EE247
Lecture 20
ADC Converters (continued)
  – Comparator design (continued)
    • Latched comparators
    • Comparator architecture examples
  – Techniques to reduce flash ADC complexity
    • Interpolating
    • Folding
    • Interpolating & folding
  – Interleaved ADCs
  – Multi-Step ADCs
    • Two-Step flash
    • Pipelined ADCs
      – Effect of sub-ADC, sub-DAC, gain stage non-idealities on overall ADC performance

Project

• Design & simulate an ADC
  – ENOB=6bit for f_{signal}<10MHz
  – Signal bandwidth 0 to 100MHz
  – Architecture of your choice targeted for minimum power dissipation
  – Description posted in the homework section
  – Teams of two preferred
  – Report due Dec. 3\(^{rd}\)
  – Powerpoint presentation ~5min/person in class on Dec. 8\(^{th}\)
CMOS Latched Comparators

Comparator amplification need not be linear
→ can use a latch → regeneration

Latch → Amplification + positive feedback

CMOS Latched Comparators
Small Signal Model

Latch can be modeled as a:
→ Single-pole amp + positive feedback

Small signal ac half circuit
CMOS Latched Comparator

Latch Delay

\[ g_m V = \frac{V}{R_C} + C \frac{dV}{dt} \]

\[ \frac{g_m}{C} \left(1 - \frac{1}{g_m R_C}\right) \frac{dV}{dt} = \frac{g_m}{C} \left(1 - \frac{1}{g_m R_C}\right) \frac{dV}{V} \]

Integrating both sides:

\[ \frac{g_m}{C} \left(1 - \frac{1}{g_m R_C}\right) \int_{t_1}^{t_2} \frac{dV}{dt} = \int_{t_1}^{t_2} \frac{1}{V} dV \]

\[ \left(1 - \frac{1}{x}\right) dx = \ln x |_{t_1}^{t_2} = \ln a - \ln b = \ln \frac{a}{b} \]

Latch Delay:

\[ t_0 = t_2 - t_1 = \frac{C}{g_m} \left(1 - \frac{1}{g_m R_C}\right) \ln \left(\frac{V_2}{V_1}\right) \]

For \( g_m R_C \gg 1 \)

\[ t_0 = \frac{C}{g_m} \ln \left(\frac{V_2}{V_1}\right) \]

Latch-Only Comparator

- Much faster compared to cascade of open-loop amplifiers

- Main problem associated with latch-only comparator topology:
  - High input-referred offset voltage (as high as 100mV!)

- Solution:
  - Use preamplifier to amplify the signal and reduce overall input-referred offset
Pre-Amplifier + Latch
Overall Input-Referred Offset

\[
\sigma_{\text{Input-Ref Offset}} = \sqrt{\sigma_{\text{Vos, Preamp}}^2 + \frac{1}{A_{\text{Preamp}}} \sigma_{\text{Vos, Latch}}^2}
\]

Example: \( \sigma_{\text{Vos, Preamp}} = 4mV \) & \( \sigma_{\text{Vos, Latch}} = 50mV \) & \( A_{\text{Preamp}} = 10 \)

\[
\sigma_{\text{Input-Ref Offset}} = \sqrt{4^2 + \frac{1}{100} \cdot 50^2} = 6.4mV
\]

Latch offset attenuated by preamp gain when referred to preamp input.
Assuming the two offset sources are uncorrelated:

Pre-Amplifier Tradeoffs

- Example:
  - Latch offset 50 to 100mV
  - Preamp DC gain 10X
  - Preamp input-referred latch offset 5 to 10mV
  - Input-referred preamplifier offset 2 to 10mV
  - Overall input-referred offset 5.5 to 14mV

→ Addition of preamp reduces the latch input-referred offset reduced by ~7 to 9X → allows extra 3-bit resolution for ADC!
Comparator Preamplifier Gain-Speed Tradeoffs

• Amplifier maximum Gain-Bandwidth product \( f_u \) or a given technology, typically a function of maximum device \( f_t \)

\[ f_u = \text{unity gain frequency}, \quad f_0 = -3dB \text{ frequency} \quad \text{&} \quad \tau_0 = \text{settling time} \]

\[ f_0 = \frac{f_u}{A_{\text{preamp}}} \]

For example assuming preamp has a gain of 10:

\[ f_0 = \frac{1GHz}{10} = 100MHz \]

\[ \tau_0 = \frac{1}{2\pi f_0} = \frac{A_{\text{preamp}}}{2\pi f_u} = 1.6\pi \text{ sec} \]

- Tradeoff:
  • To reduce the effect of latch offset \( \rightarrow \) high preamp gain desirable
  • Fast comparator \( \rightarrow \) low preamp gain

  \( \rightarrow \) Choice of preamp gain: compromise speed v.s. input-referred latch offset

Latched Comparator

Important features:

– Maximum clock rate \( f_s \) \( \rightarrow \) settling time, slew rate, small signal bandwidth
– Resolution\( \rightarrow \) gain, offset
– Overdrive recovery
– Input capacitance (and linearity of input capacitance!)
– Power dissipation
– Input common-mode range and CMR
– Kickback noise

– ...
CMOS Preamplifier + Latch Type Comparator
Delay in Response

Latch delay previously found:

\[ \tau_D = \frac{C}{g_m} \ln \left( \frac{V_C}{V_1} \right) \]

Assuming gain of \( A_c \) for the preamplifier then: \( V_1 = A_c \times V_{in} \)

\[ \tau_D = \frac{C}{g_m} \ln \left( \frac{V_0}{A_c V_{in}} \right) \]

Latched Comparator Including Preamplifier Example

Preamplifier gain:

\[ A_c = \frac{g_{M1}}{g_{M3}} = \frac{(V_{GS1} - V_{TH})}{(V_{GS2} - V_{TH})} \]

Comparator delay:
(for simplicity, preamp delay ignored)

\[ \tau_0 = \frac{C}{g_m} \ln \left( \frac{V_o}{A_c V_{in}} \right) \]
Comparator Dynamic Behavior

![Comparator Dynamic Behavior Diagram]

Comparator Resolution

![Comparator Resolution Diagram]

Note: For small $V_{\text{in}}$ the comparator may not be able to make a decision within the allotted time → output ambiguous.
Comparator Voltage Transfer Function
Non-Idealities

\[ V_{\text{out}} \]
\[ V_{\text{in}} \]
\[ V_{\text{Offset}} \]

-0.5LSB 0.5LSB

\( V_{\text{Offset}} \rightarrow \) Comparator offset voltage
\( \varepsilon \rightarrow \) Meta-Stable region (output ambiguous)

CMOS Comparator Example
Flash ADC

- Flash ADC: 8bits, \( \pm 1/2\text{LSB} \) INL @ \( fs=15\text{MHz} \) (\( V_{\text{ref}}=3.8V \), \( \text{LSB} \approx 15\text{mV} \))
- No offset cancellation

Comparator with Auto-Zero


Flash ADC
Comparator with Auto-Zero

Flash ADC
Comparator with Auto-Zero

\[ V_i = A_{P1} \cdot A_{P2} \left( (V_{in1} - V_{ref1}) - (V_{in2} - V_{ref2}) - V_{offset} \right) \]

Substituting for \((V_{in2} - V_{ref2})\) from previous cycle:

\[ V_i = A_{P1} \cdot A_{P2} \left( (V_{in1} - V_{ref1}) - (V_{in2} - V_{ref2}) \right) \]

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 we will see later are tolerant of high offset)

- Note: M1, M2, M11, M12 operate in triode mode

- M11 & M12 added to vary comparator threshold

- Conductance at node X is sum of $G_M$ & $G_{M11}$

Comparator Example (continued)

- M1, M2, M11, M12 operate in triode mode with all having equal L.
- Conductance of input devices:
  \[
  G_1 = \frac{\mu C_{ox} W_1}{L} (V_{th11} - V_g) + W_1 (V_R - V_{th})
  \]
  \[
  G_2 = \frac{\mu C_{ox} W_2}{L} (V_{th12} - V_g) + W_1 (V_R - V_{th})
  \]
  \[
  \Delta G = \frac{\mu C_{ox} W_1}{L} (V_{th11} - V_{th12}) + W_1 (V_R - V_{th})
  \]
- To 1st order, for \( W_1 = W_2 \) & \( W_{th1} = W_{th2} \)
  \( V_{th} = \frac{W_{th1} V_R}{W_1} \)
  \( V_R \) fixed, \( W_{th} \) varied from comparator to comparator.
  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 \( V_R \) & \( \phi_{2B} \) is also high \( \rightarrow \) current diverted to latch \( \rightarrow \) comparator output in hold mode.
- \( \phi_2 \) high \( \rightarrow \) \( C_S \) connected to S/Hout & comparator input (VR-S/Hout), current sent to preamp \( \rightarrow \) comparator in amplify mode.

Bipolar Comparator Example

- Used in 8bit 400Ms/s & 6bit 2Gb/s flash ADC
- Signal amplification during φ_1 high, latch operates when φ_1 low
- Input buffers suppress kick-back & input current
- Separate ground and supply buses for front-end preamp → kick-back noise reduction


Reducing Flash ADC Complexity

E.g. 10-bit "straight" flash
- Input range: 0 ... 1V
- LSB = Δ: ~ 1mV
- Comparators: 1023 with offset < 1/2 LSB
- Assuming Cin for each comparator is 0.1pF & power 3mW
  - Total input capacitance: 1023 * 100fF = 102pF
  - Power: 1023 * 3mW = 3W
  - 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 → Improved DNL

Flash ADC

Preamp Output

Zero crossings (to be detected by latches) at $V_{in} = \frac{V_{ref1}}{\Delta}$

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

```
Vin  
\arrow{Vin} \rightarrow \text{Preamp1} \\
1 \Delta Vref1 \\
\text{Preamp2} \\
2 \Delta Vref2 \\
A1 \rightarrow \text{Vin} \\
A2 \rightarrow \text{Output}
```

**Differential Preamp Output**

Differential output crossings

@ $V_{in} =$

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

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

$A_1$+$A_2$ = $A_1$+$A_2$

$\rightarrow$ cross zero at:

\[ V_{ref2} = 0.5^*(1+2) \Delta = 1.5\Delta \]
Interpolation in Flash ADC

Half as many reference voltages and preamps
Interpolation factor: x2

Example: For 10-bit straight Flash ADC need $2^8=1024$ preamps compared with $2^8-1=512$ for x2 interpolation

Possible to accomplish higher interpolation factor
$\quad\rightarrow$ Interpolation at the output of preamps

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)

$V_{o,1.5} = (V_{o,1} + V_{o,2})/2$

Interpolation in Flash ADC

Preamp Output Interpolation

With 2 sets of interpolation resistors at each preamp outputs → three extra intermediate points → 2 extra bits

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)

Ref: H. Kimura et al. “A 10-b 300-MHz Interpolated-Parallel AD Converter,” JSSC April 1993, pp. 438-446
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 w/o output saturation $> \Delta$

Sets upper bound on preamp gain: $\text{Preamp gain} < \frac{V_{DD}}{\Delta}$
Interpolated-Parallel ADC

- 10-bit overall resolution:
- 7-bit flash (127 preamps and Vref 128 resistors) & x8 interpolation
- Use of Gray Encoder minimizes effect of sparkle code & metastability


Measured Performance

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resolution</td>
<td>10 b (7+3)</td>
</tr>
<tr>
<td>Maximum conversion frequency</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>56-59 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 μm bipolar:ft=25GHz</td>
</tr>
</tbody>
</table>

Low input capacitance

Interpolation Summary

- Consecutive preamp transfer curve linear region need to have overlap $\rightarrow$ 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 $\rightarrow$ 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
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

Fold 1 → $V_{out} = V_{in}$
Fold 2 → $V_{out} = -V_{in} + V_{FS}/2$
Fold 3 → $V_{out} = +V_{in} - V_{FS}/2$
Fold 4 → $V_{out} = -V_{in} + V_{FS}$

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

V_{ref1} < V_{ref2} < V_{ref3} < V_{ref4}
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

\[ V_{in} \]

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

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

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

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

Logic
LSB bits
(to be combined with MSB bits)

Parallel Folder Outputs

- 4 folders with 4 folds each
- 16 zero crossings
- → 4 LSB bits
- Higher resolution
  - More folders
    → Large complexity
  - Interpolation

![Graph showing folder outputs](image-url)
Folding & Interpolation

Folder 4
\[ V_{\text{ref}} + 3/4 \times \Delta \]
Folder 3
\[ V_{\text{ref}} + 2/4 \times \Delta \]
Folder 2
\[ V_{\text{ref}} + 1/4 \times \Delta \]
Folder 1
\[ V_{\text{ref}} + 0/4 \times \Delta \]

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
Two-Step Example: (2+2)Bits

- Using only one ADC: output contains large quantization error
- "Missing voltage" or "residue" (-ε_q1)
- Idea: Use second ADC to quantize and add -ε_q1

Two Stage Example

- Use DAC to compute missing voltage
- Add quantized representation of missing voltage
- Why does this help? How about ε_q2?
- Since maximum voltage at input of the 2nd ADC is Vref1/4 then for 2nd ADC Vref2 = Vref1/4 and thus ε_q2 = ε_q1/4 = Vref1/16 → 4bit overall resolution
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

$$\varepsilon_{q2} = \frac{V_{ref2}}{2^2} = \frac{V_{ref1}}{2^2 \cdot 2^2}$$
Two-Stage (2+2) ADC Transfer Function

Residue or Multi-Step Type ADC

- **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 1/2 LSB
  2. Speed penalty → Need at least 1 clock cycle per extra series stage to resolve one sample
Solution to Issue (1)
Reducing Precision Required for Fine ADC

- 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

\[ \text{Vin} \rightarrow \text{Coarse} \rightarrow \varepsilon_{q1} \rightarrow \text{Fine} \rightarrow \text{Dout} = V_n + \varepsilon_{q1} - \varepsilon_{q1} + \varepsilon_{q2} \]

Solution to Issue (2)
Increasing ADC Throughput

- Conversion time significantly decreased by employing T/H between stages
  - All stages busy at all times \(\Rightarrow\) 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 \(n-1\)

\[ \text{Vin} \rightarrow \text{Coarse} \rightarrow \varepsilon_{q1} \rightarrow \text{T/H} \rightarrow \text{Fine} \rightarrow \text{Dout} = V_n + \varepsilon_{q1} - \varepsilon_{q1} + \varepsilon_{q2} \]
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 ÂDCs each 1-bit only!)
- Each stage performs coarse A/D conversion and computes its quantization error, or "residue"
- All stages operate concurrently
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
- One important feature of pipeline ADC: 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 ½ clock cycle latency
Pipeline ADC

Latency

Note: One conversion per clock cycle & 8 clock cycle latency

[Analog Devices, AD 9226 Data Sheet]

Digital Data Alignment

- Digital shift register aligns sub-conversion results in time
Cascading More Stages

- LSB of last stage becomes very small
- All stages need to have full precision
- Impractical to generate several $V_{ref}$

Pipeline ADC

Inter-Stage Gain Elements

- Practical pipelines by adding inter-stage gain $\rightarrow$ use single $V_{ref}$
- Precision requirements decrease down the pipe
  - Advantageous for noise, matching (later), power dissipation
**Complete Pipeline Stage**

Vin → \( \frac{-G}{q_1} \) → Vres

"Residue Plot"

E.g.: 
- \( B = 2 \)
- \( G = 2^2 = 4 \)

Note: None of the blocks have ideal performance.

Question: What is the effect of the non-idealities?

---

**Pipeline ADC Errors**

- Non-idealities associated with sub-ADCs, sub-DACs and gain stages → error in overall pipeline ADC performance

- Need to find means to tolerate/correct errors

- Important sources of error
  - Sub-ADC errors: comparator offset
  - Gain stage offset
  - Gain stage gain error
  - Sub-DAC error
Pipeline ADC Single Stage Model

\[ V_{\text{res}} = G \varepsilon_q \]

Pipeline ADC Multi-Stage Model

\[ D_{\text{out}} = V_{\text{in,ADC}} + \varepsilon_{\text{q1}} \left( l - \frac{G_1}{G_{d1}} \right) + \varepsilon_{\text{q2}} \left( l - \frac{G_2}{G_{d2}} \right) + \ldots + \frac{\varepsilon_{\text{qn}}}{\prod G_{ij}} \left( l - \frac{G_{n-1}}{G_{d(n-1)}} \right) + \frac{1}{\prod G_{ij}} \]
Pipeline ADC Model

- If the "Analog" and "Digital" gain/loss is precisely matched:

\[
D_{\text{out}} = V_{\text{in,ADC}} + \frac{E_{\text{qu}}}{{ \prod_{j=1}^{n-1} G_j }},
\]

\[
D.R. = 20 \log_{10} \frac{V_{\text{FS, rms}}}{V_{\text{Quant, rms}}} = 20 \log_{10} \frac{V_{\text{sat}}}{V_{\text{ref}}} = 20 \log_{10} \left( \frac{3}{\sqrt{2}} \times 2^n \times \prod_{j=1}^{n} G_j \right)
\]

- Overall conversion error does not (directly) depend on sub-ADC errors!
- Only error term in \(D_{\text{out}}\) contains quantization error associated with the last stage

\[
B_{\text{ADC}} = \log_2 \left( 2^n \times \prod_{j=1}^{n} G_j \right)
\]

Pipeline ADC Observations

- The aggregate ADC resolution is independent of sub-ADC resolution!
- Effective stage resolution \(B_j = \log_2(G_j)\)
- Overall conversion error does not (directly) depend on sub-ADC errors!
- Only error term in \(D_{\text{out}}\) contains quantization error associated with the last stage
- So why do we care about sub-ADC errors?
  - Go back to two stage example