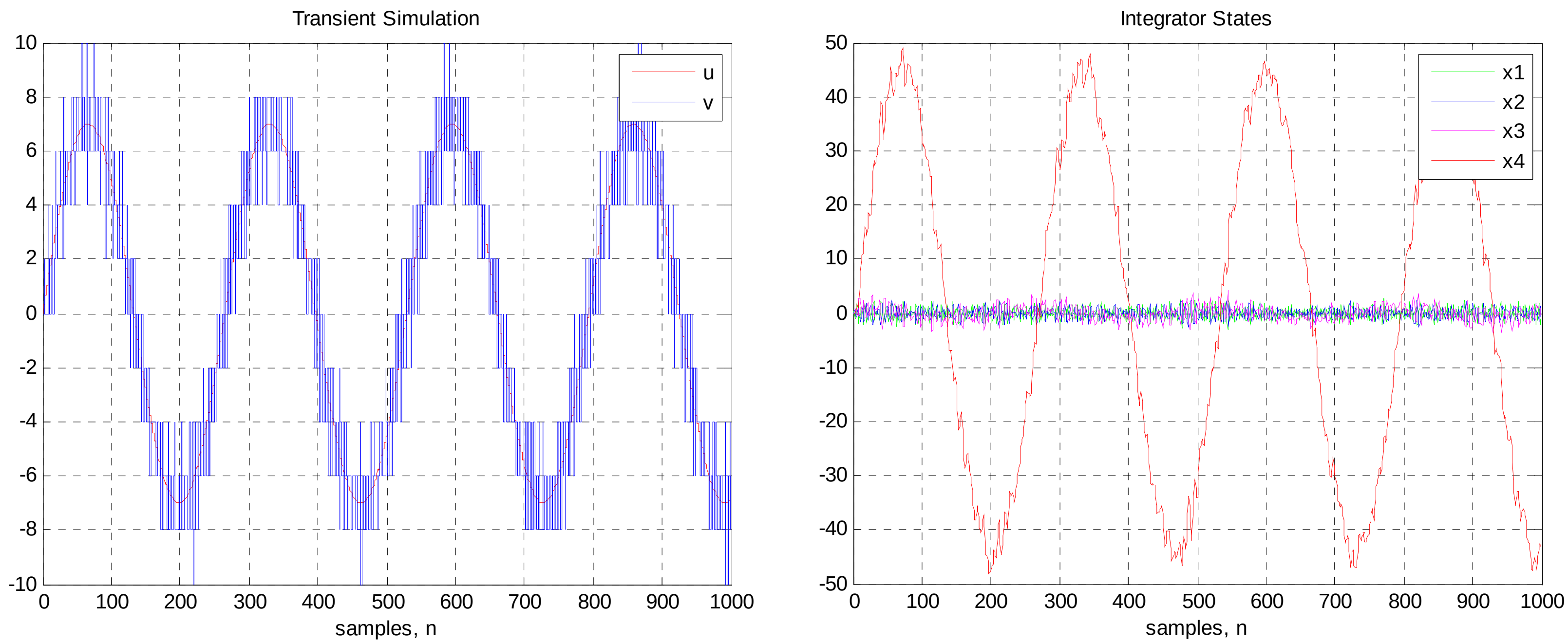
CIFF Example 2 contd. : NTF and STF



- Notice the significant STF peaking !

File: CIFF_4th_Order_2.m

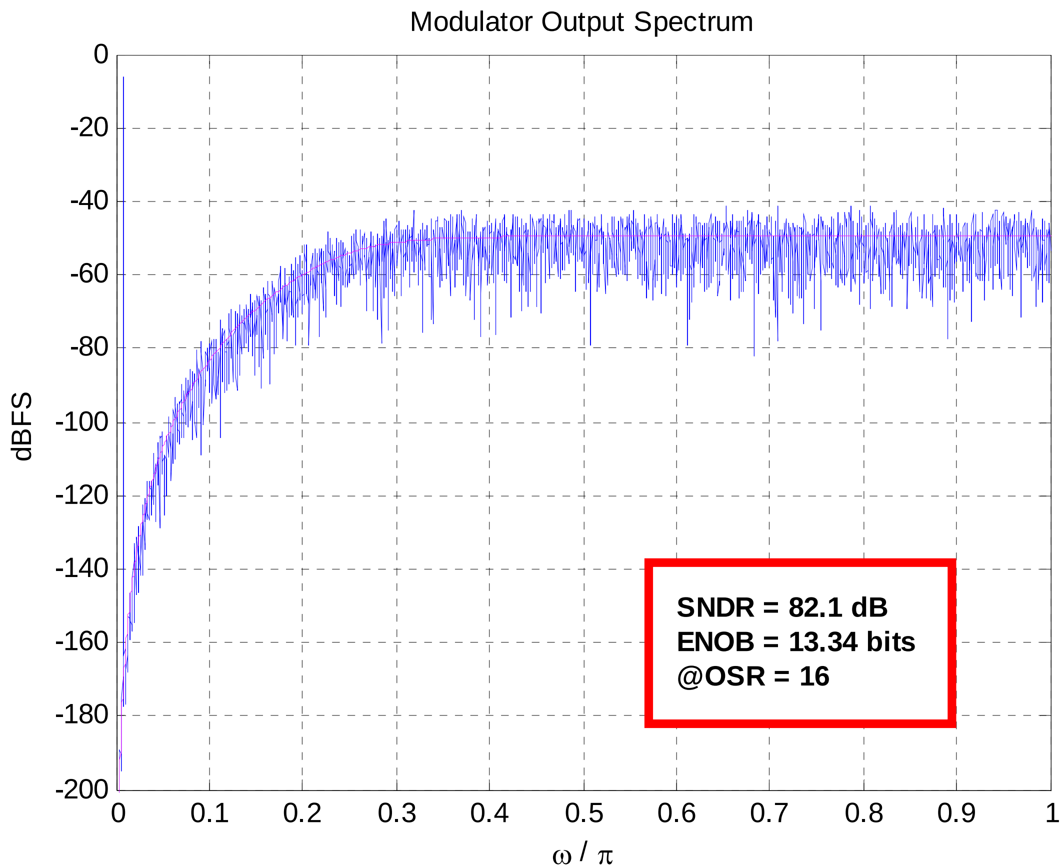
CIFF Example 2 contd. : Loop-Filter States



- Last integrator output has significant signal content
  - ✓ Use dynamic range scaling.
  - ✓ Last integrator will burn more power in this case.

File: CIFF_4th_Order_2.m

CIFF Example 2 contd. : Simulated Spectrum



File: CIFF_4th_Order_2.m

Other Examples of Feed-forward Topologies

- Low-distortion CRFF topology
  - CRFF_4th_Order_1.m
- CRFF with single feed-in
  - CRFF_4th_Order_2.m
- Low-distortion CIFF topology with optimized NTF zeros
  - CIFF_Opt_4th_Order_1.m
- CIFF with single feed-in and optimized NTF zeros
  - CIFF_Opt_4th_Order_2.m
- STF peaking in FF topologies with single feed-in is an issue
  - CT FF DSM will have STF peaking as full-feedforward branch can't be used.
  - The feed-in coefficients b's can be strategically used to realize CIFF/CRFB topology with better out-of-band STF attenuation.

L-0 Cascade Simulation

- TBD

File: TBD.m