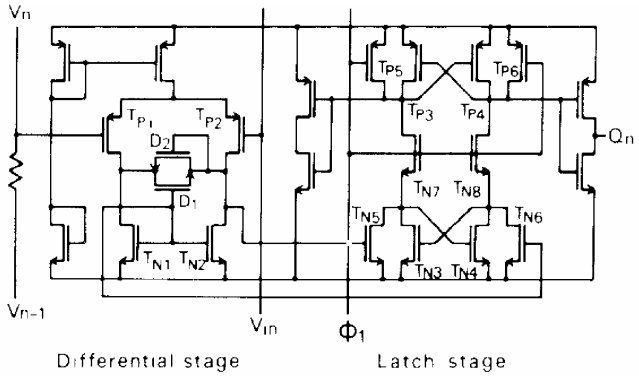
EE247
Lecture 21

ADC Converters

- Comparator design (continued)
  - Comparator architecture examples
- Techniques to reduce flash ADC complexity
  - Interpolating
  - Folding
  - Interpolating & folding
  - Multi-Step ADCs
    - Two-step flash
    - Pipelined ADCs
      - Effect of sub-ADC, sub-DAC, gain stage non-idealities on overall ADC performance

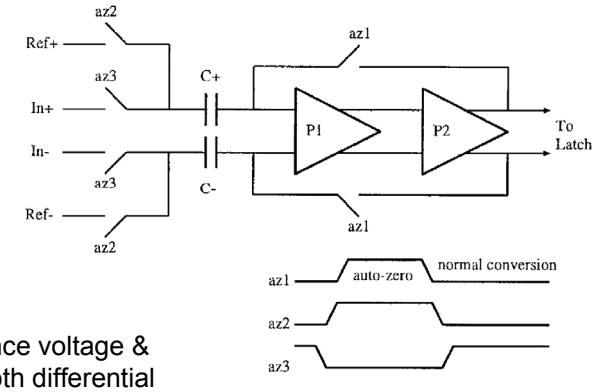
CMOS Comparator Example



- Flash ADC: 8bits, +-1/2LSB INL @ fs=15MHz (Vref=3.8V, LSB~15mV)
- No offset cancellation

Ref: A. Yukawa, "A CMOS 8-Bit High-Speed A/D Converter IC," JSSC June 1985, pp. 775-9

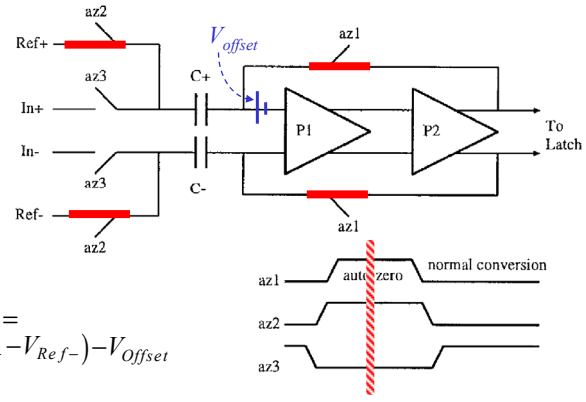
Comparator with Auto-Zero



Note:
Reference voltage &
input both differential

Ref: I. Mehr and L. Singer, "A 500-Msample/s, 6-Bit Nyquist-Rate ADC for Disk-Drive Read-Channel Applications," JSSC July 1999, pp. 912-20.

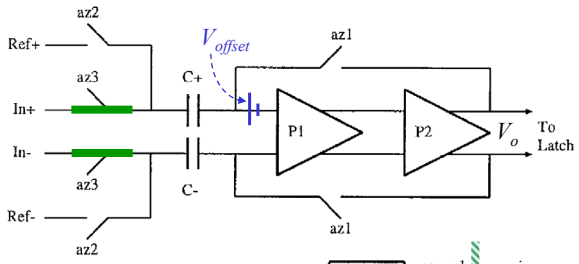
Flash ADC
Comparator with Auto-Zero



$$V_{C+} - V_{C-} = (V_{Ref+} - V_{Ref-}) - V_{Offset}$$

Ref: I. Mehr and D. Dalton, "A 500-Msample/s, 6-Bit Nyquist-Rate ADC for Disk-Drive Read-Channel Applications," JSSC July 1999, pp. 912-20.

Flash ADC
Comparator with Auto-Zero

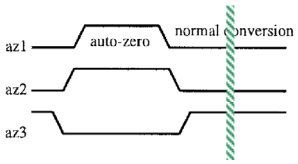


$$V_o = A_{P1} * A_{P2}\left[(V_{In+} - V_{In-}) - (V_{C+} - V_{C-}) - V_{Offset}\right]$$

Substituting for $(V_{C+} - V_{C-})$ *from previous cycle:*

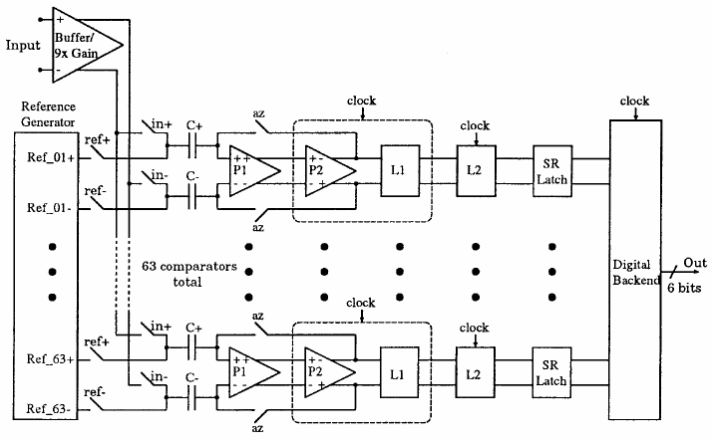
$$V_o = A_{P1} * A_{P2}\left[(V_{In+} - V_{In-}) - (V_{Ref+} - V_{Ref-})\right]$$

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



Ref: I. Mehr and D. Dalton, "A 500-Msample/s, 6-Bit Nyquist-Rate ADC for Disk-Drive Read-Channel Applications," JSSC July 1999, pp. 912-20.

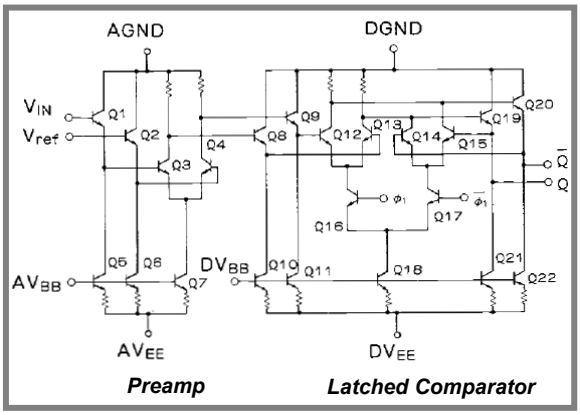
Flash ADC
Using Comparator with Auto-Zero



Ref: I. Mehr and D. Dalton, "A 500-Msample/s, 6-Bit Nyquist-Rate ADC for Disk-Drive Read-Channel Applications," JSSC July 1999, pp. 912-20.

Bipolar Comparator Example

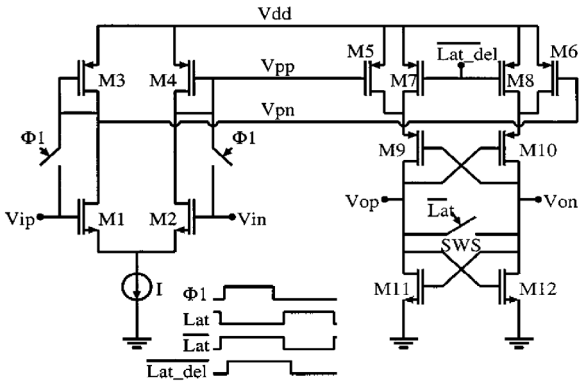
- Used in 8bit 400Ms/s & 6bit 2Gb/s flash ADC
- Signal amplification during $\phi 1$ high, latch operates when $\phi 1$ low
- Input buffers suppress kick-back & input current
- Separate ground and supply buses for front-end preamp → kick-back noise reduction



Ref: Y. Akazawa, et al., "A 400MSPS 8b flash AD conversion LSI," *IEEE International Solid-State Circuits Conference*, vol. XXX, pp. 98 - 99, February 1987

Ref: T. Wakimoto, et al, "Si bipolar 2GS/s 6b flash A/D conversion LSI," *IEEE International Solid-State Circuits Conference*, vol. XXXI, pp. 232 - 233, February 1988

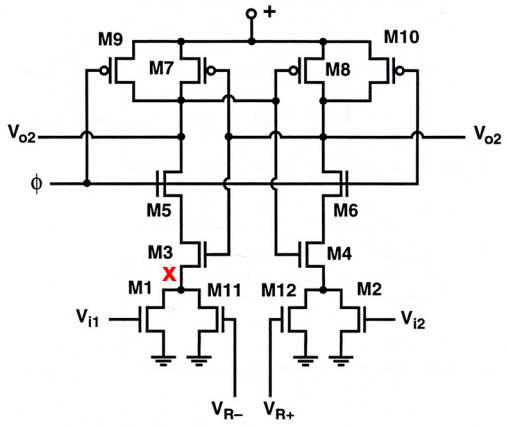
Auto-Zero Implementation



Ref:I. Mehr and L. Singer, "A 55-mW, 10-bit, 40-Msample/s Nyquist-Rate CMOS ADC," JSSC March 2000, pp. 318-25

Comparator Example
Pipelined ADC Application

- 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}



Ref: T. B. Cho and P. R. Gray, "A 10 b, 20 Msample/s, 35 mW pipeline A/D converter," *IEEE Journal of Solid-State Circuits,* vol. 30, pp. 166 - 172, March 1995

Comparator Example (continued)

- M1, M2, M11, M12 operate in triode mode with all having equal L
- Conductance of input devices:

$$G_1 = \frac{\mu C_{ox}}{L} \times \left[W_1 (V_{I1} - V_{th}) + W_{11} (V_{R-} - V_{th}) \right]$$

$$G_2 = \frac{\mu C_{ox}}{L} \times \left[W_1 (V_{I2} - V_{th}) + W_{11} (V_{R+} - V_{th}) \right]$$

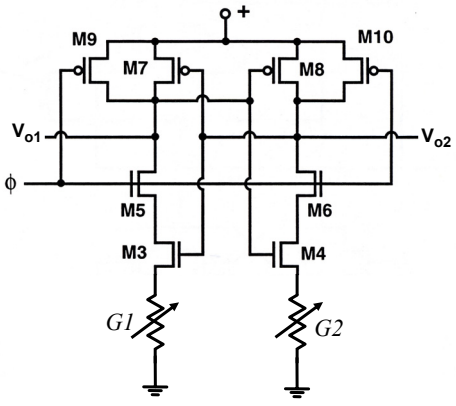
$$\rightarrow \Delta G = \frac{\mu C_{ox} W_1}{L} \times \left[(V_{I1} - V_{I2}) - \frac{W_{11}}{W_1} (V_{R+} - V_{R-}) \right]$$

- To 1st order, for $W1 = W2$ & $W11 = W12$

  $V_{th}^{latch} = W11/W1 \times V_R$

  where $V_R = V_{R+} - V_{R-}$

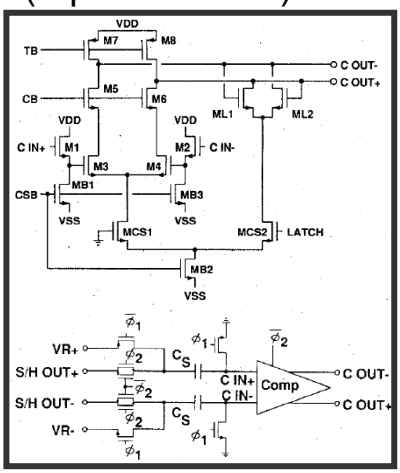
  → V_R fixed $W11, 12$ varied from comparator to comparator → Eliminates need for resistive divider (DAC)



Ref: T. B. Cho and P. R. Gray, "A 10 b, 20 Msample/s, 35 mW pipeline A/D converter," *IEEE Journal of Solid-State Circuits,* vol. 30, pp. 166 - 172, March 1995

Comparator Example (Pipelined ADC)

- Used in a pipelined ADC with digital correction
  →No offset cancellation required
- Differential reference & input
- M7, M8 operate in triode region
- Preamp gain ~10
- Input buffers suppress kick-back
- ϕ_1 high → C_s charged to VR & ϕ_{2B} is also high → current diverted to latch→ comparator output in hold mode
- ϕ_2 high→ C_s connected to S/Hout & comparator input (VR-S/Hout), current sent to preamp → comparator in amplify mode



Ref: S. Lewis, et al., "A Pipelined 5-Msample/s 9-bit Analog-to-Digital Converter" IEEE JSSC , NO. 6, Dec. 1987

Reducing Flash ADC Complexity

E.g. 10-bit "straight" flash

- Input range: 0 … 1V
- LSB = Δ: ~ 1mV
- Comparators: 1023 with offset < 1/2 LSB
- Input capacitance: 1023 * 100fF = 102pF
- Power: 1023 * 3mW = 3W

→ High power dissipation & large area & high input cap.

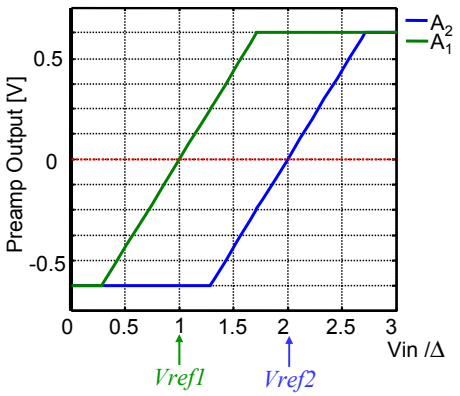
Techniques to reduce complexity & power dissipation :

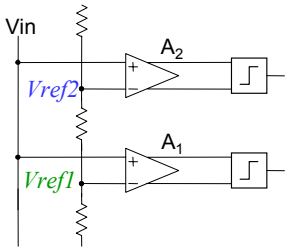
- Interpolation
- Folding
- Folding & Interpolation
- Two-step, pipelining

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



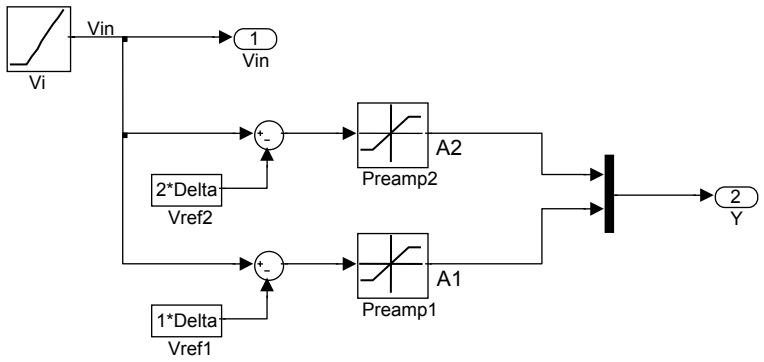


Zero crossings (to be detected by latches) at V_{in} =

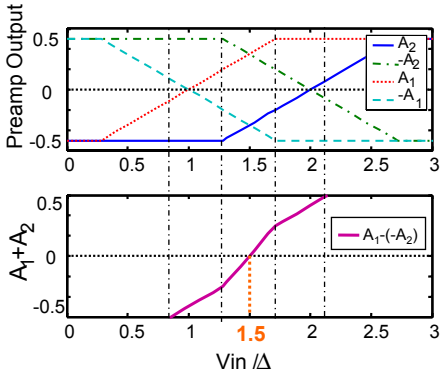
$V_{ref1} = 1\ \Delta$

$V_{ref2} = 2\ \Delta$

Simulink Model



Differential Preamp Output



Differential output crossings @ V_{in} =

V_{ref1} = 1 Δ

V_{ref2} = 2 Δ

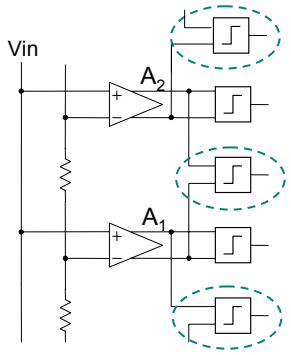
Note: Additional crossing of A_1&-A_2 (A_2&-A_1)

→ A_1-(-A_2)=A_1+A_2

→ cross zero at:

V_{ref12} = 0.5*(1+2) Δ=1.5Δ

Interpolation in Flash ADC



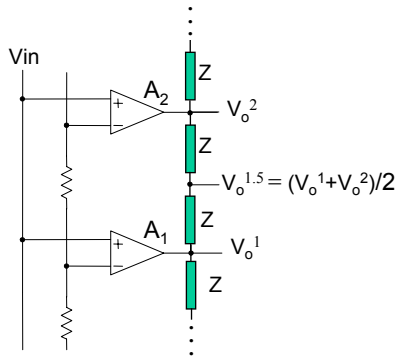
Compare A2& -A1
→ Comparator output is sign of A1+A2

Half as many reference voltages and preamps
Interpolation factor:x2

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

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

Interpolation in Flash ADC
Preamp Output Interpolation

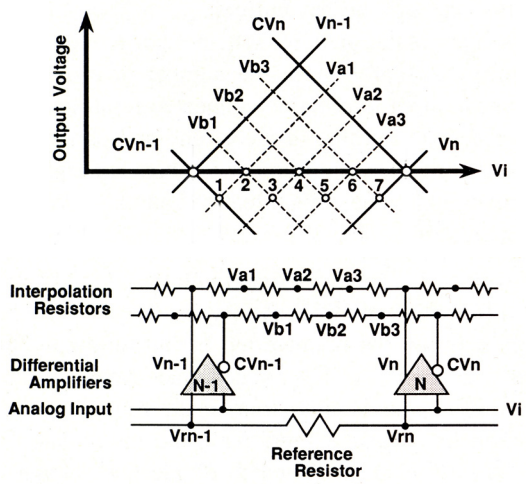


Interpolate between two consecutive output via impedance Z

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

Ref: H. Kimura et al, "A 10-b 300-MHz Interpolated-Parallel A/D Converter," JSSC, pp. 438-446, April 1993
K. Kusumoto et al, "A 10-b 20-MHz 30-mW pipelined interpolating CMOS ADC," *JSSC,* pp.1200 -1206, December 1993.
R. Roovers et al, "A 175 Ms/s, 6 b, 160 mW, 3.3 V CMOS A/D converter," *JSSC*, pp. 938 - 944, July 1996.

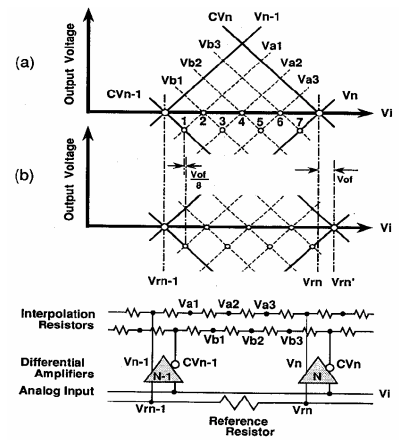
Higher Order Resistive Interpolation



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

Ref: H. Kimura et al, "A 10-b 300-MHz Interpolated-Parallel A/D Converter," JSSC April 1993, pp. 438-446

Preamp Output Interpolation DNL Improvement

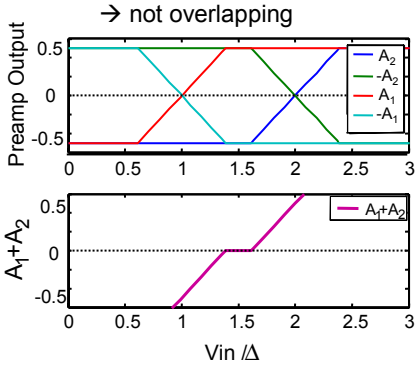


- Preamp offset distributed over M resistively interpolated voltages:
  → Impact on DNL divided by M
- Latch offset divided by gain of preamp
  → Use "large" preamp gain
  → Next: Investigate how large preamp gain can be?

Ref: H. Kimura et al, "A 10-b 300-MHz Interpolated-Parallel A/D Converter," JSSC April 1993, pp. 438-446

Preamp Input Range

Linear region of transfer curve
→ not overlapping



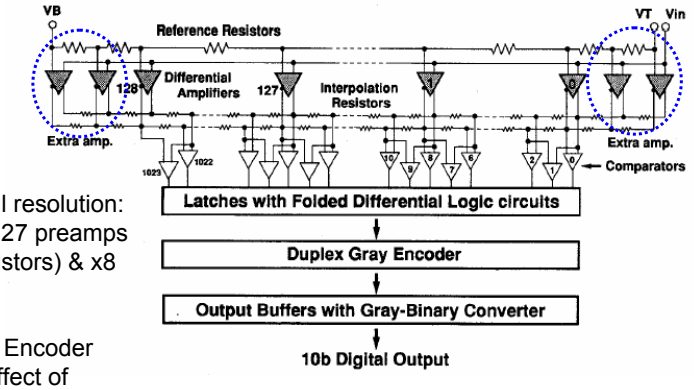
If linear region of preamp transfer curve do not overlap

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

→ Consecutive preamp linear input ranges must overlap
i.e. input range > Δ

Sets upper bound on preamp gain: $Preamp_{gain} < V_{DD} / \Delta$

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 & meta-stability

Ref: H. Kimura et al, "A 10-b 300-MHz Interpolated-Parallel A/D Converter," JSSC April 1993, pp. 438-446

Measured Performance

Resolution		**10 b (7+3)**
Maximum conversion frequency		**300 MHz**
Integral non-linearity		**±1.0 LSB**
Differential non-linearity		**±0.4 LSB**
SNR/THD	**10MHz input**	**56/-59 dB**
	50MHz input	**48/-47 dB**
Input capacitance		**8 pF**
Input range		**2 V**
Power supply		**-5.2V**
Power dissipation		**4.0W**
Chip size		**9.0 × 4.2 mm²**
Element count		**36,000**
Technology		**1.0 μm bipolar:ft=25GHz**

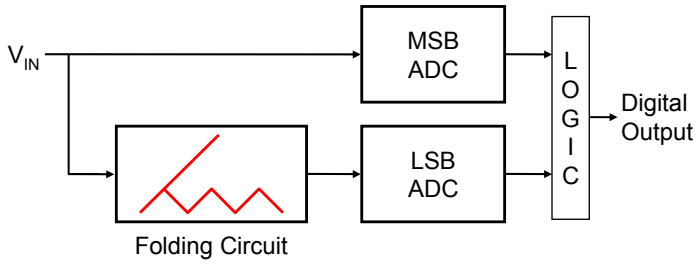
Low input capacitance

Ref: H. Kimura et al, "A 10-b 300-MHz Interpolated-Parallel A/D Converter," JSSC April 1993, pp. 438-446

Interpolation Summary

- Consecutive preamp transfer curve need to have overlap → Limits gain of preamp to ~V_{DD}/Δ
- 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

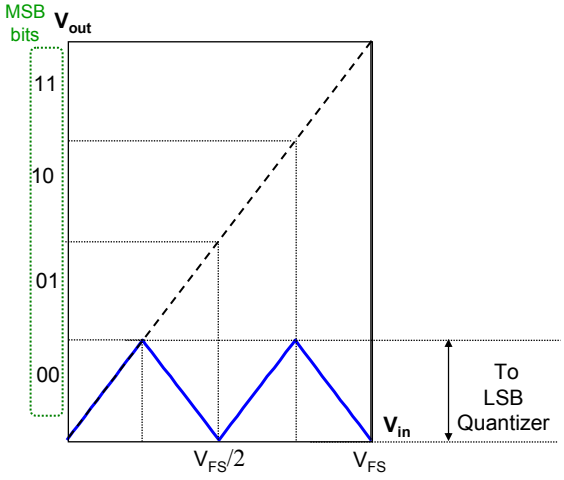
Folding Converter



- Two ADCs operating in parallel
  - MSB ADC
  - Folder + LSB ADC
- Significantly fewer comparators compared to flash
- Fast
- Typically, nonidealities in folder limit resolution to ~10Bits

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

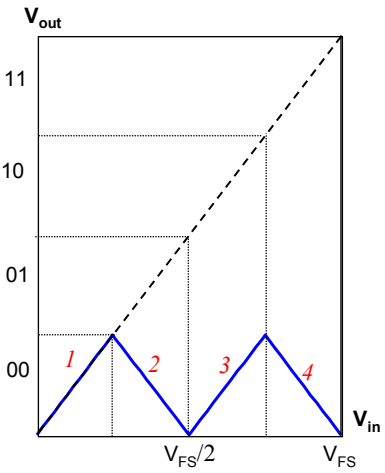


Example: Folding Factor of 4

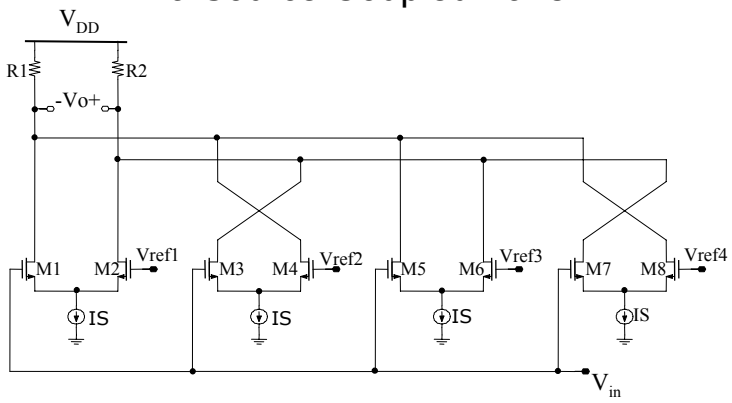
- How are folds generated?

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

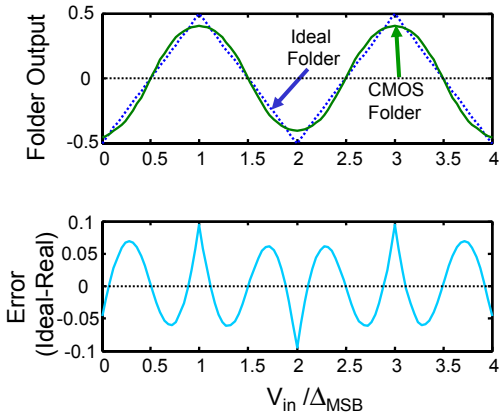


Generating Folds via Source-Coupled Pairs



Vref1 < Vref2 < Vref3 < Vref4
As Vin changes, only one of M1, M3, M5, M7 is on depending on the input level
Let us try Vref1=$1/2\Delta_{MSB}$ Vref2=$1.5\Delta_{MSB}$ Vref3=$2.5\Delta_{MSB}$ Vref4=$3.5\Delta_{MSB}$

CMOS Folder Output



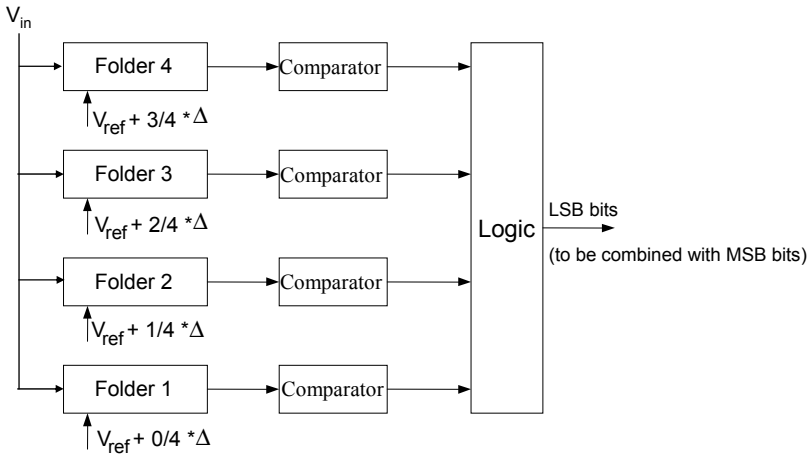
CMOS folder transfer curve max. min. portions:

→ Rounded

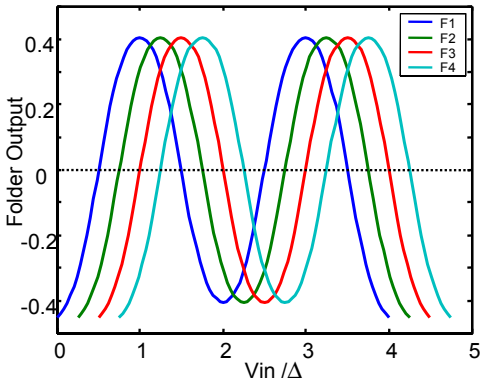
→ Accurate only at zero-crossings

In fact, most folding ADCs do not use the folds, but only the midpoint-crossings!

Parallel Folders Using Only Midpoint-Crossings



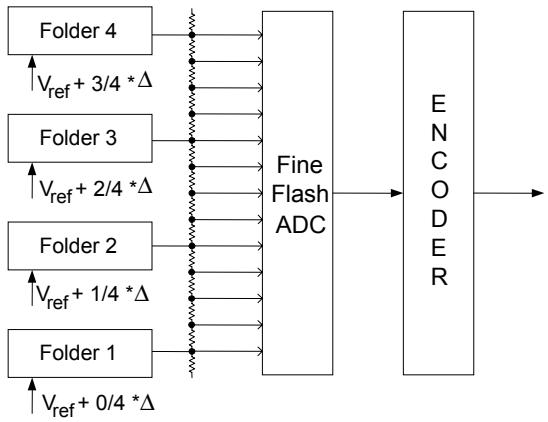
Parallel Folder Outputs



- 4 folders with 4 folds each
- 16 zero crossings
- → 4 LSB bits

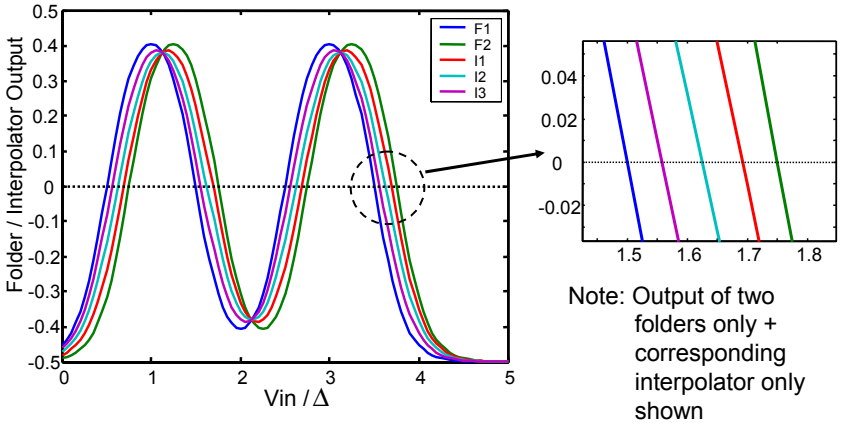
- Higher resolution
  - More folders
    → Large complexity
  - Interpolation

Folding & Interpolation



Folder / Interpolator Output

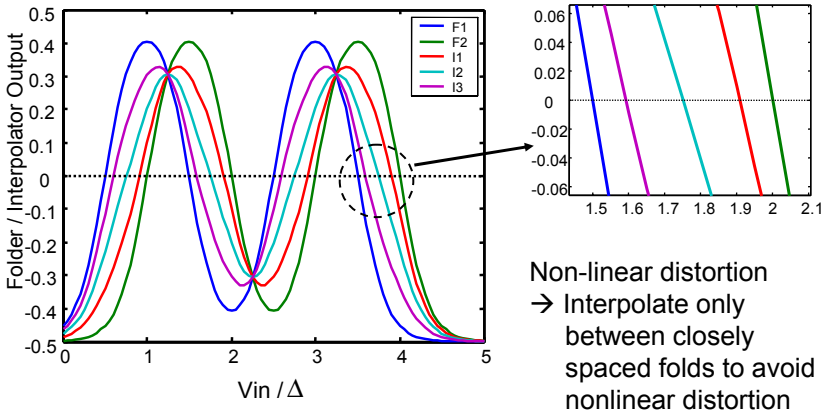
Example:4 Folders + 4 Resistive Interpolator per Stage



Note: Output of two folders only + corresponding interpolator only shown

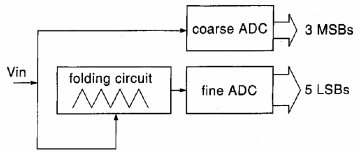
Folder / Interpolator Output

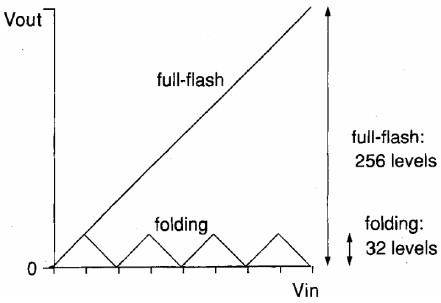
Example:2 Folders + 8 Resistive Interpolator per Stage



Non-linear distortion
→ Interpolate only between closely spaced folds to avoid nonlinear distortion

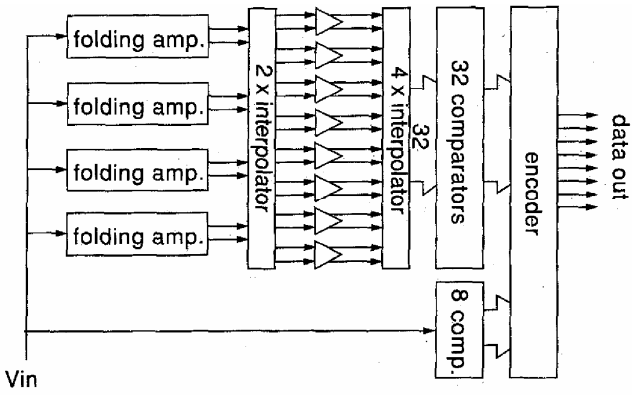
A 70-MS/s 110-mW 8-b CMOS Folding and Interpolating A/D Converter





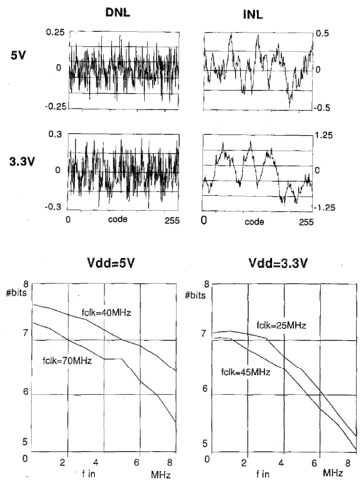
Ref: B. Nauta and G. Venes, JSSC Dec 1985, pp. 1302-8

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

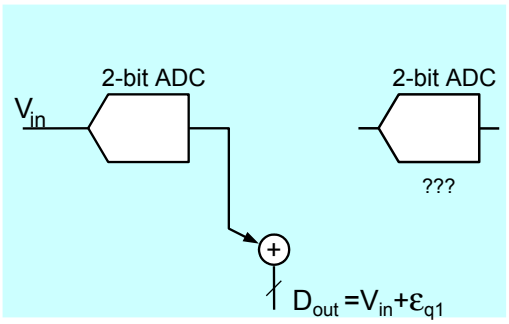
A 70-MS/s 110-mW 8-b CMOS Folding and Interpolating A/D Converter

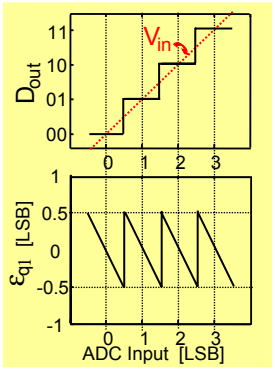


parameter		
resolution	8bit	
input capacitance	4.8 pF ←	
reference ladder resistance	720 Ω	
active area	0.7 mm²	
technology	0.8μm, 1 poly , 2 metal, CMOS	
supply voltage	V_{dd}=5V	V_{dd}=3.3V
analog input	2Vpp	1.4Vpp
Integral nonlinearity	$\pm 0.5LSB$	$\pm 1.0LSB$ ←
Differential nonlinearity	$\pm 0.2LSB$	$\pm 0.3LSB$ ←
max. clock frequency	70MHz	45MHz
power dissipation	110mW	45mW

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

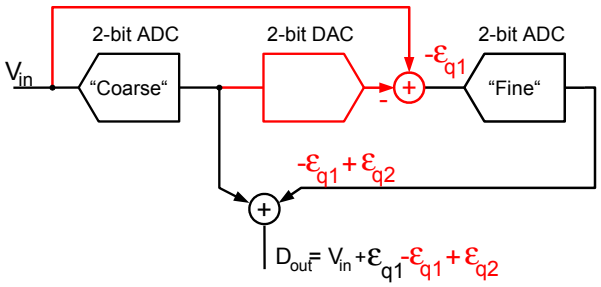
Two-Step Example: (2+2)Bits





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

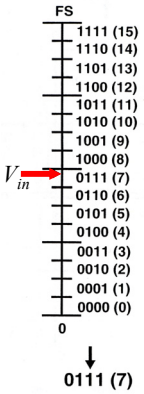
Two Stage Example



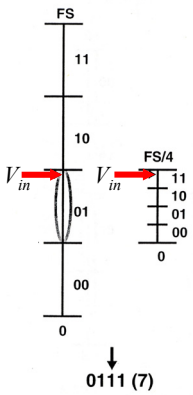
- Use DAC to compute missing voltage
- Add quantized representation of missing voltage
- Why does this help? How about ε_{q2} ?

Two Step (2+2) Flash ADC

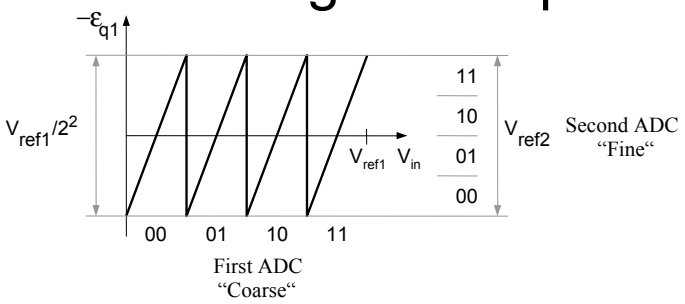
4-bit Straight Flash ADC



Ideal 2-step Flash ADC



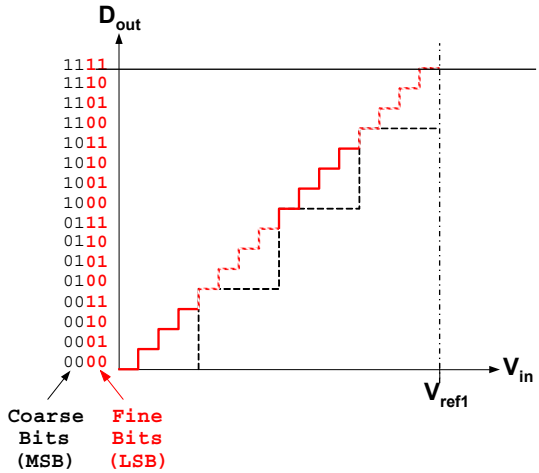
Two Stage Example



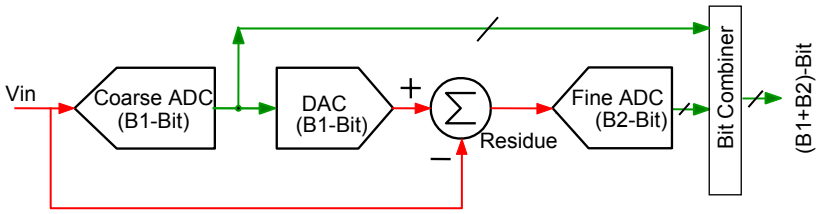
- Fine ADC is re-used 2^2 times
- Fine ADC's full scale range needs to span only 1 LSB of coarse quantizer

$$\varepsilon_{q2} = \frac{V_{ref2}}{2^2} = \frac{V_{ref1}}{2^2 \cdot 2^2}$$

Two-Stage (2+2) ADC Transfer Function

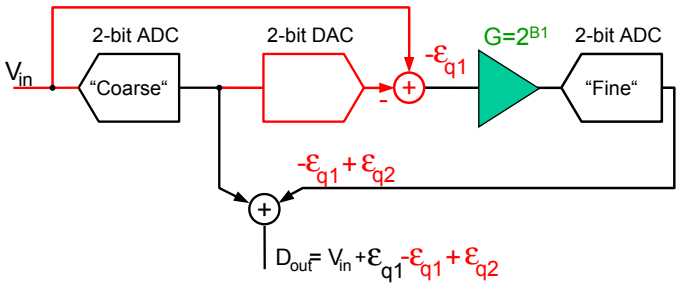


Residue or Multi-Step Type ADC Issues



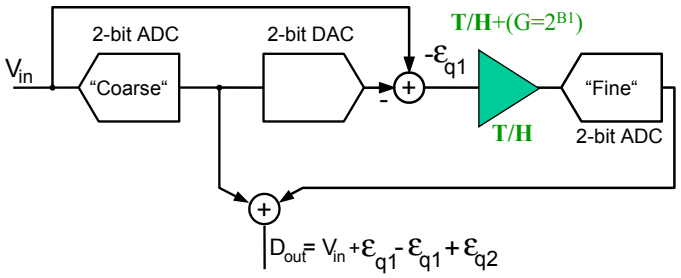
- Operation:
  - Coarse ADC determines MSBs
  - DAC converts the coarse ADC output to analog- Residue is found by subtracting (V_{in}-V_{DAC})
  - Fine ADC converts the residue and determines the LSBs
  - Bits are combined in digital domain
- Issue:
  1. Fine ADC has to have precision in the order of 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) Reducing Precision Required for Fine ADC



- Accuracy needed for fine ADC relaxed by introducing inter-stage gain
  - Example: By adding gain of x($G=2^{B1}=4$) 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)
Increasing ADC Throughput

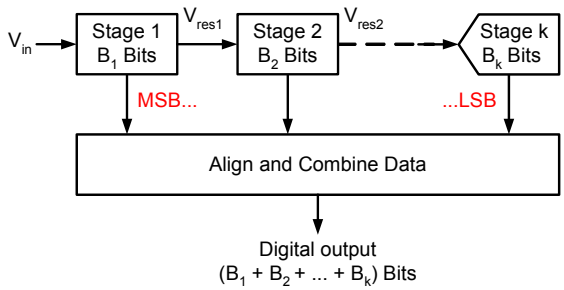


- Conversion time significantly decreased by employing T/H between stages
  - All stages busy at all times → operation concurrent
  - During one clock cycle coarse & fine ADCs operate concurrently:
    - First stage samples/converts/generates residue of input signal sample # *n*
    - While 2nd stage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

Pipelined A/D Converters

- Ideal operation
- Errors and correction
  - Redundancy
  - Digital calibration
- Implementation
  - Practical circuits
  - Stage scaling

Pipeline ADC Block Diagram

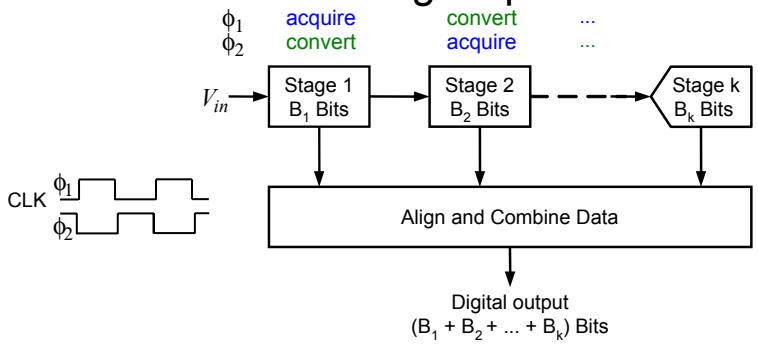


- Idea: Cascade several low resolution stages to obtain high overall resolution (e.g. 10bit ADC can be built with series of 10 ADCs each 1-bit only!)
- Each stage performs coarse A/D conversion and computes its quantization error, or "residue"
- All stages operate concurrently

Pipeline ADC Characteristics

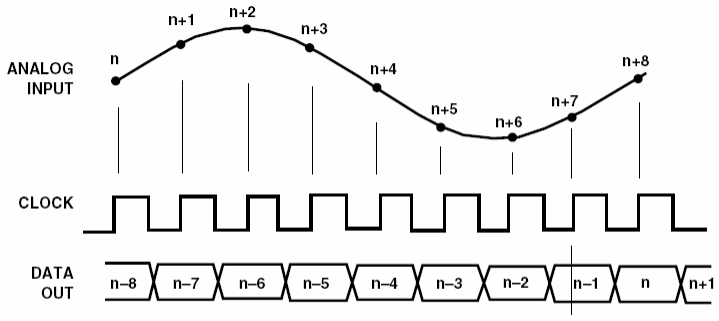
- Number of components (stages) grows linearly with resolution
- Pipelining
  - Trading latency for conversion speed
  - Latency may be an issue in e.g. control systems
  - Throughput limited by speed of one stage → Fast
- Versatile: 8...16bits, 1...200MS/s
- One important feature of pipeline ADC: many 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

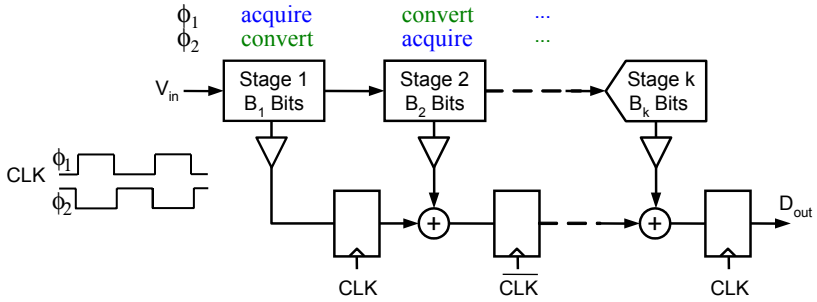
Pipeline ADC
Latency



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

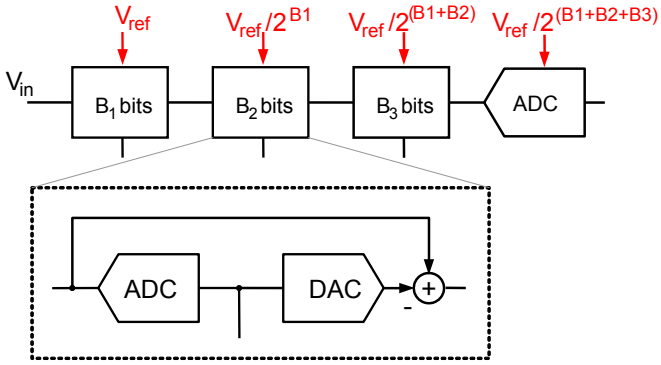
[Analog Devices, AD 9226 Data Sheet]

Pipeline ADC
Digital Data Alignment



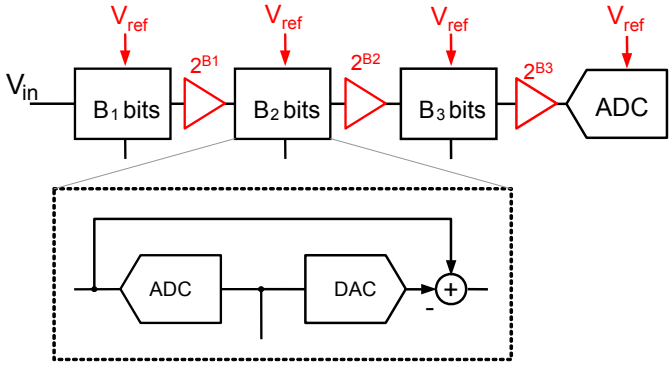
- Digital shift register aligns sub-conversion results in time

Cascading More Stages



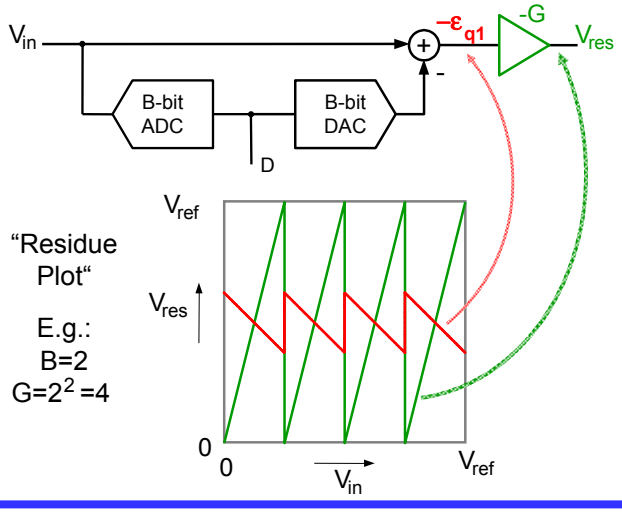
- LSB of last stage becomes very small
- Impractical to generate several V_{ref}
- All stages need to have full precision

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), power dissipation

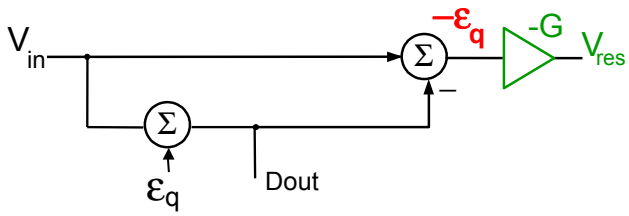
Complete Pipeline Stage



"Residue Plot"

E.g.:
B=2
$G=2^2=4$

Pipeline ADC Single Stage Model

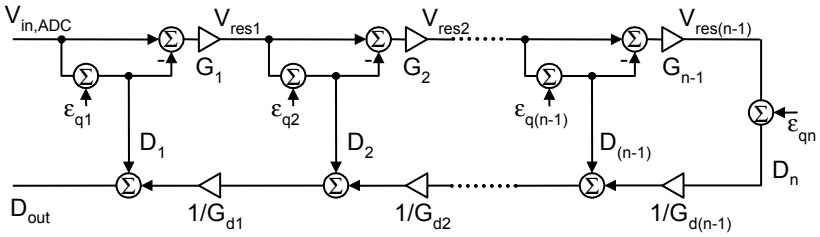


$$V_{res} = G x \varepsilon_q$$

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 Multi-Stage Model



$$D_{out} = V_{in,ADC} + \varepsilon_{q1}\left(1 - \frac{G_1}{G_{d1}}\right) + \frac{\varepsilon_{q2}}{G_{d1}}\left(1 - \frac{G_2}{G_{d2}}\right) + \ldots + \frac{\varepsilon_{q(n-1)}}{\prod_{j=1}^{n-2} G_{dj}}\left(1 - \frac{G_{(n-1)}}{G_{d(n-1)}}\right) + \frac{\varepsilon_{qn}}{\prod_{j=1}^{n-1} G_{dj}}$$

Pipeline ADC Model

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

$$D_{out} = V_{in,ADC} + \frac{\varepsilon_{qn}}{\prod_{j=1}^{n-1} G_j}$$

$$D.R. = 20\log\frac{rms\ FS\ Signal}{rms\ Quant.\ Noise} = 20\log\frac{\frac{V_{ref}}{2\sqrt{2}}}{\frac{V_{ref}}{\sqrt{12}\times 2^{B_n}\prod_{j=1}^{n-1} G_j}} = 20\log\left(\sqrt{\frac{3}{2}}\times 2^{B_n}\times\prod_{j=1}^{n-1} G_j\right)$$

$$B_{ADC} \approx \log_2\left(2^{B_n}\times\prod_{j=1}^{n-1} G_j\right)$$

$$B_{ADC} \approx B_n + \log_2\prod_{j=1}^{n-1} G_j$$

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_{out} contains quantization error associated with the last stage
- So why do we care about sub-ADC errors?
  - ➢ Go back to two stage example