High Speed Pipelined ADCs: Fundamentals and Variants

Ahmed M. A. Ali
Analog Devices

2018

Outline

- Introduction
- Pipelined ADCs
- Design challenges
  - Quantizer linearity
  - Noise
- Digitally assisted converters

INTRODUCTION

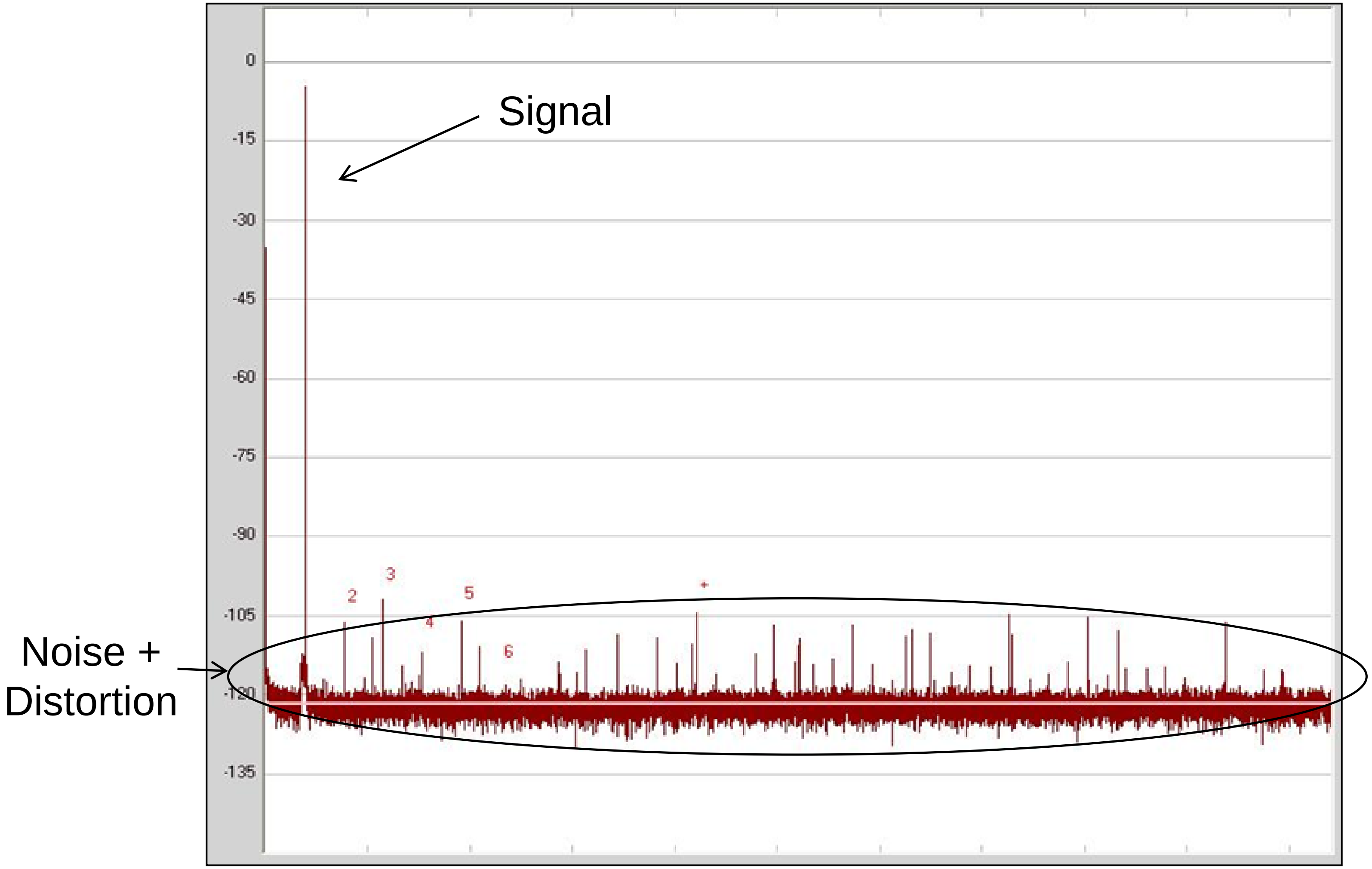
ADC Performance Metrics
Some of the Most Commonly Used

- Sampling Rate
- Resolution
- Signal-to-Noise Ratio (SNR)
- Signal-to-Noise-and-Distortion Ratio (SNDR or SINAD)
- Spurious-Free Dynamic Range (SFDR)
- Inter-modulation Distortion (IMD)
- Integral Non-linearity (INL)
- Differential Non-linearity (DNL)
- Offset and Gain Error
- Bit-Error-Rate (BER)
- Power
- Jitter

Signal-to-Noise Ratio (SNR)
Signal-to-Noise-and-Distortion Ratio (SNDR or SINAD)

- SNR/SINAD are defined as the ratio of the signal power to the noise power excluding DC:
  - SNR = 10log(Signal Power / Noise Power excluding harmonics)
  - SINAD = 10log(Signal Power / (Noise + Harmonic Power))
- They can be specified in dBc or dBFS
- Usually specified using a single tone input signal
- Sources:
  - Quantization noise and distortion
  - Thermal noise in the signal path
  - Noise and distortion in the clock path (Jitter)
  - Non-linearity in the signal path
- Effective Number of Bits (ENOB) = (SINAD-1.76dB)/6.02
- Effective Resolution = Full scale/Input referred noise in LSBs

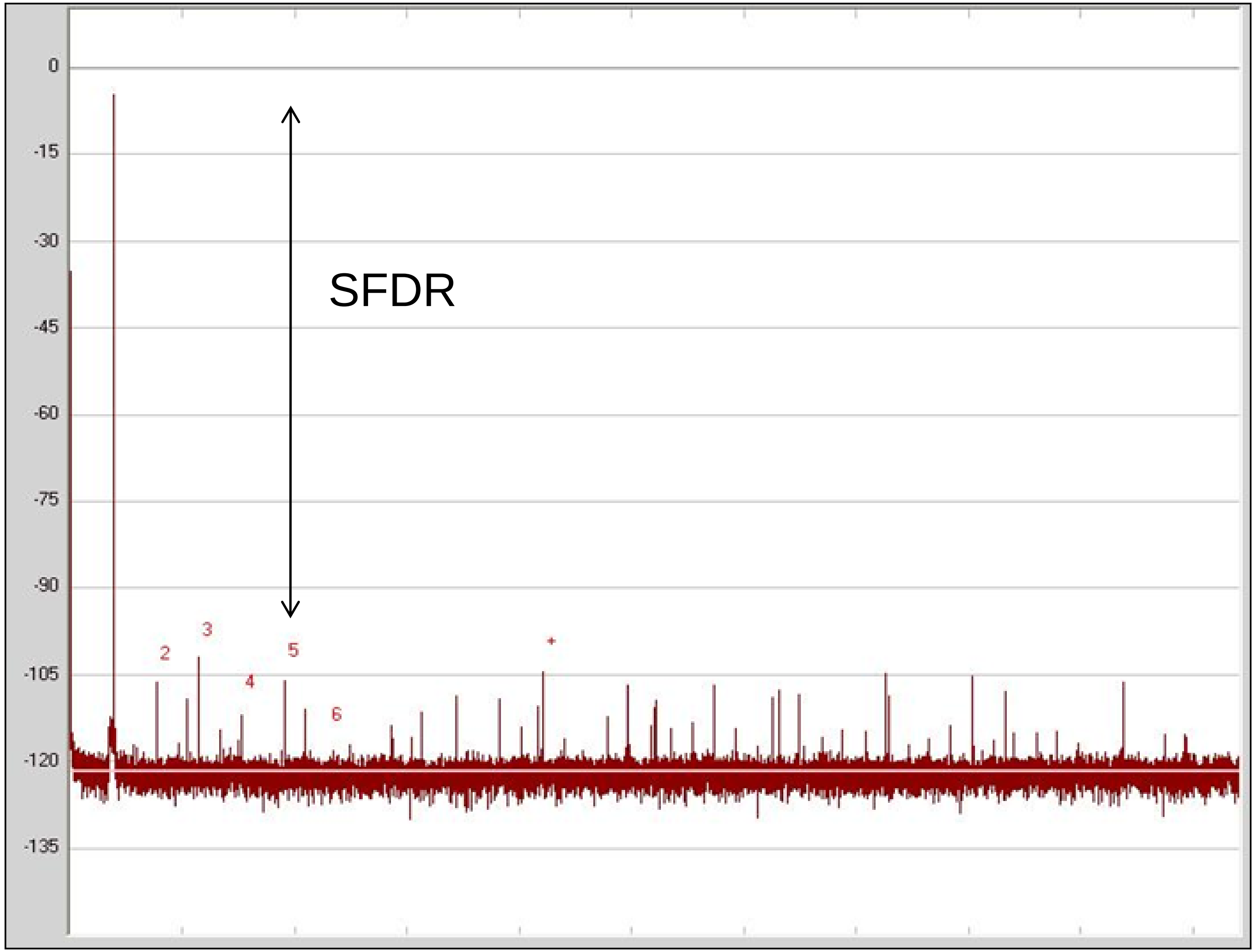
Example: AD9467 (AD80264) at 250MS/s
SNR = 74dB, SINAD = 74dB



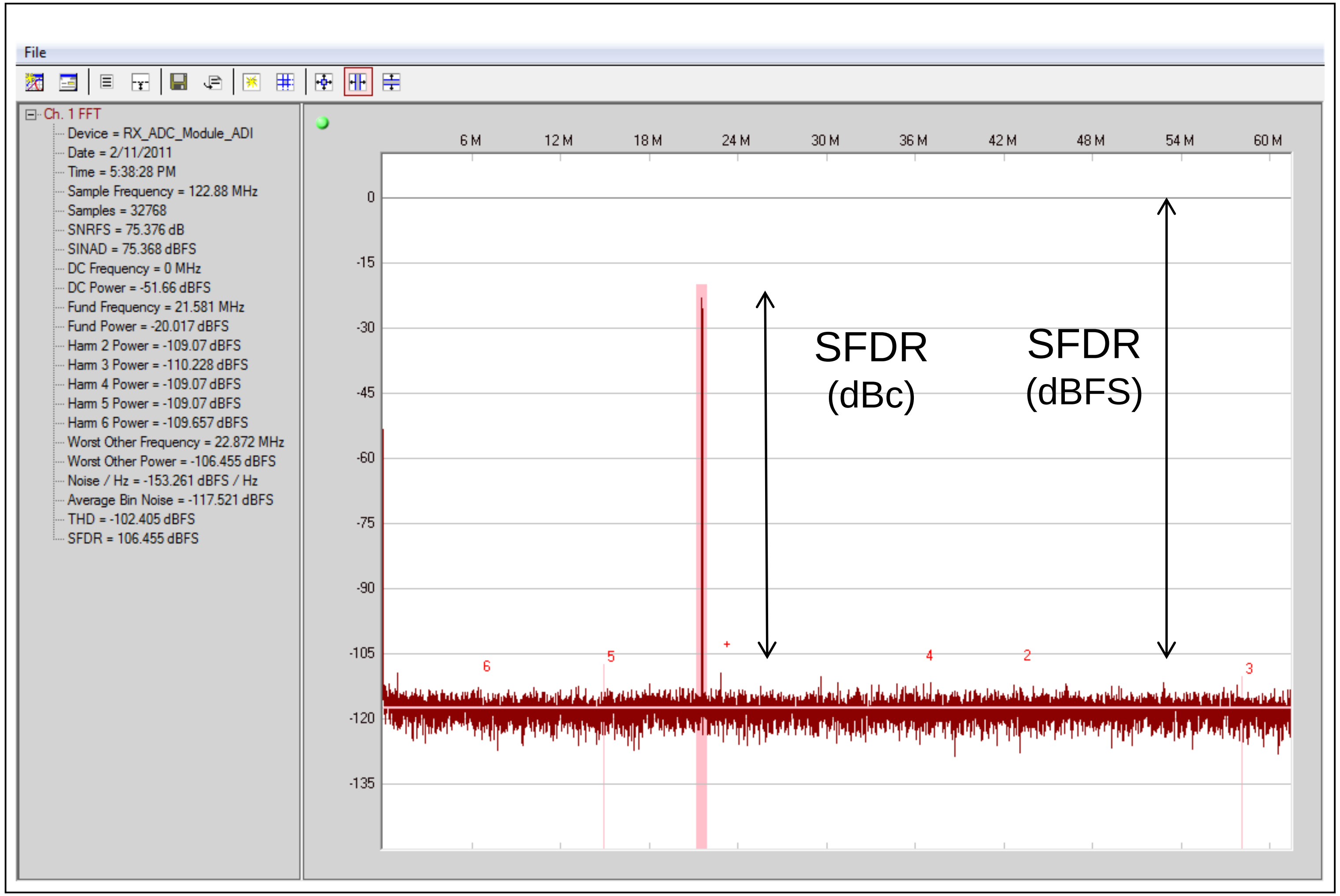
Spurious-Free Dynamic Range (SFDR)

- It is the ratio of the signal power to the power of the largest undesired harmonic or spur
- SFDR can be defined in dBc or dBFS
- Usually specified using a single tone input signal
- It is usually evaluated across the input amplitude
- THD is the ratio of the signal power to the total harmonic power of the significant harmonics

Example: AD9467 at 250MS/s
SFDR = 100 dBc

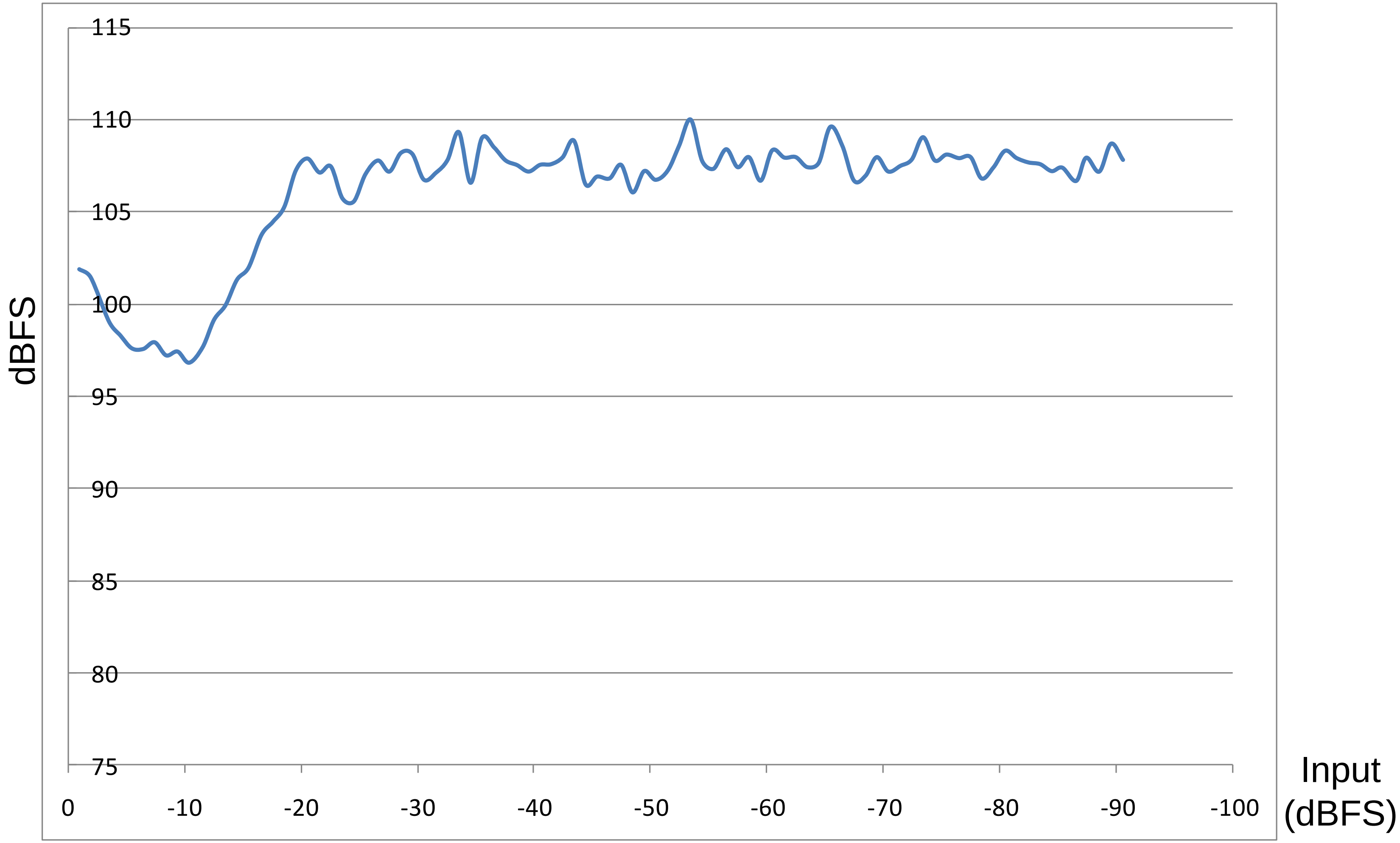


Example: AD9467 at 122.88 MS/s
At -20 dBFS, SFDR = 106 dBFS = 86dBc



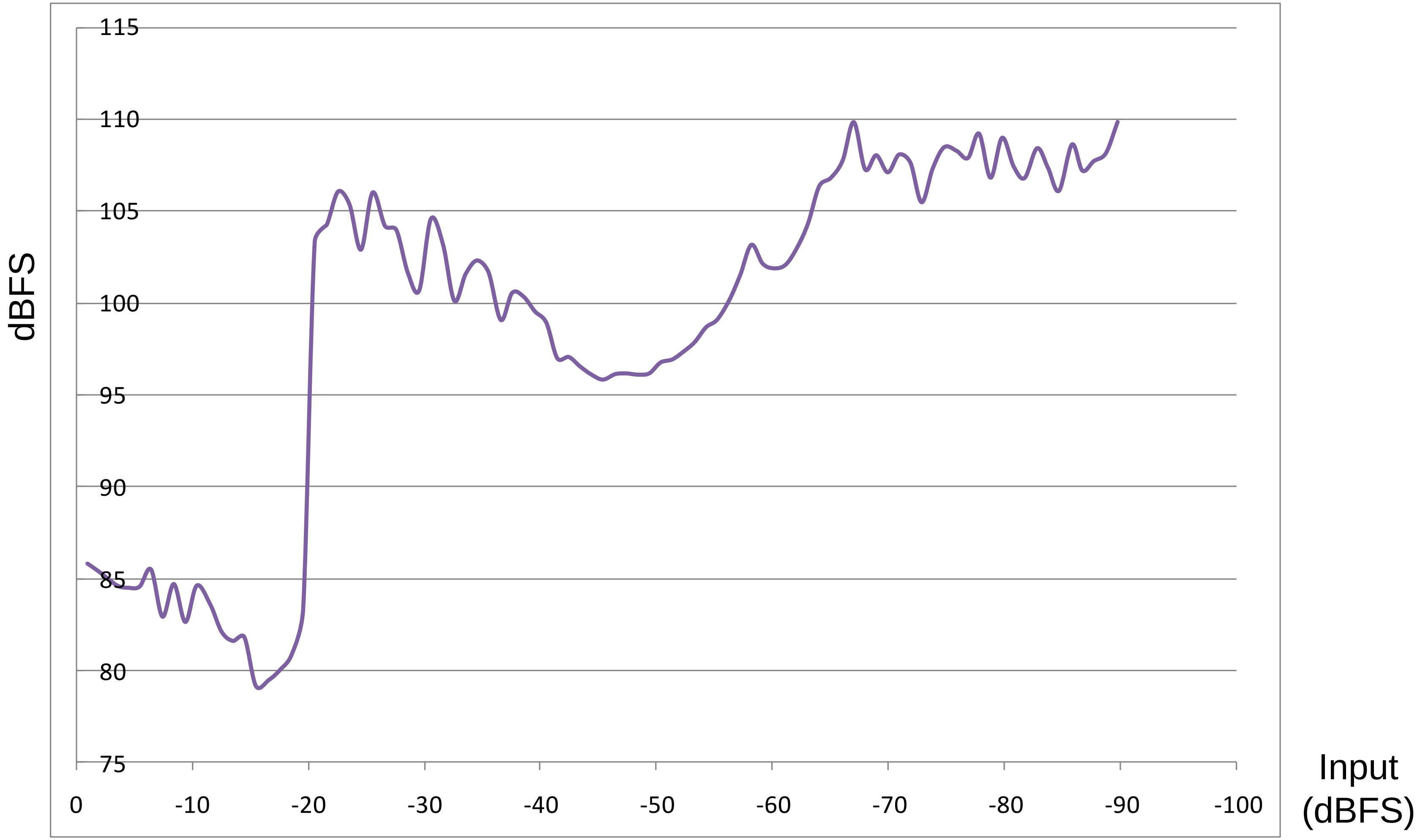
Ahmed Ali

SFDR versus Input Amplitude

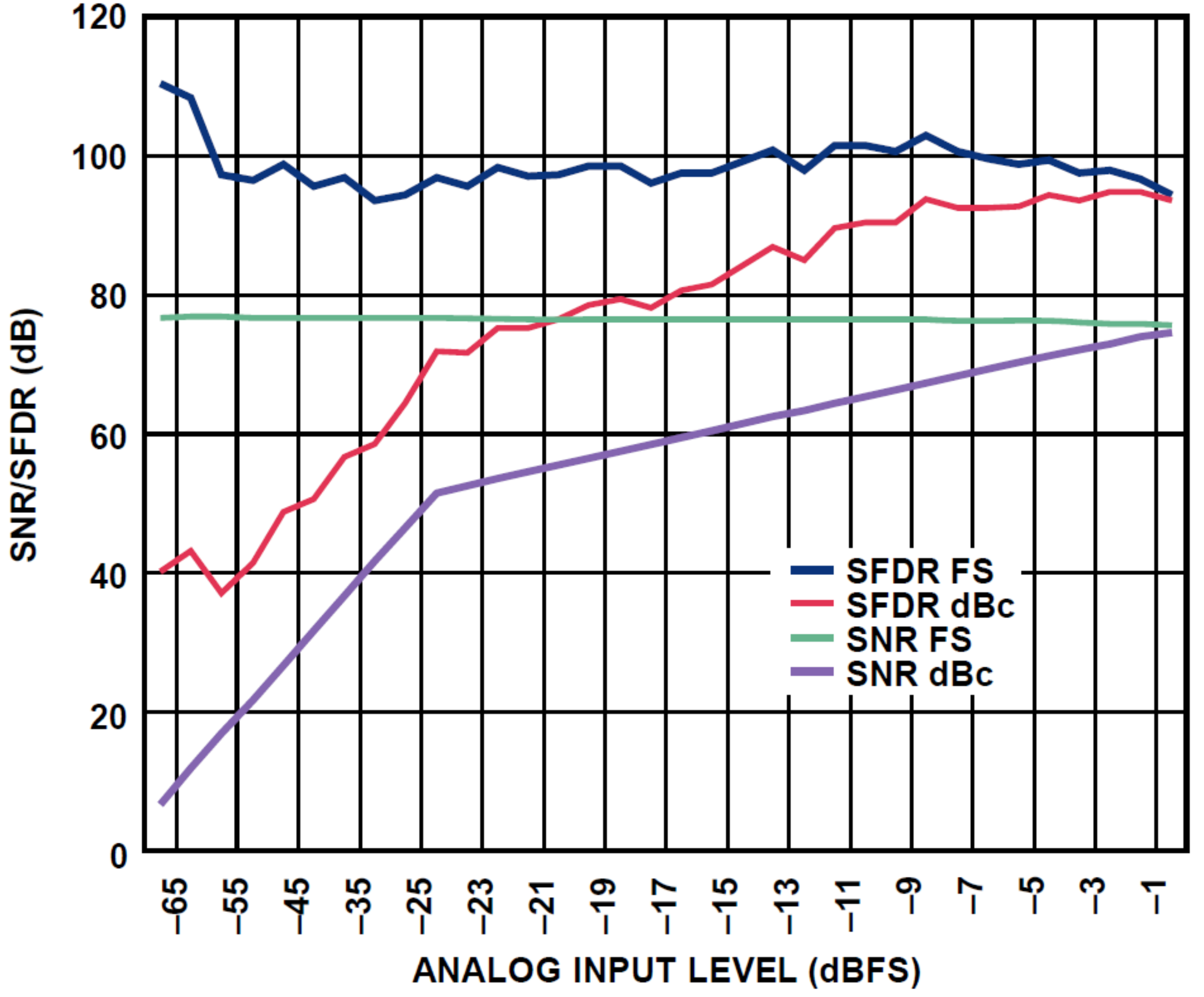


Ahmed Ali

SFDR versus Input Amplitude Example



SFDR versus Input Amplitude (Example)

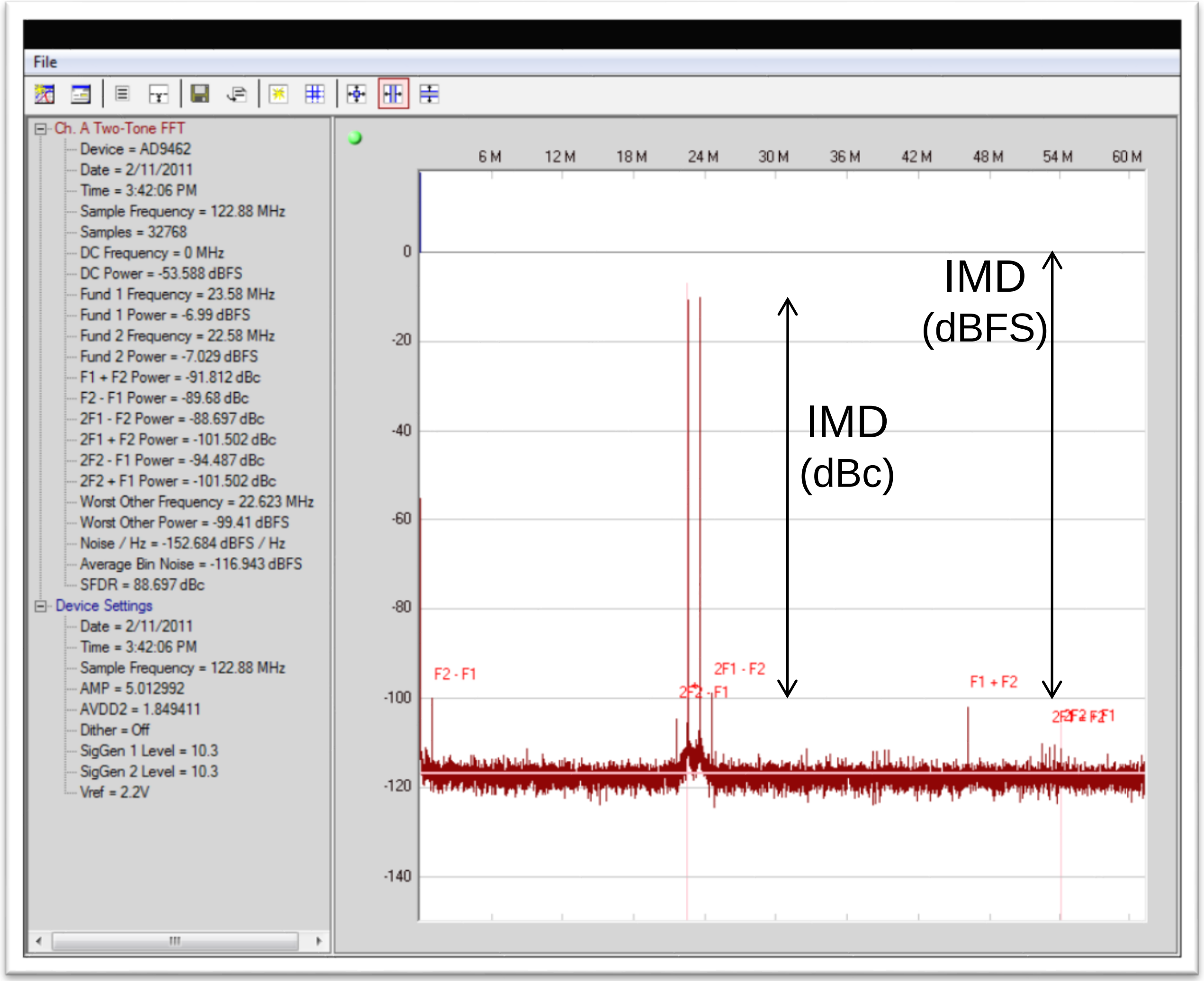


Ahmed Ali

Inter-Modulation Distortion (IMD)

- It is tested using two input tones equal in amplitude and closely spaced in frequency (f1 and f2)
- It is the ratio of the power of one of the input tones to the power of the largest undesired spur
- It is useful in some systems to evaluate distortion without the effects of filtering or frequency planning
- The second order products (IM2) are at:
  - (f1+f2) and (f1-f2)
- The third order products (IM3) are at:
  - (2f1+f2) and (2f2+f1)
  - (2f1-f2) and (2f2-f1)

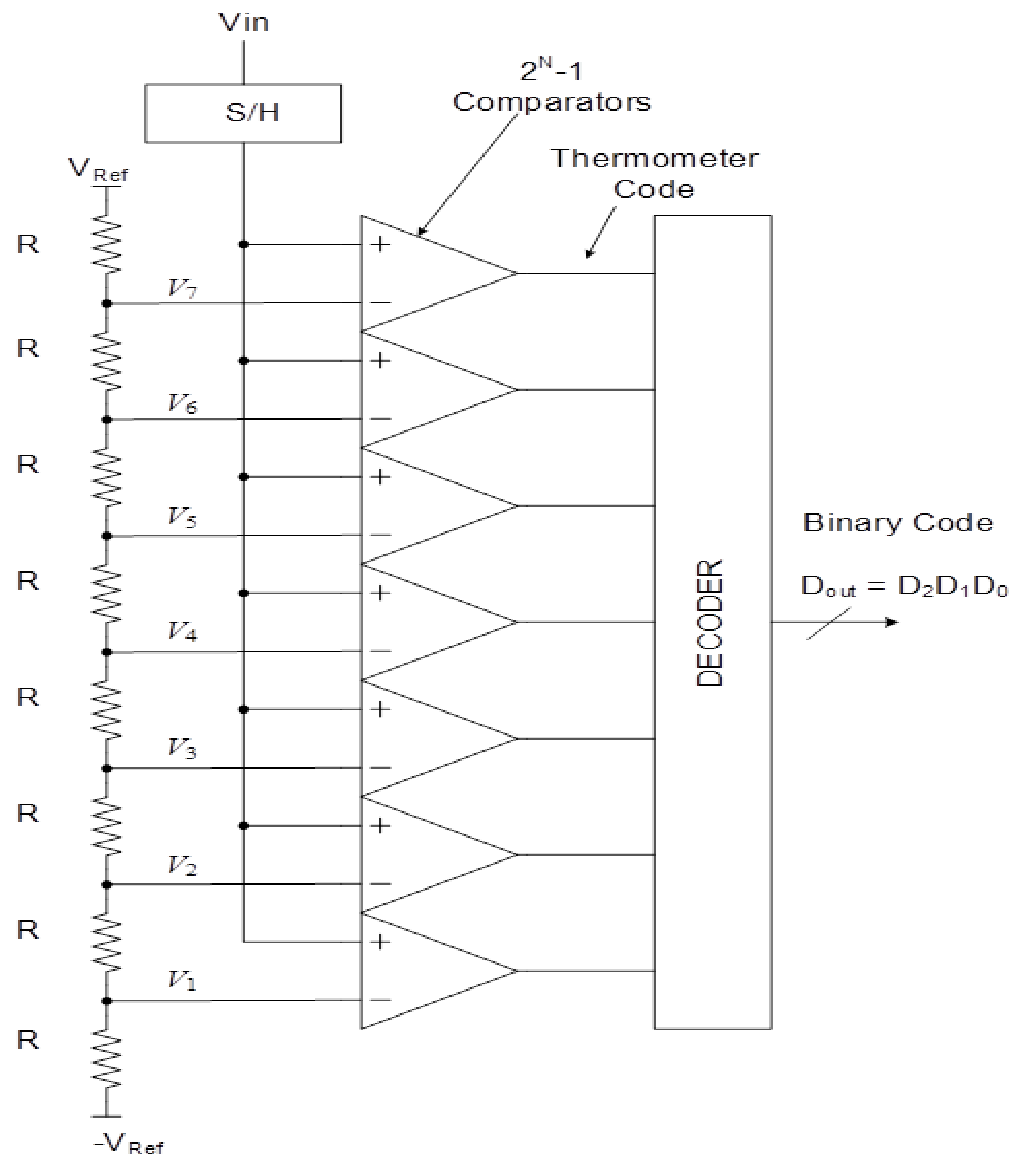
Example: AD9467 at 122.88MHz
IMD = -89dBc (-96dBFS)



Ahmed Ali

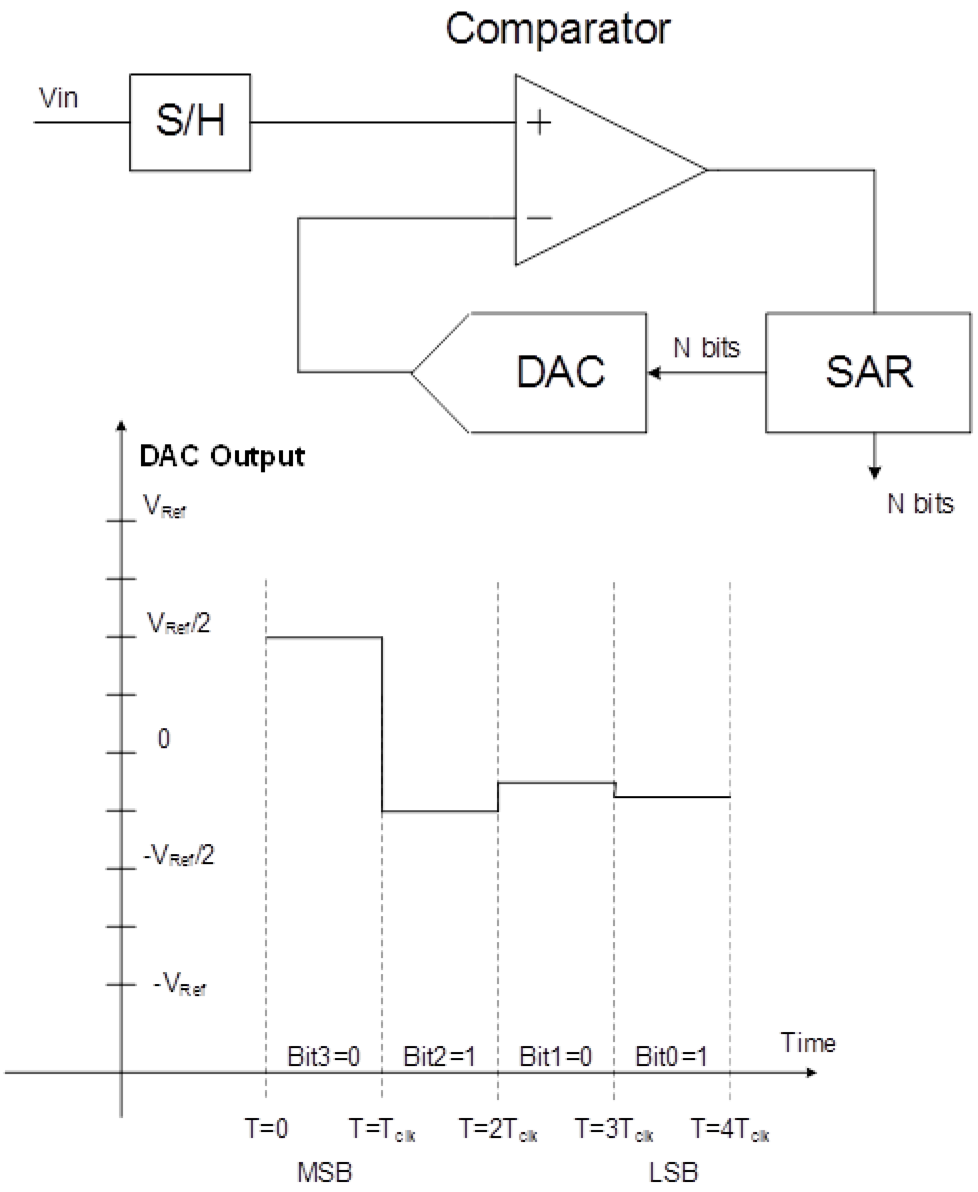
Flash A/D Converters

- Compares input to all levels simultaneously for maximum speed
- Very high speed (GS/s):
  - Sample rate fs = fclk
- Not efficient for high resolutions:
  - Area & Power a 2N
- Limited accuracy: (6-bit)
- Variations include: Interpolation and Folding Architectures

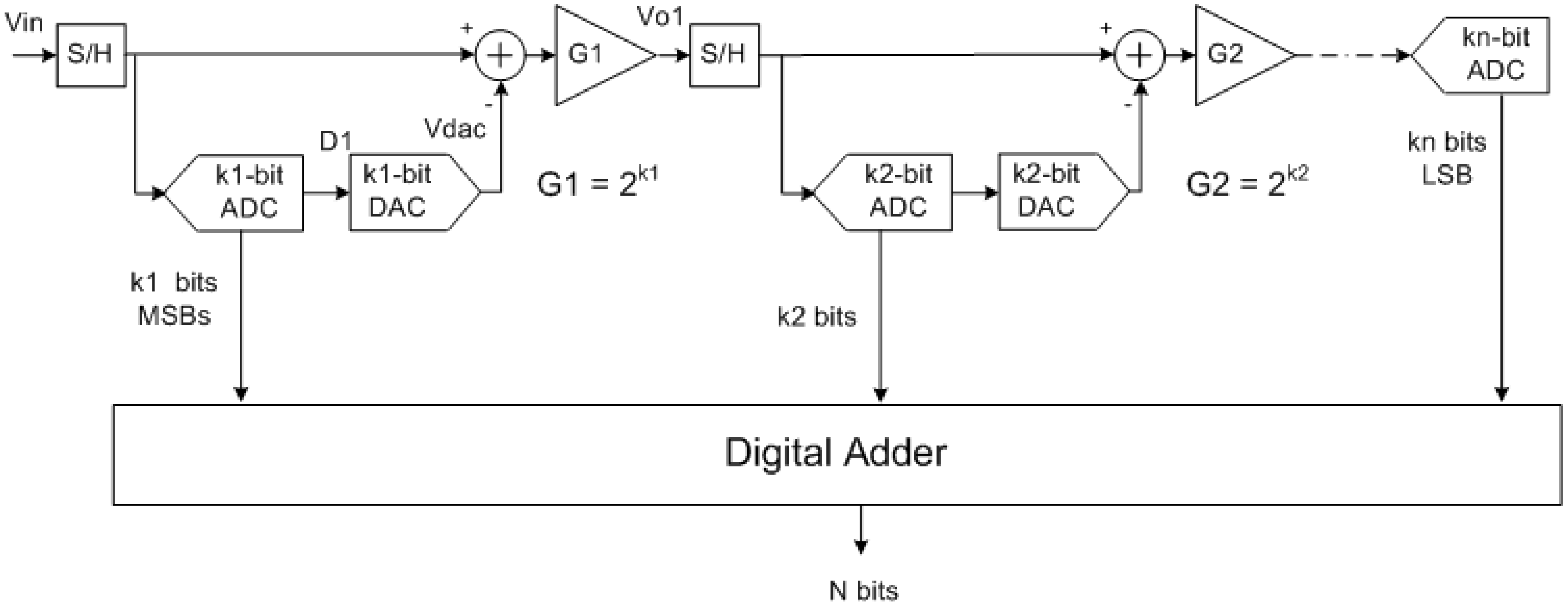


SAR A/D Converters

- Quantize the input by successively approaching the right answer using a binary search algorithm
- Very efficient in power and hardware
- Relatively low speed:
  - Conversion time proportional to N
- Small acquisition time
- High accuracy
- Currently: 18-20+ bit, 10-20+ MS/s



Pipelined A/D Converter



$$D_1 = 0, \pm 1, \pm 2, \pm 3 \ \ldots, \pm 2^{k1\text{-}1}$$

$$V_{o1} = G_1(V_{in} - V_{dac})$$

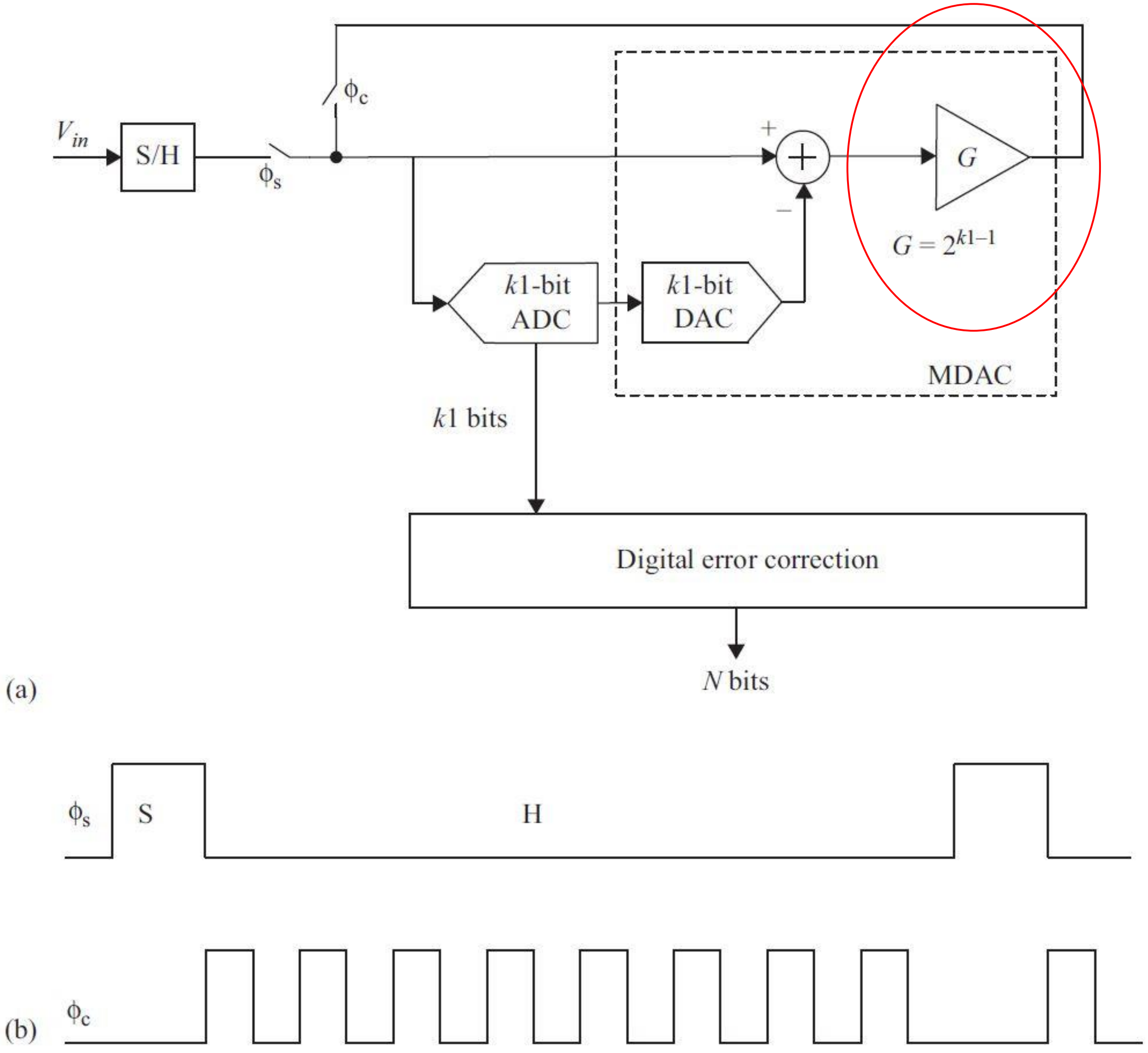
$$V_{o1} = G_1\left(V_{in} - \frac{D_1 \cdot V_{Ref}}{2^{k1-1}}\right)$$

$$\boxed{G_1 = 2^{k1}}$$

Ahmed Ali

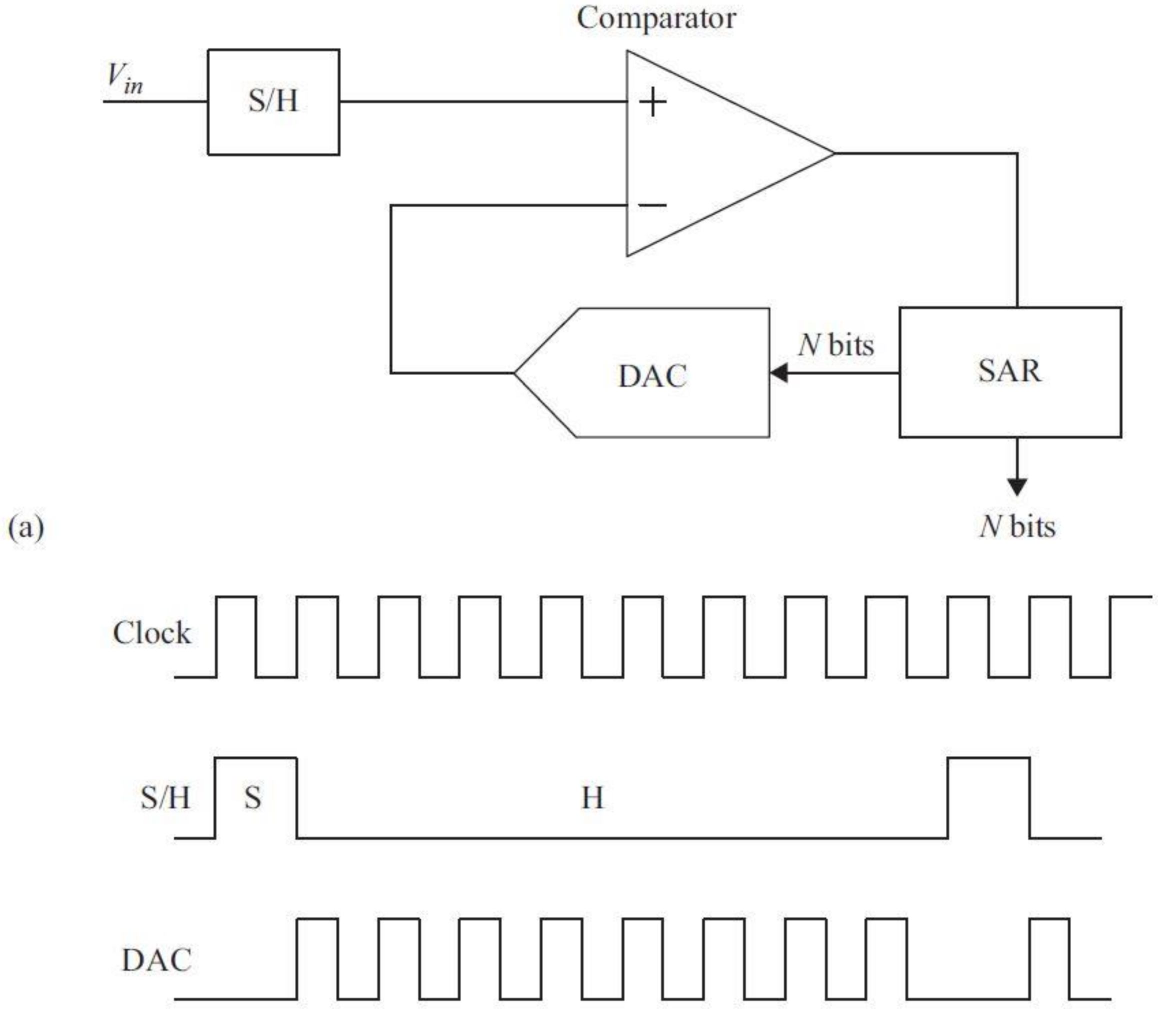
Algorithmic/Cyclic (pipelined) ADC

- Folding the n pipeline stages into one stage
- For n cycles => n times slower
- We need n in parallel to achieve the same speed
- Inter-stage amplifier is needed



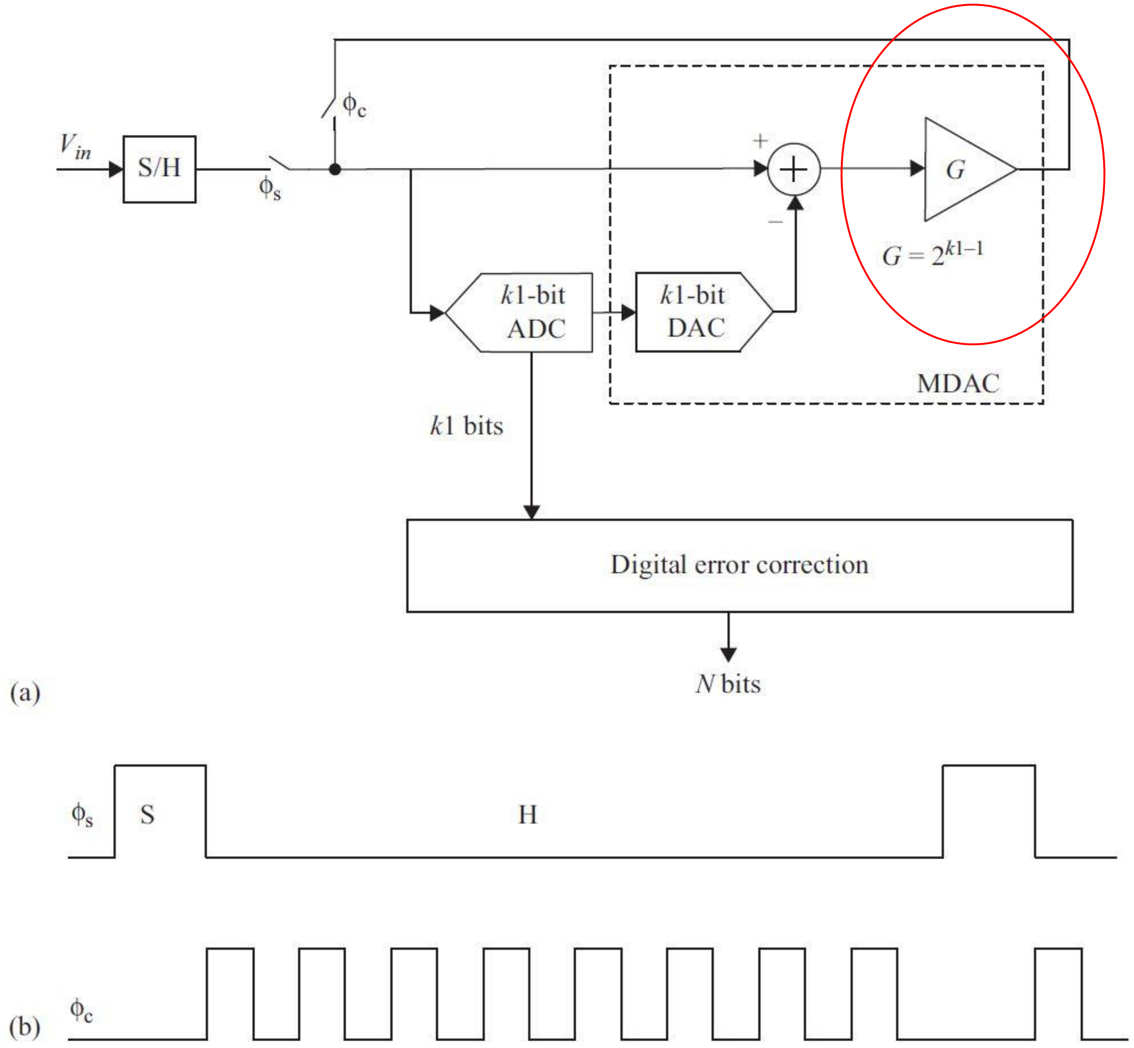
SAR ADC

- Folding the n pipeline stages into one stage
- For n cycles => n times slower
- We need n in parallel to achieve the same speed
- No inter-stage amplifier is needed



Algorithmic/Cyclic (pipelined) ADC

- Folding the n pipeline stages into one stage
- For n cycles => n times slower
- We need n in parallel to achieve the same speed
- Inter-stage amplifier is needed
- If cost of amplifier is reduced substantially
  - => similar to SAR ADC
- If unfolded into a pipeline
  - => n-times faster
  - => more efficient and higher performance than n parallel SARs



Ahmed Ali

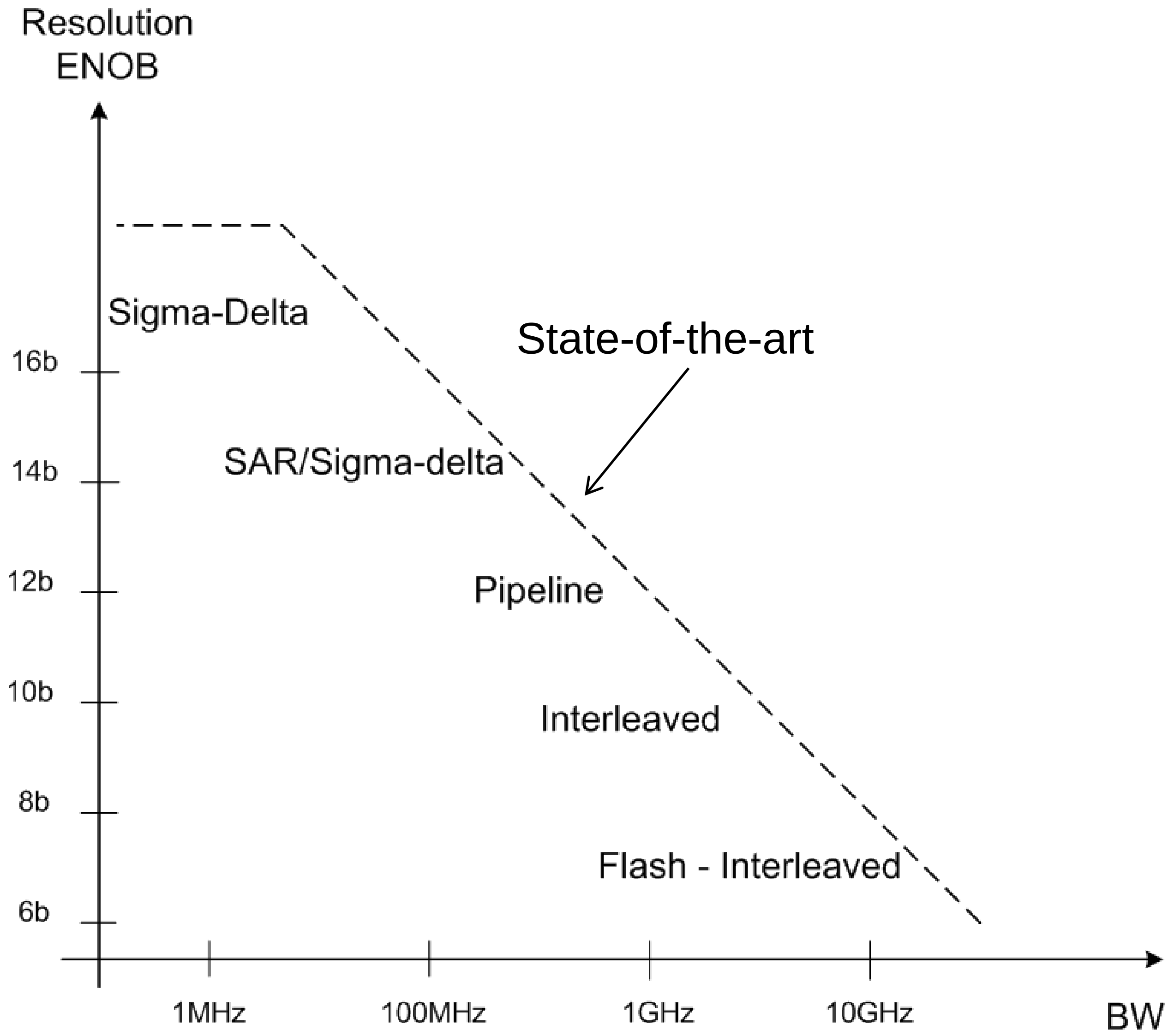
Pipelined A/D Converters
High-Speed and High-Resolution

- Employs the pipelining approach to work on multiple samples simultaneously
  - => long conversion time (latency) but high throughput (sample rate)
- Divide-and-conquer: divide the N bits conversion into n stages, each converting N/n bits (more if redundancy is employed)
- Higher resolution than flash ADCs and faster than SARs
- Can achieve resolutions up to 14-16 bits and sample rates up to 3GS/s
- Their algorithmic feed-forward nature makes them amenable to DSP techniques for performance/speed enhancement
  - => compatible with fine geometry CMOS processes and digitally assisted analog
- Speed extension through interleaving
- In the high speed space, pipelines in their different incarnations are alive and kicking: interleaved, open-loop, pipe-SARs, continuous-time pipes, etc.

The report of my death was an exaggeration!

\- Mark Twain

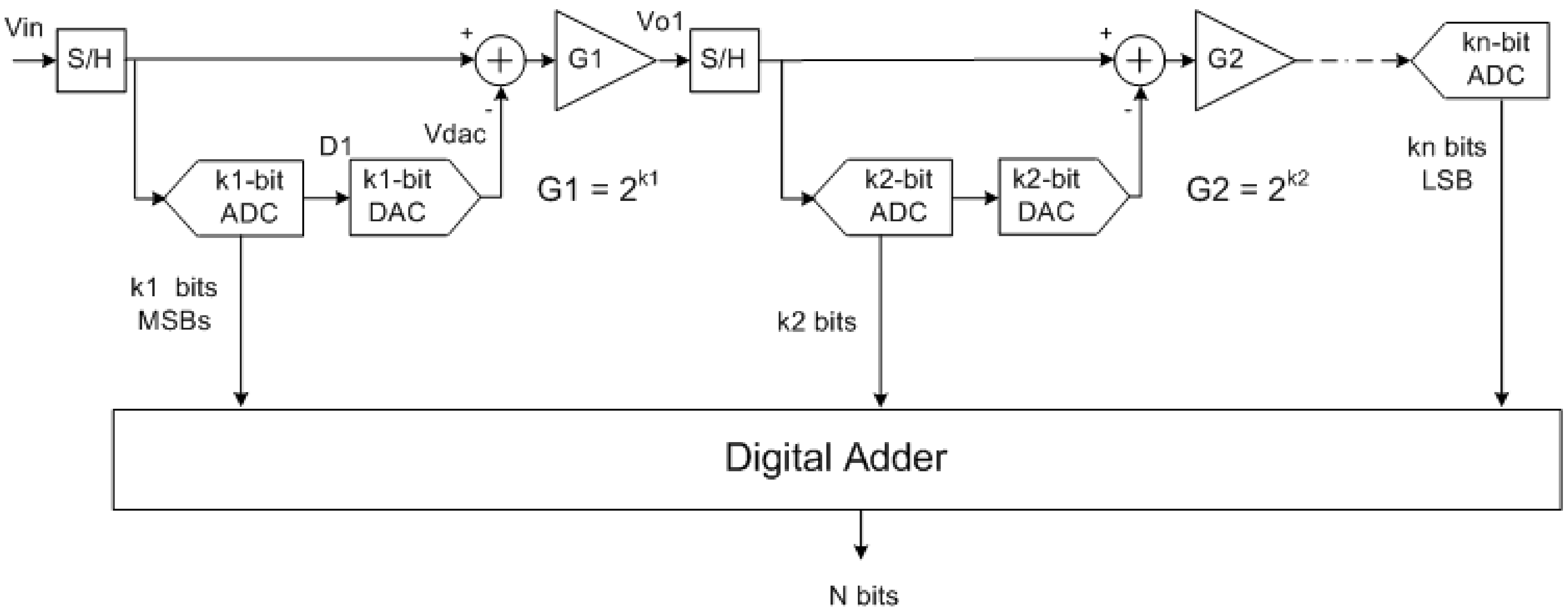
A/D Converter Architectures



Ahmed Ali

PIPELINED A/D CONVERTERS

Pipelined A/D Converter



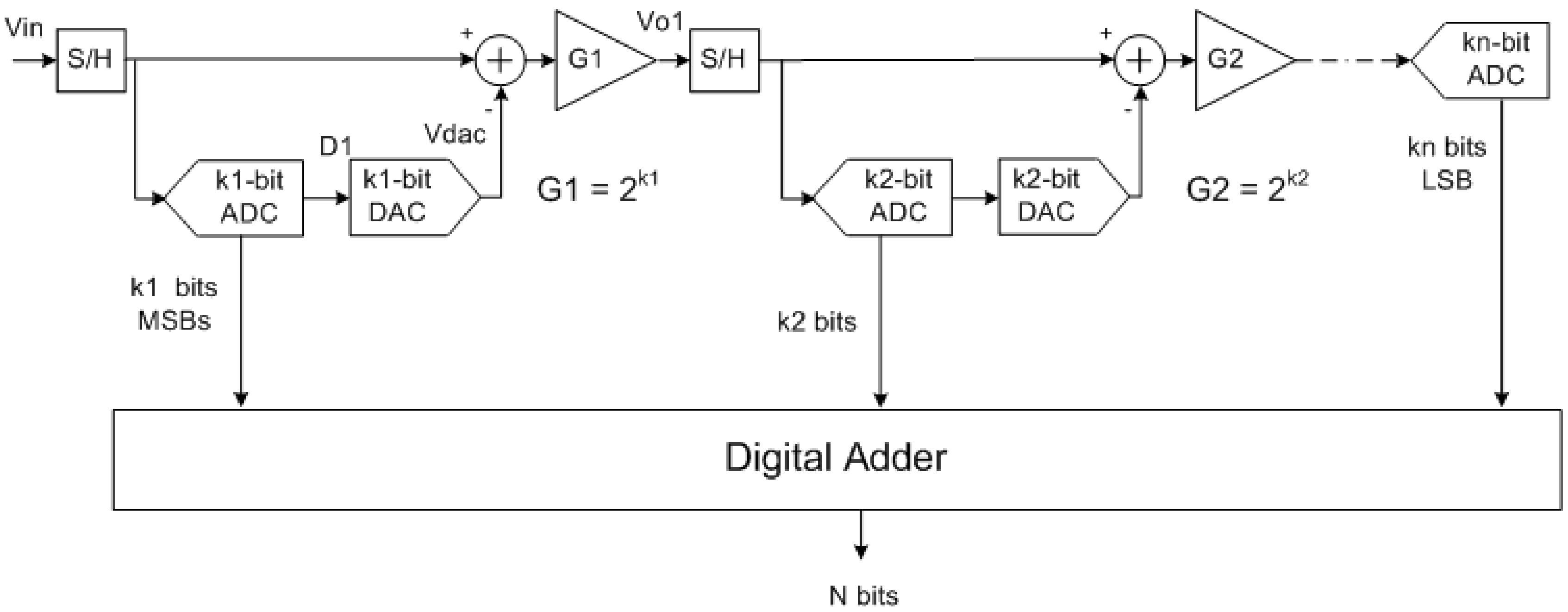
$$D_1 = 0, \pm 1, \pm 2, \pm 3\ \ldots, \pm 2^{k1\text{-}1}$$

$$V_{o1} = G_1(V_{in} - V_{dac})$$

$$V_{o1} = G_1\left(V_{in} - \frac{D_1 \cdot V_{Ref}}{2^{k1-1}}\right)$$

$$\boxed{G_1 = 2^{k1}}$$

Pipelined A/D Converter



$$D_1 = 0, \pm 1, \pm 2, \pm 3 \ \ldots, \pm 2^{k1\text{-}1}$$

$$V_{o1} = G_1 (V_{in} - V_{dac})$$

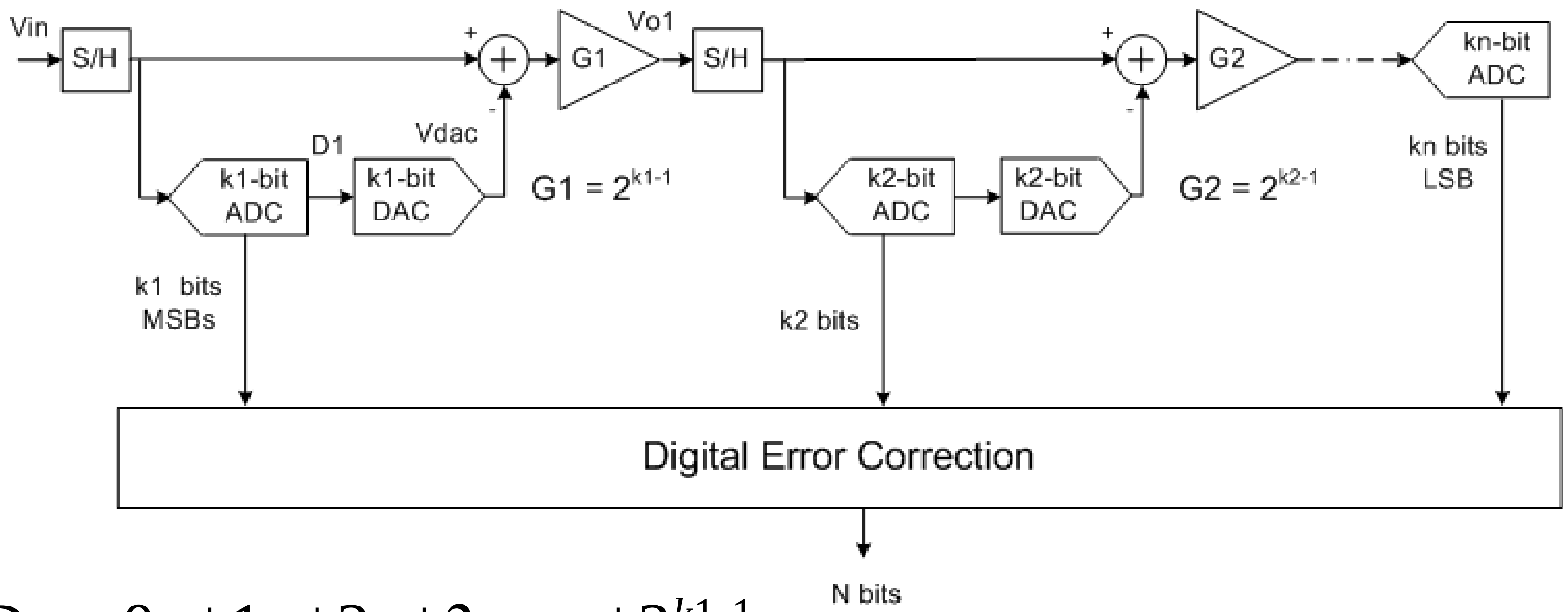
$$V_{o1} = G_1 \left(V_{in} - \frac{D_1 \cdot V_{Ref}}{2^{k1-1}} \right)$$

$$\boxed{G_1 = 2^{k1}}$$

Not Practical

Ahmed Ali

Pipelined A/D Converter With Redundancy



$$D_1 = 0, \pm 1, \pm 2, \pm 3\ \ldots, \pm 2^{k1\text{-}1}$$

$$V_{o1} = G_1(V_{in} - V_{dac})$$

$$V_{o1} = G_1\left(V_{in} - \frac{D_1 \cdot V_{Ref}}{2^{k1-1}}\right)$$

$$\boxed{G_1 = 2^{k1-1}}$$

Ahmed Ali

Pipeline Stage Transfer Function

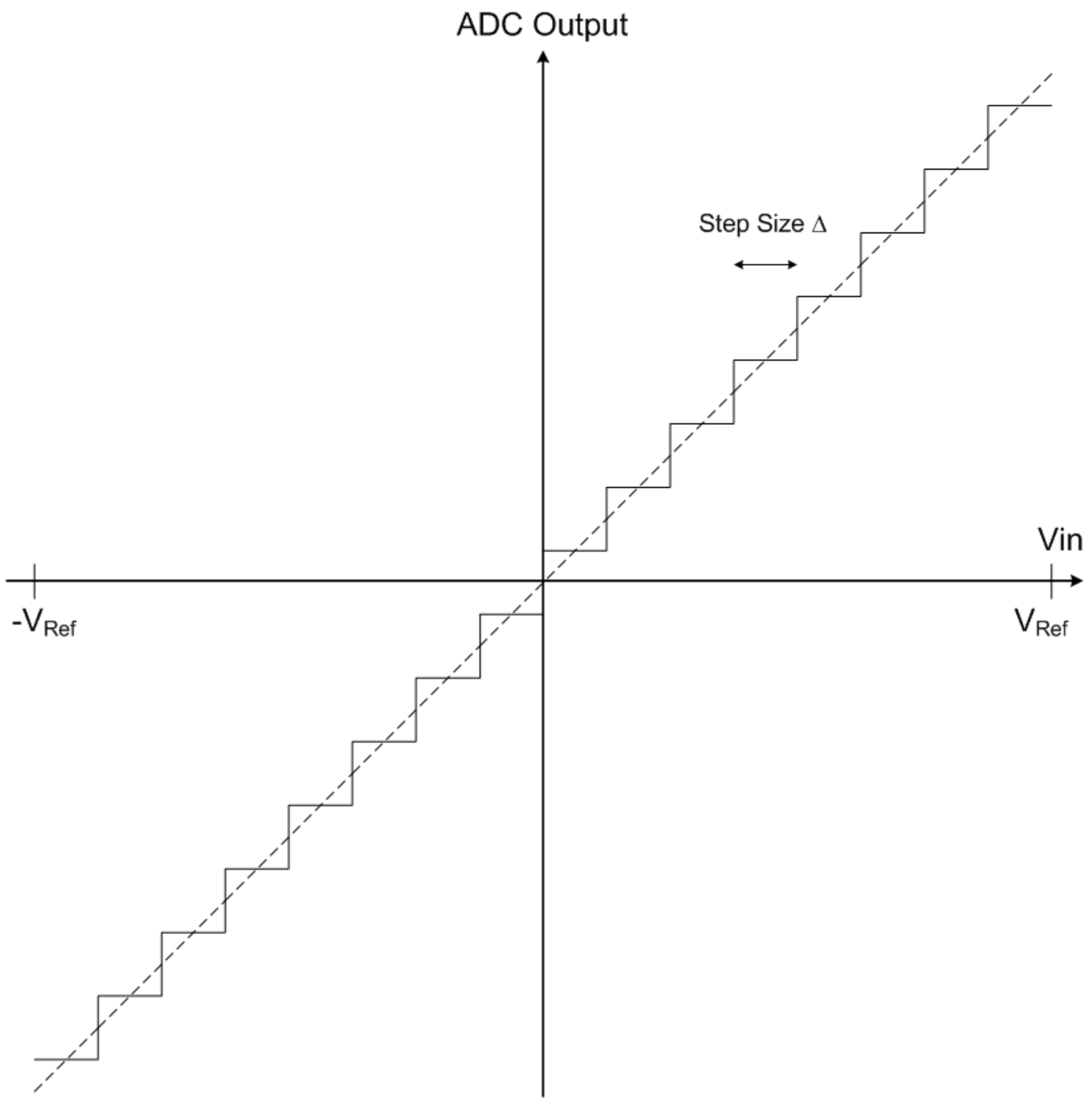
- **Residue:**

$$V_{o1} = G_1(V_{in} - V_{dac})$$

$$V_{o1} = G_1\left(V_{in} - \frac{D_1 \cdot V_{Ref}}{2^{k1-1}}\right)$$

- **The difference between the input signal and the quantized version of it**

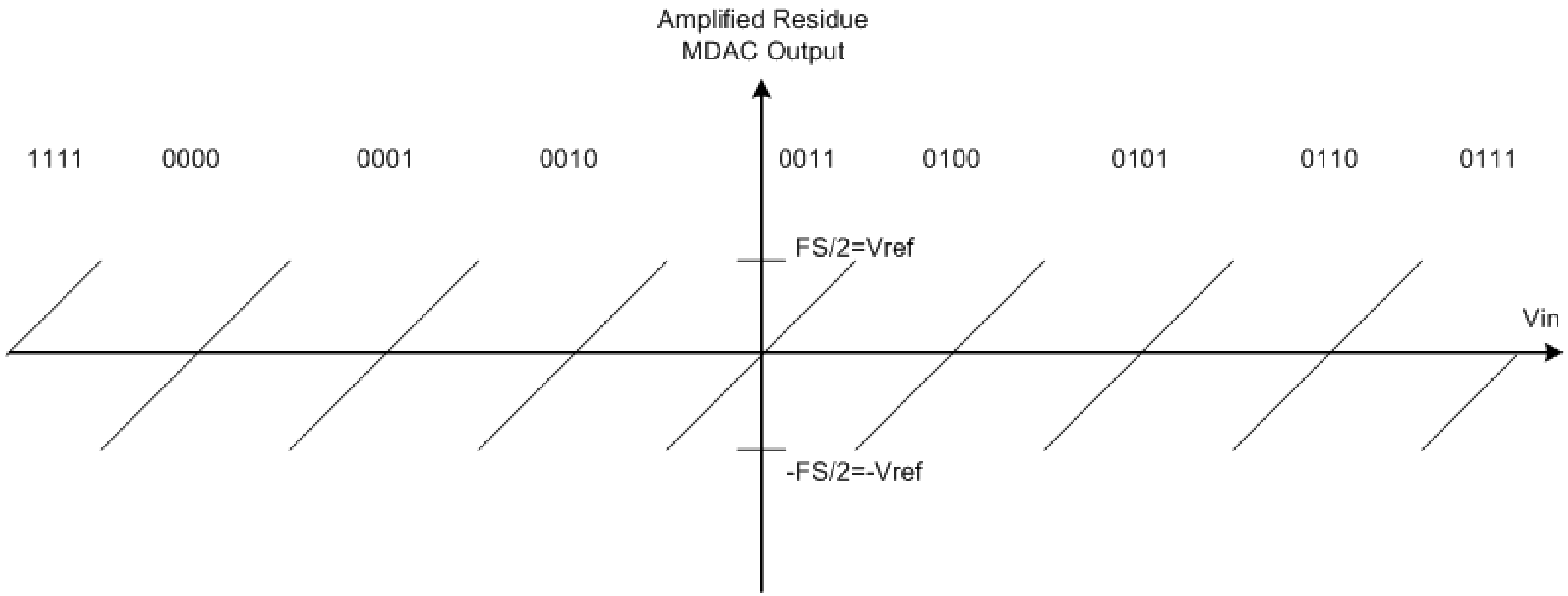
- **Ideally, the "Transfer Function" looks like a saw-tooth**



Ahmed Ali

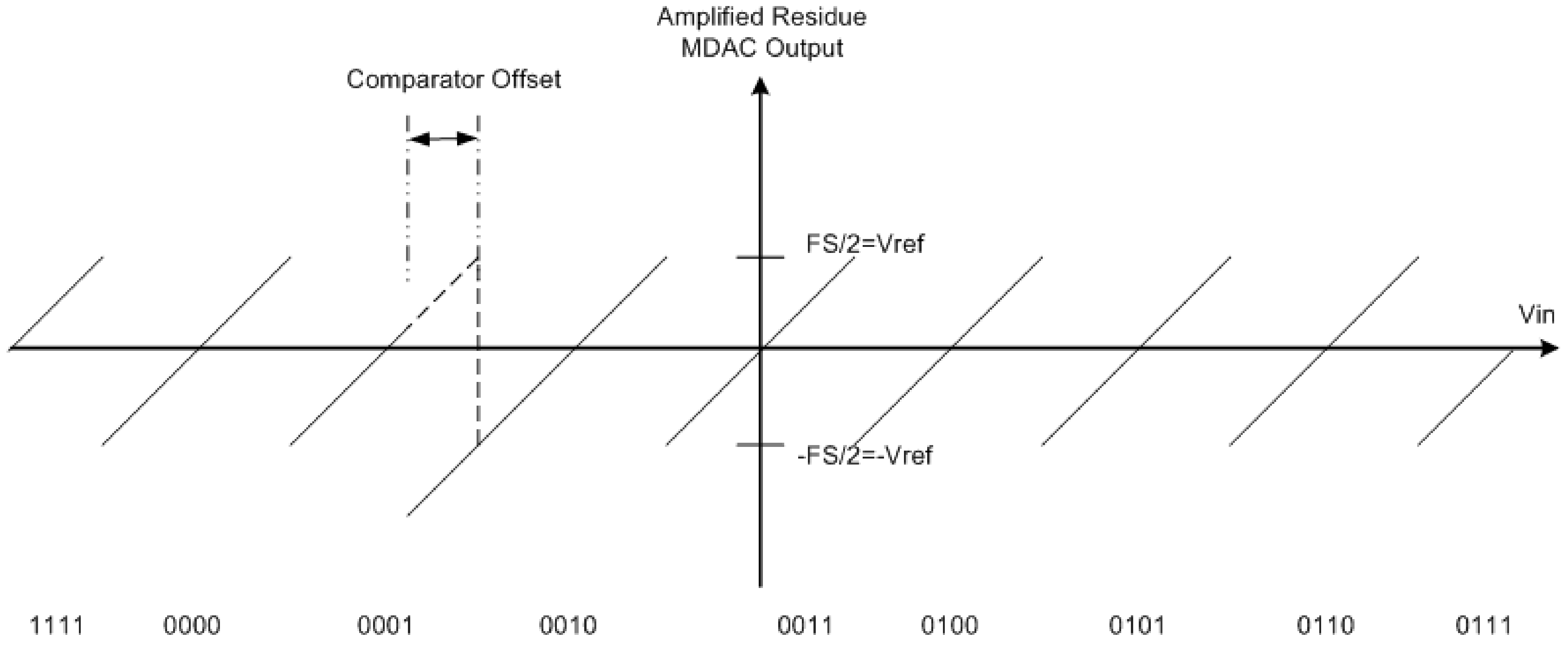
First Stage Residue
Why We Need Redundancy

- **Ideally, the residue of each stage would be within the range of the back-end ADC**



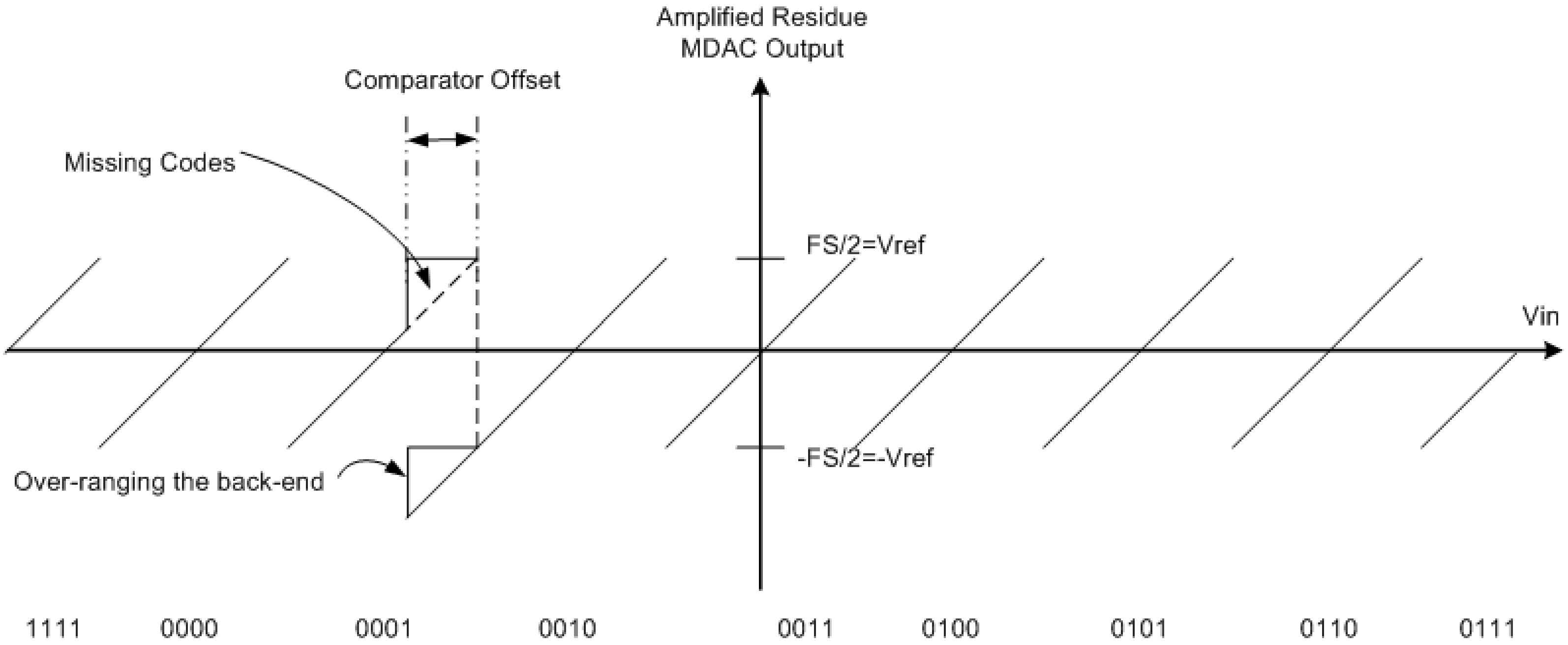
First Stage Residue With Comparator Offsets

- Offsets in the stage ADC comparators change the residue waveform



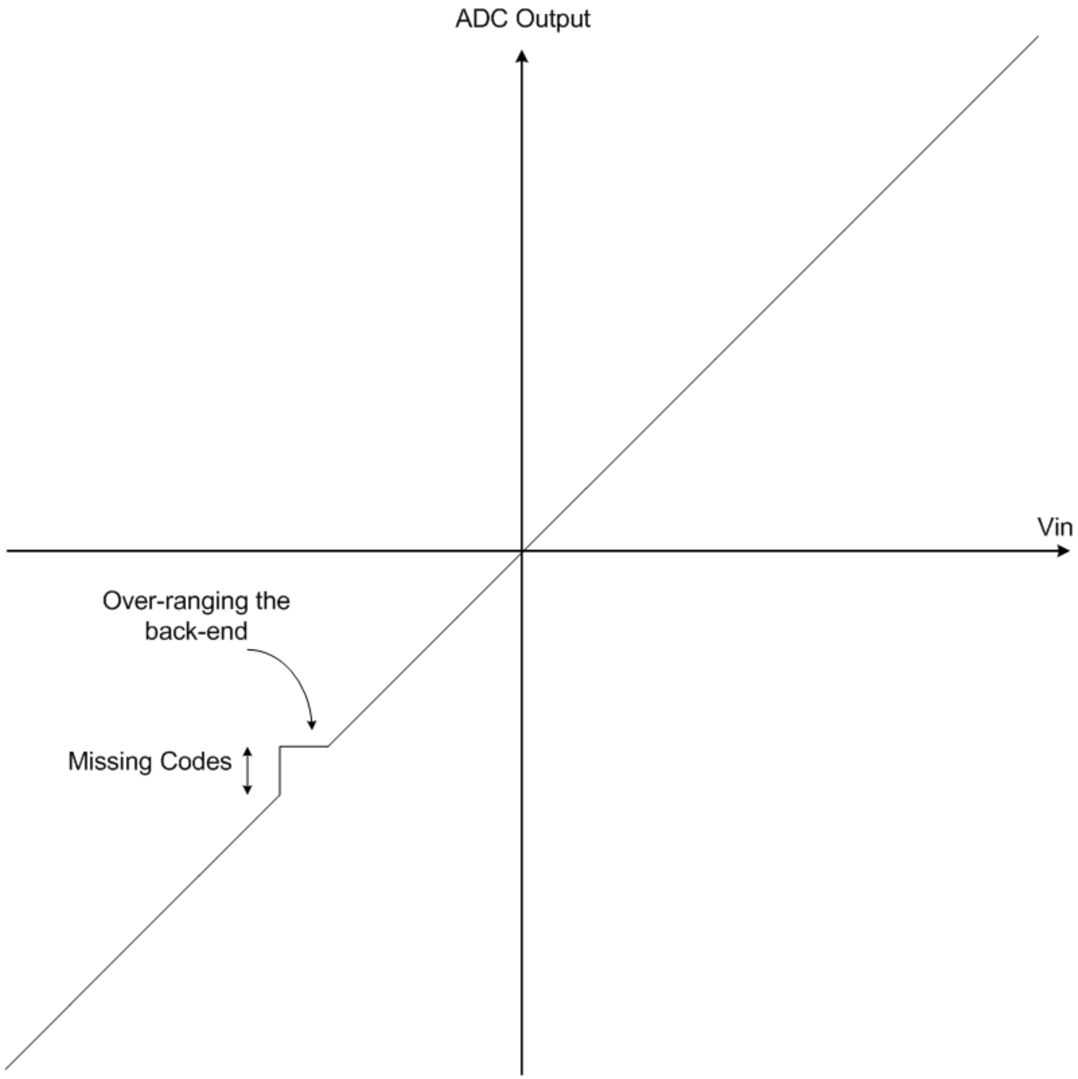
First Stage Residue With Comparator Offsets

- Offsets would lead to missing codes and over-ranging of the back-end ADC => Redundancy is needed



ADC Output
With Comparator Offsets

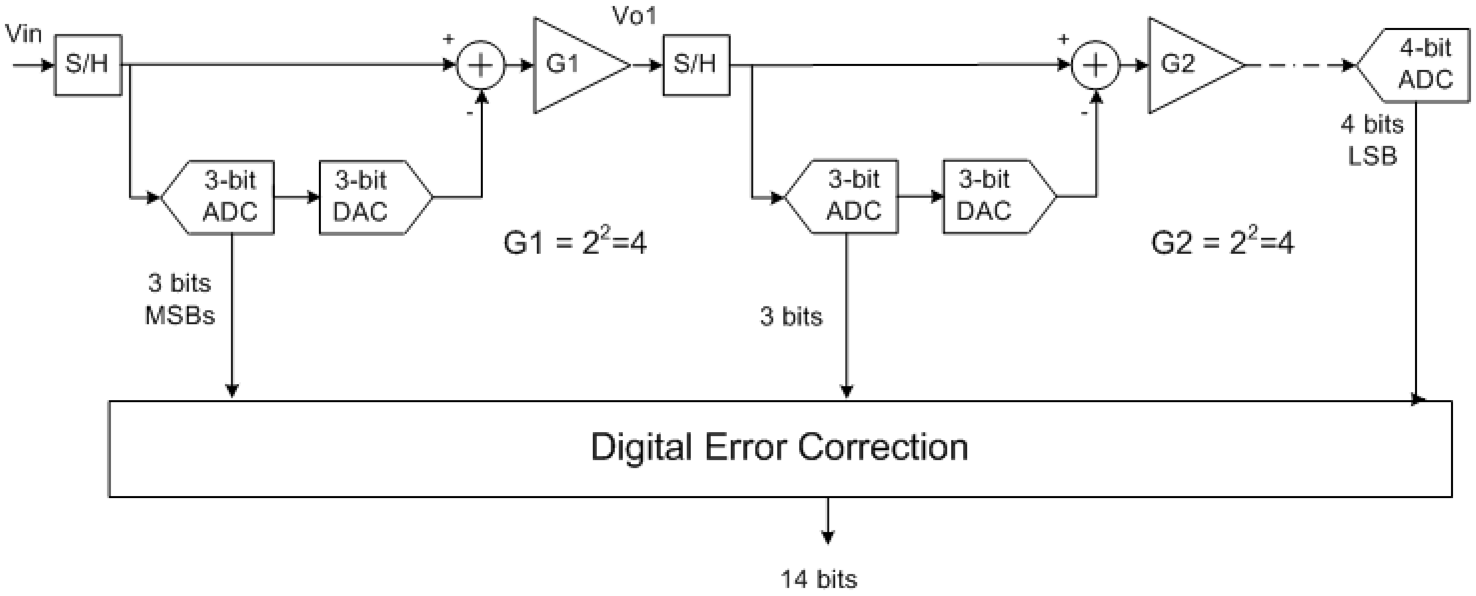
- Offsets would lead to missing codes and over-ranging of the back-end ADC
- => Redundancy is needed to de-desensitize the overall ADC to offsets in the ADC’s of each stage
- Typically, one bit of redundancy is employed



Ahmed Ali

Pipelined A/D Converter With Redundancy

$$G_1 = 2^{3-1} = 4$$



- **G = 4**
- **Allowed Offset (output referred) = ±FS/4**
- **Allowed Offset (input referred) = ±FS/16**

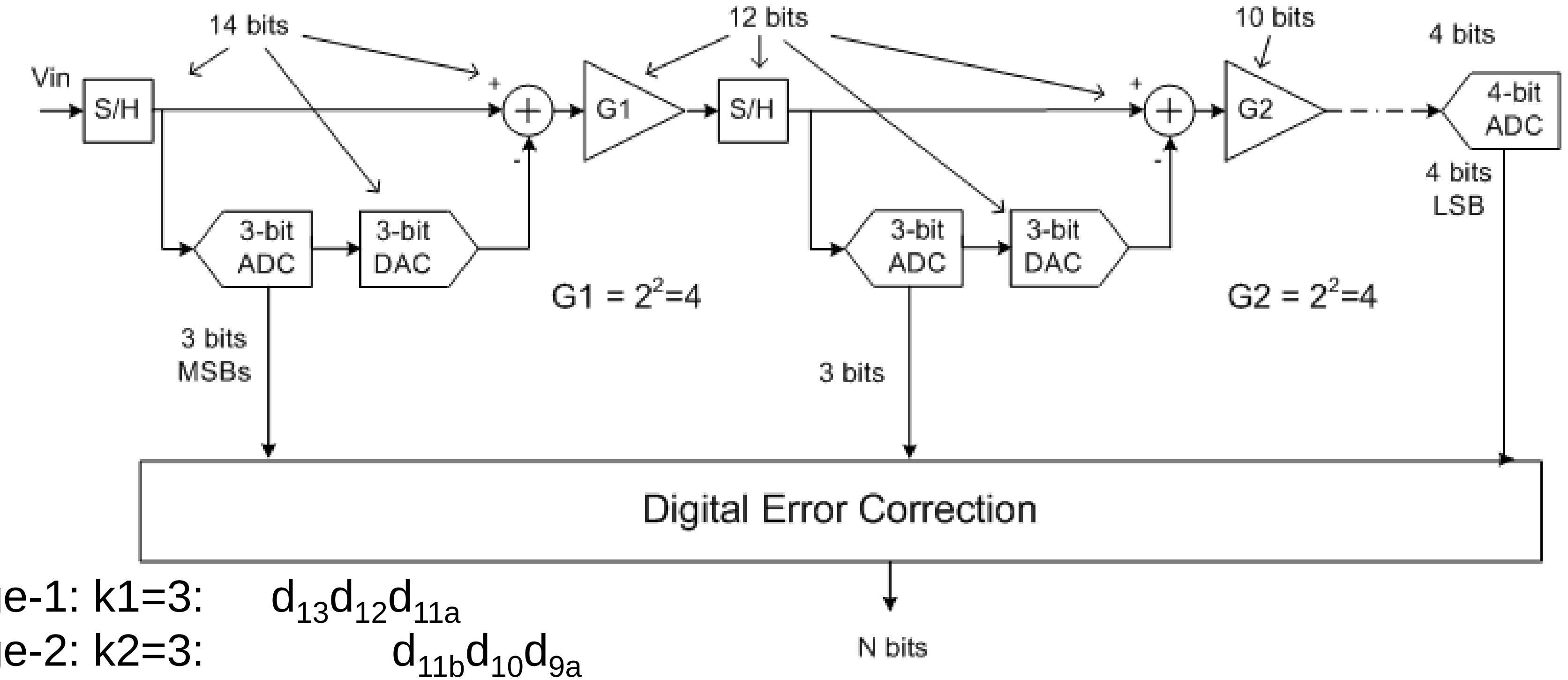
$$D_1 = 0, \pm 1, \pm 2, \pm 3 \ldots, \pm 2^{k1\text{-}1}$$

$$V_{o1} = G_1(V_{in} - V_{dac})$$

$$V_{o1} = G_1\left(V_{in} - \frac{D_1 \cdot V_{Ref}}{4}\right)$$

Ahmed Ali

Pipelined A/D Converter With Redundancy



Stage-1: k1=3: $d_{13}d_{12}d_{11a}$

Stage-2: k2=3: $d_{11b}d_{10}d_{9a}$

Stage-3: k3=3: $d_{9b}d_{8}d_{7a}$

Stage-4: k4=3: $d_{7b}d_{6}d_{5a}$

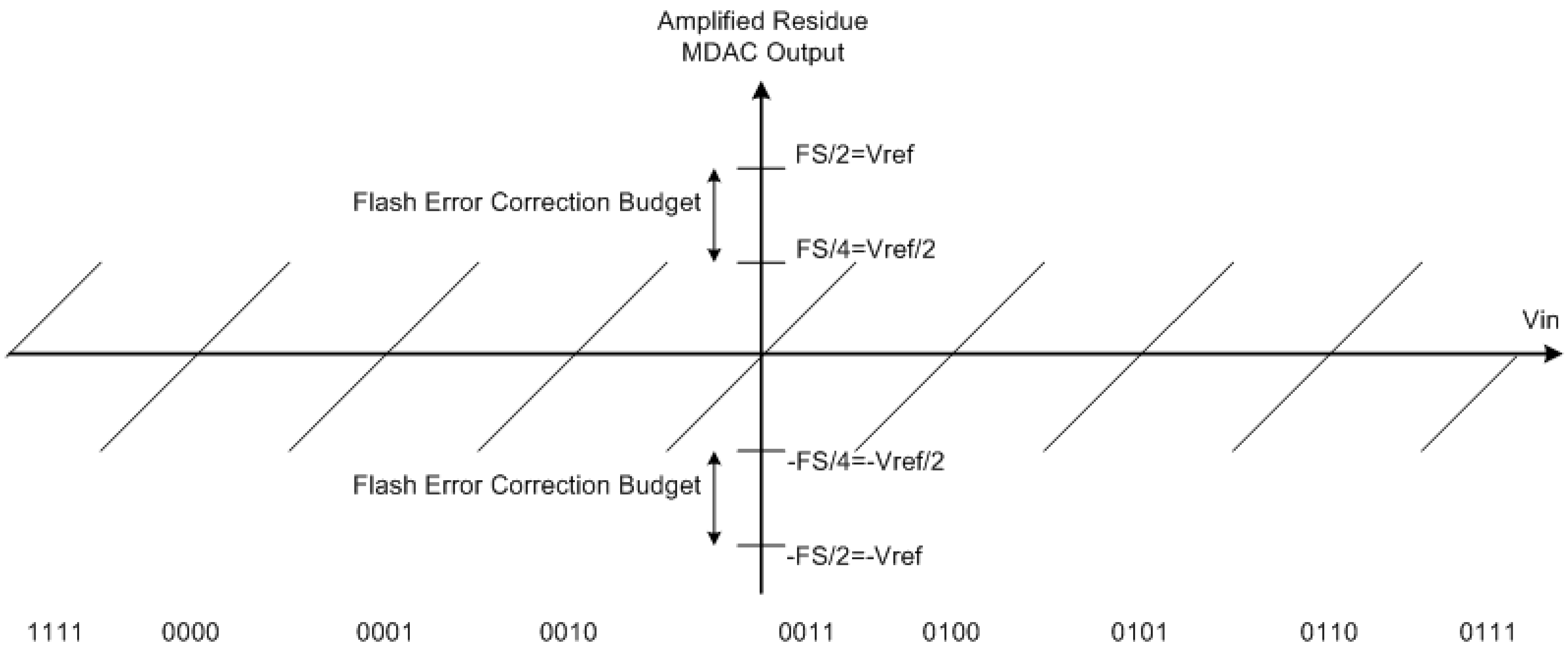
Stage-5: k5=3: $d_{5b}d_{4}d_{3a}$

Stage-6: k6=4: $d_{3b}d_{2}d_{1}d_{0}$

Output: 14-bits: $d_{13}d_{12}\,d_{11}\;d_{10}\;d_{9}\;d_{8}d_{7}\;d_{6}d_{5}\;d_{4}d_{3}\,d_{2}d_{1}d_{0}$

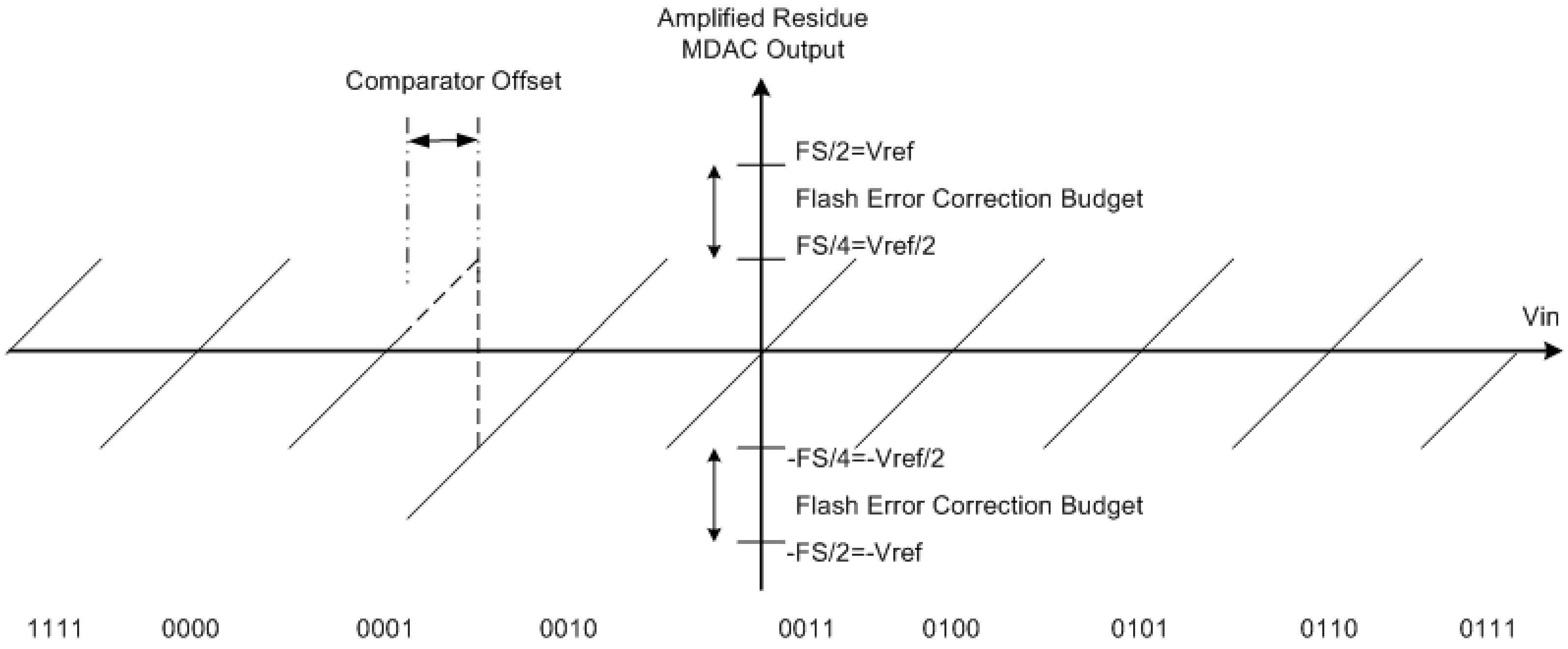
First Stage Residue With Redundancy

- Ideally, the first stage residue occupies only half of the dynamic range of the back-end ADC



First Stage Residue With Redundancy

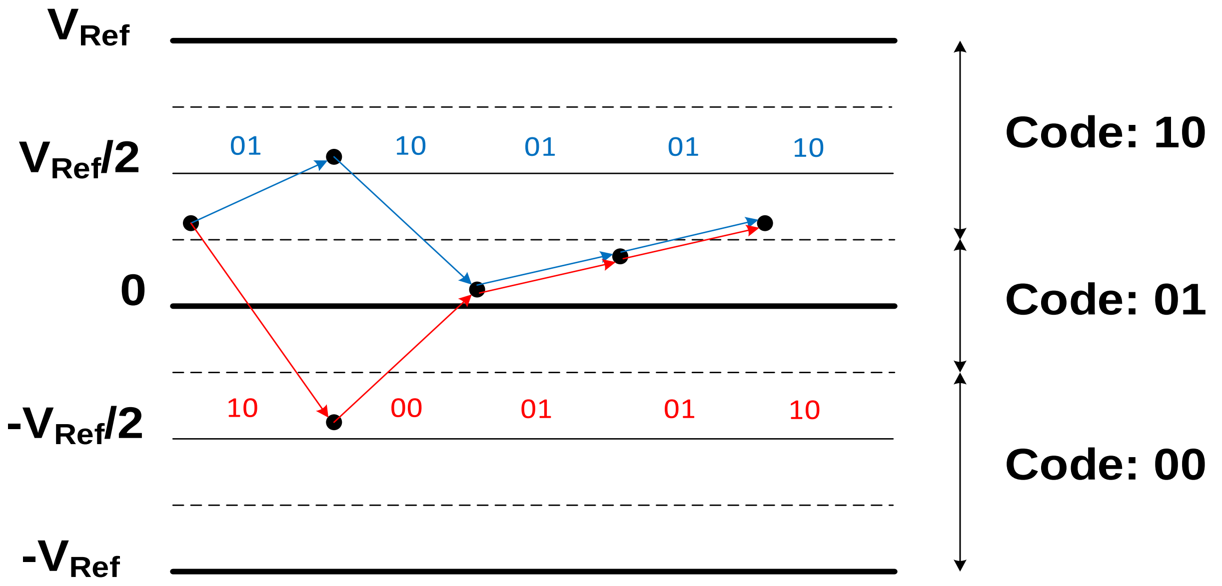
- Allowed Offset (output referred) = ±FS/4
- Allowed Offset (input referred) = ±FS/4G



Example
How Redundancy Works

- The answer is the same, independent of the path

- As long as the correction range is not exceeded



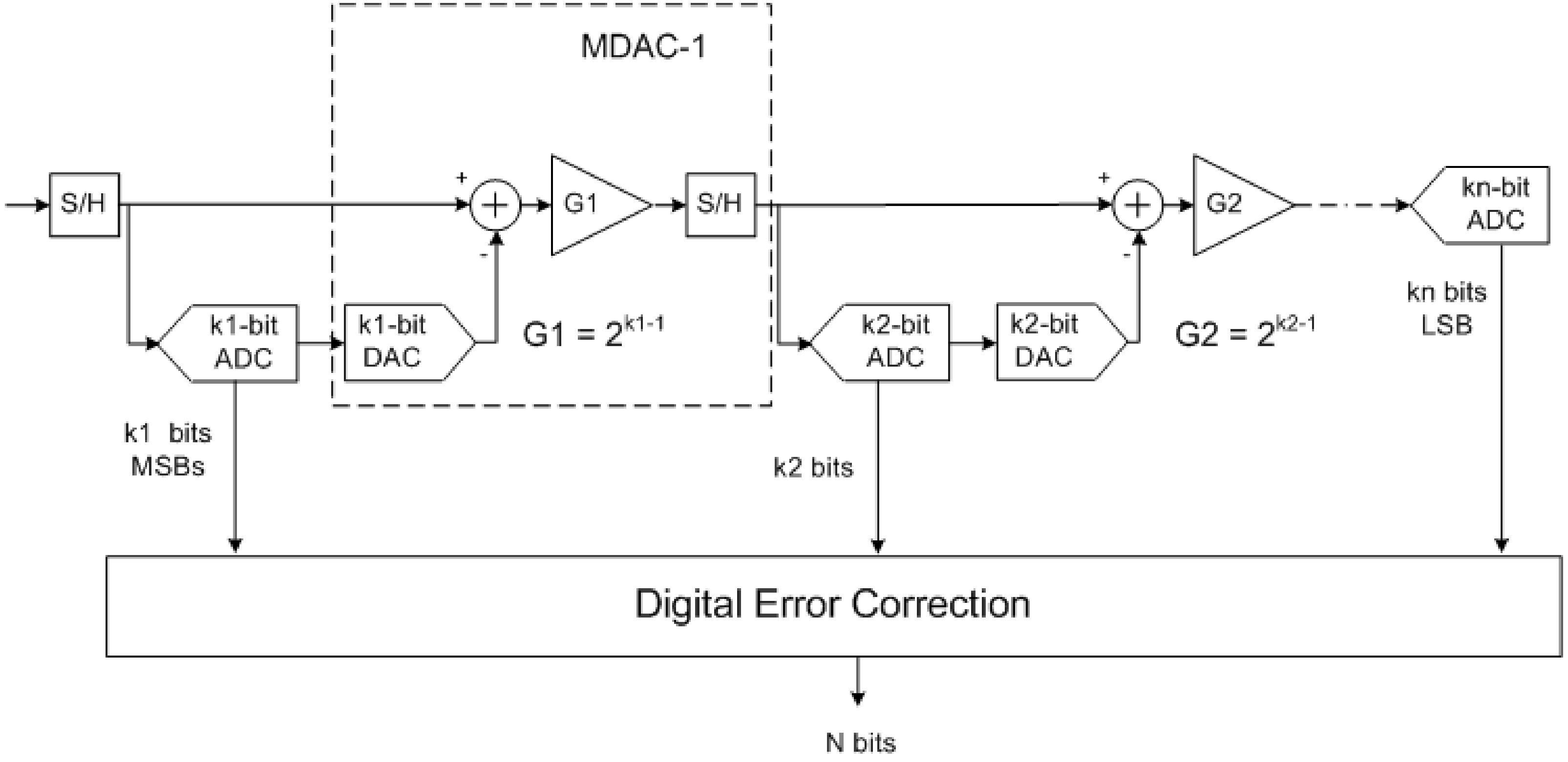
Align and sum the codes (raw bits):

Blue Path	Red Path
01	10
10	00
01	01
01	01
10	10
--------------	--------------
101000	101000

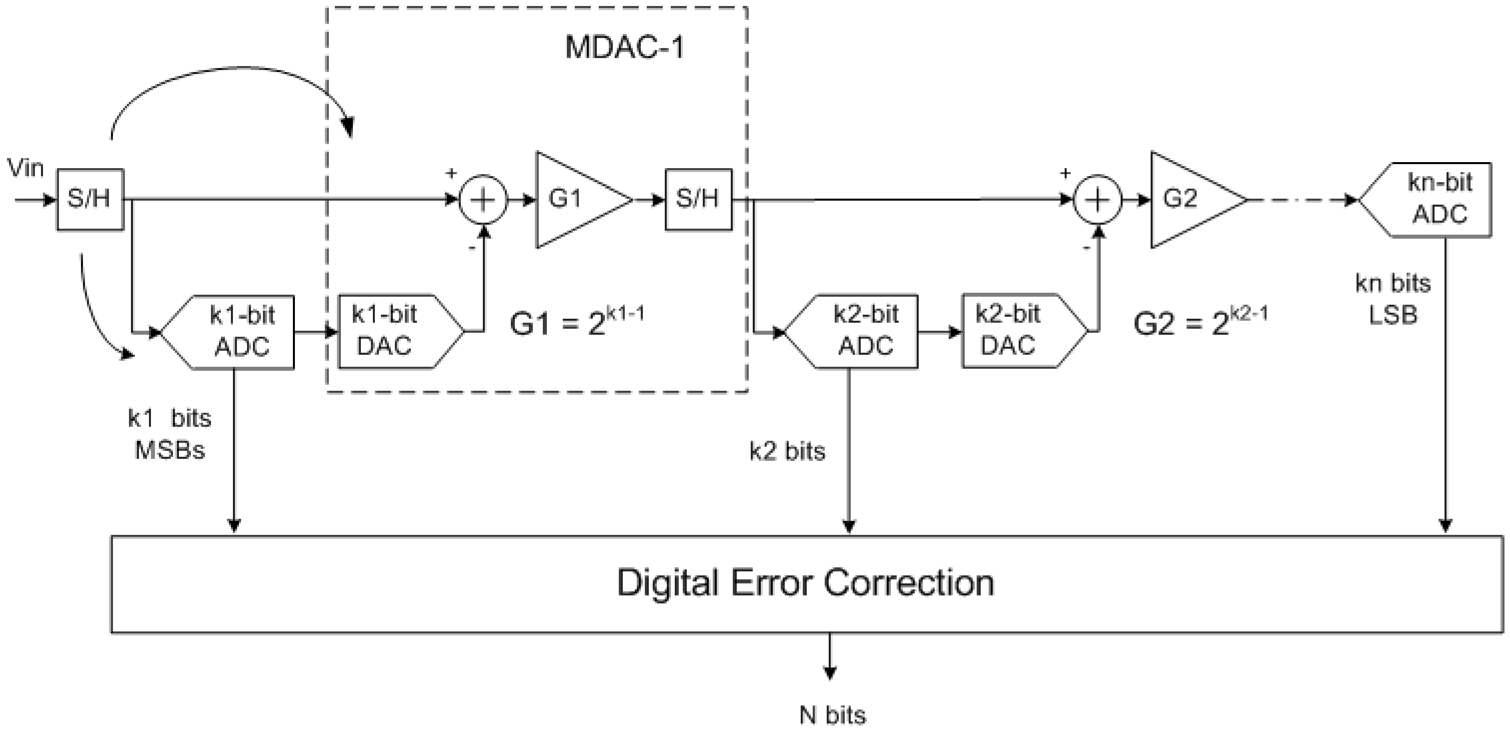
Ahmed Ali

Pipelined A/D Converter

- The DAC+Subtractor+Amplifier+S/H are called the MDAC

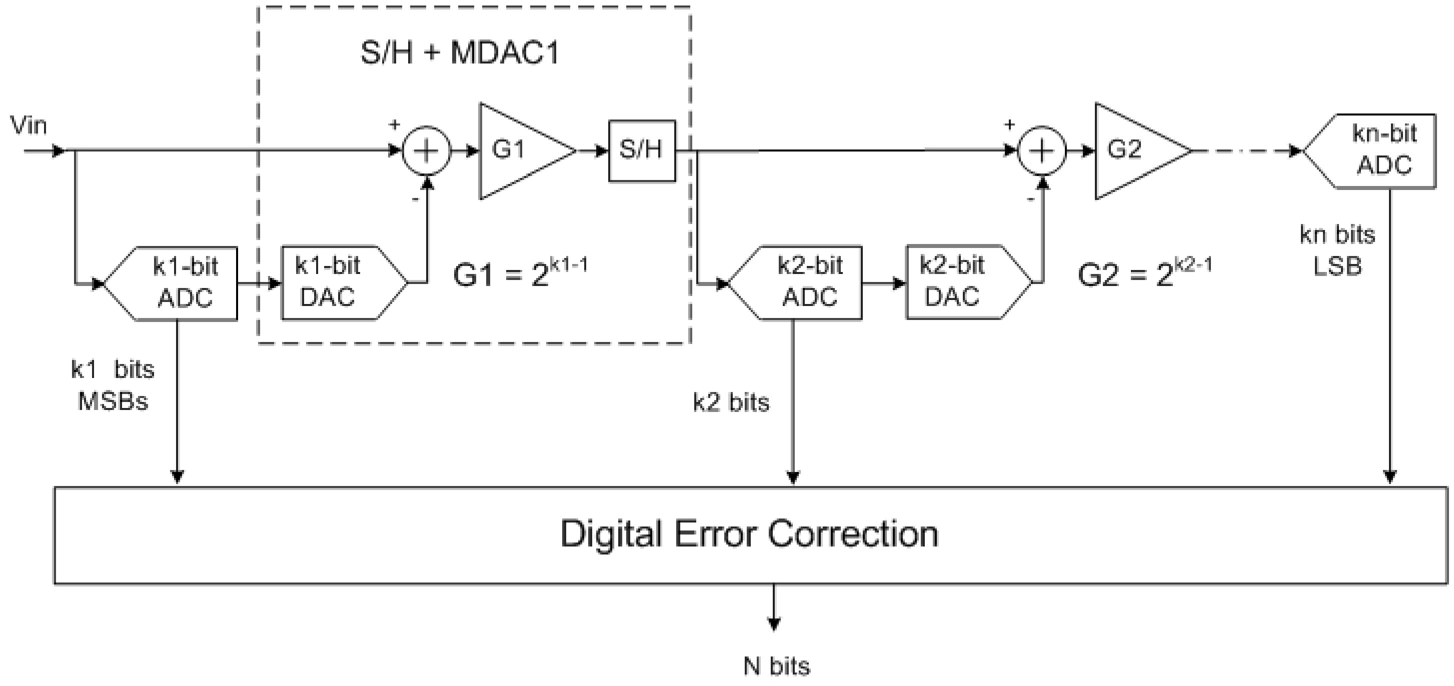


Pipelined A/D Converter



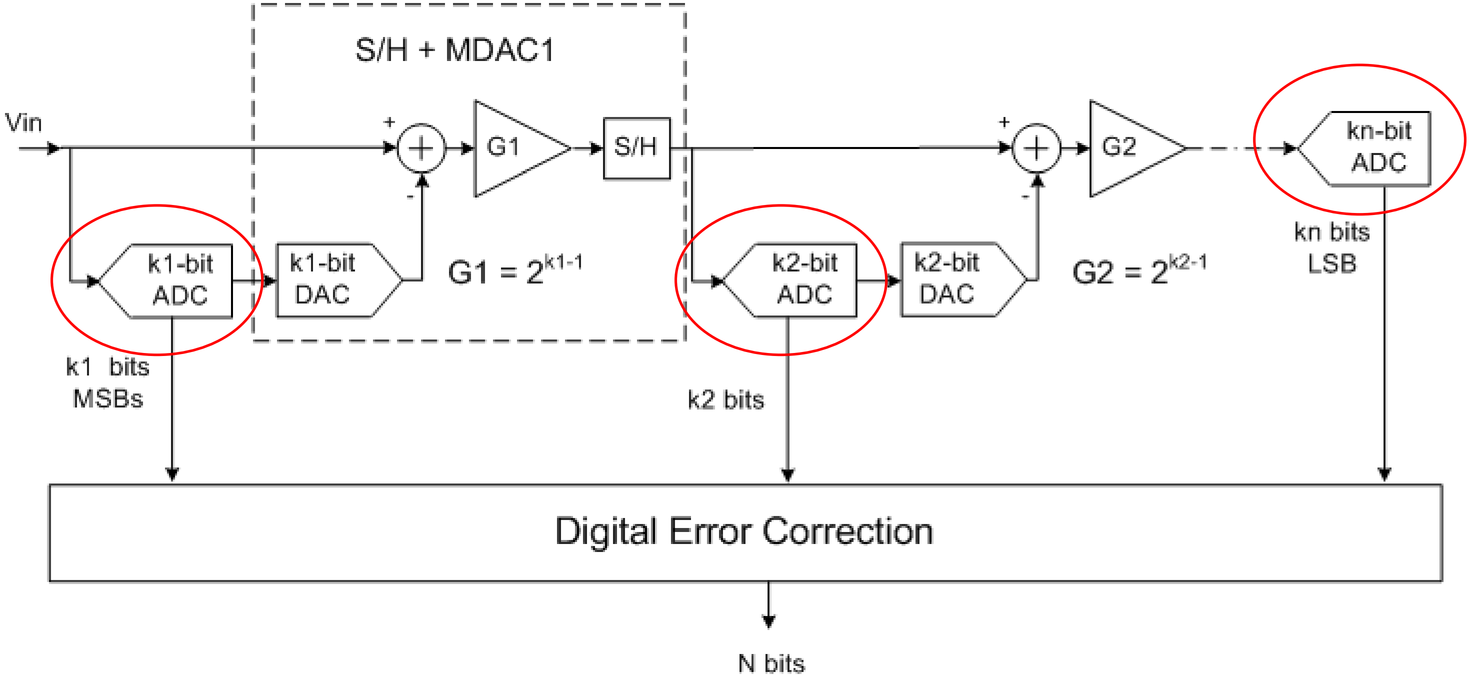
- The input S/H operation is sometimes performed by the first MDAC to save a dedicated SHA (S/H Amplifier)

Pipelined A/D Converter



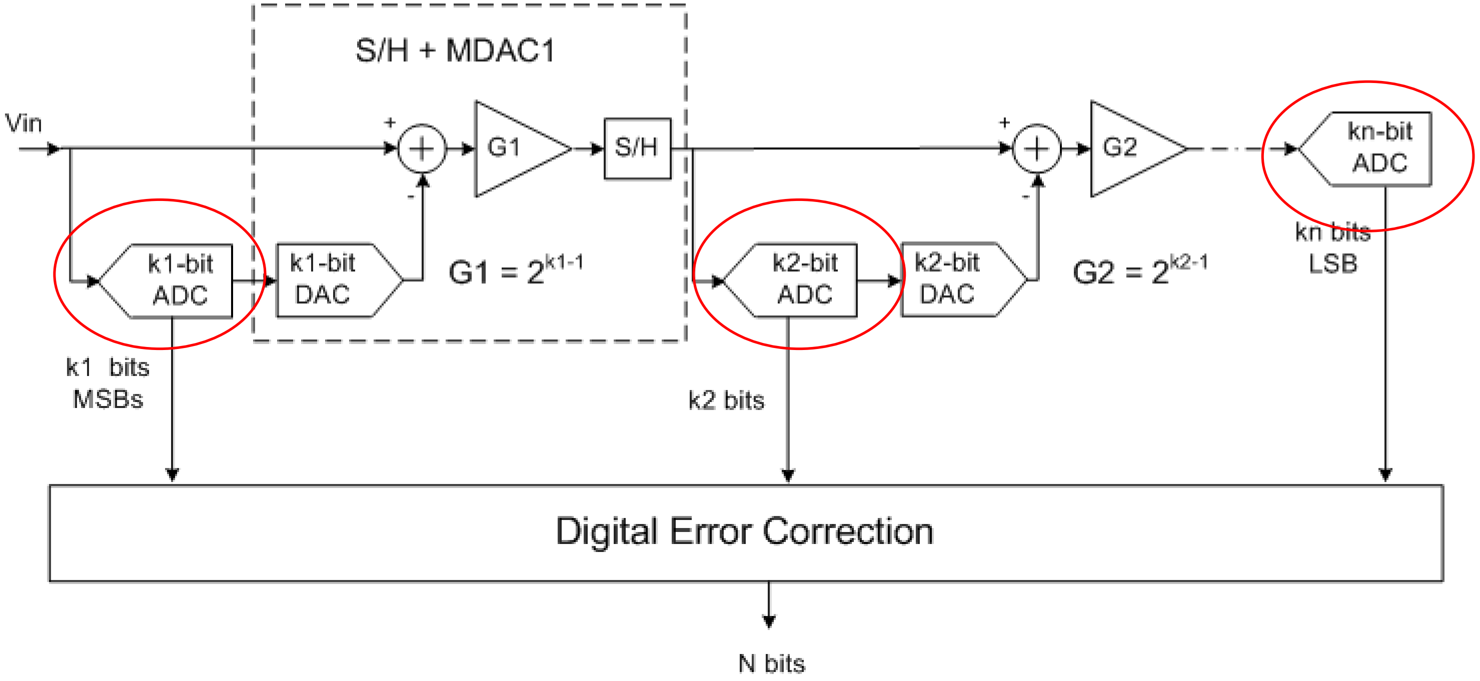
- The input S/H operation is sometimes performed by the first MDAC to save a dedicated SHA (S/H Amplifier)
- In this case, matching between the MDAC and flash is important

Pipelined A/D Converter



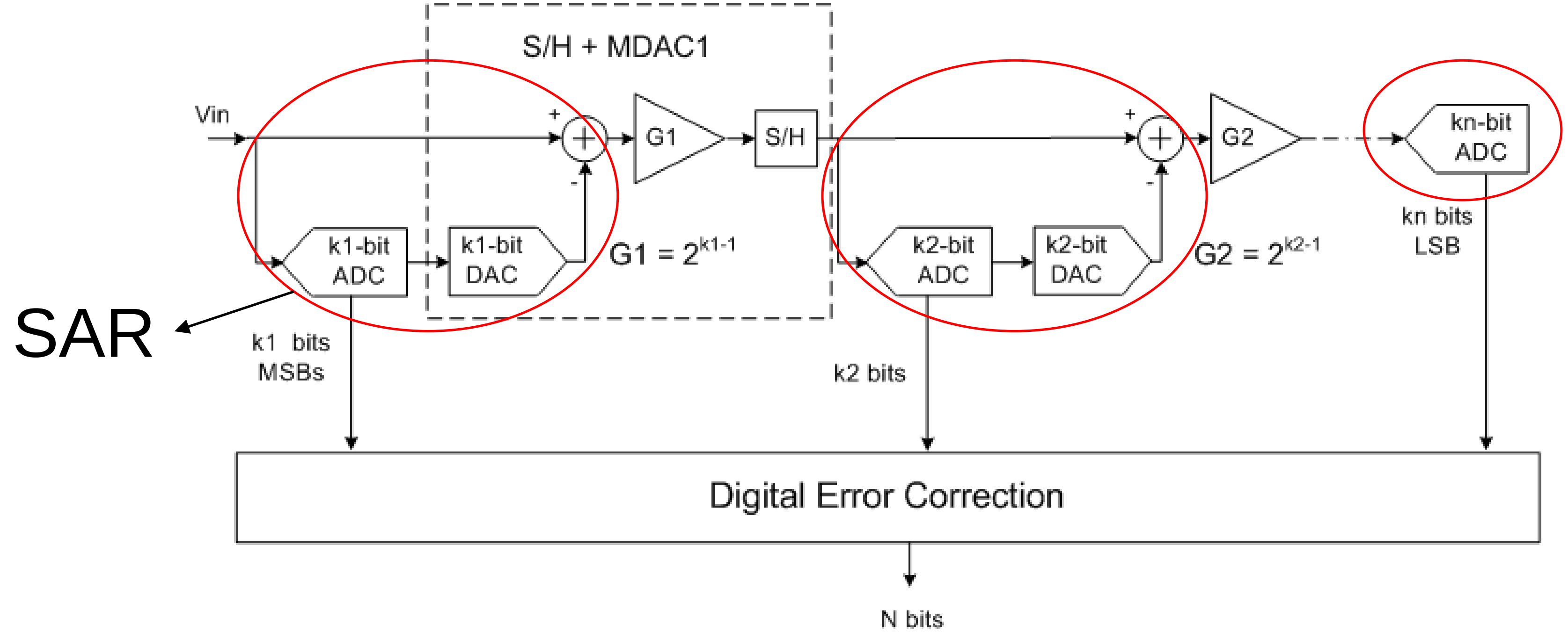
- The input S/H operation is sometimes performed by the first MDAC to save a dedicated SHA (S/H Amplifier)
- In this case, matching between the MDAC and flash is important

Pipelined A/D Converter



- The sub-ADCs were traditionally flash converters
- Power ~ 2N, so a small number of bits/stage was desirable
- Optimum for high resolution ADCs was 3-4 bit/stage
- For low resolution ADCs 1.5 bit/stage

Pipelined-SAR A/D Converter



- The sub-ADCs can be SAR ADC
- Power ~ N, so a large number of bits/stage is doable (6-7)
- Lower power and lower speed than flash-based pipeline ADC
- SARs perform the ADC, DAC and subtraction

Pipelined A/D Converters
Building Blocks

- S/H and MDAC
- Comparator
- Reference
- Clock
- Digital processor

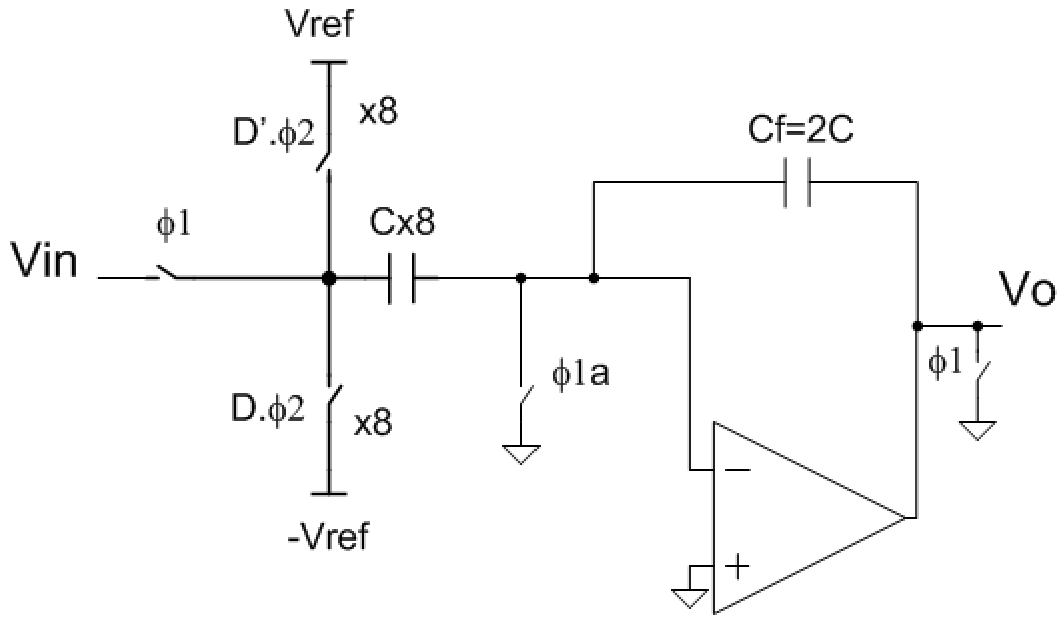
Switched Capacitor MDAC Operation

$$V_o = \frac{V_{in}.C_t / C_f}{1+(C_t + C_f + C_p)/(C_f.A)} - \sum_{i=1}^{8} \frac{\mathrm{D_i V_{ref}}.C/C_f}{1+(C_t + C_f + C_p)/(C_f.A)}$$

$$V_o = \frac{V_o \mid ideal}{1+(C_t + C_f + C_p)/(C_f.A)}$$

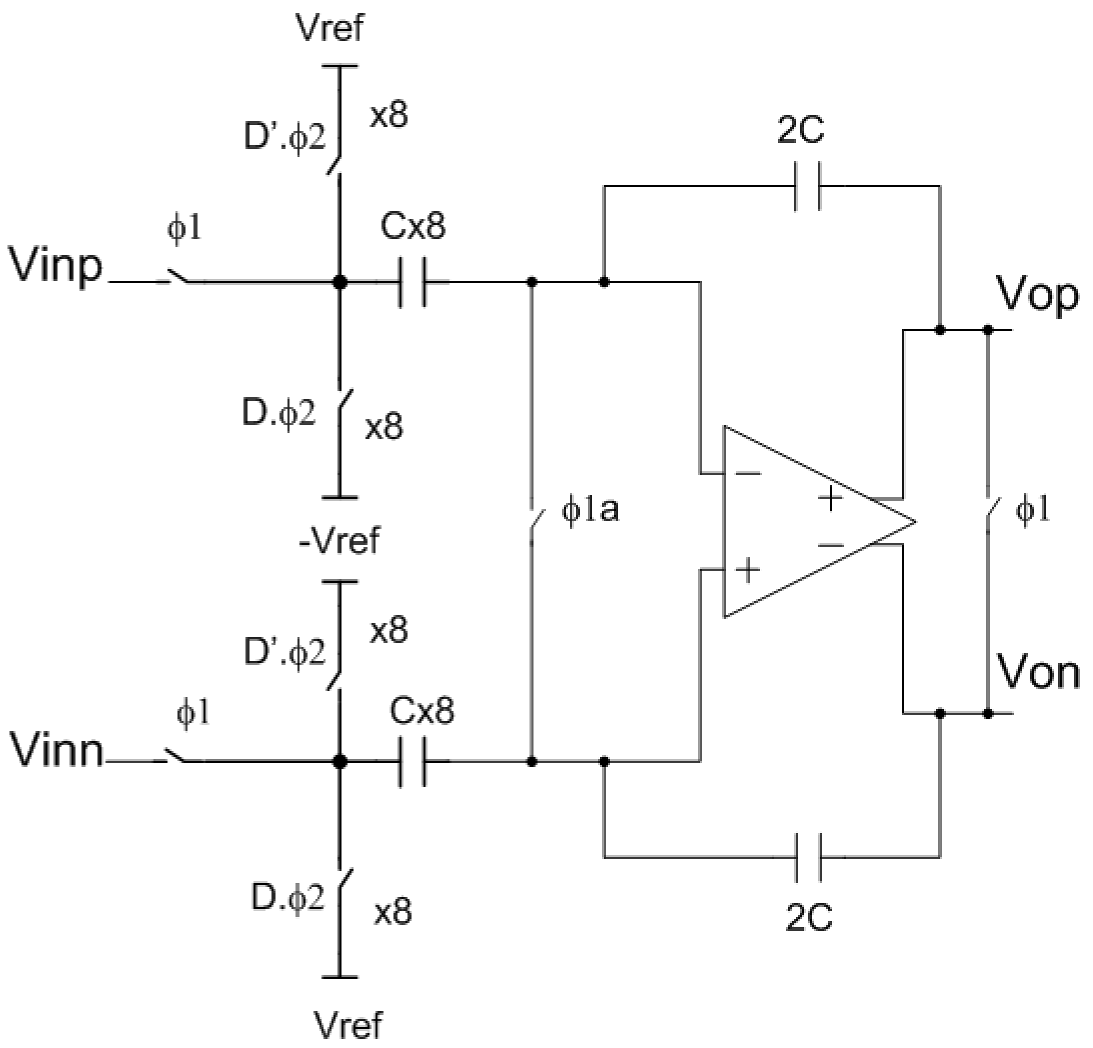
$$V_o = \frac{V_o \mid ideal}{1+K/A} = V_o \mid ideal \underbrace{- V_o K/A}_{\text{Error Term}}$$

$$\boxed{V_o \mid ideal = V_o + V_o K/A}$$



Switched Capacitor MDAC Operation

- Fully differential operation helps reduce even order harmonics and common mode sensitivity
- Input common mode variation is ideally rejected by this circuit
- Mismatches, common-mode variation in the circuit and the single-ended even-order non-linearity can still limit performance



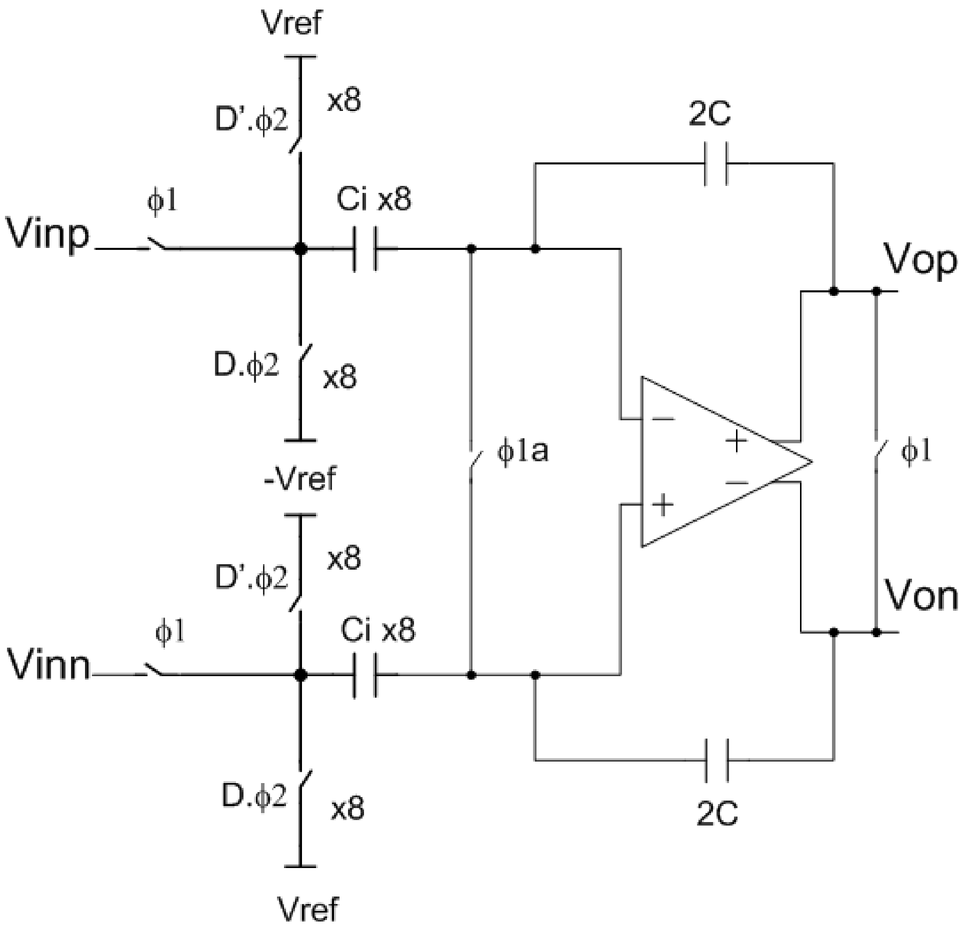
Accuracy Requirements

- The front stages need to have the highest accuracy
  - The accuracy requirement is relaxed as we go down the pipeline
- The stage ADC (flash) does not need to be very accurate
  - Usually 4-5 bit accurate
- The first DAC needs to be as accurate as the whole ADC
- The first amplifier needs to be as accurate as the back-end ADC
- The main bottle-neck is designing an MDAC that achieves the required speed (BW) and accuracy (gain)

Accuracy Requirements [1]

$$V_o \approx [\frac{V_{in}.C_t / C_f}{1+K/A} - \sum_{i=1}^{8} \frac{\mathrm{D}_i \mathrm{V}_{\mathrm{Refi}}.C_i / C_f}{1+K/A}](1-e^{-t_s\omega_t/K})$$

- If Ci ≠ Cj => DAC Errors
- If ΣCi/Cf ≠ G => Gain Error
- If Vrefi ≠ Vrefj => DAC Errors
- If Vref_stage1 ≠ Vref_stage2 => Gain Error
- Settling errors cause gain errors

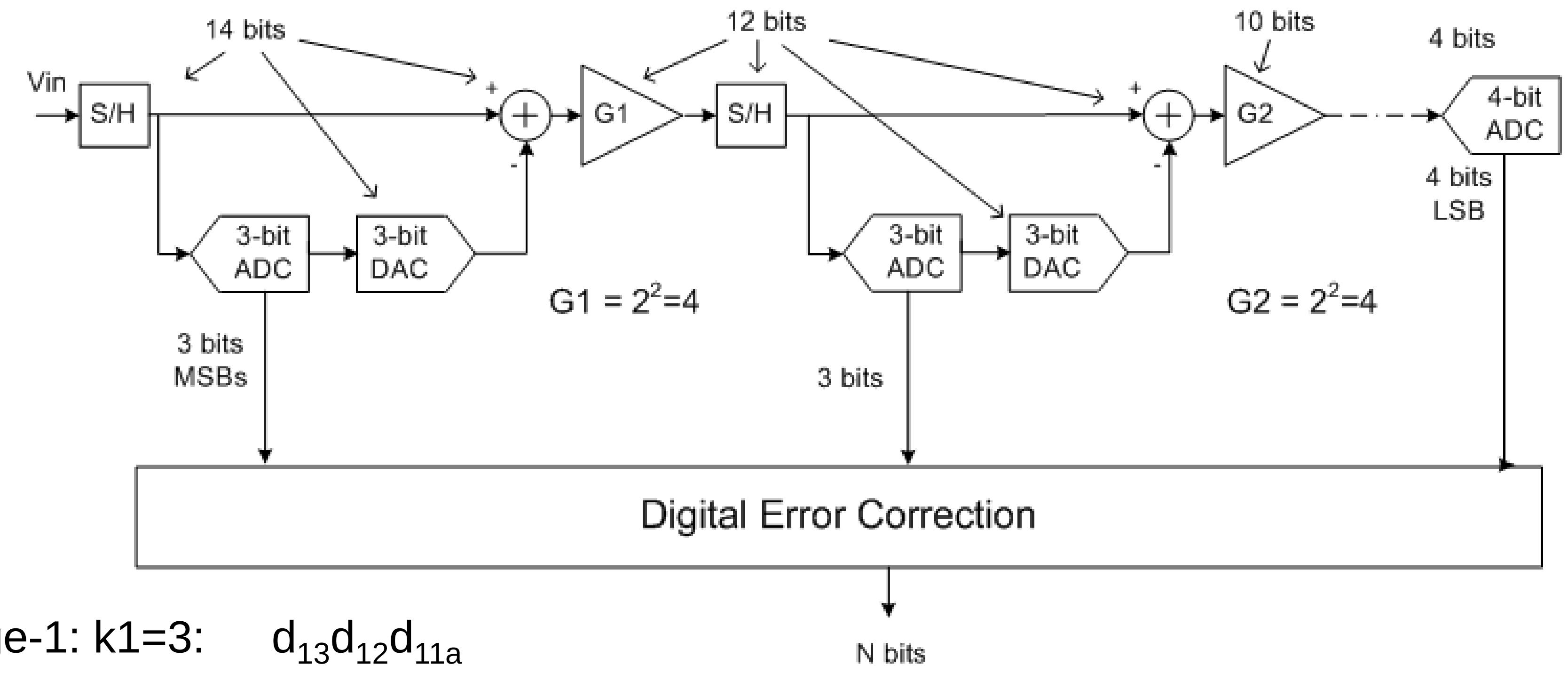


Ahmed Ali

Accuracy Requirements [1]

- **Capacitor matching:** $\Delta_{DAC_i} < 2^{-(N+j_{LS})} \times \prod_{n=0}^{i-1} G_n$
  - **Mismatches cause inter-stage gain and DAC errors**
- **Amplifier open loop gain:** $A_i > \frac{2^{-k_i} \times 2^{N+j}}{\beta_i \times \prod_{n=0}^{i-1} G_n}$
  - **Low gain causes inter-stage gain error and non-linearity**
- **Settling:** $\varepsilon_{DAC_i} < 2^{-(N+j_{LS})} \times \prod_{n=0}^{i-1} G_n$
  - **Settling errors cause inter-stage gain error, DAC errors and non-linearity**
- **Reference:** $\varepsilon_{ref_{DAC_i}} < 2^{-(N+j_{LS})} \times \prod_{n=0}^{i-1} G_n$
  - **Reference errors cause inter-stage gain and DAC errors**

Accuracy Requirements



Stage-1: k1=3: $d_{13}d_{12}d_{11a}$

Stage-2: k2=3: $d_{11b}d_{10}d_{9a}$

Stage-3: k3=3: $d_{9b}d_{8}d_{7a}$

Stage-4: k4=3: $d_{7b}d_{6}d_{5a}$

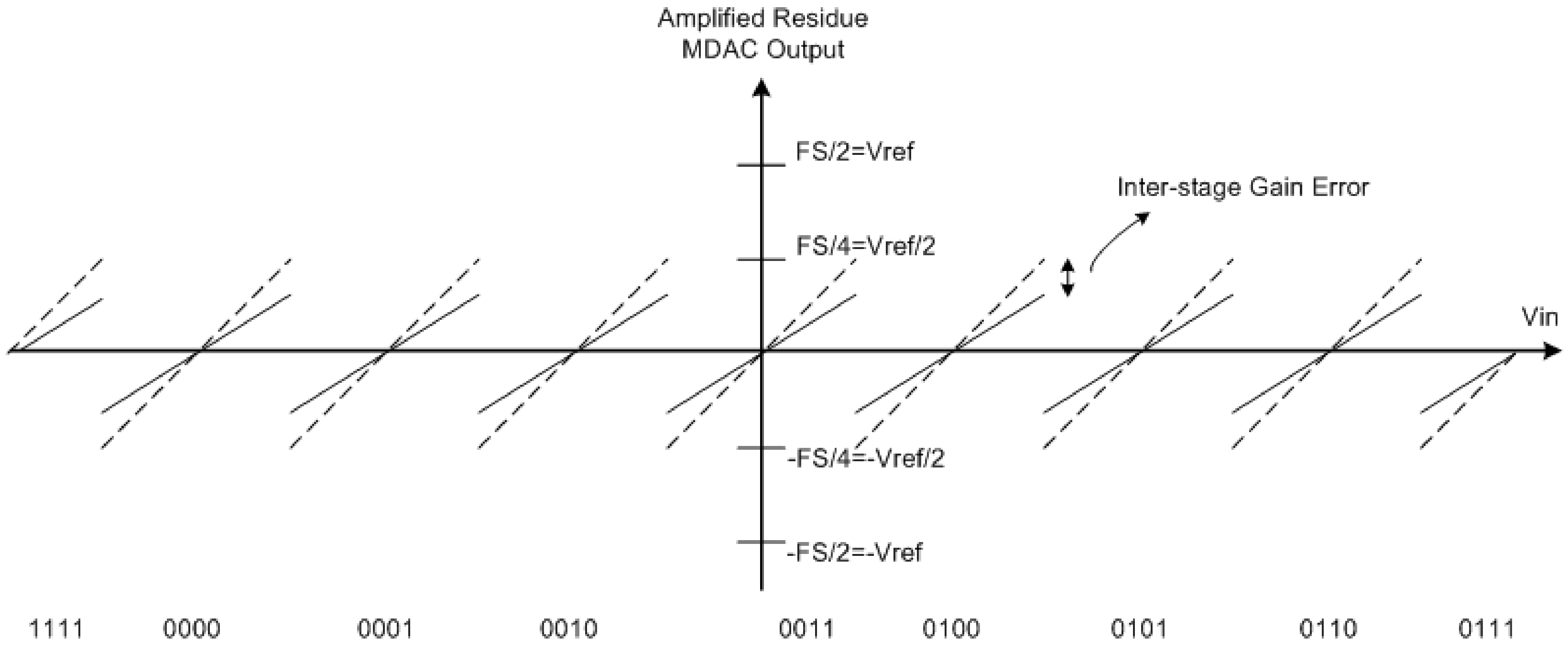
Stage-5: k5=3: $d_{5b}d_{4}d_{3a}$

Stage-6: k6=4: $d_{3b}d_{2}d_{1}d_{0}$

Output: 14-bits: $d_{13}d_{12}\ d_{11}\ d_{10}\ d_{9}\ d_{8}d_{7}\ d_{6}d_{5}\ d_{4}d_{3}\ d_{2}d_{1}d_{0}$

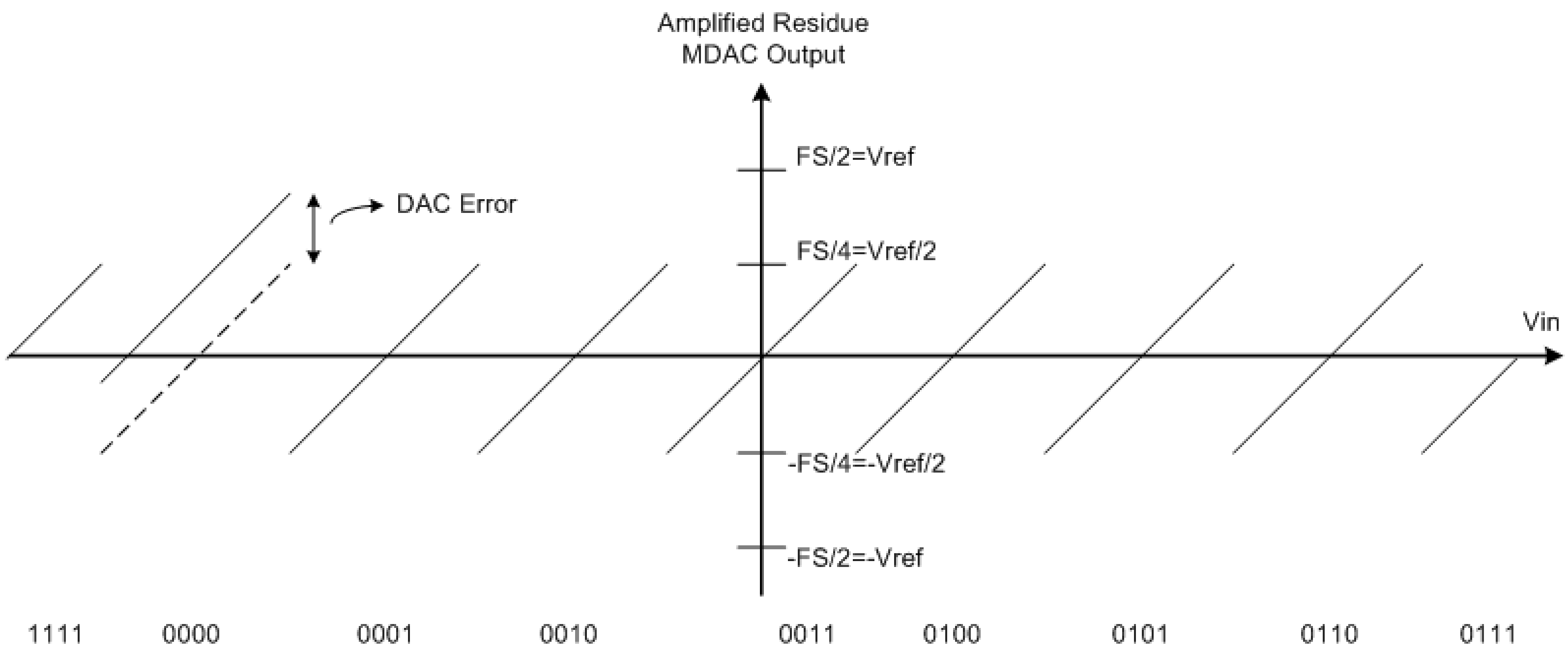
First Stage Residue
With Inter-Stage Gain Error

- **Common sources of IGE:**
  - **Cap mismatch**
  - **Low open loop gain**
  - **Reference error**
  - **Charge injection**
  - **Settling error**

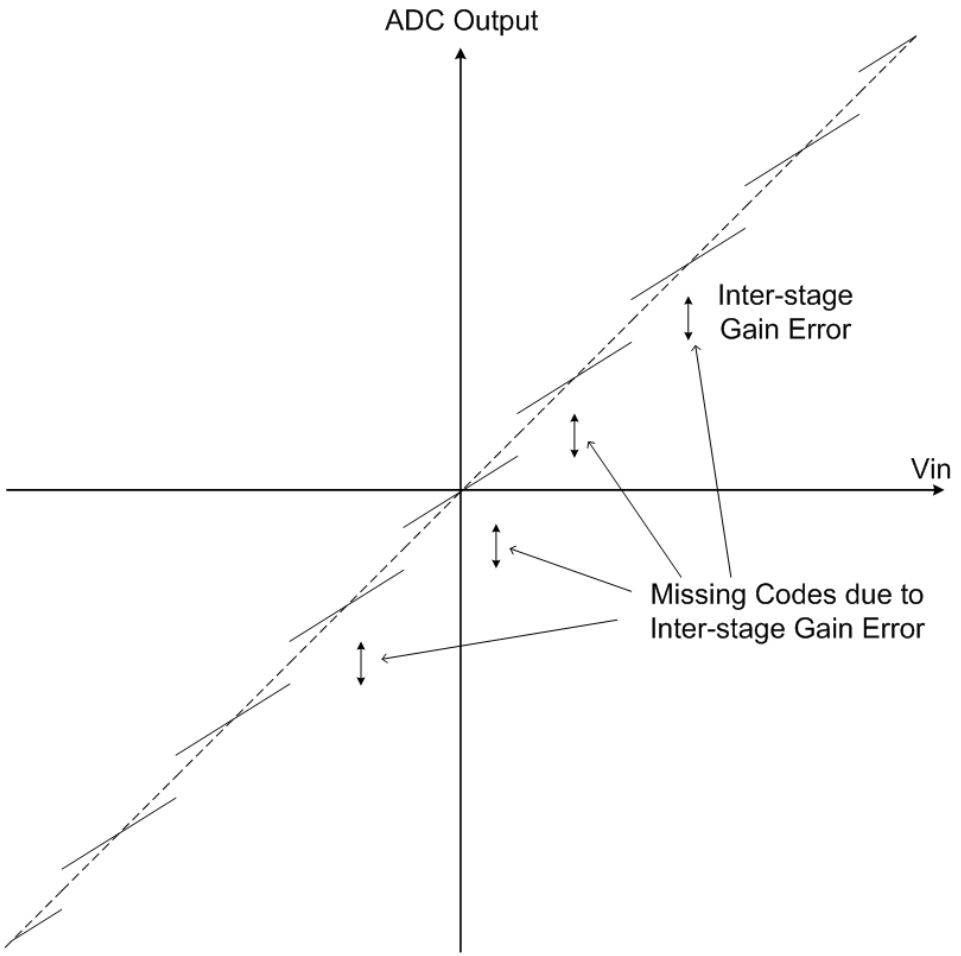


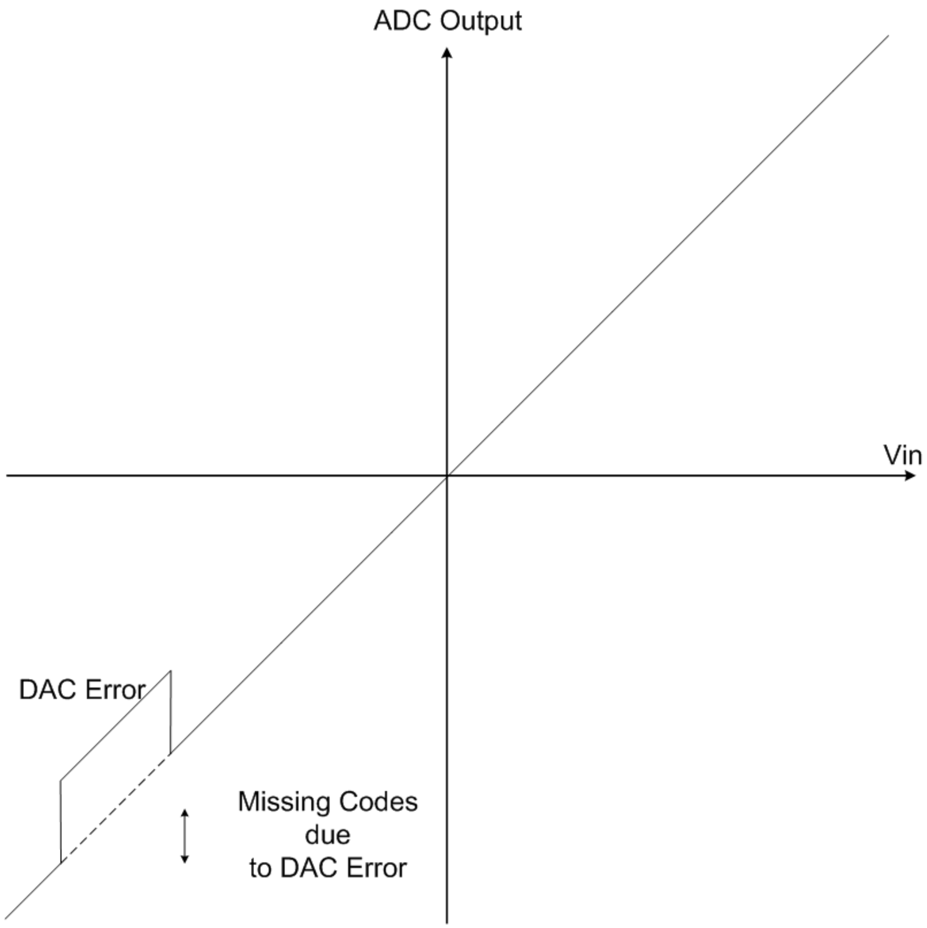
First Stage Residue With Stage DAC Error

- ◆ **Sources of DAC errors:**
  - **Cap mismatch**
  - **Reference error**
  - **Charge injection**
  - **Settling error**

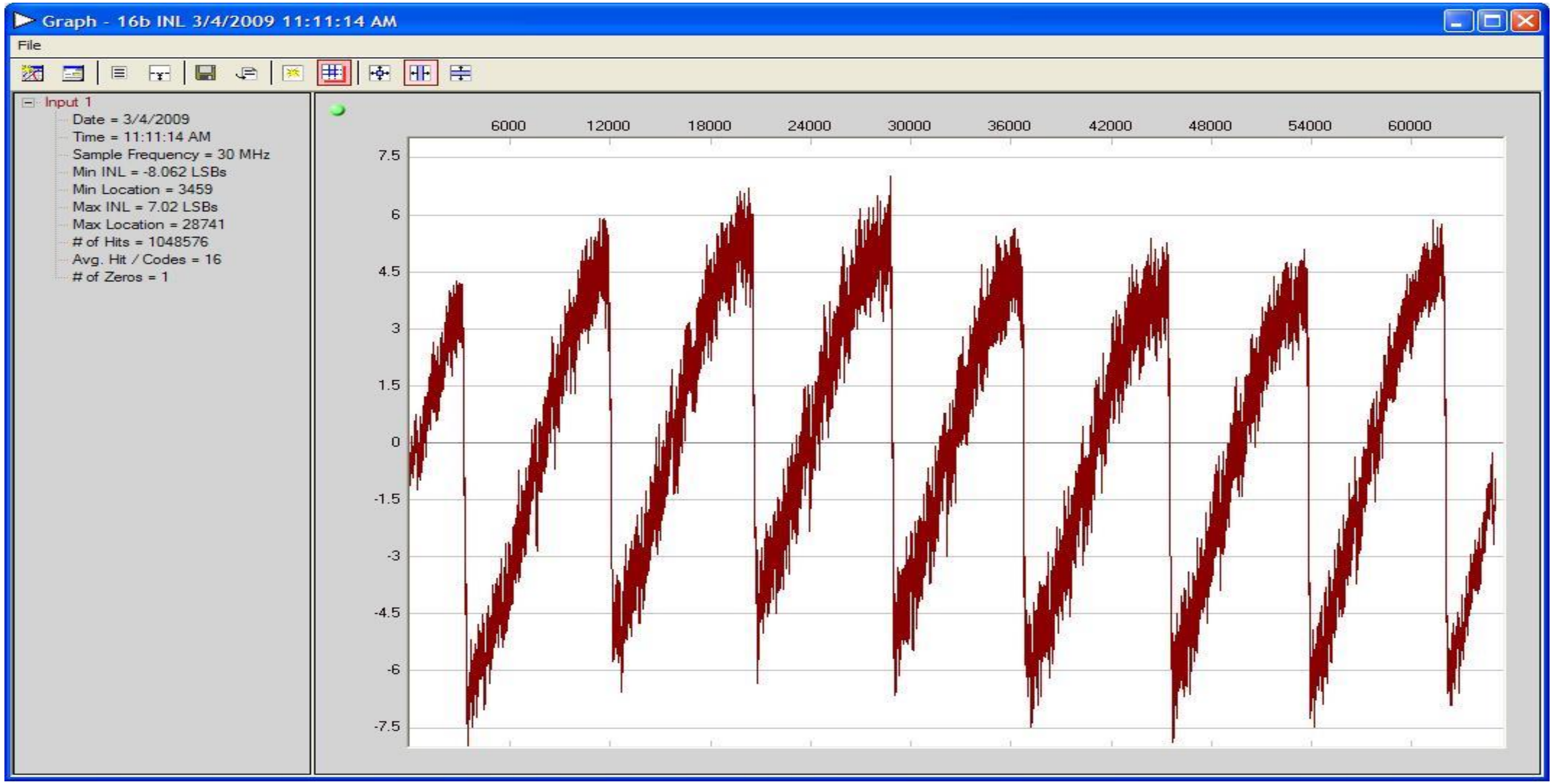


ADC Output
With Inter-Stage Gain Error and DAC Error



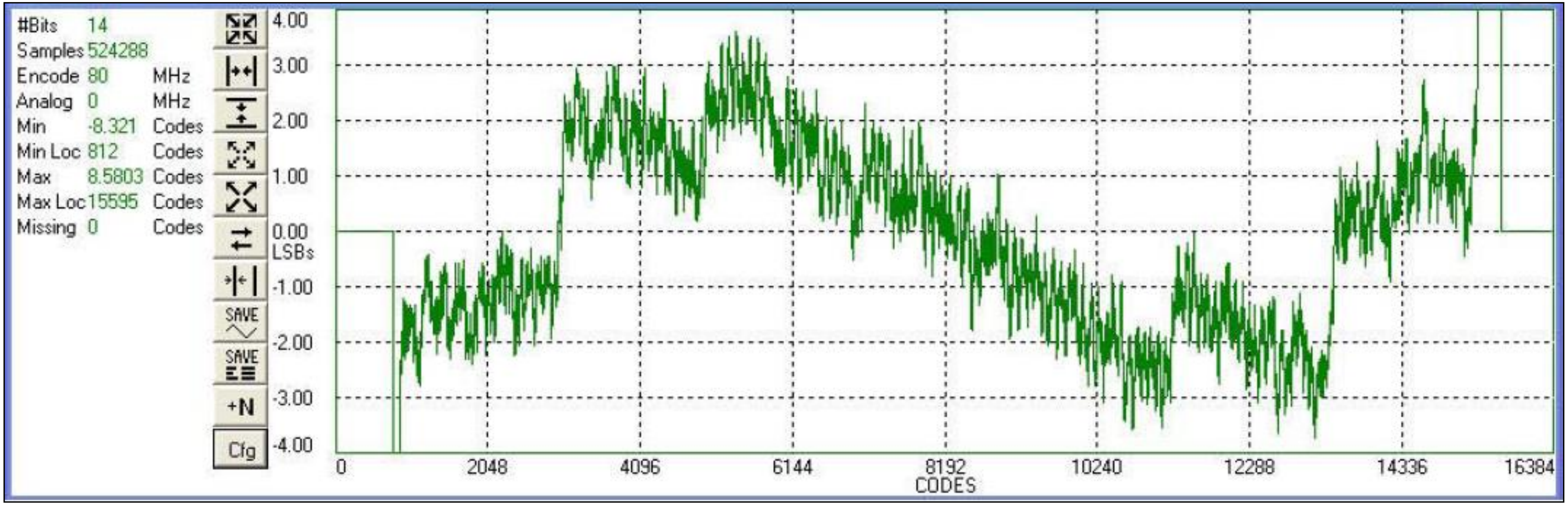


Example of INL with Inter-stage Gain Errors in the First Stage



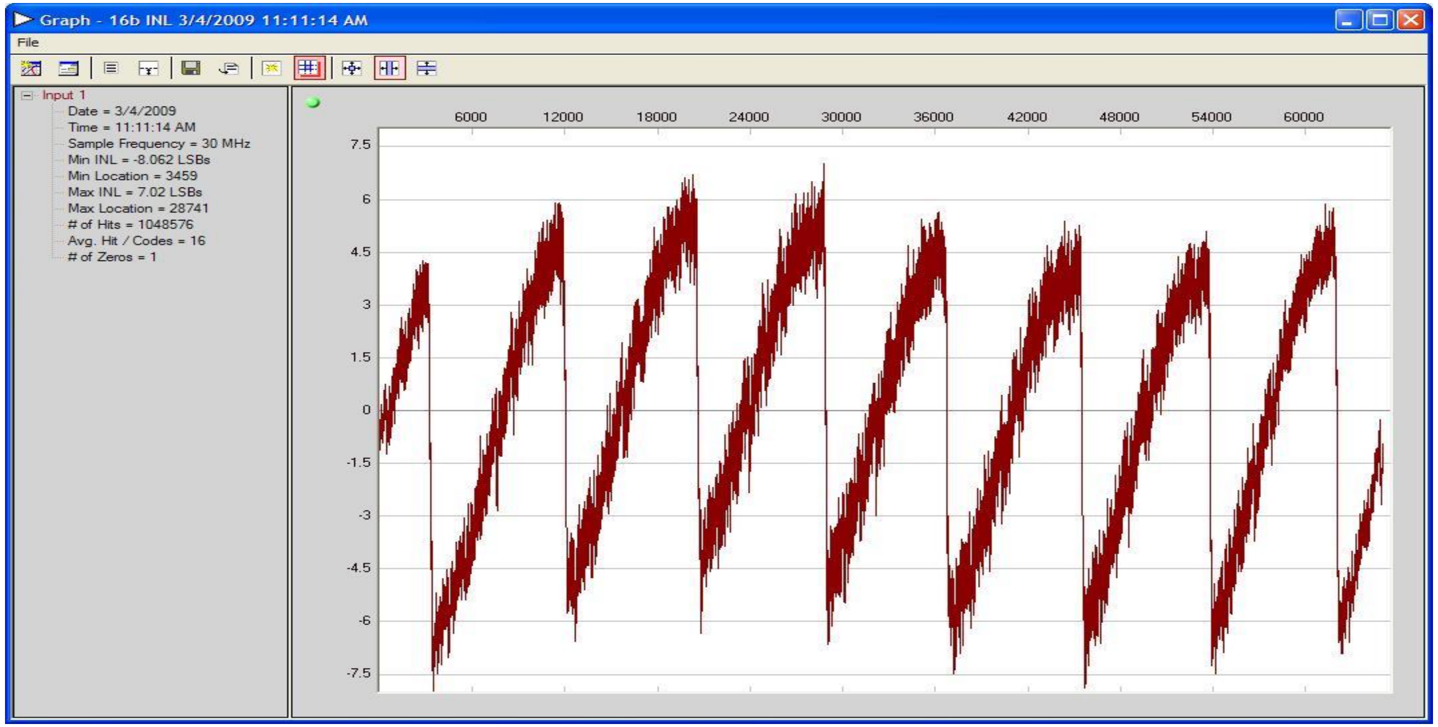
Ahmed Ali

INL Showing Inter-stage Gain Error and DAC Errors

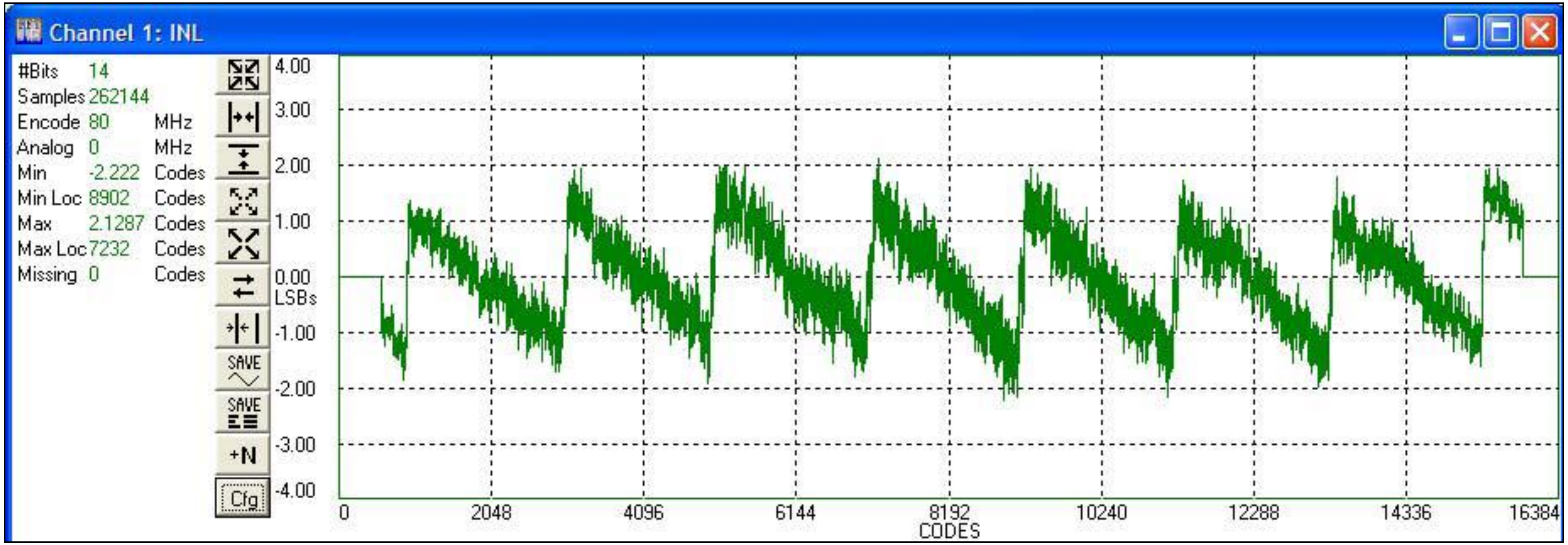


Questions

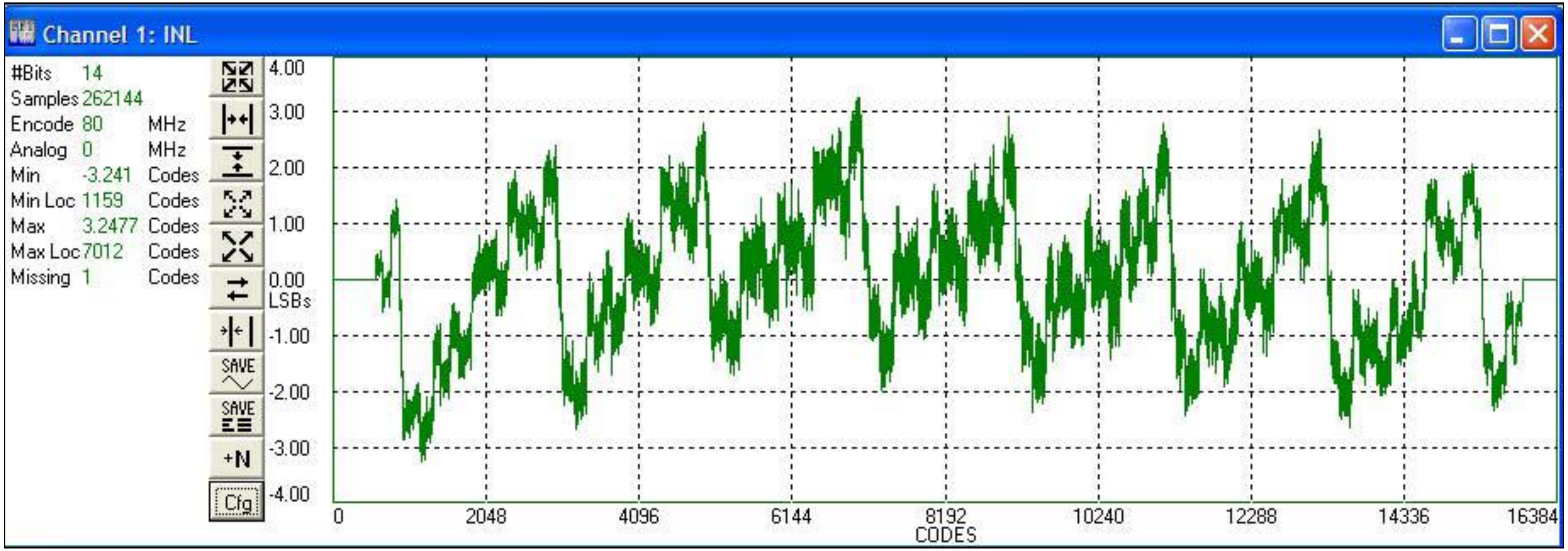
- Does this INL correspond to a small or large inter-stage gain? Missing codes or not?



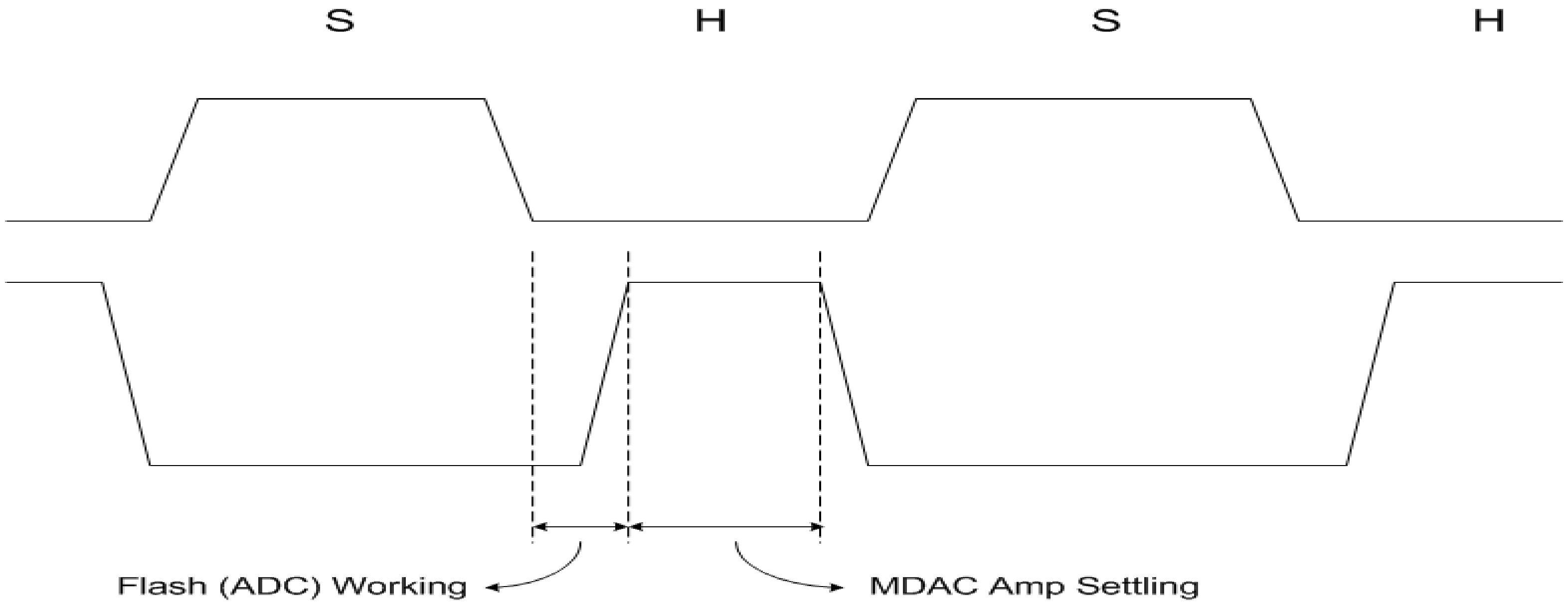
What kind of Errors?



What kind of Errors?



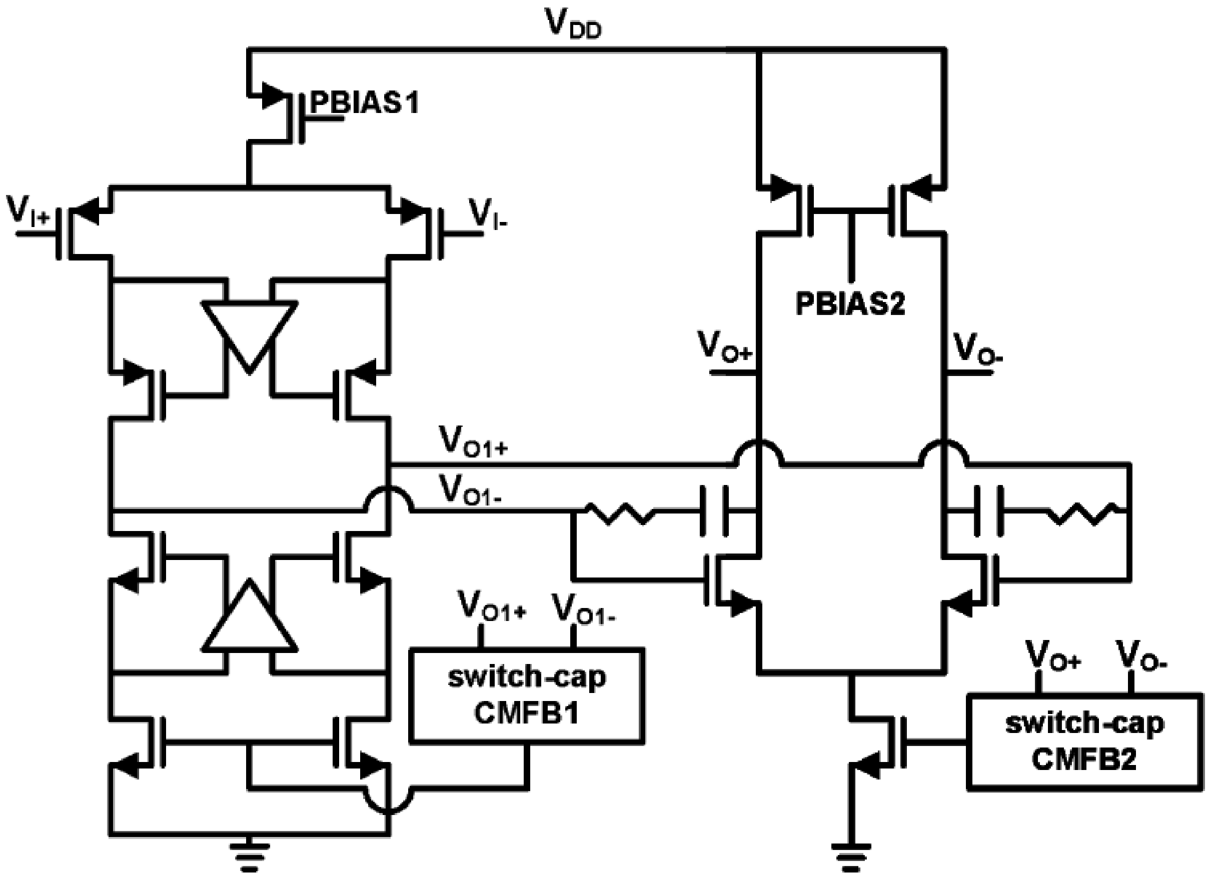
Stage Timing



- The stage ADC usually operates on the input during the same phase as the MDAC generates the output
  - The flash delay consumes portion of the hold/gain phase
  - The MDAC amplifier settling time is the hold phase minus the time consumed by the flash and switches

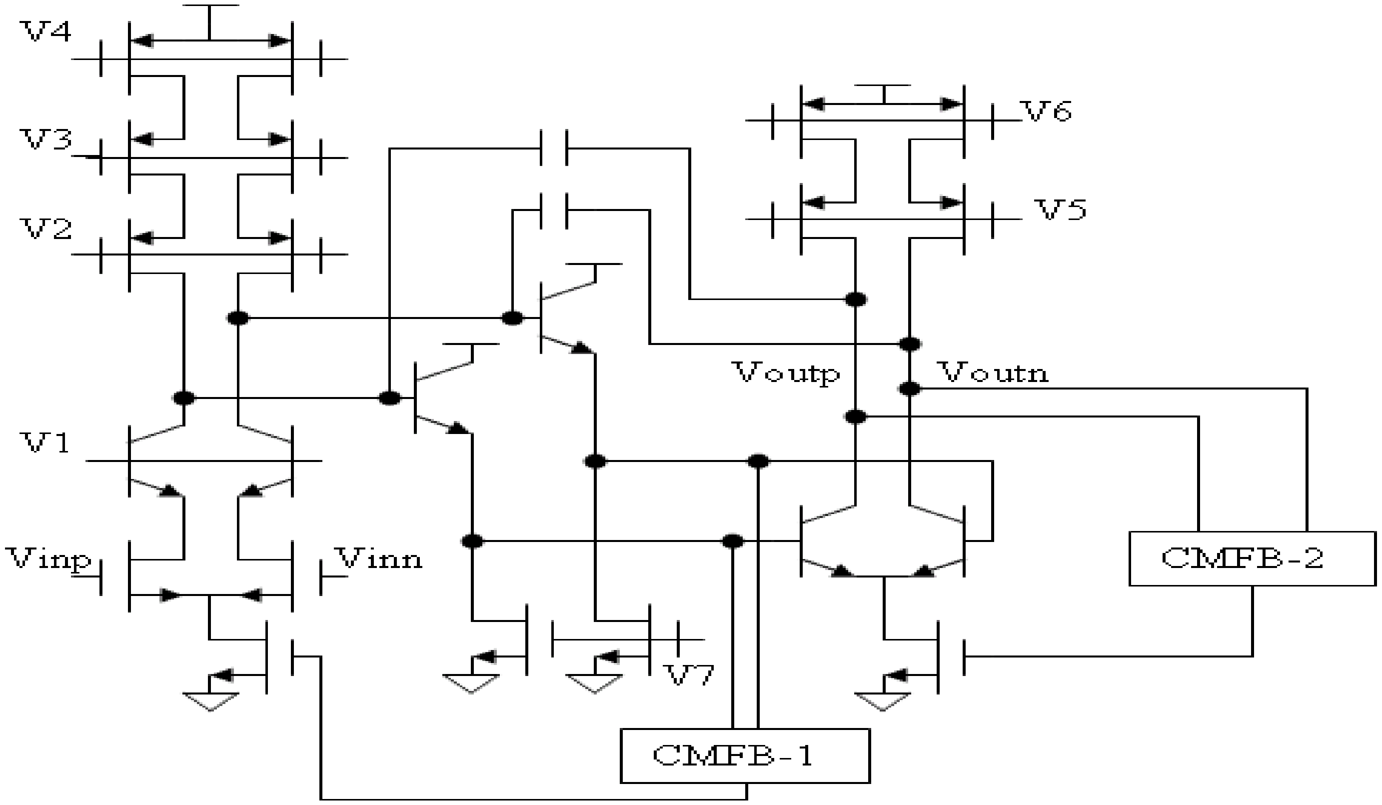
MDAC Amplifier Example [1, 4]

- Consists of two-stage Miller-compensated amplifiers
- The first stage is active cascoded
- Two independent common-mode feedback loops are used to control the output common-mode voltages of the two stages
- Can achieve high gain



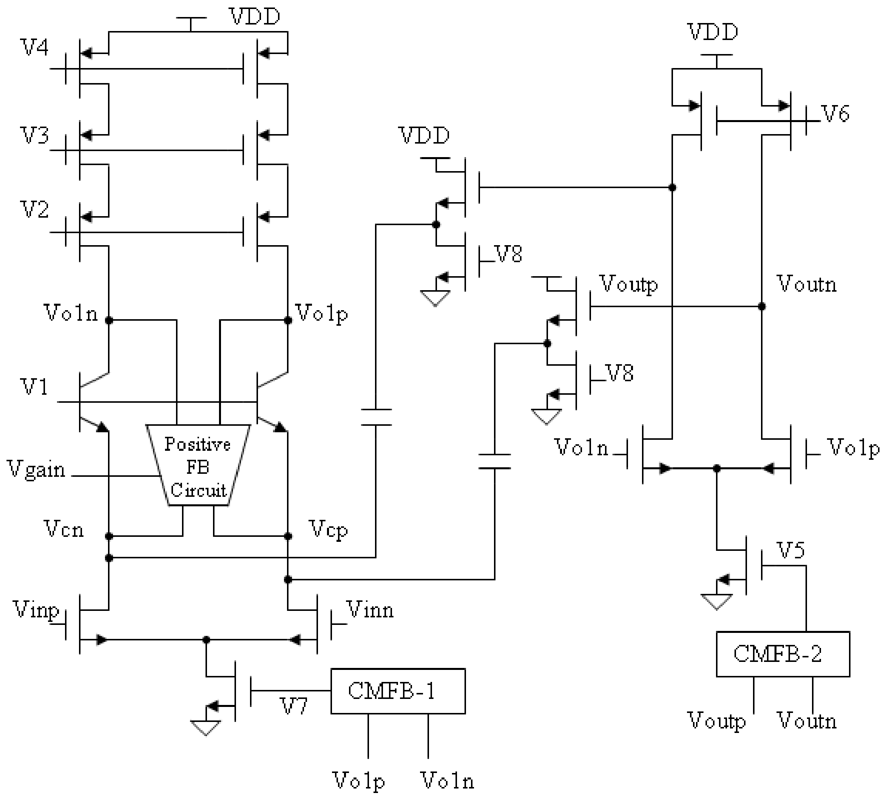
BiCMOS MDAC Amplifier Example [1]

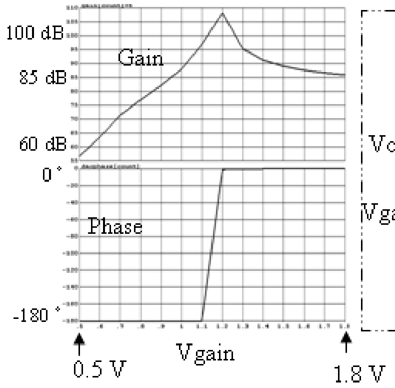
- Consists of two-stage Miller-compensated BiCMOS cascode amplifiers
- The two stages are separated by emitter followers
- Two independent common-mode feedback loops are used to control the output common-mode voltages of the two stages

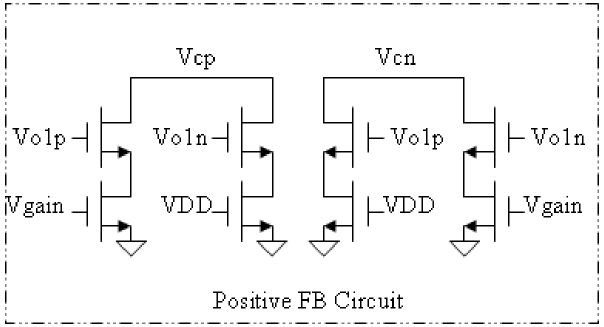


MDAC Amplifier Example [3]

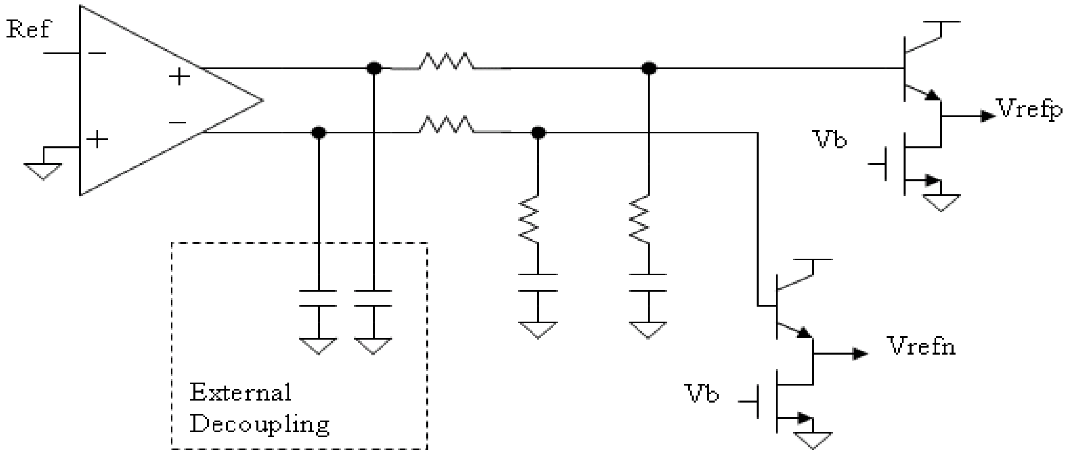
- Consists of two-stage Miller-compensated amplifiers
- The first stage is cascoded
- Two independent common-mode feedback loops are used to control the output common-mode voltages of the two stages
- Positive feedback is used to increase the gain







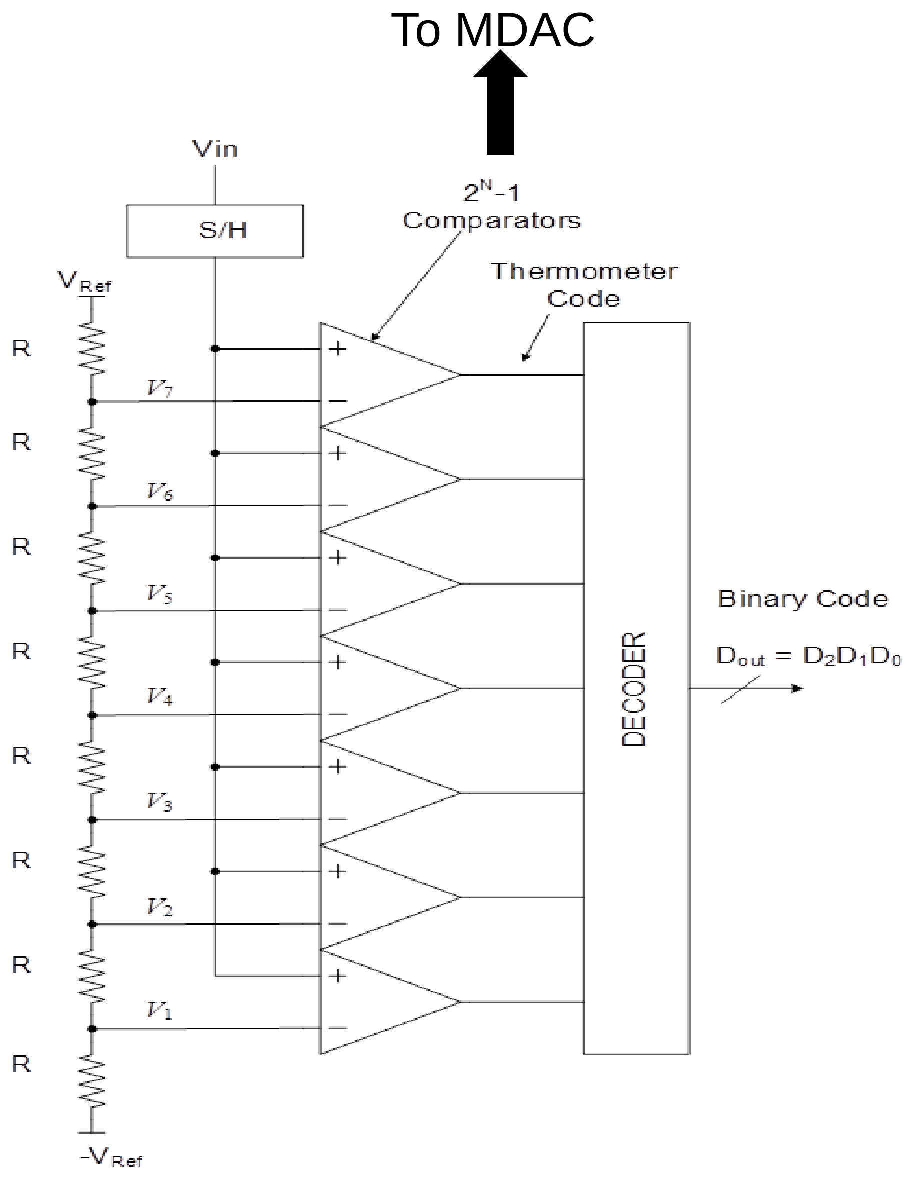
Reference Buffer Example



- **Fast settling:**
  - **Settling errors cause inter-stage gain errors and DAC errors**
- **Good supply and ground rejection**
- **Good isolation from the input signal**
- **Good common-mode rejection**
- **Quiet supplies and grounds**

Stage Flash ADC

- Generates thermometer and binary codes
- The thermometer code (or a variation of it) is routed to the MDAC
- The binary (or 2's complement) output is routed to the digital processor

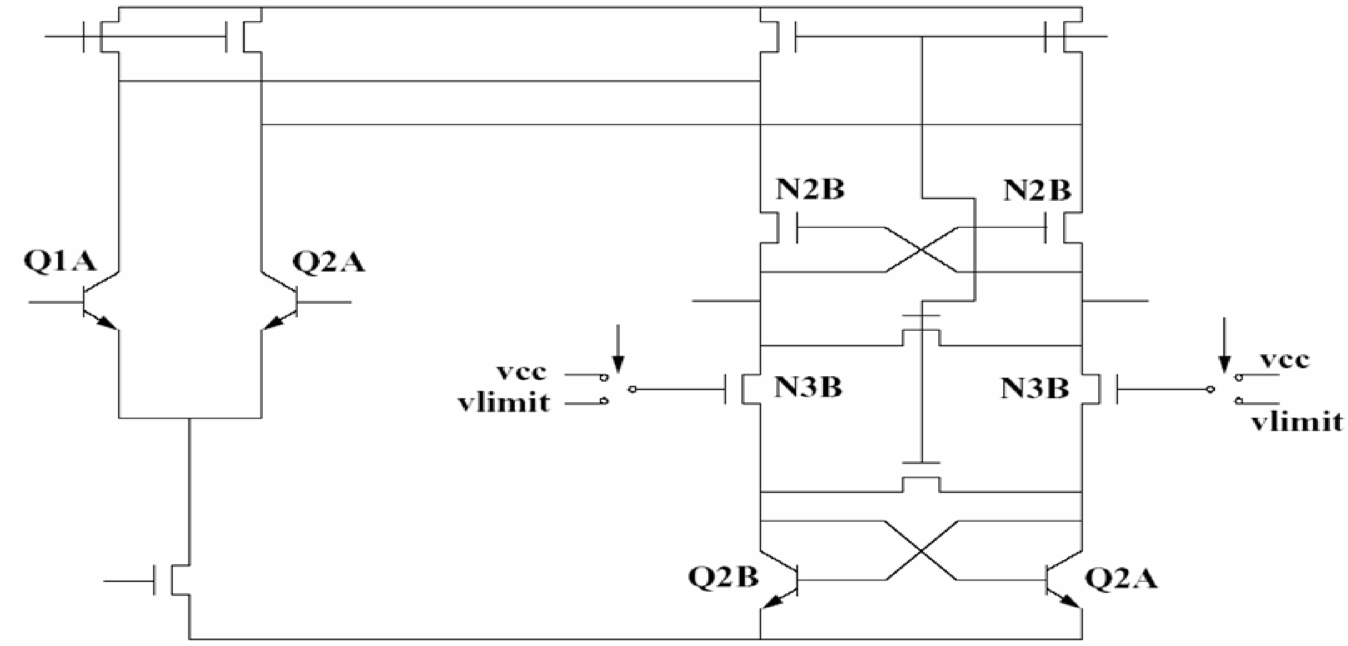


Comparator Design Considerations

- **Output propagation delay**
- **Meta-stability (regeneration time constant) determines the Bit-Error-Rate**
- **Offset needs to be within the correction range of the MDAC**
  - **The smaller the better**
- **Linearity is not critical**
- **Flash input BW needs to match the MDAC input BW in SHA-less ADCs**

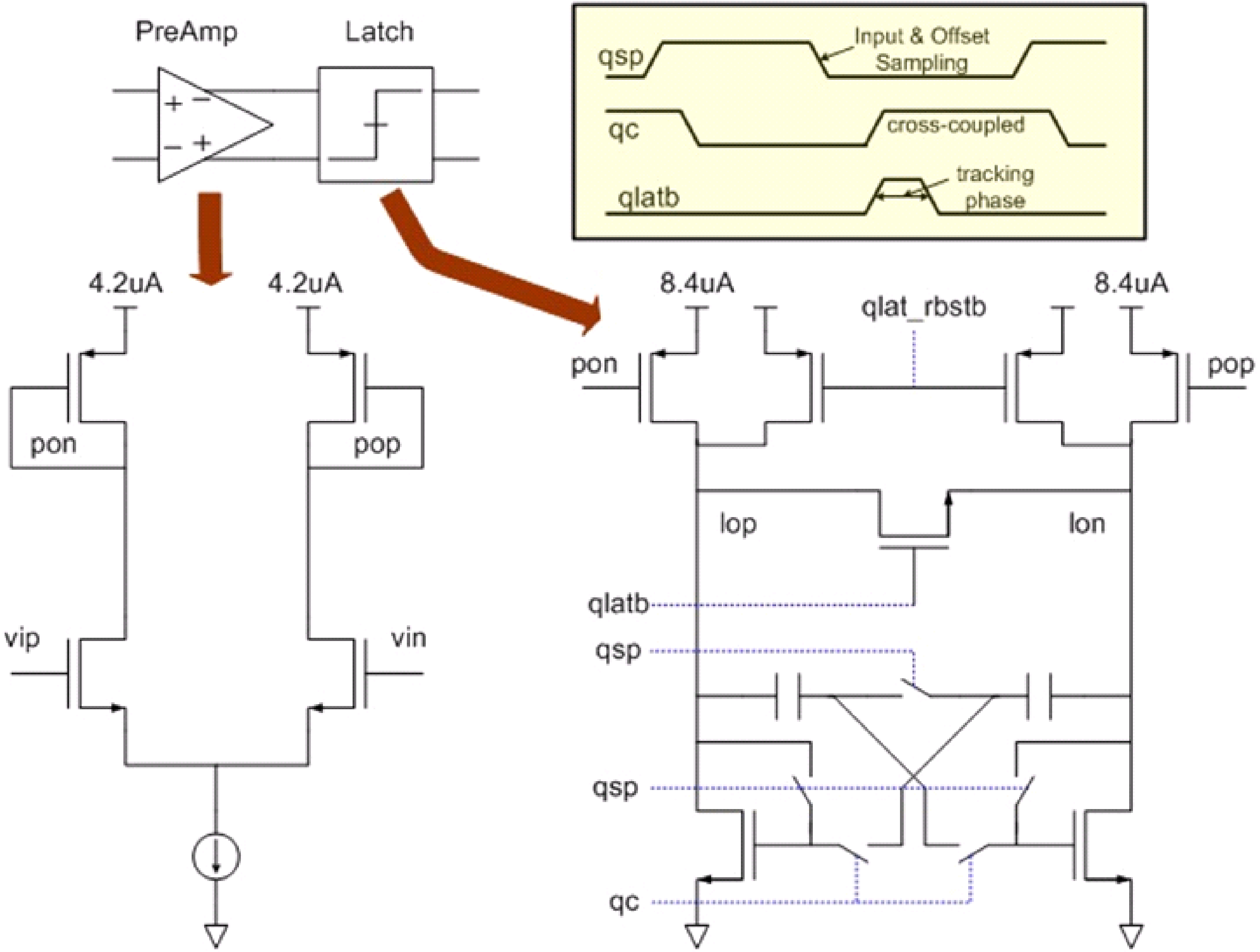
BiCMOS Comparator

- BJT's are used for their high ft
- This leads to small re-generation time and hence good met-stability
- The pre-amplifier is used for isolation and to improve the offset



CMOS Comparator

- Relatively low ft
- Offset cancellation is employed
- The pre-amplifier is used for isolation and to improve the offset



DESIGN CHALLENGES

Performance Limitations
Design Challenges

- Quantization Linearity
- Input Sampling
- Noise
- Jitter

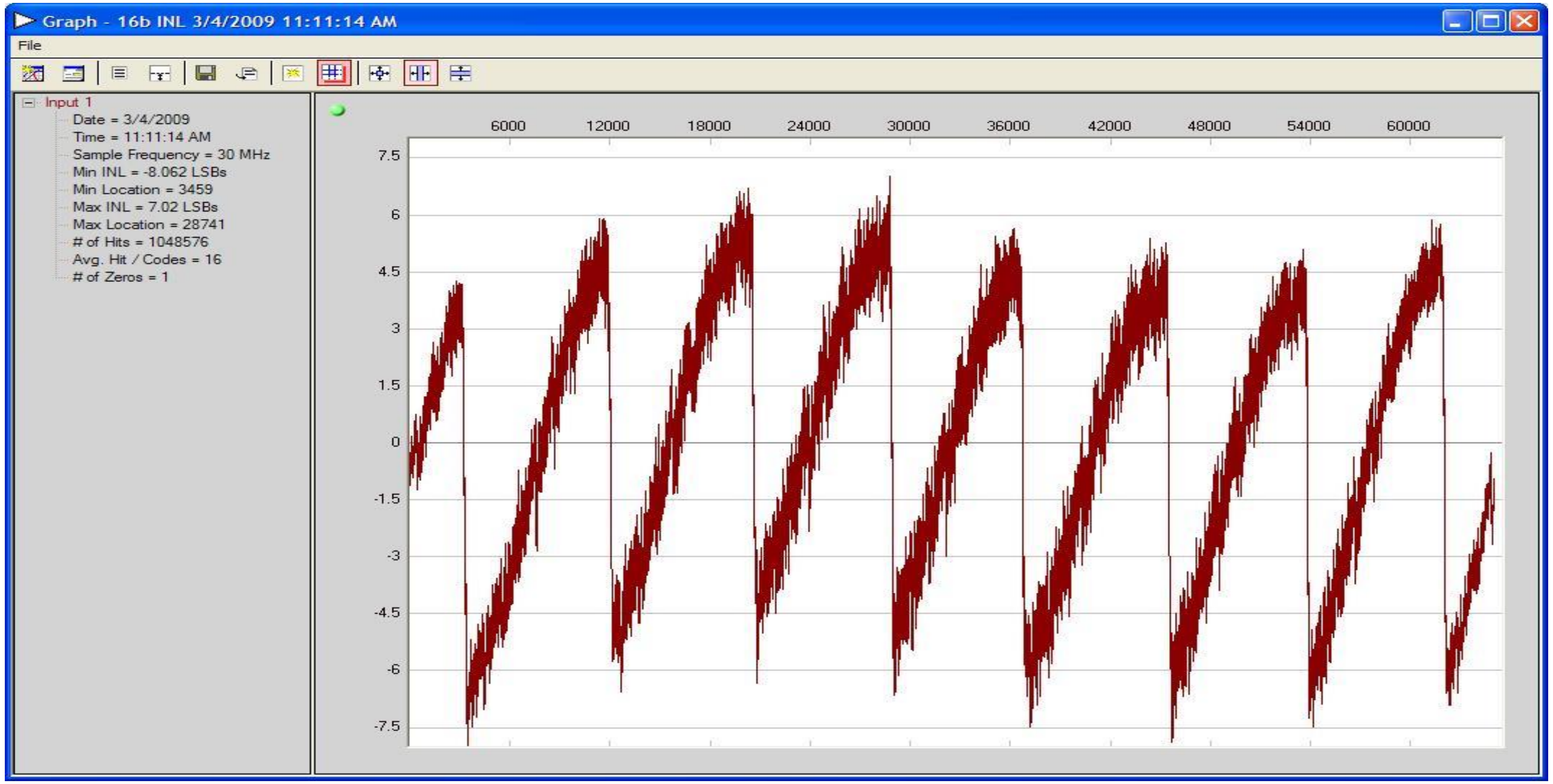
Internal Linearity

- **DAC accuracy:**
  - **Capacitor matching**
  - **Settling**
  - **Charge injection**

- **Amplifier accuracy:**
  - **Gain**
  - **Settling**
  - **Charge injection**

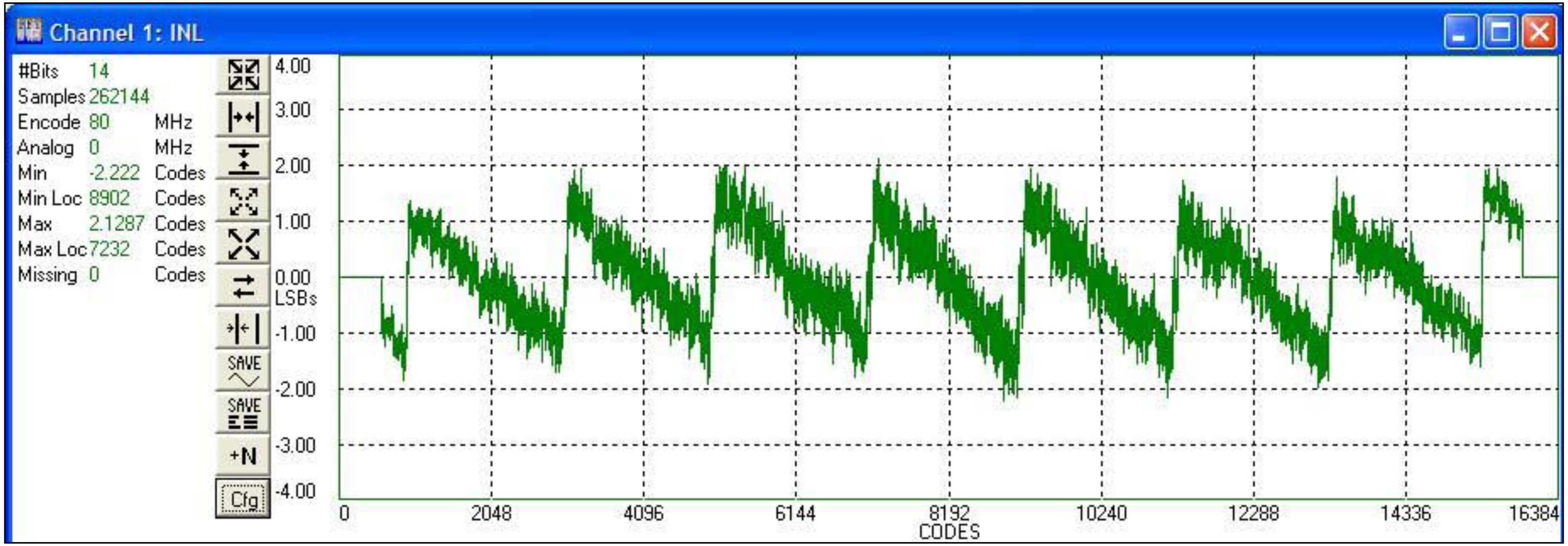
- **Reference accuracy:**
  - **Settling**

Question: What are the possible circuit causes of this INL?



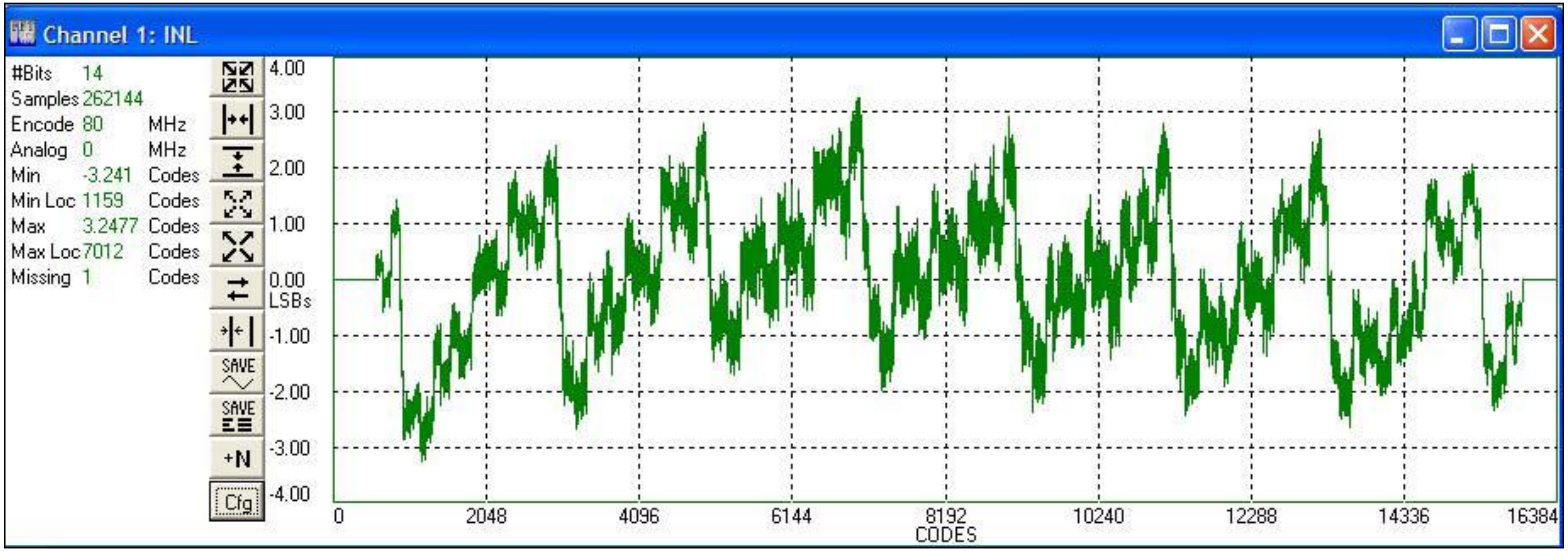
Ahmed Ali

Question: What are the possible circuit causes of this INL?

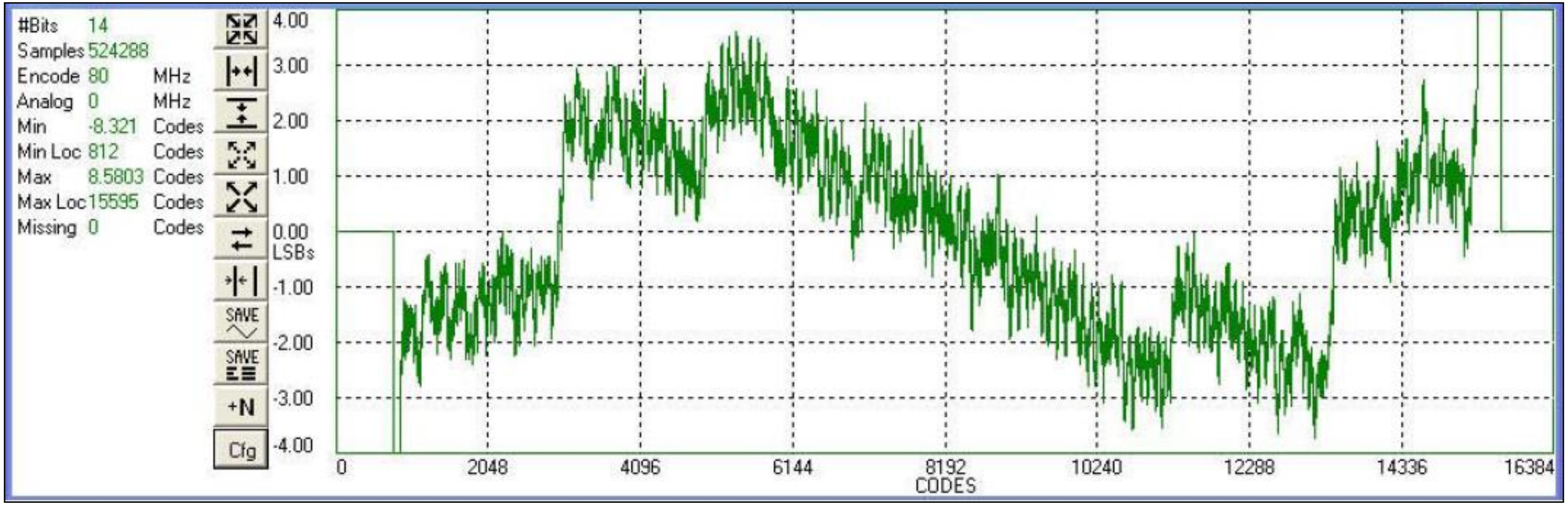


Ahmed Ali

Question: What are the possible circuit causes of this INL?



Question: What are the possible circuit causes of this INL?



Question

- A 14-bit pipelined ADC with 4 bits/stage. What is the accuracy requirement for the following?
  - First stage DAC
  - First stage amplifier
  - First Stage ADC
  - Second stage DAC
  - Second stage amplifier
  - Second stage ADC

NOISE

Sources of Noise

- Thermal noise in the signal path
  - Includes noise during both phases (sample and hold)
  - Noise contribution is lower for stages down the pipeline
  - Scaling is usually employed to lower power consumption of back-end stages
- Noise due to non-linearity (DNL/INL)
- Noise in the clock path (Jitter)
- Noise due to quantization

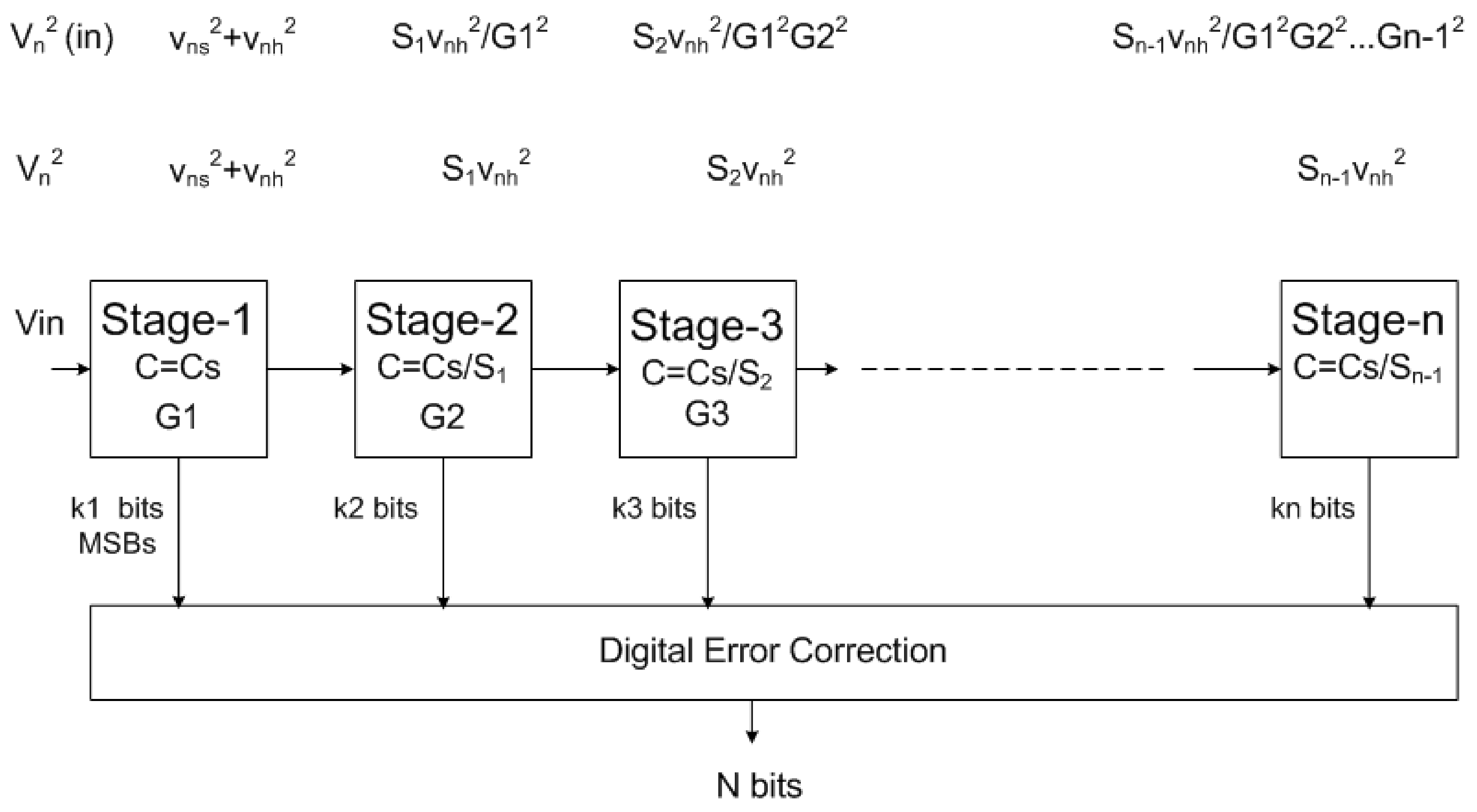
Thermal Noise

- **Sampling noise is proportional to KT/Cs**
  - **Where Cs is the sampling capacitor**

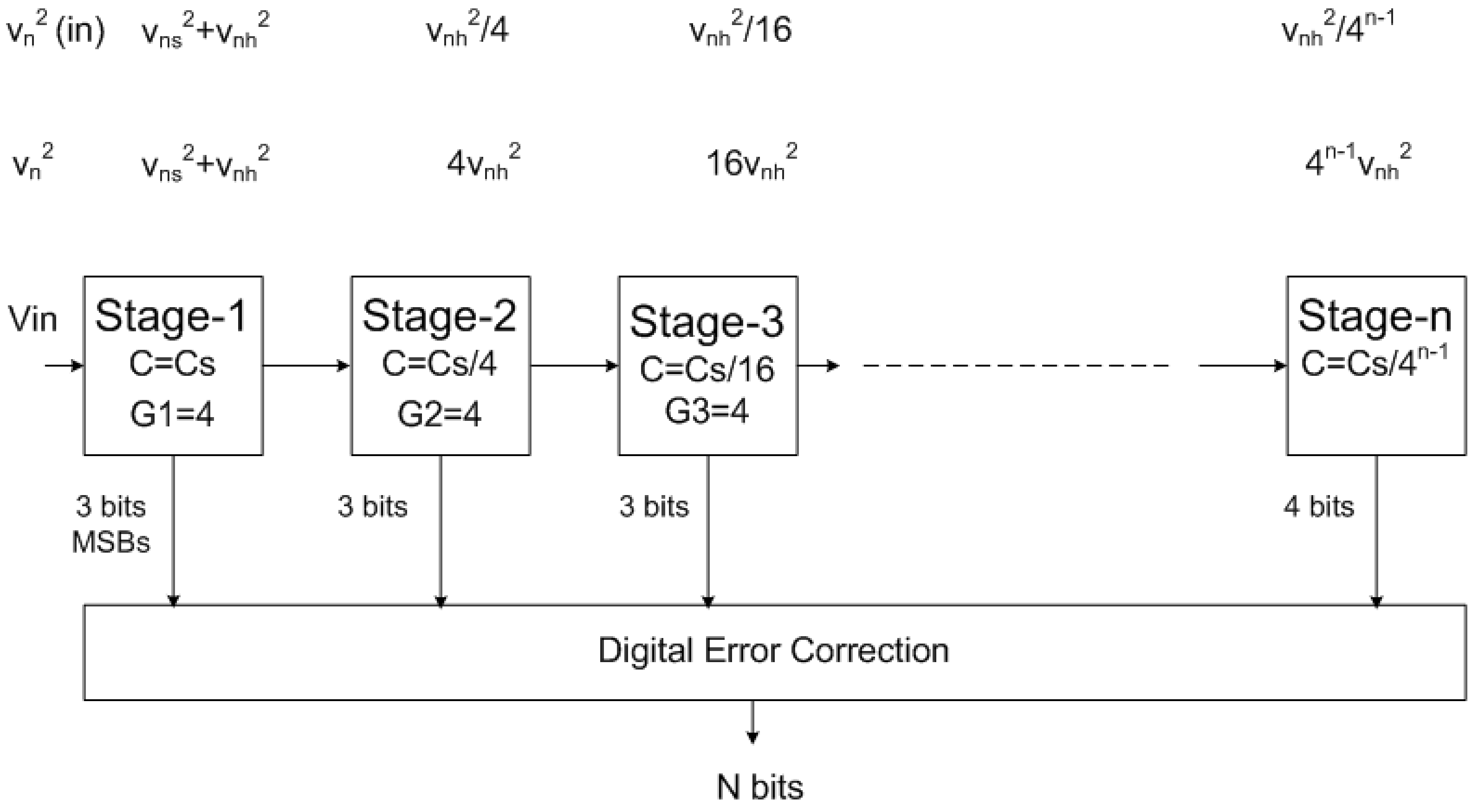
- **If the gain of each stage is G, the noise power of the kth stage is scaled down by G2(k-1) when referred to the input**

- **If a scaling factor S is employed for the capacitor of each stage:**
  - **The noise power of the kth stage, at the stage (not input referred) is scaled up by S**
  - **When referred to the input. The noise will be scaled by:**
    - **S(k-1)/ G2(k-1)**

Stage Scaling



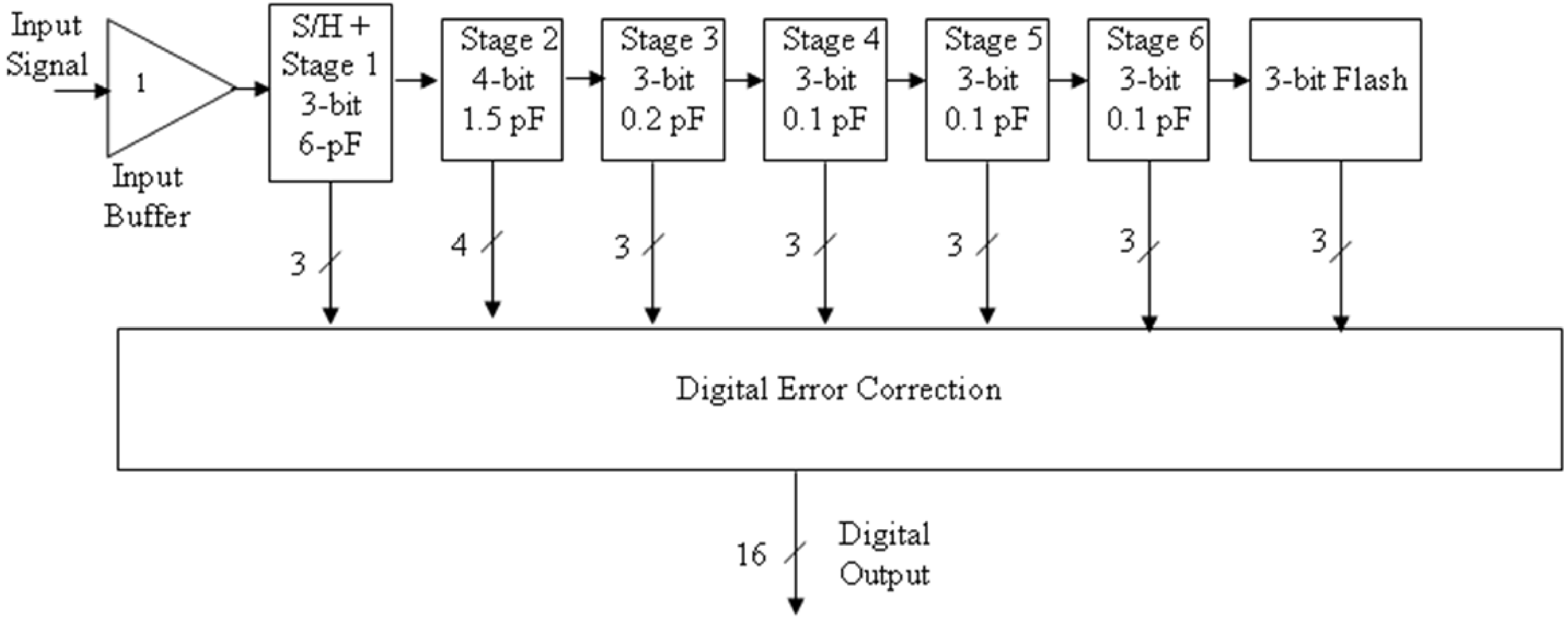
Stage Scaling



$$v_n^2(in) = v_{ns}^2 + v_{na}^2\left(1 + 1/4 + 1/16 + \ldots\right)$$

Ahmed Ali

Example [3]



Input
Signal
1
Input
Buffer
S/H +
Stage 1
3-bit
6-pF
Stage 2
4-bit
1.5 pF
Stage 3
3-bit
0.2 pF
Stage 4
3-bit
0.1 pF
Stage 5
3-bit
0.1 pF
Stage 6
3-bit
0.1 pF
3-bit Flash
3
4
3
3
3
3
3
Digital Error Correction
16
Digital
Output

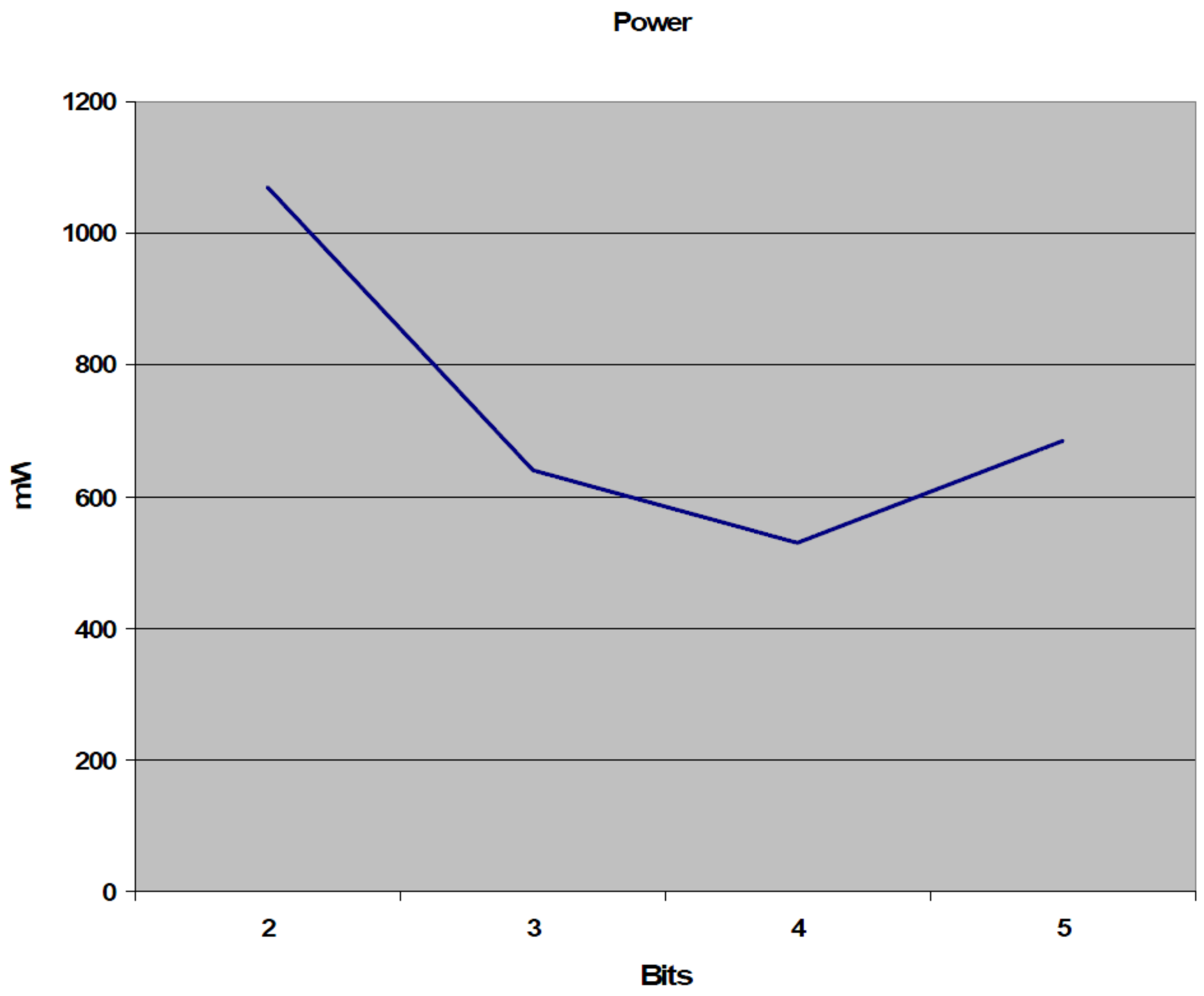
System Level Design Considerations

- Sampling Capacitor Value
- Number of Bits/Stage
- Input Buffer or Not
- Input Span
- Scaling factor

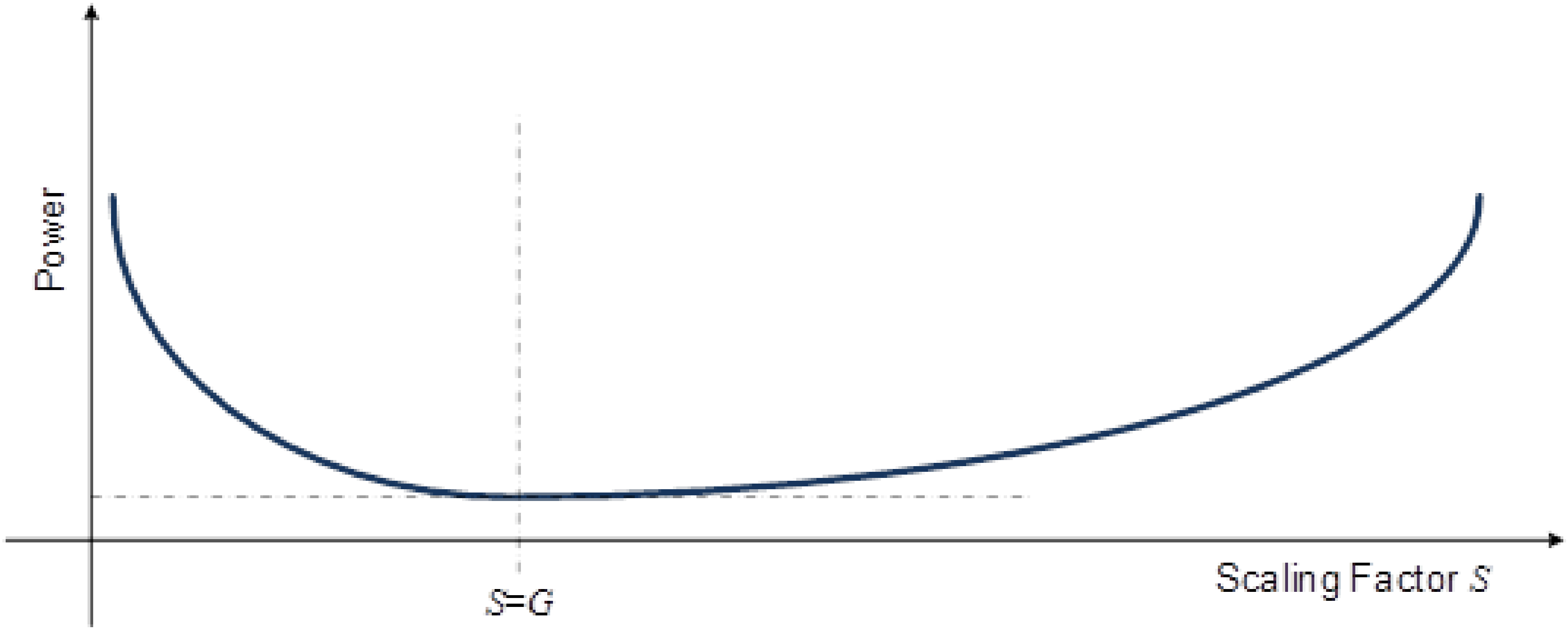
Number of Bits per Stage

- Fewer bits per stage:
  - Fewer comparators => lower power in the flash
  - Exploit finite gain-bandwidth of the amplifier:
    - Lower amplifier gain => higher speed
  - Less complex
  - With 1.5 bit/stage, DAC is inherently linear
- More bits per stage:
  - Quickly relaxes the noise and accuracy requirements of the following stages
  - The MDAC is more power efficient
- Conclusion:
  - There is an optimum resolution per stage to minimize power consumption
  - Sometimes the optimum resolution is not feasible => use lower resolution

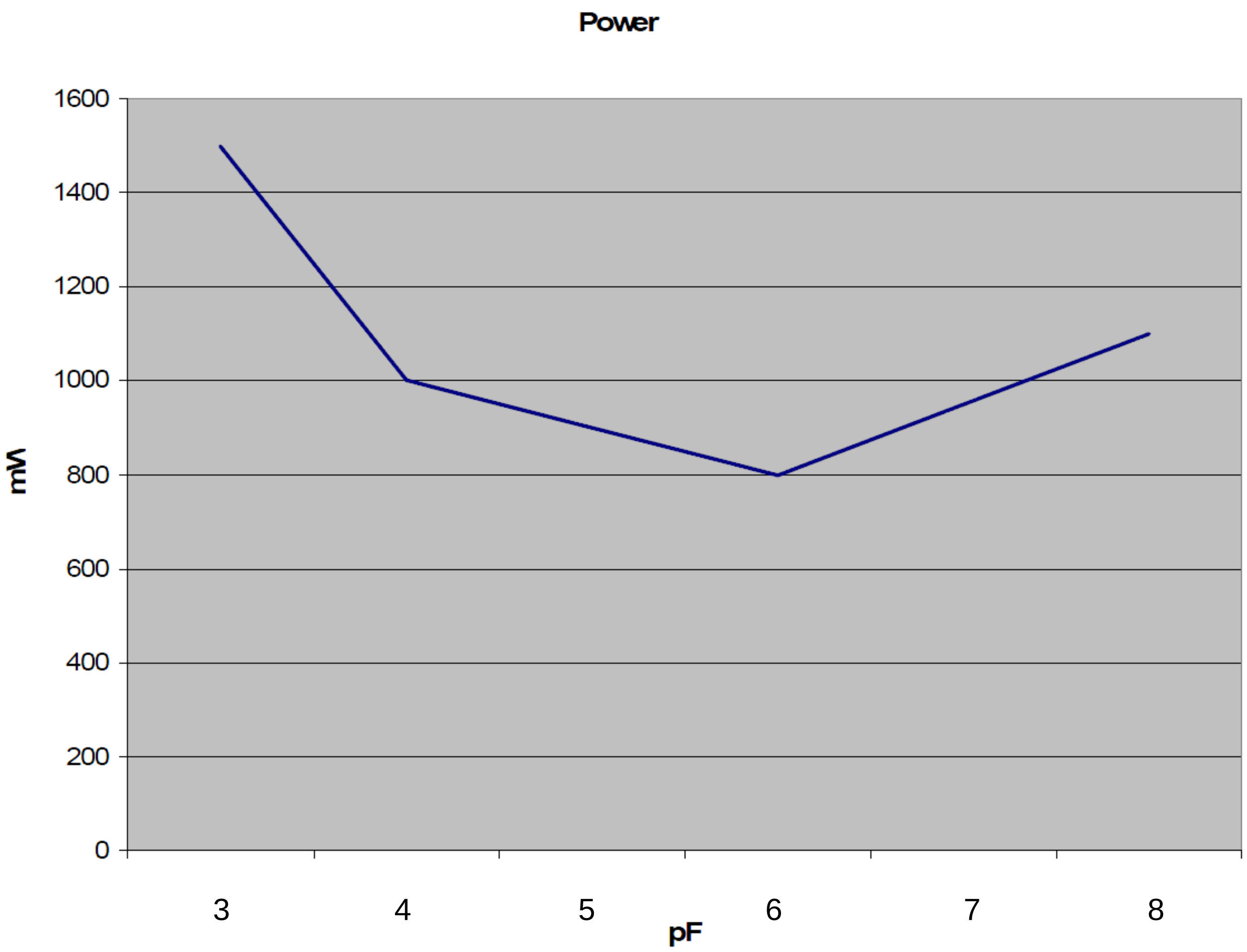
Number of Bits/Stage



Scaling Factor



Capacitance Value



DIGITALLY ASSISTED A/D CONVERTERS

Ahmed Ali

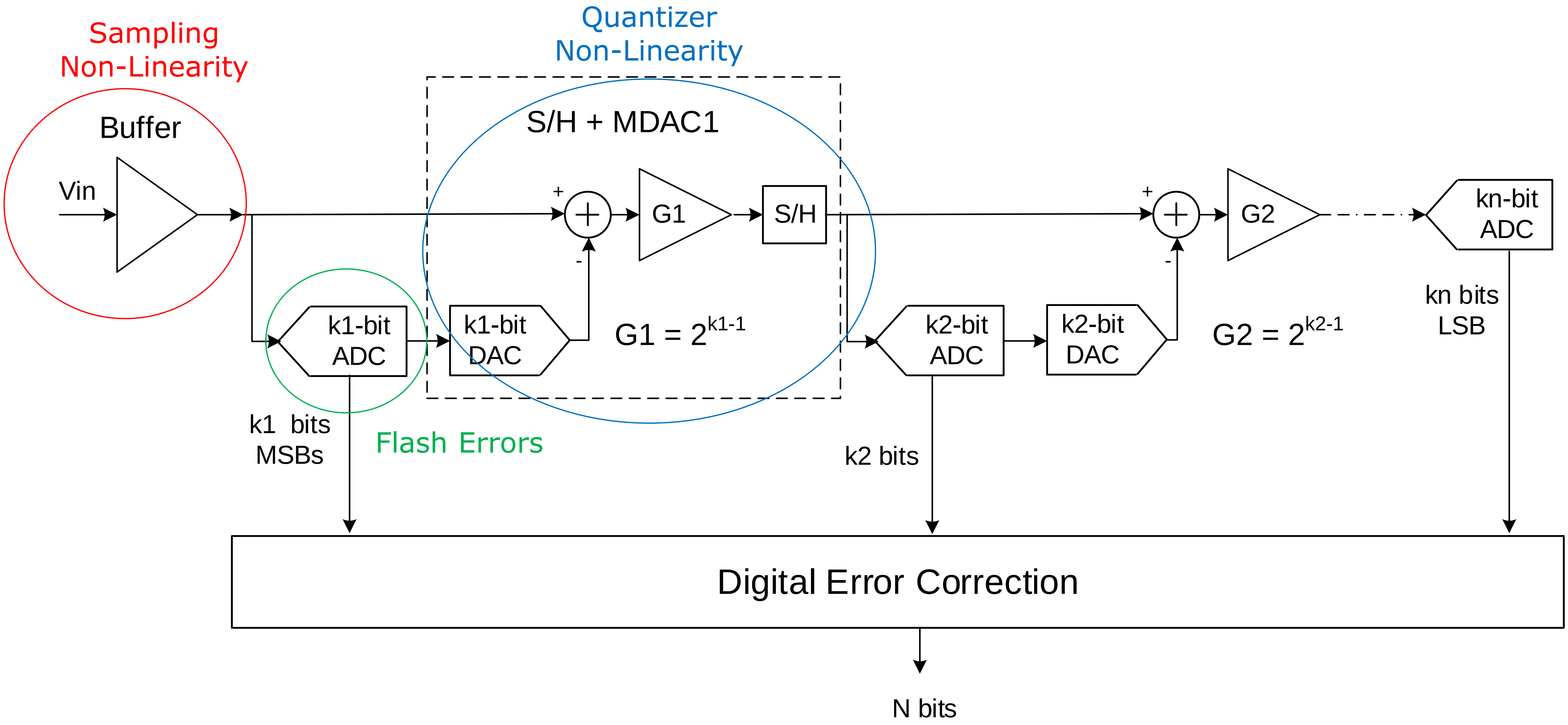
System Challenges

- **Higher sample rates for larger bandwidths:**
  - Simplifies filtering
  - Frequency planning
  - More information
  - More carriers
  - 14-12 bit at > 1GS/s ADCs
- **RF Sampling:**
  - Lower cost
  - More flexibility
- **Lower power consumption:**
  - More integration
  - More receivers
- **Higher linearity:**
  - More focus on small signal linearity and high order harmonics

Process Scaling

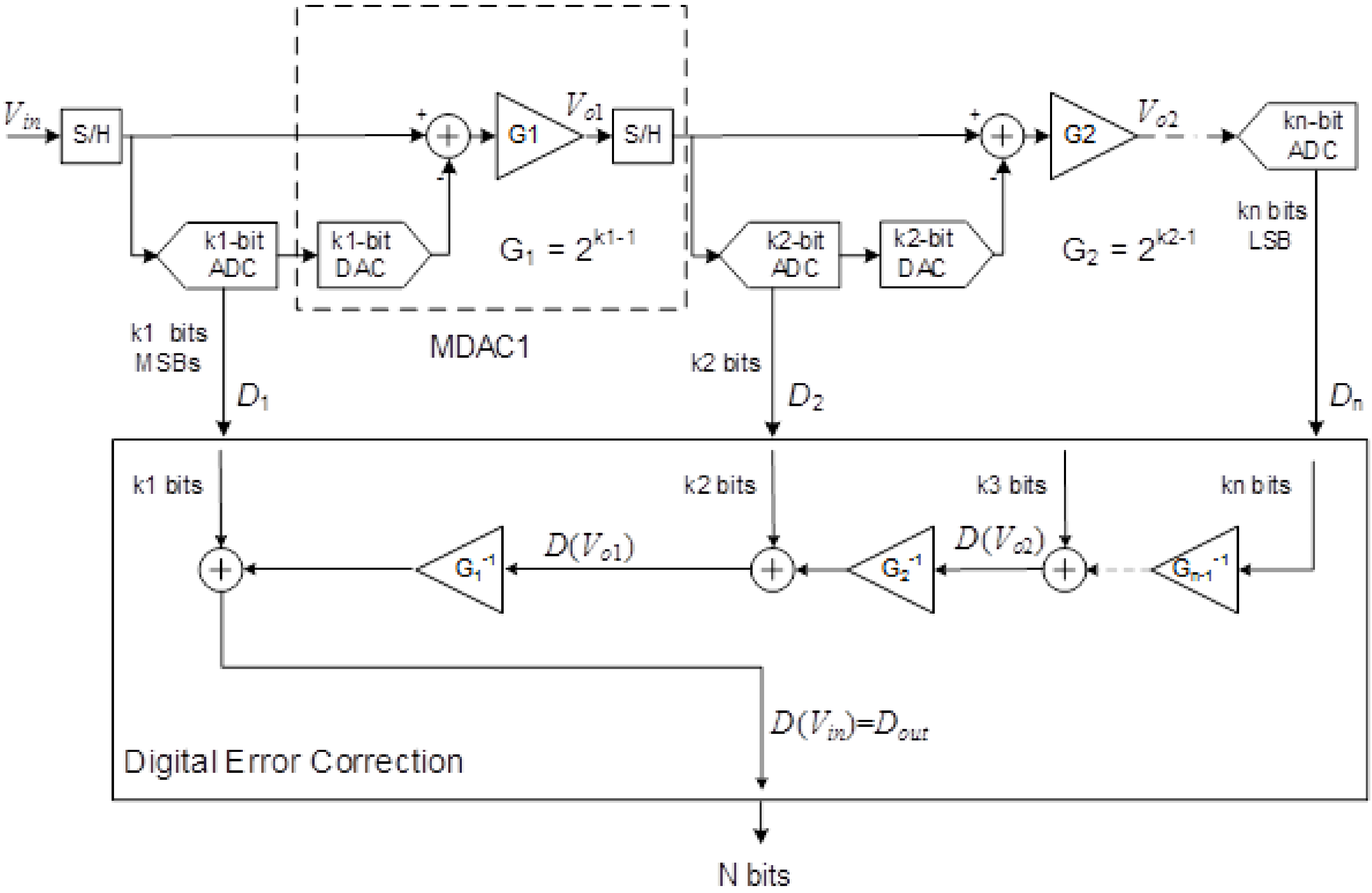
- Digital circuits have got more efficient with process scaling
- Analog circuits get more challenging and can be even less efficient
- Finer geometry gives:
  - Faster switches
  - Lower parasitics
- But …
  - Lower output impedance
  - Lower intrinsic gain
  - Worse dynamic range
- Digital processing can be used to correct for analog non-idealities
- Digital assistance has become necessary

Pipelined ADC



Ahmed Ali

Digital Error Correction



Digital Assistance in Pipelined ADCs

- **Quantizer non-linearity:**
  - Inter-stage gain and settling errors (IGE)
  - DAC errors
  - Reference errors
  - Inter-stage memory errors (IME)
  - MDAC amplifier non-linearity
- **Sampling non-linearity:**
  - Kick-back errors
  - Sampling non-linearity
- **Flash errors [5, 6]:**
  - Offset and gain errors
  - BW/timing mismatch errors

Digital Assistance in Pipelined ADCs

- **Quantizer non-linearity:**
  - **Inter-stage gain and settling errors (IGE)**
  - **DAC errors**
  - **Reference errors**
  - **Inter-stage memory errors (IME)**
  - **MDAC amplifier non-linearity**
- **Sampling non-linearity:**
  - **Kick-back errors**
  - **Sampling non-linearity**
- **Flash errors [5, 6]:**
  - **Offset and gain errors**
  - **BW/timing mismatch errors**

Factory Calibration

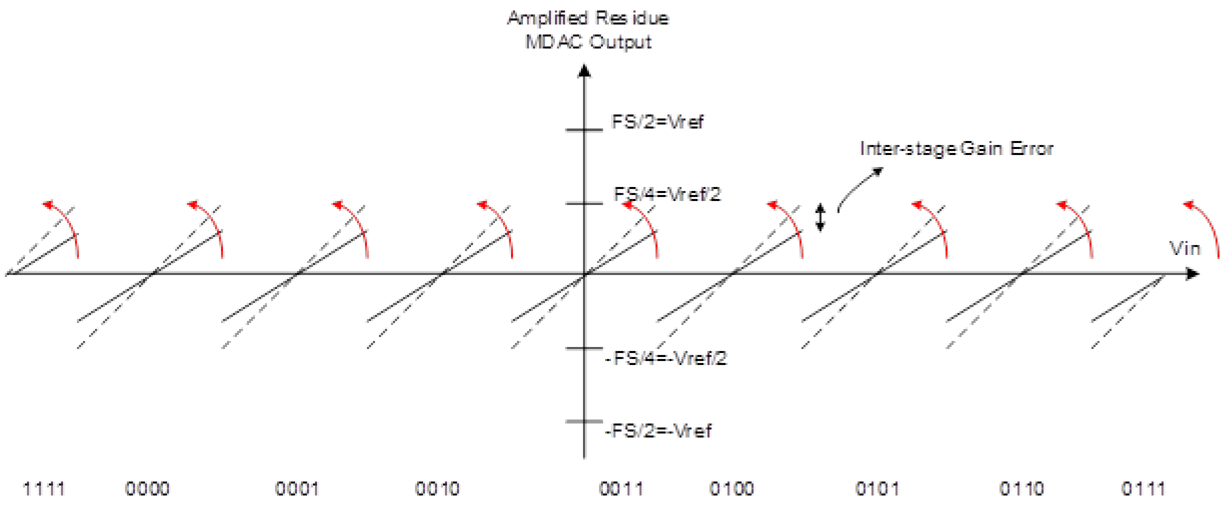
- Capacitor mismatch does not change much with temperature, supply, aging, etc.
  - => It can be factory calibrated

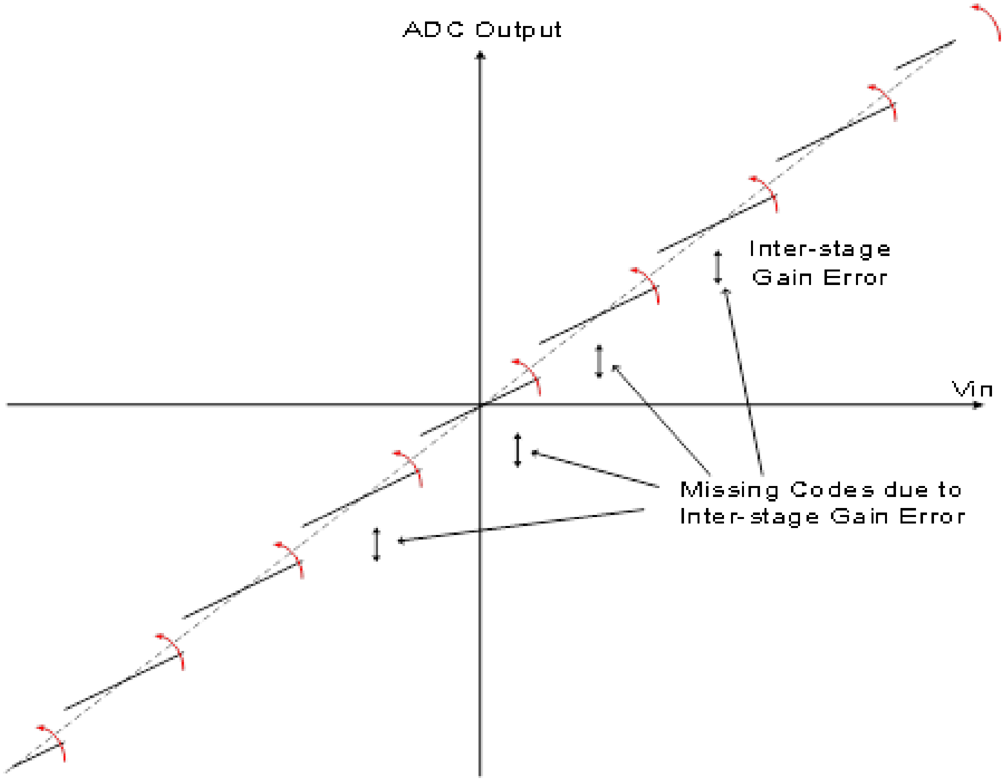
- This is done using digital coefficients in the backend

- Factory digital calibration can be employed for some other sources of error, but with less consistency and reliability

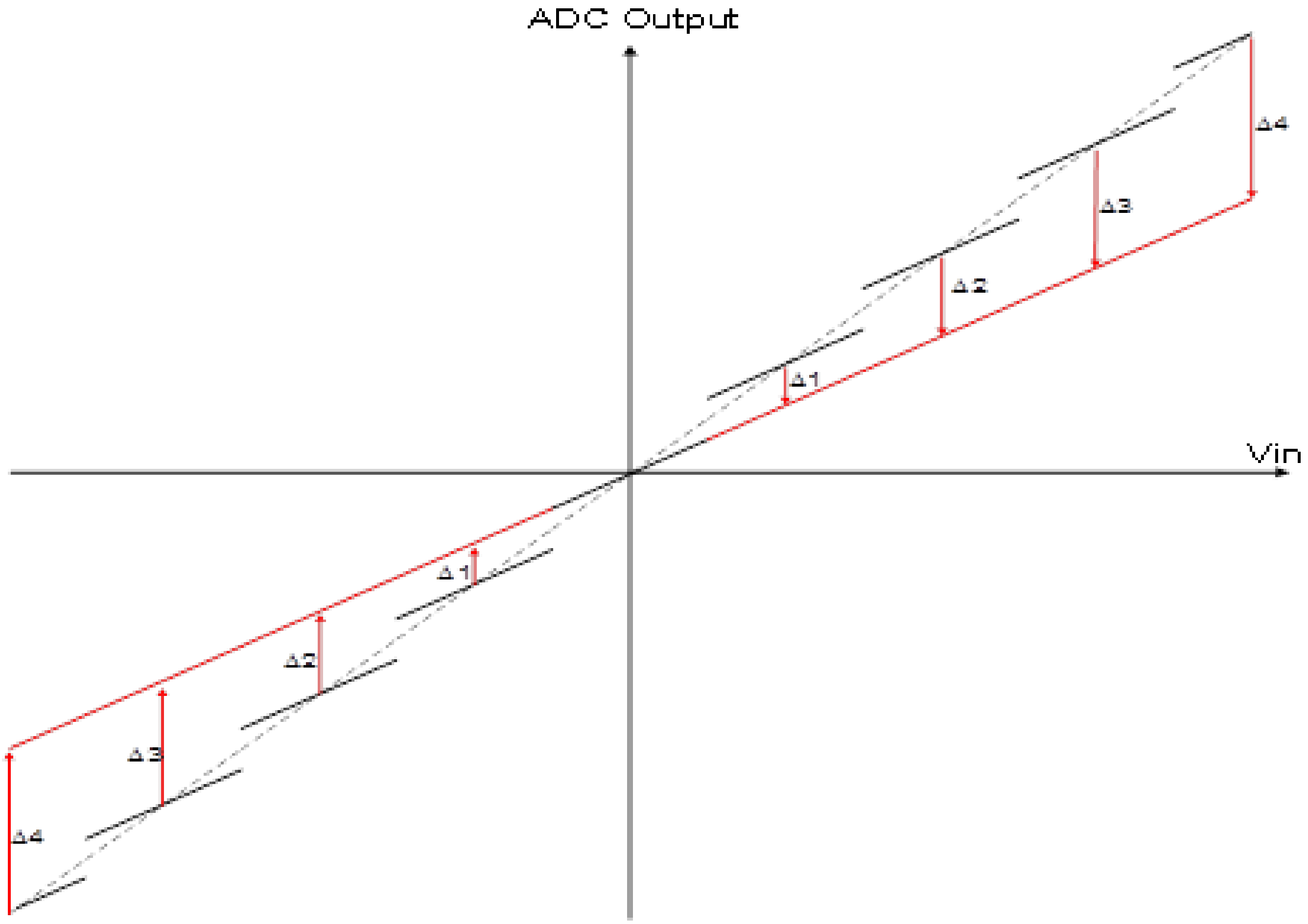
- Background calibration of internal non-linearity is increasingly becoming more important

Inter-stage Gain Error Calibration

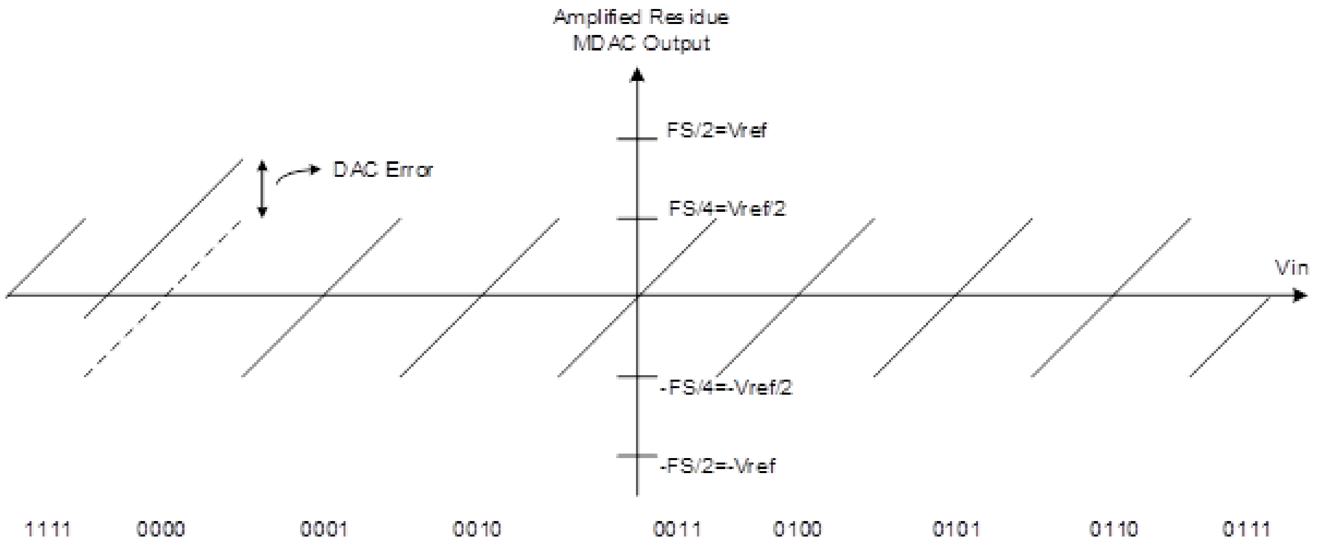


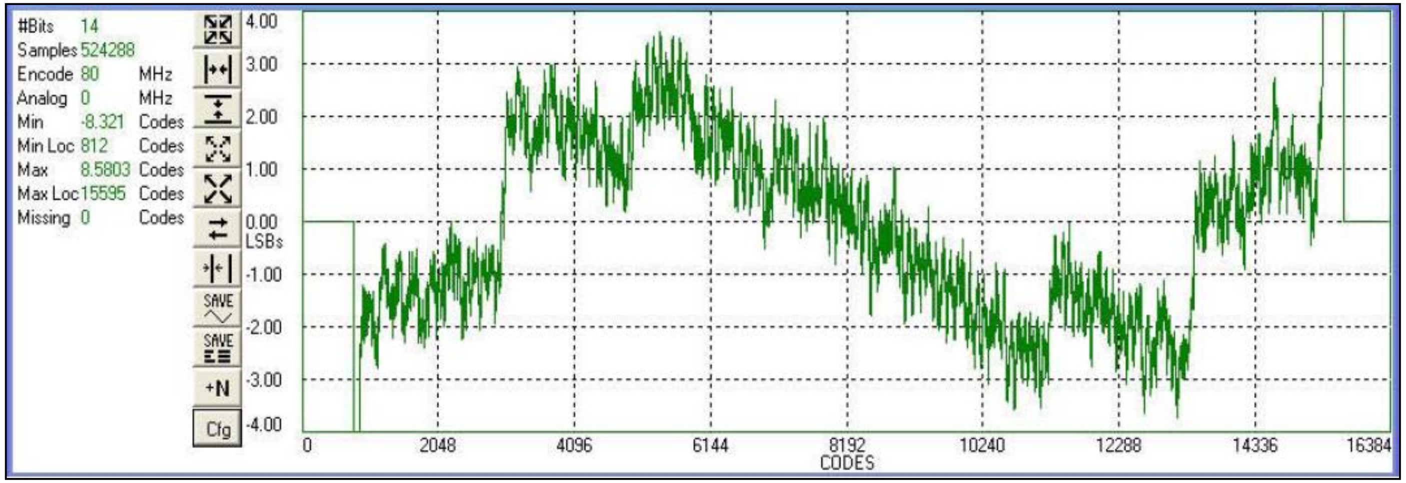


Inter-stage Gain Error Calibration Using Sub-range Shifting



DAC Errors



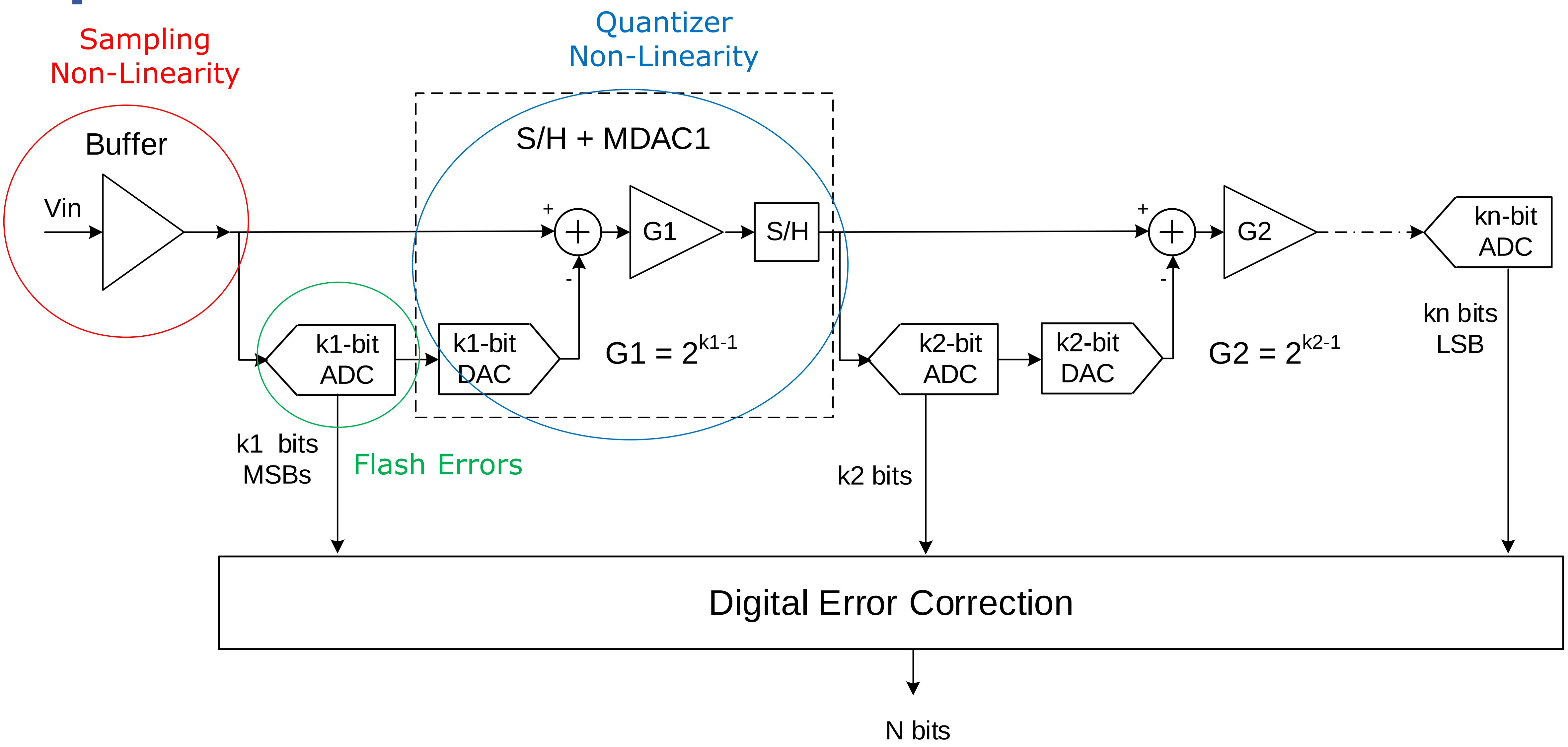


Ahmed Ali

Calibration and Dither

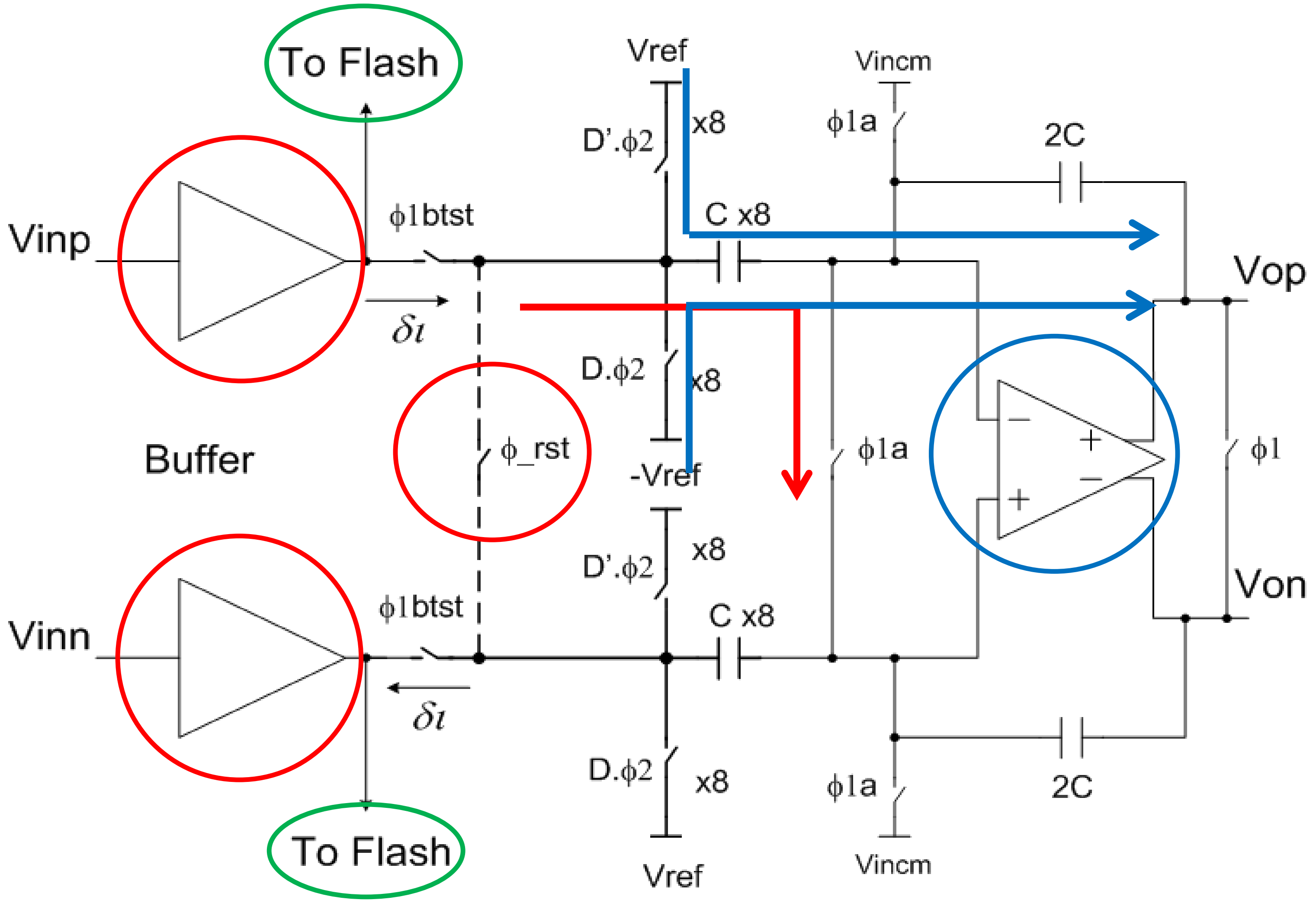
- Mostly to improve linearity and hence SNDR and SFDR
- Calibration attempts to fix non-linearity
  - Improves both SNDR and SFDR
- Dither randomizes non-linearity
  - Spreads harmonics and spurs in the noise floor
  - Improves SFDR, but not SNDR
- Which one to use?
  - Calibration for large errors that are fixable
  - Dither for small or hard to fix errors
  - Preferably, use both:
    - Calibration to maximize SNDR
    - Dither to maximize SFDR

Pipelined ADC



Ahmed Ali

Stage-1 MDAC



Ahmed Ali

Switched Capacitor MDAC

Capacitor Error Terms

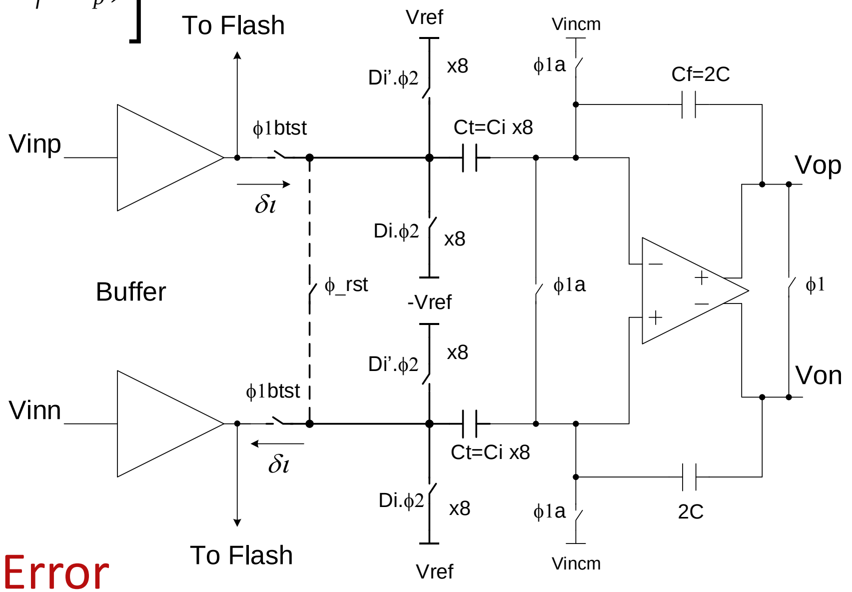
Settling Error Term

$$V_o \approx \left[\frac{V_{in}.C_t/C_f}{1+(C_t+C_f+C_p)/(C_f.A)} - \sum_{i=1}^{8}\frac{D_iV_{ref}.C_i/C_f}{1+(C_t+C_f+C_p)/(C_f.A)}\right]\left[1-e^{-t2\pi f_t C_f/(C_t+C_f+C_p)}\right]$$

$$V_o \approx \left[\frac{V_o|_{ideal}}{1+(C_t+C_f+C_p)/(C_f.A)}\right]\left[1-e^{-t2\pi f_t C_f/(C_t+C_f+C_p)}\right]$$

$$V_o \approx \frac{V_o|_{ideal}}{1+K/A}\left[1-e^{-t2\pi f_t/K}\right]$$

$$V_o \approx V_o|_{ideal} \underbrace{-V_o|_{ideal}.e^{-2\pi BW.t}}_{\text{Amplifier Settling Error}} \underbrace{-V_oK/A}_{\text{Amplifier DC Error}}$$



Ahmed Ali

DAC Errors

- Caused by:
  - Capacitor mismatch
  - Settling errors: switches can be very fast
  - Input dependent reference errors: can be minimized
- Fixed (factory) calibration is usually adequate
- Dynamic element matching (shuffling) can fix any residual errors
- Background calibration can also be employed
- Not a major bottleneck

Amplifier Gain, Settling and Non-Linearity

- **Realizing a high-gain amplifier at high speeds is a major challenge:**
  - **For a 14bit, 1GS/s ADC: 100dB with 6GHz BW**

- **The amplifier open-loop gain error depends on supply, temperature, and sometimes sample rate (settling errors).**
  - **=> Factory calibration may not be an attractive option**

- **The amplifier open-loop gain error can also be input-dependent => Non-linear**

- **The amplifier is a major bottle-neck**

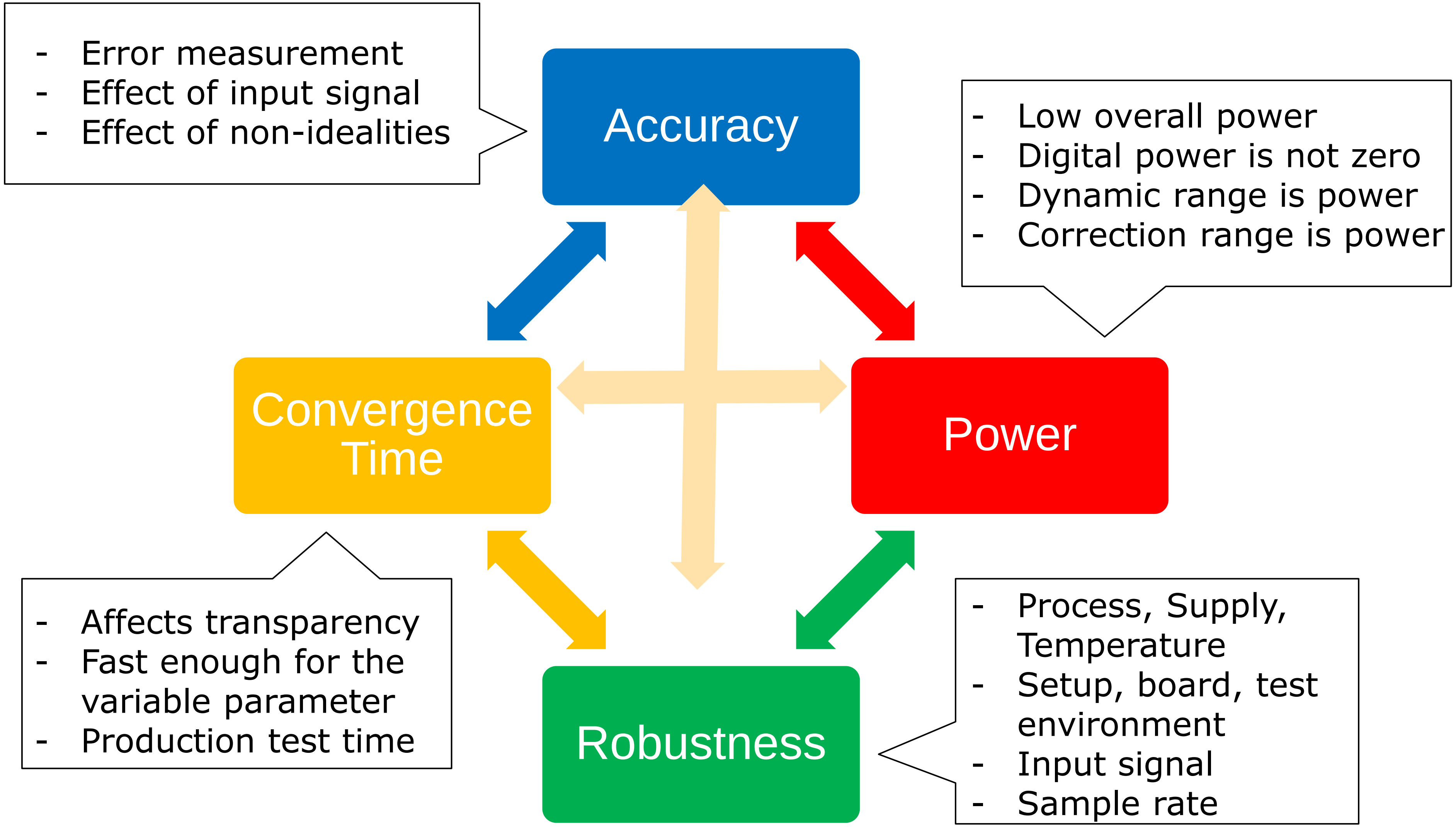
Some Calibration Techniques

- Correlation-based calibration [15, 2, 11, 12]
- Summing node calibration [3, 13]
- Split ADC method [18]
- Using reference ADC [16, 17]
- Using statistical methods [19, 20, 21]

Real World Challenges

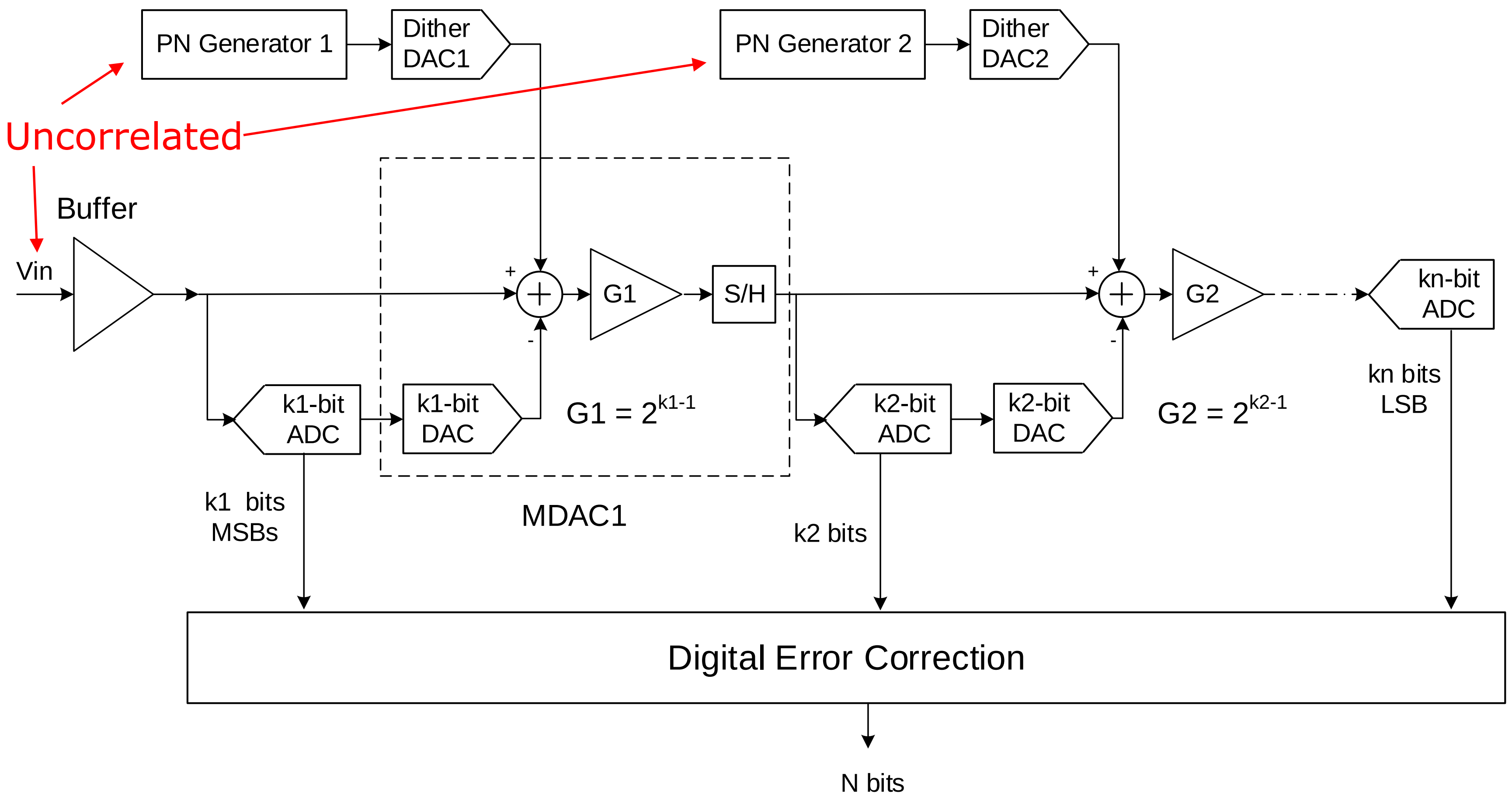
- More than publishing a paper …
- Improve performance and lower power consumption
- No disruption to normal operation => background
- Robustness with input amplitude, input frequency, sample rate, supply, temperature, etc.
- Convergence time
- Manufacturability:
  - Reasonable test time
  - Tester to bench correlation

Some Design Trade-offs



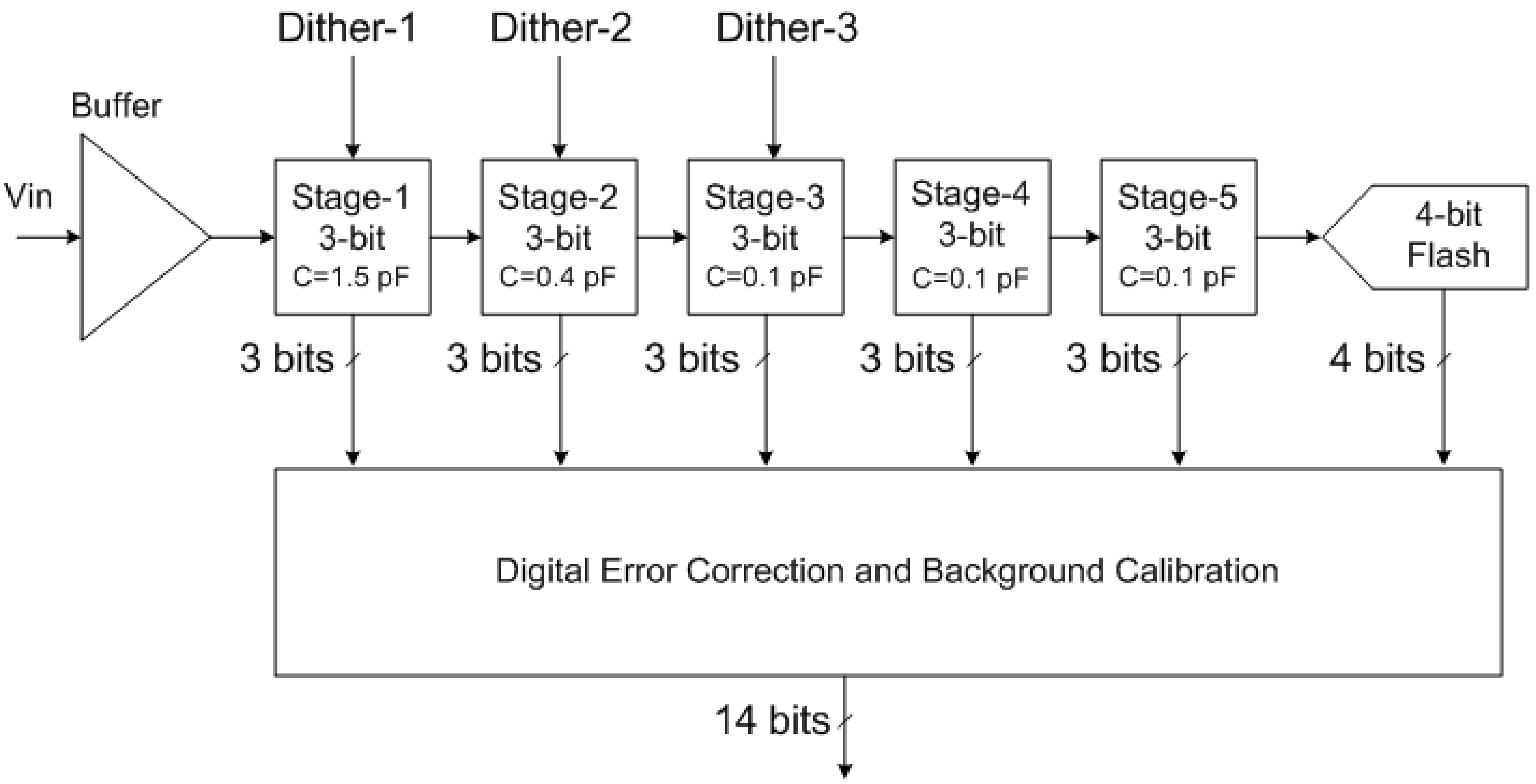
Ahmed Ali

Correlation-Based Calibration



Ahmed Ali

Pipelined ADC Architecture [2]



IGE and Settling Calibration

- IGE calibration can fix gain errors and linear settling errors
- Correction of settling errors can be limited by:
  - Non-linearity due to slewing, reference settling, DAC settling
  - Memory errors
  - Kick-back from the following stage
- The design needs to be optimized to take advantage of IGE calibration
- Memory and kick-back calibration helps improve settling error correction

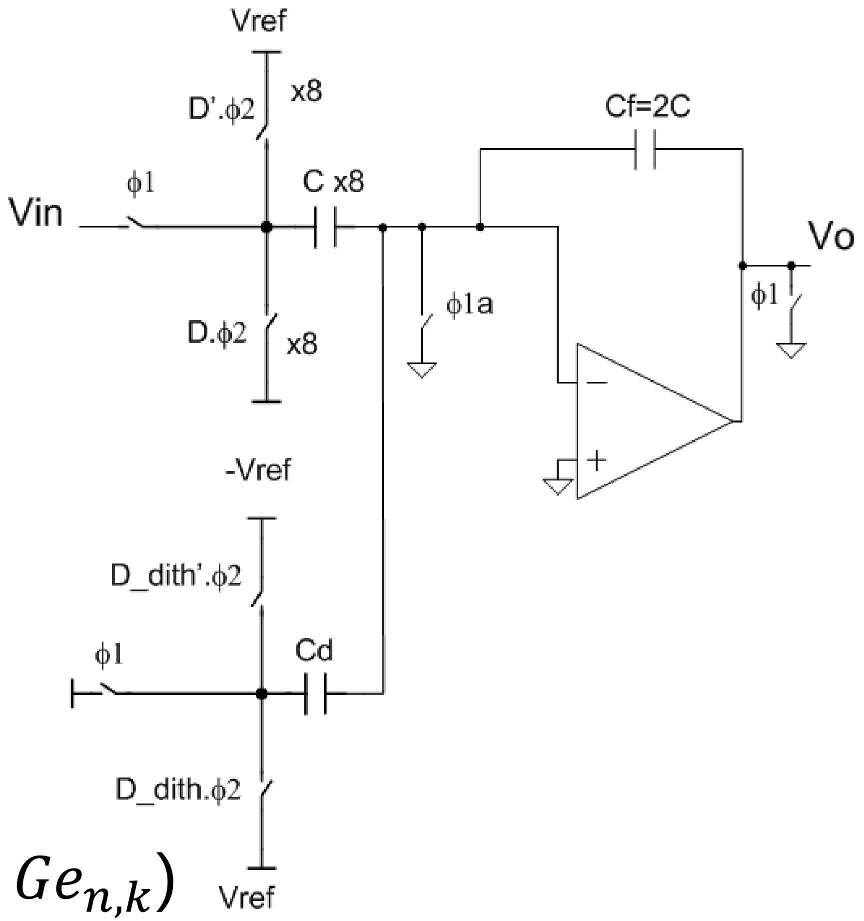
The IGE/IME Calibration [1, 2, 3, 9, 10]

- Correction:

$$V_{R\_cal} = \sum_{k=0}^{M} V_R[n-k] \times 1/Ge_{n+1,k}$$

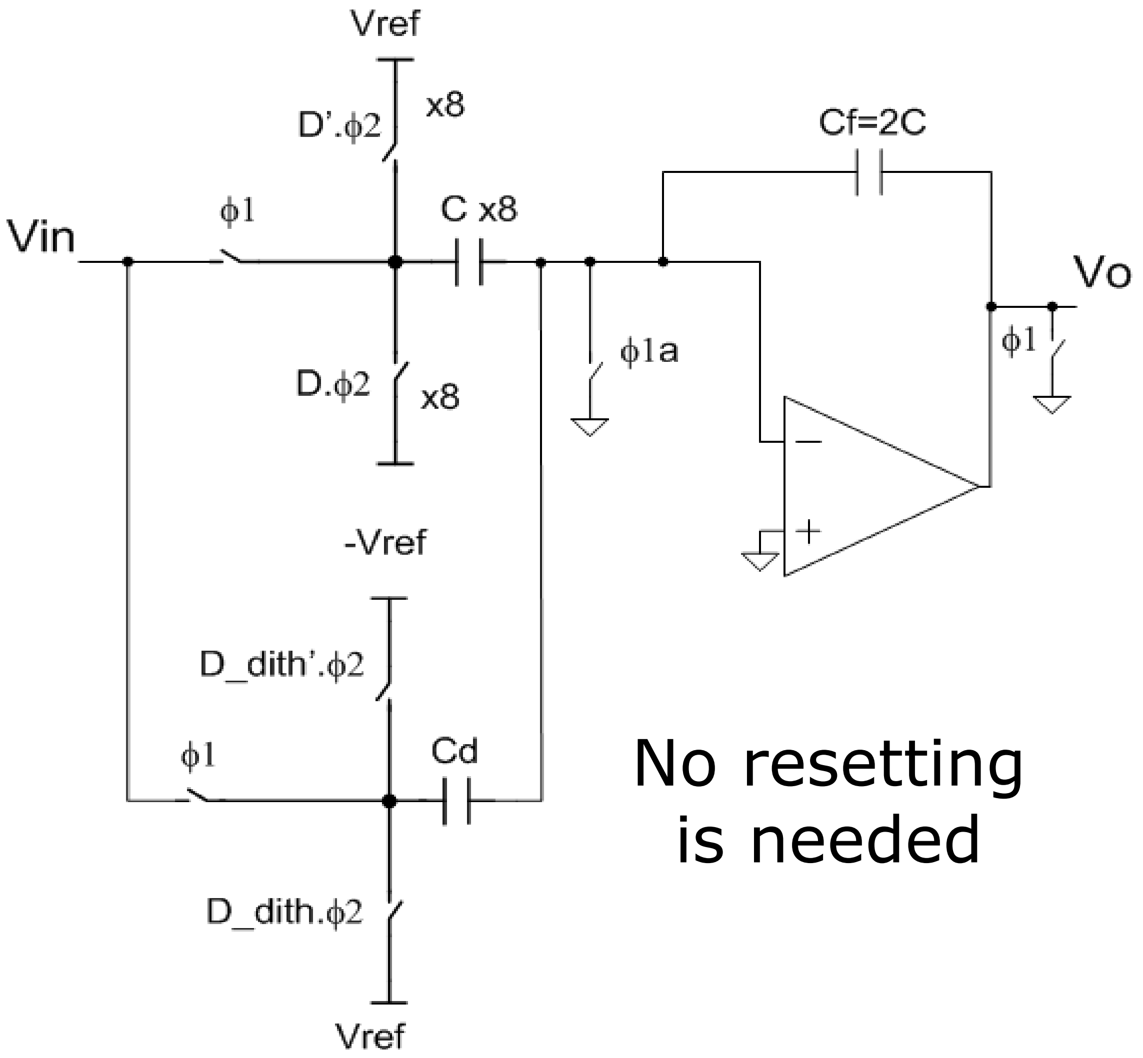
- Estimation:

$$Ge_{n+1,k} = Ge_{n,k} + \mu \cdot V_d[n-k] \cdot (V_R[n] - V_d[n-k] \times Ge_{n,k})$$



Kick-back Background Calibration [1, 2]

- The kick-back dither caps kick the input similar to the sampling capacitances' kick
- The dither's kick is sampled on the total capacitances and propagates down the pipeline



No resetting is needed

Kick-back Background Calibration [1, 2]

- **The LMS algorithms is used to estimate the gain/correction coefficient Gkb of the dither's kick of the previous sample(s)**

- **The correction coefficient Gkb is applied to the previous stage-1 flash bits**

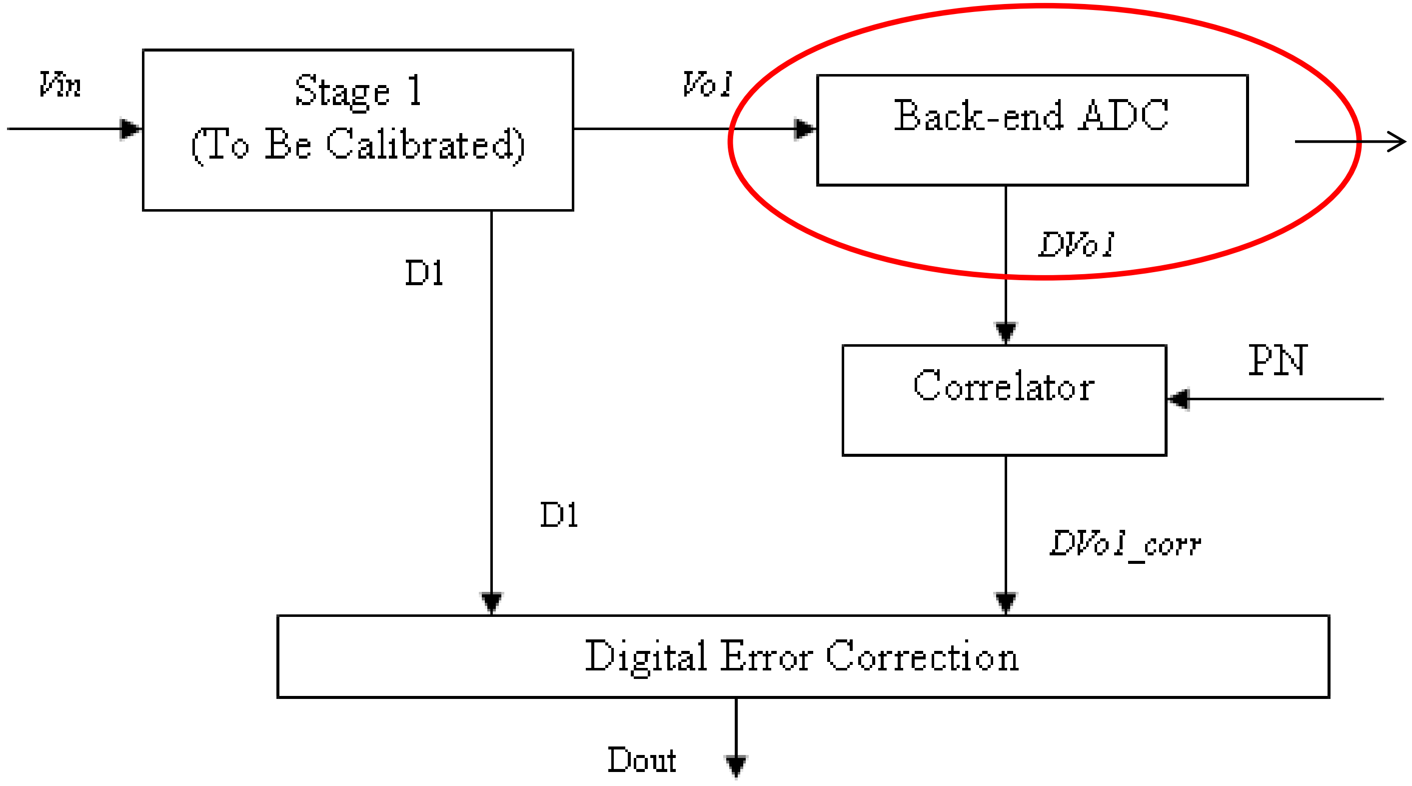
$$Gkb_{n+1,k} = Gkb_{n,k} - \mu * Vd[n-k] * (Vd[n-k] * Gkb_{n,k} - V_{in}[n])$$

$$V_{out\_kbcal}[n] = V_o[n] + \sum_{i=1}^{M_{kb}} D_1[n-i] \times G_{KB,i}$$

Challenges

- What is the accuracy of this approach?
- How is the power consumption?
- Is it robust?
- How about the convergence time?

Correlation-Based Calibration



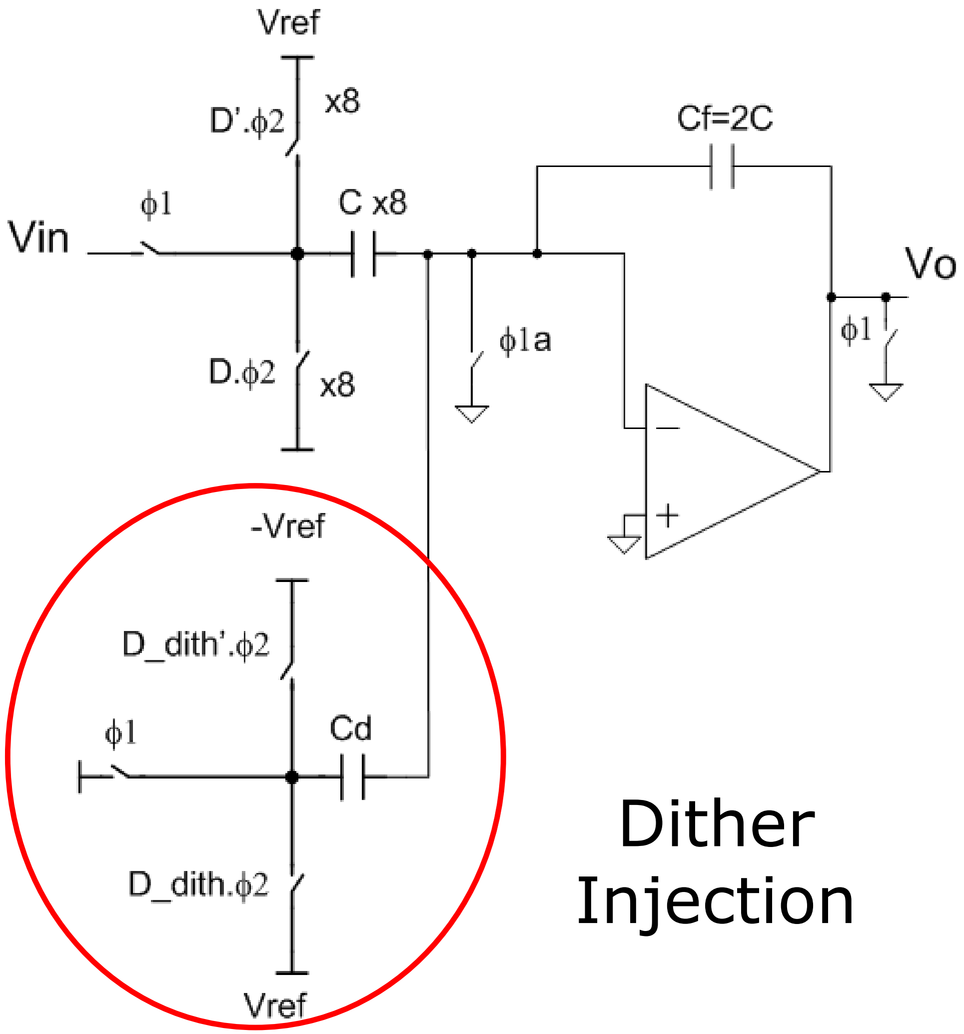
Must have adequate accuracy

LMS Algorithm: $Ge_{n+1} = Ge_n + \mu \cdot V_d[n] \cdot (V_{o1}[n] - V_d[n] \times Ge_n)$

Ahmed Ali

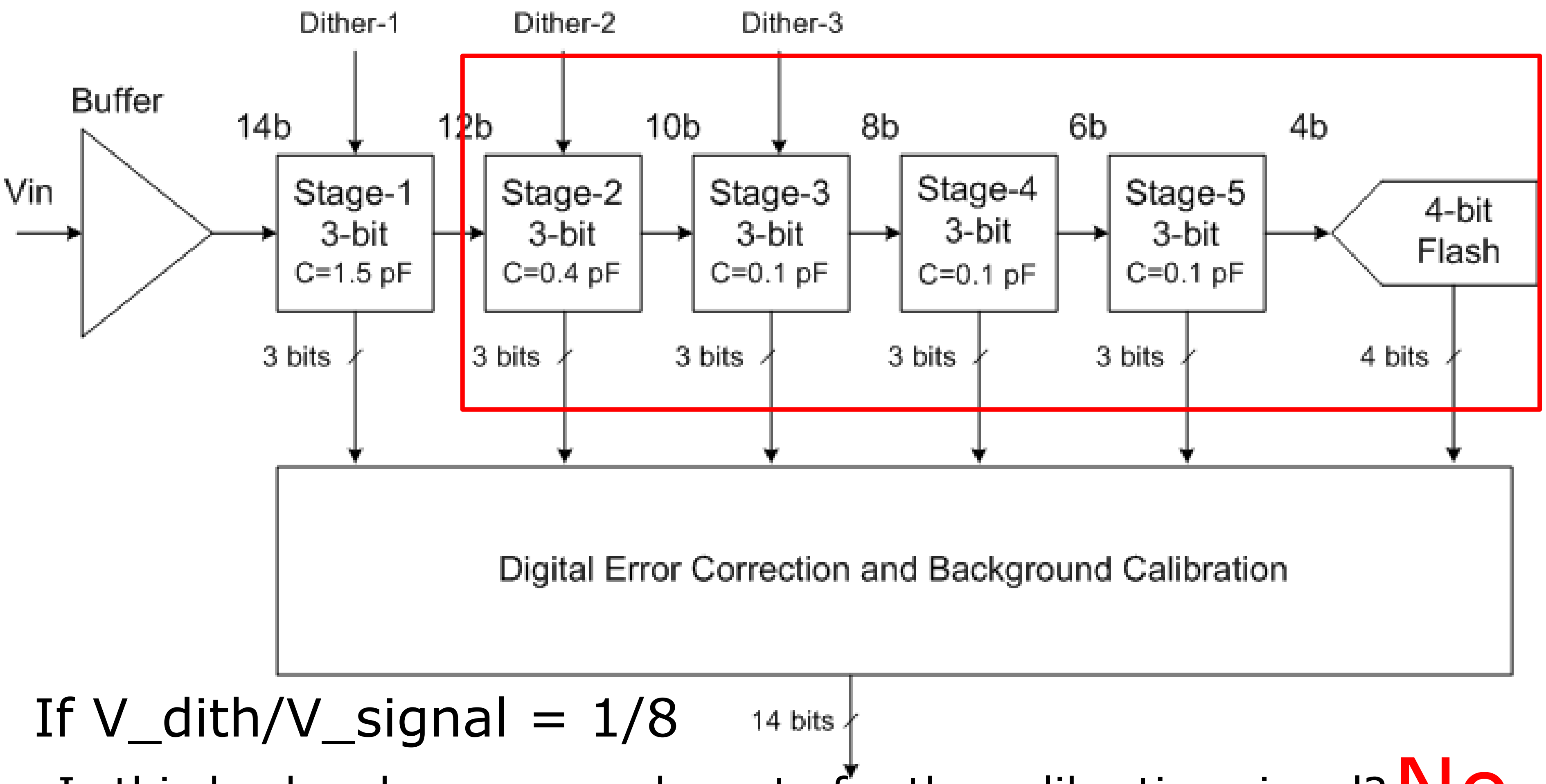
Correlation-Based Calibration

- A portion of the correction range is consumed by the calibration random sequence
- This can cause a significant power overhead
- It is desirable to minimize the amplitude of the calibration signal (dither): V_dith



Dither Injection

But ...



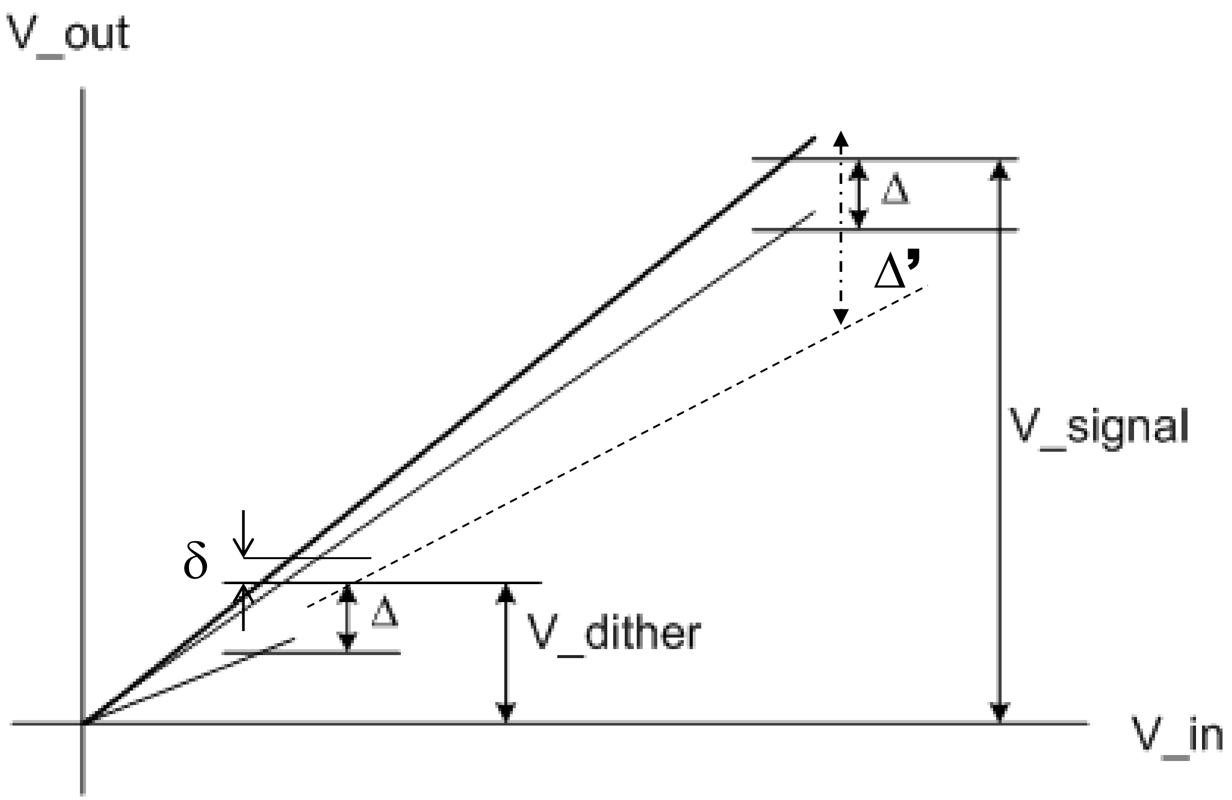
If V_dith/V_signal = 1/8

Is this backend accuracy adequate for the calibration signal? No

Effect of Using a Small Dither

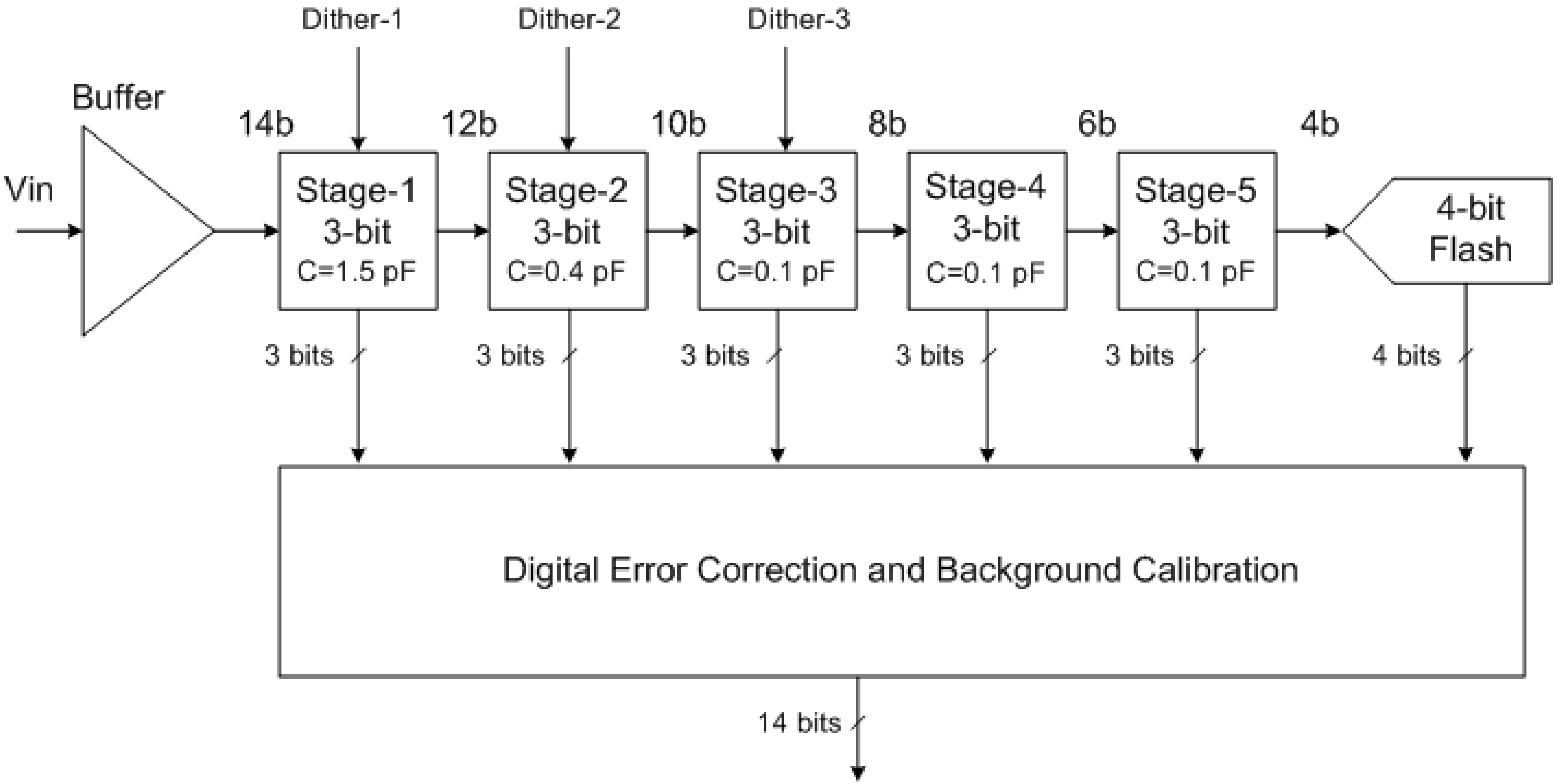
- The calibration accuracy is degraded by the ratio of:

  V_dither/V_signal
- Degrades accuracy when using small dither for calibration
- We need to improve the linearity of the back-end by at least that ratio

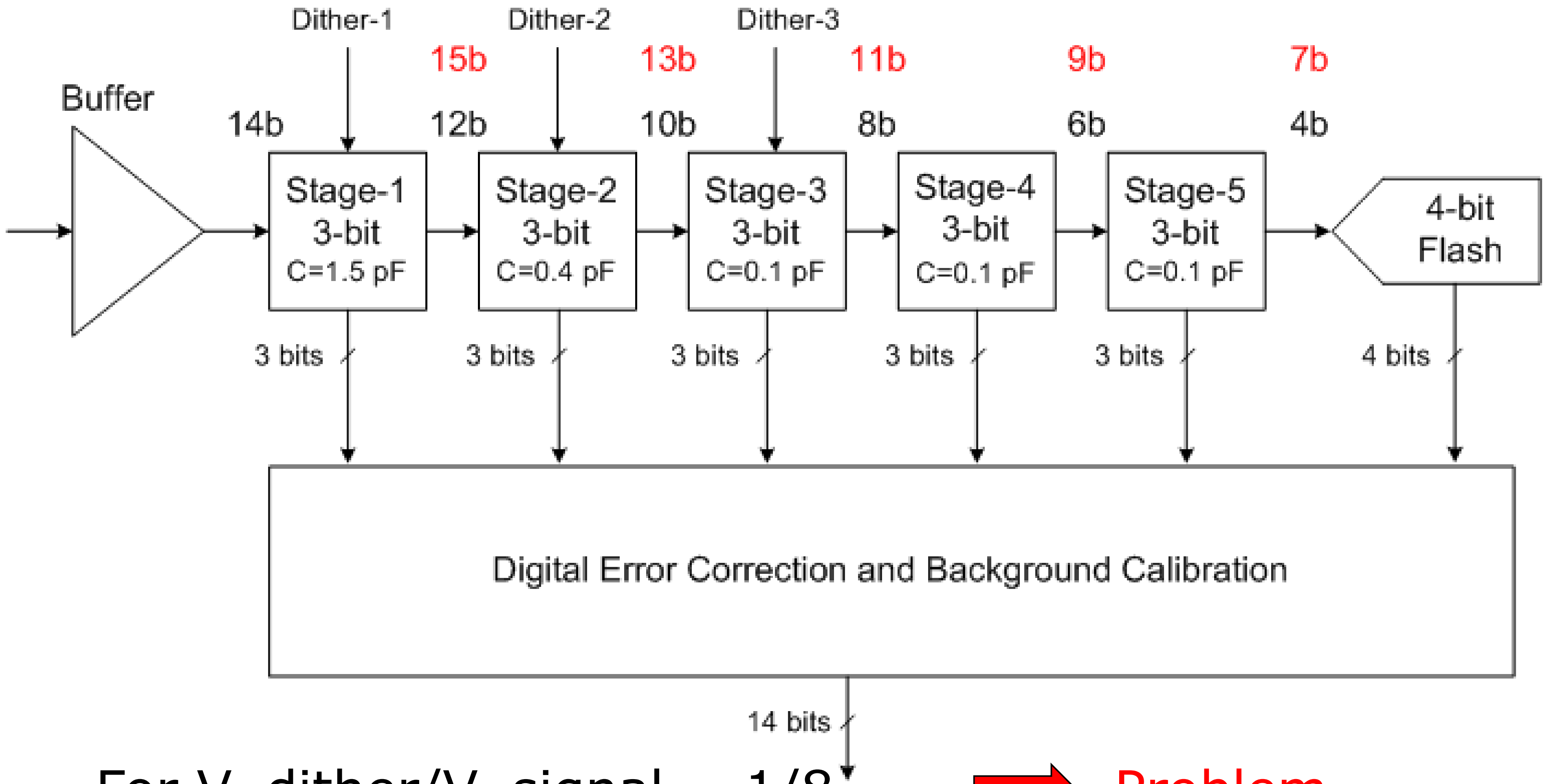


$$\delta/\Delta = \Delta/\Delta' = V\_dither/V\_signal$$

Accuracy of Pipeline Stages



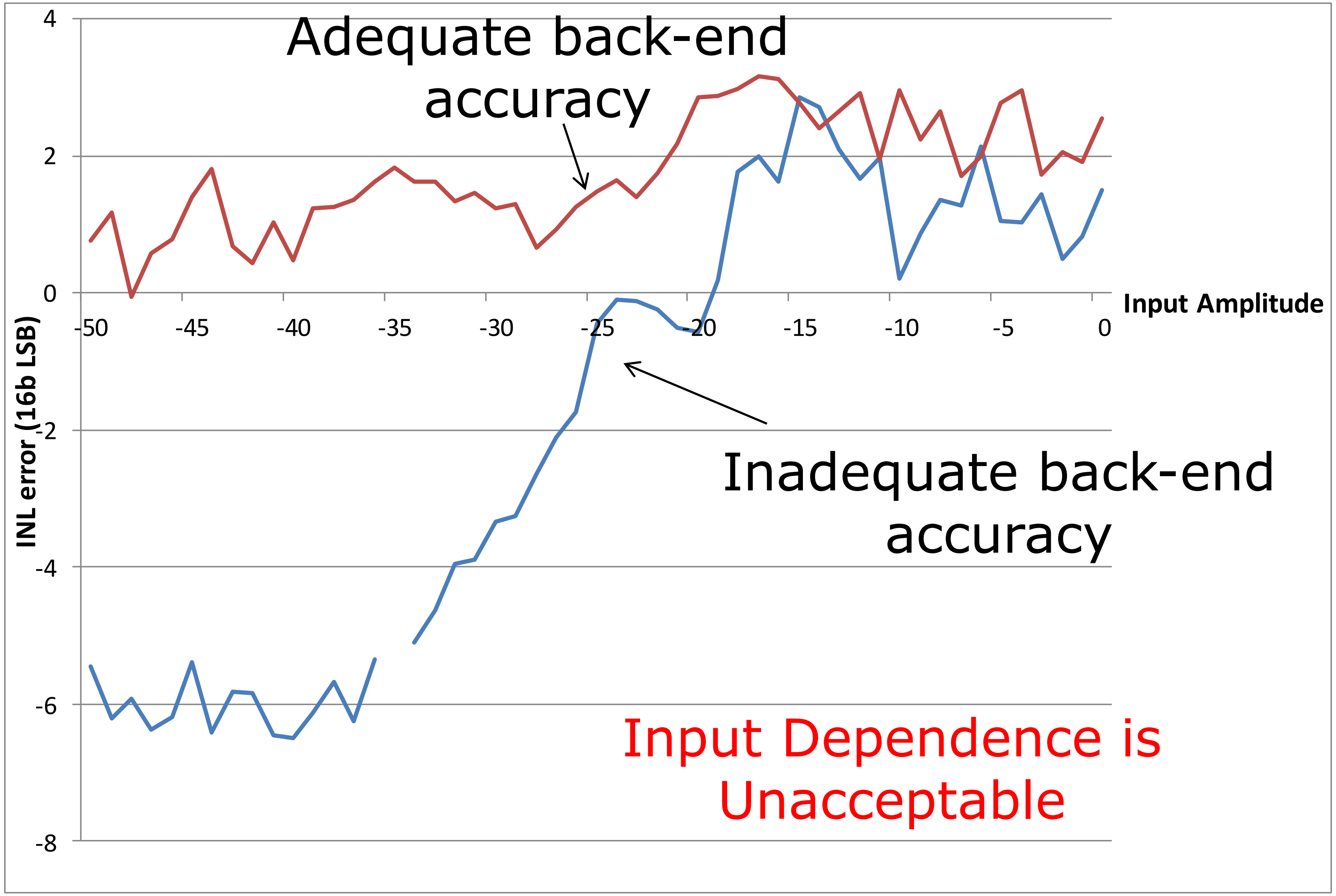
Accuracy of Pipeline Stages



For V_dither/V_signal = 1/8 ➡ Problem

Ahmed Ali

Example of the Problem

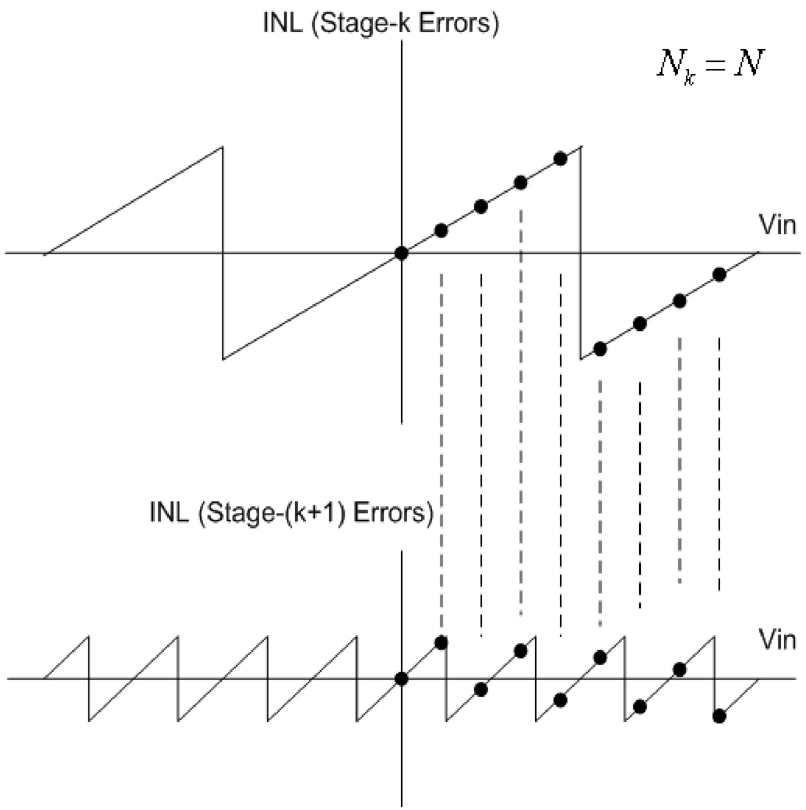


Ahmed Ali

Effective Solution

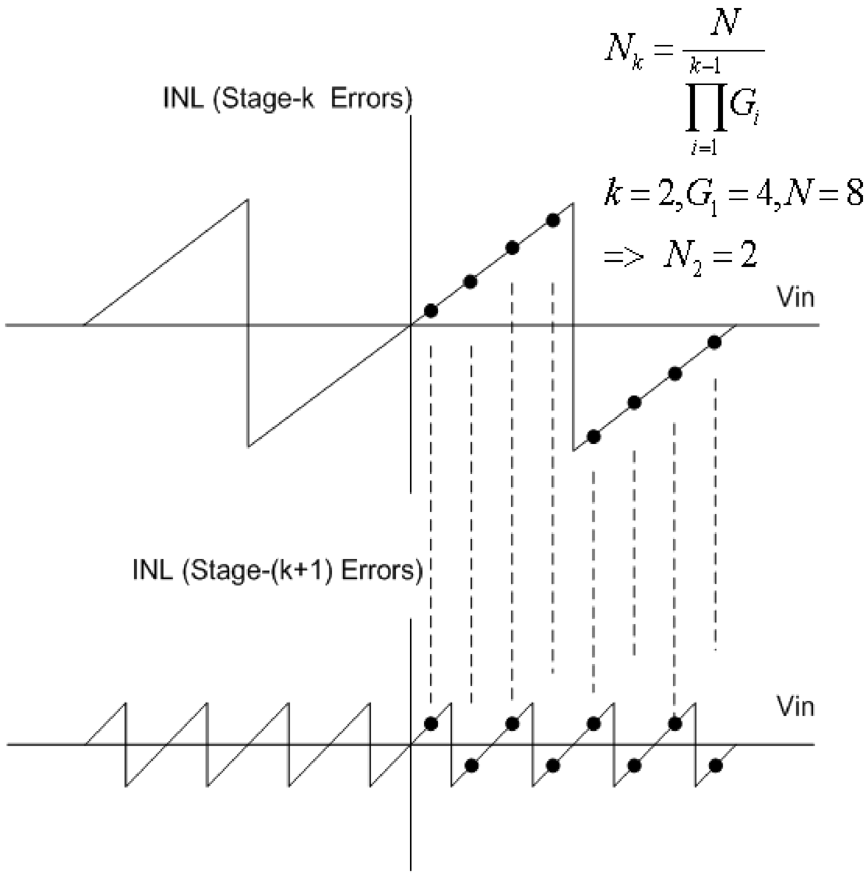
- Inject a small dither signal in the MDAC
- Since the dither is small, we only use a small part of the correction range => small power overhead
- Use multi-level dither (odd number) in every stage to improve accuracy of the back-end (9 levels => 3.2-bits)
- Dynamic element matching can be utilized
- Ensure effective dither propagation down the pipeline such that the number of dither levels for each stage is adequate for the required accuracy
- In addition, large dither injection can be employed

Dither Propagation



Dither using odd number of levels (9)

✓

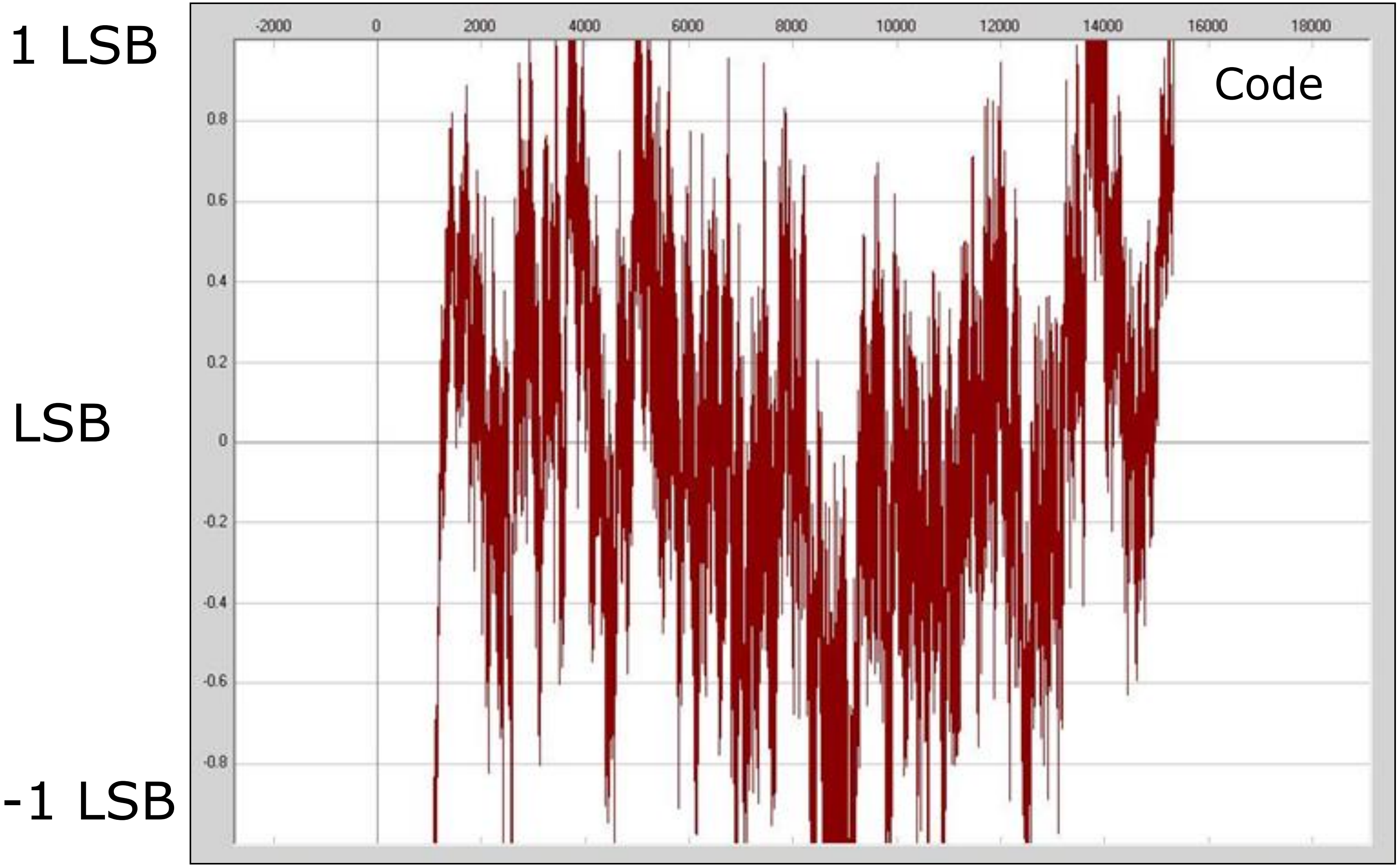


Dither using even number of levels (8)

X

Ahmed Ali

INL at Room Temperature

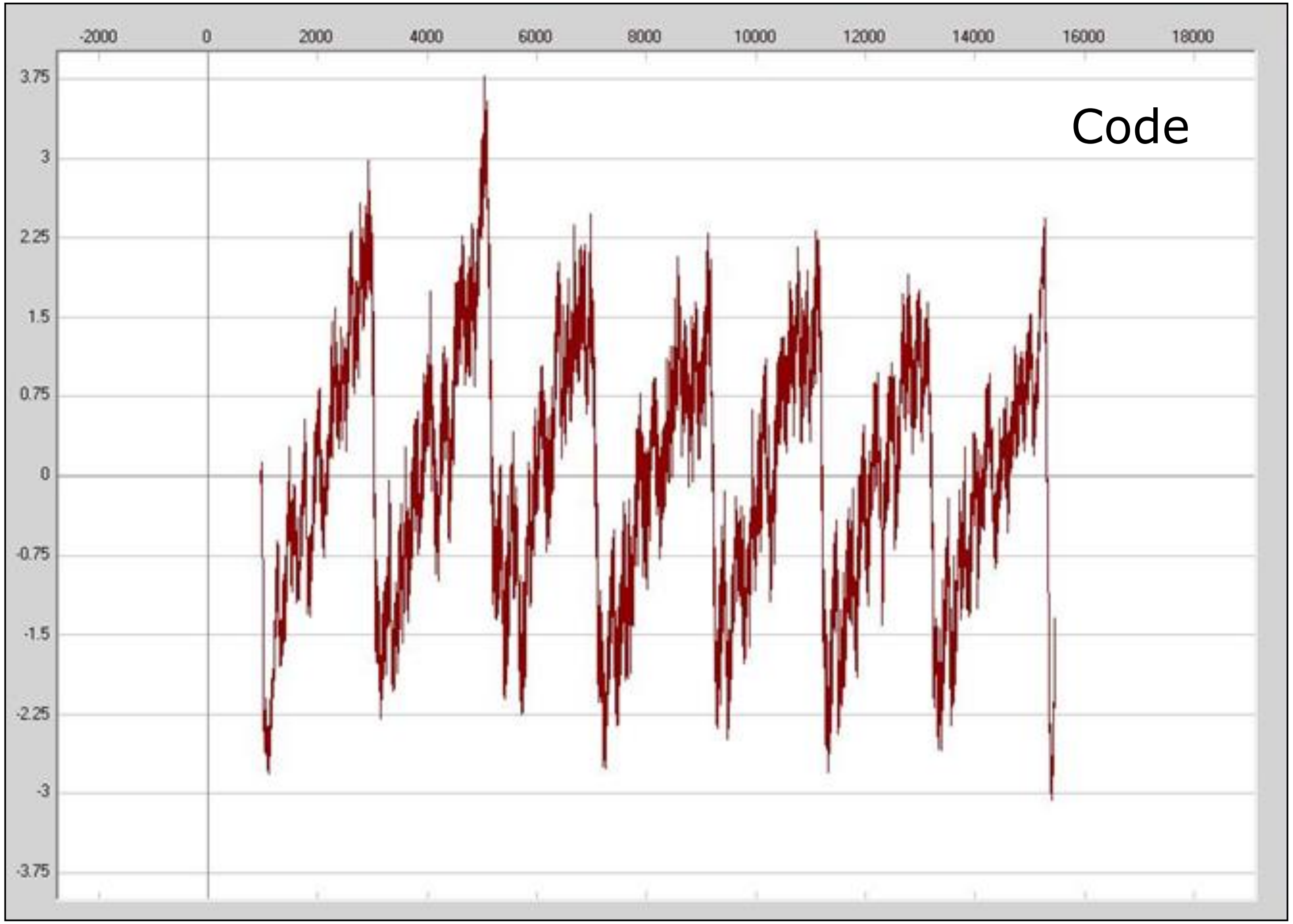


INL at High Temperature
Calibration OFF

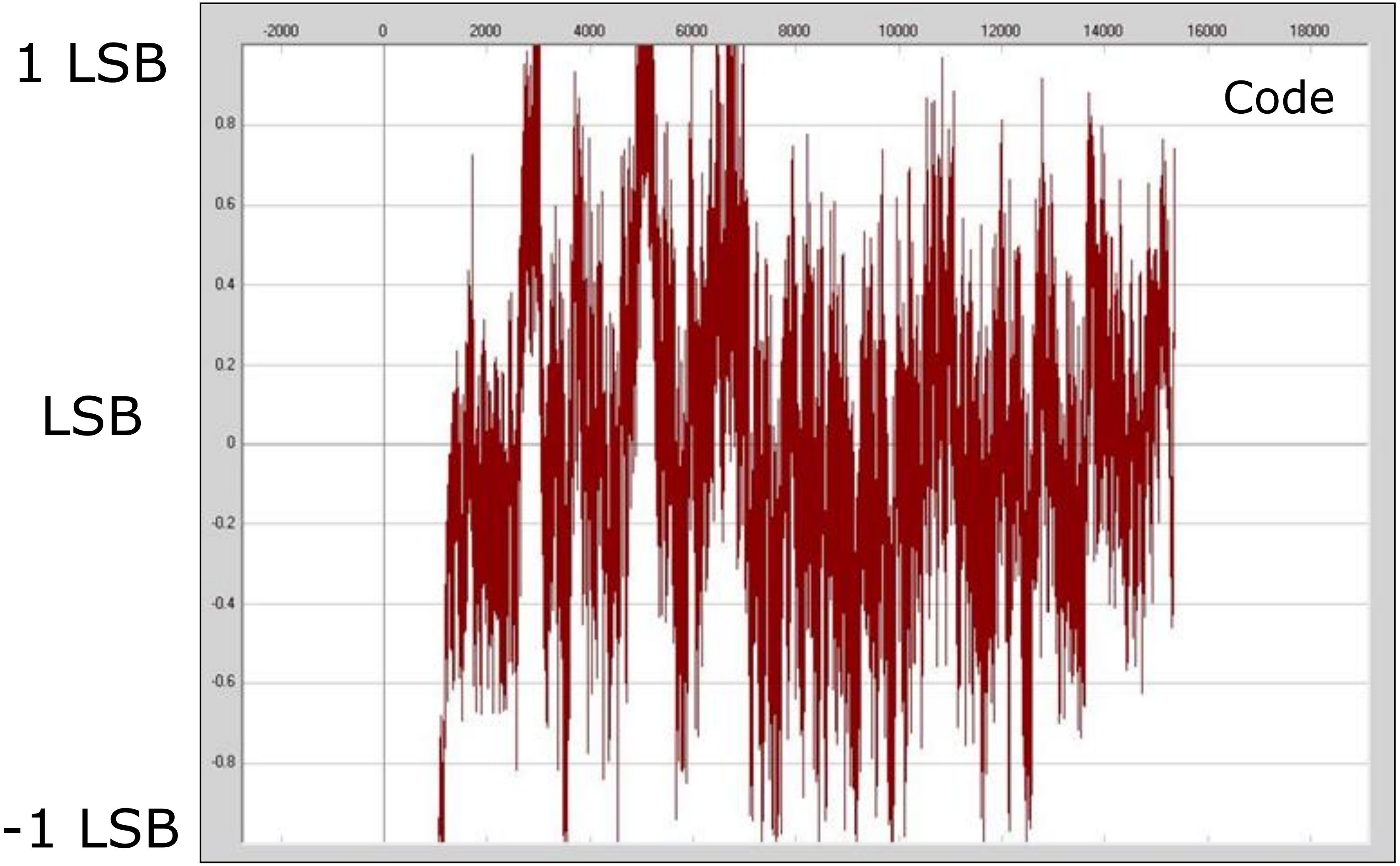
4 LSB

LSB

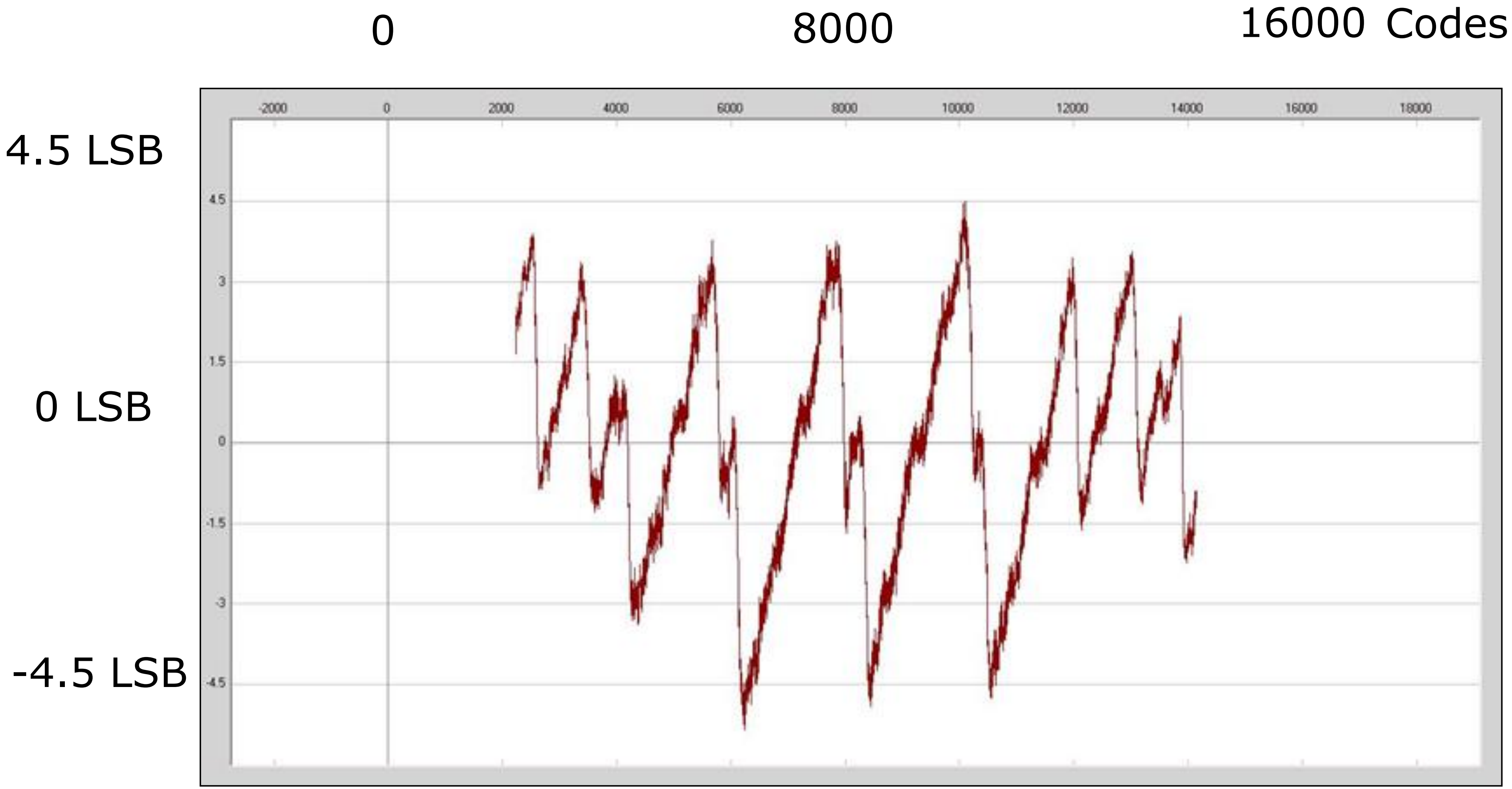
-4 LSB



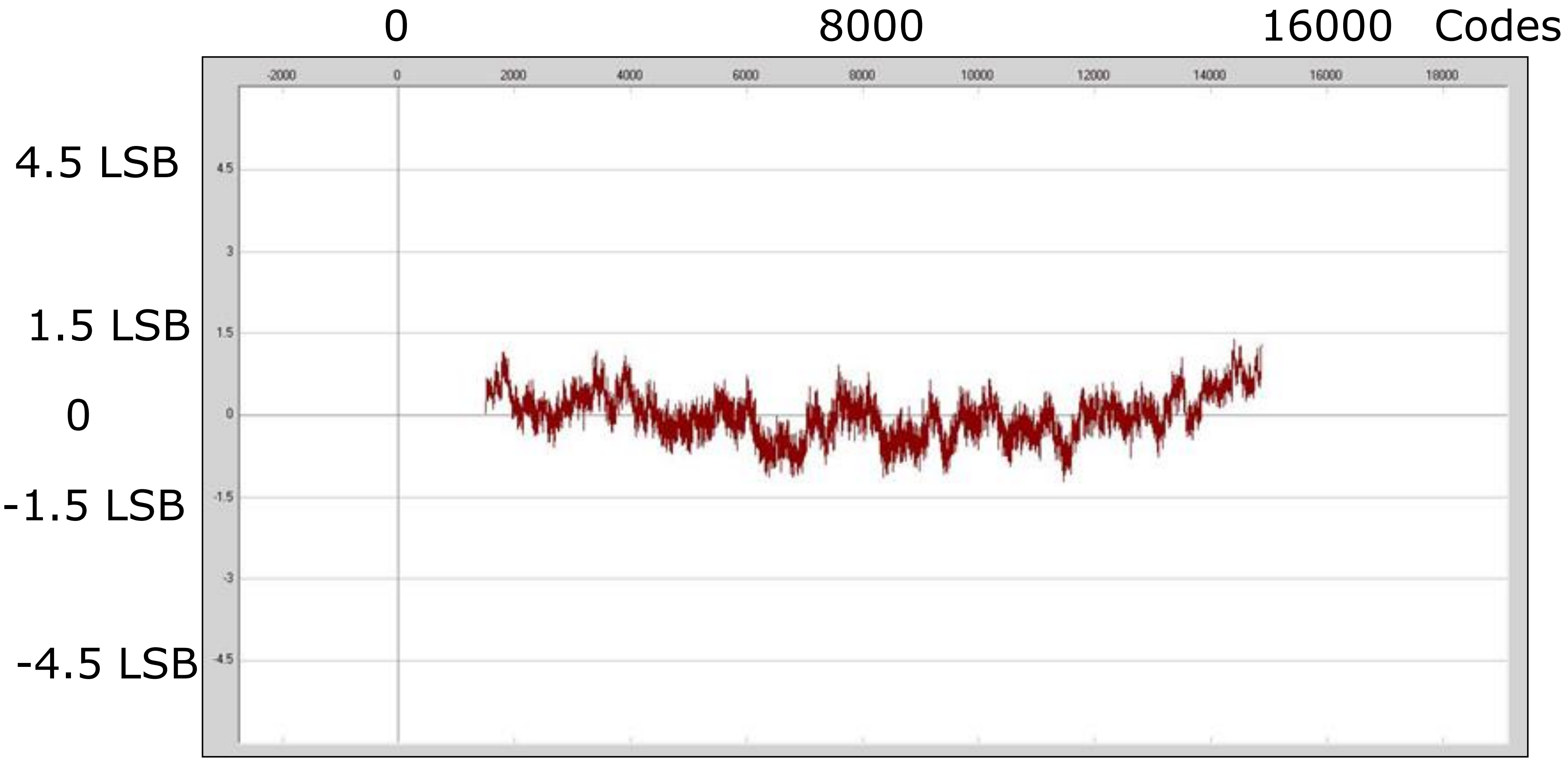
INL at High Temperature
Calibration ON



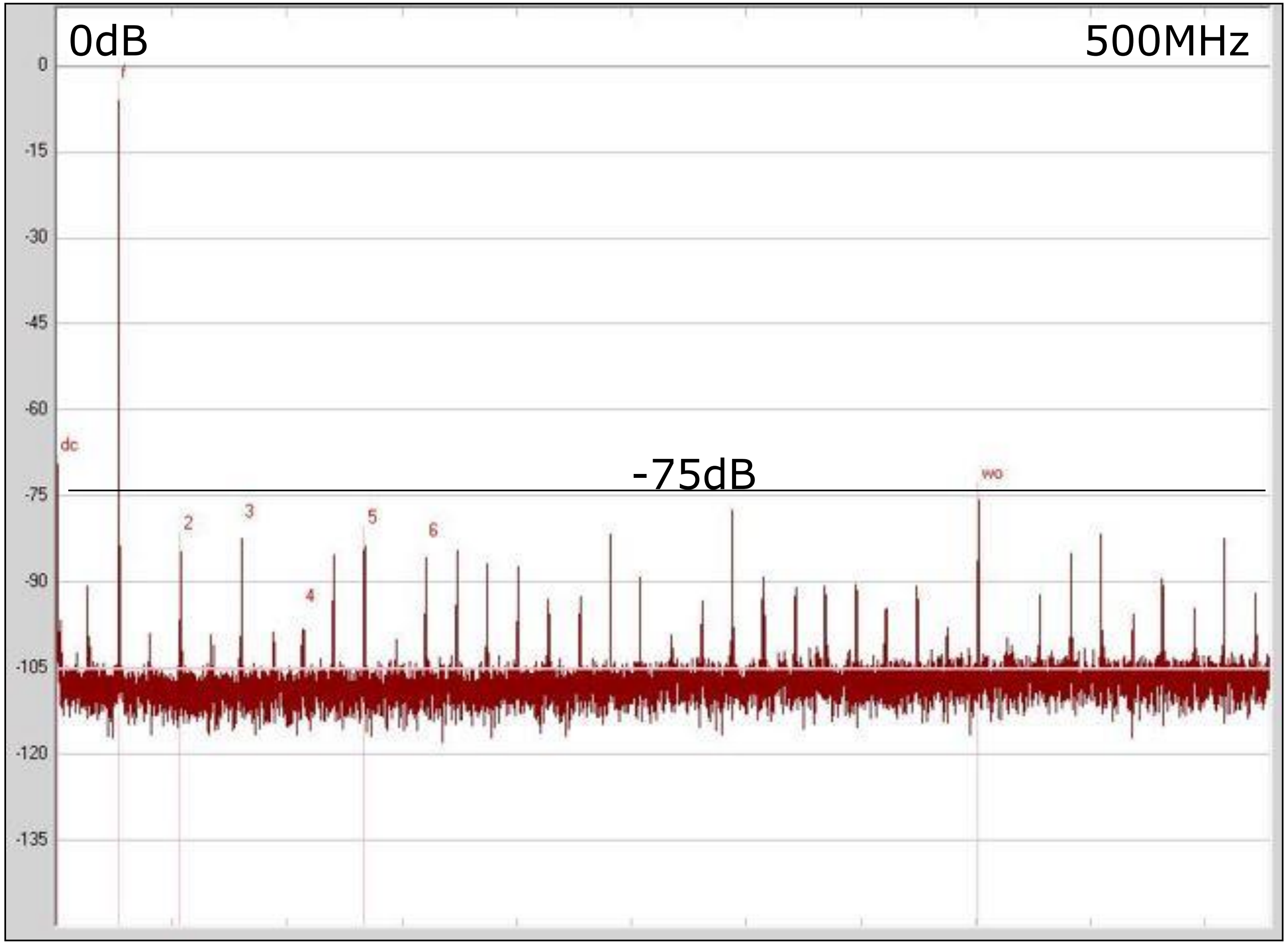
INL without Kick-back Calibration



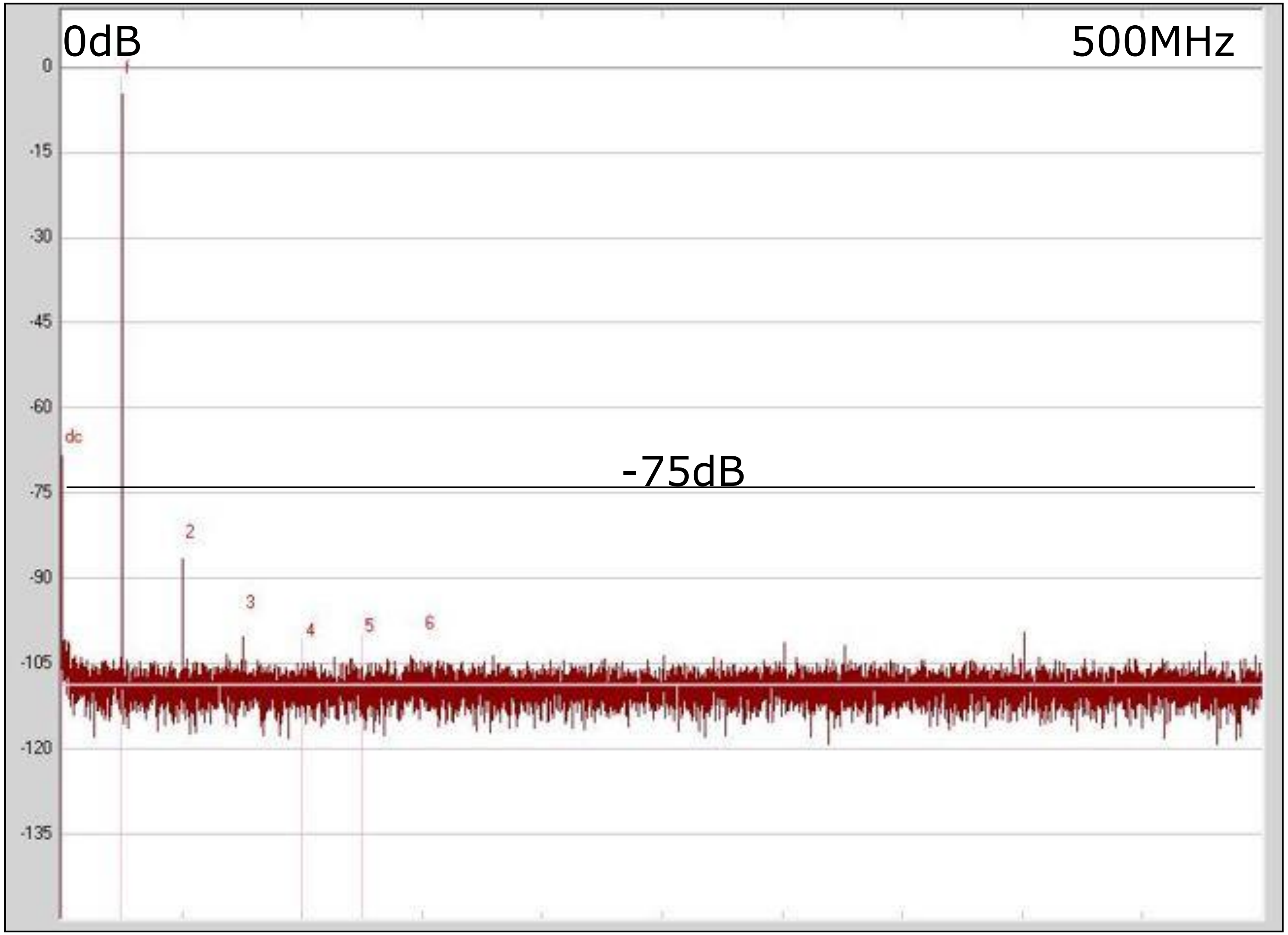
INL with Kick-back Calibration



FFT without kick-back calibration



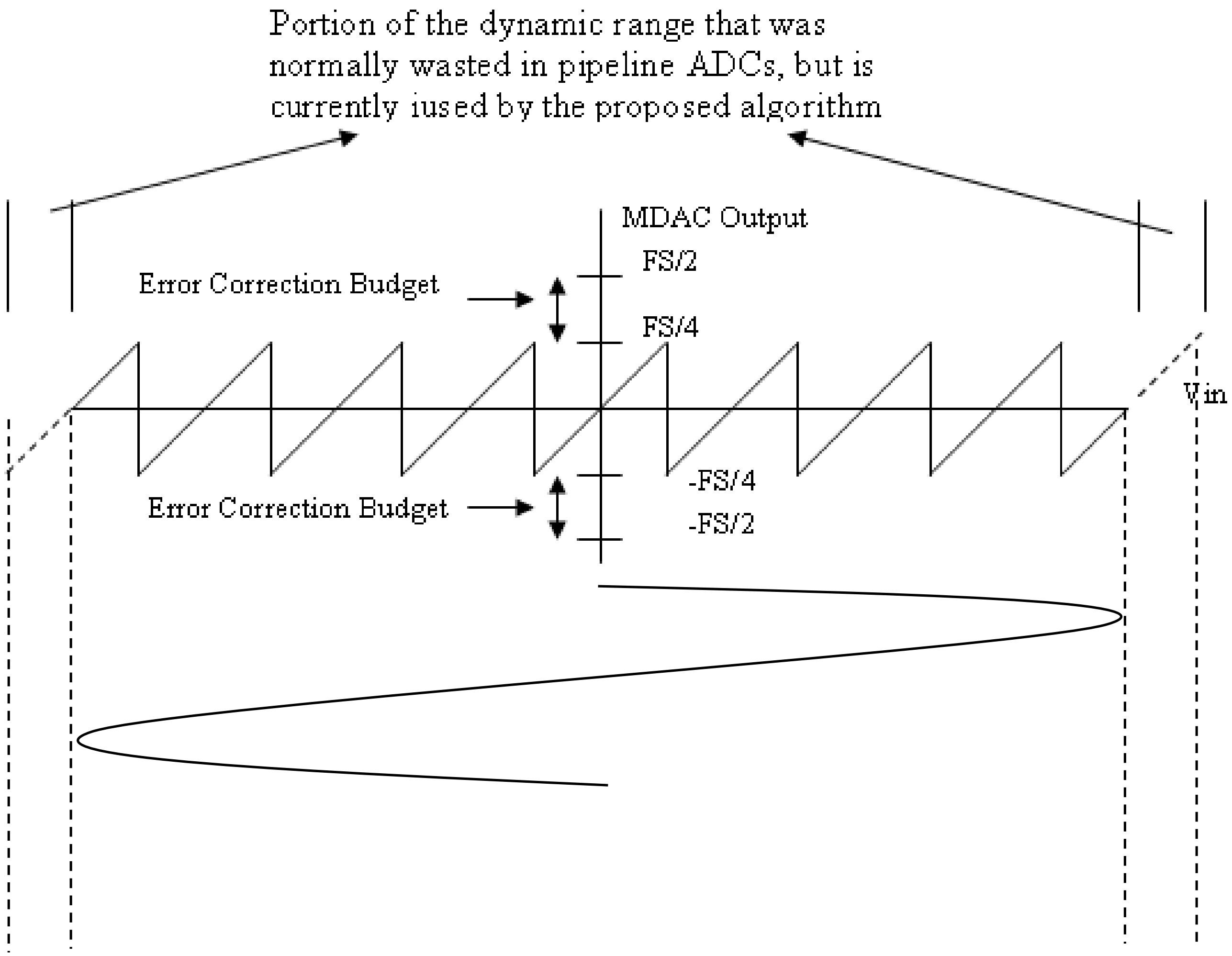
FFT with kick-back calibration



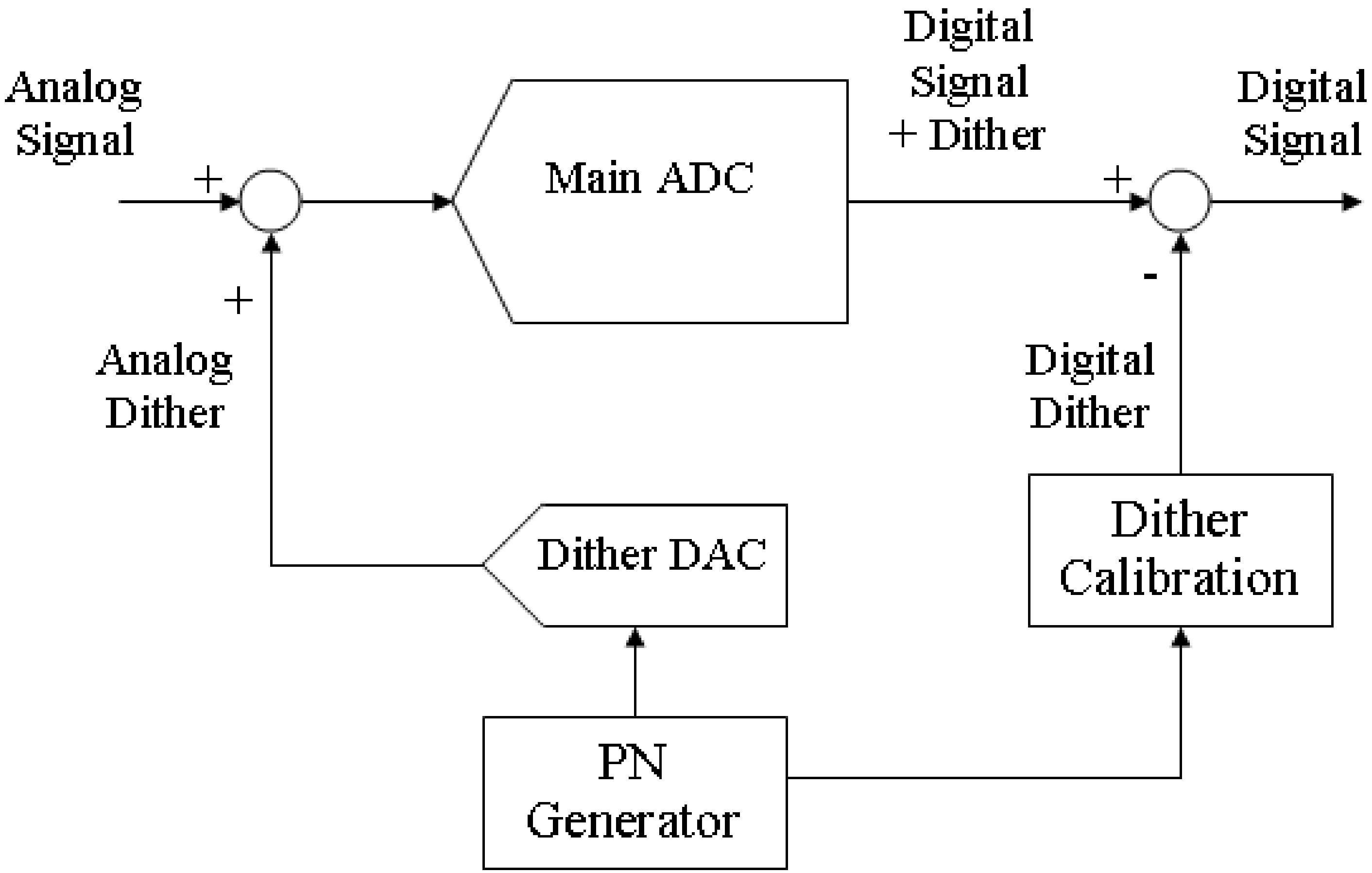
Is this dithering adequate?

- So far, multi-level dithering was used to:
  - Improve calibration accuracy
  - Dither the backend pipeline effectively
- How about dithering the front-end stages?
- Subtractive dither with large amplitude can be useful for dithering the front-end stages
- This “large” dither should not consume dynamic range

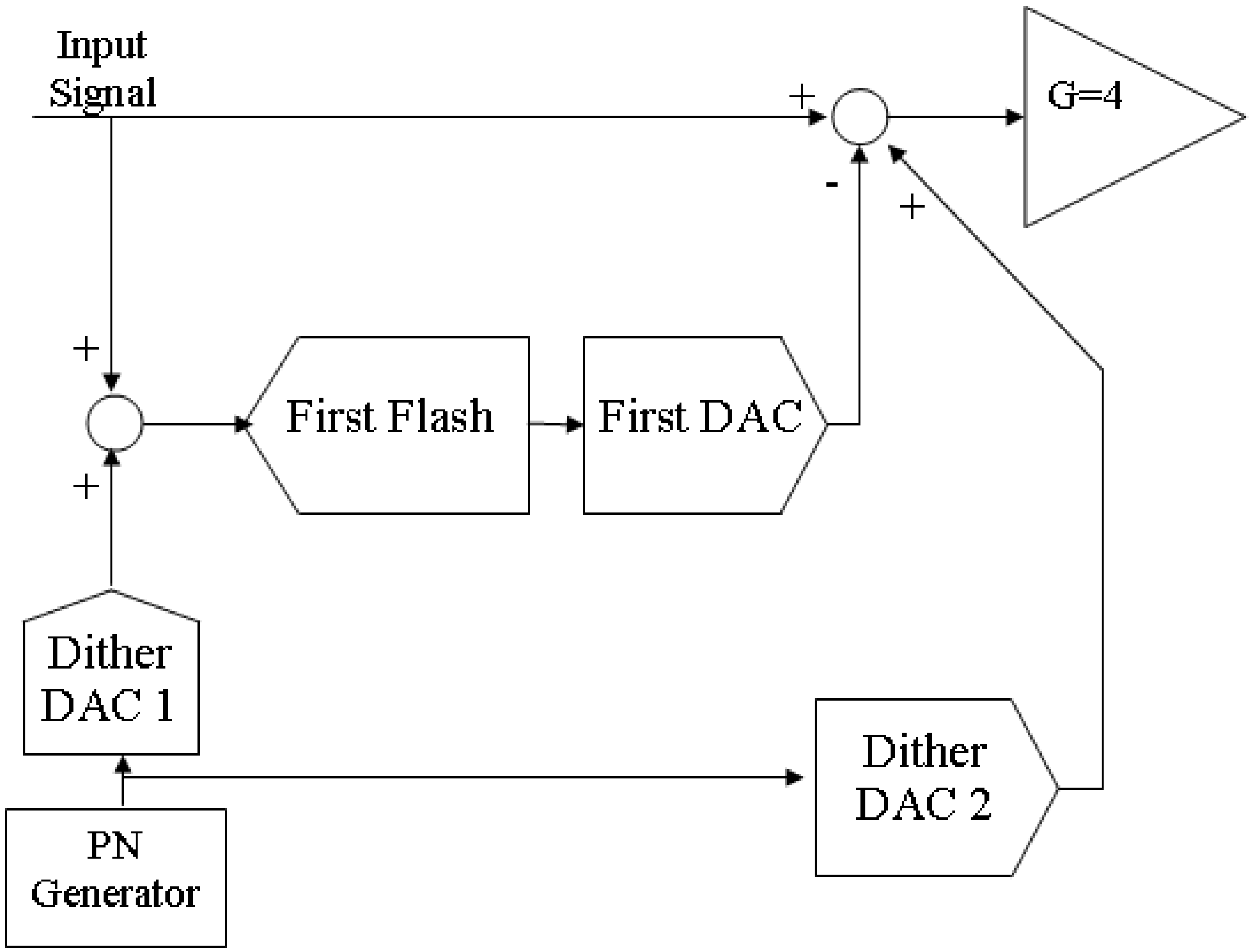
Large Dither Injection



Large Dither Injection and Subtraction

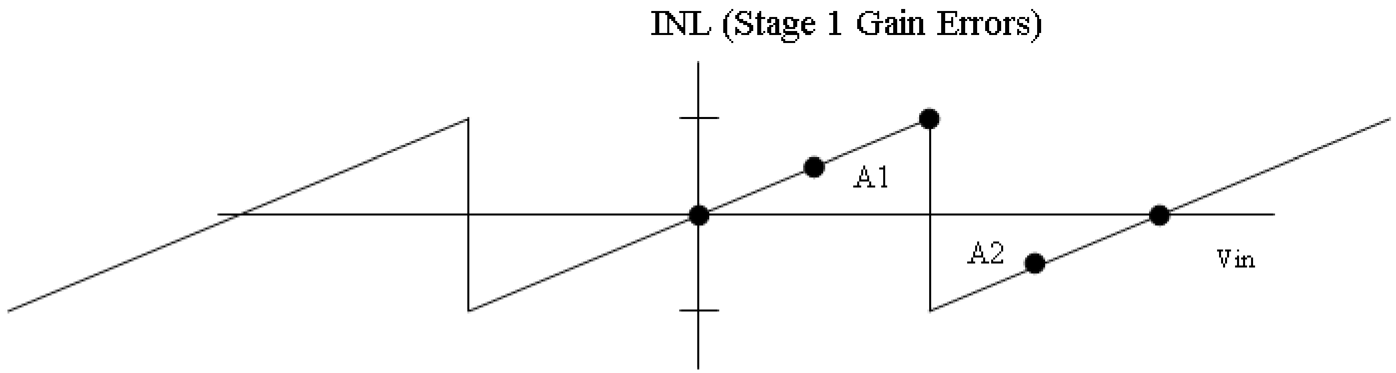


Dither Injection for a SHA-less ADC



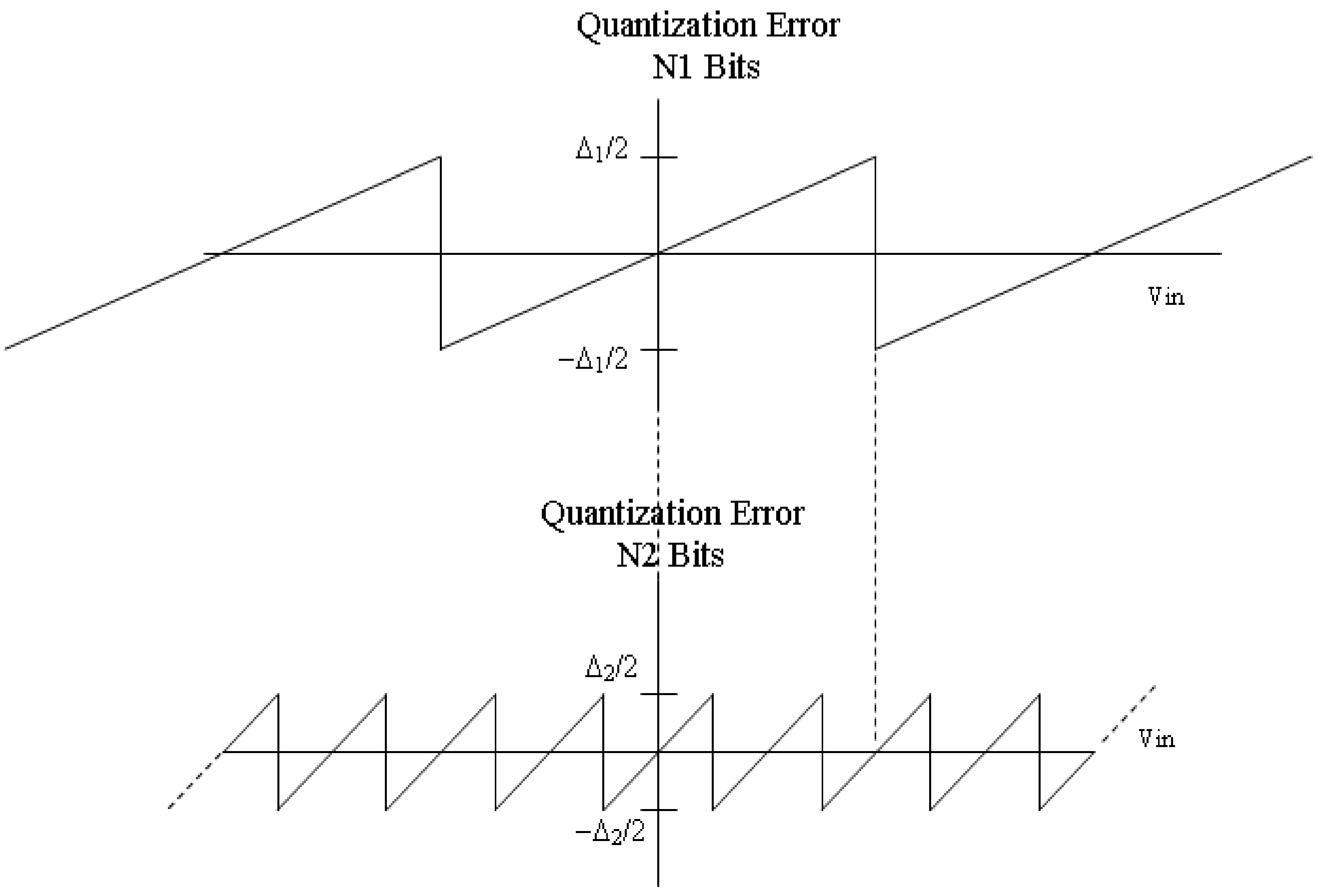
Dither Action with IGE

- IGE creates a saw-tooth pattern
- If A1 is equal to A2, then the dither will randomize (i.e. linearize the IGE)



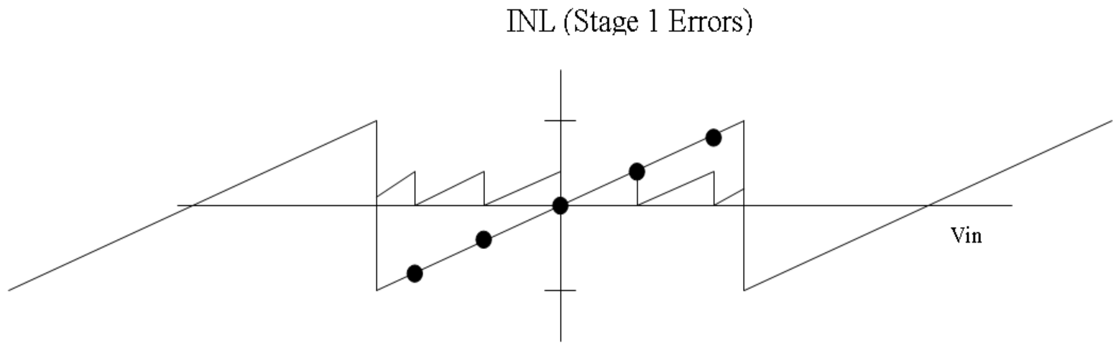
Effect of Saw-tooth Error on SFDR

- SNR = 6N + K1
- SFDR ≈ 9N – K2
- For every additional bit, the spur energy drops by 6 dB and spreads on double the spurs, which is an additional 3 dB



Effect of Dither on SFDR

- SFDR Improves by about 9n = 30 log N
- SFDR of the notch improves by 6n = 20 log N
- The location of the notch moves by 20 log N

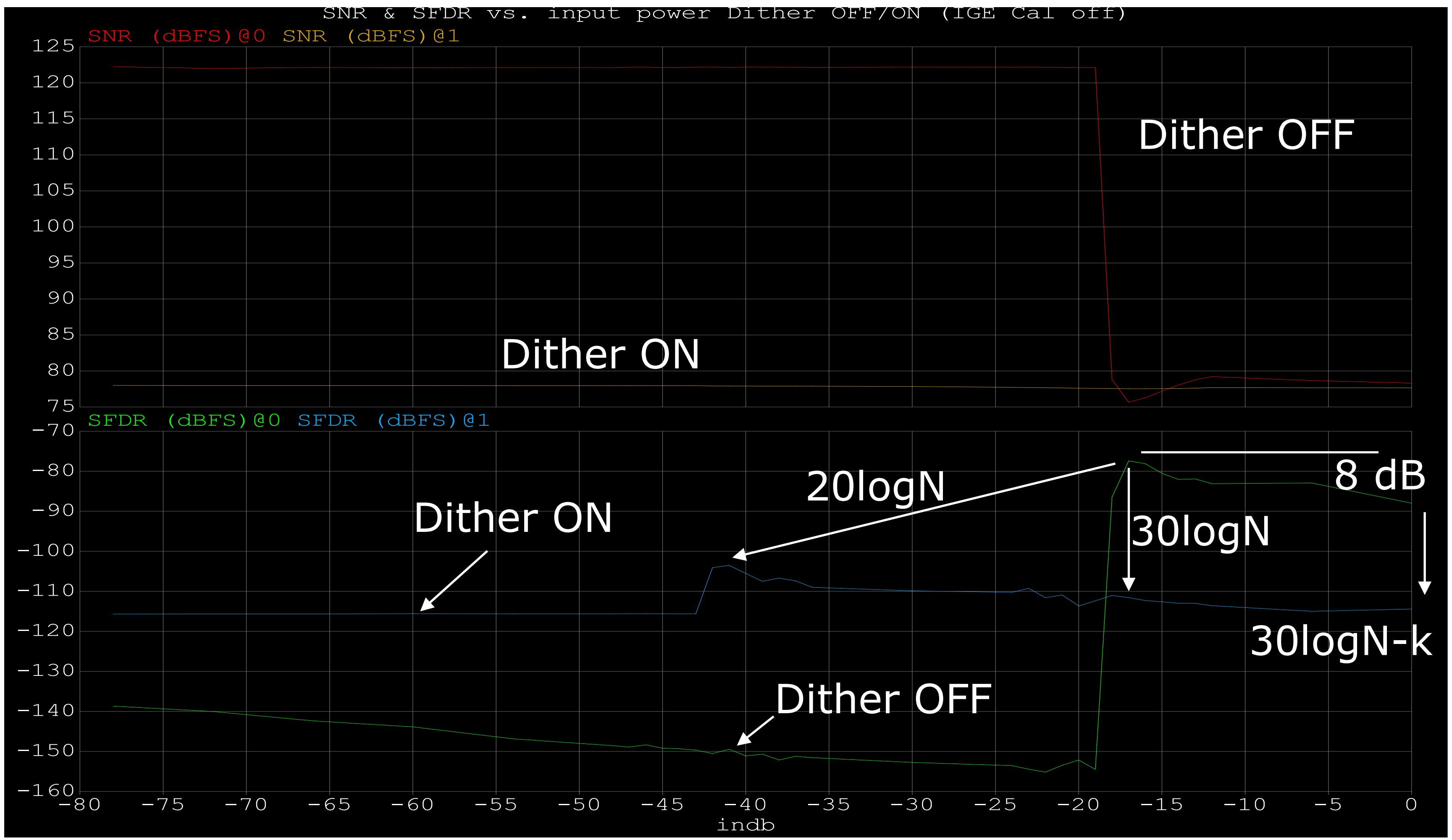


- Where n is the number of dither bits
- And N is the number of dither levels
- For every bit of dither, the SFDR improves by 6 to 9 dB

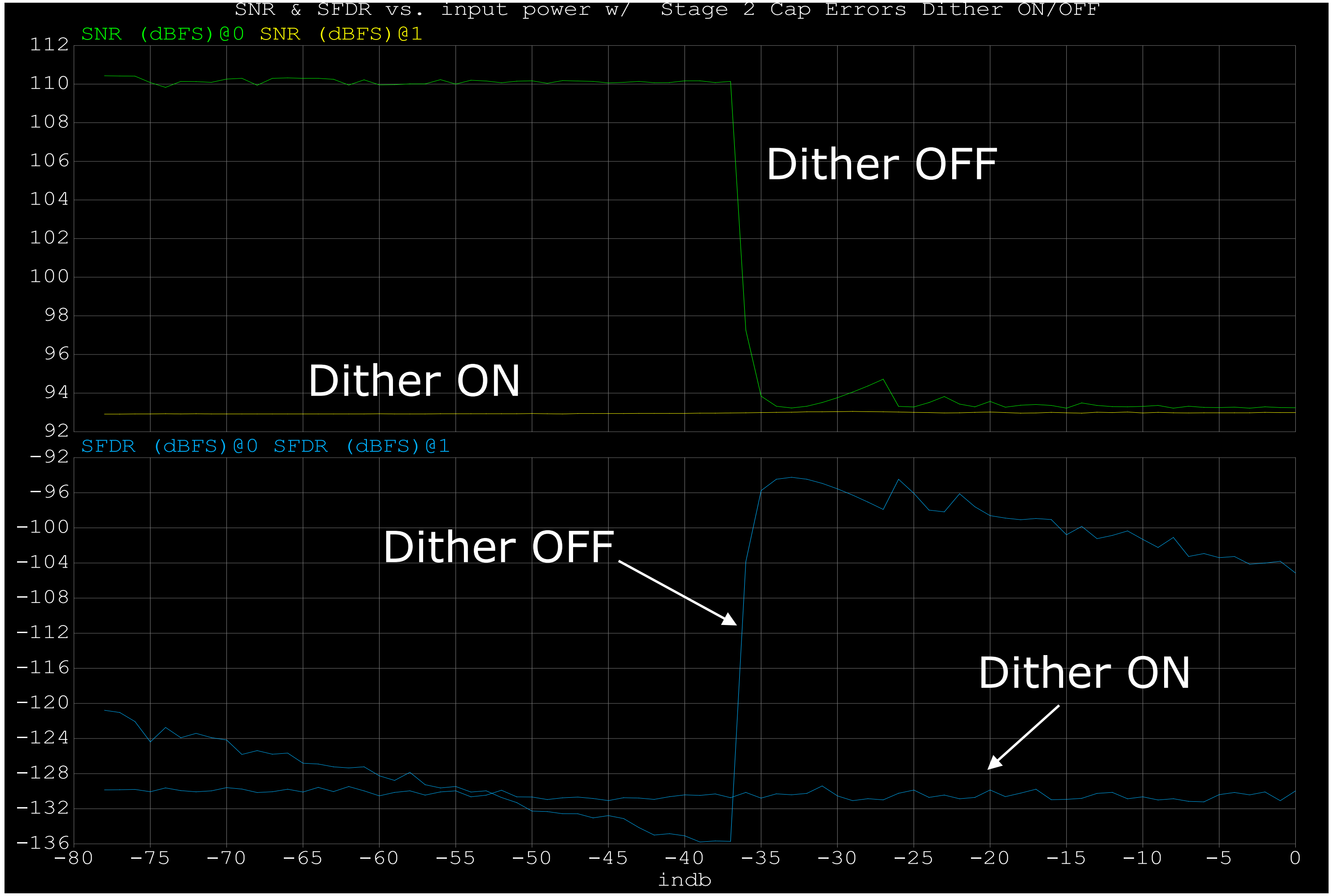
SFDR/SNR Sweep

Ahmed Ali

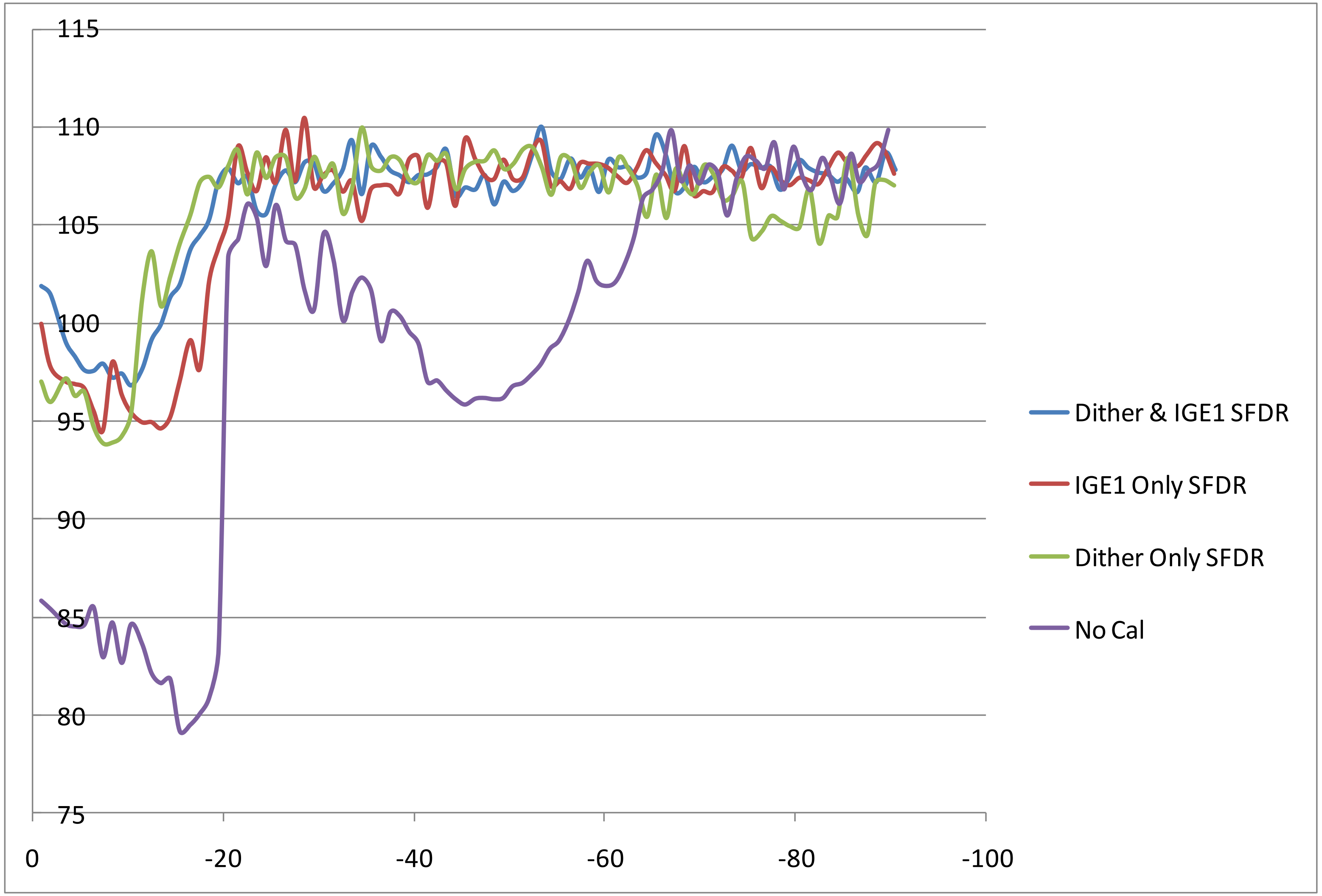
SFDR/SNR Sweep



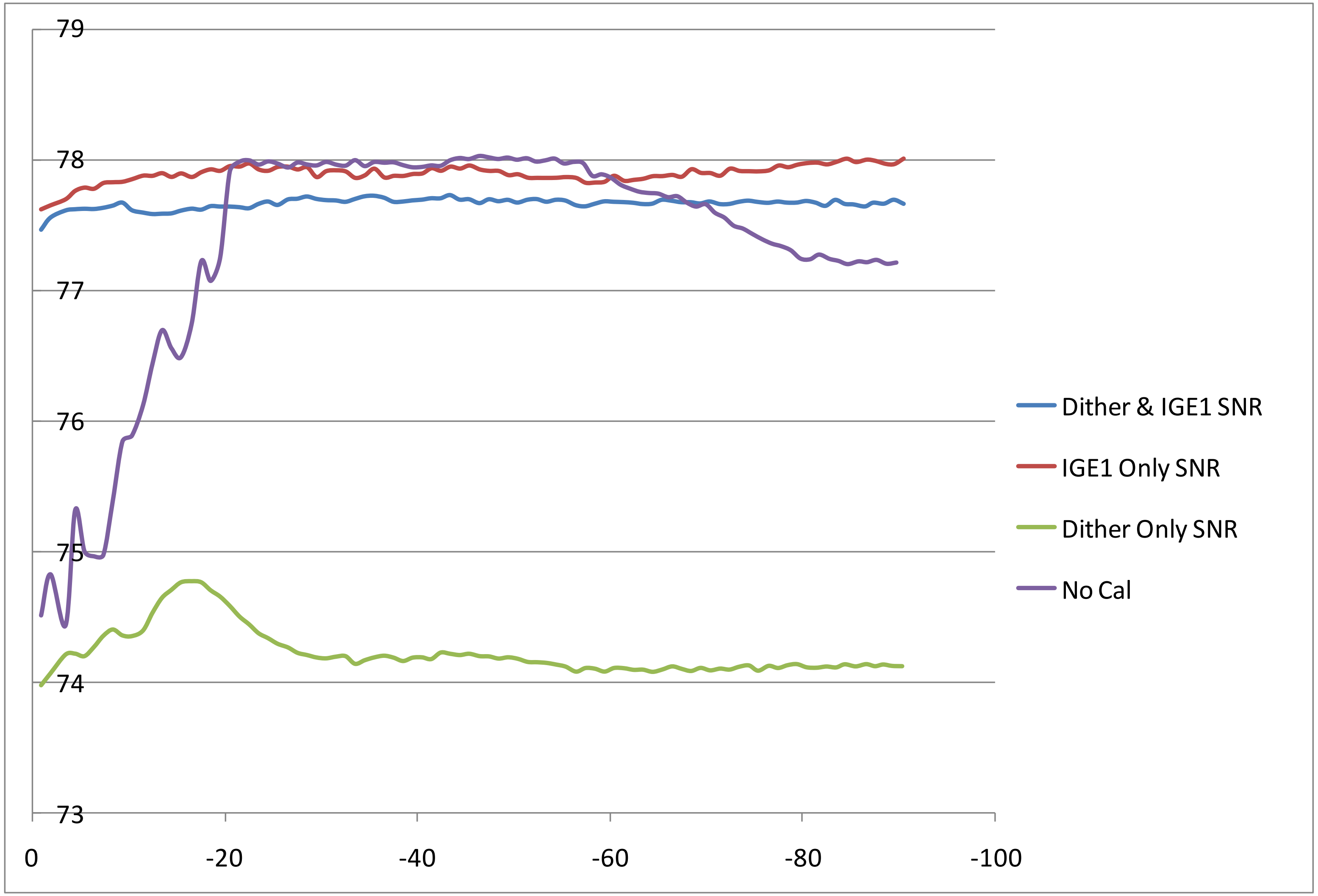
DAC Errors in Stage-2



Input Amplitude Sweep (SFDR)



Input Amplitude Sweep (SNDR)



Challenges (Recap)

- What is the accuracy of this approach?
- How is the power consumption?
- Is it robust?
- How about the convergence time?

Addressing the Challenges

- Using the above techniques, the required accuracy and robustness for SNDR can be achieved
- Dithering can be used to improve the linearity (SFDR) further
- Power overhead is minimized by reducing the dither amplitude
- Utilizing dither improves robustness and reduces power consumption
- Convergence time is in the order of seconds for linear correction and can be reduced in special cases [14]
- Convergence time for the non-linear correction can be in minutes!
  - Deterministic methods can be more attractive for non-linear correction

Conclusion

- Digital assistance can be in the form of fixing errors or dithering errors
- Calibration of the IGE, DAC, IME, and kick-back errors are examples of using DSP to improve the analog performance
- Dither can be used to improve the calibration accuracy and improve the linearity further
- Digital assistance enables ADC designers to continue building efficient, accurate and fast ADCs as process geometry shrinks

References

- [1] A.M.A. Ali, “High speed data converters”, Institution of Engineering and Technology (IET), London, UK, 2016.
- [2] A.M.A. Ali, et al., “A 14b 1GS/s RF sampling pipelined ADC with background calibration”, J. Solid-State Circuits, 49, no. 12, pp. 2846-2856, Dec. 2014.
- [3] A.M.A. Ali, et al., “A 16-bit 250-MS/s IF Sampling Pipelined ADC With Background Calibration”, J. Solid-State Circuits, 45, no. 12, pp. 2602-2612, Dec. 2010.
- [4] A.M.A. Ali, et al., “A 14-bit 2.5GS/s and 5GS/s RF sampling ADC with background calibration and dither”, IEEE VLSI Symp., 2016.
- [5] P. Huang et al., “SHA-Less Pipelined ADC With In Situ Background Clock-Skew Calibration”, IEEE J. Solid State Circuits, 46(8), ISSCC Dig. Tech. Papers, pp. 1893-1903, Aug. 2011.
- [6] M. Brandolini et al., “A 5GS/s 150mW 10b SHA-Less Pipelined/SAR Hybrid ADC in 28nm CMOS”, ISSCC Dig. Tech. Papers, pp. 468-469, Feb. 2015.
- [7] A. Panigada and I. Galton, “A 130mW 100MS/s pipelined ADC with 69dB SNDR enabled by digital harmonic distortion correction”, ISSCC Dig. Tech. Papers, pp. 162-163, Feb. 2009.
- [8] N. Rakuljic and I. Galton, “Suppression of Quantization-Induced Convergence Error in Pipelined ADCs with Harmonic Distortion Correction”, IEEE Trans. Circuits and Systems I, 60(3), pp. 593-602, March 2013.
- [9] A.M.A. Ali, “Methods and structures that reduce memory effects in analog-to-digital converters”, US Patent No 6,861,969. March, 2005.
- [10] J.P. Keane, “Digital Background Calibration for Memory Effects in Pipelined Analog-to-Digital Converters”, IEEE TCAS-I, March, 2006.

References (cont.)

- [11] A.M.A. Ali and A. Morgan, “Calibration methods and structures for pipelined converter systems”, US Patent 8,068,045, November, 2011.
- [12] A.M.A. Ali, et al., “Correlation-based background calibration of pipelined converters with reduced power penalty”, US Patent 7,786,910, August, 2010.
- [13] A.M.A. Ali, “Pipelined converter systems with enhanced accuracy”, US Patent 7,271,750, September, 2007.
- [14] A.M.A. Ali, “Method and device for improving convergence time in correlation-based algorithms ”, US Patent 8,836,558, September, 2014.
- [15] E. Siragusa and I. Galton, “A Digitally Enhanced 1.8-V 15-bit 40-MSample/s CMOS Pipelined ADC,” IEEE J. Solid State Circuits, vol. 39, Dec. 2004.
- [16] S. Sonkusale, et al., “True background calibration technique for pipelined ADC”, Electronic Letters, pp. 786-788, 36(9), 2000.
- [17] Y. Chiu et al., “Least Mean Square adaptive digital background calibration of pipelined analog-to-digital converters”, IEEE TCAS-I, Jan. 2004.
- [18] J.A. McNeill et al.,” “Split ADC” Architecture for Deterministic Digital Background Calibration of a 16-bit 1-MS/s ADC”, IEEE TCAS-I, Dec., 2005.
- [19] E. Iroaga and B. Murmann, “A 12-Bit 75-MS/s Pipelined ADC Using Incomplete Settling”, J. Solid-State Circuits, Vol. 42, no. 4, April 2007.
- [20] B. Murmann and B.E. Boser, “A 12 b 75 MS/s pipelined ADC using open-loop residue amplification”, ISSCC Dig. Tech. Papers, pp. 328 – 497, Vol.1, 2003.
- [21] B. Murmann and B.E. Boser, “Digital Domain Measurement and Cancellation of Residue Amplifier Nonlinearity in Pipelined ADCs”, IEEE Trans. Inst. Meas., Dec. 2007.