EECS 247 Lecture 8

- Finishing Continuous-time filters
  - Various Gm-C Filter implementations
  - Comparison of continuous-time filter topologies
- Switched-Capacitor Filters
  - “Analog” sampled-data filters:
    - Continuous amplitude
    - Quantized time
  - Applications:
    - First commercial product: Intel 2912 voice-band CODEC chip, 1979
    - Oversampled A/D and D/A converters
    - Stand-alone filters
      E.g. National Semiconductor LMF100

BiCMOS Gm-Cell

- MOSFET in triode mode:
  \[ I_d = \frac{\mu C_{ox}}{2} \frac{W}{L} \left[ 2(V_{gs} - V_{th})V_{ds} - V_{ds}^2 \right] \]

- Note that if Vds is kept constant:
  \[ s_m = \frac{\partial I_d}{\partial V_{gs}} = \frac{\mu C_{ox} W}{L} V_{ds} \]

- Linearity performance function of how constant Vds can be held
  - Gain @ Node X must be minimized
  \[ A_k = s_m M_1 / s_{B1} \]

- Since for a given current, gm of BJT is larger compared to MOS preferable to have BJT
- Extra pole at node X

\[ \frac{gm}{V_{ds}} \text{ can be varied by changing } V_b \text{ and thus } V_{ds} \]
BiCMOS Gm-C Integrator

- Differential - needs common-mode feedback ckt
- Freq. tuned by varying Vb

- Design tradeoffs:
  - Extra poles at the input device drain junctions
  - Input devices have to be small to minimize parasitic poles
  - Results in high input-referred offset voltage → could drive ckt into non-linear region
  - Small devices → high 1/f noise

7th Order Elliptic Gm-C LPF
For CDMA RX Baseband Application

- Gm-Cell in previous page used to build a 7th order elliptic filter for CDMA baseband applications (650kHz corner frequency)
- In-band dynamic range of <50dB achieved
BiCMOS Gm-OTA-C Integrator

- Used to build filter for disk-drive applications
- Since high frequency of operation, time-constant sensitivity to parasitic caps significant.
  → Opamp used
- M2 & M3 added to compensate for phase lag (provides phase lead)


6th Order BiCMOS Continuous-time Filter & Second Order Equalizer for Disk Drive Read Channels

- Gm-C-opamp of the previous page used to build a 6th order filter for Disk Drive
- Filter consists of 3 biquads with max. Q of 2 each
- Performance in the order of 40dB SNDR achieved for variable corner frequency in discrete steps up to 20MHz

Gm-Cell
Source-Coupled Pair with Degeneration

- Gm-cell intended for low Q disk drive filter


M7,8 operating in triode mode determine the gm of the cell
- Feedback provided by M5,6 maintains the gate-source voltage of M1,2 constant by forcing their current to be constant → helps linearize $r_{ds}$ of M7,8
- Current mirrored to the output via M9,10 with a factor of $k$
- Performance level of about 50dB SNDR at f_{corner} of 25MHz achieved
BiCMOS Gm-C Integrator

- Needs higher supply voltage compared to the previous design
- M5 & M6 configured as capacitors added to compensate for RHP zero due to Cgd of M1 & M2 (moves it to LHP) size of M5-6 is 1/3 of M1-2
- Current ID used to tune filter critical frequency
- M3, M4 operate in triode mode and added to provide CMFB


BiCMOS Gm-C Filter For Disk-Drive Application

- Using the integrators shown in the previous page
- Biquad filter for disk drives
- \( gm_1 = gm_2 = gm_4 = 2gm_3 \)
- \( Q = 2 \)
- Tunable from 8MHz to 32MHz

Summary
Continuous-Time Filters

- Opamp RC filters
  - Good linearity \(\rightarrow\) High dynamic range (60-90dB)
  - Only discrete tuning possible
  - Medium usable signal bandwidth (<10MHz)
- Opamp MOSFET-C
  - Linearity compromised (typical dynamic range 40-60dB)
  - Continuous tuning possible
  - Low usable signal bandwidth (<5MHz)
- Opamp MOSFET-RC
  - Improved linearity compared to Opamp MOSFET-C (D.R. 50-90dB)
  - Continuous tuning possible
  - Low usable signal bandwidth (<5MHz)
- Gm-C
  - Highest frequency performance (at least an order of magnitude higher compared to the rest <100MHz)
  - Dynamic range not as high as Opamp RC but better than Opamp MOSFET-C (40-70dB)

Switched-Capacitor Filters
Example: Codec Chip

Switched-Capacitor Resistor

- Capacitor C is the “switched capacitor”
- Non-overlapping clocks $\phi_1$ and $\phi_2$ control switches S1 and S2, respectively
- $v_{IN}$ is sampled at the falling edge of $\phi_1$  
  - Sampling frequency $f_s$
- Next, $\phi_2$ rises and the voltage across C is transferred to $v_{OUT}$
- Why is this a resistor?

Switched-Capacitor Resistors

- Charge transferred from $v_{IN}$ to $v_{OUT}$ during each clock cycle is:
  \[ Q = C(v_{IN} - v_{OUT}) \]
- Average current flowing from $v_{IN}$ to $v_{OUT}$ is:
  \[ i = \frac{Q}{t} = Qf_s \]
  \[ i = f_sC(v_{IN} - v_{OUT}) \]
Switched-Capacitor Resistors

\[ i = f_s C(v_{IN} - v_{OUT}) \]

With the current through the switched capacitor resistor proportional to the voltage across it, the equivalent “switched capacitor resistance” is:

\[ R_{eq} = \frac{1}{f_s C} \]

Example

\[ f = 1 \text{MHz}, C = 1 \text{pF} \]

\[ \rightarrow R_{eq} = 1 \text{Mega}\Omega \]

Switched-Capacitor Filter

- Let’s build a “SC” filter …
- We’ll start with a simple RC LPF
- Replace the physical resistor by an equivalent SC resistor
- 3-dB bandwidth:

\[ \omega_{-3dB} = \frac{1}{R_{eq} C_2} = f_s \frac{C_1}{C_2} \]

\[ f_{-3dB} = \frac{1}{2\pi} f_s \frac{C_1}{C_2} \]
Switched-Capacitor Filter Advantage versus Continuous-Time Filters

\[ f_{-3dB} = \frac{1}{2\pi} f_s \times \frac{C_1}{C_2} \]

- Corner freq. proportional to:
  - System clock (accurate to few ppm)
  - C ratio accurate \( \rightarrow 0.1\% \)

\[ f_{-3dB} = \frac{1}{2\pi} \times \frac{1}{R_{eq} C_2} \]

- Corner freq. proportional to:
  - Absolute value of Rs & Cs
  - Poor accuracy \( \rightarrow 20 \text{ to } 50\% \)

Main advantage of SC filter inherent corner frequency accuracy

Typical Sampling Process
Continuous-Time (CT) \( \Rightarrow \) Sampled Data (SD)

Continuous-Time Signal

Sampled Data

Sampled Data + ZOH

Clock
Uniform Sampling

Nomenclature:
- Continuous time signal: \( x(t) \)
- Sampling interval: \( T \)
- Sampling frequency: \( f_s = 1/T \)
- Sampled signal: \( x(kT) = x(k) \)

- Problem: Multiple continuous time signals can yield exactly the same discrete time signal
- Let's look at samples taken at 1\( \mu \)s intervals of several sinusoidal waveforms …

Sampling Sine Waves

\[ v(t) = \sin \left(2\pi (101000)t\right) \]

\[ T = 1\mu s \]
\[ f_s = 1/T = 1\text{MHz} \]
\[ f_{in} = 101\text{kHz} \]
Sampling Sine Waves

\[ v(t) = -\sin[2\pi(899000)t] \]

\[ T = 1 \mu s \]
\[ f_s = 1 \text{MHz} \]
\[ f_{in} = 899\text{kHz} \]

\[ v(t) = \sin[2\pi(1101000)t] \]

\[ T = 1 \mu s \]
\[ f_s = 1 \text{MHz} \]
\[ f_{in} = 1101\text{kHz} \]
Sampling Sine Waves

Before Sampling

Frequency domain

After Sampling

Signal scenario before sampling

Signal scenario after sampling & filtering

⇒ Signals @

⇒ Aliasing

Frequency Domain Interpretation

Aliasing

Before Sampling

Frequency Domain

After Sampling

Signal scenario after sampling & filtering

⇒ Signals @

⇒ Aliasing

Signal scenario before sampling
Aliasing

- Multiple continuous time signals can produce identical series of sampled voltages
- The folding back of signals from $nf_S \pm f_{sig}$ down to $f_{fin}$ is called aliasing
  - Sampling theorem: $f_s > 2f_{max\_Signal}$
- If aliasing occurs, no signal processing operation downstream of the sampling process can recover the original continuous time signal

How to Avoid Aliasing

- Must obey sampling theorem:
  $$f_{max\_Signal} < f_s/2$$
- Two possibilities:
  1. Sample fast enough to cover all spectral components, including "parasitic" ones outside band of interest
  2. Limit $f_{max\_Signal}$ through filtering
How to Avoid Aliasing

1- Push sampling frequency to x2 of the highest freq. → In most cases not practical

2- Pre-filter signal to eliminate signals above 1/2 sampling frequency- then sample

Anti-Aliasing Filter

Case1 - \[ B = f_{\text{max}} - \text{Signal} = f_s/2 \]

- Non-practical since an extremely high order anti-aliasing filter (close to an ideal brickwall filter) is required
- Practical anti-aliasing filter → Nonzero filter "transition band"
- In order to make this work, we need to sample much faster than 2x the signal bandwidth → "Oversampling"
Practical Anti-Aliasing Filter

Case 2: \( B = f_{\text{max}} - \text{Signal} << f_s/2 \)
- More practical anti-aliasing filter
- Preferable to have an anti-aliasing filter with:
  - The lowest order possible
  - No frequency tuning required (if frequency tuning is required then why use SC filter, just use the prefilter?!)
Effect of Sample & Hold

Using the Fourier transform of a rectangular impulse:

\[ |H(f)| = \frac{T_p}{T_s} \frac{\sin(\pi f T_p)}{\pi f T_p} \]

Effect of Sample & Hold on Frequency Response

More practical
Sample & Hold Effect
(Reconstruction of Analog Signals)

Magnitude droop due to $\sin x/x$ effect:

Case 1) $f_{\text{sig}} = f_s/4$

Droop $= -1dB$
Sample & Hold Effect
(Reconstruction of Analog Signals)

Magnitude droop due to $\text{sin}x/x$ effect:

Case 2)
\[ f_{\text{sig}} = \frac{f_s}{32} \]

Droop = -0.0035dB

→ High oversampling ratio desirable

Time domain

Frequency domain

Summary

• Sampling theorem $\Rightarrow f_s > 2f_{\text{max, Signal}}$
• Signals at frequencies $nf_s \pm f_{\text{sig}}$ fold back down to desired signal band, $f_{\text{sig}}$
  → This is called aliasing & usually dictates use of anti-aliasing pre-filters
• Oversampling helps reduce order of anti-aliasing filter
• S/H function shapes the frequency response with $\text{sin}x/x$
  → Need to pay attention to droop in passband due to $\text{sin}x/x$
• If the above requirements is not met, CT signal can NOT be recovered from SD or DT without loss of information
1st Order Filter
Transient Analysis

- ZOH: pick signal after settling (usually at end of clock phase)
- Adds delay and sin(x)/x distortion
- When in doubt, use a ZOH in periodic ac simulations
Periodic AC Analysis

- RC filter output
- SC output after ZOH
- Input after ZOH
- Corrected output
  - (2) over (3)
  - Repeats filter shape around $nf_s$
  - Identical to RC for $f < nf_s/2$

Magnitude Response
Periodic AC Analysis

- **SPICE frequency analysis**
  - ac linear, **time-invariant** circuits
  - pac linear, **time-variant** circuits

- **SpectreRF statements**
  
  V1 ( Vi 0 ) vsource type=dc dc=0 mag=1 pacmag=1
  PSS1 pss period=1u errpreset=conservative
  PAC1 pac start=1 stop=1M lin=1001

- **Output**
  
  - Divide results by sinc(t/fs) to correct for ZOH distortion

---

Spectre Circuit File

```plaintext
rc_pac
simulator lang=spectre
ahdl_include "zoh.def"

S1 ( Vi c1 phi1 0 ) relay ropen=100G rclosed=1 vt1=-500m vt2=500m
S2 ( c1 Vo_sc phi2 0 ) relay ropen=100G rclosed=1 vt1=-500m vt2=500m
C1 ( c1 0 ) capacitor c=314.159f
C2 ( Vo_sc 0 ) capacitor c=1p
R1 ( Vi Vo_rc ) resistor r=3.1831M
C2rc ( Vo_rc 0 ) capacitor c=1p
CLK1_Vphi1 ( phi1 0 ) vsource type=pulse val0=-1 val1=1 period=1u
  width=450n delay=10n rise=10n fall=10n
CLK1_Vphi2 ( phi2 0 ) vsource type=pulse val0=-1 val1=1 period=1u
  width=450n delay=550n rise=10n fall=10n
V1 ( Vi 0 ) vsource type=dc dc=0 mag=1 pacmag=1
PSS1 pss period=1u errpreset=conservative
PAC1 pac start=1 stop=3.1M log=1001
ZOH1 ( Vo_sc_zoh 0 Vo_sc 0 ) zoh period=1u delay=500n aperture=1n tc=10p
ZOH2 ( Vi_zoh 0 Vi 0 ) zoh period=1u delay=0 aperture=1n tc=10p
```
ZOH Circuit File

// Copy from the SpectreRF Primer
module zoh (Pout, Nout, Pin, Nin) (period, delay, aperture, tc)
  node [V,I] Pin, Nin, Pout, Nout;
  parameter real period=1 from [0:inf];
  parameter real delay=0 from [0:inf];
  parameter real aperture=1/100 from [0:inf];
  parameter real tc=1/500 from [0:inf];
  integer n; real start, stop;
  node [V,I] hold;
  analog {
    // determine the point when aperture begins
    n = ($time() - delay + aperture) / period + 0.5;
    start = n*period + delay - aperture;
    $break_point(start);
    // determine the time when aperture ends
    n = ($time() - delay) / period + 0.5;
    stop = n*period + delay;
    $break_point(stop);
    // Implement switch with effective series resistance of 1 Ohm
    if ($time() > start) && ($time() <= stop)
      I(hold) <- V(hold) - V(Pin, Nin);
    else
      I(hold) <- 1.0e-12 * (V(hold) - V(Pin, Nin));
    // Implement capacitor with an effective capacitance of tc
    I(hold) <- tc * dot(V(hold));
    // Buffer output
    V(Pout, Nout) <- V(hold);
    // Control time step tightly during aperture and loosely otherwise
    if ($time() >= start) && ($time() <= stop)
      $bound_step(tc);
    else
      $bound_step(period/5);
  }

First Order S.C. Filter

Switched-Capacitor Filters ➔ problem with aliasing
Example: Anti-Aliasing Filter

- Voice-band SC filter $f_{-3dB} = 4kHz$ & $f_s = 256kHz$
- Anti-aliasing filter requirements:
  - Need 40dB attenuation at clock freq.
  - Incur no phase-error from 0 to 4kHz
  - Gain error 0 to 4kHz < 0.05dB
  - Allow ±30% variation for anti-aliasing corner frequency (no tuning)

  → 2-pole Butterworth LPF with nominal corner freq. of 17kHz & no tuning (12kHz to 22kHz corner frequency)

Ease of anti-aliasing → high ratio for $f_{sampling}/f_{-3dB}$

Switched-Capacitor Noise

- Resistance of switch $S_1$ produces a noise voltage on $C$ with variance $kT/C$

- The corresponding noise charge is $Q^2 = C^2V^2 = kTC$

- This charge is sampled when $S_1$ opens
Switched-Capacitor Noise

- Resistance of switch S2 contributes to an uncorrelated noise charge on C at the end of \( \phi_2 \).

- Mean-squared noise charge transferred from \( v_{\text{IN}} \) to \( v_{\text{OUT}} \) each sample period is \( Q^2 = 2kTC \).

\[
\begin{align*}
\Delta f &= \frac{2kT}{f_s/2} = \frac{4kT}{f_s} \quad \text{using} \quad R_{\text{eq}} = \frac{1}{f_sC} \\
\Rightarrow \text{S.C. resistor noise equals a physical resistor noise with same value!}
\end{align*}
\]
Periodic Noise Analysis

Sampling Noise from SC S/H

SpectreRF PNOISE: check
noisetype=timedomain
noisetimelimit=[…]
as alternative to ZOH.
noiseskipcount=large
might speed up things in this case.

PSS pss period=100n maxacfreq=1.5G errpreset=conservative
PNOISE ( Vrc_hold 0 ) pnoise start=0 stop=20M lin=500 maxsideband=10

Sampled Noise Spectrum

Density of sampled noise with sinc distortion.

Normalized density of sampled noise, corrected for sinc distortion.
Total Noise

Sampled noise in 0 … \( f_s/2 \): 62.2\( \mu \)V rms

(expect 64\( \mu \)V for 1pF)

Switched-Capacitor Integrator

Main advantage: No tuning needed

\[ \omega_0 = f_s \times \frac{C_s}{C_I} \]

\[ V_0 = \frac{f_s \times C_s}{C_I} \int V_{in} \, dt \]
Continuous-Time versus Discrete Time Design Flow

**Continuous-Time**
- Write differential equation
- Laplace transform \((F(s))\)
- Let \(s = j\omega \rightarrow F(j\omega)\)
- Plot \(|F(j\omega)|, \text{phase}(F(j\omega))\)

**Discrete-Time**
- Write difference equation, \(\rightarrow\) relates output sequence to input sequence
- \(V_o(nT_s) = V_i [n - 1, j\omega \] \(\) 
- Use delay operator \(Z^{-1}\) to transform the recursive realization to algebraic equation in \(Z\) domain
- \(V_o(Z) = Z^{-1}V_i(Z)\)
- Set \(Z = e^{-j\omega T}\)
- Plot mag./phase versus frequency
SC Integrator

\[ \begin{align*}
\Phi_1 \rightarrow & \quad Q_1[(n-1)T_s] = C_s V_i[(n-1)T_s], \quad Q_1[(n-1)T_s] = Q_1[(n-3/2)T_s] \\
\Phi_2 \rightarrow & \quad Q_2[(n-1/2)T_s] = 0, \quad Q_2[(n-1/2)T_s] = Q_2[(n-3/2)T_s] + Q_2[(n-1)T_s] \\
\Phi_1 \rightarrow & \quad Q_1[nT_s] = C_s V_i[nT_s], \quad Q_1[nT_s] = Q_2[(n-1)T_s] + Q_2[(n-1)T_s] \\
\text{Since } V_o &= -Q_i/C_1 \text{ & } V_i = Q_i/C_1 \rightarrow C_1 V_o[nT_s] = C_1 V_o[(n-1)T_s] - C_1 V_i[(n-1)T_s] 
\end{align*} \]
Discrete Time Design Flow

- Transforming the recursive realization to algebraic equation in $Z$ domain:
  - Use Delay operator $Z$:

$$
\begin{align*}
    nT_s & \rightarrow 1 \\
    (n-1)T_s & \rightarrow Z^{-1} \\
    (n-1/2)T_s & \rightarrow Z^{-1/2} \\
    (n+1)T_s & \rightarrow Z^{+1} \\
    (n+1/2)T_s & \rightarrow Z^{+1/2}
\end{align*}
$$

SC Integrator

$$
\begin{align*}
    -C_I V_o(nT_s) &= -C_I V_o[(n-1)T_s] + C_s V_{in}[n-1]T_s \\
    V_o(nT_s) &= V_o[(n-1)T_s] - \frac{C_s}{C_I} V_{in}(n-1)T_s \\
    V_o(Z) &= Z^{-1} V_o(Z) - Z^{-1} \frac{C_s}{C_I} V_{in}(Z) \\
    \frac{V_o(Z)}{V_{in}} &= -\frac{C_I}{C_s} \frac{Z^{-1}}{1-Z^{-1}}
\end{align*}
$$

DDI (Direct-Transform Discrete Integrator)
z-Plane

- Consider variable $Z = e^{sT}$ for any $s$ in left-half-plane (LHP):
  \[ S = -a + jb \]
  \[ Z = e^{-aT} \cdot e^{jbT} = e^{-aT} (\cos bT + jsin bT) \]
  \[ |Z| = e^{-aT}, \text{angle}(Z) = bT \]
  → For values of $S$ in LHP $|Z| < 1$
  → For $a = 0$ (imag. axis in s-plane) $|Z| = 1$ (unit circle)
    if \( \text{angle}(Z) = \pi = bT \) then \( b = \pi / T = \omega \)
  Then \( \omega = \omega_s / 2 \)

z-Domain Frequency Response

- LHP singularities in s-plane map into inside of unit-circle in Z domain
- RHP singularities in s-plane map into outside of unit-circle in Z domain
- The $j\omega$ axis maps onto the unit circle
**z-Domain Frequency Response**

- Particular values:
  - \( f = 0 \rightarrow z = 1 \)
  - \( f = f_s/2 \rightarrow z = -1 \)
- The frequency response is obtained by evaluating \( H(z) \) on the unit circle at \( z = e^{j\omega T} = \cos(\omega T) + jsin(\omega T) \)
- Once \( z = 1 \) \( (f_s/2) \) is reached, the frequency response repeats, as expected

\[
(f_s/2, 0) \rightarrow (cos(\omega T), sin(\omega T))
\]

- The angle to the pole is equal to \( 360^\circ \) (or \( 2\pi \) radians) times the ratio of the pole frequency to the sampling frequency.

\[
(f_s/2, 0) \rightarrow (cos(\omega T), sin(\omega T))
\]
s-Plane versus z-Plane
Example: 2nd Order LDI Bandpass Filter

Pole-Zero Map in z-Plane

Zero from $f \to \infty$ in s-plane mapped to $z=0$, a non-physical frequency.

Zero from $f \to 0$ in s-plane mapped to $z=+1$

Distance from the pole to the unit circle is inversely proportional to pole $Q$

Pole on unit-circle $\to Q$ of infinity
Example: Mag. & phase error for:

1. $f/f_s=1/12 \Rightarrow$ Mag. Error = 1% or 0.1dB
   - Phase error = 15 degree
   - $Q_{intg} = -3.8$

2. $f/f_s=1/32 \Rightarrow$ Mag. Error = 0.16% or 0.014dB
   - Phase error = 5.6 degree
   - $Q_{intg} = -10.2$

DDI Integrator $\Rightarrow$ phase error major problem

Magnitude error no problem

Ideal Integrator

$$\frac{V_o}{V_{in}}(Z) = \frac{C_s}{C_1} \times \frac{Z^{-1}}{1-Z^{-1}} = \frac{C_s}{C_1} \times e^{-j\omega_T/2}$$

$$= -\frac{C_s}{C_1} \times e^{-j\omega_T/2} \times \frac{1}{\sin(\omega_T/2)}$$

$$= -\frac{C_s}{C_1} \frac{1}{\sin(\omega_T/2)} \times e^{-j\omega_T/2}$$

$$\Rightarrow$$ magnitude error no problem phase error major problem
SC Integrator

Sample output \( \frac{1}{2} \) clock cycle earlier

\[ \rightarrow \text{Sample output on } \Phi_2 \]

\( \Phi_1 \rightarrow Q_i[(n-1)T]\) = \( C_s V_i[(n-1)T]\),  
\( Q_I[(n-1)T_s] = Q_i[(n-3/2)T_s] \)

\( \Phi_2 \rightarrow Q_i[(n-1/2)T_s] = 0 \),  
\( Q_i[(n-3/2)T_s] = Q_i[(n-1)T_s] + Q_s[(n-1)T_s] \)

\( \Phi_1 \rightarrow Q_i[nT_s] = C_s V_i[nT_s] \),  
\( Q_I[nT_s] = Q_i[(n-1)T_s] + Q_s[nT_s] \)

\( \Phi_2 \rightarrow Q_i[(n+1/2)T_s] = 0 \),  
\( Q_i[(n+1/2)T_s] = Q_i[(n-1/2)T_s] + Q_s[nT_s] \)