Chapter

30

Implementing Data Converters

CMOS technology continues to scale towards smaller dimensions. This feature size reduction is driven mainly by the desire to implement digital systems of increased complexity in a smaller area. This natural trend in feature size reduction, with accompanying reduction in supply voltage and poorer matching, can present challenges for the CMOS circuit designer. The accompanying lower supply voltage, for example, results in an inherent reduction in dynamic range, decrease in *SNR*, and increasing challenges when implementing analog circuitry with little, ideally zero, voltage overhead.

This chapter presents and discusses implementation methods and trade-offs for designing data converters in nanometer CMOS. For DAC design, we focus on converters implemented with both resistors using *R*-2*R* networks and current sources. The benefit of, and reason we are focusing on, using *R*-2*R* networks and current sources over other methods for DAC implementation, such as charge redistribution DACs, is the absence of good poly-poly capacitors in nanometer digital CMOS processes. *R*-2*R*-based DACs can be laid out in a small area while achieving resolutions in excess of 10-bits without calibrations or trimming. Charge-scaling DACs require linear capacitors. The layout area needed for these capacitors can often be very large and practically limit both the resolution and accuracy of the DAC.

The first section of this chapter discusses *R*-2*R* and current-steering DACs. The second section of the chapter discusses the use of op-amps in data converters, while the third section presents an overview of general ADC implementations in nanometer CMOS processes. In this last section we concentrate our discussion on the implementation of pipeline analog-to-digital data converters.

It's important to note that the goal in this chapter is not to provide an exhaustive overview of data converter design but rather to provide discussions and practical insight helpful when implementing any type of data converter in CMOS technology. We assume that the reader is familiar with data converter fundamentals, Ch. 28, and data converter architectures, Ch. 29. For example, the reader knows the difference between differential nonlinearity (DNL) and integral nonlinearity (INL) or the difference between a two-step flash ADC and a pipeline ADC.

30.1 *R-2R* Topologies for DACs

We begin this section by discussing *R*-2*R* DAC topologies. The problems encountered in the traditional *R*-2*R* topologies with low-voltage overhead are illustrated. Also, concerns related to the performance of the op-amps used in data converters (both ADCs and DACs) are discussed. Finally, matching and accuracy concerns are presented along with techniques to remove these imperfections using calibration.

30.1.1 The Current-Mode *R-2R* DAC

The *R*-2*R* DAC can be classified into two categories: voltage-mode and current-mode. A current-mode *R*-2*R* DAC is shown in Fig. 30.1. The branch currents flowing through the 2*R* resistors are of a binary-weighted relationship caused by the voltage division of the *R*-2*R* ladder network and are diverted either to the inverting input of the op-amp (actually the feedback resistor) or the noninverting input of the op-amp (actually V_{REF-}). The voltage on the *R*-2*R* resistor string at the X^{th} tap (where X ranges from 0 to $N-1$), in Fig. 30.1, can be written as

$$V_{TAPX} = \frac{2^X}{2^N} \cdot (V_{REF+} - V_{REF-}) + V_{REF-} \tag{30.1}$$

where V_{REF+} and V_{REF-} are the N-bit DAC's reference voltages. The current that flows through the 2*R* resistor at the X^{th} tap is then

$$I_{TAPX} = \frac{V_{TAPX} - V_{REF-}}{2R} = \frac{1}{2R} \cdot \frac{2^X}{2^N}(V_{REF+} - V_{REF-}) \tag{30.2}$$

This current is summed at the inverting input of the op-amp and flows through the feedback resistor to the DAC output, V_{out}. The output voltage of the DAC can be written as

$$V_{out} = V_{REF-} - R \cdot \sum_{X=0}^{N-1} (b_X \cdot I_{TAPX}) \text{ for } V_{REF+} > V_{REF-} \tag{30.3}$$

where b_X is either a 1 or 0, or

$$V_{out} = V_{REF-} + R \cdot \sum_{X=0}^{N-1} (b_X \cdot I_{TAPX}) \text{ for } V_{REF+} < V_{REF-} \tag{30.4}$$

Using these equations, we can see the main problem with the basic current mode topology of Fig. 30.1 in a nanometer CMOS process using low-power supply voltages, namely, limited output swing. If V_{REF-} is set to 0 V, with $V_{REF+} > 0$, then the output of the DAC must be negative, which, of course, can't happen when the only power supply voltage is (positive) *VDD*. If V_{REF-} is set to *VDD*, then we can see from Eq. (30.4) that this would require $V_{out} > VDD$. After reviewing Eqs. (30.1)-(30.4), we see that the range of output voltages associated with the current mode *R*-2*R* DAC is limited to *VDD*/2, e.g., 0 to *VDD*/2, *VDD*/2 to *VDD* or 0.25*VDD* to 0.75*VDD*, etc. Giving up half of the power-supply range in a DAC and correspondingly reducing the dynamic range, is usually not desirable.

By removing the requirement that the noninverting input of the op-amp be tied to V_{REF-} and that the feedback resistor be *R* (the same value used in the *R*-2*R* string), we can increase the output range of the DAC. The output of the op-amp is level-shifted by the voltage on the noninverting input of the op-amp and by the increased value of closed-loop gain of the op-amp. Similarly, we could add a gain stage to the output of the DAC (two

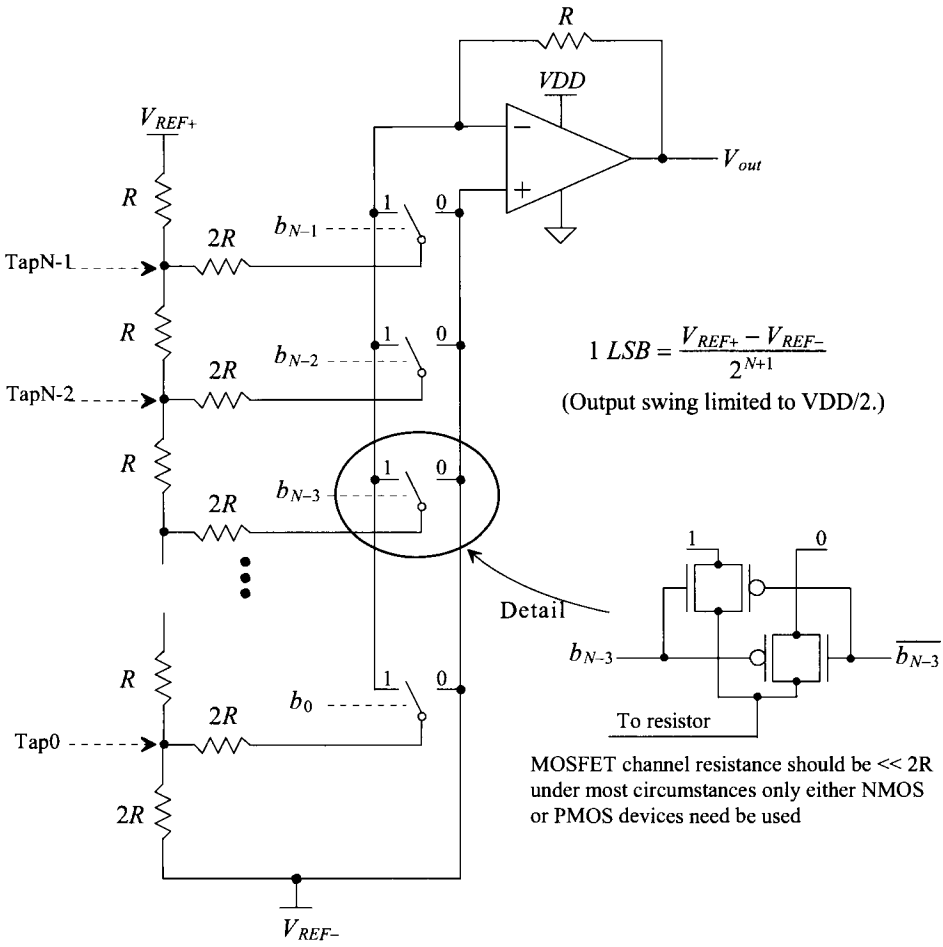


Figure 30.1 Traditional current-mode R-2R DAC.

op-amps would then be used) to achieve wider DAC output swing. We don't cover these options any further here because they either put more demand on the op-amp design, such as increased op-amp open-loop gain/speed, increased power consumption, or won't, in a practical implementation, result in a rail-to-rail output swing.

30.1.2 The Voltage-Mode *R-2R* DAC

Figure 30.2 shows a schematic of a voltage-mode DAC. The voltage on the non-inverting input of the op-amp can be written as

$$V_+ = \frac{b_{N-1} \cdot V_{REF+} + \overline{b_{N-1}} \cdot V_{REF-}}{2^1} + \frac{b_{N-2} \cdot V_{REF+} + \overline{b_{N-2}} \cdot V_{REF-}}{2^2} + \ldots + \frac{V_{REF-}}{2^N} \qquad (30.5)$$

or, in general terms,

$$V_+ = \sum_{k=1}^{N} \frac{b_{N-k} \cdot V_{REF+} + \overline{b_{N-k}} \cdot V_{REF-}}{2^k} + \frac{V_{REF-}}{2^N} \tag{30.6}$$

The output of the N-bit voltage-mode DAC can be written as

$$V_{out} = \left[1 + \frac{R_F}{R_I}\right] \cdot \left[\sum_{k=1}^{N} \frac{b_{N-k} \cdot V_{REF+} + \overline{b_{N-k}} \cdot V_{REF-}}{2^k} + \frac{V_{REF-}}{2^N}\right] \tag{30.7}$$

If the input code is all zeroes, with $V_{REF-} = 0$, $V_{REF+} = VDD$, and the op-amp in the follower configuration, then $V_{out} = V_{REF-}$. If the input code is all ones, then the output of the DAC is $V_{REF+} - 1$ LSB.

By using the voltage-mode DAC, we would seem to have solved the problem of the limited output swing associated with the current-mode DAC of Fig. 30.1. However, consider how the finite common-mode rejection ratio (*CMRR*) of the op-amp in Fig. 30.2 can affect the linearity of the overall DAC design. We know the effects of finite *CMRR* can be modeled as a variable offset voltage, ΔV_{OS} (see Ch. 24), in series with the noninverting input of the op-amp that is a function of the change in the op-amp common-mode voltage, ΔV_C, or

$$\Delta V_{OS} = \frac{\Delta V_C}{CMRR} \tag{30.8}$$

We should see the problem at this point: that is, ΔV_{OS} is in series with the *R*-2*R* resistor string and will ultimately limit the linearity of the DAC. To further illustrate the problem, let's assume the *CMRR* of the op-amp in Fig. 30.2 is 20 dB at 1 MHz. Since the common-mode voltage on the input of the op-amp, again assuming $V_{REF+} = VDD$, $V_{REF-} = 0$, and the op-amp in the unity follower configuration can range from zero to approximately *VDD*, the change in the offset voltage used to model finite *CMRR* when the DAC's inputs are changing at 1 MHz is 10% of *VDD*. At first glance we might simply consider the resulting offset as a nonlinear gain error affecting only the large-signal linearity (INL). However, it is unlikely in any practical op-amp design that the *CMRR* will vary linearly with changes in the input common-mode voltage and so the small-signal linearity (DNL) will be affected as well. Since, for this example, 1 LSB = $VDD/2^N$, the resolution of the DAC, because of the finite *CMRR* and assuming 1 LSB > ΔV_{OS}, is limited to 4 bits! Performing DC or audio-frequency tests on the voltage-mode DAC made with an op-amp with a *CMRR* of, for example, 120 dB at DC results in no practical resolution limit (indicating that if DAC speed isn't a concern, the voltage-mode configuration may still be used for high resolutions). Note how *CMRR* isn't a concern with the current-mode *R*-2*R* DAC (assuming no secondary effects, such as common mode substrate noise, are present on the input of the op-amp). *For precision, high-speed data converter design we must use an inverting op-amp topology where the inputs of the op-amp remain at a fixed voltage.*

30.1.3 A Wide-Swing Current-Mode R-2R DAC

We've shown that it is desirable to have a wide output swing, as is provided by the voltage-mode *R*-2*R* DAC, while at the same time having a fixed input common mode voltage, as is provided by the current-mode *R*-2*R* DAC. Figure 30.3 shows a wide-swing

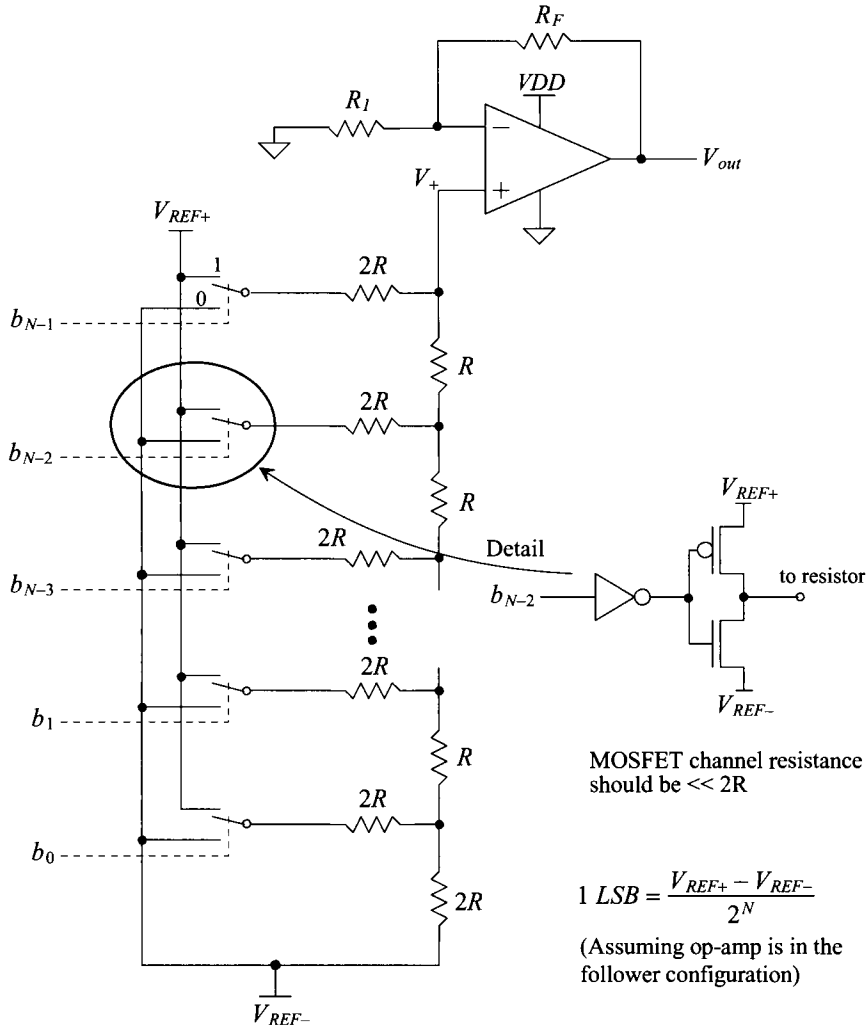


Figure 30.2 Traditional voltage-mode R-2R DAC.

current-mode R-$2R$ DAC configuration that has a rail-to-rail output swing while keeping the input common-mode voltage of the op-amp fixed at the common mode voltage, V_{CM}, or $(V_{REF+} + V_{REF-})/2$.

Like traditional current-mode R-$2R$ DACs, the DAC scheme shown in Fig. 30.3 operates on currents. Using superposition and assuming V_{REF-} is the reference for calculations, the current flowing in the feedback resistor, R_F, is given by

$$I_F = -\frac{V_{REF+} - V_{REF-}}{2R} + \frac{V_{REF+} - V_{REF-}}{2R} \cdot \left[1 \cdot \overline{b_{N-1}} + \frac{1}{2} \cdot \overline{b_{N-2}} + \ldots + \frac{1}{2^{N-1}} \cdot \overline{b_0} \right] \quad (30.9)$$

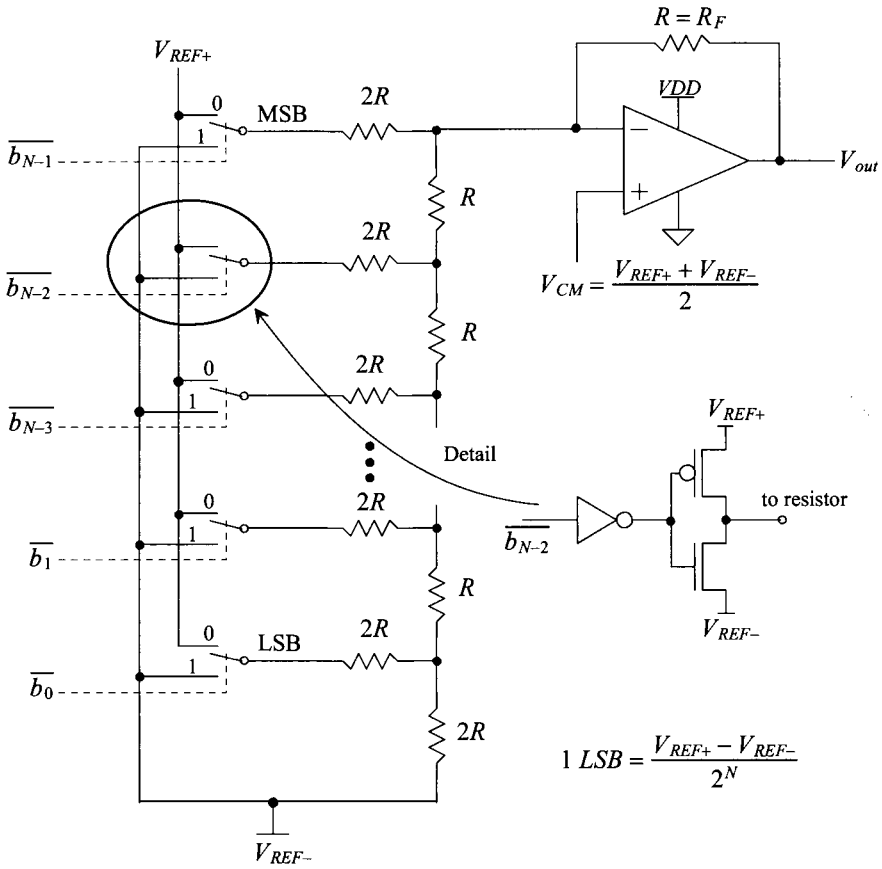


Figure 30.3 Wide-swing current-mode R-2R DAC.

noting the inversion used in the control logic seen in Fig. 30.3. The output voltage of the DAC is then given, assuming $R = R_F$, by

$$V_{out} = V_{REF-} + \frac{V_{REF+} - V_{REF-}}{2} + I_F \cdot R \qquad (30.10)$$

or

$$V_{out} = V_{REF-} + (V_{REF+} - V_{REF-}) \cdot \left[1 - \left(\frac{1}{2} \cdot \overline{b_{N-1}} + \frac{1}{4} \cdot \overline{b_{N-2}} + \ldots + \frac{1}{2^N} \cdot \overline{b_0} \right) \right] \qquad (30.11)$$

From this equation, we see that as the digital input code is sequenced through 000... to 111... the output of the DAC changes in steps of $(V_{REF+} - V_{REF-})/2^N$ (= 1 LSB) from V_{REF+} (when the input code is 111...) to V_{REF-} + 1 LSB (when the input code is 000...). Setting V_{REF-} to ground and V_{REF+} to *VDD* allows the DAC output to swing from rail to rail.

In practice, since any rail-to-rail output op-amp has high nonlinearity close to its power-supply rails, a slightly "shrunk" output range from power rails is often desired. For example, we can set $V_{REF+} = 0.9 \cdot VDD$ and $V_{REF-} = 0.1 \cdot VDD$. The output will change between 10 and 90% of VDD centered at $VDD/2$. Another way to shrink the output range is to make the feedback resistance R_F smaller than R (as seen in Eq. [30.10]) by either trimming or programming the value of the feedback resistor R_F.

The matching between the resistors of the R-$2R$ ladder is one of the most important and limiting factors that determine the linearity (e.g., DNL and INL) of the entire DAC. It is helpful, when designing any type of resistor string DAC, if we can estimate the resistor matching requirements based on a desired resolution.

DNL Analysis

It was shown in Ch. 29 that for a binary-weighted DAC the worst case DNL condition tends to occur at midscale when the code transitions from 01...11 to 10...00. Let's assume in a worst-case scenario the $2R$ resistance of the MSB input in Fig. 30.3 has a maximum positive mismatch of ΔR, and all other resistors have a maximum negative mismatch of $-\Delta R$. *In this case, the current provided by the MSB has to match the sum of currents provided by all other lower input bits plus one LSB.* Again using the superposition principle we can verify that the step error of the current flowing through the feedback resistor R_F, caused by the resistor mismatch at the midscale transition, is approximately equal to

$$\Delta I = \frac{V_{REF+} - V_{REF-}}{2(R - \Delta R)} \cdot \left[1 - \frac{1}{2^{N-1}}\right] - \frac{V_{REF+} - V_{REF-}}{2(R + \Delta R)} \qquad (30.12)$$

Assuming $R_F = R$, the final output step error (DNL) is approximately

$$DNL = \Delta I \cdot R \approx (V_{REF+} - V_{REF-}) \cdot \left[\frac{\Delta R}{R} - \frac{1}{2^N}\right] \qquad (30.13)$$

For the DNL to be within 1 LSB (1 LSB equals to $[V_{REF+} - V_{REF-}]/2^N$) the matching required of the resistors is

$$\text{Resistor mismatch} = \left|\frac{\Delta R}{R}\right| \le \frac{1}{2^{N-1}} \qquad (30.14)$$

For a 10-bit data converter to have a DNL of less than 1 LSB requires the MSB resistor to match within 0.2% (= $\Delta R/R$) of the lower resistors (which were assumed to have the same value, i.e., the maximum mismatch from the MSB resistor) in the R-$2R$ string. Equation (30.14) results in a pessimistic estimate for the matching required of the resistors because the variation in resistance along the string does not vary abruptly at the MSB resistor but rather, in most cases, varies linearly from LSB to MSB. As we'll see in the experimental results discussed in the next section, the matching requirements result in a practical limit of 10-bits for an R-$2R$-based converter with no special layout or circuit techniques (for example, averaging variations using multiple resistor strings or using segmentation).

INL Analysis

Since any change of the $2R$ resistance in the MSB has the largest influence on the ladder output current among that of all the branch resistors ($2R$), the worst case INL tends to occur when the input code is 01...11. (The gain error is nulled from the INL calculation

here, and therefore there is no INL error, but a gain error instead, if all the resistors have a maximum mismatch.) Assuming that the $2R$ resistance of the MSB has a maximum positive mismatch of $\Delta R/R$, the error in the current flowing through R_F from its ideal value caused by the resistance mismatch is

$$\Delta I = \frac{V_{REF+} - V_{REF-}}{2(R+\Delta R)} - \frac{V_{REF+} - V_{REF-}}{2R} \approx -\frac{V_{REF+} - V_{REF-}}{2} \cdot \frac{\Delta R}{(R+\Delta R)\cdot R} \quad (30.15)$$

The worst-case INL tends to occur, assuming $R_F = R$, when

$$INL = -\Delta I \cdot R \approx \frac{V_{REF+} - V_{REF-}}{2} \cdot \frac{\Delta R}{R+\Delta R} \quad (30.16)$$

For the INL to be within 1 LSB, this also approximately yields

$$\text{Resistor mismatch} = \left|\frac{\Delta R}{R}\right| \le \frac{1}{2^{N-1}} \quad (30.17)$$

Again, as was mentioned in the DNL analysis, this is a pessimistic estimate if the sheet resistance varies linearly with distance. Equations (30.14) and (30.17) indicate that a resistance matching to within $1/2^N$ is required for less than ½ LSB of DNL and INL for the DAC scheme in Fig. 30.3. Layout of R-$2R$ resistors was discussed in Ch. 5.

Switches

The switches (MOSFETs) used in the R-$2R$ DAC should have an effective switching resistance much less than the resistors used in the R-$2R$ ladder. The inherent switching time of the switches is extremely fast (speeds comparable to logic gate delays). Since the switches are in series with the branch resistances of the R-$2R$ ladder, the R-$2R$ relationship is broken if the switch resistance is not negligible, and this affects both the INL and the DNL. Also note that we can try to compensate for the switch-effective resistance by making the length of the $2R$ resistor slightly shorter than the length of the R resistor. However, if not careful, this may lead to problems over the process corners and temperature.

Experimental Results

The wide-swing, current-mode R-$2R$ DAC, based on the scheme in Fig. 30.3, was fabricated in a 210 nm, VDD = 1.8 V, CMOS process with resolutions of 8, 10, and 12 bits. The cell dimensions of the 12-bit DAC are 150 μm by 300 μm. The goal of the experimental results was to verify that the topology of Fig. 30.3 would indeed perform as predicted by Eqs. (30.14) and (30.17) and to generate a low-power, small-area DAC cell for general-purpose, mixed-signal circuit designs. Unsilicided n+ poly was used for the R-$2R$ resistances as discussed earlier (see Table 4.1). The mismatch indicated in Table 4.1 for an unsilicided n+ poly resistor is 0.005 (= $\Delta R/R$). Using Eqs. (30.14) and (30.17), we would estimate that our resolution is limited to 8.6 bits if we want both INL and DNL less than 1 LSB. The results in Table 30.1, however, show that the resolution is better than estimated. This may be because of our pessimistic assumption of how the resistor values change with position as discussed in the derivation of these equations. The nominal resistor value used in these experimental DACs is 10k. To enhance the resistance matching, dummy resistors are implemented at both ends of the R-$2R$ ladder (see Fig. 5.28). The output range of the DAC is programmable by choosing the value of the feedback resistance or by the setting of the reference voltages V_{REF+} and V_{REF-}.

Table 30.1 Summary of experimental results.

	8-bit	10-bit	12-bit
DNL (LSB)	0.150	0.450	2.000
INL (LSB)	0.200	1.000	3.000
Settling time	200 ns		
Power	3.88 mW (driving a 1k load)		
Area (mm^2)	0.045		
$f_{clk,max}$	4 MHz		
Output swing	$0 < V_{out} < VDD\ (= 1.8\ V)$		

The measured INL and DNL profiles of the three DACs with resolutions of 8, 10, and 12 bits are shown in Fig. 30.4. The outputs of the DACs are configured to swing to both rails ($V_{REF+} = VDD$ and $V_{REF-} = 0$). The first several points, adjacent to the two rails, are not shown in Fig. 30.4 due to the high nonlinearity of the op-amp in those regions. Major performance results are maximum DNLs of 0.15 LSB, 0.45 LSB, and 2 LSB for

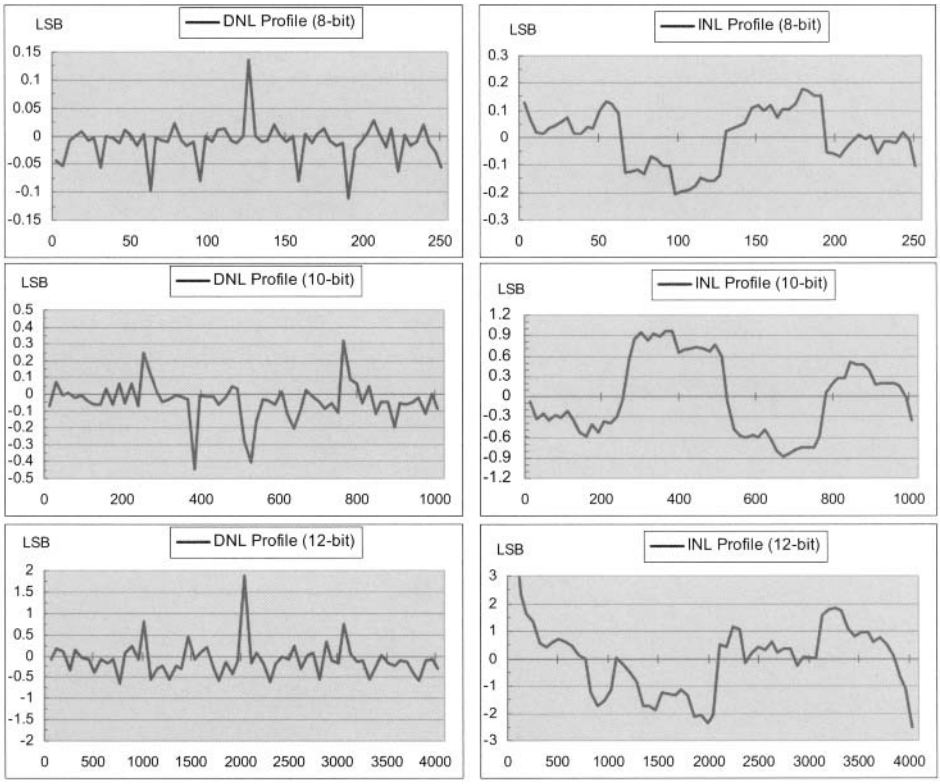


Figure 30.4 Experimental results for the wide-swing DAC of Fig. 30.3.

8-bit, 10-bit, and 12-bit resolutions, respectively, with no special circuit techniques (laid out as shown in Fig. 5.28) or trimming (adjustments). The corresponding maximum DAC INLs are 0.2 LSB, 1 LSB, and 3 LSB, respectively. Notice that the LSB of the 8-, 10-, and 12-bit DACs are 7.03 mV, 1.75 mV, and 439 μV, respectively. The DNL/INL can be written in terms of a voltage as 1.05 mV/1.4 mV for the 8-bit DAC, 0.788 mV/1.75 mV for the 10-bit DAC, and 0.878 mV/1.31 mV for the 12-bit DAC. The measurements were taken while the DAC was driving a 1k load. The power dissipated by the DAC, with 1.8 V output, while driving a 1k resistor is 3.88 mW. The unloaded power dissipation of the DAC is approximately 500 μW. The DACs were designed using op-amps with simulated unity gain frequencies of 10 MHz. The measured DAC settling time was approximately 200 ns.

Improving DNL (Segmentation)

After reviewing the DNL plots in Fig. 30.4, we see that the worst-case DNL occurs when the input code transitions from 01111... to 10000... (midscale) where the current in the top 2R should be 1 LSB (equivalent in current) greater than the sum of all of the currents contributed by the lower resistors. As an example, consider the 12-bit *R*-2*R* ladder in Fig. 30.5 where we have used 1 μA to indicate an LSB of current contribution to the feedback

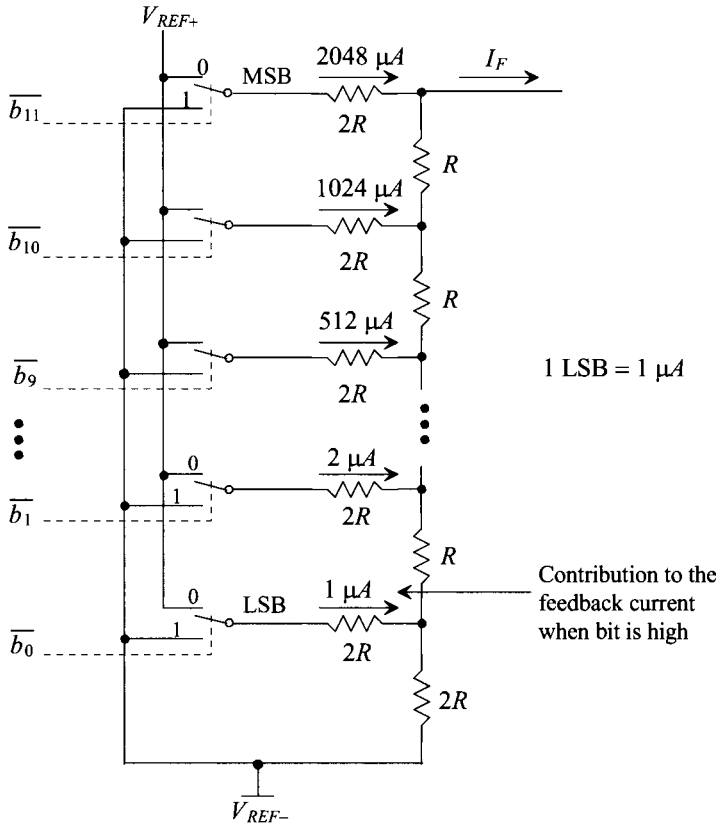


Figure 30.5 Showing how currents sum into the feedback current.

path. When the input digital code is 0111 1111 1111, the feedback current is 2047 μA. When the code changes to 1000 0000 0000, the feedback current becomes (ideally) 2048 μA. If a 1/2 LSB error (0.5 μA) is the maximum error allowable, then the accuracy required of the currents when transitioning is 0.5/2048 or 0.0244%. If we use fewer bits, say eight, then the accuracy required when transitioning from 255 μA to 256 μA is 0.5/256 or 0.2%.

Let's consider segmenting the upper four bits in Fig. 30.5 so that the four bits control 16 segments each contributing 256-μA to the feedback current. This segmentation makes attaining good DNL with less accurate components possible. A segmented wide-swing DAC is shown in Fig. 30.6. In this figure we've taken the upper four bits and segmented their current contributions to the feedback resistor. If we use the numbers from Fig. 30.5, then when the code 0000 1111 1111 (255 μA) transitions to 0001 0000 0000 (256 μA) the 1 output of the decoder goes high and the bottom resistor connected to the

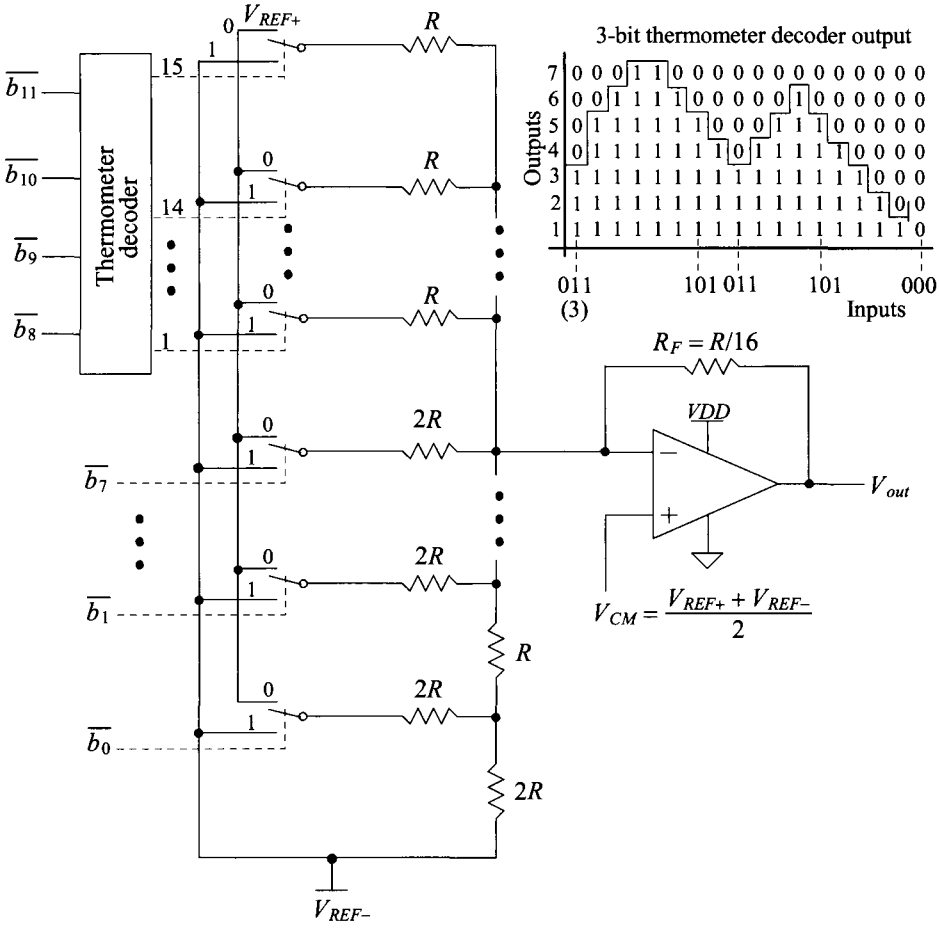
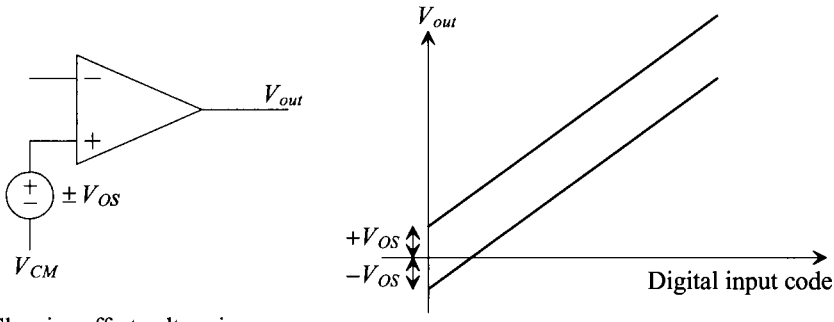


Figure 30.6 Segmentation in a wide-swing R-2R DAC.

output of the decoder contributes 256 μA to the feedback path. When the code changes from 0001 1111 1111 (511 μA) to 0010 0000 0000 (512 μA), both lower outputs (1 and 2) of the thermometer decoder are high. Since the 1 decoder output continues to contribute to the output current, the step height is set by the difference between the 2 decoder outputs and the contributions from the lower eight bits. This makes the accuracy requirements for 1/2 LSB DNL in a 12-bit converter set by 8-bit matching. Note that while segmentation reduces DNL error, it does nothing for INL. Segmentation can also be used to reduce the glitch area associated with the changing DAC output.

Trimming DAC Offset

Figure 30.7 shows how the op-amp's offset voltage shifts the DAC's output. It may be desirable in some situations to trim or remove this offset. The offset may be the result of an inherent systematic offset in the op-amp or the result of random variations in the characteristics of the MOSFETs used in the op-amp. An offset may also result because of the voltage dependence of the resistors used to generate the common-mode voltage, V_{CM}.



(a) Showing offset voltage in an op-amp.

(b) DAC transfer curves showing offset.

Figure 30.7 Showing how an op-amp offset affects the DACs transfer curves.

Figure 30.8 shows one possible method to generate a common-mode voltage that is adjustable with a digital code. Here we are assuming that V_{CM} is ideally 0.5 V. We should recognize the *R*-2*R* ladder from Fig. 30.2. The output voltage of this ladder, as seen in Eq. (30.6), is an analog voltage related to the digital input word (assuming the voltage divider made up of R_{big} and the two *R* resistors connected to the output in Fig. 30.8 doesn't load the circuit). Figure 30.9 shows the output of this circuit for all possible digital input words when *R* is 10k and R_{big} is 100k. This figure also shows that the adjustability of the output is approximately 1 mV. To decrease this value, we can either increase R_{big} (resulting in a decrease in the output swing) or increase the number of bits in the *R*-2*R* DAC. The value, *R*, of the resistors on the output can be decreased, but this can result in an increase in power dissipation.

Note that the accuracy required of the 5-bit DAC can be very loose. N-well resistors can be used to implement the offset trimming circuit to reduce area and power. The main concerns are considering the possibility of substrate noise injection and making sure that the same resistive material is used for the entire circuit. We wouldn't want the temperature behavior of an n-well resistor used in a circuit with a poly resistor because

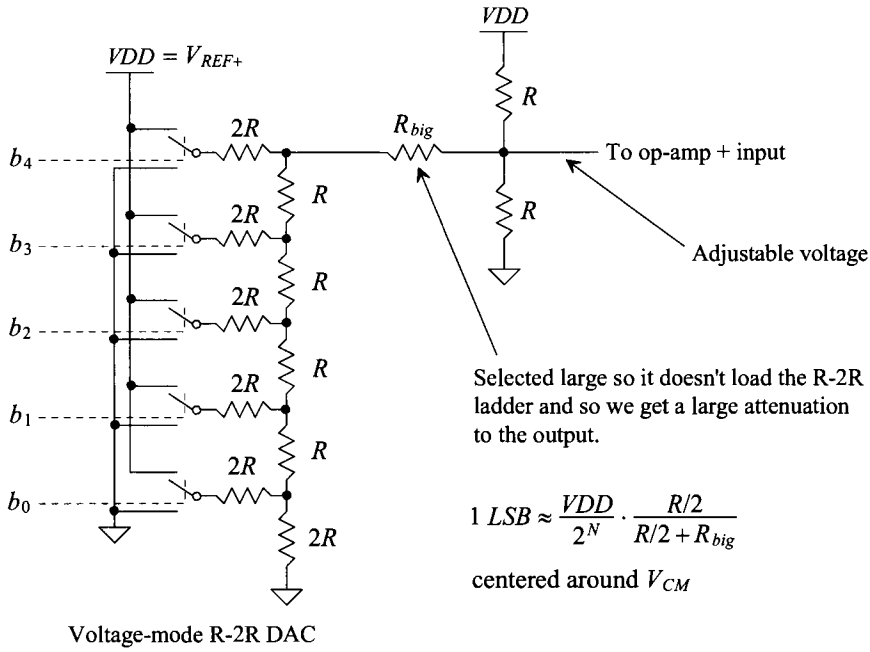


Figure 30.8 Trimming circuit for DAC offset.

the temperature dependencies are different (the offset trimming would only be effective at the temperature it was performed). Finally, note that in a practical circuit it is a good idea to add capacitors from the output of the circuit to both *VDD* and ground to ensure that the + op-amp input is connected to a good AC ground.

Trimming or calibrating out the offset can be performed at a time prior to packaging the chip, or it can be performed with some autocalibration sequence after the chip has been fabricated where the output of the DAC is compared to a known voltage reference. The concern, as with any calibration, is to adjust only one known error at a time (known as orthogonal tuning in filter design). For example, the DAC may not have any offset but may have an INL error for a given input code, Fig. 30.10a. If we were only

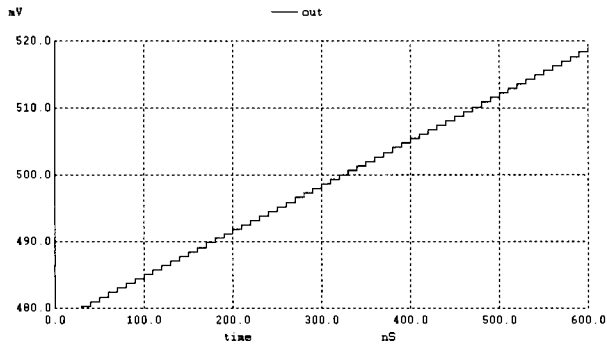
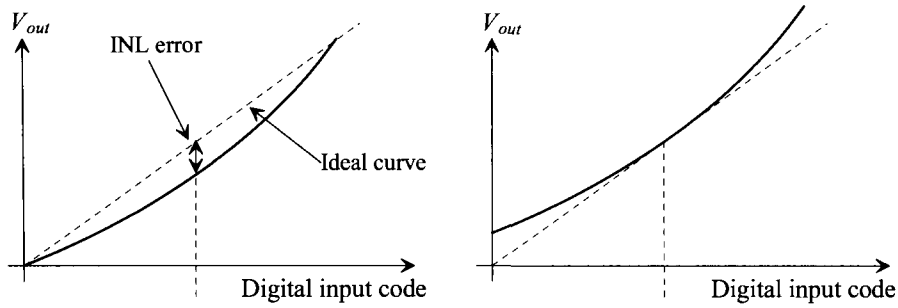


Figure 30.9 Output of the circuit in Fig. 30.8 for all possible digital codes.



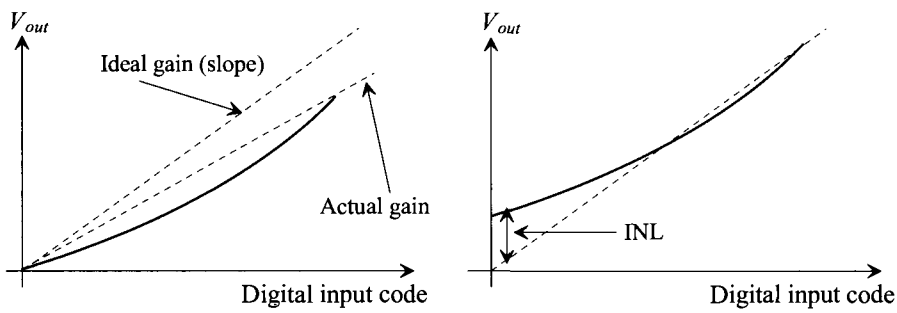
(a) DAC transfer curves before calibration. (b) DAC transfer curves after offset calibration

Figure 30.10 Showing how INL can be seen as an offset error.

to look at this one input code, say 10000... (V_{CM} in binary offset), we wouldn't know if the error is an INL error or an offset error. After the offset is calibrated out, Fig. 30.10b, we would then perform an INL calibration to pull the end-points of the transfer curve back to the ideal straight line transfer curve.

Trimming DAC Gain

We assumed in Fig. 30.10 that the gain of the DAC was one, in other words, there wasn't any gain error in the DAC's transfer function. If there is a gain error, the offset calibration can lead to poorer INL. Consider Fig. 30.11a showing gain and INL errors without any offset. Performing an offset calibration, Fig. 30.11b, can result in significant INL error. We can avoid this situation by calibrating out the gain error by trimming the op-amp's feedback resistor prior to offset calibration. A reference voltage close to the ends of the transfer curve is used while adjusting the gain of the op-amp used in the DAC. If V_{REF+} is less than *VDD* (to avoid op-amp saturation as its output approaches the supply rails), then it can be compared directly to the output of the DAC (keeping in mind the maximum output of the DAC may be $V_{REF+} - 1\ LSB$). Having gone through all of this discussion, it



(a) DAC transfer curves with gain error.

(b) DAC transfer curves after offset calibration with gain error.

Figure 30.11 Showing gain error and how it can cause problems in an offset calibration.

still would be nicer if we could simply perform two calibrations, offset calibration and INL calibration, effectively using the INL calibration to remove the gain error. The drawback of this two calibration method is the requirement that an INL calibration circuit be capable of removing very large INL errors.

Improving INL by Calibration

We can calibrate out errors in our wide-swing DAC in two basic ways as seen in Fig. 30.12. The method shown in part (a) adds or subtracts a current from the feedback path to adjust the DAC output to the correct value. In part (b) the noninverting input of the op-amp is varied to force the DAC output to the correct value. The offset calibration described earlier uses the method shown in part (b). Note that the resistance looking from the inverting op-amp terminal back through the ladder to AC ground is simply R, so using the method in part (b) results in a noninverting op-amp configuration with a gain of two. (A variation of 1 mV on the + op-amp terminal causes an output variation of 2 mV.) Because we already have a circuit, Fig. 30.8, to make adjustments to the DAC output and the topology of part (b) doesn't provide any DC load to the calibrating voltage source and provides the least interaction with the main R-$2R$ ladder, we will use this topology to illustrate how we can calibrate out INL errors.

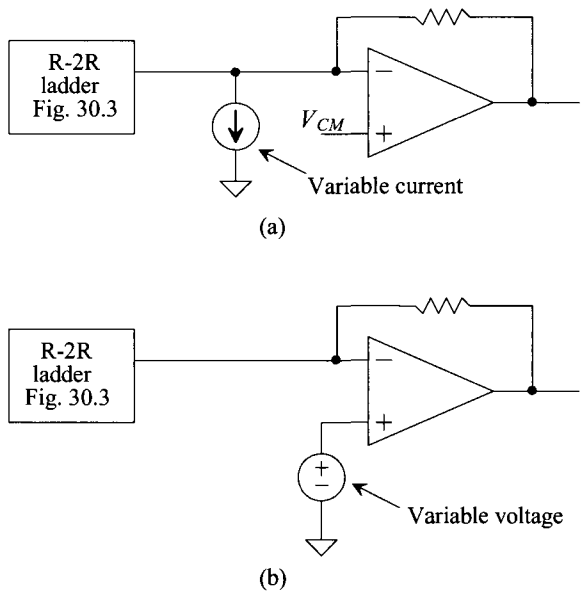


Figure 30.12 Trimming the output of the DAC using (a) current and (b) voltage.

Consider the calibration circuit shown in Fig. 30.13. In this figure the five most significant bits of a 12-bit DAC, that is, b_{11}, b_{10}, b_9, b_8, and b_7 are applied to the 12-bit DAC and to the address input of a 32-to-1 MUX with 5-bit input and output words. The MUX drives the R-$2R$ circuit of Fig. 30.8. The 5-bit register feeding each MUX input is used to store the calibration values. Again the calibration can be performed after the DAC

is manufactured or during its use by employing a self-calibration sequence. In this scheme the DC offset calibration shown in Fig. 30.10 is simply one case of the 32 calibrations performed. (The top five bits of the input word are 10000 for the DC calibration.) This technique can be used to more precisely set the linearity of the DAC, perhaps up to 16-bits. Note that segmentation must still be employed to keep the DNL small. Also note that adding a large capacitance, C, to the noninverting input of the op-amp can result in a long time constant $\left(\approx \frac{R}{2} \cdot C\right)$, slowing the settling of the circuit and affecting the high-frequency *SNDR*.

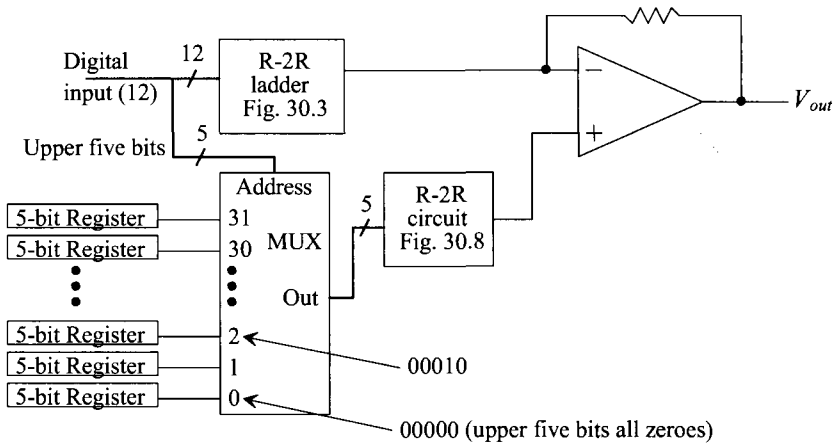


Figure 30.13 Calibration scheme for 12-bit DAC.

30.1.4 Topologies Without an Op-Amp

We discuss the requirements of op-amps used in data converters in detail in the next section. In this section we present an overview of topologies using voltage and current mode DACs based on both *R-2R* and *W-2W* topologies. The benefit of using a DAC with an op-amp is that an arbitrary load impedance can be connected to the DAC's output. Without the op-amp, the load impedance must either be known, capacitive, or very large, since it will load the DAC's output and affect INL, DNL, and, ultimately, the *SNR*. The benefits of not using an op-amp are faster-speed (with light loads) and guaranteed stability.

The Voltage-Mode DAC

The simplest voltage-mode DAC is the *R-2R* string shown in Fig. 30.14. We should recognize this circuit from both Figs. 30.2 and 30.8. For the moment we assume the load is purely capacitive so that errors resulting from sourcing a DC current are not present in the DAC. Here, through several examples, we discuss settling time, resistor voltage coefficient, matching, and the effects of a DC load. While the fact that this DAC can't drive an arbitrary load resistance may be an issue in a particular design, the simplicity and small size of the topology make it very useful in many situations.

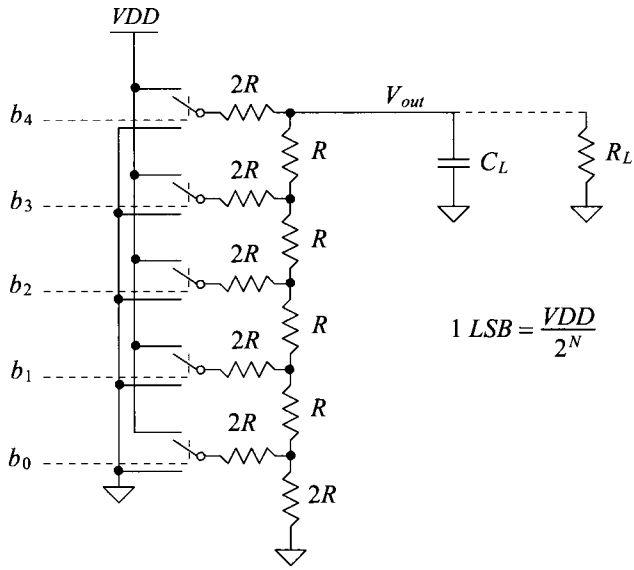


Figure 30.14 Voltage-mode (5-bit) DAC without an op-amp.

Example 30.1

Suppose a 10-bit, voltage-mode DAC with the topology seen in Fig. 30.14 is implemented where R = 10k and C_L = 10 pF. Estimate the maximum clocking frequency that can be used to clock the register supplying the input words to the DAC. Verify your answer using SPICE.

For complete settling the DAC must be 10-bit accurate to within 0.5 LSBs over its full-scale range

$$\text{Accuracy} = \frac{0.5\ LSB}{\text{Full scale range } (VDD)} = \frac{VDD/2^{N+1}}{VDD} = \frac{1}{2^{11}} = 0.04883\%$$

The time constant associated with the DAC and capacitive load is

$$RC_L = 10k \cdot 10p = 100\,ns$$

This time constant can be related to the final ideal output voltage, $V_{outfinal}$, and the actual output voltage, V_{out}, using

$$V_{out} = V_{outfinal}(1 - e^{-t/RC_L})$$

or, relating this to the required accuracy,

$$\frac{1}{2^{N+1}} = 1 - \frac{V_{out}}{V_{outfinal}} = e^{-t_{settling}/RC_L}$$

The required settling time is then

$$t_{settling} = RC_L \cdot \ln 2^{N+1} \qquad (30.18)$$

Using the numbers from this example results in $t_{settling}$ = 762 ns. The SPICE simulation results are shown in Fig. 30.15. The maximum clock frequency is then estimated as

$$f_{clk,\max} = \frac{1}{t_{settling}} = \frac{1}{RC_L \cdot \ln 2^{N+1}} \tag{30.19}$$

For this example, $f_{clk,\max}$= 1.3 MHz. Note that the fundamental way to decrease the settling time is to decrease the resistance in the R-$2R$ ladder (assuming we have no control over the load capacitance). The practical problem then becomes implementing the switches (MOSFETs) with a resistance small compared to R. ■

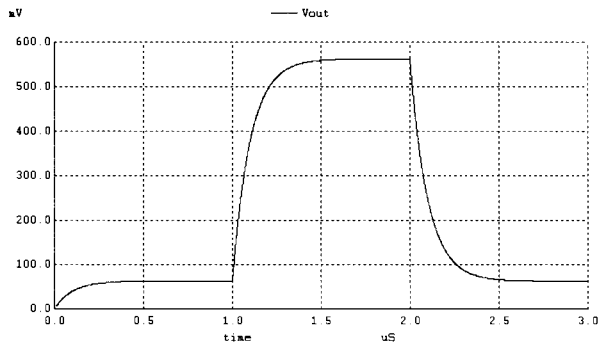


Figure 30.15 Example output for the 10-bit DAC in Ex. 30.1 showing settling time limitations.

Example 30.2

Suppose that the $2R$ MSB resistor in the DAC described in Ex. 30.1 experiences a 0.5% mismatch. Estimate the resulting DAC's INL and DNL. Use SPICE to verify your answer.

The 0.5% mismatch ($\Delta R/R$ or 1 σ [standard deviation]) is the mismatch specified for the unsilicided n+ polysilicon resistors in Table 4.1. Again it is desirable to use poly resistors because they sit above the substrate, on the field oxide, and are more immune to substrate noise. Note that the voltage coefficient can (will) also cause nonlinearities. However, instead of the worst-case situation of an abrupt mismatch between the lower resistors and the MSB $2R$ resistor, as used in this example, a first-order voltage coefficient error will cause a linear variation of the resistor values from the LSB resistor up to the MSB resistor and so the effects of the voltage coefficient, for reasonably small values, are generally not significant compared to the random mismatch effects.

Rewriting Eqs. (30.14) and (30.15) to estimate the maximum number of bits possible with 1 LSB INL or DNL results in

$$N = 1 - 3.3 \cdot \log\left(\frac{\Delta R}{R}\right) \tag{30.20}$$

Using this equation with $\Delta R/R = 0.005$ results, again, in N = 8.6 bits. For a 10-bit DAC, we would estimate both the INL and DNL as 2.4 bits.

To verify these results using SPICE, let's input a code of 01 1111 1111 (ideally 499 mV) and then step the input code to 10 0000 0000 (ideally, 500 mV). With the MSB $2R$ resistor changed to 20.1k (a 0.5% mismatch from its ideal 20k value) the simulation results are shown in Fig. 30.16. With this mismatch the output of the DAC is 500.3 mV when the input is 01 1111 1111. The INL with this input code is 1.25 LSBs (roughly 1.5 mV). The INL when the input digital code is 10 0000 0000 is −1.25 LSBs. The DNL at this worst-case point is −2.5 LSBs. Note that the DAC is nonmonotonic (DNL < −1 LSB). An increase in the digital input code results in a decrease in the output voltage. Nonmonotonic DACs can result in circuits that don't function properly (an example being a successive approximation ADC). A DNL of −1 LSB would indicate the output voltage of the DAC doesn't change when the input code changes.

To improve the DNL, the upper bits of the DAC must be segmented as seen in Fig. 30.6. Improving the INL relies on calibrating out the mismatch errors (see Figs. 30.12 and 30.13). Also note, again, that mismatch can be improved by layout techniques (e.g., common-centroid) and by averaging the outputs of multiple resistor strings. ■

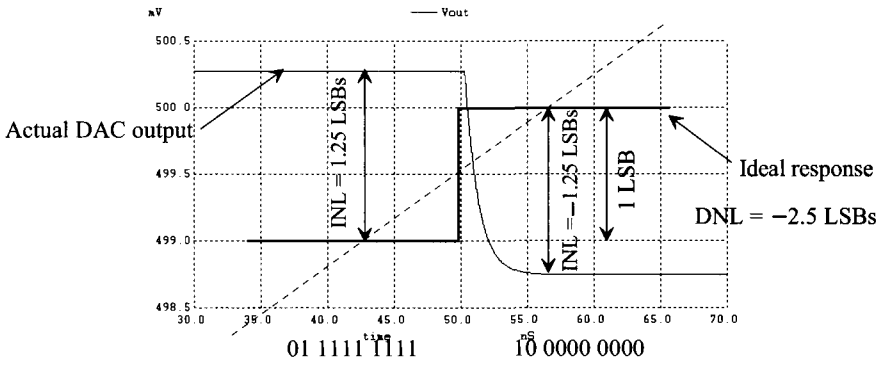


Figure 30.16 Output if MSB resistor in Fig. 30.14 experiences a 0.5% mismatch.

We can characterize the effects of a DC load resistance, R_L, as seen in Fig. 30.14, by noticing that R_L forms a divider with the R-$2R$ ladder. The LSB with a load can be written as

$$1 \text{ LSB} = \frac{VDD}{2^N} \cdot \frac{R_L}{R + R_L} \tag{30.21}$$

Notice that if $R_L \to \infty$, this equation reduces to the LSB value given in Fig. 30.14. The time constant associated with driving an output capacitance can now be written as

$$\tau = R \| R_L \cdot C_L \tag{30.22}$$

Two Important Notes Concerning Glitches

Note that we have assumed that the RC delay through the resistors used in the R-2R ladder is negligible. This may not be the case in many practical situations (especially if diffused or implanted resistors are used), resulting in a DAC output glitch. Also, we have

been simulating with perfectly aligned digital signals, that is, signals that change at the exact same moment. When the digital signals are slightly misaligned, a significant glitch can occur in the DAC's output. *This means that the inputs to the DAC should be provided by the same digital hold register.* Using segmentation with the required thermometer decoder can result in the digital signals driving the *R*-2*R* ladder seeing differing delays. Care must be exercised when designing the DAC input clocking circuit (e.g., add small dummy delays).

Example 30.3

Repeat Ex. 30.2 if a 200 ps skew is experienced by the lower nine bits in the digital inputs with relation to the MSB.

The simulation results are shown in Fig. 30.17. When comparing this result to Fig. 30.16, the magnitude of the glitch in relation to the much smaller final step in the output voltage should be obvious. The small 200 ps skew in the digital inputs causes a code of 00 0000 0000 to be applied to the DAC for 200 ps. For this very short period of time the output begins to discharge from 500 mV down to ground. Note in this figure and in Fig. 30.16 the load capacitance was reduced to 0.1 pF to decrease the settling time. ■

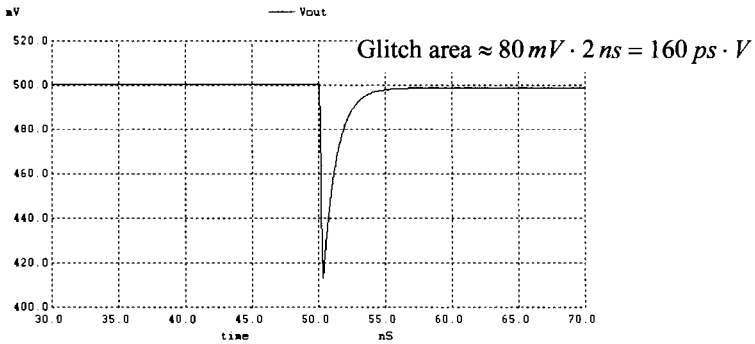


Figure 30.17 Showing glitch if the lower 9-bits are skewed by 200 ps in Ex. 30.2.

The Current-Mode (Current Steering) DAC

Figure 30.18 shows the two basic cells used in the implementation of a current-mode DAC. In this section we focus on the use of the current-source based cell. The advantage of the current-source cell is the fact that the value of the current can be adjusted, via the bias voltage, to compensate for process variations, while the value of the resistor, in the resistor-based cell is fixed. The advantages of the resistor-based cell are wider output swing (the MOSFET current source must remain in the saturation region), better voltage coefficient (no channel length modulation or other finite MOSFET output resistance effects), and better substrate noise immunity (assuming the resistors are integrated on the top of the field oxide and not down in the substrate with the MOSFETs). Notice, in this figure, that we've implemented the cells with two complementary outputs. Having complementary outputs is useful, for example, in a DAC used with a video monitor (driving two complementary 75 Ω loads). The load resistors are labeled the left load resistor, R_{LL}, and the right load resistor, R_{RL}.

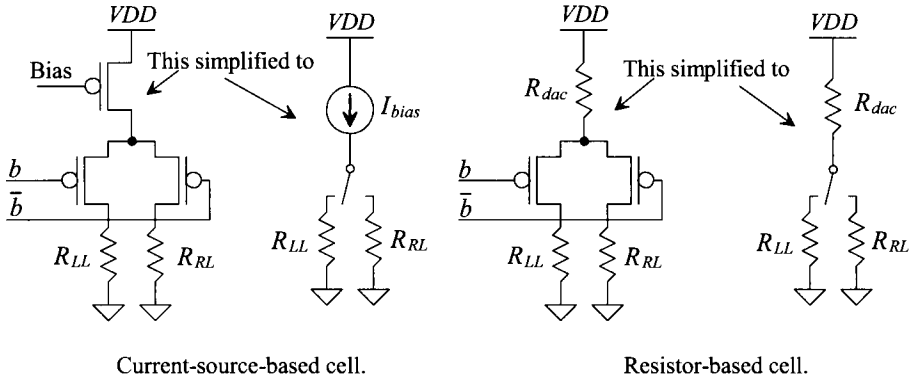


Figure 30.18 Basic cell used in a current-mode DAC.

Figure 30.19 shows the block diagram implementation of a current-mode DAC. The output voltages depend on both I_{REF} and the load resistors. The current sources are implemented using the PMOS cell in Fig. 30.18 and a PMOS *W*-2*W* mirror (see Fig. 30.20 for the NMOS version of the circuit) to improve layout area. The combination of binary-weighted current sources must be used together with the required segmentation of the upper bits to reduce DNL. Although not drawn so in Fig. 30.18, the current sources can be cascoded to increase their output resistance (decrease their voltage coefficient). Again, the switches connected to the loads should be controlled by signals from the same register to avoid significant glitches in the outputs.

While the matching requirements of current-mode (current steering) DACs were discussed in Ch. 29, we should comment on ways to improve matching. The layout of the *W*-2*W* ladder should follow the basic techniques discussed in Ch. 20, i.e., devices oriented the same way, use of dummy poly and source/drain implants, attempt to keep the source-drain voltages constant, and use of long *L* devices. The layout of the *W*-2*W* mirror should look similar to the layout of the *R*-2*R* string in Fig. 5.28, but it can also employ two or more *W*-2*W* segments to average variations. The upper segmented bits can be laid out adjacent to the *W*-2*W* and can also benefit from averaging the outputs of several DAC layouts. In some cases the number of bits used in the *W*-2*W* ladder equals the number of bits used for the upper segments. For example, a 10-bit DAC would use a 5-bit *W*-2*W* ladder (whose MSB is $I_{REF}/2$) and 31 segments with values of I_{REF}. To improve INL separate layouts can be connected together to average the random mismatch effects and increase the linearity of the DAC. Of course, the current output levels increase for the same reference biasing levels.

Finally, as mentioned earlier, we can use a *W*-2*W* current mirror to generate the binary weighted currents used in a current-steering DAC, Fig. 30.20. What we are going to do, with the binary weighted current mirror, is utilize the fact that MOSFETs in series and parallel can be combined, as seen in this figure, into single MOSFETs. Two MOSFETs placed in series with equal widths are equivalent to a single MOSFET with the sum of the two MOSFET's lengths. Two MOSFETs in parallel with the same length sum to form a single MOSFET with the sum of the two MOSFET's widths. The benefit of this topology is the removal of the effects of oxide encroachment and lateral diffusion.

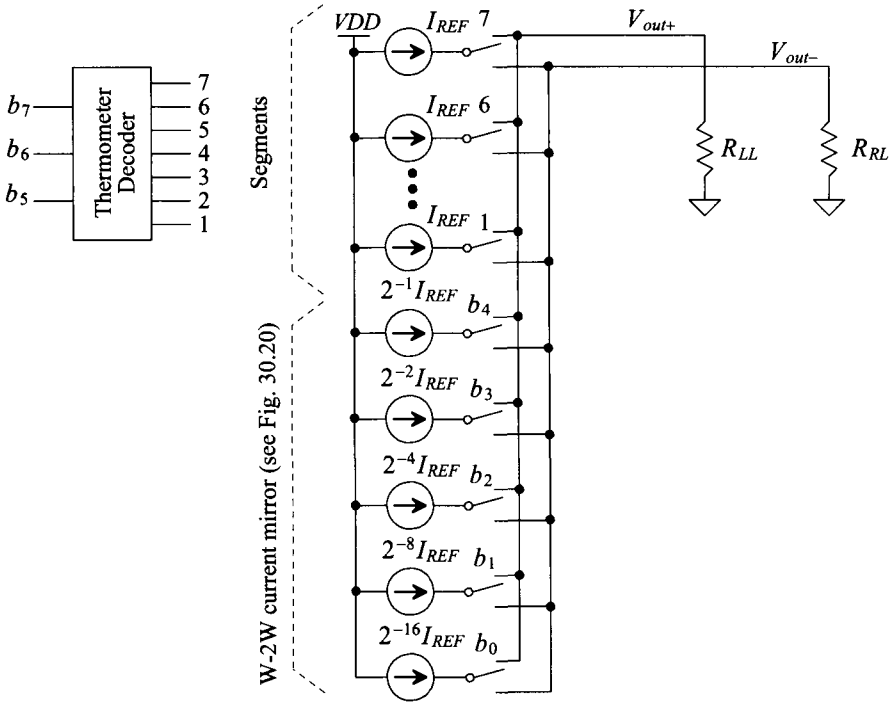


Figure 30.19 Implementation of a current-mode DAC.

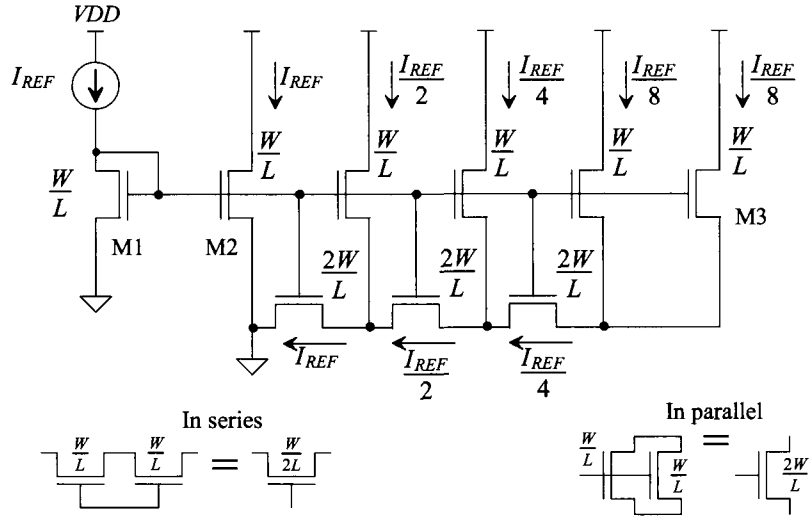


Figure 30.20 W-2W current mirror.

30.2 Op-Amps in Data Converters

The open-loop magnitude and phase responses of a typical op-amp are shown in Fig. 30.21. In this section we discuss the gain and bandwidth requirements of op-amps used in either a DAC or an ADC. *We assume* that the op-amp is designed to have a phase margin of 90 degrees under full load conditions and over process variations. (We should point out that this assumption is easily met using an OTA that is compensated by a load capacitance as discussed in Ch. 24.) It's important to understand why having a 90-degree phase margin is important, namely, to avoid a second-order step response with the associated ringing. If the phase margin is 90 degrees, we get an RC-like settling response shape as seen in Fig. 30.15. Figure 26.60 shows the step response of a mixed-signal op-amp. The phase margin of this op-amp was less than 90 degrees and thus the step response shows ringing. Decreasing the phase margin increases the peak amplitude of the ringing and can lengthen the settling time (the time it takes the op-amp's output to settle to within 1/2 LSB of the ideal final value).

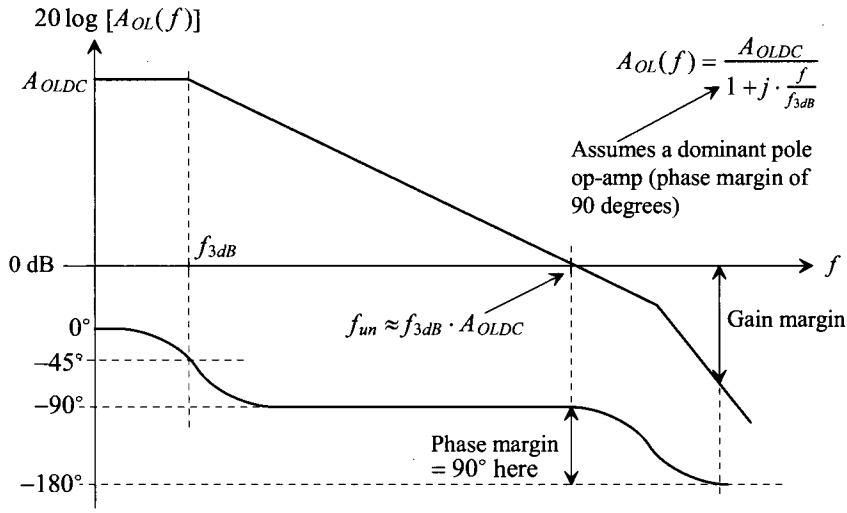


Figure 30.21 Magnitude and phase responses of an op-amp.

Gain Bandwidth Product of the Noninverting Op-Amp Topology

Figure 30.22 shows the basic topology of a noninverting op-amp amplifier. The voltage on the inverting op-amp input can be written as

$$v_- = v_{out} \cdot \overbrace{\frac{R_1}{R_1 + R_2}}^{\beta} \tag{30.23}$$

where β is the feedback factor for this *series-shunt* feedback amplifier (see Ch. 31, the ideal closed-loop gain, A_{CL}, is 1/β or $1 + R_2/R_1$). The output of the amplifier is

$$v_{out} = (v_{in} - v_-) \cdot A_{OL}(f) \tag{30.24}$$

Solving these equations for the closed-loop bandwidth of the amplifier, $f_{CL,3dB}$, gives

$$f_{CL,3dB} \approx \beta \cdot A_{OLDC} \cdot f_{3dB} = \beta \cdot f_{un} \tag{30.25}$$

The gain bandwidth product of the noninverting amplifier is then

$$\text{Gain} \cdot \text{bandwidth} = f_{un} \tag{30.26}$$

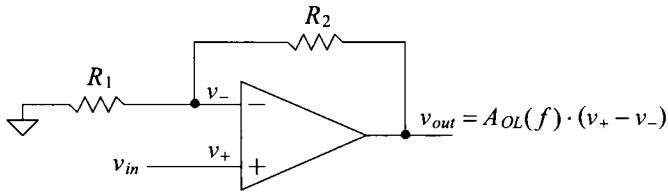


Figure 30.22 Noninverting op-amp topology.

Gain Bandwidth Product of the Inverting Op-Amp Topology

Figure 30.23 shows the schematic diagram of an inverting op-amp topology. Summing the currents at the inverting input node gives

$$\frac{v_{in} - v_-}{R_1} = \frac{v_- - v_{out}}{R_2} \tag{30.27}$$

The output of the amplifier is related to the op-amp's input terminals using

$$v_{out} = (-v_-) \cdot A_{OL}(f) \tag{30.28}$$

Solving these two equations for the closed-loop bandwidth once again results in Eq. (30.25) with β defined as indicated in Eq. (30.23). This can be confusing because the feedback factor, β, for the inverting amplifier is not the same as for the noninverting amplifier. The inverting op-amp is an example of a *shunt-shunt* amplifier (current input and voltage output). The feedback factor for this amplifier is $-1/R_2$ (= β) where

$$\frac{v_{out}}{i_{in}} = -R_2 \tag{30.29}$$

If we assume the input current source has a source resistance of R_1 so that $v_{in} = i_{in} \cdot R_1$ then

$$|A_{CL}| = \left|\frac{v_{out}}{v_{in}}\right| = \frac{R_2}{R_1} \tag{30.30}$$

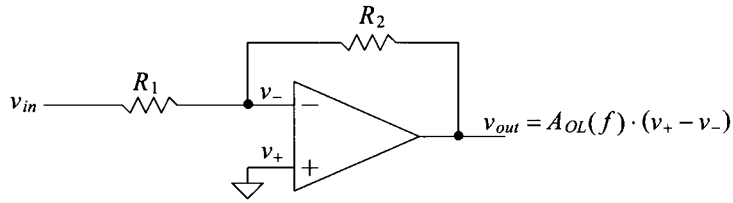


Figure 30.23 Inverting op-amp topology.

Keeping in mind that the closed-loop bandwidth of the inverting amplifier is still, from Eq. (30.25),

$$f_{CL,3dB} \approx \frac{R_1}{R_1 + R_2} \cdot f_{un} \tag{30.31}$$

we can write

$$\text{Gain} \cdot \text{bandwidth} = \frac{R_2}{R_1 + R_2} \cdot f_{un} \tag{30.32}$$

Example 30.4

Compare the bandwidth of a +1 gain amplifier implemented using a noninverting op-amp topology (Fig. 30.22) to the bandwidth of a −1 gain amplifier using the inverting op-amp topology (Fig. 30.23).

Using Eq. (30.26), the bandwidth of the +1 gain amplifier is f_{un}. This amplifier is commonly known as a unity voltage follower and has $R_1 = \infty$ (an open) and $R_2 = 0$ (a short). The bandwidth of the inverting, −1, gain amplifier can be determined using Eq. (30.32) with $R_1 = R_2$ and is $0.5f_{un}$. *This result is important because it shows that for the fastest speed the noninverting op-amp topology offers the best choice.* Practically, however, the nonlinearities related to the finite *CMRR* (see Eq. [30.8]) force the use of inverting op-amp topologies. As discussed earlier, the *input common-mode voltage must remain constant* in any precision application. Note that a fully-differential op-amp topology is also a *shunt-shunt* amplifier with a gain bandwidth product given by Eq. (30.32). ■

Example 30.5

Comment on the derivations of Eqs. (29.59)-(29.60) in the last chapter.

The equations are still valid; however, if the op-amp topology is designed to keep the common-mode input voltage constant we would need to halve the value of β used in the equations. For example, for an ideal amplifier $\beta = 1/(2A_{CL})$. ■

30.2.1 Op-Amp Gain

In this section we answer the question of how large the DC open-loop gain of the op-amp, A_{OLDC}, must be in a data converter with a resolution of N bits. We know that the op-amp must amplify signals to within 1/2 LSB of the ideal value. Further we know that the closed-loop gain of an amplifier can be written as

$$A_{CL} = \frac{A_{OL}(f)}{1 + \beta \cdot A_{OL}(f)} \tag{30.33}$$

The feedback factor can be written, after reviewing Figs. 30.23 or 30.24, as

$$\beta = \frac{R_1}{R_1 + R_2} \text{ or } \frac{C_F}{C_F + C_I} \left[= \frac{1/j\omega C_I}{1/j\omega C_I + 1/j\omega C_F} \right] \tag{30.34}$$

As discussed in Ch. 29 the output of the amplifier will be equal to its ideal value minus some maximum deviation, ΔA. We can write the gain of the amplifier in Fig. 30.24

$$|A_{CL}| = \frac{C_I}{C_F} \tag{30.35}$$

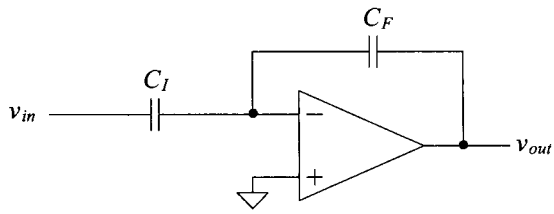


Figure 30.24 Inverting op-amp topology.

Next, let's write

$$|A_{CL}| = \frac{C_I}{C_F} - \Delta A = \frac{A_{OLDC}}{1 + A_{OLDC} \cdot \frac{C_F}{C_F + C_I}} \tag{30.36}$$

If the maximum value of ΔA is at most 1/2 LSB of the ideal gain, or,

$$\Delta A = \frac{C_I}{C_F} \cdot \frac{1/2\ LSB}{\text{Full scale output}} = \frac{C_I}{C_F} \cdot \frac{1/2 \cdot (V_{REF+} - V_{REF-})/2^N}{(V_{REF+} - V_{REF-})} = \frac{C_I}{C_F} \cdot \frac{1}{2^{N+1}} \tag{30.37}$$

then we can estimate the minimum required DC open-loop gain as

$$|A_{OLDC}| \geq \frac{1}{\beta} \cdot 2^{N+1} \tag{30.38}$$

If $\beta = 1/2$, as in the R-$2R$ DAC of Fig. 30.3 or when $C_I = C_F$, then

$$|A_{OLDC}| \geq 2^{N+2} \tag{30.39}$$

A 12-bit ADC or DAC requires the use of an op-amp with a gain greater than 16k while a 16-bit converter must have $|A_{OLDC}| \geq 256k$. Clearly, this estimate can present a real design concern. Note that Eq. (30.38) is optimistic. For a general design, an error of 1/2 LSB due just to op-amp gain is not desirable (so a larger value of A_{OLDC} must be used).

30.2.2 Op-Amp Unity Gain Frequency

The speed of a data converter is mainly limited by the op-amp used. In general, the minimum op-amp gain-bandwidth product, f_{un}, required for a specific settling time t (where t is less than $1/2f_{clk} = T_{clk}/2$, within a dead band of ±1/2 LSB) can be estimated, assuming no slew-rate limitations (see also Eqs. [30.18] and [30.19]), by

$$v_{out} = v_{outfinal}\left(1 - \frac{1}{2^{N+1}}\right) = v_{outfinal}(1 - e^{-t/\tau}) \tag{30.40}$$

where, once again,

$$\tau = \frac{1}{2\pi \cdot \beta \cdot f_{un}} \tag{30.41}$$

The minimum required op-amp unity gain frequency is then given by

$$f_{un} \geq \frac{f_{clk} \cdot \ln 2^{N+1}}{\pi \cdot \beta} \tag{30.42}$$

or, again assuming $\beta = 1/2$,

$$f_{un} \geq 0.44 \cdot (N+1) \cdot f_{clk} \tag{30.43}$$

If we design a 12-bit ADC that is clocked at 100 MHz, we need to use op-amps with unity gain frequencies, f_{un}, of 572 MHz (and a DC gain of at least 16k). Again, this estimate for the unity gain frequency is optimistic. A good design would use a larger f_{un} than what is specified by Eq. (30.43).

30.2.3 Op-Amp Offset

A critical characteristic of any op-amp used in a data converter is its offset voltage. We introduced the concept of reducing the offset voltage of an op-amp back in Ch. 25. Here we provide additional comments and possibilities for offset reduction.

Adding an Auxiliary Input Port

A simple method of nulling the offset voltage of an op-amp is shown in Fig. 30.25. In this figure the added MOSFETs, M1 and M2 (which operate in the triode region) are essentially used to balance the current flowing in the current mirror load. We can think of the added MOSFETs as providing an auxiliary input port for offset calibration.

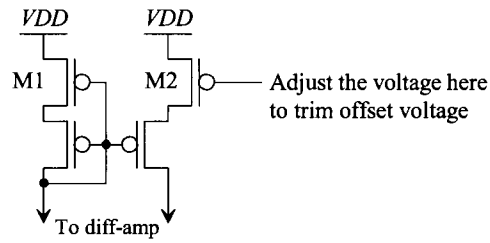


Figure 30.25 Trimming offset using an auxiliary input port.

Figure 30.26 shows how we would use the auxiliary input port to remove (lower) the offset. When zeroing out the offset, the op-amp is removed from the circuit by opening S1 (and possibly a switch [not shown] in series with the op-amp's output). This is followed by closing S2 and S3 so that a control voltage is stored on C. Note that we have assumed that the op-amp is used in an inverting configuration (that is, the noninverting input of the op-amp, +, is tied to V_{CM}). The offset removal is dynamic and will have to be performed periodically. We could also use a simple R-$2R$ DAC with a topology similar to what is seen in Fig. 30.8 to calibrate out the offset (eliminating the dynamic nature of the method). The output of the DAC would be connected to the auxiliary input port. Note that an increase in voltage on the auxiliary input port must result in a decrease in output

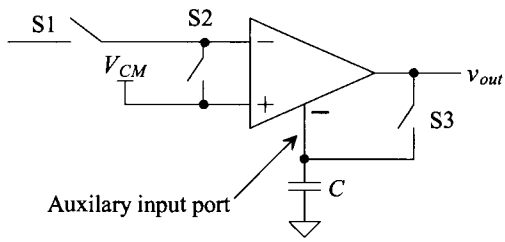


Figure 30.26 Using an auxiliary input port to lower offset.

voltage. (There must be negative feedback when connecting the output of the op-amp to the input port.)

The practical problem with the topology of Fig. 30.26 is the charge injection and capacitive feedthrough resulting from shutting off (opening) S3. This "glitch" of charge causes a change in the auxiliary port's input voltage and can place a significant limitation on the minimum possible offset voltage attainable after calibration. The amplitude of the glitch can be reduced by increasing C or by increasing the length of the MOSFET used in the op-amp (M2 in Fig. 30.25). Increasing the length results in a decrease in the MOSFET's transconductance (keeping in mind that the MOSFET is operating in the triode region) making the amplitude of the glitch less harmful. The drawback of increasing the MOSFET's length is that the range of offset voltages we can remove is reduced.

Example 30.6

Suppose perfect switches are available for the circuit of Fig. 30.26. Estimate the residual offset voltage in terms of the op-amp's gain, A_G, from the auxiliary port to the op-amp output.

If the offset voltage before reduction is V_{OS}, then the offset voltage after reduction is V_{OS}/A_G. For reasonable values of A_G the final inherent offset voltage is negligible. The point of this example is that the charge injection and capacitive feedthrough from the switches is the dominant source of offset error using this technique. ■

We've seen the problem of charge injection and capacitive feedthrough before. The most common technique for reducing its effect is to use a fully-differential topology. Figure 30.27 shows a modification of Figs. 30.25 and 30.26 to compensate for charge injection. The idea is that when S4 and S3 turn off (open) the variation in voltages on the gates of M1 and M2 are equal resulting in a common change in each MOSFET's resistance. Ideally then the current will remain balanced in the diff-amp. Note that while we've shown the use of triode-operating MOSFETs M1/M2 in Figs. 30.25 and 30.27 in series with the load of a diff-amp on the input of an op-amp, we could also use this concept in later stages of the op-amp.

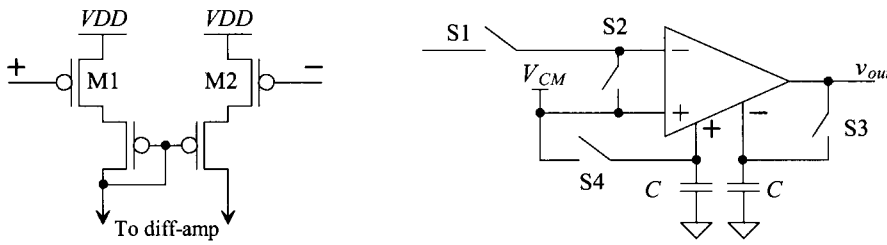


Figure 30.27 Using an auxiliary input port to lower offset (two terminals).

Figure 30.28 shows another possible topology for offset removal using an auxiliary input. An additional diff-amp is added in parallel to the main input diff-amp stage of an op-amp to balance the currents and zero out the offset voltage. Again, long

length MOSFETs are used in the added input so that the glitches resulting from the imperfections in the MOSFET switches (S4 and S3 in Fig. 30.27) have the least effect on the operation of the circuit.

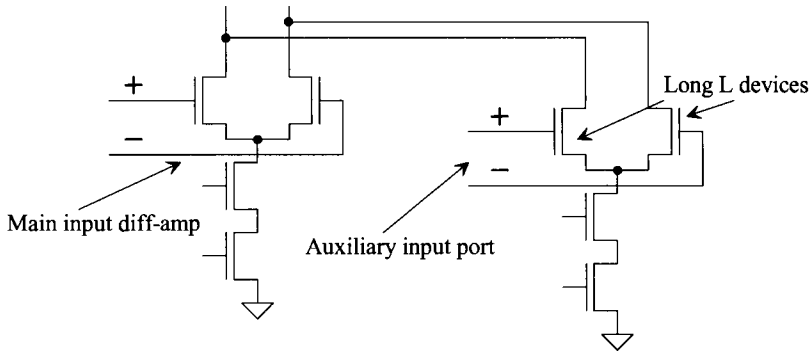


Figure 30.28 Using an auxiliary diff-amp for balancing current in an op-amp's input.

We can estimate the maximum offset voltage we can zero out using the technique of Fig. 30.28 by writing the imbalance in the main diff-amp's currents because of its offset voltage as

$$g_m \cdot V_{OS,\max} = i_d \tag{30.44}$$

The auxiliary input must sum the opposite of this current in the main diff-amp's load to balance the currents in the main diff-amp (and hence eliminate the offset voltage). If we label the transconductance of the diff-amp used in the auxiliary input g_{maux} and the maximum allowable differential voltage on the auxiliary input for linear operation $V_{aux,\max}$ then we can write

$$g_m \cdot V_{OS,\max} = g_{maux} \cdot V_{aux,\max} \tag{30.45}$$

Because we are using long length devices in the auxiliary input $g_{maux} << g_m$ for the same biasing current levels. If $V_{aux,\max} = 200$ mV (a differential voltage of ± 200 mV will cause all of the diff-amp tail current to flow through one side of the diff-amp) and $g_m/g_{maux} = 10$, then we can zero out at most 20 mV of op-amp offset.

The offset storage technique shown in Fig. 30.27 relies on the removal of the op-amp from the circuit while autozeroing the offset. The scheme in Fig. 30.29 shows how the technique can be extended to remove the offset while leaving the main op-amp, O1, in the circuit at all times. When S2, S3, and S4 are closed, the offset of O2 is zeroed out. At this time switches S1 and S5 are open. After O2's offset is stored, S2, S3, and S4 are then opened. Next S1 and S5 close. O2 is used to precisely set the inverting input of O1 to V_{CM} through the feedback around O1 (not shown). When O2 goes back to zeroing out its own offset (S2-S4 close) the capacitor connected to the auxiliary port of O1 retains the charge, and thus voltage, needed to keep O1's offset nulled out to zero. Again this capacitor should be large to avoid problems from the imperfections of S5.

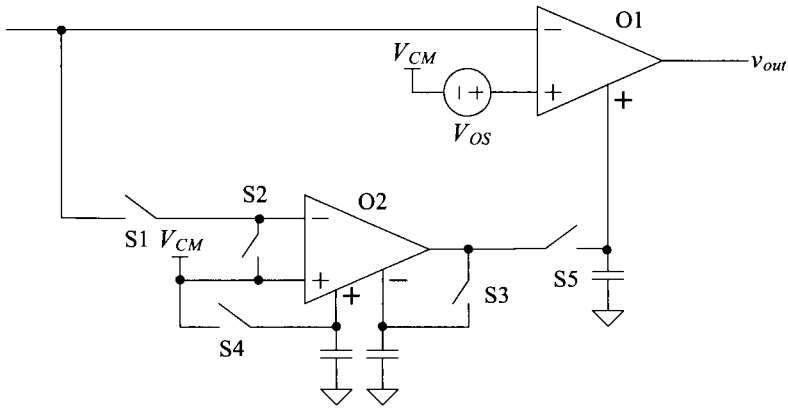


Figure 30.29 Continuous-time offset removal.

30.3 Implementing ADCs

In this section we continue to discuss implementing data converters with design concerns for S/Hs, cyclic ADCs, and pipeline ADCs.

30.3.1 Implementing the S/H

We assume that the reader is familiar with the fundamental implementation of a CMOS S/H discussed in Ch. 25 and in particular bottom-plate sampling. Figure 30.30 shows the more general implementation of a S/H. Note that if C_I goes to 0 (an open) this topology reduces to the basic S/H given in Ch. 25 (repeated in Fig. 30.31 for convenience).

We can determine the relationship between the input of the S/H and its output by writing the charge stored on C_I and C_F when the ϕ_1 and ϕ_2 switches are closed (the ϕ_3 switches are open) as

$$Q_{I,F}^{\phi_1} = C_{I,F} \cdot (v_{in} - V_{CM} \pm V_{OS}) \tag{30.46}$$

where V_{OS} is the offset voltage of the op-amp and the input (and output) voltages are referenced to ground (v_{in} [v_{out}] varies from 0 to $2V_{CM}$ [= VDD here]). Note that the reason why the ϕ_2 switches turn off slightly after the ϕ_1 switches was discussed in Sec. 25.2.1 (bottom plate sampling, an important technique used in the circuits we present here). When ϕ_3 goes high, the charge on C_I is

$$Q_I^{\phi_3} = C_I \cdot (V_{CM} - V_{CM} \pm V_{OS}) \tag{30.47}$$

The difference between $Q_I^{\phi_1}$ and $Q_I^{\phi_3}$ is transferred to C_F when ϕ_3 goes high. The output voltage is then determined knowing charge must be conserved

$$C_F \cdot (v_{out} - V_{CM} \pm V_{OS}) = \overbrace{C_F \cdot (v_{in} - V_{CM} \pm V_{OS})}^{Q_F^{\phi_1}} + \overbrace{C_I \cdot (v_{in} - V_{CM} \pm V_{OS})}^{Q_I^{\phi_1}} - \overbrace{C_I \cdot (V_{CM} - V_{CM} \pm V_{OS})}^{Q_I^{\phi_3}} \tag{30.48}$$

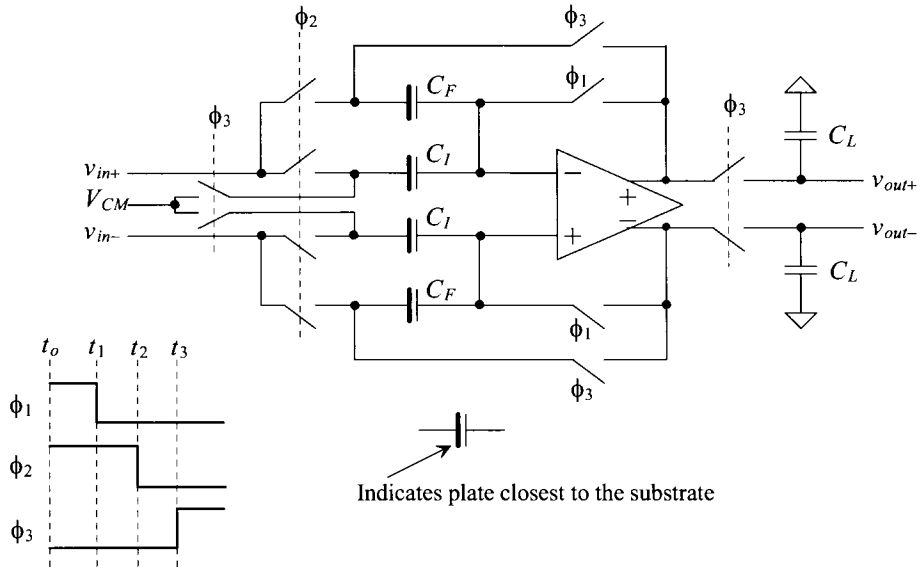


Figure 30.30 Data converter S/H building block.

or when ϕ_3 goes high

$$v_{out} = \left(1 + \frac{C_I}{C_F}\right) \cdot v_{in} - \frac{C_I}{C_F} \cdot V_{CM} \tag{30.49}$$

Notice how the op-amp offset is autozeroed out. The ideal residual offset is V_{OS}/A_{OL}. Practically, the residual offset is limited by the imperfections in the switches (which, once again, forces us to use fully-differential topologies). Also note, in Fig. 30.30, that we have drawn the input capacitance of the next stage as a load, C_L. This was so that the output of the S/H would appear to change only on the rising edge of ϕ_3 (plus the output settling time). Finally, if the S/H is clocked at $f_{clk} = 1/T_{clk}$, the output of the S/H must settle to less than 1/2 LSB, worst-case, in a time of $T_{clk}/2$. The minimum required value of op-amp unity gain frequency was specified in Eq. (30.42).

Equation (30.49) can be used to determine the relationship between v_{in} and v_{out} for fully-differential signals

$$v_{out} = v_{out+} - v_{out-} = \left(1 + \frac{C_I}{C_F}\right) \cdot (v_{in+} - v_{in-}) \tag{30.50}$$

Note how, as we would expect, the common-mode voltage subtracts out of the relationship when we take the difference between v_{out+} and v_{out-}. A block diagram representing the S/H of Fig. 30.30 is shown in Fig. 30.32. The use of block diagrams can be very useful to describe data converter architectures.

Example 30.7

Simulate the operation of the data converter S/H building block shown in Fig. 30.30. Assume $C_I = C_F = 1$ pF and $f_s = 100$ MHz.

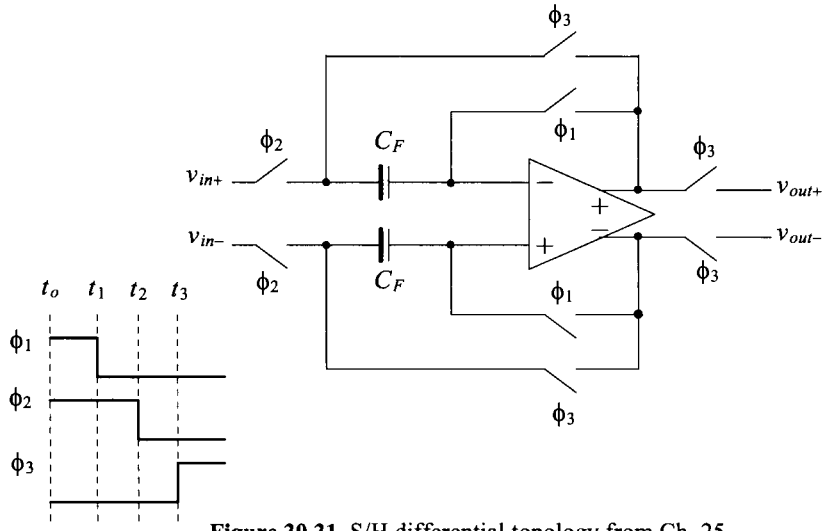


Figure 30.31 S/H differential topology from Ch. 25.

The simulation results are shown in Fig. 30.33. In part (a) the clock signals are shown. Unlike the clock signals shown in Fig. 30.30 where the falling edge of ϕ_2 is delayed from ϕ_1, the simulation sets the signals so they go low at the same time. This was to avoid the outputs of the op-amp changing to very large values for the small amount of time the op-amp operates open-loop with an input signal applied.

In part (b) we show the op-amp outputs. Note how, when ϕ_1 goes high, both outputs are set to the common-mode voltage by forcing the op-amp into a follower configuration (which may lead us to use switches to short the terminals of the op-amp to V_{CM} when ϕ_1 is high if offset isn't important). When ϕ_3 goes high, the circuit behaves as an S/H with a gain of two. Part (c) of the figure shows the outputs connected through ϕ_3 switches, as seen in Fig. 30.30, driving 10 pF load capacitances. ■

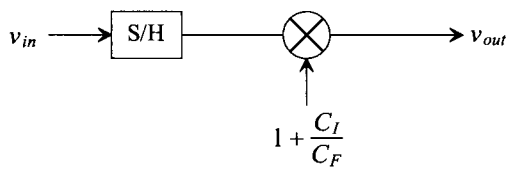


Figure 30.32 Block diagram for the S/H of Fig. 30.30.

A Single-Ended to Differential Output S/H

Note how we have assumed, in the S/H of Fig. 30.30, that the input voltage was fully-differential. In many practical applications the input to the ADC is single-ended. While we can connect v_{in-} to V_{CM} in an attempt to change the single-ended input into a fully-differential sampled output, the practical problem is the variation of the op-amp's

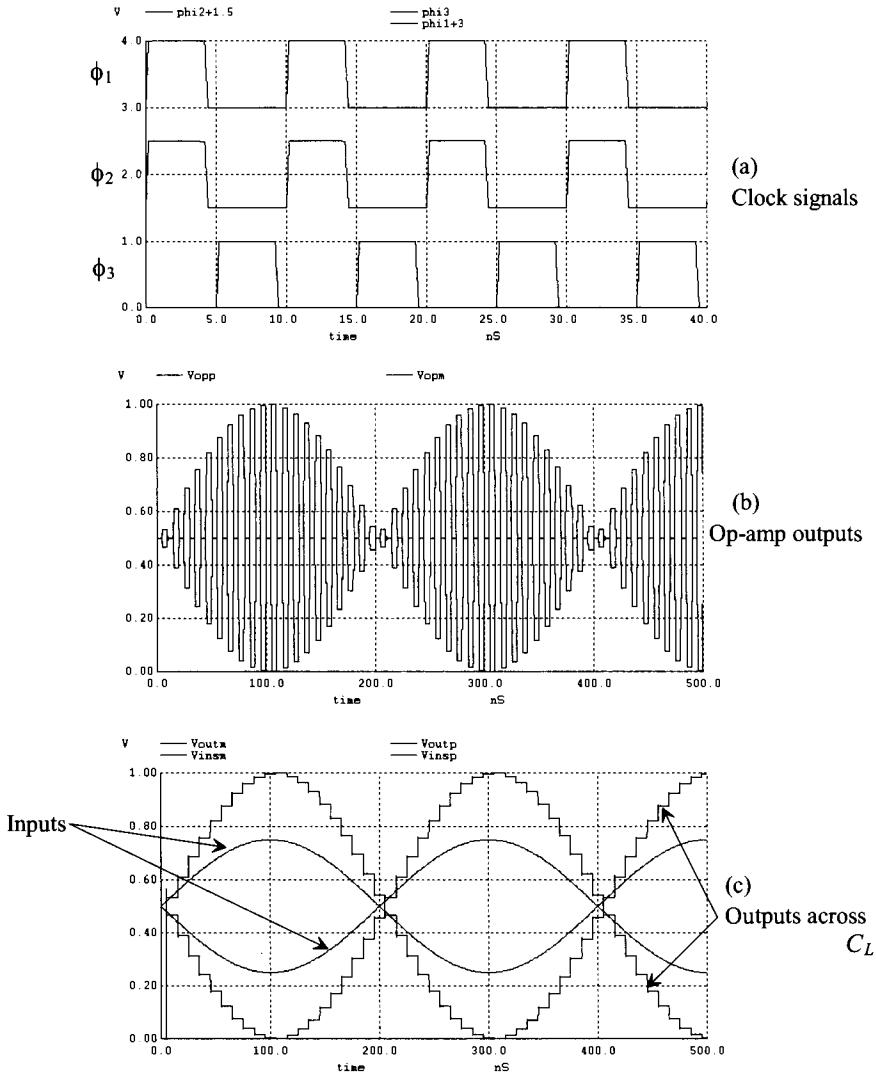


Figure 30.33 SPICE simulations of the operation of Fig. 30.30.

input common-mode voltage. As we've already discussed, precision data converters must use op-amp configurations where the input common-mode voltage is constant. Also, and perhaps more practically, the range of allowable common-mode voltages can be very restricted when designing low-voltage circuits. As we saw when designing mixed-signal op-amps in Ch. 26, the minimum input common-mode voltage can be very close to V_{CM} in nanometer CMOS. A technique to keep the op-amp's input common-mode voltage at V_{CM} when a single-ended input is applied to the S/H is seen in Fig. 30.34. During the sample phase all of the bottom plates of the capacitors are connected to V_{CM} except for one, which is connected to the single-ended input. The op-amp is in the unity-follower configuration

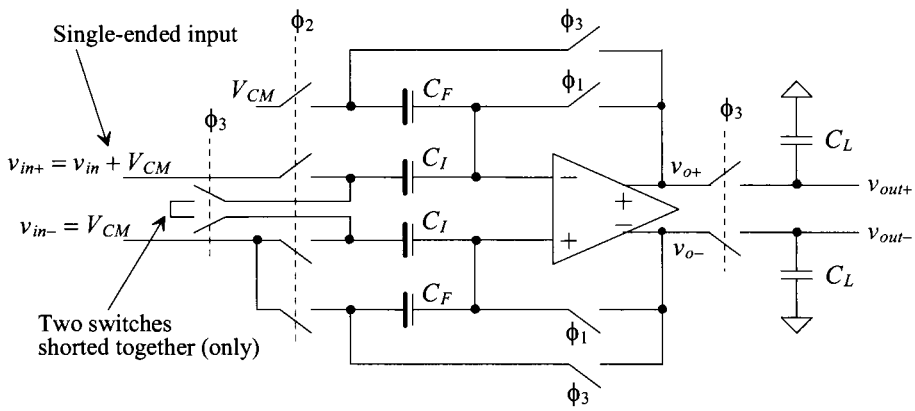


Figure 30.34 Single-ended to differential S/H.

and the bottom-plate sampling techniques we discussed earlier are still used. When the ϕ_3 switches are closed (ϕ_1 and ϕ_2 are open) the bottom plates of the C_I capacitors are shorted together but to nothing else. This causes the charge stored on the bottom-plate of the top C_I capacitor (the capacitor connected to the single-ended input v_{in+}) to redistribute between both C_I capacitors making the input appear to be fully differential. Because of the single-ended input and changed connections to V_{CM}, the gain of the topology is C_I/C_F.

Example 30.8

Simulate the operation of the S/Hs shown in both Figs. 30.31 and 30.34. Assume the S/H is clocked at 100 MHz, v_{in+} is a sinewave that swings from ground to *VDD*, and v_{in-} is connected to V_{CM} (the input signal is single-ended and covers the entire supply range). Show how the op-amp's input common-mode voltage range changes or doesn't change. Assume all capacitors are 1 pF and then show how a 10% mismatch in the capacitors affects the output of the S/H.

Figure 30.35 shows the simulation results for the S/H seen in Fig. 30.34. Notice how the op-amp's input common-mode voltage stays constant at V_{CM}. While the simulation results show the op-amp's outputs, v_{o+} and v_{o-}, and not the difference in the outputs, if we do take the difference (with all capacitors at 1 pF and so the gain of the S/H should be one) we indeed see that the gain is one. Next if we modify the simulation and change one C_I capacitor to 1.1 pF the simulation results show a gain error. As we'll discuss in a moment the big benefit of using the topology seen in Fig. 30.31 is that capacitor matching isn't important.

Next, Fig. 30.36 shows the simulation results for the S/H seen in Fig. 30.31. In part (a) the input common-mode voltage of the op-amp is shown. When the ϕ_1 switches are closed, the voltage returns to V_{CM} (the op-amp is placed in a follower configuration where the input signal charges the two capacitors). The input common mode voltage varies between 750 mV and 250 mV, a concern for most op-amp designs using a single input diff-amp. In part (b) the rail-to-rail input signal is converted into a differential S/H output signal. Note that the gain of the S/H is one. Note also how, unlike the op-amp outputs in Fig. 30.33, the outputs

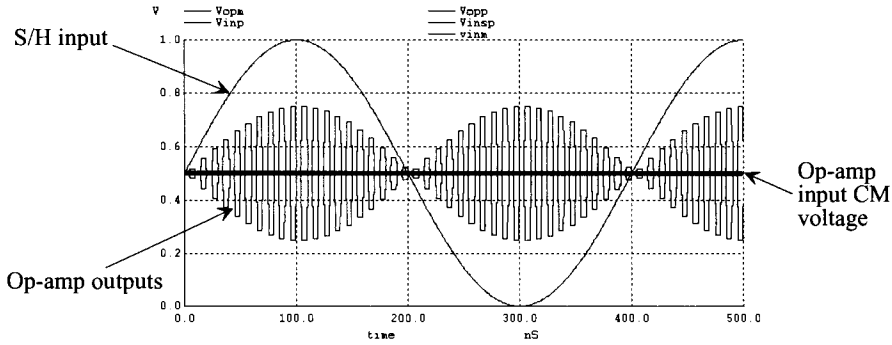
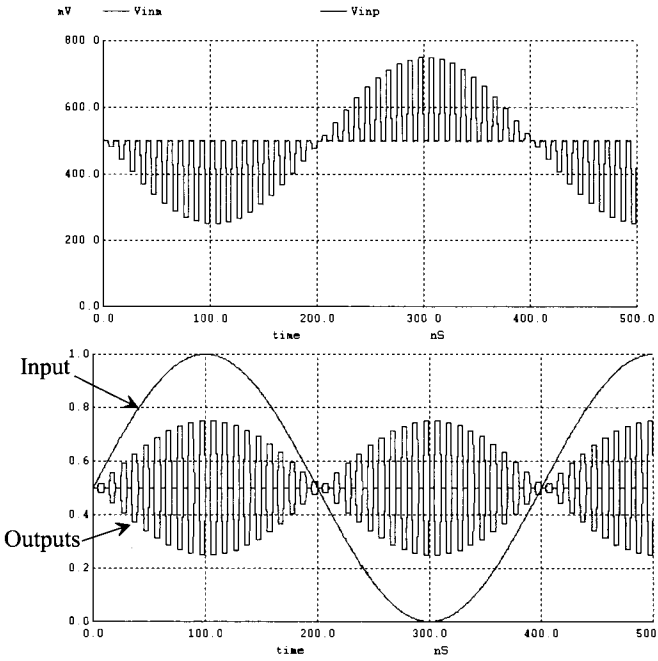


Figure 30.35 Simulating the operation of the S/H in Fig. 30.34.

are limited to $V_{CM} + V_p/2$ where V_p is the peak amplitude of the input sinewave (= 500 mV here). When the input sinewave has an amplitude of 1 V (500 mV above V_{CM}), the positive output is 750 mV and the negative op-amp output is 250 mV. Subtracting the op-amp outputs results in 500 mV (the same voltage as the input signal referenced to V_{CM}). It's important to understand how going from a single-ended signal to a fully-differential signal may result in a reduction in the op-amp output swing.



(a) Showing how the input common-mode voltage can vary in a S/H with a single-ended input.

(b) Input and outputs of a S/H performing single-ended to differential coversion.

Figure 30.36 Simulating the S/H in Fig. 30.31.

The simulation results seen in Fig. 30.36 were generated using 0.9 pF and 1.0 pF sampling capacitors (labeled C_F in Fig. 30.31). Because the feedback factor is unity, these capacitors will not affect the gain of the S/H. The point here is that the matching of the capacitors isn't important for a precise gain of one when using this (Fig. 30.31) S/H topology. (The op-amp open-loop gain, however, is still important.) Again, while the topology seen in Fig. 30.34 is sensitive to capacitor matching, it still may be the S/H topology of choice because the op-amp's input CMR remains constant with a single-ended input signal. ■

Before leaving this section let's remember that a common-mode feedback (CMFB) circuit is still required to precisely balance the outputs of the op-amp. In addition to the op-amp designs discussed in Ch. 26, Fig. 30.37 shows another possible design. When the ϕ_1 switches are closed, the outputs are connected to the CMFB amplifier (see Fig. 30.37a). Also when the ϕ_1 switches are closed the op-amp is placed in the unity feedback configuration (not shown). The CMFB circuit is used to ensure that the input voltages are $V_{CM} \pm V_{OS}$. The op-amp in Fig. 30.37b is derived from the designs presented in Ch. 26 but without the output buffers. The benefit of removing the output buffers when driving purely capacitive loads is the ease of attaining a 90-degree phase margin (and so clean settling behavior). The drawbacks of not using an output buffer are the reduced gain and the need to increase the biasing currents and device sizes to drive a given load capacitance. Again, the gain-boosting amplifiers labeled N and P in part (b) can be compensated, if needed, using capacitors at their outputs to ground (or *VDD*). The CMFB amplifier is seen in Fig. 30.37c.

30.3.2 The Cyclic ADC

Cyclic or algorithmic DACs were first discussed back in Sec. 29.1.5. Here we present the concept of a cyclic ADC. A block diagram of a cyclic ADC is seen in Fig. 30.38 assuming an 8-bit ($N = 8$) conversion. The input signal is sampled on the rising edge of every *eighth* (N) clock pulse. On the rising edge of every clock pulse the comparator determines if the S/H input is above or below the common-mode voltage. If it is below V_{CM}, nothing is subtracted from the S/H output. If it is above V_{CM}, then V_{CM} is subtracted from the S/H output. In either case the resulting output is multiplied by two and cycled back to the S/H input. Each time the comparator output goes high the value is stored in a shift register. When the conversion is complete, the digital word stored in the shift register, which corresponds to the analog input voltage, is shifted into a hold register. The next conversion then begins on the following clock pulse starting with sampling the input voltage, v_{in}. Note that it takes N clock cycles for one conversion.

Example 30.9

Determine the output of the ADC in Fig. 30.38 if the input voltage is 1 V.

We begin by sampling the input voltage of 1 V. The output of the comparator is a logic 1 (MSB). Next, V_{CM} (= 0.5 V) is subtracted from the S/H output resulting in an output of 0.5 V. This output is multiplied by 2 resulting in 1 V. This 1 V output (the output of the multiplier) is cycled back to the S/H input.

Next, we sample the fed back voltage of 1 V, the output of the comparator is, again, a logic 1 (MSB – 1). V_{CM} (= 0.5 V) is subtracted from the S/H output

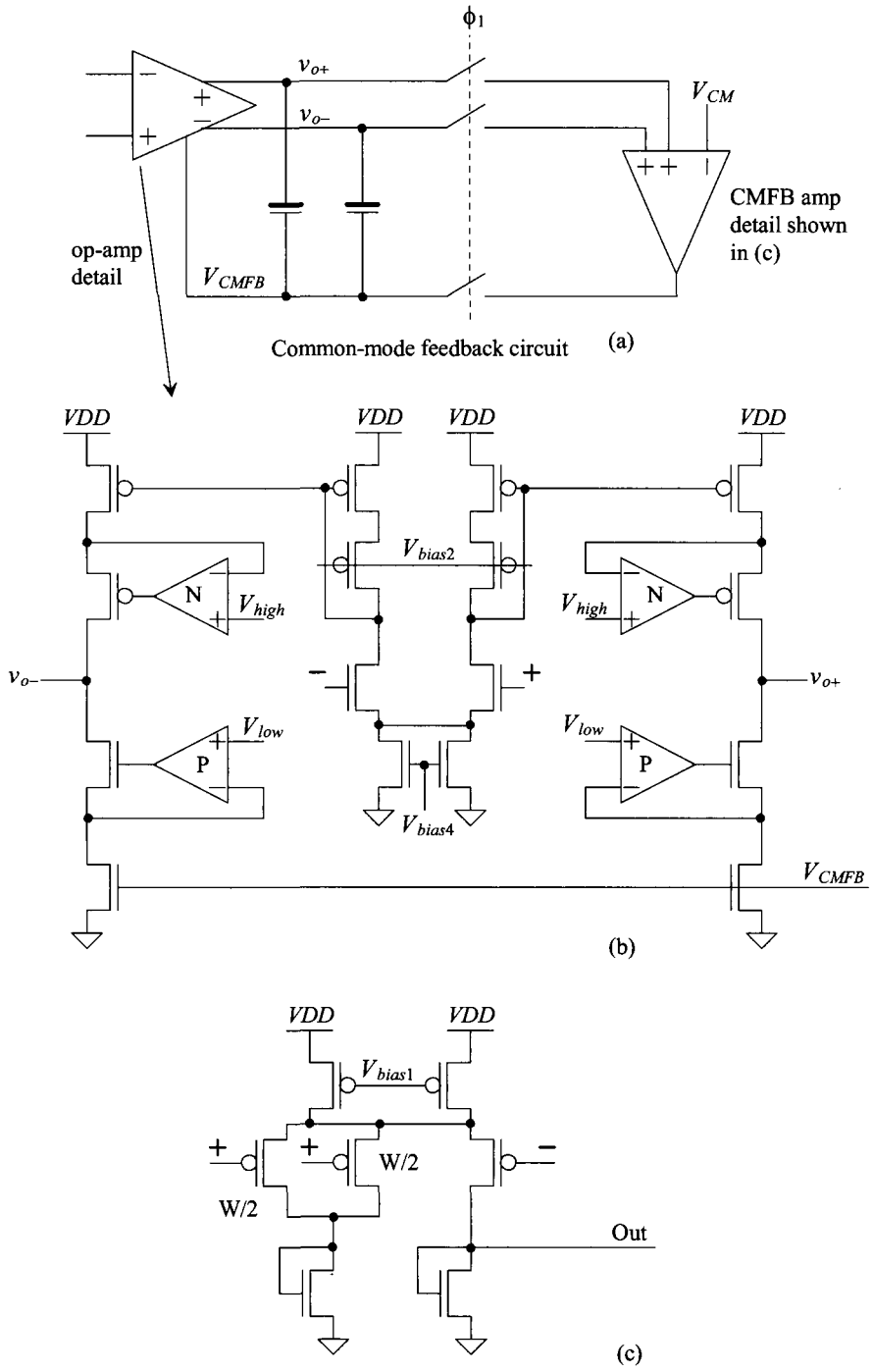


Figure 30.37 Mixed-signal op-amp for use in a S/H with CMFB.

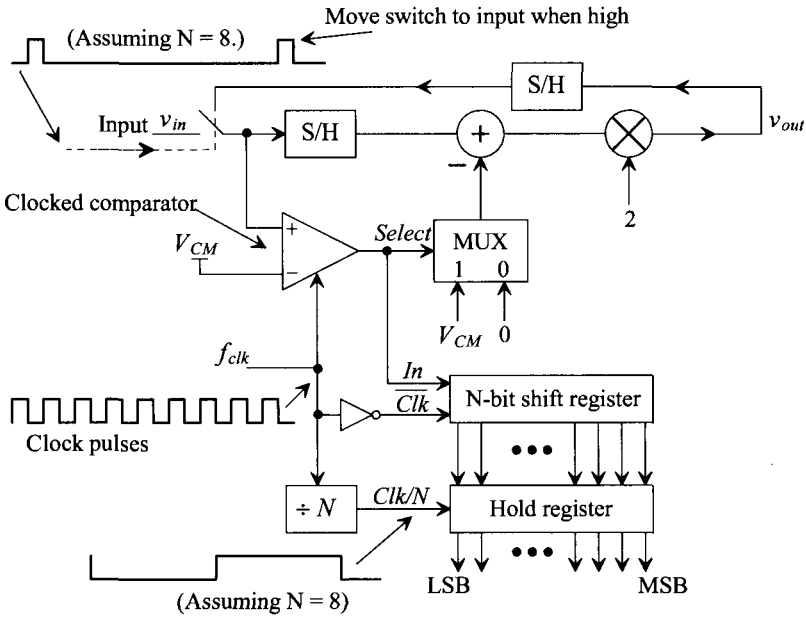


Figure 30.38 Block diagram of a cyclic ADC.

resulting in 0.5 V. This output is multiplied by 2 resulting in 1 V. This 1 V output (the output of the multiplier) is cycled back to the S/H input.

This continues and the final output of the ADC hold register is 1111 1111 (binary offset format). ■

Example 30.10

Repeat Ex. 30.9 if the Cyclic ADC input is 600 mV.

1. Sample the 600 mV input voltage. The comparator output goes high (MSB, b_7 , = 1). The output of the multiply by 2, after subtracting V_{CM} (= 500 mV) from the S/H output, is 200 mV.

2. Sample the 200 mV fed back voltage. The comparator output goes low (b_6 = 0). The output of the multiplier is 400 mV.

3. Sample 400 mV. The comparator output goes low (b_5 = 0). The output of the multiplier is 800 mV.

4. Sample 800 mV. The comparator output goes high (b_4 = 1). The output of the multiplier is 600 mV.

5. Sample 600 mV (b_3 = 1) output of the multiplier is 200 mV.

6. Sample 200 mV (b_2 = 0) output of the multiplier is 400 mV.

7. Sample 400 mV ($b_1 = 0$) output of the multiplier is 800 mV.

8. Sample 800 mV ($b_0 = 1$) output of the multiplier is 600 mV.

9. Sample the new input voltage and begin conversion again. The output word in the hold register is 1001 1001 (binary offset). ■

Comparator Placement

We showed the inverting input of the comparator in Fig. 30.38 connected to the common-mode voltage. In practice, however, we know that the comparator will have an offset or that the fed back signal may have a common-mode voltage slightly different than the ideal value. If the common-mode voltage of the fed back signals was, for example, 10 mV different than the ideal value, the comparator can make a wrong decision. Further, if the common-mode voltage is varying because of power supply noise or temperature changes then a wrong decision can occur even if some calibration scheme is employed. To avoid a wrong decision, the comparator is most often used in a fully-differential configuration, as seen in Fig. 30.39, with offset storage. The comparator can have significant kickback noise. By adding the ϕ_2 switches in series with the comparator input, we ensure the kickback noise doesn't corrupt the S/H input voltage. Note that since ϕ_3 and ϕ_2 are nonoverlapping, we guarantee that the comparator and S/H are disconnected when the comparator is clocked (and the kickback noise is generated). When the ϕ_1 switches are closed, the offset voltage of the comparator or the op-amp is zeroed out. The performance requirements of the comparator (gain and offset) can be greatly reduced (offset storage is not required) if we use 1.5 bits per clock cycle instead. We discuss this further in the next section.

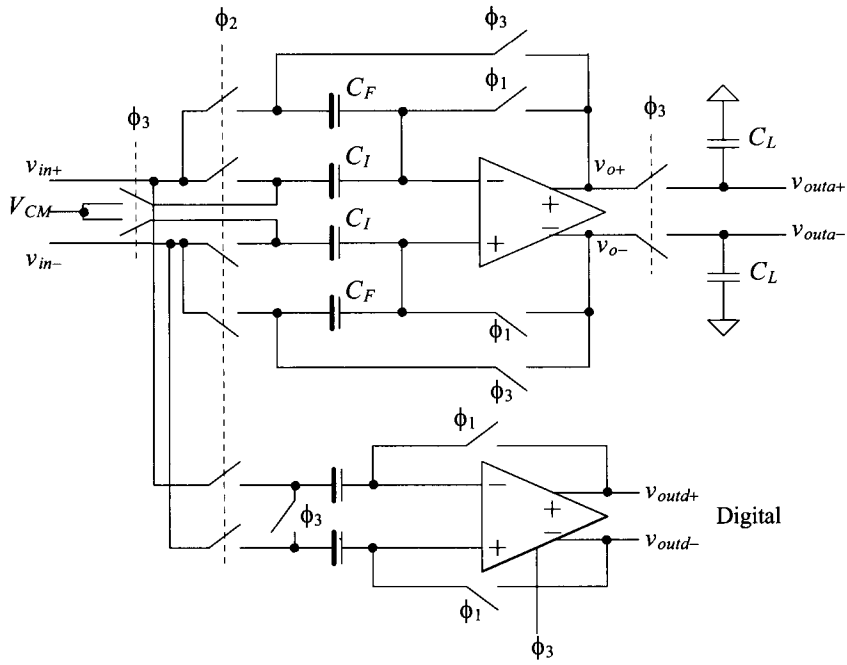


Figure 30.39 Implementation of the comparator with an S/H for use in a cyclic ADC.

Example 30.11

Estimate the gain required of a comparator used in a 10-bit cyclic ADC if V_{REF+} = 1 V and $V_{REF-} = 0$.

The LSB of this converter is $1/2^{10} \approx 1\ mV$. This means that the comparator gain must be large enough so that the output can fully transition to either *VDD* or ground with less than 1 mV difference on its inputs (assuming the offset voltage is zeroed out). In other words, the gain must be well above 1/1 mV or 2^{10}. Clearly this presents a real design concern. The gain of the positive feedback comparator may be very large because of the positive feedback used. However, the delay time of the comparator (time delay between the clock going high and the outputs transitioning all the way to *VDD* and ground) may be too long with such a little input voltage difference. Increasing the gain of the comparator, without care, can result in the comparator being unstable when placed in the unity feedback condition (the ϕ_1 switches closed in Fig. 30.39). To increase the comparator gain and avoid instability, a diff-amp (or two) can be added as a pre-amp in front of the positive feedback portion of the comparator. The offset of the diff-amp can then be zeroed out by placing the ϕ_1 switches between the diff-amp's output and its input. ■

Implementing Subtraction in the S/H

Notice in Fig. 30.38 how we can implement the S/H and then multiply by two by simply setting $C_F = C_I$ in Fig. 30.30. Reviewing Fig. 30.38 we see that it would also be useful to implement the subtraction in the S/H. In this figure we see that if the output of the MUX is 0 V, nothing needs to be changed in Fig. 30.30. However, if the MUX output is V_{CM}, then the S/H output must be reduced by V_{CM}. Consider what happens if, when ϕ_3 goes high, instead of connecting the bottom plate of C_I to V_{CM} in Fig. 30.30 we connect it to a voltage V_{CI+} (see Fig. 30.40). Doing this, after reviewing Eqs. (30.46) to (30.49), results in

$$Q_I^{\phi_3} = C_I \cdot (V_{CI+} - V_{CM} \pm V_{OS}) \tag{30.51}$$

or

$$v_{out+} = \left(1 + \frac{C_I}{C_F}\right) \cdot v_{in+} - \frac{C_I}{C_F} \cdot V_{CI+} \tag{30.52}$$

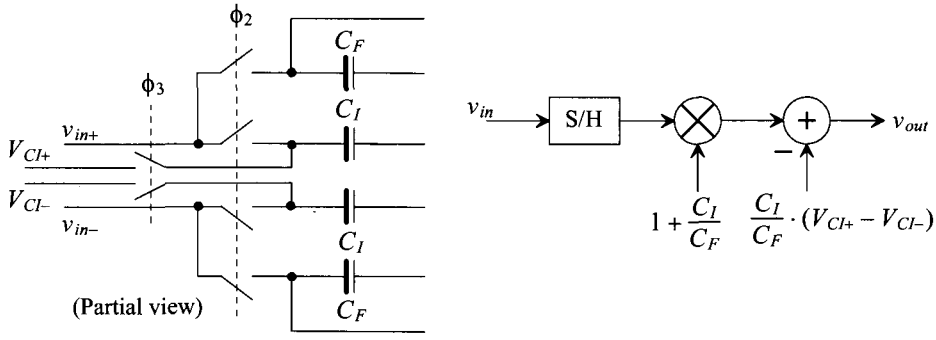


Figure 30.40 Implementing subtraction in the S/H.

The differential output voltage is then given by

$$v_{out} = v_{out+} - v_{out-} = \left(1 + \frac{C_I}{C_F}\right) \cdot (v_{in+} - v_{in-}) - \frac{C_I}{C_F} \cdot (V_{CI+} - V_{CI-}) \qquad (30.53)$$

We can easily rearrange the block diagram of Fig. 30.40 so that it more closely resembles the block diagram of the cyclic converter (Fig. 30.38), as seen in Fig. 30.41. If $C_F = C_I$, for our needed gain of two, we end up subtracting V_{CM} when V_{CI+} is $1.5V_{CM}$ and V_{CI-} is $0.5V_{CM}$. For the fully-differential I/O signal case then we are actually *subtracting* $V_{CM}/2$ when $v_{in+} > v_{in-}$ and *adding* $V_{CM}/2$ when $v_{in+} < v_{in-}$.

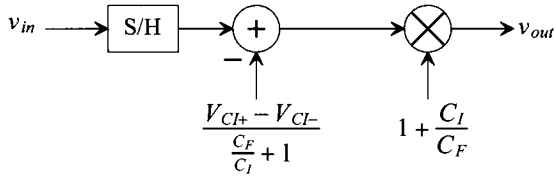


Figure 30.41 Block diagram of Fig. 30.30 with bottom plates of C_I tied to V_{CI}.

Example 30.12

Simulate the operation of the S/H shown in Fig. 30.42 if f_s = 100 MHz, $C_F = C_I$ = 1 pF, $V_{CI+} = 1.5V_{CM}$, and V_{CI-} is $0.5V_{CM}$. Comment on the resulting output.

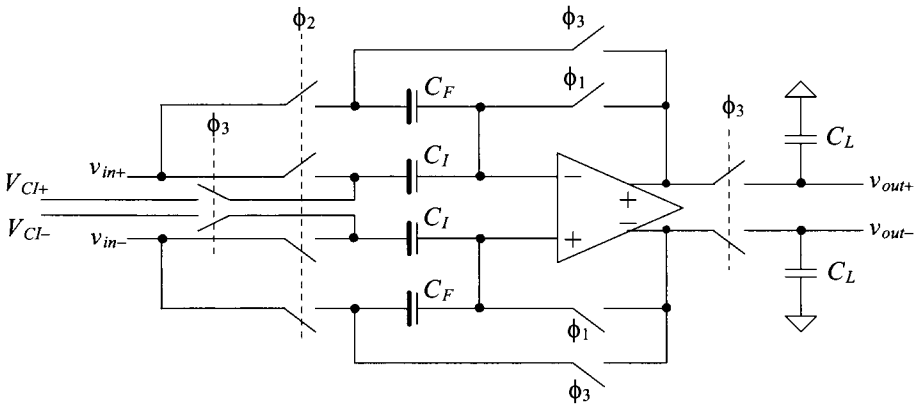


Figure 30.42 S/H used in Ex. 30.12

The simulation results are shown in Fig. 30.43. We only show the situation when we would want to subtract $V_{CM}/2$ from the differential input signal of the cyclic ADC, that is, when $v_{in+} > v_{in-}$. When the inputs are approximately the same voltage, the + input is approximately the same voltage as the – input and v_{in} is 0. We subtract 250 mV ($V_{CM}/2$) from the input and multiply by 2. The result, when the input is 0, is – 500 mV ($-V_{CM}$). When this happens, v_{out+} is 250 mV and v_{out-} is

750 mV. Taking the difference in these signals results in – 500 mV. At 100 ns in Fig. 30.43, for example, v_{in} is + V_{CM}. After we subtract $V_{CM}/2$ and multiply by 2, we get V_{CM} again (as indicated in the figure). ■

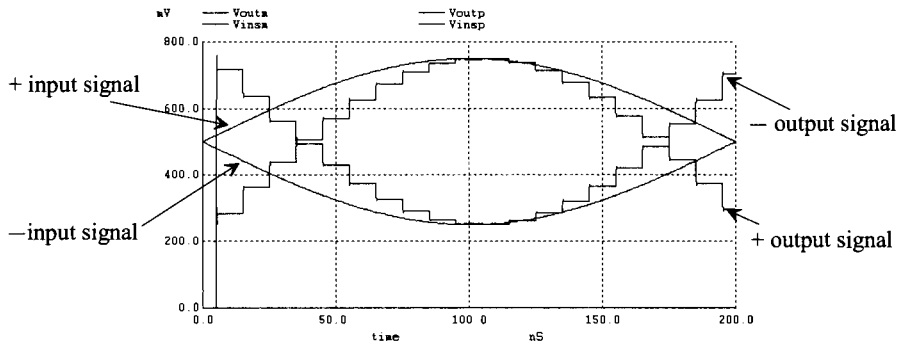


Figure 30.43 Simulation results for Ex. 30.12.

Example 30.13

Repeat Ex. 30.12 if we want to add $V_{CM}/2$ to the input signal.

We only want to add $V_{CM}/2$ to the input signal when $v_{in+} < v_{in-}$. In this situation we set $V_{CI+} = 0.5V_{CM}$ and V_{CI-} to $1.5V_{CM}$. The simulation results are shown in Fig. 30.44. ■

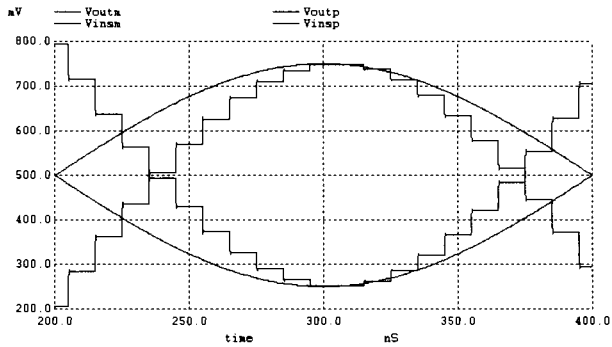


Figure 30.44 Simulation results for Ex. 30.13.

Let's write the analog output of the single-ended cyclic stage as

$$v_{out} = 2 \cdot (v_{in} + \overline{b} \cdot 0 - b \cdot V_{CM}) \tag{30.54}$$

where b is the digital (1 or 0) output of the comparator. Figure 30.45 shows the transfer curve for the cyclic stage. If the comparator output, b, is a 1 ($v_{in} > V_{CM}$), then we subtract V_{CM} from v_{in} before multiplying by two. Note that we have assumed

$$\frac{C_I}{C_F} = 1 \tag{30.55}$$

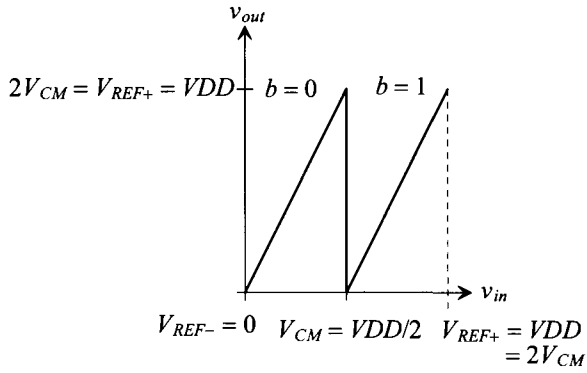


Figure 30.45 Transfer curve for the cyclic ADC (single-ended case).

Understanding Output Swing

After reviewing Fig. 30.45, the reader may wonder how the ADC will perform when the op-amp output must swing all the way up to *VDD* and down to ground. Any practical op-amp output voltage will become nonlinear as it approaches the supply rails. What must be realized is that the single-ended voltage, which ranges from 0 to *VDD*, is changed into a fully-differential voltage using the circuit of Fig. 30.30, which varies from *VDD*/4 to 3*VDD*/4 (as seen in Fig. 30.37). If we use our common-mode voltage as a reference (V_{CM} = *VDD*/2) for single-ended inputs, then we can show some conversions from single-ended to differential.

Single-ended input	Differential outputs
1. $v_{in} = 0.5VDD$	$v_{out+} = v_{out-} = V_{CM}$, $v_{out} = v_{out+} - v_{out-} = 0$
2. $v_{in} = VDD$	$v_{out+} = 0.75VDD, v_{out-} = 0.25VDD$, or $v_{out} = VDD/2$
3. $v_{in} = 0$	$v_{out+} = 0.25VDD, v_{out-} = 0.75VDD$, or $v_{out} = -VDD/2$
4. $v_{in} = 0.6VDD$	$v_{out+} = 0.55VDD, v_{out-} = 0.45VDD$, or $v_{out} = 0.1VDD$

Figure 30.46 shows the transfer curve of Fig. 30.45 redrawn to indicate that the signals, both input and output, are fully-differential. Note, as seen in Fig. 30.39, that the comparator output transitions from a 0 to a 1 when $V_{in+} > V_{in-}$ ($V_{in} > 0$).

We can rewrite Eq. (30.54) for fully-differential signals by looking at Fig. 30.46 and noticing that now, because the inputs and outputs are fully-differential and referenced to V_{CM} instead of 0 so to must be the voltages we add and subtract from the input, Eq. (30.54) can be written as

$$v_{out+} - v_{out-} = 2 \cdot \left(v_{in+} - v_{in-} + \overline{b} \cdot 0 - b \cdot V_{CM} \right) + \overline{b} \cdot V_{CM} + b \cdot V_{CM} \qquad (30.56)$$

or

$$v_{out+} - v_{out-} = 2 \cdot \left(v_{in+} - v_{in-} + \overline{b} \cdot 0 - b \cdot V_{CM} + \overline{b} \cdot \frac{V_{CM}}{2} + b \cdot \frac{V_{CM}}{2} \right) \qquad (30.57)$$

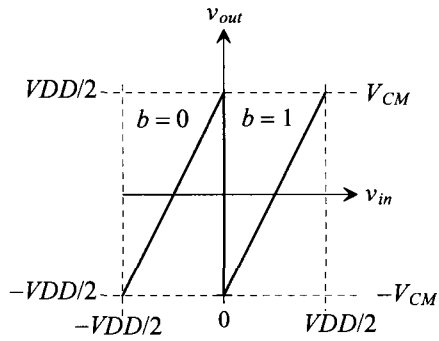


Figure 30.46 Transfer curve for the cyclic ADC when using fully-differential signals.

and finally

$$\overbrace{v_{out+} - v_{out-}}^{v_{out}} = 2 \cdot \left(\overbrace{v_{in+} - v_{in-}}^{v_{in}} + \overline{b} \cdot \frac{V_{CM}}{2} - b \cdot \frac{V_{CM}}{2} \right) \qquad (30.58)$$

Example 30.14

Using the fully-differential S/H stage of Fig. 30.42 simulate the transfer curve shown in Fig. 30.46.

The results of the simulation are shown in Fig. 30.47. To simulate v_{in}, the v_{in+} input is a ramp that varies from 250 mV to 750 mV while, at the same time, the v_{in-} is a ramp that varies from 750 mV to 250 mV. This results in v_{in} changing linearly from $-V_{CM}$ to $+V_{CM}$ (knowing $V_{CM} = VDD/2$). When the two ramps are equal, that is, when they are both V_{CM} ($v_{in} = 0$), V_{CI+} changes from $0.5V_{CM}$ to $1.5V_{CM}$, and V_{CI-} changes from $1.5V_{CM}$ to $0.5V_{CM}$ so that we go from adding $V_{CM}/2$ when $v_{in} < 0$ to subtracting $V_{CM}/2$ when $v_{in} > 0$. ■

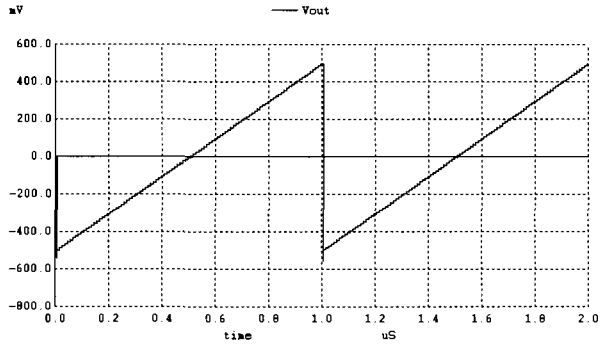


Figure 30.47 Simulating the transfer curves of a cyclic ADC stage.

30.3.3 The Pipeline ADC

One drawback of the cyclic ADC discussed in the previous section is the requirement of N clock cycles for each N-bit conversion. From Ch. 29 we know that the flash and pipeline (after an N-bit latency) topologies can perform an analog-to-digital conversion in one clock cycle. Another possibility for an N-bit conversion in one clock cycle is to use a time-interleaved topology. The basic idea is seen in Fig. 30.48. The S/Hs are sequentially clocked so that during each clock cycle the input voltage, v_{in}, is sampled, held, and applied to the input of an N-bit ADC. If the outputs of each ADC are then sequentially available through a MUX, the overall topology behaves as if it were a single ADC with flash-like performance. The practical problem with this topology is the matching between the ADCs and the sensitivity to clock skew and jitter. Differences in the DC characteristics of each ADC, for example, can result in measuring digital output values that change with time when a DC input signal is applied (a ripple on the output similar to what was seen in a noise-shaping data converter output). Mismatches in the ADCs can also result in a reduced (from ideal) signal-to-noise plus distortion ratio (*SNDR*).

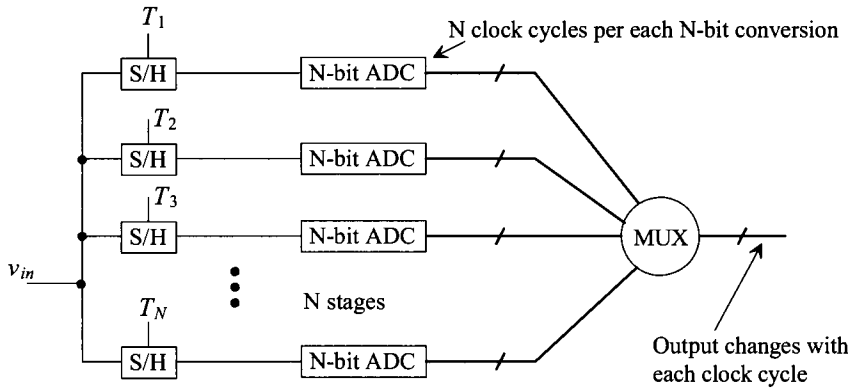


Figure 30.48 Time-interleaved operation of an ADC.

The pipelined ADC (see Sec. 29.2.3) can be thought of as an amplitude-interleaved topology where errors from one stage are correlated with errors from previous stages. The basic block diagram implementation of an N-bit pipelined ADC using the cyclic stage (see Fig. 30.42 for example) is seen in Fig. 30.49. Instead of cycling the analog output of the 1 bit/stage section back to its input, we feed the output into the next stage. The stages are clocked with opposite phases of the master clock signal. The comparator outputs are labeled *digital* in the figure. The digital comparator outputs are delayed through latches so that the final digital output word corresponds to the input signal sampled N clock cycles earlier. The first stage in Fig. 30.49 must be N-bit accurate (again see Sec. 29.2.3). It must amplify its analog output voltage, v_{N-1}, to within 1 LSB of the ideal value (after the subtraction of 0 or V_{CM} from the input signal and the multiplication by two). The second stage output, v_{N-2}, must be an analog voltage within 2 LSBs of its ideal value. The third stage output, v_{N-3}, must be an analog voltage within 4 LSBs of its ideal value and so forth. The important point here is that because the required accuracy of each stage decreases as we move down the line, the settling time, gain

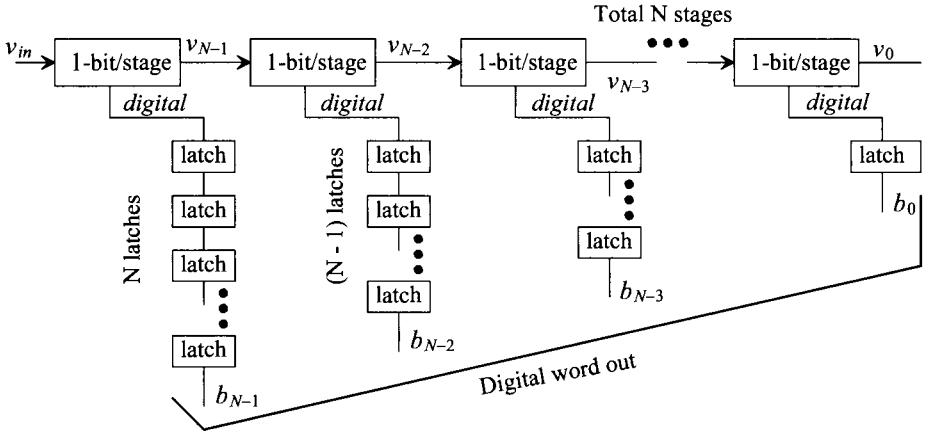


Figure 30.49 Pipelined ADC based on the cyclic stages discussed in the last section.

accuracy, and offsets all become less important. Smaller (and thus lower power) stages can be used for the later stages having, possibly, a dramatic effect on both layout size and power dissipation. While we're showing 1 bit/stage in Fig. 30.49, most commercially available pipeline ADCs use the *digital error correction* present in the 1.5 bit/stage topology discussed later.

We know from our analysis of pipeline ADC errors in Ch. 29 that they can be the result of comparator offsets, gain or linearity errors, and amplifier offsets. In the remainder of this section we discuss how to reduce the effects of these errors. We begin with a discussion of using the 1.5 bit/stage topology to make the comparator offsets (and the reference voltages used with the comparators) a "don't care." While we might think that using six stages with 1.5 bits/stage would result in an ADC with 9-bit resolution, we find that the extra 0.5 bit/stage is used to correct for errors introduced by the comparators. Using six stages will still result in a 6-bit ADC. We then cover the use of capacitor error averaging to set the gains of the amplifiers to precisely two. The cost of this technique is the increase in the number of clock cycles required for each stage's operation and a slightly more complex switching scheme. Finally, we cover some other topologies useful in amplifier offset removal and discuss offsets in general.

Using 1.5 Bits/Stage

As we saw in Ex. 30.11, the gain of the comparator (and the offset) can present a practical limitation to the operation of the ADC at high resolutions. The transfer curve of Fig. 30.45 relates the analog input of the cyclic ADC to its analog output voltage. The important point in this figure, besides the gain and linearity, is where the output of the comparator, *b*, transitions from a 0 to a 1. One-bit corresponds to two levels, i.e., a 0 or a 1. Two bits corresponds to four levels, i.e., 00, 01, 10, and 11. If we were to use three levels then we would get 1.5 bits of resolution. Using a thermometer code or decimal numbers, the three levels would then be

Thermometer, *ab*	Decimal
11	3
01	1
00	0

Next consider the transfer curves shown in Fig. 30.50 where three levels are used (1.5 bits). We can rewrite Eq. (30.54) for the 1.5 bit case as

$$v_{out} = 2 \cdot (v_{in} - \bar{a}\bar{b} \cdot 0 - \bar{a}b \cdot V_{CM} - ab \cdot 2V_{CM}) + V_{CM} \qquad (30.59)$$

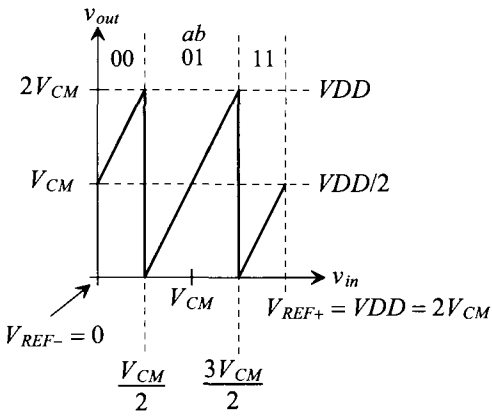
or

$$v_{out} = 2 \cdot (v_{in} + \bar{a}\bar{b} \cdot \frac{V_{CM}}{2} - \bar{a}b \cdot \frac{V_{CM}}{2} - ab \cdot \frac{3V_{CM}}{2}) \qquad (30.60)$$

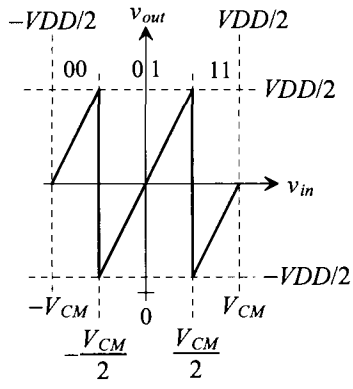
or if ab = 00 ($v_{in} < V_{CM}/2$), then $v_{out} = 2 \cdot (v_{in} + V_{CM}/2)$, if ab = 01 ($V_{CM} < v_{in} < 3V_{CM}/2$) then $v_{out} = 2 \cdot (v_{in} - 0.5V_{CM})$, and if ab = 11 then $v_{out} = 2 \cdot (v_{in} - 1.5V_{CM})$. For the fully-differential situation Eq. (30.59) can be rewritten as

$$v_{out+} - v_{out-} = 2 \cdot (v_{in+} - v_{in-} + \bar{a}\bar{b} \cdot V_{CM} - \bar{a}b \cdot 0 - ab \cdot V_{CM}) \qquad (30.61)$$

where all we did was reference, to V_{CM}, the voltages added/subtracted to v_{in} as seen in Eq. (30.58).



Single-ended input and output



Double-ended input and output

Figure 30.50 Transfer curves for using 1.5-bits per clock cycle.

Figure 30.51a shows how the comparators would be set up to determine *ab* and how we would provide the outputs. The outputs of the comparators are used as address inputs to a MUX. The MUX is used to set the bottom plate voltage of the C_I capacitor in Fig. 30.42. Figure 30.51b shows how we would implement the comparators if the input and output signals are double-ended.

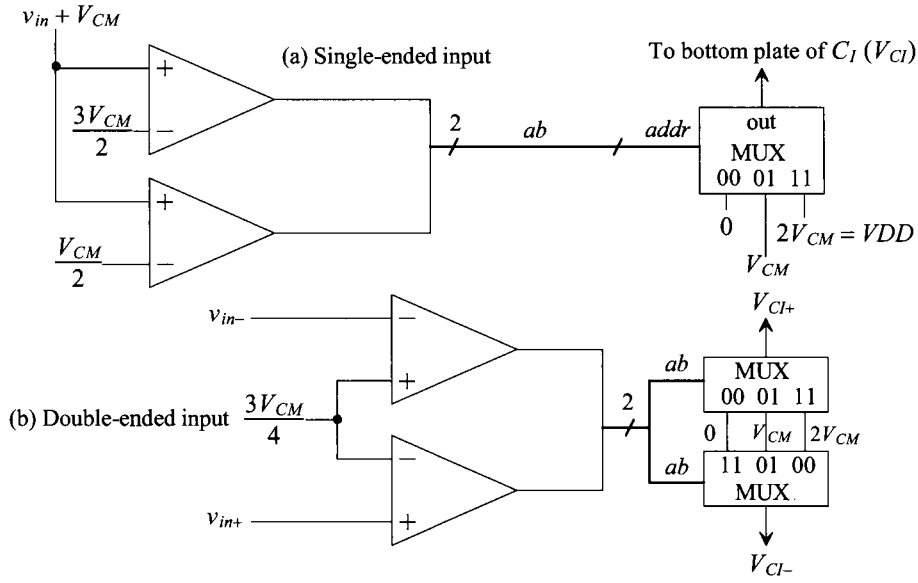


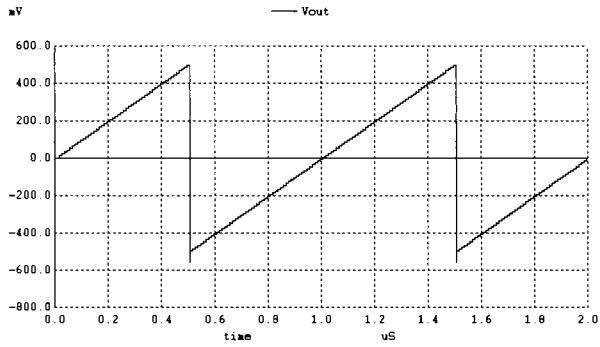
Figure 30.51 Implementing comparators and MUX for 1.5 bits.

Example 30.15

Repeat Ex. 30.14 if 1.5 bits/stage are used.

The simulation results are shown in Fig. 30.52. The same signals were used for the inputs (two ramps) here as were used in Ex. 30.14. The voltage V_{CI+} is 0 V when v_{in} is less than – 250 mV and is 0.5 when v_{in} is between – 250 mV and + 250 mV, while it is 1 V (= $VDD = 2V_{CM}$) when v_{in} is greater than + 250 mV. Notice how we only need three precision voltages, unlike the 1-bit/stage case, that is, VDD, V_{CM}, and 0. ■

Before going any further, let's answer, *"How can using 1.5 bits/stage eliminate the need for precision comparators?"* After reviewing Fig. 30.50, we see that an output of 11 cannot be followed by another output of 11 because we subtract V_{CM} from the input prior to multiplication by two. Even if the comparator has a reasonable offset (say less than 100



$VDD = 1\ V$

Figure 30.52 Simulating the transfer curves for 1.5 bits/stage.

mV) or low gain (say less than 50) resulting in a wrong decision, it's impossible for a 11 to be followed by another 11 or for a 00 to be followed by another 00. It is possible, however, to get a continuous string of 01 outputs (a simple example is when V_{CM} is applied to the ADC).

Let's now discuss how the outputs of the 1.5-bit stage are combined into the final ADC output word. Let's assume, again, that $V_{REF+} = VDD = 1$ V, $V_{REF-} = 0$, and $V_{CM} = 500$ mV. We know that the final N-bit output word can be converted back into an output voltage (with the unwanted quantization noise) using a DAC with the following weighting, Eq. (30.27),

$$v_{out} \text{ (if a DAC or } v_{in} \text{ if an ADC)} = b_{N-1} \cdot V_{CM} + b_{N-2} \cdot \frac{V_{CM}}{2} + b_{N-3} \cdot \frac{V_{CM}}{4} + \ldots + b_0 \cdot \overbrace{\frac{V_{CM}}{2^{N-1}}}^{1\ LSB} \tag{30.62}$$

Reviewing Eq. (30.54) note how if, on the first cycle, $b = 1$, we subtract V_{CM} from the input and then multiply the result by two. After a little thought, we should be able to see how we derive Eq. (30.62) from Eq. (30.54) after N clock cycles.

If the first thermometer code output of the 1.5-bit stage is labeled $a_{1.5N-1}b_{1.5N-1}$ and the second output is $a_{1.5N-2}b_{1.5N-2}$, etc., then we can write, with the help of Eq. (30.59), the relation between the ADC input (analog) and the ADC outputs (digital) as

$$\begin{aligned} v_{in} = {} & \overline{a_{1.5N-1}}b_{1.5N-1} \cdot V_{CM} + a_{1.5N-1}b_{1.5N-1} \cdot 2V_{CM} - \frac{V_{CM}}{2} \\ & + \overline{a_{1.5N-2}}b_{1.5N-2} \cdot \frac{V_{CM}}{2} + a_{1.5N-2}b_{1.5N-2} \cdot \frac{2V_{CM}}{2} - \frac{V_{CM}}{4} \\ & + \overline{a_{1.5N-3}}b_{1.5N-3} \cdot \frac{V_{CM}}{4} + a_{1.5N-3}b_{1.5N-3} \cdot \frac{2V_{CM}}{4} - \frac{V_{CM}}{8} + \ldots \end{aligned} \tag{30.63}$$

noting that we can group

$$-\frac{V_{CM}}{2} - \frac{V_{CM}}{4} - \frac{V_{CM}}{8} - \ldots = -V_{CM} \cdot \frac{2^N - 1}{2^N} = -V_{CM} + 0.5\ LSB + \overbrace{\frac{V_{REF-}}{2^N}}^{=0 \text{ here}} \tag{30.64}$$

which is nothing more than a level shift. Clearly we can combine outputs in the following manner to arrive at an equation similar to Eq. (30.62)

$$\begin{aligned} v_{in} = {} & (\overline{a_{1.5N-1}}b_{1.5N-1} + a_{1.5N-2}b_{1.5N-2}) \cdot V_{CM} + (\overline{a_{1.5N-2}}b_{1.5N-2} + a_{1.5N-3}b_{1.5N-3}) \cdot \frac{V_{CM}}{2} \\ & + (\overline{a_{1.5N-3}}b_{1.5N-3} + a_{1.5N-4}b_{1.5N-4}) \cdot \frac{V_{CM}}{4} + \ldots \\ & a_{1.5N-1}b_{1.5N-1} \cdot 2V_{CM} - V_{CM} + 0.5\ LSB \end{aligned} \tag{30.65}$$

Next notice that, when using a thermometer code, the only time $a_{1.5N-X}b_{1.5N-X}$ (the logical AND of ab) can be high is when both are high. The term $a_{1.5N-X}$ cannot be high while $b_{1.5N-X}$ is low (there is no such output code as 10, see Fig. 30.50). This means that we can replace $a_{1.5N-X}b_{1.5N-X}$ with simply $a_{1.5N-X}$. We can rewrite Eq. (30.65) as

$$v_{in} = a_{1.5N-1} \cdot 2V_{CM} + (\overline{a_{1.5N-1}} b_{1.5N-1} + a_{1.5N-2}) \cdot V_{CM} + (\overline{a_{1.5N-2}} b_{1.5N-2} + a_{1.5N-3}) \cdot \frac{V_{CM}}{2}$$

$$+ (\overline{a_{1.5N-3}} b_{1.5N-3} + a_{1.5N-4}) \cdot \frac{V_{CM}}{4} + \ldots + \overline{a_{1.50}} b_{1.50} \cdot \frac{V_{CM}}{2^{N-1}} - (V_{CM} - 0.5\ LSB) \qquad (30.66)$$

The + symbol in Eq. (30.66) indicates addition rather than a logical OR. If $\overline{a_{1.5N-1}} b_{1.5N-1} = 1$ and $a_{1.5N-2} = 1$, then the second term in this equation is $2V_{CM}$ and the first term must be 0. When this occurs, the addition of the two terms is 0 and a carry is generated. We can now use this information to write the relationship between Eq. (30.62) and Eq. (30.63) knowing $\oplus$ indicates an exclusive OR

$$b_0 = \overline{a_{1.50}} b_{1.50} \text{ with carry} = c_0 = a_{1.50} \qquad (30.67)$$

$$b_1 = \overline{a_{1.51}} b_{1.51} \oplus c_0 \text{ with } c_1 = \overline{a_{1.51}} b_{1.51} c_0 \qquad (30.68)$$

$$b_2 = \overline{a_{1.52}} b_{1.52} \oplus a_{1.51} \oplus c_1 \text{ with } c_2 = \overline{a_{1.52}} b_{1.52} a_{1.51} + c_1(\overline{a_{1.52}} b_{1.52} + a_{1.51}) \qquad (30.69)$$

noting each bit and carry are the outputs of a full adder. To simplify the carry equation, and the subsequent equations, we can substitute c_1 to get

$$c_2 = \overline{a_{1.52}} b_{1.52} a_{1.51} + c_1 \overline{a_{1.52}} b_{1.52} \qquad (30.70)$$

We can write the general form of b_2 through b_{N-1}, using full adders, as

$b_{N-1} = \overline{a_{1.5N-1}} b_{1.5N-1} \oplus a_{1.5N-2} \oplus c_{N-2}$ and

$$c_{N-1} = \overline{a_{1.5N-1}} b_{1.5N-1} a_{1.5N-2} + c_{N-2}(\overline{a_{1.5N-1}} b_{1.5N-1} + a_{1.5N-2}) \qquad (30.71)$$

The bit b_N has a weighting of $2V_{CM}$ and thus the final output word size is $N + 1$

$$b_N = a_{1.5N-1} \oplus c_{N-1} \qquad (30.72)$$

Before we sketch the implementation of this digital circuit, let's make a few comments. To begin, notice that the word size is one larger than N (the resolution). In the 1 bit/stage circuit our maximum output is (assuming $N = 8$)

$$1111\ 1111 = VDD - 1\ \text{LSB} = 2V_{CM} - 1\ \text{LSB}\ \text{(1 bit/stage)}$$

Now our maximum output is

$$1\ 0000\ 0000 = VDD = 2V_{CM}\ \text{(1.5 bit/stage)}$$

The resolution is still essentially eight bits; we just have a slightly larger (1 LSB) output range. To make the words exactly match, we can throw out the MSB and assume 1 0000 0000 is an out-of-range condition. Note that an input of VDD – 1 LSB using 1.5 bits/stage gives an output code of 0 1111 1111.

One more comment: if we go through the logic in Eq. (30.66), we don't get the correct outputs unless we subtract V_{CM} – 0.5 LSB. The binary offset representation can be written as

$$V_{CM} - 0.5\ \text{LSB} = 0011111111... \qquad (30.73)$$

For example, applying V_{CM} (single-ended, see Fig. 30.51) to the ADC input results in a continuous output (ab) of 01. The output prior to subtraction, from Eqs. (30.67) to (30.72), is then 01111111... (b_N = 0, b_{N-1} = 1, b_{N-2} = 1, etc.). After subtracting 0011111111..., we get 01000000 or V_{CM} knowing the weighting of the second bit is V_{CM}.

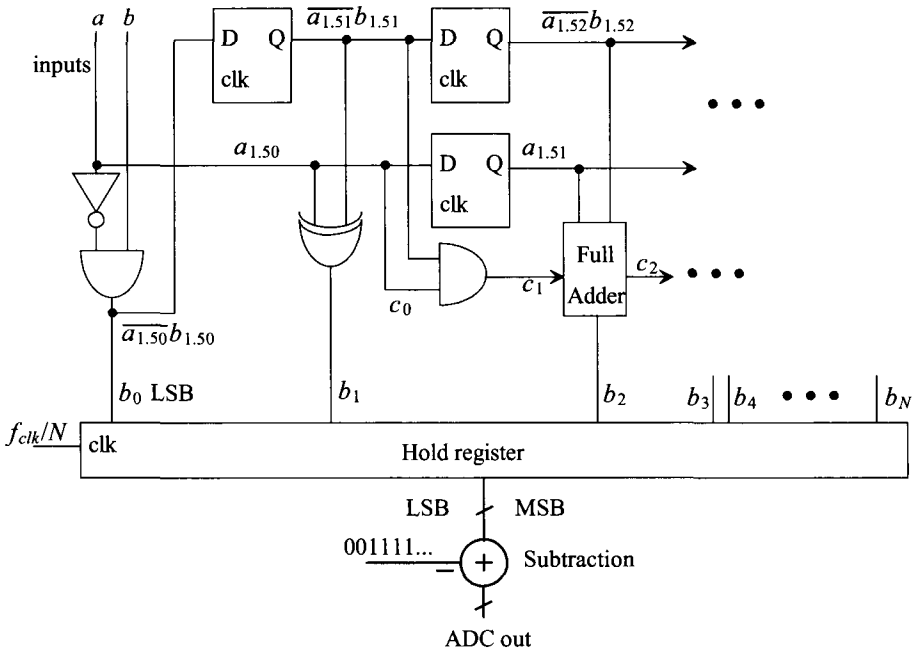


Figure 30.53 Combining the outputs of the cyclic ADC when 1.5 bits/stage is used.

Figure 30.53 shows one possible implementation of Eqs. (30.67)-(30.72) for a cyclic ADC. The state shown, i.e., the *ab* values, is valid at the end of the conversion (after N clock cycles). When starting the algorithm (on the first rising edge of clock), all latches are reset to zeroes and the first comparator outputs, *ab*, are applied. On the second rising edge of clock the output b_2 corresponds to the final b_N. After N clock cycles, the hold register is clocked (and the latches are reset). The final ADC output is the contents of the hold register after subtracting 00111111.... Changing the numbers to two's complement may be useful when implementing this stage (assuming the MSB has a weighting of V_{CM}, i.e., throw out b_N). Note the subtraction can precede the hold register.

Example 30.16

Repeat Ex. 30.10 if 1.5 bits/stage are used. Assume the converter is ideal and the comparators switch precisely at $V_{CM}/2$ (= 250 mV here) and $3V_{CM}/2$ (= 750 mV here). Assume all latches initially contain zeroes.

v_{in}	$a_{1.5X}b_{1.5X}$	v_{out}	Digital out
600 mV ($N-1=7$)	01	700 mV	$b_7 = \overline{a_{1.57}}b_{1.57} \oplus a_{1.56} \oplus c_6 = 0$
$c_7 = \overline{a_{1.57}}b_{1.57}a_{1.56} + c_6(\overline{a_{1.57}}b_{1.57} + a_{1.56}) = 1$			
700 mV ($N-2=6$)	01	900 mV	$b_6 = \overline{a_{1.56}}b_{1.56} \oplus a_{1.55} \oplus c_5 = 0$
$c_6 = \overline{a_{1.56}}b_{1.56}a_{1.55} + c_5(\overline{a_{1.56}}b_{1.56} + a_{1.55}) = 1$			

900 mV ($N-3=5$) 11 300 mV $b_5 = \overline{a_{1.55}} b_{1.55} \oplus a_{1.54} \oplus c_4 = 0$

$c_5 = \overline{a_{1.55}} b_{1.55} a_{1.54} + c_4(\overline{a_{1.55}} b_{1.55} + a_{1.54}) = 0$

300 mV ($N-4=4$) 01 100 mV $b_4 = \overline{a_{1.54}} b_{1.54} \oplus a_{1.53} \oplus c_3 = 1$

$c_4 = \overline{a_{1.54}} b_{1.54} a_{1.53} + c_3(\overline{a_{1.54}} b_{1.54} + a_{1.53}) = 0$

100 mV ($N-4=3$) 00 700 mV $b_3 = \overline{a_{1.53}} b_{1.53} \oplus a_{1.52} \oplus c_2 = 1$

$c_3 = \overline{a_{1.53}} b_{1.53} a_{1.52} + c_2(\overline{a_{1.53}} b_{1.53} + a_{1.52}) = 0$

700 mV ($N-6=2$) 01 900 mV $b_2 = \overline{a_{1.52}} b_{1.52} \oplus a_{1.51} \oplus c_1 = 0$

$c_2 = \overline{a_{1.52}} b_{1.52} a_{1.51} + c_1(\overline{a_{1.52}} b_{1.52} + a_{1.51}) = 1$

900 mV ($N-7=1$) 11 300 mV $b_1 = \overline{a_{1.51}} b_{1.51} \oplus c_0 = 0$

$c_1 = \overline{a_{1.51}} b_{1.51} a_{1.50} + c_0(\overline{a_{1.51}} b_{1.51} + a_{1.50}) = 0$

300 mV ($N-8=0$) 01 100 mV $b_0 = \overline{a_{1.50}} b_{1.50} = 1$

$c_0 = a_{1.50} = 0$

noting that $b_8 = a_{1.57} \oplus c_7 = 1$. We can reorder the bits so the MSB is on the left, the LSB is on the right, yielding 1 0001 1001 and subtract 0 0111 1111 yielding

$$\begin{array}{r} 1\ 0001\ 1001\ (281) \\ -\ \underline{0\ 0111\ 1111}\ (127) \\ 0\ 1001\ 1010\ (154) \end{array}$$

This is the result given in Ex. 30.10 (1001 1001, or decimal 153) plus 1 LSB. The 1 LSB discrepancy can be traced to Eq. (30.66) where we used 0.5 LSBs. Because our resolution is at best 1 LSB, sometimes the result will experience a round-off error. To understand this in the subtraction above, the more correct decimal representation of V_{CM} – 0.5 LSBs is 127.5 and the more correct decimal output is 153.5. ■

Example 30.17

Repeat Ex. 30.16 if the comparators switch at 305 mV (a 55 mV offset) and 675 mV (a –75 mV offset).

v_{in}	$a_{1.5X} b_{1.5X}$	v_{out}	Digital out
600 mV ($N-1=7$)	01	700 mV	$b_7 = \overline{a_{1.57}} b_{1.57} \oplus a_{1.56} \oplus c_6 = 0$

$c_7 = \overline{a_{1.57}} b_{1.57} a_{1.56} + c_6(\overline{a_{1.57}} b_{1.57} + a_{1.56}) = 1$

700 mV ($N-2=6$) 11 –100 mV $b_6 = \overline{a_{1.56}} b_{1.56} \oplus a_{1.55} \oplus c_5 = 0$

$c_6 = \overline{a_{1.56}} b_{1.56} a_{1.55} + c_5(\overline{a_{1.56}} b_{1.56} + a_{1.55}) = 0$

–100 mV ($N-3=5$) 00 300 mV $b_5 = \overline{a_{1.55}} b_{1.55} \oplus a_{1.54} \oplus c_4 = 0$

$c_5 = \overline{a_{1.55}} b_{1.55} a_{1.54} + c_4(\overline{a_{1.55}} b_{1.55} + a_{1.54}) = 0$

300 mV ($N-4=4$) 00 1.1 V $b_4 = \overline{a_{1.54}} b_{1.54} \oplus a_{1.53} \oplus c_3 = 1$

$c_4 = \overline{a_{1.54}} b_{1.54} a_{1.53} + c_3(\overline{a_{1.54}} b_{1.54} + a_{1.53}) = 0$

1100 mV ($N-4=3$) 11 700 mV $b_3 = \overline{a_{1.53}} b_{1.53} \oplus a_{1.52} \oplus c_2 = 1$

$c_3 = \overline{a_{1.53}} b_{1.53} a_{1.52} + c_2(\overline{a_{1.53}} b_{1.53} + a_{1.52}) = 0$

700 mV ($N-6=2$) 11 −100 mV $b_2 = \overline{a_{1.52}} b_{1.52} \oplus a_{1.51} \oplus c_1 = 0$

$c_2 = \overline{a_{1.52}} b_{1.52} a_{1.51} + c_1(\overline{a_{1.52}} b_{1.52} + a_{1.51}) = 0$

−100 mV ($N-7=1$) 00 300 mV $b_1 = \overline{a_{1.51}} b_{1.51} \oplus c_0 = 0$

$c_1 = \overline{a_{1.51}} b_{1.51} a_{1.50} + c_0(\overline{a_{1.51}} b_{1.51} + a_{1.50}) = 0$

300 mV ($N-8=0$) 00 1.1V $b_0 = \overline{a_{1.50}} b_{1.50} = 0$

$c_0 = a_{1.50} = 0$

and $b_8 = a_{1.57} \oplus c_7 = 1$. Again, we can reorder the bits so the MSB is on the left, the LSB is on the right, yielding 1 0001 1000 and subtract 0 0111 1111 yielding

```
  1 0001 1000 (280)
- 0 0111 1111 (127)
  0 1001 1001 (153)
```

This is the exact result given in Ex. 30.10 (1001 1001, or decimal 153). In this case the 0.5 LSB round-off worked in our favor.

While the comparator performance can be extremely poor, the circuit of Fig. 30.42 must subtract and amplify to an accuracy set by the least significant bit of the converter. When we calculated values in Ex. 30.16 and here in this example we assumed subtractions of exactly 0, V_{CM}, and $2V_{CM}$ followed by a multiplication of exactly two.

Finally, notice that the negative output of −100 mV (single-ended) and the 1.1 V output that is greater than *VDD* (= 1 V here) are easily accommodated when using fully-differential op-amps. ■

Capacitor Error Averaging

While using 1.5 bits/stage has made the comparator offset unimportant, we still have to amplify the signal by a precise factor of 2. Toward the goal of precision gain consider the redrawn version of our basic S/H, Fig. 30.42, shown in Fig. 30.54. In this figure we've shown the clock phases (but not the slightly delayed clock signals also used) and the capacitors with mismatch. Ideally, $\Delta C_{+,-} = 0$ and the gain of the S/H is precisely 2 (assuming sufficiently high op-amp open-loop gain, see Eq. [30.38]). From Eq. (30.52) we can write

$$v_{out+} = \left(2 + \frac{\Delta C_+}{C_+}\right) \cdot v_{in+} - \left(1 + \frac{\Delta C_+}{C_+}\right) V_{CI+} \qquad (30.74)$$

where $\Delta C/C$ is the capacitor mismatch (say, ±1% or ±0.01, noting ΔC may be positive or negative). For the negative output

$$v_{out-} = \left(2 + \frac{\Delta C_-}{C_-}\right) \cdot v_{in-} - \left(1 + \frac{\Delta C_-}{C_-}\right) V_{CI-} \qquad (30.75)$$

where $v_{out} = v_{out+} - v_{out-}$. Note that when using this topology in a pipeline converter one stage can be in the hold state while the next stage can be in the sampling state effectively

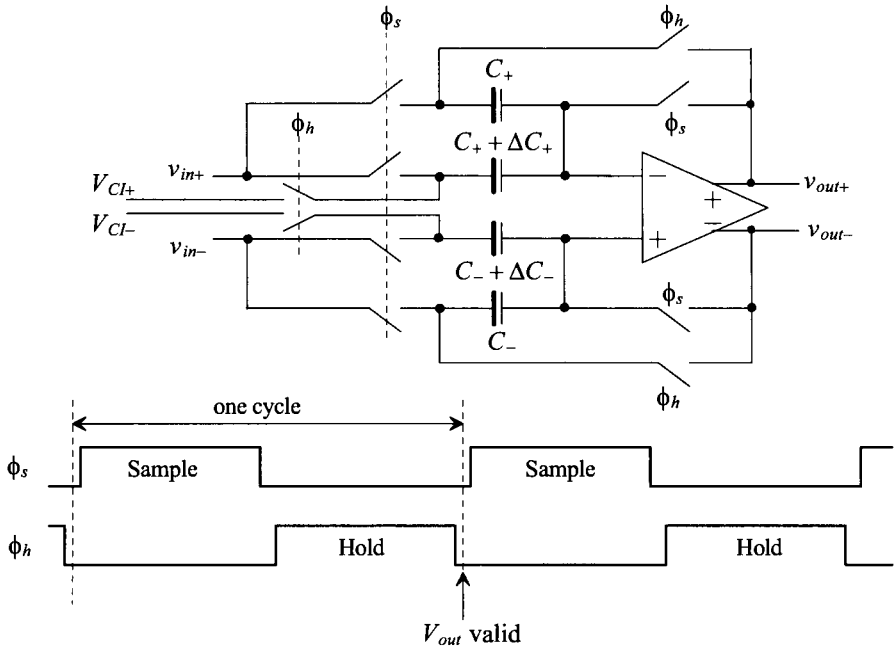


Figure 30.54 S/H of Fig. 30.42 with mismatched capacitors.

sharing the time. The result of this sharing is the need for only one clock cycle for each stage in the conversion.

Next consider what happens if, instead of sampling the input voltage again, we simply switch the positions of the C and $C+\Delta C$ capacitors, that is, we connect C to V_{CI} and $C+\Delta C$ to V_{out}. Figure 30.55 shows this switch and the modified S/H using an extra half-clock cycle. The sample and amplify phases of the clock are exactly the same as before, and so Eqs. (30.74) and (30.75) are still valid. We denote the outputs at the end (falling edge) of the amplify phase as v_{outa+} and v_{outa-}

$$v_{outa+} = 2 \cdot v_{in+} - V_{CI+} + \frac{\Delta C_+}{C_+} \cdot (v_{in+} - V_{CI+}) \qquad (30.76)$$

and

$$v_{outa-} = 2 \cdot v_{in-} - V_{CI-} + \frac{\Delta C_-}{C_-} \cdot (v_{in-} - V_{CI-}) \qquad (30.77)$$

where the ideal situation is $\Delta C = 0$.

At the end of the amplify phase the charge on the capacitors is (assuming the op-amp input voltages are 0 and only looking at the + path to simplify the equations)

$$Q_{a+} = (C_+ + \Delta C_+) \cdot (V_{CI+}) + C_+ \cdot (v_{outa+}) \qquad (30.78)$$

and at the end of the hold phase

$$Q_{h+} = (C_+ + \Delta C_+) \cdot (v_{outh+}) + C_+ \cdot (V_{CI+}) \qquad (30.79)$$

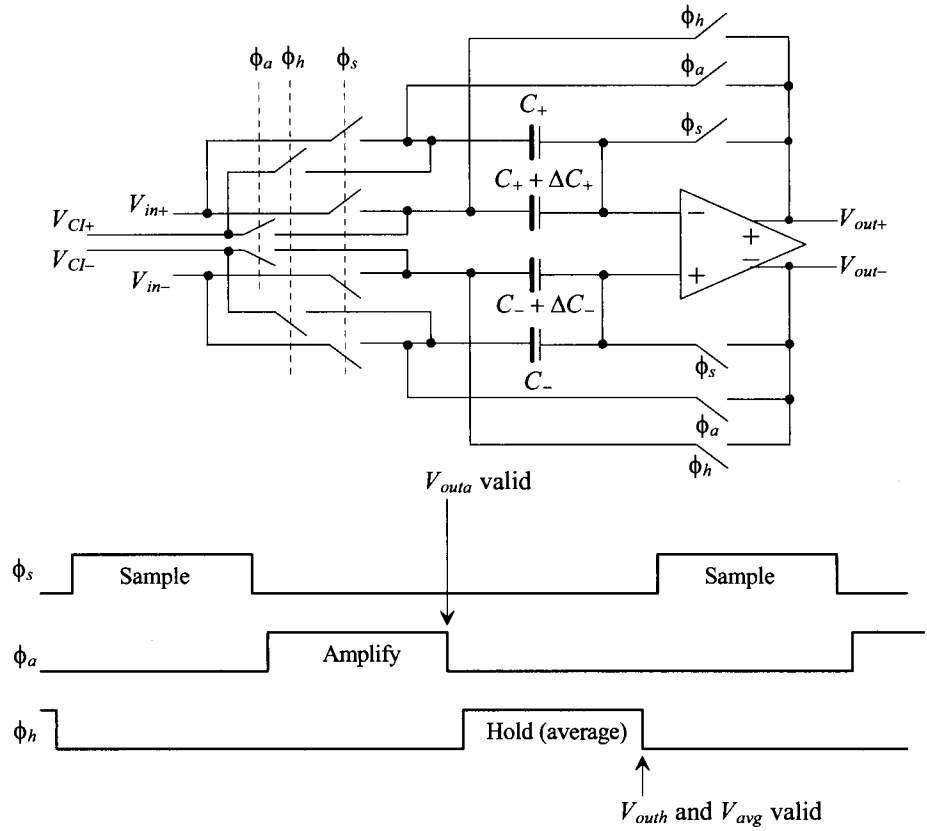


Figure 30.55 S/H using capacitor error averaging.

Because charge must be conserved, $Q_{a+} = Q_{h+}$, and therefore

$$v_{outh+} = V_{CI+} + \frac{C_+}{C_+ + \Delta C_+} \cdot v_{outa+} - \frac{C_+}{C_+ + \Delta C_+} \cdot V_{CI+} \tag{30.80}$$

It will be useful to use

$$\frac{C_+}{C_+ + \Delta C_+} = \frac{1}{1 + \Delta C_+ / C_+} \approx 1 - \frac{\Delta C_+}{C_+} \tag{30.81}$$

Rewriting Eq. (30.80) gives

$$v_{outh+} = \left(1 - \frac{\Delta C_+}{C_+}\right) \cdot v_{outa+} + \frac{\Delta C_+}{C_+} \cdot V_{CI+} \tag{30.82}$$

Substituting Eq. (30.76) for v_{outa+} results in

$$v_{outh+} = 2v_{in+} - V_{CI+} - \frac{\Delta C_+}{C_+} \cdot (v_{in+} - V_{CI+}) \tag{30.83}$$

where the terms containing $(\Delta C_+/C_+)^2$ are assumed negligible. If the matching is 1%, then $(\Delta C_+/C_+)^2 = 10^{-4}$.

Clearly averaging v_{outa+} (Eq. [30.76]) and v_{outh+} (Eq. [30.83]) results in the precise desired gain. The question now becomes: "How do we perform the averaging without introducing more error?" Ideally we want

$$\begin{aligned} v_{avg+} &= \frac{v_{outa+} + v_{outh+}}{2} \\ &= \frac{1}{2} \cdot \left(2v_{in+} - V_{CI+} + \frac{\Delta C_+}{C_+} \cdot (v_{in+} - V_{CI+}) + 2v_{in+} - V_{CI+} - \frac{\Delta C_+}{C_+} \cdot (v_{in+} - V_{CI+}) \right) \\ &= 2v_{in+} - V_{CI+} \end{aligned} \tag{30.84}$$

or more generally

$$v_{avg} = v_{avg+} - v_{avg-} = 2(v_{in+} - v_{in-}) - (V_{CI+} - V_{CI-}) \tag{30.85}$$

This equation should be compared to Eq. (30.61).

Consider the averaging amplifier shown in Fig. 30.56. We can write the charge on the four capacitors when the ϕ_a switches are closed (actually at the falling edge of the ϕ_a clock, assuming complete settling) as

$$Q_+^{\phi_a} = (v_{outa+} - V_{CM} \pm V_{OS}) \cdot C_F + (v_{outa-} - V_{CM} \pm V_{OS}) \cdot C_I \tag{30.86}$$

and

$$Q_-^{\phi_a} = (v_{outa-} - V_{CM} \pm V_{OS}) \cdot C_F + (v_{outa+} - V_{CM} \pm V_{OS}) \cdot C_I \tag{30.87}$$

Note that the common-mode voltage and op-amp offset subtract out when we take the difference between the balanced signals and so we do not include V_{CM} or V_{OS} in the remaining analysis. (The offset from the common-mode feedback circuit results in an unharmful variation in V_{CM} and is discussed later in this section.) On the falling edge of ϕ_h, and following the same procedure as used in Eq. (30.48), we can write

$$C_F \cdot v_{avg+} = C_F \cdot v_{outa+} + C_I \cdot v_{outa-} - C_I \cdot v_{outh-} \tag{30.88}$$

and

$$C_F \cdot v_{avg-} = C_F \cdot v_{outa-} + C_I \cdot v_{outa+} - C_I \cdot v_{outh+} \tag{30.89}$$

We can then write

$$v_{avg+} = v_{outa+} + (v_{outa-} - v_{outh-}) \cdot \frac{C_I}{C_F} \tag{30.90}$$

$$v_{avg-} = v_{outa-} + (v_{outa+} - v_{outh+}) \cdot \frac{C_I}{C_F} \tag{30.91}$$

noting that if $v_{outa-} = v_{outh-}$ and $v_{outa+} = v_{outh+}$, the matching of the capacitors in Fig. 30.55 is perfect and $v_{avg+} - v_{avg-} = v_{outa+} - v_{outa-}$ (the output perfectly follows Eq. [30.61]). Taking the difference of these two equations results in

$$v_{avg+} - v_{avg-} = v_{outa+} - v_{outa-} - \overbrace{\frac{(v_{outa+} - v_{outh+}) - (v_{outa-} - v_{outh-})}{2}}^{\text{Error adjustment term}} \tag{30.92}$$

If we substitute Eqs. (30.76), (30.77), and (30.83) into this equation, we get

$$v_{avg+} - v_{avg-} = 2(v_{in+} - v_{in-}) - (V_{CI+} - V_{CI-}) + (v_{in+} - V_{CI+}) \cdot \frac{\Delta C_+}{C_+} - (v_{in-} - V_{CI-}) \cdot \frac{\Delta C_-}{C_-} - \frac{2(v_{in+} - V_{CI+})}{C_F/C_I} \cdot \frac{\Delta C_+}{C_+} + \frac{2(v_{in-} - V_{CI-})}{C_F/C_I} \cdot \frac{\Delta C_-}{C_-} \quad (30.93)$$

which reduces to Eq. (30.85) or (30.61) assuming C_I/C_F is 1/2 (C_F is twice as large as C_I). Note how the selection (and matching) of the capacitor ratios, C_F/C_I, is not that important. The capacitors do not have to match to the final ADC resolution. A matching of 1% will result in an error term that is one-hundreth of the error that would be present if the error averaging were not used. Also note, as indicated at the beginning of the section, that only the first stages (say the first five in a 14-bit ADC) need use capacitor error averaging because of the reduced accuracy requirements placed on the later stages as we move through the converter.

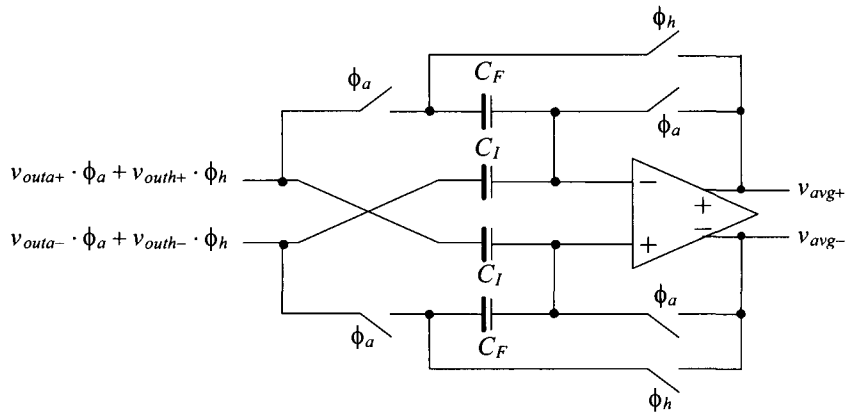


Figure 30.56 Averaging amplifier for use on the output of Fig. 30.55.

The penalty for this precision technique is the increase in conversion time used in each stage (and increased noise because two op-amps are used in each 1.5-bit section). As seen in Fig. 30.55, an extra half-clock cycle is required. For 20 Msamples/s a 30 MHz clock would be required. Again while one stage is in the hold (average) state, the next stage in the pipeline can be in the sample state so that the conversion time is shared (but still requiring 1.5 clock cycles). Using the capacitor error averaging technique for precision gain (and subtraction) and using 1.5-bit/stage sections, a 14-bit ADC, using fully-differential input signals, has been attained at 20 Msamples/s without trimming or calibration [5].

Example 30.18

Simulate the operation of the circuit shown in Fig. 30.57. Comment on the ideal outputs and the simulation results.

The capacitor values were chosen arbitrarily. The input voltage, v_{in}, is 0.75 – 0.25 or 0.5 V. The ideal output voltage, v_{out}, is then 1 V. v_{out+} should ideally be 1 V, and v_{out-} should ideally be 0 V. The simulation results are shown in Fig. 30.58. The

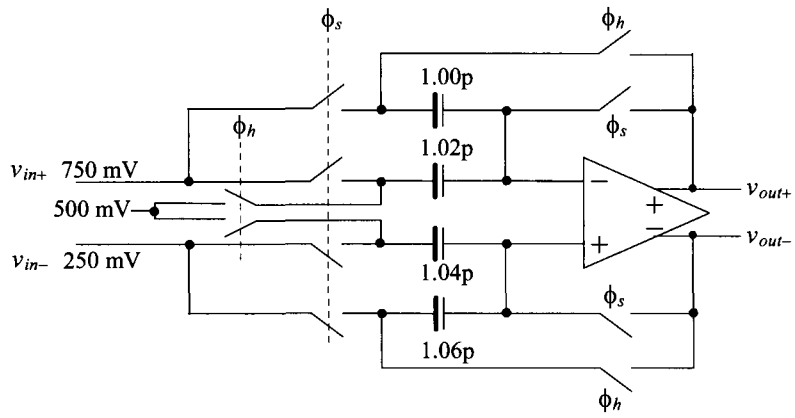


Figure 30.57 Matching errors in capacitors (see Ex. 30.18).

error plotted in this figure is the result of taking the difference between the ideal output and the actual output. The error for the capacitor values shown in Fig. 30.57 is approximately 5 mV. Clearly, at the risk of stating the obvious, capacitor mismatch is a fundamental limitation to ADC accuracy. ■

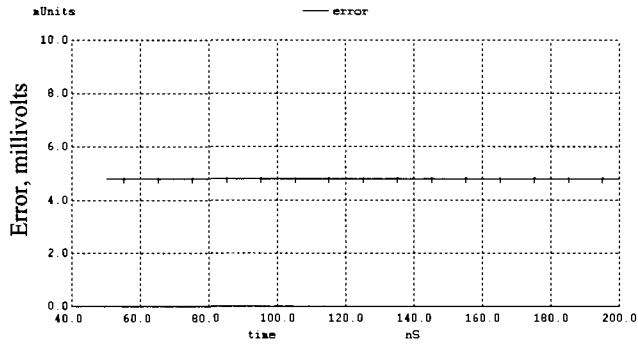


Figure 30.58 Simulation results from Ex. 30.18 showing amplification error.

Example 30.19

Simulate the operation of the circuit shown in Fig. 30.59 (a cascade of Figs. 30.55 and 30.56). Comment on the ideal outputs and the simulation results.

This circuit shows the same mismatched capacitors in the multiply-by-two stage as we saw in Ex. 30.18. The averaging stage also shows mismatched capacitors with arbitrary values. Again, as in Ex. 30.18, the ideal output voltage is 1.0 V (v_{out+} = 1 V and v_{out-} = 0 V). The simulation results are shown in Fig. 30.60. The error has dropped from 5 mV to – 67 μV. It may be instructive to resimulate the capacitor error averaging topology of Fig. 30.59 using different values of capacitors in order to get a feeling for just how forgiving the topology is to mismatches.

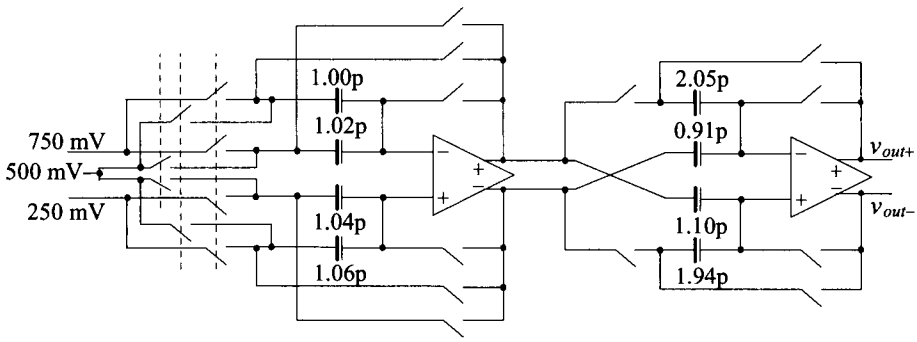


Figure 30.59 Implementation of error averaging (see Ex. 30.19).

Note that in the simulation netlist, where we are using near ideal op-amps with open loop gains of 100 million, we added switches in series with the C_l capacitors in the averaging circuit to avoid the situation of the op-amp operating open-loop with its outputs going to millions of volts when both ϕ_a and ϕ_h are low. (The op-amp operates open loop when ϕ_s is high, which is usually not a problem in a practical circuit). ■

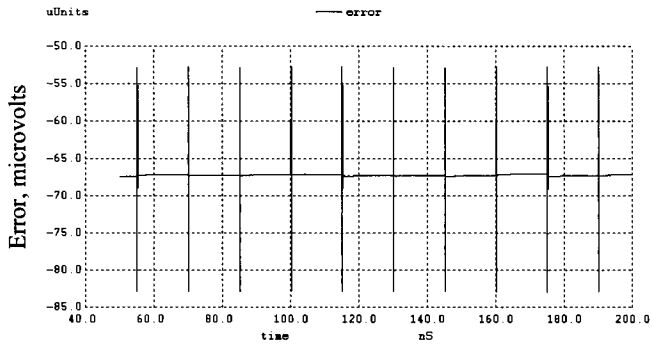


Figure 30.60 Simulation results from Ex. 30.19 showing amplification error.

Example 30.20

Repeat Ex. 30.19 if the op-amps used have open-loop gains of 1,000.

The simulation results are shown in Fig. 30.61. The error has increased by a factor greater than 10. As indicated earlier in Eq. (30.39), the minimum op-amp open-loop gain is set by the resolution of the data converter. If we were designing a 14-bit ADC, the op-amp used would require open-loop gains greater than 2^{16} or 64k (96 dB). ■

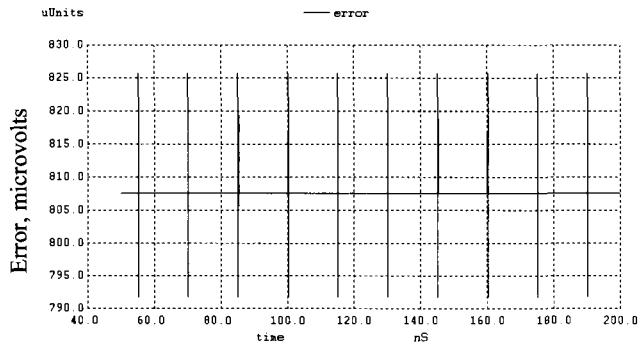


Figure 30.61 Regeneration of Fig. 30.60 using op-amps with open-loop gains of 1,000.

Comparator Placement

The implementation of a pipeline ADC showing how the clock signals are used in each error averaging stage is shown in Fig. 30.62. The important thing to note is the use of only three phases of an input clock signal. The connection of the clock phases to the switches changes in each stage allowing a stage to sample an input signal while the previous stage is in the hold mode. We'll discuss the generation of these clock signals in the next section; here we discuss comparator placement and performance.

As seen in Fig. 30.55 the voltages V_{CI+} and V_{CI-} must be valid and stable when the amplify-and-hold clock phases are high. Reviewing Fig. 30.51 we can implement the two comparators using a clock signal followed by a latch, Fig. 30.63. The comparators can be clocked on the rising edge of the amplifying clock (which is a different clock phase in each of the three possible clocking schemes). It's important to place a separate clocked latch on the outputs of each comparator (*clocked with a slightly delayed clock signal which isn't shown in the figure*) to ensure that comparator metastability isn't a factor when generating the MUX addressing (select inputs). We can get away with this type of clocking scheme because we are using three levels/stage (1.5 bits/stage). The comparators don't have to be N-bit accurate, as discussed earlier, but can withstand offsets/errors approaching $V_{CM}/4$. The signals V_{CI+} and V_{CI-} do have to be N-bit accurate, though, and must settle and stay settled to this accuracy by the end of the amplify phase.

The two bits coming out of the comparators (actually the latches connected to the comparator outputs) can be thought of as the first delay element shown in Fig. 30.49 (except now it is a 2-bit delay element). Because each of these first delays are clocked on the rising edge of one phase of the clock signal, we have a problem with synchronizing the bits together prior to application to the digital correction logic (similar to Fig. 30.53). Reviewing the clock signals in Fig. 30.62, we see that if we clock all other delay elements in the pipeline of Fig. 30.49 on the rising edge of "amplify, one" we can synchronize all of the digital data together.

Clock Generation

Generating the nonoverlapping clock signals used in, for example, Fig. 30.55 can be accomplished by using the basic circuit shown in Fig. 30.64. To understand the operation of this circuit, note that when one of the input signals goes low the corresponding output

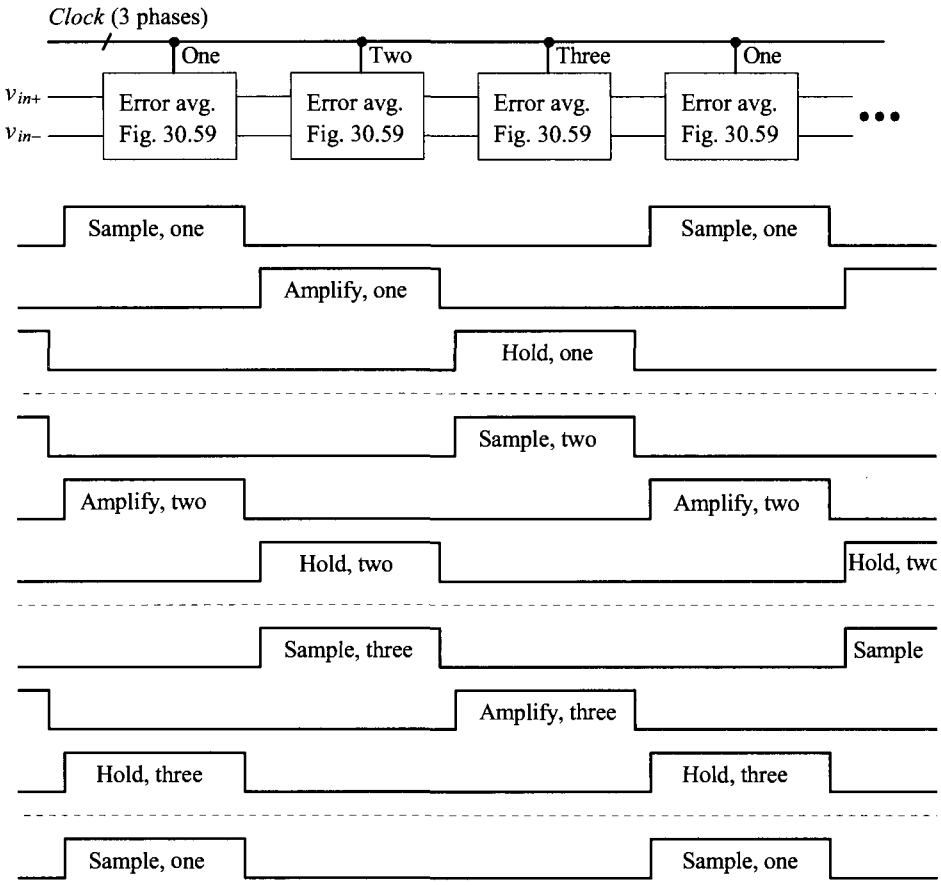


Figure 30.62 Implementation of a pipeline ADC using error averaging (Fig. 30.59).

phase goes high. The feedback ensures that the output doesn't go high until the previous phase goes back low (both inputs of the NOR gate must be low before its output goes high). The amount of nonoverlap time, again, is set by the delay in series with the output of the NOR gates.

The input signals (overlapping clock signals) are generated with the circuit shown in Fig. 30.65. The outputs of this circuit change states on the rising edge of the input clock signal, *Clkin*. After a start-up transient time of a couple of clock cycles, the feedback through the NAND gate to the top D-Flip-Flop ensures that only one bit of In1, In2, or In3 is low any given time. Again, the output of this circuit is then used to drive the nonoverlapping clock circuit seen in Fig. 30.64. While we show 3 signals here it's straightforward to expand the number of clock signals. Note that this circuit is essentially a 3-bit shift register. Also note that changing the NAND gate to a NOR gate or simply inverting the In1 - In3 outputs results in a signal that can be used to drive the nonoverlapping clock generator seen back in Fig. 14.19.

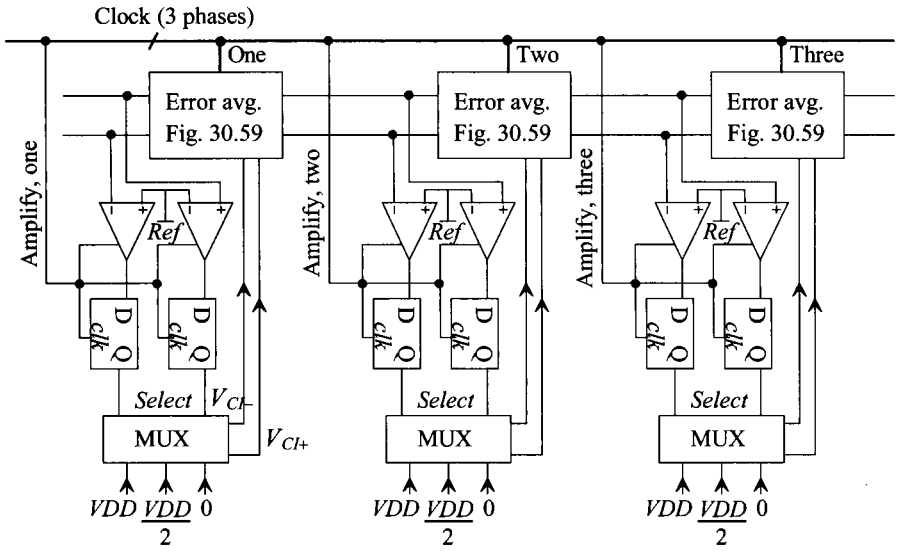


Figure 30.63 Placement of comparators in a pipeline ADC.

Offsets and Alternative Design Topologies

When discussing offsets, we've concentrated on the op-amp's offset. The offset associated with the common-mode voltage hasn't been discussed in detail. What happens if the output signals are moving around a voltage of $V_{CM} + V_{OS}$? Taking the difference in the output signals will eliminate both the common-mode voltage and the offset. A problem

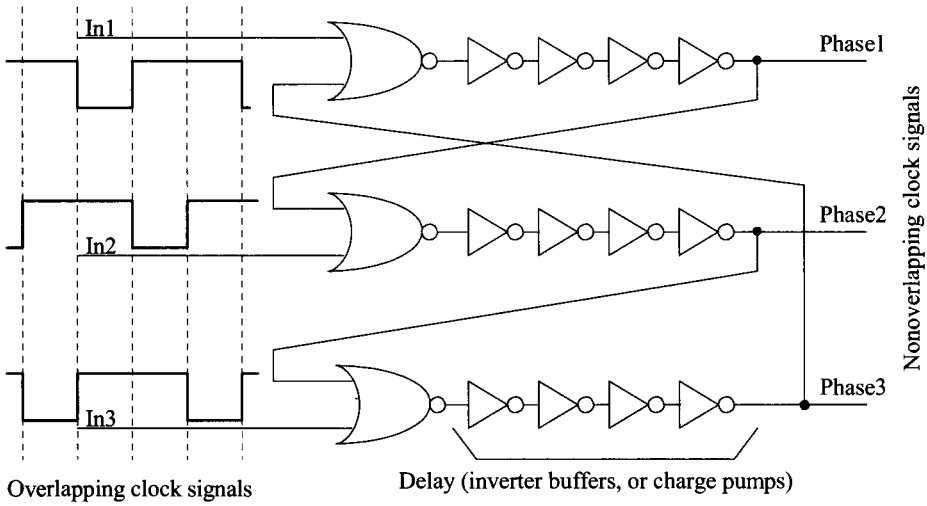


Figure 30.64 Circuit used for generating three phases of a nonoverlapping clock.

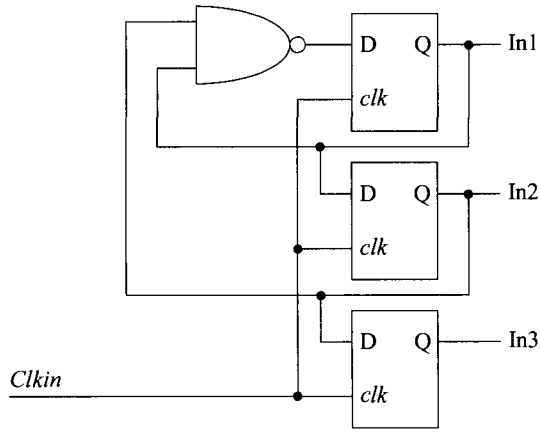


Figure 30.65 Generating the overlapping input clock signals for Fig. 30.64.

could occur if the CMFB circuit doesn't affect each output the same. (The result is a difference-mode signal.)

Example 30.21

Simulate the operation of the circuit shown in Fig. 30.66. Show the input and output signals as the difference between the two differential input signals. Assume the common-mode voltage coming out of the op-amp is precisely 500 mV.

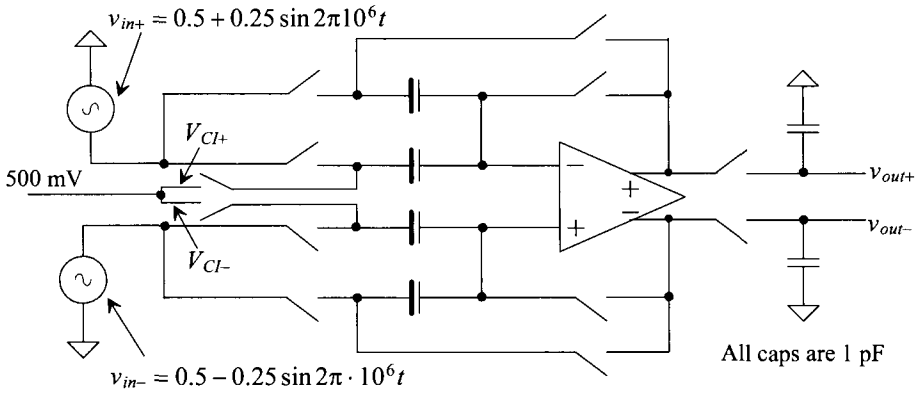


Figure 30.66 ADC building block discussed in Ex. 30.21.

The simulation results are shown in Fig. 30.67. Note how, as we would expect, the gain of the circuit is two. The signals shown are ideal. ■

Example 30.22

Repeat Ex. 30.21 if the common-mode voltage coming out of the op-amp sees an offset of 50 mV, i.e., it is 550 mV.

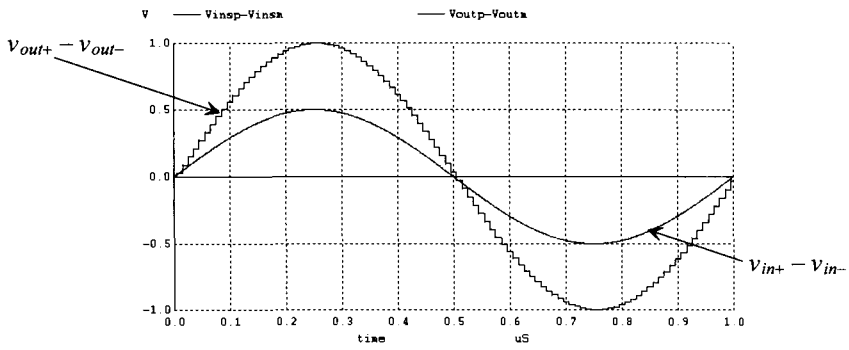


Figure 30.67 Input and output for the circuit of Fig. 30.66.

The differential input and output signals with this output common-mode offset look exactly like what is seen in Fig. 30.67. The single-ended output signals, Fig. 30.68, show the offset (the signals swing around 550 mV in Fig. 30.68). Clearly a common-mode offset in the op-amp output signals isn't a concern (unless it's so large that it limits the op-amp output swing range). Likewise an offset in the common-mode voltage of the input signals isn't a concern (this comment is easy to verify with simulations). What is a concern, though, is the value of the voltages used for V_{CI+} and V_{CI-}. ■

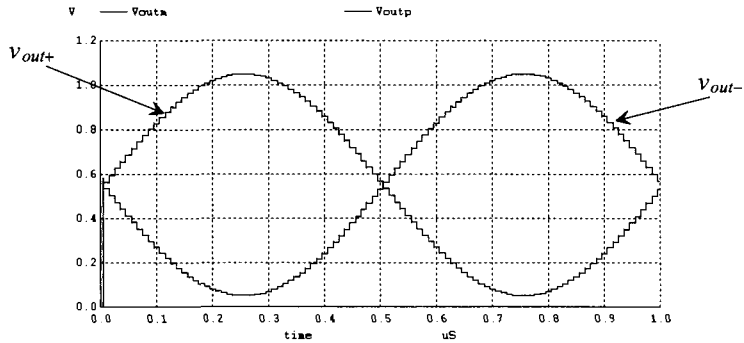


Figure 30.68 Output signals for the circuit of Fig. 30.66 if common-mode voltage is 0.55 V.

In Fig. 30.66 V_{CI+} and V_{CI-} are shorted together and connected, through a switch, to 0.5 V (V_{CM}). It can be shown that when this occurs, the absolute value of the voltage is irrelevant because it is common to both input signal paths, see Eq. (30.53). This means that if we used ground instead of V_{CM} in Fig. 30.66, we would get the same outputs. A problem does occur if the difference in the voltages (when adding or subtracting a voltage from the input signal) isn't exactly what is desired, see Eq. (30.61).

While an offset in the common-mode voltage isn't important, the op-amp's offset is a concern. The op-amp offset voltage is zeroed out when using the topology shown in Fig. 30.66. Equation (30.50) was derived assuming the op-amp had a nonzero offset voltage and shows the offset will not (ideally) affect the building block's output signals. To show the removal of the offset using simulations, consider the following example.

Example 30.23

Simulate the operation of the circuit shown in Fig. 30.69. This schematic is Fig. 30.66 redrawn with a 50 mV op-amp offset.

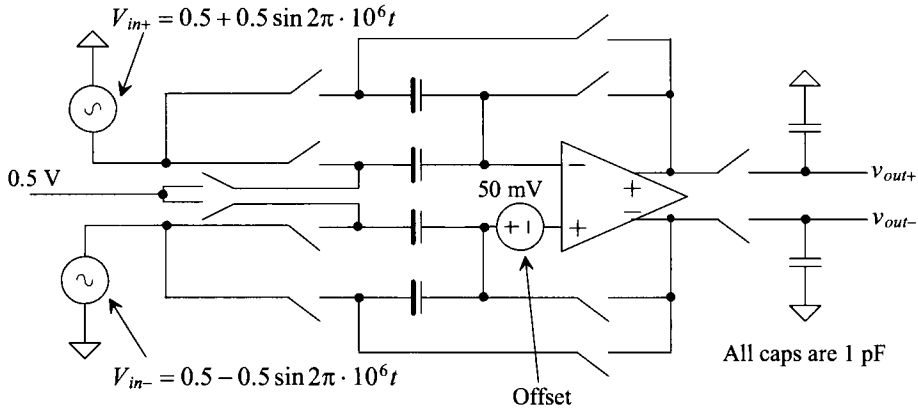


Figure 30.69 ADC building block discussed in Ex. 30.23.

The simulation results are shown in Fig. 30.70. Figure 30.70 should be compared to Fig. 30.67. Note how the large offset has no effect on the building block's output signals. ■

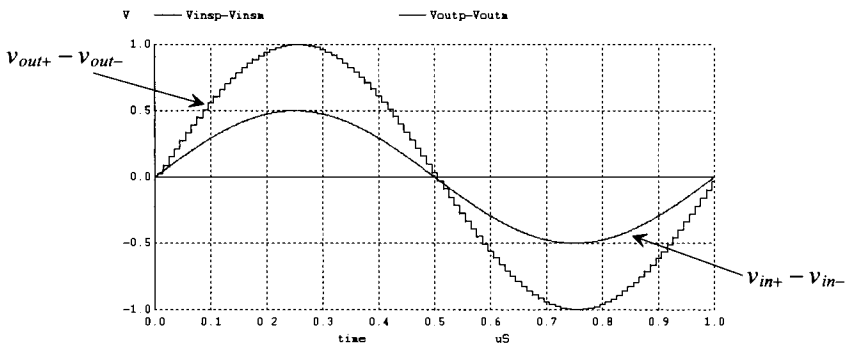


Figure 30.70 Input and output for the circuit of Fig. 30.69 with op-amp offset of 50 mV.

If the op-amp offset is zeroed out and doesn't affect the circuit's outputs when using the basic topology shown in Fig. 30.30, why would we want to consider some other topology? The answer to this question comes from the realization that the op-amp's outputs must settle to the final accuracy of the ADC, referring to Fig. 30.55, by the falling edge of all three phases of the clock. It would be nice to make the settling during one of these three (or two) phases irrelevant. Also, and perhaps more importantly, it would be nice to use other types of CMFB (discussed in the next section).

Consider the building-block topology shown in Fig. 30.71. During the sampling phase, the inputs of the op-amp are shorted to the common-mode voltage. During this time the op-amp operates open loop, and settling time is meaningless. The status of the op-amp outputs during this time is discussed in the next section. The charge stored on the capacitors during the storage phase is

$$Q_{I,F}^{\phi_s} = C_{I,F} \cdot (v_{in} - V_{CM}) \tag{30.94}$$

while during the hold phase

$$Q_I^{\phi_h} = C_I \cdot (V_{CI} - V_{CM} \pm V_{OS}) \text{ and } Q_F^{\phi_h} = C_F \cdot (v_{out} - V_{CM} \pm V_{OS}) \tag{30.95}$$

Once again knowing charge must be conserved, the output voltage can be written as

$$v_{out} = v_{out+} - v_{out-} = \left(1 + \frac{C_I}{C_F}\right) \cdot (v_{in+} - v_{in-}) - \frac{C_I}{C_F} \cdot (V_{CI+} - V_{CI-}) \pm \overbrace{\left(1 + \frac{C_I}{C_F}\right) \cdot V_{OS}}^{\text{unwanted term}} \tag{30.96}$$

This equation should be compared to Eq. (30.53) where the offset was zeroed out during the sample phase of the clock (the phases ϕ_1 and ϕ_2). The topology would be useful in an ADC where we can guarantee that the offset contribution is well below the data converter's LSB or in a later stage in the ADC where the accuracy requirements are relaxed.

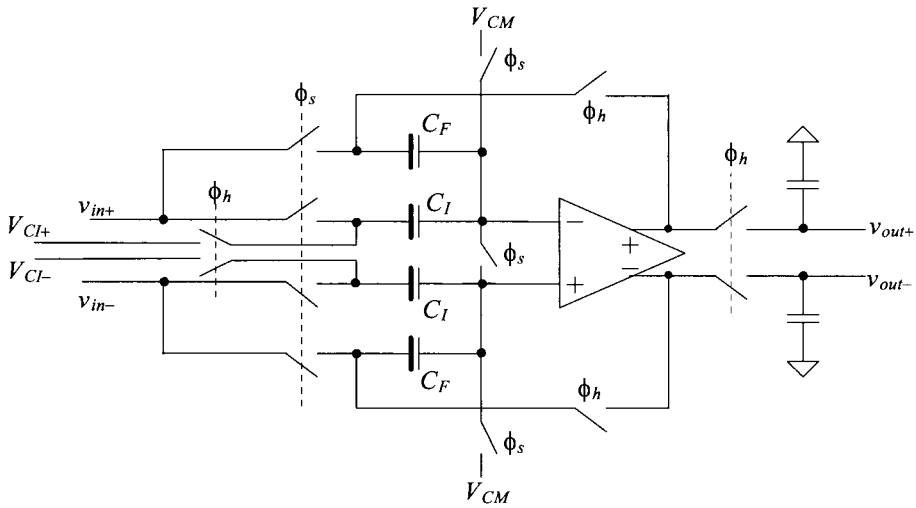


Figure 30.71 Alternative ADC building block.

Example 30.24

Repeat Ex. 30.23 using the topology shown in Fig. 30.71.

The simulation results are shown in Fig. 30.72. The ratio of C_I to C_F is one and so, ideally, the output is just twice as large as the input. The output should swing

from 1 V down to −1 V. However, because of the 50 mV offset, which is also multiplied by a factor of 2, the output is shifted downwards so that it swings from 0.9 V to −1.1 V. ■

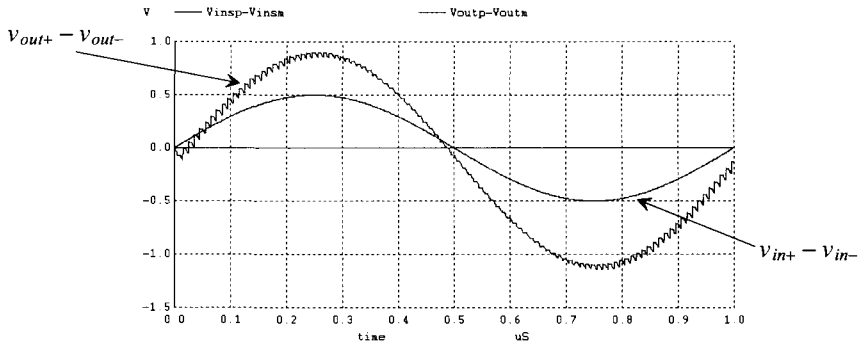


Figure 30.72 Input and output for the circuit of Fig. 30.71 with op-amp offset of 50 mV.

Dynamic CMFB

The CMFB circuits discussed earlier employ an amplifier to sense the average of the outputs and feedback a correction to center the signals around V_{CM}. In Ch. 26 we discussed a CMFB technique used in an op-amp that was dynamic and doesn't employ an amplifier. A simplification of this dynamic CMFB can be realized by noting that during the sampling phase in Fig. 30.71 the op-amp inputs are forced to V_{CM}. The scheme we are about to present won't force the inputs to $V_{CM} \pm V_{OS}$ during the sample phase as in the other topologies based on Fig. 30.30.

The basic dynamic CMFB circuit is shown in Fig. 30.73. During the sample phase of the clock the inputs and outputs of the op-amp are shorted to the common-mode voltage. Also during this time the common-mode feedback voltage, V_{CMFB}, is set to a bias voltage (V_{bias4} if the op-amp of Fig. 30.37 is used). During the hold phase, the CMFB capacitors on the output of the circuit are disconnected from both V_{CM} and V_{bias4} and are used to sense the average value of the outputs. If the outputs move in a balanced fashion, then V_{CMFB} remains equal to V_{bias4}. If the average of the outputs moves upwards above V_{CM}, then V_{CMFB} increases, pulling the output common-mode voltage downwards. Again, because the CMFB loop utilizes negative feedback, an increase in V_{CMFB} must result in a decrease in $(v_{o+} + v_{o-})/2$.

Looking at Fig. 30.73 we see that because of the op-amp's offset voltage the outputs of the op-amp will source/sink a current into V_{CM} during the sample phase of the circuit's operation. By adding an extra pair of switches to disconnect the CMFB capacitors from the op-amp outputs we can avoid this situation. Adding the switches causes the op-amp outputs to approach the power supply rails during the sample phase (because of the offset voltage). This output railing isn't a problem if an OTA (single-stage) topology like the one in Fig. 30.37 is used. The outputs have no capacity to hold charge and so when the CMFB capacitors are reconnected to the op-amp outputs, the outputs immediately go to V_{CM} (neglecting the connection of the feedback capacitors

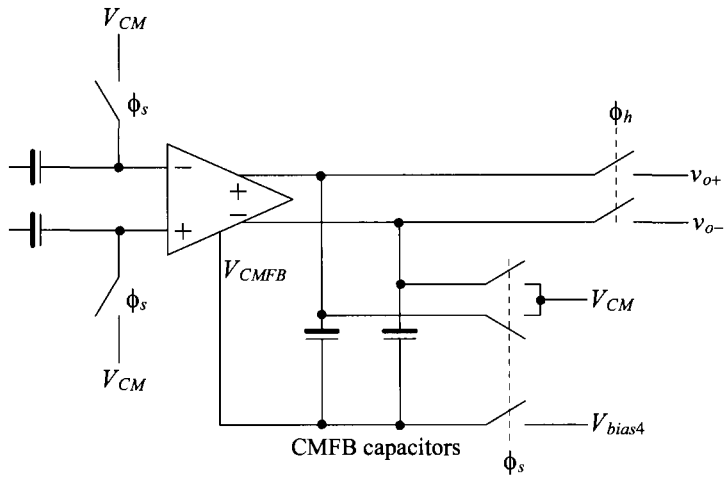


Figure 30.73 Dynamic CMFB.

which are not shown in Fig. 30.73). If a two-stage op-amp is used then care must be taken to ensure that the compensation capacitor (which, of course, does have the capacity to hold charge) doesn't charge to the rail resulting in a large recovery time when the op-amp circuit is put in the hold mode.

One thing that may help with the offset problem (when using the topology of Fig. 30.71) is to AC couple the output of the first op-amp stage to the input of the second op-amp stage, as seen in Fig. 30.74 (add a capacitor between the two stages to remove the DC component). The first stage's offset is stored on the capacitors during the sample phase, while the second stage's inputs are shorted to V_{bias1}. When the op-amp is used in the hold phase of operation, the second stage's offset is referred back to the input of the op-amp after dividing by the first stage's gain. For example, if the first stage of the op-amp has a gain of 10 and the second stage's offset is 10 mV, then the input-referred offset of the op-amp in the hold mode is 1 mV. Note that the gain of the first stage (diff-amp) can't be too large. If the diff-amp's gain were 100 and it had an input-referred offset of 10 mV then the diff-amp's outputs would move towards the supply rails causing devices (one of the PMOS loads and one of the NMOS in the diff-pair) to triode or shut-off. This limits the use of high-impedance loads in the diff-amp. As long as the MOSFETs remain in the saturation region, the offset storage works as desired.

Depending on the load of the diff-amp in Fig. 30.74, the op-amp would be compensated using either a load capacitance (using gate-drain connected loads as seen in the figure so that the topology is an OTA) or using indirect-feedback compensation if a current source load is used in the diff-amp, see Ch. 24. Again, care must be exercised when setting-up the biasing if using a current source load to avoid the PMOS load fighting the diff-amp's tail current for control of the current that flows in the diff-amp.

Layout of Pipelined ADCs

Before leaving this chapter, let's comment on the layout of pipeline ADCs. Using a fixed-height layout structure, with the automatically routed power and ground busses (when the

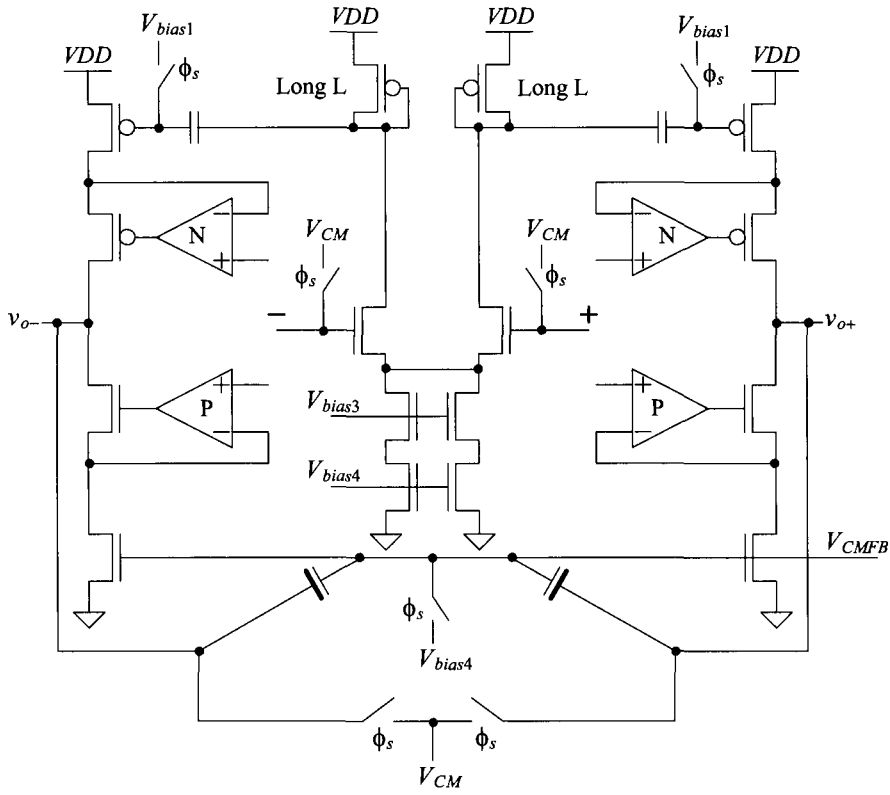


Figure 30.74 Implementation of dynamic CMFB.

cells are laid end-to-end), is a good place to start when laying out the op-amp. As seen in Fig. 30.75, a height (with example values shown) is selected with the width variable. It's important, as discussed in Ch. 28, to keep the analog signals separate, both physically and by distance, from the digital signals.

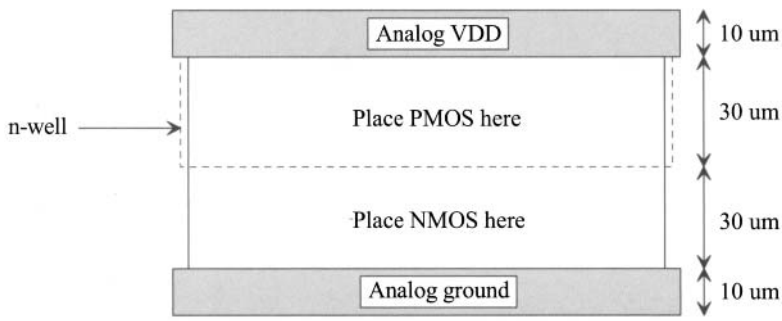


Figure 30.75 Fixed-height layout structure.

A possible block diagram of the placement of the fixed-height cells together with the capacitors and switches used to implement a stage of the ADC is seen in Fig. 30.76. Looking at the input signal, we notice that they are laid out next to each other and routed as close as possible to the input of the first stage of the ADC. All of the differential analog signals in the ADC should follow this practice to help make any coupled noise truly common-mode. Notice how we have placed ground pads adjacent to the input signals. These pads are not used for ground connections on-chip. The pads are used to help reduce noise coupling into the input signals. Ideally, these ground pins provide a termination for the noise keeping the input signals "clean." This is especially important when bonding wires connect the integrated circuit to a padframe in the final packaged part. The bonding wires used for digital signals tend to radiate more than enough signal to corrupt the input signal and ruin the ADC's SNR when placed close to an analog low-level signal.

Next notice that we must have a clock signal in our analog domain. This signal, as we have seen, is used for clocking the switches in the S/H stage. Although in the figure we show the placement of the clock adjacent to the input signals, it may be better to move the pad and, if possible, the routing of the clock signals away from the inputs. The main goal when routing the clock signals is to keep the layout regular. Routing clock signals all over the layout is asking for problems.

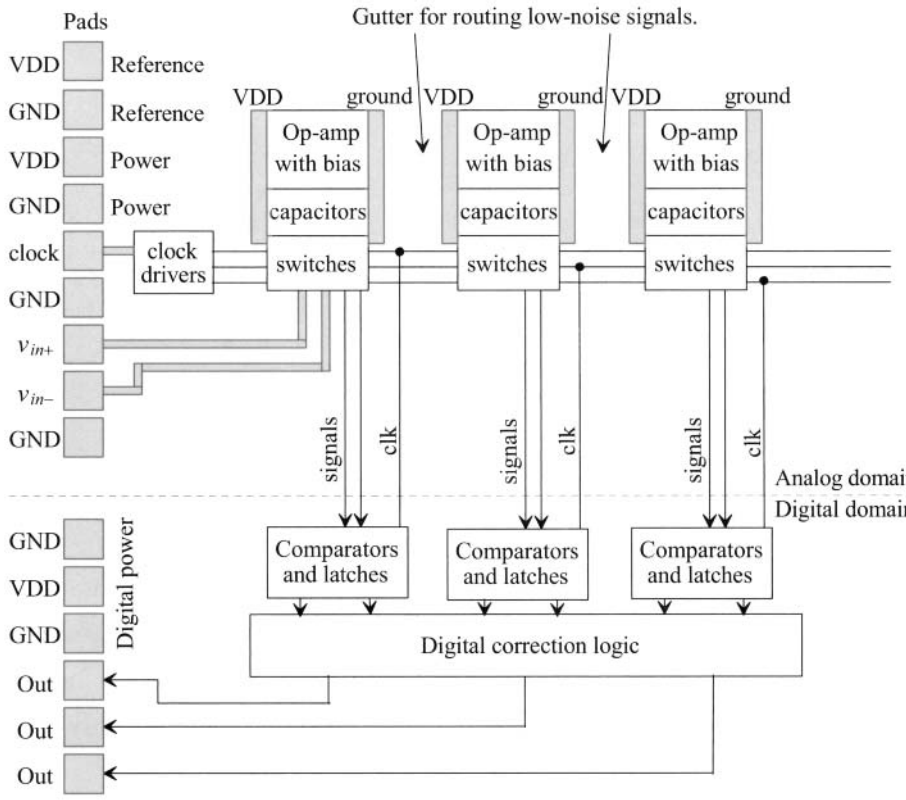


Figure 30.76 Block diagram layout of a pipeline stage.

Two sets of power and ground pads are used (more if possible) in the analog domain for the implementation of the ADC. One set of pads is used for supplying power to the op-amps while the other set is used for supplying the reference voltages to each stage. The power and ground supplies are common to both digital and analog sections off-chip. Off-chip the supplies are connected together and decoupled (bypassed) using large capacitors (actually a wide range of capacitor values are connected in parallel between the *VDD* and ground to avoid the increase in a single large capacitor's effective series resistance with frequency). On-chip decoupling capacitors can be used as well. The analog and digital power and ground connections are not shared, and so care must be taken not to decouple the analog *VDD* to digital ground or digital *VDD* to analog ground. Figure 30.77 shows one example of how the decoupling capacitors can be connected.

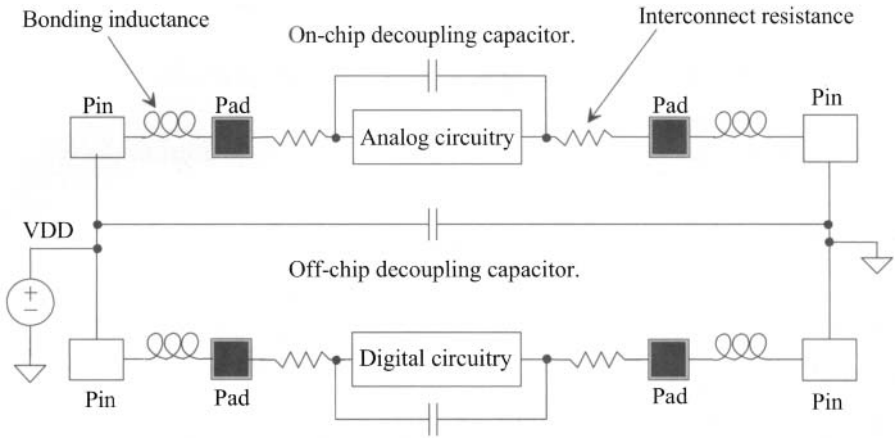


Figure 30.77 Showing how decoupling capacitors are used in a mixed-signal chip.

In general we don't want any DC current flowing on our reference voltages (those voltages used for V_{CI} in our op-amp) because of the possible voltage drop along the supplying line. There may also be a voltage drop along the metal lines supplying power to the op-amps. However, small (DC) changes in these voltages are usually not a significant factor in the precision of the ADC. These changes could be a factor if the output signal approaches the power-supply voltages where the op-amp runs out of head room. AC changes in the power-supply voltages can be a significant factor limiting the ADC's performance.

Ground planes and wide conductors should be used where possible. Using power and ground planes not only provides good distribution of power and ground but also increases the capacitance between the supplies. Areas, labeled "gutter" in Fig. 30.76 should be provided for low-level analog signals. These areas are free to allow the quiet routing of the switch inputs and outputs to the op-amp outputs.

It's also a good idea to use guard rings around the sensitive analog circuits (e.g., the switched capacitors) to avoid coupled substrate noise. Figure 30.78 shows the basic idea. In this figure the capacitors are laid out over an n-well. The n-well is tied to analog

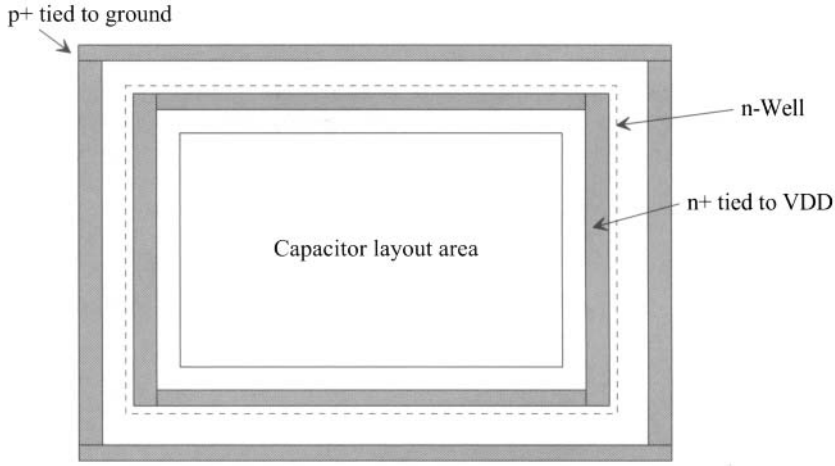


Figure 30.78 Using guard rings for protection in sensitive analog blocks.

VDD through an n+ implant in the n-well and metal. Surrounding the n-well is a ring of p+. This ring is tied to analog ground. The idea is that the p+ will provide a sink for any current injection from the surrounding circuitry. Because ground is the lowest potential in the circuit, the noise will terminate on this p+ and not penetrate the area under the capacitors (and then hopefully not couple into the capacitors). While this works well by itself, it may not be enough. Noise currents may still move deep in the substrate and work their way up under the capacitors. Because the n-well under the capacitor is held at the most postive potential in the circuit, any noise that does get into the n-well will hopefully be swept out through *VDD* and not couple into the capacitors.

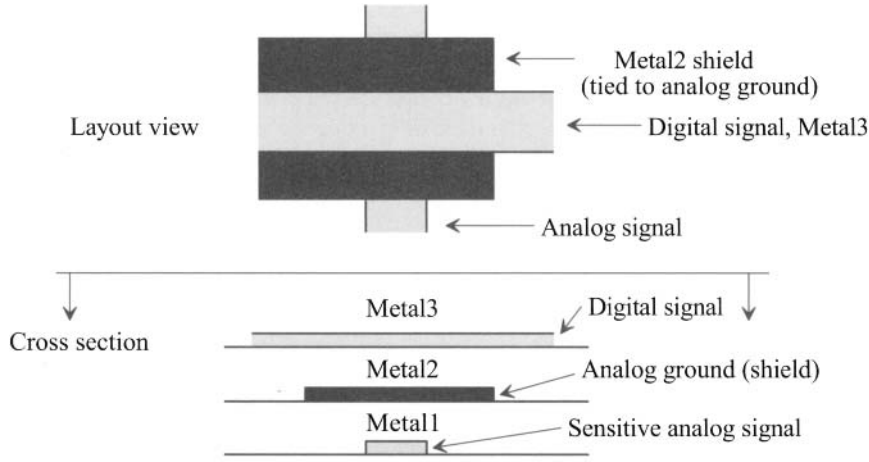


Figure 30.79 Shielding a sensitive analog signal from a digital signal.

Finally, if a sensitive analog signal does need to cross a digital signal (an example being the potential need to feed the switch inputs and outputs across the three phases of the switch clock signals in Fig. 30.76) a shield should be used, Fig. 30.79. In this figure the sensitive analog signal is assumed to exist on Metal1, while Metal2 is used for an analog ground shield from the digital signal on Metal3. This shield is used for isolation providing a terminating plane for the electric fields resulting from the voltages on both the digital and analog signals.

ADDITIONAL READING

[1] R. J. Baker, *CMOS: Mixed-Signal Circuit Design, Second Edition*, Wiley-IEEE Press, 2009.

[2] Engineering Staff Analog Devices Inc., *Data Conversion Handbook (Analog Devices)*, Newnes, 2005. ISBN 978-0750678414

[3] R. J. van de Plassche, *CMOS Integrated Analog-to-Digital and Digital-to-Analog Converters*, Second Edition, Springer, 2003.

[4] S. Franco, *Design with Operational Amplifiers and Analog Integrated Circuits*, Third Edition, McGraw-Hill, 2003.

[5] H. S. Chen, B. S. Song, and K. Bacrania, "A 14-b 20-Msample/s CMOS Pipelined ADC," *IEEE Journal of Solid State Circuits*, Vol. 36, No. 6, pp. 997-1001, June 2001.

[6] M. Gustavsson, J. J. Wikner, and N. Tan, *CMOS Data Converters for Communications*, Springer, 2000. ISBN 978-0792377801

[7] C. C. Enz and G. C. Temes, "Circuit Techniques for Reducing the Effects of Op-Amp Inperfections: Autozeroing, Correlated Double Sampling, and Chopper Stabilization," *Proceedings of the IEEE*, Vol. 84, No. 11, pp. 1584-1614, November 1996.

[8] C. G. Yu and R. L. Geiger, "An Automatic Offset Compensation Scheme with Ping-Pong Control for CMOS Operational Amplifiers," *IEEE Journal of Solid-State Circuits*, Vol. 29, No. 5 May 1994, pp. 601-610.

[9] B. Ginetti, P. G. A. Jespers, and A. Vandemeulebroecke, "A CMOS 13-b Cyclic RSD A/D Converter," *IEEE Journal of Solid-State Circuits*, Vol. 27, No. 7, July 1992.

[10] B. S. Song, M. F. Tompsett, and K. R. Lakshmikumar, "A 12-bit, 1-MSample/s Capacitor Error-Averaging Pipelined A/D Converter," *IEEE Journal of Solid State Circuits*, Vol. 23, No. 6, pp. 1324-1333, December 1988.

[11] W. C. Black and D. A. Hodges, "Time Interleaved Converter Arrays," *IEEE Journal of Solid-State Circuits*, Vol. SC-15, No. 6, December 1980.

[12] J. A. Schoeff, "An Inherently Monotonic 12-bit DAC," *IEEE Journal of Solid-State Circuits*, Vol. SC-14, No. 6 December 1979, pp. 904-911.

[13] R. E. Suarez, P. R. Gray, and D. A. Hodges, "All-MOS Charge Redistribution Analog-to-Digital Conversion Techniques - Part II," *IEEE Journal of Solid-State Circuits*, Vol. 10, No. 6, pp. 379-385, December 1975.

Problems

30.1 Assuming the DAC shown in Fig. 30.1 is 8 bits and V_{REF+} = 1 V and V_{REF-} = 0, what are the voltages on each of the *R*-2*R* taps?

30.2 Give an example of how the traditional current-mode DAC will have limited output swing.

30.3 Repeat problem 30.1 for the DAC shown in Fig. 30.2.

30.4 For the wide-swing current mode DAC shown in Fig. 30.3, what are the voltages at the taps along the *R*-2*R* string assuming 8 bits, V_{REF+} = 1 V, V_{REF-} = 0, and a digital input code of 0000 0000?

30.5 Can the op-amp shown in Fig. 30.37 be used in fully-differential implementations of the DACs shown in Figs. 30.1 - 30.3? Why or why not?

30.6 Show the detailed derivation of Eqs. (30.12)-(30.14).

30.7 Why would we want to use both current segments and binary-weighted currents to implement a current-mode DAC? (Why use segmentation?)

30.8 Why do we subtract ΔA in Eq. (30.36)? Why not add the gain variation?

30.9 Does the matching of the capacitors matter in the S/H of Fig. 30.31? Why or why not?

30.10 Derive the transfer function of the S/H in Fig. 30.30 if V_{CM} on the left side of the schematic is replaced with ground so that the bottom plates of the C_1 capacitors are grounded when ϕ_3 goes high.

30.11 Determine the transfer function of the S/H in Fig. 30.34 if the top left ϕ_2-controlled switch is connected to the input instead of V_{CM}. Include the effects of offsets and simulate the operation of the circuit to verify your calculations.

30.12 Repeat Ex. 30.10 if the cyclic ADC's input is 0.41 V.

30.13 Is kick-back noise from the comparator a concern for the circuit of Fig. 30.39?

30.14 Derive the transfer function for the circuit shown in Fig. 30.80.

30.15 Repeat Ex. 30.16 if the input voltage is 0.41 V.

30.16 Repeat Ex. 30.17 if the input voltage is 0.41 V.

30.17 Resketch the clock waveforms for Fig. 30.54 if bottom plate sampling is used.

30.18 Show the derivation leading up to Eq. (30.83). Show, using practical values for mismatch, how the squared mismatch terms are negligible.

30.19 What happens to the error adjustment term in Eq. (30.92) if the capacitors in the S/H are perfectly matched?

30.20 Repeat Ex. 30.18 if all capacitors are 1 pF (the ideal situation) and verify that the error out of the stage is zero.

30.21 Sketch a circuit to provide the inputs for the four-phase, nonoverlapping clock generator shown in Fig. 30.81.

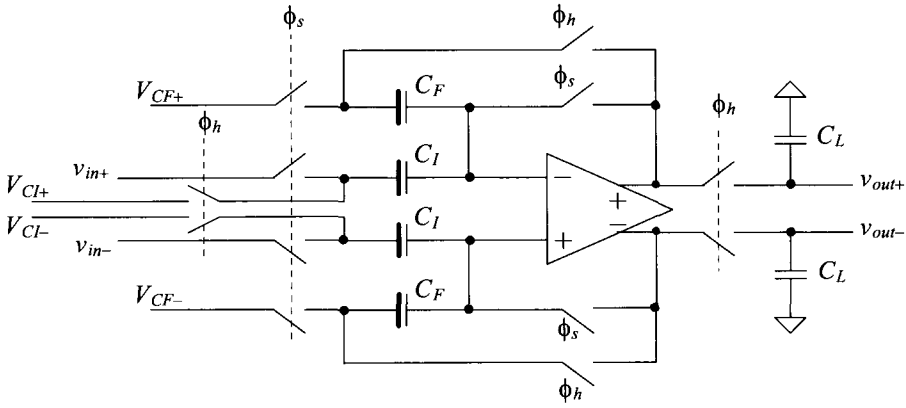


Figure 30.80 Circuit used in problem 30.14.

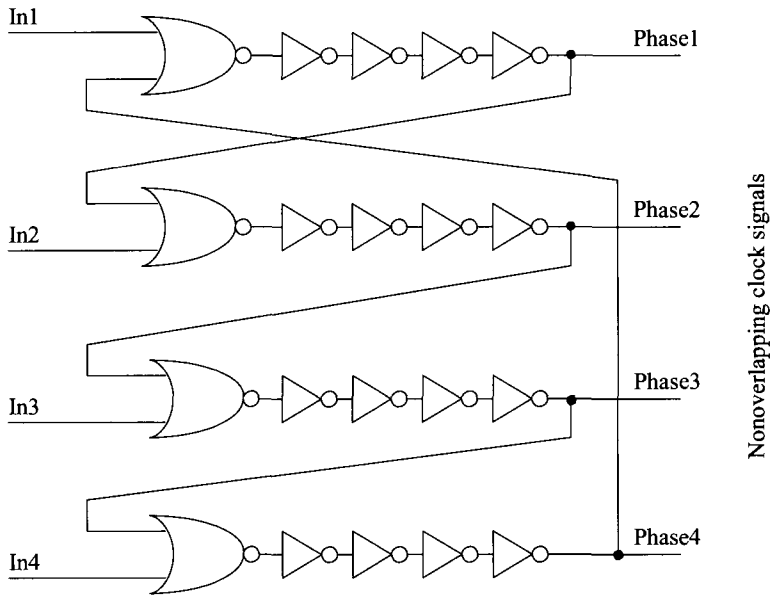


Figure 30.81 Four-phase, nonoverlapping clock generator.

30.22 What is the main advantage of using dynamic CMFB over other CMFB circuits? What is the main disadvantage?

30.23 Can MOSFETs be used to implement the on-chip decoupling capacitors in Fig. 30.77?

30.24 Sketch the cross-sectional view of the layout in Fig. 30.78.

30.25 Figure 30.82 shows the implementation of a pipeline DAC. How would we implement this DAC using a topology similar to Fig. 30.42? Sketch the DAC's implementation and the timing signals (clock phases) used.

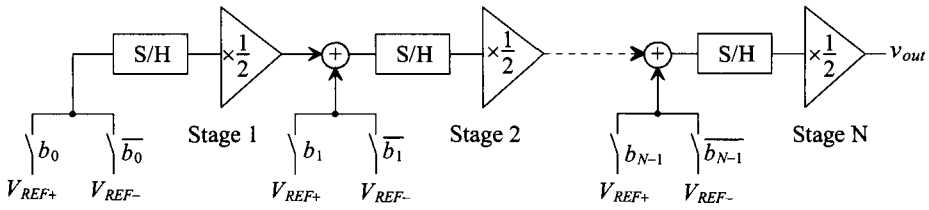


Figure 30.82 A pipeline digital-to-analog converter.