LECTURE 37 – TESTING OF ADCS AND MODERATE SPEED NYQUIST ADCS

LECTURE ORGANIZATION

Outline

- Introduction
- Testing of ADCs
- Serial ADCs
- Successive approximation ADCs
- Single-bit/stage pipeline ADCs
- Iterative ADCs
- Self calibration techniques
- Summary

***CMOS Analog Circuit Design*, 3rd Edition Reference**

Pages 557-572

TESTING OF ADCs

Input-Output Test for an ADC

Test Setup:

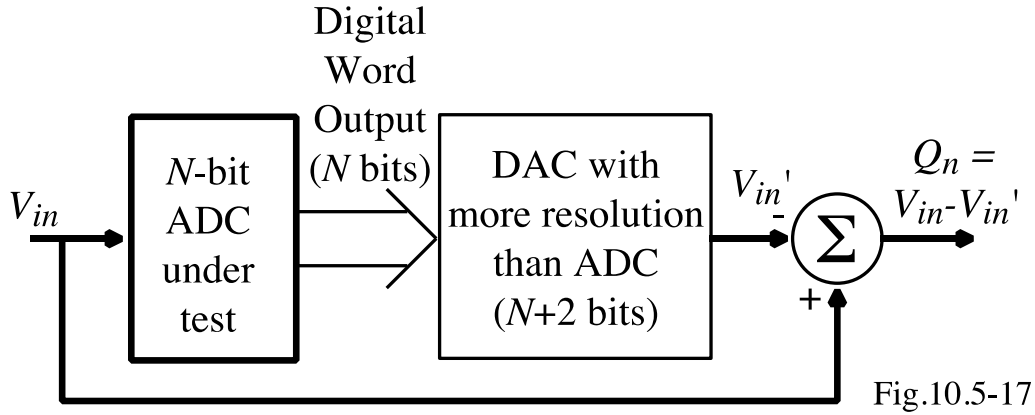


Fig.10.5-17

The ideal value of Q_n should be within $\pm 0.5LSB$

Can measure:

- Offset error = constant shift above or below the 0 *LSB* line
- Gain error = contant increase or decrease of the sawtooth plot as V_{in} is increased
- *INL* and *DNL* (see following page)

Illustration of the Input-Output Test for a 4-Bit ADC

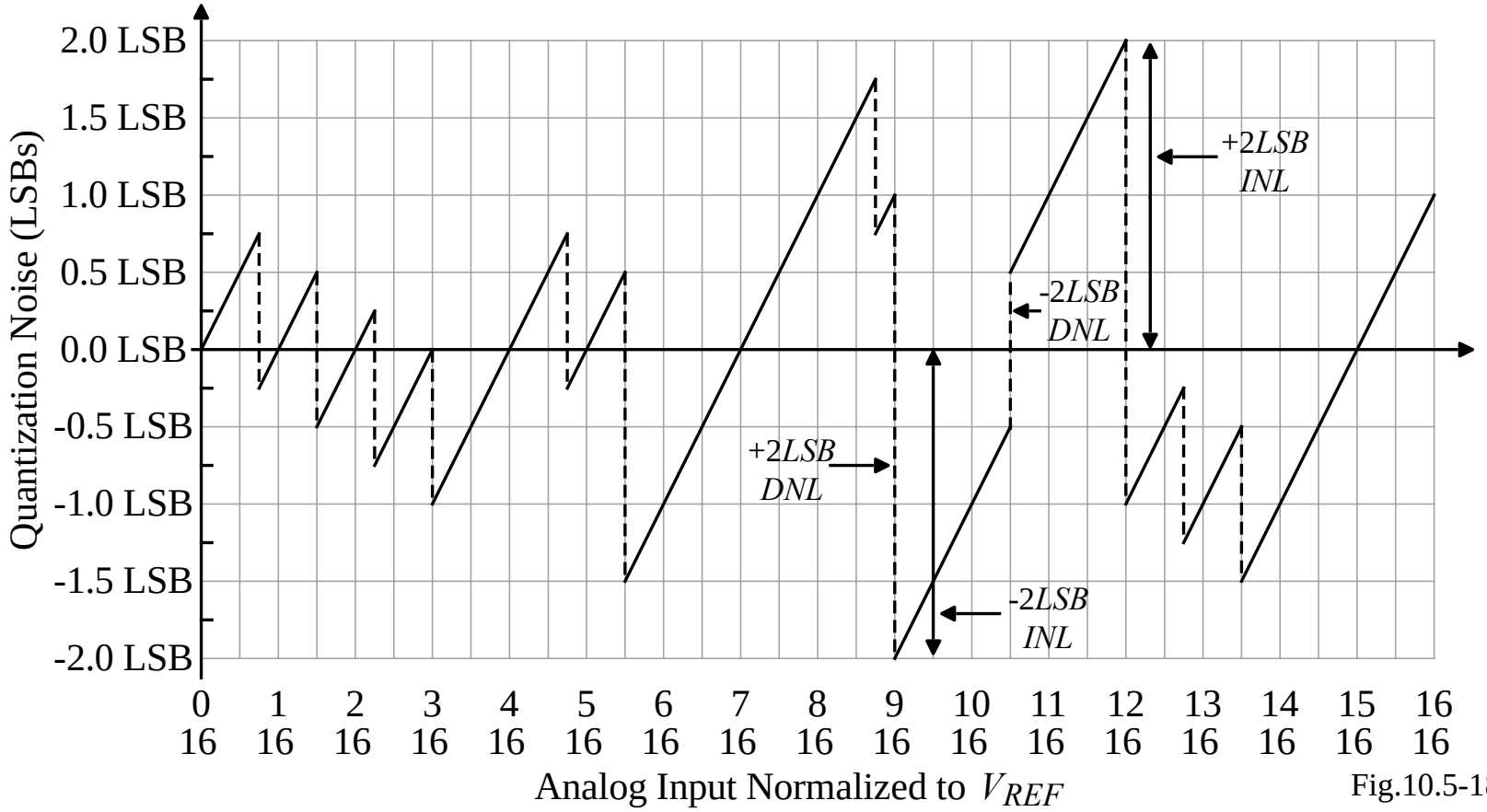


Fig.10.5-18

Measurement of Nonlinearity Using a Pure Sinusoid

This test applies a pure sinusoid to the input of the ADC. Any nonlinearity will appear as harmonics of the sinusoid. Nonlinear errors will occur when the dynamic range (*DR*) is less than 6*N* dB where *N* = number of bits.

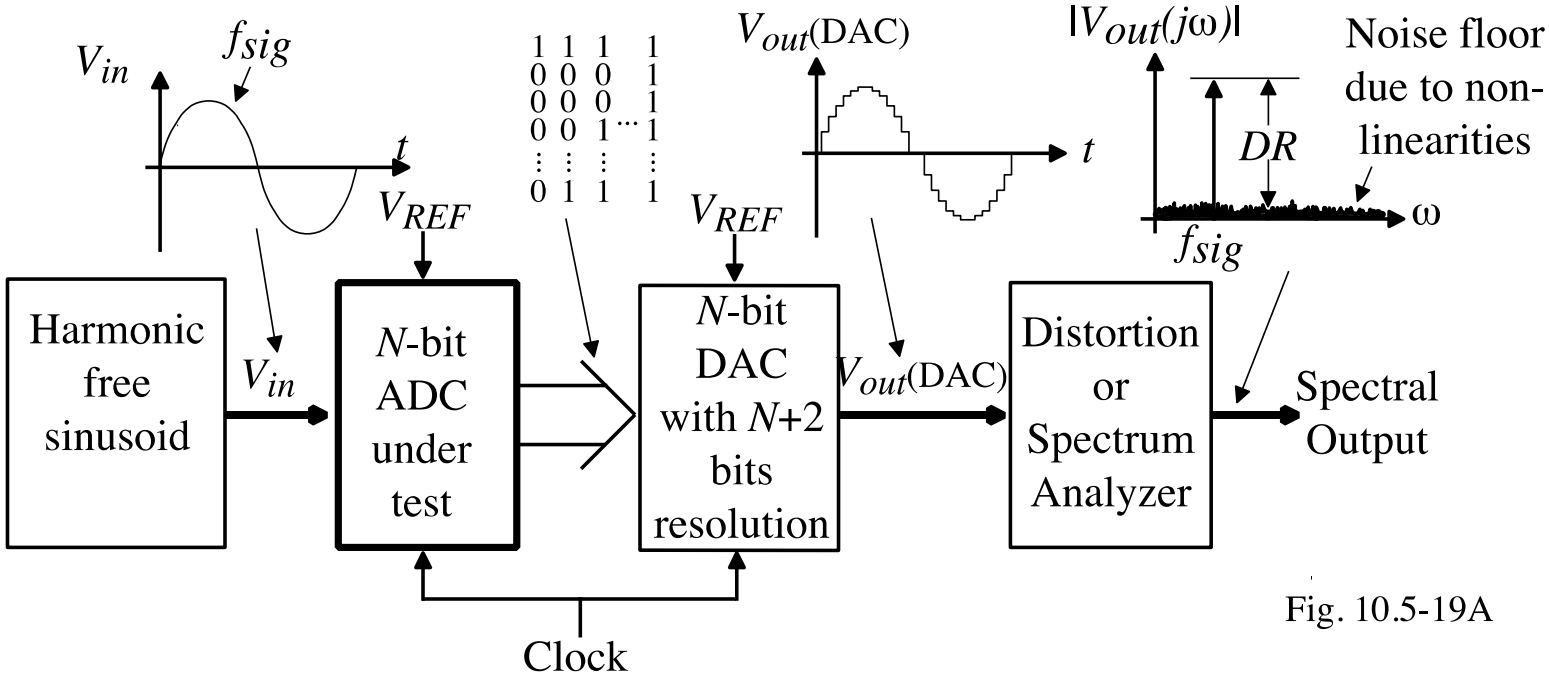


Fig. 10.5-19A

Comments:

- Input sinusoid must have less distortion that the required dynamic range
- DAC must have more accuracy than the ADC

FFT Test for an ADC

Test setup:

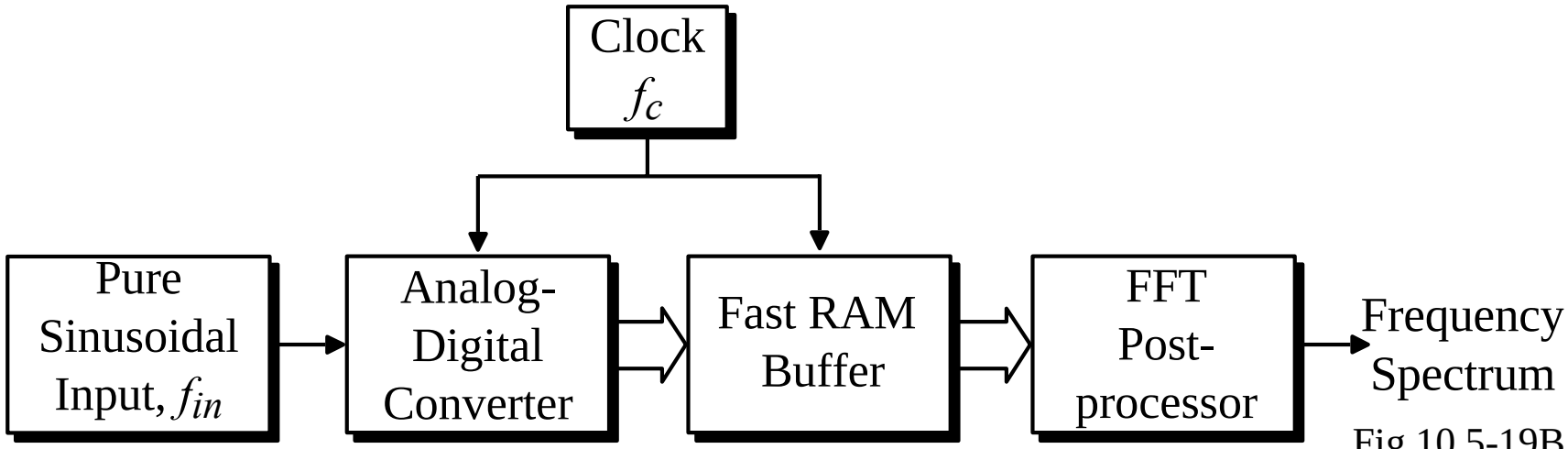


Fig.10.5-19B

Comments:

- Stores the digital output codes of the ADC in a RAM buffer
- After the measurement, a postprocessor uses the FFT to analyze the quantization noise and distortion components
- Need to use a window to eliminate measurement errors (Raised Cosine or 4-term Blackmann-Harris are often used)
- Requires a spectrally pure sinusoid

Histogram Test for an ADC

The number of occurences of each digital output code is plotted as a function of the digital output code.

Illustration:

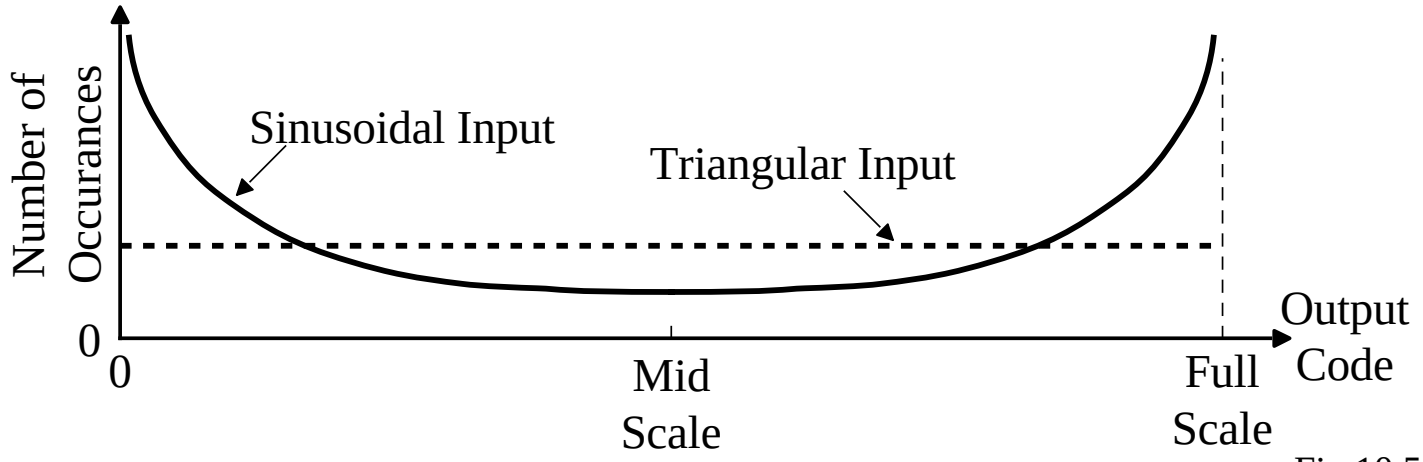


Fig.10.5-20

Comments:

- Emphasizes the time spent at a given level and can show *DNL* and missing codes
- *DNL*

$$DNL(i) = \frac{\text{Width of the bin as a fraction of full scale}}{\text{Ratio of the bin width to the ideal bin width}} - 1 = \frac{H(i)/N_t}{P(i)} - 1$$

where

$H(i)$ = number of counts in the ith bin

N_t = total number of samples

$P(i)$ = ratio of the bin width to the ideal bin width

- *INL* is found from the cumulative bin widths

Comparison of the Tests for Analog-Digital Converters

Other Tests

- Sinewave curve fitting (good for *ENOB*)
- Beat frequency test (good for a qualitative measure of dynamic performance)

Comparison

Test → Error ↓	Histogram or Code Test	FFT Test	Sinewave Curve Fit Test	Beat Frequency Test
DNL	Yes (spikes)	Yes (Elevated noise floor)	Yes	Yes
Missing Codes	Yes (Bin counts with zero counts)	Yes (Elevated noise floor)	Yes	Yes
INL	Yes (Triangle input gives *INL* directly)	Yes (Harmonics in the baseband)	Yes	Yes
Aperature Uncertainty	No	Yes (Elevated noise floor)	Yes	No
Noise	No	Yes (Elevated noise floor)	Yes	No
Bandwidth Errors	No	No	No	Yes (Measures analog bandwidth)
Gain Errors	Yes (Peaks in distribution)	No	No	No
Offset Errors	Yes (Offset of distribution average)	No	No	No

Bibliography on ADC Testing

1.) D. H. Sheingold, *Analog-Digital Conversion Handbook*, Analog Devices, Inc., Norwood, MA 02062, 1972.

2.) S.A. Tretter, *Introduction to Discrete-Time Signal Processing*, John Wiley & Sons, New York, 1976.

3.) J. Doernberg, H.S. Lee, and D.A. Hodges, "Full-Speed Testing of A/D Converters," *IEEE J. of Solid-State Circuits*, Vol. SC-19, No. 6, December 1984, pp. 820-827.

4.) "Dynamic performance testing of A to D converters," *Hewlett Packard Product Note* 5180A-2.

INTRODUCTION TO MODERATE SPEED ADCS

Moderate Speed ADC Topics

- Serial ADCs - require $2^N T$ for conversion where T = period of the clock

  Types:

  - Single-slope
  - Dual-slope

- Successive approximation ADCs – require NT for conversion where T = the clock period
- 1-bit per stage, pipeline ADCs – require T for conversion after a delay of NT
- Iterative ADCs – require NT for conversion
- Self-calibration techniques

SERIAL ANALOG-DIGITAL CONVERTERS

Single-Slope ADC

Block diagram:

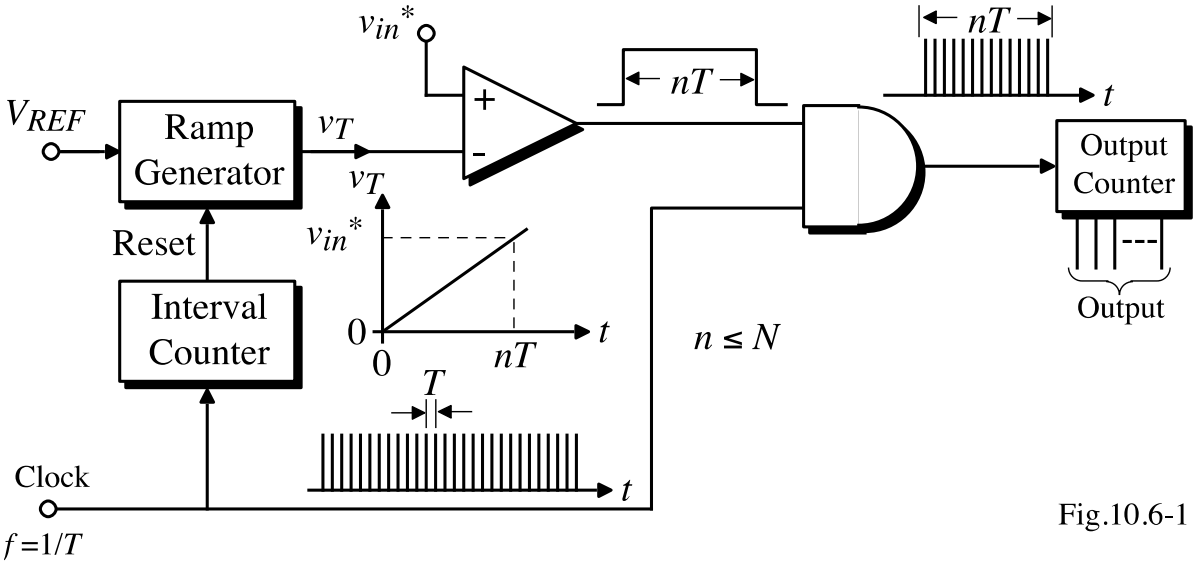


Fig.10.6-1

Attributes:

- Simplicity of operation
- Subject to error in the ramp generator
- Long conversion time $\le 2^N T$

Dual-Slope ADC

Block diagram:

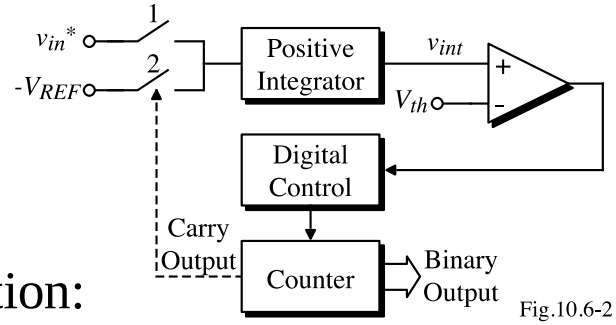


Fig.10.6-2

Waveforms:

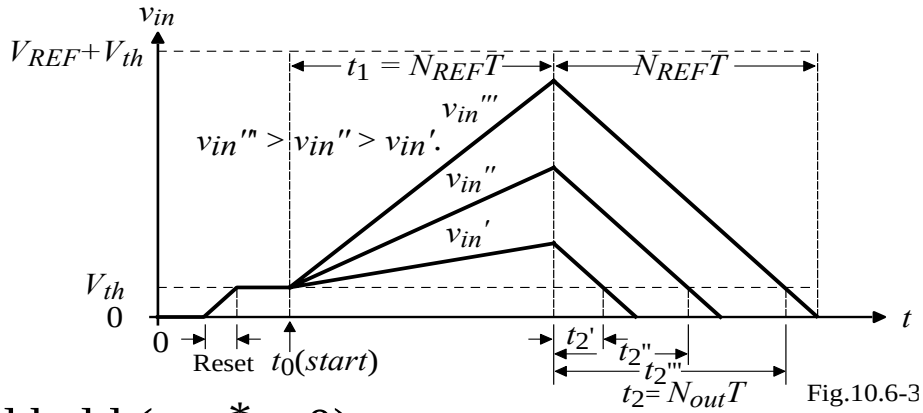


Fig.10.6-3

Operation:

1.) Initially $v_{int} = 0$ and v_{in} is sampled and held ($v_{IN}^* > 0$).

2.) Reset the positive integrator by integrating a positive voltage until $v_{int}(0) = V_{th}$.

3.) Integrate v_{in}^* for N_{REF} clock cycles to get,

$$v_{int}(t_1) = K\int_0^{N_{REF}T} v_{in}^*\,dt + v_{int}(0) = KN_{REF}Tv_{in}^* + V_{th}$$

4.) After N_{REF} counts, the carry output of the counter closes switch 2 and -V_{REF} is applied to the positive integrator. The output of the integrator at $t = t_1+t_2$ is,

$$v_{int}(t_1+t_2) = v_{int}(t_1)+K\int_{t_1}^{N_{out}T}(-V_{REF})dt = V_{th} \rightarrow KN_{REF}Tv_{in}^*+V_{th} - KN_{out}TV_{REF} = V_{th}$$

5.) Solving for N_{out} gives, $N_{out} = N_{REF}\,(v_{in}^*/V_{REF})$

Comments: Conversion time $\leq 2(2^N)T$ and the operation is independent of V_{th} and K.

SUCCESSIVE APPROXIMATION ANALOG-DIGITAL CONVERTERS

Introduction

Successive Approximation Algorithm:

1.) Start with the *MSB* bit and work toward the *LSB* bit.

2.) Guess the *MSB* bit as 1.

3.) Apply the digital word 10000.... to a DAC.

4.) Compare the DAC output with the sampled analog input voltage.

5.) If the DAC output is greater, keep the guess of 1. If the DAC output is less, change the guess to 0.

6.) Repeat for the next *MSB*.

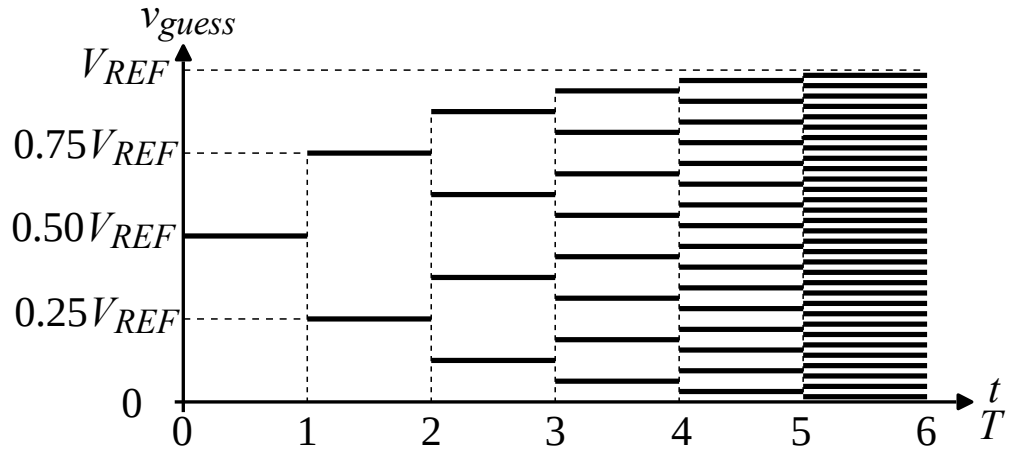


Fig.10.7-2

Block Diagram of a Successive Approximation ADC†

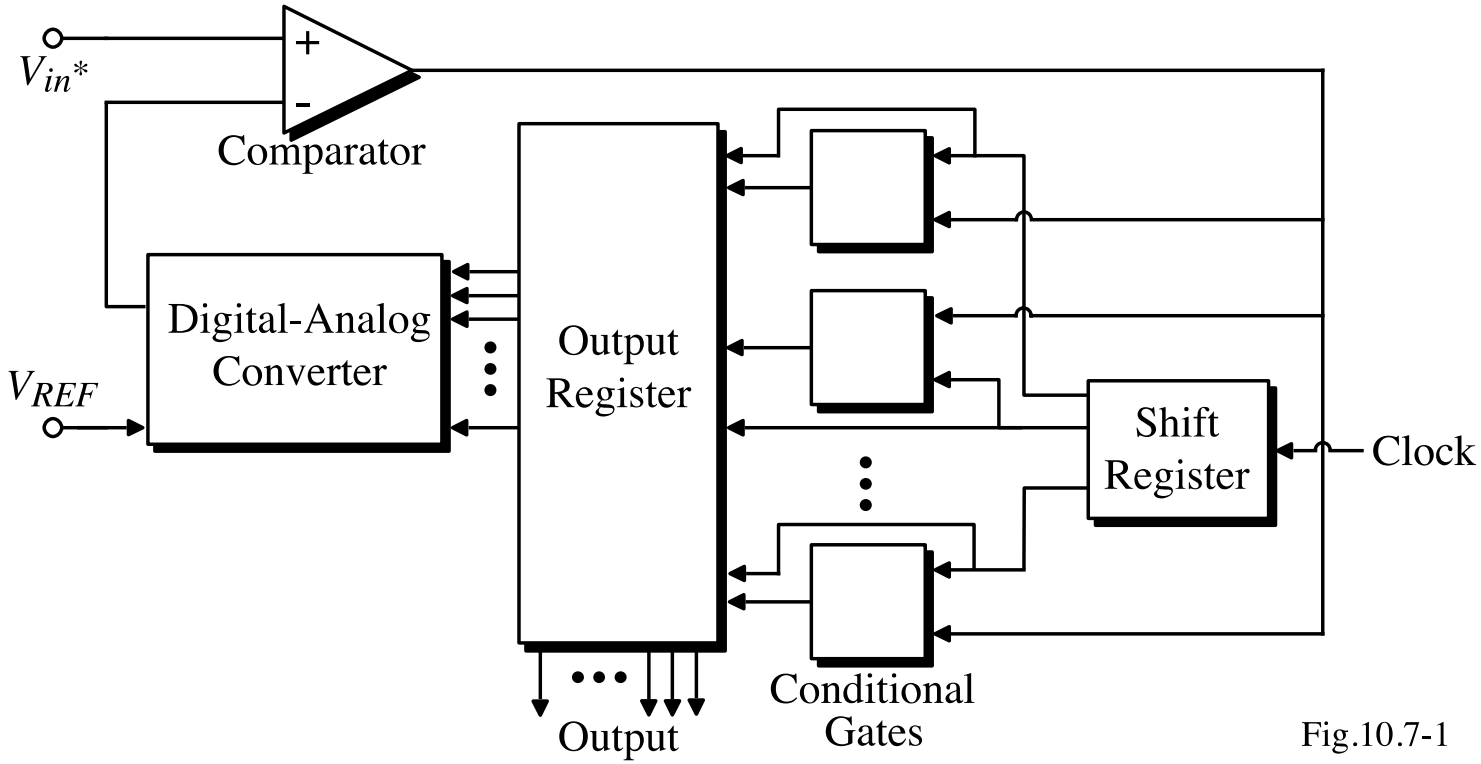


Fig.10.7-1

† R. Hnatek, *A User's Handbook of D/A and A/D Converters*, John Wiley and Sons, Inc., New York, NY, 1976.

5-Bit Successive Approximation ADC

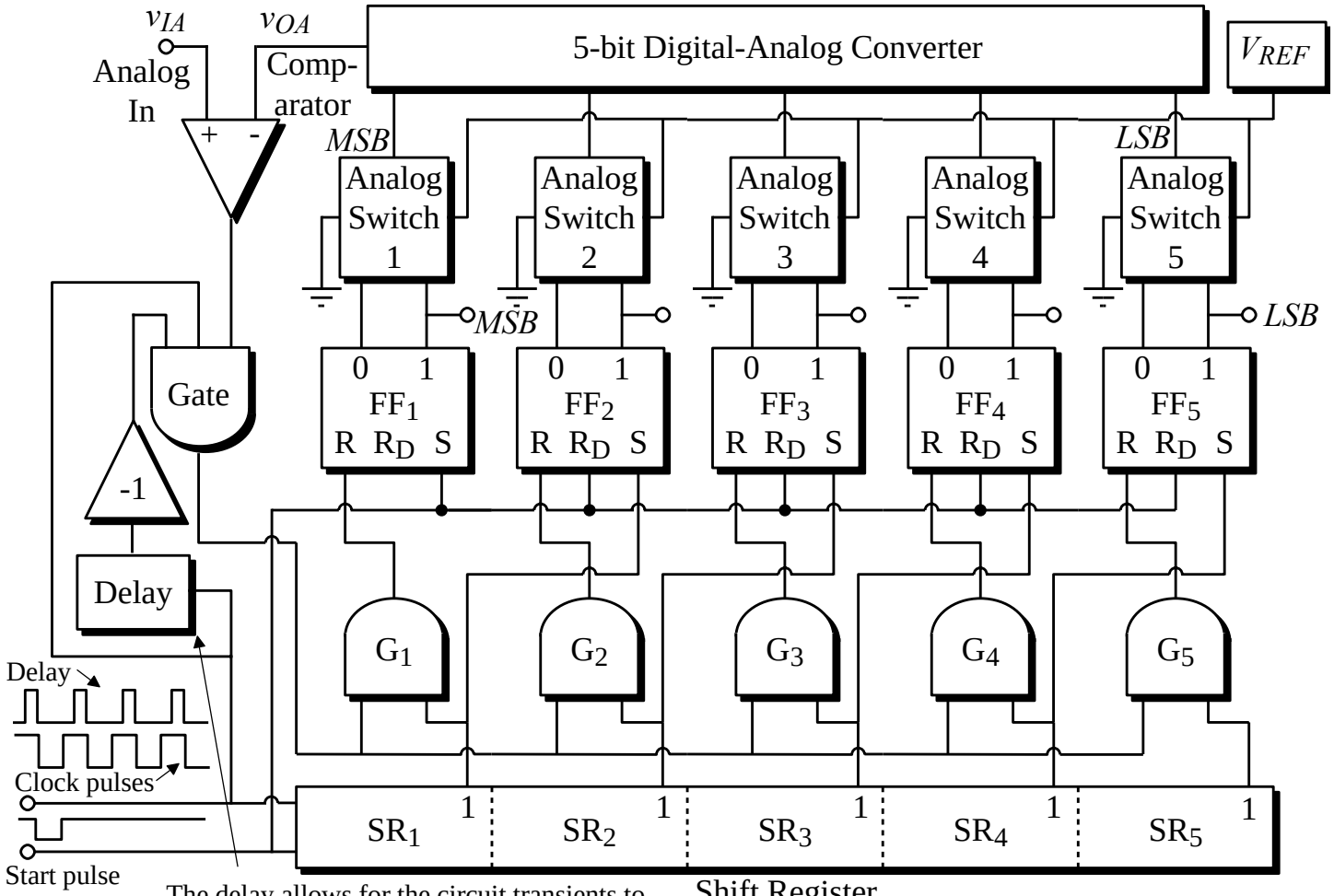


Fig.10.7-3

m-Bit Voltage-Scaling, *k*-Bit Charge-Scaling Successive Approximation ADC

Operation:

1.) With the two S_F switches closed, all capacitors are paralleled and connected to V_{in}^* which autozeros the comparator offset voltage.

2.) With all capacitors still in parallel, a successive approximation search is performed to find the resistor segment in which the analog signal lies.

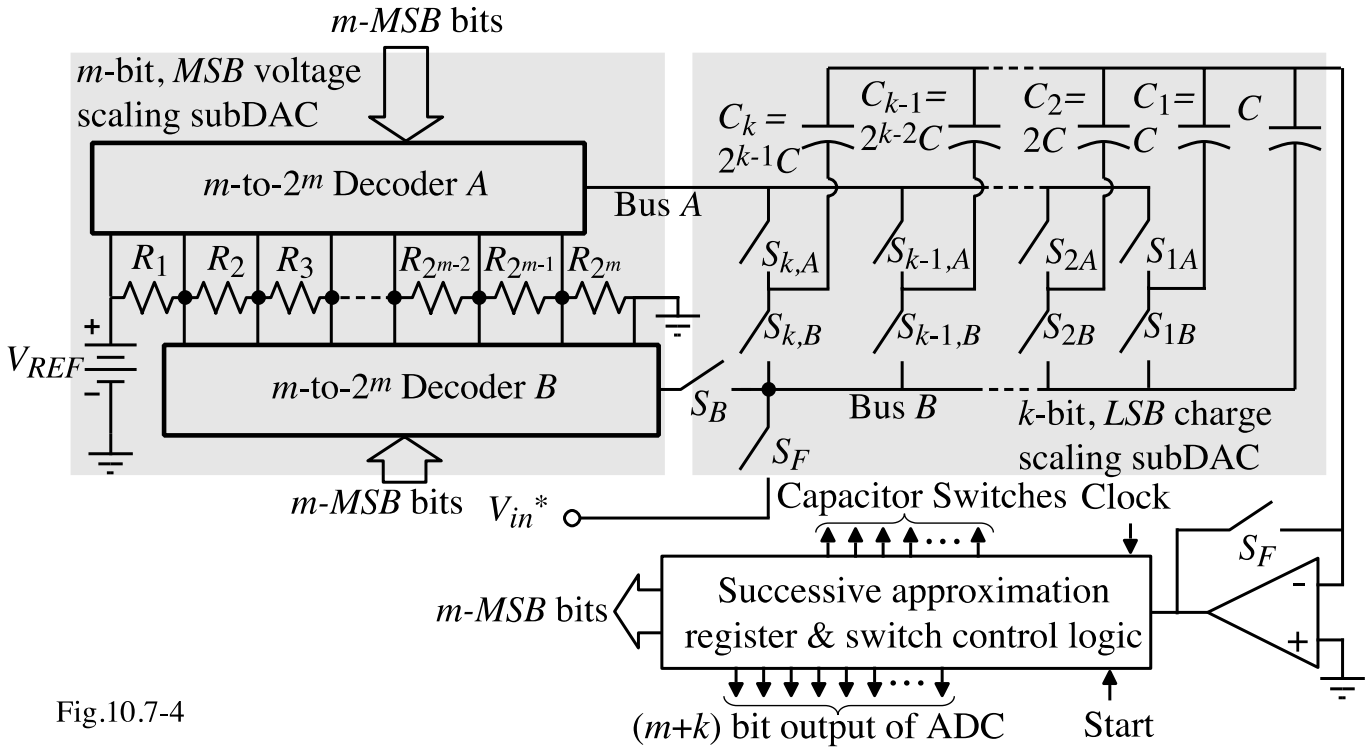


Fig.10.7-4

3.) Finally, a successive approximation search is performed on charge scaling subDAC to establish the analog output voltage.

Voltage-Scaling, Charge-Scaling Successive Approximation ADC - Continued

Autozero Step

Removes the influence of the offset voltage of the comparator.

The voltage across the capacitor is given as,

$$v_C = V_{in}^* - V_{OS}$$

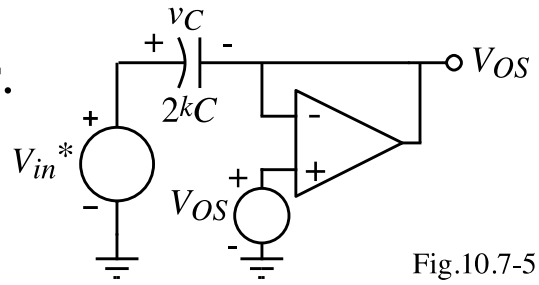


Fig.10.7-5

Successive Approximation Search on the Resistor String

The voltage at the comparator input is

$$v_{comp} = V_{Ri} - V_{in}^*$$

If $v_{comp} > 0$, then $V_{Ri} > V_{in}^*$, if $v_{comp} < 0$, then $V_{Ri} < V_{in}^*$

Successive Approximation Search on the Capacitor SubDAC

The input to the comparator is written as,

$$v_{comp} = (V_{Ri+1} - V_{in}^*)\frac{C_{eq}}{2^kC} + (V_{Ri} - V_{in}^*)\frac{2^kC - C_{eq}}{2^kC}$$

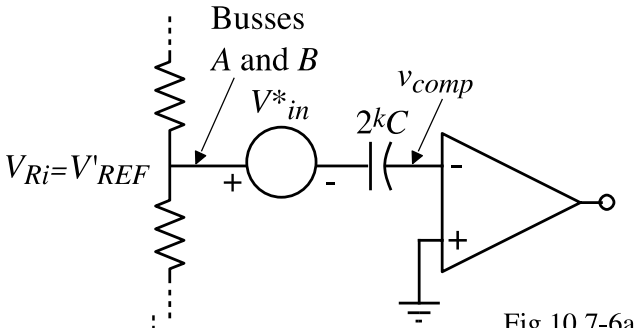


Fig.10.7-6a

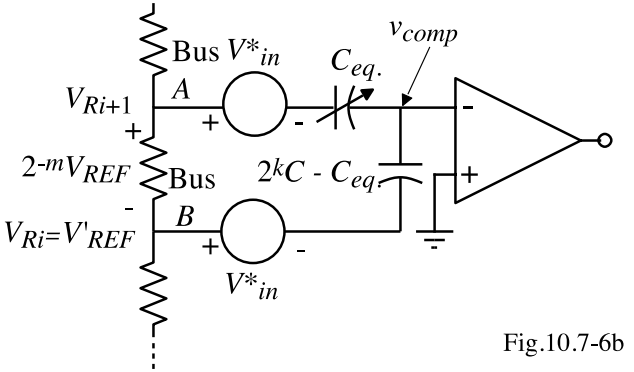


Fig.10.7-6b

However, $V_{Ri+1} = V_{Ri} + 2^{-m}V_{REF}$

Combining gives,

$$v_{comp} = (V_{Ri} + 2^{-m}V_{REF} - V_{IN}^*)\frac{C_{eq}}{2^kC} + (V_{Ri} - V_{IN}^*)\frac{2^kC - C_{eq}}{2^kC}$$

$$= V_{Ri} - V_{IN}^* + 2^{-m}V_{REF}\frac{C_{eq}}{2^kC}$$

SINGLE-BIT/STAGE, PIPELINE ANALOG-DIGITAL CONVERTERS

Single-Bit/Stage Pipeline ADC Architecture

Implementation:

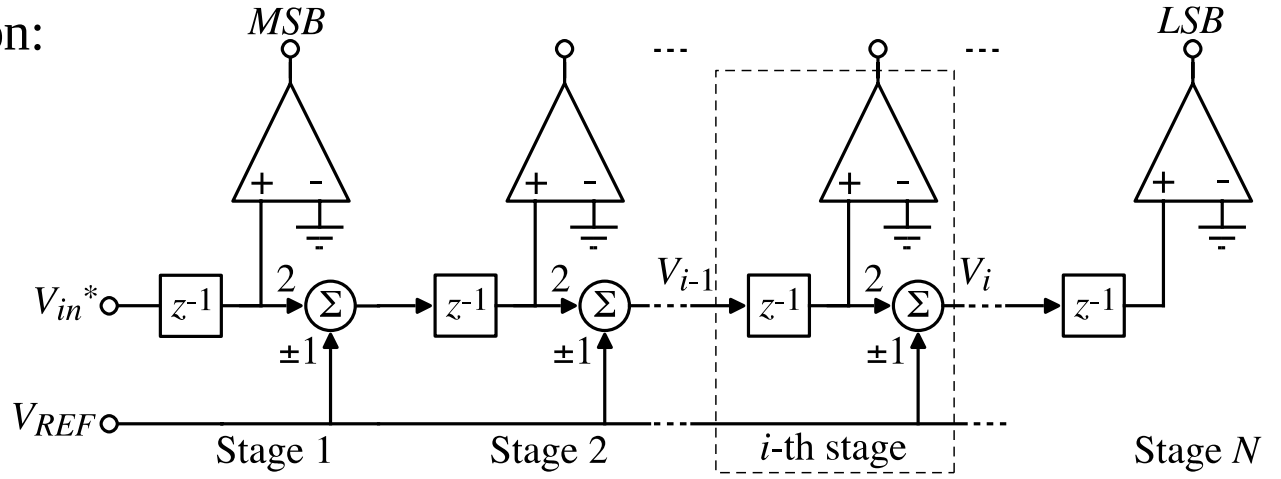


Fig.10.7-9

Operation:

- Each stage multiplies its input by 2 and adds or subtracts V_{REF} depending upon the sign of the input.
- *i*-th stage,

$$V_i = 2V_{i-1} - b_i V_{REF}$$

where b_i is given as

$$b_i = \begin{cases} +1 \text{ if } V_{i-1}>0 \\ -1 \text{ if } V_{i-1}<0 \end{cases}$$

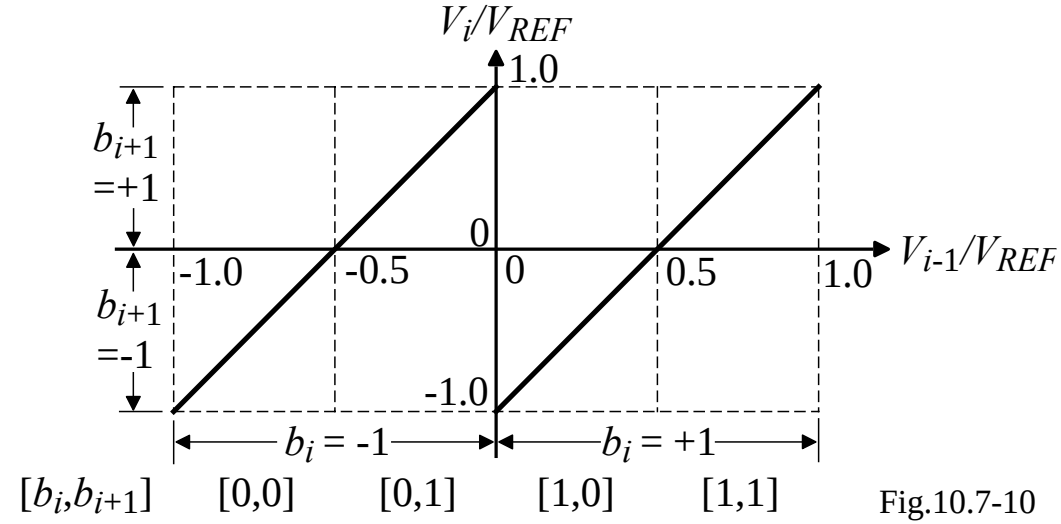


Fig.10.7-10

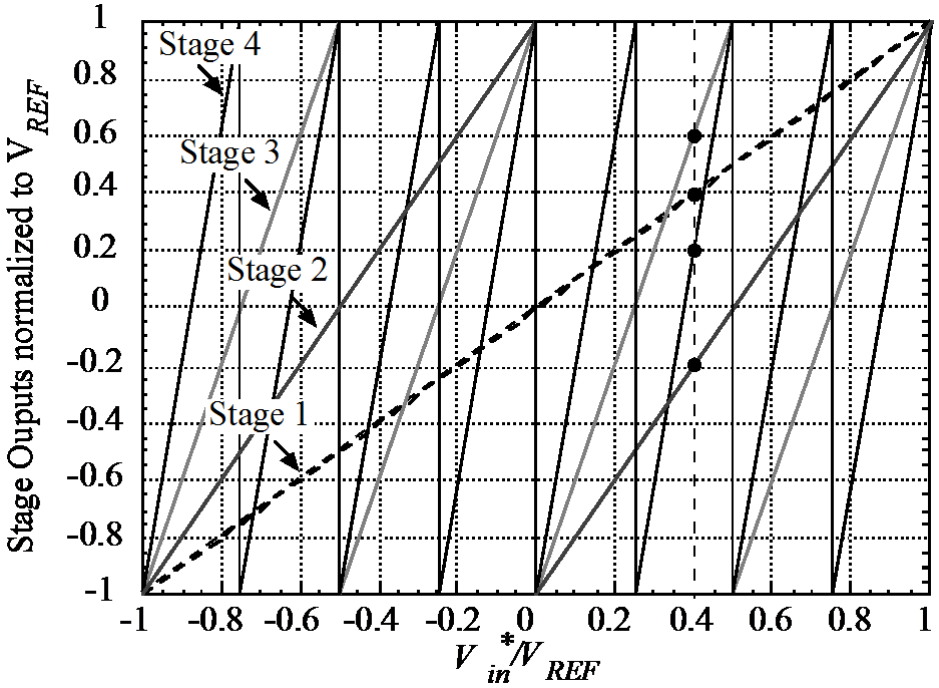
Example 37-1 - Illustration of the Operation of the Pipeline ADC

Assume that the sampled analog input to a 4-bit pipeline analog-digital converter is 2.00 V. If V_{REF} is equal to 5 V, find the digital output word and the analog equivalent voltage.

Solution

Stage No.	Input to the ith stage, V_{i-1}	$V_{i-1} > 0$?	Bit i
1	2V	Yes	1
2	(2V·2) - 5 = -1V	No	0
3	(-1V·2) + 5 = 3V	Yes	1
4	(3V·2) - 5 = 1V	Yes	1

Illustration:



$$V_{\text{analog}} = 5\left(\frac{1}{2} - \frac{1}{4} + \frac{1}{8} + \frac{1}{16}\right)$$
$$= 5(0.4375) = 2.1875$$

where $b_i = +1$ if the ith-bit is 1
and $b_i = -1$ if the ith bit is 0

Achieving the High Speed Potential of the Pipeline ADC

If shift registers are used to store the output bits and align them in time, the pipeline ADC can output a digital word at every clock cycle with a latency of NT.

Illustration:

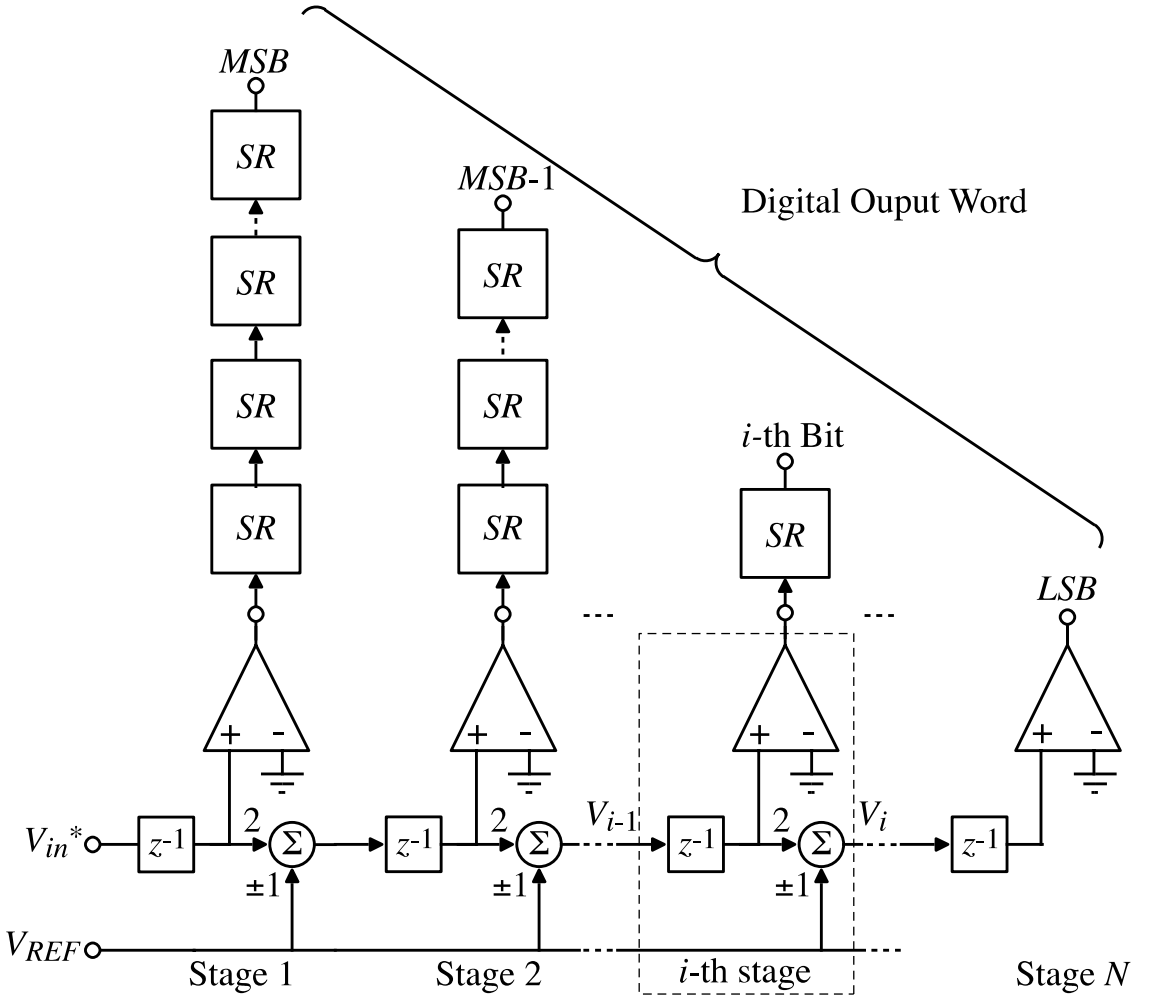


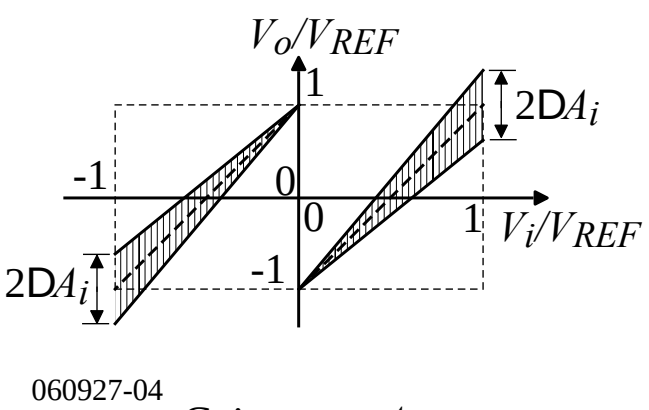
Fig.10.7-9B

Errors in the Pipeline ADC

Types of errors:

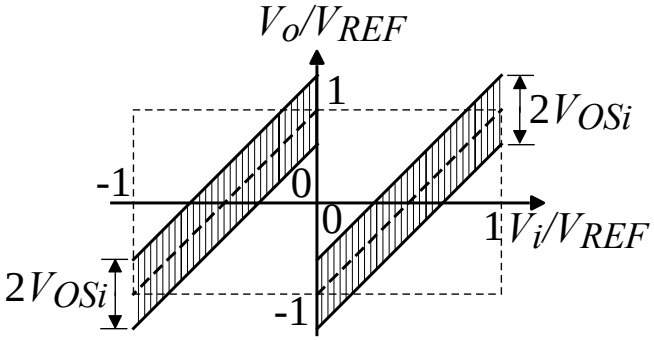
- Gain errors – x2 amplifier or summing junctions
- Offset errors – comparators or summing junctions

Illustration of errors:

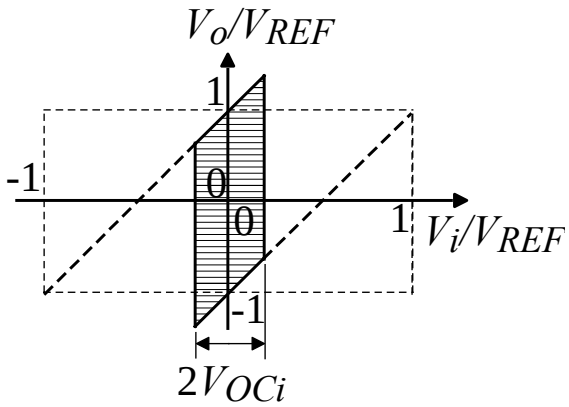


060927-04

Gain error, A_i.



System offset error, V_{OSi}.



Comparator offset error, V_{OCi}.

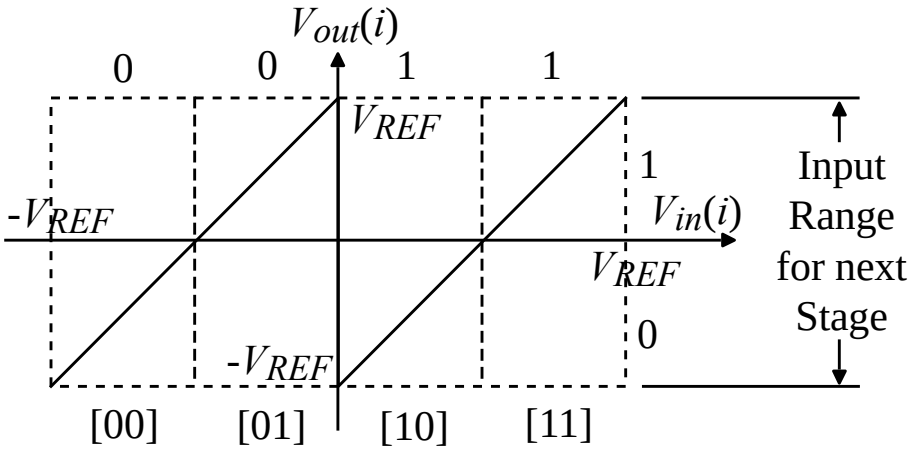
An error will occur if the output voltage of one stage exceeds $\pm V_{REF}$ (saturates).

Digital Error Correction

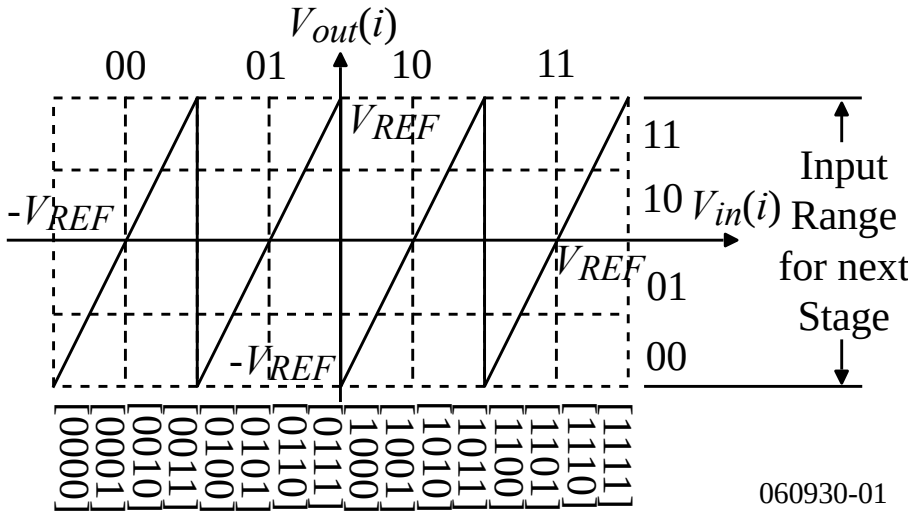
In the previous slide, we noted that if the analog output to the next stage exceeds $\pm V_{REF}$ that an error occurs. This error can be detected by adding one more bit to the following stage for the purposes of detecting the error.

Illustration (2nd bit not used for error correction):

Input/output characteristics of a 1-bit stage

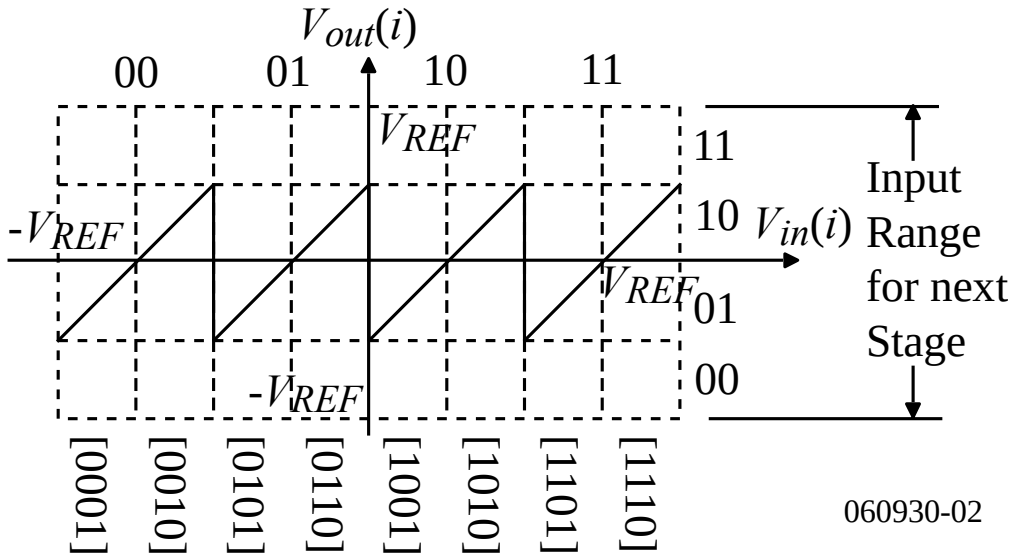


Input/output characteristics of a 2-bit stage



Digital Error Correction – Continued

If the gain of 4 amplifier is reduced back to 2, the input/output characteristics of the 2-bit stage become:

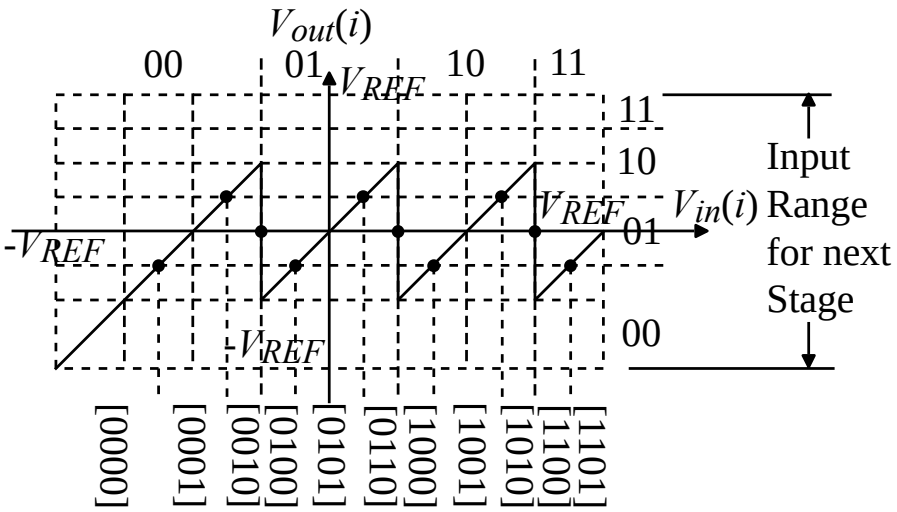


The output bits can be used to determine the error. If these bits are 00, then 0.5*LSB* must be added to get the correct digital output. If the bits are 11, then 0.5*LSB* must be subtracted to get the correct digital output.

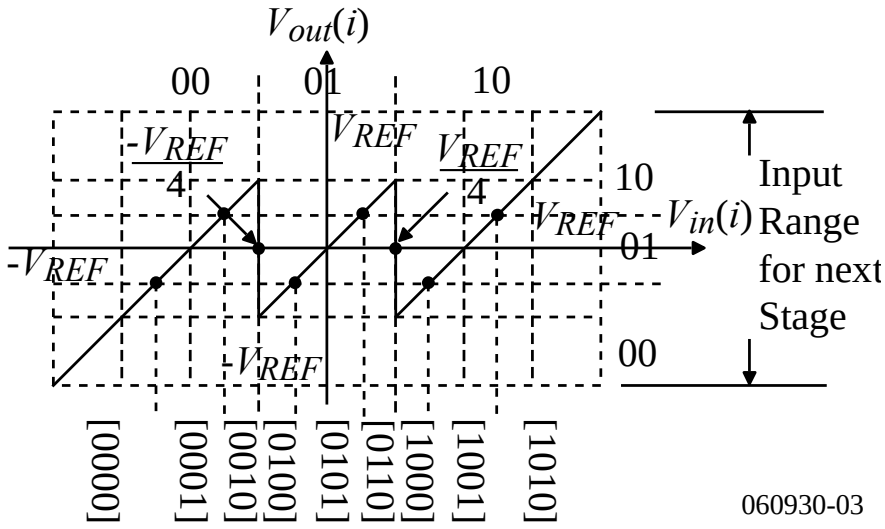
Modified Digital Error Correction (1.5 bits per stage)

In the previous slide, it was necessary sometimes to perform digital subtraction which is not easy to implement. To avoid this problem, a $0.5LSB$ shift has been added to the input/output characteristic resulting in the following.

Movement of all comparator thresholds to the right by $0.5LSB$.



Removal of the comparator at $0.75LSB$.



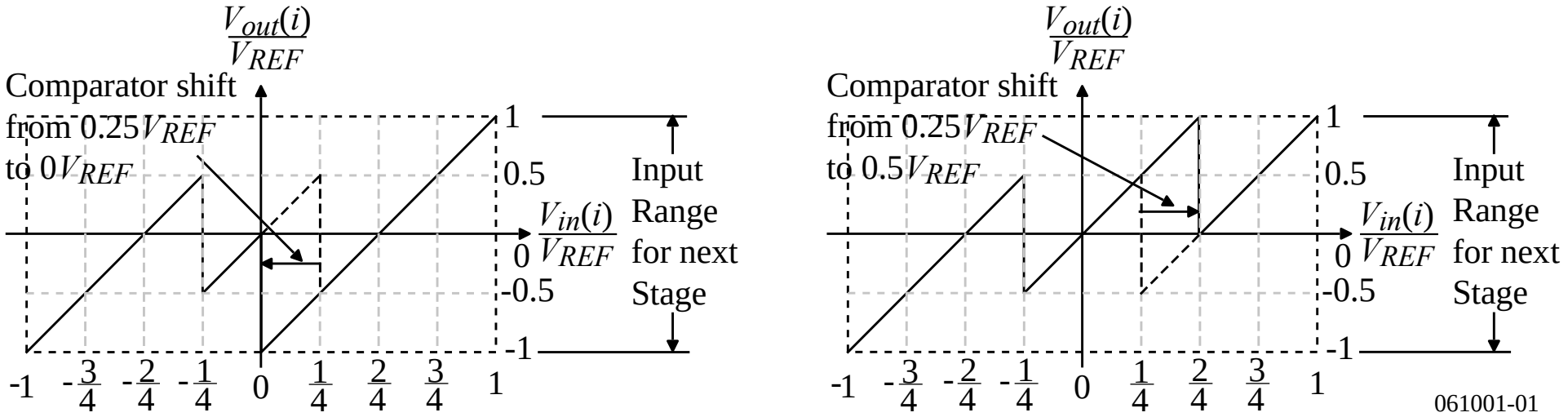
To obtain code 11 out of the stage after correction, the correction logic must increment the output of the stage.

To obtain code 00 from this stage after correction, the correction logic need do nothing.

Therefore, only two comparators are needed to produce outputs of (00, 01, 10) as shown on the right-hand characteristic.

How Does the 1.5 Bit Stage Correct Offset Errors?

Consider a $\pm 0.25 V_{REF}$ comparator offset shift in the input-output characteristics of the 1.5 bit stage.

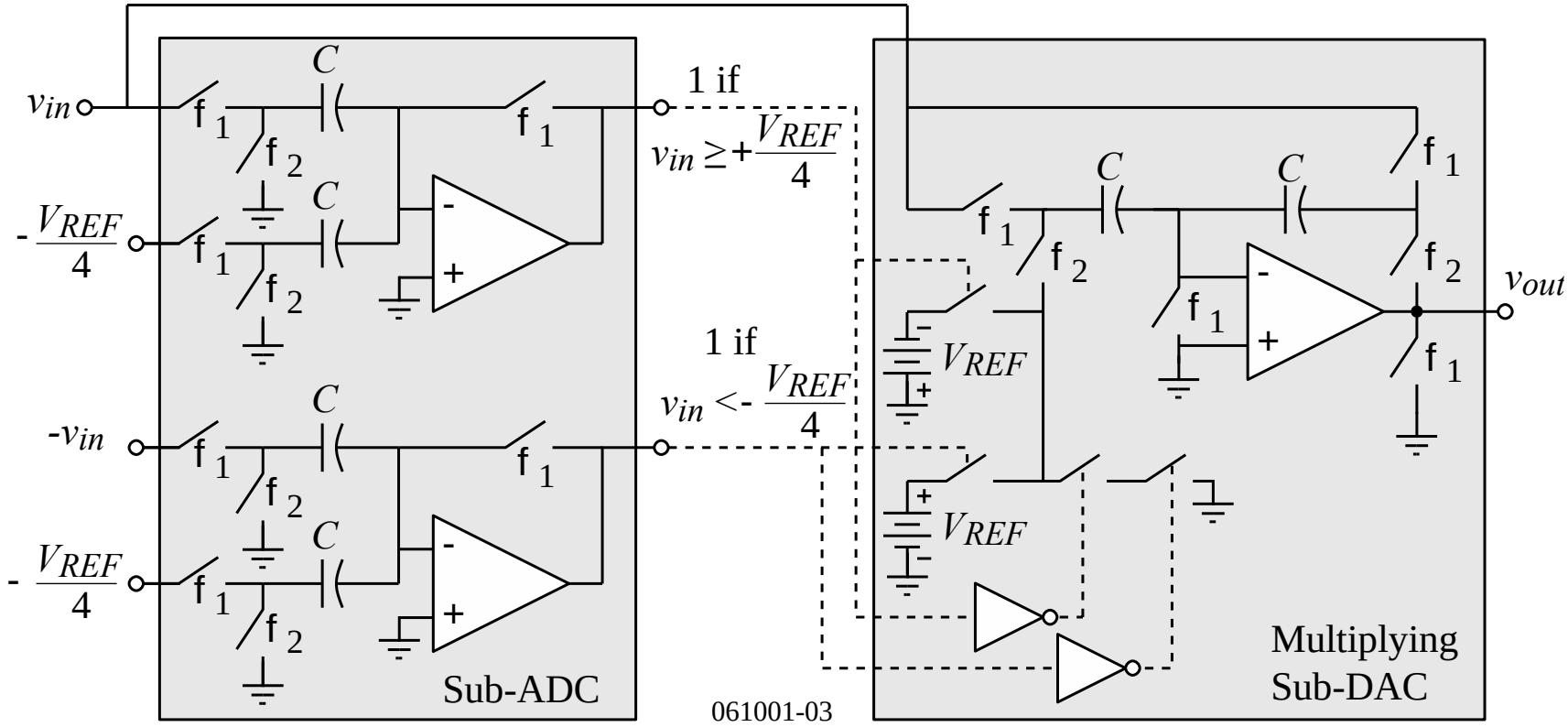


When the shift is to the left, the comparator will not be in error until the shift is greater than 0.25 V_{REF}. This is because the comparator thresholds were shifted to the right by 0.5 V_{REF}.

When the shift is to the right, the input to the next stage will be greater than $0.50 V_{REF}$. This will cause the output code 10 which indicates that the digital word should be incremented by 1 bit.

The range of correction $\pm V_{REF} / 2^{B+1}$ where B is the number of bits per stage.

Implementation of the 1.5 Bit Stage



The multiplying Sub-DAC must implement the following equation:

$$V_{out} = \begin{cases} 2 \cdot v_{in} - V_{REF} & \text{if} \quad v_{in} > V_{REF}/4 \\ 2 \cdot v_{in} & \text{if} \quad -V_{REF}/4 \leq v_{in} \leq V_{REF}/4 \\ 2 \cdot v_{in} + V_{REF} & \text{if} \quad v_{in} < -V_{REF}/4 \end{cases}$$

Example 37-2 - Accuracy requirements for a 5-bit pipeline ADC

Show that if $V_{in} = V_{REF}$, that the pipeline ADC will have an error in the 5th bit if the gain of the first stage is 2-(1/8) =1.875 which corresponds to when an error will occur. Show the influence of V_{in} on this result for V_{in} of $0.65V_{REF}$ and $0.22V_{REF}$.

Solution

For $V_{in} = V_{REF}$, we get the results shown below. The input to the fifth stage is 0V which means that the bit is uncertain. If A_1 was slightly less than 1.875, the fifth bit would be 0 which is in error. This result assumes that all stages but the first are ideal.

i	V_i(ideal)	Bit i (ideal)	$V_i(A_1=1.875)$	Bit i (A_1=1.875)
1	1	1	1.000	1
2	1	1	0.875	1
3	1	1	0.750	1
4	1	1	0.500	1
5	1	1	0.000	?

Now let us repeat the above results for $V_{in} = 0.65V_{REF}$. The results are shown below.

i	V_i(ideal)	Bit i (ideal)	$V_i(A_1=1.875)$	Bit i (A_1=1.875)
1	+0.65	1	0.6500	1
2	+0.30	1	0.2188	1
3	-0.40	0	-0.5625	0
4	+0.20	1	-0.1250	0
5	-0.60	0	0.7500	1

Example 37-2 - Continued

Next, we repeat for the results for $V_{in} = 0.22V_{REF}$. The results are shown below. We see that no errors occur.

i	V_i(ideal)	Bit i (ideal)	$V_i(A_1=1.875)$	Bit i (A_1=1.875)
1	+0.22	1	0.2200	1
2	-0.56	0	-0.5875	0
3	-0.12	0	-0.1750	0
4	+0.76	1	0.6500	1
5	+0.52	1	0.3000	1

Note the influence of V_{in} in the fact that an error occurs for A_1= 1.875 for $V_{in} = 0.65V_{REF}$ but not for $V_{in} = 0.22V_{REF}$. Why? Note on the plot for the output of each stage, that for $V_{in} = 0.65V_{REF}$, the output of the fourth stage is close to 0V so any small error will cause problems. However, for $V_{in} = 0.22V_{REF}$, the output of the fourth stage is at $0.65V_{REF}$ which is further away from 0V and is less sensitive to errors.

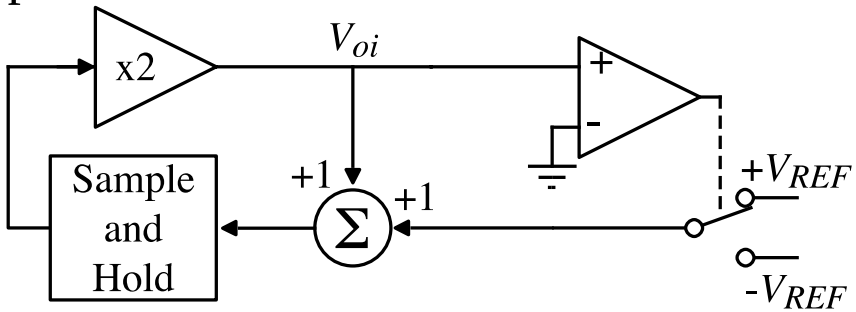
$\therefore$ *The most robust values of* V_{in} *will be near* $-V_{REF}$ *, 0 and* $+V_{REF}$ *or when each stage output is furthest from the comparator threshold, 0V.*

ITERATIVE ANALOG-DIGITAL CONVERTERS

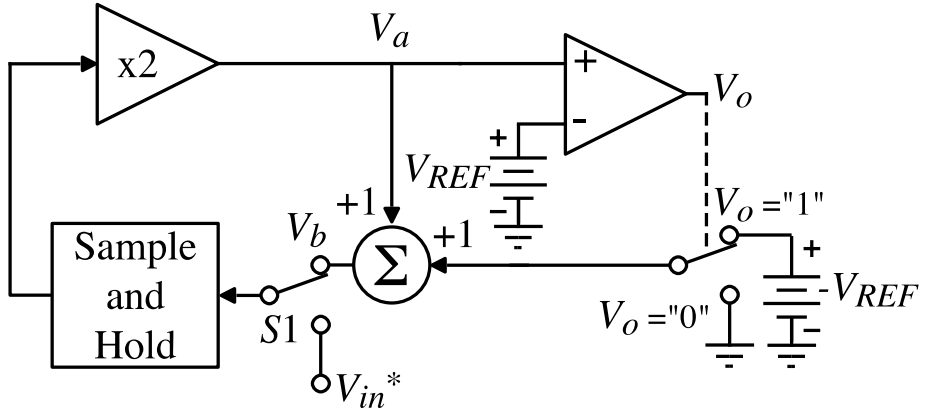
Iterative (Cyclic) Algorithmic Analog-Digital Converter

The pipeline ADC can be reduced to a single stage that cycles the output back to the input.

Implementation:



Iterative algorithm ADC



Different version of iterative algorithm ADC implementation

Fig. 10.7-13

Operation:

1.) Sample the input by connecting switch $S1$ to V_{in}*.

2.) Multiply V_{in}* by 2.

3.) If $V_a > V_{REF}$, set the corresponding bit = 1 and subtract V_{REF} from V_a.
If $V_a < V_{REF}$, set the corresponding bit = 0 and add zero to V_a.

4.) Repeat until all N bits have been converted.

Example 37-3 - Conversion Process of an Iterative, Algorithmic Analog-Digital Converter

The iterative, algorithmic analog-digital converter is to be used to convert an analog signal of $0.8V_{REF}$. The figure below shows the waveforms for V_a and V_b during the process. T is the time for one iteration cycle.

1.) The analog input of $0.8V_{REF}$ gives $V_a = 1.6V_{REF}$ and $V_b = 0.6V_{REF}$ and the *MSB* as 1.

2.) V_b is multiplied by two to give $V_a = 1.2V_{REF}$. The next bit is also 1 and $V_b = 0.2V_{REF}$.

3.) The third iteration gives $V_a = 0.4V_{REF}$, making the next bit is 0 and $V_b = 0.4V_{REF}$.

4.) The fourth iteration gives $V_a = 0.8V_{REF}$, giving $V_b = 0.8V_{REF}$ and the fourth bit as 0.

5.) The fifth iteration gives $V_a = 1.6V_{REF}$, $V_b = 0.6V_{REF}$ and the fifth bit as 1.

The digital word after the fifth iteration is 11001 and is equivalent to an analog voltage of $0.78125V_{REF}$.

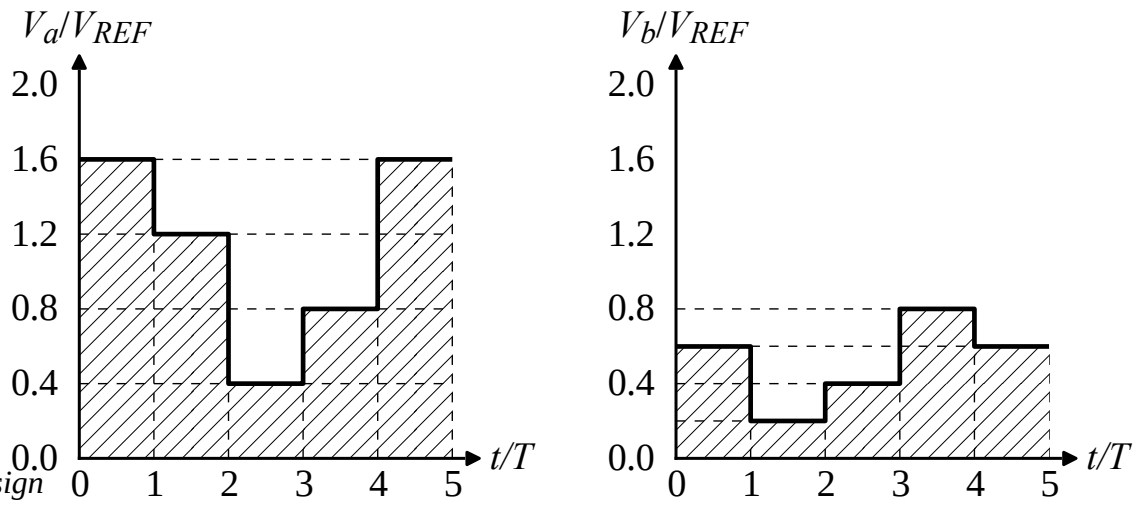


Fig. 10.7-14.

SELF-CALIBRATION TECHNIQUES

Self-Calibrating Analog-Digital Converters

Self-calibration architecture for a *m*-bit charge scaling, *k*-bit voltage scaling successive approximation ADC

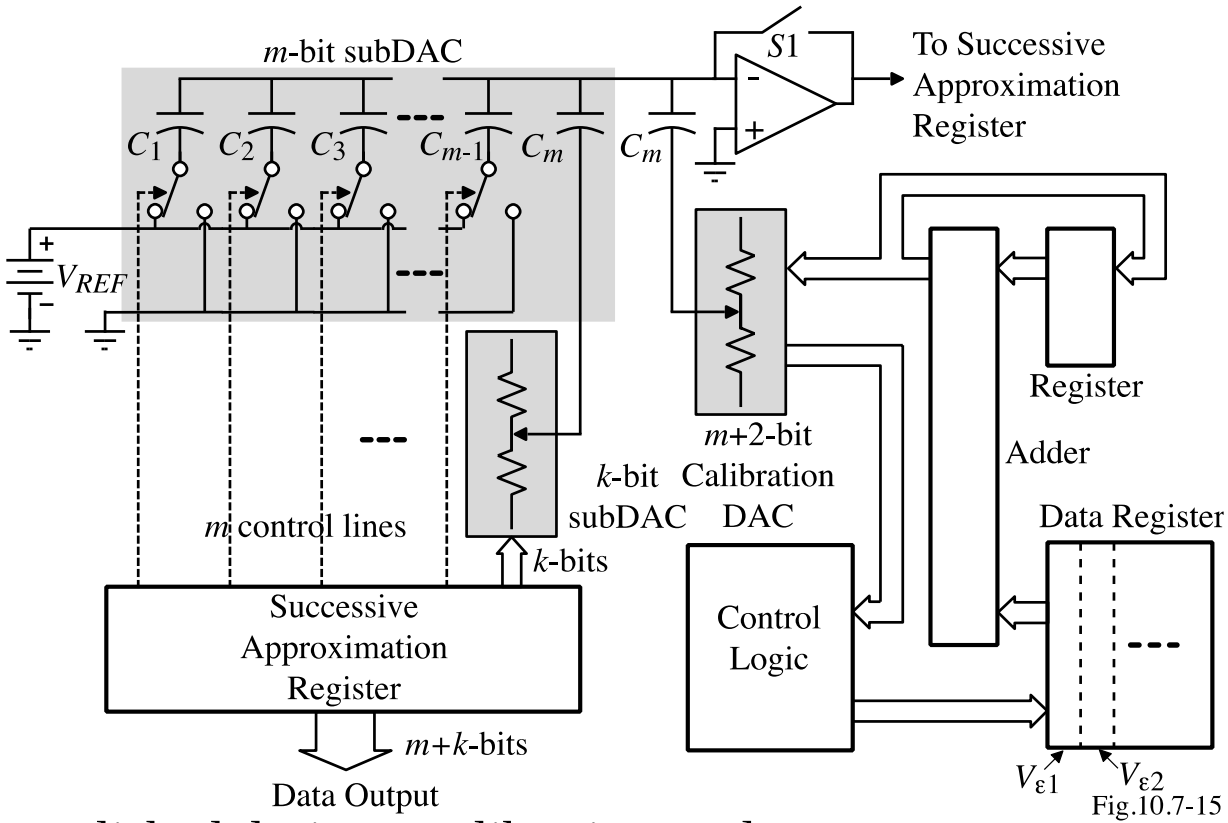


Fig.10.7-15

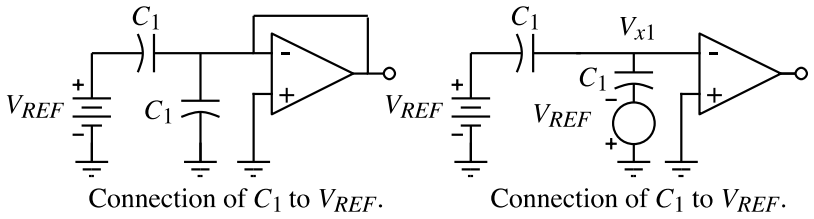
Comments:

- Self-calibration can be accomplished during a calibration cycle or at start-up
- In the above scheme, the *LSB* bits are not calibrated
- Calibration can extend the resolution to 2-4 bits more that without calibration

Self-Calibrating Analog-Digital Converters - Continued

Self-calibration procedure starting with the *MSB* bit:

1.) Connect C_1 to V_{REF} and the remaining capacitors ($C_2+C_3+\cdots+C_m+C_m = \overline{C_1}$) to ground and close S_F.



Connection of C_1 to V_{REF}. Connection of C_1 to V_{REF}.
Fig.10.7-16

2.) Next, connect C_1 to ground and $\overline{C_1}$ to V_{REF}.

3.) The result will be $V_{x1} = \left(\frac{\overline{C_1} - C_1}{C_1 + \overline{C_1}}\right) V_{REF}$. If $C_1 = \overline{C_1}$, then $V_{x1} = 0$.

4.) If $V_{x1} \neq 0$, then the comparator output will be either high or low. Depending on the comparator output, the calibration circuitry makes a correction through the calibration DAC until the comparator output changes. At this point the *MSB* is calibrated and the *MSB* correction voltage, $V_{\varepsilon 1}$ is stored.

5.) Proceed to the next *MSB* with C_1 out of the array and repeat for C_2 and $\overline{C_2}$. Store the correction voltage, $V_{\varepsilon 2}$, in the data register.

6.) Repeat for C_3 with C_1 and C_2 out of the array. Continue until all of the capacitors of the *MSB* DAC have been corrected.

Note: For normal operation, the circuit adds the correct *combined* correction voltage.

SUMMARY

- Tests for the ADC include:
  - Input-output test
  - Spectral test
  - FFT test
  - Histogram test
- Moderate Speed ADCs:

Type of ADC	Advantage	Disadvantage
Serial ADC	High resolution	Slow
Voltage-scaling, charge-scaling successive approximation ADC	High resolution	Requires considerable digital control circuitry
Successive approximation using a serial DAC	Simple	Slow
Pipeline ADC	Fast after initial latency of NT	Accuracy depends on input
Iterative algorithmic ADC	Simple	Requires other digital circuitry

- Successive approximation ADCs also can be calibrated extending their resolution 2-4 bits more than without calibration.