Pipelined Analog-to-Digital Converters

Vishal Saxena, Boise State University
(vishalsaxena@boisestate.edu)

Multi-Step A/D Conversion Basics

Motivation for Multi-Step Converters

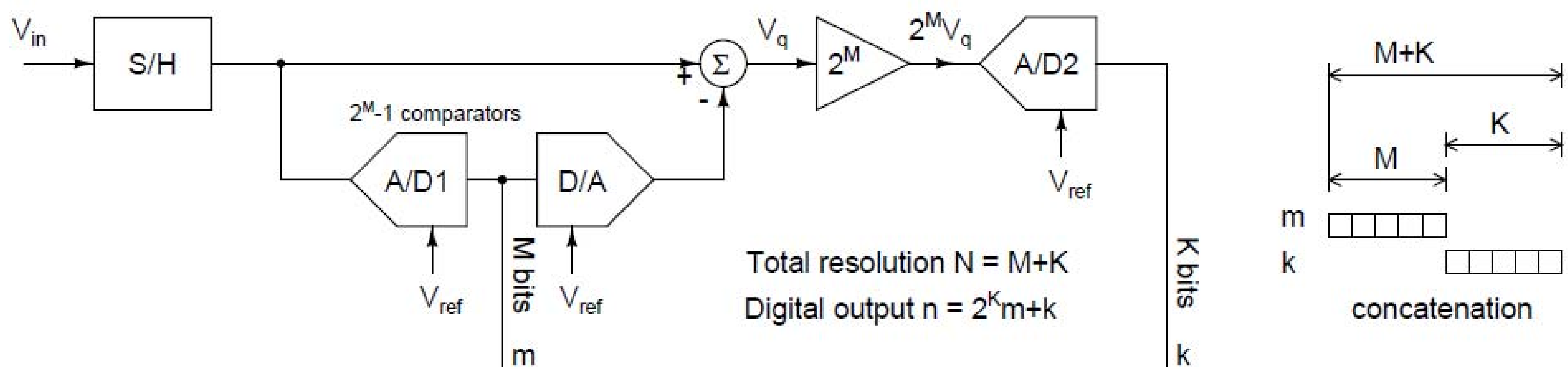
- Flash A/D Converters
  - Area and power consumption increase exponentially with
  - number of bits N
  - Impractical beyond 7-8 bits

- Multi-step conversion-Coarse conversion followed by fine conversion
  - Multi-step converters
  - Sub-ranging converters

- Multi step conversion takes more time
  - Pipelining to increase sampling rate

- Objective: Understand digital redundancy concept in multi-step converters

Two-step A/D Converter - Basic Operation

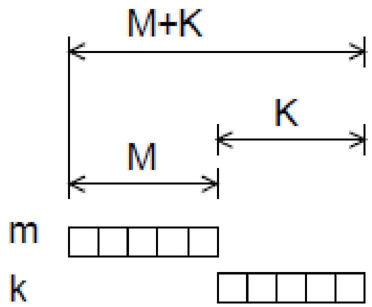


- Second A/D quantizes the quantization error of first A/D converter
- Concatenate the bits from the two A/D converters to form the final output
- Also called as two-step Flash ADC

Two-step A/D Converter - Basic Operation

- A/D1, DAC, and A/D2 have the same range V_{ref}
- Second A/D quantizes the quantization error of first A/D
  - Use a DAC and subtractor to determine residue V_q
  - Amplify V_q to full range of the second A/D

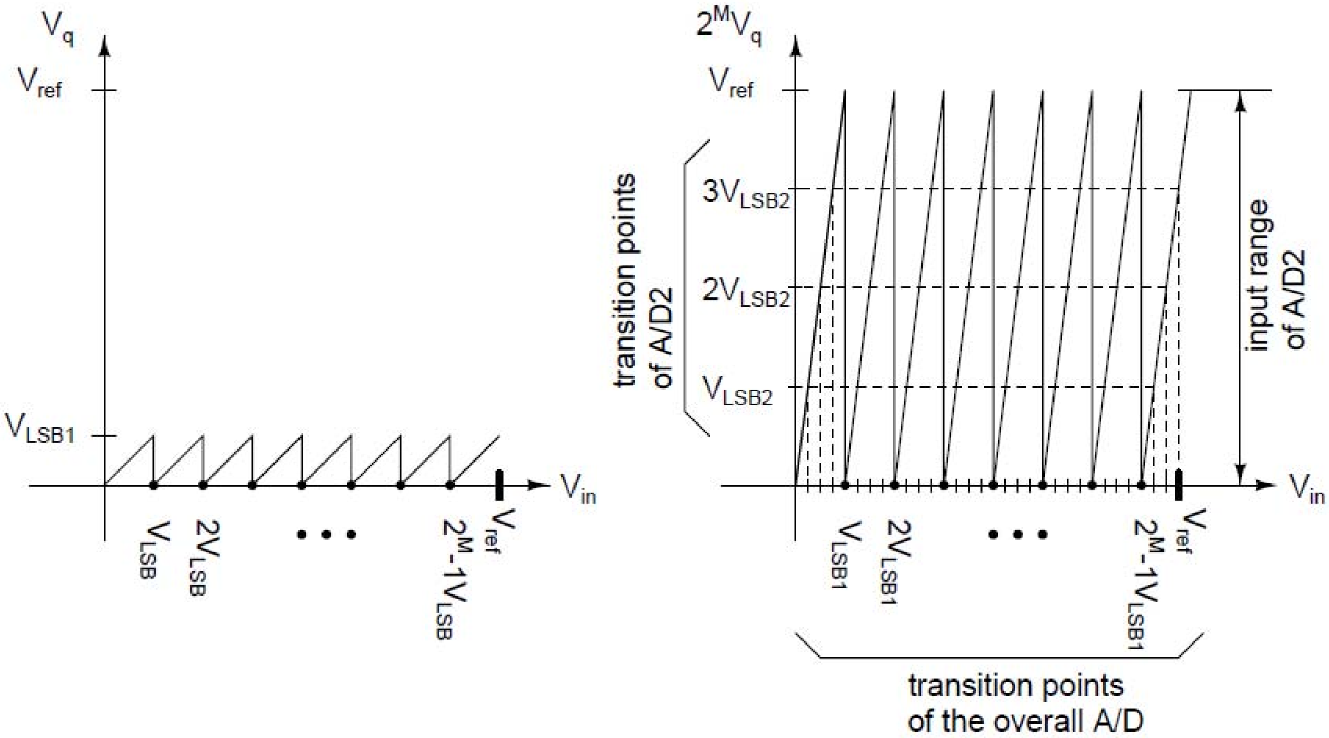
- Final output *n* from *m, k*
  - A/D1 output is *m* (DAC output is $m/2^M V_{ref}$)
  - A/D2 input is at k^{th} transition ($k/2^K V_{ref}$)
  - $V_{in} = k/2^K V_{ref} \times 1/2^M + m/2^M V_{ref}$
  - $V_{in} = (2^K m + k)/2^{M+K} V_{ref}$



concatenation

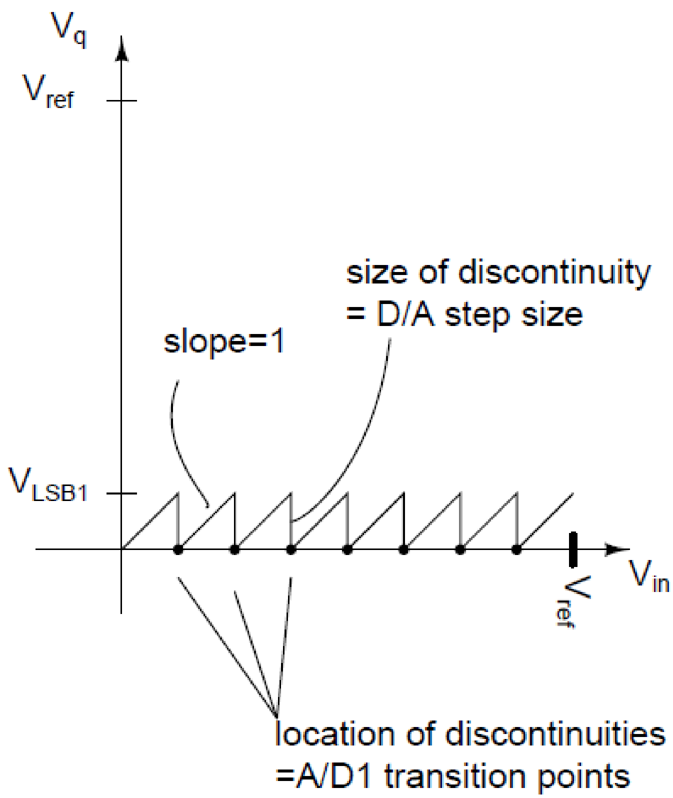
- Resolution $N = M + K$
  - output $\Rightarrow n = 2^K m + k$
  - Concatenate the bits from the two A/D converters to form the final output

Two-step A/D Converter – Example with M=3, K=2



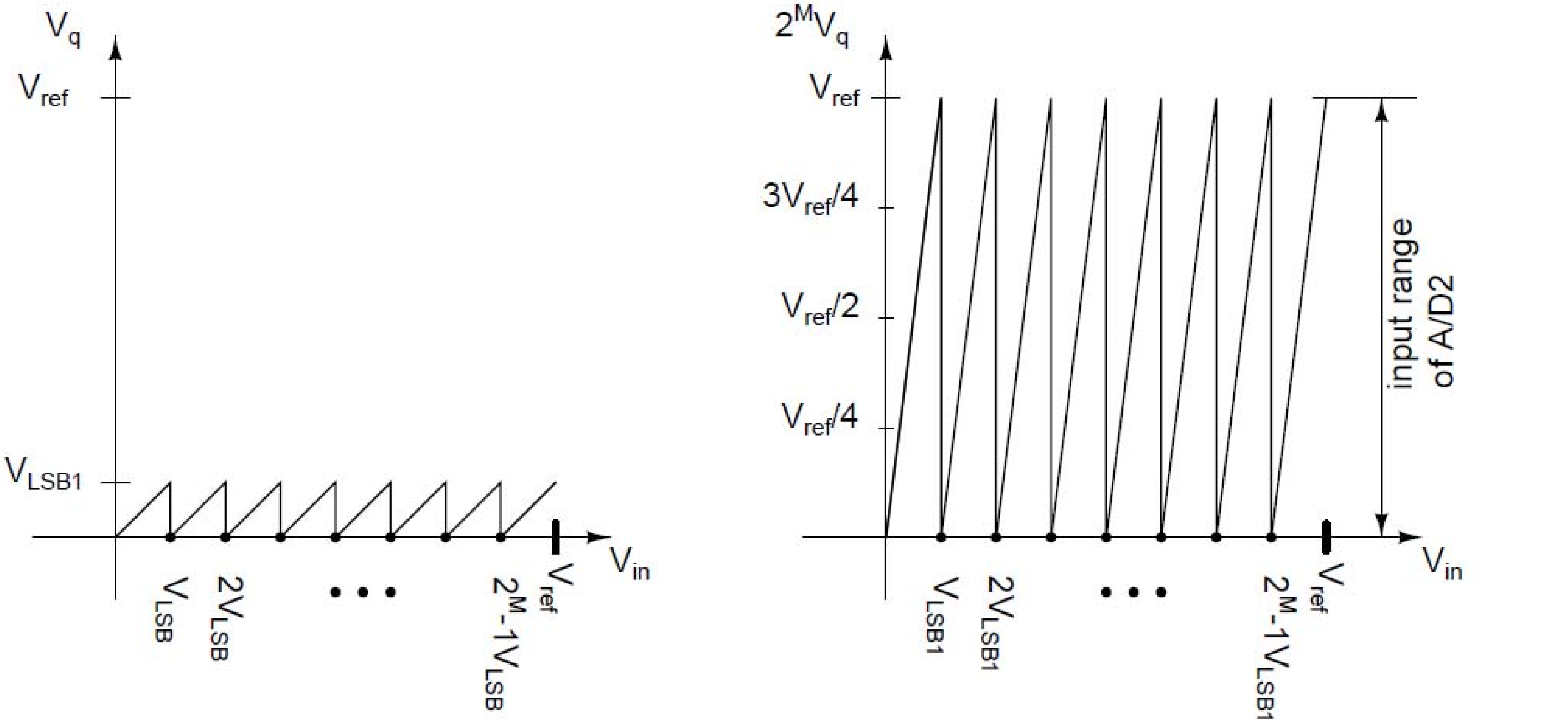
- Second A/D quantizes the quantization error of first A/D
- Transitions of second A/D lie between transitions of the first, creating finely spaced transition points for the overall A/D

Residue V_q



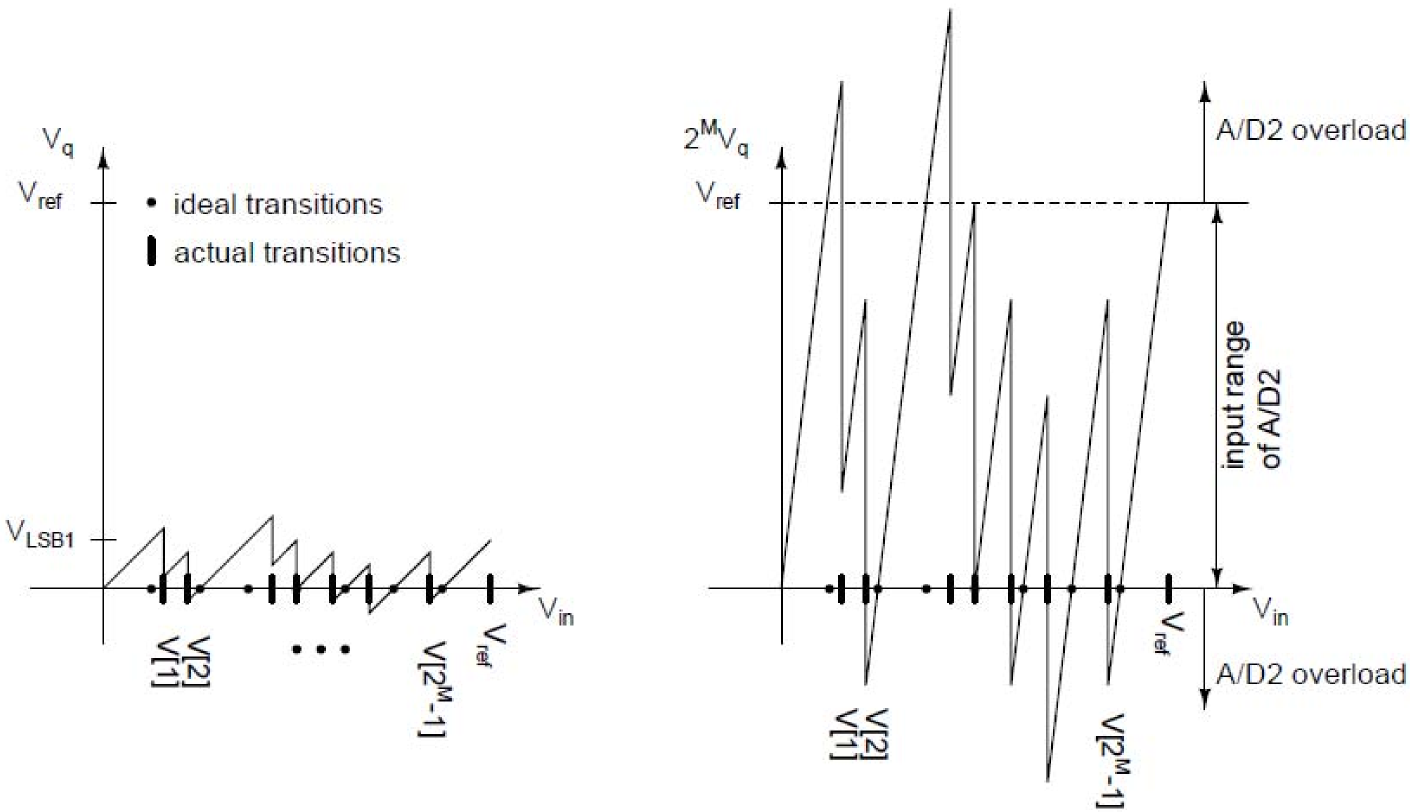
- V_q vs. V_{in}: Discontinuous transfer curve
  - Location of discontinuities: Transition points of A/D1
  - Size of discontinuities: Step size of D/A
  - Slope: unity

Two-step A/D Converter—Ideal A/D1



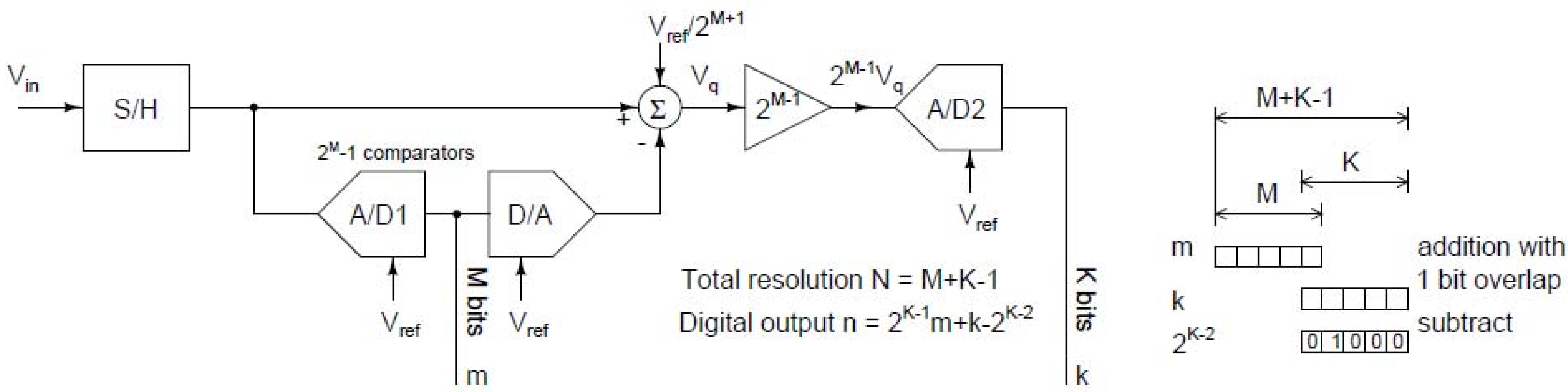
- A/D1 transitions exactly at integer multiples of $V_{ref}/2^M$
- Quantization error V_q limited to $(0, V_{ref}/2^M)$
- $2^M V_q$ exactly fits the range of A/D2

Two-step A/D converter—M bit accurate A/D1



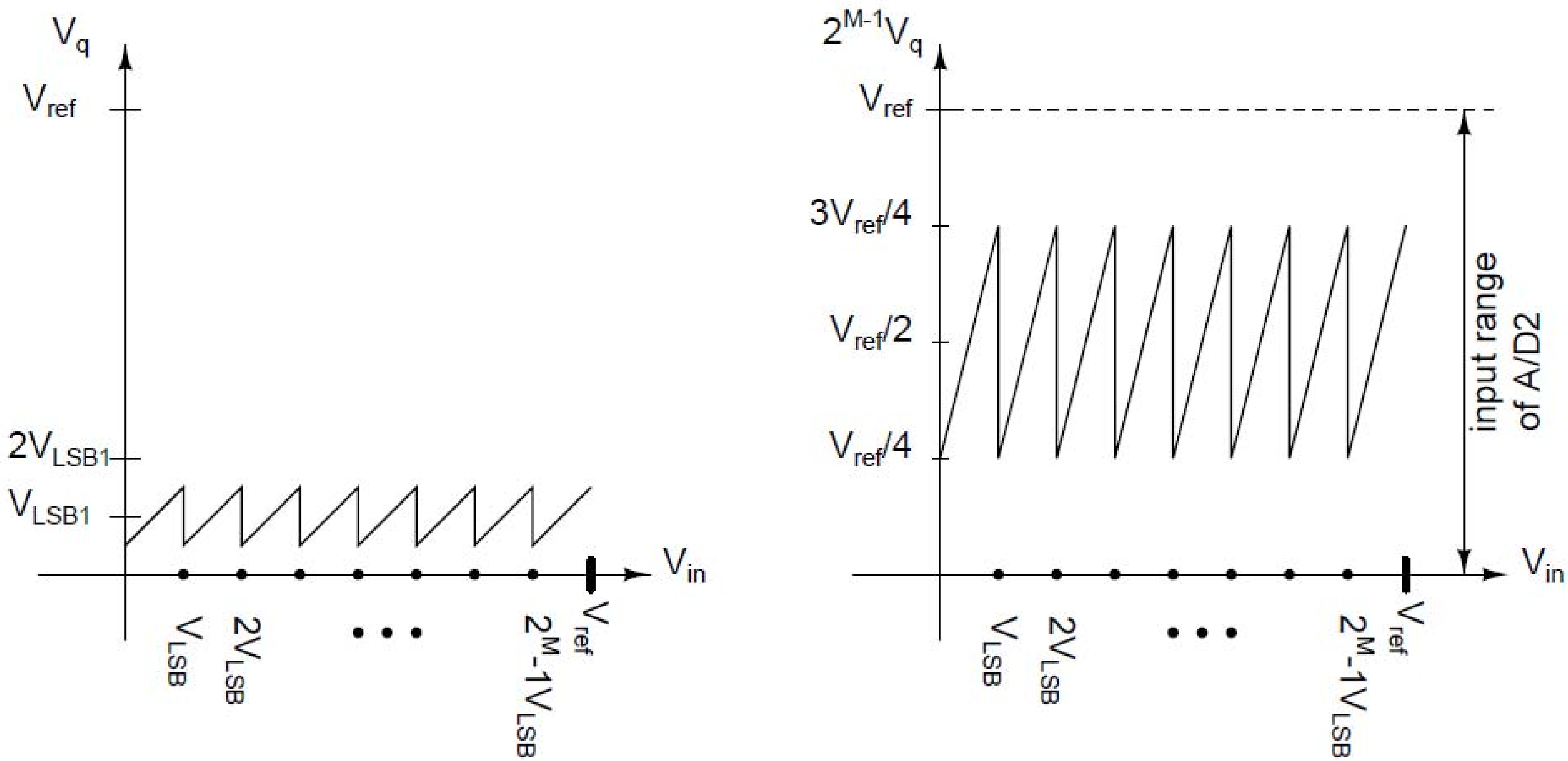
- A/D1 transitions in error by up to $V_{ref}/2^{M+1}$ (= 0.5 LSB)
- Quantization error V_q limited to
  - $(-V_{ref}/2^{M+1}, 3V_{ref}/2^{M+1})$—a range of $V_{ref}/2^{M-1}$
- $2^M V_q$ overloads A/D2

Two-step A/D with Digital Redundancy (DR)



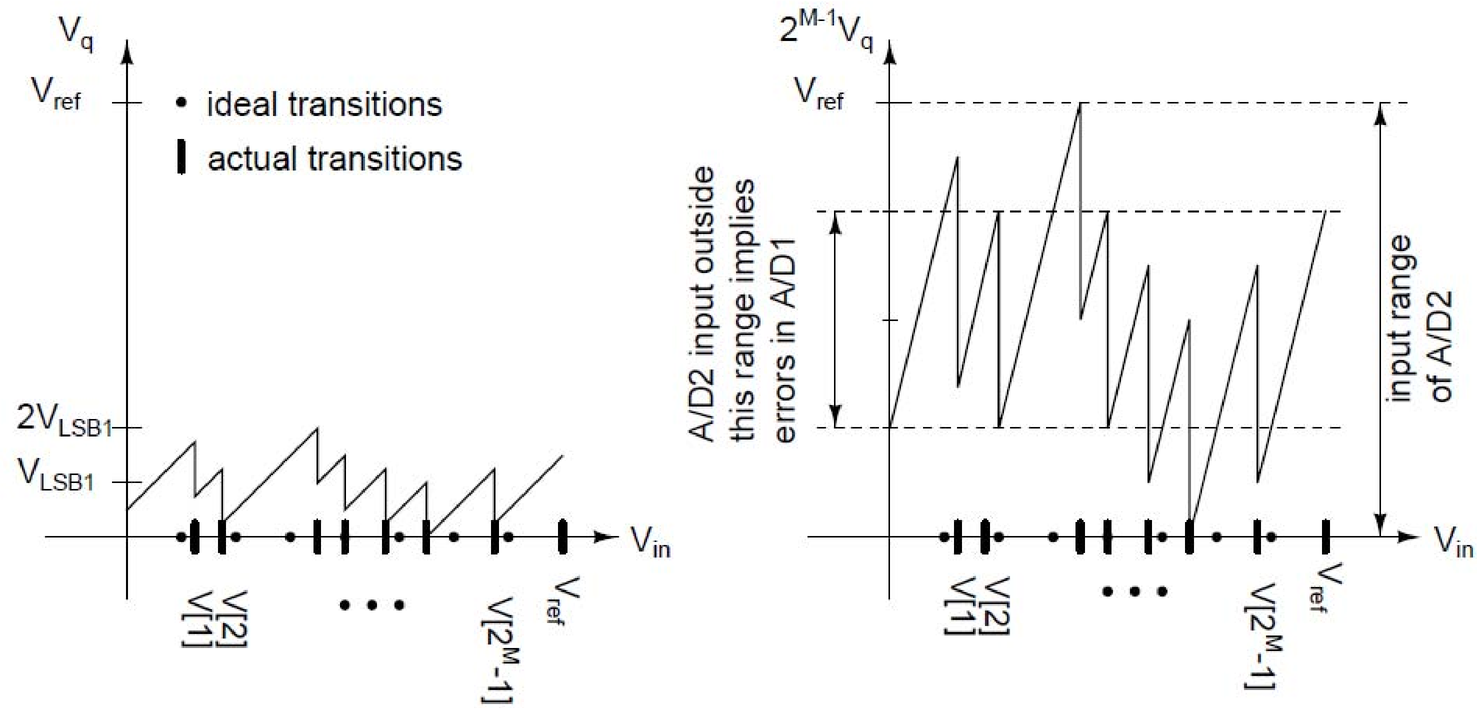
- Reduce interstage gain to 2^{M-1}
- Add $V_{ref}/2^{M+1}$ (0.5 LSB_1) offset to keep V_q positive
- Subtract 2^{K-2} from digital output to compensate for the added offset
  - Digital code in A/D2 corresponding to 0.5 LSB_1 = $(V_{ref}/2^{M+1})/(V_{ref}/2^{K+1}) = 2^{K-2}$
- Overall accuracy is $N = M + K - 1$ bits
  - A/D1 contributes M – 1 bits
  - A/D2 contributes K bits; 1 bit redundancy
- Output $n = 2^{K-1}m + k - 2^{K-2}$

Two-step A/D with DR: Ideal A/D1 Scenario



- $2^{M-1}V_q$ varies from $V_{ref}/4$ to $3V_{ref}/4$
- $2^{M-1}V_q$ outside this range implies errors in A/D1

Two-step A/D with DR: M-bit accurate A/D1



- $2^{M-1}V_q$ varies from 0 to V_{ref}
- A/D2 is not overloaded for up to 0.5 LSB errors in A/D1
- Issue: Accurate analog addition of 0.5 LSB_1 is difficult

Two-step A/D with DR: M-bit accurate A/D1

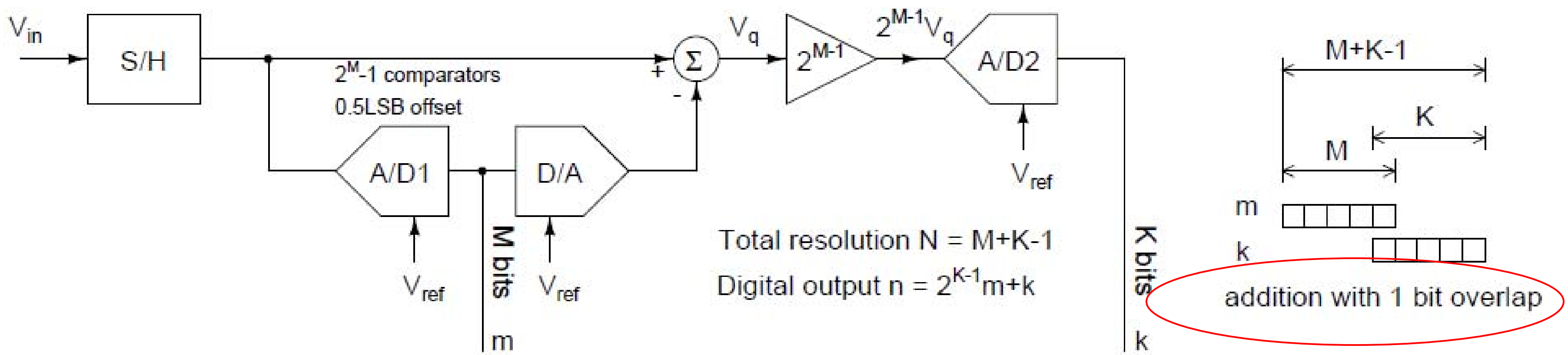
- Recall that output $n = 2^{K-1}m + k - 2^{K-2}$

- A/D1 Transition shifted to the left
  - m greater than its ideal value by 1
  - k lesser than its ideal value by 2^{K-1}
  - A/D output $n = 2^{K-1}m + k - 2^{K-2}$ doesn't change

- A/D1 Transition shifted to the right
  - m lesser than its ideal value by 1
  - k greater than its ideal value by 2^{K-1}
  - A/D output $n = 2^{K-1}m + k - 2^{K-2}$ doesn't change

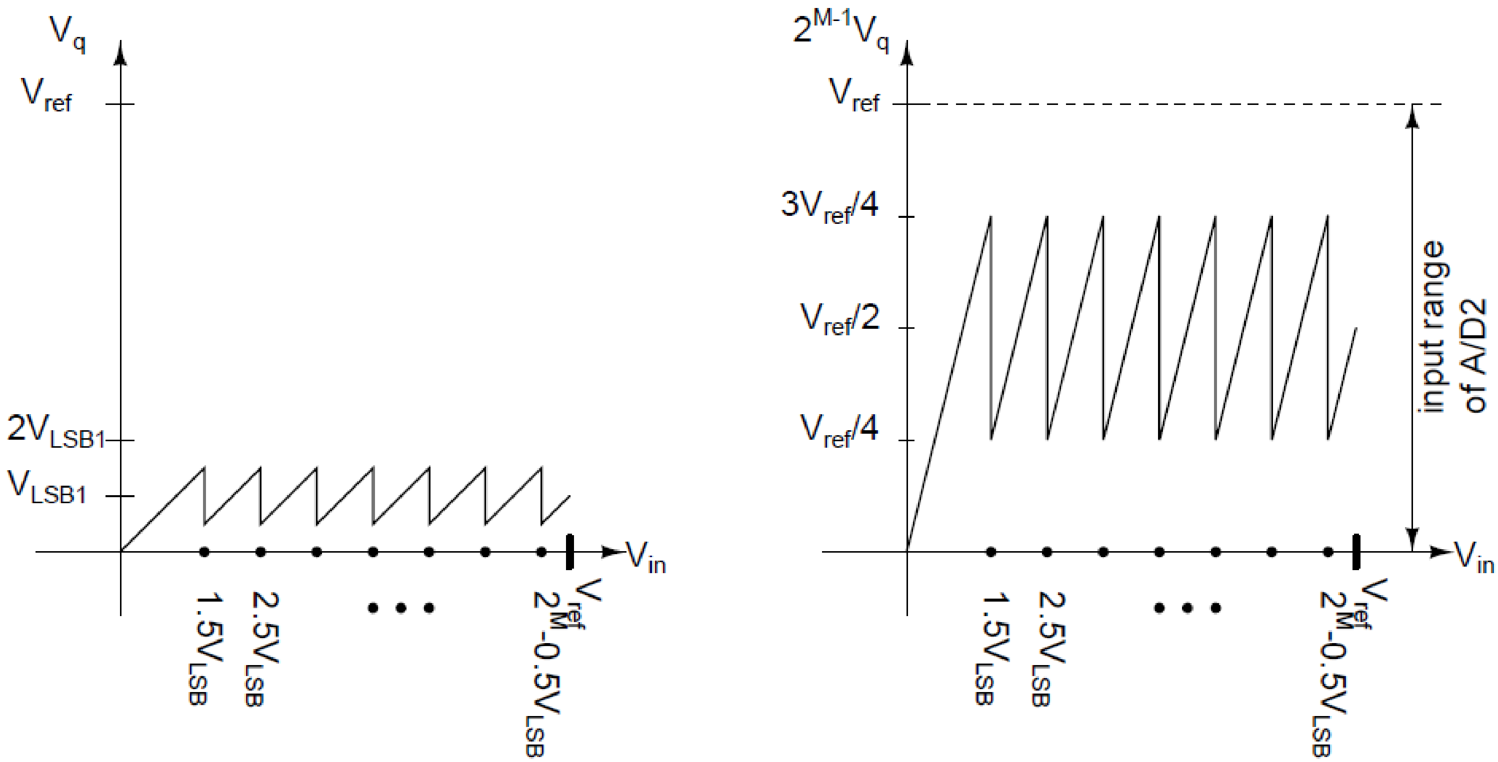
- 1 LSB error in m can be corrected

Two-step A/D with Digital Redundancy (II)



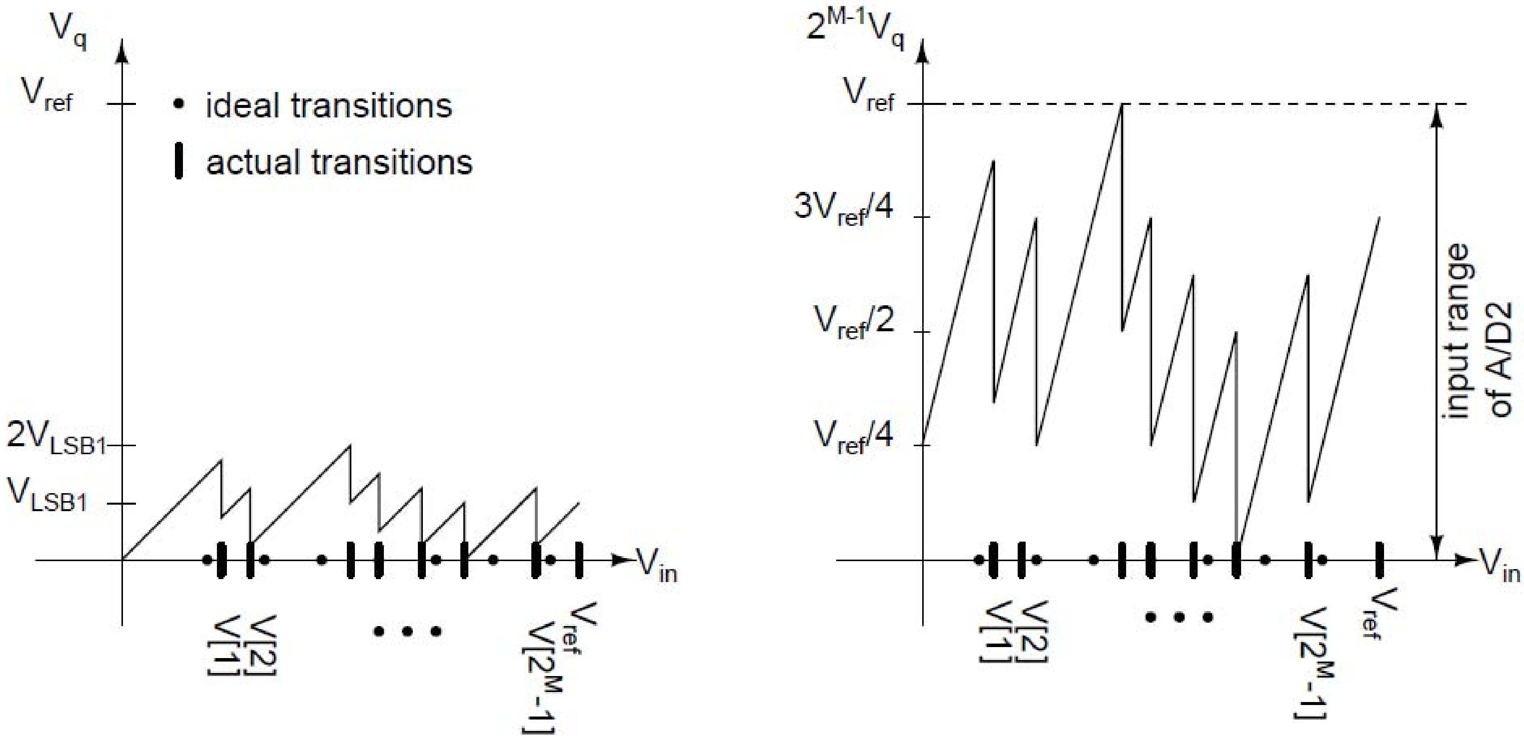
- Use reduced interstage gain of 2^{M-1}
- Modification: Shift the transitions of A/D1 to the right by $V_{ref}/2^{M+1}$ (0.5 LSB_1) to keep V_q positive
  - Eliminates analog offset addition and achieves same effect as last scheme
- Overall accuracy is $N = M + K - 1$ bits; A/D1 contributes
  - ✓ $M - 1$ bits, A/D2 contributes K bits; 1 bit redundancy
- Output $n = 2^{K-1}m + k$, no digital subtraction needed
  - ✓ Simpler digital logic

Two-step A/D with DR(II)-Ideal A/D1 Scenario



- $2^{M-1}V_q$ varies from 0 to $3V_{ref}/4$; $V_{ref}/4$ to $3V_{ref}/4$ except the first segment
- $2^{M-1}V_q$ outside this range implies errors in A/D1

Two-step A/D with DR (II): M bit acc. A/D1



- $2^{M-1}V_q$ varies from 0 to V_{ref}
- A/D2 is not overloaded for up to 0.5 LSB errors in A/D1

Two-step A/D with DR(II): M-bit acc. A/D1

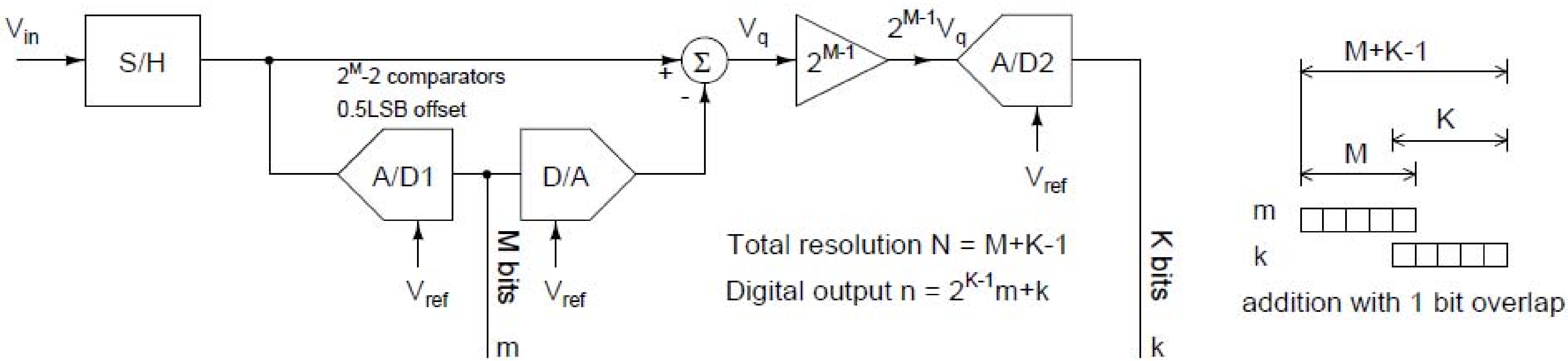
- Recall that output $n = 2^{K-1}m + k$

- A/D1 Transition shifted to the left
  - m greater than its ideal value by 1
  - k lesser than its ideal value by 2^{K-1}
  - A/D output $n = 2^{K-1}m + k$ doesn't change

- A/D1 Transition shifted to the right
  - m lesser than its ideal value by 1
  - k greater than its ideal value by 2^{K-1}
  - A/D output $n = 2^{K-1}m + k$ doesn't change

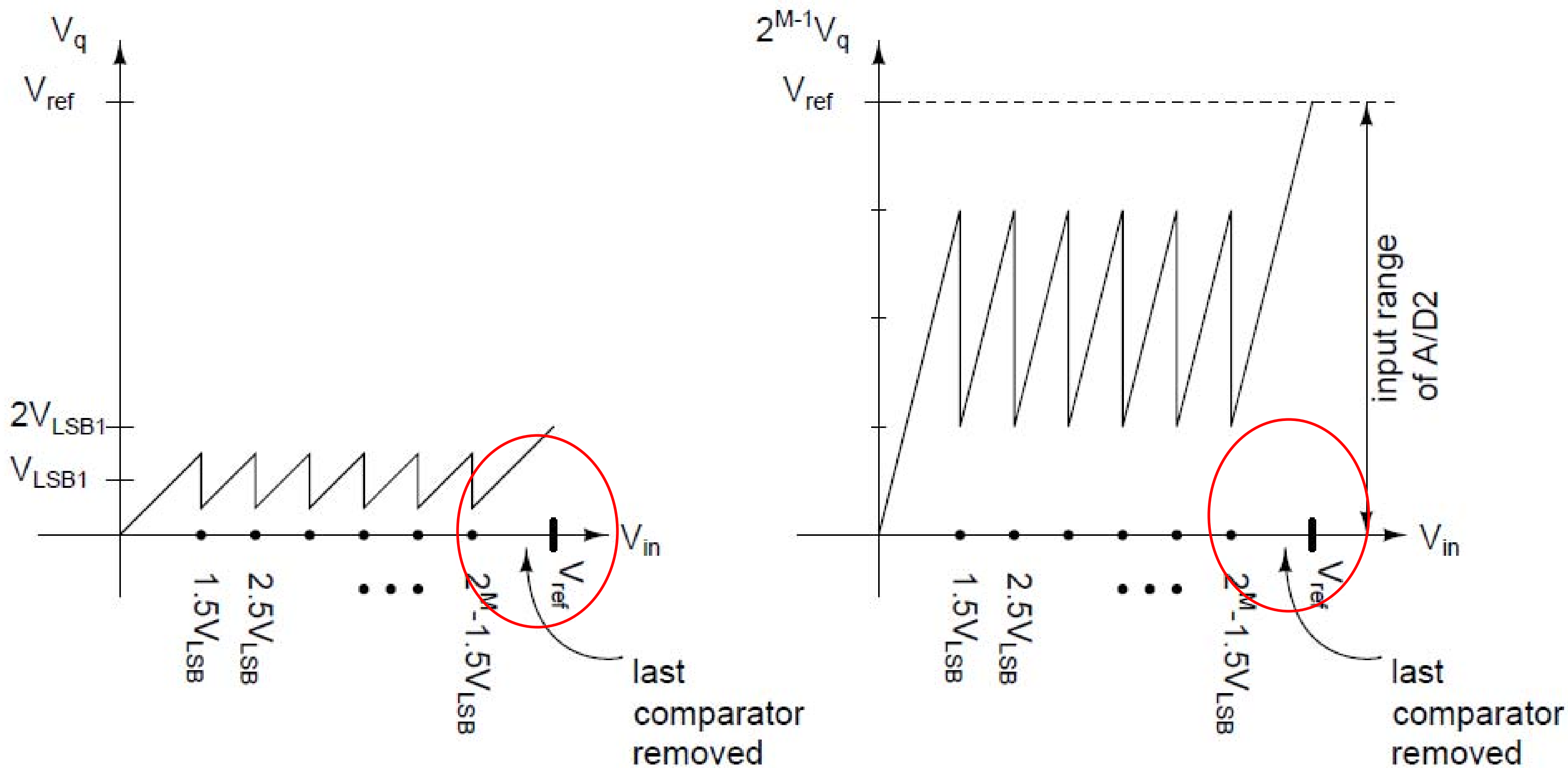
- 1 LSB error in m can be corrected

Two-step A/D with DR (III)



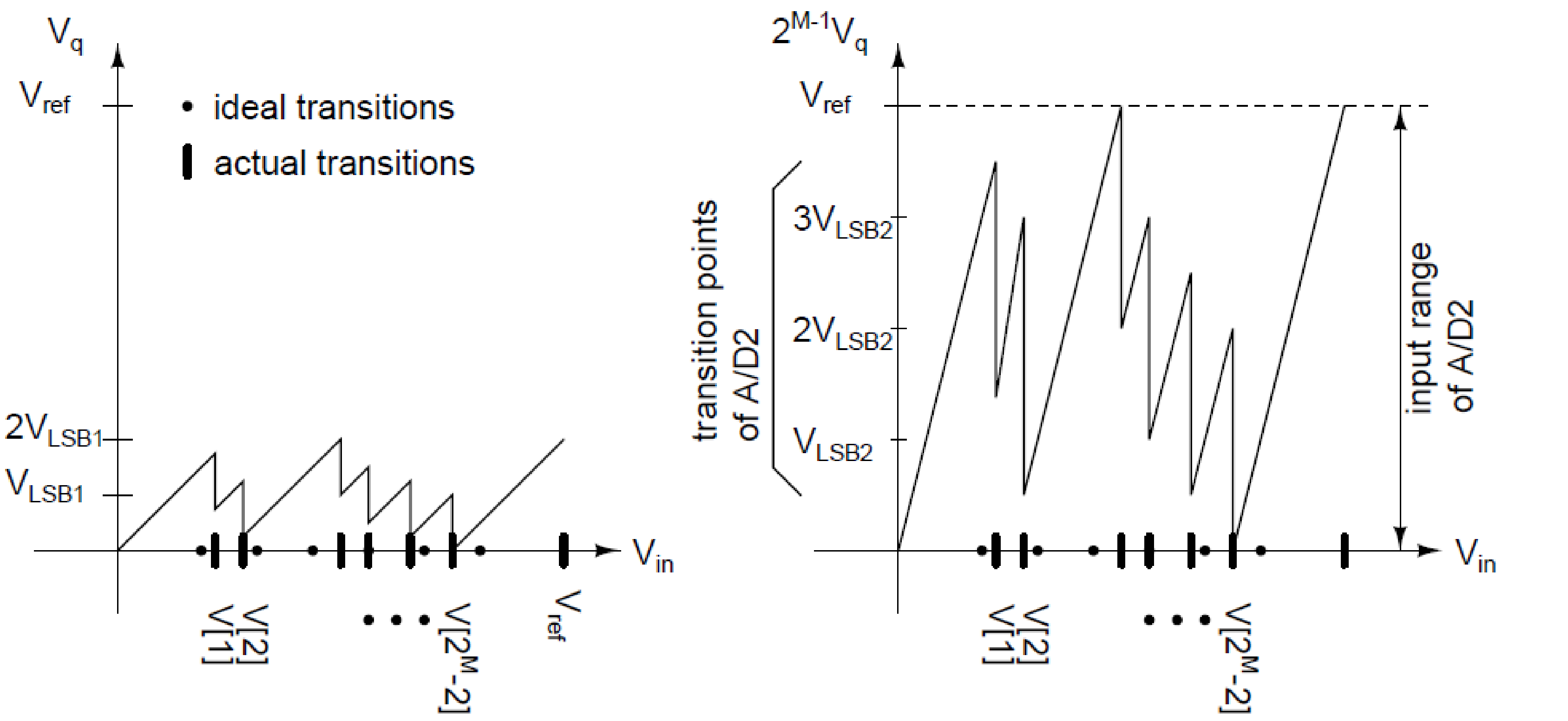
- 0.5 LSB ($V_{ref}/2^{M-1}$) shifts in A/D1 transitions can be tolerated
- If the last transition ($V_{ref} - V_{ref}/2^{M-1}$) shifts to the right by $V_{ref}/2^{M-1}$, the transition is effectively nonexistent
  - Still the A/D output is correct
- Remove last comparator $\Rightarrow$ M bit A/D1 has 2^{M-2} comparators set to $1.5V_{ref}/2^M$, $2.5V_{ref}/2^M$, . . . ,$V_{ref} - 1.5V_{ref}/2^M$
- Reduced number of comparators

Two-step A/D with DEC (III)-Ideal A/D1



- $2^{M-1}V_q$ varies from 0 to $3V_{ref}/4$; $V_{ref}/4$ to $3V_{ref}/4$ except the first and last segments
- $2^{M-1}V_q$ outside this range implies errors in A/D1

Two-step A/D with DR (III): M bit acc. A/D1



- $2^{M-1}V_q$ varies from 0 to V_{ref}
- A/D2 is not overloaded for up to 0.5 LSB errors in A/D1

Two-step A/D with DR(III): M-bit acc. A/D1

- Recall that output $n = 2^{K-1}m + k$

- A/D1 Transition shifted to the left
  - m greater than its ideal value by 1
  - k lesser than its ideal value by 2^{K-1}
  - A/D output $n = 2^{K-1}m + k$ doesn't change

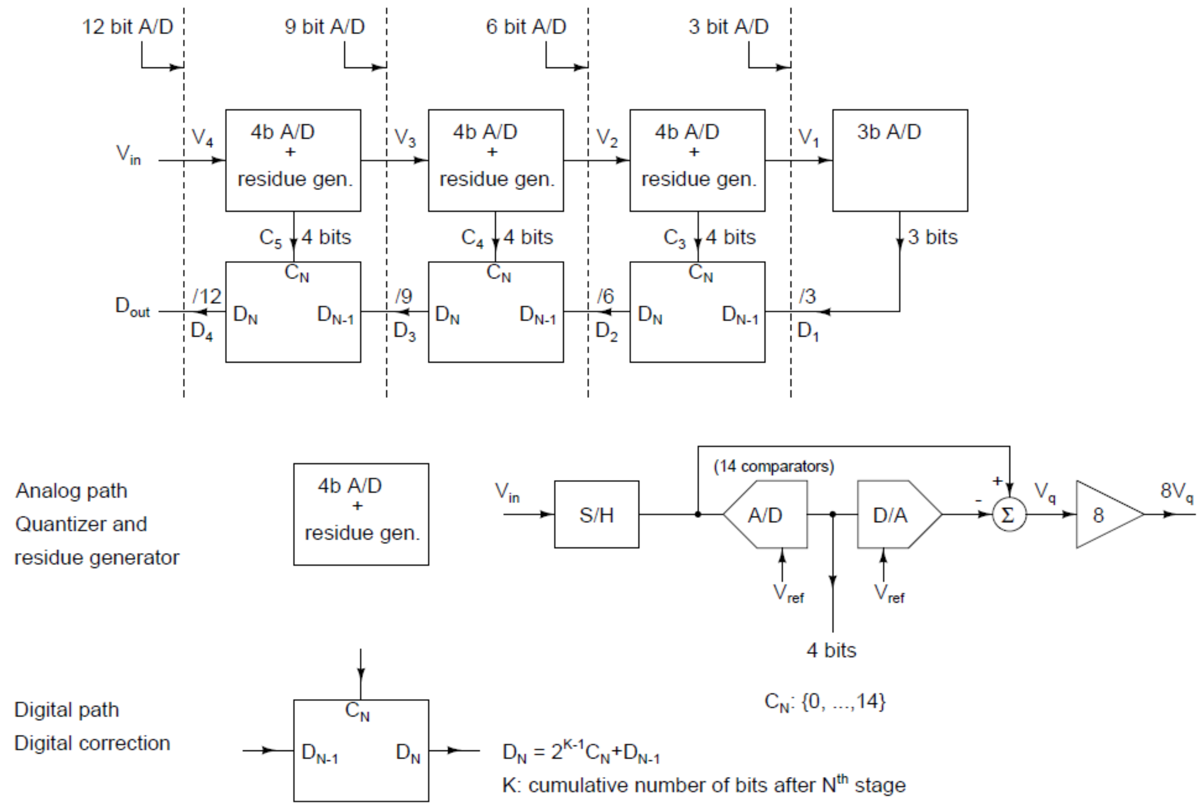
- A/D1 Transition shifted to the right
  - m lesser than its ideal value by 1
  - k greater than its ideal value by 2^{K-1}
  - A/D output $n = 2^{K-1}m + k$ doesn't change

- 1 LSB error in m can be corrected

Multi-step Converters

- Two-step architecture can be extended to multiple steps
- All stages except the last have their outputs digitally corrected from the following A/D output
- Number of effective bits in each stage is one less than the stage A/D resolution
- Accuracy of components in each stage depends on the accuracy of the A/D converter following it
- Accuracy requirements less stringent down the pipeline, but optimizing every stage separately increases design effort
- Pipelined operation to obtain high sampling rates
- Last stage is not digitally corrected

Multi-step or Pipelined A/D Converter



- 4,4,4,3 bits for an effective resolution of 12 bits
- 3 effective bits per stage
- Digital outputs appropriately delayed (by 2^{K-1}) before addition

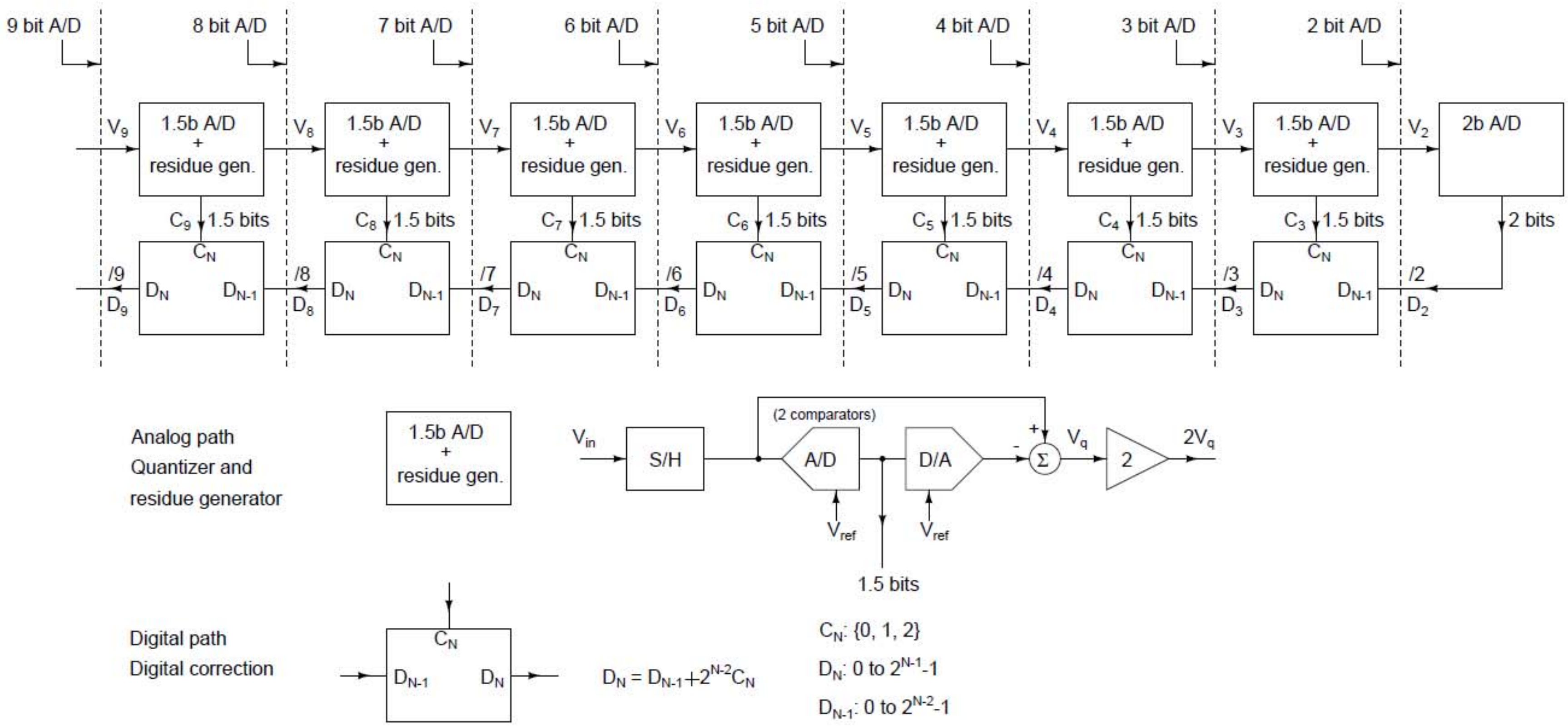
Multi-step Converter Tradeoffs

- Large number of stages, fewer bits per stage
  - Fewer comparators, low accuracy-lower power consumption
  - Larger number of amplifiers-power consumption increases
  - Larger latency
- Fewer stages, more bits per stage
  - More comparators, higher accuracy designs
  - Smaller number of amplifiers-lower power consumption
  - Smaller latency
- Typically 3-4 bits per stage easy to design

1.5b/Stage Pipelined A/D Converter

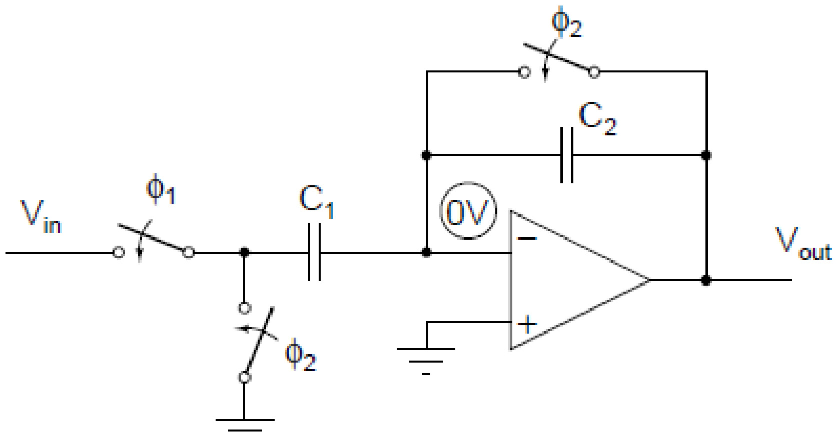
- To resolve 1 effective bit per stage, you need $2^2 - 2$, i.e. two comparators per stage
- Two comparators result in a 1.5 bit conversion (3 levels)
- Using two comparators instead of three (required for a 2 bit converter in each stage) results in significant savings

1.5b/Stage Pipelined A/D Converter

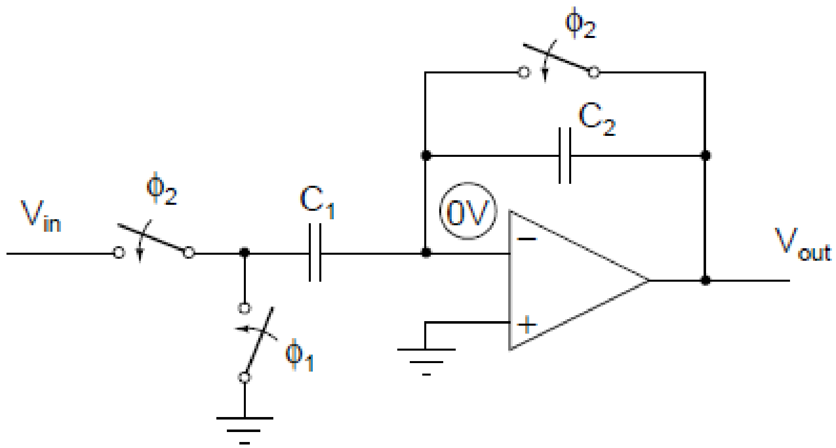


- Digital outputs appropriately delayed (by 2^{N-2}) before addition
- Note the 1-bit overlap when C_N is added to D_{N-1}
  - Use half adders for stages 2 to N

SC Amplifiers

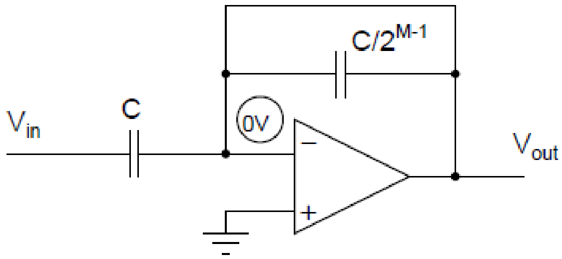


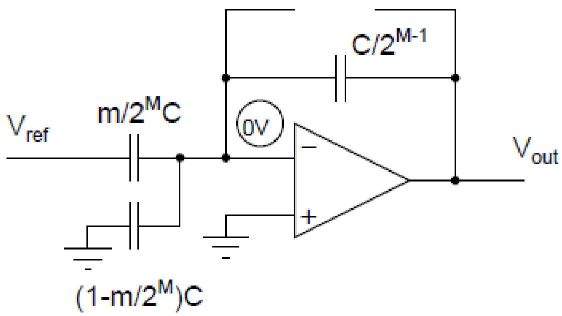
$$V_{out} = -(C_1/C_2)V_{in}$$



$$V_{out} = +(C_1/C_2)V_{in}$$

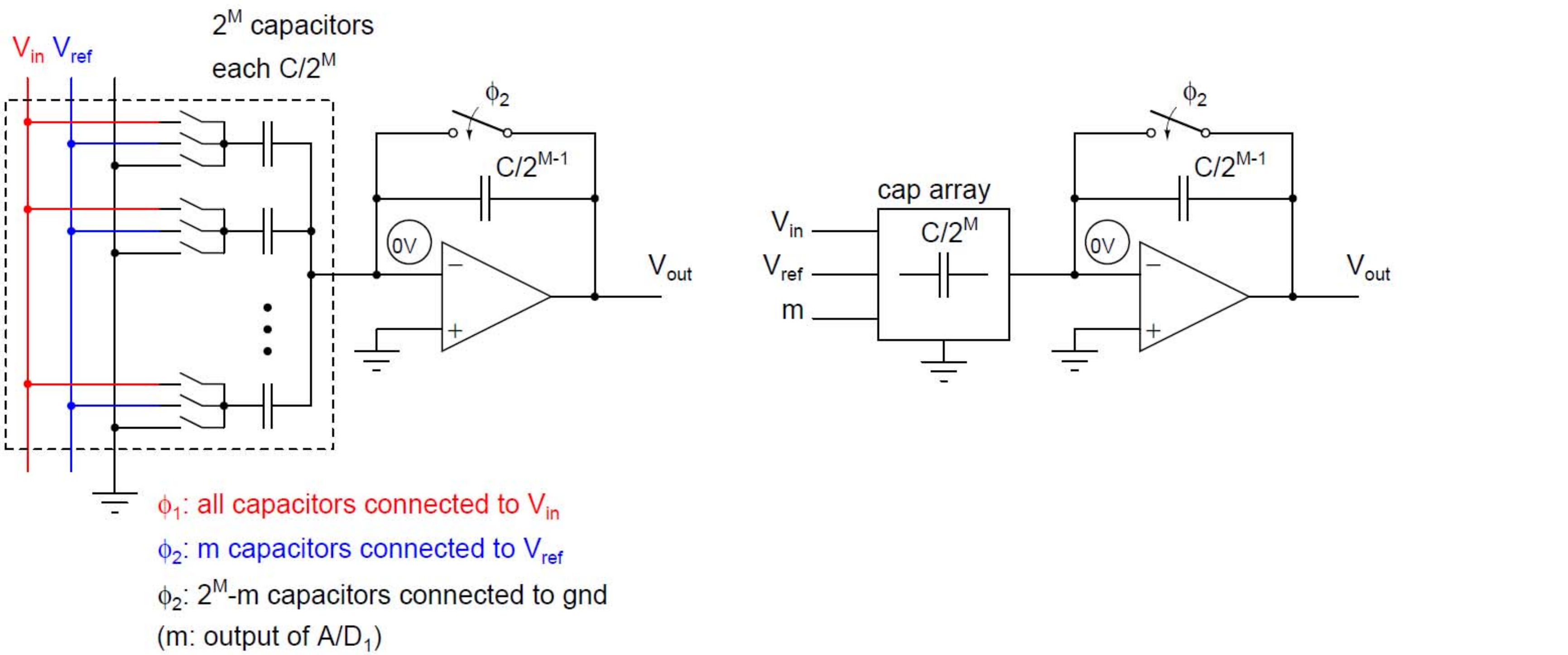
SC Realization (I) of DAC and Amplifier





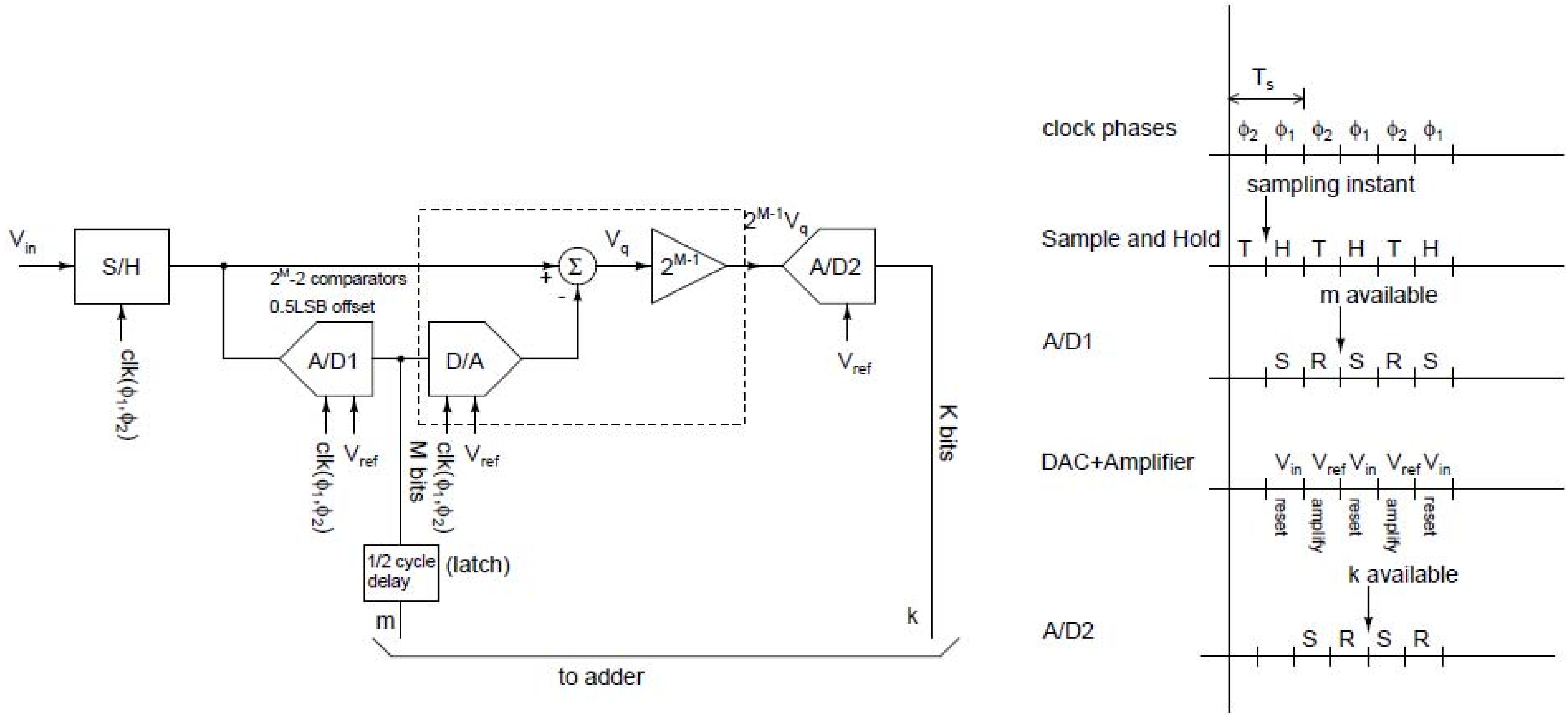
- Pipelined A/D needs DAC, subtractor, and amplifier
- V_{in} sampled on C in Φ_2 (positive gain)
- V_{ref} sampled on $m/2^M C$ in Φ_1 (negative gain).
- At the end of Φ_1, $V_{out} = 2^{M-1} (V_{in} - m/2^M V_{ref})$

SC Realization of DAC and Amplifier



- $m/2^MC$ realized using a switched capacitor array controlled by A/D1 output

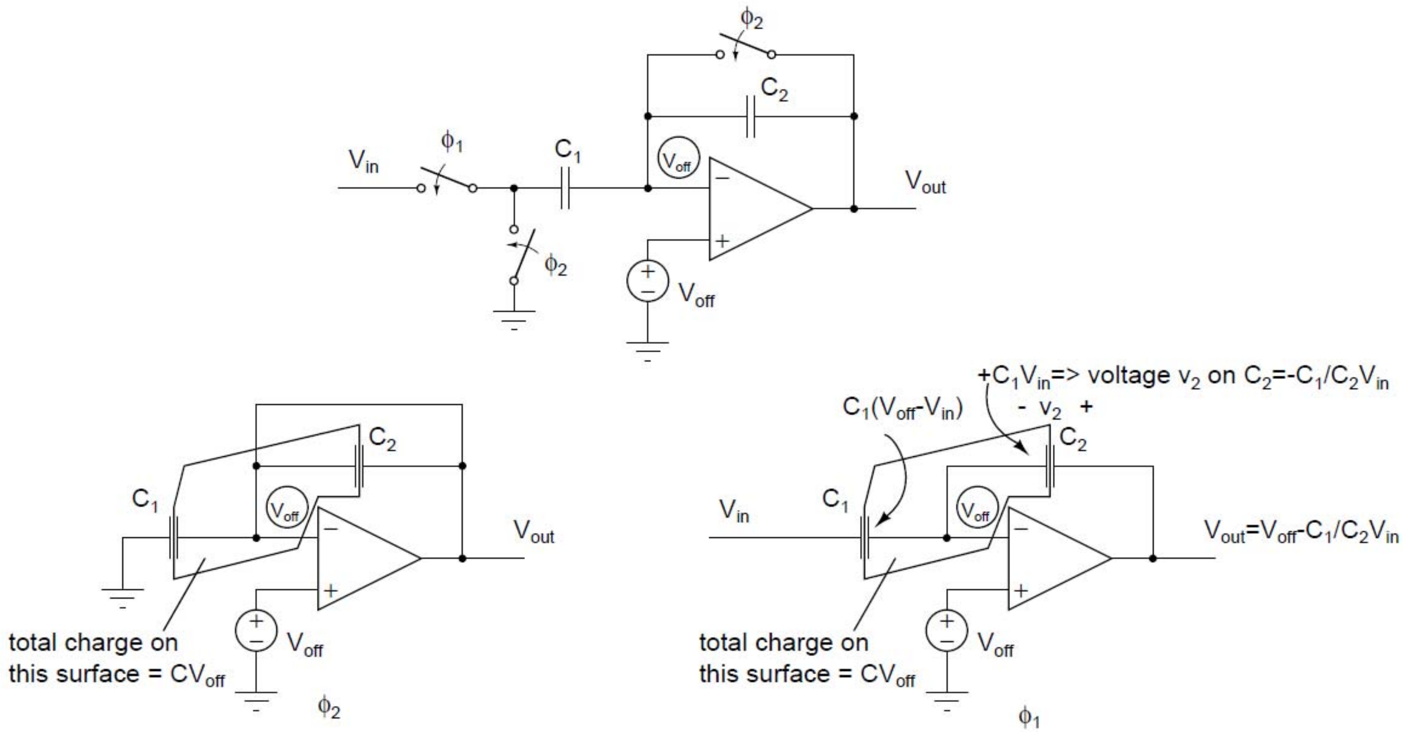
Two stage converter timing and pipelining



Two stage converter timing and pipelining

- Φ_1
  - S/H holds the input $V_i[n]$ from the end of previous Φ_2
  - A/D1 samples the output of S/H
  - Amplifier samples the output of S/H on C
  - Opamp is reset
- Φ_2
  - S/H tracks the input
  - A/D1 regenerates the digital value m
  - Amplifier samples V_{ref} of S/H on $m/2^M C$
  - Opamp output settles to the amplified residue
  - A/D2 samples the amplified residue
- Φ_2
  - A/D2 regenerates the digital value k. m, delayed by ½ clock cycle, can be added to this to obtain the final output
  - S/H, A/D1, Amplifier function as before, but on the next
  - sample $V_i[n+1]$
- In a multistep A/D, the phase of the second stage is reversed when compared to the first, phase of the third stage is the same as the first, and so on

Effect of opamp offset



- Φ_2: C_1 is charged to V_{in} - V_{off} instead of V_{in}
  - input offset cancellation; no offset in voltage across C_2
- Φ_2 : $V_{out} = -C_1/C_2V_{in} + V_{off}$
  - Unity gain for offset instead of 1 + C_1/C_2 (as in a continuous time amplifier)

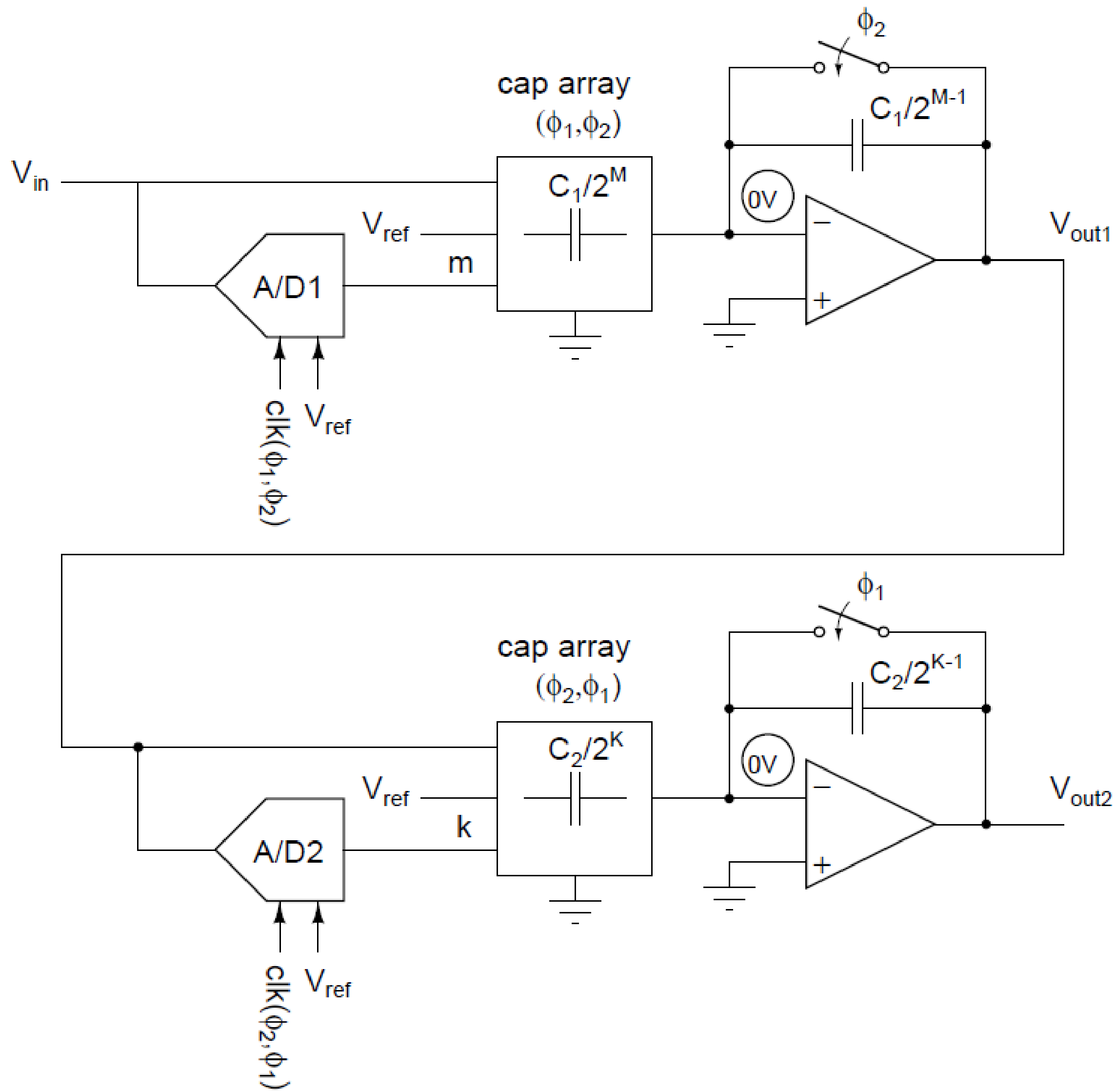
Circuit Non-idealities

- Random mismatch
  - Capacitors must be large enough (relative matching $\propto \frac{1}{\sqrt{WL}}$ to maintain DAC and amplifier accuracy
- Thermal noise
  - Capacitors must be large enough to limit noise below 1 LSB
  - Opamp's input referred noise should be small enough.
- Opamp DC gain
  - Should be large enough to reduce amplifier's output error to $V_{ref}/2^{K+1}$
- Opamp Bandwidth
  - Should be large enough for amplifier's output settling error to be less than $V_{ref}/2^{K+1}$

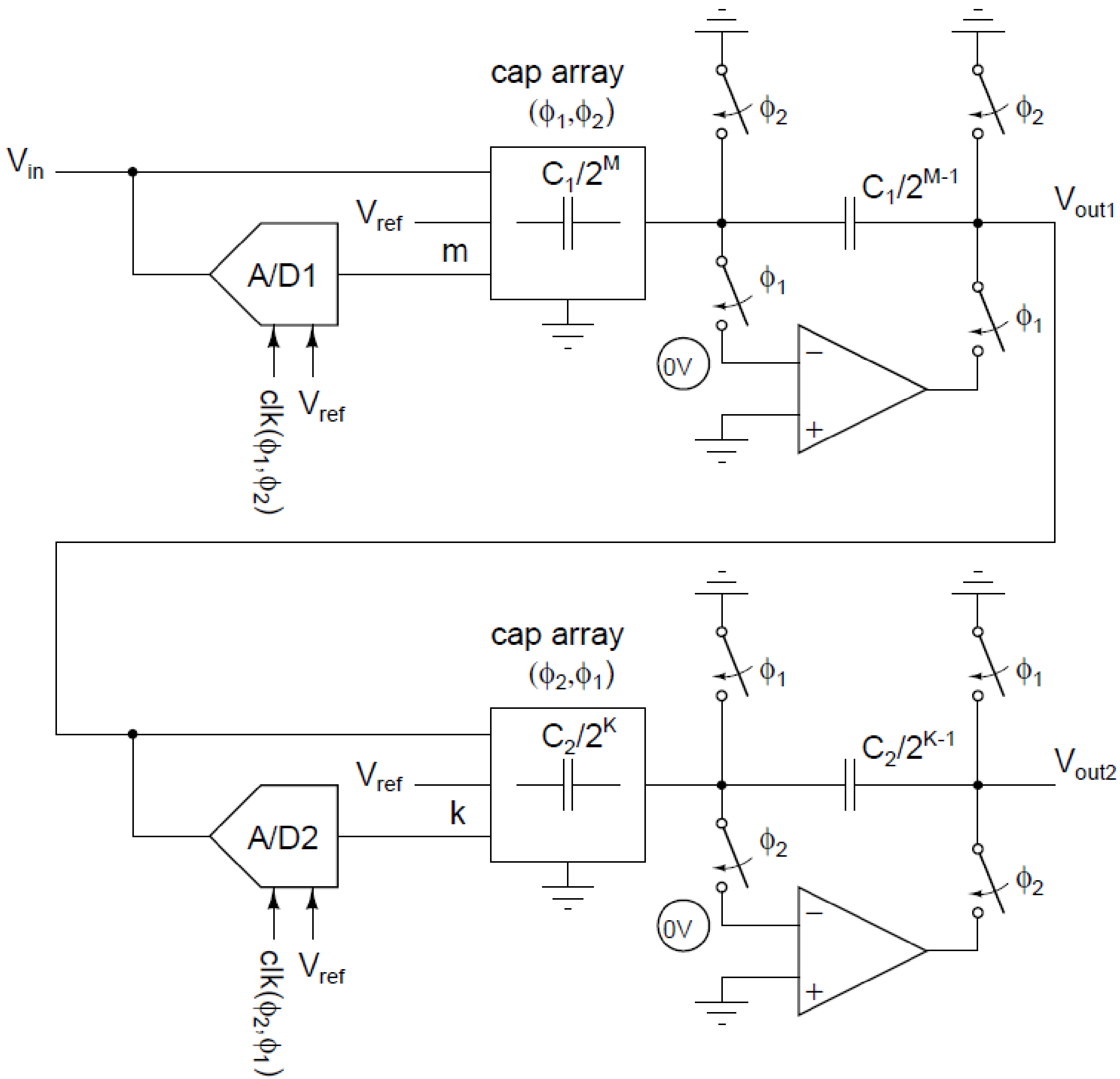
Opamp Power Consumption

- Opamp power consumption a large fraction of the converter power consumption
- Amplification only in one phase
- Successive stages operate in alternate phases
- Share the amplifiers between successive stages

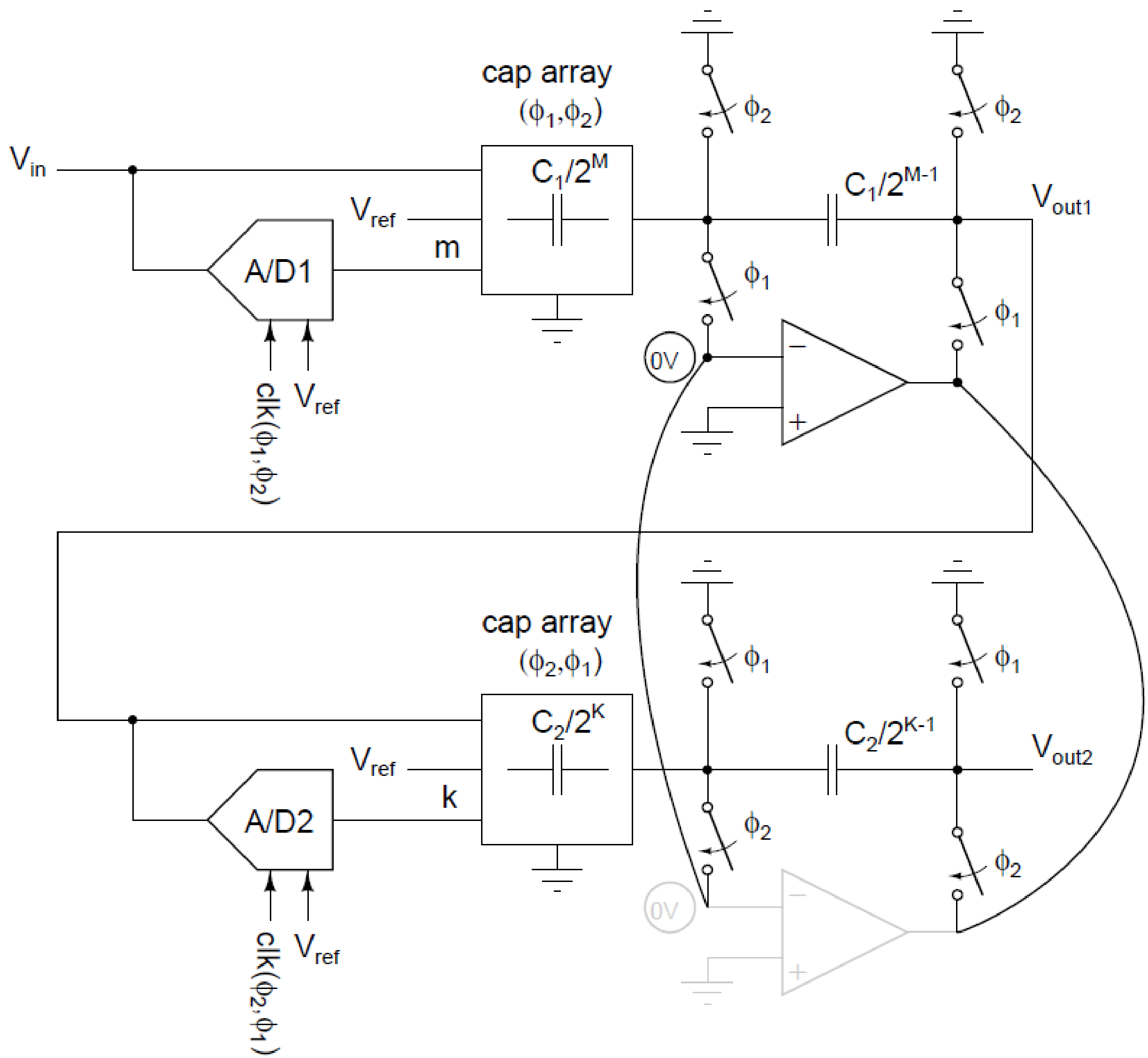
MDAC Stages: With Offset Cancellation



MDAC Stages: With Offset Cancellation



MDAC Stages: Opamp Sharing



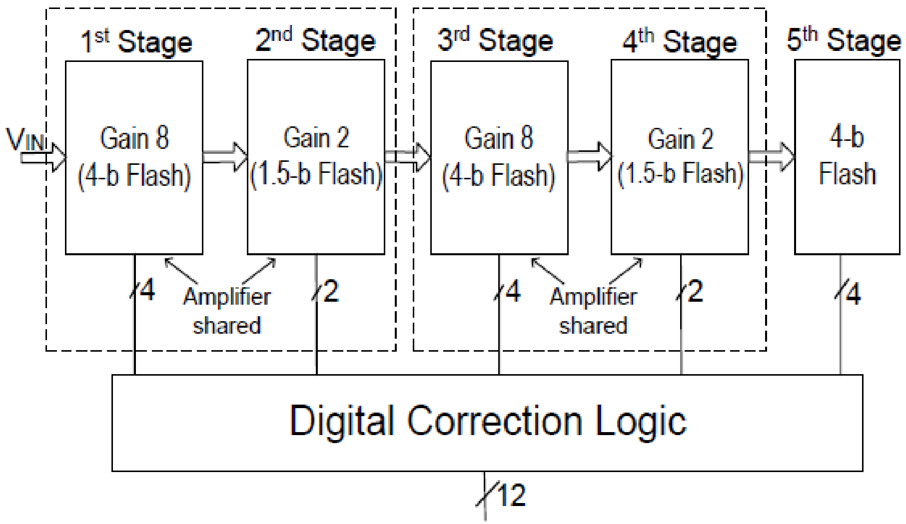
Opamp Sharing

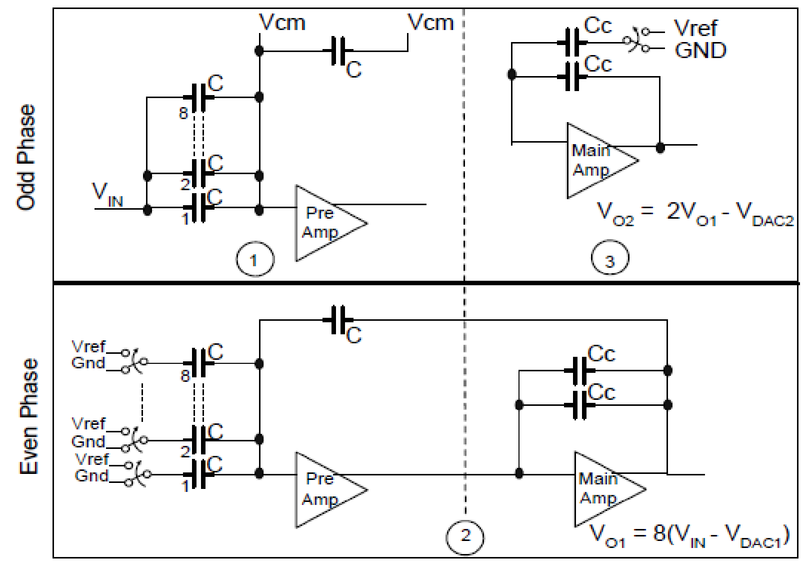
- First stage uses the opamp only in Φ_1
- Second stage uses the opamp only in Φ_2
- Use a single opamp
  - Switch it to first stage in Φ_1
  - Switch it to second stage in Φ_2
- Reduces power consumption
- Cannot correct for opamp offsets
- Memory effect because of charge storage at negative input of the opamp

Opamp Sharing: Split Amp

- Alternate stages with more and fewer bits e.g. 3-1-3-1
- Optimized loading
- Use a two stage opamp for stage 1 (High gain)
- Use a single stage opamp for stage 2 (Low gain)
- Use a single two stage opamp
  - Use both stages for stage 1 of the A/D
  - Use only the second stage for stage 2 of the A/D
  - Feedback capacitor in stage 2 of the A/D appears across the second stage of the opamp—Miller compensation capacitor
- Further Reduces power consumption

Opamp Sharing: Split Amp





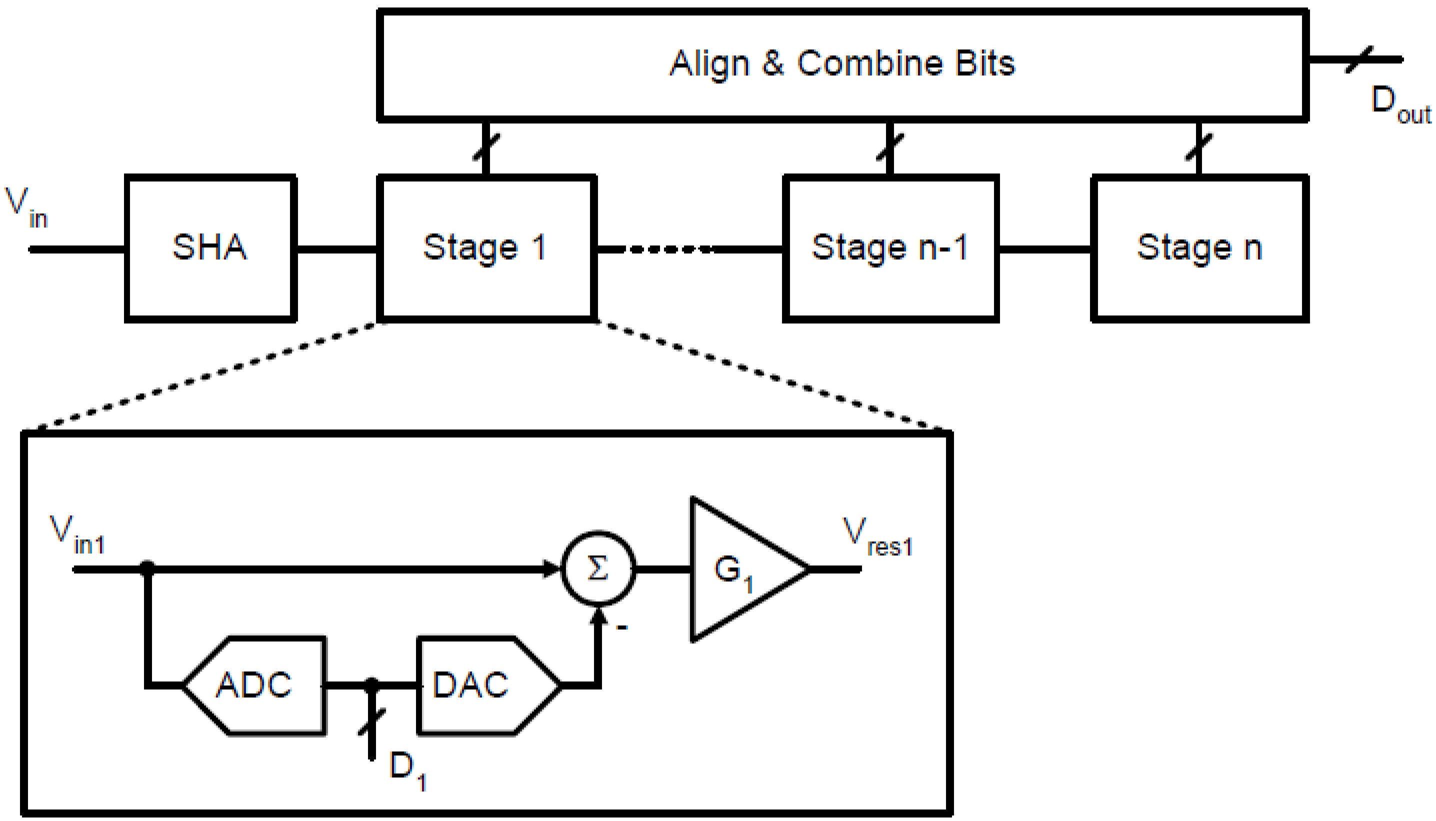
(1)- Sampling for high bits stage. (2)- Amplify for high bits stage and sampling for low bits stage (3)- Amplify for low bits stage

18.4 A 30mW 12b 21MSample/s Pipelined CMOS ADC

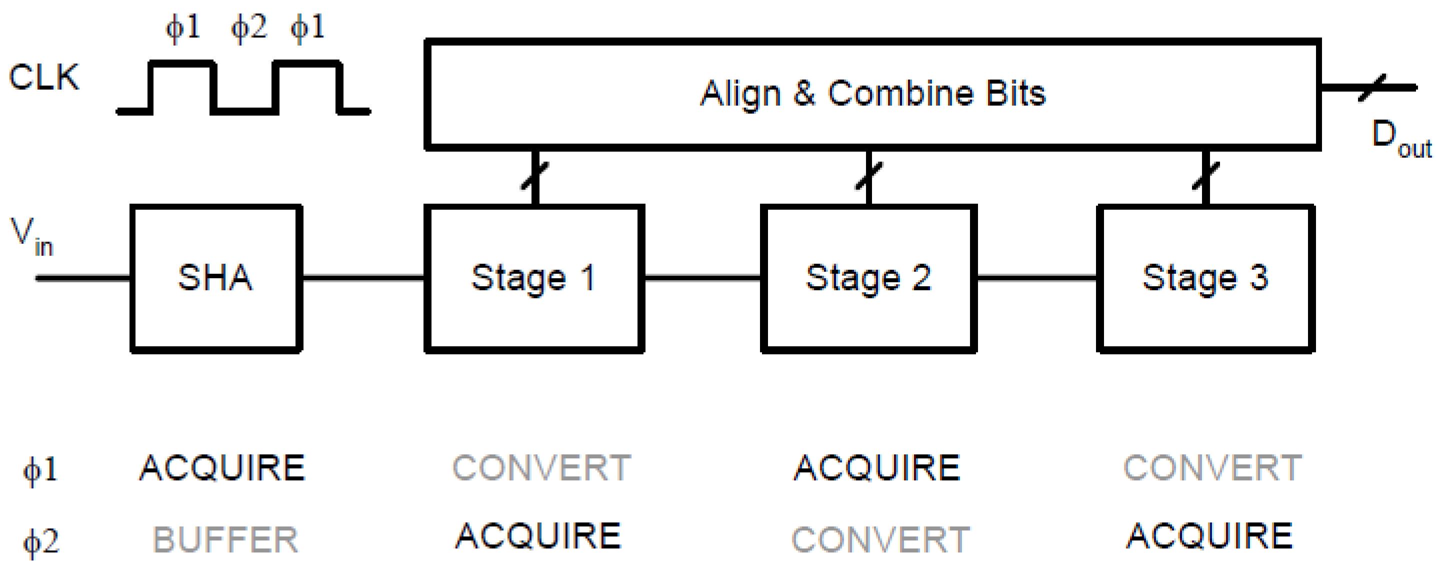
Suhas Kulhalli, Visvesvaraya Penkota, Ravishankar Asv

Pipelined A/D Implementation

Pipelined ADC Architecture

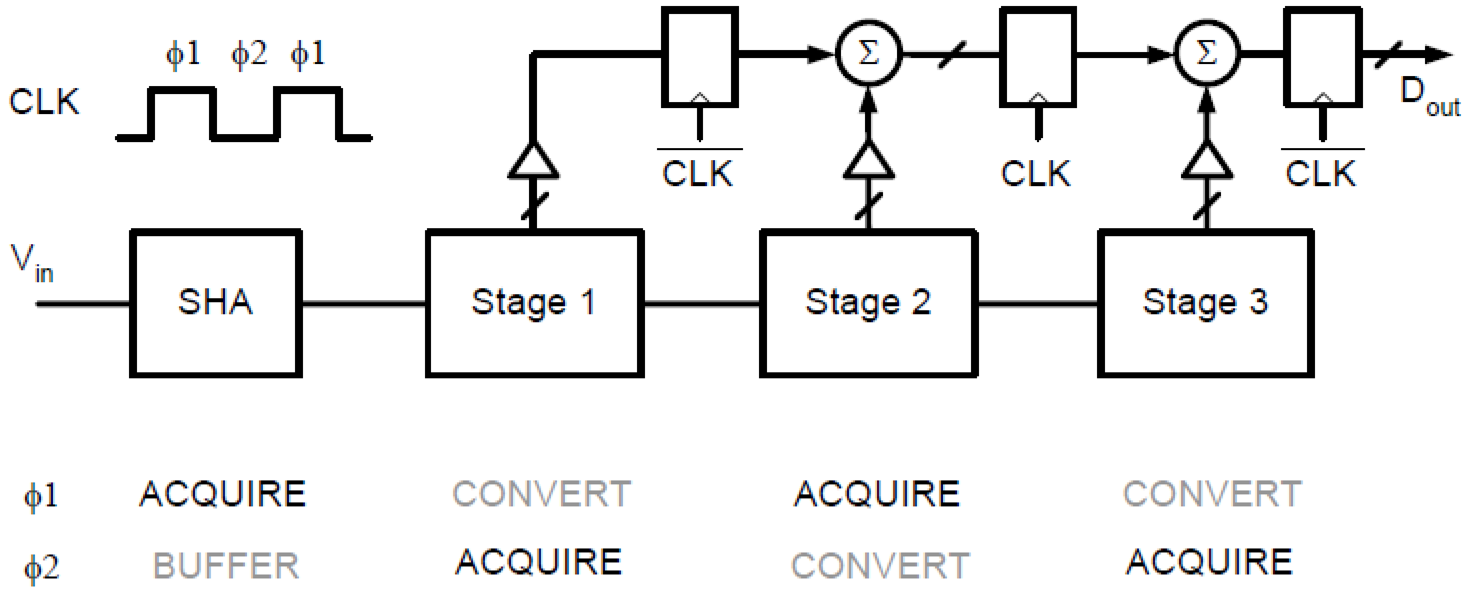


Concurrent Stage Operation



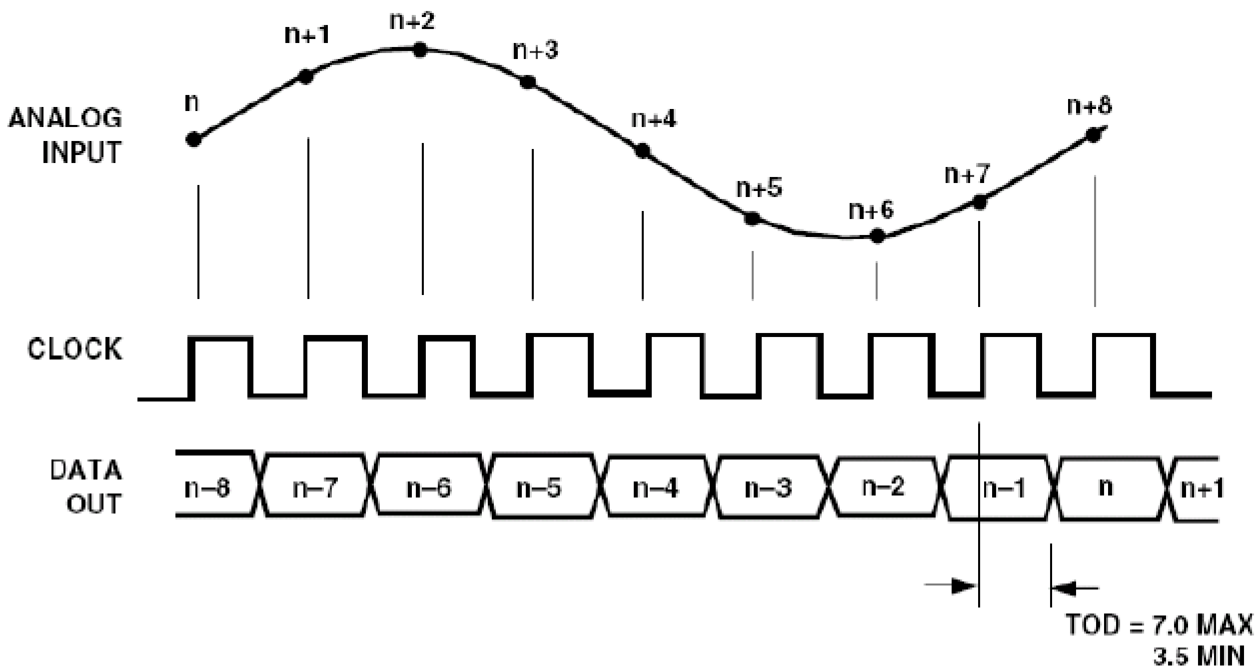
- Dedicated S/H for better dynamic performance
- Pipelined MDAC stages operate on the input and pass the scaled residue to the to the next stage
- New output every clock cycle, but each stage introduces 0.5 clock cycle latency

Data Alignment



- Digital shift register aligns sub-conversion results in time
- Digital output is taken as weighted sum of stage bits

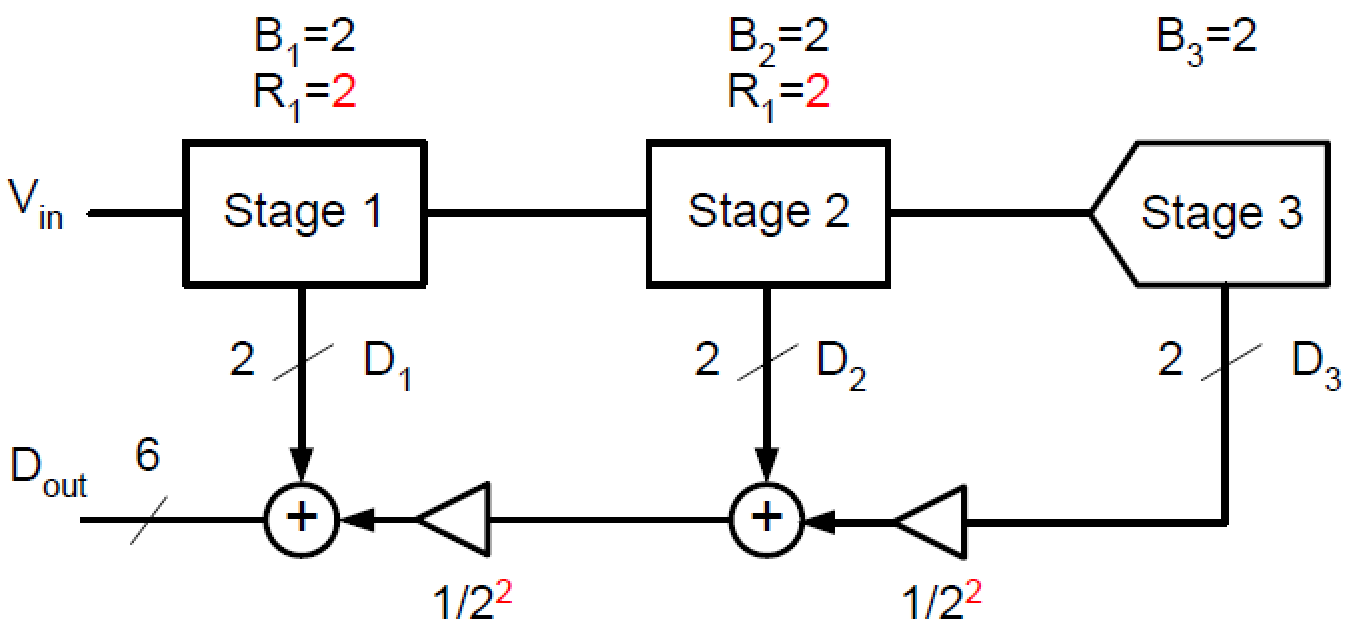
Latency



[Analog Devices, AD9226 Data Sheet]

Combining the Bits: Ideal MDAC

- Example1: Three 2-bit stages, no redundancy

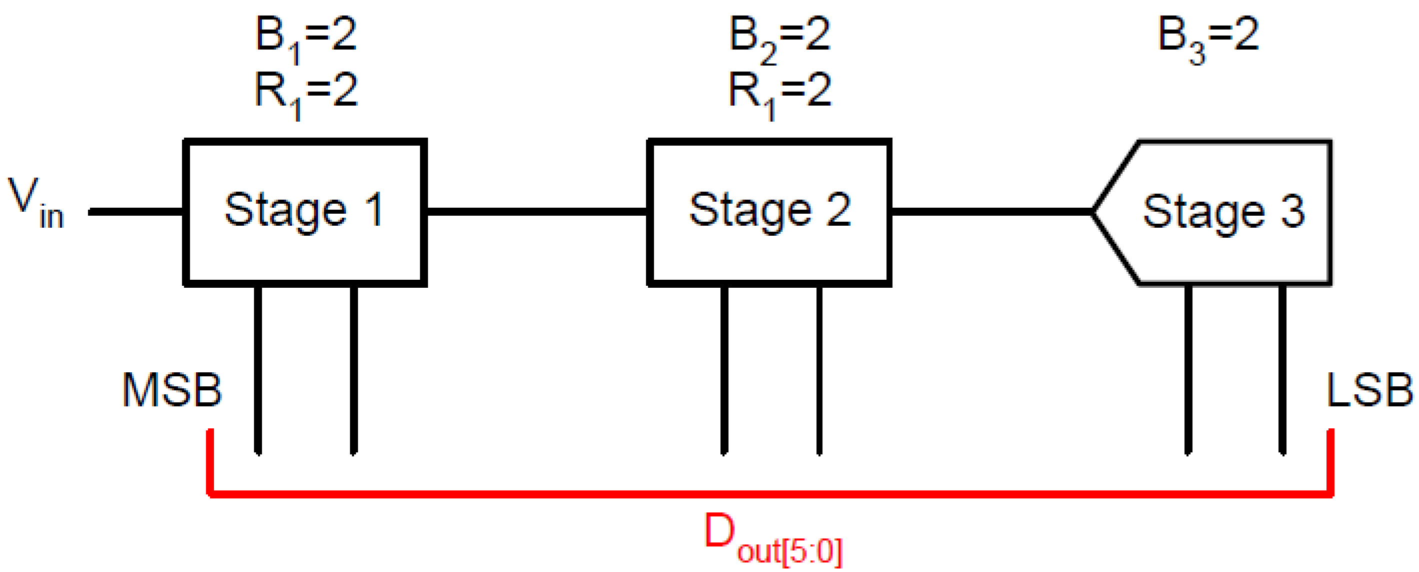


$$D_{out} = D_1 + \frac{1}{4} D_2 + \frac{1}{16} D_3$$

Combining the Bits contd.

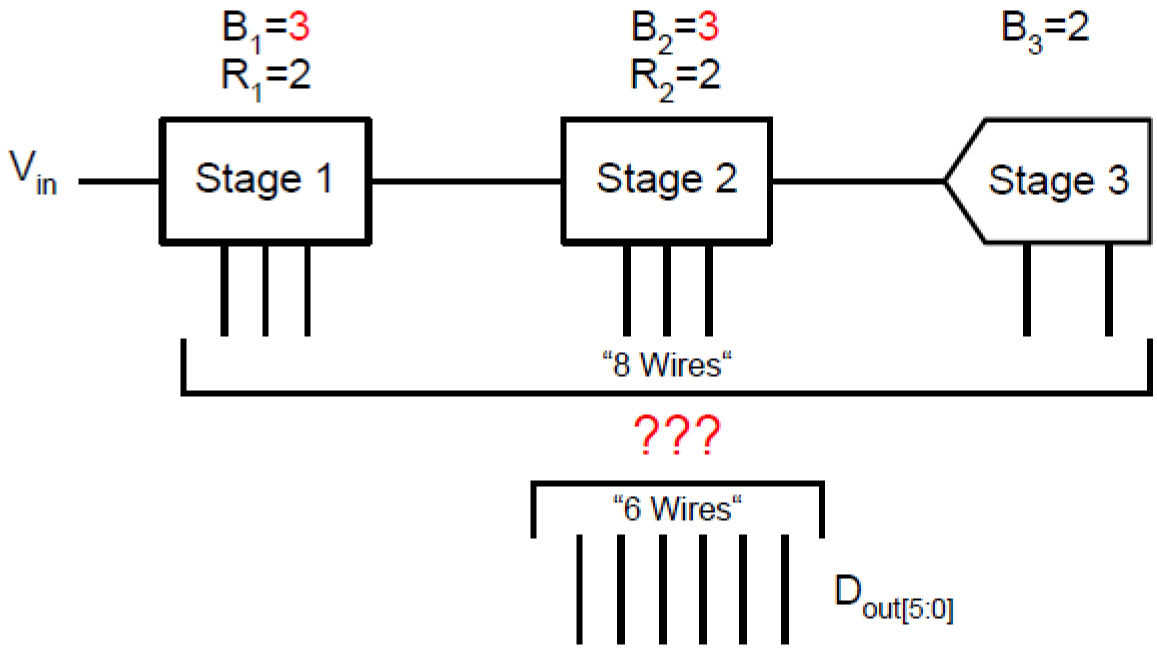
```
D1    XX
D2       XX
D3          XX
-------------
Dout  DDDDDD
```

- Only bit shifts
- No arithmetic circuits needed



Combining the Bits: With Redundancy

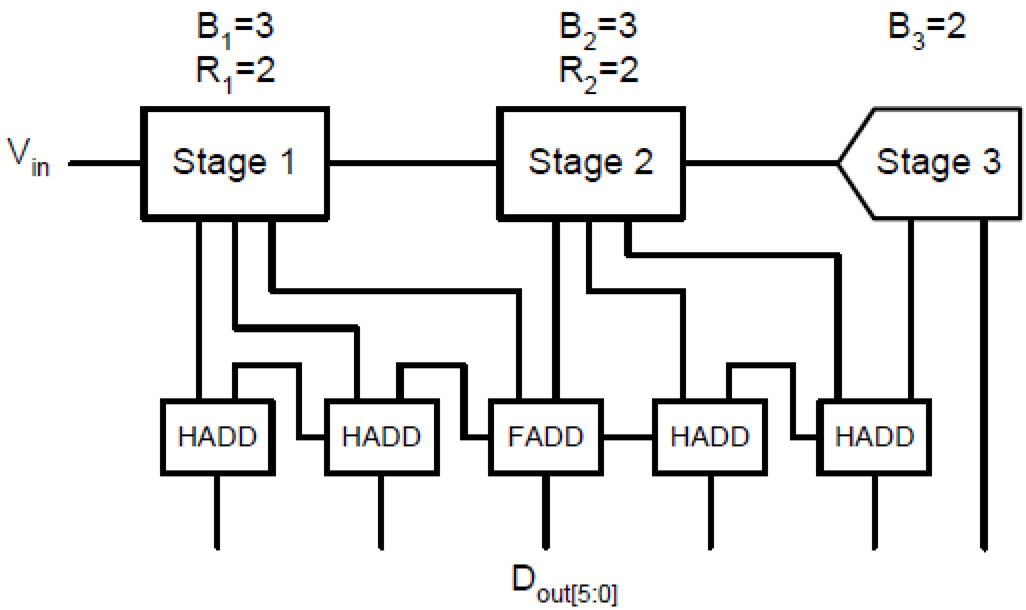
- Example2: Three 2-b it stages, one bit redundancy in stages 1 and 2 (6-bit aggregate ADC resolution)



Combining the Bits: With Redundancy

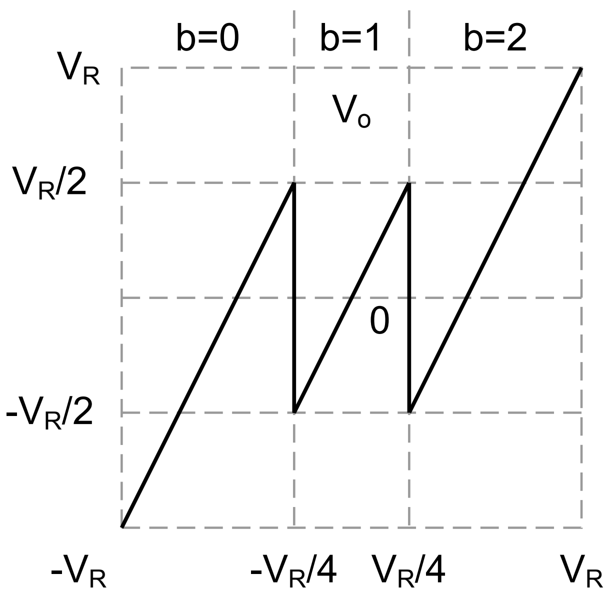
$$D_{out} = D_1 + \frac{1}{4}D_2 + \frac{1}{16}D_3$$

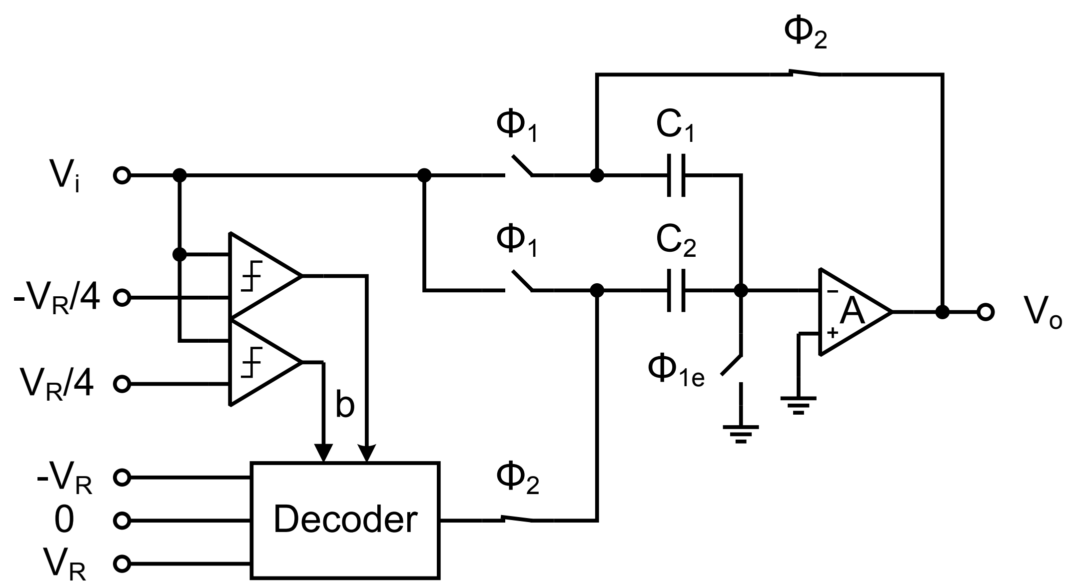
```
D1     XXX
D2       XXX
D3         XX
-------------
Dout   DDDDDD
```



- Bits overlap by the amount of redundancy
- Need half adders for addition

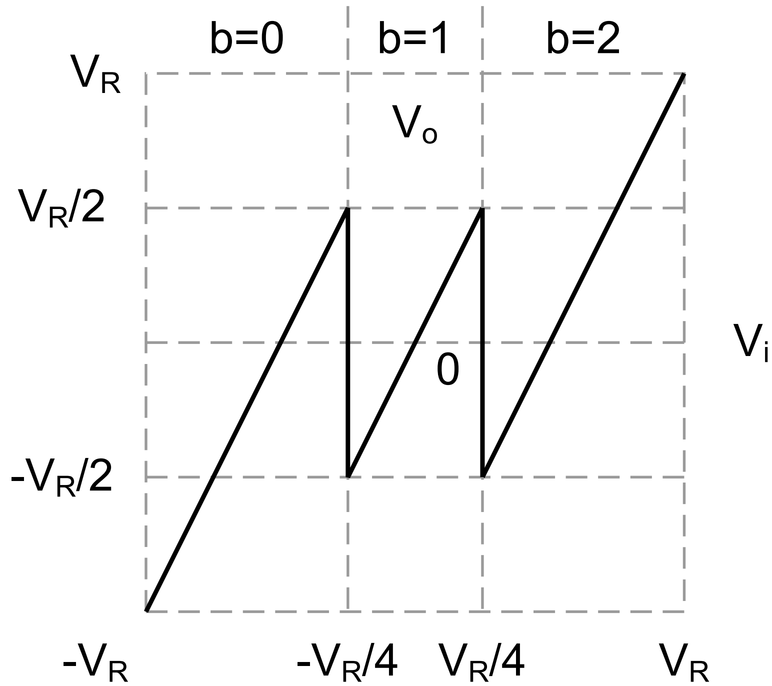
A 1.5-Bit Stage





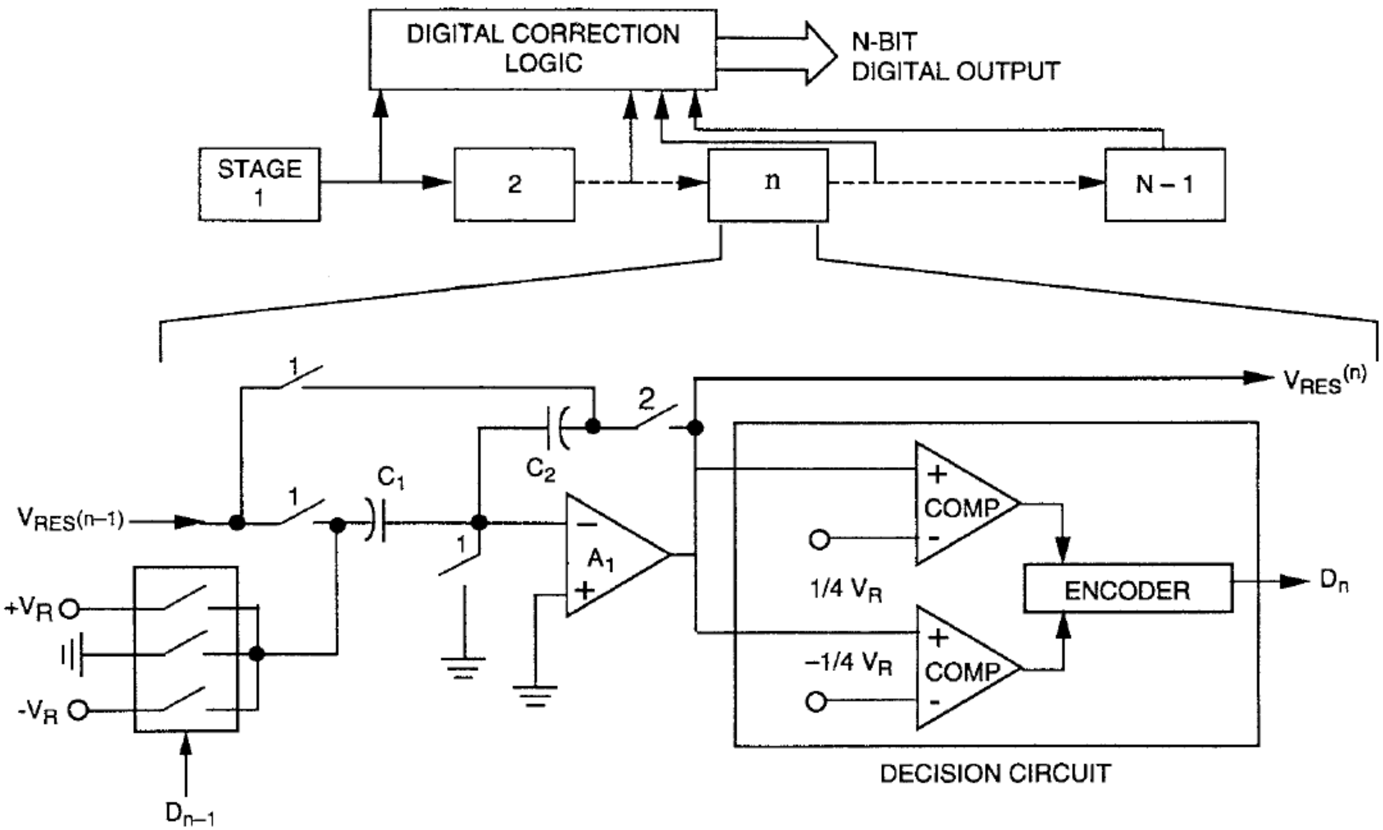
- 2X gain + 3-level DAC + subtraction all integrated
- Digital redundancy relaxes the tolerance on CMP/RA offsets

A 1.5-Bit Stage: Residue Plot



$$V_o = \begin{cases} \left(1 + \frac{C_s}{C_f}\right) V_i - \frac{C_s}{C_f} V_{ref} & \text{if } V_i > V_{ref}/4 \\ \left(1 + \frac{C_s}{C_f}\right) V_i & \text{if } -V_{ref}/4 \le V_i \le +V_{ref}/4 \\ \left(1 + \frac{C_s}{C_f}\right) V_i + \frac{C_s}{C_f} V_{ref} & \text{if } V_i < -V_{ref}/4 \end{cases}$$

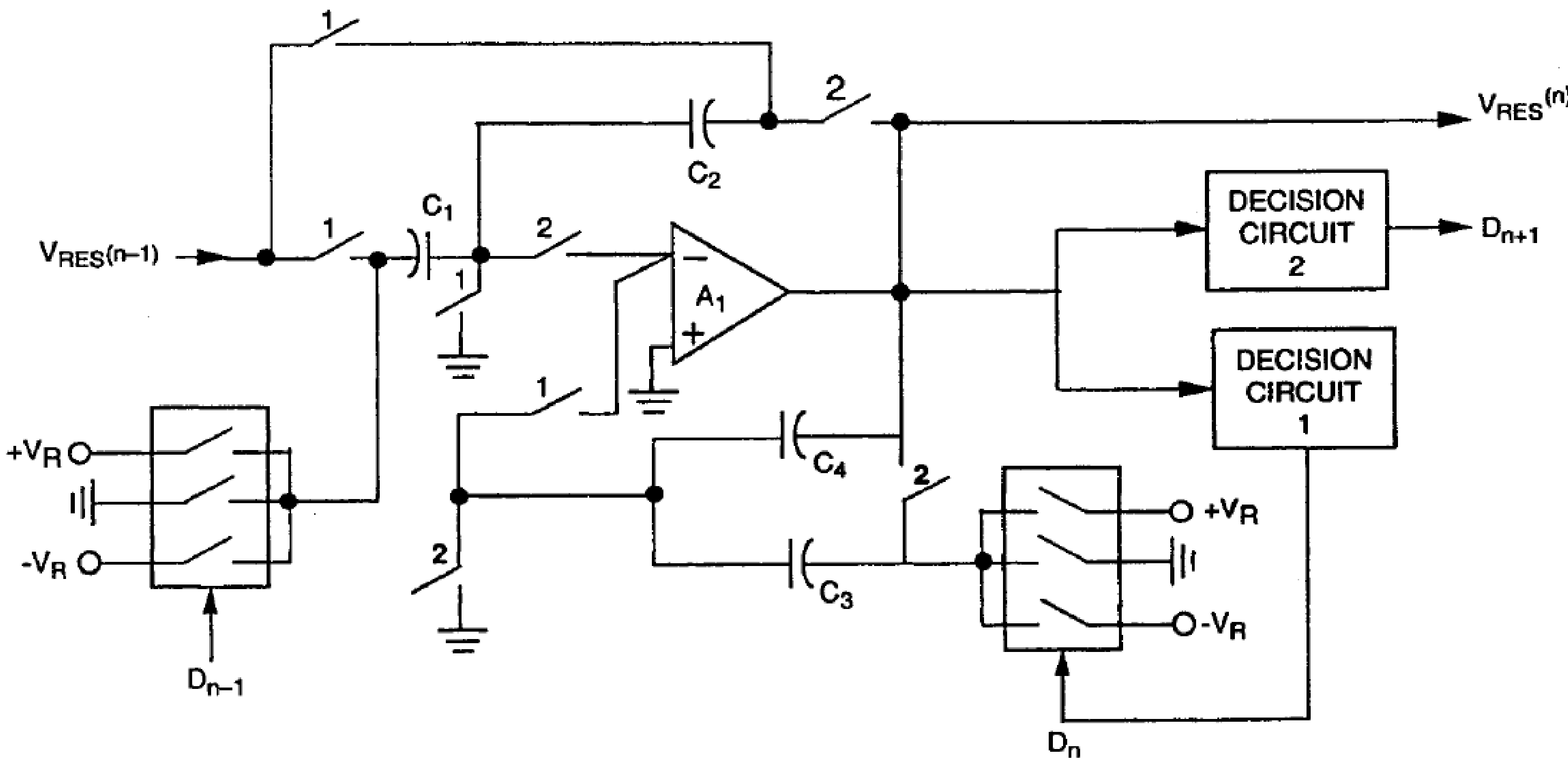
A 1.5-Bit Stage



A 250-mW, 8-b, 52-Msamples/s Parallel-Pipelined A/D Converter with Reduced Number of Amplifiers

Krishnaswamy Nagaraj, *Senior Member, IEEE*, H. Scott Fetterman, Joseph

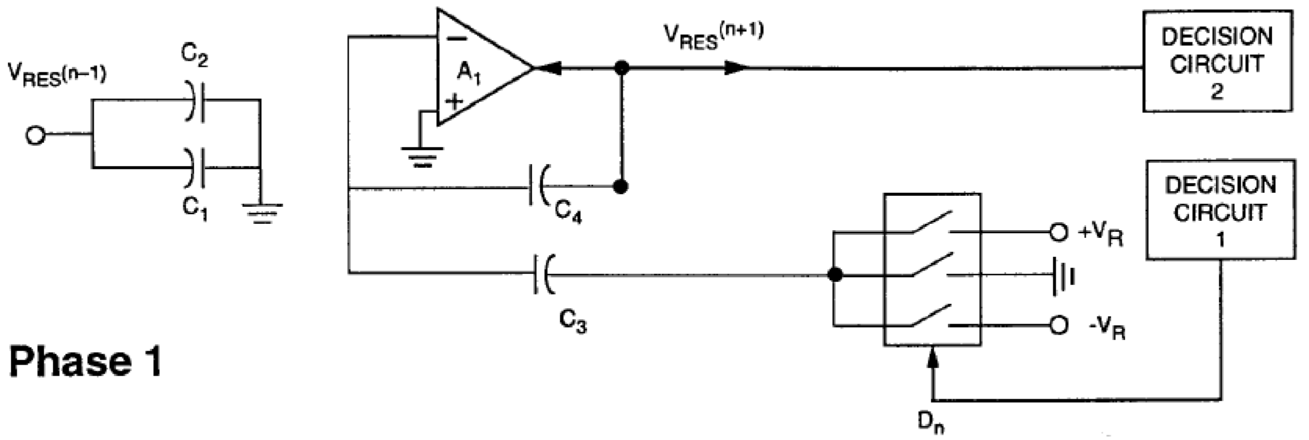
A 1.5-Bit Stage: Opamp Sharing

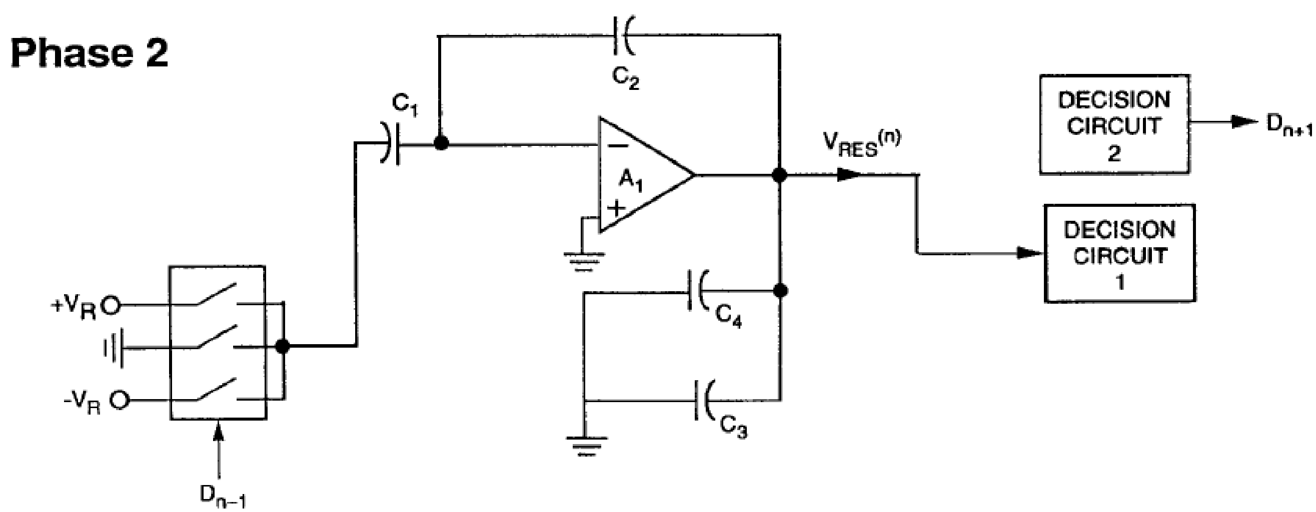


A 250-mW, 8-b, 52-Msamples/s Parallel-Pipelined A/D Converter with Reduced Number of Amplifiers

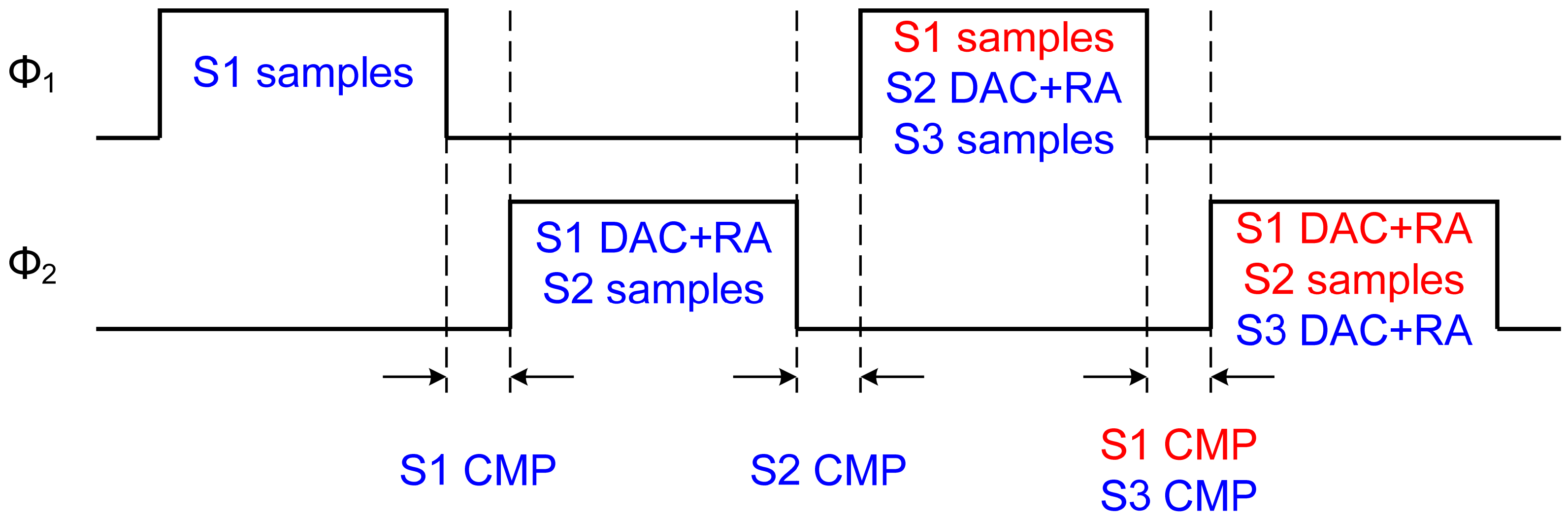
Krishnaswamy Nagaraj, *Senior Member, IEEE*, H. Scott Fetterman, Joseph

A 1.5-Bit Stage: Opamp Sharing contd.



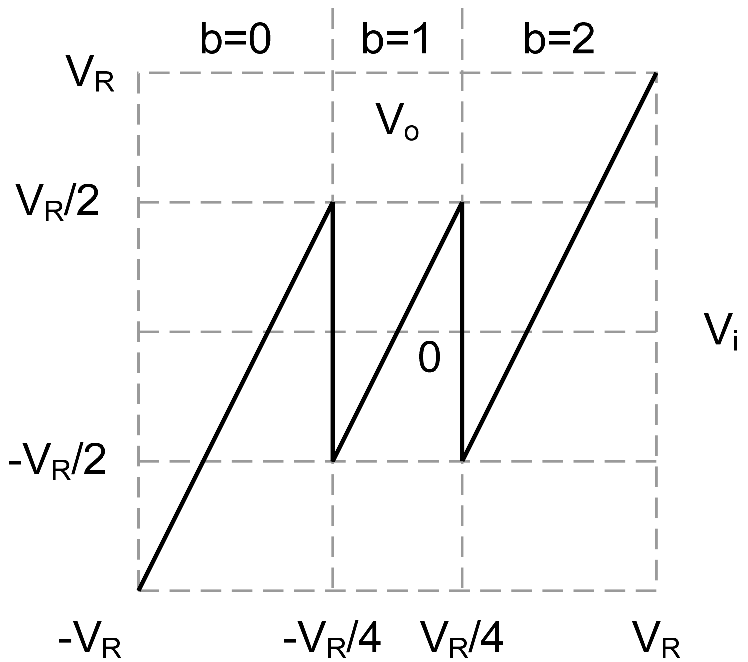


Timing Diagram of Pipelining



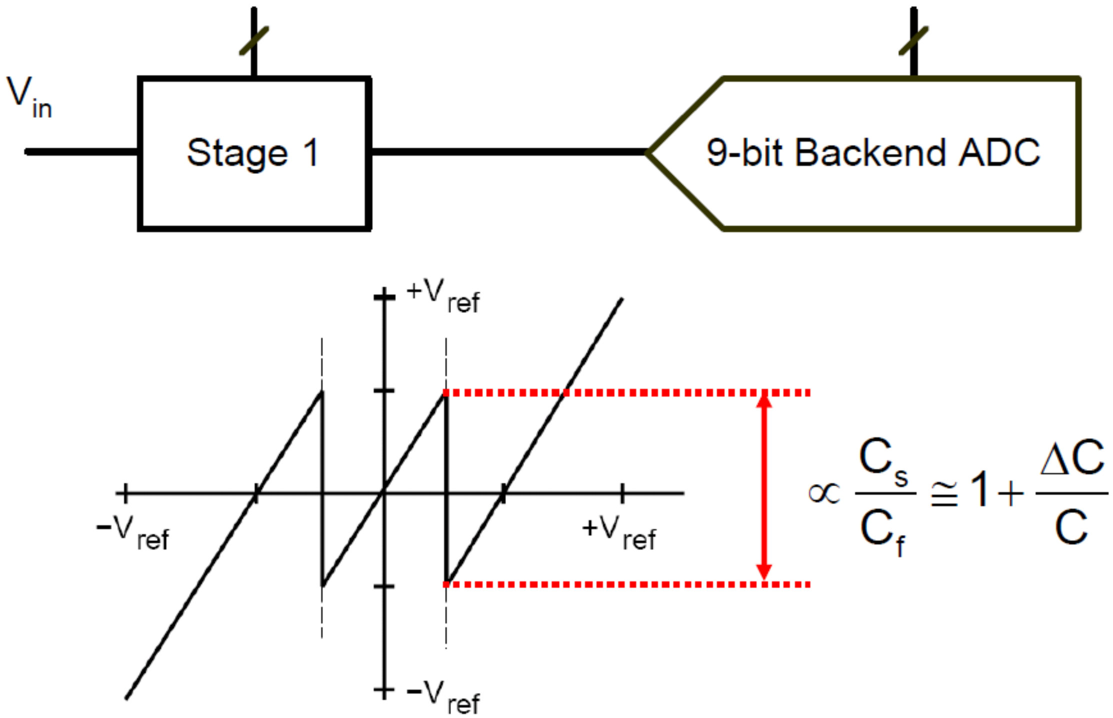
- Two-phase non-overlapping clock is typically used, with the coarse ADCs operating within the non-overlapping times
- All pipelined stages operate simultaneously, increasing throughput at the cost of latency (what is the latency of pipeline?)

1.5-Bit Decoding Scheme



b	0	1	2
b-1	-1	0	+1
C_2	$+V_R$	0	$-V_R$

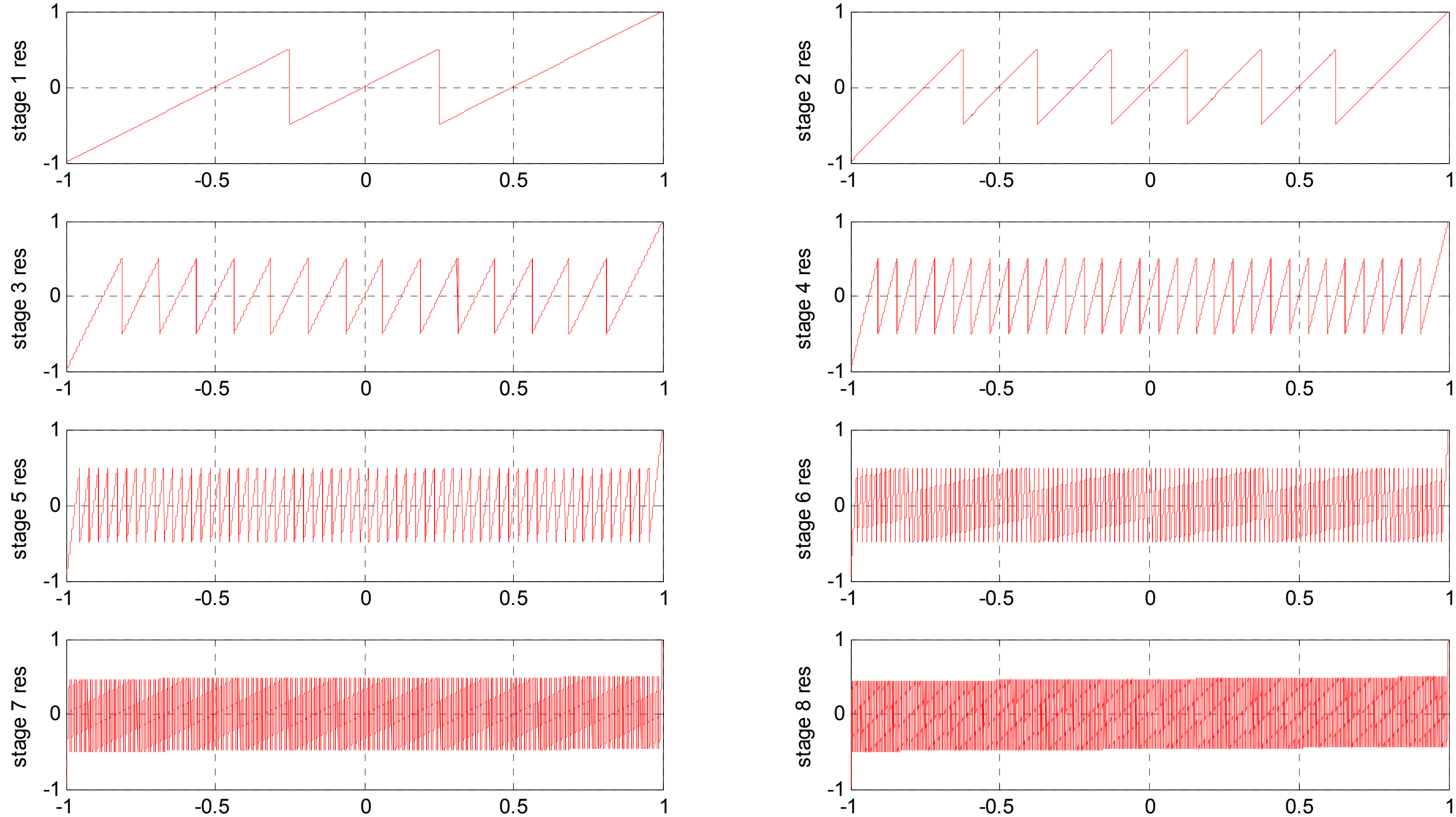
Stage 1 Matching Requirements



- Error in residue transition must be accurate to within a fraction of 9-bit backend LSB

– Typically want $\Delta C/C$ ~0.1% or better

1.5b/stage Residues

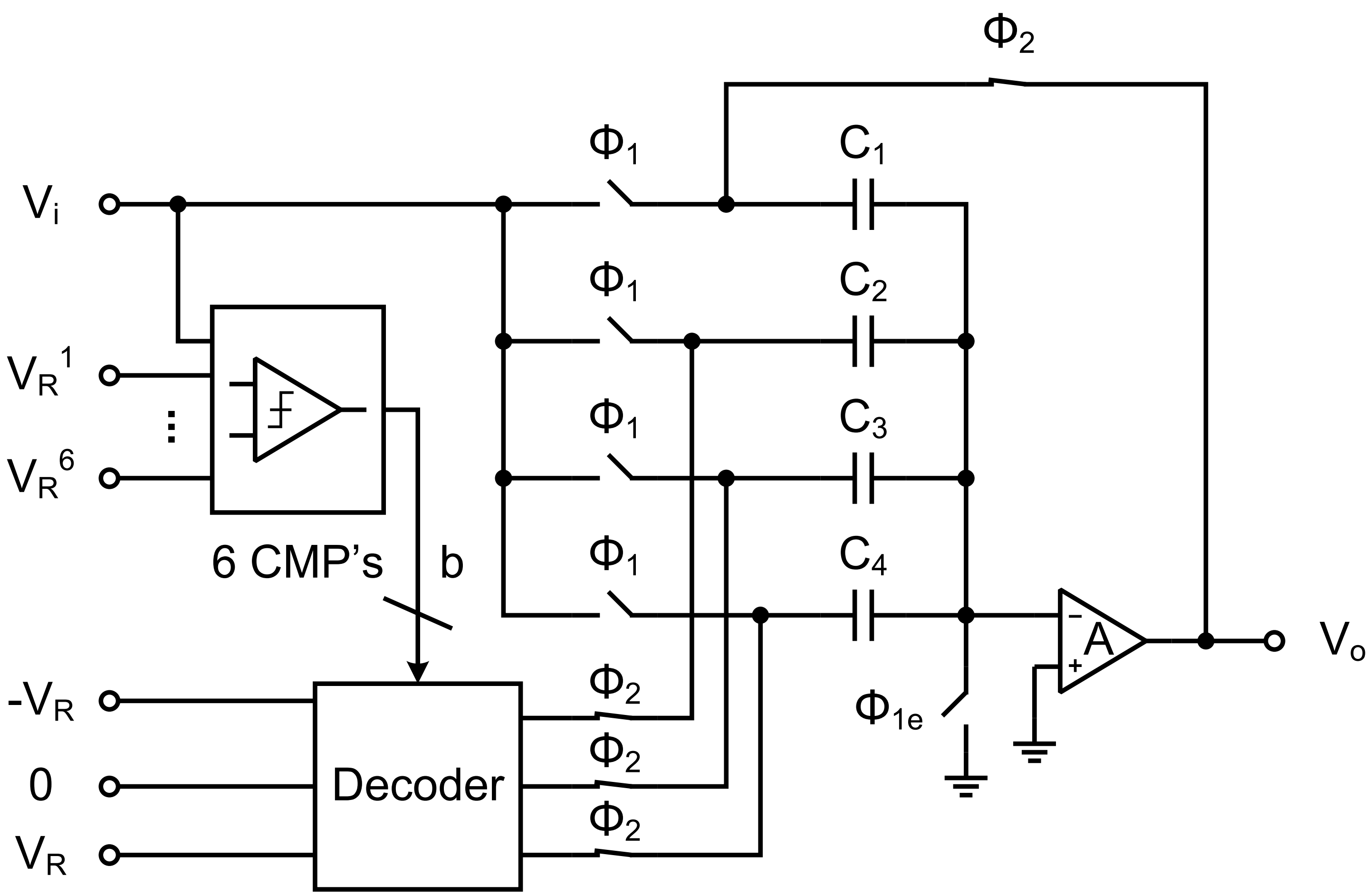


- Residues after every stage with ideal MDACs

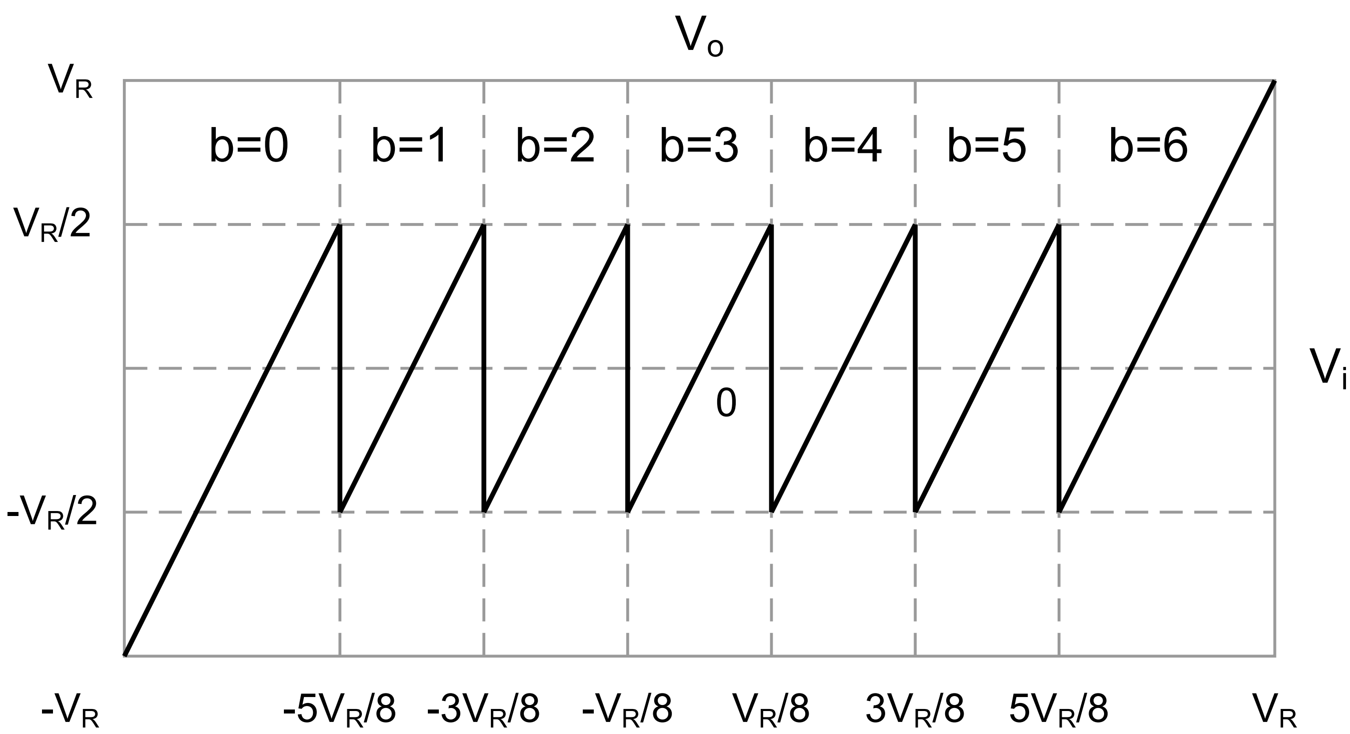
Capacitor Matching

- 0.1% "easily" achievable in current technologies
  - Even with metal sandwich caps, see e.g. [Verma 2006]
    - Beware of metal density related issues, "copper dishing"
  - For MIMCap matching data see e.g. [Diaz 2003]
- What if we needed much higher resolution than 10 bits?
  - Digital calibration
  - Multi-bit first stage
    - Each extra bit resolved in the first stage alleviates precision requirements on residue transition by 2x
    - For fixed capacitor matching, can show that each (effective) bit moved into the first stage
      - Improves DNL by 2x
      - Improves INL by sqrt(2)x
    - Multi-bit examples: [Singer 1996] [Kelly 2001] [Lee 2007]

A 2.5-Bit Stage



2.5-Bit RA Transfer Curve



- 6 comparators + 7-level DAC are required
- Max tolerance on comparator offset is $\pm V_R/8$

2.5-Bit Decoding Scheme

b	0	1	2	3	4	5	6
b-3	-3	-2	-1	0	+1	+2	+3
b_1	-1	-1	-1	0	+1	+1	+1
b_2	-1	-1	0	0	0	+1	+1
b_3	-1	0	0	0	0	0	+1
C_2	$+V_R$	$+V_R$	$+V_R$	0	$-V_R$	$-V_R$	$-V_R$
C_3	$+V_R$	$+V_R$	0	0	0	$-V_R$	$-V_R$
C_4	$+V_R$	0	0	0	0	0	$-V_R$

- 7-level DAC, 3×3×3 = 27 permutations of potential configurations → multiple choices of decoding schemes!
- Choose the scheme to minimize decoding effort, balance loading for reference lines, etc.

Design Parameters

- Stage resolution, stage scaling factor
- Stage redundancy
- Thermal noise/quantization noise ratio
- Opamp architecture
  - Opamp sharing?
- Switch topologies
- Comparator architecture
- Front-end SHA vs. SHA-less design
- Calibration approach (if needed)
- Time interleaving?
- Technology and technology options (e.g. capacitors)

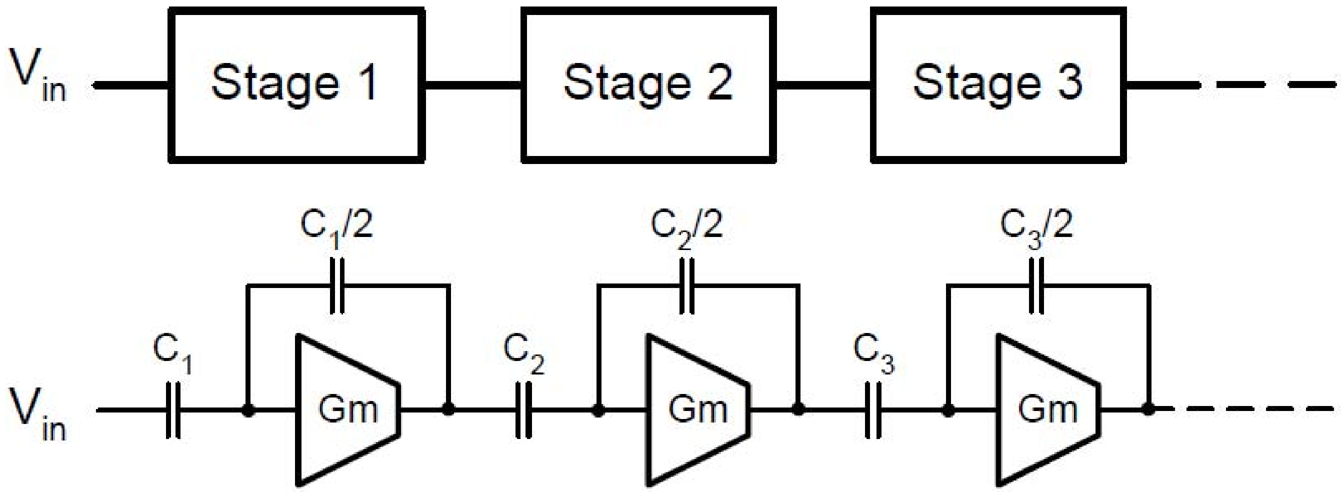
A very complex optimization problem!

Thermal Noise Considerations

- Total input referred noise
  - Thermal noise + Quantization noise
  - Costly to make input thermal noise smaller than quantization noise
- Example: VFS=1V, 10-bit ADC
  - $$E_q = \frac{V_{LSB}{}^2}{12} = \frac{1}{12}\left(\frac{1}{2^{10}}\right)^2 = \left(280\mu V_{rms}\right)^2$$
  - Design for total input referred thermal noise 280μV_{rms} or larger is SNR target allows
- Total input referred thermal noise of the ADC is the sum of thermal noise contribution from all stages
  - How should the thermal noise (kT/C) of the stages be distributed?

Stage Scaling

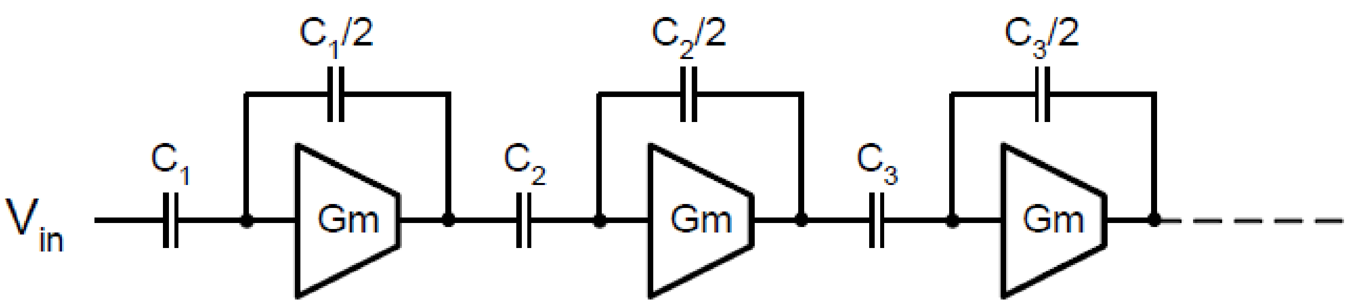
- Example: Pipeline using 1-bit (effective) stages (G=2)



- Total input referred noise power

$$N_{tot} \propto kT\left[\frac{1}{C_1} + \frac{1}{4C_2} + \frac{1}{16C_3} + \ldots\right]$$

Stage Scaling contd.

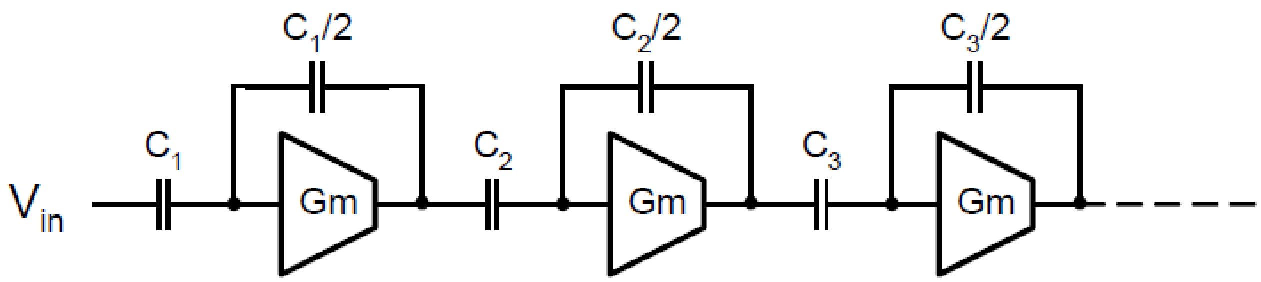


$$N_{tot} \propto kT\left[\frac{1}{C_1}+\frac{1}{4C_2}+\frac{1}{16C_3}+\ldots\right]$$

If we make all caps the same size, backend stages contribute very little noise

- Wasteful, because Power ~ Gm ~ C

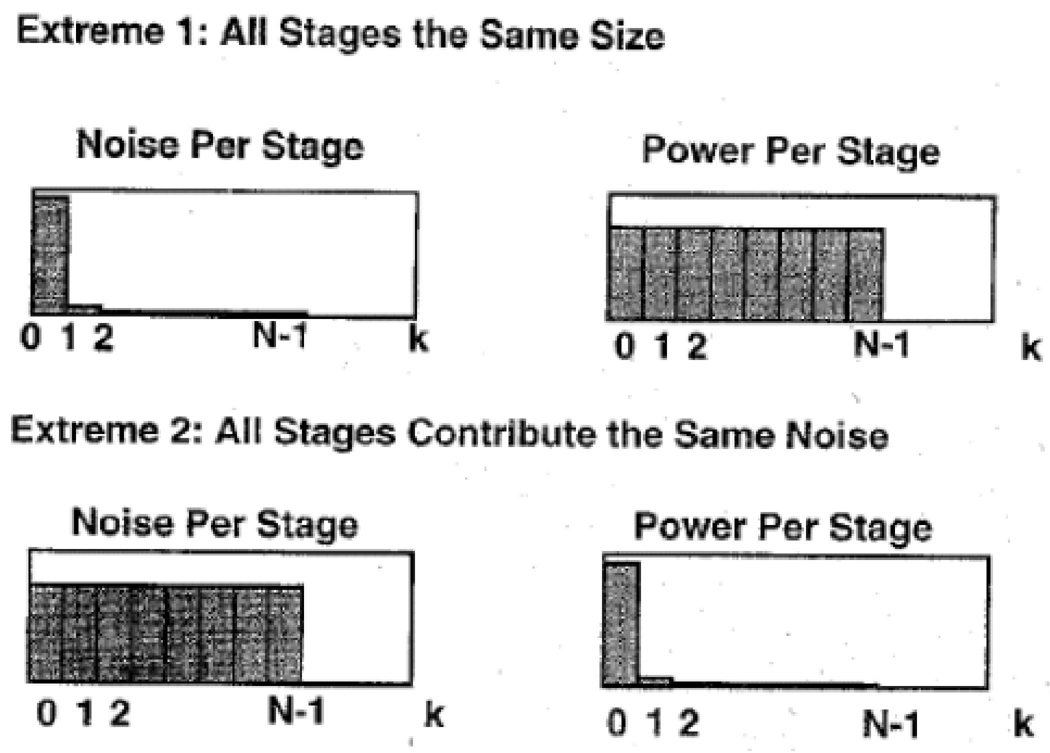
Stage Scaling contd.



$$N_{tot} \propto kT\left[\frac{1}{C_1}+\frac{1}{4C_2}+\frac{1}{16C_3}+\dots\right]$$

- How about scaling caps down by 2^M=4X every stage?
  - Same amount of noise from each stage
  - All stages contribute significant noise
  - Noise from the first stage must be reduced
  - Power ~Gm and C goes up!

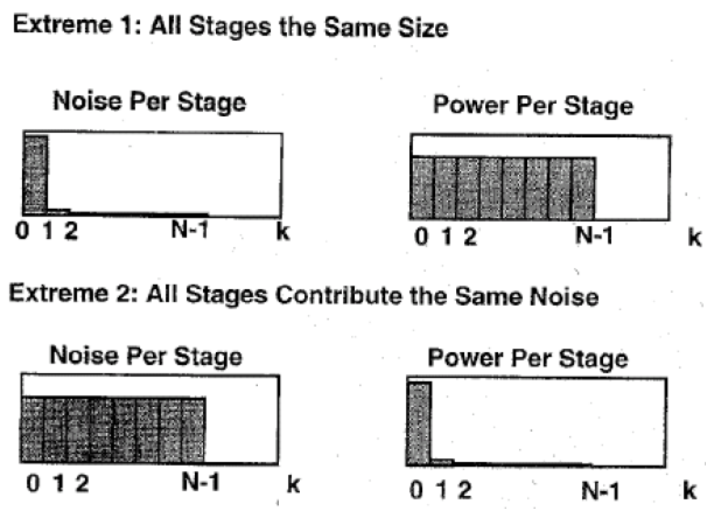
Stage Scaling contd.

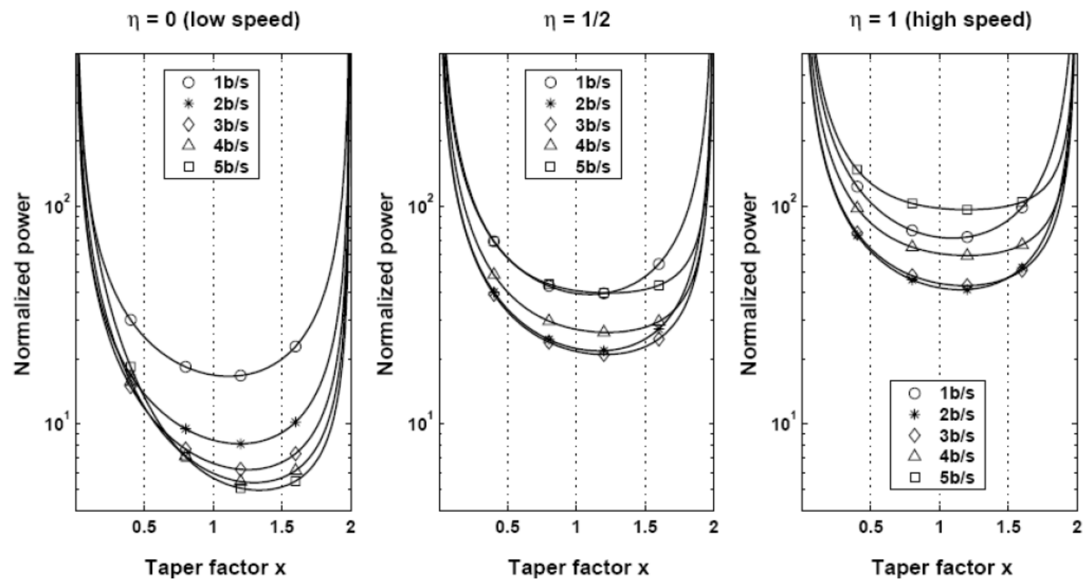


[Cline 1996]

•Optimum capacitior scaling lies approximately midway between these two extremes

Stage Scaling contd.

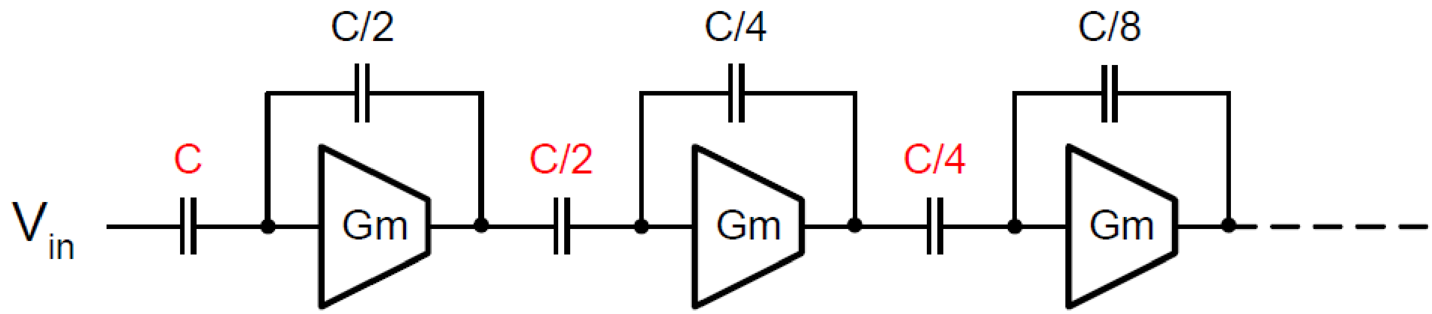




- •Optimum capacitor scaling lies approximately midway between these two extremes [Cline 1996]
- •Capacitor scaling factor 2^{RX}
  - •x=1 → scaling exactly by the stage gain [Chiu 2004]

Optimum Stage Scaling

•Start by assuming caps are scaled precisely by stage gain
– E.g. for 1-bit effective stages, caps are scaled by 2



•Refine using first pass circuit information & Excel spreadsheet

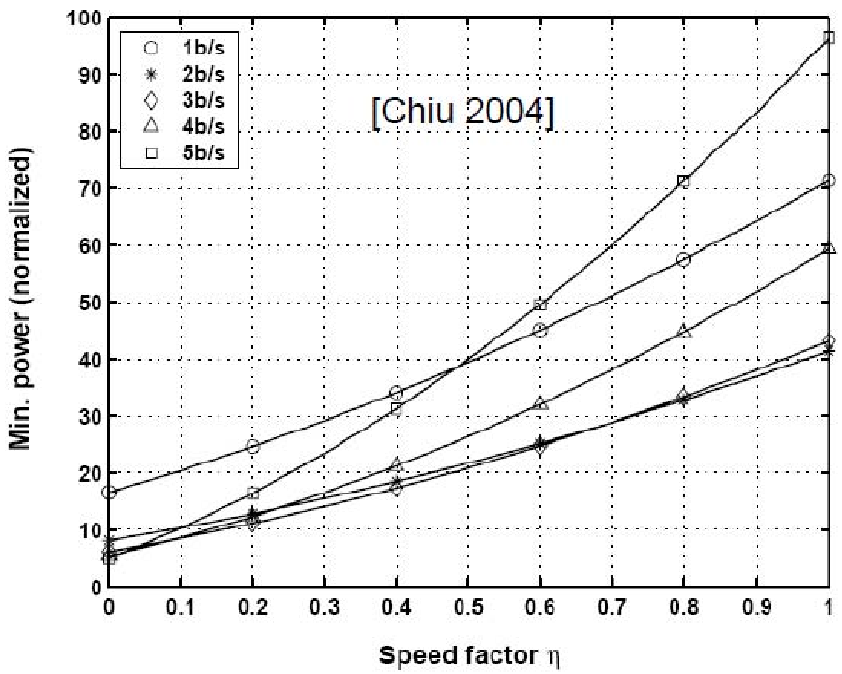
Use estimates of OTA power, parasitics, minimum feasible sampling capacitance etc.

Can develop optimization subroutines in MATLAB

How Many Bits per Stage?

- Low per-stage resolution (e.g. 1-bit effective)
  - – Need many stages
  - + OTAs have small closed loop gain, large feedback factor
    - High speed
- High per-stage resolution (e.g. 3-bit effective)
  - + Fewer stages
  - – OTAs can be power hungry, especially at high speed
  - – Significant loading from flash-ADC
- Qualitative conclusion
  - Use low per-stage resolution for very high speed designs
  - Try higher resolution stages when power efficiency is most important constraint

Power Tradeoff with Stage Resolution



η = parasitic cap at output/total sampling cap in each stage (junctions, wires, switches, …)

- Power tradeoff is nearly flat!
- ADC power varies only ~2X across different stage resolutions

Examples

Reference	[Yoshioka, 2007]	[Jeon, 2007]	[Loloee 2002]	[Bogner 2006]
Technology	90nm	90nm	0.18um	0.13um
Bits	10	10	12	14
Bits/Stage	1-1-1-1-1-1-1-3	2-2-2-4	1-1-1-1-1-1-1-1-1-1-2	3-3-2-2-4
SNDR [dB]	~56	~54	~65	~64
Speed [MS/s]	80	30	80	100
Power [mW]	13.3	4.7	260	224
mW/MS/s	0.17	0.16	3.25	2.24

- Low power is possible for a wide range of architectures!

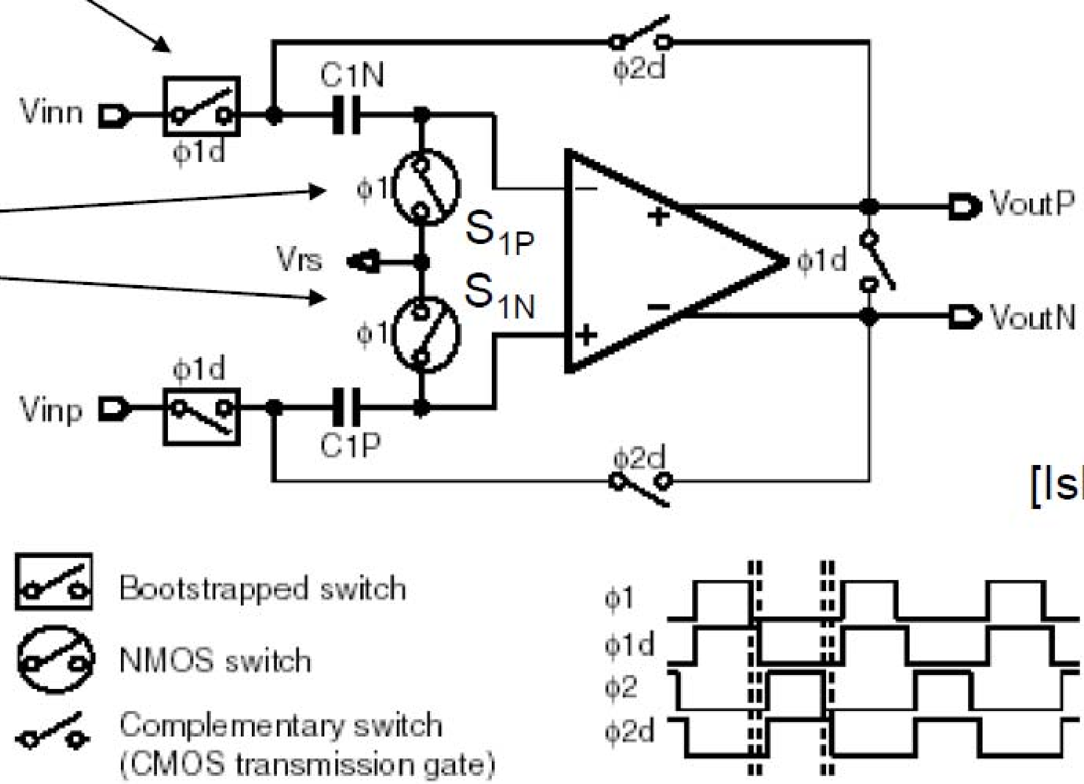
Cap Sizing Recap

- Choosing the "optimum" per-stage resolution and stage scaling scheme is a non-trivial task
  - –But – optima are shallow!
- Quality of transistor level design and optimization is at least as important (if not more important than) architectural optimization…
- Next, look at circuit design details
  - Assume we're trying to build a 10-bit pipeline
    - ~0.13um CMOS or smaller
    - Moderate to high-speed ~100MS/s
    - 1-bit effective/stage, using "1.5-bit" stage topology
    - Dedicated front-end SHA

Front-End SHA

Need constant R_{ON} here to minimize signal dependent charge injection from S_{1N}, S_{1P}

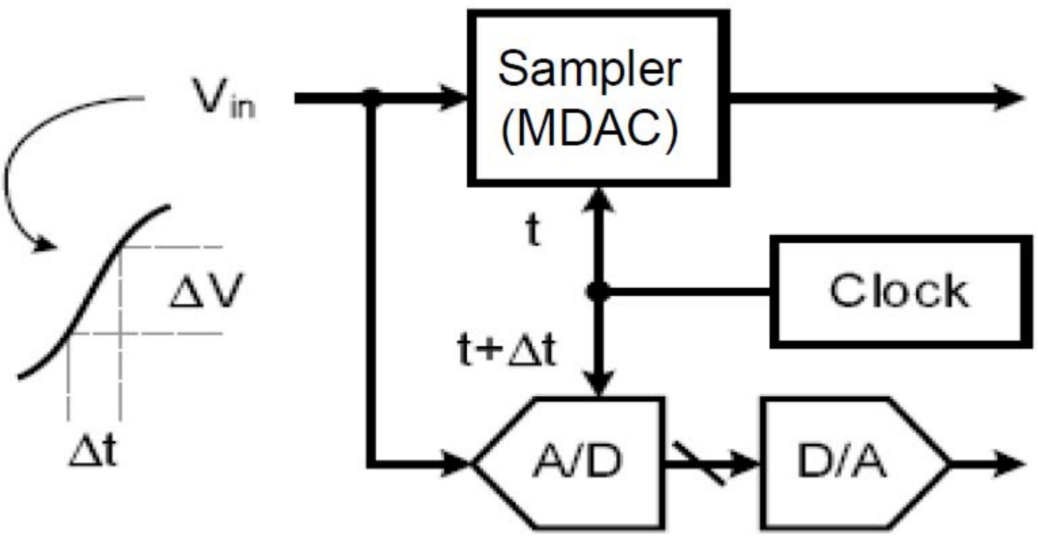
Minimize Jitter!



[Ishii 2005]

SHA-less Architectures

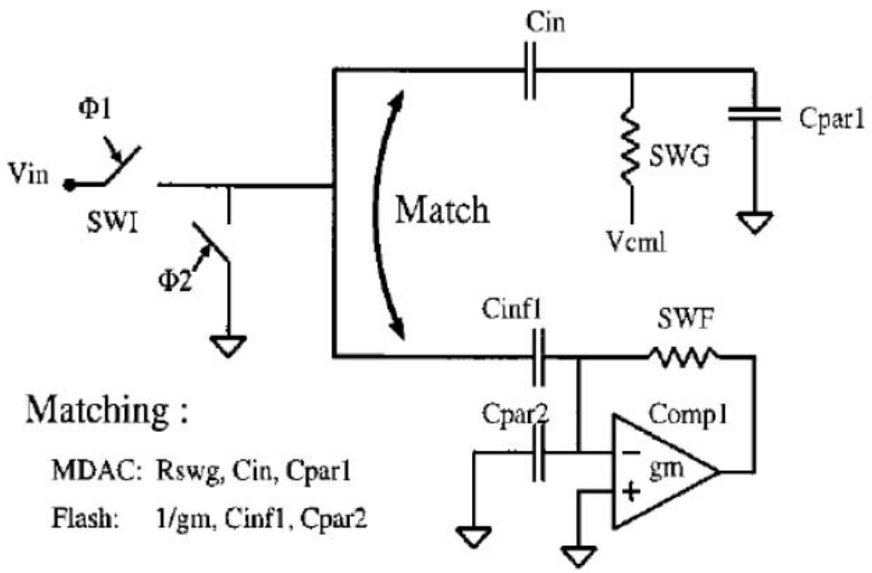
- Motivation
  - SHA can burn up to 1/3 of total ADC power
- Removing front-end SHA creates acquisition timing mismatch issue between first stage MDAC & Flash



[Chiu 2004]

SHA-less Architectures contd.

- Strategies
  - Use first stage with large redundancy; this can help absorb fairly large skew errors
  - Try to match sampling sub-ADC/MDAC networks
    - Bandwidth and clock timing

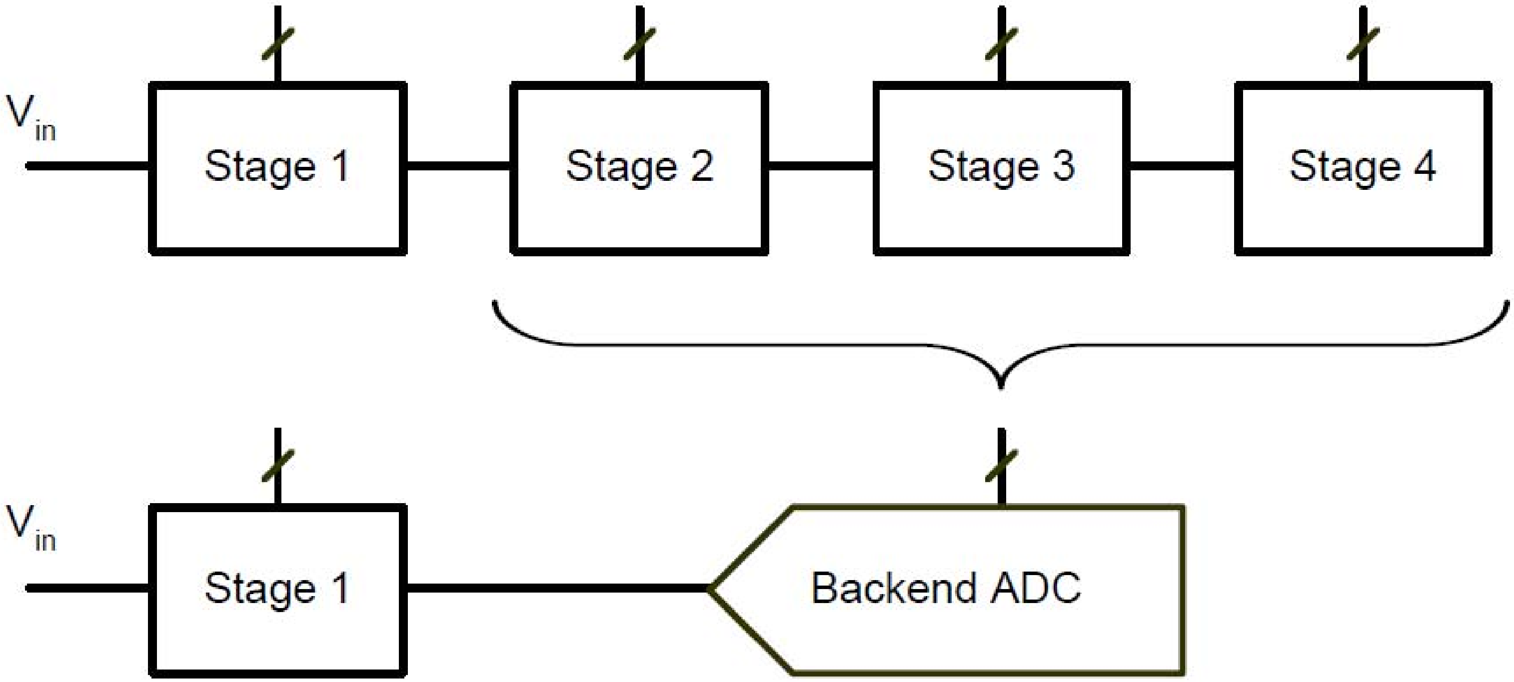


[Mehr 2000]

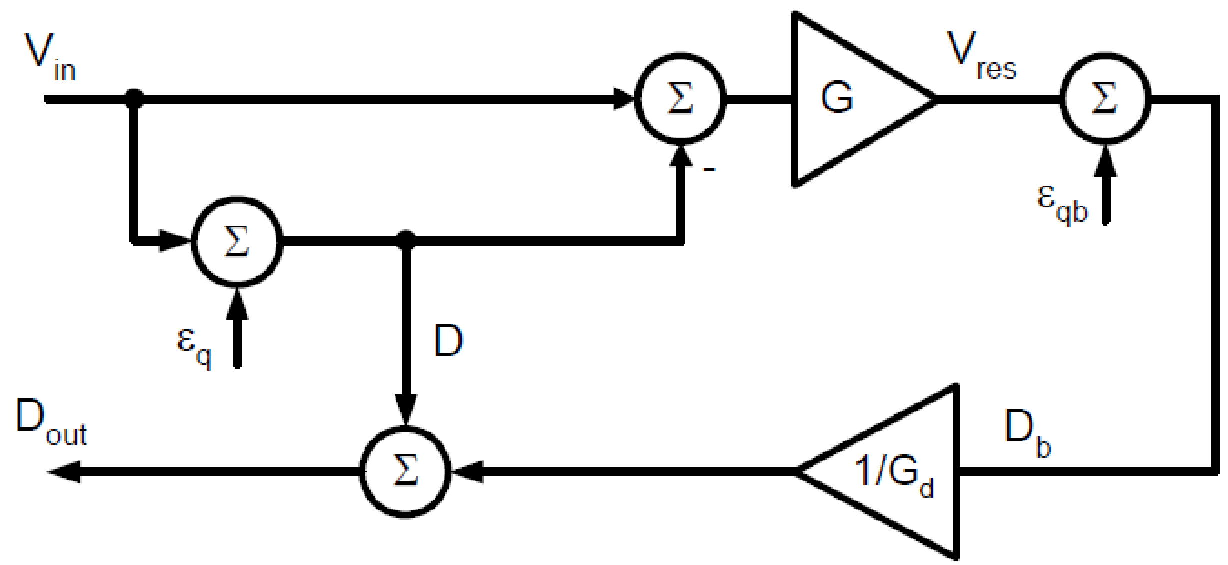
Pipelined A/D Conversion Errors

Pipeline Decomposition

- Often convenient to look at pipeline as single stage plus backend ADC



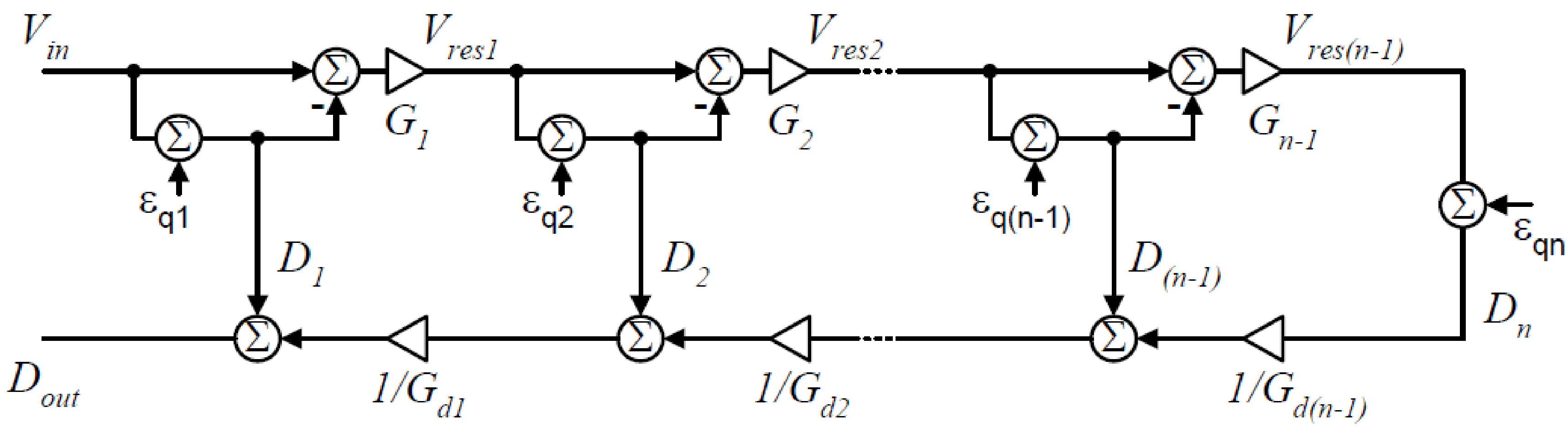
Resulting Model



$$D_{out} = V_{in} + \varepsilon_q \left(1 - \frac{G}{G_d}\right) + \frac{\varepsilon_{qb}}{G_d}$$

With G_d=G

Canonical Extension



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

- First stage has most stringent precision requirements
- Note that above model assumes that all stages use same reference voltage (same full scale range)
  - This is true for most designs, one exception is [Limotyrakis 2005]

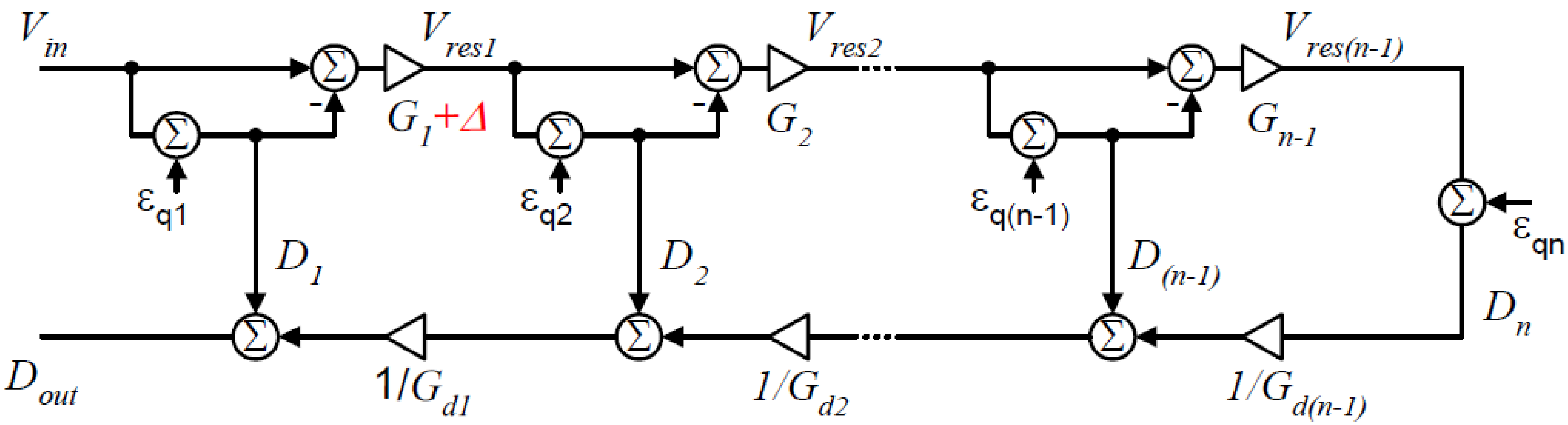
General Result – Ideal Pipeline ADC

- With ideal DACs and ideal digital weights ($G_{dj}=G_j$)

$$D_{out} = V_{in} + \frac{\varepsilon_{qn}}{\prod_{j=1}^{n-1} G_j} \quad \Rightarrow \quad B_{ADC} = B_n + \sum_{j=1}^{n-1} log_2 G_j$$

- The only error in D_{out} is that of last quantizer, divided by aggregate gain
- Aggregate ADC resolution is independent of sub-ADC resolutions in stage 1...n-1 (!)
- Makes sense to define "effective" resolution of j^{th} stage as $R_j=\log_2(G_j)$

Gain Errors



$$D_{out} = V_{in} + \varepsilon_{q1}\left(1 - \frac{G_1 + \Delta}{G_{d1}}\right) + \ldots + \frac{\varepsilon_{qn}}{\prod_{j=1}^{n-1} G_{dj}}$$

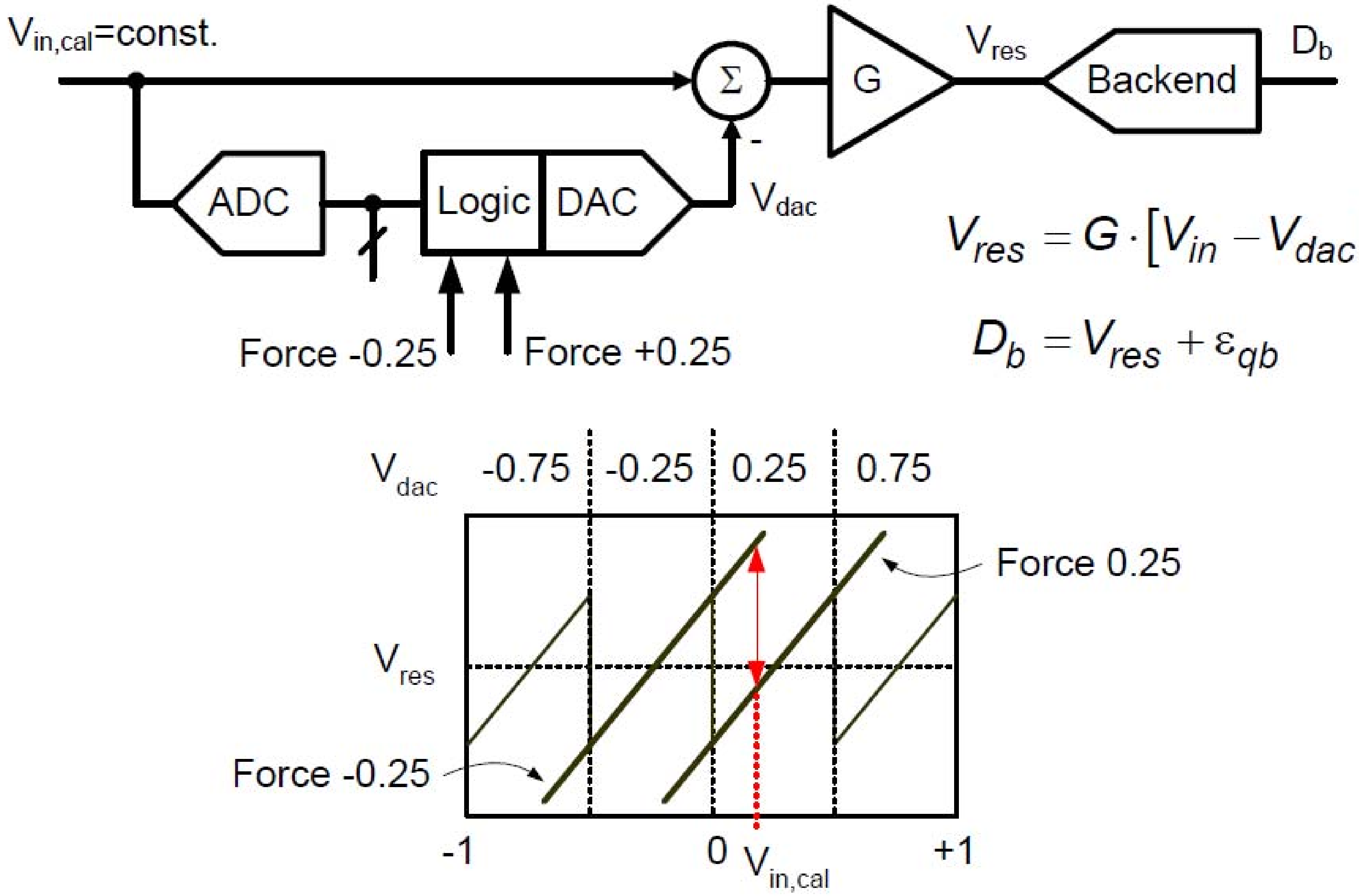
- Want to make $G_{d1} = G_1 + \Delta$

Digital Gain Calibration (1)

- Error in analog gain is not a problem as long as "digital gain term" is adjusted appropriately
- Problem
  - Need to measure analog gain precisely
- Example
  - Digital calibration of a 1-bit first stage with 1-bit redundancy (R=1, B=2)

- Note
  - Even if all G_{dj} are perfectly adjusted to reflect the analog gains, the ADC will have non-zero DNL and INL, bounded by ±0.5LSB. This can be explained by the fact that the residue transitions may not correspond to integer multiples of the backend-LSB. This can cause non-uniformity in the ADC transfer function (DNL, INL) and also non-monotonicity (see [Markus, 2005]).
  - In case this cannot be tolerated
    - Add redundant bits to ADC backend (after combining all bits, final result can be truncated back)
    - Calibrate analog gain terms

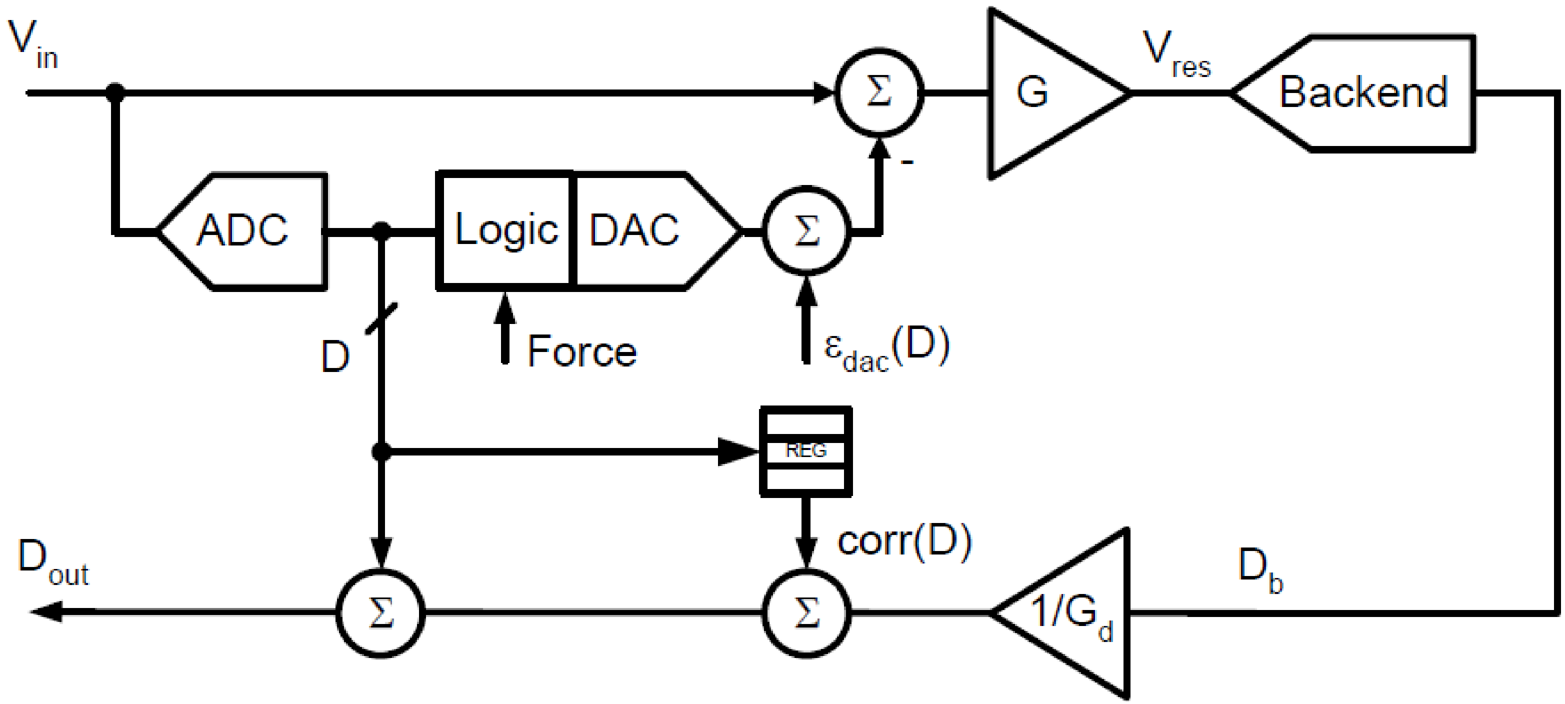
Digital Gain Calibration (2)



$$V_{res} = G \cdot [V_{in} - V_{dac}]$$

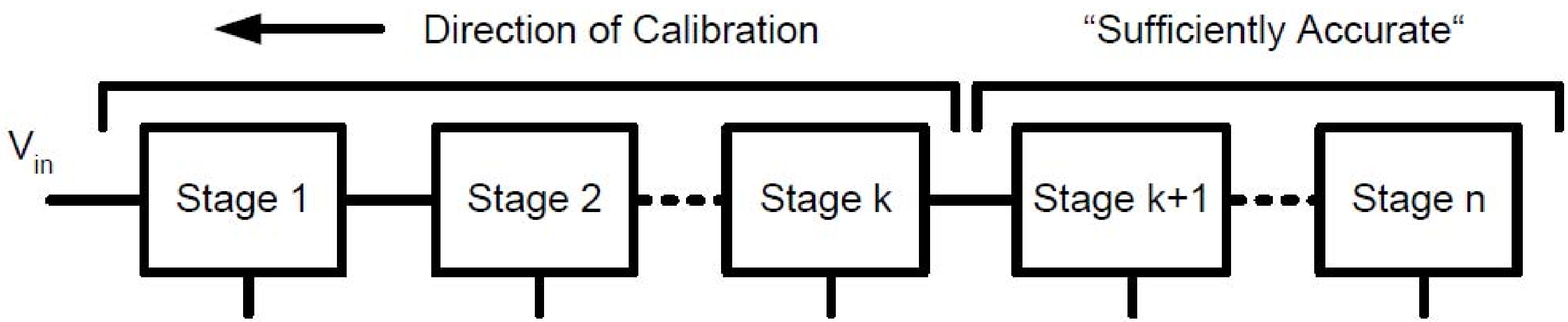
$$D_b = V_{res} + \varepsilon_{qb}$$

Digital Gain Calibration (3)



- Essentially same concept as gain calibration
  - Step through DAC codes and use backend to measure errors
- Store coefficients for each DAC transition in a look-up table

Recursive Stage Calibration



- First few stages have most stringent accuracy requirements
  - Errors of later stages are attenuated by aggregate gain
- Commonly used algorithm [Karanicolas 1993]
  - Take ADC offline
  - Measure least significant stage that needs calibration first
  - Move to next significant stage and continue toward stage 1

References

1. Rudy van de Plassche, “CMOS Integrated Analog-to-Digital and Digital-to-Analog Converters,” 2nd Ed., Springer, 2005.
2. M. Gustavsson, J. Wikner, N. Tan, *CMOS Data Converters for Communications*, Kluwer Academic Publishers, 2000.
3. N. Krishnapura, *Pipelined Analog to Digital Converters Slides*, IIT Madras, 2009.
4. Y. Chiu, *Data Converters Lecture Slides,* UT Dallas 2012*.*
5. B. Boser, *Analog-Digital Interface Circuits Lecture Slides,* UC Berkeley 2011*.*
6. K. Nagaraj et al., “A 250-mW, 8-b, 52-Msamples/s parallel-pipelined A/D converter with reduced number of amplifiers”, *IEEE Journal of Solid-State Circuits*, pp. 312-320, vol. 32, no. 3, March 1997.
7. S. Kulhalli et al., “A 30mW 12b 21MSample/s pipelined CMOS ADC”, *2002 IEEE International Solid State Conference*, pp. 18.4, vol. I, pp. 248-249,492, vol. II.
8. N. Sasidhar et al., “A Low Power Pipelined ADC Using Capacitor and Opamp Sharing Technique With a Scheme to Cancel the Effect of Signal Dependent Kickback”, *IEEE Journal of Solid State Conference*, pp. 2392-2401, vol. 44, no. 9, Sep 2009.

References

- Implementation

  - T. Cho, "Low-Power Low-Voltage Analog-to-Digital Conversion Techniques using Pipelined Architecures, PhD Dissertation, UC Berkeley, 1995, http://kabuki.eecs.berkeley.edu/~tcho/Thesis1.pdf.
  - L. A. Singer at al., "A 14-bit 10-MHz calibration-free CMOS pipelined A/D converter," VLSI Circuit Symposium, pp. 94-95, Jun. 1996.
  - A. Abo, "Design for Reliability of Low-voltage, Switched-capacitor Circuits," PhD Dissertation, UC Berkeley, 1999, http://kabuki.eecs.berkeley.edu/~abo/abothesis.pdf.
  - D. Kelly et al., "A 3V 340mW 14b 75MSPS CMOS ADC with 85dB SFDR at Nyquist," ISSCC Dig. Techn. Papers, pp. 134-135, Feb. 2001.
  - A. Loloee, et. al, "A 12b 80-MSs Pipelined ADC Core with 190 mW Consumption from 3 V in 0.18-um, Digital CMOS", Proc. ESSCIRC, pp. 467-469, 2002
  - B.-M. Min et al., "A 69-mW 10-bit 80-MSample/s Pipelined CMOS ADC," IEEE JSSC, pp. 2031-2039, Dec. 2003.
  - Y. Chiu, et al., "A 14-b 12-MS/s CMOS pipeline ADC with over 100-dB SFDR," IEEE JSSC, pp. 2139-2151, Dec. 2004.
  - S. Limotyrakis et al., "A 150-MS/s 8-b 71-mW CMOS time-interleaved ADC," IEEE JSSC, pp. 1057-1067, May 2005.
  - T. N. Andersen et al., "A Cost-Efficient High-Speed 12-bit Pipeline ADC in 0.18-um Digital CMOS," IEEE JSSC, pp. 1506-1513, Jul 2005.

References

- P. Bogner et al., "A 14b 100MS/s digitally self-calibrated pipelined ADC in 0.13um CMOS," ISSCC Dig. Techn. Papers, pp. 832-833, Feb. 2006.
- D. Kurose et al., "55-mW 200-MSPS 10-bit Pipeline ADCs for Wireless Receivers," IEEE JSSC, pp. 1589-1595, Jul. 2006.
- S. Bardsley et al., "A 100-dB SFDR 80-MSPS 14-Bit 0.35um BiCMOS Pipeline ADC," IEEE JSSC, pp. 2144-2153, Sep. 2006.
- A. M. A. Ali et al, "A 14-bit 125 MS/s IF/RF Sampling Pipelined ADC With 100 dB SFDR and 50 fs Jitter," IEEE JSSC, pp. 1846-1855, Aug. 2006.
- S. K. Gupta, "A 1-GS/s 11-bit ADC With 55-dB SNDR, 250-mW Power Realized by a High Bandwidth Scalable Time-Interleaved Architecture," IEEE JSSC, 2650-2657, Dec. 2006.
- M. Yoshioka et al., "A 0.8V 10b 80MS/s 6.5mW Pipelined ADC with Regulated Overdrive Voltage Biasing," ISSCC Dig. Techn. Papers, pp. 452-453, Feb. 2007.
- Y.-D. Jeon et al., "A 4.7mW 0.32mm2 10b 30MS/s Pipelined ADC Without a Front-End S/H in 90nm CMOS," ISSCC Dig. Techn. Papers, pp. 456-457, Feb. 2007.
- K.-H. Lee et al., "Calibration-free 14b 70MS/s 0.13um CMOS pipeline A/D converters based on highmatching 3D symmetric capacitors," Electronics Letters, pp. 35-36, Mar. 15, 2007.
- K. Honda et al., "A Low-Power Low-Voltage 10-bit 100-MSample/s Pipeline A/D Converter Using Capacitance Coupling Techniques," IEEE JSSC, pp. 757-765, Apr. 2007.

References

- Per-Stage Resolution and Stage Scaling

  - D. W. Cline et al., "A power optimized 13-b 5 MSamples/s pipelined analog-to-digital converter in 1.2um CMOS," IEEE JSSC, Mar. 1996
  - Y. Chiu, "High-Performance Pipeline A/D Converter Design in Deep-Submicron CMOS," PhD Dissertation, UC Berkeley, 2004.
  - H. Ishii et al., "A 1.0 V 40mW 10b 100MS/s pipeline ADC in 90nm CMOS," Proc. CICC, pp. 395-398, Sep. 2005.

- OTA Design, Noise

  - B. E. Boser, "Analog Circuit Design with Submicron Transistors," Presentation at IEEE Santa Clara Valley, May 19, 2005, http://www.ewh.ieee.org/r6/scv/ssc/May1905.htm
  - B. Murmann, EE315A Course Material, http://ccnet.stanford.edu/cgi-bin/handouts.cgi?cc=ee315a
  - R. Schreier et al., "Design-oriented estimation of thermal noise in switched-capacitor circuits," IEEE TCAS I, pp. 2358-2368, Nov. 2005.

- Capacitor Matching Data

  - C. H. Diaz et al., "CMOS technology for MS/RF SoC," IEEE Trans. Electron Devices, pp. 557-566, Mar. 2003.
  - A. Verma et al., "Frequency-Based Measurement of Mismatches Between Small Capacitors," Proc. CICC, pp. 481-484, Sep. 2006.

- Reference Generator

  - T. L. Brooks et al., "A low-power differential CMOS bandgap reference," ISSCC Dig. Techn. Papers*, pp. 248-249, Feb. 1994.*

References

- Digitally-Assisted Pipelined

  - B. Murmann et al., "A 12-bit 75-MS/s Pipelined ADC using Open-Loop Residue Amplification," IEEE JSSC, pp. 2040-2050, Dec. 2003.
  - J. Fiorenza et al., "Comparator-Based Switched-Capacitor Circuits for Scaled CMOS Technologies, " IEEE JSSC, pp. 2658-2668, Dec. 2006.
  - E. Iroaga et al. "A 12b, 75MS/s Pipelined ADC Using Incomplete Settling," IEEE JSSC, pp. 748-756, Apr. 2007.
  - J. Hu, N. Dolev and B. Murmann, "A 9.4-bit, 50-MS/s, 1.44-mW Pipelined ADC using Dynamic Residue Amplification,“ VLSI Circuits Symposium, June 2008.