ECE1371 Advanced Analog Circuits
Lecture 10

ADVANCED $\Delta\Sigma$

Richard Schreier
richard.schreier@analog.com

Trevor Caldwell
trevor.caldwell@utoronto.ca

Course Goals

- **Deepen understanding of CMOS analog circuit design through a top-down study of a modern analog system— a delta-sigma ADC**
- **Develop circuit insight through brief peeks at some nifty little circuits**

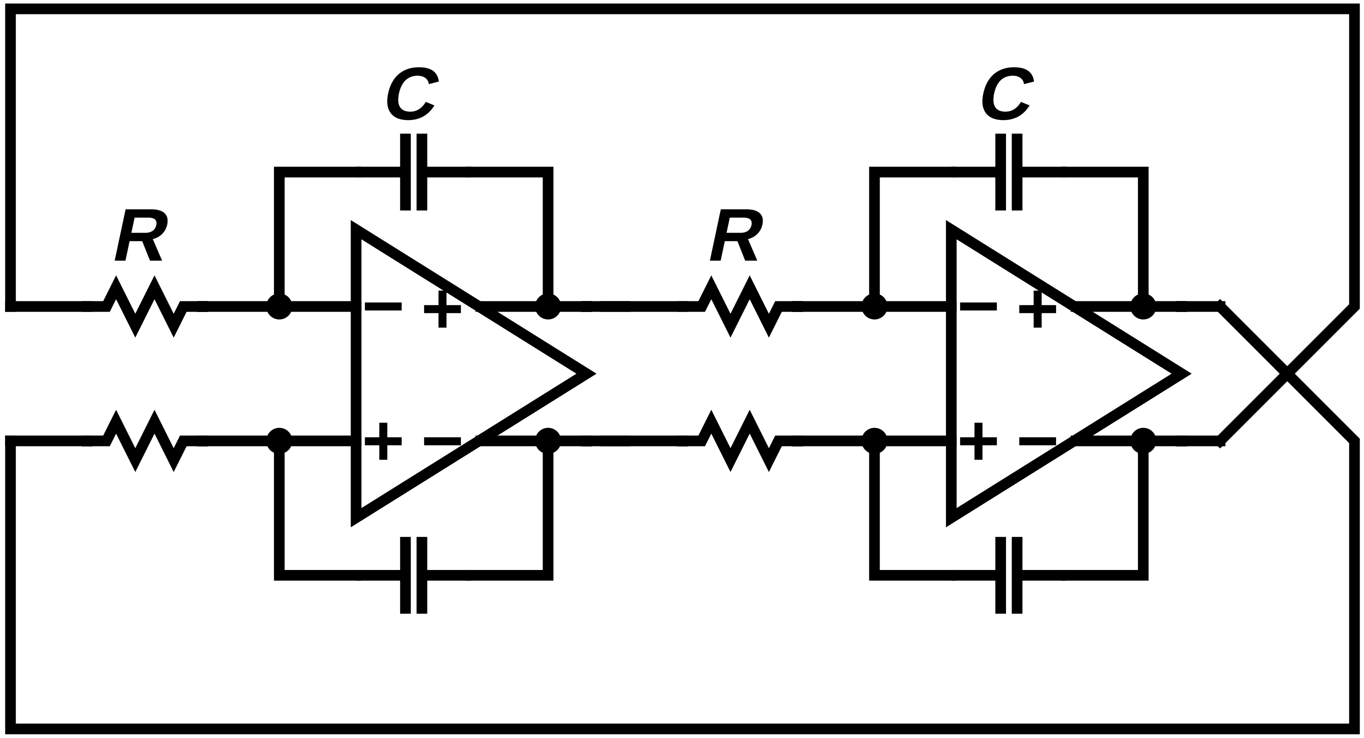
  The circuit world is filled with many little gems that every competent designer ought to know.

NLCOTD: High-Q Resonator

- Want $Q \gg \sqrt{3}\dfrac{f_0}{BW}$ for small SQNR degradation
- In a TV tuner ADC f_0 = 44 MHz and BW = 8.5 MHz, so we needed Q $\gg$ 9
  - In actuality the requirement was Q > 20.
- How can Q be kept high despite finite amplifier gain and bandwidth?

Date	Lecture (M 13:00-15:00)			Ref	Homework
2015-01-05	RS	1	MOD1 & MOD2	ST 2, 3, A	1: Matlab MOD1&2
2015-01-12	RS	2	MOD*N* + ΔΣ Toolbox	ST 4, B	2: ΔΣ Toolbox
2015-01-19	RS	3	Example Design: Part 1	ST 9.1, CCJM 14	3: Sw.-level MOD2
2015-01-26	RS	4	Example Design: Part 2	CCJM 18	
2015-02-02	TC	5	SC Circuits	R 12, CCJM 14	4: SC circuit
2015-02-09	TC	6	Amplifier Design		
2015-02-16	Reading Week– No Lecture				
2015-02-23	TC	7	Amplifier Design		5: SC Int w/ Amp
2015-03-02	RS	8	Comparator & Flash ADC	CCJM 10	Project
2015-03-09	TC	9	Noise in SC Circuits	ST C	
2015-03-16	RS	10	Advanced ΔΣ	ST 6.6, 9.4	
2015-03-23	TC	11	Matching & MM-Shaping	ST 6.3-6.5, +	
2015-03-30	TC	12	Pipeline and SAR ADCs	CCJM 15, 17	
2015-04-06	Exam			Proj. Report Due Friday April 10	
2015-04-13	Project Presentation				

Active-RC Resonator Structure



$$f_0 = \frac{1}{2\pi RC}$$

- **Frequency-tuning: adjust C until the desired resonant frequency is achieved**

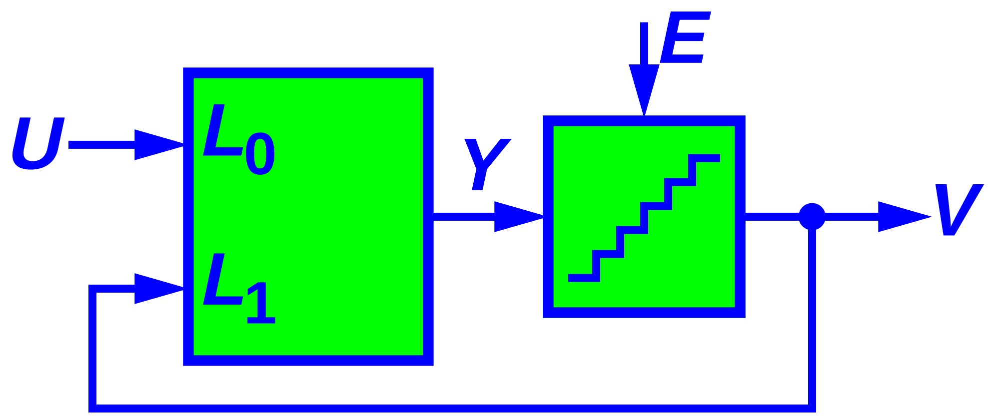
  No Q-tuning.

– **Amplifier drives both R and C $\Rightarrow$ Q trouble?**

Highlights
(i.e. What you will learn today)

1. **Feedback vs. Feedforward topology**
2. **State-space (ABCD) representation of the loop filter in the ΔΣ Toolbox**
3. **MASH Modulators**
4. **Continuous-Time Modulators**
5. **Bandpass and Quadrature Bandpass ΔΣ**

Review: Generic Single-Loop ΔΣ ADC



$$Y = L_0 U + L_1 V$$
$$V = Y + E$$
$$\Rightarrow \boxed{V = STF \cdot U + NTF \cdot E}, \text{ where}$$

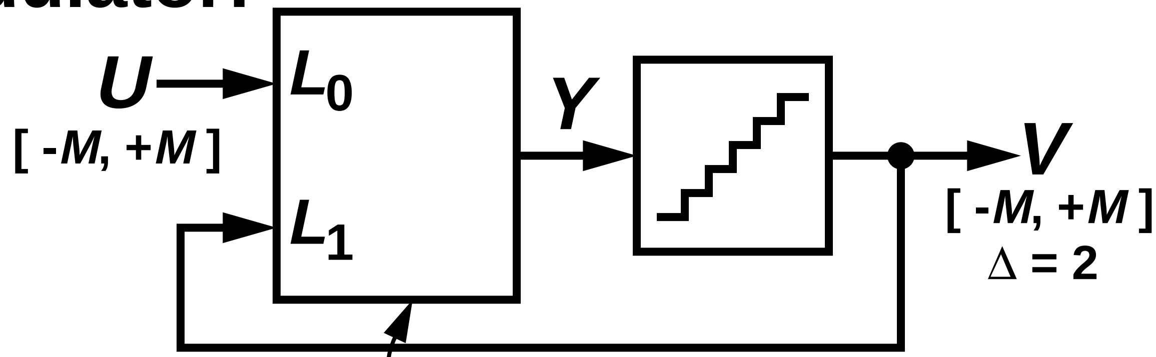
$$NTF = \frac{1}{1 - L_1} \quad \& \quad STF = L_0 \cdot NTF$$

Inverse Relations:

$L_1 = 1 - 1/NTF$, $L_0 = STF / NTF$

Review: ΔΣ Toolbox Model

Modulator:

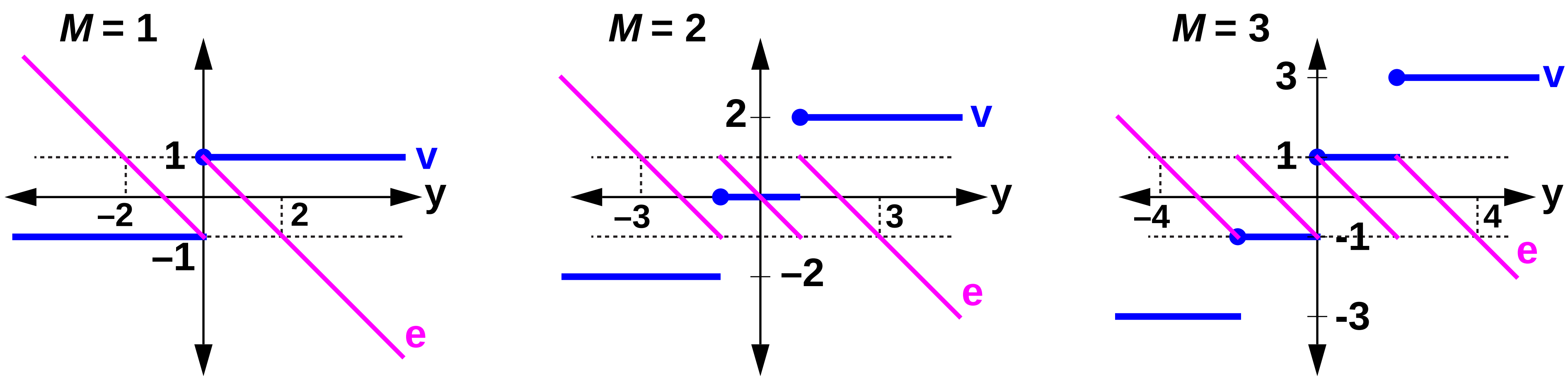


Loop filter can be specified by NTF or
by ABCD, a state-space representation

$$NTF = \frac{1}{1 - L_1}$$

$$STF = \frac{L_0}{1 - L_1}$$

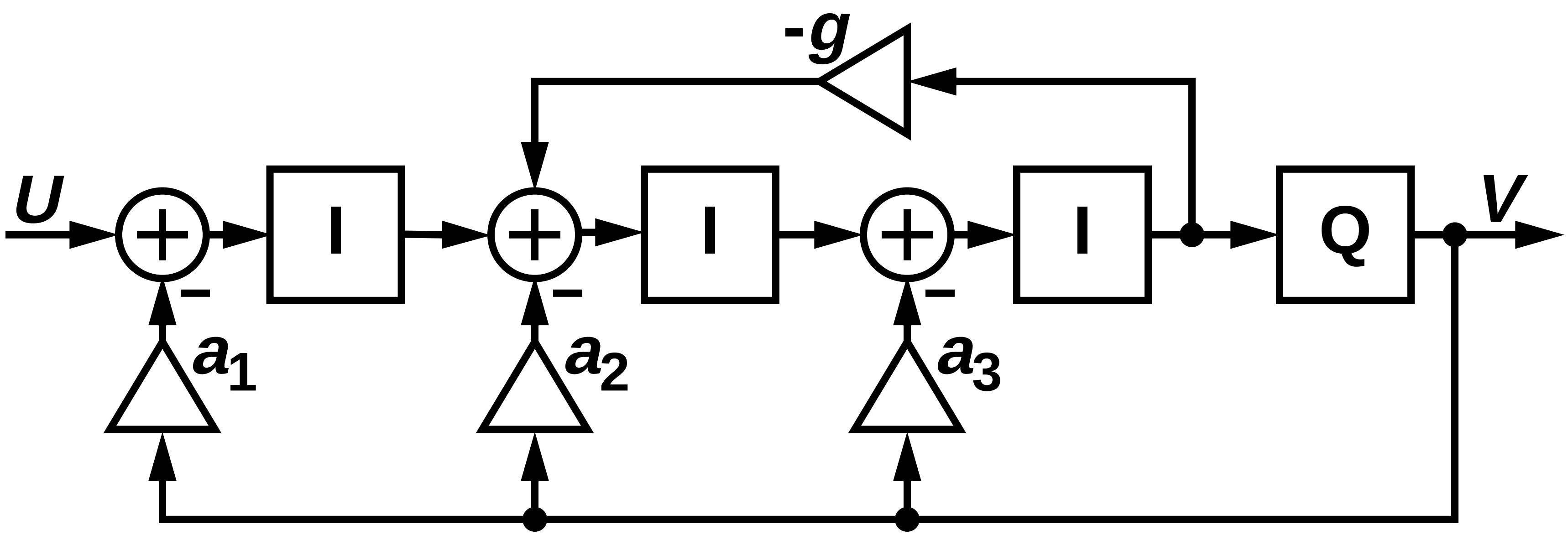
Quantizer:



Mid-tread quantizer;
v: even integers [–*M*,+*M*]

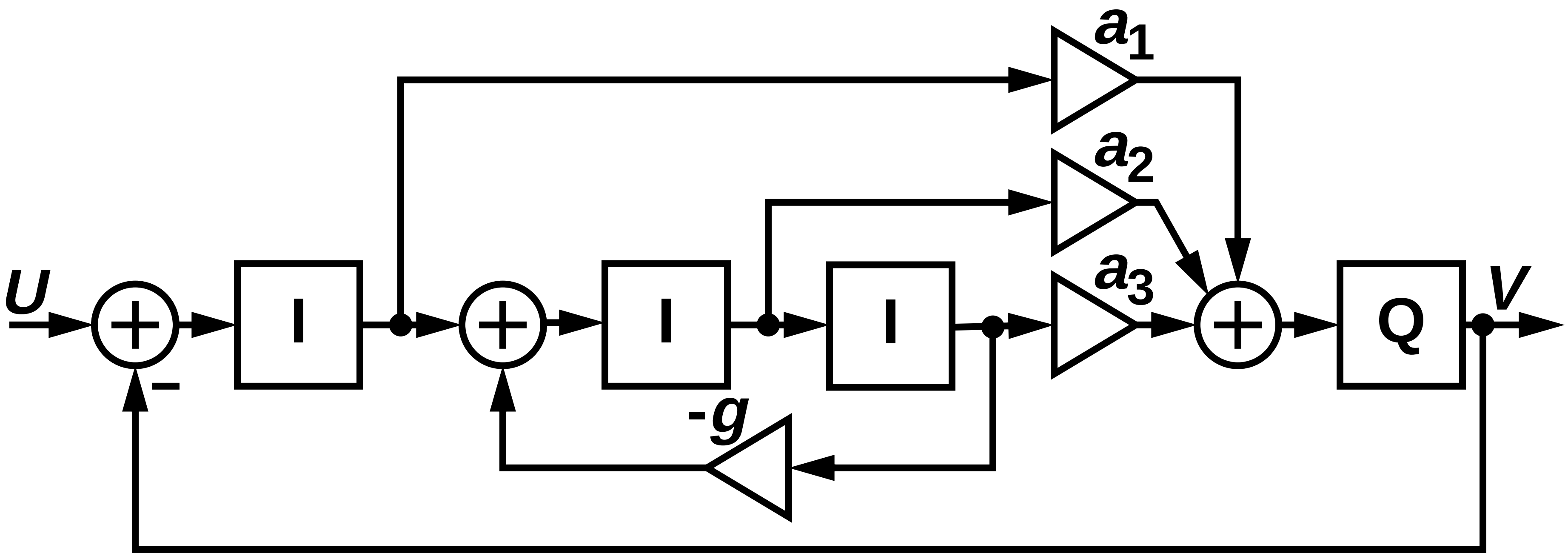
Mid-rise quantizer;
v: odd integers [–*M*,+*M*]

Feedback Topology



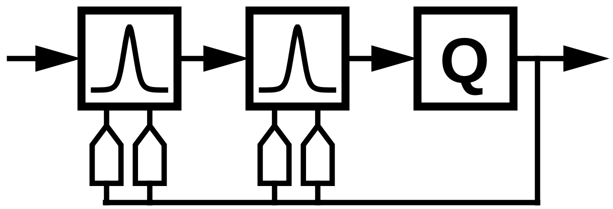
- *N* integrators precede the quantizer
- Feedback from the quantizer to the input of each integrator (via a DAC)
- Local feedback around pairs of integrators to set the NTF's zeros
- Multiple input feed-in branches are possible

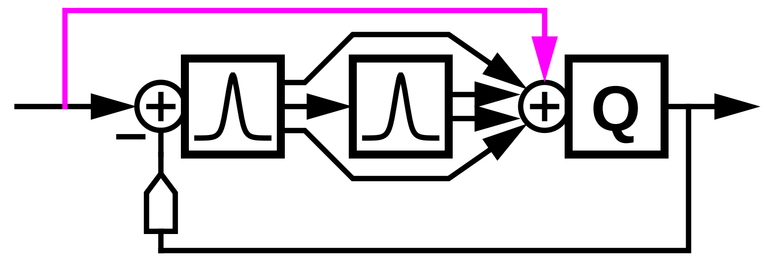
Feedforward Topology



- *N* integrators in a row
- Each integrator output is fed forward to the quantizer
- Local feedback around pairs of integrators to control NTF zeros
- Multiple input feed-in branches also possible

Feedback vs. Feedforward STF





$$L_0(z) = \frac{N_0(z)}{D(z)} \qquad L_1(z) = \frac{N_1(z)}{D(z)}$$

$$NTF(z) = \frac{1}{1 - L_1(z)} = \frac{D(z)}{N_1(z) - D(z)}$$

poles of LF are zeros of NTF

$$STF(z) = \frac{L_0(z)}{1 - L_1(z)} = \frac{N_0(z)}{N_1(z) - D(z)}$$

same poles as NTF

zeros = zeros of L_0

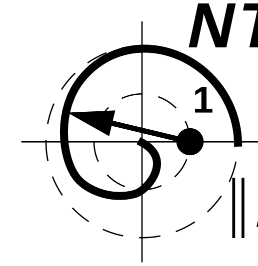
STF often has no zeros, only poles.

$$L_0(z) = -L_1(z) = L(z)$$

$$NTF(z) = \frac{1}{1 - L_1(z)} = \frac{1}{1 + L(z)}$$

$$STF(z) = \frac{L(z)}{1 + L(z)} = 1 - NTF(z)$$

$NTF(e^{j\omega})$



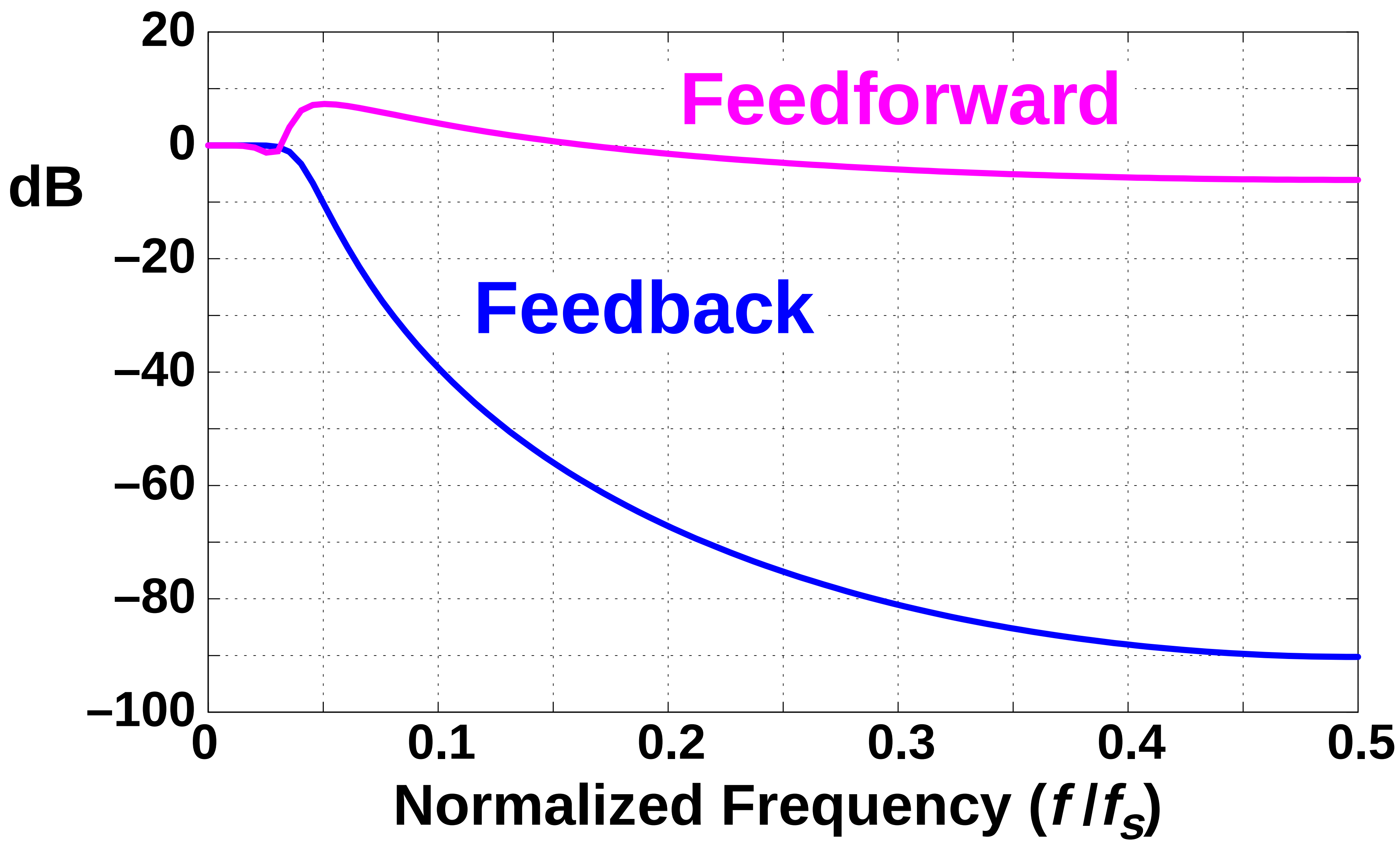
$\|NTF\|_\infty$

$$\|STF\|_\infty \approx \|NTF\|_\infty + 1$$

With extra feed-in to Q, $STF(z) = 1$.

STF Comparison

5^{th}-Order; Single Feed-In



Feedforward vs. Feedback

- **FF has relaxed dynamic range requirements**

  "All stages except the last have attenuated signal components."

- **FB has better STF and, for CT modulators, a better AAF**

  In a discrete-time modulator, the STF of FF can be made unity by adding a signal feedforward term to the input of the quantizer.

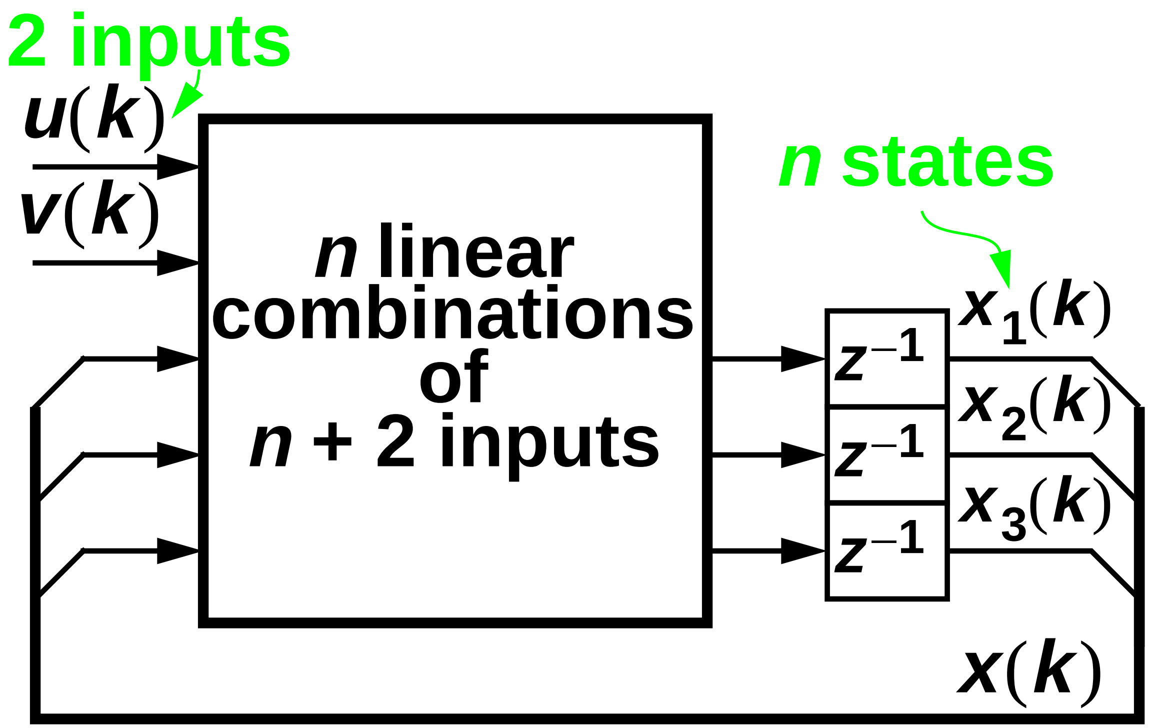
- **FB needs many DACs;**
  FF needs a summation block

  Can do partial summation before the last integrator.

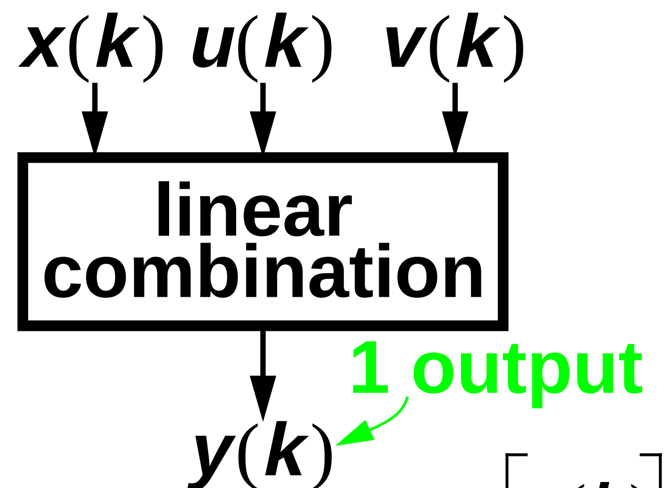
- **FF with signal to quantizer: timing can be tricky**

  Need to quantize *u* and feed it back in zero time.

ABCD: A *State-Space* Representation of the Loop Filter



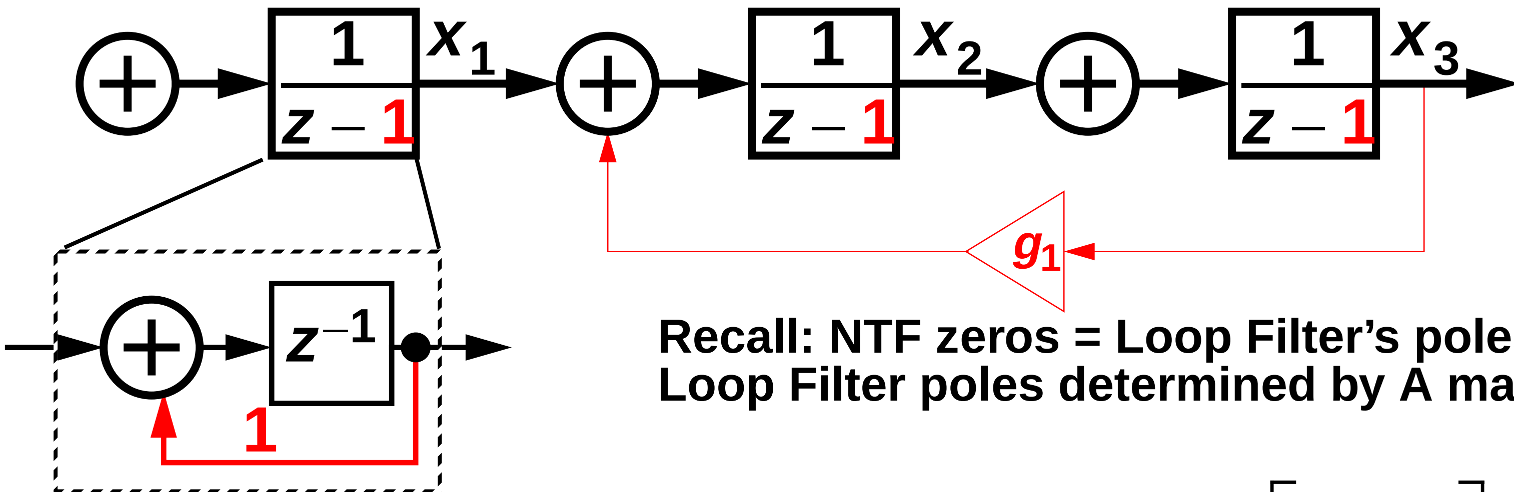
$$x(k+1) = \overset{n\times n}{A}x(k) + \overset{n\times 2}{B}\begin{bmatrix}u(k)\\ v(k)\end{bmatrix}$$



$$y(k) = \underset{1\times n}{C}x(k) + \underset{1\times 2}{D}\begin{bmatrix}u(k)\\ v(k)\end{bmatrix}$$

$$ABCD = \begin{array}{c|cc} & n & 2 \\ \hline n & A & B \\ 1 & C & D \end{array}$$

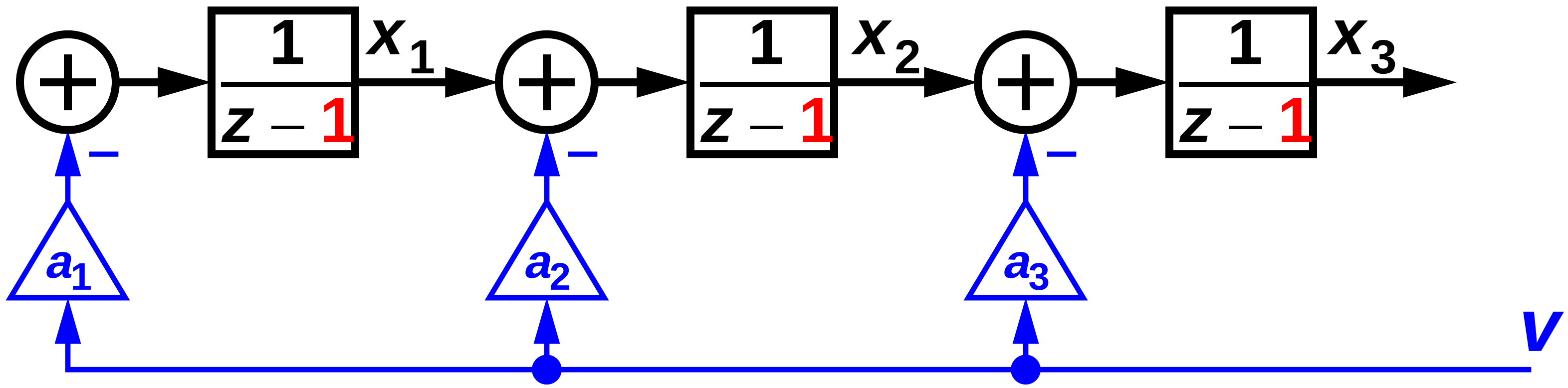
Ex.: Cascade of Integrators Feedback (CIFB) Topology



Recall: NTF zeros = Loop Filter's poles.
Loop Filter poles determined by A matrix.

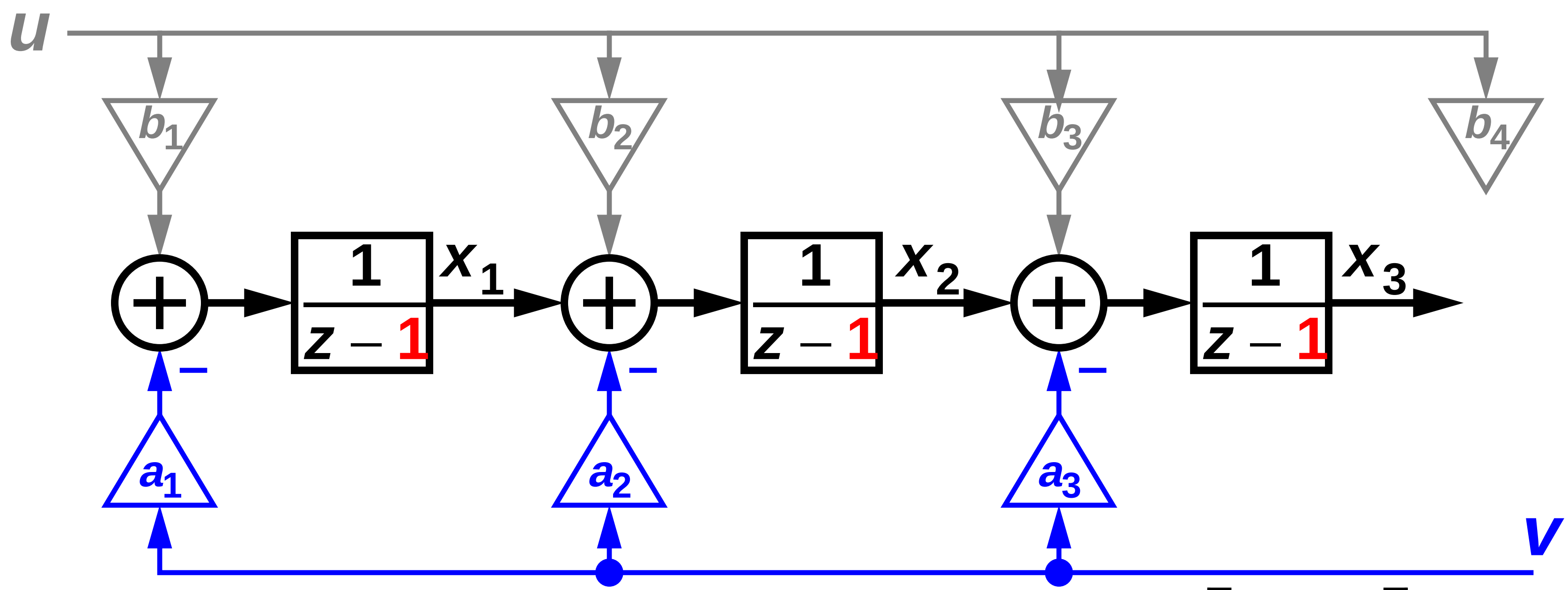
$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \\ y(k) \end{bmatrix} = \left[\begin{array}{ccc|cc} 1 & 0 & 0 & - & - \\ 1 & 1 & g_1 & - & - \\ 0 & 1 & 1 & - & - \\ \hline 0 & 0 & 1 & - & - \end{array}\right] \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ u(k) \\ v(k) \end{bmatrix}$$

CIFB cont'd:
a_i Control NTF & STF Poles



$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \\ y(k) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & - & -a_1 \\ 1 & 1 & 0 & - & -a_2 \\ 0 & 1 & 1 & - & -a_3 \\ 0 & 0 & 1 & - & 0 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ u(k) \\ v(k) \end{bmatrix}$$

b_i Control STF Zeros



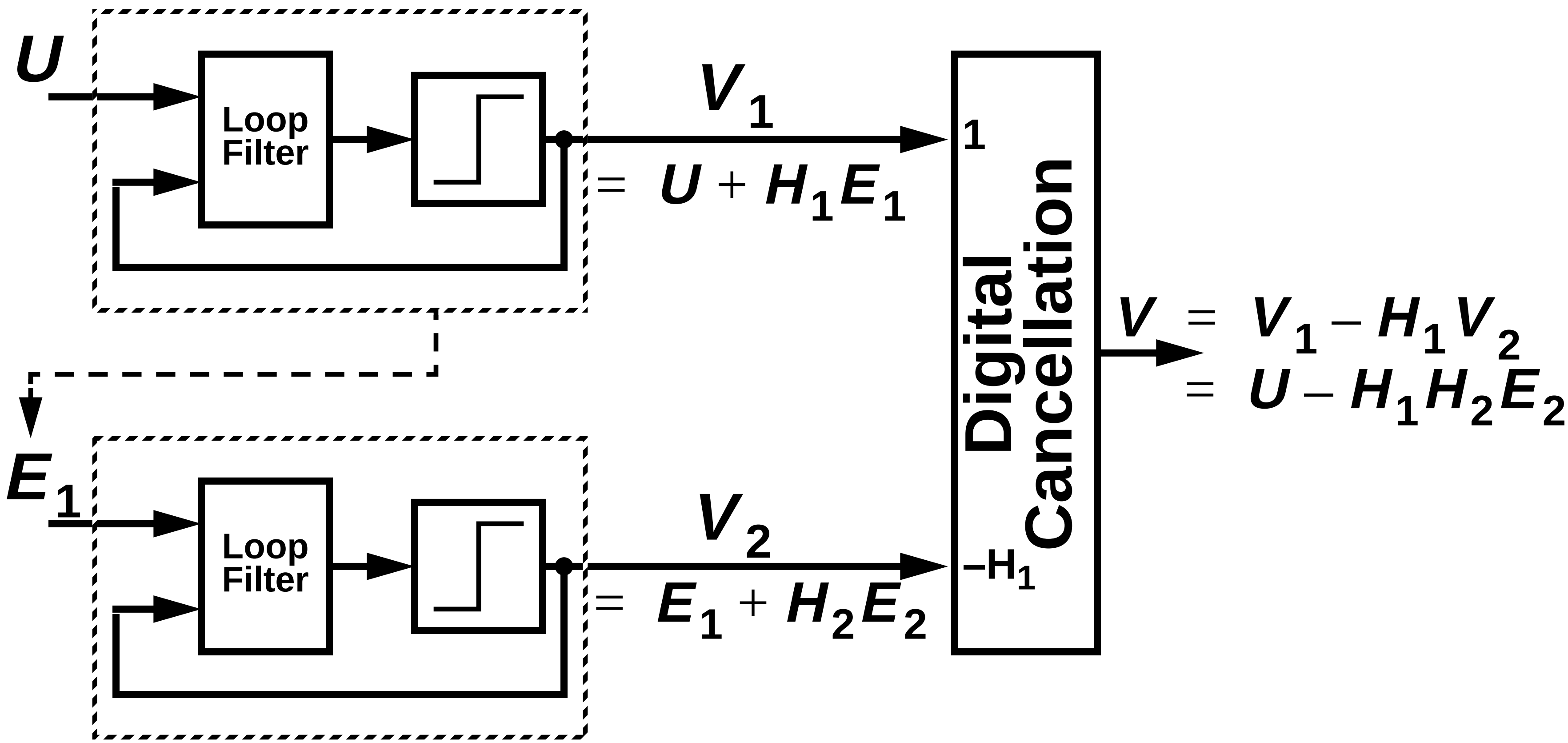
$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \\ y(k) \end{bmatrix} = \left[\begin{array}{ccc|cc} 1 & 0 & 0 & b_1 & -a_1 \\ 1 & 1 & 0 & b_2 & -a_2 \\ 0 & 1 & 1 & b_3 & -a_3 \\ \hline 0 & 0 & 1 & b_4 & 0 \end{array}\right] \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ u(k) \\ v(k) \end{bmatrix}$$

ABCD and the Toolbox

- **`simulateDSM` simulates a modulator given an ABCD description of its loop filter**
- **`realizeNTF` gives (unscaled) coefficients for any of the supported topologies**
- **`stuffABCD` produces an ABCD matrix given the coefficients for one of the supported topologies**
  - **`mapABCD` performs the inverse operation.**
- **`scaleABCD` does dynamic range scaling on any ABCD matrix**
- **`calculateTF` calculates the NTF and STF from ABCD**
  - **Useful for checking implementation of new topologies.**

Cascade (MASH) Modulators

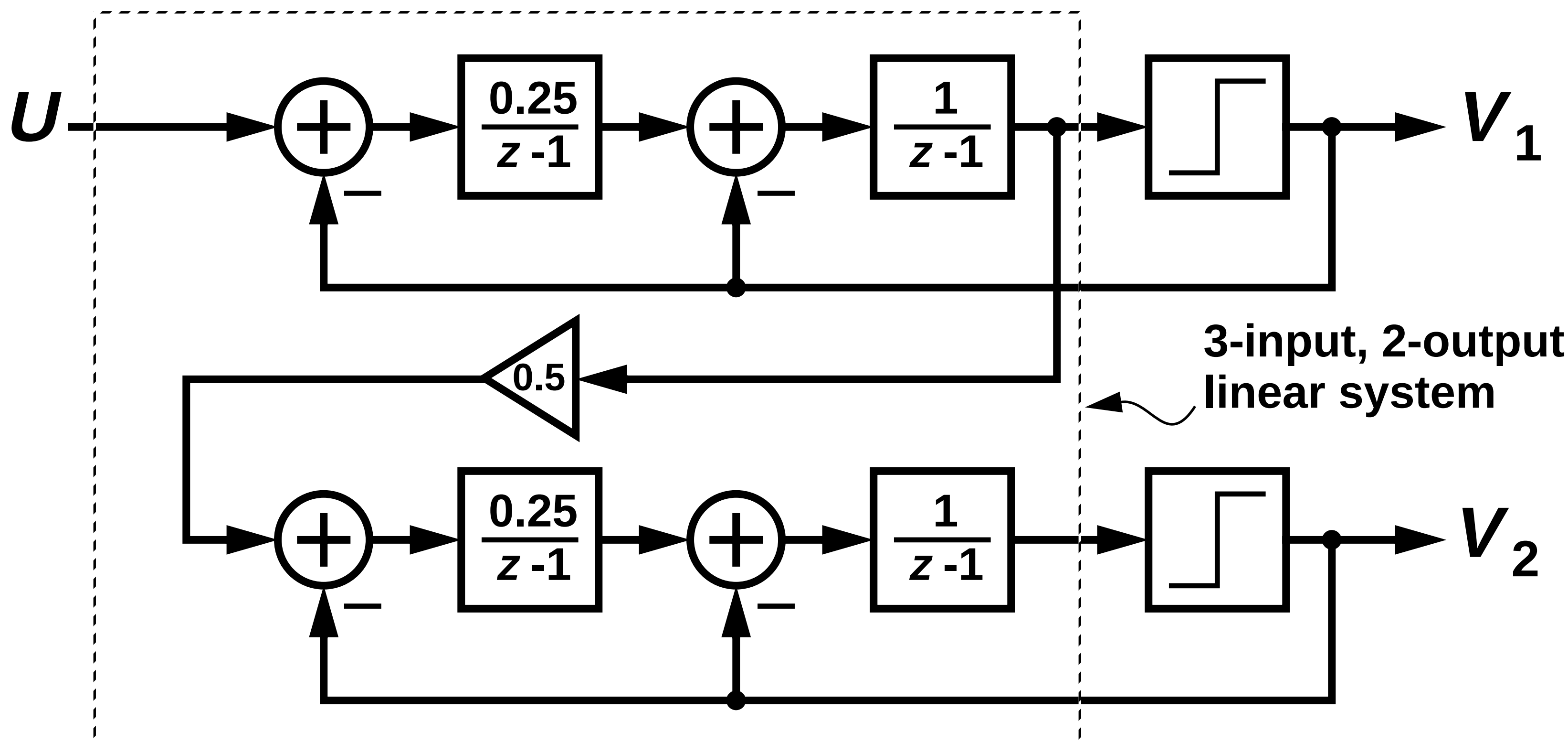
- **Put two (or more) modulators in "series"**



- **The resulting NTF is the *product* of the individual NTFs**

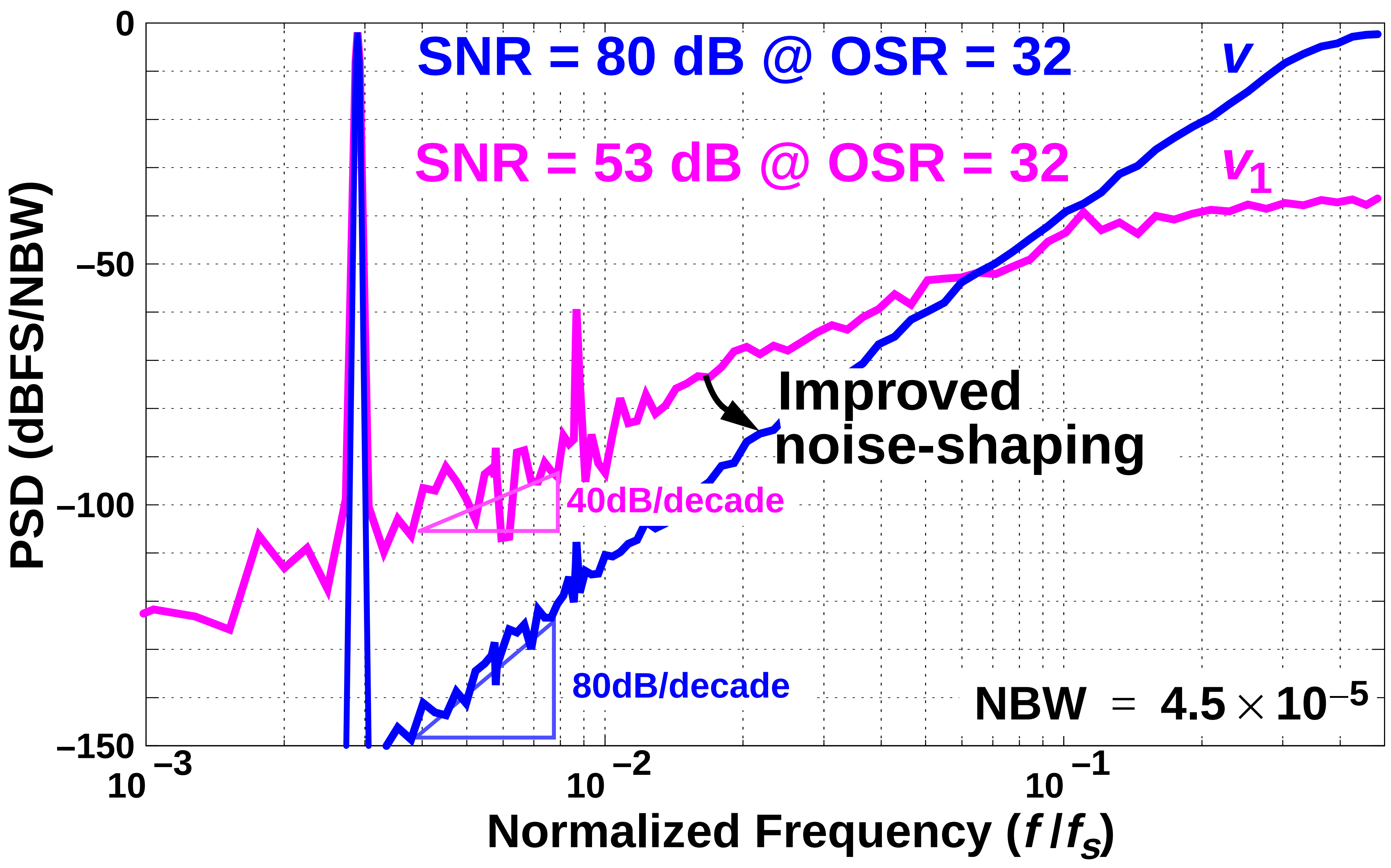
Example: 2-2 Cascade

- **Use Two MOD2b:** $H(z) = \left(\dfrac{z-1}{z-0.5}\right)^2$

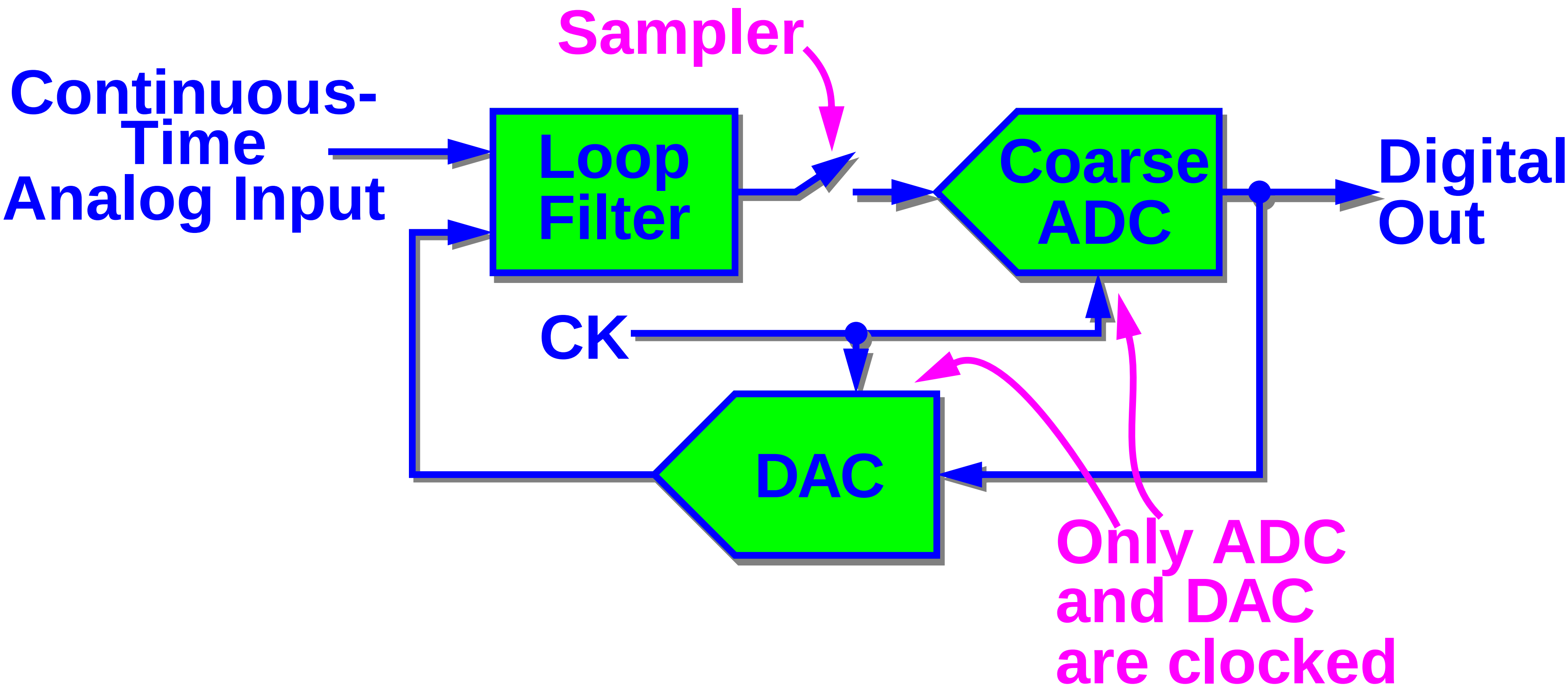


$$\left(V = \frac{1}{z^3}V_1 + \frac{8(z-1)^2(z-0.5)^2}{z^3(z-0.75)}V_2\right) \Rightarrow \left(H(z) = \frac{8(z-1)^4}{(z-0.75)}\right)$$

Example MASH Spectra



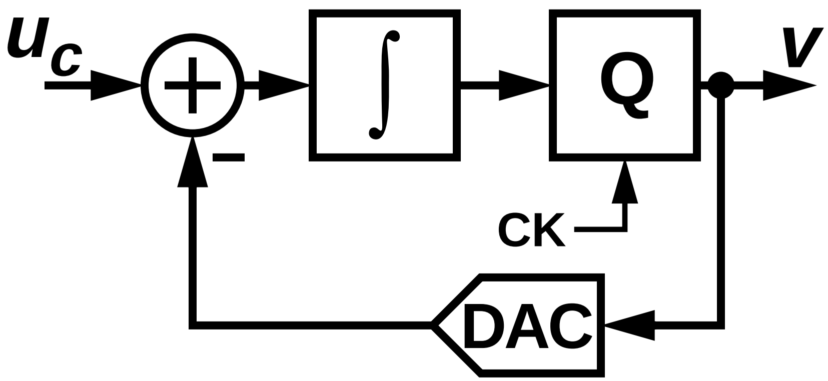
A Continuous-Time ΔΣ ADC



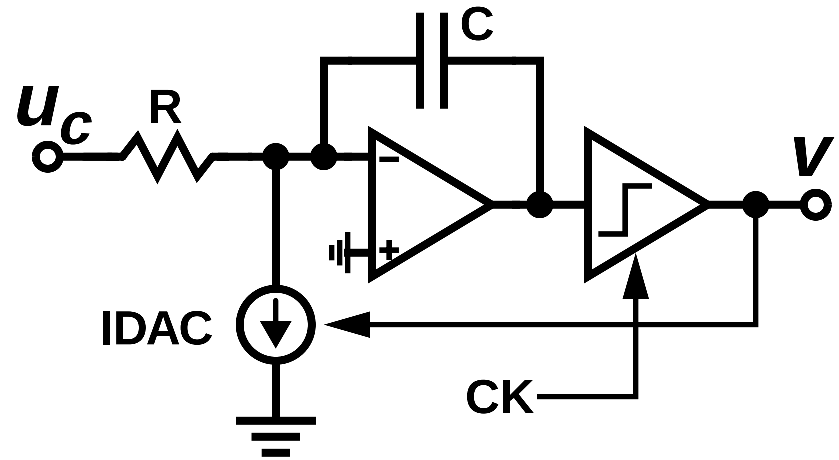
- Loop filter implemented with continuous-time circuitry
- Sampling occurs after the loop filter

Example: MOD1-CT

Block Diagram



Schematic

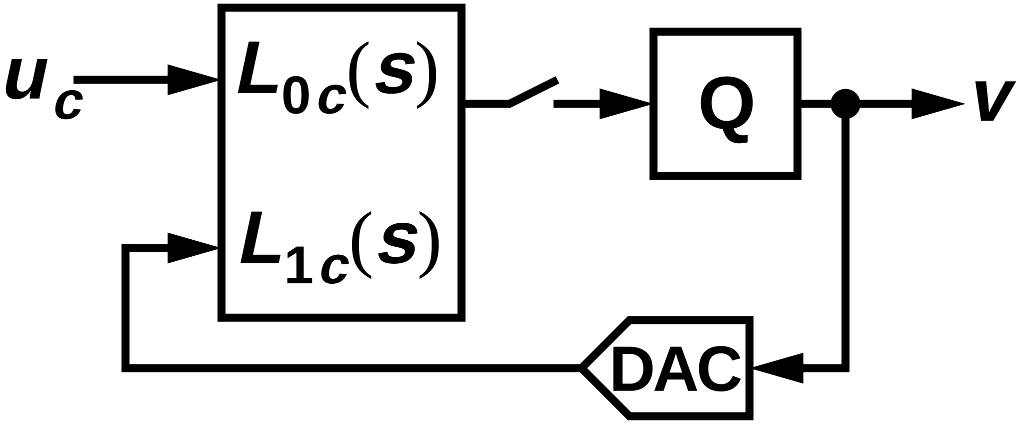


- **Note: Input is a simple resistor, *not* a switched capacitor**

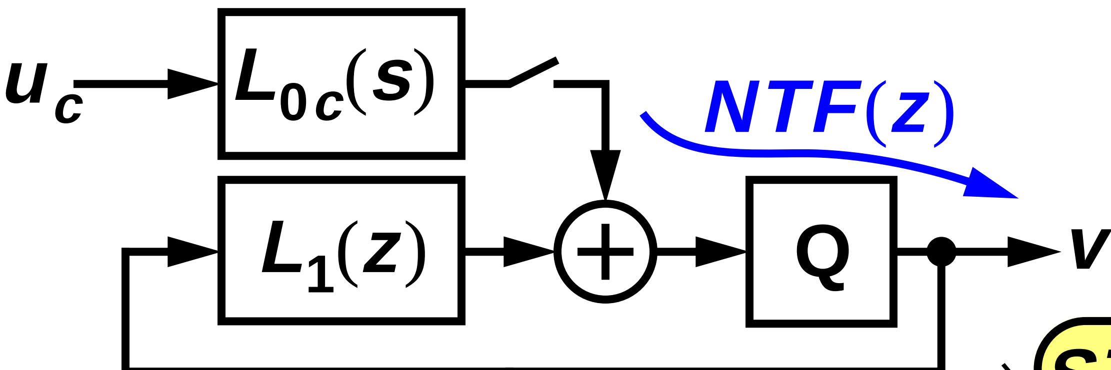
  CT ADCs are easier to drive than DT ADCs.

Inherent Anti-Aliasing

- $\Delta\Sigma$ **ADC with CT Loop Filter**



- **Equivalent system with DT feedback path**

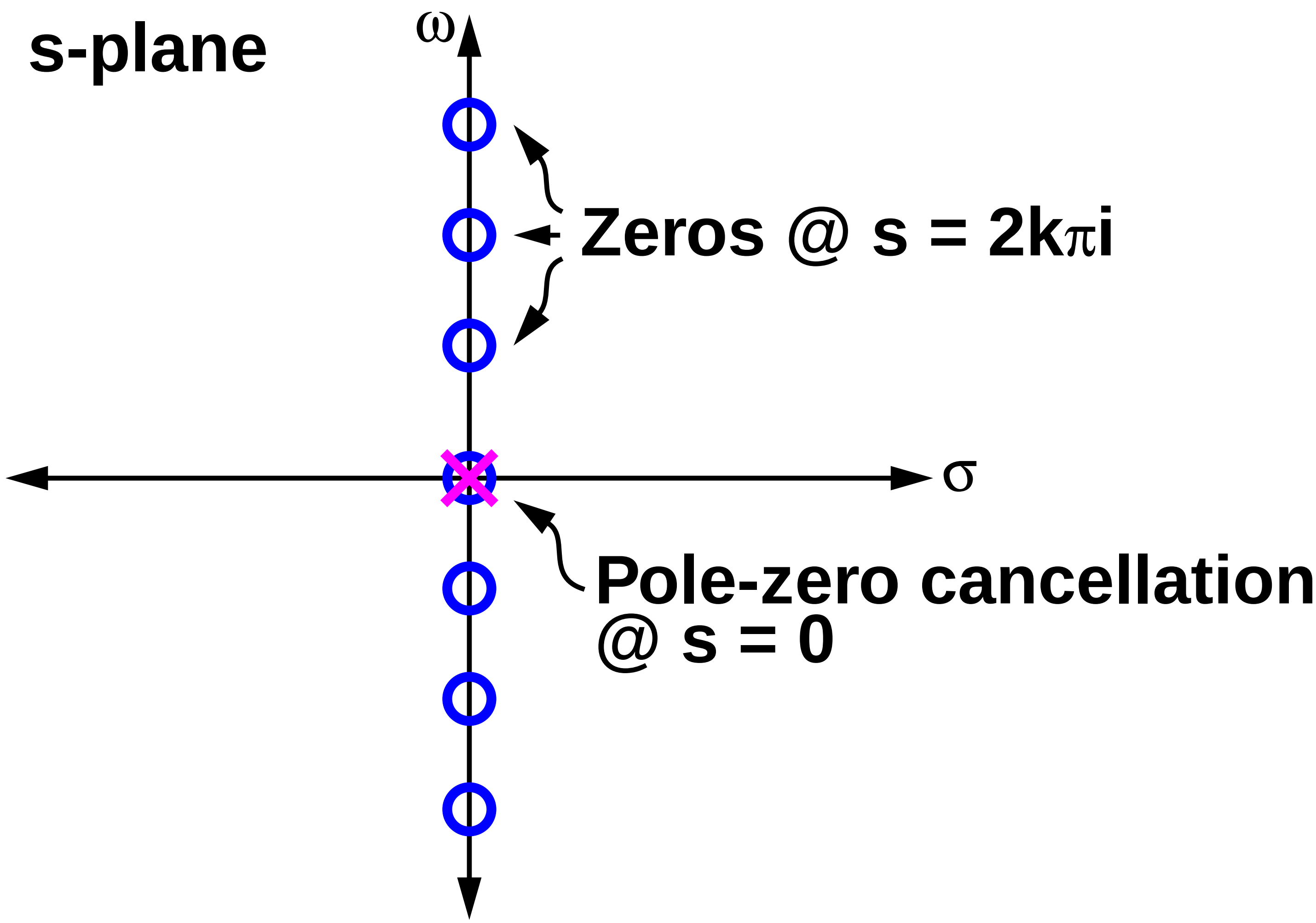


$\Rightarrow$ $\boxed{STF = L_{0c}(s) NTF(z)}$

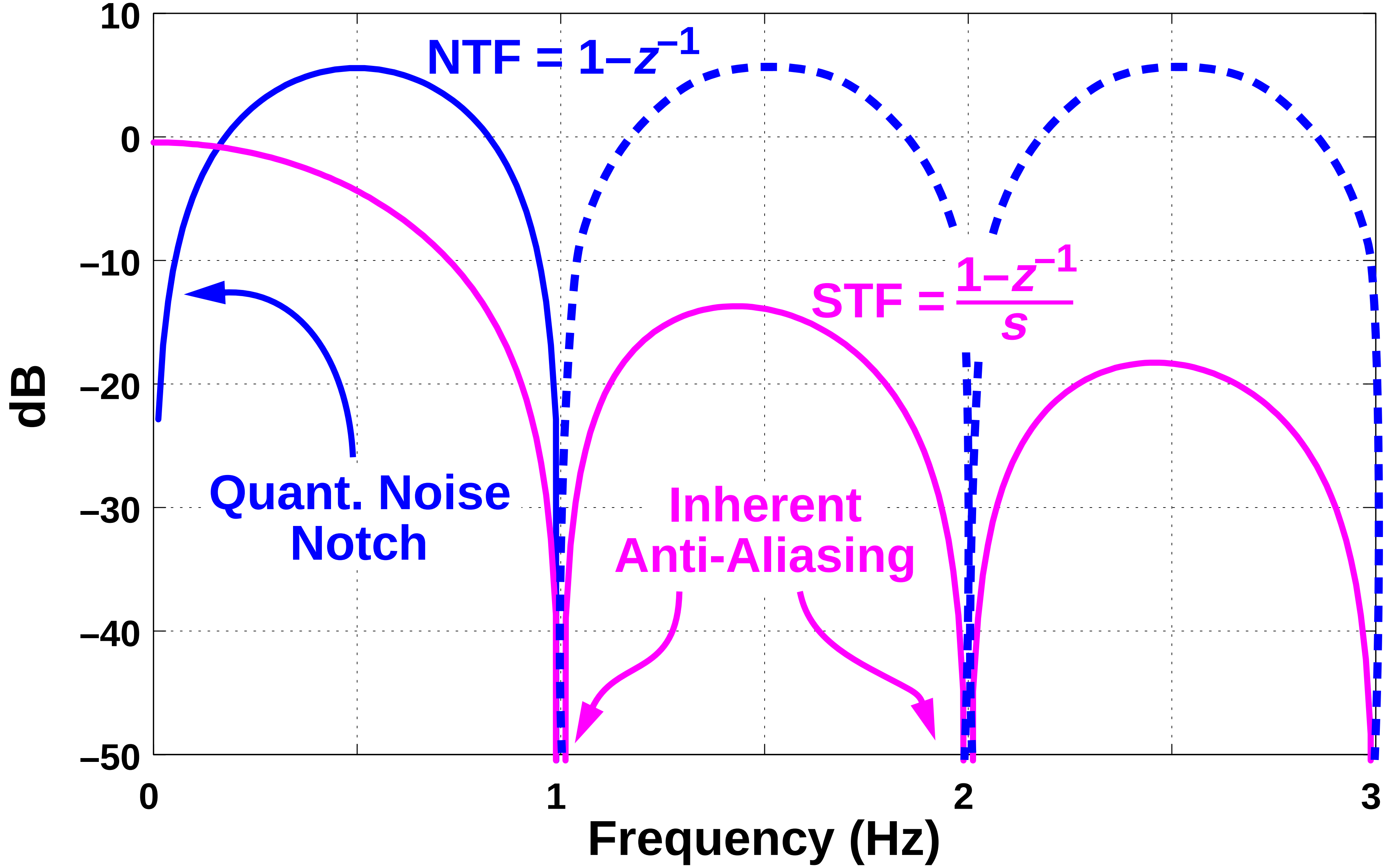
Example: MOD1-CT $STF = \frac{1 - z^{-1}}{s}$

Recall $z = e^s$

s-plane



MOD1-CT Frequency Responses



Inherent AAF Summary

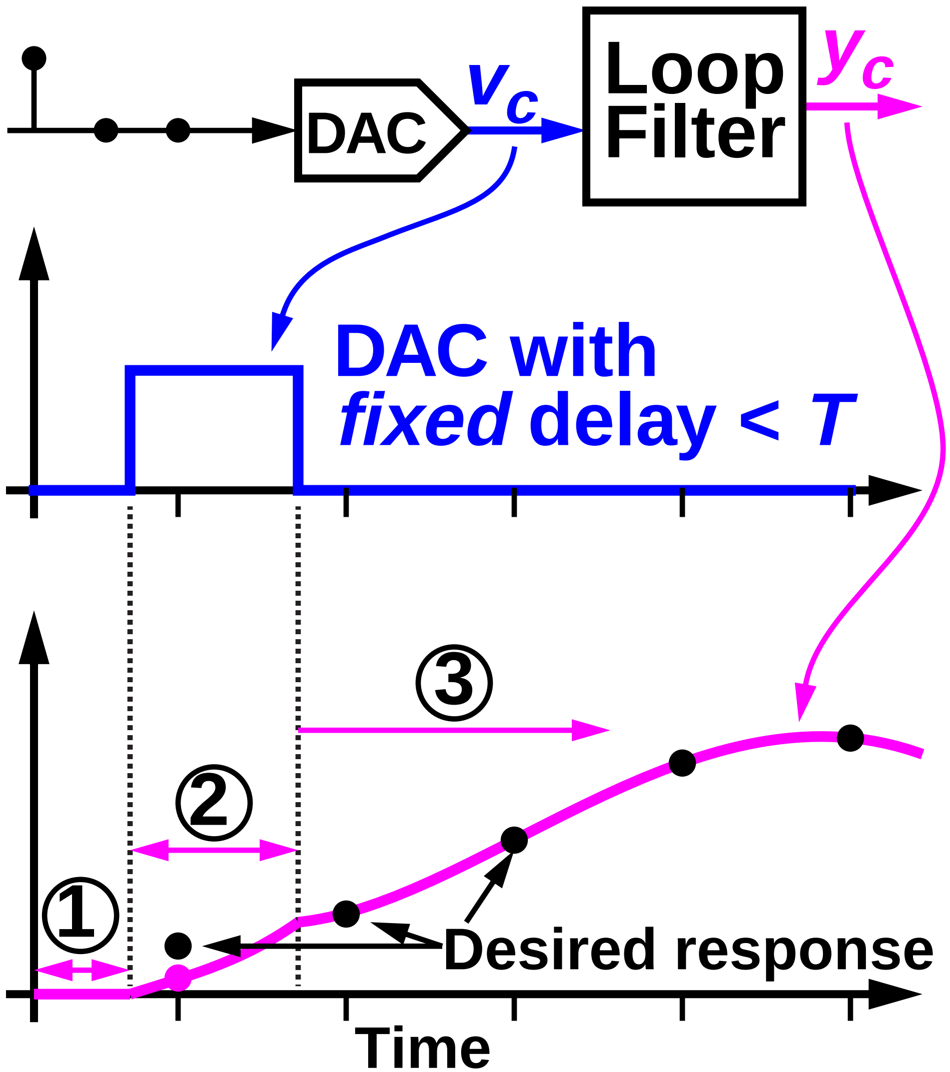
$$STF = L_{0c}(s)NTF(z)$$

- STF contains the zeros of the NTF
- Any frequency which aliases to the passband is attenuated by at least as much as the quantization noise

  Anti-alias performance tracks modulator order.
  Also true for MASH systems.
- The effective anti-alias filter is

$$AAF(f) = \frac{STF(f)}{STF(f_{alias})} = \frac{L_{0c}(f)}{L_{0c}(f_{alias})}$$

Effect of Quantizer/DAC Delay



1 Loop outputs zero until DAC pulse begins

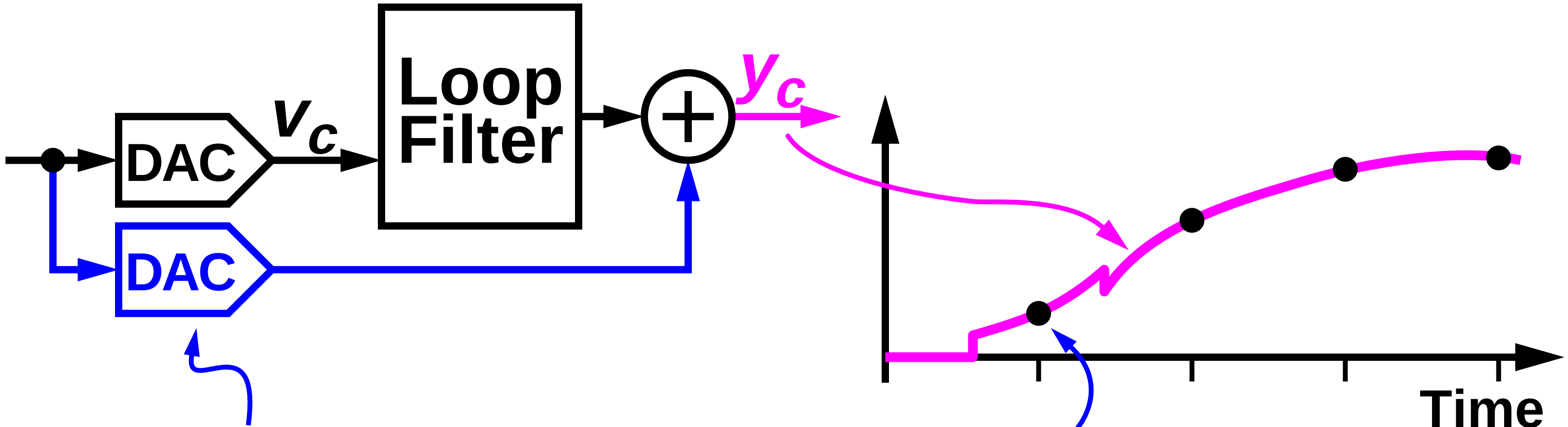
2 Loop responds as if input were a step

3 Loop follows trajectory of an n^{th}-order linear system with zero input but non-zero initial conditions

⇒ At best, samples of the pulse response will match the desired impulse response except at the first point.
The NTF will be wrong.

Compensating for Feedback Delay

- **To fix the first sample, add a *direct feedback* DAC**

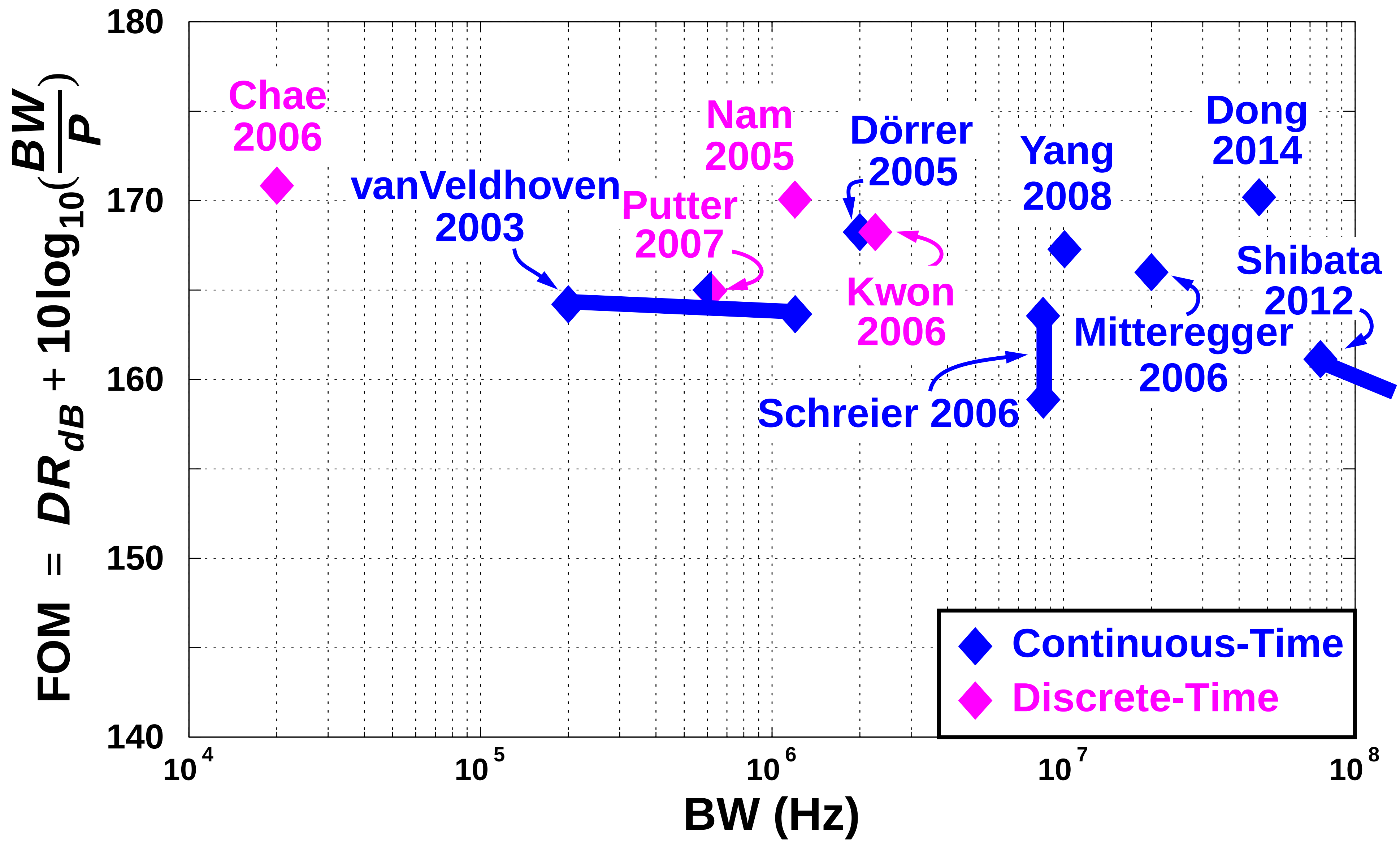


Direct Feedback DAC corrects the first sample

- **With enough DACs, one for each errant sample, any finite number of points can be repaired**

  In principle, the delay of the main feedback path can be anything, but the system becomes sensitive to coefficient errors.

DT ΔΣ vs. CT ΔΣ



180
170
160
150
140
10^4
10^5
10^6
10^7
10^8
BW (Hz)
$FOM = DR_{dB} + 10\log_{10}(\frac{BW}{P})$
Chae 2006
Nam 2005
Dörrer 2005
Yang 2008
Dong 2014
vanVeldhoven 2003
Putter 2007
Kwon 2006
Shibata 2012
Mitteregger 2006
Schreier 2006
Continuous-Time
Discrete-Time

References— DT vs. CT

BW (Hz)	DR (dB)	P (mW)	FOM	Reference	Architecture *N*=order (*M*=#steps)
20k	85	0.036	172	Chae, ISSCC 2008:27.2	ΔΣ SC 3(1)
614k	82	3.1	165	Putter, ISSCC 2007:13.4	ΔΣ A-RC+SC 6(1)
1.2M	82	8	164	vanVeldhoven, ISSCC 2003:3.4	QΔΣ gm-C 5(1)
1.2M	96	44	171	Nam, JSSC 2005-09	ΔΣ SC 2(32)-2(8)
2M	80	3.0	168	Dörrer, ISSCC 2005:27.1	ΔΣ A-RC 3(15)
2.2M	86	14	168	Kwon, ISSCC 2006:3.4	ΔΣ SC 2(4)
8.5M	88	375	162	Schreier, ISSCC 2006:3.2	QBPΔΣ A-RC 4(16)
10M	87	100	167	Yang, ISSCC 2008:27.6	ΔΣ A-RC 5(7)
20M	76	20	166	Mitteregger, ISSCC 2006:3.1	ΔΣ A-RC 3(15)
53M	88	235	172	Dong, JSSC 2014-12	ΔΣ A-RC 0(16)-3(16)
75M	80	550	161	Shibata, JSSC 2012-12	BPΔΣ LC/A-RC 6(16)

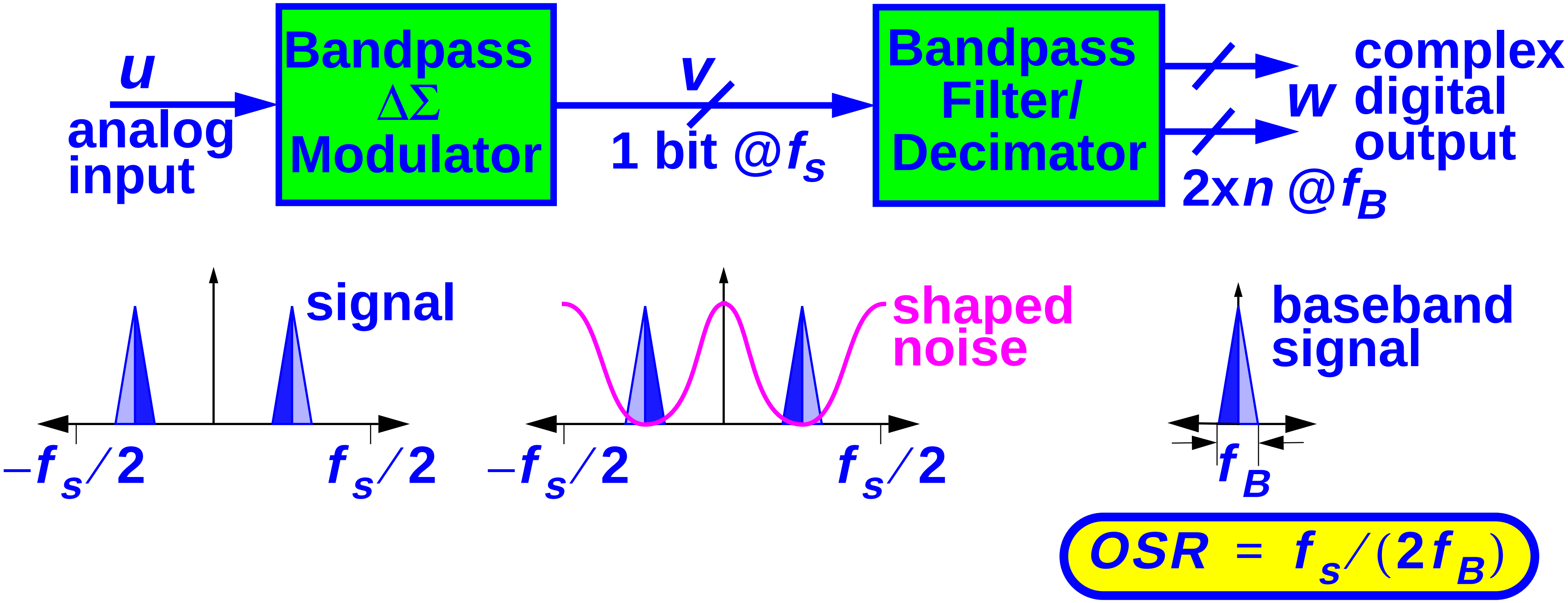
Advantages of Discrete-Time

1. **Less sensitive to jitter**
2. **Accurate transfer functions regardless of f_{CK}**
3. **FF topology with no STF-peaking is possible**
4. **DAC dynamics are non-critical**
5. **Math is simpler**

Advantages of Continuous-Time

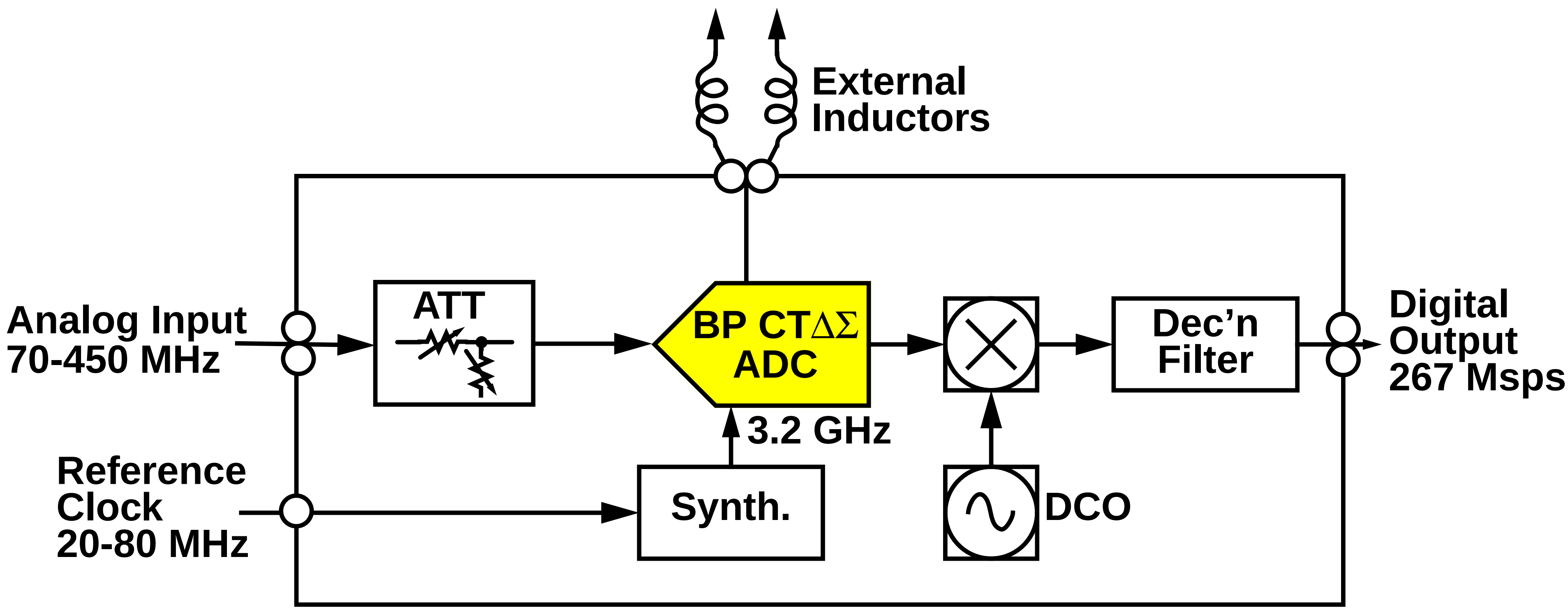
1. **Higher speed**
2. **Inherent anti-aliasing**
3. **Easier to drive (well-defined Z_{in})**
4. **Sampling is non-critical**
5. **Lower power (?)**

A Bandpass ΔΣ ADC System



- ADC converts its analog input into a noise-shaped digital output
- DSP removes out-of-band noise (and signals) and translates the signal to baseband

Example CT Bandpass ΔΣ ADC: The AD6676 [Shibata 2012]

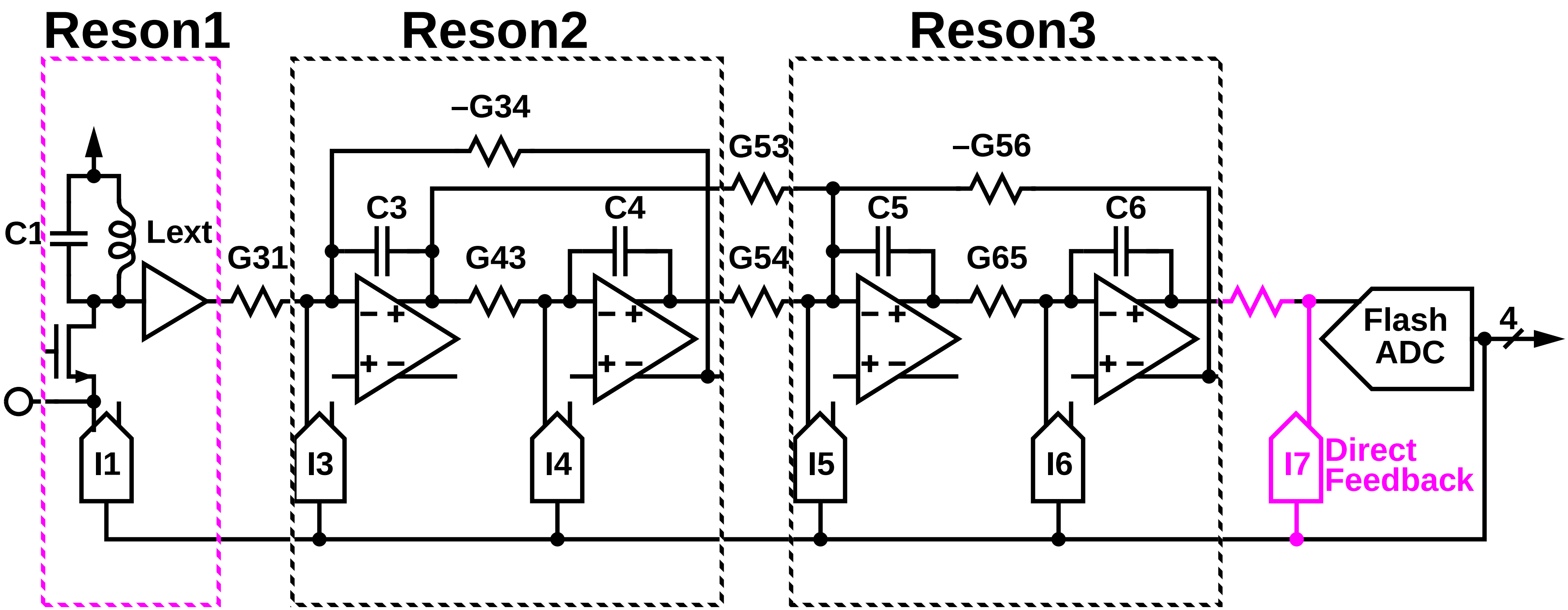


- IF subsystem containing attenuator, synthesizer, CT BP ΔΣ ADC and digital filter

AD6676 Specs

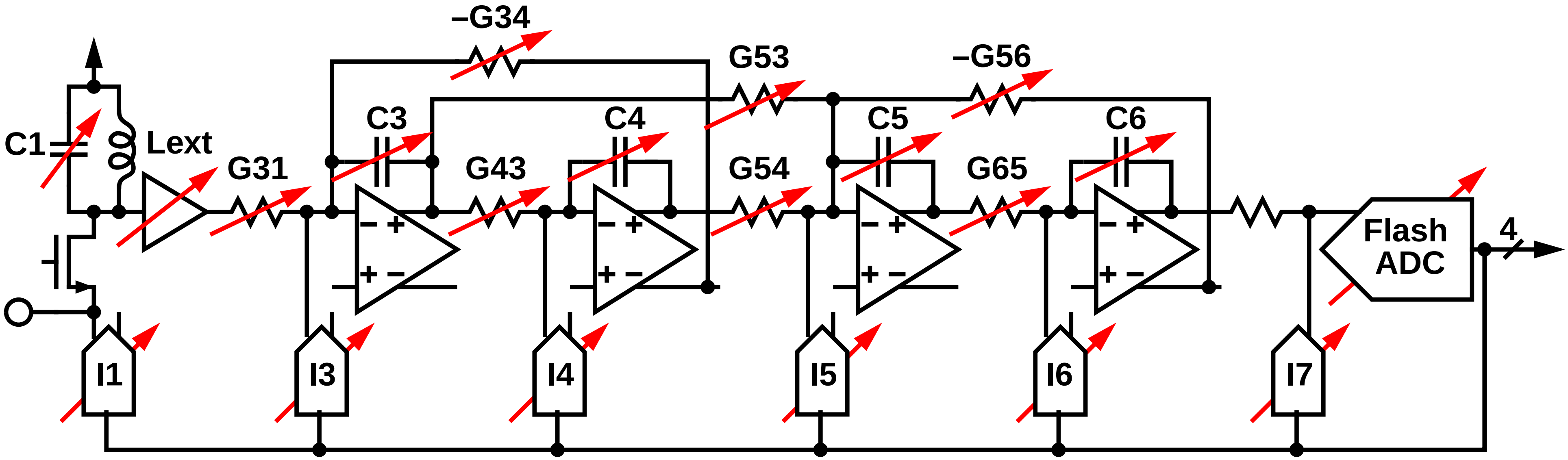
Parameter	Value	Notes
Z_{in}	50 Ω	
ADC FS	–16 to –4 dBm	
Attenuation	0 to +27 dB	
NSD	< –157 dBFS/Hz	BW = 75 MHz; 12-dB att.
IF	70-450 MHz	
BW	up to 100 MHz	<3-dB NSD degradation
F_{ck}	2-3.2 GHz	
Power	1.25 W	Includes digital filter & JESD204B interface

ADC Schematic



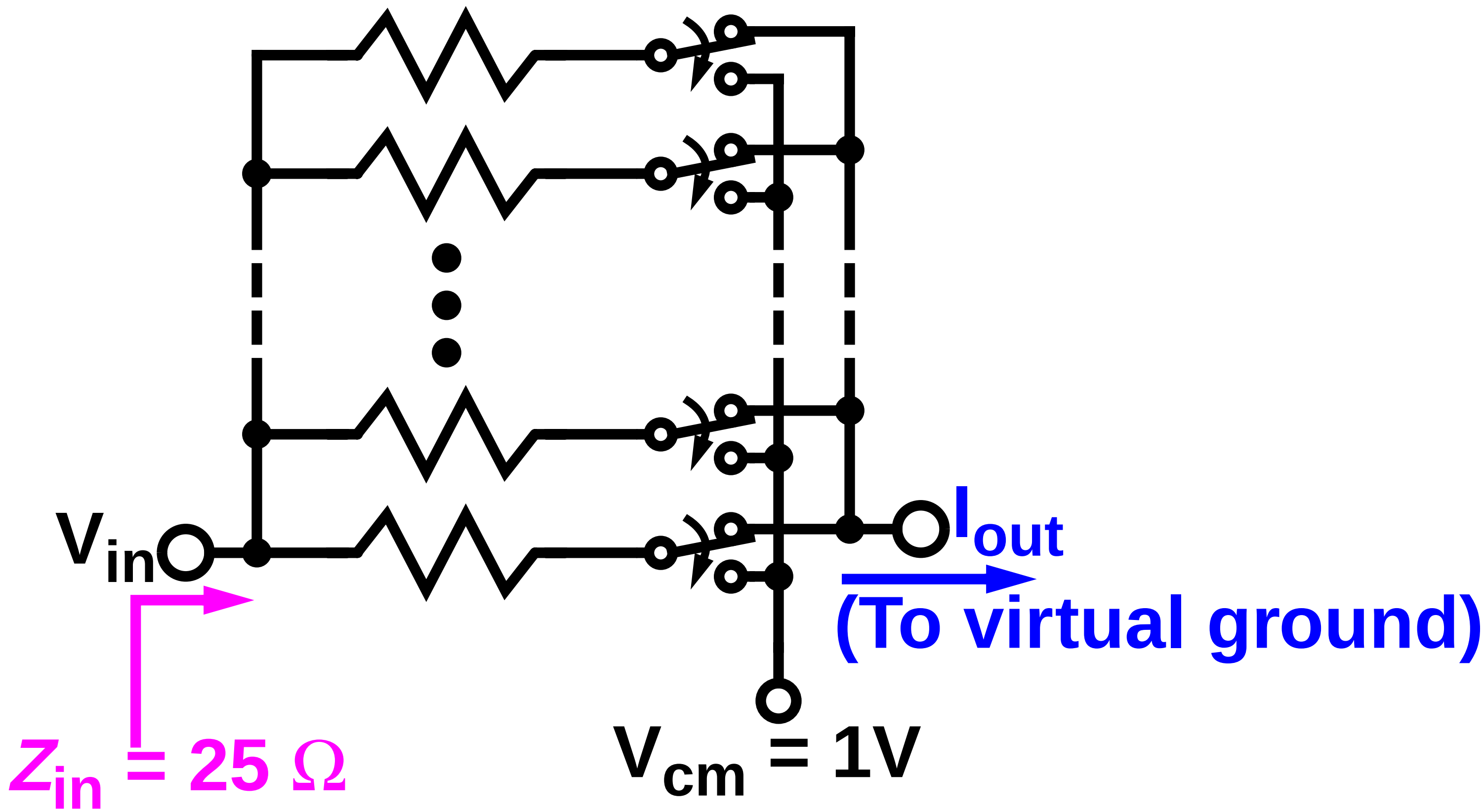
- 6^{th}-order FB-style loop filter
  One LC resonator plus two active-RC resonators.
  G53 makes up for missing DAC2.
- 16-step quantization, [1 2] DAC timing

Programmable Everything



- **Feedback currents, integrating capacitors, inter-stage conductances, flash LSB**
  - **The inductors were also selectable via the cascode (2 choices).**

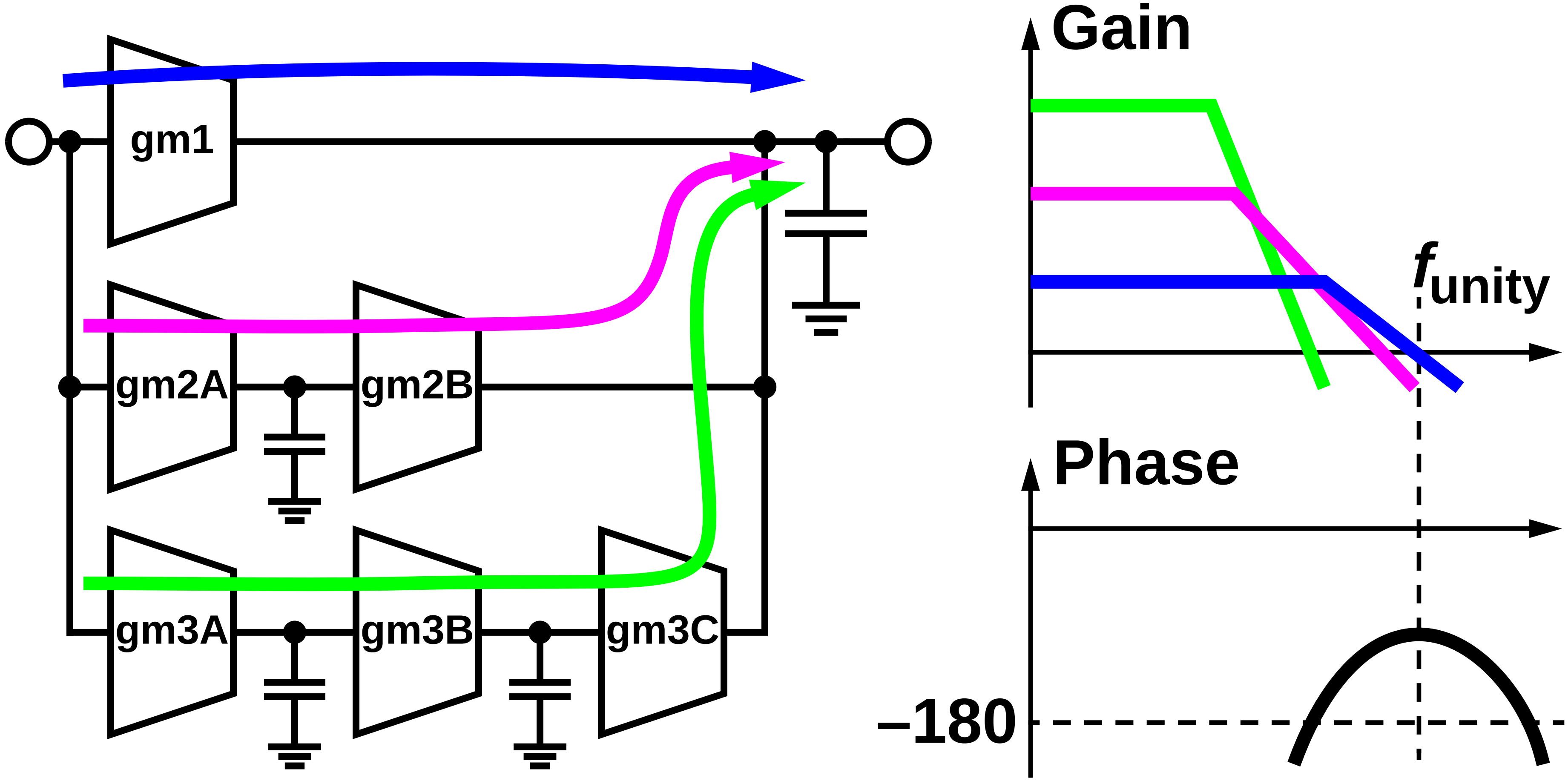
Attenuator Schematic



- **Resistors switched between virtual ground provided by cascode and V_{cm}**

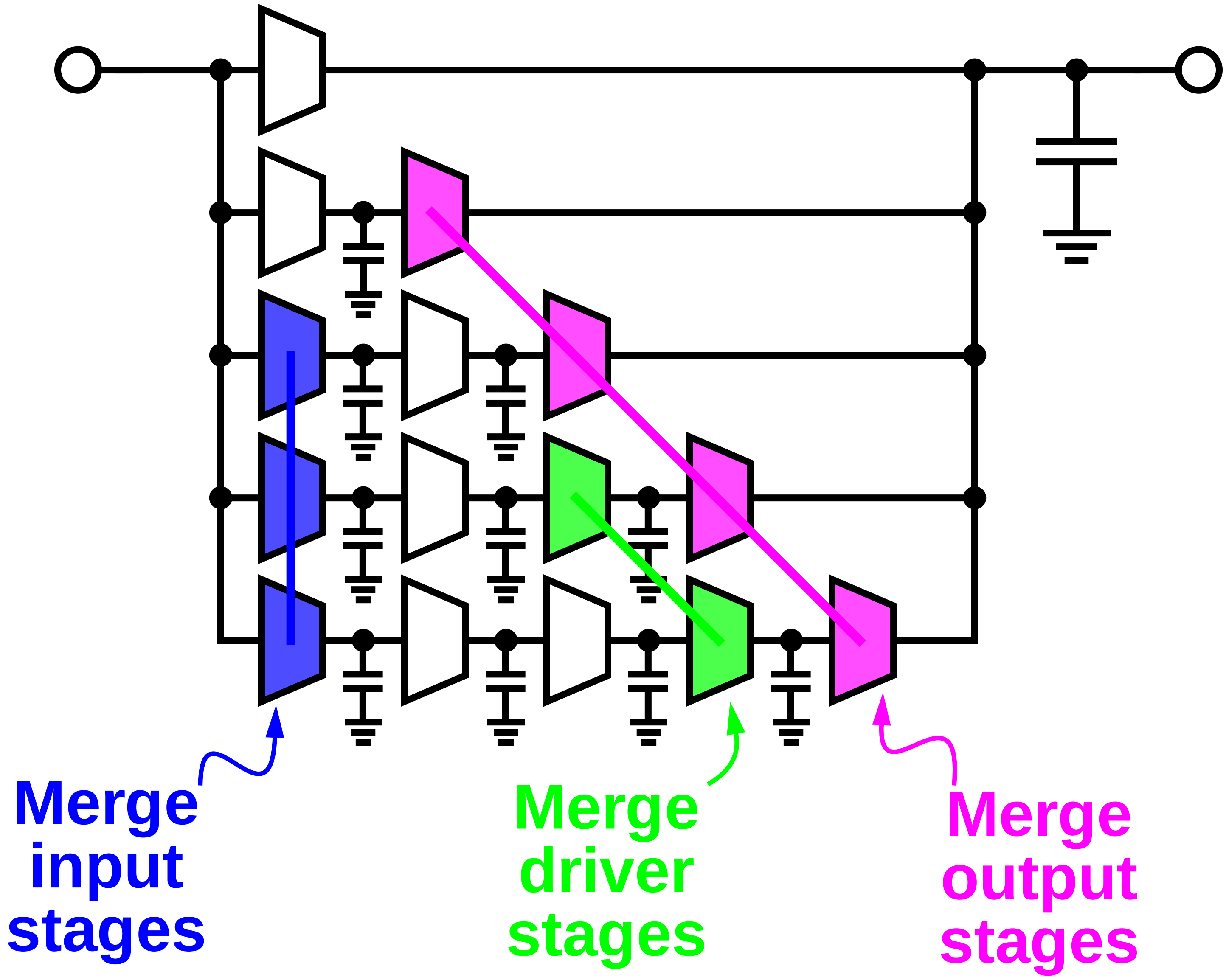
Maintains matching independent of attenuation.

Feed-Forward Amplifier Concept



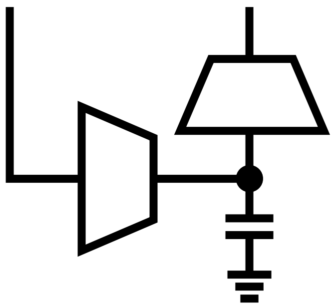
- **High gain provided by longest path**
- **Stability by shorter, higher-bandwidth paths**

FF Amp— Stage-Sharing

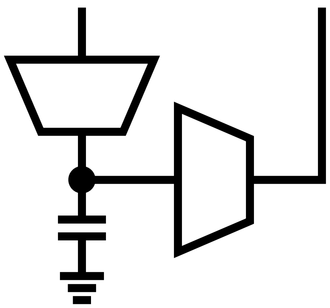


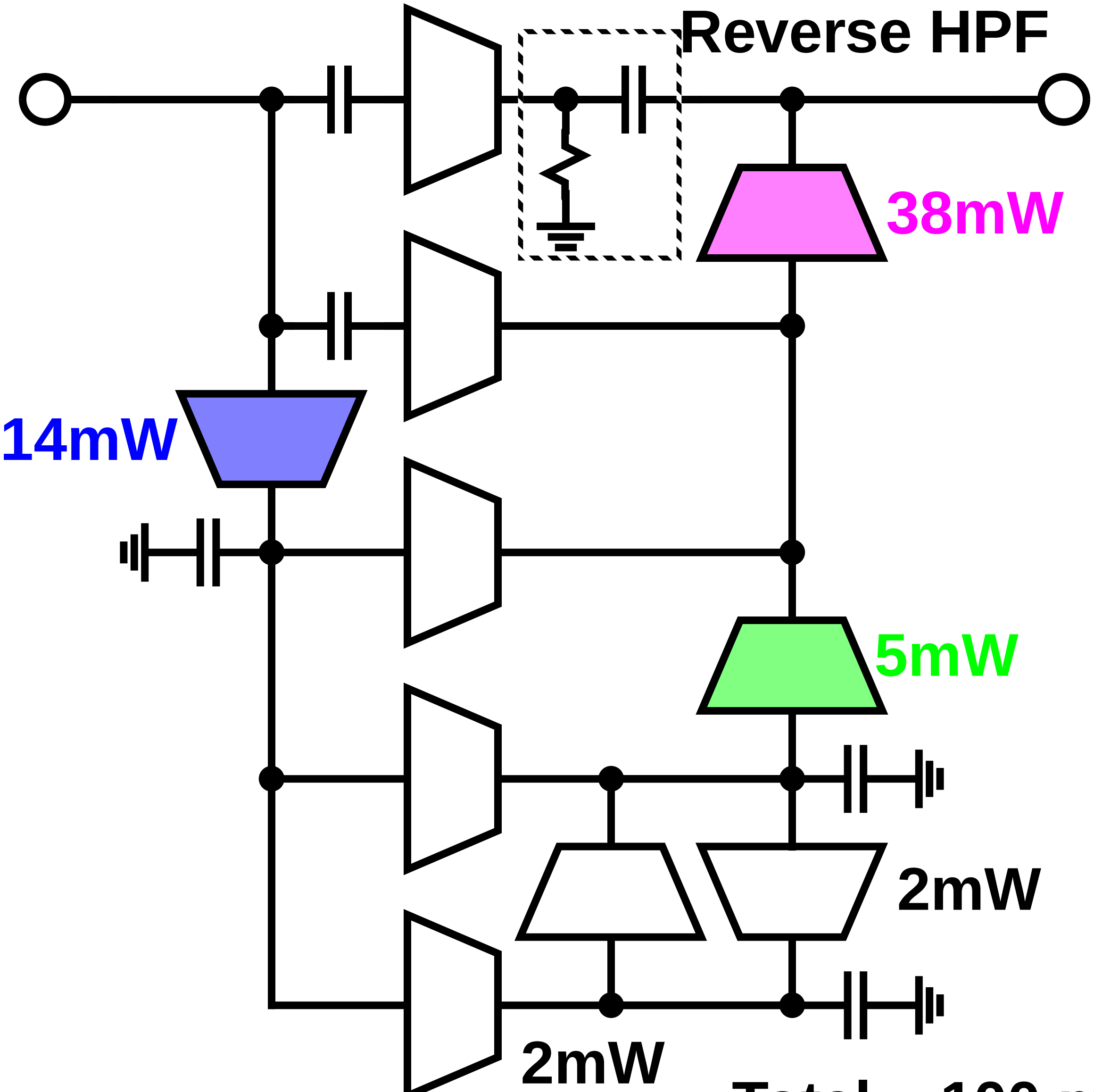
FF Amp— A1 Architecture

Built up by adding L-sections:



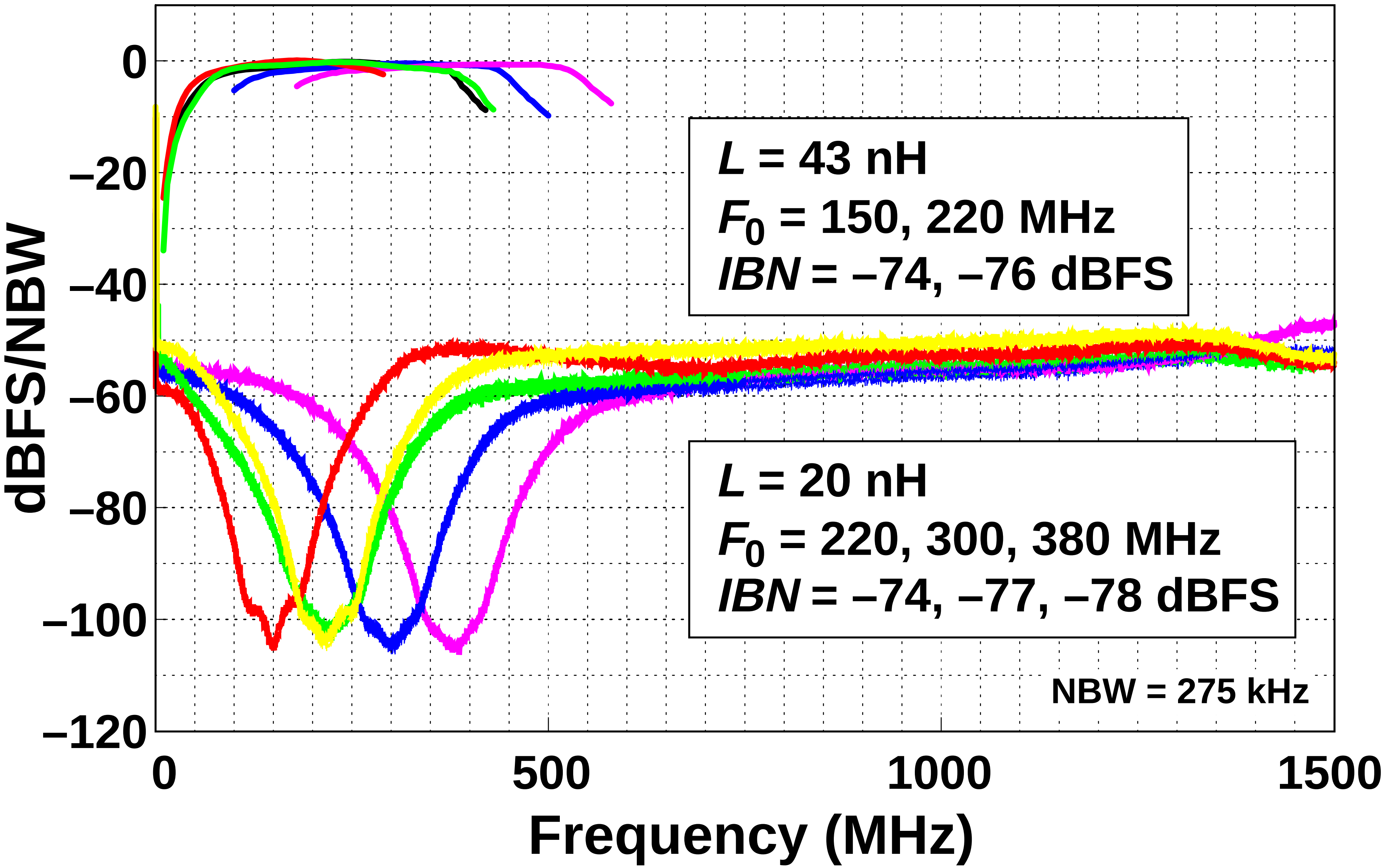
or





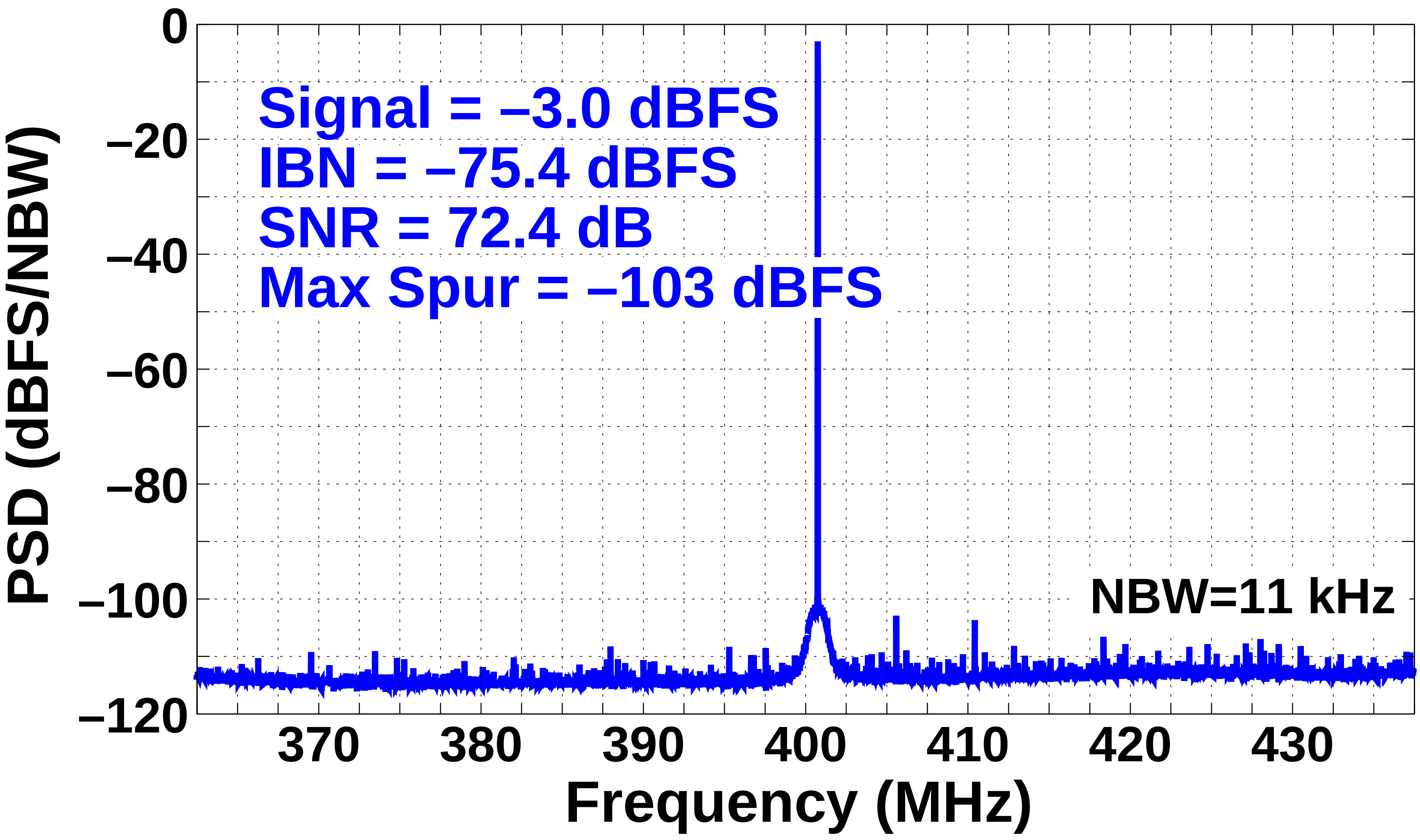
Total = 100 mW

F_0 = 150-400 MHz; F_s = 3 GHz

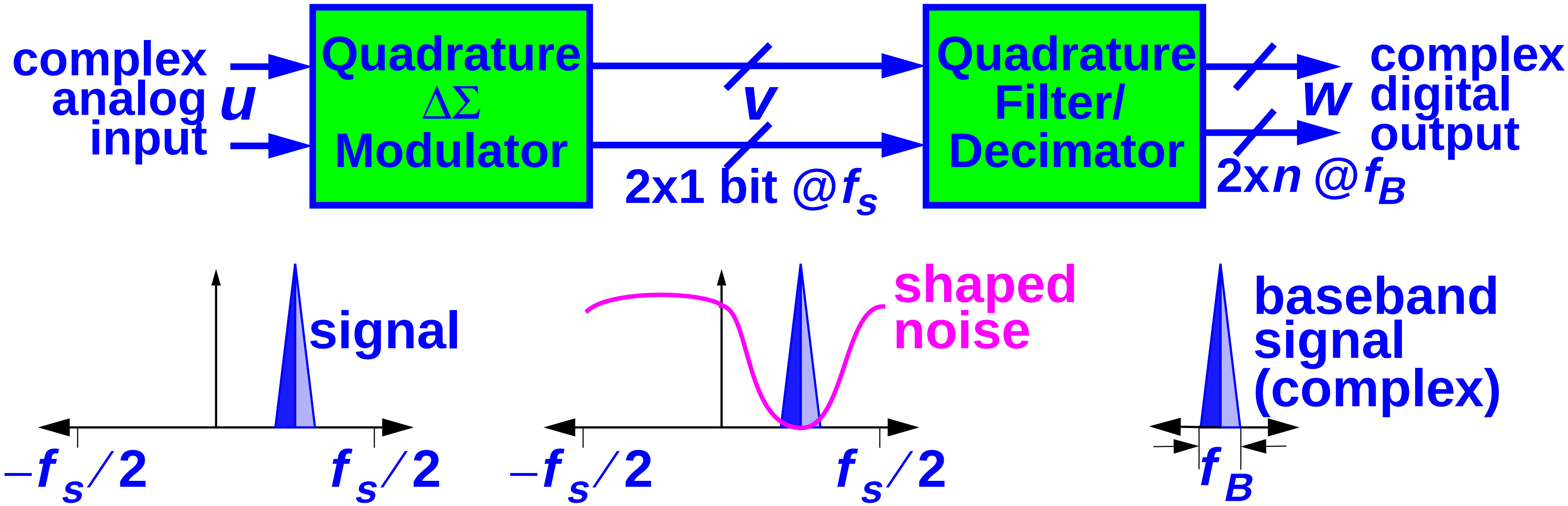


Example In-Band Spectrum

IF = 400 MHz, BW = 75 MHz



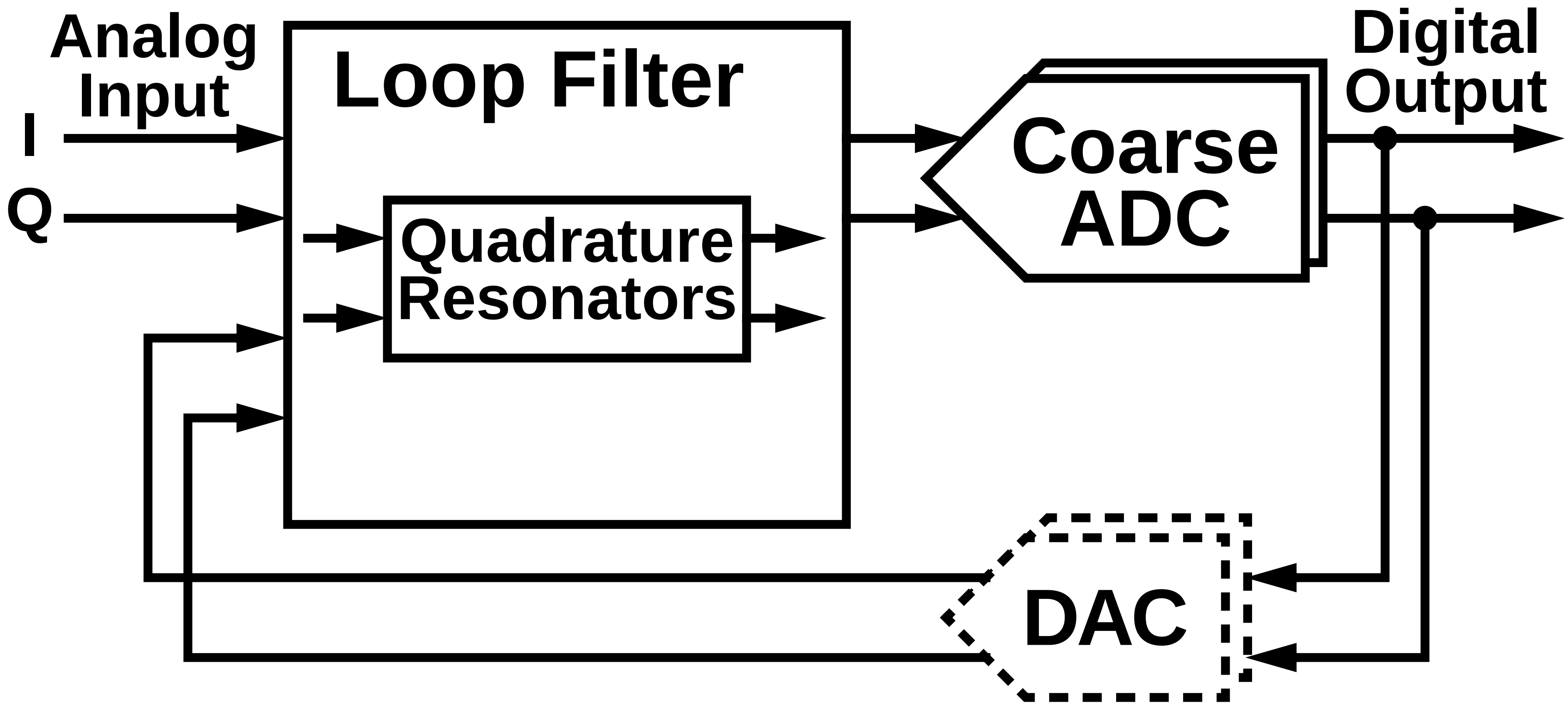
A Quadrature ΔΣ ADC System



$$OSR = f_s / f_B$$

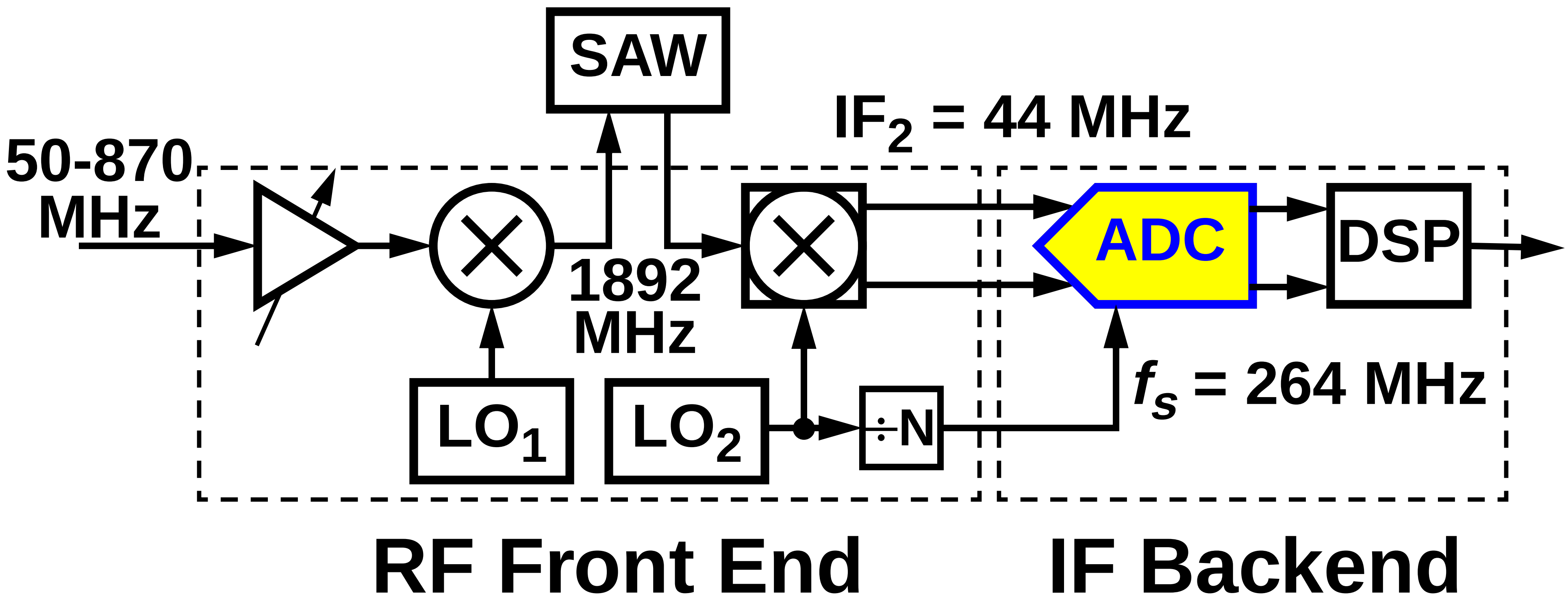
- Modulator converts its quadrature analog input into a pair of bit-stream outputs
- DSP removes out-of-band noise and translates the signal to baseband

A Quadrature ΔΣ Modulator



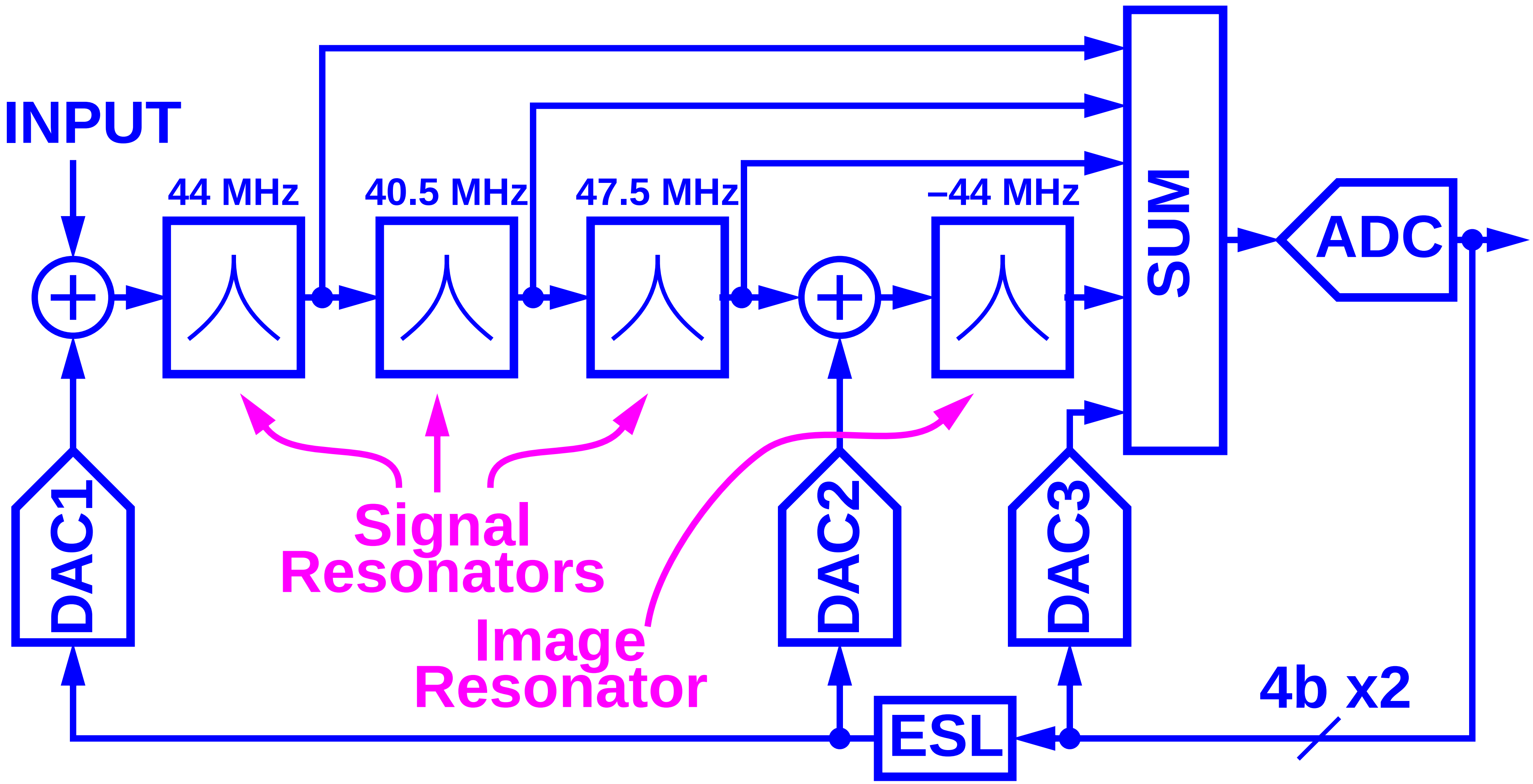
- **A ΔΣ ADC with quadrature everything**
  - **NTF and STF are complex.**

Example: A Quadrature ΔΣ ADC for a TV Tuner System



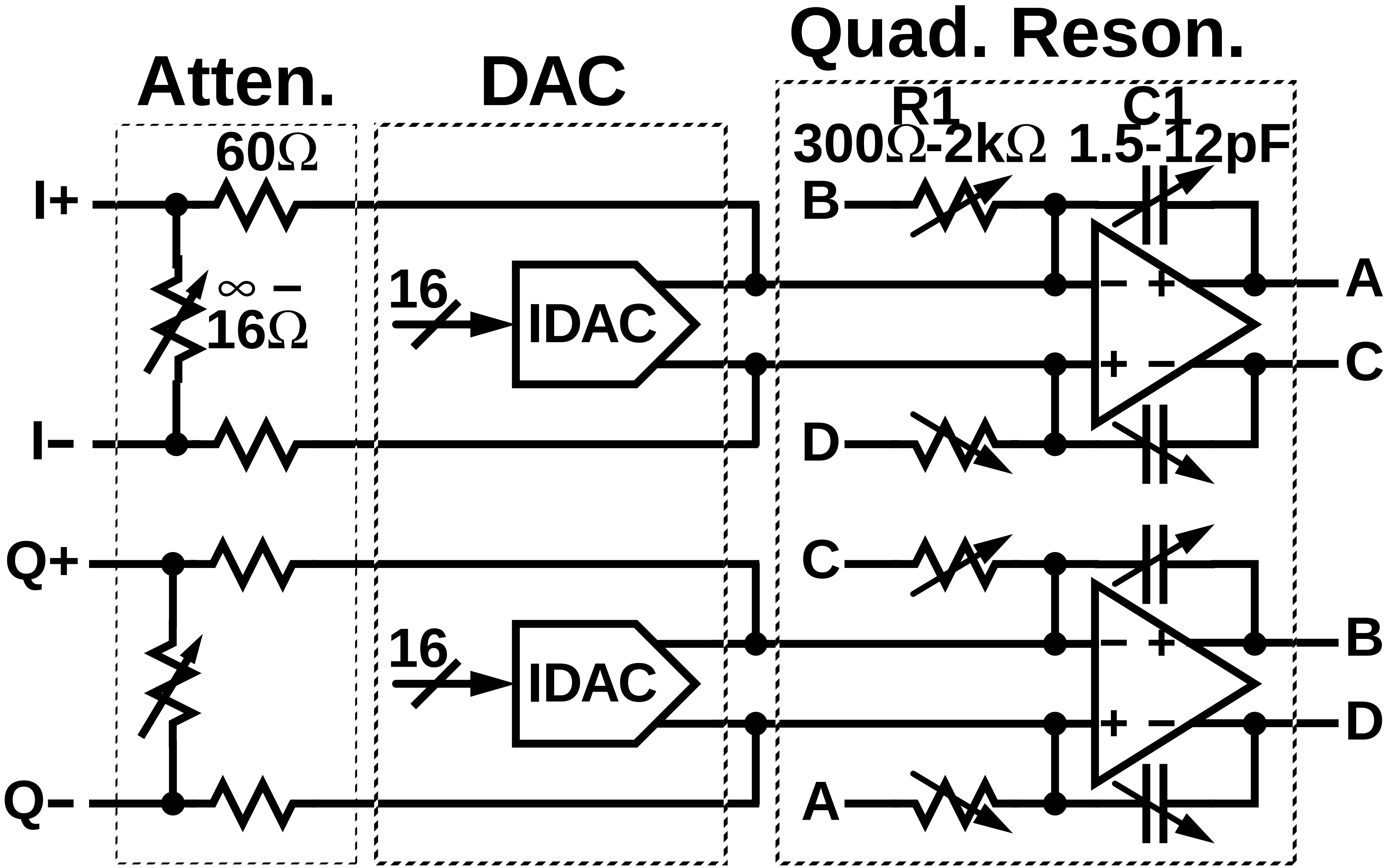
- **Dual-conversion super-heterodyne receiver containing a quadrature bandpass ΔΣ ADC**

ADC Architecture

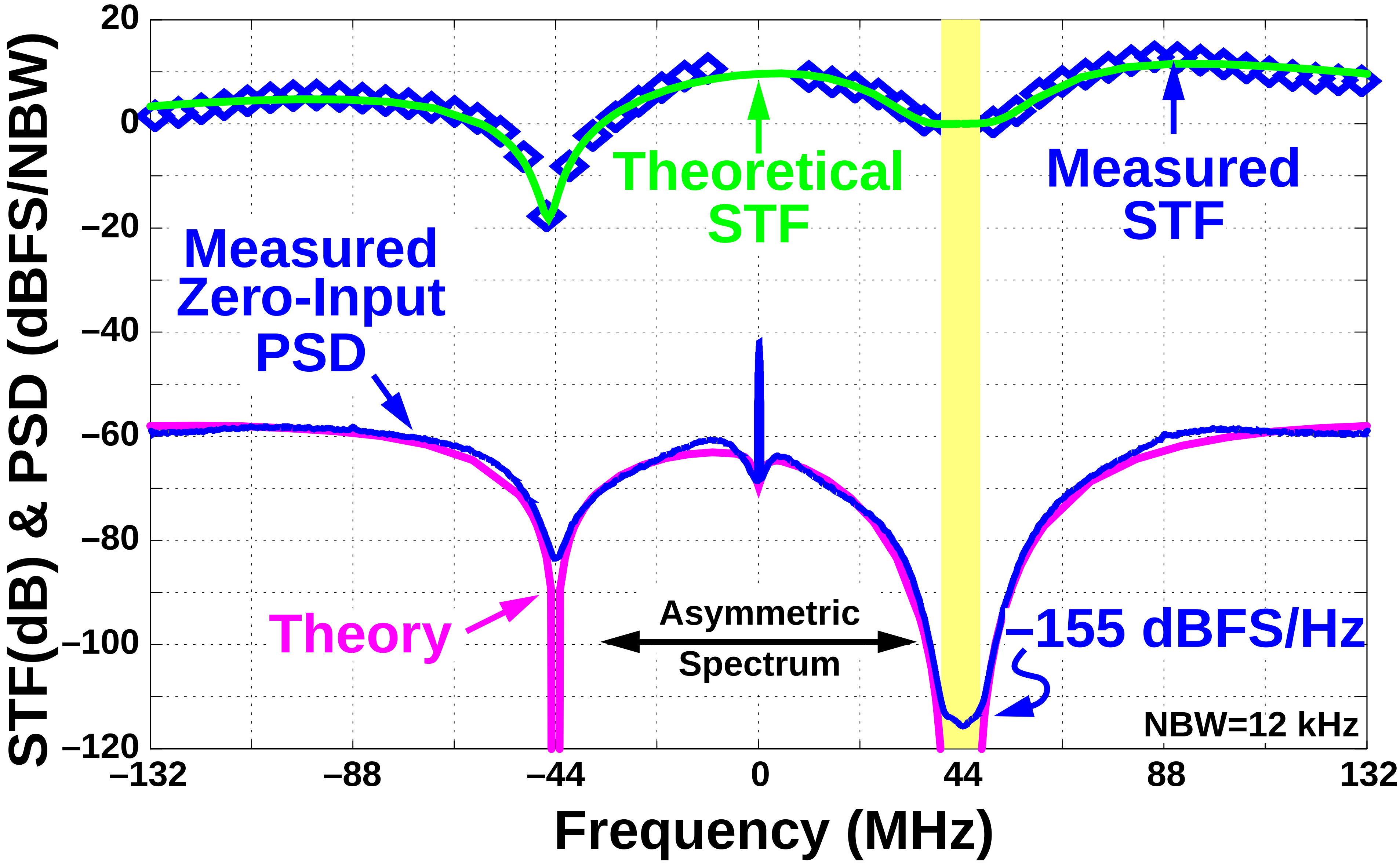


- (3+1)th-order, 4-b, feedforward A-RC modulator

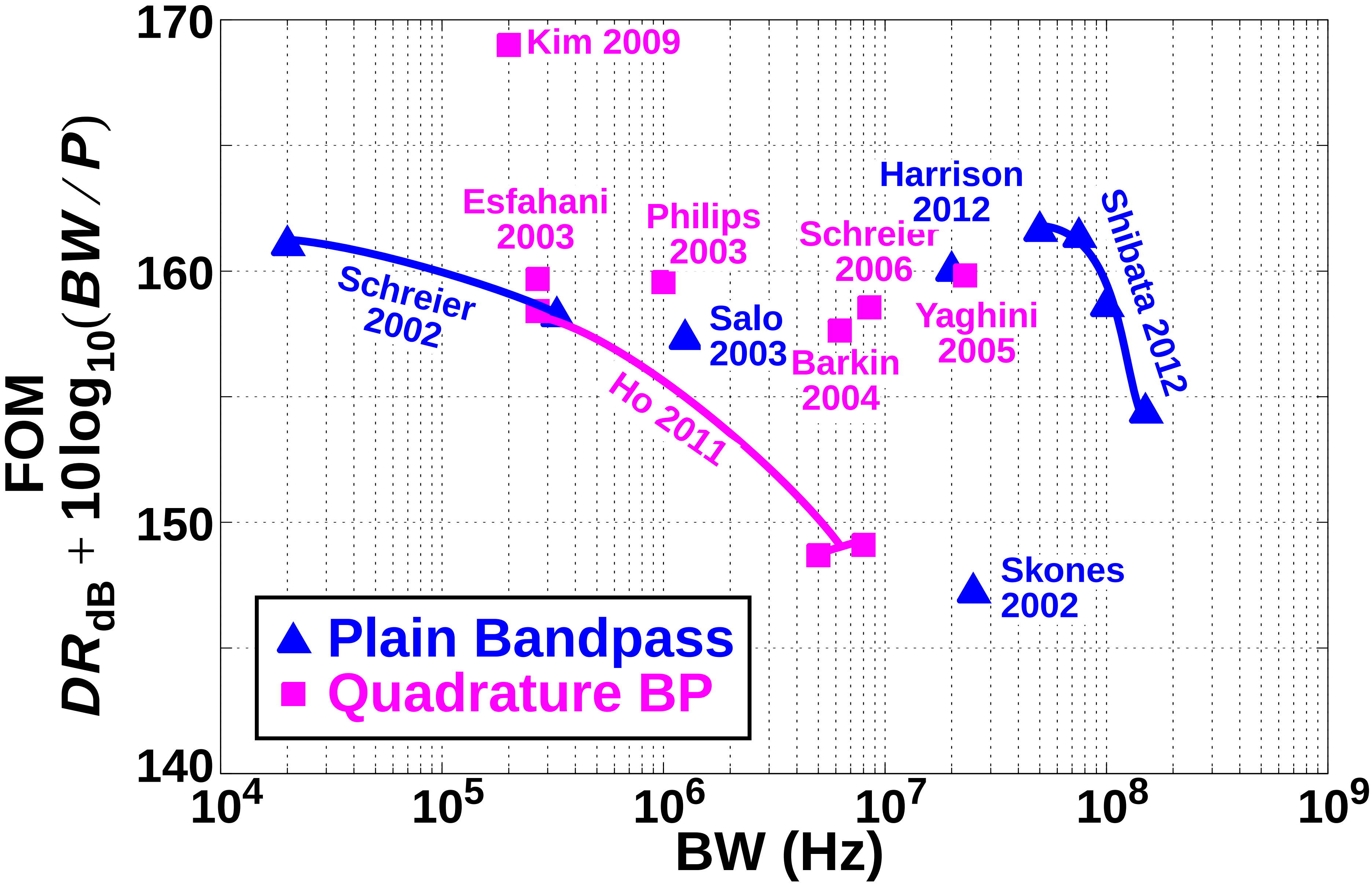
Quadrature ADC Front-End



STF & NTF


20
0
–20
–40
–60
–80
–100
–120
STF(dB) & PSD (dBFS/NBW)
–132
–88
–44
0
44
88
132
Frequency (MHz)
Theoretical STF
Measured STF
Measured Zero-Input PSD
Theory
Asymmetric Spectrum
–155 dBFS/Hz
NBW=12 kHz

FOM Comparison



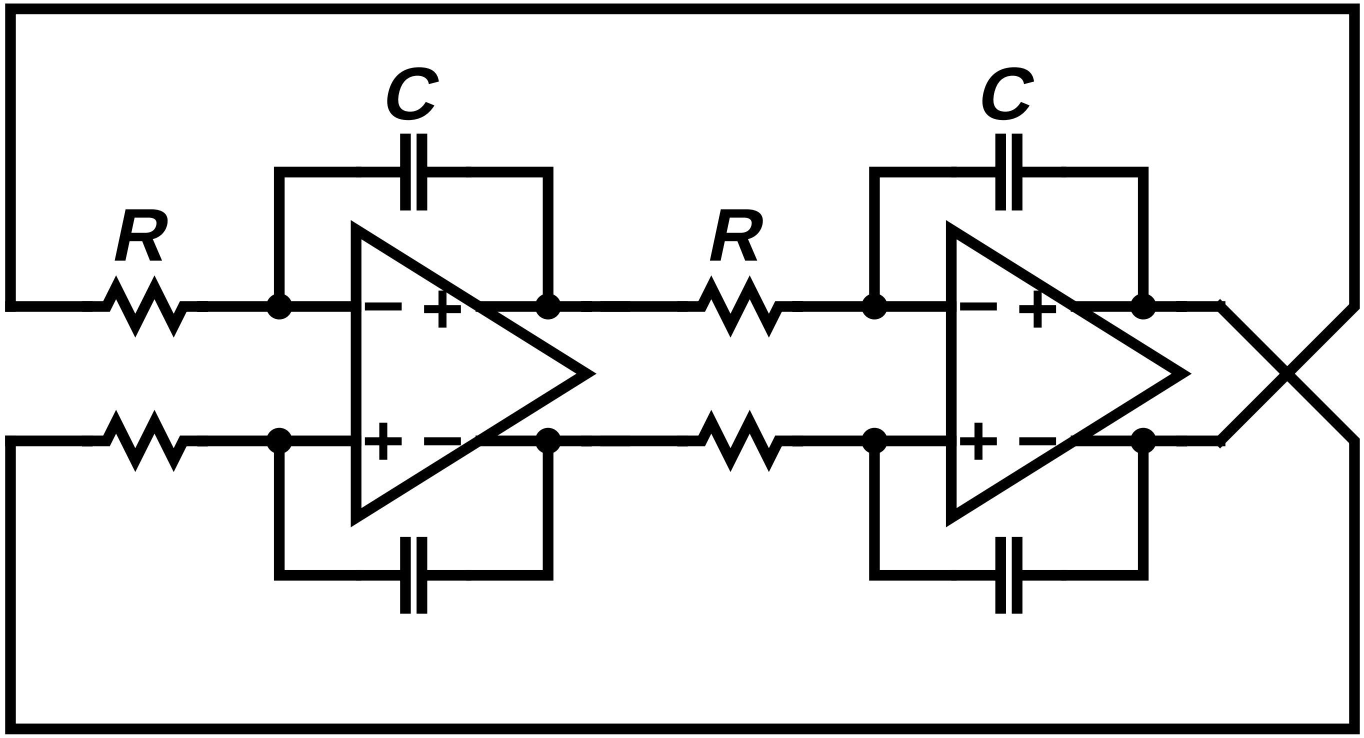
NLCOTD: High-Q Resonator

- **Want $Q \gg \sqrt{3}\dfrac{f_0}{BW}$ for small SQNR degradation**
- **In a TV tuner ADC f_0 = 44 MHz and BW = 8.5 MHz, so we needed Q $\gg$ 9**

  Actual requirement was Q > 20.

  How can Q be kept high despite finite amplifier gain and bandwidth?

Active-RC Resonator Structure



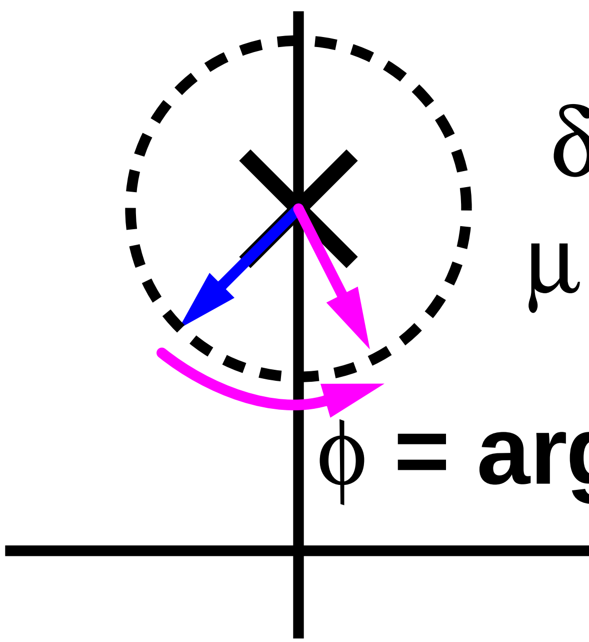
$$f_0 = \frac{1}{2\pi RC}$$

- **Tuned by adding positive feedback to make an oscillator and adjusting C until the desired resonance is achieved**

– **Amplifier drives both R and C $\Rightarrow$ trouble?**

Amplifier Gain and Phase

- **Finite gain degrades Q**
- **Phase lag enhances Q**
- **Analysis shows $\phi = 45^\circ$ yields high Q, regardless of amplifier gain**

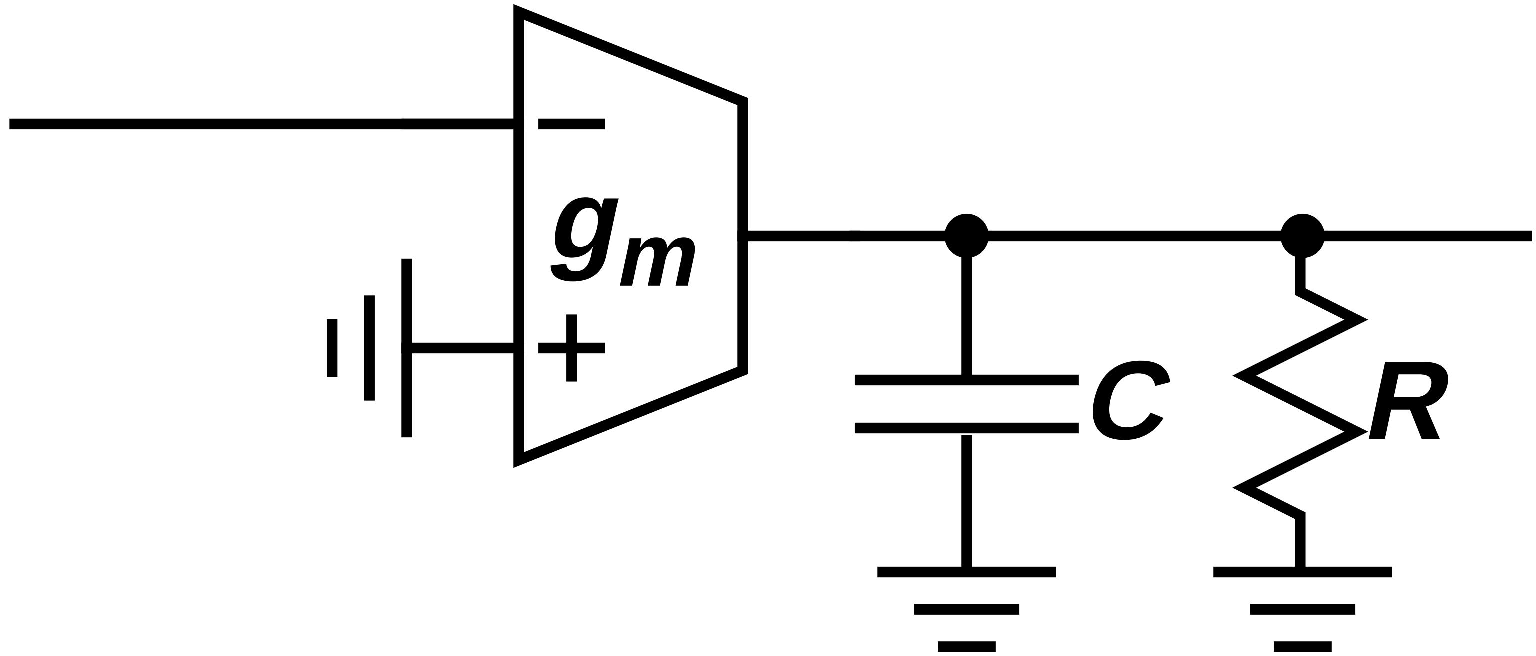


$$\delta = -\mu(\mathbf{1} + \mathbf{j})\omega_0$$

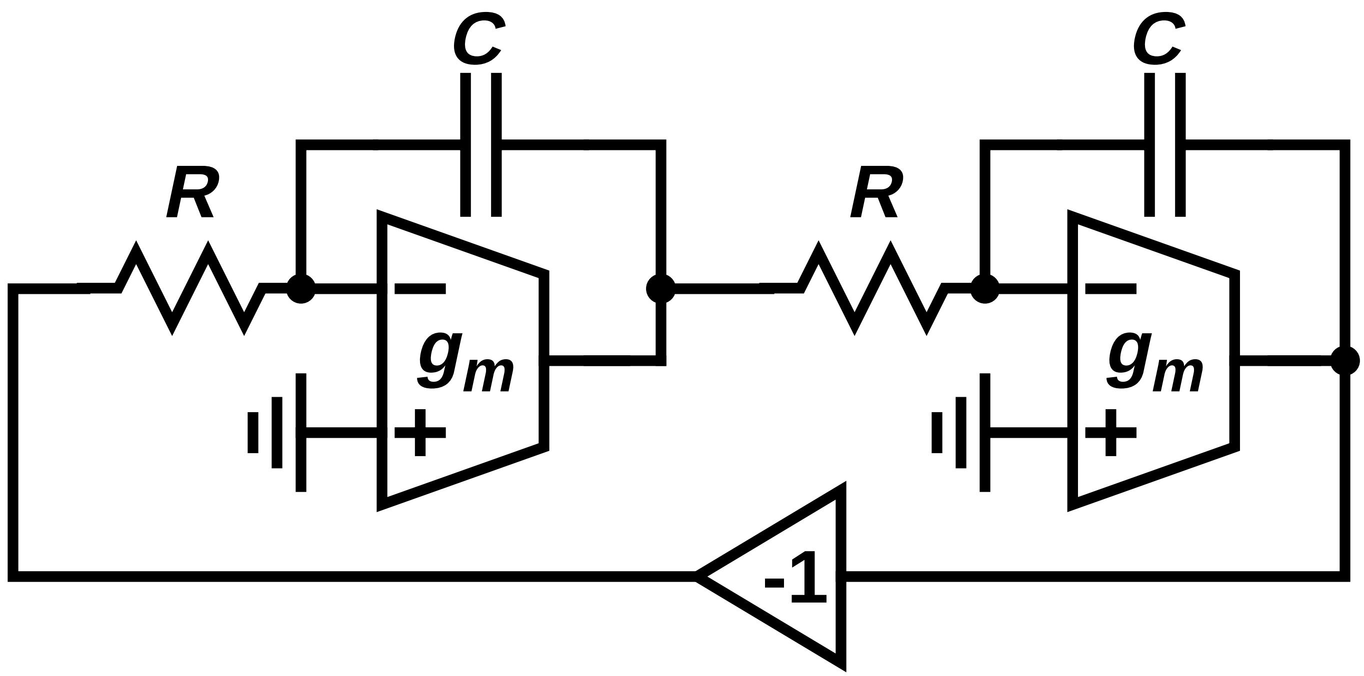
$$\mu = \mathbf{1}/\boldsymbol{A}$$

$$\phi = \mathbf{arg}(\mu)$$

An Amplifier with ϕ = 45° @ f_0:

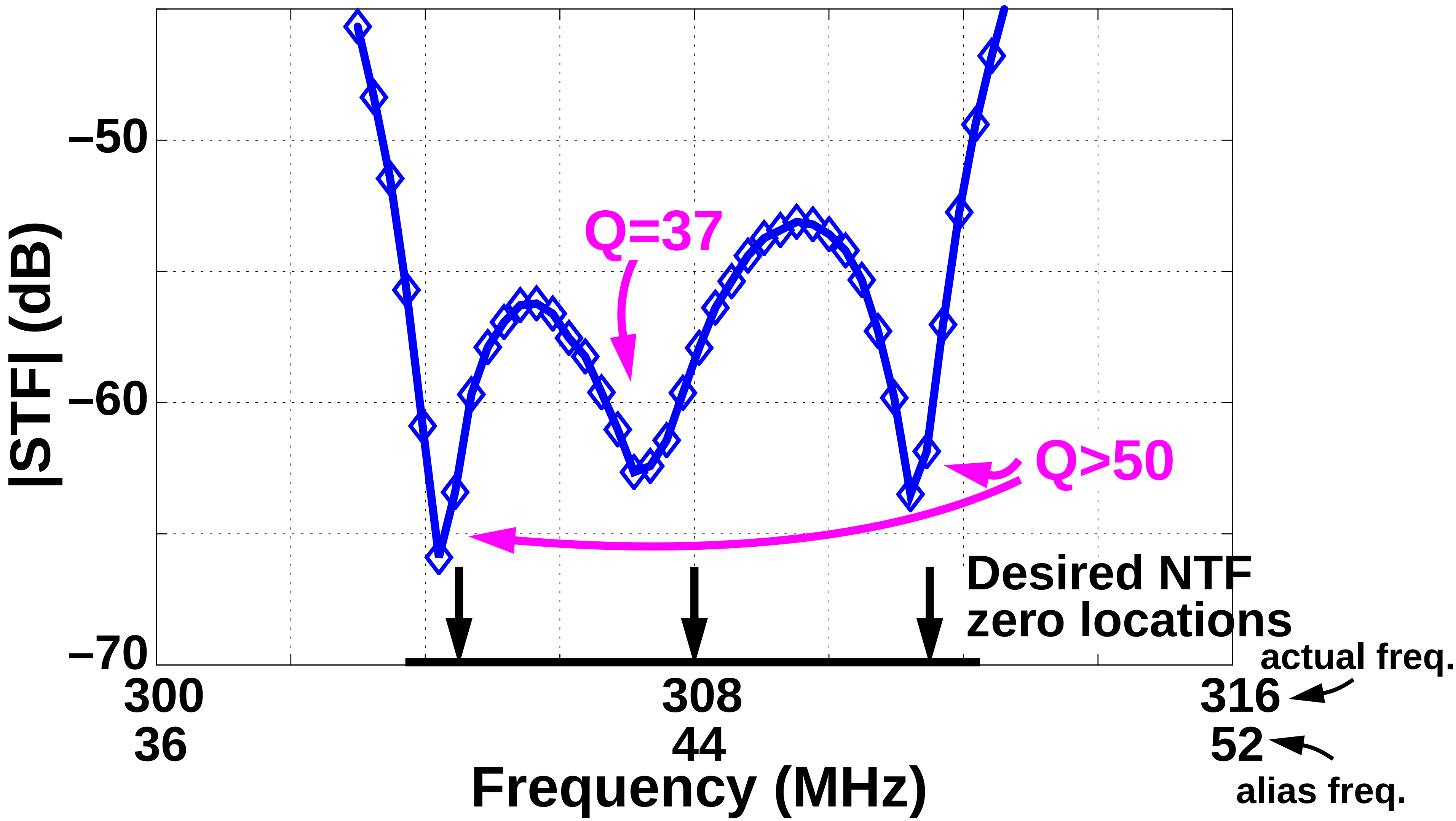


Resulting High-Q Resonator



- Amplifier load yields $\phi = 45^\circ$ @ f_0
- Finite g_m shifts the pole frequency, but does not degrade Q!

Measured STF in an Alias Band



- **Resonator Q is well above the design target!**

What You Learned Today

1. **Feedback vs. Feedforward topology**
2. **State-space (ABCD) representation of the loop filter in the ΔΣ Toolbox**
3. **MASH Modulators**
4. **Continuous-Time Modulators**
5. **Bandpass and Quadrature Bandpass ΔΣ**