Chapter

17

Sensing Using ΔΣ Modulation

In the last chapter we performed sensing by clocking a sense amplifier that compared two inputs and determined which one had the greater value. To illustrate some concerns with this approach, as signals get smaller or noisier, consider the water analogy seen in Fig. 17.1. In this figure we show two buckets filled with water. Our sensing circuit should determine if the water is above or below the line indicated on the bucket. In (a) our sense-amp can clearly determine that the water is below the line. In (b), however, the water is sloshing around, so it's difficult to determine if the level is above or below the line. To make a more accurate determination in (b), we might try averaging, at different times, several sense amplifier outputs. For example, we might get, after strobing the sense-amp four times, outputs of: yes (it's above the line), no (it's not), yes, yes. Averaging these responses results in the answer "yes, it is above the line." The averaging can be thought of as reducing the noise in the signal (the variations in the water level because of sloshing).

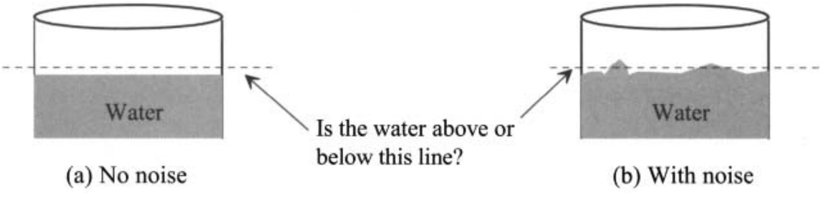


Figure 17.1 Using a water analogy to illustrate the problems with sensing.

Let's take this a step further. The output in the previous analogy resulted in answering the question, "Is the water above or below the line?" In many applications (multi-level memory cells, imaging sensors, analog-to-digital conversion, etc.) this isn't enough information. We need to determine the actual height of the water in the bucket. This means we need more than just a "1" (yes) or "0" (no) output. The purpose of this chapter is to present a powerful and practical circuit technique called *delta-sigma modulation* (also known as *sigma-delta modulation*) for sensing applications (where the desired signal we are sensing is a constant but may be corrupted with noise).

17.1 Qualitative Discussion

Figure 17.2 shows how delta-sigma modulation (DSM) works. The height of the water in the bucket that is "sensed" is changed into a rate of water flow. If the height of the water level increases, the float moves upwards causing the valve to open up and more water to fall into the "sigma" bucket. If the water level gets too high (above an arbitrary line in the sigma bucket), we remove a cup of water. The delta is the difference in how many times we remove a cup of water per time with the rate the water flows into the sigma bucket. By averaging the number of times we remove a cup of water over time, we can determine the rate the water flows out of the valve and thus the relative height of the water in the bucket we are sensing. The best way to understand the fundamentals of DSM is with examples.

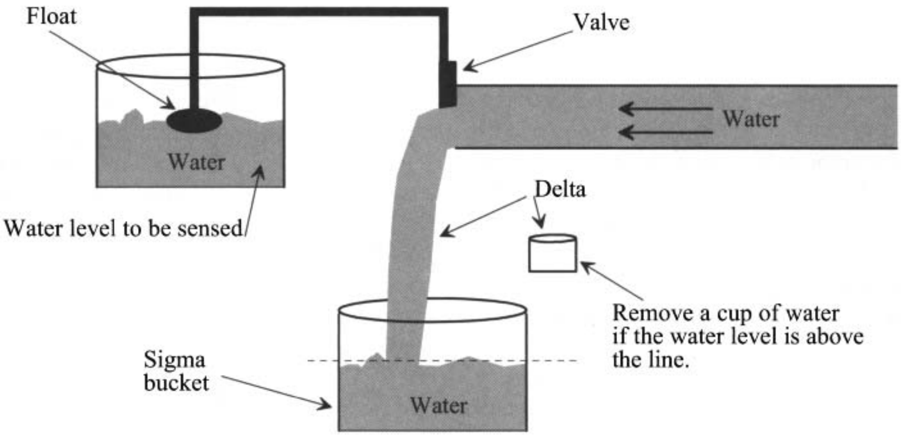


Figure 17.2 Changing the water height (pressure or voltage) into a water flow (current) using the float and valve. The delta comes from the difference in the number of cups we remove from the bucket with the water we add to the bucket. The sigma (sum) is the storage of the difference in the bucket.

17.1.1 Examples of DSM

Let's say the rate the water is flowing into the sigma bucket is one cup every 40 seconds (0.25 cups per 10 seconds). Further, let's say we check the height of the water in this bucket every 10 seconds. If we start the sense off with the water height in the sigma bucket at (arbitrarily) 5 cups (our reference line), we can generate the data in Table 17.1. At a time of 10 seconds 0.25 cups have fallen into the sigma bucket and so the water level is 5.25 cups. Since this water level is > 5 cups, we remove a cup of water from the bucket (at 10 seconds) leaving 4.25 cups in the bucket. At 20 seconds, after another 0.25 cups of water have fallen into the bucket, we have 4.5 cups. Notice that the longer we average, the closer our output moves to 0.25 cups/10 seconds.

Another key point to notice is that if we make a mistake when determining if the water level in the sigma bucket is > 5 cups, it doesn't really matter. The error averages out over time. What does matter though is how carefully we fill up the cup when removing water. If we don't fill it up all the way or if the water spills out of the cup, the level of the water in the sigma bucket changes. The result limits the precision of the sense. One other limiting factor is the sigma bucket. If it is "leaky" and the water it holds leaks out, the quality of the sense will be affected.

Table 17.1 An example of DSM.

Time, seconds	**Water level in the sigma bucket (cups)**	**Remove cup? (water level > 5?)**	**Running average**
0	5		0
10	5.25	Yes	1
20	4.5	No	0.5
30	4.75	No	0.33
40	5	No	0.25
50	5.25	Yes	0.4
60	4.5	No	0.33
70	4.75	No	0.29
80	5	No	0.25
90	5.25	Yes	0.33
100	4.5	No	0.3
110	4.75	No	0.27
120	5	No	0.25
130	5.25	Yes	0.31
140	4.5	No	0.29
150	4.75	No	0.26
160	5	No	0.25
170	5.25	Yes	0.29

The Counter

We might wonder how, in a practical sensing circuit, we can get decimal numbers like the ones seen in Table 17.1? The answer to this question is seen in Fig. 17.3. If a "yes, remove a cup of water" is indicated by the output of the DSM sensing circuit going high, a counter can be used to generate the number. If N is the total number of times we clock the DSM, then

$$\text{Output number} = \frac{\text{number of Yes outputs}}{N} \qquad (17.1)$$

In Table 17.1 N is 17. The number of "yes" outputs is 5. The final output number is then 5/17 or 0.29. If a 10-bit counter is used (assuming it was reset to zeroes at the beginning of the sense), then its digital output is 00 0000 0101. To find the absolute value of the input, we multiply this number by the size of the feedback signal (here one cup) to get

Input signal = Ouput number × fed back signal size (17.2)

or here 0.29 cups (per 10 seconds). Note that to converge on the actual rate of 0.25 cups/10 seconds we would need to average more points.

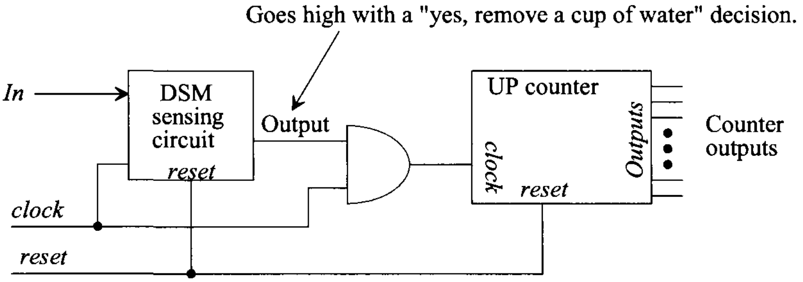


Figure 17.3 How a counter is used to average the outputs of a DSM sensing circuit.

Cup Size

The size of our fed back signal, here the cup size, is very important. If we use a small cup, we can converge on the correct digital representation (the counter output) of our analog signal (the flow of water into the sigma bucket) quicker. However, if we use too small of a cup, then we can't remove the water fast enough from the bucket and it will overfill. For a large range, we need a big cup. Using a big cup means that we have to average longer (the sensing lasts for a longer period of time).

Another Example

Figure 17.4 shows another water analogy where DSM can be used for measuring an analog quantity (in this case the water flowing out of the sigma bucket). When the water level gets too low, we add a cup of water to the bucket. Again, averaging how often we add the cup of water to the bucket gives a digital representation of the rate at which water leaves the bucket.

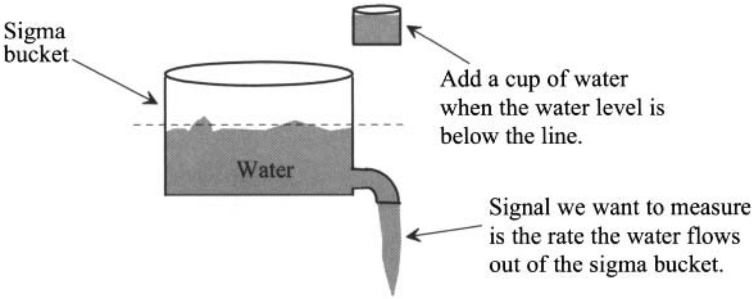

Sigma bucket
Water
Add a cup of water when the water level is below the line.
Signal we want to measure is the rate the water flows out of the sigma bucket.

Figure 17.4 Using DSM to measure the water leaving a bucket.

17.1.2 Using DSM for Sensing in Flash Memory

Let's discuss sensing using DSM in a floating gate memory cell technology (Flash). Figure 17.5a shows the characteristics of the Flash memory cell discussed in the last chapter (see Fig. 16.64). When we are sensing the state of the Flash cell, that is, erased or programmed, we hold the row line at ground so that the cell's V_{GS} is 0. If the bit line is held at a potential above ground, say 1 V, then a current of either I_{erased} or I_{prog} (ideally) flows in the cell. In reality variations in the production of the cell and the consistency of the erase from cell to cell affect these values. Using DSM we can more precisely determine the drain current in the cell. This allows us to make a more intelligent decision about the state of the cell. Further, DSM can also be used to program the cell to precisely set the programmed current flow. This allows us to make a memory cell out of a single transistor, which can be used to store several logic levels (values of programmed current). Because of the ability to precisely control the programming operation in a floating gate technology, programming using Fowler-Nordheim Tunneling (FNT) can be abandoned (gate oxides of around 80 Å) for direct tunneling (gate oxides < 20 Å). Of course, the issues with data retention (the charge leaking off the floating gate) are still a concern. In this section we qualitatively discuss how DSM can be used for sensing in Flash technology.

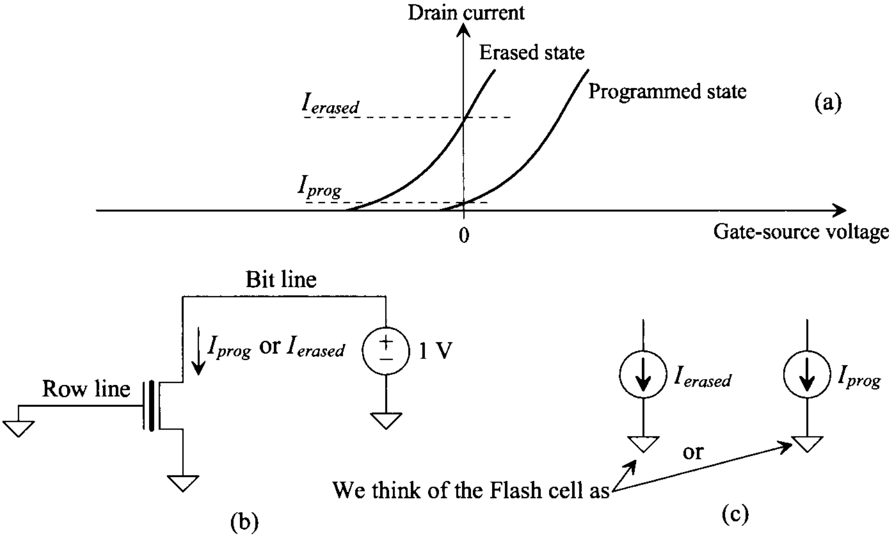


Figure 17.5 The IV characteristics of a NAND memory cell and how we think of thε cell when sensing.

The Basic Idea

Figure 17.6 shows the basic idea. The sigma bucket in this figure is the bit line capacitance, C_{bit}. As seen in Fig. 17.4, the signal we are measuring is the rate current flows out of this bucket, that is I_{erased} or I_{prog}. The comparator is used to determine if the voltage on the bit line is below 1 V (again an arbitrary reference). The memory cell is continuously removing charge from the bit line. When the bit line voltage is below 1 V,

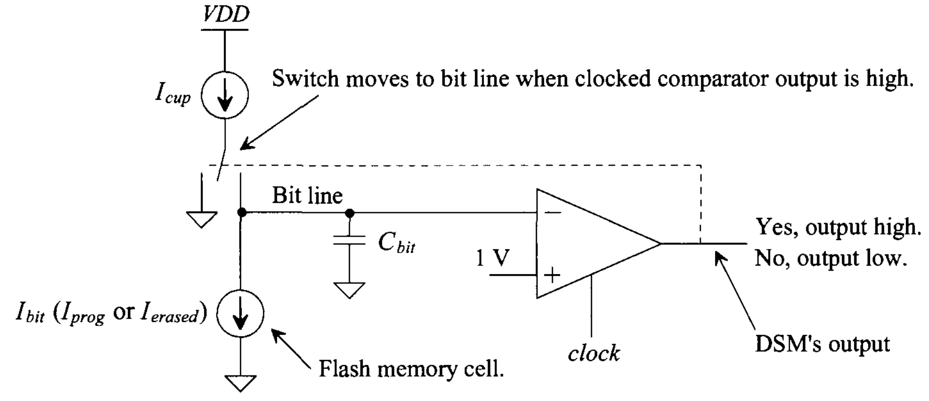

VDD
I_{cup}
Switch moves to bit line when clocked comparator output is high.
Bit line
C_{bit}
1 V
−
+
Yes, output high.
No, output low.
I_{bit} (I_{prog} or I_{erased})
Flash memory cell.
clock
DSM's output

Figure 17.6 Sensing a Flash memory cell using DSM.

the current source, I_{cup} , is connected to C_{bit} to provide the water to the bucket. Note that *we cannot connect the bit line to a voltage source when the output of the comparator goes high.* A voltage source, in our water analogy, is a pressure with an infinite supply of water. The connection of the voltage source would fill the bucket up (the bit line capacitance) to the pressure of the source (the voltage of the source). Again, by looking at the number of times the output of the DSM sensing circuit goes high, we can determine, very precisely, the value of I_{bit}.

If we are clocking the comparator at a rate of f at a period of T (= $1/f$), then the rate that charge is removed from the bit line capacitance (the sigma bucket) is

$$\frac{I_{bit}}{C_{bit}} = \frac{\Delta V_{bit}}{T} \qquad (17.3)$$

where ΔV_{bit} is the change in the bit line voltage. The amount of charge removed from the bit line in one clock cycle T is

$$Q_{bit} = I_{bit}T = C_{bit} \cdot \Delta V_{bit} \qquad (17.4)$$

These quantities are constant. However, the rate at which we add charge to the bit line isn't a constant (unless the output of the comparator is always the same). If N is the total number of clock cycles (the total number of times we clock the comparator) and M is the number of times the output of the comparator goes high (the number of "Yes, add charge to the capacitor" signals), then the rate we add charge to the bit line from I_{cup} is

$$Q_{cup} = I_{cup} \cdot \frac{M}{N} \cdot T \qquad (17.5)$$

In order for the voltage on the bit line (the water level in the bucket) to remain, on average, constant, we require that the amount of charge leaving the bucket (Q_{bit}) equal the amount of charge entering the bucket (Q_{cup}) or

$$Q_{bit} = I_{bit}T = Q_{cup} = I_{cup} \cdot \frac{M}{N} \cdot T \qquad (17.6)$$

or

$$\frac{I_{bit}}{I_{cup}} = \frac{M}{N} \tag{17.7}$$

This result is important. It relates the counter output code, M, and the total number of times the DSM sensing circuit is clocked, N, to the ratio of the Flash memory cells current, I_{bit} and the fed-back signal I_{cup}. Note again that the charge leaving the bit line capacitance can't be greater than the charge entering the bit line capacitance, that is, we require $I_{cup} \geq I_{bit}$. As mentioned earlier, using a large cup (large I_{cup}) increases the sensing time for a required resolution (we're averaging a larger variable).

Notice that Eq. (17.7) doesn't include the clock frequency or period. We might think that these quantities aren't important. While, if the DSM is designed correctly, it doesn't directly affect the output of the sensing operation, we can have the situation where the bucket (bit line capacitance) empties or overflows (goes to ground or VDD). To avoid this situation, we may require that the maximum deviation on the bit line, $\Delta V_{bit,\max}$ be less than some value over a time T. We can write, assuming $I_{cup} \geq I_{bit}$,

$$\Delta V_{bit,\max} = \frac{I_{cup}T}{C_{bit}} = \frac{I_{cup}}{C_{bit} \cdot f_{clk}} = I_{cup} \cdot R_{sc} \tag{17.8}$$

For example, if the bit line capacitance is $500 fF$, the clock frequency is 100 MHz (T = 10 ns), and I_{cup} is 10 μA, then $\Delta V_{bit,\max} = 0.2\ V$. If we clock the DSM slower, we must increase our bucket size (add capacitance in parallel with C_{bit} on the input of our DSM).

Example 17.1

Suppose that the DSM sensing circuit in Fig. 17.6 is used to determine the current flowing in a programmed Flash memory cell. If the single transistor Flash memory cell is programmed to conduct 1, 3, 5, or 7 μA of current, estimate the counter output codes if the DSM is clocked 15 times (a 4-bit counter is used) and I_{cup} is 10 μA. Estimate the maximum bit line voltage change if the DSM is clocked at 100 MHz and the bit line capacitance is $500 fF$.

For the 1 μA program current, using Eq. (17.7), we get

$$\frac{1}{10} = \frac{M}{15} \rightarrow M = 1.5 \text{ so the counter output would be 2 (0010)}$$

For the 3 μA program current through the Flash memory cell,

$$\frac{3}{10} = \frac{M}{15} \rightarrow M = 4.5 \text{ so the counter output would be 5 (0101)}$$

For the 5 μA program current,

$$\frac{5}{10} = \frac{M}{15} \rightarrow M = 7.5 \text{ so the counter output would be 8 (1000)}$$

Finally, for the 7 μA program current,

$$\frac{7}{10} = \frac{M}{15} \rightarrow M = 10.5 \text{ so the counter output would be 11 (1011)}$$

The maximum deviation of the voltage on the bit line would be when the program current is 1 μA (leaving the bit line) and I_{cup} is connected (10 μA charging the bit line). This would give a net current of 9 μA into the bit line so

$$\Delta V_{bit,\max} = \frac{9\mu A \cdot 10\, ns}{500 fF} = 180\, mV$$

Let's show the output decisions for the case when I_{prog} is 1 µA, Table 17.2. Note that if 1 µA is removed from the 500 *fF* bit line capacitor for 10 ns, the bit line voltage drops 20 mV. If (net) 9 µA of current is put into the bit line for 10 ns, the bit line voltage increases by 180 mV. Note that as calculated, the maximum change in the bit line voltage is 180 mV. Also note that if the comparator had output a Yes at 100 ns instead of at 110 ns, it wouldn't have affected the final output. ■

Table 17.2 See Ex. 17.1.

Time, nanoseconds	Bit line voltage	Add current? (V_{bit} < 1 V?)	Bit line current	Counter output
0	1		−1 µA	0 (0000)
10	0.98	Yes (1)	9 µA	1 (0001)
20	1.16	No (0)	−1 µA	1 (0001)
30	1.14	No (0)	−1 µA	1 (0001)
40	1.12	No (0)	−1 µA	1 (0001)
50	1.1	No (0)	−1 µA	1 (0001)
60	1.08	No (0)	−1 µA	1 (0001)
70	1.06	No (0)	−1 µA	1 (0001)
80	1.04	No (0)	−1 µA	1 (0001)
90	1.02	No (0)	−1 µA	1 (0001)
100	1	No (0)	−1 µA	1 (0001)
110	0.98	Yes (1)	9 µA	2 (0010)
120	1.16	No (0)	−1 µA	2 (0010)
130	1.14	No (0)	−1 µA	2 (0010)
140	1.12	No (0)	−1 µA	2 (0010)
150	1.1	No (0)	−1 µA	2 (0010)

Notice, in the previous example, that we can clock the DSM indefinitely. The only limitation is the size of the counter (after a time the counter will roll over). This is important because, to get better resolution, all we have to do is sense for a longer period of time. We can define the *dynamic range* of the sense operation, in dB, as the maximum counter output code (*N*) to the smallest output code (1)

$$\text{dynamic range (DR)} = 20 \cdot \log\frac{N}{1} = 20 \cdot \log N \qquad (17.9)$$

noting that no signal would give a counter output code of zero. Clocking the DSM 1,000 times results in, ideally, a DR of 60 dB. Clocking the DSM 15 times results in a DR of 23.5 dB.

The maximum input signal, $I_{bit,\max}$, equals the fed-back signal I_{cup} $(I_{cup} \geq I_{bit})$. The minimum resolvable signal is then

$$\text{minimum resolvable signal} = \frac{\text{fed-back signal}}{N} \tag{17.10}$$

Again, this illustrates that the larger the fed-back signal, the more averages, N, are needed for a given resolution.

Example 17.2

Determine the minimum resolvable programmed current, I_{bit} , in Ex. 17.1. Verify the answer with a table similar to Table 17.2.

Using Eq. (17.10), the minimum resolvable current is 10 μA/15 or 0.666 μA. If this current is removed from the bit line capacitance for 10 ns, the voltage drop is 13.33 mV. If the bit line capacitance is charged with 9.333 μA for 10 ns, its voltage increases to 186.66 mV. Table 17.3 tabulates the operation of the DSM sensing circuit when I_{prog} is 666.6 nA. Note that in all situations where I_{prog} is not zero the first decision is a Yes. However, after neglecting this decision, N clock cycles later the counter increments to 2 when the cell is sinking the minimum resolvable signal (here 666.6 nA). ■

Table 17.3 See Ex. 17.2.

Time, nanoseconds	Bit line voltage (millivolts)	Add current? (V_{bit} < 1 V?)	Bit line current	Counter output
0	1,000		–0.666 μA	0 (0000)
10	986.67	Yes (1)	9.333 μA	1 (0001)
20	1,173.33	No (0)	–0.666 μA	1 (0001)
30	1,160	No (0)	–0.666 μA	1 (0001)
40	1,146.67	No (0)	–0.666 μA	1 (0001)
50	1,133.34	No (0)	–0.666 μA	1 (0001)
60	1,120	No (0)	–0.666 μA	1 (0001)
70	1,106.67	No (0)	–0.666 μA	1 (0001)
80	1,093.33	No (0)	–0.666 μA	1 (0001)
90	1,080	No (0)	–0.666 μA	1 (0001)
100	1,066.67	No (0)	–0.666 μA	1 (0001)
110	1,053.33	No (0)	–0.666 μA	1 (0001)
120	1,040	No (0)	–0.666 μA	1 (0001)
130	1,026.67	No (0)	–0.666 μA	1 (0001)
140	1,013.33	No (0)	–0.666 μA	1 (0001)
150	1,000	No (0)	–0.666 μA	1 (0001)
160	986.67	Yes (1)	9.333 μA	2 (0010)

The Feedback Signal

The precision of a sense operation in a DSM is limited by how precisely current can be guided into or out of the bit line capacitance. The bit line capacitance can be thought of as integrating (an integrator, *sigma*) the difference (*delta*) between the fed-back signal (I_{cup} in Fig. 17.6) and the signal we are measuring (I_{bit}). If the amount of charge (current) steered into the bit line isn't consistent from one clock period to the next, errors will occur. To illustrate the possibility of errors, examine the circuit seen in Fig. 17.7. When the switch is connected to ground $C_{parasitic}$ (the parasitic capacitance on the output of the current source), is discharged. Now when the switch is connected to the bit line I_{cup} must charge both $C_{parasitic}$ and C_{bit}. Because the bit line voltage will be moving around and the switch may stay connected to the bitline for two or more consecutive cycles, errors in the sense will occur that limit the resolution. For example, the charge supplied to $C_{parasitic}$ may be $1.05 \cdot C_{parasitic}$ (a bit line voltage of 1.05V) in one clock cycle, while in a different clock cycle it may be $1.02 \cdot C_{parasitic}$. In our water analogy in Fig. 17.4 the amount of water in the cup effectively changes due to this parasitic. If the size of the bit line capacitance is large (or if capacitance is added to the input of the DSM to purposely increase this capacitance) or only a course sense is used (say 128 or less clock cycles), then $C_{parasitic}$ will probably not affect the sense in any significant way. Note that if the switch were connected to 1V in Fig. 17.7 instead of ground, the errors from $C_{parasitic}$ would be reduced.

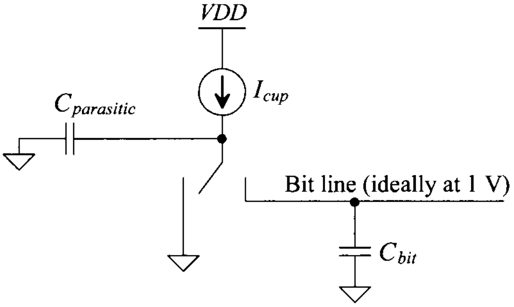


Figure 17.7 How the parasitic capacitance on the output of the current source causes errors.

In an attempt to minimize the power (power is burned unnecessarily when I_{cup} is connected to ground in Figs. 17.6 or 17.7) and the effects of parasitics consider the switched-capacitor (SC) circuit seen in Fig. 17.8. In the following we assume the bit line reference voltage (here 1 V) is greater than the threshold voltage of the PMOS transistors so that the PMOS switches used in the SC can turn fully on. The two clock signals, ϕ_1 and ϕ_2 form nonoverlapping clock signals (they are never low at the same time). This is important because we never want to connect the bit line directly to *VDD* (which occurs if the clock signals are all low at the same time). When ϕ_1 is low, C_{cup} charges to *VDD*. Next, ϕ_1 goes high. Between ϕ_1 going high and ϕ_2 going low, the comparator is clocked (on the rising edge of *clock*). If charge needs to be added to the bit line, the comparator output goes high and the gate of M3 is driven low. Note that the inverter can be removed

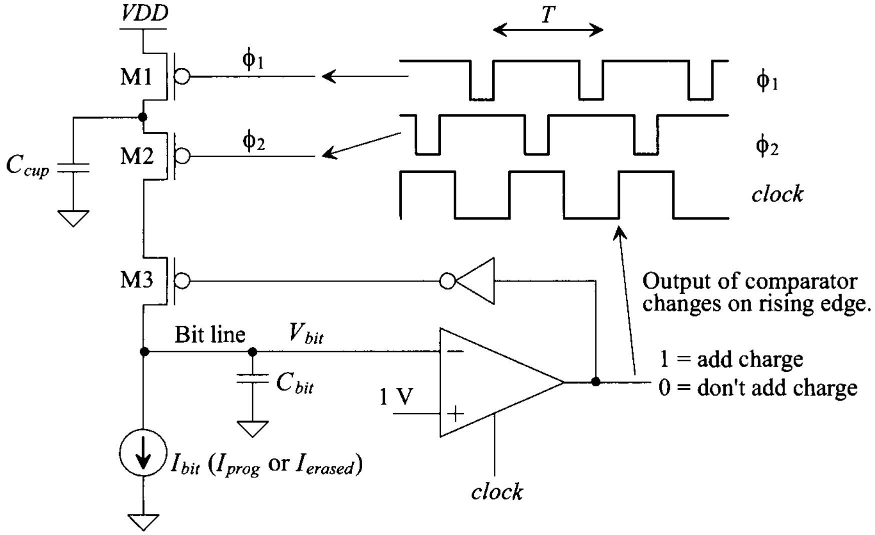


Figure 17.8 Using a switched-capacitor to add charge to the bit line.

if we swap the inputs of the clocked comparator so that a 1 output indicates "don't add charge" and a 0 output indicates "add charge." When ϕ_2 goes low, the charge on C_{cup} is dumped into the bit line. The charge leaving the bit line is still given by Eq. (17.4). The charge dumped into the bit line from C_{cup} is

$$Q_{cup} = C_{cup} \cdot (VDD - V_{bit}) \tag{17.11}$$

When we compare this amount of charge to Eq. (17.5), we see a benefit when Q_{cup} is not a function of the clock period. However, the big drawback is that the charge we add to the bit line is a function of the bit line voltage. This limits the accuracy of the sense. The water analogy to this problem is that the cup is being filled up to a level dependent on the sigma bucket water level. The amount of water dumped into the bucket from the cup should be independent of the water level in the bucket.

To make the charge we add to the bit line independent of the bit line voltage, consider the addition of another PMOS device in Fig. 17.9. When both the signal from the comparator and ϕ_2 are low, M2 and M3 are on. The charge from C_{cup} is dumped into the source of M4. The result is an initial increase in M4's source potential (M4 turns on). However, after the charge is dumped into the bit line, M4 shuts off. Its V_{SG} goes to V_{THP} (M4's source potential goes to $V_{REF} + V_{THP}$). This keeps the potential across C_{cup} constant. Equation (17.11) can be rewritten as

$$Q_{cup} = C_{cup} \cdot (VDD - V_{REF} - V_{THP}) \tag{17.12}$$

knowing, to keep M4 from being fully on, that is, $V_{SD,sat}$ not zero

$$V_{REF} + V_{THP} > V_{bit,\max} \tag{17.13}$$

where $V_{bit,\max}$ is the maximum voltage on the bit line (and, again, $V_{bit,\min} > V_{THP}$).

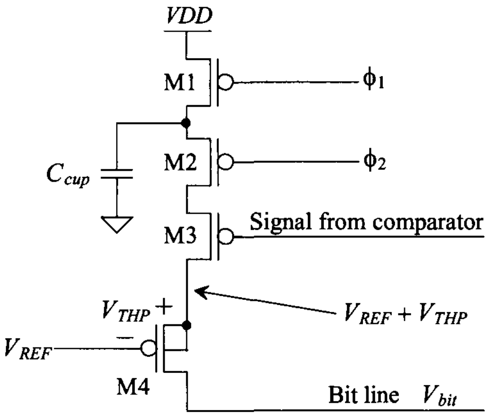


Figure 17.9 Adding a MOSFET to set the swing across the capacitor.

Example 17.3
Using SPICE demonstrate the validity of Eqs. (17.11) and (17.12).

To demonstrate the validity of Eq. (17.11), examine the circuit in Fig. 17.10. We know that a PMOS device passes a voltage from *VDD* to V_{THP}, so we make sure that the bit line is always at a potential greater than V_{THP} (= 280 mV for the short channel process used in this book). If it's not, the PMOS devices don't behave like switches. In the simulation we'll sweep the bit line voltage from 0.3 to 1V and look at the current through the bit line voltage source, V_{bit}. M3 is always on so we can look at the SC's operation alone (without the effects of the comparator). If Eq. (17.11) is valid, we should see a linear decrease in the current pulses (the charge) as V_{bit} is increased. Figure 17.11a shows the nonoverlapping clocks used in the simulation. Note how neither clock (ϕ_1 or ϕ_2) is low at the same time (again, this is important). Figure 17.11b shows the current through V_{bit}. As expected it decreases as V_{bit} increases. As Eq. (17.11) indicates. the current goes to zero when

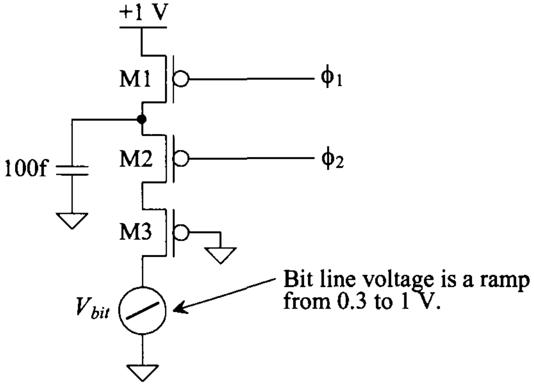
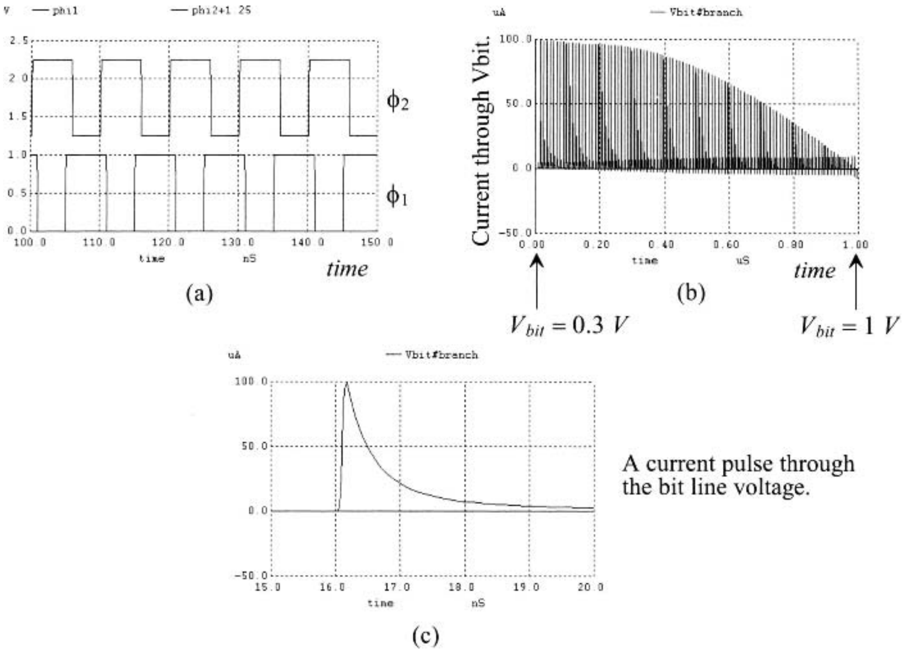


Figure 17.10 Verifying Eq. (17.11), see Ex. 17.3.



A current pulse through the bit line voltage.

Figure 17.11 The operation of the circuit in Fig. 17.10.

V_{bit} is 1 V (= VDD). Let's use Eq. (17.11) to calculate the charge supplied by C_{cup} when V_{bit} is 300 mV

$$Q_{cup} = 100fF \cdot (1 - 0.3) = 70fC \qquad (17.14)$$

Figure 17.11c shows the pulse of current that is dumped into V_{bit} at the beginning of the simulation. To estimate the amount of charge in this pulse, we take the amplitude and multiply it by an estimate of the pulse's width. That is, $100\ \mu A \cdot 0.7\ ns = 70 fC$ or the same result as seen in Eq. (17.14).

To validate Eq. (17.12), we can add M4 from Fig. 17.9 to the circuit seen in Fig. 17.10. The only parameter we need to calculate before simulating is the value of V_{REF}. Using Eq. (17.13) and knowing V_{THP} = 280 mV, we see that if we set V_{REF} to a large voltage, Q_{cup} gets small. If, for example, $V_{REF} = VDD - V_{THP}$ (= 750 mV in this example), Q_{cup} goes to zero. We can write

$$V_{REF} < VDD - V_{THP} \qquad (17.15)$$

Let's use a V_{REF} of $VDD/2$ or 500 mV. Figure 17.12 shows the simulation results. Note, in Fig. 17.9, that we place M4 in its own well. Since we care about how the threshold voltage varies, we've eliminated the body effect in this device. The simulation results are seen in Fig. 17.12. In this simulation we've swept the bit line voltage from 0.3 to 0.6 V. Comparing the plot to Fig. 17.11b, we see much better linearity (the charge added to the bit line doesn't change with bit line voltage). The cost for this improvement is more limited voltage swing on the bit line. ■

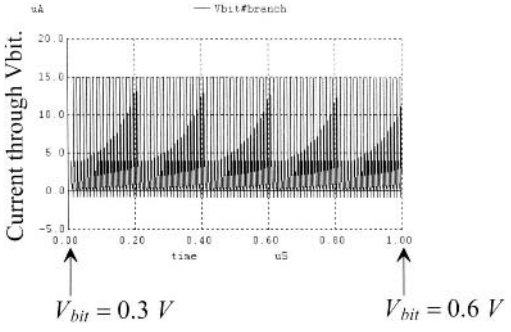


Figure 17.12 How the charge supplied to Vbit becomes linear if we add M4, from Fig. 17.9, to the simulation in Fig. 17.10. Note the reduced bit line voltage swing and the reduction in the current pulse amplitudes.

Incomplete Settling

If we zoom in on the current pulses in Fig. 17.12, we get a view like the one seen in Fig. 17.13. The glitches at 30 and 40 ns are the result of the ϕ_2 clock going high and M2 shutting off. Notice how the current through M4 (and V_{bit}) isn't zero when M2 shuts off. Using our water analogy, this is equivalent to stopping the pouring of the water out of the cup before the cup is empty. This incomplete emptying of the cup or capacitor is termed *incomplete settling.* The currents in the circuit haven't gone to zero before the clock transitions. The parasitic capacitances on the sources of M3/M4 continue to discharge through V_{bit} after M2 shuts off, resulting in nonzero current. Incomplete settling has the effect of making the capacitor, C_{cup} , appear smaller. It won't affect the linearity of the sense but it can affect the maximum value of the sensed current. To eliminate the incomplete settling behavior we can increase the width of M4. This results in the source of M4 being held closer to $V_{REF} + V_{THP}$ when M2 and M3 turns on. We could also slow the clock frequency down to ensure that the circuit settles. Because the amount of charge transferred from C_{cup} is constant from one clock cycle to the next, the effects of incomplete settling on DSM are simply a limit on the maximum current that can be removed from the bit line and a modification to the digital output for the absolute value of the input signal, that is, Eq. (17.2) (a smaller value of fed-back signal size is used).

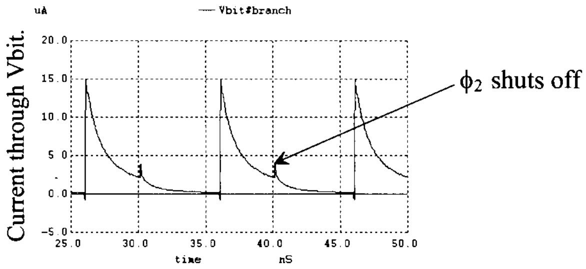


Figure 17.13 The incomplete settling in Fig. 17.12.

17.2 Sensing Resistive Memory

Let's apply the techniques we've just discussed to sensing a resistive memory cell. Figure 17.14a shows a schematic of one-transistor, one-resistor (1T1R) memory cell. Notice the similarity to the 1T1C DRAM memory cell seen in Fig. 16.9. *Ideally* the resistor is either zero ohms (which we'll call *programmed*) or infinite (which we'll call *erased*), Fig. 17.14b. To sense in this ideal case, we simply precharge the bit line to *VDD*. When the word line is driven high, the access device turns on, resulting in the bit line either moving to *VDD*/2 (if the cell is programmed) or not moving at all (if the cell is erased). In a practical circuit, however, the cell's resistance will be nonzero and not infinite (say 10kΩ to 100kΩ as seen in Fig. 17.14c).

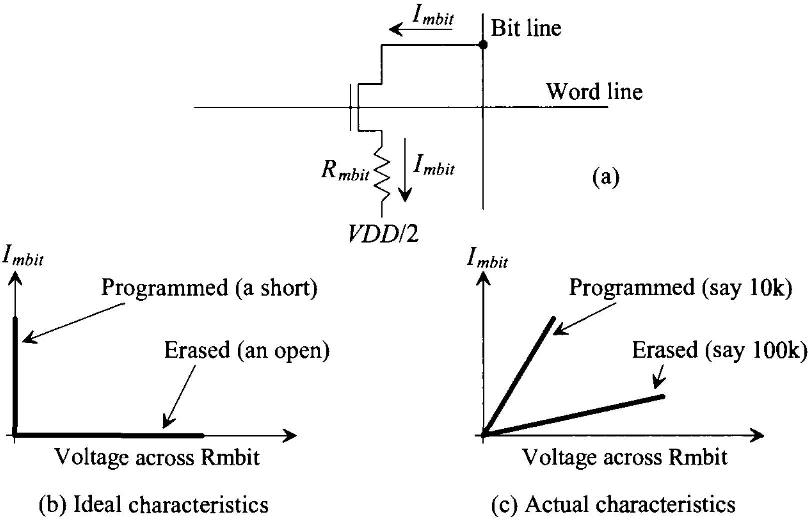


Figure 17.14 The one-transistor, one-resistor (1T1R) RAM memory cell.

The Bit Line Voltage

Consider the block diagram of a DSM seen in Fig. 17.15 with a resistive memory cell. To keep the schematic simpler, we won't include the access MOSFET. We know from the previous discussions that the DSM tries to hold the bit line at precisely the reference voltage used by the comparator. In Fig. 17.15 if the reference voltage used by the comparator is *VDD*/2, then the current that flows through the memory resistor is zero. To avoid this, we might use a reference that is offset from *VDD*/2 by V_{os}. The current that flows in the resistor is then

$$I_{mbit} = \frac{V_{os}}{R_{mbit}} \qquad (17.16)$$

In the ideal condition, the erased resistor results in zero I_{mbit}. The output of the DSM is then simply a string of zeroes (we never have to add current to the bit line because none is leaving). If the resistor is programmed, the bitline gets pulled down towards *VDD*/2. The current source can't supply enough charge to pull the bit line up and the DSM output is a constant string of ones (yes, add current).

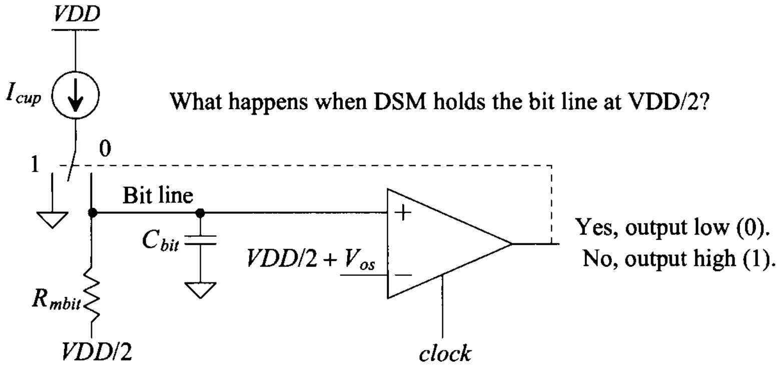


Figure 17.15 Sensing a resitive memory cell using DSM. Notice that we've eliminated the inverter from Fig. 17.8 by switching the input terminals of the comparator.

Adding an Offset to the Comparator

It's important to minimize the number of reference voltages used in a memory sense scheme. We'll use *VDD* and *VDD*/2 and then design a comparator with a built-in offset voltage. The comparator design is seen in Fig. 17.16 (see Figs. 16.32 and 16.35 from the last chapter). Simulation results showing the offset are seen in Fig. 17.17. When In+ gets 50 mV above In−, the output of the comparator changes states. It's possible to design the comparator to simplify the sense circuitry, e.g., clock the comparator on the falling edge of ϕ_2, remove the NAND gates on the output, and combine M2 and M3 in Fig. 17.9. In a

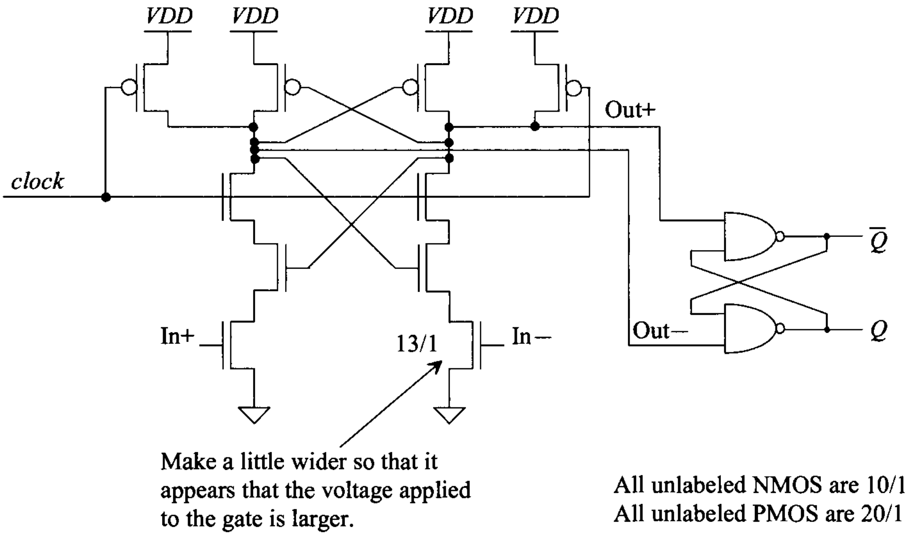

VDD
VDD
VDD
VDD
Out+
clock
$\overline{Q}$
In+
13/1
In−
Out−
Q
Make a little wider so that it appears that the voltage applied to the gate is larger.
All unlabeled NMOS are 10/1
All unlabeled PMOS are 20/1

Figure 17.16 Designing a clocked comparator with a built-in offset.

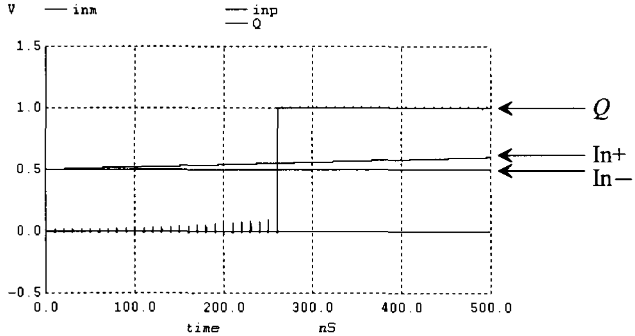


Figure 17.17 The output of the comparator switching states when the + input is 50 mV above the - input.

production part we might want to do this. However, here, where we want to minimize the glitches that may turn some switches inadvertently partially on, we don't try to simplify the circuitry (see problems at the end of the chapter).

Schematic and Design Values

The schematic for a DSM sensing circuit for resistive memories is seen in Fig. 17.18. The capacitance associated with the bit line is shown explicitly (even though it's a parasitic associated with the line). When we are sensing the value of a resistive cell, a word line goes high and connects the resistance, R_{mbit} , to the digit line. The comparator, with its built-in offset, tries to hold the bit line, through the feedback loop, at $VDD/2 + V_{OS}$. The

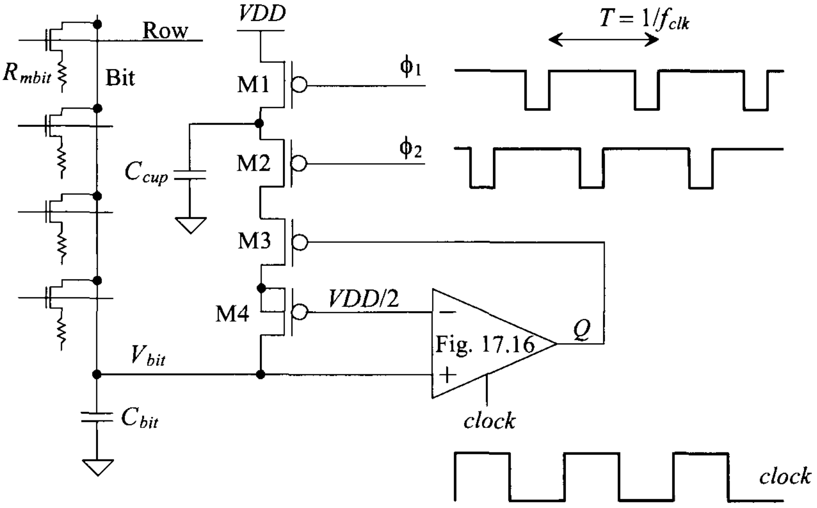


Figure 17.18 Sensing circuit for resistive memory.

current through the cell is then given by Eq. (17.16). Again, if N is the total number of times the DSM is clocked and M is the number of times M3 is enabled so that C_{cup} can dump its charge into the bit line, then we can write (reviewing Eqs. [17.3] to [17.7], [17.12], and [17.16])

$$\frac{V_{OS}}{R_{mbit}} = Q_{cup} \cdot \frac{M}{N} \cdot \frac{1}{T} = (VDD - VDD/2 - V_{THP}) \cdot C_{cup} \cdot \frac{M}{N} \cdot \frac{1}{T} \qquad (17.17)$$

or

$$R_{mbit} = \frac{V_{OS} \cdot T}{(VDD/2 - V_{THP}) \cdot C_{cup} \cdot \frac{M}{N}} \qquad (17.18)$$

Notice that if R_{mbit} goes to zero (programmed), the output of the DSM is always low (M is always a 1). If R_{mbit} is infinite, then the output of the DSM is always high (M is always a 0). To design the DSM, we need to pick an R_{mbit} value that separates what we define as a low resistance (a programmed state) from what we define as a high resistance (an erased state). Let's use 50k. When the resistance is 50k, then $M = N/2$ (half of the clock cycles the output of the DSM goes high). If $M > N/2$, then the cell is programmed. If $M < N/2$, the cell is erased. If V_{OS} is 50 mV, $VDD = 1\ V$, $V_{THP} = 280$ mV, and $f_{clk} = 100$ MHz then

$$C_{cup} = \frac{0.05}{0.22 \cdot 50k \cdot \frac{1}{2}} \cdot 10\,ns = 90\,fF \qquad (17.19)$$

Since offset won't be precisely 50 mV, we'll round this up to $100\,fF$ (and it will vary with process runs). To estimate the value of the resistor given both N and M, we use Eq. (17.18)

$$R_{mbit} \approx 25k \cdot \frac{N}{M} \qquad (17.20)$$

Again, this is an approximation since our value for the offset won't be exactly 50 mV. Note that the minimum value of resistor we can sense occurs when $N = M$ ($R_{mbit} = 25k$).

One more thing that we must calculate before simulating the design is the variation of the bit line voltage (see Eq. [17.8]). The minimum voltage on the bit line is $VDD/2$ (when R_{mbit} is small). The maximum voltage will be $VDD/2 + V_{os} + \Delta V_{bit}$ (where the last term is the maximum change in the bit line voltage). To determine ΔV_{bit}, we can write (knowing charge must be conserved)

$$C_{cup} \cdot (VDD - VDD/2 - V_{THP}) = \Delta V_{bit} \cdot C_{bit} \qquad (17.21)$$

or

$$\Delta V_{bit} = \frac{C_{cup}}{C_{bit}} \cdot (VDD/2 - V_{THP}) \qquad (17.22)$$

Using the numbers above with a C_{bit} of 500 fF gives a $\Delta V_{bit} = 37\ mV$. Note that we were very concerned about how the bit line voltage change affected the amount of charge supplied by C_{cup} (see Figs. 17.8, 17.9, and the associated discussion). Here (and Eq. [17.16]) we aren't concerned with how changes in the bit line voltage affect the current through R_{mbit}. The *average* current through R_{mbit} is V_{OS}/R_{mbit}. The DSM holds the bit line, on average, at $VDD/2 + V_{OS}$. The charge supplied by C_{cup}, however, is not constant on average (as discussed earlier), and that is why M4 was added to the DSM.

Figure 17.19 shows the DSM sensing circuit's output for various values of R_{mbit}. Let's compare the theoretical estimate, that is, Eq. (17.20) to the simulated results. In all of the simulations seen in Fig. 17.19 we clock the DSM 50 times (a 100 MHz clock for 500 ns). In (a), with $R_{mbit} = 25\text{k}$, we get an output of 14 zeroes and 36 ones (= M). We can then write

$$R_{mbit} = 25k \cdot \frac{50}{36} = 35k \text{ (actual value 25k)}$$

For (b), we see $M = 17$ so

$$R_{mbit} = 25k \cdot \frac{50}{17} = 73k \text{ (actual value 50k)}$$

and for (c) and (d),

$$R_{mbit} = 25k \cdot \frac{50}{10} = 125k \text{ (actual value 100k)}$$

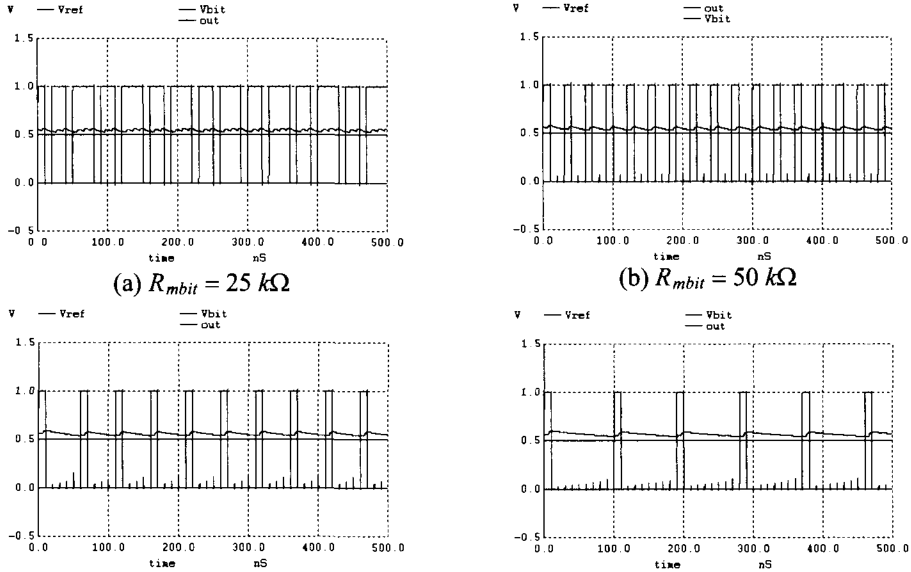
$$R_{mbit} = 25k \cdot \frac{50}{6} = 208k \text{ (actual value 200k)}$$

Notice that as we increase the value of R_{mbit}, we start to get nonlinear. The maximum finite value we can sense with 50 clock pulses is

$$R_{mbit,\max} = \frac{V_{OS} \cdot T \cdot N}{(VDD/2 - V_{THP}) \cdot C_{cup}} \text{ (= 1.25M here)} \tag{17.23}$$

Again, the minimum resistance is

$$R_{mbit,\min} = \frac{V_{OS} \cdot T}{(VDD/2 - V_{THP}) \cdot C_{cup}} \text{ (= 25k here)} \tag{17.24}$$



(a) $R_{mbit} = 25\ k\Omega$

(b) $R_{mbit} = 50\ k\Omega$

(c) $R_{mbit} = 100\ k\Omega$

(d) $R_{mbit} = 200\ k\Omega$

Figure 17.19 Outputs of the DSM for various Rmbit values.

Notice that we get good linearity with (actual) resistance values ranging from 25k to 75k. To sense larger values of resistances with good linearity, we must clock the DSM sensing circuit more times (increase N). However, if we are simply trying to separate a "big" resistor from a "small" resistor, we see that the scheme works very well.

A Couple of Comments

We might wonder why the simulated values are different from the actual values. The answers come from incomplete settling (making C_{cup} appear smaller than it actually is) and from the fact that the offset voltage is not precisely 50 mV. For the incomplete settling problem we can clock the circuit slower or try using a larger width for M4. For the offset voltage problem we might try to generate a precise reference voltage ($VDD/2 + V_{OS}$) for the comparator. Practically, this could lead to problems. In a real memory system, the voltages are noisy. Reviewing the designs in Figs. 17.15 and 17.18, we see that if there is noise on $VDD/2$ it feeds evenly into the comparator circuit and doesn't affect the operation (the – input of the comparator is connected to $VDD/2$ and so is R_{mbit}). Further, even if we could generate noise-free reference voltages, the comparator will exhibit an offset because the MOSFETs won't be perfectly matched. In most sensing applications, the relative values are usually more important than the absolute values.

Example 17.4

Show that M1, M2, M3 and C_{cup} in Fig. 17.18 can be thought of as a resistor. What is the resistor's value?

Figure 17.20 shows the circuit portion from Fig. 17.18. The current that flows in the resistor is

$$I_{avg} = \frac{VDD - VDD/2 - V_{THP}}{R_{SC}} \qquad (17.25)$$

The amount of charge that flows during one clock cycle, if M3 is on, is

$$Q_{cup} = (VDD - VDD/2 - V_{THP}) \cdot C_{cup} \qquad (17.26)$$

or, if this amount of charge is only allowed to flow M times out of N

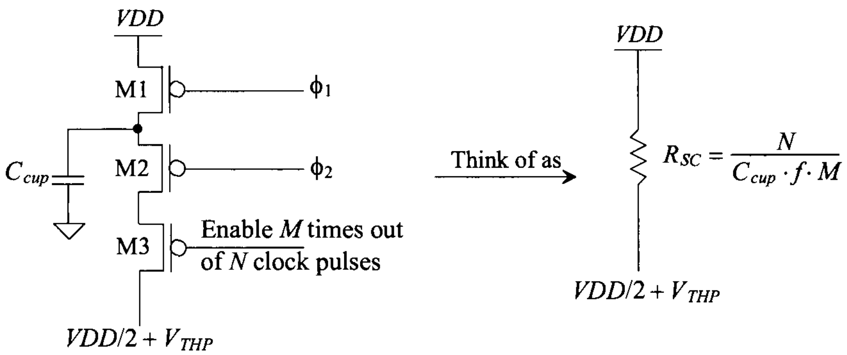

VDD
M1
ϕ1
C_{cup}
M2
ϕ2
M3
Enable M times out of N clock pulses
$VDD/2 + V_{THP}$
Think of as
VDD
$R_{SC} = \frac{N}{C_{cup} \cdot f \cdot M}$
$VDD/2 + V_{THP}$

Figure 17.20 How a switched-capacitor resistor is modeled.

$$I_{avg}=\frac{Q_{cup}}{T}\cdot\frac{M}{N}=Q_{cup}\cdot f\cdot\frac{M}{N} \tag{17.27}$$

or

$$R_{SC}=\frac{N}{C_{cup}\cdot f\cdot M} \tag{17.28}$$

Noting that if M3 is always enabled (the circuit with only M1 and M2) we get a switched-capacitor resistance of

$$R_{SC}=\frac{1}{C_{cup}\cdot f} \tag{17.29}$$

which, for Fig. 17.18 and the associated discussion, is 25k. ■

Example 17.5

Show that M1, M2, and C_{cup} can be replaced with a resistor in the DSM sensing circuit of Fig. 17.18. Show the DSM's output when R_{mbit} is 50k and 200k.

The schematic for the DSM modulator is seen in Fig. 17.21. We might think that the errors due to incomplete settling would be eliminated using this scheme. However, the source of M4 will not be precisely at $VDD + V_{THP}$, so an error will still be present. Again, increasing the width of M4 will lessen the error's effects.

Using the same values as used to generate Fig. 17.19b and d, we get the simulation results seen in Fig. 17.22. The big benefit of this topology over the one in Fig. 17.18 using the switched capacitor resistor is simpler design (no nonoverlapping clocks are needed) and lower power (there are not as many parasitic capacitances to charge and discharge). The current pulled from *VDD* in Fig. 17.18 is 26 μA when clocked at 100 MHz, while the circuit in Fig. 17.21 uses 20 μA. The big drawback of using the simple resistor is the inability to adjust the resistance by changing the clock frequency. In a practical circuit the fabrication

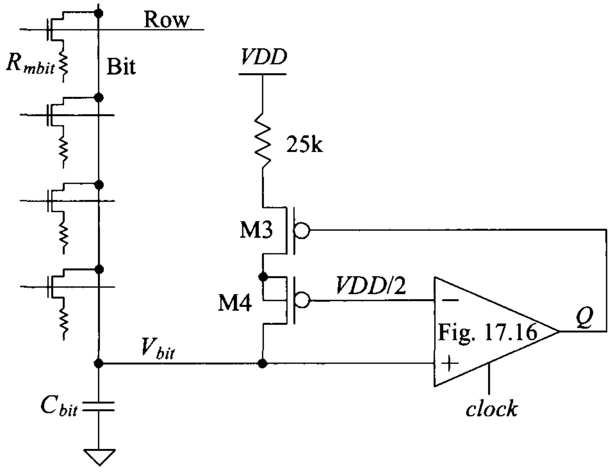
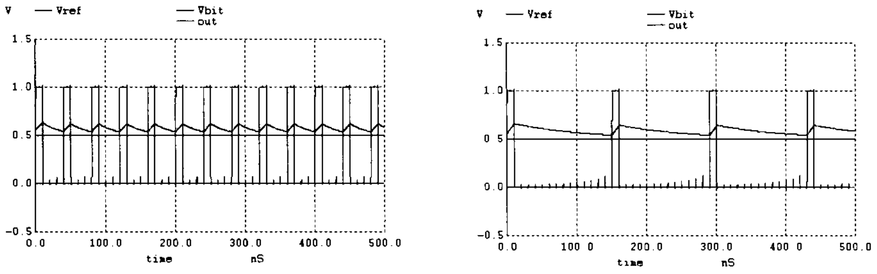


Figure 17.21 Simpler DSM for sensing resistive memory.



(a) $R_{mbit} = 50\ k\Omega$ (b) $R_{mbit} = 200\ k\Omega$

Figure 17.22 Simulation results for the circuit in Fig. 17.21.

process characteristics and temperature vary. To center the value of sense to half way between a programmed and an erased resistor value, the clock frequency can be adjusted. Adjusting the clock frequency in Fig. 17.21 simply adjusts the variation in the bit line voltage. ■

17.3 Sensing in CMOS Imagers

Another area where DSM (delta-sigma modulation) can be used for sensing is in CMOS imaging chips that acquire images in cameras or video recording. A schematic of a CMOS active pixel sensor (APS) is seen in Fig. 17.23. Light is applied through a filter (so that only red, green, or blue light passes) to the photodiode. The photodiode changes the light intensity into a charge. The charge is converted into a voltage and passed to the column line. The brighter the light, the bigger voltage change we get on the column line.

Resetting the Pixel

Prior to acquiring an image each pixel in the imaging chip is reset. This is accomplished by driving the reset row line (*ResetN*) to a voltage, *VDDP*, greater than *VDD* + V_{THN} (with body effect). This turns M1 on and sets the voltage across the photodiode to *VDD*. This condition is called the *dark or reference level* of the pixel. We can then turn M3 on by driving *RowN* to *VDDP* and, with M2 behaving as a source follower, driving the column line to the reference voltage level, V_R. This is important because each pixel in an imaging array will have slightly different characteristics. The differences in the pixels can result in speckles in the image. We can subtract this reference level from the actual measured signal level to get an accurate idea of the light intensity applied to the pixel (to eliminate the pixel gain differences).

The point here is that our DSM circuit will have to sense, and store, a reference level, V_R, at the beginning of the sense.

The Intensity Level

After we have stored the reference level voltage V_R (on a capacitor), the *ResetN* signal goes low. This allows the light striking the reverse-biased photodiode (through the color filter) to generate electron-hole pairs that cause the voltage across the diode to decrease. After a time, the information stored on the gate of M2 (the decrease in the voltage across the photodiode) is sampled on the column line (assuming M3 is on). The time between

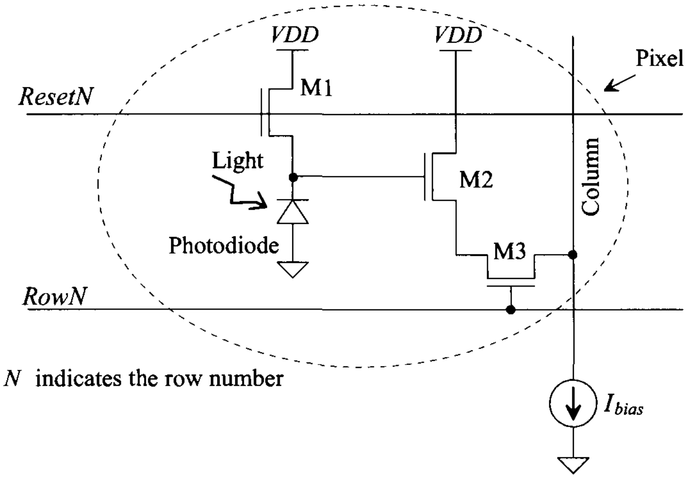


Figure 17.23 A CMOS active pixel sensor (APS).

the *ResetN* going low and the data (the column voltage, V_I, corresponding to the intensity of the light striking the photodiode) being sampled on the column line is called the *aperture* time. Note that a dark signal corresponds to a large voltage on the column line ($VDD - V_{THN}$), while a bright signal corresponds to a lower voltage (less than $VDD - V_{THN}$).

Sampling the Reference and Intensity Signals

When *ResetN* and *RowN* are high, the pixel's reference (or dark) voltage, V_R, is placed on the column line, Fig. 17.24. At this time the sample and hold reference signal, *SHR*, goes high and V_R is sampled onto a hold capacitor. Next, *ResetN* goes low and the photodiode changes light into charge. After the aperture time, the information from the photodiode (the intensity of the light) is on the column line, V_I. This voltage is then sampled and held on a hold capacitor when the signal *SHI* (sample and hold the intensity of the light) goes

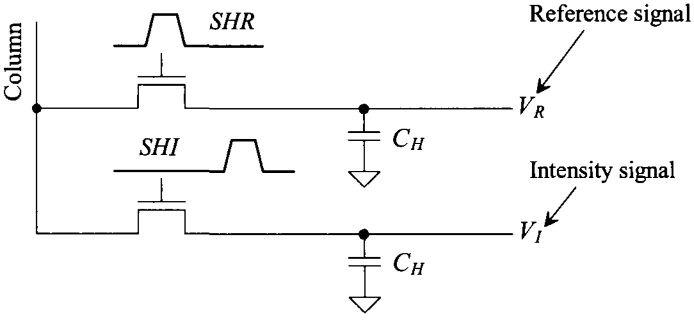


Figure 17.24 Sampling the reference and intensity signals.

high. What we want is to design a circuit that takes the difference in V_R and V_I and generates a digital number.

Noise Issues

Circuit noise limits the dynamic range of the sense, resulting in the blurring of the images or the inability to detect low light levels or distinguish between bright or high-intensity light levels. The major noise sources in CMOS, as discussed in Chs. 8 and 9, are flicker and thermal noises. The first design value we must select is the size of the hold capacitor in Fig. 17.24. The size of this capacitor limits the thermal noise in the sample. As seen in Table 8.1, using a hold capacitor of 1 pF results in an RMS noise voltage, just due to thermal noise, of 64 μV. This corresponds to a peak-to-peak voltage in the time domain of roughly 400 μV (six times the RMS value as seen in Fig. 8.33). Using a larger capacitor takes up more layout area and takes longer to charge but lowers the amount of thermal noise sampled onto C_H.

The output current of the pixel (when M3 is on and M2 is a source follower) also contains flicker noise. When this pixel is connected to the large hold capacitance and the capacitance of the bit line, the flicker noise current will be integrated. The result is a noise power spectral density with a $1/f^{\,3}$ spectral shape. As discussed in Ex. 8.14, the RMS value of the resulting noise signal will grow linearly with measurement time. What this means is that to *achieve a low noise* sample onto C_H we want to minimize the amount of time *SHR* and *SHI* are high *and* the time difference between the two signals. *SHR* and *SHI* should only go high long enough to charge C_H. The amount of time between M1 turning on and *SHI* shutting off should be as small as possible.

The amount of noise in V_R and V_I cannot be reduced after they are captured on the hold capacitors. For example, ideally V_I may be 0.5 V. However, because of noise V_I may be 0.501 V or 0.495 V, etc. We *may* be able to reduce the noise by averaging successful samples though (see Eq. [8.48] and Ex. 8.14).

Ideally, the sensing circuitry (the circuitry used to change the analog voltages, V_R and V_I, into a digital number) doesn't introduce any additional noise. When using the DSM, the counter can be thought of as a lowpass digital filter, Fig. 17.25. (See also the book entitled *CMOS Mixed-Signal Circuit Design* for much more detailed description of the frequency response of a counter.) As seen in this figure, if the sense time is the total number of times the DSM is clocked, *N*, multiplied by the clock's period *T*, that is $T_{meas} = NT$, then increasing the sense times lowers the bandwidth of the digital filter. This has the effect of lowering the noise bandwidth (so that the DSM sensing circuit does not contribute any further noise to the measured signal).

An example where a counter is used that doesn't result in filtering the measured signal is seen in Fig. 17.26. Here the column voltage corresponding to the intensity of the light is used as the – input to a comparator. For the + input, a constant current is used to charge a capacitor to generate a voltage ramp. When the ramp's voltage gets above V_I, the output of the comparator goes high and stops the counter. In this scenario the counter is simply used to give a digital representation of a time or ramp voltage. The counter doesn't provide any filtering (or more precisely averaging as used in DSM, see Tables 17.1 to 17.3). Notice also that this sensing topology very likely adds noise to the measured signal. For example, we know that using a constant current to charge a

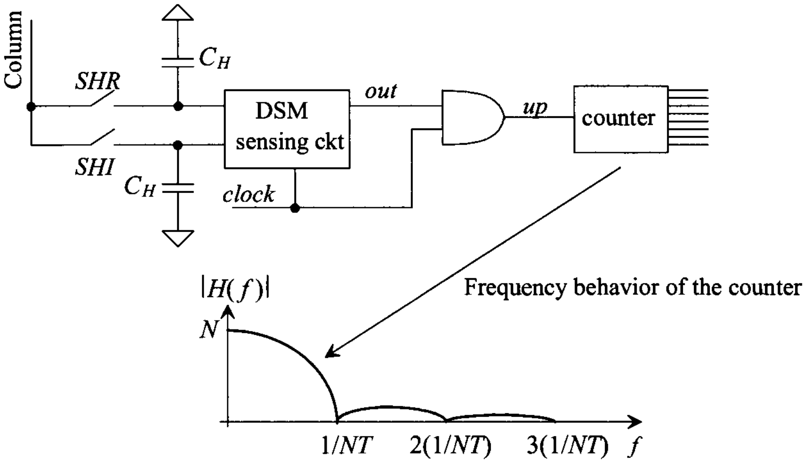


Figure 17.25 Thinking of the counter as a digital low pass filter

capacitor results in a voltage signal containing $1/f^3$ noise. The ramp, with noise, voltage is connected directly to the comparator and causes noise to be added to the measured signal. Further, if the comparator makes an error and switches states too early or too late because of an offset or noise coupling from the adjacent column sensing circuits, again, the sense adds noise to the measured signal (the digital output code isn't constant but rather moves around even though the inputs to the sense amplifier may be constant).

Also note that using the DSM, we can run the sense operation indefinitely, while the scheme in Fig. 17.26 is limited by the ramp rate. Further increasing the clock frequency in Fig. 17.26 won't increase the resolution if the sense is noise limited. In the DSM sensing circuit, increasing the clock frequency, as seen in Fig. 17.25, lowers the bandwidth of the digital filter and increases the resolution of the sense.

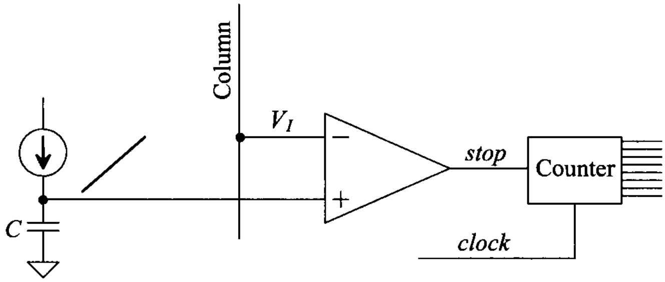


Figure 17.26 A circuit where the counter doesn't behave like a lowpass filter.

Subtracting V_R from V_I

As we mentioned earlier, each pixel has slightly different characterisitics. For example, the threshold voltage of M2 in one pixel may be 10 mV different from the threshold voltage of M2 in a different pixel. To remove this error, we subtract the measured reference voltage, V_R, from the measured signal intensity, V_I. It's important, during the sense, not to change these voltages with our sensing circuit. Let's convert these voltages to currents and then subtract the currents to get the difference (and not try to subtract the voltages directly).

Examine the voltage-to-current converter seen in Fig. 17.27. This circuit is simply a source follower that is made wide so that its V_{SG} is always approximately the threshold voltage. The relationship between the drain current and the column voltage V_{col} (V_I or V_R) is

$$I = \frac{VDD - V_{THP} - V_{col}}{R} \qquad (17.30)$$

for

$$V_{col} < VDD - V_{THP} \qquad (17.31)$$

Reviewing Fig. 17.23, we see that if the threshold voltage drop of M2 (with body effect) is more than V_{THP} (without body effect because we placed the PMOS device in Fig. 17.27 in its own well), Eq. (17.31) will always be satisfied.

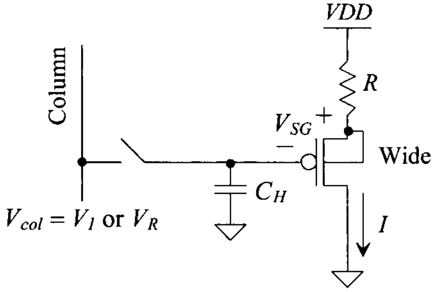


Figure 17.27 Linear voltage-to-current conversion.

Example 17.6

Using SPICE determine the linearity of the current in the circuit seen in Fig. 17.28. Use the short-channel CMOS process and compare the simulation results to hand caculations.

If VDD is 1 V, $V_{THP} = 250$ mV, and R is 10MEG, then

$$I = 75 - 100 \cdot V_{col}\, nA$$

Noting that the ideal slope of the line is $-1/R$ ($= -100 \times 10^{-9}$). Figure 17.29 shows the simulation results. In (a) we see from the transfer curve how the output, I, changes with the input, V_{col}. In (b) we take the derivative of I to see if the slope is perfectly flat. The linearity is pretty good (1%) for $V_I < 400$ mV. ■

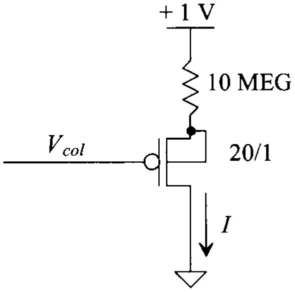
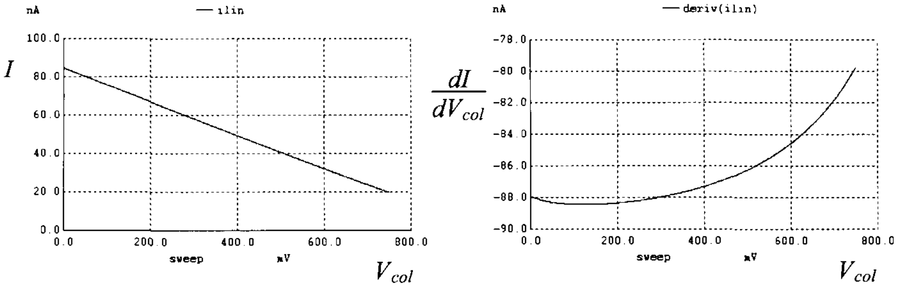


Figure 17.28 Circuit used in Ex. 17.6.



(a) Showing transfer curve

(b) Taking the derivative of (a)

Figure 17.29 Simulating the operation and linearity of the voltage-to-current converter in Fig. 17.28.

We might wonder, from the last example, if it would be a good idea to try to make the linearity of our voltage-to-current converter even better. Before doing this, let's look at the linearity of the source follower in the pixel itself. Figure 17.30 shows a simplified schematic of the source follower (M2) used in the APS. We've modeled the finite output resistance of the current source with a 10MEG resistor. The simulation results in Fig. 17.31 show the transfer relation of this circuit and its linearity. For column voltages between 100 mV and 600 mV, the linearity is 0.5% (which is comparable so we won't concern ourselves any further with trying to better the linearity at this point).

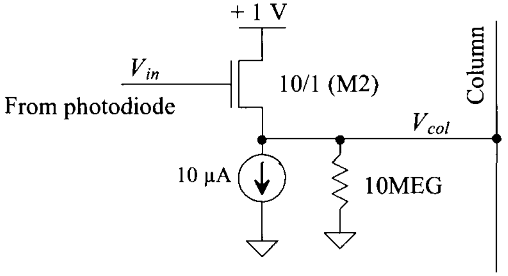


Figure 17.30 Looking at the linearity of the source follower in the pixel.

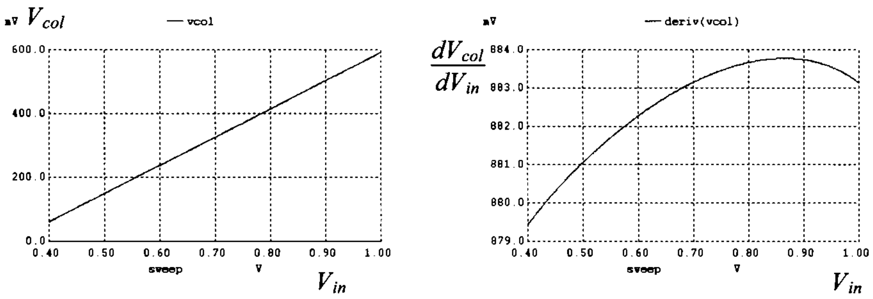


Figure 17.31 Showing the transfer curve and linearity of the circuit in Fig. 17.30.

To take the difference in the currents, let's use a current mirror as seen in Fig. 17.32. The current corresponding to the reference voltage is

$$I_R = \frac{VDD - V_{THP} - V_R}{R_R} = \frac{V_{R,shift}}{R_R} \tag{17.32}$$

and the current corresponding to the intensity of light is

$$I_I = \frac{VDD - V_{THP} - V_I}{R_I} = \frac{V_{I,shift}}{R_I} \tag{17.33}$$

The difference in these currents is summed (sigma) in the bucket capacitor, as seen in Fig. 17.32. When we add our comparator and feedback to form a DSM, the charge on the capacitor, averaged over time, is a constant. This occurs when

$$I_R = I_I = \frac{V_{R,shift}}{R_R} = \frac{V_{I,shift}}{R_I} \tag{17.34}$$

or

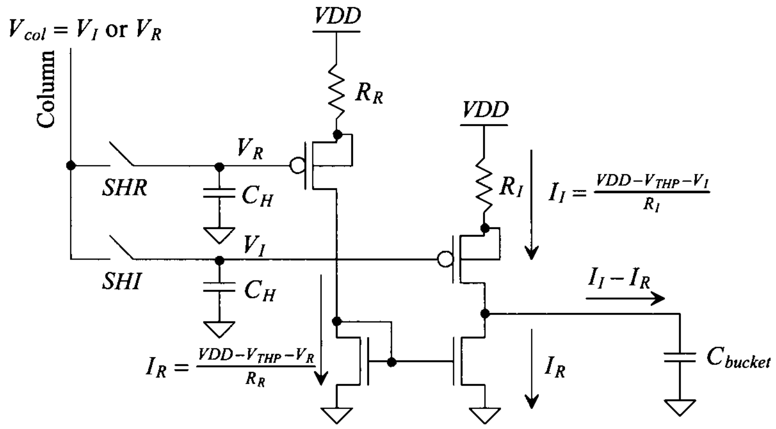


Figure 17.32 Subtracting the currents.

$$V_{I,shift} = \frac{R_I}{R_R} \cdot V_{R,shift} \tag{17.35}$$

The ratio of the resistances gives us the information we need to determine the relative (to the reference level, V_R) intensity of light on the pixel. To implement the resistors, let's use the switched-capacitor resistor as seen in Fig. 17.20. For the reference voltage we know

$$V_{I,shift} \geq V_{R,shift} \tag{17.36}$$

and so $(R_I \geq R_R)$. Let's use (M3 always on in Fig. 17.20)

$$R_R = \frac{1}{f \cdot C_{cup}} \tag{17.37}$$

and for R_I (which *will* be enabled via M3 when its gate goes low)

$$R_I = \frac{1}{f \cdot C_{cup} \cdot \frac{\overline{M}}{N}} \tag{17.38}$$

Rewriting Eq. (17.35), knowing $\overline{M} = N - M$, gives

$$V_{I,shift} = V_{R,shift} \cdot \frac{N}{N-M} \tag{17.39}$$

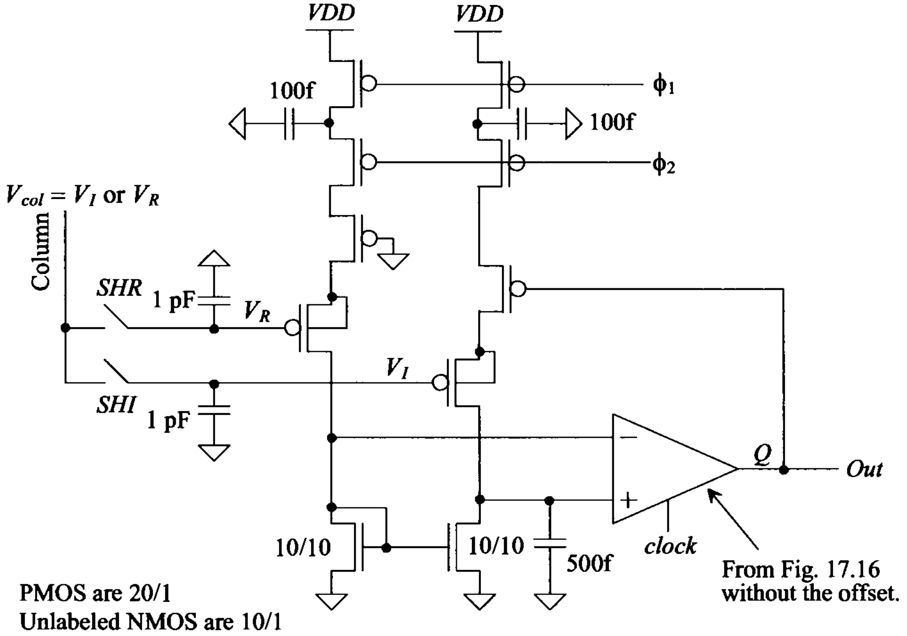
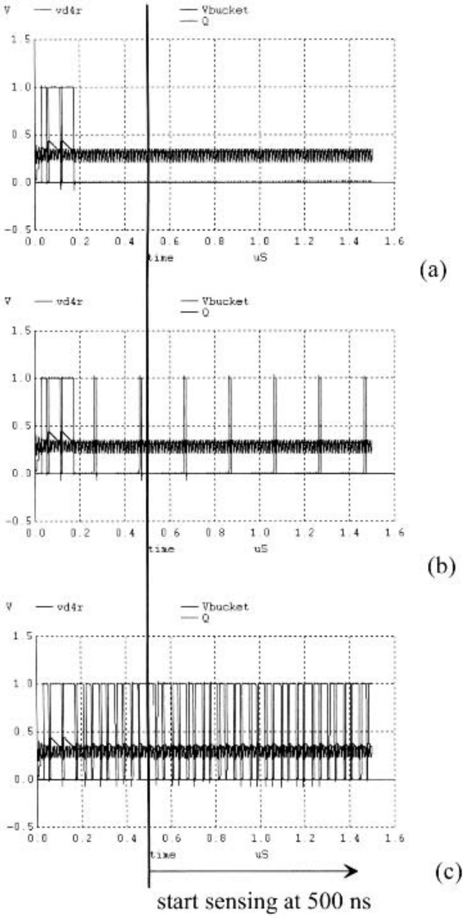


Figure 17.33 Schematic of a DSM for sensing in CMOS imaging chips.

If the DSM is clocked 1,000 times (*N*), then *M* can range from 0 (the pixel is not illuminated, that is, the dark or reference level) to 1,000 (very bright). Figure 17.33 shows a schematic of the sensing circuit with some typical values.

The circuit in Fig. 17.33 was made as symmetrical as possible so that power supply or ground noise affected each signal path equally. The MOSFETs in the current mirror are made long (10 drawn) so that the voltages on the input of the comparator are greater than the NMOS threshold voltage. The size of the capacitors isn't that critical. We want the C_{cup} capacitors to be less than the C_{bucket} capacitor. We don't have to worry about overcharging C_{bucket} in this scheme because the input signal contributions are limited by the switched capacitor resistors. (We only get one C_{cup} every clock cycle.)

Figure 17.34 shows the simulation results for the DSM circuit of Fig. 17.33. We assume that the counter, Fig. 17.25, is enabled after 500 ns. Prior to this time, the



$V_R = 650\ mV \qquad V_I = 650\ mV$

$V_{R,shift} = VDD - V_{THP} - V_R = 100\ mV$

$V_{I,shift} = VDD - V_{THP} - V_I = 100\ mV$

$V_{I,shift} = 100\ mV \cdot \dfrac{100}{100-0} = 100\ mV$ (sensed)

$V_R = 650\ mV \qquad V_I = 645\ mV$

$V_{R,shift} = VDD - V_{THP} - V_R = 100\ mV$

$V_{I,shift} = VDD - V_{THP} - V_I = 105\ mV$

$V_{I,shift} = 100\ mV \cdot \dfrac{100}{100-5} = 105\ mV$ (sensed)

$V_R = 650\ mV \qquad V_I = 400\ mV$

$V_{R,shift} = VDD - V_{THP} - V_R = 100\ mV$

$V_{I,shift} = VDD - V_{THP} - V_I = 350\ mV$

$V_{I,shift} = 100\ mV \cdot \dfrac{100}{100-72} = 357\ mV$ (sensed)

Figure 17.34 How the DSM sensing circuit in Fig. 17.33 operates.

reference and intensity signals are sampled onto the hold capacitors. If $V_R = 650$ mV, then

$$V_{R,shift} \approx 1 - 0.25 - 0.65 = 100\ mV$$

In Fig. 17.34a we apply the same signal to the DSM sensing circuit, that is 650 mV, for V_I. As expected, the output stays low for all times. In (b) we drop V_I to 645 mV and see, during the 500 ns to 1,500 ns sensing time, 5 output ones. As seen in the figure, the sensed value indicates the intensity is 5 mV below the reference. In (c) we drop the intensity signal to 400 mV resulting in $V_{I,shift} = 350$ mV. The sensed value with 72 of the 100 clock cycles being high is 357 mV. Let's look at what would happen if we sensed 71 ones

$$100\ mV \cdot \frac{100}{100-71} = 345\ mV$$

In either case (71 or 72 ones) the resolution was so coarse that we couldn't resolve the actual signal. If we think about this for a moment, we see that if the counter output code is small then the resolution of the measurement is better. For example, counter outputs of 1 and 2 result in

$$100\ mV \cdot \frac{100}{100-1} = 101\ mV \text{and } 100\ mV \cdot \frac{100}{100-2} = 102\ mV$$

Looking at Eq. (17.39), we see that the dependence on M (the number of times the output of the DSM goes high) is *not linearly related* to the light intensity, V_I. What we want is an equation like (17.7), that is,

$$V_{I,shift} = V_{R,shift} \cdot \frac{M}{N} \tag{17.40}$$

To get a relationship like this, we might try to control the value of R_R too as seen in Fig. 17.35. The complementary output of the comparator is fed back so that

$$R_R = \frac{1}{f \cdot C \cdot \frac{M}{N}} \tag{17.41}$$

and thus

$$V_{I,shift} = V_{R,shift} \cdot \frac{M}{N-M} \tag{17.42}$$

Again, we do not have a linear relationship. Further, half or more of our outputs must be used to enable the switched-capacitor resistor in series with the reference signal. If, for example, $V_{R,shift} = V_{I,shift}$, then $M = N/2$. Since $V_R \geq V_I$, then $0 \leq M \leq N/2$ (not good design).

If we review the derivations leading up to Eqs. (17.7) and (17.20), the common theme is that the feedback signal controlled by the comparator is a constant addition to the capacitive bucket (or bit line). In Fig. 17.18, for example, we used M4 to ensure that the charge from C_{cup} was a constant added to C_{bit}. In our current sense amplifier, Fig. 17.33, the signal we feedback is not a constant but rather a function of V_I. When sensing in a CMOS imager, the signal fed back should be a function of the reference level, V_R. The comparator's output should control a fed-back signal that is derived from V_R.

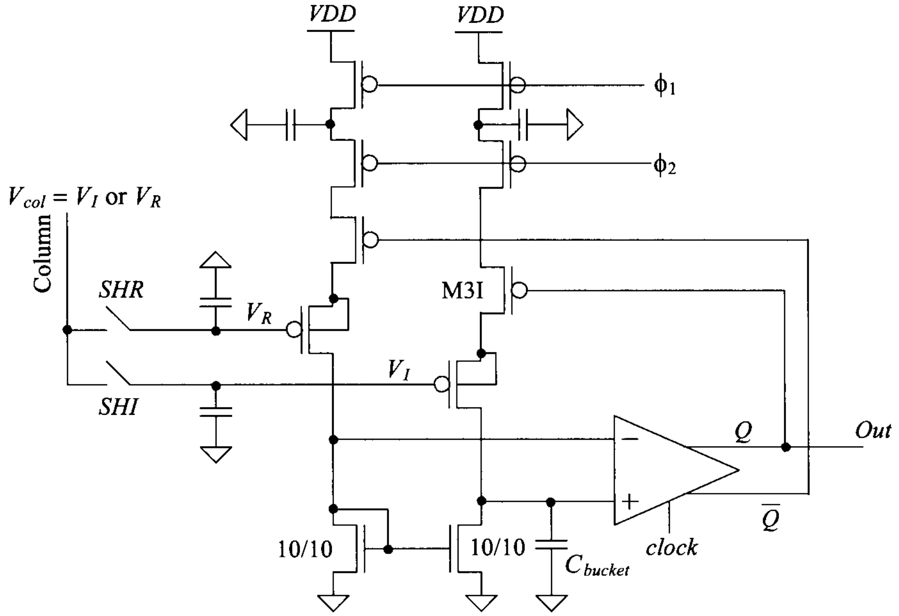
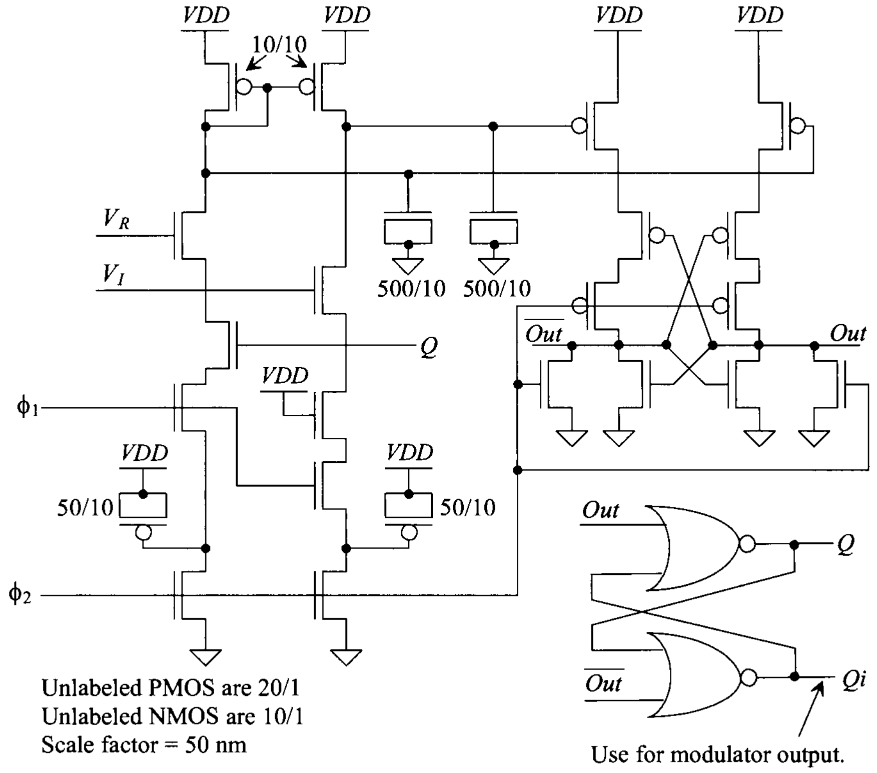


Figure 17.35 Using both comparator outputs for a DSM sensing circuit.

Note that it's a bad idea to ground the gate of M3I in Fig. 17.35 and only have a single feedback path. Since $V_R > V_I$, we won't be able to supply enough current through the reference signal path. The charge supplied by the intensity path will always be greater than the charge supplied from the reference path (and so C_{bucket} will overfill). A new topology is needed.

Figure 17.36 shows an NMOS version of Fig. 17.33. We've replaced the capacitors with MOSFETs to show that the DSM sensor can be implemented using a single-poly CMOS process. The PMOS devices are used for the "cup" capacitors. We use PMOS instead of NMOS because, for the topology seen, the PMOS devices always remain in strong inversion. The NMOS devices, for example, move towards accumulation mode when ϕ_2 turns on and discharges the capacitors. The result is a nonlinear capacitance (the size of the cup varies). Similarly, we use NMOS for the bucket capacitors because, for the topology used, they will always remain in strong inversion (the capacitance won't vary with the changes in the voltage on their gates). Note that a second "bucket" capacitor was added across the 10/10 diode connected (gate and drain tied together) PMOS device. This addition serves two purposes. The first is to ensure that ground noise affects the comparator inputs equally (noise on ground will feed evenly through the bucket capacitors to the input of the comparator). The second reason is that it smoothes out the summation of the currents. Note that if the added capacitor is too big, stability can be an issue (the added capacitor adds a delay in series with the feedback path).



Unlabeled PMOS are 20/1
Unlabeled NMOS are 10/1
Scale factor = 50 nm

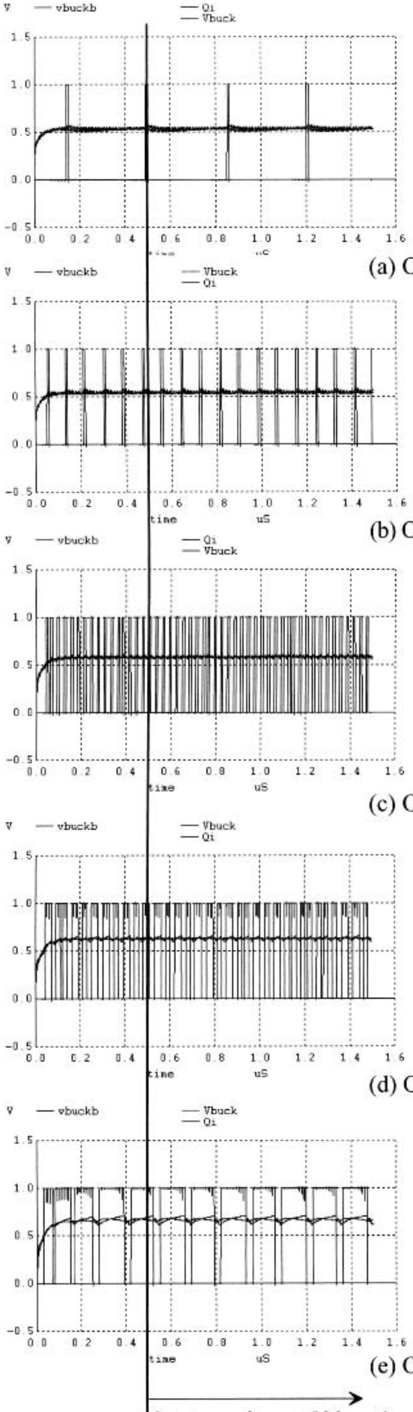
Use for modulator output.

Figure 17.36 DSM circuit for sensing in a CMOS imaging chip.

In Fig. 17.36 we show a comparator design using PMOS imbalance MOSFETs (as seen in Fig. 16.32 for the NMOS flavor). When ϕ_2 goes high, the outputs of the comparator, *Out* and $\overline{Out}$ are driven low. When ϕ_2 goes low, on the falling edge, the comparator makes a decision concerning which of the bucket capacitors has the higher potential across it. Based on this decision, the outputs of the comparator are latched with the NOR-based SR latch. The *Q* output is fed back to enable or disable the summation of charge via the reference path. We've used the *Qi* output as the DSM's output. When the intensity level is the same as the reference level, the output stays low for all times. It may be possible to eliminate this SR latch in a production part. However, we include it here because it makes the simulation results easier to look at (the glitches in the comparator's output are reduced).

Figure 17.37 shows some simulation results for the DSM circuit seen in Fig. 17.36. Notice, in this figure, that we've *assumed* the threshold voltage of the NMOS device is 250 mV. Since our reference level (the black level for the pixel) is 650 mV, our shifted reference level, $V_{R,shift}$, is 400 mV. For 100 samples then we can estimate the resolution as

$$V_{res} = \frac{V_{R,shift}}{N} \rightarrow 4\,mV \tag{17.43}$$



$V_R = 650\ mV \qquad V_I = 640\ mV$

$V_{R,shift} = V_R - V_{THN} = 400\ mV$

$V_{I,shift} = V_I - V_{THN} = 390\ mV$

$V_{I,shift} = 400\ mV \cdot \dfrac{98}{100} = 392\ mV$ (sensed)

(a) Output goes high 2 times out of 100 (M = 98).

$V_R = 650\ mV \quad V_I = 600\ mV$

$V_{I,shift} = V_I - V_{THN} = 350\ mV$

$V_{I,shift} = 400\ mV \cdot \dfrac{88}{100} = 352\ mV$ (sensed)

(b) Output goes high 12 times out of 100 (M = 88).

$V_R = 650\ mV \quad V_I = 500\ mV$

$V_{I,shift} = V_I - V_{THN} = 250\ mV$

$V_{I,shift} = 400\ mV \cdot \dfrac{65}{100} = 264\ mV$ (sensed)

(c) Output goes high 35 times out of 100 (M = 65).

$V_R = 650\ mV \quad V_I = 400\ mV$

$V_{I,shift} = V_I - V_{THN} = 150\ mV$

$V_{I,shift} = 400\ mV \cdot \dfrac{44}{100} = 176\ mV$ (sensed)

(d) Output goes high 56 times out of 100 (M = 44).

$V_R = 650\ mV \quad V_I = 300\ mV$

$V_{I,shift} = V_I - V_{THN} = 50\ mV$

$V_{I,shift} = 400\ mV \cdot \dfrac{24}{100} = 96\ mV$ (sensed)

(e) Output goes high 76 times out of 100 (M = 24).

Start sensing at 500 ns (enable the counter)

Figure 17.37 The operation of the DSM sensing circuit in Fig. 17.36.

Looking at the figure, we may think that there is a large nonlinearity in the sense because, for lower input voltages, $V_{I,shift}$, the sensed value doesn't exactly match the shifted value. However, notice that the input changes from, say, 650 mV to 600 mV we get a code difference of 12 (24/100 mV), or from 600 mV to 500 mV (23/100 mV), from 400 mV to 300 mV (20/100 mV). To get a better estimate for the resolution, let's use an average change of 23 counts per 100 mV to estimate the resolution, that is,

$$V_{res} = \frac{100\ mV}{23} = 4.35\ mV \tag{17.44}$$

and so the shift in the reference voltage can be more accurately predicted as

$$V_{R,shift} = 435\ mV \text{ because } (= N \cdot V_{res}) \text{ so } V_{THN} = 215\ mV \tag{17.45}$$

Using this, the sensed outputs in Fig. 17.37 can be rewritten as:

(a), $V_{I,shift} = 640\ mV - 215\ mV = 425\ mV$ and the sensed value $435\ mV \cdot \frac{98}{100} = 426\ mV$

(b), $V_{I,shift} = 600\ mV - 215\ mV = 385\ mV$ and the sensed value $435\ mV \cdot \frac{88}{100} = 382\ mV$

(c), $V_{I,shift} = 500\ mV - 215\ mV = 285\ mV$ and the sensed value $435\ mV \cdot \frac{65}{100} = 283\ mV$

(d), $V_{I,shift} = 400\ mV - 215\ mV = 185\ mV$ and the sensed value $435\ mV \cdot \frac{44}{100} = 191\ mV$

(e), $V_{I,shift} = 300\ mV - 215\ mV = 85\ mV$ and the sensed value $435\ mV \cdot \frac{24}{100} = 104\ mV$

Indicating a linear sense that becomes nonlinear at the edges of operation (when V_I becomes comparable to the V_{THN}).

It's important to understand the robustness of this sensing scheme. If the comparator makes a mistake, it is averaged out (comparator gain and offset aren't important). If noise is coupled into the sense amplifier, it will be averaged out. The sensing operation can be indefinite (noting that the hold capacitor voltages changing because of charge leaking off of the capacitors will ultimately limit the length of the sense). To increase the resolution of the sense, the clock frequency can be increased (noting the size of the counter used must be increased too). Finally, the topology requires little power. For the topology in Fig. 17.36 the current supplied by *VDD* is approximately 25 μA. If 1,000 of these sense amplifiers are used at the same time (say on the bottom of an imaging array with 1,000 columns), then the current required from *VDD* is 25 mA. The current can be further reduced by designing the DSM without the NOR latch seen in Fig. 17.36 or by using smaller capacitors.

The input-referred thermal noise is set by the sampling capacitors and is characterized using *kT*/*C* (see Table 8.1) and the associated discussion. We might think that using smaller capacitors results in an increase in the thermal noise. However this noise is averaged by *N*, the number of clock cycles (the counter), so we can rewrite Eq. (8.24) to show the decrease in the thermal noise with averaging as

$$V_{onoise,RMS} = \sqrt{\frac{kT}{NC}} \tag{17.46}$$

Sensing Circuit Mismatches

The point of sampling the reference or dark level and then subtracting the desired signal (the intensity of light on the pixel) was to subtract out mismatches in the pixel. For example, M2 in one pixel, may have a threshold voltage of 250 mV, while in a different pixel the threshold voltage may be 230 mV. The result, without the subtraction, would be two pixels with different output voltages even though the light applied to each is exactly the same. After thinking about this for a moment, we might realize that the DSM sensing circuit, having two separate paths for the reference and intensity signal paths, will also be subject to a mismatch. If, for example, one sensing circuit on the bottom of a column in an imaging chip has different characteristics than the sensing circuits directly adjacent to it, then the image will show vertical streaks. We might try, using layout techniques, to reduce the mismatch. However, the human eye is very perceptive and will likely detect any differences in the sense (especially if the image is a single color). This is why the majority of imaging chips used a single pipeline ADC (see Ch. 29) operating at a high conversion rate at the time of this writing. Each pixel sees the exact same sensing circuit.

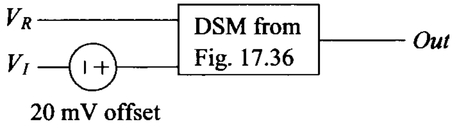


Figure 17.38 Modeling the differences in the signal paths in the DSM of Fig. 17.36 with a voltage in series with the intensity signal path.

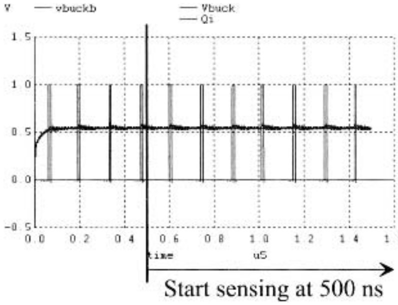


Figure 17.39 How a 20 mV offset changes the outputs in Fig. 17.37a.

To illustrate this problem, let's resimulate the DSM in Fig. 17.36 with an offset, as seen in Fig. 17.38. This offset voltage, which is an unknown that may be positive or negative, simply models a random difference in the signal paths. Using the input values seen in Fig. 17.37a and a 20 mV offset, we get the simulation results seen in Fig. 17.39. Instead of getting two outputs going high, we now get seven.

To reduce the effects of mismatch on the sense, consider, halfway through the sense time, switching the inputs to the DSM, as seen in Fig. 17.40. By switching the inputs halfway through the sense, the effects of path mismatch average, ideally, to zero.

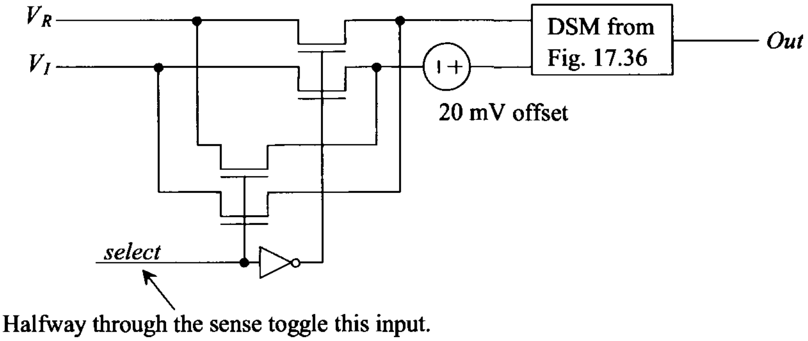
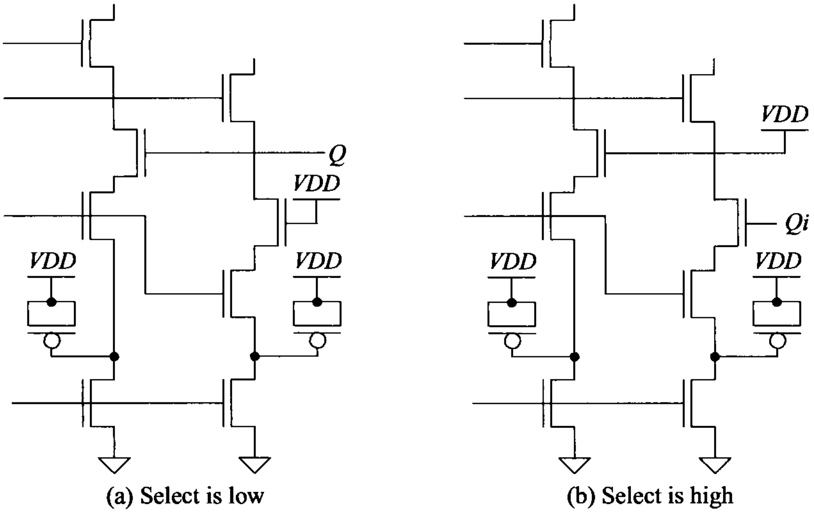


Figure 17.40 Switching the inputs of a DSM to eliminate path mismatch.



(a) Select is low

(b) Select is high

Figure 17.41 The change in the feedback with change in select.

Note that by switching the inputs we also have to change the feedback internal to the DSM. As seen in Fig. 17.36, we must switch the gate connections of Q and *VDD* in series with the switched capacitor resistors, Fig. 17.41, (the outputs of the NOR latch are switched).

Note at signal intensities *close to the reference*, V_R , that the averaging will not work out with just an up counter (as seen in Fig. 17.3). The signal, V_I, must move away from the reference by more than V_{OS}. For example, if we swap the inputs to the DSM at one μs in the simulation seen in Fig. 17.39, then the output of the DSM after 1 μs is always low and the output code is 3 (the ideal output code from Fig. 17.37a is 2). This (the averaging won't work unless the $|V_R - V_I| < V_{OS}$) shouldn't be a problem since the output codes at these levels correspond to dark signals. Again note that feeding back both Q and Qi , Fig. 17.35, results in a nonlinearity, as seen in Eq. (17.42).

Finally note that whenever starting or switching the inputs during a sense operation there will be a start-up transient (see Fig. 17.34 for example). What this means is that the counter should be disabled at the beginning of the sense or in the middle (if the inputs to the DSM are swapped, as seen in Fig. 17.40, halfway through the sense operation). Not disabling the counter during these times can result in sensing errors.

ADDITIONAL READING

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

[2] R. J. Baker, "Method and system for reducing mismatch between reference and intensity paths in analog to digital converters in CMOS active pixel sensors," US Patent 7,515,188, April 7, 2009.

[3] J. Taylor and R. J. Baker, "Method and apparatus for sensing flash memory using delta-sigma modulation," US Patent 7,366,021, April 29, 2008.

[4] R. J. Baker, "Per column one-bit ADC for image sensors," US Patent 7,456,885, November 25, 2008.

[5] R. J. Baker, "Resistive memory element sensing using averaging," US Patent 6,504,750, January 7, 2003.

PROBLEMS

17.1 Regenerate Table 17.1 in which the water level where a cup of water is removed is 4.7 instead of 5. How are the results affected? If the sensing time is increased, how are the final results affected (compare a water level of 5 against 4.7).

17.2 Generate a table, similar to Table 17.1, for the situation seen in Fig. 17.4 if the amount of water leaving the bucket is 0.3 cups per 10 seconds.

17.3 Rederive Eqs. (17.3) – (17.8) if I_{cup} in Fig. 17.6 is replaced with a resistor. Assume that the clock frequency is large (why?) to simplify the equations. What is the requirement for the current through the resistor when it is connected to the bit line in terms of the maximum bit current, I_{bit}.

17.4 Using SPICE simulations demonstrate that the error because of parasitics, as seen in Fig. 17.7, is reduced by connecting the switch to a 1 V source instead of ground. Illustrate, with drawings, what is happening.

17.5 Show, using simulations, that if the output of the comparator swings from *VDD* to *VDD*/2 we can eliminate M4 in Fig. 17.9 and still have $Q_{cup} = C_{cup} \cdot (VDD - V_{REF} - V_{THP})$. Why? Does the amount of current supplied by *VDD*/2 increase? Could this be a problem?

17.6 Show how the incomplete settling seen in Fig. 17.13 can be made more complete by reducing the clock frequency or increasing the width of M4.

17.7 Demonstrate, using SPICE and discussions, that the circuit in Fig. 17.42 may be used in place of the comparator and M3 in Fig. 17.18. How do output glitches affect the sensing circuit's operation?

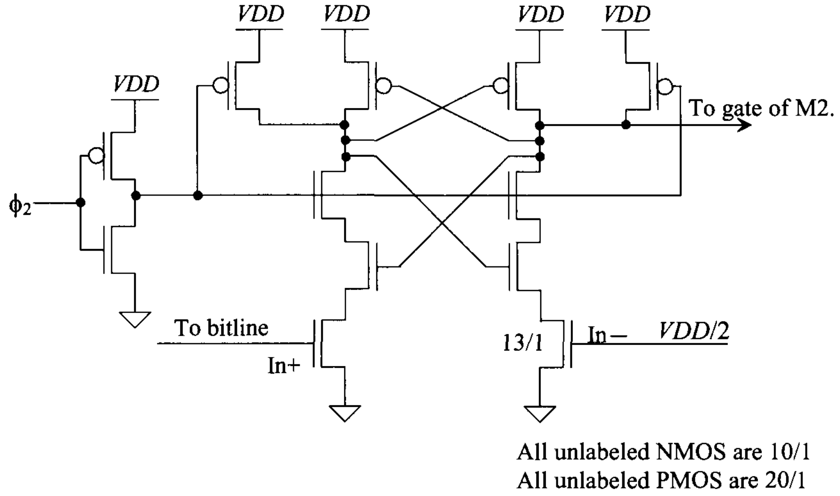


Figure 17.42 Simplifying the comparator in Figs. 17.16 and 17.18.

17.8 Design a DSM sensing circuit that will determine the value of a resistor that may range from 100k to 10 MΩ. Simulate your design with SPICE and comment on the design trade-offs concerning operating frequency and resistor (both the sensed and, if used in the DSM, the feedback resistance) changes with process variations or temperature.

17.9 Suppose that it is desired to have a noise floor of 100 μV RMS in a CMOS imager. Further suppose that the sensing circuit doesn't contribute any noise to the sense (the transformation from the analog column voltage to a digital word). Estimate the size of the hold capacitors used to sample both the reference and the intensity signals.

17.10 Using simulations, determine if the linearity of the voltage-to-current converter can be made better by adjusting the length and width of the PMOS device seen in Fig. 17.28. Why does, or doesn't, the performance get better?

17.11 Using the short-channel CMOS devices determine the average current that flows in the circuit seen in Fig. 17.43. The clocks are nonoverlapping (never low at the same time) as used throughout the chapter and have a frequency of 100 MHz. Verify your answer using SPICE.

17.12 What happens, to the simulation results seen in Fig. 17.34, if the time step used in the transient simulation is increased to 1 ns? How do the nonoverlapping clocks look with this time step?

17.13 Simulate the operation of the circuit seen in Fig. 17.44. Do the comparator outputs make full logic transitions? Are glitches a concern? Why? How do the simulation results compare to the results seen in Fig. 17.37? Note that the outputs of the comparator go low each time ϕ_2 goes high so that $\overline{Out}$ can be used to clock a counter directly.

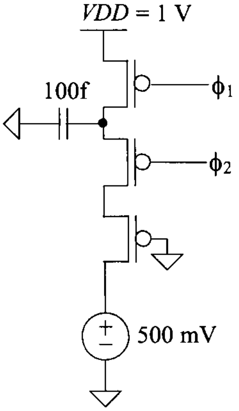
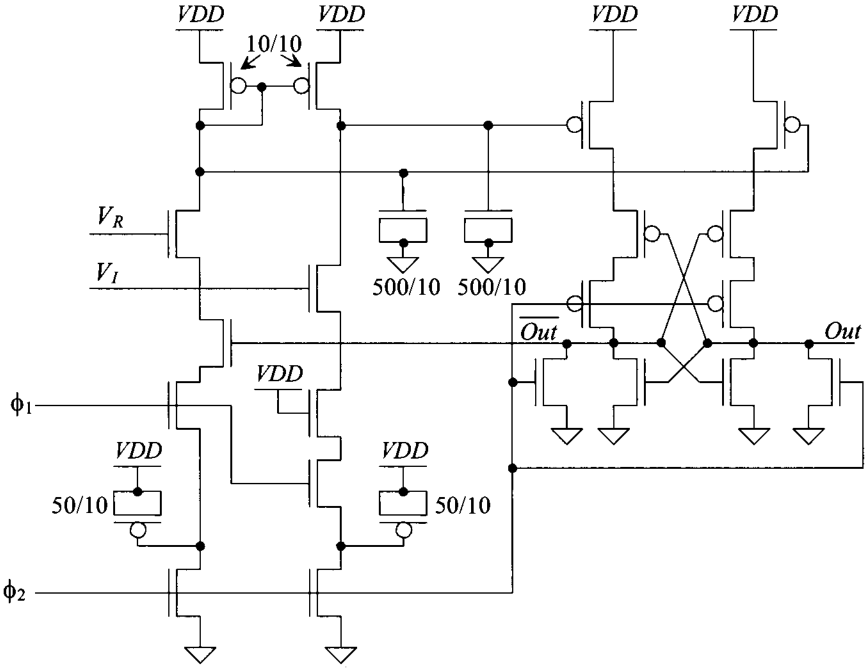


Figure 17.43 Determining the average current that flows in a switched-capacitor resistor. See Problem 17.11.



Unlabeled PMOS are 20/1
Unlabeled NMOS are 10/1
Scale factor = 50 nm

Figure 17.44 Simplifying the DSM sensing circuit, see Problem 17.13.