EE247  
Lecture 22

ADC Converters
- Comparator design (continued)
  - Comparator architecture examples

- Techniques to reduce flash ADC complexity
  - Interpolating
  - Folding
  - Multi-Step ADCs
    - Two-Step flash
    - Pipelined ADCs

Summary Last Lecture

ADC Converters
- Comparator design
  - Single-stage high-gain open-loop amplifier
  - Cascade of open-loop amplifiers
  - Problem associated with DC offset
    - Cascaded output series cancellation
    - Input series cancellation
    - Offset cancellation through additional input pair plus offset storage capacitors
  - Latched comparators
  - Comparator examples
Comparator with Auto-Zero


Flash ADC
Comparator with Auto-Zero


\[ V_{C+} - V_{C-} = (V_{Ref+} - V_{Ref-}) - V_{Offset} \]
Flash ADC
Comparator with Auto-Zero

\[ V_i = A_1 \cdot A_2 \left( V_{in} - V_{ref} \right) - \left( V_{in} - V_{c} \right) - V_{offset} \]

Substituting for \( (V_{c} - V_{i}) \) from previous cycle:

\[ V_i = A_1 \cdot A_2 \left( V_{in} - V_{ref} \right) - \left( V_{in} - V_{c_{prev}} \right) \]

Note: Offset is cancelled & difference between input & reference established.

Auto-Zero Implementation


Comparator Example

- Variation on Yukawa latch used w/o preamp
- Good for low resolution ADCs (in this case 1.5bit/stage for a pipeline)
- Note: M1, M2, M11, M12 operate in triode mode
- M11 & M12 added to vary comparator threshold
- Conductance at node X is sum of G_{M1} & G_{M11}

Comparator Example (continued)

- M1, M2, M11, M12 operate in triode mode with all having equal L
- Conductance of input devices:
  \[ G_1 = \frac{\mu C_{ox} W_1}{L} \left[ (V_{th} - V_{th1}) + W_1 (V_R - V_{th}) \right] \]
  \[ G_2 = \frac{\mu C_{ox} W_2}{L} \left[ (V_{th2} - V_{th}) + W_2 (V_R + V_{th}) \right] \]
  \[ \Delta = \frac{G_1}{G_2} \left[ (V_{th1} - V_{th2}) - \frac{W_1 L}{W_2 L} (V_R + V_R) \right] \]
- To 1st order, for \( W_1 = W_2 \) & \( W_{11} = W_{12} \)
  \[ V_{th} = W_{11}/W_1 \times V_R \]
  where \( V_R = V_R^+ - V_R^- \)
  \( \Rightarrow \) fixed \( W_{11}, W_{12} \) varied from comparator to comparator
  \( \Rightarrow \) Eliminates need for resistive divider


Comparator Example

- Used in a pipelined ADC with digital correction
  \( \Rightarrow \) no offset cancellation required
- Differential reference & input
- M7, M8 operate in triode region
- Preamp gain ~10
- Input buffers suppress kick-back
- \( \phi_1 \) high \( \Rightarrow \) \( C_s \) charged to \( V_R \) & \( \phi_2 \) is also high \( \Rightarrow \) current diverted to latch \( \Rightarrow \) comparator output in hold mode
- \( \phi_2 \) high \( \Rightarrow \) \( C_s \) connected to S/H output \& comparator input (\( V_R - S/Hout \)), current sent to preamp \( \Rightarrow \) comparator in amplify mode

Bipolar Comparator Example

• Used in 8-bit 400Ms/s & 6-bit 2Gb/s flash ADC
• Signal amplification during $\phi_1$ high, latch operates when $\phi_1$ low
• Input buffers suppress kick-back & input current
• Separate ground and supply buses for front-end preamp $\rightarrow$ kick-back noise reduction


Reducing Flash ADC Complexity

E.g. 10-bit “straight” flash
– Input range: 0 … 1V
– LSB = $\Delta$: $\sim$ 1mV
– Comparators: 1023 with offset $< 1/2$ LSB
– Input capacitance: 1023 * 100fF = 102pF
– Power: 1023 * 3mW = 3W
$\rightarrow$ High power dissipation & large area & high input cap.

Techniques to reduce complexity & power dissipation:
– Interpolation
– Folding
– Folding & Interpolation
– Two-step, pipelining
Interpolation

- **Idea**
  - Reduce number of preamps & instead interpolate between preamp outputs

- **Reduced number of preamps**
  - Reduced input capacitance
  - Reduced area, power dissipation

- **Same number of latches (2^B-1)**

- **Important “side-benefit”**
  - Decreased sensitivity to preamp offset
    $\rightarrow$ Improved DNL

---

**Flash ADC**

**Preamp Output**

Zero crossings (to be detected by latches) at $V_{in} =$

- $V_{ref1} = 1 \Delta$
- $V_{ref2} = 2 \Delta$
Simulink Model

Differential Preamp Output

Differential output crossings
@ $V_{in}$ =

- $V_{ref1} = 1 \Delta$
- $V_{ref2} = 2 \Delta$

Note: Additional crossing of $A_1$ & $-A_2$ ($A_2$ & $-A_1$)

$\Rightarrow A_1$ & $-A_2$)
$\Rightarrow$ cross zero at:

$V_{ref2} = 0.5^*\Delta = 0.5 \Delta = 1.5 \Delta$
Interpolation

- Idea
  - Reduce number of preamps & instead interpolate between preamp outputs

- Reduced number of preamps
  - Reduced input capacitance
  - Reduced area, power dissipation

- Same number of latches \((2^B-1)\)

- Important “side-benefit”
  - Decreased sensitivity to preamp offset
  \(\rightarrow\) Improved DNL

Preamp Output

Zero crossings (to be detected by latches) at \(V_{in} = \)

\[V_{ref1} = 1 \Delta\]
\[V_{ref2} = 2 \Delta\]
Differential Preamp Output

Differential output crossings
@ \( V_{\text{in}} = \)
\[ V_{\text{ref}1} = 1 \Delta \]
\[ V_{\text{ref}2} = 2 \Delta \]

Note: Additional crossing of
\( A_1 & - A_2 \) (\( A_2 & - A_1 \))
Æ \( A_1 - (-A_2) = A_1 + A_2 \)
Æ cross zero at:
\[ V_{\text{ref12}} = 0.5^*(1+2) \Delta = 1.5 \Delta \]

Interpolation in Flash ADC

Half as many reference voltages
and preamps
Interpolation factor: x2

Example: For 10-bit straight Flash
ADC need \( 2^{10} = 1024 \) preamps
compared \( 2^{9} - 1 = 512 \) for x2 interpolation

Possible to accomplish higher
interpolation factor
Æ Interpolation at the output of
preamps

Compare \( A_2 & - A_1 \)
Æ Comparator output is sign of \( A_1 + A_2 \)
Interpolation in Flash ADC

**Preamp Output Interpolation**

Interpolate between two consecutive output via impedance $Z$

**Choices of $Z$:**
1. Resistors (Kimura)
2. Capacitors (Kusumoto)
3. Current mode (Roovers)

![Interpolation Diagram](image)


Higher Order Resistive Interpolation

- Resistors produce additional levels
- With 4 resistors per side, the “interpolation factor” $M=8$ → extra 3 bits
- $(M \rightarrow$ ratio of latches/preamps)

Preamp Output Interpolation
DNL Improvement

- Preamp offset distributed over M resistively interpolated voltages:
  → Impact on DNL divided by M

- Latch offset divided by gain of preamp
  → Use “large” preamp gain
  → Next: Investigate how large preamp gain can be


Preamp Input Range

If linear region of preamp transfer curve do not overlap

→ Dead-zone in the interpolated transfer curve!
Results in error

→ Linear consecutive preamp input ranges must overlap
  i.e. input range > Δ

Sets upper bound on preamp gain: Preampgain < VDD / Δ
Interpolated-Parallel ADC

- 10-bit overall resolution:
- 7-bit flash (127 preamps and 128 resistors) & x8 interpolation

Use of Gray Encoder minimizes effect of sparkle code & metastability


Measured Performance

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resolution</td>
<td>10 b (7+3)</td>
</tr>
<tr>
<td>Maximum conversion freq</td>
<td>300 MHz</td>
</tr>
<tr>
<td>Integral non-linearity</td>
<td>±1.0 LSB</td>
</tr>
<tr>
<td>Differential non-linearity</td>
<td>±0.4 LSB</td>
</tr>
<tr>
<td>SNR/THD</td>
<td></td>
</tr>
<tr>
<td>10MHz input</td>
<td>56/59 dB</td>
</tr>
<tr>
<td>50MHz input</td>
<td>48/47 dB</td>
</tr>
<tr>
<td>Input capacitance</td>
<td>8 pF</td>
</tr>
<tr>
<td>Input range</td>
<td>2 V</td>
</tr>
<tr>
<td>Power supply</td>
<td>-5.2V</td>
</tr>
<tr>
<td>Power dissipation</td>
<td>4.0W</td>
</tr>
<tr>
<td>Chip size</td>
<td>9.0 x 4.2 mm(^2)</td>
</tr>
<tr>
<td>Element count</td>
<td>36,000</td>
</tr>
<tr>
<td>Technology</td>
<td>1.0 (\mu)m bipolar: (f_t=25)GHz</td>
</tr>
</tbody>
</table>

Interpolation Summary

- Consecutive preamp transfer curve need to have overlap → Limits gain of preamp to $\sim V_{DD}/\Delta$
- The added impedance at the output of the preamp typically reduces the bandwidth and affects the maximum achievable frequencies
- DNL due to preamp offset reduces by interpolation factor $M$
- Interpolation reduces # of preamps and thus reduces input C-
  however, the # of required latches the same as "straight" Flash
  → Use folding to reduce the # of latches

Folding Converter

- Two ADCs operating in parallel
  - MSB ADC
  - Folder + LSB ADC
- Significantly fewer comparators compared to flash
- Fast
- Typically, nonidealities in folder limit resolution to $\sim 10$Bits
Example: Folding Factor of 4

- Folding factor → number of folds
- Folder maps input to smaller range
- MSB ADC determines which fold input is in
- LSB ADC determines position within fold
- Logic circuit combines LSB and MSB results

- How are folds generated?

  Fold 1: \( V_{out} = V_{in} \)
  Fold 2: \( V_{out} = -V_{in} + \frac{V_{FS}}{2} \)
  Fold 3: \( V_{out} = V_{in} - \frac{V_{FS}}{2} \)
  Fold 4: \( V_{out} = -V_{in} + V_{FS} \)

- Note: Sign change every other fold + reference shift
Generating Folds via Source-Couple Pairs

Vref1 < Vref2 < Vref3 < Vref4
As Vin changes, only one of M1, M3, M5, M7 is on depending on the input level

CMOS Folder Output

CMOS folder transfer curve max. min. portions:
   → Rounded
   → Accurate only at zero-crossings

In fact, most folding ADCs do not use the folds, but only the zero-crossings!
Parallel Folders Using Only Zero-Crossings

- Folder 1: \( V_{\text{ref}} + 0/4 \ast \Delta \)
- Folder 2: \( V_{\text{ref}} + 1/4 \ast \Delta \)
- Folder 3: \( V_{\text{ref}} + 2/4 \ast \Delta \)
- Folder 4: \( V_{\text{ref}} + 3/4 \ast \Delta \)

Comparator

Logic

LSB bits
(to be combined with MSB bits)

Vin

• 4 folders with 4 folds each
• 16 zero crossings
• \( \rightarrow \) 4 LSB bits
• Higher resolution
  • More folders
    \( \rightarrow \) Large complexity
  • Interpolation

Parallel Folder Outputs
Folding & Interpolation

Folder 4
\[ V_{\text{ref}} + \frac{3}{4} \Delta \]

Folder 3
\[ V_{\text{ref}} + \frac{2}{4} \Delta \]

Folder 2
\[ V_{\text{ref}} + \frac{1}{4} \Delta \]

Folder 1
\[ V_{\text{ref}} + \frac{0}{4} \Delta \]

Fine Flash ADC

Folder / Interpolator Output

Example: 4 Folders + 4 Resistive Interpolator per Stage

Note: Output of two folders only + corresponding interpolator only shown.
Folder / Interpolator Output

Example: 2 Folders + 8 Resistive Interpolator per Stage

Non-linear distortion

→ Interpolate only between closely spaced folds to avoid nonlinear distortion

A 70-MS/s 110-mW 8-b CMOS Folding and Interpolating A/D Converter

Ref: B. Nauta and G. Venes, JSSC Dec 1985, pp. 1302-8
A 70-MS/s 110-mW 8-b CMOS Folding and Interpolating A/D Converter

Note:
Total of 40 (MSB=8, LSB=32) comparators compared to $2^{8}-1=255$ for straight flash

---

Ref: B. Nauta and G. Venes, JSSC Dec 1985, pp. 1302-8
Time Interleaved Converters

• Example:
  – 4 ADCs operating in parallel at sampling frequency $f_s$
  – Each ADC converts on one of the 4 possible clock phases
  – Overall sampling frequency= $4f_s$
  – Note T/H has to operate at $4f_s$!

• Extremely fast:
  Typically, limited by speed of T/H

• Accuracy limited by mismatch among individual ADCs (timing, offset, gain, …)

Two-Step Example: (2+2)Bits

• Using only one ADC: output contains large quantization error
  • "Missing voltage" or "residue" ($-\epsilon_{q1}$)
  • Idea: Use second ADC to quantize and add $-\epsilon_{q1}$
Two Stage Example

- Use DAC to compute missing voltage
- Add quantized representation of missing voltage
- Why does this help? How about $\epsilon_{q2}$?

Two Step (2+2) Flash ADC

4-bit Straight Flash ADC

Ideal 2-step Flash ADC
Two Stage Example

- Fine ADC is re-used $2^2$ times
- Fine ADC's full scale range needs to span only 1 LSB of coarse quantizer

$$\epsilon_{q^2} = \frac{V_{ref2}}{2^2} = \frac{V_{ref1}}{2^2 \cdot 2^2}$$

Two-Stage (2+2) ADC Transfer Function

- $V_{ref1}$
- $V_{ref2}$
- $V_{in}$
- $D_{out}$

Coarse Bits (MSB)  Fine Bits (LSB)
Residue or Multi-Step Type ADC

Issues

• Operation:
  – Coarse ADC determines MSBs
  – DAC converts the coarse ADC output to analog- Residue is found by subtracting \((V_{in} - V_{DAC})\)
  – Fine ADC converts the residue and determines the LSBs
  – Bits are combined in digital domain

• Issue:
  1. Fine ADC has to have precision in the order of overall ADC 1/2LSB
  2. Speed penalty → Need at least 1 clock cycle per extra series stage to resolve one sample

Solution to Issue (1)

• Accuracy needed for fine ADC relaxed by introducing inter-stage gain
  – Example: By adding gain of \(x(G=2^{B_1}=4)\) prior to fine ADC in (2+2)bit case, precision required for fine ADC is reduced to 2-bit only!
  – Additional advantage- coarse and fine ADC can be identical stages
Solution to Issue (2)

• Conversion time significantly decreased by employing T/H between stages
  – All stages busy at all times → operation concurrent
  – During one clock cycle coarse & fine ADCs operate concurrently:
    • First stage samples/converts/generates residue of input signal sample # \( n \)
    • While 2nd samples/converts residue associated with sample # \( n-1 \)

\[ D_{\text{out}} = V_n + \epsilon_q + \epsilon_q' + \epsilon_{q2} \]

Pipelined A/D Converters

• Ideal operation
• Errors and correction
  – Redundancy
  – Digital calibration
• Implementation
  – Practical circuits
  – Stage scaling
Pipeline ADC Block Diagram

- Idea: Cascade several low resolution stages to obtain high overall resolution (e.g. 10bit ADC can be built with series of 10 ADCs each 1-bit only!)
- Each stage performs coarse A/D conversion and computes its quantization error, or "residue"
- All stages operate concurrently

Digital output: \((B_1 + B_2 + \ldots + B_k)\) Bits

Pipeline ADC Characteristics

- Number of components (stages) grows linearly with resolution
- Pipelining
  - Trading latency for conversion speed
  - Latency may be an issue in e.g. control systems
  - Throughput limited by speed of one stage ➔ Fast
- Versatile: 8...16bits, 1...200MS/s
- Many analog circuit non-idealities can be corrected digitally
Pipeline ADC
Concurrent Stage Operation

- Stages operate on the input signal like a shift register
- New output data *every* clock cycle, but each stage introduces at least \( \frac{1}{2} \) clock cycle latency

![Pipeline ADC Diagram](image)

Digital output \((B_1 + B_2 + \ldots + B_k)\) Bits

Pipeline ADC
Latency

Note: One conversion per clock cycle & 7 clock cycle latency
[Analog Devices, AD 9226 Data Sheet]
Pipeline ADC

Digital Data Alignment

- Digital shift register aligns sub-conversion results in time

Cascading More Stages

- LSB of last stage becomes very small
- Impractical to generate several $V_{ref}$
- All stages need to have full precision
Pipeline ADC Inter-Stage Gain Elements

- Practical pipelines by adding inter-stage gain → use single $V_{ref}$
- Precision requirements decrease down the pipe
  - Advantageous for noise, matching (later)

Complete Pipeline Stage

- "Residue Plot"
- E.g.: $B=2$, $G=2^2 = 4$