Analog-to-Digital Conversion

- Background
- Types of ADCs
  - Flash
  - Integrating
  - Oversampled
  - Successive Approximation
  - Time-Based SAR ADC
- Trend
<table>
<thead>
<tr>
<th>Lecture No.</th>
<th>Date</th>
<th>Subject</th>
<th>Reference</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Jan 8 (T)</td>
<td>Introduction</td>
<td>Rby-Ch1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Jan 10 (Th)</td>
<td>Power, Energy Basics</td>
<td>Rby-Ch3</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Jan 15 (T)</td>
<td>Circuit level power optimization</td>
<td>Rby-Ch4</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Jan 17 (Th)</td>
<td>Systems level power optimization</td>
<td>Rby-Ch5</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Jan 22 (T)</td>
<td>continued</td>
<td>Rby-Ch5</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Jan 24 (Th)</td>
<td>Interconnects and clock signaling</td>
<td>Rby-Ch6</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Jan 29 (T)</td>
<td>Memristive computing</td>
<td>Invited lecture</td>
<td>Dr. S. Shin</td>
</tr>
<tr>
<td>8</td>
<td>Jan 31 (Th)</td>
<td><strong>Midterm exam</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Feb 5 (T)</td>
<td>Low power CAS</td>
<td>Rby-Ch8</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Feb 7 (Th)</td>
<td>Low power CAS</td>
<td>Rbt-Ch10</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Feb 12 (T)</td>
<td>Ultra low power/voltage design</td>
<td>Rby-Ch11</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Feb 14 (Th)</td>
<td>Ultra low power/CAD, Sub Vth circuits</td>
<td>Rby-Ch 11, etc.</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>Feb 19 (T)</td>
<td><strong>Project proposal presentation</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>Feb 21 (Th)</td>
<td>High Bandwidth I/O for memory</td>
<td>Invited lecture</td>
<td>Prof. C. Kim</td>
</tr>
<tr>
<td>15</td>
<td>Feb 26 (T)</td>
<td>Ultra low power ADC Circuits</td>
<td>Sarpeshkar Ch15</td>
<td>Dr. S. Shin</td>
</tr>
<tr>
<td>16</td>
<td>Feb 28 (Th)</td>
<td>Low power RF Telemetry</td>
<td>Sarpeshkar Ch18</td>
<td>Dr. S. Shin</td>
</tr>
<tr>
<td>17</td>
<td>Mar 5 (T)</td>
<td><strong>Project presentation</strong></td>
<td>Sarpeshkar Ch18</td>
<td>Dr. S. Shin</td>
</tr>
<tr>
<td>18</td>
<td>Mar 7 (Th)</td>
<td><strong>Continued</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>Mar 12 (T)</td>
<td>Low power design flows</td>
<td>Invited lecture</td>
<td>Dr. Chin-Chi Teng</td>
</tr>
<tr>
<td>20</td>
<td>Mar 14 (Th)</td>
<td>Course review</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Analog-to-Digital Conversion

- Everything in the physical world is an analog signal
  - Sound, light, temperature, pressure, etc.
- Need to convert continuous-time signals into discrete-time binary-coded form, which enables computer analysis and digital transmission of the signal.
A/D Conversion Process

- There are two related steps in A-to-D conversion:
  - Sampling
  - Quantization

- **Sampling**
  - The analog signal is extracted, usually at regularly spaced time instants.
  - The samples have real values.

- **Quantization**
  - The samples are quantized to discrete levels.
  - Each sample is represented as a digital value.
Fundamental Issues

- Most common way of performing A/D conversion
  - Sample signal uniformly in time
  - Quantize signal uniformly in amplitude

- Key questions
  - How fast do we need to sample?
  - How much "noise" is added due to amplitude quantization?
  - How can we reconstruct the signal back into analog form?
Analog Input Range

- What do the sample values represent?
  - Some fraction within the range of values

⇒ What range to use?

Range Too Small

Range Too Big

Ideal Range
Sampling Process

- Representing a **continuous time domain** signal at **discrete and uniform** time intervals
- Determines **maximum bandwidth** of sampled (ADC) signal (**Nyquist Criteria**)
- Frequency Domain- **“Aliasing”** for an ADC
Sampling

What is a suitable sampling period for a signal?
Sampling Theorem

An analogue signal $x(t)$ with frequencies of no more than $F_{\text{max}}$ can be reconstructed exactly from its samples if the sampling rate satisfies:

$$F_s \geq 2 \times F_{\text{max}}.$$  

Significance

- If maximum frequency of the signal is $F_{\text{max}}$, the sampling rate should be at least:

$$\text{Nyquist rate} = 2 \times F_{\text{max}}$$

- If the sampling rate is $F_s$, the maximum frequency in the signal must not exceed:

$$\text{Nyquist frequency} = \frac{1}{2} F_s$$
Quantization

DIGITAL OUTPUT

111
110
101
100
011
010
001

ANALOG INPUT

1/8  2/8  3/8  4/8  5/8  6/8  7/8

1 LSB
Quantization Process

• Quantization Process
  – Representing an analog signal having infinite resolution with a digital word having finite resolution
  – Determines Maximum Achievable Dynamic Range
  – Results in Quantization Error/Noise

Any analog input in this range gives the same digital output code
Sampling & Quantization

8 levels (n=3)

quantization error
analogue signal
quantized signal
sampling points

3-bit ADC
Ideal ADC “Quantizer”

- Accepts analog input & generates its digital representation
- Quantization step:
  \[ \Delta = 1 \text{ LSB} \]
- Full-scale input range:
  \[ -0.5\Delta \ldots (2^N-0.5)\Delta \]
- E.g. \( N = 3 \) Bits
  \[ \Rightarrow V_{FS} = -0.5\Delta \text{ to } 7.5\Delta \]
Quantization Error

- For an ideal ADC:
  - Quantization error is bounded by $-\Delta/2$ ... $+\Delta/2$ for inputs within full-scale range
Quantization Error

- Probability density function (PDF) **Uniformly** distributed from $-\Delta/2 \ldots +\Delta/2$ provided that:
  - Busy input
  - Amplitude is many LSBs
  - No overload
- Not Gaussian!

- Zero mean
- Variance

$$\bar{e^2} = \int_{-\Delta/2}^{\Delta/2} \frac{e^2}{\Delta} \, de = \frac{\Delta^2}{12}$$


Signal-to-Quantization Noise Ratio

- If certain conditions the quantization error can be viewed as being "random", and is often referred to as “noise”

- In this case, we can define a peak “signal-to-quantization noise ratio”, SQNR, for sinusoidal inputs:

\[
SQNR = \frac{1}{2} \left( \frac{2^N \Delta}{2} \right)^2 = 1.5 \times 2^{2N} \frac{\Delta^2}{12}
\]

\[
= 6.02N + 1.76 \text{ dB} \quad \text{Accurate for } N > 3
\]

- Real converters do not quite achieve this performance due to other sources of error:
  - Electronic noise
  - Deviations from the ideal quantization levels
Quantization Noise

- If the quantization noise is uncorrelated with the frequency of the AC input signal, the noise will be spread evenly over the Nyquist bandwidth of $F_s/2$.
- If, however, the input signal is locked to a sub-multiple of the sampling frequency, the quantization noise will no longer appear uniform, but as harmonics of the fundamental frequency.
## Resolution vs. Quantization

<table>
<thead>
<tr>
<th>Resolution, Bits (n)</th>
<th>$2^n$</th>
<th>LSB, mV (2.5V FS)</th>
<th>% Full Scale</th>
<th>ppm Full Scale</th>
<th>dB Full Scale</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>256</td>
<td>9.77</td>
<td>0.391</td>
<td>3906</td>
<td>-48.0</td>
</tr>
<tr>
<td>10</td>
<td>1024</td>
<td>2.44</td>
<td>0.098</td>
<td>977</td>
<td>-60.0</td>
</tr>
<tr>
<td>12</td>
<td>4096</td>
<td>0.610</td>
<td>0.024</td>
<td>244</td>
<td>-72.0</td>
</tr>
<tr>
<td>14</td>
<td>16,384</td>
<td>0.153</td>
<td>0.006</td>
<td>61</td>
<td>-84.0</td>
</tr>
<tr>
<td>16</td>
<td>65,536</td>
<td>0.038</td>
<td>0.0015</td>
<td>15</td>
<td>-96.0</td>
</tr>
<tr>
<td>18</td>
<td>262,164</td>
<td>0.0095</td>
<td>0.00038</td>
<td>3.8</td>
<td>-108.0</td>
</tr>
</tbody>
</table>
Non-Idealities: Offset & Gain Error

No Gain Error:
Zero Error = Offset Error

With Gain Error:
Offset Error = 0
Differential Non-Linearity (DNL)

- Differential Non-Linearity (DNL) is the deviation of an actual code width from the ideal 1 LSB code width.
- Results in narrow or wider code widths than ideal and can result in missing codes.
- Results in additive noise/spurs beyond the effects of quantization.
SNR Degradation due to DNL

- Degradation in dB:

\[
SQNR\_deg = 1.76 - 10 \log\left(\frac{1}{8} \cdot \frac{DNL^2}{1 + \frac{DNL}{12}}\right)
\]

Valid only for cases where with no missing codes

---

SNR Degradation [dB] vs. DNL [LSB]

Graph showing the relationship between SNR degradation in dB and DNL [LSB].
Integral Non-Linearity (INL)

- Integral Non-Linearity (INL) is the deviation of an actual code transition point from its ideal position on a straight line drawn between the end points of the transfer function.
- INL is calculated after offset and gain errors are removed.
- Results in additive harmonics and spurs.
ADC Dynamic Performance

- Harmonic Distortion, Worst Harmonic
- Total Harmonic Distortion (THD)
- Signal-to-Noise Ratio (SNR)
- Signal-to-Noise-and-Distortion Ratio (SNDR)
- Effective Number of Bits (ENOB)
- Spurious Free Dynamic Range (SFDR)
- Two-Tone Intermodulation Distortion
- Analog Bandwidth (Full-Power, Small-Signal)
- Etc.
SNR / SNDR / ENOB

- **SNR** (Signal-to-Noise Ratio, or Signal-to-Noise Ratio Without Harmonics)
  - The ratio of the rms signal amplitude to the mean value of the root-sum-squares (RSS) of all other spectral components, excluding the first five harmonics and dc

- **SNDR** (Signal-to-Noise-and-Distortion Ratio)
  - The ratio of the rms signal amplitude to the mean value of the root-sum-squares (RSS) of all other spectral components, including harmonics, but excluding dc

- **ENOB** (Effective Number of Bits)

\[
ENOB = \frac{SNDR - 1.76dB}{6.02}
\]
SFDR / THD / SNR

**SFDR** = The difference between the rms power of the fundamental and the largest spurious signal in a given bandwidth.

**THD** = The ratio of the rms sum of the first six harmonics to the amplitude of the fundamental.

**SNR** = The ratio of the rms value of the fundamental to the rms sum of all noise components in the Nyquist bandwidth (excluding harmonics).
ADC Architectures vs. Performance

- Oversampled & Serial
- Algorithmic
e.g. Succ. Approx.
- Subranging
e.g. Pipelined
- Folding & Interpolative
- Parallel &
Time Interleaved
Flash ADC

- $2^N - 1$ comparators form the digitizer array, where $N$ is the ADC resolution.
- Analog input is applied to one side of the comparator array, a 1 lsb reference ladder voltage is applied to the other inputs.
- The comparator array is clocked simultaneously and decides in parallel.
- Output logic converts from thermometer code to binary.
Flash ADC

• Advantages of Flash A/D converters
  – Fastest conversion speed (up to 1GSPS)
  – Low data latency

• Disadvantages
  – Higher power consumption
  – High capacitive input that is difficult to drive

• Typical Applications
  – Video digitization
  – High-speed data acquisition
Pipelined Sub-Ranging ADC

- Divided into discrete pipelined conversion stages

- Example
  - 1st Stage ADC: 6-bit Flash
  - 2nd Stage ADC: 7-bit Flash
  - Total 12-bits resolution (one bit used for error correction)
Pipelined Sub-Ranging ADC

• Advantages of Pipelined A/D converters
  – Higher resolution at high speed
  – Digitize wideband inputs

• Disadvantages
  – Higher power consumption
  – Larger die area

• Typical Applications
  – Communications
  – Medical imaging
  – Radar
Successive Approximation ADC

“Recursive” One-Bit Sub-Ranging Architecture

\[ V_{\text{in}} \rightarrow \text{S/H} \rightarrow \text{Comp} \rightarrow V_C \rightarrow \text{Control Circuit} \rightarrow \text{SAR} \rightarrow \text{Digital Output} \]

\[ V_{\text{DAC}} \]

\[ V_{\text{DAC}} \rightarrow \text{D/A Converter} \]

\[ V_{\text{in}}, V_{\text{DAC}} \rightarrow V_{\text{ref}}, 3V_{\text{ref}}, \frac{V_{\text{ref}}}{4}, \frac{V_{\text{ref}}}{2}, V_{\text{ref}} \]

\[ t \]

\[ \text{Reset} d_1 = 1 \quad d_2 = 1 \quad d_3 = 1 \quad d_4 = 0 \quad d_5 = 1 \quad d_6 = 0 \]

(\text{MSB}) (\text{LSB})
Successive Approximation ADC

- Algorithmic type ADC
- Based on binary search over DAC output
Successive Approximation ADC

- High accuracy achievable (16+ Bits)
- Required N clock cycles for N-bit conversion (much faster than slope type)
- Moderate speed proportional to N (typically MHz range)
Goal: Find digital value of $V_{in}$

- 8-bit ADC
- $V_{in} = 7.65$
- $V_{full\ scale} = 10$
SAR ADC Example

- **MSB ➔ LSB**
  - Average high/low limits
  - Compare to $V_{in}$
    - $V_{in} > \text{Average} \Rightarrow \text{MSB} = 1$
    - $V_{in} < \text{Average} \Rightarrow \text{MSB} = 0$

- **Bit 7**
  - $(V_{\text{full scale}} + 0)/2 = 5$
  - $7.65 > 5 \Rightarrow \text{Bit 7} = 1$

$V_{\text{full scale}} = 10, V_{in} = 7.65$
**SAR ADC Example**

- **MSB → LSB**
  - Average high/low limits
  - Compare to $V_{in}$
    - $V_{in} >$ Average $→$ MSB = 1
    - $V_{in} <$ Average $→$ MSB = 0

- **Bit 6**
  - $(V_{full\text{ scale}} + 5)/2 = 7.5$
  - $7.65 > 7.5$ $→$ Bit 6 = 1

$V_{full\text{ scale}} = 10$, $V_{in} = 7.65$
SAR ADC Example

• MSB \(\rightarrow\) LSB
  • Average high/low limits
  • Compare to \(V_{\text{in}}\)
    • \(V_{\text{in}} >\) Average \(\rightarrow\) MSB = 1
    • \(V_{\text{in}} <\) Average \(\rightarrow\) MSB = 0

• Bit 5
  • \((V_{\text{full scale}} + 7.5)/2 = 8.75\)
  • \(7.65 < 8.75 \rightarrow\) Bit 5 = 0

\[
\begin{array}{c}
1 \\
1 \\
0 \\
\end{array}
\]
SAR ADC Example

- MSB → LSB
  - Average high/low limits
  - Compare to $V_{in}$
    - $V_{in} > \text{Average} \rightarrow \text{MSB} = 1$
    - $V_{in} < \text{Average} \rightarrow \text{MSB} = 0$

- Bit 4
  - $(8.75+7.5)/2 = 8.125$
  - $7.65 < 8.125 \rightarrow \text{Bit 4} = 0$

$V_{in} = 7.65$

\[ \begin{array}{cccccc}
1 & 1 & 0 & 0 & 0 & 0
\end{array} \]
SAR ADC Example

• MSB → LSB
  • Average high/low limits
  • Compare to $V_{in}$
    • $V_{in} >$ Average → MSB = 1
    • $V_{in} <$ Average → MSB = 0

• Bit 3
  • $(8.125+7.5)/2 = 7.8125$
  • $7.65 < 7.8125$ → Bit 3 = 0

$V_{in} = 7.65$

| 1 | 1 | 0 | 0 | 0 | 0 |

[39/62]
**SAR ADC Example**

- **MSB ➔ LSB**
  - Average high/low limits
  - Compare to $V_{in}$
    - $V_{in} >$ Average ➔ MSB = 1
    - $V_{in} <$ Average ➔ MSB = 0

- **Bit 2**
  - $(7.8125+7.5)/2 = 7.65625$
  - $7.65 < 7.65625$ ➔ Bit 2 = 0

| 1 | 1 | 0 | 0 | 0 | 0 | 0 |

$V_{in} = 7.65$
SAR ADC Example

- **MSB → LSB**
  - Average high/low limits
  - Compare to $V_{in}$
    - $V_{in} >$ Average → MSB = 1
    - $V_{in} <$ Average → MSB = 0

- **Bit 1**
  - $(7.65625+7.5)/2 = 7.578125$
  - $7.65 > 7.578125$ → Bit 1 = 1

![Diagram of SAR ADC Example]

$V_{in} = 7.65$

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>1</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
</table>

[41/62]
SAR ADC Example

• MSB \( \Rightarrow \) LSB
  • Average high/low limits
  • Compare to \( V_{in} \)
    • \( V_{in} > \) Average \( \Rightarrow \) MSB = 1
    • \( V_{in} < \) Average \( \Rightarrow \) MSB = 0

• Bit 0
  • \((7.65625 + 7.578125)/2 = 7.6171875\)
  • \(7.65 > 7.6171875 \Rightarrow \) Bit 0 = 1

\[ \begin{array}{cccccccc}
1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\
\end{array} \]
SAR ADC Example

- $11000011_2 = 195_{10}$
- 8-bits, $2^8 = 256$
- Digital Output
  - $195/256 = 0.76171875$
- Analog Input
  - $7.65/10 = 0.765$
- Resolution
  - $(V_{\text{max}} - V_{\text{min}})/2^n \Rightarrow 10/256 = 0.039$
Single Slope Integrating ADC

“Time-based” architecture by vtg-to-time & time-to-digital conversions

- Counter starts counting @ $V_{\text{Ramp}} = 0$
- Counter stops counting for $V_{\text{IN}} = V_{\text{Ramp}}$
  $\rightarrow$ Counter output proportional to $V_{\text{IN}}$
Single Slope Integrating ADC

\[ v_R = \frac{1}{CR} V_{\text{REF}} t \]

\[ \frac{1}{CR} V_{\text{REF}} T = V_{\text{IN}} \]

\[ T = RC \frac{V_{\text{IN}}}{V_{\text{REF}}} \]

\[ k \times T_S = T \]

\[ k = \frac{RC}{T_S} \frac{V_{\text{IN}}}{V_{\text{REF}}} \]
Single Slope Integrating ADC

• **Advantages of Single Slope A/D converters**
  – Low complexity & simple architecture
  – INL depends on ramp linearity, independent on components matching
  – Inherently monotonic

• **Disadvantages**
  – Slow ($2^N$ cycles for N-bit conversion)
  – Hard to generate precise ramp
  – Need to calibrate ramp slope vs. $V_{IN}$
Dual Slope Integrating ADC

- First: $V_{IN}$ is integrated for a fixed time ($2^N \times T_{CLK}$)
  \[ V_o = 2^N \times T_{CLK} \frac{V_{IN}}{\tau_{intg}} \]
- Next: $V_o$ is de-integrated with $V_{REF}$ until $V_o = 0$
  \[ \text{Counter output} = 2^N \frac{V_{IN}}{V_{REF}} \]
Dual Slope Integrating ADC

Example

- $V_{\text{in}}$
- $R$
- $C$
- $V_{\text{REF}}$
- $V_{C}$
- $\text{Comp}$
- $\text{SW1}$
- $\text{SW2}$
- $\text{Integrator}$
- $\text{Clock}$
- $\text{Control logic}$
- $\text{N-bit Counter}$
- $\text{Reset}$
- $\text{overflow}$
- $d_0d_1$
- $d_{N-1}$
- (Digital output)
Dual Slope Integrating ADC

- **Charging period**
- **Discharging period**
- **Variable slope**
- **Constant slope**

- **Fixed integration period, \( T_1 \)**
- **Variable integration period, \( T_2 \)**

The diagram illustrates the dual-slope integrating ADC process, showing the charging and discharging periods, as well as the variable and constant slopes during the integration process.
Dual Slope Integrating ADC

First Integration Period, $T_1$ (-$V_{in}$)

$$V_c = -\frac{1}{C} \int_0^{T_1} -\frac{V_{IN}}{R} \, dt = \frac{V_{IN} \cdot T_1}{RC}$$

Second De-Integration Period, $T_2$ ($V_{REF}$)

$$V_c = \frac{V_{IN} \cdot T_1}{RC} - \frac{V_{REF} \cdot T_2}{RC}$$

If $V_c$ reaches zero,

$$V_c = \frac{V_{IN} \cdot T_1}{RC} - \frac{V_{REF} \cdot T_2}{RC} = 0 \Rightarrow V_{IN} \cdot T_1 = V_{REF} \cdot T_2$$

$$k = \frac{V_{in}}{V_{REF}} \cdot 2^N$$

Eliminated R & C dependency !! (Accuracy Enhanced)
Dual Slope ADC

• Advantages of Dual Slope A/D converters
  – Accuracy is independent of integrator time-constant
  – Insensitive to most of linear error sources
  – High accuracy achievable (16+ bits)

• Disadvantages
  – Slow (maximum $2 \times 2^N$ cycles for $N$-bit conversion)
  – Integrator offset results in ADC offset
  – Finite opamp gain $\rightarrow$ INL
Oversampling Sigma-Delta ADC

A

Nyquist Operation

ADC

B

Oversampling
+ Digital Filter
+ Decimation

ADC → DIGITAL FILTER → DEC

C

Oversampling
+ Noise Shaping
+ Digital Filter
+ Decimation

ΣΔ MOD → DIGITAL FILTER → DEC

Quantization Noise = $\Delta / \sqrt{12}$, where $\Delta = 1$ LSB

Digital Filter

Removed Noise

Removed Noise

Kf_s

Kf_s
Sigma-Delta ADC

• Advantages of Oversampled converters
  – High resolutions and accuracy (up to 24-bits)
  – Excellent DNL and INL performance
  – Noise shaping capability

• Disadvantages
  – Limited input bandwidth
  – Slower sampling rates

• Typical Applications
  – Precision data acquisition and measurement
  – Medical instrumentation
Time-Based SAR ADC
Time-Based SAR ADC

- 2 MSBs
  - Identical to the standard dual slope converter
  - Quantization with two bit counter
- Lower N-2 bits
  - Equivalent to recursive time-based SAR conversion
  - Subtraction in time, and amplification through time-to-voltage & voltage-to-time conversions
Time-Based SAR ADC

- Positive and Negative index counting
  - Quantize $T_{\text{resB}}$ as opposed to the actual residue, $T_{\text{res}}$.
  - First stage: ‘negative’ index for the negative residue, $T_{\text{res}}$.
  - Second stage: ‘positive’ index for the negative of the negative residue in First stage.
  - Third, Forth, ...
  - After completing N-2 bit indexing, invert all negatively indexed bits.
- Conversion Time: $\sim 2N$ cycles
Comparator Delay Cancellation

- Comparator delay “added” during Time-to-Voltage conversion is “subtracted” during Voltage-to-Time conversion step.
Problem with Small Residues

- Issue with small time residue (ε)
  - For time-to-voltage conversion, the residue needs to be larger than the comparator delay and pulse-width (T_{resB} > T_{cd+pw}).
  - Which requires an extra clock period for small residue cases, such that T_{resB} = ε + T_{clk} > T_{cd+pw}.
  - The average conversion time is increased by additional 3N.
Issues with Comparator Offset

- If the comparator offset is static, the offset is amplified and accumulated through the SAR process.
  - Degrades output SNR

Results Comparison

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Technology</td>
<td>0.35 µm</td>
<td>0.8 µm</td>
<td>0.6 µm</td>
<td>1.2 µm</td>
<td>0.25 µm</td>
</tr>
<tr>
<td>ENOB (Input Freq.)</td>
<td>11.0 (1 kHz)</td>
<td>13.4 (0.1 kHz)</td>
<td>8.5 (10 kHz)</td>
<td>7.9 (1 kHz)</td>
<td>7.0 (4.61 kHz)</td>
</tr>
<tr>
<td>Sampling Rate</td>
<td>31.25 kHz</td>
<td>10 kHz</td>
<td>200 kHz</td>
<td>50 kHz</td>
<td>100 kHz</td>
</tr>
<tr>
<td>Power Dissipation</td>
<td>75 µW</td>
<td>50 mW</td>
<td>7 mW</td>
<td>0.34 mW</td>
<td>3.1 µW</td>
</tr>
<tr>
<td>Active Area</td>
<td>0.45 mm²</td>
<td>1.68 mm²</td>
<td>5.5 mm²</td>
<td>&lt; 3.24 mm²</td>
<td>0.053 mm²</td>
</tr>
</tbody>
</table>

   Time-based SAR Type, 31KS/s, 12bit resolution
   SAR Type, 10KS/s, 14bit resolution
    Bias-based Pipeline Type, 200KS/s, 10bit
    SAR Type (resistor array), 50KS/s, 8bit
    SAR Type (capacitor array), 100KS/s, 8bit
## Comparison with Other Low-Power ADCs

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Technology (um)</td>
<td>0.35</td>
<td>0.18</td>
<td>0.25</td>
<td>0.065</td>
<td>0.18</td>
</tr>
<tr>
<td>ENOB</td>
<td>11</td>
<td>8.6</td>
<td>7.6</td>
<td>8</td>
<td>9.4</td>
</tr>
<tr>
<td>Sampling Rate</td>
<td>31.25 KHz</td>
<td>150 KHz</td>
<td>100 KHz</td>
<td>1 MHz</td>
<td>100 KHz</td>
</tr>
<tr>
<td>Power Dissipation</td>
<td>75 uW</td>
<td>30 uW</td>
<td>3.1 uW</td>
<td>1.9 uW</td>
<td>3.8 uW</td>
</tr>
<tr>
<td>Energy consumption</td>
<td>1.17 pJ</td>
<td>0.53 pJ</td>
<td>0.16 pJ</td>
<td>4.4 fJ</td>
<td>0.56 pJ</td>
</tr>
<tr>
<td>/ quantization level</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Active Area (mm$^2$)</td>
<td>0.45</td>
<td>0.11</td>
<td>0.053</td>
<td></td>
<td>0.24</td>
</tr>
</tbody>
</table>

[2] A 0.5V 1uW successive approximation ADC (JSSC 2003)
ADC Architectures vs. Performance

![Graph showing trend in energy per conversion step and cost versus performance for different ADC architectures.](image)

**Figure 2.** IEEE Figure of Merit, $FOM_c$, as a function of time for ADCs introduced in the given year. The energy per conversion step is decreasing by an order of magnitude per decade of time.

**[FIG7]** Cost versus performance $P$ for different structures.