T3: Fundamentals of Data Converters

Yun Chiu

(chiu.yun@utdallas.edu)

University of Texas at Dallas
Richardson, TX

Feb 19, 2023

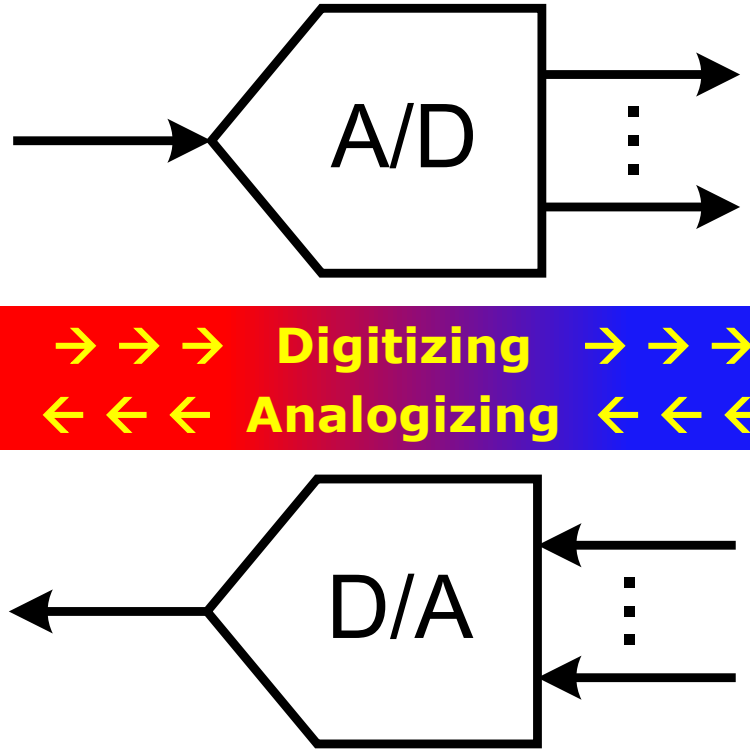
Outline

- Quantization and Sampling
- Overview of Data Converter Architectures
- How To Measure Data Converter Performance?
- Nyquist-Rate ADC and DAC Architectures
- Circuit Building Blocks
- Redundancy
- Digital Correctability and Calibration
- ADC Figure-of-Merits
- Data Converter Papers to See This Year
- References

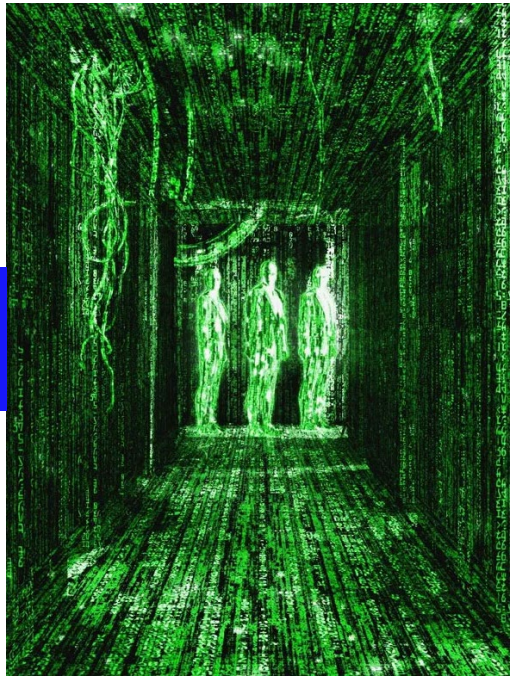
Data Conversion

CT, CA

Analog World



DT, DA



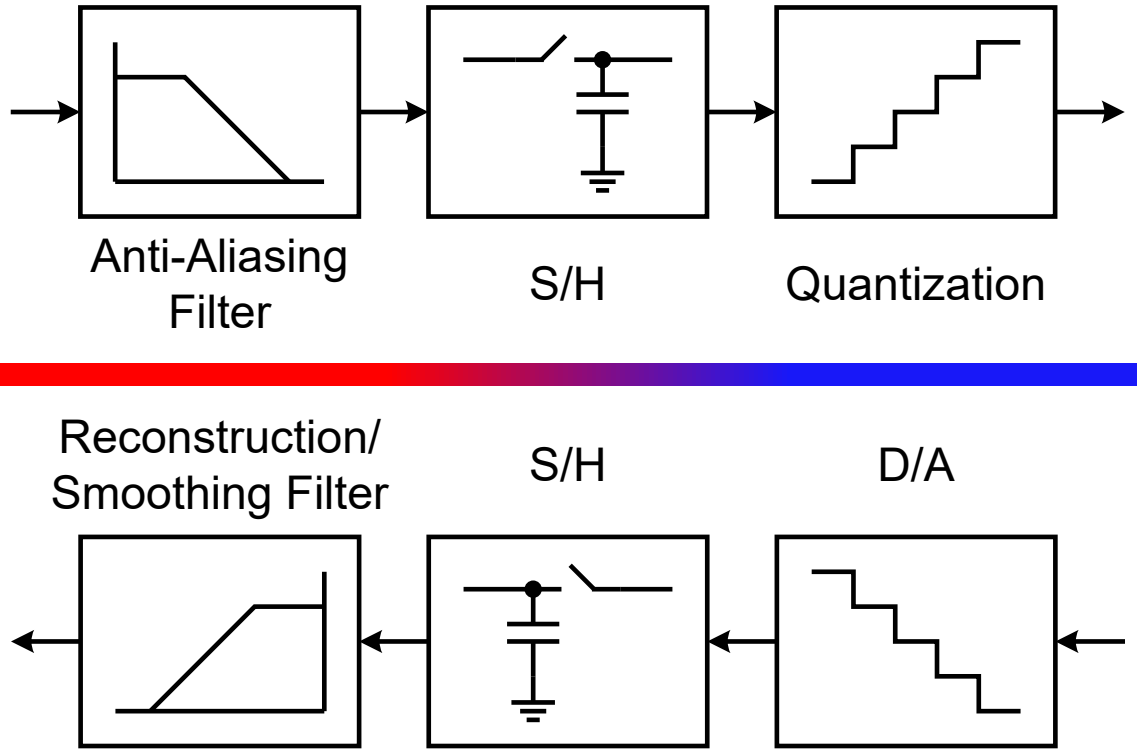
Digital World

- Data converters bridge analog (physical) world and digital (virtual) world

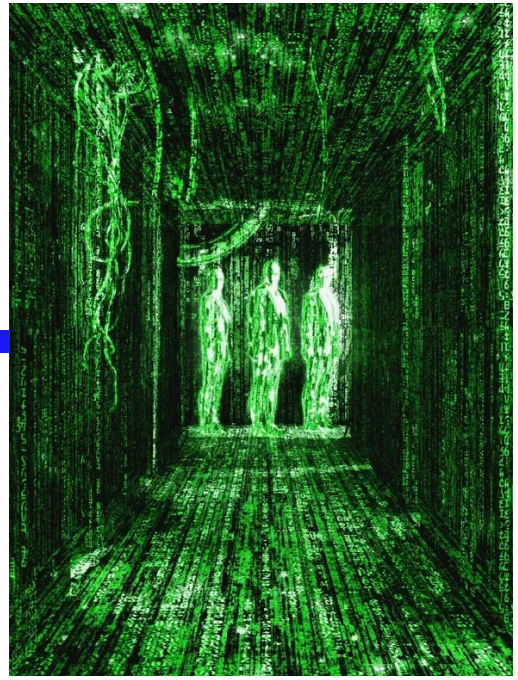
Data Conversion

CT, CA

Analog World



DT, DA



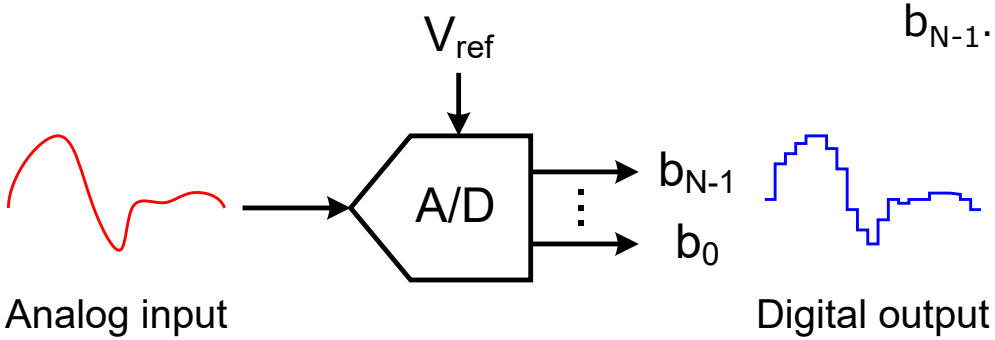
Digital World

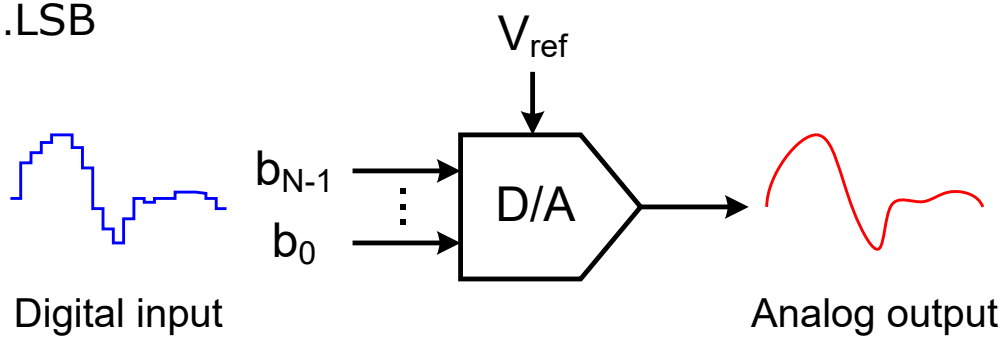
- Three key functions involved: **Filtering, Sampling, Quantization**

QUANTIZATION

A Mathematical View

$b_{N-1}...b_0$: MSB...LSB





$$D_{out} = \left\lfloor \frac{V_{in}}{V_{FS}} 2^N \right\rfloor = \left\lfloor \frac{V_{in}}{\Delta} \right\rfloor$$

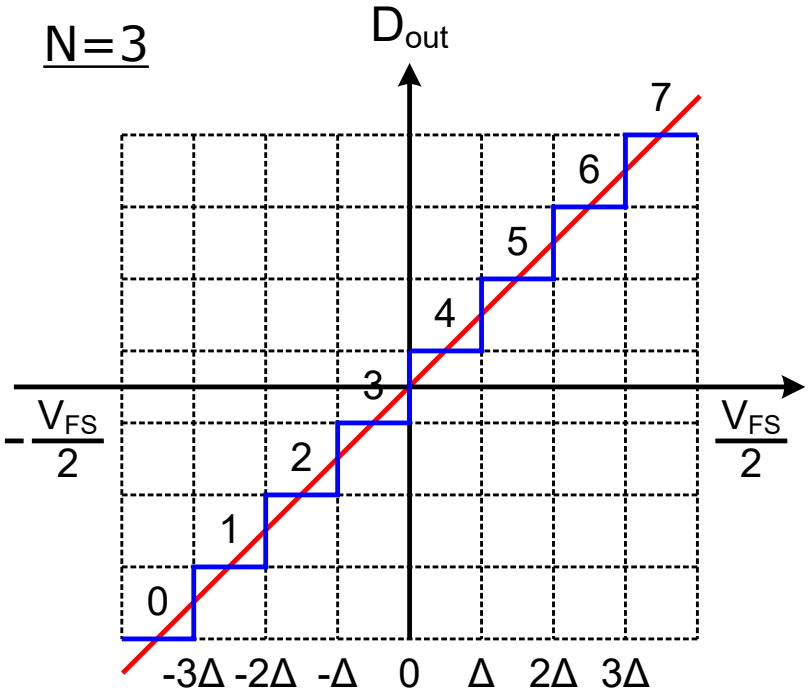
Division

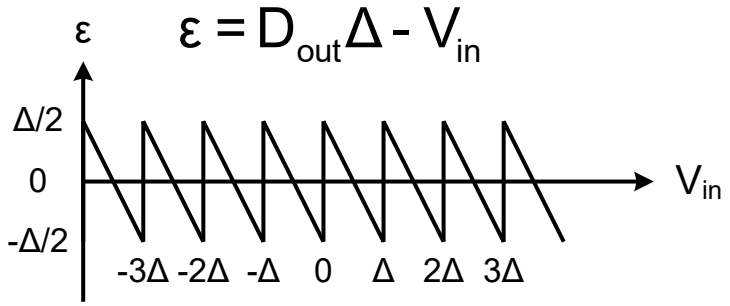
$$V_{out} = \sum_{i=0}^{N-1} b_i \cdot 2^i \left(\frac{V_{FS}}{2^N} \right) = \sum_{i=1}^{N} b_i 2^{i-1} \Delta$$

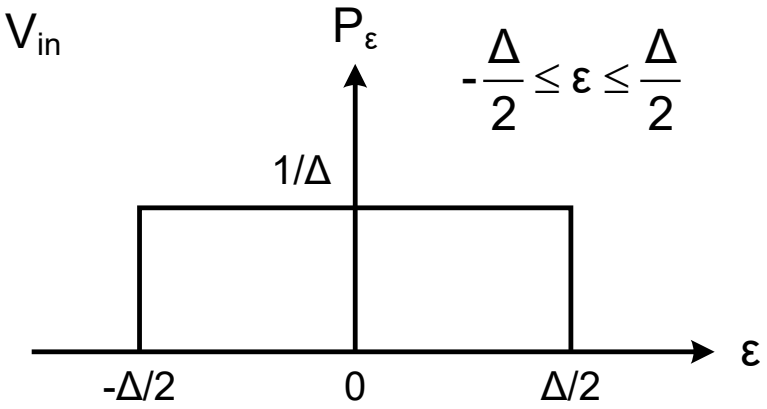
Multiplication

- N = # of bits, V_{FS} = Full-Scale range, $\Delta = V_{FS}/2^N$ = 1 LSB, $b_i = \{0, 1\}$
- Full-Scale range (V_{FS}) is set by reference voltage V_{ref}

Quantization Error







Assumptions:

- ❑ N is large (Δ is small)
- ❑ V_{in} » Δ and is active
- ❑ ε is uniformly distributed
- ❑ Spectrum of ε is white

$$\sigma_\varepsilon^2 = \int_{-\Delta/2}^{\Delta/2} \varepsilon^2 \cdot \frac{1}{\Delta} \cdot d\varepsilon = \frac{\Delta^2}{12}$$

Ref. [1]

- ❑ "Random" **quantization error** is usually regarded as **quantization noise**

Signal-to-Quantization Noise Ratio (SQNR)

- For sinusoidal input with $V_{pp}=V_{FS}$

$$SQNR = \frac{V_{FS}^2/8}{\sigma_\varepsilon^2} = \frac{(2^N\Delta)^2/8}{\frac{\Delta^2}{12}} = 1.5\times 2^{2N}$$

$$SQNR = 6.02\times N + 1.76 \ [dB]$$

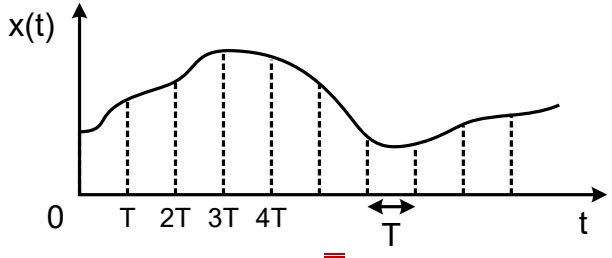
N (bits)	SQNR (dB)
8	49.9
10	62.0
12	74.0
14	86.0
…	…

- SQNR depicts the theoretical performance of an ideal converter
- Practical converter performance can be limited by many other factors
  - Noise: thermal, 1/f, supply/substrate/coupling, etc.
  - Distortion: DC & AC nonlinearities, measured by THD, SFDR, IM3, etc.
  - Metastability (ADC): comparator fails to resolve within time limit

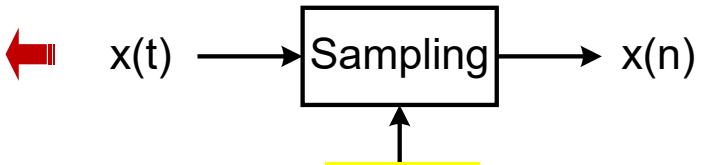
SAMPLING

Sampling (quite some math)

Continuous Time

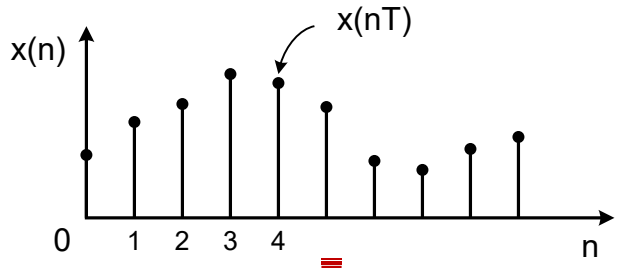


$\Omega T = \omega$



Sample rate: $f_s = 1/T$

Discrete Time

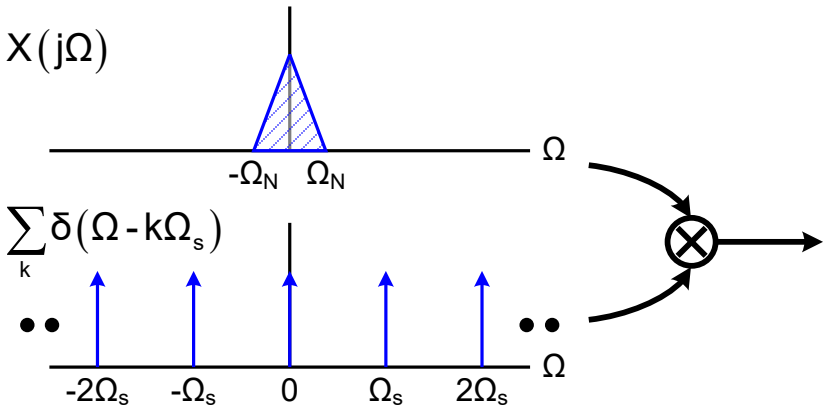


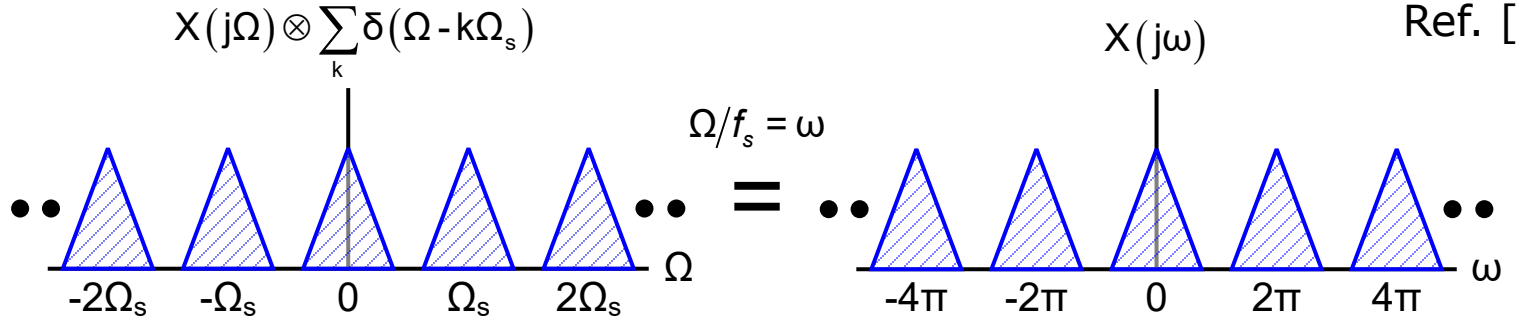
$$x(t) \overset{FT}{\rightleftharpoons} \int_{-\infty}^{\infty} x(t)e^{-j\Omega t}dt = X(j\Omega)$$

$$X(j\omega) = \frac{1}{T}X(j\Omega) \otimes \sum_k \delta(\Omega - k\Omega_s)$$

impulse train

$$x(n) \overset{ZT}{\rightleftharpoons} \sum_{n=-\infty}^{\infty} x(n)z^{-n}\Big|_{z=e^{j\omega}} = X(j\omega)$$



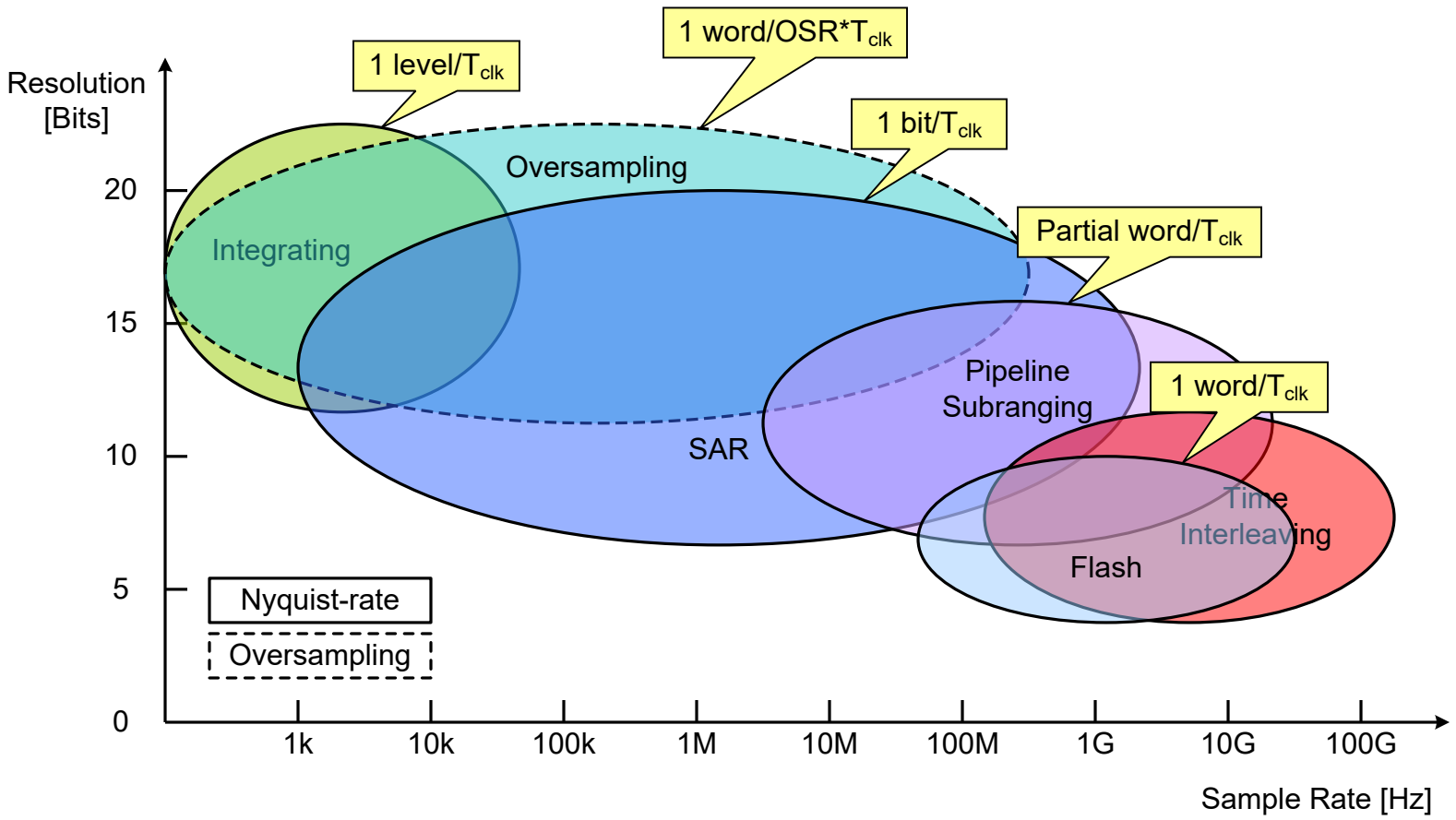


Ref. [2]

No aliasing if $f_{s,min} \geq 2f_N$ → Nyquist-rate sampling

OVERVIEW OF DATA CONVERTER ARCHITECTURES

Overview of ADC Architectures



SAR, ΔΣM, TI ADC array are popular lately!

Per-step resolving (Conv. time)

- N bits† (1 step) ← fast
  - **Flash**
  - Folding
- 1 level (2^N steps) ← slowest
  - Integration (Serial)
- 1 bit (N steps) ← slow
  - **SAR**
  - Algorithmic (Cyclic)
- M bits (N/M steps) ← medium
  - **Pipeline**
  - Subranging
- Others
  - Time interleaving
    → high-speed ADC array
  - Oversampling + NS
    → ΔΣ modulator

Overview of DAC Architectures

- Nyquist-rate DACs (oversampling DACs skipped in this tutorial):

Technology \ Topology	Binary-weighted (B.W.)	Unit-element (U.E.)	Segmented (S.M.)
Resistive	R2R	Resistor string	N.C.
Capacitive	Charge redistribution (can support high speed [3])		
I-steering	N.C.	High speed	High speed & high resolution

- Some (fun or not so fun) facts about DACs:
  - Every ADC has a DAC built in!
  - DACs are conceptually straightforward, but tough to design or calibrate at high speed (GS/s+)!

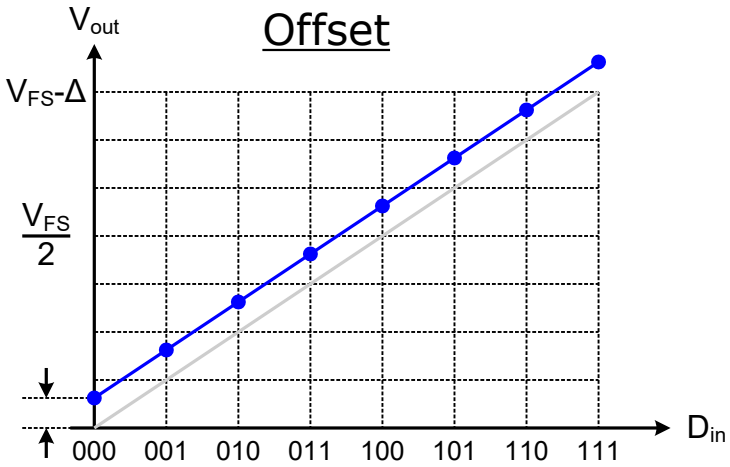
PERFORMANCE MEASUREMENT OF DATA CONVERTERS

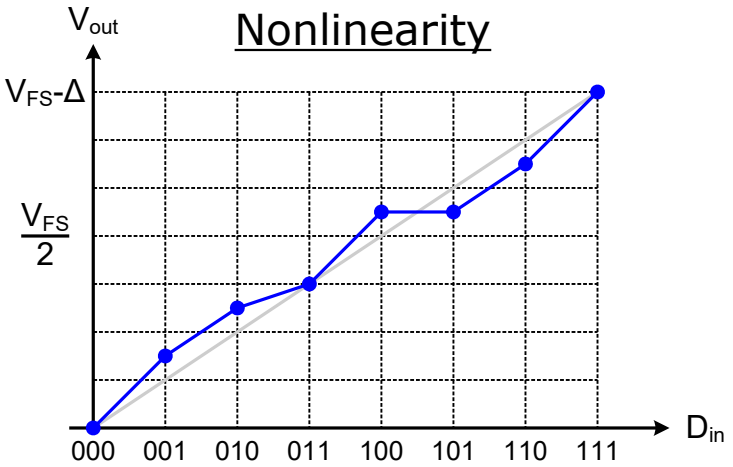
Data Converter Measurements

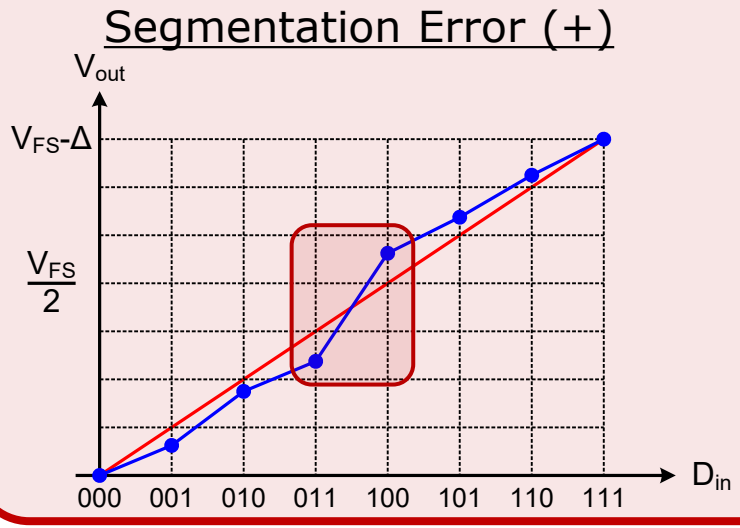
- Static measurement of converter transfer function (TF)
  - Differential Non-Linearity (DNL), Integral Non-Linearity (INL)
  - Code Density Test (CDT)
- Dynamic frequency- / time-domain measurement of signal quality
  - Frequency domain techniques: FFT, SNR, SNDR (SINAD), SFDR, IM3 etc.
  - Time-domain techniques: Sine-fit

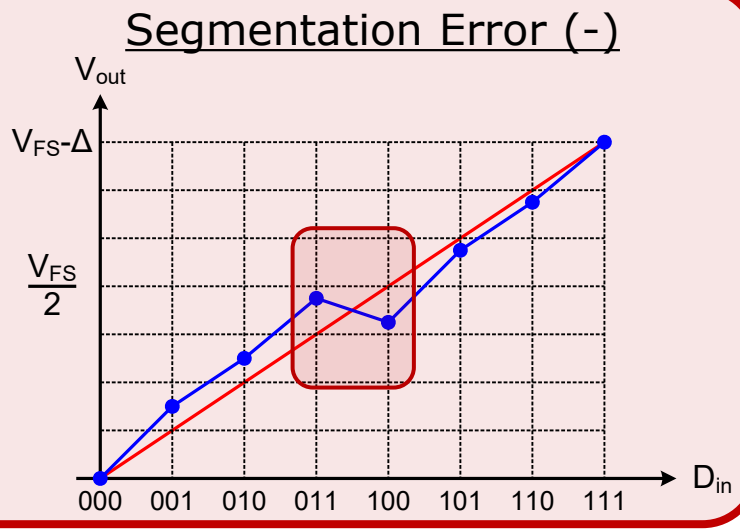
DNL AND INL

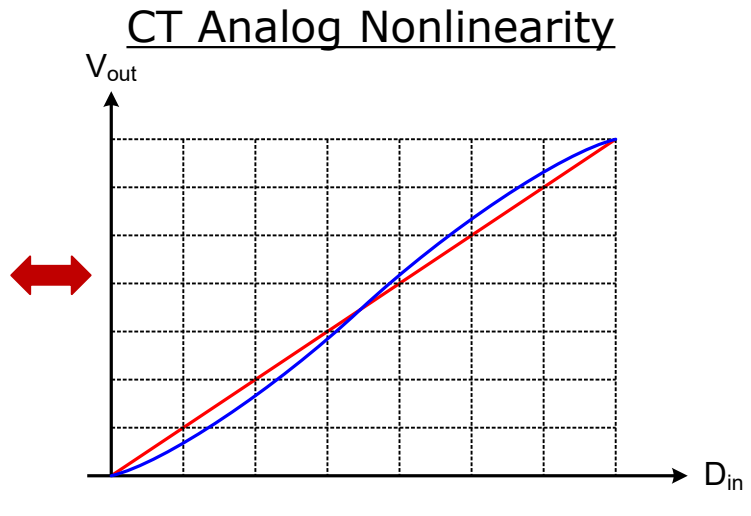
DAC Static Nonlinearity



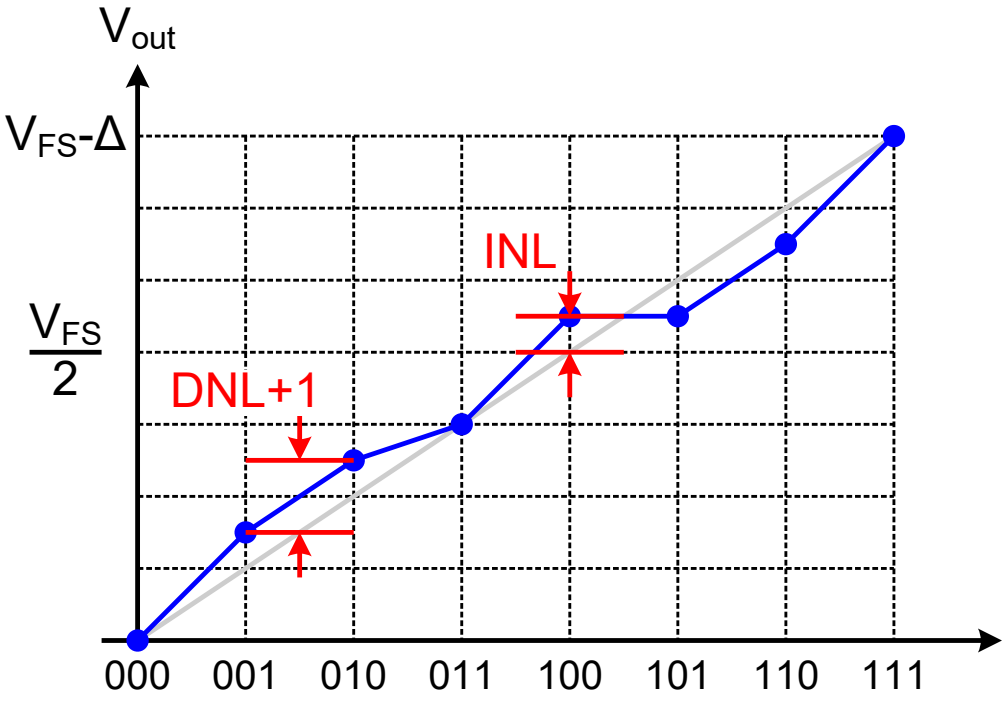






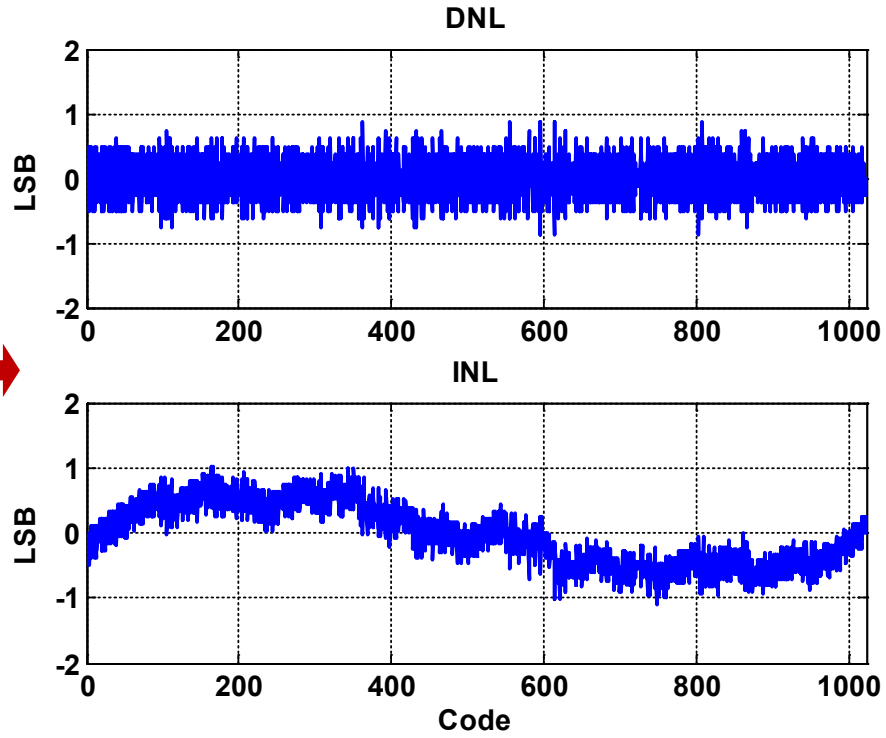


DNL and INL (DAC)



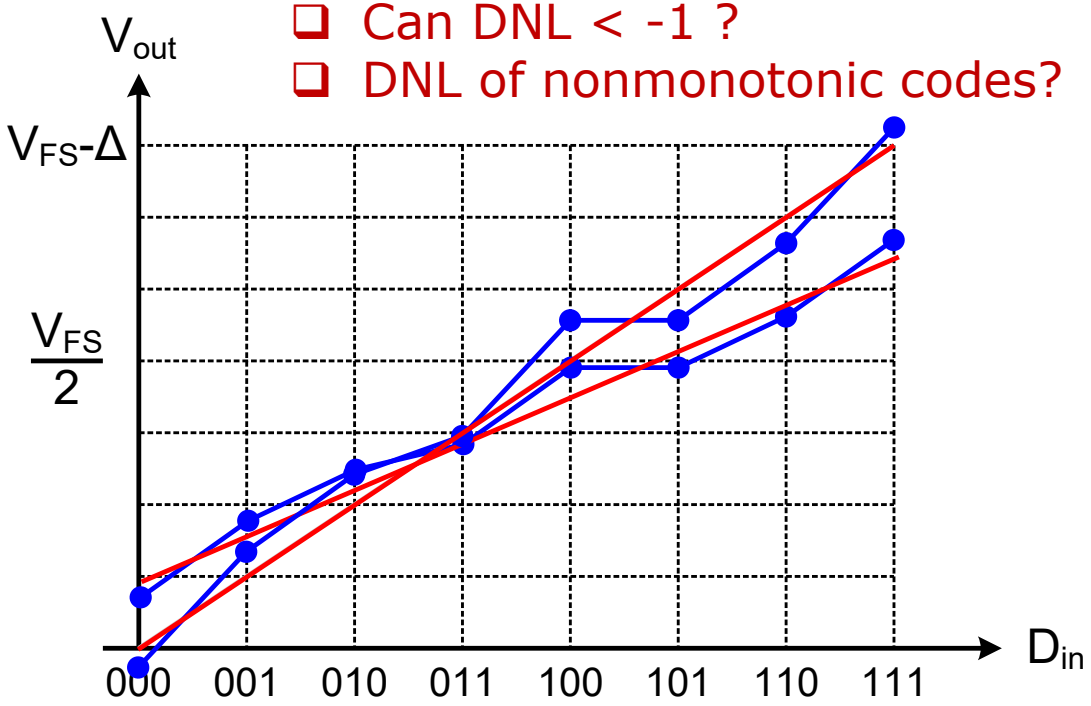
$$DNL_i = \frac{\Delta_i - \Delta}{\Delta}$$

$$INL_i = \sum_{j=0}^{i} DNL_j$$



- DNL: deviation of a conversion step (V_j-V_{j-1}) from 1 LSB (Δ) – **Incremental**
- INL: deviation of the output (V_j) from ideal transfer curve – **Cumulative**

A More Realistic Measurement

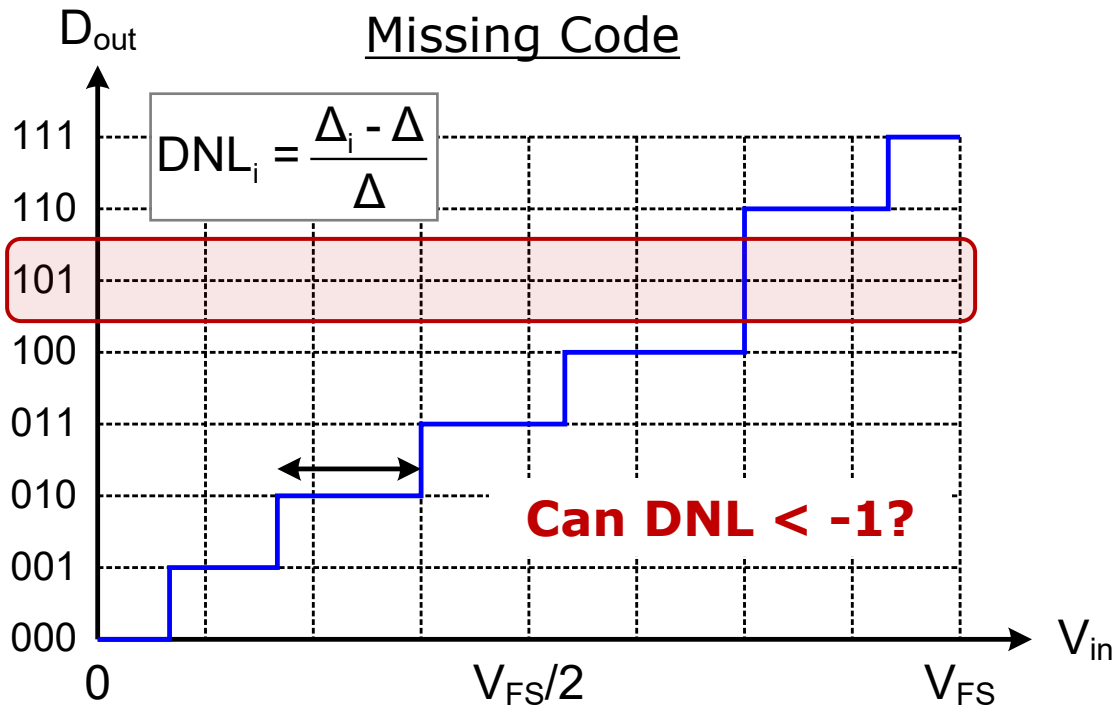


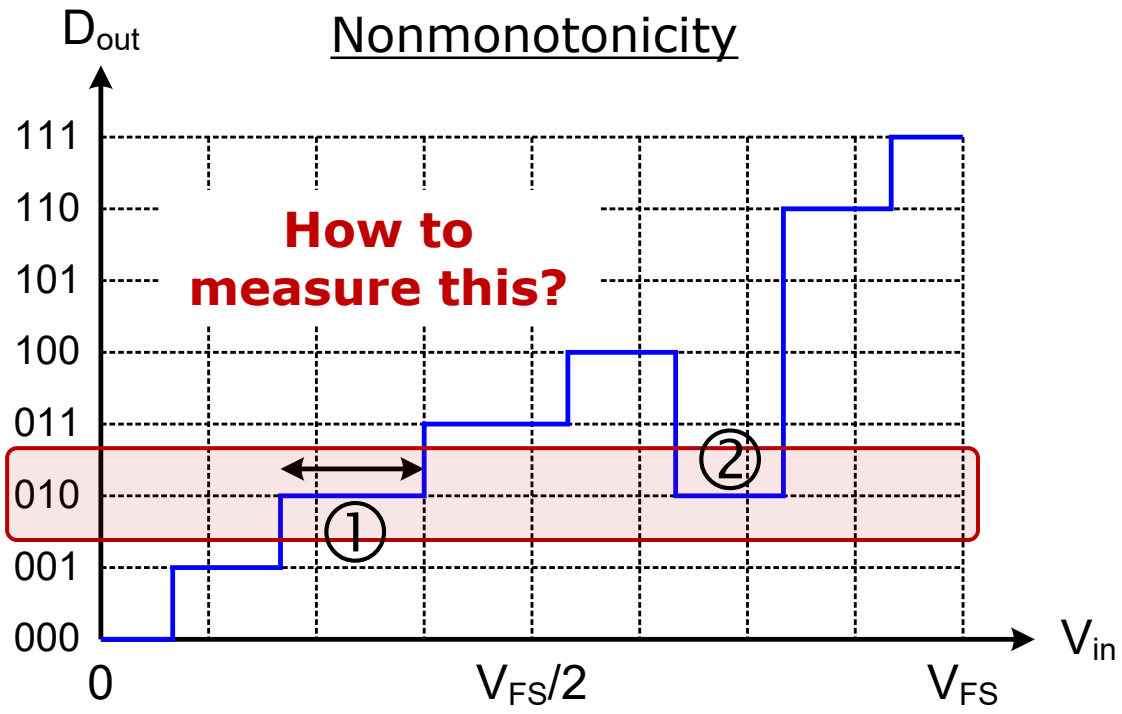
- Can DNL < -1 ?
- DNL of nonmonotonic codes?

- Measurement procedure:
  - Least-square fit transfer curve
  - Stretch fitted line to ideal position
  - Determine DNL and INL
- MATLAB built-in function "detrend"
  - **Σ(INL) = 0** – why?

- In practice, endpoints of TF may not end up at 0 and V_{FS}-Δ →
  Assumption: Gain error and Offset are of less concern than DNL and INL

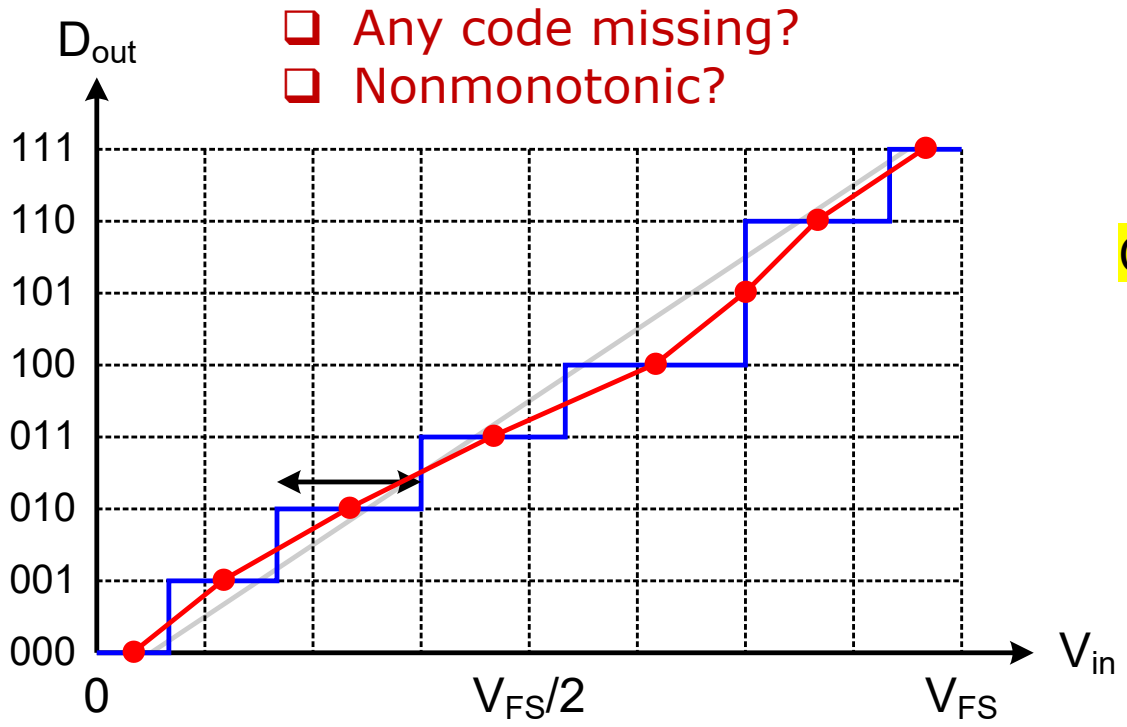
ADC Static Nonlinearity



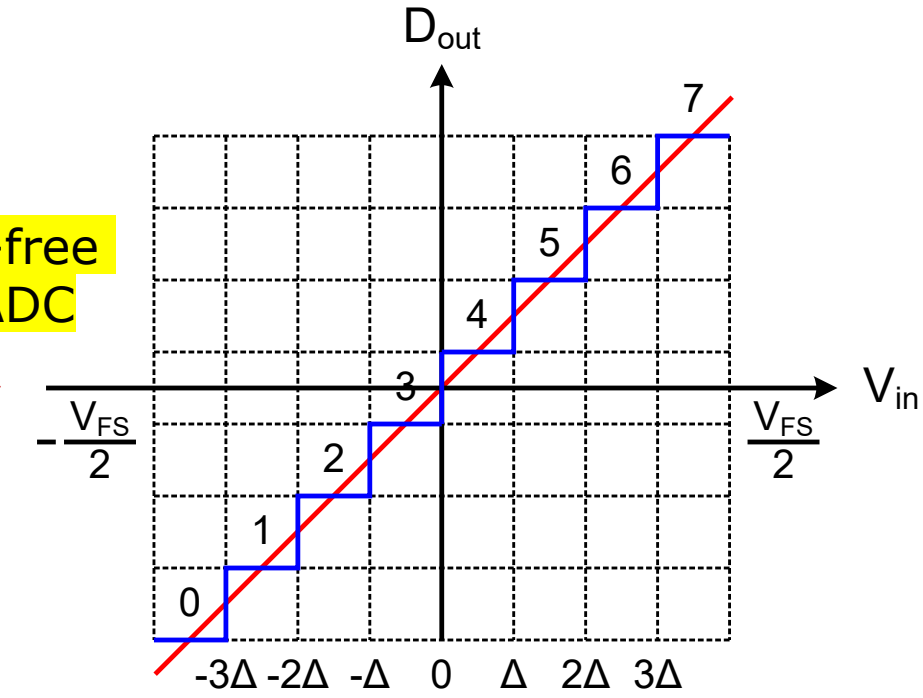


- ADC static errors also include gain, offset, nonlinearity (DNL and INL), but
- Watch out for missing code and nonmonotonicity!

INL and DNL (ADC)

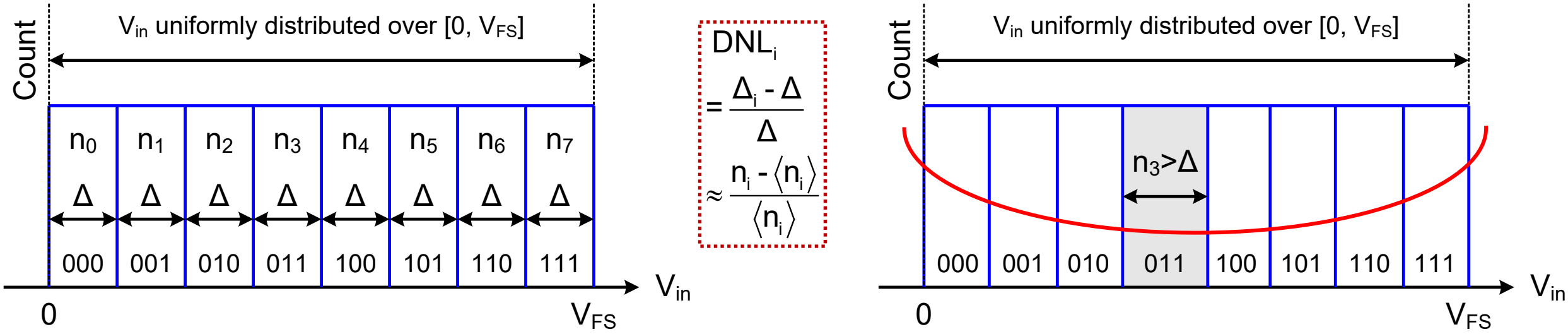


Offset-free diff. ADC



- Connect midpoints of treads to determine the INL profile (also "detrend")
- Direct TF measure requires a high-resolution DAC to produce/record the exact input analog values

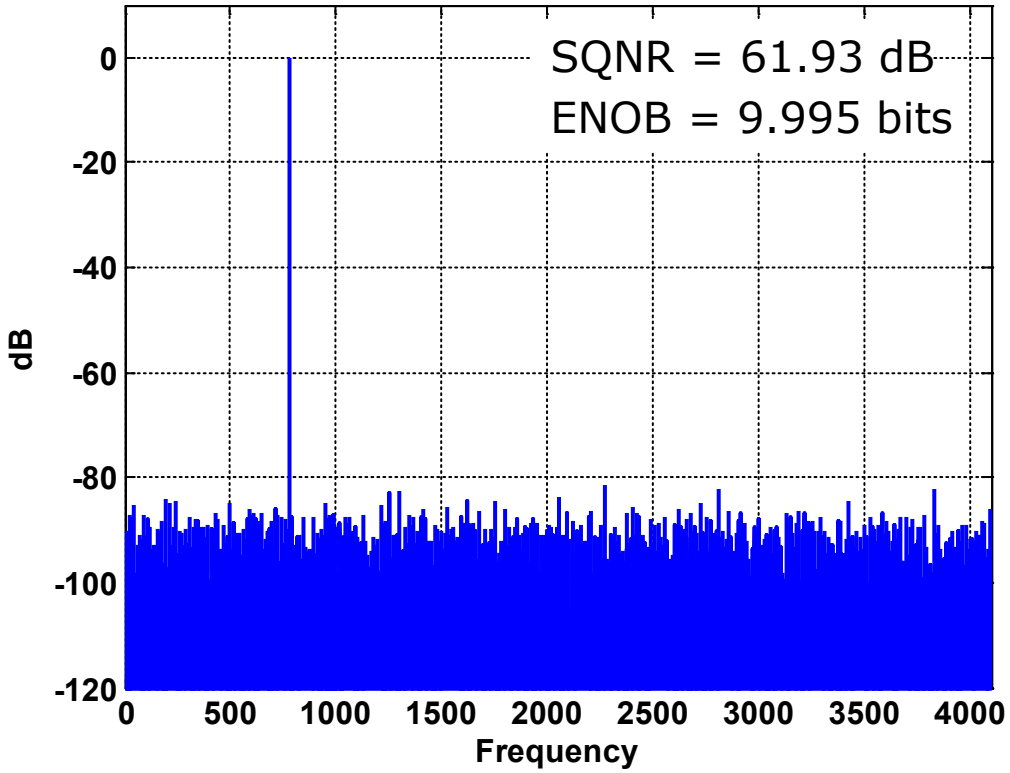
Code Density Test



- Apply a linear ramp to ADC input and collect the output code histogram
  - Ball-casting problem: # of balls collected proportional to bin size (DNL)
  - Missing codes → DNL = -1
  - Nonmonotonic codes → all lumped together, **very misleading!**
- Linear ramp is difficult to generate in practice → use filtered sinewave instead (but needs to correct the bowl-shaped density curve) Ref. [4]

FREQUENCY- & TIME-DOMAIN MEASUREMENTS

Spectrum of Quantized Signal

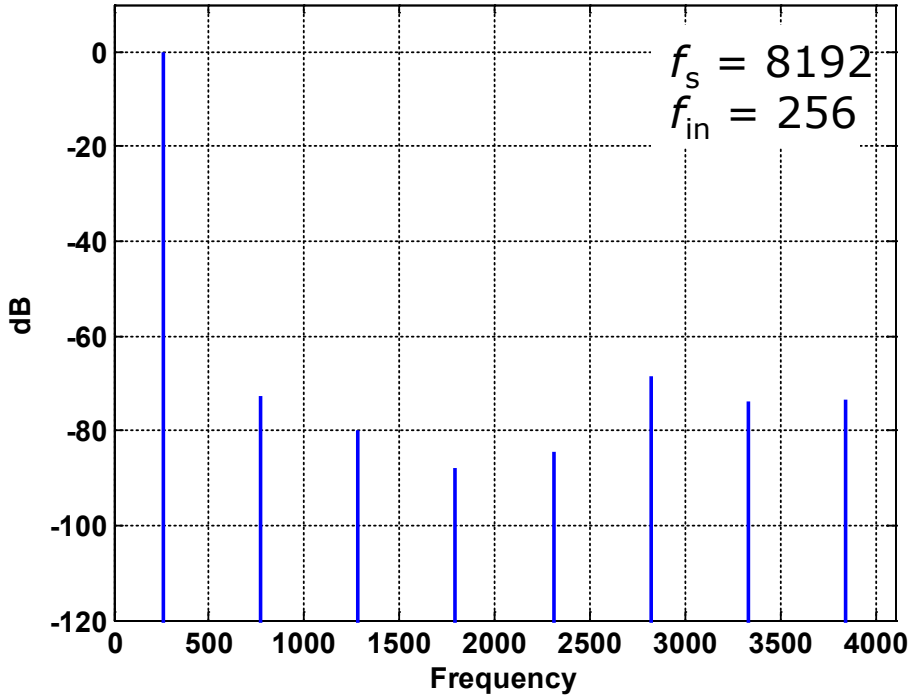


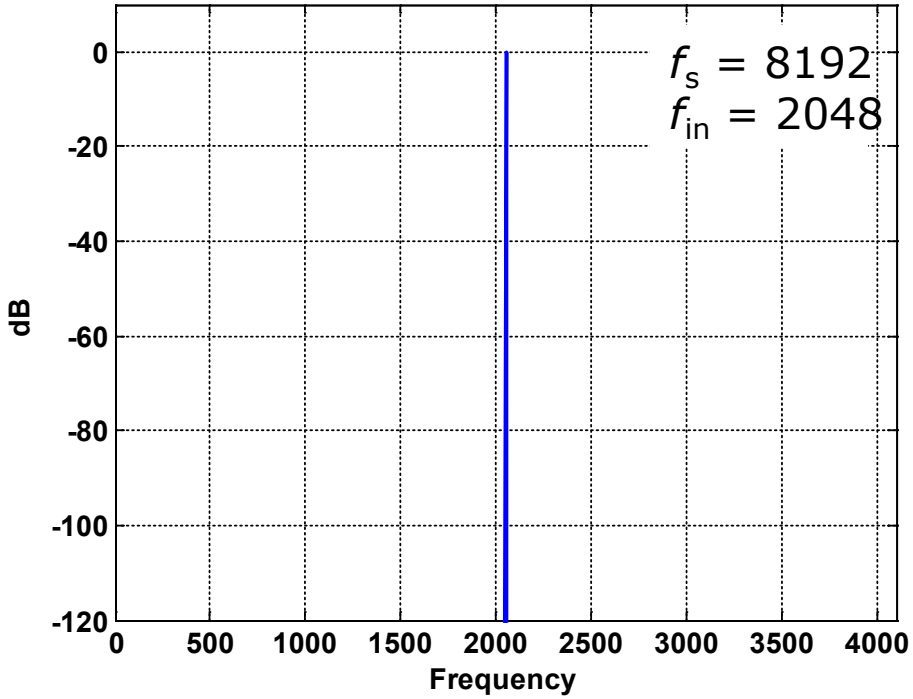
- N = 10 bits
- f_s = 8192, f_{in} = 779
- 8192 samples, only [0, f_s/2] shown
- Normalized to the amplitude of V_{in}
- Effective # of bits (sinusoidal):

$$ENOB = \frac{SQNR - 1.76\,dB}{6.02\,dB}$$

- FFT requires power-of-two number of samples to compute the spectrum
- f_{in} and f_s must be incommensurate, e.g., f_{in} and f_s are co-prime

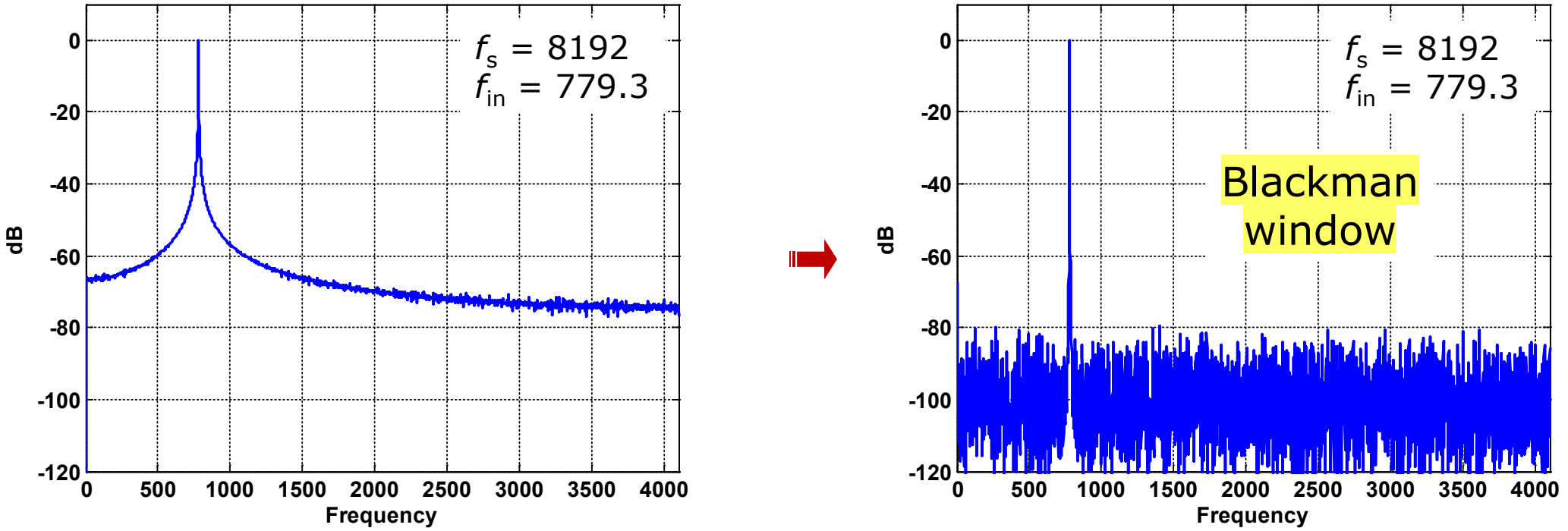
Commensurate f_s and f_{in}





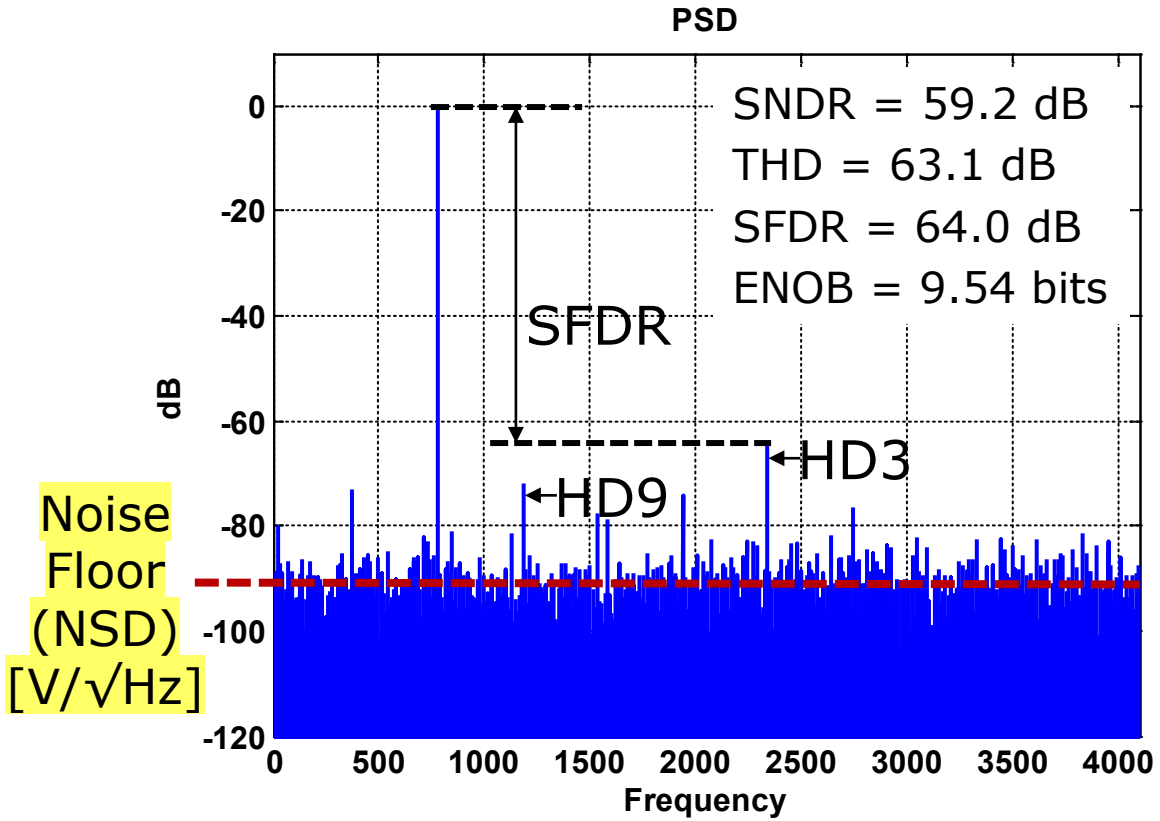
- Repeated/periodic samples lead to periodic quantization errors, manifested as harmonic distortions

Spectrum Leakage



- Samples must include integer # of cycles of input signal
- Windowing can be applied to eliminate spectrum leakage
  - Tradeoff b/t main-lobe width and sideband rejection of different windows

Frequency-Domain Performance Eval

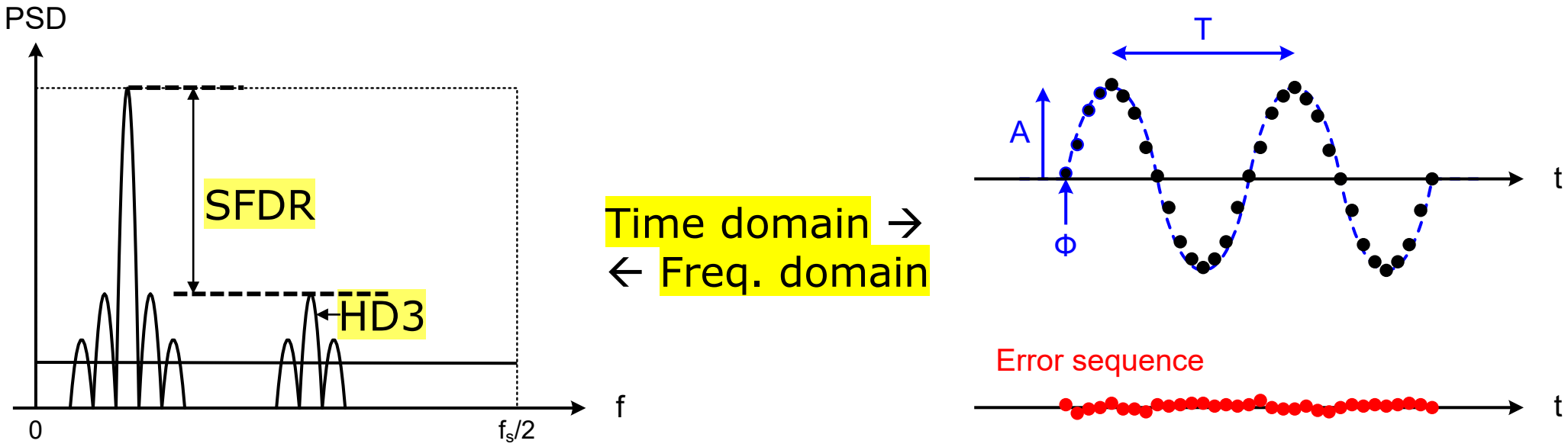


- Signal-to-noise plus distortion ratio (SNDR)
  - ALL bins except input are noise + disto
- Total harmonic distortion (THD)
  - Usually count first 10 HDs
- Spurious-free dynamic range (SFDR)
- Effective # of bits (sinusoidal):

$$\text{ENOB} = \frac{\text{SNDR} - 1.76\,\text{dB}}{6.02\,\text{dB}}$$

- High-order harmonics are aliased back, visible in band [0, $f_s/2$]
  - E.g., HD3 @ 779x3+1=2338, HD9 @ 8192-9x779+1=1182
- Two-tone/IM3 test is useful for ∑ΔM and DACs, but uncommon for Nyq. ADCs

Sine-Fit: A Time-Domain Method

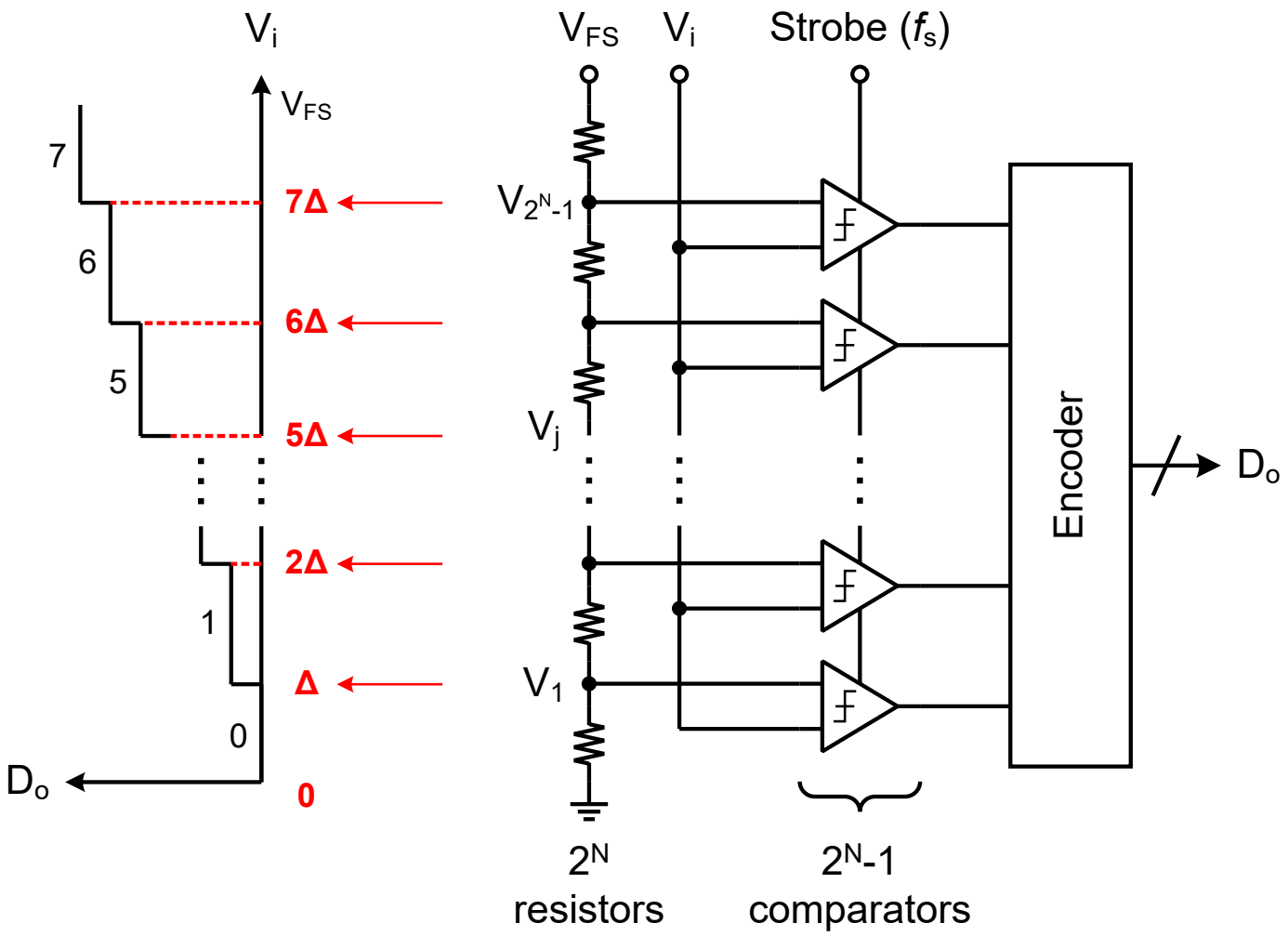


- Windowing, phase noise of signal/clock generator etc. can make frequency-domain SNDR measurement quite difficult
- If input is a sinewave, directly fit the measured (or simulated) samples to an ideal sinewave can yield accurate SNDR measurement (in MATLAB)
  - Nonlinear least-square fit by optimizing over (A, $\omega=2\pi/T$, Φ)

NYQUIST ADC ARCHITECTURES

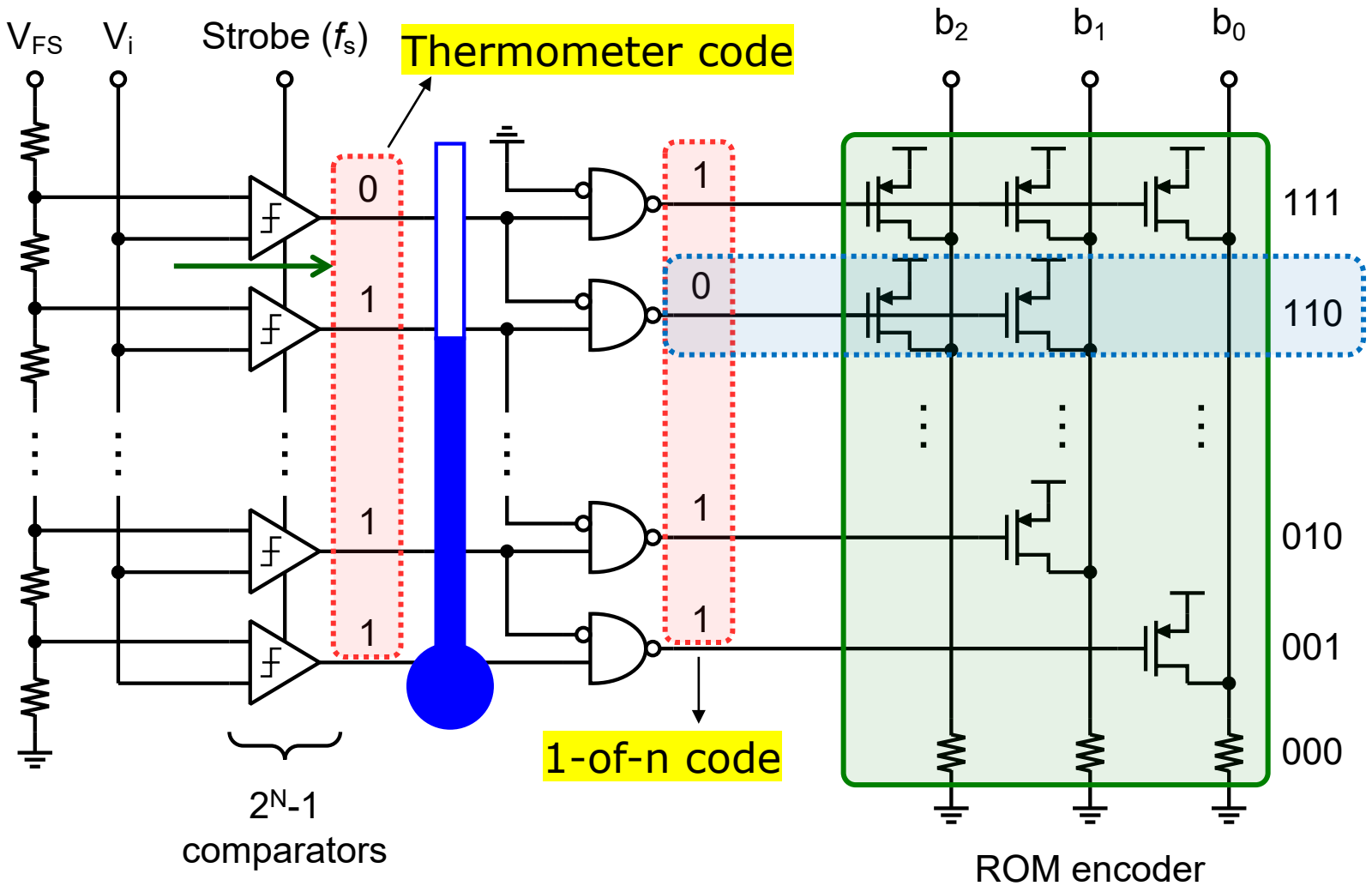
FLASH, SAR, AND PIPELINE

Flash ADC – Exhaustive Search



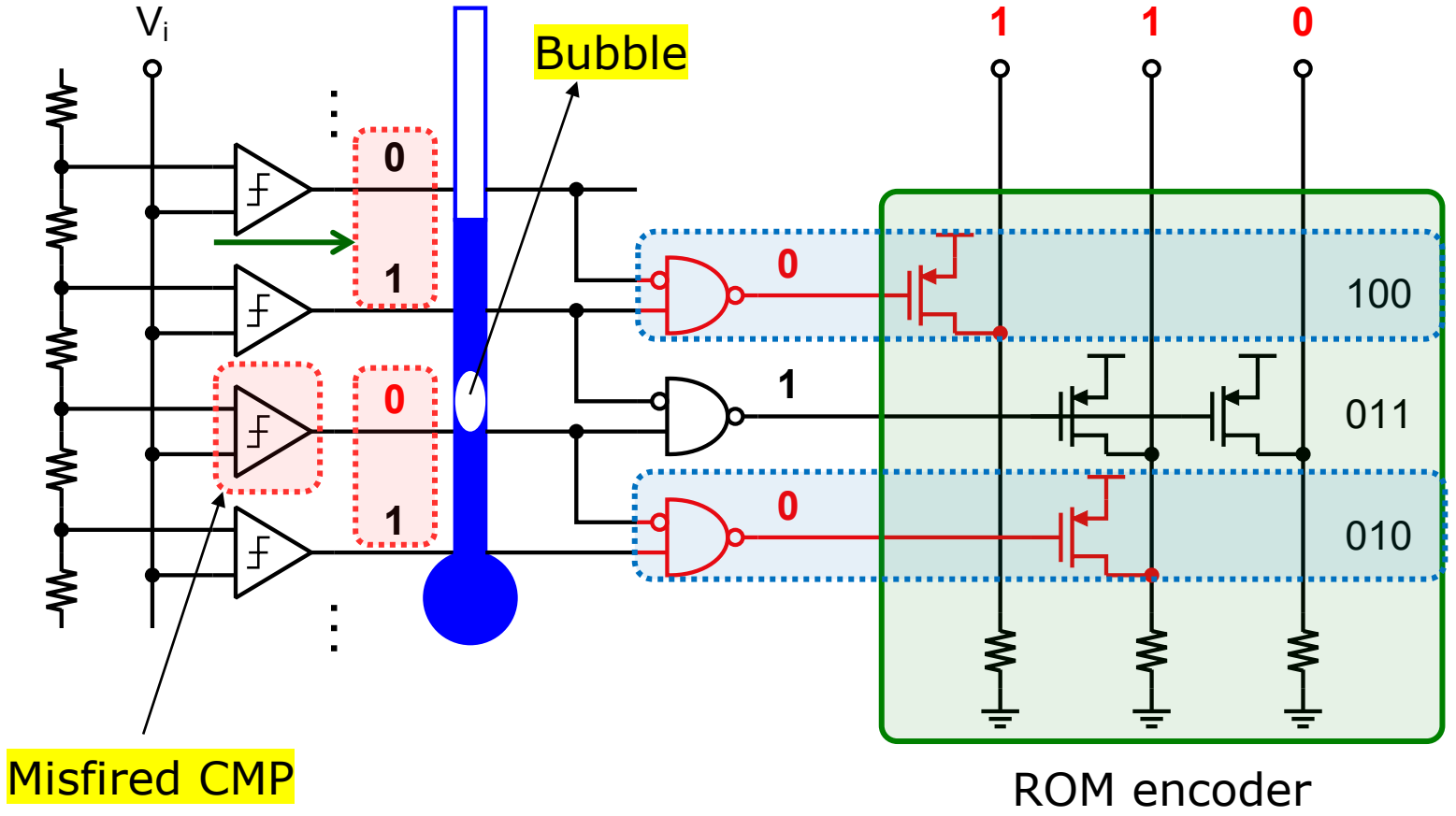
- Massive parallelism
- Very fast (N bits/step)
- Reference ladder: 2^N equal-sized resistors
- Input compared to 2^N-1 ref. taps (V_j)
- Throughput = f_s
- Complexity ~2^N
- Rarely used for more than 6-8 bits

Thermometer Code



- Thermometer code
- 1-of-n code
- T2B encoder

Bubble / Sparkle Errors



- 3-input boundary det.
- Majority voting
- Gray code

Quantization – Long Division

N = 3

FS = 1000

Δ = FS/8
= 125

V_{in} = 735

$$D_o = \left\lfloor \frac{V_{in}}{\Delta} \right\rfloor \qquad \underbrace{735}_{\text{Vin}} \div \underbrace{125}_{\text{LSB}} = \underbrace{[1,0,1]_b}_{D_o\,(\text{Quotient})} \; r \; \underbrace{110}_{\text{QN (Remainder)}}$$

Step 1: 1st bit

$$\begin{array}{r} 1 \\ 500\,)\,735 \\ 500 \\ \hline 235 \end{array}$$

Step 2: 2nd bit

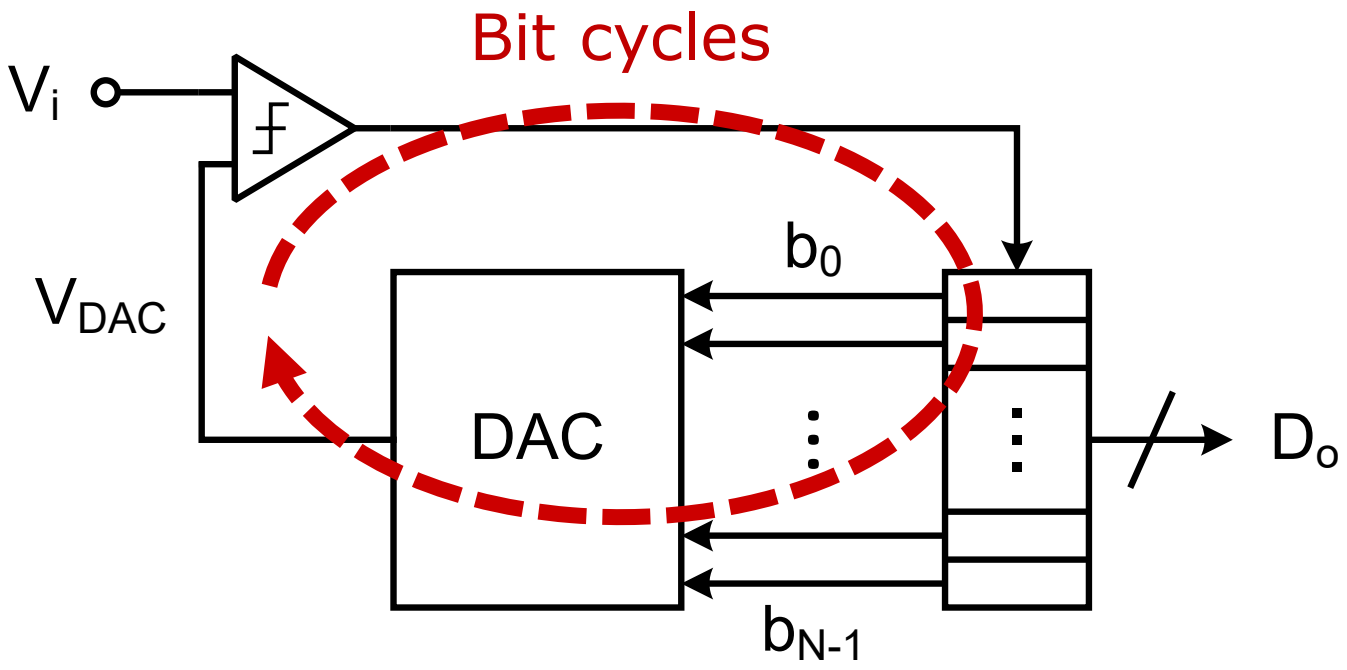
$$\begin{array}{r} 0 \\ 250\,)\,235 \\ 0 \\ \hline 235 \end{array}$$

Step 3: 3rd bit

$$\begin{array}{r} 1 \\ 125\,)\,235 \\ 125 \\ \hline 110 \end{array}$$

- The procedure is also known as **binary (radix=2) search**

Successive-Approximation (SAR) ADC

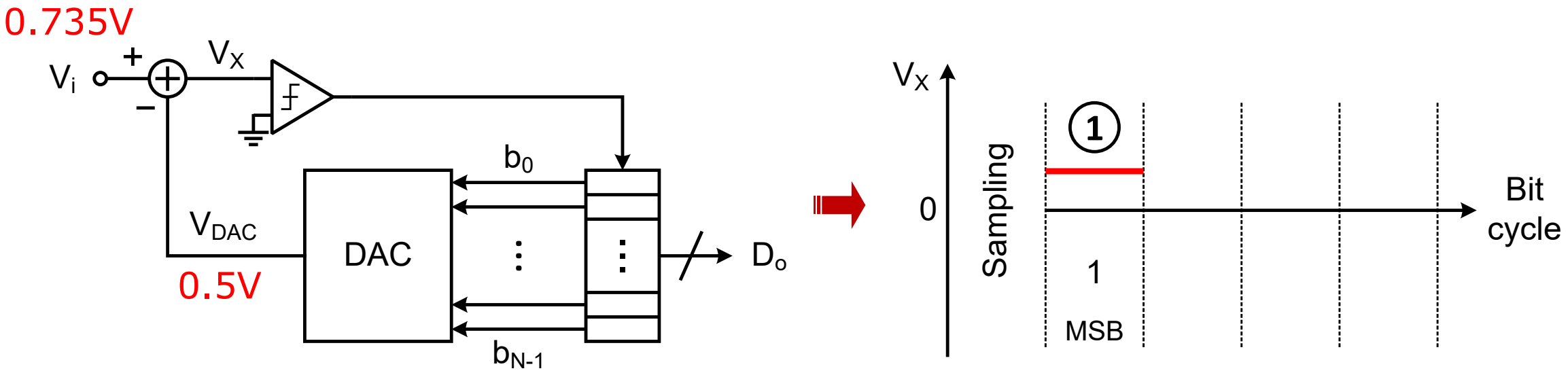


- 1 comparator
- 1 DAC
- Some digital logic

- Serial architecture (1 bit/step) → NOT built for high speed operation
- Hardware reuse / very efficient, minimal architecture

Binary Search: MSB Cycle

$N = 3$, FS = 1 V, $\Delta = 0.125$ V, $V_{in} = 0.735$ V

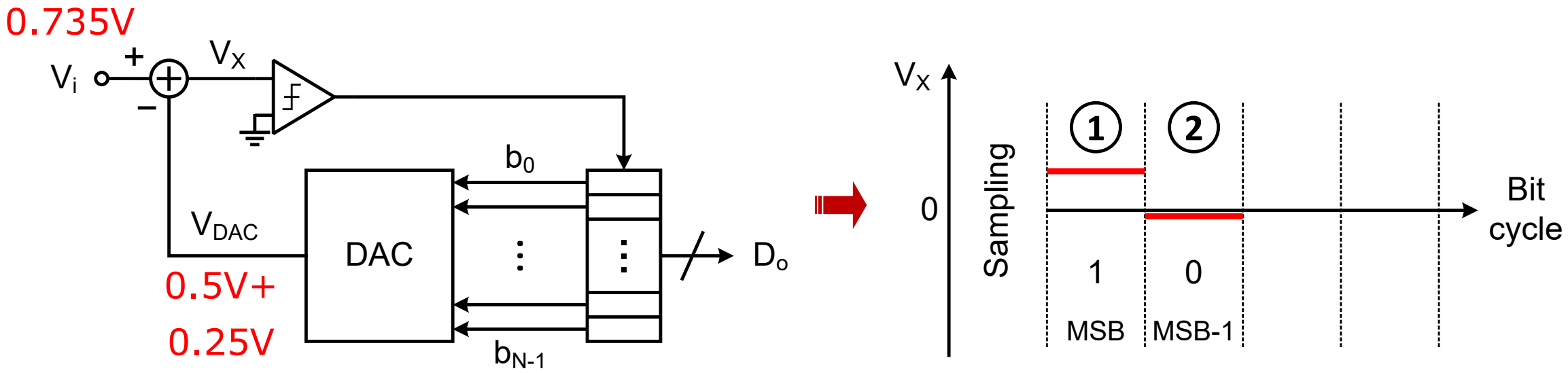


① $V_X = V_i - 0.5V$;

② if $V_X > 0$, MSB = 1, keep current $V_X \rightarrow V_X$;
otherwise, MSB = 0, restore $V_X \rightarrow V_X + 0.5V$;

Binary Search: MSB-1 Cycle

$N = 3$, FS = 1 V, $\Delta = 0.125$ V, $V_{in} = 0.735$ V

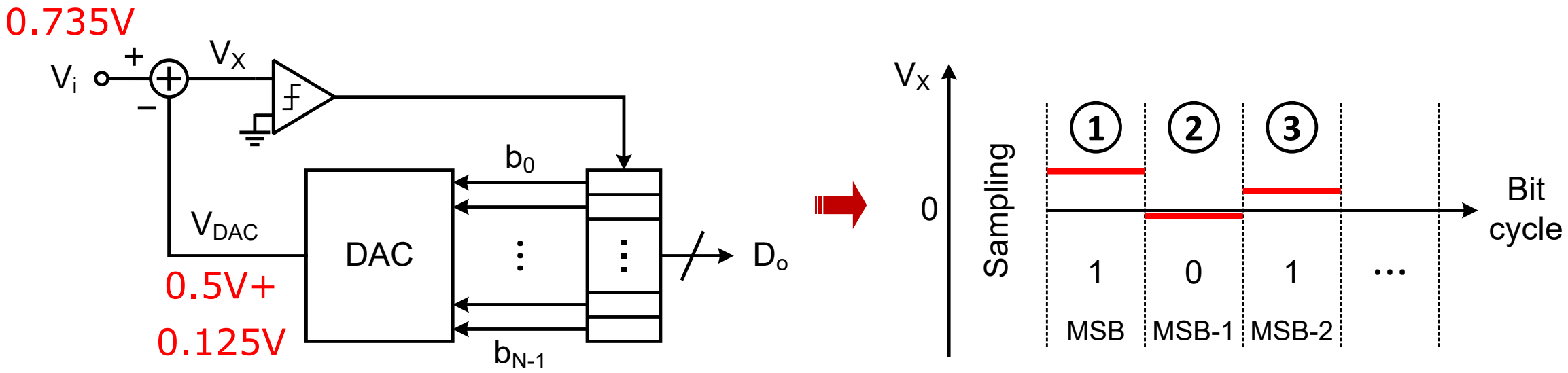


① $V_X = V_X - 0.25V$;

② if $V_X > 0$, MSB-1 = 1, keep current $V_X \rightarrow V_X$;
otherwise, MSB-1 = 0, restore $V_X \rightarrow V_X + 0.25V$;

Binary Search: MSB-2 Cycle

$N = 3$, FS = 1 V, $\Delta = 0.125$ V, $V_{in} = 0.735$ V



① $V_X = V_X - 0.125V$;

② if $V_X > 0$, MSB-2 = 1, keep current $V_X \rightarrow V_X$;
otherwise, MSB-2 = 0, restore $V_X \rightarrow V_X + 0.125V$;

Modified Binary Search

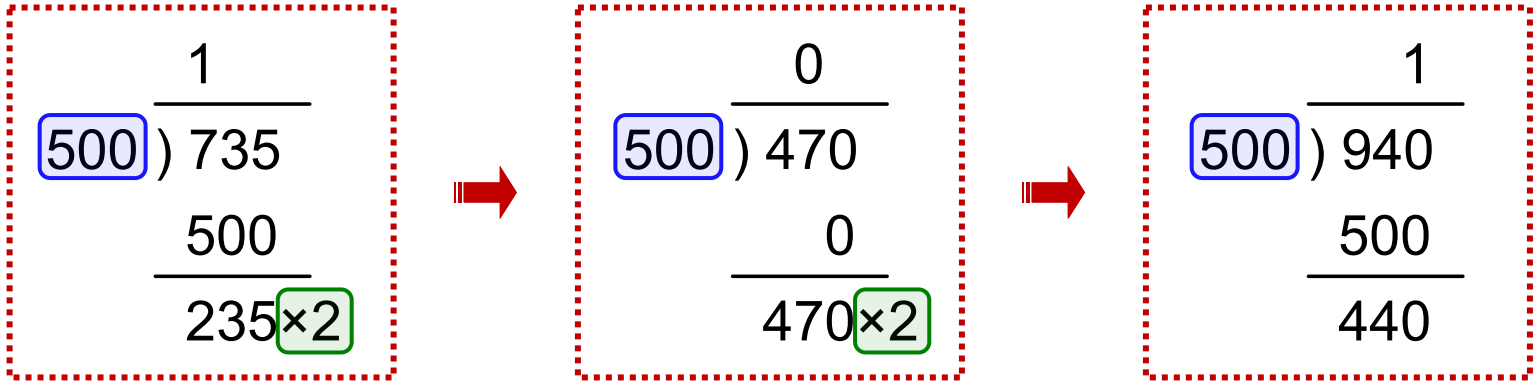
N = 3

FS = 1000

Δ = FS/8
= 125

V_{in} = 735

$$D_o = \left\lfloor \frac{V_{in}}{\Delta} \right\rfloor \qquad \underbrace{735}_{\text{Vin}} \div \underbrace{125}_{\text{LSB}} = \underbrace{[1,0,1]_b}_{D_o\,(\text{Quotient})} \; r \; \underbrace{440/4}_{\text{QN (Remainder)}}$$



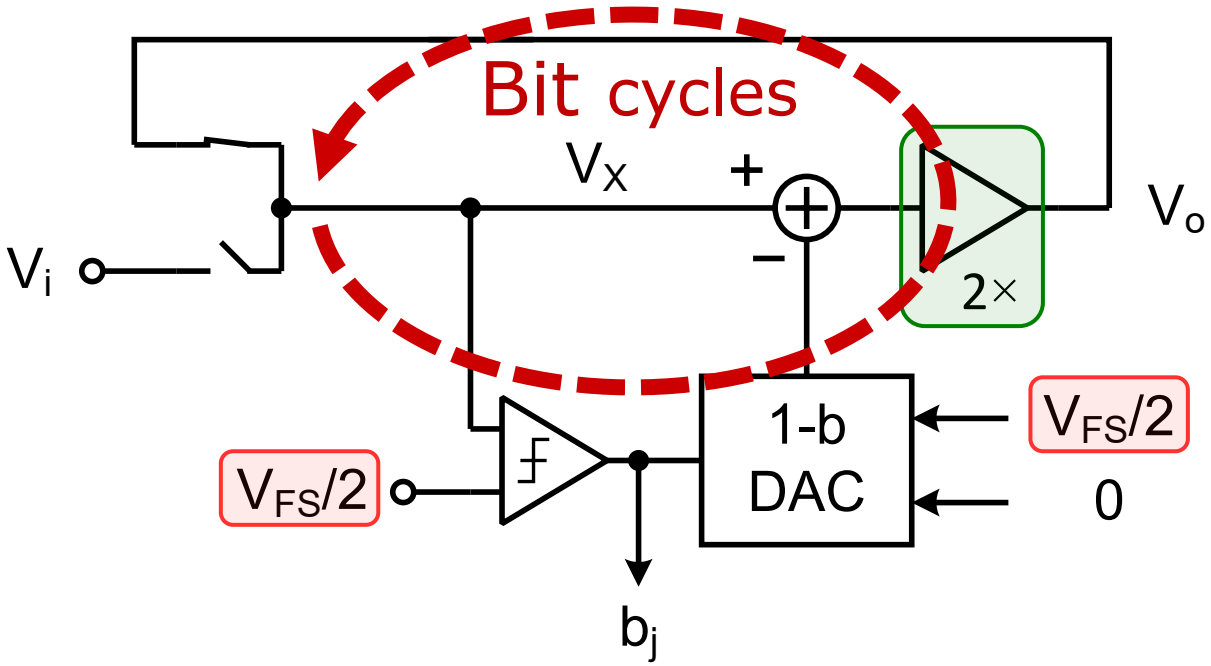
Step 1:
1st bit

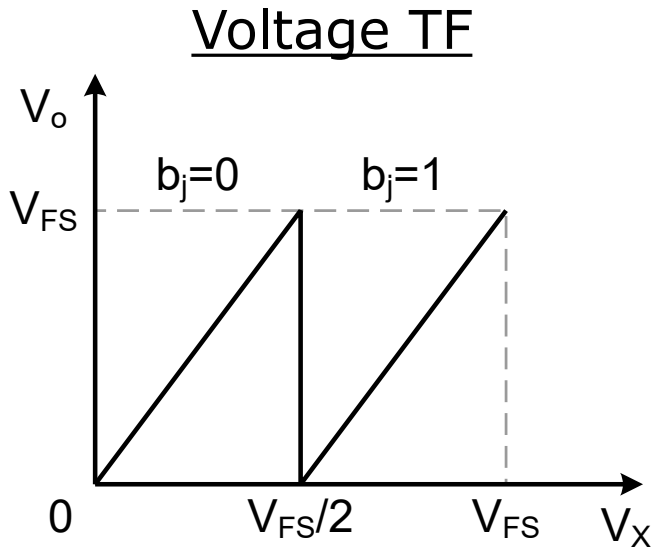
Step 2:
2nd bit

Step 3:
3rd bit

- Always use the same divisor but **amplify** the remainder / residue

Algorithmic (Cyclic) ADC

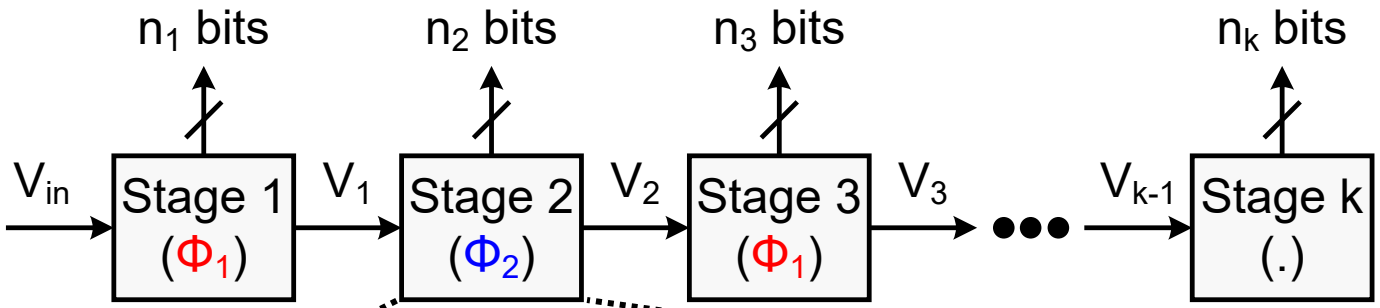


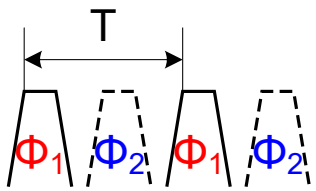


$$V_o = 2 \cdot \left(V_X - b_j \cdot \frac{V_{FS}}{2} \right)$$

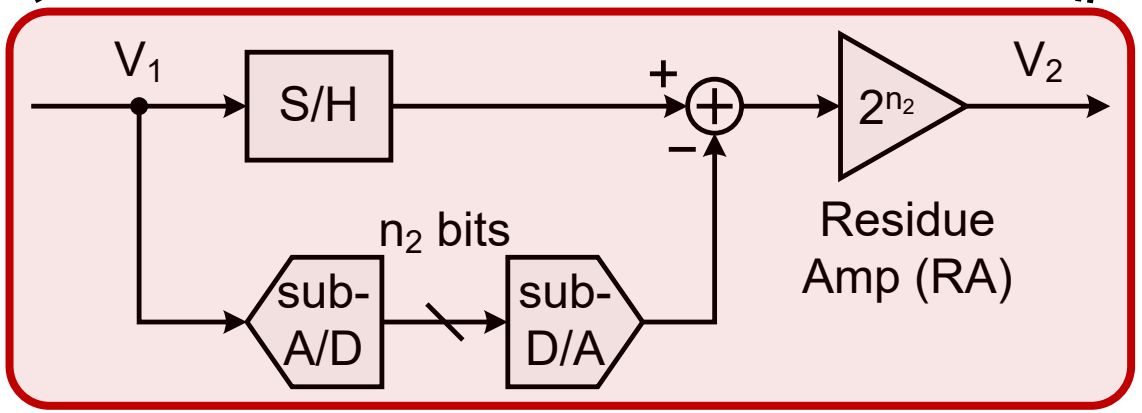
- Fixed comparison threshold ($V_{FS}/2$) + 1b DAC + 2X Residue Gain
  - Comparison → if $V_X < V_{FS}/2$, then $b_j = 0$; otherwise, $b_j = 1$
  - Residue generation → $V_o = 2 \cdot (V_X - b_j \cdot V_{FS}/2)$

Pipelined ADC





two-phase nonoverlapping clock

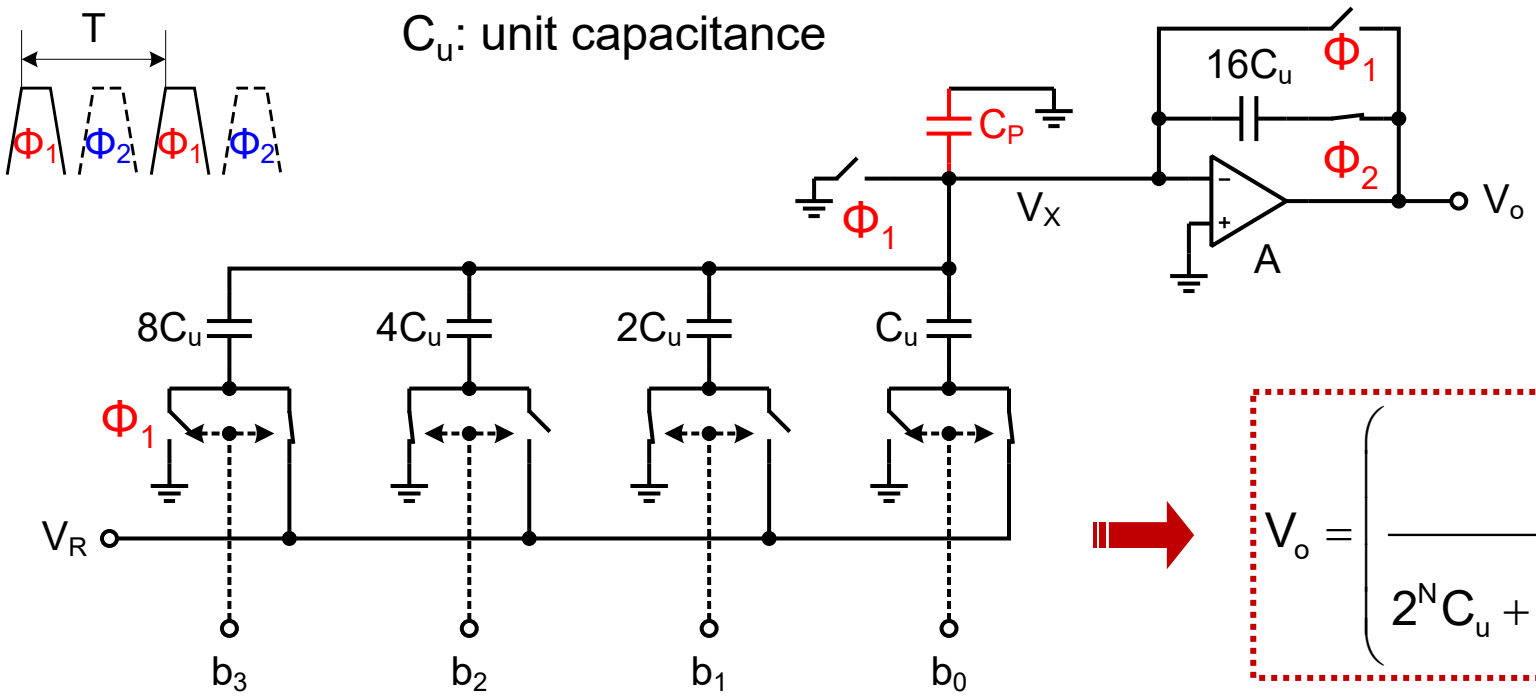


- Concurrent stage operation
- Latency ~ k/2
- Throughput ~ f_s
- Complexity ~ N
- Scalable to M-bit/stage (1≤M<N)
- Power saving possible by stage tapering along pipeline

- Algorithmic loop unrolled → pipeline enables high conversion throughput

NYQUIST DAC ARCHITECTURES
BW, UE, AND SEGMENTED

Binary-Weighted (BW) DAC



$$V_o \approx \sum_{i=0}^{N-1} b_i \cdot \left(2^i \frac{V_R}{2^N}\right)$$

Multiplier

$$V_o = \left(\frac{2^N C_u}{2^N C_u + \frac{C_p + \left(2^{N+1} - 1\right) C_u}{A}}\right) \cdot \sum_{i=0}^{N-1} b_i \cdot \left(2^i \frac{V_R}{2^N}\right)$$

- Charge redistribution (CR) of a BW capacitor array → very efficient architecture
- N-bit DAC requires N switching elements, w/ direct binary bits passthrough control
- Can be generalized to resistive, R2R and current-steering topologies

Midscale DNL / INL (MSB Transition)

Code 0111

$$V_o(0111) = \left(\frac{C_1 + C_2 + C_3}{16C_u}\right) \cdot V_R$$

Code 1000

$$V_o(1000) = \left(\frac{C_4}{16C_u}\right) \cdot V_R$$

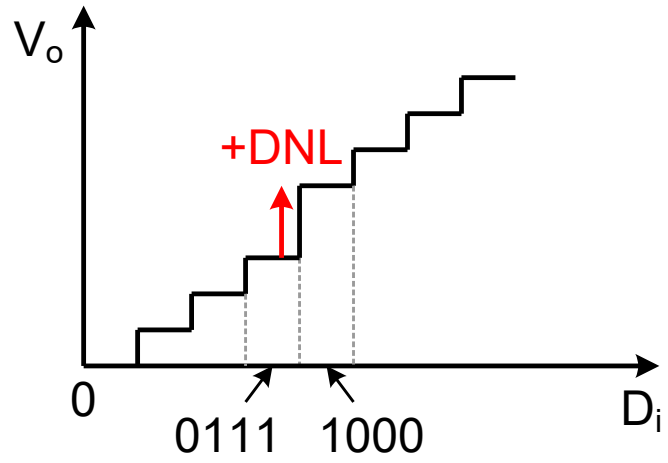
half array / other half

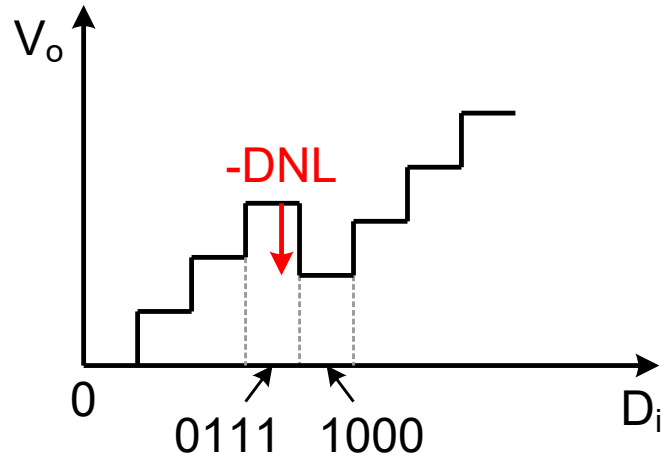
$$\text{Assume: } C_4 - (C_1 + C_2 + C_3) = C_u + \delta C$$

$$\text{DNL} = \frac{V_o(1000) - V_o(0111) - 1\ \text{LSB}}{1\ \text{LSB}}$$

$$= \frac{\delta C}{C_u}$$

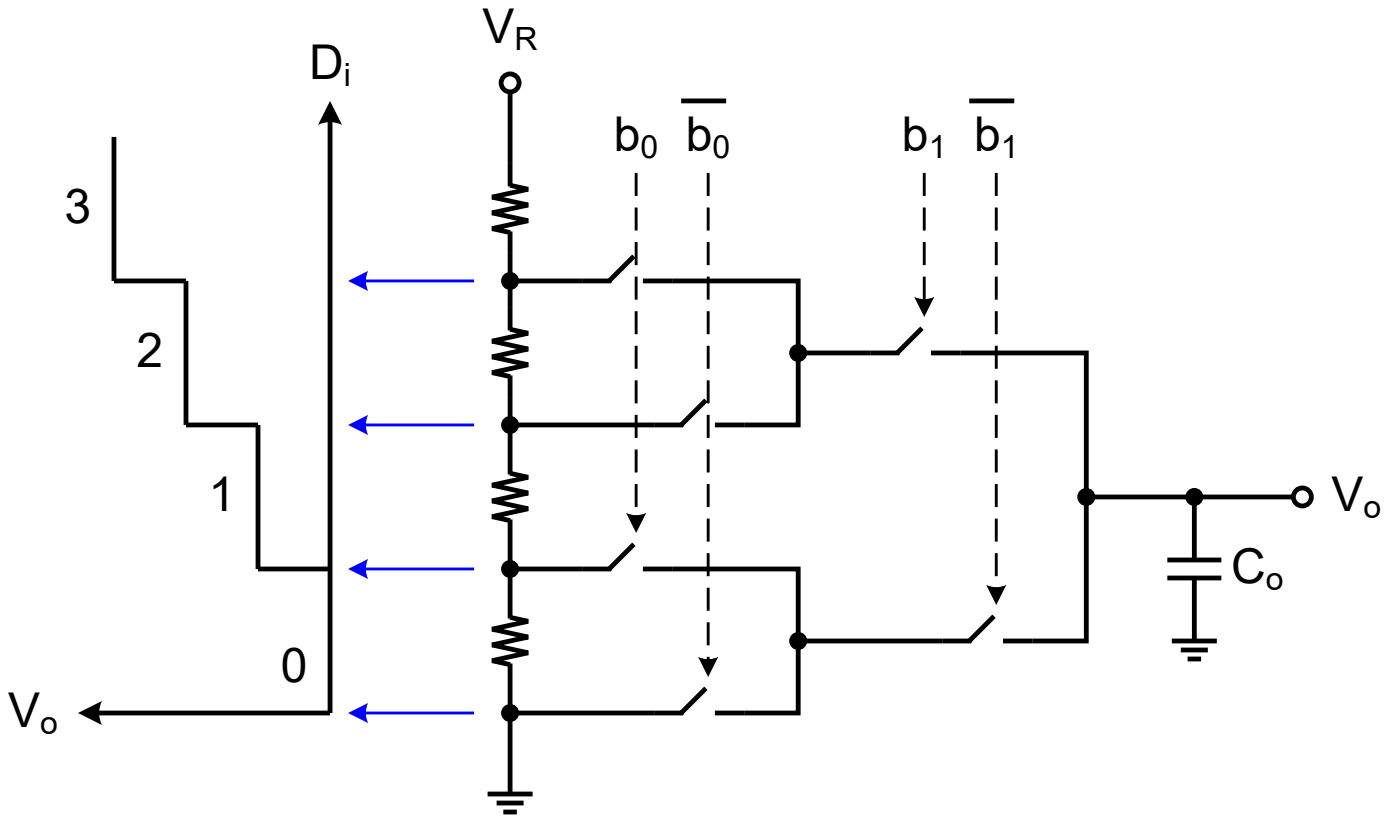
→ often the largest DNL error





- Architectural efficiency comes at a cost of linearity
- Original CR SAR ADC employed this type of DAC

Unit-Element (UE) DAC



UE DAC:

$$\sigma_{DNL} = \frac{\sigma_R}{R}$$

$$\sigma_{INL}(max) \approx \frac{\sqrt{2^N}}{2}\left(\frac{\sigma_R}{R}\right)$$

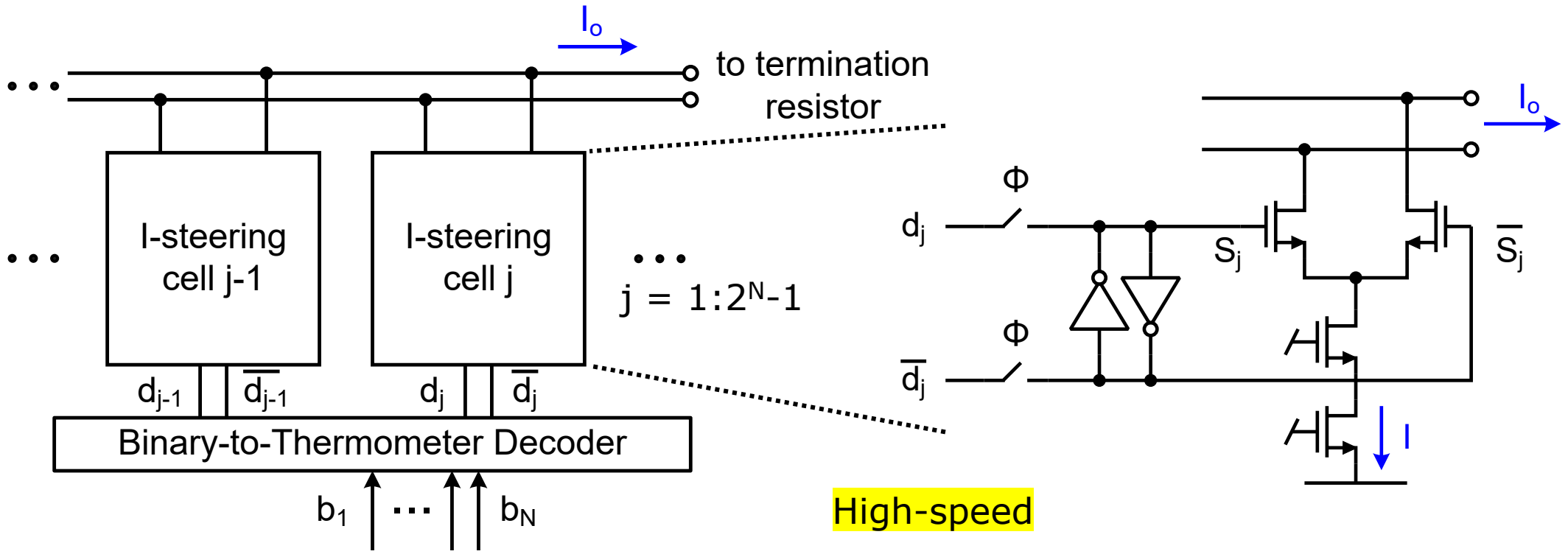
BW DAC:

$$\sigma_{DNL}(max) \approx \sqrt{2^N}\left(\frac{\sigma_R}{R}\right)$$

$$\sigma_{INL}(max) \approx \frac{\sqrt{2^N}}{2}\left(\frac{\sigma_R}{R}\right)$$

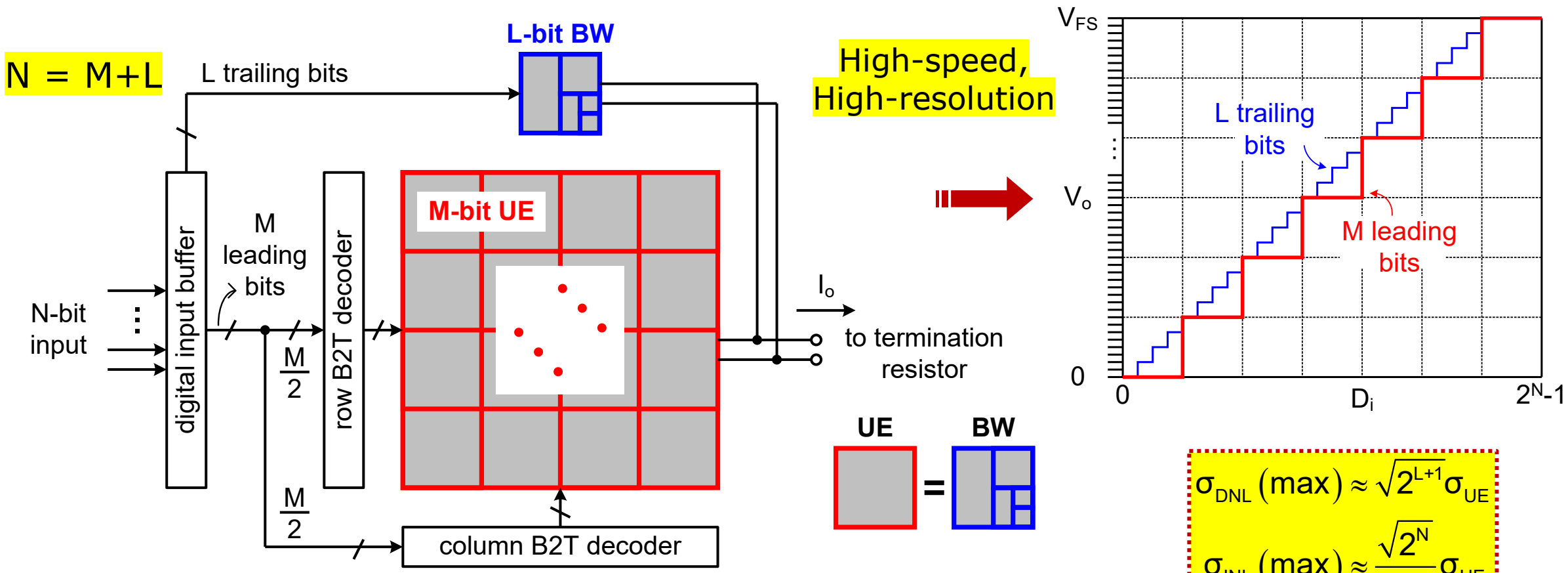
- Inherently monotonic → good DNL performance (what about INL?)
- ~2^N switching elements → complexity ↑ speed ↓ for large N → typically N≤8 bits
- Needs binary-thermometer decoder

Current-Steering DAC (UE)



- Fast switching, inherently monotonic → good DNL performance
- ~2^N current cells decomposed into a ($2^{N/2}$×$2^{N/2}$) matrix of rows and columns

Segmented (SM) DAC



$$\sigma_{DNL}(max) \approx \sqrt{2^{L+1}}\sigma_{UE}$$

$$\sigma_{INL}(max) \approx \frac{\sqrt{2^{N}}}{2}\sigma_{UE}$$

- A divide-and-conquer approach to tackle complexity of UE DAC
- # of switching elements ~$2^{M}+L$ << 2^{M+L}

Comparison of DAC Architectures

BW DAC:

$$\sigma_{DNL}(max) \approx \sqrt{2^N}\sigma_{UE}$$

$$\sigma_{INL}(max) \approx \frac{\sqrt{2^N}}{2}\sigma_{UE}$$

UE DAC:

$$\sigma_{DNL} = \sigma_{UE}$$

$$\sigma_{INL}(max) \approx \frac{\sqrt{2^N}}{2}\sigma_{UE}$$

SM DAC:

$$\sigma_{DNL}(max) \approx \sqrt{2^{L+1}}\sigma_{UE}$$

$$\sigma_{INL}(max) \approx \frac{\sqrt{2^N}}{2}\sigma_{UE}$$

Example: N = 12, M = 8, L= 4, σ_{UE} = 1%

Architecture	σ_{INL} [LSB]	σ_{DNL} [LSB]	# of switching elements
BW	0.32	0.64	12 (=N)
UE	0.32	0.01	4095 ($\approx 2^N$)
SM	0.32	0.057	259 ($\approx 2^M$+L)

DATA CONVERTER CIRCUIT BUILDING BLOCKS

Common Circuit Building Blocks

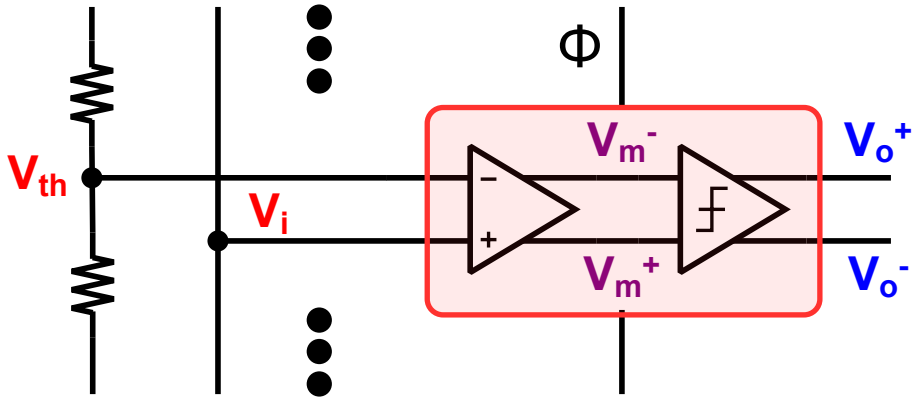
- DAC: resistive, capacitive, current types, unit-element, binary-weighted, segmented architectures – every ADC has a DAC built-in!
- Comparator: decision device, regeneration speed, metastability
- Sample and Hold (S/H) or Track and Hold (T/H): CT-DT conversion

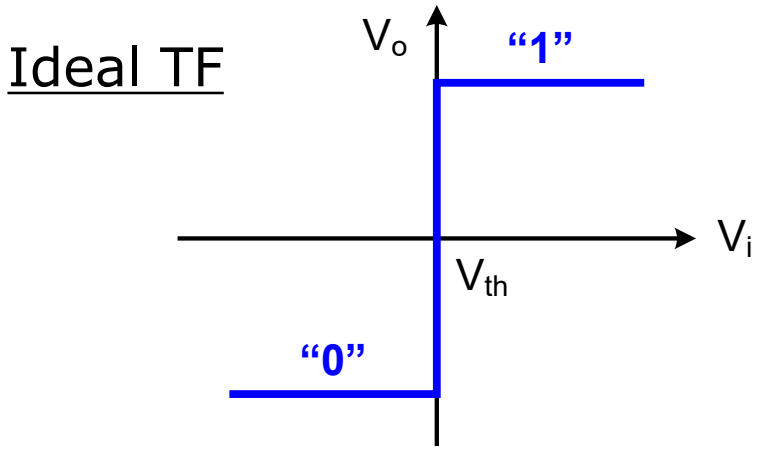
} Covered in this section

- Residue Amplifier (RA): closed-loop vs. open-loop, static vs. dynamic amplifiers
- Reference and Biasing: BGR, current mirror, on-chip / off-chip bypass
- Clocking Circuits: jitter critical for high-speed ADC and DAC (power consumption)
- Digital Logics: combinatory, sequential, memory, I/O, etc.
- Input and Reference Buffers: peripheral, critical, and power hungry (but often not included in the converter core)

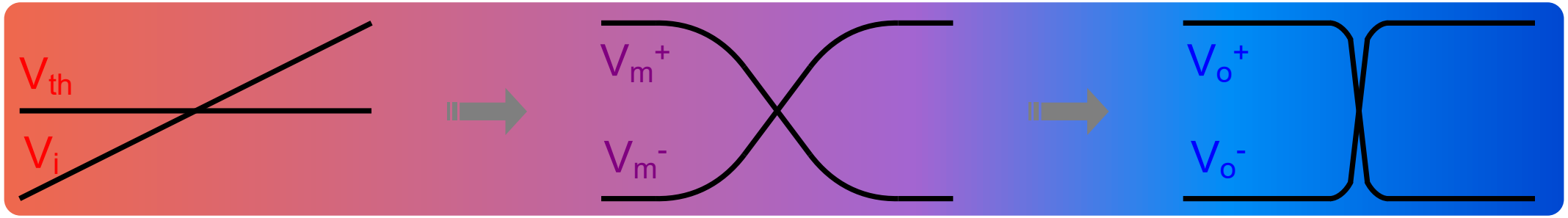
COMPARATOR

Comparator: Decision-Making Device



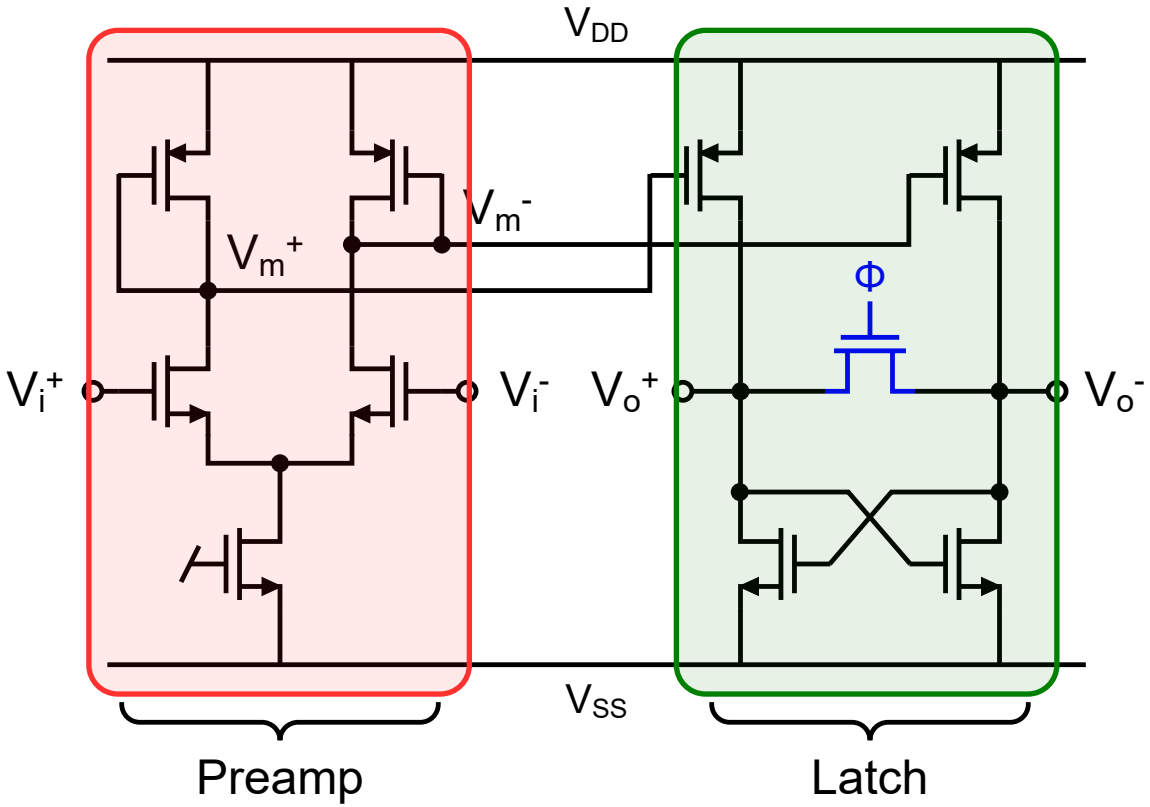


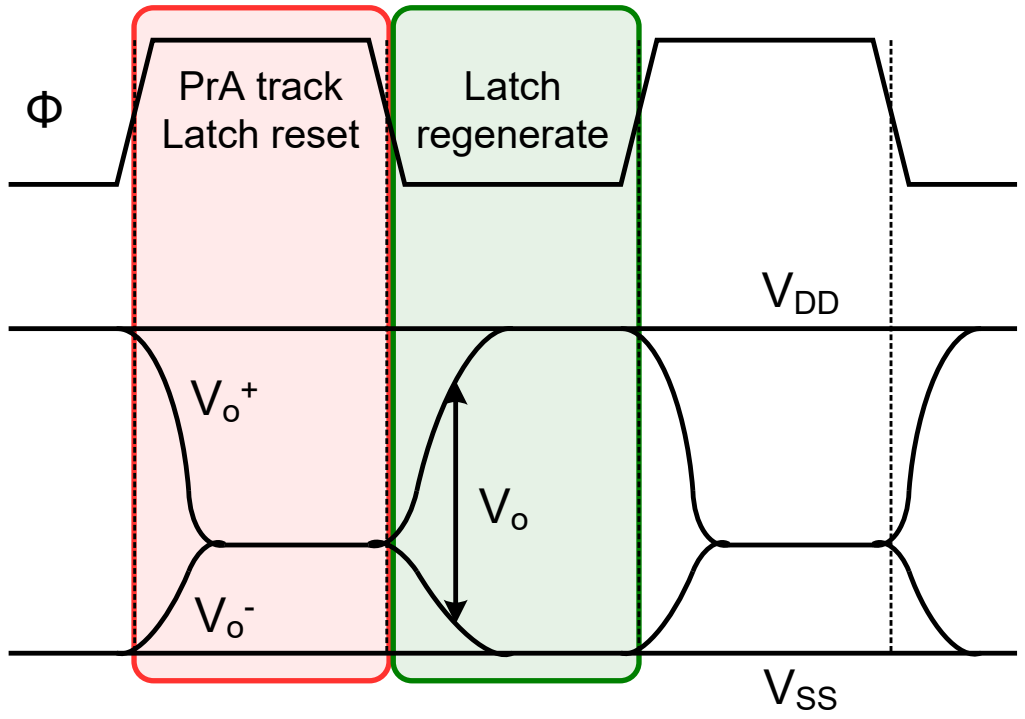
→ From soft to hard decision →



- Precise gain and linearity unnecessary → simple, open-loop, low-gain, and wideband Preamplifier (PrA) followed by Latch (positive feedback)

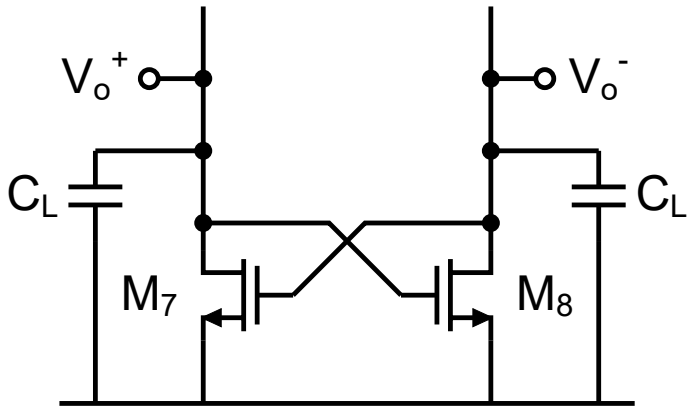
A Simple Comparator

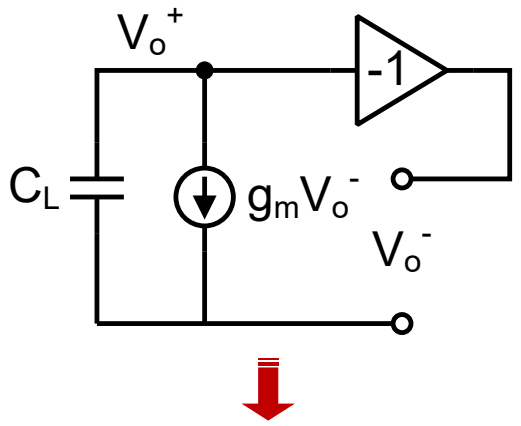




- Preamp establishes a seed voltage for latch regeneration → dominates offset
- Usually no return after regeneration starts

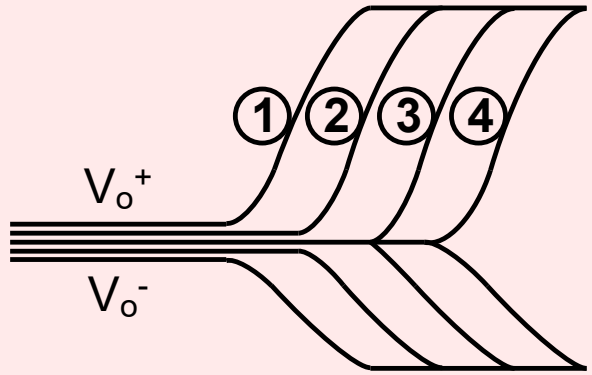
Latch Regeneration





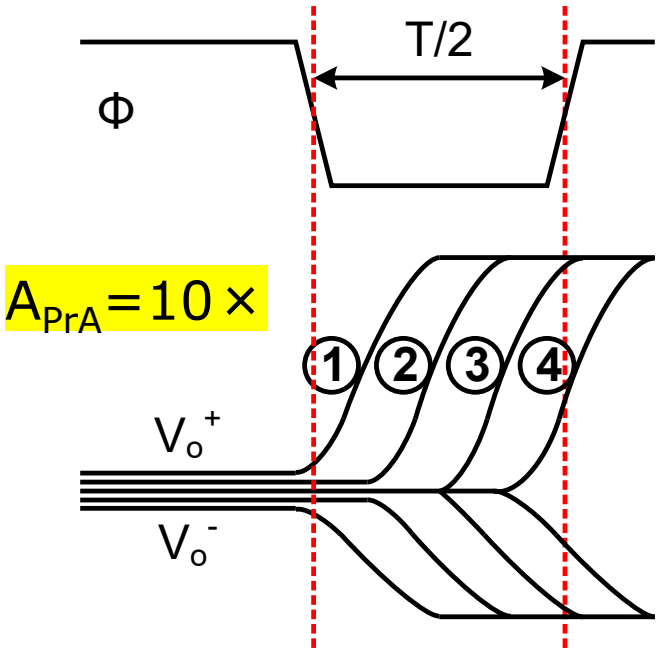
$$\begin{cases} V_o^+ = -V_o^- \\ V_o^+ = -g_m \cdot V_o^- / sC_L \end{cases} \quad \text{or} \quad \begin{pmatrix} 1 & 1 \\ 1 & g_m / sC_L \end{pmatrix} \begin{pmatrix} V_o^+ \\ V_o^- \end{pmatrix} = 0 \Rightarrow \begin{matrix} s_p = g_m / C_L \\ (\text{RHP pole}) \end{matrix}$$

$$V_o(t) = V_o(t=0) \cdot \exp\left(t \cdot \frac{g_m}{C_L}\right) \quad \text{or} \quad t = \frac{C_L}{g_m} \cdot \ln\left(\frac{V_o(t)}{V_o(t=0)}\right)$$



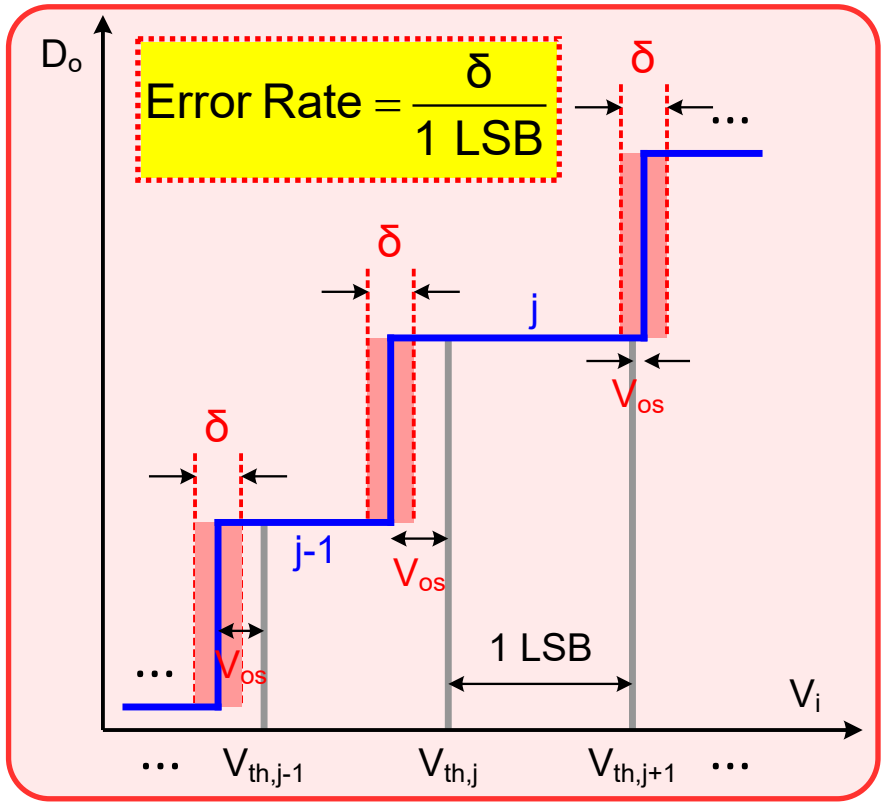
Case	$V_o(t=0)$	V_o	$t/(C_L/g_m)$
①	100mV	1V	2.3
②	10mV	1V	4.6
③	1mV	1V	6.9
④	100μV	1V	9.2

Metastability



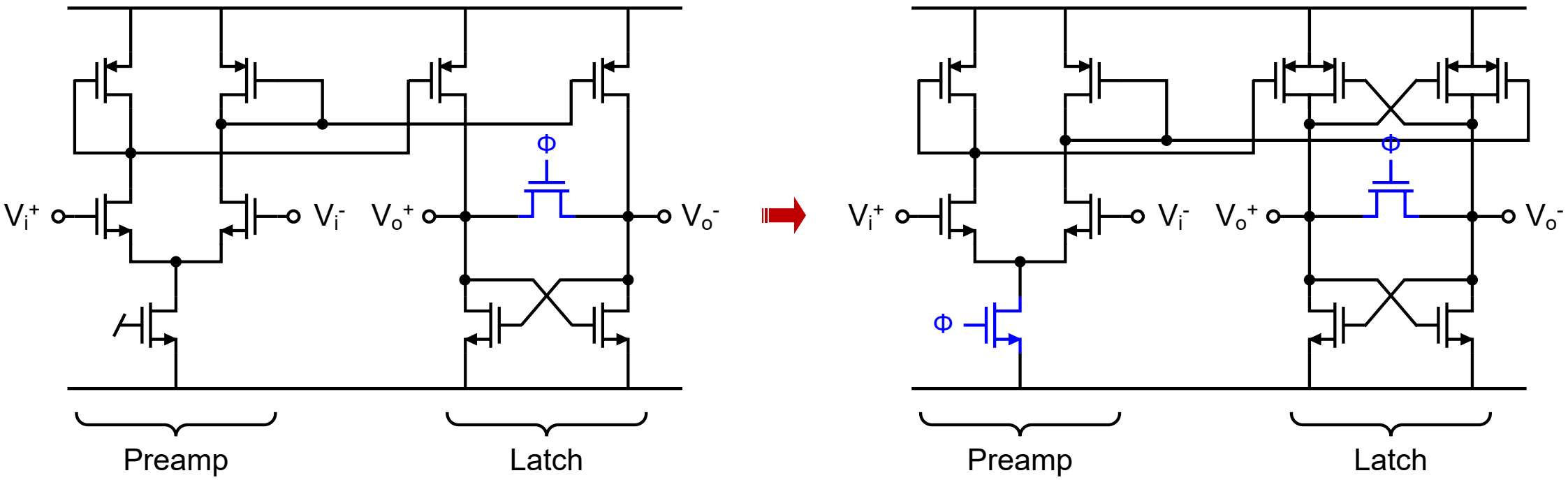
$$V_o(t) = A_{PrA} \cdot V_i(0) \cdot \left(t \cdot \frac{g_m}{C_L} \right)$$

Curve	$V_i(t=0)$	$t/(C_L/g_m)$
①	10mV	2.3
②	1mV	4.6
③	100μV	6.9
④	10μV	9.2



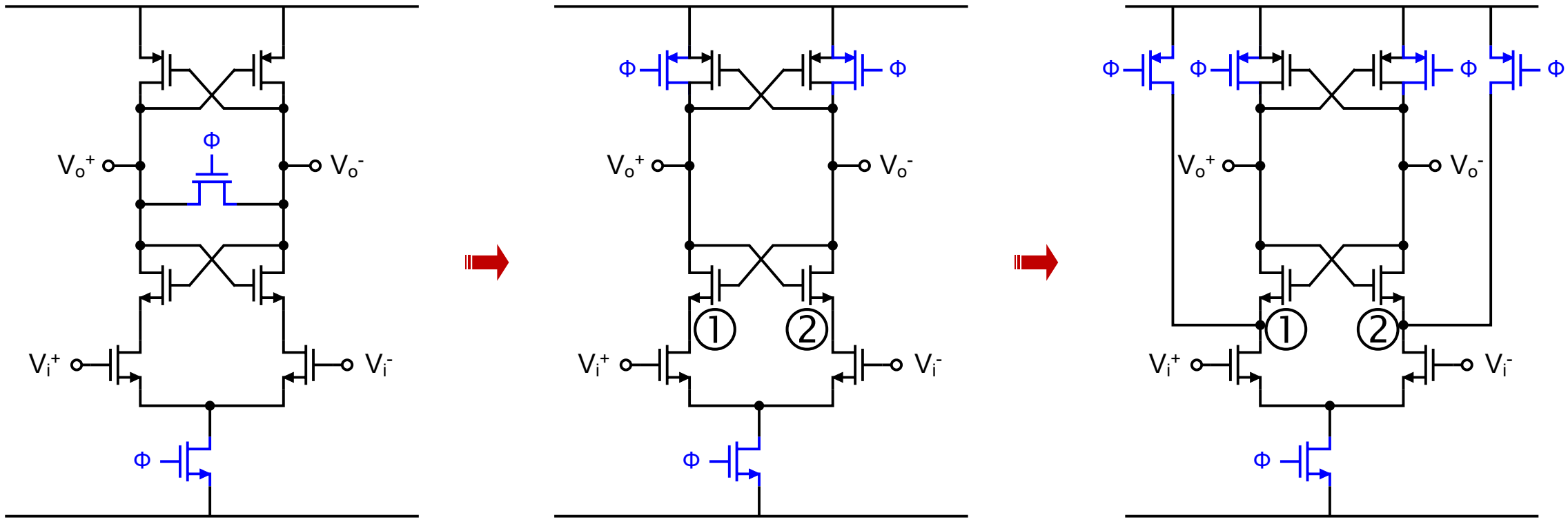
- Comparator fails to produce valid logic output for tiny inputs → metastability
- No decision is often worse than a wrong decision! (conversion may halt!)

Dynamic Comparator



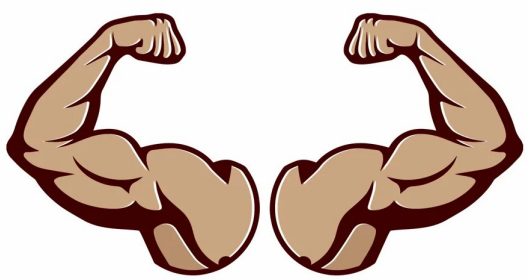
- Starting with the static comparator (i.e., preamp + latch)
- First, add a PMOS cross-coupled pair in the latch to fully regenerate logic levels
- Second, turn the static preamp into a dynamic one

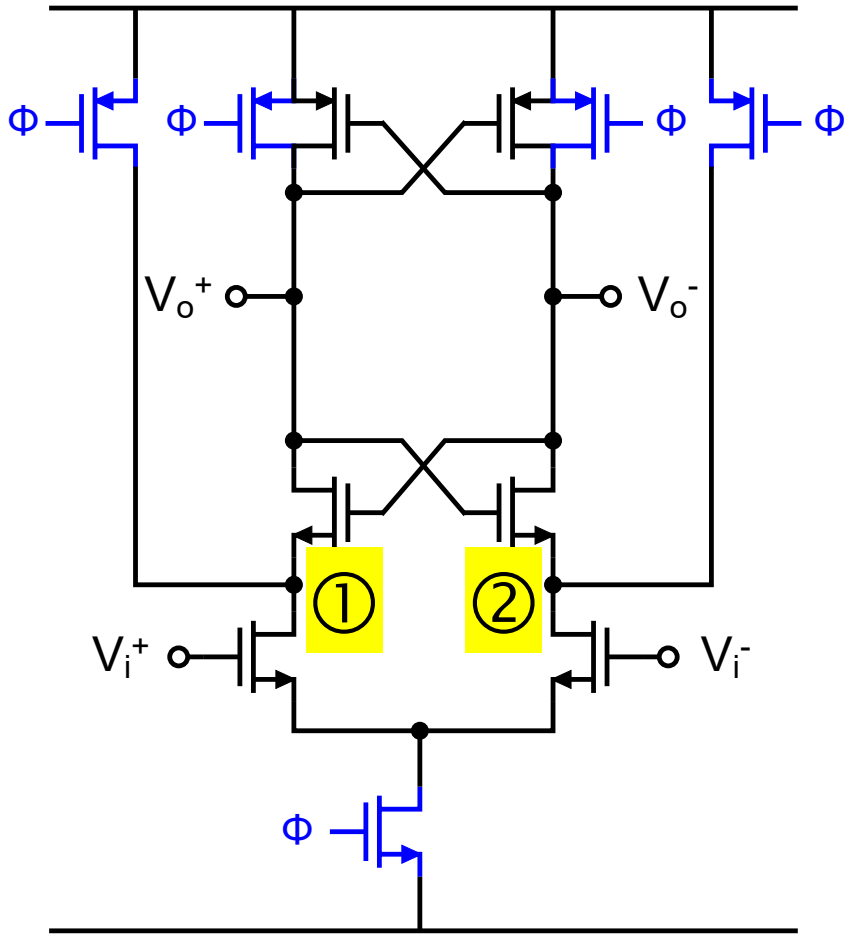
Dynamic Comparator



- Third, stack the two together for current reuse (to save power)
- Then, replace the crowbar by conspicuous PMOS reset (high) switches
- Lastly, completely reset ALL internal nodes to remove memory

Strong-Arm Dynamic Comparator

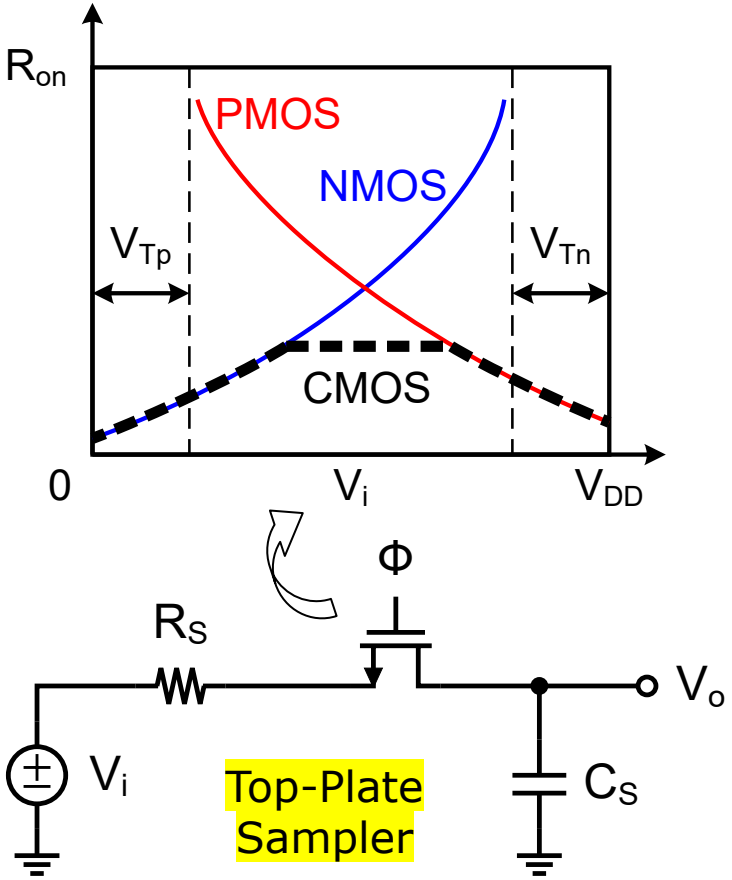




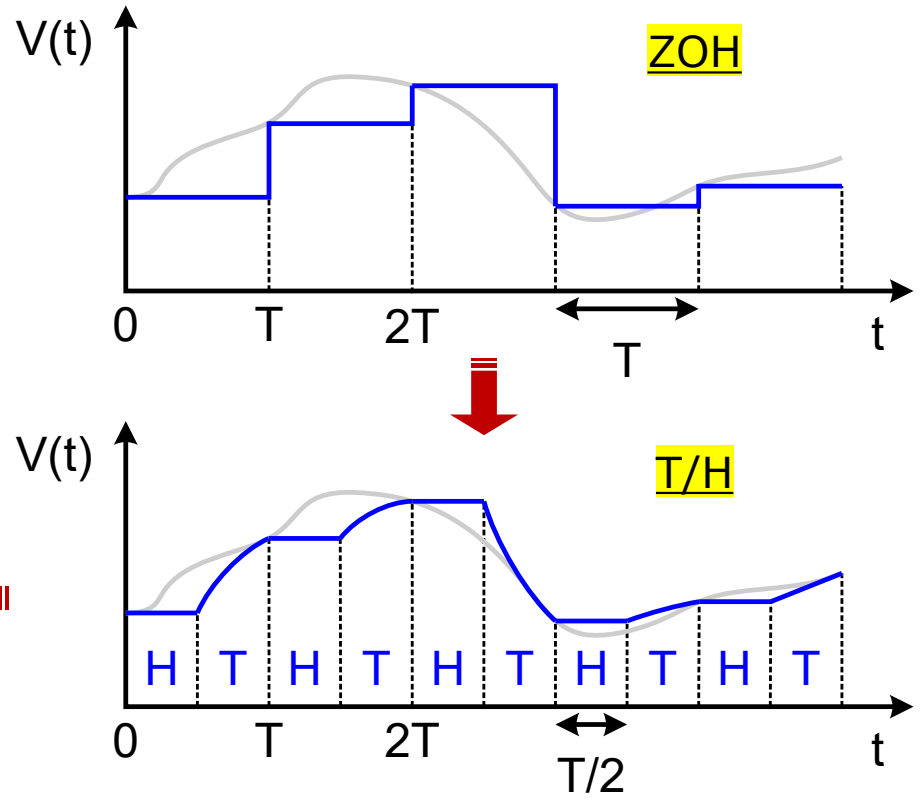
- Compact (PrA + latch)
- Fully dynamic operation → low power
- Single-phase clock (Low reset, high eval.)
- Both nodes ① and ② start @V_{DD}, end @Gnd → large CM kickback

MOS SAMPLE-AND-HOLD

ZOH, S/H, and T/H



$$R_{on}^{-1} = \mu C_{ox} \frac{W}{L}(V_{DD} - V_{th} - V_i)$$

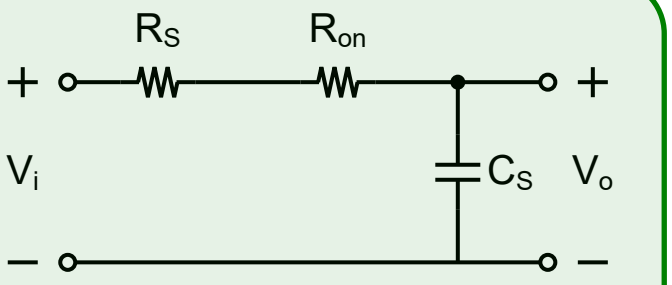


- A.k.a. S/H
- Zero acq. time
- Not realistic

- T/2 acq. time
- Practical

- MOS technology naturally suitable for implementing T/H
- Terms S/H and T/H are often used interchangeably

MOS Switch Performance



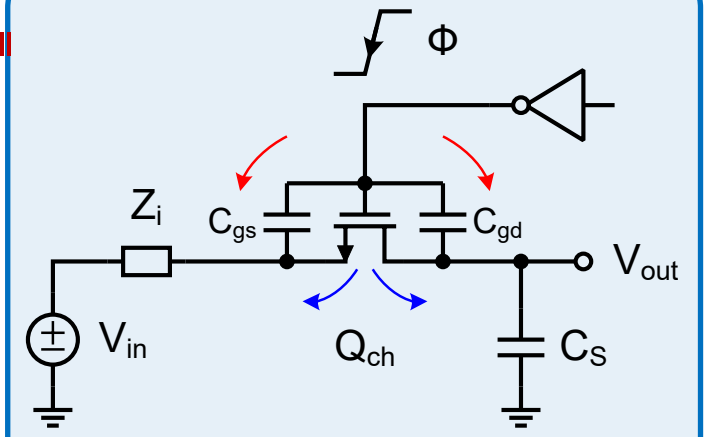
$$R_{on} = \frac{1}{\mu C_{ox}\frac{W}{L}(V_{DD} - V_{th} - V_i)} = \frac{L^2}{\mu Q_{ch}}$$

Accuracy	t_{acq}
0.1% (10b)	≥ 7τ
0.01% (13b)	≥ 9τ

Charge Inj.: $\Delta V = \frac{\Delta Q}{C_S} \approx \frac{1}{2}\frac{Q_{ch}}{C_S}$

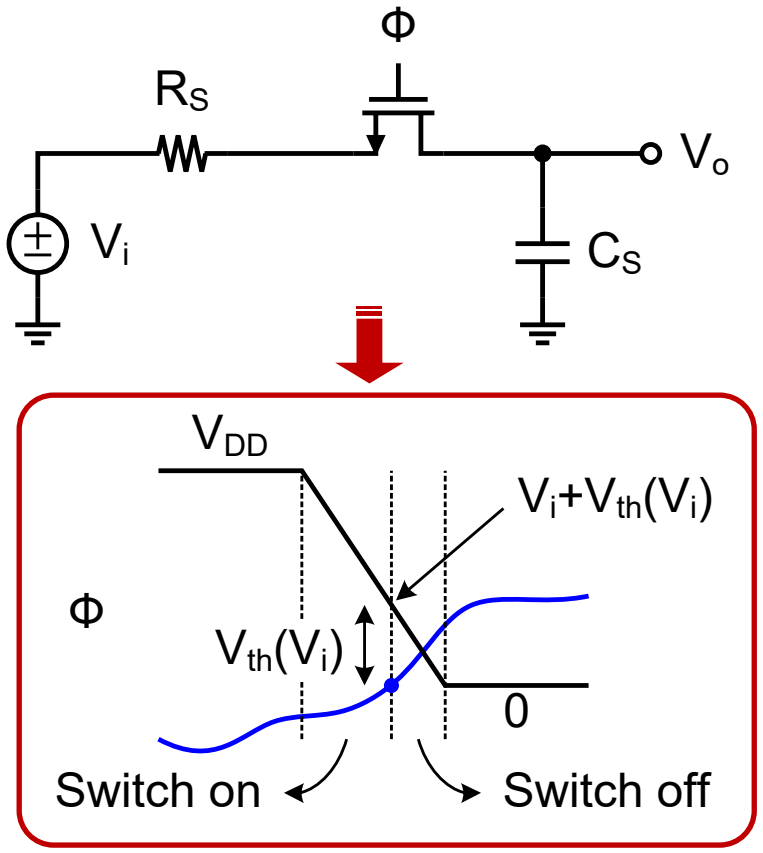
Tracking BW: $TBW \approx \frac{1}{R_{on}C_S} = \frac{\mu Q_{ch}}{L^2 C_S}$

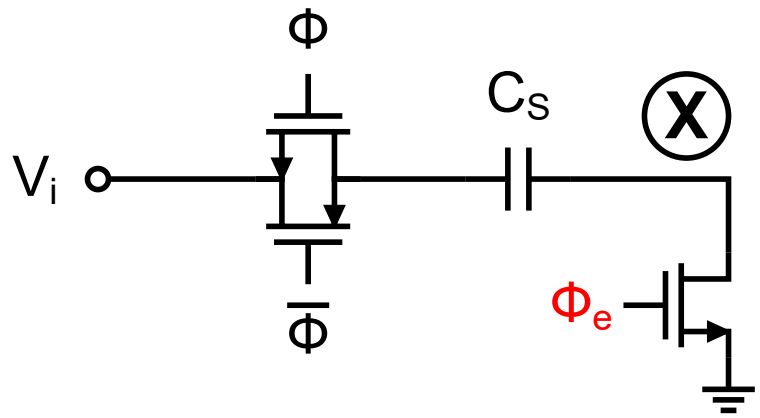
Performance metrics: $\frac{TBW}{\Delta V} \approx \frac{\mu Q_{ch}}{L^2 C_S} \cdot 2\frac{C_S}{Q_{ch}} = \frac{2\mu}{L^2}$



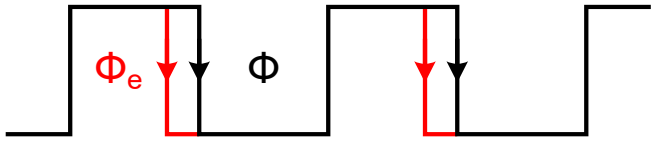
- Tracking bandwidth limited by switch on-resistance R_{on} (consider clock bootstrapping)
- Clock Feedthrough (CF) and Charge Injection (CI) are network/clock/signal dependent
- Technology scaling improves T/H performance! (a.l.a. leakage allows)

Aperture Error and Bottom-Plate Sampling



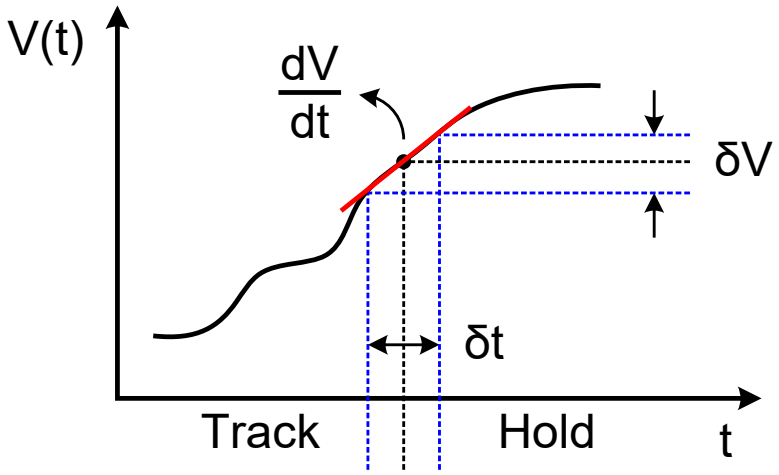


Bottom-Plate Sampling



- Exact sampling moment (aperture) depends on input in top-plate sampler
- Bottom-plate sampling avoids aperture error and CF+CI distortion by V.G. switching

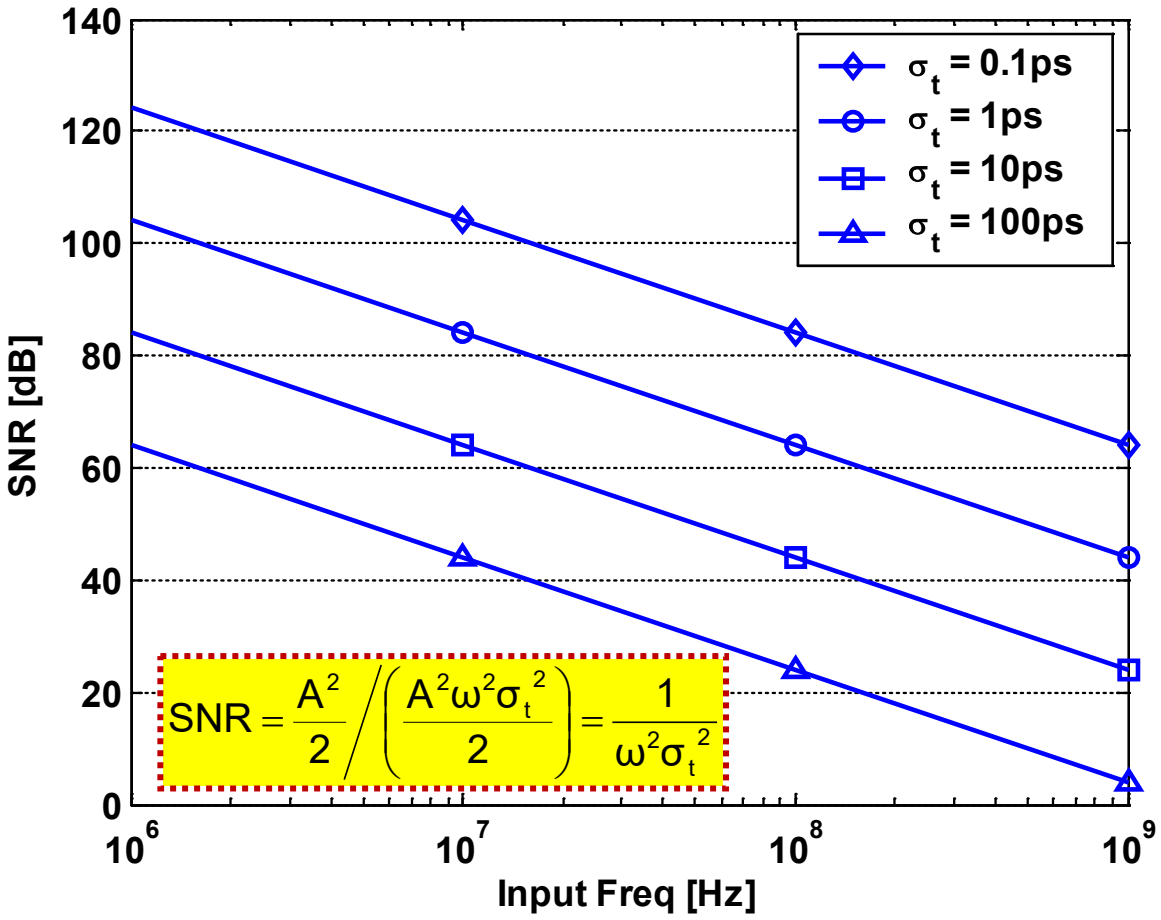
Aperture Jitter



Ref. [5]

$$\overline{\delta V^2(t)} = \frac{A^2\omega^2\sigma_t^{\,2}}{2}$$

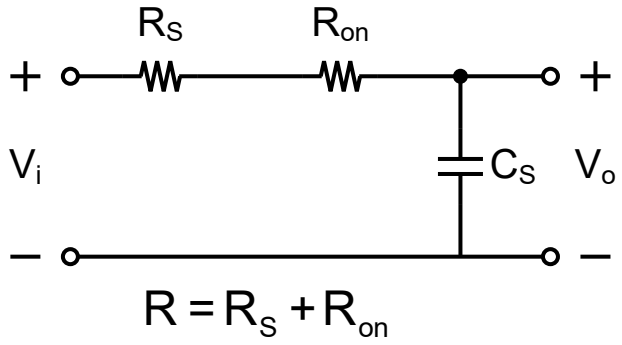
- Assuming sinewave
- Cyclostationary error



- Aperture jitter limits sampler SNR especially at high frequencies

Evaluating T/H Performance

Sampling network:

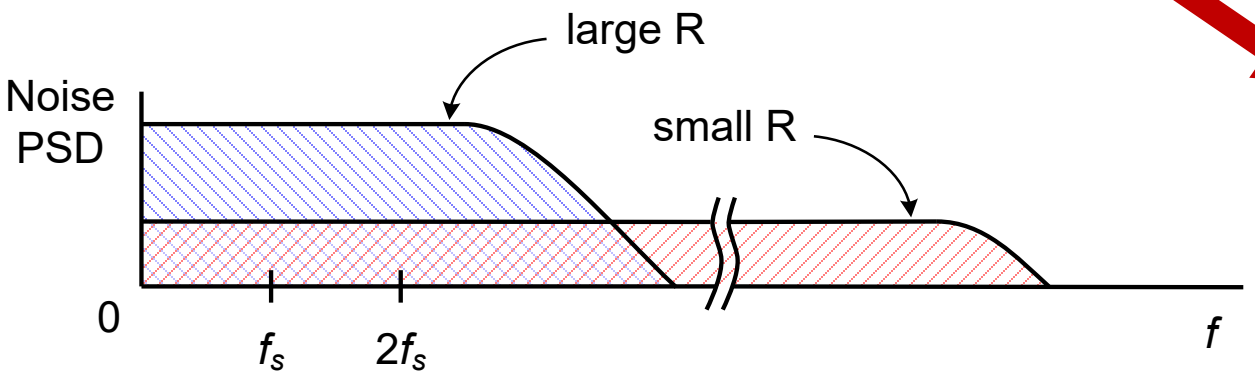


kT/C noise:

$$\overline{V_N^2} = \int_0^\infty 4kTR \cdot \left| \frac{1}{1 + j2\pi \cdot f \cdot RC_S} \right|^2 df = \frac{kT}{C_S}$$

$$SNDR = \frac{\overline{V_i^2}}{\overline{V_N^2} + \frac{A^2\omega^2}{2}\overline{\sigma_t^2} + \overline{V_{DISTO}^2}}$$

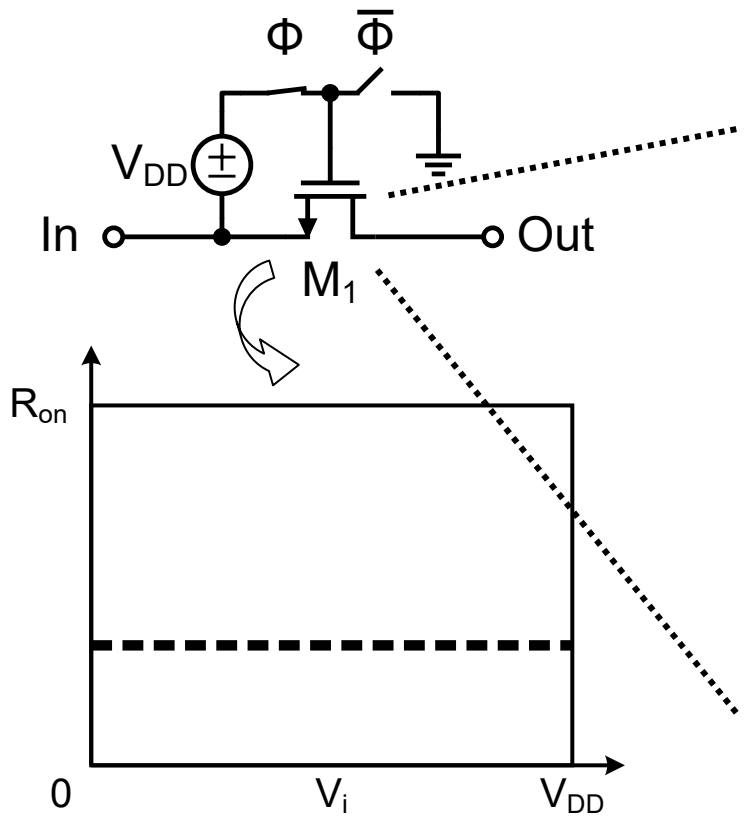
Noise | Jitter | Distortion

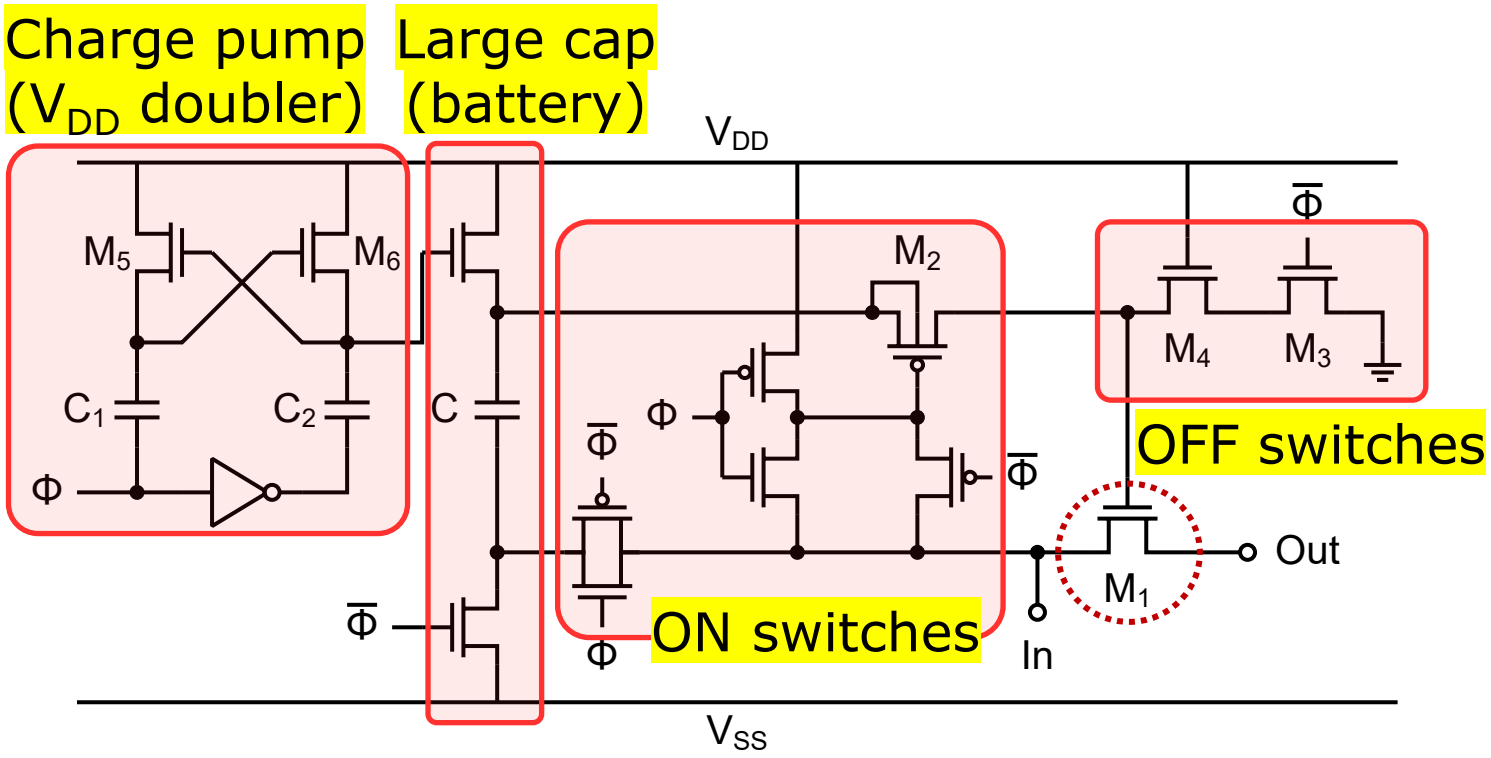


C_S	100pF	**1pF**	10fF
$\sqrt{\frac{kT}{C_S}}$	6.4μV	**64μV**	640μV

T = 300K

Clock Bootstrapping



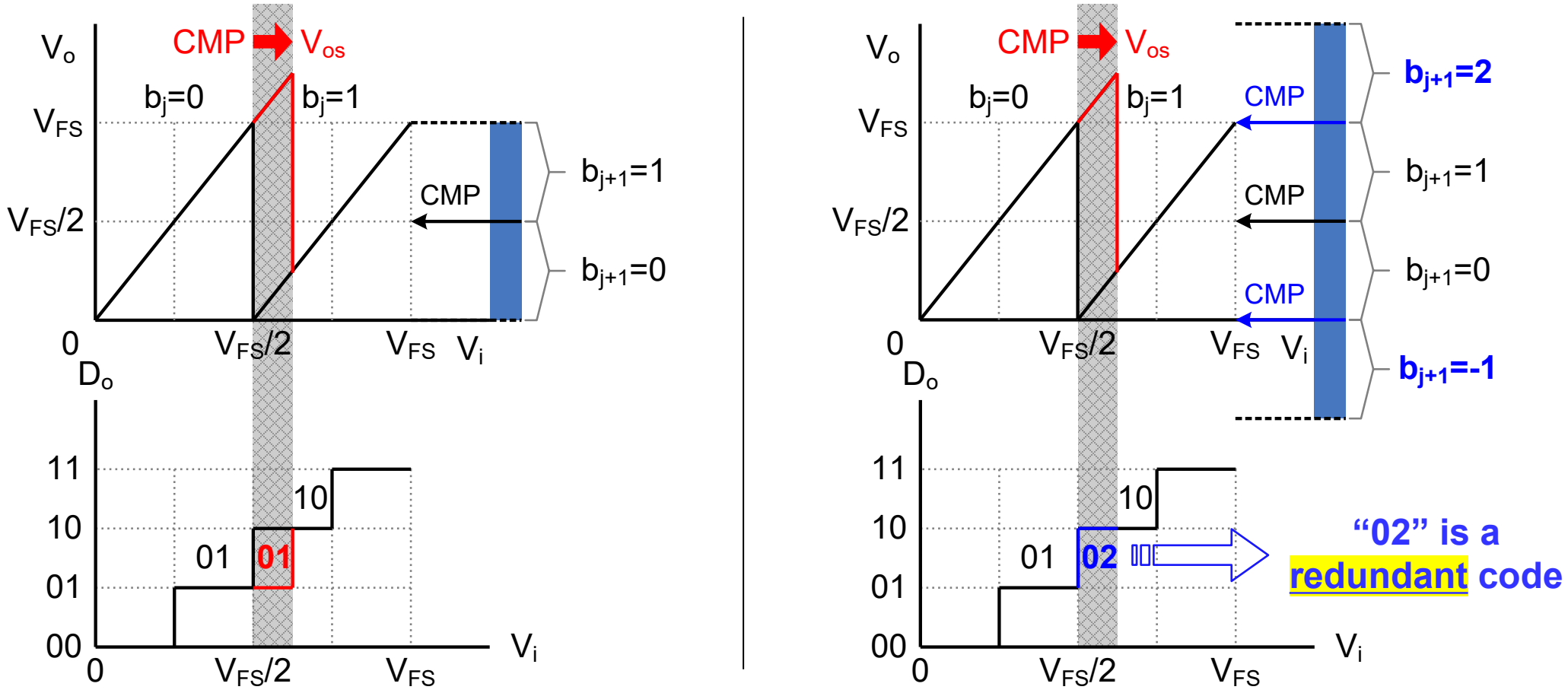


Ref. [6]

- Constant gate overdrive $V_{GS} = V_{DD}$ → R_{on} independent of V_{in} to the first order (body effect, charge sharing) → larger tracking BW, better linearity

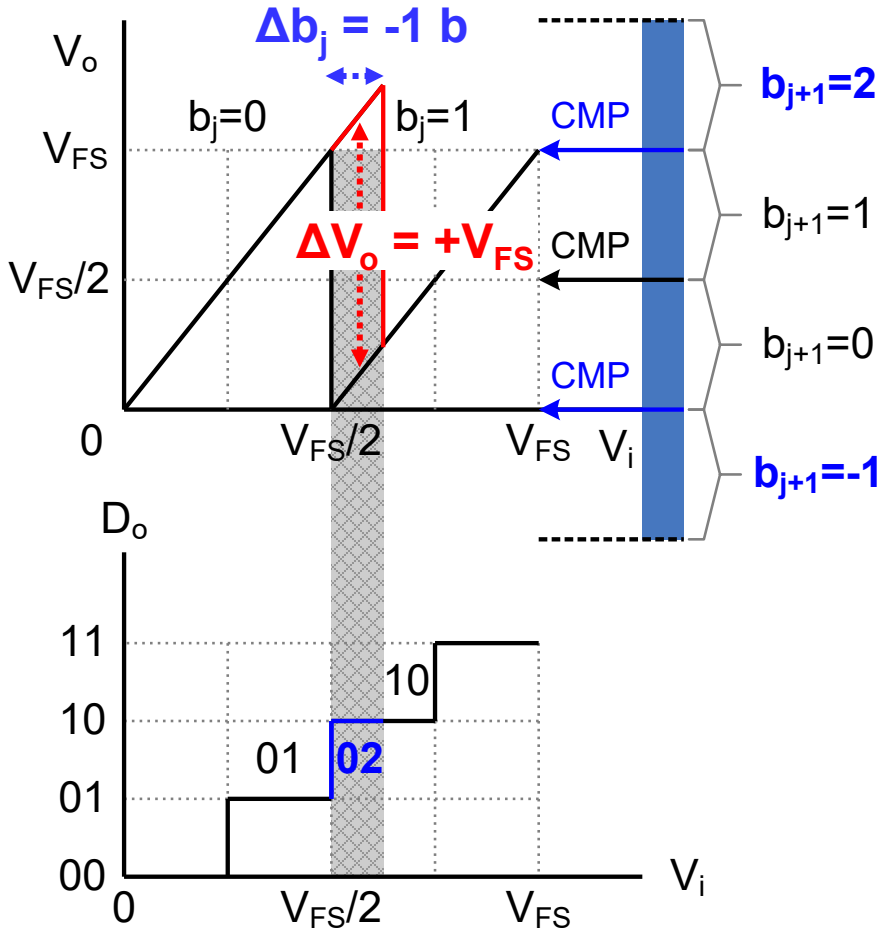
REDUNDANCY
(NO MULTI-STEP ADC WORKS WITHOUT IT!)

Comparator Offset in Pipelined ADC



- Nearly Zero Tolerance (< 1 LSB) on comparator offset!!
- **Architectural Redundancy** introduced by over-/under-range comparators

How Exactly Does Redundancy Work?



- Max tolerance of comparator offset is $\pm V_{FS}/4$ → simple comparators
- Key to understand redundancy:

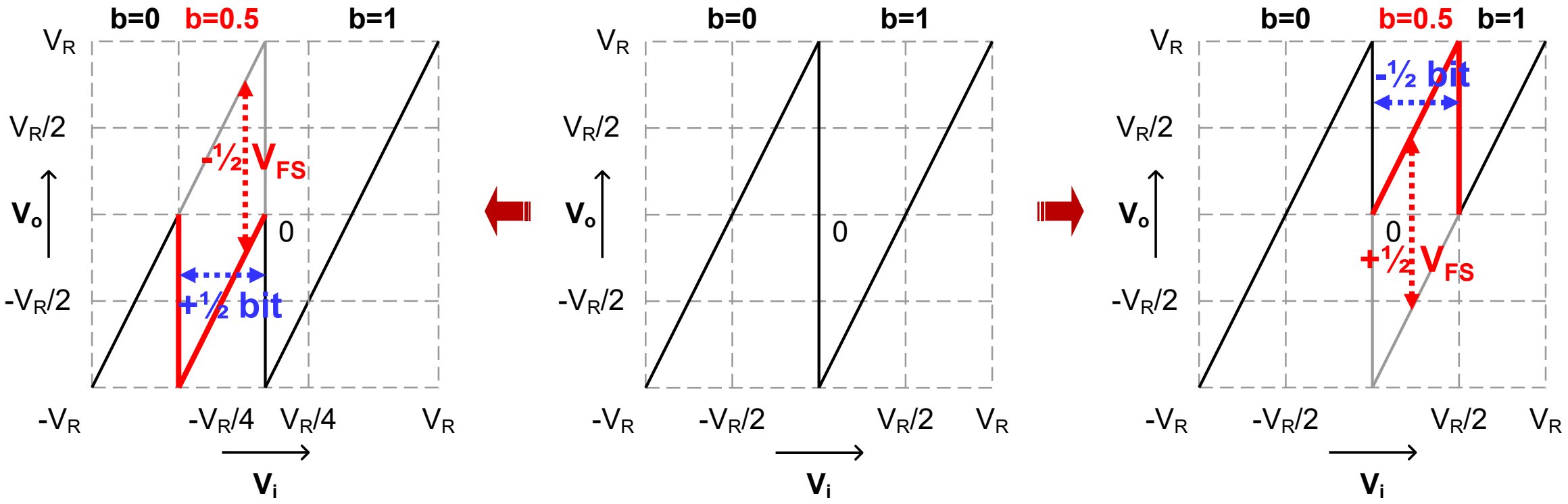
$$V_i = \left(b_j + \Delta b_j\right)\cdot\frac{V_{FS}}{2} + \frac{\left(V_o + \Delta V_o\right)}{2}$$

$$= \left(b_j - 1\right)\cdot\frac{V_{FS}}{2} + \frac{\left(V_o + V_{FS}\right)}{2}$$

$$= b_j\cdot\frac{V_{FS}}{2} + \frac{V_o}{2}$$

- Complementary **analog** and **digital** errors cancel each other w/ redundancy

From 1-bit To 1.5-bit Architecture

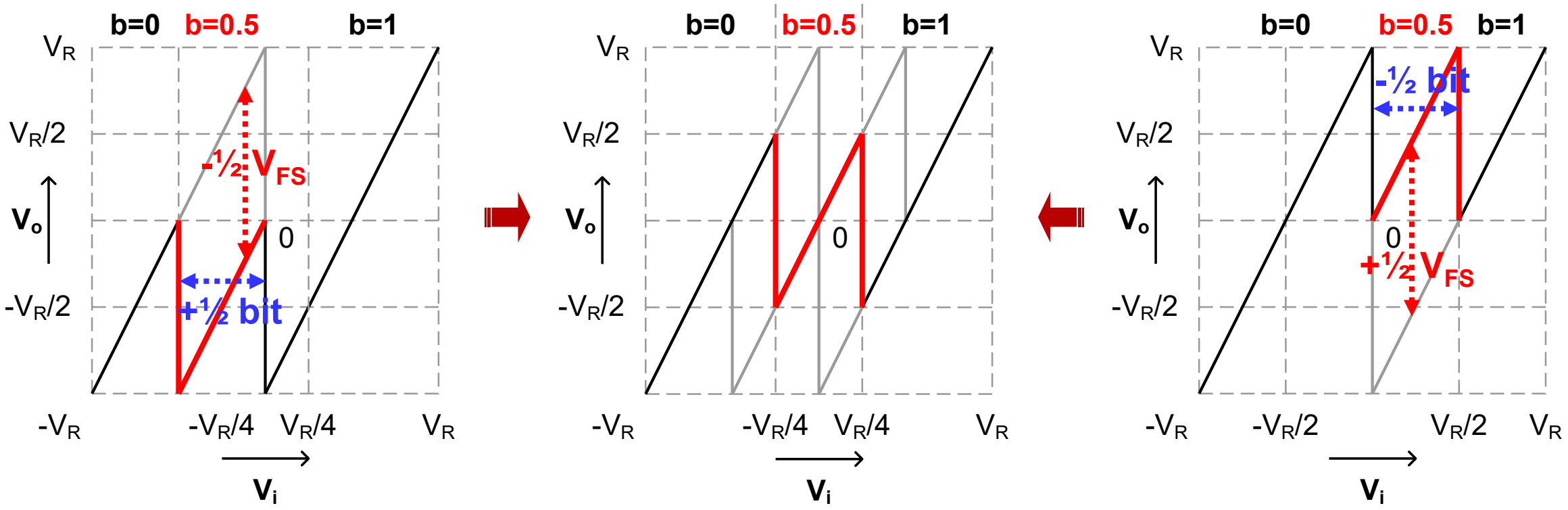


$$\Delta b + \frac{\Delta V_o}{2} = 0$$

1-bit residue TF
→ no redundancy

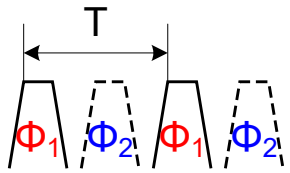
$$\Delta b + \frac{\Delta V_o}{2} = 0$$

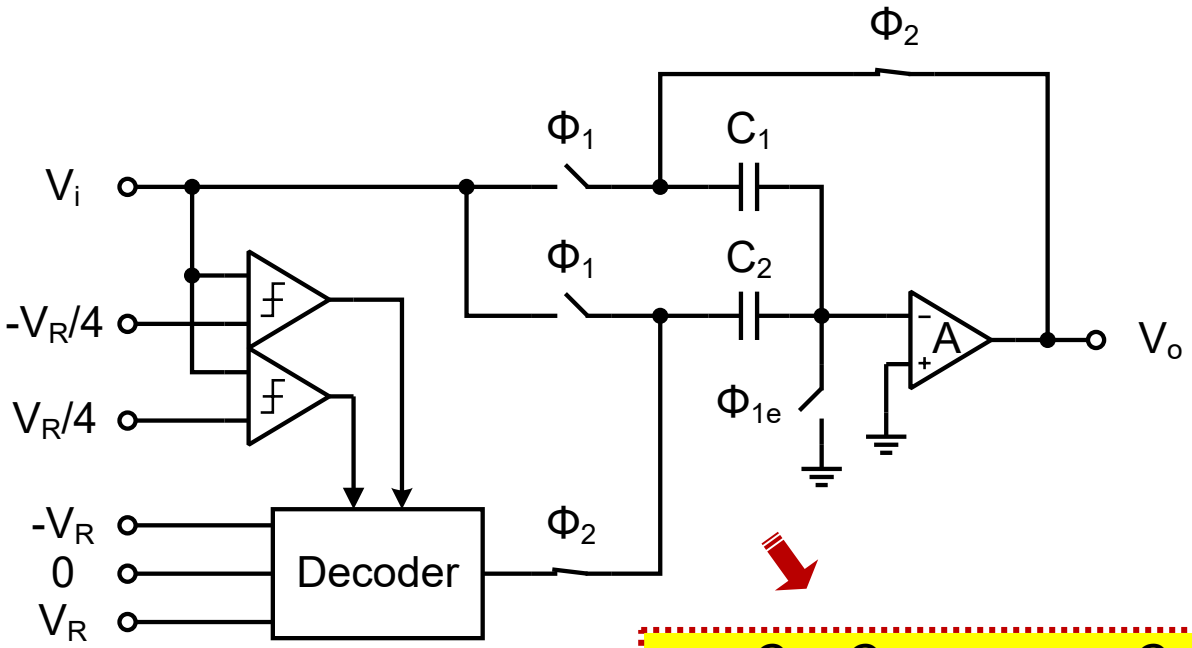
From 1-bit To 1.5-bit Architecture

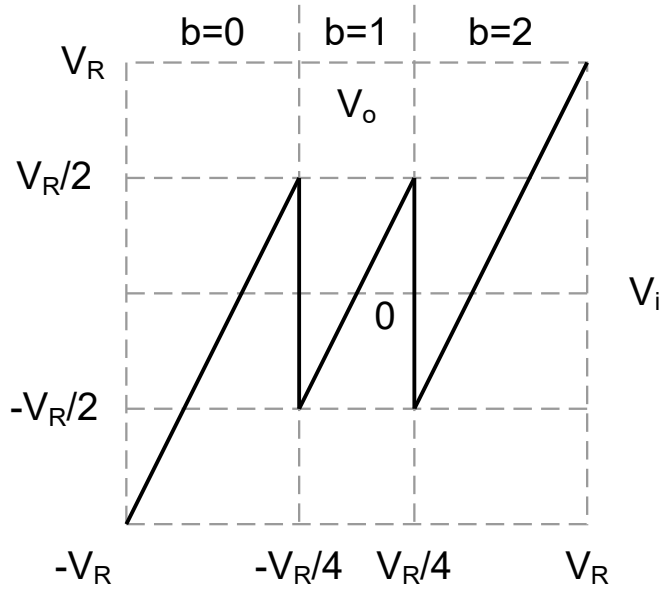


- **1.5-bit residue TF:** 3 decision levels → ENOB = $\log_2 3 \approx 1.58$
- Max tolerance of comparator offset is $\pm V_R/4$
- Also known as Sweeny-Robertson-Tocher (SRT) division algorithm

1.5-bit Multiplier DAC (MDAC)







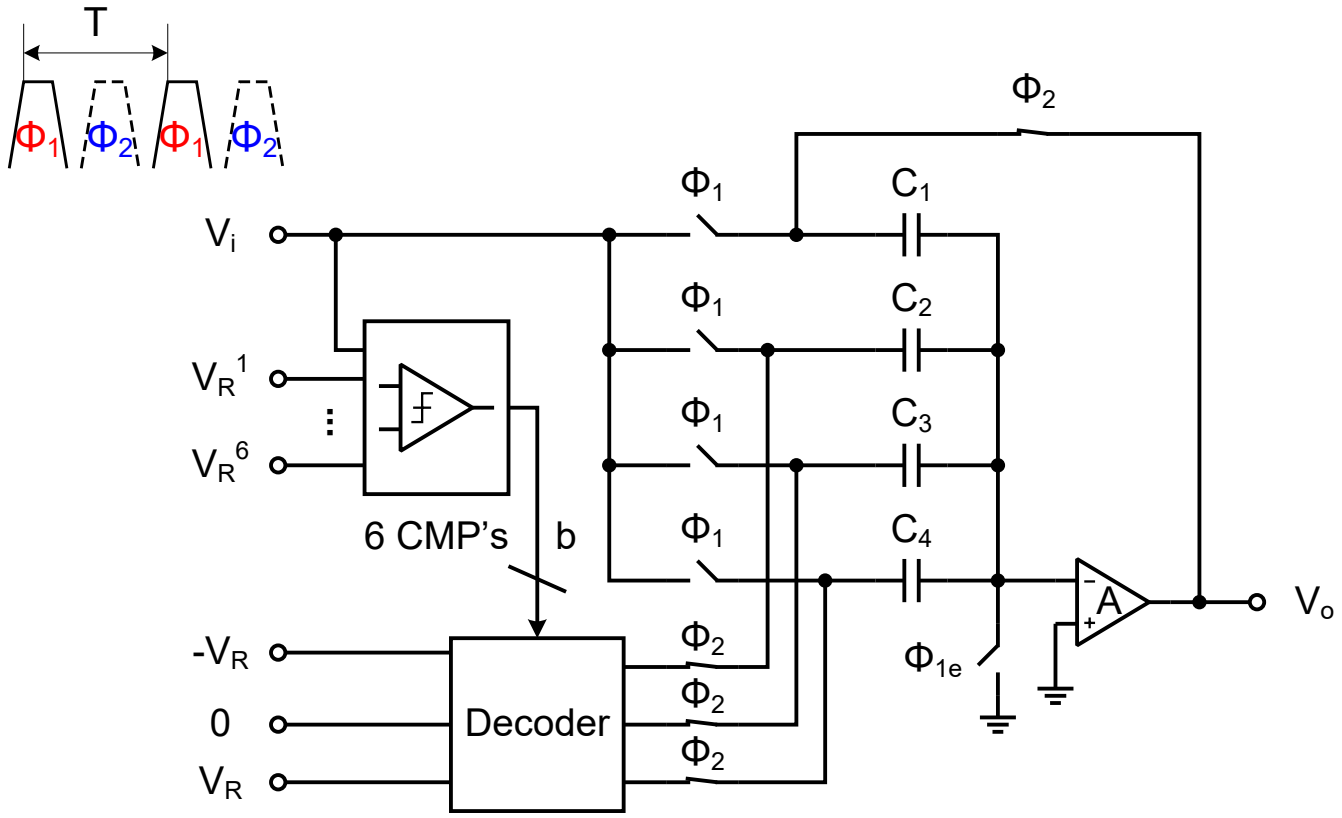
$$V_o = \frac{C_1 + C_2}{C_1} \cdot V_i + (1-b)\frac{C_2}{C_1} \cdot V_R$$

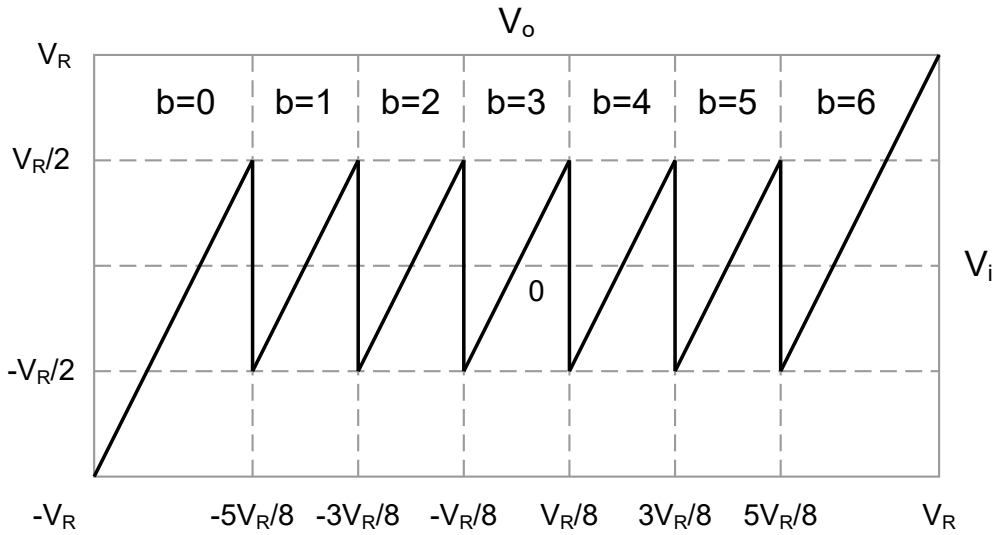
$$V_o = 2 \cdot V_i + (1-b) \cdot V_R$$

- 2X gain + 3-level DAC + summation all integrated
- Can be generalized to n.5-bit architectures

Ref. [7]

2.5-bit MDAC

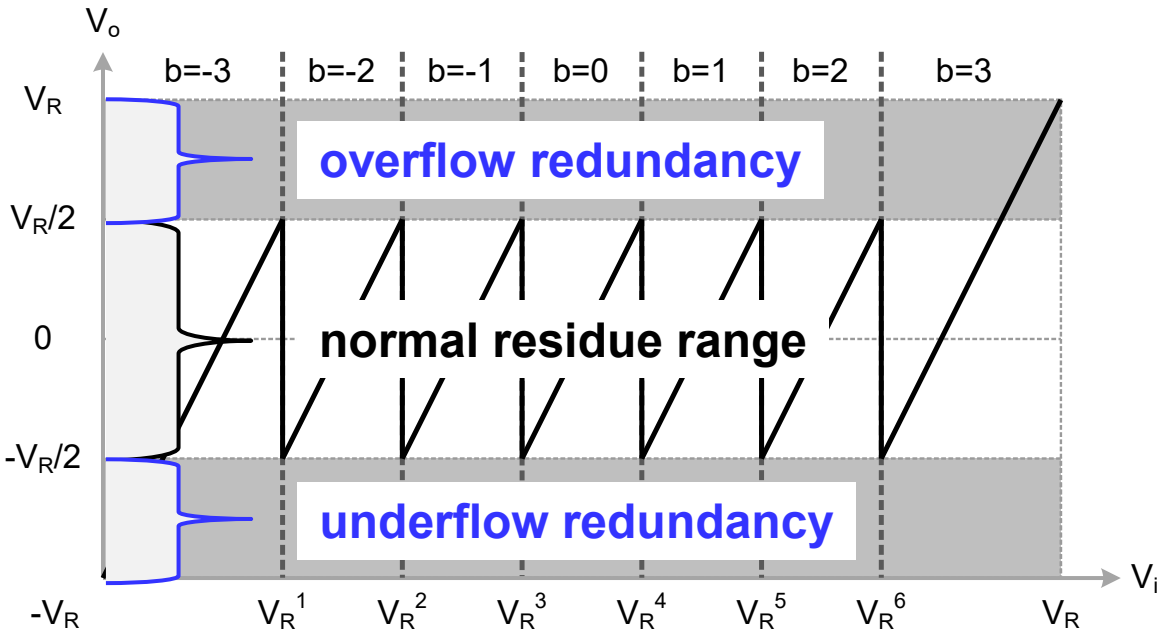




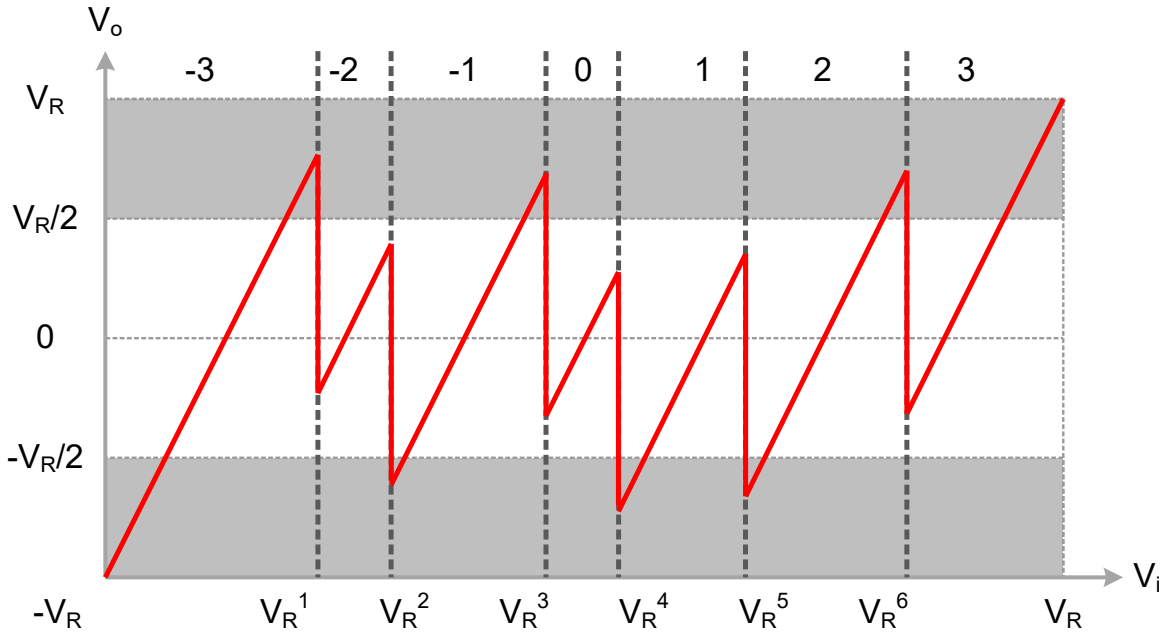
$$V_o = 4 \cdot V_i + (3 - b) \cdot V_R$$

- 4X gain + 7-level DAC + summation all integrated

Error-Free Conversion With Redundancy



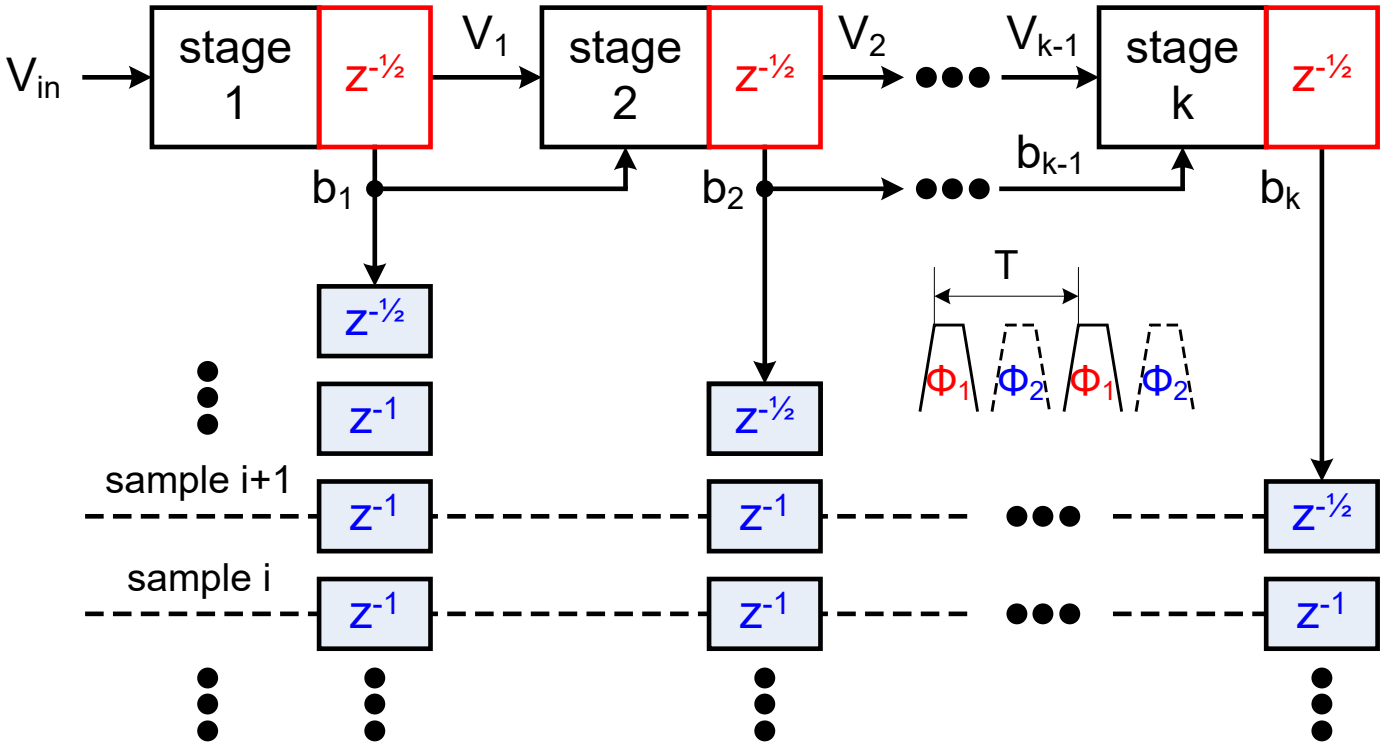
Ideal Residue TF



A More Realistic Residue TF

- Comparator and residue amplifier offsets absorbed by **Internal Redundancy**
  - Error-free conversion when residue is confined in the range [$-V_R$, $+V_R$]

Pipeline Redundant Bits Assembly



For pipelined ADC,

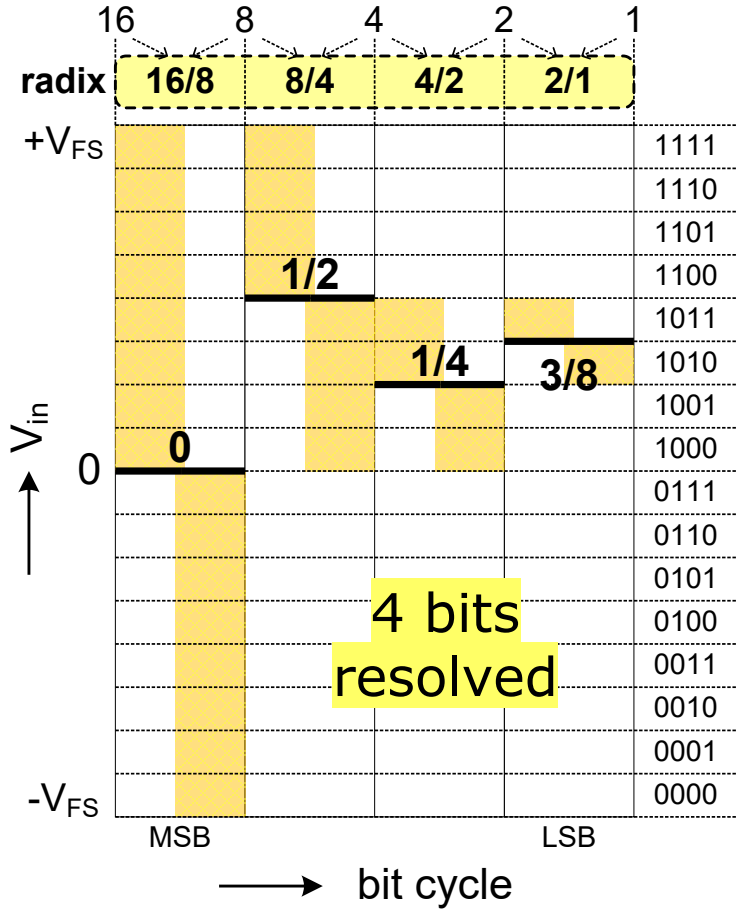
$$D_o = \underline{W} \cdot \underline{D}_o = \sum_j w_j \cdot b_j$$

b_1=2	**1**	**0**				
b_2=0		**0**	**0**			
b_3=1			**0**	**1**		
b_4=2				**1**	**0**	
b_5=1					**0**	**1**
D_o	**1**	**0**	**1**	**0**	**0**	**1**

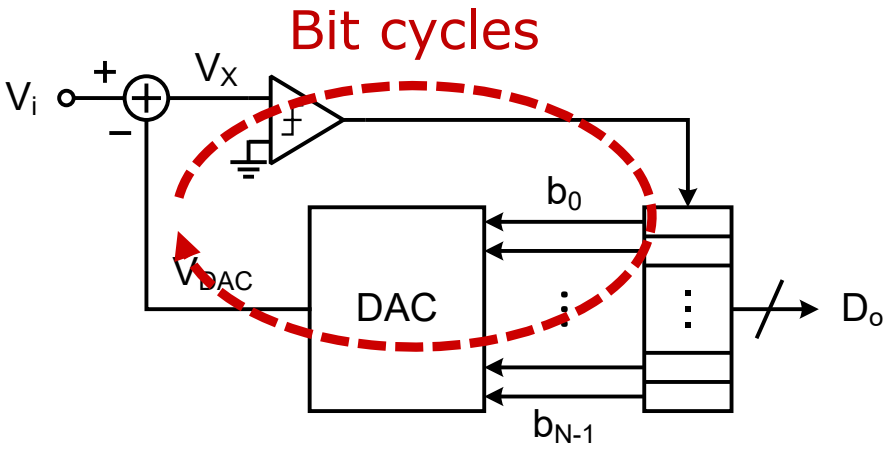
MSB LSB

Decision Vector: $\underline{D}_o = \{b_1, b_2, \ldots, b_k\}$, **Scalar Decision**: $D_o = f(\underline{D}_o)$

How Does Redundancy Work For SAR?

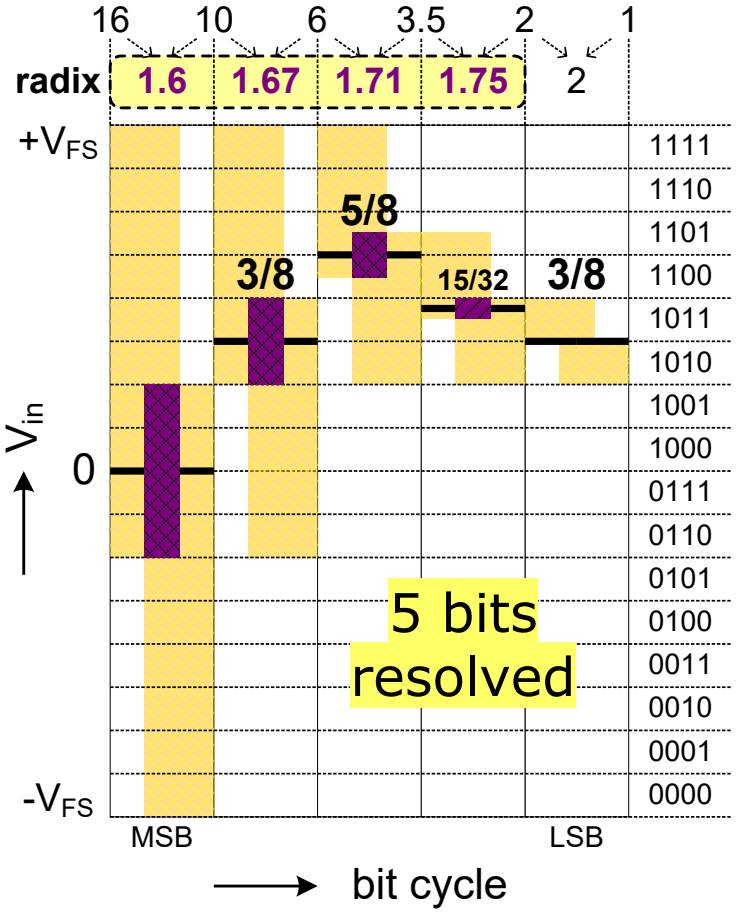


Binary search



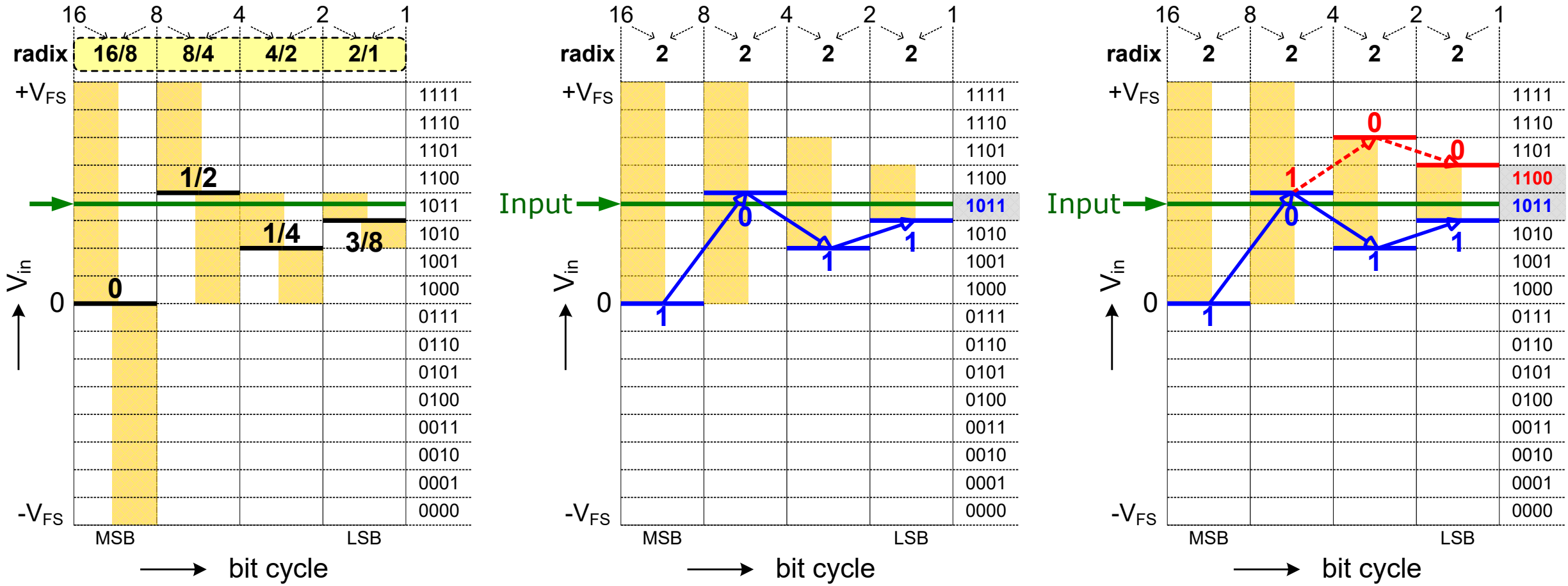
w/o redundancy

w/ redundancy



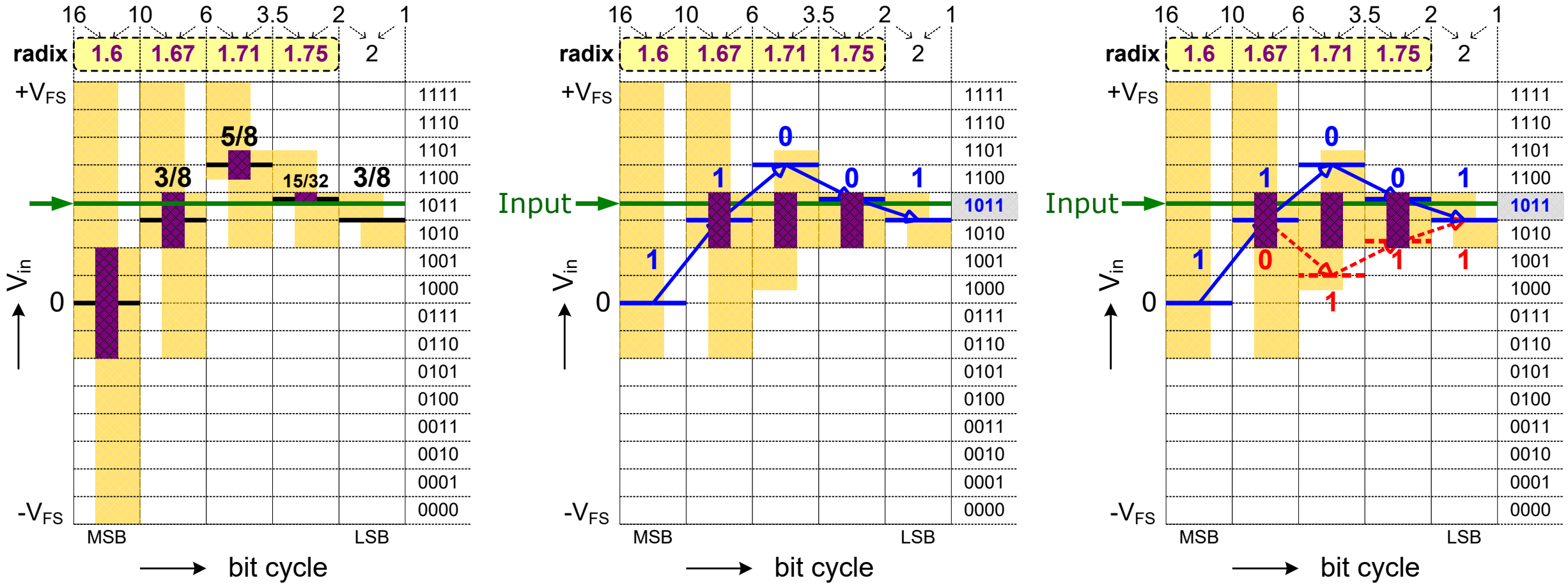
Sub-binary search

Binary Search Revisited



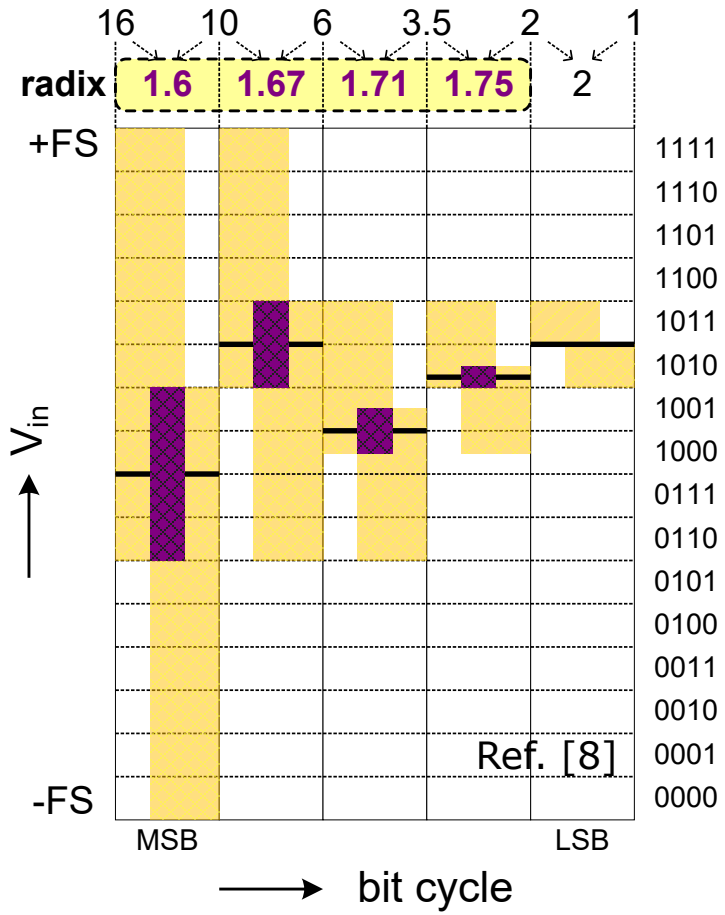
- Binary search is efficient, but displays zero error tolerance

Sub-Binary Search – Redundancy

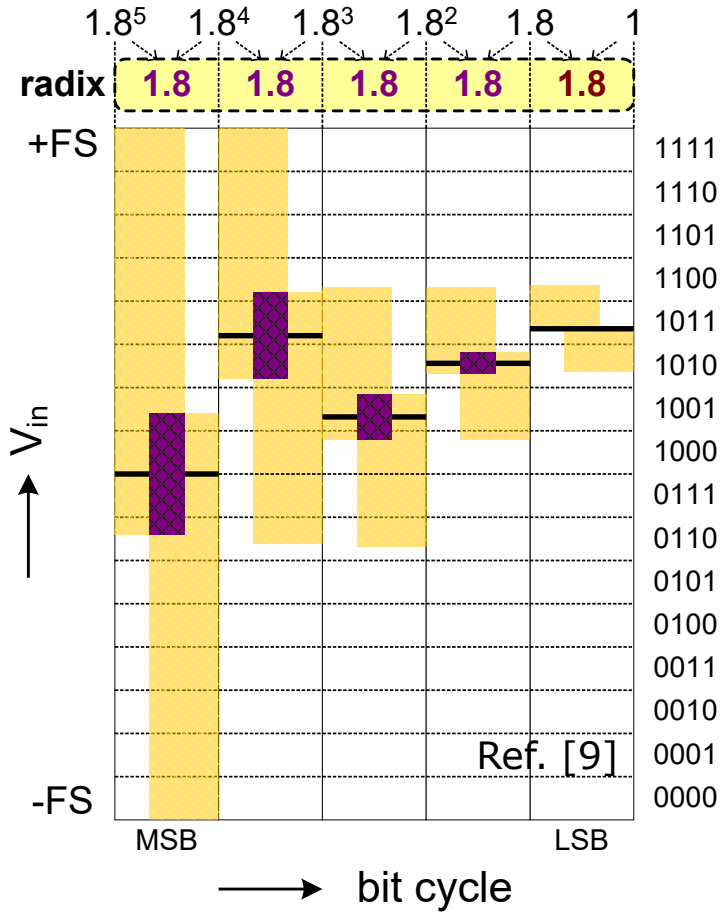


- Redundancy of sub-binary search can absorb intermediate decision errors

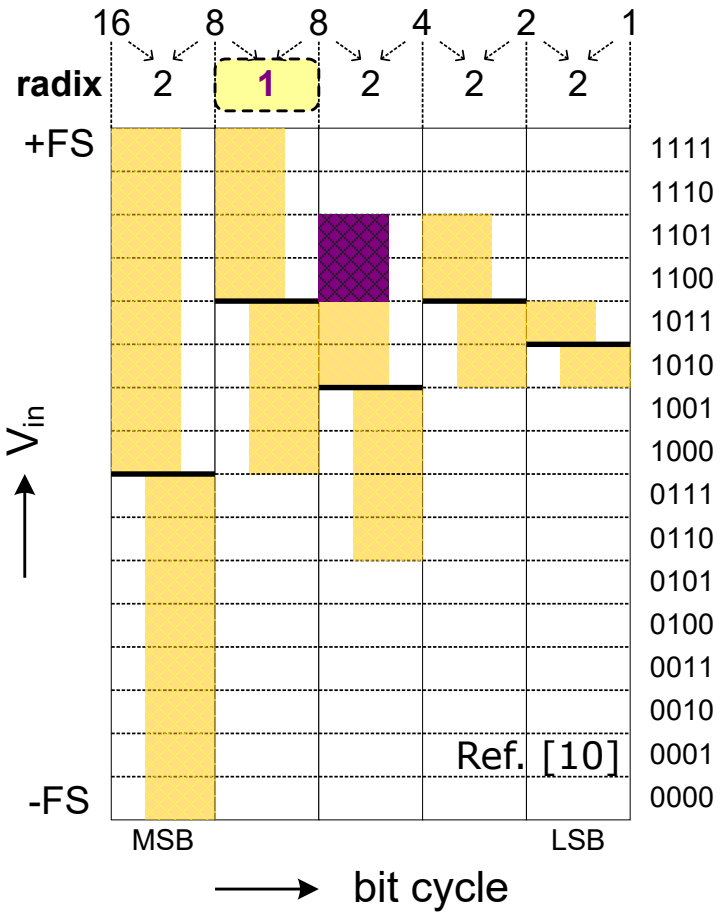
Sub-Radix-2 Implementations



Nonuniform radix



Uniform radix



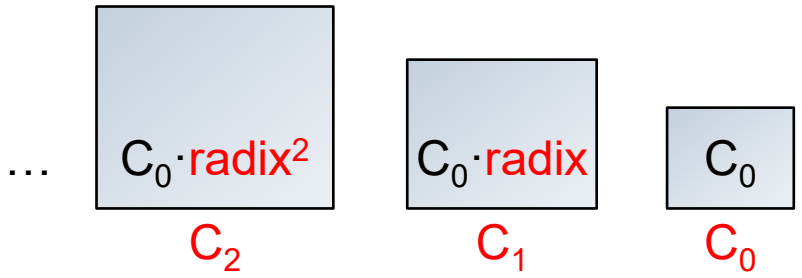
Occasional sub-radix

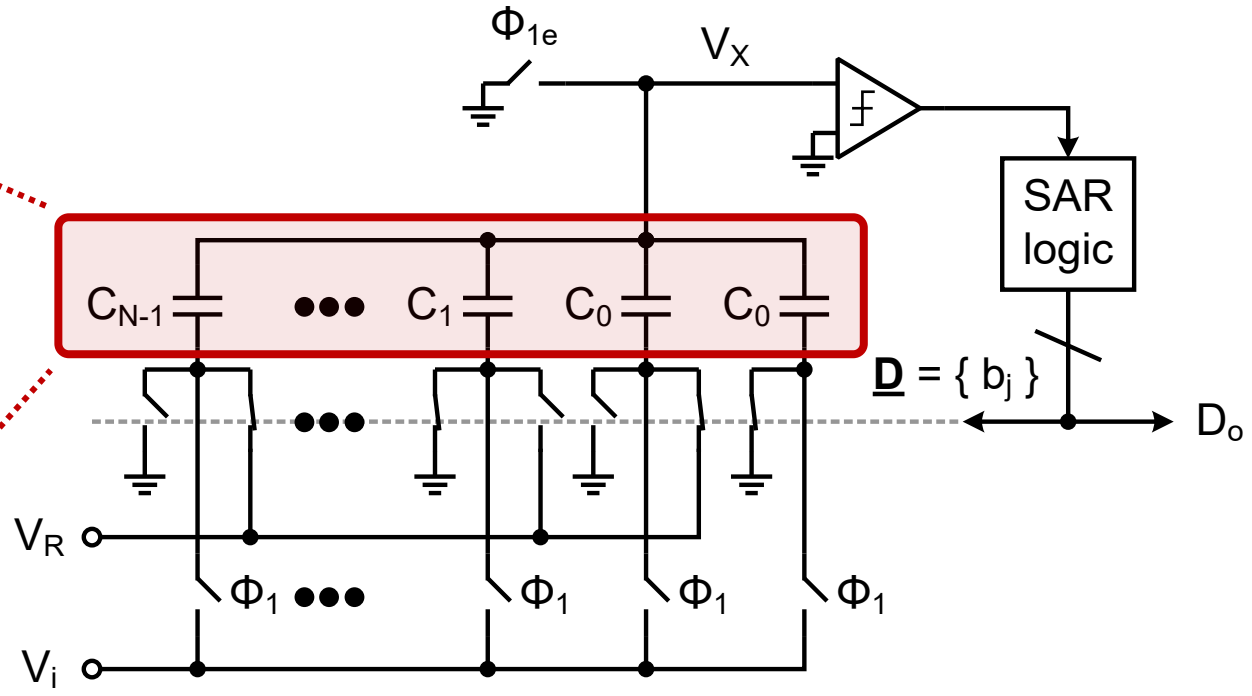
Uniform Sub-Radix-2 SAR ADC

For redundant bits,

$$\frac{C_{j+1}}{C_j} = \text{radix} < 2$$

e.g., in a uniform radix SAR





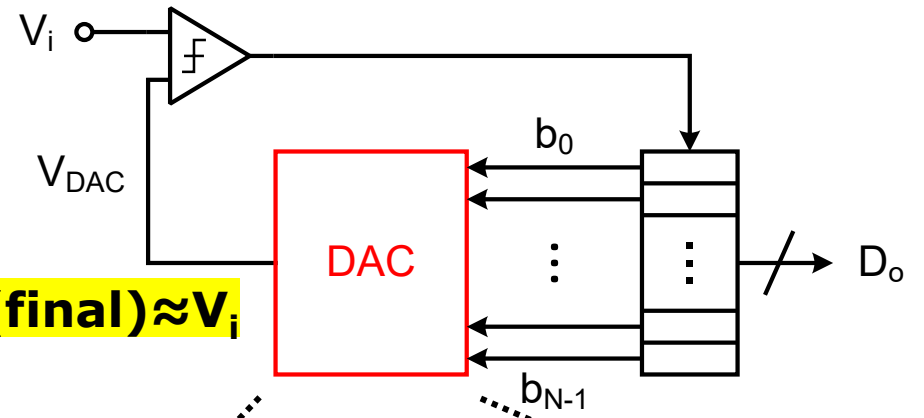
- Hard-coded radices dependent on the DAC capacitor ratios
- Minimal analog complexity, no additional decoding effort
- Suitable for high-resolution SAR ADC (that requires radix calibration)

SAR Redundant Bits Assembly

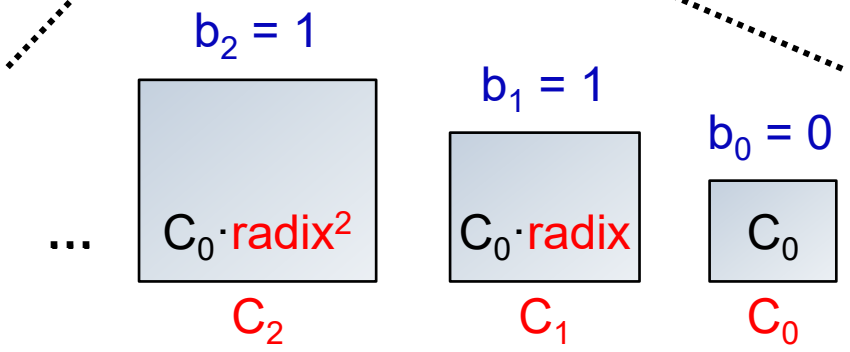
$\{w_j\}$: bit weights

$$D_o \approx \sum_{j=0}^{N-1}\left(\frac{C_j}{\sum C_j}\right)\cdot b_j = \sum_{j=0}^{N-1} w_j \cdot b_j$$

$$\sum Q_{DAC} = V_R \sum_{j=0}^{N-1} C_j \cdot b_j = V_i \sum_{j=0}^{N-1} C_j + \text{Noise}$$



$V_{DAC}(final) \approx V_i$



$$D_o = \underline{W} \cdot \underline{D_o} = \sum_j w_j \cdot b_j$$

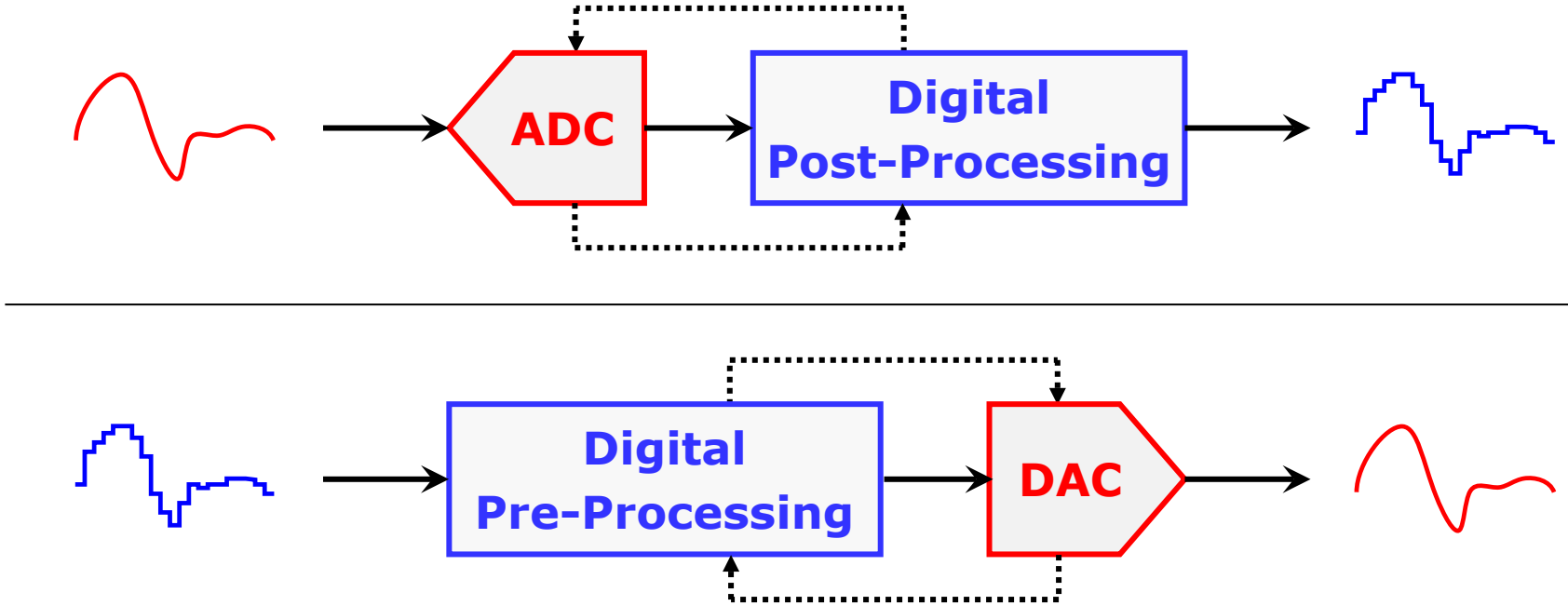
e.g., uniform radix 1.8

$b_4=1$	$\mathbf{1}\cdot 1.8^4$
$b_3=0$	$\mathbf{0}\cdot 1.8^3$
$b_2=1$	$\mathbf{1}\cdot 1.8^2$
$b_1=1$	$\mathbf{1}\cdot 1.8^1$
$b_0=0$	$\mathbf{0}\cdot 1.8^0$
D_o	**15.5**

Decision Vector: $\underline{D_o} = \{b_{N-1}, b_{N-2},\ldots,b_0\}$, **Scalar Decision**: $D_o = f(\underline{D_o})$

DIGITAL CORRECTABILITY AND CALIBRATION

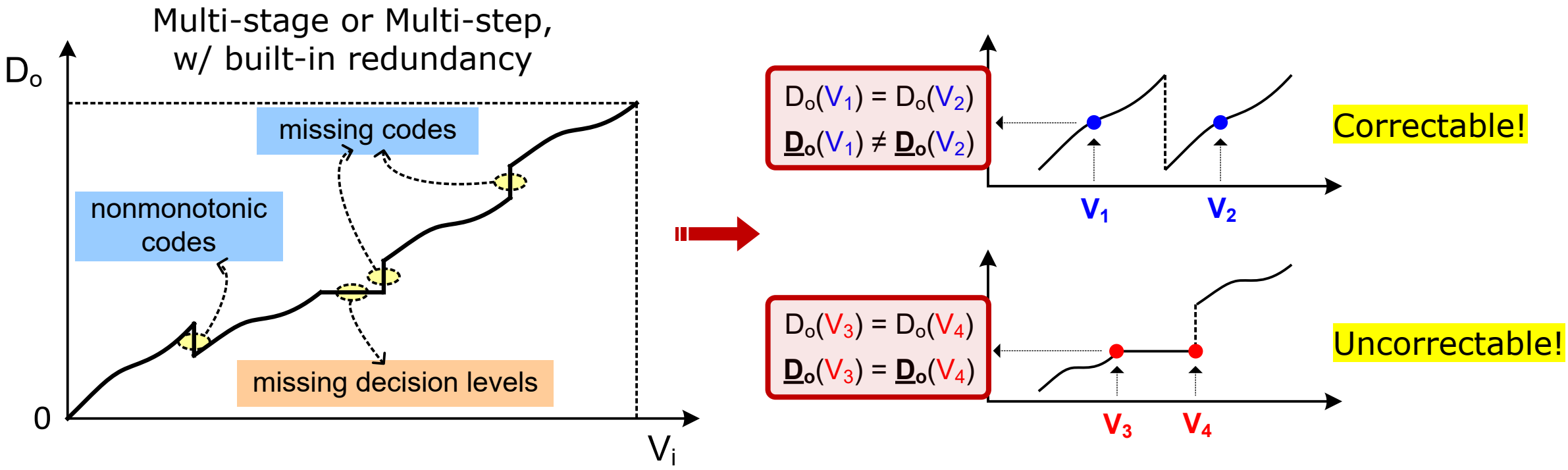
Digital Calibration



Refs. [11-27]

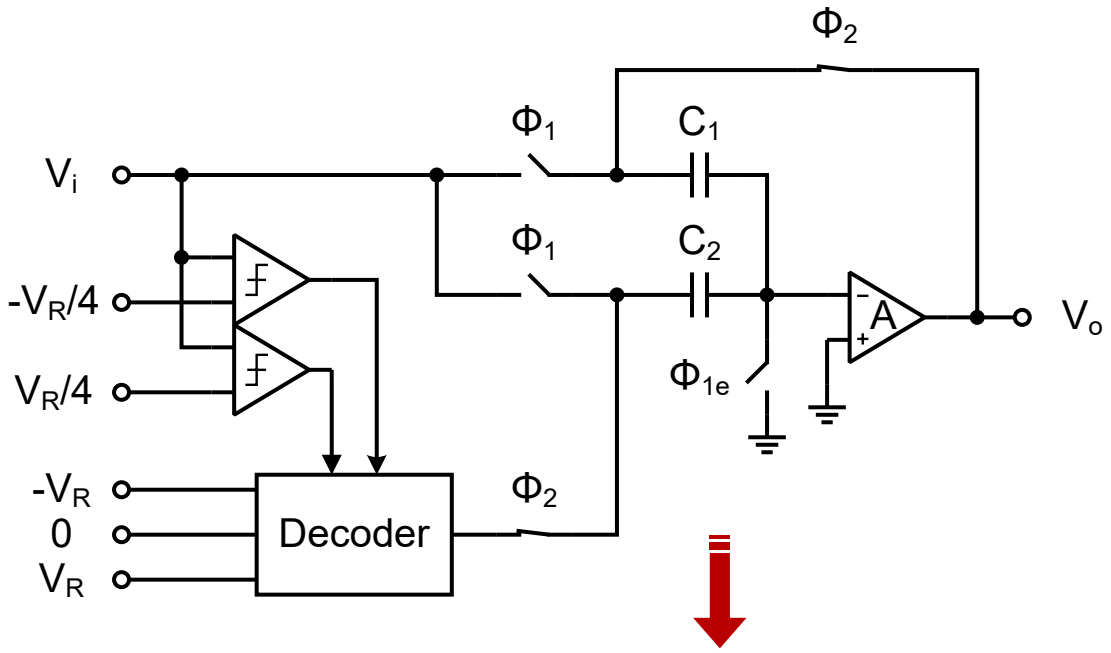
- Many analog errors in data converters, e.g., offset, gain error, nonlinearity, can be corrected digitally, which is widely termed **Digital Calibration**
- Digital calibration (aka digital enhancement, digital assistance, self-healing) is an opportunity and gift presented by the Moore's law!

ADC Digital Correctability



- Decision Vector $\underline{\mathbf{D_o}}$ reveals more info about input than Scalar Decision D_o
- A.l.a $\underline{\mathbf{D_o}}$ is unique (even D_o is not), a LUT can in theory be found to recover the input faithfully (i.e., mapping from A → D must be unique, thus, reversible)

Residue Errors in Pipelined ADC (1.5-b)



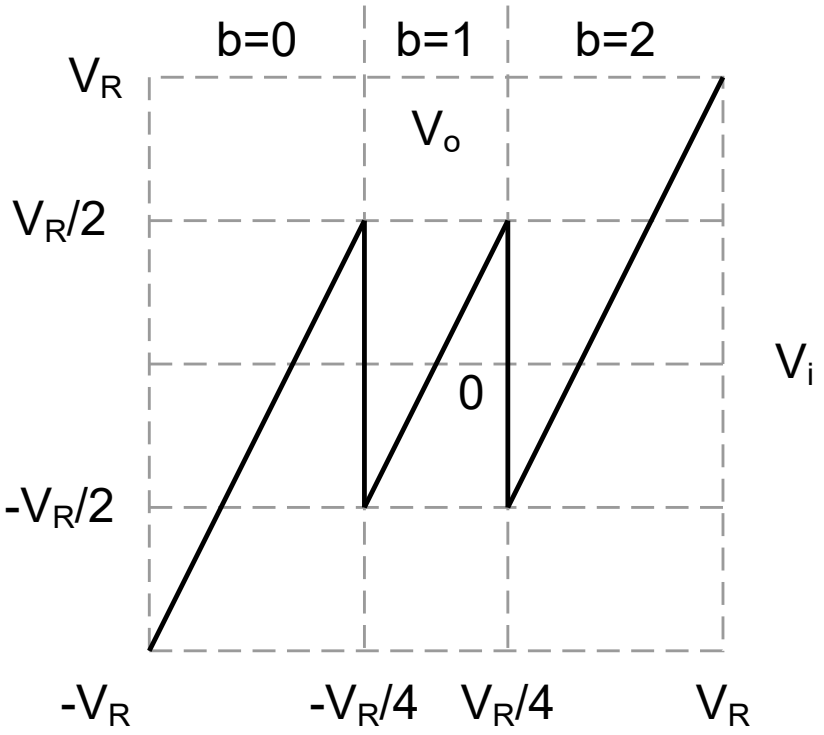
- Static errors:
  - Capacitor mismatch/nonlinearity
  - RA finite gain and nonlinearity
  - S/H nonlinearity
- Dynamic errors:
  - RA slewing/settling error
  - Switch-induced errors
  - Reference settling error
  - Memory errors

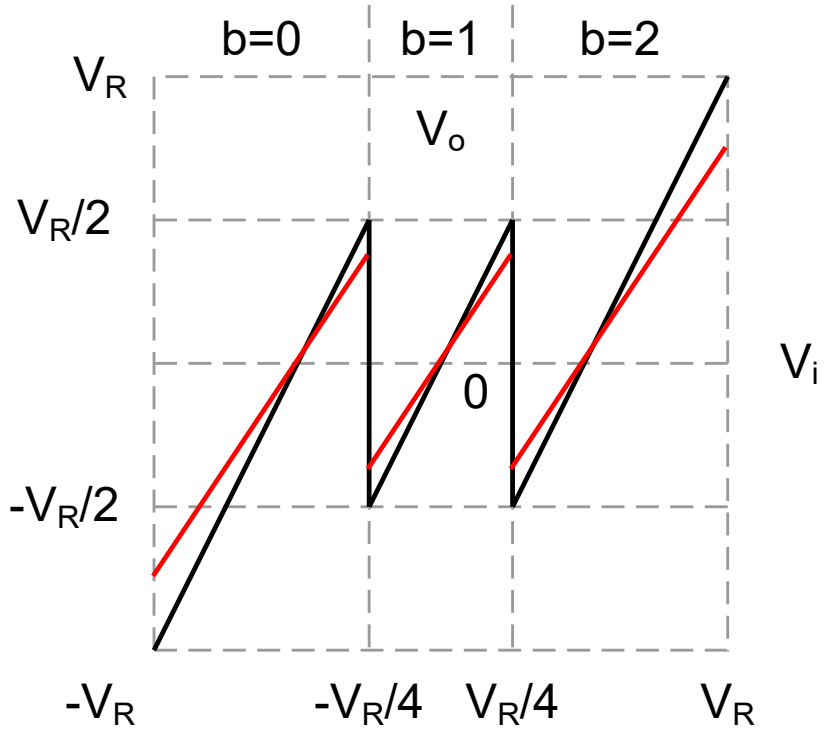
$$V_o = \frac{C_1 + C_2}{C_1 + \frac{C_1 + C_2}{A(V_o)}} \cdot f_{S/H}(V_i) + \frac{C_2}{C_1 + \frac{C_1 + C_2}{A(V_o)}} \cdot (1-b) \cdot V_R$$

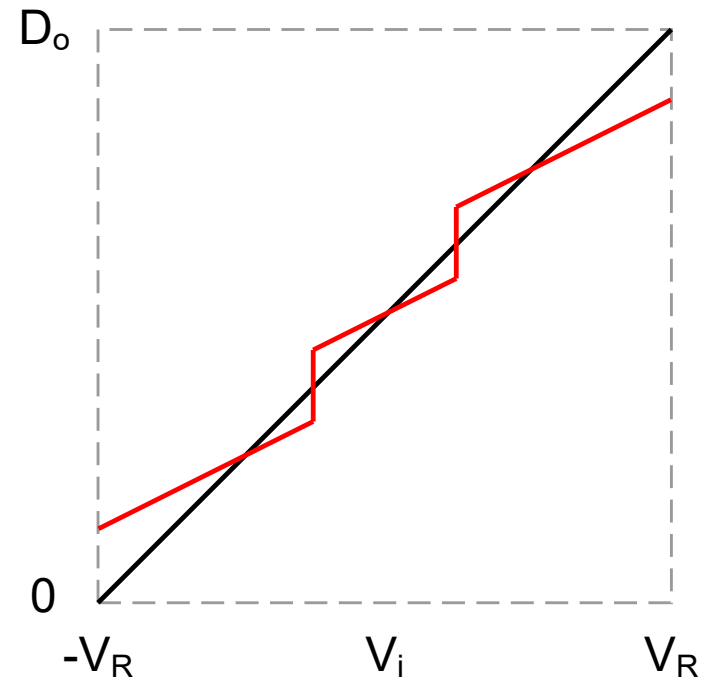
Ideal residue TF:

$$V_o = 2 \cdot V_i + (1-b) \cdot V_R$$

Residue Errors in Pipelined ADC (1.5-b)

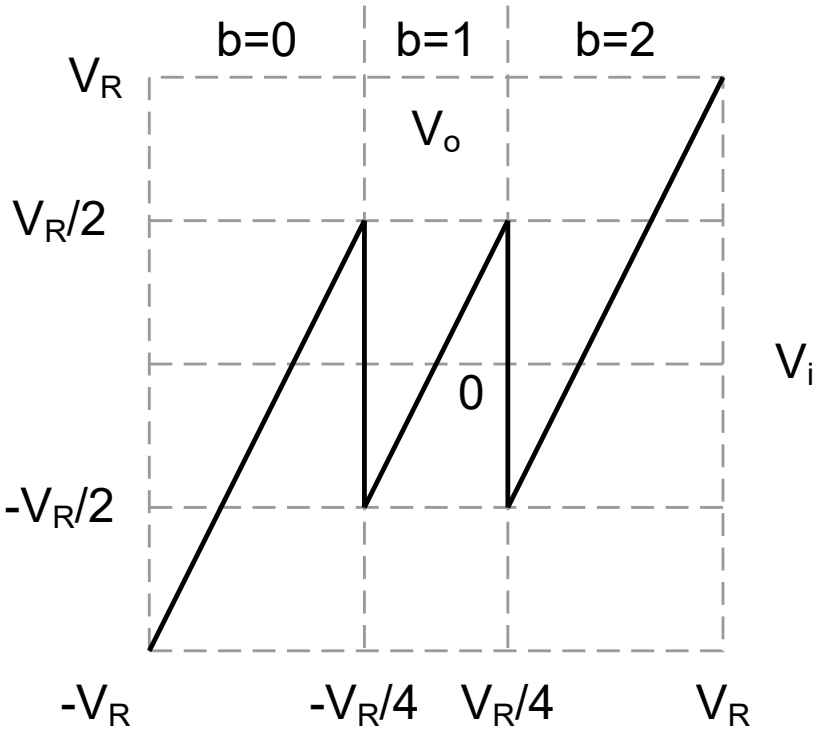


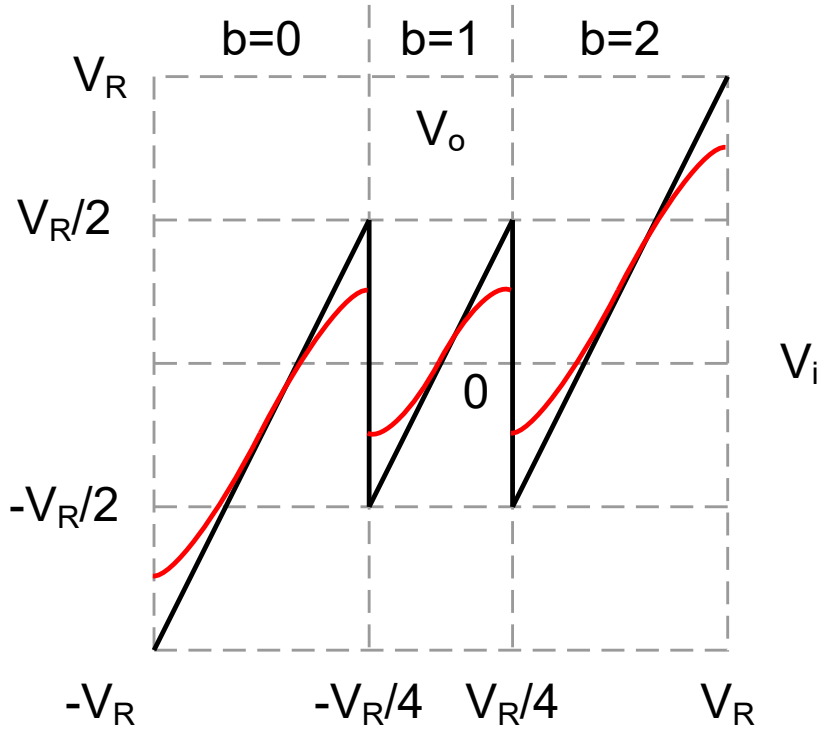


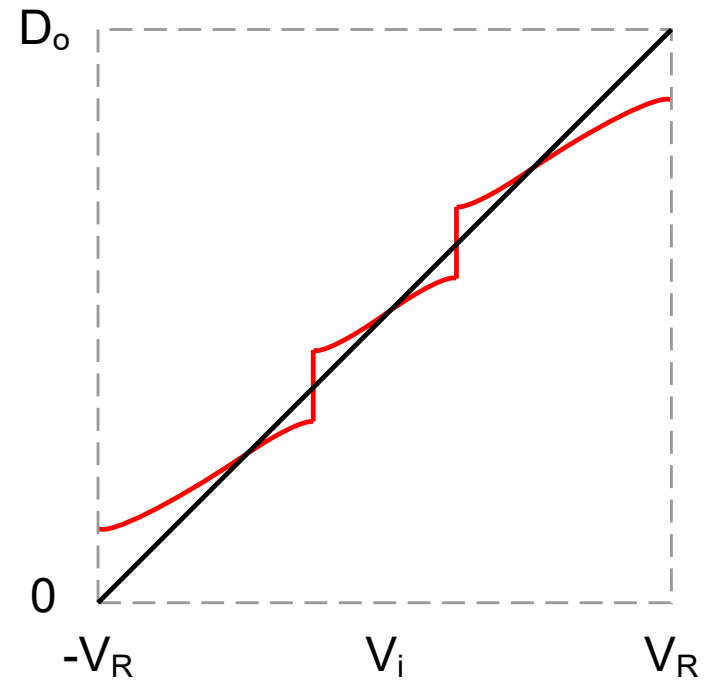


- Simple parametric errors (e.g., capacitor mismatch) can be easily corrected with digital calibration

Residue Errors in Pipelined ADC (1.5-b)

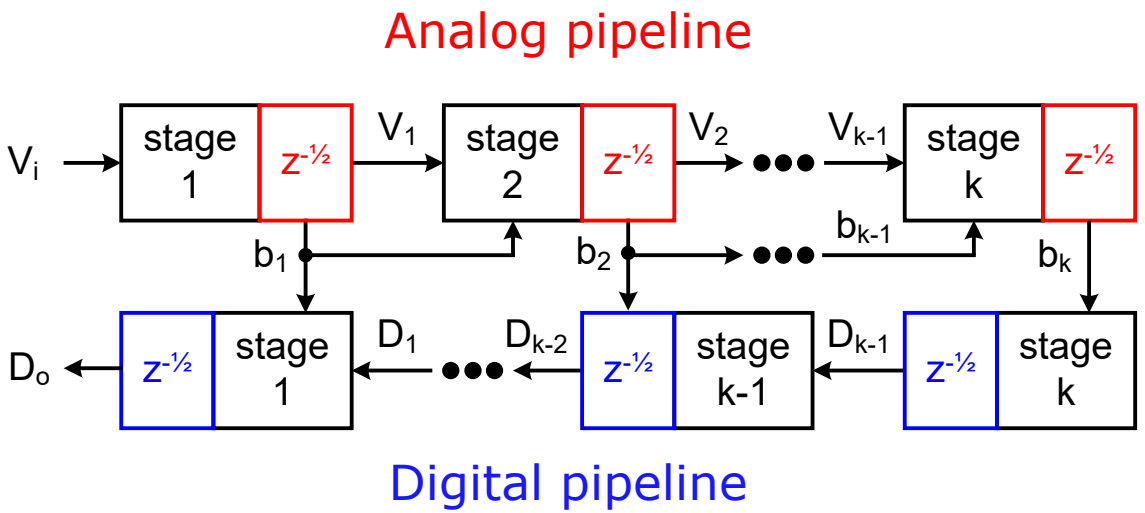






- Static nonlinearities (e.g., INL and DNL), dynamic errors (e.g., RA settling errors [25]), and even memory errors [26] can also be calibrated

Digital Calibration of Pipelined ADC



For stage j, we have

$$V_j = \frac{C_1 + C_2}{C_1 + \frac{C_1 + C_2}{A}} \cdot V_{j-1} + \frac{C_2}{C_1 + \frac{C_1 + C_2}{A}} \cdot (1 - b_j) \cdot V_R$$

$$\frac{V_{j-1}}{V_R} = \underbrace{\left(\frac{C_1 + \frac{C_1 + C_2}{A}}{C_1 + C_2} \right)}_{\alpha} \cdot \frac{V_j}{V_R} + \underbrace{\left(\frac{C_2}{C_1 + C_2} \right)}_{\beta} \cdot (b_j - 1)$$

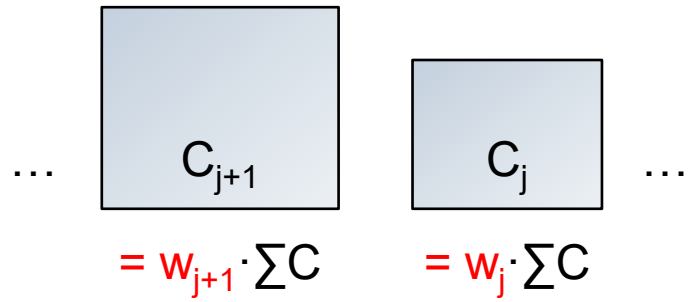
$$D_{j-1} = \alpha \cdot D_j + \beta \cdot (b_j - 1) \quad \leftarrow \text{linear}$$

$$D_{j-1} = \sum_m \alpha_m \cdot D_j^m + \beta \cdot (b_j - 1) \quad \leftarrow \text{nonlinear}$$

- Needs to determine coefficients { α, β } ☜ Ref. [13]
- Needs to determine coefficients { α_m, β } Ref. [21]

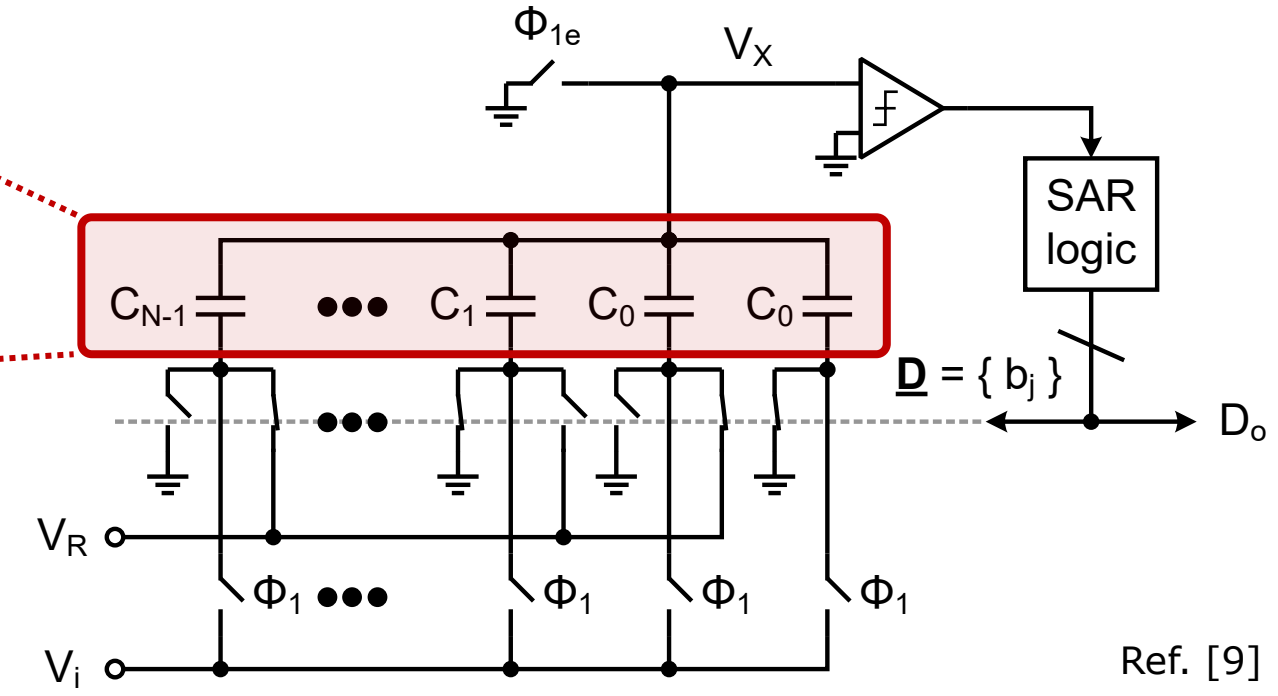
☞ typically, α indicates inter-stage residue gain error while β indicates DAC capacitor mismatch errors

Digital Calibration of SAR ADC



Key cal. eqn.:

$$D_o \approx \sum_{j=0}^{N-1} \left(\frac{C_j}{\sum C_j} \right) \cdot b_j = \sum_{j=0}^{N-1} w_j \cdot b_j$$

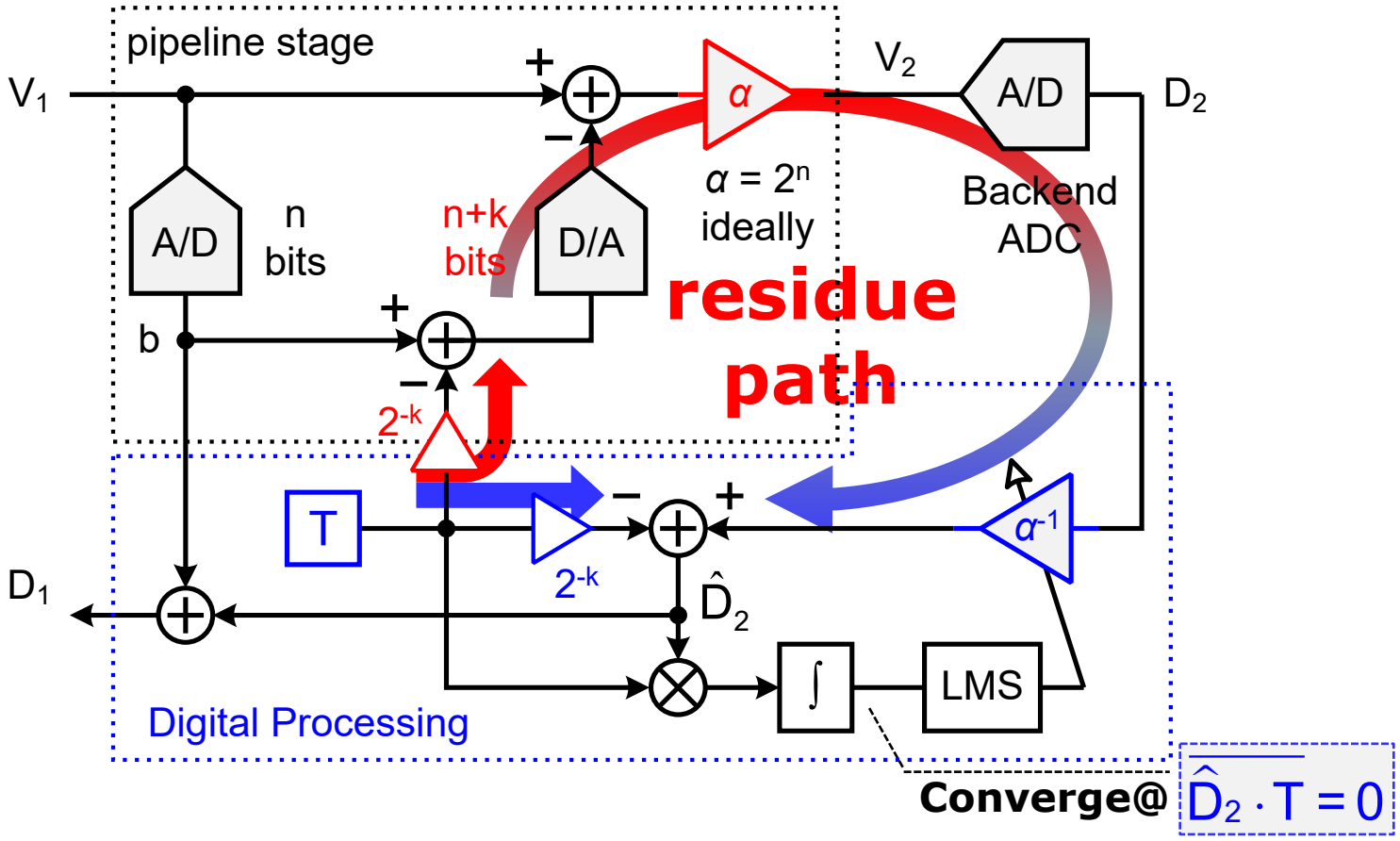


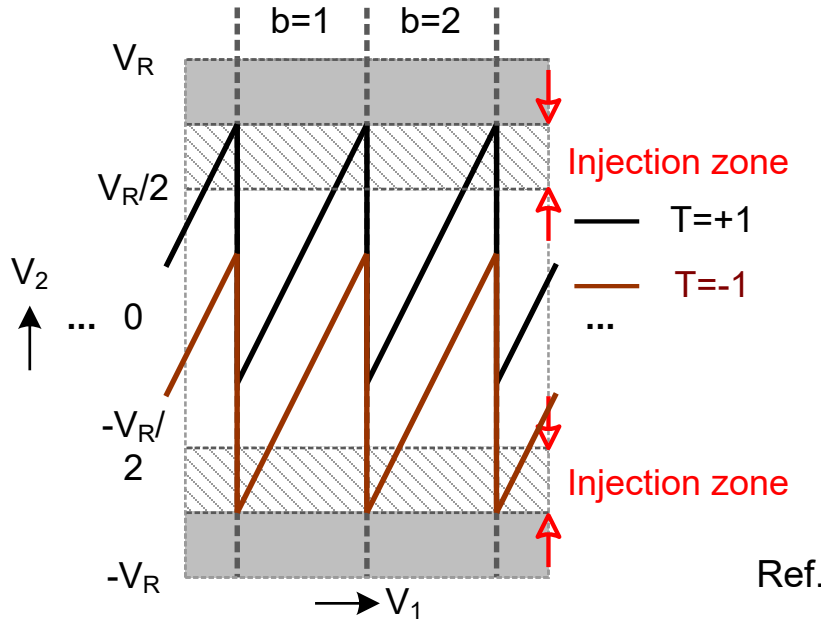
Ref. [9]

- DAC capacitors are subject to random mismatch → need to determine the exact values of bit weights { w_j }

EXPLOITING REDUNDANCY FOR DIGITAL BACKGROUND CAL.

Residue Gain Cal. (DAC Dither)



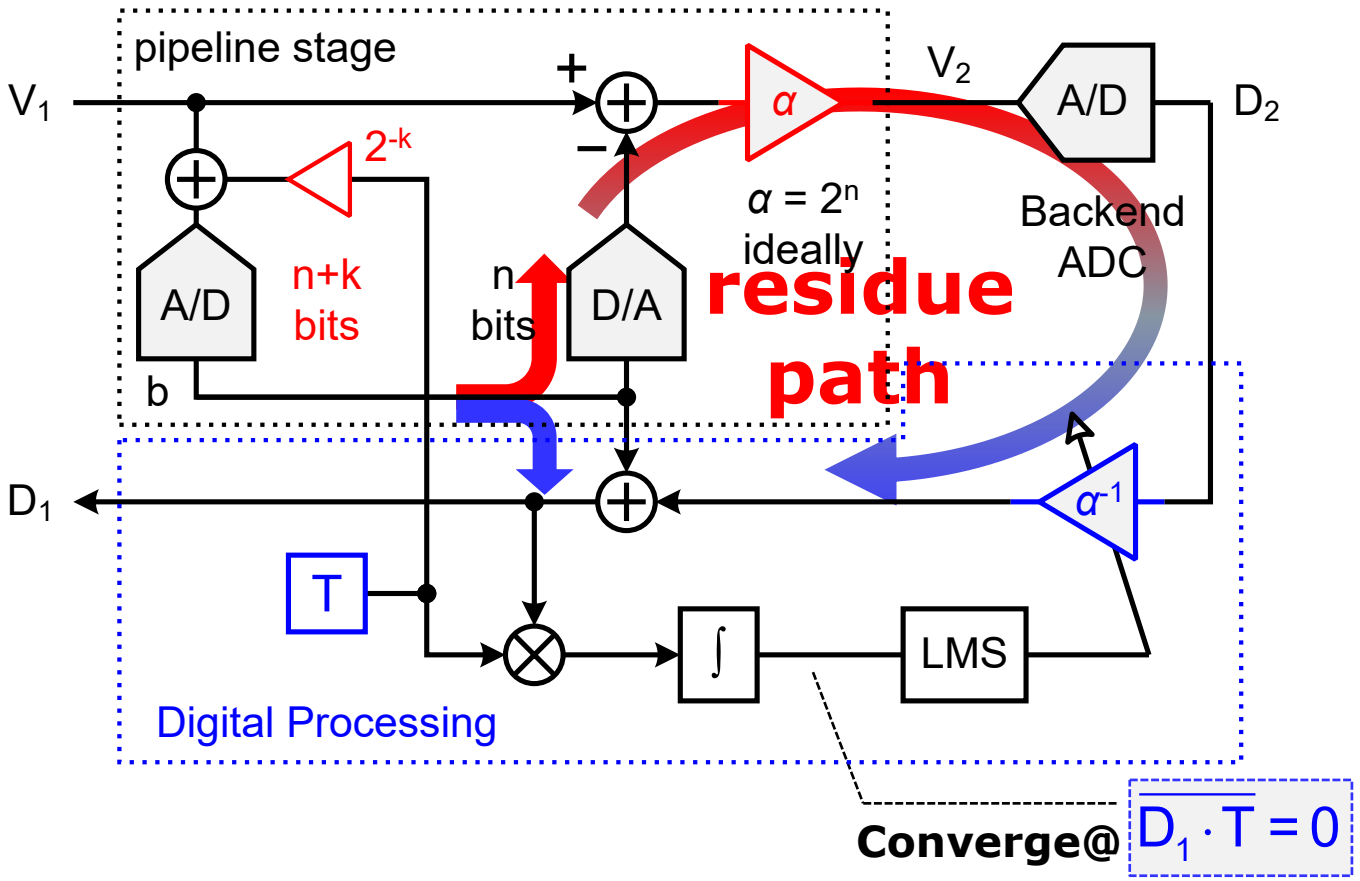


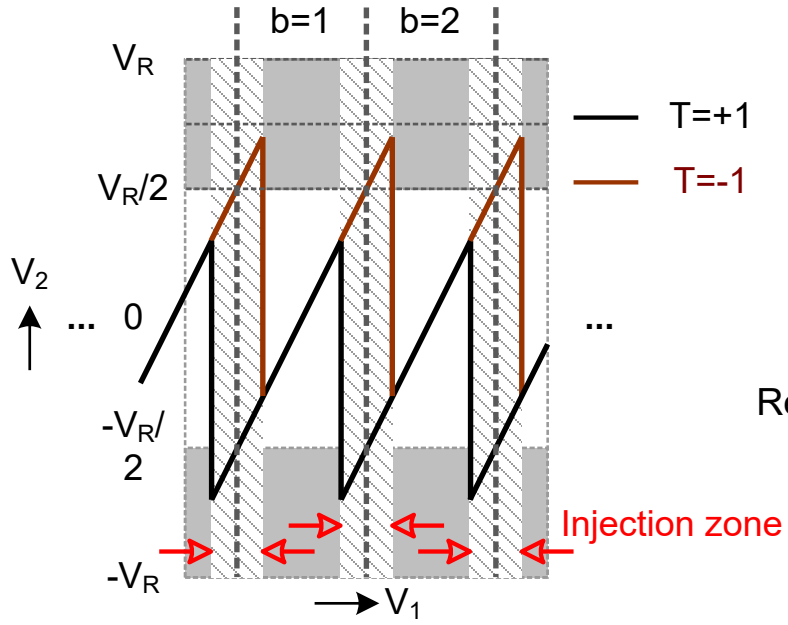
Ref. [18]

- Small dither added to the residue path, then subtracted out in the digital domain (after correction)

- In steady state, analog gain (α) and digital gain inverse (α^{-1}) cancel exactly
- $2^{-k} \leq ¼$ to avoid overflow, DAC adds 2 bits or more to accommodate dither

Residue Gain Cal. – Comparator Dither



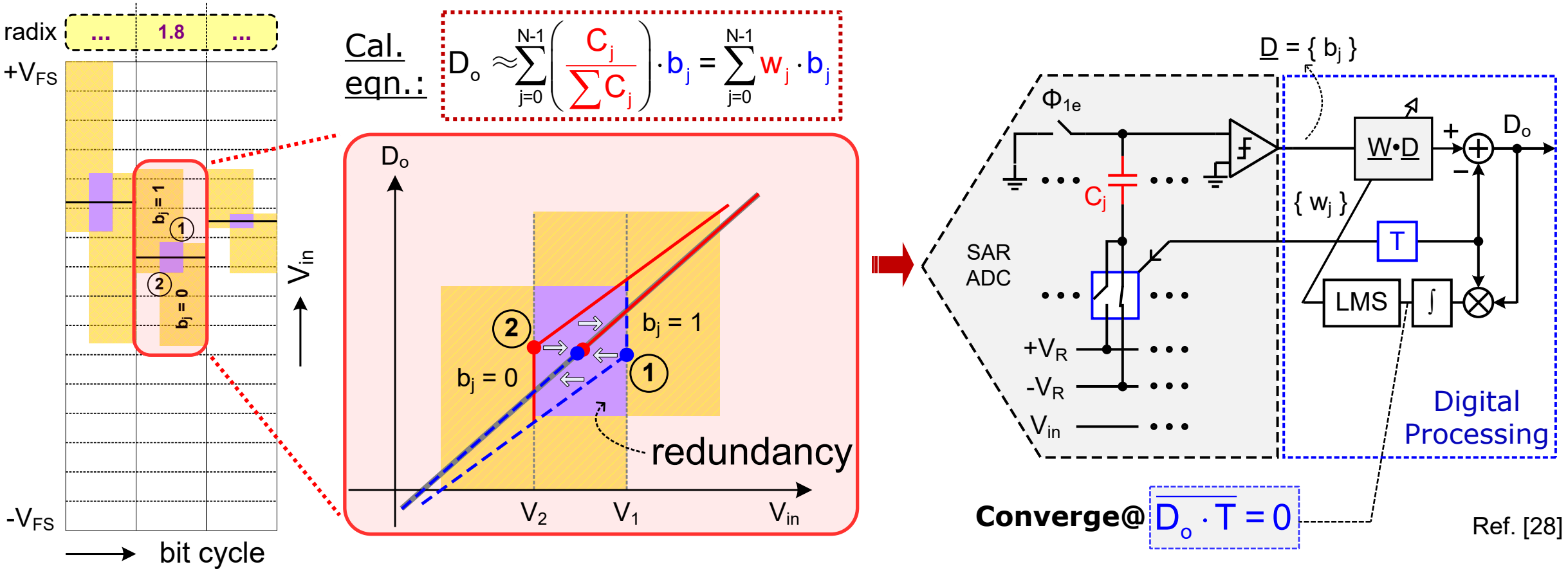


Ref. [15]

- Dither forces ADC traverse TWO redundant conversion paths randomly
- Ideally, D_o is indep. of conversion path

- In steady state, analog gain (α) and digital gain inverse (α^{-1}) cancel exactly
- Arbitrary dither amplitude a.l.a. no overflow occurs

SAR Radix Cal. – Sub-Radix Dither



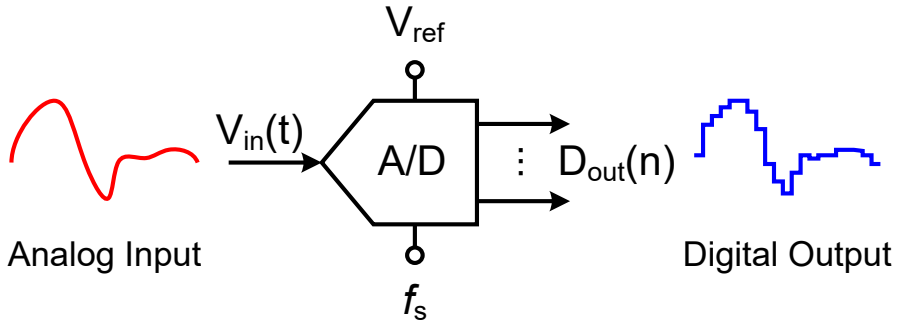
- Dither forces ADC traverse TWO redundant conversion paths randomly
- In steady state, accurate $\{w_j\}$ is identified s.t. D_o is indep. of T

Caveats of Background Cal.

- **Vis-à-vis rule**: what you see is what you get – model coefficients often depend on input statistics!
- **PVT rule**: error model coefficients often depend on P.V.T. variations!
- Complexity, memory, digital power…
- Convergence time (tracking speed)
- After all, solution needs to be practical and cost-efficient…

FIGURE-OF-MERIT (FOM)
WALDEN, SCHREIER AND ALPHA FOMS

Precision-Speed-Power Tradeoffs



- BW: min{f_s/2, ERBW}
- f_s: sample rate (Nyquist ADC)
- ERBW: effective resolution BW (ΔΣ ADC)

Bandwidth

$$\text{RA bandwidth} \sim \omega_u = \frac{g_m}{C} \sim \frac{1}{C}\frac{I_d}{V_{ov}}$$

Noise

$$\text{Thermal noise } \sigma_N \sim \sqrt{\frac{kT}{C}}$$

Matching

$$\text{Mismatch } \sigma_{MM} \sim \sqrt{\frac{1}{\text{Area}}} \sim \sqrt{\frac{1}{C}}$$

- ADC Power ~ Bandwidth or Sample Rate
- **A**. ADC Power (and Area) ~ Precision^2 (4^{ENOB}) for noise- or matching-limited design
  B. ADC Power (and Area) ~ Precision (2^{ENOB}) o.w.☜

☞ For example, a low-resolution flash ADC's power and area ~ its # of conversion steps = 2^N ($\geq 2^{ENOB}$)

Walden vs. Schreier FoM

Walden FoM

$$\mathrm{FoM_W} = \frac{\mathrm{P}}{f_s \cdot 2^{\mathrm{ENOB}}} \left[\frac{\mathrm{Joule}}{\mathrm{Conversion\,Step}}\right]$$

Ref. [29]

Schreier FoM

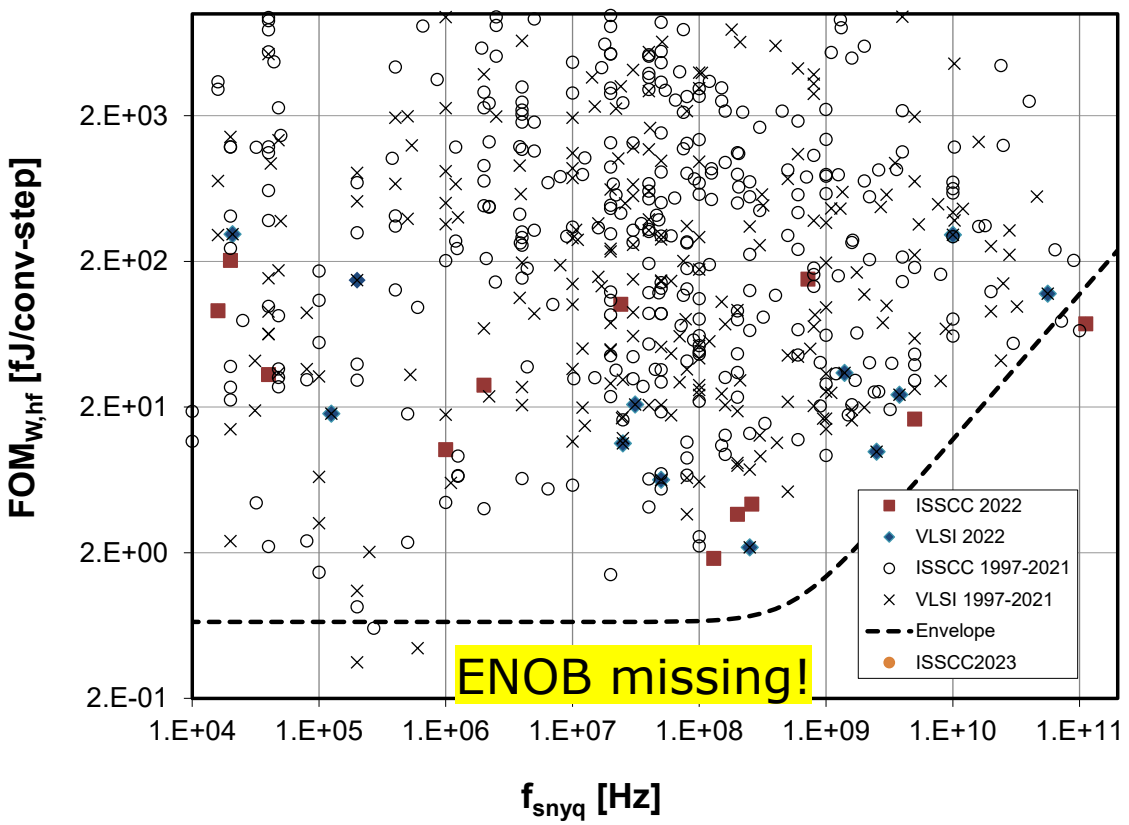
$$\mathrm{FoM_S} = \mathrm{SNDR} + 10\log_{10}\left(\frac{\mathrm{BW}}{\mathrm{P}}\right) \mathrm{[dB]}$$

Ref. [30]

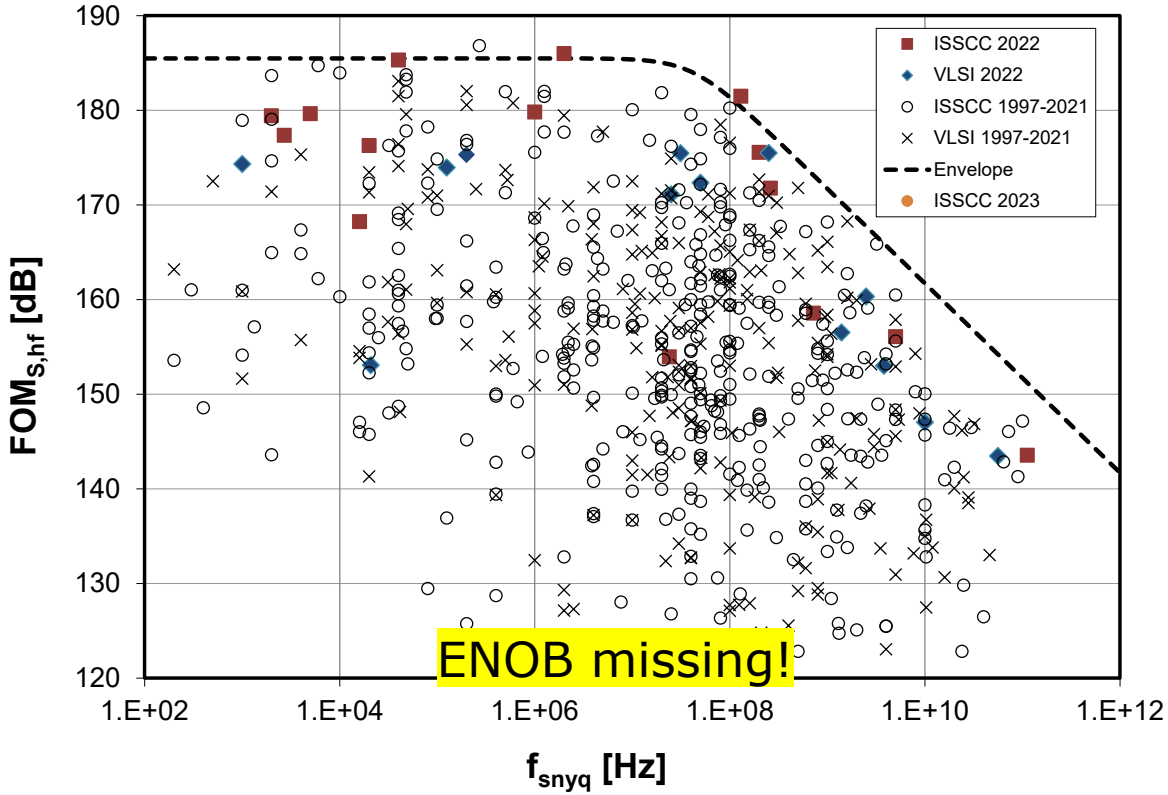
- Schreier FoM:
  - Implicitly assumes Noise- or Matching-limited ADC of **Scenario A**
  - The dimension of $\mathrm{FoM_S}$ is dB → Schreier FoM naturally measures SNDR, or Performance, not just # of effective quantization steps
- Walden FoM:
  - Implicitly assumes Nyquist-rate ADC of **Scenario B**
  - The dimension of $\mathrm{FoM_W}$ is Joule/Conversion-Step → Walden FoM naturally measures Energy Efficiency

Walden vs. Schreier FoM

Walden FoM



Schreier FoM



Ref. [31]

- A complete FoM vs. Performance rendering requires a 3D plot

But Actually…

$$\mathrm{FoM_W} = \frac{P}{f_s \cdot 2^{ENOB}} = \frac{P}{2 \cdot BW \cdot 2^{ENOB}}$$

Common Part: $\frac{P}{BW \cdot \alpha^{ENOB}}$ where $\alpha = 2 - 4$

$$\mathrm{FoM_S} = SNDR - 1.76 + 10\log_{10}\left(\frac{BW}{P}\right) + 1.76$$
$$= 6.02 \cdot ENOB + 10\log_{10}\left(\frac{BW}{P}\right) + 1.76$$
$$= 10\log_{10}\left(4^{ENOB}\right) + 10\log_{10}\left(\frac{BW}{P}\right) + 1.76$$
$$= 10\log_{10}\left(\frac{BW \cdot 4^{ENOB}}{P}\right) + 1.76$$

$\alpha = 4 \rightarrow$ SNDR (scenario A)

$\alpha = 2 \rightarrow$ design complexity (scenario B)

- The common part between the two seemingly distinct FoMs is striking!

Alpha FoM

Define **"α" efficiency**

$$\mathbf{FoM_{\alpha}} = \frac{P}{BW \cdot \alpha^{ENOB}} \quad \text{where } \alpha = 2 - 4$$

- $\alpha = 2 \rightarrow$ Walden FoM
- $\alpha = 4 \rightarrow$ Schreier FoM

Also define **"α" performance**

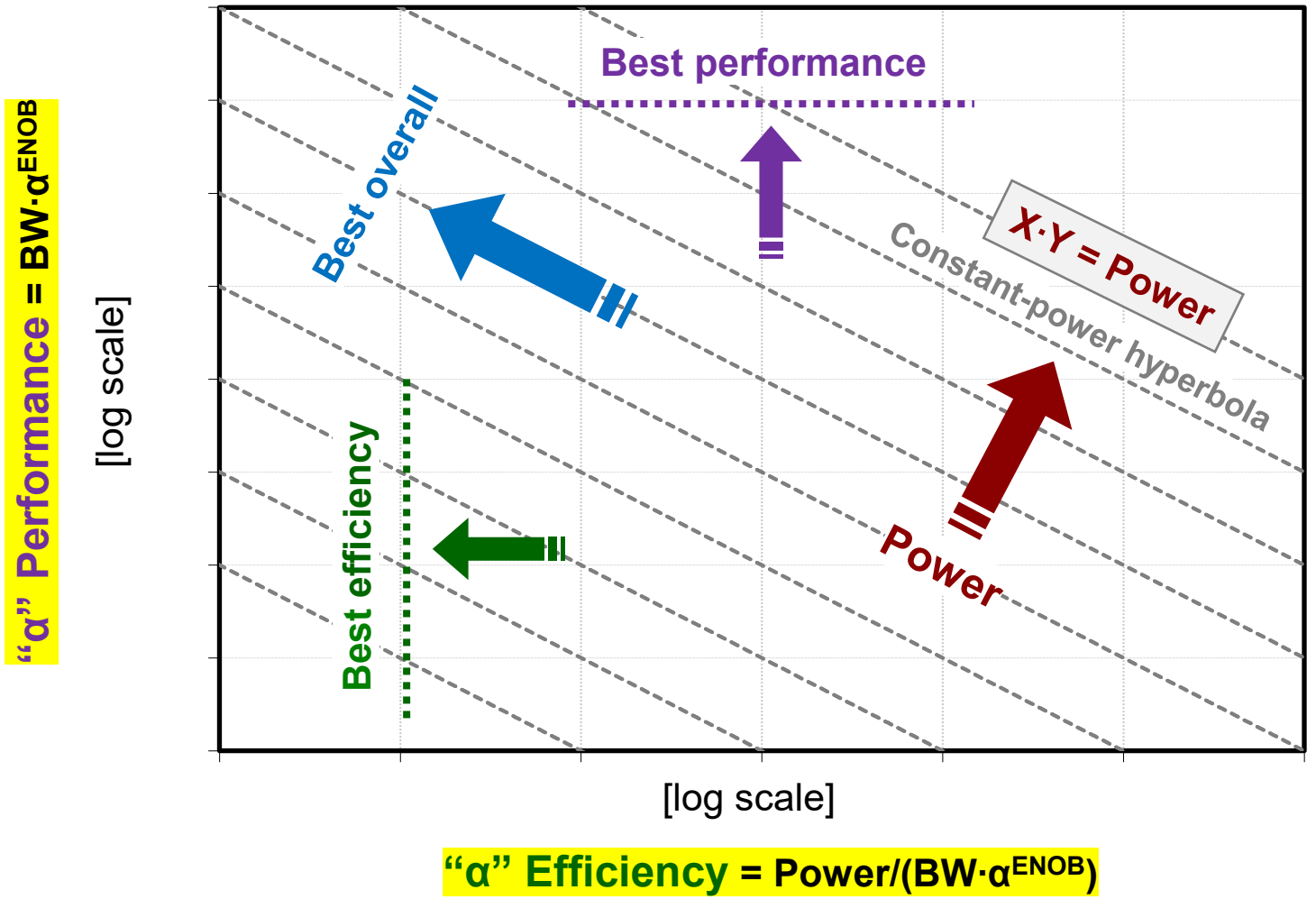
$$\mathbf{Performance} = BW \cdot \alpha^{ENOB} \quad [Hz \cdot Precision]$$

- α^{ENOB} indicates "Precision"

$$\mathbf{Efficiency} \times \mathbf{Performance} = \frac{P}{BW \cdot \alpha^{ENOB}} \times BW \cdot \alpha^{ENOB} = \mathbf{Power}$$

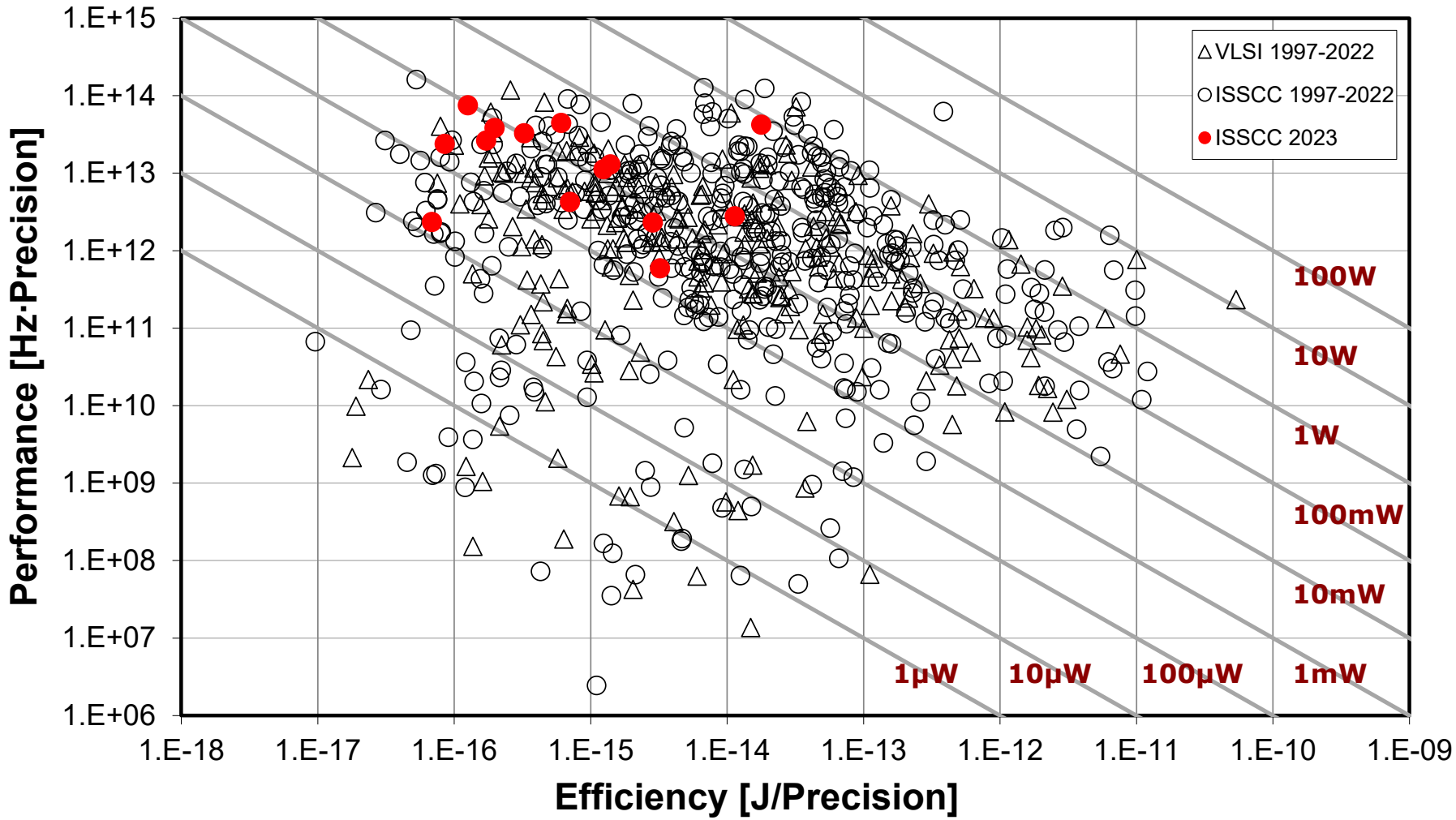
- Performance vs. Efficiency plot gives more info when comparing designs
- Given Power, Performance and Efficiency trade hyperbolically - **Alpha law**

Performance vs. Efficiency (PE) Chart



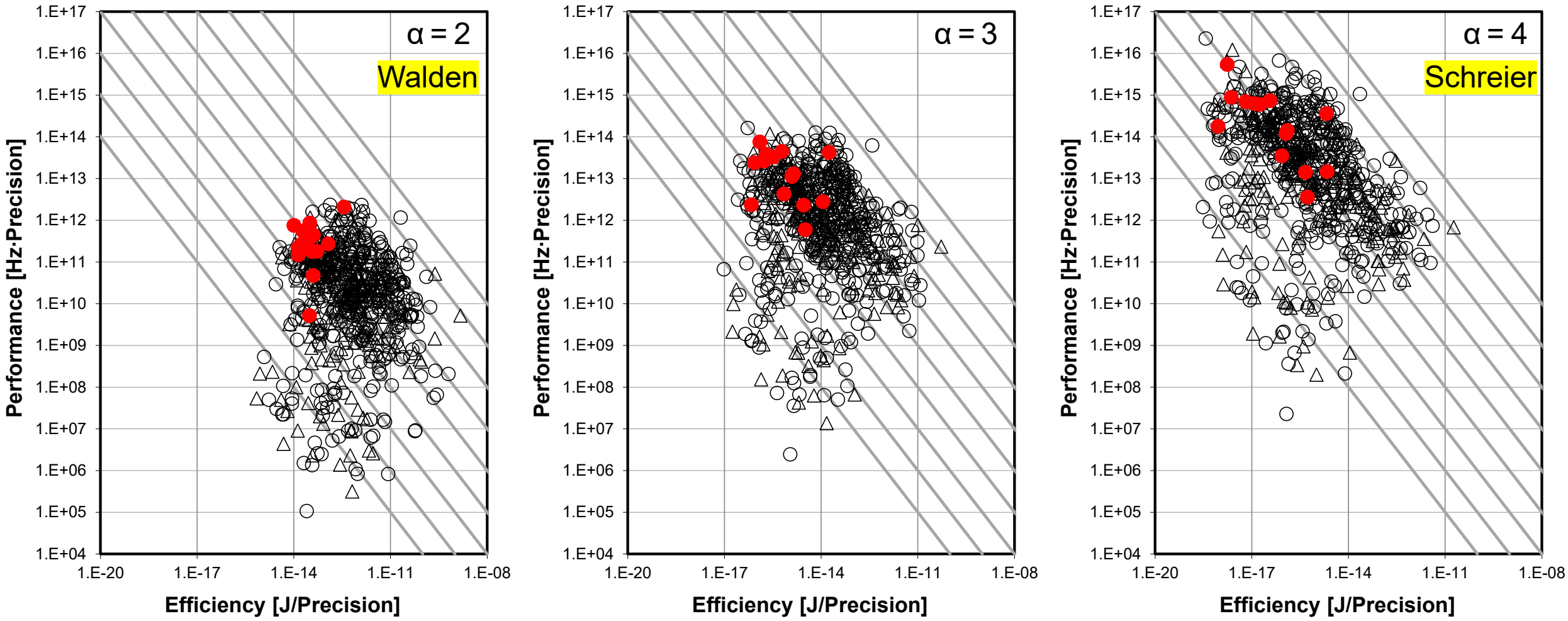
$\alpha = 2 - 4$

ADC PE Chart (1997-2023)

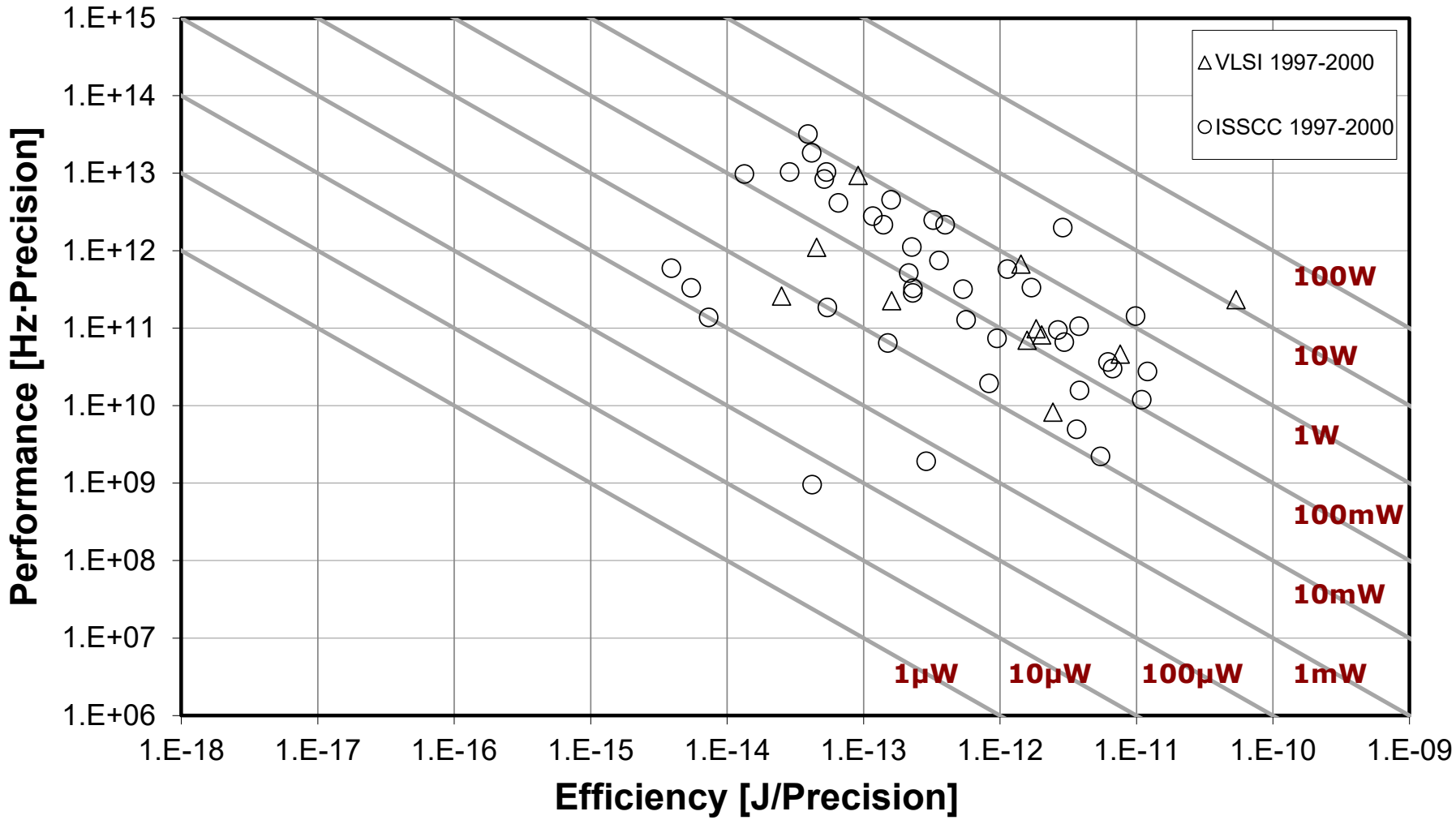


$\alpha = 3$

ADC PE Chart: α=2–4

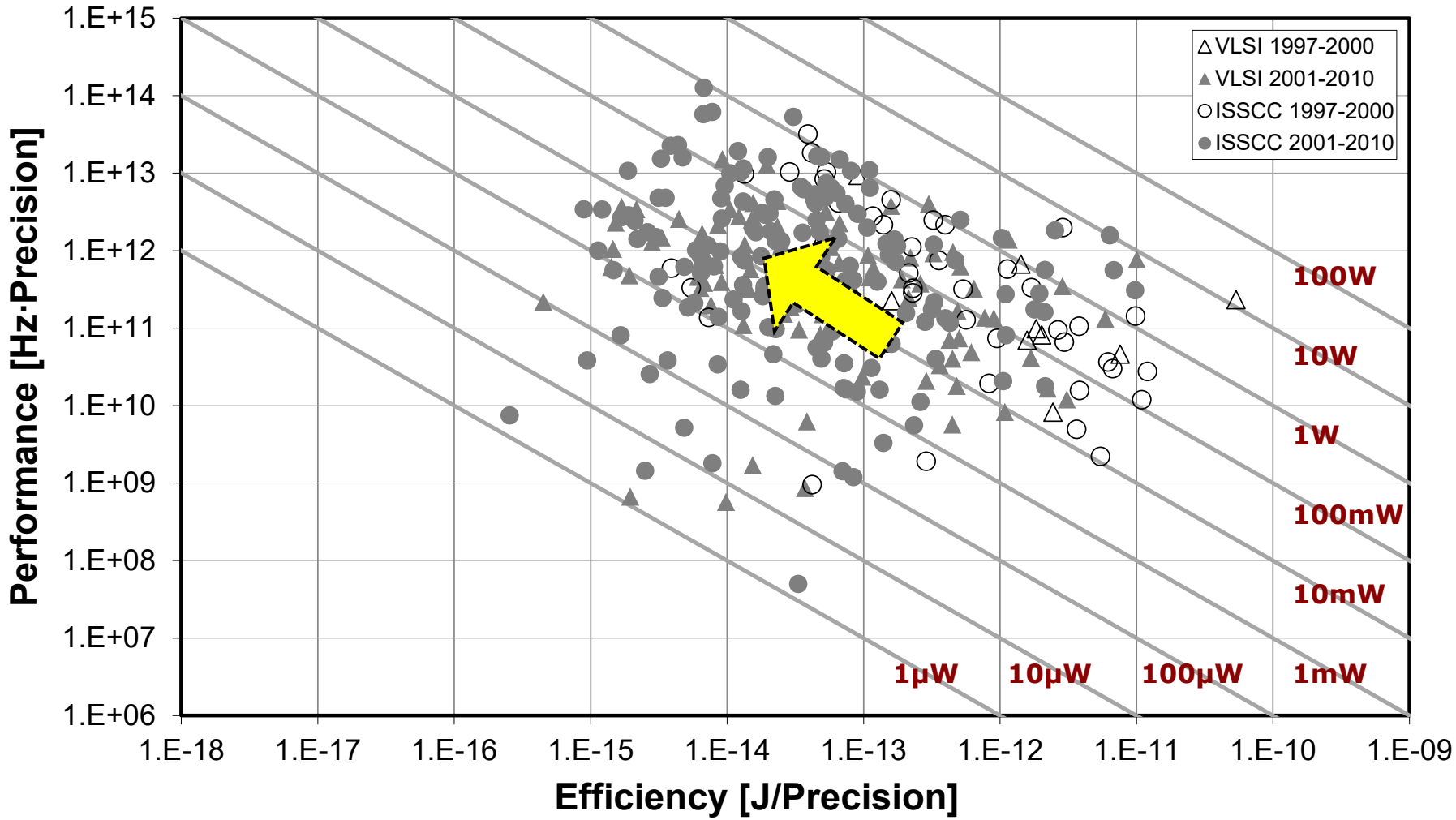


ADC PE Chart (1997-2000)



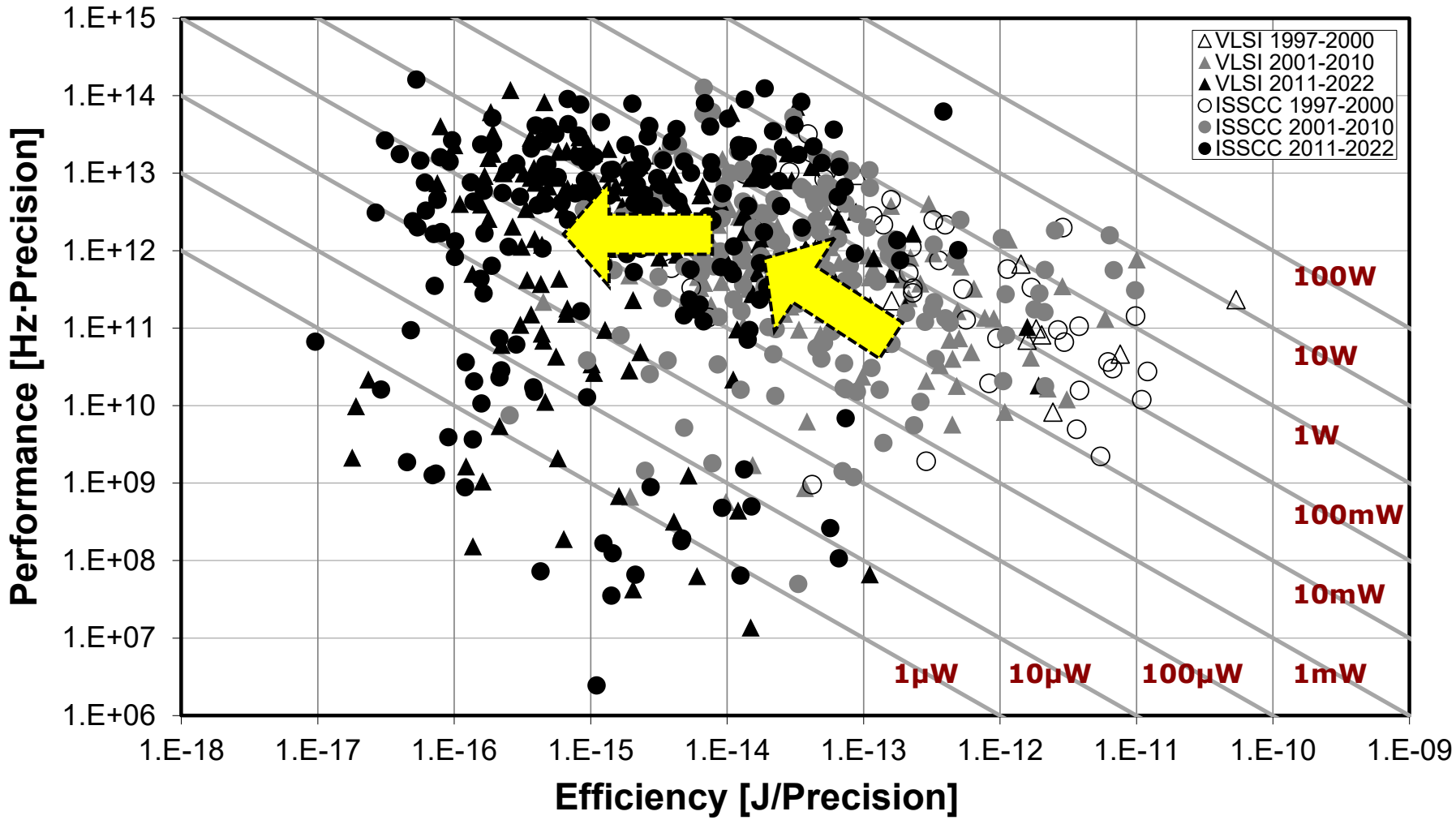
$\alpha = 3$

ADC PE Chart (1997-2010)



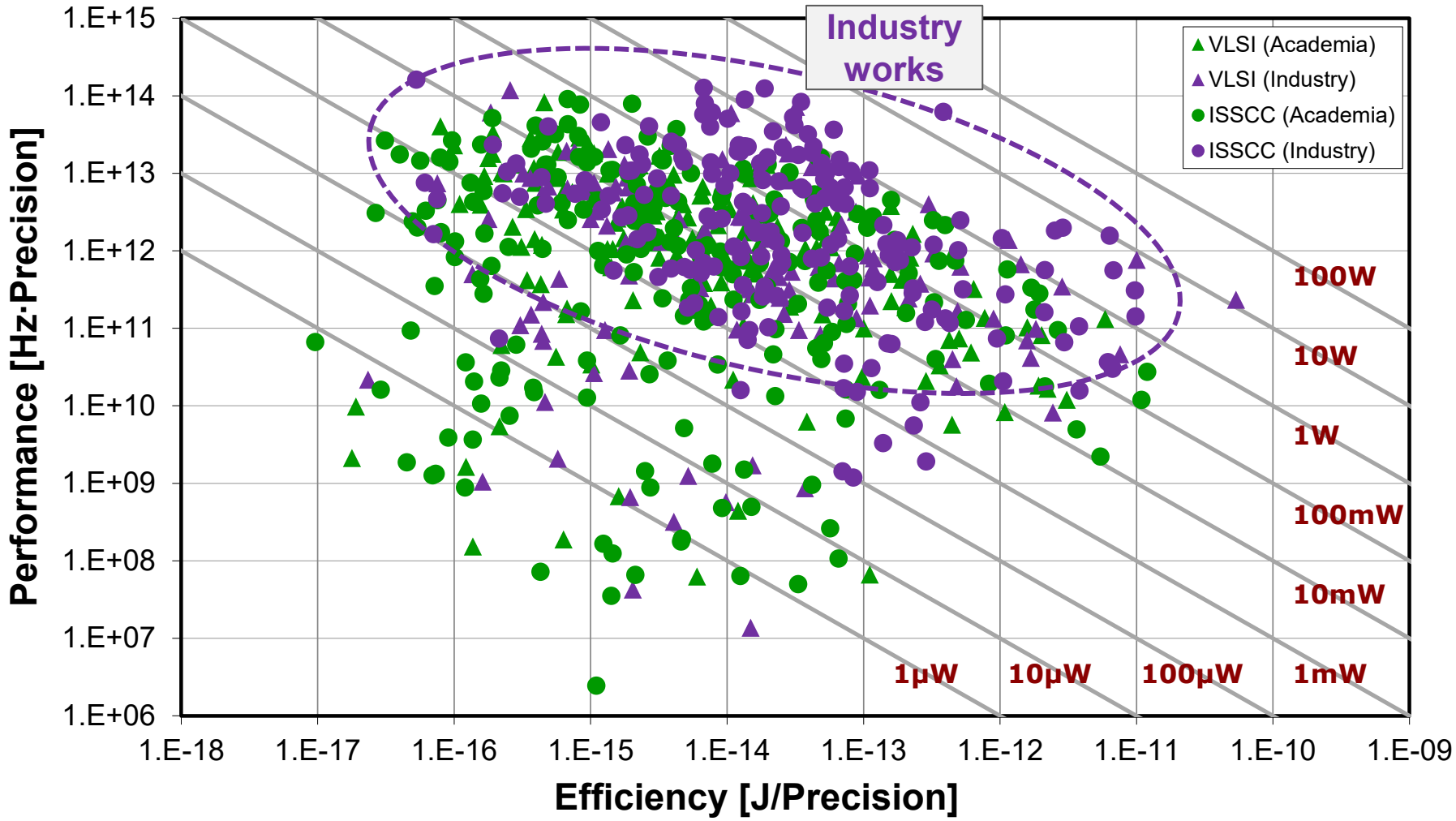
$\alpha = 3$

ADC PE Chart (1997-2022)



$\alpha = 3$

ADC PE Chart (1997-2022)



$\alpha = 3$

Converter Papers to See This Year

- **Session 10 - Pipelined and Noise-Shaping ADCs:**
  - 10.1, 10.2 – high-speed pipelined ADCs
  - 10.4 – high-resolution two-step SAR ADC with predictive level-shifting
- **Session 17 - High-Speed Data Converter:**
  - 17.3 –high-speed direct-RF DAC
  - 17.4, 17.6 – digitally calibrated time-interleaved SAR ADCs
  - 17.8 – high-speed time-domain ADC

References

1. W.R. Bennett, "Spectra of quantized signals," *Bell Syst. Tech. J.*, pp. 446-472, Jul. 1948.
2. Oppenheim and Schafer, *Discrete-Time Signal Processing*, 2nd ed., Prentice Hall, 1998.
3. D. Gruber et al., "A 12b 16GS/s RF-sampling capacitive DAC for multi-band soft-radio base-station applications with on-chip transmission-line matching network in 16nm FinFET," in *ISSCC*, 2021.
4. B. Razavi, *Principles of Data Conversion System Design*, IEEE press, 1995.
5. M. Shinagawa, Y. Akazawa, and T. Wakimoto, "Jitter analysis of high-speed sampling systems," *JSSC*, no 1, 1990.
6. A.M. Abo and P. R. Gray, "A 1.5-V, 10-bit, 14.3-MS/s CMOS pipeline ADC," *JSSC*, Mar. 1999.
7. S.H. Lewis et al., "A 10-b 20-MS/s analog-to-digital converter," *JSSC*, Mar. 1992.
8. F. Kuttner, "A 1.2-V 10-b 20-MS/s nonbinary successive approximation ADC in 0.13-μm CMOS," in *ISSCC*, 2002.
9. W. Liu et al., "A 600MS/s 30mW 0.13μm CMOS ADC array achieving over 60dB SFDR with adaptive digital equalization," in *ISSCC*, 2009.
10. C.-C. Liu et al., "A 10b 100MS/s 1.13mW SAR ADC with binary-scaled error compensation," in *ISSCC*, 2010.
11. H.-S. Lee, D.A. Hodges, and P.R. Gray, "A self-calibrating 15 bit CMOS A/D converter," *JSSC*, no. 6, 1984.
12. S.-U. Kwak, B.-S. Song, and K. Bacrania, "A 15-b 5-Msamples/s low spurious CMOS ADC," *JSSC*, Dec. 1997.
13. D. Fu et al., "A digital back-ground calibration technique for time-interleaved analog-to-digital converters," in *ISSCC*, 1998.
14. P. Kiss et al., "Adaptive digital correction of analog errors in MASH ADC's—Part II: Correction using test-signal injection," *TCAS-II*, Jul. 2000.

References

15. J. Li and U.-K. Moon, "Background calibration techniques for multistage pipelined ADC's with digital redundancy," *TCAS-II*, Sept. 2003.
16. X. Wang et al., "A 12-bit 20-MS/s pipelined ADC with nested digital background calibration," in *CICC*, 2003.
17. Y. Chiu et al., "Least mean square adaptive digital background calibration of pipelined analog-to-digital converters," *TCAS-I*, Jan. 2004.
18. E.J. Siragusa and I. Galton, "A digitally enhanced 1.8V 15b 40MS/s CMOS pipelined ADC," in *ISSCC*, 2004.
19. J. McNeill et al., "Split ADC architecture for deterministic digital background calibration of a 16b 1MS/s ADC," in *ISSCC*, 2005.
20. Y.-S. Shu and B.-S. Song, "A 15b linear, 20MS/s, 1.5b/stage pipelined ADC digitally calibrated with signal-dependent dithering," in *VLSI*, 2006.
21. B. Murmann and B. Boser, "A 12b 75MS/s pipelined ADC using open-loop residue amplification," in *ISSCC*, 2003.
22. C. Tsang et al., "Background ADC calibration in digital domain," in *CICC*, 2008.
23. A. Panigada and I. Galton, "A 130mW 100MS/s pipelined ADC with 69dB SNDR enabled by digital harmonic distortion correction," in *ISSCC*, 2009.
24. B.D. Sahoo and B. Razavi, "A 12-bit 200-MHz CMOS ADC," *JSSC*, Sept. 2009.
25. E. Iroaga and B. Murmann, "A 12 b, 75 MS/s pipelined ADC using incomplete settling," in *VLSI*, 2006.
26. J.P. Keane et al., "Digital Background Calibration for Memory Effects in Pipelined Analog-to-Digital Converters", *TCAS-I*, Mar. 2006.

References

27. A. Ali et al., A 12-b 18-GS/s RF sampling ADC with an integrated wideband track-and-hold amplifier and background calibration," in *ISSCC*, 2020.
28. W. Liu, P. Huang, and Y. Chiu, "A 12-bit 50-MS/s 3.3-mW SAR ADC with background digital calibration," in *CICC*, 2012.
29. R.H. Walden, "Analog-to-digital converter survey and analysis," *IEEE J. Select. Areas Comm.*, Apr. 1999.
30. R. Schreier and G.C. Temes, *Understanding Delta-Sigma Data Converters*, Wiley, 2005.
31. B. Murmann, "ADC Performance Survey 1997-2022," [Online]: www.stanford.edu/~murmann/adcsurvey.html.