BOISE STATE UNIVERSITY

Integrating ADCs: A Tutorial

ECE614 - Advanced Analog IC Design
Presented by Antonio Oblea
May 19, 2008

Tutorial Outline

- Integrating ADC Overview
- Single-slope ADC
  - Operation
  - Analysis
  - Implementation/Simulation
  - Errors
- Dual-slope ADC
  - Operation
  - Analysis
  - Errors

Integrating ADC Overview

- The integrating ADC is also known as the ramp and slope ADC. In any case, the conversion is based on integration of a voltage, either a reference voltage or a reference voltage and an input voltage.
- There are different flavors of the integrating ADC, the single-, dual-, triple-slope, etc. Adding another “slope”, increases accuracy at the cost of conversion time.
- Integrating ADCs are used mostly in sensor applications and devices like voltmeters and ammeters, where precision is valued over speed.

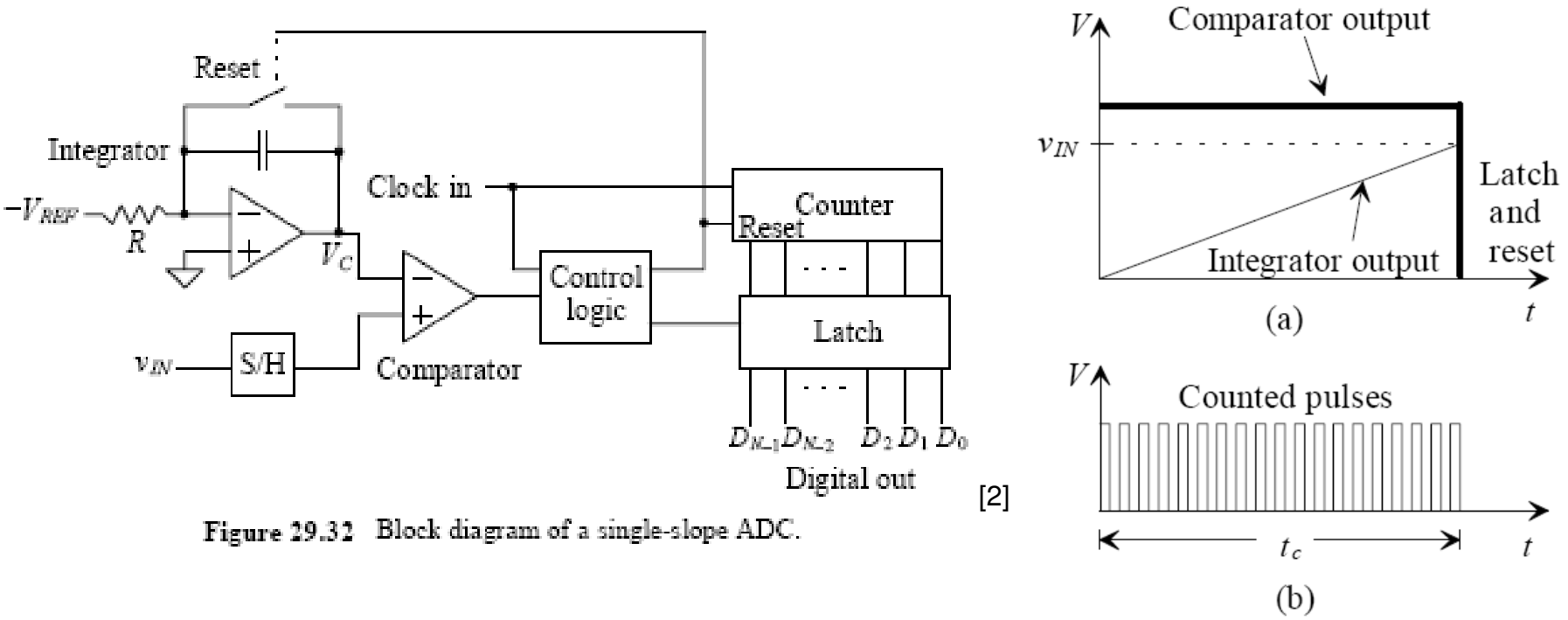
Strengths and Weaknesses

- The integrating ADC is a high-resolution converter, ranging from 12 bits and up.[1]
- These ADCs are highly linear. Essentially, the input is compared to an integrated reference voltage to determine the output, so the linearity will be based on the precision of the comparator.
- There are relatively few components used to implement these devices, thus the circuitry is relatively simple and low-cost to produce.[2]

- The main drawback of the integrating ADC is the slow conversion speed. As will be discussed, the output can take as long as 2^N clock cycles to convert a single value!

[1] W. M. C. Sansen, *Analog Design Essentials*, Springer, 2006
[2] R.J. Baker, *CMOS: Circuit Design, Layout, and Simulation, 2nd ed.,* Wiley-IEEE Press, 2005

Single-slope ADC Operation

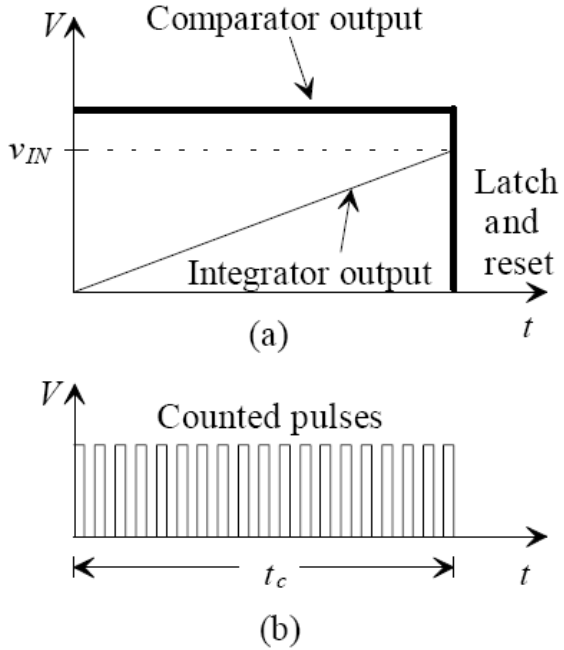


[2]

Figure 29.32 Block diagram of a single-slope ADC.

The comparator checks the value of the input voltage with that of the integrated reference voltage, V_{REF} (note we make V_{REF} negative since it is connected to the inverting input of the op-amp). At the same time the number of clock cycles is being counted. When the integrator output equals v_{IN} the comparator outputs a logic '0', triggering the counter and integrator to reset and the latch to hold the digital output.

Single-slope ADC Analysis



$$t_c = \frac{v_{IN}}{LSB} \cdot T_{CLK} = \frac{v_{IN}}{V_{REF}} 2^N \cdot T_{CLK} \qquad \text{eq. (1)}$$

(see why it's slow?)

Notice that we are keeping track of the time it takes to reach the value v_{IN}, by counting each LSB per clock period, T_{CLK} (remember the clock feeds directly into the counter and so each clock period is a LSB). And realizing LSB = $V_{REF} / 2^N$, we can derive equation (1) for t_c.

Examining equation (1) we can see why the integrating ADC is slow; for the worst-case, when $v_{IN} = V_{REF}$, we have to wait for 2^N clock periods. If N = 20 (20 bits) and T_{CLK} = 1 us. This means we'd have to wait for ~ 1 s for a conversion! This is quite slow when talking ADC.

Single-slope ADC Analysis (cont.)

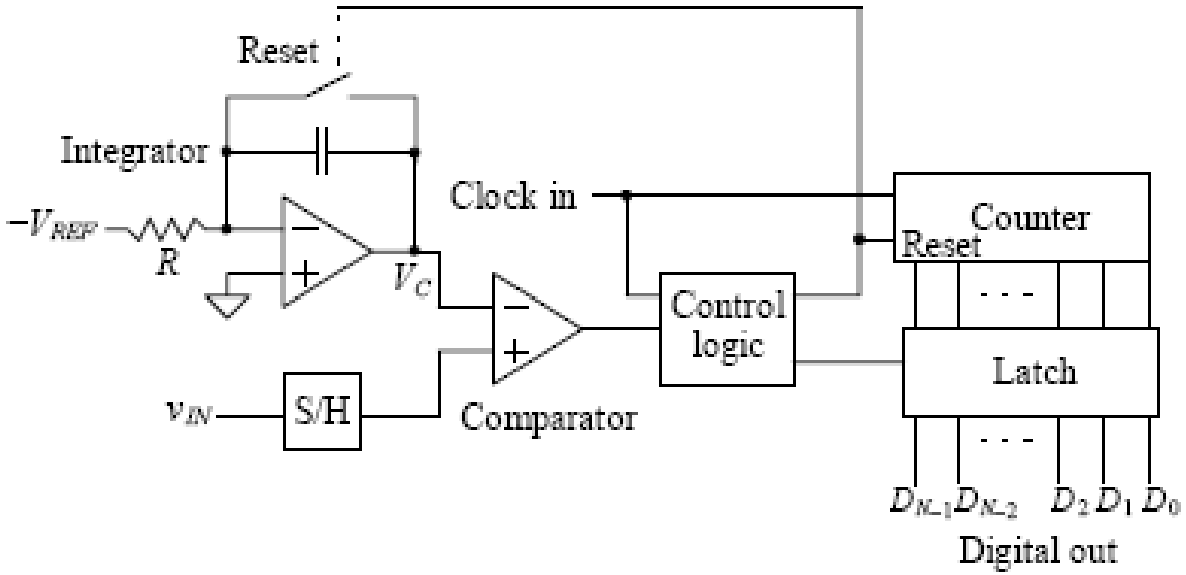


Figure 29.32 Block diagram of a single-slope ADC.

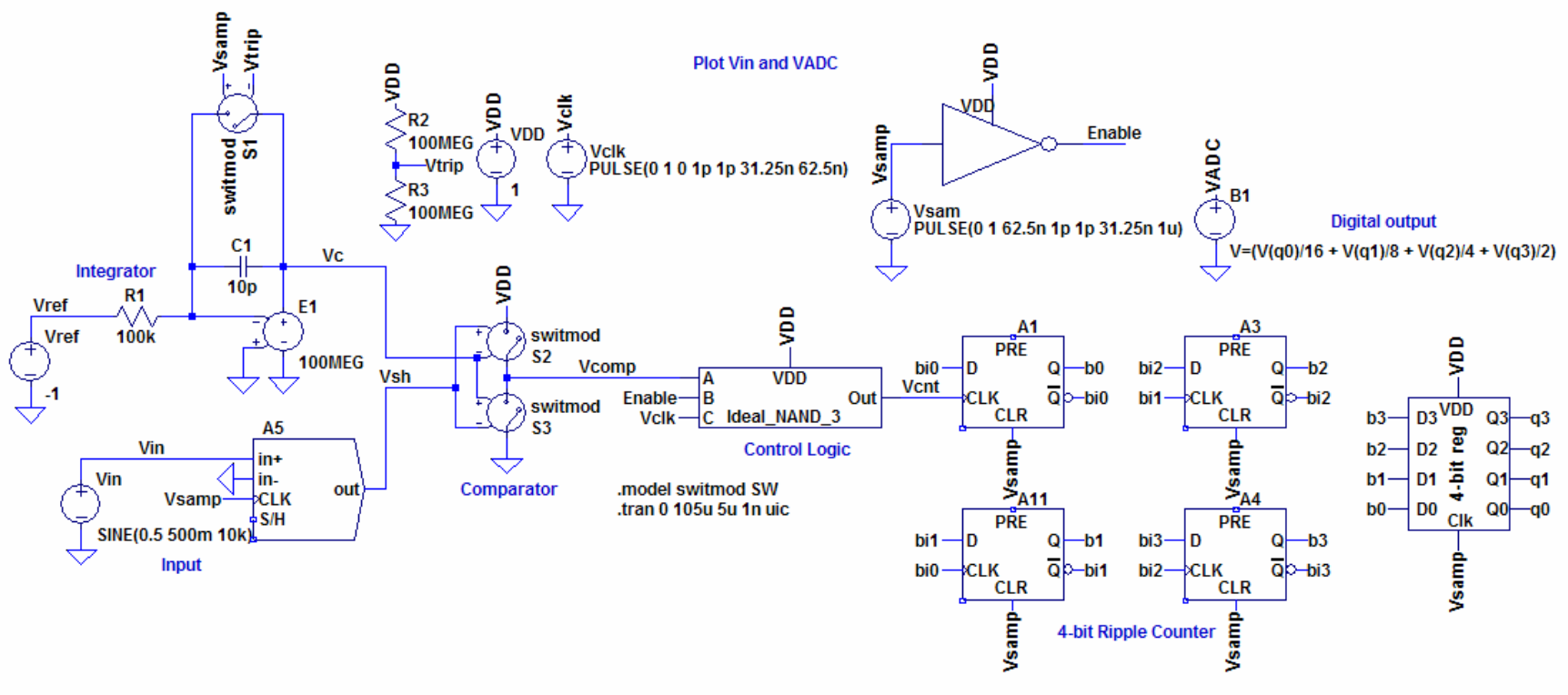
Performing KCL at the inverting input of the op-amp (note it is at virtual ground), we can determine V_C, equation (2).

Substituting in equation (1) we find an direct expression for V_C, equation (3).

$$\frac{0-(-Vref)}{R} = C\frac{dVc}{dt} \rightarrow Vc = \frac{1}{RC}\int_0^{t_c} Vref \cdot dt = \frac{Vref \cdot t_c}{RC} \qquad \text{eq. (2)}$$

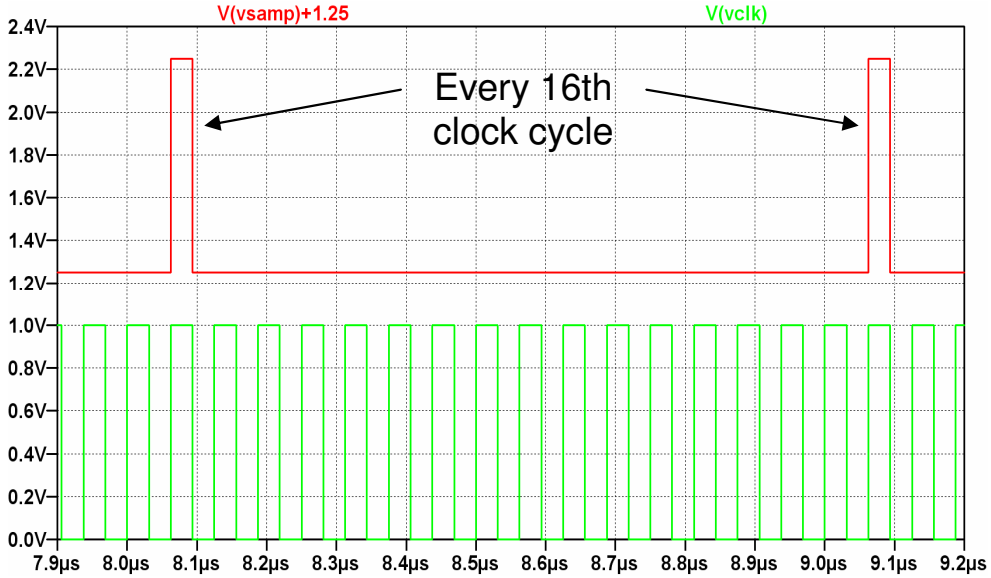
$$Vc = \frac{V_{REF} \cdot t}{RC} = \frac{v_{IN}}{RC \cdot f_{CLK}} 2^N \qquad \text{eq. (3)}$$

Single-slope ADC Implementation



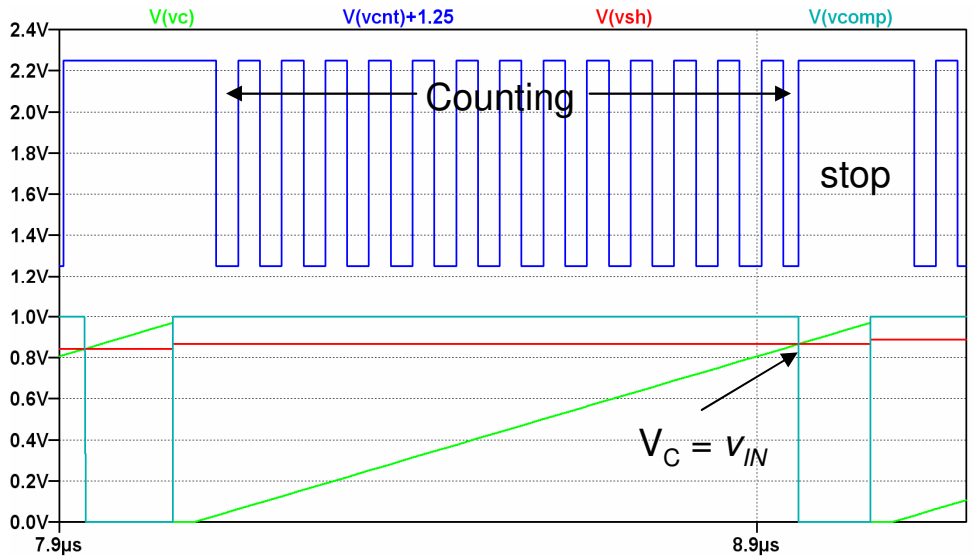
This schematic shows a slightly different method of implementing a single-slope ADC. In this implementation, V_C still ramps to the sampled v_{IN}, causing V_{COMP} to drop. However, when this happens the control logic does *not* reset the counter or latch the output. Instead, with $V_{COMP} = 0$, V_{CNT} stays high and the counter stops. The counter and V_C are then reset and the register is latched at a *pre-defined* time interval, V_{SAMP}, of 16 (or 2^N) clock cycles.

Single-slope ADC Implementation (cont.)

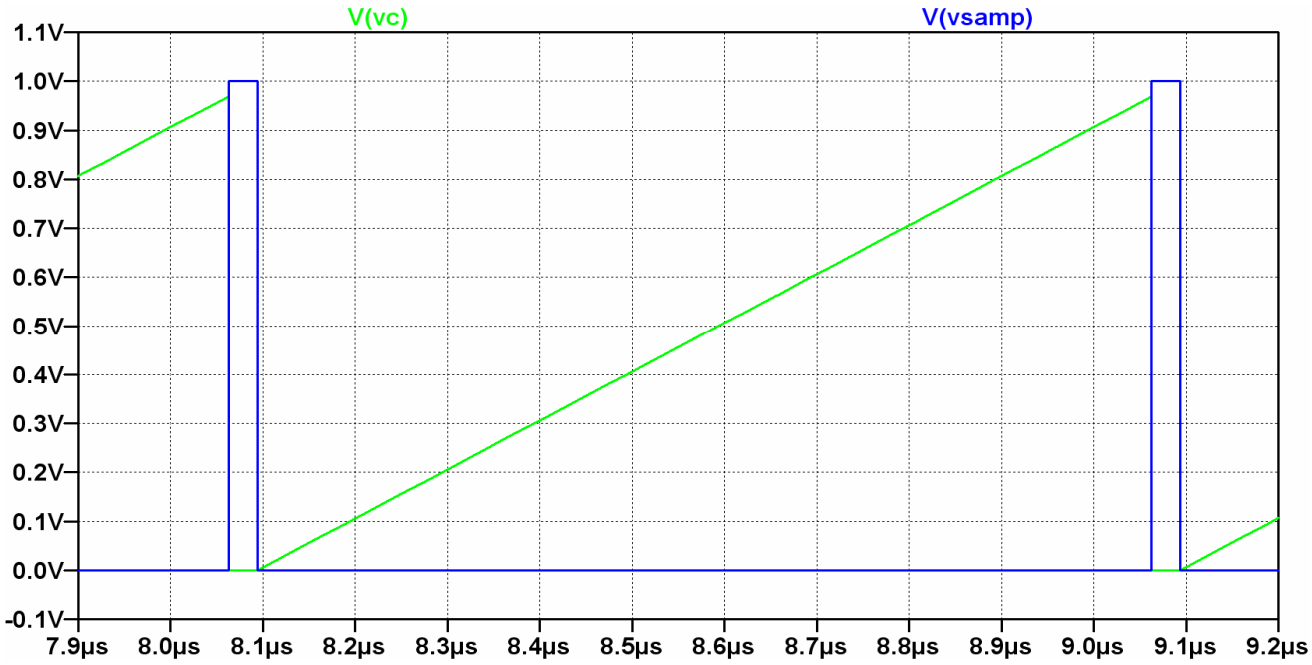


The waveform shows V_{SAMP} and V_{CLK}. The period for V_{SAMP} is 2^N*T_{CLK} to ensure if $v_{IN} = V_{REF}$ then the counter can count to 2^N before it is reset, equation (1).

Keep in mind that we are still just counting the number of clock cycles it takes for V_C to reach v_{IN}. After $V_C = v_{IN}$, the counter stops and waits for V_{SAMP} to reset.



Single-slope ADC Implementation (cont.)

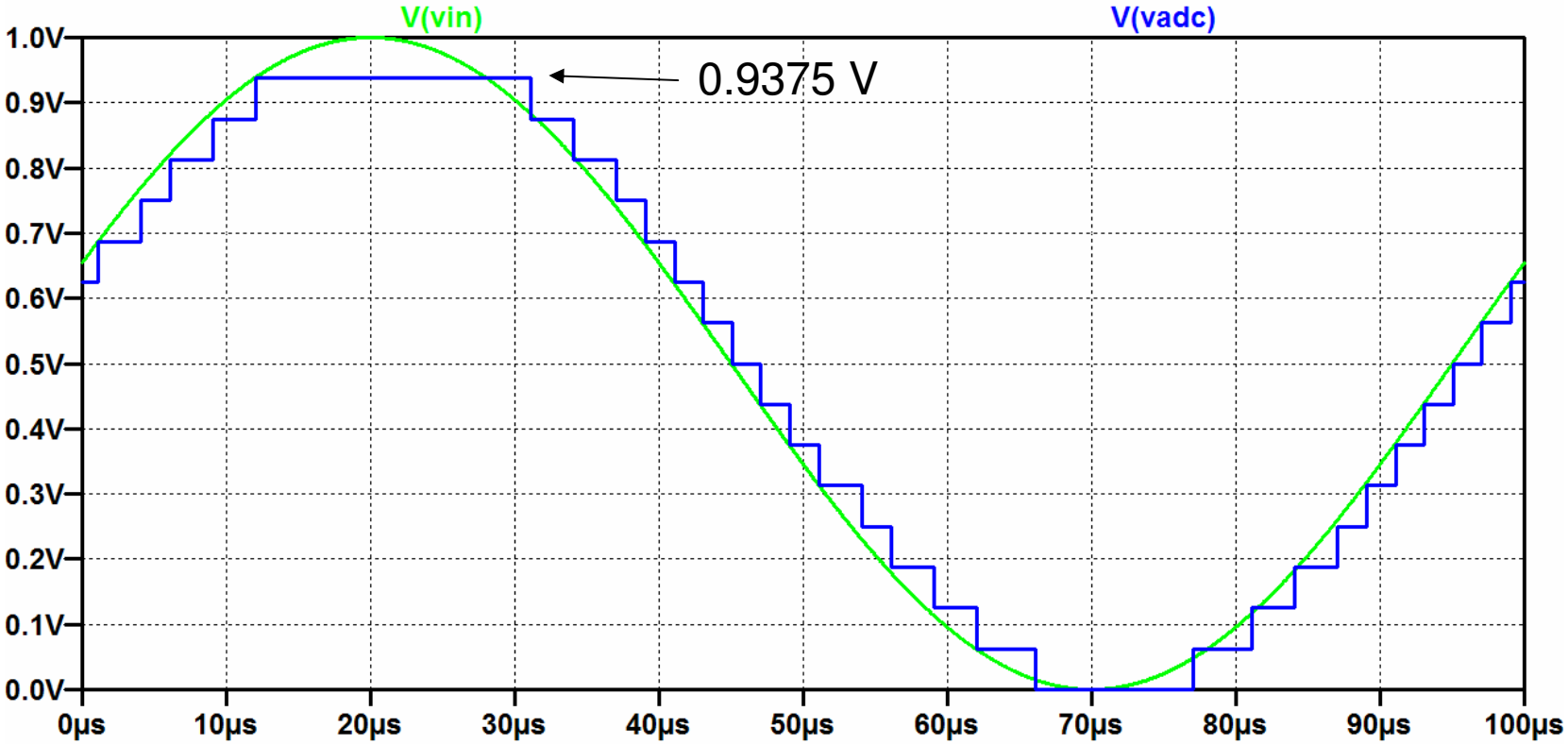


If we want $V_C = V_{REF}$ at the *end* of our conversion time, t_C, then by equation (2):

$$Vc = \frac{Vref \cdot t_c}{RC} \rightarrow RC = t_c$$

where t_C is 1us (period of sample voltage). Using a decently sized 10pF capacitor, the noise on the output of the integrator can be reduced, since the noise voltage is inversely proportional to the square of the capacitance (see pages 245-246 of [2]). This mandates R = 100kΩ.

Single-slope ADC Simulation



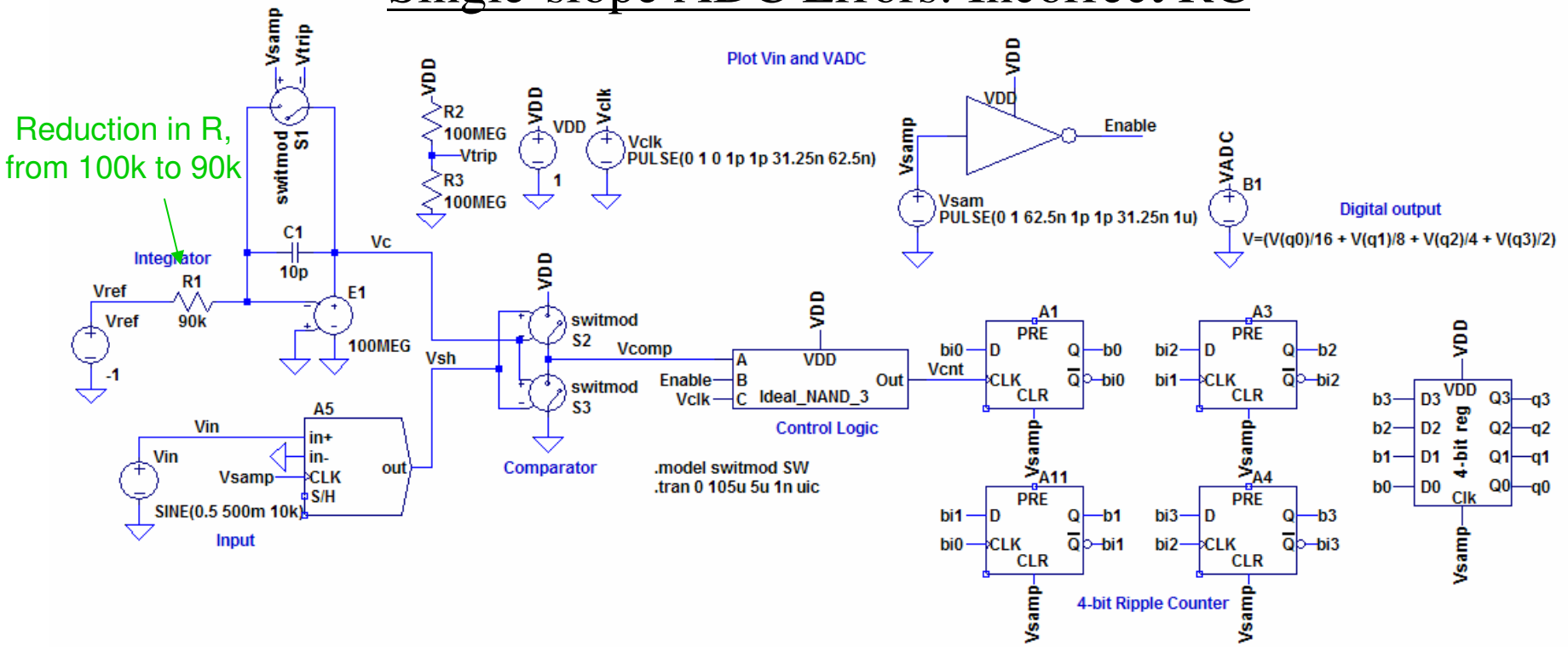
The waveform shows the analog input and the ADC output. Why does the output only go to 15/16 or 0.9375 V (Hint: How many bits are there?). The interested reader is encouraged to run (and investigate) the file Singleslope_IntegratingADC.asc.

Single-slope ADC Errors

- There are several sources of error in the (non-ideal) single-slope ADC, including: clock jitter, imprecise capacitor and resistors (refer to equation (3) to see how these affect V_C, which in turn affects the output), offsets in the op-amp, input sample-and-hold or comparator.

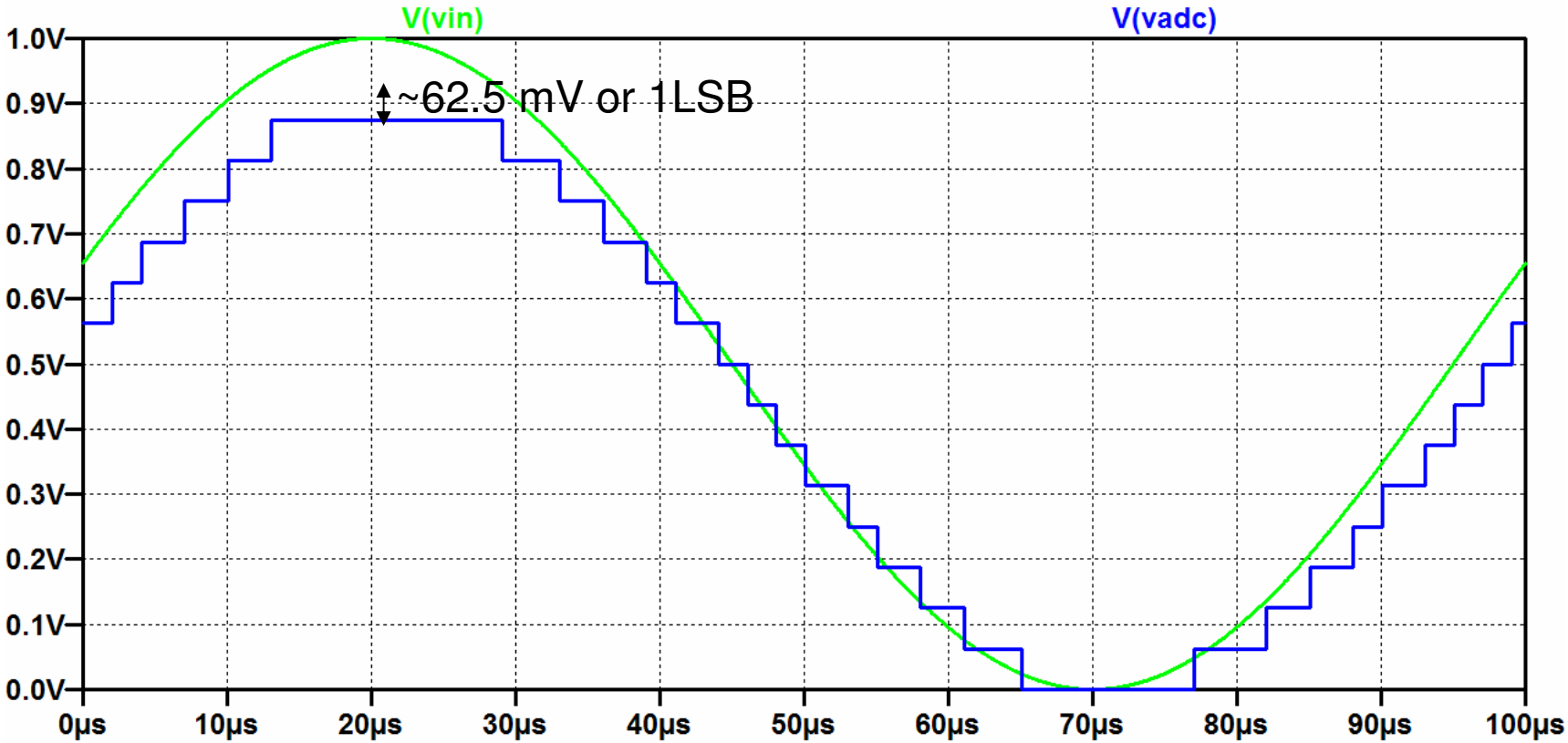
- The following examples will show how an imprecise RC or an offset in the comparator can negatively affect the conversion. The interested reader is encouraged to investigate the other sources of error independently.

Single-slope ADC Errors: Incorrect RC



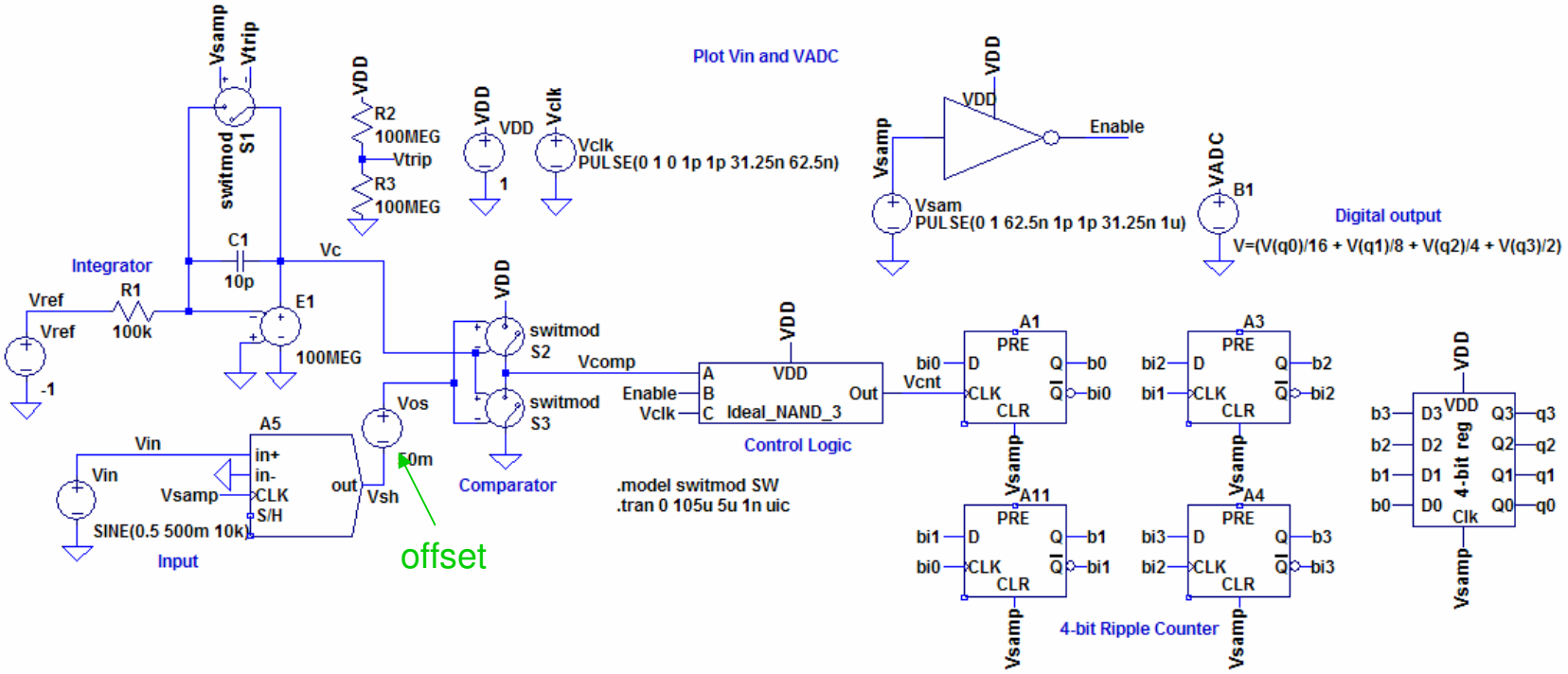
Several factors influence the actual value of capacitors and resistors. For capacitors, parasitics between different layers and temperature coefficients in the poly layer can result in capacitors that differ from the nominal value. For resistors, variations in sheet resistance (as much as 20%[2]), temperature and voltage coefficients will alter the value of a resistor. In the schematic above, R1 has been reduced by 10%, resulting in a RC time constant of 900 ns. With a lower time constant, V_C will ramp faster, equation (3), resulting in less clock cycles counted and a lower digital output.

Single-slope ADC Errors: Incorrect RC (cont.)



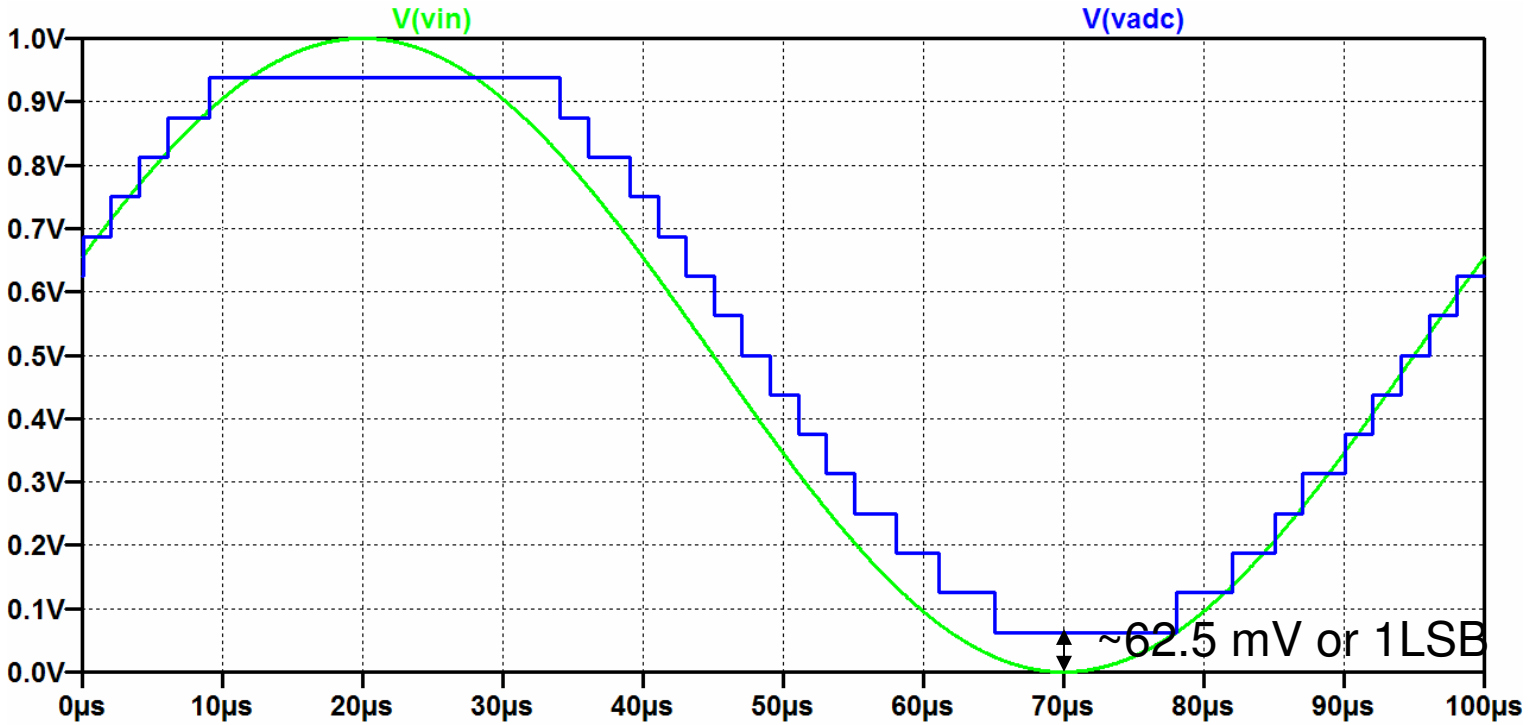
The simulation above shows the errors due to a reduction in the RC time constant. As shown, there is an error of approximately 1 LSB (i.e. V_C reached v_{IN} one clock cycle faster than anticipated). The simulation Singleslope_IntegratingADC_RC.asc can be used to investigate the influence of the RC time constant. What happens if R is reduced by 10% and C is raised by 10%? (Hint: Are R and C the same value?)

Single-slope ADC Errors: Comparator offset



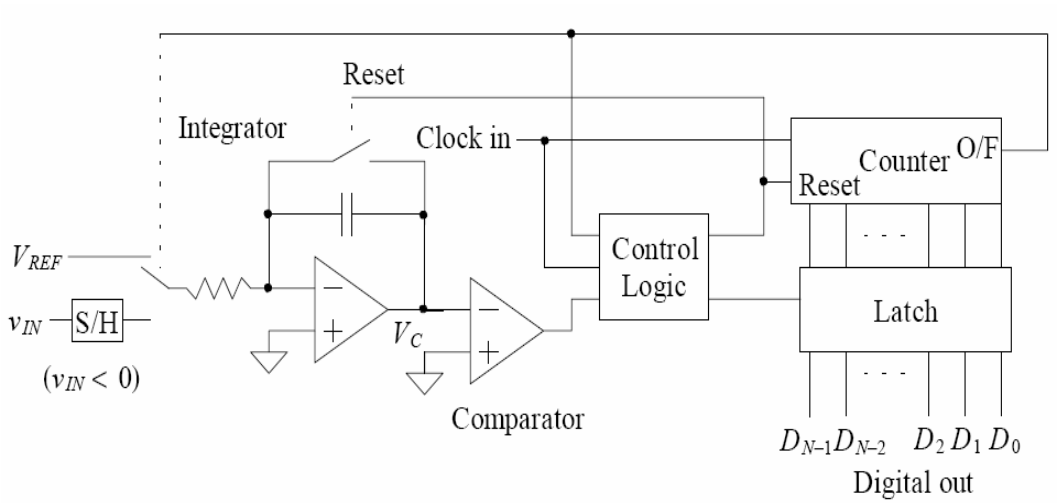
The inputs of a non-ideal comparator are tied to MOSFET gates of a differential-amplifier (see chapter 27 in [2]). Mismatches in the length, width, and threshold voltage (due to process variations such as lateral diffusion, oxide encroachment, and doping variations) result in a current, i_d, or transconductance, g_m, mismatch. In MOSFET small-signal analysis $i_d = g_m v_{gs}$. So the mismatches induce an offset on the gate voltage, modeled above as V_{OS} (review page 725 in [2] if necessary).

Single-slope ADC Errors: Comparator offset (cont.)

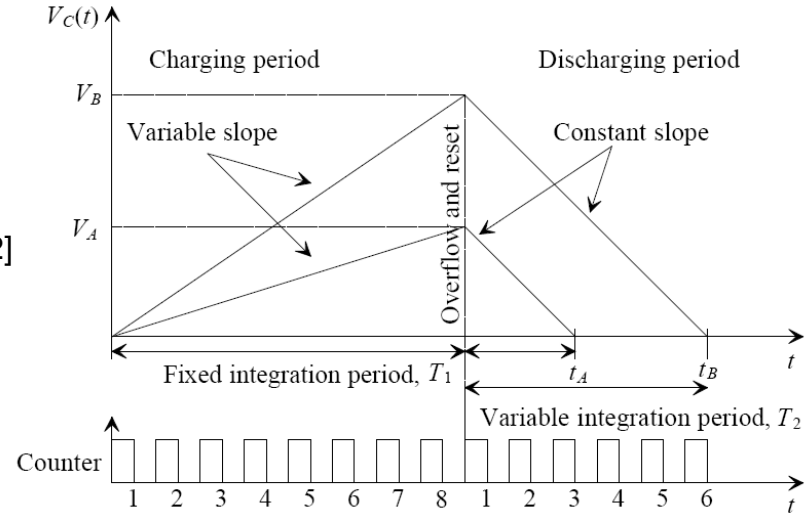


With a +50mV offset on the input of the comparator, the sampled input voltage looks larger by 50 mV. So V_C will ramp to ~50mV higher than the input, resulting in an extra clock period being counted and the output 1 LSB higher than the input. What happens if there are offsets of different polarities on the op-amp and comparator? What if the polarities are the same? (Hint: Think about whether the offsets make V_C and/or v_{IN} appear larger or smaller). Using the simulation Singleslope_IntegratingADC_offset.asc, investigate effects from offsets on the op-amp and sample and hold.

Dual-slope ADC Operation



[2]

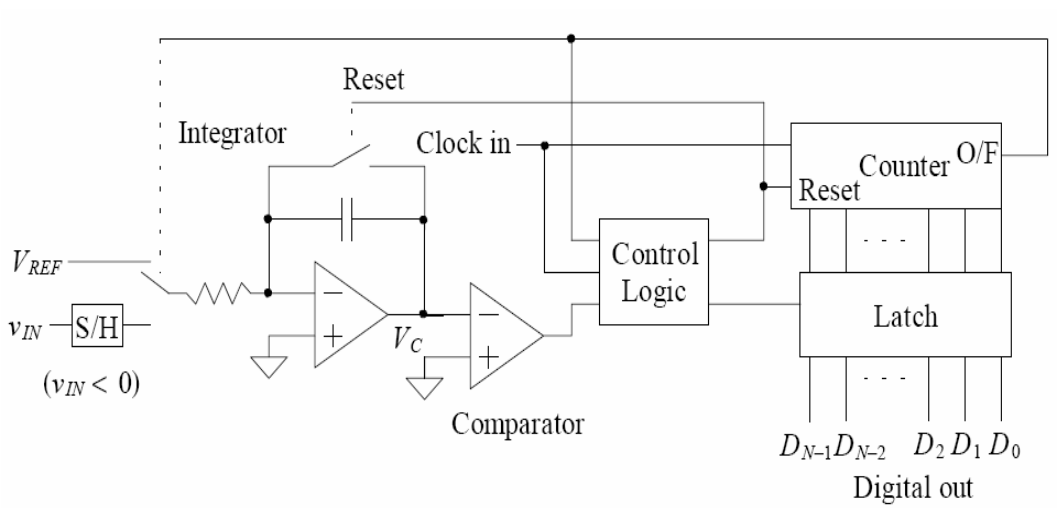


The dual-slope operation differs from the single-slope in that V_C is now compared to ground and two voltages, V_{REF} and v_{IN} are integrated. Initially a negative input is connected to the integrator, ramping V_C until the counter overflows. Since a negative value is being integrated on the inverting input, the integrator output will always positive and greater than zero, so the counter will continue until it overflows, which happens at 2^N clock cycles (= T_1).

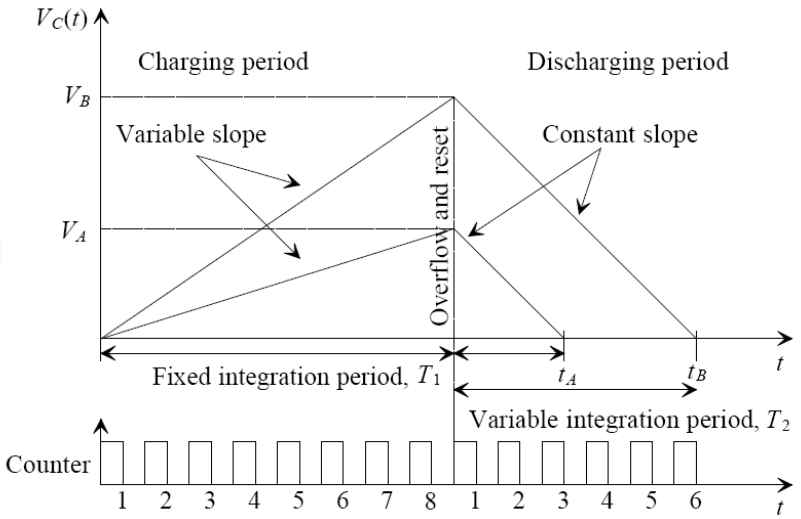
Equation (4) gives a value for V_C. Looking at the slope (V_{C1}/T_1), the dependence on v_{IN} is clear and since v_{IN} is a variable, so is the slope.

$$V_{C1} = \frac{-1}{C}\int_0^{T1} \frac{v_{IN}}{R}\,dt = \frac{v_{IN}\cdot T_1}{RC} \qquad \text{eq. (4)}$$

Dual-slope ADC Operation (cont.)



[2]



After overflow switches, V_{REF} is integrated; at the same time, the control logic triggers the reset to clear the counter. The counter now begins checking how many clock cycles are needed to ramp V_C down to zero. Equation (5) gives the value of V_C for this time period, T_2. Note, equation (5) is negative since V_{REF} is on the inverting input; this means that the integrator is now ramping down.

$$V_{C2} = \frac{1}{C}\int_0^{T2} \frac{-V_{REF}}{R}\,dt = \frac{-V_{REF}\cdot T_2}{RC} \qquad \text{eq. (5)}$$

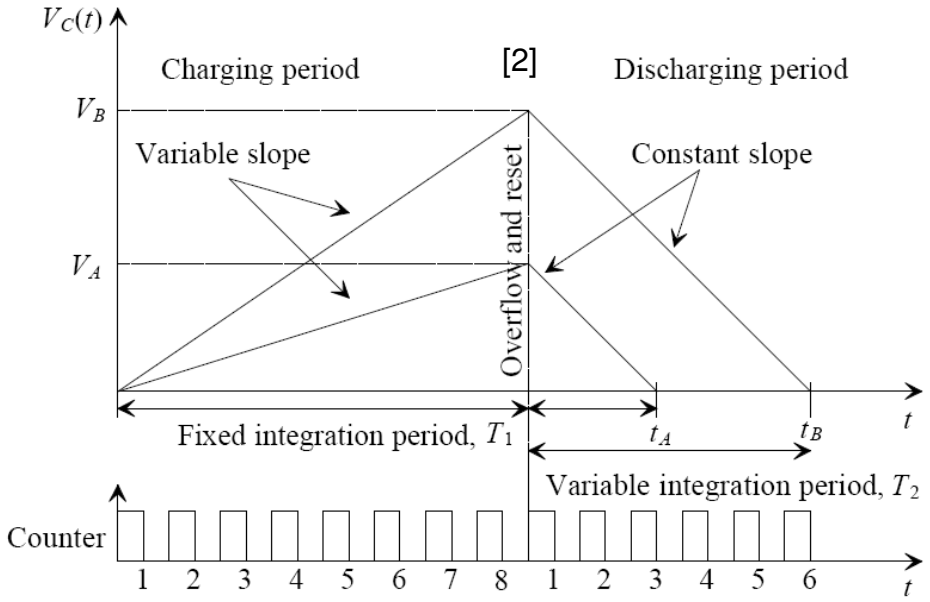
Again, looking at the slope (V_C/T_2), it is clear that it will be constant since V_{REF} and RC do not change.

Dual-slope ADC Analysis

The value of V_C after T_2 will be the sum of V_{C1} and V_{C2}, and as the illustration shows, it should be zero.

$$V_C = V_{C1} + V_{C2} = 0$$

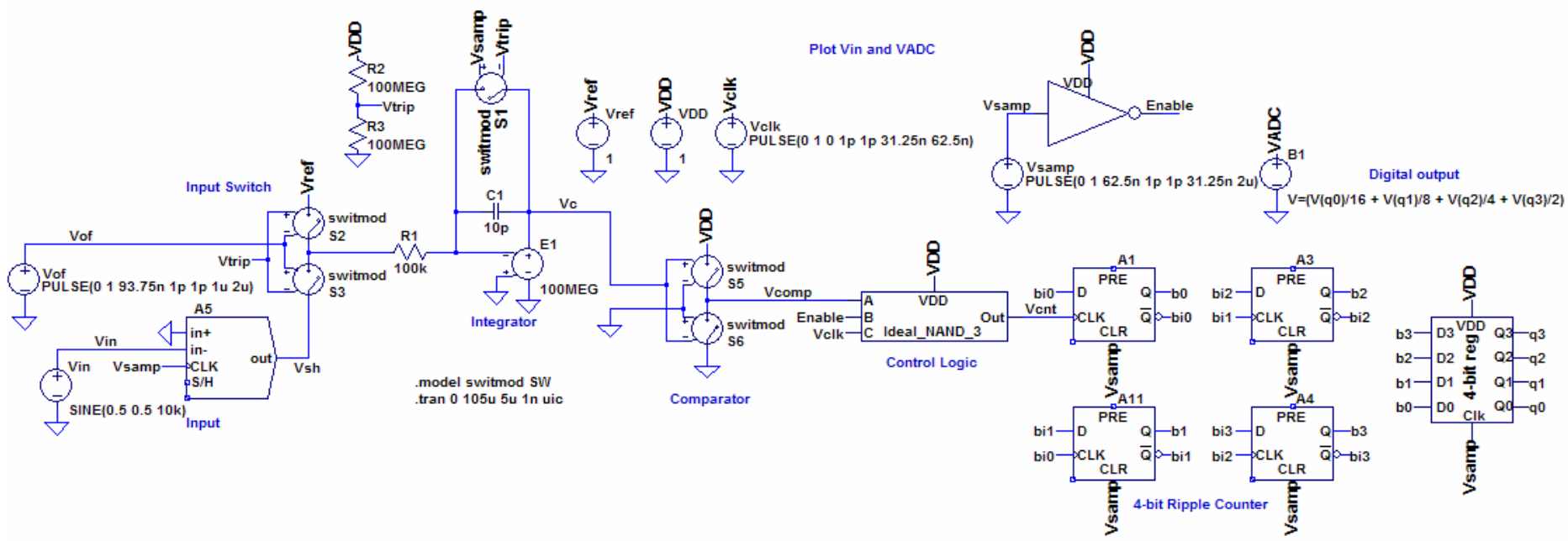
$$\rightarrow 0 = \frac{v_{IN} \cdot T_1}{RC} - \frac{V_{REF} \cdot T_2}{RC}$$



$$\rightarrow \frac{v_{IN} \cdot T_1}{RC} = \frac{V_{REF} \cdot T_2}{RC} \rightarrow \frac{v_{IN}}{V_{REF}} = \frac{T_2}{T_1} \qquad \text{eq. (5)}$$

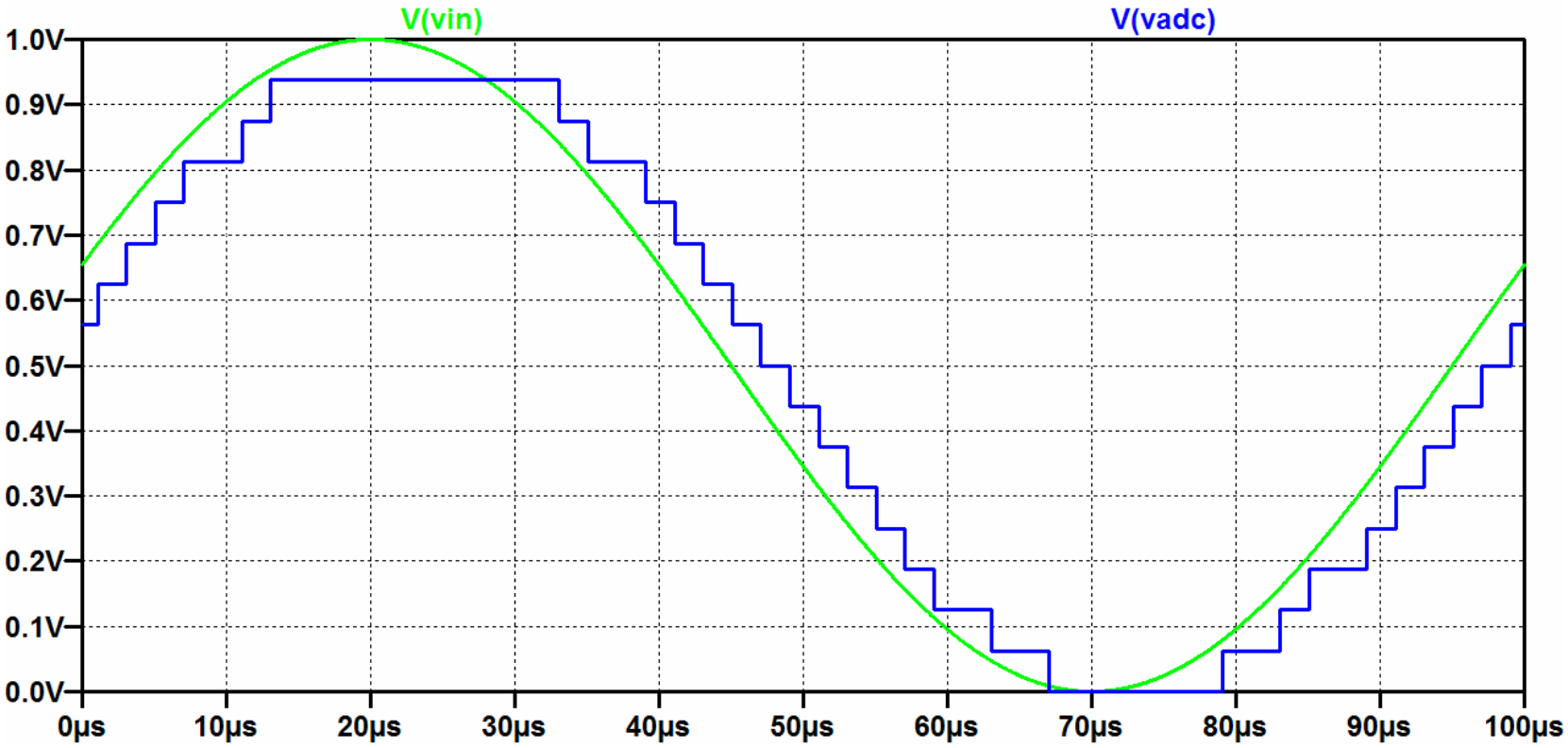
T_2/T_1 gives the digital output of the input voltage. Equation (5) is revealing in that there is no dependence on RC. Essentially, errors in the slope of V_C (determined by RC) will cancel out since the same integrator is used to calculate T_1 and T_2. Also, since the same clock is used to measure these times, errors in the clock (such as jitter) should also cancel out. So there are fewer sources of errors in the dual-slope than in the single-slope ADC. The cost of course is time; in the worse case there will be 2^N cycles in T_1 and T_2, so the total time will be 2^{N+1}.

Dual-slope ADC Implementation



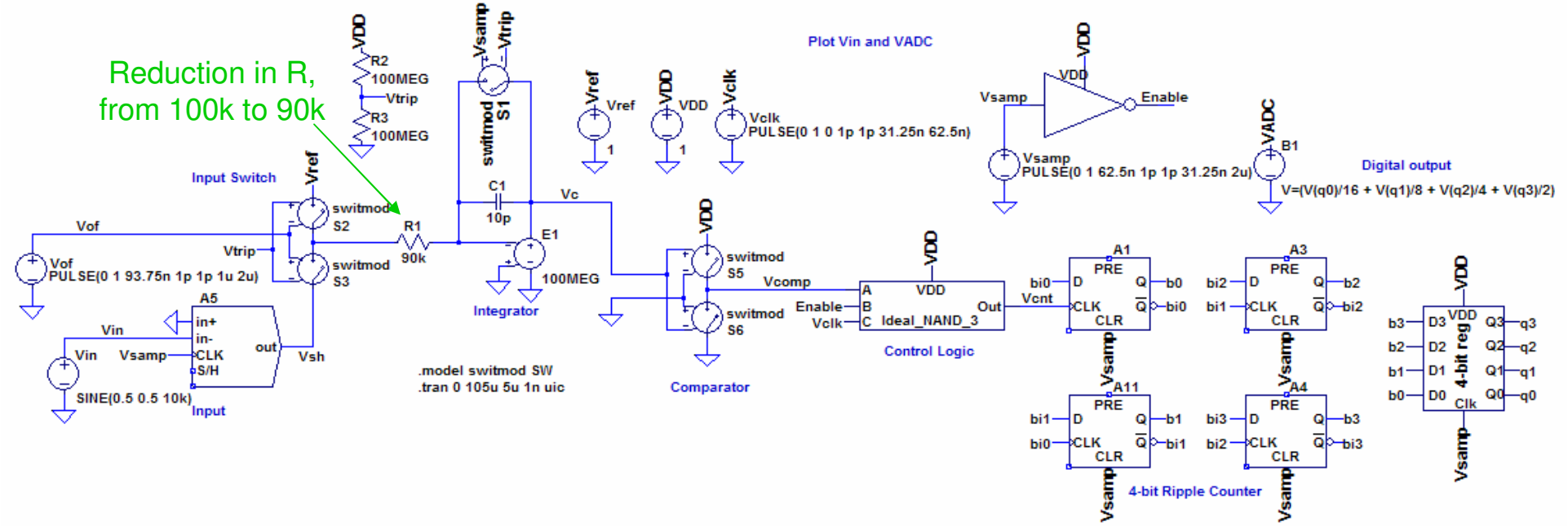
This schematic shows an implementation of a dual-slope integrating ADC. It behaves essentially as described before, with just a couple minor adjustments. First, the overflow is modeled as a pulsed voltage source rather than adding another DFF in the counter. After 1us (or 2^N clock cycles), it switches from a negative input (notice the input is on the 'in-' S/H terminal) to a positive V_{REF}. As in the single-slope implementation, the counter is not reset by the control logic; instead, the control logic stops the counter and at a *pre-defined* time interval (in this case 2us), V_{SAMP} resets the counter, latches the output and resets V_C to zero.

Dual-slope ADC Simulation



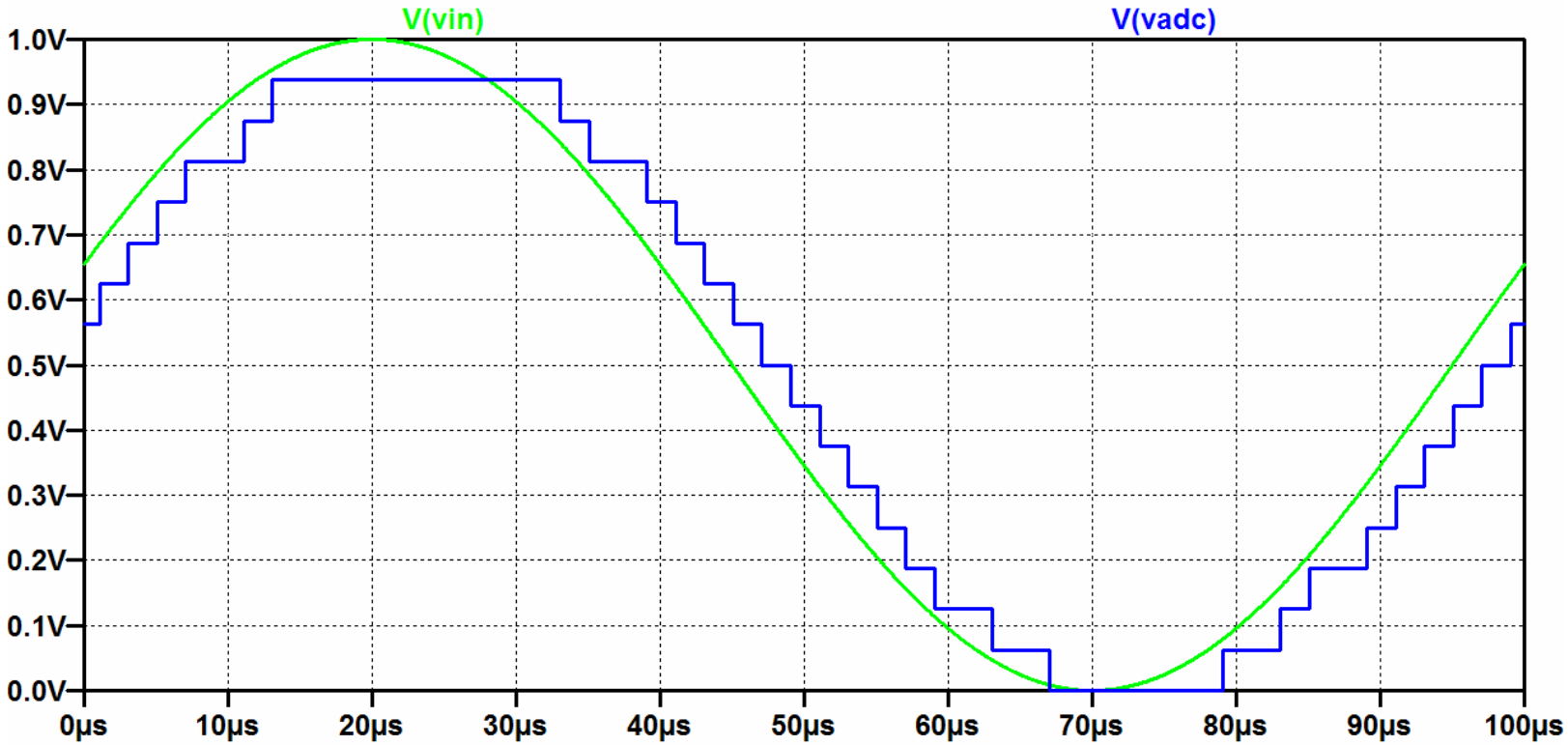
The waveform shows the analog input and the ADC output. Notice there is a delay on the output, VADC. This is due to the extra time needed for V_C to ramp up then down. The interested reader is encouraged to run (and investigate) the file Dualslope_IntegratingADC.asc.

Dual-slope ADC: Incorrect RC



As discussed previously, there are several issues with maintaining constant and accurate resistor and capacitor values. Ideally, the RC constant should equal the time period, T_1. However, the analysis confirmed that the dual-slope ADC accuracy is not dependent on RC. In the simulation Dualslope_IntegratingADC_RC.asc the resistor value is again decreased by 10% to 90 kΩ.

Dual-slope ADC: Incorrect RC (cont.)



Comparing this simulation with that on page 21 verifies the analysis; even with a slightly lower RC time constant, the output of the ADC does not change.

Unfortunately, the ability of the dual-slope ADC to reject errors is associated mostly with the *slope* of the integrator. Issues with offsets in the comparator or integrator can still cause problems by adding or subtracting a LSB from the output. The curious reader is referred to Dualslope_IntegratingADC_offset.asc to investigate.

Summary

- There are several types of integrating ADCs. In general, the integrating ADC can be of high-resolution and linearity, though they are relatively slow.

- The two ADCs, single-slope and dual-slope, had their advantages and disadvantages.
  - The single-slope ADC is faster than the dual-slope, with a worse-case conversion time of 2^N clock cycles. However, the output is subject to errors from the RC constant and the op-amp, comparator and sample-and-hold.
  - The dual-slope ADC has fewer sources of error than the single-slope since two integrations are performed. Errors associated with the slope of the integrator will cancel out. The higher accuracy comes at the price of time, where the worse-case conversion time would be 2^{N+1} (or twice that of the single-slope).