Delta-Sigma Analog-to-Digital Converters

From System Architecture to Transistor-level Design

Vishal Saxena, Boise State University
(vishalsaxena@boisestate.edu)

Delta-Sigma Analog-to-Digital Converters

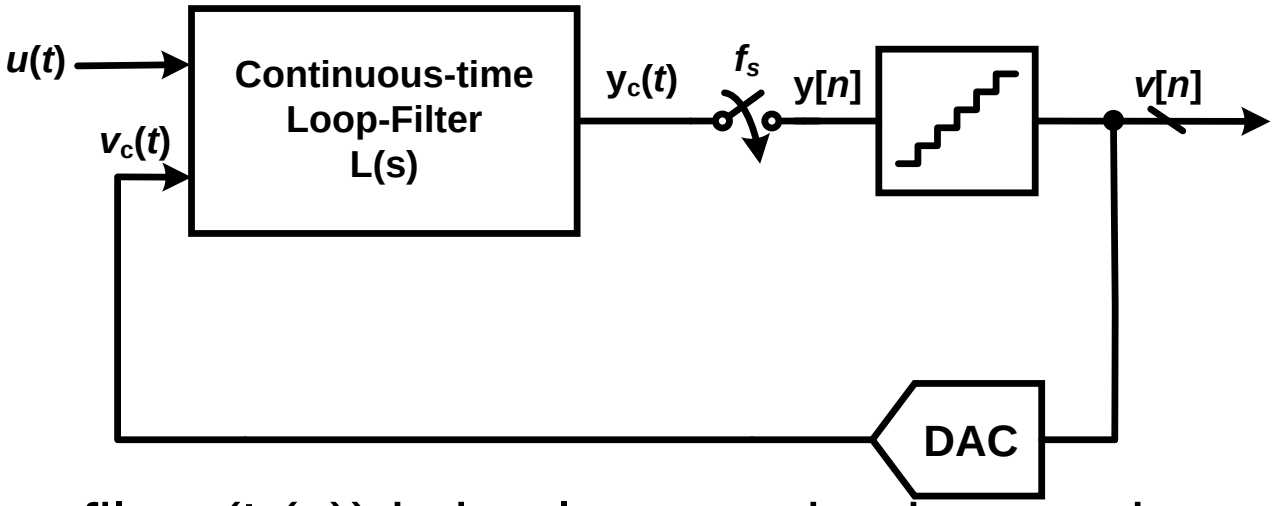
Session II – Continuous-Time $\Delta\Sigma$ ADCs

Vishal Saxena, Boise State University

Agenda

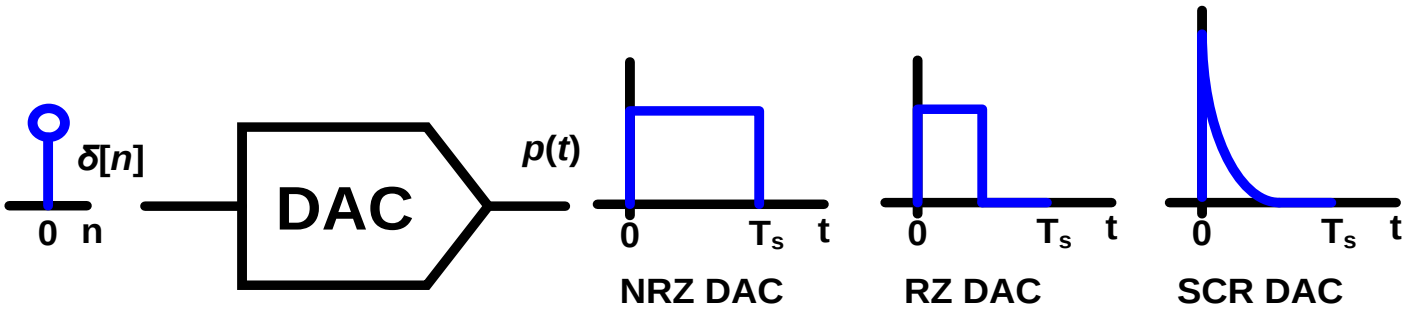
- Fundamentals of continuous-time (CT) ADCs
  - impulse invariance and CT loop-filter mapping
- Inherent anti-alias filtering in CT ΔΣ ADCs
  - feedback vs. feedforward architectures.
- CT ΔΣ ADC non-idealities:
  - Quantizer Excess loop-delay (ELD)
  - Clock jitter sensitivity
  - RC time-constant variation
- CT- ΔΣ Design techniques:
  - NRZ, RZ and switched-capacitor DACs
  - Active-RC and gm-C implementations
  - ELD compensation techniques

Continuous-Time ΔΣ Modulators



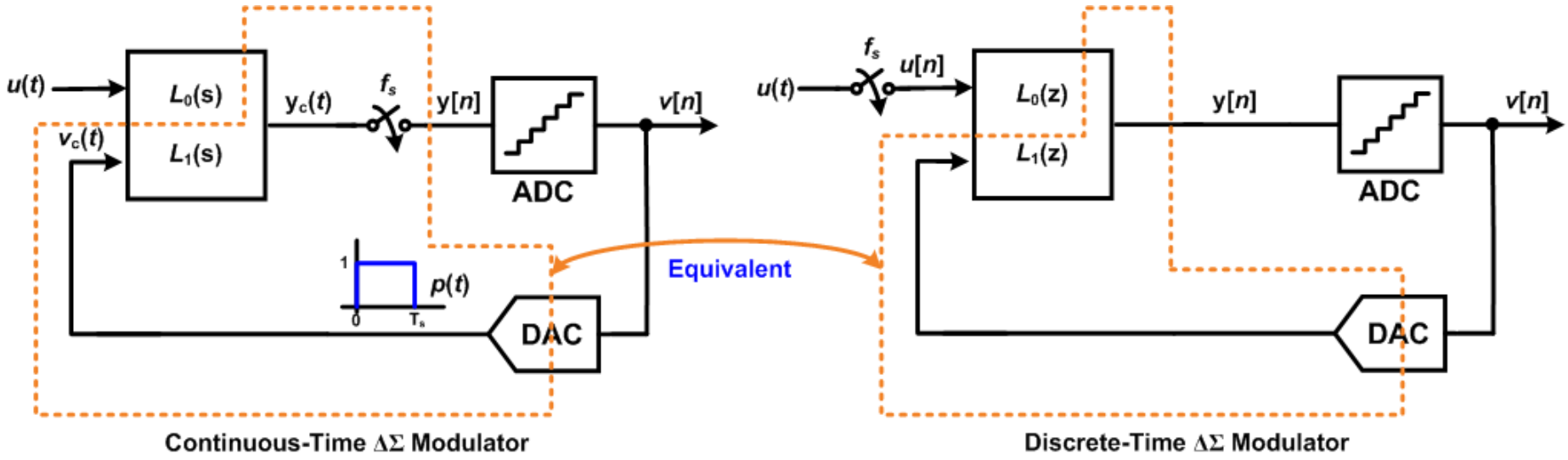
- The loop-filter (L(s)) is implemented using continuous-time circuitry
  - The modulator is still a discrete-time system
- How to design the hybrid continuous- and discrete-time loop?

Feedback DAC



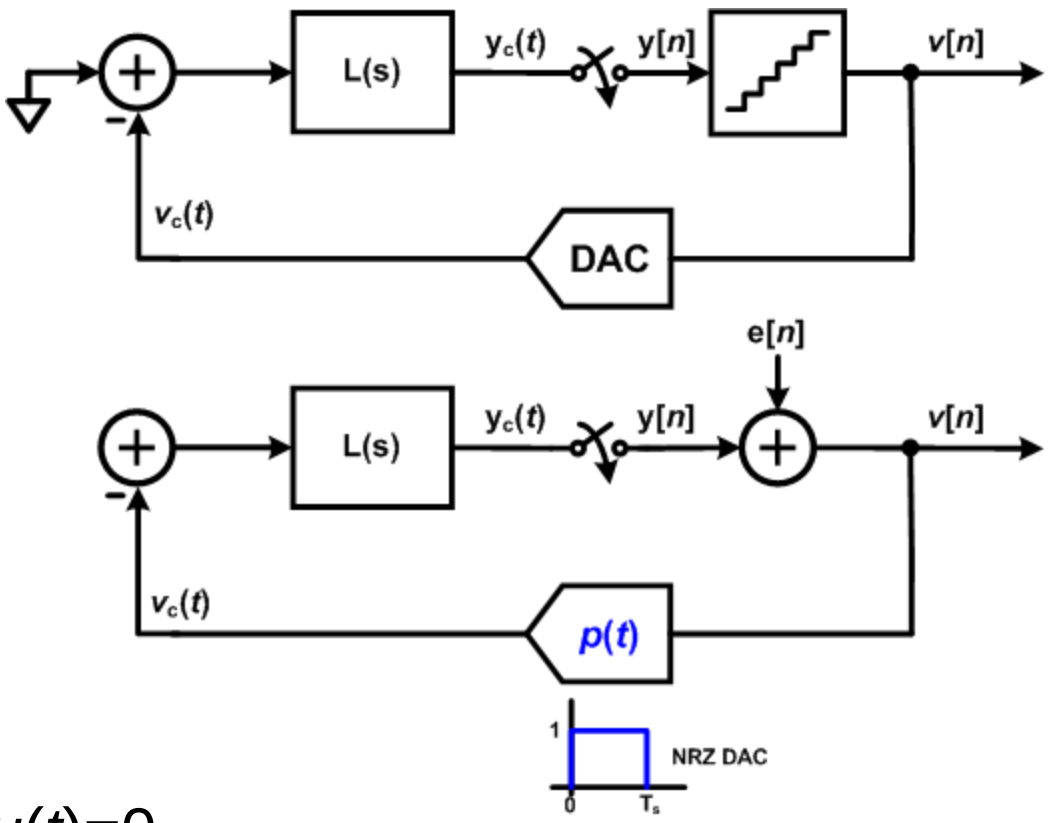
- The input to the DAC is a digital code with time-period T_s
- The DAC output is an analog waveform
- The response of the DAC to the DT impulse, δ[n], is called the pulse-shape $p(t)$
- Several choices for $p(t)$ –
  - Return to zero (RZ)
  - Non-Return to zero (NRZ)
  - Switched capacitor Resistor (SCR), etc.

Loop Modeling



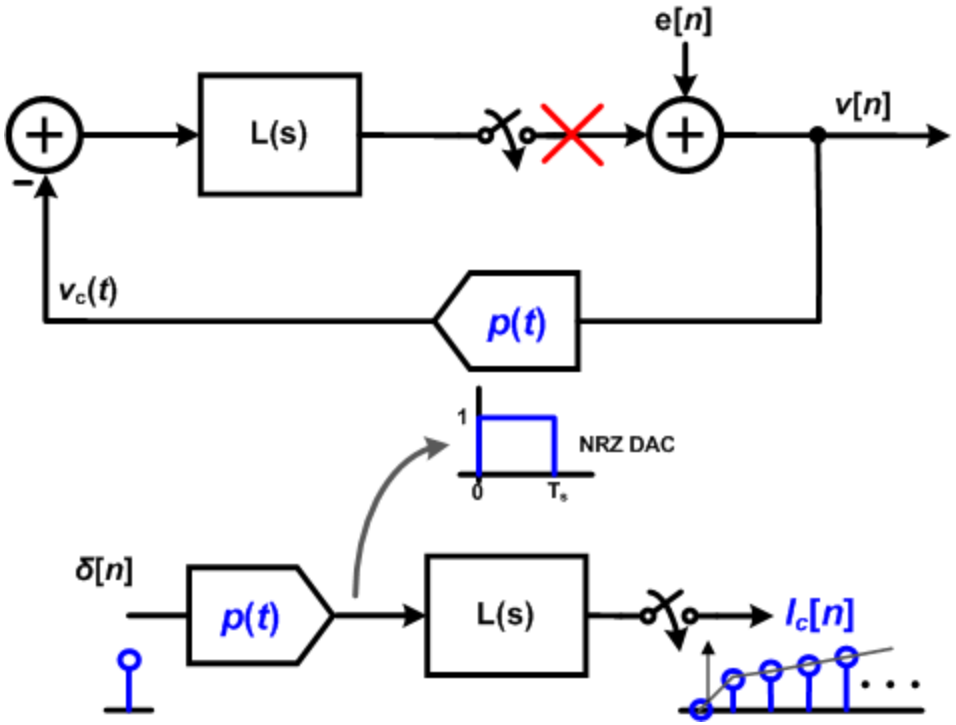
- In general loop-filter is a 2-input, 1-output LTI system
- $L_0(s)$ is the loop-filter seen by the input signal
- $L_1(s)=L(s)$, is the loop-filter seen by the feedback signal
- The DT loop-response around the ADC-DAC block should be preserved in the CT loop

Loop Modeling



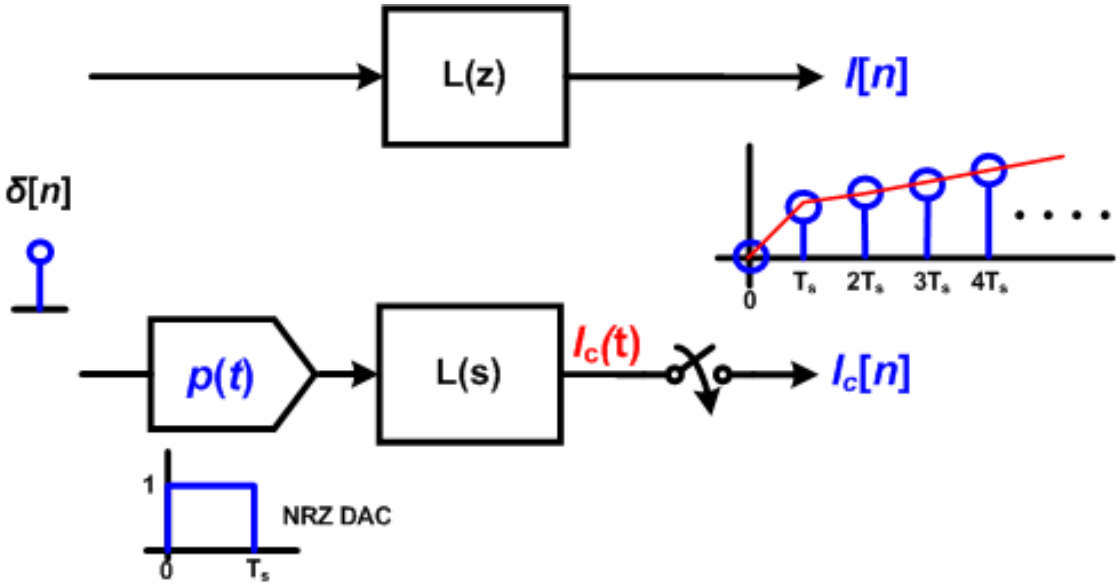
- Set input $u(t)=0$
- Replace ADC-DAC with the linear additive quantization noise model
- Model DAC as a filter with impulse response $p(t)$

Loop Modeling



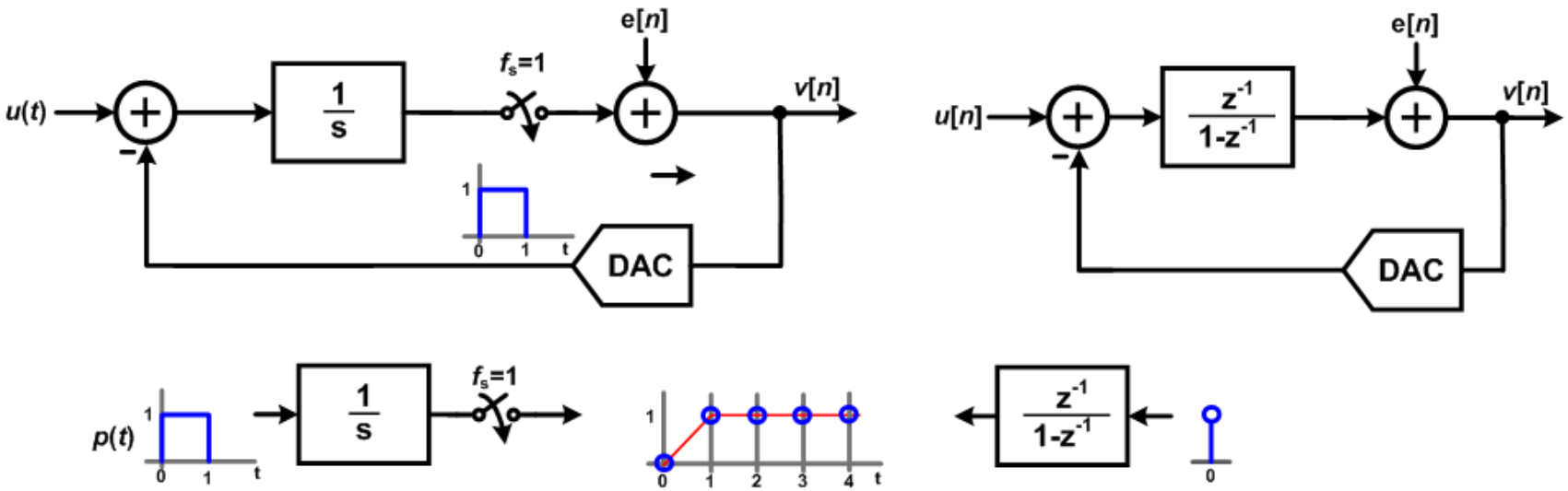
- Break the loop after the sampler
- Apply a DT impulse
- The sampled output is $l_c[n] = p(t) \otimes l(t)\,|_{nT_s}$
- The output sequence ($l_c[n]$), should be identical to the DT loop-filter response $l[n] \xrightarrow{Z} L(z)$

Synthesis of CT Loop-Filter



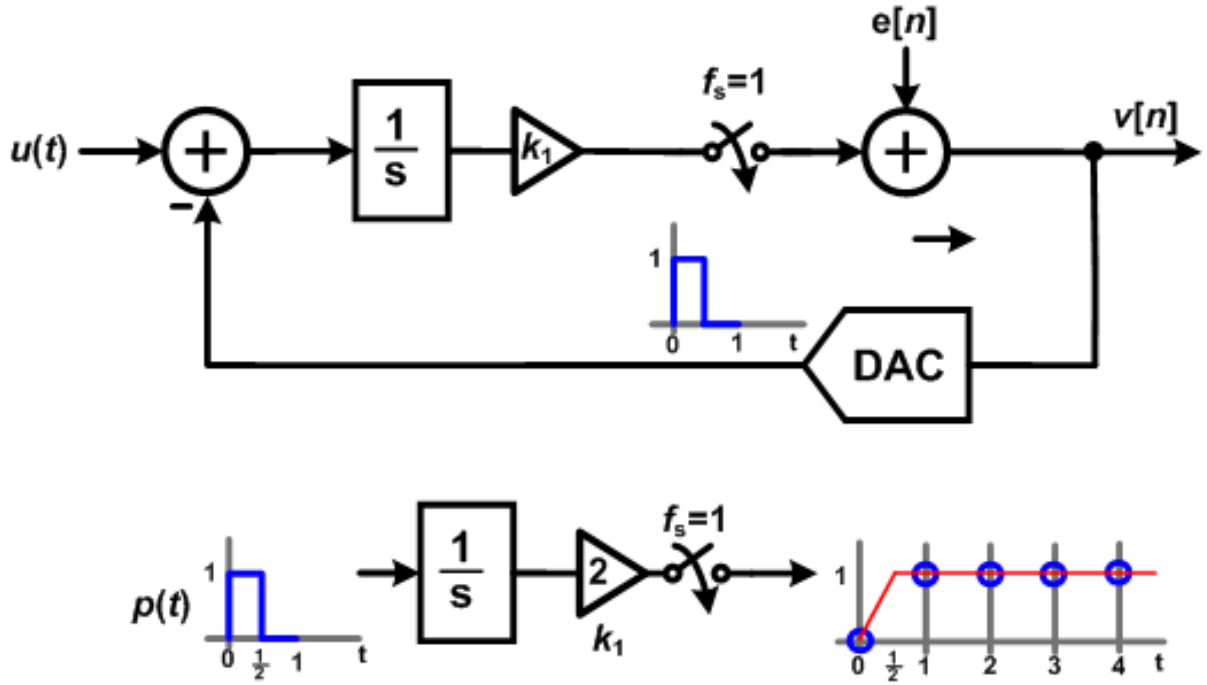
- Map the sampled CT loop-response to its DT equivalent
  - Called Impulse Invariant Transformation (IIT)
  - $l_c[n] = p(t) \otimes l(t)\,|_{nT_s} \equiv l[n]$
- *NTF*(*z*) is preserved in the DT-to-CT if the transformation is correctly performed

First-order Example – NRZ DAC



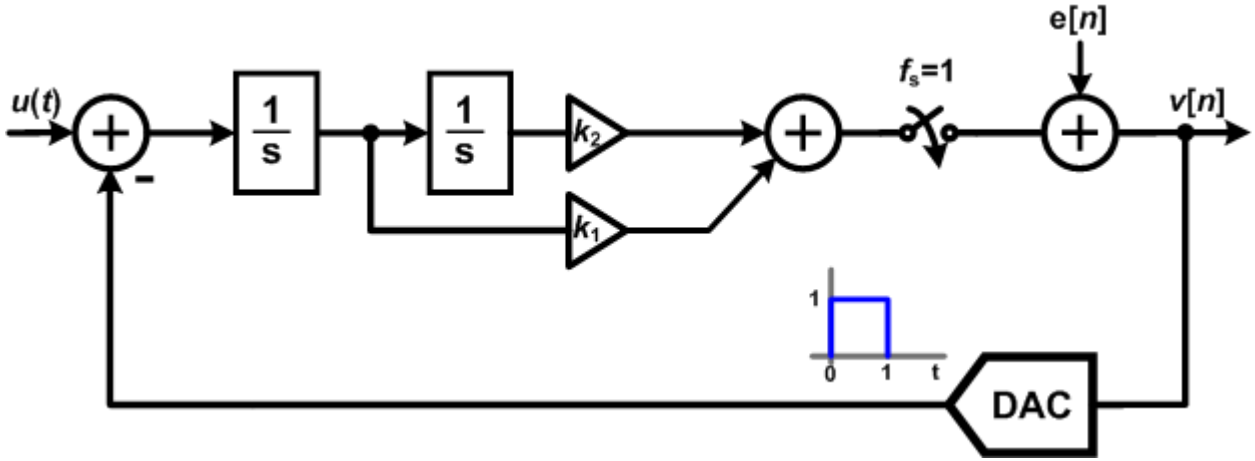
- Without loss of generality, let f_s=1 Hz, T_s=1 s
- Sampled loop-response $l_c[n]$={0,1,1,1,1,….}
  - Identical to the DT loop-response
- $$L(z) = \frac{z^{-1}}{1 - z^{-1}}$$
- $$NTF(z) = \frac{1}{1 + L(z)} = 1 - z^{-1}$$

First-order Example – RZ DAC



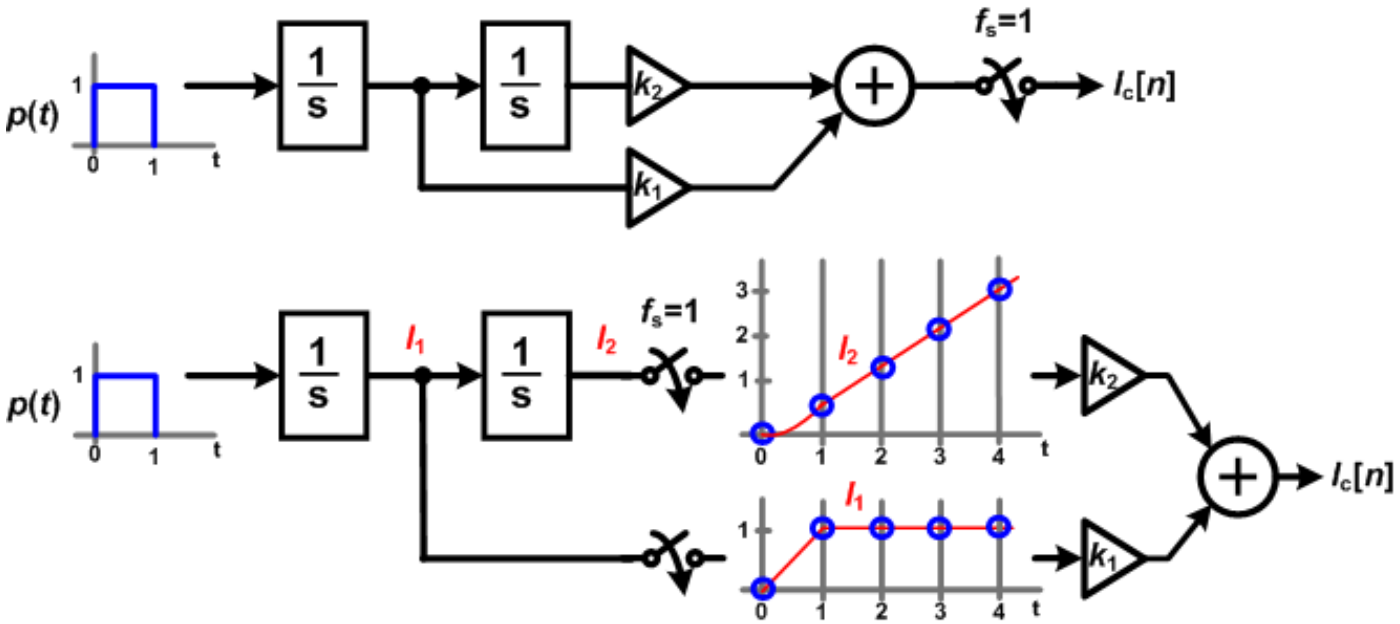
- With RZ DAC, scaling by a factor of k_1=2 is need to restore the loop-response
- Can have any number of CT responses to result in the same DT loop-response
  - The procedure can be conceptually applied to any other DAC pulse shape

Second-order Example



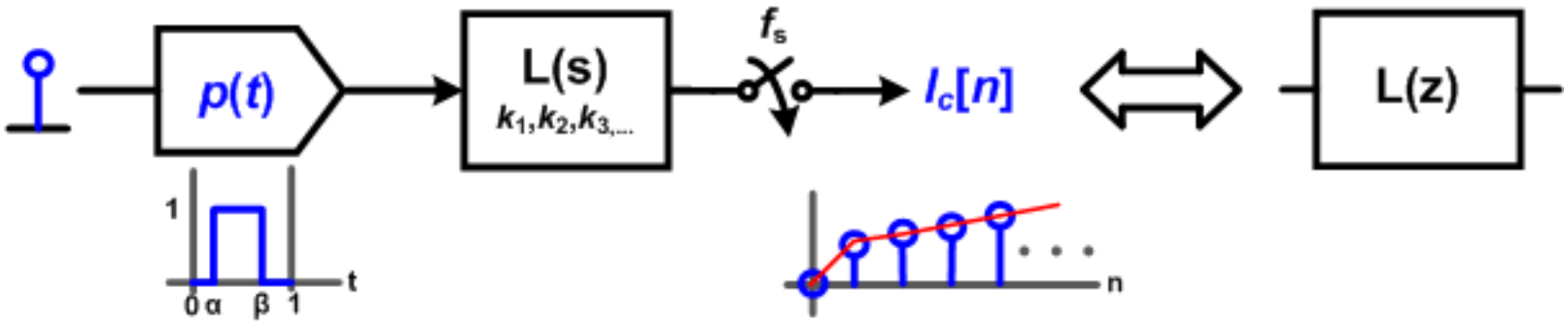
- $NTF(z) = \left(1 - z^{-1}\right)^2$
- $L(z) = \frac{1}{NTF(z)} - 1 = \frac{1}{\left(1 - z^{-1}\right)^2} - 1 = \frac{z^{-1}}{1 - z^{-1}} + \frac{z^{-1}}{\left(1 - z^{-1}\right)^2}$
- $l[n]$ = {0,1,1,…} + {0,1,2,3,…} = {0,2,3,4,5…}
- The loop-response is implemented using a cascade of two CT integrators
  - Need to find suitable loop-filter coefficients k_1 and k_2 using IIT

Second-order Example



- $I[n]$ = {0,2,3,4,5…}
- $I_c[n] = k_1 \cdot \{0,1,1,1,\ldots\} + k_2 \cdot \{0,1.5, 2.5, 3.5,\ldots\}$
- Solving for $I_c[n] = I[n]$: $k_1=1.5$ and $k_2=1$
- For NRZ DAC, CT loop-filter, $L(s) = \frac{1.5}{s} + \frac{1}{s^2}$
- Show that for RZ DAC, $k_1=2.5$ and $k_2=2$

Impulse-Invariant Transformation



- Impulse-invariant transformation maps DT modulator to CT
  - $l[n] = p(t) \otimes l(t)\,|_{t=nT_s}$
  - $\mathbf{Z}^{-1}\{L(z)\} = \mathbf{L}^{-1}\{P(s)\cdot L(s)\}\,|_{t=nT_s}$
- These relations have been derived for standard z-domain poles
  - Uses generic rectangular DAC pulse defined by (α, β), 0≤α<β ≤ 1
  - Tables available for z-domain to s-domain pole equivalence and vice versa [Cherry]
  - A DT pole a $z=z_k$ transforms to a CT pole at $s_k=\ln(z_k)$ with the same multiplicity (l)

z-Domain to s-Domain Loop Filter Poles

s-DOMAIN EQUIVALENCES FOR z-DOMAIN LOOP FILTER POLES

z-domain pole	s-domain equivalent	Limit for $z_k = 1$
$\frac{1}{z-z_k}$	$\frac{r_0}{s-s_k} \times \frac{1}{z_k^{1-\alpha}-z_k^{1-\beta}}$ $r_0 = s_k$	$\frac{r_0}{s-s_k}$ $r_0 = \frac{1}{\beta-\alpha}$
$\frac{1}{(z-z_k)^2}$	$\frac{r_1 s+r_0}{(s-s_k)^2} \times \frac{1}{z_k(z_k^{1-\alpha}-z_k^{1-\beta})^2}$ $r_1 = q_1 s_k + q_0$ $r_0 = q_1 s_k^2$ $q_1 = z_k^{1-\beta}(1-\beta) - z_k^{1-\alpha}(1-\alpha)$ $q_0 = z_k^{1-\alpha} - z_k^{1-\beta}$	$\frac{r_1 s+r_0}{(s-s_k)^2}$ $r_1 = \frac{1}{2}\frac{\alpha+\beta-2}{\beta-\alpha}$ $r_0 = \frac{1}{\beta-\alpha}$
$\frac{1}{(z-z_k)^3}$	$\frac{r_2 s^2+r_1 s+r_0}{(s-s_k)^3} \times \frac{1}{z_k^2(z_k^{1-\alpha}-z_k^{1-\beta})^3}$ $r_2 = \frac{1}{2}q_2 s_k - q_1$ $r_1 = -q_2 s_k^2 + q_1 s_k + q_0$ $r_0 = \frac{1}{2}q_2 s_k^3$ $q_2 = (1-\beta)(2-\beta)(z_k^{1-\beta})^2$ $+ (1-\alpha)(2-\alpha)(z_k^{1-\alpha})^2$ $+ [\beta(\beta+3)+\alpha(\alpha+3)$ $- 4(1+\alpha\beta)]z_k^{1-\alpha}z_k^{1-\beta}$ $q_1 = (\frac{3}{2}-\beta)(z_k^{1-\beta})^2 + (\frac{3}{2}-\alpha)(z_k^{1-\alpha})^2$ $+ (\alpha+\beta-3)z_k^{1-\alpha}z_k^{1-\beta}$ $q_0 = (z_k^{1-\alpha} - z_k^{1-\beta})^2$	$\frac{r_2 s^2+r_1 s+r_0}{(s-s_k)^3}$ $r_2 = \frac{1}{12}\frac{1}{\beta-\alpha}[\beta(\beta-9)$ $+ \alpha(\alpha-9)+4\alpha\beta+12]$ $r_1 = \frac{1}{2}\frac{\alpha+\beta-3}{\beta-\alpha}$ $r_0 = \frac{1}{\beta-\alpha}$

s-Domain to z-Domain Loop Filter Poles

z-DOMAIN EQUIVALENCES FOR s-DOMAIN LOOP FILTER POLES

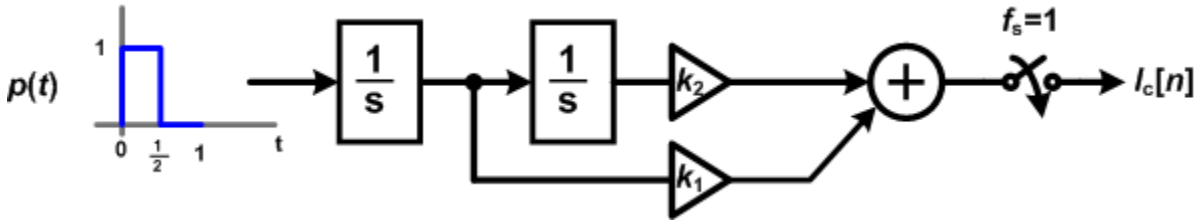
s-domain pole	z-domain equivalent	Limit for $s_k = 0$
$\frac{1}{s-s_k}$	$\frac{y_0}{z-z_k} \times \frac{1}{s_k}$ $y_0 = z_k^{1-\alpha} - z_k^{1-\beta}$	$\frac{y_0}{z-z_k}$ $y_0 = \beta - \alpha$
$\frac{1}{(s-s_k)^2}$	$\frac{y_1 z+y_0}{(z-z_k)^2} \times \frac{1}{s_k^2}$ $y_1 = z_k^{1-\beta}[1 - s_k(1-\beta)] - z_k^{1-\alpha}[1 - s_k(1-\alpha)]$ $y_0 = z_k^{2-\alpha}(1 + s_k\alpha) - z_k^{2-\beta}(1 + s_k\beta)$	$\frac{y_1 z+y_0}{(z-z_k)^2}$ $y_1 = \frac{1}{2}[\beta(2-\beta) - \alpha(2-\alpha)]$ $y_0 = \frac{1}{2}(\beta^2 - \alpha^2)$
$\frac{1}{(s-s_k)^3}$	$\frac{y_2 z^2+y_1 z+y_0}{(z-z_k)^3} \times \frac{1}{s_k^3}$ $y_2 = z_k^{1-\beta}[-1 + s_k(1-\beta) + \frac{s_k^2}{2}(1-\beta)^2] - z_k^{1-\alpha}[-1 + s_k(1-\alpha) + \frac{s_k^2}{2}(1-\alpha)^2]$ $y_1 = z_k^{2-\beta}[2 - s_k(1-2\beta) + \frac{s_k^2}{2}(-1-2\beta+2\beta^2)] + z_k^{2-\alpha}[2 - s_k(1-2\alpha) + \frac{s_k^2}{2}(-1-2\alpha+2\alpha^2)]$ $y_0 = z_k^{3-\alpha}(1 + s_k\alpha + \frac{s_k^2}{2}\alpha^2) - z_k^{3-\beta}(1 + s_k\beta + \frac{s_k^2}{2}\beta^2)$	$\frac{y_2 z^2+y_1 z+y_0}{(z-z_k)^3}$ $y_2 = \frac{1}{6}(\beta^3 - \alpha^3) - \frac{1}{2}(\beta^2 - \alpha^2) + \frac{1}{2}(\beta - \alpha)$ $y_1 = \frac{1}{3}(\beta^3 - \alpha^3) - \frac{1}{2}(\beta^2 - \alpha^2) - \frac{1}{2}(\beta - \alpha)$ $y_0 = \frac{1}{6}(\beta^3 - \alpha^3)$

IIT Example: 2nd-order NTF, RZ DAC

- Expand L(z) as pole-form partial fractions

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

$$L(z) = \frac{2}{z-1} + \frac{1}{(z-1)^2}$$



$$p(t) \square [\alpha, \beta] = [0, 0.5]$$

- From the equivalence tables

$$\frac{1}{z-1} \xrightarrow{IIT} \frac{1}{(\beta-\alpha)} \cdot \frac{1}{s - s_k} = \frac{2}{s}$$

$$\frac{1}{(z-1)^2} \xrightarrow{IIT} = \frac{\frac{1}{2}\left(\frac{\alpha+\beta-2}{\beta-\alpha}\right)s + \frac{1}{(\beta-\alpha)}}{s^2} = \frac{(2-1.5s)}{s^2}$$

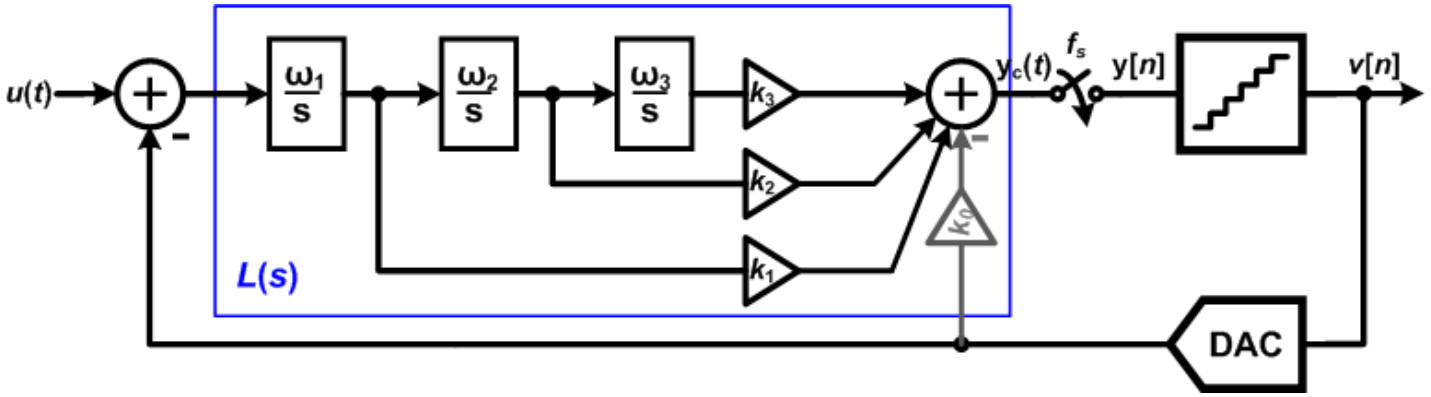
- Combining the terms, we get

$$\Rightarrow L(s) = \frac{2}{s} + \frac{2-1.5s}{s^2} = \frac{2+2.5s}{s^2}$$

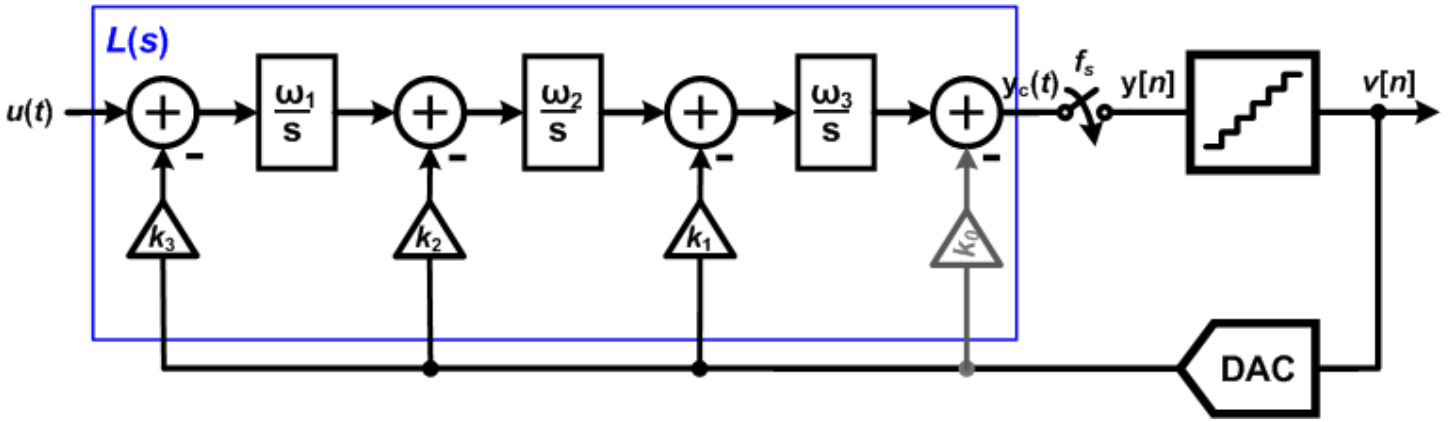
Continuous-Time ΔΣ Summary

- A DT loop can be emulated using a CT loop-filter
- The DT-to-CT transformation depends on the DAC pulse shape
- The transformation can be performed using analytical as well as numerical methods
- This technique can be extended to higher-order NTFs
  - From the desired NTF(z), find L(z)
  - Convert L(z) into L(s) using the DAC pulse shape
  - Schreier ΔΣ Toolbox function **realizeNTF_ct** can be used for rectangular DAC pulses
  - Implement L(s) using a suitable loop-filter topology
- A CT ΔΣ modulator has significant advantages over it's DT implementation…. stay tuned

Loop-Filter Architectures

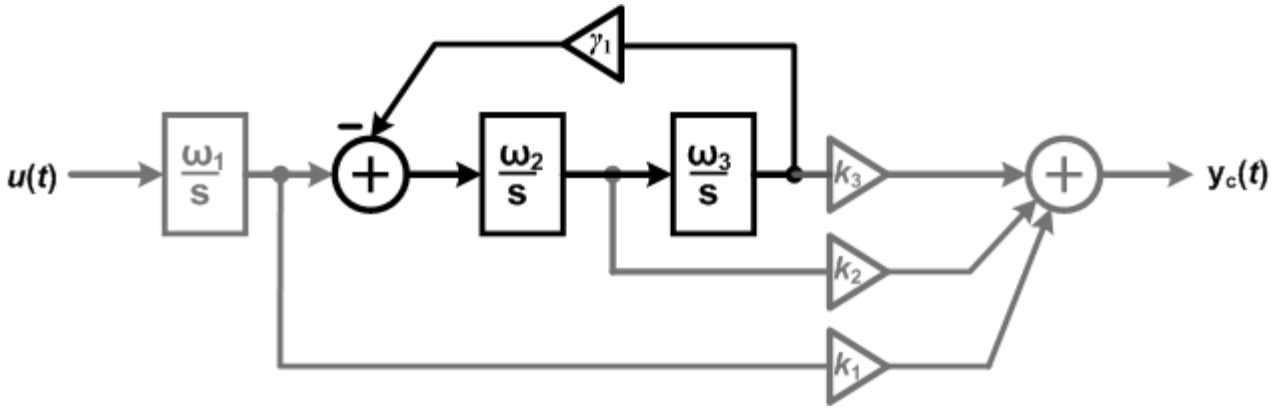


Cascade of Integrators with Feedforward Summation (**CIFF**)



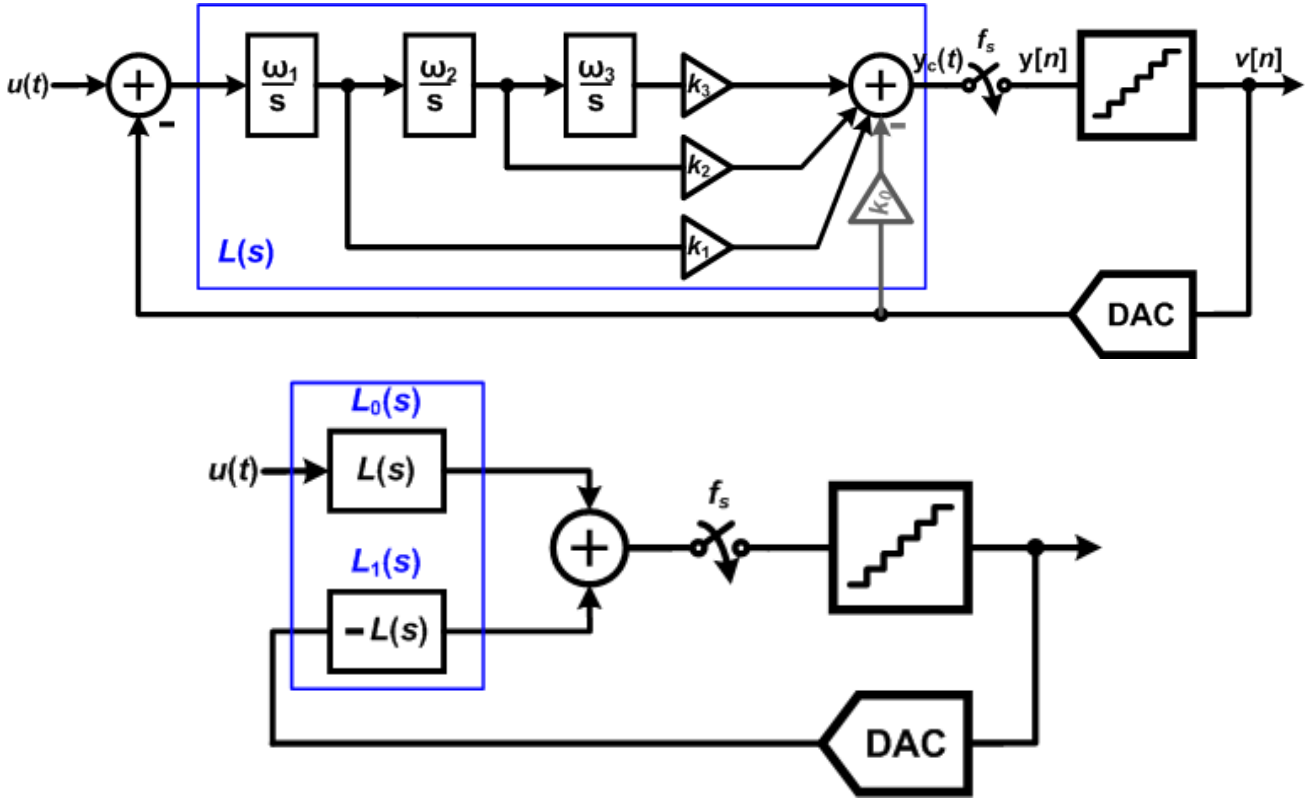
Cascade of Integrators with Distributed Feedback (**CIFB**)

Resonators for Complex NTF Zeros



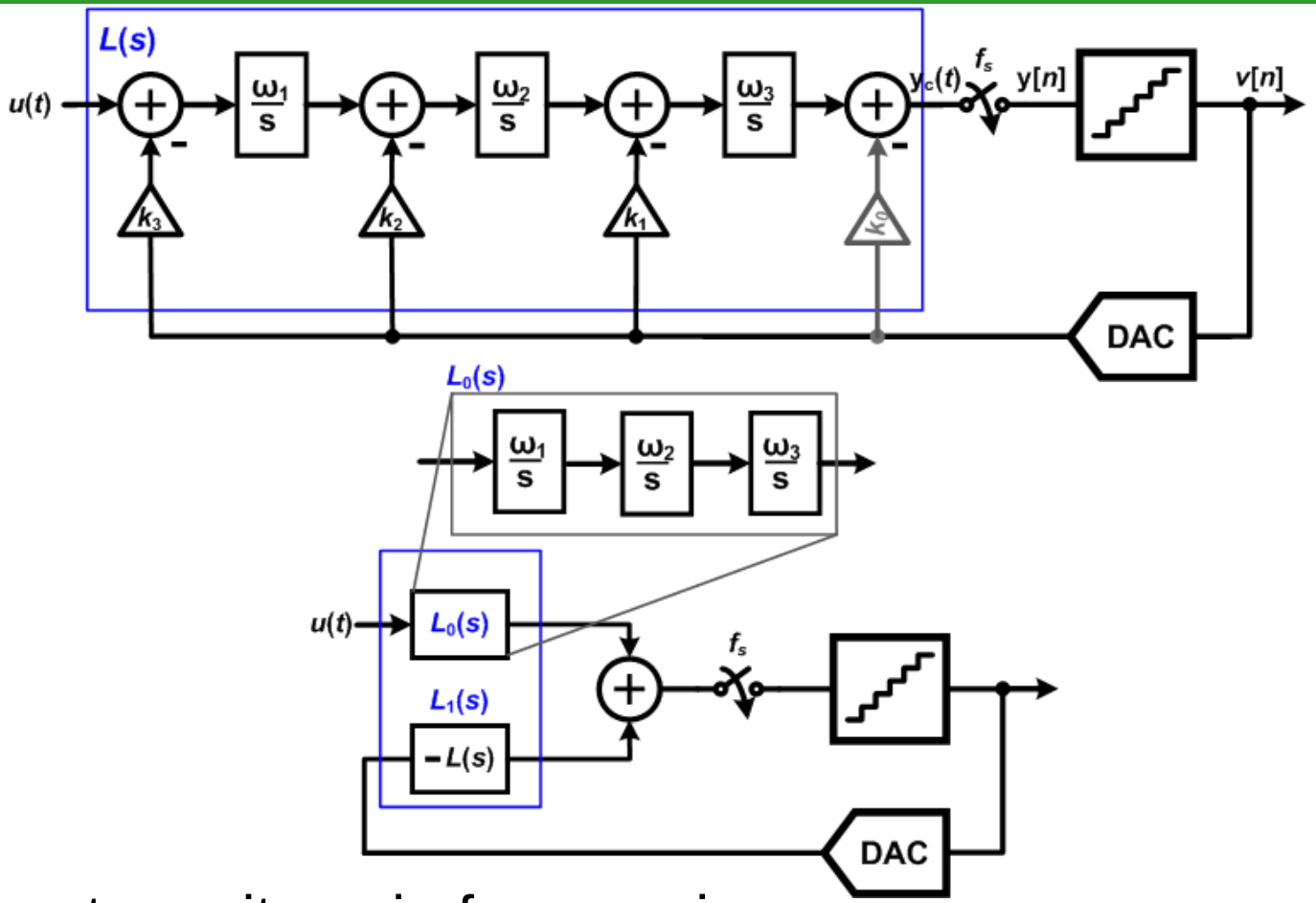
- Complex NTF(z) zeroes, $z_k = e^{\pm j\alpha}$ transform into conjugate loop-filter poles at $s_k = \pm\left|\angle z_k\right| = \pm j\alpha$
- Here, $\alpha = \sqrt{\gamma_1 \omega_2 \omega_3}$

CIFF Loop-Filter



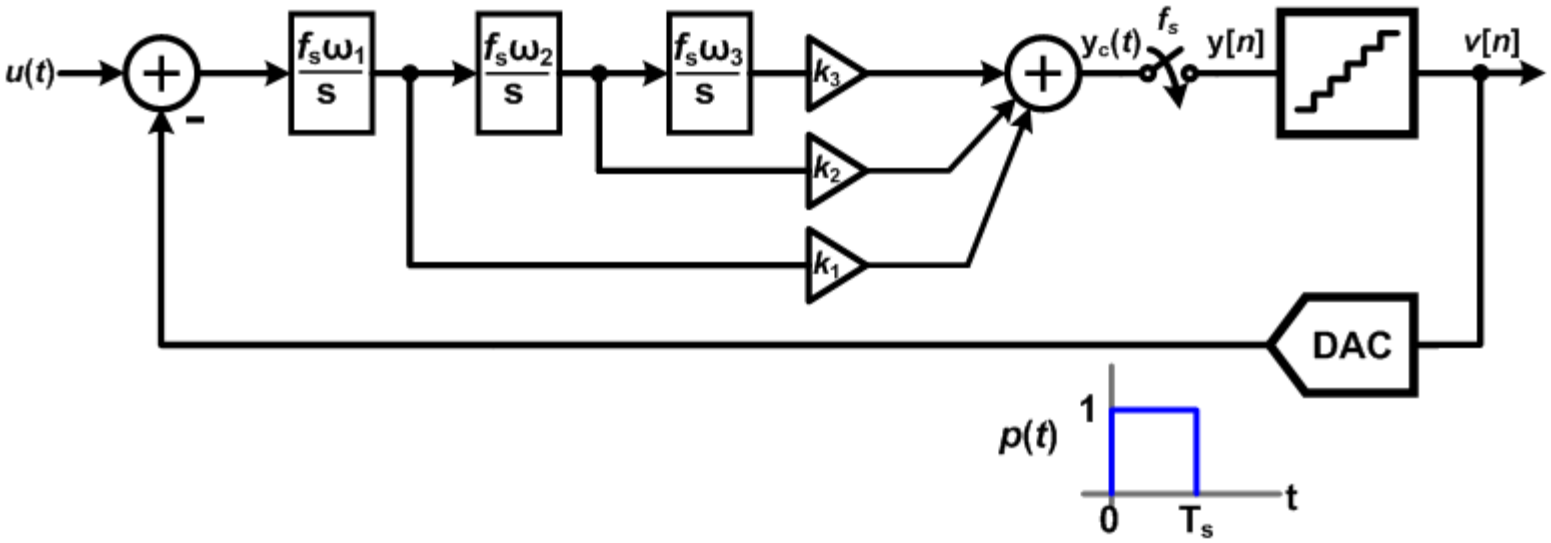
- Integrator unity-gain frequencies $\omega_1 > \omega_2 > \omega_3$
  - First integrator is the fastest while the last is slowest
- Since the first integrator needs to be fastest for noise and linearity considerations, results in lower power loop-filter

CIFB Loop-Filter



- Integrator unity-gain frequencies $\omega_1 < \omega_2 < \omega_3$
  - First integrator is the slowest while the last is fastest
- Not power efficient compared to CIFF

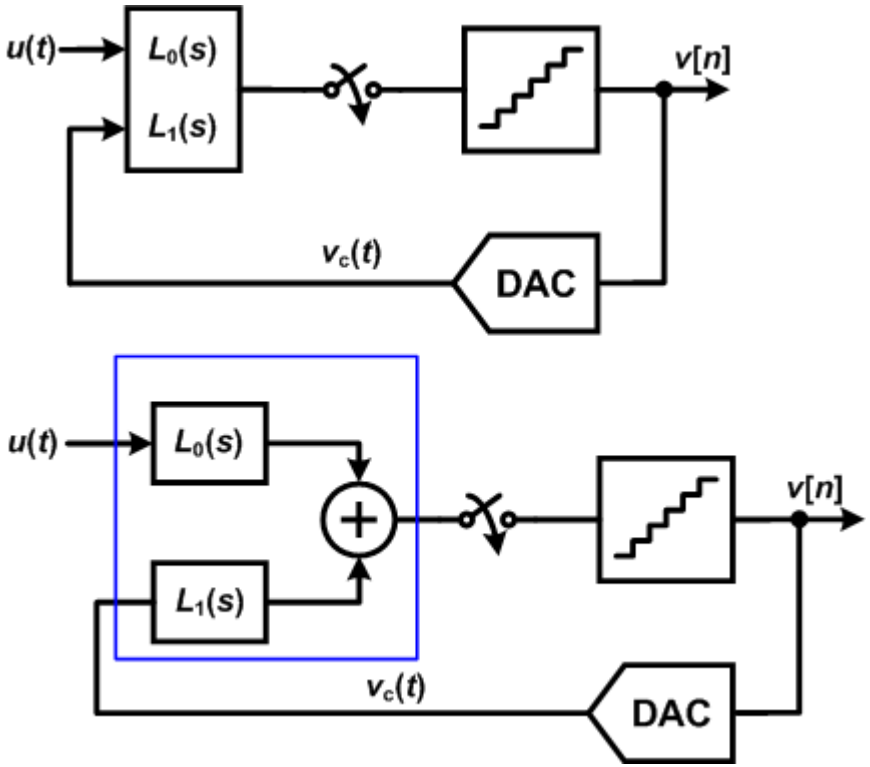
Loop-Filter Time-Scaling



- Changing the sampling rate for the normalized design from 1Hz to f_s
- DAC pulse-shape stretches by T_s
- loop-filter scales as

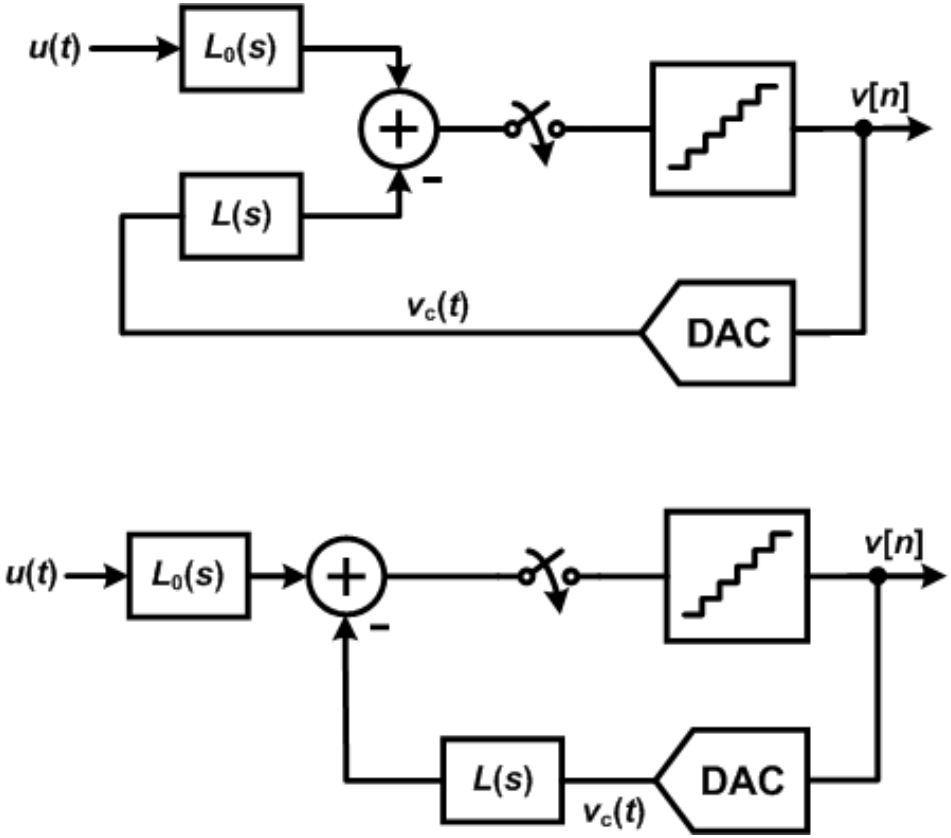
$$\frac{1}{s} \rightarrow \frac{f_s}{s} = \frac{1}{sT_s}$$

Implicit Anti-Aliasing in CT ΔΣ Modulators



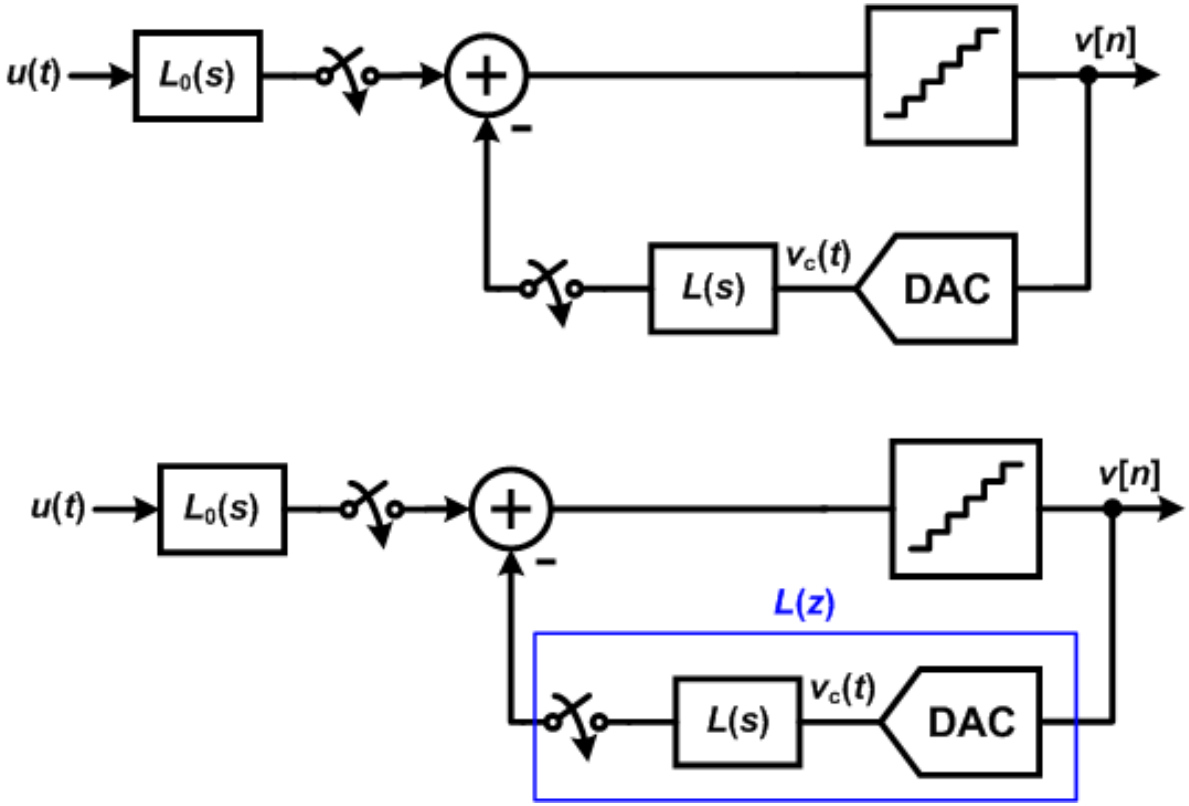
- Recall that the loop-filter may appear different to the input signal $u(t)$, and the feedback signal, $v_c(t)$.

Implicit Anti-Aliasing in CT ΔΣ Modulators



- Re-arrange the blocks
  - Recall that $L_1(s)=-L(s)$ due to the negative feedback inversion

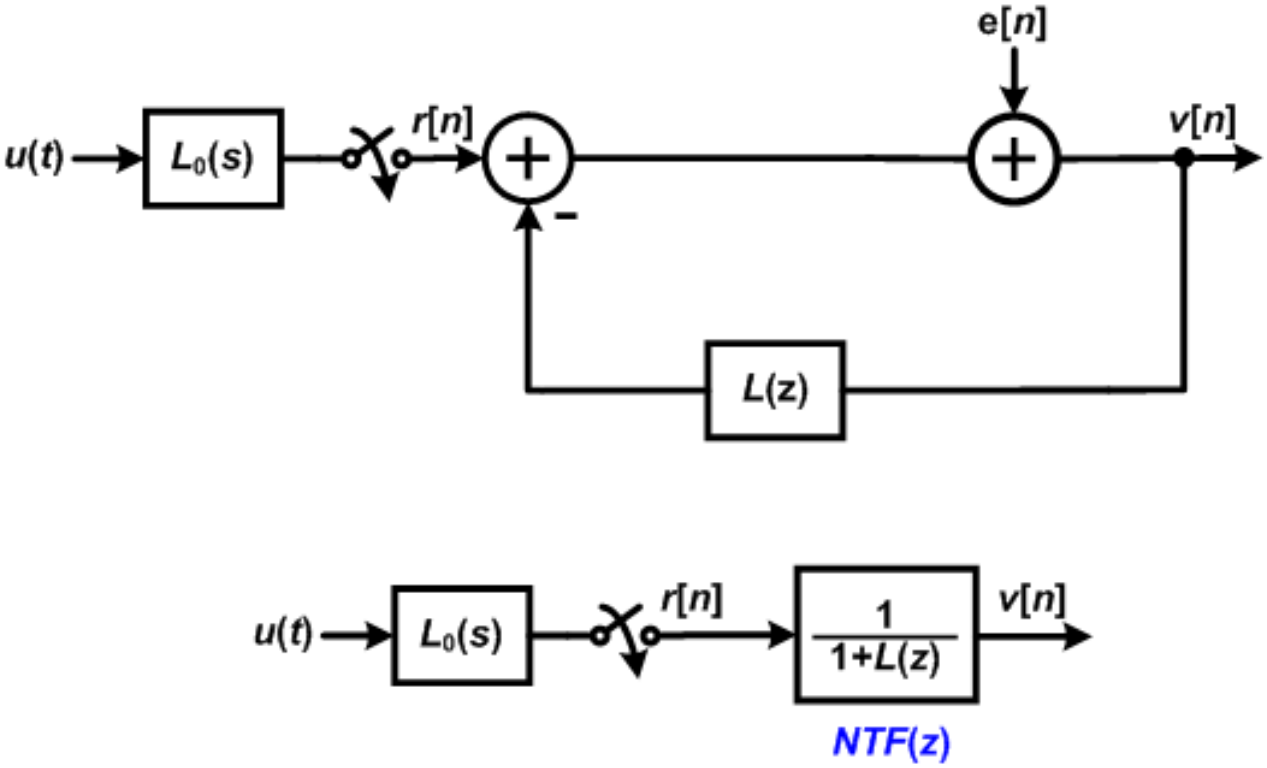
Implicit Anti-Aliasing in CT ΔΣ Modulators



- Move the sampler outside the loop
- Apply impulse-invariant transformation on the feedback path

$$\mathbf{Z}^{-1}\{L(z)\} = \mathbf{L}^{-1}\{P(s)\cdot L(s)\}|_{t=nT_s}$$

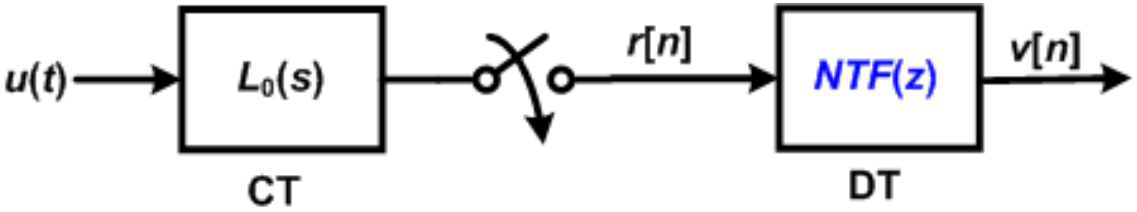
Implicit Anti-Aliasing in CT ΔΣ Modulators



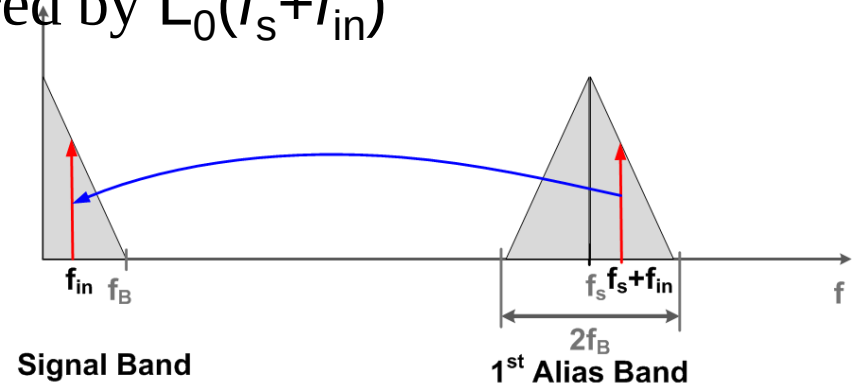
- Transfer function from *r* to *v* is given by

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

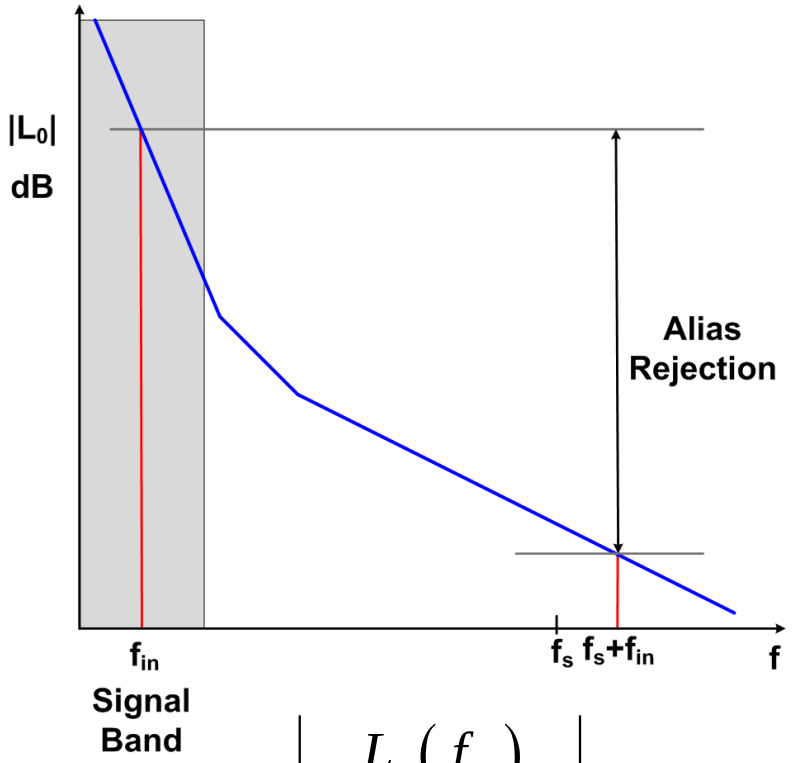
Implicit Anti-Aliasing in CT ΔΣ Modulators



- The input signal is pre-filtered by a CT LPF, $L_0(s)$, before sampling
- For an input tone at frequency f_{in} in the signal band
- Pre-filtered output before sampling is given by $L_0(f_{in})$
- Due to sampling, a tone at f_s+f_{in} can alias at f_{in}
  - i.e. the first alias-band $[f_s-f_B, f_s+f_B]$
- In CT ΔΣ, the alias tone is filtered by $L_0(f_s+f_{in})$

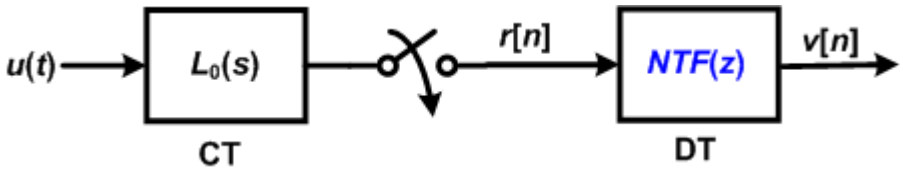


Implicit Anti-Aliasing in CT ΔΣ Modulators



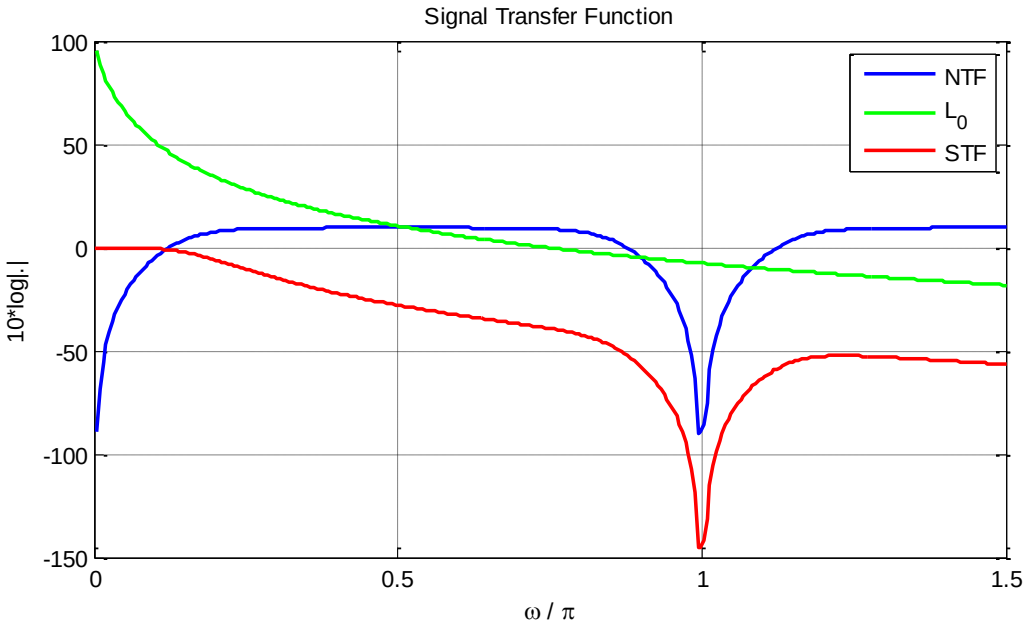
- Alias rejection is given by $\left| \frac{L_0(f_{in})}{L_0(f_s + f_{in})} \right|$
- Implicit anti-aliasing due to pre-filtering by L_0
  - Explicit AAF can be eliminated!
- Important distinguishing feature of CT ΔΣ

Implicit Anti-Aliasing in CT ΔΣ Modulators

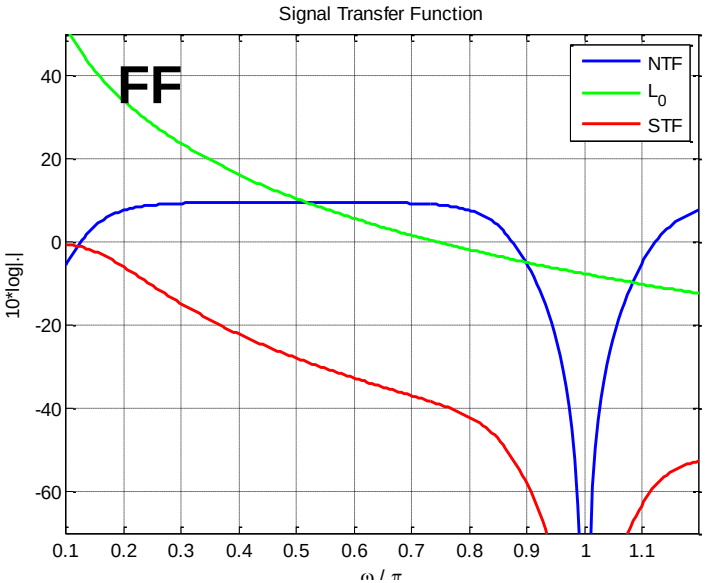


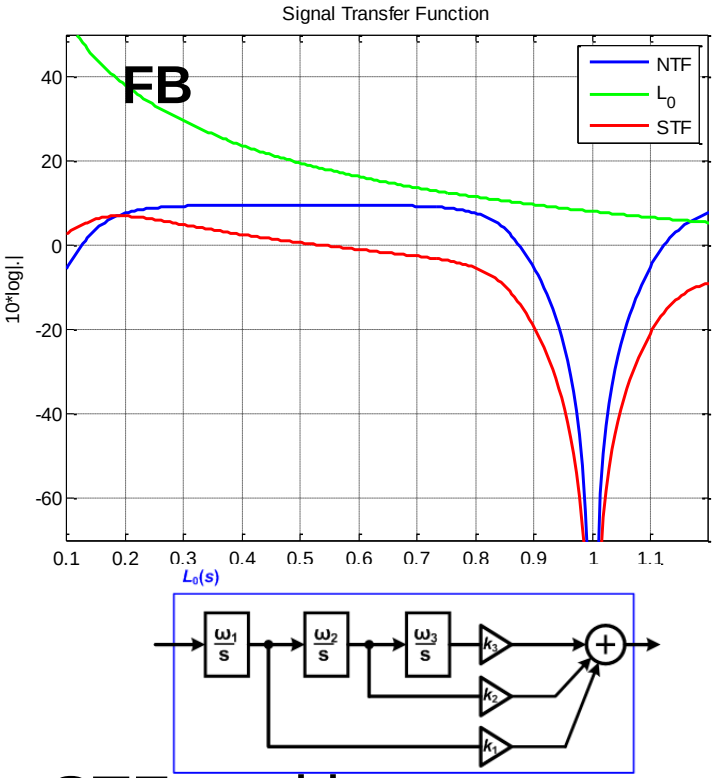
- Overall signal transfer function (STF) is given by

$$STF(j\omega) = L_0(j\omega) \cdot NTF\left(e^{j\omega}\right)$$



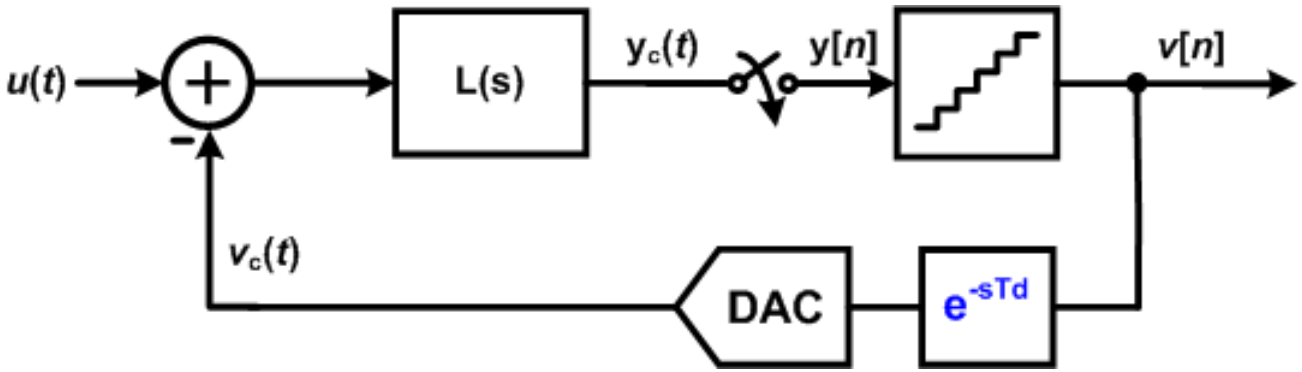
Implicit AAF – FF vs FB Loop-Filter





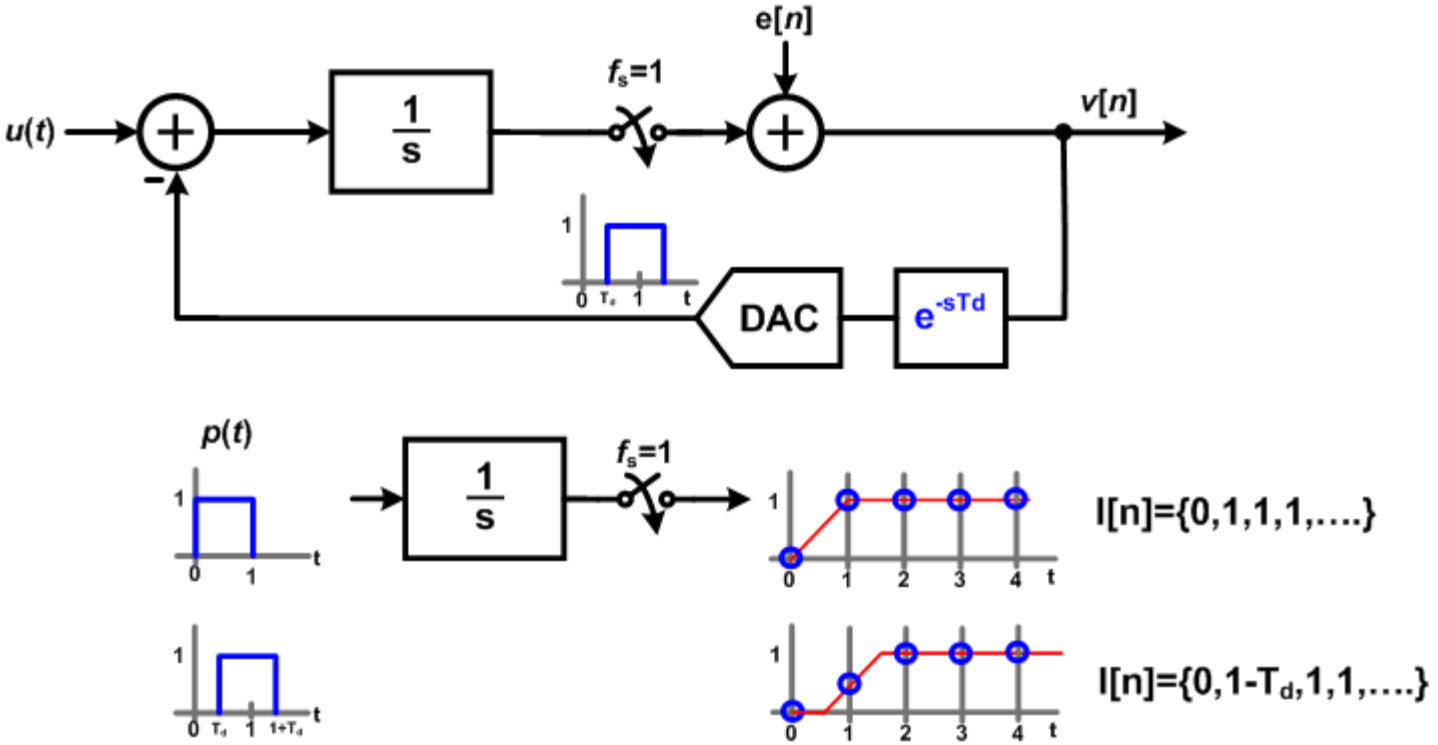
- Feedforward loop-filter exhibits STF peaking
- STF peaking due to zeroes in $L_0(s)$
  - Zeros at higher frequency flatten $|L_0(j\omega)|$ response
- For FB, $L_0(s)$ has all-pole form
  - Best STF performance (blocker rejection)

Excess Loop-Delay in CT ΔΣ



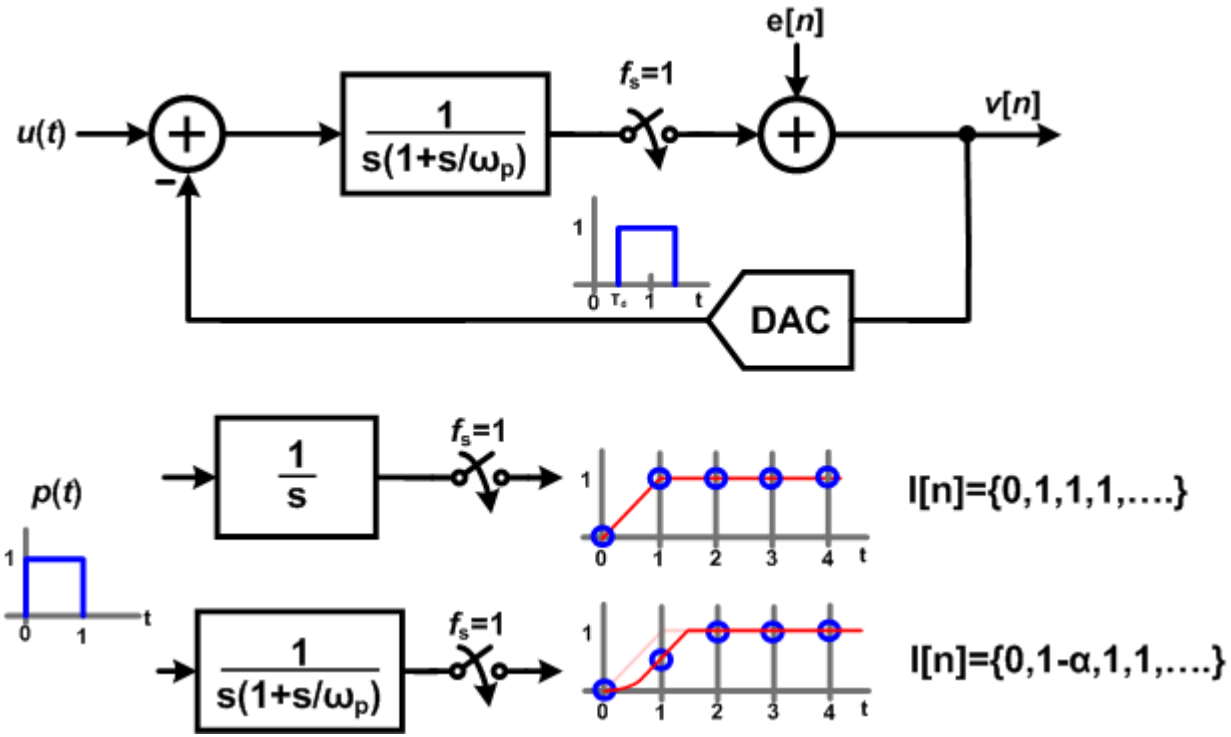
- Real circuits blocks introduce excess-loop delays (ELD)
  - Quantizer requires finite regeneration time
  - ELD due to finite gain-bandwidth of opamps in the loop-filter
  - DEM/DWA logic in multi-bit ΔΣ adds finite delay
  - Finite DAC rise/fall time (relatively small)
- Important concern in high-sampling rate ΔΣ converters
- What effect does ELD have on the loop-response?

ELD – First-order Example



- f_s=1 Hz, ELD due to quantizer delay: 0<T_d<1
- First sample after unit delay is affected (1-T_d)
- Resulting DT equivalent loop-response $L(z) = \frac{z^{-1}}{1-z^{-1}} - T_d z^{-1}$

ELD – First-order Example

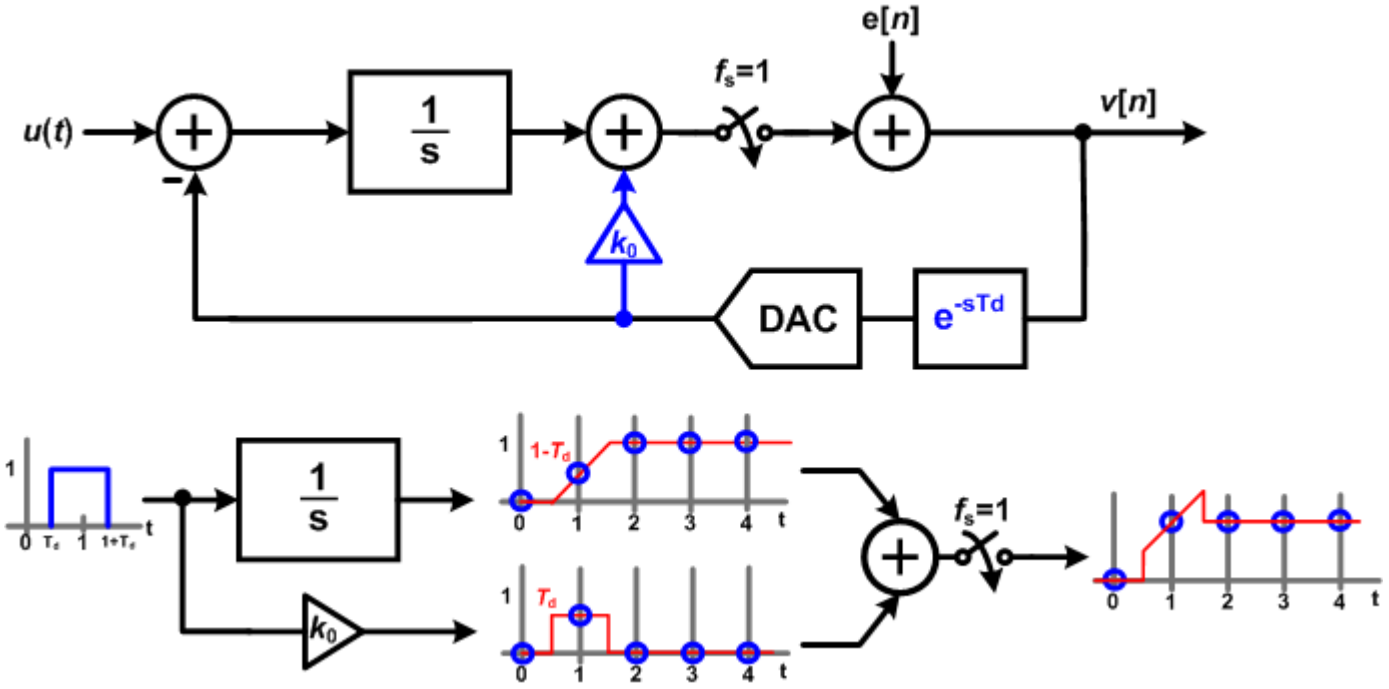


- f_s=1 Hz
- Sampled loop-response is delayed due to the finite gain-bandwidth of the opamp
- First sample after unit delay is affected (1-T_d)

Excess Loop-Delay in CT $\Delta\Sigma$

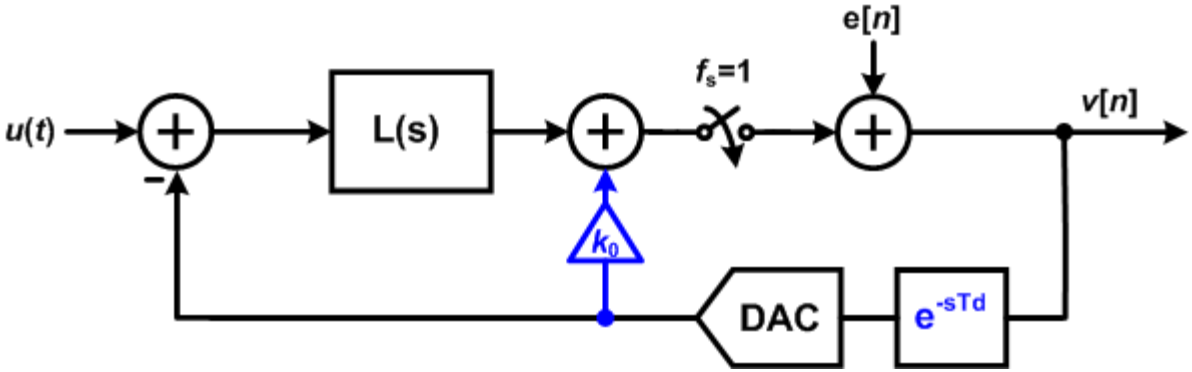
- NTF with ELD $$NTF(z) = \frac{\left(1 - z^{-1}\right)}{1 - T_d z^{-1} + T_d z^{-2}}$$
  - Order of the loop is increased due to the extra delay
  - Loop gets unstable at T_d=1
  - NTF peaking as T_d increases, leading to modulator becoming more sensitive and prone to instability

ELD Compensation– Basic Idea



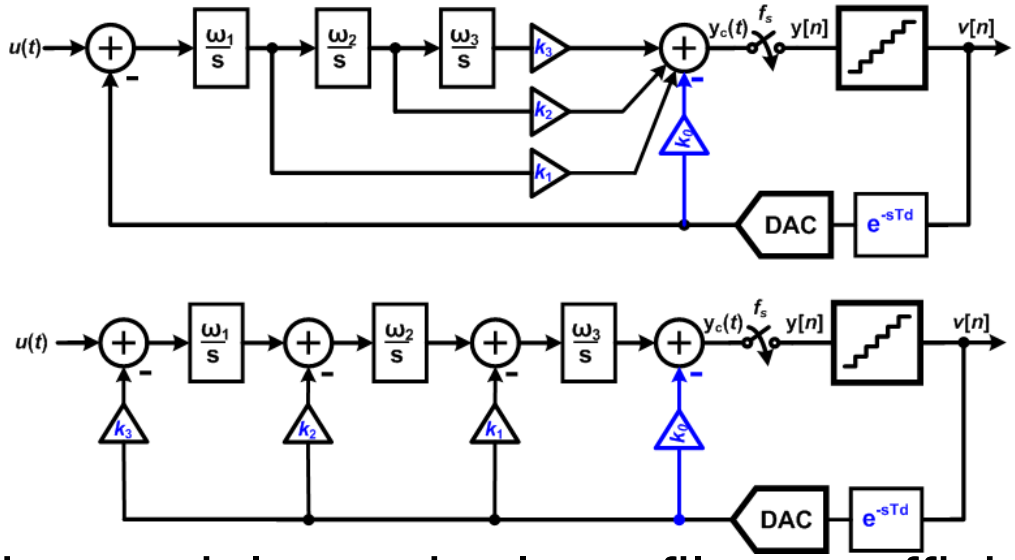
- Add a direct path (k_0) around the quantizer to provide the first sample $l[1]$
- Here, k_0=T_d restores L(z)
  - modulator NTF(z) is restored!
- Several other solutions are possible to restore L(z)

ELD Compensation



- Normalized ELD due to quantizer delay = $T_d/T_s=\tau$
- A direct path around the quantizer (k_0) is used to restore the DT loop-response L(z)
- Approach valid for higher-order modulators
  - For order>1, L(s) is also modified (requires coefficient tuning)
- Schreier's $\Delta\Sigma$ Toolbox function realizeNTF_ct
  - can synthesize CT loop-filters with ELD<1 and rectangular DAC pulse response.

ELD Compensation



- For n^{th} order modulator, the loop-filter coefficients also need to be tuned. For NRZ DAC:

$$k_0' = k_1\tau + \frac{k_1}{2!}\tau^2 + \ldots + \frac{k_n}{n!}\tau^n = \sum_{i=1}^{n}\frac{k_i}{i!}\tau^i$$

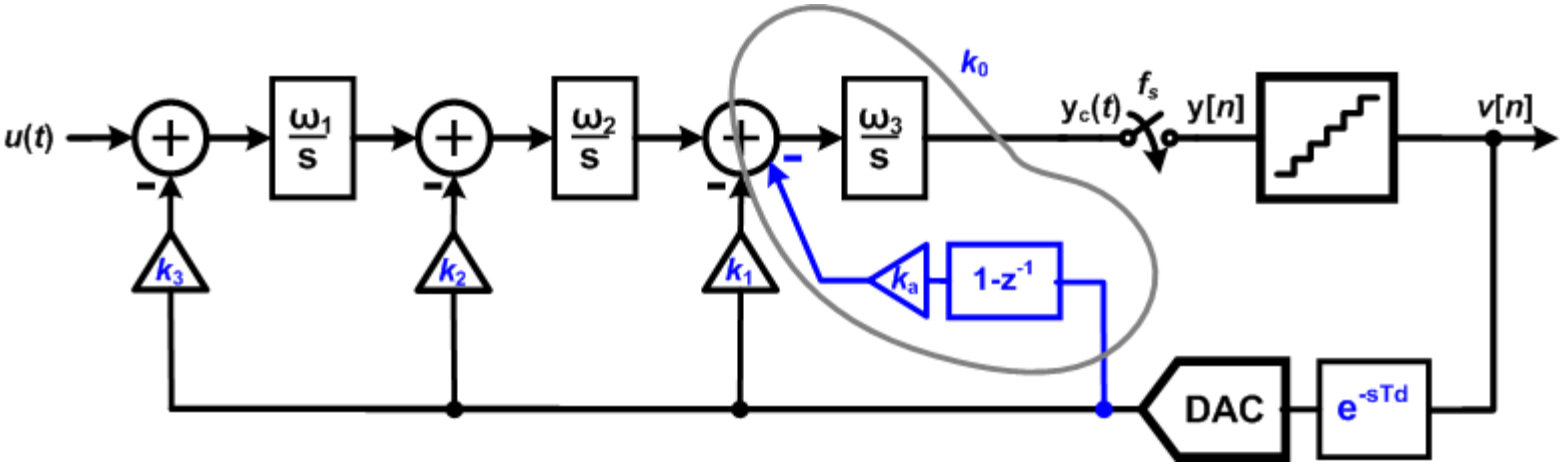
$$k_1' = k_1 + k_2\tau + \ldots + \frac{k_n}{n-1!}\tau^{n-1} = \sum_{i=1}^{n}\frac{k_i}{(i-1)!}\tau^i$$

$$\vdots$$

$$k_n' = k_n$$

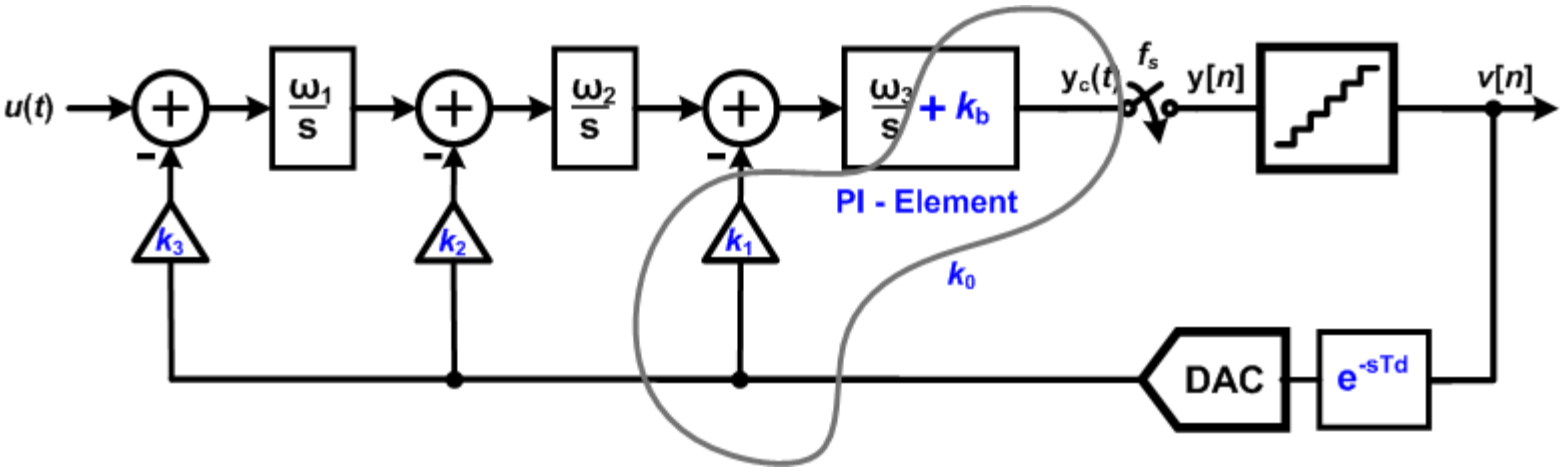
- Coefficient tuning is better done numerically.

ELD Comp. using DT Differentiator



- Proportional path = CT integrator + DT differentiator
- Avoids the extra summer at the expense of a register
- Scaling factor $k_a = k_0 / \omega_3$
- Other variations include
  - Using a $(1-z^{-0.5})$ differentiator, $k_a = k_0 / 2\omega_3$
  - Can tap first integrator output and perform CT differentiation

ELD Comp. using a PI-Element



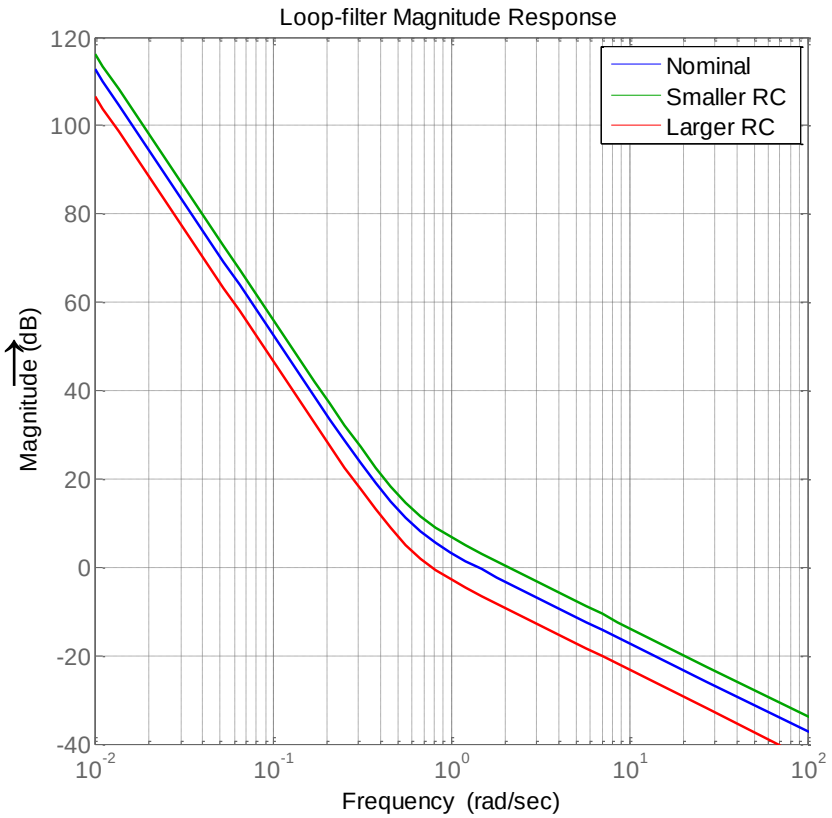
- Combine the two inner loops into a single PI-element
- Note that k_b affects other inner loops too
  - Other coefficients need to be tuned accordingly
- Can aggravate STF peaking due to additional zero

Effect of RC Variation

- CT-ΔΣ loop-filter coefficients k_i are implemented as $1/RC$ (or g_m/C)
- The RC time-constants can vary with process and temperature by
  - ±1% on a single die
  - ±20% with from die-to-die
- DT-ΔΣ employs implements k_i using ratio of capacitors
  - Much tightly controlled on-chip and hence accurate
- What is the impact of RC time-constant variation on CT-ΔΣ modulator performance?

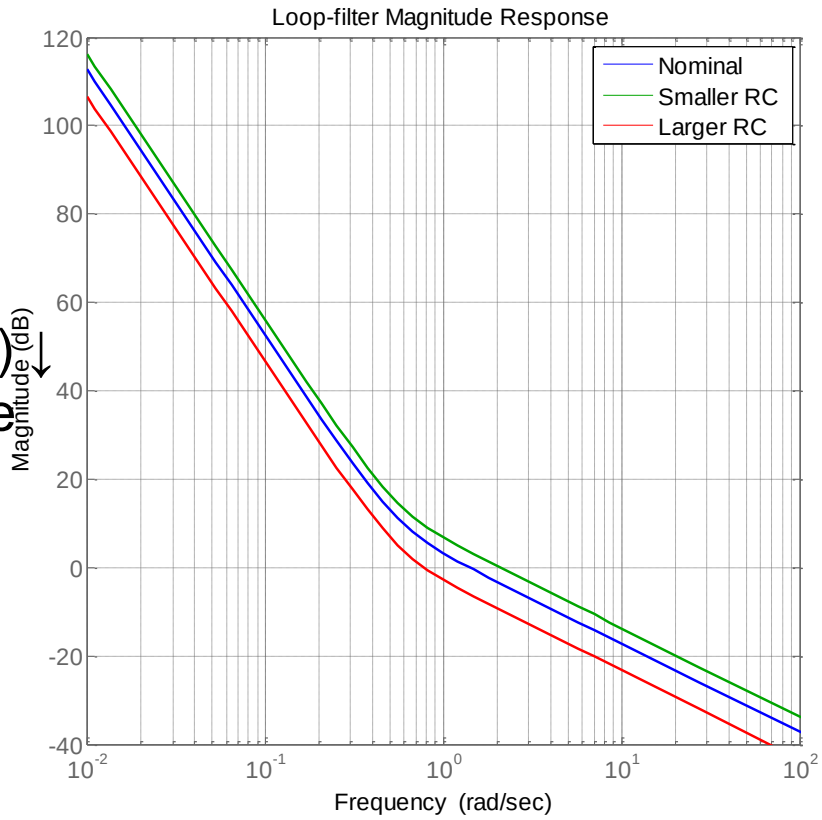
RC Time-Constant Variation

- With an increase in RC time-constant
  - Coefficients k_i decrease (↓)
  - Loop-filter bandwidth ↓
  - In-band loop-gain ↓
  - NTF OBG ↓
  - In-band quantization noise ↑
  - Maximum stable amplitude (MSA) ↑
  - "More" stable but degraded NTF performance
  - Slower design



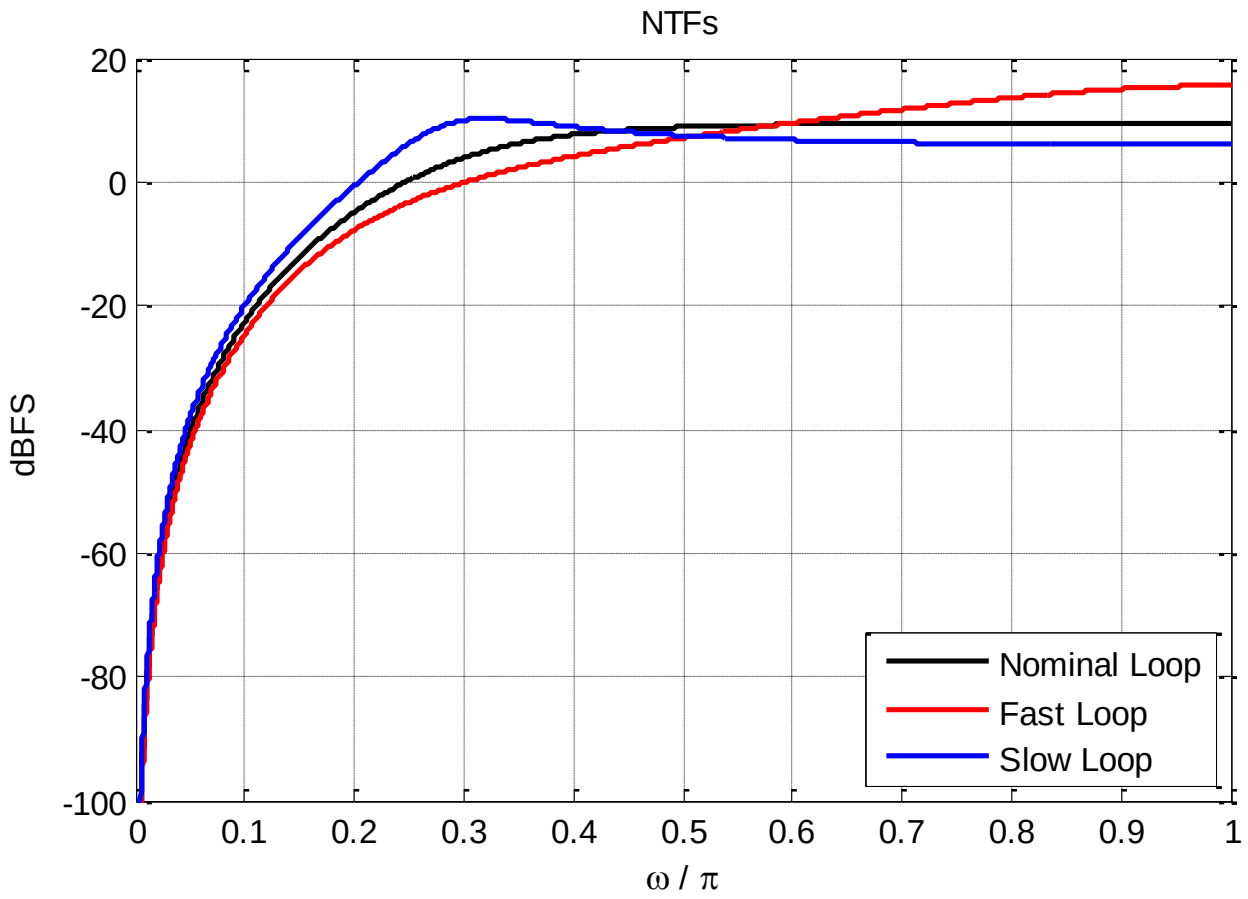
RC Time-Constant Variation

- With a decrease in RC time-constant
  - Coefficients k_i increase (↑)
  - Loop-filter bandwidth ↑
  - In-band loop-gain ↑
  - NTF OBG ↑
  - In-band quantization noise ↓
  - Maximum stable amplitude (MSA) ↓
  - "Less" stable but more aggressive NTF performance
  - Faster design



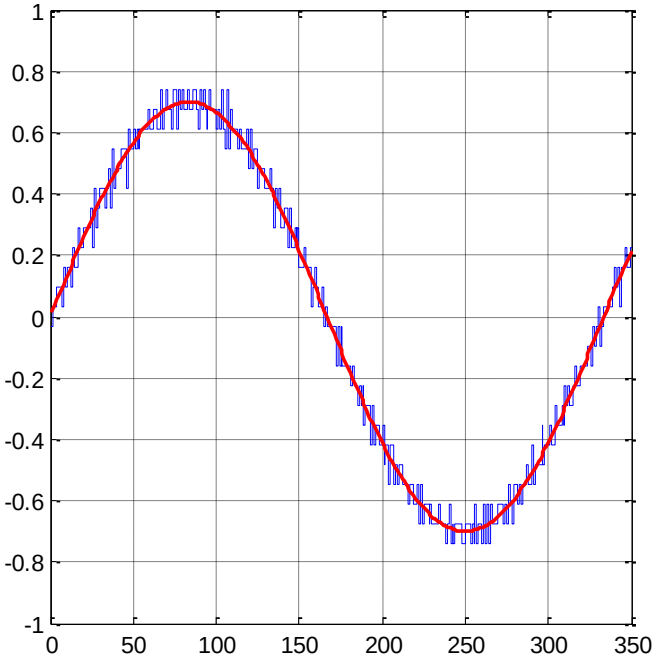
RC Time-Constant Variation

- Maximally flat 3rd-order NTF with nominal OBG=3
  - Modulators with ±30% RC variation

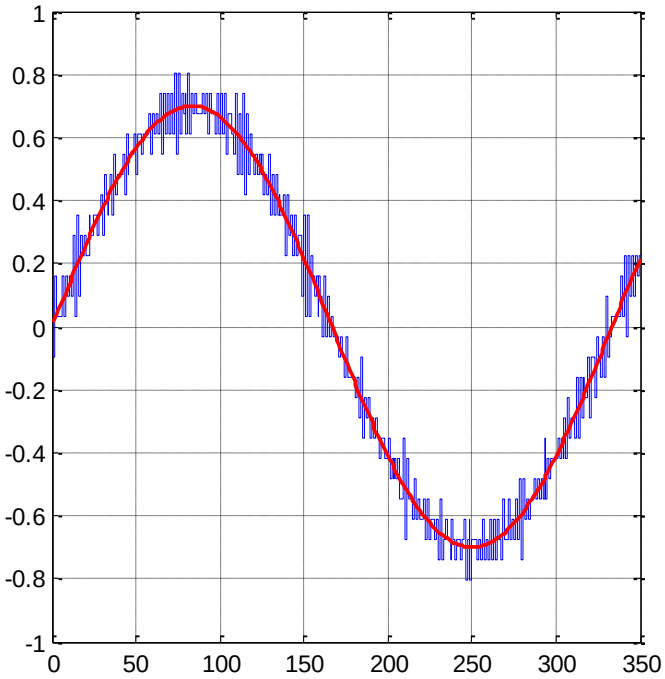


RC Time-Constant Variation

- Maximally flat 3rd-order NTF with nominal OBG=3, nLev=2^3
  - Modulators with ±30% RC variation

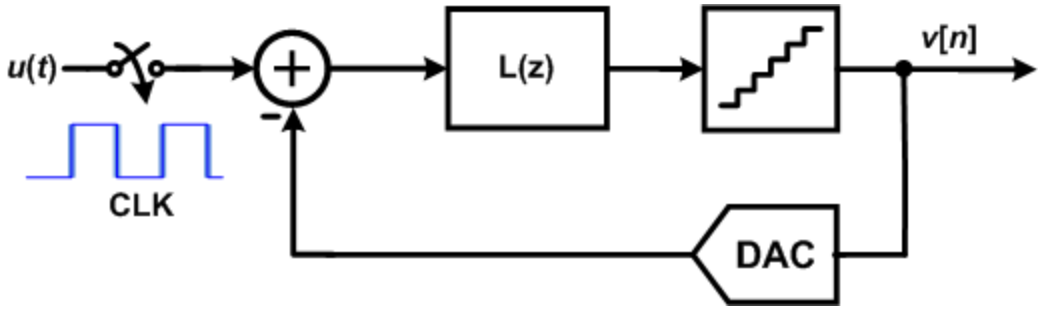


Nominal Loop



Fast Loop

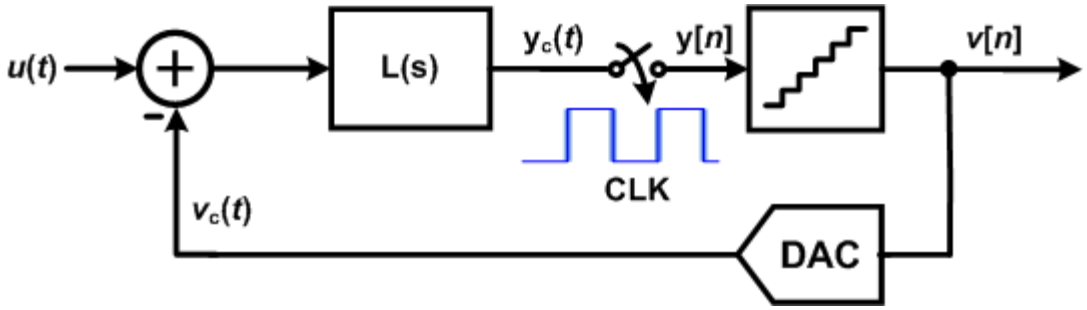
Clock Jitter in DT-ΔΣ



- DT-ΔΣ the input is sampled before the modulator loop
- Assume white clock jitter with RMS value σ_j
- RMS jitter noise in the signal band: $\frac{\sigma_j A\omega_{in}}{OSR\sqrt{2}}$
- Clock jitter limited SNR:

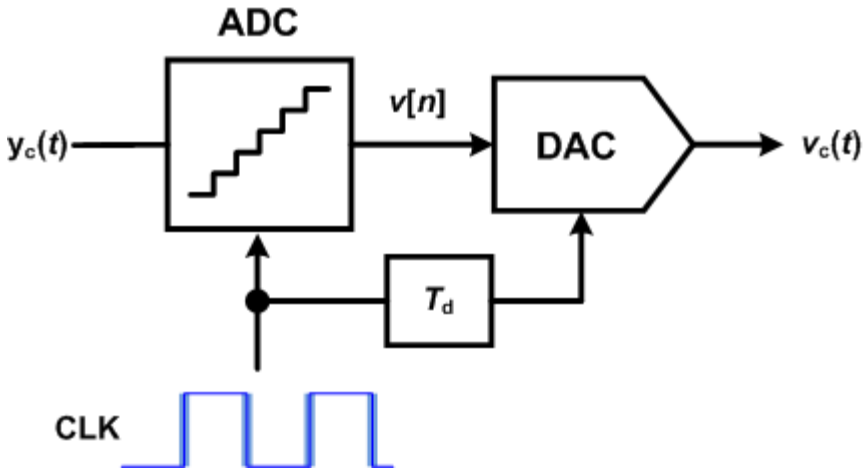
$$SNR_j = -20 \cdot \log_{10}\left(2\pi f_{in}\sigma_j OSR\right)$$

Clock Jitter in CT ΔΣ



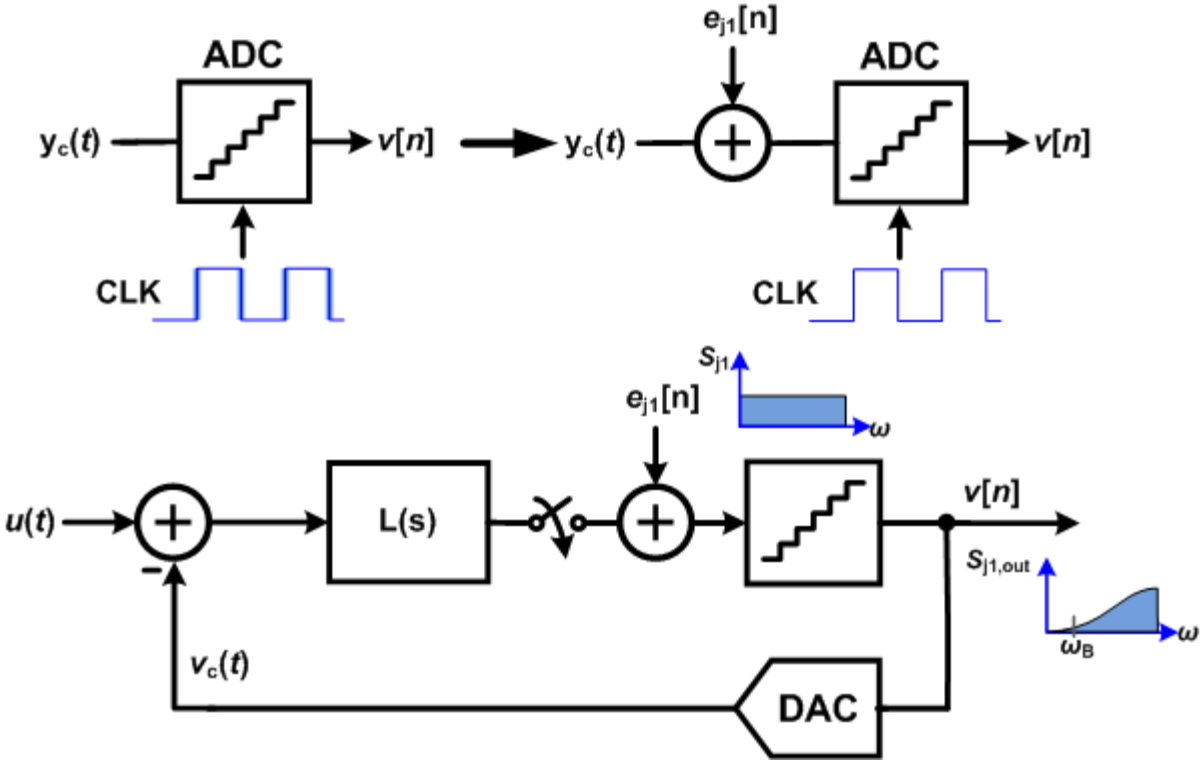
- The input signal is sampled inside the modulator loop
- Clock jitter affects both ADC decisions as well as the DAC output

Real Quantizer with Jittery Clock



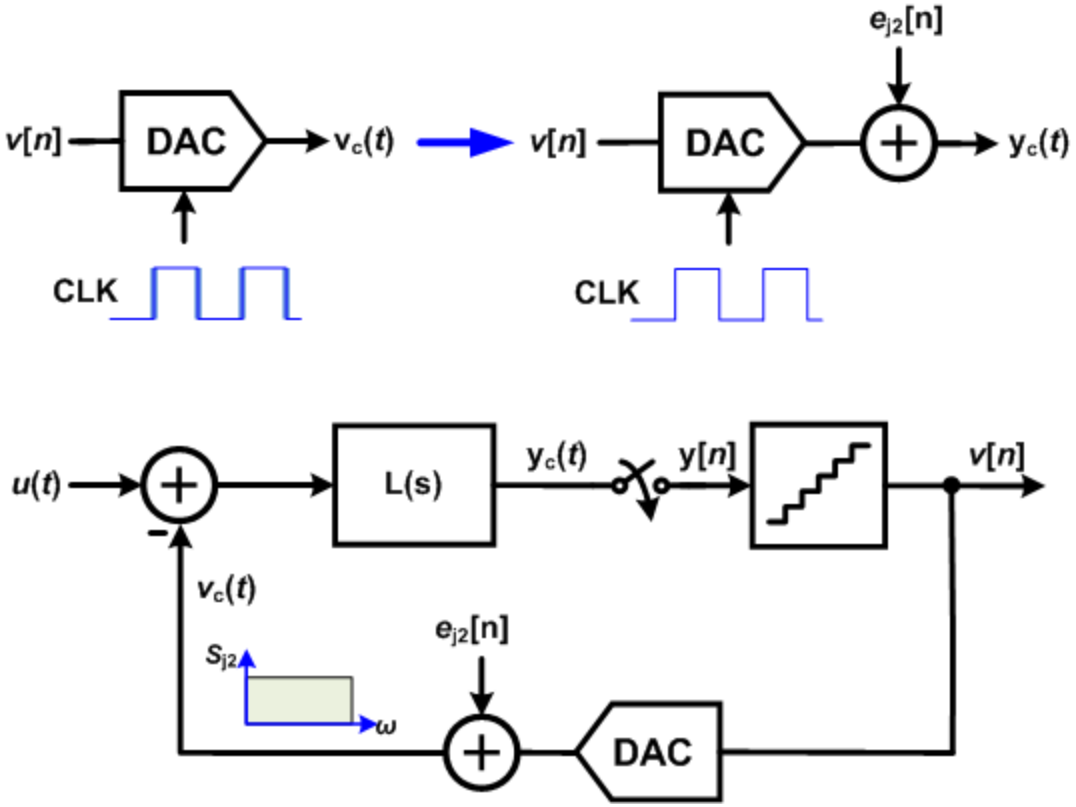
- ADC samples and quantizes CT input y_c digital output $v[n]$
  - ADC regeneration time - T_d
- DAC reconstructs samples of v[n] into CT output – v_c
- DAC is clocked with a delay to capture the ADC output
- Both ADC and DAC clocks have jitter

ADC Sampling Jitter



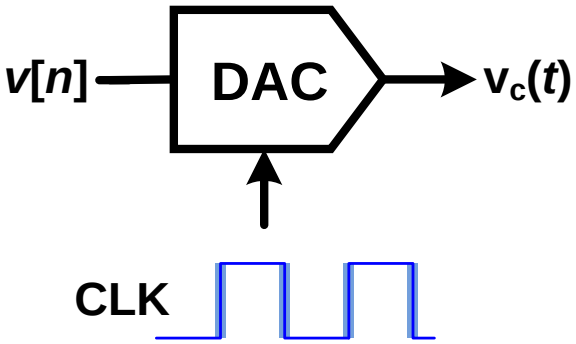
- ADC sampling jitter is modeled as AWGN
- Noise due to sampling jitter is shaped by the loop (NTF)

DAC Reconstruction Jitter

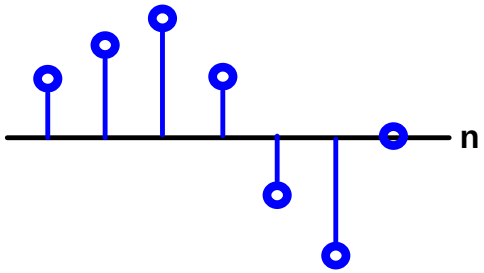


- DAC reconstruction error (e_{j2}) follows the DAC
- Directly adds to the input
- Major contributor of jitter induced noise

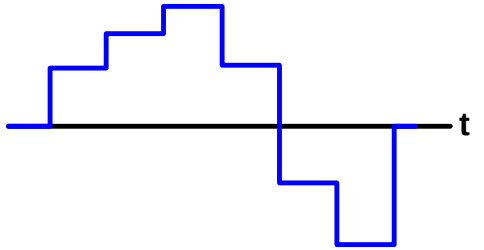
DAC Output Waveforms



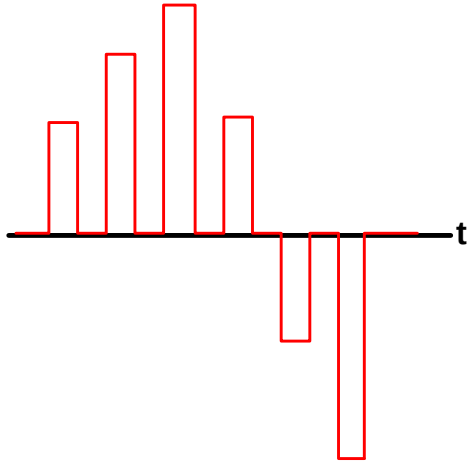
$v[n]$



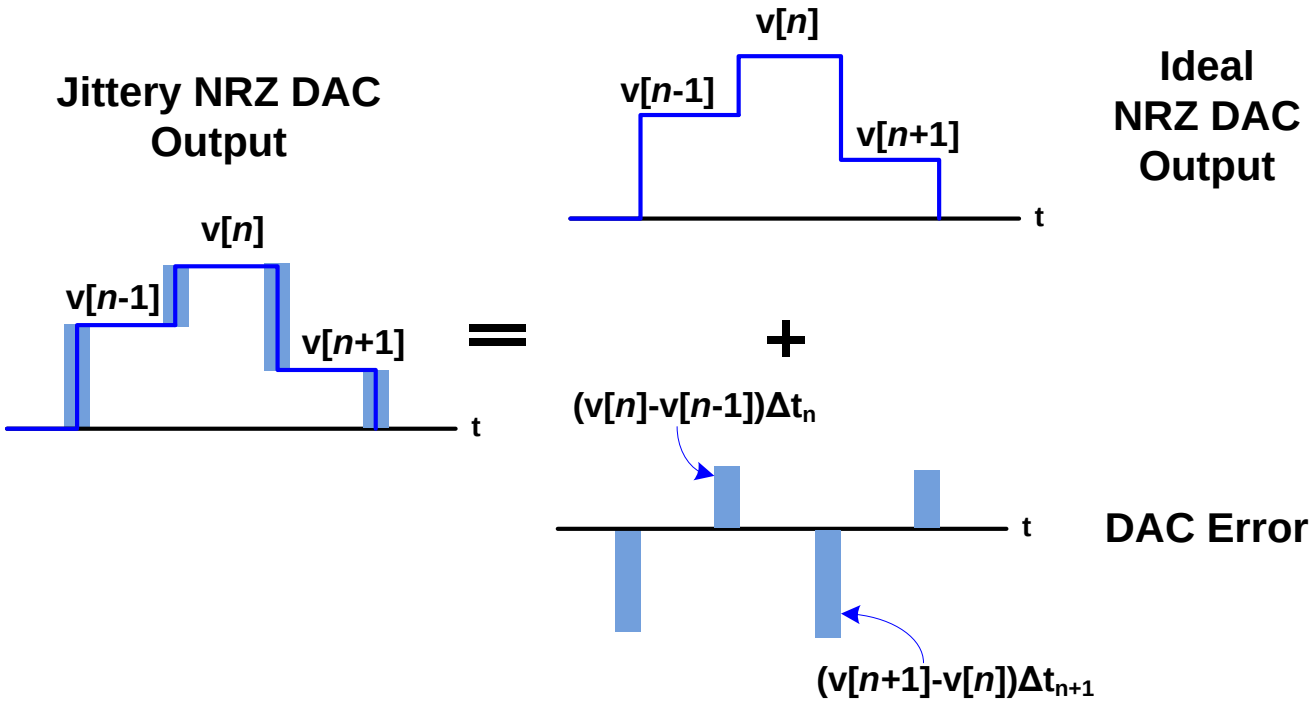
$v_c(t)$
NRZ DAC



$v_c(t)$
RZ DAC

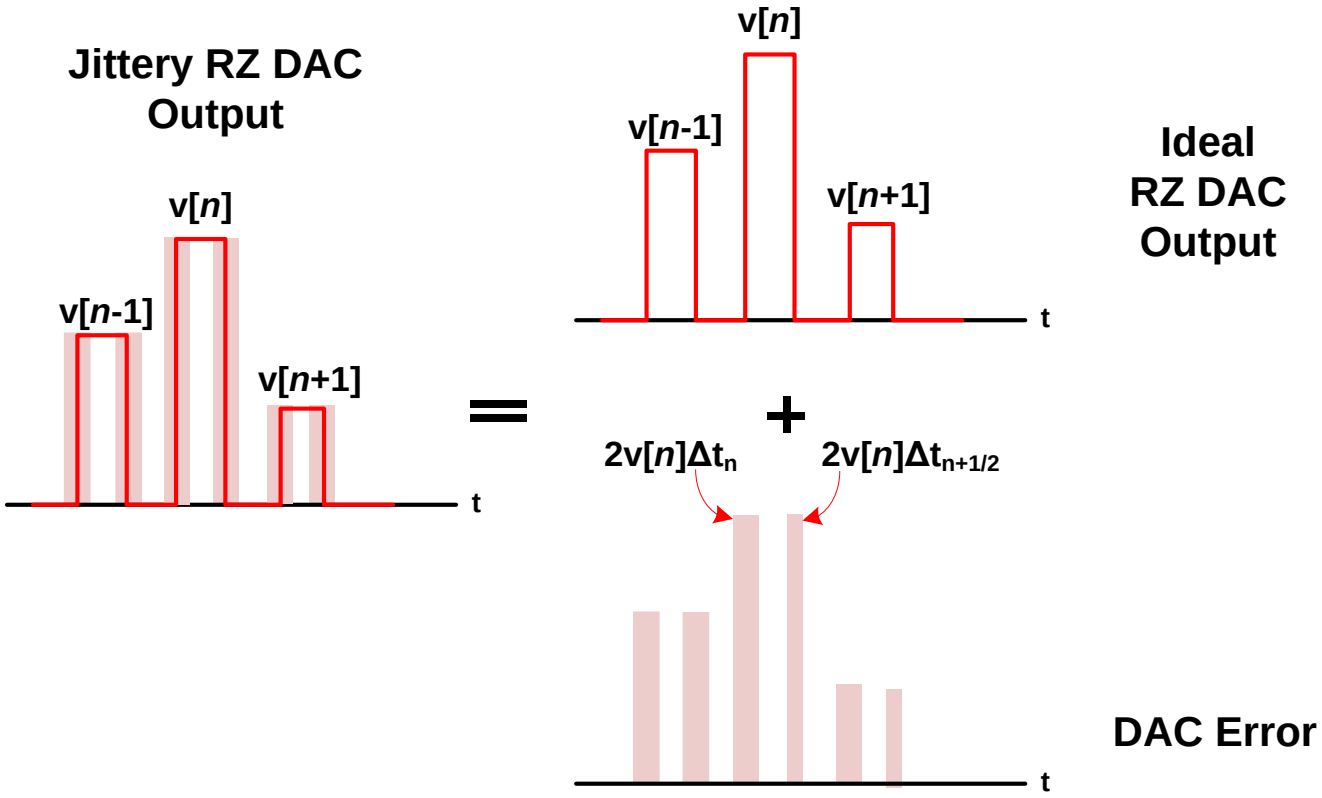


Jitter Induced DAC Error- NRZ



- Error depends on the transition height v[*n*]-v[*n*-1]
  - Granularity of LSB size
- Multi-bit NRZ DACs have lower jitter induced error

Jitter Induced DAC Error- RZ



- Error depends on twice the pulse height 2v[*n*]
  - Two transitions per time period
- RZ DACs are much more sensitive to clock jitter

Jitter Noise with NRZ DAC

- Error due to jitter $e_j[n] = (v[n] - v[n-1])\frac{\Delta t_n}{T} = \delta v[n]\frac{\Delta t_n}{T}$

$$\sigma_{ej}^2 = \sigma_{\delta v}^2 \cdot \frac{\sigma_{\Delta t}^2}{T_s^2}$$

- Modulator output variance (neglecting slowly varying input)

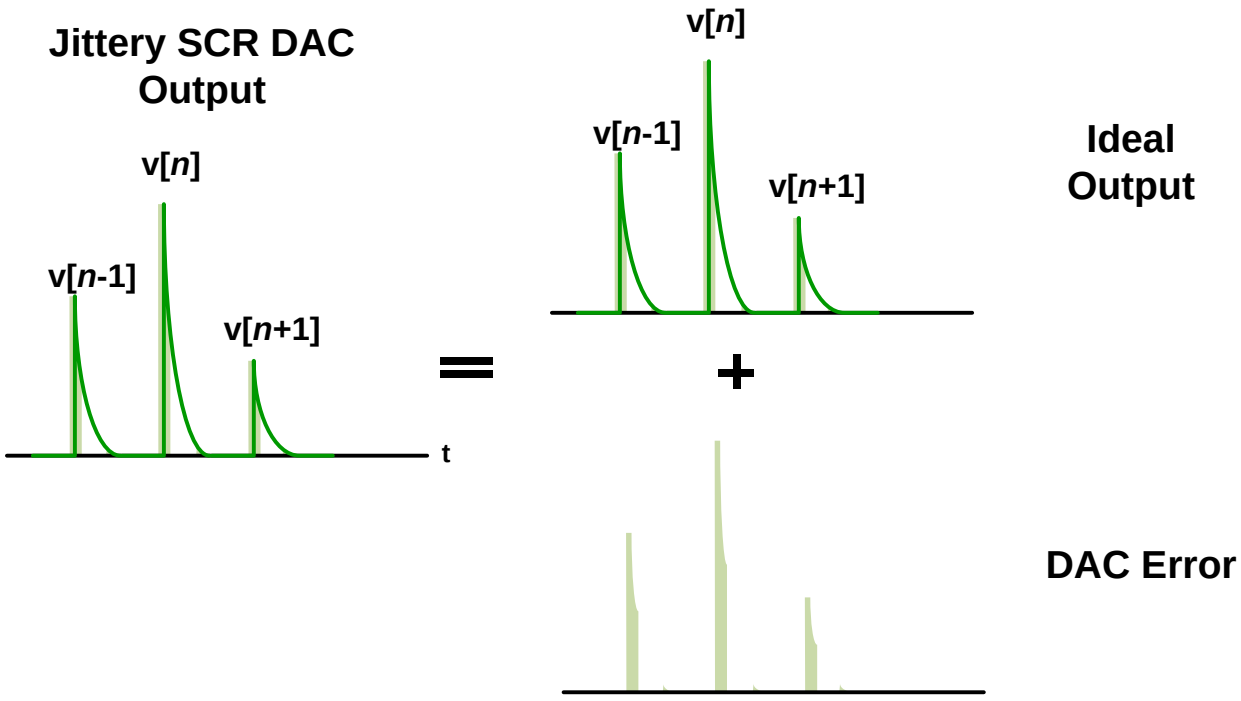
$$\sigma_{\delta v}^2 \approx \left(\frac{\sigma_{LSB}^2}{\pi}\right)\int_0^{\pi}\left|\left(1-e^{-j\omega}\right)NTF(e^{j\omega})\right|^2 d\omega$$

- In-band jitter noise

$$J \approx \frac{\sigma_{\Delta T_s}^2}{T_s^2}\cdot\frac{\sigma_{LSB}^2}{\pi \cdot OSR}\int_0^{\pi}\left|\left(1-e^{-j\omega}\right)NTF(e^{j\omega})\right|^2 d\omega$$

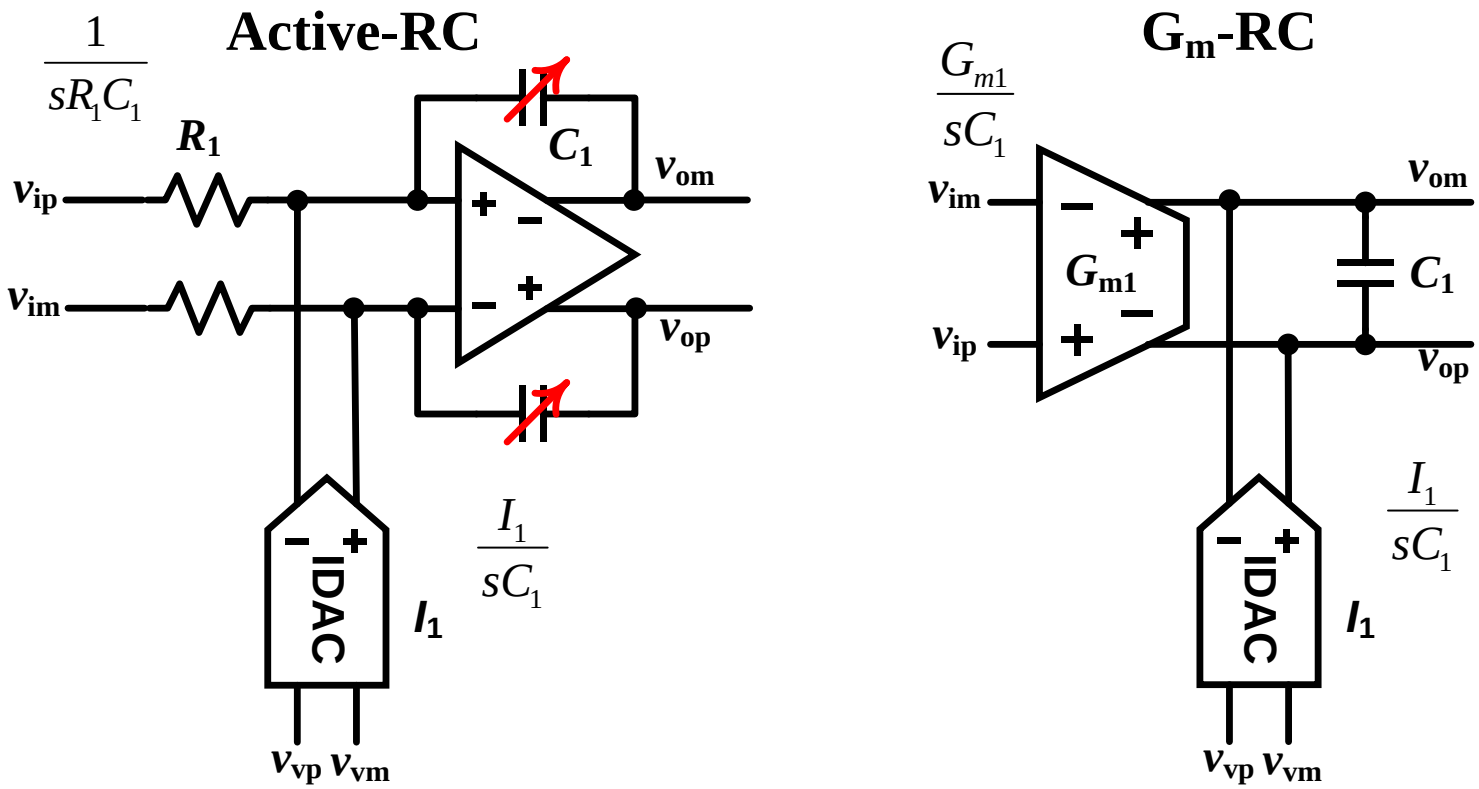
  - depends upon the area of differentiated NTF response
- NTF behavior at higher frequencies determines most of *J*
- To reduce J: OBG↓, V_{LSB} ↓, $\frac{\sigma_{\Delta T_s}}{T_s}$ ↓ , OSR↑

SCR DAC for Lower Jitter Noise



- Use sloping DACs to reduce the error due to jitter
  - Switched-cap resistor DAC (SCR), cosine-shape, etc.
- More stringent requirements on the opamp performance
  - Possible linearity and AAF degradation

CT ΔΣ Implementation



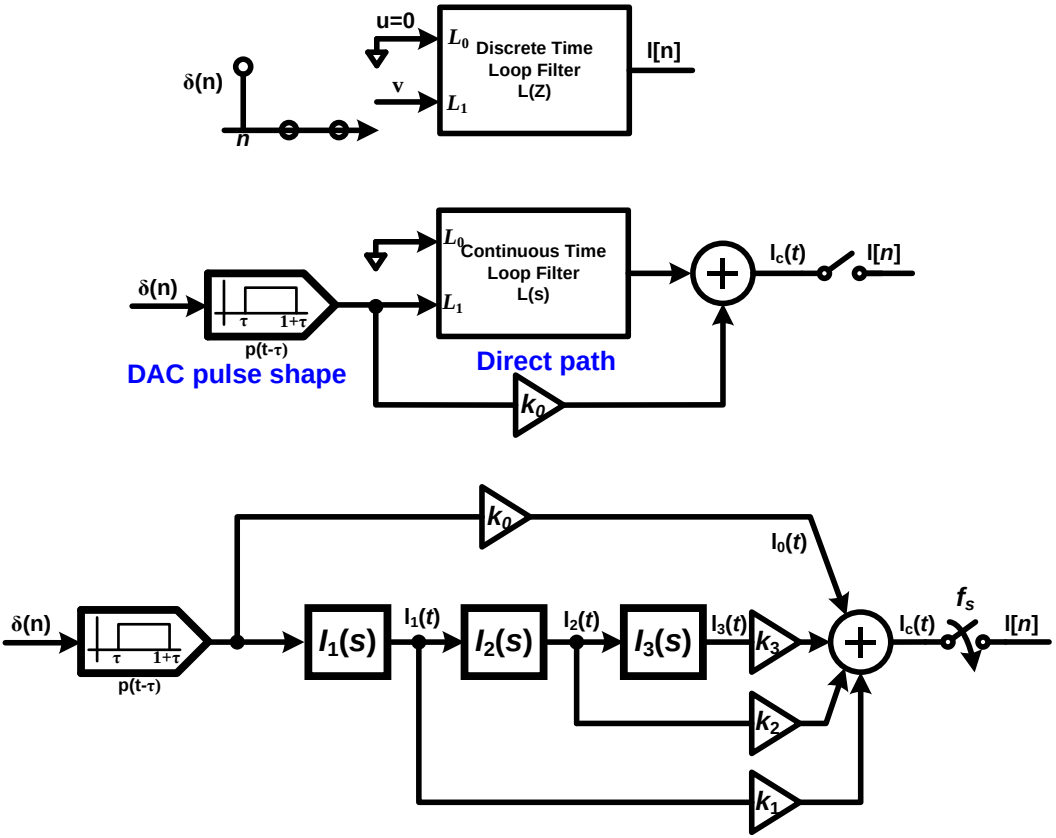
- CT Integrators used as the loop-filter stages
  - **Active-RC, G_m-C**, Active G_m-C, Active-MOS-C, Current-mode integrators, log-domain integrators
  - Active-passive Hybrid stages can alternate

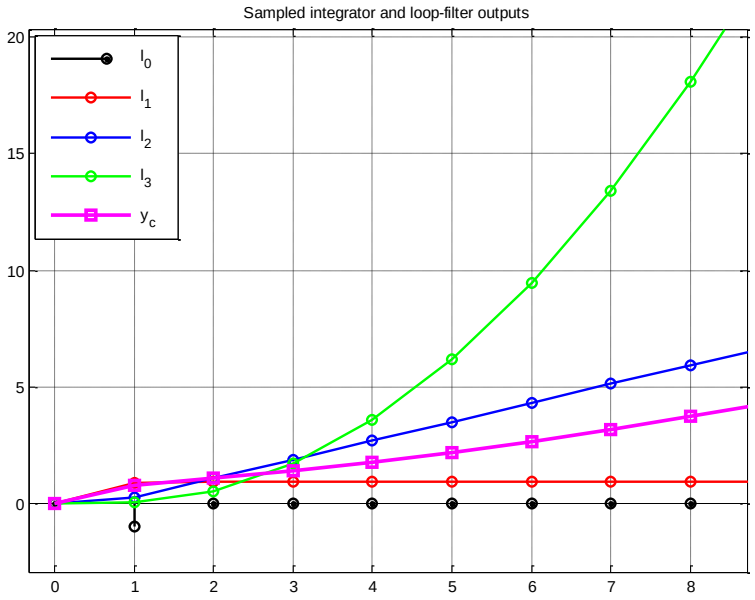
CT ΔΣ Implementation

	Active-RC	G_m-C
Linearity	✓	✗
Power consumption	✗	✓
Frequency range	✗	✓
Tunability	✗	✓
Dynamic Range	✓	✗
Voltage headroom	✓	✗
Parasitic Sensitivity	✓	✗
Realization of FF Summation	✗	✓

- Overall Active-RC preferred for superior dynamic range in CT ΔΣ designs
- Active-RC with two- or higher stage opamp for better linearity

Loop-filter Coefficient Tuning





Loop-filter Coefficient Tuning

- Loop-filter coefficients are typically determined using the Schreier's ΔΣ Toolbox
  - using impulse-invariance transformation tables is unwieldy for higher-order modulators and with ELD

Algorithm

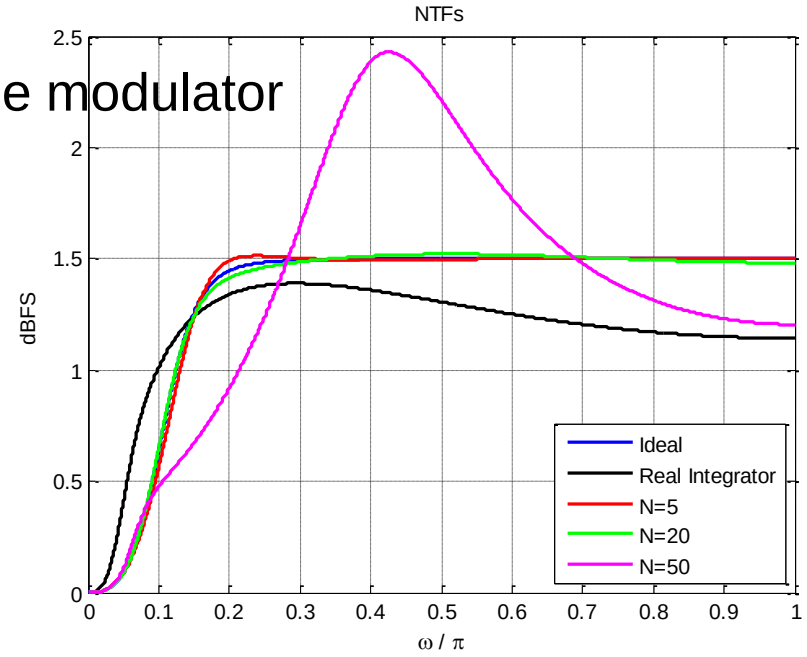
- If the sampled outputs of the direct path and the integrators are given by $l_0[n]$, $l_1[n]$, $l_2[n]$ and $l_3[n]$, and the open-loop impulse response is $l[n]$. Here, $Z(l[n]) = L(z) = \frac{1}{NTF(z)} - 1$.
- The coefficients $\mathbf{K} = [\ k_0 \quad k_1 \quad k_2 \quad k_3\ ]^T$ are determined by solving

$$[\ l_0[n] \quad l_1[n] \quad l_2[n] \quad l_3[n]\ ]\,\mathbf{K} = l[n]$$

  - Solved using LMS data fitting for *N* samples (pseudo-inverse)

Problems with Coefficient Tuning

- Practical integrators are implemented using opamps
  - finite opamp gain (A_{OL}) and unity-gain bandwidth (f_{un}), and with extra poles and zeros.
- The excess loop-delay due to finite f_{un} causes significant amount of gain peaking in the resulting NTF.
- Coefficient tuning yields different results depending on the number of samples used for LMS fitting.
  - tuning is numerically unstable
  - higher OBG leads to instability in the modulator



Systematic Design Centering

- Instead of fitting the open-loop response, fit NTF(z)(1+L(z)) to 1

$$\Rightarrow h[n] + h[n] \otimes l[n] = \delta[n] \qquad h[n] = Z^{-1}(NTF(z))$$

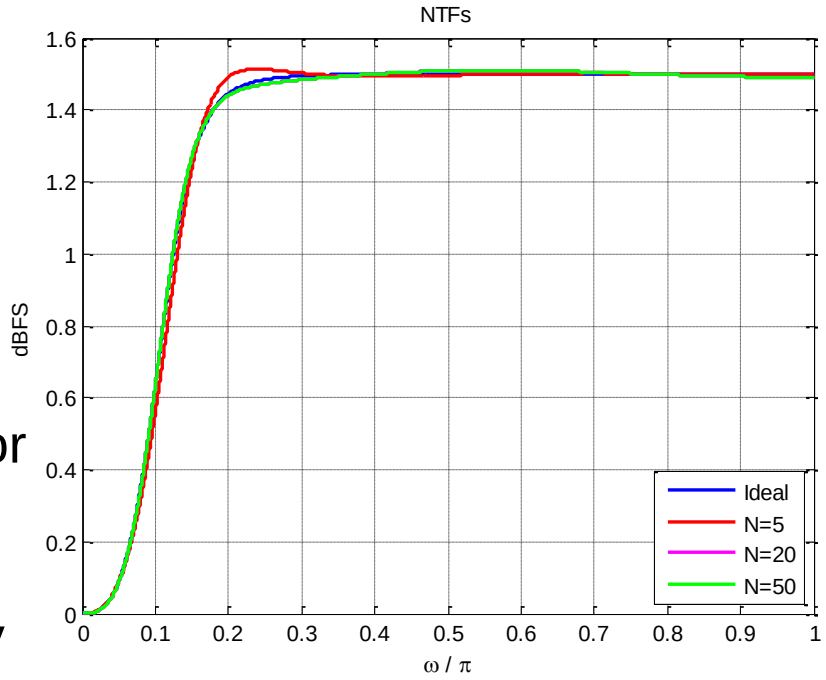
Algorithm

- Let $h_i[n] = l_i[n] \otimes h[n]$, for $i = 0, 1, .., 3$
- The coefficients $\mathbf{K} = [\ k_0 \quad k_1 \quad k_2 \quad k_3\ ]^T$ are determined by solving

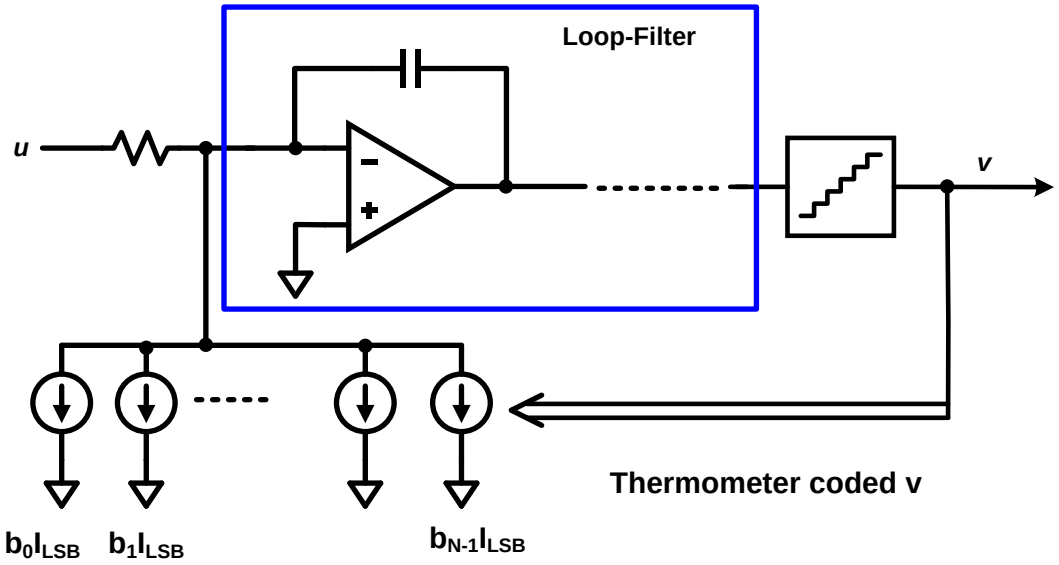
$$[\ h_0 \quad h_1 \quad h_2 \quad h_3\ ]\,\mathbf{K} = \delta[n] - h[n]$$

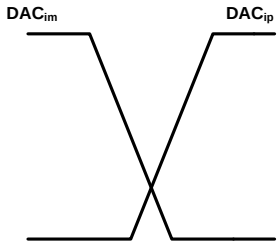
Systematic Design Centering

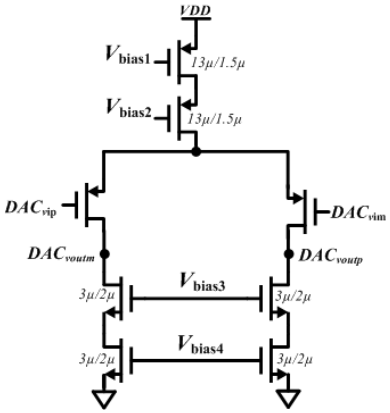
- The loop-filter coefficients are tuned to compensate for the excess loop delay due to the op-amps and the quantizer delay
  - NTF response with non-ideal integrators is close to the ideal NTF
- The NTFs are indistinguishable for any value of N=5,20,50
  - Coefficient tuning is numerically stable



Feedback DAC Architecture

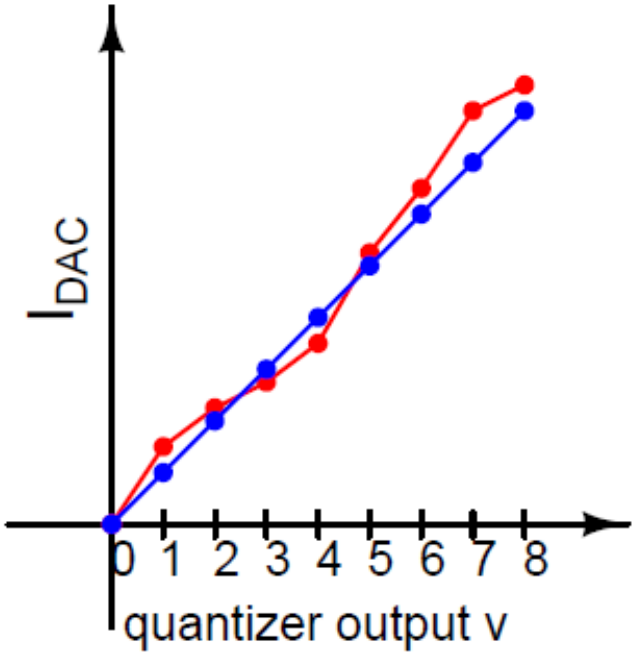


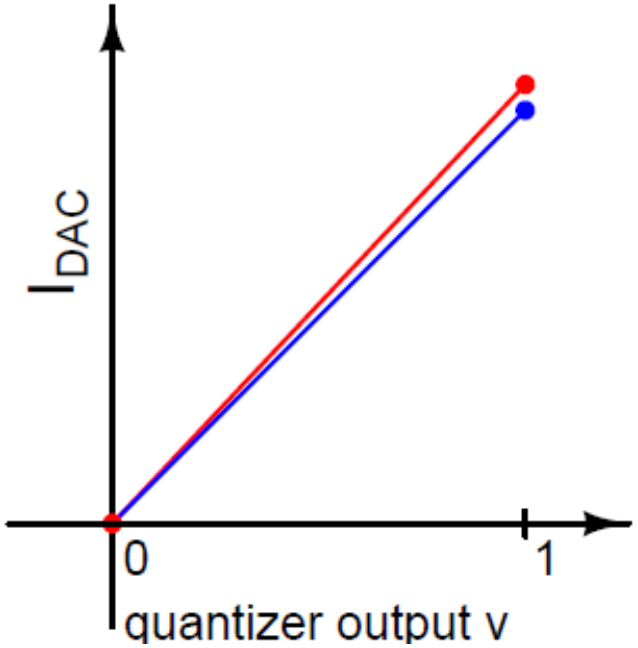




- Array of unit-weighted current steering DACs to pull and push current from the opamp current summing node
- Use high crossover pre-drivers to reduce DAC glitching noise

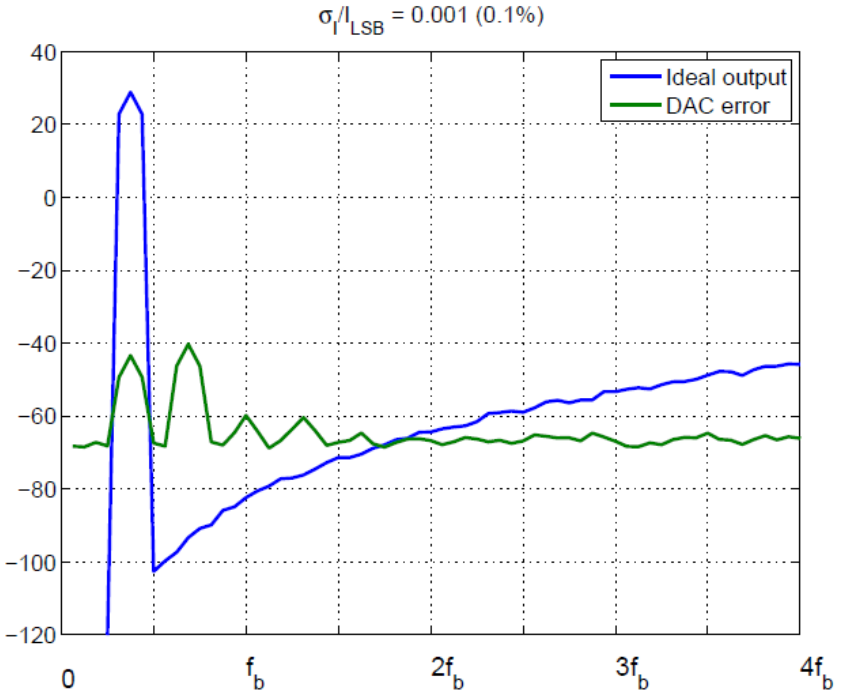
Feedback DAC Nonlinearity





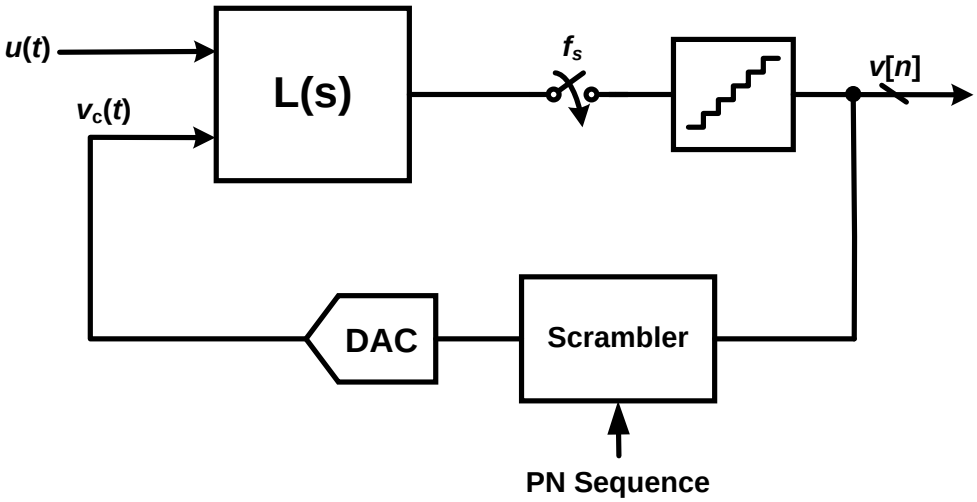
- In a multibit DAC, element mismatch leads to non-linearity
  - *v* is related to the input (*u*) by inverse non-linearity of the DAC
- A single-bit DAC is always linear

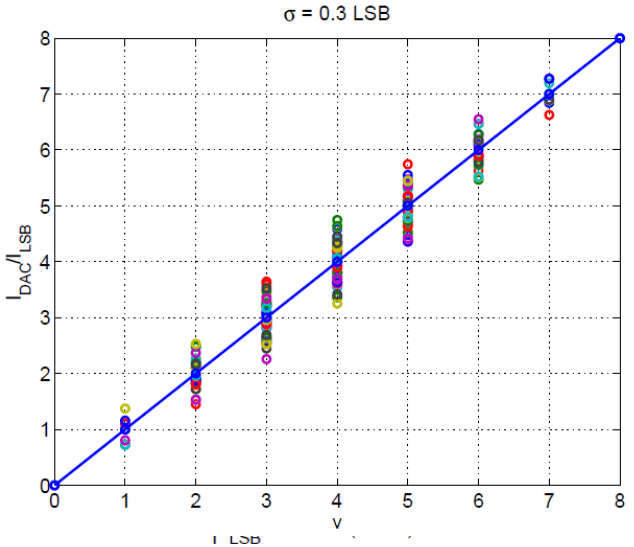
Feedback DAC Nonlinearity

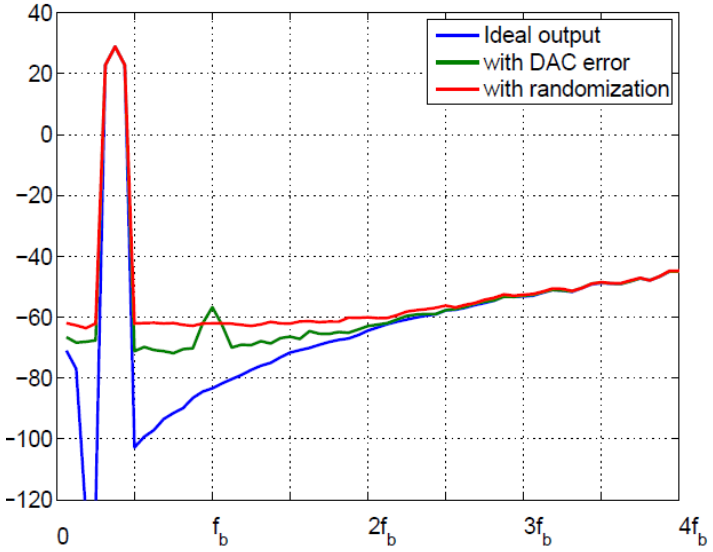


- Leads to distortion
- intermodulation of quantization noise into the signal band

Dynamic Element Matching (DEM)



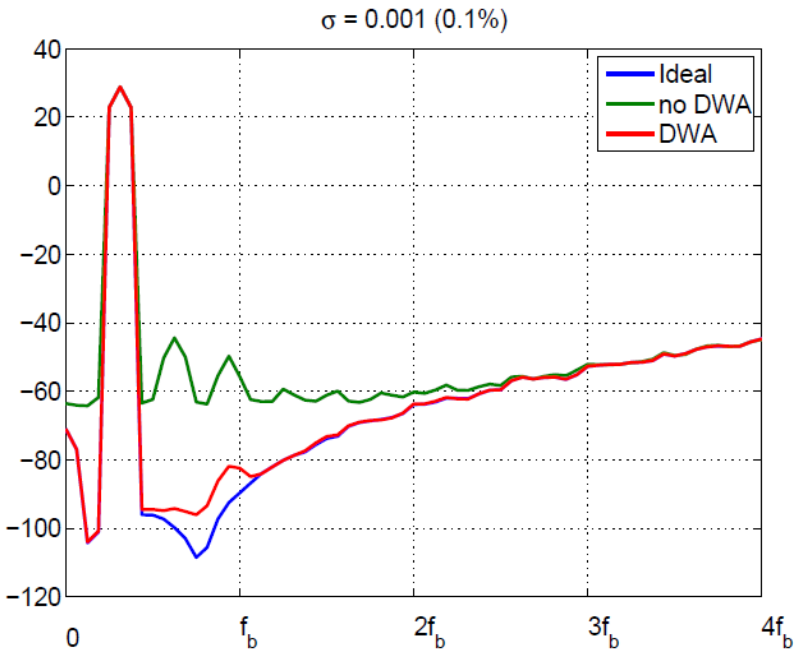


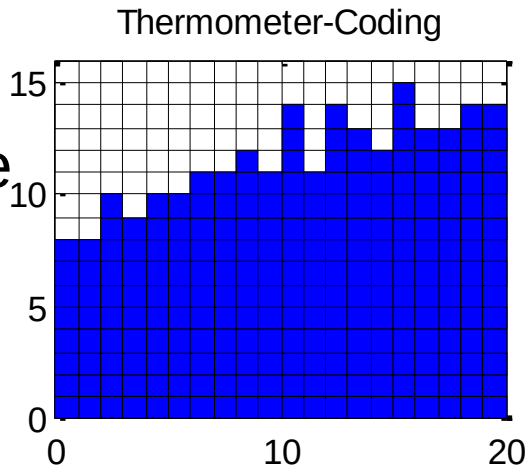


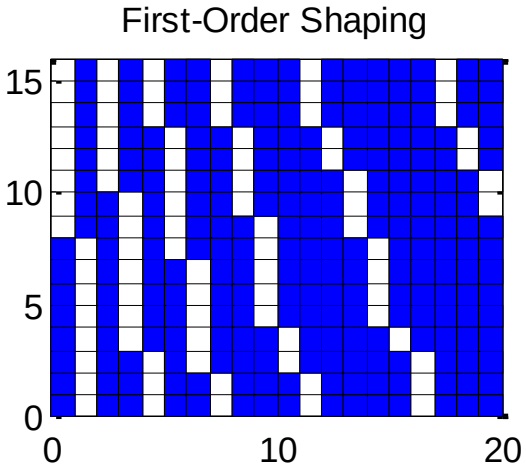
- Randomize the DAC elements
- Distortion components converted to noise
- Increased noise floor

Data Weighted Averaging (DWA)

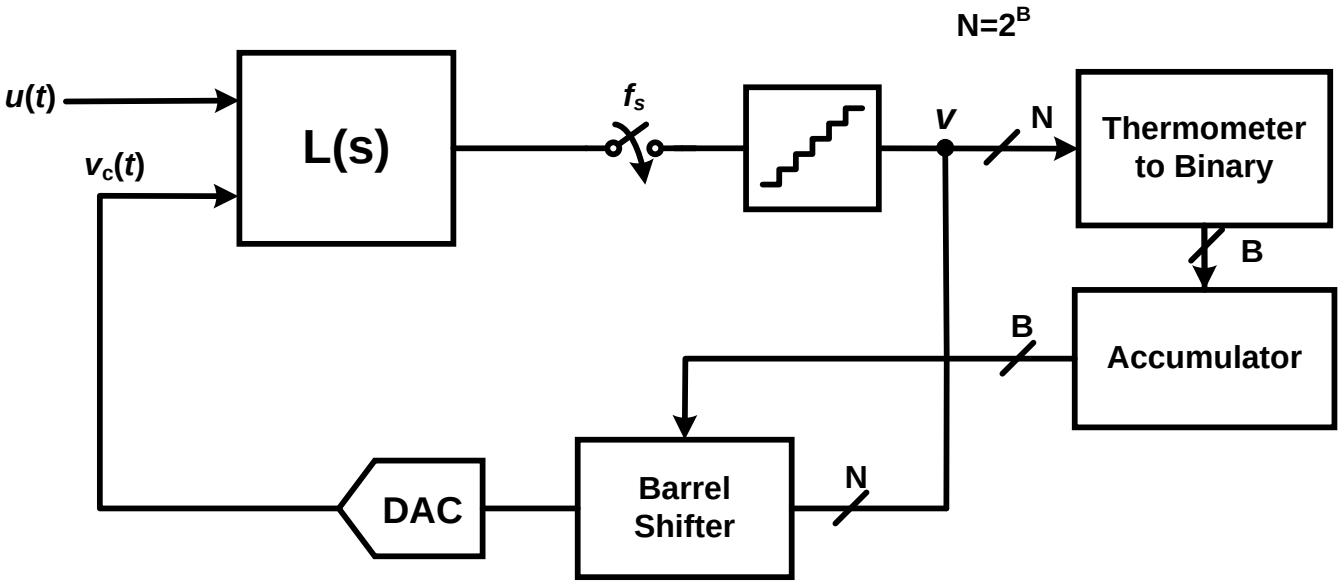
- Cycle through all the current elements as fast as possible
- Accumulate the input code and move the pointer
- First-order mismatch noise shaping







Data Weighted Averaging (DWA)

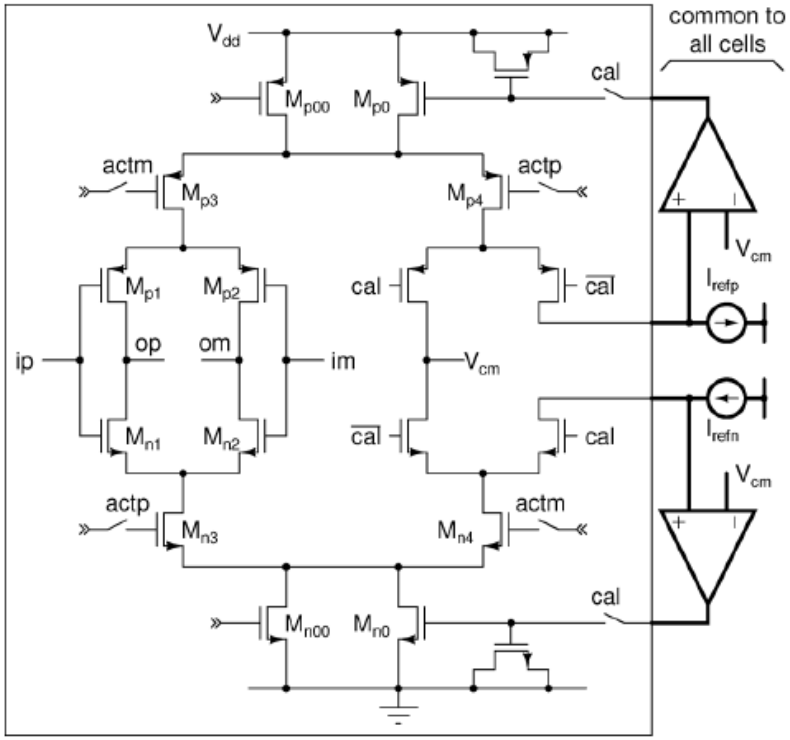


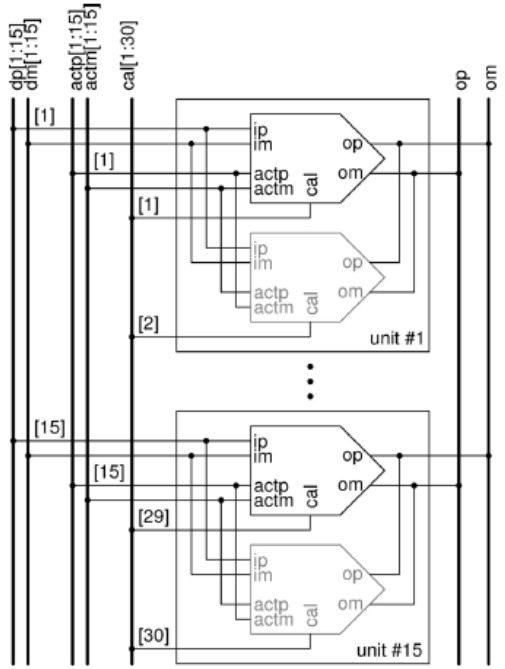
- Barrel shifter delay in the signal path increases loop delay
- Not viable at higher sampling rates (>400 MHz)

DAC Calibration

A 16 MHz BW 75 dB DR CT ΔΣ ADC Compensated for More Than One Cycle Excess Loop Delay

Vikas Singh, Nagendra Krishnapura, Shanthi Pavan, Baradwaj Vigraham, Debasish Behera, and Nimit Nigania





- Digital calibration – estimate element error and subtract from the output
  - Increased Decimation filter complexity
- Analog calibration – calibrate the elements with respect to a master
  - Need to calibrate at every cycle

CT ΔΣ Advantages Summary

- Lower-power implementation
  - Relaxed bandwidth requirements for the integrators
- Inherent Anti-aliasing filtering (AAF)
  - Eliminates/relaxes input filtering
- Fixed resistive input impedance
- Higher sampling-rates extending to GHz-range
  - Suitable for RF integration
- Reduces supply and ground noise impact
- Less complicated clocking (compared to DT)

CT ΔΣ Challenges Summary

- Complicated design due to hybrid CT-DT modeling
- Design doesn't scale with clock frequency
  - Loop-filter coefficient tuning for clock frequency migration
- High sensitivity to clock jitter (DAC reconstruction error)
- Excess loop-delay sensitivity
- Tuning required for RC time-constant variation
- Sensitive to DAC pulse shape, rise/fall time at high-speeds
- Comparator metastability at high speeds
- Higher-level simulation is challenging compared to DT
- Cascaded (MASH) designs are difficult due design complexity and mismatch in analog and digital transfer functions
  - Background calibration techniques

Design Tools

- MATLAB and Simulink
  - Dr. Richard Schreier's ΔΣ Toolbox
  - SIMSIDES Toolbox from Dr. Jose de la Rosa at University of Seville, Spain (Available under NDA)
- Verilog-A/AMS behavioral modeling in Cadence/Spectre
  - *Config view* in Virtuoso comes in handy for simulations
- Spectre Simulink Co-simulation Toolkit
- Berkeley Design Automation FastSpice for full-chip sims
- Synopsis tools for digital logic simulation and synthesis
- Mentor Graphics Calibre for DRC/LVS/Extraction

Use top-down design methodology with carefully thought mixed block-level simulations

References

Data Conversion Fundamentals

A.1 M. Gustavsson, J. Wikner, N. Tan, *CMOS Data Converters for Communications*, Kluwer Academic Publishers, 2000.

A.2 B. Razavi, *Principles of Data Conversion System Design*, Wiley-IEEE Press, 1994.

A.3 ADC and DAC Glossary by Maxim.

A.4 B. Murmann, "ADC Performance Survey 1997-2009," [Online].

A.5 S. Pavan, N. Krishnapura, EE658: Data Conversion Circuits Course at IIT Madras [Online].

A.6 The Fundamentals of FFT-Based Signal Analysis and Measurement, T.I. App Note here.

References

Delta-Sigma Data Converters

B.1 R. Schreier, G. C. Temes, *Understanding Delta-Sigma Data Converters*, Wiley-IEEE Press, 2005 (the Green Bible of Delta-Sigma Converters).

B.2 S. R. Norsworthy, R. Schreier, G. C. Temes, *Delta-Sigma Data Converters: Theory, Design, and Simulation*, Wiley-IEEE Press, 1996 (the Yellow Bible of Delta-Sigma Converters).

B.3 S. Pavan, N. Krishnapura, "Oversampling Analog to Digital Converters Tutorial," 21st International Conference on VLSI Design, Hyderabad, Jan, 2008.

B.4 S. H. Ardalan, J. J. Paulos, "An Analysis of Nonlinear behavior in Delta-Sigma Modulators," *IEEE TCAS*, vol. 34, no. 6, June 1987.

B.5 R. Schreier, "An Empirical Study of Higher-Order Single-Bit Delta-Sigma Modulators," *IEEE TCAS-II*, vol. 40, no. 8, pp. 461-466, Aug. 1993.

B.6 J. G. Kenney and L. R. Carley, "Design of multibit noise-shaping data converters," *Analog Integrated Circuits Signal Processing Journal*, vol. 3, pp. 259-272, 1993.

B.7 L. Risbo, "Delta-Sigma Modulators: Stability Analysis and Optimization," Doctoral Dissertation, Technical University of Denmark, 1994 [Online].

B.8 R. Schreier, J. Silva, J. Steensgaard, G. C. Temes,"Design-Oriented Estimation of Thermal Noise in Switched-Capacitor Circuits," *IEEE TCAS-I*, vol. 52, no. 11, pp. 2358-2368, Nov. 2005.

References

Continuous-Time Delta-Sigma Converters

C.1 M. Ortmanns, F. Gerfers, *Continuous-Time Sigma-Delta A/D Conversion: Fundamentals, Performance Limits and Robust Implementations*, Springer, 2006.

C.2 J.A. Cherry, "Theory, Practice, and Fundamental Performance Limits of High-Speed Data Conversion Using Continuous-Time Delta-Sigma Modulators," PhD Thesis, Carleton University, 1998 [Online].

C.3 K. Reddy, S. Pavan, "Fundamental Limitations of Continuous-time Delta-Sigma Modulators due to Clock Jitter," *IEEE TCAS-I*, vol. 54, no. 10, pp. 2185-2194, Oct. 2007.

C.4 S. Pavan, N. Krishnapura, R. Pandarinathan and P. Sankar, "A Power Optimized Continuous-time Delta-Sigma Modulator for Audio Applications," *IEEE JSSC*, vol. 43, no. 2, pp. 351-360, Feb. 2008.

C.5 S. Pavan, "Excess Loop Delay Compensation in Continuous-time Delta-Sigma Modulators", *IEEE TCAS-II: Express Briefs*, vol. 55, no. 11, pp. 1119-1123 , Nov. 2008.

C.6 Z. Li, T. S. Fiez, "A 14-bit Continuous-Time Delta-Sigma A/D Modulator with 2.5 MHz Signal Bandwidth," *IEEE JSSC*, vol. 42, no. 9, pp. 1873-1883, Sept. 2007.

C.7 S. Pavan and N. Krishnapura, "Automatic Tuning of Time-Constants in Continuous-Time Delta-Sigma Modulators," *IEEE TCAS-II: Express Briefs*, vol. 54, no. 4, pp. 308-312, Apr. 2007.

References

Continuous-Time Delta-Sigma Converters

C.8 Krishnapura, N. Singh, V. and S. Pavan, "Compensating for Quantizer Delay in Excess of One Clock Cycle in Continuous-time Delta-Sigma Modulators," in Circuits and Systems II: Express Briefs, IEEE Transactions on, vol. 57, no. 9, 2010, pp. 676-680.

C.9 M. Ortmanns, F. Gerfers, and Y. Manoli, "Compensation of Finite Gain-Bandwidth Induced Errors in Continuous-time Sigma-Delta Modulators," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 51, no. 6, pp. 1088-1099, 2004.

C.10 S. Pavan, "Systematic Design Centering of Continuous-time Oversampling Converters," IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 57, no. 3, pp. 158-162, 2010.

C.11 S. Pavan and N. Krishnapura, "Automatic Tuning of Time Constants in Continuous-time Delta-Sigma Modulators," IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 54, no. 4, pp. 308-312, 2007.

C.12 G. Mitteregger, C. Ebner, S. Mechnig, T. Blon, C. Holuigue, and E. Romani, "A 20-mW 640-MHz CMOS Continuous-Time SigmaDelta ADC With 20-MHz Signal Bandwidth, 80-dB Dynamic Range and 12-bit ENOB," IEEE Journal of Solid-State Circuits, vol. 41, no. 12, pp. 2641-2649, 2006.

C.13 Y. Ke, J. Craninkx, and G. Gielen, "Multi-standard Continuous-time Sigma-Delta Converters for 4G Radios," Circuits and Systems for Future Generations of Wireless Communications, pp. 203-221, 2009.

References

CAD for Mixed-Signal Design

D.1 K. Kundert, "Principles of Top-Down Mixed-Signal Design," *Designer's Guide Community* [Online].

D.2 R. Schreier, Matlab Delta-Sigma Toolbox, 2009 [Online], [Manual], [One page summary].

D.3 Jose de la Rosa, "SIMSIDES Toolbox: An Interactive Tool for the Behavioral Simulation of Discrete-and Continuous-time SD Modulators in the MATLAB," University of Sevilla, Spain, [Contact the authors for the software].

D.4 P. Malcovati, Simulink Delta-Sigma Toolbox 2, 2009. Available [Online].

Example Datasheets

E.1 A 16-bit, 2.5MHz/5 MHz/10 MHz, 30 MSPS to 160 MSPS Dual Continuous Time Sigma-Delta ADC – AD9262, Analog Devices, 2008.

E.2 A 24-bit, 192 kHz Multi-bit Audio ADC – CS5340, Cirrus Logic, 2008.