ECE1371 Advanced Analog Circuits
Lecture 11

ADVANCED ΔΣ

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

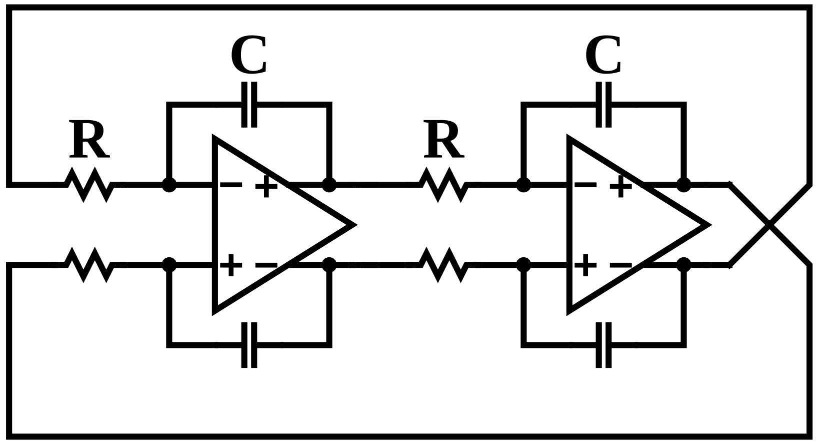
- Want $Q \gg \sqrt{3}\frac{f_0}{BW}$ for small SQNR degradation
- In a TV tuner ADC f_0 = 44 MHz and BW = 8.5 MHz, so we needed Q >> 9

  In actuality the requirement was Q > 20.

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

Date	Lecture			Ref	Homework
2012-01-12	RS	1	Introduction: MOD1	ST 2, A	1: MOD1 in Matlab
2012-01-19	RS	2	MOD2 & MOD*N*	ST 3, 4, B	2: MOD2 in Matlab
2012-01-26	RS	3	Example Design: Part 1	ST 9.1, CCJM 14	3: Sw.-level MOD2
2012-02-02	TC	4	SC Circuits	R 12, CCJM 14	4: SC Integrator
2012-02-09	TC	5	Amplifier Design		
2012-02-16	TC	6	Amplifier Design		5: SC Int w/ Amp
2012-02-23	Reading Week + ISSCC– No Lecture				
2012-03-01	RS	7	Example Design: Part 2	CCJM 18	Start Project
2012-03-08	RS	8	Comparator & Flash ADC	CCJM 10	
2012-03-15	TC	9	Noise in SC Circuits	ST C	
2012-03-22	TC	10	Matching & MM-Shaping	ST 6.3-6.5, +	
2012-03-29	RS	11	Advanced ΔΣ	ST 6.6, 9.4	
2012-04-05	TC	12	Pipeline and SAR ADCs	CCJM 15, 17	
2012-04-12	No Lecture				
2012-04-19	Project Presentation				

Active-RC Resonator Structure



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

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

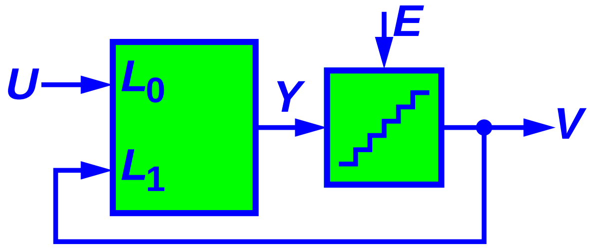
  No Q-tuning.

– **Amplifier drives both R and C ⇒ Q trouble?**

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

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

Review:
Generic Single-Loop ΔΣ ADC



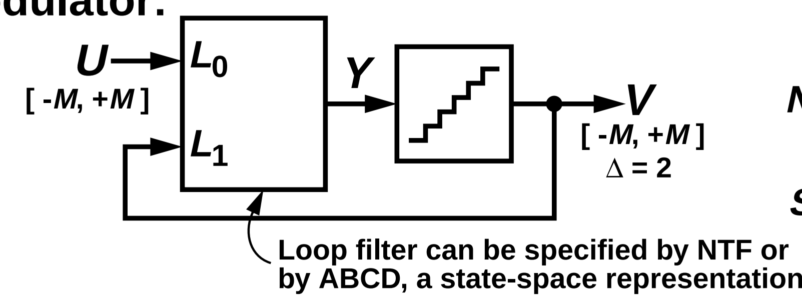
$$Y = L_0 U + L_1 V$$
$$V = Y + E$$
$$\Rightarrow \boxed{V = STF \cdot U + NTF \cdot E}\text{, where}$$

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

Inverse Relations:
$L_1 = 1 - 1/NTF$, $L_0 = STF / NTF$

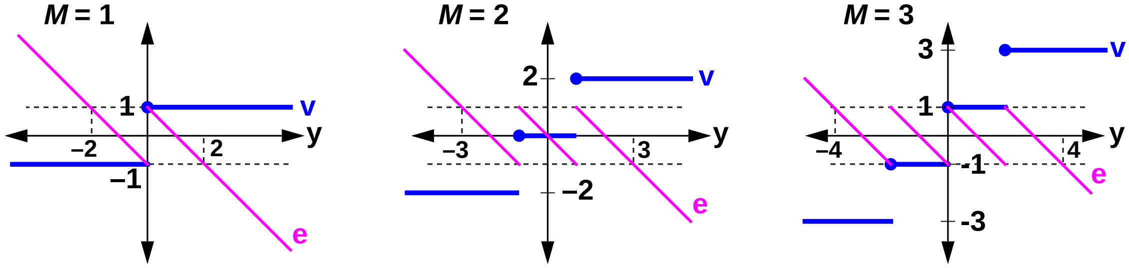
Review: ΔΣ Toolbox Model

Modulator:



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

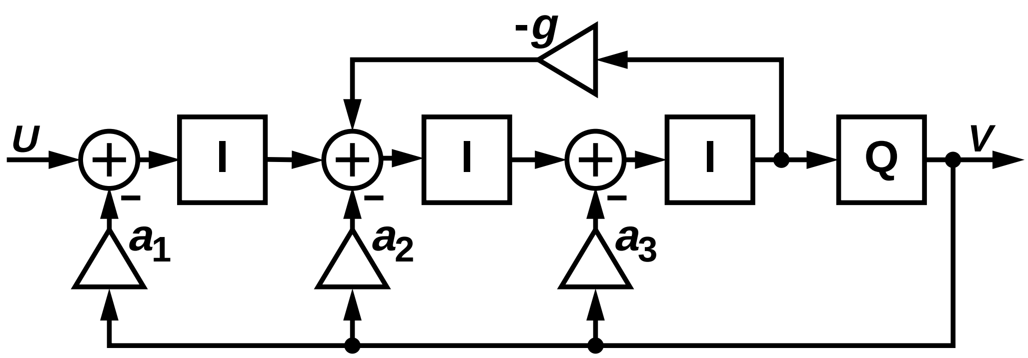
Quantizer:



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

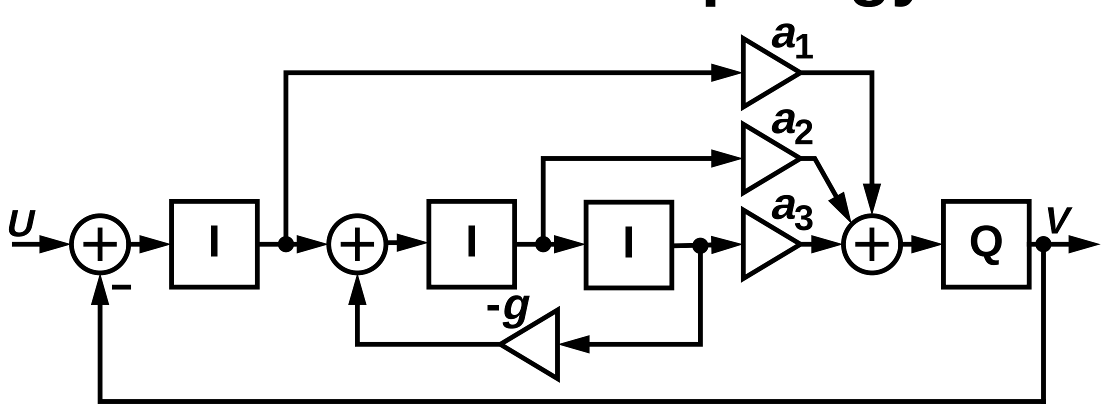
Mid-rise quantizer;
v: odd integers [–*M*,+*M*]

Feedback Topology



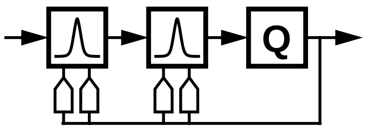
- ***N*** **integrators precede the quantizer**
- **Feedback from the quantizer to the input of each integrator (via a DAC)**
- **Local feedback around pairs of integrators to set the NTF's zeros**
- **Multiple input feed-in branches are possible**

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



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

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

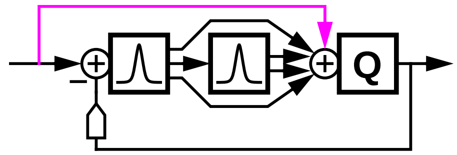
poles of LF are zeros of NTF

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

same poles as NTF

zeros = zeros of L_0

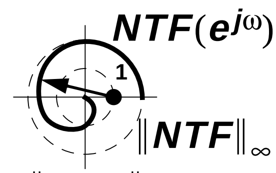
STF often has no zeros, only poles.



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

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

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

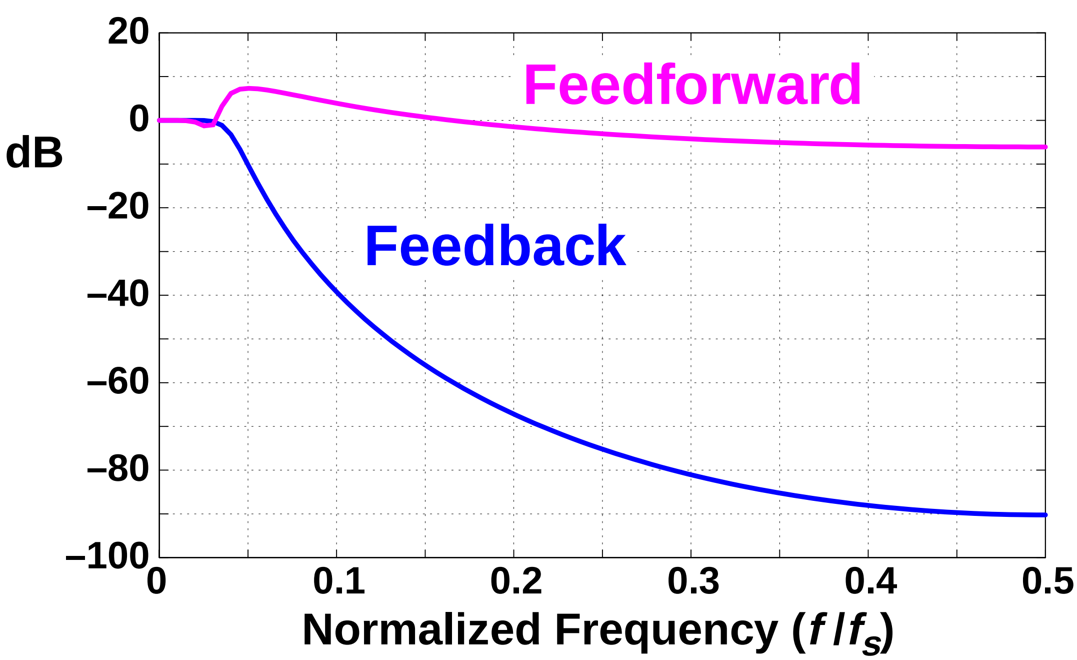


$$\|STF\|_\infty \approx \|NTF\|_\infty + 1$$

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

STF Comparison

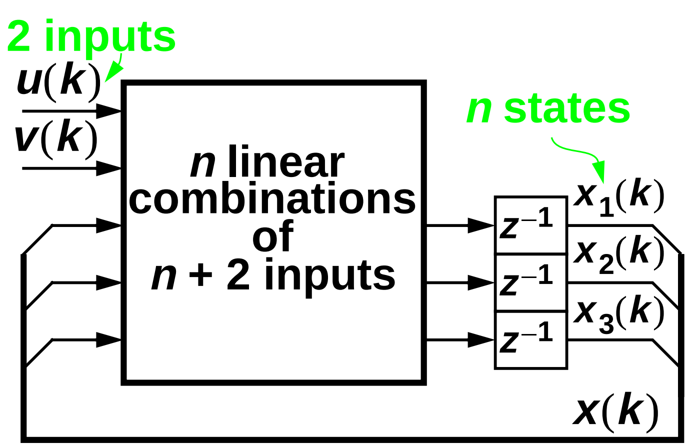
5^{th}-Order; Single Feed-In



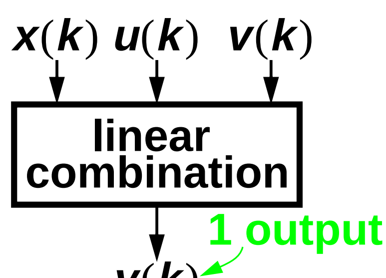
Feedforward vs. Feedback

- **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 timing can be tricky**
  - **Need to quantize *u* and feed it back in zero time.**

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



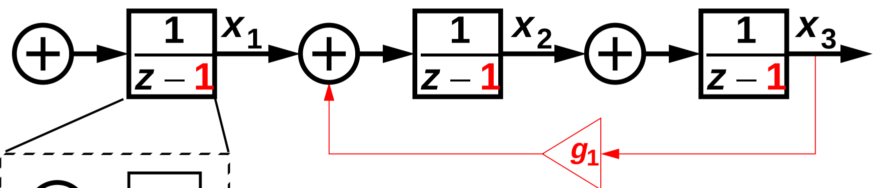
$$x(k+1) = \underset{n \times n}{A}x(k) + \underset{n \times 2}{B}\begin{bmatrix} u(k) \\ v(k) \end{bmatrix}$$

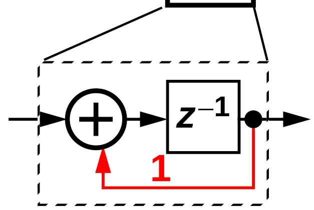


$$y(k) = \underset{1 \times n}{C}x(k) + \underset{1 \times 2}{D}\begin{bmatrix} u(k) \\ v(k) \end{bmatrix}$$

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

Ex.: Cascade of Integrators Feedback (CIFB) Topology

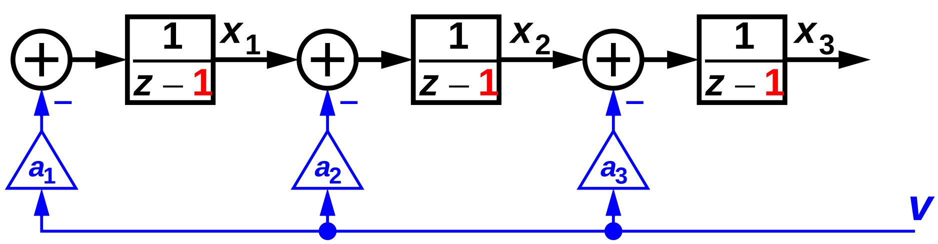




Recall: NTF zeros = Loop Filter's poles.
Loop Filter poles determined by A matrix.

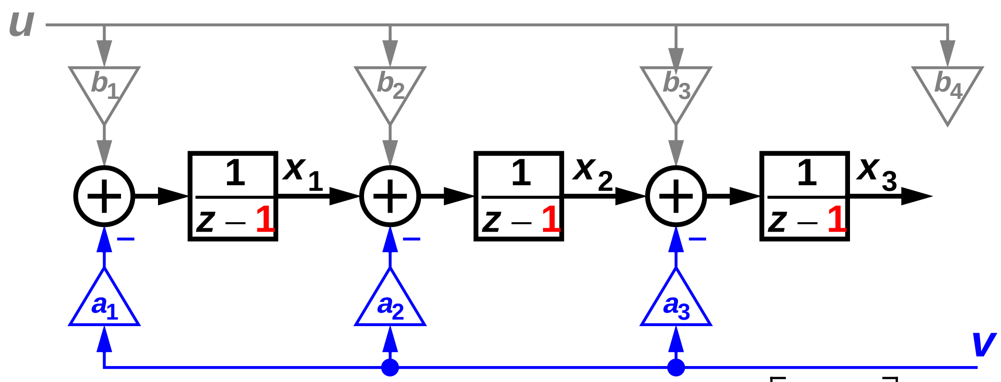
$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \\ y(k) \end{bmatrix} = \left[\begin{array}{ccc|cc} 1 & 0 & 0 & - & - \\ 1 & 1 & g_1 & - & - \\ 0 & 1 & 1 & - & - \\ \hline 0 & 0 & 1 & - & - \end{array}\right] \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ u(k) \\ v(k) \end{bmatrix}$$

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



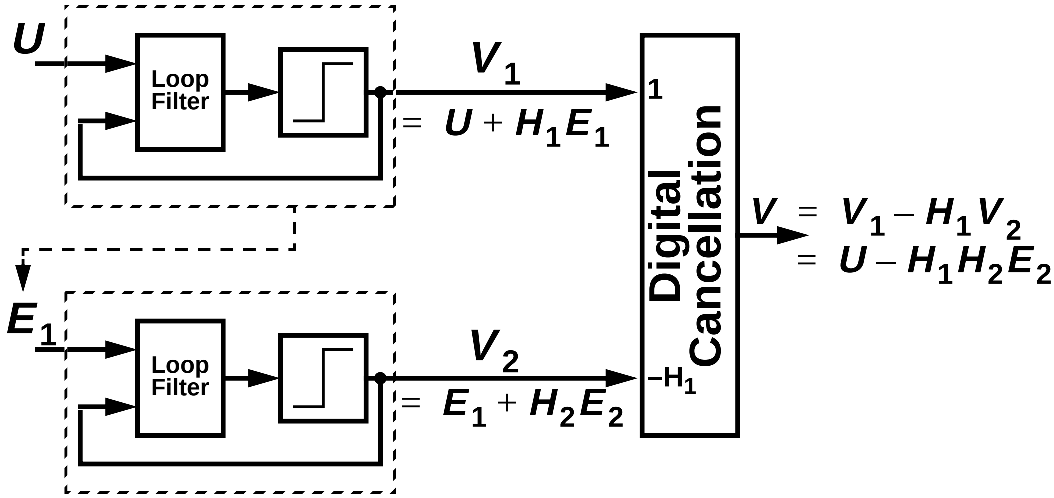
$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \\ y(k) \end{bmatrix} = \left[\begin{array}{ccc|cc} 1 & 0 & 0 & b_1 & -a_1 \\ 1 & 1 & 0 & b_2 & -a_2 \\ 0 & 1 & 1 & b_3 & -a_3 \\ \hline 0 & 0 & 1 & b_4 & 0 \end{array}\right] \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \\ u(k) \\ v(k) \end{bmatrix}$$

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

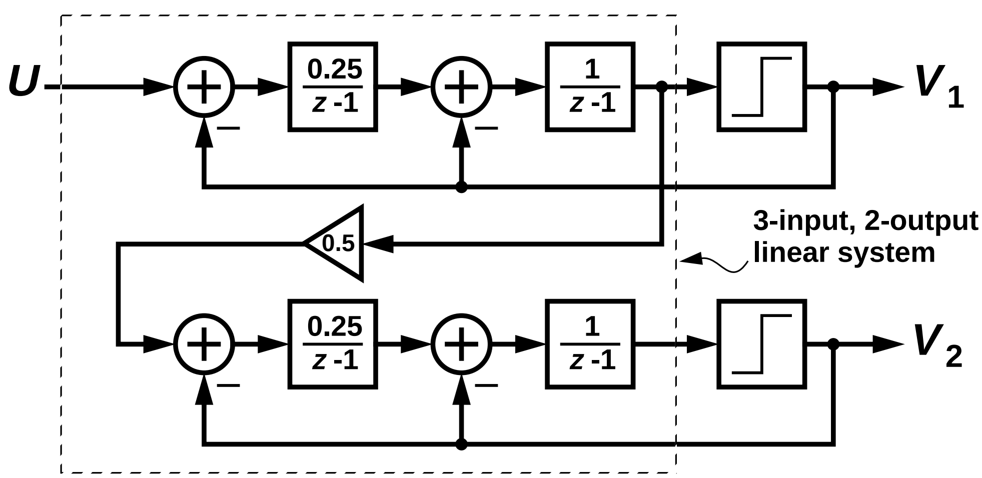
- **Put two (or more) modulators in "series"**



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

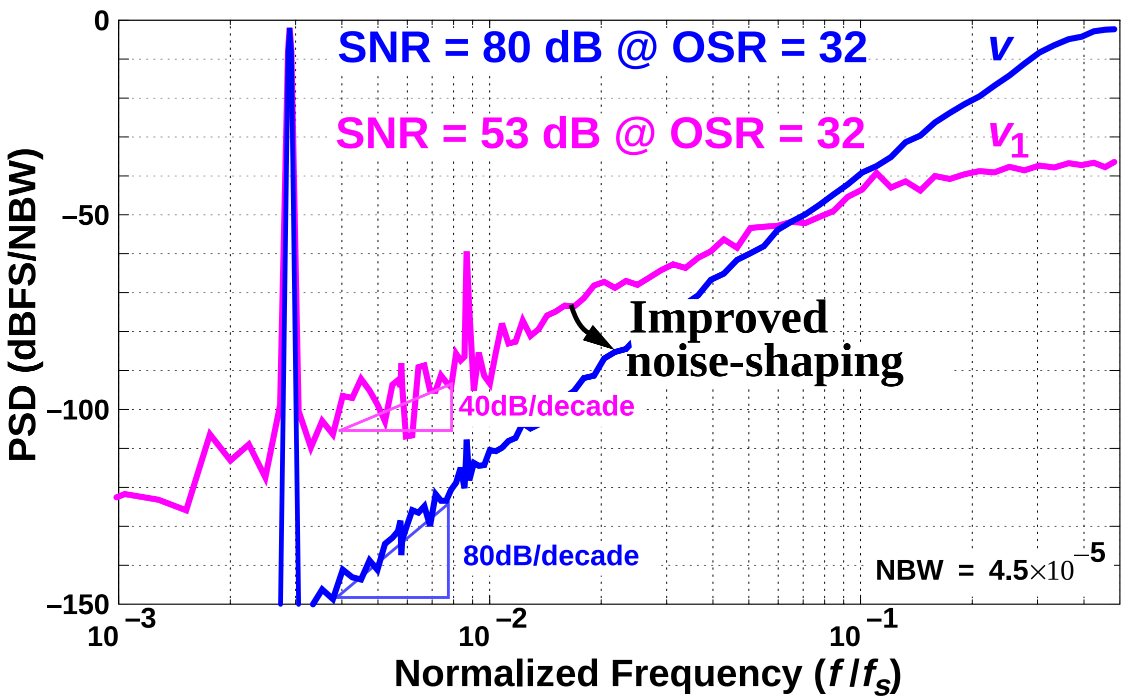
Example: 2-2 Cascade

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

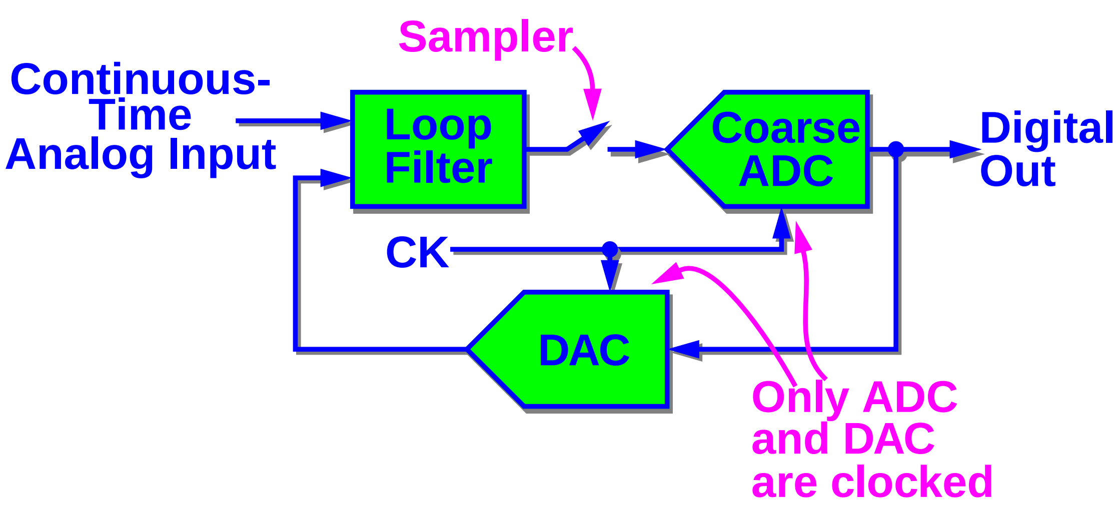


$$\left(V = \frac{1}{z^3}V_1 + \frac{8(z-1)^2(z-0.5)^2}{z^3(z-0.75)}V_2\right) \Rightarrow \left(H(z) = \frac{8(z-1)^4}{(z-0.75)}\right)$$

Example MASH Spectra



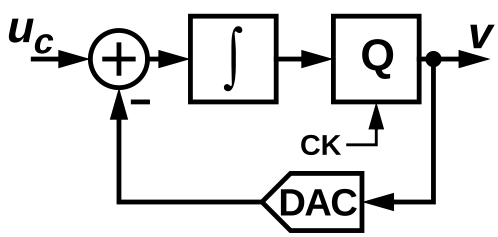
A Continuous-Time ΔΣ ADC



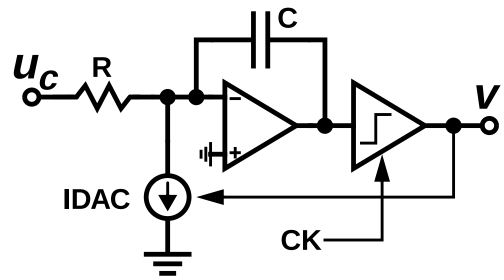
- **Loop filter implemented with continuous-time circuitry**
- **Sampling occurs after the loop filter**

Example: MOD1-CT

Block Diagram



Schematic

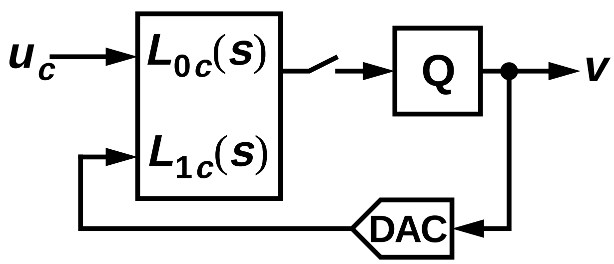


- **Note: Input is a simple resistor,** ***not*** **a switched capacitor**

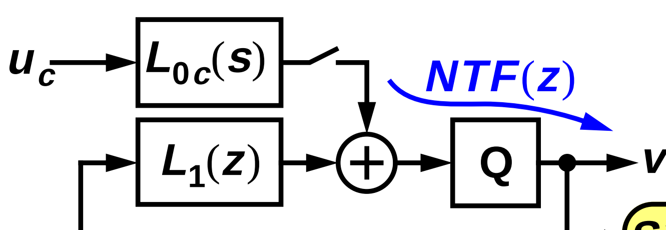
  CT ADCs are easier to drive than DT ADCs.

Inherent Anti-Aliasing

- **ΔΣ ADC with CT Loop Filter**



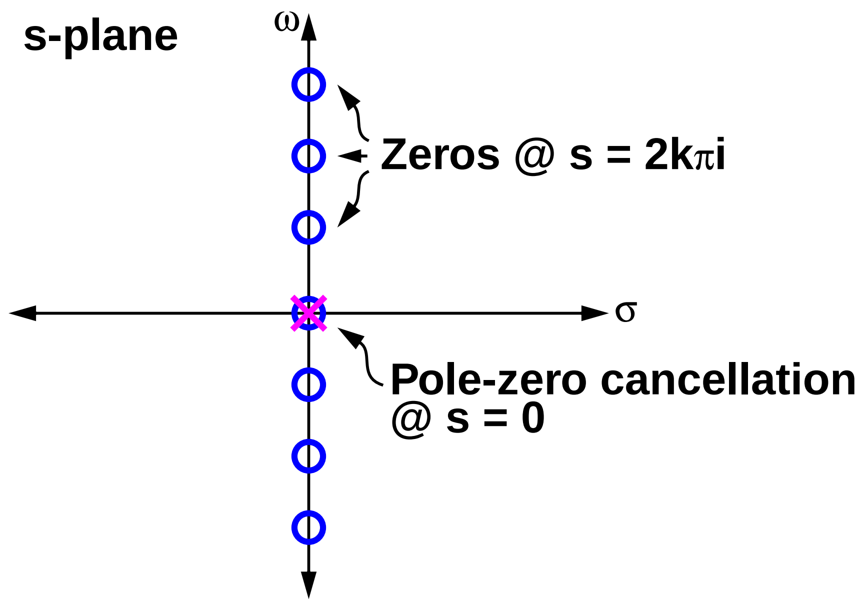
- **Equivalent system with DT feedback path**



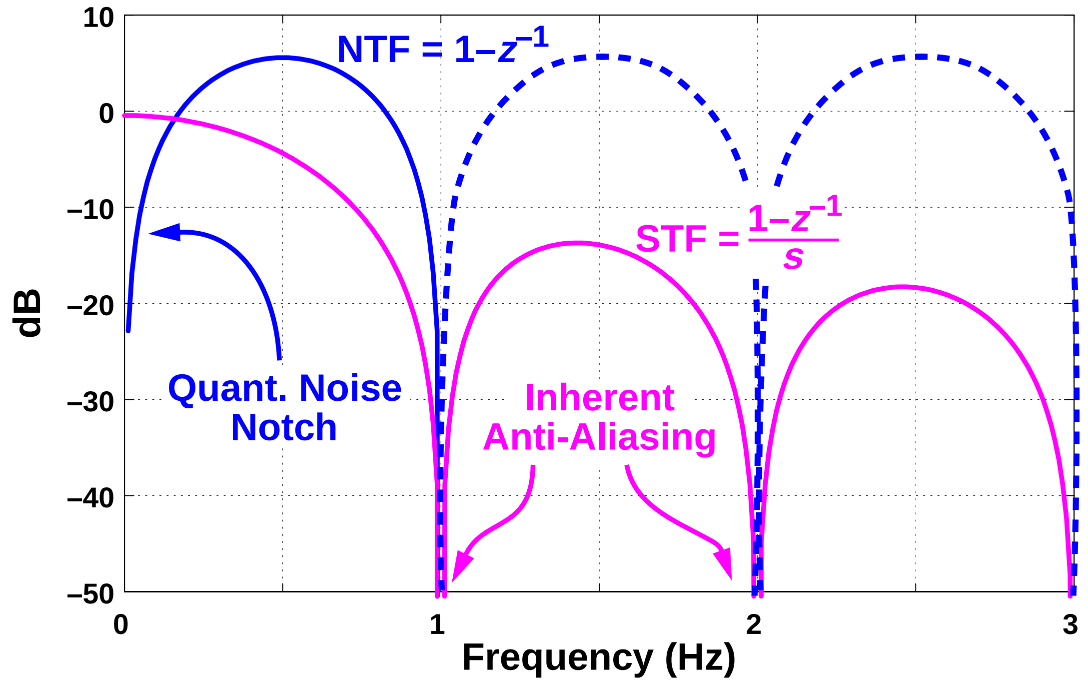
$\Rightarrow$ $STF = L_{0c}(s) NTF(z)$

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

Recall $z = e^s$



MOD1-CT Frequency Responses



Inherent AAF Summary

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

- **STF contains the zeros of the NTF**

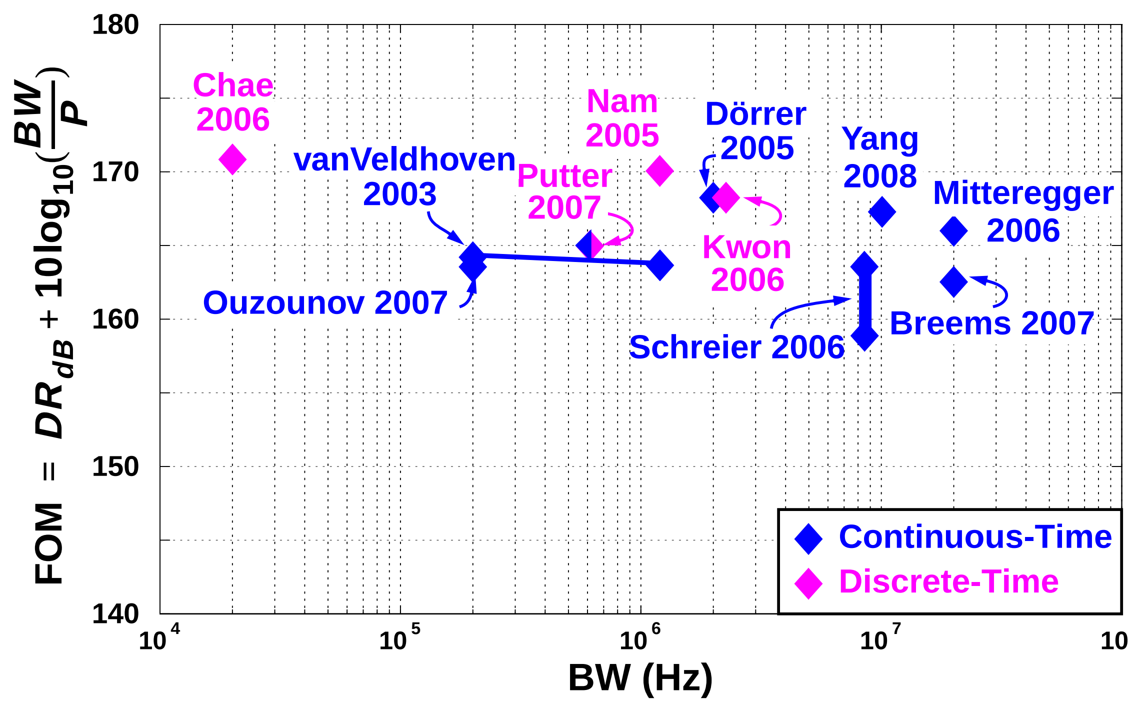
- **Any frequency which aliases to the passband is attenuated by at least as much as the quantization noise**

  Anti-alias performance tracks modulator order.
  Also true for MASH systems.

- **The effective anti-alias filter is**

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

DT ΔΣ vs. CT ΔΣ



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)
200k	82	1.4	164	Ouzounov, ISSCC 2007:13.3	ΔΣ A-RC 5(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)
20M	77	56	164	Breems, ISSCC 2007:13.1	QΔΣ A-RC 2(15)-2(15)

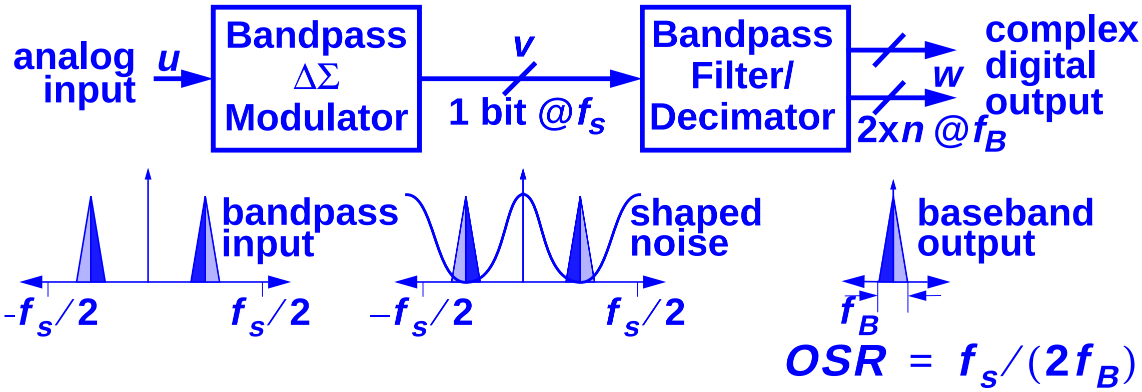
Advantages of Discrete-Time

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

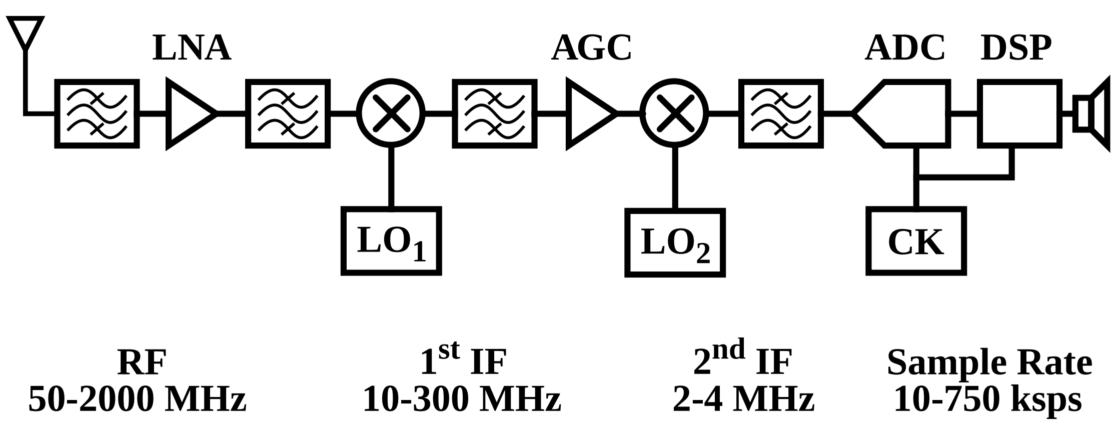
1. Higher speed
2. Inherent anti-aliasing
3. Easier to drive (well-defined Z_{in})
4. Sampling is non-critical
5. Lower power (?)

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**

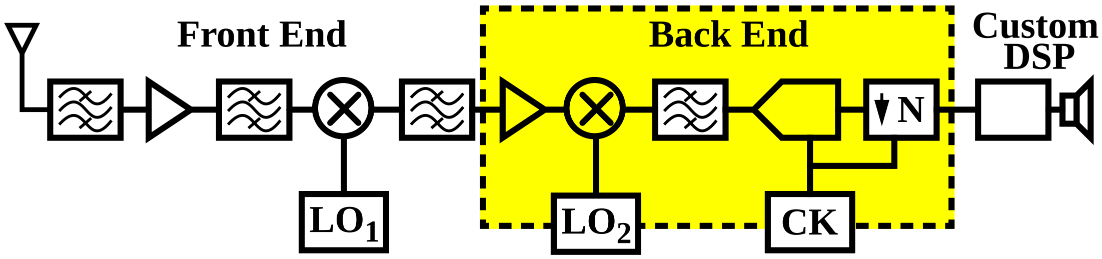
A Dual-Conversion Superheterodyne Receiver



- **A bandpass ADC fits naturally into this narrowband system**

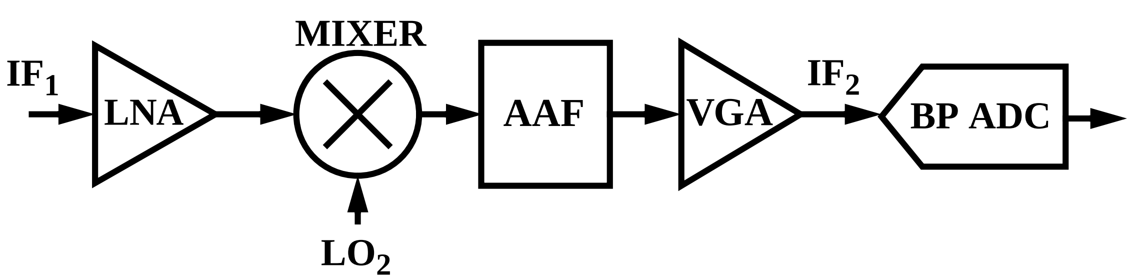
  Perfect I/Q, high dynamic range. Low power?

System Partitioning



- **Goal: a general-purpose, high-performance, low-power back-end**

Traditional Implementation

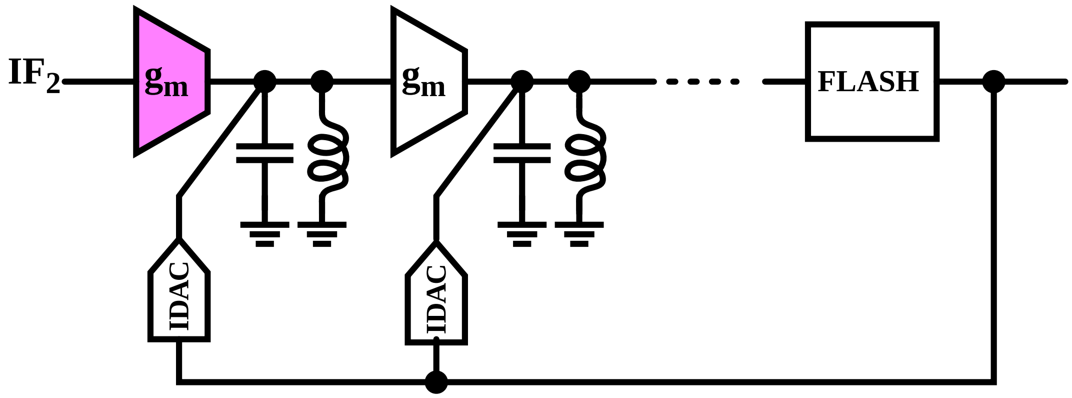


☹ **Numerous high-dynamic range blocks**

☹ **Noise and power budgets are very tight**

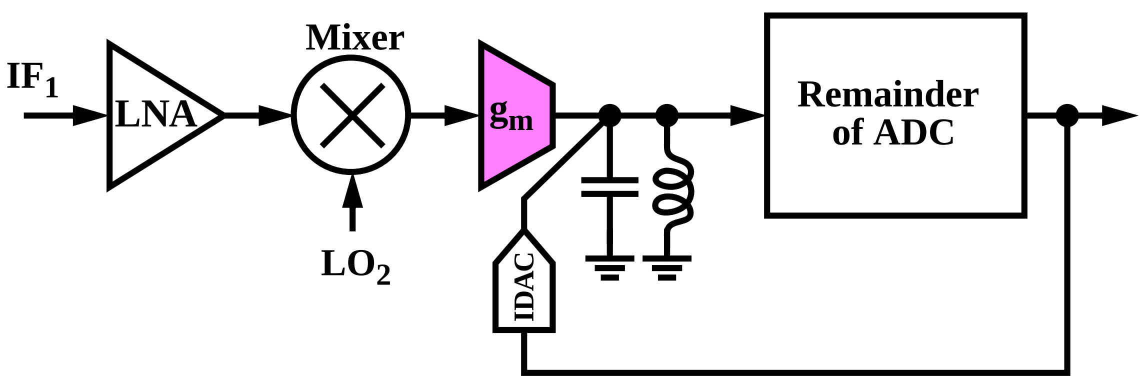
☹ **Large VGA range needed**

Eliminating the AAF with a Continuous-Time BP ΣΔ ADC



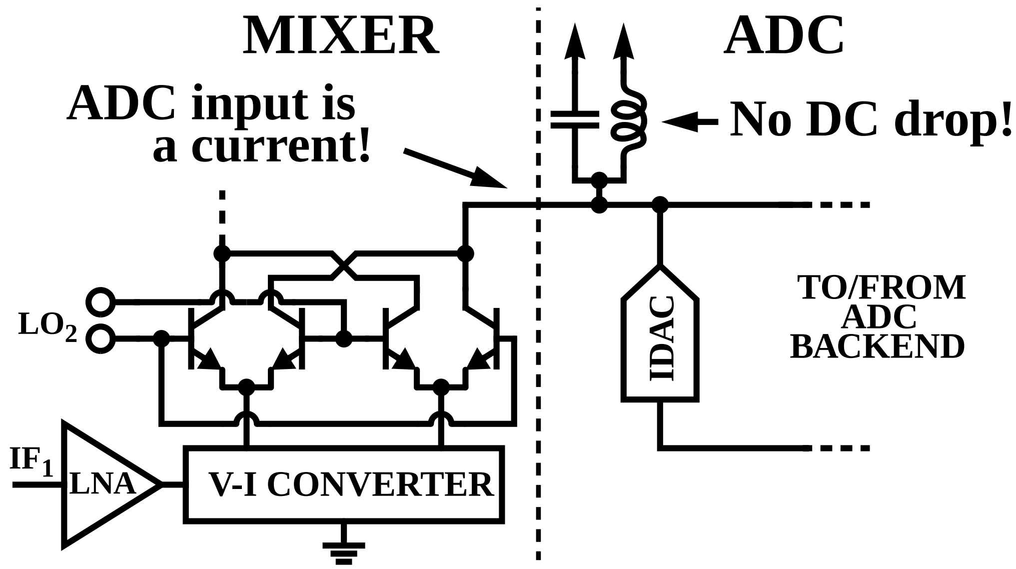
- Anti-alias filtering is inherent
- But still need a low-noise, linear V-I converter

Eliminating the Input g_m



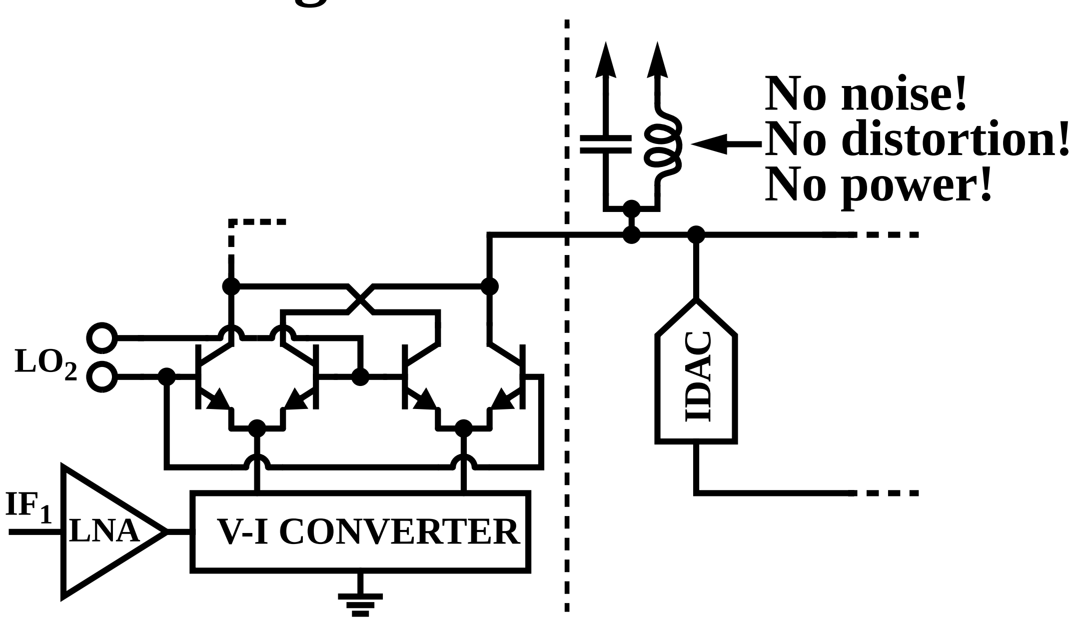
- The output of the mixer is available in current form, so …

Merge ADC with Mixer!



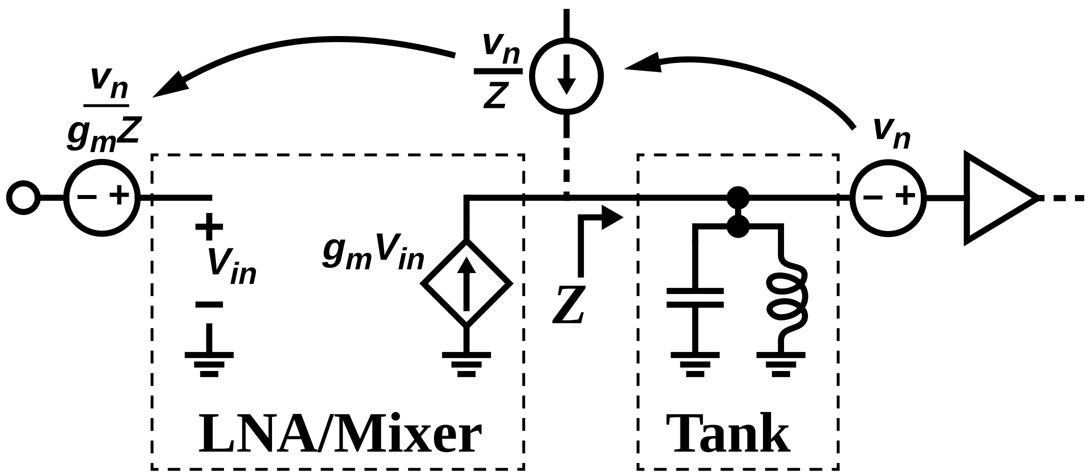
- **Eliminates redundant I-V & V-I conversion**
- **Gives mixer and IDAC more headroom**

Merge ADC with Mixer!



LC tank effectively adds gain, without adding noise, adding distortion or consuming power

Noise Analysis



- **Noise in the ADC backend is attenuated by g_m times the tank impedance**
  In this work, $g_m \approx 10$ mA/V

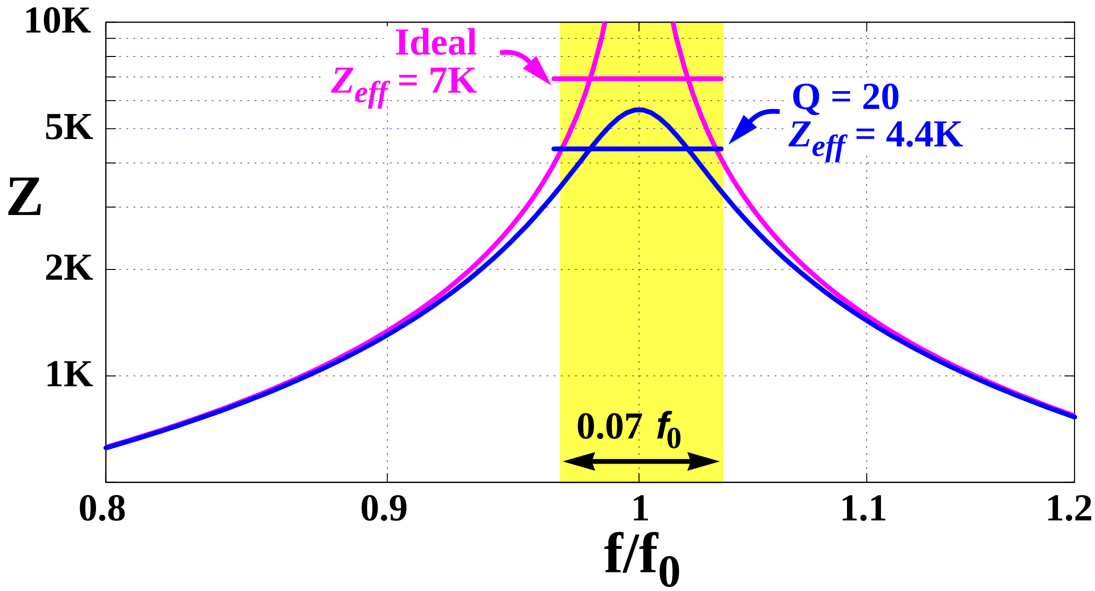
Z_{eff}

- **Near resonance, $|Z_L| = |Z_C|$**
  $|Z_{L,C}| \approx 300\Omega$ in this design
- **At resonance, $|Z| \approx Q \bullet |Z_{L,C}|$**
  About 6kΩ for $Q = 20 \Rightarrow g_m Z \approx 60$.
- **More generally, the effective tank impedance is found by integrating the input-referred noise over the band of interest:**

$$\int \left(\frac{v_n}{g_m Z(\omega)}\right)^2 d\omega = \left(\frac{v_n}{g_m}\right)^2 \int Y(\omega)^2 d\omega = \left(\frac{v_n}{g_m Z_{eff}}\right)^2 \Delta\omega$$

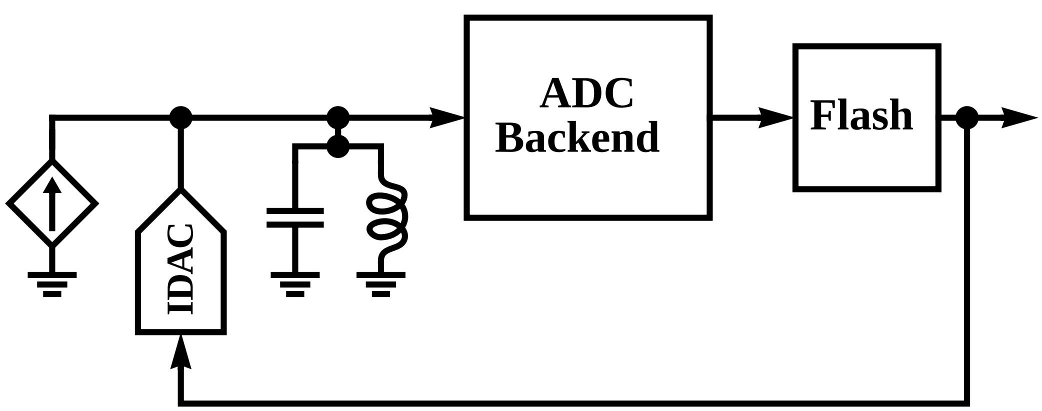
$$\Rightarrow Z_{eff} = (Y_{rms})^{-1}$$

Z vs. Frequency



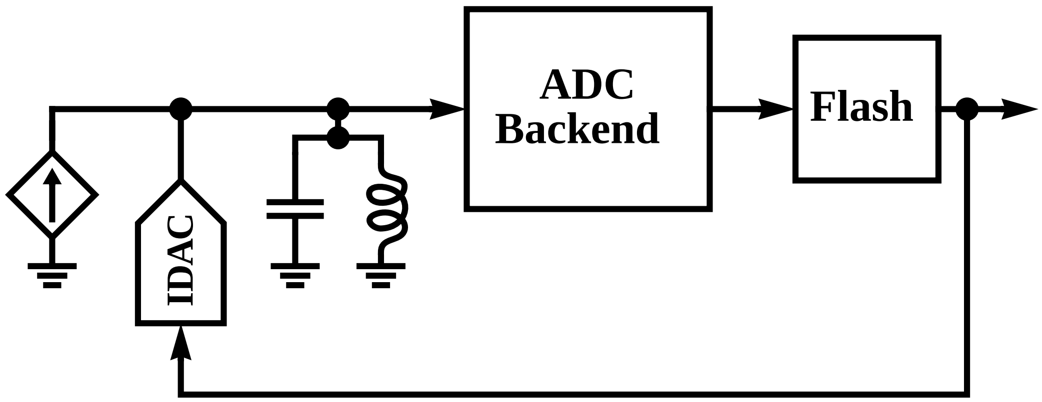
- Q = 20 reduces Z_{eff} by about 4 dB

Remainder of ADC?



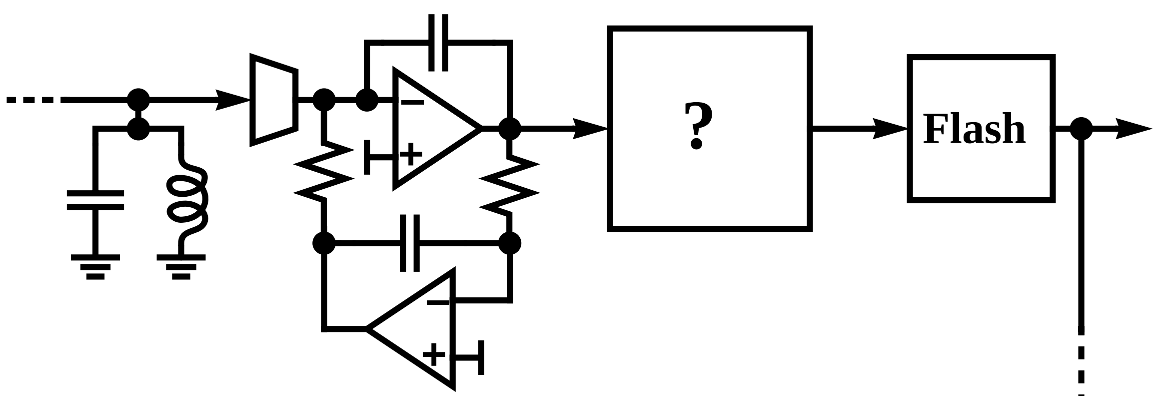
- Add resonator stages until the quantization noise of the flash is low enough

Second Resonator?



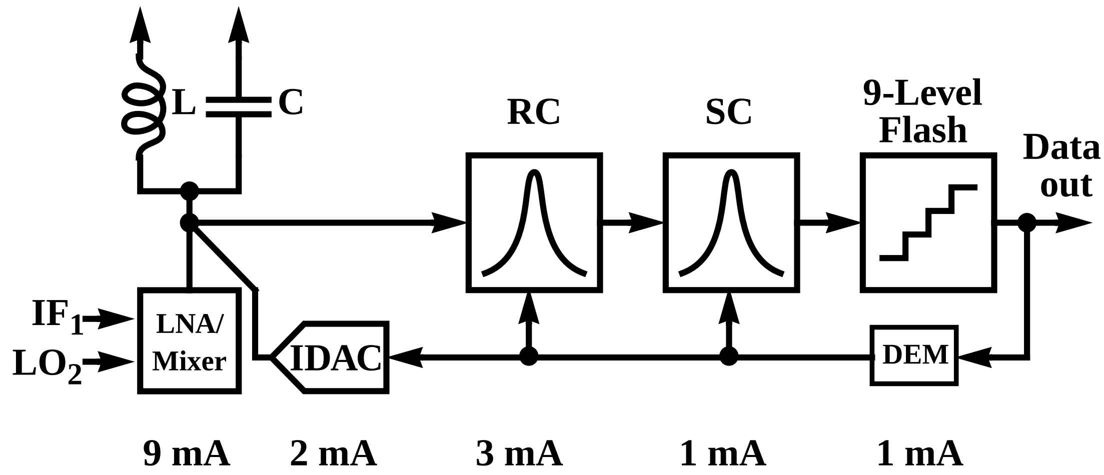
- **LC: Needs more external components plus associated pins**
- **Active-RC: 2 mA for 50 nV/$\sqrt{\text{Hz}}$ input-referred noise**
- **Switched-Cap: est. >10 mA for same noise**

Third Resonator?



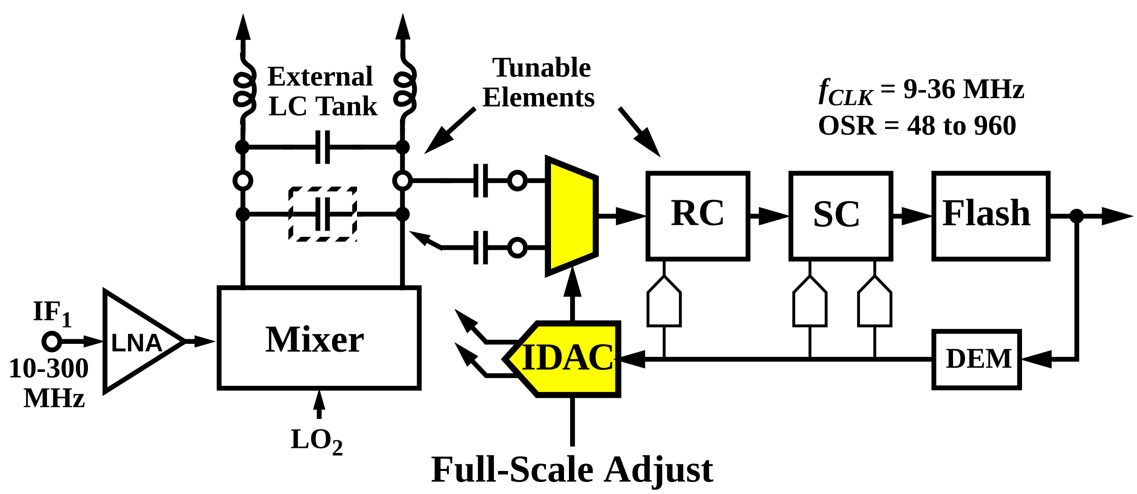
- **Active-RC: $Q \approx 10 \Rightarrow$ Need 4th resonator**
- **Switched-Cap: Q is high & drift is low;**
  <1 mA for 300 nV/$\sqrt{\text{Hz}}$ i.r.n.

Complete ADC



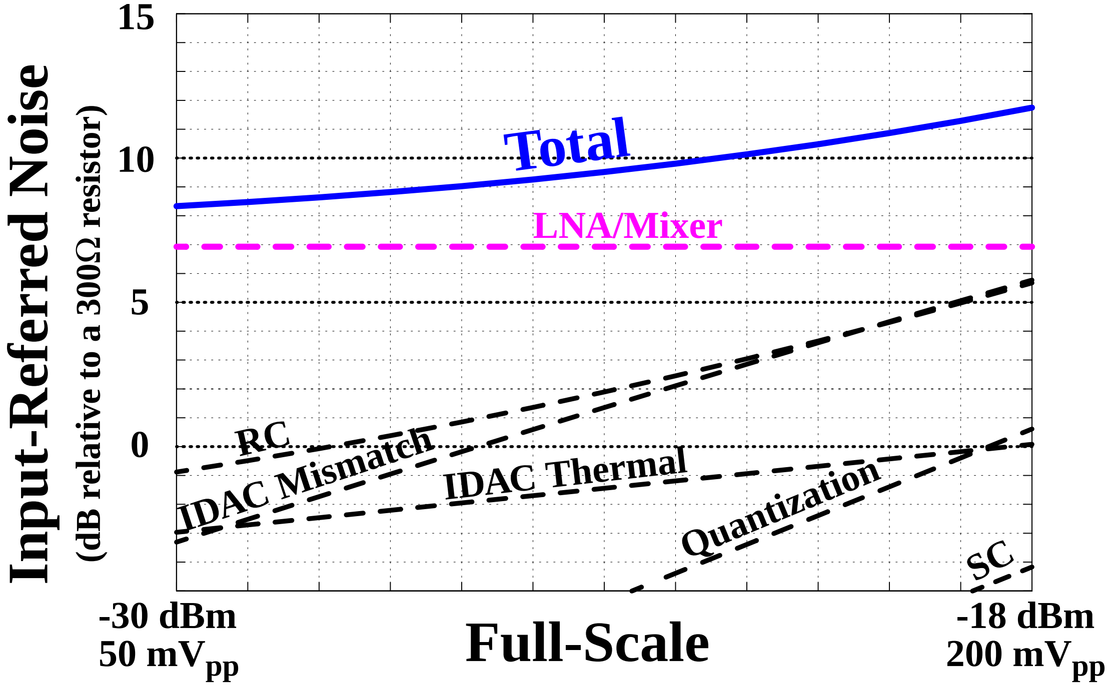
- **Eliminates high-power VGA & AAF**
  2/3 of the total power used by LNA, Mixer & IDAC
- **Uses CT and DT time elements, plus multi-bit quantization and mismatch-shaping**

ADC in More Detail

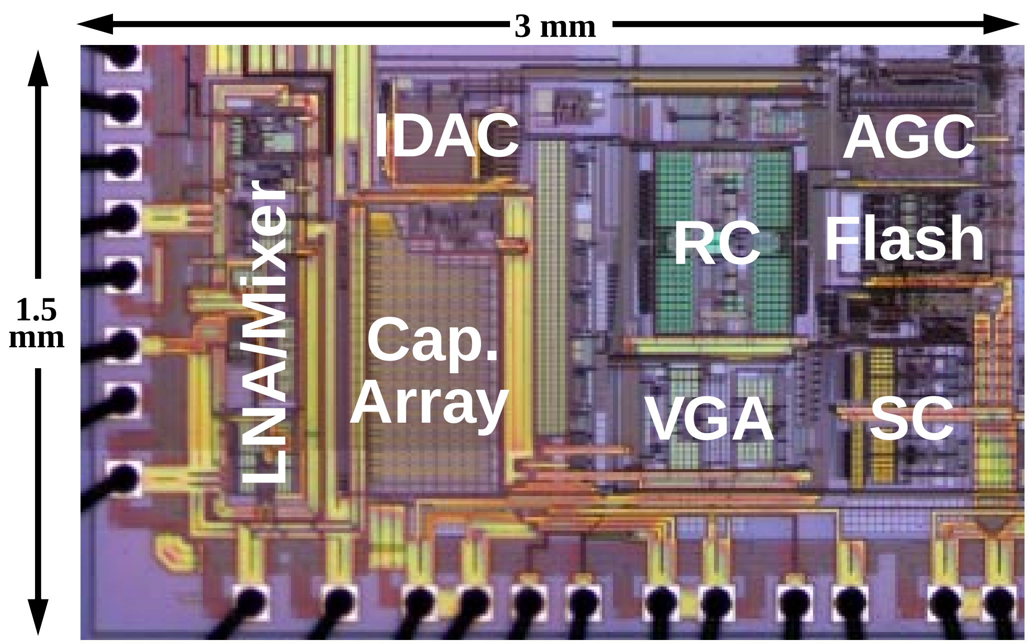


- **Can save power under small-signal conditions by reducing IDAC's full-scale**
  By a factor of 4 in this product.

Noise vs. Full-Scale

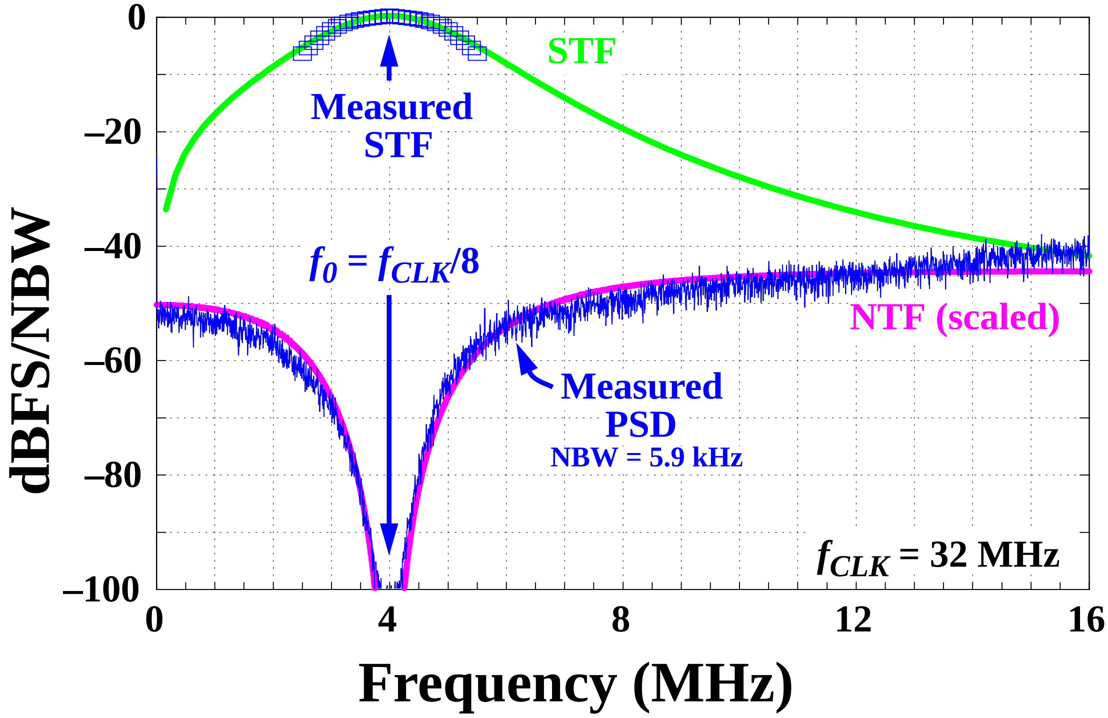


Die Photo

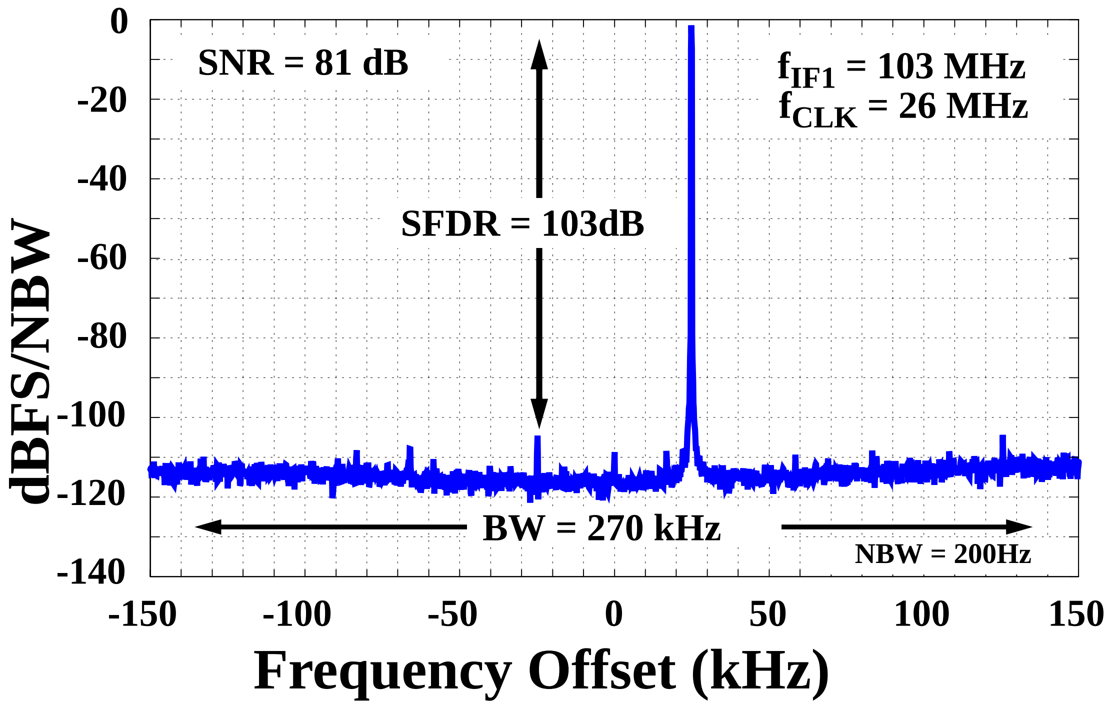


- **0.35µm BiCMOS process**

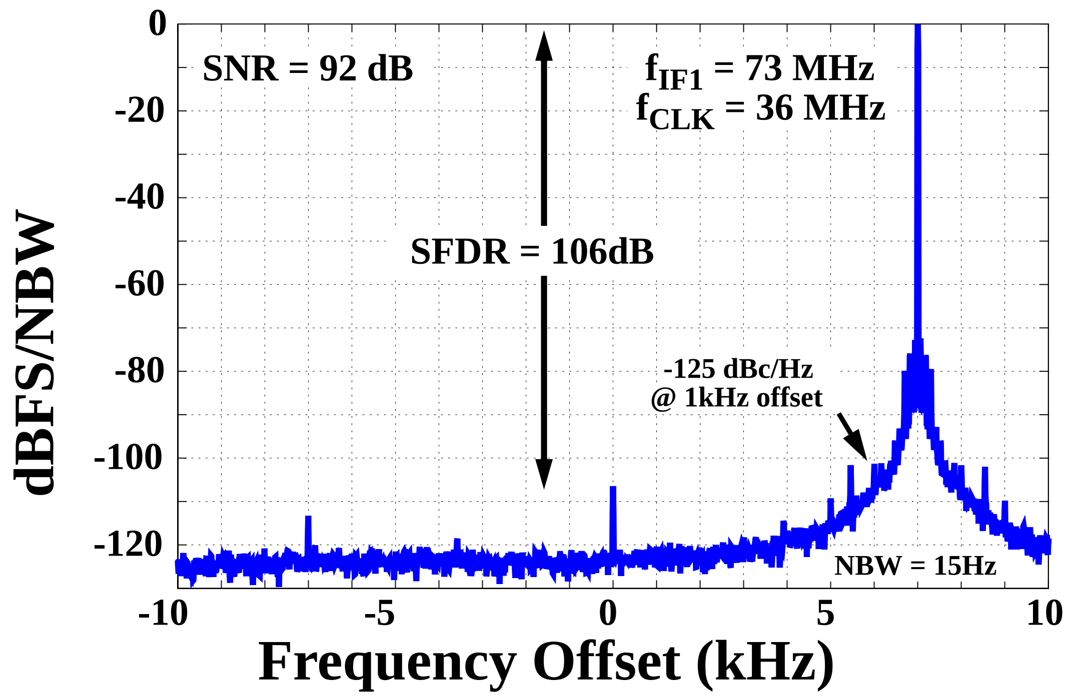
Measured STF & NTF



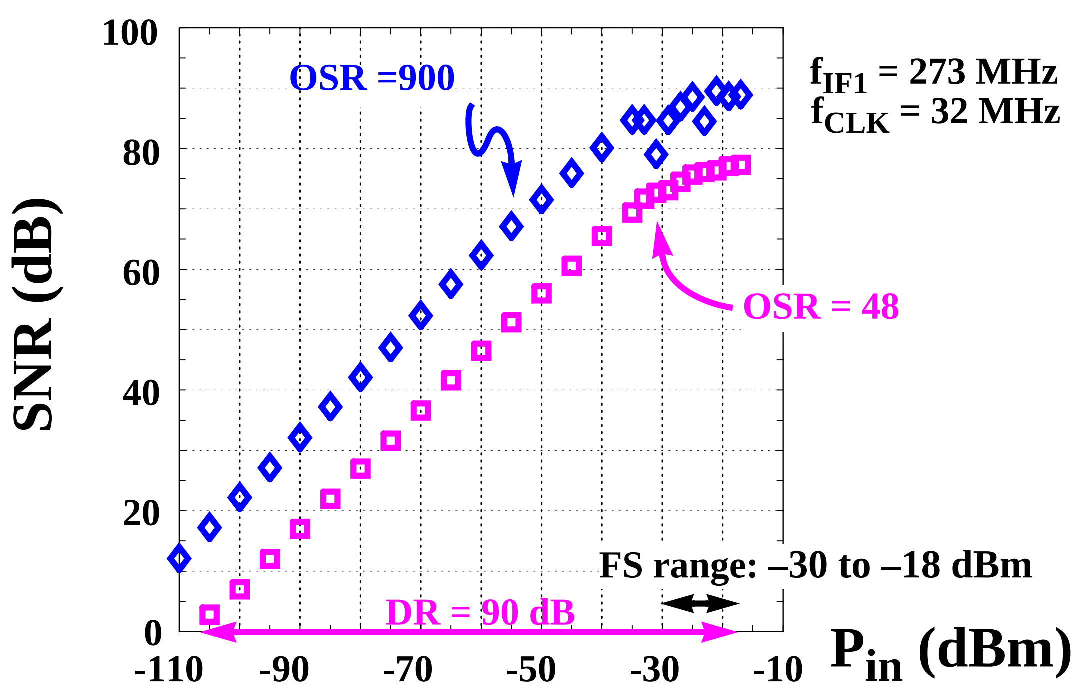
In-Band Spectrum (OSR=48)



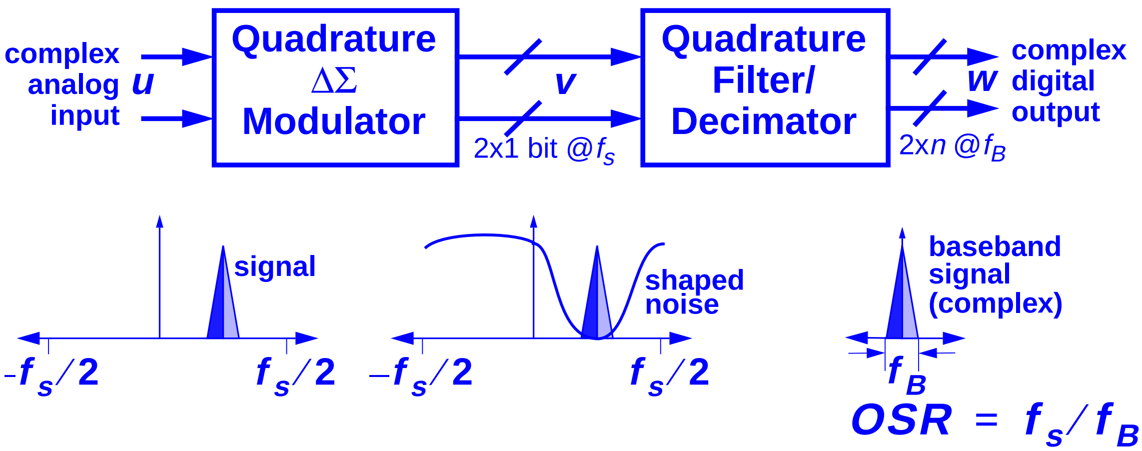
In-Band Spectrum (OSR=900)



SNR vs. Input Power



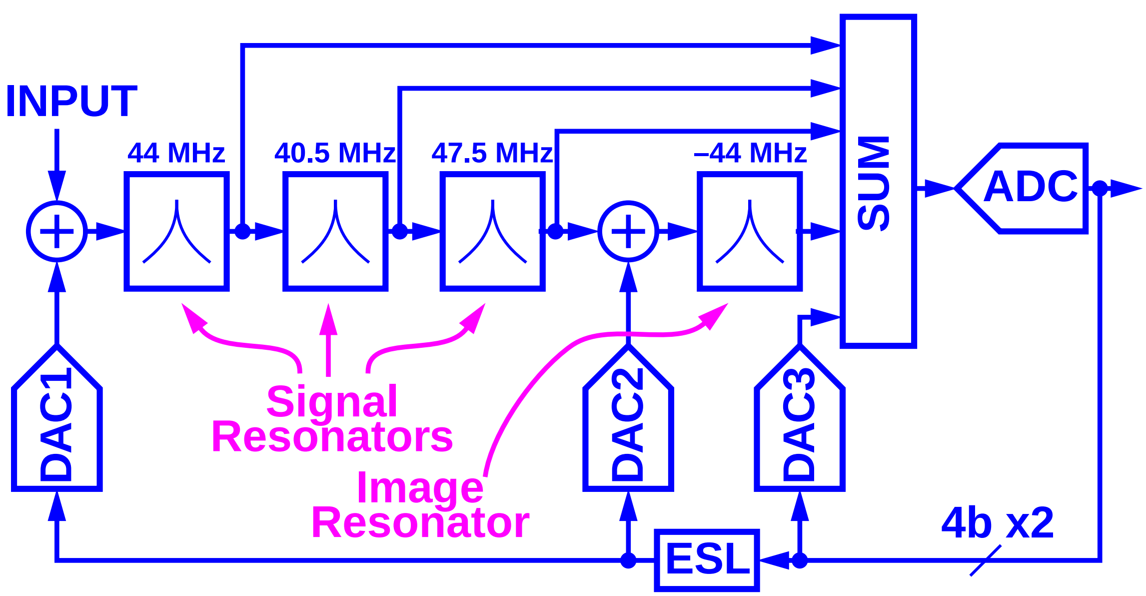
A Quadrature ΔΣ ADC System



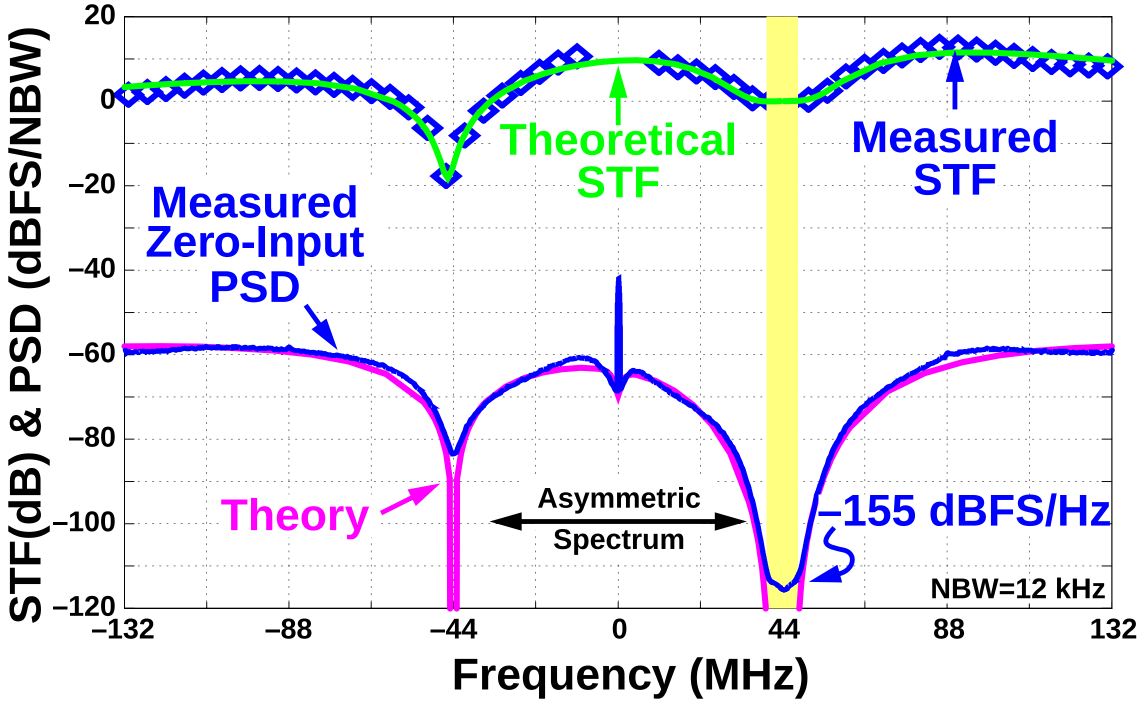
- Modulator converts its quadrature analog input into a pair of bit-stream outputs
- DSP removes out-of-band noise and translates the signal to baseband

Example Quadrature ΔΣ ADC

AD9322, AD9328



STF & NTF



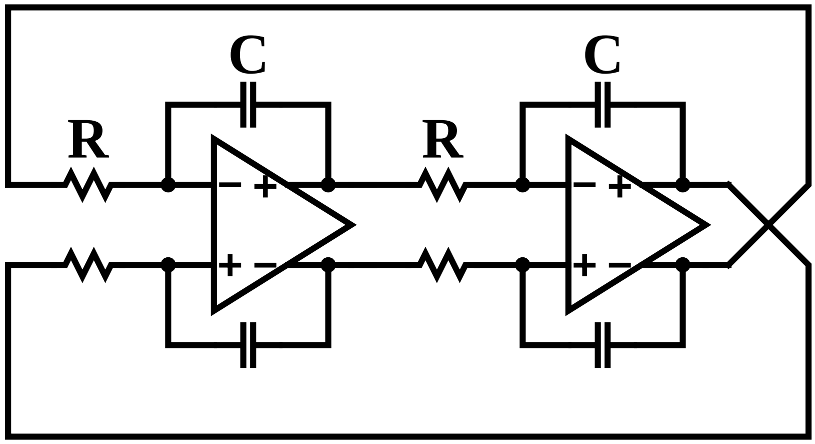
NLCOTD: High-Q Resonator

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

  Actual requirement was Q > 20.

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

Active-RC Resonator Structure

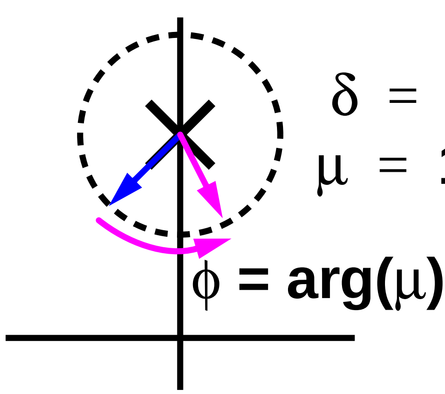


$$f_0 = \frac{1}{2\pi RC}$$

- **Tuned by adding positive feedback to make an oscillator and adjusting C until the desired resonance is achieved**
  - **Amplifier drives both R and C ⇒ trouble?**

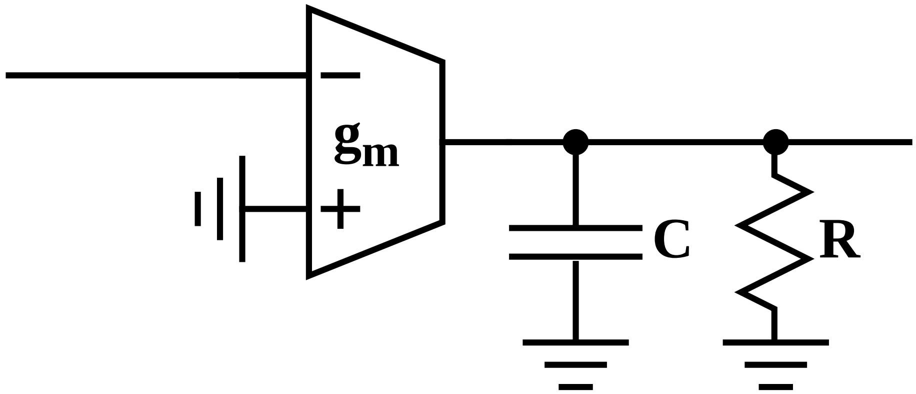
Amplifier Gain and Phase

- **Finite gain degrades Q**
- **Phase lag enhances Q**
- **Analysis shows ϕ = 45º yields high Q, regardless of amplifier gain**

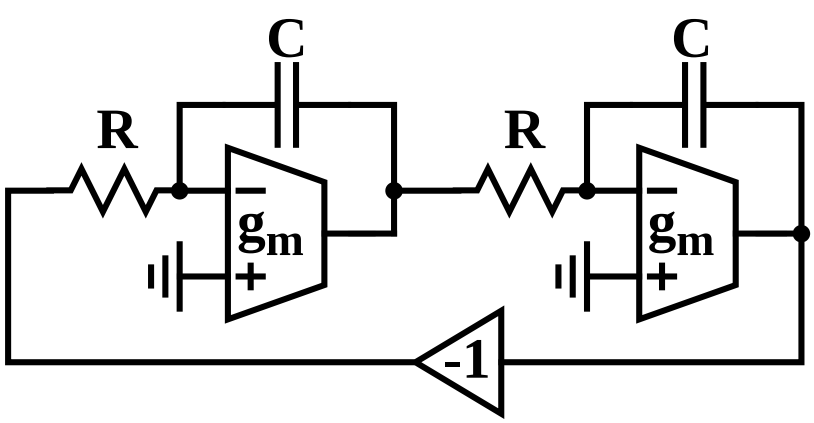


$$\delta = -\mu(\mathbf{1} + \mathbf{j})\omega_{\mathbf{0}}$$
$$\mu = \mathbf{1} / A$$

An Amplifier with $\phi = 45^o$ @ f_0:

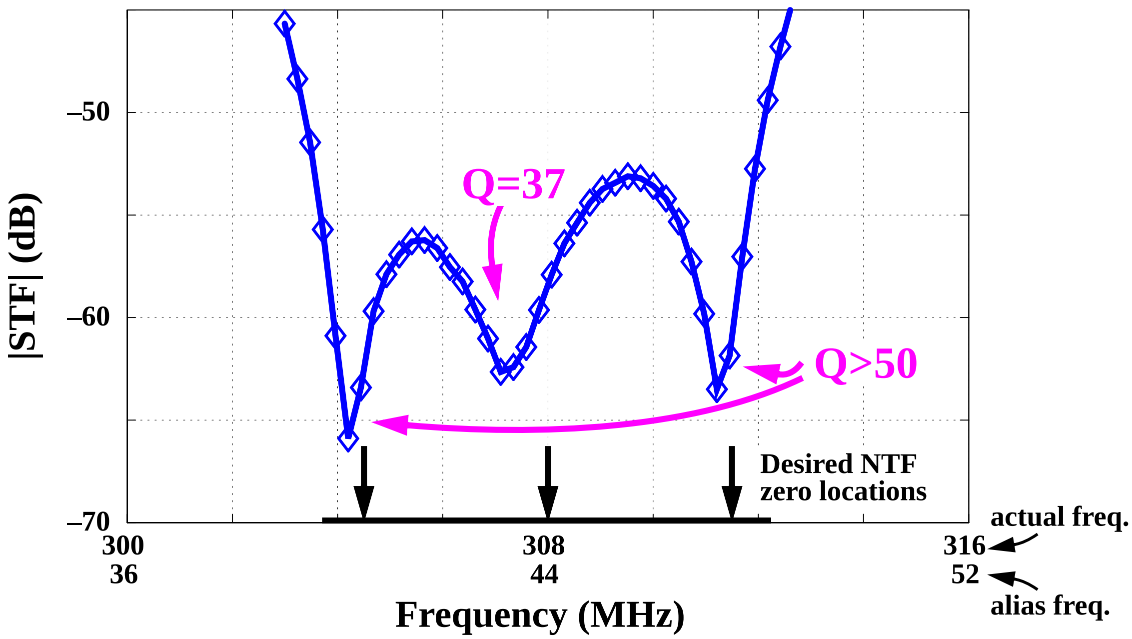


Resulting High-Q Resonator



- **Amplifier load yields $\phi = 45^o$ @ 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. **State-space (ABCD) representation of the loop filter in the ΔΣ Toolbox**
2. **MASH Modulators**
3. **Continuous-Time Modulators**
4. **Bandpass and Quadrature Bandpass ΔΣ**