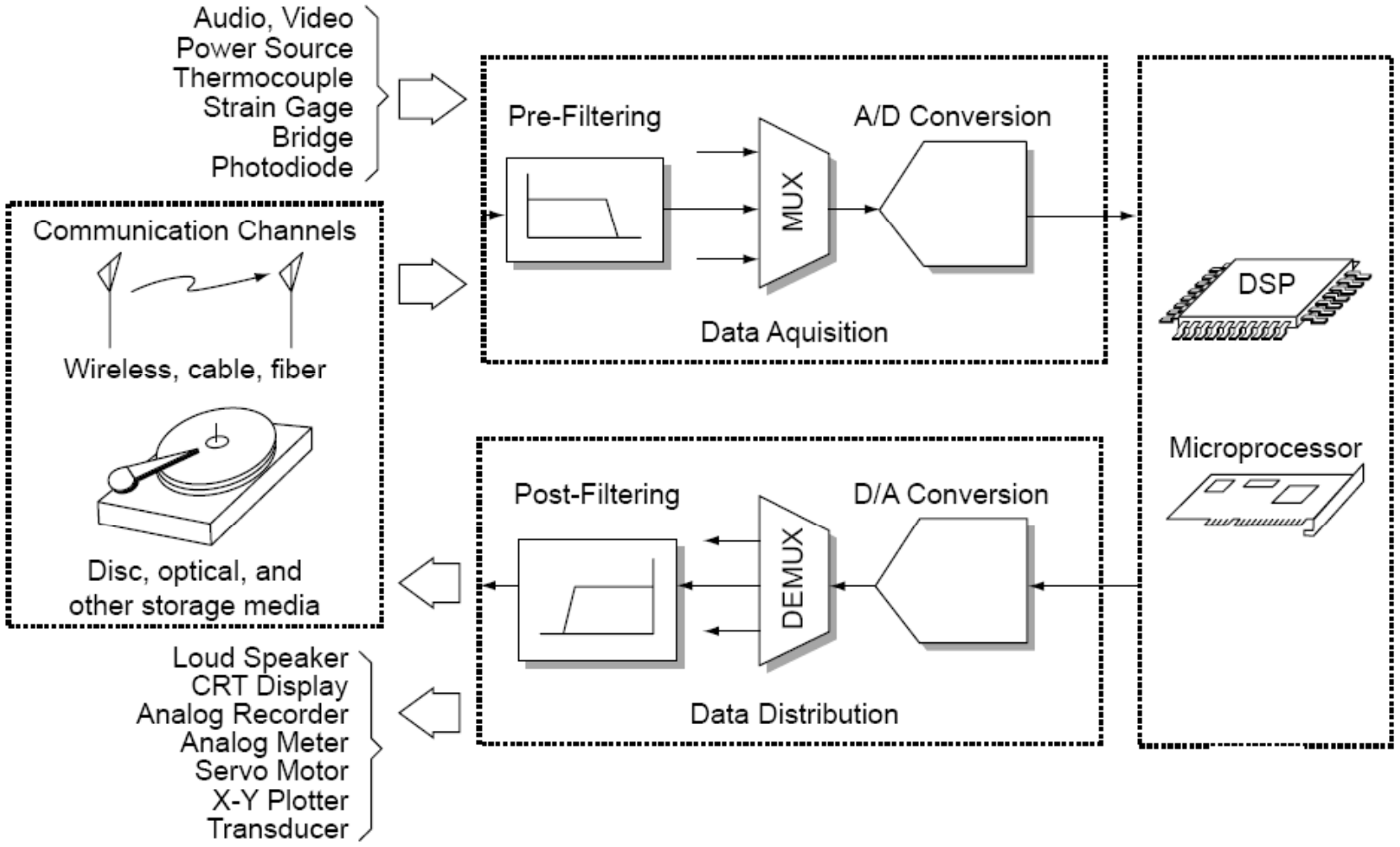
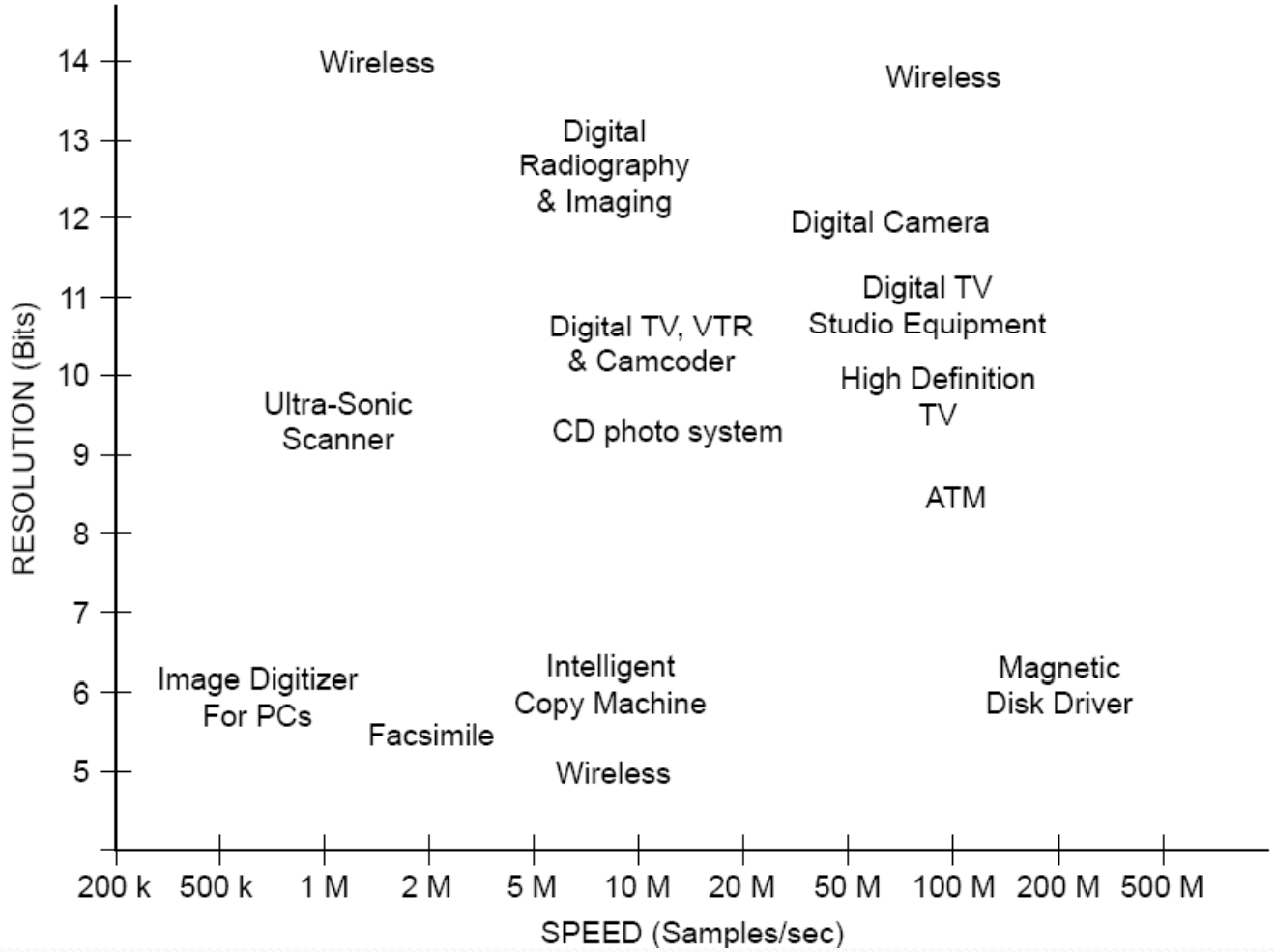
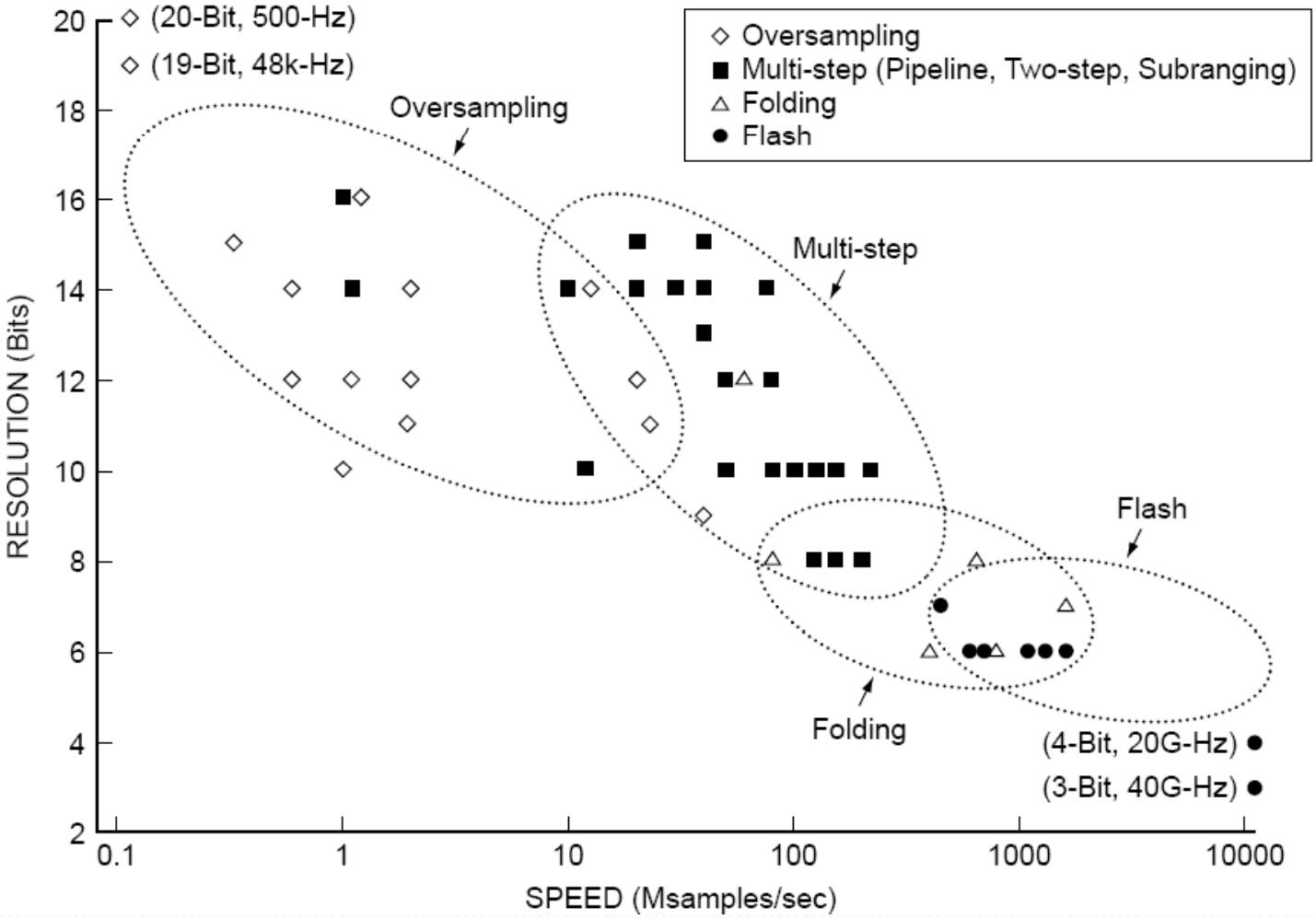
# A/D and D/A Converters



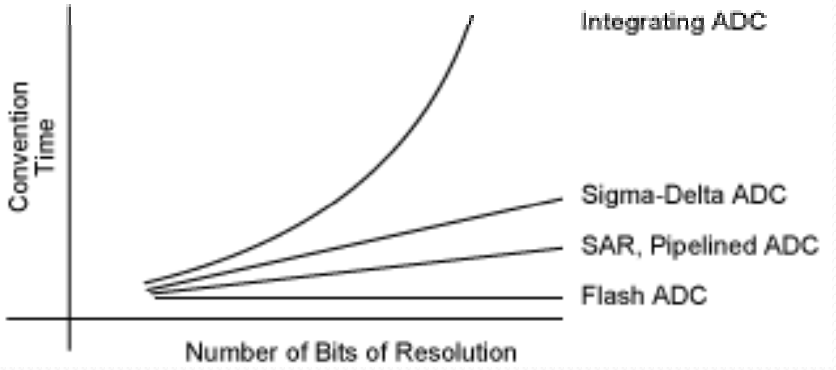


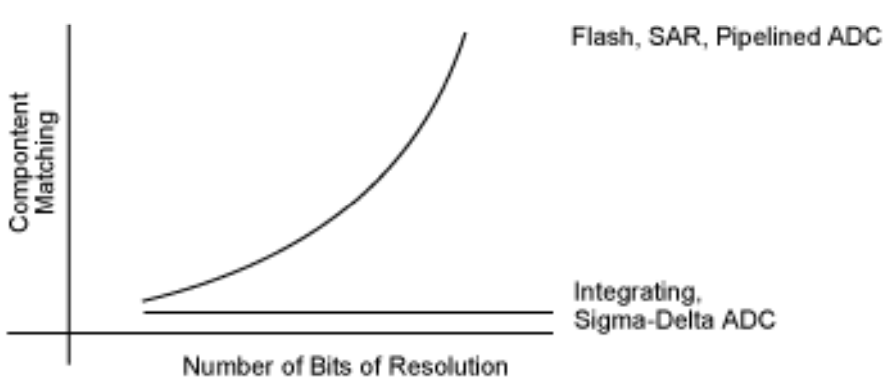

High-speed ADC applications

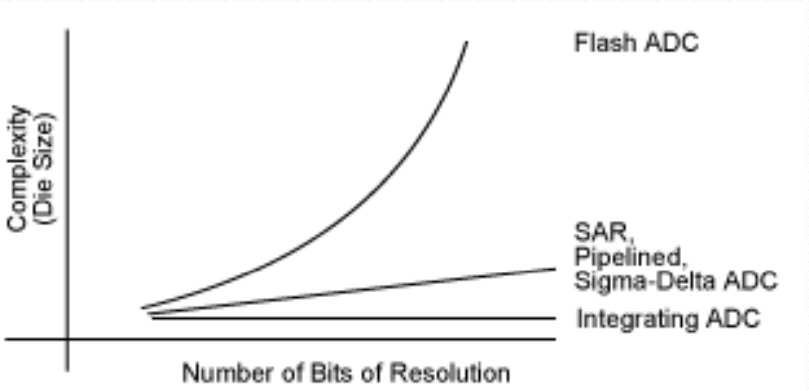


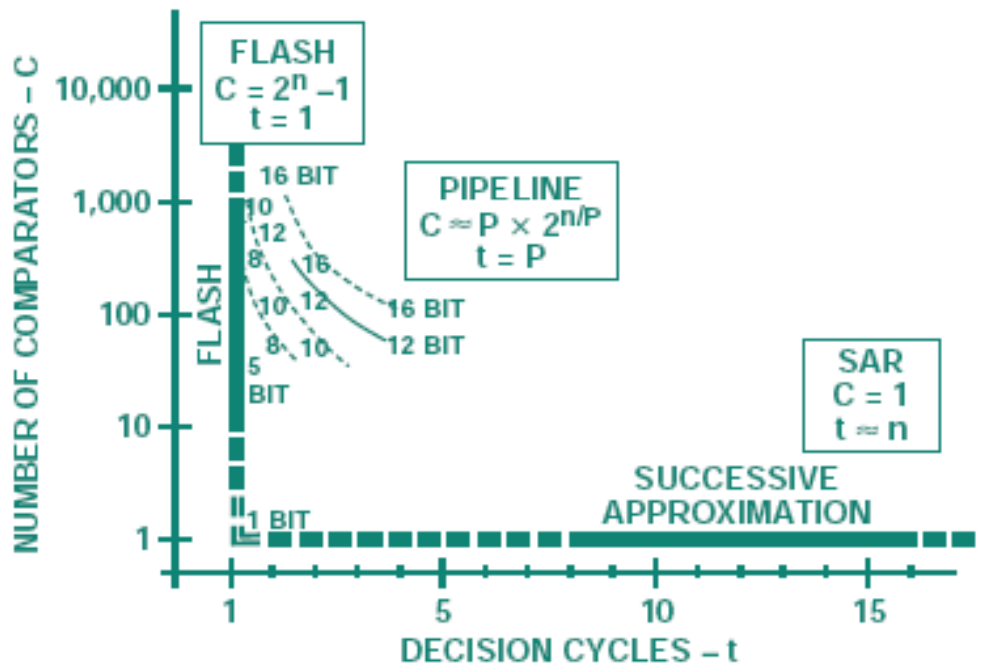

Performance of recently published ADCs: resolution versus speed.

# ADC architectures trade-off



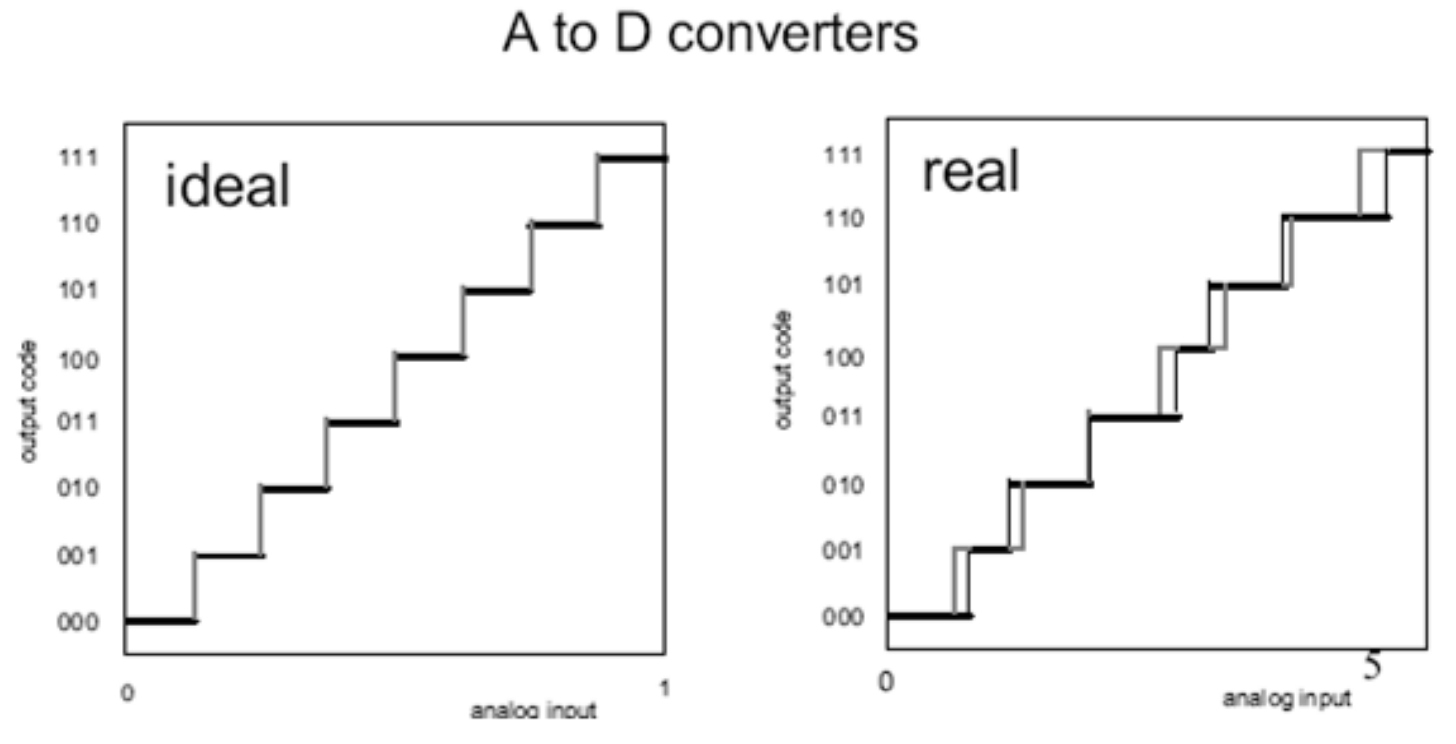



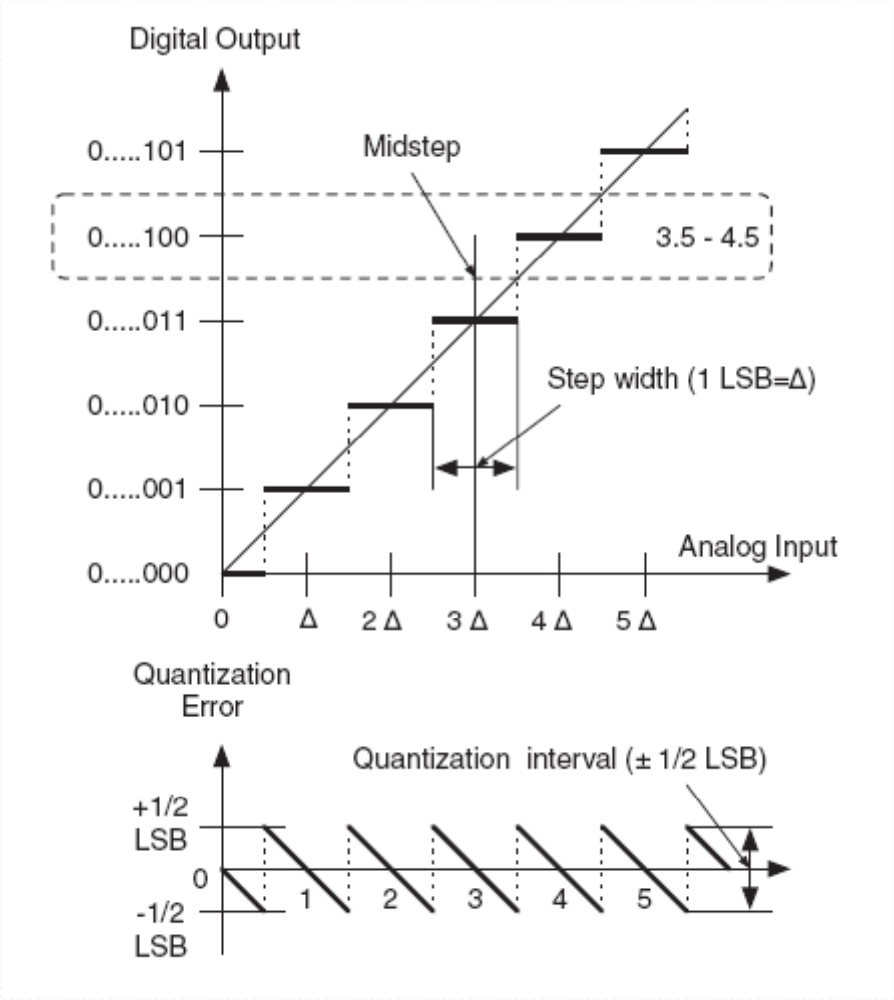



Tradeoff between decision cycles and comparators

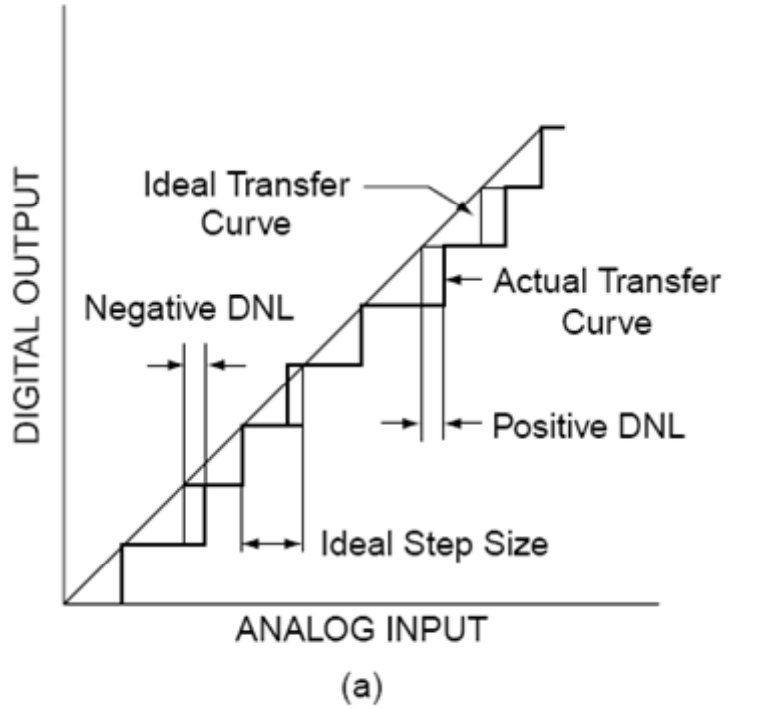
## Static transfer characteristic



# A/D Converter (Nyquist Rate)

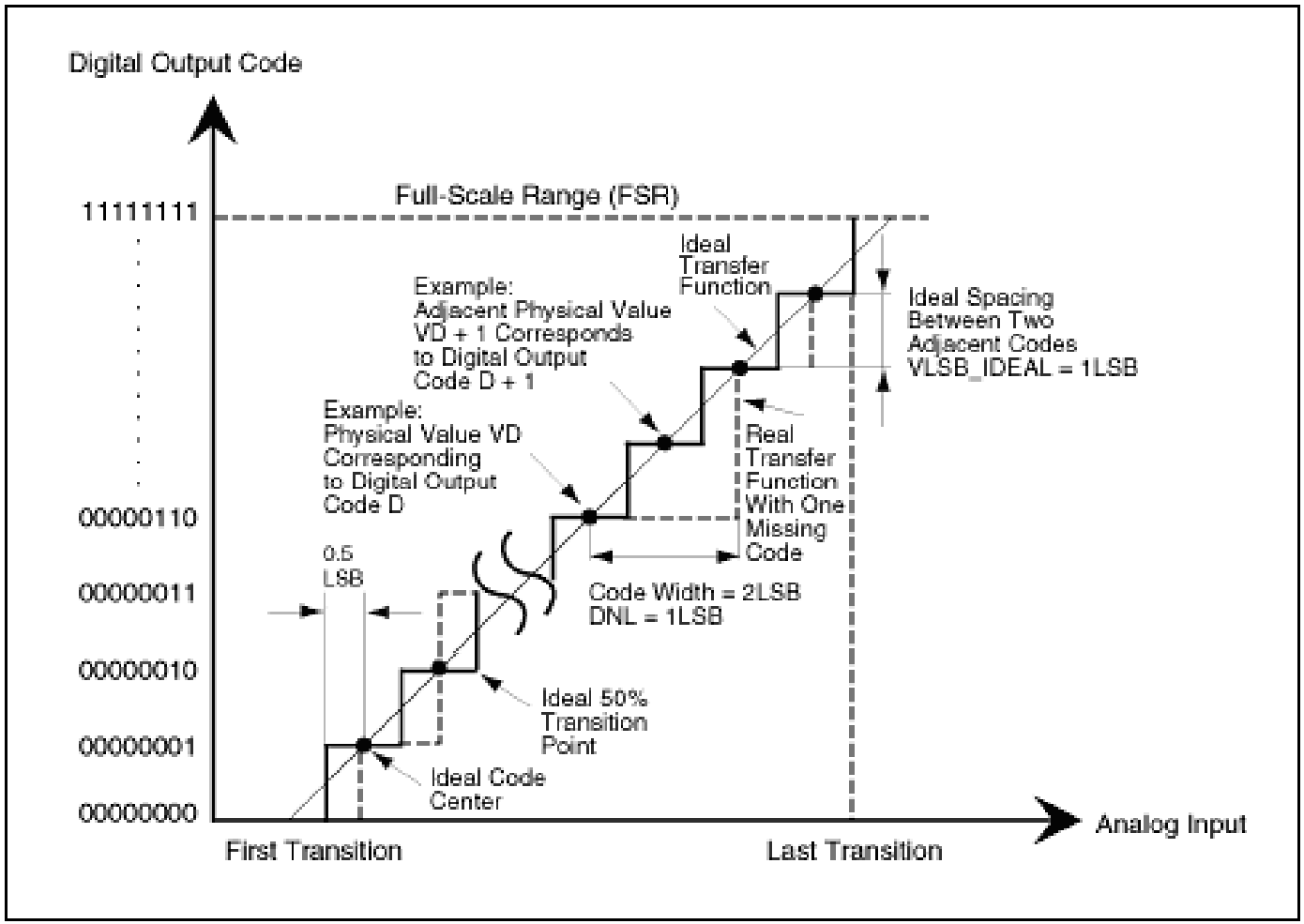


# A/D Nonlinearities

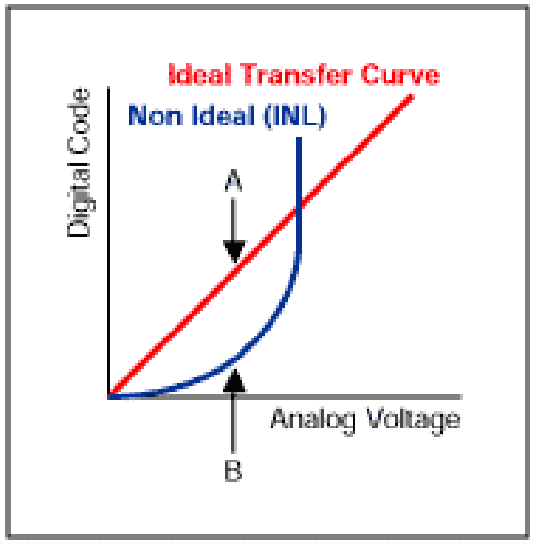


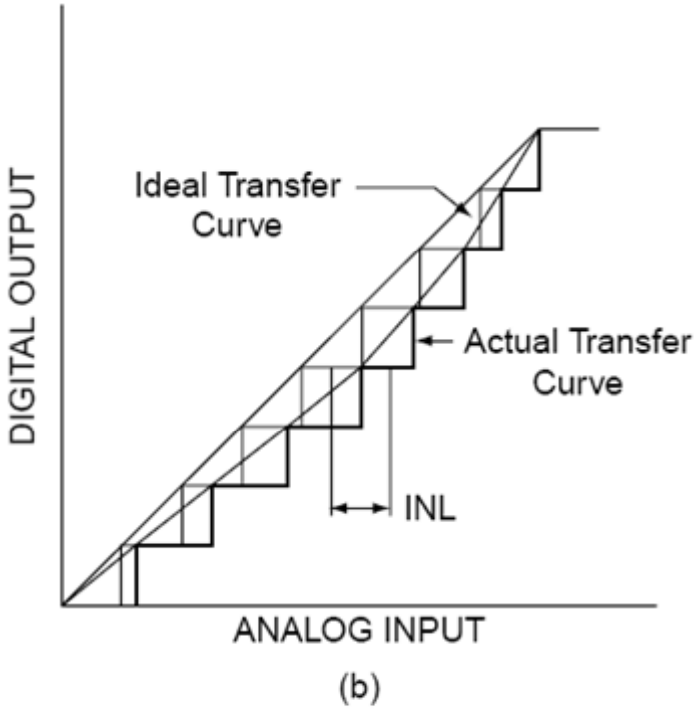

Definition of ADC nonlinearities: (a) DNL

• DNL error is defined as the difference between an actual step width and the ideal value of 1LSB




• to guarantee no missing codes + monotonic transfer function, DNL should be less than 1LSB (+/- 0.5LSB)



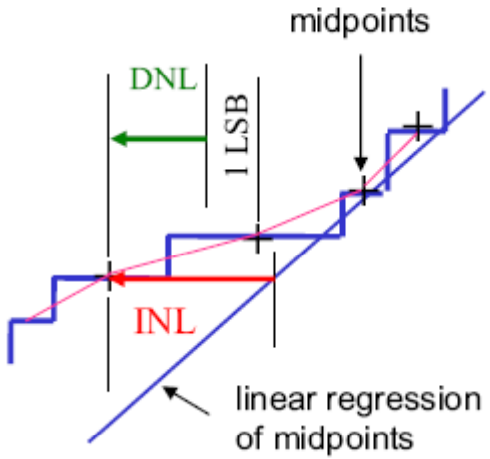



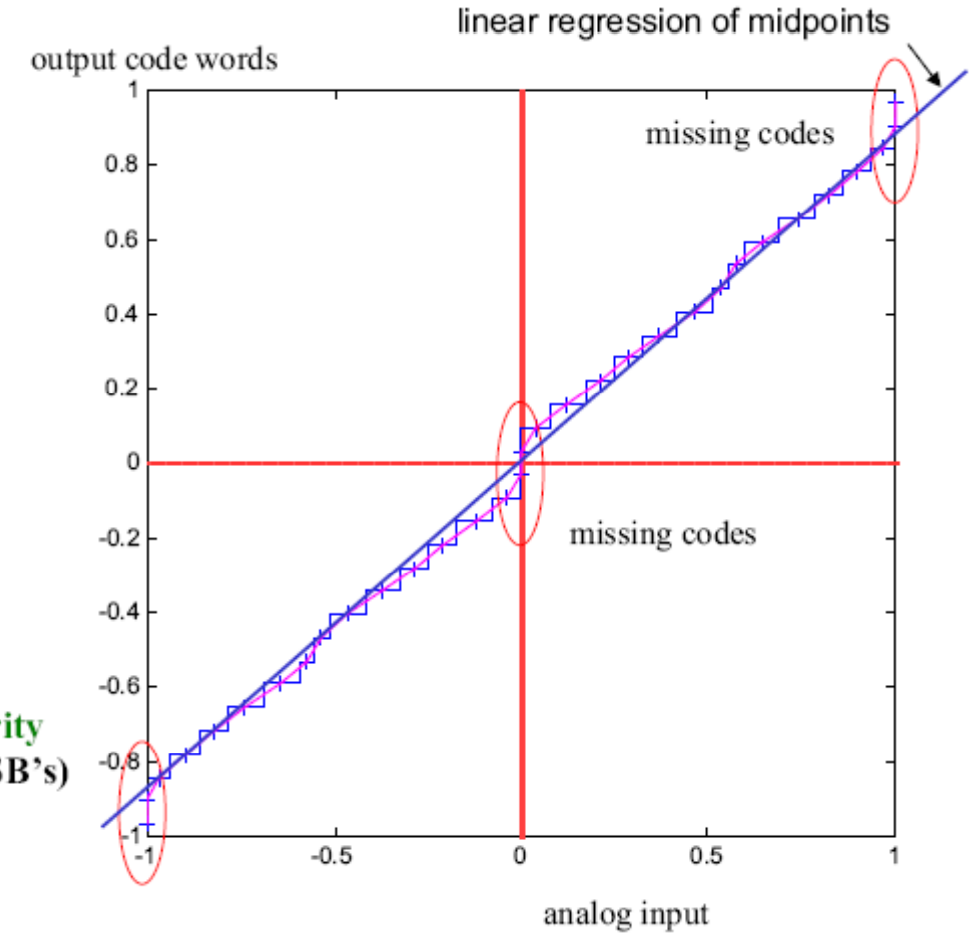

Definition of ADC nonlinearities: (b) INL.

• Integral Non-Linearity is defined as the maximum deviation of the ADC transfer function from the ideal transfer

# Linearity characterization

- A to D converters

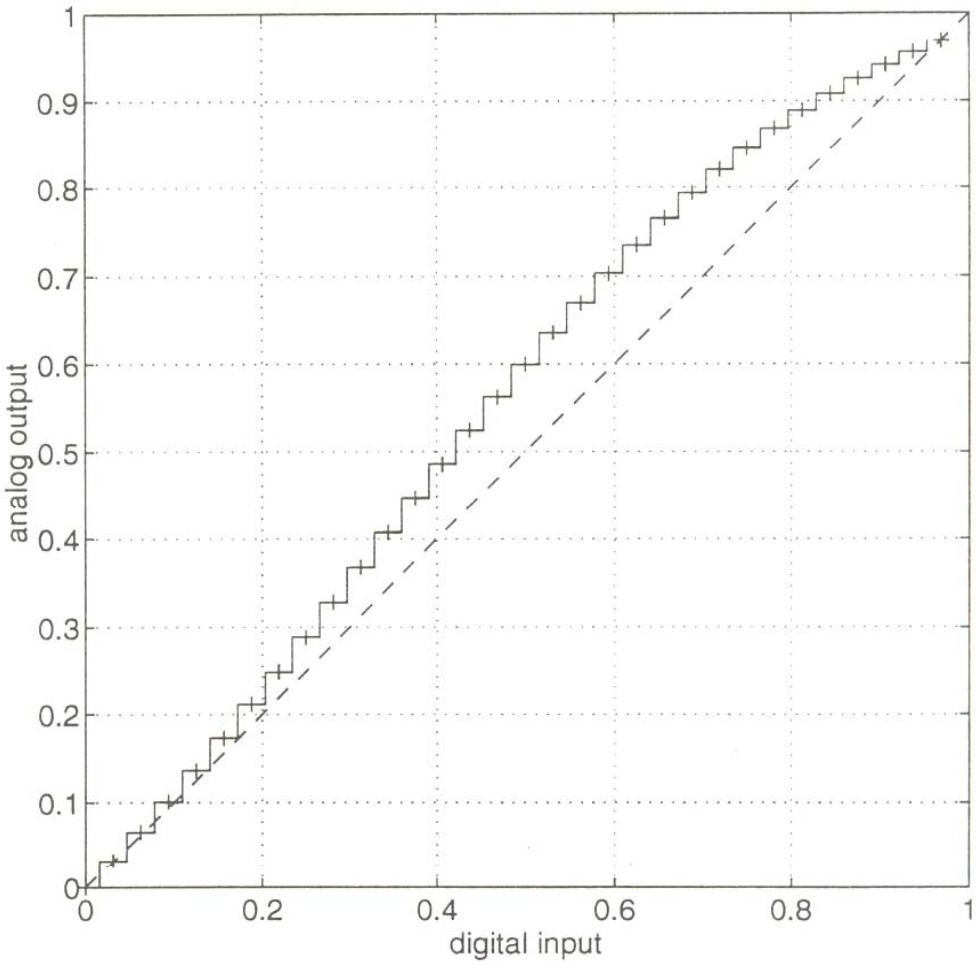


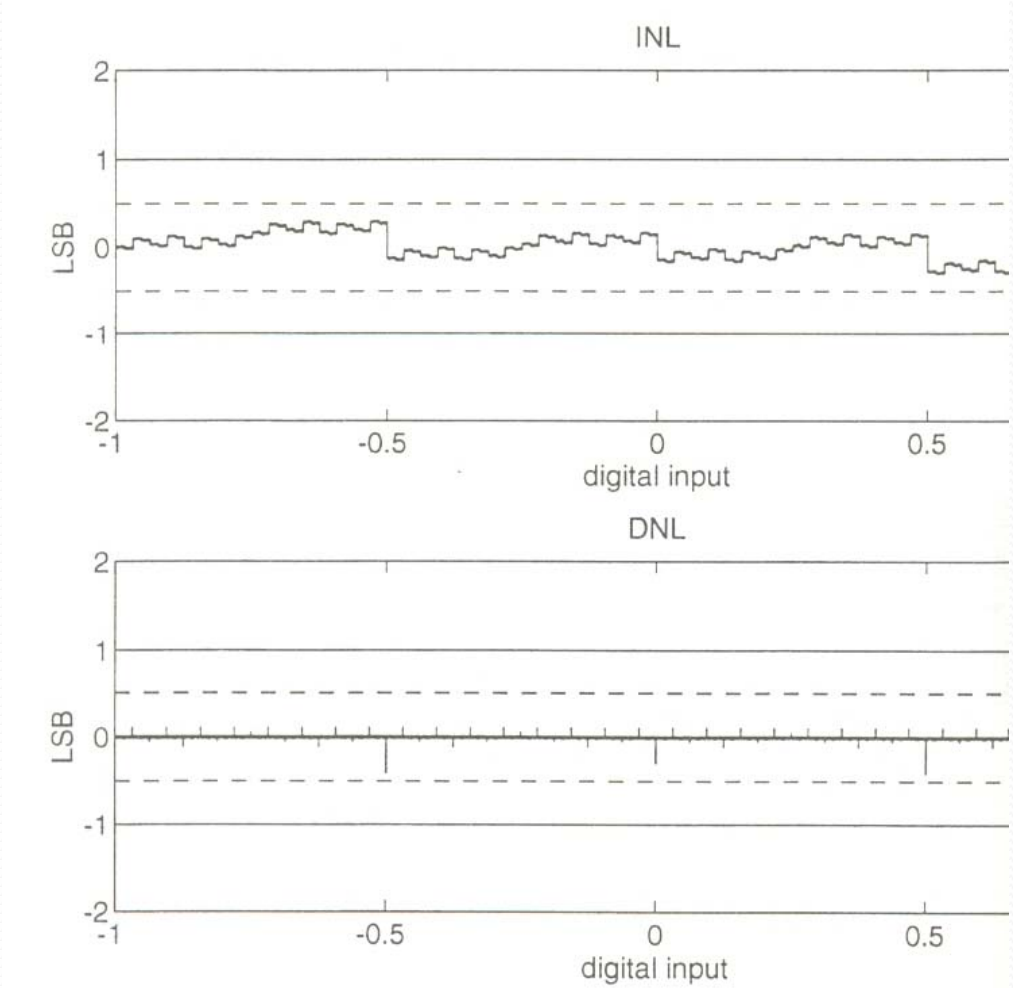

**INL** **Integral Non Linearity**

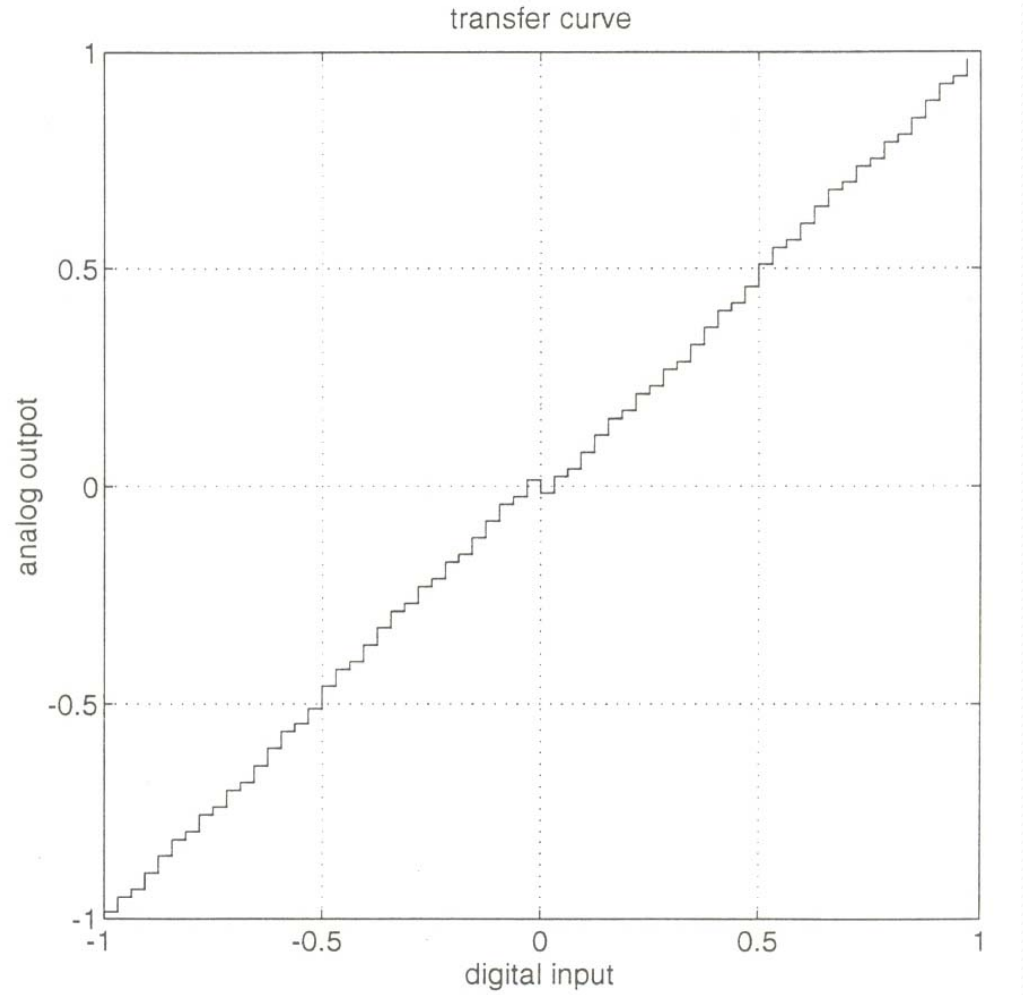
**DNL** **Differential Non Linearity**

**(both are expressed in terms of LSB's)**




D to A converter showing a strong Integral Non-Linearity (*INL*).

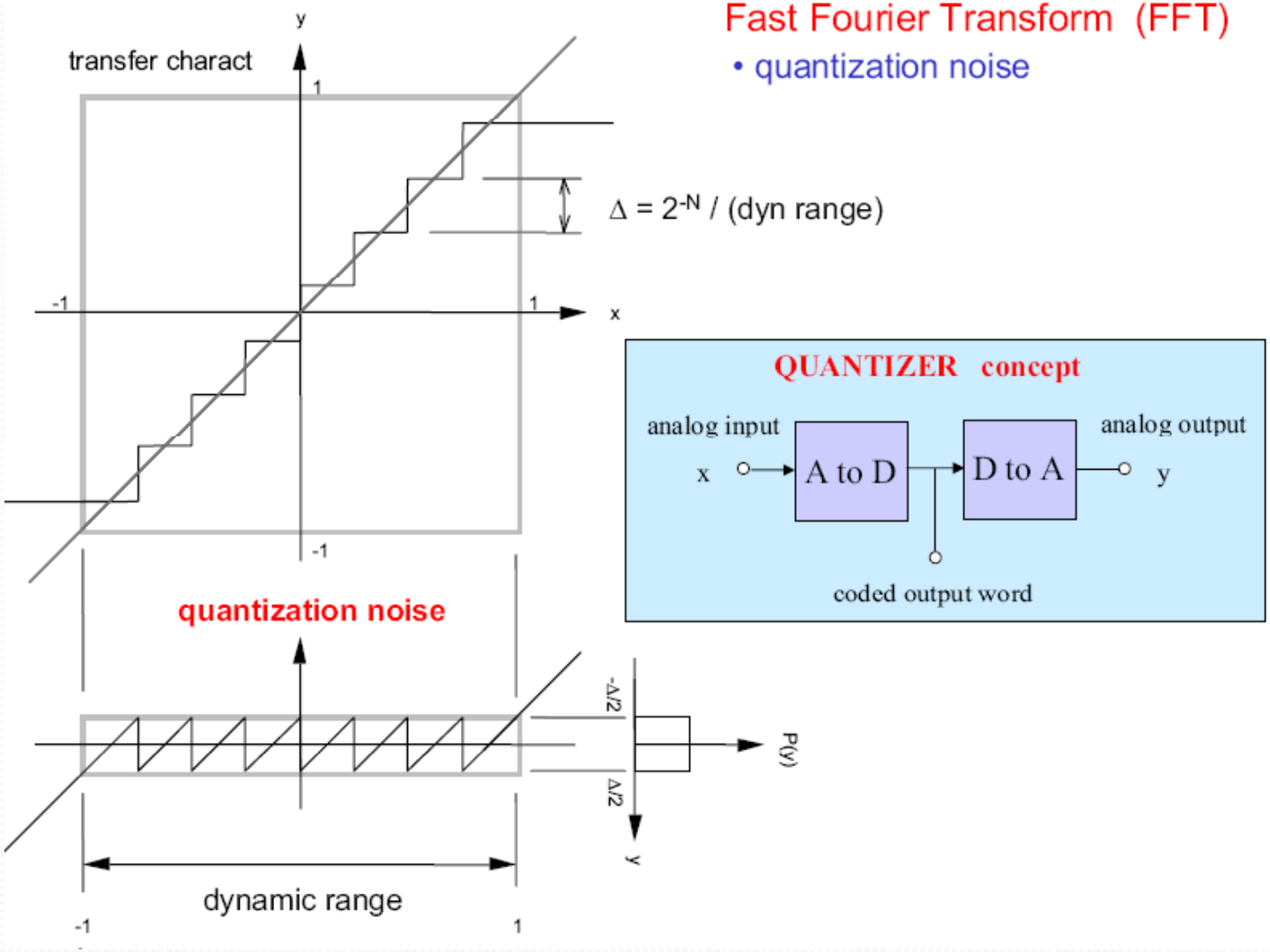

INL
2
1
0
-1
-2
LSB
-1
-0.5
0
0.5
digital input
DNL
2
1
0
-1
-2
LSB
-1
-0.5
0
0.5
digital input

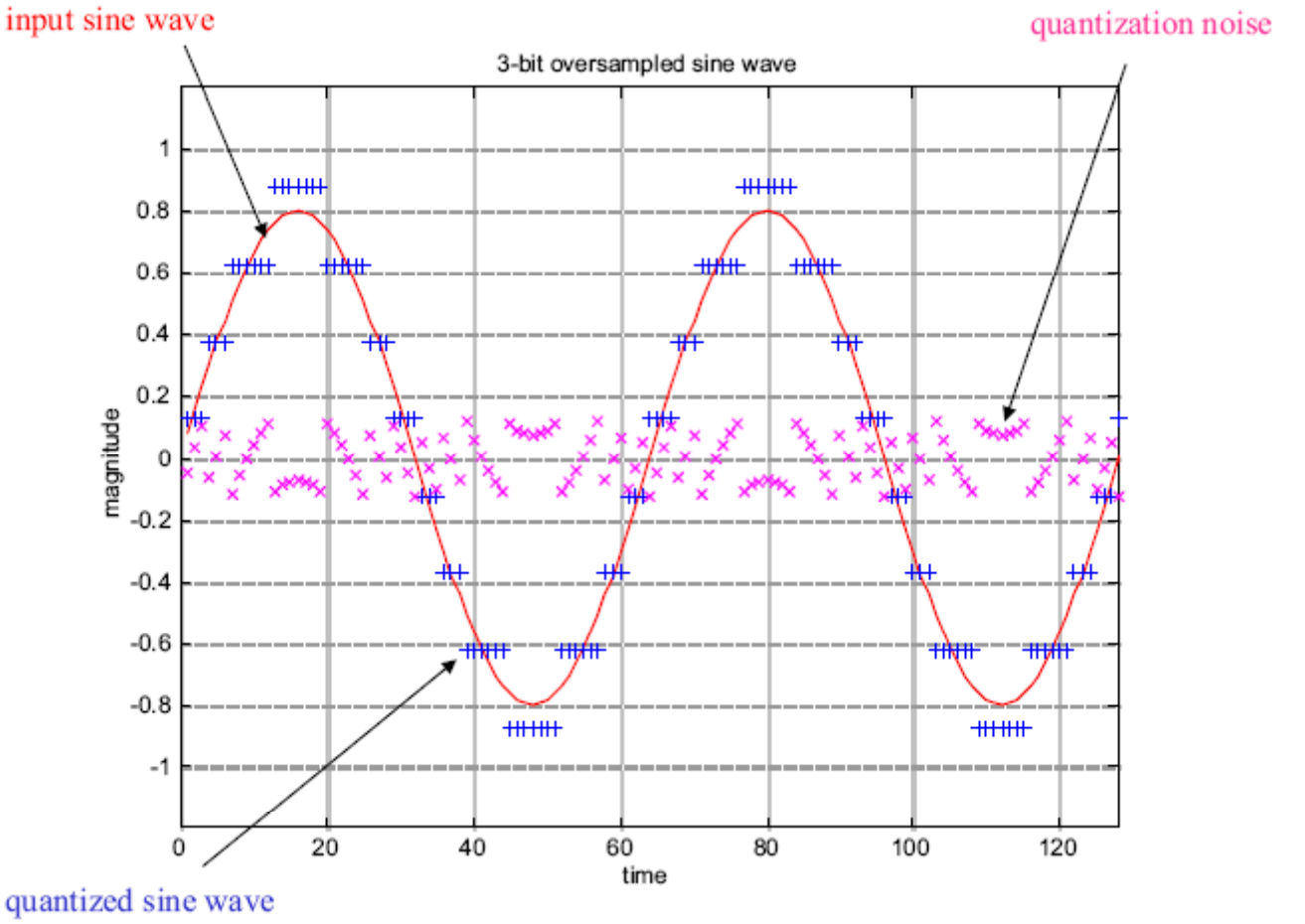


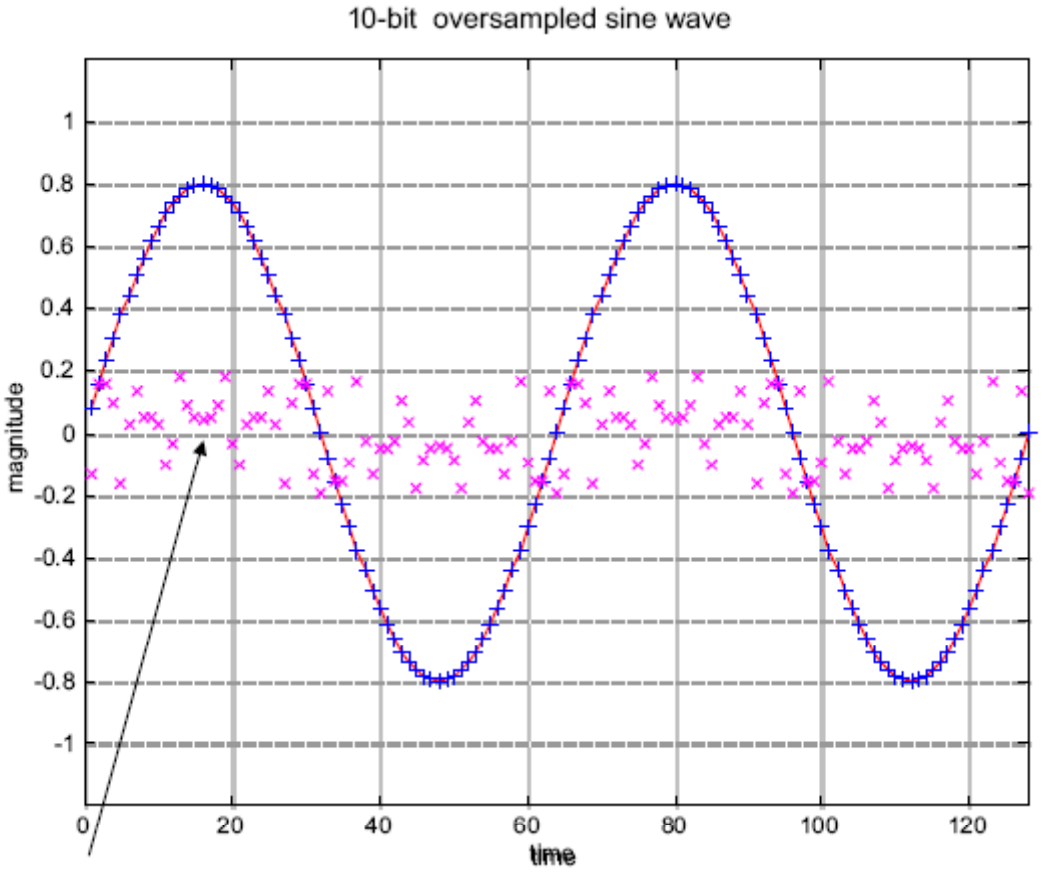

Unequal heights of consecutive steps are measured by the Differential Non-Linearity (*DNL*). In the middle, severe *DNL* causes non-monotonicity.

# Fast Fourier Transform (FFT)

- quantization noise

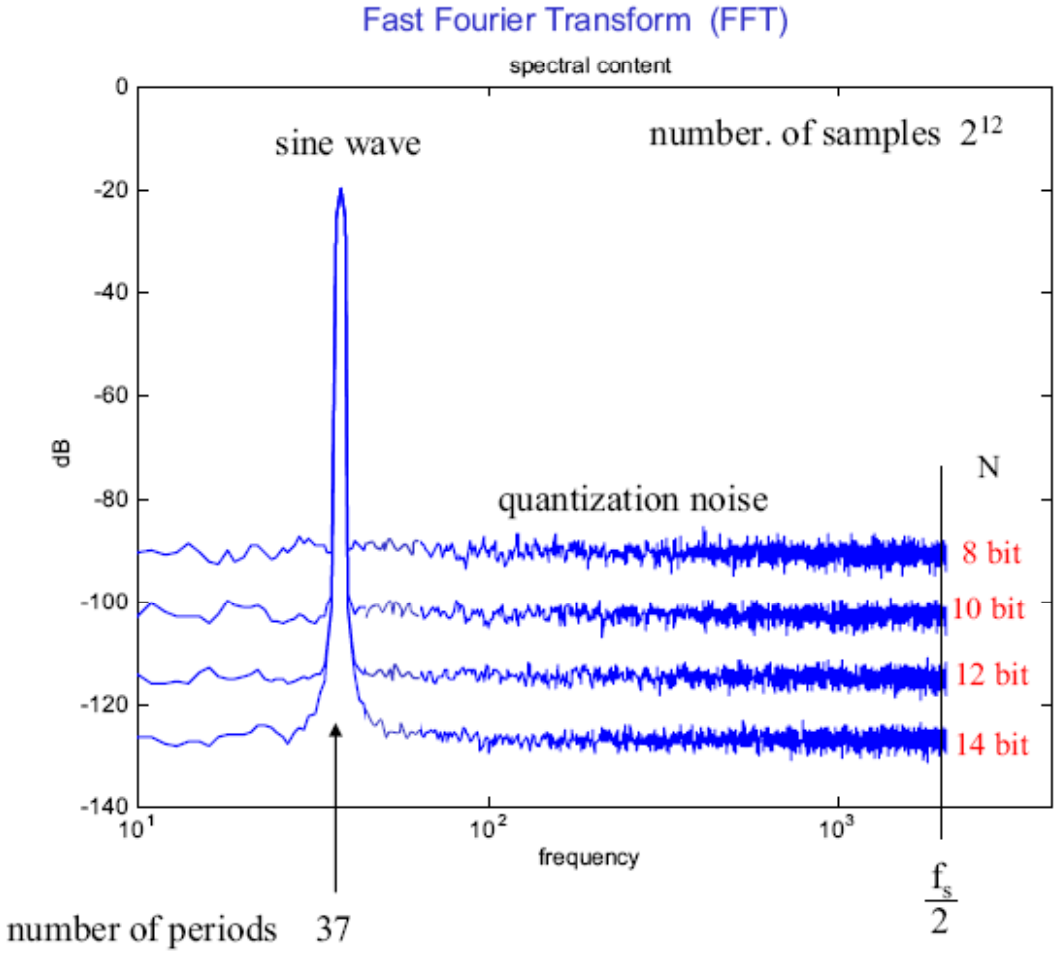



input sine wave
quantization noise
3-bit oversampled sine wave
magnitude
time
quantized sine wave

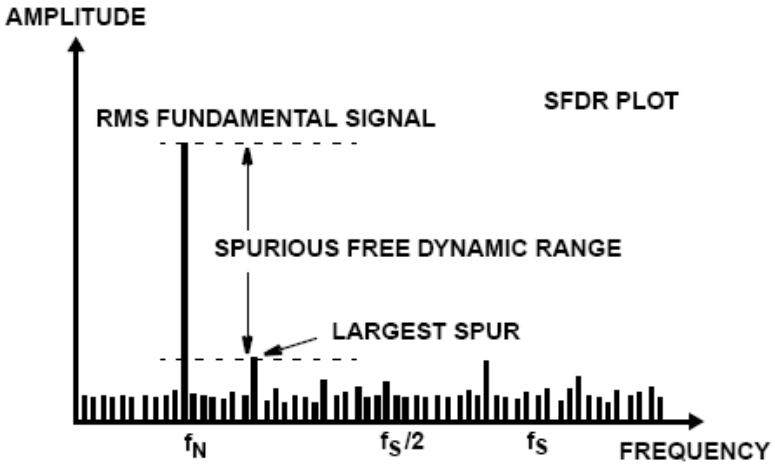


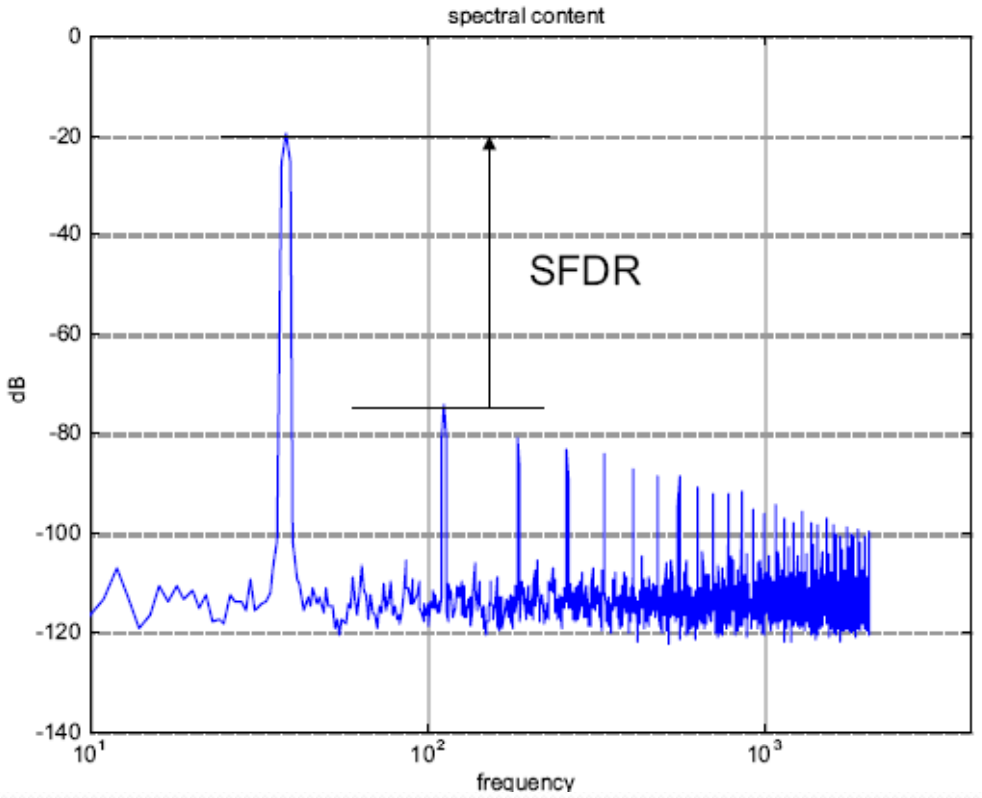

quantization noise
(magn multiplied by 200)

## power density spectrum of quantized sine waves



# Spurious Free Dynamic Range




Assumption:
quantization noise is uniformly distributed

Real SFDR measurement

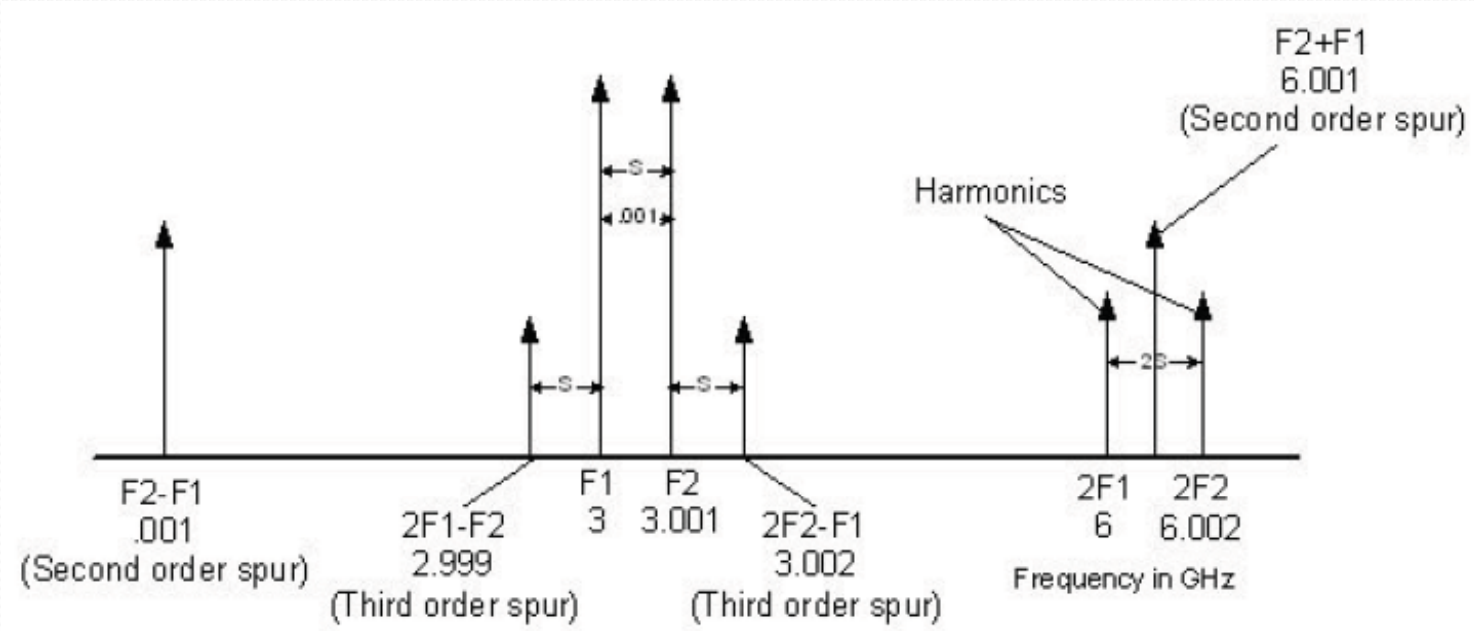
Spurious Free Dynamic Range is the usable dynamic range of a A/D or D/A converter before spurious noise interferes or distorts the fundamental signal.

SFDR: ratio of the RMS value of the signal (carrier) at the input of the ADC or DAC to the RMS value of the next largest noise or harmonic distortion component (which is referred to as a "**spurious**" or a "**spur**") at its output. SFDR is usually measured in dBc (i.e. with respect to the carrier frequency amplitude).

$\Rightarrow$ SFDR as an indicator of fidelity.

**$SNR_{IDEAL}$ = 6.02 (N) +1.76** **(only quantization noise!!)**

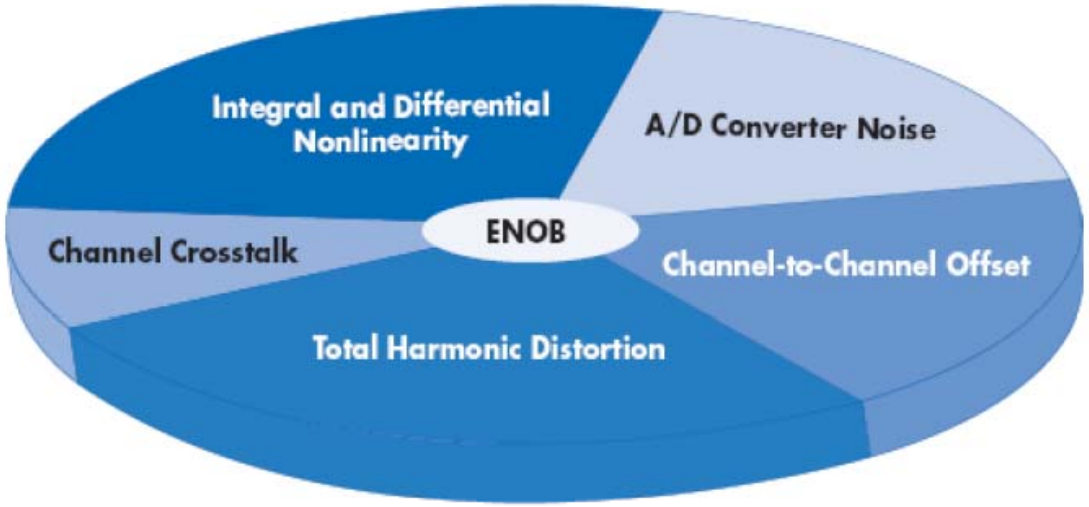
**(best-case:** assumption that quantization noise is uniformly distributed)



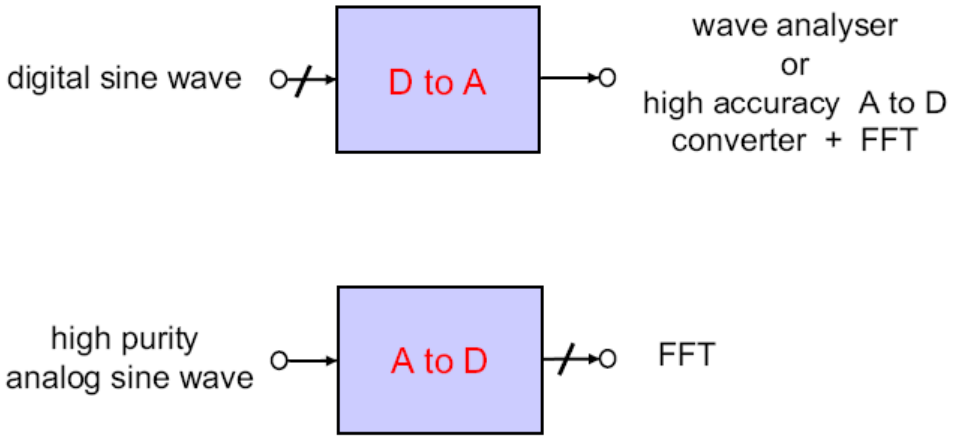

generation of spurious second and third-order products

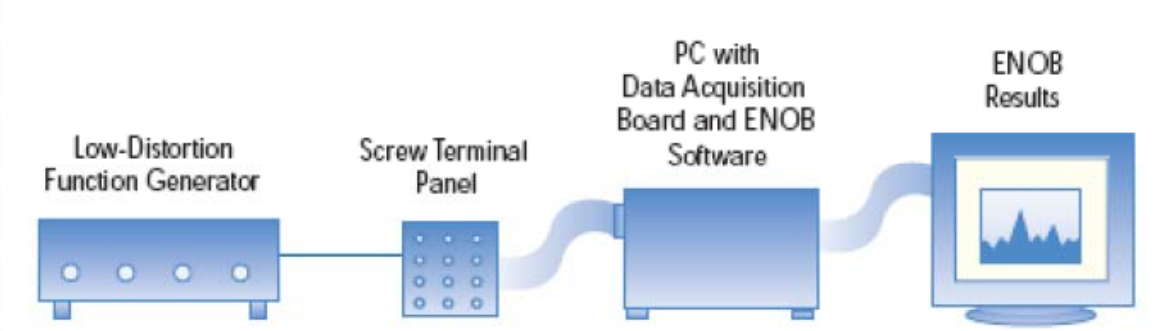
# ENOB (Effective Number of Bits): A measure of overall accuracy under real-world conditions

*ENOB Measures the Combined Effects of Multiple Sources of Noise and Distortion*

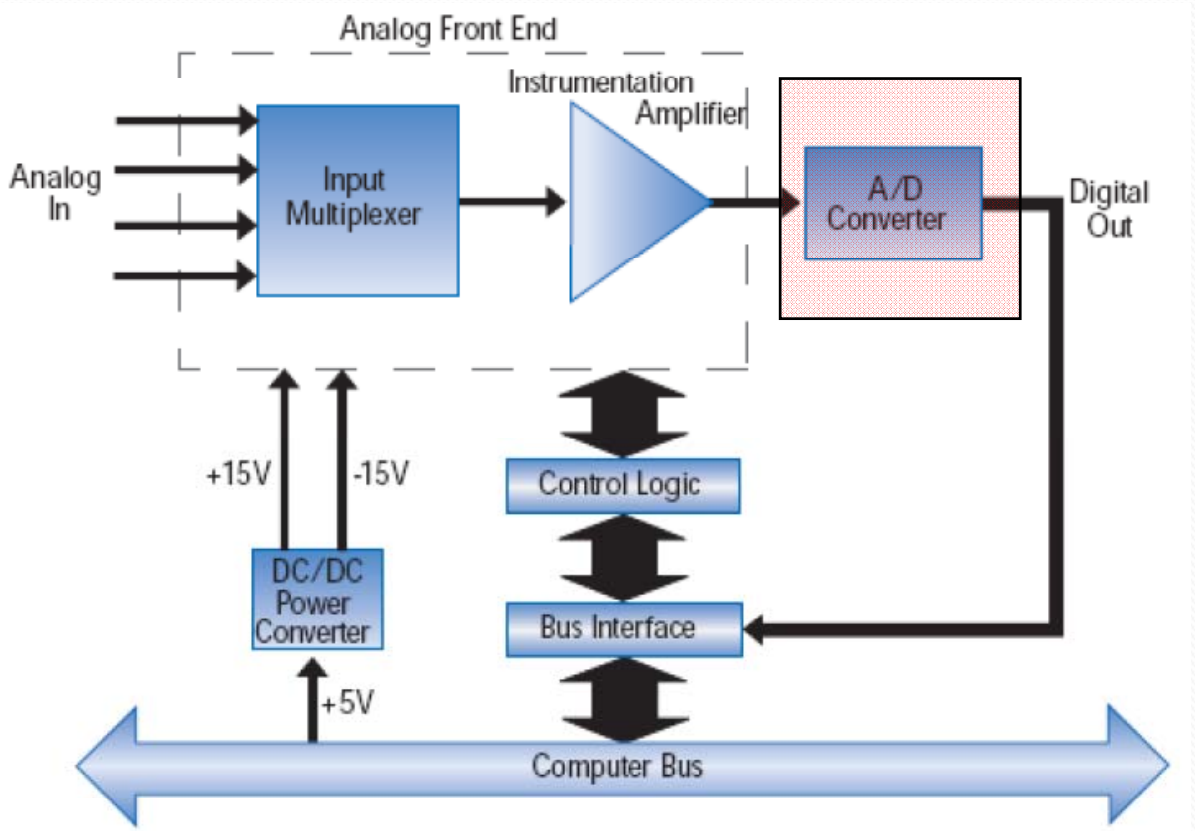


## Evaluation of ENOB




THD < 0.001%@1KHz




overall noise picked up by A/D

# Error (Distortion) Sources in ADCs

**DC:**

**i)** **integral and differential nonlinearity errors:** irreducible distortion, as ADC output is not continuous , but quantized. Even perfect ADC exhibit some integral and differential nonlinearity.

**ii)** **noise:**

a: quantization error - inherent to data conversion process

b: generated within the converter itself.

**iii)** **channel-to-channel offset:** difference in the characteristics of analog input channels

**AC:**

**i)** **THD:** ratio of all harmonics generated to the original signal frequency.

**ii)** **channel crosstalk:** signal interference analog input channels.

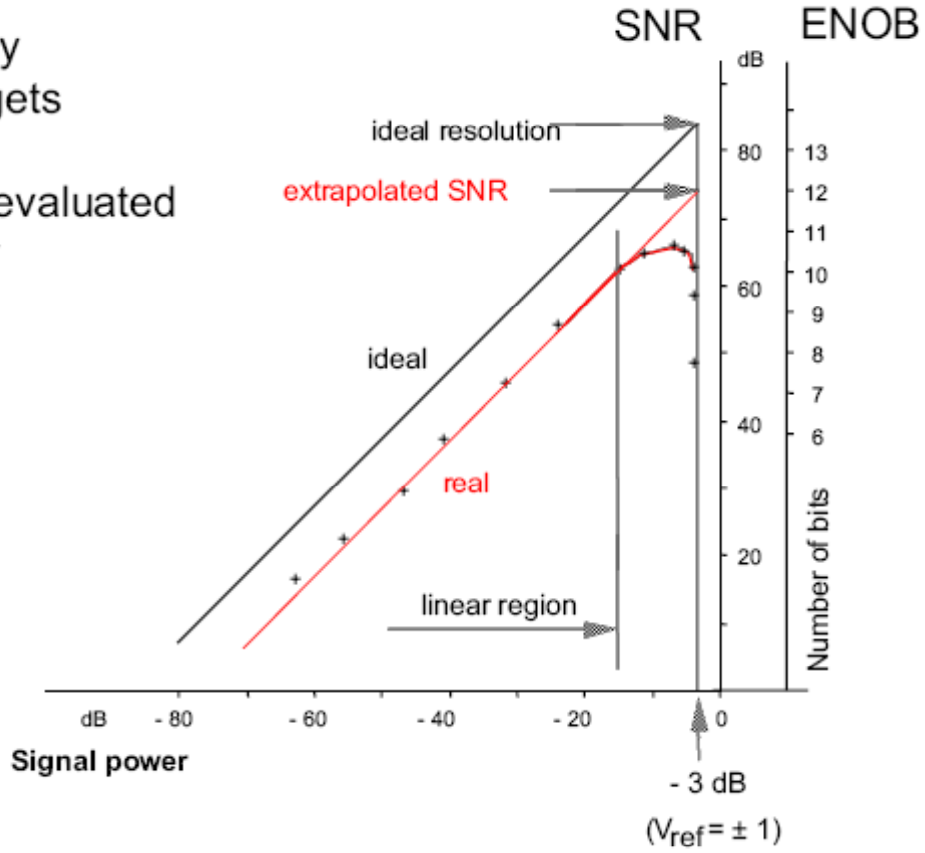
# ENOB Measurement of 16-bit ADC



**ENOB of 13.6 and 14.1 bits**

$$ENOB = \frac{SNR_{dB} - 1.8}{6}$$

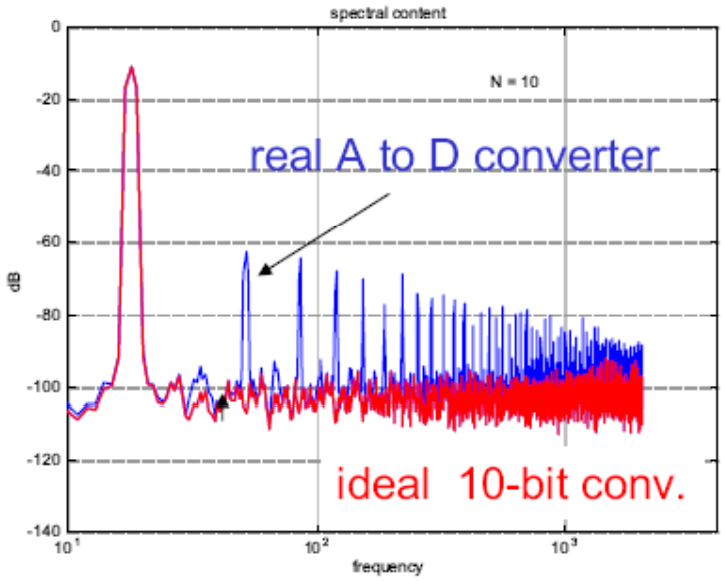
because saturation is likely to occur when the signal gets large the SNR, which determines the ENOB, is evaluated by extrapolating the linear part of the plot until F.S.



$$SNR_{IDEAL} = 6.02\,(N) + 1.76$$

**(best-case:** assumption that quantization noise is uniformly distributed)

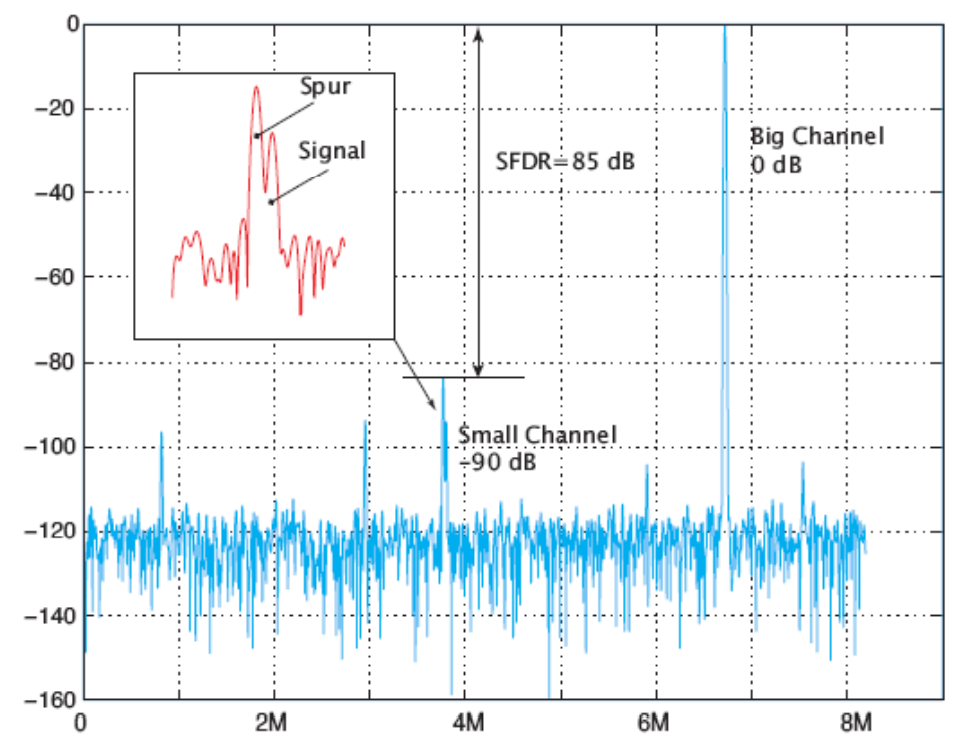
where N is the number of bits of the converter. For N=10, SNR = 62dB.

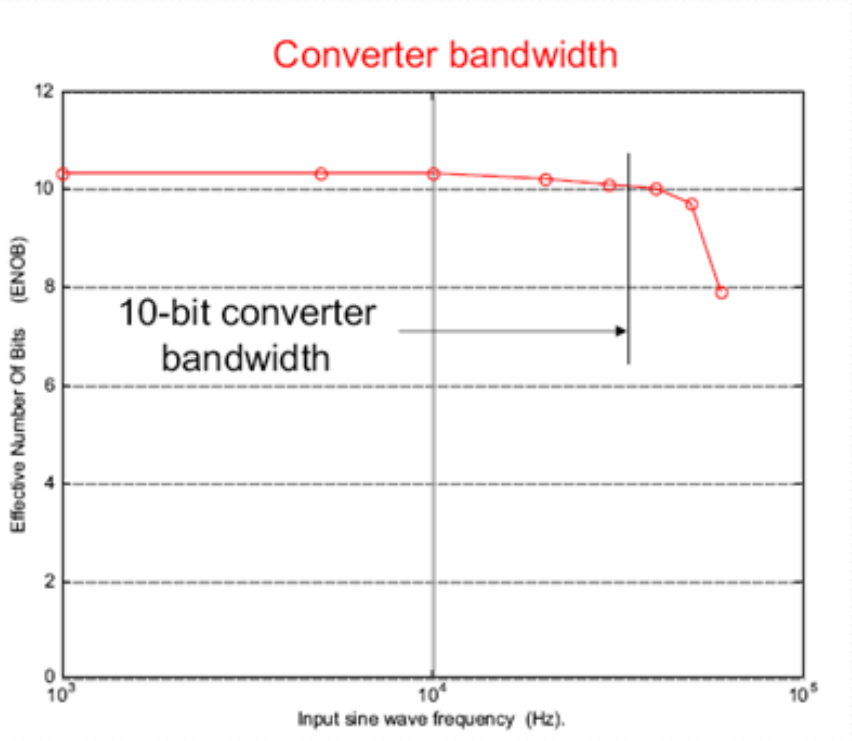


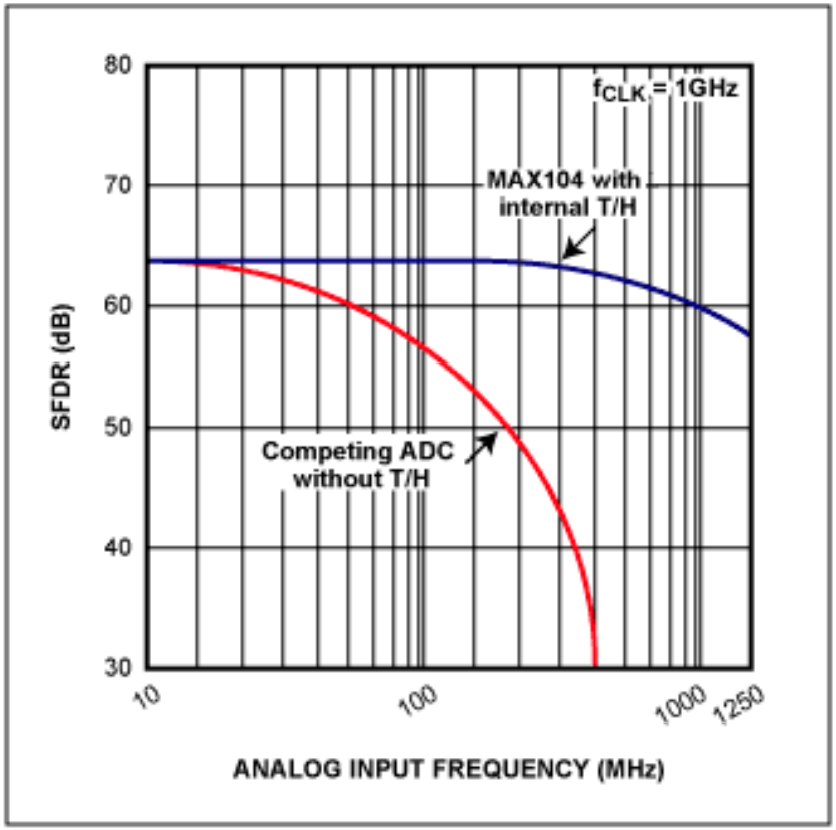

$$ENOB = \frac{SNR_{dB} - 1.8}{6}$$

*example*: SNR = 53 dB
ENOB = 8.5

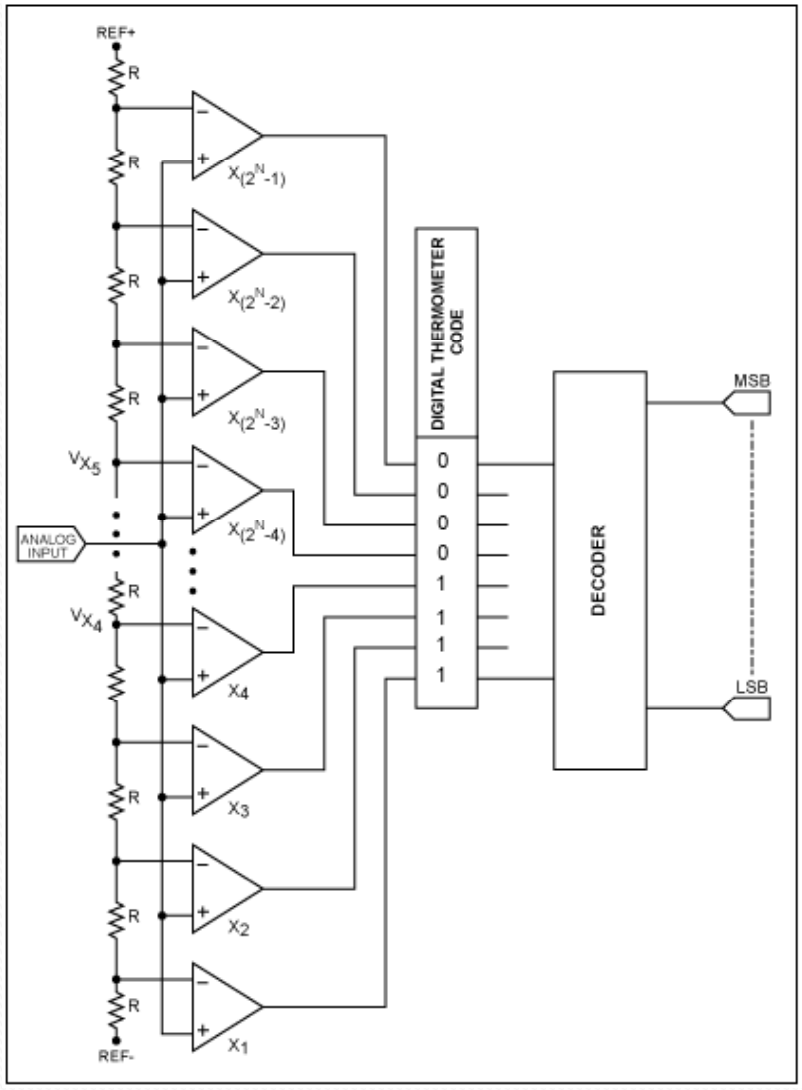
Ex: The input signal has two channels: a strong one (0*dB) @6.72 MHz, and a small one (−90 dB) @3.8 MHz. The sampling frequency* is 16.*4 MHz.*
The big component generates a large third harmonic at 20.*16 MHz which gets folded down to 3.76 MHz, just 40 kHz from the* small channel. Even if the *SFDR is 85 dB the spur almost completely masks* the *−90 dB signal.*





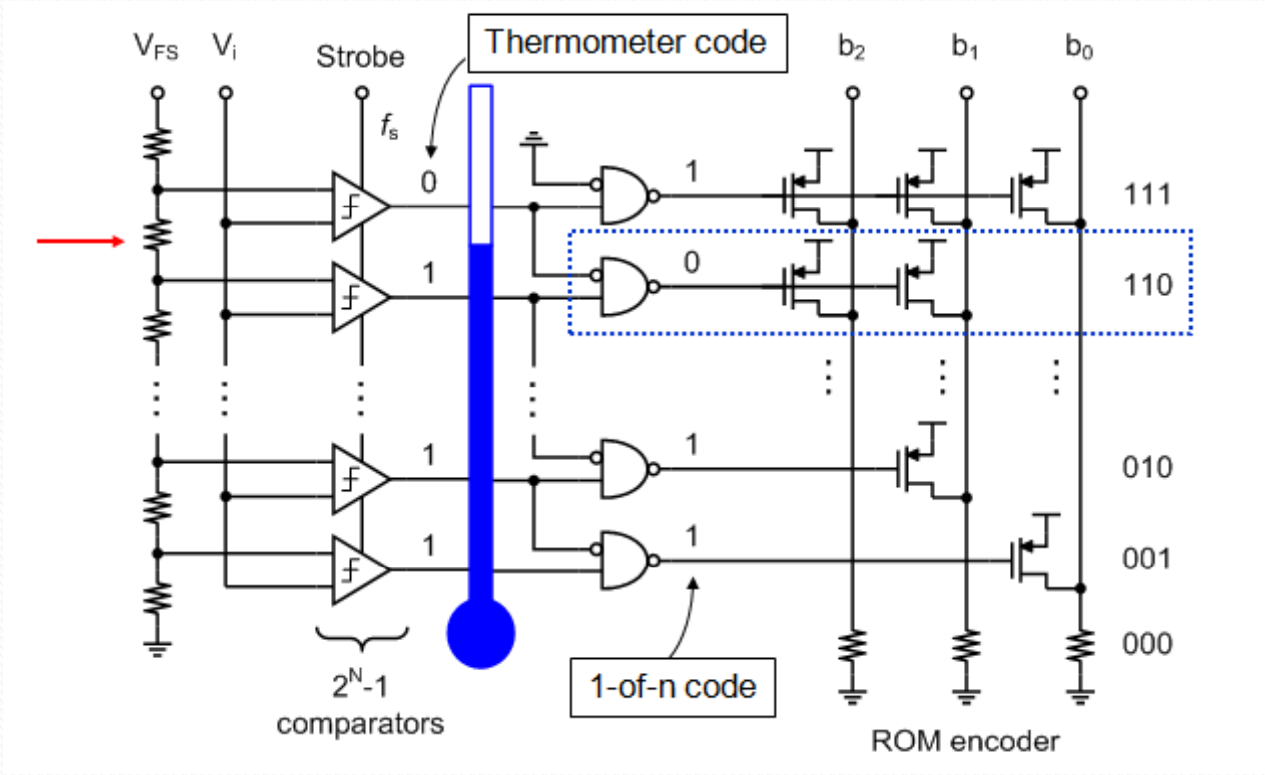




*Spurious free dynamic range as a function of input frequency*



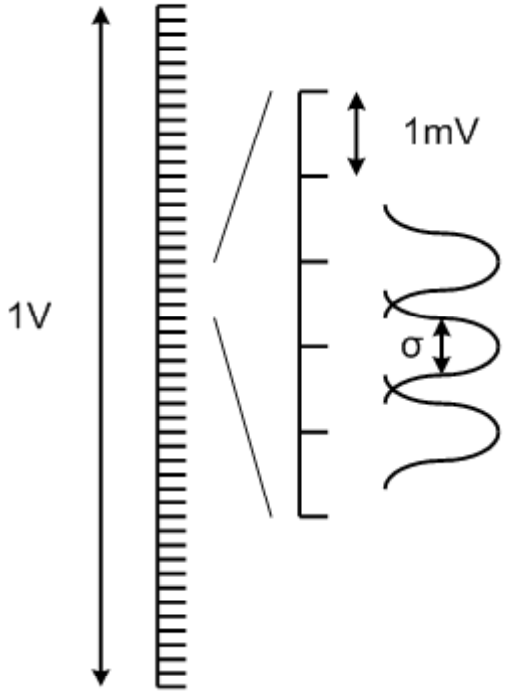

# FLASH ADC

- *$2^N$ – 1 comparators*
- *typically consume more power than other ADC architectures*
- *generally limited to 8-bits resolution*
- *relatively low resolution and expensive*
- *ideal for applications requiring very large bandwidth*
- exceed giga-sample per second (Gsps) conversion rates
- *each comparator represents 1 LSB*
- *output code can be determined in one compare cycle.*


$V_{FS}$
$V_i$
Strobe
$f_s$
Thermometer code
$b_2$
$b_1$
$b_0$
0
1
1
1
1
0
1
1
111
110
010
001
000
$2^N$-1
comparators
1-of-n code
ROM encoder

| Thermometer | | | | | | | Gray | | | Binary | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 |
| $T_1$ | $T_2$ | $T_3$ | $T_4$ | $T_5$ | $T_6$ | $T_7$ | $G_3$ | $G_2$ | $G_1$ | $B_3$ | $B_2$ | $B_1$ |

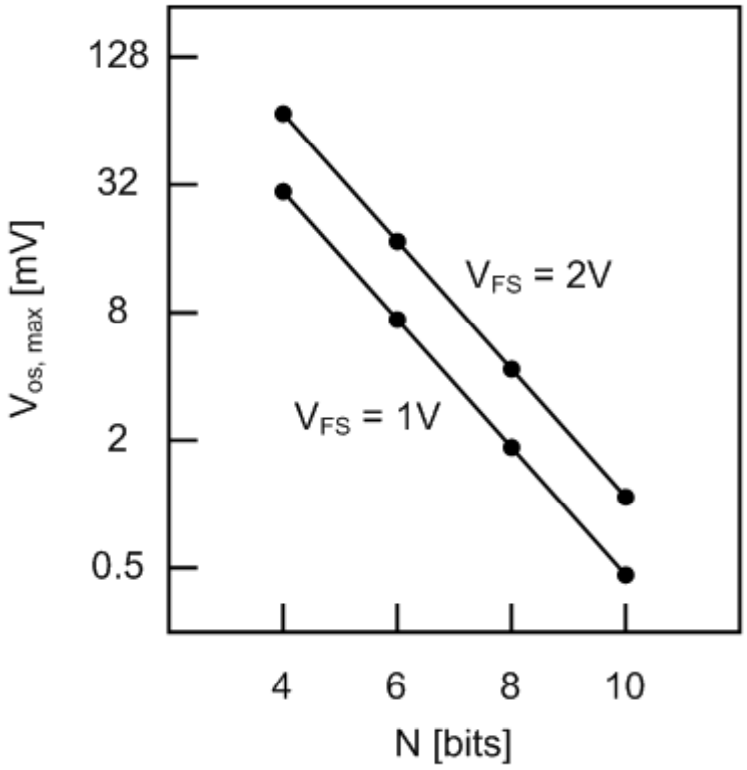
# A 10-bit Flash ADC




- $V_{DD}$ = 1.8 V
- 10-bit → 1023 comparators
- $V_{FS}$ = 1 V → 1 LSB = 1 mV
- DNL < 0.5 LSB → $V_{os}$ < 0.5 LSB
- 0.5 mV = 3-5 σ → σ = 0.1-0.2 mV

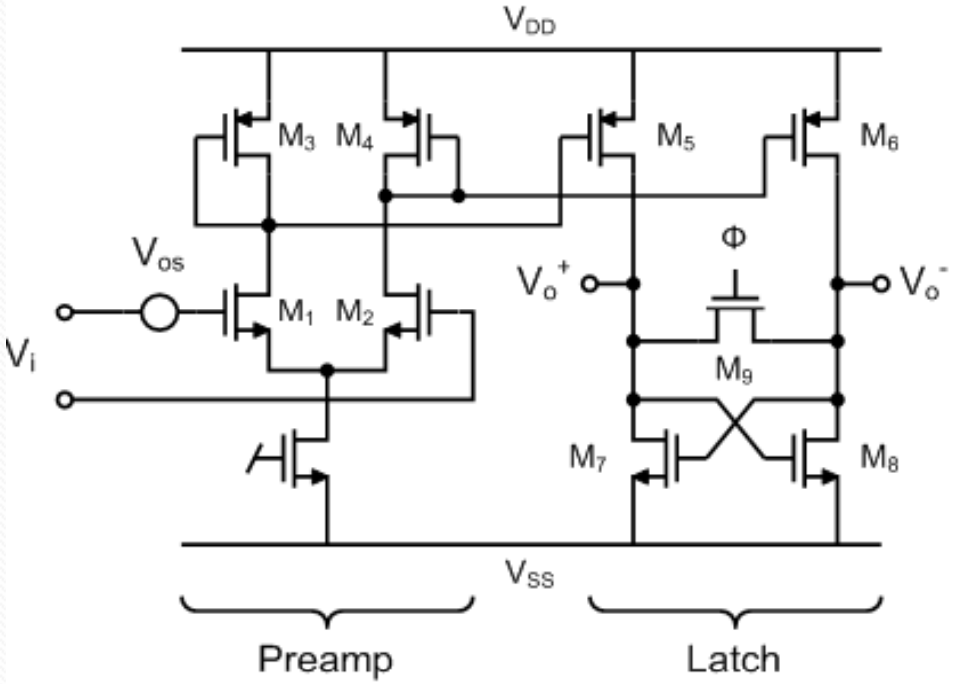
- $2^N$-1 very large comparators
- Large area, large power consumption
- Very sensitive design
- Limited to resolutions of 4-8 bits

# Offset vs. Resolution




- DNL < 0.5 LSB
- Large $V_{FS}$ relaxes offset tolerance
- Small $V_{FS}$ benefits conversion speed (settling, linearity of building blocks)

• Typical CMOS comparator

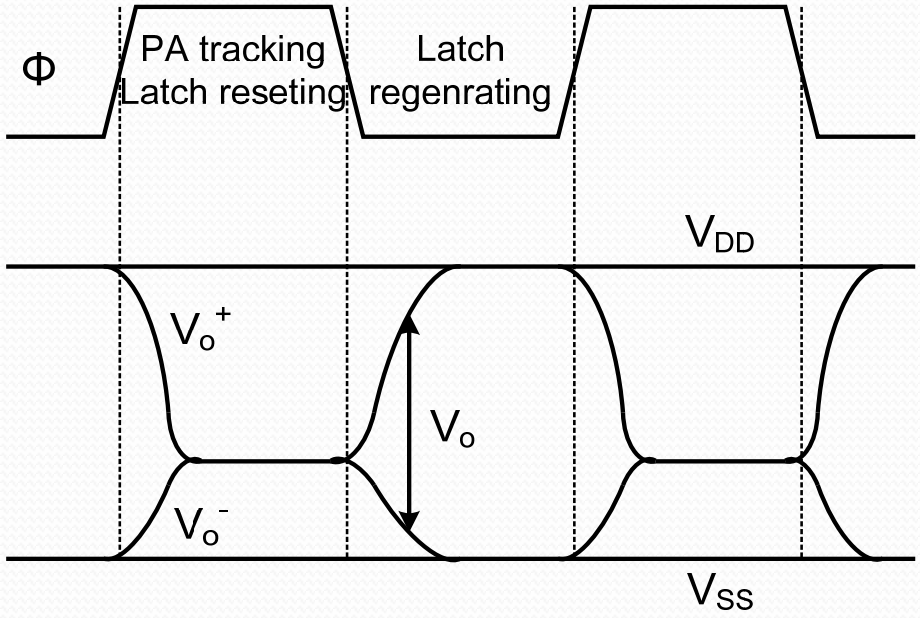



$V_{os}$ derives from:

- Preamp input pair mismatch ($V_{th}$,W,L)
- PMOS loads and current mirror
- Latch mismatch
- Cl / CF imbalance of $M_9$
- Clock routing
- Parasitics

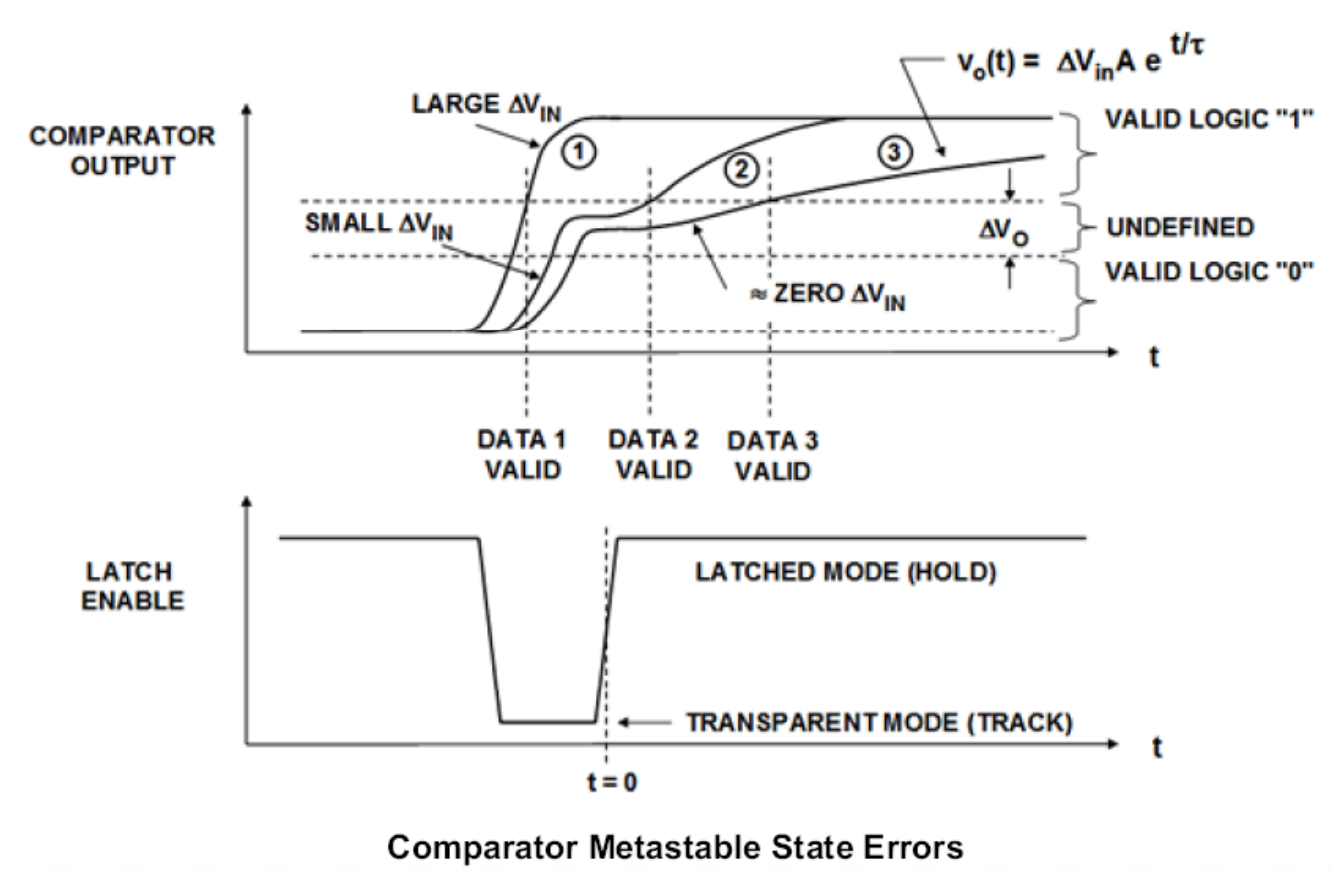
# Latch Regeneration

- exponential regeneration due to positive feedback of $M_7$ and $M_8$

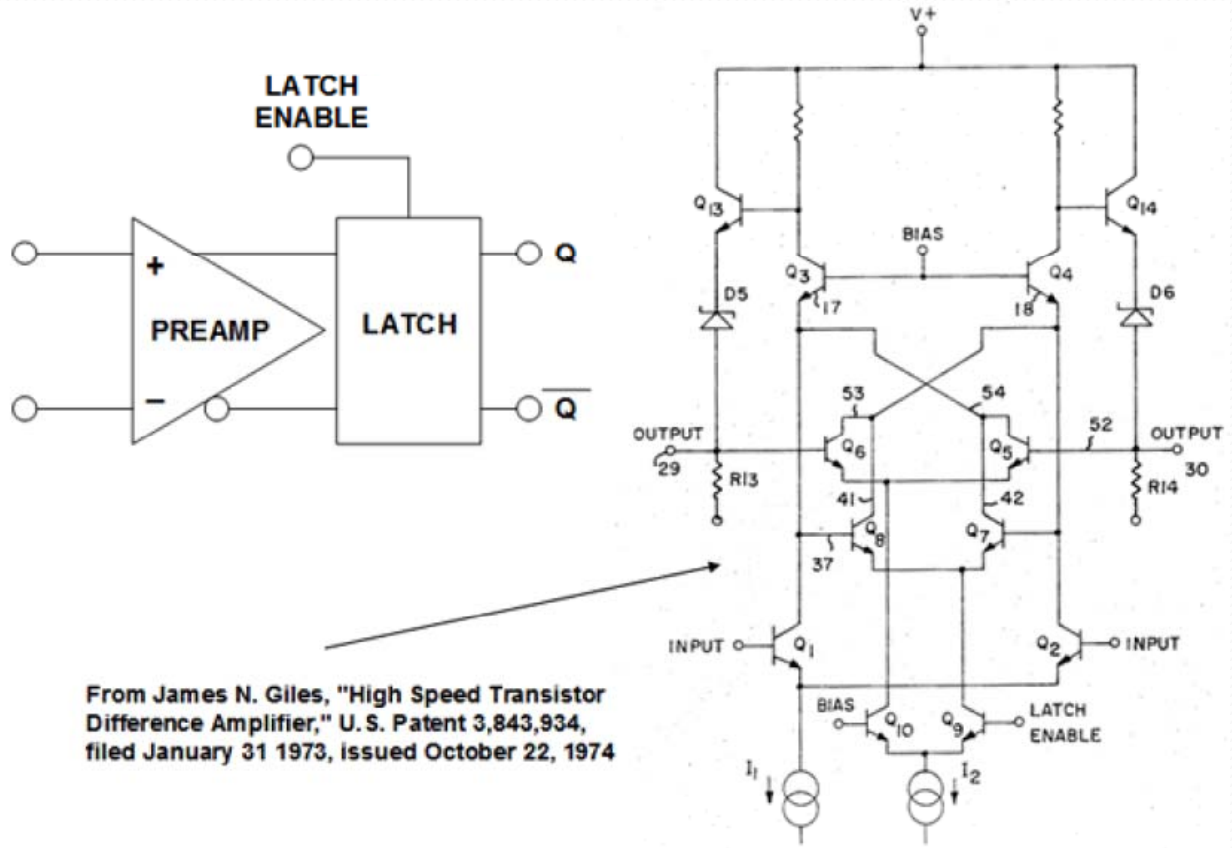


(1) large differential input voltage
(2) small differential input voltage
(3) ≅ zero differential input voltage

ΔVIN: the differential input voltage at the time of latching,
A = the gain of the preamp at the time of latching,
τ = regeneration time constant of the latch,
t = the time that has elapsed after the comparator output is latched




Comparator Metastable State Errors



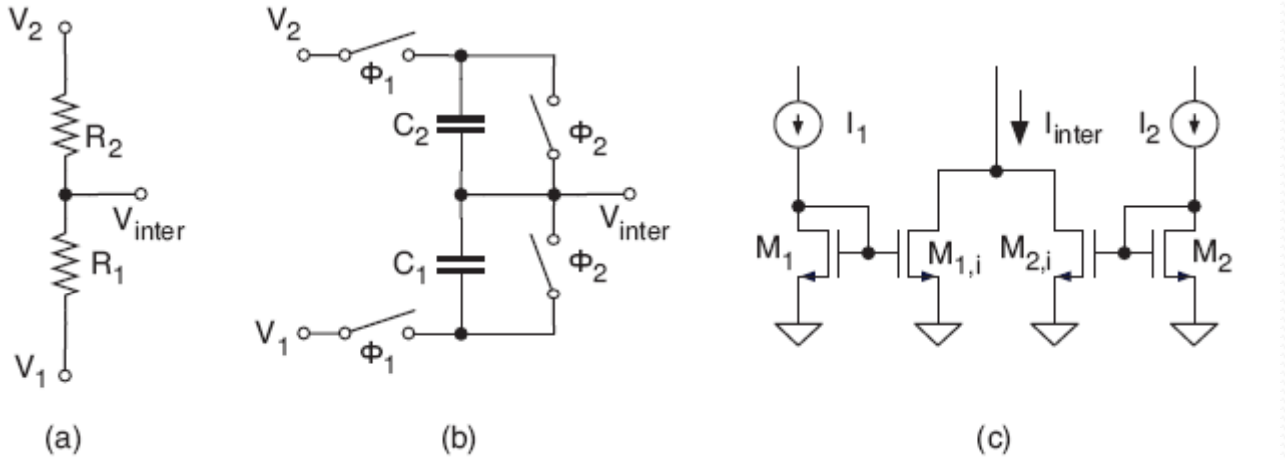

The AM685 ECL Comparator (1972)

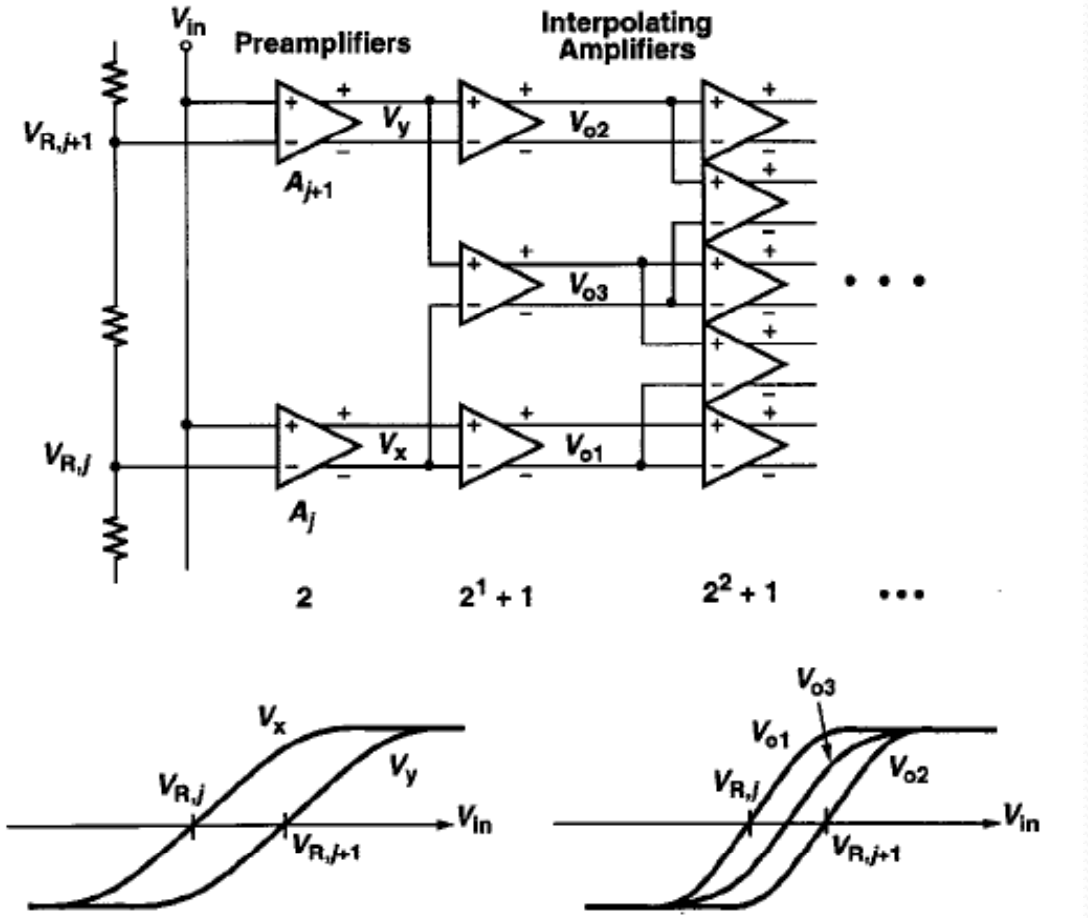
# Interpolation A/D Converters

An interpolator generates an electrical value that is intermediate between two other electrical quantities by using, for voltage inputs: resistive or capacitive dividers and, for current inputs: schemes based on current mirrors.

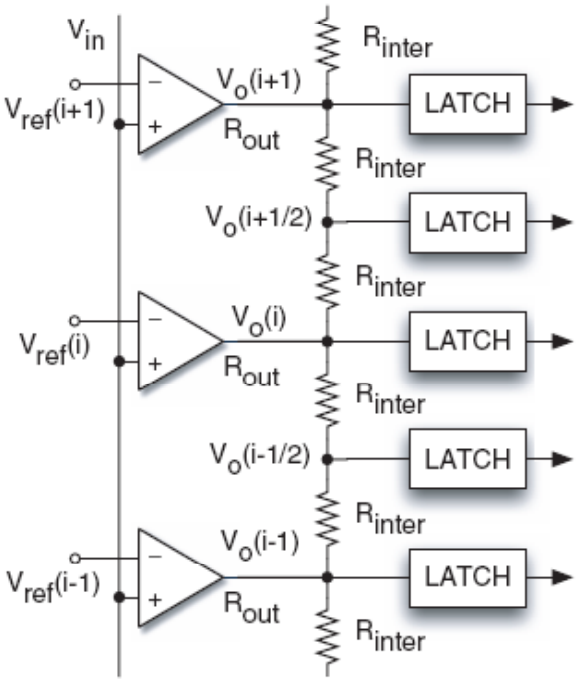
$$V_{inter} = \frac{V_1 R_2 + V_2 R_1}{R_1 + R_2}. \qquad V_{inter}(\phi_2) = \frac{V_1 C_1 + V_2 C_2}{C_1 + C_2}. \qquad \frac{(W/L)_{1,i}}{(W/L)_1} = \alpha; \quad and \quad \frac{(W/L)_{2,i}}{(W/L)_2} = 1 - \alpha$$

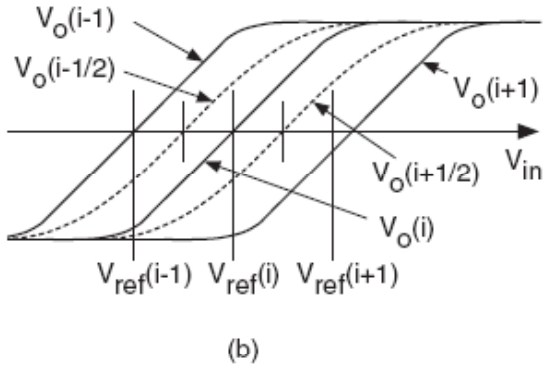
$$I_{inter} = \alpha \cdot I_1 + (1 - \alpha) \cdot I_2 .$$

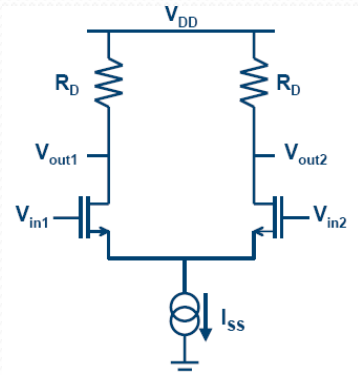


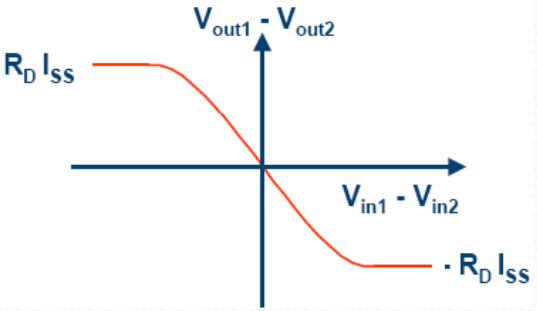



Traditional active 2x interpolation architecture.



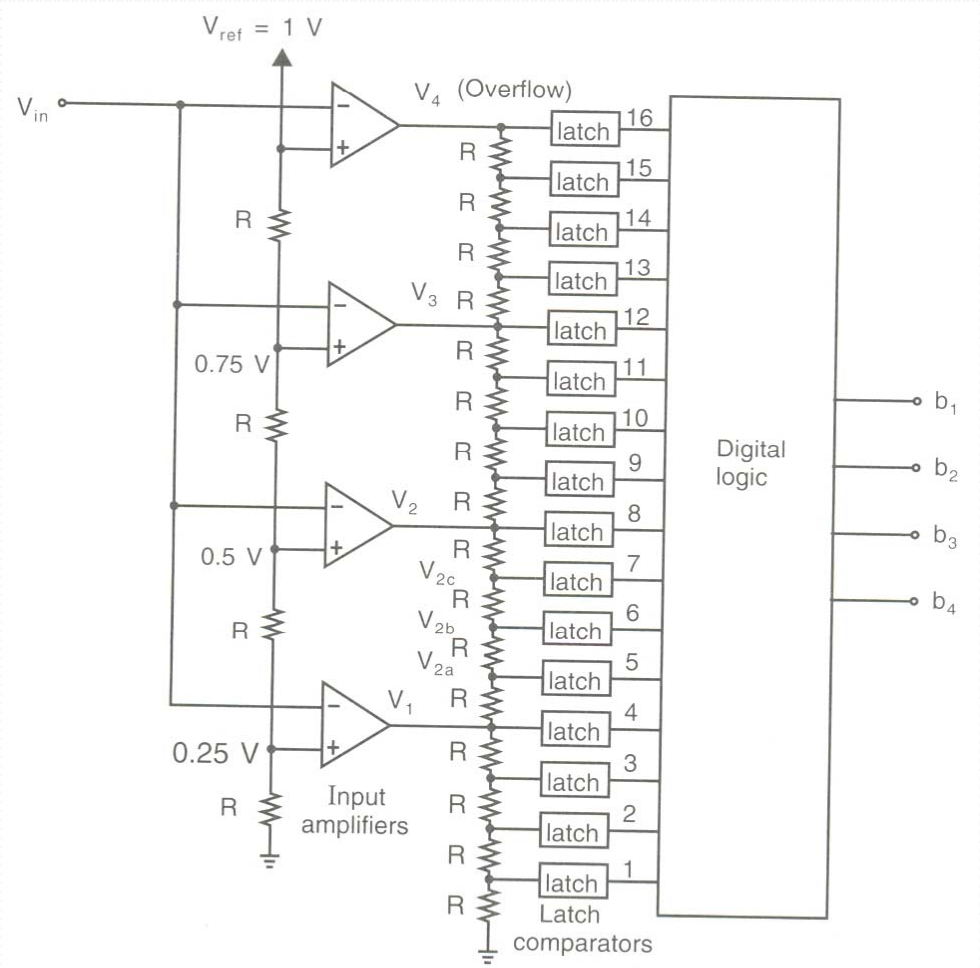



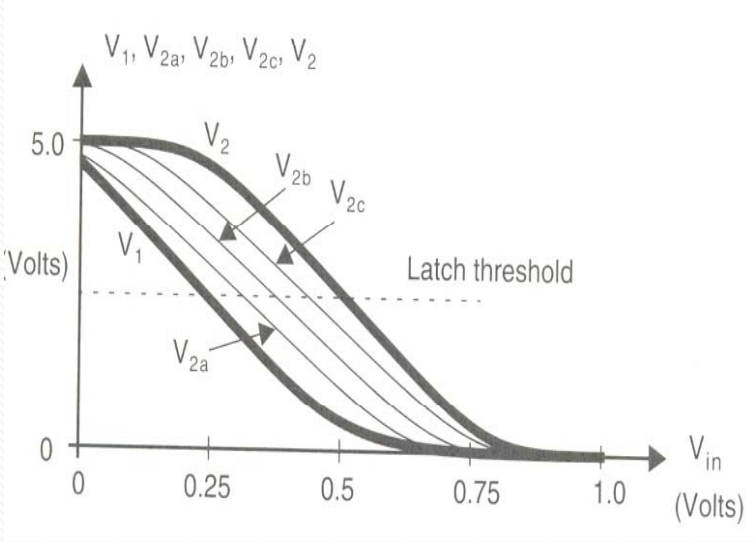




a) Use of interpolation in flash converters. b) Outputs and interpolated responses.

# 4-bit interpolating flash A/D Converter

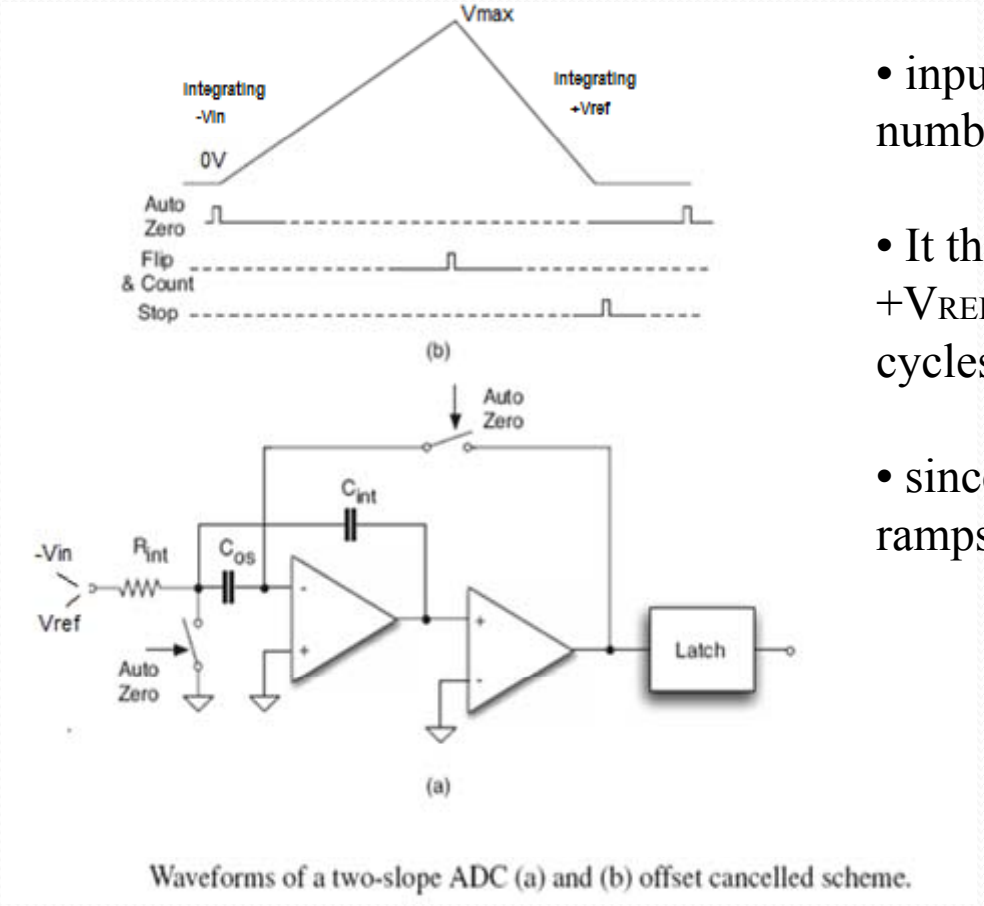



## $2^n$ latches



# Integrating A/D Converter

• Single, dual and multi-slope ADCs can achieve high resolutions of 16-bits or more are relatively inexpensive and dissipate materially less power.




Waveforms of a two-slope ADC (a) and (b) offset cancelled scheme.

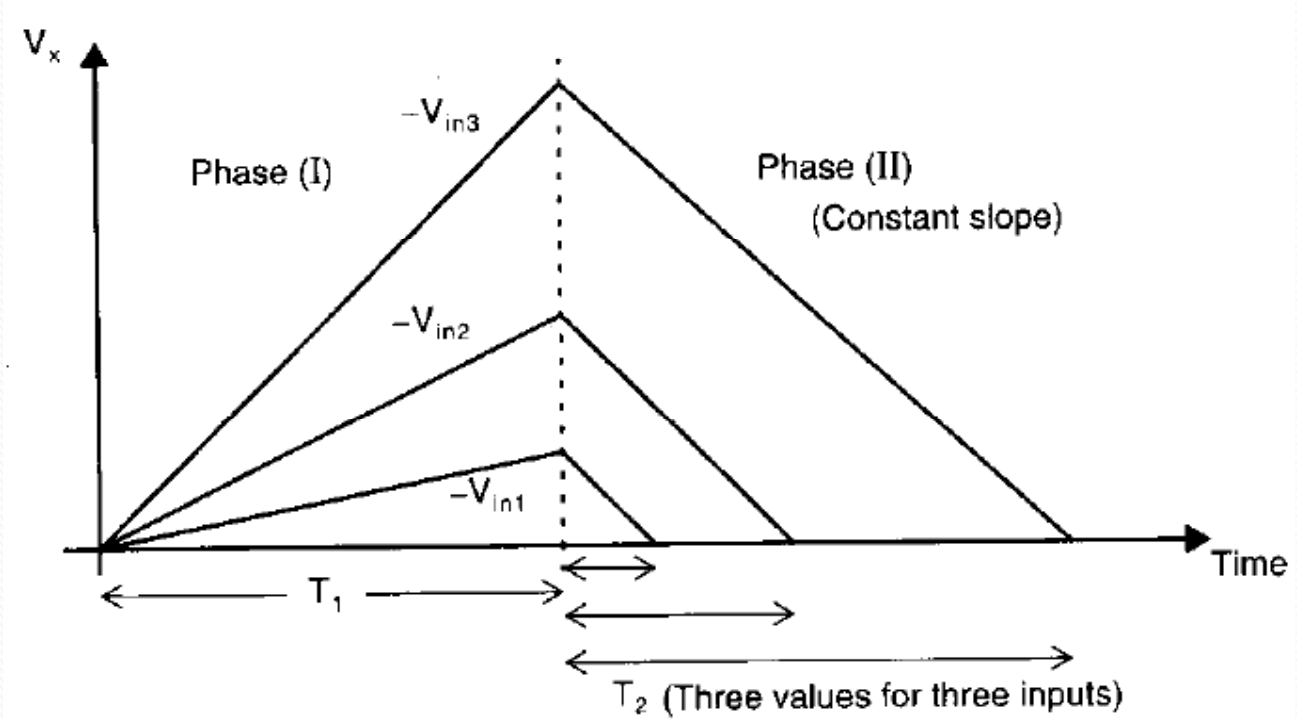
• input voltage (-$V_{IN}$) is integrated for a fixed number of clock periods, $2^n$.

• It then"de-integrates" Vmax by changing to +$V_{REF}$ at input and counts the number of clock cycles $k$ until it crosses zero.

• since peak amplitudes of rising and falling ramps are equal $\Rightarrow$

$$V_{in}\frac{2^n T_{ck}}{\tau} = V_{ref}\frac{k \cdot T_{ck}}{\tau}$$

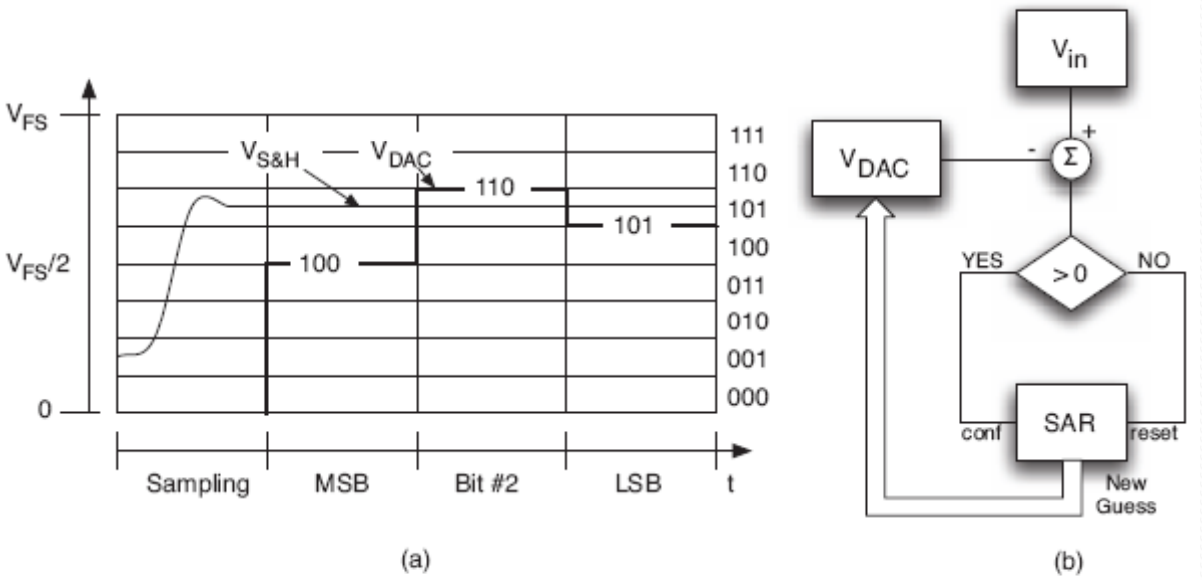
$$k = 2^n \frac{V_{in}}{V_{ref}}$$




Operation of the integrating converter for three different input voltages.

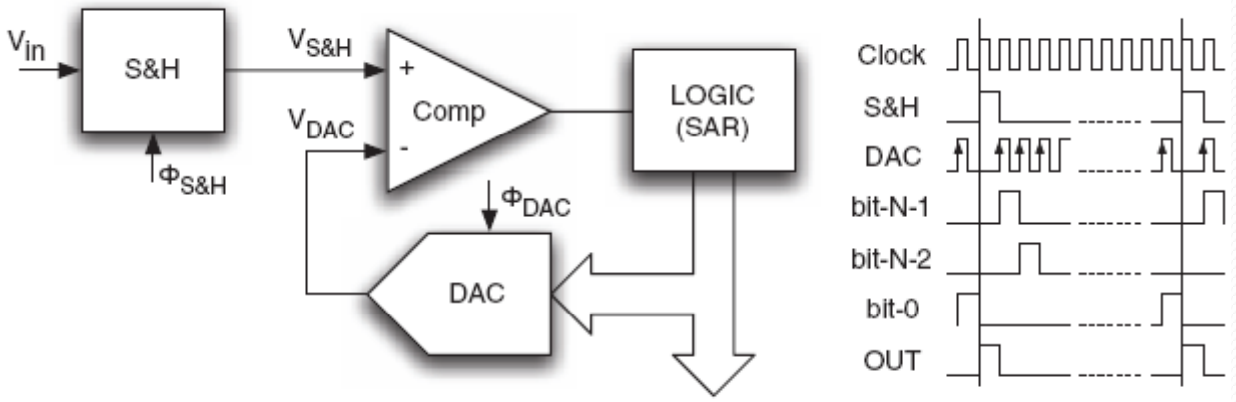
# Successive-approximation A/D

- quick convertion time and moderate complexity
- binary-search algorithm




Timing (a) and flow diagram (b) of the successive approximation technique.

*Successive Approximation Converter*



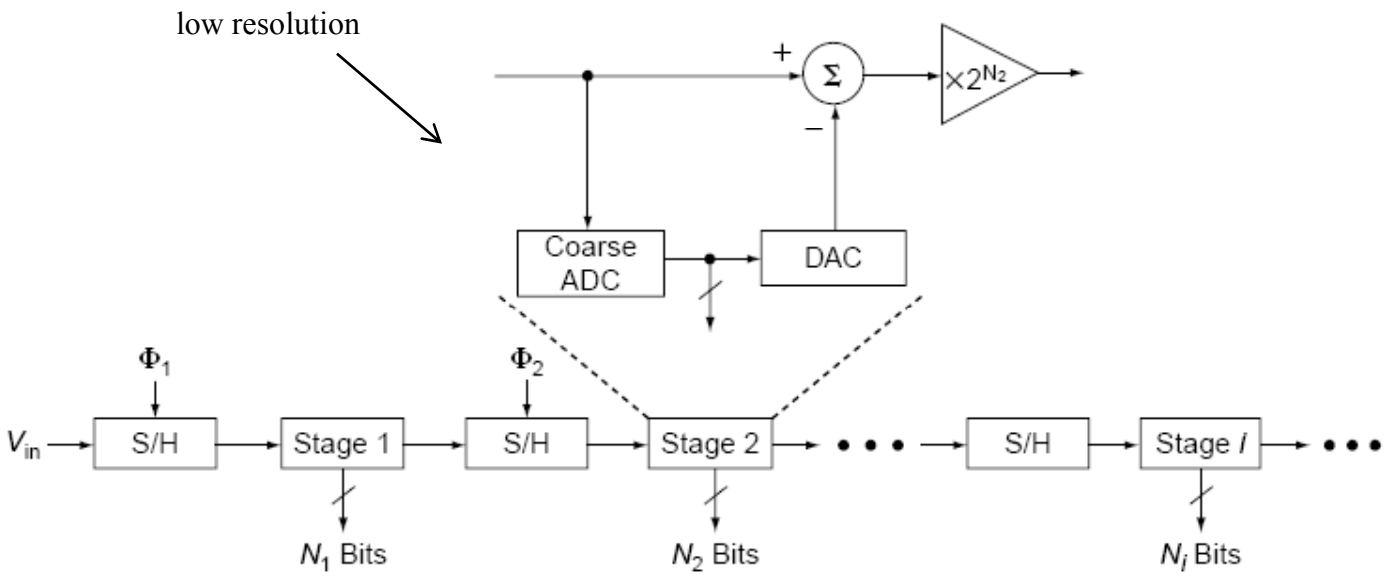

Basic circuit diagram of the successive approximation algorithm.

• The method uses one clock period for the S&H and one clock period for the determination of every bit

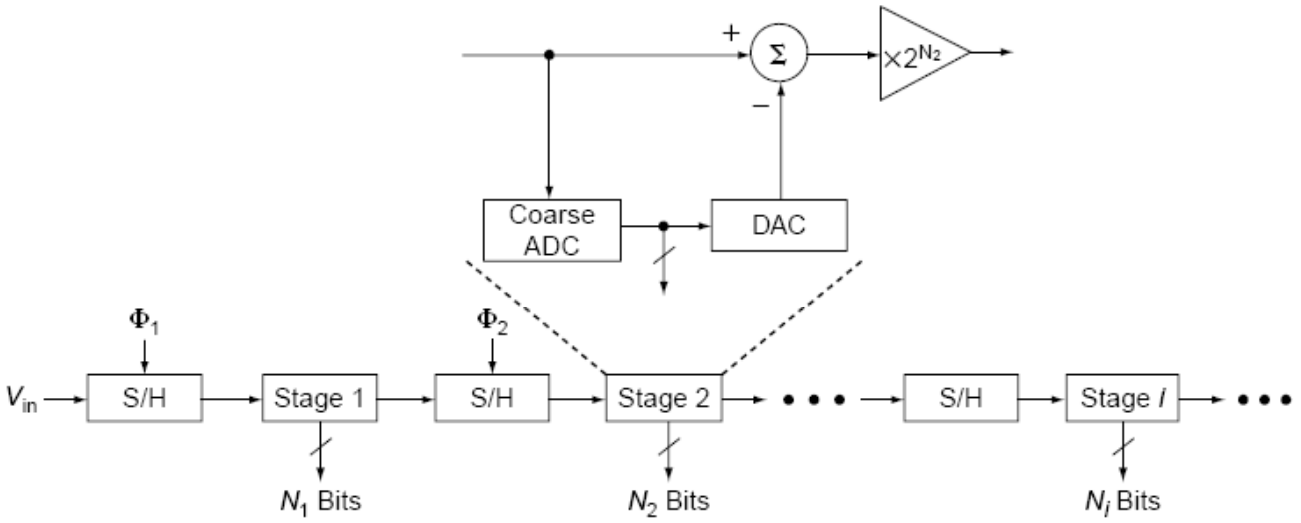
⇒ (n + 1) clock intervals for an n-bit conversion

# Pipeline (sub-ranging, half-flash) A/D Converter

• it reduces the number of bits to be converted into smaller groups, which are then run through a lower resolution flash converter.
• 100Msps at 8 to 14-bit resolutions
• compared to a flash converter, this approach reduces the number of comparators and logic complexity




Pipeline ADC architecture.



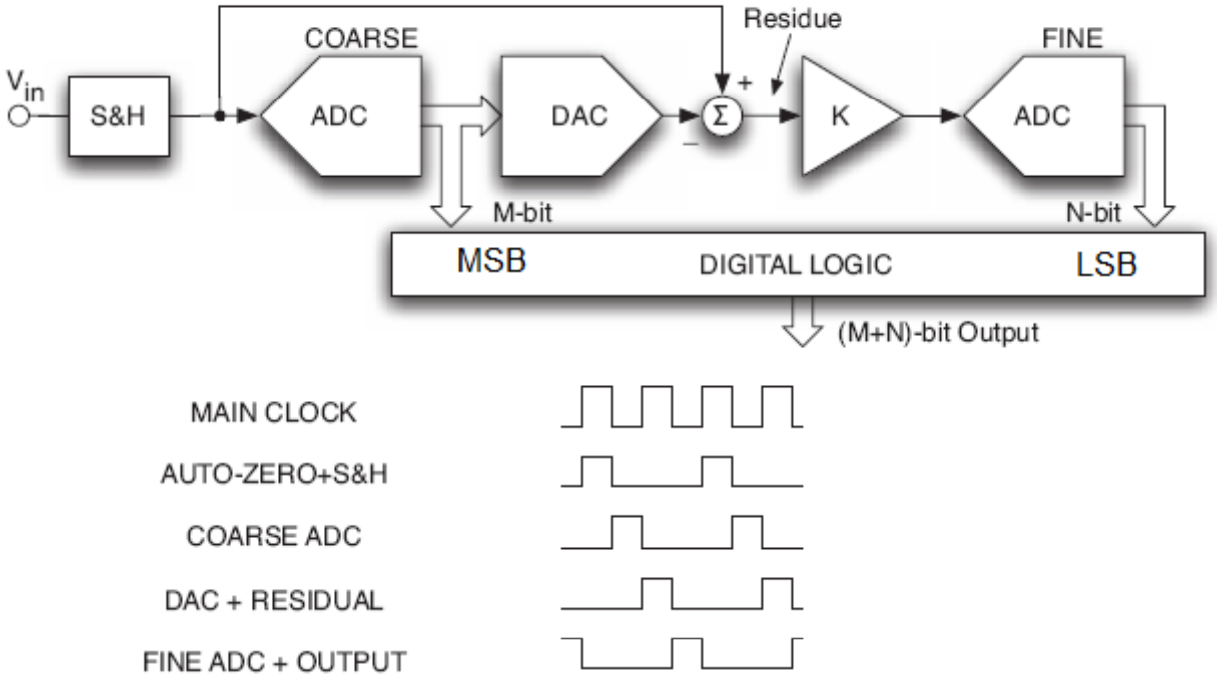

Pipeline ADC architecture.

• each stage resolves a few bits quickly and transfers the residue to the following stage so that the residue can be resolved further in the subsequent stages

• on each stage a DAC converts the output of the coarse ADC back to analog and subtracts from the input. The residue is then amplified.

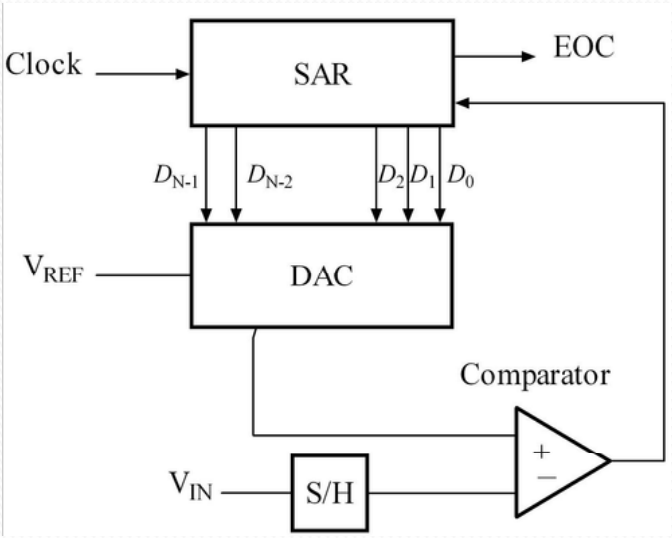
**⇒ latency time increases with number of stages K. Delay = (K+1) clock cycles**

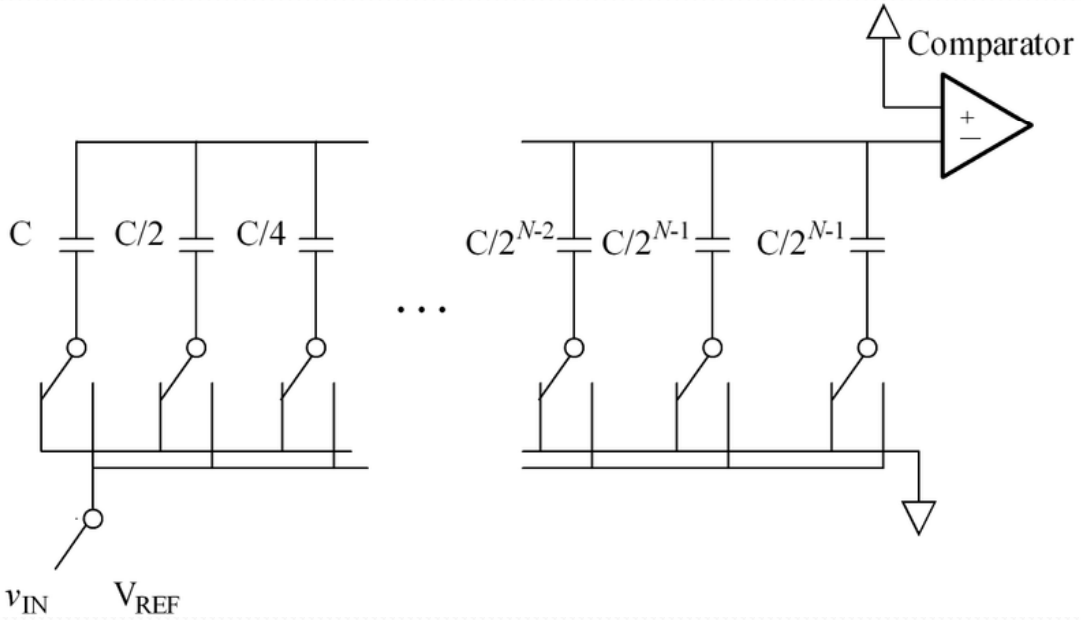
two-step algorithm: two clock periods, one for converting the MSBs and the other for the LSBs.



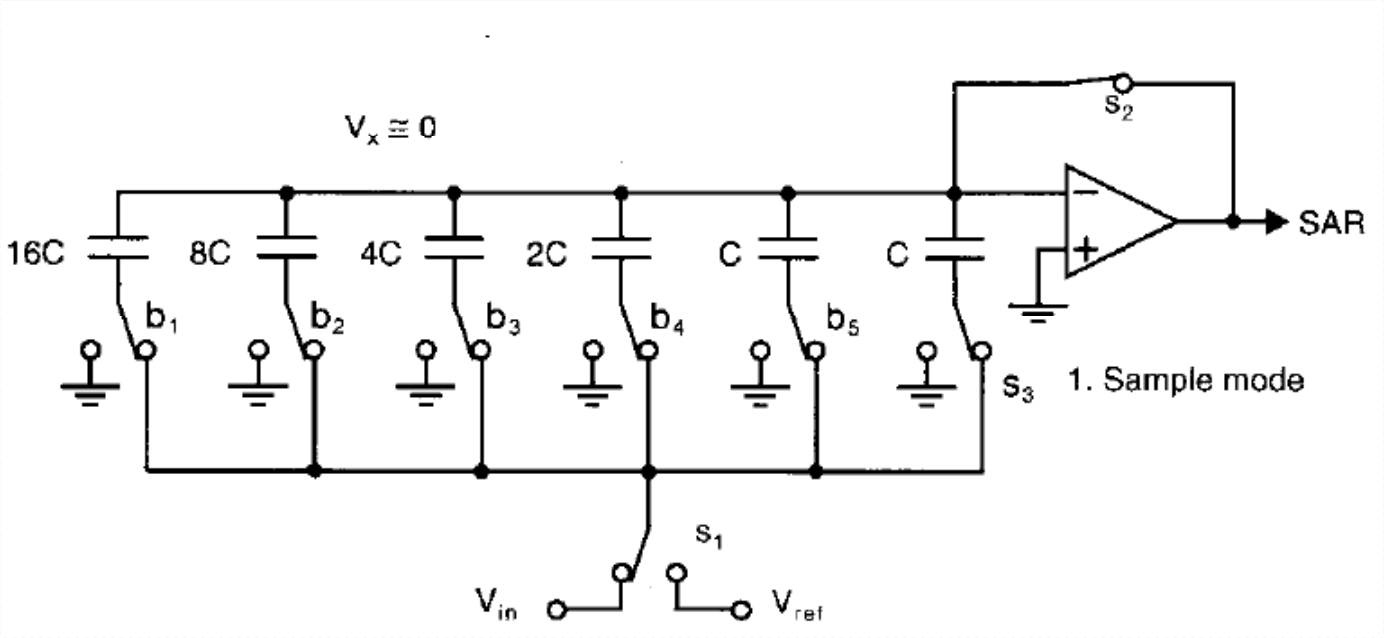

Block diagram of sub-ranging (K=1) and two-step architectures (K>1).

# Charge-Redistribution Successive Approximation ADC

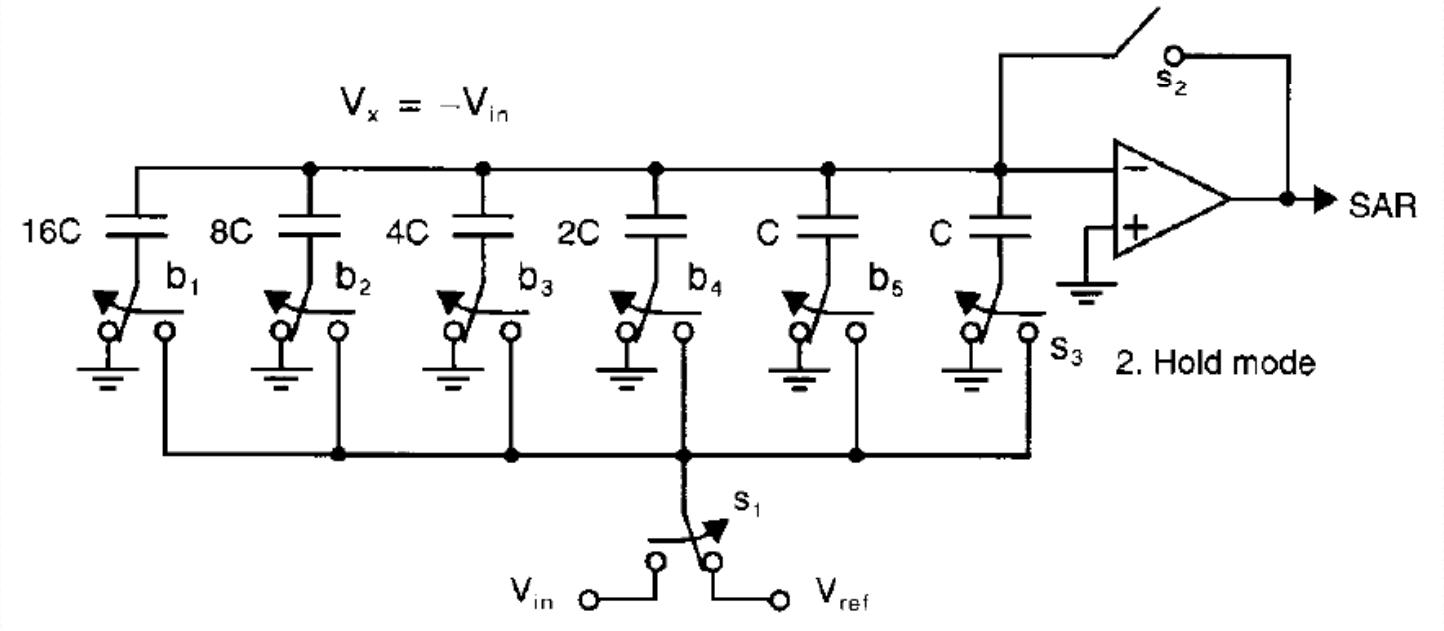


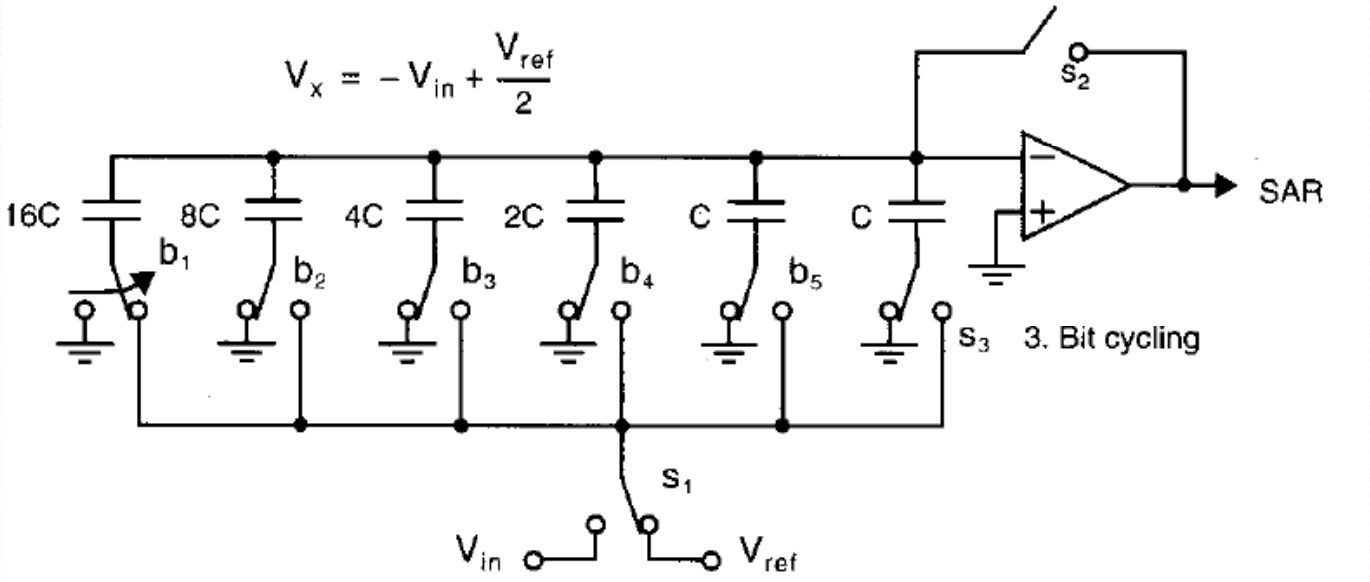
- Ex: 5-bit charge-redistribution A/D Converter



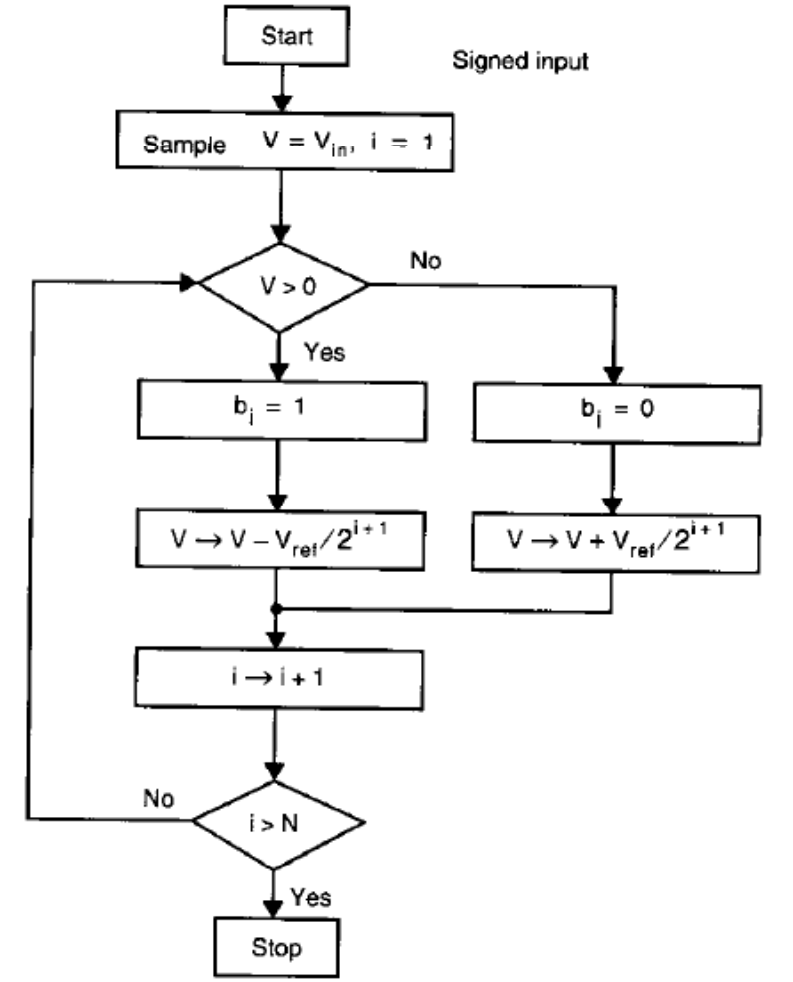

1. *Sample Mode*: in this first step, all capacitors are charged to Vin




2. *Hold Mode*: The comparator is activated by opening S2, and then all capacitors are switched to ground ⇒ Vx = -Vin
S1 is switched, so that VREF is applied to the capacitor array, during next step (bit cycling)

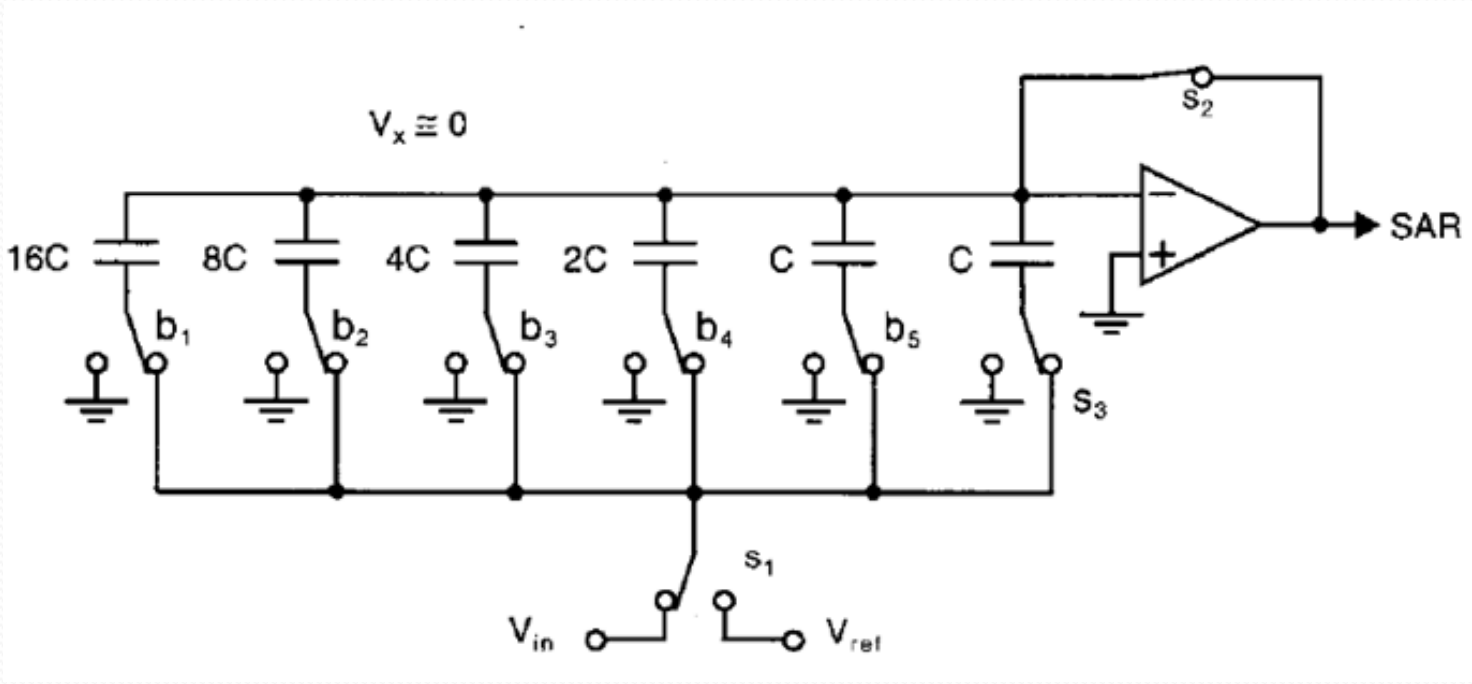



3. *Bit cycling*: The largest capacitor (16C in the example) is switched to VREF $\Rightarrow$ Vx = -Vin + VREF/2. If VX is negative, then Vin > VREF/2 and MSB capacitor (16C) remains connected to VREF, with b1 considred now =1. Otherwise, the MSC capacitor is reconnected to ground and b1 is taken as 0. This process is repeated N times, with a smaller capacitor being switched each time, until the conversion is finished.



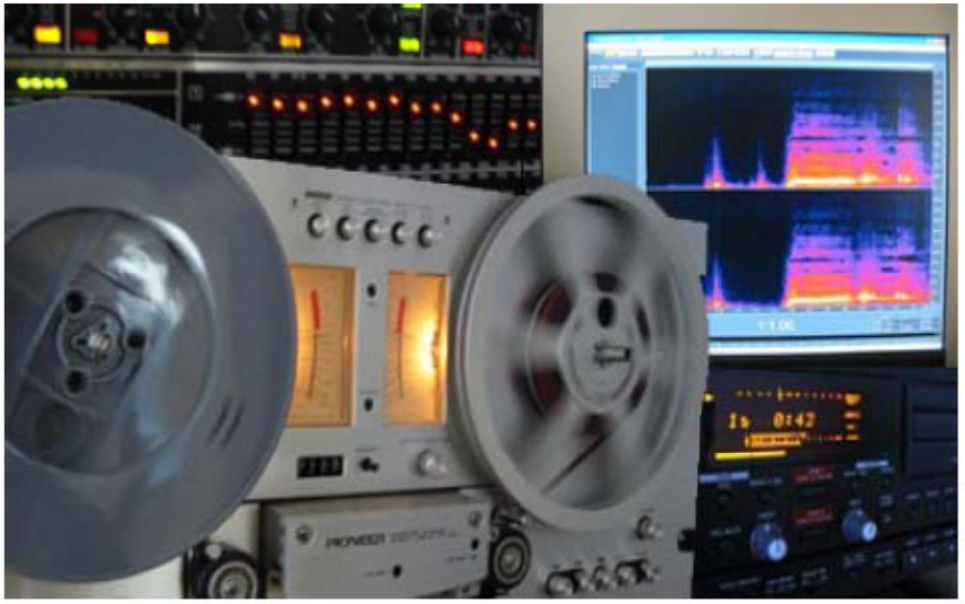

Flow graph for a successive approximation

Exercise: Find intermediate node voltages at Vx during the operation of the 5-bit DAC below, when Vin = 1.23V and VREF = 5V. Assume a parasitic capacitance of 8C exists on the node at Vx.

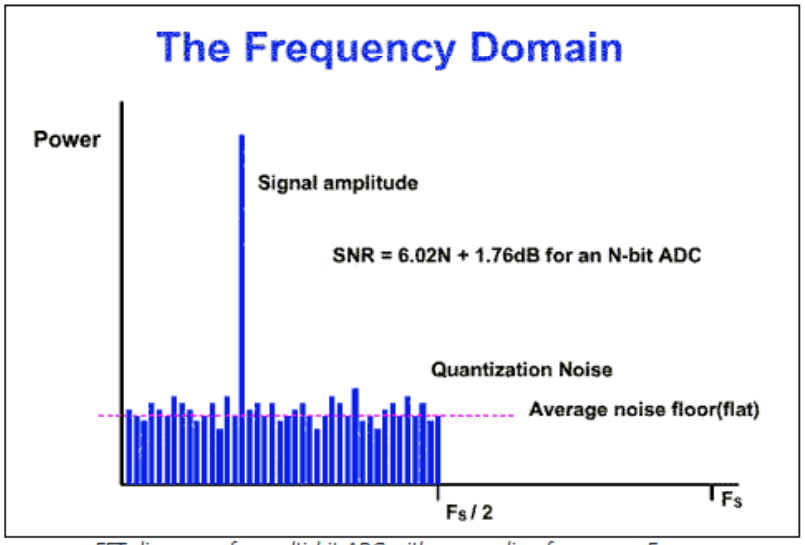


# Σ–Δ A/D Converters

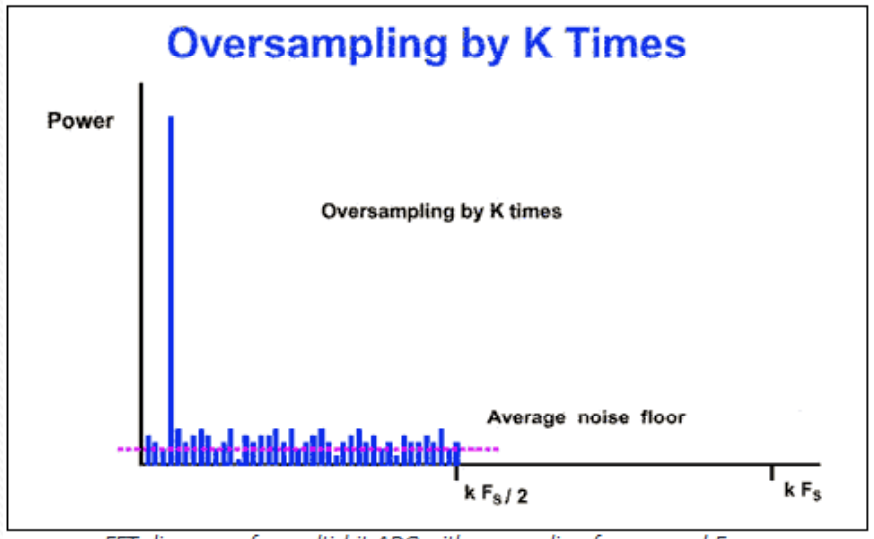
- Applications:
  - High-resolution needed applications;
  - Audio, video, medical applications and so forth.



- Nyquist rate $f_N = 2B$
- Oversampling rate $M = f_s/f_N >> 1$




*FFT diagram of a multi-bit ADC with a sampling frequency $F_S$.*

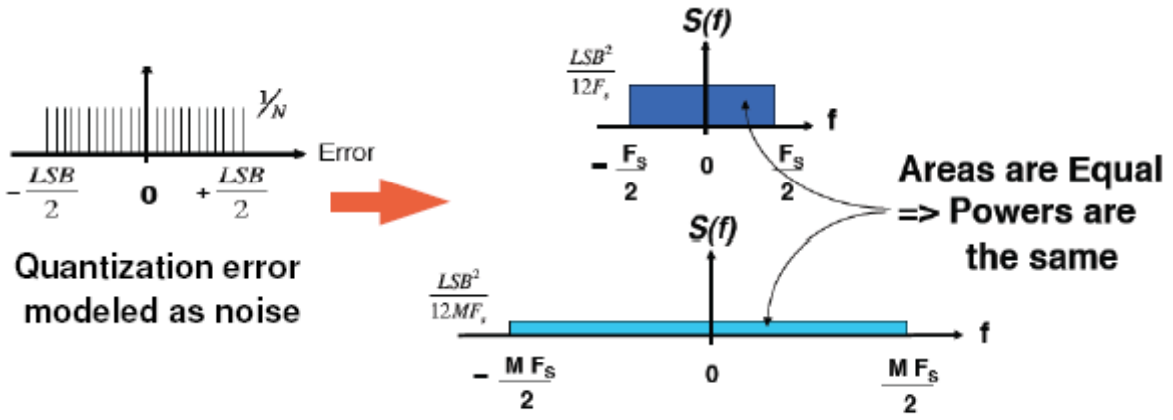



*FFT diagram of a multi-bit ADC with a sampling frequency $kF_S$.*

noise energy spread over a wider frequency range

$$SNR = 6.02N + 1.76dB$$

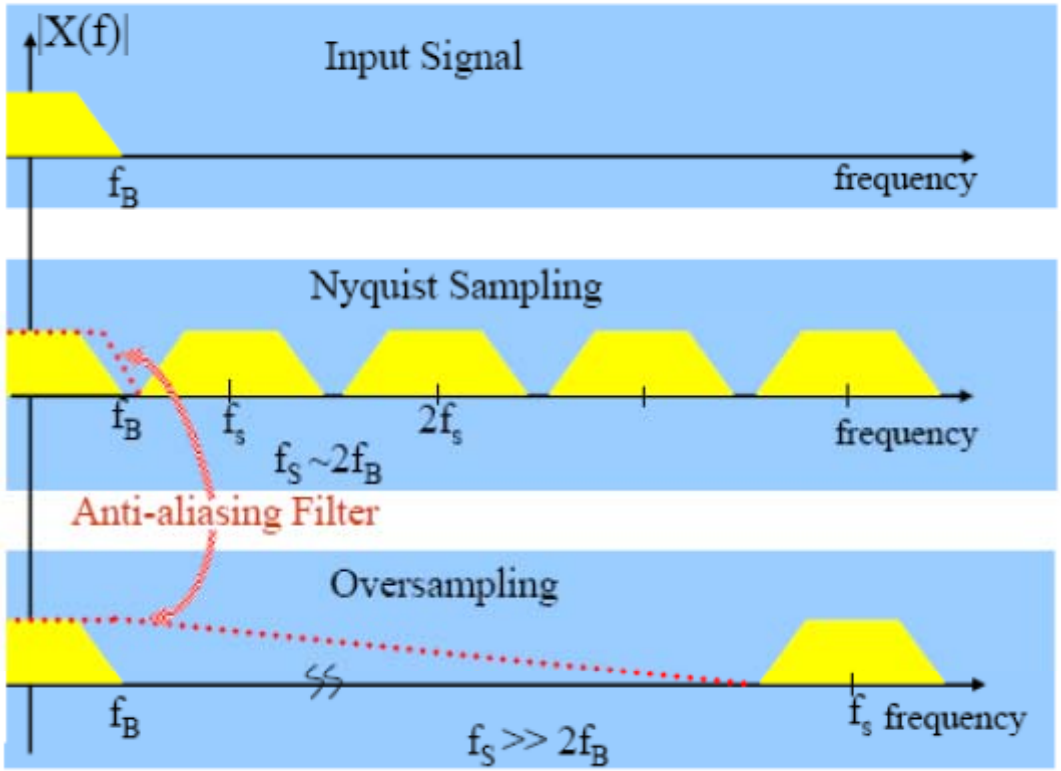
## Oversampling

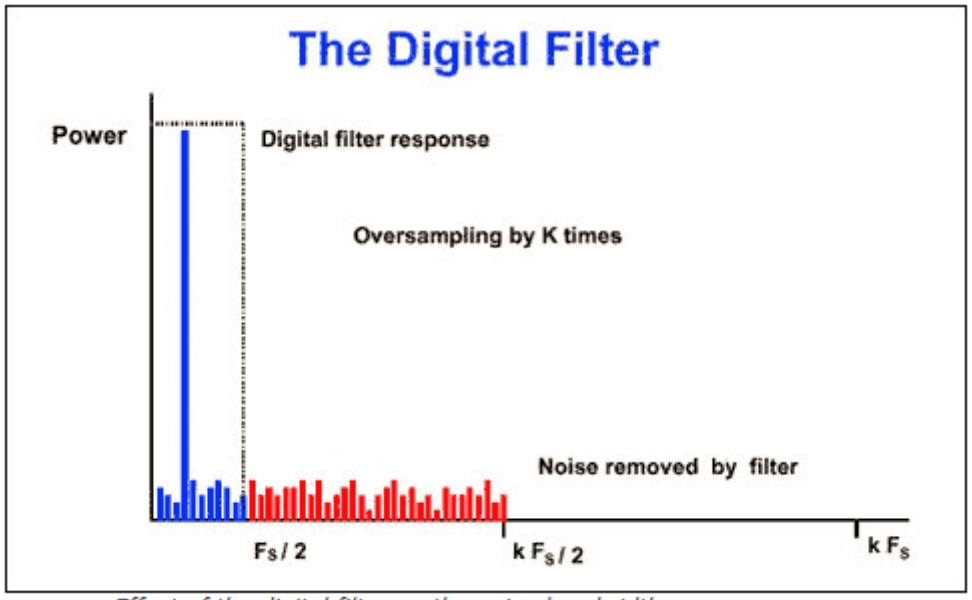



$$M = \frac{F_s/2}{f_{3dB}}: \text{ Oversampling Ratio (OSR)}$$

## Anti-aliasing Filter

Relaxed requirements with oversampling





*Effect of the digital filter on the noise bandwidth.*

• each factor-of-4 oversampling increases the SNR by 6dB, and each 6dB increase is equivalent to gaining one bit.

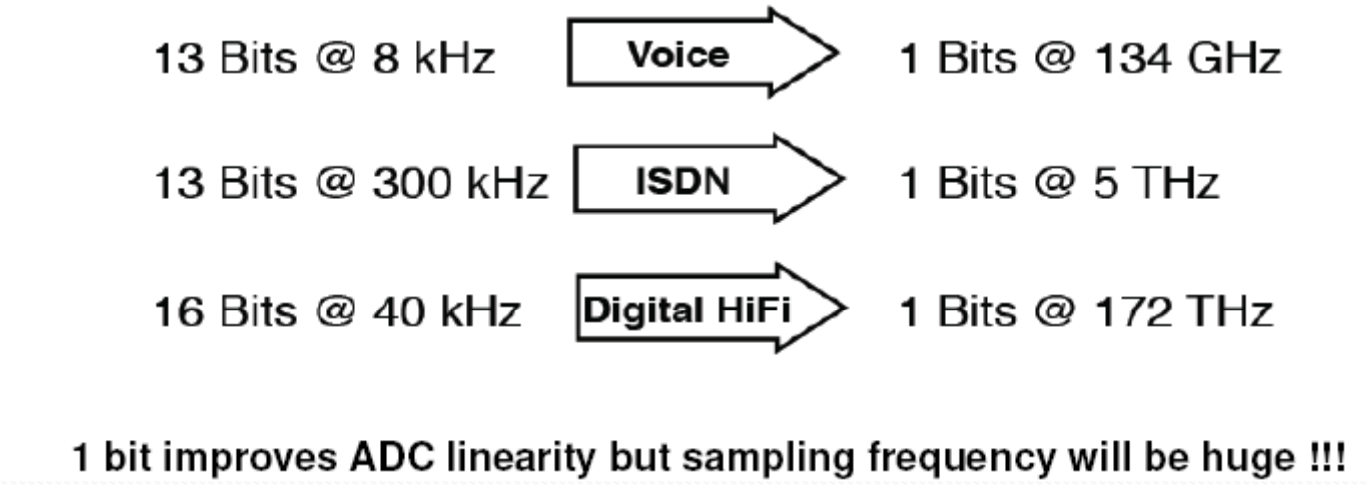
•A 1-bit ADC with 24x oversampling achieves a resolution of four bits

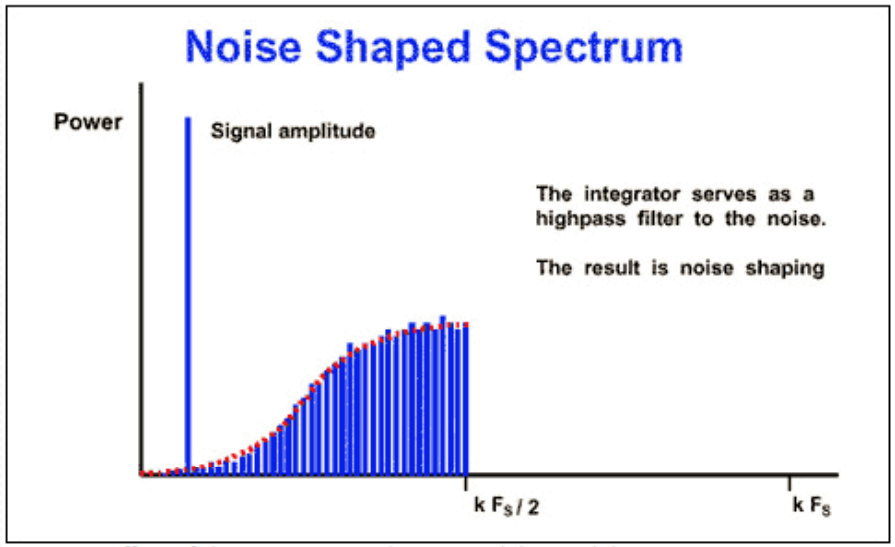
• to achieve 16-bit resolution , one must oversample by a factor of $4^{15}$, which is not realizable $\Rightarrow$ **noise shaping technique**

without noise shaping:

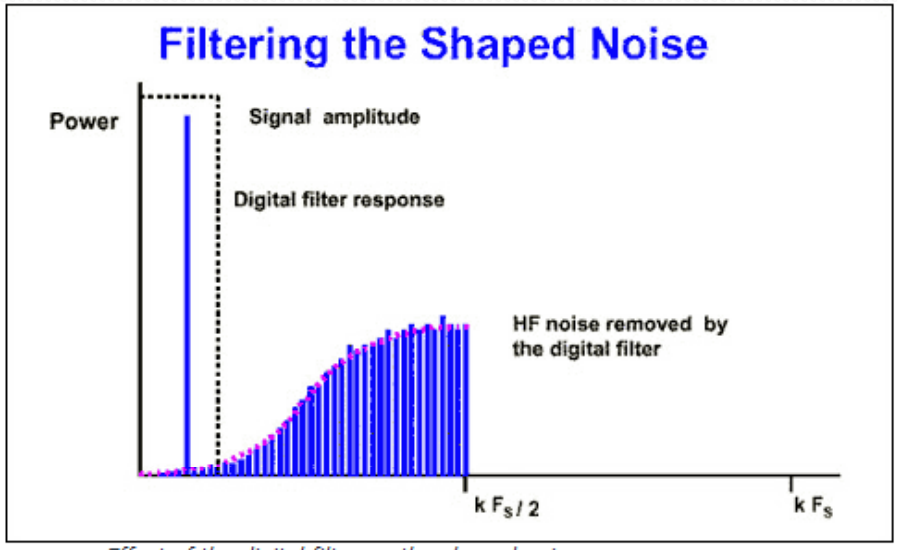
$$SNR = 6.02N + 1.76 + 10\log(OSR) \qquad \text{dB}$$

4x OSR → 1bit resolution

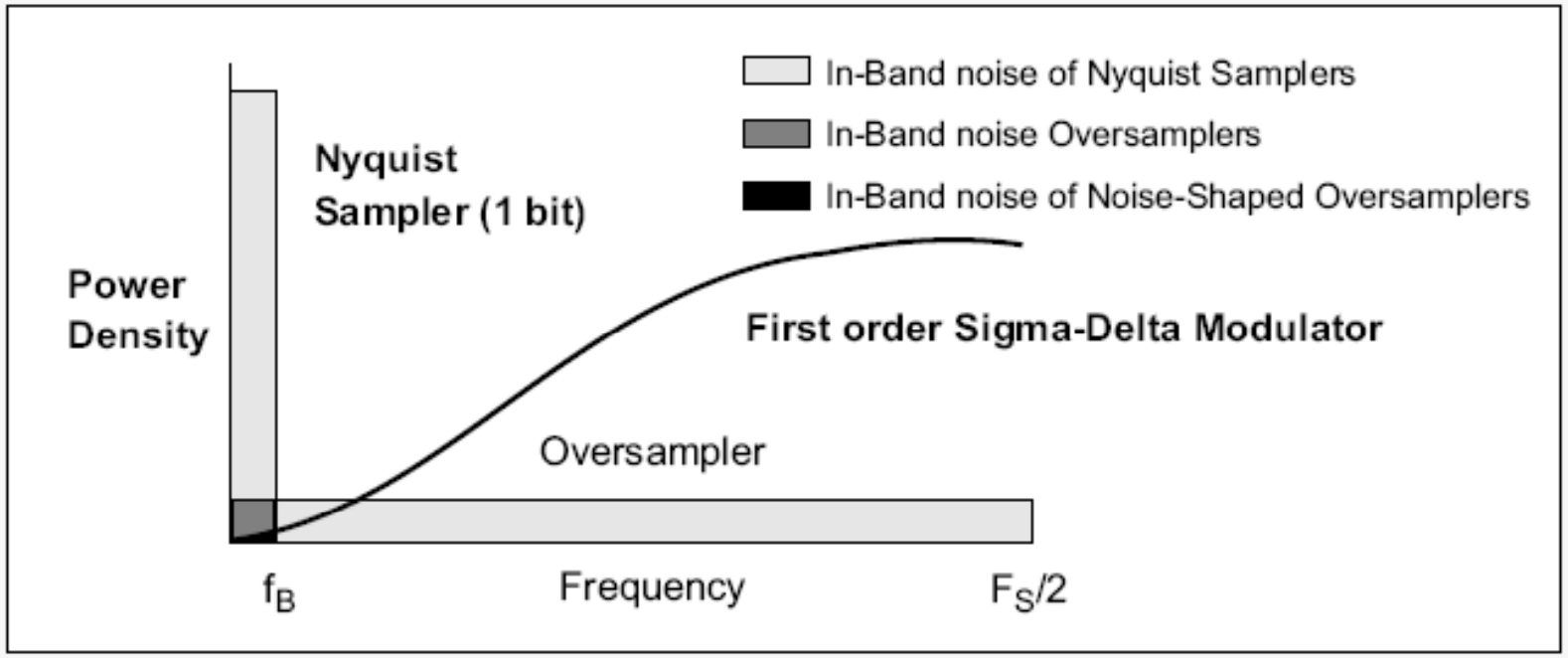





*Affect of the integrator in the sigma-delta modulator.*




*Effect of the digital filter on the shaped noise.*



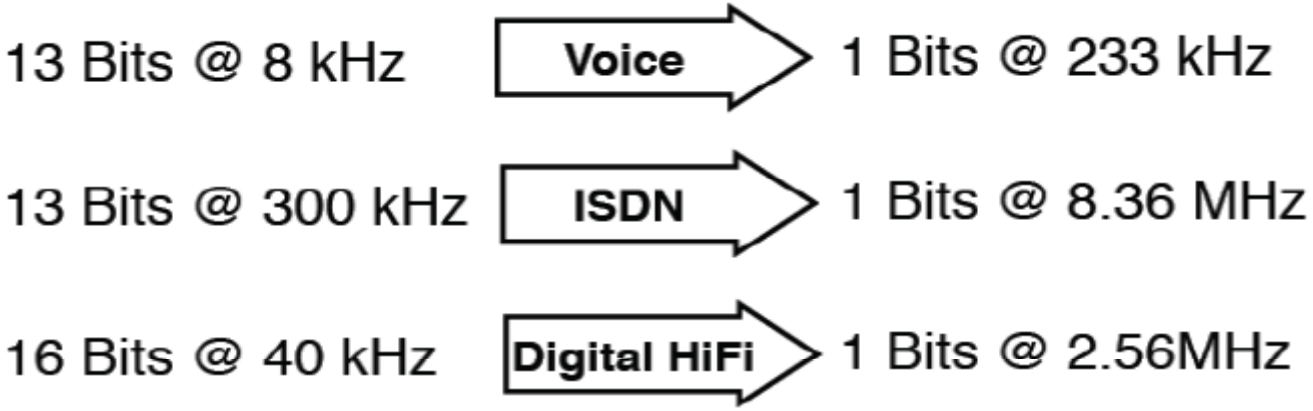

*Spectrum of a First-Order Sigma-Delta Noise Shaper*

with noise shaping (first-order loop):

$$SNR = 6.02N + 1.76 - 5.17 + 9.03\log_2(OSR) \quad \text{dB}$$

$$SNR = 6.02N + 1.76 - 5.17 + 30\log_{10}(OSR) \quad \text{dB}$$

2x OSR → 1.5bit resolution

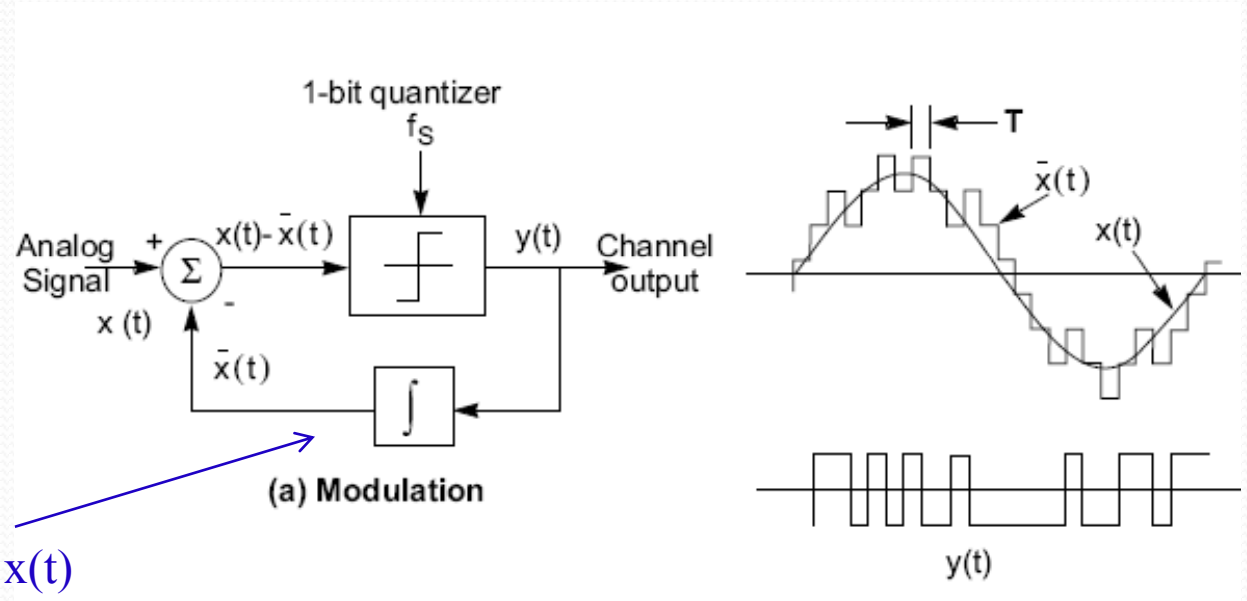


**summary**:

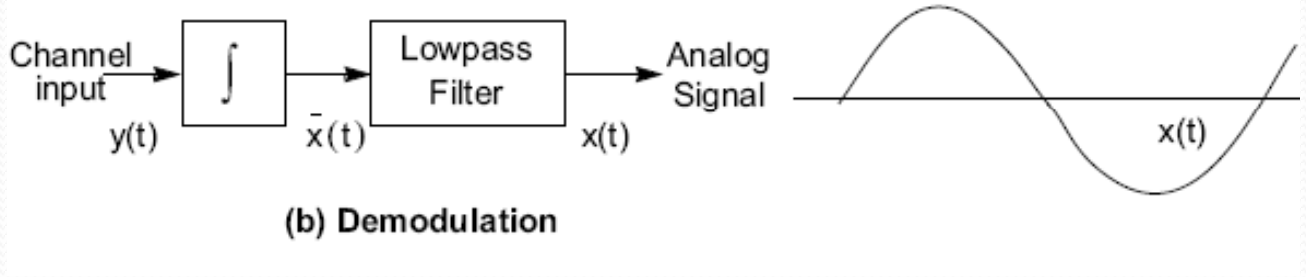
Why ∑Δ?

- Quantization Noise reduction in baseband.
- High-resolution.
- Anti-aliasing filter relaxing.
- Analog blocks flexibility.

# Delta modulation and demodulation




Predictor of x(t)

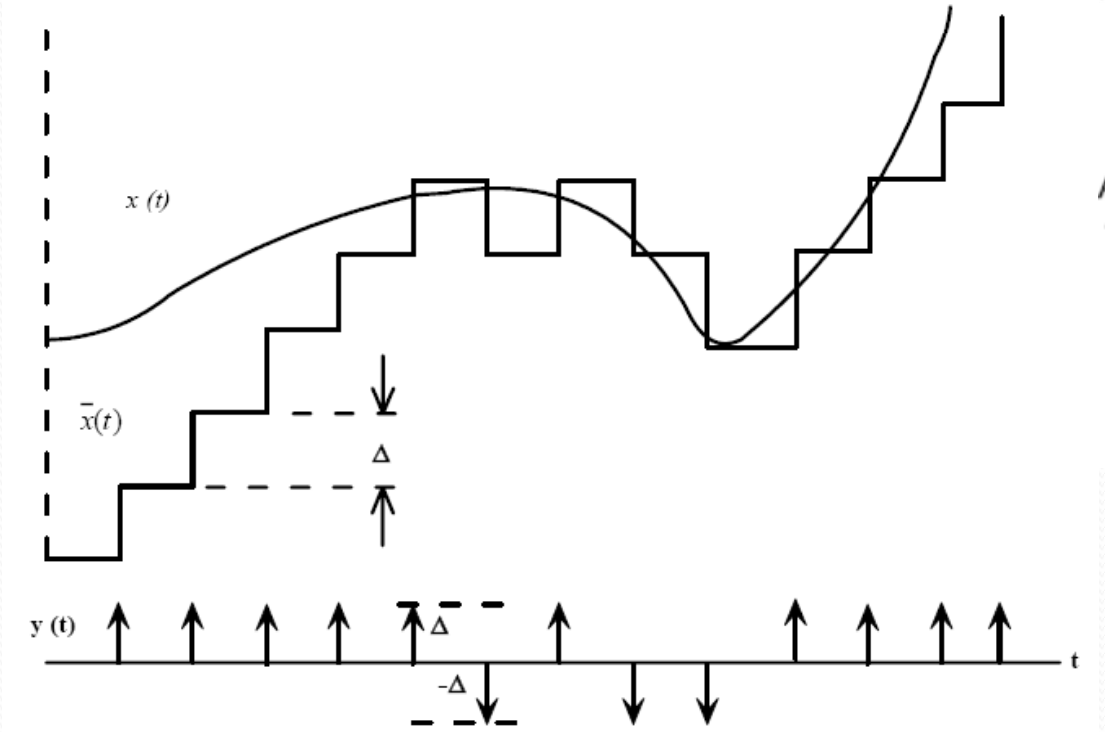


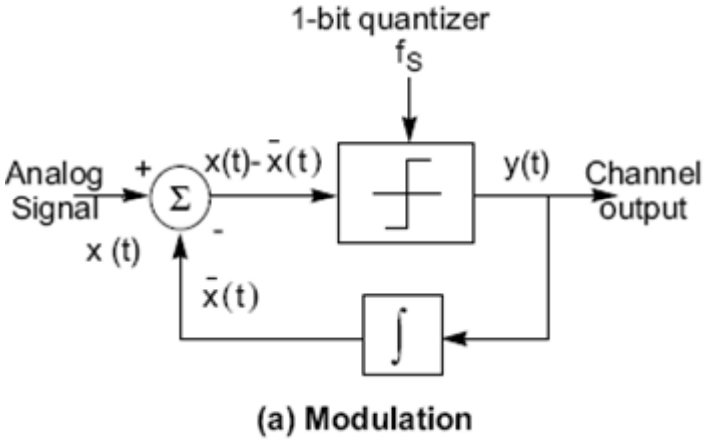
• Delta modulation is based on quantizing the signal amplitude change from sample to sample rather its absolute value (rate of change)

• since the integrator output tries to predict the input x(t), the integrator works as a predictor.

• prediction-error term $x(t) - \overline{x(t)}$ is quantized and used to make the next prediction.

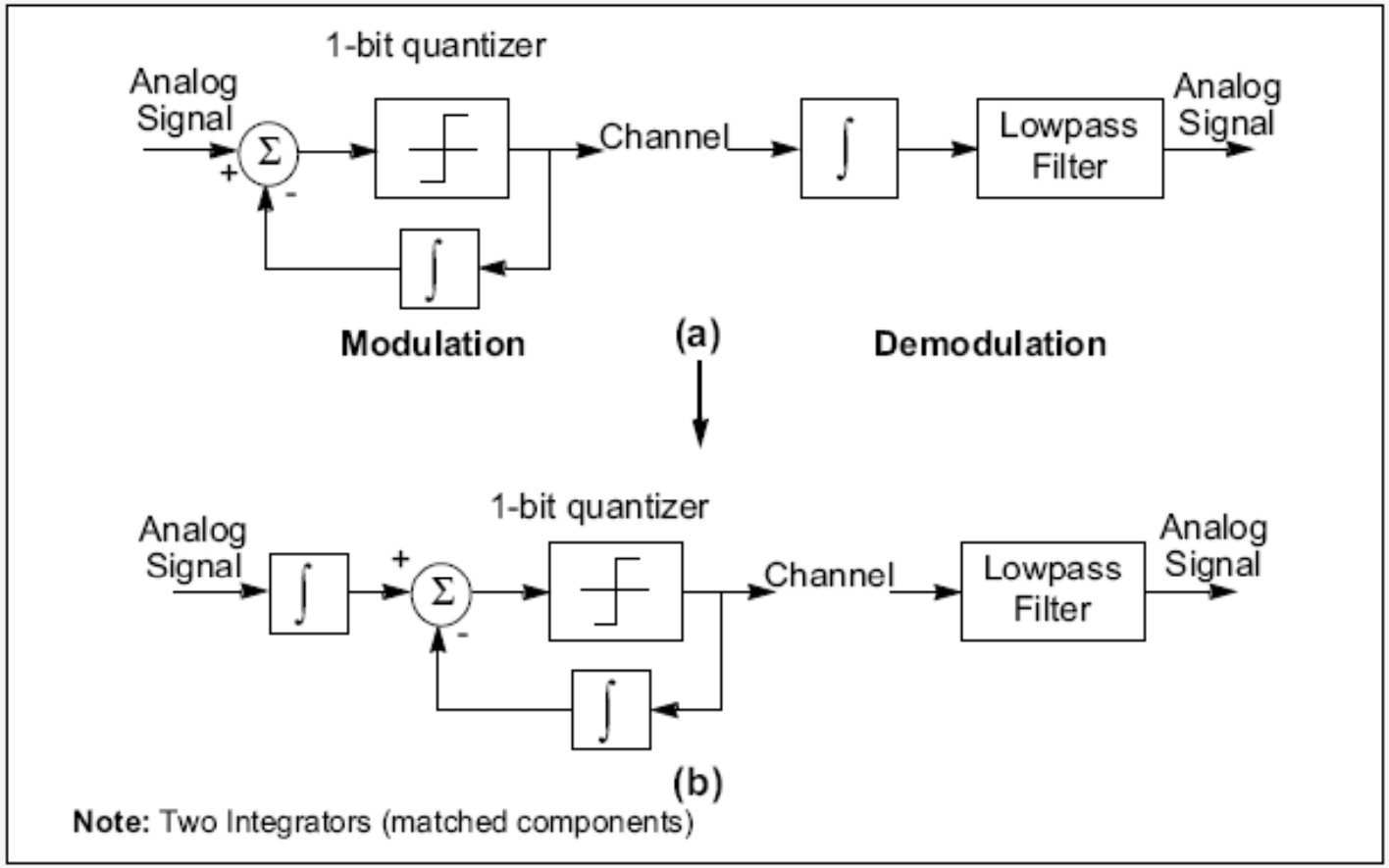
• prediction error (delta modulation output) is integrated in the receiver just as it is in the feedback loop $\Rightarrow$ receiver predicts the input signals



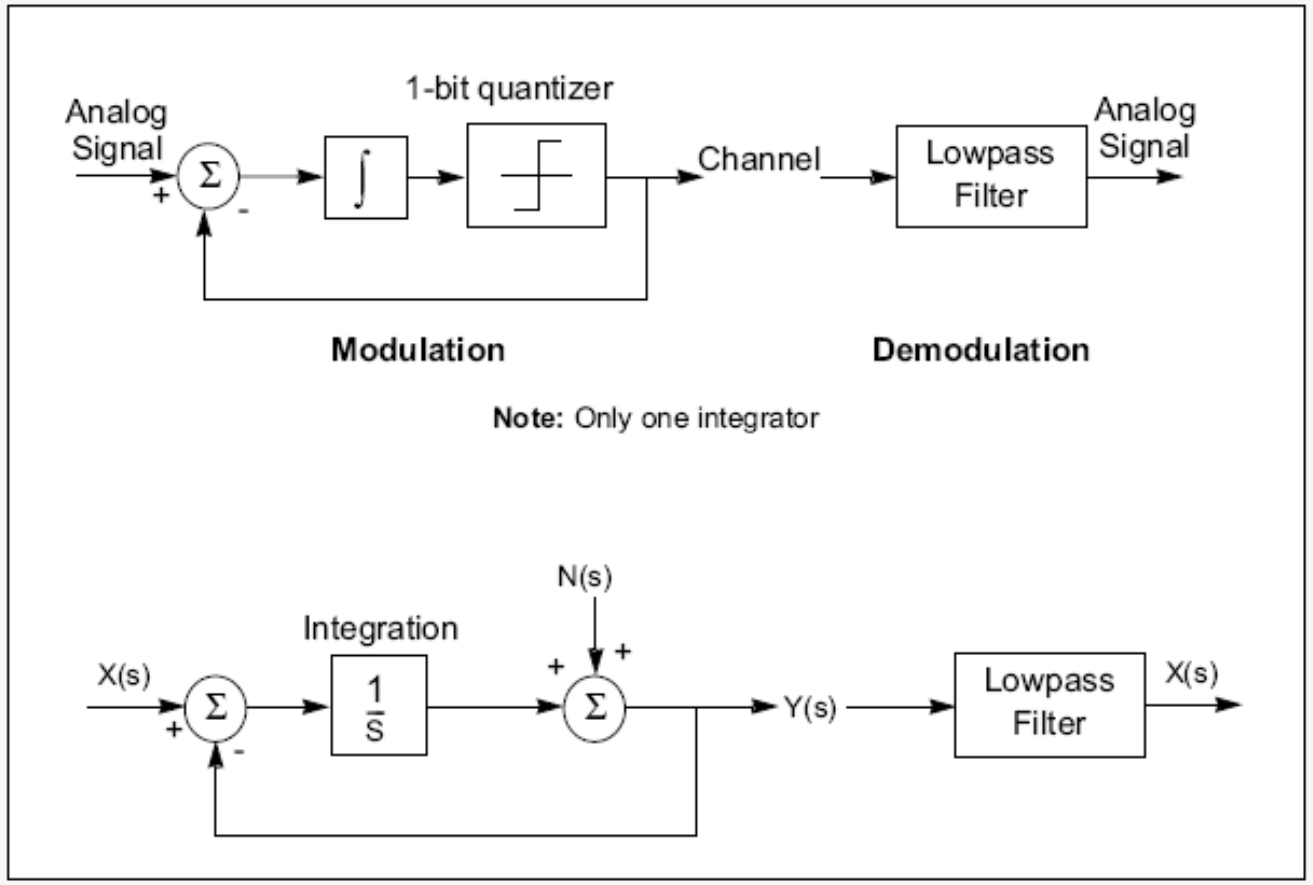




(a) Modulation

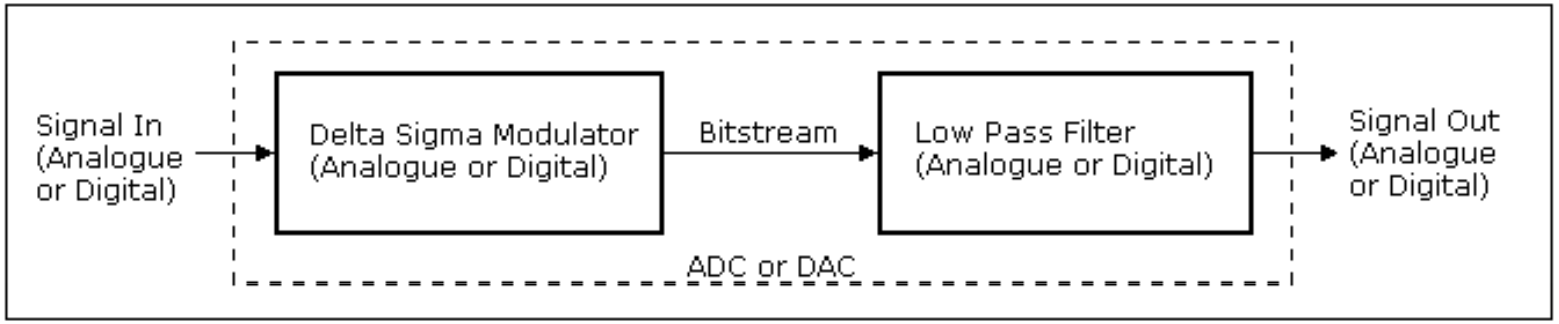
# Σ–Δ modulation




*Derivation of Sigma-Delta Modulation from Delta Modulation*

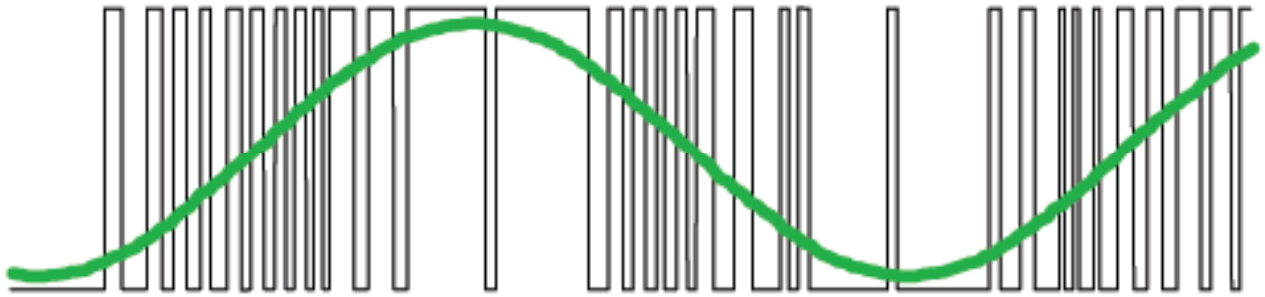


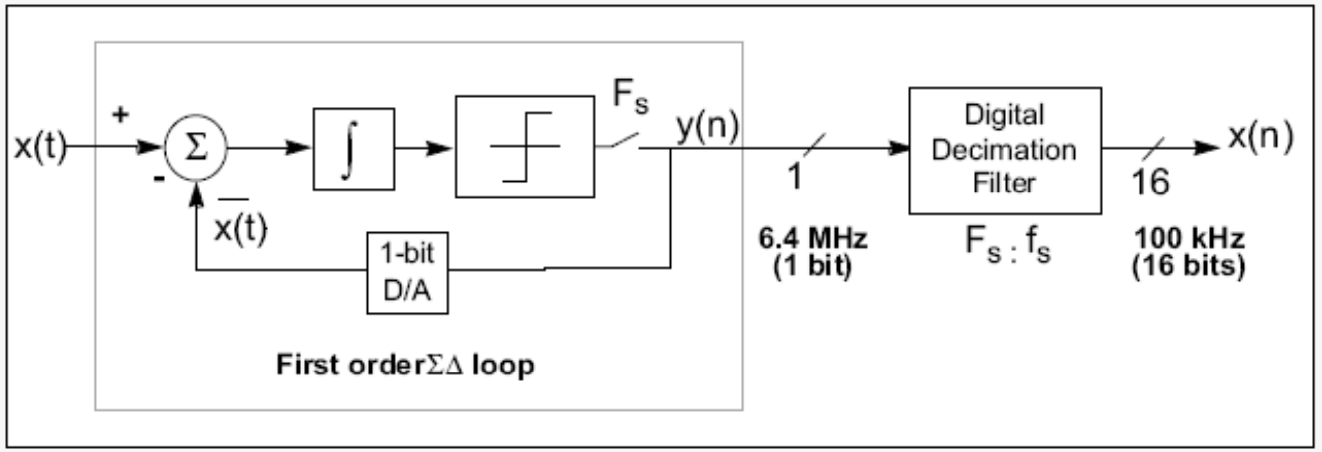

*Block Diagram of Signa-Delta Modulation*




**Block Diagram of a Delta Sigma Converter**

• The bitstream is a one-bit serial signal with a very high bit rate. Its major property is that **its average level represents the average input signal level**.





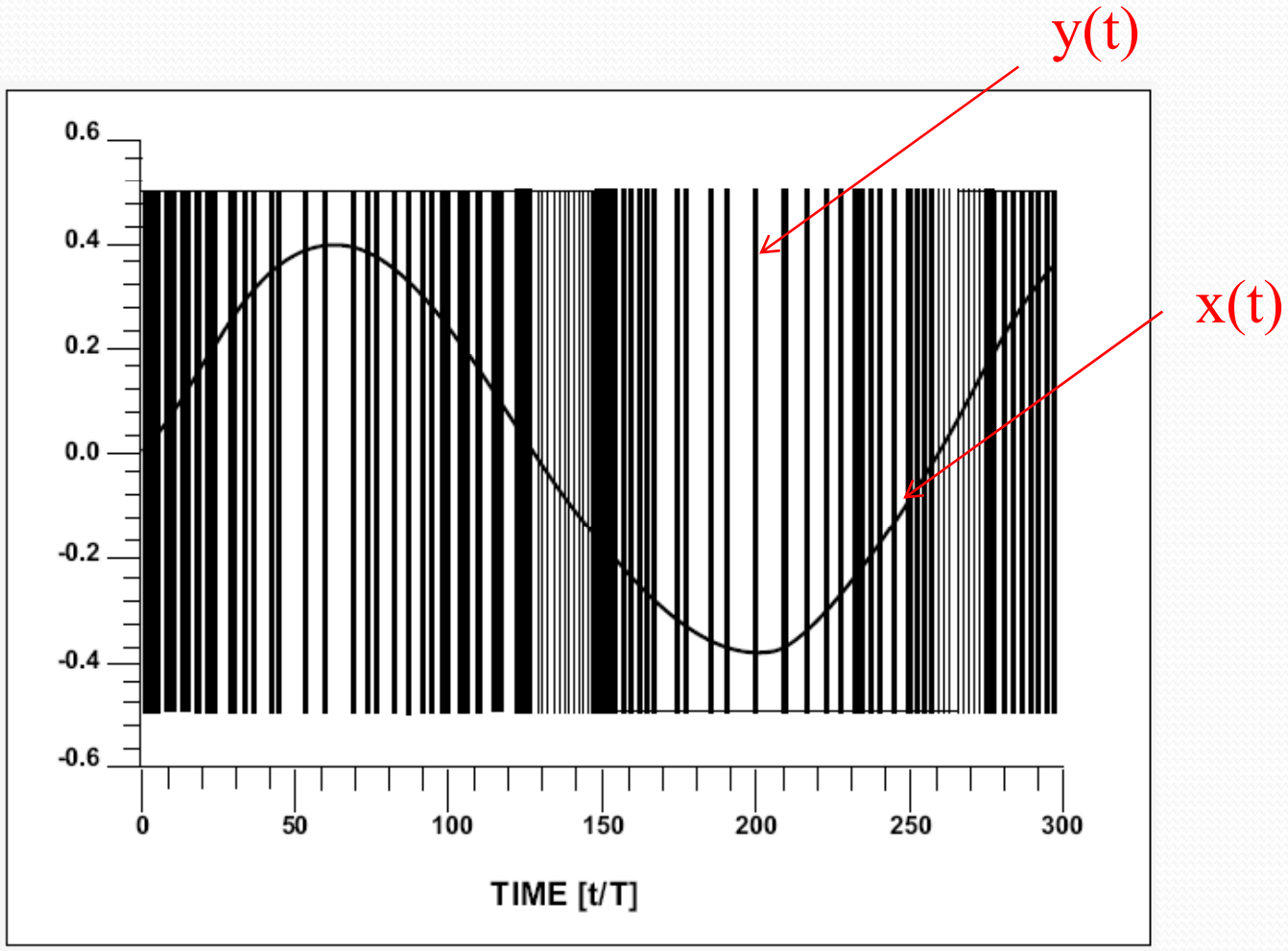

*Block Diagram of First-Order Sigma-Delta A/D converter*

• the term Sigma-Delta comes from putting the integrator (sigma) in front of the delta modulator. Σ–Δ modulator produces a bitstream, **whose average level represents the input signal level.**

• input to the integrator is the difference between the input signal x(t) and the quantized output value y(n), converted back to the predicted analog signal, $\overline{x}(t)$.

• difference x(t) - $\overline{x}(t)$ at the integrator input is equal to the quantization error.

• error is summed up in the integrator and then quantized by the 1-bit ADC

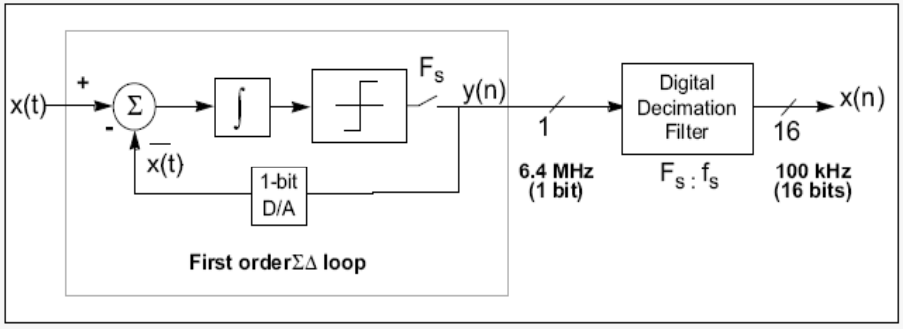


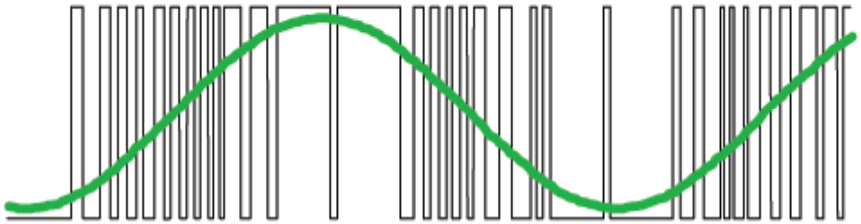

*Input and Output of a First-Order Sigma-Delta Modulator*

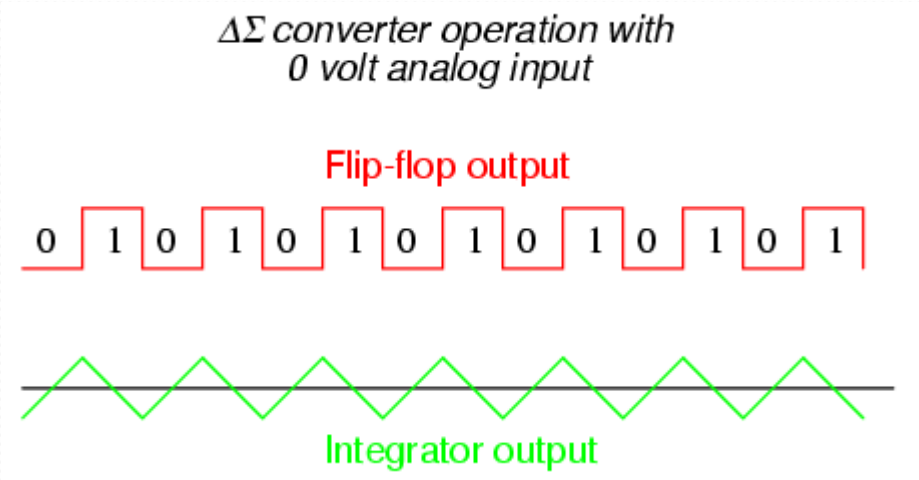
• in each clock cycle, output of the modulator is either plus or minus full scale, according to the results of the 1-bit A/D conversion

• when the sinusoidal x(t) is close to a plus full scale, the output is positive during most clock cycles. Conversely, when x(t) is close to minus full scale, output is mostly negative. In both cases, the local average of the modulator output tracks the analog input.

• When x(t) is near zero, the modulator output varies rapidly between a plus and a minus full scale with approximately zero mean.

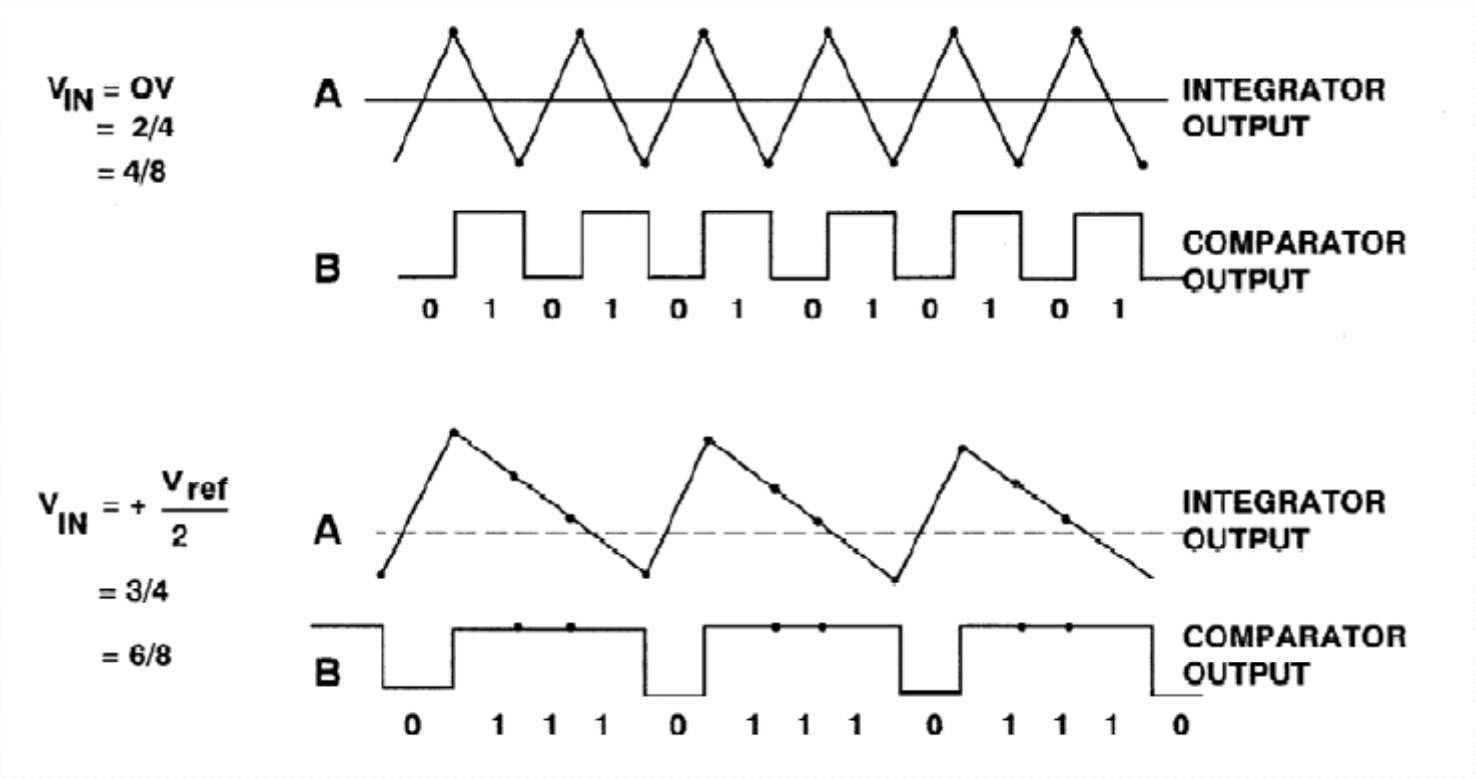


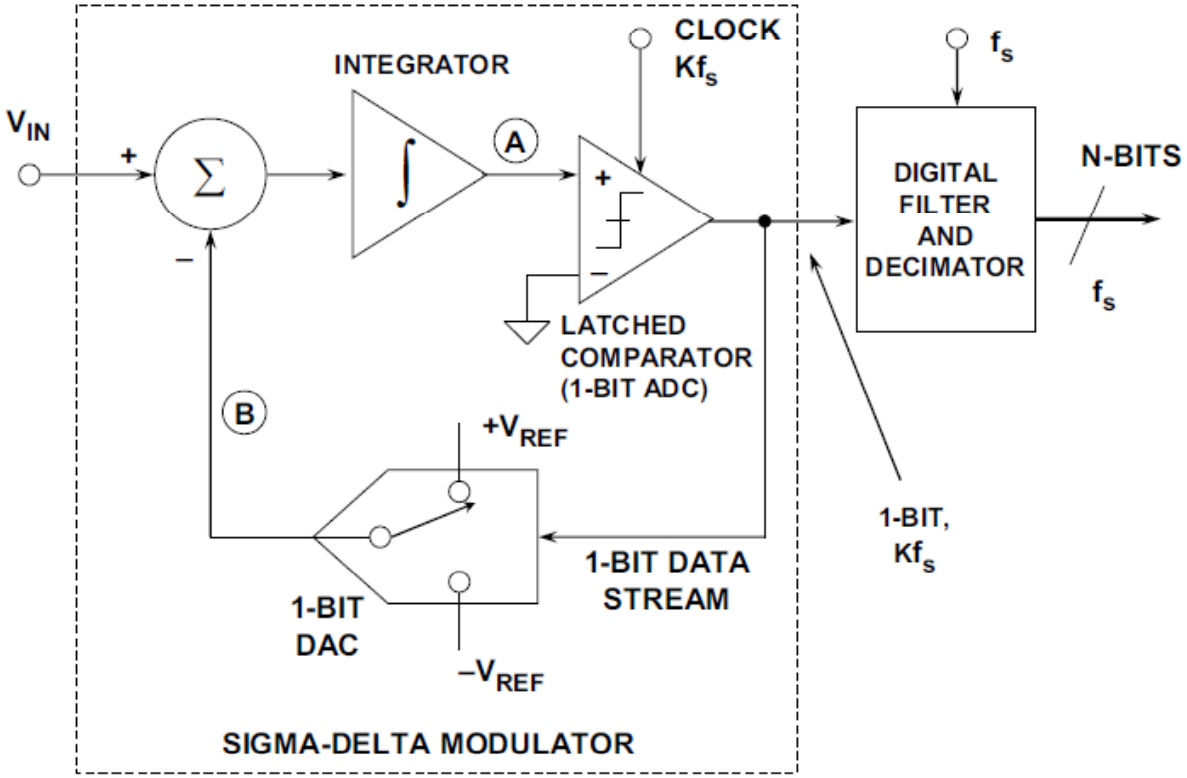



• if the analog input is 0V (ac), the integrator will have no tendency to ramp either positive or negative, except in response to the feedback voltage

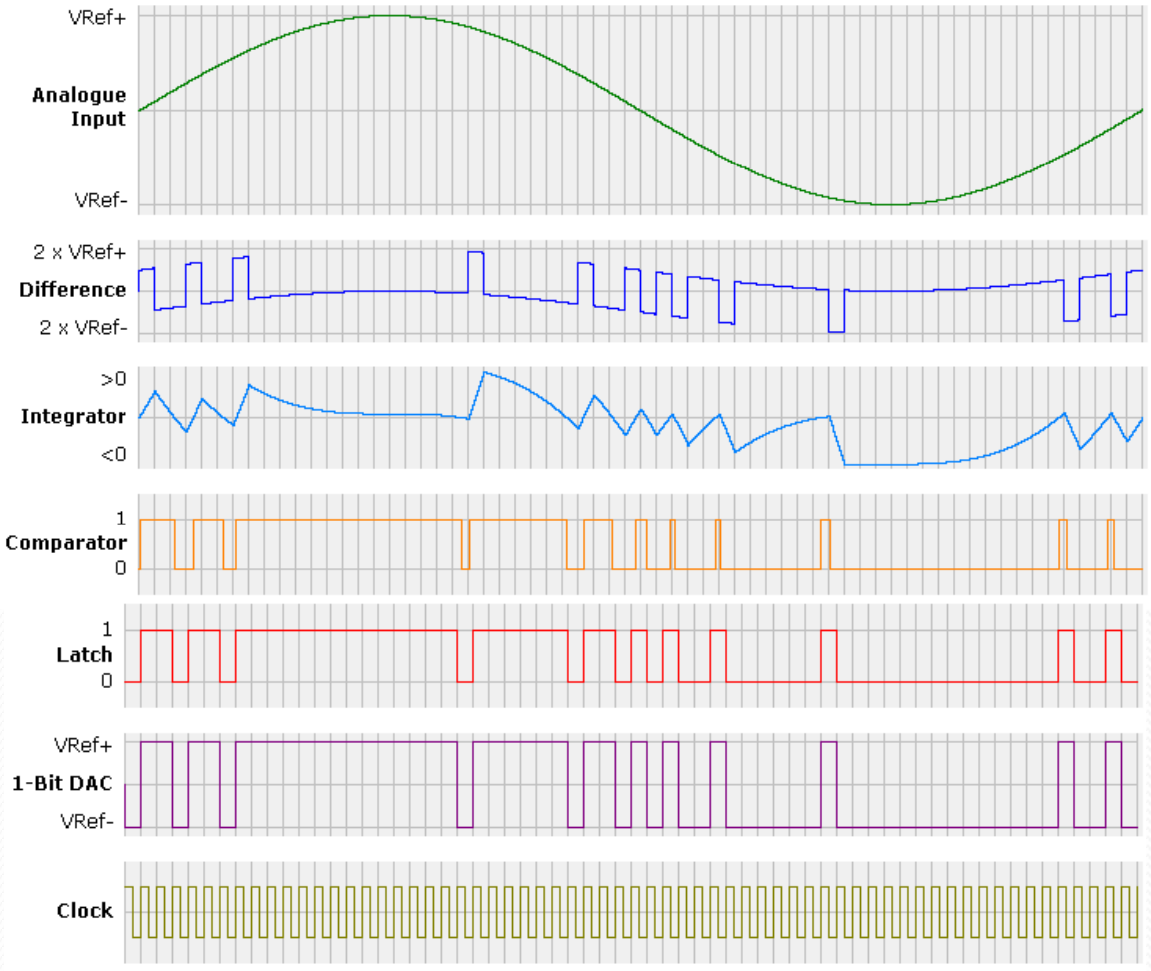
$\Rightarrow$ FF output will continually oscillate between "high" and "low," as the feedback system goes back and forth, trying to maintain the integrator output at 0V.

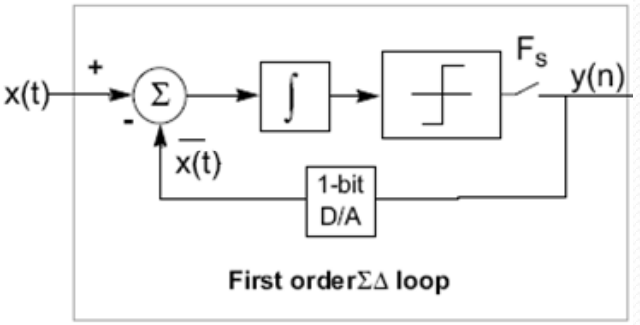



- averaging 4 samples gives 2 bits of resolution
- averaging 8 samples gives 3 bits of resolution

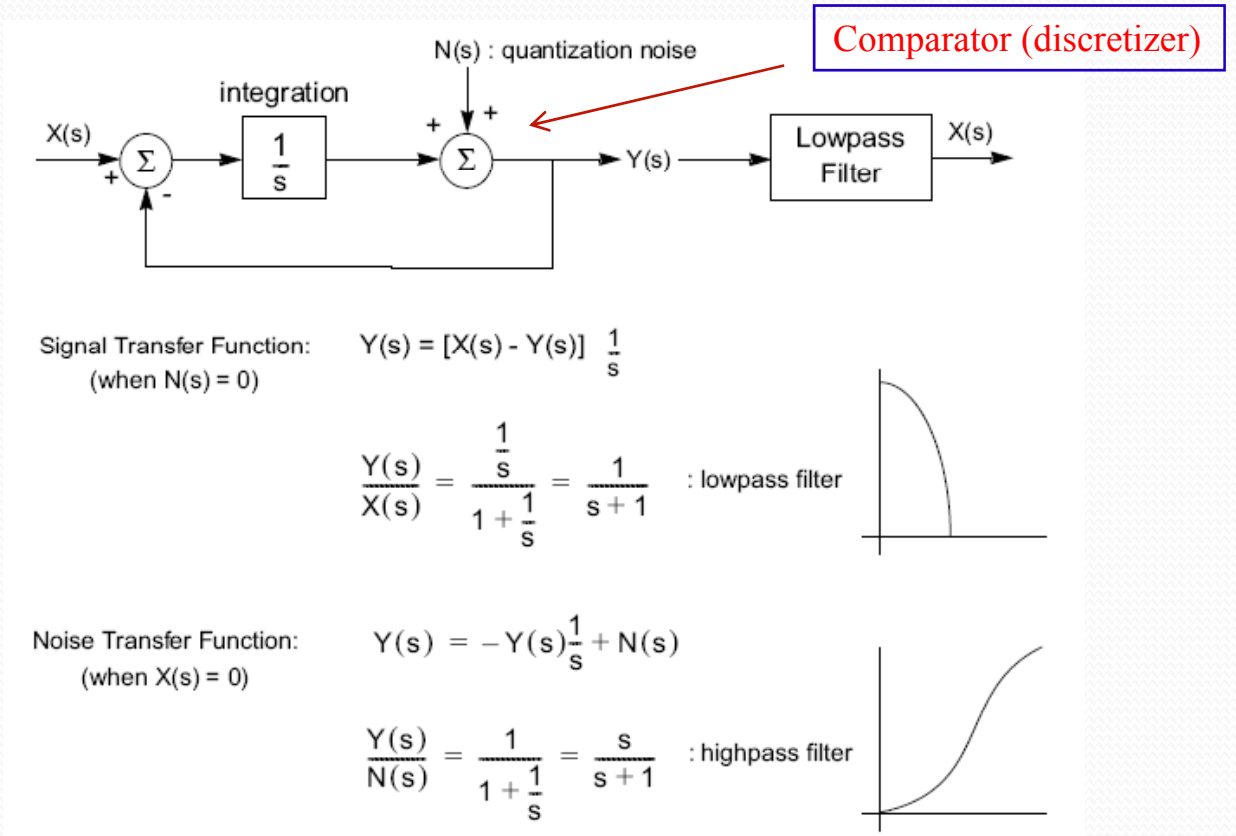



*First-Order Sigma-Delta ADC*



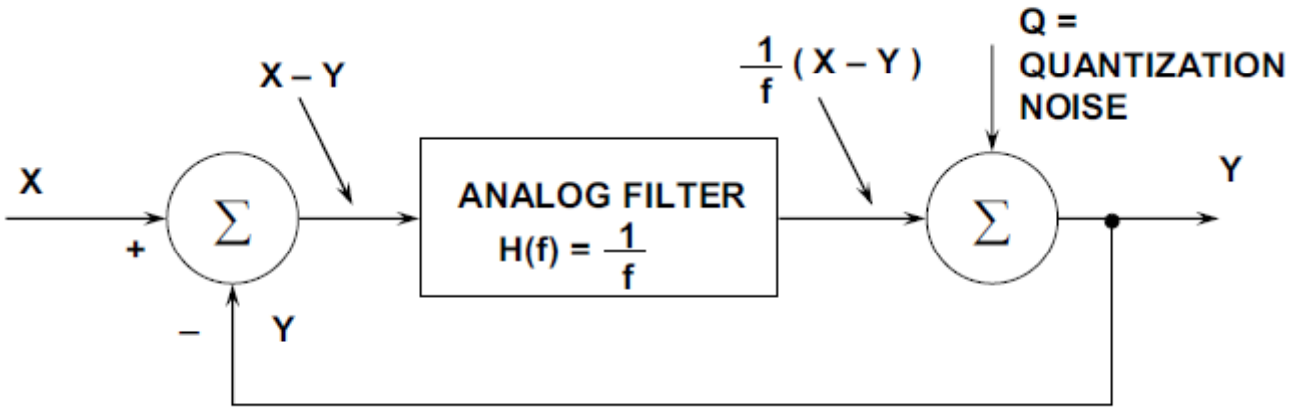




**Signals within a First Order Analogue Modulator**




- loop integrates the error between the sampled signal and the input signal
- signal is low-passed and noise is high-passed

$\Rightarrow$ the signal is left unchanged as long as its bandwidth doesn't exceed the filter's cutoff frequency, but the $\Sigma-\Delta$ loop pushes the noise into higher frequencies

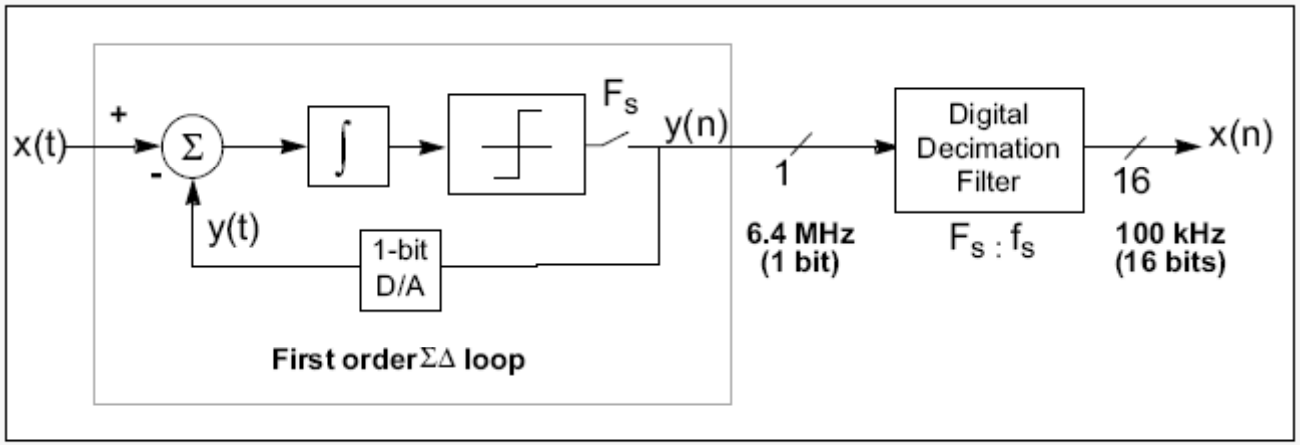



$$Y = \frac{1}{f}(X - Y) + Q$$

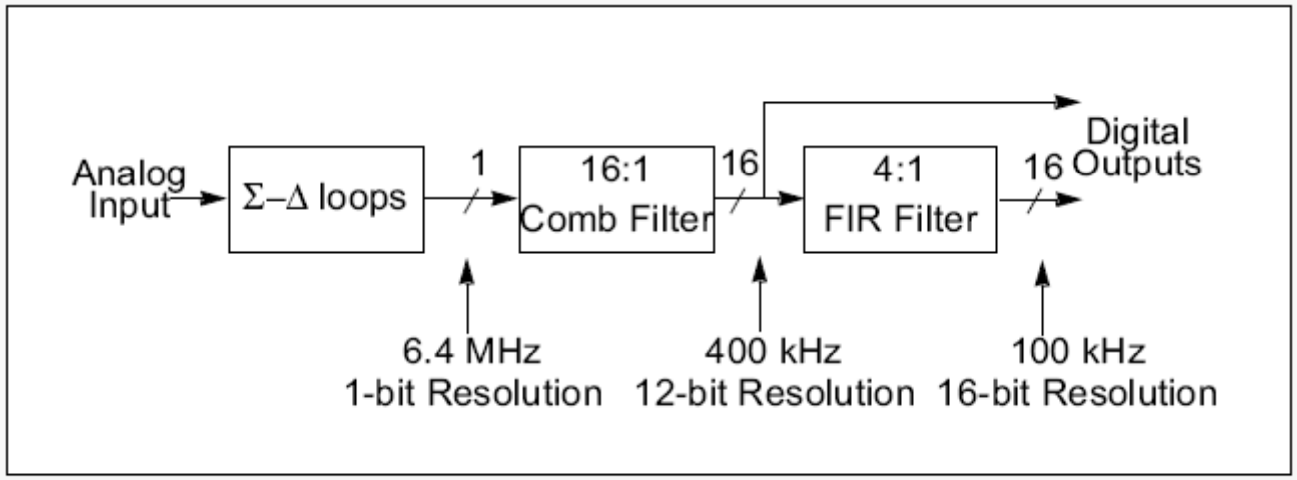
REARRANGING, SOLVING FOR Y:

$$Y = \frac{X}{f+1} + \frac{Q\,f}{f+1}$$

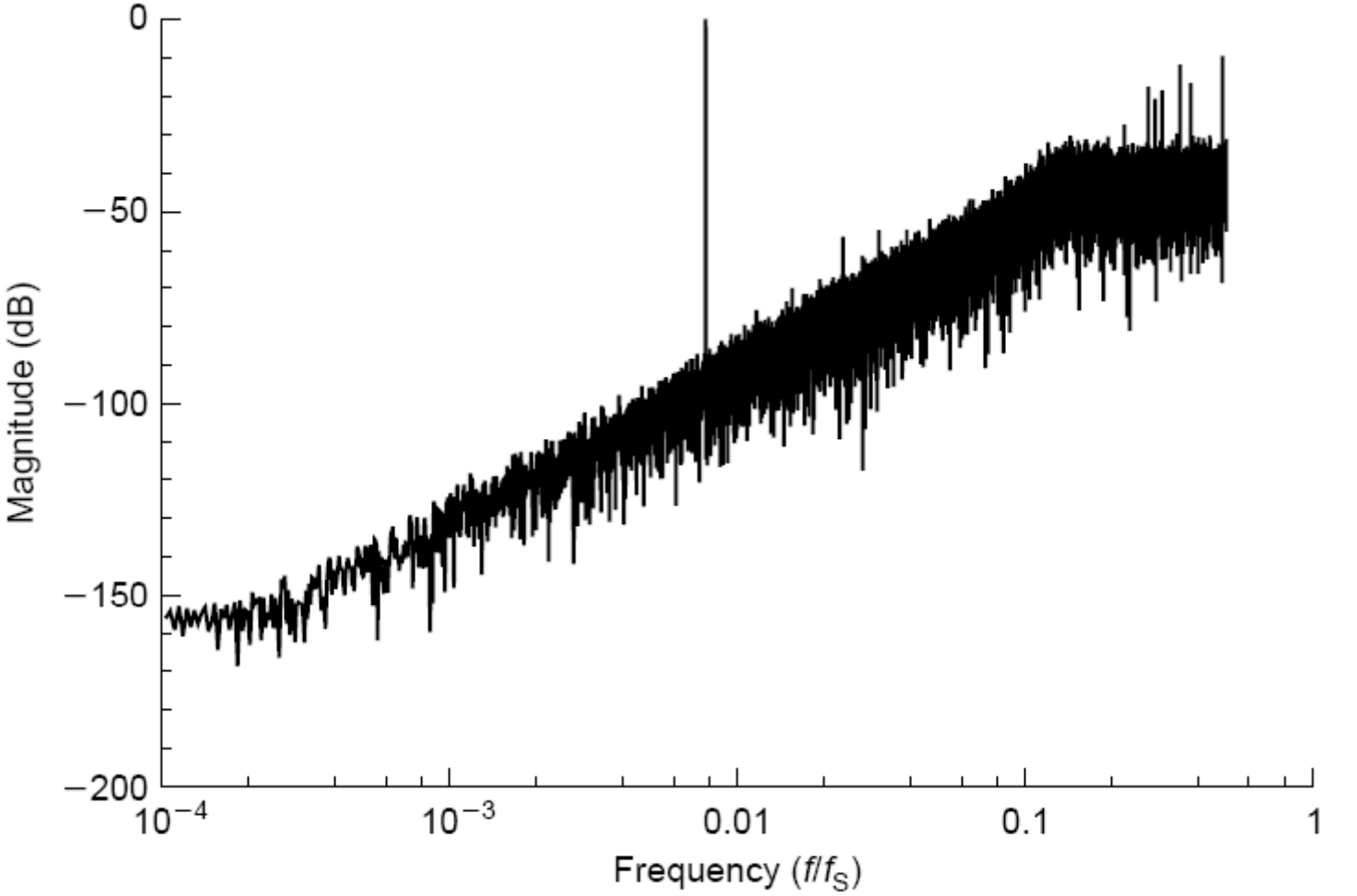
SIGNAL TERM NOISE TERM

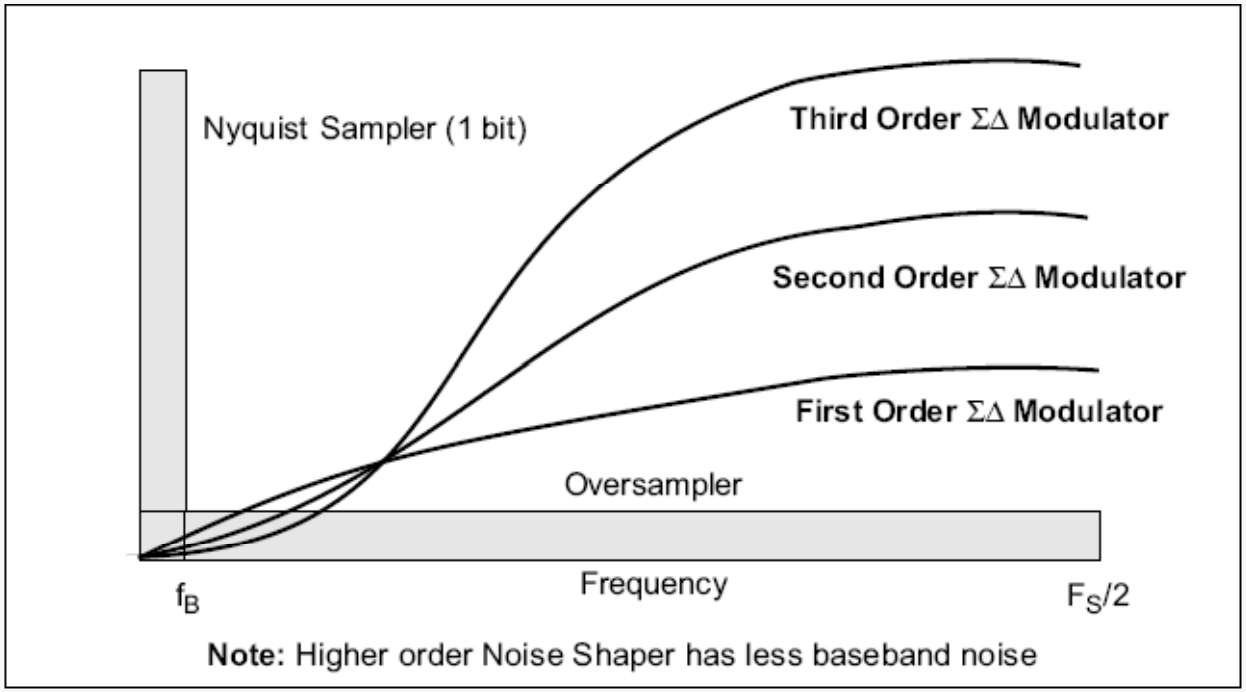



*Block Diagram of First-Order Sigma-Delta A/D converter*




*Digital Decimation Process*

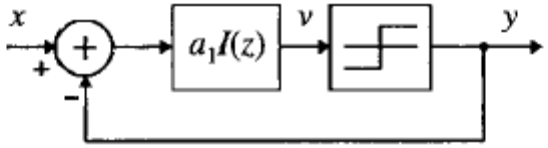

0
−50
−100
−150
−200
Magnitude (dB)
10−4
10−3
0.01
0.1
1
Frequency (f/fS)



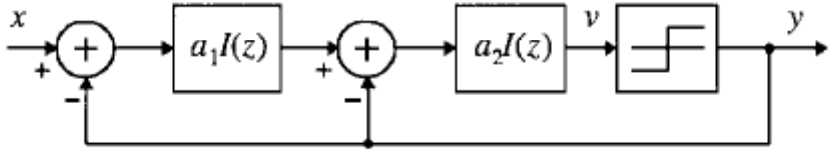

*Multi-Order Sigma-Delta Noise Shapers*

$$SNR = \frac{3\pi}{2} A_x^2 (2L+1)(2B-1)^2 \left(\frac{OVR}{\pi}\right)^{2L+1}$$

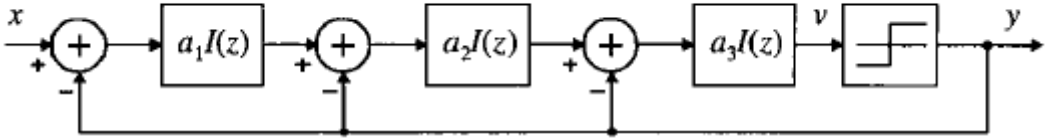
- L : loop order
- B: no. of bits in discriminator
- $A_x = 2^B - 1$



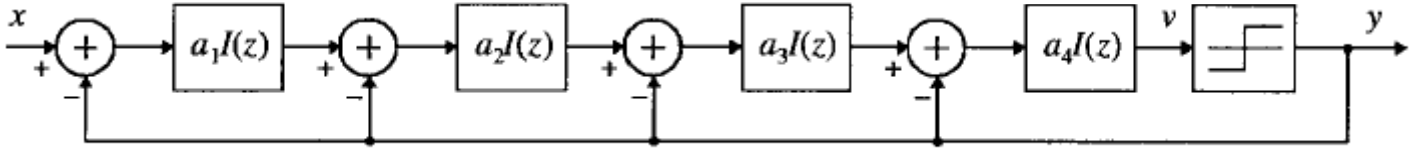

First-order $\Delta\Sigma$ modulator block diagram.



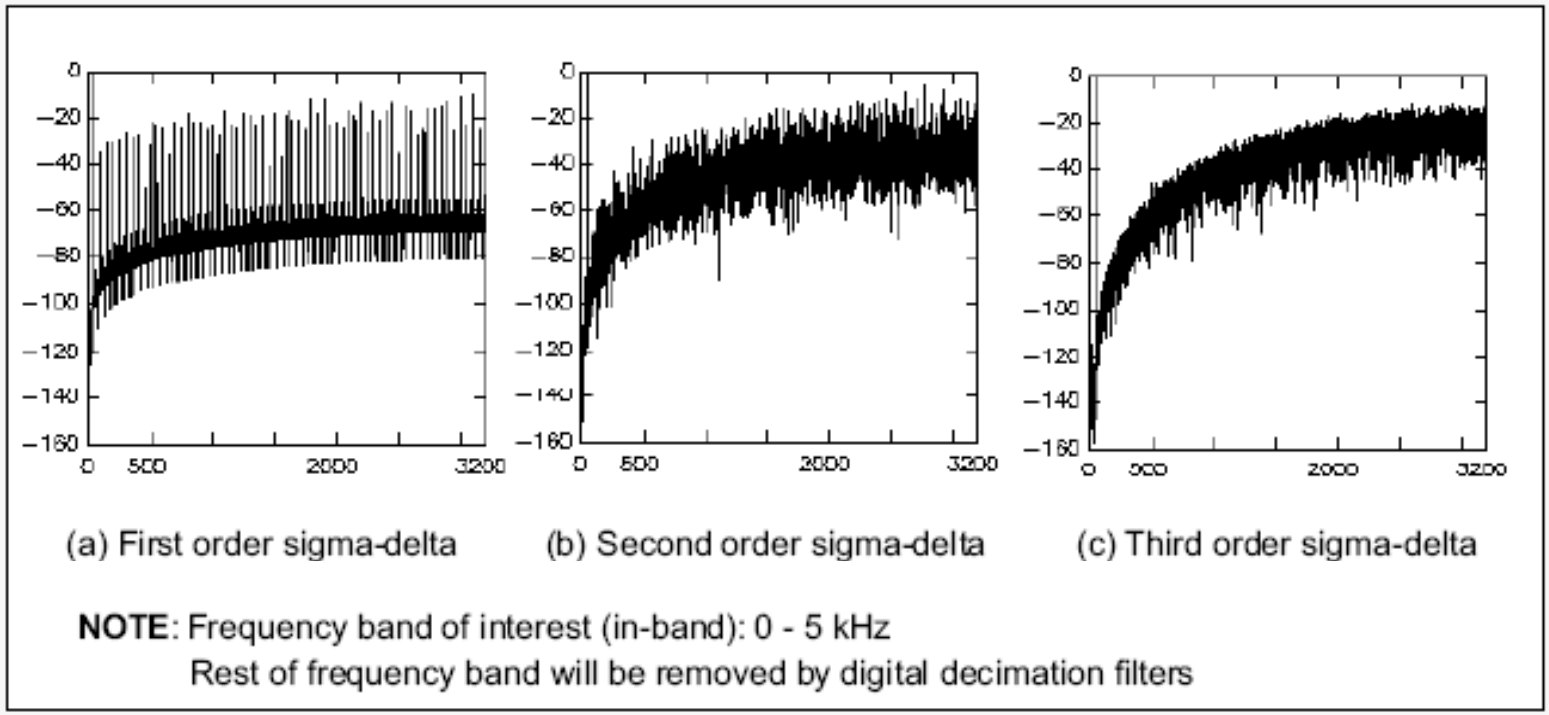

Second-order $\Delta\Sigma$ modulator block diagram.



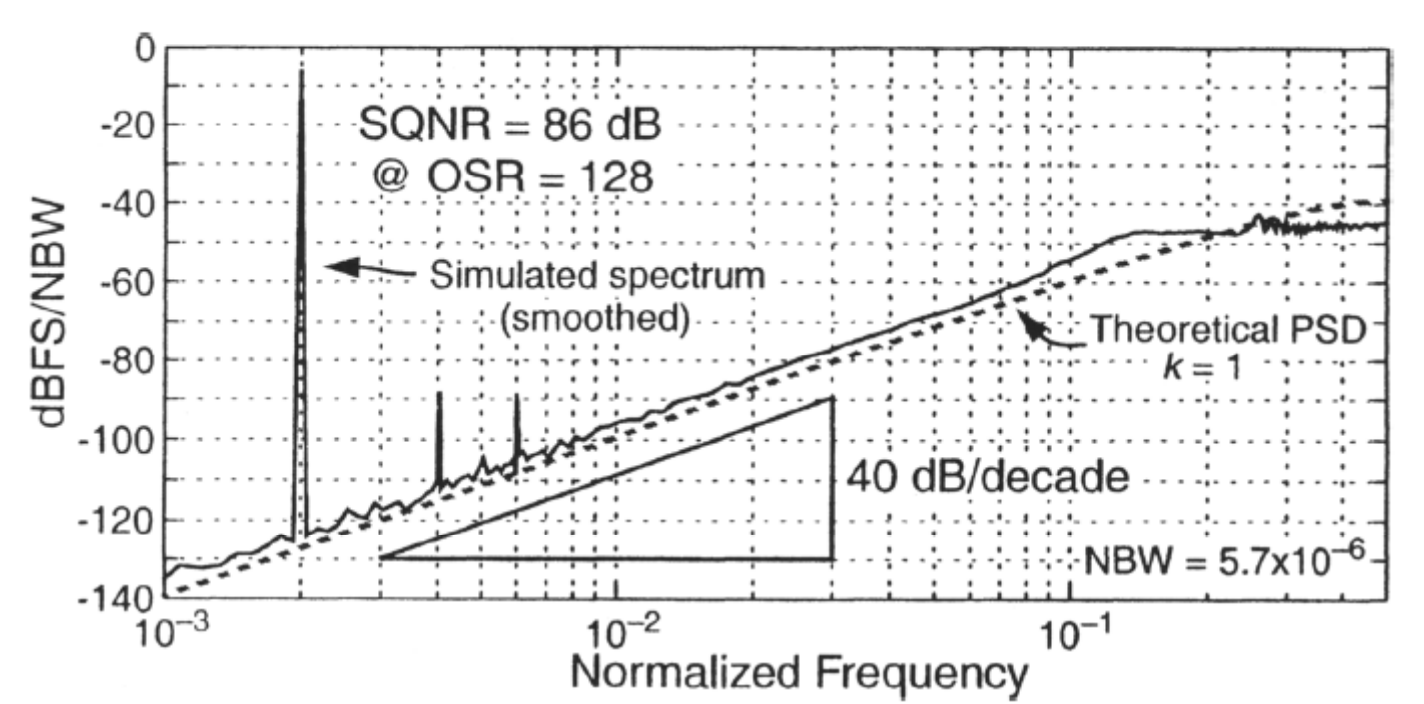

Third-order $\Delta\Sigma$ modulator block diagram.




Fourth-order $\Delta\Sigma$ modulator block diagram.

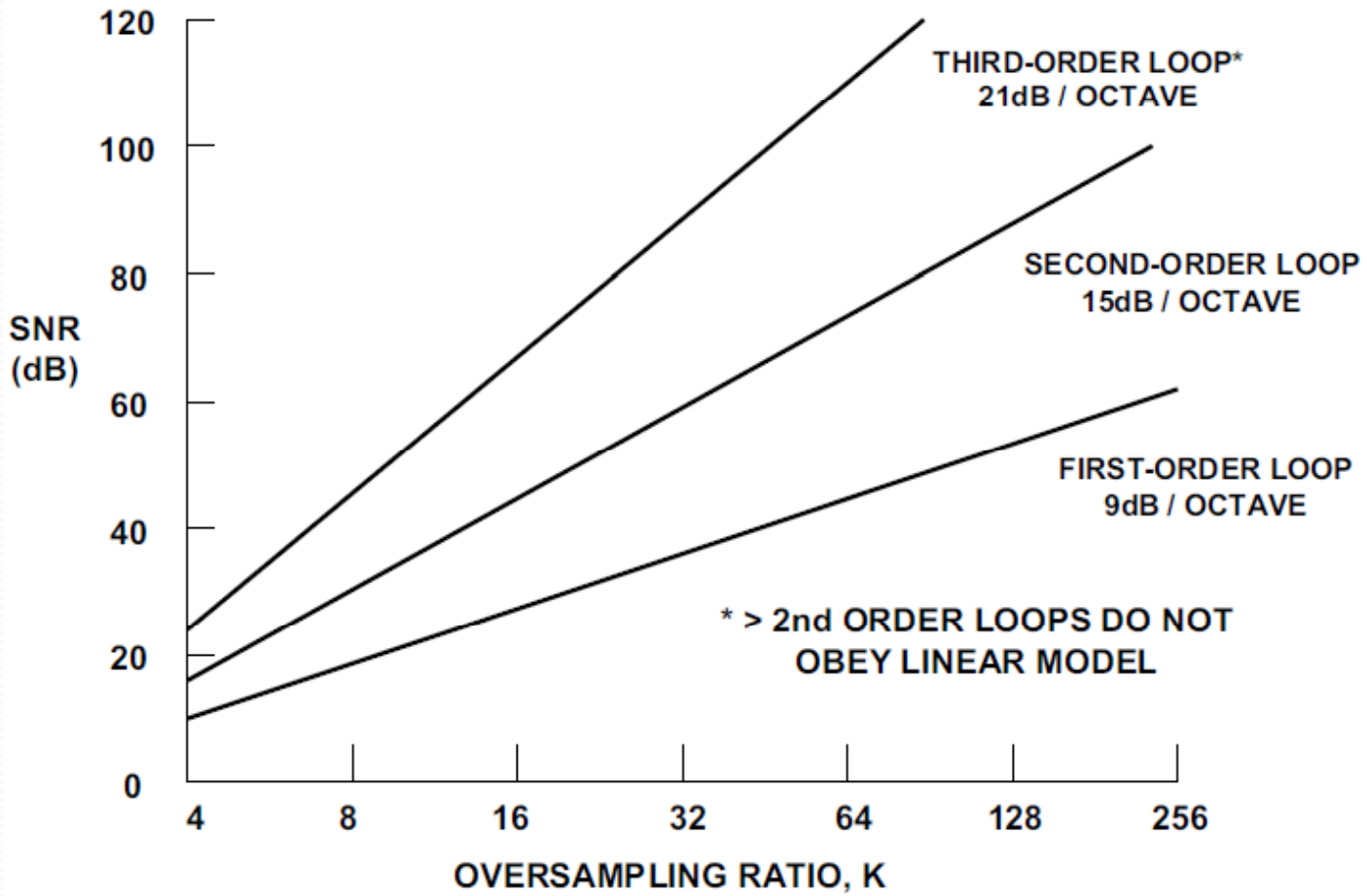



*Spectra of Three Sigma-Delta Noise Shapers*




Spectrum of 2nd-order modulator

Ref: Schreirer and Temes

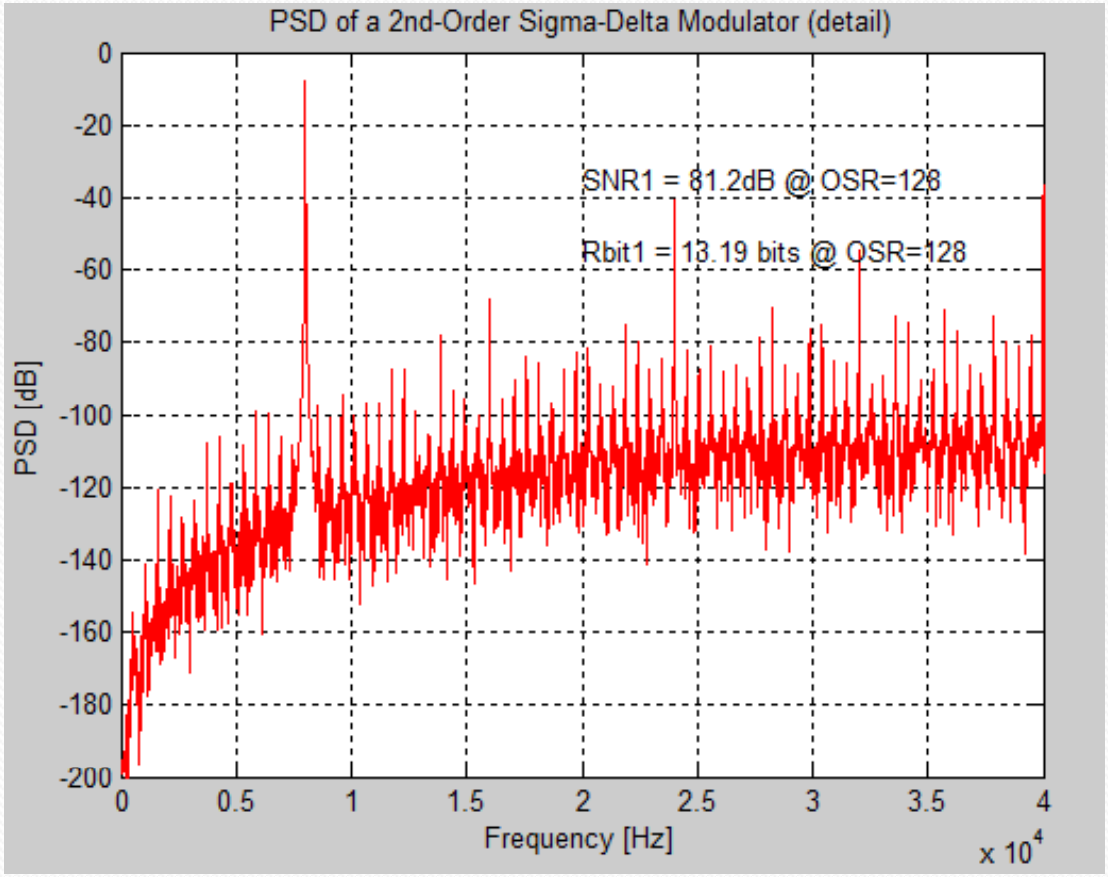


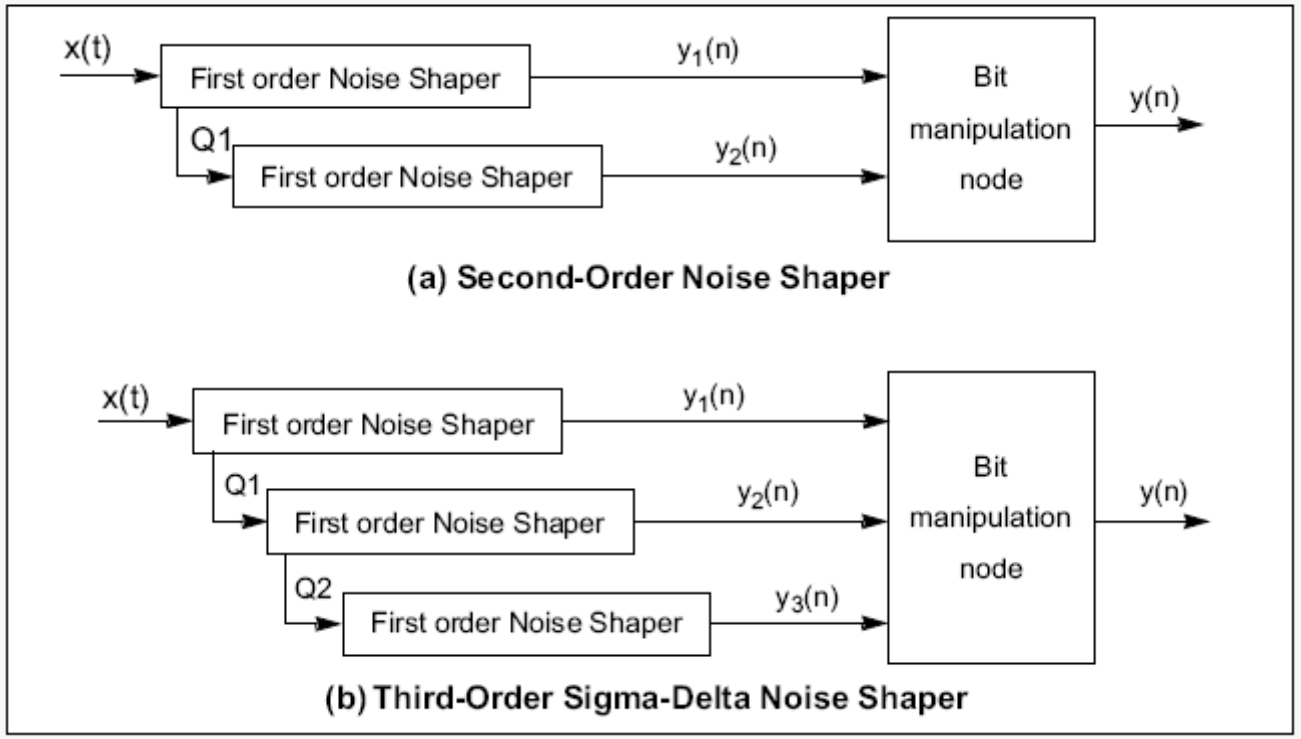

Ref: Kester, M. - Analog Devices, MT - 022 Tutorial

CALCULATED PERFORMANCE OF THE IDEAL $\Delta\Sigma$ MODULATOR (1-BIT/4-BIT QUANTIZER)

| *n* | 2 | | | 3 | | | 4 | | |
|---|---|---|---|---|---|---|---|---|---|
| OVR | *SNR* | *DR* | *OL* | *SNR* | *DR* | *OL* | *SNR* | *DR* | *OL* |
| *16* | *37 / 61* | *49 / 73* | *1* | *53 / 76* | *65 / 88* | *1* | *68 / 91* | *80 / 103* | *1* |
| *32* | *52 / 76* | *64 / 88* | *1* | *74 / 97* | *86 / 109* | *1* | *95 / 118* | *107 / 131* | *1* |
| *64* | *67 / 91* | *79 / 103* | *1* | *95 / 118* | *107 / 130* | *1* | *122 / 146* | *134 / 158* | *1* |
| *128* | *82 / 106* | *94 / 118* | *1* | *116 / 139* | *128 / 151* | *1* | *149 / 173* | *161 / 185* | *1* |

OL (overload level) : defined as the maximum input signal amplitude for which A/D still operates correctly

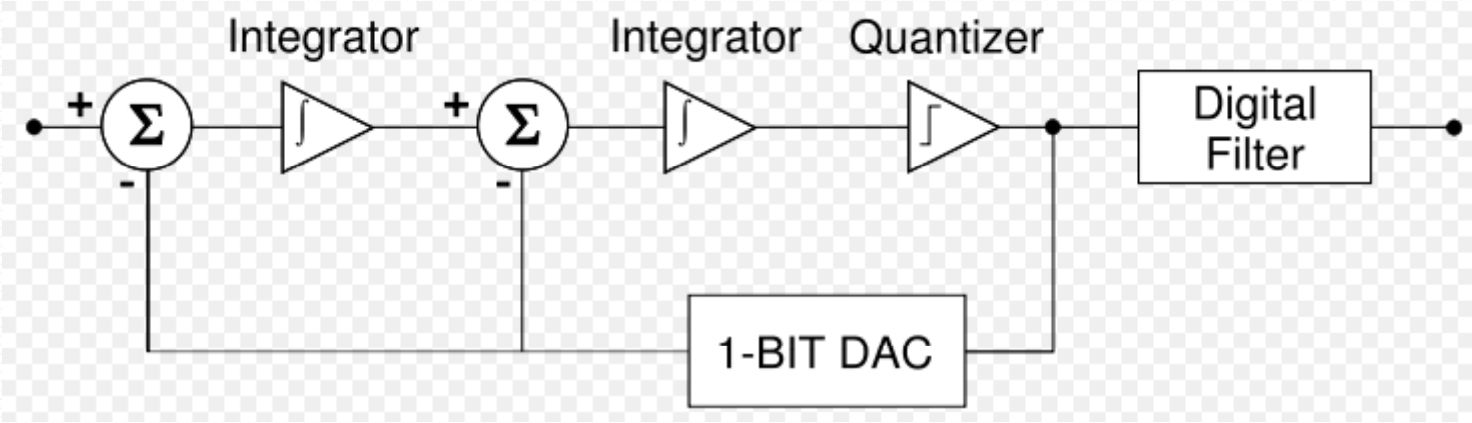

PSD of a 2nd-Order Sigma-Delta Modulator (detail)
SNR1 = 81.2dB @ OSR=128
Rbit1 = 13.19 bits @ OSR=128
PSD [dB]
Frequency [Hz]
x 10^4
0
-20
-40
-60
-80
-100
-120
-140
-160
-180
-200
0
0.5
1
1.5
2
2.5
3
3.5
4




*Second-Order and Third-Order Sigma-Delta Noise Shapers*

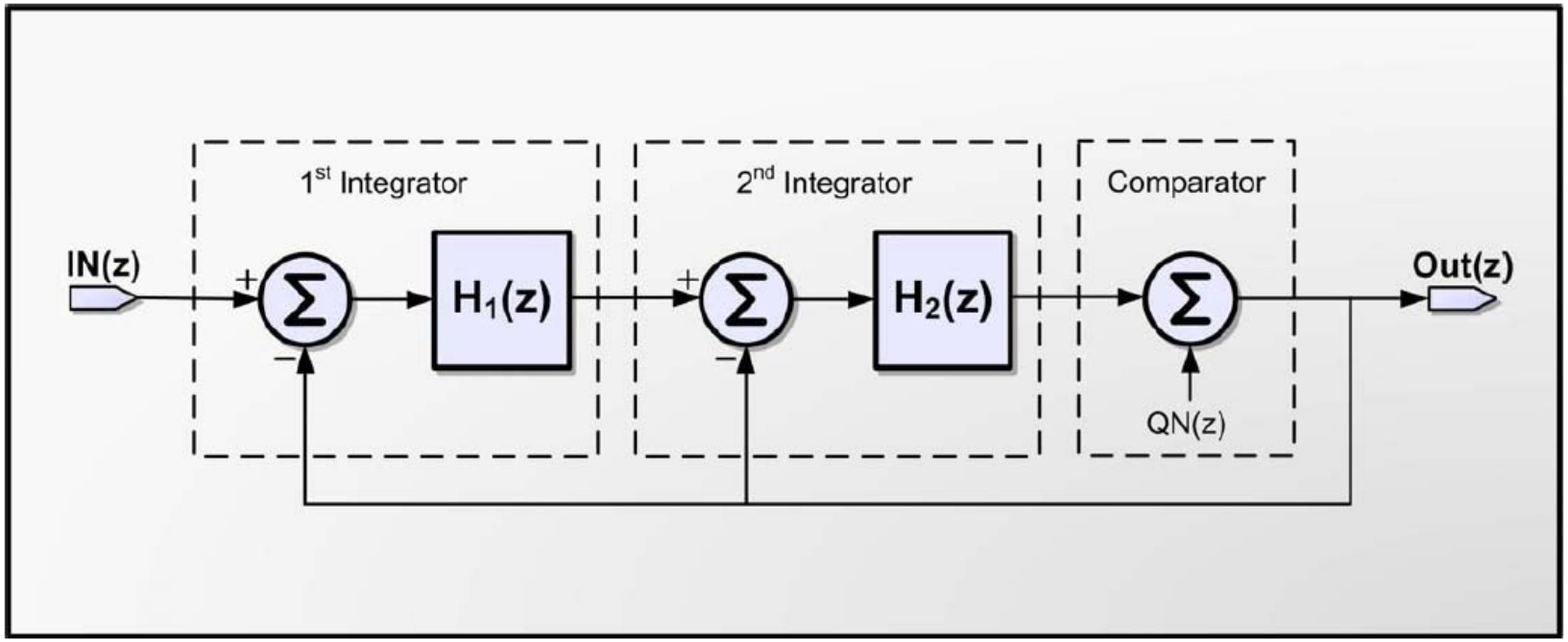
- $2^{nd}$-order Σ–Δ modulator

  - Continuous-Time (CT) Integrators

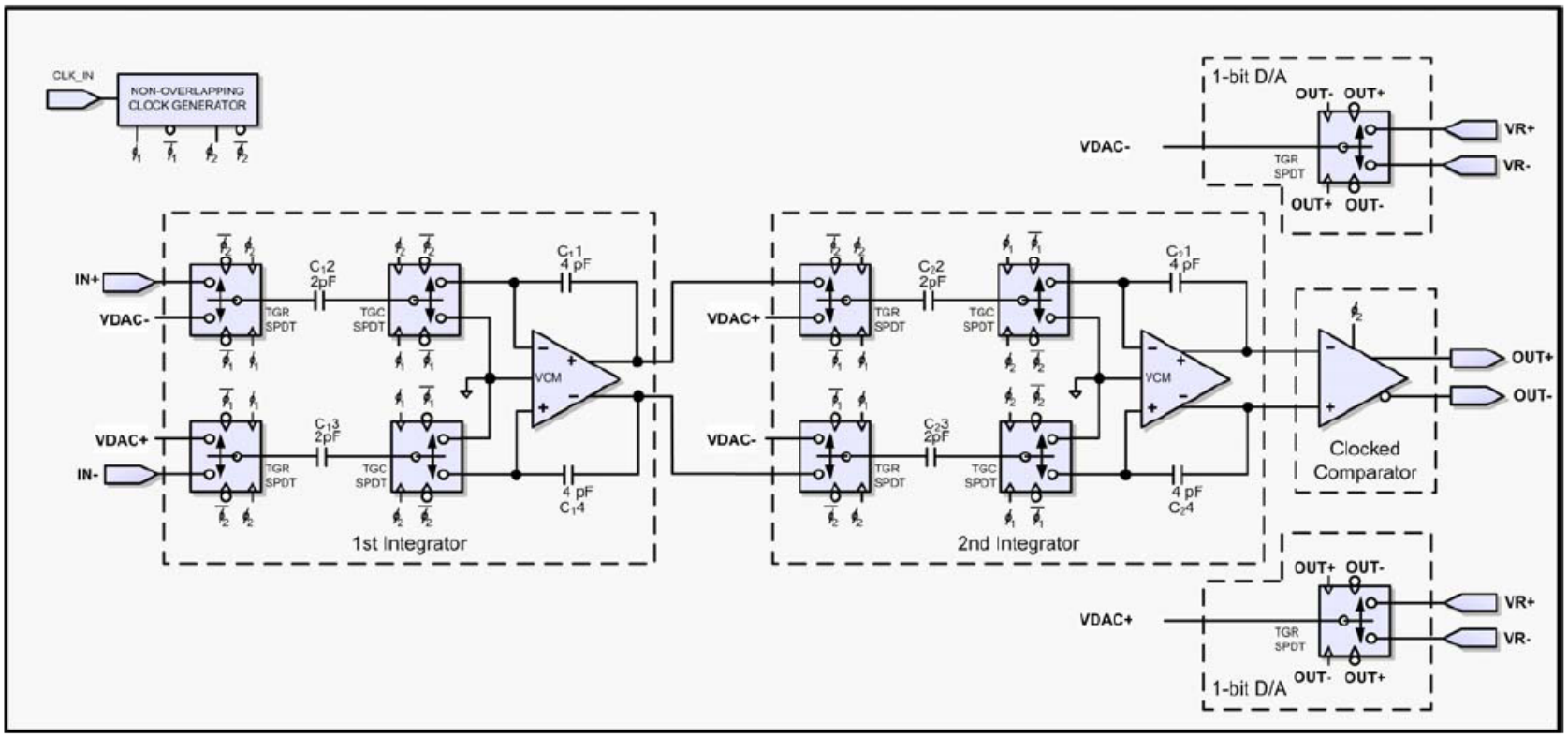


- 2nd-order Σ–Δ modulator

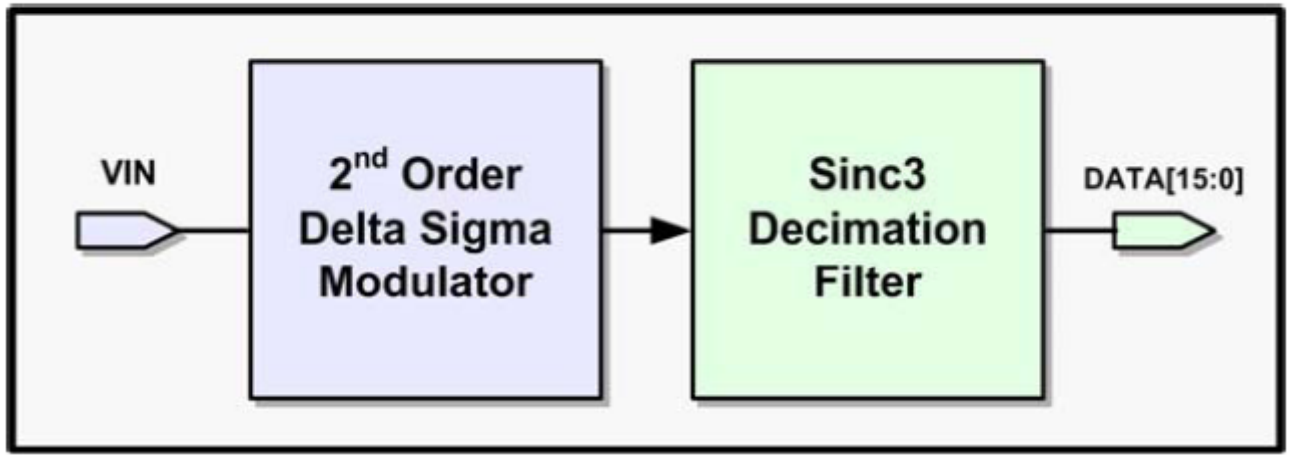
  - Discrete-Time (CT) Integrators




**Block diagram of a second order ΣΔ modulator**




**Differential 2nd order ΣΔ modulator**




High-level view of delta-sigma modulator

# Decimation

• decimation is used in a Σ−Δ ADC to

i) downsampling (eliminate redundant data at the output); re-sample the signal at Nyquist rate.
ii) attenuate noise above baseband

• Nyquist theorem: sample rate only needs to be 2x the input-signal bandwidth to reliably reconstruct the original signal

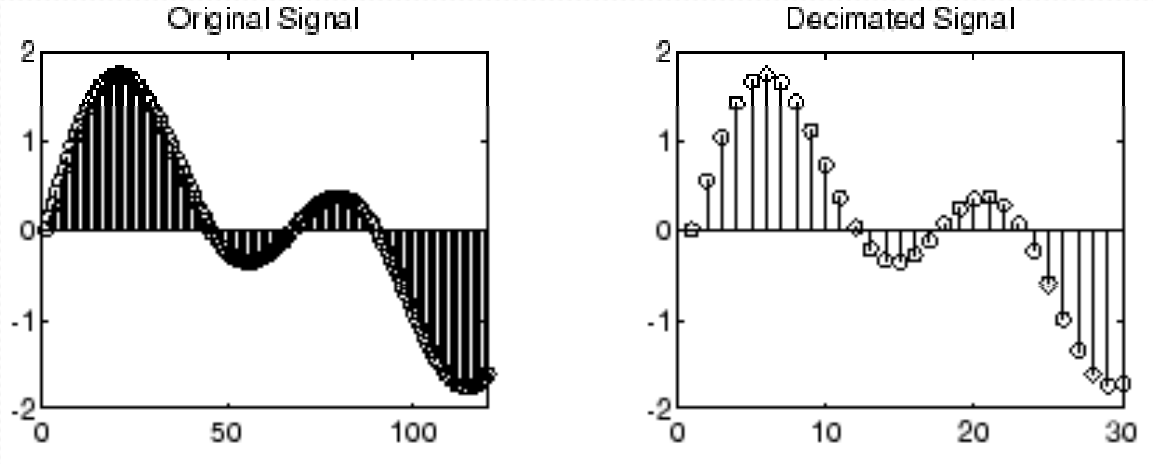
• However, the input signal was grossly oversampled to reduce the quantization noise

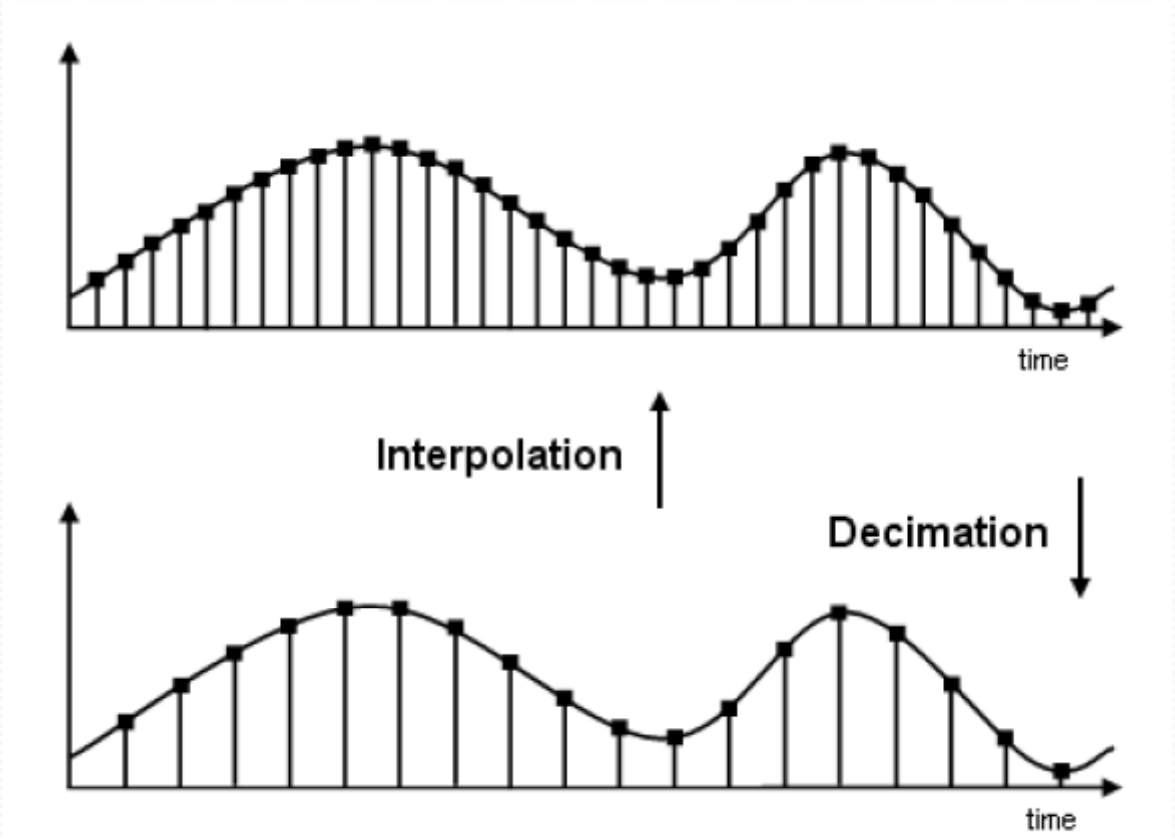
⇒ there is redundant data that can be eliminated without introducing distortion to the conversion result.

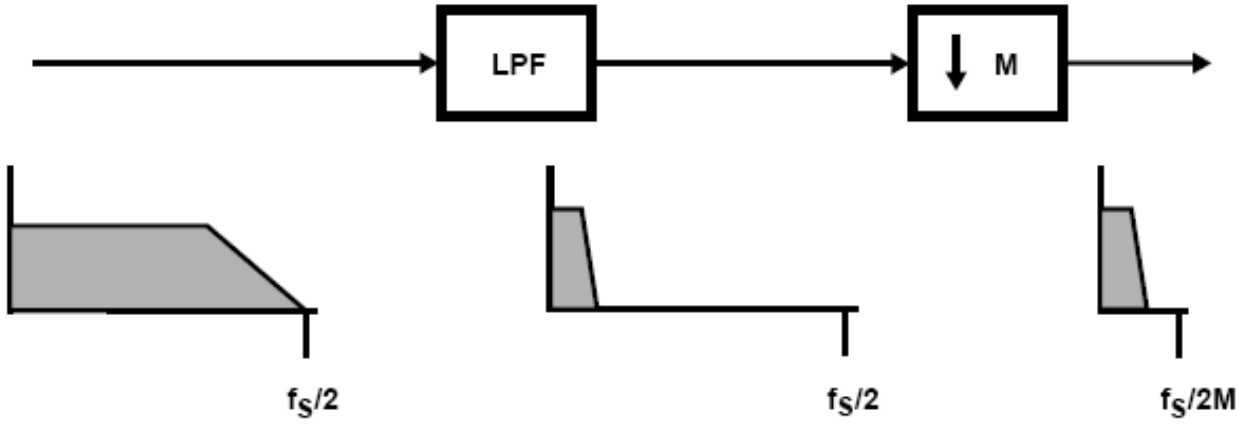
Essentially, decimation is both an averaging filter function and a rate reduction function performed simultaneously.

Decimation reduces the original sampling rate to a lower rate, the opposite of interpolation.

$\Rightarrow$ output sampling rate by ignoring all but every Mth sample

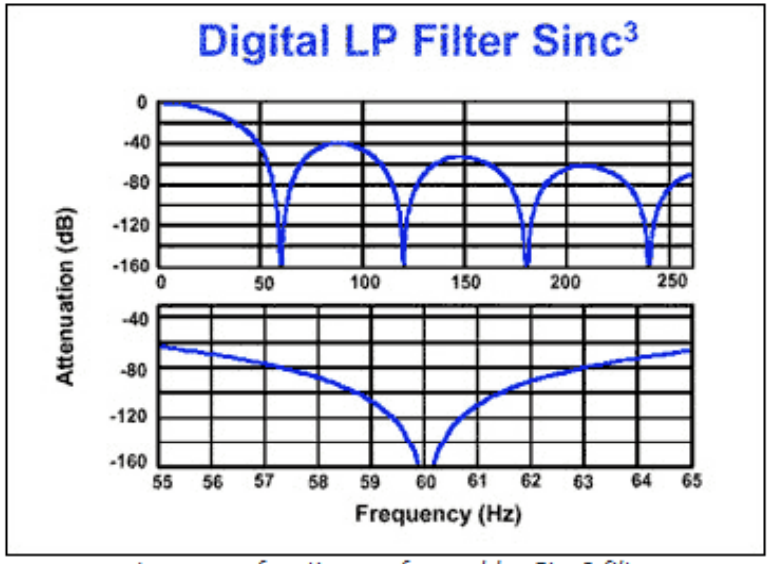



time
Interpolation
Decimation
time

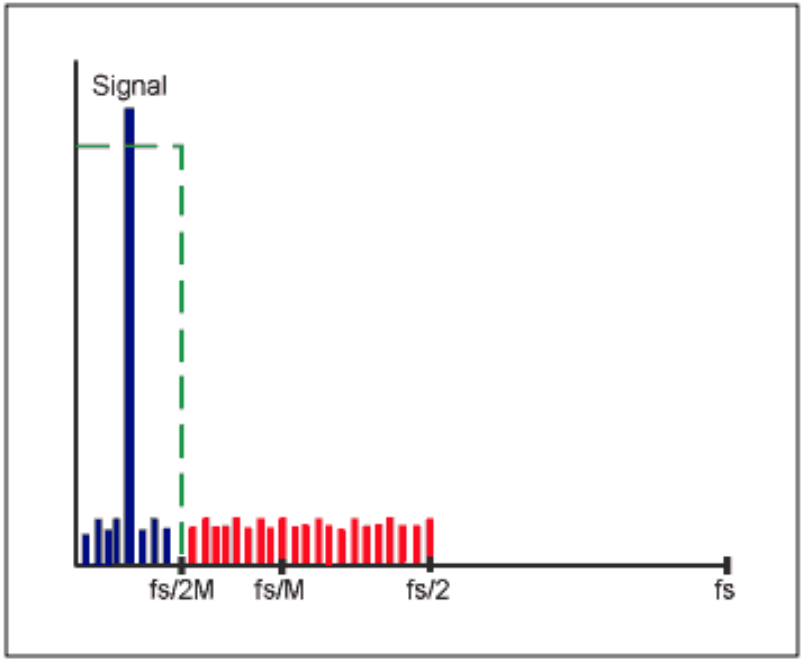



BLOCK DIAGRAM AND SPECTRAL REPRESENTATION OF THE DECIMATION PROCESS

- decimation comb filter

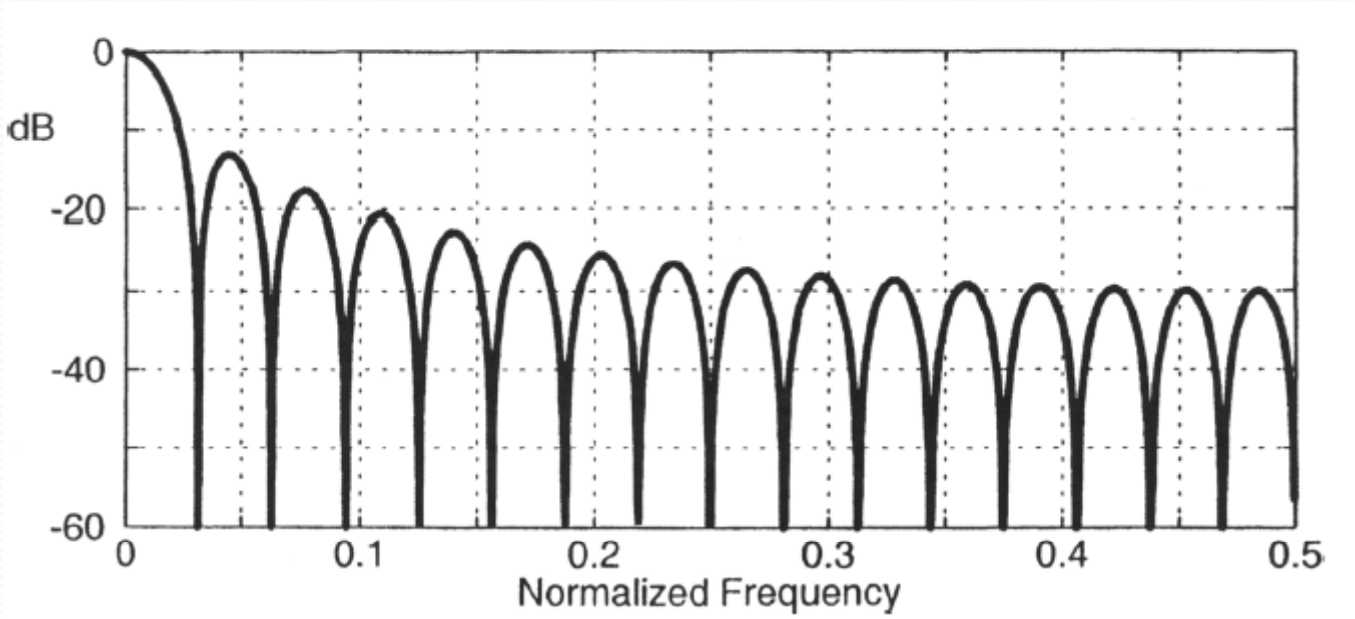



*Low-pass function performed by Sinc³ filter.*




*Decimation does not cause any loss of information.*

- programmable decimation ratio
- the digital filter is generally designed so that zeros occur at the mains frequencies of 50 Hz/60 Hz.

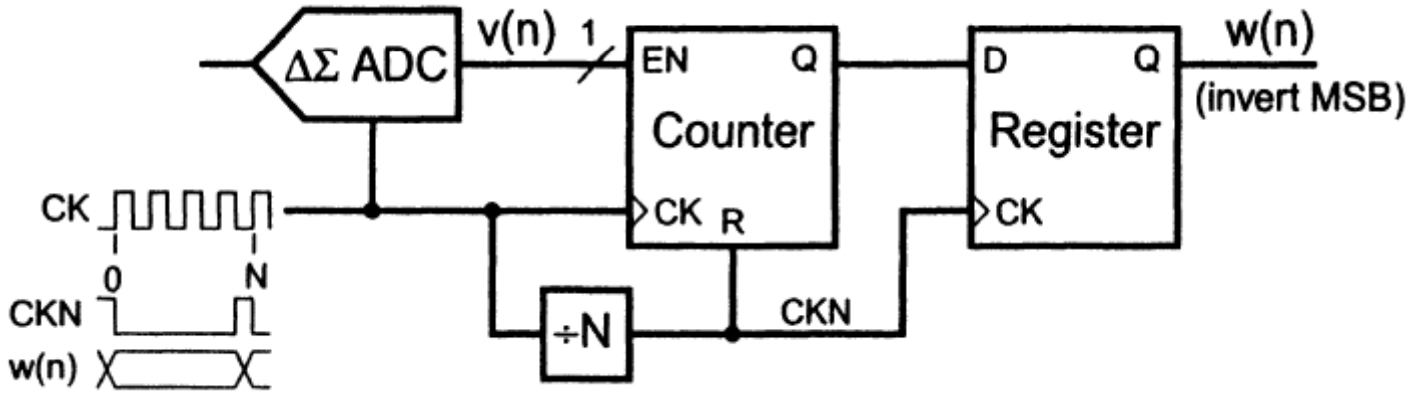



**Frequency response of a sinc filter with $N = 32$.**

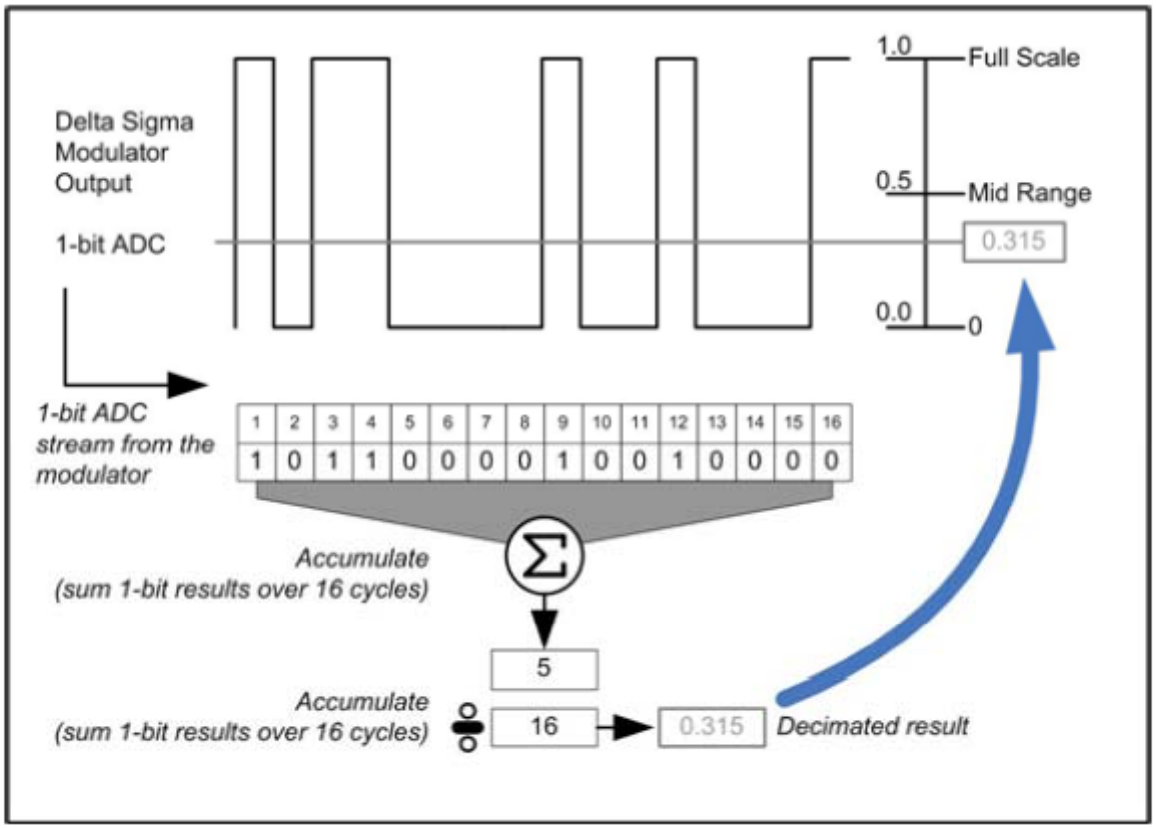
• ΣΔ - ADC can also be viewed as a synchronous voltage-to-frequency converter followed by a counter.

"If the number of "1"s in the output data stream is counted over a sufficient number of samples, the counter output will represent the digital value of the input"

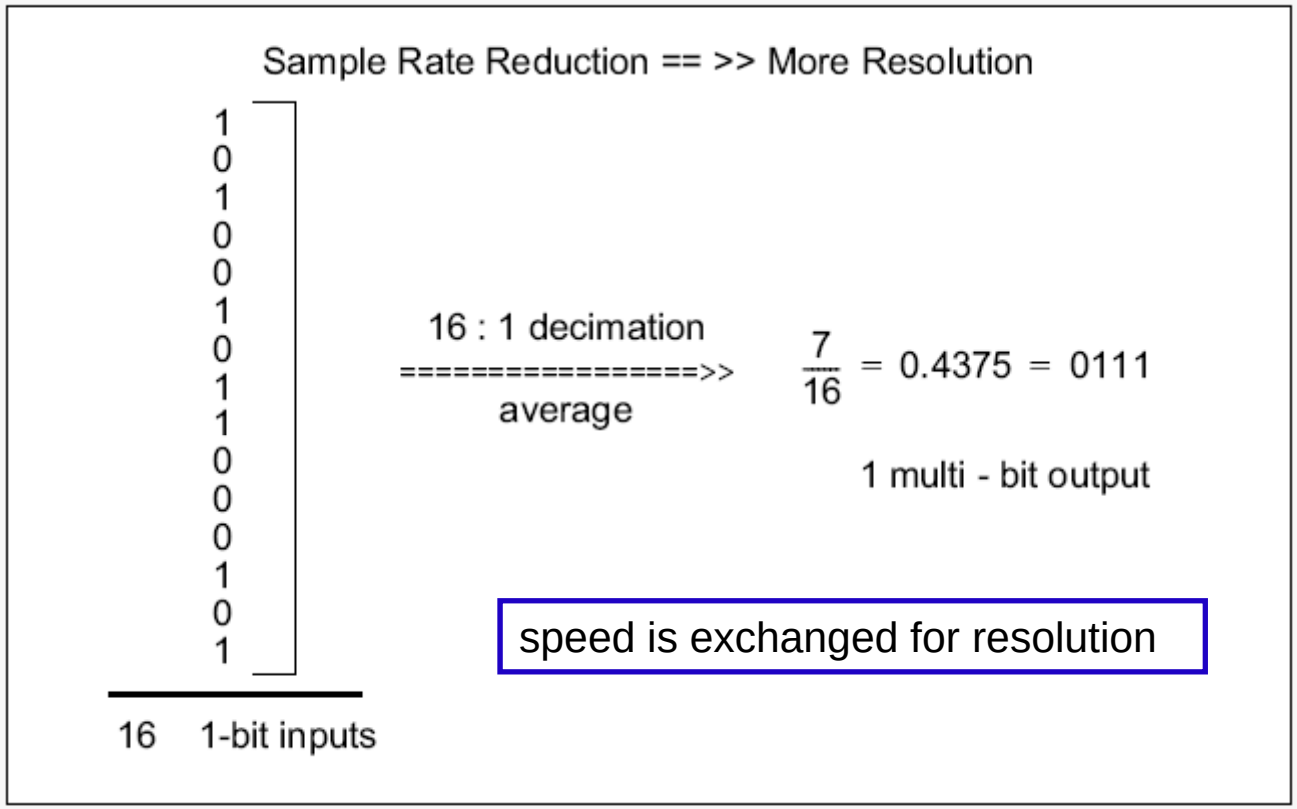
- This method of averaging only works for dc or very slowly changing input signals.
- $2^N$ clock cycles must be counted in order to achieve N-bit effective resolution, thereby severely limiting the effective sampling rate.



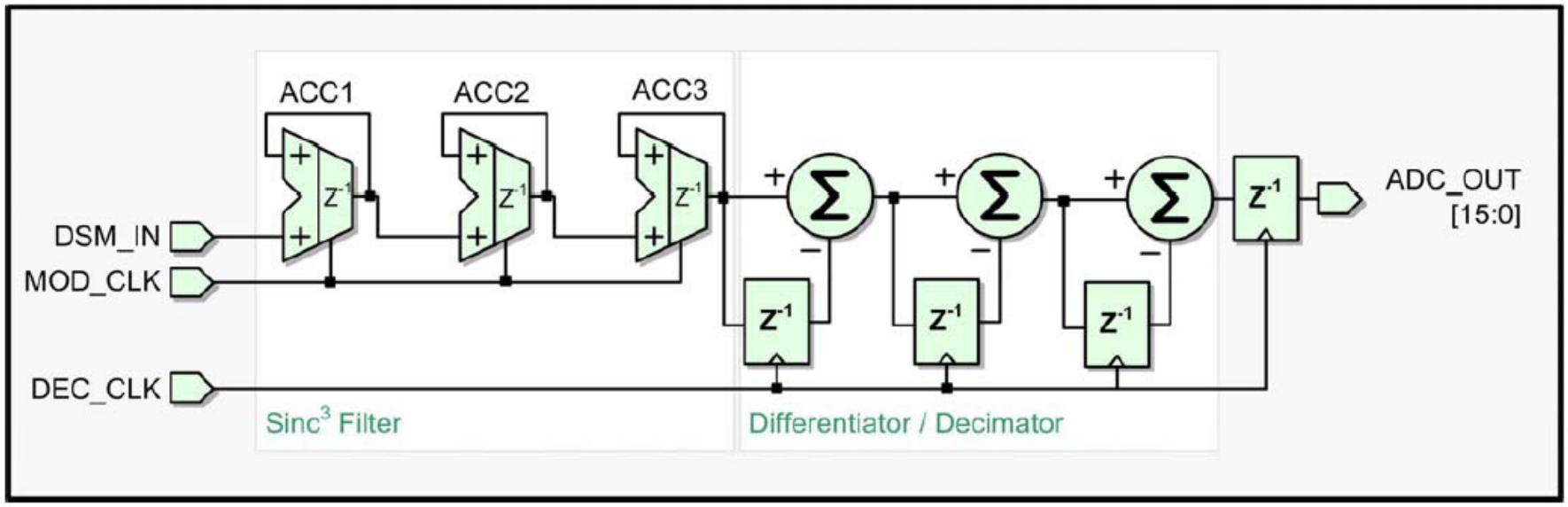

**Sinc filter realized with a counter.**




**ΣΔ A/D stream accumulated and decimated to represent an n-bit value of the input**



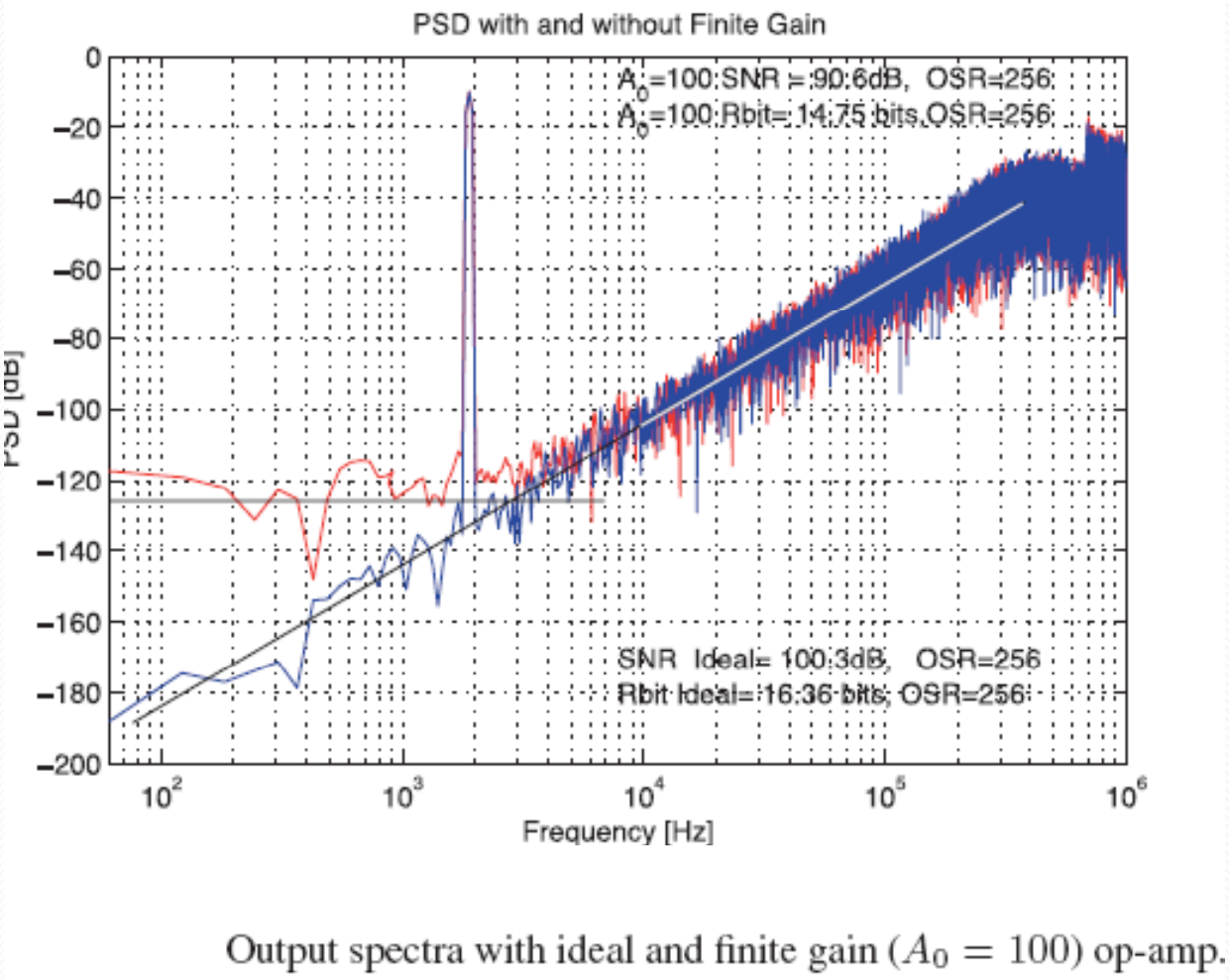

*Simple Example of Decimation Process*



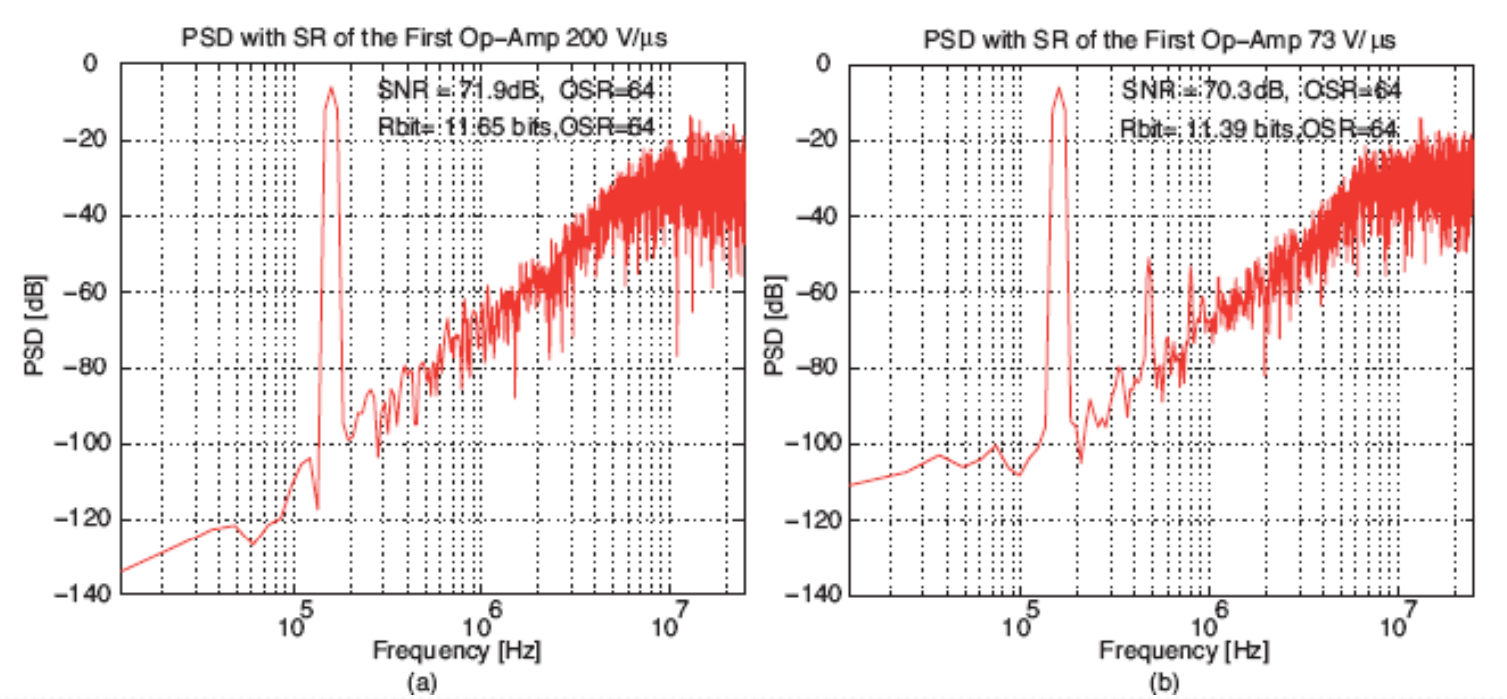

**Sinc3 digital decimation filter**

```
// decimation.v
// Decimation filter with Sinc3 filter followed by
// Differentiator and Decimation
//
module decimation(
        DSM_i,
        DSM_clk_i,
        WordClk_i;
        Reset_i,
        DWord_ro);

input           DSM_clk_i;  // DSM-rate clock (bit
clock)
input           WordClk_i;  // Output Word-rate clock
input           Reset_i;    // Active-hi reset
input           DSM_i;      // Input from Modulator
output  [15:0]  DWord_ro;   // 16-bit Output Word

reg     [23:0]  Acc1_r;
reg     [23:0]  Acc2_r;
reg     [23:0]  Acc3_r;
reg     [23:0]  Acc3_q1_r;
reg     [23:0]  Acc3_q2_r;
reg     [23:0]  Diff1_r;
reg     [23:0]  Diff2_r;
reg     [23:0]  Diff3_r;
reg     [23:0]  Diff1_q1_r;
reg     [23:0]  Diff2_q2_r;
reg     [15:0]  DWord_ro;
//
// Internal Wires
//
// 2's-comp version of DWord
wire    [23:0]  DWord_2comp_w;
// Sinc Filter
assign DWord_2comp_w = (DSM_i==1'b0) ? 24'd0 : 24'd1;

// Accumulator (Integrator)

always @(negedge DSM_clk_i or posedge Reset_i)
if (Reset_i) begin
    /* initialize acc registers on Reset_i */
    Acc1_r <= 24'd0;
    Acc2_r <= 24'd0;
    Acc3_r <= 24'd0;
end
else begin
    /* perform accumulation process */
    Acc1_r <= Acc1_r + DWord_2comp_w;
    Acc2_r <= Acc2_r + Acc1_r;
    Acc3_r <= Acc3_r + Acc2_r;
end
```

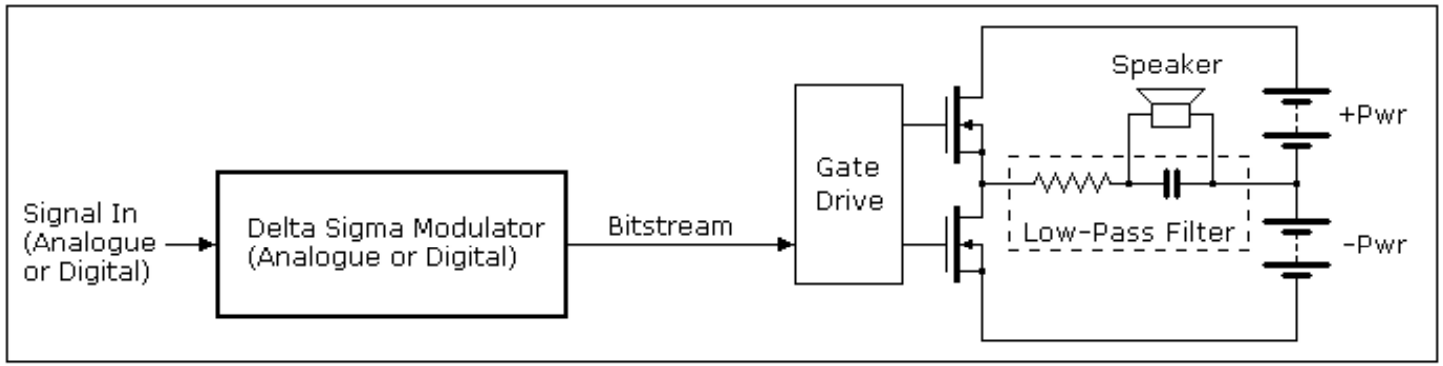
```
//
// Decimation Filter
//
/* Decimation stage (MClkOut/ WordClk) */
always a(negedge DSM_clk_i or posedge Reset_i)
if (Reset_i)
    word_count <= 8'd0;
else
    word_count <= word_count + 8'd1;

//
// Differentiator and Decimation
//
always @ (posedge WordClk_i or posedge Reset_i)
if(Reset_i) begin
    Acc3_r_d2   <= 24'd0;
    Diff1_q1_r <= 24'd0;
    Diff2_q1_r <= 24'd0;
    Diff1_r     <= 24'd0;
    Diff2_r     <= 24'd0;
    Diff3_r     <= 24'd0;
end
else begin
    Diff1_r     <= Acc3_r - Acc3_q2_r;
    Diff2_r     <= Diff1_r - Diff1_q1_r;
    Diff3_r     <= Diff2_r - Diff2_q1_r;
    Acc3_q2_r   <= Acc3_r;
    Diff1_q1_r <= Diff1_r;
    Diff2_q1_r <= Diff2_r;

    DWord_ro <= Diff3_r[23:8];
end
endmodule
```




Output spectra with ideal and finite gain ($A_0 = 100$) op-amp.


PSD with SR of the First Op–Amp 200 V/µs
SNR = 71.9dB, OSR=64
Rbit= 11.65 bits,OSR=64
PSD [dB]
0
−20
−40
−60
−80
−100
−120
−140
10^5
10^6
10^7
Frequency [Hz]
(a)
PSD with SR of the First Op–Amp 73 V/ µs
SNR = 70.3dB, OSR=64
Rbit= 11.39 bits,OSR=64
PSD [dB]
0
−20
−40
−60
−80
−100
−120
−140
10^5
10^6
10^7
Frequency [Hz]
(b)

**Class-D Power Amplifiers**:




**Power Amplifiers using a Digital Output Stage**

- Both output transistors operate in push-pull mode

- bitstream is amplified to the high level of the power supply and is available at the output with low impedance